SITI YASMIN OTHMAN

2015

AN IMPROVED FUZZY LOGIC CONTROLLER FOR DEPTH CONTROL OF THE VIDEORAY PRO III UNDERWATER VEHICLE

SITI YASMIN BINTI OTHMAN

BACHELOR OF MECHATRONICS ENGINEERING

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

"I hereby declare read through this report entitle "An Improved Fuzzy Logic Controller for Depth Control of the VideoRay Pro III Underwater Vehicle" and found that it has comply the partial fulfillment for awarding the degree of Bachelor of Mechatronic Engineering"

Signature : ……………………………………….

Supervisor's Name : ……………………………………….

Date : ……………………………………….

AN IMPROVED FUZZY LOGIC CONTROLLER FOR DEPTH CONTROL OF THE VIDEORAY PRO III UNDERWATER VEHICLE

SITI YASMIN BINTI OTHMAN

A report submitted in partial fulfillment of the requirements for the degree of Bachelor of Mechatronic Engineering

Faculty of Electrical Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2015

“I declare that this report entitle “An Improved Fuzzy Logic Controller for Depth Control of the VideoRay Pro III Underwater Vehicle” 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 candidate of any other degree.

Signature : ………………………………………

Name : ………………………………………

Date : ………………………………………

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who provide me the possibility to complete my report. They are all being helpful and contributed their time to me. I would like to take this opportunity to show my sincere appreciation to En Mohd Shahrieel Bin Mohd Aras whose contribution in stimulating suggestions and encouragement helped me to coordinate my project especially in writing this report.

En Mohd Shahrieel Bin Mohd Aras had always been helpful in giving me suggestion while I do my project, he never hesitates to teach me all he can and share his experience to me. I would like to thank to Universiti Teknikal Malaysia Melaka (UTeM) for providing us all the needed facilities and resources such as online IEEE journal and library. This is very useful, especially when I need to do my literature review.

ABSTRACT

Nowadays, human ability is limited in deep water or seabed. There some places in underwater that human unable to reach due to dangerous and high pressure. However, the underwater vehicle is created to overcome the problem. Underwater vehicle function to help scientist make an underwater research and commonly used in deep water industries. The main point is the ability of underwater vehicle able to be controlled. However, the conventional like PD also has a problem to control nonlinear operation. The PID controller also hardly to achieved zero overshoot. Thus fuzzy logic controller is introduced to overcome the problem. In this project, the objectives are to design and improved fuzzy logic controller (FLC) for depth control of underwater vehicle (based on VideoRay Pro III), to analyze performance of system response of depth control in terms of zero overshoot, faster rise time and small steady state error using FLC and to verify the system response of the depth control using hardware implementation between Matlab/Simulink and Microbox 2000/2000C. For the methodology, the pressure sensor MX5700ap, step down voltage, microbox 2000/2000C, air compressor, thruster and multimeter are used during an experiment. The experiment was setup to analyze performance of PID and FLC in terms of zero overshoot, faster rise time and small steady state error. The final experiment carried out to study the effect of membership function of real-time fuzzy logic controller using open loop simulation data. The result shows fuzzy logic controller display a best performance in term of faster rise time, zero overshoot and small steady state error than mathematical modelling PID and real time PID.

ABSTRAK

Pada masa kini, kemampuan manusia adalah terhad di dalam air yang dalam atau dasar laut. Terdapat beberapa tempat di dalam air yang manusia tidak dapat mencapai disebabkan tekanan berbahaya dan tinggi. Walau bagaimanapun, kenderaan bawah air dibuat untuk mengatasi masalah tersebut. Fungsi kenderaan dalam air untuk membantu ahli sains membuat penyelidikan di bawah air dan biasanya digunakan dalam industri dalam air. Apa yang penting ialah keupayaan kenderaan bawah air dapat dikawal. Walau bagaimanapun, konvensional seperti PD juga mempunyai masalah untuk mengawal operasi tidak linear. Pengawal PID juga hampir tidak dicapai terlajak sifar. Oleh itu pengawal logik kabur diperkenalkan untuk mengatasi masalah ini. Dalam projek ini, objektif adalah untuk mereka bentuk dan bertambah baik pengawal logik kabur dan untuk mengawal kedalaman kenderaan bawah air (berdasarkan VideoRay Pro III), untuk menganalisis prestasi tindak balas sistem kawalan kedalaman dari segi terlajak sifar, lebih cepat meningkat masa dan kecil ralat, dan untuk mengesahkan tindak balas sistem kawalan kedalaman menggunakan pelaksanaan perkakasan antara Matlab / Simulink dan Microbox 2000 / 2000C. Untuk kaedah ini, sensor tekanan MX5700ap, penurun voltan, microbox 2000 / 2000C, pemampat udara, pendorong dan multimeter digunakan semasa eksperimen. Eksperimen adalah persediaan untuk menganalisis prestasi PID dan kawalan logik kabur dari segi terlajak sifar, lebih cepat meningkat masa dan kecil ralat. Percubaan terakhir dijalankan untuk mengkaji kesan fungsi keanggotaan masa nyata pengawal logik fuzzy menggunakan simulasi gelung terbuka. Hasil kajian menunjukkan pengawal logik fuzzy memaparkan prestasi yang terbaik dalam tempoh lebih cepat meningkat masa, terlajak sifar dan kecil ralat daripada pemodelan matematik PID dan masa sebenar PID.

TABLE OF CONTENTS

CHAPTER TITLE

3 METHODOLOGY PAGE

CHAPTER **TITLE** **PAGE**

4 **RESULT AND DISCUSSION**

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

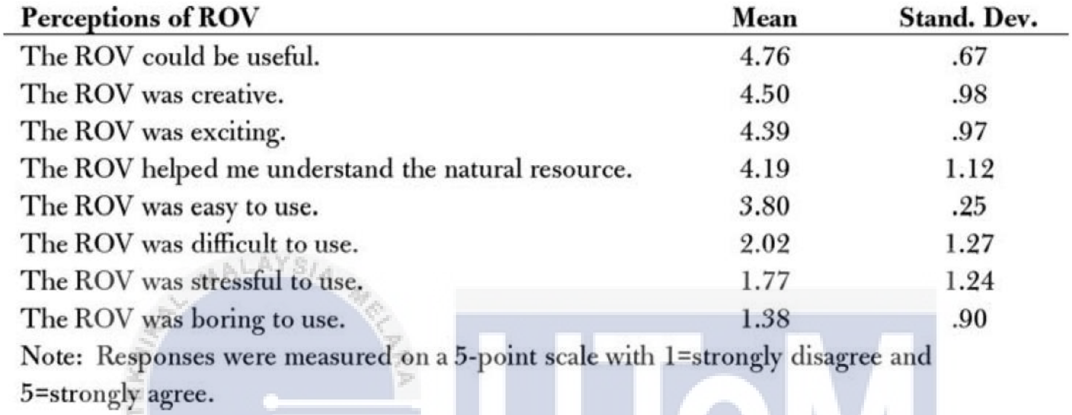
1.1 Project background

The ROV is a tethered underwater vehicle which is in unmanned categories. ROV commonly used in deep water industries which is involved in oil and gas activities. ROVs widely uses in offshore construction, military and scientific community. ROV used to as replace manned rescue system in military and helps scientist to make a research about underwater knowledge, deep sea animal and plants. The final year project is focused in designing the fuzzy logic controller to improve system response in terms of minimum overshoot, faster rise time, small, steady state error for depth control of the ROV (based on VideoRay Pro ROV III)

1.2 Motivation

The main encouragement to choose this title ‘An improved Fuzzy Logic Controller for depth control of the VideoRay Pro III underwater vehicle’ rather than other project because ROV is an interesting knowledge. ROV are widely use in several of application. ROV also can be used to explore science or natural environment at seabed. In paper [8] mentioned about impacts using ROV which is the two hundred ninety individuals completed the questionnaire in summer 2005. The respondent in the study raged from 12 to 84 years old. The question was designed to examine the impacts using an ROV had on individual’s interactions with and connection to the natural environment. Table 1.1 shows that positive perceptions regarding ability of the ROV to be fun, safe and interesting to use as well as safe, low impact and conservation oriented. Furthermore, respondents suggests it was good science and research tool. However, negative perceptions were noted only among adults and included the possibility of becoming disconnected from nature. From the case study shows that the ROV offers an alternative to who's loved to explore natural environment but may fear the water, have physical limitations or want to explore depths not physically possible.

Table 1.1: ROV Perceptions [8]



Perceptions of ROV	Mean	Stand. Dev.
The ROV could be useful.	4.76	.67
The ROV was creative.	4.50	.98
The ROV was exciting.	4.39	.97
The ROV helped me understand the natural resource.	4.19	1.12
The ROV was easy to use.	3.80	.25
The ROV was difficult to use.	2.02	1.27
The ROV was stressful to use.	1.77	1.24
The ROV was boring to use.	1.38	.90

Note: Responses were measured on a 5-point scale with 1=strongly disagree and 5=strongly agree.

Other than that, ROV had function to solve underwater tragedy like deep water horizon oil spill in the Gulf of Mexico. The broken oil pipe in the Gulf of Mexico dumped 300 million gallons of water a day into Charles River (figure 1.1) and affected the water supply to for two million people The problem was determine by enlisted ROV to investigate the immediate wellhead area. The ROV discover two leaks which one from a kink in the riser and a primary link from the end of the riser, where it broken off from the rig. It a risk for human to dive in 5000' down in the Gulf of Mexico and ROV is a solution for critical situation [7].

Mysterious tragedy MH370 also used ROV in search black box in a seabed of Hindi Ocean. ROV can firm, scan and crucially pick up things from the seabed (figure 1.2).Other example is Remora which can function 6000metres which is used in salvage AF447 and other crashed planes [6]. Another important thing in ROV is the system response for depth control. In order to develop a better response in depth control for future, an analysis from fuzzy logic controller is introduced.

Figure 1.1: Broken 21” oil pipe 5000’ down in the Gulf of Mexico taken by ROV [7]

Figure 1.2: ROV helps missing MH370 [6]

1.3 Problem Statement

There are many problems happen in a remotely underwater vehicle that related with control system. The conventional controller like PD also has a problem with depth control of the ROV. The PD controller is not suitable in a nonlinear operation of depth control. The conventional PID controller also hardly to achieve zero overshoot in system response of depth control. This problem is crucial because it might cause damage to the remotely underwater vehicle if it contact directly with the seabed. Thus, intelligent control system such as fuzzy logic controller is needed in order to solve PD and PID problems.

The fuzzy logic controller is considered as new controller method to improve depth control of the ROV. Therefore, a shifting membership function will be used to analyze the effect of system response of depth control. The results is one simple contribution to this field of study.

1.4 Objectives

The objectives of final year project are:-

1. To design and improve the fuzzy logic controller for depth control of underwater remotely operated vehicle (based on VideoRay PRO ROV III)
2. To analyze performance of system response of depth control in terms of zero overshoot, faster rise time and small steady state error using FLC
3. To verify the system response of the depth control using hardware implementation between MATLAB/Simulink and Micro-box 2000/2000C.

1.5 Scope and Limitation

This project is mainly about the control system. The prototype based on VideoRay Pro III was used in this project. The prototype of the ROV is built by following parameters of thruster construction VideoRay Pro III (2 horizontal thruster and 1 vertical thruster). The dimension of prototype built up by refer to VideoRay Pro 3s (30.5 x 22.5 x 21cm). Since this project related with depth control, the movement of ROV covered a vertical up and down. The depth of ROV while doing an experiment is set less than 5m only. This project were carried out in a condition disturbance will be assumed to zero. This project were implement the intelligent control system which is Microbox 2000/2000C. The experiment related with Microbox 2000/2000C was setup in CIA Lab, FKE since a Microbox 2000/2000C is prohibited to use out of CIA lab, FKE.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter will focus on ROV depth control. The topic cover about controllers that use in ROV. The main point in this chapter are to know an advantages and disadvantages each of controllers that effect the performance of depth control. Related journal of implementation controller for underwater robot also will be review and study to gain a knowledge and able to improve an existing method.

2.1.1 Conventional controller

The conventional controller includes PID, PD and PI controller. Conventional controller widely used in industrial control system. In term of stability and overshoot, intelligent system is more effective than conventional system. However, both system have their own advantages and disadvantaged by depending on applications.

- PID controller is a combination proportional, integral and derivative controller. The sum of three elements to calculate the output of PID controller.

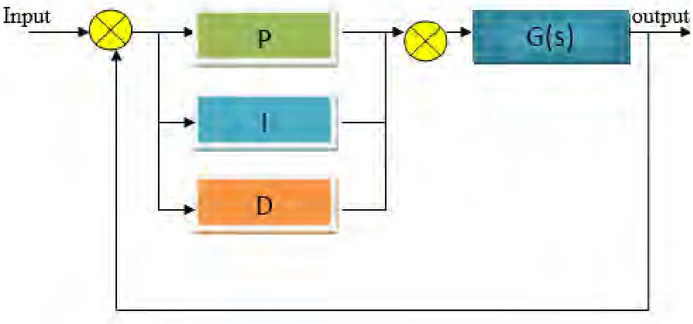


Figure 2.1: Block diagram of PID controller

Where,

K_p: Proportional gain, a tuning parameter

K_i : Integral gain, a tuning parameter

K_d: Derivative gain, a tuning parameter

e : Error = SP-SV

t : Time or instantaneous time (the present)

T : Variable of integration; takes on values from time 0 to the present t.

2.1.2 Intelligent controller

Nowadays, an intelligent control shown some success in a control method. For example neural network, fuzzy logic controller and genetic algorithm. Intelligent controller like fuzzy logic show a highly time consuming but it is suitable for nonlinear motion and need some tuning process in order to increase a performance. Fuzzy logic widely used in washing machine, rice cooker, and others. Block diagram of fuzzy logic controller as shown in Figure 2.2.

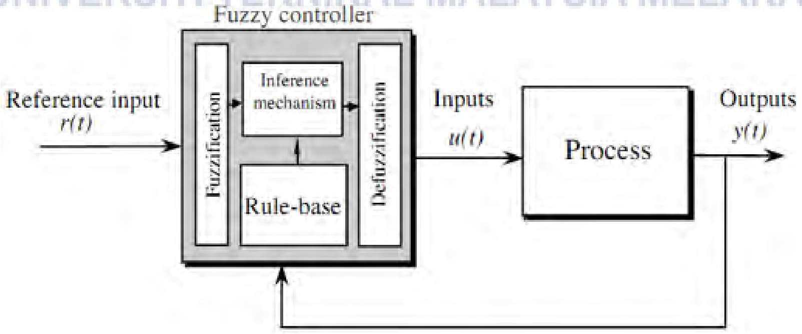


Figure 2.2: Block diagram of fuzzy logic controller [9]

Component:

- Rule-base – set of rules regarding on how to control
- Fuzzification – transforming process of numeric input into any form that can be used and detected by inference

- Inference mechanism – this mechanism use information that is formed from fuzzification and decide which rules will be applied in that of situation
- Defuzzification – convert the conclusion into numeric input for the plant that is reached by inference mechanism.

2.2 Related Previous Work

According to S. M. Zanoli [1] the PID controller with an input smoothing pre filter is introduced as a tuning of the pre filter parameters shown to reduce an overshoot. The Newtonian or a Lagrangian formalism is used as a system equation to derive a general non-linear model that described the dynamic of an underwater vehicle. The depth control divide in two different method which is continuous input smoother (CIS) as shown in Figure 2.3 and discrete fuzzy smoother DFS as shown in figure 2.4 [1].

Figure 2.3: CIS Control scheme [1]

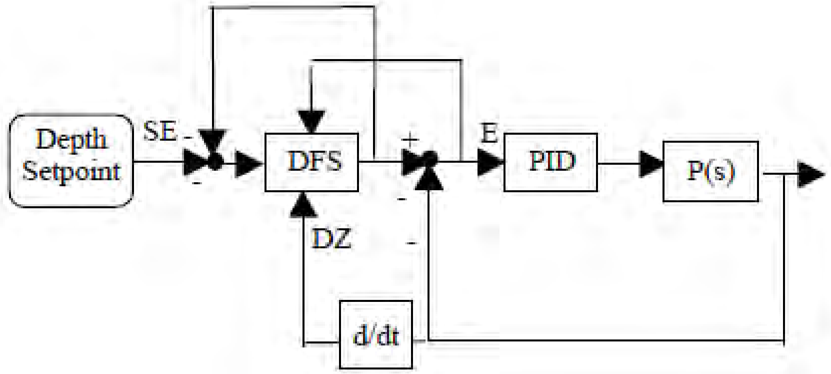


Figure 2.4: DFS Control Scheme [1]

The both method are concern to keep the amplitude of the overshoot is drastically limited to a depth set-point change while keeping the response time reasonably contained. The reason is to make sure the vehicle's safety while near the bottom and in order to prevent possible cable stress. The first method is CIS. This method effective in the reduction of the overshoot in the set point response. The disadvantaged of this method is CIS parameters need to be tuned off line and different tuning suit with different working conditions. The fuzzy-PID ideas (discrete fuzzy smoother) is introduced to overcome a problem faced by CIS. The response using DFS differs greatly, the rising time is kept reasonably low and overshoot is practically suppressed compared to CIS as shown in figure 2.5 [1].

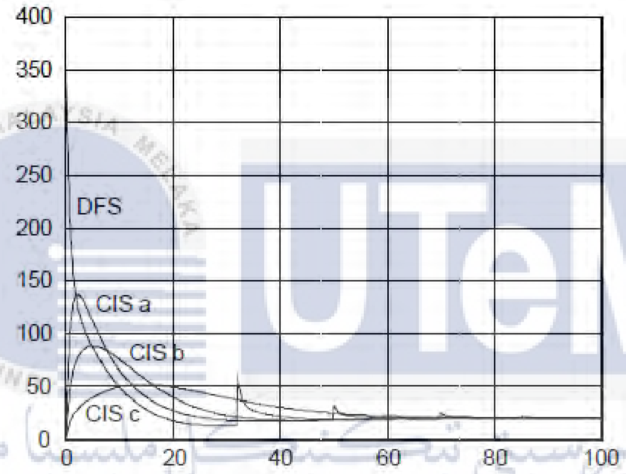


Figure 2.5: Control effort of the CIS and DFS control scheme for the linear model [1]

According to Aras M.S.M [2], system identification is used in developing ROV for depth control. System identification concept is a process of obtaining model based on a set of data that collected from experiments. The first step is ROV will be tested in open loop condition in order to get input and output signal value which is using 5m as a set point for depth control. The recorded value from input and output was analyzed to infer a model as shown in Figure 2.6. Then, system identification toolbox in MATLAB will be applied to generate model of ROV. This research also make a comparison between mathematical modelling and system identification. The result show a mathematical modelling better than system identification as shown in figure 2.7 but system identification more towards in term of real time applications which is includes environmental disturbances in lab tank test or in a swimming pool [2].

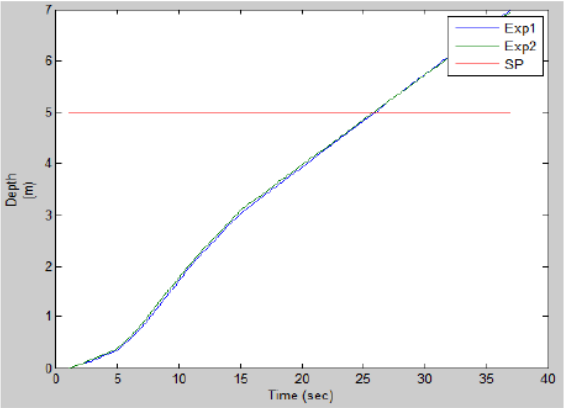


Figure 2.6: Experiment Results Testing Open Loop System for ROV [2]

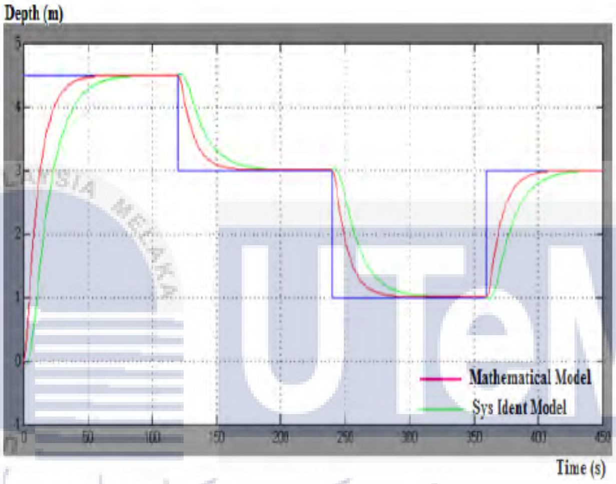


Figure 2.7: Comparison between Mathematical Models with System Identification Model [2]

According to M.S.M Aras [3], the investigation on linear approximation control surface method for tuning single input fuzzy logic controller (SIFLC) is focus on slope of linear equation as shown in Figure 2.8. Firstly, the optimum operating conditions are determined in order to generalize output equation of linear surface. The derivation from output equation of linear surface, it show that the control surface shape is determined by the peak location of the input and output of membership function. Lastly examples of different linear approximation and its original relationship to FLC will be described. In this journal, the best slope of linear equation is 0.5 as shown in figure 2.9 where gives better performances than others. If the slope bigger, the response of system is not good and chattering happen. In depth control, the chattering must be eliminated in order to avoid damage to the ROV [3].

