Development of Sensors and Microcontrollers for Underwater Robots

By

Ali Jebelli

Thesis submitted to
The Faculty of Graduate and Postdoctoral Studies
In partial fulfilment of the degree requirements of
Master of Applied Science
In
Electrical and Computer Engineering

Ottawa-Carleton Institute for Electrical and Computer Engineering
School of Electrical Engineering and Computer Science
University of Ottawa
Ottawa, Ontario, Canada

Abstract

Nowadays, small autonomous underwater robots are strongly preferred for remote exploration of unknown and unstructured environments. Such robots allow the exploration and monitoring of underwater environments where a long term underwater presence is required to cover a large area. Furthermore, reducing the robot size, embedding electrical board inside and reducing cost are some of the challenges designers of autonomous underwater robots are facing. As a key device for reliable operation-decision process of autonomous underwater robots, a relatively fast and cost effective controller based on Fuzzy logic and proportional-integral-derivative method is proposed in this thesis. It efficiently models nonlinear system behaviors largely present in robot operation and for which mathematical models are difficult to obtain. To evaluate its response, the fault finding test approach was applied and the response of each task of the robot depicted under different operating conditions. The robot performance while combining all control programs and including sensors was also investigated while the number of program codes and inputs were increased.

Acknowledgements

First, I would like to express my honestly gratitude and appreciation to my supervisor, Professor Mustapha C.E. Yagoub for his encouragement, guidance, critics and friendship. Without his continued support and interest, this thesis would not have been the same as presented here. My sincere appreciation is extended to everyone from School of Electrical Engineering and Computer Science and Department of Mechanical Engineering of the University of Ottawa for their advice and comments.

Finally, a special thanks to my family for the support they provided me through my entire life and in particular, I must acknowledge my wife, Nafiseh, without whose love and encouragement, I would not have finished this thesis.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Variables

Acronym	Definition
AUV	Autonomous Underwater Vehicles
ADC	Analog to Digital Converter
Cd	Drag coefficient of the object
DOF	Degree of Freedom
KT	Thrust coefficient
Fd	Force of drag
g	Acceleration of gravity
h	Height of the fluid above the object
KQ	Torque coefficient
m_b	Buoyant mass
NN	Neural network
P	Proportional controller
PI	Proportional-integral controller
PID	proportional-integral-derivative controller
ρ	Density of the fluid
Q	Torque of the propeller
ROV	Remotely Operated Vehicles
RPM	Speed of the motor or propeller
SDO	Serial data output
SMC	Sliding mode control Measured thrust
T°	Temperature
v	Speed of the object relative to the fluid
AV	Advance speed of the propeller

CHAPTER 1

INTRODUCTION

1.1 Motivation

Seas and oceans have always fascinating humans. As quoted in the Ocean Portal Team[1] of the National Museum of Natural History, “Deep below the ocean’s surface is a mysterious world that takes up 95% of Earth’s living space.” Also, in the National Oceanography Centre webpage[2], we can read that “The ocean environment holds a wealth of resources that we rely on, from fuel sources to food supplies. Most of our oil and gas reserves lie beneath the sea floor, and many are yet to be discovered. In the future more electricity will be generated from waves, tidal currents, tidal barrages and offshore wind. Exhausted offshore oil and gas wells will be used to store the carbon dioxide produced by burning fossil fuels. Around the world more people are relying on food from our seas and oceans.”

[1] http://ocean.si.edu/deep-sea
[2] http://noc.ac.uk/science-technology/marine-resources

However, this human desire to utilize resources of seas and oceans cannot become real without a deep understanding of their environment. Because understanding cannot be achieved without thoughtful exploration of the underwater world, the development of underwater vehicles for submarine exploration is of high interest from both industrials and the scientific community.

Unfortunately, manned submarines usually experience difficulties and can become unusable due to the fatigue of their operators, their huge size, high construction and transportation costs and safety issues. Thus, researchers decided to construct small unmanned submarines like Autonomous Underwater Vehicles (AUVs). In fact, their small size lets them move in any direction with less equipment and less power needs than manned submarines.

Autonomous robots, as a research issue, aimed at analyzing some matters such as sensing, actuation, powering, communication, control theory and artificial intelligence. Purpose of autonomous robot design is to create a machine able to function in the real world. But the problem is, as the AUVs become smaller, their functional ability falls and sometimes they do not work at all due to high water depth. Thus, although a huge part of the globe is covered with water, human being or submarine can hardly reach specific spots due to matters of water temperature, extent of water and safety.

Unmanned underwater robots should be able to properly receive the environment’s changes through sensors and by interpreting the received data, they ought to have the ability to choose the correct alterations and make a reasonable decision and do the right thing. But this procedure can be very difficult for a senseless computer that cannot understand the environment’s structure, interpret the received data and decide correctly in order to go on with its missions. To solve the aforementioned issues, an artificial intelligence technique can be used; thus, the required characteristics in a robot’s intelligent behavior were gathered in control architecture and mechatronic was used to coordinate the entire system of a robot.

Mechatronic is an essential and important system in the design of an intelligent robot; mechatronic systems consist of mechanical, electronic and software parts that give a robot the power to conceive artificial intelligence. Furthermore, when all these three parts are properly united, they can reinforce each other’s functions so that the robot can make the best decision. But due to the high amount of energy it utilizes, it may experience restrictions, especially underwater submarines that need huge amounts of energy in water depths [1], [2].

1.2 Problem Statement

Mechatronics has been also extensively introduced in robotics since principles of mechanical machines and electronic systems are closely related. In many cases, due to the integrated nature of the design, operation of these elements cannot be separated. Some tangible and explanatory examples of advanced Mechatronics are Autonomous robots.

Nowadays, most AUVs are heavy and expensive as they are highly equipped. But in a small AUV, many of these issues are solved and its motion in narrow routes has been smoothed. An autonomous robot must be able to act in a given environment by changing its states or be able to sense the state of that environment. Finally, it must decide what to do, find the relation between this environment conditions and its operation possibilities in order to achieve a predefined goal.

However, these small underwater robots experience issues namely, high driving force requirement and risk of water waves and whirlpools. As a result, design of these structures demands accuracy, speed and control. In fact, design of such a robot is a very complicated process, since it deals with a complex environment changing consistently.

Exchanging data with a constantly changing environment requires efficient sensor network. Therefore, sensors are inseparable parts of autonomous underwater submarines. The condition/operation relation, required for human intelligence, is a complex function for a computerized system, especially in unknown environments. To resolve this issue, two problems should be removed. The first is the condition interpretation problem, that is, recognition of what a sensor distinguish. The second is the operation-decision problem that is decision about the robots movements.

Indeed, some of the most influential factors that can degrade the performance of AUVs are those directly related to sensor efficiency in terms of receiving and perceiving time, data transmission to the controller, interpretation and analysis of the data and finally reaching back to the balanced state. Thus, choosing compatible sensors is one of the issues and purposes of this thesis.

1.3 Thesis Objectives

The aim of this thesis was to design a cost-effective AUV for use in water depths up to 6 meters. The focus was then made on designing a small, light, and affordable AUV.

Therefore, various mechanical and electrical parts of autonomous submarines were discussed, such as cost-effective sensors and tiny microcontroller, to properly perform the calculation and navigation requirements, primary storage of the information and data and their interpretation. Indeed, beside sensors, one of the major units, i.e., the control unit, was investigated and a microcontroller designed. Based on Fuzzy Logic and PID (proportional-integral-derivative) control approach, it is intended to be used in small autonomous underwater robots. Using a fuzzy control approach is a proper method to solve the difficulties of information control. This special architecture enables the robot to respond fast enough and provide higher levels of deliberation and perception as well as reliable performance in the robot’s activities [3]-[5].

Also, an intelligent system of temperature and energy control was designed to let the robot move as much as possible with minimum energy consumption.

The proposed design can be divided into three main categories: hardware (electronic systems), mechanical, and software (programming). Figure 1.1 shows a robot design flow.

1.3.1 Mechanical Issues

The class of AUV will guide relevant aspects of the design particularly buoyancy, water tight integrity, power supply, thrusters positioning, drag, restoring forces and construction.

The most common design is based on the work class AUV is provided by cylindrical shape body which is the best way to get best space and to get the least resistance. Because of the main movement of robot that to be expected horizontally, the resistance of robot's body in this direction is low.

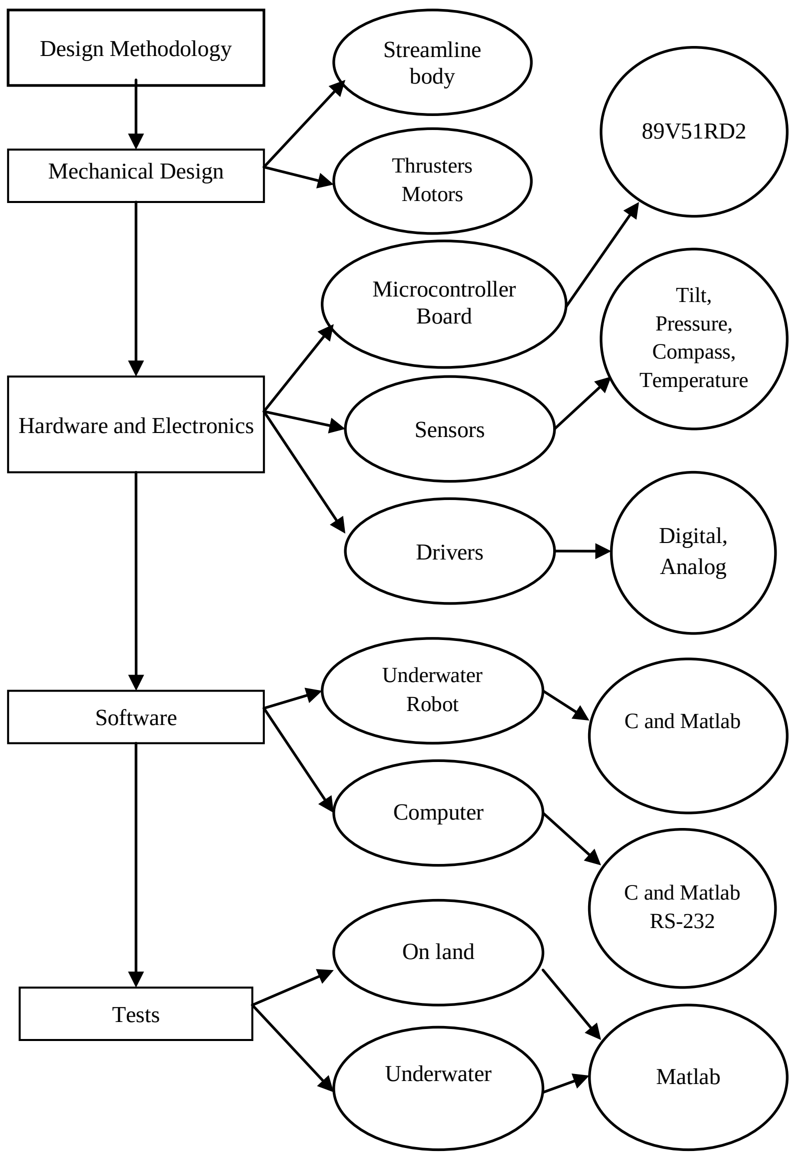

Figure 1.1 Robot design flow.

1.3.2 Controls and Electronics

Autonomous underwater robots are typical examples of mechatronic system requiring intelligent control. Such vehicles comprise of many subsystems operating in a real time domain. These systems perform complex and amending tasks in any environment predominantly regarded as hostile.

Sensors are added enabling environmental data and relevant control laws (in real time) to be integrated during processing thus enabling of these vehicles to achieve a predicted outcome, for this work some sensors which are essential for robot movements including compass sensor to find the heading point and robot's direction, tilt sensor to find the tilt degree of the robot, and pressure sensor to find the depth of the robot are used and temperature sensor to control the heating dissipation due to the motors and batteries.

This work is based on autonomous underwater robot requirements to perform some behavioral tasks intelligently on embedded microcontroller with limited memory and processing capabilities. Because of some limitations related to design of small underwater robots, the robot cannot use high efficient handy boards and computer board, so the board must be as smallest as it can be which is one of the main parts of this work, involved the implementation of the fuzzy logic algorithm using C language into microcontroller for some behaviors which are dealing directly with sensors and thrusters. All of the computation will be done by PHILIPS 89V51RD2 microcontroller. Also for communicate with robot, one of the simple ways is to use a RS-232 link which can communicate with two wires and it is also cheap and we don't have to deal with noises and some issues which we would face with when we use wireless communication.

1.3.3 Software Design

In this work, the commercial software SolidWorks was utilized to simulate the robot while virtual real-world environments were built to test the produced designs before manufacturing. Such tests included tests against durability, static and dynamic response, motion of assembly, heat transfer, and fluid dynamic as well as with intelligent single line or traditional multiline schematics. In this aim, we developed embedded electrical system designs in real-time, and a 3D robot model within a collaborative environment for test of mechanical and electrical parts.

To design the microcontroller, the SDCC (Small Device C Compiler) was used. It is a free C compiler developed for 8-bit microcontrollers and used to develop firmware for the DS89C430/450 family of ultra-high-speed 8051-compatible microcontrollers.

To simulate fuzzy logic and communication by RS-232, one of the popular software is Matlab. We then used its fuzzy-logic toolbox to create and edit fuzzy inference systems. Then, graphical tools or command-line functions were used to test the fuzzy system in a block diagram simulation environment.

1.4 Robot's Specifications

This thesis documents the development of a low-cost underwater manipulator, which specifications are summarized below:

- Weight: 7.5 kg,
- Operating Depth: Up to 6 m,
- Material used to build the body: Plastic
- Speed: 1 to 3 m/s

1.5 Contributions

In this work, several contributions have been added to cost-effective and light AUV design:

- Design of a small lightweight autonomous underwater robot, taken as reference, with minimum sensors.
- Design, fabrication, and test of compatible and inexpensive sensors,
- Design, fabrication, and test of a microcontroller for AUVs, based on a design fuzzy logic and PI control algorithms to stabilize the robot movements in both vertical and horizontal directions.

- Development of different fuzzy-based codes in C and Matlab to drive the sensors as well as the microcontroller for proper robot operation.

The different hard and soft parts were successfully tested on land and in a pool (a 25mx10m swimming pool with suctions/vacuums to change the water pressure).

This work generated the following publications (4 journals, 2 conferences):

1. A. Jebelli *et al.*, "Fuzzy-based controller design for intelligent robot arm," *Int. Review of Mechanical Engineering*, Vol. 8, N^{o} 1, pp. 214-222, Jan. 2014.
2. A. Jebelli *et al.*, "Intelligent control of a small climbing robot," *Int. Review of Automatic Control*, Vol. 6, N^{o} 6, pp. 751-758, Nov. 2013.
3. A. Jebelli *et al.*, "Fuzzy-based system for efficient and cost-effective control of light power consumption in autonomous vehicles," *Int. Review of Automatic Control*, Vol. 6, N^{o} 4, pp. 489-493, July 2013.
4. A. Jebelli *et al.*, "Design and construction of an underwater robot based fuzzy logic controller," *Int. Review of Mechanical Engineering*, Vol. 7, N^{o} 1, pp. 147-153, Jan. 2013.
5. A. Jebelli *et al.*, "Fuzzy logic temperature controller for small robots," *IEEE Symp. Series on Computational Intelligence*, Singapore, Singapore, pp. 1-4, April 2013.
6. A. Jebelli *et al.*, "Development of sensors and microcontrollers for small temperature controller systems", *Int. Conf. on Robotics and Automation Engineering*, Kuala Lumpur, Malaysia, Oct. 2014.

1.6 Thesis Overview

This thesis is divided into five chapters. In the second chapter, underwater robots are discussed while the third chapter concerns the selection of the different sensors that can be connected to the control unit for proper robot operation. In the fourth chapter, the performance of the sensors and the microcontroller are investigated through various tests, both on land and in underwater, and their results discussed. Finally, the fifth chapter summarizes the work and points out the contributions and future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Underwater vehicles can be grouped into three major groups namely:

- Manned Submersible Vehicles (MSVs), capable to perform complex tasks with the help of human interventions and human intelligence;
- Remotely Operated Vehicles (ROVs), remotely controlled and corded vehicles to transfer power, sensor data, and control commands between the operators on the surface and the underwater vehicle;
- Autonomous Underwater Vehicles (AUVs), unmanned, tether-free vehicles powered by onboard energy sources like battery or solar-cells and equipped with

various navigation equipment and sensors such as inertial measurement units (IMU), pressure sensors, sonar sensors, and controlled by onboard computers to perform certain tasks [11]-[14] (Table 2.1).

Both ROVs and AUVs are effective in deep-sea monitoring. ROVs can be controlled in real time: an experienced pilot can control the vehicle while a scientist can provide him/her with mission-level directives. The pilot usually uses live videos provided by on-board cameras to monitor the vehicle [7]-[11].

However, by doing so, the pilot should keep in mind all information concerning the ocean environment and thus, should make complex integration.

On the other side, AUVs can overcome all these issues because they are more portable and with on-board power and intelligence control systems that could help AUV self-react properly to changes in the device and its environment, avoiding any harmful situation. Such on-board controllers, built with preprogrammed instructions, must be of high flexibility to safely and reliably operate (Table 2.2).

Also, since high levels of intelligence and independence are required for optimized performance, comprehensive advanced algorithms are necessary to explain/control the process of operations. With the continuous advance in control, navigation, artificial intelligence, material science, computer, sensor, and communication, AUVs have become a very attractive platform in various exploring areas [7]-[11] and numerous AUV prototypes have been proposed [7]-[14] under different configurations and with a large panel of embedded systems [7]-[14] (Table 2.3). Thus, new technologies in the field of AUV control need to be developed.

Note that, because of the huge pressure water exerted on vehicles in oceans at depths of 6,000 to 11,000 m, AUVs are usually torpedo-shaped including small pressure hulls made with composite materials for on-board electronics and batteries. Table 2.4 shows different shapes of fairings. It should be noted that the streamline fuselage has been retained for this application because of its low water penetration resistance.

Table 2.1: Development of AUVs [11]-[14].

Year	Vehicle	Purpose	Depth(m)
1990	UROV-2000	Bottom survey	2000
1990	No Name	Testbed precise control vehicle	10
1990	Musaku	Testbed precise control vehicle	10
1990	UUV (II)	Testbed	NA
1991	AROV	Search and mapping	NA
1992	AE1000	Cable inspection	1000
1992	Twin Burger	Testbed	50
1992	ALBAC	Water column	300
1992	MAV	Mine countermeasures	NA
1992	Doggie	Bottom/sub-bottom survey	6000
1992	Dolphin	Water characteristics monitoring	6000
1992	ABE	Bottom survey	6000
1992	Phoenix	Testbed	10
1992	ODIN	Testbed	30
1993	Ocean Voyager 11	Science mission	6000
1993	Odyssey 11	Science mission	6000
1993	ARUS	Bottom survey	NA
1993	ODAS	Survey	900
1993	Hugin	Survey	600
1993	Marius	Survey	600
1994	Large-D UUV	Military/tested	300
1994	OTTER	Testbed	1000
1994	Explorer	Pipeline inspection	1000
1995	ODIN 11	Shallow water	30
1995	RI	Bottom survey	400
1995	Autosub 1	Environmental monitoring	750
1996	Theseus	Survey under Arctic sea-ice	1000
1997	REMUS	Survey	150
1997	VORAM	Testbed	200

1998	Solar AUV	Testbed	N/A
1998	AUV-HMI	Testbed	N/A
1998	AMPS	Military	200
1998	SIRENE	Undersea shuttle	4000
1999	SAUVIM	Military/scientific intervention	6000
2000	REMUS 100	Military/academic applications	100
2001	HUGIN 1000	Military /survey	1000
2003	REMUS 600	Military/academic applications	600
2006	REMUS 3000	Environmental monitoring/survey	3000
2008	REMUS 6000	Mapping/Imaging	6000

Table 2.2: Potential applications of underwater robots [11]-[14].

Oceanography	• Seafloor mapping • Rapid response to oceanographic and geothermal events • Geological sampling
Environment	• Long term monitoring (e.g., hydrocarbon spills, radiation leakage, pollution) • Environmental remediation • Inspection of underwater structures, including pipelines, dams, etc.
Military	• Shallow water mine search and disposal • Submarine off-board sensors
Ocean mining and oil industry	• Ocean survey and resource assessment • Construction and maintenance of undersea structures
Other applications	• Ship hull inspection and ship tank internal inspection Nuclear power plant inspection Underwater communication & power cables installation and inspection Entertainment-underwater tours Fisheries-underwater ranger

Table 2.3: Configurations of some existing AUVs [11]-[14]. (Th: Thrusters)

AUV	Main CPU	Processor	Battery	Th	Sensory system
AE 1000 (1992)	VMEMC 68040/4M	3 DSP + image processor	Lead-acid	3	Camera; obstacle avoidance sonar; rate gyroscope; accelerometers; radio beacon depth meter …
Phoenix (1992)	GESPACMC 68030/2M		Lead-acid gel	6	Altitude sonar; ST1000 *, ST725 *; collision avoidance sonar; gyroscopes …
ABE (1992)	68CH11	T800 SAIL network	Lead-acid gel alkaline-Li	6	Avoidance sonar; gyroscopes; Fluxgate compass; magnetic heading; angular rate …
Ocean Voyager II (1993)	VMEMC 68030/8M	LONTalk network	Lead-acid Ag-Z	1	3 axis angle/rate; whisker sonar; speedometer; pressure sensor; altitude sonar …
Odyssey II MIT (1993)	MC68030/8M	MC68HC11 SAIL network	Ag-Z	1	Altimeter; temperature sensor; acoustic modem; obstacle avoidance sonar; PING ultrasonic sensor * …
OTTER MBARI (1994)	MVME167 (68040)	MVME167N DDS protocol	Ni-Cd	8	Fluxgate compass; 2-axis inclinometer; motion-pack, 3-axis angle/rate; pressure sensor; sharp sonic ranging and positioning system …
ODIN II UH (1995)	VMEMC 68040		Lead-acid	8	Pressure sensor; 3-axis angle/rate sensor …
REMUS-100 (2000–2001)	REMUS 100	ArduSat Payload Processor Module	Lithium ion	4	Acoustic Doppler current profiling; side-scan sonar; measure of conductivity, temperature, bathymetry, depth, heading, roll and pitch…

HUGIN 1000 (2001)	HISAS 1030	Payload Processor Module	Li-Polymer	3	Synthetic aperture sonar; interferometry sonar; side scan sonar; acoustic Doppler current profiler…
REMUS 600 (2003)	REMUS 600	ArduSat Payload Processor Module	Li-ion	1	Acoustic Doppler current profiler; inertial navigation unit; side scan sonar; measure of pressure, conductivity and temperature; GPS …
REMUS 3000 (2006)	REMUS 3000	ArduSat Payload Processor Module	Li-ion	1	Acoustic Doppler current profiler; inertial navigation unit; side scan sonar; measure of conductivity, temperature, pressure…
REMUS 6000 (2008)	REMUS 6000	ArduSat Payload Processor Module	Li-ion	2	Acoustic Doppler current profiler; inertial navigation unit; side scan sonar; measure of conductivity, temperature …

* *ST1000: sonar used in AUV positioning and to ensure vehicle stability.*
* *ST725 sonar used in progressive complex static environments to clear images.*
* *PING ultrasonic sensor used in distance measurement.*

Table 2.4: Comparison of various vehicles shapes [11].

