FAKULTI KEJURUTERAAN ELETRIK

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LAPORAN PROJEK

SARJANA MUDA (PSM2)

FUZZY LOGIC INVERTED PENDULUM

AHMAD BADRUN HISHAM BIN ABDUL RAHIM

Bachelor of Mechatronics Engineering

JUNE 2014

DECLARATION

I declare that this report entitle "*Fuzzy Logic Inverted Pendulum"* is the 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 :

Authors Name: AHMAD BADRUN HISHAM BIN ABDUL RAHIM

Date : JUNE 2014

SUPERVISOR'S ENDORSEMENT

I hereby declare that I have read through this report entitle "*Fuzzy Logic Inverted Pendulum*" and found that it has comply the partial fulfillment for awarding the degree of Bachelor of Mechatronics Engineering

Signature :

Supervisor's Name: CIK NUR MAISARAH BINTI MOHD SOBRAN

Date : JUNE 2014

DEDICATION

To my beloved mother and father

ACKNOWLEDGEMENT

First and foremost I would like to express my profound gratitude to my supervisor, Cik Nur Maisarah Binti Mohd Sobran, whom had guided me throughout this Final Year Project. Her precious guidance, knowledge and support allowed me to accomplish my work successfully. Her continuous advices, feedback and comments are very much appreciated. Her willingness to discuss and accept ideas has been crucial in further improving my project. Her also extensive knowledge in coding MATLAB and derivation for Fuzzy Logic were had guided me during the beginning of the learning process.

The last but not lease, I would like to thank my parent and friend for their understanding and not interfering with my time schedules. Thanks to all those who had been there to aid me during my times of need.

ABSTRACT

Inverted Pendulum (IP) System is a device that uses a cylindrical rod that is usually made of aluminum and free swinging around on a fixed axis. This pivot mounted on a cart that can move in the horizontal direction. The rod is normally will tend to fall down from the top vertical position due to the instability in the control system. The goal of this project is to control and develop the hardware to stabilize the vertical position of the inverted pendulum system approach for purpose in the education tranning kit. The system consists of rotary variable resistance as input sensor that determines the angle variable for each movement of the pendulum. In order to maintain the pendulum in upright position of 90^0 degree, the motor horizontal position control via PWM will eventually become the response mechanism for the system. As an initial insight of the desired performance of Inverted Pendulum system, a simulation via MATLAB was done using the mathematical modely of the system. The inabilities for the Inverted Pendulum to maintain the upright position lead to the usage of the Fuzzy Logic controller onto the system. After that, the hardware development of the Inverted Pendulum system was done using selected materials. The performance of both, expected result from simulation and actual result were analysed. The performance of Inverted Pendulum has shown the reliability and accuracy of the pendulum in the upright position even though there was some error around 0.02% during the test performed.

ABSTRAK

"*Inverted Pendulum* (IP) system" ialah sebuah alat yang mengunakan satu batang rod berbentuk silinder yang biasanya diberbuat daripada aluminium dan bebas berayun di sektar satu paksi yang tetap. Rod ini dipasang di atas sebuah kereta yang boleh bergerak dengan arah melintang. Rod ini biasanya akan cenderung untuk jatuh dari keadaan menegak disebabkan oleh ketidakstabilan bagi sistem kawalan. Sasaran bagi projek ini ialah untuk mengawal dan menghasilkan perkakasan bagi menstabilkan kedudukan menegak untuk sistem "*Inverted Pendulum*" (IP) bagi pendekatan latihan pembelajaran. Sistem ini terdiri daripada putaran rintangan yang dijadikan sebagai masukan kepada sistem untuk mengesan pembolehubah bagi setiap sudut pegerakan bandul. Bagi memastikan bandul dalam keadaan tegak 90^0 darjah, pengawalan motor dalam kedudukan mendatar yang dikawal oleh PWM menjadi mekanisme yang bertindak balas dalam sistem ini. Sebagai gambaran awal bagi menetukan prestasi yang diperlukan oleh bandul terbalik, simulasi melalui perisian MATLAB telah dilakukan dengan menggunakan matematik model sistem. Ketidakmampuan bandul terbalik bagi mengekalkan kedudukan diatasi dengan penggunaan pengawal *"Fuzzy Logic"* ke dalam system. Selepas itu, pembinaan peralatan sistem bandul terbalik ini dilakukan dengan menggunakan peralatan-peralatan yang terpilih. Prestasi kedua-dua bahagian yand dibandingkan, hasil yang dijangkakan dari simulasi dan keputusan sebenar telah di analisis. Prestasi bandul terbalik telah menunjukkan kebolehpercayaan dan ketepatan bandul dalam kedudukan tegak walaupun terdapat sedikit ralat sekitar 0.02% semasa ujian dilakukan.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS

AI	-	Artificial Intelligence
DE	-	Difference Error
E	-	Error
FLC	-	Fuzzy Logic Controller
FSF	-	Full State Feedback
G(s)	-	Transfer Function
IP	-	Inverted Pendulum
LQR	-	Linear Quadratic Regulator
l	-	Length of pendulum
M	-	Mass of pendulum and mobile robot
PID	-	Proporational Integral Derivative

CHAPTER 1

INTRODUCTION

1.1. Motivation

Today, control of inverted pendulum system is shown as a benchmark problem for various controller designs and widely used. It is use in a very detailed, especially in the control laboratories for research and educational purpose. This inverted pendulum system became popular because of the easy preparation and some interesting features such as instability and nonlinearity. The application of this system as show in Figure 1.1 on the next page where it used to missile launchers Segway, rocket launching, luggage carrying Pendubots and etc [6]. In addition inverted pendulum (IP) is needed in the learning approach for the students knowledge, especially in learning Artificial Intelligence (AI) which require practical methods for comprehension in learning.

For instance in University of Technical Malaysia Melaka (UTEM), used the Inverted Pendulum system in Control courses as a part for learning approach in laboratories. The experimental in this learning part are used Inverted Pendulum Rotary types where the limitation of the movement is fixed. In this project, it used in the same system but different concept for Inverted Pendulum (IP) System where by using a Mobile Robot system. Compared of this type with the old version in University, Inverted Pendulum (IP) in system of cart will be functioning in widely limitation range and more suitable for learning approach.

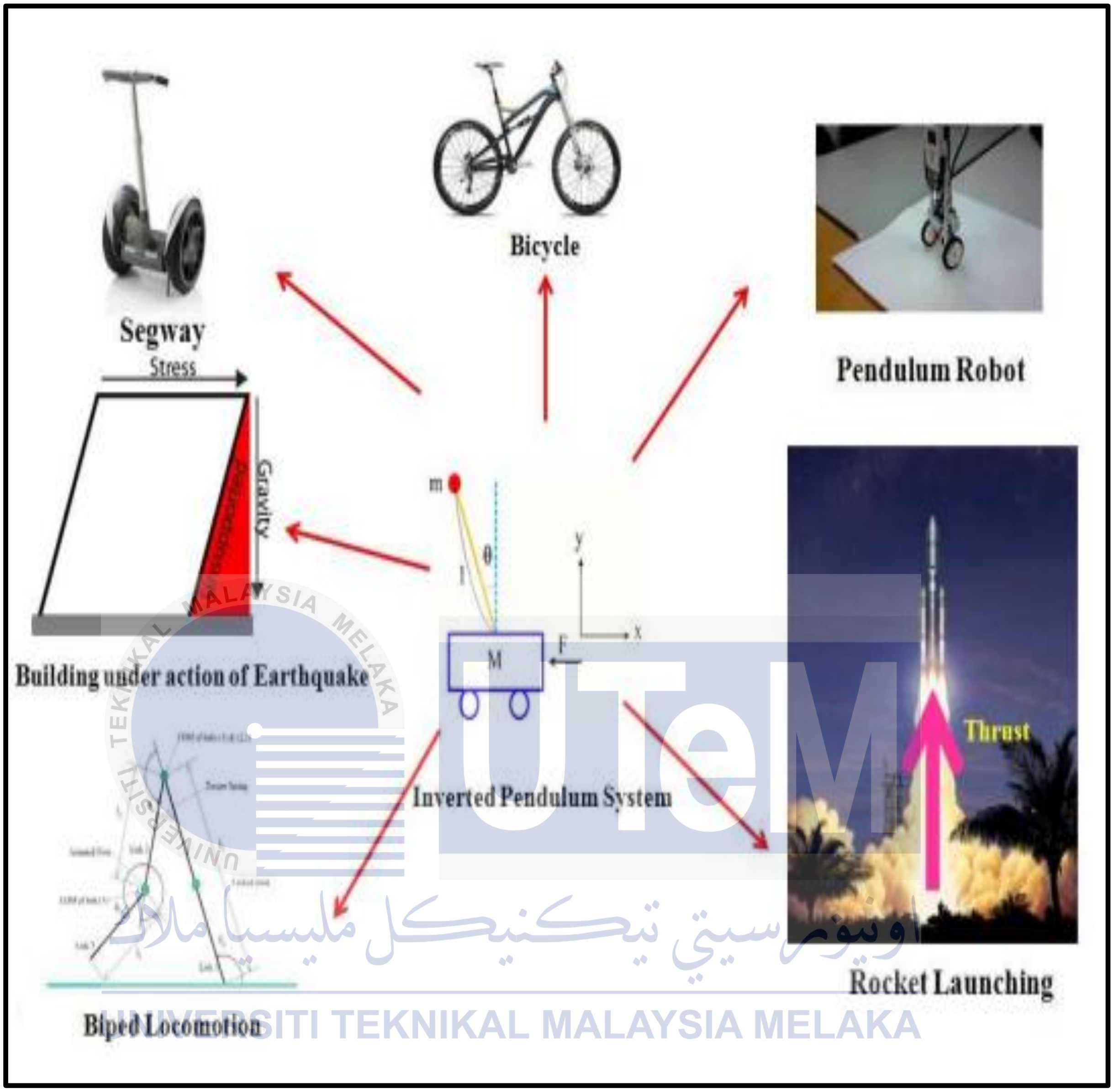


Figure 1.1: Application for Inverted Pendulum system. [6]

1.2. Problem Statement

Inverted Pendulum (IP) is a very complex system and wide application used in industry. Since this system is very complicated, their implementations is the need to ensure the accuracy and efficiency of the Inverted Pendulum (IP) system is always in a stable upright position. In this situation has been described by all researchers that this system is the influence from various aspects, especially in terms of stabilizing the pendulum.

However, for this project there are several problems that must be faced to complete the whole of this Inverted Pendulum (IP) system. The problem is divided into two parts where it started in the execution system by using simulation software development and execution systems in real situation from hardware development. In the software parts, the most importance things is to develop a system to ensure the pendulum always maintain in upright position. This process requires to get proper transfer function G(s) and excellent in drive. All of the variables are describe by the movement of a pendulum which every angle that involve with rate of speed a cart driven to ensure the pendulum does not falling. In fact, this also include mass (*M*) of the pendulum and a cart, length of centroid pendulum (*l*) and friction of cart (*b*) where all the variable is fixed to get transfer function by using mathematical method.

Next problem in this project is faced by hardware part, where the selection of the appropriate tools and equipment to ensure the system is always running smoothly. The device selection is involved in this project including microcontroller circuit and also the device to allow the pendulum to know differ angles depending on the situation. In terms of equipment, the selection of designing a cart, the materials used for the pendulum and also the selection motor to drives a cart from dropping the pendulum. Lastly, the problem is faced from installation software into the hardware part which is to ensure the pendulum always maintain in upward position by using controlling the speed of wheel. Then to strength the result for performance this system, data from simulation will be compared where all the equipment and tools will combine together to get the Inverted Pendulum (IP) systems with movement a cart in full functioning.

1.3. Objective of the project

The objective for this project is

1. To design and develop the control algorithm of controller for Inverted Pendulum (IP) system.
2. To develop hardware of the Inverted Pendulum (IP) system for classroom approach.
3. To analyze the performance of the Inverted Pendulum (IP) system.

1.4. Scope of the project

The scope for this project, it just focuses as a part of learning approach that will cover in Artificial Intelligence subject. A mobile robot that will use in this project is function as a cart where it is easier for learning part proses. The pendulum can only move in X-Y plane and the system fixed in mass of cart (M), mass of pendulum (m) and length of rod (l). For pendulum angle (ϕ) and cart position coordinate (x) it vary depend on to maintain the system**.**

CHAPTER 2

LITERATURE REVIEW

2.1. Controller selection analysis

“Robust Control of Inverted Pendulum Using Fuzzy Logic Controller” – Sandeep Kr. Tripathi, Himanshu Panday and Prerna Gaur. This journal describe about the different design of controller techniques that to analysis the performance and reliability of the system. This is also present to get optimize the work of Inverted Pendulum (IP) system and robust performance for a nonlinear system by using fuzzy logic algorithm. There has two method of controller to archive their research by using simulation MATLAB Simulink namely fuzzy logic controller and Robust control in association with fuzzy produce. This experiment started from mathematical analysis by using newton 2nd law of motion in horizontal and vertical direction applied on free body a cart. The resulting of the mathematical method can get the plan for transfer function of the system where it will see on step respond and also impulse response. However, to get the robust controller there have some sort of method to complete the system such as by applying complex mathematic equation. From the robust equation, new transfer function will produce due to presence of noise and disturbance. The performance of this system where comparing between system simulation on Simulink of fuzzy logic and robust controller. Performance index shows that the robust controller with Fuzzy Logic Controller much better than conventional Fuzzy Logic Controller. [1]

“Real-Time Controlling of Inverted Pendulum by Fuzzy Logic” – Yanmei Liu, Zhen Cheng Dingyu and Xue Xinhe Xu. This journal explain about to balance a real pendulum in the position and also built the mathematical model for fuzzy logic controller. The method of this system using MATLAB Simulink to show real time of applying fuzzy logic controller to get the good performance of system. Due to controlling is difficult to stabilize the pendulum in short time, Takagi-Sugeno has been choose for data-driver of the system to functioning in stabilizing control and test the system with semi-physical simulation. The system consist of an Inverted Pendulum with type hinged on a cart rail which free movement in *x* direction. For mathematical model, these systems apply newton 2nd law where it by assuming neglected for nonlinear friction applied to the linear cart. These experiments showed controlling result in Simulink on simulation in MATLAB software. This will consist by using MATLAB RTW real time toolbox where it to develop and build a real time control then to add in simulation structure of Inverted Pendulum. The resulting by using RTW real time toolbox in the simulation structure just takes 1 sec to bring a real pendulum in upright position. From the overall in this system is state fuzzy controller proving more effective and precise of angular control and this result is shown from Figure 2.1 below.[2]

Figure 2.1: Result of experiment the Inverted Pendulum [2]

"Modeling and Control of a Rotary Inverted Pendulum Using Various Methods, Comparative Assessment and Result Analysis" – Md. Akhtaruzzaman and A.A.Shafie. This conference paper discuss about the steps to design controller for rotary inverted pendulum type and to analyze the controller system. This system will be compared in three control system namely 2DOF Proportional-Integral-Derivative (PID) controller, Full State Feedback (FSF) and Linear

Quadratic Regulator (LQR) to get the best performance of rotary motion. The method in this system started by using mathematical modeling of rotary motion where the derivation system in dynamics equation. Then to complete this system, controller has been design in terms of t hree controllers by using Simulink in simulation MATLAB software. However, for this over all result state that the LQR controller more suitable to control rotary type because of it easier to swing up the pendulum and maintain in upright position. By following in this concept by using rotary motion is shown on Figure 2.2 below. [3]

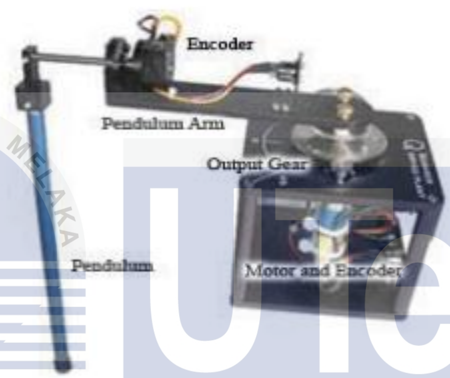


Figure 2.2: Rotary motion of Inverted Pendulum [3]

“Control of Non-Linear Inverted Pendulum Using Fuzzy Logic Controller” – Arpit Jain, Deep Tayal and Neha Sehga. This journal describe about to implement the fuzzy controller and apply it into Inverted Pendulum system. In this system by using a cart concept, mathematical modeling is produce by using derivation of Lagrange’s equation but in this equation not really mention in more details of the system. For the designing fuzzy logic controller, Simulink has been test in simulation MATLAB software by using 3 membership functions namely pendulum angle and angular velocity on input variable and for output variable it used force. The result is to check the stability and validation of performance by applied to pendulum.[4]

“Fuzzy Logic Control vs. Conventional PID controller of an Inverted Pendulum Robot” – M.I.H. Nour, J.Ooi and K.Y.Chan. This conference paper is focus about to implement and optimize the fuzzy logic control algorithms for to balance the inverted pendulum. In other that these systems want to reduce the computation time of the controller. This system will be compared the performance between fuzzy logic controller and PID controller by using simulation in MATLAB software. By applied simple mathematical model by using drawing of the cart, the equation of this require to obtain the *m* is mass of the pendulum, μ is the coefficient of friction and *I* is the moment of inertia the pendulum. In fuzzy logic system, there have required 3 membership functions namely angle, angular velocity and position for input variable and for output variable, membership function is used force. In analysis by using Simulink, Fuzzy logic controller more robust comparing conventional PID controller where fuzzy give smallest overshoot of the system and shorter settling time. [5]

2.2. Comparison system selection

From Table 2.1 below will be explanation about the overview concept of the Inverted Pendulum system according to five (5) journal papers.

Table 2.1: Comparisons method, controller and type of the system

Criteria	Paper 1 [1]	Paper 2 [2]	Paper 3 [3]	Paper 4 [4]	Paper 5 [5]
Method	1. Newton 2nd Law 2. MATLAB Simulink 3. Robust Mathematical 4. Mamdani Rules	1. Newton 2nd Law 2. MATLAB Simulink 3. RTW real time toolbox 4. Takagi-Sugeno	1. Dynamic equation 2. MATLAB Simulink	1. Lagrange equation 2. MATLAB Simulink	1. Newton 2nd law 2. MATLAB Simulink 3. Takagi-Sugeno
Controller	1. Fuzzy Logic Controller 2. Robust with FLC algorithm 3. PID Controller	1. Fuzzy Logic Controller	1. PID Controller 2. FSF Con troller LQR Controller	1. Fuzzy Logic Controller	1. Fuzzy Logic Controller 2. PID Controller
Type	1. Inverted Pendulum in Cart system	1. Inverted Pendulum in Hinged on a Cart of Rail	1. Inverted Pendulum in Rotary Motion	1. Inverted Pendulum in Cart concept system	1. Inverted Pendulum in Cart drawing concept

Table 2.2: Comparisons accuracy and performance of the system

Criteria	Paper 1 [1]	Paper 2 [2]	Paper 3 [3]	Paper 4 [4]	Paper 5 [5]
Accuracy	1. Robust with FLC and FLC conventional nearly on maintain position 2. PID more overshoot before to maintain position	1. Result FLC in graph decrease in term of overshoot and improve in settling time	1. LQR and FSF improve in theta and alpha graph compared between PIC	1. Graph for with noise try to nearly in maintain position	1. PID controller more overshoot and increase settling time
Performance	1. Robust with FLC performed in Rise Time, Overshoot and Settling Time 2. PID less performed between FLC	1. FLC more effective and feasible of the angular control	1. Both of LQR and FSF effective to control rotary motion but LQR more robust 2. PID not robust in controlling rotary motion	1. Compared FLC with noise and without noise	1. FLC more robust comparing convention al PID controller 2. FLC smallest overshoot and shorter settling time

2.3. Summary of the system selection

Based on research in five (5) journal has described before this, each of the paper evaluated by following method, controller, performance, accuracy and type of system. Through for this project, the research done to get the best performance and the method in term of to controlling the Inverted Pendulum. For this project, [1] has been choosing where it gives suitable the all criteria to develop control and also to choosing the type of this project. These criteria will be based solely on the terms of the method used such as mathematical equation and also type of the system. However, on controlling system this project just focused only with fuzzy logic controller without adding other controller such as robust controller. This is because to shows the performance by using FLC conventional and to prove this controller is the best choice. This proven has been shown in [1] whereas in terms of accuracy and performance table that FLC less overshoot and settling time comparing between PID co ntrollers.

CHAPTER 3

RESEARCH METHODOLOGY

1.1. Introduction

This chapter will describe in detail the development system and designing Inverted Pendulum (IP) from the initial stage to implement the results of the system. In this project has been divided into two major parts that need to be focused, it is included in the simulation part as well as the hardware part. In order to produce this project smoothly, two type of software have been used namely MATLAB where to obtain the results from simulation and SolidWork software for designing projects

Before starting part of development and designing, the most important things in generating this methodology is plans to produce a detailed Inverted Pendulum system to get good results. This is because to ensure the project will be develop step by step according the plan of procedure. Base on flow chart in Figure 3.1 as show clearly proses on this flow of project.

1.2. Flow chart of the system

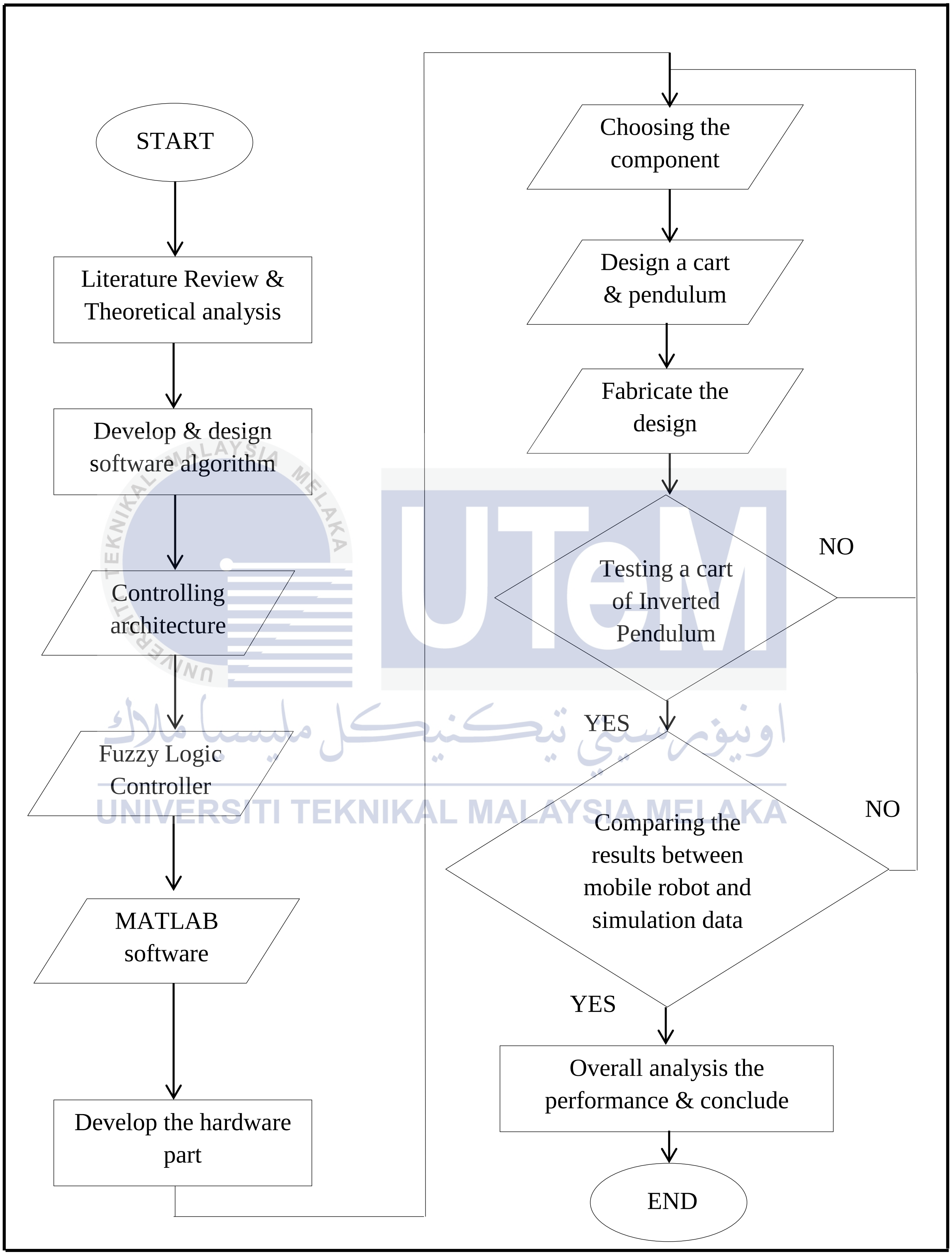


Figure 3.1: Flow chart of Inverted Pendulum system

In this flowchart of Inverted Pendulum began with a literature review where it used for the research the system concept. Next, the project is implemented in the software which invoves the simulation and continue with produce hardware development. Both of these systems were tested to show the performance of the system by using a mobile robot as a demonstration. This test depends on the function of proses, if the resulting test failed so it must check based on the turning for fuzzy controller, programming or connection circuit. Then if the resulting test is function, it continues by comparing simulated with actual system to find out and analyze the performance of the system. However, when comparing the two systems also fail then this system should be check in the system done before.

