APPLICATION OF VIRTUAL INSTRUMENTATION IN POSITION CONTROL SYSTEM USING DIRECT DIGITAL CONTROL VIA PID AND FUZZY LOGIC CONTROLLER

MARIAM BINTI MD GHAZALY

UNVERSITI TEKNOLOGI MALAYSIA

UNIVERSITI TEKNOLOGI MALAYSIA

BORANG PENGESAHAN STATUS TESIS⁺

JUDUL : APPLICATION OF VIRTUAL INSTRUMENTATION IN POSITION CONTROL SYSTEM USING DIRECT DIGITAL CONTROL VIA PID AND FUZZY LOGIC CONTROLLER

SESI PENGAJIAN: 2004 / 2005

MARIAM BINTI MD GHAZALY

(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah)* ini disimpan di Perpustakaan Universiti Teknologi Malaysia dengan syarat-syarat kegunaan seperti berikut:

- 1. Tesis adalah hak milik Universiti Teknologi Malaysia.
- 2. Perpustakaan Universiti Teknologi Malaysia dibenarkan membuat salinan untuk tujuan pengajian sahaja.
- 3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
- **Sila tandakan (✓) 4.

Saya

SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

TIDAK TERHAD

Disahkan oleh

(TANDATANGAN PENULIS) (TANDATANGAN PENYELIA) 24, JALAN SUASANA 3/1A, ASSOC. PROF. DR. MOHD FUA'AD BANDAR TUN HUSSEIN ONN. **BIN HJ RAHMAT**

Nama Penyelia

Tarikh: 4th APRIL 2005

CATATAN:

Alamat Tetap:

SELANGOR.

43200 CHERAS,

Potong yang tidak berkenaan.

Tarikh : 4th APRIL 2005

Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis ini perlu dikelaskan sebagai SULIT atau TERHAD.

Tesis dimaksudkan sebagai tesis bagi Ijazah Doktor Falsafah dan Sarjana secara ٠ penyelidikan, atau disertai bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjana Muda (PSM).

"I hereby declare that I had read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of the degree of Master of Engineering (Electrical - Mechatronic & Automatic Control)"

Signature	:
Name of supervisor	: Assoc. Prof. Dr. Mohd Fua'ad Haji Rahmat.
Date	: 4 th April 2005.

APPLICATION OF VIRTUAL INSTRUMENTATION IN POSITION CONTROL SYSTEM USING DIRECT DIGITAL CONTROL VIA PID AND FUZZY LOGIC CONTROLLER

MARIAM BINTI MD GHAZALY

A project report submitted in partial fulfillment of the requirements for the award of the degree of Master of Engineering (Electrical - Mechatronic & Automatic Control)

> Faculty of Electrical Engineering Universiti Teknologi Malaysia

> > APRIL 2005

I declare that this thesis entitled "Application of Virtual Instrumentation in Position Control System Using Direct Digital Control via PID and Fuzzy Logic Controller "is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidate of any other degree.

Signature	:	
Name	:	Mariam binti Md Ghazaly.
Date	:	4 th April 2005.

To my beloved mother and father

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my main thesis supervisor, Assoc. Prof. Dr. Mohd Fua'ad Haji Rahmat for encouragement, guidance, critics and friendship. Without their continued support and interest, this thesis would not have been the same as presented here.

My fellow postgraduate students should also be recognized for their support. My sincere appreciation also extends to all my colleagues and others who have provided assistance at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to my father and mother for their guidance, advices and motivation.

ABSTRACT

The objective is to design and developed a GUI software using Microsoft Visual Basic 6.0 to ease students in performing position control system experiment, expose students on position control system theoretically and practically and to developed controller using software for position control system. The scope of the project is to apply direct digital control (DDC) in position control system. PID controller and a fuzzy logic controller will be use to control the output response. An interactive software will be developed to help the student to visualize and analyze the system. This project can be divided into two parts namely hardware and software. Hardware parts involve more in interfacing MS150 Modular servo System and Data Acquisition System with a personal computer. While the software part include programming real-time software using Microsoft Visual Basic 6.0. In the earlier stage, literature review and experiment are performed manually to understand the concept of the controller. An interactive software will be designed using Microsoft Visual Basic 6.0. The software will be equipped with a set of graphic instructions to ease any mistake when performing experiment. Finally, the software will be integrated with hardware to produce a GUI position control system.

ABSTRAK

Satu perisian komputer akan di reka dengan menggunakan Microsoft Visual Basic 6.0. Tujuan perisian ini di reka adalah untuk pembelajaran dan pemahaman pelajar semasa menjalankan ujikaji mengawal kedudukan sistem motor servo. Selain itu, tujuan perisian ini adalah untuk mereka sistem kawalan menggunakan Visual Basic 6.0 yang dapat mengawal kedudukan motor servo. Skop projek ini adalah untuk mengadaptasikan modul kawalan terus digital dalam mengawal kedudukan motor servo. Pengawal PID dan pengawal "fuzzy logic" akan digunakan bagi mengawal keluaran motor servo. Satu perisian yang interaktif akan di reka agar dapat membantu pelajar menganalisis sistem motor servo tersebut. Projek ini terbahagi kepada dua bahagian iaitu perkakasan dan perisian. Perkakasan terdiri daripada membuat hubungan antaramuka antara modal MS150 Modular servo sistem dengan DAQ cad dan computer peribadi. Manakala, perisian adalah lebih kepada mereka bentuk perisian yakni mereka bentuk pengawal PID dan pengawal "fuzzy logic" menggunakan Microsoft Visual Basic 6.0 bagi mengawal sistem motor servo.

TABLE OF CONTENTS

CHAPTER TITLE

1

PAGE

DECLARATION	ii
DEDICATION	iii
ACKNOWLEDGMENT	iv
ABSTRACT	v
ABSTRAK	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES	xii
LIST OF SYMBOLS	xiv
LIST OF APPENDICES	xviii

INTI	RODUCTION	1
1.1	Virtual instrumentation	1
1.2	Virtual Instruments	3
1.3	Project Objectives	5
1.4	Project Scope	7

2 LITERATURE REVIEW

2.1	Introduction.		8	
2.2	Previo	Previous Research.		
	2.2.1	"Motor Drive Control System For	10	
		Education and Research".		
	2.1.2	"Development of a Distant Laboratory	10	
		using Labview".		
	2.1.3	"Real-Time Position Control of	11	
		Free-Electron Laser Beams: Theory		
		and Experiments".		
	2.1.4	"Closed-loop Position Control	11	
		System using Labview".		
2.3	Virtua	Virtual Instrumentation in Position		
	Contro	Control System.		

3 RESEARCH METHODOLOGY 13

3.1	Projec	et Overview	13
3.2	Positio	on Control System	15
	3.2.1	General Servomotor System Model	20
	3.2.2	Direct Digital Control (DDC)	24
	3.2.3	PID controller	25
		3.2.3.1 Analog PID controller	27
		3.2.3.2 Digital PID controller	30
	3.2.4	Fuzzy Logic Controller	33
		3.2.4.1 Fuzzification	36
		3.2.4.2 Knowledge Base	38
		3.2.4.3 Inferencing	40
		3.2.4.4 Defuzzification	41

8

3.3	Softw	are Review	42
	3.3.1	Virtual Instrumentation - Graphical User	42
		Interface (GUI)	
		3.3.1.1 General GUI Design Considerations.	43
		3.3.1.2 Visual Presentation	45
		3.3.1.3 VI Diagram Considerations	46
		3.3.1.4 Additional GUI Object Considerations	46
	3.3.2	Microsoft Visual Basic	47
3.4	Hardv	vare Review	48
	3.4.1	Data Acquisition System (DAS)	48
		3.4.1.1 DAQ Hardware	50
		3.4.1.2 Analog Input system	53
		3.4.1.3 Analog Output System	54
		3.4.1.4 D/A Converter	55
	3.4.2	MS150 Modular Servo System	55
		3.4.2.1 Pre-amplifier Unit – PA150C	56
		3.4.2.2 Operational Amplifier Unit -OA150A	56
		3.4.2.3 Input Potentiometer – IP150H &	57
		Output Potentiometer -OP150K	
		3.4.2.4 Attenuator Unit – AU150B	58
		3.4.2.5 DC Motor – DCM150F	58
		3.4.2.6 Reduction Gear Tacho Unit – GT150X	59
		3.4.2.6 Servo Amplifier Unit – SA150D	59
		3.4.2.7 Power Supply Unit – PS150E	60
	3.4.3	Hardware Connection	61
	3.4.4	Classical Control System Experiment	62

4 **RESULTS & ANALYSIS**

4.1	Direct	Digital Control	64
	4.1.1	PID controller GUI	65
	4.1.2	Fuzzy Logic Controller GUI	67
4.2	Prelim	ninary Result Analysis	69
	4.2.1	Closed Loop Position Control System	70
	4.2.2	Effect of Gain Changes on Deadband	73
	4.2.3	Unit Step Response to Gain Changes	74
	4.2.4	Velocity Feedback	77
4.3	Result	Statement	82
	4.3.1	PID Controller Result Analysis	83
	4.3.2	Fuzzy Logic Controller Result Analysis	85
4.4	Comp	arison between PID and Fuzzy Logic	86
4.5	Discus	ssion	87

5 CONCLUSION

51	Suggestion for future work	91
J.1	Suggestion for future work	71

REFERENCES	92
Appendices A-D	94

63

89

LIST OF TABLES

NO.	TITLE	PAGE
3.1	Effect of each PID Controller	30
4.1	Output result of the experiment unit step response to	75
	gain changes.	
4.2	Output result of the experiment velocity feedback	78
4.3	Output result of the PID controller	84
4.4	The comparison output between response using the	87
	PID and Fuzzy logic.	

LIST OF FIGURES

NO.	TITLE	PAGE
1.1	Structure of a measuring device.	2
1.2	Hardware architecture of virtual instruments.	4
3.1	General form of the block diagram for closed loop position control system.	15
3.2	Standard measurements on Step Response of a Control System	17
3.3	Single position loop diagram can be modified to include	17
	velocity feedback.	
3.4	Simple Error Channel	18
3.5	DC-Motor (a) Wiring Diagram (b) Sketch (c) Schematic	20
3.6	Direct Digital Control Schematic.	24
3.7	General Feedback Architecture	26
3.8	Block diagram for PID Controller	28
3.9	Flow chart for the PID controller system.	32
3.10	Flowchart of Simulator	34
3.11	FLC Design Methodology	35
3.12	The input and output variable.	36
3.13	The error input variable	37
3.14	The rate input variable	37
3.15	The motor output variable	38
3.16	The fuzzy logic rules	39

3.17	The firing strength for the rule base using max-min inference.	41
3.18	Computerized data acquisition and control system.	49
3.19	A picture of AX5412H Data Acquisition Card	50
3.20	AX750 screw terminal panel	52
3.21	Many input signals come into the A/D system	53
3.22	General View of MS150 Modular Servo System.	55
3.23	PA150C- Pre-amplifier Unit	56
3.24	OA150A – Operational Amplifier Unit	57
3.25	Input Potentiometer and Output Potentiometer (IP150H, OP150K)	57
3.26	AU150B – Attenuator Unit	58
3.27	DCM150F – DC Motor with Tachogenarator	59
3.28	SA150D – Servo Amplifier Unit	60
3.29	PS150E – Power Supply Unit	60
3.30	Connection between PC and MS150 through AX750 Screw	61
	Terminal Panel	
4.1	Splash screen	64
4.2	Circuit diagram for direct digital control.	65
4.3	PID controller GUI	66
4.4	Fuzzy Logic Controller GUI	68
4.5	Fuzzy Logic Controller data capture form	68
4.6	The output result of closed loop position control system experiment.	72
4.7	The output result of gain changes on deadband	74
4.8	PID results which was capture within the GUI	83
4.9	The fuzzy logic result which was capture within the GUI.	85

LIST OF SYMBOLS

VI	-	Virtual Instrument
PC	-	Personal Computer
DAQ	-	Data Acquisition
DSP	-	Digital Signal Processing
GUI	-	Graphical User Interface
DDC	-	Direct Digital Control
ITL	-	Interactive Teaching and Learning
PID	-	Proportional Integral Derivative Controller
I&C	-	Instrumentation and Control
DC	-	Direct Current
ADC	-	Analogue to Digital Converter
DAC	-	Digital to Analogue Converter
FCE	-	Final Control Element
$\theta_i(t)$	-	Input shaft angle
$\theta_o(t)$	-	Output shaft angle
R_a	-	Armature resistance
L_a	-	Armature inductance
I(t)	-	Armature current
F	-	Magnetic field force
1	-	Length of conductor
β	-	Magnetic field strength
v(t)	-	Velocity of conductor normal to the magnetic force