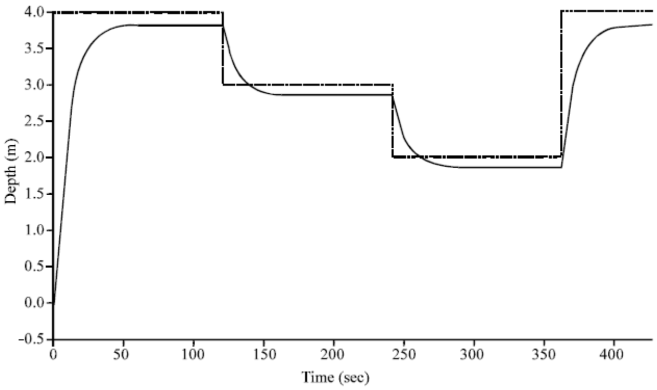


Figure 2.8: The system response of ROV system based on linear equation [3]

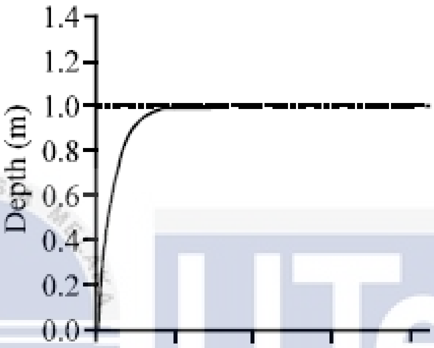
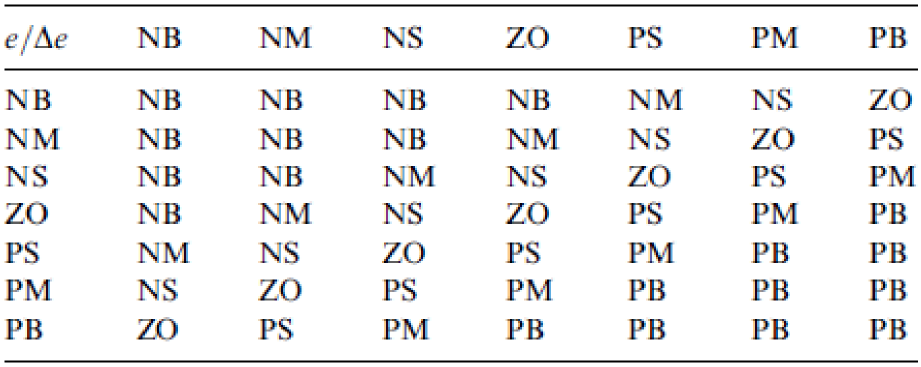


Figure 2.9: Slope of 0.5 linear equation [3]

According to G.N Roberts [4], fuzzy logic controller (FLC) can be applied to the system that nonlinear and where the mathematical models are difficult to obtain. The journal stated FLC is able to apply heuristic rules that reflects experiences of the human experts while conventional PD controllers suitable for high sensitivity and tend to increase the stability of the overall feedback. PD controllers also can reduce overshoot and able to using larger gain by adding damping to the system. The main tasks of FLC part of fuzzy like PD is a structure which means need to determine the architecture of a controller input/output variables of controller, fuzzy control rules as shown in Table 2.1 and the number of rules [4].

Table 2.1: The rule base of a Fuzzy-like PD in tabular form [4]



$e/\Delta e$	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZO
NM	NB	NB	NB	NM	NS	ZO	PS
NS	NB	NB	NM	NS	ZO	PS	PM
ZO	NB	NM	NS	ZO	PS	PM	PB
PS	NM	NS	ZO	PS	PM	PB	PB
PM	NS	ZO	PS	PM	PB	PB	PB
PB	ZO	PS	PM	PB	PB	PB	PB

Naming of rule base as follows:-

NB= Negative big

NM= Negative medium

NS= Negative small

PB= Positive big

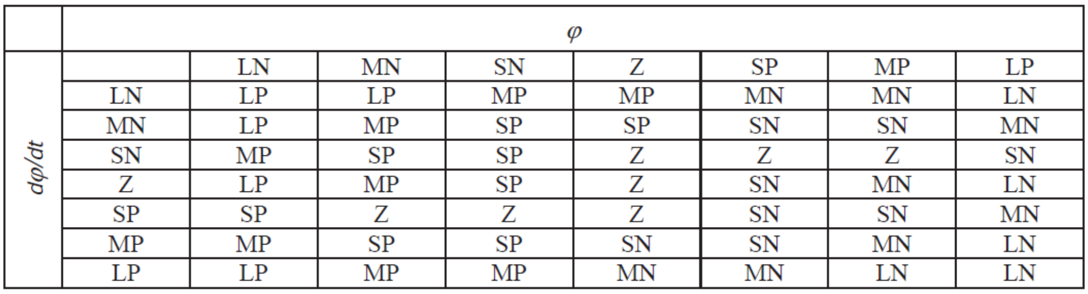
PM= Positive medium

PS= Positive small

ZO= Zero

According to Andrzej Zak [5], the author designed a system which consists of fuzzy logic control system in disturbance condition. The mathematical model of underwater vehicle is generated in a first stage which is the complex problem. In analysis of sailing objects, the basic environment's disturbances like waviness, wind are included. The fuzzy controller is built by using membership function and rule matrix as shown in Table 2.2 The outcome from this journal, fuzzy logic is suitable for nonlinear operation and stabilization in environment's disturbances compared to classical PID controller. PID controller is easy to implement but changing parameters effect times of regulation prolongation [5].

Table 2.2: Rule matrix of fuzzy controller [5]



		φ						
$d\varphi/dt$		LN	MN	SN	Z	SP	MP	LP
	LN	LP	LP	MP	MP	MN	MN	LN
	MN	LP	MP	SP	SP	SN	SN	MN
	SN	MP	SP	SP	Z	Z	Z	SN
	Z	LP	MP	SP	Z	SN	MN	LN
	SP	SP	Z	Z	Z	SN	SN	MN
	MP	MP	SP	SP	SN	SN	MN	LN
	LP	LP	MP	MP	MN	MN	LN	LN

Naming a rule matrix of the table as follows:-

LN= Large negative

MN= Medium negative

SN= Small negative

LP= Large Positive

MP= Medium Positive

SP= Small Positive

Z= Zero

2.3 Summary of Literature Review

Table 2.3: Summary of controller

Controller	Advantages	Disadvantages
Continuous input smoother (CIS) [1]	Effective in the reduction of the overshoot in the set point response	Required different tuning suit with different working conditions.
Single input fuzzy logic controller (SIFLC) [3]	Slope of linear equation give optimum performances	Different control surface of piecewise linear region effect performance of depth control
PD controller [4]	Able to reduce overshoot. Suitable for larger gain by adding damping to the system	Not suitable for non-linear operation.
PID Controller with disturbance occurrence [5]	Easy to implement	Changing parameters effect times of regulation prolongation
Fuzzy Logic Controller [1][3][4][5]	Suitable for non-linear operation and able to apply heuristic rules that reflect experiences of the human experts.	Fine tuning process is highly time consuming.

According to journal [1], the continuous input smoother (CIS) method is effective in the reduction of the overshoot in the set point response but it requires a different tuning suit with different condition. Journal [2] shows that by using system identification for depth control, it is more towards in term of environmental disturbances but lack in term of system response. In the journal [4], PD controller is able to reduce overshoot in the system, but it is not suitable for non-linear operation. Journal [5] state that PID is easy to implement, but changing parameters in disturbance occurrence will effect times of regulation prolongation. According to journal [1][3][4][5], the fuzzy logic controller is suitable for non-linear operation and have a better time compare to CIS controller but the fine tuning process is highly time consuming. Therefore, the fuzzy logic controller is chosen to be a final year project.

CHAPTER 3

METHODOLOGY

3.1 Project Plan and Process Flow

The final year project period is covered for two semester. First semester focus on research regarding existing method control system and simulation analysis for depth control of the ROV. The prototype of VideoRay Pro III and analysis the system response of real time with Microbox 2000/2000C covered for semester two. The effect of system response by shifting membership function of fuzzy logic controller also included in this project. The K-chart also helped to categorize a project as shown in Figure 3.1. The flowchart of the project report is shown in figure 3.2. The flow of final year project as shown in Figure 3.3.

Figure 3.1: K-chart vehicle

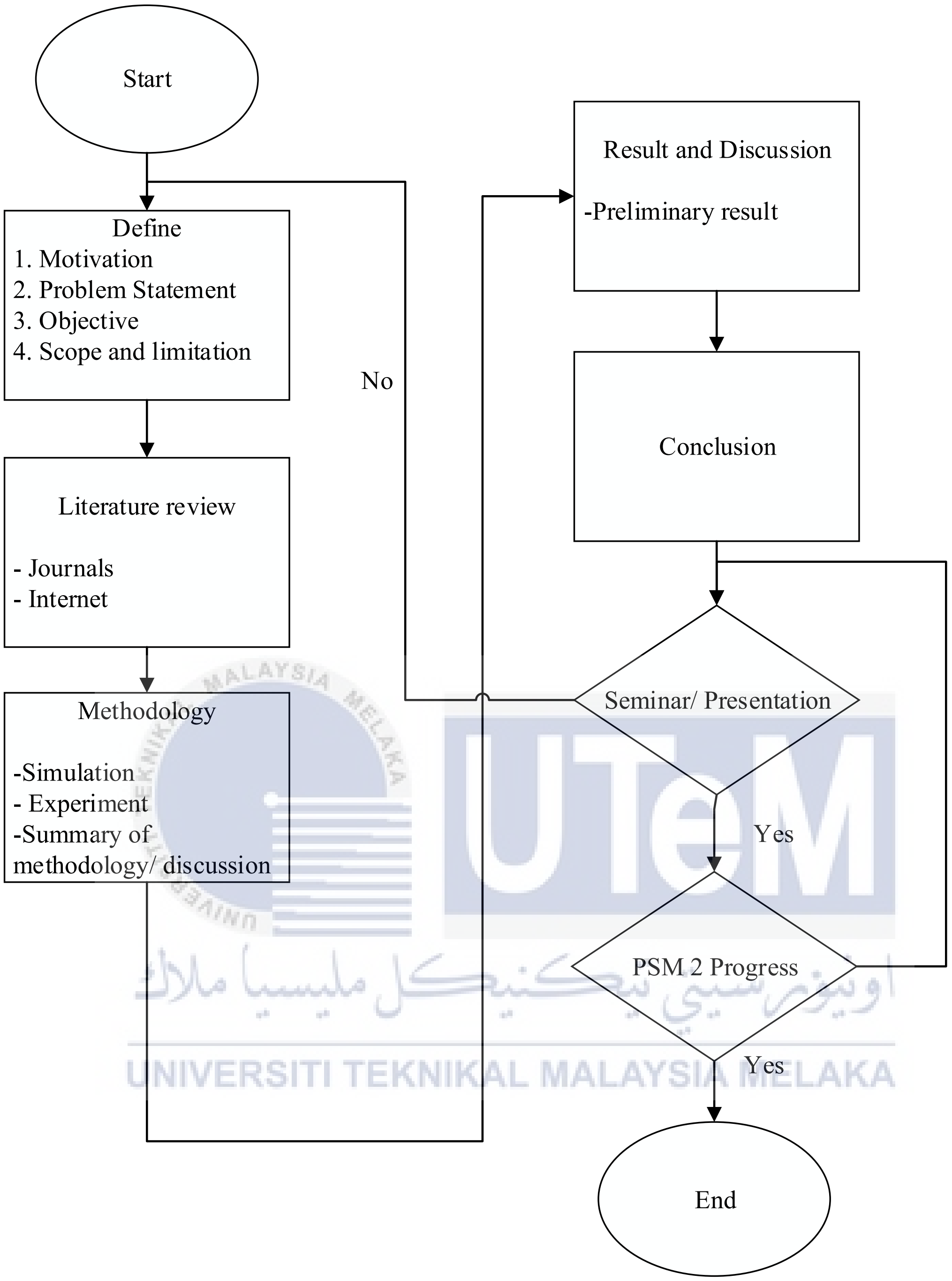


Figure 3.2: Flowchart diagram of project report PSM 1

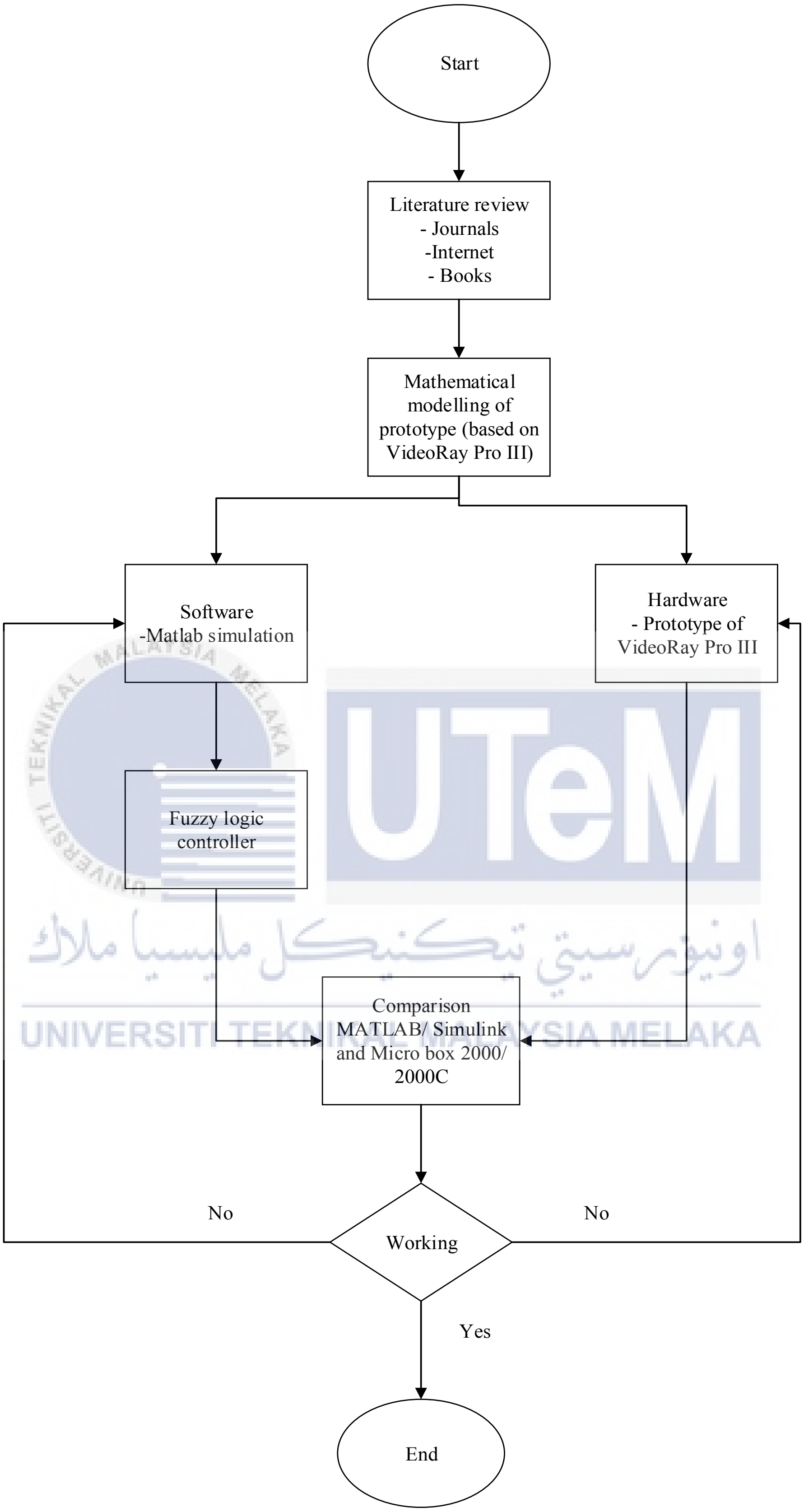


Figure 3.3: Flowchart diagram of project PSM 2

3.2 Technical research

The technical research for this project related with mechanical, electronics, programming language and sensor. All gathered information that will be used in this project are comes from journals, internet, and books. This project will be divide into three categories:-

1. Hardware

   Make a prototype and use a suitable sensor (pressure sensor)

2. Software

   Simulation using Matlab/Simulink, Proteus (ISIS) and MikroC

3. Experiment

   Experiment related with Microbox 2000/2000c and prototype (based on VideoRay pro III thruster configuration)

3.3 Hardware

3.3.1 Prototype (Based on VideoRay Pro III underwater vehicle)

Prototype based on VideoRay Pro III underwater vehicle will be used in this project. VideoRay Pro III is a small inspection class personal as shown in Figure 3.4. The vehicle have three control thrusters, one for vertical movement and two for horizontal movement. It designed for depth control of 152 meters deep. The vehicle include sensor, front and rear facing camera, depth gauge and heading meter. Mapping thruster based on this underwater vehicle will be implement in this project as shown in Figure 3.6.

Figure 3.4: VideoRay Pro III underwater vehicle

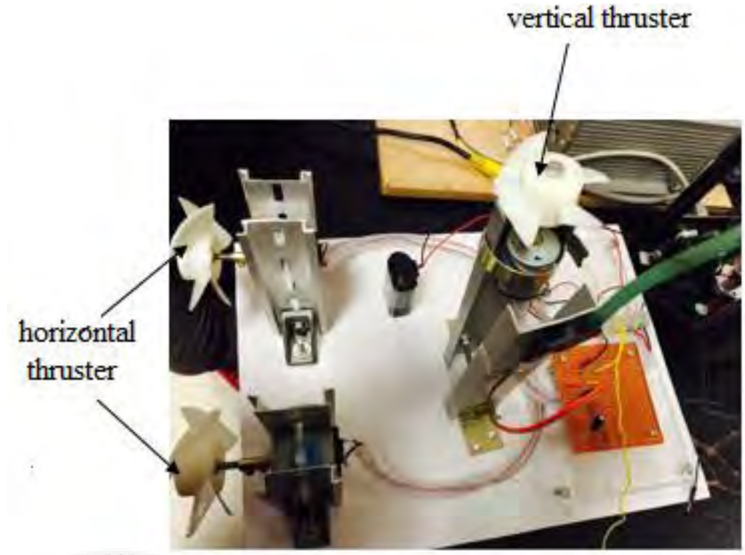


Figure 3.5: Prototype based on VideoRay Pro III

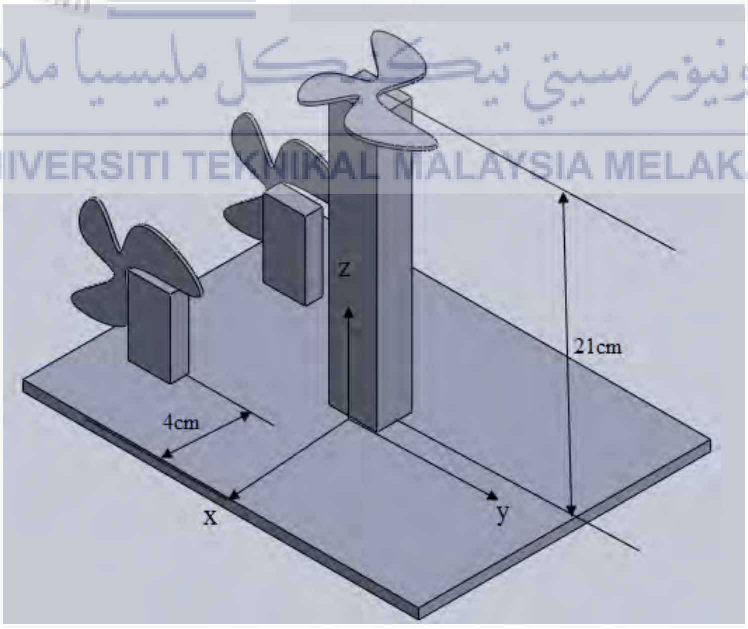


Figure 3.6: Thruster configuration based on VideoRay Pro III

3.3.2 Microbox 2000/2000C

Microbox 2000/2000c is a high performance industrial PC with no moving parts inside and in a compact size as shown in Figure 3.7. This industrial PC designed to works with MATLAB/Simulink and related control modules. It allow user to make a simulation of control system without complicated debug process. The prototype was connect with Microbox 2000/2000C to collect a real time data.

Figure 3.7: Microbox 2000/2000c

3.3.3 Pressure sensor

Pressure sensor MPX5700 as shown in Figure 3.8 has a high level analog output signal that is proportional to the applied pressure. This sensor suitable to use in research project and a cost is reasonable. This sensor is a more practical with the light weight of body. Maximum pressure for MPX 5700 is 700KPa (7 bar). The pressure sensor will connect with step down voltage and pressure sensor circuit. MPX5700 function to detect air pressure in order to control depth of ROV.

Figure 3.8: Pressure sensor MPX5700

3.4 Software

3.4.1 Matlab/Simulink

Matlab/ Simulink use to draw a block diagram and as a simulation of control system method. Simulink Library as shown in Figure 3.9. The control system was build up using Simulink as shown in Figure 3.10.

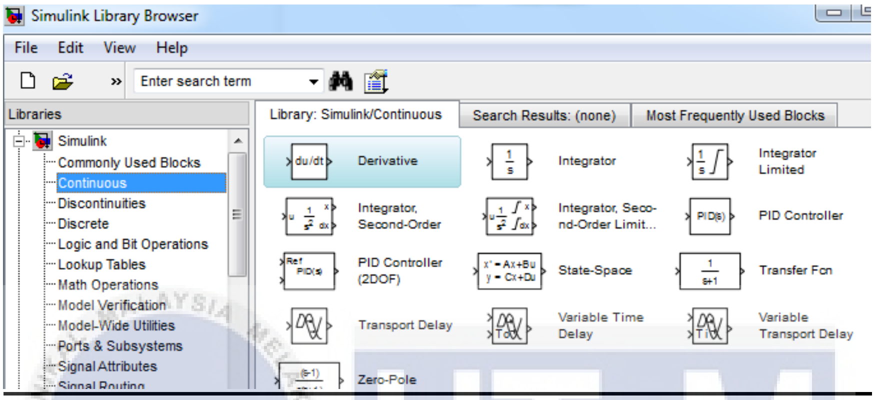


Figure 3.9: Simulink library

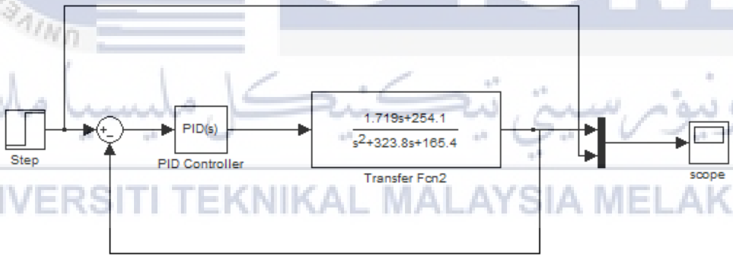
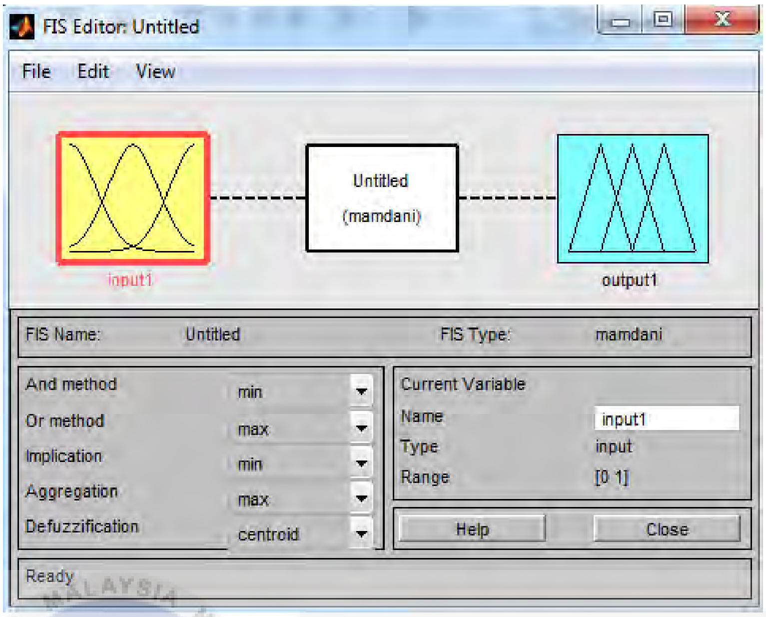


Figure 3.10: PID closed loop system

3.4.2 Fuzzy Logic controller

Matlab software also used to create a fuzzy logic controller based on fuzzy logic toolbox as shown in Figure 3.11. In order to design a closed loop fuzzy logic controller, the pressure sensor experiment need to be performed and able to obtain real-time data. The data will be evaluated by system identification. The system identification use to identify the best fit data and to generate transfer function equation which is able to implement in close loop fuzzy logic controller system. The rules editor use to construct rule statement of the fuzzy logic as shown in Figure 3.13. Figure 3.14 show the rule viewer of rules that we set and figure 3.15 show surface of rules in 3D.