1.3. Controller Design

These proses began by studying and analyzing controller for the Inverted Pendulum (IP) system. To ensure this controlling has a good flow, there are architecture to understand the whole steps where is shown in Figure 3.2 below.

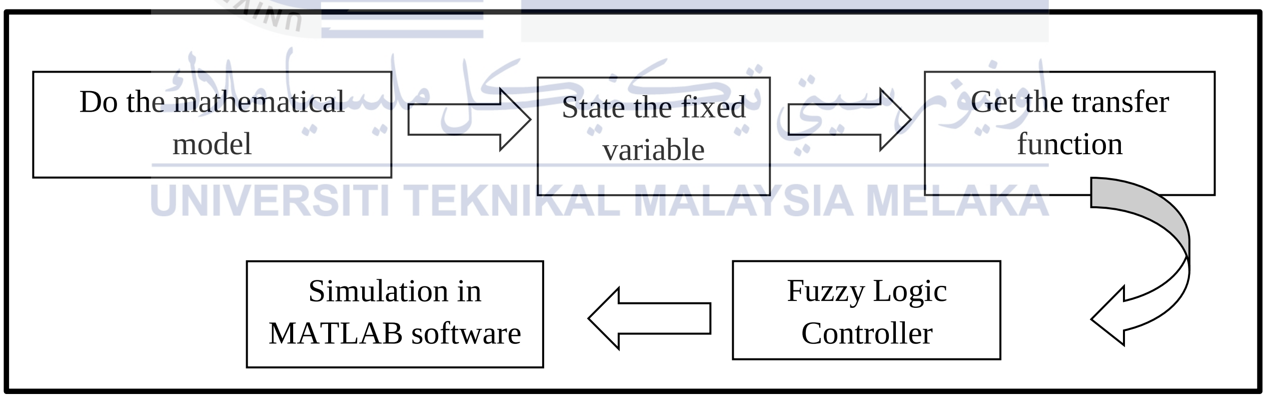


Figure 3.2: Controlling architecture system

1.3.1. Controlling Architecture

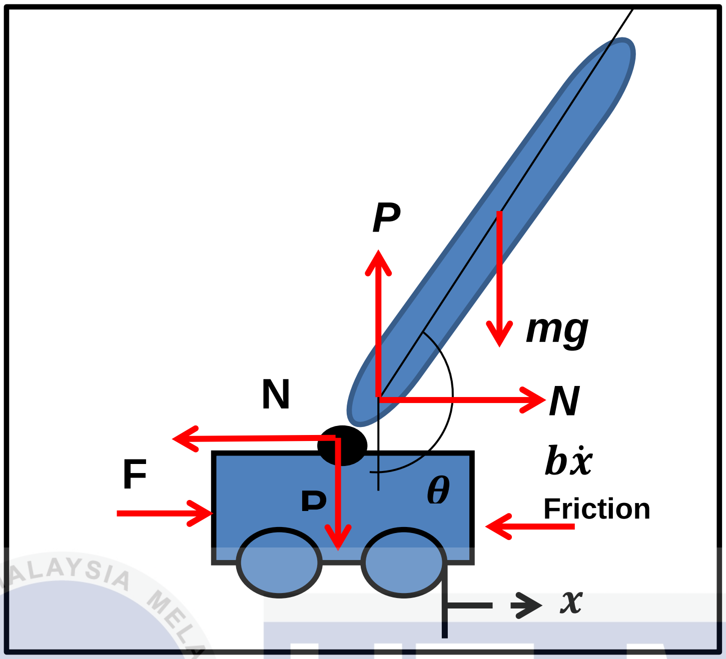


Figure 3.3: Free body diagram combination with two parts

In Figure 3.3 above has shown the free body diagram system that will be used in this project. This system used angle and angular velocity of pendulum as an input variable and the output of this system will manage as a voltage for speed motor. This combination is to make sure the pendulum is always maintain in upright position without falling.

1.3.1.1. Mathematical Model

To ensure control system always maintain in upright position, mathematical model was create the result of Simulink in MATLAB software before it is transferred in the real system. The equation of this is to obtain the transfer function by using mathematical model [1] to comp lete for Simulink of Inverted Pendulum system.

By following in Figure 3.3 shows above were referring to used Newton 2nd law equation of motion a cart. This movement in linear direction condition will be neglect cause of coulomb friction. The following dynamic equations in this cart, horizontal and vertical direction are sta te:

I. Horizontal direction

Force in free body diagram

$$M\ddot{x} = F - b\dot{x} - N$$

$$F = M\ddot{x} + b\dot{x} + N \quad (3.1)$$

Due to force exerted in horizontal direction, moment on the pendulum as determined;

$$N = m\frac{d^2}{dt^2}\,(x + l\sin\theta) \quad (3.2)$$

Summing force in free body diagram, get an equation for N

$$N = m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta \quad (3.3)$$

By substitute equation (3.1) into (3.3), first equation on this system will get as follow;

$$F = M\ddot{x} + b\dot{x} + m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta$$

$$F = (M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta \quad (3.4)$$

II. Vertical direction

To get 2nd equation of motion, by summing the force perpendicular to the pendulum with throughout of the axis

$$P\sin\theta + N\cos\theta - mg\sin\theta = ml\ddot{\theta} + m\ddot{x}\cos\theta \quad (3.5)$$

To eliminate the P and N terms, sum the moments about the centroid of the pendulum then get the following equation.

$$-Pl\sin\theta - Nl\cos\theta = I\ddot{\theta} \quad (3.6)$$

By combining equation (3.5) and (3.6), second governing equation will get.

$$(I + ml^2)\ddot{\theta} + mgl\ \sin\theta = -mlx\ \ddot{\cos}\,\theta \quad (3.7)$$

$$I = \frac{1}{3}ml^2 \quad (3.8)$$

Since to maintain the pendulum in upright equilibrium position, $\theta = \pi$ will assume that the system nearly to equilibrium. Let $\emptyset$ as describe the deviation of the pendulum to balance, $\theta = \pi + \emptyset$ as derivation for position from equilibrium. Again using a small deviation ($\emptyset$) from equilibrium, can use the following with a small angle estimate of the nonlinear function in this system equation:

$$\cos\theta = \cos(\pi + \emptyset) \approx -1$$

$$\sin\theta = \sin(\pi + \emptyset) \approx -\emptyset$$

$$\dot{\theta}^2 = \dot{\emptyset}^2 \approx 0 \quad (3.9)$$

By substitute the equation (3.4) and (3.7) into (3.9), it will archive two linearized equation of motion and note u is the substitute from input F.

$$(M + m)\ddot{x} + b\dot{x} - ml\ddot{\emptyset} = u \quad (3.10)$$

$$(I + ml^2)\ddot{\emptyset} - mgl\emptyset = ml\ddot{x} \quad (3.11)$$

III. Transfer Function

To obtain the transfer function for pendulum in maintain upward position, Laplace transform are used in this system equation were assuming zero initial condition. The equation of (3.10) and (3.11) changing as shown below.

$$(I + ml^2)\emptyset(s)s^2 - mgl\emptyset(s) = mlX(s)s^2 \qquad (3.12)$$

$$(M + m)X(s)s^2 + bX(s)s - ml\emptyset(s)s^2 = U(s) \qquad (3.13)$$

From the equation (3.12), by rearranging the equation it will get equation of position *X(s)*

$$\frac{\emptyset(s)}{X(s)} = \frac{mls^2}{(I + ml^2)s^2 - mgl}$$

$$X(s) = \left(\frac{(I+ml^2)s^2 - mgl}{mls^2}\right)\emptyset(s) \qquad (3.14)$$

Rearranging the equation (3.13) and substitute (3.14) into the equation

$$(M + m)X(s)s^2 + bX(s)s - ml\emptyset(s)s^2 = U(s)$$

$$U(s) = (M + m)\left[\left(\frac{(I+ml^2)s^2 - mgl}{mls^2}\right)\emptyset(s)\right]s^2 + b\left[\left(\frac{(I+ml^2)s^2 - mgl}{mls^2}\right)\emptyset(s)\right]s - ml\emptyset(s)s^2 \qquad (3.15)$$

$$U(s) = \left((M + m)\left[\left(\frac{(I+ml^2)s^2 - mgl}{ml}\right)\right] + b\left[\left(\frac{(I+ml^2)s^2 - mgl}{mls}\right)\right] - mls^2\right)\emptyset(s) \qquad (3.16)$$

$$U(s) = \left(\frac{\left((M+m)\left[\left(\frac{(I+ml^2)s^3 - mgl}{ml}\right)\right] + b\left[\left(\frac{(I+ml^2)s^2 - mgl}{mls}\right)\right] - mls^3\right)}{s}\right)\emptyset(s) \qquad (3.17)$$

Finally after rearranging the equation, transfer function becomes the following.

$$G(s) = \frac{\emptyset(s)}{U(s)} = \left(\frac{s}{\left((M+m)\left[\left(\frac{(I+ml^2)s^3 - mgl}{ml}\right)\right] + b\left[\left(\frac{(I+ml^2)s^2 - mgl}{mls}\right)\right] - mls^3 \right)} \right) \tag{3.18}$$

1.3.1.2. Physical Parameters

There are some of fixed variable where used in this system to get the value of transfer function before doing the Simulink process. The parameter used in the hardware part of Inverted Pendulum system. As mentioned previously, the following parameters are used by the hardware specifications of the mobile robot as shown in Table 3.1 below.

Table 3.1: The Physical Parameter of the system

No	Parameters	Description	Value	Units
1	*M*	Mass of the cart	0.95	Kg
2	*m*	Mass the pendulum	0.11	Kg
3	*b*	Friction of the cart	0.1	N/m/sec
4	*L*	Length of the pendulum	0.5	m
5	*I*	Moment of Inertia (pendulum)	0.0452	Kg-m2
6	*g*	Acceleration due to gravity	9.81	m/s/s

1.3.1.3. Transfer Function

To obtain the transfer function to maintain the upward position of pendulum system, by using equation (3.14) and substitute into equation (3.18) will get the overall of system. This equation will be substitute the value from the physical parameter where it is a fixed variable use in this system.

$$X(s) = \left(\frac{(I+ml^2)s^2 - mgl}{mls^2}\right)\emptyset(s)$$

(3.14)

$$X(s) = \left(\frac{(0.0452 + (0.11)(0.5)^2)s^2 - (0.11)(9.81)(0.5)}{(0.542)(0.5)s^2}\right)\emptyset(s)$$

$$X(s) = \left(\frac{0.0727s^2 - 0.53955}{0.055s^2}\right)\emptyset(s)$$

(3.19)

From the equation (3.19), substitute into equation (3.18)

$$G(s) = \frac{\emptyset(s)}{U(s)} = \left(\frac{s}{\left((M+m)\left[\left(\frac{(I+ml^2)s^3 - mgl}{ml}\right)\right] + b\left[\left(\frac{(I+ml^2)s^2 - mgl}{mls}\right)\right] - mls^3\right)}\right)$$

(3.20)

$$\frac{\emptyset(s)}{U(s)} = \left(\frac{s}{\left((1.5+0.542)\left[\left(\frac{0.0727s^3-0.53955}{0.055}\right)\right]+(0.1)\left[\left(\frac{0.0727s^2-0.53955}{0.055}\right)\right]-(0.11)(0.5)s^3\right)}\right)$$

(3.21)

$$\frac{\emptyset(s)}{U(s)} = \left(\frac{s}{(1.4011273s^3-10.3986s+0.132182s^2-0.981-0.055s^3)}\right)$$

(3.22)

From the equation (3.22), simply to get the equation were become the plant transfer function of the system

$$\frac{\emptyset(s)}{U(s)} = \left(\frac{s}{(1.3461273s^3-10.3986s+0.132182s^2-0.981)}\right)$$

(3.23)

$$G(s) = \frac{\emptyset(s)}{U(s)} = \left(\frac{0.74287s}{s^3+0.098194s^2-7.724827s-0.72876}\right)$$

(3.24)

1.3.2. Controlling Setup

1.3.2.1. FLC design

According in this project will be used Fuzzy Logic Controller, there are several part that mentioned by following of overview in this Inverted Pendulum system. This system whereby began with four (4) elements importance thing in fuzzy logic controller. As mention before this, this shows in Figure 3.4 below to complete the Inverted Pendulum (IP) system.

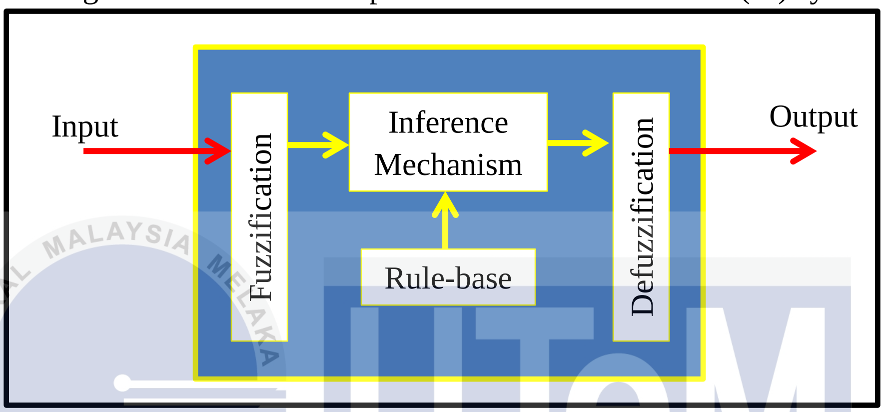


Figure 3.4: Basic structure of Fuzzy Logic Controller

From the figure above, there have several steps for mention each part whereby will be used in this project as follow.

1.3.2.2. Input

In this project, two inputs are used in this system is by change the angle motion of the pendulum as an error and the angular velocity as the difference error. This crisp value will be transform to fuzzification of the system to produce fuzzy value.

1.3.2.3. Fuzzification

Fuzzification is a process will produce to convert the crisp value or quantity such as sensor to the fuzzy value. This converted make the system can be defined as fuzzy system with membership function. In this project will used this IF-THEN rule of fuzzification that to mention in fuzzy system.

1.3.2.4. Membership Function

Membership functions as determined for point in the input space where it is membership value between 0 and 1. There is having several types of membership functions but in this project used Triangular Membership Function. These Triangular shape is simple to produce and easy to getting the accurate value.

1.3.2.5. Rule-based system

In this project, the system is implemented by using the principle of IF-AND-THEN rule and the types of rule base as used Mamdani Fuzzy Rule Base System. These is because connective "AND" is direct from human language with using fuzzy intersection and same as the Mamdani types where there are direct consequence of the human. Other rules will be represented in form table as shown in Table 3.2 below that know as Fuzzy Associative Memory (FAM) table.

Table 3.2: Rule Base for the Inverted Pendulum

E \ DE	VNEG	NEG	ZE	POS	VPOS
VNEG	VNEG	VNEG	VNEG	NEG	ZE
NEG	VNEG	VNEG	NEG	ZE	POS
ZE	VNEG	NEG	ZE	POS	VPOS
POS	NEG	ZE	POS	VPOS	VPOS
VPOS	ZE	POS	VPOS	VPOS	VPOS

Where,

VNEG – Very Negative; NEG – Negative; ZE – Zero; POS – Positive; VPOS – Very Positive

1.3.2.6. Inference Mechanism

The consequence is to reshape the implication process system after the fuzzification process. In this project, max-product method had been choosing to archive the result of the process. To complete this process, aggregation must be used to obtaining overall consequent. This aggregation usually used in output process where it is combining from the implication process to get one fuzzy set.

1.3.2.7. Defuzzification

Defuzzification has a several method but in this project only choose max membership method to get the final crisp value. This is because the implication and aggregation result shown as the max-product method whereby it is suitable to use the max membership method that has a limited to peak output function only.

1.3.2.8. Output

To get the results of fuzzy system, the output is the important to the controller where it will be transformed to plant (TF) parts so that each output value can be changed as a whole system This system shows in Figure 3.5 below for the overview of the Inverted Pendulum system. The plan will be implement the transfer function of the system that as a mention before this in equation (3.24).

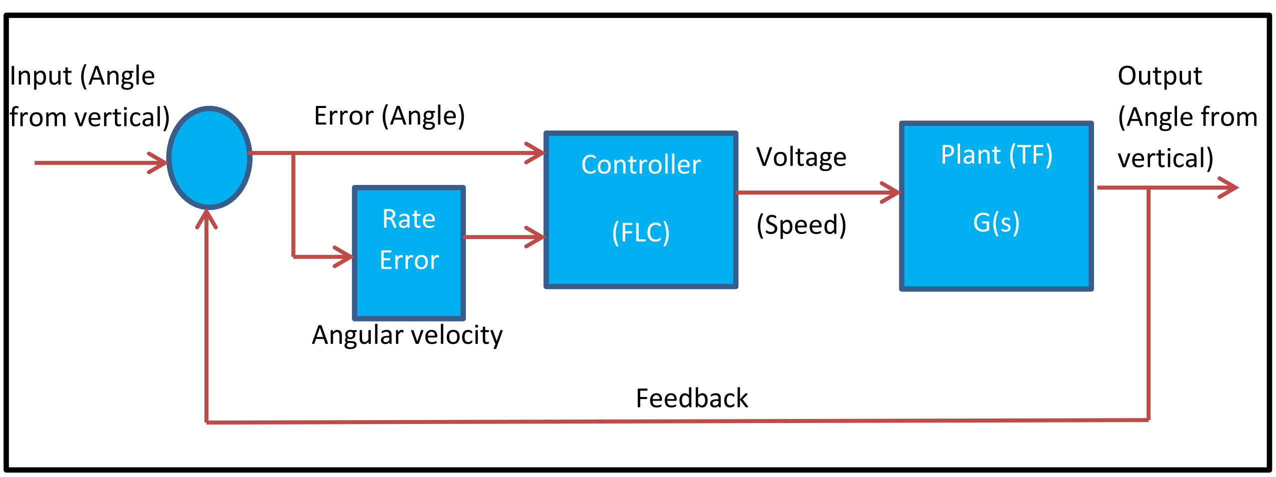


Figure 3.5: Block diagram for Inverted Pendulum system.

1.3.3. Controlling Analysis

This section started by using MATLAB software whereas it used to perform the simulation result based on mathematical method. In this project, the Simulink will be test the transfer function of the Inverted Pendulum by using the equation (3.24) as mentioned before. Based on Figure 3.6, 3.7 and 3.8 below, respectively shown the Inverted Pendulum system without using fuzzy logic controller, system with fuzzy logic controller and system FLC have a noise.

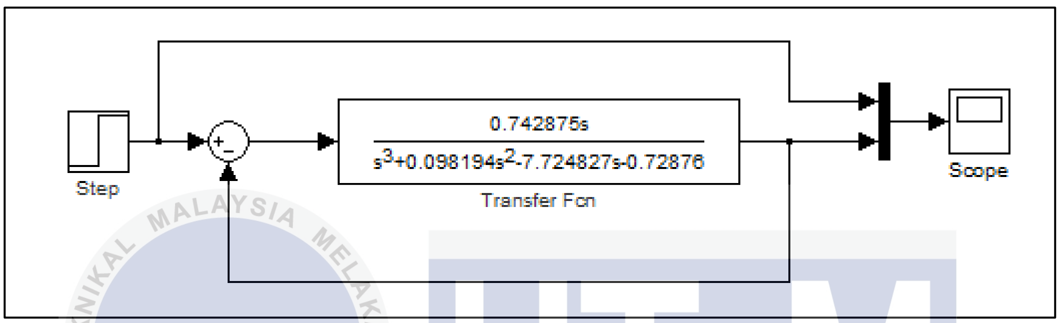


Figure 3.6: Inverted Pendulum system without FLC

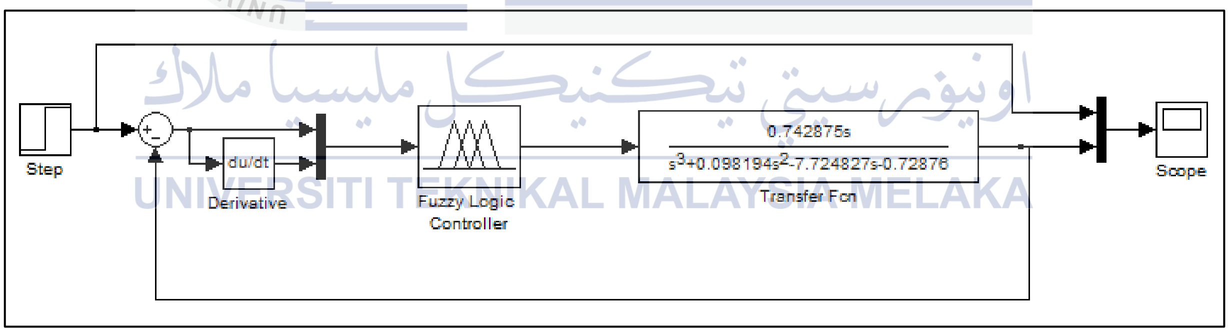


Figure 3.7: Inverted Pendulum system with FLC

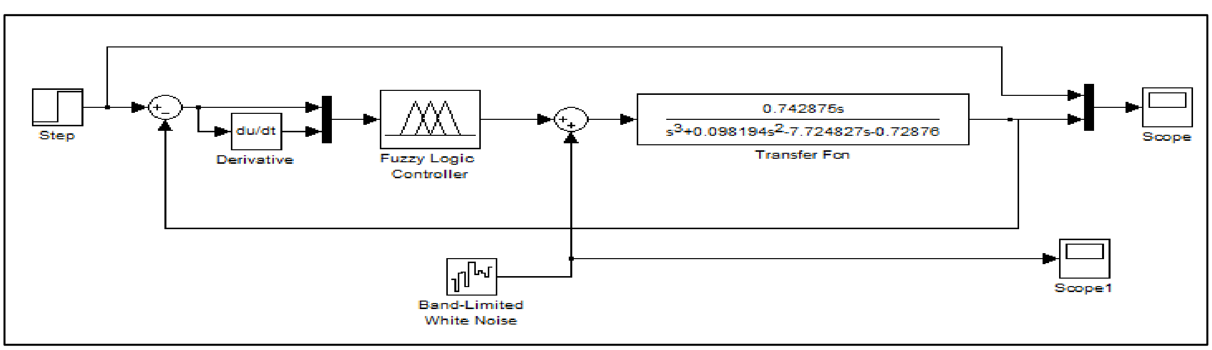


Figure 3.8: FLC system with noise

1.3.3.1. FLC using MATLAB software

From the Figure 3.7 that has been produce the Inverted Pendulum system using FLC, by using the FIS function window where it will be implemented the membership function to complete the system. This system will produce two (2) variable of function and one (1) output variable whereby to mention based on angle, angular velocity and position. The figure below shows about the overview of the system by in terms of fuzzy controller.

