Fuzzy Logic Based Temperature Controller

Major Project Report

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology

In

Electronics & Communication Engineering

(Communication Engineering)

By

Parth Sharma

(14MECC23)

Electronics & Communication Engineering

Department of Electrical Engineering

Institute of Technology

Nirma University

Ahmedabad-382 481

May 2016

Fuzzy Logic Based Temperature Controller

Major Project Report

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology

In

Electronics & Communication Engineering

(Communication Engineering)

By

Parth Sharma

(14MECC23)

Under the Guidance of

Prof. Manisha Upadhyay

Electronics & Communication Engineering

Department of Electrical Engineering

Institute of Technology

Nirma University

Ahmedabad-382 481

May 2016

Declaration

This is to certify that

i) The thesis comprises my original work towards the degree of Master of Technology in Communication Engineering at Nirma University and has not been submitted elsewhere for a degree.

ii) Due acknowledgement has been made in the text to all other material used.

Parth Sharma

NIRMA
UNIVERSITY
INSTITUTE OF TECHNOLOGY
NAAC ACCREDITED ‘A’ GRADE

Certificate

This is to certify that the Major Project entitled **”Fuzzy Logic Based Temperature Controller”** submitted by **Parth Sharma (14MECC23)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in Communication Engineering at Nirma University, Ahmedabad is the record of work carried out by his under our supervision and guidance. In our opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of our knowledge, haven’t been submitted to any other university or institution for award of any degree or diploma.

Prof. Manisha Upadhyay
Associate Professor
Electronics & Communication
Institute of Technology
Nirma University, Ahmedabad

Dr. D K Kothari
Program Coordinator
Electronics & Communication
Institute of Technology
Nirma University, Ahmedabad

Dr. P. N. Tekwani
HOD(EE) & Director
Institute of Technology
Nirma University, Ahmedabad

Certificate

This is to certify that the Major Project entitled **"Fuzzy Logic Based Temperature Controller"** submitted by **Parth Sharma (14MECC23)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in Communication Engineering at Nirma University, Ahmedabad is the record of work carried out by his under our supervision and guidance. In our opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of our knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Shri. Mahesh Kumar
Scientist Engineer-SD
DCD/ODCG/SNPA
Space Applications Centre
ISRO, Ahmedabad

Shri. TVS Ram
GD - ODCG
DCD/ODCG/SNPA
Space Applications Centre
ISRO, Ahmedabad

Acknowledgements

I would like to express my gratitude and sincere thanks to Dr P N Tekwani, Head of Electrical Engineering Department and Dr. D. K. Kothari, Coordinator M.Tech Communication Engineering program for allowing me to undertake this thesis work and for his guidelines during the review process.

I am deeply indebted to my internal guide Prof. Manisha Upadhyay for her constant guidance and motivation. She has devoted significant amount of her valuable time to plan and discuss the thesis work. Without her experience and insights, it would have been very difficult to do quality work.

I wish to thank to my external guide Shri. Mahesh Kumar, Scientist/Engineer, without whose insight and help, this project would not have been completed. I thank Smt. Durga Digdarsini, Scientist/Engineer, for her valuable inputs and suggestions in the project.

I thank Shri. TVS Ram, Group Director,ODCG for his guidance and constant motivation during the term of the training.

I am deeply indebted to Shri. J. Ravishankar, HRDD, Space Applications Centre, ISRO, Ahmedabad, for providing me the rare opportunity to carry out training at one of the finest research organizations of India.

I wish to thank my friends for their delightful company which kept me in good humor throughout the year.

- Parth Sharma
14MECC23

Abstract

Navigation systems are designed to provide precise position details to the user. For getting high accuracy in terms of position, precise timing systems are required. These timing systems require to be thermally stable within sub degree centigrade.

There are various controllers available in literature to provide precise temperature control like, PID controller and its variants (P, I, D and their combinations) and Fuzzy controller. There are no overshoots in the response of Fuzzy controller whereas in response of PID controller, they are present[2] [12]. In precise timing systems, overshoots are not allowed so, Fuzzy controllers are batter to control their temperature. Fuzzy controller gives better performance because they are independent of plant model and plant environment and they are controlled by set of rules. Fuzzy systems are adaptive in nature, manual tuning is not require.

Digital implementation serves better to achieve thermal stability. Nowadays, FPGAs are most perfect devices as compare to DSP Processors and Micro-controllers because of its faster computation, high density and low power consumption.

The project deals with design of Fuzzy Logic based Temperature Controller which have been implemented on Actel-FPGA to achive precision within sub degree centigrade which consist a closed loop system. This close loop system contains signal conditioning circuit, FPGA, drive electronics and sensors to achieve desired set- point . Simulations have been carried out in MATLAB/Simulink-R2013b.

Results shows that, there are no overshoots in close loop system response.

Contents

List of Tables

List of Figures

Abbreviation

GPS	Global Positioning System
QZSS	Quasi-Zenith Satellite System
FPGA	Field Programmable Gate Array
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
RTD	Resistor Temperature Device
ASIC	Application Specific Integrated Circuit
ROM	Read Only Memory
FIFO	First In First Out
CCC	Clock Conditioning Circuit
PLL	Phase Lock Loop
FLC	Fuzzy Logic Control
PID	Proportional Integrator
MF	Membership Function
FIS	Fuzzy Inference System

Chapter 1

Introduction

Temperature controllers are designed to achieve stable temperature in the thermal systems. It takes an input from a temperature sensor then process and controls the system such as a heater or fan. It can be broadly classified in two types of controller:

- Open Loop Controller
- Close Loop Controller

Open loop is a very basic form of temperature controllers. It implies either continuous warming or refrigerating without the knowledge of actual temperature. Closed loop controller are rather much complex but better than the open loop controllers. In this, the temperature is continuously monitored and set to desired temperature by processing the error in closed loop. Fig.1.1 and Fig.1.2 shows the basic block diagram of an open loop and closed loop controllers.

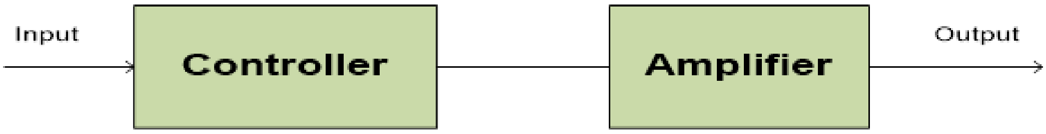


Figure 1.1: Open Loop Controller

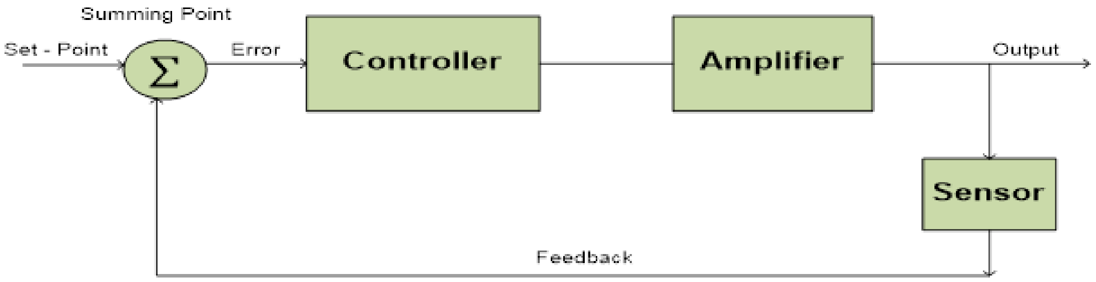


Figure 1.2: Close Loop Controller

It depicts that there is no feedback in open loop which makes it simpler to use as compared to close loop. Close loop controllers utilize the feedback error to achieve the desired accuracy. Closed loop temperature controllers are used in applications where precise control of temperature is required despite of its complexity.

1.1 Need of stability in Timing Systems

There are numerous timing system required in satellite based navigation systems like GPS, GALILEO and QZSS to achieve the precise positional accuracy. Precise time directly implies accurate position in the field of navigation. These system needs to be thermally stable for precise positioning.

1.2 Motivation

- To control the temperature of thermally critical system and the stability of timing and getting accuracy.

- The problems pertaining to classical controllers such as Proportional-Integral-Derivative which require much tuning, overshoot, dead zones saturation and hysteresis.

- To employ the state of the art smart temperature controller technique such as fuzzy logic, in which time delays, overshoot manual tuning procedures and non-linearities are properly taken care.

1.3 Objective

To design an FPGA based Fuzzy Temperature Controller with sub degree Celsius precision for thermally critical system.

1.4 Project Idea

In thermal systems, temperature controller which provides less overshoot and precision of the order of sub degree centigrade are very much required for stable, reliable and good performance. Overshoot in the system response can be sometimes precarious for thermally critical systems as it can cause undesired system response or a permanent damage to the system. So, it has to be taken care in the design of any temperature controller. This thesis presents a Fuzzy based temperature controller which provides sub-degree control over temperature without producing any overshoot in the system response as compared to other available control techniques[**?**].