FIS Name: Untitled
FIS Type: mamdani
And method: min
Or method: max
Implication: min
Aggregation: max
Defuzzification: centroid
Current Variable
Name: input1
Type: input
Range: [0 1]
Ready

Figure 3.11: FIS Editor

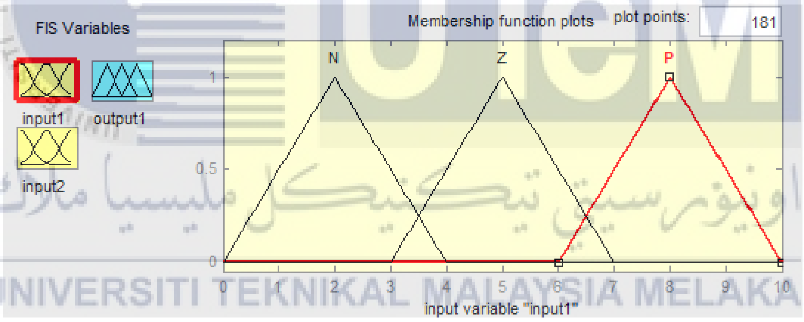
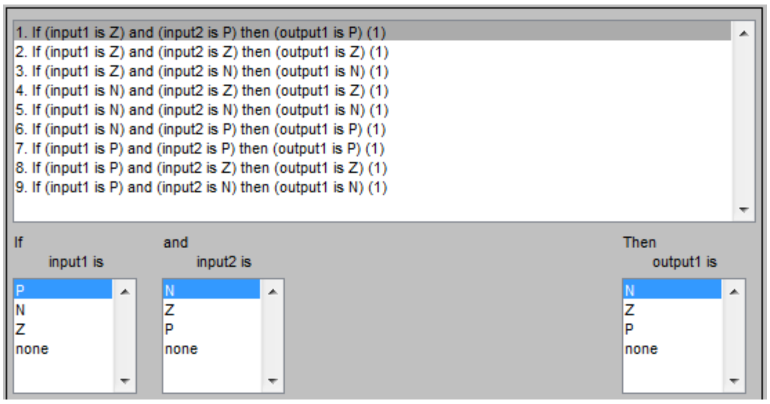


Figure 3.12: Input 1 membership function



1. If (input1 is Z) and (input2 is P) then (output1 is P) (1)
2. If (input1 is Z) and (input2 is Z) then (output1 is Z) (1)
3. If (input1 is Z) and (input2 is N) then (output1 is N) (1)
4. If (input1 is N) and (input2 is Z) then (output1 is Z) (1)
5. If (input1 is N) and (input2 is N) then (output1 is N) (1)
6. If (input1 is N) and (input2 is P) then (output1 is P) (1)
7. If (input1 is P) and (input2 is P) then (output1 is P) (1)
8. If (input1 is P) and (input2 is Z) then (output1 is Z) (1)
9. If (input1 is P) and (input2 is N) then (output1 is N) (1)

Figure 3.13: Rules of fuzzy logic controller

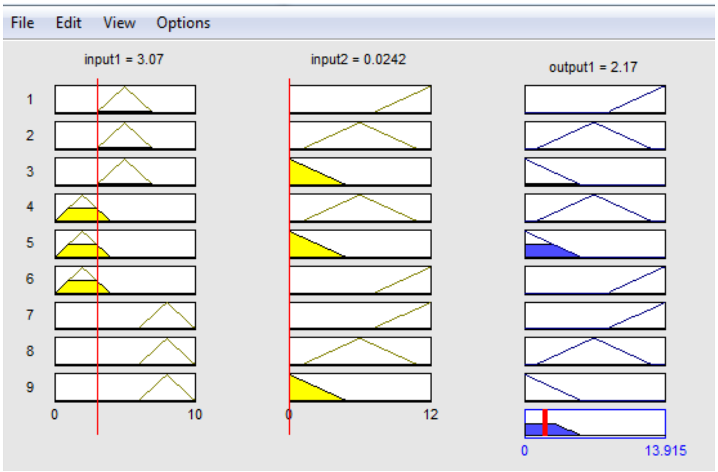


Figure 3.14: Rules viewer of fuzzy logic controller

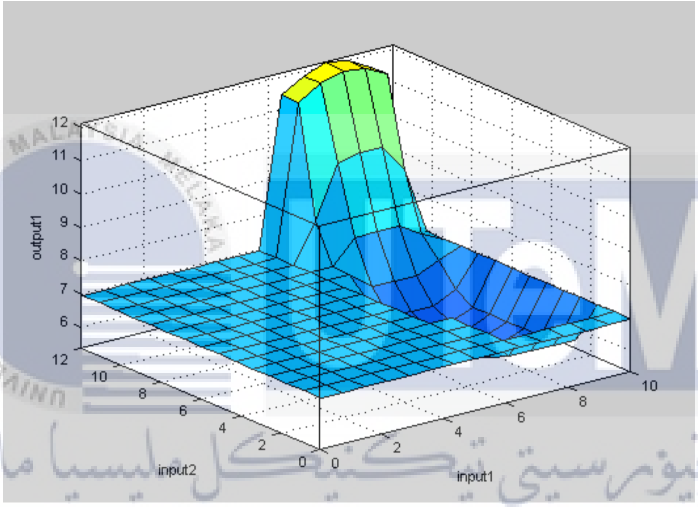


Figure 3.15: Rules surface of fuzzy logic controller

3.4.3 Mathematical Modelling

Mathematical modelling use by substitute properties and coefficients of VideoRay Pro III data in journal [10] into matrix using Newton-Euler motion equation. The generated equation will import to workspace Matlab as shown in Figure. The equation use in simulation modelling of ROV to produce an output response of PID. Mathematical modelling is derived from the Newton-Euler motion equation 3.1.

$$M\acute{v} + C(V) + D(V)V + G = T \qquad (3.1)$$

Where

M= mass and inertia matrix

$C(V)$= Coriolis and centripetal

$D(V)$= Hydronamic damping

V= Velocity

G= Gravitional and buoyancy vector

T= External force and torque input vector

3.4.4 State space

State space is the one of methods for controller design. State space using to analyze whether the open loop system without any control is stable. Controllability system given by equation. The observability of the system can be determined from the input U(t) and the output Y(t) over a finite interval of time. Open loop system can be analyzed the controllability and observabilty by giving appropriate coding to command window Matlab.

$$\dot{x} = Ax + Bu \tag{3.2}$$

$$y = Cx + Du \tag{3.3}$$

Where

A=System Matrix

X=State Vector

B=Control Input Matrix

U=Input Vector

C=Output Matrix

Y=Output Vector

D=Direct Matrix

3.5 Experiment Implementation

3.5.1 Experiment 1: Use Simulink software to get an output response of PID

Objective:

To analyze performance of PID in terms of no overshoot, faster rise time, and small steady state error.

Procedure**:**

1. The Matlab/ Simulink software was used to find a controller output performance.
2. The obtained data was being process by Matlab (Simulink) to implement a transfer function that can be obtained [7].
3. After obtain a transfer function, a block diagram was drawn as shown Figure 3.16. The output performance is evaluated by change a parameter of PID.
4. A same step was repeated by using a different value of Kp, Ki, and Kd.
5. The result was analyze in term of overshoot, rise time and steady state error as shown in Figure 3.17

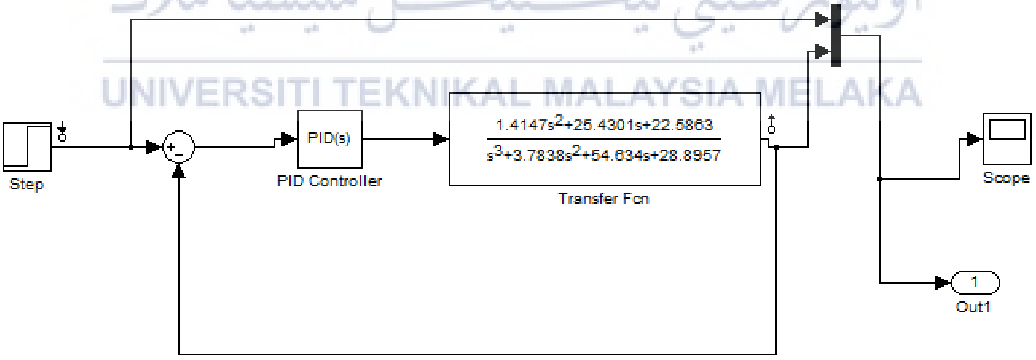


Figure 3.16: Block diagram of PID close loop system

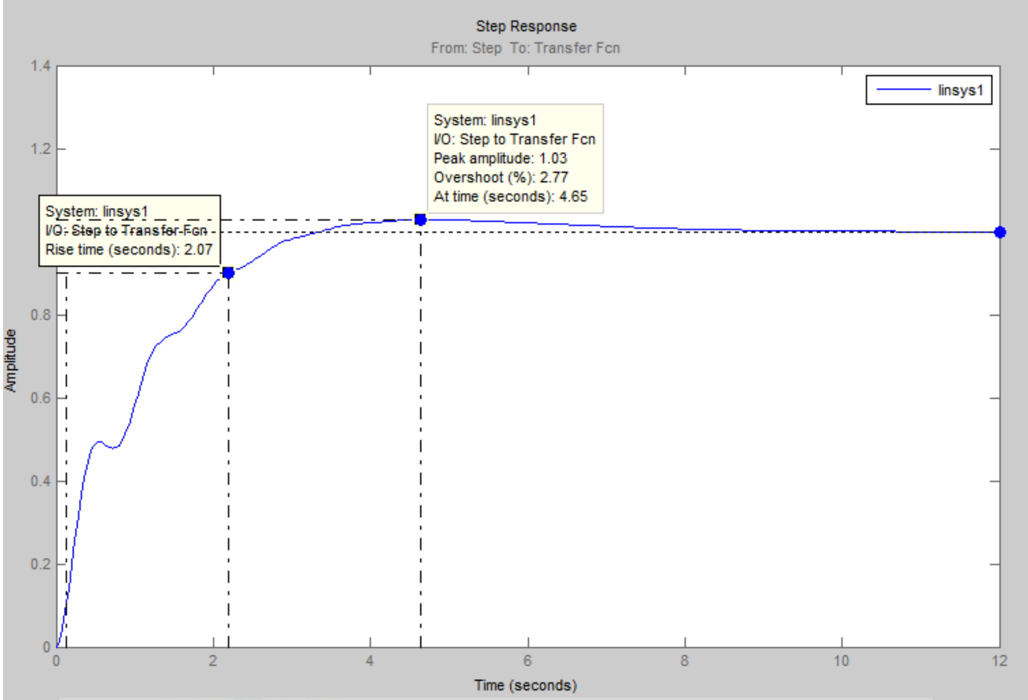


Figure 3.17: Output graph for control system

3.5.2 Experiment 2: Fuzzy logic controller using Matlab Simulink

Objective:

To analyze system response in terms of no overshoot, faster rise time and small steady state error using FLC.

Procedures:

1. Fuzzy logic controller was implemented by using Matlab/Simulink as shown in Figure 3.18.
2. Block diagram was constructed by obtain transfer function from journal[7]
3. The 3x3 rules was using in fuzzy logic controller. The nine rules were implement in rules editor.
4. Output response of fuzzy logic controller was analyze in term of overshoot, rise time, settling time and steady state error.

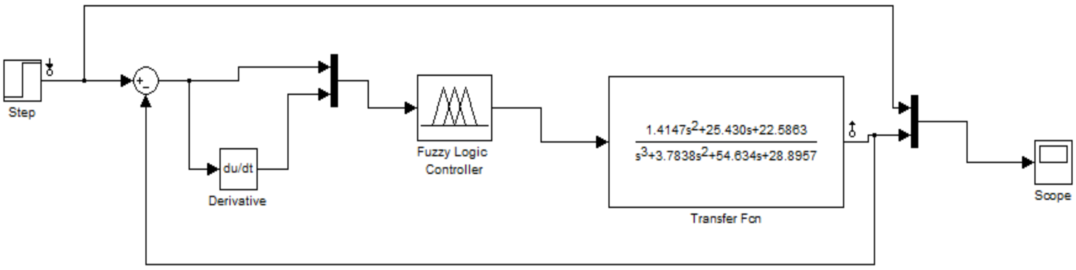
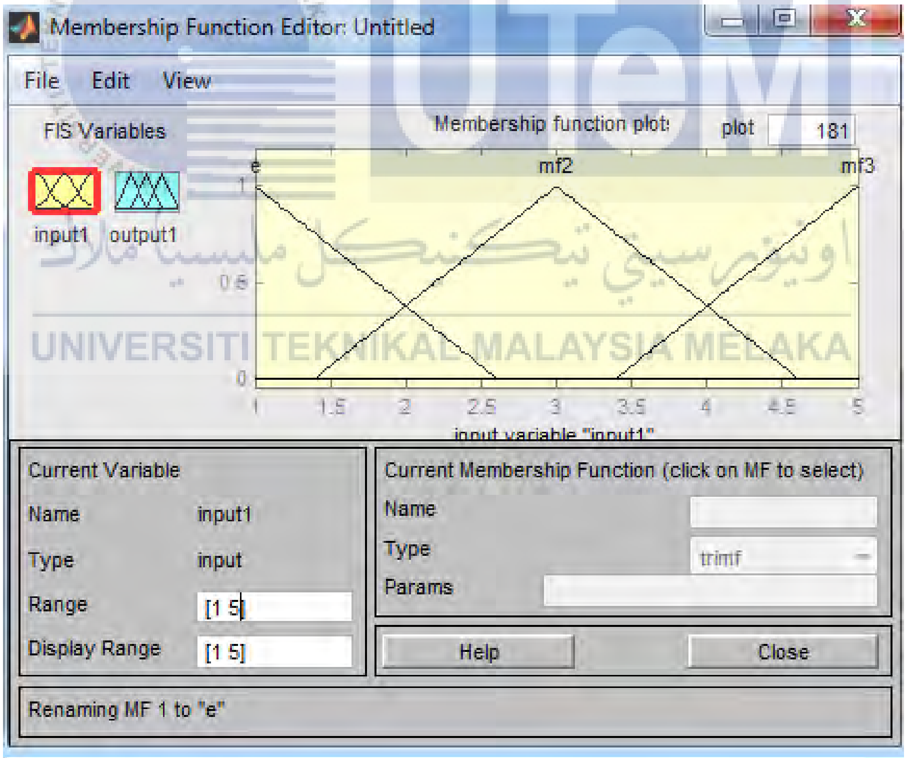


Figure 3.18: Matlab/Simulink block diagram

A 3x3 membership function is chosen and the rule based obtained from journal. FIS Editor was used to insert input and output membership function as shown in Figure 3.19. Rule editor is use to insert a rule based of membership function as shown in Figure 3.20. After saving a FIS Editor and export it to Simulink block diagram. Run a simulation to obtain an output response of the control system.



plot 181

Current Variable

Name input1

Type input

Range [1 5]

Display Range [1 5]

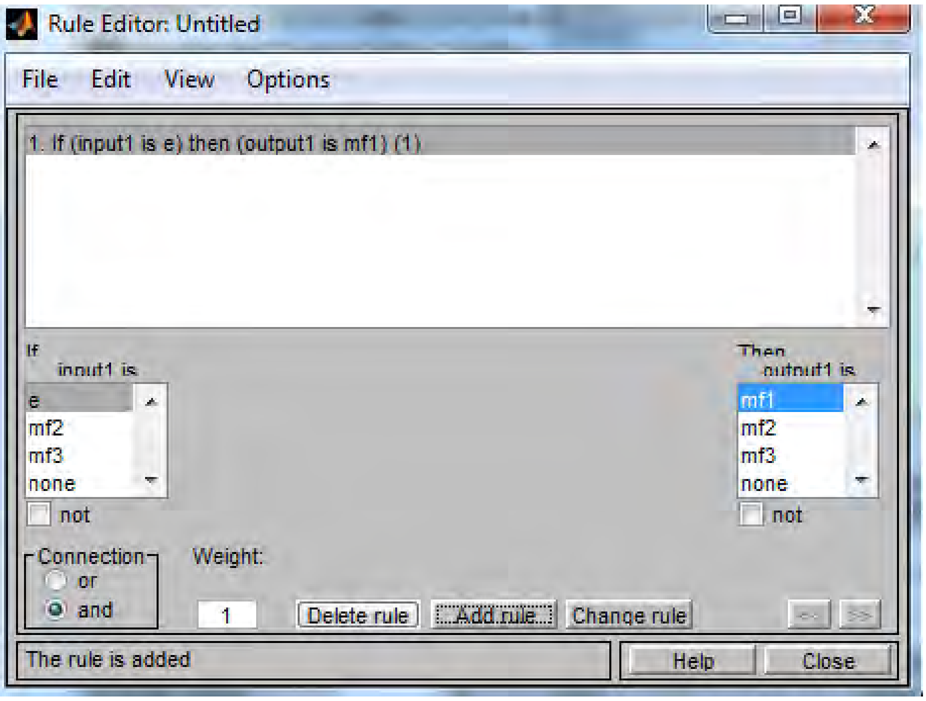
Name

Type trimf

Params

Renaming MF 1 to "e"

Figure 3.19: Membership Function Editor



1. If (input1 is e) then (output1 is mf1) (1)

Weight: 1

The rule is added

Figure 3.20: Rule Editor

3.5.3 Experiment 3: Mathematical modelling

Objectives:

To obtain performance of VideoRay Pro III using mathematical modelling method.

Procedures:

1. Mathematical modelling matrix using Newton-Euler motion equation were simplified based on VideoRay Pro III
2. The matrix equation were substitute with properties and coefficient for VideoRay Pro III data from journal [10]
3. The modelling of ROV was construct as shown in Figure 3.21 and figure 3.22.
4. The related data was import to workspace.
5. Output performance in term of rise time, settling time, overshoot and steady state error were tabulated.

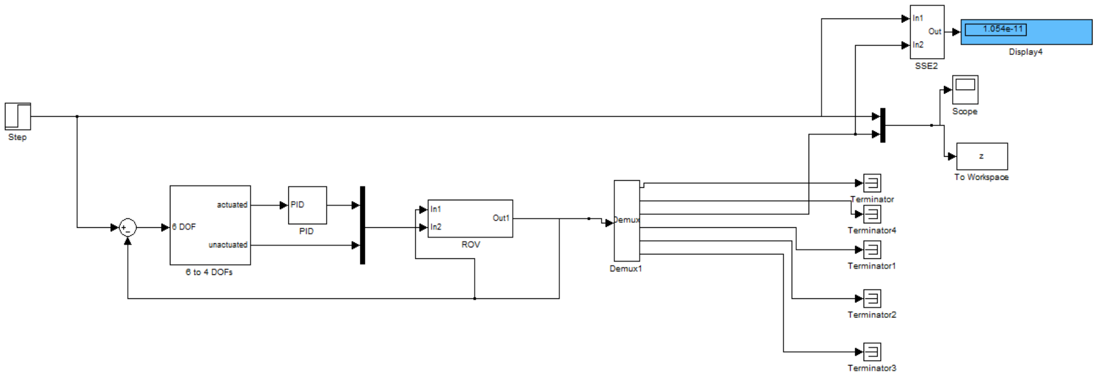


Figure 3.21: Simulation of ROV modelling

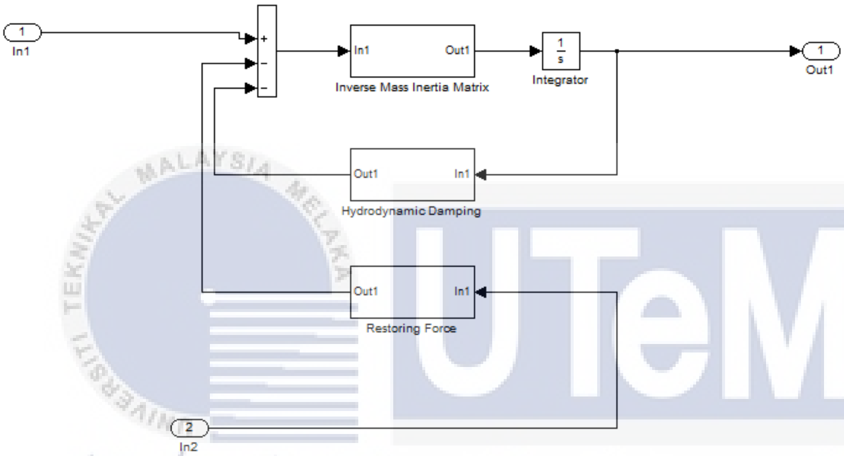


Figure 3.22: Subsystem of ROV

3.5.4 Experiment 4: Pressure sensor

Objectives:

To obtain sensor value data.