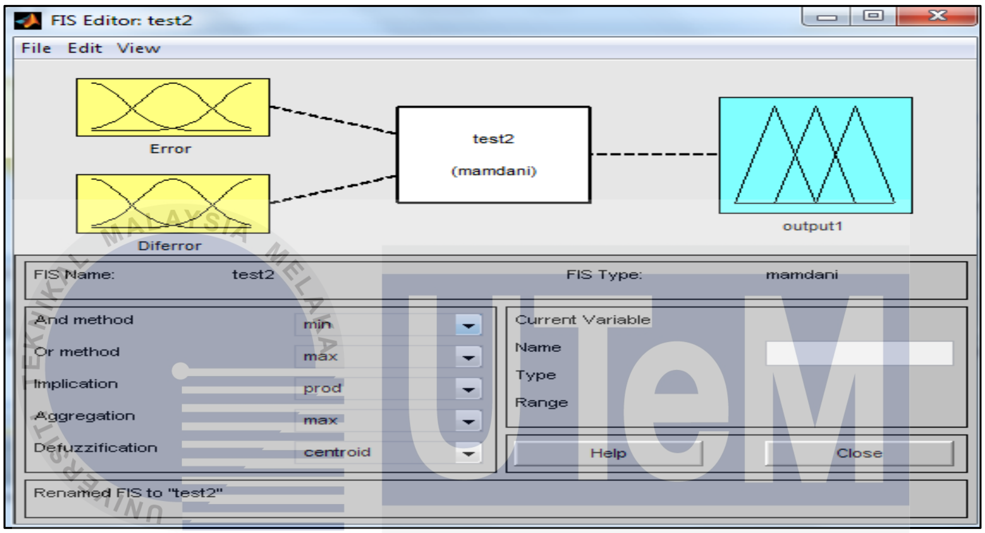


Figure 3.9: Variable input and output in IP system

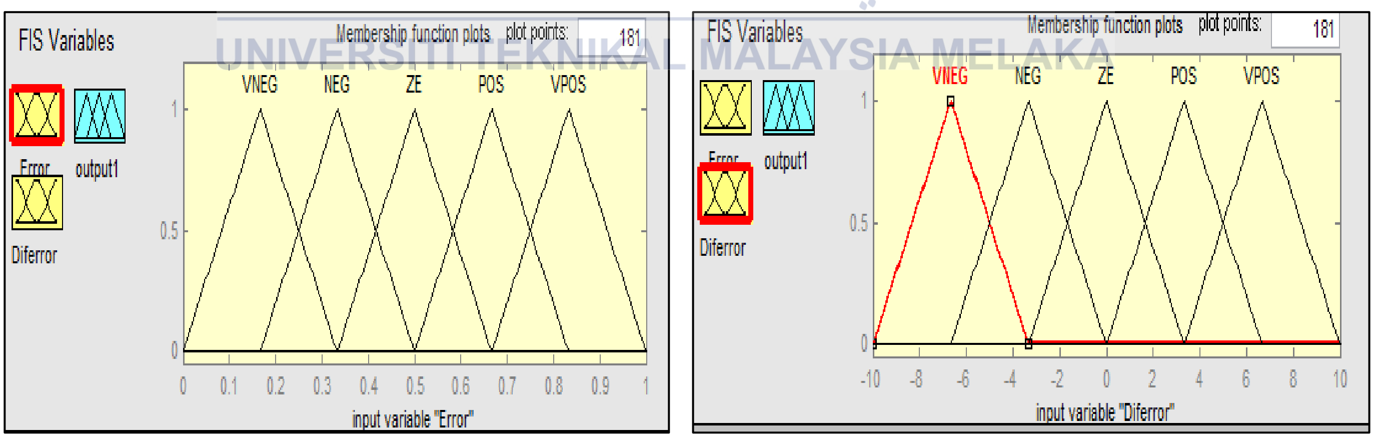


Figure 3.10: Error and Difference error input in membership function

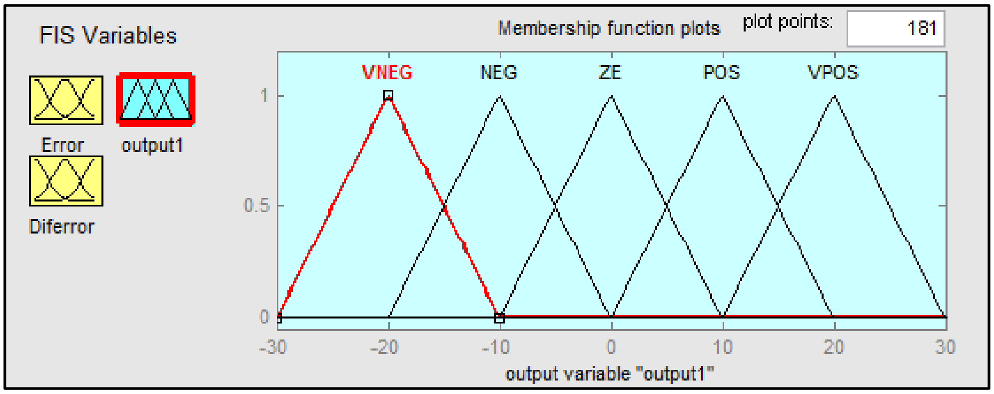


Figure 3.11: Output in membership function

In Figure 3.9 above shows the input variable for this project system by using triangle membership function and this function same as using output variable in figure 3.10. After done tuning in membership function, for next method is to create a rule by following Mamdani rule for this Inverted Pendulum system. This is shown in Figure 3.11 below

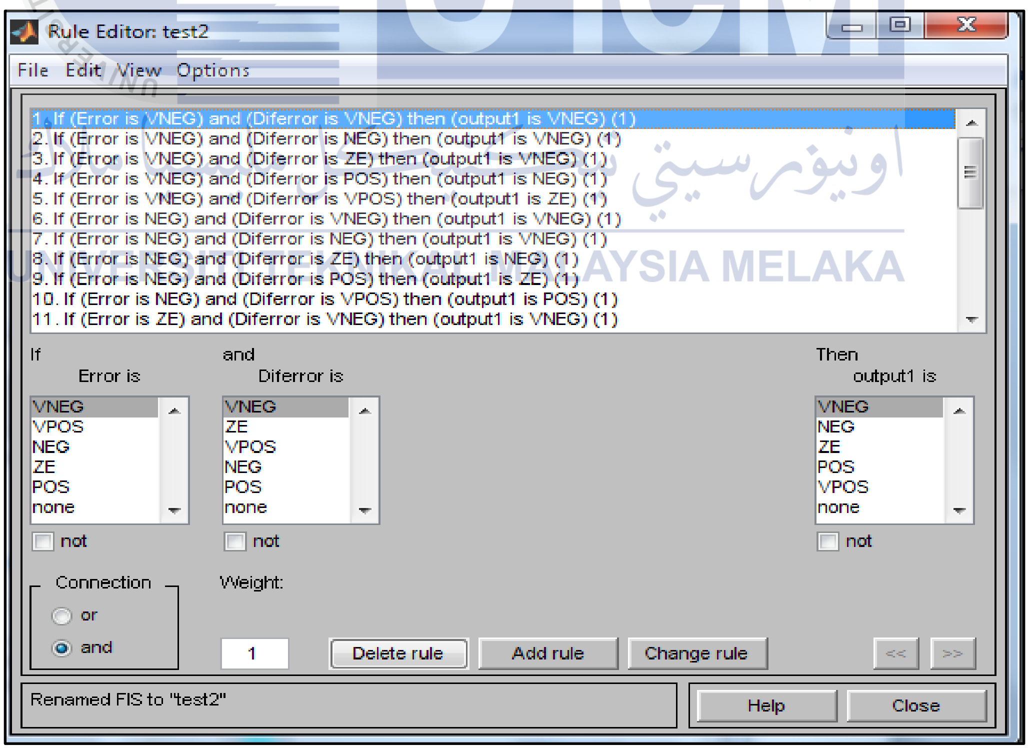


Figure 3.12: Rule base

Then by following this rule base, it will implement in transfer function on Simulink to start the simulation process. The simulation result will be shown on graph simulation and the reading of the system by taken on step response graph.

1.4. Hardware Parts

In this project hardware parts is also importance things after software development. This is because in oder to compare the results of using the MATLAB Simulink software process between the real part of the Inverted Pendulum (IP) system. These parts began with choosing the right device to maintain the pendulum in upright position. In Figure 3.13 below shows the configuration of the system for the hardware development of Inverted Pendulum (IP) system.

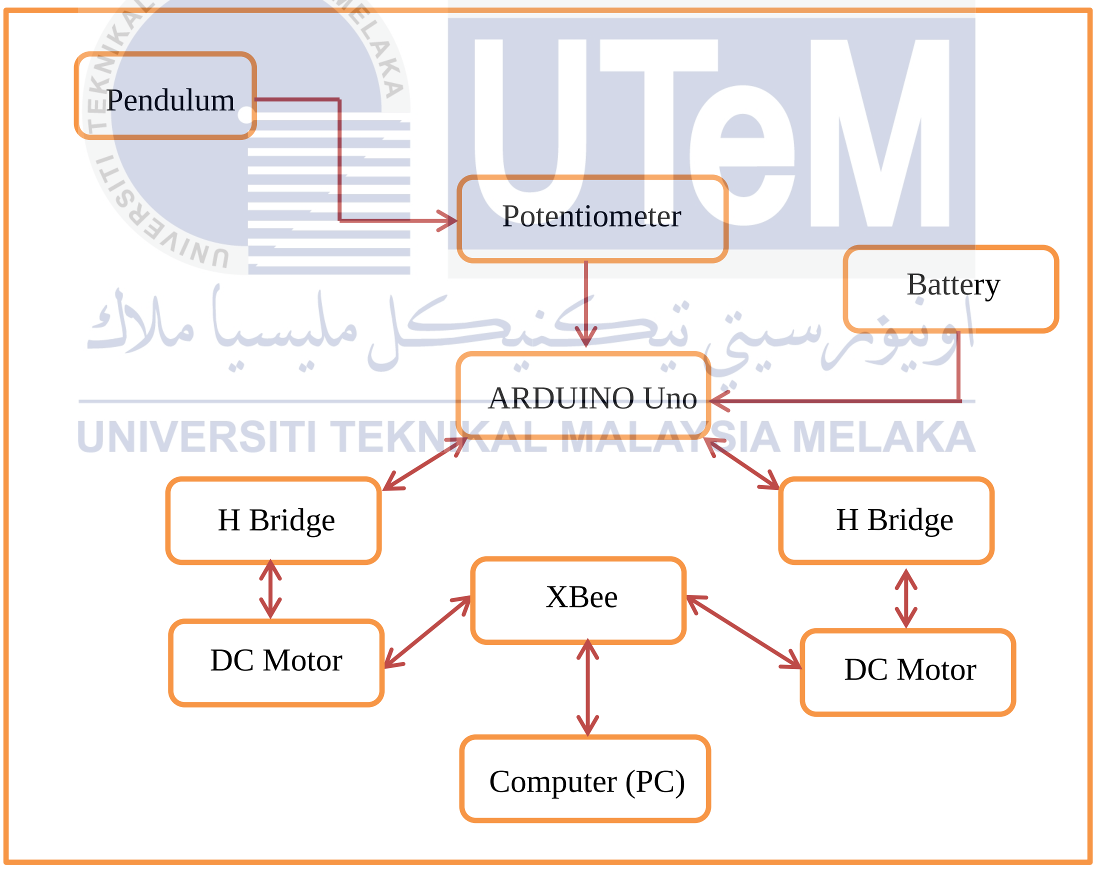


Figure 3.13: Configuration Inverted Pendulum System

In Figure 3.12, as show potentiometer has used for the main detection changing the angle to maintain the pendulum in upward position. This potentiometer to detect the angular rate for pendulum movement then the signal will be transferred to ARDUINO as function a microcontroller the system. This reading to make the DC motor to control the speed and velocity of cart than it will be transferred to Xbee as a wireless to get the data. Then the XBee device will transfer the data to computer (PC) and save the data to plot the graf .

1.4.1. Component Selection

1.4.1.1. Potentiometer

This potentiometer 150K ohm has been used in this project as sensor of the pendulum. This potentiometer will detect the angle whereby pendulum in this project might be varying in digital value then will convert to angle direction . The choosing this device for Inverted Pendulum project is to make the system always in maintains upright position on the cart.

Figure 3.14: Potentiometer

1.4.1.2. ARDUINO UNO

This Inverted Pendulum system has been choosing the ARDUINO UNO board as a microcontroller. This is because ARDUINO board is a complementary component to simplify the programming and to make easier connection to other circuit. It is also that consists of 16 digital Input Output pins and also include for fast speed operating system where it cover in 16Mhz clock speed.

Figure 3.15: ARDUINO UNO board

1.4.1.3. H Bridge Dual motor driver

For this system, motor is an importance device to move a cart then to control the pendulum from falling. To expedite this motor, motor driver are needed to show the direction of movement of the motor. In this project, the device L298 type Dual H-Bridge motor driver has selected than it will receive signal from the microcontroller to move a cart

.

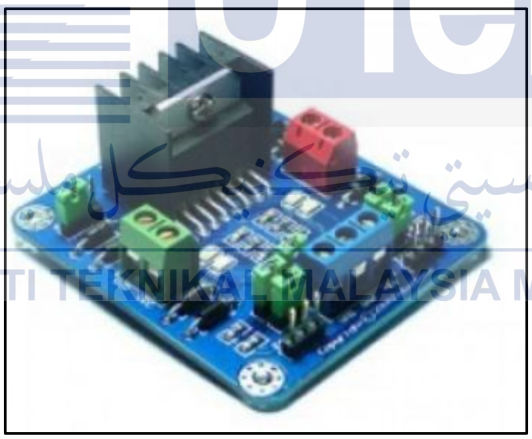


Figure 3.16: L298 Dual H-Bridge motor driver

1.4.1.4. DC motor

Motor is an importance device to move a cart then to control the pendulum from falling. Selected motor to move the cart of Inverted Pendulum system by using DC motor types SPG30-20K. This specification of motor that consist high torque, moderate in speed and use 12V for the supply.

Figure 3.17: DC motor

1.4.1.5. XBee Series 1

This device used for transfer data from mobile robot (hardware) then it will link to computer (PC) as a wireless functioning. This selection for series 1 has been choosing because this series is more easily functioning compare with others. This XBee device from transmitter (hardware) data transfer it to receiver connection (computer) where it will be taken from software X-Ctu to show the result or data from mobile robot functioning.

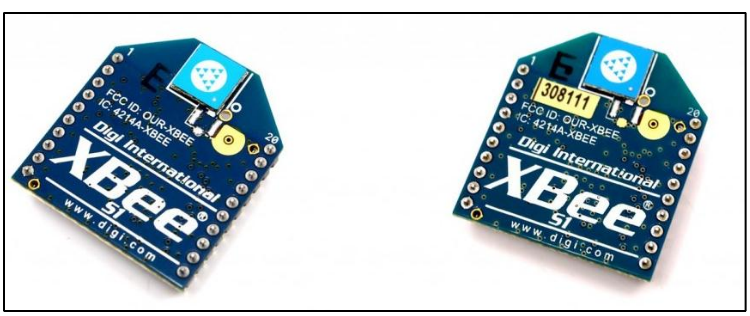


Figure 3.18: XBee Series1 transmitter and receiver

1.5. Experimental Setup

This section to present about the sequence includes the setup for Inverted Pendulum (IP) system using mobile robot from the beginning until the last part which is to analyze the performance of system. A block diagram of this system as shown in Figure 3.19 below which is the components involved are driver motor, DC motor, a cart with pendulum, potentiometer as a feedback and a computer with Simulink MATLAB software for signal processing.

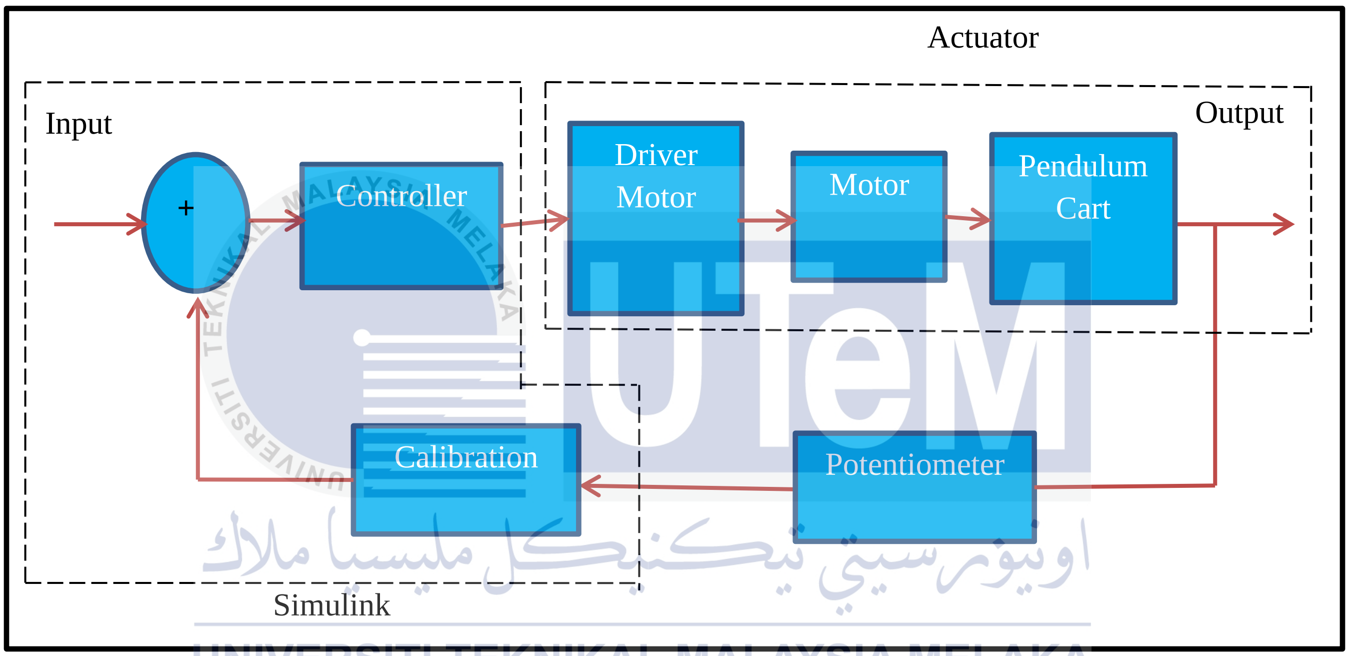


Figure 3.19: Pendulum Cart system Block Diagram

1.5.1. Software

A system of this project began by using Simulink MATLAB software which is to implement the controller and to test the performance of controller before it transferred into hardware section. This test come out with three (3) different criteria which is to test the system without the FLC controller, system using FLC controller and test the system if have disturbance by including the noise. All this part was designed by using FLC algorithms as shown in figure 3.6, figure 3.7 and figure 3.8 in controlling analysis section. After the testing in simulation done, it will implement this section into programming part.

Due to the board of the IP system becomes a central several only for the mobile robot, it required to fast movement and high speed operating to handle some of the calculations and complex control algorithm. The Arduino Uno R3 would be more than enough to control the mobile robot which is has 16Mhz clock speed. In additional, this microcontroller has own programming source where it easy to write the coding for this mobile robot. Testing for this programming be used Proteus 7 Professional with Arduino Uno board circuit to know the functioning of the system. After it functioning on software, it will implement into mobile robot to get the accuracy and performance of pendulum to always maintain in upright position.

The last part of this section is back to used Simulink software for signal processing and controller implementation. Signals from hardware or mobile robot are connected to Arduino Uno as an interface to manage and calibrate the feedback data or signal. This part is to analyze the performance of mobile robot and also to compare the data between simulation parts.

1.5.2. Hardware

This section will cover about actuator components. This began with signal from output controller into the driver motor. This is to driving the motor robot or pendulum cart which consists to control the speed of the motor. The speed of the motor will be effect for stabilization of pendulum in upright position. This system needed to consist of two gear motors with enough for speed and torque to be able to right the robot if it rotates away from vertical and wheels that had enough grip to be able to maintain the pendulum.

For this project, L298 H-Bridge motor driver was selected which is to drive the DC geared motor type SPG30-20K which 78.4 mN.m toque rate and a rated speed 185rpm. This system operation by used 12V power supply battery to support power drive of DC motor to get good in performance. Figure 3.20 below as shows the mobile robot to test the performance of accuracy of pendulum with implement the protractor to facilitated data taken. The detail procedure for this setup was provided by referring to the Appendix B.