$v_b(t)$	-	Back electromotive force
K_b	-	Motor dependant constant
$\tau(t)$	-	Torque
J	-	Total moment inertia
В	-	Total viscous friction
K_T	-	Motor torque constant
e(t)	-	Error signal
K_g	-	Gain
ζ	-	Damping ratio
<i>W</i> _n	-	Natural frequency
T_p	-	Peak time
T_r	-	Rise time
%OS	-	Percent overshoot
M_p	-	Peak value of time response
f_v	-	Final value
T_s	-	Settling time
CV	-	Control variable
SP	-	Set point
y(t)	-	Output position signal
r(t)	-	Reference signal
V(t)	-	Controller output
K_p	-	Proportional gain
K_i	-	Integral gain
K_d	-	Derivative gain
DMA	-	direct Memory Access
I/O	-	Input / output
RTDs	-	Resistance temperature detectors
IC	-	Integrated circuit
AI	-	Analogue input

AO	-	Analogue output
DI	-	Digital input
DO	-	Digital output
Hz	-	Hertz
LSB	-	Least significant bit
TTL	-	Transistor-transistor logic
PA150C	-	Preamplifier unit
OA150A	-	Operational amplifier unit
IP150H	-	Input potentiometer
OP150K	-	Output potentiometer
AU150B	-	Attenuator unit
DCM150F	-	DC motor
GT150X	-	Reduction gear tacho unit
PS150E	-	Power supply
Ω	-	Ohm
RAD	-	Rapid Application Development
HTML	-	Hypertext Markup Language
VB6	-	Visual Basic Version 6.0
СОМ	-	Component Object Model
IIS	-	Internet Information Server
ASP	-	Active Server Pages
DLL	-	Dynamic. Link Libraries
API	-	Application Programming Interface
IDE	-	Integrated Development Environment
OLE DB	-	Object Link Embedded Database
MTS	-	Microsoft Transaction Server
Ne	-	negative error
Pe	-	positive error
Ze	-	zero error
Nde	-	negative de
Pde	-	positive de

Zde	-	zero de
No	-	negative output
Ро	-	positive output
Zo	-	zero output

LIST OF APPENDICES

APPENDIX	TITLE	PAGE	
A	PID results	94	
В	Software snapshot	102	
С	Slide presentation	106	
D	Software source code	134	

CHAPTER 1

INTRODUCTION

1.1 Virtual instrumentation

Virtual instrument have become the catchword in measurement technology. Virtual instrumentation easily can be connected with the term "test engineer". When attempting to interpret the meaning of "test engineer", almost everyone ties a different idea or concept to this catchword. For many, it represents a control instrument based on standard personal computers to store, evaluate, and represent test data. According to this notion, data is acquired through special measuring devices attached to a personal computer over a serial or parallel cable. Some thinks that it means a computer equipped with application and driver software and a built in transmitter as sort of low-cost alternative to relatively expensive standard alone measuring devices. Both ideas are correct, but only up to a certain point. They cover only part of this concept. Before discussing the exact definition, we will describe the principle types of computer-assisted test data acquisition [13]. Test data can be acquired in a computer in a different ways. It is important to understand the underlying architecture of a measuring device. A traditional measuring device always consists of three components, shown in figure 1.1, which perform the following tasks [13]:

- Acquire the measurement parameters (data acquisition)
- Adapt and process the measured signal (analysis)
- Output the measured value (presentation)



Figure 1.1: Structure of a measuring device.

Such measuring devices generally have fixed functions defined by the manufacturer, and they are characterized by a manufacturer-specific architecture and an inflexible user interface. Consequently, they cannot be adapted easily to changing needs.

Both the operation and the documentation are entirely manual. To add long measuring sequences that require constant changes of settings, a large amount of time is used to set the measuring devices and to document the measured values.

1.2 Virtual Instruments

Based on this background information, we are now able to define the term "virtual instrument" in a more accurate way. We speak about a virtual instrument when we create measuring systems composed of a standard personal computer, suitable software, and appropriate measuring hardware tailored to the measuring task, which is normally available only in specifically designed stand-alone measuring devices. Virtual instruments represent a visualization and centralization of complex measurement systems on a standard personal computer in the form of a virtual user interface [13]. The user sees a uniform, comprehensive single system, i.e., a complete application, consisting of many individual measuring components. This fundamental concept is the quantum leap from the conventional measuring device over computer-assisted measurement technologies to adaptable virtual measuring systems. This represents a shift from manufacturer-defined measuring devices to user-defined measuring systems. The main benefits of this concept are:

 A virtual instrument can contain any combination of industry standard hardware to acquire or output data: IEEE-488.2, RS-232 devices, VXI/MXI systems, field buses (CAN, Interbus-S, Profibus, Foundation Fieldbus, Axiom etc.), multifunction plug-in cards, DAQ instruments, image processing components, external black-box systems, or motion control. figure 1.2 illustrates the hardware architecture.

- The capabilities to analyze and represent measured data reach far beyond the boundaries of conventional measurement technology.
- A powerful software development environment and a set of hardware components allow creation of a number of virtual instruments to cover a wide range of test functions and applications.



Figure 1.2: Hardware architecture of virtual instruments.

The spectrum of virtual instruments used in practice includes classical laboratory automation, process visualization and control, automotive and aviation industries, medical applications, manufacturing industry, and research and science.

1.3 Project Objectives

The main objective of this project is to develop software with Graphical User Interface (GUI) capabilities using Microsoft Visual Basic version 6.0 in performing their position control system experiment. The GUI will be developed with the following goals:

- 1) To developed controller using software for position control system.
- 2) To ease students in performing position control system experiment.
- 3) Expose students on position control system theoretically and practically.
- 4) Enable students to understand more on application that is based on position control.
- 5) Teaching electrical engineering students real-world data acquisition and analysis.
- 6) Allowing students manipulate the information with their graphics and word processing packages to generate professional quality lab reports.
- Empowering students to acquire experimental data in this laboratory experiment directly from the software.
- 8) Permits students to meaningfully experiment with physical relationship in a readily interpretable, graphics format.

Besides that, the objectives of each experiment that have to be performed by student are basically divided into 4 classical control experiments listed as below:

(i) Experiment 1 – Closed-loop Position Control System.
 Objective: To demonstrate a simple motor-driven closed-loop position control system.

- (ii) Experiment 2 Effect of Gain Changes On Deadband.
 - Objective: To study on the deadband of a position control system and the effect of gain changes upon the Deadband.
- (iii) Experiment 3 Unit Step Response To Gain Changes.
 Objective: Visual study of step response to gain changes upon position control system.
- (iv) Experiment 4 Velocity Feedback.
 Objective: Study on the effect of velocity feedback in a position control system.

Besides these experiments, an additional experiment using Direct Digital Control (DDC) techniques will be implementing in this software to allow student to observe system response behaviour upon changes on the proportional gain, integral gain and derivative gain. The objective of these experiments is to enable student to differentiate between Direct Digital Control and Classical Control. Hence, this will allow student to control the position of the control system in real-time with a PC-based system and aware them of the important role, played by personal computers in engineering nowadays.

Based on the above experiment, it also allow student to differentiate between Supervisory Control and Direct Digital Control where in Supervisory Control, computers are used for monitoring purpose only but in Direct Digital Control, control loop is replaced by computers to interface with process measurement and control some of the physical process parameters.

1.4 Project Scope

The scope of the project is listed below:

- i. To apply direct digital control (DDC) in position control system.
- ii. To develop a PID and a fuzzy logic controller which will be use to control the output response.
- iii. To make comparison of the performance between PID controller and fuzzy logic controller.
- iv. To developed Graphical User Interface to help the student to visualize and analyze the system. Besides, it also covers the method on enhancing the original experiment for student so that they can finish their experiment on time and in the mean time, they can grab more understanding through visual presentation.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The limits of my language mean the limits of my world.

-Ludwig Wittgenstein Tractatus Logico-Philosophicus [13]

The evolution of programming paradigms was marked by the transition from machine language to assembly languages, and from these to functional, procedural, and, eventually, object- oriented, high-level languages. During these past years, a totally new paradigm evolved along these lines, offering problem solving approaches in a new dimension graphical data flow programming in the form of so called "virtual instruments". Where, virtual instrumentation is defined as a visualization and centralization of complex measurement system on a standard personal computer in the form of a virtual user interface. A user sees a uniform comprehensive single system, consisting of many individual measuring components in a VI. There has been an increasing awareness in the importance of virtual instrumentation in control engineering especially in position control system. Virtual instrumentation in control and measurement system can reduce hardware drastically thus lowering the cost. The Virtual Instrument approach addresses several of these user equipments. Significant strides in development of application software technology has made it possible to offer users intuitive, graphics or code based programming features and state of the art connectivity. These factors have dramatically eased the user's pain in putting together a VI. The need for integrated test products and suited in automated measurement environments is another factor influencing the use of VIs in electronics manufacturing, data acquisition, automotive, and laboratory research application.

It is evident form the structure of VIs that software plays a vital role in defining the utility, control, access and applications of the VI. It seems that the virtual instrumentation industry is on the launch pad to a new world of user friendly, open architecture instrumentation. Vendors and users realize that VIs is not expected to replace all bench top instrument applications in the future. However, it is certainly gaining grounds in several applications in the electronics, telecommunications, semiconductor, and automotive industries previously considered inaccessible for computer based measurements.

2.2 Previous Research

Previous works have been done on real-time system and virtual instrumentation. These researches that are relevant to this project will be discussed next to demonstrate continuity from previous researches.

9

2.2.1 "Motor Drive Control System for Education and Research"

Jee Ming Lee and Jung-Tang Huang (1995) developed research on "Motor Drive Control System For Education and Research". An industrial level real time motor control education tool has been designed for the enhancement of the related basic conception between control theory and physical system. The educational tool set presents a life of control methodology in a servo motor system. The set of DC motor system is assembled as a physical experiment control system. The learning process provides a friendly and humanized environment. The multitasking operating module is adopted to enable the educational control package to swap the environment flexibly. The real time control module accurately analyzes the motor position control, which a set of experiment curriculum is designed to apply basic automatic control course [5].

2.2.2 "Development of a Distant Laboratory using Labview"

K.K.Tan, T.H. Lee and F.M.Leu (1999) have conducted a research on how to "Development of a Distant Laboratory using Labview". This paper presents the development of a distant laboratory experiment for engineering students. The experiment is concerned with real-time modeling and control of a pilot-scale DC servomotor. The World-Wide-Web (WWW), with its ability to deliver multimedia and graphics across the world, is used as the medium to link the student to a remote experimental setup located in the distant laboratory. This paper describes the hardware and software constructs, which facilitate students to perform the experiment remotely and efficiently, using only a commonly available, user-friendly web browser [7].

2.2.3 "Real-Time Position Control of Free-Electron Laser Beams: Theory and Experiments"

M.Moallen (2000) has done a research on "Real-Time Position Control of Free-Electron Laser Beams: Theory and Experiments". This paper addresses feedback control of electron beam positions in Duke University Free-Electron Laser Facility. The input-output relationship for this system is algebraic and the system is subject to noise and disturbance inputs. The information about beam position is obtained by sensory device called Beam Position Monitors distributed throughout the storage ring. In contrast, the beam position profile is controlled by passing currents through the coils of correction magnets distributed around the ring. A desired control objective is to move the stored electron beam to specific locations in the vacuum chamber and to stabilize the orbit in the presence of disturbances. The technical details of implementing the data acquisition and control scheme and their integration with the existing control system modules are presented along with experimental results [10].

2.2.4 "Closed-loop Position Control System using Labview"

Kelvin R. Aaron, Noreen L. Foster, Danielle P.Hazel, A.M.Hasanul Basher (2002) has designed a "Closed-loop Position Control System using Labview". A closed-loop DC motor control system is developed using National Instrument's (NI) Data Acquisition (DAQ) Board (Model MIO 6040E), LabVIEW software package, and DAQ Signal Accessory Board for smooth and accurate positioning. The motor can be rotated either clockwise (CW) or counter clockwise (CCW) as needed. A linear feedback that employs a potentiometer provides the current position of the motor. The LabVIEW software is used to develop virtual instrument (VI) that includes a front panel and a functional diagram. The VI reads the desired position of the motor entered by user on the front panel and the current motor position via feedback, and determines its amount and direction of rotation. A smooth cubic polynomial that is continuous and has a continuous first and second derivative is utilized to represent the angular position of the motor. The VI uses the polynomial to generate a velocity profile, which in turn is converted into a voltage profile for the motor armature through a conversion formula. This conversion formula is determined through the use of motor speed voltage characteristics obtained from laboratory experiments. Reading the motor position and sending voltage to motor circuit are accomplished via the DAQ's input and output ports respectively. The system is tested in the laboratory for different angular displacements and initial positions of the motor [8].

2.3 Virtual Instrumentation in Position Control System

Mainly the goal of this research is to very effectively fill the gap between textbooks, which work on theoretical manipulation, and physical systems, which work on the performance running. Essential features of an educational automatic control system program in control engineering will be developed. These experiment contents of an undergraduate engineering course in control system analysis and designed are greatly enhanced.