Procedures:

1. The experiment was setup to analyze characteristic of MPX5700ap.
2. The output voltage of pressure sensor was determined.
3. The data used to obtain linear equation of real data voltage.
4. The real voltage were compared with ideal voltage (datasheet MPX5700ap)

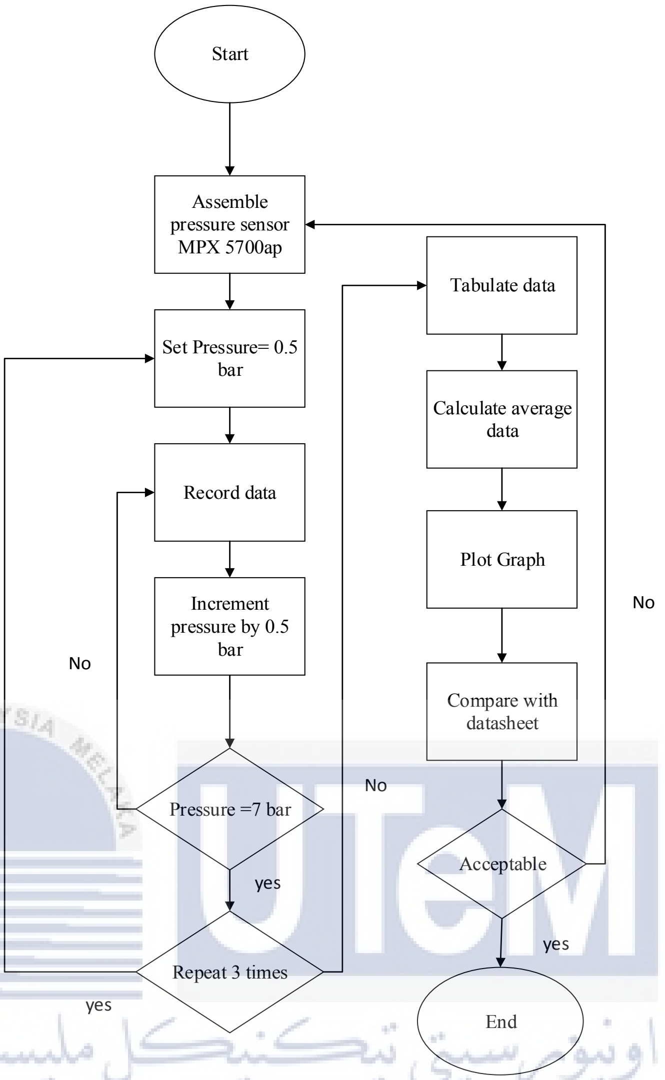


Figure 3.23: Flowchart of pressure sensor experiment

3.5.4.1 Experiment record

The experiment was setup as follows:

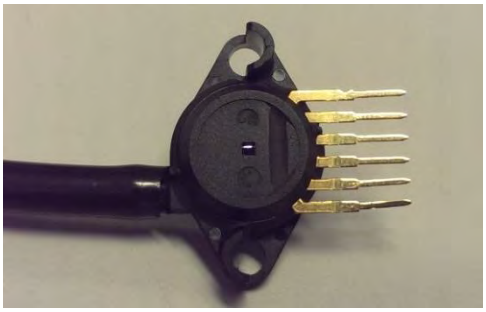


Figure 3.24: Pressure sensor MPX5700ap

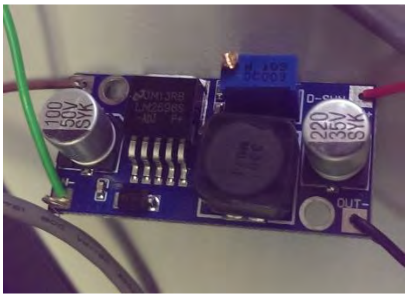


Figure 3.25: Step down voltage regulator

Figure 3.26: Multimeter

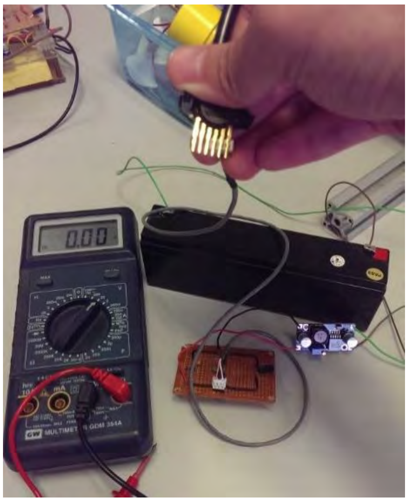


Figure 3.27: Experiment setup to obtained data

Pressure sensor MPX5700ap was used to obtain a real voltage by set a required air pressure (maximum 7 bar). A step down voltage regulator function to step down 9v battery to 5v. Multimeter was used to read a voltage for each air pressure that supply to the pressure sensor. Limitation for pressure sensor MPX5700 is 700KPa (7 bar). The data was recorded for three times. The obtained data was tabulated using Microsoft Excel and the data was analyzed with real voltage versus pressure.

3.5.5 Experiment 5: Effect a real-time simulation system using Microbox 2000/2000C with prototype

Objective: To study the real-time data using Microbox 2000/2000C.

Procedure:

1. The experiment was designed to obtained a real time data by connect prototype (based on VideoRay Pro III) with Microbox 2000/2000C.
2. The desire depth will be set to microbox 2000/2000c and the mini air pump was start to provide the pressure sensor pressure.
3. The MATLAB Simulation was started to collect data using 'out' block by giving appropriate command using the MATLAB command to Microbox 2000/2000c.

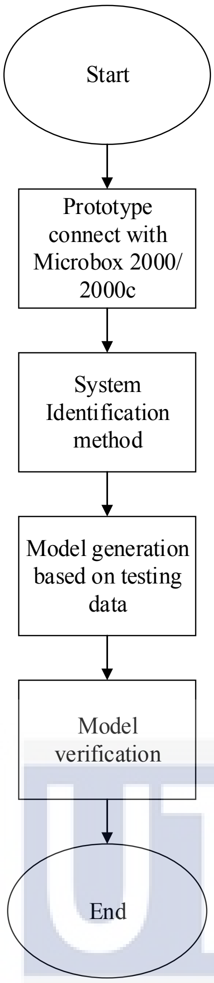


Figure 3.28: Real-time simulation using Microbox 2000/2000C.

After the experiment, the several data was analyzed by system identification. The best data was chosen to use as a model to study the performance of output response using PID and fuzzy logic controller.

3.5.5.1 Experiment record

The experiment was setup in CIA lab. The several equipment was needed to perform an experiment such as Microbox 2000/2000c, prototype (based on VideoRay Pro III), pressure sensor circuit, mini compressor, and multimeter.

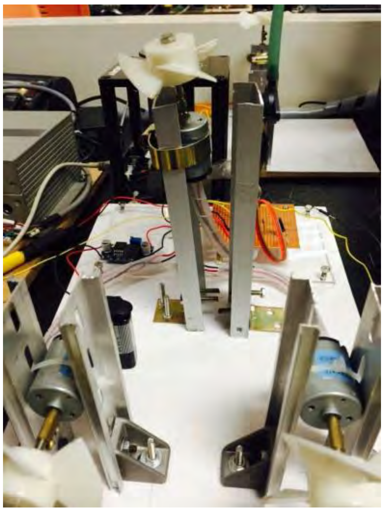


Figure 3.29: Prototype (based on VideoRay Pro III)

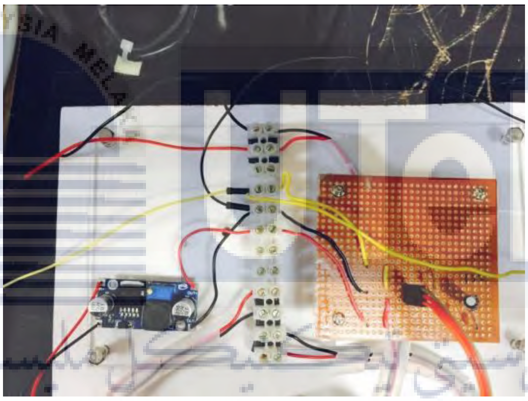


Figure 3.30: Pressure sensor circuit

Figure 3.31: Mini compressor

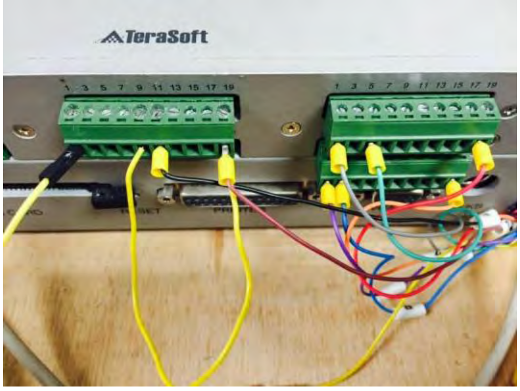


Figure 3.32: Wire connection at Microbox 2000/2000c

Table 3.1: Connection with Microbox 2000/2000C, sensor and thruster

Component		Connector 2	Connector 3
Pressure sensor	Analog to Digital	1	
	AD ground	9	
Thruster	Digital to Analog	11	
	Ground	19	19

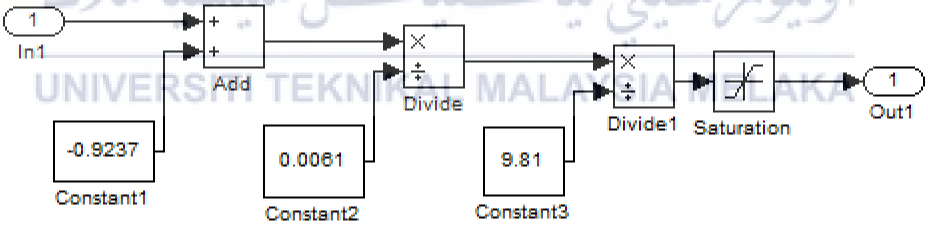


Figure 3.33: Analog to digital converter system

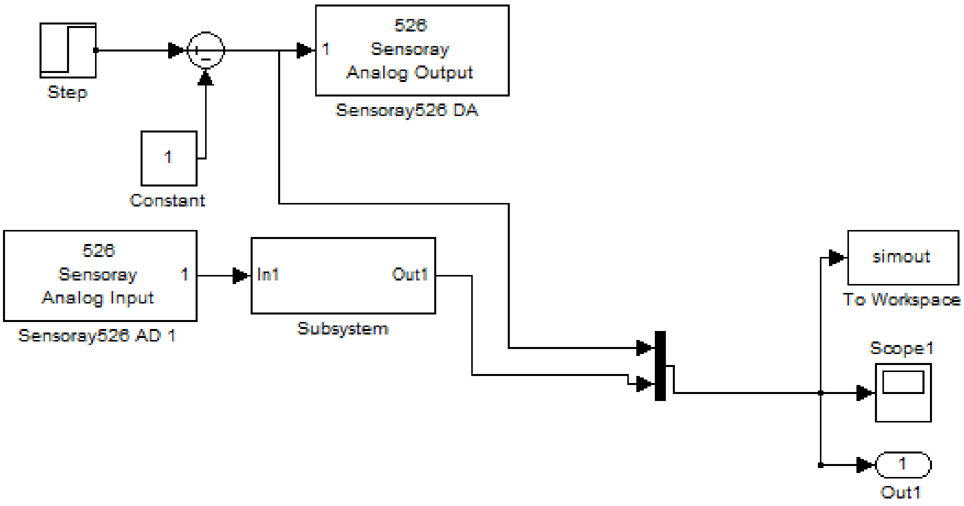


Figure 3.34: Real time open loop system

3.5.6 Experiment 6: System identification selection

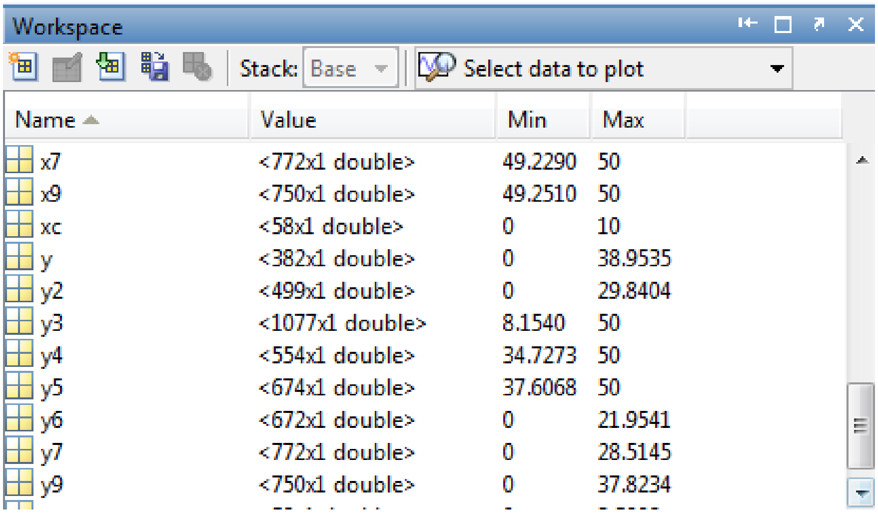
Objective: To verify percentage and select a best fit real time data

Procedure:

1. The collected real time data that obtained from prototype and Microbox 2000/2000C were analyzed by system identification.

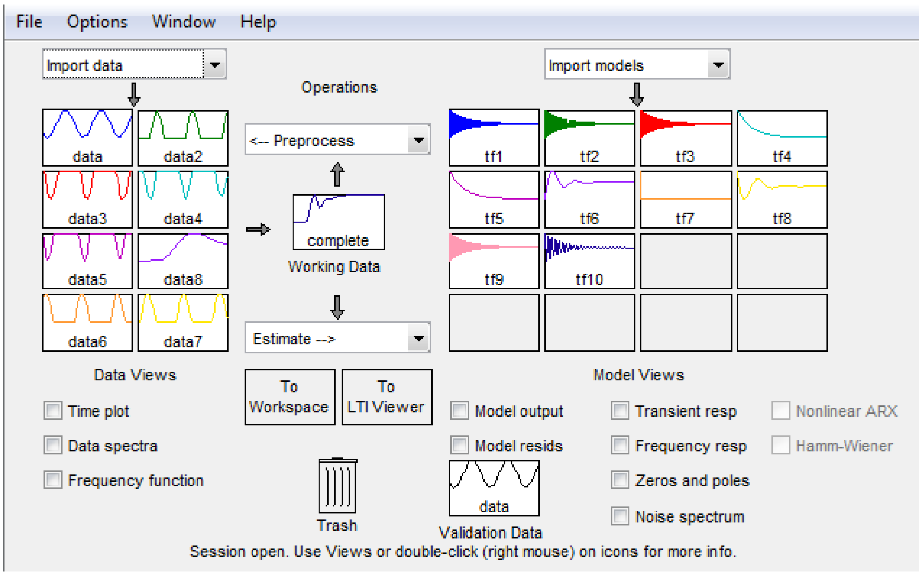
3.5.6.1 Experiment record

The data from real time simulation were obtained from Matlab command. Each data were analyzed by using system identification as Figure 3.36.



Name	Value	Min	Max
x7	<772x1 double>	49.2290	50
x9	<750x1 double>	49.2510	50
xc	<58x1 double>	0	10
y	<382x1 double>	0	38.9535
y2	<499x1 double>	0	29.8404
y3	<1077x1 double>	8.1540	50
y4	<554x1 double>	34.7273	50
y5	<674x1 double>	37.6068	50
y6	<672x1 double>	0	21.9541
y7	<772x1 double>	0	28.5145
y9	<750x1 double>	0	37.8234

Figure 3.35: Real time data simulation



Session open. Use Views or double-click (right mouse) on icons for more info.

Figure 3.36: Data in system identification

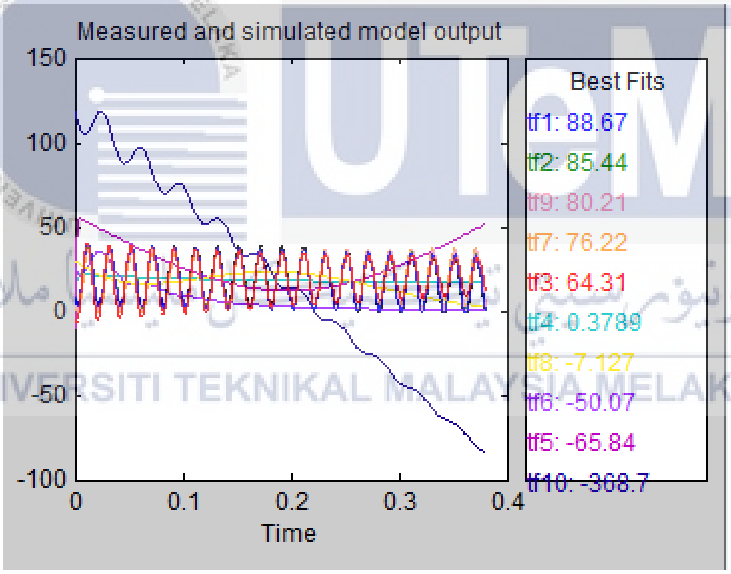


Figure 3.37: Best fits of data

3.5.7 Experiment 7: Stability of real time data

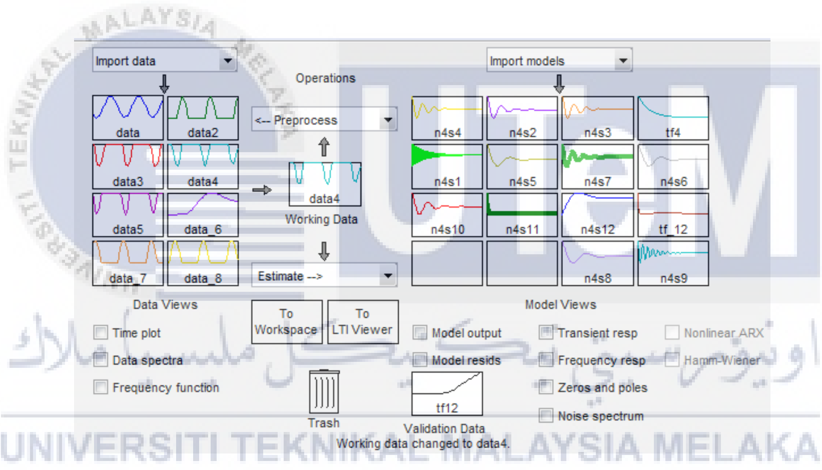
Objective: To test a controllability, observability and asymptotic stability of real time data

Procedure:

1. Transfer function of real time data were changed to state space equation
2. Matrix value were obtained from state space equation
3. Each real time data were tested by using coding at Matlab command windows

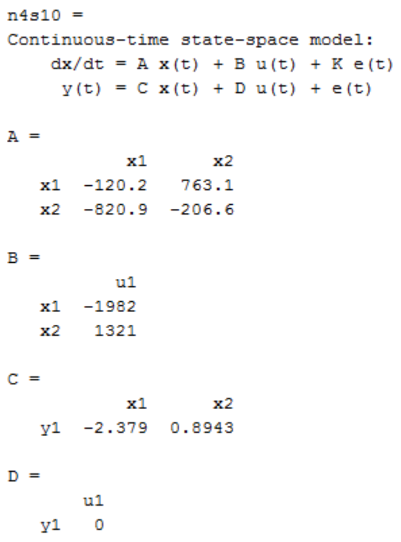
3.5.7.1 Experiment record

The real time data were changed to state space equation as shown in Figure 3.38. The result of matrix equation were obtained as shown in Figure 3.39. A real time data already being tested by using coding of 2x2 matrix as shown in Figure 3.40. The tested data were tabulated in a table.



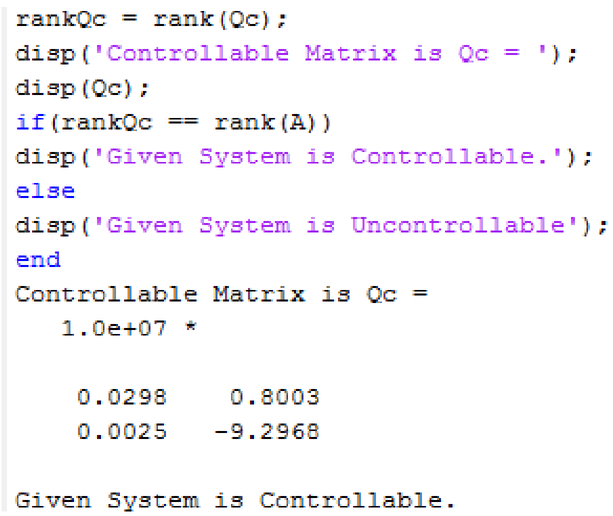
Working data changed to data4.

Figure 3.38: System identification



```
n4s10 =
Continuous-time state-space model:
    dx/dt = A x(t) + B u(t) + K e(t)
     y(t) = C x(t) + D u(t) + e(t)

A =
           x1      x2
   x1  -120.2   763.1
   x2  -820.9  -206.6

B =
          u1
   x1  -1982
   x2   1321

C =
           x1      x2
   y1  -2.379  0.8943

D =
       u1
   y1   0
```

Figure 3.39: State space matrix equation



```
rankQc = rank(Qc);
disp('Controllable Matrix is Qc = ');
disp(Qc);
if(rankQc == rank(A))
disp('Given System is Controllable.');
else
disp('Given System is Uncontrollable');
end
Controllable Matrix is Qc =
   1.0e+07 *

    0.0298    0.8003
    0.0025   -9.2968

Given System is Controllable.
```

Figure 3.40: Test stability of data

3.5.8 Experiment 8: Effect of membership function of real-time Fuzzy Logic Controller

Objective: To study effect of membership function of real-time fuzzy logic controller

Procedure:

1. The simulation of fuzzy logic controller will be construct using Matlab as shown in Figure 3.41
2. The rule of membership function were set as shown in table 3.2.
3. The desire depth was set to 5m
4. The Matlab simulation was start collect a data from scope data by giving appropriate command using Matlab command window.
5. Graph of step response and data such as rise time, percent overshoot, settling time, and steady state error were obtained using matlab command window.
6. Step 3 and 4 were repeated with different zero membership function adjustment. The data of rise time, ssttling time, percent overshoot and steady state error of the fuzzy logic were recorded in a table.
7. The result were compared with modelling simulation, real time simulation and PID.