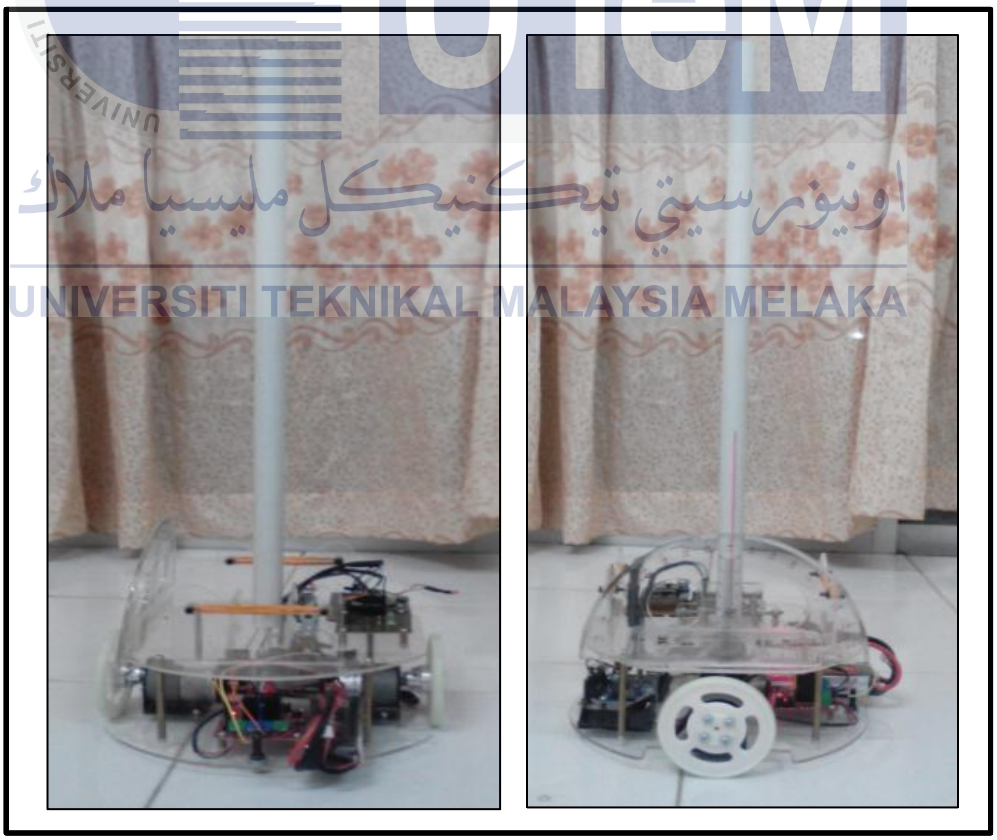


Figure 3.20: Mobile Robot for test performance

1.5.3. Design

Designing of this project is naturally based on basic idea of an Inverted Pendulum system. There would be heavy at the top of long rod of pendulum and some kind of mechanism attached in bottom or base of mobile robot that would be used to balance the weight at the top. Unlike most of Inverted Pendulum experiment that consist of a cart on a tract that limits mobility, but this project was construct a mobile robot that has a greater range of mobility. Figure 3.20 below shows the sketching design of the mobile robot.

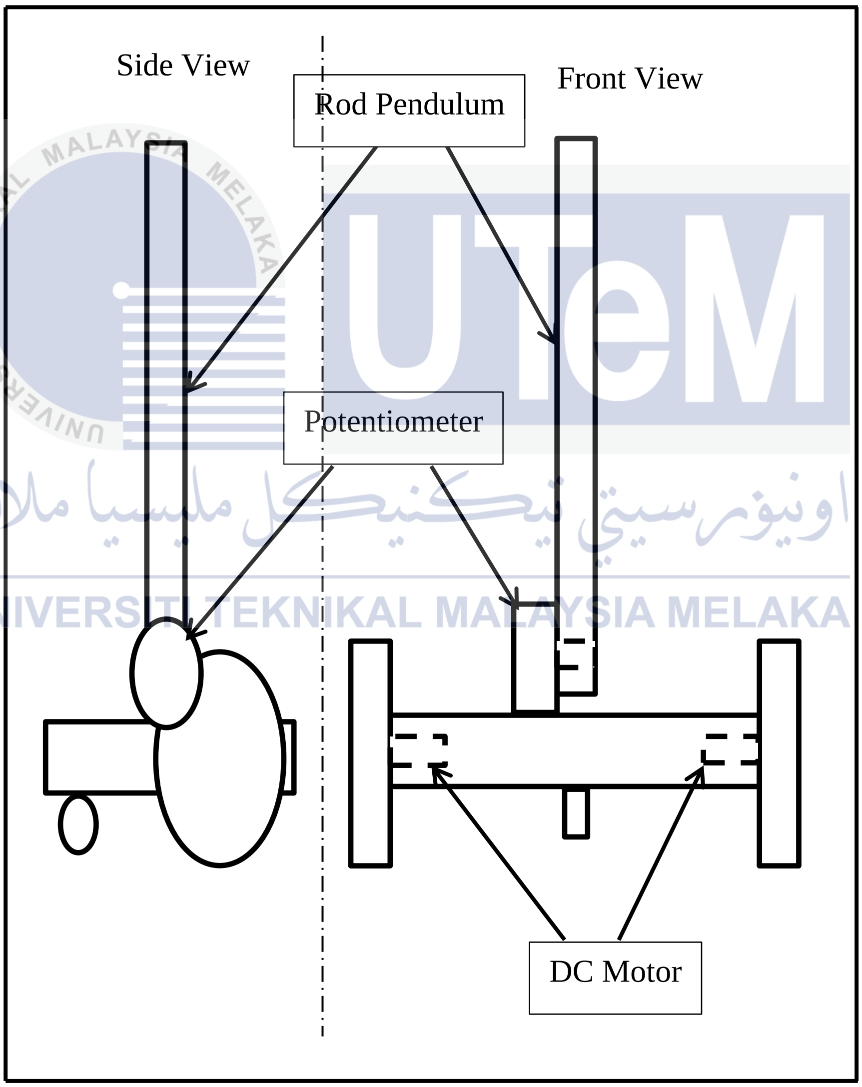


Figure 3.21: Sketching of Mobile Robot

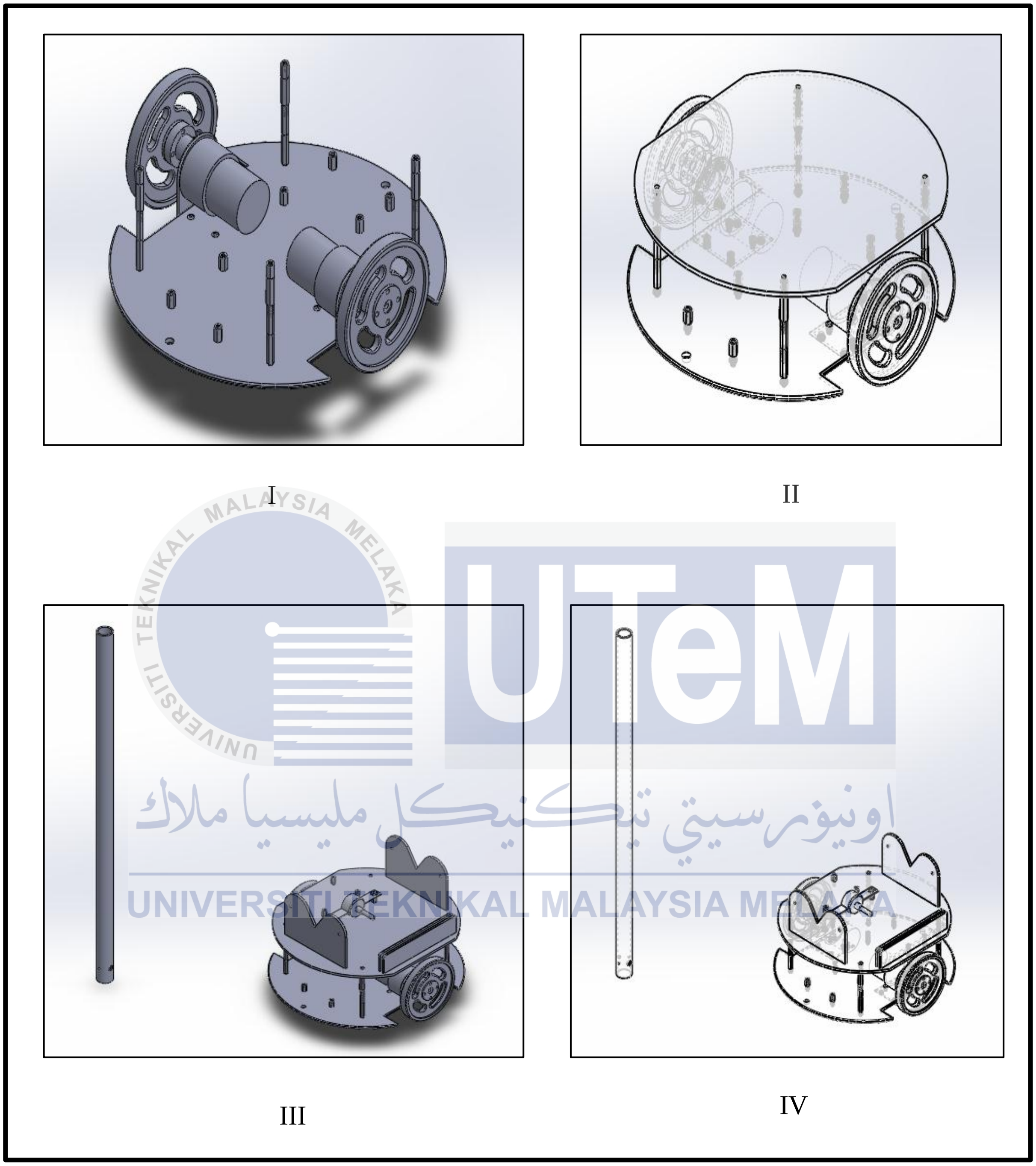


Figure 3.22: Drawing side view for IP Mobile Robot

After finalise do the sketch of drawing, this has been shows in figure 3.22 above where it is part of drawing by using SolidWork software. This copcept of the drawing was selected by in term of stability, safety and light weight for IP mobile robot where it will used for classroom approach. The complete drawing has shows in Figure 3.23 below where mobile robot combines together with the pendulum rod.

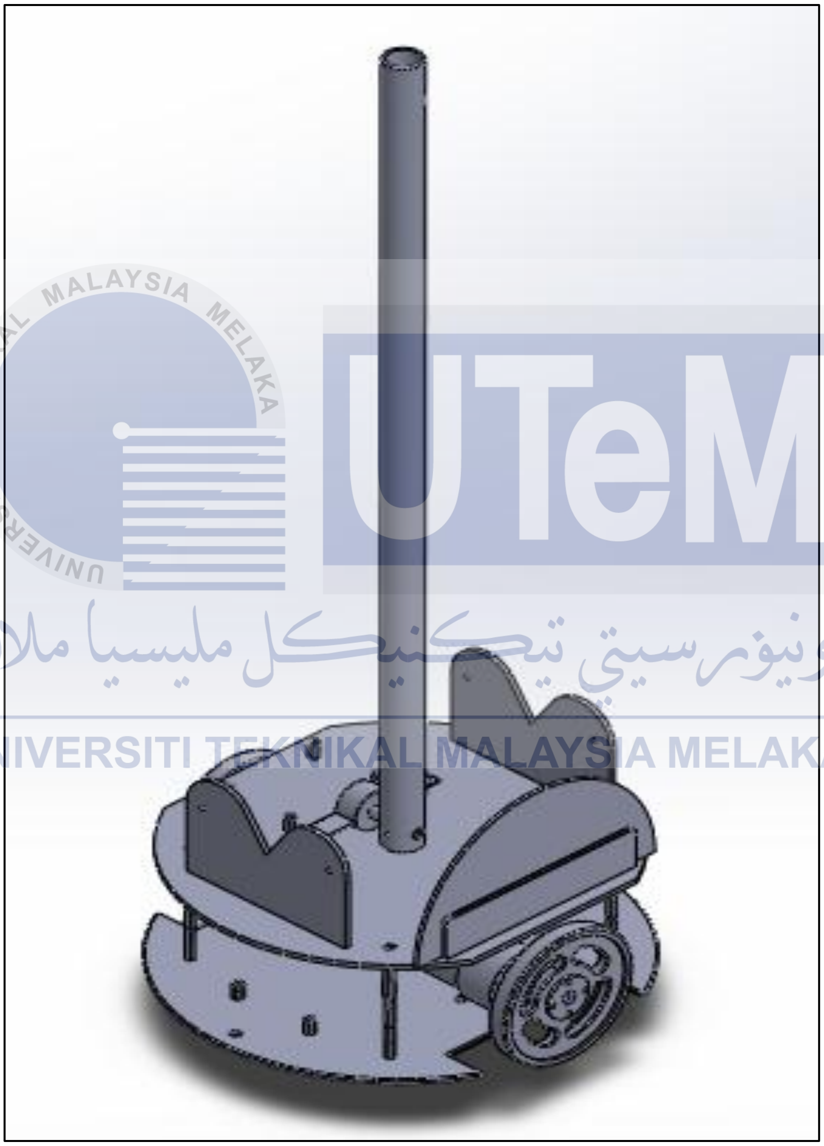


Figure 3.23: Full drawing using software for IP Mobile Robot

1.5.4. Material

1.5.4.1. Equipment

This section was started with prepare some part to do the mechanical part which is hardware for IP mobile robot. This material that been use to complete part of IP mobile robot as shows in Table 3.3 below. The equipment that use to IP mobile robot was has been selected as the appropriateness of the work to be done.

Table 3.3: Equipment for IP Mobile Robot

Part	Apparatus / Tools	Description
1.	Player, knife perspex, ruler and compasses	This tool used to complete base part for IP mobile robot. All of this had been use to design on perspex and get some of curve.
2.	Sand paper and filing	To clean or remove some of fragments on the part that was cut. This tool is required at the perspex and PVC pipe in the form of produce.
3.	Measuring tape, hand saw and marker pen	To measure the length of PVC pipe needed for preparation of the pendulum and marker pen has been use to mark the part that applies. Hand saw are needed to cut the excess part of the pipe.
4.	Knife and rubber for door	This tools and material used to complete part of pendulum where it will insert the potentiometer into the pipe.
5.	Cordless hand drill and drill bit	To do work where it will make a hole for base IP robot and also for pipe that used to make a pendulum.

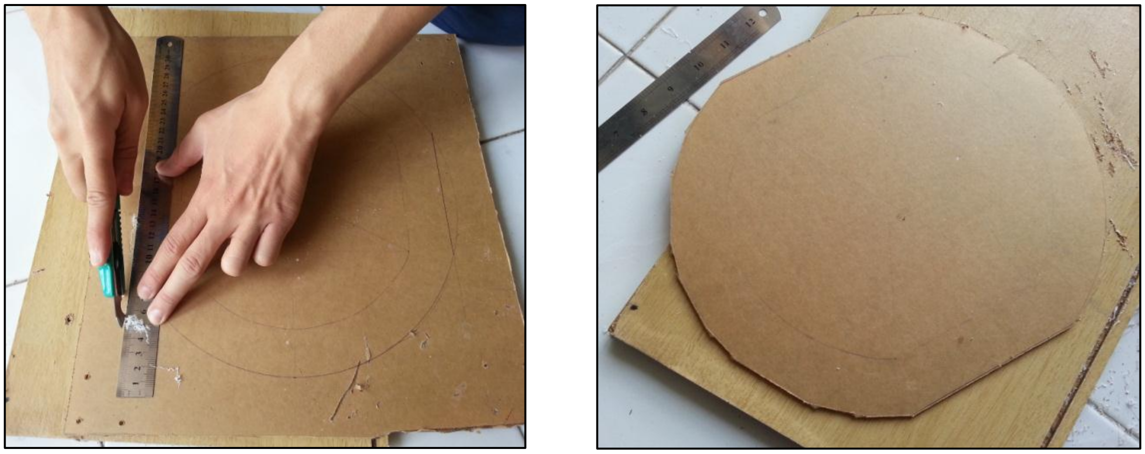
1.5.5. Procedure for hardware part (Mechanical part)

1.5.5.1. Task 1

This section will explain the procedure to complete base part for IP mobile robot. On this section, apparatus or tools in part number 1 and 2 in Table 3.3 that previous topic was used. After all preparation of the tools is complete, the first thing is draw or sketch shape of base /body for IP mobile robot. This as shows in Figure 3.24 and 3.25 below where the shape of IP mobile robot was sketch on perspex before it continues to another session.

Figure 3.24: Tools for sketch and produce base for IP mobile robot

Figure 3.25: Sketching base IP mobile robot on perspex



i ii

Figure 3.26: Shape for base IP mobile robot

After the sketching was complete draw on perspex, second step for this section is by using ruler and also knife perspex to cutting the parts are needed only. This part as shows on Figure 3.26, were the base of IP mobile robot was completed. Lastly on this section, sand paper and also filing was used to remove or clean all fragments after cutting in this part. This is because to make side of base surface smoothly without any accident will happends. This tools and part as shows on Figure 3.27 below.

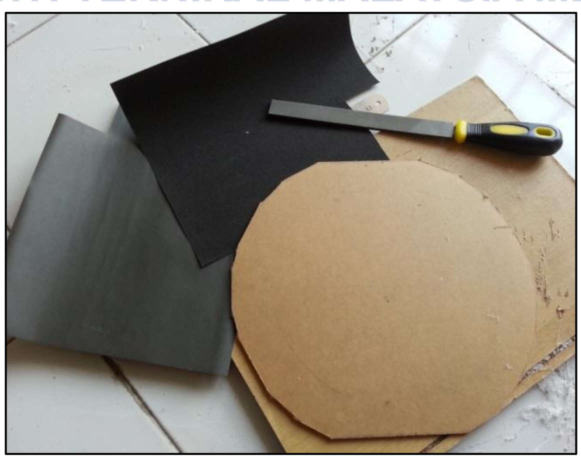


Figure 3.27: Finishing for base IP mobile robot

1.5.5.2. Task 2

On this task will explain about the step to compelet part of pendulum rod. After finish in previous task, to be continuing on this project it is about how to install the pendulum rod at mobile robot without any problems. This section is more focus between others task due it is the main part for stabilization of IP mobile robot. As shown in Figure 3.28 and 3.29, it is about material and also tools had been choose. For this section, apparatus or tools in part number 3 and 4 in Table 3.3 that previous topic was used. Length of the pendulum is 0.5 meter that had mention before this in Table 3.1 about the physical parameter of the system. This is to ensure the parameter in both of software and hardware are similar due after this it will do the analysis both of them.

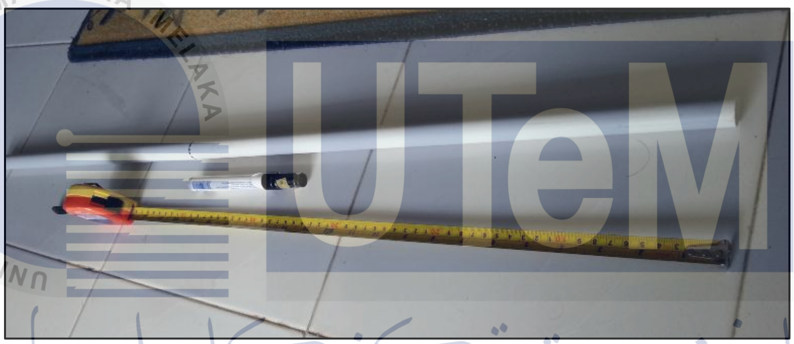


Figure 3.28: Tools for pendulum IP mobile robot

i ii

Figure 3.29: Measure and marking on pendulum rod

i ii

Figure 3.30: Direction for ruber into the pendulum rod

After complete cut the PVC pipe which is rod of pendulum, next procedure is it will combine together with rubber. This material was choosing due to ensure there are nothing happens during the installation for potentiometer. This fact will be cause for loose or instability during the pendulum motion or function. In Figure 3.30 and 3.31, as showns is about the step was do on this part.

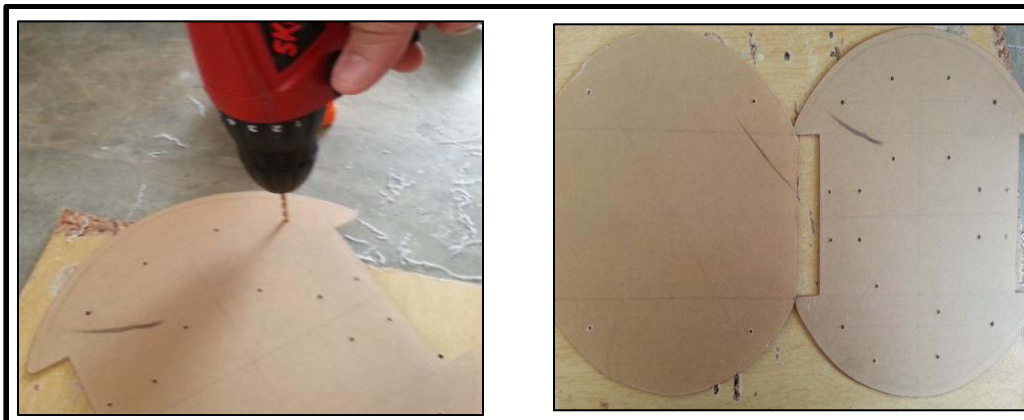
Figure 3.31: Rubber into the pendulum rod

1.5.5.3. Task 3

On this task, it divided by two parts with is for base of IP mobile robot and pendulum rod. After complete for previous tast, this task is to make a hole for some part in IP mobile robot. For this section, apparatus or tools in part number 5 in Table 3.3 that previous topic was used. This hole as used for during installation part and it will make the mobile robot firmer and appear neat. This as shows in Figure 3.32 below for tools where are using on this task.

Figure 3.32: Toos for make a hole of IP mobile robot

In base part, this hole has been make to install some of part such as bracket motor, circuit board (Arduino UNO and L298 driver motor) and also battry. All of the hole in this base part consist 3mm diameter where it will make screw to tighten.This work as shows in Figure 3.33 below.



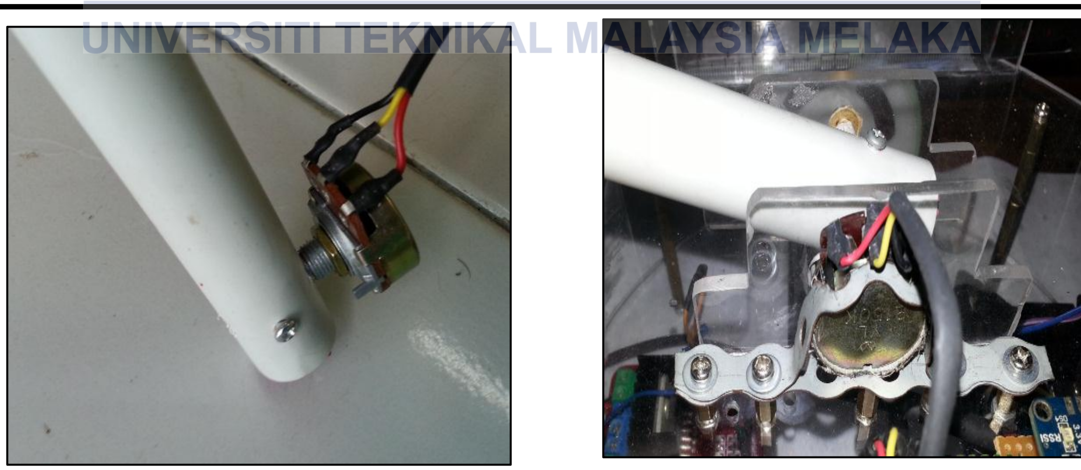
i ii

Figure 3.33: Hole for base IP mobile robot

For pendulum part, the hole also had been making to install potentiometer. In this rod, the hole consist 6.5 mm diameter where it is the diameter for potentiometer. The value of diameter of this part must be fixed due to ensure potentiometer do not have any problem with loose during installation. These as shows in Figure 3.34 below for make a hole on pendulum rod. Last part of this task is installing the potentiometer into the pendulu rod and also on mobile robot. Theses part as shows in Figure 3.35 below.

i ii

Figure 3.34: Hole for pendulum rod



i ii

Figure 3.35: Installation potentiometer into pendulum rod and on robot

1.5.6. Procedure for hardware part (Electrical part)

This section will explain the procedure to complete electrical part for IP mobile robot. On this section, wiring has been done by several boards that use for function this IP mobile robot. The board was selected according to criteria for some of functioning to ensure this IP mobile robot can move smoothly. All about pin that used for each board has mention in Table 3.4 below.

