1998

# An experimental investigation of the fuzzy logic control of an inverted pendulum

Chen-Sheng Chou
*Lehigh University*

Chou, Chen-Sheng

# An Experimental Investigation of the Fuzzy Logic Control of an Inverted Pendulum

January 10, 1999

# An Experimental Investigation of the Fuzzy Logic Control of an Inverted Pendulum

by

Chen- Sheng Chou

A Thesis

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Mechanical Engineering and Mechanics

Lehigh University

November 24, 1998

This thesis accepted and approved in partial fulfillment of the requirements for the Master of Science.

12/8/1998

Date

Professor Meng S. Chew
Associate Professor

Professor Charles R Smith
Chairman of Department

## Acknowledgement

I would like to thank Professor Chew providing me with an opportunity to learn the fuzzy logic control for this thesis work. Professor Chew gave me many variable instructions and encouraged me to learn for myself. By this encouragement, I learn not only the knowledge for this thesis, but also the skills of solving problems independently. Also, I would like to thank my wife, Wen-Chi Lin, who gave me spiritual support and took care of housework. So that I maybe able to concentrate in the thesis work. Finally, I would like to thank Lehigh University for providing me with excellent library system. The abundant book collections gave me up-to-date knowledge and helped me to solve several difficult problems.

## Table of Contents

## List of tables

## List of Figures

## Nomenclature

e: Input Variable

$\Delta e$: Change of Input Variable

u: Output Variable

$\Delta u$: Change of output variable

U: Universal of Discourse

$\mu$: Menbership Function

min: Minimum Operation

max: Maximum Operation

$\cup$: Union Operation

$\cap$: Intersection Operation

V: Voltage

$V_I$: Input Voltage

$V_o$: Output Voltage

C: Capacitor

R: Resistor

$R_P$: Potentiometer Sensor

$R_G$: Resistor for Imstrumentation Amplifier Gain adjustment

G: Gain of the Instrumentation Amplifier

$\theta$: Angular Position of the Inverted Pendulum

$\omega$: Angular Velocity of the Inverted Pendulum

## Abstract

This research addresses the experimental investigation of fuzzy control of an inverted pendulum system. An inverted pendulum system is constructed. It consists of an inverted pendulum mechanism, a microcontroller, signal conditioning circuits, a DC motor, and a fuzzy logic controller. The fuzzy controller design is based on a trial-and-error approach. Four different values of fuzzy controllers were constructed to test the stability performance for the inverted pendulum system. The experiment simulations show that the fuzzy control is more robust than those conventional control methods. Finally, a discussion on the advantages of the fuzzy control is provided.

# Chapter 1

## Introduction

This thesis presents an experimental study of application of fuzzy logic control to the control of an inverted pendulum. In the experiment, an inverted pendulum is hinged on a cart. The cart moves linearly backward and forward to balance the rod pendulum. This is similar to a human balance a rod pendulum on his or her hand.

There are two reasons why engineers use the inverted pendulum as a control simulation tool. First, the mathematical model for the inverted pendulum is a nonlinear system, by which researches into the inverted pendulum can be transferred to many other nonlinear physical systems. Secondly, the inverted pendulum is an excellent unstable model for many control systems, such as, control of aircraft, rocket lunching, and chemical processing [1]. Engineers are interested in how to control these unstable systems. The engineers, therefore, use the inverted pendulum as a simple and cost efficient tool for nonlinear control simulation.

In the traditional control methods, such as, root locus, frequency response, and state space method, design of control algorithms is based on a linear mathematical model of the physical system. Since the most of the physical systems are nonlinear systems, such as the inverted pendulum, a linearization process is required to modify the nonlinear model into the linear model, so that the traditional control methods can be applied.

The disadvantage of linearization is that the linear mathematical model for the inverted pendulum system is not accurate. Also, when the angle of the pendulum exceeds an assumed angle, the linearization brakes down, because the errors from the linearized model become too large. Therefore, the linearized model is not good enough for the control of the physical systems.

Fuzzy logic control is a nonlinear control method, which provides better control performance than that from traditional controls. One special feature of fuzzy logic control is that it utilizes the expertise of humans to control the physical system, so that complex system can be controlled without extensive modeling of the relationship between the input and output of the system [2].

The earliest work for fuzzy logic control of inverted pendulum is done by Yamakawa [3], who designed a fuzzy logic control system to stabilize the inverted pendulum. Later, Zang [4] developed a two stage inverted pendulum utilizing fuzzy control with a trial-and-error method. Hwang [5] used a sliding mode to modeling the parameter of the rule-bases.

This thesis will concentrate in experimental design and fabrication of an inverted pendulum system controlled by fuzzy logic control. Simulations for the inverted pendulum system will be presented to demonstrate the advantages of fuzzy logic control of such nonlinear system.

The basic outline for this thesis as follows: Chapter 2 introduces the basic theory of fuzzy logic and fuzzy control. Chapter 3 explains the design of the inverted pendulum experiment, which includes the design for the inverted pendulum mechanism, the electrical circuits, and the fuzzy logic controller. Chapter 4 presents the experiment results. Chapter 5 presents a conclusion for the experiment.

# Chapter 2

## Theory of Fuzzy Logic Control

Lotfi A. Zadeh, professor of University of California at Berkeley, invented fuzzy logic in 1965. The theory of fuzzy logic brought forth a lot of controversies since his invention. Some scholars viewed fuzzy logic as an idea conflicts with basic scientific principles. Some believed that everything fuzzy logic can handle, conventional methods can handle as well and better [6]. After 20 years of development, an increasing number of fuzzy applications begin to appear in the market. The theory of fuzzy logic has been gradually accepted as a powerful tool for solving difficult problems.

The basic fuzzy theories for control application include fuzzy set, membership function, fuzzy inference process, and fuzzy logic controller design as will be explained in following sections.

### 2.1 Fuzzy Set and Membership Function

The concept of the fuzzy set came from the spirit of human operation, where human judges many things in life based their knowledge and experience. For an example, people sometimes judge that some people are "tall", but than what is the meaning of "tall" ?

In a crisp set, binary set elements are defined as either a member of a given set or they are not (1or 0 respectively). The meaning of "tall" can be presented as in Figure 2.1 by using the crisp set. Obviously, the crisp set can not fairly reflect the meaning of "tall", because two people with 250cm and 180cm in height are described with a same definition "tall". However, a person with 179.9cm in height is defined as "not tall", and this is obviously illogically.

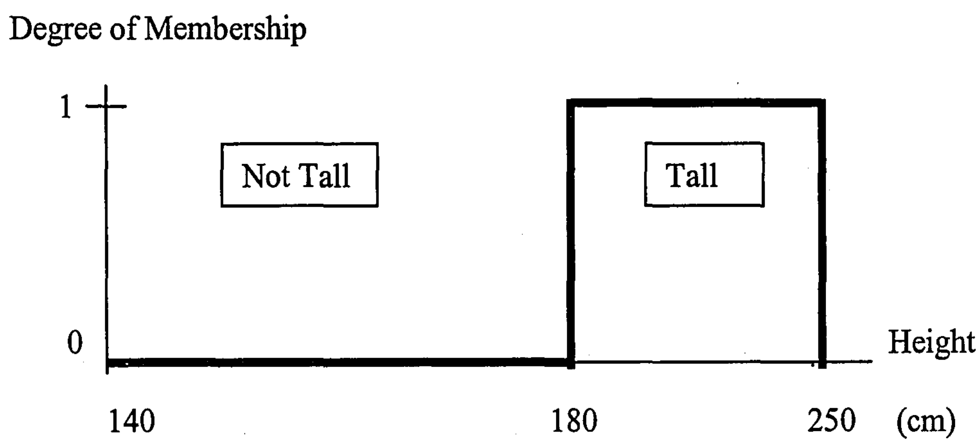



Figure 2.1 Definition of "tall " People Described by Crisp Set

The theory of fuzzy set has a different approach to define the vague description of the term "tall" in the previous example. The definition of fuzzy set has membership interval values going by degrees between 0 and 1. As the value approaches the value of 1, higher is the rank corresponding to the definition of the label. Therefore, a fuzzy set **A** is characterized by a membership function $\mu_A(u)$, which can be written as :

$$A=\{(u,\ \mu_A(u)/u,\ u\in U\} \tag{2.1}$$

where U is a universe of discourse and u is a generic element of U.

The example of that which consider as "tall " people by using the fuzzy set can be more accurately represented by a crisp set (Figure 2.2). For example, two people with 179cm and 180cm in heights may be defined as "average tall" with membership degree of 0.39

and 0.40 respectively; A person with 200cm in height is “very tall” with membership degree of 0.8.

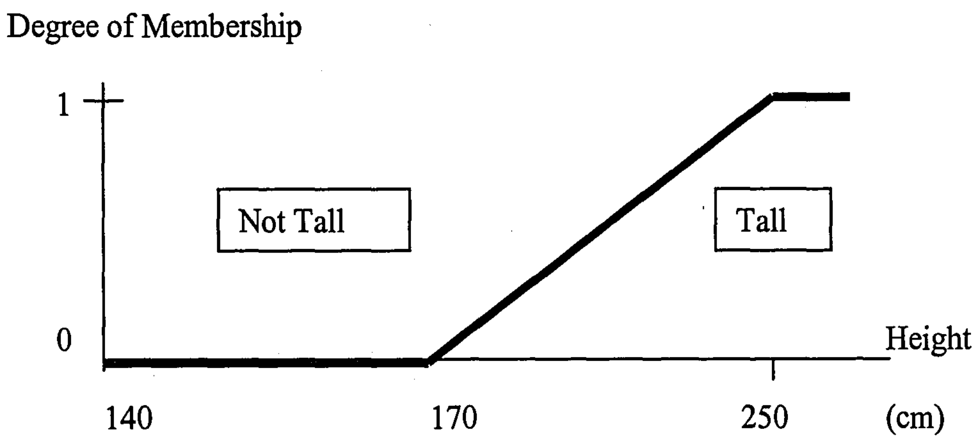



Figure 2.2 Definition of "tall " People Described by Fuzzy Set.

## 2.2 IF-THEN Rule

The concept of the fuzzy control is based on human fuzzy decriptions. Human tends to use their knowledge, experience, and intuition to control the physical system in the real world. Consider an automotive cruise control example with a single input and single output (SISO) control variable by using the IF-THEN rules (Table 2.1)

Table 2.1 shows that IF-THEN rules are linguistic decision making processes, where the terms come after the IF rules is called antecedents, and terms come after the THEN rules is called consequences.

| | |
|---|---|
| $R_1$: | **IF** the error of car's speed is positive big **THEN** put a big force to the accelerator |
| $R_2$: | **IF** the error of car's speed is modulate **THEN** put a modulate force to the accelerator |
| $R_3$: | **IF** the error of car's speed is negative big **THEN** put a small force to the accelerator |

Table 2.1 IF-THEN Rules for an Automotive Cruse Control Example.

The operation for an IF-THEN rule is that if the conditions of the antecedent parts are satisfied, then the condition of the consequent part will be active. Mamdani is the first researcher introducing the ideal of IF-THEN rules into the real world applications [7]. The method developed by Mamdani is called Mamdani-type fuzzy logic controller, which is described in Section 2.4.

## 2.3 Fuzzy Inference Process

The fuzzy inference process is a methodology for processing the membership functions and the IF-THEN rules to obtain the fuzzy output results. Basically, the inference process consists of four steps: fuzzification, rule evaluation, aggregation, and defuzzification (Figure 2.3).

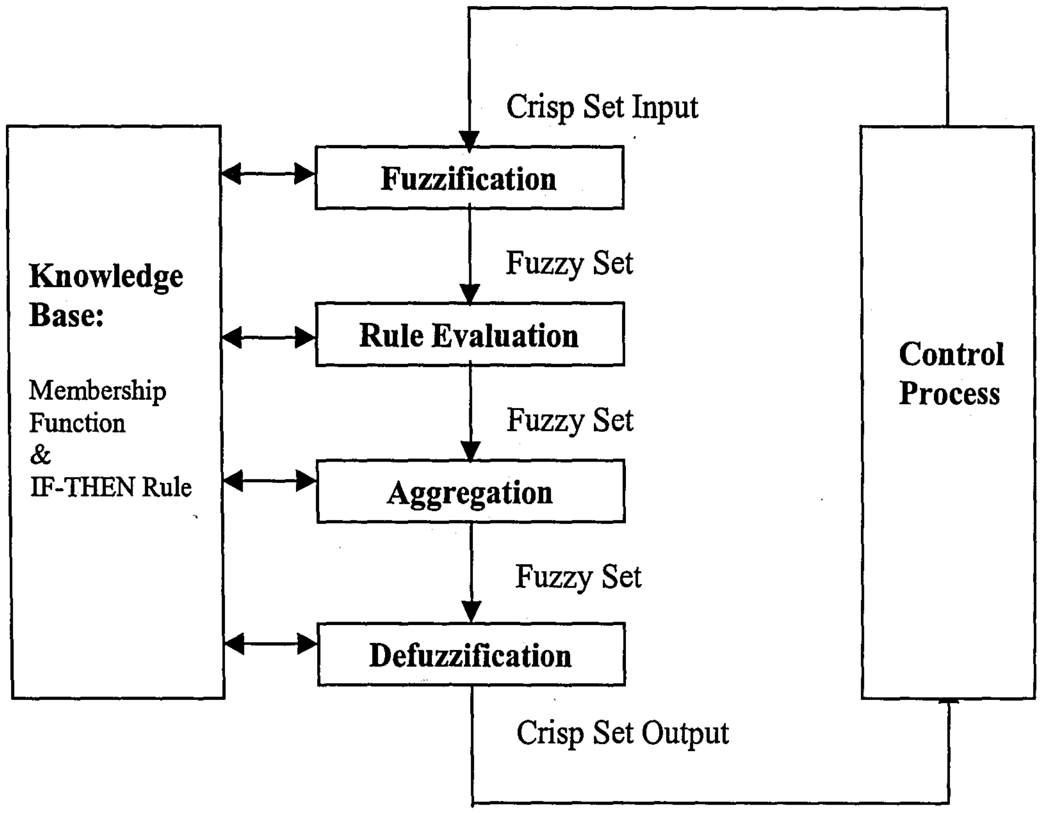



Figure 2.3 Flow Chart for the Fuzzy Inference Process

The four steps in the methodology for the inference process are described as below.

1) Fuzzification: The fuzzication process transforms the crisp input values into the grades of the membership functions (Figure 2.4), where $\mu_{A1}(x_0)$, $\mu_{A2}(x_0)$, $\mu_{B1}(y_0)$, and $\mu_{B2}(y_0)$ are the fuzzification values based on input crisp values $x_0$ , $y_0$ and the membership functions $A_1$, $A_2$, $B_1$, and $B_2$ respectively.

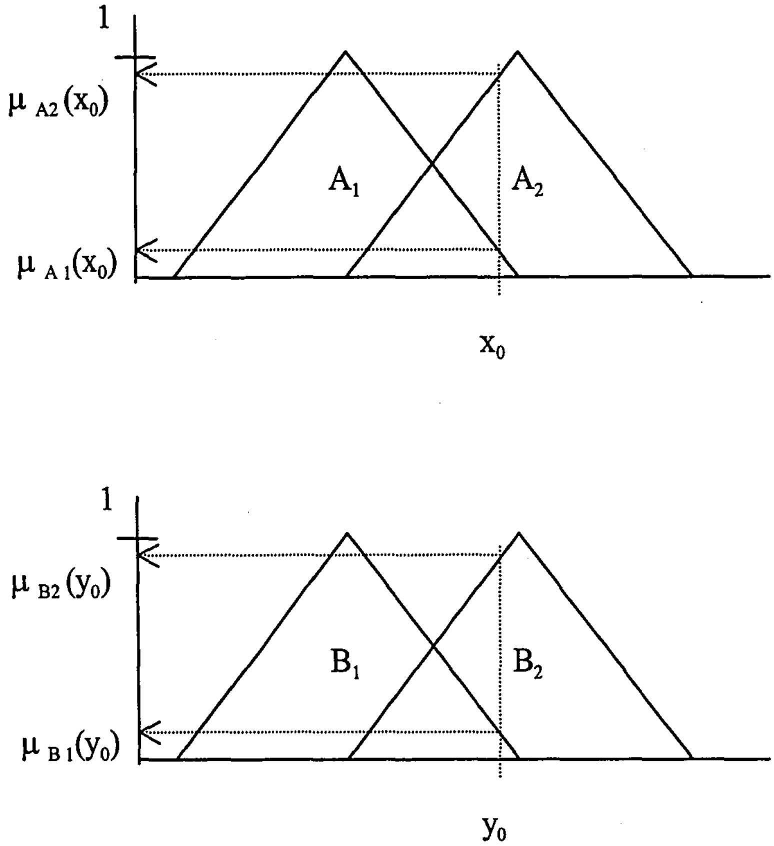



Figure 2.4 Fuzzication for Membership Function $A_1$, $A_2$, $B_1$, and $B_2$.

