CONTROLLER DESIGN OF MULTIFUNCTIONAL PROSTHETIC HAND

MOHAMMAD ZYZERULL BIN SAAD

A report submitted in partial fulfilment of the requirements for the degree of Bachelor in Electrical Engineering (Control, Instrumentation & Automation)

Faculty of Electrical Engineering
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2016

Supervisor’s Declaration

“I hereby declare that I have read through this report entitle ‘Controller Design for Multifunctional Prosthetic Hand’ and found that it has comply partial fulfilments for awarding the Degree of Bachelor of Electrical Engineering (Control, Instrumentation and Automation)”.

Signature:
Supervisor’s Name: Dr Rozaimi Bin Ghazali
Date:

Student’s Declaration

“I hereby declare that this report entitles ‘Controller Design for Multifunctional Prosthetic Hand’ is result of my own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature:
Student’s Name: Mohammad Zyzerull Bin Saad
Date:

This thesis is dedicated to my father, who taught me that the best kind of knowledge to have is which is learned for its own sake. It is also dedicated to my mother, who taught me that even the largest task can be accomplished if it is done one step at a time.

ACKNOWLEGMENT

In preparing this final year report, I was contact with many people. First of all, I would like to express my gratitude to Dr. Rozaimi bin Ghazali to his willingness to spend his precious time to guide me throughout our final year project and provide with useful information to complete the final year project. Besides that, I would like to express sincere appreciation to my family members for everything that have provided such as money and moral support to help us in accomplishing this project.

Moreover, I would like to express sincere appreciation to Universiti Teknikal Malaysia Melaka (UTeM) because of giving me an opportunity to accomplish this final year project which required us to think creative and critically.

Not to forget my course mates that had given me many precious ideas to generate conceptual design for this final year project. Their opinion and ideas are useful for me and the whole project really brought us to the true value of friendship and respect to each other.

ABSTRACT

Prosthetics hand is replacement of original hands that lose or damage because of war, trauma, accident or congenital anomalies. However, problems often occur on a prosthetics hand when dealing with the control capabilities and devising functional. Thus, an advanced with control approach is required to improve the performance in terms of quality control in prosthetics hand and also enhance existing capabilities to the optimum level. This project presents the development of an enhanced prosthetics hand since the conventional prosthetic hand are incapable to functioning as a real human hand. The main objective of this project is to create a functional prosthetics hand at upper limb, which will focus on position of human hand particularly using the movement of finger instructions. In this project, an intelligent controller is proposed to realize accurate force control with high performance. A mechanical hand designed according to the physiological alignment of muscle and attempted to control the pre-recorded movement of an actual hand. In addition, performance of prosthetics hand can be assessed through the existing issue, whether is in visual or data. Firstly, the relationship and mathematical model of the position of flex signals using system identification technique will be developed. Then, Fuzzy-PID Controller is designed. Finally, evaluation on the effectiveness of the controller design will be conducted with the flex signal for the developed prosthetic hand. It is expected that the prosthetics hand will move and functional like a real hand using the flex signals and Fuzzy-PID Controller to track the desired movement of finger in prosthetic hand. The fabrication and design development of the prosthetics hand will solve the problem for people with disabilities to continue their routine life on grasping an object with an appropriate position. As a conclusion, the Fuzzy-PID controller that has been developed to control the performance of prosthetic hand model is outperform better controller than PID and Fuzzy controller, where it’s improve the transient response and steady state error especially eliminate the overshoot and higher accuracy based on comparison of performance for the three different controllers.

ABSTRAK

Tangan prostetik adalah penggantian tangan asal yang disebabkan kehilangan atau kerosakan fizikal kerana kesan dari perang, trauma, kemalangan atau kecacatan kongenital. Walau bagaimanapun, kebiasaanya masalah sering berlaku pada tangan prostetik yang sedia ada terhadap keupayaan kawalan dan keupayaan merangka. Oleh itu, kemajuan terhadap pendekatan kawalan amat diperlukan bagi meningkatkan prestasi dari segi kawalan untuk tangan prostetik dan juga meningkatkan keupayaan sedia ada ke tahap yang optimum. Projek ini membentangkan pembangunan bagi tangan prostetik yang lebih efisien seperti tangan manusia sebenar. Objektif utama projek ini adalah untuk mewujudkan tangan prostetik yang pelbagai fungsi pada anggota badan yang akan memberi tumpuan kepada kedudukan pada tangan manusia terutamanya menggunakan arahan isyarat Flex. Dalam projek ini, pengawal pintar dicadangkan bagi merealisasikan ketepatan pada kawalan kedudukan pada tahap yang optimum. Satu tangan mekanikal direka mengikut penjajaran fisiologi otot dan cuba mengawal pergerakan yang telah dirakam daripada tangan yang sebenar. Selain itu, prestasi tangan prostetik boleh dinilai melalui maklumat yang sedia ada sama ada dari segi visual atau data. Projek ini mewujudkan hubungan dan matematik model berdasarkan kedudukan isyarat Flex menggunakan teknik pengenalpastian system. Kemudian, Pengawal Fuzzy-PID direka. Akhir sekali, penilaian ke atas keberkesanan reka bentuk pengawal akan dijalankan dengan menggunakan isyarat Flex terhadap tangan palsu. Tangan prostetik akan berfungsi seperti tangan sebenar menggunakan Flex dan Pengawal Fuzzy-PID untuk mengesan daya menggenggam tangan berdasarkan dikehendaki oleh pengguna. Pembuatan dan reka bentuk pembangunan tangan prostetik yang akan menyelesaikan masalah untuk orang kurang upaya untuk meneruskan kehidupan rutin mereka di menggenggam objek dengan kedudukan yang sesuai. Secara keseimpulan, pengawal Fuzzy-PID digunakan untuk mengawal prestasi model tangan palsu merupakan pengawal yang lebih baik daripada pengawal PID dan pengawal kabur di mana ia meningkatkan sambutan fana dan ralat keadaan mantap terutamanya dapat menghapuskan terlajak dan ketepatan yang lebih tinggi berdasarkan perbandingan prestasi bagi tiga pengawal yang berbeza.

TABLE OF CONTENT

PAGE

CHAPTER

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Recent years, the number of cases requiring assistance in daily life has been increased for physical disability. The physical disability is the people who suffer from permanent disability or loss of a limb or limbs as a result of disease, trauma, accident, disability congenital and so on. For the activities and movements of daily life, they will facing many difficulties and hardships [1]. Therefore, most of them need help from their family's members both in terms of physical or mental support. However, some patients who have lost limbs had the desire to do anything without problems and they expect to rely on prosthetics hand to help them in their daily lives. Prosthetics devices are the tools that commonly used in medicine in the era of technology and it is quite easy for patients who are fitted with artificial limbs. Advances in prosthetic devices should be expanded aggressively where many studies that need to be improved so that the development of robotic artificial limbs can achieve great advances and their availability at a high level [1], [2].

The development of prosthetic hand is a very demanding endeavour which is essential to support the daily activities of amputated people. Design criteria for prosthetic hands are cosmetics, comfort, controllability and also low power consumption. The purpose of a prosthetic hand control system is to interpret the limited set of control signals available to the user in a way that maximizes the amount of available actions without requiring too much of the user's attention. The function of controller also affected in the design of prosthetic hand where efficiency of prosthetic hand which must attain a real ability of human hand [2].

One type of control, namely fuzzy control has appeared as one of the most popular controller explored and means of research in industrial process where they do not depend on traditional techniques. Fuzzy logic used in the fuzzy control logic system which had a closer link with human thinking and natural language. Fuzzification, defuzzification strategies and fuzzy control rules are used in fuzzy reasoning mechanism. Various fields of application ranging from automatic train operation system shows the cost effectiveness of using fuzzy logic control. Control engineers can use fuzzy logic to develop control strategies in the application area characterized by dynamic low- order non- linearity is poor.

However, the control approach contains weakness due to lack of quantitative data on the input and output relationship. Besides that, it is difficulty to develop fuzzy rules and also membership functions and fuzzy output can be interpreted in several ways to make the analysis difficult. In addition, it must expertise to develop a fuzzy system and requires lot of data. It does not deliver the results not convincing and programs have been conducted for each individual patient. Therefore, clinical applicability and use of the software is difficult without pre - programmed for different pathologies and clinical basic training to use the program.

To optimize a particular control system, many industrial implement a proportional-differential-integral (PID) controller arrangement can be adjusted. For controller design, PID is the easiest design controller compared to existing controllers. Thus, PID is the most commonly used controller in industry. Addition, many industries use controller either PID or improved version PID controller. The serial controller, parallel controller and mixed controller are basic type of PID controller. The design velocity algorithm, which also known as incremental algorithm used PID controller algorithm. In the industry, PID controllers are the most common control method to use in real applications [3].

There are many unique elements and advantages of fuzzy logic controller and PID controller over another controller such as PD controller and PI controller. The main advantages of fuzzy logic controller it is intuitive knowledge base design and flexibility. Consistency, redundancy and completeness also part of validation can be checked in rule bases. PID controller has all the necessary dynamics: proportional gain controls the effect of reducing error and derivative gain control the effect of reducing the overshoot. While for integral gain act to improving the transient response and increasing the stability of the system [4]. However, PID controller, it has some drawbacks, when unique processes are required to perform the task when PID controllers are significantly limited in their capabilities. PID controller is capable of measuring a variety of input and calculates the difference between them.

To reduce vulnerability in existing controls on the PID controller and Fuzzy controller where the combination between the two controllers was created to accommodate all the weaknesses that exist. In addition, it also allows them to control the prosthetic hand directly and easily. Therefore, patient can to do things with their own wishes and also act alone independently. Now, it is interesting to know the detailed parts of the controller either fuzzy logic or PID controller and study the method to be used for tuning in the same time and then apply this method during the process of designing controller for prosthetic devices.

1.2 Motivation

Prosthetics hand developed widely in medical field with many several of implementation as the controller. Many types of controllers available for use to control the performance of the system such as intelligent controller. Intelligent controller is combination of one or more controller such as Fuzzy Logic, Artificial Neuron Network and other. While, controller commonly used in industrial such as PI, PD, PID and so on. For this research, combination of Fuzzy Logic and PID controller that are used that will affect the result of this analysis. Without a proper or a well-developed of system, the control of prosthetic hand will either have disturbances on great amount of which leads to the significant effect in the performance of the system.

1.3 Problem Statement

It has been addressed in much research that the causes of the dissatisfaction are low cosmetic value, functionality and controllability of prosthetics hand [4]. Research areas regarding prosthetic control in general are aimed at:

I. Controller design: Most embedded controllers of position control are used separately on prosthetics hand.
II. Processing these signals to get information out of them: Number of researchers has used simple classifiers focusing on simplicity and robustness while others use methods involving fuzzy logic and neural networks to extract information from voltage signals [4].
III. Performance of controller: Performance of controllers are differ based on the type of controllers.

1.4 Objective

Purpose of this project is to create a prosthetic hand at upper limb, which will focus on human hand in position movement particularly using the flex sensor based finger instructions. Below are the objectives that must to achieve to complete this project:

I. To develop a functional prosthetics hand hardware.
II. To design an intelligent controller that capable to control the position by using the combination of Fuzzy controller and PID controller.
III. To evaluate the performance of PID controller, Fuzzy Logic controller and Fuzzy-PID in the prosthetics hand in simulation.

1.5 Project Scope

This project will focus primarily on the controller of prosthetic hand and has a limitation during develop this project. Below are the limitations that be applied in this project:

I. The design of prosthetic hand will be focused on one fingers only.
II. The data from flex system will be collecting at the prosthetic finger.
III. Data input from the flex signal must be a voltage signal to a reference data.
IV. In the mechanical design of prosthetic hand, CAD software will be utilized.
V. MATLAB software will be implemented in collecting the data and controller design process.
VI. A combination of fuzzy logic and proportional-integral-derivative (PID) controller will be used to control the feedback and minimize the error.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter contains the overview of prosthetic hands, PID controller, Fuzzy logic controller, Fuzzy-PID controller and the overview of the research that related to the project**.**

2.2 Prosthetic Hand

This part explained about the main components related with prosthetic hand for this research: development of prosthetic hand and selection of tool.

2.2.1 Development of Prosthetic Hand

An electromyography method (EMG) is a medical method that rendered by the medical field to measure muscle reaction to anxious stimulation. Electromyograph instrument is used to perform EMG techniques to produce the record of the so-called electromyograph. An electromyograph sense the electrical signal generated by muscle cells when these cells tense.

A myoelectric prosthetics hand using EMG signals or possible of voluntary muscle contraction in the rest of one's body to the skin to control the movement of the prosthesis, such as elbow flexion, open hand, finger cap or wrist supination/pronation (rotation). This prosthesis using a neuro-muscular system of the human body to manage the rest of the functions of an electric-powered prosthetic [5]. This contrasts with the switch electric prostheses, which requires ropes and cables driven by the movement of the body to move or operate a switch that controls the movement of the prosthesis or a completely mechanical. It has a jack hang person by taking electrodes placed on the flexors and extensors of extension and extension movements respectively. Advances in technology myoelectric in current years has made it the top end of forged components are far superior to the governing body equivalent [2].

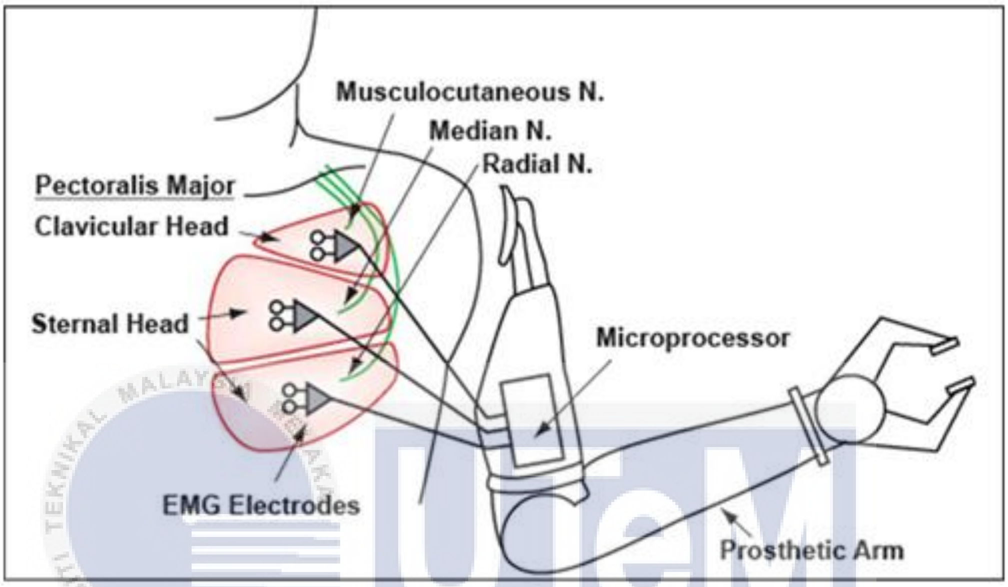


Figure 2.1: Electromyography Method in Prosthetic Hand

In Figure 2.1 shows pectoral muscles have the four main brachial nerves (musculocutaneous, median, radial & ulnar) after be transferred where it are little functional use to a person with shoulder disarticulation. After the nerves are communicate to distinct regions of pectoral tissue by anastomosis, conventional surface electrodes can monitor these restored sources of neural signals that now correspond to musculature that would have been distal to the amputation (elbow, wrist, and hand).

The movement of a prosthetic arm requires amenities control system for controlling the movement consists of three main parts: the input signal, motor driver part and motor control algorithms. Part processing the input signal include real-time data and also its buffer part. Additionally, it measure electromyograph signals from the hands of the forearm muscle subject and generate a stable flow of root mean square (RMS) value [1]. The next movement of the arm can be determined by using the range of RMS value. After that, motor control will be determine the motor movement based on the range of RMS value and device driver acts to control the performance of motor. By using correct selection of threshold and fitness level, any type of exercise can be successfully trained in the application.