Table 3.4: Connection for each electrical circuit board

No.	Name for circuit board	Pins on board	Arduino UNO pins (5V)	Others source
1.	L298 Motor driver	ENA & ENB	5	
		IN1	10	
		IN2	9	
		IN3	8	
		IN4	7	
		12 V & GND	GND	Battery 12V
		Motor port		Motor DC
2	XBee Series 1	5V	5V	
		GND	GND	
		TX	RX / 0	
		RX	TX / 1	
3	External circuit for potentiometer	5V	5V	
		Signal	A0	
		GND	GND	

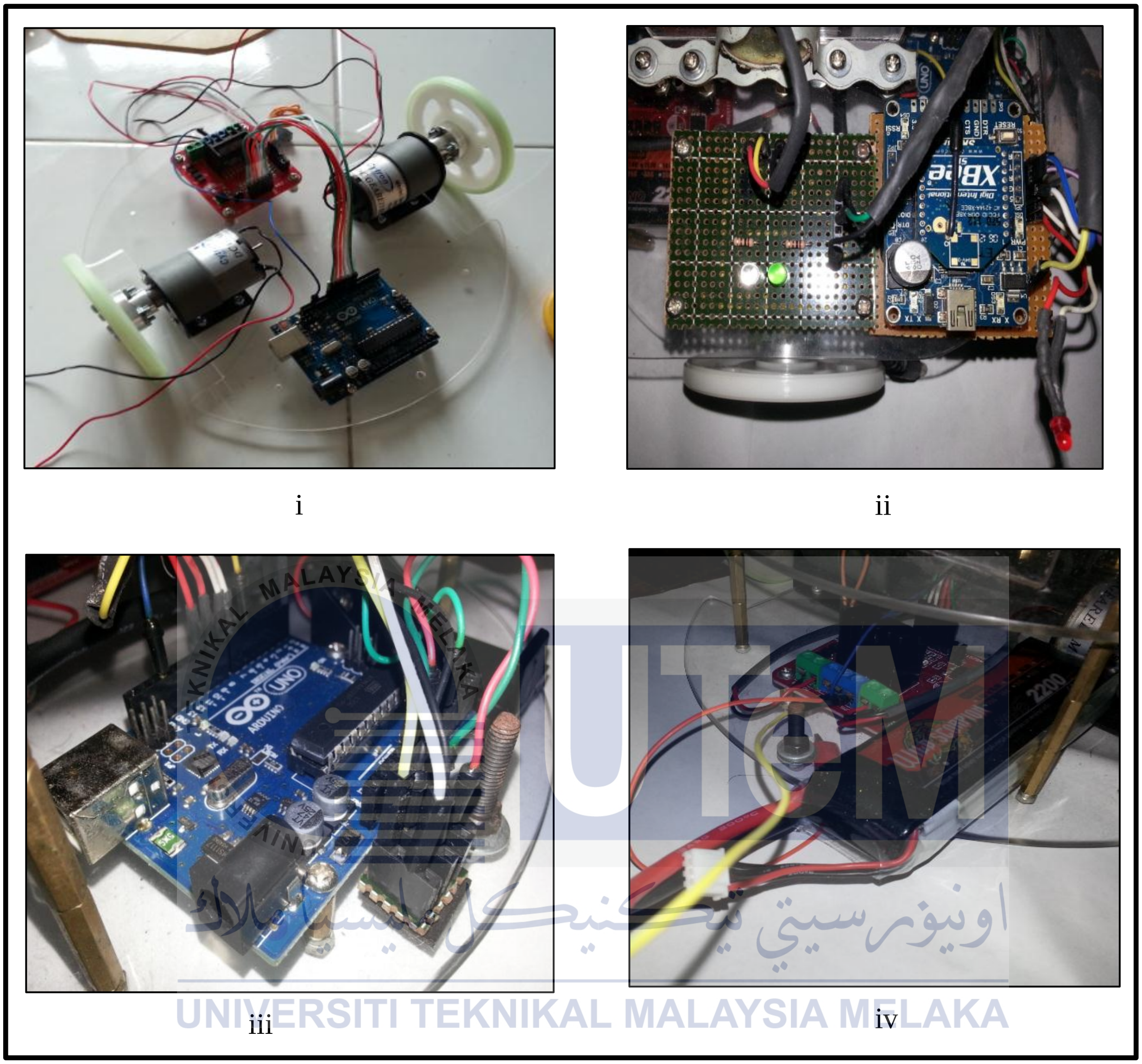


Figure 3.36: Electrical circuit board for IP mobile robot

In Figure 3.36 above as shows the circuit where implement in IP mobile robot. This entire functioning is closely bound to ensure the pendulum robot always in upright position. For Arduino board is as a microcontroller board where each of functioning in this programming robot. This board was connected with motor driver where function as a drive motor direction. These direction of movement for motor where depends on potentiometer rotation where it connected with pendulum rod. The data from movement will get from transmitter Xbee and receive on computer (PC). This situation will continues movement for robot until it is switch off mode. From Figure 3.37 as shows below is about complete installation for IP mobile robot.

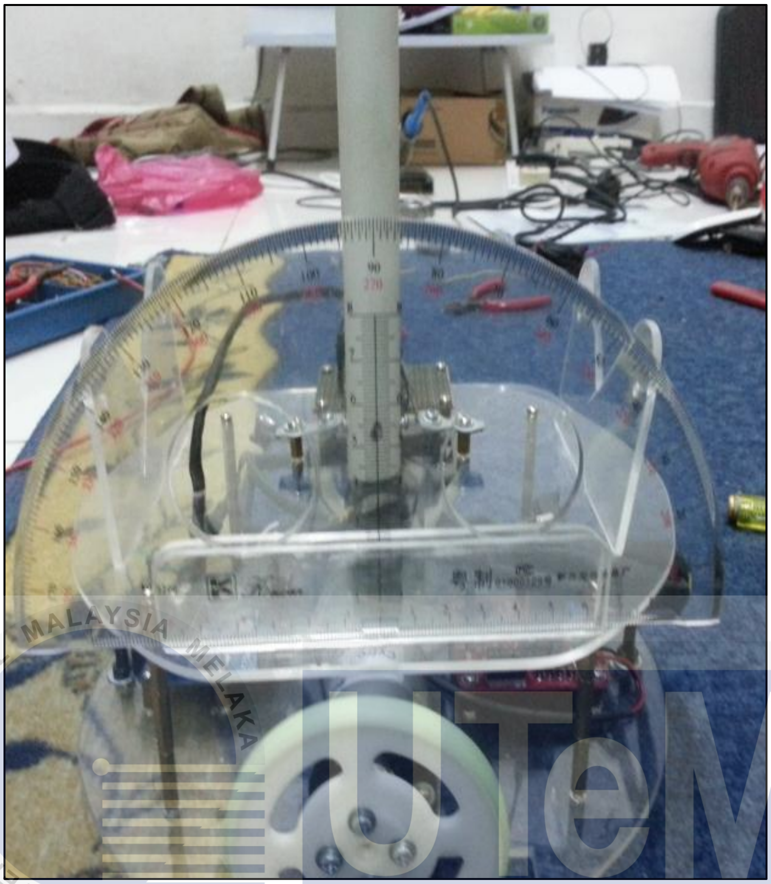


Figure 3.37: Inverted Pendulum mobile robot

1.5.7. Procedure for software part

On this section will discuss about software part in term of programming and also ways to get data from robot movement. This part is the main for the robot system to function the movement and also to ensure pendulum always in stabilize position. Arduino software has been used to write a programming for Arduino UNO board where it is as microcontroller for the IP mobile robot system. This as shows in Figure 3.38 below, where arduino software succesfull verified the programming and it will upload into the Arduino UNO board. In this software also can get the data from robot movement by using serial monitor but it must used wire cable where connect to USB computer (PC). Since IP mobile robot will move quickly and this would be troublesome data taken using the cable wire, so to solve this problem the next action is to use the Xbee wireless data.

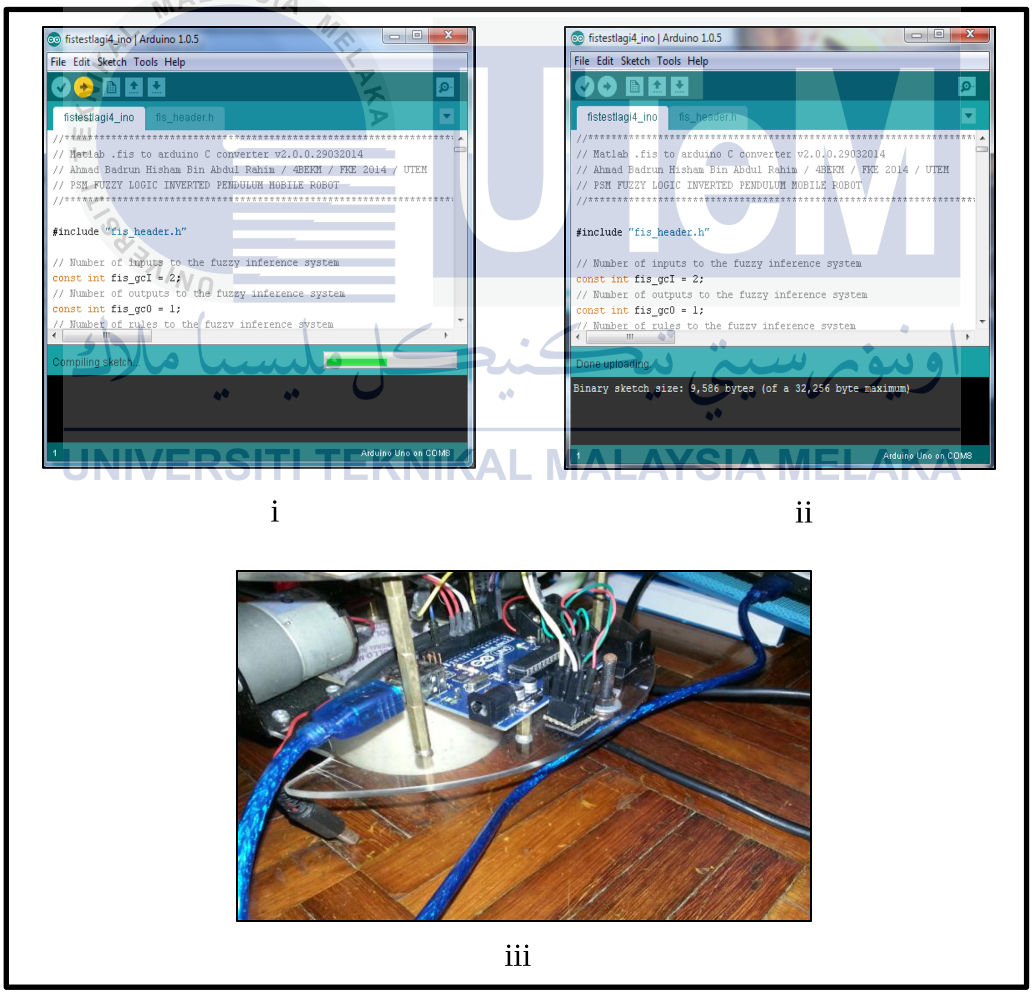


Figure 3.38: Uploaded Arduino programming software into IP mobile robot and connection

For using XBee connection to get the data, X-CTU software was used as an interface connection between XBee boards and USB computer. This software as shows in Figure 3.39 below, where it is shows screen monitor for X-CTU software. XBee connection will read from receiver XBee where it connects with USB computer. Otherwise, trasmmiter XBee will implement on IP mobile robot where it will transfer the data to receiver XBee. The connection for XBee transmitter and also XBee receiver was mention in Figure 3.40 below.

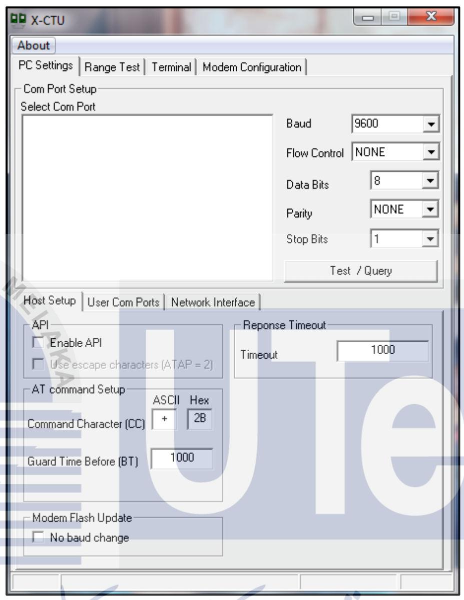
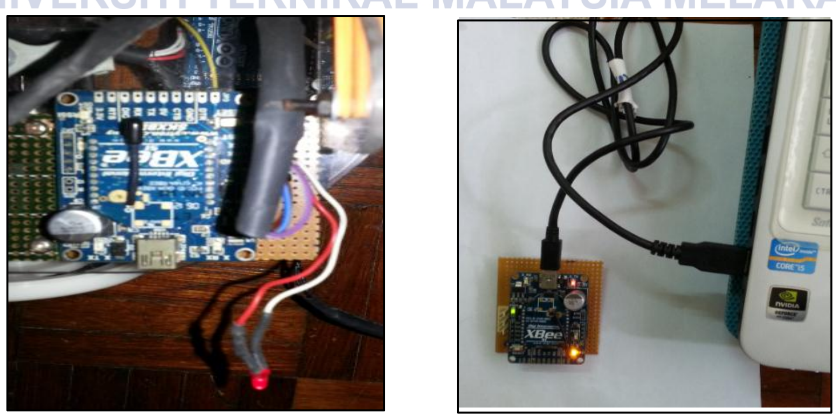


Figure 3.39: X-CTU software screen monitor



i ii

Figure 3.40: Connection for transmitter and receiver XBee board

CHAPTER 4

RESULT AND DISCUSSION

1.1. Introduction

For this chapter may be helpful the reader to understand details of the system based on the result of the Inverted Pendulum (IP) project have been got through some analysis. The first se ction in this chapter will review via Simulink on simulation MATLAB software by obtain the system of transfer function without using fuzzy logic controller. For next section, still using Simulink on simulation MATLAB software but in this part result will focus the system of transfer function by using fuzzy logic controller. For the first result, outcome from the both simulation result will be used to analyze. Second of this result will explain about the performance of hardware functioning and stabilize of the pendulum on upright position. Finally according to the objective of this project, the end of results will analyze the performance and the comparison between software and hardware for the Inverted Pendulum system using a mobile robot.

1.2. Result for software

1.2.1. System of Inverted Pendulum without FLC

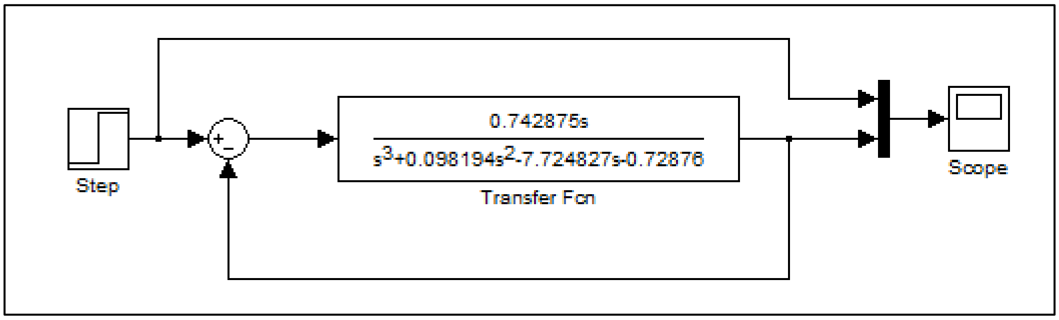


Figure 4.1: Transfer Function Inverted Pendulum without FLC

From Figure 4.1 above shows the transfer function of the system without using fuzzy logic controller. This output of the system has been shows as figure 4.2 and figure 4.3 below where it faced to Inverted Pendulum system.

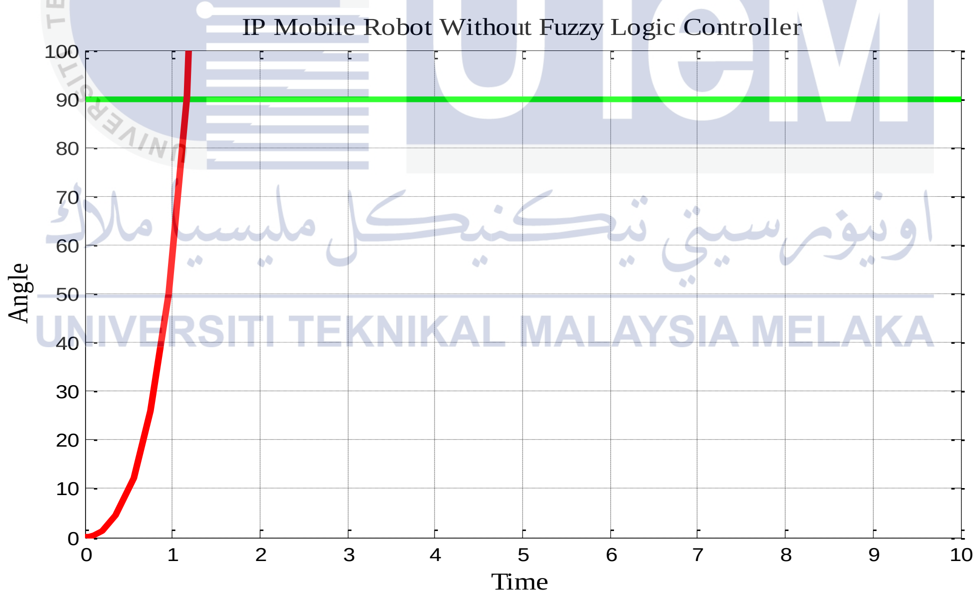


Figure 4.2: Output from scope without FLC

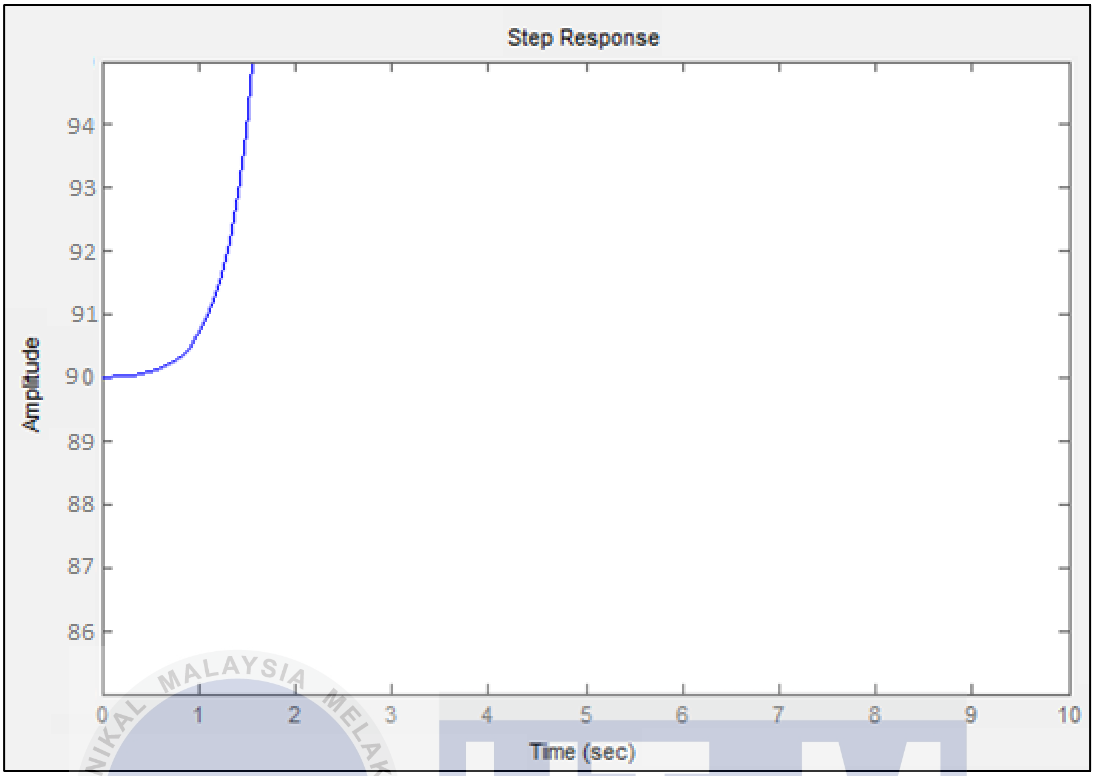


Figure 4.3: Step response of Inverted Pendulum without FLC

The output has been shown in figure 4.2 from using scope on Simulink and the second graph in Figure 4.3 as shown using SISOTOOL where both of is from MATLAB software. As can see, the result from Figure 4.2 shows graph angle with respectively time has two (2) different colors which are green for input and the red color stand for output the system. This result shows the performance of system without using FLC controller. The pendulum for this system has unstable system where the result of angle trough to infinity without back to origin position (input wave) on 90 degrees. This situation might be the pendulum has falling without back to maintain in upright position.

1.2.2. Transfer function of Inverted Pendulum with FLC

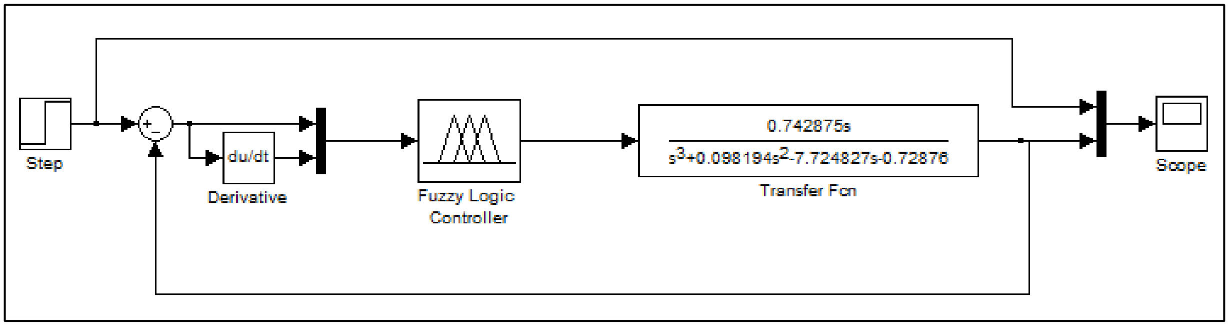
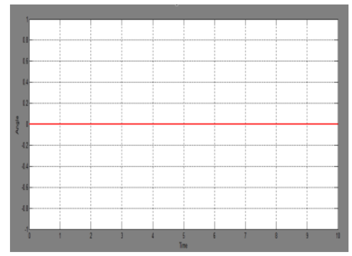
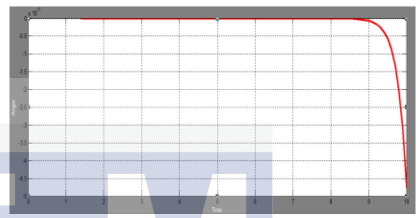
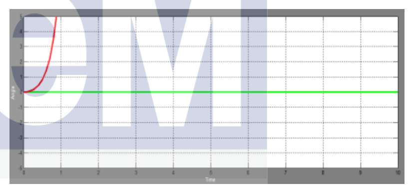
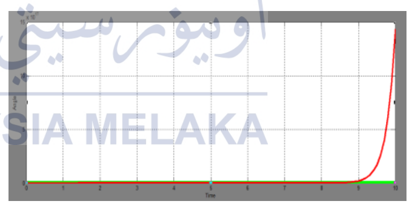
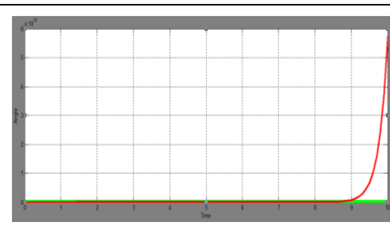


Figure 4.4: Transfer function Inverted Pendulum with FLC

From Figure 4.4 above shows the transfer function of the system using fuzzy logic controller. This system is used derivative for rate of error to get output of the system. There are a few analysis to refine the membership functions to get the best performance of system. These are shows on a table and figure below where the different graph using transfer functions on Figure 4.4 above.