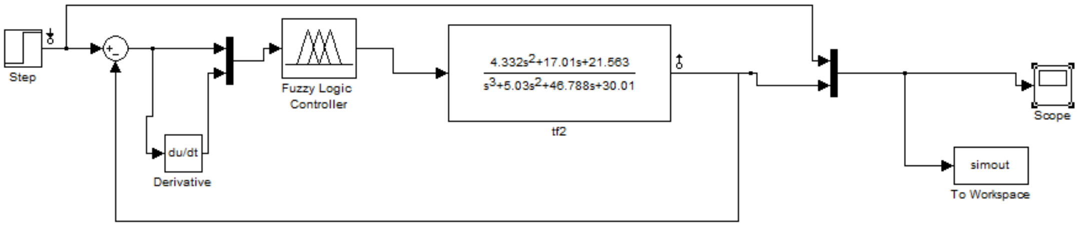


Figure 3.41: Simulation real-time of fuzzy logic controller

Table 3.2: Rule table for fuzzy logic

R W	Z	N	P
P	P	P	P
N	N	N	N
Z	Z	Z	Z

Legend:

R= Input 1

W= Input 2

Z= Zero

N= Negative

P= Positive

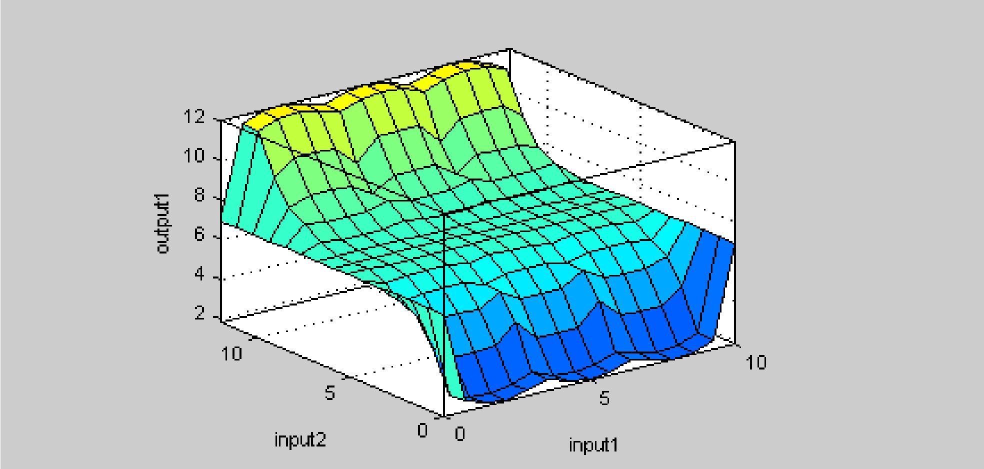


Figure 3.42: Rule viewer surface of fuzzy logic controller

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

This section will shows, tabulate, and analyze data being collected during the experiments

4.2 Experiments implementation

4.2.1 Experiment 1: Use Simulink software to get an output response of PID

Condition 1: Ki value changed

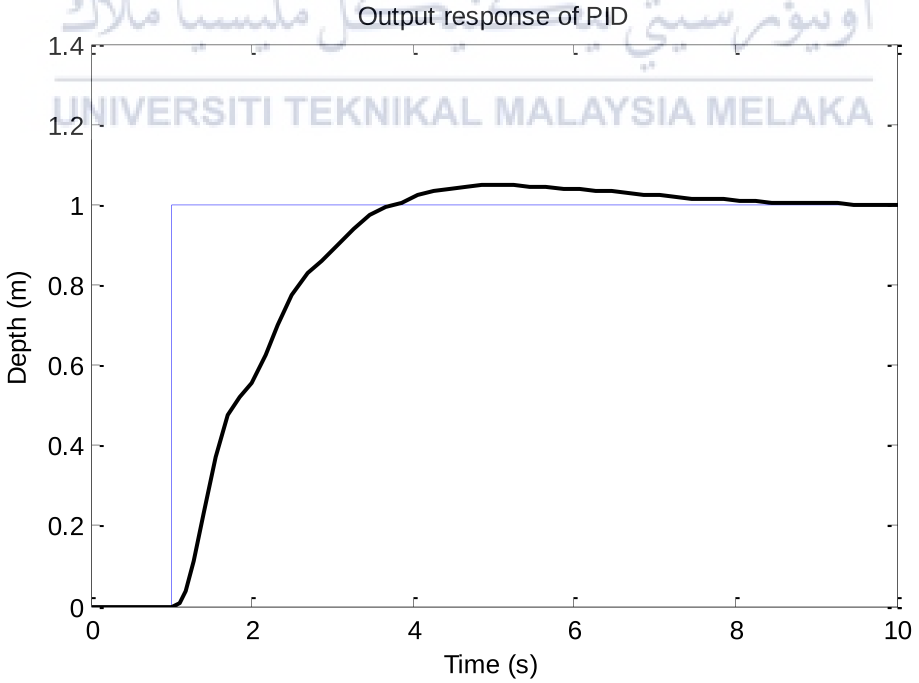


Figure 4.1: Graph depth vs. time for i= 1.5

i. Kp = 0

ii. Ki = 1.5

iii. Kd = 0

iv. Rise time = 1.80s

v. Overshoot = 4.84 %

vi. Steady state error = 1

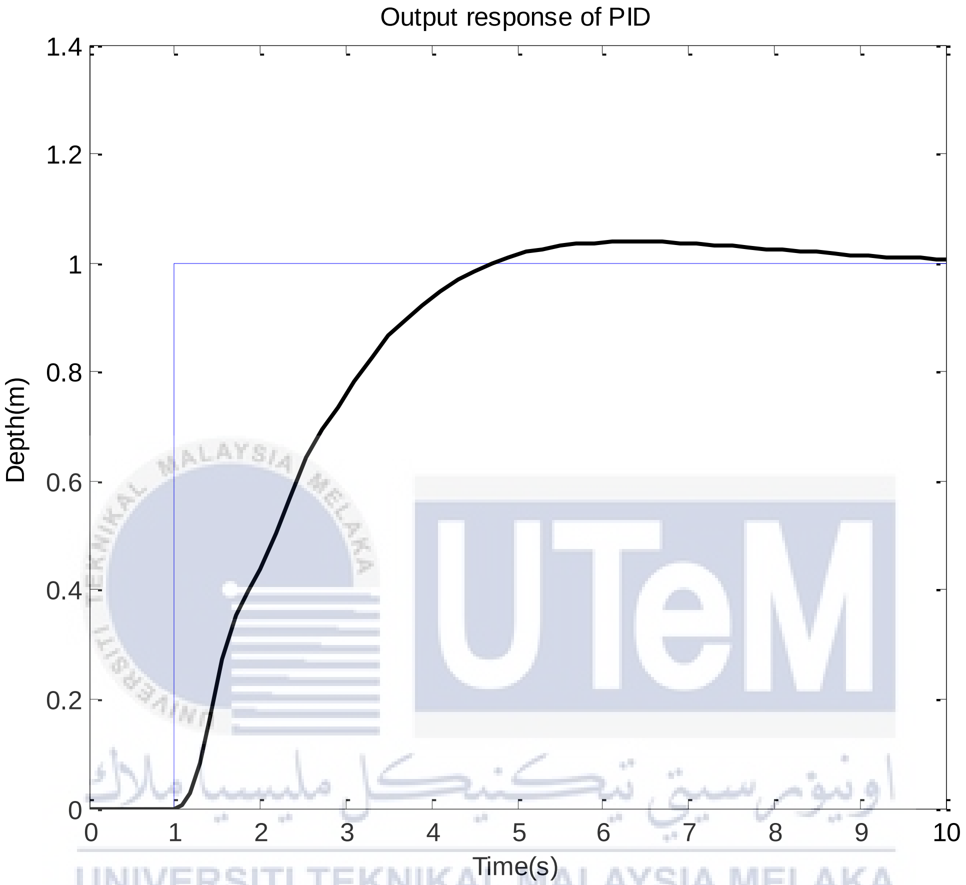


Figure 4.2: Graph depth vs. time for i= 1.07

i. Kp = 0

ii. Ki = 1.07

iii. Kd = 0

iv. Rise time = 2.4s

v. Overshoot = 3.89 %

vi. Steady state error = 1

Condition 2: Ki value set at 1; Kp value changed

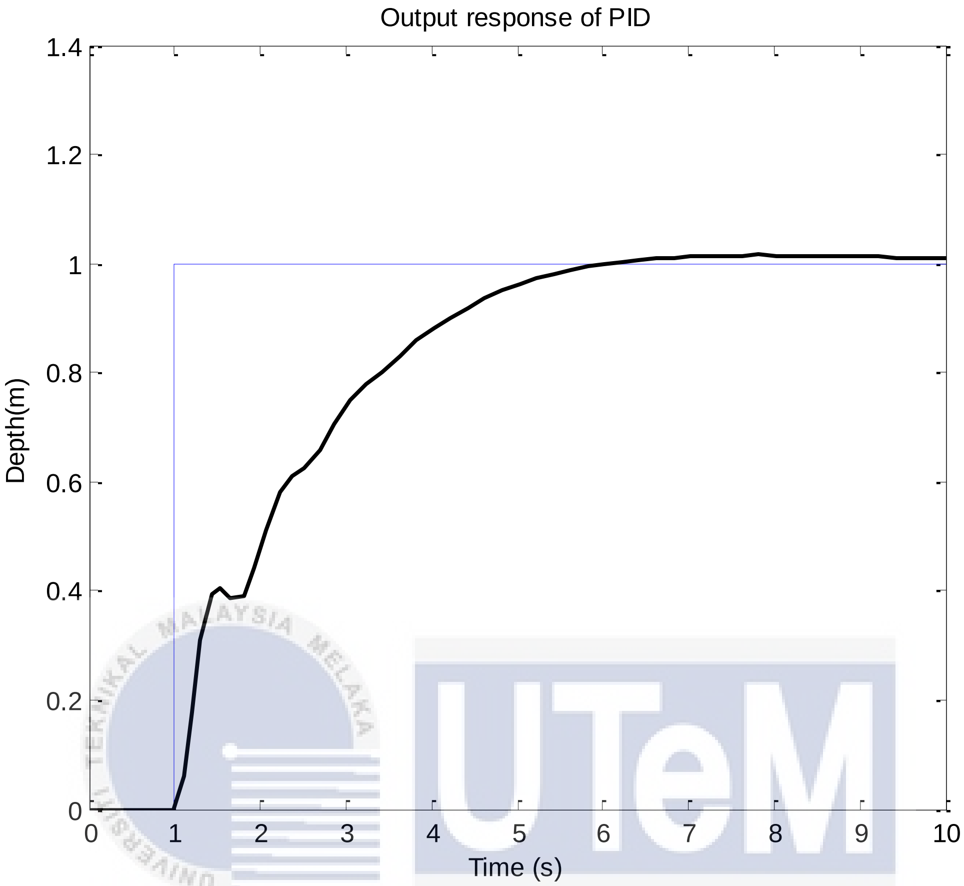


Figure 4.3: Graph depth vs. time for p= 0.4

i. Kp = 0.4
ii. Ki = 1
iii. Kd = 0
iv. Rise time = 2.64s
v. Overshoot = 3.72 %
vi. Steady state error = 1

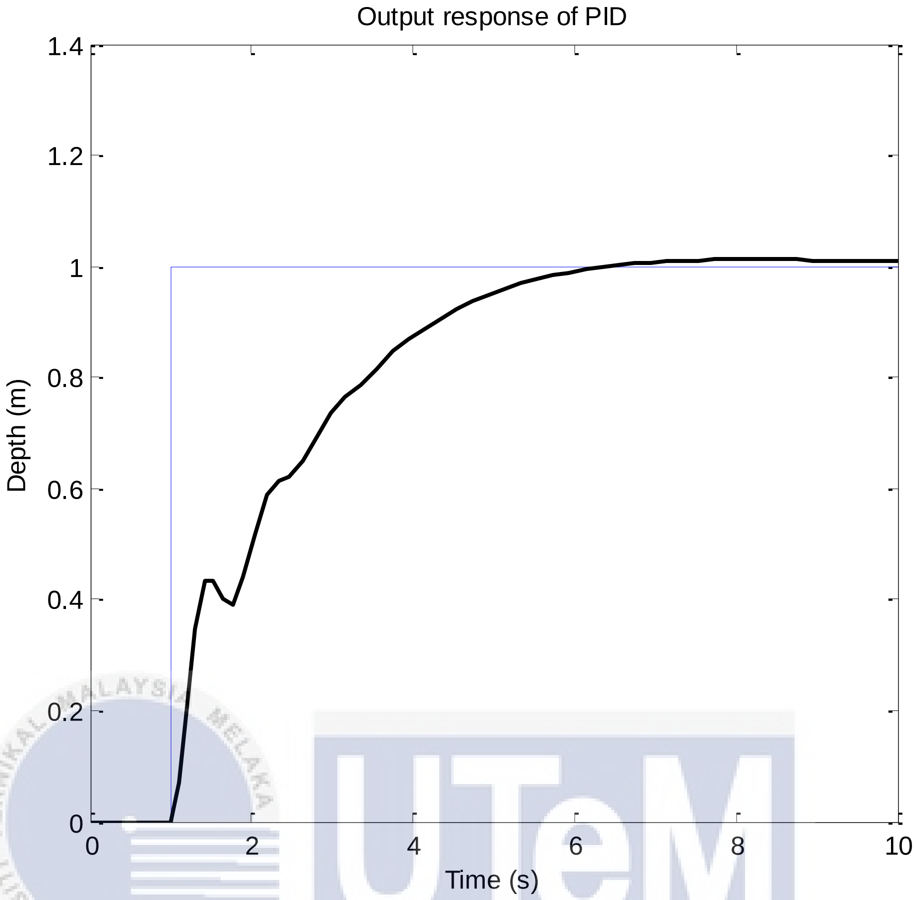


Figure 4.4: Graph depth vs. time for p=0.5

i. Kp = 0.5

ii. Ki = 1

iii. Kd = 0

iv. Rise time = 3.13s

v. Overshoot = 1.6 %

vi. Steady state error = 1

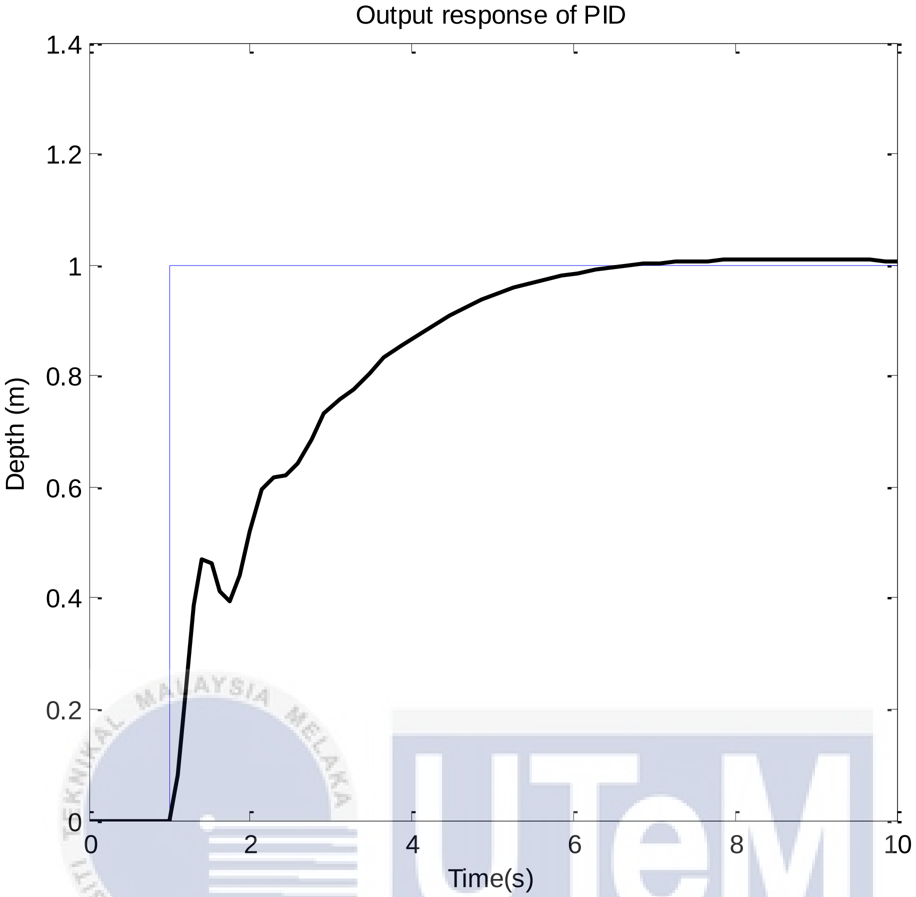


Figure 4.5: Graph depth vs. time for p= 0.6

i.	Kp = 0.6	ii.	Ki = 1
iii.	Kd = 0	iv.	Rise time = 3.21s
v.	Overshoot = 1.25 %	vi.	Steady state error = 1

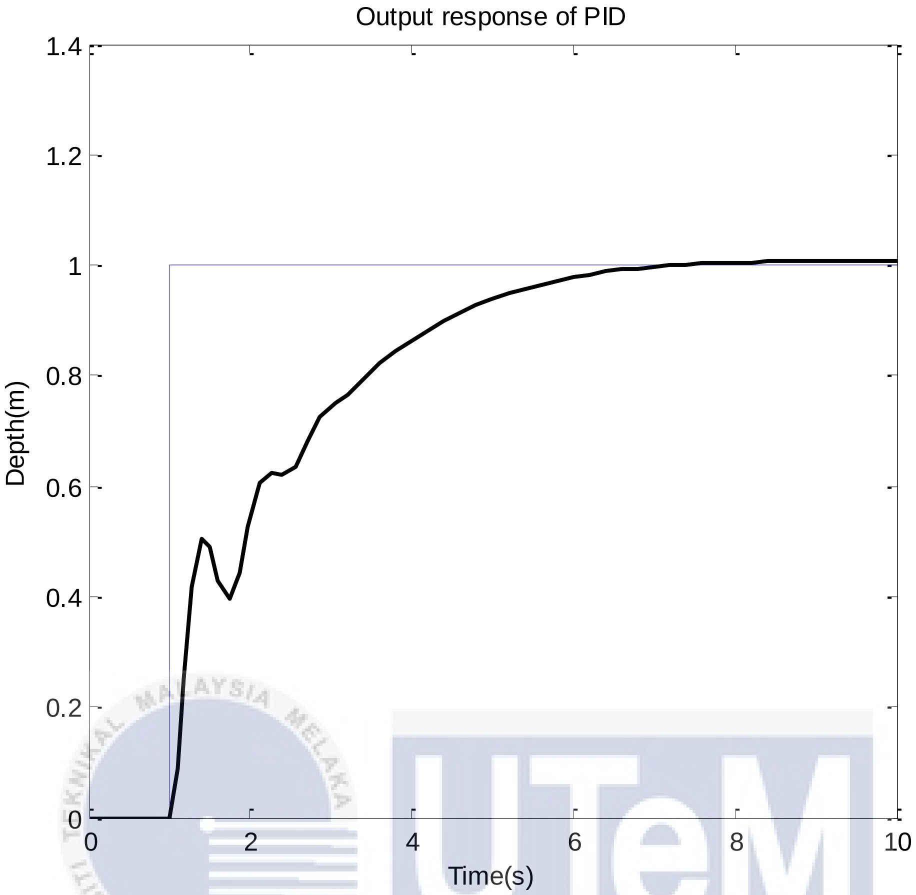


Figure 4.6: Graph depth vs. time for p= 0.7

i. Kp = 0.7

ii. Ki = 1

iii. Kd = 0

iv. Rise time = 3.28s

v. Overshoot = 0.96 %

vi. Steady state error = 1

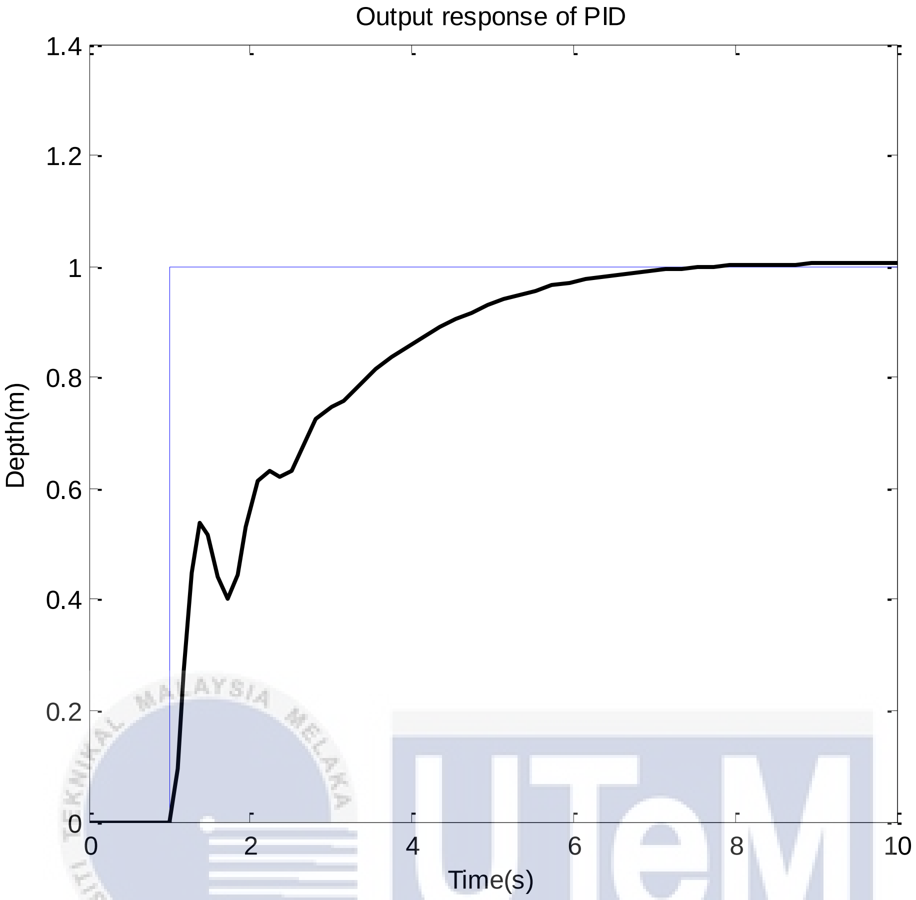


Figure 4.7: Graph depth vs. time for p= 0.8

i. Kp = 0.8
ii. Ki = 1
iii. Kd = 0
iv. Rise time = 3.42s
v. Overshoot = 0.519 %
vi. Steady state error = 1