2.2.2 Selection of Tool

The selection tool is a very important component to complete a project or to obtain more specific data based on hardware physical model. Selection of tool should be selected properly to expedite the process of getting data running smoothly. Thus, instruments like servo motor, flex sensors, Arduino and others are also accounted in this research. The use of servo motors in the prosthetic hand industry is particularly widespread. Characteristic in automatic control of DC servo motor has played a key role in the electromechanical engineering world. For sensor, it plays a crucial role in robotics where it implicated in the special aids such as prosthetic hand. Besides that, sensors is also used to measure the current state of the system. Robotic industries require good sensors with high degrees of precision, reliability and repeatability. One of the sensors used in industry, namely, flex sensor is such an instrument, which accomplish the above task with great degree of accuracy. The movement of position finger in robotic hand can be efficiently controlled using microcontroller. This work is an educational based concept as robotic control is an exciting and high challenge research work in recent year.

2.2.2.1 Servo Motor

Servo motor is a rotary actuator or linear actuator that produce specific control of three term: angular or position, velocity and acceleration. Thus, servo motor consists of a DC motor coupled to a sensor for position feedback. In addition, the servo motor also requires a fairly sophisticated controller which is designed to be used with servomotors for achieving better performance.

Figure 2.2: Servo Motor

2.2.2.2 Power Supply

Power supply is a key element in the development of various type of electronic applications. Power consumption in the servo motor requires stable voltage to move the servo motor. In prosthetic hand, it used 5 servo motor with the capacity of 6Kg/cm where each servo motor consume approximately 1Amp at 5Volt supply. Thus, the consumption of 5-6 Amps will be needed for 5 servo motor. So the challenge in the design of servo motor is to design a larger power supply that can deliver enough current without effecting the servo motor.

2.2.2.3 Servo Driver

Servo driver is a heart of the servo motor. Servo motors require continuous pulse width modulation (PWM) signals to achieve continuous angle settings. Besides that, it requires microcontroller circuit to control the servo simultaneously. Capability of microcontroller in servo controller that can generate constant PWM pulse that will set need desired angle without affecting any other motor in the same time. The PWM pulse should also be continuous angle else it can make the angle sustainability will get loose cause the servo motor moving not properly.

2.2.2.4 Flex sensor

Flex sensor in other word known as bend sensor. This sensor capability is to sensing any kind of minute bend in its structure. Flex Sensor that be produced achieves better form-factor on a thin flexible substrate. This sensor produces a resistance output correlated to the bend radius when the substrate is bent. Theoretically, the smaller the radius then the higher the resistance value. This carbon layer is divided into small sections and connected together in series by conductive layer.

Figure 2.3: Flex Sensor

2.2.2.5 Arduino

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. Thus, Arduino hardware is an open-source circuit board with a microprocessor and input/output (I/O) pins for communication and controlling physical objects. The board will be typically be power-up via USB or an external power supply which enough supply provided to other hardware, sensors and other component. Arduino also has an open-source software interface which is similar to C++. This interface that called (IDE) allows users create the coding, compile, and then upload it to Arduino for standalone use in prototyping and projects. Many design of Arduino hardware that be developed from Arduino team. One of them is current version of the Arduino hardware reference design is called the “Mega”. It provides four basic functional elements:

- An Atmel “ATmega2560” AVR microcontroller
- Power supply for 5V or 3.3V.
- A USB-to-serial converter for loading new programs onto the board
- I/O headers for connecting sensors, actuators, expansion boards, etc

Figure 2.4: Arduino Mega

2.2.2.5.1 Power (USB / Barrel Jack)

Every Arduino board needs a way to be connected to a power source. The Arduino UNO can be power-up from a USB cable coming from the computer or a wall power supply that is terminated in a barrel jack. Average of power consumption in Arduino is less than 20 Volts. The recommended voltage for most Arduino models is between 6 and 12 Volts.

2.2.2.5.2 List of Pin

Each pin on the Arduino had different functions. In addition, there is the Arduino pin that is where the wire is connected to a circuit that may build together a breadboard and some wires. Each pin can be used as input and output to complete the circuit.

GND: Abbreviation for ‘Ground’. There are several GND pins on the Arduino, any of which can be used to ground the circuit.

- **5V & 3.3V**: For 5V pin supplies 5 volts of power, and the 3.3V pin supplies 3.3 volts of power. Most of the simple components used with the Arduino running between of 5 or 3.3 volts.

- **Analog**: Pins under the ‘Analog In’ label (A0 through A9 on the Mega2560) are Analog In pins. The usefulness of this pin is to read the signal from an analog sensor such as humidity sensor and convert it into a digital value that can be read.
- **Digital**: Pin facing the analog pin is pin digital (0 to 48 in Mega2560). Furthermore, this pin can be used for both digital inputs such as tell if a button is pressed and digital output for powering LEDs and others.
- **PWM**: Referring to the symbol tilde (~) next several digital pin (3, 5, 6, 9, 10, and 11 on Mega2560). In addition, this pin acts as a normal digital pin, but can also be used for pulse width modulation (PWM)
- **AREF**: Refers for Analog Reference. This pin is rarely used and it can be left with just that. In addition, it is sometimes used to set the external voltage (between 0 and 5 volts) as the upper limit for the analog input pin.

2.3 PID Controller

This part explained about component that related with PID controller for this research: overview of PID controller and PID tuning method.

2.3.1 Overview of PID controller

In industrial control systems, the PID controller is the frequently feedback controller used such as automation industry. Function of PID controller is to calculate the error of the system between the measured and a desired set point of the system [6]. For reducing error by changing the controller gains which are is called as K_p, K_i and K_d. In the controller field, the PID controller are effective controller where it can affect the steady state error (SSE) and transient response. But, for the best performance of PID controller, appropriate PID parameters should be tuned rigorously for particular system [7].

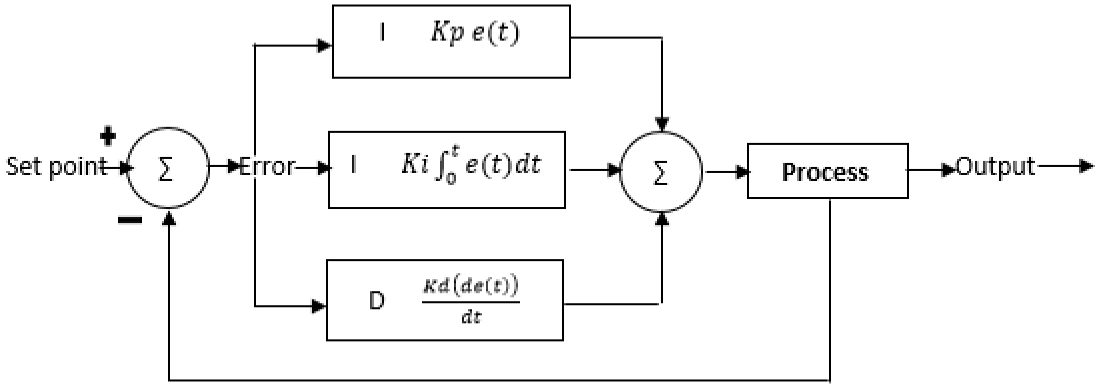


Figure 2.5: Block Diagram of Proportional-Integral-Derivative (PID) Controller

The PID controller algorithm contains three different variables which are called as three-term controller. A proportional gain acts to identify the response to the error of the system and for the integral gain is to identify the response based on the summation of current errors changed. To determine the response based on the rate by adjusting the derivative gain. The entire weight of these three-term controller is used to adjusting the process via a feedback of control variables or an actuator such as movement of a control valve, position and speed of the robotic hand and other [8], [9].

Theoretically, for the P value depends directly on error, I value expect on the reserve of previous errors and lastly, D value is a forecast of future errors of the system based on present rate of change. A PID controller will be called a PI, PD, P or I controller in the absence of the respective control actions. The proportional action, it is assigned to make changes to the output is proportional to the present error [8]. A value of K_p make the proportional response to be adjusted. Integral action is also known as reset because their contribution which is proportional to both magnitude and duration of the error. The accumulated Offset accumulated previously can be corrected by integrating the error. For cumulative error, it is multiplied by the integral gain and then this error will add to the controller output of the system. To get magnitude of the integral term to the entire control system by determine of the integral gain, K_i. By calculating the rate of change of the process error by using proses of identify the slope of the error over time and derivative gain K_d used to multiply the rate of change. By term of derivative gain, K_d, data of the magnitude of the contribution of the derivative action to the overall control action will be obtained [9].

To calculate the output of the PID controller, the proportional, integral, and derivative terms shall be added. Defining u(t) as the controller output, the final form of the PID algorithm is:

$$u(t) = Kpe(t) + Ki\int_{0}^{t} e(t)\,dt + Kd\left(\frac{d}{dt}\right)e(t)$$

Where the parameters are tuning:

u(t) = Output

e(t) = error

K_p = Proportional gain

K_i = Integral gain

K_d = Derivative gain

The controller can implement control action designed for specific process requirements by through the process of tuning the three gains in the PID controller algorithm namely K_p, K_i and K_d. In terms of the responsiveness of the controller to an error, the degree of system oscillation and the degree to which the controller overshoots the set point, a response of the controller of the system can be treated [10].

Table 2.1: The Effect of Tuning the Three Gain

Gain	**Effect**
Proportional gain, K_p:	Larger value in K_p means response is faster since error is larger and likewise the proportional term composition. Process instability and oscillation due to extremely large proportional gain.
Integral gain, K_i:	Larger value in K_i act to eliminates the error faster. However, the effect seen in bigger overshoot: any negative error in system mixed during transient response must be to the process of integrated away to positive error before achieve steady state.
Derivative gain, K_d:	Larger values in K_d act to decrease the overshoot, but make the transient response is slow down and maybe stability of system is not stable due to signal noise amplification.

2.3.2 PID Tuning Method

The PID controller tuning methods are separated into two main categories: open loop methods and closed loop methods. For open loop methods refer to methods that tune the controller when it is in manual state and the plant operates in open loop. While, closed loop tuning methods refer to methods that tune the controller during automatic state in which the plant is operating in closed loop.

There are several methods available for tuning the PID loop control. Some PID tuning method is only effective on certain models involves some form of model development process and thus the selection of P, I, and D based on the model parameters. Almost PID tuning methods can be done by manually considered ineffective where it involve the performance of system has a reaction time [11].

Table 2.2: Overview between Different Tuning Techniques

Choosing a tuning method		
Method	**Advantages**	**Disadvantages**
Manual Tuning	No math required. Online method.	Requires experienced personnel
Ziegler-Nichols	Proven Method. Online method.	Process upset, some trial- and-error, very aggressive tuning
Software Tools	Consistent tuning. Online or offline method. May include valve and sensor analysis. Allow simulation before downloading.	Some cost and training involved.
Cohen-Coon	Good process models.	Some math. Offline method. Only good for first-order processes.

2.4 Fuzzy Logic

This part explained about component that related with Fuzzy controller for this research: overview of Fuzzy controller and Fuzzy tuning method.

2.4.1 History of Fuzzy Logic

In 1965, Lofti Zadeh is a professor at the University of California has inspired the concept of a method called Fuzzy Logic (FL). However, this study was not presented as a method of control, but as a way to process the data by allowing some of the skill set fresh set of membership or non-members. Until the '70s, this approach to theory implies that the control system due to lack of capacity towards computer at the time. Professor Zadeh said that people do not need the right data and input numerical information, but they were able to control the highly adaptive. In such situations there is a feedback controller, which can be programmed to act to receive noisy and imprecise input, the control will be more efficient and may be more easily implemented. However, this method has not been received with so extreme for US manufacturers than Europe and Japan are implementing aggressive to build real products in the vicinity [12].

2.4.2 Theory of Fuzzy Logic Controller

Figure 2.3 shows a block diagram of a simple Fuzzy Logic Controller (FLC). For Fuzzy Logic Controller (FLC) contains four main components comprising the fuzzification, fuzzy knowledge base, fuzzy inference engine and defuzzification.

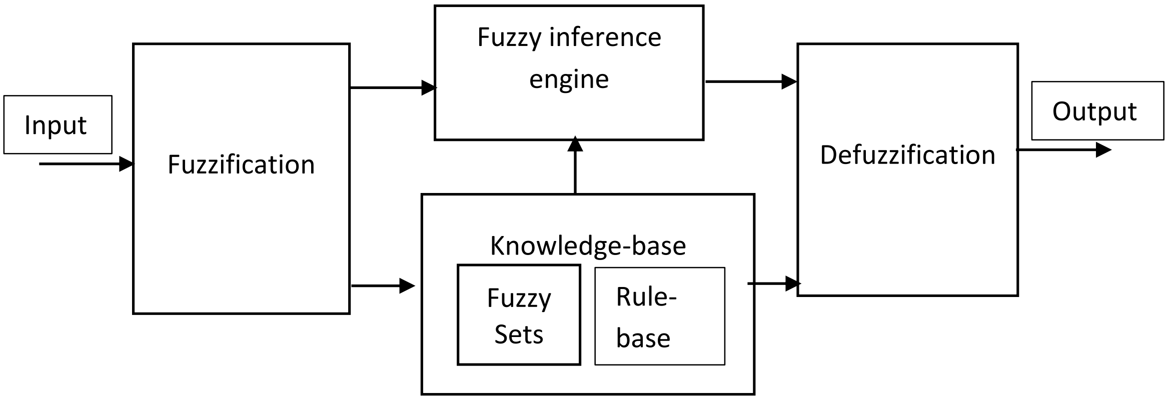


Figure 2.6: Fuzzy System Structure

2.4.3 Overview of Fuzzy Logic in Prosthetic Hand

Fuzzy logic control (FLC) using concept of if-then to provide a mathematical framework to allow for model uncertainties associated with estimates of reasoning, over to the control system in which a mathematical model is difficult to obtain, including information processing and human perceptual. For fuzzy system, fuzzifies input starting to Using a set of membership functions (MF), the fuzzy system, a prelude to fuzzifies input to the value of the interval [0,1] which is 'HIGH' and 'LOW'. In three pattern recognition of myoelectric signals, some researchers have used concept of Fuzzy Logic for hand prosthesis [13]. The fuzzy logic controller (FLC) consists of three main components: fuzzification, defuzzification, fuzzy inference engine and role of fuzzy rule base is to allow the process of uncertainty.

Such information can be disclosed to other forms of linguistic imprecise as 'low', 'medium' and 'high'. In a fuzzy system, the subjective judgment formulated by the fuzzification acting to transform them into fuzzy linguistic variables according to functional characteristics of expertise in a certain universe of discourse. For Fuzzy Logic (FL) pattern recognition techniques, it has been used to train three differ output patterns, with a correctly classified rate above 85% [25]. Based on earlier studies, six myoelectrodes collected sEMG raw signal and the three differ input features to the Fuzzy Logic have 63Hz, 125Hz and 250Hz band. All three patterns of output were lateral grasp, palmar 3-finger grasp and hook grasp. In a previous research work in [14] the similarities and differences of sEMG signal have been measured for six major grasping patterns of the human hand, such as lateral, cylindrical, palmar, hook, spherical, and tip. To collect myoelectric raw signals of six patterns by using twenty needle myo-electrode sensor where it is insert into in ten extrinsic hand muscles. To assess the extent of overlap electromyographically holding pattern, a method has been used that called Fuzzy c-means (FCM).

In biomedical signal processing and classification, Fuzzy Logic system is better controller be used [14], [15]. For biomedical signal is received, it is not always strictly repetitive and may sometimes be contradictory. Additionally, when using the system to escape easily trained, to find patterns in data that are not easily detected with other methods are not possible, as is also possible with neural networks. Most advantage of Fuzzy Logic over ANN is that in Fuzzy Logic system it is possible to integrate this incomplete but valuable knowledge into the fuzzy logic system due to the system's reasoning style, which is similar to the human decision-making and more closely than ANN [15]. Figure 2.7 shows the flowchart of tuning Fuzzy Controller.