Table 4.1: Tuning for Membership Function FLC system

	INPUT		OUTPUT	DISPLAY (Graph)
	Error	Difference error		
Range Membership Function	30 90	-30 90	-30 30	
	30 90	-30 90	-40 30	
	30 90	-30 90	-30 40	
	30 90	-30 90	0 30	
	30 90	-30 90	30 90	

In table 4.1 above shows after tuning the value on output range (red color) and input range (green color), it will know which the best performance and maintain in stable system. This system used on try and error for get the output result. This might be change the output result graph if the actual testing whereby has some of disturbance on this system. On figure 4.5 below shown the result after finalize changing the output value.

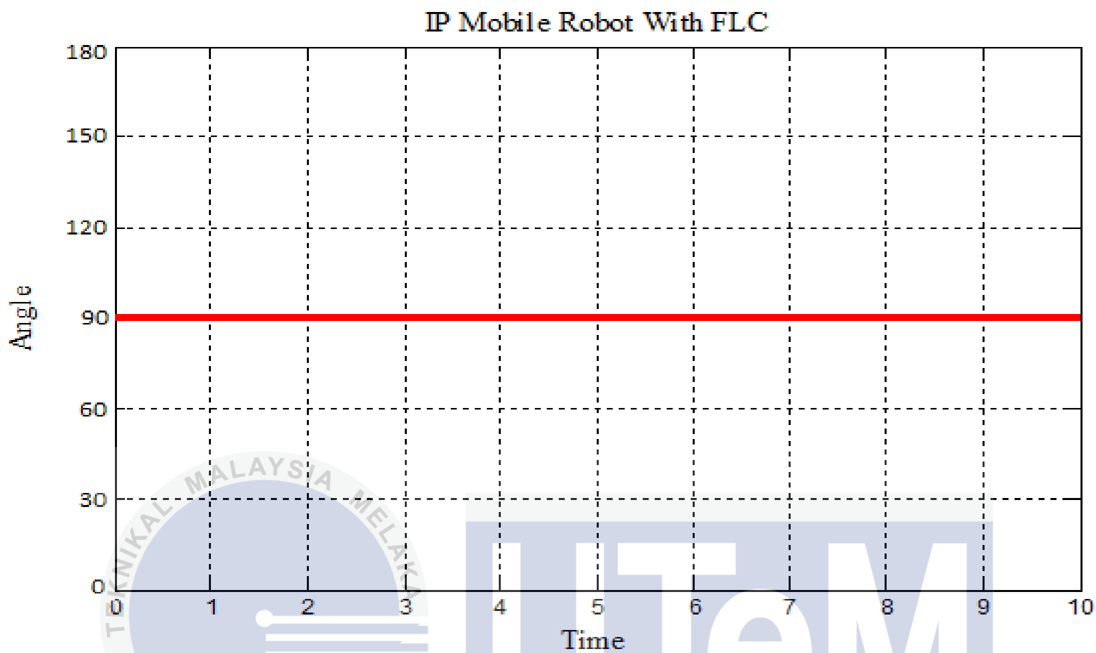


Figure 4.5: Output from scope with FLC

1.2.3. Inverted Pendulum system with FLC noise

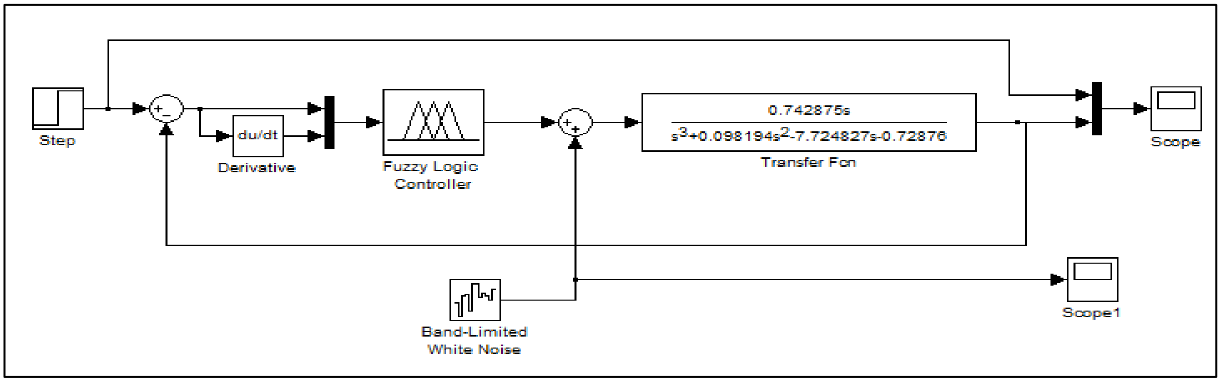


Figure 4.6: Inverted Pendulum system with FLC noise

From figure 4.6 above shows the transfer function of the system using fuzzy logic controller with has noise. This system is used to show the performance of the FLC if the system has noise or disturbance situation. The result for this testing by using simulation to shows the wave of pendulum in a real system if have any noise. Output wave (red color) started on 90 degrees and it change at 0.2s where the system have overshoot because of noise. This system try to maintain back the position on 2.1s where the output (red color) nearly to input wave (green color) to make the pendulum in 90 degrees or in upright position but just a few second then the system back a small overshoot.

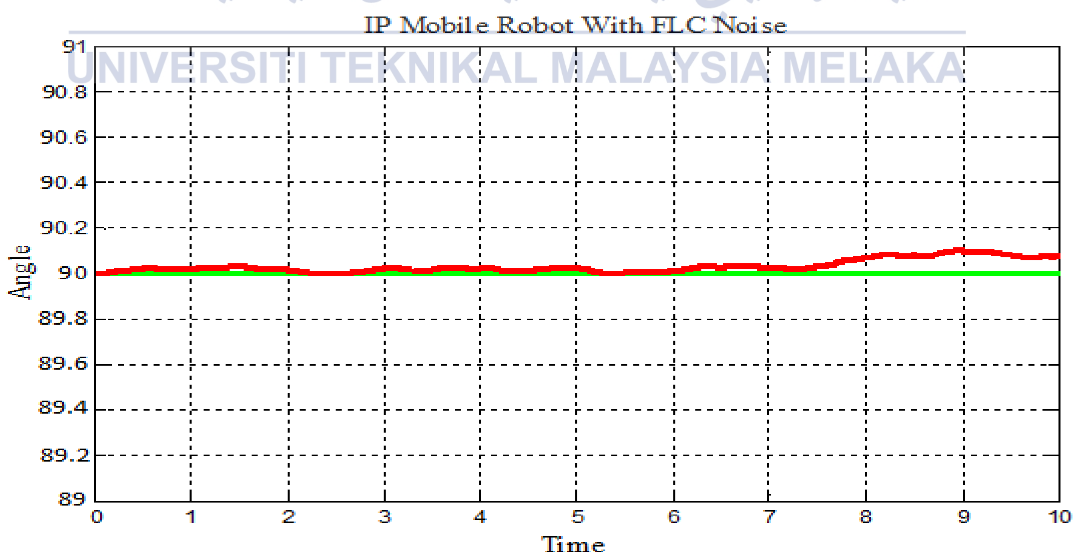


Figure 4.7: Output from scope system FLC with noise

1.3. Result for hardware

1.3.1. Test the performace

The first step to test the performance stability of the pendulum is located on the mobile robot to check for each angle using a protractor. This calibration was carried out to assess the performance of each angle speed where it imposed by the pendulum. The calibration has made in this test due to ensure fast-moving motor speed or rotation alongside according to the different values. This as shows in Table 4.2 below where the analysis for a different angle from the vertical angle which is upright position of pendulum until before it falling down. The detail figure for this calibration was provided by referring to the Appendix C.

Table 4.2: Calibrate speed of motor for each angle pendulum

No.	Angle	Speed(PWM)
1.	90^0	0
2.	85^0	112.04
3.	80^0	117.46
4.	75^0	121.25
5.	70^0	123.67
6.	65^0	124.89
7.	60^0	125

The Figure 4.8 below as shows, the graph angle versus speed for this analysis of this calibration. The speed control is generated from the implementation of fuzzy logic controller in membership function where it will control and also to enable the pendulum in stable condition. In Figure 4.9 below shows the result of speed and angle by using serial print command on serial monitor Arduino software.

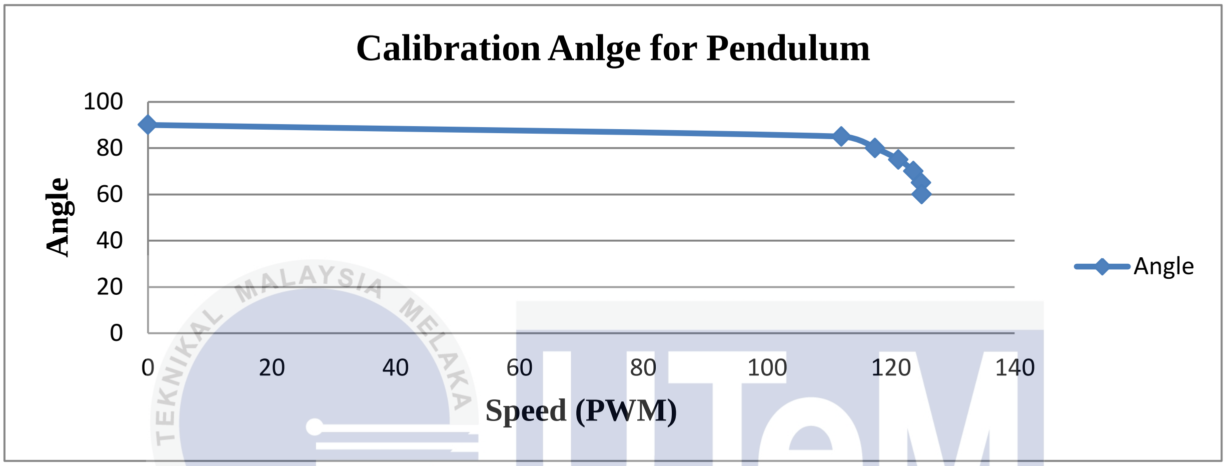


Figure 4.8: Graph angle versus speed (PWM)

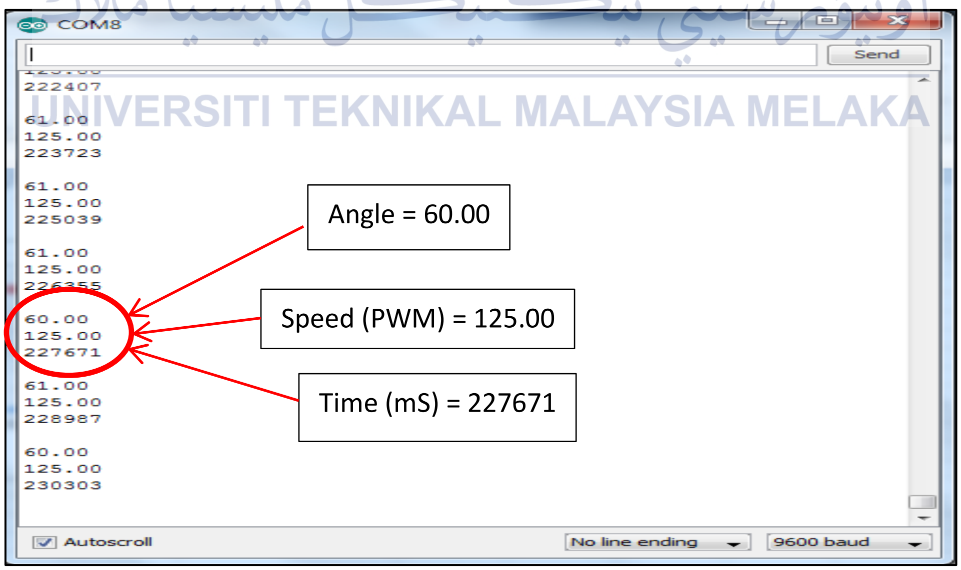


Figure 4.9: Serial monitor from Arduino software

1.4.Analysis performance

1.4.1. Software part

This performance of the system has shown on Table 4.3 below for different type been use in IP system. The result of this performance shows without FLC in IP system will be unstable system which is all of this infinity performance. However, for with FLC the system has good enough performance where all the criteria of index show in stable system. This is might be the pendulum is maintained in upright position in a real situation. For FLC with noise, the result shows smallest in overshoot and settling time which is the system have disturbance in a real situation and the pendulum will be try to maintain in upright position even it changing on angle position.

Table 4.3: Performance Indices for software IP systems

System Type	Performance Index			
	Time Peak, TP (sec)	Settling Time, Ts (Sec)	Steady State Error, Ess (Sec)	Overshoot, Os (%)
Without FLC	∞	∞	∞	∞
With FLC	0	0	0	0
FLC with noise	0.04	2.2	0.35	0

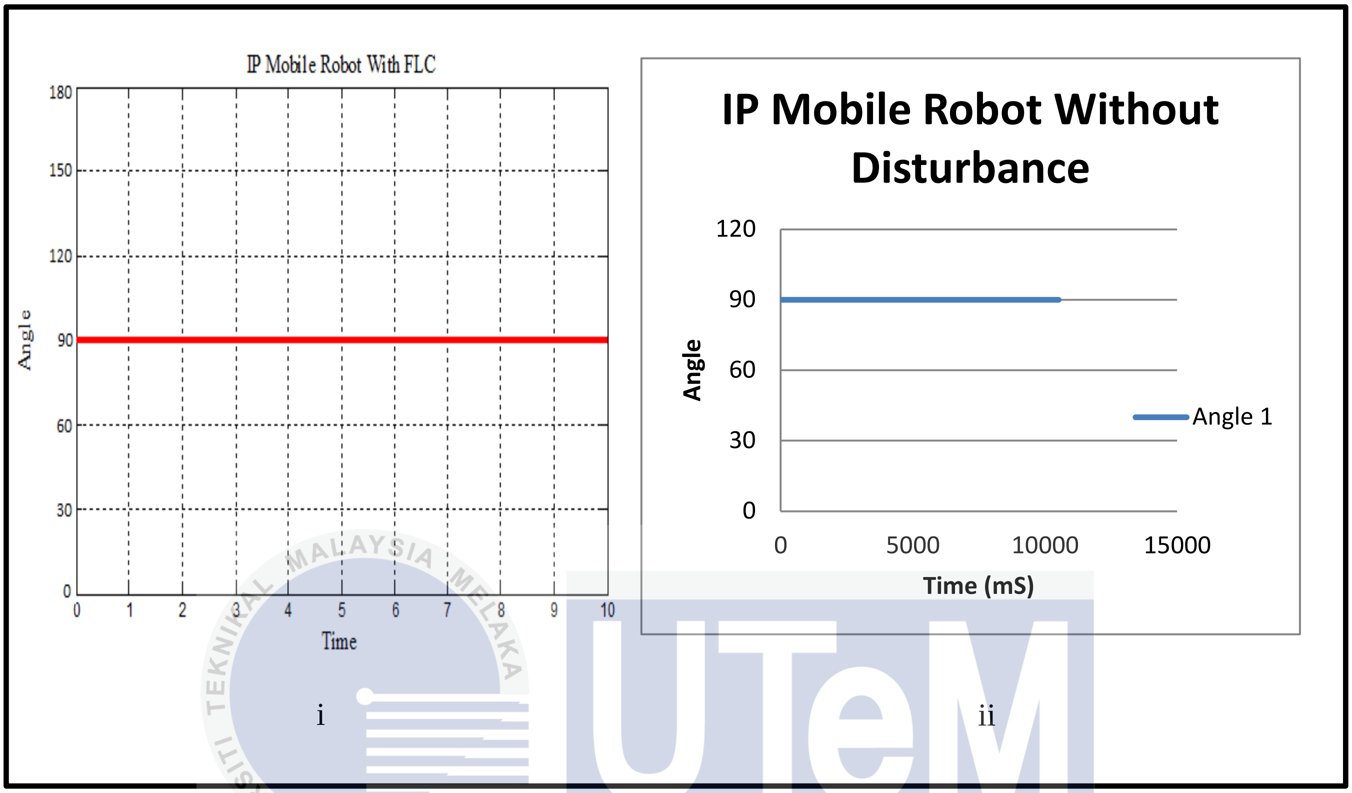
1.4.2. Hardware part

This performance for the hardware part has shown on Table 4.4 below to test the ability and efficiency of the inverted pendulum using the cart or mobile robot as a tester. Analysis was performed on the surface of the tile floor as test the efficiency and performance of the motor speed to stabilize the pendulum. Revenue from experiments that have been done on the mobile robot, the pendulum will always be in an upright position without any interruption and the speed of the motor in stationary. However, if some disturbance imposed on the pendulum in the upright position, the speed has begun to change according to the setting made on fuzzy logic controller. The data were taken from these experiments are shown in Table 4.4 below.

Table 4.4: Pendulum imposed disturbance

No.	Time (mS)	Angle	Speed (PWM)
1.	0	89^0	105.75
2.	1314	85^0	107.5
3.	2629	91^0	103.35
4.	3944	88^0	107.5
5.	5259	90^0	0
6.	6574	90^0	0
7.	7888	90^0	0
8.	9103	90^0	0
9	10518	90^0	0

1.4.3. Comparision between hardware and software



Fiigure 4.10: Comparision graph between simulation and real time without disturbance

After analyzing the results of previous experiments that have been performed on sotware and also in hardware, further analysis by performing experiments through the comparison between the software and the hardware. As can be seen in the Figure 4.10 above shows (i) is used for simulation MATLAB Simulink and (ii) from real time of Inverted Pendulum graph without interruption. Both of the graphs produced were show straight line at 90 degree angle position. The results of these studies show that the performance of fuzzy logic control implemented by the inverted pendulum on a mobile robot is proved interms of efficiency and accurancy by using this controller without being charged any disturbance.

Then proceeded to analyze experimental disturbance that imposed on the pendulum rod is in a stable position. From Figure 4.12 as shows below results that has made by using simulation Simulink Matlab software with take data readings by 10 seconds. In this software results by including noise as shows that the output wave (red color) started on 90 degrees and it change the position where the system have overshoot because of noise. This system try to maintain back the position on 7s where the output (red color) nearly to input wave (green color) to make the pendulum in 90 degrees or in upright position but just a few second then the system returned in conditions irregularly.

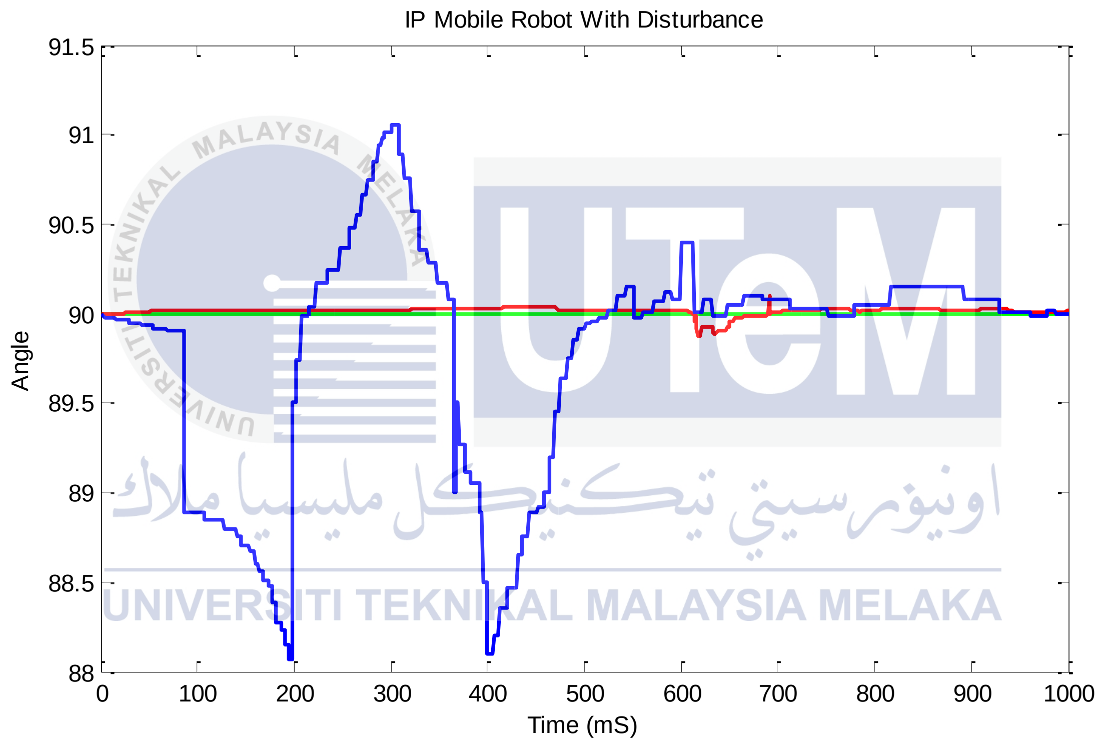


Figure 4.12: Comparision graph between simulation and real time

Comparative seen through experimental results performed on the hardware (blue color) where the resulting graphs also shown in the Figure 4.12 above. The result of this studies that have taken form analysis hardware data on Table 4.4 in previous topic by implement disturbance when the pendulum in upright position. In this graph results as shows that the output wave from hardware (blue color) started on 89 degrees and for the stabilization of the pendulum wave which is on 90 degrees (green color). Analysis is performed by observing the ability of the pendulum return to the vertical position. Result of the study found that the pendulum will be able to approach the wave or line angle of 90 degrees to maintain stability. However, the pendulum returns to the original position in term of the stable at the time of 7 seconds and this would remaind until there are other distractions that apply.

For comparision between the two parts, the analysis found that there were some errors in hardware compared to the software. This probable from the type of interference imposed on the hardware due is has a strong push on the results force exerted by the pendulum compared to the software that no force was applied but only using the noise toolbox.