2) Rule Evaluation: The rule evaluation processes evaluate the fuzzy results from individual IF-THEN rules. Figure 2.5 shows a single rule evaluation process, where the IF-THEN rule is "If **x** is $\mathbf{A_j}$ and **y** is $\mathbf{B_j}$ then **z** is $\mathbf{C_j}$". The "and" operation in the IF-THEN rule is an intersection, which is described as $A \cap B$. The intersection between the membership functions $\mu_{Ai}(w)$ and $\mu_{Bi}(w)$ can be interpreted as a minimum of these two membership functions (Figure 2.5). The definition for the minimum can be defined as:

$$\mu_{Ai \cap Bi}(w) = \min\{\mu_{Ai}(w), \mu_{Bi}(w)\}, \quad w \in U \tag{2.2}$$

The next step is to apply the implication method to the rule evaluation result, which is to perform another intersection operation between the membership functions of $\mu_{(A \cap B)}$ and $\mu_C$ (Figure 2.5). The results of implication processes can be evaluated by another minimum operation (Equation 2.3) (Figure 2.5).

$$\mu_{(Ai \cap Bi) \cap Ci}(w) = \min\{\mu_{(Ai \cap Bi)}(w), \mu_{Ci}(w)\}, \quad w \in U \tag{2.3}$$

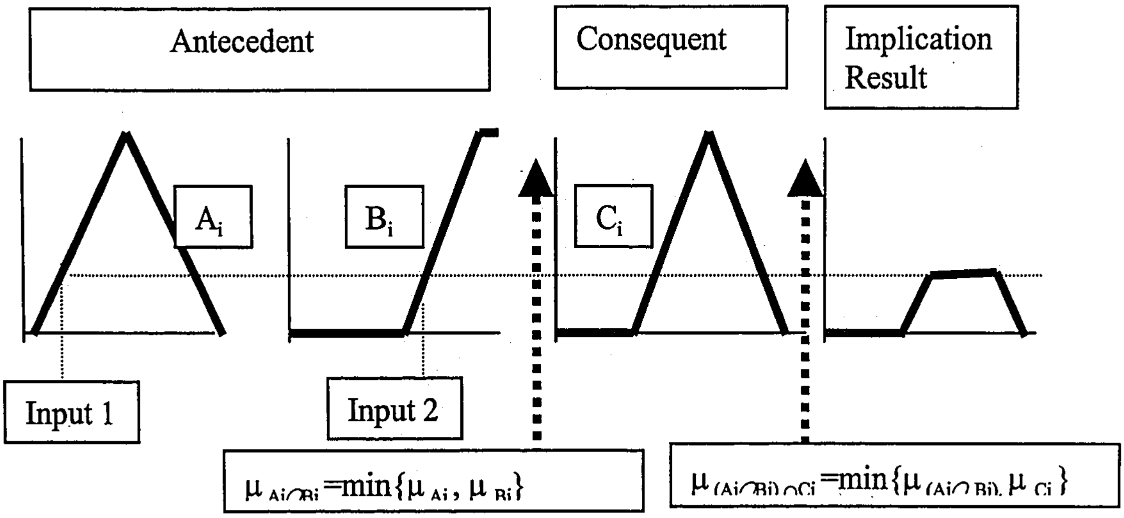



Figure 2.5 Graphical Interpretation of Rule Evaluation

3) Aggregation: The aggregation processes summarize all the implication results, which is to perform a union operation. The unit operation is to perform a maximum operation to all the implication results. Equation 2.4 and 2.5 define the union and the

maximum operation respectively. Figure 2.6 shows the aggregation result by using the graphic interpretation method.

$$\mu_C(w) = \{\mu_{C1}(w) \cup \mu_{C2}(w) \cup \ldots \cup \mu_{Ci}(w)\}, w \in U \qquad (2.4)$$

$$\mu_C(w) = \max\{\Sigma \mu_{Ci}(w)\}, w \in U \qquad (2.5)$$

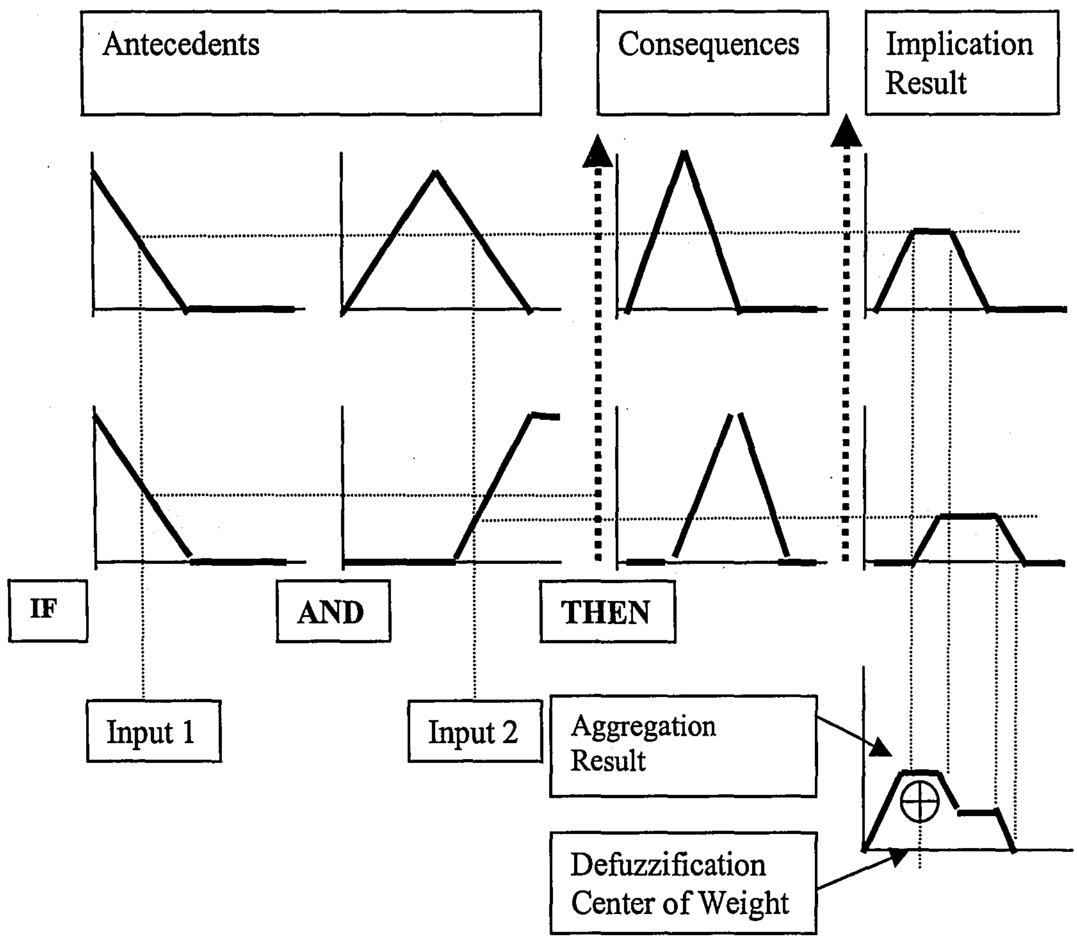



Figure 2.6 Graphic Interpretation Method for the Fuzzy Inference Processes

4) Defuzzification: The fuzzy result obtained from the aggregation process is a particular shape of membership function in the fuzzy domain, which can not be used directly. The defuzzification process, therefore, transfers the fuzzy output values from the fuzzy set domain into a crisp set domain.

There are several different strategies for defuzzification, such as the maximum criterion method, the mean of the maximum method, and the center of weight method, etc. One of the most common strategies is applying the center of weight method, which calculates the center of the gravity from the aggregation result. In case of a defuzzification result in a discrete universe, the equation can be presented as Equation 2.6:

$$C_0 = \frac{\sum_{i=1}^{n} \mu_c(w_i) \cdot w_i}{\sum_{i=1}^{n} \mu_c(w_i)} \qquad (2.6)$$

## 2.4 TVFI Inference Method

Truth-Value Flow Inference (TVFI) method (Figure 2.7) uses singletons in the output membership functions, which simplifies the calculation of the inference processes and deduces the fuzzy inference time performed by the microcontroller.

Figure 2.7 shows the fuzzy inference process by the TVFI method, where the output membership functions have singleton values. After the inference process, the implication, aggregation, and fuzzification results will have the singleton values also.

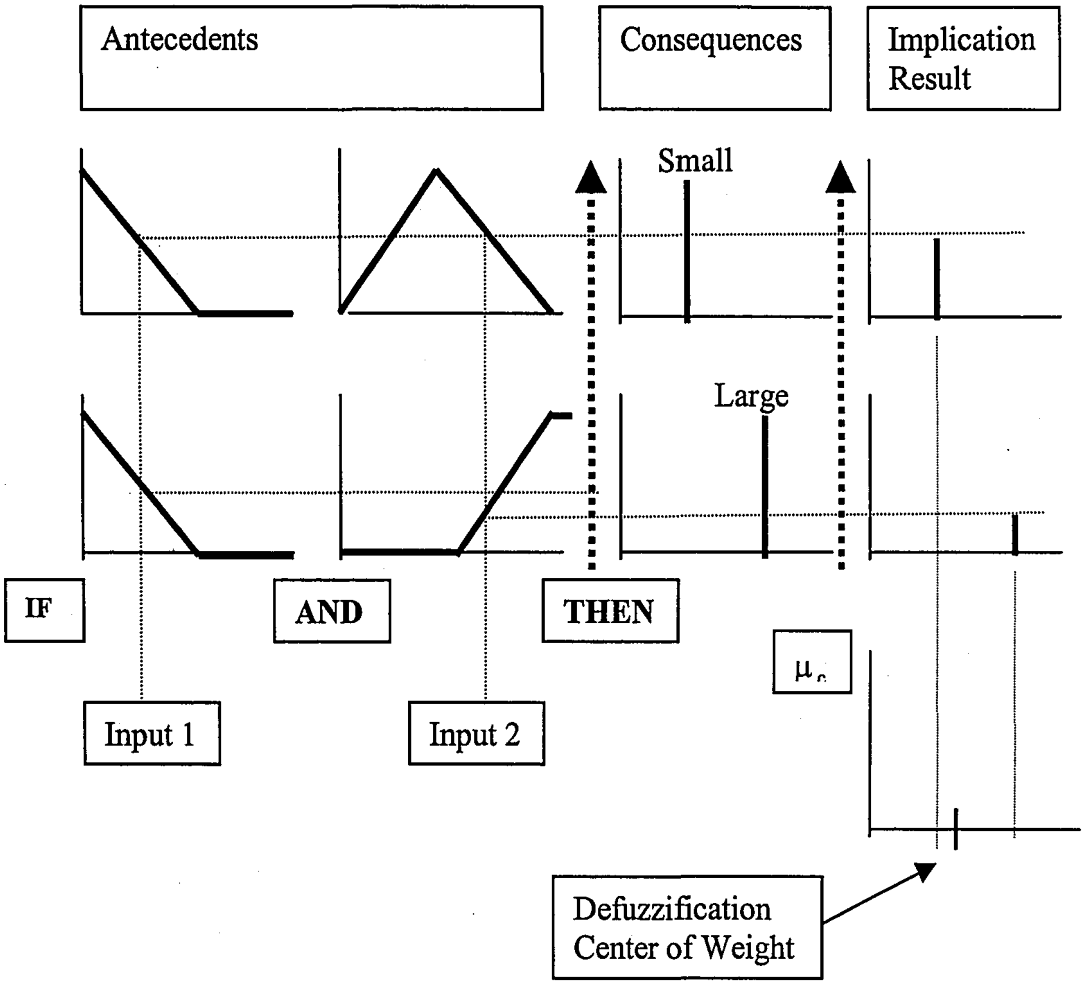



Figure 2.7 Graphic Interpretation for a TVFI method.

## 2.4 Structure of a PD-Type Fuzzy Control System

The structure of fuzzy control system is similar to conventional discrete control system. Equation 2.7 indicates a control law for a discrete control system.

$$u(k+1) = F( e(k), e(k-1),\ldots,e(k-m+1), u(k), u(k-1),\ldots,u(k-m+1) ) \tag{2.7}$$

or

$$u(k) = F\left( \sum_{m=0}^{m} e(k-m),\ \sum_{m=1}^{m} u(k-m) \right) \tag{2.8}$$

Where e(k),...e(k-m+1) are the input variables and u(k-1),...,u(k-m+1) are the output variables

In conventional discrete control system, the control law F is described in difference and transfer equations, while the fuzzy system uses linguistic variables in the control law F. Mamdani combines this idea with the IF-THEN rules thereby developing the first fuzzy logic controller (Equation 2.9):

$R_i$: IF e(k) is $A_{i,1}$ and e(k-1) is $A_{i,2}$ and,... e(k-m+1) is $A_{i,n}$

and u(k) is $B_{i,1}$ and u(k-1) is $B_{i,2}$ and,... u(k-m+1) is $B_{i,m}$

THEN u(k+1) is $C_i$ (2.9)

Where $R_i$ represents the ith rule, i=1,2,3...u(k+1) is the output results.

For m=1, a two-input and single output case, the Mamdani-type fuzzy logic controller can be presented as Equation 2.10:

$R_i$: IF e(k) is $A_{i,1}$ and $\Delta$e(k) is $A_{i,2}$

and u(k) is $B_{i,1}$ and $\Delta$u(k) is $B_{i,2}$

THEN $\Delta$u(k) is $C_i$ (2.10)

The structure of a PD-type fuzzy logic control system is shown in Figure 2.7, where e(t) is the error, Δe(t) is change of error, u(t) is output, and Δu(t) is change of output.

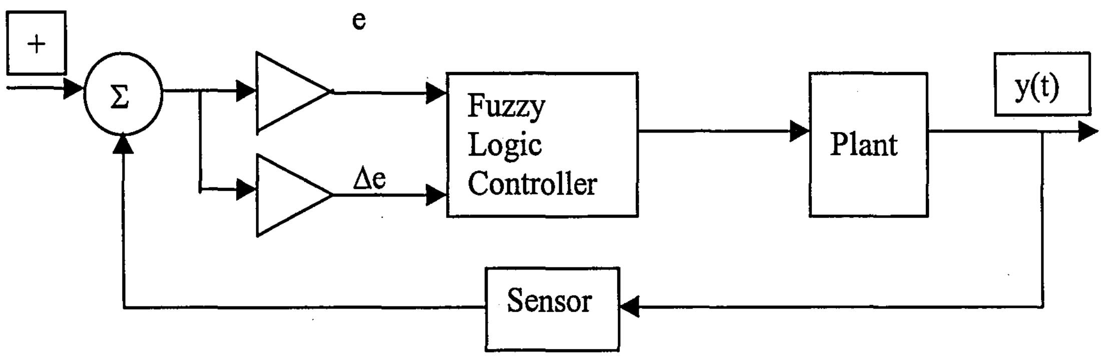



Figure 2.8 Block Diagram for a PD-Type Fuzzy Controller

We will utilize this PD-type fuzzy controller with the rule-bases to control the inverted pendulum system.