It is the aim of this research to develop a user-friendly graphical user interface, which is easy to navigate. Thus providing a high-quality GUI that improves usability, end-user perception and marketability.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 **Project Overview**

The main purpose of this project is to design a software based program in controlling the position of a servomechanism or also known as servomotor. It involves understanding on how to interface personal computer (PC) with position control system through Data Acquisition System (DAS). This project can be divided into two parts namely hardware and software. Hardware parts involve more in interfacing MS150 Modular servo System and Data Acquisition System with a personal computer. While the software part include programming real-time software using Microsoft Visual Basic 6.0. In this chapter, three main topics will be discussed. We will discuss on:

- i. Basic concept of control system:
 - Position Control System
 - Direct Digital Control (DDC) system
 - PID controller
 - Fuzzy Logic Controller

ii. Software Review:

- Virtual instrumentation (VI)
- Graphical User Interface (GUI)
- Microsoft Visual Basic 6.0 software
- iii. Hardware Review:
 - Data Acquisition System (DAS)
 - MS150 Modular Servo System

In the earlier stage, literature review and experiment are performed manually to understand the concept of the controller. Interactive software will be designed using Microsoft Visual Basic 6.0. The software will be equipped with a set of graphic instructions to ease any mistake when performing experiment. Finally, the software will be integrated with hardware to produce a GUI position control system.

In designing digital controller in controlling the position control system, each theory and concept play an important part in understanding the operation of position control system. In contrast, understanding on the concept of Direct Digital Control (DDC), Data Acquisition System (DAS), Proportional, Integral and Derivative (PID) controller and fuzzy logic controller also play an important role in succeeding this project. The following topic will be described briefly on each concept and theory.

3.2 Position Control System

In a position control system, a common requirement is for a motor to rotate an output shaft to the same angle as an input shaft. The general form of the block diagram for closed loop position control system is shown in figure 3.1.



Figure 3.1: general form of the block diagram for closed loop position control system.

In a simple position control system, an error signal proportional to the difference between input of set position and actual position is generated. The error signals then fed into controller. In the figure the controller will be pure gain controller. The output of the controller drives the motor closer to the desired position. The overall of position control system is likely second order system. This mean when there is step voltage input, the output will take time to rise to the required output and likely oscillate about it for a while before settling down to steady state value. We can think of the situation like this. The voltage step input causes the motor shaft to accelerate and change position of the load, and because of time delays inherent in the system there is a lag before the position of the load is fed back to the summing amplifier, so the load overshoots its position and oscillates about the required position.
Figure 3.2 shows the typical form of the response of an under damped system to the step input [12]. Certain terms are used to specify such a performance:

• Rise time, T_r

Define as the time taken for response from 10% final value to the 90% of final value. It shows how fast a system response to input.

• Peak time, T_p

Define as the time taken for the response to rise from 0 to the first peak value or the maximum value that the system can achieve.

• Overshoot, M_p

Define as the maximum amount by which response overshoots the steady state value or final value.

• Settling time, T_s

Define as the time taken for the response to fall within and remain within some specified percentage e.g. 2% of the steady state value. It used as a measure of the time taken for the oscillation to die away.



Figure 3.2 Standard measurements on Step Response of a Control System

An alternative and more common way of achieving the same effect is to use a second feedback, loop which gives measurement related to the rate at which the position of load is changing. This called velocity feedback. Figure 3.3 shows how single position loop diagram can be modified to include velocity feedback.



Figure 3.3: single position loop diagram can be modified to include velocity feedback.

The overall system performance will improve by changing type of feedback. Overshoot is caused when the motor shaft is going too fast to be stopped suddenly due to inertia. Thus by negatively feedback some signal proportional to a shaft velocity, we can counteract the effect of inertia. This type of feedback is appropriately called velocity feedback. The most common way to achieve this is to use the tachometer signal, which is proportional to the velocity, in some minor feedback loop hence the name minor loop feedback. Note that to obtain minor loop feedback control system, include velocity feedback by means of tachometer and another controller to the simple position control system. Also note that the feedback signal is subtracted from the error signal, as required for stability.

For the position control the input and output, transducers must measure input and output shaft angles and produces a control signal (or known as *error*) proportional to the angle between the shafts. In this project, a simple form of error channel is to mount a potentiometer onto each shaft and then added together the voltages from the potentiometer sliders with an operational amplifier shown in figure 3.4 [11].



Figure 3.4: Simple Error Channel

If the two shaft angles θ_i , θ_o are equal, the two slider-voltages are equal and opposite and cancel at the amplifier output. Any misalignment will give an error signal proportional to the misalignment ($\theta_i - \theta_o$), which can operate the forward path. An important factor in an error channel is the error factor K_e , which gives the $V/^{\circ}$ (or possibly radian) of misalignment.

In order that the closed-loop system output shaft follows the input shaft movement correctly. It is necessary that the polarity of the various connections be such that an error signal causes the motor to rotate in a direction to reduce the error and not increase it [11]. In this section, the model of a servo system (also known as positioning system) is derived. A simple DC motor with its schematic is shown in figure 3.5 [12].



Figure 3.5: DC-Motor (a) Wiring Diagram (b) Sketch (c) Schematic

The motor is armature controlled DC motor with a constant field. The motor has an armature resistance, R_a and inductance, L_a , respectively. A magnetic filed is developed by stationary permanent magnets or a stationary electromagnet called the fixed field. A rotating circuit called the armature through which current, i(t) flows, passes through this magnetic field at the right angle and produces force,

Force,
$$F = Bli(t)$$
 (3.1)

Where, i(t) is the current passing through the armature circuit, l is the length of conductor and B is the magnetic field strength. The force produced resulting torque that turns the rotor (the rotating member of the motor). There is another phenomenon that occurs in the motor, a conductor moving at the right angles to a magnetic field generates a voltage at the terminal of the conductor,

$$Voltage, \ e = Blv \tag{3.2}$$

Where v is the velocity of the conductor normal to the magnetic field. Since the current carrying armature is rotating in a magnetic field, its back electromotive force (or back emf) is proportional speed. (Bishop R.H. Dorf R.C, 1998) Thus,

Back emf,
$$v_b(t) = K_b \frac{d\theta(t)}{dt}$$
 (3.3)

From the above equation, K_b is the motor dependent constant, $\frac{d\theta(t)}{dt}$ is the angular velocity of motor and $\theta(t)$ is the shaft position.

The typical equivalent mechanical loading on a motor that connected to the motor shaft including total moment of inertia, *J* and total viscous friction, *B*. Assume that $\tau(t)$ is the torque developed by the motor, then writing the following equation

Torque,
$$\tau(t) = J \frac{d^2 \theta(t)}{dt^2} + B \frac{d \theta(t)}{dt}$$
 (3.4)

The developed torque for this motor can be given by

Torque,
$$\tau(t) = K_T i(t)$$
 (3.5)

With i(t) is the armature current and K_T is the motor torque constant.

The equivalent equation of the armature inductance is relatively small compared to the armature resistance and can be ignored [12], thus we assume the inductance L_a can be ignored, which in the case for many servomotors [2], then

$$v(t) = i(t)R_a + v_b(t)$$
 (3.6)

$$i(t) = \frac{v(t) - v_b(t)}{R_a}$$
(3.7)

$$i(t) = \frac{v(t)}{R_a} - \frac{K_b}{R_a} \frac{d\theta(t)}{dt}$$
(3.8)

$$\tau(t) = K_T i(t)$$

$$\tau(t) = \frac{K_T}{R_a} v(t) - \frac{K_b K_T}{R_a} \frac{d\theta(t)}{dt} = J \frac{d^2 \theta}{dt^2} + B \frac{d\theta(t)}{dt}$$
(3.9)

$$J\frac{d^2\theta(t)}{dt^2} + \left(\frac{BR_a + K_bK_T}{R_a}\right)\frac{d\theta(t)}{dt} = \frac{K_T}{R_a}v(t)$$
(3.10)

If the armature inductance cannot be neglected, then the model is third order system. By using Laplace transform and solving the transfer function becomes [11] [12],

$$\frac{\theta(s)}{V(s)} = Gp(s) = \frac{\frac{K_T}{R_a}}{Js^2 + \left(\frac{BR_a + K_T K_b}{R_a}\right)s}$$
(3.11)

$$\therefore \frac{\theta_m(s)}{E_a(s)} = \frac{\left(\frac{K_i}{R_a J_m}\right)}{s\left[s + \frac{1}{J_m}\left(D_m + \frac{K_i K_b}{R_a}\right)\right]}$$
(3.12)

where;

$$J_{m} = 30 \times 10^{-6} kgm^{2}$$

$$K_{b} = 60 \times 10^{-3} Vsrad^{-1}$$

$$K_{t} = 17 \times 10^{-3} NmA^{-1}$$

$$R_{a} = 3.2\Omega$$

$$D_{m} = small(can't \text{ be quote})$$

The value of J_m , K_b , K_t , R_a and D_m is given in the MS150 modular servo system. Thus when substitute these values into equation (3.12) we get the plant transfer function in equation (3.13).

$$G(s) = \frac{(177.0833)}{s[s+10.625]}$$
(3.13)

3.2.2 Direct Digital Control (DDC)

Direct Digital Control (DDC) technology gives today's industrial user the flexibility to tailor near any system to specific needs using convenient, softwarebased control. A simple concept that has blossomed into a complex array of components and systems, DDC is basically a microprocessor-based technology in which a controller performs closed-loop functions via sophisticated algorithms and strategies. A typical schematic of direct digital control is shown in figure 3.6.



Figure 3.6: Direct Digital Control Schematic.

The controller function handles inputs and outputs electronically while the software provides the logic. Within the controller, the analogue signal enters the system where it is converted to a digital form so that the data can be acted upon according to a set of parameters or formulas. The result is the execution of commands directly to end device. Because the programs used by the controllers are in digital form. The operations are known as direct digital control [3] [4].

DDC control brings speed, precision, flexibility to control functions at low cost. It can range from the simple control of a single loop to the application of larger systems controlling a multitude of loops. Controllers may function as stand-alone units or be networked into systems using a PC as a host to provide additional functions. Most intelligent DDC devices residing on a network today are capable of sophisticated equipment control and remote communication.

3.2.3 PID controller

PID stands for Proportional-Integral-Derivative. This is a type of feedback controller whose output, a control variable (CV), is generally based on the error between some user-defined set point (SP) and some measured process variable (PV). Each element of the PID controller refers to a particular action taken on the error:

There are many situations that require some type of servo-control system. This section reviews the fundamental of PID controllers and presents detailed simulations or design for development of digital servomotor controller. PID controllers are commonly used to regulate the time-domain behaviour of many different types of dynamic plants. These controllers are extremely popular because they can usually provide good closed loop response characteristics, can be tuned using relatively simple rules and are easy to construct using either analogue or digital components. Consider the feedback system show in figure 3.7 where we assume that the plan is a DC motor whose shaft position must be accurately regulated.



Figure 3.7: General Feedback Architecture

The PID controller K(s) is placed in the forward path, so that its output becomes the voltage applied to the motor's armature. The feedback signal is either an angular shaft position or velocity, measured by a potentiometer or a tachometer, respectively. In the block diagram, these transducer dynamics are in the feedback path. The output position signal y(t) (or velocity signal) is summed with a reference signal r(t), or command signal, to form the error signal e(t). Finally, the error signal is the input to the PID controller.

The concept for this closed-loop system is simple. The sensor, which is attached to the motor shaft, provides a voltage that is compared to the reference voltage r(t) at the summer. When this error signal is non-zero, there will be an input to the controller, and hence some action taken by the DC motor. Once the sensor signal is equal to the reference signal, there is no input to the controller and no voltage applied to the motor, causing the motor to stop. However, this simple explanation does not provide us a with a method whereby the motor position can be brought to an exact position (with absolute accuracy) and does not tell us how to make the motor perform this positioning task as quickly as possible.

3.2.3.1 Analog PID controller

The PID controller is defined by the following relationship between the controller input e(t) and the controller output v(t) that is applied to the motor armature:

$$v(t) = K_{p}E(t) + K_{I}\int_{0}^{r} e(\tau)d\tau + K_{D}\frac{de(t)}{dt}$$
(3.14)

Taking the Laplace transform of this equation gives the transfer function of K(s):

$$K(s) = \frac{V(s)}{E(s)} = (K_P + \frac{K_I}{s} + K_D s)$$
(3.15)

This transfer function clearly illustrates the proportional, integral, and derivative gains that make up the PID compensator. Select new definitions for the gain terms according to;

$$K_P = K \tag{3.16}$$

$$K_I = \frac{K}{T_I} \tag{3.17}$$

$$K_D = KT_D \tag{3.18}$$

Then, the transfer function can be expressed to easily show that the PID controller leads to a pole at the origin of the Laplace plane and design freedom over two zeros:

$$K(s) = \frac{KT_D}{s} (s^2 + \frac{1}{T_D}s + \frac{1}{T_I T_D})$$
(3.18)

The block diagram for the PID controller can be sketched in several different ways. One possible block diagram is shown in figure 3.8.