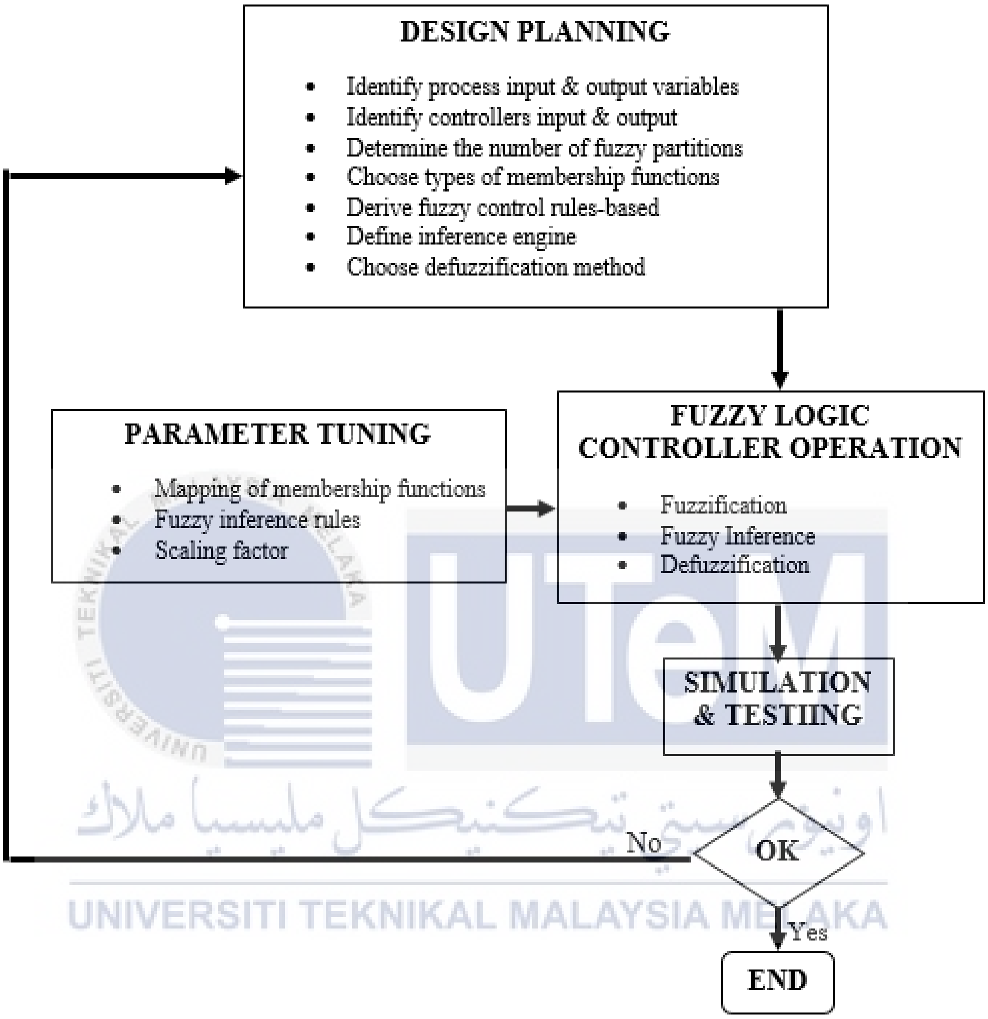


Figure 2.7: Flowchart to Design FLC

2.5 Fuzzy-PID Controller

This part explained about component that related with Fuzzy-PID controller for this research: overview of Fuzzy-P controller and Fuzzy PID controller.

2.5.1 Proportional Gain

In block diagram, input to a Fuzzy controller called error and while the output is the control signal. Figure 2.8 shows the block diagram proportional gain in Fuzzy Controller. This is the simplest fuzzy controller by using P only. It is relevant for state- or output-feedback in a state space controller. Crisp proportional control the fuzzy P controller can be compared with the usual fuzzy controller has two gains GE and GU instead of just one. As a custom, in lower case before gains and upper case after gains written by signal, for instance $E = GE * e$. Normally, gains are used to obtain for response of system, but since there are two gains, it can also be used for scaling the input signal onto the input. For GE and GCE are called the input scaling factors, while GU are called the output scaling factors.

The controller output is the control signal U_n, a nonlinear function of e_n

$$Un = f(GE * en) * GU \tag{1}$$

Function of variable f is the fuzzy input-output map. The linear approximation be used$f(GE * en) = GE * en$, then

$$Un = GE * en * GU = GE * GU * en \tag{2}$$

With the product of the gain factors be compared is equivalent to the proportional gain,

$$GE * GU = \mathrm{K_p} \tag{3}$$

Through the membership functions and the rules where accuracy of the approximation depends from there. In result, the approximation that be generate is best, however, if by choose the same universe on both input and output side, for example [-200 200]. The rule base

1. If J is Pos then K is 200
2. If J is Neg then K is -200

With Pos and Neg membership as defined previously, is equivalent to a P controller. Given a target K_p from the Ziegler-Nichols rules, this algorithm helps to select the gains. Since the fuzzy P controller has one more gain factor than the incisive P controller make the equation has one DOF.

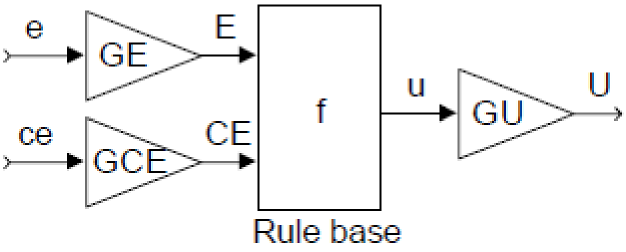


Figure 2.8: Fuzzy P Controller (FP)

2.5.2 Proportional, Integral and Derivative Control

In Fuzzy PID controller, this controller easily identifiable with three input term, error, integral error and derivative error. For three input term, it have the own gain. A rule base in fuzzy logic with three different inputs, however, easily becomes rather big and rules concerning the integral action are troublesome. Therefore it is common to separate the integral action as in the *Fuzzy PD+I*, (FPD+I) controller in Figure 2.9. Whereas, GE are scaling factor related to error, GCE are scaling factor related to change in error and GIE are scaling factor related to integral error respectively. For GU is output scaling factor for fuzzy PID based on Table 2.3.

The integral error is computed as,

$$ien = \sum_i (ei \ * Ts) \tag{4}$$

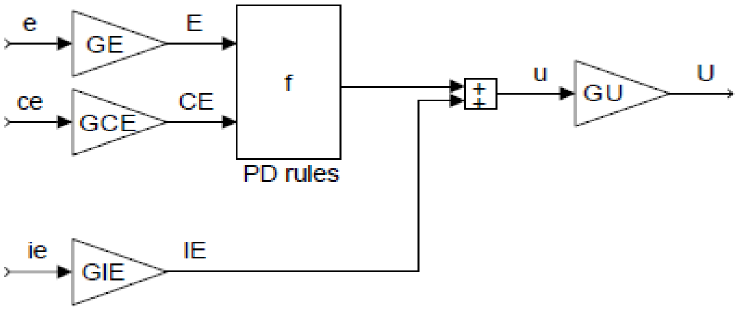


Figure 2.9: Fuzzy PID (FPD+I)

The controller is thus a function of the three inputs

$$Un = [f(GE * en), GCE * cen + GIE * ien] * GU \tag{5}$$

Its linear approximation is

$$Un = [GE \ * \ en + GCE * cen + GIE * ien] * GU \tag{6}$$

$$= GE * GU * [en + \frac{GCE}{GE} * cen + \frac{GIE}{GE} * ien]$$

In the last line we have assumed a nonzero GE.

$$GE * GU = K_p \tag{7}$$

$$\frac{GCE}{GE} = Td \tag{8}$$

$$\frac{GIE}{GU} = \frac{1}{Ti} \tag{9}$$

Table 2.3: Parameter of Fuzzy-PID Controller

Parameter	**Remark**
GE	Scaling factor related to error
GCE	Scaling factor related to change of error
GIE	Scaling factor related to integral error
GU	Output scaling factor
En	Error
i_{en}	Integral error
c_{en}	Change of error
K_p	Proportional gain
T_d	Derivative time
T_i	Integral time

This controller provides all the benefits of PID control, but also the disadvantages regarding derivative kick and integrator windup.

CHAPTER 3

DESIGN METHODOLOGY

3.1 Introduction

This chapter will describe the methodology of this project. Summary of the methods used in related works are show in Section 3.2. Section 3.3 is discussing the collecting data. While, Section 3.4 is show the Gantt chart for this project.

3.2 Development Process of Prosthetic Hand

Figure 3.1 shows the flow chart of development process prosthetic hand. This flowchart explained about overall process in prosthetic hand model.

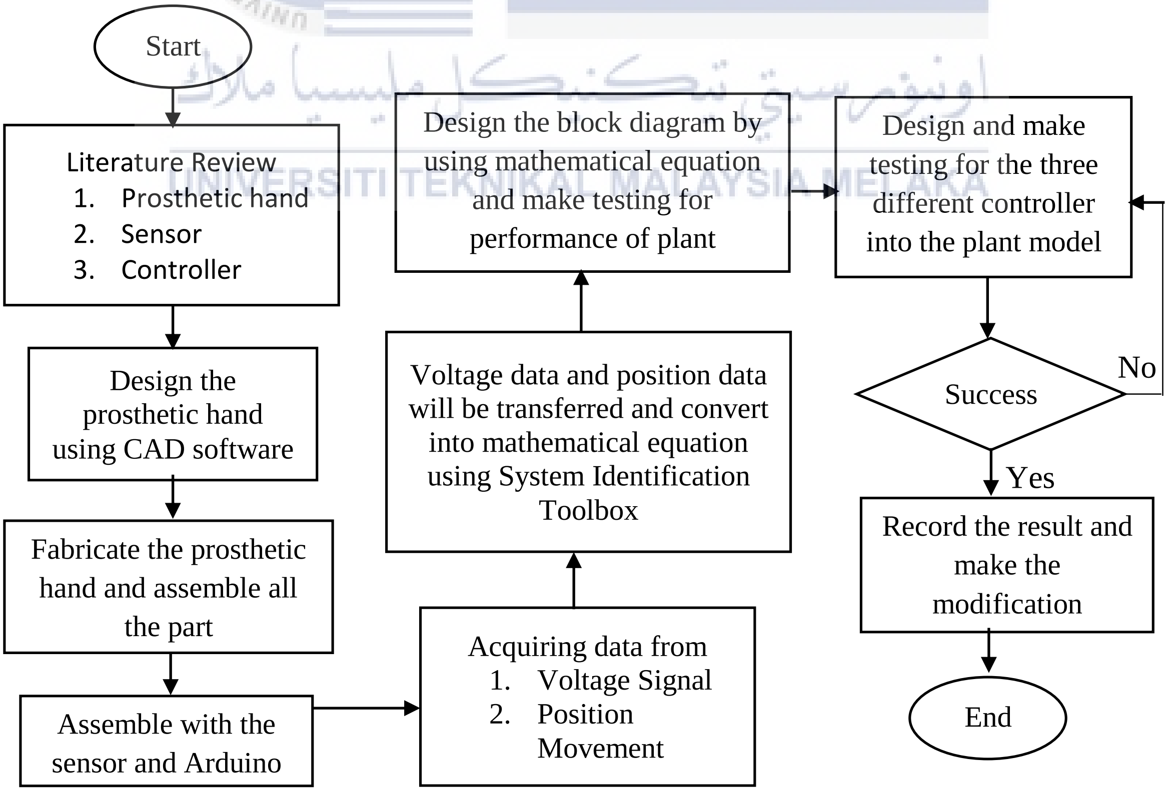


Figure 3.1: Flowchart of Development Process Prosthetic Hand

3.3 Development of Prosthetics Hand

The prosthetic hand will be designed and fabricated according to the condition and research that has been made. The design of prosthetic hand will be suitable with the noninvasive Flex system [5]. The existing prosthetic hand is the complexity associated with the human hand, by using single actuators and flex sensor, which makes use of movement of finger. The physical size of the prosthetic hand requires a large degree of freedom to fit all needed components to replica a human hand.

3.4 Phase 1: Design of the Mechanical Prosthetics Hand in CAD

Detailed on mechanical design will be accomplished through the use of rough hand drawn sketches and CAD software. All backup data including design and part files were stored throughout the design of prosthetic hand process and can be made available upon request and can be made available upon request. Individual parts in prosthetic hand were carefully modelled with as much detailed information and accurate dimensions as possible. All aspects of the design such limited space, size components and accuracy will be the priority. Figure 3.1 shows flowchart of designing prosthetic hand model.

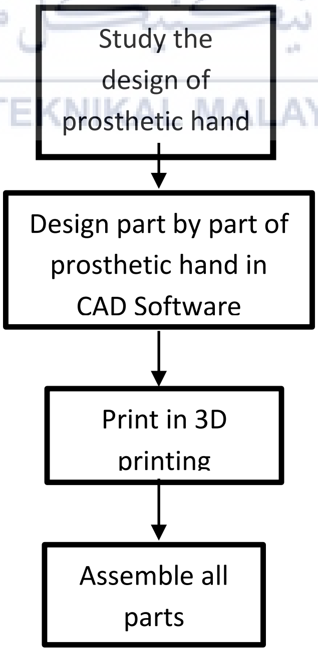


Figure 3.2: Flowchart of Designing Prosthetic Hand Model.

3.4.1 Mechanical Design

The mechanical design for this prosthetic hand is done by Gael Langevin who is a French designer and maker. On January 2012, personal project for Gael Langevin called InMoov was initiated as the first Open Source for prosthetic hand via Internet. Thus, InMoov design is the first Open Source 3D printed for human limb.

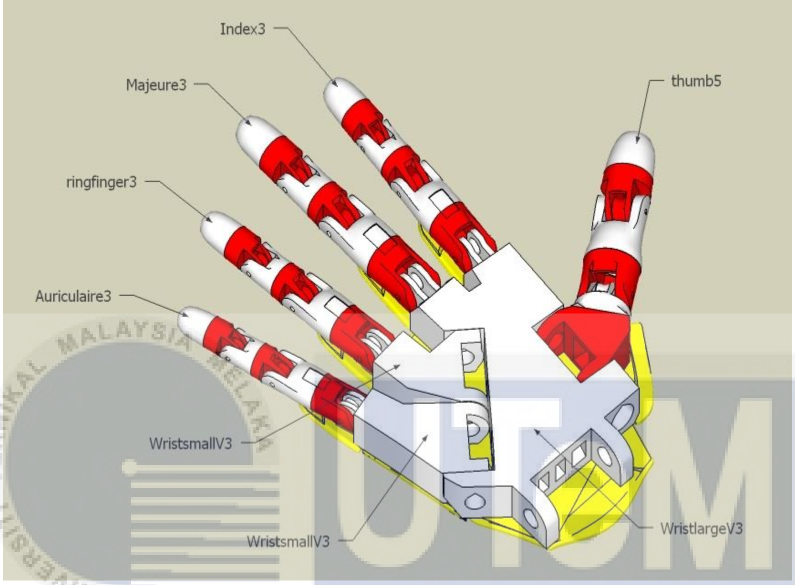


Figure 3.3: Lower View of InMoov Hand

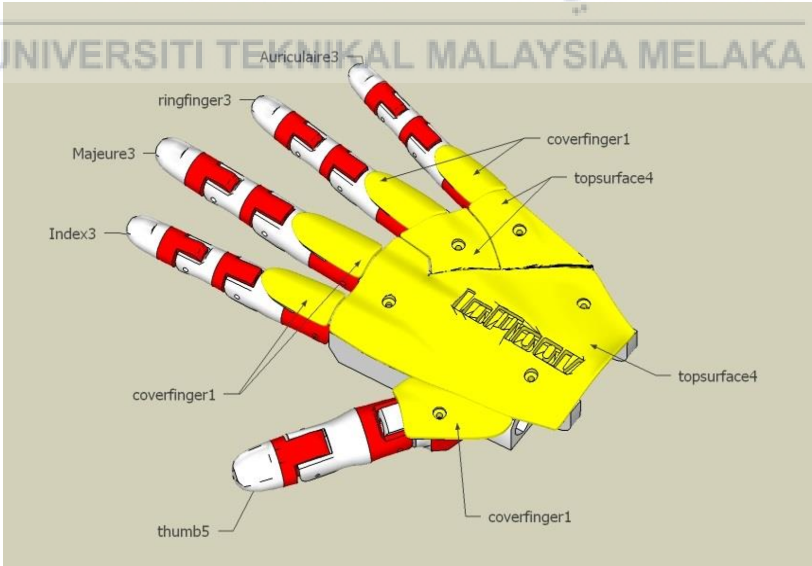


Figure 3.4: Upper View of InMoov Hand.

Figure 3.3 and 3.4 shows the lower and lower view of InMoov hand. All the part in design InMoov is Auriculaire, Ring finger, Majeure, Index, Cover finger, Thumb, and Top surface is printed by 3D printer and fabricate with the procedure given.