## Charter 3

### Design of an Experiment for Fuzzy Control of an Inverted Pendulum

Designing an experiment for the fuzzy control of inverted pendulum calls upon a synthesis of knowledge in both mechanical and electrical engineering. The basic experiment set-up (Figure 3.1) consists of a host personal computer, a microcontroller, a potentiometer sensor, signal conditioning circles, a motor driver, a DC geared motor, an inverted pendulum, and drive mechanism.

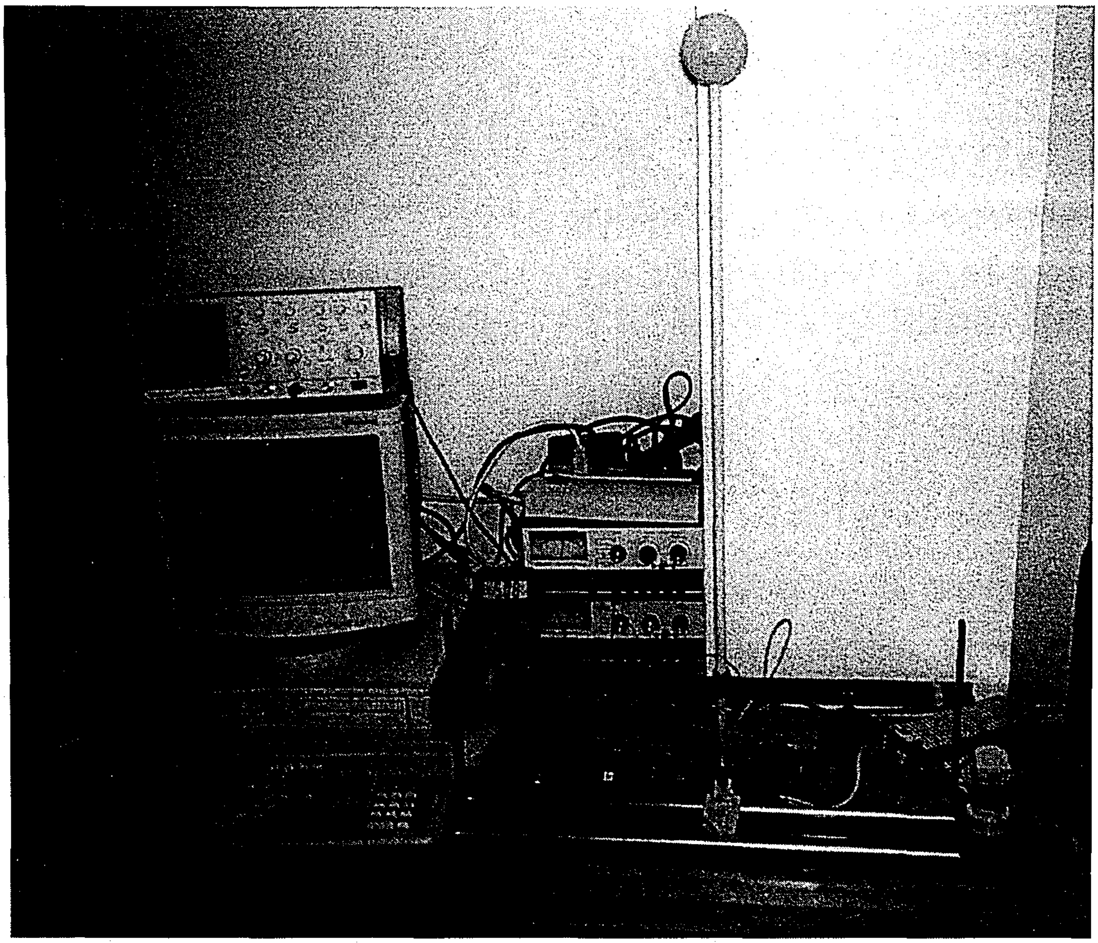


Figure 3.1 Experiment Set-Up for the Fuzzy Control of an Inverted Pendulum

# Charter 3

## Design of an Experiment for Fuzzy Control of an Inverted Pendulum

Designing an experiment for the fuzzy control of inverted pendulum calls upon a synthesis of knowledge in both mechanical and electrical engineering. The basic experiment set-up (Figure 3.1) consists of a host personal computer, a microcontroller, a potentiometer sensor, signal conditioning circles, a motor driver, a DC geared motor, an inverted pendulum, and drive mechanism.

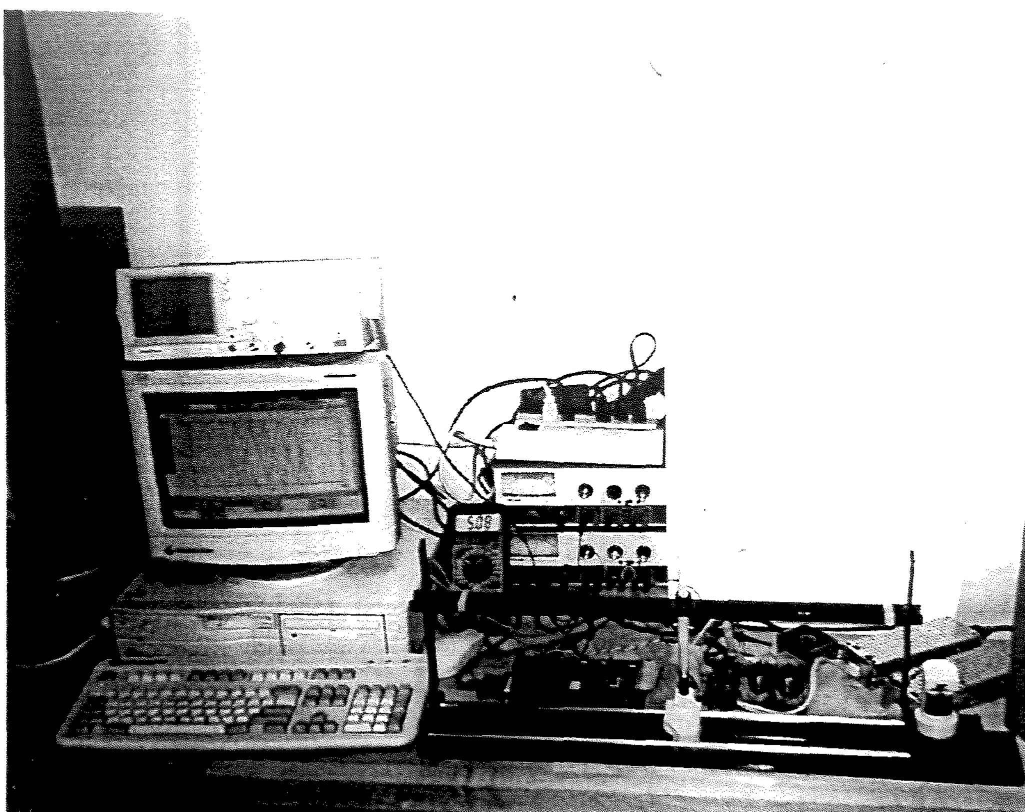


Figure 3.1 Experiment Set-Up for the Fuzzy Control of an Inverted Pendulum

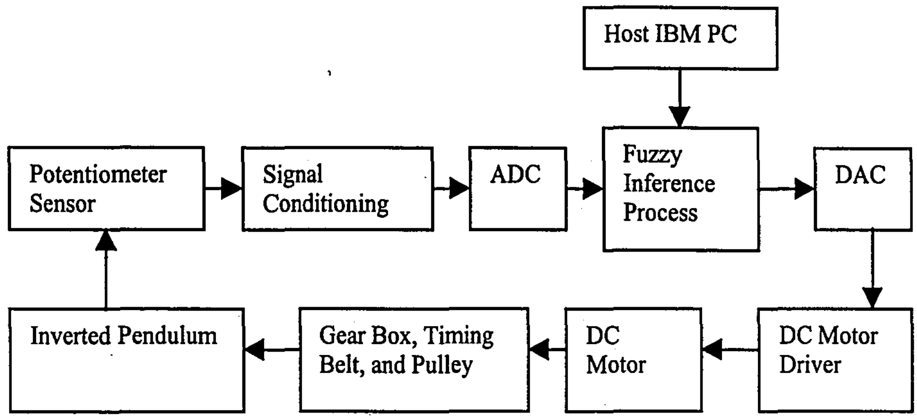



Figure 3.2 Block Diagram for the Fuzzy Control of the Inverted Pendulum.

The block diagram (Figure 3.2) about illustrates the fuzzy control processes, where the host computer is used to edit the fuzzy inference program, to compile the Motorola assembly program, and to interface the fuzzy control code into the RAM of the microcontroller. The actually control actions are performed by the microcontroller, which samples the input data from the sensor, calculates fuzzy inference results, and then control the speed of the DC motor. Finally, the DC motor drives the cart to balance inverted pendulum rod.

Basically, the design of the inverted pendulum experiment can be categorized into three parts: (a) Inverted Pendulum mechanism, (b) electric circuits, and (c) fuzzy logic controller.

## 3.1 Design of an Inverted Pendulum Mechanism

The inverted pendulum mechanism consists of a cart and a rod pendulum with a potentiometer sensor and a DC gear-head motor mounted in the mechanism. For a well-built experiment, the design criteria for the inverted pendulum mechanism are:

1) The cart must be driven smoothly in a linear motion.
2) The cart is a lightweight (0.1 kg) for a higher accelerating rate and a lower friction loss.
3) The DC-motor-drive-train must securely drive the cart without sliding

After the final design was decided, the inverted pendulum mechanism was constructed at Lehigh University with following features (Figure 3.2):

1) The cart is securely driven by a pair of 3mm timing belt and pulley.
2) The cart is tracked by two pieces of precision bars with a linear accuracy of 1/1000.
3) The cart is constructed of engineering plastic with a mass of 0.1 kg.
4) The friction of the siding surface between the cart and the tracks are reduced by two pieces of bearings coated with a Teflon film. The Teflon film also reduces the electrical noise generated by relative motion between the track and the bearings.
5) The base of the mechanism is constructed by a piece of heavy aluminum plate, which absorbs some vibration and provides a stable structure to the mechanism.

## 3.2 Design of Electric Circuits

The electric circuits consist of a microcontroller, a potentiometer sensor, a motor driver, and signal conditioning circuits. Basically, the electric circuits are constructed by different functions of operational amplifiers, such as an instrumentation amplifier, a differential amplifier, differentiator amplifier, and a motor power op-amp. To transfer signals between the analog and digital world, an analog-to-digital converter (ADC) and a digital to analog converter (DAC) is used between the digital microcontroller and the analog control circuits (Figure 3.3).

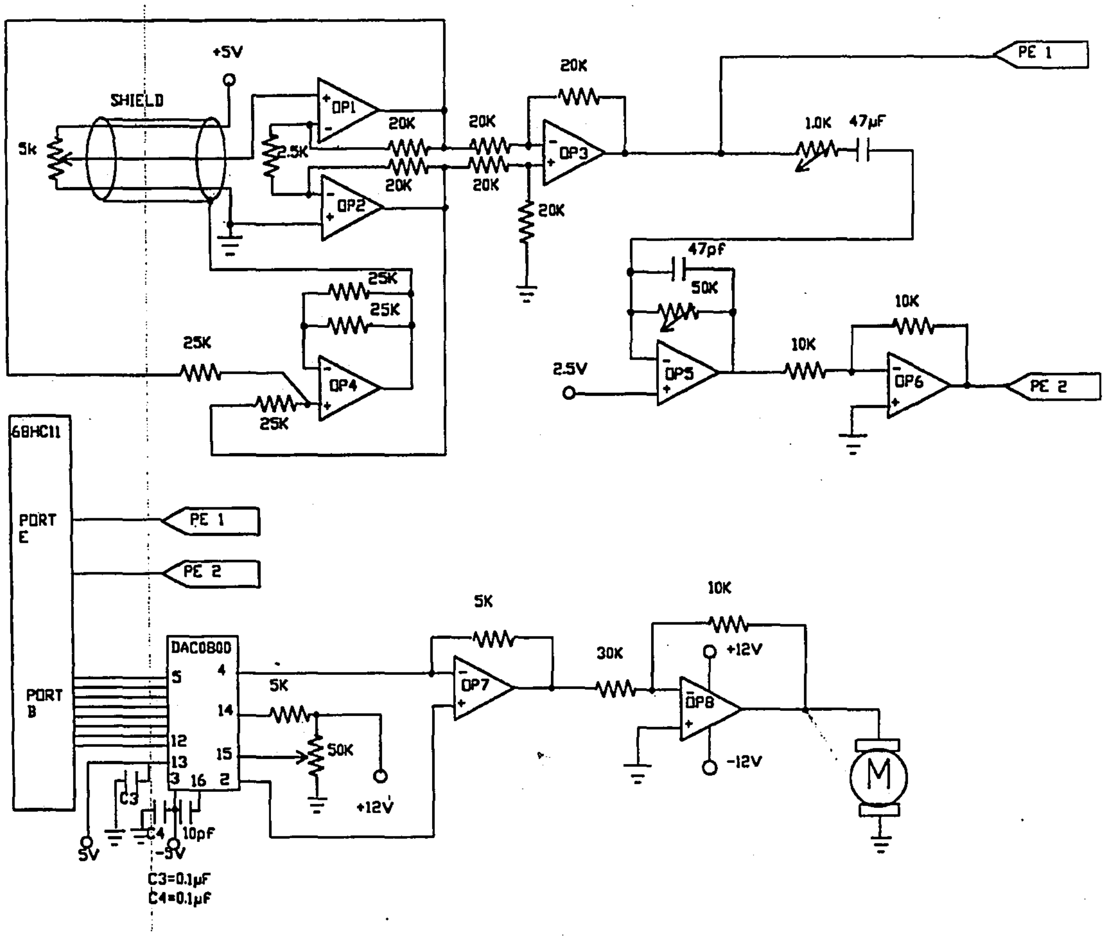



Figure 3.3 Electric Circuits for the Experiment

### 3.2.1 Instrumentation Amplifier

The instrumentation amplifier (Figure 3.4) is a high input impedance device, which provides a gain mapping the data into the universe of the membership function for the fuzzy inference processes. The gain of the instrumentation amplifier is determined by the value of the resistor $R_G$.

The gain G for the instrumentation amplifier in the Figure 3.4 can be presented as:

$$G = \frac{V_o}{(V_{i2} - V_{i1})} \tag{3.1}$$

Where $V_o$ is the output signal voltage, and $V_{i1}$ and $V_{i2}$ are the input signal voltage for the op-amps OP1 and OP2 respectively. The voltage output $V_{o1}$ of the op-amp OP1 can be solve as a non-inverting input of OP1 minus the inverting input of the op-amp OP2 through the resistor $R_G$:

$$V_{o1} = V_{i1}\left(\frac{R_1}{R_G}+1\right) - V_{i2}\left(\frac{R_1}{R_G}\right) \tag{3.2}$$

The same reason for the op-amp OP2

$$V_{o2} = V_{i2}\left(\frac{R_2}{R_G}+1\right) - V_{i1}\left(\frac{R_2}{R_G}\right)$$

(3.3)

The op-amp OP3 has a gain of 1, therefore the signal output of op-amp OP3 is:

$$V_o = (V_{o2} - V_{o1}) \tag{3.4}$$

Substituting the:

$$G = 1 + \frac{40K\Omega}{R_G} \tag{3.5}$$

The rotational range for potentiometer $R_P$ is from 0° to 340° with the voltage output range from 0 to 5 volt. If the potentiometer rotates from 164.5° to 175.5° ( for the inverted pendulum oscilating within ± 5.5°) and the voltage output range between 0 and 5 volt. The voltage output range must be amplified. This amplified gain is given by 340/11=31.91.

Therefore, the gain required for the imstrumentation amplifier is 31.91. Substituting this gain into the Equation 3.5. The value for the resistor $R_G$ is to be 1.294 KΩ. An adjustable resistor is needed to obtain this value.