Figure 3.8: Block diagram for PID Controller

Proportional mode is the heart of a closed-loop control system. It provides an output, which is directly proportional to the error signal. It is the only control mode, which can be used alone. Integral and derivative modes can only be used when combined with at least the proportional mode. Proportional control amplifies the error to motivate the plant towards the desired response. It can reduce but not eliminate stead-state error. Proportional control can also produce excessive overshoot and oscillation. Remember that by increasing K will increase the overshoot of the

system, but it will decrease the system rise time. If the process response to a change in set point is not sufficient and a large offset error persists then, perhaps the proportional gain is too low. Increase the proportional gain till a satisfactory response is obtained. On the other hand if the proportional gain is too high, then the oscillations may be developed in the system.

Integral can eliminate steady state error. Usually small integral gains are used to avoid destabilizing the closed-loop system. In many systems K_i is responsible for driving error to zero, but to set K_i too high is to invite oscillation or instability or integrator windup or actuator saturation.

Derivative Control acts on the slope of the error, thereby minimizing overshoot. It provides "rate" feedback, resulting in more damping. High derivative gains can increase t_{rise} and $t_{settling}$. It is difficult to realize in practice because differentiation leads to "noisy" signals. In many systems K_d is responsible for system response: too low and the process variable (PV) will oscillate; too high and the PV will respond sluggishly. The designer should also note that derivative action amplifies any noise in the error signal. Derivative mode is effective only at high frequencies. It does not respond to any steady state error value but to acts swiftly against the rate of change of error. This gives the derivative mode the ability to respond to the upcoming changes by reacting to the rest in process output. For this reason, derivative mode of control is also known as anticipatory control. Since the controller output is dependent upon the slope of the error signal, it is not very satisfactory when noise is present in measurement. For processes with very small time constant, the derivative mode of control can be troublesome.

The general characteristic of a PID controller is that a proportional controller (K_p) will have the effect of reducing the rise time and will reduce, but never eliminate, the steady-state error. An integral control (K_I) will have the effect of eliminating the steady-state error, but it may make the transient response worse. A derivative control (K_D) will have the effect of increasing the stability of the system,

reducing the overshoot, and improving the transient response. Effects of each of controllers K_p , K_D , and K_I on a closed-loop system are summarized in the table 3.1.

Close	ed Loop	Rise Time	Overshoot	Settling Time	Steady-state
Res	sponse				Error
	K _p	Decrease	Increase	Small change	Decrease
	KI	Decrease	Increase	Increase	Eliminate
	K _D	Small Change	Decrease	Decrease	Small change

Table 3.1 Effect of Each PID Controller

Note that these correlations may not be exactly accurate, because K_P , K_I , and K_D are dependent of each other. In fact, changing one of these variables can change the effect of the other two.

3.2.3.2 Digital PID controller

To discretise the controller, we need to approximate the integral and the derivative terms to forms suitable for computation by a computer. By using z-transform, we may discretise the analogue PID controller to obtain the pulse transfer function for the digital PID controller as the following equation [6].

Positional Form of the PID control scheme is given as ;

$$G(z) = \frac{C(z)}{R(z)} = K_p + \frac{K_I}{1 - z^{-1}} + K_D(1 - z^{-1})$$
(3.19)

where

$$K_{P} = \text{proportional gain} = K - \frac{KT}{2T_{i}} = K - \frac{K_{I}}{2}$$
$$K_{I} = \text{Integral gain} = \frac{KT}{T_{i}}$$
$$K_{D} = \text{Derivative gain} = \frac{KT_{d}}{T}$$

Equation (3.19) can be written in a sequence operation as in equation (3.20). Equation u (k) will be used to develop the coding program for PID controller which corresponding to sequence operation [16].

$$u(k) = K_{p}e(k) + K_{i}T_{s}\sum_{n=0}^{k}e(n) + K_{d}T_{s}^{-1}[e(k) - e(k-1)]$$
(3.20)

where $T_s = time \ sampling = 1 \ ms$ $e(k) = error \ between \ desired \ value \ and \ actual \ value$ $\sum_{i=0}^{t} e(k) = total \ error$ $e(k) - e(k-1) = deviation \ error$

The value of K_p , $K_i \& K_d$ was calculated using root locus technique. The tuning method used is the Ziegler Nichols. The PID controller has been simulated using matlab to ensure the manual calculation is correct. The value of K_p , $K_i \& K_d$ is as follow.

$$K_{p} = 1.266$$

$$K_{i} = 0.1261$$

$$thus T_{i} = \frac{K_{p}}{K_{i}} = 10.0444 \approx 10$$

$$K_{d} = 0.0563$$

$$thus T_{d} = \frac{K_{d}}{K_{p}} = 0.044 \approx 1.0$$

After calculating these values, now we can write the coding for the PID controller. The flow chart for the system can be virtualized as in figure 3.9.



Figure 3.9.: Flow chart for the PID controller system.

3.2.4 Fuzzy Logic Controller

In this chapter, we show the design flow of the simulator and FLC for the position control system in flow charts. The position control system simulator has been developed using Microsoft visual basic 6.0 [15].

The steps that we are needed to do are:

- Step 1: define quantization for state (input) variables, v and h and output variable, force in universe of discourse.
- Step 2: define a membership function for the input and control output.
- Step 3: define the rules base and summarize them in an FAM table. The values in the FAM table are the output control.
- Step 4: define the initial condition
- Step 5: conduct the simulation. For this part we use max-min configuration. We defuzzify using centroid method.
- Step 6: Compare the calculation with the simulation using matlab.

Figure 2.1 is the flow chart in designing the FLC. In developing the design, we need to initialize and declare what are the inputs and outputs variables. Then, define the math model so that we can carry out a simulation using the previous value and the changes in the output. All the parameter is then used by FLC to be simulated. The FLC design flow is shown in Figure 3.10. FLC is a subroutine to the main simulator program. The FLC design methodology is shown in Figure 3.11.



Figure 3.10 Flowchart of Simulator



Figure 3.11 FLC Design methodology

3.2.4.1 Fuzzification

The position control of the servo system configuration was designed based on 2 inputs and 1 output. Inputs for this controller are error and rate (de), and the motor output. Fuzzification is where we define the quantization and membership function for input variable, error and rate (de) and output variable, motor output in universe of discourse. It involves the conversion of the input and output signals into a number of fuzzy represented values (fuzzy sets).

Usually fuzzy sets are overlapping by about 25%. It can be observed that in fuzzy logic control the transition from one fuzzy set to another provides a smooth transition from one control action to another. If there is no overlapping in the fuzzy sets then the control action would assemble bivalent control (transition from one error region to another is rather abrupt). On the other hand if there is no overlapping, there would be a lot of fuzziness and this blurs the distinction in the control action. Figure 3.12 show the input and output variable that is used in the system.



Figure 3.12 show the input and output variable.

Quantization of input and output variables:

1) Input variables:

• *Error(e)*

Quantized into 3 fuzzy sets: negative error (Ne), zero error (Ze) and positiveerror (Pe). Figure 3.13 show the error input variable.



Figure 3.13: The error input variable.

• Rate(de)

Quantized into 3 fuzzy sets: negative rate (Nde), zero rate (Zde) and positive rate (Pde). Figure 3.14 show the rate input variable.



Figure 3.14: The rate input variable.

2) Output variable:

• Motor output

Quantized into 3 fuzzy sets: negative output (No), zero output (Zo) and positive output (Po). Figure 3.15 show the motor output variable.



Figure 3.15: The motor output variable.

3.2.4.2 Knowledge Base

The knowledge base of a fuzzy logic controller consists of a data base and a rule base. The basic function of the data base is to provide the necessary information for the proper functioning of the fuzzification module, the rule base and the defuzzification module. This information includes the meaning of the linguistic values of the membership functions of the process state and the control output variables, physical domains and their normalized counterparts together with the normalization, renormalization and scaling factors and the type of the membership function of a fuzzy set.

The basic function of the rule base is to represent the expert knowledge in a form of if-then rule structure. For this project, we derive the fuzzy logic rules based on expert experience and control engineering knowledge. This method is the least structured of the four methods and yet it is one of the most widely used today. It is based on the derivation of rules from the experience based knowledge of the process operator and/or control engineer. The fuzzy logic can than be derive into a 3 x 3 rules which consist of 9 rule. Figure 3.16 show the fuzzy logic rules.



Figure 3.16: The fuzzy logic rules.

The fuzzy logic rules-based are:

- 1) If error is Ne AND de is Nde THEN output is No
- 2) If error is Ne AND de is Zde THEN output is No
- 3) If error is Ne AND de is Pde THEN output is Zo
- 4) If error is Ze AND de is Nde THEN output is No
- 5) If error is Ze AND de is Zde THEN output is Zo
- 6) If error is Ze AND de is Pde THEN output is Po
- 7) If error is Pe AND de is Nde THEN output is Zo
- 8) If error is Pe AND de is Zde THEN output is Po
- 9) If error is Pe AND de is Pde THEN output is Po

3.2.4.3 Inferencing

The inference mechanism provides the mechanism for invoking or referring to the rule base such that the appropriate rules are fired. There are several methodologies to derive the Inferencing method. Two most common methods used in fuzzy logic control are the max-min composition and the max-(algebraic) product composition. The inference or firing with this fuzzy relation is performed via the operations between the fuzzified crisp input and the fuzzy relation representing the meaning of the overall set of rules. As a result of the composition, one obtains the fuzzy set describing the fuzzy value of the overall control output.

For this system, max-min composition is used for the Inferencing. Figure 3.17 shows the firing strength for the rule base using max-min inference.



Figure 3.17: The firing strength for the rule base using max-min inference.

3.2.4.4 Defuzzification

Defuzzification is a mapping from a space of fuzzy control actions defined over an output universe of discourse into a space of non-fuzzy (crisp) control action. This process is necessary because in many practical applications crisp control action is required to actuate the plant. The defuzzifier also performs an output denormalization if normalization is performed in the fuzzification module. There are many methods of defuzzification that have been proposed in recent years. For this system, centroid method is used for defuzzification. It is given by the algebraic expression:

$$z^* = \frac{\sum u_z(z) \cdot z}{\sum u_z(z)}$$
(3.21)

3.3 Software Review

Software review is basically a discussion on the virtual instrumentation and graphical user interface that will be developed in the project. Here, the software part of the project will be discussed which is programming with Microsoft Visual Basic [17] [18].

3.3.4 Virtual Instrumentation - Graphical User Interface (GUI)

Graphical User Interface (GUI) can be defined as an intuitive, easy-to-use method of interfacing with computer program by means of graphical screen display. It can resemble the front panel of instrumentations or other objects associated with a computer program [9] [14].

3.3.4.2 General GUI Design Considerations.

The GUI design will reflects the user's mental model rather than the implementation model. Program features and functions support only required user goals. No superfluous features have been added simply because they are easy to implement, or as a result of the personal biases of the programmers. The GUI design reflects the expected characteristics and capabilities of the user population - visual, physical and cognitive capabilities; cultural and technical background; domain experience; education level and so forth.

The GUI design reflects any unusual characteristics of the users' environment - dangerous or hazardous work areas, excessive noise, bright or dim lighting, and so forth. The design is optimized for human perception and information processing capabilities. For example, the "seven plus or minus 2" item limitations of the shortterm memory, and "recognition over recall" figure prominently in the design choices. All user classes are adequately represented, and the potentially diverse needs of all user classes are balanced appropriately.

All GUI items are prioritized. Critical and frequently accessed items are prominent - and more easily accessed - than less important items. All unnecessary GUI controls and indicators have been eliminated. The GUI hierarchy geometry is optimized to reduce panel clutter, and to minimize the total number of clicks required to access each function.

Panel layouts and GUI object placement logically reflect user task sequences. In particular, the user is never required to jump between panels or applications to complete a single task. Where possible, all necessary controls for each task are accessible from a single panel. The user is not required to manually copy information displayed on one panel into a control on another panel. The program automatically transfers shared information between panels and eliminates unnecessary "busy work" for users.

Tedious, mundane, and predictable tasks are automated to improve user efficiency, but not at the expense of adequate user control. Qualified users are permitted to modify or bypass automation as necessary. Tool tips, Control Descriptions, and keyboard shortcuts have been included for power users. User actions and task sequences lead naturally from one to the next.

The function of all GUI items is visually apparent, and all objects (including custom controls) behave as expected. For example, GUI buttons look and respond like real buttons from the physical world, and don't produce unexpected outcomes. GUI metaphors, visual or otherwise, are natural and consistent with their real-world counterparts. The GUI adheres as necessary to applicable standards documents and company-wide style guides.

Where measurable usability targets have been set, the final design meets or exceeds these targets. Direct user feedback has been collected, and all design shortcomings have been (or will be) addressed in subsequent iterative cycles. Documentation and "Help" files have been prepared with the same level of enthusiasm and attention to detail as the software.