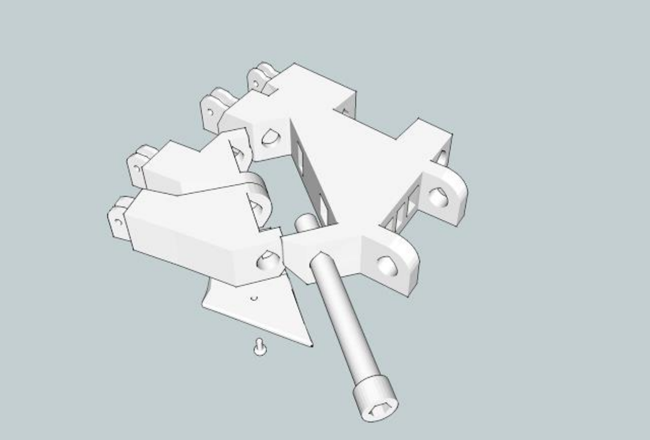


Figure 3.5: Palm Assembly

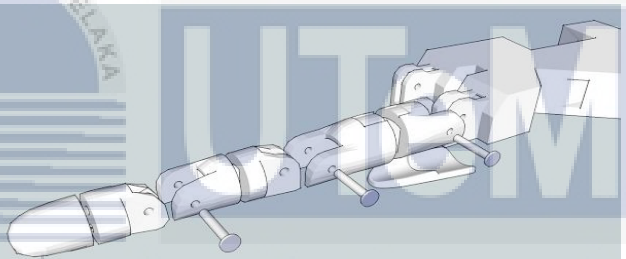


Figure 3.6: Finger Assembly.

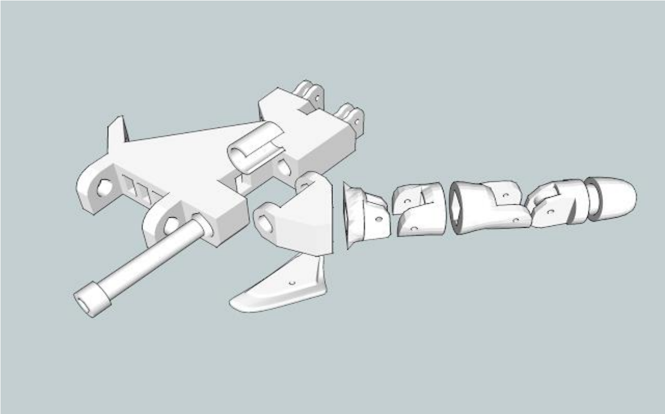


Figure 3.7: Thumb Assembly

Figure 3.5, 3.6 and 3.7 shows the procedure to assembly part of InMoov hand. For the rest of the finger can be assembly according the example in Figure 3.6. First, the palm of InMoov hand must be assembly before assembly the finger part. The top cover finger and top cover palm is glued after the palm and finger complete fabricate. For top part at each finger should not attach to make installation of fishing line easier.

Figure 3.8: Forearm Part

Figure 3.8 shows the forearm part from the wrist and before to elbow. This part is important because the servo motor bed for control finger is placed in this part. The parts are RotaWrist3, LeftRotaWrist1V2, LeftRobPart2V2, LeftRobPart5V2, LeftRobPart4V3, LeftRobPart3V3 and LeftWrist2. There are part not mention in this figure which is BedservoV3, TopBed1 and MiddleBed1. This part is implementing inside forearm InMoov to placed servo motor and fishing line route from servo motor to each finger.

3.5 Phase 2: Identification and Controller Design

An identification and control algorithm be designed using Simulink in MATLAB to perform a closed-loop intelligent control on a prosthetics hand. This project been focus primarily on the identification for multifunctional prosthetic hand. The prosthetics hand only has one degree of freedom (DOF) that only can move to another position [6]. In addition, Flex sensor as input to manipulate the position of finger will control the position at prosthetics hand. The data from Flex system will be collected at forearm which it near with muscle that controls the fingers. System identification toolbox will be utilized in order to obtain the connection between muscle signals and generated force from the prosthetics hand. The system identification will analyze the data that get from the output and make the transfer function.

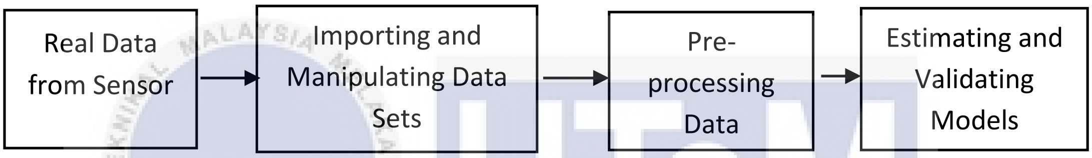


Figure 3.9: Block Diagram of System Identification

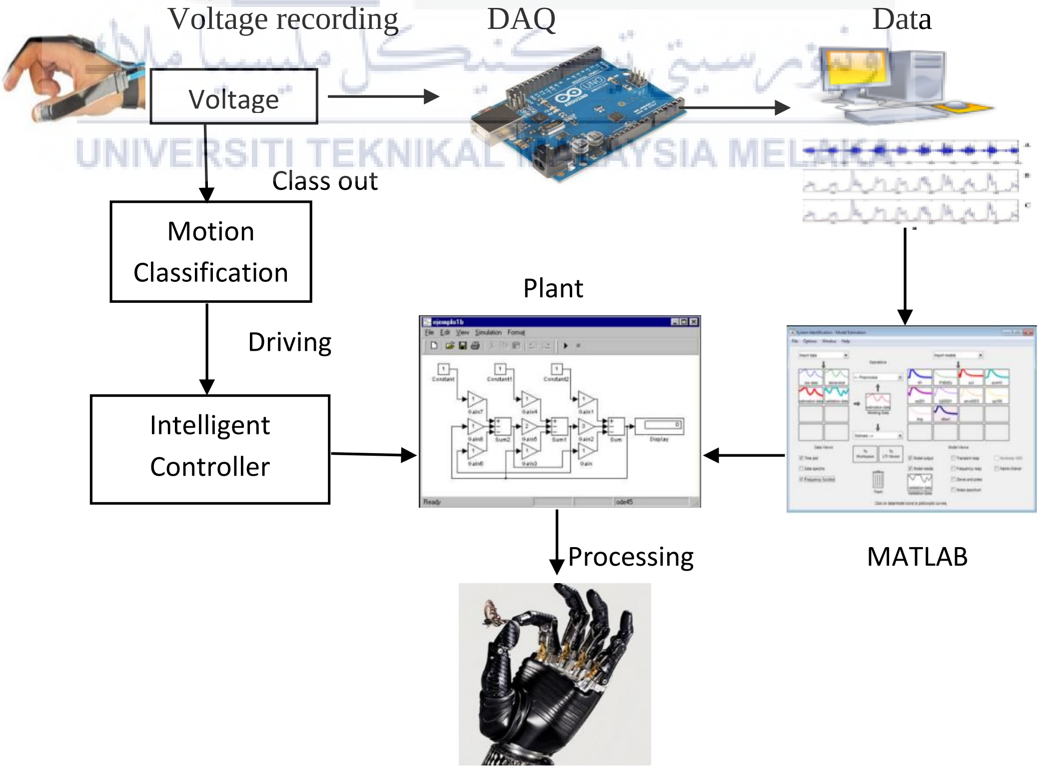
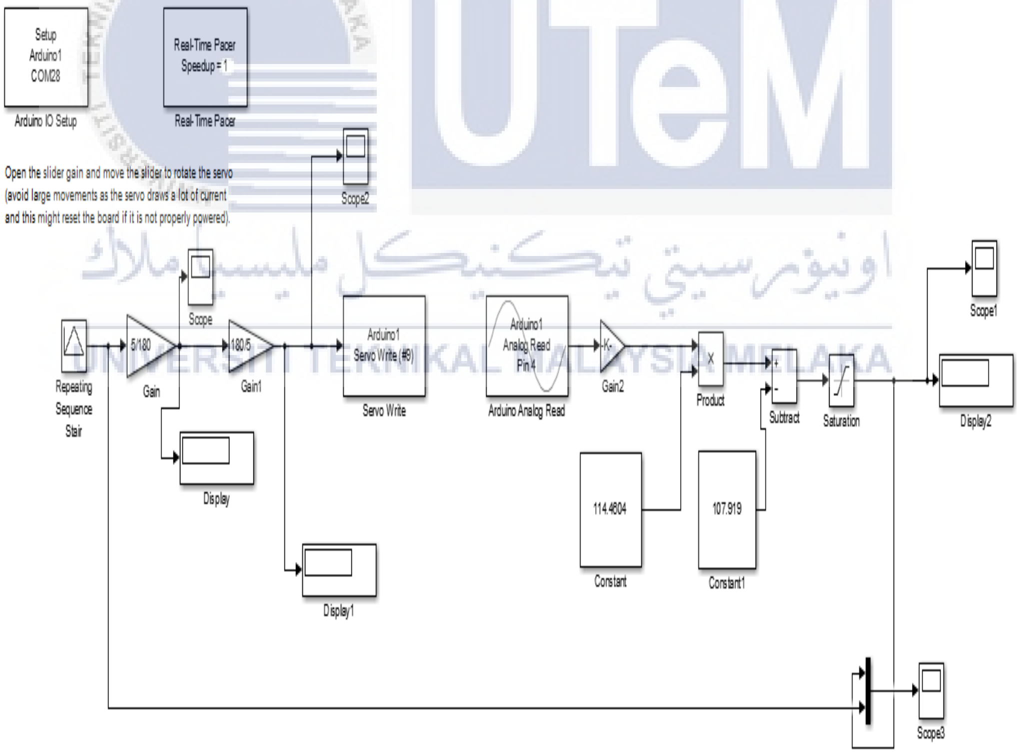


Figure 3.10: Identification and Controller Design Process

3.5.1 Circuit Design

The control circuit on the prosthetic hand will be discussed in this section. A block diagram has been designed by using Simulink in MATLAB to perform the circuit to control a position on a prosthetics hand. By using Arduino Package in MATLAB, the triangle wave as input that be used to generate the signal to move the servo motor. A gain that be used to convert the signal into voltage be using formula gain = 5V/180^0. A servo motor be using a Standard Servo Write in Arduino package. For output data, an Analog Output in Arduino package be used to determine the bit that produced when the finger of prosthetic hand moved. Besides that, a gain that be used to convert bit to a voltage be using formula gain = 5V/1023bit. A scope will be used to display the waveform for input and output of the plant. Figure 3.11 shows the control servo circuit by using Simulink.



Open the slider gain and move the slider to rotate the servo (avoid large movements as the servo draws a lot of current and this might reset the board if it is not properly powered).

Figure 3.11: Control Servo Circuit

3.5.2 Estimation and Validation of System Identification

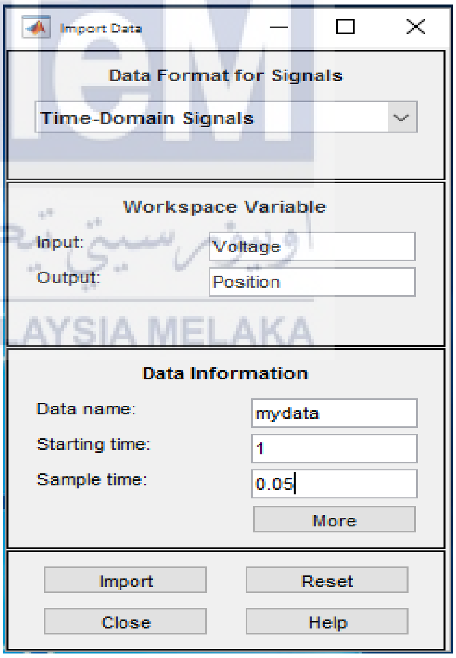
A system identification has been used in Simulink in MATLAB to perform a closed-loop intelligent control on a prosthetics hand. This part will focus primarily on the identification for multifunctional prosthetic hand. The prosthetics hand only has one degree of freedom (DOF) that only can move to another position [6]. In addition, triangle wave as input to manipulate the position will control the position at prosthetics hand. The data from triangle wave will be collected at Arduino that be connected into MATLAB. After that, output from flex sensor that be collected to get the position of finger in prosthetic hand. System identification toolbox will utilize in order to obtain the connection between input signals and generated position from the prosthetics hand. The system identification will analyze the data that get from the output and produce the transfer function.

Range: A1:B901
Variable Names Row: 1
unimportable cells with NaN

usah dikenang.xlsx

	A	B
	VarName1	VarName2
	Number	Number
1	5	0
2	4.9700	0
3	4.9400	0
4	4.9200	0
5	4.8900	0
6	4.8600	0
7	4.8300	1
8	4.8100	1
9	4.7800	1
10	4.7500	1
11	4.7200	2
12	4.6900	2
13	4.6700	2
14	4.6400	2
15	4.6100	3
16	4.5800	4
17	4.5600	4
18	4.5300	4
19	4.5000	4
20	4.4700	4
21	4.4400	5
22	4.4200	5
23	4.3900	6

Figure 3.12: Imported Data into Workspace



Time-Domain Signals

Input: Voltage
Output: Position

Data name: mydata
Starting time: 1
Sample time: 0.05

Figure 3.13: Data Input and Output

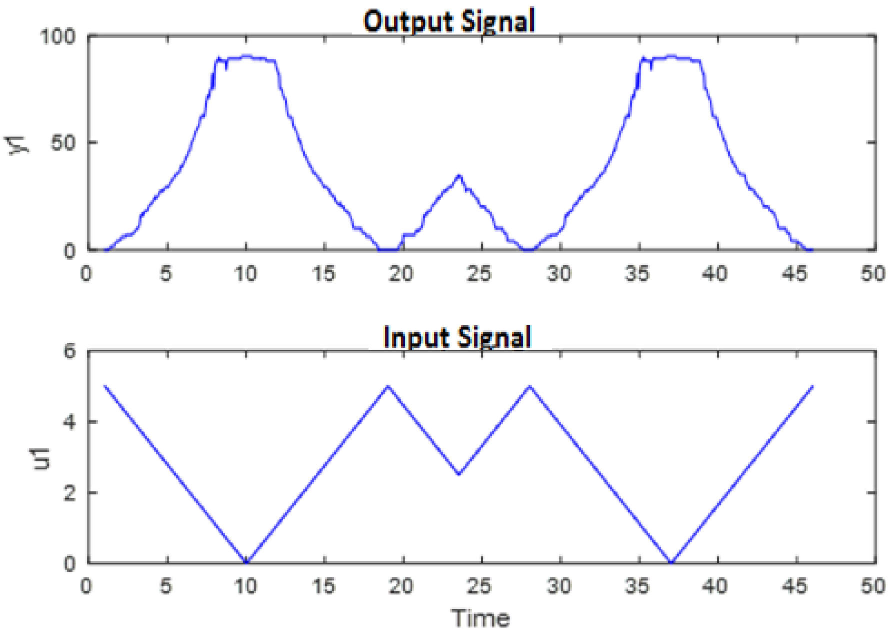


Figure 3.14: Input and Output Signal

To get the transfer function, range of estimation and validation in System Identification Toolbox that be used to separate estimation of the range input and validation of the range input.