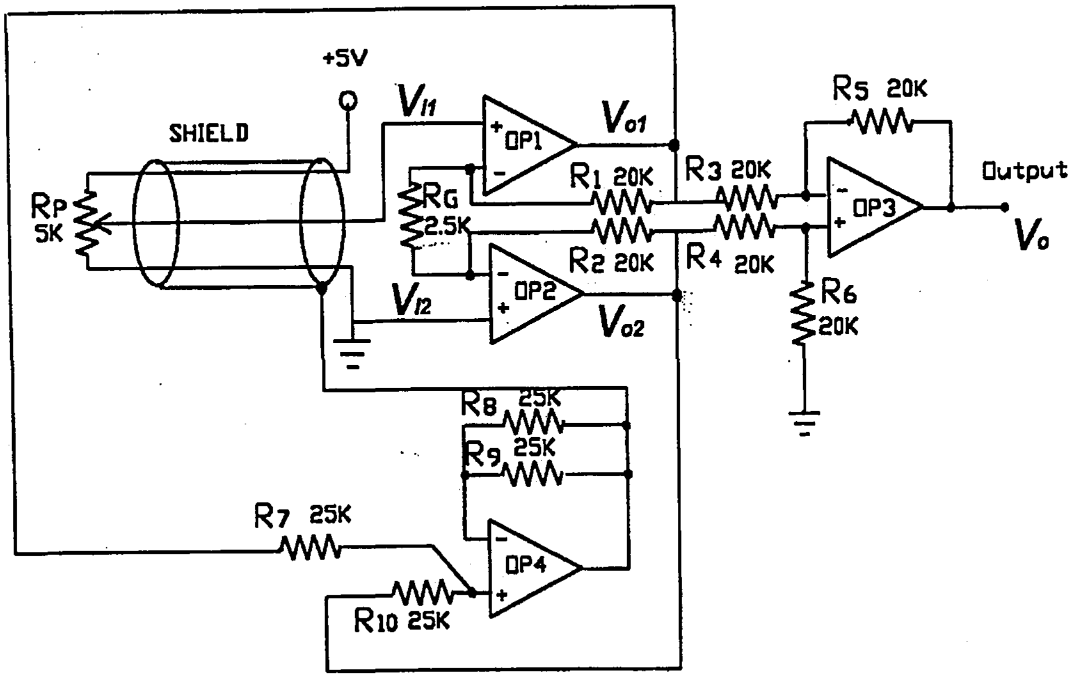



Figure 3.4 Instrumentation Amplifier (Op-Amps OP1, OP2, and OP3) , Potentiometer Sensor ($R_P$), Differential Amplifier, and Shield Guard [8].

### 3.2.2 Active Shield Guard

A 60 Hz external noise was one of the most troublesome problems for signal conditioning in the experiment, which provides the fuzzy inference process with unstable fuzzy inference processes and made the experiment fail. The noise source came from the capacitive coupling effect, which caused by the unshielded signal wires receiving the 60 Hz frequency from the local AC power source to the signal conditioning circuits. The solution for this problem is to construct a shield guard to the signal wires and drive the shield guard by an differential op-amp to protect the signal wires from the coupling effect and conduct the 60 Hz noise in to the ground (Figure 3.4).

### 3.2.3 Differentiator

The differentiator (Figure 3.5) provides the fuzzy controller with the angular velocity input data. The current at the inverse input of op-amp OP5 can be written as:

$$I_i = C_1 \,( dV_i / dt \,) \tag{3.6}$$

Then the output voltage $V_O$ can be written as:

$$V_o = I_i R_{12} = -R_{12}\, C_1 \,( dV_i / dt\,) \tag{3.7}$$

Because of the internal phase shifts and high gain in the op-amp OP5, the differentiator tends to be unstable and requires resistor $R_{11}$ and Capacitor $C_2$ to roll-off some noise frequencies. The values of resistor $R_{11}$ and capacitor $C_2$ is decide by trial-error method with $C_2$=47pf fixed and adjusting the value for resistor $R_{11}$ (50KΩ).

The inverted output of op-amp (OP6) is used to shift the signal direction back to the coordinate definition, because the differentiator provides an opposite output signal direction from the experiment's coordinate system.

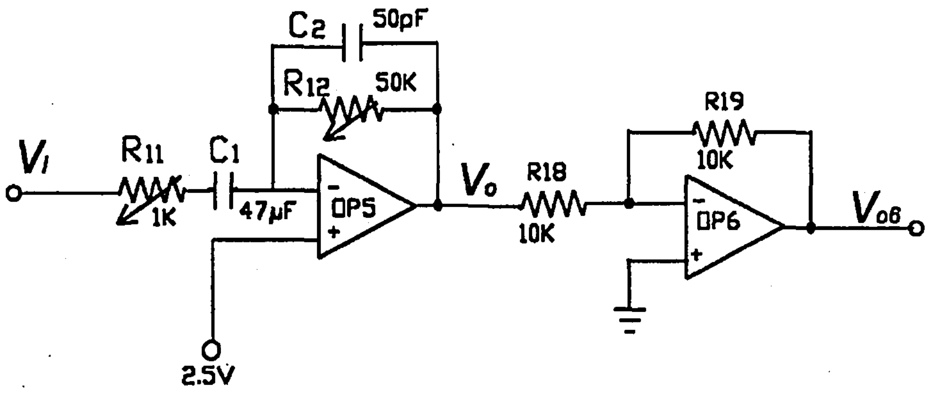



Figure 3.5 Differentiator (OP5) and Inverter (OP6)

### 3.2.4 Digital to Analog Converter

The experiment utilizes a National Semiconductor DAC0800, 8-bit of DAC, to convert the digital output from the microcontroller to the analog world (Figure 3.6). The op-amp after the DAC is used to provide with a bipolar output operation for control of speed and direction of the DC motor.

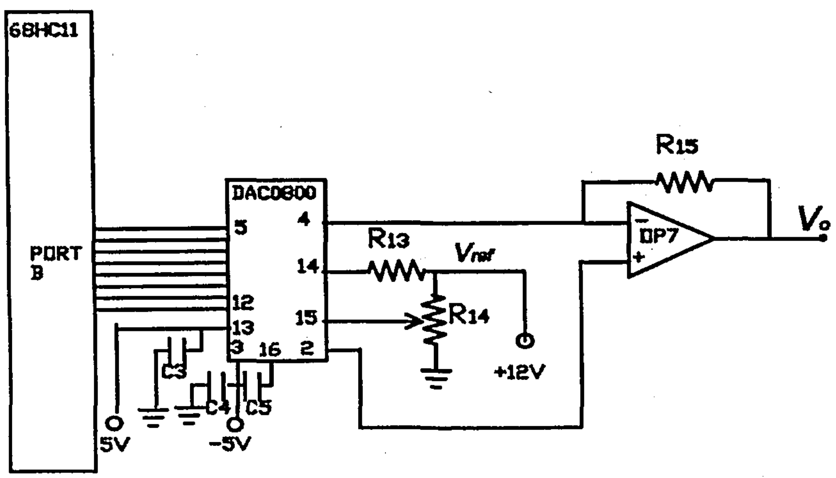



Figure 3.6 DAC with Symmetrical Bipolar Output Operation [9].

### 3.2.5 The Motor Driver

The motor driver was constructed by a National Semiconductor LM12CLK power op-amp (Figure 3.7). This op-amp is capable of providing a continue power up to 80 watt to drive the DC motor. The connection for the power op-amp is an inverse input and a gain of three, by which the DC motor can be controlled by a signal input between –5V and +5V and driven by a current between –15V to +15V.

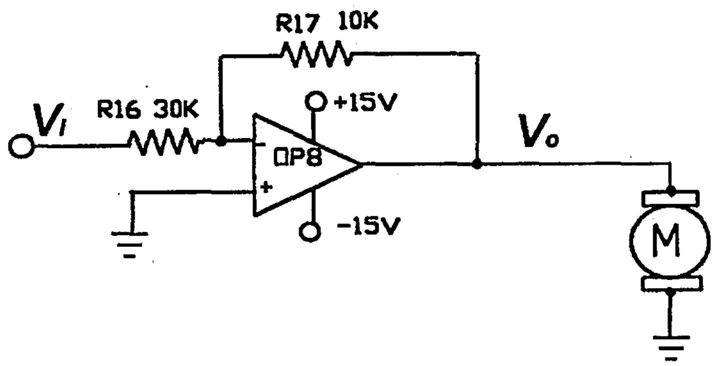



Figure 3.7 LM12 DC Motor Driver and DC Motor [10].

## 3.3 Fuzzy Logic Controller Design

Generally, there are four design procedures for conventional control: (1) Analyze the mode of the operation of the physical system. (2) Apply the model into a controller. (3) Simulate the performance of controller and adjusting the model parameters. 4) Apply the controller design to the physical system and adjust the modeling parameters for an improved control performance.

Since the special feature of fuzzy control contains knowledge-based IF-THEN rules and membership functions, the system modeling procedure are eliminated and apply with a trial-and-error method. Generally, fuzzy controller design can be reduced to two steps: (1) Design the rule-bases and the membership functions. (2) Simulate the controller in the physical system and adjust the IF-THEN rules and membership functions for an improved control performance.

### 3.3.1 Modeling of Fuzzy Parameter

The fuzzy controller for the inverted pendulum system has two input and one output variables, which can be defined by fuzzy sets **A**, **B** and **C** for angular position input, angular velocity input, and output force respectively.

$$\mathbf{A} = \{\, \mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_7 \,\} = \{\, \mathbf{NL}, \mathbf{NM}, \mathbf{NS}, \mathbf{ZO}, \mathbf{PS}, \mathbf{PM}, \mathbf{PL} \,\} \tag{3.8}$$

$$\mathbf{B} = \{\, \mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_7 \,\} = \{\, \mathbf{NL}, \mathbf{NM}, \mathbf{NS}, \mathbf{ZO}, \mathbf{PS}, \mathbf{PM}, \mathbf{PL} \,\} \tag{3.9}$$

$$\mathbf{C} = \{\, \mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_7 \,\} = \{\, \mathbf{NL}, \mathbf{NM}, \mathbf{NS}, \mathbf{ZO}, \mathbf{PS}, \mathbf{PM}, \mathbf{PL} \,\} \tag{3.10}$$

Where **N**: Negative **P**: Positive **L**: Large **M**: Medium **S**: Small **ZO**: Zero

Applying the parameters **A**, **B**, and **C** into the membership functions, we can plot the membership functions as Figure 3.8, 3.9, and 3.10. The definition for the coordinates is indicated in Figure 3.11.

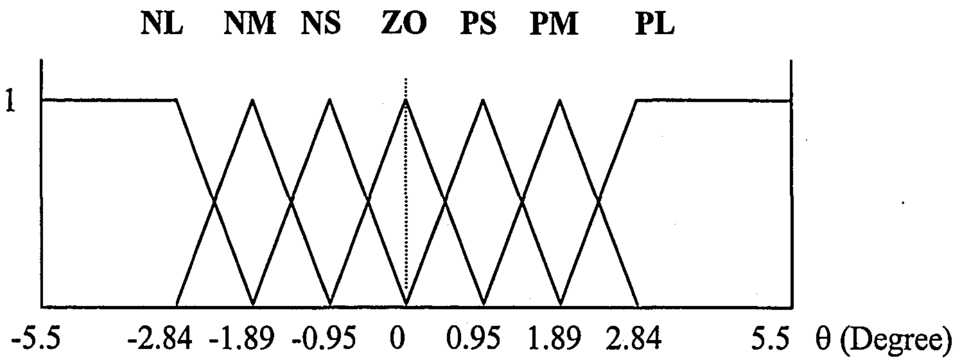



Figure 3.8 Membership Functions for the Angular Position Input.

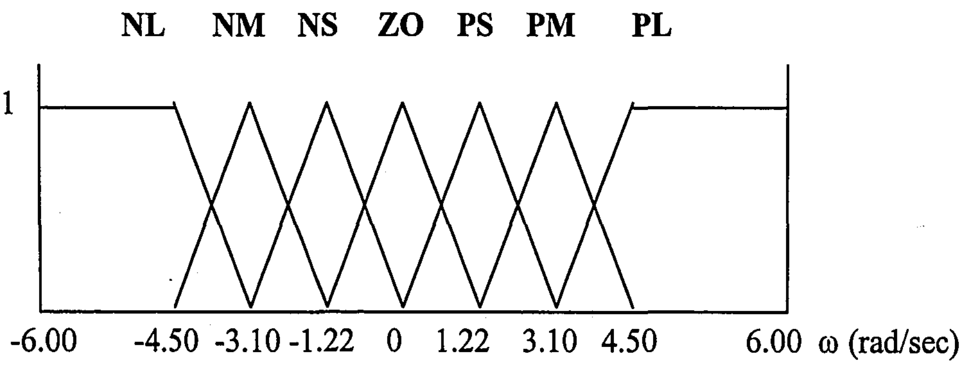



Figure 3.9 Membership Functions for the Angular Velocity Input.

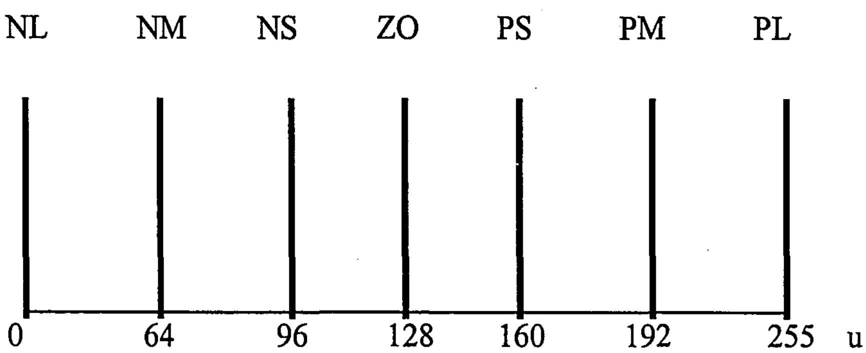



Figure 3.10 Membership Functions for the Force Output.

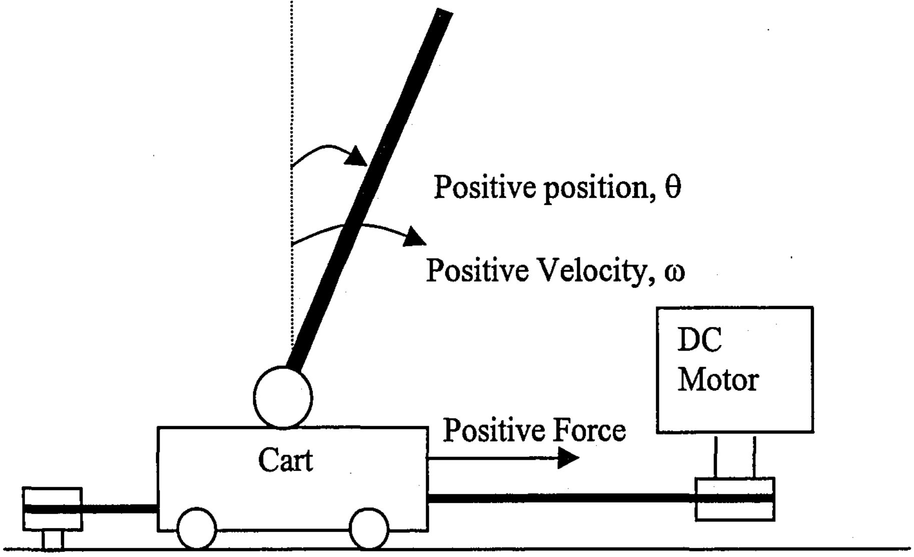



Figure 3.11 Coordinate System for the Inverted Pendulum System

### 3.3.2 Modeling of IF-THEN Rules

The fuzzy control is a rule-based control system. The rules express that if the antecedents are satisfied, then the consequent part will be performed, where antecedent parts are the input variable and the consequence parts are the output variables.