1.5 Block diagram of the system

The basic block diagram of the Fuzzy based temperature controller system is shown in Fig.1.3. It mainly comprises an FPGA, signal conditioning circuit, Drive electronics and assembly. The functionality of each part is discussed in details in subsequent section below.

1.5.1 Functions of Control loop

The functionality of the closed loop system as described in Fig.1.3 is:

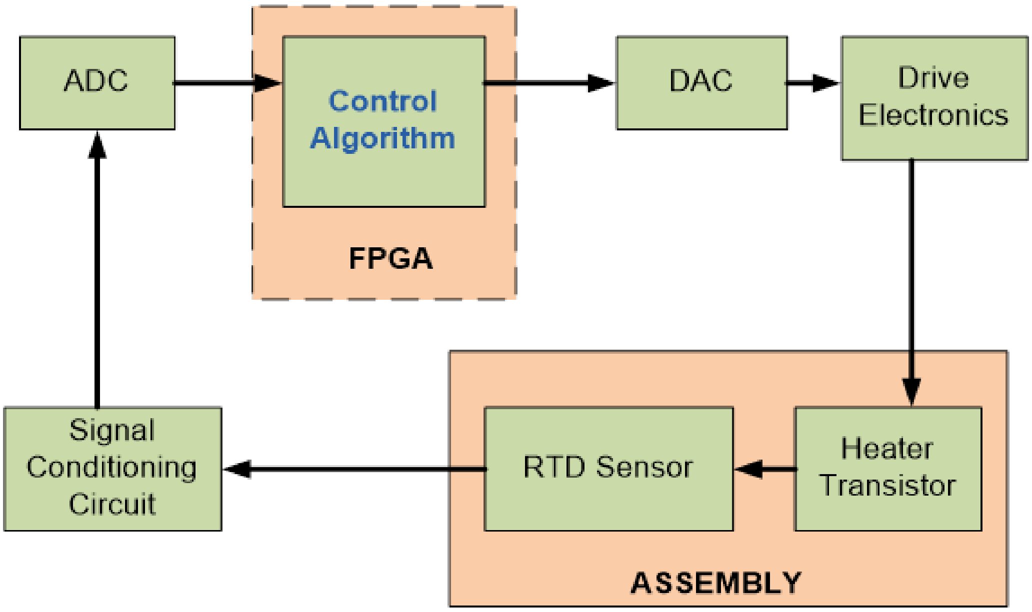


Figure 1.3: Block Diagram of System

- To sense the temperature of the assembly through an RTD sensor.
- To convert the changes in temperature into an equivalent voltage variations.
- To generate the error signal by taking difference between measured and set- point value
- To provide the appropriate gain to the error signal to drive the heater the measured signal to the controller
- To drive the heating element with the processed error.

1.5.2 ADC and DAC

Digital implementation implies the conversion of analog variations of temperature into an equivalent digital values or vice-versa. ADC and DAC are used for this purpose. Resolution of ADC and DAC are very important for achieving the desired precision. 12-bit serial ADC & DAC are used in the present design to achieve a precision of sub degree centigrade.

1.5.3 Resistor Temperature Sensor

Resistive temperature sensors generally have positive temperature coefficient (PTC). The RTD used in the present design is PT100. It has sensitivity of 0.4 ohm /°C. and standard resistance range of 100ohm at 0C to 140ohm at 100C. -200 to +600C. RTD is a resistive device, so resulting voltage can be measured by passing a current through them. However, due to self heat of the resistive wires as the current flows through it and results into variation in resistance (as Ohms Law, I2R) and causes an error in the readings. The RTD is usually connected into a Whetstone Bridge network for avoiding this kind of problem.

1.5.4 Signal conditiong circuit

Analoge signal is modified and make it as per requirement of next stage for further processing is known as signal conditioning. It includes filtering, isolation, range matching, converting, amplification.

Here, electronic temperature sensor gives signal in millivolts range which is very less for driving ADC directly, so need to carry up the voltage level. So that amplification process is required here. Output signal of sensor become more strengthen which can drive ADC by this amplification process.

In this project, we have used Differential amplifier in signal conditioning circuit.

It does not amplify the particular voltages but amplify the difference between two voltages. Two inputs in differential amplifier, one is input signal and other is feedback signal. Inverting and Non-inverting both properties of op-amp is merge up in differential amplifier.

1.5.5 Drive Electronics

An circuit or component which was used for controlling another circuit or component, is known as drive electronic.

In this project driver electronics consist of high power amplifier transistor which is used as the heating element which dissipates power as heat energy. The transistor is common emitter configured.

This circuit drives a resistive heater element with a low-frequency, pulse-width modulation (PWM) voltage source, providing heat output that's directly and linearly proportional to the duty cycle of the drive signal. The circuit permits full output at any duty cycle, as long as the resistive heating element's resistance exceeds a minimum value.

1.5.6 FPGA

Embedded control systems are found in consumer electronics, medical equipment, robotics, automotive products, and industrial processes. Embedded control systems typically use microprocessors, micro-controllers or Digital Signal Processors (DSPs) for their implementation.

The processor itself is connected to various peripherals such as memories, Analog to Digital converters, and other I/O devices. Alternatively, FPGAs are increasingly becoming popular as implementation platforms on which the control algorithms can be implemented by programming.

FPGA gives batter performance over embedded controller. Re-programmable, lower power consumption, lower frequency of operation are good characteristics of FPGA. FPGA is one type of FPD (Field Programmable Device). FPGA cover an array of free circuit elements, interconnect resources called logic blocks. FPGA shape is done through programming by the end user. FPGA preserves very high logic capacity.

The primary manufacturers of FPGAs are Xilinx, Altera, Actel. Random logic, integrating multiple SPLDs, device controller, communication encoding and filtering, small to medium sized systems with SRAM blocks are basic applications of FPGAs.

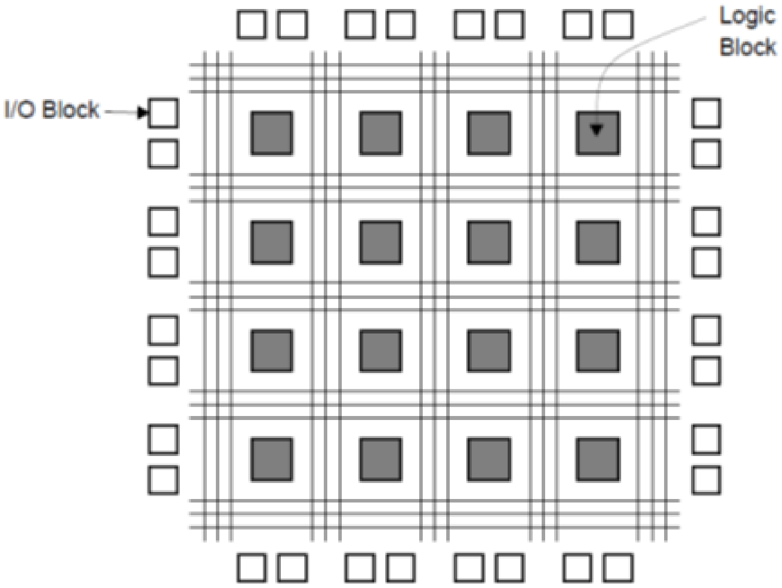


Figure 1.4: Design of FPGA

Here, ProASIC3E named FPGA is used, given by Actel. The proprietary ProASIC3E architecture provides granularity comparable to standard-cell ASICs.

The ProASIC3E device consists of five distinct and programmable architectural features,

- FPGA VersaTiles
- Dedicated Flash ROM
- Dedicated SRAM/FIFO memory
- Extensive CCCs and PLLs
- Pro I/O structure

The FPGA core consists of a group of VersaTiles. Each VersaTile can be configured as