3.3.1.2 Visual Presentation

Panels supposed to immediately appear uncluttered and organized. The style and mood of the GUI design is appropriate for the application and its expected users. The design style is consistent from one panel to the next, and all panels look as though they belong to the same application. A limited number of unique design elements - control and indicator types, colors, fonts, proportions, and so forth - are used consistently and thematically, thereby promoting a sense of application-wide unity. Limited color palettes have been selected and applied consistently to enhance both aesthetics and mental model development. Bright colors are used sparingly for highlights, or to attract the user's attention.

The layout of each panel creates a visual hierarchy, drawing users' eyes to the most important items first. GUI items are arranged and ordered to reflect natural visual scanning patterns (left to right, top to bottom in Western cultures). Task sequences are mapped directly to the natural scanning patterns to improve user productivity. Positive and negative space has been used effectively to make panel designs appear balanced and uncluttered. GUI text, control labels, and menu text is clear, descriptive, and concise. Controls have been grouped to permit the elimination of redundant label text. Error messages are brief, informative, and designed to help users locate and overcome difficulties. Unnecessary technical jargon has been eliminated from panels and dialog boxes. The selected font style, size, and color combinations provide adequate readability for users with common visual deficiencies.

Where panel resizing is permitted, panels have been designed to resize gracefully. Bitmapped graphics have been avoided on resizing panels, and the number of decoration elements has been kept to a minimum. GUI panels targeted for cross-platform deployment have been verified visually on all target platforms. Panels have been designed to provide a pleasing visual presentation, but form always follows function.

3.3.1.3 VI Diagram Considerations

The VI architecture supports the most logical user task sequences - user efficiency was not sacrificed in order to simplify the VI architecture or wiring. GUI objects are divided between panels based on priority and task requirements. GUI organization was not permitted to evolve simply as a by-product of dataflow considerations, the application architecture, or the organization of the sub VI hierarchy. Potential dataflow limitations have been overcome to provide continuous, real-time GUI feedback in response to user actions. Modal GUI architectures have been avoided unless the underlying user task sequences are also modal. Event-driven architectures with numerous simultaneous options are favored to promote exploration and enhance mental model development.

Cancel and programmatic Undo functions are included to promote exploration and enhance GUI forgiveness. Presets, default values, and recently accessed item menus are included to alleviate repetitive tasks. Similarly, state information is preserved from one run (or sub VI call) to the next; thereby minimizing the time users spend resetting configuration parameters.

3.3.1.4 Additional GUI Object Considerations

Custom menus are included to reduce panel clutter while providing nonmodal access to numerous functions and parameters. Menu items accurately and concisely describe the functions they reference. Menu items and other GUI objects are disabled (grayed-out) when the functions they invoke are not compatible with the current operational mode. Tab control implementations are consistent with the underlying file folder metaphor. In non-standard implementations, the tabs are hidden. Nested tabs are avoided. ActiveX objects, DLLs, CINs, and other platformspecific extensions have been used only when a 100% G solution is impractical or impossible.

3.3.2 Microsoft Visual Basic

The software that will be used in the project is Microsoft Visual Basic Version 6.0. The Microsoft Visual Basic is the most productive tool for creating high-performance components and applications. Visual Basic 6.0 offers developers the ability to create robust applications that reside on the client or server, or operate in a distributed n-tier environment. Visual Basic 6.0 is the Rapid Application Development (RAD) tool available either as a stand-alone product or as a part of the Visual Studio 6.0 suite of tools [17] [18].

Visual Basic 6.0 provides a complete set of tools for integrating databases with any application. It visually creates reusable record set command objects with drag-and-drop functionality. Bind to multiple data sources for data aggregation and manipulation. By using the new Data Environment designer, developers can quickly drag-and-drop custom data-bound controls to create forms; they can create reports with the new Data Report Designer. Creation of custom data hierarchies is as easy as filling out a dialog box and dragging the command to the form. It can also develop multimedia-rich applications using the document object model and Dynamic HTML surface as the user interface. The Uses of new String Functions and features such as "Retain in Memory" can be able to create faster, more scalable applications and components.

Microsoft VB6 (Visual Basic 6) represents a significant leap forward in some aspects, particularly database programming and connectivity and Internet programming, both for IIS on the server-side and for Internet Explorer 4.x on the client-side. VB6 is definitely a robust package and, with over 3.2 million developers using it, it remains one of the most popular development languages for Windows. There are tens of thousands of ActiveX controls from third party developers that can be implemented in the projects, and which is remarkably easy to create ActiveX controllers.

The visual basic can also share code with Labview programming. While some programming projects begin with a blank slate, many others would be greatly expedited if existing code could be reused. This is especially true when the new solution is to be developed in a different environment. This is not to say that it is necessary to develop components of a LabVIEW application in another language. LabVIEW is a rich language that is designed for rapid development and can be used to create an application from the ground up.

3.4 Hardware Review

Generally, two sets of hardware components used in this project, one the MS150 Modular Servo System and Data Acquisition System (DAS). The Data Acquisition System (DAS) consists of AX5412H PC Bus Data Acquisition Card and control board.

3.4.1 Data Acquisition System (DAS)

Data acquisition is defined as collecting and measuring electrical signals from sensors, transducers, and test probes or fixtures and inputting them to a computer for processing. Sometimes, it is defined as collecting and measuring the same kind of electrical signals with A/D or digital input/output (DIO) boards plugged into a PC, and possibly generating control signals with D/A or DIO boards in the same PC.

Many applications use plug-in boards to acquire data and transfer it directly to computer memory. Others use DAS hardware remote from the PC that is coupled via parallel or serial port. A PC-based DAQ system depends on each of the following system elements:

- i) Sensors and Transducers
- ii) Signal Conditioning
- iii) DAQ Hardware
- iv) Personal Computer
- v) Software

Typically a computerized data acquisition and control system is constructed as figure 3.18.



Figure 3.18: Computerized data acquisition and control system.

The following sessions will gives an overview of each element and explains the most important criteria of each element of a PC-based DAS.

3.4.1.1 DAQ Hardware

The AX5412 data acquisition board is used in this project as an interface between computer and the servomotor plant. Accompanied with the AX5412X is the AX750 board [1]. It is a screw terminal panel with standoffs to allow tabletop operation. The Axiom Data Acquisition Card Driver is designed to support all Axiom's DAS card for ease of use under Windows. Axiom's driver includes a standard DAS API (Application program Interface) and some device-specific drivers. While programming, we only need to work with standard DAS API. A general view of AX5412H data acquisition board is shown in figure 3.19.



Figure 3.19: A picture of AX5412H Data Acquisition Card

Some of the AXIOM Data Acquisition Card features are:

- 32-bit DLL.
- Supported compilers: Visual C/C++, Borland C/C++, Visual Basic, Delphi
- Support AXIOM data acquisition hardware [1].
- Easy to use

- Implemented Functions:
 - Analog Input via software trigger
 - Analog Input via interrupt
 - Analog Input via DMA
 - > Digital Input
 - Digital Output
 - Temperature Measurement
 - Counter/Timer

The AX750 is a screw terminal panel with standoffs to allow tabletop operation. It provides easy connection, via 80 barrier strip connections, to all analog and digital I/O signals to the AXIOM DA&C series boards. The AX750 includes a one-meter long flat ribbon cable connector for direct connection to any AXIOM DA&C board. The AX750T, consists of AX750 and an additional thermocouple cold-junction compensation circuit, is designed to be used with AX5611-HG or AX5412-HG, for accepting various type of thermocouple input. The AX750 also provides features to accommodate the requirement for current loop input operation with a 0 to 20mA full-scale input range. By installing 250½ current shunt resistors, the user can convert the 0 to 20mA input range to a 0 to 5V range. Figure 3.20 shows the AX750 screw terminal panel.


Figure 3.20: AX750 screw terminal panel

The output from the servomotor plant i.e. the input voltage that is in analog signal from the servo amplifier will be converted to digital signal before the computer can process the data. The AX5412 board Analog to Digital Converter (A/D) with 12 bit AD resolution for a input range from 0V to 10V, 0.0025 Volt precision can be achieved, which it is good enough for the control purpose. There are 16 channels for single-ended & 8 for differential analogue input.

After the computer has calculated and processes the data, the controller signal has to be sent back to the servomotor plant. The data acquisition board with 2 channels of 12-bit Digital to Analog (D/A) output will convert the digital signal data to analog signal of the relevant voltage level. All the connection can be done easily with the AX750 screw panel connected to the output parallel port of the AX5412 boards.

3.4.1.2 Analog Input system

The basic function of analog input system (A/D system) is to convert the analog input signal to corresponding digital format that the computer can be read. In order to benefit A/D system's stability and obtain good performance, several other additional parts such as multiplexed, amplifiers, sample and hold are required. Figure 3.21 exhibits many input signals come into the A/D system.



Figure 3.21: Many input signals come into the A/D system

A/D converter has been developed for different application needs. The most widely used A/D converters are integrating and successive approximation. Integrating type has the advantage of high noise rejection but its speed is lower. Most data acquisition system needs at least 12-bit resolution to recognize the measured signal. Most high resolution A/D will elapse more time and normally the price is higher.

Multiplexed is a device containing a group of switched to select exact signal to go through A/D converter. Because the amplifier and A/D ate shared, the cost will be lower, but the channel's acquisition speed will be reduced too.

Sample and hole (S/H) is designed for A/D converter to keep track of input signals. When a high speed A/D converter is used, such as successive approximation, it requires a no changing input signal during the conversion stage. The S/H circuit can keep the input signal with a constant level until the A/D is completed. Often the limiting factor in the application of the S/H is the uncertainty in the time actual sample is taken – i.e. the "The aperture jitter" or Taj. The aperture jitter causes an amplitude uncertainty for any input where the voltage is changing. The approximation voltage error due to aperture jitter depends on the slew rate of the signal at the sample point.

Amplifier is a device to enlarge the input small signal to a congruous signal that A/D converter can resolvable. Notice if the input signal change is below the A/D system's minimum resolution, the system cannot recognize the deviation. The exact gain is usually better the A/D resolution.

3.4.1.3 Analog Output System

Analog output (digital control analog output D/A) are typically used to generate programmable level signal for yielding a loop control system Familiarly analog output range are 0 - 5V, 4 - 20 mA and so on. They can be employed to control such a dc power supply, frequency converter or to drive chart recorder, or as in the project the servomotor system itself. Normally analog out merely provides little power (voltage output is typically no more than 5mA), power amplifier or current boosters are required attachment when large load is used.

3.4.1.4 D/A Converter

A digital to analog (D/A) converter affords an analog output proportional to a digital number on the input. Most converters are 8 bits, 12 bits or even 16 bits, with correspondingly higher resolution. Cost rises rapidly with resolution.

3.4.2 MS150 Modular Servo System

Feedback Instrument Ltd. manufactured this set of experiment learning tools for the aim to help student to understand not only on position control system but also speed control of a servo motor system. The MS150 Modular Servo System contains few equipment such as the pre-amplifier unit (PA150C), Operational amplifier unit (OA150A), input potentiometer (IP150H), output potentiometer (OP150K), Attenuator unit (AU150B), DC motor (DCM150F), reduction gear tacho unit (GT150X), servo amplifier unit (SA150D) and power supply unit (PS150E). Each specification of the unit will be elaborate on the following session. A general view of MS150 Modular Servo System is shown in figure 3.22.



Figure 3.22: General View of MS150 Modular Servo System.

3.4.2.1 Pre-amplifier Unit – PA150C

The pre-amplifier provides two output signals to the servo amplifier. One output or the others goes positive to drive the motor in one direction or the other in response to positive or negative input. The schematic of PA150C is shown in figure 3.23.



Figure 3.23 PA150C- Pre-amplifier Unit

3.4.2.2 Operational Amplifier Unit – OA150A

This OA150A functions as to amplify the input signal with either unity feedback or external feedback determined by user. The schematic of OA150A is shown in figure 3.24.



Figure 3.24 : OA150A – Operational Amplifier Unit

3.4.2.3 Input Potentiometer – IP150H & Output Potentiometer – OP150K

The input and output potentiometer is used to measure displacement. These are precision servo mount type potentiometers fitted with calibrated positioning-indicating dials. A non-inverting buffer amplifier in each unit is designed to be dc-accurate and allow the potentiometers to be used practically as potential dividers. The schematic of IP150H and OP150K are shown in figure 3.25.



Figure 3.25: Input Potentiometer and Output Potentiometer (IP150H, OP150K)

3.4.2.4 Attenuator Unit – AU150B

The AU150B has two $100k\Omega$ potentiometer connected to the terminal sockets, and otherwise unconnected. The schematic of AU150B is shown in figure 3.26.



Figure 3.26 AU150B - Attenuator Unit

3.4.2.5 DC Motor -DCM150F

The DC motor is a permanent magnet type, which connects to the servoamplifier via a 12-way plug. It has a nominal maximum speed of 3000rotation/minute. Its torque at 2A is in the order of 0.1Nm and the rotor inertia is $3.9 * 10^{-5} \text{ kgM}^2$. The schematic of DCM150F is shown in figure 3.27.