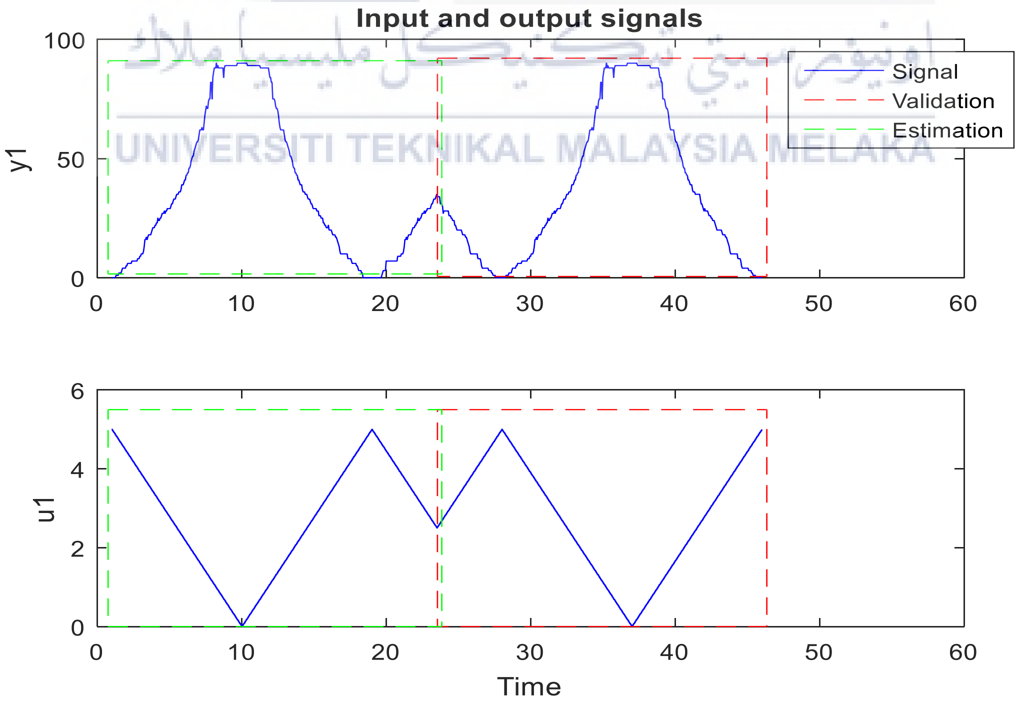


Figure 3.15: Estimation and Validation of Input and Output Signal

From the estimation and validation of input and output signal, the value of poles and zero be determined from System Identification Toolbox. The value of poles is 2 and value of zeros is 1. Result from the estimation shows the best fit from this transfer function is 81.17%.

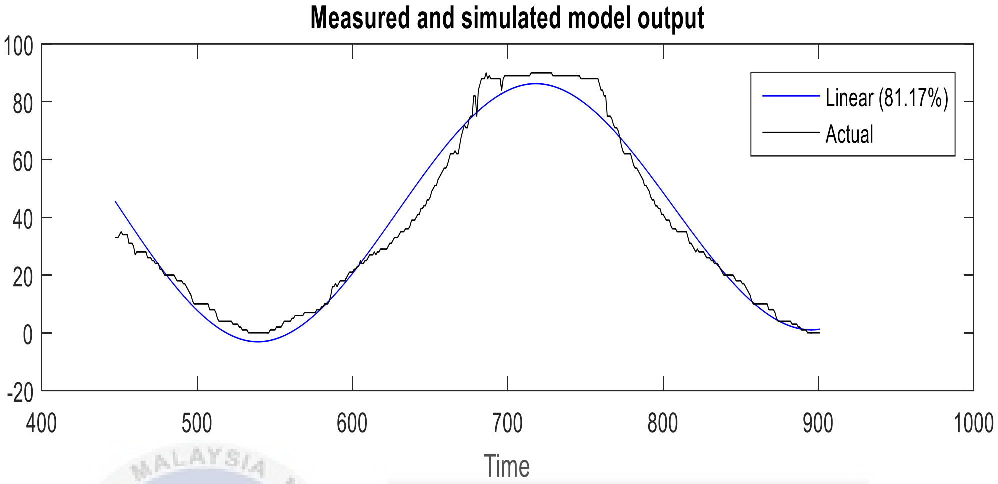


Figure 3.16: Measured and Simulated Model Output

Transfer function that be generated from System Identification Toolbox is:

$$tf = \frac{0.6858s + 1.712}{s^2 + 1.225e - 10s + 0.06855}$$

3.5.3 PID Controller

This part are focus primarily how to tune a PID controller into a transfer function of prosthetic hand that be generated. A PID controller block in MATLAB that be used to control the performance of output. A gain of PID controller can be tuned either manually or using tuning formula. And also, it can using auto tuning in Simulink. Figure 3.17 shows the flowchart of step to tune PID Controller.

By using method Auto-tuning, the value of Kp, K_I and K_D that be generated by using formula from MATLAB. A PID block to obtain an initial PID design:

$$C = Kp + \frac{Ki * Ts * z}{z - 1} + Kd * \frac{z - 1}{Ts * z}$$

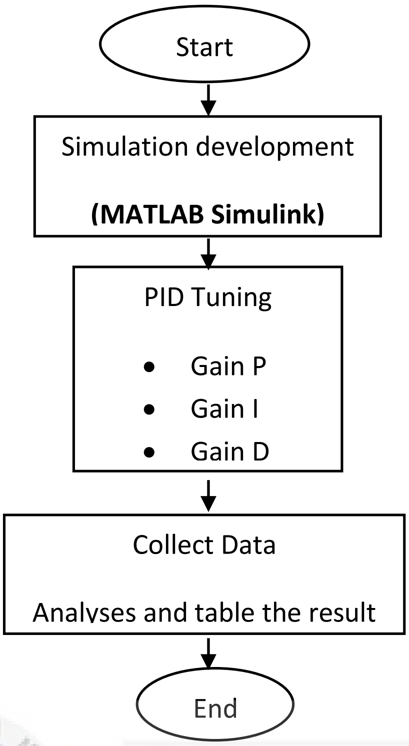
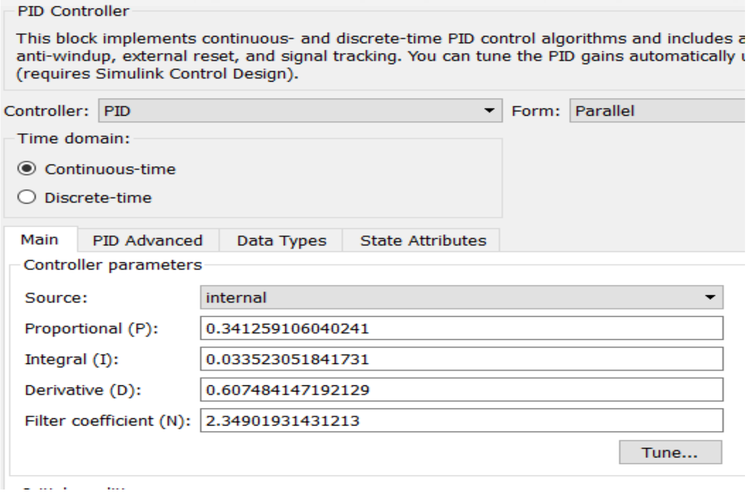


Figure 3.17: Flowchart of Tuning PID Controller

Figure 3.18 show the PID tuner in SIMULINK. When the PID Tuner in Simulink launches, the software in MATLAB analyzes a linearized plant model seen by the controller. This software automatically recognizes the plant input and output of system, and uses the current operating point for the linearization. Thus, the plant can be adjusted either for order or time delay.



PID Controller

This block implements continuous- and discrete-time PID control algorithms and includes
anti-windup, external reset, and signal tracking. You can tune the PID gains automatically
(requires Simulink Control Design).

Controller: PID
Form: Parallel

Time domain:
● Continuous-time
○ Discrete-time

Controller parameters

Source: internal
Proportional (P): 0.341259106040241
Integral (I): 0.033523051841731
Derivative (D): 0.607484147192129
Filter coefficient (N): 2.34901931431213

Figure 3.18: PID Tuner

The PID Tuner analyzes an initial PID controller based on the plant to achieve a logical tradeoff between robustness and performance. By default, step reference tracking performance for transient response and steady state error in the plot. Figure 3.19 shows the PID Tuner dialog with the adjusted design.

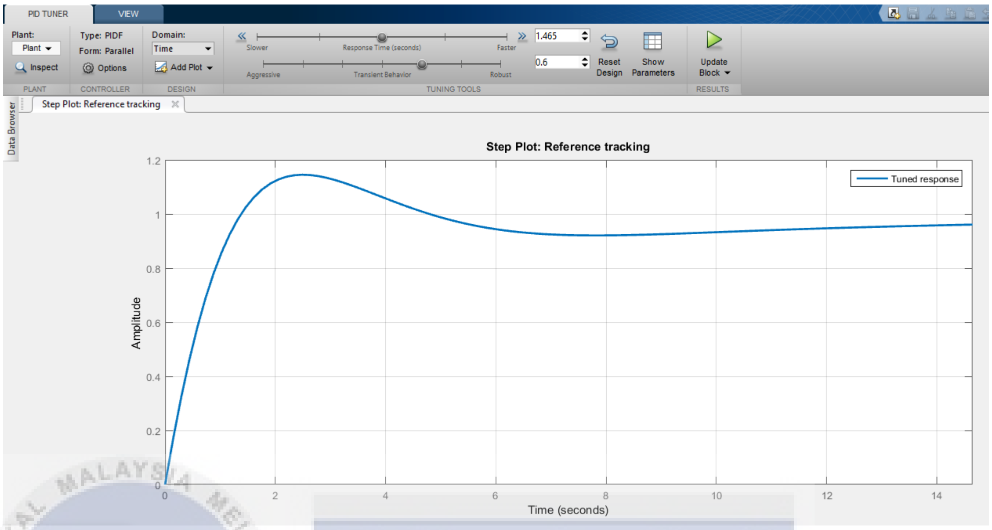
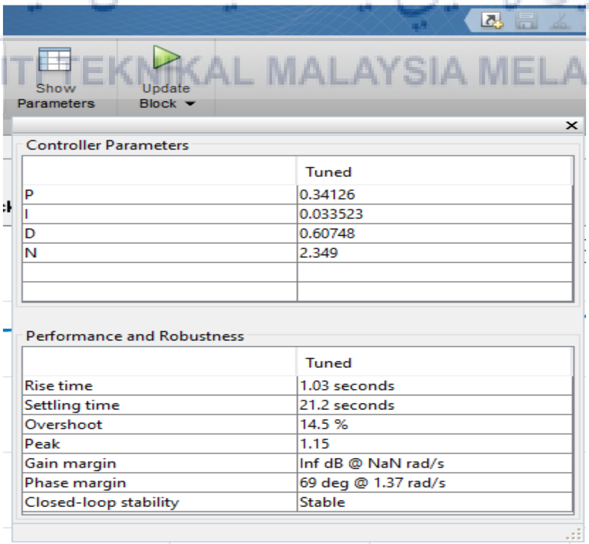


Figure 3.19: PID Tuner Dialog with the Adjusted Design

Besides that, Figure 3.20 shows PID tuner can be view controller parameters P, I and D, and a set of performance and robustness measurements. In addition, performance and robustness of system can changing by manually by adjusting the tuning tool.



Controller Parameters

	Tuned
P	0.34126
I	0.033523
D	0.60748
N	2.349

Performance and Robustness

	Tuned
Rise time	1.03 seconds
Settling time	21.2 seconds
Overshoot	14.5 %
Peak	1.15
Gain margin	Inf dB @ NaN rad/s
Phase margin	69 deg @ 1.37 rad/s
Closed-loop stability	Stable

Figure 3.20: Performance and Robustness Parameter in PID tuner

3.5.4 Fuzzy Controller

This part to be explain about step to tune a Fuzzy controller into a transfer function of prosthetic hand that be generated. By adjusting the FIS and selecting three membership, it is used to control the performance of output in prosthetic hand. Two input that be used into the Fuzzy controller where the error (e) and rate of error (de) is reused. An output from Fuzzy controller that be used. This FIS setting summarized below are based on design choices described in:

- Mamdani method fuzzy inference system that be used.
- Algebraic product for AND connective that be used.
- The ranges of input 1 and input 2 are normalized to [-90 90].
- The range of output is [-180 180].
- The defuzzification by using center of gravity method (COG).

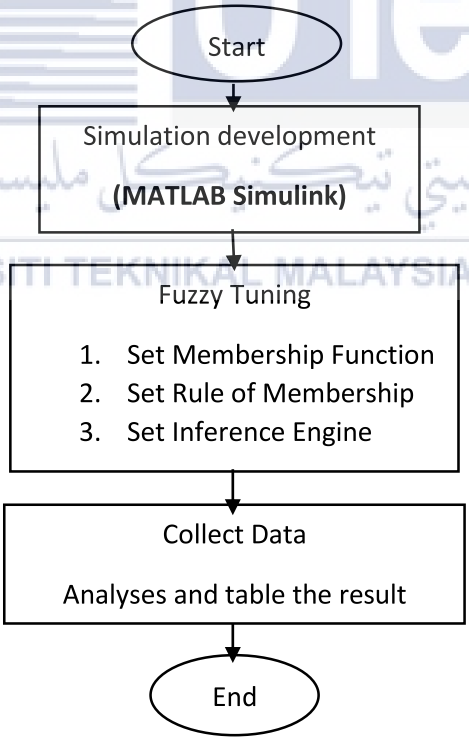
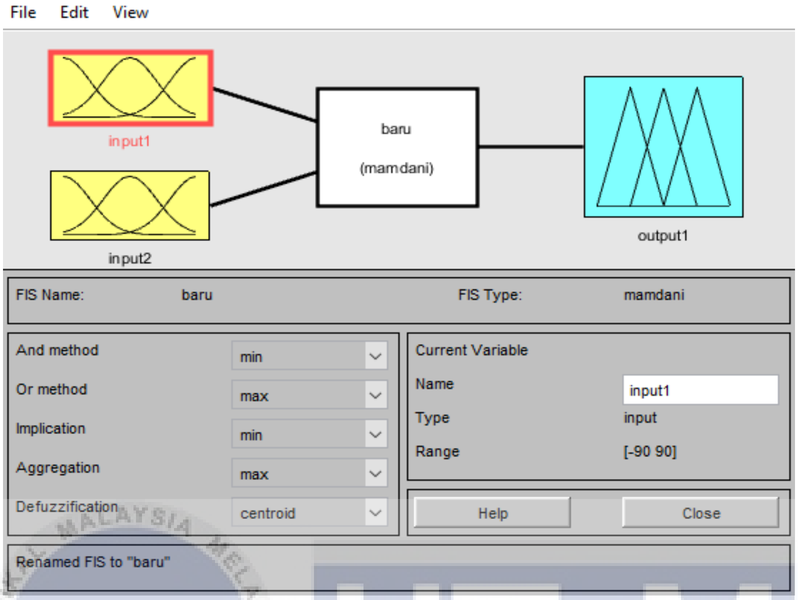


Figure 3.21: Flowchart of Tuning Fuzzy Logic Controller

Figure 3.21 shows the flowchart of tuning Fuzzy Logic Controller. While figure below shows the step by step how to create FIS by using Fuzzy Logic Toolbox in Simulink. Besides that, figure shows the flowchart for overall process for fuzzy controller.