The construction of IF-THEN rules is based on an analysis of the tracking the path of an inverted pendulum rod with respect to the phase plane and the step response plot to get a approximate desired trajectory respond (Figure 3.12). The analytical procedures are listed in Table 3.1 [11] [12].

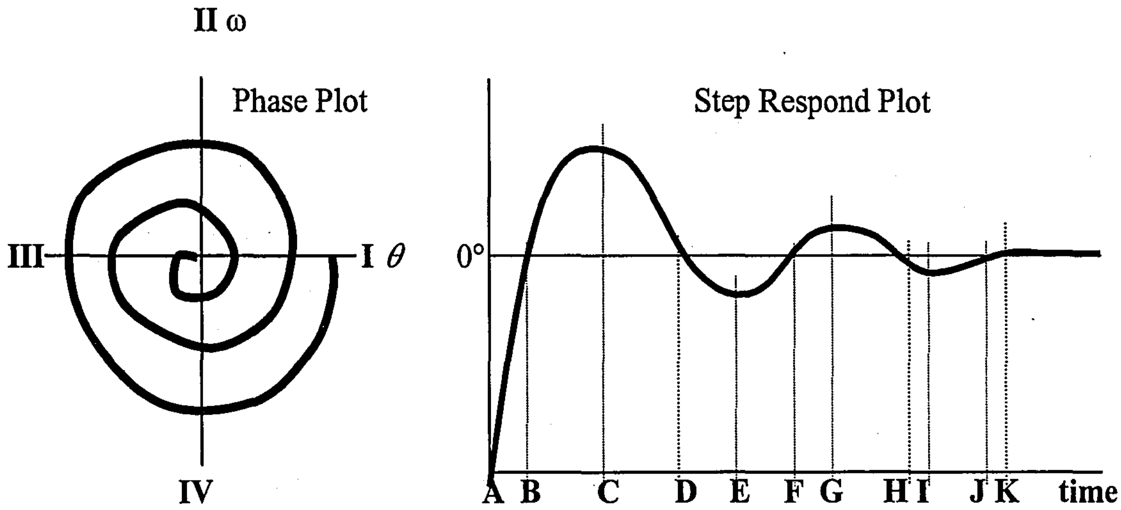



Figure 3.12 Phase Plane and Step Response for the Rule Justification

| Region in the Phase Plane | Point in the Step Response | Corresponded IF-THEN Rule |
|---|---|---|
| I | A | **IF** theta is PL **and** omega is ZO **THEN** voltage is PL |
| IV | B | **IF** theta is ZO **and** omega is NL **THEN** voltage is NL |
| III | C | **IF** theta is NL **and** omega is ZO **THEN** voltage is NL |
| II | D | **IF** theta is ZO **and** omega is PL **THEN** voltage is PL |
| I | E | **IF** theta is PM **and** omega is ZO **THEN** voltage is PM |
| IV | F | **IF** theta is ZO **and** omega is NM**THEN** voltage is NM |
| III | G | **IF** theta is NM **and** omega is ZO **THEN** voltage is NM |
| II | H | **IF** theta is ZO **and** omega is PS **THEN** voltage is PS |
| I | I | **IF** theta is PS **and** omega is ZO **THEN** voltage is NS |
| IV | J | **IF** theta is ZO **and** omega is NS **THEN** voltage is NS |
| III | K | **IF** theta is ZO **and** omega is ZO **THEN** voltage is ZO |

Table 3.1 Analytical Procedures for the Rule Justification. The IF-THEN Rules are analyzed from the phase plan and step response plot (Figure 3.12).

There are total 36 IF-THEN rules (6×6), which is constructed by six angular position input labels, six angular velocity input labels, and six output variables. Table 3.1 presents the analytical procedures leading the system toward a stable condition. The IF-THEN rules show in the Table 3.1 called the fundamental rules. With the help of the fundamental rules, the rest of the IF-THEN rules can be easily defined. Table 3.2 shows the fundamental rules in the matrix of the state variable form.

| | | Angular Velocity, ( $\omega$ ) | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | PL | PM | PS | ZO | NS | NM | NL |
| Angular Position ( $\theta$ ) | PL | | | | PL | | | |
| | PM | | | | PM | | | |
| | PS | | | | NS | | | |
| | ZO | PL | | PS | ZO | NS | NM | NL |
| | NS | | | | | | | |
| | NM | | | | NM | | | |
| | NL | | | | NL | | | |

Table 3.2 Fundamental Rules from the Analytical Procedures in Table 3.1.

The IF-THEN rules can be easily defined by analyzing the correlation between the adjacent rules in Table 3.2. For example, in the first row, the only choice for the linguistic terms in the left-hand side of PL is PL and in right hand side of the PL, the values are gradually decreased such as PM, PS, and NS.

After filling in all the remaining rules, a complete set of IF-THEN rules is presented in Table 3.3. The IF-THEN rules in the Table 3.3 show that the linguistic labels is divided into two groups, a positive and a negative group. Each group is symmetrically arranged from the large forces to the small forces

| | | Angular Velocity ( ω ) | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | PL | PM | PS | ZO | NS | NM | NL |
| Angular Position ( θ ) | PL | PL | PL | PL | PL | PM | PS | ZO |
| | PM | PL | PL | PL | PM | PS | ZO | NS |
| | PS | PL | PL | PM | PS | ZO | NS | NM |
| | ZO | PL | PM | PS | ZO | NS | NM | NL |
| | NS | PM | PS | ZO | NS | NM | NL | NL |
| | NM | PS | ZO | NS | NM | NL | NL | NL |
| | NL | ZO | NS | NM | NL | NL | NL | NL |

Table 3.3 IF-THEN Rules Developed from the Fundamental Rules.

The next analysis for the fuzzy controller design is to arrange the distribution of the input and the output membership functions, which provide the cart with proper output forces relating to the angular position and velocity changes and lead the inverted pendulum system towards a stable condition. The method of the justification to the membership function is based on a trial-and-error method. We will design four different values of fuzzy controllers to check their stability performance in the next chapter.

### 3.4 Interface Programming for the Fuzzy Controller

FIDE is a fuzzy control software, which provides us with environment to edit membership functions and IF-THEN rules and provides the microcontroller with an interface code in an assembly language. We use the assembly language, because it can provide our 8-bit microcontroller with faster fuzzy inference processes.

The FIDE are very easy to use. The basic steps for using the FIDE are four::

(1) Chose a fuzzy inference process method.

(2) List the input and output variables and the membership functions.

(3) Type the IF-THEN rules.

(4) Generate the interface codes in assembly language .

An example of the FIDE inference code is enclosed in Appendix B and an assembly interface code generated by FIDE is listed in Appendix C.

## Charter 4

## The Experiment Results

The structure of a fuzzy controller consists of two main categories, IF-THEN rules and membership functions. The IF-THEN rules are modeled from the step and the phase response as explained in previous chapter. The values of the membership functions are determined by way of trial-and-error method.

During the trial-and-error process, four different values of fuzzy controllers are specified with the same IF-THEN rules to simulate four different lengths of aluminum-round-bars (Table 4.1). The purpose of testing four different lengths of the aluminum bars is to test the robustness of the fuzzy control system. The initial condition for the pendulum is at the angular position $\theta = -5.5\,°$ and the angular velocity $\omega = 0$ rad/sec.

| Diameter x Length (mm) | Mass (kg) | Mass Moments of Inertia ($kg\text{-}m^2$) |
|---|---|---|
| 12.7 x 228.6 | 0.08 | $3.49 \times 10^{-4}$ |
| 12.7 x 330.2 | 0.11 | $1.00 \times 10^{-3}$ |
| 12.7 x 431.8 | 0.15 | $2.33 \times 10^{-3}$ |
| 12.7 x 533.4 | 0.18 | $4.27 \times 10^{-3}$ |

Table 4.1 Specification for Four Different Lengths of Pendulums in the Experiment

## 4.1 Simulations for the First Fuzzy Controller

The first fuzzy controller has the smallest ratio of the force-outputs to the angular-position-inputs among the four fuzzy controllers in the experiment. Simulations show that the first controller provides stable control performance to pendulum 1 (Figure 4.2) and the controller gradually become increasingly unstable when applied to pendulum 2 to pendulum 3, and to pendulum 4 (Figure 4.3, 4.4, and 4.5).

The angles of the pendulum 1 are stabilized in the range under ± 0.5°. Although pendulums 2 and 3 seem stable, the simulations indicate that the angles of pendulums 2 and 3 are gradually falling with a rate of 0.10 degree/sec. This unstable condition become more obvious in the simulation of pendulum 4 (Figure 4.5), where pendulum 4, the longest pendulum, falls with a rate of 0.5 degree/sec from the time of 0.25 seconds to 2.5 seconds and falls very rapidly thereafter.

In order to demonstrate the robustness of the fuzzy controller, the first controller will be modified. The approach is to increase ratio of the force-outputs to the angular-position-inputs in the membership functions, so that the fuzzy controller is capable of stabilizing all four different lengths of pendulums in the experiment.

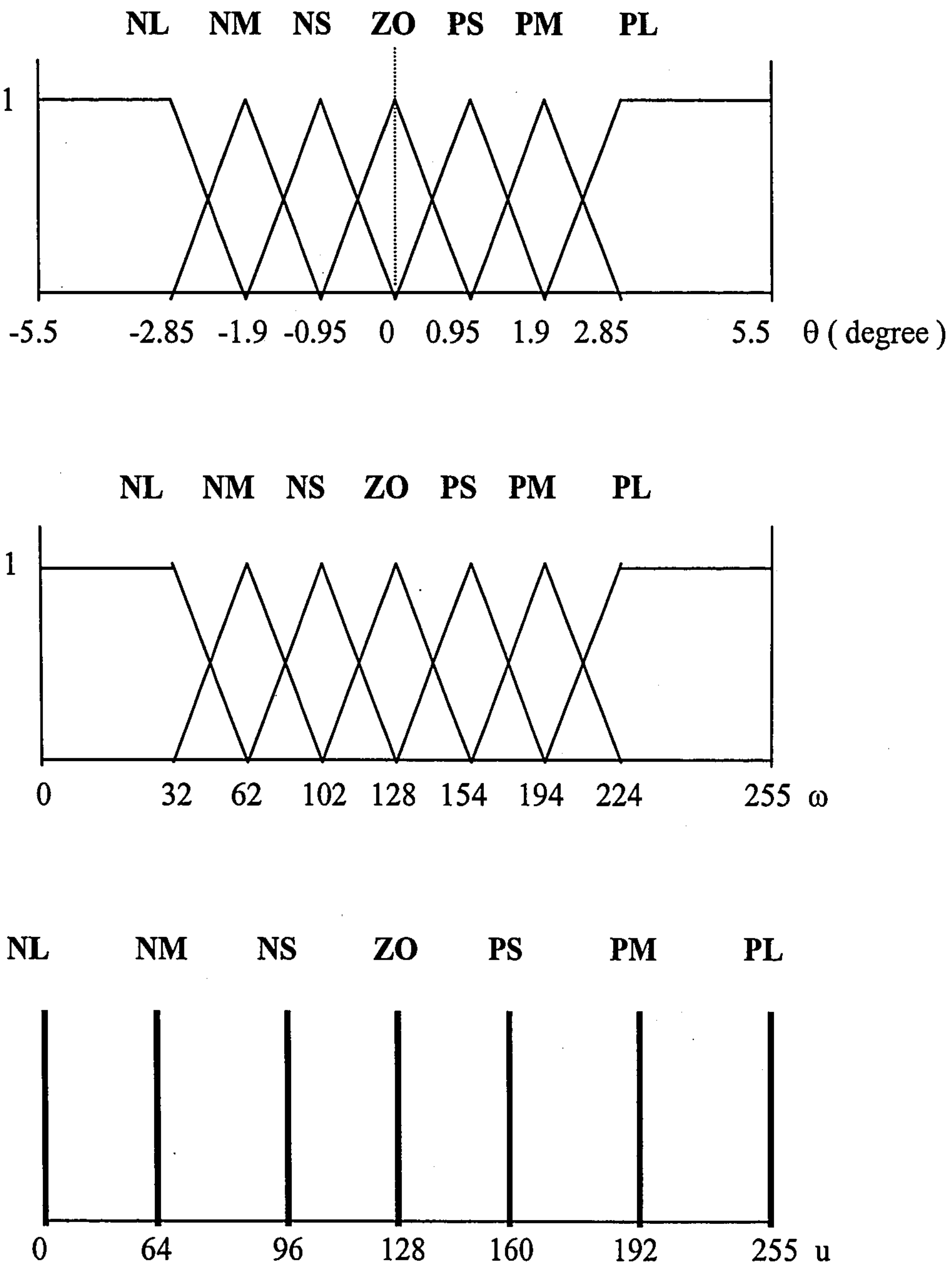



Figure 4.1 Membership Functions for the First Set of Fuzzy Controller

**Fig 4.2 Simulation for Controller 1 to Pendulum 1**

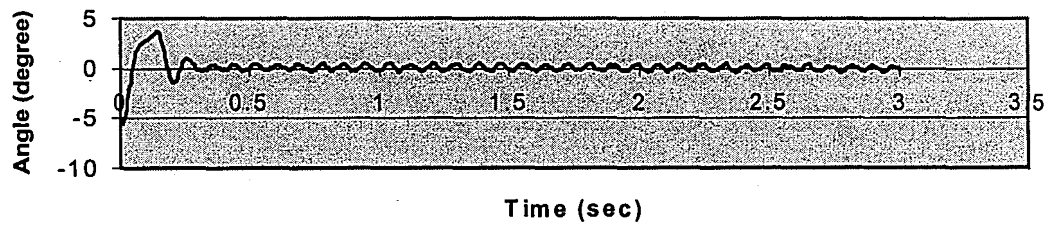



**Fig 4.3 Simulation for Controller 1 to Pendulum 2**

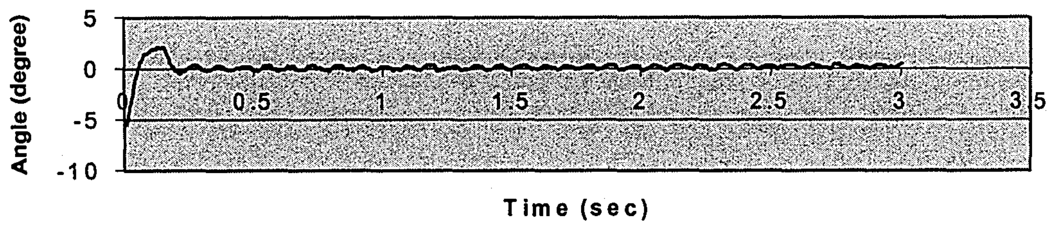



**Fig 4.4Simulation for Controller 1 to Pendulum 3**

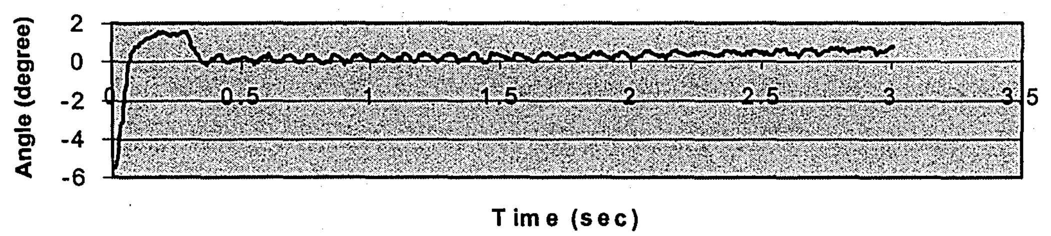
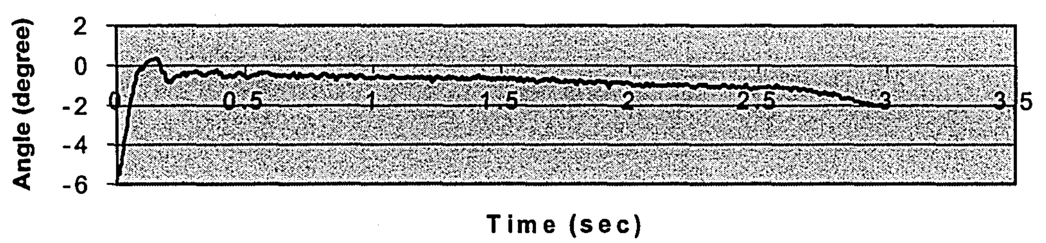



**Fig 4.5 Simulation for Controller 1 to Pendulum 4**



## 4.2 Simulations for the Second Fuzzy Controller

The second fuzzy logic controller (Figure 4.6), controller 2, is modified from controller 1, which increases the ratio of the output-forces to the angular positions in the membership functions of controller 1. Therefore, the controller is capable of providing a greater output force with the angular changes than that of controller 1.