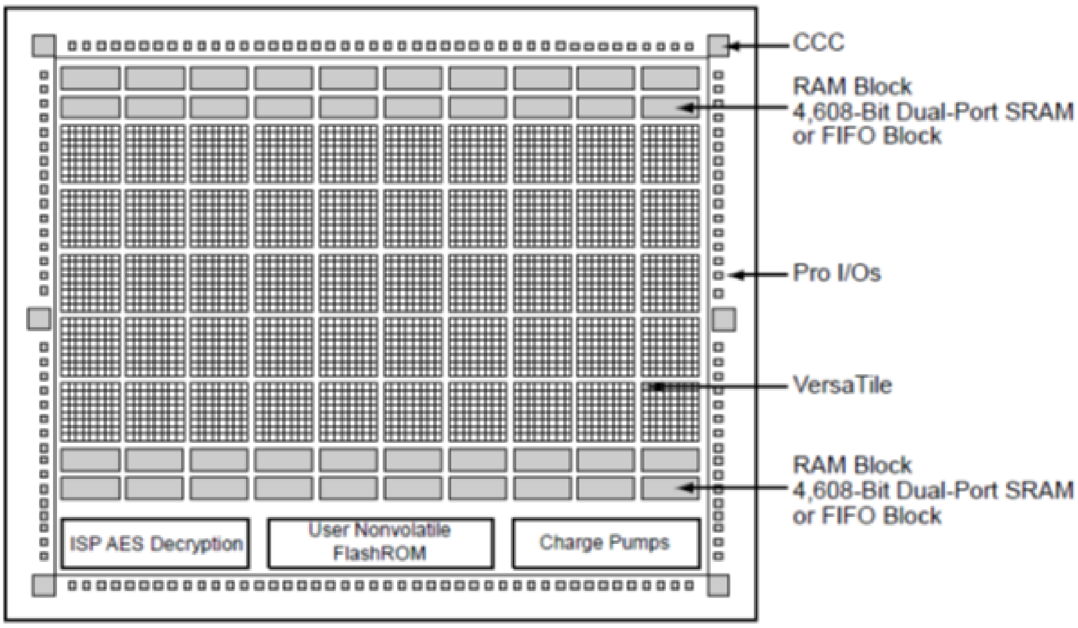


Figure 1.5: Architecture of ProASIC3E

a three input logic function, a D-flip-flop (with or without enable), or a latch by programming the appropriate flash switch interconnections. The versatility of the ProASIC3E core tile as either a three-input lookup table (LUT) equivalent or as a D-flip-flop/latch with enable allows for efficient use of the FPGA fabric. The VersaTile capability is unique to the ProASIC family of third generation architecture Flash FPGAs. VersaTiles are connected with any of the four levels of routing hierarchy. Flash switches are distributed throughout the device to provide nonvolatile, reconfigurable interconnect programming. Maximum core utilization is possible for virtually any design.

1.6 Organization Of Thesis

The rest of thesis is organized as follows,

Chapter 2 *Literature Review*

Background analysis, available controllers and their comparative analysis are described in this chapter. Advantages, disadvantages and application of fuzzy logic also included in this chapter.

Chapter 3 *Design of fuzzy logic*

Designing steps of fuzzy logic and types of fuzzification & defuzzification are part of this chapter. MATLAB fuzzy Toolbox is described here.

Chapter 4 *Development of Fuzzy : Example*

Fuzzy development process in depth, is explained here from starting to end using one example. This chapter is heart of thesis for understanding fuzzy logic.

Chapter 5 *Simulations and Results*

Different plants are taken and fuzzy rule sets are designed for them and simulations are carried out and results are produced. Comparative analysis are carried out from results. Own subsystems and Floating Point modeling are also included in this chapter.

Chapter 6 *Hardware Implementation*

General Hardware Implementation & Fuzzy Hardware Implementation methods are briefed here. Hardware Flow and analysis of main stages in implementation part is explained in this chapter.

Chapter 2

Literature Review

2.1 Background Analysis

The fuzzy controller is first industrial application of fuzzy logic. It was used by two Danish civil engineers, L.P. Holmblad and J.J. Ostergaard, around 1980 for cement kilns [12].

In Sendai City, the Japanese designed fuzzy logic and applied it for subway trains. The end product was very proper, and was generally honored more than other comparable systems based on classical control. This was focused for other applications, like elevator control systems and air conditioning systems. In the early 1990s, the Japanese began to apply fuzzy controller in consumer products, like camcorders, washing machines, vacuum cleaners, and cars.

Takagi and Sugeno expected a fuzzy identification algorithm for modeling human operators control actions. Systematic approach to the design of an FLC was given by Sugeno and his experimental results are pretty notable. However, some steps of this Sugenos algorithm, such as the choice of process state variables, the fuzzy partition of input spaces, and the choice of the membership functions of primary fuzzy sets, depend on trial and error.

Search engine based on fuzzy logic was developed by professor R.R. Yager, Machine

Intelligence Institute, US. It was first established to the public at the Joint International Conference of Artificial Intelligence in 1992 in Chambery, France.

Intelligent dishwasher based on a fuzzy controller introduced by Mayfang in 1995. Which contains a thermistor, for temperature measurement; a conductivity sensor, to measure detergent level from the ions present in the wash; a turbidity sensor that measures scattered and transmitted light to measure the soiling of the wash; and a magnetostrictive sensor to read spin rate. The system determines the optimum wash cycle for any load to obtain the best results with the least amount of energy, detergent, and water.

Canon developed an auto-focusing camera that uses a charge-coupled device (CCD) to measure the clarity of the image in six regions of its field of view and use the information provided to determine if the image is in focus. It also tracks the rate of change of lens movement during focusing, and controls its speed to prevent overshoot. The camera's fuzzy control system uses 12 inputs: 6 to obtain the current clarity data provided by the CCD and 6 to measure the rate of change of lens movement. The output is the position of the lens. The fuzzy control system uses 13 rules and requires 1.1 kilobytes of memory.

An industrial air conditioner designed by Mitsubishi uses 25 heating rules and 25 cooling rules. A temperature sensor provides input, with control outputs fed to an inverter, a compressor valve, and a fan motor. The fuzzy controller heats and cools five times faster compared to the previous design, reduces power consumption by 24 , increases temperature stability by a factor of two, and uses fewer sensors.

2.2 Available Controllers

Following controllers are available in literature,

2.2.1 On/Off Temperature Controller

It is the simple and less expensive temperature control. Device output is either on or off. There is no middle state. When temperature crosses the set-point, it will on or off. The output is on when temperature is below the set-point and the output is off when temperature is above the set-point for heating control.

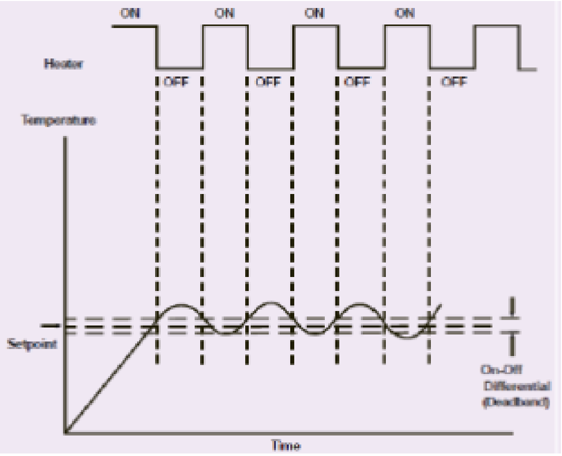


Figure 2.1: On-Off Controller

This way cycling process is endlessly going on. When cycling occurs rapidly, and prevents damage to system, an on-off differential or hysteresis is included to operations of controller. When temperature goes below then it triggers heater and raise the temperature back to set value.

On-off controller is mostly use when a precious control is not necessary, where the temperature change is extremely slowly or the temperature alarm.

2.2.2 Proportional Controller

Power supply is increased or decreased and as per temperature changes and it will respond before temperature change. The temperature range is known as proportional band. The output on: off ratio is 1:1 at the midpoint of proportional band. If the temperature is below set-point, the output will be on for long time and if the temperature is too high, the output will be off for long time.

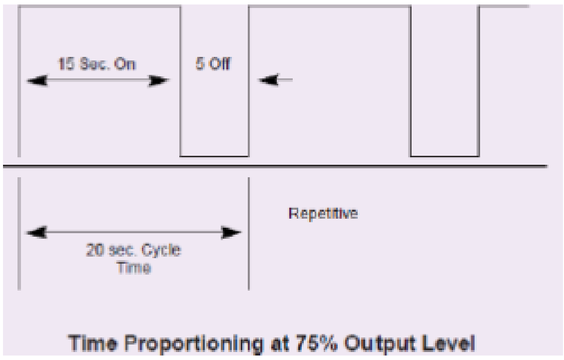


Figure 2.2: Proportional Controller

In proportional controller damping ratio is reduced and so it reduces the stability. Overshoots are coming in picture in these type of controllers. As increases gain steady state error is reduced over here. One of the advantages of this controller is simplicity of operation.

2.2.3 PID Controller

It is a loop feedback controller. It is widely used in industries. It is computing an error signal which is getting by difference between measured value and desired set value. There are three terms in PID Proportional, Integral and Derivative. Output of PID controller is getting by addition of these three components.

$$U(t) = MV(t) = K_P e(t) + K_I \int e(t) + K_D \frac{d}{dt} e(t) \tag{2.1}$$

Here,

U(t)= controller output

K_P = Proportional gain

K_D = Derivative gain

K_I = Integral gain

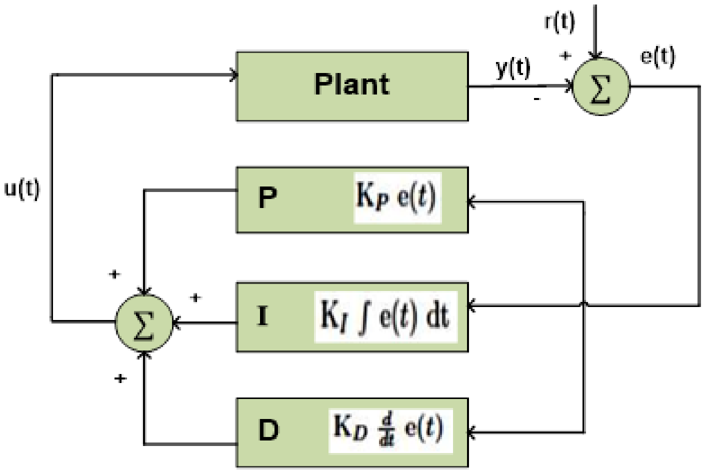


Figure 2.3: PID Controller

The proportional value allows controlling recent situation and regulating according to that. The integral value takes past proportional control situation in to account and derivative value determines the rate at which past control situation has been changing. So, it combines current data, past data and rate of changing data this controller takes control action to control the temperature.