Shape	Advantages	Disadvantages
Single sphere	Low weight/vol. (W/V) ratio, excellent for deep diving vehicles	Low optimum length/diameter(L/D) ratio
Cylinder	Ease of fabrication, high optimum vehicle L/D ratio	High W/V ratio, end closures
Saucer	Improved hydrodynamics in horizontal plane, ease of hovering in currents	Low controllability, limited to shallow depths
Egg	Good hydrodynamics, good W/V ratio	Difficult to design and fabricate

2.2 AUV Kinematics

AUVs' kinematic and dynamic models are not linear, thus making control design a difficult challenge [15]. When moving on a horizontal plane, AUVs present similar dynamic behavior to under actuated surface vessels. Underwater vehicle equations of motion are controlled by restoring, thruster, and hydrodynamic forcing functions, together with nonlinear and time-varying disturbances such as waves and currents. Dynamics of underwater robotic vehicles, including hydrodynamic parameter uncertainties, are highly nonlinear, coupled, and time varying. Several modeling and system identification techniques for underwater robotic vehicles have been proposed by researchers. Euler's equations of motion of a body show the kinematic of an AUV and essential physics namely, the actual three-dimensional rotation of a rigid body under the action of external forces [15]-[19].

2.3 Control Systems

Among the points involved in control underwater robots, we can list the following [20]-[22]:

- Highly nonlinear behaviors,
- Time-varying dynamic behavior of the robot,
- Uncertainties in hydrodynamic coefficients,
- Higher order and redundant structure when the manipulator is attached,
- Disturbances by ocean currents,
- Changes in the centers of the gravity and buoyancy due to the manipulator motion which also disturbs the robot's main body.

The set-up of the control gains during operation in water is complex. Therefore, a robot control system that can be adjusted automatically is a suitable solution to consider. The complexity and implementation of a control system may increase/decrease due to changes in the dynamics of the robot and its environment. In fact, linear controllers cannot perform well because of system changes during the AUV operation.

Various advanced underwater robot control systems have been proposed in the literature. While many underwater remotely operated vehicles (ROVs) have mechanical manipulators, most AUVs do not have manipulators. There are indeed few papers about the coordinated motion of the vehicle and manipulator [10], [23]-[38].

Mahesh *et al.* [23] have developed a coordinated control scheme using a discrete-time approximation of the dynamic model of underwater robotic systems, which controls the vehicle and compensates for end-effectors errors resulting from motion of the vehicle.

Yoerger and Slotine [39] proposed a Sliding Mode Controller (SMC) for an underwater vehicle to control trajectory. A fault tolerant system consists of three areas: fault detection, fault isolation, and fault accommodation. The fault detection process is a high-level function that monitors the robot's overall systems - both hardware and software for any signals that exceed any pre-set tolerance or measured sensor values. Several methods for fault-tolerant control of autonomous underwater robots have been discussed in the literature. Canudas-de-Wit *et al*. [27] designed a robust nonlinear control for a vehicle system to compensate for the coupling effects due to an onboard robot arm. Yuh [40] and Choi and Yuh [41] developed and implemented a new Multiple Input Multiple Output (MIMO) adaptive controller using bound estimation for underwater robotic systems and experimented with the Omni Directional Intelligent Navigator (ODIN) control system.

However, sliding mode control limits the system states inside a specific subspace of the entire state space and makes them asymptotically convergent to their balance point [30]. A raw estimation of the system parameters and an estimation of the system uncertainty for the switching surface design and variable-structure control law design can be then used. When a generalized disturbance makes the system state exceeding the sliding mode tolerance layer, the exceeding value is used to update the nonlinear model variables for time-optimal switching surface design, and uses fuzzy logic to configure this surface [30]. The concepts of robust/optimal control are mathematical computation of variations, while direct usage of optimal control is not possible because of lack of reliable model of AUV systems [30]-[41].

Adaptive control can revise gain based on the changes in the process dynamics and the disturbances [30]. However, adaptive control may not be reliable when the dynamics change more rapidly than its ability to adapt.

Neural networks (NN) attracted many researchers because they can achieve nonlinear mapping [30]. Application of NN is effective since recognition of dynamics of the controlled system is not necessary. This makes NN suitable for underwater vehicle control. However, one of the disadvantages of NN is that there are no mathematical calculations. Therefore, use of mentioned model and direct learning are considered in application of NN for control purposes. In the former approach, generally, the forward model will be educated by the output error or state error and then used for gain derivation, while in the latter approach, for training the network, the Back Propagation (BP) algorithm is used but more time will be needed and several variables should be defined, so not suitable for real time control [30]-[41].

Another direction will be to consider fuzzy logic [42]. Any real continuous function over a compact set can be approximated to any degree of accuracy by a fuzzy inference system. Researchers have been using fuzzy logic to form a smooth approximation of a nonlinear mapping from the system input space to the system output space to make a suitable nonlinear system control. DeBitetto [43] investigated a 14-rule fuzzy logic controller for the depth control of an AUV. Tsukamoto *et al*. [44] experimentally implemented four model-free control systems for the position and velocity control of a single thruster system: on-line neural net controller, off-line neural net controller, fuzzy control, and non-regression based adaptive control [22].

However, determining the linguistic rules and the membership functions requires experimental data [30].

2.4 Navigation and Sensors

Vehicle autonomy development is of some restrictions because of the sensory system. Different sensors are needed for various works like acoustic imaging, Doppler, sonar inertial system, optical, and laser scanners for inspection; gyroscope for navigation; magnetometer, laser scanner, magnetic scanner, and chemical scanner for recovery; and force, tactile, and proximity sensors for construction. Blidberg and Jalbert [45] described mission and system sensors, and reviewed current navigation sensors and sonar imaging sensors.

The vehicle's sensors can be divided into three groups [46], [47]:

- Navigation sensors, for sensing the motion of the vehicle.
- Mission sensors, for sensing the operating environment.
- System sensors, for vehicle diagnostics.

Various sensors are often needed for the same task. For instance, information concerning the objects and local land around the vehicle can be obtained via a combination of sonar imaging, laser triangulation and optical imaging. Laser triangulation can provide data at a slower rate but with the additional capability of operating in turbid water. This resulting sensor fusion problem must be handled by the intelligent system. Another issue is in *x*-*y* position sensing because there are no internal system sensors for the *x*-*y* vehicle position [48], [49]. Some viewpoints used in vehicles are acoustic long baseline (LBL), short baseline (SBL), or ultra-short baseline (USBL) methods requiring external transponders. Due to signal attenuation with frequency, temperature, and distance with acoustic beacons are often are not available because they are expensive [11], [22], [50].

2.5 Communications

The most common approach for remotely operated underwater vehicle ROV communications uses an umbilical line with coaxial cables or fiber optics. This tether supplies duplex communications. While coaxial cables would be effective for simple operations with limited data transmission, fiber optic cables can transmit more data with less electromagnetic interference and are lighter, thinner cables. This is important since cables cause substantial drag and often become snagged. About ten percent of ROVs are lost because of broken tethers. Research and development of untethered autonomous vehicles is needed but communicating with AUVs presents formidable challenges. The main approach today for through-water transmission involves acoustics in which transducers convert electrical energy into sound waves. Since the ocean rapidly weakens the acoustic energy as the frequency is increased, relatively low frequencies are desirable for longer range communications. But at very low frequencies, the required transducer size is impractically large and the data rates are lower. The speed and direction of sound signals vary depending on surface waves, temperature, tides, and currents [22], [46], [51]-[53].

2.6 Important Principles in Fluid Mechanic for Submarines

Moving an item into a fluid leads to distribution of molecules. A relationship between air and solid objects is then created. The object shape, speed and volume will determine the size of forces. Movement can be divided into two components: frictional drag and pressure drag. Frictional drag comes from friction between the fluid and the surfaces over which it is flowing. Pressure drag comes from the eddying motions that are set up in the fluid by the passage of the body. When the drag is dominated by viscous drag, the body is streamlined, and when it is dominated by pressure drag, the body is bluff. A streamlined body looks like a fish, or an airfoil at small angles of attack, whereas a bluff body looks like a brick, a cylinder, or airfoil at large angles of attack. For a given frontal area and velocity, a streamlined body will always have a lower resistance than a bluff body [54], [55].

2.7 Fuzzy Logic Control

2.7.1 Introduction

Since their introduction in 1965, fuzzy sets and fuzzy logic have found applications in a wide variety of disciplines, such as automatic control in which these techniques have received considerable attention, not only from the scientific community but also from industry [56]-[58]. While most of the early design methods for fuzzy control were based on heuristic considerations [59], recent research has focused on the development of model-based fuzzy control techniques [60]-[62]. In the model-based approach, a fuzzy model is first developed to approximate the behavior of a complex process to be controlled. Based on this model, a controller can be designed.

Fuzzy set techniques have been recognized as a powerful tool for the development of models for systems that are not amenable to conventional modelling approaches due to lack of precise/formal knowledge about the system and/or strongly nonlinear behavior or time varying characteristics. The rule-based nature of fuzzy models allows the use of information expressed in the form of natural language statements. This makes the models transparent to qualitative interpretation and analysis. At the computational level, fuzzy models can be regarded as flexible mathematical structures, similar to neural networks that can approximate a large class of complex nonlinear systems to a desired degree of accuracy [63]-[65].

2.7.2 Proprieties of Fuzzy Logic

The advantages of using fuzzy logic can be summarized as follow [66], [67]:

- Fuzzy logic is tolerant of imprecise data. Everything is imprecise if you look closely enough, but more than that, most things are imprecise even on careful inspection. Fuzzy reasoning builds this understanding into the process rather than tacking it onto the end.
- Fuzzy logic can model nonlinear functions of arbitrary complexity.
- Fuzzy logic can be built on top of the experience of experts. In direct contrast to neural networks, which take training data and generate opaque, impenetrable models, fuzzy logic lets you rely on the experience of people who already understand your system.
- Fuzzy logic is conceptually easy to understand.
- The mathematical concepts behind fuzzy reasoning are very simple. Fuzzy logic is a more intuitive approach without the far-reaching complexity.
- Fuzzy logic is flexible.

- With any given system, it is easy to layer on more functionality without starting again from scratch.
- Fuzzy logic can be blended with conventional control techniques. Fuzzy systems don't necessarily replace conventional control methods. In many cases fuzzy systems augment them and simplify their implementation.
- Fuzzy logic is based on natural language. The basis for fuzzy logic is the basis for human communication. This observation underpins many of the other statements about fuzzy logic. Because fuzzy logic is built on the structures of qualitative description used in everyday language, fuzzy logic is easy to use.

However, even if this technique exhibits significant advantages, this method also has disadvantages which can be summarized as follow:

- Difficult to estimate membership function
- There are many ways of interpreting fuzzy rules, combining the outputs of several fuzzy rules and de-fuzzifying the output.

2.7.3 Fuzzy Design Process

A fuzzy controller is a system that applies linguistic rules to a given input vector to compute an output vector. The fuzzy design process for embedded controller can be executed in three main steps including fuzzification of the inputs, inferencing the rule based knowledge and defuzzification of the output. The three steps to compute the outputs can be illustrated in Figure 2.1.

2.7.4 Fuzzification

The fuzzification step transforms an input crisp value into a fuzzy value representing a degree of membership, called as alpha value. In this research, one approach of fuzzification was chosen namely, the Memory Oriented Fuzzification (MOF), in which the system computes a membership degree for each input off line and stores them in memory.

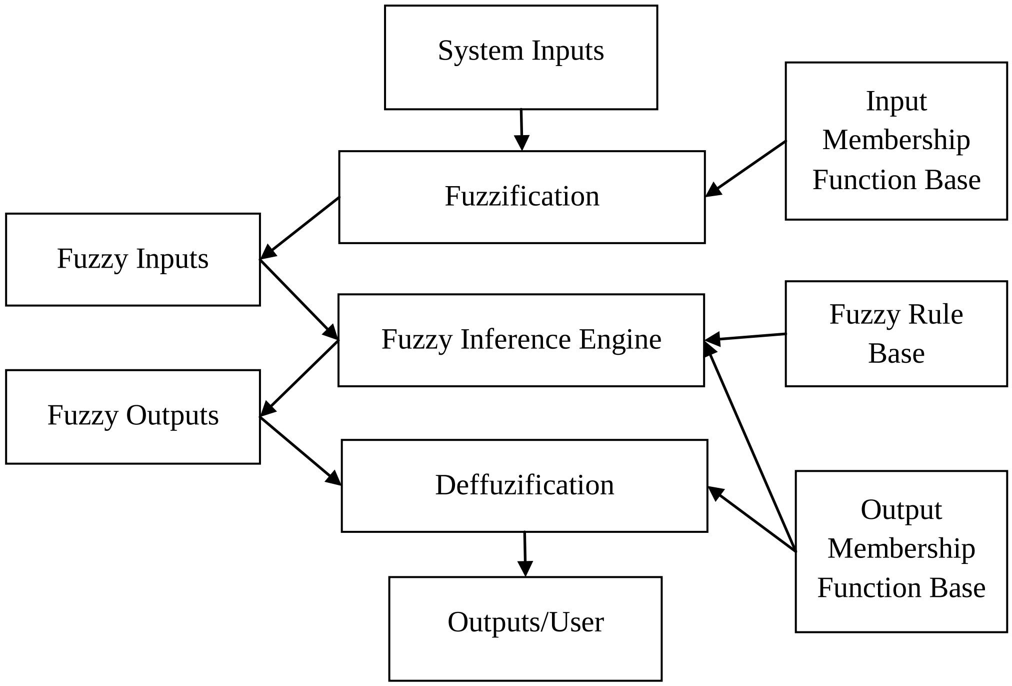

Figure 2.1: Fuzzy steps for output computation.

In this case, the membership function shapes do not influence the computational load of the algorithm. Despite this advantage for high-resolution computations, this approach requires large amounts of memory [66]-[69].

2.7.5 Membership Function

In designing a microprocessor fuzzy controller, the first step is to create the input and output membership function and the fuzzy rule table. Next, a code has to be generated to describe the processes of the controller. The code then needs to be compiled and downloaded into the targeted processor. When designing the number of membership functions for an input variable, labels must initially be determined for the membership functions. The numbers of labels correspond to the number of regions that the universe should be divided, such that each label describes a region of behavior.

A scope must be assigned to each membership function that numerically identifies the range of input values that correspond to a label. The shape of the membership function should be representative of the variable. However the computing resources available also restrict this shape. Complicated shapes require more complex descriptive equations or large lookup tables. In general, triangular shapes and singletons are preferred [70], [71]. Both will be used in the present work. They can be represented by point-slope format. Singletons require one byte for description and triangular three bytes; two point locations on the variable axis and one slope or grade values (Figure 2.2) such as [72]:

$$T(u;a,b,c)=\begin{cases}0 & \text{for } u<a\\ (u-a)/(b-a) & \text{for } a\le u\le b\\ (c-u)/(c-b) & \text{for } b\le u\le c\\ 0 & \text{for } u>c\end{cases} \tag{2.1}$$

where a, b and c represent the x coordinates of the three vertices of $\mu_A(x)$ in a fuzzy set A (a represents the lower boundary and c the upper boundary with membership degrees equal to zero, b is the centre with membership degree equals to 1).

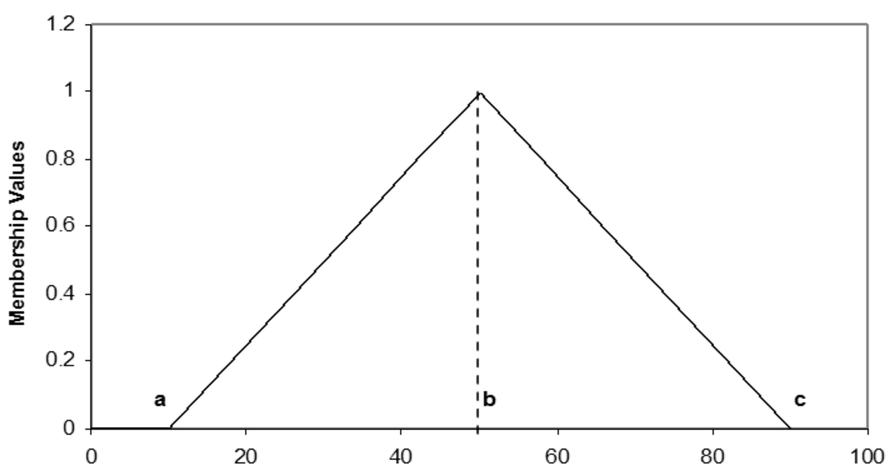

Figure 2.2. Triangular membership Function

2.7.6 Fuzzy Sets and Fuzzy Systems

A static or dynamic system, which makes use of fuzzy sets or fuzzy logic and of the corresponding mathematical framework, is called a fuzzy system. Most common are fuzzy systems defined by means of if-then rules with fuzzy predicates. Fuzzy systems can serve different purposes, such as modelling, data analysis, prediction or control. Historically, automatic control was among the first technical application domains of fuzzy systems [59], [73]. The basic idea of a fuzzy logic controller (FLC) is to formulate the control strategy of a human operator, which can be represented as a collection of if-then rules, in a way tractable for computers and for mathematical analysis. As an example, consider a control rule from a handbook for cement kiln operators [74], [75].

If x is A *then* y is B (2.2)

The first part of the rule (x), called the *antecedent*, specifies the conditions under which the rule holds, while the second part (y), called the *consequent*, prescribes the corresponding control action. These terms represent an approximate, quantized, the magnitude of different variables involved. When the if-then rules are to be represented in a form tractable for computers, one needs to define the linguistic terms, the logical connectives that operate on them, and the representation of the if-then relations. These concepts are explained below [76]. In computer implementations, fuzzy sets are usually represented in two methods - on discrete domains as a list of ordered pairs:

$$\{xi, \mu(xi)\}, xi \in X \quad (2.3)$$

and on continuous domains as an analytical formula of the membership function, e.g.

$$\mu(xi) = \frac{1}{1+x^2}, x \in \mathbb{R} \quad (2.4)$$

2.7.7 Fuzzy Set Operations

To use fuzzy rules, one needs to combine fuzzy sets using logical connectives such as AND (conjunction), OR (disjunction) or NOT (complement). Thus, the logical connectives from conventional Boolean logic have been extended to their fuzzy equivalents.

Operations for Function-theoretic terms: consider 2 sets: A and B, on the universe X.

Union:

$$A \cup B \rightarrow \chi_{A \cup B}(x) = \chi_A \vee \chi_B = \max(\chi_A, \chi_B) \quad (2.5)$$

Intersection:

$$A \cap B \rightarrow \chi_{A \cap B}(x) = \chi_A \wedge \chi_B = \min(\chi_A, \chi_B) \quad (2.6)$$

Complement:

$$\overline{A} \rightarrow \chi_{\overline{A}}(x) = 1 - \chi_A(x) \quad (2.7)$$

The use of fuzzy sets provides a basis for the systematic manipulation of vague and imprecise concepts using fuzzy set operations performed by manipulating the membership functions.

Equality: Two fuzzy sets A and B are equal if they are defined on the same universe and the membership function is the same for both, that is:

$$A = B \quad if \;\; \mu_A(u) = \mu_B(u) \quad \forall\, u \in U \quad (2.8)$$

Union: The union of two fuzzy sets A and B is the fuzzy set whose membership function is given by:

$$\mu_{A \cup B}(u) = \max\{\mu_A(u), \mu_B(u)\} \quad \forall\, u \in U \quad (2.9)$$

Intersection: The intersection of two fuzzy sets A and B is a fuzzy set whose membership function is given by:

$$\mu_{A \cap B}(u) = \min\{\mu_A(u), \mu_B(u)\} \quad \forall\, u \in U \quad (2.10)$$

Containment:

$$A \subseteq B \rightarrow \chi_A(x) \leq \chi_B(x) \tag{2.11}$$

If the fuzzy sets are defined in different domains, such as X and Y, the operator has to be applied in the Cartesian product space of X and Y, i.e., for all possible (x, y) pairs [70]-[72].

2.7.8 Defuzzification

Defuzzification typically involves weighting and combining a number of fuzzy sets resulting from the fuzzy inference process in a calculation, which gives a single crisp value for each output.

Among the variety of defuzzification methods proposed in the literature. Singleton fuzzy output is chosen due to its faster processing speed. It can be defined as [77], [78]:

$$Z^* = \frac{\sum_{t=1}^{n} B_n K_n}{\sum_{t=1}^{n} B_n} \tag{2.12}$$

where B_n is the weight of the rule which is fired and K_n the singleton output value for that specific rule.