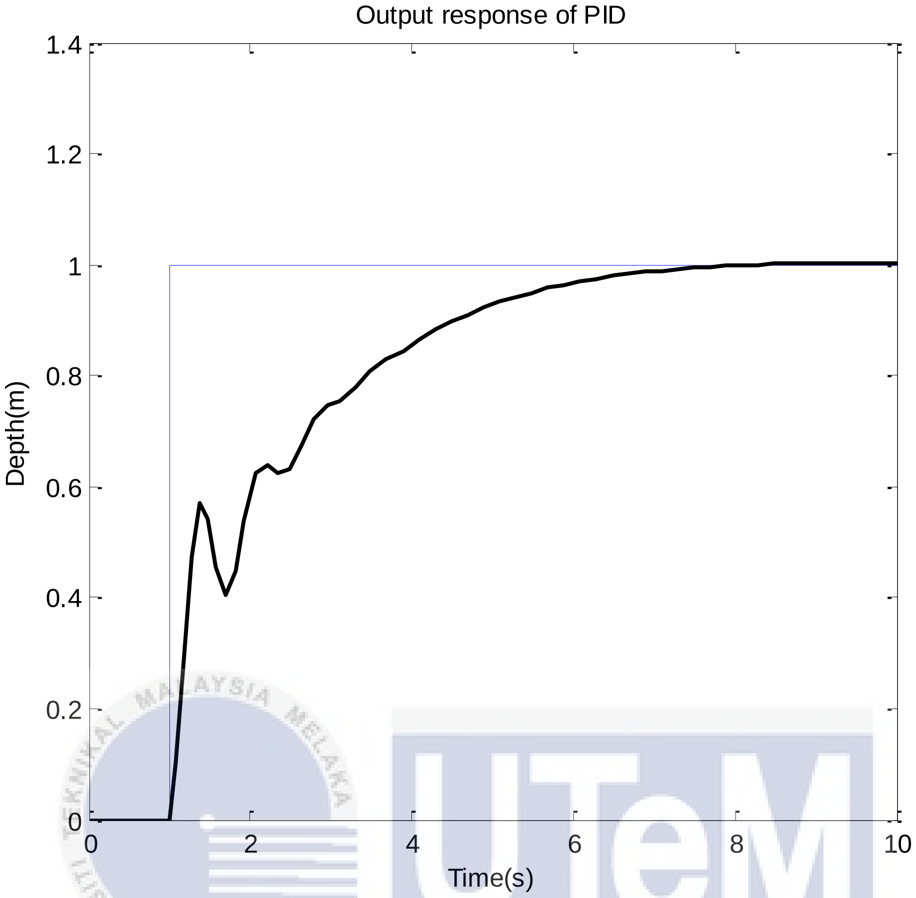


Figure 4.8: Graph depth vs. time for p= 0.9

i.	Kp = 0.9	ii.	Ki = 1
iii.	Kd = 0	iv.	Rise time = 3.49s
v.	Overshoot = 0.36 %	vi.	Steady state error = 1

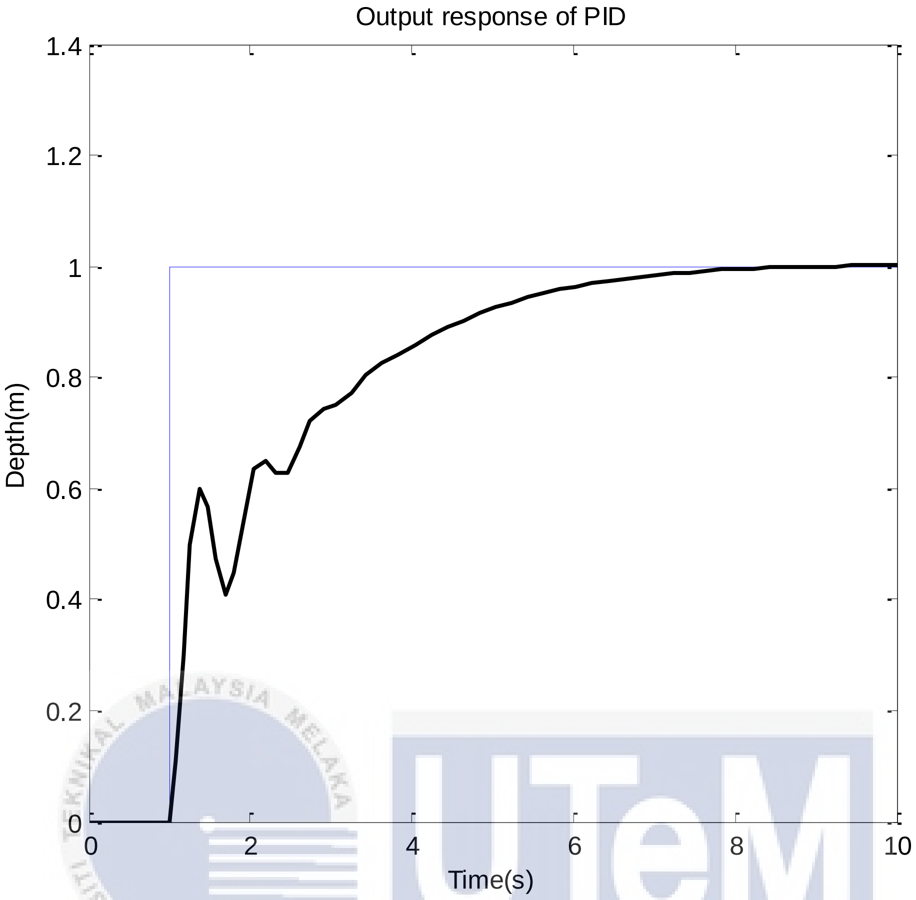


Figure 4.9: Graph depth vs. time for p= 1

i.	Kp = 1	ii.	Ki = 1
iii.	Kd = 0	iv.	Rise time = 3.56s
v.	Overshoot = 0.236 %	vi.	Steady state error = 1

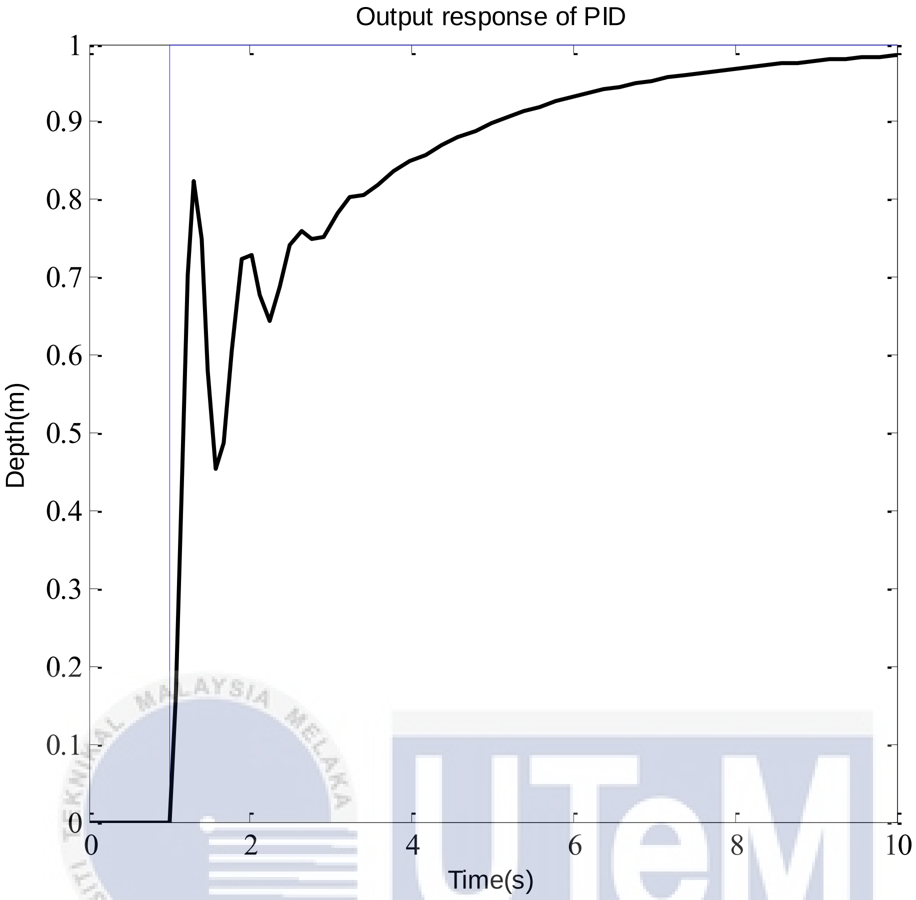


Figure 4.10: Graph depth vs. time for p= 2

i. Kp = 2
ii. Ki = 1
iii. Kd = 0
iv. Rise time = 4s
v. Overshoot = 0 %
vi. Steady state error = 1

Table 4.1: Output response of PID with different parameters

PID parameter	Rise time (s)	Overshoot (%)	Steady state error
Kp=0 Ki=1.5 Kd=0	1.80	4.84	1
Kp=0 Ki=1.07 Kd=0	2.4	3.89	1
Kp=0.4 Ki=1 Kd=0	3.72	2.64	1
Kp=0.5 Ki=1 Kd=0	3.13	1.6	1
Kp=0.6 Ki=1 Kd=0	3.21	1.25	1
Kp=0.7 Ki=1 Kd=0	3.28	0.96	1
Kp=0 Ki=1.5 Kd=0	3.35	0.718	1
Kp=0.8 Ki=1 Kd=0	3.42	0.519	1
Kp=0.9 Ki=1 Kd=0	3.49	0.36	1
Kp=1 Ki=1 Kd=0	3.56	0.236	1

Kp=2 Ki=1 Kd=0	0	4	1

Table 4.1 shows an output response for different value of Kp, Ki and Kd. In order to analyze the performance of PID, the value of PID parameter is changed to get no overshoot, faster rise time and small steady state error. However, when zero overshoot condition is achieved, a value of rise time is increased and steady state error value maintains at 1. The objective to achieve the three conditions of control system is not accessible. So, the fuzzy logic control method is introduced to overcome this problem.

4.2.2 Experiment 2: Fuzzy logic controller using Matlab Simulink

The naming of shifting as follows:-

Center = Original zero membership function

Left = Zero membership function shifted to left

Right = Zero membership function shifted to right

Condition 1: Output shifting to "center"

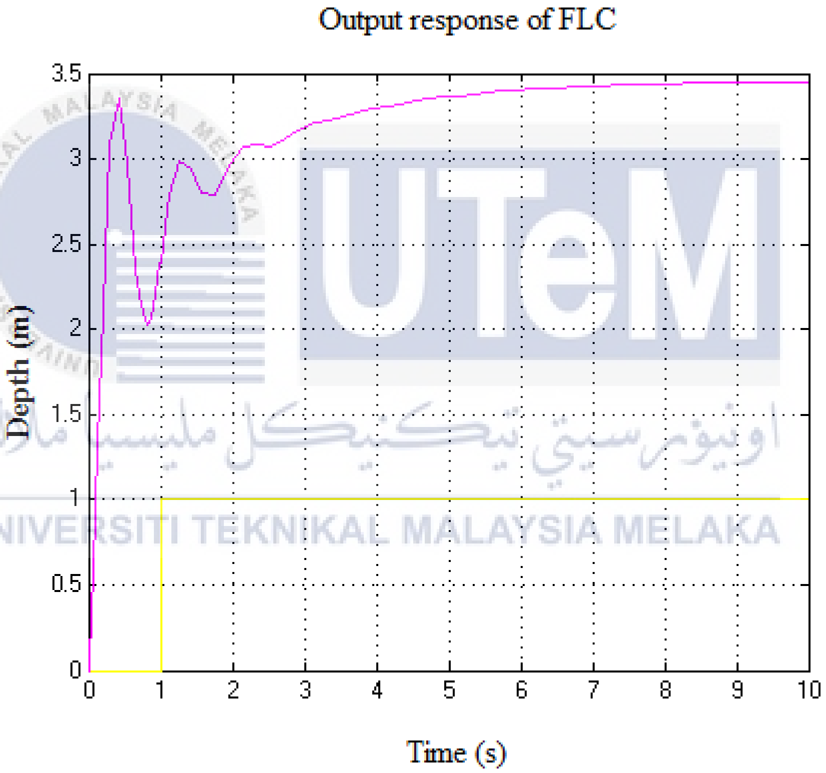


Figure 4.11: Graph depth vs. time for output "center"

i. Rise time = 3.36s

ii. Overshoot = 0 %

iii. Steady state error = -0.443

Condition 2: Output shifting to “right”

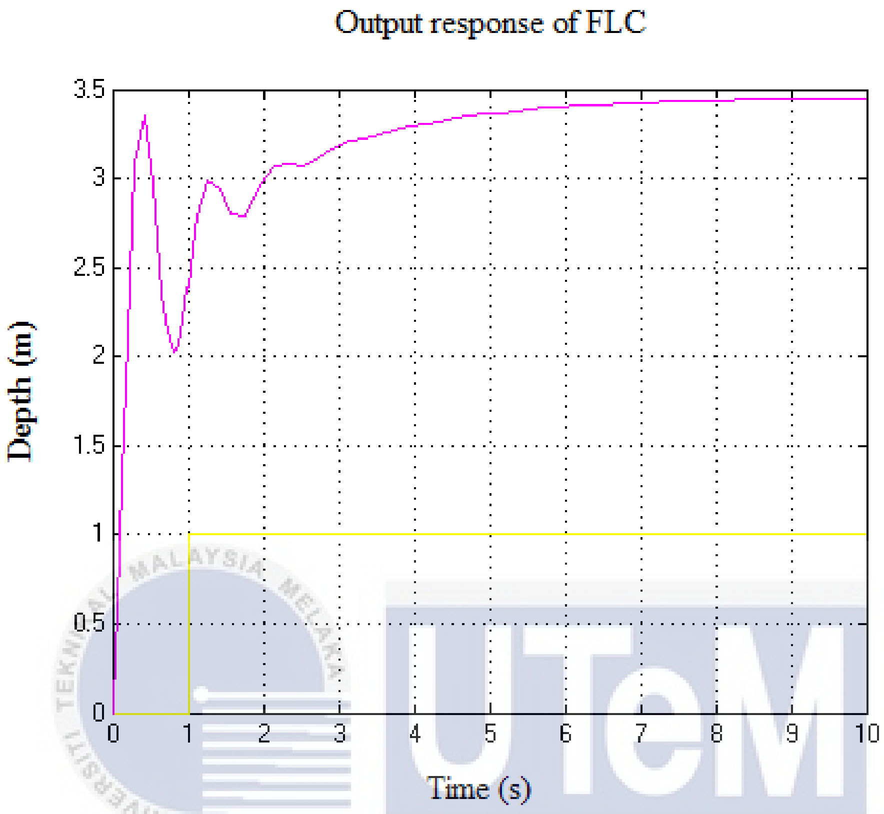


Figure 4.12: Graph depth vs. time for output “right”

i. Rise time = 3.45s

ii. Overshoot = 0 %

iii. Steady state error = -0.479

Condition 3: Output shifting to “left”

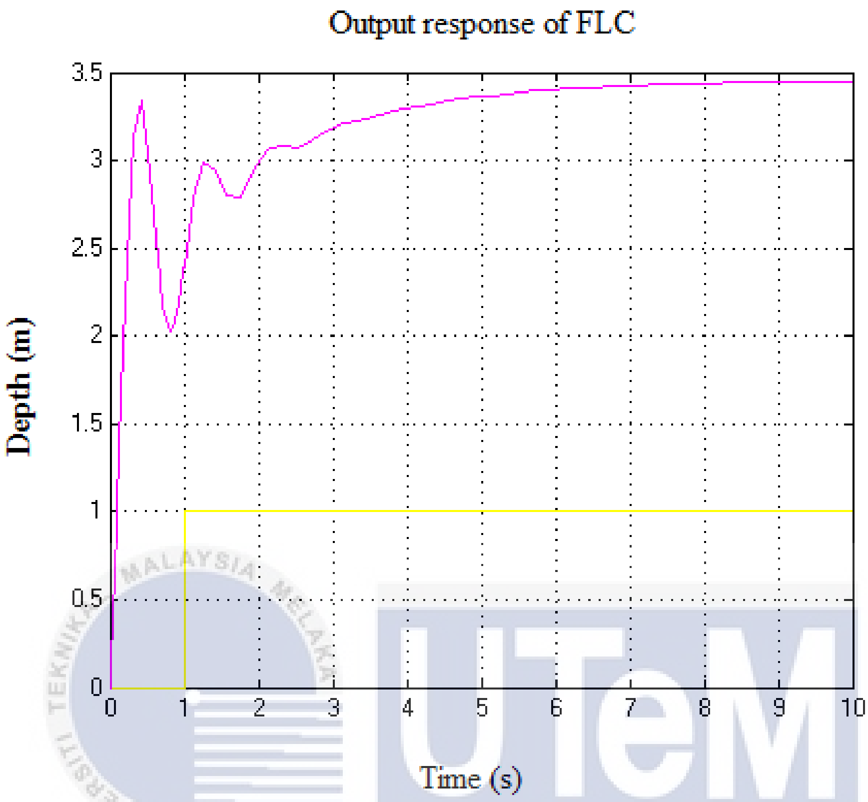


Figure 4.13: Graph depth vs. time for output “left”

i. Rise time = 3.31s

ii. Overshoot = 0 %

iii. Steady state error = -0.417

Condition 4: Input 1 shifting to “center”

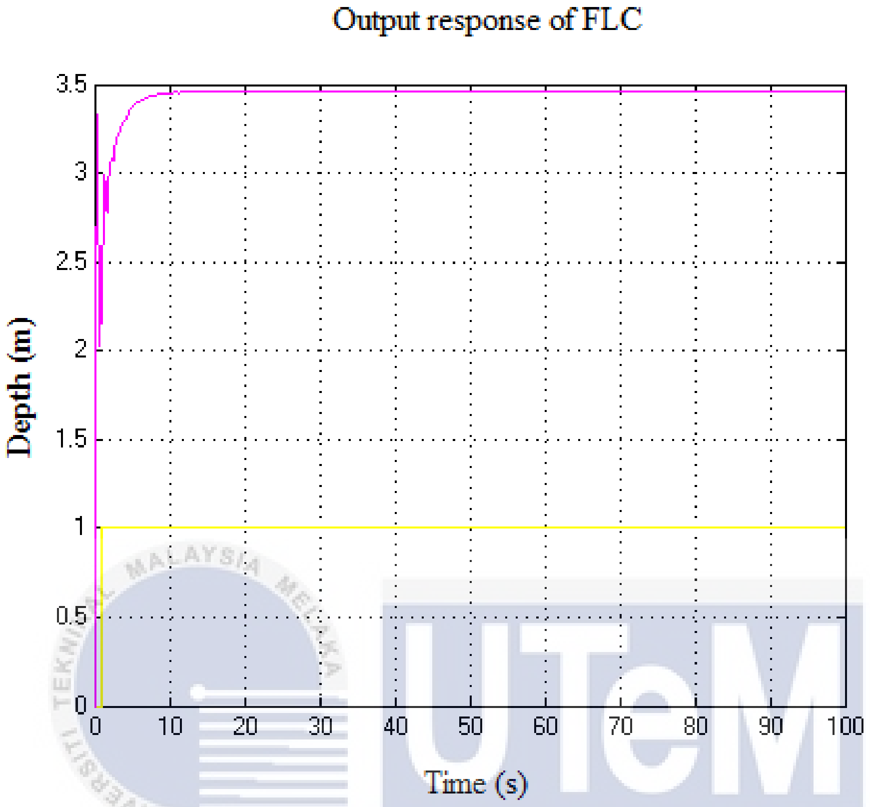


Figure 4.14: Graph depth vs. time for input 1 “center”

i. Rise time = 3.28s

ii. Overshoot = 0 %

iii. Steady state error = -0.41

Condition 5: Input 1 shifting to "right"

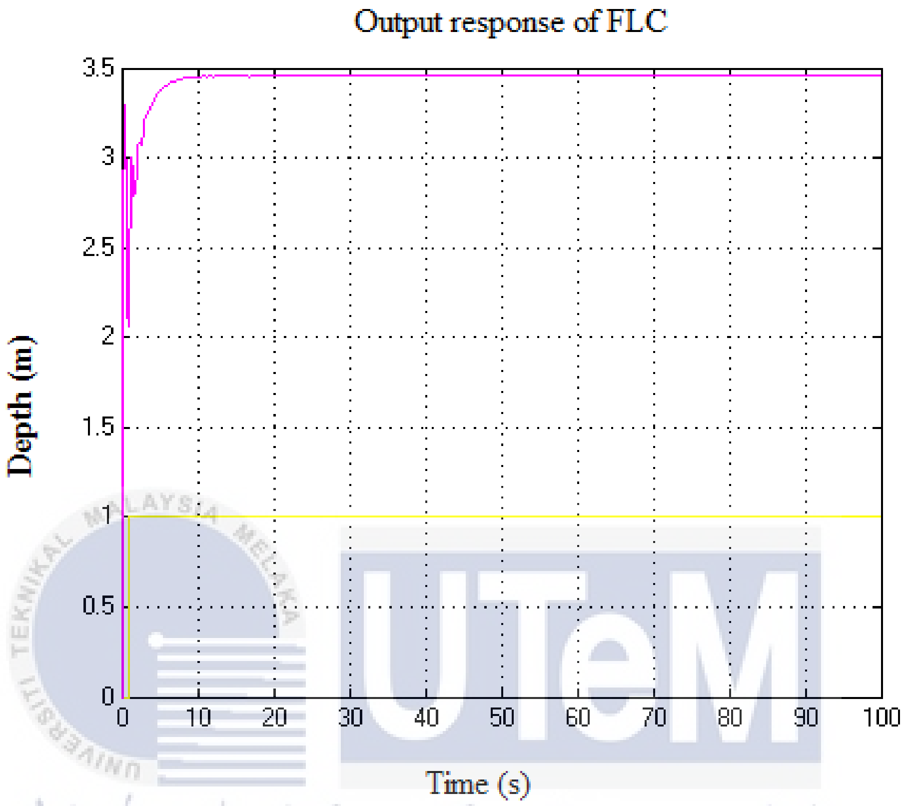


Figure 4.15: Graph depth vs. time for input 1 "right"

i. Rise time = 3.30s

ii. Overshoot = 0 %

iii. Steady state error = -0.43

Condition 6: Input 1 shifting to “left”