2.2.4 Fuzzy Logic Controller

Fuzzy logic used to control a process that a human can control physically by gaining expertise from experience. The control rules are created by human. Fuzzy gives straight and simple path to define proper results from vague information.

The fuzzy logic controller converts the expert knowledge into an automatic control system. Fuzzy logic is ideal for controlling nonlinear and complex systems. Fuzzy logic gives a powerful support to use human reasoning in the control algorithm. Fuzzy control has been effectively used for complex badly-defined processes. Nonlinear control problems can be solved easily by fuzzy logic compare to old techniques and it was proved by researchers.

2.2.5 Comparative Analysis

As mentioned above, different controllers have number of different characteristics. Available controllers are compared based on control characteristics in below table.

From table, conclusion comes out, among all controllers, fuzzy system gives no overshoots.

Table 2.1: Comparisons of Different Algorithms

Controller	**Rise Time**	**Settling Time**	**Overshoot**	**Steady State Error**
ON-OFF	High	High	Yes	Less
Proportional	High	High	Yes	High
PI	Less	High	Yes	Less
PD	High	Less	Yes	High
PID	Less	High	Yes	Less
Fuzzy	*High*	*Less*	*No*	*Minimal*

As fuzzy system has no overshoots, so it cannot damage over system. Also, characteristics of fuzzy are matches with our requirements for temperature control. So, we have finalized to work with fuzzy controller for design temperature controller.

2.3 Fuzzy: Advantages, Disadvantages and Applications

It is concluded that fuzzy algorithm gives batter results compare to other available algorithms, by making analysis of literature. Advantages and disadvantages of fuzzy logic are listed below.

2.3.1 Advantages

- No need to have a mathematical model of system.

- It is possible to implement expert human knowledge and experience using comprehensible linguistic rules.
- It is possible to control non-linear plants.

2.3.2 Disadvantages

- No general way for choosing the number of rules because a large number of factors are involved in the decision. Ex. Performance of controller, efficiency of computation, human operator, behavior, the choice of linguistic variables etc.
- There is no standard and systematic method for the conversion of the human knowledge or experience into the rule base.
- System model is not known so that is not possible to show the stability.
- It is not sure that rules are proper. It is possible to have an incompatibility between the rules.
- Computing time could be long, because of the complex operations such as fuzzification and mainly defuzzification.

2.4 Applications of Fuzzy Logic

There are unlimited applications of fuzzy logic. Some applications are listed below,

- Water quality control
- Traffic control
- Washing machine
- Model car parking and turning
- Automatic train operation system Automatic transmission control
- Fuzzy logic controller hardware system

- Fuzzy memory devices
- Fuzzy computer
- Nuclear reactor control

Other applications investigated or implemented include: character and handwriting recognition; optical fuzzy systems; robots, including one for making Japanese flower arrangements; voice-controlled robot helicopters, control of flow of powders in film manufacture, elevator systems, and so on.

As above mention, there are plenty of various fields in which Fuzzy logic is used. As per requirement it is used by user. Design of Fuzzy logic is explained in next chapter.

Chapter 3

Design of Fuzzy Logic

Common design of fuzzy system consists of four steps. Starting with Fuzzification, fuzzy system is finished by Defuzzification process.

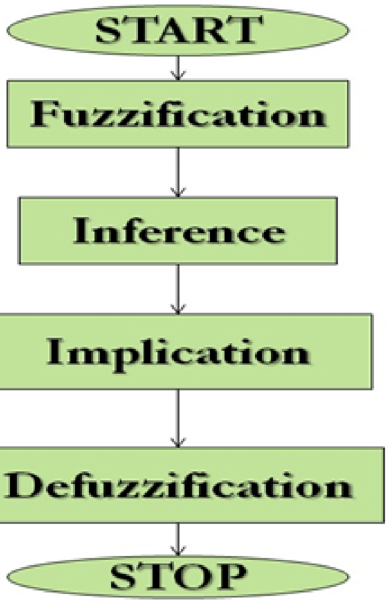


Figure 3.1: Fuzzy System

Intermediate steps are Inference and Implication. Above figure gives idea about flow of fuzzy system and its sequence of steps.

3.1 Fuzzification

Crisp values are converted into fuzzy sets, which is known as membership functions. Mapping of inputs to proper membership functions is done in this step. Membership functions should wide enough to reduce the sensitivity to noise if data is affected by noise.

3.2 Inference

It is based on IF- THEN rules. Here IF part is known as antecedent and THEN part is known as consequents. The logic connections between premises by fuzzy AND or fuzzy OR. In this part, control rule is deciding as per current time and relevant current input is decided. Inference system calculates membership value as output.

3.3 Implication

This step is determination of output of individual rule. Appropriate rules are selected to be fired based on the fuzzy variables, which are chosen based on the regions where they fall in.

For example, if space divides into 3 regions where each region containing only two fuzzy variables. So, only two fuzzy variables are triggered at any given time. As a result, each fuzzy variable results in firing two rules. As a result, a total of 5 rules are fired.

3.4 Defuzzification

In this step, fuzzy variable is converted into crisp value. It converts joined fuzzified result back into a specific crisp control output value using some mechanisms. It makes one surface by conversion of one by one variable based on rules. Smooth surface results into smooth response.

3.5 Membership functions

The membership function exactly defines degree of membership based on a property such as temperature or pressure. There is much type of membership functions Triangular, Gaussian, trapezoidal, Gaussian, Bell-shaped etc.

Bell shaped MF is a more flexible because it has three parameters to adjust compared to the Gaussian-shaped MF which has only two parameters.

The membership functions are constructed from expert knowledge and then modified by proper graphing the control response of the process. Shape of membership function is depending on applications. Mostly, researchers take not care for selection of membership function. Mostly triangular functions used with equal span throughout universe of discourse.

Main advantage of choosing of membership function is that it reduced the difficulties in analyzing the structure of FLC. Membership function range should be narrow for getting closer response to the set-point.

3.6 Types of fuzzification and defuzzification

There are many methods of fuzzifications and defuzzifications are available in literature. Some of them are explained below.

Table 3.1: Fuzzification and Defuzzification Methods

Fuzzification	**Defuzzification**
Max - Dot	Mean of Maxima
Min - Max	Center of Gravity
Tsukamoto	Tsukamoto
Takagi and Sugeno	Weighted Average

3.6.1 Max Dot and Mean of Maxima

Max-Dot Fuzzification

The final output membership function for each output is produced by combination of fuzzy sets assigned to that output in a winding up after their degree of membership values to peak at the degree of membership values for the corresponding premise.

By winding up at one peak it is known as max-dot method. Combination of all membership produce one surface and then defuzzification can be applied on that. Mean of Maxima is the defuzzification method corresponding to this max-dot fuzzification process.

Mean of Maxima Defuzzification

This method focusses on maximum degree of membership. By averaging values with maximum membership mean of maxima is calculated. In the case of discrete universe:

$$Z = \sum_{i=1}^{l} \frac{r_i}{l} \tag{3.1}$$

Where, l is the number of the quantized r values which reach their maximum membership.

It covers less area so, results into lower resolution.

3.6.2 Min Max and Centre of Gravity

Min-Max Fuzzification

The final output membership function is the combination of the fuzzy sets assigned to that output after cutting their degree of membership values at the degree of membership for the corresponding premise.

It covers more area compare than previous method and make one surface. It uses Centre of Gravity method for Defuzzification purpose.

Centre of Gravity Defuzzification

It generates a center of gravity of the resulting fuzzy set of a control action. If we distinct the universe it is:

$$Z = \frac{\sum_{i=1}^{n} r_i Z_i}{\sum_{i=1}^{n} Z_i} \tag{3.2}$$

Where, n is the number of quantization levels, ri is the amount of control output at the quantization level i and zi represents its membership value.

It results batter, because it is given by center of summed area, which is contributed by the different fuzzy output. It gives smoothness, because small degree of membership taken into account.

This whole procedure of fuzzification and Defuzzification is practically implementable in MATLAB/Simulink. It is known as Mamdani method.