2.7.9 Construction of the Fuzzy Rule Base

The relationships between the variables are represented by means of fuzzy IF - THEN rules and need to combine fuzzy sets using logical connectives such as AND (conjunction), OR (disjunction) or NOT (complement). For this purpose, the logical connectives from conventional Boolean logic have been extended to their fuzzy equivalents.

Depending on the consequent format, three main types of rule-based fuzzy models can be distinguished:

- Linguistic fuzzy model: both antecedent and consequent terms are fuzzy propositions.
- Fuzzy relational model: as generalization of the linguistic model, relation between antecedent and consequent terms is fuzzy.
- Takagi-Sugeno (TS) fuzzy model, the antecedent is a fuzzy proposition while the consequent is a crisp function.

Because of its ease of implementation, the Linguistic fuzzy model was used in this work. Analytical details on how the rule-base table is designed can be found in [75]-[79].

2.8 PID Control

A proportional-integral-derivative controller (PID controller) is a control loop feedback mechanism (controller) widely used in industrial control systems. A PID controller calculates an "error" value as the difference between a measured process variable and a desired set point. The controller attempts to minimize the error in outputs by adjusting the process control inputs. A PID controller will be called a PI, PD, P or I controller in the absence of the respective control actions. PI controllers are fairly common, since derivative action is sensitive to measurement noise, whereas the absence of an integral term may prevent the system from reaching its target value due to the control action [80].

The PID controller algorithm involves three separate constant parameters and is accordingly called three-term control; P stands for proportional, I for integral, and D for derivative [81]. It is described by:

$$u(t) = k\left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d(\tau) + T_d \frac{de(t)}{dt} \right) \tag{2.13}$$

where y is the measured process variable, r the reference variable, u the control signal and e the control error ($e = y_{sp} - y$). The reference variable is often called the set point. The control signal is thus a sum of three terms: the P-term (which is proportional to the error), the I-term (which is proportional to the integral of the error), and the D-term (which is proportional to the derivative of the error). The controller parameters are proportional gain K, integral time T_i, and derivative time T_d [81].

2.9 Conclusion

Because AUVs should self-react properly to changes in the device and its environment, their on-board intelligence control systems, such on-board controllers, must be of high flexibility to safely and reliably operate. However, to efficiently model the behavior of an AUV and reliably control, in real-time, its response to predicted and/or unpredicted changes is very challenging.

Because the goal of the present work is to develop a microcontroller to be implemented in AUVs, the next chapter will discuss about the design of such unit and its peripheral devices (like sensors) from both hardware and software perspectives.

CHAPTER 3

HARDWARE AND SOFTWARE DESIGN

3.1 Introduction

In this chapter, we will discuss the design of the different parts of the underwater robot, which can be divided into three sections: mechanical, electrical and programming. This will help better understanding the environment the microcontroller will have to manage and therefore, how it should perform. Figure 3.1 shows the block diagram of the robot development process.

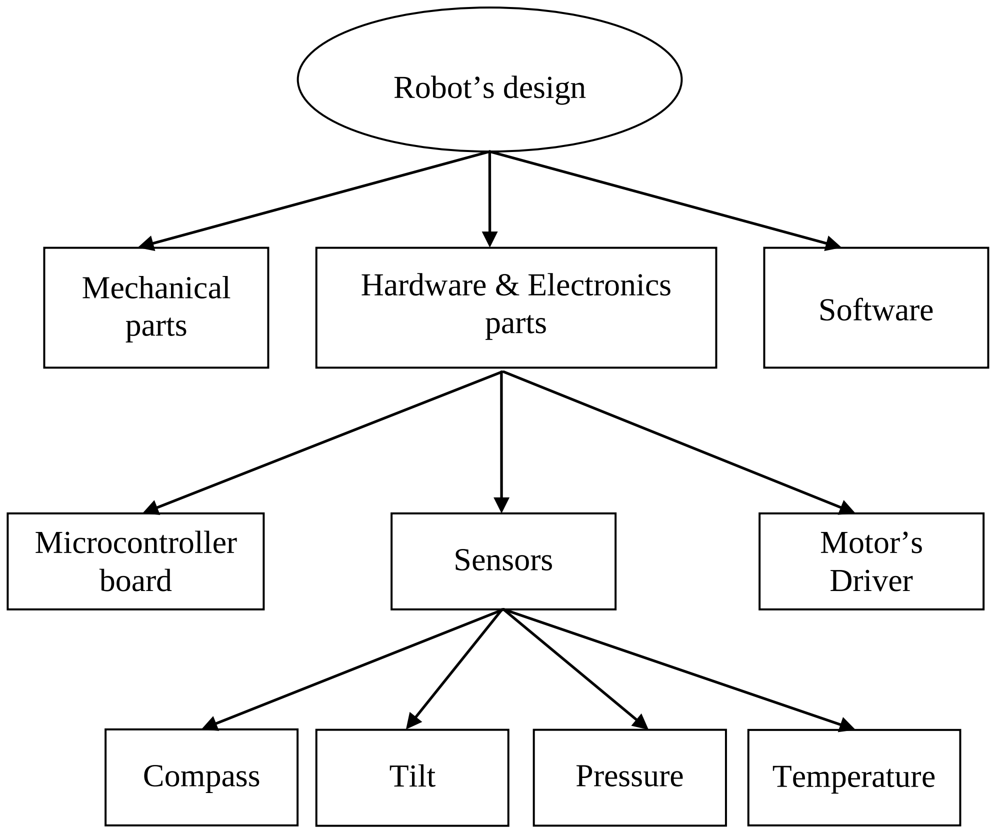

Figure 3.1: Robot: block diagram of the design process

3.2 Mechanical and Electronic Components

3.2.1 Robot Body Design

As for the body of the robot, we selected the streamline fuselage. In fact, the streamlined body has a lower resistance than the bluff body (when the drag is primarily a viscous drag, the body is streamlined, and when the drag force is primarily a pressure drag, the body is called a bluff body, Figure 3.2).

The traction power produced by an object depends directly on its shape and the way it moves in fluids (elasticity coefficient). According to the traction law, the acceleration is inversely proportional to the medium resistance.

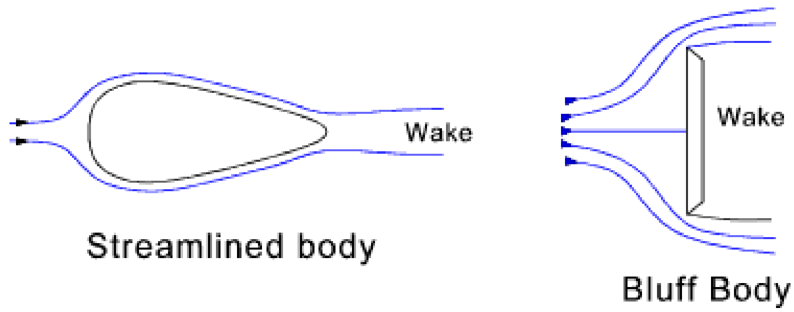

Figure 3.2: Streamlined body and Bluff body [82].

It means that the lower velocity is equal to lower resistance.

$$F_D = 0.5.\rho.\ v^2.C_D.A \tag{3.1}$$

In this equation, F_D is the drag force, ρ is the density of the fluid, v is the speed of the object relative to the fluid, A is the area, and C_D is the drag coefficient [83]. Then, as illustration, with a maximum speed of 0.1 m/s, the change and deviation in water pressure's path and its behavior can be seen Figure 3.3.

The calculation of the resistance of the robot's structural components was done using the Cosmos Work software. The robot's real isometric design is shown in Figure 3.4. The industrial plan has been modeled using SolidWork software and based on accurate calculation of the entire robot's pieces.

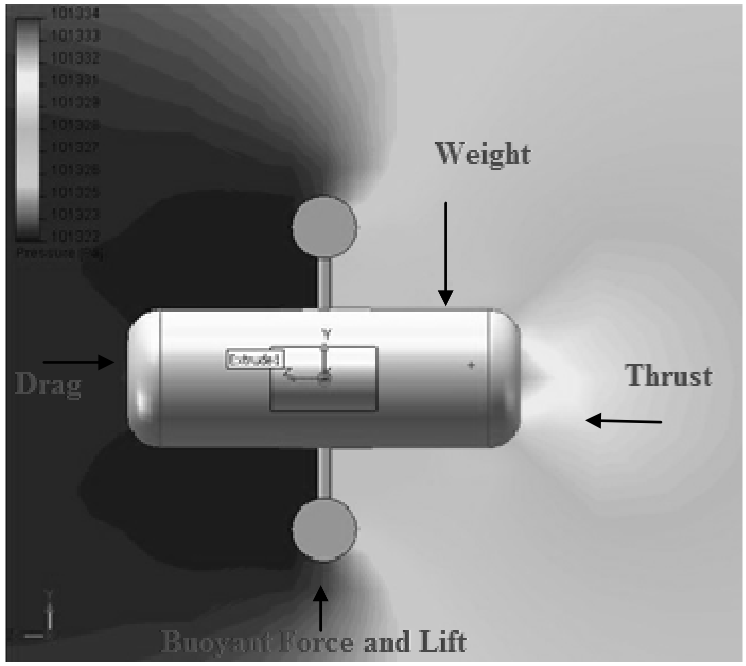

Figure 3.3: Variation of pressure of water

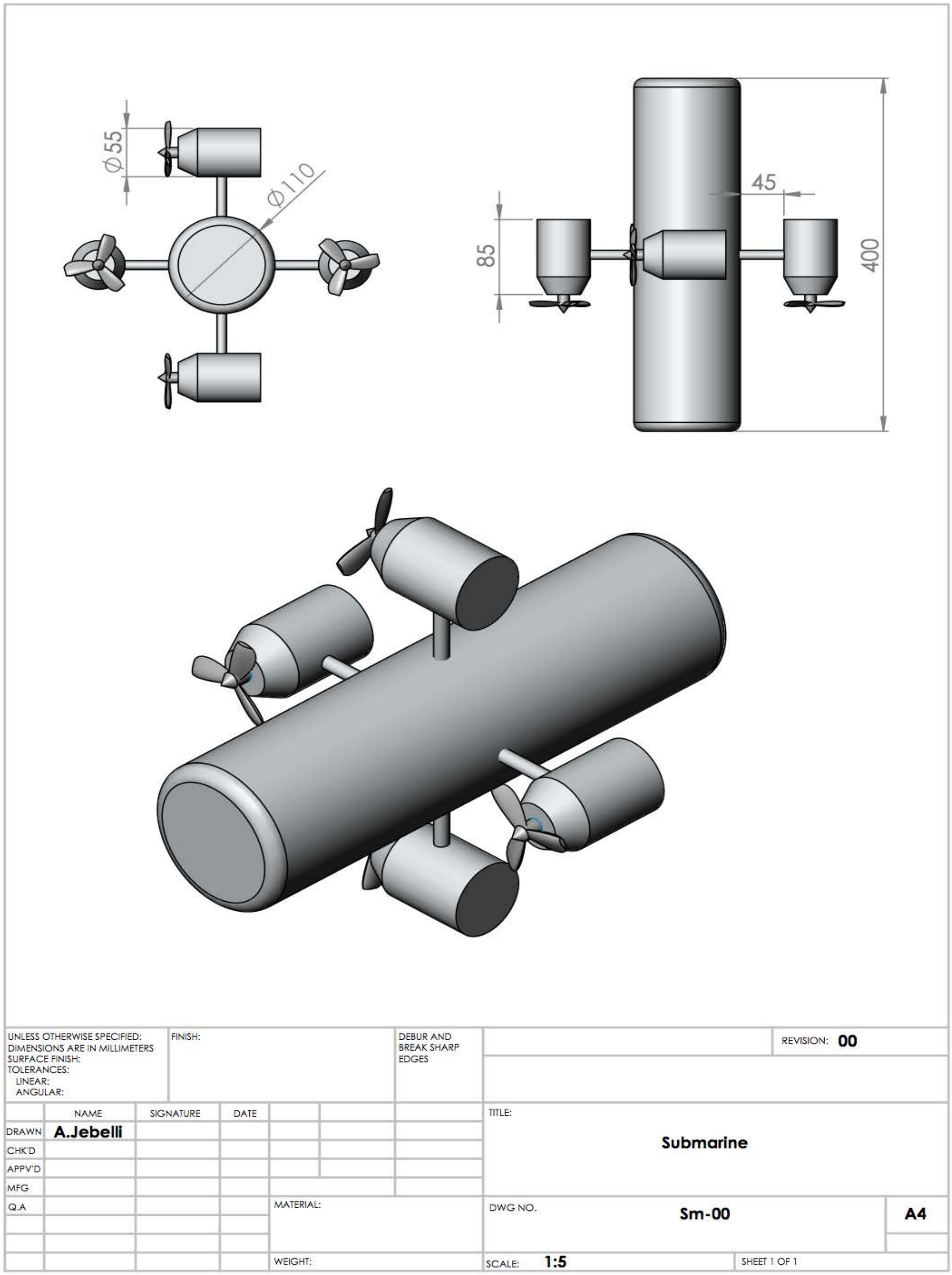

Figure 3.4: The isometric scheme of the robot.

3.2.2 Microcontroller - Schematic

The type of microcontroller used in this work is a P89V51RD2 with 64 KB flash (80C51 controller with 1024-byte RAM). The P89V51RD2 priority allows using it in both state, X2 mode (6 clocks on machine cycles to reach twice the throughput at the same clock frequency) and the conventional 80C51 clock rate (12 clocks on machine cycle). The flash state supports both parallel programming (increasing the programming speed as well as reducing its expense, thus saving additional time) and serial programming (which allows the device to be programmed) [84], [85]. There are many other microcontrollers and microcontroller platforms available for physical computing but the above offers some advantages over other available microcontrollers (Table 3.1):

- 80C51 Central Processing Unit
- 5 V operating voltage from 0 to 40 MHz
- 64 KB of on-chip Flash program memory with ISP (In-System programming) and IAP (In-Application programming)
- Supports 12-clock (default) or 6-clock mode selection via software or ISP
- SPI (Serial Peripheral Interface) and enhanced UART
- PCA (Programmable Counter Array) with PWM and Capture/Compare functions
- Four 8-bit I/O ports with three high-current Port 1 pins (16 mA each)
- Three 16-bit timers/counters
- Programmable Watchdog timer (WDT)
- Eight interrupt sources with four priority levels
- Second DPTR register
- Low EMI mode (ALE inhibit)
- TTL- and CMOS-compatible logic levels
- Low power modes

Table 3.1: 80C51 8-bit Flash microcontroller family [86].

Type number	Package		
	Name	Description	Version
P89V51RB2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89V51RB2FN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1
P89V51RB2BBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89V51RC2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89V51RC2FBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89V51RC2FN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1
P89V51RD2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89V51RD2FBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89V51RD2BN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1
P89V51RD2FN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1

Furthermore, the microcontroller should be:

- self-programmable, do not need a separate programming,
- simple to design,

This microcontroller board, which schematic is shown in Figure 3.5, is used for programming and communication with the microcontroller. Its design is shown in Figure 3.6. The microcontroller board is connected to the computer using the RS-232 standard and all peripherals are connected to the microcontroller as reported in Table 3.2. The programming part was achieved by compiling C programs and using Flash Magic[3] programming tool for programming the microcontroller.

[3]http://www.flashmagictool.com/

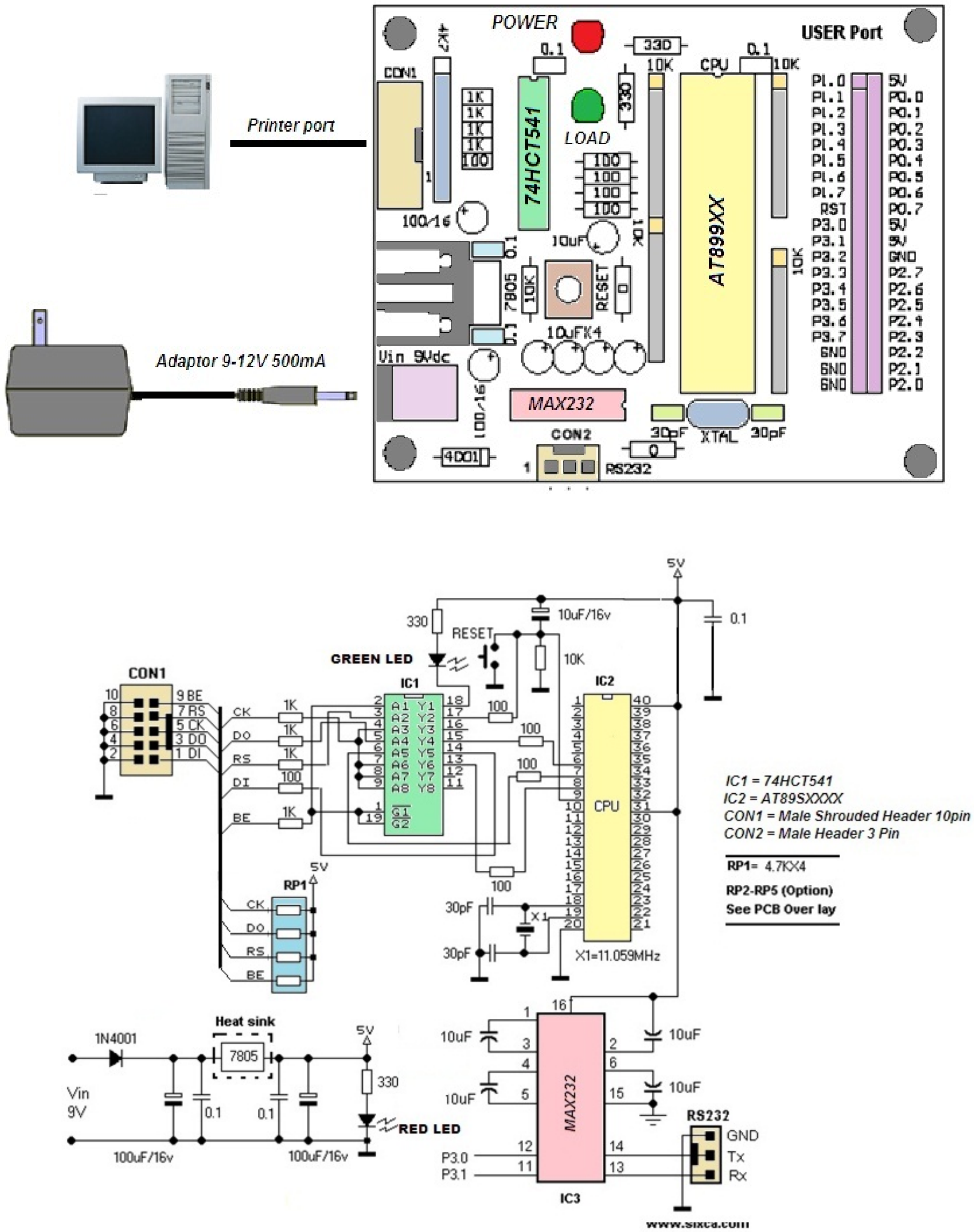

Figure 3.5: Electronic Circuit Schematics - Microcontroller based schematics [87].

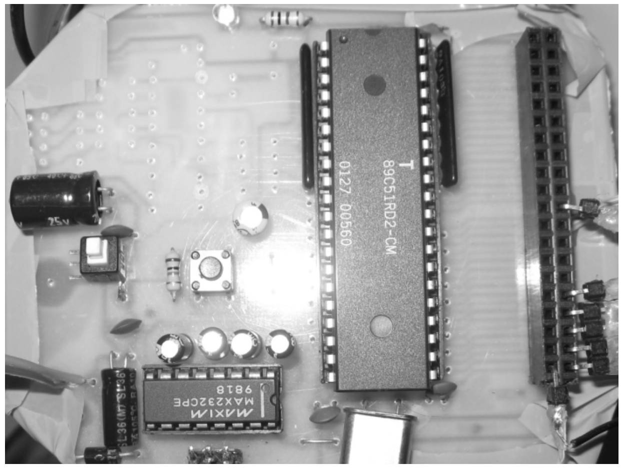

Figure 3.6: P89V51RD2: completed microcontroller board.

Table 3.2: Microcontroller connection pins.

Sensor and indicators	Port	Pin descriptions
Compass	2	P2.0 = Reset P2.1 = SS,P/C P2.2 = EOC P2.3 = SDO P2.4 = SCLK
Tilt	2	P2.7 = Duty cycle
Pressure	3	P3.4 = CS P3.5 = DIN P3.6 = CLK P3.7 = DOUT
7-Segment	0	P0.0 = D0P0.1 = D1 P0.2 = D2P0.3 = D3
Motor-Driver	1	P2.1 = DIRECTION-MOTOR 1 P2.1 = DIRECTION-MOTOR 2 P2.1 = DIRECTION-MOTOR 3 P2.1 = DIRECTION-MOTOR 4 P2.1 = PWM-CHANNEL 1 P2.1 = PWM-CHANNEL 2 P2.1 = PWM-CHANNEL 3 P2.1 = PWM-CHANNEL 4

3.2.3 Microcontroller - Programming

In this section, we will detail the software part of the fuzzy-logic microcontroller.

In various control applications, fuzzy controllers are widely used; in one of the most common methods, called direct control, a fuzzy control feedbacks the system to compare the output with the input. This approach can be efficient in most cases, but when multiple signals are used in a system having multiple functions, it might not be reliable. Thus, PID fuzzy controllers can be the alternative.

In Figure 3.7, to compensate for the fuzzy system and to obtain more accuracy in the performance, the feedback has been eliminated.

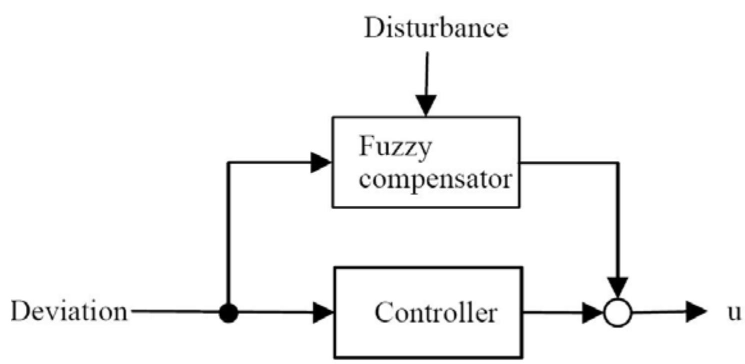

Figure 3.7: Compensated disturbance.