FIS Name: baru

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

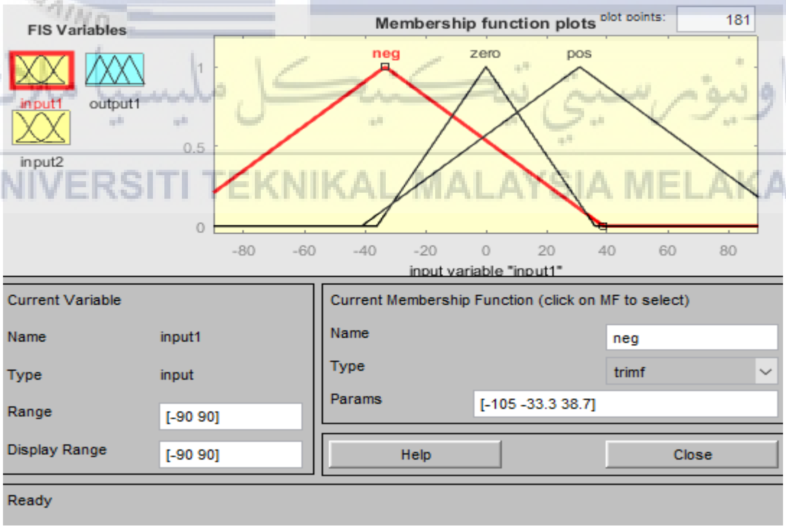
Name: input1

Type: input

Range: [-90 90]

Renamed FIS to "baru"

Figure 3.22: Value of Input and Output



plot points: 181

Current Variable

Name: input1

Type: input

Range: [-90 90]

Display Range: [-90 90]

Current Membership Function (click on MF to select)

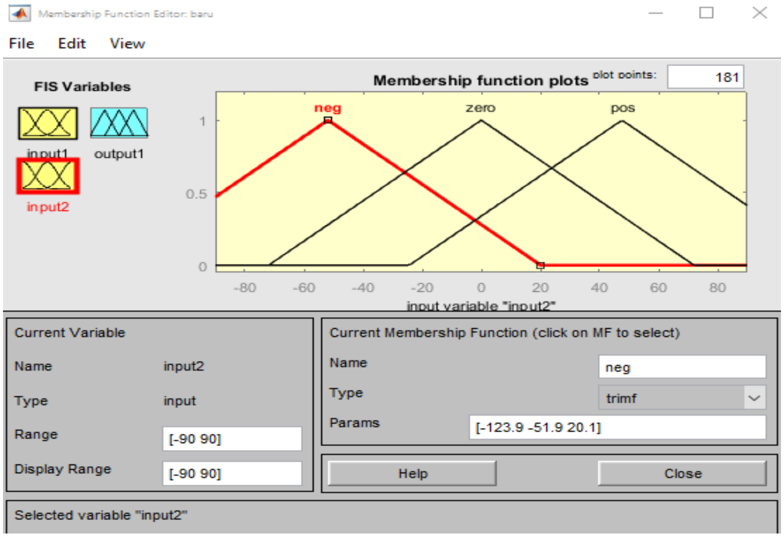
Name: neg

Type: trimf

Params: [-105 -33.3 38.7]

Ready

Figure 3.23: Membership Functions in Input 1 Constructed



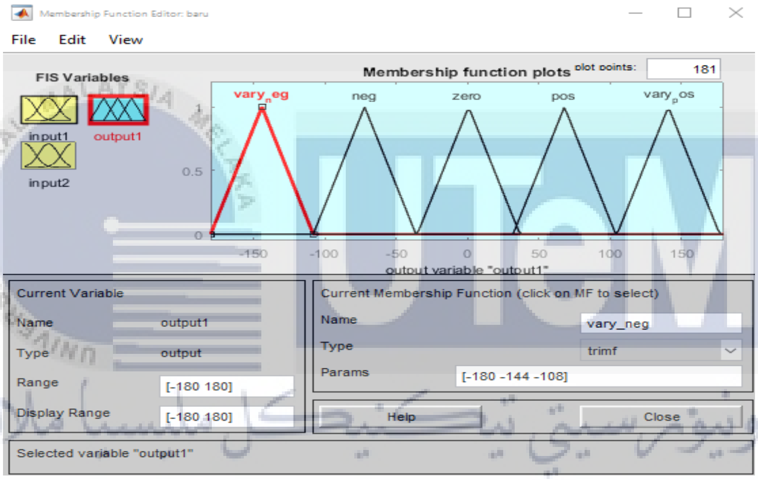
plot points: 181

Current Variable
Name: input2
Type: input
Range: [-90 90]
Display Range: [-90 90]

Current Membership Function (click on MF to select)
Name: neg
Type: trimf
Params: [-123.9 -51.9 20.1]

Selected variable "input2"

Figure 3.24: Membership Functions in Input 2 Constructed



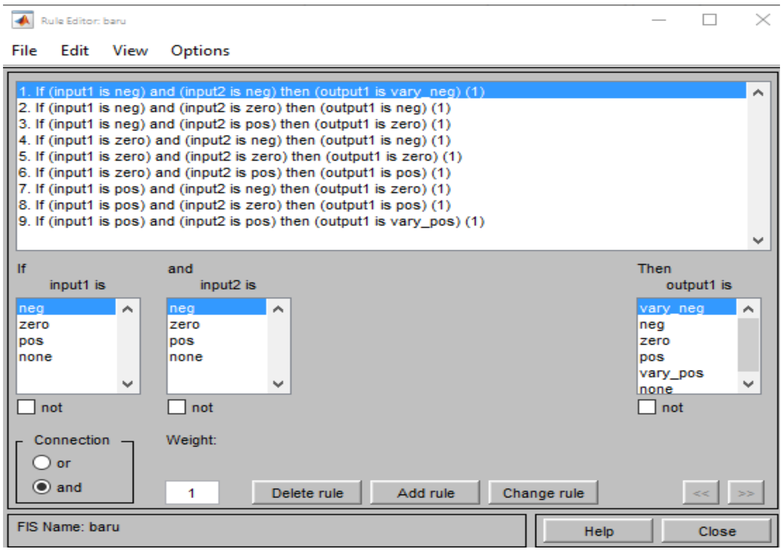
plot points: 181

Current Variable
Name: output1
Type: output
Range: [-180 180]
Display Range: [-180 180]

Current Membership Function (click on MF to select)
Name: vary_neg
Type: trimf
Params: [-180 -144 -108]

Selected variable "output1"

Figure 3.25: Membership Functions in Output Constructed



1. If (input1 is neg) and (input2 is neg) then (output1 is vary_neg) (1)
2. If (input1 is neg) and (input2 is zero) then (output1 is neg) (1)
3. If (input1 is neg) and (input2 is pos) then (output1 is zero) (1)
4. If (input1 is zero) and (input2 is neg) then (output1 is neg) (1)
5. If (input1 is zero) and (input2 is zero) then (output1 is zero) (1)
6. If (input1 is zero) and (input2 is pos) then (output1 is pos) (1)
7. If (input1 is pos) and (input2 is neg) then (output1 is zero) (1)
8. If (input1 is pos) and (input2 is zero) then (output1 is pos) (1)
9. If (input1 is pos) and (input2 is pos) then (output1 is vary_pos) (1)

FIS Name: baru

Figure 3.26: Rule Base in Rule Editor

3.5.5 Fuzzy-PID Controller

This method using two different controller became one controller where PID controller and Fuzzy Controller be used. Combination of both controller be used where this method using method of transfer the PID gains to the linear fuzzy controller. Figure 3.27 shows a block diagram about process to tuning Fuzzy PID controller.

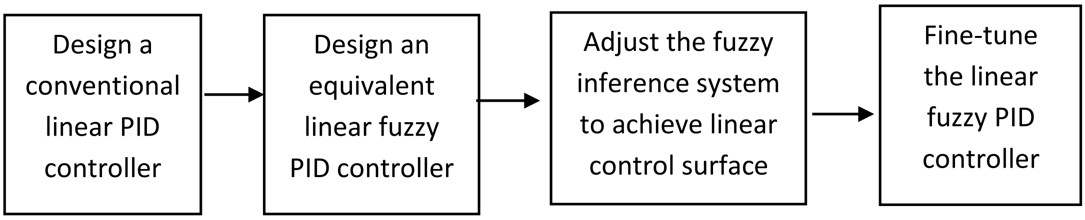


Figure 3.27: Process to Designing Fuzzy-PID Controller

The value gain of K_p, K_i and K_d be optimized by using manually to get better performance and robustness. By using PID in parallel method, value of K_p and K_d connected directly to the Fuzzy controller. While output in Fuzzy controller be connected with gain K_i using the sum block to complete block diagram PID controller. Figure 3.28 shows block diagram PID in parallel method to tuning a Fuzzy Controller.

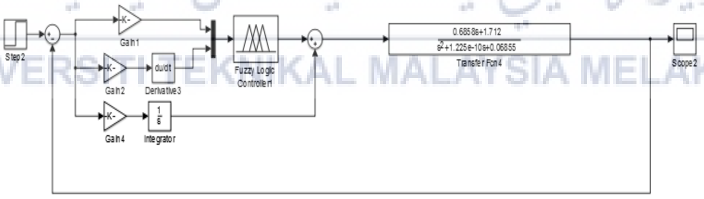


Figure 3.28: Fuzzy PID Controller in Simulink

Two input that be used into the Fuzzy controller where the error (e) be connected with gain K_p and while rate of error (de) connected with K_d is reused. An output from Fuzzy controller that be used. This FIS setting summarized below are based on design choices described in:

- Mamdani method fuzzy inference system that be used.
- Algebraic product for AND connective that be used.
- The ranges of input 1 and input 2 are normalized to [-90 90].
- The range of output is [-180 180]

3.6 Gantt Chart

Table 3.1: Process Development of Prosthetic Hand for PSM 1 during Semester 15/16

Task	PSM1													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Research about Prosthetic Hand	■	■	■	■	■	■	■	■						
Research about forearm		■	■	■	■	■	■	■	■					
Research about Electromyograph			■	■	■	■	■	■	■	■				
Measurement on force signal								■	■	■	■	■		
Measurement on EMG signal									■	■	■	■	■	
PID & Fuzzy controller									■	■	■	■	■	■

Table 3.2: Process Development of Prosthetic Hand for PSM 2 during Semester 15/16

Task	PSM 2													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Design prosthetic hand by using Solidwork	■	■	■	■	■									
Fabrication prosthetic hand					■	■	■	■	■	■				
Assemble prosthetic hand										■				
Test the prosthetic hand										■	■	■		
Modification (If need)												■		
Test the prosthetic hand (2nd)												■	■	■

3.7 Project Milestone

Table 3.3: Project Timeline for the Development of Prosthetic Hand

Task	Duration	Start	End	Percentage Complete
Internet and library search	5 weeks	7 September 2015	12 October 2015	5%
Design Platform	9 weeks	5 October 2015	4 December 2015	20%
Software development	5 weeks	12 October 2015	16 November 2015	10%
Controller test	3 weeks	9 November 2015	30 November 2015	15%
PSM 1 preparation	1 weeks	1 December 2015	8 December 2015	10%
Hardware	2 weeks	11 January 2016	25 January 2016	10%
Assemble and fabrication	5 weeks	2 February 2016	5 March 2016	10%
Combination hardware with Flex sensor	4 weeks	7 March 2016	31 March 2016	20%
Functional test	4 weeks	3 April 2016	4 May 2016	5%
Final report	3 weeks	10 May 2016	30 May 2016	15%

CHAPTER 4

RESULT AND ANALYSIS

4.1 Introduction

This chapter described the result of this project. Summary of the result obtained through related works are show in Section 4.2 to Section 4.8. In Section 4.2 shows the development of prosthetic hand has been completed. Whereas, Section 4.3 shows the transfer function of the plant system that be generated in MATLAB. For Section 4.4 to Section 4.7 shows the performance of prosthetic hand model by using three different controller. In Section 4.8 shows the comparison between three different controllers.

4.2 Prosthetic Hand

This part shows the assembly part by part in prosthetic hand was completed where the glove will be connected with flex sensor to move the finger of prosthetic hand.

Figure 4.1: InMoov 3D Printed Robot with Glove

4.3 Transfer function of system plant

The transfer function for this simulation that been generated through System Identification Toolbox based on collected data for evaluate the performance of plant. This equation is:

$$tf = \frac{0.6858s + 1.712}{s^2 + 1.225e - 10s + 0.06855}$$

4.4 Performance of System Plant

Simulation are conducted in order to get the performance of the open loop system and closed loop system that be simulated into prosthetics hand model. Figure 4.2 shows the block diagram open loop of prosthetic hand model. The output that be generated through the simulation been conducted on prosthetic model plant shows the output waveform does not reach the desired value that shown on Figure 4.3.

Figure 4.2: Open Loop for Prosthetic Hand Model Plant

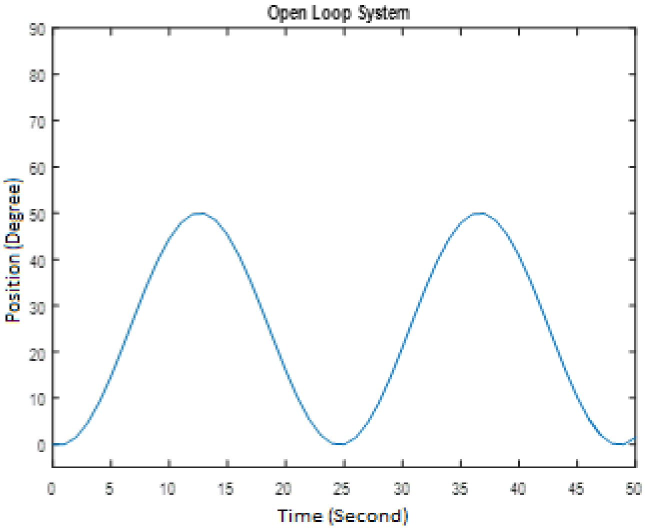


Figure 4.3: Output Waveform for Open Loop System

Whereas, Figure 4.4 shows the closed loop system of prosthetic hand where the feedback from the output of plant is fed back to the sum block to determine the error.

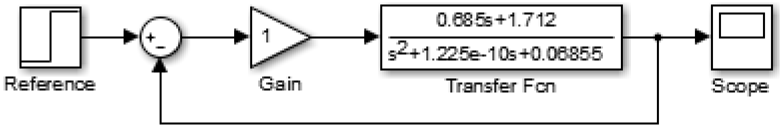


Figure 4.4: Closed Loop System for Prosthetic Hand Model

Figure 4.5 shows the step response of closed loop system for prosthetic hand. Based on Table 4.1, the step response showed performance of transient response in which the rise time takes 0.774 second. For the settling time, response shows the time taken from 10.2 seconds to stabilize. While, the overshoot in step response shows 50.3% of system plant. And also the steady state error indicates the value is not approaching the reference value which shows 0.38.

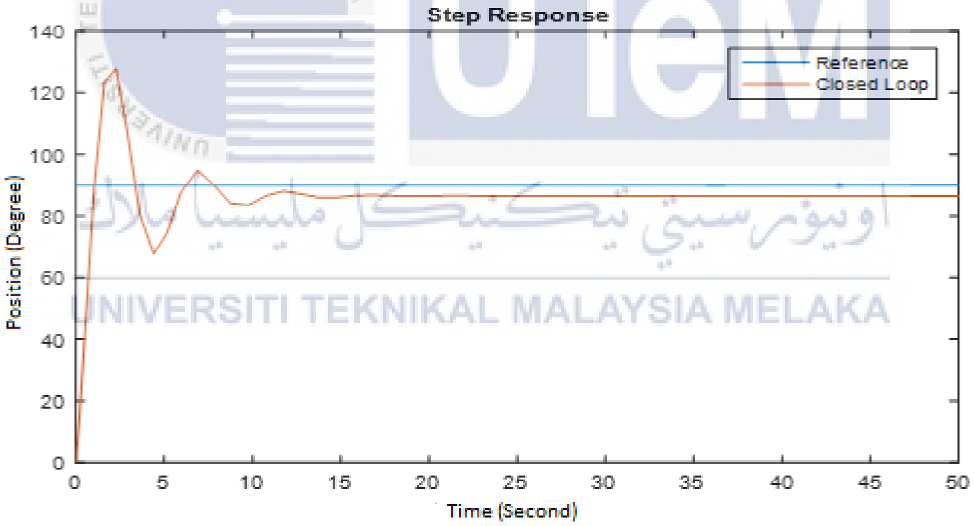


Figure 4.5: Step Response of Closed Loop for Prosthetic Hand

Table 1.1: Parameter of Transient Response in Closed Loop System

	Uncompensated
Rise time	0.774 second
Settling time	10.2 second
Overshoot	50.3%
Steady State Error	0.38

4.5 Results and Performance of PID Controller

Simulation works are conducted in order to get the performance of the PID controller for the simulated prosthetics hand model. The simulation in tuning the PID controller has been conducted with combinations of gain constant. The PID controller is being auto tune to review the consequence of all the gain constant in order to obtain optimal response. Figure 4.6 shows the design block diagram of PID Controller.

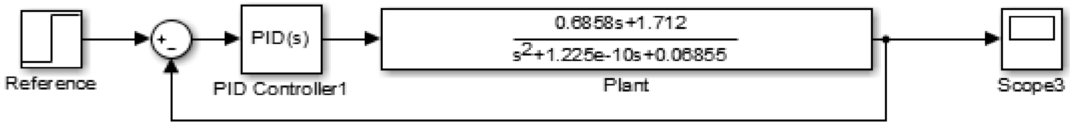


Figure 4.6: Block diagram of PID Controller

In PID tuner, a value of K_p, K_i and K_d will be set based on Table 4.2. This table shows the parameter of K_p, K_i and K_d by using auto-tuning in Simulink. From the tuning process in PID tuner, gain K_p is set to be 2.6413, gain K_i is set to be 0.6836 and for gain K_d is set to be 0.7260.

Table 4.2: Parameter of K_p, K_i & K_d for PID

Parameter	**Value**
Proportional Gain, K_p	2.6413
Integral Gain, K_i	0.6836
Derivative Gain, K_d	0.7260
Filter Coefficient	3.4311

Figure 4.7 shows tuned response showed in which the rise time increase from 0.774 second to 0.381 second. As well as with settling time, tuned response shows the time that be taken is 4.09 seconds to stabilize where the time taken is less than the response without controller is 10.2 seconds. While, the overshoot shows a sharp decline from 50.3% to 21.5% due to the gain K_p and K_d are working to improve the transient response. For steady state error, graph shows the response of the input reference and the output of a system in the same condition after tuning by using K_i. Therefore, steady state error of tuned response is zero.