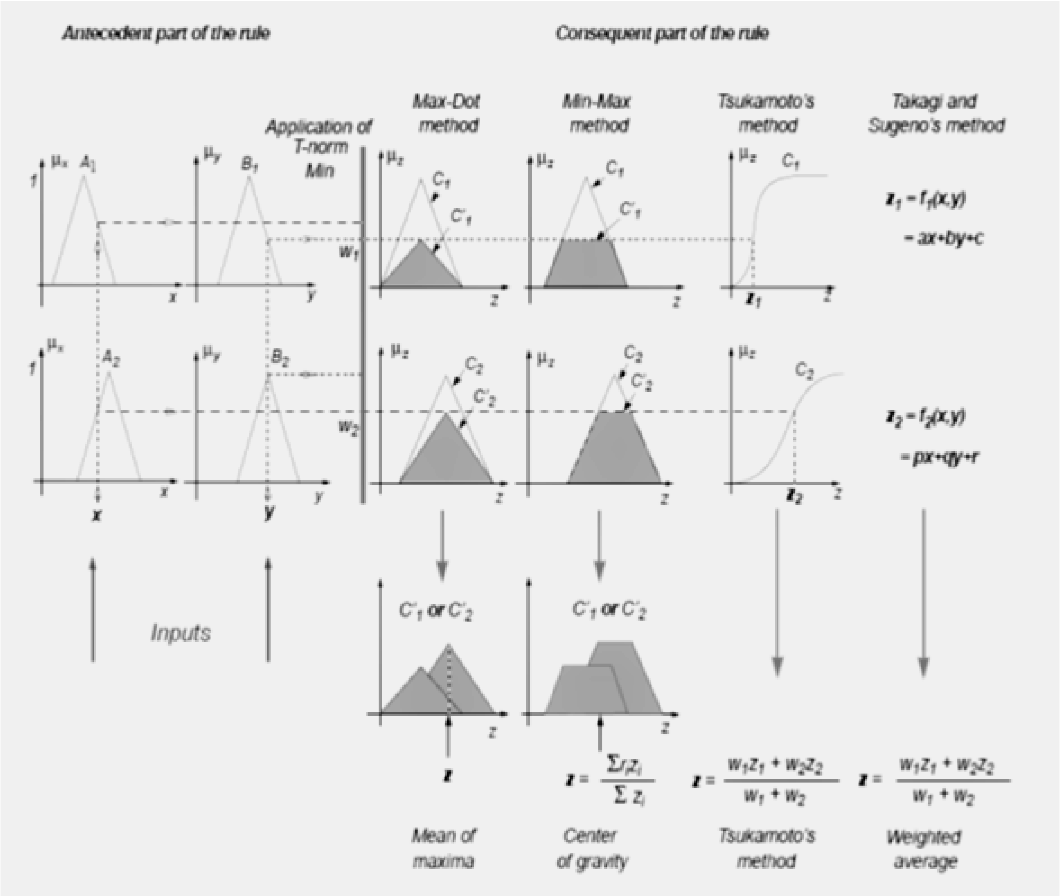


Figure 3.2: Fuzzification and Defuzzification Methods [1]

3.6.3 Tsukamotos method

Tsukamotos Fuzzification

The output membership function has to be monotonously non-decreasing. Then the overall output is the weighted average of each rules crisp output induced by the rule power and output membership function.

Tsukamotos Defuzzification

If monotonous membership functions are used, then the crisp control action can be calculated as follows:

$$Z = \frac{\sum\limits_{i=1}^{n} W_i Z_i}{\sum\limits_{i=1}^{n} Z_i} \tag{3.3}$$

Where, n is the number of rules with firing strength wi is grater then zero and zi is the amount of control action recommended by the rule i.

This whole fuzzification and Defuzzification process is not practically implementable. It is a process given by scientist. Previous two methods have used triangular membership function whereas here, it uses monotonous membership function. So, it is not a generic one.

3.6.4 Takagi and Sugeno and Weighted Average

Takagi and Sugeno Fuzzification

Each rules output is a linear combination of input variables. The crisp output is the weighted average of each rules output.

Example:
X and Y are input and Z is output.
Rules:
If $X = A_1 and Y = B_1 then Z = C_1$
If $X = A_2 and Y = B_2 then Z = C_2$
——-
——-
If $X = A_n and Y = B_1 then Z = C_n$
Suppose two rules are activated (A rule is activated when its firing strength is different than zero).
R1: $X = A_1 and Y = B_1 then Z = C_1$

R2: $X = A_2 and Y = B_2 then Z = C_2$
Here, the fuzzy operator AND means min.
The firing strength of the first activated rule is,
$W1 = min(\mu_{A1}(x), \mu_{B1}(y))$
and firing strength of second activated rule is,
$W2 = min(\mu_{A2}(x), \mu_{B2}(y))$.

It is given by Takagi and Sugeno scientists. It is most similer to Min Fuzzification mathod. It uses weighted average defuzzification method.

Weighted Average defuzzification

This method is used when the fuzzy control rules are the functions of their inputs. Generally, consequent part of the rule is
$Z = f(x, y)$
If W_i is the firing strength of the rule i, then the crisp value is given by:

$$Z = \frac{\sum_{i=1}^{n} W_i f(x_i, y_i)}{\sum_{i=1}^{n} Z_i} \tag{3.4}$$

Where, n is the number of firing rules.

Weighted average gives good precision than normal mean method. This whole process gives batter result compare to previous processes. It is practically implementable in MATLAB/Simulink.

3.7 Designing of Fuzzy System in MATLAB/Simulink

In MATLAB/Simulink we can design practical fuzzy system by two methods,

- Mamdani
- Sugeno

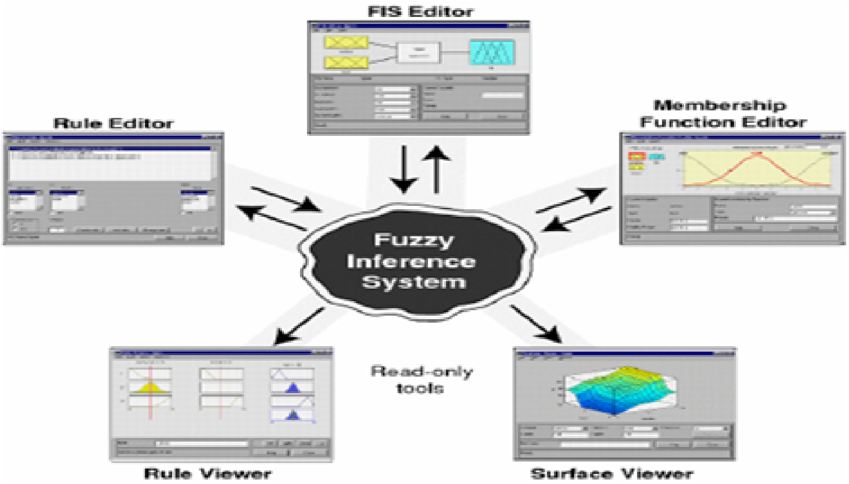


Figure 3.3: Fuzzy Inferance System Tool

Difference between Mamdani and Sugeno

Mamdani

- Widely accepted for getting expert knowledge
- More human like method
- Mostly centroid defuzzification method is used

Sugeno

- Work well with adaptive and optimization technique
- Adaptive technique customizes the membership functions so it is best for modeling data so in Nuro-Fuzzy technique this Sugeno method is used
- Mostly weighted average dufuzzification method is used

Chapter 4

Development of Fuzzy system: Example

Intelligent control is performed with the help of Fuzzy Logic as a tool. Fuzzy Logic enables the development of rule-based behavior. The knowledge of an expert can be coded in the form a rule-base, and used in decision making.

Fuzzy logic Tool-Box is directly available in MATLAB/Simulink Software. Let us understand Fuzzy development by one example,

Let's take two input case

Here, two inputs are Error (error signal) and CError (rate of change in error) as well as

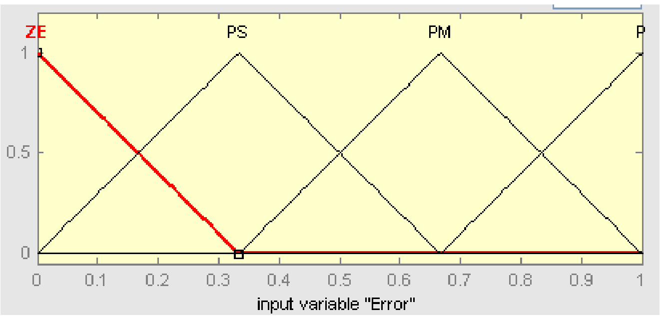


Figure 4.1: Input1 Membership Function

one output signal.

Based on two inputs Fuzzy logic find out amplitude of voltage signal that is necessary to be sent to heater to maintain constant temperature. That is provided by Output signal.

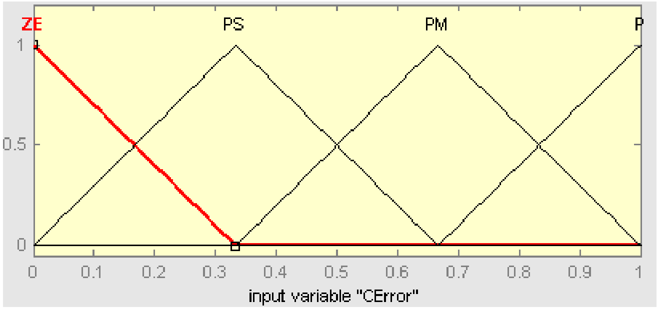


Figure 4.2: Input2 Membership Function

Error, CError and Output all are divided by four membership functions

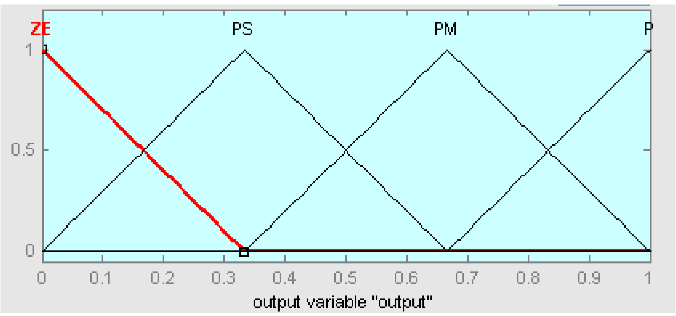


Figure 4.3: Output Membership Function

Four fuzzy variables : ZE (Zero), PS (Positive Small), PM (Positive Medium) and PL (Positive Large).