This design shows that regardless of the internal interferences, it can be stated that a PID controller can coordinate between linear and nonlinear controlling functions; the fuzzy controller can be seen as a complementary to the nonlinear controller.

Tilt sensor is a sensor that enables the fuzzy control operation. In our design, two motors have been used to facilitate the mobile's vertical motion. One of their duties is to provide the robot's balance and its movements towards depth. This sensor sends out the suitable fuzzy output orders by calculating the steep parallel to the vertical motors and the microcontroller along with the fuzzy program.

The robot performance will be determined based on the changes of the membership function and its regulations. The final results could be obtained after regulating these values. A first C program was written in Matlab. The following regulations were exerted for the fuzzy programs (Figure 3.8).

IF Tilt-Error is Low THEN PWM is High.

IF Tilt-Error is Low AND Tilt-Error is Zero THEN PWM is Medium.

IF Tilt-Error is Zero THEN PWM is Zero.

IF Tilt-Error is Low AND Tilt-Error is Zero THEN PWM is Medium.

IF Tilt-Error is High THEN PWM is High.

These values have been depicted in Figure 3.9. The PWM value will not be changed after reaching 25 degree and its value will be 260.

Figure 3.10 shows the microcontroller output values using fuzzy logic programming. The fuzzy output for -90 to -25 and also 90 to 25 degree is equal to 260.

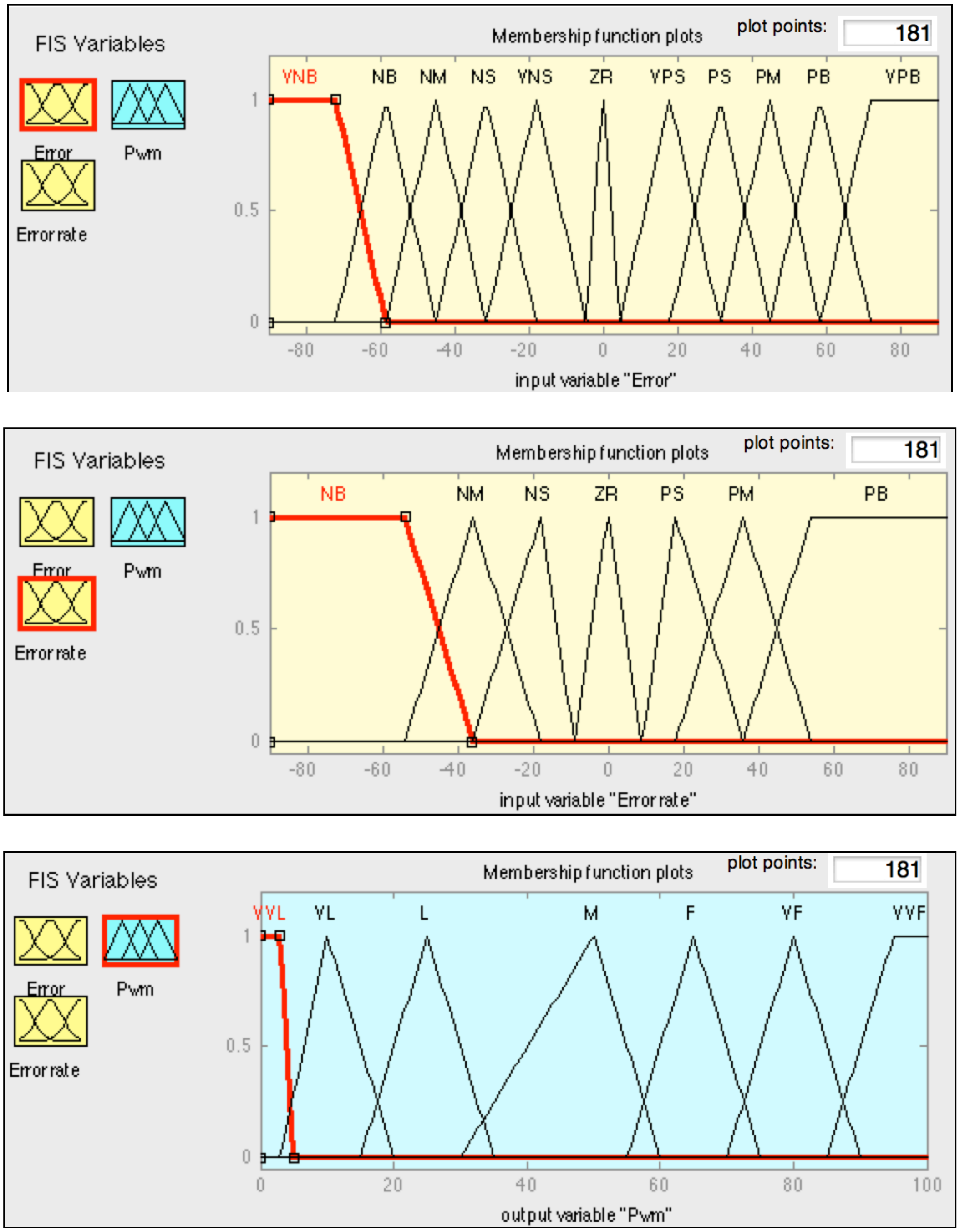

Figure 3.8: The initial fuzzy program using Matlab software.

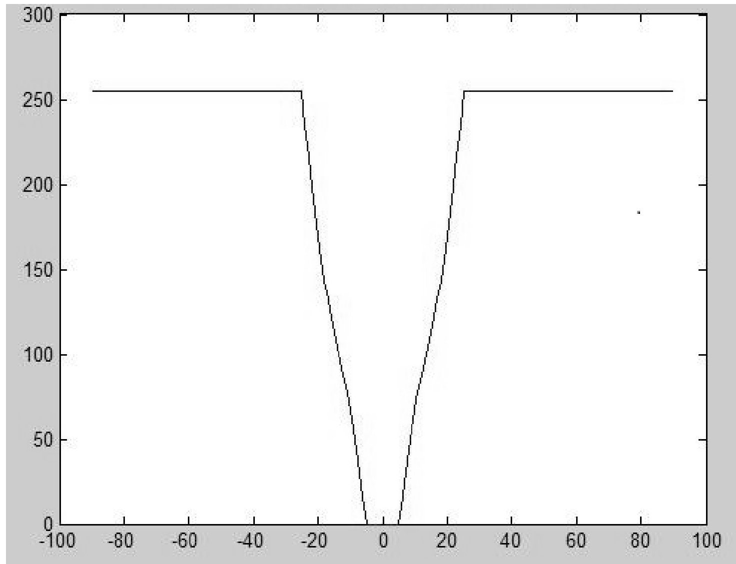

Figure 3.9: Plotting the output values of simulation using Matlab.

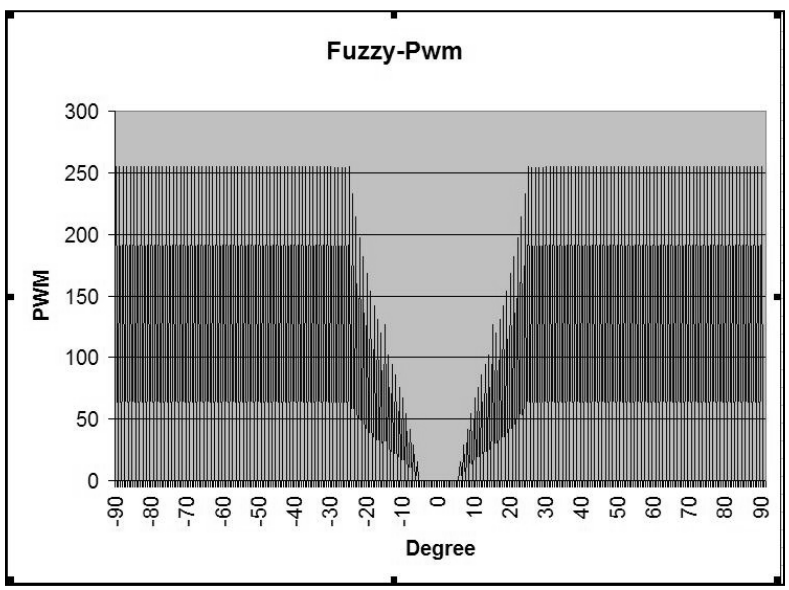

Figure 3.10: Plotting the output values of microcontroller.

3.3 Programming Microcontroller Using Tilt Sensor and Fuzzy PI Control

One of the issues of the fuzzy control in dynamic systems is that they cannot be forecasted. The balanced error is one of the main factors that needs time to reach the desirable situation. One of the ways is to use the fuzzy PI controller to reduce the risk in the situation of robot's balance.

The fuzzy program was written using the following instructions:

Fuzzy P rules:

IF Tilt-error is Positive Small THEN P-PWM is Positive Big.
IF Tilt-error is Positive Small AND Tilt-Error is Zero THEN P-PWM is Positive Medium.
IF Tilt-Error is Zero THEN P-PWM is Zero.
IF Tilt-Error is Positive Small AND Tilt-Error is Zero THEN P-PWM is Positive Medium.
IF Tilt-error is Positive Medium THEN P-PWM is Positive Big.
IF Tilt-Error is Positive High THEN P-PWM is Positive High.

Fuzzy I rules:

IF Integral is Positive Small THEN I-PWM is Positive Big.
IF Integral is Positive Small AND Tilt-Error is Zero THEN I-PWM is Positive Medium
IF Integral is Zero THEN I-PWM is Zero.
IF Integral is Positive Small AND Tilt-Error is Zero THEN I-PWM is Positive Medium.
IF Integral is Positive Big THEN I-PWM is Positive Big.

Table 3.3 shows the rule table, including seventy seven rules.

Table 3.3: The decision table by the fuzzy controller

E \ ΔE	VNB	NB	NM	NS	VNS	ZR	VPS	PS	PM	PB	VPB
NB	VVF	M	M	L	VVL	VVL	VVL	L	M	M	VVF
NM	VF	F	F	M	VVL	VVL	VVL	M	F	F	VF
NS	VF	VVF	VF	M	VL	VVL	VL	M	VF	VVF	VF
ZR	VVF	VF	F	M	VL	VVL	VL	M	F	VF	VVF
PS	VVF	VVF	VF	VF	M	VL	M	VF	F	VVF	VVF
PM	VVF	VVF	VF	VF	F	VL	F	VF	VF	VVF	VVF
PB	VVF	VVF	VVF	VF	VF	VL	VF	VF	VVF	VVF	VVF

The terms in the table are:

NB: negative big,

NM: negative medium,

NS: negative small,

ZR: zero,

PS: positive small,

PM: positive medium,

PB: positive big

The values for P and I membership functions are as below:

P: positive small = -60, zero = 0,

positive medium = 100, positive big = 160.

I: positive small = 40, positive big = 80.

The corresponding Fuzzy PI surface is plotted in Figure 3.11.

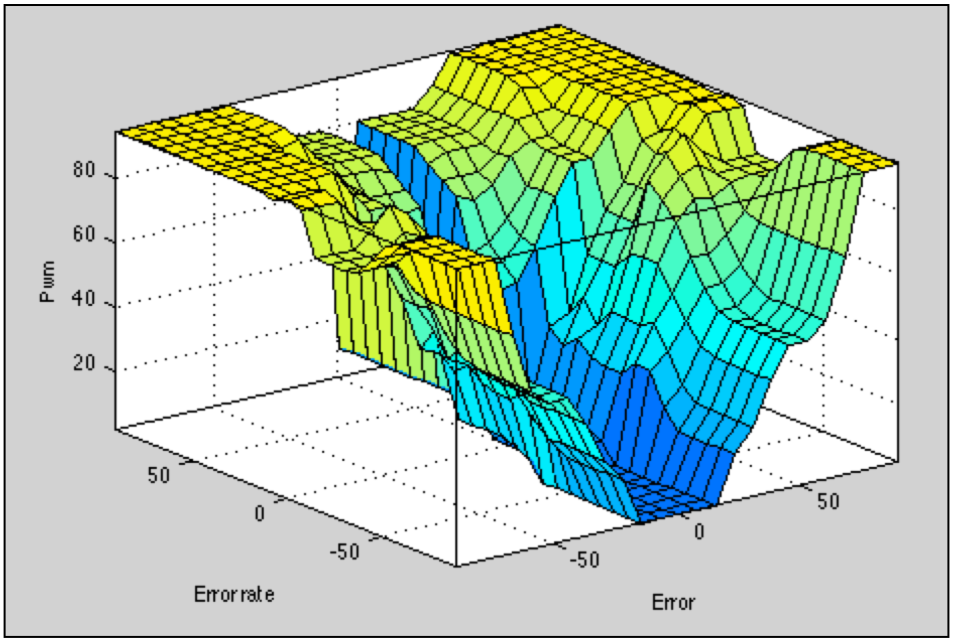

Figure 3.11: Fuzzy PI surface.

3.4 Programming Microcontroller Using Temperature Sensor and Fuzzy PI Control

One of the drawbacks in small AUVs is the heat generated by the motors and batteries. Therefore, to prevent damages to the electrical system and avoid degrading its performance, it is required to maintain a constant temperature inside the underwater robot. An intelligent fuzzy-based temperature control system was then designed (Figure 3.12 where E represents the Error and ΔE the Error rate).

The Fuzzy rules for this fuzzy program are as follow:

- E = Ts (Set Point Temperature) – Tc (Current Temperature)
    - IF E > 0 Turn On the Heater
    - IF E < 0 Turn On the Fan

- ΔE = |E(K)| - |E(K-1)|
    - IF ΔE > 0, Away from the desired temperature
    - IF ΔE < 0, Near the point
    - IF ΔE = 0, steady
- IF Error is high, increase the speed of the Fan
- IF Error is small and
    - ΔE ≤ 0, select low fan speed
    - IF ΔE > 0, select high fan speed

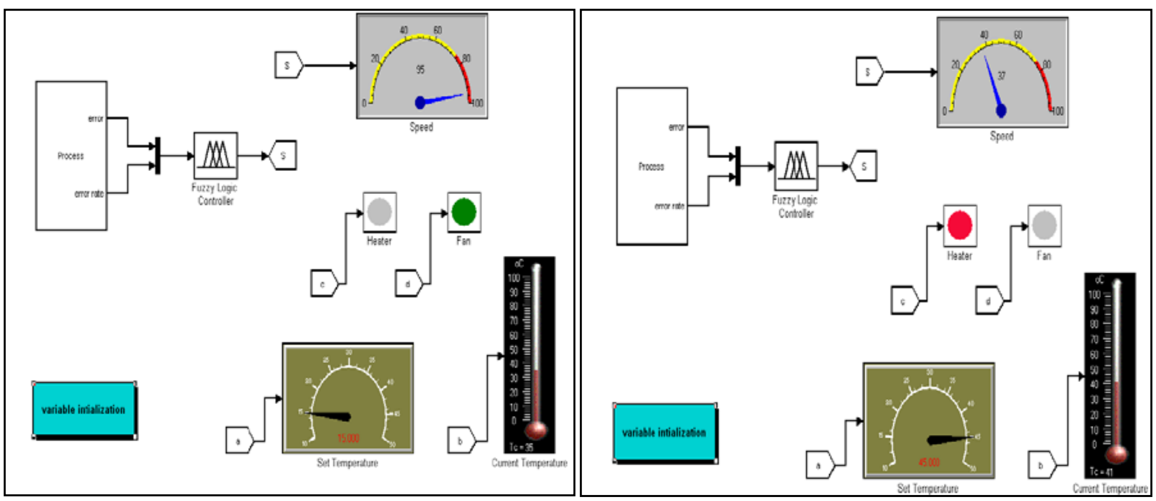

Figure 3.12: Temperature Controller Simulation (Left: The fan is working, Right: The heater is working)

When the robot is subject to a sudden increase of heat, it is able to change from a critical condition to a stable condition. This condition happens because of the high value of the PWM value in the output membership function.

As shown in Figure 3.13, the robot has been able to balance temperature rapidly, at about 78% of the fan speed, after a sudden increase of temperature by 13 °C. For a temperature change of 12°C, the speed of the fan was optimized to 72% (Figure 3.14), demonstrating the ability of our control system to adapt the system response to the operating conditions.

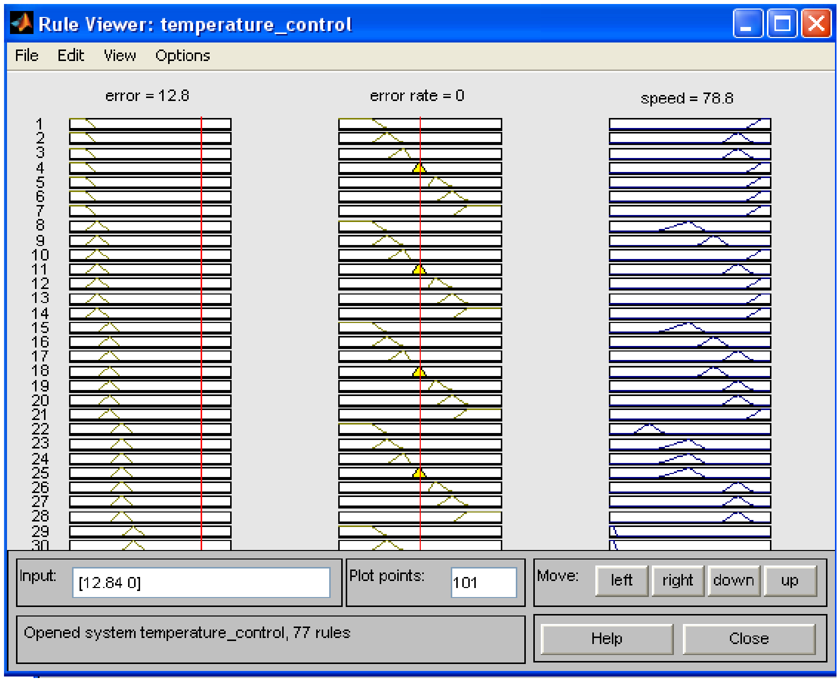

Figure 3.13: Temperature Controller Simulation for a 13°C change in temperature.

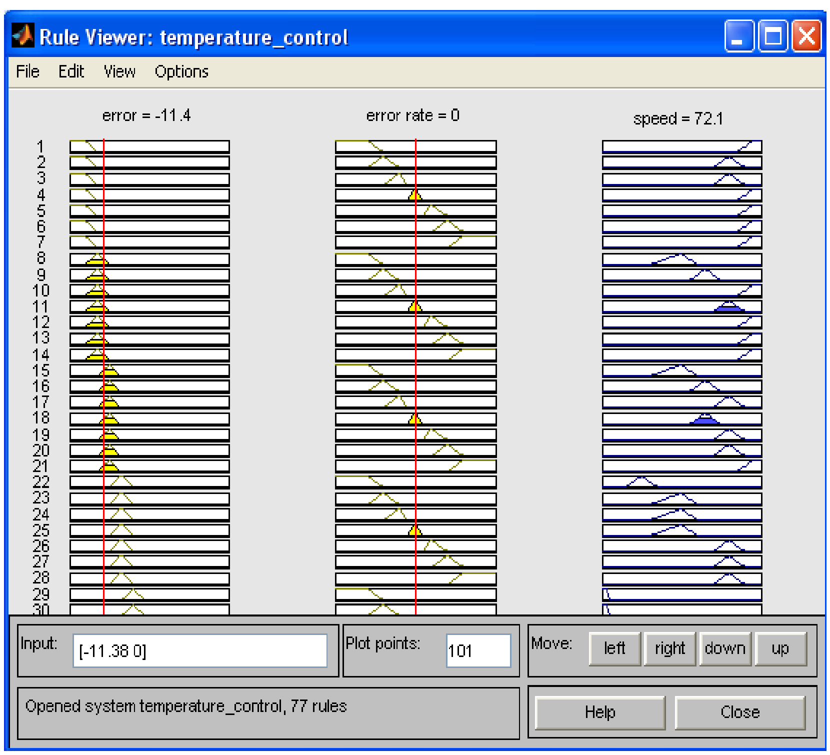

Figure 3.14: Temperature Controller Simulation for a 12°C change in temperature.

Also, when the temperature approaches its desired value, the error rate is reduced and the fan speed will start to decrease accordingly (Figure 3.15).

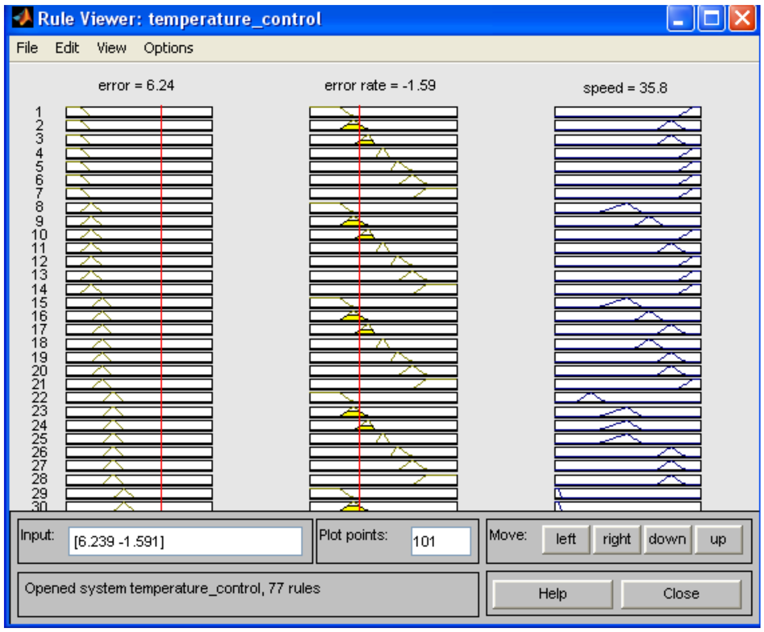

Figure 3.15: Temperature Controller Simulation when the temperature is close to its desired value.