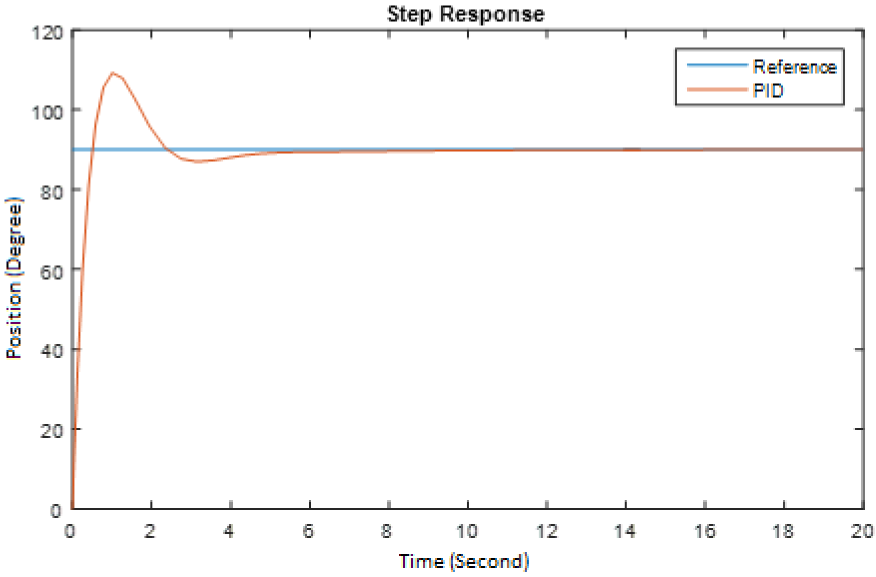


Figure 4.7: Step Response using PID Controller

Figure 4.8 shows the comparison between reference signal, compensated signal and uncompensated signal after tuning by using PID controller. This figure shows the compensated signal closely follows the reference response after the PID controller to reduce the errors from the plant system compared the uncompensated signal. Figure 4.8 shows the comparison of PID with another response. Table 4.3 shows the parameter of transient response using PID controller.

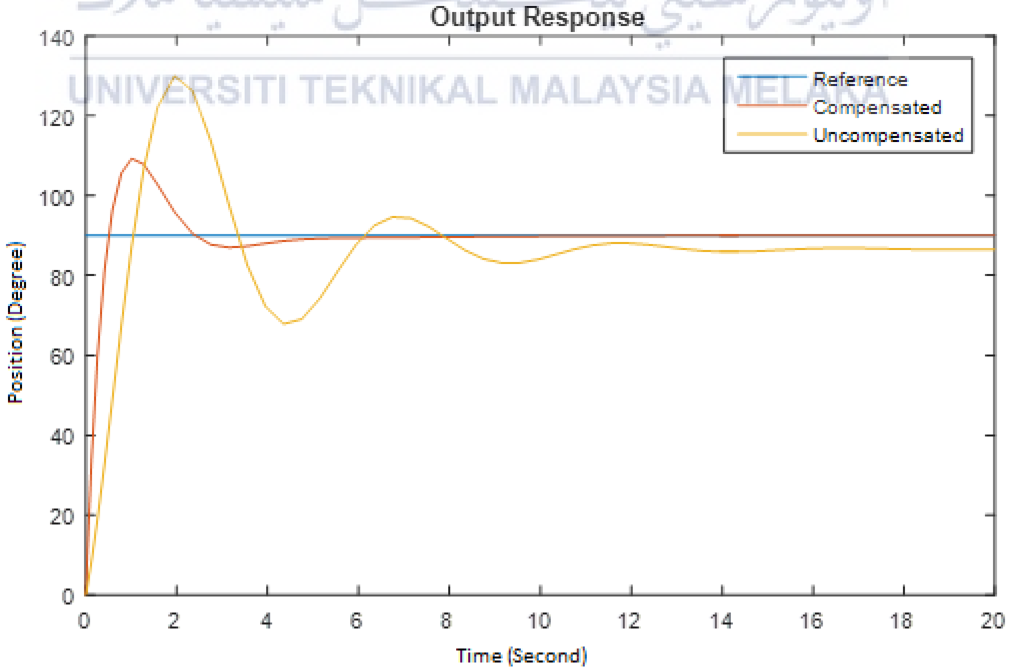


Figure 4.8: Comparison between Reference Signals, Compensated Signal and Uncompensated Signal after Tuning by using PID Controller.

Table 4.3: Parameter of Transient Response using PID

	Compensated	**Uncompensated**
Rise time	0.381 second	0.774 second
Settling time	4.09 second	10.2 second
Overshoot	21.5 %	50.3%
Steady State Error	0	0.38

4.6 Results and Performance of Fuzzy Logic Controller

Simulation has been conducted by using Fuzzy Logic controller that been designed in MATLAB. In order to obtain the output that optimized in prosthetic hand model, the input and output membership functions of the Fuzzy Logic controller is being tuned continuously based on the rule base. At the same time, the simulation of the Fuzzy controller into the prosthetic hand model is ongoing continuously to obtain optimum performance including transient response and steady state error. Figure 4.9 shows the design block diagram of Fuzzy Logic controller.

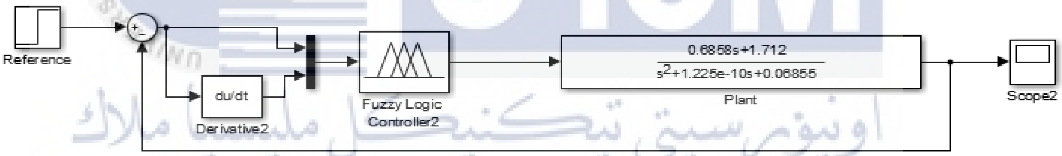


Figure 4.9: Block Diagram of Fuzzy Logic Controller.

Table 4.4: Rule Base of Fuzzy Logic

Input 1/Input 2	**Neg**	**Zero**	**Pos**
Neg	**Neg**	**Pos**	**Zero**
Zero	**Pos**	**Zero**	**Neg**
Pos	**Zero**	**Neg**	**Pos**

By setting the rule base based on Table 4.4 in Fuzzy Logic Toolbox. Based on Table 4.5, transient response has been improved in tuned response in which the overshoot shows the where significant changes from 50.3% to 3.3998%. Besides that, the settling time in tuned response shows the time taken to stabilize takes 6.1036 seconds than blocked response is 10.2 seconds. But, for rise time, became increase from 0.774 seconds to 1.441 seconds. For steady state error, graph shows the response of the input reference and the output of a system in the same condition after tuning by using K_i. Therefore, less change in steady state error where from 0.38 to 0.12. Figure 4.10 shows the step response of plant after tuning Fuzzy controller.

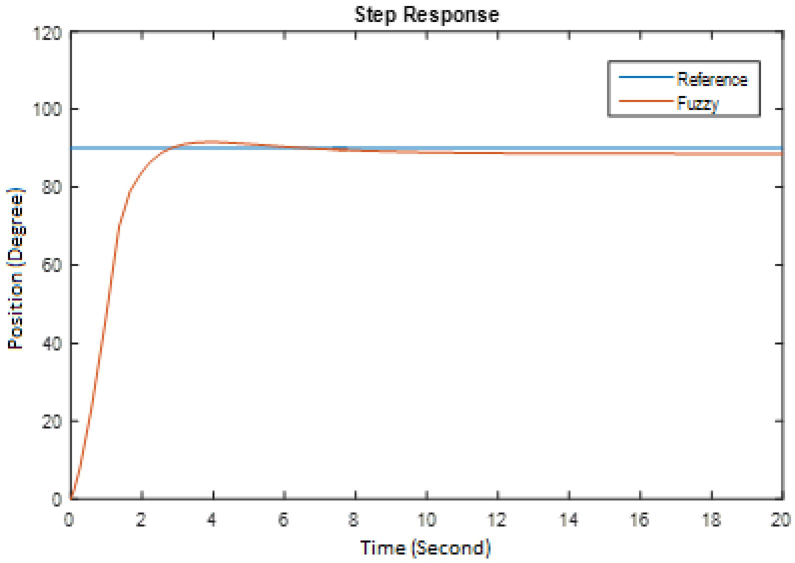


Figure 4.10: Step Response using Fuzzy Controller

Figure 4.11 shows the comparison between reference signal, compensated signal and uncompensated signal after tuning by using Fuzzy controller. This figure shows the compensated signal closely follows the reference response after the Fuzzy controller to mostly reduce the overshoot from the plant system compared the uncompensated signal of plant but less steady state error reduction.

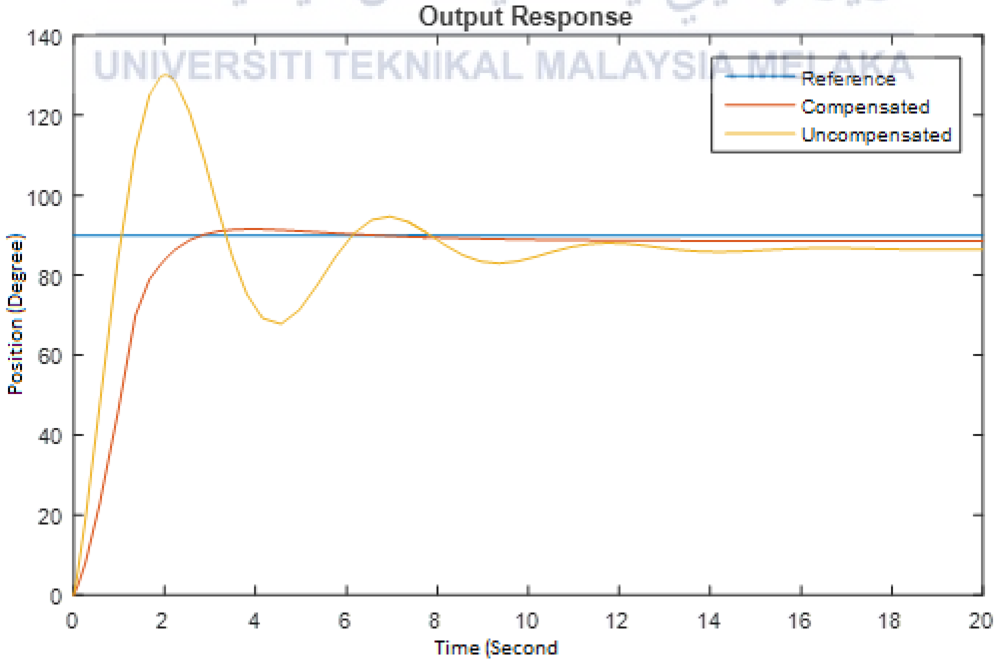


Figure 4.11: Comparison between Reference Signals, Compensated Signal and Uncompensated Signal after Tuning by using Fuzzy Controller

Table 4.5 Parameter of Transient Response using Fuzzy

	Compensated	**Uncompensated**
Rise time	1.441 second	0.774 second
Settling time	6.1036 second	10.2 second
Overshoot	3.3998%	50.3%
Steady state error	0.12	0.38

4.7 Results and Performance of Fuzzy-PID Controller

Simulation has been conducted by using Fuzzy-PID controller that been designed in MATLAB. In order to obtain optimized membership functions, the input and output membership functions of the Fuzzy Logic controller is being tuned continuously based on rule base. Whereas, to get optimum transient response and steady state error where the simulation is ongoing by using the Fuzzy-PID controller with combination of value for Kp, Ki and Kd and Fuzzy controller. Figure 4.12 shows the design block diagram of Fuzzy-PID controller.

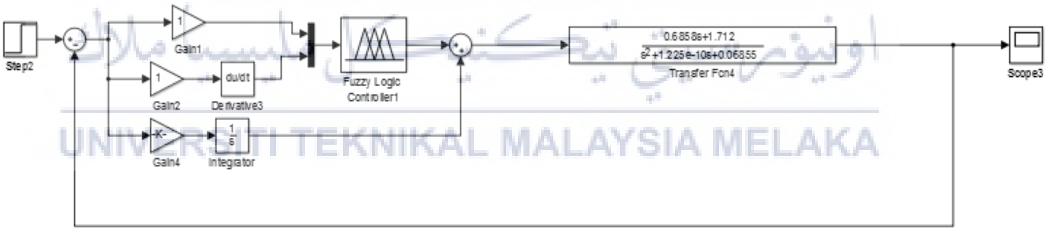


Figure 4.12: Block Diagram of Fuzzy-PID Controller.

By setting the rule base based on Table 4.7 in Fuzzy Logic Toolbox. Whereas, Table 4.6 shows the parameter of K_p, K_i and K_d by using manually in Simulink. From the tuning process in PID tuner, gain K_p is set to be 1, gain K_i is set to be 0.005 and for gain K_d is set to be 1.

Table 4.6: Parameter of K_p, K_i and K_d for Fuzzy-PID

Parameter	**Value**
Proportional Gain, K_p	1
Integral Gain, K_i	0.0005
Derivative Gain, K_d	1

Table 4.7: Rule Base of Fuzzy-PID Logic

Input 1/Input 2	**Neg**	**Zero**	**Pos**
Neg	**Neg**	**Pos**	**Zero**
Zero	**Pos**	**Zero**	**Neg**
Pos	**Zero**	**Neg**	**Pos**

Based on Table 4.8 tuned response showed improvement in transient response in which the rise time became increase from 0.774 second to 1.3021 second. For the settling time, tuned response shows the time taken to stabilize takes 4.0075 second than blocked response is 10.2 second. But, the overshoot became decrease shows the where significant changes from 50.3 % to 0%. Whereas, steady state error of tuned response is zero. Figure 4.13 shows step response of plant by using Fuzzy PID controller.

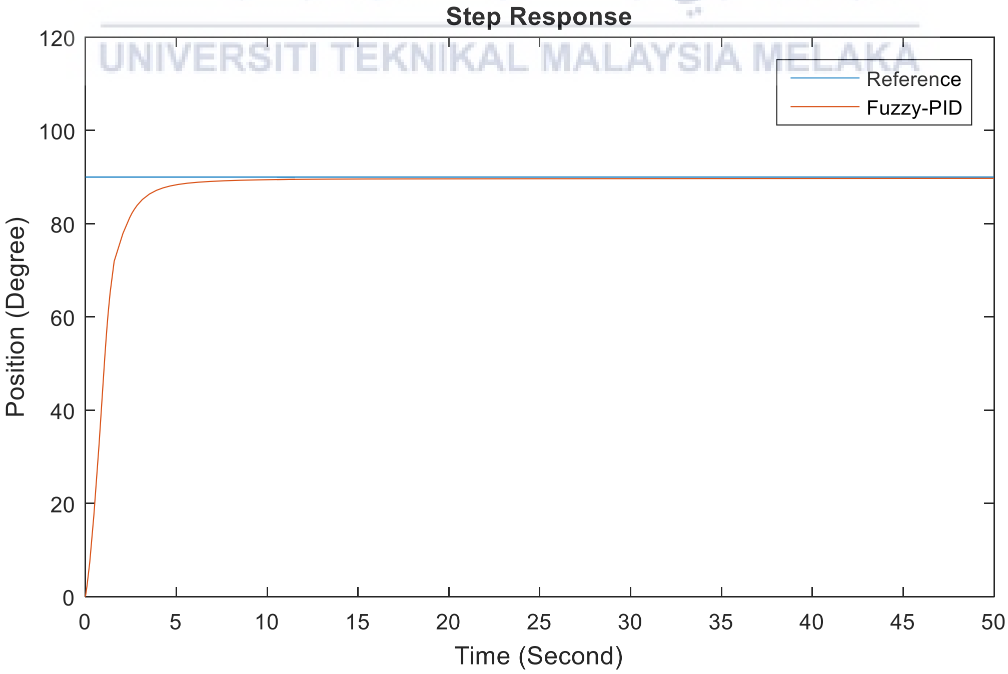


Figure 4.13: Step Response using Fuzzy-PID Controller

For comparison between reference signals, compensated signal and uncompensated signal after tuning by using Fuzzy-PID controller shows the compensated signal closely follows the reference response after the Fuzzy-PID controller to eliminate the overshoot from the plant system compared than the uncompensated signal. In addition, the transient response that be improved and steady state has been eliminated be shown on Figure 4.14.