4.1 Rule Base Preparation

Error = 0.35 and CError = 0.82

According, two functions for each f (0) = 0.05, f (1) =0.95, f (2) =0.54 and f (3) =0.46

Table 4.1: Rule Base

Error CError	**ZE**	**PS**	**PM**	**P**
ZE	ZE	PS	PM	p
PS	PS	Ps	PM	P
PM	PM	PM	PM	P
P	P	P	P	P

4.2 Fuzzification

Fuzzification is the process to convert crisp value into corresponding Fuzzified variable.

CError = 0.82 intersection, got 0.46 and 0.54. It is intersecting PM and P variable

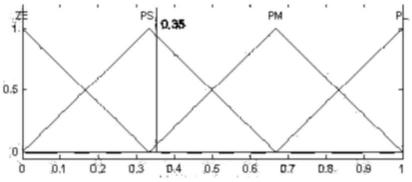


Figure 4.4: Error value = 0.35

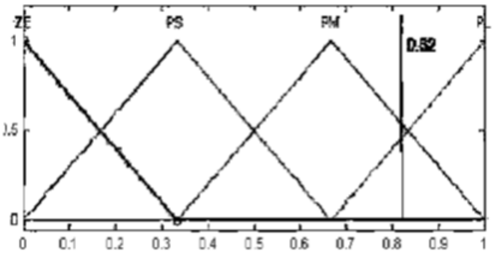


Figure 4.5: CError value = 0.82

In Fig. - 4.4 and Fig. - 4.5 two inputs are taken in to account, Error and Cerror. Crisp values of these two variables are converted into Fuzzy variable according to region.

4.3 Inference

In inference module rules are chosen which should be fired according region where they fall in.

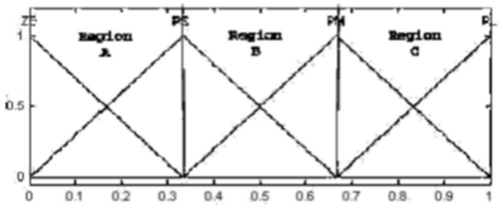


Figure 4.6: Region division for Error

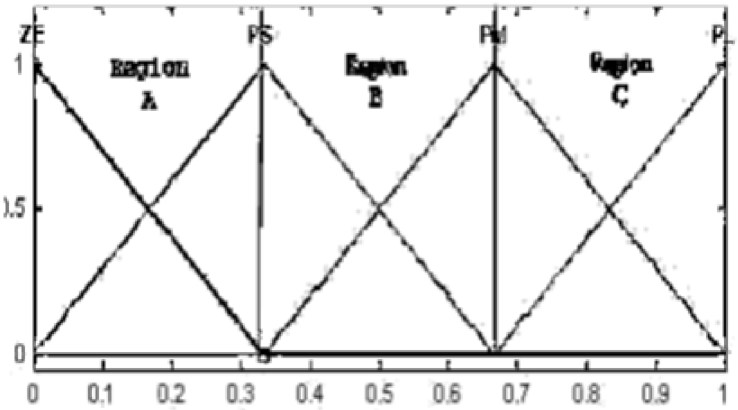


Figure 4.7: Region division for CError

For Inputs

Table 4.2: Region wise range for Input

Input	Region	Range
Error/CError	A	0-0.33
	B	0.33-0.66
	C	0.66-1

For Output

Table 4.3: Region wise range for Output

Region	**Single Value**	**Value On Region**
ZE	0	0
PS	33.33	0.33
PM	66.66	0.66
P	100	1

Now by Rule Base,

If Error is PS and CError is PM then output is PM

If Error is PS and CError is P then Output is P

If Error is PM and CError is PM then output is PM

If Error is PM and CError is P then output is P

So, by applied these rules getting rule [0] = 66.66, rule [1] = 100, rule [2] = 66.66, rule [3] = 100

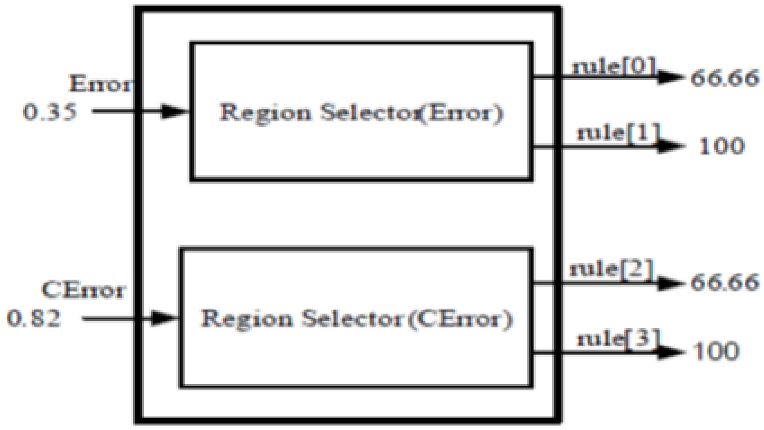


Figure 4.8: Inference

4.4 Implication

It is Mamdani Min implication operation

Here, according to Min (AND) operation is done.

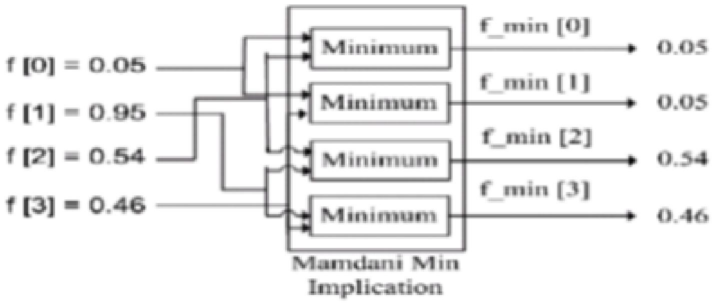


Figure 4.9: Implication

$f(min[0]) = f[0]andf[2] = 0.05\hat{0}.54 = 0.05$

$f(min[1]) = f[0]andf[3] = 0.05\hat{0}.46 = 0.05$

$f(min[2]) = f[1]andf[2] = 0.95\hat{0}.54 = 0.54$

$f(min[3]) = f[1]andf[3] = 0.95\hat{0}.46 = 0.46$

4.5 Defuzzification

Center of gravity (Centroid) method is expressed as below equation,

$$Z = \frac{\sum f(min) * rule[i]}{\sum f(min)} \tag{4.1}$$

Apply this equation to our case,

$$\frac{f(min[0]) * rule[0] + f(min[1]) * rule[1] + f(min[2]) * rule[2] + f(min[3]) * rule[3]}{f(min[0]) + f(min[1]) + f(min[2]) + f(min[3]} \tag{4.2}$$

$= (0.05 * 66.66) + (0.05 * 100) + (0.54 * 66.66) + (0.46 * 100)/0.05 + 0.05 + 0.54 + 0.46$

$= 0.821$

$= 82.1\%$

By, above example, whole development process [12] becomes clear. All points are calculated and whole surface is prepared by step by step procedure. Defzzification process use center of gravity method and convert fuzzy variable into crisp value back. By inspiring this development, simulation for various plants are carried out in next chapter.

Chapter 5

Simulations and Results

Simulation is one of the test points for any algorithm or design. Fuzzy logic is independent of plant and environment. Understanding about the designing of rule-base in Fuzzy system is vary basic requirement.

MATLAB/Simulink - Fuzzy Toolbox is used as a platform of simulations. Before starting simulation of fuzzy first of all input and output range should be known. According to range input and output are divided into fuzzy variables. After that, Fuzzy rule-base is prepared according to control actions.

Input and output range are measured using PID tuning. First of all, PID put as a controller and tune it and measure ranges into scope.

In the following, Different plants are controlled by Fuzzy controller for getting hands on the logic.

Plant – 1	$1/(2s^2+10s+2)$
Plant – 2	$1/(10s^2+20s+10)$
Plant – 3	$1/(10s^2+20/3s+1)$
Plant – 4	$1/(1s^3+2s^2+5s+1)$
Plant – 5	$1/(1s^3+3s^2+3s+1)$

Figure 5.1: Different Plant Models

5.1 Plant Models

5.2 Simulations

Fig. - 5.2 is a plant model, which consist of second order transfer function.