3.5 Thrusters Motors

To assure adequate moving under water in both vertical and horizontal directions, the robot has been equipped with four motors. These motors should have sufficient power to move but also the smallest possible volume according to the shape and size of the robot as well as consuming minimum power and creating minimum noise and oscillations. According to these constraints, RS-380PH-4050 motors have been used with suitable blades as well as high torque and proper speed [88].

Their specifications are:

- Operating voltage: DC 2.4V-15V, Nominal 10.5V
- Speed under without load: 21900 r/min, 18570 at maximum efficiency
- Torque under 7V: 839g.cm / Torque under 12V: 959g.cm
- Current drawn: 3.9A at maximum efficiency

As shown in Figure 3.16, the motor has been installed in a cylindrical container smaller than the blades’ radius due to the increase in the blades’ operation. The blades are insulated properly using water resistant glue and attached to the robot’s fuselage by two metal bases. Two motors have been installed vertically to enable motion to the depth and two horizontally to enable motion towards the horizontal axis both on the surface and under water.

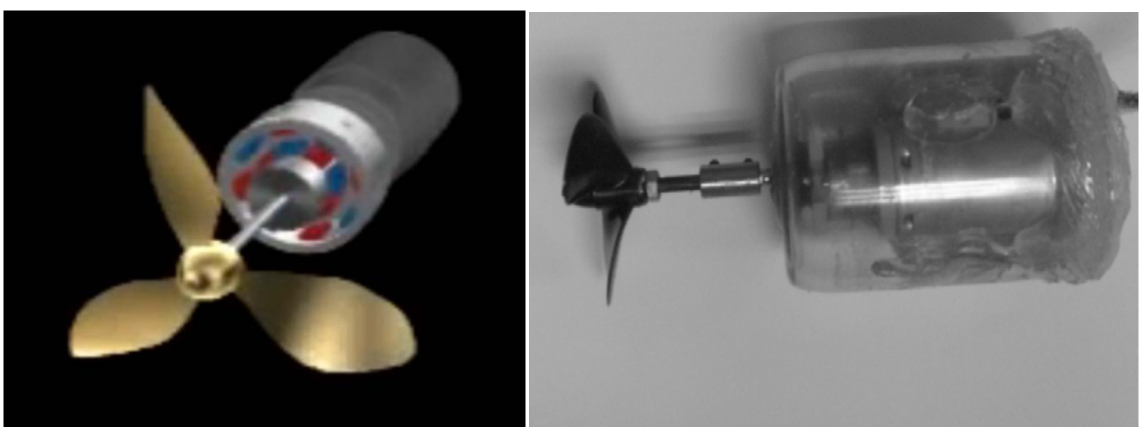

Figure 3.16: Thrusters.

One of the other characteristics of the DC motors is their requirement to perform in high currents by using a relay. To switch forward and backward on the path, the relay used on the system is an SRD with a performance speed of 10 ms and reaction time of 5 ms (Figure 3.17).

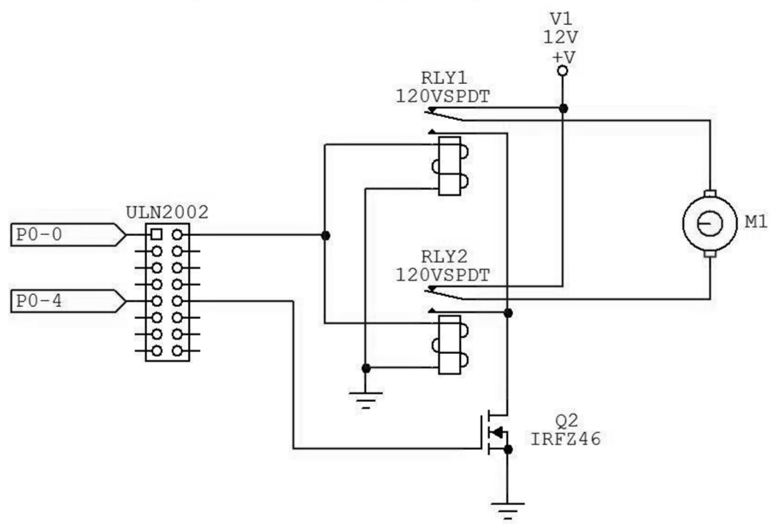

Figure 3.17: scheme of the relays.

3.6 Power Supply and 7-Segment Indicator

The robot's electrical power is provided via a 12V battery installed below the robot. This battery is cabled to a 5V regulator and attached to the robot. There is also a 7-segment display connected to the microcontroller to show its internal operations.

3.7 Sensors

3.7.1 Compass Sensor

The compass module was specifically designed for use in robots as aid to navigation. The aim is to produce a unique number to represent the direction the robot is taking. The retained compass uses a KMZ51 magnetic field sensor and requires a 5V power supply at a nominal current of 15mA.

There are two ways of getting the bearing from the module: through a PWM signal or via an I2C interface.

The PWM signal is a pulse width modulated signal with the positive width of the pulse representing the angle. The pulse width varies from 1mS (0°) to 36.99mS (359.9°) – in other words 100µS/° with a +1mS offset. The signal goes low for 65ms between pulses, so the cycle time is 102 ms (65ms + pulse width of 66ms). The pulse is generated by a 16 bit timer in the processor giving a 1µS (0.1°=10µS) resolution [89].The timing diagram is shown in Figure 3.18. For the I2C communication protocol, the compass has a 16 byte array of registers, some of which double up as 16 bit registers as reported in Table 3.4. The shape of the mounted board is shown in Figure 3.19.

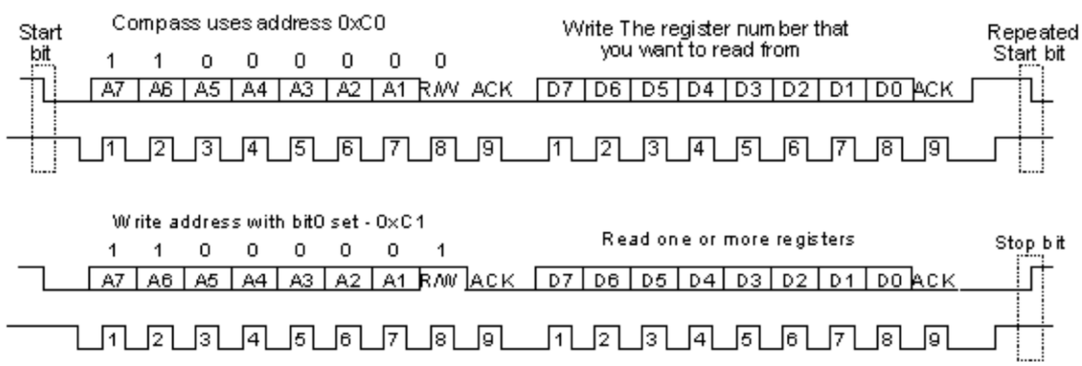

Figure 3.18: RAW Data Mode Format (Slave Mode only, 16-bit Data Format). [89]

Table 3.4: Connection pins for compass and microcontroller.

Register	**Function**
0	Software Revision Number
1	Compass Bearing as a byte, i.e. 0-255 for a full circle
2,3	Compass Bearing as a word, i.e. 0-3599 for a full circle, representing 0-359.9^o
4,5	Internal Test –Sensor1 difference signal - 16 bit signed word
6,7	Internal Test –Sensor2 difference signal - 16 bit signed word
8,9	Internal Test - Calibration value 1 - 16 bit signed word
10,11	Internal Test - Calibration value 2 - 16 bit signed word
12, 13	Unused - Read as Zero
14	Unused - Read as Undefined
15	Calibrate Command - Write 255 to perform calibration step. See text.

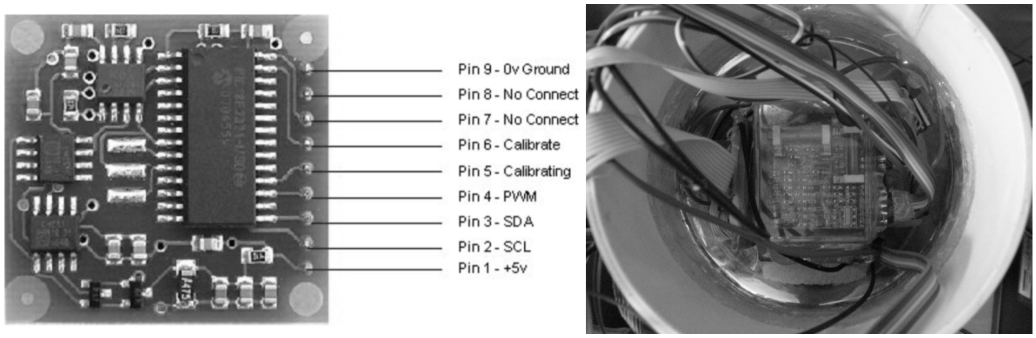

Figure 3.19: Complete Compass sensor.

3.7.2 Tilt Sensor

To measure the tilt of the AUV, we used the IC ADXL320 sensor. There exist many other tilt sensors available for physical computing, but the selected one offers some advantages over other sensors namely, small, low power (one of the critical design constraints in the present work) and dual-axis accelerometer with signal conditioned voltage outputs. It can measure both dynamic acceleration (vibration) and static acceleration (gravity) (Figure 3.20). The output signals are analog voltages that are proportional to acceleration. The accelerometer measures static acceleration forces, such as gravity, which allows it to be used as a tilt sensor: deflection of the structure is measured using a differential capacitor that consists of independent fixed plates and plates attached to the moving mass. The fixed plates are driven by 180° out-of-phase square waves [90]. Acceleration deflects the beam and unbalances the differential capacitor, resulting in an output square wave whose amplitude is proportional to acceleration. Phase-sensitive demodulation techniques are then used to rectify the signal and determine the direction of the acceleration. The acceleration-degree conversion curve, displayed in Figure 3.21, can be mathematically fitted by the following polynomial expression

$$y = 62.241x^5 - 6E\text{-}10x^4 - 41.569x^3 + 7E\text{-}10x^2 + 66.169x + 2E\text{-}09 \qquad (3.2)$$

The range of output is -90 to +90 degrees.

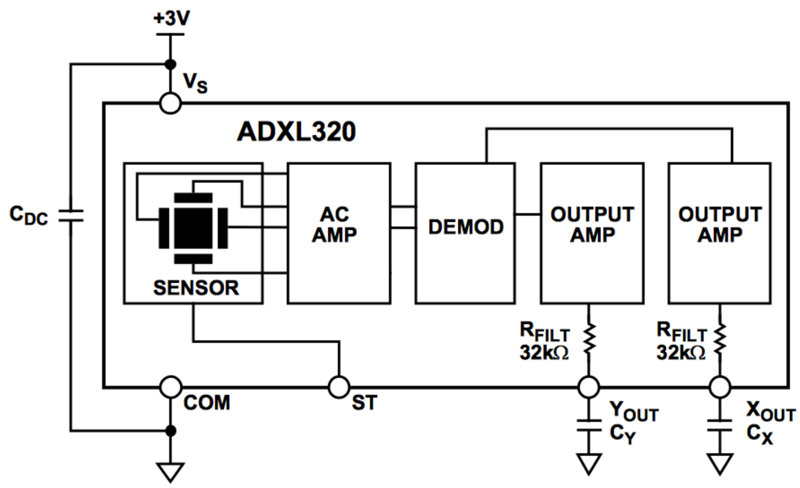

Figure 3.20: Accelerometer Functional block diagram [90].

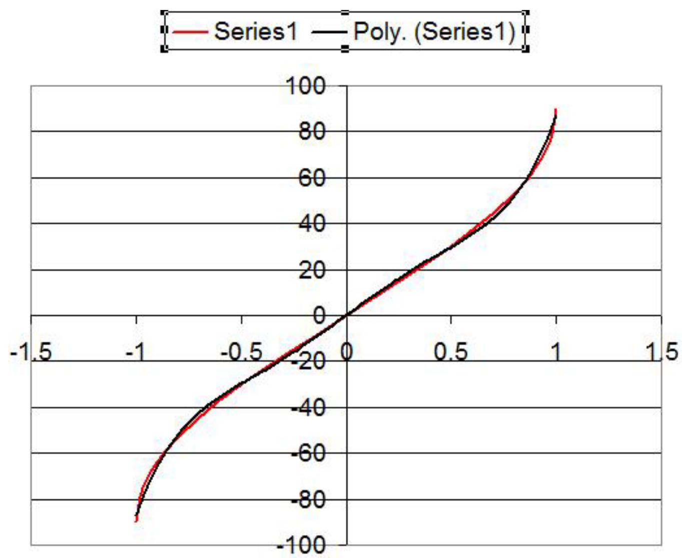

Figure 3.21: Conversion *g* values to degree values.

3.7.3 Pressure Sensor

To calculate the water depth we need to measure the pressure. In this work, since water resistant sensors are very expensive, we used a typical 26PC pressure sensor, which is mainly used to measure air pressure.

It presents many advantages such as [91]

- Cost effective sensor with temperature compensation and calibration
- Variety of gage pressure port configurations
- Calibrated Null and Span
- Ideal for wet/wet differential applications

Obviously, an air pressure sensor cannot accurately replace a water resistant sensor. However, since accuracy in water depth measurement is not a critical issue in this work, we opted for an air pressure sensor, knowing that the approximations introduced by such sensor replacement are acceptable. Since we know that the pressure of each 10 meters of water is equal to 1 atmosphere and 1 atmosphere is about 14.7 psi, we can roughly estimate that if the robot goes down in the water up to 8 meters, it would be 1 atmosphere higher than sea pressure plus 0.8 atmosphere for 7 meters of the water depth: 14.7 * 1.8 = 26.46 psi. So sensor needs to read up to approximately 26 psi.

To use this sensor, we added an LM2902 amplifier which consists of four operational amplifiers (OA) to increase the voltage gain. Thus, since the output signal is analog, we used the 8 bit ADC0838 series converter which works from 19mV to 5V. Hence, considering the maximum resolution to receive the minimum pressure of 0.1 psi, the ADC output voltage should be increased for every 10 cm depth.

3.7.4 Temperature Sensor

Many types of temperature sensors such as thermocouples, thermistors, infrareds …are available on the market to control the heating dissipation due to the motors and batteries. In this work, we retained the cost-effective LM35 as temperature sensor mainly due to its linear input-output relationship and ease of use (Figures 3.22 and 3.23). Furthermore, it is already scaled in ^{o}C while most temperature sensors are calibrated in °K. This prevents the user to subtract a large constant voltage from the output to obtain the convenient centigrade scaling.

It also provides a typical accuracy of ±1⁄4°C at room temperature and ±3⁄4°C over a full −55°C to +150°C temperature range, draws only 60 μA from the supply, and has very low self-heating of less than 0.1°C in air [92].

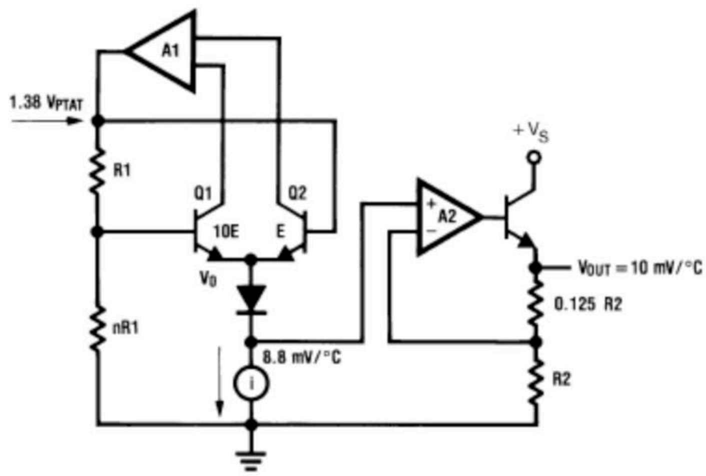

Figure 3.22: LM35 Sensor Circuit Schematic. [92]

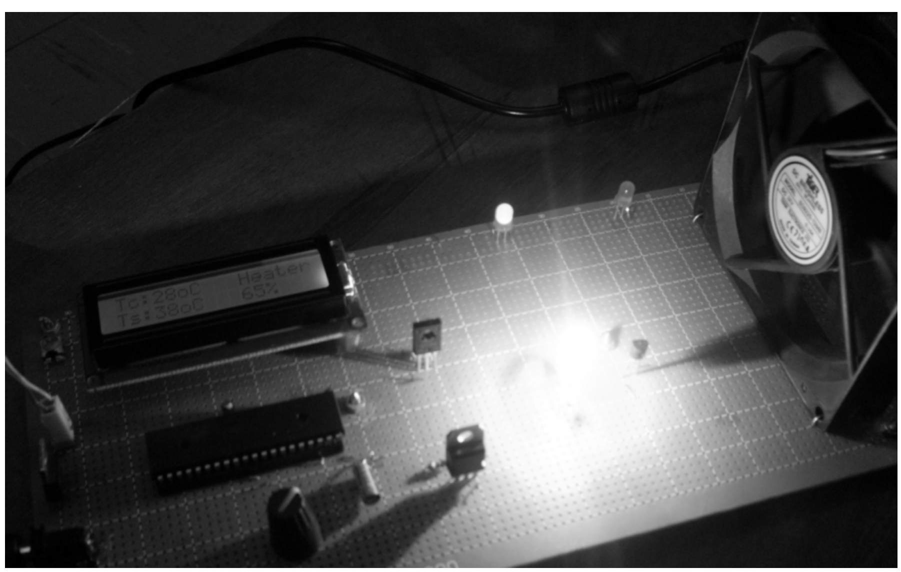

Figure 3.23: The completed temperature board.

3.8 Serial Input – Output with MATLAB

There are many approaches to communicate with robot; one of the best is using a code to communicate with a RS232 port (Figure 3.24) [93]. The RS-232 standard specifies that logic "1" is to be sent as a voltage in the range -15 to -5 V and that logic "0" is to be sent as a voltage in the range +5 to +15 V.

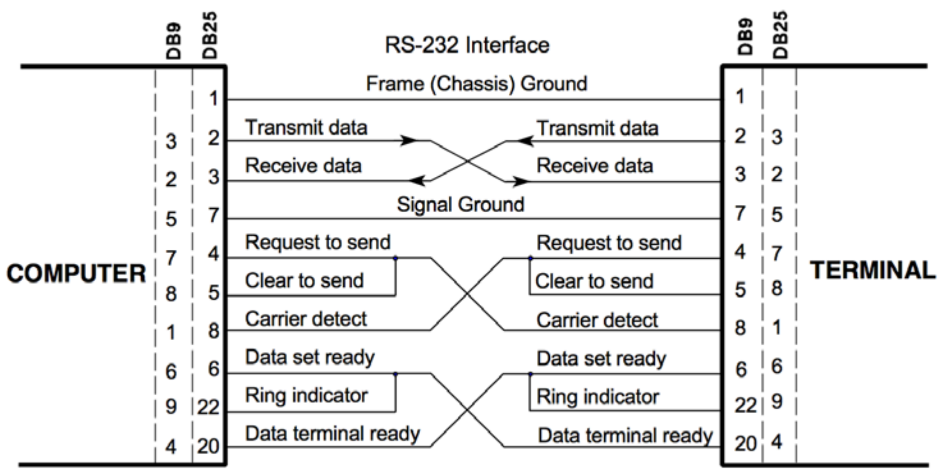

Figure 3.24: RS232 connections [93].

Also, it specifies that voltages of at least 3V in amplitude will always be recognized correctly at the receiver according to their polarity, so that appreciable attenuation along the line can be tolerated. The transfer rate is rated to be greater than 20 kbps and a distance up to 15m. The load impedance of the terminator side of the interface must be between 3000 and 7000 Ω.

The Matlab software was used to code the communications through a Serial IO data communication using the following steps:

1. Create serial port object. Use Matlab command serial
2. Configure serial port object. get and set commands

3. Connect to the device. Fopen
4. Configure if required. get and set commands
5. Write data with fprintf command. Read data with fscanf command.
6. Disconnect device on transmission over. Use fclose

3.9 Conclusion

In this chapter we reviewed the design of the different mechanical and electrical parts that constitute the robot. The fuzzy-logic based-software governing the core of the control unit (microcontroller) has been detailed. In the next chapter, we will discuss the obtained results.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

In the previous chapter, we discussed the design of different mechanical and electrical parts of a robot as well as the software controlling the robot operation. In this chapter, experiments will be performed to analyze the following matters:

- The performance of the fuzzy logic and PI control in stabilizing the robot's motion and providing its balance;
- The performance of the microcontroller improperly controlling the robot operation;
- The membership functions and PI values in balancing of the robot;
- The robot's performance while working at its maximum power during long periods.

These experiments were done both on land and in a 25mx10m swimming pool (with suctions/vacuums to change the water pressure).

4.2 Experiment on Land Using Tilt Sensor

One of the cheapest and most suitable ways to evaluate and examine the entire robot performance is (to experiment it) on land, since we can examine the microcontroller performance as well as the other parts while the system is not exposed to any risk. In Figures 4.1 (reference) and 4.2, the operation of the land gradient sensor is shown.