Simulations (Figure 4.7, 4.8, 4.9 and 4.10) show that controller 2 successfully stabilizes all the different lengths of pendulums. The simulations indicate that the longer pendulums are more stable in control performance than the shorter pendulums. In the simulation, the shortest pendulum stabilizes between ± 0.4 degrees from the set point, while the longest pendulum stabilizes between ± 0.15 degrees from the set point.

Also, the relationship between the lengths of the pendulums and overshoots is that the rate of overshoot gradually decreases as the lengths of the pendulums increase; the shortest pendulum has an overshoot of 4.0 degrees and the longest pendulum has an overshoot of only 0.8 degrees.

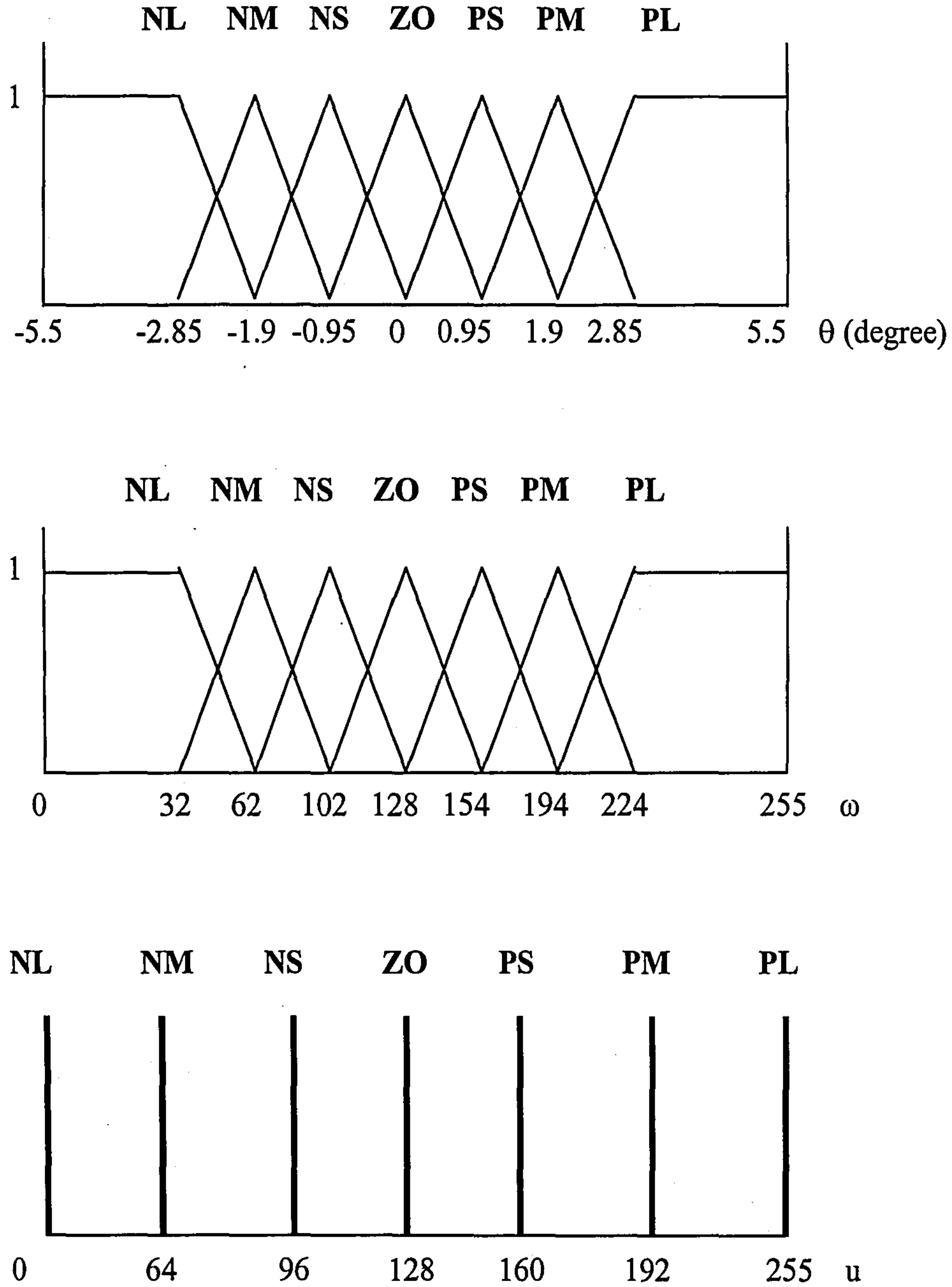



Figure 4.6 Membership Functions for the Second Set of Fuzzy Controller

Fig 4.7 Simulation for Controller 2 to Pendulum 1

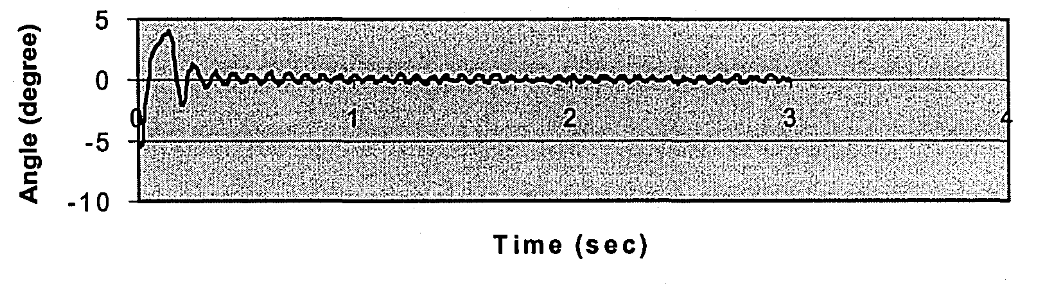



Fig 4.8 Simulation for Controller 2 to Pendulum 2

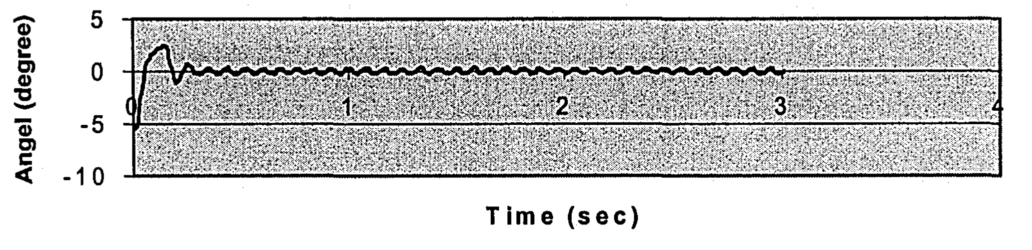



Fig 4.9 Simulation for Controller 2 to Pendulum 3

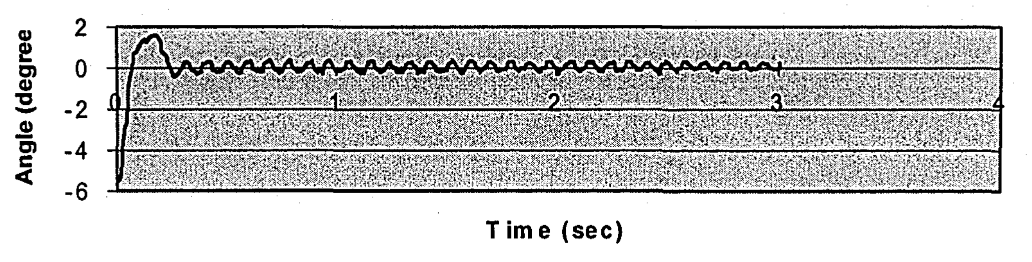
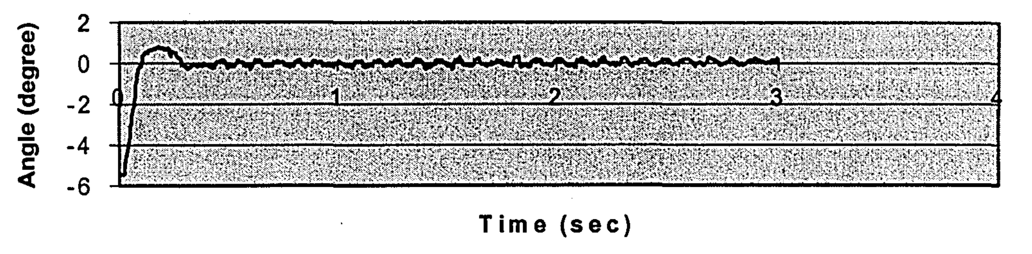



Fig 4.10 Simulation for Controller 2 to Pendulum 4



## 4.3 Simulations for the Third and Fourth Set of Fuzzy Controllers

Control performance was improved from controller 1 to controller 2. Simulations demonstrated that controller 2 successfully stabilize all different lengths of the pendulums in the experiment. If the ratio of the force-outputs to the angular changes in the membership functions are continue increased, what would be the results?

Controller 3 (Figure 4.11) is designed by increasing the ratio of force-outputs to the angle-inputs in the membership functions over that controller 2. The simulations using controller 3 to each of pendulums are shown in Figures 4.12, 4.13, 4.14, and 4.15. The control performance for the shortest pendulum is unstable (Figure 4.12). The control performances are gradually improved as the lengths of the pendulum increase.

Controller 4 (Figure 4.16) is designed by increasing the ratio of force-outputs to the angle-inputs over that of controller 3. The simulations show that the controller 4 provides pendulum 3 and 4 with a stable control performance (Figure 4.19 and 4.20) and provides pendulum 1 and 2 with an unstable condition (Figure 4.17 and 4.18).

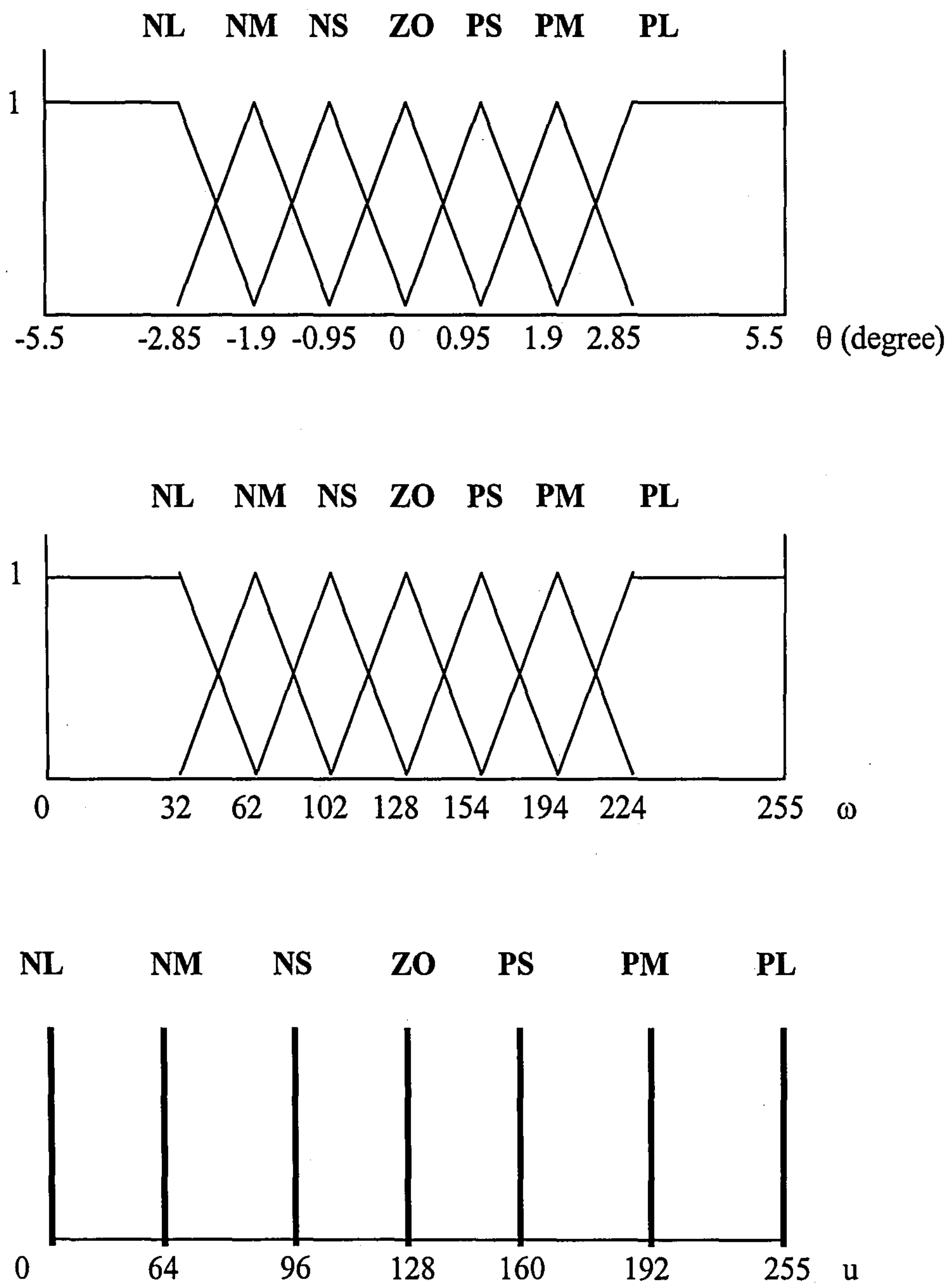



Figure 4.11 Membership Functions for the Third Set of Fuzzy Controller

Fig 4.12 Simulation for Controller 3 to Pendulum 1

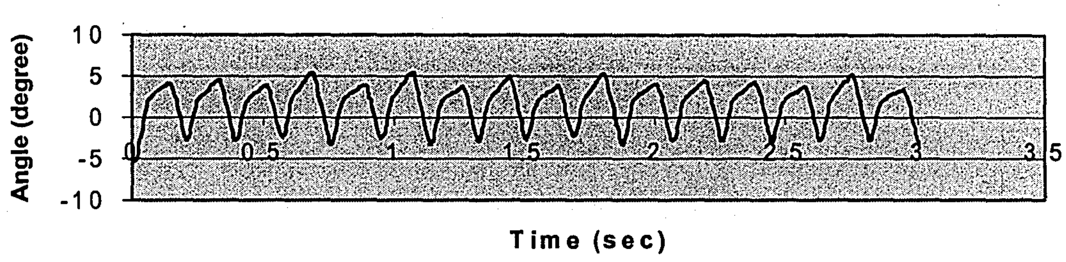



Fig 4.13 Simulation for Controller 3 to Pendulum 2

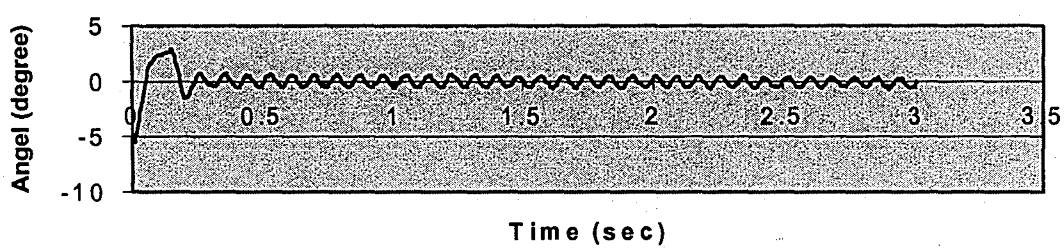



Fig 4.15 Simulation for Controller 3 to Pendulum 4

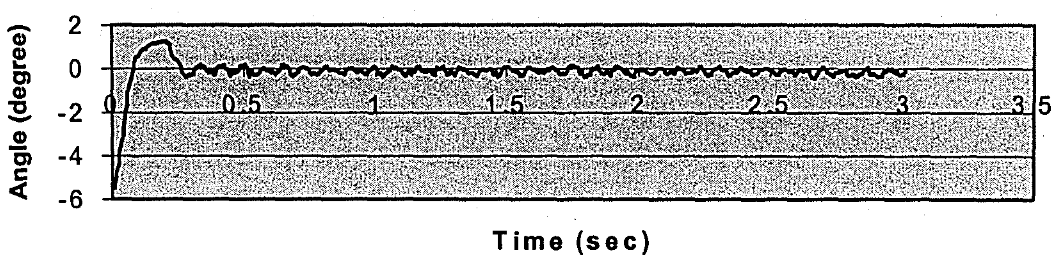
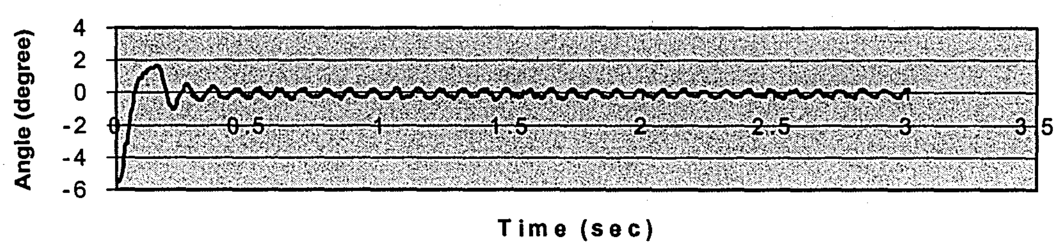