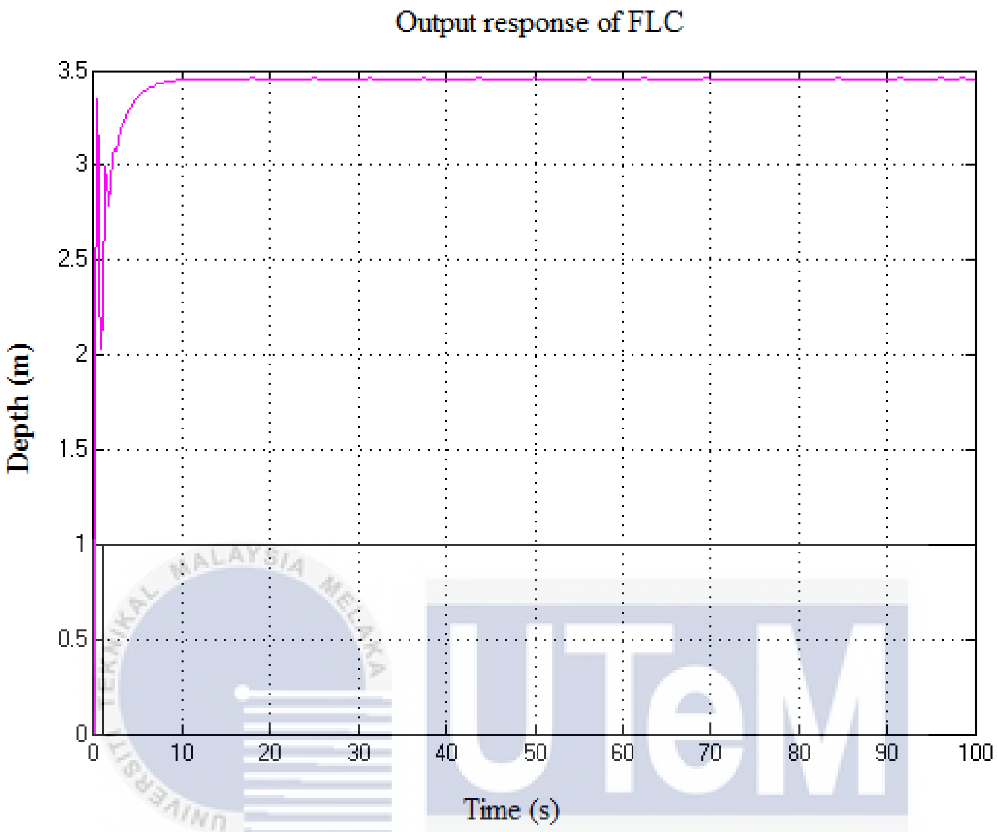


Figure 4.16: Graph depth vs. time for input 1 “left”

i. Rise time = 3.25s

ii. Overshoot = 0 %

iii. Steady state error = -0.40

Table 4.2: Summary output response of shifting membership function

Membership function	Rise time (s)	Overshoot (%)	Steady state error
Output “center”	3.36	0	-0.443
Output “right”	3.45	0	-0.479
Output “left”	3.31	0	-0.417
Input 1 “center”	3.28	0	-0.41
Input 1 “right”	3.30	0	-0.43
Input 1 “left”	3.25	0	-0.40

Table 4.3: Comparison system response between PID and FLC

Type of controller	Rise time (s)	Diff. rise time (s)	Overshoot (%)	Steady state error	Diff. steady state error
PID	4	NA	0	1	NA
FLC	3.25	0.75	0	-0.40	0.60

Based on a table 4.2, a shifting membership function method in the fuzzy logic controller is used. The input and output membership function will be shifted to the left, right or center. An output response for each shifting will be analyzed using Matlab/Simulink. The result of system response in terms of no overshoot, faster rise time and small steady state error shows a better value than PID result as shown in Table 4.3. The rise time of FLC 0.75s faster than PID and steady state error improved with 0.60 difference. The zero overshoot condition is achieved. The result of rise time and steady state error also improved.

4.2.3 Experiment 3: Mathematical modelling

The mathematical modelling was derived as shown below. The value of matrix based on properties and coefficient of VideoRay Pro III [10]. The mass, m= 43kg follow a mass of VideoRay Pro 3s. The value of -16.24 implies that the vehicle has residual buoyancy. The residual buoyancy equates to 4% of the vehicle's weight.

$$MRB= \begin{vmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & Ix & 0 & 0 \\ 0 & 0 & 0 & 0 & Iy & 0 \\ 0 & 0 & 0 & 0 & 0 & Iz \end{vmatrix} ; \tag{4.1}$$

$$MRB= \begin{vmatrix} 43 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 43 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.02532 \end{vmatrix} \tag{4.2}$$

$$MA= \begin{vmatrix} X\grave{u} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Z\hat{w} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & N\check{r} \end{vmatrix} \tag{4.3}$$

$$MA = \begin{vmatrix} 1.9404 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3.9482 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0321 \end{vmatrix} \tag{4.4}$$

$$D(v) = \begin{vmatrix} 10.586 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 26.3392 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0137 \end{vmatrix} \tag{4.5}$$

$$G = \begin{vmatrix} 0 \\ 0 \\ -16.24 \\ 0 \\ 0 \\ 0 \end{vmatrix} \tag{4.6}$$

The table 4.4 shown the output performance in term of rise time, settling time, overshoot and steady state error for mathematical modelling approach. The result show no overshoot, faster rise time and small steady state error achieved. The output response of mathematical modelling based on VideoRay Pro III as shown in Figure 4.17

Table 4.4: Output response of mathematical modelling

Type of control system	Tr (s)	Ts (s)	Overshoot (%)	Steady state error
Mathematical modelling (PID)	2.1407	5.5639	0	0

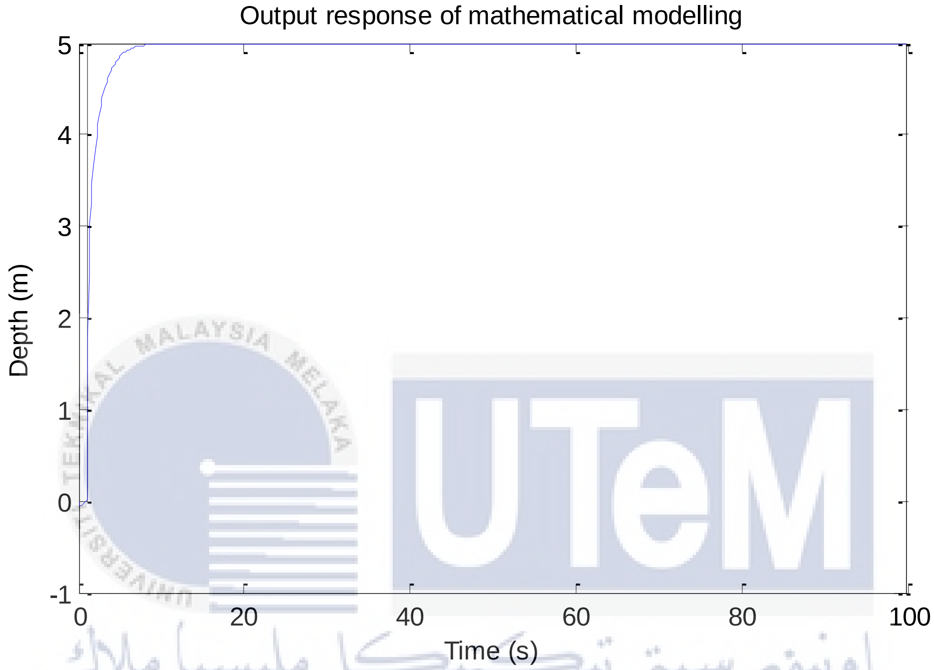


Figure 4.17: Graph depth vs. time for mathematical modelling

4.2.4 Experiment 4: Pressure sensor

The depth was converted using formula as below:

$$P = \rho g h \tag{4.7}$$

$$P = \rho g h$$

$$h = \frac{P}{\rho g} = \frac{P}{9810} \tag{4.8}$$

Where P = pressure (Pa)

ρ = density of water (1000 kgm^{-3})

g = gravity (9.81ms^{-2})

h = depth (m)

$$V_{average} = \frac{V_1+V_2+V_3}{3} \tag{4.9}$$

The output voltage of the experiment were taken by 3 times. An average voltage was calculated by using formula 4.9. The result of experiment was shown in table 4.5. Graph of ideal voltage that obtained from datasheet was shown in Figure 4.18 and the real voltage show a voltage that obtained in an experiment as shown in Figure 4.19.

Table 4.5: Table of pressure to depth vs. voltage

KPa	**Bar**	**Voltage (v)**				**Depth (m)**
		V1	**V2**	**V3**	**Average**	
0	0	0.84	0.84	0.84	0.84	0.00
50	0.5	1.24	1.25	1.2	1.23	5.10
100	1	1.52	1.53	1.52	1.52	10.20
150	1.5	1.84	1.85	1.85	1.85	15.30
200	2	2.17	2.14	2.16	2.16	20.40
250	2.5	2.44	2.47	2.49	2.47	25.50
300	3	2.7	2.78	2.81	2.76	30.60
350	3.5	3.1	3.05	3.13	3.09	35.70
400	4	3.42	3.42	3.44	3.43	40.80
450	4.5	3.72	3.75	3.76	3.74	45.90
500	5	4.05	4.07	4.07	4.06	51.00
550	5.5	4.36	4.37	4.4	4.38	56.10
600	6	4.66	4.7	4.7	4.69	61.20
650	6.5	4.96	4.95	4.96	4.96	66.30
700	7	4.96	4.96	4.96	4.96	71.40

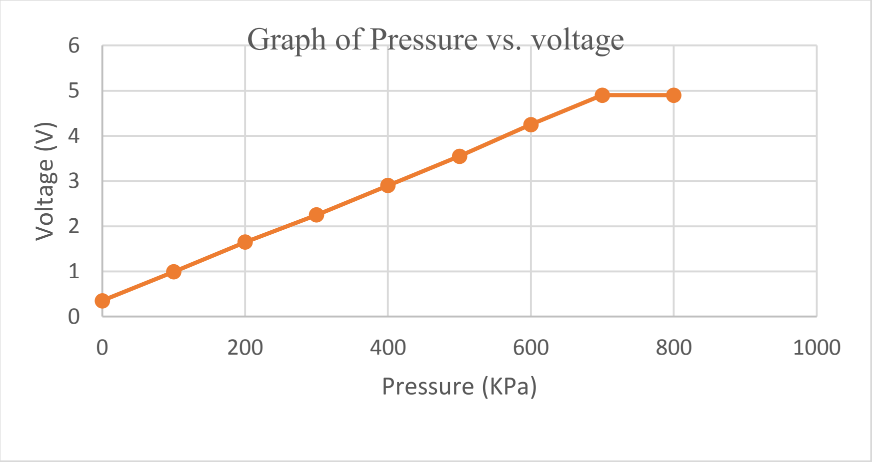


Figure 4.18: Graph of ideal voltage

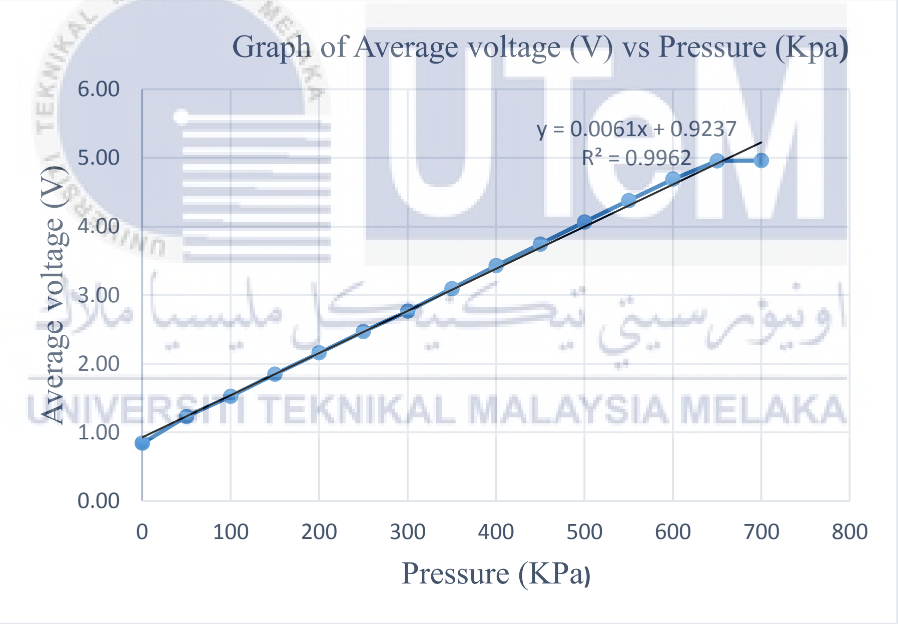


Figure 4.19: Graph of real voltage

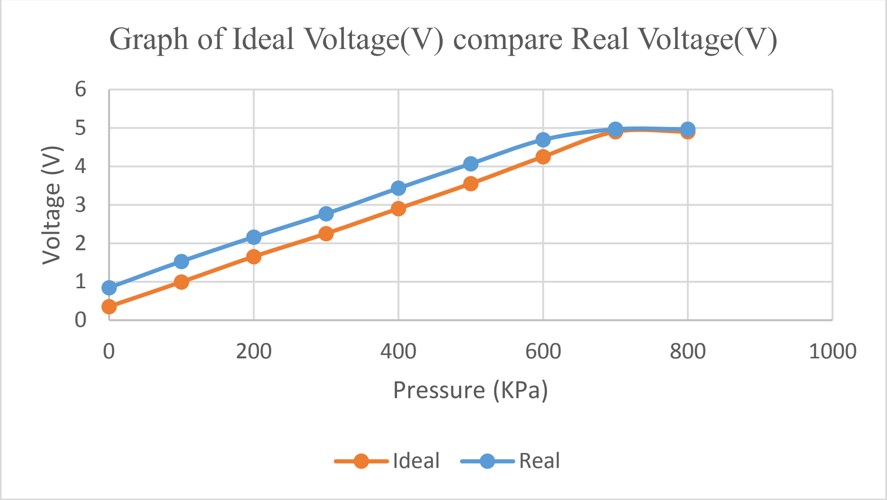


Figure 4.20: Graph of ideal voltage compare with real voltage

Formula equation 4.10 was used to obtain linear equation of the pressure sensor and to construct analog to digital converter. Several data was analyze and the percentage error was calculated to choose a best result.

$$Y = mX + c \tag{4.10}$$

$$m = \frac{2.93-1.44}{150-75} \tag{4.11}$$

$$Y = 0.0061x + C$$

$$m = \frac{Y2-Y1}{x2-x1} \tag{4.12}$$

$$c = -0.9237$$

$$\text{Percentage of error} = \frac{Experimental - Theoretical}{Theoretical} \times 100\% \tag{4.13}$$

Table 4.6 until Table 4.9 shown the percentage error between ideal depth and real depth. The 4.13 equation is used to calculate percentage error for each reading. Ideal depth refer to the actual depth that obtain from converting bar to depth value. The real depth refer to experimental depth. The real depth was obtained from analog to digital converter simulation.

Reading 1

m= 0.00640

C= - 0.86330

Table 4.6: Percentage error of reading 1

Voltage (V)	Real depth (m)	Ideal depth (m)	Difference	Error (%)
0.84	0	0	0	0
1.23	5.84	5.10	0.74	14.51
1.52	10.46	10.20	0.26	2.55
1.85	15.72	15.30	0.42	2.75
2.16	20.65	20.40	0.25	1.23
2.47	25.59	25.50	0.09	0.35
Total error				21.39
Average Error				3.57

Reading 2

m= 0.00633

c= -0.89390

Table 4.7 Percentage error of reading 2

Voltage (V)	Real depth (m)	Ideal depth (m)	Difference	Error (%)
0.84	0	0	0	0
1.23	5.412	5.10	0.312	6.12
1.52	10.08	10.20	0.12	1.18
1.85	15.4	15.30	0.10	0.65
2.16	20.39	20.40	0.01	0.04
2.47	25.38	25.50	0.12	0.47
Total error				8.46
Average Error				1.41

Reading 3

m= 0.00607

c= -0.94230

Table 4.8: Percentage error of reading 3

Voltage (V)	Real depth (m)	Ideal depth (m)	Difference	Error (%)
0.84	0	0	0	0
1.23	5.10	5.10	0.26	5.10
1.52	10.20	10.20	0.498	4.88
1.85	15.30	15.30	0.06	0.39
2.16	20.40	20.40	0.05	0.25
2.47	25.50	25.50	0.16	0.63
Total error				11.25
Average Error				1.88

Reading 4

m= 0.0061

c= -0.9237

Table 4.9: Percentage error of reading 4

Voltage (V)	Real depth (m)	Ideal depth (m)	Difference	Error (%)
0.84	0	0	0	0
1.23	5.119	5.10	0.019	0.37
1.52	10.20	10.20	0.235	4.88
1.85	15.30	15.30	0.18	1.18
2.16	20.40	20.40	0.26	1.27
2.47	25.50	25.50	0.34	1.33
Total error				6.45
Average Error				1.08

Table 4.10 shows the summary of average error for each reading. The highest error value is reading 1 which is 3.57% while the less error is reading 4 which is 1.08%. The reading 4 was chosen in term of less error than other reading. The value of m= 0.0061 and c= -0.9237 were accepted to use in analog to digital converter simulation system.

Table 4.10: Summary of average error

Reading	Total error (%)	Average error (%)
1	21.39	3.57
2	8.46	1.41
3	11.25	1.88
4	6.45	1.08

4.2.5 Experiment 5: Effect a real-time simulation system using Microbox 2000/2000C with prototype

Table 4.11 show the system performance of real time data in term of rise time, settling time, overshoot and steady state error. Several real time data was tested and verified by using system identification. In the table below, a data 11 shows the better performance in terms on no overshoot, faster rise time, settling time and small steady state error value. The transfer function of data 11 was chosen to use in PID and fuzzy logic controller simulation system.

Table 4.11: System performance of real time data

Data	Rise Time (Tr)	Settling Time (Ts)	Overshoot (%)	Steady state
1	296	452	0	-38.3
2	2.58	75.4	13	1.22e03
3	0.00471	5.99	2.72e03	0.115
4	256	453	0	34
10	1.9	36.1	57.4	-6.81e04
11	0.202	4.94	0	0.719

Table 4.12 shows an output response of real-time simulation PID controller before tuning process. The result of the rise time, settling time, overshoot and steady state error become increase than real-time open loop simulation result. The automatic tuning process was applied to the simulation system in order to get a better performance. The table 4.13 shows a result after 4 times applied tuning process. When times of tuning process is increased, the percentage of overshoot display a better value but the rise time and settling time shows an increment value while steady state error remains the same.

Table 4.12: Output response of real-time simulation PID controller

Real time result	Tr	Ts	Overshoot (%)	Ess
	6.94s	11.1s	0.275	1

Table 4.13: Result of automatic tuning PID

Tuning process	Rise Time(s)	Settling Time(s)	Overshoot (%)	Steady state error
1	6.94	11.1	0.275	1
2	9.02	15.5	0.00002	1
3	12.3	22.1	0	1
4	10.3	18	0	1

Table 4.14: Simulation result of automatic tuning PID

Tuning process	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Ess	Diff. Ess
1	6.94	NA	11.1	NA	0.275	NA	1	NA
2	9.02	-2.080	15.5	-4.4000	0.00002	0.2749	1	0
3	12.3	-5.3600	22.1	-11.00	0	0.2750	1	0
4	10.3	-3.3600	18	-6.9	0	0.2750	1	0

4.2.6 Experiment 6: System identification

The result of real time simulation was analyzed by system identification as shown in table 4.15. Several data with high best fits was chosen to obtain a transfer function model. Based on table 4.15, the data 11 shows the best value which zero overshoot, faster rise time (0.202s) and setting time (4.94s). The transfer function of data 11 was used to study effect output response of PID and effect of membership function of real-time fuzzy logic controller.

Table 4.15: System identification of real time data

Data	Best fits (%)
1	88.67
2	41.95
3	20.31
4	0.7294
5	0.2646
6	88.05
7	34.79
8	12.75
9	64.7
10	69.7
11	89.07

4.2.7 Experiment 7: Stability

The table 4.16 below indicate the stability of data in term of controllability, obervability and asymptotic stability. In order to implement control system to the transfer function that obtained from data, the stability function need to identify either the system able to observe and control or not. The stability of data was identify in state space condition. The data 1 only show an asymptotic unstable based on eigenvalue of matrix A. The other data shows in controllable, observability and asymptotic stable mode.

Table 4.16: Stability test of real-time data

Data	Controllability	Observability	Asymptotic stability
1	/	/	X
2	/	/	/
3	/	/	/
4	/	/	/
5	/	/	/
6	/	/	/
7	/	/	/
8	/	/	/
9	/	/	/
10	/	/	/
11	/	/	/

Legend:

Controllable =/

Uncontrollable= X

Observable=/

Unobservable= X

Asymptotically stable =/

Asymptotic unstable= X

4.2.8 Experiment 8: Effect of membership function for real-time Fuzzy Logic Controller.

Table 4.17 shows the result for change range of input 1 of fuzzy logic. The result show when range input 1 was increased, the output response display unchanged condition (not applicable)

Table 4.17: Simulation result for change range input 1

Input 1									
Range	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Settling max	Ess	Diff. Es
0-2	NA	NA	NA	NA	0	0	NA	NA	NA
0-4	NA	NA	NA	NA	0	0	NA	NA	NA
0-6	NA	NA	NA	NA	0	0	NA	NA	NA
0-8	NA	NA	NA	NA	0	0	NA	NA	NA
0-10	NA	NA	NA	NA	0	0	NA	NA	NA

Table 4.18 shows the result by changing range input 2 of fuzzy logic while the range input 1 set to 0-10. The output response display unchanged condition same as output response at table 4.17.

Table 4.18: Simulation result for change range input 2

Input 2									
Range	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Settling max	Ess	Diff. Es
0-2	NA	NA	NA	NA	0	0	NA	NA	NA
0-4	NA	NA	NA	NA	0	0	NA	NA	NA
0-6	NA	NA	NA	NA	0	0	NA	NA	NA
0-8	NA	NA	NA	NA	0	0	NA	NA	NA
0-10	NA	NA	NA	NA	0	0	NA	NA	NA

The table 4.19 show output response of fuzzy logic simulation by changing range of output. The condition of output response change when output range in 0-14. In this range the faster rise time and small steady state error able to obtain while zero overshoot condition was not achieved. When the range change to 0-13.91, the zero overshoot condition able to achieve while rise time and settling time value was increased. The steady state error value remains same.