Tilt value:

Columns 1 through 7

0	0	0.0320	0.0320	0.0470	0.0470	0.0780

Columns 8 through 14

0.0780	0.0940	0.0940	0.1250	0.1250	0.1570	0.1570

Columns 15 through 21

0.1720	0.1720	0.2030	0.2030	0.2190	0.2190	0.2500

Columns 22 through 28

0.2500	0.2660	0.2660	0.2970	0.2970	0.3130	0.3130

Columns 29 through 35

0.3440	0.3440	0.3600	0.3600	0.3910	0.3910	0.4220

Columns 36 through 42

0.4220	0.4380	0.4380	0.4690	0.4690	0.4850	0.4850

Columns 43 through 49

0.5160	0.5160	0.5160	0.5470	0.5470	0.5630	0.5630

Columns 50 through 56

0.5630	0.5940	0.5940	0.6100	0.6100	0.6410	0.6410

Columns 57 through 63

0.6570	0.6570	0.6880	0.6880	0.7190	0.7190	0.7350

Columns 64 through 70

0.7350	0.7660	0.7660	0.7820	0.7820	0.8130	0.8130

Columns 71 through 77

0.8130 0.8440 0.8440 0.8600 0.8600 0.8600 0.8910

Columns 78 through 84

0.8910 0.9070 0.9070 0.9220 0.9530 0.9530 0.9850

Columns 85 through 91

1.0000 1.0000 1.0160 1.0160 1.0470 1.0630 1.0630

Columns 92 through 98

1.0940 1.1100 1.1100 1.1410 1.1410 1.1570 1.1570

Columns 99 through 100

1.1880 1.1880

Time value:

Columns 1 through 12

-4 -6 -6 -6 -4 -4 -4 -3 -3 -4 -4 -4

Columns 13 through 24

-4 -4 -4 -6 -6 -7 -6 -6 -6 -7 -7 -8

Columns 25 through 36

-8 -8 -8 -9 -9 -10 -10 -9 -8 -8 -8 -8

Columns 37 through 48

-8 -7 -6 -4 -1 0 0 1 2 3 5 6

Columns 49 through 60

8 11 13 15 15 16 17 19 28 19 17 11

Columns 61 through 72

13 16 20 21 20 16 12 8 6 7 8 7

Columns 73 through 84

5 3 0 0 -3 -8 -8 -10 -10 -12 -14 -18

Columns 85 through 96

-20 -22 -23 -26 -27 -28 -26 -22 -19 -15 -13 -10

Columns 97 through 100

-8 -8 -6 0

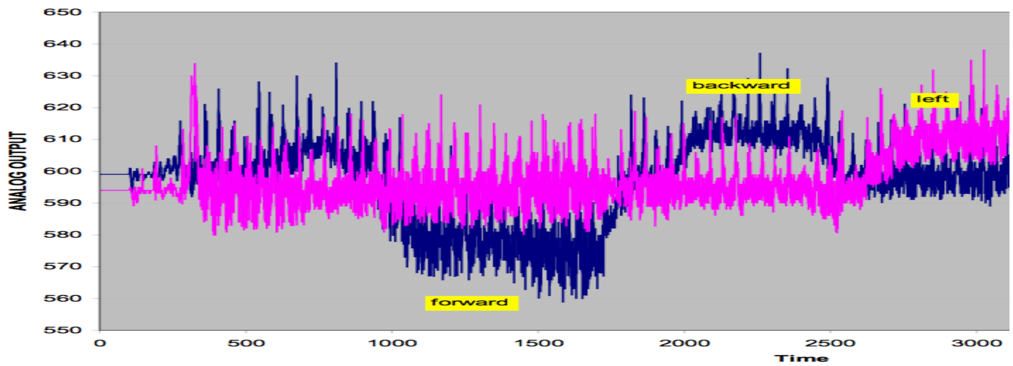

Figure 4.1: Accelerometer Output – Engine On [94].

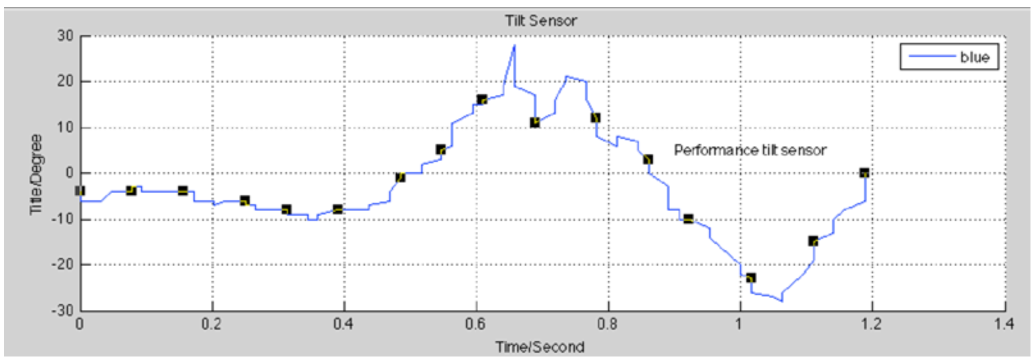

Figure 4.2: Plot of tilt sensor values.

4.3 Experiment on Land Using Compass Sensor

The compass sensor installed in the robot measures the inclination of the robot from the axis. It is equipped with two horizontal motors. Using this sensor, the microcontroller controls the horizontal motors via the PI system (Figure 4.3). In Figure 4.4, the operation of the compass sensor on the land in Matlab software is shown.

Compass Values:

Columns 1 through 7

0.2500 0.4530 0.6870 0.9210 1.1560 1.3900 1.6250

Columns 8 through 14

1.8280 2.0620 2.2960 2.5310 2.7650 3.0000 3.2340

Columns 15 through 21

3.4680 3.7030 3.9060 4.1400 4.3750 4.6090 4.8430

Columns 22 through 28

5.0780 5.2810 5.5150 5.7500 5.9840 6.2180 6.4530

Columns 29 through 35

6.6870 6.9210 7.1250 7.3590 7.5930 7.8280 8.0620

Columns 36 through 42

8.2960 8.5310 8.7650 8.9680 9.2030 9.4370 9.6710

Columns 43 through 49

9.9060 10.1400 10.3750 10.5780 10.8120 11.0460 11.2810

Column 50

11.5150

Time Values:

Columns 1 through 13

23 28 43 57 67 62 43 19 351 334 309 290 277

Columns 14 through 26

287 330 17 46 62 72 81 89 99 105 111 109 103

Columns 27 through 39

95 81 61 32 3 332 315 301 289 281 297 330 7

Columns 40 through 50

34 52 60 64 70 76 81 86 76 60 34

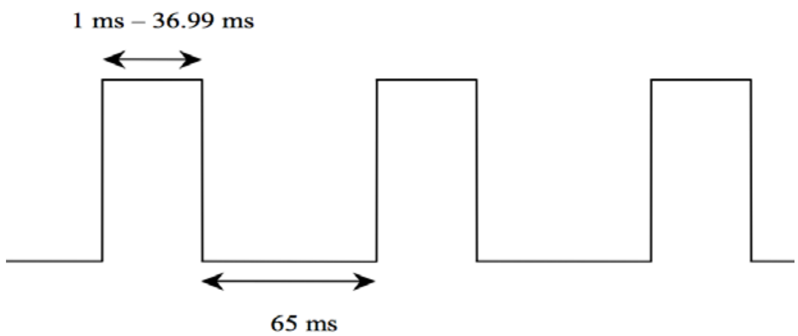

Figure 4.3: PWM signal output of compass sensor [95].

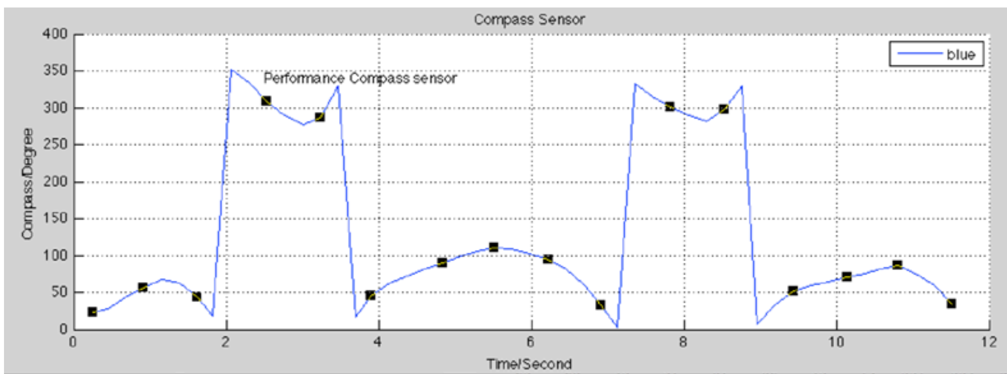

Figure 4.4: Plotted values of compass sensor on land.

The curve due to the noise generated by the motors on the jack, and considering that the robot is suspended in water, the noise is eliminated almost entirely.

4.4 Experiment on Land Using Pressure Sensor

The pressure sensor is used to calculate the accurate depth in water. It has been designed according to the sensor tests that the manufacturing company has provided and was then, programmed in the software section.

As mentioned in the previous chapter, the atmosphere pressure value which is equal to 14.7 psi. Because of the gain of amplifier, the sensor can only sense 26 psi because the values more than this are more than reference voltage of ADC and they are equal to 260. The voltage value of atmosphere pressure is equal to 3 volts (the reference voltage is equal to 5.5 volts). So the digital output value is equal to 173as shown in Figure 4.5. By analyzing the sensor test results for the robot, it can be inferred that:

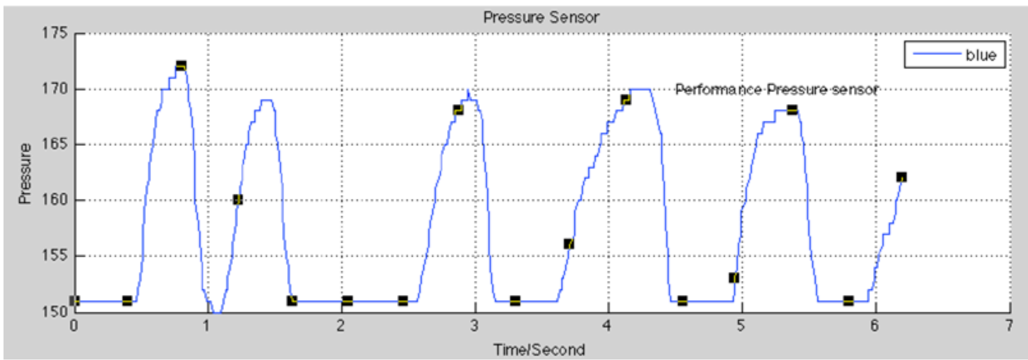

Figure 4.5: Plotted values of pressure sensor on land.

In dry environments, air compressors are used to simulate the real world. Figure 4.6 shows the deviation between the digital and depth, the digital output value being equal to 173.

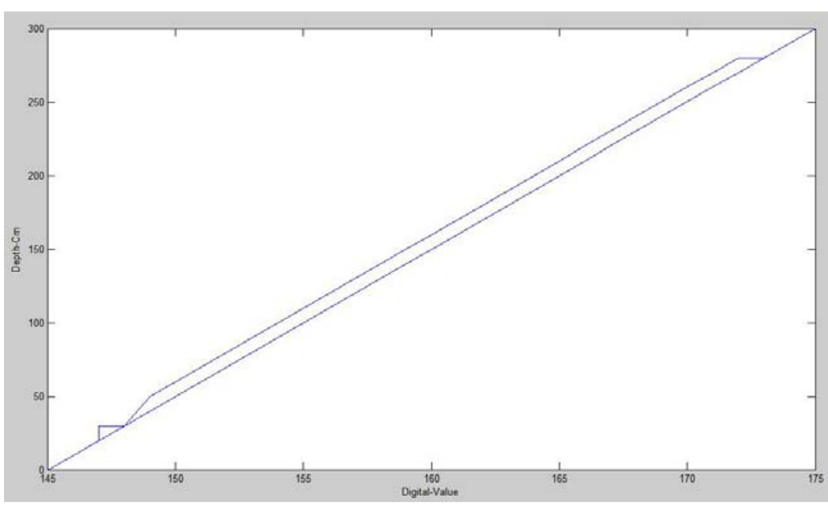

Figure 4.6: Test of pressure sensor in underwater.

4.5 Experiment on Land Using All Sensors (Tilt, Compass, and Pressure)

Another experiment performed in a dry environment is to test the operation of all sensors while simultaneously connected to the controller. Preliminarily results indicated that the robot is able to read and correctly transmit sensor data, but with some delay.

4.6 Experiments in Underwater Using Tilt Sensor and Fuzzy P Control

In a second step, experiments were performed underwater using the Fuzzy P control.

As seen in Figure 4.7, when the robot is exposed to a small deviation shock, around 20 degrees, it cannot stabilize its position and even this instability increased to about 20 degrees due to the high value of PWM in the output membership function. This part of the fuzzy program results in the robot's response. Thus, by changing the membership function from 150 to 200, new results were observed as shown in Figure 4.7.

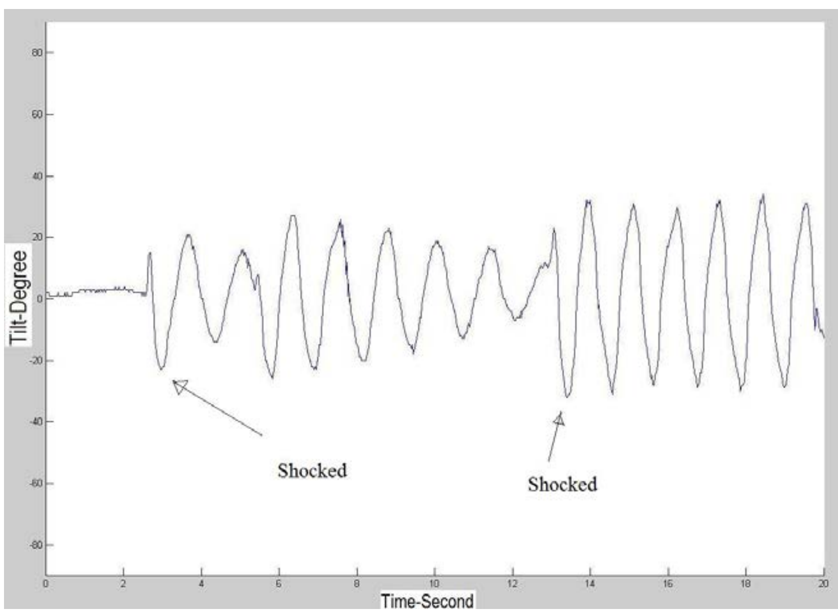

Figure 4.7: First response of the robot without fuzzy program. (0s-3s: stable, at 3s and 13s: create a shock, 3s-13s and 13s-20s: shock not absorbed)

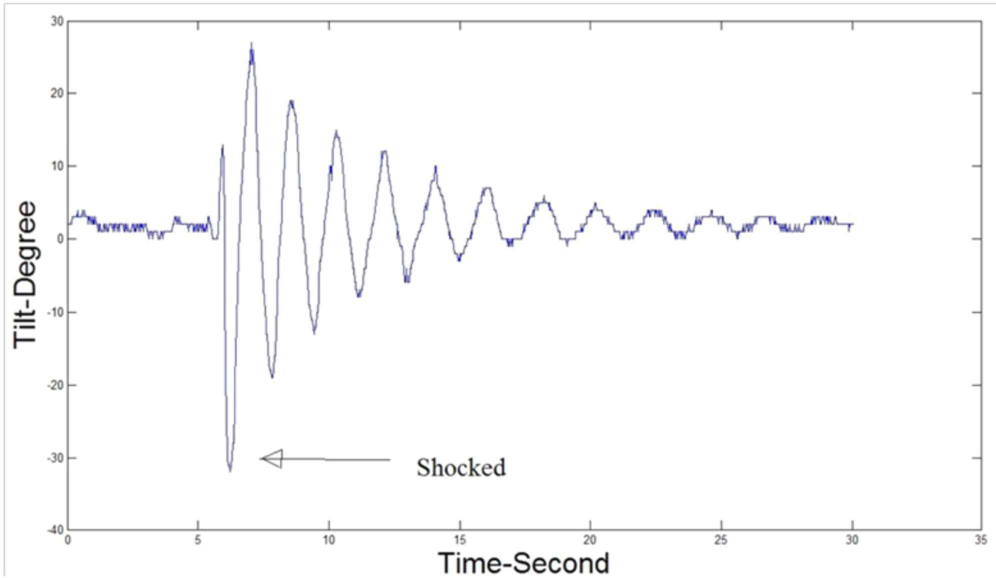

Figure 4.8: Tuned fuzzy output. (0s-5s: stable, 5s: create a shock, 5s-30s: shock absorbed)

This figure shows that even after a large deviation shock, around 30 degrees, the robot quickly reaches equilibrium but not yet a stabilized situation because of slower motors' response. In fact, reduction in the PWM value resulted in lower motor power (Figure 4.9). Hence, it should not happen since the robots are expected to perform properly.

Note that when the batteries are not fully charged, the response of the robot may fluctuate. So we performed the experiments with full charged batteries.

When the shock exceeds about 60 degrees more than the first shock, the motors efficiency maximize their performance and accordingly, the PWM value could be reduced and thus, better results may be obtained by changing the membership functions.

Also, after a period of time, the internal temperature increases inside the robot. The designed temperature control system adapted the membership functions for Low membership to start from -5 and high membership function to start from +5. The output PWM values for medium and high are 250 and 450, respectively. As seen in Figure 4.10, the robot is therefore more stabilized that the last experiment.

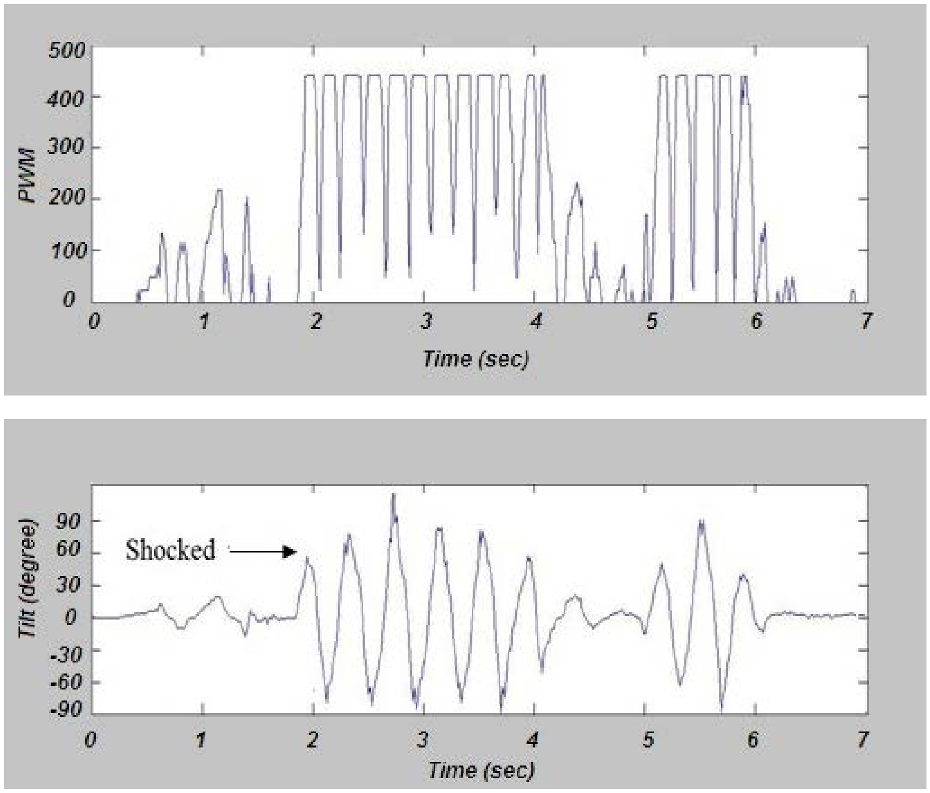

Figure 4.9: Response of fuzzy PWM and tilt without temperature control system. (0s-2s: stable, 2s-4s: create a shock, 4s-7s: shock not absorbed)

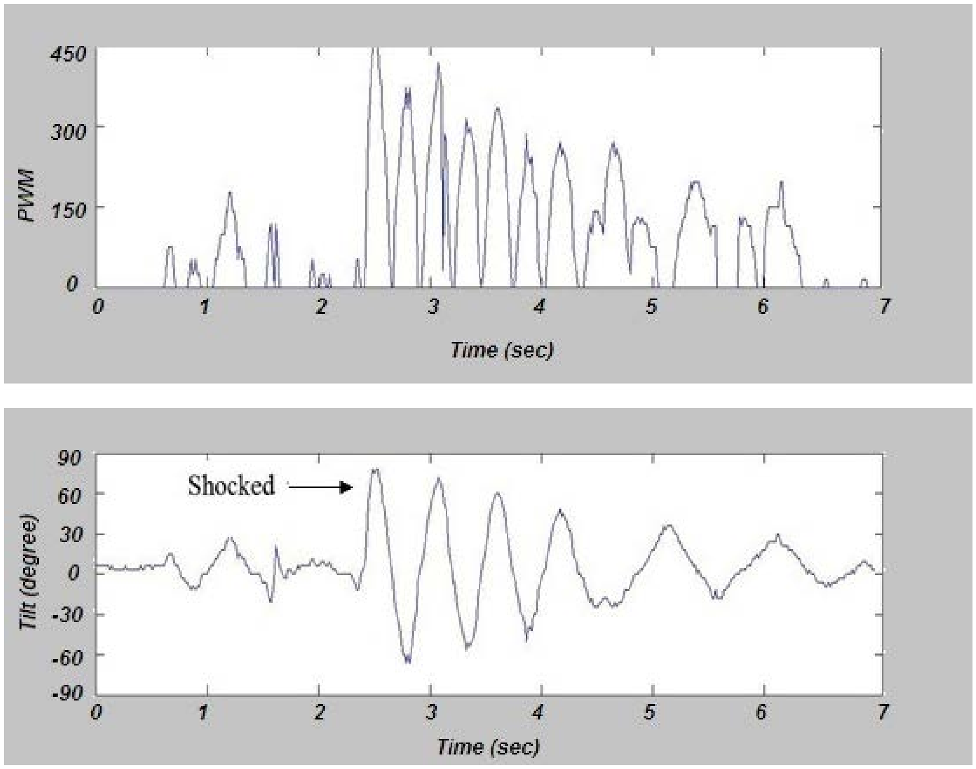

Figure 4.10: Response of fuzzy PWM and tilt with temperature control system (0s-2.5s: stable, 2.5s: Create a shock, 4s-7s: shock absorbed).