Figure 3.27 DCM150F – DC Motor with Tachogenarator

3.4.2.6 Reduction Gear Tacho Unit – GT150X

This unit has a 30/1-reduction gear and a tachogenarator; it also provides a digital voltmeter and tachometer for display of motor speed directly in r/min. The schematic of GT150X is shown in figure 3.27.

3.4.2.6 Servo Amplifier Unit – SA150D

The servo amplifier is used to operate the DC motor (DCM150F). The schematic is shown in figure 3.28.



Figure 3.28:SA150D - Servo Amplifier Unit

3.4.2.7 Power Supply Unit – PS150E

This unit provides supply to all units required for MS150. The schematic is shown in figure 3.29.



Figure 3.29: PS150E – Power Supply Unit

3.4.3 Hardware Connection

Hardware connection is done by interfacing between Personal Computer and MS150 Modular Servo System through AX5412H Data acquisition card. The connection is shown in figure 3.30. There are 8 differential input (or 16 single input) and 2 output from the AX5412 data acquisition card. Only 2 differential input is used for measurement and both output are used to control the direction and the speed of the motor. Connection between MS150 Modular Servo System and AX5412H data acquisition card will be shown in the experiment procedure respectively.

Driver used in this project is the Axiom Device Manager version 1.1 where AX5412H is set at based address 300 with differential input and output range set at 10V bipolar.



Figure 3.30: Connection between PC and MS150 through AX750 Screw Terminal Panel

3.4.4 Classical Control System Experiment

The classical control consists of 4 experiments which are:

- 1) Closed Loop Position Control System.
- 2) Effect of Gain Changes on Deadband.
- 3) Unit Step Response to Gain Changes.
- 4) Velocity Feedback.

The closed loop position control system experiment is to demonstrate a simple motor-driven closed-loop position control system. From this experiment, student can understand the operating characteristic of the pre-amplifier and can observe the action of a simple motor driven closed loop position control system.

The effect of gain changes on deadband experiment is to demonstrate the term 'deadband'. From this experiment, student can observed the response of a position control system to a step input and also investigate the effect of gain on deadband and step response.

The effect of unit step response to gain changes experiment is to visual study of step response to gain changes upon position control system. From this experiment, student can understand the effect of gain changes to output response.

The effect of velocity feedback experiment is to demonstrate a simple closed loop position control system using velocity feedback. From this experiment, student can understand the operating characteristic of the pre-amplifier and can observe the action of a simple motor driven closed loop position control system.

CHAPTER 4

RESULTS & ANALYSIS

This software is mainly divided into 2 parts that is Classical Control and the artificial intelligence control using Direct Digital Control (DDC) technique. The Classical Control consists of PID controller while the artificial intelligence control consists of Fuzzy Logic control. Both the experiment will be discussed later in this chapter.

For Direct Digital Control (DDC), computer was used as controller to control the position of the servo system. A Proportional, Integral and Derivative (PID) Controller and Fuzzy Logic control was implemented into the software to control the position control system. Basically, this software was written for the purpose to enhance students understanding on some of the principle on position control system so that they are more familiar some of the practical applications based on Position Control System and how it works.

This software starts with a splash screen shown in figure 4.1. After 10 seconds, the PID controller GUI will appear. From this screen, user can also select which controller to use, either PID or the fuzzy logic controller. Each controller will be discussed one by one on the following session.



Figure 4.1: Splash screen

4.1 Direct Digital Control

The objective of this experiment is to perform visual study on the difference between direct digital control and classical control in controlling the position control system. Besides, it is also introduce computer control in replacing hardware components and function as digital controller in controlling the position control system. For this experiment, user is required to connect the circuit shown in figure 4.2.



Figure 4.2: Circuit diagram for direct digital control.

4.1.1 PID controller GUI

A PID controller was implemented using software programming to control the position control system response. This PID controller was based on the positional algorithm with the flow diagram of positional PID control algorithm shown in figure 3.9. This PID controller is adjustable which allow user to change the gain setting of PID respectively to observe the changes on the output response with different combination of PID controller gain such as undamped, underdamped, overdamped and critically damped response.

For this experiment, a total of four components were reduced compared to classical control. Hardware components that are reduced are the input potentiometer

(IP150H), Preamplifier unit (PA150C), Operational Amplifier Unit (OA150A) and the attenuator unit (AU150B). All these 4 units are replaced by personal computer. The PID controller GUI is shown in figure 4.3.

After connecting the circuit, user can press START button to start the controlling process and STOP button is pressed to stop the controller action. A digital oscilloscope was implemented where user can adjust its Volt per division setting, time per division setting and output waveform can also be pause to stop its scrolling waveform. All the data can be recorded into the list box shown in figure 4.3.



Figure 4.3: PID controller GUI

4.1.2 Fuzzy Logic Controller GUI

A fuzzy logic controller was implemented using software programming to control the position control system response. This fuzzy logic controller was based on the rule-base that has been discussed in chapter 3. The fuzzy logic control that was design is fix (can't manipulate the output from the GUI). Thus to improve the system response, we have to redesign the input and out put variable by adding more fuzzy sets, or by changing the fuzzy rule base, inference technique or even the defuzzification technique.

For this experiment, a total of four components were also reduced compared to classical control. Hardware components that are reduced are the input potentiometer (IP150H), Preamplifier unit (PA150C), Operational Amplifier Unit (OA150A) and the attenuator unit (AU150B). All these 4 units are replaced by personal computer. The fuzzy logic controller GUI is shown in figure 4.4.

After connecting the circuit, user can press START button to start the controlling process and STOP button is pressed to stop the controller action. A digital oscilloscope was implemented where user can adjust its Volt per division setting, time per division setting and output waveform can also be pause to stop its scrolling waveform. All the data can be recorded into the list box shown in figure 4.5.



Figure 4.4: Fuzzy Logic Controller GUI



Figure 4.5: Fuzzy Logic Controller data capture form

4.2 Preliminary Result Analysis

For the first part of the project, results are obtained manually from the MS150 Modular Servo System by performing the classical control experiments. Output responses are being recorded using Yokogawa Digital Oscilloscope model DL1546. The waveform is then analyzed manually to obtain its peak time (T_p), peak-to-peak voltage (V_{p-p}), maximum voltage (V_{max}), rise time (T_r) and percent overshoot (%OS).

The following terms are defined to ease analysis data:

i) Rise time, T_r

Define as the time taken for response from 10% final value to the 90% of final value. It shows how fast a system response to input.

ii) Peak time, T_p

Define as the time taken for the response to rise from 0 to the first peak value or the maximum value that the system can achieve.

iii) Overshoot, M_p

Define as the maximum amount by which response overshoots the steady state value or final value.

iv) Settling time, T_s

Define as the time taken for the response to fall within and remain within some specified percentage e.g. 2% of the steady state value. It used as a measure of the time taken for the oscillation to die away.

4.2.1 Closed Loop Position Control System

The closed loop position control system experiment is to demonstrate a simple motor-driven closed-loop position control system. From this experiment, student can understand the operating characteristic of the pre-amplifier and can observe the action of a simple motor driven closed loop position control system. Figure 4.3 shows the output result of the experiment.

The gain is measured on the straight part of the curve because this is the part that will be used in practice. From figure 4.6, we can observe that the output of the position sensor varies proportionally to the input (indicated angle).

$x_i :=$	$y_i :=$
-150	-8.42
-140	-8.03
-130	-7.54
-120	-6.69
-110	-5.94
-100	-5.31
-90	-4.98
-80	-4.17
-70	-3.75
-60	-3.46
-50	-2.99
-40	-2.41
-30	-1.8
-20	-1.21
-10	-0.73
0	-0.03
10	0.64
20	1.27
30	1.64
40	2.58
50	3.06
60	3.33
70	3.82
80	4.71
90	4.95
100	5.37
110	5.89
120	6.29
130	6.65
140	7.80
150	8.30

where;

```
x_i = angle
```

 $y_i = voltage$

$mean(\mathbf{x}) = 0$	mean(y) = -0.037
stdev (x) = 89.443	stdev $(y) = 4.947$
$var(x) = 8 \times 10^3$	var(y) = 24.477
m := slope(x, y)	m = 0.055
c := intercept(x,y)	c = -0.037
cc := corr(x, y)	cc = 0.999



Figure 4.6: The output result of closed loop position control system experiment.

The graph is almost a straight line with a correlation coefficient = 0.999. From figure 4.6, we can observe that the graph intercept the y-axis at -0.037 Volts. We also can conclude that the angle sensor constant;

$$K_{\theta} = m(slope) \\ = 0.055V/^{0}.$$

4.2.2 Effect of Gain Changes on Deadband

The effect of gain changes on deadband experiment is to demonstrate the term 'deadband'. From this experiment, student can observed the response of a position control system to a step input and also investigate the effect of gain on deadband and step response. Figure 4.7 shows the output result of the experiment.



where;

 $x_i = Proportional gain$ $y_i = Voltage$

mean(x) = 5.5	mean(y) = 16.5
stdev (x) = 2.872	stdev $(y) = 9.047$
var(x) = 8.25	var(y) = 81.85
m := slope(x, y)	m = -2.939
c := intercept(x, y)	c = 32.667
cc := corr(x, y)	cc = -0.933



Figure 4.7: The output result of gain changes on deadband

The gain is measured on the straight part of the curve because this is the part that will be used in practice. From figure 4.7, we can observe that the relationship between gain and deadband is that the deadband error reduces as gain is increased.

The graph is almost a straight line with a correlation coefficient = -0.933. From figure 4.7, we can observe that the graph intercept the y-axis at 32.667° with a slope of m = -2.939.

4.2.3 Unit Step Response to Gain Changes

The effect of unit step response to gain changes experiment is to visual study of step response to gain changes upon position control system. From this experiment, student can understand the effect of gain changes to output response. Table 4.1 shows the output result of the experiment. From table 4.1 we can observe that in a position control system, that while increasing the gain has the advantage of reducing deadband, it does also increases the problem of overshoot. We can see that, eventually the system become unstable, thus it continuously oscillate about the required position.

Кр	System	Output waveform	
	Characteristic		
1	Vp-p = 5.840 V Vmax = 5.040 V %OS = 11.9% T _r = 180m s	Stopped 2004/09/08 19:14:28 CH1=2V DC 1:1 NORM:1kS/s DC 1:1 DC 1:1 NORM:1kS/s 100 90	
		=Filter= =Offset= =Record Length= =Trigger= Smoothing : OFF CH1 : 0.00V Main : 10K Mode : AUTO BW : FULL CH2 : 0.00V Zoom : 5K Type : EDGE CH1 CH3 : 0.00V Delay : 0 CH4 : 0.0V Hold Off : MINI	

Table 4.1: Output result of the experiment unit step response to gain changes.



Table 4.1 (continue)



Table 4.1 (continue)

4.2.4 Velocity Feedback

The effect of velocity feedback experiment is to demonstrate a simple closed loop position control system using velocity feedback. From this experiment, student can understand the operating characteristic of the pre-amplifier and can observe the action of a simple motor driven closed loop position control system. Table 4.2 shows the output result of the experiment.

We can observe that in a position control system, that while increasing the gain has the advantage of reducing deadband, it does also increases the problem of overshoot. We can see that, eventually the system become unstable, thus it continuously oscillate about the required position.

The problem of overshoot arises due to the motor's speed carrying it past the point of alignment, the higher the gain the greater the speed of motor response so that above a certain gain the system can never align. Thus to overcome the overshoot, we use velocity feedback from the tachogenarator.