Table 4.19: Simulation result for change range output

Output									
Range	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Max Settling	Ess	Diff. Es
0-2	NA	NA	NA	NA	0	0	NA	NA	NA
0-4	NA	NA	NA	NA	0	0	NA	NA	NA
0-6	NA	NA	NA	NA	0	0	NA	NA	NA
0-8	NA	NA	NA	NA	0	0	NA	NA	NA
0-10	NA	NA	NA	NA	0	0	NA	NA	NA
0-12	NA	NA	NA	NA	0	0	NA	NA	NA
0-14	0.1990	NA	4.5633	NA	0.2016	0	4.9974	0.9974	NA
0-13.91	0.2016	-0.0026	4.9741	-0.4108	0	0.2016	4.9974	0.9974	0

Based on result table 4.19, changing the range of output fuzzy logic controller will affect the output response. The input 1 and input 2 range value were not affected the performance of fuzzy logic controller system.

The experiment continue by shifting membership function of fuzzy logic to increase the output performance. Table 4.20 show the output performance in term of average rise time and settling time. In table, the input 1 was shifting to the center, left and right. The faster average rise time and settling time indicate the best performance. The shifting input 1 to the ‘center’ display the best value than ‘left’ and ‘right’.

Table 4.20: Average rise time and settling time for input 1 membership function

Shifting condition	Tr				Ts			
	1	2	2	Average	1	2	3	Average
Center	0.2015	0.2015	0.2015	0.2015	4.9482	4.9481	4.9481	4.9481
Left	0.2029	0.2009	0.2013	0.2017	4.9685	4.9739	4.9685	4.9664
Right	0.2015	0.2015	0.2015	0.2015	4.9483	4.9481	4.9481	4.9482

Table 4.21 below show the output response (overshoot and steady state error) by shifting input 1 in three condition. Output response display the same value although in different shifting condition.

Table 4.21: Average percent overshoot and steady state error of input 1

Shifting condition	Overshoot (%)				Max Settling				Ess
	1	2	2	Average	1	2	3	Average	Average
Center	0	0	0	0	4.9992	4.9992	4.9992	4.9992	0.99
Left	0	0	0	0	4.9979	4.9974	4.9979	4.9977	0.99
Right	0	0	0	0	4.9992	4.9992	4.9992	4.9992	0.99

The summary of average output performance for input 1 was tabulated as shown in Table 4.22. The change of each performance was calculated in order to evaluate the best shifting membership function. The 'center' condition show the same performance. The 'left' condition display decreased performance in term of rise time and settling time. The 'right' condition remains the same performance except settling time show a decreasing performance. Based on table 4.23, the shifting membership function input 1 at 'center condition' show the best performance than other condition.

Table 4.22: Summary of average output performance for input 1

Input 1									
Shifting condition	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Max. Settling	Ess	Diff. Ess
Center	0.2015	NA	4.9481	NA	0	NA	4.9992	0.99	NA
Left	0.2017	-0.0002	4.9664	-0.0183	0	0	4.9977	0.99	0
Right	0.2015	0	4.9482	-0.0002	0	0	4.9992	0.99	0

Table 4.23: Simulation performance of input 1 membership function

Shifting condition	Tr	Ts	%OS	Ess
Center	🟨	🟨	🟨	🟨
Left	🟥	🟥	🟨	🟨
Right	🟨	🟥	🟨	🟨

Legend	
🟨	Same performance
🟩	Increasing performance
🟥	Decreasing performance

Average output performance of rise time and settling by shifting membership function input 2 as shown in table 4.24. The best rise time and settling time when input 2 in ‘right’ shifting condition. The table 4.25 show the average percentage overshoot and steady state error input 2 which remains same in all shifting condition.

Table 4.24: Average rise time and settling time of input 2 membership function

Shifting condition	Tr				Ts			
	1	2	2	Average	1	2	3	Average
Center	0.2015	0.2015	0.2015	0.2015	4.9482	4.9481	4.9481	4.9482
Left	0.2046	0.2015	0.2015	0.2025	4.9481	4.9481	4.9481	4.9481
Right	0.2015	0.2015	0.2015	0.2015	4.9481	4.9481	4.9481	4.9481

Table 4.25: Average rise time and settling time of input 2 membership function

Shifting condition	Overshoot (%)				Max Settling				Ess
	1	2	2	Average	1	2	3	Average	Average
Center	0	0	0	0	4.9992	4.9992	4.9992	4.9992	0.99
Left	0	0	0	0	4.9992	4.9992	4.9992	4.9992	0.99
Right	0	0	0	0	4.9992	4.9992	4.9992	4.9992	0.99

Table 4.26 indicate the summary of average output performance for shifting membership function input 2. The comparison between three shifting condition as shown in Table 4.27. ‘Center’ shifting condition remains the same performance. The ‘left’ condition display decreasing performance in term of rise time. The increasing performance show in settling time. The overshoot and steady state error remains same performance at ‘center’ condition. The overshoot, steady state error and rise time of ‘center’ condition show same performance while settling time get e better performance.

Table 4.26: Summary of average output performance for input 2

Input 2									
Shifting condition	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Max. Settling	Ess	Diff. Ess
Center	0.2015	NA	4.9482	NA	0	NA	4.9992	0.99	NA
Left	0.2025	-0.001	4.9481	0.0001	0	0	4.9992	0.99	0
Right	0.2015	0	4.9481	0.0001	0	0	4.9992	0.99	0

Table 4.27: Simulation performance of input 2 membership function

Shifting condition	Tr	Ts	%OS	Ess
Center	🟨	🟨	🟨	🟨
Left	🟥	🟩	🟨	🟨
Right	🟨	🟩	🟨	🟨

Legend	
🟨	Same performance
🟩	Increasing performance
🟥	Decreasing performance

Average output performance of rise time and settling time by shifting membership function output as shown in table 4.28. The best rise time and settling time when output in ‘right’ shifting condition than ‘center’ and ‘left’. Overshoot and steady state value for ‘left’ and ‘right’ remains same while ‘center’ in not applicable condition as shown in Table 4.29.

Table 4.28: Average rise time and settling time of output membership function

Shifting condition	Tr				Ts			
	1	2	2	Average	1	2	3	Average
Center	0.2015	0.2015	NA	NA	4.9554	5.8813	NA	NA
Left	0.2015	0.2015	0.2015	0.2015	4.9253	4.9493	9.2299	6.3682
Right	0.2015	0.2015	0.2015	0.2015	4.9664	4.9560	4.8574	4.9266

Table 4.29: Average overshoot and steady state error of output membership function

Shifting condition	Overshoot (%)				Max Settling				Ess
	1	2	2	Average	1	2	3	Average	Average
Center	0	0	NA	NA	4.9992	4.9992	NA	NA	NA
Left	0	0	0	0	4.9992	4.9992	4.9992	4.9992	0.99
Right	0	0	0	0	4.9992	4.9992	4.9992	4.9992	0.99

The change of each output performance in three different shifting condition shown in Table 4.30. The performance was evaluated between 'left' and 'right' shifting condition. When the output shifting to the 'center', the rise time, settling time, percentage overshoot and steady state error remains same performance. When the output shifting to the 'right' the settling time show increasing performance while other parameter indicate the same performance as shown in Table 4.31.

Table 4.30: Summary of average output performance for output

Output									
Shifting condition	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Max. Settling	Ess	Diff. Ess
Center	NA	NA	NA	NA	NA	NA	NA	NA	NA
Left	0.2015	NA	6.3682	NA	0	NA	4.9992	0.99	NA
Right	0.2015	0	4.9481	1.4201	0	0	4.9992	0.99	0

Table 4.31: Simulation performance of output membership function

Shifting condition	Tr	Ts	%OS	Ess
Center				
Left	🟧	🟧	🟧	🟧
Right	🟧	🟩	🟧	🟧

Legend	
🟧	Same performance
🟩	Increasing performance
🟥	Decreasing performance
⬜	NA

The table 4.32 show result for shifting membership function input 1, input2, and output. Based on table 4.33, the same performance of ‘center’ and ‘right’ for rise time. The ‘left’ condition show a decreasing performance. The settling time of ‘left’ show increasing performance but ‘right’ condition show decreasing performance while ‘center’ remains same performance. The zero overshoot were achieved by shifting membership function to the ‘center’ and ‘right’ only. The steady state error at ‘left’ condition show an increasing performance than ‘center’ and ‘right’.

Table 4.32: Simulation result for shifting membership function

Input 1, Input 2, Output									
Shifting condition	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Max. Settling	Ess	Diff. Ess
Center	0.2015	NA	4.9481	NA	0	NA	4.9992	0.99	NA
Left	0.2045	-0.003	3.4590	1.4891	1.6347	-1.6347	5.0817	0.08	0.91
Right	0.2015	0	4.9483	-0.0002	0	1.6347	4.9992	0.99	0

Table 4.33: Simulation performance for shifting of membership function

Shifting condition	Tr	Ts	%OS	Ess
Center	🟧	🟧	🟧	🟧
Left	🟥	🟩	🟥	🟩
Right	🟧	🟥	🟧	🟧

Legend	
🟧	Same performance
🟩	Increasing performance
🟥	Decreasing performance

Table 4.34: Summary result for effect of shifting membership function

Summary result for effect of shifting membership function								
Shifting condition	Tr	Diff. Tr	Ts	Diff. Ts	%OS	Diff. %OS	Ess	Diff. Ess
Input 1 (Center)	0.2015	NA	4.9481	NA	0	NA	0.99	NA
Input 2 (Right)	0.2015	0	4.9481	0	0	0	0.99	0
Output (Right)	0.2015	0	4.9481	0	0	0	0.99	0
Input 1, Input 2, Output (Center)	0.2015	0	4.9481	0	0	0	0.99	0

Based on table 4.35, it shows the best result output performance of fuzzy logic controller in experiment 8 that involves three shifting conditions.

Table 4.35: Summary of output response for each membership function

Membership function	Tr(s)	Ts(s)	%OS	Ess
Input 1 ‘center’	0.2015	4.9481	0	0.99
Input 2 ‘right’	0.2015	4.9481	0	0.99
Output ‘right’	0.2015	4.9481	0	0.99
Input 1, input 2, output ‘center’	0.2015	4.9481	0	0.99

The output response for different type of controllers shown in Figure 4.21, figure 4.22 and figure 4.23. Based on table 4.36, it clearly show the output performance of rise time, settling time, overshoot and steady state error with different type of controller. The fuzzy logic controller show the faster rise time and settling time than mathematical modelling and PID. All type of control achieved no overshoot condition. The mathematical modelling show the small steady state error than PID and fuzzy logic controller.

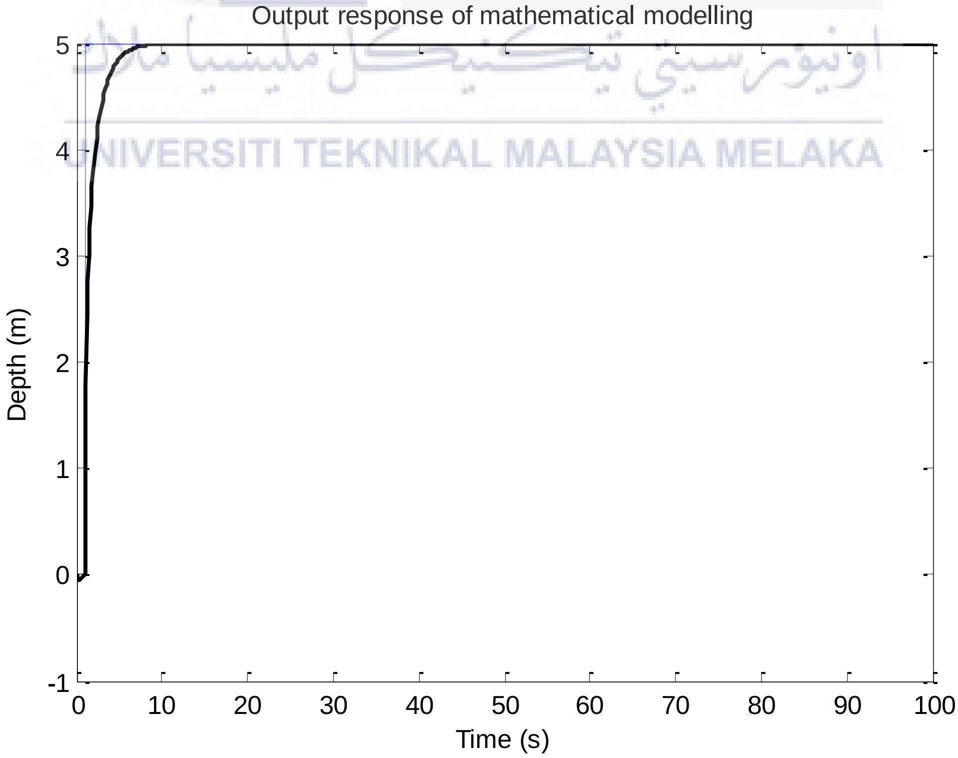


Figure 4.21: Graph depth vs. time for mathematical modelling

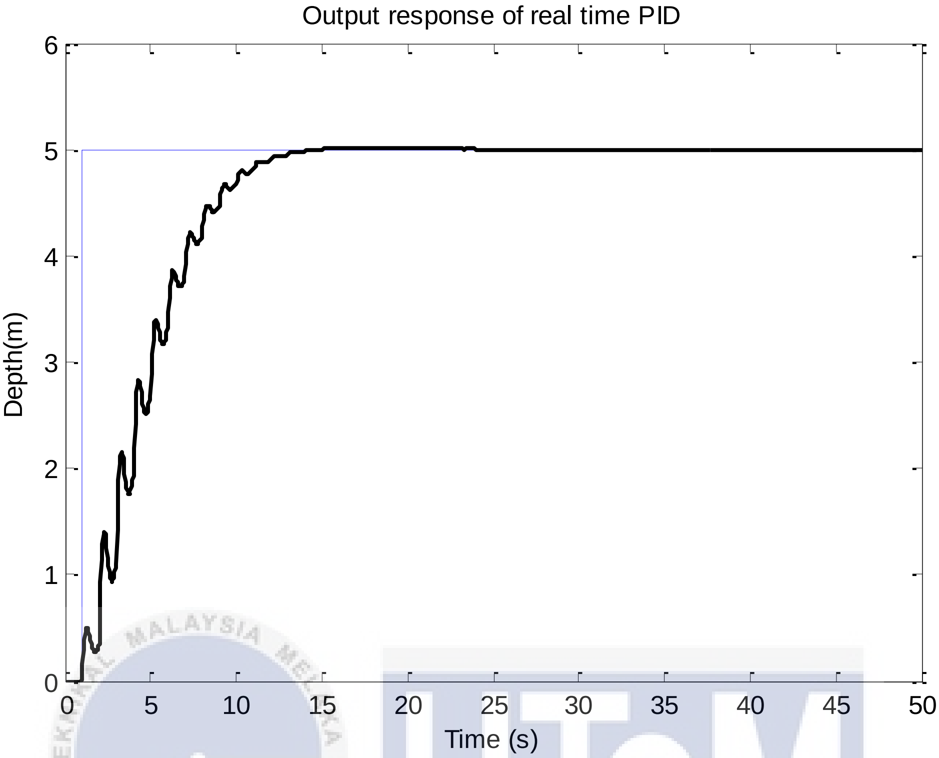


Figure 4.22: Graph depth vs. time for real time PID

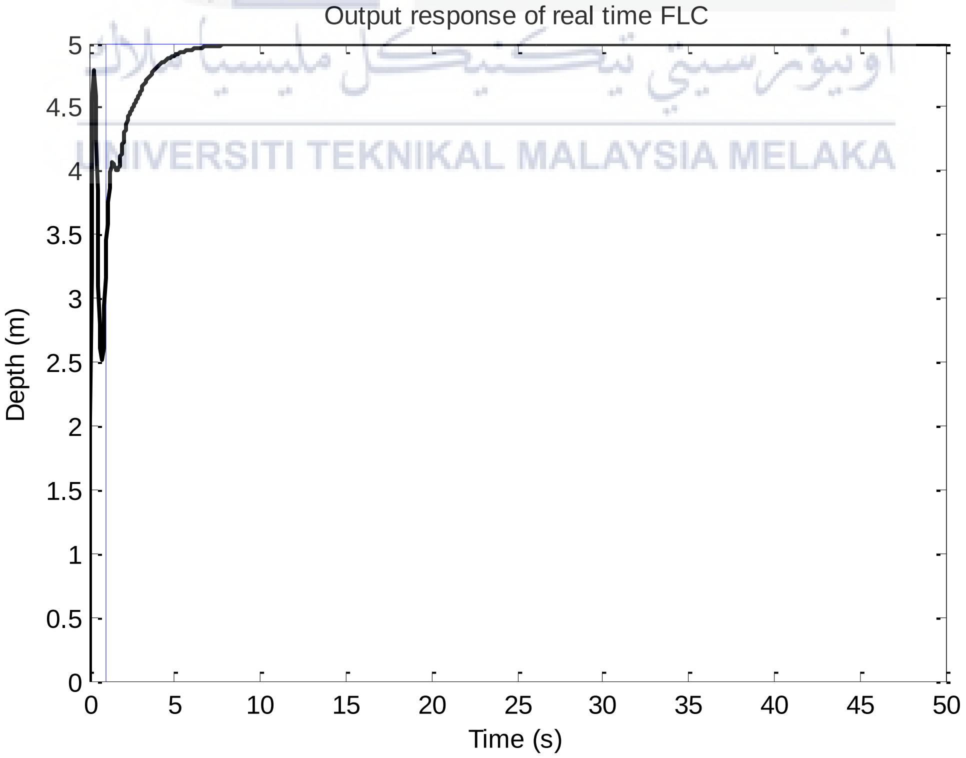


Figure 4.23: Graph depth vs. time for real time FLC

Table 4.36: Comparison output response with different type of controller

Type of controller	Rise time (s)	Settling time (s)	Overshoot (%)	Steady state error
Mathematical modelling	2.1407	5.5639	0	0
Real time PID	10.3	18	0	1
Real time of fuzzy logic controller	0.2015	4.9481	0	0.99

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 CONCLUSION

As a conclusion, the experiment 1 shows by using PID controller, a zero overshoot performance condition is achieved. However the value of rise time is increased.The second experiment, fuzzy logic controller was used as a control system in order to achieve a better output performance. Based on result, it clearly shows a fuzzy logic controller display a better performance which is 0.75s faster rise time than PID and 0.60 difference in term of steady state error. The output performance of FLC in term of faster rise time, zero overshoot and small steady state error were better than PID. In experiment 4, the mathematical modelling of ROV is used by using properties and coefficient of VideoRay Pro III [10]. The output response of modelling simulation show a smooth shape of graph. The zero overshoot with faster rise time and small steady state error was achieved. Experiment 4 was related with pressure sensor that used as a feedback in control system. The result of pressure sensor was compared between ideal voltage and real voltage. The analog to digital converter able to construct by using pressure sensor data. Experiment 5 shows a real-time performance. The zero overshoot was able to achieve by using real-time PID simulation but performance of rise time and settling time were decreased. The steady state error maintain at 1. Experiment 6 was performed to analyze the best fits real-time data. The data 11 was chosen to implement into fuzzy logic controller. The stability test shown in experiment 7. All real-time data shown observable and controllable result. However, the data 1 shown asymptotic unstable. The experiment 8 was conduct to study the effect of real time fuzzy logic controller. Result show that the fuzzy logic controller display the best response for faster rise time and settling time. The zero overshoot and small steady state error also achieved.

5.1 RECOMMENDATION

As a recommendation, the study for implementation of fuzzy logic controller for ROV should be continued because there are several method that can be applied such as adjusting rules of membership function which is not cover in this project. Further implementation involving sensor detect a wave pattern, disturbance or noise of environment in order to improve accuracy of depth control. The new approach which is combination fuzzy-neural network system can be implemented to study the effect output performance. The analysis of fuzzy-neural can be compared with fuzzy logic controller method.

REFERENCES

[1] S. M. Zanoli and G. Conte, “Remotely operated vehicle depth control,” Control Eng. Pr., vol. 11, no. 4, pp. 453–459, 2003.

[2] Aras M.SM, Abdullah S.S and Rashid M.Z.A, “Development and Modelling Of Unmanned Underwater Remotely Operated Vehicle Using System Identification for Depth Control" Theoretical and Applied Information Technology, 2013, Vol. 56.No.1.

[3] Aras M.SM, Abdullah S.S and Rashid M.Z.A, “Tuning Process of Single Input Fuzzy Logic Controller Based on Linear Control Surface Approximation Method for Depth Control of Underwater Remotely Operated Vehicle" Engineering and Applied Sciences, 208-214,2013

[4] I. S. Akkizidis, G. N. Roberts, P. Ridao, and J. Batlle, “Designing a Fuzzy-like PD controller for an underwater robot,” Control Eng. Pr., vol. 11, no. 4, pp. 471–480, 2003.

[5] Andrzej Zak, “Fuzzy Controller for Underwater Remotely Operated Vehicle Which Is Moving In Conditions Of Environment Disturbance Occurance,” Kones Poertarain and Transport, vol. 18, no. 2, 2011.

[6] CNN. “Behind The Scenes: Filming Remotely Operated Underwater Vehicles at Sea,” edition.cnn.com.[Online].Available:edition.cnn.com/2014/04/01/world/ europe/scotland-rov-demonstration [Accessed: Oct 1, 2014].

[7] BP. “Investigations and Legal Proceedings,” www.bp.com. [Online]. Available: www.bp.com/en/global/corporate/gulf-of-mexico-restoration/investigations-and-legal- proceedings.html [Accessed: Oct 1, 2014].

[8] California State University, Northridge. “Fluid Pressure,” www.csun.edu [Online]. Available: www.csun.edu/science/activities/pressure/fluid- pressure [Accessed: November 28, 2014].

[9] M.P. Kevin, “Intelligent Control: An Overview of Techniques”, the Ohio State University, “Perspectives in Control: New Concepts and Applications,” *IEEE Press*, 2001

[10] Wei Wang and Christopher M. Clark, “Modelling and Simulation of the VideoRay Pro III Underwater Vehicle" University of Waterloo, Canada, 208-214,201.