4.7 Experiments in Underwater Using Tilt Sensor and Fuzzy PI Control

As mentioned in the previous chapter, one way of reducing steady-state error is using Fuzzy PI; thus, the experiment was performed underwater using a Fuzzy PI controller inside the robot. At first, the values for P and I membership functions are as:

P:

Positive Small = -60,

Positive Medium = 100,

Positive Large = 160.

I:

Positive Small = 40,

Positive Large = 80.

The output's PWM is based on the PI's mathematical equation. First, the fuzzy PI was not tuned and the response of the system was with predefined variables. As seen in Figure 4.11, the robot was unable to be balanced and it was going to oscillate.

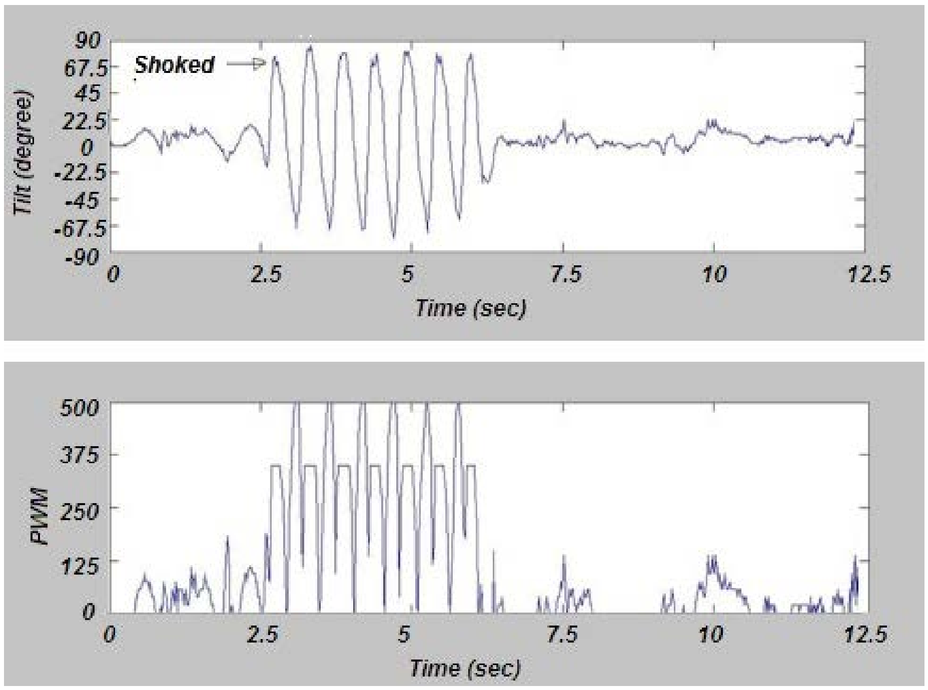

Figure 4.11: First response of the Fuzzy PI. (0s-2.5s: stable, 2.5s: create a shock, 3s-5.5s: shock begins to rise and 6s-12.5s: robot forced to keep the same direction)

Then, we tuned the fuzzy variables. The new values for P and I membership functions were set as:

P:

Positive Small = -40,

Positive Medium = 50,

Positive Large = 90.

I:

Positive Small = 45,

Positive Large = 60

The response of the system with these new values is depicted in Figure 4.12. By comparing the obtained results, it can be stated that the balanced error fades but it is still very sensitive and might get out of the balanced condition with slight changes.

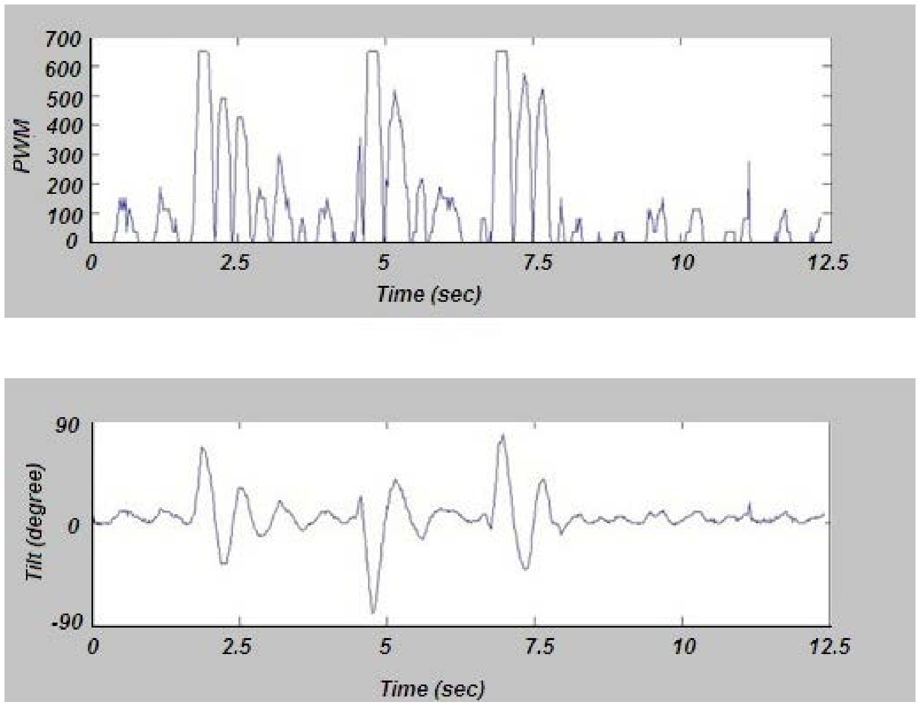

Figure 4.12: Last tuning of Fuzzy PI (0s-2s: stable, 2s: create a small shock, 2.5s-4.5s: shock absorbed, 4.5s: create a small shock, 5s-6s: shock absorbed)

As for the forward movement of the robot using spinning horizontal motors, we performed a first experience without neither the feedback nor the controller system to manage the robot balance and stabilization (Figure 4.13).As expected, the robot's gradient is increasing when it is moving forward, and then reaches zero when it stops. When it starts again, we have an increasing gradient again.

After adding feedback and fuzzy logic to the gradient value and by regulating the final settings of the fuzzy PI, the robot's movements become more stabilized as shown in Figure 4.14 (the 35 deviation gradient degree without feedback becomes less than 10 degrees with feedback).

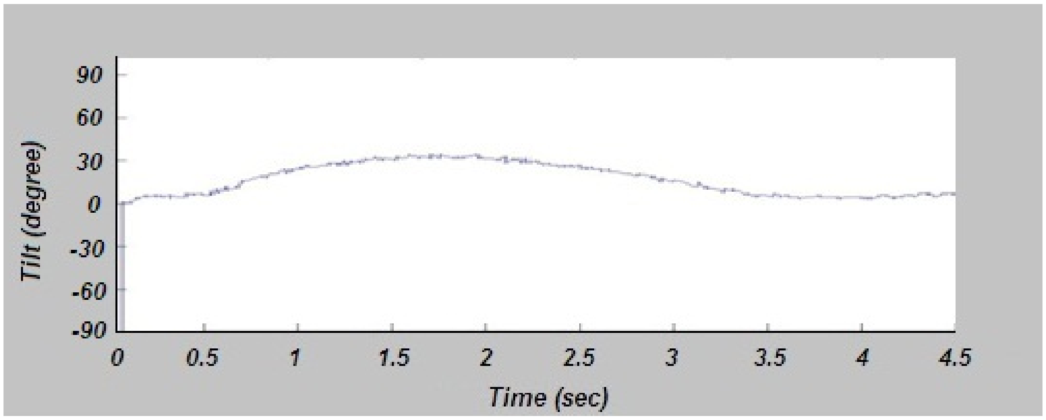

Figure 4.13: Tilt's changes, without control feedback, when the robot is moving forward.

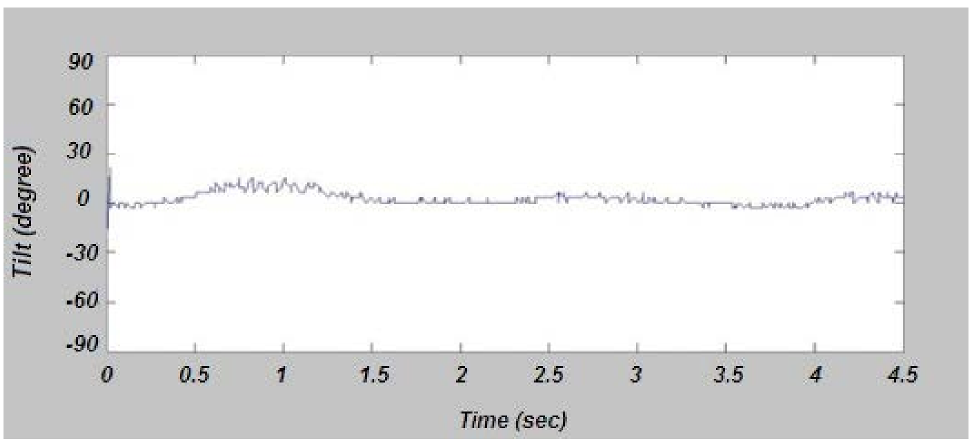

Figure 4.14: Tilt's changes, with fuzzy control, when the robot is moving forward.

4.8 Experiments in Underwater Using Compass Sensor and PI Control

For this set of experiments, we used the two horizontal motors installed on the robot. However, these two motors were not able to move in the same direction without the help of the PI control feedback (Figure 4.15).

However, by introduced the PI control, the robot's response, sent as two different values from the PI controller motors, is significantly improved as shown in Figures 4.16 and 4.17.

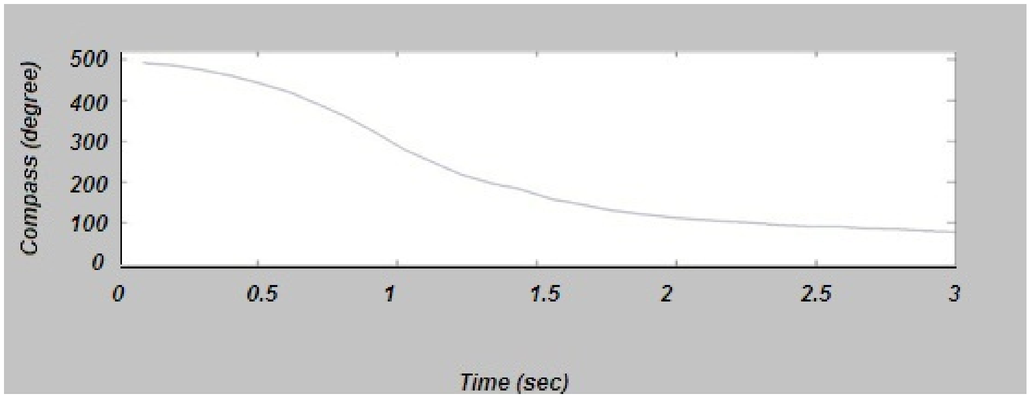

Figure 4.15: Change of Compass's degree, when the robot is moving forward without Control feedback.

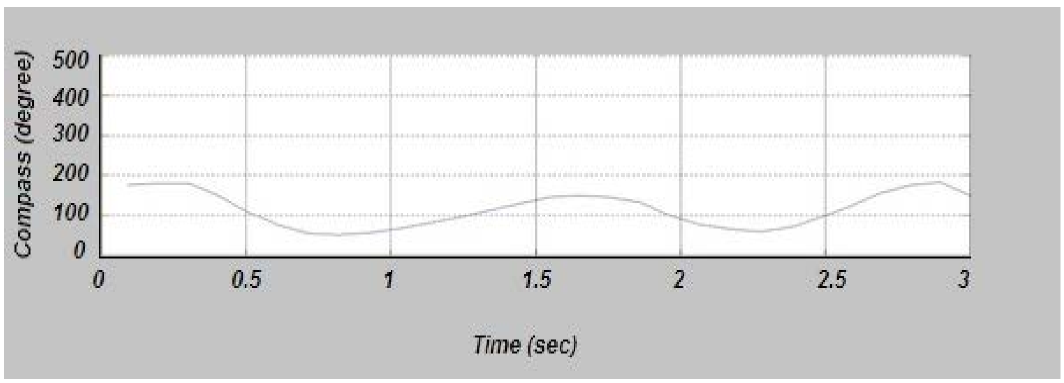

Figure 4.16: Change of Compass's degree, when the robot is moving forward with the first control feedback.

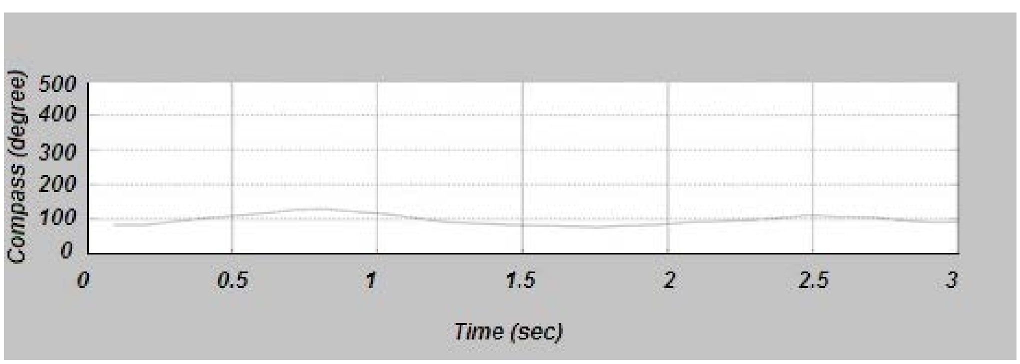

Figure 4.17: Change of Compass's degree, when the robot is moving forward with the second control feedback.

4.9 Experiments in Underwater Using Compass and Tilt Sensors

This experiment, again performed underwater, evaluated the simultaneous performance of the compass sensor and the tilt sensor on the robot. The PI values were altered in order to attain the maximum balance. Due to the probable interference of the two sensor's information, a software error may occur. But at the end, after making appropriate tuning and regulating the PI performance, the robot's performance becomes very close to the desired one (Figures 4.18 and 4.19).

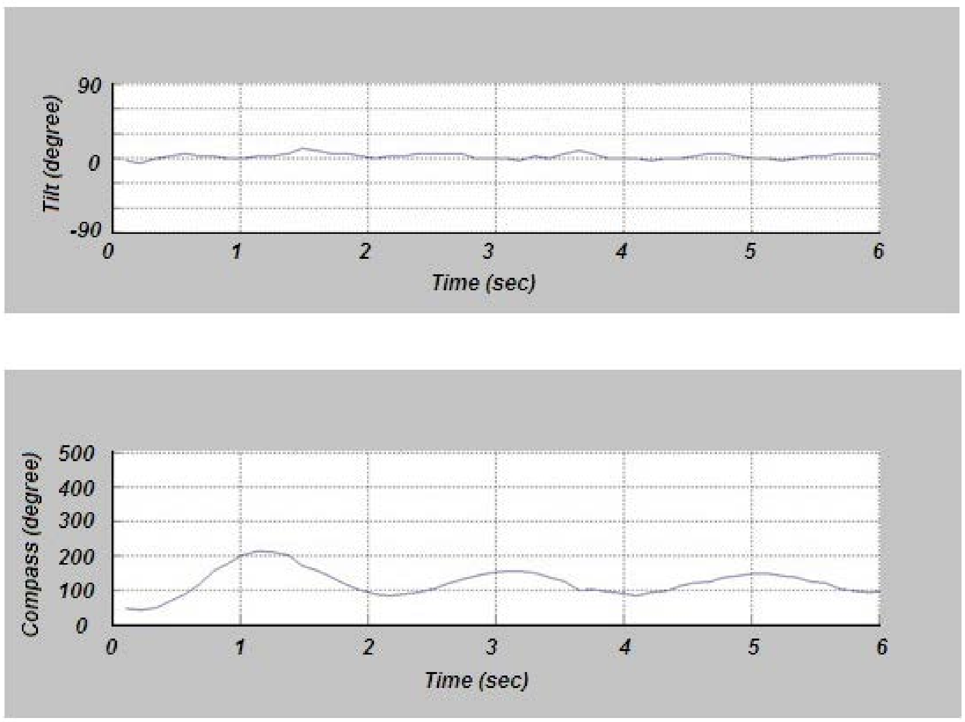

Figure 4.18: Change of Compass and tilt's degree, when the robot is moving forward without feedback control.

Generally, by increasing the changing input values, the microcontroller helps the robot's balance by performing the controlling functions and each of the control programs prevents the other control program to run according to priority. In each cycle, some of the output values become disabled and they follow the instructions that had been sent to them before. The priority of a control function temporarily belongs to a cycle which contains the higher deviation input value which indicates the largest error in a balanced situation.

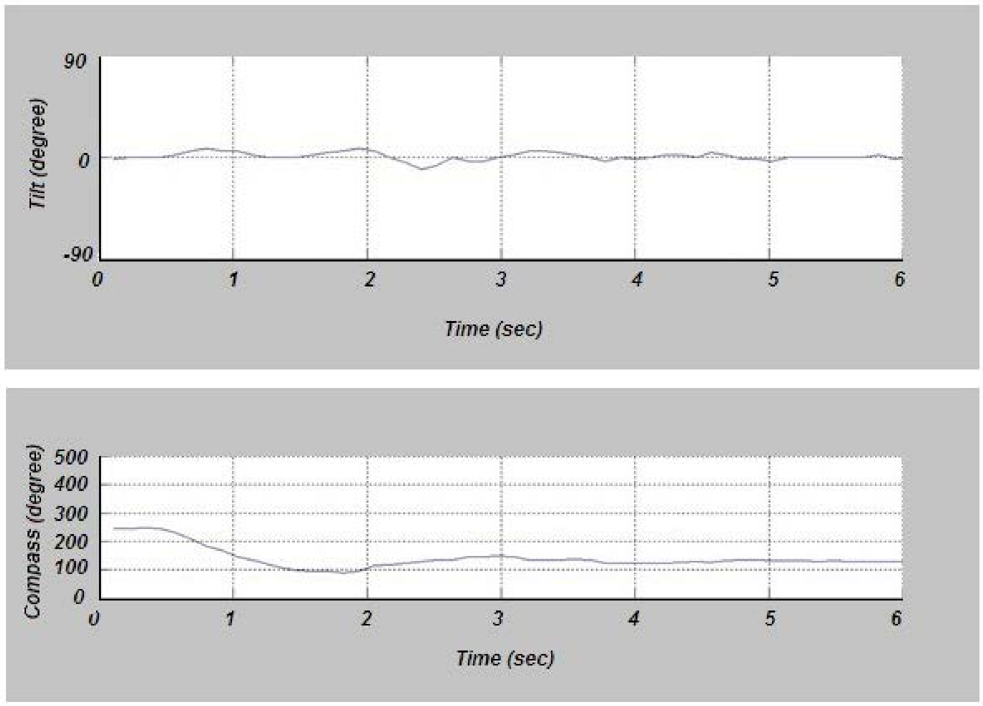

Figure 4.19: Change of Compass and tilt's degree, when the robot is moving forward with feedback control.

4.10 Experiments in Underwater Using Pressure and Tilt Sensors

Generally, the robot should remain on the water due to the design of its structure. However, the two vertical motors made the robot to sink in water. Also, some of the water's fluctuations are due to the water's movement in the pool's depth due to the existence of a suction pump, electrical vacillations, and motor operation. Thus, the diagram would not be totally smooth.

Furthermore, fuzzy control in gradient sensors, for the sinking condition in a pool has been implemented implicitly. Figure 4.20 represents the two sensors' performance where the robot's balance at water depth is indicated by the PI controller.

The determined depth for sensors is approximately five meters, but due to the suction pumps at the pool bed, it was not possible to reach the maximum depth. Also, according to its technical specifications, the robot needs to work with an approximate voltage of 0.5 volt. Due to using washers around the shaft and their insulation and water pressure, the voltage increases to about 2 to 5 volts, but preliminary setups showed that this change in voltage did not affect the robot's motion. This, according to fuzzy settings and PI control, resulted in specification errors of approximately less than 5 degrees and 15 cm, respectively.

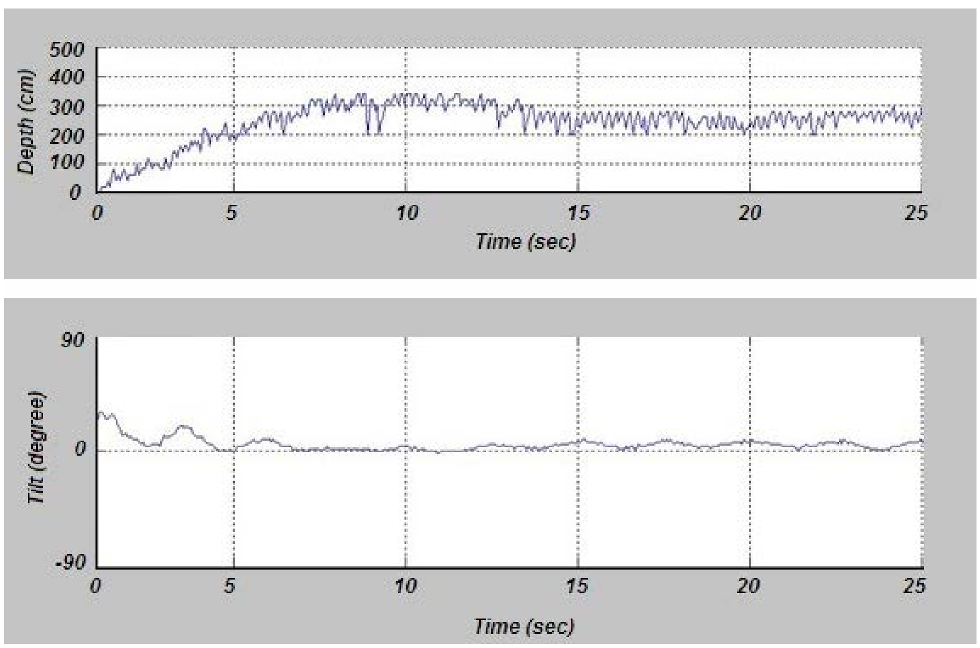

Figure 4.20: change of Compass's and degree, when the robot is moving forward with the control feedback.