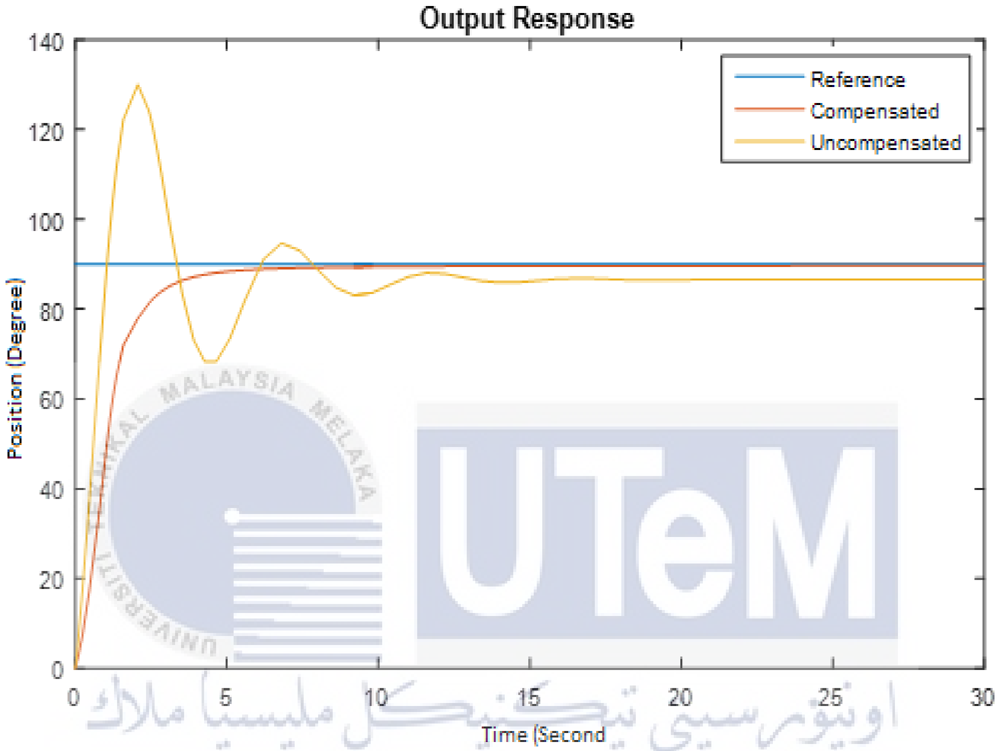


Figure 4.14: A Comparison between Reference Signals, Compensated Signal and Uncompensated Signal after Tuning by using Fuzzy-PID

. Table 4.8: Parameter of Transient Response using Fuzzy-PID

	Compensated	**Uncompensated**
Rise time	1.3021 second	0.774 second
Settling time	4.0075second	10.2 second
Overshoot	0 %	50.3%
Steady state error	0	0.38

4.8 Comparison between Performance of PID, Fuzzy and Fuzzy PID Controller

Simulation has been conducted by using PID, Fuzzy and Fuzzy-PID controller that been designed in MATLAB. Figure 4.15 shows the comparison of step response between three different controller that be used to control the performance of prosthetic hand model. Performance of Fuzzy PID controller shows the better transient response and steady state error than the PID and Fuzzy controller.

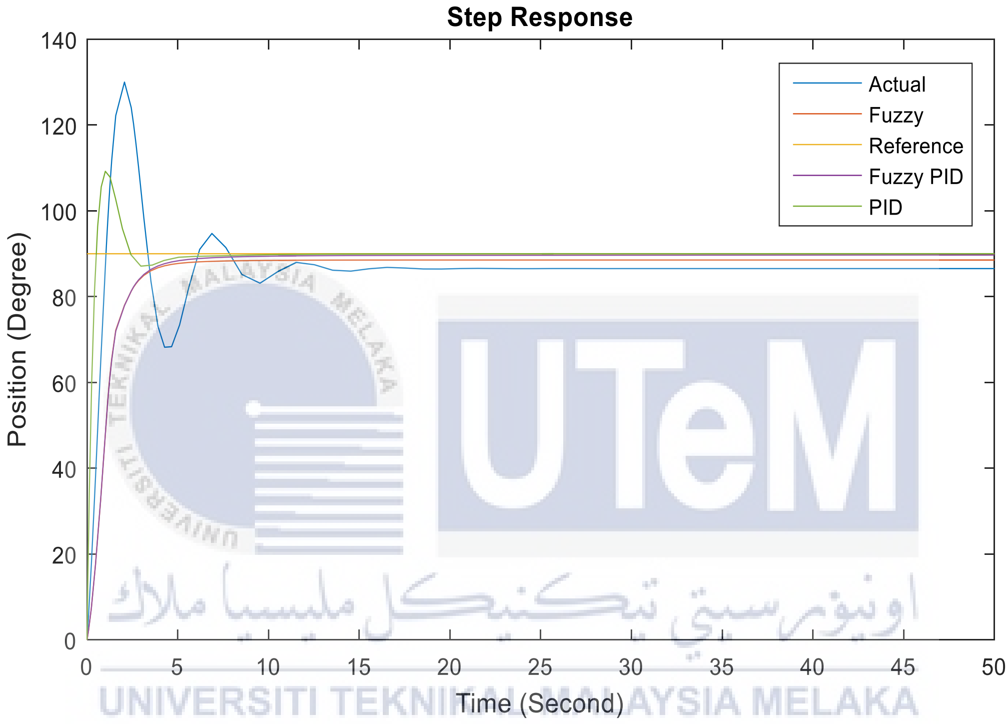


Figure 4.15: Comparison of Step Response between Three Different Controllers

Furthermore, three different controllers were tested with minimal root mean square error (RMSE) method. This method is used to indicate the minimum error when the plant of system tuned by the controller so that it reaches the desired value. Besides that, this method that ensured finding the minimum RMS value of the error between the reference and controller output. Based on Table 4.9 shows the root mean square (RMS) value of Fuzzy - PID controller lower than the other whereby showing 0.4073. Whereas, the PID controller shows the 0.7206 and Fuzzy controller shows the 1.466. Figure 4.16 shows the block diagram of three controller be tested by using RMS block.

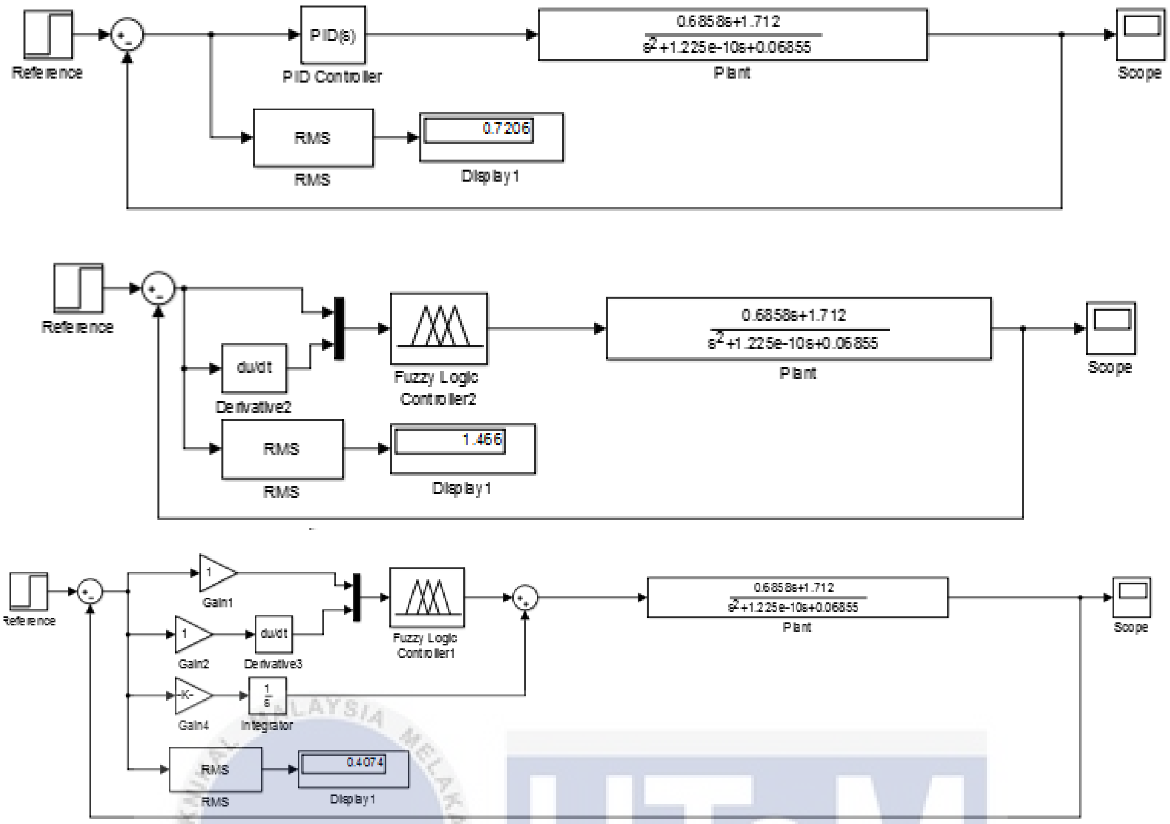


Figure 4.16: Block diagram of Three Controller Tested by using RMS Block

Table 4.9: RMS Value for Three Different Controller

Controller	**RMS value**
PID	0.7206
Fuzzy	1.466
Fuzzy-PID	0.4073

CHAPTER 5

CONCLUSION & RECOMMENDATION

5.1 Introduction

The overall conclusion for this report is summarized in Section 5.2 where conclude that Fuzzy PID can improves the performance of the prosthetic hand model. Section 5.3 describes the recommendation in this project that be helped in future.

5.2 Conclusion

In this report, the design and control of a prosthetics hand has been presented. The mechanical design of prosthetic hand is important element of which can control movement and grasping of hand for do life routine. Since the mechanical design of the prosthetic hands is separated into bicep and forearm design, it is relatively can imitate the human movements although the smoothness on prosthetic hand is doubted. Hence the introduction of the intelligent controller to the system might improve the movements of the prosthetics hand. Performance of controller also can be tested by using different controller. In this project, three different controller for the prosthetic hand has been developed. By select either controller mode with proper tuning, the prosthetic hand can reach the desired position but the performance of the system for three different controller is not same. The response graphs for different controller approach are obtained. This research shows performance of PID, Fuzzy Logic and Fuzzy-PID controllers for prosthetic hand be compared. PID controller has reduces less overshoot and able to deal with more changing conditions and Fuzzy controller has reduce mostly the overshoot and better transient response. It can be concluded that the Fuzzy-PID controller was better controller than PID and Fuzzy controller where it improve the transient response especially eliminate the overshoot and higher accuracy.

5.3 Recommendations

For the future work of the project prosthetic hand, some suggestions have been proposed based on the problems or deficiencies encountered during the work in progress include to improve the performance of controller.

- Intelligent controller can be designed in many ways to obtain better performance of the system. Type of intelligent controller such as Artificial Neural-Network, Genetic Algorithm Fuzzy Logic Controller, ANFIS and so on can be developed to get better control of the prosthetic hand.

REFERENCES

[1] T. Zhang, J. Zhao, L. Jiang, and H. Liu, “Design and control of a multisensory five-finger prosthetic hand,” *Proceeding 11th World Congr. Intell. Control Autom.*, pp. 3327–3332, 2014.

[2] K. Singh, H. S. Ryait, and R. Bhatia, “Precise and Accurate Multifunctional Prosthesis Control Based on Fuzzy Logic Techniques,” *2011 Int. Conf. Commun. Syst. Netw. Technol.*, pp. 188–193, 2011.

[3] A. B. Ajib, “A Multifunction Prosthesis Controller based on Fuzzy-Logic Techniques .,” *Annu. Int. Conf. IEEE EMBS*, pp. 1678–1681, 2003.

[4] D. Nishikawa and W. Yu, “EMG prosthetic hand controller using real-time learning method,” *Syst. Man, …*, pp. 153–158, 1999.

[5] C. Cipriani, F. Zaccone, G. Stellin, L. Beccai, G. Cappiello, M. C. Carrozza, and P. Dario, “Closed-loop controller for a bio-inspired multi-fingered underactuated prosthesis,” *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 2006, no. May, pp. 2111–2116, 2006.

[6] Z. X. Ming and L. S. Yu, “Simulation Study on Fuzzy PID Controller for DC Motor Based on DSP,” *2012 Int. Conf. Ind. Control Electron. Eng.*, pp. 1628–1631, 2012.

[7] X. Jingjing and L. Jiaoyu, “Neural network PID controller auto-tuning design and application,” Publication Year: 2013, pp. 1370–1375, 2013.

[8] R. Arulmozhiyal and R. Kandiban, “Design of Fuzzy PID controller for Brushless DC motor,” *Int. Conf. Comput. Commun. Informatics (ICCCI -2012)*, 2012.

[9] A. Khodayari, M. Talari, and M. M. Kheirikhah, “Fuzzy PID controller design for artificial finger based SMA actuators,” *IEEE Int. Conf. Fuzzy Syst.*, pp. 727–732, 2011.

[10] S. Bennett, “Development of the PID controller,” *Control Syst. IEEE*, vol. 13, no. 6, pp. 58–62, 64–5, 1993.

[11] G. M. Van der Zalm, “Tuning of PID-type Controllers: Literature Overview,” pp. 1–22, 2004.

[12] C.-H. Chen, D. S. Naidu, A. Perez-Gracia, and M. P. Schoen, “A hybrid adaptive control strategy for a smart prosthetic hand.,” *Annu. Int. Conf. IEEE Eng. Med. Biol.*

Soc. EMBC'09, vol. 2009, pp. 5056–5059, 2009.

[13] C.-H. Chen and D. S. Naidu, "Fusion of fuzzy logic and PD control for a five-fingered smart prosthetic hand," *2011 IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE 2011)*, pp. 2108–2115, 2011.

[14] L. Cammarata and L. Yliniemi, *Department of Process Engineering Development of a Self-Tuning Fuzzy Logic Controller for a Rotary Dryer*, no. December. 1999.

[15] J. Jantzen, "Tuning Of Fuzzy PID Controllers," *Tech. Univ. Denmark Dep. Autom. Bldg*, vol. 871, no. 98, pp. 1–22, 1998.

[16] M. S. Bin Mohd Aras, S. M. S. B. Syed Abdul Hamid, F. B. A. Azis, F. A. B. Ali, and S. Shah B Abdullah, "Study of the effect in the output membership function when tuning a Fuzzy logic controller," *Proc. - 2011 IEEE Int. Conf. Control Syst. Comput. Eng. ICCSCE 2011*, pp. 1–6, 2011.

[17] Beng Guey Lau, " An intelligent prosthetic hand using hybrid actuation and mtelectric control". PhD thesis, University of Leeds, 2009

[18] Erik Scheme, MSc, PEng; Kevin Englehart, " Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use", PhD, PEng Institute of Biomedical Engineering, University of New Brunswick, Fredericton, Canada DOI:10.1682/JRRD.2010.09.0177

APPENDIX

A. MATLAB Command

```
num=[0 0.6858 1.712]; den=[1 1.225e-10 0.06855]; G=tf(num,den)

Gs=feedback(G,1)

pzmap(G)

pole(G)

zero(G)

step(G)

rlocus(G)

sisotool(G)

Kp=2.6413

Ki=0.6836

Kd=0.7260

C=pid(Kp,Ki,Kd)

T=feedback(C*G,1)

t = 0:0.01:2

step(T,t)

hold on

Kp1=2.567

Ki1=0.6789

Kd1=0.973

C1=pid(Kp1,Ki1,Kd1)

T1=feedback(C1*G,1)

t = 0:0.01:2

step(T1,t)

hold on

Kp2=3.789

Ki2=0.5734

Kd2=0.795

C2=pid(Kp2,Ki2,Kd2)
```

```
T2=feedback(C2*G,1)

t = 0:0.01:2

step(T2,t)

hold on

step(G)

hold on

stepinfo(T)

stepinfo(T1)

stepinfo(T2)

stepinfo(G)
```