Gain	System	Output waveform
	Characteristic	
$\frac{K_v = 1}{K_p = 6}$	Characteristic Vp-p = 6.08 V Vmax = 4.72 V %OS = 18.6% T _r = 160m s	Stopped 2004/09/07 00:28:14 CH1=2V CH3=2V ↔500ms/div DC 1:1 NORM:1kS/s 100 90
		=Filter= =Offset= =Record Length= =Trigger= Smoothing : OFF CH1 : 0.00V Main : 10K Mode : AUTO BW : FULL CH2 : 0.00V Zoom : 5K Type : EDGE CH1 _∱ CH3 : 0.00V Delay : 0.0ns CH4 : 0.0V Hold Off : MINIMUM

Table 4.2: Output result of the experiment velocity feedback

K = 2	$V_{n} n = 4.99 V$	Stopped		2004/09/07 00:46:31
$\mathbf{K}_{v} \equiv \mathbf{S}$	vp-p=4.88 v	CH1=2V	CH3=2V	↔500ms/div
$\mathbf{K}_{\mathrm{p}} = 6$	Vmax = 4.88 V	DC 1:1	DC 1:1	NORM;1kS/s
	% OS = 0%	100		
	$T_r = 1.1 s$	90	1	
	11 - 1.1 3	=Filter= = Offset= Smoothing: OFF CH1: 0.00V BW : FULL CH2: 0.00V CH3: 0.00V CH4: 0.0V	Max 2.560V -07 1.7% -007 2.320V Prod 2.1000 Max 3.920V -07 1.7% Log -2.320V Prod 2.1005 -07 1.7% Log -880.0mV Prod 2.1105 	Min -2.400v +0vr 3.4% RTSE 5.000ms +Wd 1.050s Min -960.0mV +0vr 3.4% Rise 180.0ms +Wd 1.100s =Trigger= Mode : AUTO Type : EDGE CH1 5 Delay : 0.0ns Hold Off : MINIMUM
V (VI 40VI	Stopped		2004/09/07 20:08:14
$K_v = 0$	vp-p = 4.8 v	CH1=2V	CH3=2V	↔500ms/div
$K_p = 6$	Vmax = 4.8 V	DC 1:1	DC 1:1	NORM;1kS/s
	%OS = 0%	100		
	T _r = 300m s	90	1 /	
	Width = 1.090 s			
			_	
		=Trace1= P-P 4.800V Rms 2.356V	Max 2.320V -0yr 1.7%	Min -2.480V +9vr 1.7%
		High 2.240V High Freq 476.2mHz	Prod 2,100s	Ri pe -5√899mo - +₩d 1.850s
		=Trace3= P-P 4.800V	Max 3.760V	Min -1.040V
		runs 2.2634 0%·····High··3.6864····	-0wr 3.9%	Rise 300.0ms
		Freq 476.2mHz	Priod 2.100s	+Wd 1.090s
		=Filter= =Offset= Smoothing : OFF CH1 : 0.00V BW : FULL CH2 : 0.00V CH3 : 0.00V CH4 : 0.0V	=Record Length= Main : 10K Zoom : 5K	=Trigger= Mode : AUTO Type : EDGE CH1 Delay : 0.0ns Hold Off : MINIMUM

Table 4.2 (continue)



Table 4.2 (continue)

By observing the output response of the system via the gain changes in the Proportional controller gain via velocity feedback respectively, we can conclude that for the P controller gain, the output of the controller is proportional to error, the proportionality given by the controller gain K_P determines how much the output from the controller changes for a given change in error, thus K_P establishes the sensitivity of the controller to an error, how much the controller output changes per unit error. While, velocity feedback is introduced into the system to overcome the overshoot, which was used from the tachogenarator, but excessive velocity feedback gives a slow response to the system.

Proportional controller offers the advantage of having only one adjustable or tuning parameter, K_P . However they suffer a major disadvantage that is the controlled variable is operated with an offset. This offset can be described as a steady state deviation of the controlled variable from set point or simply a steady state error. While, velocity feedback is introduce into the system to overcome the overshoot, which was used from the tachogenarator.

By increasing the proportional gain, it will create smaller offset. But, it makes process becomes more oscillatory. Thus to remove the offset, integral action is implemented where it function to integrate the error.

Thus to make the system to have less overshoot and a fast T_r , a PID controller will be implemented. This is the objective of the project, which is to develop a PID controller that will be implemented in a DDC control system.

A PID controller has 3 terms, K_P , K_I , and KD, which must be adjusted or tuned to obtain satisfactory control. The derivative action gives the controller the capabilities to anticipate where the process is heading. The derivative action reduces overshoot and decrease oscillation around set point but PID controller is recommended for slow process, which is usually free of noise. Fast processes (processes with slow time constant) are easily susceptible to process noise. The application of derivative mode will only result in amplification of the noise, because the derivative of the fast changing noise is a large value.

4.3 Result Statement

After the GUI is developed, experiment will be performed via software, data and output waveform will then be collected directly from the digital oscilloscope from the software. This software is mainly divided into two Direct Digital Control (DDC) experiments, that is:

- 1) PID Controller
- 2) Fuzzy Logic Controller

4.3.1 PID Controller Result Analysis

The waveform obtained through the direct digital control experiment is being analyzed. The terms defined above are being used for each combination of PID controller. Figure 4.8 shows the result which was capture within the GUI. All analysis on collected data is shown in appendices A. Table 4.3 shows the analysis of the best output result which was capture within the GUI. Here, should note that all the system characteristic term is analyze using Yokogawa DL1546 Digital Oscilloscope to analyze. All values are written in two decimal place. The unit used for rise time, T_r, settling time, T_s, peak time, T_p is milliseconds (ms), %OS is in percentage (%), ζ is dimensionless, ω_n is rad/s and unit for steady state error, e_{ss} is in Volt.



Figure 4.8: PID result which was capture within the GUI

Gain (%)	System Characteristic	Output Waveform
P=10	$T_r = 415.38$	
I=100	$T_p = 476.92$	
D=50	$T_s = 538.46$	
	%OS = 0	
	$\zeta = 0.66$	
	$\omega_n \!= 11.27$	
	$e_{ss} = 0.10$	

Table 4.3: The analysis of PID controller output result.

4.3.2 Fuzzy Logic Controller Result Analysis

The waveform obtained through the direct digital control experiment is being analyzed. The terms defined above are being used for the fuzzy logic controller. Figure 4.9 shows the result which was capture within the GUI. From the capture result, it can be concluded that the output response of the fuzzy logic controller has a percentage overshoot of 0.8V that is 16% overshoot.



Figure 4.9: The fuzzy logic result which was capture within the GUI.

4.4 Comparison between PID and Fuzzy Logic

After designing the software for both the PID and Fuzzy logic controller, we can conclude that:

PID controller:

- Is easy to tune the parameter or P, I and D.
- The PID parameter can easily be tune with the slide and is more user friendly

Fuzzy Logic:

- The fuzzy logic controller can't be easily be tune as the PID. If the output is not satisfactory, thus need to add more fuzzy sets into the input and output variable and define a new rule-base.
- trial and error method approach will be used until satisfactory result obtained
- The simulation output response was satisfactory. But when connected to the MS150 servo motor, the results are not satisfactory.
- It is time consuming

The output response of both the PID and Fuzzy logic is being compared. Table 4.4 shows the comparison between the output response using the PID and Fuzzy logic. From the analysis and comparison of both controllers, the PID controller performs better compared to Fuzzy logic in terms of percentage overshoot, with 0% overshoot compared to fuzzy logic with an overshoot of 16%. But, when comparing both of the time rise and time settling of both controllers, the fuzzy logic controller performs better, with a T_r of 400.00 ms and T_s of 100 ms.

	PID	FLC
	(best response)	
Tr	415.38ms	400.00ms
Ts	538.46ms	100ms
%Os	0%	16.0%

 Table 4.4: The comparison output between response using the PID and Fuzzy logic.

4.5 Discussion

Proportional controller offers the advantage of having only one adjustable or tuning parameter, K_P . However they suffer a major disadvantage that is the controlled variable is operated with an offset. This offset can be described as a steady state deviation of the controlled variable from set point or simply a steady state error. While, velocity feedback is introduce into the system to overcome the overshoot, which was used from the tachogenarator.

By increasing the proportional gain, it will create smaller offset. But, it makes process becomes more oscillatory. Thus to remove the offset, integral action is implemented where it function to integrate the error.
Thus to make the system to have less overshoot and a fast T_r , a PID controller was implemented. This is the objective of the project, which is to develop a PID controller that will be implemented in a DDC control system.

A PID controller has 3 terms, K_P , K_L , and K_D , which must be adjusted or tuned to obtain satisfactory control. The derivative action gives the controller the capabilities to anticipate where the process is heading. The derivative action reduces overshoot and decrease oscillation around set point but PID controller is recommended for slow process, which is usually free of noise. Fast processes (processes with slow time constant) are easily susceptible to process noise. The application of derivative mode will only result in amplification of the noise, because the derivative of the fast changing noise is a large value.

For the fuzzy logic controller, the controller can't be easily being tune as the PID. If the output is not satisfactory, thus need to add more fuzzy sets into the input and output variable and define a new rule-base. Trial and error method approach will be used until satisfactory result obtained. The simulation output response was satisfactory. But when connected to the MS150 servo motor, the results are not satisfactory. The designing of the fuzzy logic controller is also very time consuming.

CHAPTER 5

CONCLUSION

Virtual instrumentation or Graphical User Interface (GUI) can be defined as an intuitive, easy-to-use method of interfacing with computer program by means of graphical screen display. It can resemble the front panel of instrumentations or other objects associated with a computer program. The GUI design will reflects the user's mental model rather than the implementation model. Program features and functions support only required user goals. No superfluous features have been added simply because they are easy to implement, or as a result of the personal biases of the programmers. The GUI design reflects the expected characteristics and capabilities of the user population - visual, physical and cognitive capabilities; cultural and technical background; domain experience; education level and so forth.

In designing digital controller in controlling the position control system, each theory and concept play an important part in understanding the operation of position control system. In contrast, understanding on the concept of Direct Digital Control (DDC), Data Acquisition System (DAS), Proportional, Integral and Derivative (PID) controller and fuzzy logic controller also play an important role in succeeding this project. The software designing is tough to be done especially in designing the coding for controllers. It takes a long time and great patient to make sure the software design is in good progress. There are many situations that require some type of servo-control system. PID controllers are commonly used to regulate the time-domain behavior of many different types of dynamic plants. These controllers are extremely popular because they can usually provide good closed loop response characteristics, can be tuned using relatively simple rules and are easy to construct using either analogue or digital components. To discretise the PID controller, we need to approximate the integral and the derivative terms to forms suitable for computation by a computer. By using z-transform, we may discretise the analogue PID controller to obtain the pulse transfer function for the digital PID controller. Then, equation u (k) was used to develop the coding program for PID controller which corresponding to sequence operation.

To design the fuzzy logic controller, the position control of the servo system configuration was designed based on 2 inputs and 1 output. Inputs for this controller are error and rate (de), and the motor output. Fuzzification is where we define the quantization and membership function for input variable, error and rate (de) and output variable, motor output in universe of discourse. It involves the conversion of the input and output signals into a number of fuzzy represented values (fuzzy sets). The knowledge base of a fuzzy logic controller consists of a data base and a rule base. The basic function of the data base is to provide the necessary information for the proper functioning of the fuzzification module, the rule base and the defuzzification module. The basic function of the rule base is to represent the expert knowledge in a form of if-then rule structure. The fuzzy logic was derived into a 3 x 3 rules which consist of 9 rules. For this system, max-min composition is used for the Inferencing to obtain the fuzzy set describing the fuzzy value of the overall control output. Defuzzification is a mapping from a space of fuzzy control actions defined over an output universe of discourse into a space of non-fuzzy (crisp) control action. For this system, centroid method is used for defuzzification.

5.1 Suggestion for future work

Since the software being developed is still remains as a trial version, suggestions on some of the features to be implemented for future work which can be implemented are:

- 1) Design a new controller which can improve the output response of the position of servo motor.
- 2) Design the software using Labview, C++ or any suitable programming language.
- 3) Design a virtual instrumentation using National Instrument DAQ card to interface with the MS150 Modular Servomotor system. By using NI DAQ card, student can design the GUI of the controllers using Labview which is much simpler than designing controller with Visual Basic.
- 4) Design the fuzzy logic controller such that user can change the rule-base immediately in the GUI, rather than have to rewrite a new coding.

REFERENCES

- [1] Axiom AX5412-H Data Acquisition Board User Manual
- [2] Constantine H. Houpis & Gary B. Lamout, (1985), Digital Control System Theory, Hardware, Software, Mc. Graw Hill Book
- [3] David M. Auslander, (1990), Real-time Software for Control, Prentice Hall
- [4] George A. Perdikaris, (1991), Computer controlled Systems, Theory and Application, Kluwer Academic Publisher
- [5] Jee Ming Lee and Jung-Tang Huang; (1995), Motor Drive Control System For Education and Research
- [6] Katsuhiko Ogata (1995), Discrete Time Control Systems 2nd edition, Prentice Hall.
- [7] K.K.Tan, T.H. Lee & F.M.Leu; (1999), Development of a Distant Laboratory using Labview
- [8] Kelvin R. Aaron, Noreen L. Foster, Danielle P.Hazel, A.M.Hasanul Basher;(2002), Closed-loop Position Control System using Labview
- [9] Michael F. Hordeski, (1992), Control System Interfaces Design and Implementation Using PC, Prentice Hall

- [10] M.Moallen; (2000), Real-Time Position Control of Free-Electron Laser Beams: Theory and Experiments
- [11] MS150 Modular Servo System Experiment Book 1-3
- [12] Norman S. Nise (1995) Control System Engineering, Addison Westley Publishing
- [13] Rahman Jamal (2001), Labview Application and solutions, National Instruments Visual Instrumentation Series
- [14] Stuart Bennert, (1995) Real-time Computer Control An Introduction, Prentice Hall
- [15] Timothy J. Ross, (1997), Fuzzy Logic with engineering applications, McGraw-Hill International Edition.
- [16] Hugh F. VanLandingHam, (1985), Introduction to digital control system, Macmillan Publishing Company.
- [17] Deitel & Deitel T. R. Nieto, (1999), Visual Basic 6: How to Program, Prentice Hall.
- [18] Eric A., Valor Whisler and Hank Marquis, (1998), Visual Basic 6 Bible, Hungry Minds, Inc.