Fig 4.14 Simulation for Controller 3 to Pendulum 3



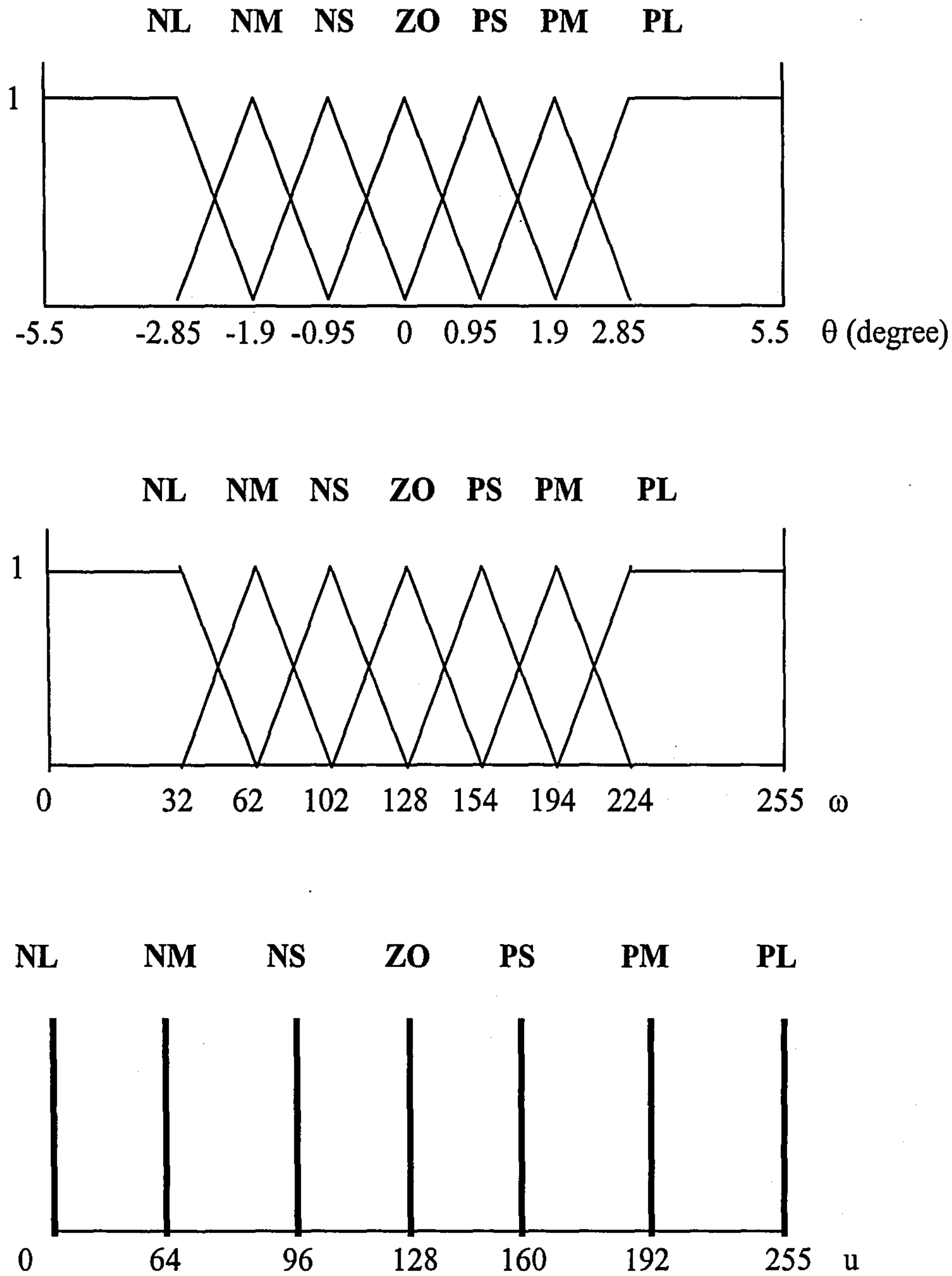



Figure 4.16 Membership Functions for the Fourth Set of Fuzzy Controller

# Chapter 5

## Conclusion

The original purpose of fuzzy logic as invented by Zadeh, was to introduce an approach to modeling humanistic systems, such as linguistics and biological sciences; systems that are difficult to apply conventional methods for modeling mathematically. However, because of the special properties of the fuzzy set and rule-bases, the theory of fuzzy logic control has become widely used as powerful tool for control applications.

The experimental control of an inverted pendulum done here utilizes the basic concepts of a phase plot and a step response to model the IF-THEN rules and applies the trial-and-error method to determine the values for the membership functions. The simulation results prove that the fuzzy controller provides four different lengths of the inverted pendulums with robust control performance, which is very difficult to accomplish using traditional control methods.

The experimental results also show that fuzzy logic control is very tolerant to imprecision. This is very important to many real-world applications. For example, in airplane elevation control, if the system is capable of tolerance of a wide variety of disturbances, such as wind, rain, and the mass transfer inside the airplane, the airplane will then be stably flown at the desired elevation.

Based on its many advantages of fuzzy control, fuzzy control will be increasingly used for many control applications. However, the main disadvantage of fuzzy control is that design of a fuzzy controller lacks a systematical approach in modeling the membership functions and the IF-THEN rules.

For the absence model within fuzzy control, a lot of time is needed to determine through trial-and-error the proper controller, especially for complex systems. For example, if an additional sensor is added to measure the position and velocity of the cart in the inverted pendulum experiment, the relative complexity of the new system will be at least 49 times than that of the original system. The new system will require four input variables and one output variable to construct 2401 ( $7^4$ ) of IF-THEN rules comparing to the 49 IF-THEN rules ( $7^2$ ) of the original system. Therefore, a model-based fuzzy control approach is essential to the future development of effective and efficient fuzzy control.

## References

[1] Jan Jantzen (1998) Internet course in fuzzy control, Denmark: Technical University of Denmark.

[2] Ching-Fang Lin (1994) Advanced Control System Design Englewoood Cliffs, NJ: Prentice Hall Inc.

[3] T. Yamakawa (1989) Stabilization of an Inverted Pendulum by a High-speed Fuzzy Logic Controller Hardware Systems, Fuzzy Set and Systems 32 (1989) 161-180

[4] Weijing Zhang (1992) Two-Stage Inverted Pendulum, Santa Clara, CA: Aptronix Inc Fuzzy Net Application Note.

[5] Guang-Chyan Hwang, Shih-Chang Lin (1992) A stability approach to fuzzy control design for nonlinear system, Fuzzy Sets and Systems 48.

[6] Le-Xin Wang (1997) A Course In Fuzzy Systems and Control, Cliffs, NJ: Prentice-Hall, Inc.

[7] E. H. Mamdani (1974) Applications of fuzzy algorithms for a simple dynamic plants, IEE 121, P.1585-1558

[8] Burr-Brown Inc. (1993) Application note-Precision Instrumentation amplifier, Tucson, AZ.

[9] National Semiconductor application note (1995) DAC 0800 8-bit DAC, Arlington, TX

[10] National Semiconductor application note (1995) LM12 80W operational amplifiers, Arlington,TX.

[11] Chuen Chien Lee (1190) Fuzzy logic in control system: Fuzzy logic controller –Part I IEEE Transaction on Systems, Man, and Cybernetics, Vol 2, No. 2, March/April.

[12] Chuen Chien Lee (1990) Fuzzy logic in control system: Fuzzy logic controller –Part II IEEE Transaction on Systems, Man, and Cybernetics, Vol 20, No. 2, March/April.

[13] Aptronix Inc (1994) FIDE User's Manual, Santa Clara, CA.

[14] Charles Fraser, John Milne (1994) Electro-Mechanical Engineering – An Integrated Approach NY,NY: IEEE Press.

[15] Ronald R. Yager and Dimitar P. Filev (1994) Essentials of Fuzzy Modeling and Control NY, NY : John Wiley & Sons, Inc .

[16] Rainer Palm, Dimiter Driankov, and Hans Hellendoorn (1997) Model Based Fuzzy Control NY, NY: Springer-Verlag Berlin.

[17] Moohammed Jamshidi, Series Editor (1997) Applications of Fuzzy Logic –Towards High Machine Intelligence Quotient Systems, NJ: Prentice-Hall, Inc.

[18] Kevin M. Passino, Stephen Yurkovich (1998) Fuzzy Control, Menlo Park, CA: Addison Wesley Longman, Inc.

[19] Katsuhiko Ogata (1990) Modern Control Engineering , Cliffs, NJ: Prentice-Hall, Inc.

[20] Texas Instruments application note (1992) General purpose Operational amplifiers. Dallas, TX.

[21] Motorola literature Distribution (1991) Motorola M68HC11 Reference Manual P.O. Box 20912, Phoenix, AZ.

**Appendix A: The FIDE Fuzzy Inference Code**

```
$ FileName: pend11.fil
$ Date Created: 1998
$ Date Last Modified: 1- 18, 1998
$ this symbol is the comment symbol
$ the part from symbol $ to the end of line is ignored
$ next line is empty and is ignored


fiu tvfi (min max) 8;
$ Unit type: fiu, feu, or fou
$ Inference method: mamdani or tvfi
$ Operators: any pair of  (min max prod sum binter bunion)
$ Resolution of grade: 4, 5, ..., 32 (fixed); *4, *5, ...,
*32 (floating);


invar theta "degree"    [
$ theta is an input variable, This variable use default
range: 0()255
      NL (@0,255, @50,255,  @76,0),
      NM (@50,0,  @76, 255, @102,0),
      NS (@76,0,  @102,255, @128,0),
      ZE (@102,0, @128,255, @154,0),
      PS (@128,0, @154,255, @180,0),
      PM (@154,0, @180,255, @206,0),
      PL (@180,0, @206,255, @255,255)
    ];                                  $ end of labels


invar omega " "  [
$ theta is an input variable, This variable use default
range: 0()255
      NL (@0,255, @32,255,  @52,0),
      NM (@32,0,  @52, 255, @78,0),
      NS (@52,0,  @78,255,  @128,0),
      ZE (@78,0,  @128,255, @178,0),
      PS (@128,0, @178,255, @204,0),
      PM (@178,0, @204,255, @224,0),
      PL (@204,0, @224,255, @255,255)
    ];                                  $ end of labels
```

```
outvar voltage "volt" centroid ( NL = 0,    $ voltage is an
output variable
$ defuzzy method: centroid,  max, right max, left max.
          NM = 90,
          NS  = 106,
          ZE  = 128,
          PS  = 150,
          PM = 166,
          PL  = 255
        );                                $ end of output
labels


$ rules start here:

if theta is NL and omega is NL then voltage is NL;
if theta is NL and omega is NM then voltage is NL;
if theta is NL and omega is NS then voltage is NL;
if theta is NL and omega is ZE then voltage is NL;
if theta is NL and omega is PS then voltage is NM;
if theta is NL and omega is PM then voltage is NS;
if theta is NL and omega is PL then voltage is ZE;
if theta is NM and omega is NL then voltage is NL;
if theta is NM and omega is NM then voltage is NL;
if theta is NM and omega is NS then voltage is NL;
if theta is NM and omega is ZE then voltage is NM;
if theta is NM and omega is PS then voltage is NS;
if theta is NM and omega is PM then voltage is ZE;
if theta is NM and omega is PL then voltage is PS;
if theta is NS and omega is NL then voltage is NL;
if theta is NS and omega is NM then voltage is NL;
if theta is NS and omega is NS then voltage is NL;
if theta is NS and omega is ZE then voltage is NM;
if theta is NS and omega is PS then voltage is ZE;
if theta is NS and omega is PM then voltage is PS;
if theta is NS and omega is PL then voltage is PM;
if theta is ZE and omega is NL then voltage is NL;
if theta is ZE and omega is NM then voltage is NM;
if theta is ZE and omega is NS then voltage is NS;
if theta is ZE and omega is ZE then voltage is ZE;
```

```
if theta is ZE and omega is PS then voltage is PS;
if theta is ZE and omega is PM then voltage is PM;
if theta is ZE and omega is PL then voltage is PL;
if theta is PS and omega is NL then voltage is NM;
if theta is PS and omega is NM then voltage is NS;
if theta is PS and omega is NS then voltage is ZE;
if theta is PS and omega is ZE then voltage is PM;
if theta is PS and omega is PS then voltage is PL;
if theta is PS and omega is PM then voltage is PL;
if theta is PS and omega is PL then voltage is PL;
if theta is PM and omega is NL then voltage is NS;
if theta is PM and omega is NM then voltage is ZE;
if theta is PM and omega is NS then voltage is PS;
if theta is PM and omega is ZE then voltage is PM;
if theta is PM and omega is PS then voltage is PL;
if theta is PM and omega is PM then voltage is PL;
if theta is PM and omega is PL then voltage is PL;
if theta is PL and omega is NL then voltage is ZE;
if theta is PL and omega is NM then voltage is PS;
if theta is PL and omega is NS then voltage is PM;
if theta is PL and omega is ZE then voltage is PL;
if theta is PL and omega is PS then voltage is PL;
if theta is PL and omega is PM then voltage is PL;
if theta is PL and omega is PL then voltage is PL;
end
```