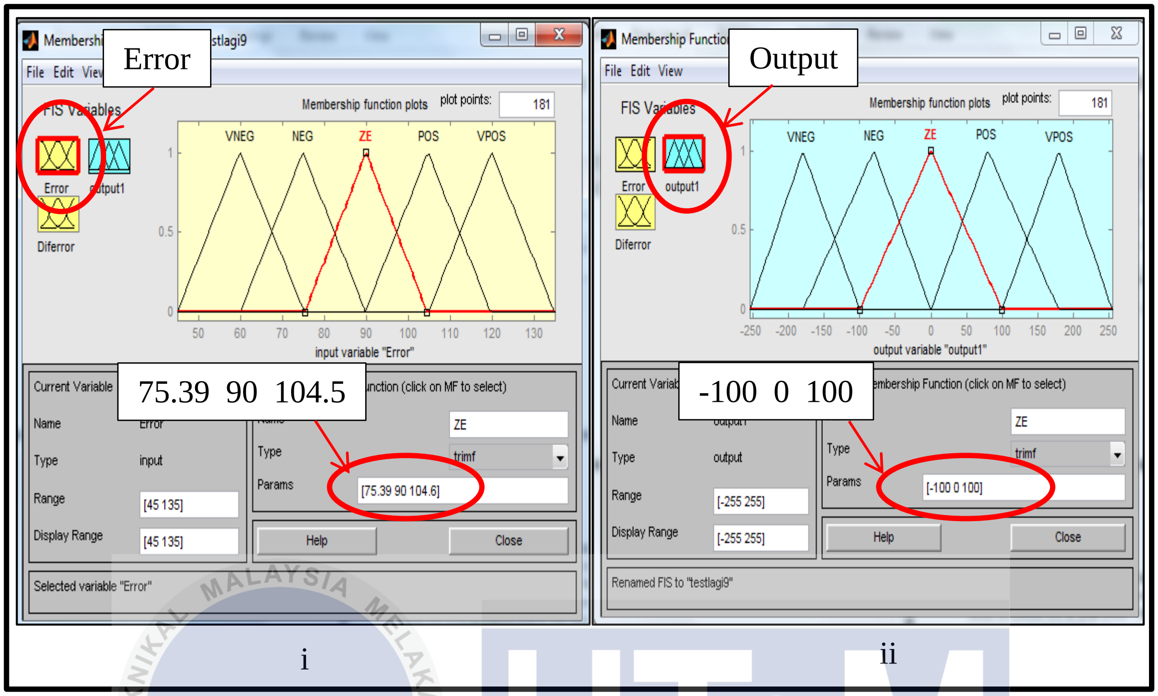


Figure 4.13: Membership function for IP mobile robot FLC

After analyzing experiments on the system inputs, the final analysis was done to ensure the stability of the pendulum experiment with more accurancy is reviewing the situation on the output side. Figure 4.13 above shows the membership function for fuzzy logic control settings for the inverted pendulum system was added. Based on the diagram there are two parts which input or named as an error and the output side.Within the input diagram show the value of 75.39, 90 and 104.5 which represent each angle and for -100,0 and 100 are shown value of ouput. This shows that the current situation is between angle 90 degrees, then the output is 0 that means the speed of the motor is stationary.

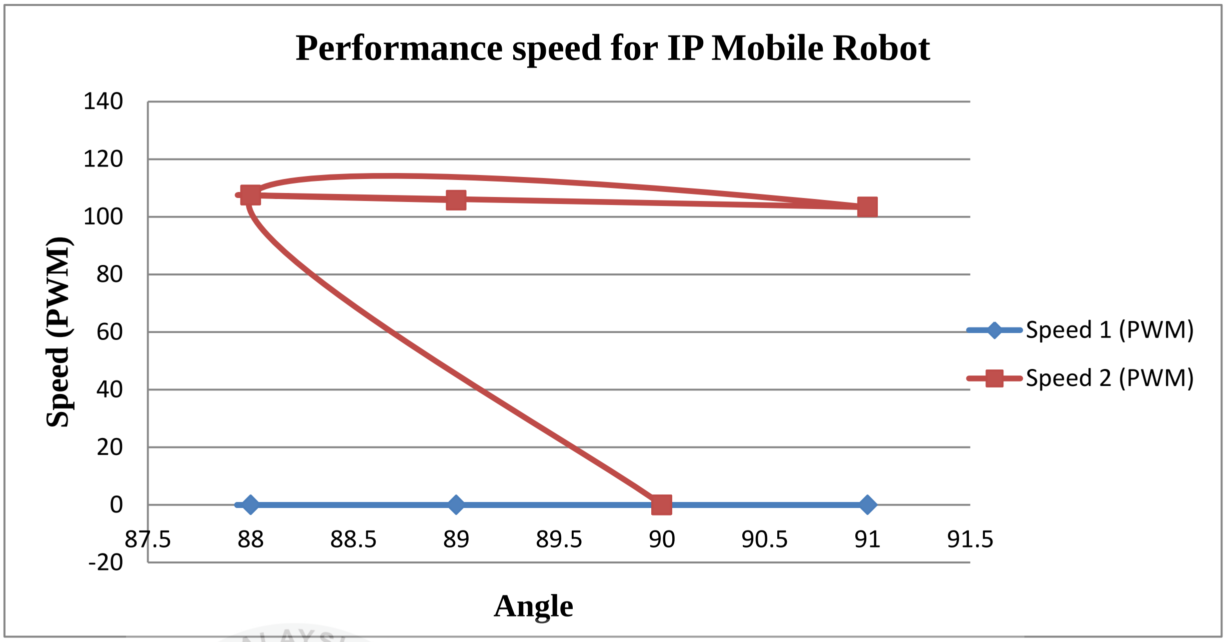


Figure 4.14: Graph for speed versus angle IP mobile robot

Comparisons are performed on Figure 4.13 can be seen through the results of experiments carried out on the hardware where the resulting graph is shown in the Figure 4.14 above and 4.15 below. The result of this studies that have taken form analysis hardware data on Table 4.4 in previous topic by implement disturbance when the pendulum in upright position. In this graph results as shows that the output wave (red color) started on 89 degrees which is in 105.75 speed in PWM and for the stabilization of the pendulum wave which is on 90 degrees (blue color) where motor in stationary. Analysis is performed by examining each input and output range of the software and compared with the range of values produced by the hardware value according to the specification provided. Result of the study found that there is little error where in around 5 percent (5%) through the output of the motor speed (PWM) when compared with the membership function which has been settings. This probable the result might be from the environmental conditions that affect the mobile robot and also when it was moving the pendulum where will impact on performance of speed motor while it is on running. In Figure 4.15 below as show the speed of motor for IP mobile robot in the real time. In this graph results as shows that the output wave (red color) started on 105.75 speeds in PWM and for the stabilization of the pendulum wave where motor in stationary (blue color).

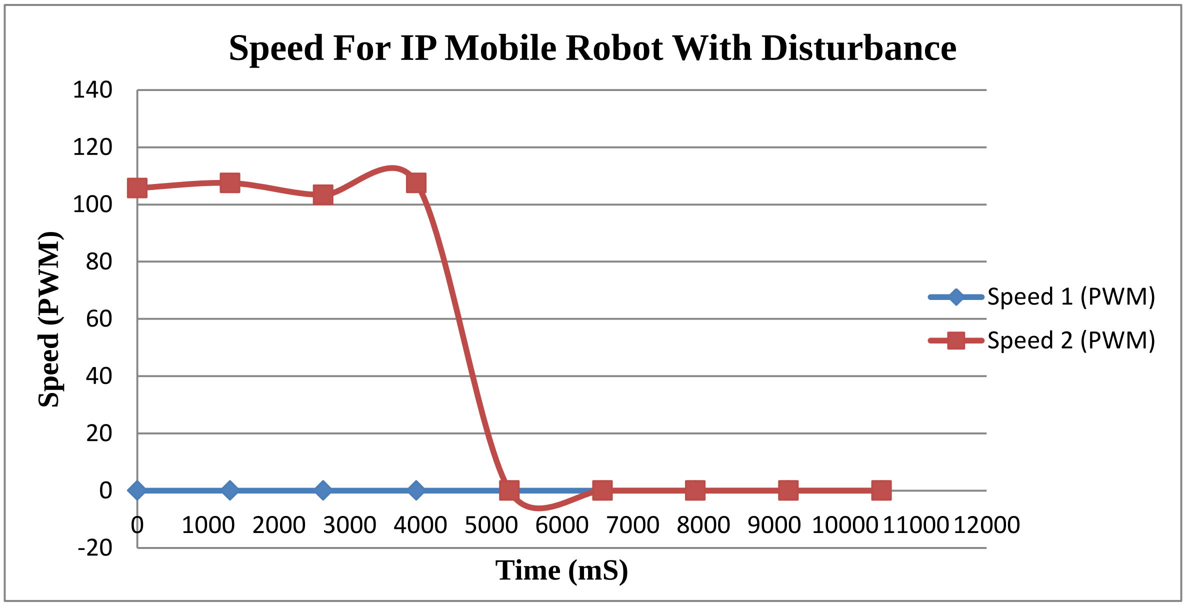


Figure 4.15: Graph for speed versus real time IP mobile robot

1.5. Discussion

Based on the first result, the simulation system without using fuzzy logic control as shows the graph deviated in towards infinity. This is because the Inverted Pendulum system cannot stabilize the pendulum from the upright position. Without any control, inverted pendulum cannot know the circumstances in which the maintain condition in upward. However, the result from second part whereby using the fuzzy logic controller much better than first part. This result shows that the pendulum can be stabilize in upright position where it totally improve the overshoot and also improve on response time of the system. On FLC with noise the result also shows the performance of pendulum system whereas it has any disturbance in a real situation. From the analysis, the real situation show Inverted Pendulum system using FLC as the system controller is able to stabilize the pendulum from falling.

CHAPTER 5

CONCLUSION & RECOMMENDATION

5.1. Conclusion

In this chapter is the last part in report which will explain the findings and to concluded through the overall performance of the project. According to the first objective that had been state before, in this project which develop and design a control system for the Inverted Pendulum has been produced by Simulink in MATLAB simulation software. The results of the both of studies performed was appears that fuzzy logic control more accurate and efficient during many times for testing according to response of pendulum system. This is viewed in the form of the graph which the system is able to stabilize the pendulum in a real situation. Other than that, on software situation also can improve overshoot and response time. Otherwise, for condition in real time the system show the speed of motor via PWM will control the pendulum from falling.

5.2. Recommendation

Based on result in simulation, fuzzy logic control has demonstrated in Simulink that this controller has advantage in term of accuracy to control system. However, some of work can be done in the future to improve the controller such as by using robust control where it make a robust for this system. For the selection of a suitable device, using a potentiometer as the change in the angle of the pendulum is a simple way to control the stability and easy to programming but it is difficult to get the accurancy value for change angle. However, in future work to improve the performace in stabilization of the pendulum by using the IMU or encoder as a sensor to ensure the pendulum always maintain upright position.

REFFERENCE

[1] Sandeep Kr. Tripathi, Himanshu, Prerna Gaur, "Robust Control of Inverted Pendulum using Fuzzy Logic Controller," *IEEE paper,* 2013

[2] Yanmei Liu, Zhen Chen, Dingyu Xue, Xinhe Xu, "Real-Time Controlling of Inverted Pendulum by Fuzzy Logic." *IEEE/ION International Conference on Automation and Logistics*, pp. 1180–1183, August 2009.

[3] Md. Akhtaruzzaman, A.A. Shafie, "Modeling and Control of a Rotary Inverted Pendulum Using Various Methods, Comparative Asessment and Result Analysis," 2010 *IEEE/ION International Conference on Mechatronics and Automation*, pp. 1342–1347, August 2010.

[4] Arpit Jain, Deep Tayal, Neha Sehgal, "Control of Non-Liner Inverted Pendulum using Fuzzy Logic Controller," *IEEE/ION International Journal of Computer Application*, pp. 7–11, May 2013.

[5] `M.I.H. Nour, J. Ooi, K.Y. Chan, "Fuzzy Logic Control vs. Conventional PID Control of an Inverted Pendulum Robot," *IEEE/ION International Conference on Intelligent and Advance System*, pp. 209–214. 2007

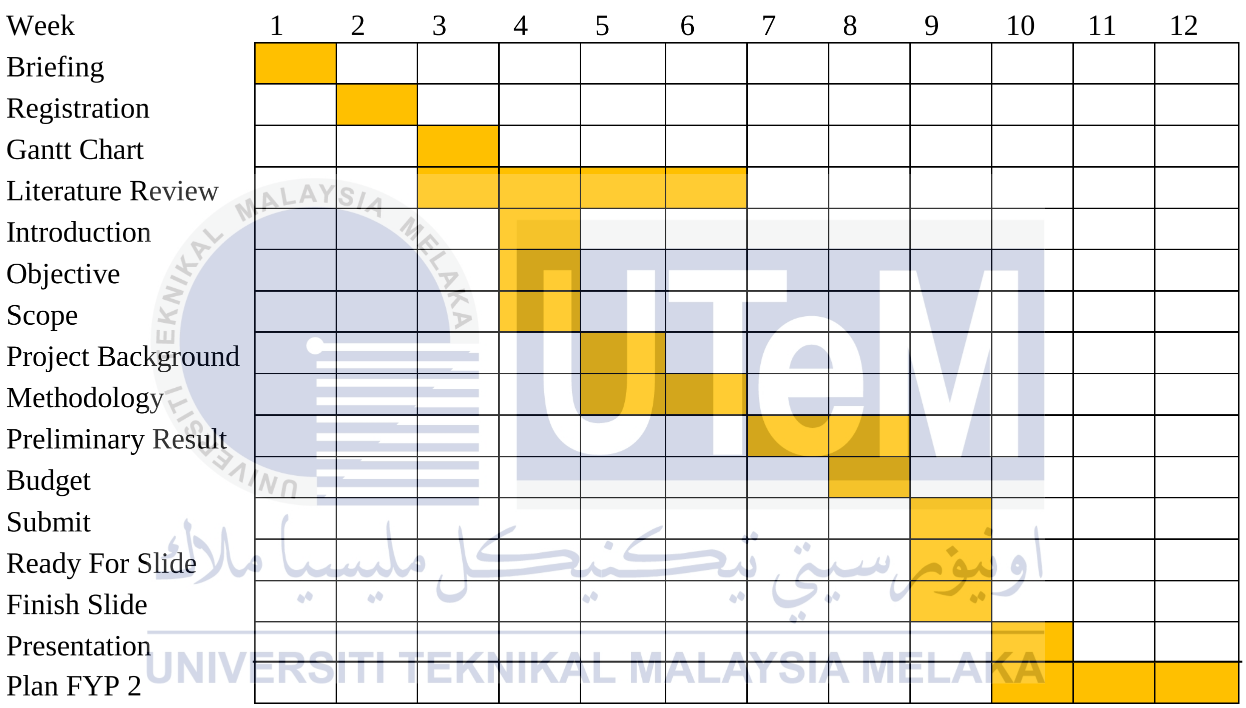
[6] T Rakesh Krishnan, "On Stabilization of Cart-Inverted Pendulum System: An Experimental Study," *M.S Thesis of Technology by Research in Electrical Engineering, National Institute of Technology*, India, July 2012

[7] Hung T.Nguyen, Nadipuram R.Prasad, Carol L.Walker, Elbert A.Walker, "*A First Cource in Fuzzy and Neural Control",*4th ed., Washington D.C.: Chapman& Hall/CRC,2003

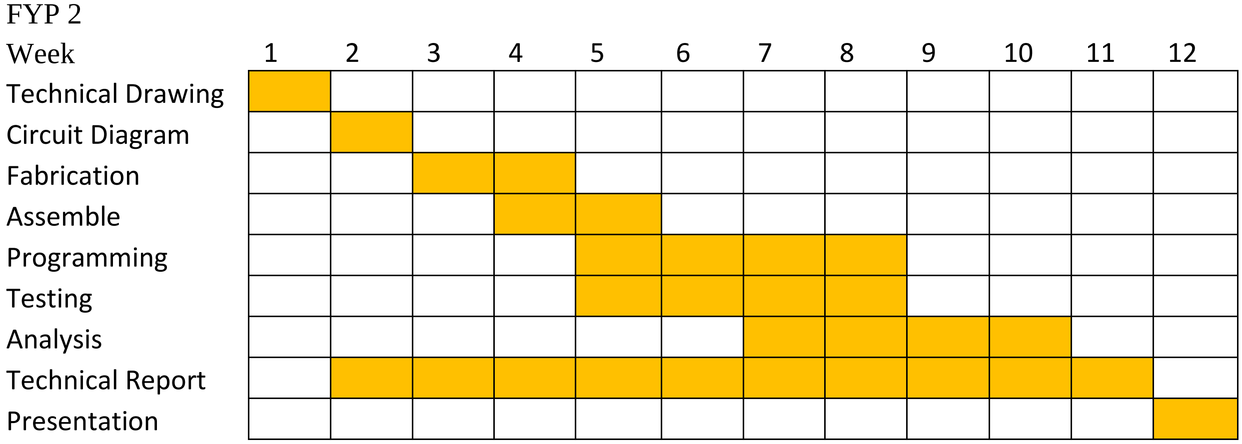
APPENDIX A

Gantt chart

FYP 1



FYP 2



APPENDIX B

Procedure

1. Mobile robot system will be functioning after push button ON in Arduino Uno board. This mode is to make the robot in standby mode before it start running the programming.
2. Pendulum is stand on vertical position and hold it before pressing the start button to keep the pendulum in the origin or initial position in order to maintain the stability. This will be testing the performance of pendulum when it starts function the cart motion.
3. When the step 2 are archive which is to make the pendulum maintain on upright position without holding anything, the analysis of the performance this system can get with use little noise or disturbance on the pendulum. This concept as shown in figure 3.19 at hardware part section experimental setup by changing the degrees of angle pendulum. The protractor is implemented on nearly to pendulum rod to facilitate data taken.
4. Data will be taken at the specific degree for the both side. Whereas over than the degree, will be unstable system because it is out of range of membership function FLC set. The table below data will take for each degree.

Table: Testing the performance of Pendulum

Disturbance (θ)	Performance	Speed of Cart
80^0 & 100^0	✓	Slow
75^0 & 104^0	✓	Medium
45^0 & 135^0	✓	Fast
Below 40^0	X	Fast
Above 140^0	X	Fast

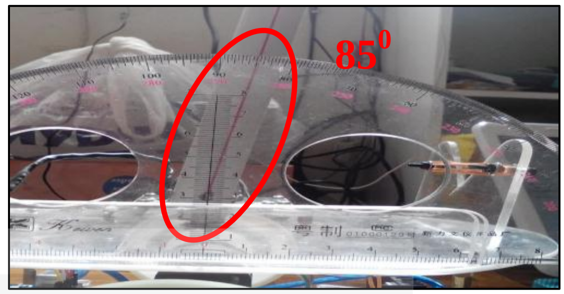
Legend:

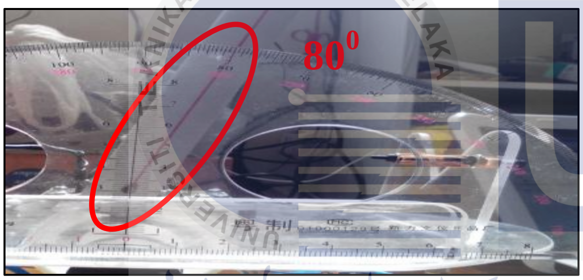
✓ = Acceptable
X = Out of range

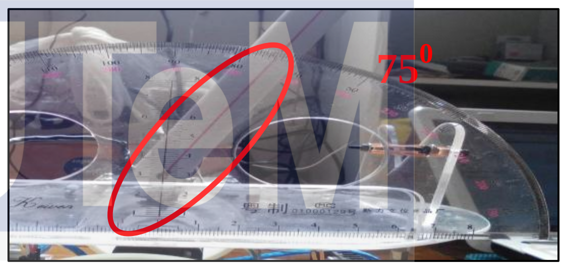
5. The step 4 will repeat for 10 times where equal to 1000 data to take repeatability performance of mobile robot for IP system and also this will show the reliability of the system.
6. Then the result from step 5 will be implementing on graph to show the performance of the mobile robot.
7. Next step, data from motor will be taken to computer by using Arduino Uno as an interface between motor and computer. This data will compare with simulation in Simulink graph which is to analyze the proven data from software in a real situation.