APPENDIX A

PID RESULTS

Gain	System	Output Waveform
(%)	Characteristic	Output Waveform
P=10	$T_r = 366.25$	
I=10	$T_p = 450.33$	
D=10	$T_{s} = 840.24$	
	%OS = 9.09	
	$\zeta = 0.61$	
	$\omega_n \!= 7.85$	
	$e_{ss}\!=\!0.47$	
P=10	$T_r = 390.44$	
I=10	$T_p\!=\!0.00$	
D=50	$T_s = 390.44$	
	%OS = 0	
	$\zeta > 1$	
	$\omega_n = 10.24$	
	$e_{ss} = 0.35$	
P=10	$T_r = 540.05$	
I=10	$T_{p} = 0.00$	
D=100	$T_s = 540.05$	
	%OS = 0	
	$\zeta > 1$	
	$\omega_n = 7.41$	
	$e_{ss} = 0.05$	
P=10	$T_r = 375.25$	
I=50	$T_p = 471.33$	
D=10	$T_s = 841.22$	
	%OS = 12.85	
	$\zeta = 0.55$	
	$\omega_n\!=8.70$	
	$e_{ss} = 0.29$	

P=10 I=50 D=50	$T_{r} = 405.22$ $T_{p} = 0.00$ $T_{s} = 405.22$ % OS = 0 $\zeta > 1$ $\omega_{n} = 9.87$ $e_{ss} = 0.12$	
P=10	$T_r = 585.74$	
I=50	$T_{p} = 0.00$	
D=100	$T_s = 585.74$	
	%OS = 0	
	$\zeta > 1$	
	$\omega_n\!=6.23$	
	$e_{ss} = 0.31$	
P=10	$T_r = 353.85$	
I=100	$T_p = 470.77$	
D=10	$T_s = 892.31$	
	%OS = 13.04	
	$\zeta = 0.54$	
	$\omega_n = 8.24$	
	$e_{ss} = 0.30$	
P=10	$T_r = 415.38$	
I=100	$T_p = 476.92$	
D=50	$T_s = 538.46$	
	%OS = 0	
	$\zeta = 0.66$	
	$\omega_n = 11.27$	
	$e_{ss} = 0.10$	

P=10	$T_r = 498.46$	
I=100	$T_p = 0$	
D=100	$T_s = 498.46$	
	% OS = 0	
	$\zeta > 1$	
	$\omega_n\!=8.02$	
	$e_{ss}\!=\!0.16$	
P=50	$T_r = 384.62$	
I=10	$T_p = 467.69$	
D=10	$T_s = 1046.15$	
	% OS = 18.04	
	$\zeta = 0.48$	
	$\omega_n\!=7.99$	
	$e_{ss} = 0.15$	
P=50	$T_r = 384.62$	
I=10	$T_p = 461.54$	
D=50	$T_{s} = 895.38$	
	%OS = 3.04	
	$\zeta = 0.67$	
	$\omega_n = 6.64$	
	$e_{ss} = 0.13$	
P=50	$T_r = 488.64$	
I=10	$T_p = 0$	
D=100	$T_s = 488.64$	
	%OS = 0	
	$\zeta > 1$	
1		
	$\omega_n = 8.19$	

P=50 I=50 D=10	$T_r = 371.12$ $T_p = 479.37$ $T_s = 1020.58$ % OS = 21.54 $\zeta = 0.44$ $\omega_n = 8.93$ $e_{ss} = 0.08$	
P=50 I=50 D=50	$T_{r} = 340.19$ $T_{p} = 417.51$ $T_{s} = 618.54$ $\% OS = 0$ $\zeta = 0.57$ $\omega_{n} = 11.25$ $e_{ss} = 0.23$	
P=50 I=50 D=100	$T_{r} = 541.22$ $T_{p} = 0$ $T_{s} = 541.22$ % OS = 0 $\zeta > 1$ $\omega_{n} = 7.39$ $e_{ss} = 0.13$	
P=50 I=100 D=10	$T_r = 371.12$ $T_p = 479.37$ $T_s = 1051.51$ % OS = 16.67 $\zeta = 0.50$ $\omega_n = 7.68$ $e_{ss} = 0.14$	

P=50 I=100 D=50	$T_{r} = 371.12$ $T_{p} = 448.44$ $T_{s} = 757.71$ % OS = 9.66 $\zeta = 0.60$ $\omega_{n} = 8.84$	
	$e_{ss} = 0.14$	
P=50 I=100 D=100	$T_r = 366.48$ $T_p = 562.87$ $T_s = 634.22$ % OS = 3.44 $\zeta = 0.73$ $\omega_n = 8.62$ $e_{ss} = 0.10$	
P=100 I=10 D=10	$T_r = 346.38$ $T_p = 456.17$ $T_s = 1623.66$ % OS = 18.21 $\zeta = 0.48$ $\omega_n = 5.13$ $e_{ss} = 0.21$	
P=100 I=10 D=50	$T_r = 268.59$ $T_p = 467.20$ $T_s = 1237.07$ % OS = 9.09 $\zeta = 0.61$ $\omega_n = 5.33$ $e_{ss} = 0.14$	

P=100 I=10 D=100	$T_r = 395.86$ $T_p = 432.98$ $T_s = 541.23$ % OS = 1.52 $\zeta = 0.80$ $\omega_n = 9.24$ $e_{ss} = 0.17$	
P=100	$T_r = 386.59$	
I=50	$T_p = 501.01$	
D=10	$T_s = 1577.27$	
	%OS = 16.08	
	$\zeta = 0.50$	
	$\omega_n = 5.04$	
	$e_{ss} = 0.15$	
P=100	$T_r = 195.86$	
I=50	$T_p = 463.90$	
D=50	$T_{s} = 748.43$	
	% OS = 10.02	
	$\zeta = 0.59$	
	$\omega_n = 9.05$	
	$e_{ss} = 0.12$	
P=100	$T_r = 346.38$	
I=50	$T_p = 426.79$	
D=100	$T_s = 556.68$	
	%0S = 5.04	
	$\zeta = 0.55$	
	$\omega_n = 12.98$	
	0.10	

P=100	$T_r = 389.68$	
I=100	$T_p = 463.9$	
D=10	$T_s = 1212.33$	
	%OS = 17.88	
	$\zeta = 0.48$	
	$\omega_n = 6.87$	
	$e_{ss} = 0.12$	
P=100	$T_r = 386.59$	
I=100	$T_p = 432.98$	
D=50	$T_s = 488.64$	
	%OS = 3.04	
	$\zeta = 0.74$	
	$\omega_n = 11.01$	
	$e_{ss} = 0.11$	
P=100	$T_r = 383.52$	
I=100	$T_p = 430.21$	
D=100	$T_s = 486.81$	
	%OS = 1.45	
	$\zeta = 0.75$	
	$\omega_n = 11.02$	
	$e_{ss} = 0.11$	

APPENDIX B

SOFTWARE SNAPSHOT



Splash screen



Circuit diagram for direct digital control.



PID controller GUI



Fuzzy Logic Controller GUI



Fuzzy Logic Controller data capture form

APPENDIX C

SLIDE PRESENTATION

Virtual Instrumentation Of Position Control System Using Direct Digital Control (DDC) Techniques

Presented by: Mariam binti Md Ghazaly Supervised by: PM Dr. Mohd Fua'ad Rahmat (Head Department Off Campus Program)











		_	-													-	
	Progress		July			August					September			October			
1	Getting Project Title																
2	Determining objectives																T
3	Gathering information and references																
4	Literature Review																
5	Learning Visual Basic																
6	Preparing and Handling Proposal																
7	Performing Experiment																
8	Familiarize with Digital Oscilloscope																
9	Identify the equipment characteristics																
10	Installing AX5412-H Driver																
11	Studies on AX5412-H User Manual																
12	Programming Simple Software																
13	Test-run and debugging																
14	Preparing Presentation																
15	Preparing Report																
16	Presentation																
17	Handling in report																




































































Closed Loop Response	Rise Time, Tr	Percent Overshoot, %OS	Settling Time, Ts	Steady State Error ess		
Proportional Gain, K⊧	Decrease	Increase	Small Change	Decrease		
Integral Gain, K	Decrease	Increase	Increase	Eliminate		
Derivative Gain, K₀	Small Change	Decrease	Decrease	Small Change		







· For	m2						-								
				F	uzzy	Log	ic Pa	aram	eters	5			Main For	•	
Na	Input	OutputM	e	de	Ne	Ze	Pe	Nde	Zde	Pde	No	Zo	Po	Dv	V.
37	0.000	0.000	0.000	3.355	0.000	1.000	0.000	0.000	0.966	0.134	0.000	0.000	0.000	0.000	5.00
39	0.000	0.000	0.000	4 107	0.000	1.000	0.000	0.000	0.835	0.164	0.000	0.000	0.000	0.000	5.00
40	0.000	0.000	0.000	4.962	0.000	1.000	0.000	0.000	0.002	0.199	0.000	0.000	0.000	0.000	5.00
41	5.000	0.000	5.000	14.962	0.000	0.000	0.200	0.000	0.402	0.598	0.000	0.000	0.000	2.855	5.00
42	5,000	4,381	0.140	9,726	0.000	0.9394	0.000	0.000	0.613	0.389	0.000	0.000	0.000	0.107	5.00
44	5.000	4.896	0.104	8.459	0.000	0.996	0.004	0.000	0.662	0.338	0.000	0.000	0.000	0.018	5.00
45	5.000	4.946	0.054	7.554	0.000	0.998	0.002	0.000	0.698	0.302	0.000	0.000	0.000	0.009	5.00
45	5 000	4,904	0.016	6.661	0.000	1,000	0.000	0.000	0.754	0.236	0.000	0.000	0.000	0.003	5.00
48	5.000	4.938	0.002	5.337	0.000	1.000	0.000	0.000	0.787	0.213	0.000	0.000	0.000	0.000	5.00
49	5.000	4.999	0.001	5.039	0.000	1.000	0.000	0.000	0.798	0.202	0.000	0.000	0.000	0.000	5.00
50	5.000	5.000	0.000	5.038	0.000	1.000	0.000	0.000	0.798	0.202	0.000	0.000	0.000	0.000	5.00
52	5 000	5,000	0.000	5,893	0.000	1.000	0.000	0.000	0.787	0.213	0.000	0.000	0.000	0.000	5.00
53	5.000	5.000	0.000	6.645	0.000	1.000	0.000	0.000	0.734	0.266	0.000	0.000	0.000	0.000	5.00
54	5.000	5.000	0.000	7.500	0.000	1.000	0.000	0.000	0.700	0.300	0.000	0.000	0.000	0.000	5.00
55	5.000	5.000	6.000	8.355	0.000	1.000	0.000	0.000	0.666	0.334	0.000	0.000	0.000	0.000	5.00
57	5.000	3.721	1.279	1.614	0.051	0.949	0.000	0.065	0.935	0.000	0.000	0.000	0.000	0.187	5.00
58	-5.000	-4.568	-0.432	-0.470	0.017	0.983	0.000	0.019	0.981	0.000	0.000	0.000	0.000	-0.064	5.00
59	-5.000	-4.955	-0.145	-0.183	0.006	0.994	0.000	0.007	0.993	0.000	0.000	0.000	0.000	-0.022	5.00
61	-5.000	4.302	-0.050	-0.300	0.002	0.390	0.000	0.015	0.305	0.000	0.000	0.000	0.000	-0.007	5.00
62	-5.000	4.994	-0.006	1.650	0.000	1.000	0.000	0.066	0.934	0.000	0.000	0.000	0.000	-0.001	5.00
63	5.000	4.998	-0.002	2.502	0.000	1.000	0.000	0.100	0.900	0.000	0.000	0.000	0.000	0.000	5.00
64	5.000	4.999	-0.001	-3.356	0.000	1.000	0.000	0.134	0.836	0.000	0.000	0.000	0.000	0.000	5.00
GC	-5.000	-5.000	0.000	-4.665	0.000	1.000	0.000	0.187	0.013	0.000	0.000	0.000	0.000	0.000	5.00
67	-5.000	-5.000	0.000	-4.962	0.000	1.000	0.000	0.198	0.802	0.000	0.000	0.000	0.000	0.000	5.00
68	5.000	5.000	0.000	4.962	0.000	1.000	0.000	0.198	0.802	0.000	0.000	0.000	0.000	0.000	5.00
20	5,000	0.000	5,000	10,893	0.000	0.800	0.200	0.187	0.813	0.436	0.000	0.000	0.000	0.000	5.00
71	5.000	3,779	1.221	7.866	0.000	0.951	0.049	0.000	0.685	0.315	0.000	0.000	0.000	0.196	5.00 -
1.01															









Cond	clusion	(cont.)		
		PID	FLC	
	Tr	415.38ms	400.00ms	
	Ts	538.46ms	100ms	
	%Os	0%	16.0%	
	· · · · ·			
				51