## Appendix B: The MC68HC11 Assembly code

```
**********************************************************
*
*
*     Fide - (C) Copyright 1992-1994, by Aptronix, Inc.
*     Real Time Code Generator Version 2.0, 1994
*     MC68HC11 Assembly Code for IASM11
*
*
*     File created Fri Sep 04 20:51:21 1998
*     by Fide Real-Time Code Generator
*
*     Original(Source) File Name: pend6h.fil
*
*                               Inputs:  2
*                              Outputs:  1
*                 Membership Functions:  21
*                      Rule statments:  49
*         Total Machine Code (Bytes):  1027
*         RAM Variables Table (Bytes):  41
*              FIU Data Table (Bytes):  497
*                 FIU Program (Bytes):  489
*    Execution Time (Machine Cycles):  13944
*
**********************************************************


***************** RAM Variables Table ******************

       ORG $00
INPUTS    RMB     $02
OUTPUTS   RMB     $01
TVLST     RMB     $0E
GT_BUF    RMB     1
IN_BUF    RMB     1
MF1_REG   RMB     1
MF2_REG   RMB     1
AND_REG   RMB     1
OR_REG    RMB     1
TUH_REG   RMB     1
TUL_REG   RMB     1
SUH_REG   RMB     1
SUL_REG   RMB     1
POINT     RMB     2
```

```
FIUID     RMB     2
AND_SUB   RMB     2
OR_SUB    RMB     2
CD_BUF    RMB     2
IV_BUF    RMB     2
OV_BUF    RMB     2



       ORG $D000

FIULIST       FDB FIU_pend6h

******************* FIU Data Table **********************

FIU_pend6h
          FDB CODE_pend6h
          FCB $FF
          FDB MIN1_OP
          FDB MAX2_OP
          FDB INRANG_pend6h
          FDB OUTRANG_pend6h
          FDB MF_pend6h_1
          FDB MF_pend6h_2
          FDB MF_pend6h_3
          FDB MF_pend6h_4
          FDB MF_pend6h_5
          FDB MF_pend6h_6
          FDB MF_pend6h_7
          FDB MF_pend6h_8
          FDB MF_pend6h_9
          FDB MF_pend6h_10
          FDB MF_pend6h_11
          FDB MF_pend6h_12
          FDB MF_pend6h_13
          FDB MF_pend6h_14
          FDB MF_pend6h_15
          FDB MF_pend6h_16
          FDB MF_pend6h_17
          FDB MF_pend6h_18
          FDB MF_pend6h_19
          FDB MF_pend6h_20
          FDB MF_pend6h_21
INRANG_pend6h
          FCB $00
```

```
            FCB  $00
            FCB  $FF
            FCB  $00
            FCB  $00
            FCB  $FF
OUTRANG_pend6h
            FCB  $00
MF_pend6h_1
            FCB  $02
            FCB  $07
            FCB  $00
            FCB  $00
            FCB  $FF
            FCB  $4A
            FCB  $E3
            FCB  $FF
            FCB  $5C
            FCB  $00
            FCB  $00
            FCB  $FF
MF_pend6h_2
            FCB  $04
            FCB  $0F
            FCB  $00
            FCB  $00
            FCB  $00
            FCB  $4A
            FCB  $E3
            FCB  $00
            FCB  $5C
            FCB  $E3
            FCB  $FF
            FCB  $6E
            FCB  $00
            FCB  $00
            FCB  $FF
MF_pend6h_3
            FCB  $04
            FCB  $0F
            FCB  $00
            FCB  $00
            FCB  $00
            FCB  $5C
            FCB  $E3
            FCB  $00
            FCB  $6E
```

```
            FCB  $E3
            FCB  $FF
            FCB  $80
            FCB  $00
            FCB  $00
            FCB  $FF
MF_pend6h_4
            FCB  $04
            FCB  $0F
            FCB  $00
            FCB  $00
            FCB  $00
            FCB  $6E
            FCB  $E3
            FCB  $00
            FCB  $80
            FCB  $E3
            FCB  $FF
            FCB  $92
            FCB  $00
            FCB  $00
            FCB  $FF
MF_pend6h_5
            FCB  $04
            FCB  $0F
            FCB  $00
            FCB  $00
            FCB  $00
            FCB  $80
            FCB  $E3
            FCB  $00
            FCB  $92
            FCB  $E3
            FCB  $FF
            FCB  $A4
            FCB  $00
            FCB  $00
            FCB  $FF
MF_pend6h_6
            FCB  $04
            FCB  $0F
            FCB  $00
            FCB  $00
            FCB  $00
            FCB  $92
            FCB  $E3
```

```
            FCB $00
            FCB $A4
            FCB $E3
            FCB $FF
            FCB $B6
            FCB $00
            FCB $00
            FCB $FF
MF_pend6h_7
            FCB $00
            FCB $07
            FCB $00
            FCB $00
            FCB $00
            FCB $A4
            FCB $E3
            FCB $00
            FCB $B6
            FCB $00
            FCB $FF
            FCB $FF
MF_pend6h_8
            FCB $02
            FCB $07
            FCB $00
            FCB $00
            FCB $FF
            FCB $20
            FCB $CC
            FCB $FF
            FCB $34
            FCB $00
            FCB $00
            FCB $FF
MF_pend6h_9
            FCB $04
            FCB $0F
            FCB $00
            FCB $00
            FCB $00
            FCB $20
            FCB $CC
            FCB $00
            FCB $34
            FCB $9D
            FCB $FF
```

```
          FCB $4E
          FCB $00
          FCB $00
          FCB $FF
MF_pend6h_10
          FCB $04
          FCB $0F
          FCB $00
          FCB $00
          FCB $00
          FCB $34
          FCB $9D
          FCB $00
          FCB $4E
          FCB $52
          FCB $FF
          FCB $80
          FCB $00
          FCB $00
          FCB $FF
MF_pend6h_11
          FCB $04
          FCB $0F
          FCB $00
          FCB $00
          FCB $00
          FCB $4E
          FCB $52
          FCB $00
          FCB $80
          FCB $52
          FCB $FF
          FCB $B2
          FCB $00
          FCB $00
          FCB $FF
MF_pend6h_12
          FCB $04
          FCB $0F
          FCB $00
          FCB $00
          FCB $00
          FCB $80
          FCB $52
          FCB $00
          FCB $B2
```

```
            FCB  $9D
            FCB  $FF
            FCB  $CC
            FCB  $00
            FCB  $00
            FCB  $FF
MF_pend6h_13
            FCB  $04
            FCB  $0F
            FCB  $00
            FCB  $00
            FCB  $00
            FCB  $B2
            FCB  $9D
            FCB  $00
            FCB  $CC
            FCB  $CC
            FCB  $FF
            FCB  $E0
            FCB  $00
            FCB  $00
            FCB  $FF
MF_pend6h_14
            FCB  $00
            FCB  $07
            FCB  $00
            FCB  $00
            FCB  $00
            FCB  $CC
            FCB  $CC
            FCB  $00
            FCB  $E0
            FCB  $00
            FCB  $FF
            FCB  $FF
MF_pend6h_15
            FCB  $00
MF_pend6h_16
            FCB  $5A
MF_pend6h_17
            FCB  $6A
MF_pend6h_18
            FCB  $80
MF_pend6h_19
            FCB  $96
MF_pend6h_20
```

```
            FCB $A6
MF_pend6h_21
            FCB $FF
CODE_pend6h
            FCB $02
            FCB $07
            FCB $00
            FCB $01
            FCB $02
            FCB $03
            FCB $04
            FCB $05
            FCB $06
            FCB $07
            FCB $07
            FCB $08
            FCB $09
            FCB $0A
            FCB $0B
            FCB $0C
            FCB $0D
            FCB $0D
            FDB $0003
            FDB $6007
            FDB $0002
            FDB $7009
            FDB $0002
            FDB $7008
            FDB $0002
            FDB $7007
            FDB $0001
            FDB $7009
            FDB $0001
            FDB $7008
            FDB $0001
            FDB $7007
            FDB $0000
            FDB $700A
            FDB $0000
            FDB $7009
            FDB $0000
            FDB $7008
            FDB $0000
            FDB $7807
            FDB $000E
            FDB $0004
```

```
FDB $6007
FDB $0003
FDB $7008
FDB $0002
FDB $700A
FDB $0001
FDB $700A
FDB $0000
FDB $780B
FDB $000F
FDB $0005
FDB $6007
FDB $0004
FDB $7008
FDB $0003
FDB $7009
FDB $0001
FDB $700B
FDB $0000
FDB $780C
FDB $0010
FDB $0006
FDB $6007
FDB $0005
FDB $7008
FDB $0004
FDB $7009
FDB $0003
FDB $700A
FDB $0002
FDB $700B
FDB $0001
FDB $700C
FDB $0000
FDB $780D
FDB $0011
FDB $0006
FDB $6008
FDB $0005
FDB $7009
FDB $0003
FDB $700B
FDB $0002
FDB $700C
FDB $0001
FDB $780D
```

```
            FDB  $0012
            FDB  $0006
            FDB  $6009
            FDB  $0005
            FDB  $700A
            FDB  $0004
            FDB  $700A
            FDB  $0003
            FDB  $700C
            FDB  $0002
            FDB  $780D
            FDB  $0013
            FDB  $0006
            FDB  $600D
            FDB  $0006
            FDB  $700C
            FDB  $0006
            FDB  $700B
            FDB  $0006
            FDB  $700A
            FDB  $0005
            FDB  $700D
            FDB  $0005
            FDB  $700C
            FDB  $0005
            FDB  $700B
            FDB  $0004
            FDB  $700D
            FDB  $0004
            FDB  $700C
            FDB  $0004
            FDB  $700B
            FDB  $0003
            FDB  $780D
            FDB  $0014
            FDB  $D000
            ORG   $D500
            LDAA $80
            STAA $1031
            LDAA #$FF
            STAA $1007
            STAA $1039
LOOP        JSR   SAMPLE
            LDAA $1033
            STAA INPUTS+$00
            LDAB $1034
```

```
            STAB INPUTS+$01
            JSR  FIU
            LDAA OUTPUTS+$00
            STAA $1004
            JMP  LOOP
SAMPLE      LDAA #$30
            STAA $1030
            LDAB #$43
LOOP_1      DECB
            BNE  LOOP_1
            RTS

********************** FIU Program *******************

FIU
       LDD FIULIST
       STD FIUID

       LDX FIUID
       LDD $0,X
       STD POINT
       LDAA $2,X
       STAA GT_BUF
       LDD $3,X
       STD AND_SUB
       LDD $5,X
       STD OR_SUB
       LDD $7,X
       STD IV_BUF
       JSR INIT
       CLRA
       CLRB
       STD CD_BUF
       STD OV_BUF
       LDX POINT
       LDAB $0,X
       STAB TUH_REG
IN_LOOP LDX POINT
       INX
       STX POINT
       LDAB $0,X
       STAB TUL_REG
       LDX CD_BUF
       LDAA INPUTS,X
       INX
       STX CD_BUF
```

```
        LDX IV_BUF
        ASL $0,X
        BCC IN_L0
        ADDA #$80
IN_L0   CMPA $1,X
        BCC IN_L
        CLR IN_BUF
        JMP IN_L4
IN_L    CMPA $2,X
        BCS IN_L3
        LDAA $2,X
IN_L3   SUBA $1,X
        STAA IN_BUF
IN_L4   XGDX
        ADDD #$3
        STD IV_BUF
LB_LOOP LDX POINT
        INX
        STX POINT
        LDAB $0,X
        CMPB #$FF
        BEQ PASS_L
        CLRA
        ASLD
        ADDD FIUID
        XGDX
        LDX $B,X
        JSR MF_SUB
        JMP OUT_L
PASS_L  LDAA IN_BUF
OUT_L   LDX OV_BUF
        STAA TVLST,X
        INX
        STX OV_BUF
        DEC TUL_REG
        BNE LB_LOOP
        DEC TUH_REG
        BNE IN_LOOP
RULEXE: JSR INIT
        LDX FIUID
        LDD $9,X
        STD OV_BUF
        CLR IV_BUF
        CLR IV_BUF+1
AND_LOOP LDX POINT
        INX
```

```
        INX
        STX POINT
        LDD $0,X
        STD CD_BUF
        ASL CD_BUF
        BCS OUT_LOOP
        CLRA
        XGDY
        LDAA TVLST,Y
        ASL CD_BUF
        BCS AND_OP
        STAA AND_REG
        JMP OR_LOOP
AND_OP     LDX AND_SUB
        JSR $0,X
OR_LOOP    ASL CD_BUF
        BCC AND_LOOP
        ASL CD_BUF
        BCS OR_OP
        LDAA AND_REG
        STAA OR_REG
        JMP DFZ_LOOP
OR_OP      LDX OR_SUB
        JSR $0,X
DFZ_LOOP   ASL CD_BUF
        BCC AND_LOOP
        LDX POINT
        INX
        INX
        STX POINT
        LDD $0,X
        ASLD
        ADDD FIUID
        XGDX
        LDX $B,X
        LDAB $0,X
        ASL CD_BUF
        BCS MAX_DFZ
        JSR DFZ_C
        JMP AND_LOOP
MAX_DFZ    JSR DFZ_M
        JMP AND_LOOP
OUT_LOOP   ASL CD_BUF
        BCS C_DFZ
        LDAA OR_REG
        JMP END_LOOP
```

```
C_DFZ     ASL CD_BUF
          BCS L_MAX
          JSR DIV_SUB
          JMP END_LOOP
L_MAX     ASL CD_BUF
          BCS R_MAX
          LDAA SUL_REG
          JMP END_LOOP
R_MAX     ASL CD_BUF
          BCS M_MAX
          LDAA SUH_REG
          JMP END_LOOP
M_MAX     LDAA SUL_REG
          ADDA SUH_REG
          RORA
END_LOOP  LDX IV_BUF
          LDY OV_BUF
          ADDA $0,Y
          INY
          STY OV_BUF
          STAA OUTPUTS,X
          INX
          STX IV_BUF
          JSR INIT
          ASL CD_BUF
          BCS END_1
          JMP AND_LOOP
END_1     RTS
MIN1_OP   CMPA AND_REG
          BCC MIN1_E
          STAA AND_REG
MIN1_E    RTS
MAX2_OP   LDAA AND_REG
          CMPA OR_REG
          BCS MAX2_E
          STAA OR_REG
MAX2_E    RTS
DFZ_C     LDAA OR_REG
          MUL
          TAB
          CLRA
          ADDD TUH_REG
          STD TUH_REG
          CLRA
          LDAB OR_REG
          ADDD SUH_REG
```

```
        STD SUH_REG
        RTS
DFZ_M       LDAA TUL_REG
        CMPA OR_REG
        BCS DMAX_1
        BEQ DMAX_2
        RTS
DMAX_1      STAB SUH_REG
        STAB SUL_REG
        LDAA OR_REG
        STAA TUL_REG
        RTS
DMAX_2      CMPB SUL_REG
        BCS DMAX_3
        CMPB SUH_REG
        BCC DMAX_4
        RTS
DMAX_3      STAB SUL_REG
        RTS
DMAX_4      STAB SUH_REG
        RTS
DIV_SUB     LDD  TUH_REG
        LDX  SUH_REG
        FDIV
        BVS DIV_1
        XGDX
        RTS
DIV_1       LDAA #$FF
        RTS

INIT        CLR TUL_REG
        CLR TUH_REG
        CLR SUL_REG
        CLR SUH_REG
        RTS
MF_SUB      LDAA $0,X
        STAA MF1_REG
        INX
        LDAA $0,X
        STAA MF2_REG
        LSL MF1_REG
        LSL MF2_REG
        INX
MF_1        LDAA $3,X
        LSR MF1_REG
        LSR MF2_REG
```

```
        CMPA IN_BUF
        BCC MF_11
        LDAB #$3
        ABX
        JMP MF_1
MF_11       LDAA IN_BUF
        SUBA $0,X
        LDAB $1,X
        MUL
        LSR MF2_REG
        BCC MF_12
        LSRD
        LSRD
        LSRD
        LSRD
MF_12       TSTA
        BEQ MF_13
        LDAB #$FF
MF_13       LSR MF1_REG
        BCS MF_2
        LDAA 2,X
        ABA
        BCC MF_14
        LDAA #$FF
MF_14       RTS
MF_2        STAB MF1_REG
        LDAA 2,X
        SUBA MF1_REG
        BCC MF_21
        CLRA
MF_21       RTS
```

# Vita

Chen-Sheng Chou, son of Cha-Oh Chou and Chih-Huei Chen Chou, was bone in 1964 in Taipei, Taiwan, Republic of China. He received his B.S. in mechanical engineering in 1995 from University of Colorado at Denver. He had several work experiment in automotive and heavy machinery industrial. He interests in subjects of microprocessor-based control, mechatronics, and mechanism.

END
OF
TITLE