APPENDIX C

Check performance for hardware

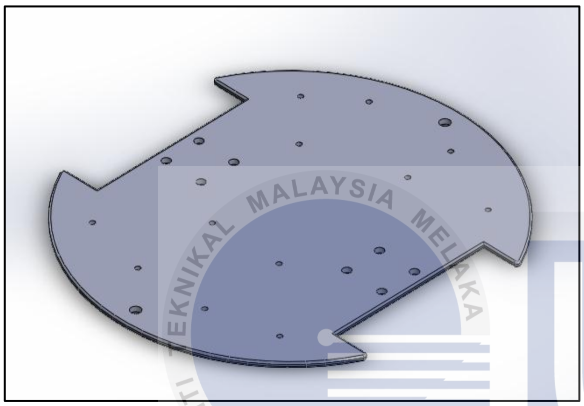


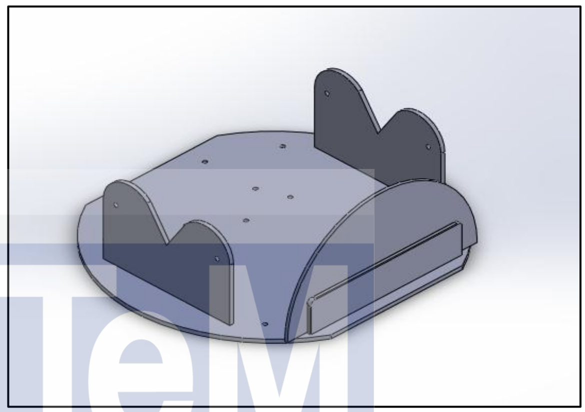


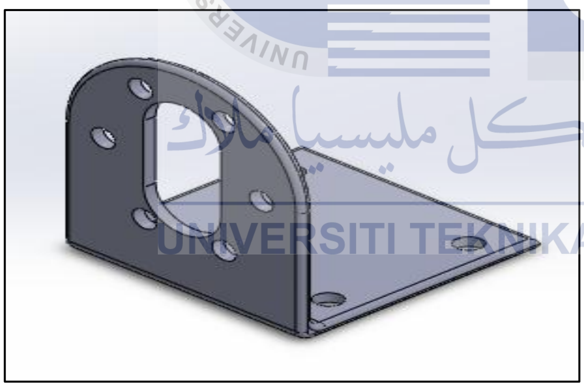


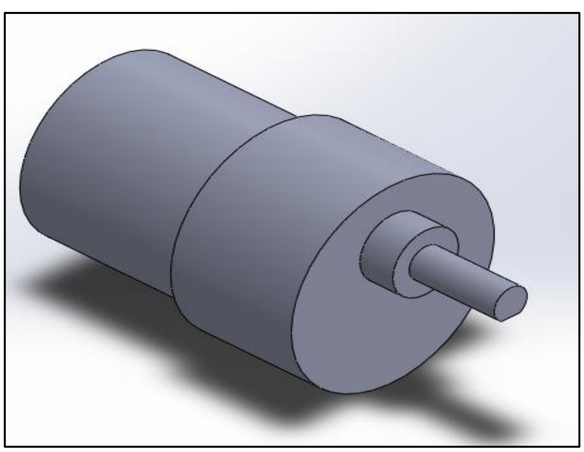
APPENDIX D

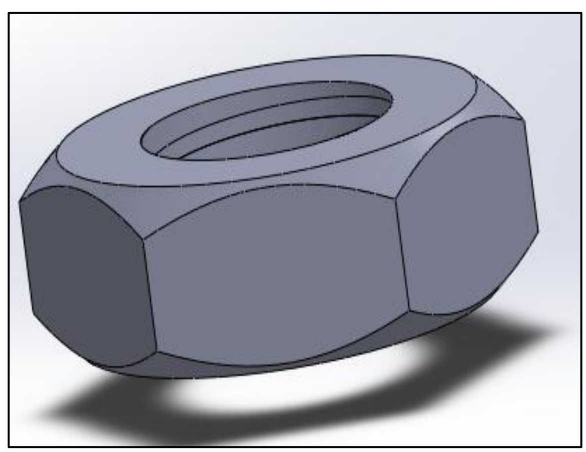
Parts of design for IP Mobile Robot

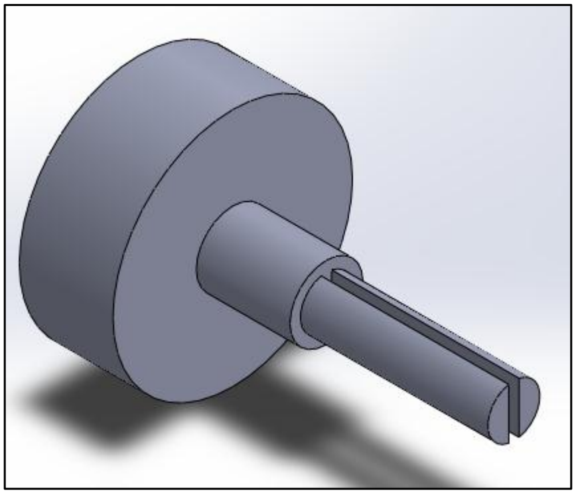


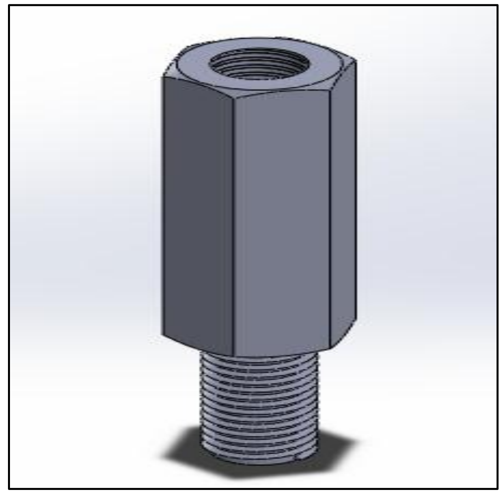


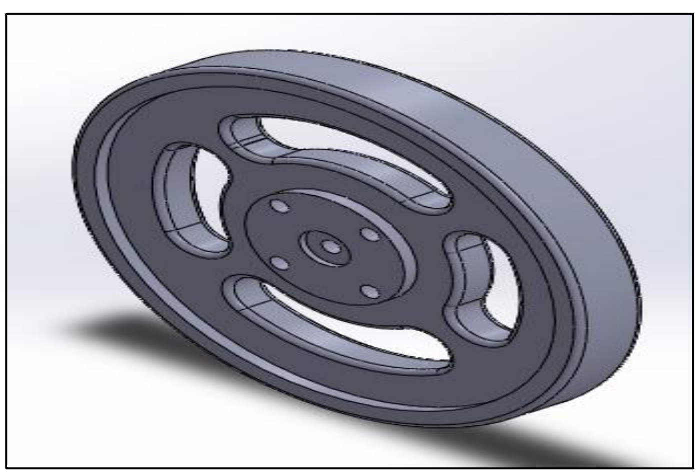




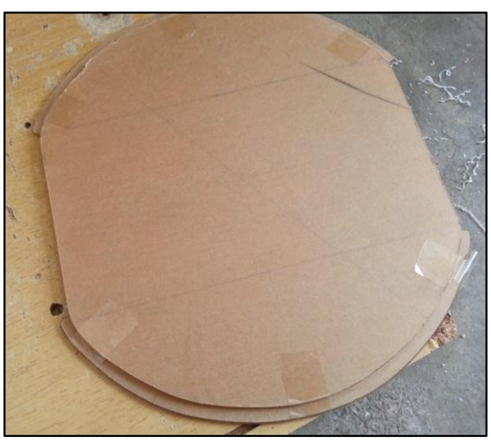


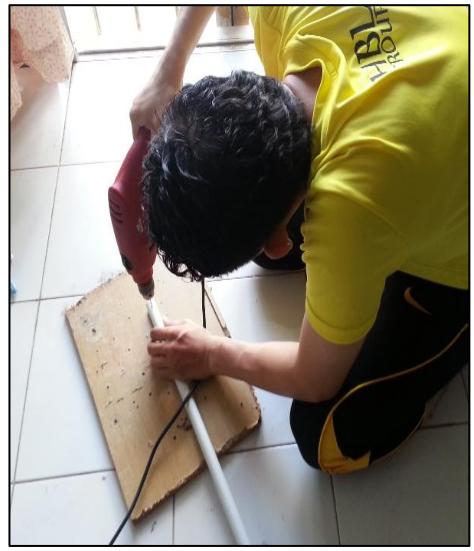


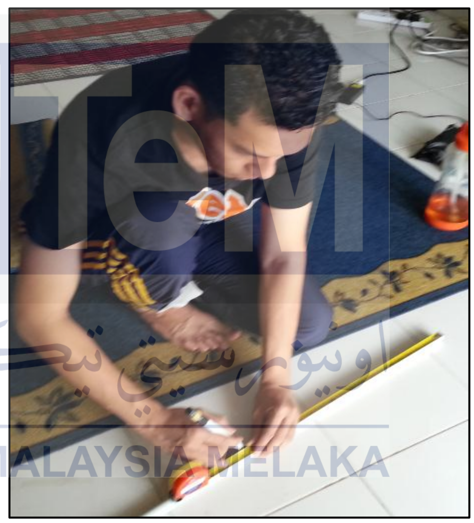




Parts of hardware for IP Mobile Robot







APPENDIX E

List and price of component

No.	Component	Product Code	Unit	Price (RM)
1	Potentiometer 100K	-	1	3.00
2	ARDUINO UNO – Main Board	ARDUINO-UNO	1	80.00
3	H-Bridge Dual motor driver	-	1	28.00
4	DC motor	SPG30-20K	2	106.00
5	Perspect	-	2	30.00
6	PVC	-	1	2.00
7	Wire (5 meter)	-	2	3.00
8	Shring	-	31	15.50
9	Spacer (long + medium + small)		24	19.80
10	Bracket motor		2	24.00
11	Coupling motor	CO-SPG	2	50.00
12	DC socket + DC jack		2	3.90
13	12V 2A adapter		2	36.00
	TOTAL		73	401.2

APPENDIX F

Programming for IP mobile robot

```
//*************************************************************************
// Matlab .fis to arduino C converter v2.0.0.29032014
// Ahmad Badrun Hisham Bin Abdul Rahim / 4BEKM / FKE 2014 / UTEM
// PSM FUZZY LOGIC INVERTED PENDULUM MOBILE ROBOT
//*************************************************************************
#include "fis_header.h"
const int fis_gcI = 2;   // Number of inputs to the fuzzy inference system
const int fis_gcO = 1;   // Number of outputs to the fuzzy inference system
const int fis_gcR = 25;  // Number of rules to the fuzzy inference system

int lastErr;
int lastTime;
unsigned long now;
unsigned char pot = 0;
float angle, angle2;
float res;
```

```
//define pin for motor A
int pinA1 = 10;    // direction pin 1
int pinA2 = 9;    // direction pin 2

//define pin for motor B
int pinB1 = 8;
int pinB2 = 7;

FIS_TYPE g_fisInput[fis_gcI];
FIS_TYPE temp[fis_gcI];
FIS_TYPE g_fisOutput[fis_gcO];

// Setup routine runs once when you press reset:
void setup()
{
   // initialize the Analog pins for input.
   // Pin mode for Input: Error
   pinMode(A0 , INPUT);

   // initialize the Analog pins for output.
   Serial.begin(9600);
   Serial.print("Welcome To Test Inverted Pendulum Mobile Robot =)");
   Serial.println();
   Serial.println();
```

```
  pinMode(5 , OUTPUT);
  pinMode(pinA1, OUTPUT);
  pinMode(pinA2, OUTPUT);
  pinMode(pinB1, OUTPUT);
  pinMode(pinB2, OUTPUT);
}

// Loop routine runs over and over again forever:
void loop()
{
  // Read Input: Error (potential)
  res = analogRead(A0)>>2;  //to chnge 10bits to 8bits

  //convert to angle
  angle = res-40;;

 g_fisInput[0] = angle;
  temp[1] = g_fisInput[0];

  //check time running
  unsigned long now = millis();

  // Read Input: Diferror
  int differ = (g_fisInput[0] - lastErr)/(now - lastTime);
  lastErr = temp[1];
```

```
lastTime = now;

g_fisInput[1] = differ;
g_fisOutput[0] = 0;
fis_evaluate();

analogWrite(5, g_fisOutput[0]);

if(angle >= 90 && angle <= 90 )  //Upright position
{
  stop();
  Serial.print("Potent Input = ");
  Serial.println(angle);
  Serial.print("Motor stop =  0");
  Serial.println();
  Serial.print("Time = ");
  Serial.println(now);
  Serial.println();
 }

 if(angle >= 91)  //91 degrees until 135
 {
  forward();
  Serial.print("Potent Input = ");
  Serial.println(angle);
```

```
    Serial.print("Motor speed forwrd=    ");
    Serial.println(g_fisOutput[0]);
    Serial.print("Time = ");
    Serial.println(now);
    Serial.println();
    }

    if(angle <= 89)  //89 degrees until 45
    {
     backward();
     Serial.print("Potent Input = ");
     Serial.println(angle);
     Serial.print("Motor speed backwrd =    ");
     Serial.println(g_fisOutput[0]);
     Serial.print("Time = ");
     Serial.println(now);
     Serial.println();
     }
  delay(1000);
}
//***********************************************************************
// Support functions for Fuzzy Inference System
//***********************************************************************
// Triangular Member Function
FIS_TYPE fis_trimf(FIS_TYPE x, FIS_TYPE* p)
```

```
{
    FIS_TYPE a = p[0], b = p[1], c = p[2];
    FIS_TYPE t1 = (x - a) / (b - a);
    FIS_TYPE t2 = (c - x) / (c - b);
    if ((a == b) && (b == c)) return (FIS_TYPE) (x == a);
    if (a == b) return (FIS_TYPE) (t2*(b <= x)*(x <= c));
    if (b == c) return (FIS_TYPE) (t1*(a <= x)*(x <= b));
    t1 = min(t1, t2);
    return (FIS_TYPE) max(t1, 0);
}
FIS_TYPE fis_min(FIS_TYPE a, FIS_TYPE b)
{
    return min(a, b);
}
FIS_TYPE fis_max(FIS_TYPE a, FIS_TYPE b)
{
    return max(a, b);
}
FIS_TYPE fis_prod(FIS_TYPE a, FIS_TYPE b)
{
    return (a * b);
}
FIS_TYPE fis_array_operation(FIS_TYPE *array, int size, _FIS_ARR_OP pfnOp)
{
```

```
    int i;
    FIS_TYPE ret = 0;
    if (size == 0) return ret;
    if (size == 1) return array[0];
    ret = array[0];
    for (i = 1; i < size; i++)
    {
        ret = (*pfnOp)(ret, array[i]);
    }
    return ret;
}
//************************************************************************
// Data for Fuzzy Inference System
//************************************************************************
// Pointers to the implementations of member functions
_FIS_MF fis_gMF[] =
{
    fis_trimf
};

// Count of member function for each Input
int fis_gIMFCount[] = { 5, 5 };
// Count of member function for each Output
int fis_gOMFCount[] = { 5 };
// Coefficients for the Input Member Functions
```

```
FIS_TYPE fis_gMFI0Coeff1[] = { 45, 60, 75.36 };

FIS_TYPE fis_gMFI0Coeff2[] = { 104.9, 120, 135 };

FIS_TYPE fis_gMFI0Coeff3[] = { 60, 75, 90 };

FIS_TYPE fis_gMFI0Coeff4[] = { 75.39, 90, 104.6 };

FIS_TYPE fis_gMFI0Coeff5[] = { 89.76, 104.8, 119.8 };

FIS_TYPE* fis_gMFI0Coeff[] = { fis_gMFI0Coeff1, fis_gMFI0Coeff2, fis_gMFI0Coeff3,
fis_gMFI0Coeff4, fis_gMFI0Coeff5 };

FIS_TYPE fis_gMFI1Coeff1[] = { -40, -10.83, 18.33 };

FIS_TYPE fis_gMFI1Coeff2[] = { 18.33, 47.5, 76.68 };

FIS_TYPE fis_gMFI1Coeff3[] = { 76.68, 105.8, 135 };

FIS_TYPE fis_gMFI1Coeff4[] = { -10.83, 18.33, 47.5 };

FIS_TYPE fis_gMFI1Coeff5[] = { 47.5, 76.68, 105.8 };

FIS_TYPE* fis_gMFI1Coeff[] = { fis_gMFI1Coeff1, fis_gMFI1Coeff2, fis_gMFI1Coeff3,
fis_gMFI1Coeff4, fis_gMFI1Coeff5 };

FIS_TYPE** fis_gMFICoeff[] = { fis_gMFI0Coeff, fis_gMFI1Coeff };


// Coefficients for the Input Member Functions

FIS_TYPE fis_gMFO0Coeff1[] = { -255, -180, -100 };

FIS_TYPE fis_gMFO0Coeff2[] = { -180, -80, 0 };

FIS_TYPE fis_gMFO0Coeff3[] = { -98.6507936507937, 1.34920634920633,
101.349206349206 };

FIS_TYPE fis_gMFO0Coeff4[] = { -0, 80, 180 };

FIS_TYPE fis_gMFO0Coeff5[] = { 100, 180, 255 };

FIS_TYPE* fis_gMFO0Coeff[] = { fis_gMFO0Coeff1, fis_gMFO0Coeff2,
fis_gMFO0Coeff3, fis_gMFO0Coeff4, fis_gMFO0Coeff5 };

FIS_TYPE** fis_gMFOCoeff[] = { fis_gMFO0Coeff };

// Input membership function set
```

```
int fis_gMFI0[] = { 0, 0, 0, 0, 0 };

int fis_gMFI1[] = { 0, 0, 0, 0, 0 };

int* fis_gMFI[] = { fis_gMFI0, fis_gMFI1};

// Output membership function set

int fis_gMFO0[] = { 0, 0, 0, 0, 0 };

int* fis_gMFO[] = { fis_gMFO0};

// Rule Weights

FIS_TYPE fis_gRWeight[] = { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 };

// Rule Type

int fis_gRType[] = { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 };

// Rule Inputs

int fis_gRI0[] = { 1, 1 };

int fis_gRI1[] = { 1, 4 };

int fis_gRI2[] = { 1, 2 };

int fis_gRI3[] = { 1, 5 };

int fis_gRI4[] = { 1, 3 };

int fis_gRI5[] = { 3, 1 };

int fis_gRI6[] = { 3, 4 };

int fis_gRI7[] = { 3, 2 };

int fis_gRI8[] = { 3, 5 };

int fis_gRI9[] = { 3, 3 };

int fis_gRI10[] = { 4, 1 };

int fis_gRI11[] = { 4, 4 };

int fis_gRI12[] = { 4, 2 };

int fis_gRI13[] = { 4, 5 };
```

```
int fis_gRI14[] = { 4, 3 };

int fis_gRI15[] = { 5, 1 };

int fis_gRI16[] = { 5, 4 };

int fis_gRI17[] = { 5, 2 };

int fis_gRI18[] = { 5, 5 };

int fis_gRI19[] = { 5, 3 };

int fis_gRI20[] = { 2, 1 };

int fis_gRI21[] = { 2, 4 };

int fis_gRI22[] = { 2, 2 };

int fis_gRI23[] = { 2, 5 };

int fis_gRI24[] = { 2, 3 };

int* fis_gRI[] = { fis_gRI0, fis_gRI1, fis_gRI2, fis_gRI3, fis_gRI4, fis_gRI5, fis_gRI6,
fis_gRI7, fis_gRI8, fis_gRI9, fis_gRI10, fis_gRI11, fis_gRI12, fis_gRI13, fis_gRI14,
fis_gRI15, fis_gRI16, fis_gRI17, fis_gRI18, fis_gRI19, fis_gRI20, fis_gRI21, fis_gRI22,
fis_gRI23, fis_gRI24 };

// Rule Outputs

int fis_gRO0[] = { 1 };

int fis_gRO1[] = { 1 };

int fis_gRO2[] = { 1 };

int fis_gRO3[] = { 2 };

int fis_gRO4[] = { 3 };

int fis_gRO5[] = { 1 };

int fis_gRO6[] = { 1 };

int fis_gRO7[] = { 2 };

int fis_gRO8[] = { 3 };

int fis_gRO9[] = { 4 };
```

```
int fis_gRO10[] = { 1 };

int fis_gRO11[] = { 2 };

int fis_gRO12[] = { 3 };

int fis_gRO13[] = { 4 };

int fis_gRO14[] = { 5 };

int fis_gRO15[] = { 2 };

int fis_gRO16[] = { 3 };

int fis_gRO17[] = { 4 };

int fis_gRO18[] = { 5 };

int fis_gRO19[] = { 5 };

int fis_gRO20[] = { 3 };

int fis_gRO21[] = { 4 };

int fis_gRO22[] = { 5 };

int fis_gRO23[] = { 5 };

int fis_gRO24[] = { 5 };

int* fis_gRO[] = { fis_gRO0, fis_gRO1, fis_gRO2, fis_gRO3, fis_gRO4, fis_gRO5,
fis_gRO6, fis_gRO7, fis_gRO8, fis_gRO9, fis_gRO10, fis_gRO11, fis_gRO12, fis_gRO13,
fis_gRO14, fis_gRO15, fis_gRO16, fis_gRO17, fis_gRO18, fis_gRO19, fis_gRO20,
fis_gRO21, fis_gRO22, fis_gRO23, fis_gRO24 };

// Input range Min

FIS_TYPE fis_gIMin[] = { 45, -40 };

// Input range Max

FIS_TYPE fis_gIMax[] = { 135, 135 };

// Output range Min

FIS_TYPE fis_gOMin[] = { -255 };

// Output range Max
```

```
FIS_TYPE fis_gOMax[] = { 255 };
//***********************************************************************
// Data dependent support functions for Fuzzy Inference System
//***********************************************************************
FIS_TYPE fis_MF_out(FIS_TYPE** fuzzyRuleSet, FIS_TYPE x, int o)
{
   FIS_TYPE mfOut;
   int r;

   for (r = 0; r < fis_gcR; ++r)
   {
      int index = fis_gRO[r][o];
      if (index > 0)
      {
         index = index - 1;
         mfOut = (fis_gMF[fis_gMFO[o][index]])(x, fis_gMFOCoeff[o][index]);
      }
      else if (index < 0)
      {
         index = -index - 1;
         mfOut = 1 - (fis_gMF[fis_gMFO[o][index]])(x, fis_gMFOCoeff[o][index]);
      }
      else
      {
         mfOut = 0;
```

```
        }

        fuzzyRuleSet[0][r] = fis_prod(mfOut, fuzzyRuleSet[1][r]);
    }
    return fis_array_operation(fuzzyRuleSet[0], fis_gcR, fis_max);
}
FIS_TYPE fis_defuzz_centroid(FIS_TYPE** fuzzyRuleSet, int o)
{
    FIS_TYPE step = (fis_gOMax[o] - fis_gOMin[o]) / (FIS_RESOLUSION - 1);
    FIS_TYPE area = 0;
    FIS_TYPE momentum = 0;
    FIS_TYPE dist, slice;
    int i;
    // calculate the area under the curve formed by the MF outputs
    for (i = 0; i < FIS_RESOLUSION; ++i){
        dist = fis_gOMin[o] + (step * i);
        slice = step * fis_MF_out(fuzzyRuleSet, dist, o);
        area += slice;
        momentum += slice*dist;
    }
    return ((area == 0) ? ((fis_gOMax[o] + fis_gOMin[o]) / 2) : (momentum / area));
}
//***********************************************************************
// Fuzzy Inference System
//***********************************************************************
```

```
void fis_evaluate()
{
    FIS_TYPE fuzzyInput0[] = { 0, 0, 0, 0, 0 };
    FIS_TYPE fuzzyInput1[] = { 0, 0, 0, 0, 0 };
    FIS_TYPE* fuzzyInput[fis_gcI] = { fuzzyInput0, fuzzyInput1, };
    FIS_TYPE fuzzyOutput0[] = { 0, 0, 0, 0, 0 };
    FIS_TYPE* fuzzyOutput[fis_gcO] = { fuzzyOutput0, };
    FIS_TYPE fuzzyRules[fis_gcR] = { 0 };
    FIS_TYPE fuzzyFires[fis_gcR] = { 0 };
    FIS_TYPE* fuzzyRuleSet[] = { fuzzyRules, fuzzyFires };
    FIS_TYPE sW = 0;
    // Transforming input to fuzzy Input
    int i, j, r, o;
    for (i = 0; i < fis_gcI; ++i)
    {
        for (j = 0; j < fis_gIMFCount[i]; ++j)
        {
            fuzzyInput[i][j] =
                (fis_gMF[fis_gMFI[i][j]])(g_fisInput[i], fis_gMFICoeff[i][j]);
        }
    }
    int index = 0;
    for (r = 0; r < fis_gcR; ++r)
    {
        if (fis_gRType[r] == 1)
```

```
{
   fuzzyFires[r] = FIS_MAX;
   for (i = 0; i < fis_gcI; ++i)
   {
      index = fis_gRI[r][i];
      if (index > 0)
         fuzzyFires[r] = fis_min(fuzzyFires[r], fuzzyInput[i][index - 1]);
      else if (index < 0)
         fuzzyFires[r] = fis_min(fuzzyFires[r], 1 - fuzzyInput[i][-index - 1]);
      else
         fuzzyFires[r] = fis_min(fuzzyFires[r], 1);
   }
}
else
{
   fuzzyFires[r] = FIS_MIN;
   for (i = 0; i < fis_gcI; ++i)
   {
      index = fis_gRI[r][i];
      if (index > 0)
         fuzzyFires[r] = fis_max(fuzzyFires[r], fuzzyInput[i][index - 1]);
      else if (index < 0)
         fuzzyFires[r] = fis_max(fuzzyFires[r], 1 - fuzzyInput[i][-index - 1]);
      else
         fuzzyFires[r] = fis_max(fuzzyFires[r], 0);
```

```
        }
      }
      fuzzyFires[r] = fis_gRWeight[r] * fuzzyFires[r];
      sW += fuzzyFires[r];
    }
    if (sW == 0)
    {
      for (o = 0; o < fis_gcO; ++o)
      {
        g_fisOutput[o] = ((fis_gOMax[o] + fis_gOMin[o]) / 2);
      }
    }
    else
    {
      for (o = 0; o < fis_gcO; ++o)
      {
        g_fisOutput[o] = fis_defuzz_centroid(fuzzyRuleSet, o);
      }
    }
}
//*****************************************************************
//Subroutine Function motor
//*****************************************************************
void stop()
{
```

```
  analogWrite(5, 0);
}
void forward()
{
  digitalWrite(pinA1, HIGH);
  digitalWrite(pinA2, LOW);
  digitalWrite(pinB1, HIGH);
  digitalWrite(pinB2, LOW);
  analogWrite (5, g_fisOutput[0]);
}
void backward()
{
  digitalWrite(pinA1, LOW);
  digitalWrite(pinA2, HIGH);
  digitalWrite(pinB1, LOW);
  digitalWrite(pinB2, HIGH);
  analogWrite (5, g_fisOutput[0]);
}
```