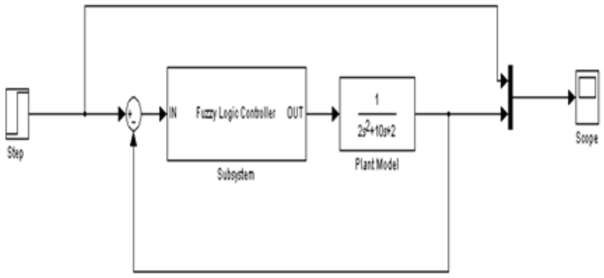


Figure 5.2: Simulation model for Fuzzy Logic Controller of plant 1

Fuzzy logic controller is developed in MATLAB/ Simulink. It is named as a subsystem here.

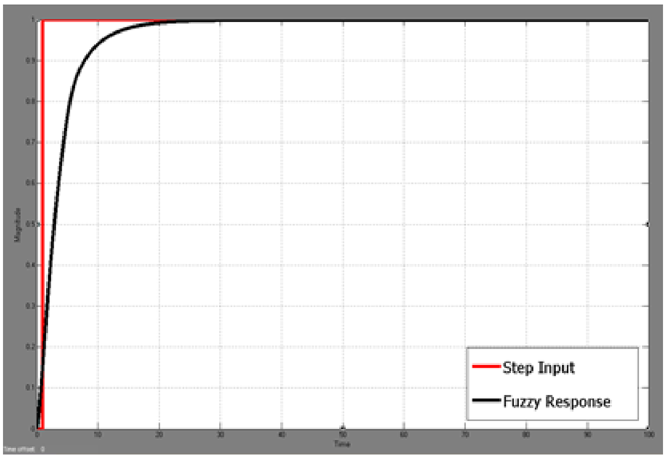


Figure 5.3: Controller response of plant - 1

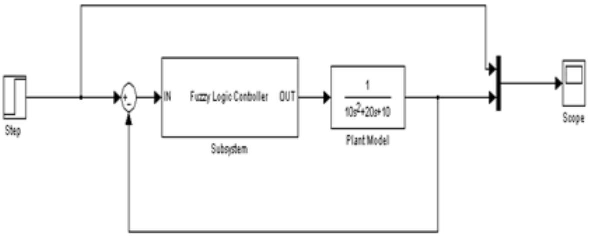


Figure 5.4: Simulation model for Fuzzy Logic Controller of plant 2

As seen a controller result in scope, It is decided that, designed rule-sets are proper and fuzzy response can set on set-point as step input goes.

Same as plant model-1, fuzzy system is developed for another plant-2 which also consist second order transfer function with different poles in Fig. - 5.4.

Fuzzy response sets on set-point accordingly step input in Fig. - 5.5 .

Third plant model is also similar to first and second plant model in Fig. - 5.6, only difference is in terms of poles of transfer function.

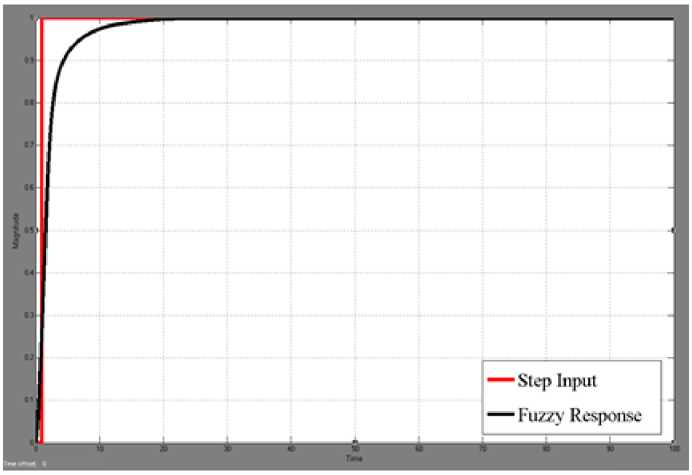


Figure 5.5: Controller response of plant - 2

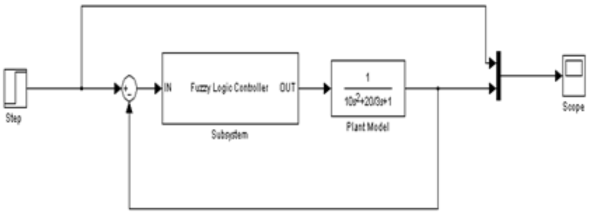


Figure 5.6: Simulation model for Fuzzy Logic Controller of plant 3

Same way of plant model 1 and 2, in third plant also, fuzzy rule sets are proper designed and which can control the plant and sets on set-point as per step input in Fig. - 5.7.

This plant model-4 is third order system Fig. - 5.8. Accordingly, rule-sets are designed for control this plant.

The responses of all plant model shows that, the designed Fuzzy Logic Controller can control different plants by variation in Rule-Sets. Each Rule-Sets are corresponding to

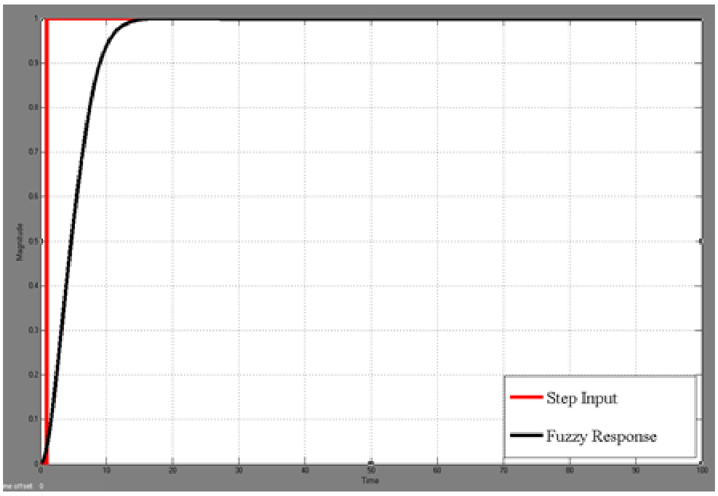


Figure 5.7: Controller response of plant - 3

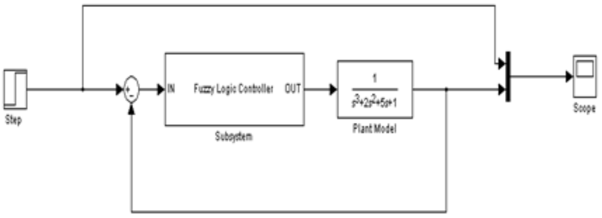


Figure 5.8: Simulation model for Fuzzy Logic Controller of plant 4

particular plant characteristics.

Different fuzzy systems and PID are simulated in Fig. - 5.12. Comparisons are observed in Fig. - 5.13. In PID system, overshoots and undershoots are present while there are no overshoot and undershoot in fuzzy system. In single input fuzzy system only error is take in account.

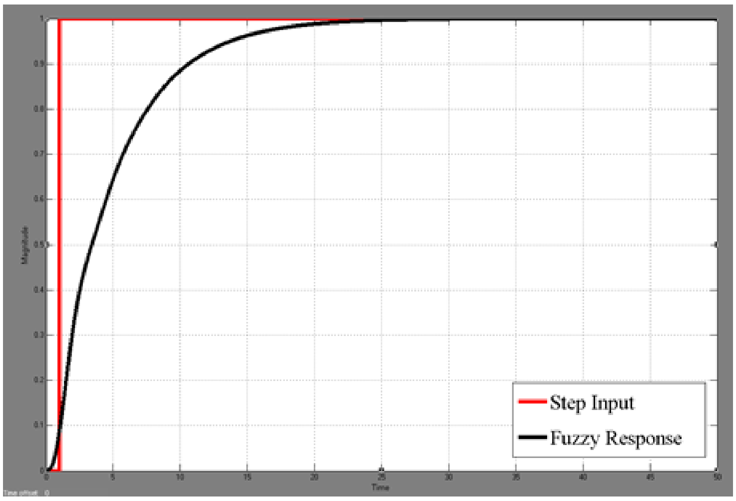


Figure 5.9: Controller response of plant - 4

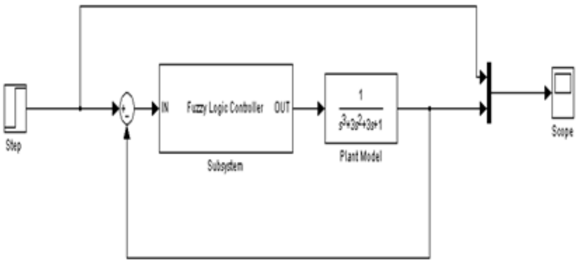


Figure 5.10: Simulation model for Fuzzy Logic Controller of plant 5

By applied negative feedback error values are generated. In two input case error and change in error, two variables are taken in account. For getting change in error, derivative is applied over here. In three input case three input variables are considered. They are error, change in error and change of change in error. For getting change of change in error, integration is applied over here.

As and when numbers of inputs are increase, complexities are also increase. So, generally in most of case researchers prefer two input fuzzy system and also this two input case gives simplicity to construct fuzzy rules also.

Rule-sets are properly designed so that fuzzy response settle down on set-point.