4.11 Conclusion

The developed software as well as the designed hardware systems were successfully tested both on land and underwater to demonstrate their correct operation. In fact, the different fuzzy-based controllers and their sensors responded adequately to the imposed changes.

Note that during the above experiments, all tunings were set manually and regulated based on estimates but, in the real world, these tunings should be made automatically according to the environment requirements and changes.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

This chapter discusses the obtained results concerning the design of a fuzzy-based controller coupled to sensors for underwater robot operation. It also gives recommendations for future suggested developments.

5.1 Conclusion

In this thesis, simulations and experiments were carried out to study the implementation of a relatively fast and cost-effective controller for small underwater robots. Based on fuzzy logic and PI control architecture, a new mechanical and electrical control system for autonomous underwater robot systems was designed. Furthermore, in order for the embedded robot system to efficiently interact to dynamically changing environments, six degrees of freedom have been considered for stabilized moves in both vertical and horizontal directions.

The main contributions of this thesis are:

- Design of a small cost-effective autonomous underwater robot, taken as reference,
- Design, fabrication, and test of compatible and inexpensive sensors to be used in cost-effective small autonomous robots,
- Design, fabrication, and test of a microcontroller for AUVs,
- Development of different fuzzy-based codes in C and Matlab to drive the sensors as well as the microcontroller for proper robot operation.

The different hard and soft parts were successfully tested on land and in a pool.

5.2 Future Developments

A cost-effective underwater vehicle robot was simulated and the desired objectives successfully achieved. However, this robot should be effectively fabricated and experimentally tested to further demonstrate the proper operation of the developed codes and their related hardware (controller and sensors) in real situations. Furthermore, these systems should be further enhanced to incorporate more features to improve the overall robot performance. The improvement can be achieved by making modification on the mechanical parts as well as on the electronic instrumentation aspect. In fact, one of the important points to be improved is the mechanical structure of the robot because to make it more flexible and easy to maintain and/or upgrade (as suggested in Figure 5.1).We can also

- reduce the robot size and weight for higher performance, by using more flexible and lighter materials,
- optimize the number of sensors required for reliable robot operation, by changing the design and structure of the robot,
- develop and design infrared sensors to control the underwater robot,

- design a self-tuning gain depth controller for autonomous underwater vehicles with mass mechanism to avoid manual settings,
- design a distributed negative power control system for high efficient move of underwater robots,
- design and implement a multiple degree-of-freedom arm system on the underwater robot for search and explore in the underwater environment.

As future development, we could also include a vision system for the AUV. Performing 3D measurements of underwater objects using a fish-eye stereo camera will allow creating a 3D database which can be later used to efficiently model the robot environment, and afterwards, to optimize the robot operation while exploring the underwater world in a predefined region.

Finally, note that during the above experiments, all tunings were set manually and regulated based on estimates but, in the real world, these tunings should be made automatically according to the environment requirements and changes.

REFERENCES

[1] Appelqvist, P., *Mechatronics Design of a Robot Society-A Case Study of Minimalist Underwater Robots for Distributed Perception and Task Execution.* PhD thesis, Series A: Research Reports No. 22. Helsinki University of Technology, Finland, 2000.

[2] Milella, A., Di Paola, D., and Cicirelli, G., *Mechatronic Systems, Applications*. *In-Teh*, Vukovar, Croatia, 2010.

[3] Pothiya, S., Ngamroo, I., and Kongprawechnon, W., Design of optimal fuzzy logic-based PID controller using multiple tabu search algorithm for AGC including SMES units. *Int. Conf. Power Engineering*, pp. 838-843, 2007.

[4] Rastogi, M., Arora, A., and Hotel, Y.V., *Design of Fuzzy Logic Based PID Controller for an Unstable System*. Springer Berlin Heidelberg, CCIS 157, pp. 566-571, 2011.

[5] Repoulias, F., and Papadopoulos, E., Three dimensional trajectory control of under actuated AUVs. *European Control Conf*., Kos, Greece, pp. 3492-3499, 2007.

[6] Perez, M.C., *A Proposal of behaviour based control architecture with reinforcement learning for an autonomous underwater robot*. PhD. Thesis. University of Girona. 2003.

[7] Caccia, M., Bono, R., Bruzzone, G., and Veruggio, G., Bottom following for remotely operated vehicles. *Control Engineering Practice*, Vol. 11, pp. 461–470, 2003.

[8] Mat Saad, E.H., and Yaakob, O., Design and implementation of electronic control system for UTM-AUV, *Tencon Proc.*, Vol. II, pp 332-337, 2000.

[9] Choi, S.K., J. Yuh, J., and Takashige, G.Y., Development of the omni directional intelligent navigator, *IEEE Robot and Automat. Mag*., Vol. 2, pp. 44–53, 1995.

[10] Zhao, S., and Yuh, J., Experimental study on advanced underwater robot control, *IEEE Trans. Robotics*, Vol. 21, pp. 695-703, 2005.

[11] Yuh, J., Design and control of autonomous underwater robots: A Survey. *Autonomous Robots*, Vol. 8, pp. 7–24, 2000.

[12] Autonomous Underwater Vehicle, REMUS, Oceanographic Systems Laboratory. *Woods Hole Oceanographic Institution*, 2014.

[13] Naval AUV product range. The HUGIN & REMUS Family, *Kongsberg Maritime AS*, 2009.

[14] Autonomous Underwater Vehicle –AUV. The HUGIN Family, *Kongsberg Maritime AS*, Version 2.

[15] Fossen, T.I., *Guidance and Control of Ocean Vehicles*, New York: Wiley, 1994.

[16] Fossen, T.I., *Underwater Vehicle Dynamics*, in *Underwater Robotic Vehicles: Design and Control*, J. Yuh (Ed.), 1995.

[17] Fowles, G.R., and Cassiday, G.L., *Analytical Mechanics*. 7th Ed., Thomson Brooks/Cole, 2005.

[18] Khalil, H.K. *Nonlinear Systems*, 2nd Ed., Prentice Hall, 1996.

[19] Bessa, W.M., Dutra, M.S., and Kreuzer, E. *Dynamic Positioning of Underwater Robotic Vehicles with Thruster Dynamics Compensation*, InTech. Open, 2013.

[20] Muljowidodo, K., Sapto Adi, N., Budiyono, A., and Prayogo, N., Design of SHRIMP ROV for surveillance and mine sweeper, *Indian J. of Marine Sciences*, Vol. 38, pp 332-337, 2009.

[21] Shi, Y., Qian, W., Yan, W., and Li, J., Adaptive depth control for autonomous underwater vehicles based on feed forward neural networks, *Int. J. of Computer Science & Applications*, Vol. 4, pp 107-118, 2007.

[22] Yuh, J., Underwater Robotics. *IEEE Int. Conf. on Robotics and Automation*, pp. 932-937, 2000.

[23] Mahesh, M., Yuh, J., and Lakshmi, R., A coordinated control of an underwater vehicle and robotic manipulator. *J. of Robotic Systems on Underwater Robotics*, Vol. 8, pp. 339–370, 1991.

[24] McLain, T.W., Rock, S.M., and Lee, M.J., Experiments in the coordinated control of an underwater arm/vehicle system. *Underwater Robots*, Vol. 3, pp. 213–232, 1996.

[25] Shoults, G.A., *Dynamics and control of an underwater robotic vehicle with an N-axis manipulator*, Ph.D. Thesis, Washington University, 1996.

[26] Antonelli, G. and Chiaverini, S., Task-priority redundancy resolution for underwater vehicle-manipulator systems. *IEEE Int. Conf. on Robotics and Automation*, pp. 768–773,1998.

[27] Canudas-de-Wit, C., Diaz, E.O., and Perrier, M., Robust nonlinear control of an underwater vehicle/manipulator system with composite dynamics. *IEEE Int. Conf. on Robotics and Automation*, pp. 452–457, 1998.

[28] Vestgard, V., Hansen, R., Jalving B., and Pedersen, O.A., The HUGIN 3000 Survey AUV, *Int. Offshore and Polar Eng. Conf. on Offshore Technology*, pp. 679-684, 2001.

[29] Stokey, R., Roup, A., Alt, C.V., Allen, B., Forrester, N., Austin T., Goldsborough R., Purcell, M., Jaffre, F., Packard, G., and Kukulya, A., Development of the REMUS 600 autonomous underwater vehicle. *IEEE Int. Conf. on Oceans*, pp. 1301-1304, 2005.

[30] Zhao, S., and Yuh, J., Experimental study on advanced underwater robot control, *IEEE Trans. Robotics*, Vol. 21, pp. 695–703, 2005.

[31] Yoerger, D.R., Bradley, A. M., Jakuba, M., German, C.R., Shank, T., and Tivey, M., Autonomous and remotely operated vehicle technology for hydro-thermal vent discovery, exploration, and sampling. *Oceanography Society*, Vol. 20, pp. 152–161, 2007.

[32] Kinsey, J.C., Whitcomb, L.L., Yoerger, D.R., Howland, J.C., Ferrini, V.L., and Hegrenæs, O., New navigation post-processing tools for oceanographic

submersibles. *American Geophysical Union, Fall Meeting,* Abstract OS33A-1678, 2006.

[33] Liu, F., and Chen, H., Motion control of intelligent underwater robot based on CMAC-PID. *Int. Conf. on Information and Automation*, pp. 1308– 1311, 2008.

[34] Doniec, M., Vasilescu, I., Detweiler, C., and Rus, D., Complete SE3 underwater robot control with arbitrary thruster configurations. *IEEE Int. Conf. Robotics and Automation*, pp. 5295-5301, 2010.

[35] Santhakumar, M., and Asokan, T., Planar tracking control of an under actuated autonomous underwater vehicle. *World Academy of Science, Engineering and Technology*, Vol. 5, pp. 57-62, 2011.

[36] Simpson, A.K., *Autonomous Robot Control for Underwater Tunnel Mapping*. *COS 398: Spring Independent Work*. California Polytechnic State University, 2012.

[37] Yue, C., Guo, S., Li, M., and Li, Y., Passive and active attitude stabilization method for the spherical underwater robot (SUR-II). *IEEE Int. Conf. Robotics and Biomimetics,* pp. 1019-1023, 2013.

[38] Srinonchat, J., Enhancement embedded system for control and monitoring design underwater robot, *Int. Conf. on Advances in Electronics and Electrical Technology*, pp. 1- 4, 2014.

[39] Yoerger, D.R., and Slotine, J.J.E., Robust trajectory control of underwater vehicles. *IEEE J. of Oceanic Engineering*, Vol. 10, pp. 462–470, 1985.

[40] Yuh, J., An adaptive and learning control system for underwater robots. *In: 13th World Congress Int. Federation of Automatic Control*, pp. 145-150, 1996.

[41] Choi, S.K., and Yuh, J., Experimental study on a learning control system with bound estimation for underwater vehicles. *Int. J. of Autonomous Robots*, Vol. 3, pp. 187–194, 1996.

[42] Cao, B., Li, T.F., and Zhang, C.Y., *Fuzzy Information and Engineering*, Vol. 2, Springer, 2009.

[43] DeBitetto, P.A., Fuzzy logic for depth control of unmanned undersea vehicles. *Symp. of Autonomous Underwater Vehicle Technology*, pp. 233–241,1994.

[44] Tsukamoto, C.L., Yuh, J., Choi, S.K., Lee, W.C., and Lorentz, J., Experimental study of advanced controllers for an underwater robotic vehicle thruster system, *Int. J. of Intelligent Automation and Soft Computing*, Vol. 5, pp. 225–238, 1999.

[45] Blidberg, D.R., and Jalbert, J., AUV mission & system sensors. In underwater Robotic Vehicles: Design and Control, *Int. Symp. on Unmanned Untethered Submersible Technology*, pp. 106-117, 1995.

[46] Srinonchat, J., Enhancement embedded system for control and monitoring design of underwater robot, *Int. Conf. on Advances in Electronics and Electrical Technology*, pp 1-4, 2014.

[47] Sasiadek, J.Z., Lu, Y., and Polotski, V., Navigation of autonomous mobile robots, robot motion and control, *Springer Lecture Notes in Control and Information Sciences*, No. 360, pp. 187-208, 2007.

[48] Black, M.R., and Butler, B., Arctic ocean trials of track point ultra-short baseline acoustic positioning system. *IEEE AUV Technology*, pp. 297–302,1994.

[49] Austin, T.C., The application of spread spectrum signalling techniques to underwater acoustic navigation. *IEEE AUV Technology*, pp. 443–449,1994.

[50] Vaganay, J., Bellingham, J.G., and Leonard, J.J., Outlier rejection for autonomous acoustic navigation, *IEEE Int. Conf. on Robotics and Automation*, Vol. 3, pp. 2174–2181, 1996.

[51] Christ, R.D., and Wernli, R.L., The ROV Manual: A User Guide for Remotely Operated Vehicles. *Butterworth-Heinemann*, 2007.

[52] Chappell, S.G., Jalbert, J.C., Pietryka, P., and Duchesney, J., Acoustic communication between two AUVs, *IEEE AUV Technology*, pp. 462–469, 1994.

[53] Herold, D., and Johnson, M., A compact underwater acoustic modem. *IEEE AUV Technology*, pp. 393–398, 1994.

[54] Kambe, T., Elementary Fluid Mechanics. *World Scientific*, 2007.

[55] Fluid mechanics submarine design, *Australian Maritime College*, Group Two, 2011.

[56] Yasunobu, S., and Miyamoto, S., *Automatic Train Operation System by Predictive Fuzzy Control*, M. Sugeno, Ed., Industrial Applications of Fuzzy Control, pp. 1-18, 1985.

[57] Østergaard, J.J., Fuzzy II: The new generation of high level kiln control. *ZementKalk Gips (Cement-Lime-Gypsum)*,Vol. 43, pp. 539–541,1990.

[58] Kandel, A., and Langholz, G., *Fuzzy Control Systems*. CRC Press, 1994.

[59] Mamdani, E.H., Applications of fuzzy algorithms for control of simple dynamic plant. *Proc. IEE*, Vol. 121, pp. 1585-1588, 1974.

[60] Palm, R., Driankov, D., and Hellendoorn, H., *Model Based Fuzzy Control*. Springer, 1997.

[61] Driankov, D., and Palm, R., *Advances in Fuzzy Control*. Physica-Verlag, 2013.

[62] Babuska, R., *Fuzzy Modeling for Control*. Kluwer Academic, 1998.

[63] Kosko, B., Fuzzy systems as universal approximators, *IEEE Trans. Computers*, Vol. 43, pp. 1329-1333, 1994.

[64] Wang, L.X., *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*, Prentice Hall, 1994.

[65] Zeng, X.J., and Singh, M.G., Approximation theory of fuzzy systems -MIMO case. *IEEE Trans. Fuzzy Systems*, Vol. 3, pp. 219-235, 1995.

[66] Vaclav Bezdek, Possible use of fuzzy logic in database, *J. of Systems Integration*, Vol. 2, pp 31-46, 2011.

[67] Lee, K.H. First course on fuzzy theory and applications, *Springer Berlin Heidelberg*, Vol. 27, 2005.

[68] Costa, A., De Gloria, A., Giudici, F., and Olivieri, N., Fuzzy logic microcontroller. *IEEE Micro*, Vol. 17, pp 66-74, 1997.

[69] Bai, Y., Zhuang, H., and Wan, D., *Advanced Fuzzy Logic Technologies in Industrial Applications*, Springer, 2006.

[70] Fuzzy Logic Toolbox, *User's Guide*, *The MathWorks Inc.*, 2014.

[71] Solano, S.S., and Baturone, I., Microelectronic design of universal fuzzy controllers. *Int. Conf. on Mathware & Soft Computing*, pp. 303-319, 2001.

[72] Sivanandam, S. N., Sumathi, S., and Deepa, S. N., *Introduction to Fuzzy. Logic using MATLAB*, Springer, 2006.

[73] Yasunobu, S., and Miyamoto, S., Automatic train operation system by predictive fuzzy control. *Industrial Applications of Fuzzy control*, pp. 1-18, 1985.

[74] Ostergaard, J.J., *Fuzzy Logic Control of a Heat Exchanger Process*, in Fuzzy Automata and Decision Process, Gupta, M.M., Saridis, G.N., and Gaines, B.R., Eds., North-Holland, pp. 285-320,1977.

[75] Pedrycz, W., *Fuzzy Control and Fuzzy Systems*, 2nd Ed., Research Studies Press Ltd., 1993.

[76] Verbruggen, H.B., and Babuska, R., *Fuzzy Logic Control: Advances in Applications*, in World Scientific Series in Robotics and Intelligent Systems, Vol. 23, 1999.

[77] Watts, M. J., Fuzzy Inference and Defuzzification. *Lecture Outline*, Last access, April 1st , 2014. http://mike.watts.net.nz/Teaching/IIS/Lecture5.pdf

[78] Yen, J., Langari, R., and Hall, P., *Fuzzy Logic: Intelligence, control, and Information*, Prentice Hall, 1998.

[79] Holmblad, L.P., and Østergaard, J.J., *Control of a Cement Kiln by Fuzzy Logic*, in Fuzzy Information and Decision Processes, North-Holland, pp. 389–399, 1982.

[80] Taguchi, H., and Araki, M., Two-degree-of-freedom PID controllers, *IFAC Workshop on Digital Control: Past, Present and Future of PID Control*, pp. 91–96, 2000.

[81] Astrom, K.J. Control system design, Lecture note, *University of California*, 2002.

[82] http://perguruanfarhan.wordpress.com, Last access, May 1st, 2014.

[83] Bayliss, M., Bax, M., and Short, D., *Underwater Inspection*, E. and F.N. Spon, 1998.

[84] National University of Singapore, school of Computing, P89V51RD2, http://www.comp.nus.edu.sg/~adityak/Model_Contactor/Report/report.pdf, Last access, April 1st, 2014.

[85] Koninklijke, *Philips Electronics* N.V., 2014.

[86] P89V51RB2_RC2_RD2_4, *Philips Electronics*, 2007.

[87] www.thinklabs.in, Last access April 1st, 2014.

[88] RS-380PH, Mabuchi Motor co., Japan, 2010.

[89] http://www.robot-electronics.co.uk/htm/cmps3tech.htm, Last access, April 1st, 2014.

[90] One Technology Way, *Analog Devices*, ADXL320, 2004.

[91] Pressure Sensors. 26PC Series. Gage and Differential/Unamplified- Compensated. *Honeywell Sensing and Control*, 2014.

[92] Texas Instruments, *LM Precision Centigrade Temperature Sensors*, 2013.

[93] http://www.tscm.com/rs-232.pdf, Last access, April 1st, 2014.

[94] Prakongpan, N., Special Sensor Report, EEL 5666 Intelligent Machines Design Laboratory, *University of Florida*, 2005.

[95] Dandavate, G., and Kwok, S.Y., Compass Localization. CSCI 445, Robo-Soccer Competition, Final Report, *University of Southern California*, 2004.

[96] Fowles, G.R., and Cassiday, G.L., *Analytical Mechanics*, 7th Ed., Thomson Brooks/Cole, 2005.

[97] Axisa, F., and Antunes, J., *Modelling of Mechanical Systems: Fluid-Structure Interaction*, Elsevier, 2007.

[98] Girard, A.R., Smith, S.M., and Ganesan, K., A convenient form for discrete event control of autonomous underwater vehicles, *Int. Workshop on Autonomous Underwater Vehicles for shallow water and coastal Environment*, pp. 125-130, 1998.

[99] Goheen, K.R., and Jeffery, R.E., Multivariable self-tuning autopilots for autonomous and remotely operated underwater vehicles, *IEEE J. of Oceanic Engineering*, Vol. 15, pp. 144–151, 1990.

[100] Healey, A.J., and Lienard, D., Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles, *IEEE J. of Oceanic Engineering*, Vol. 18, pp. 327–339, 1993.

[101] Vasilescu, I., Kotay, K., Rus, D., Dunbabin, M., and Corke, P., Data collection, storage, and retrieval with an underwater sensor network, *Int. Conf. on Embedded Networked Sensor Systems*, pp. 154-165, 2005.

[102] Johnson, M., Herold, D., and Catipovic, J., The design and performance of a compact underwater acoustic network node, *IEEE Oceans*, Vol. 3, pp. 467–471, 1994.

[103] Judge, J. J., Remote operated vehicles—a driving force for improved outages. *Nuclear Engineering Int.*, Vol. 24, pp. 34–36, 1992.

[104] Ueno, M., Nimura, T., Ando, H., Maeda, K., and T., Kenkichi., On the descending motion of a deep-sea robot. *Control Engineering Practice*, Vol. 16, pp. 446-456, 2006.

[105] Nakamura, Y., and Savant, S., Nonlinear tracking control of autonomous underwater vehicles, *IEEE Int. Conf. on Robotics and Automation*, Vol. 3, pp. A4–A9, 1992.

[106] Nie, J., Yuh, J., Kardash, E., and Fossen, T.I., On-board sensor based adaptive control of small UUVs in the very shallow water, *IFAC-Control Applications in Marine Systems*, pp. 201–206, 1998.

[107] Norhayati, A.M., *A Behaviour-Based trajectory velocity learning control architecture for an autonomous mobile robot.* Master Thesis, Universiti Technology Malaysia, 2007.

[108] Edgar, P. and Smith, S.M., An experimental self-motion study of the ocean explorer AUV in controlled sea states, *IEEE J. Oceanic Engineering*, Vol. 23, pp. 274-284, 1998.

[109] Ishii, K., and Ura, T., An adaptive neural-net controller system for an underwater vehicle, *Control Engineering Practice*, Vol. 8, pp. 177-184, 2000.

[110] Choa, B.H., Byun, S.H., Shin, C.H, Yang, J.B., Songa, S.I., and Ohb, J.M., KeproVt, Underwater robotic system for visual inspection of nuclear reactor internals, *Nuclear Engineering and Design*, Vol. 231, pp. 327-335, 2004.