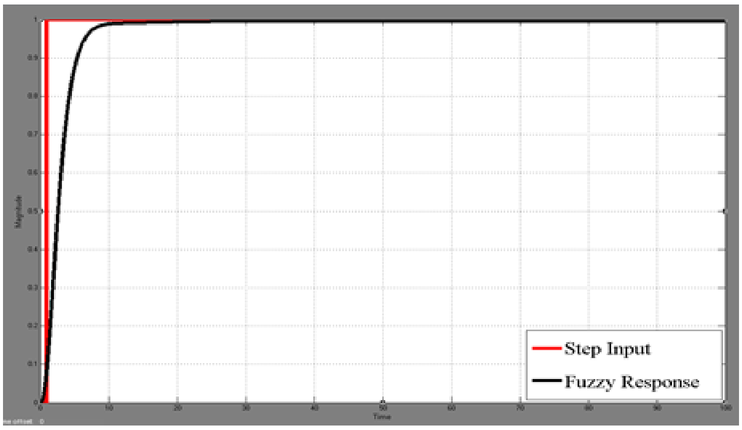


Figure 5.11: Controller response of plant - 5

Stair case input is applied and results are observed Fig. - 5.14 and Fig. - 5.15. As and when in industrial plant when set points are change then system should be quite intelligent so that as per requirement it can control the plant. Two set points are taken in consideration. First set point is 1 and after some time it is 2. By applied proper rules fuzzy controller can adaptively control the plant and set it as per requirement of set point.

There are different types of membership functions are available in the MATLAB toolbox. They are trimf, trapmf, gaussmf, gauss2mf, gbellmf, sigmf, dsigmf, psigmf, pimf, smf and zmf. Among all these here some are simulated. There is not much difference for different membership function. Mostly at the initial level trimfs are used. It is observed that for particular plant model gbellmf takes less settling time compare to others Fig. - 5.16 and Fig. - 5.17. In each and every plant model choice of membership functions are based on trial and error.

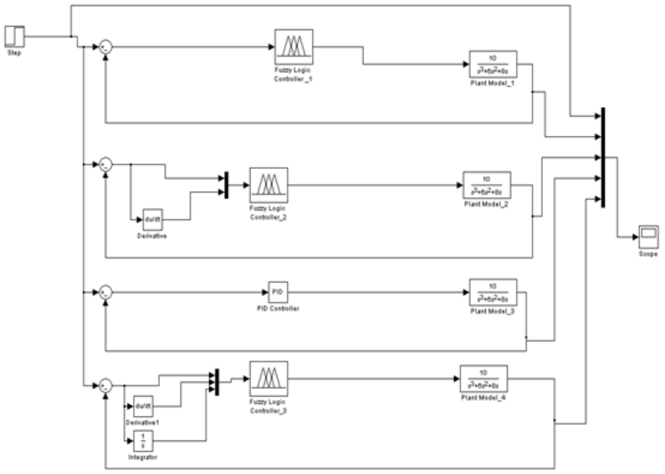


Figure 5.12: Simulation model for Single input, two input, three input Fuzzy and PID controllers

5.3 Results

All types of simulations are done in MATLAB/Simulink and results are observed.

5.3.1 Different type of Control systems

Compression results with PID controller and three different fuzzy controllers are listed in below table [5]. Comparisons are done based on their characteristics.

Table 5.1: Comparison of different type of control systems

	Rise Time	**Set Time**	**Overshoot**
PID	Very Less	High	Yes
One Input Fuzzy	High	Less	No
Two Input Fuzzy	Moderate	Moderate	No
Three Input Fuzzy	Less	Very Less	No

As listed in table, we can say that, there is no overshoots in fuzzy systems and settling time is more in fuzzy system compare to PID controller.

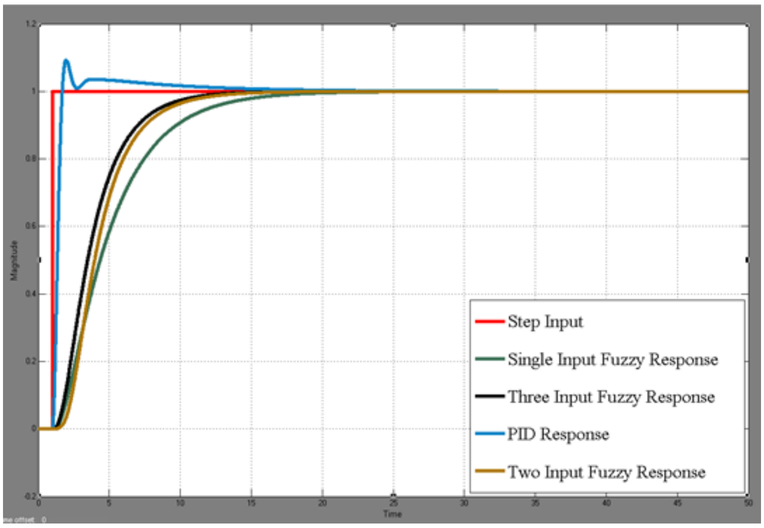


Figure 5.13: Controller responses for single input, two input, three input

5.3.2 PID and Fuzzy

PID controller is compare with fuzzy controller using their characteristics in below table [4]

Table 5.2: Comparison of Fuzzy and PID system

	Overshoot	**Set Time**	**Rise Time**
PID	Yes	High	Less
Fuzzy	No	Less	High

Fuzzy system gives batter result compare to PID, because it has no overshoot.

5.4 Simulations based on Hardware Aspects

There are two problems have been faced in hardware implementation.

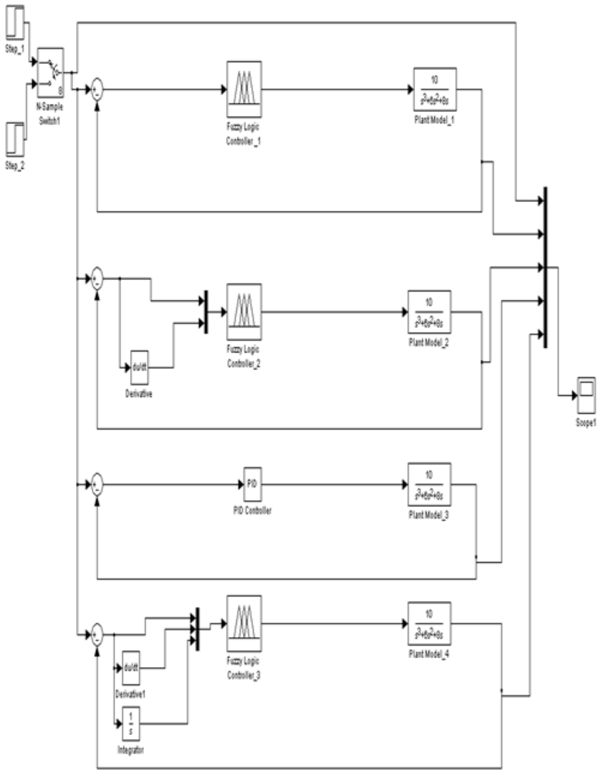


Figure 5.14: Simulation model for different set points Fuzzy Logic Controller

5.4.1 Problem 1

Fuzzy controller is not directly available in hardware.

5.4.2 Solution

Division of fuzzy system into sub parts which are available in hardware Fig. - 5.18.

It is observed form simulation that, Toolbox of MATLAB/Simulink and new designed subsystem gives mostly similar results Fig. - 5.24.

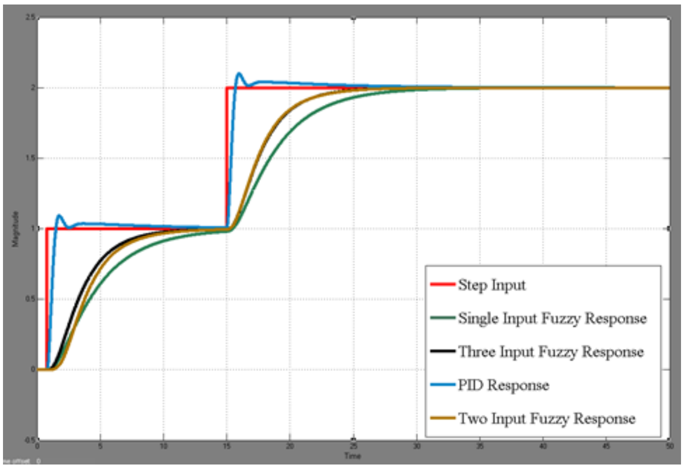


Figure 5.15: Simulation response for different set points Fuzzy system

5.4.3 Problem 2

Floating point model cannot feasible method for hardware implementation.

5.4.4 Solution

Conversion of floating point model into fixed point model

Need of Fixed Point Modeling

In MATLAB (64 or 32 bit), large number of bits are used for all operation, so large number of flip-flops, large area and more power consumption, so it has converted into fixed at 12 bit in fix point modeling.

Solutions in Fixed Point modeling

- Quantization errors can be reduced by selecting appropriate scaling or resolution (position of binary or radix point) and word length or range (limited by hardware).
- Longer development time can be mitigated by using Model-Based Design, which helps to reduce development time and resources.

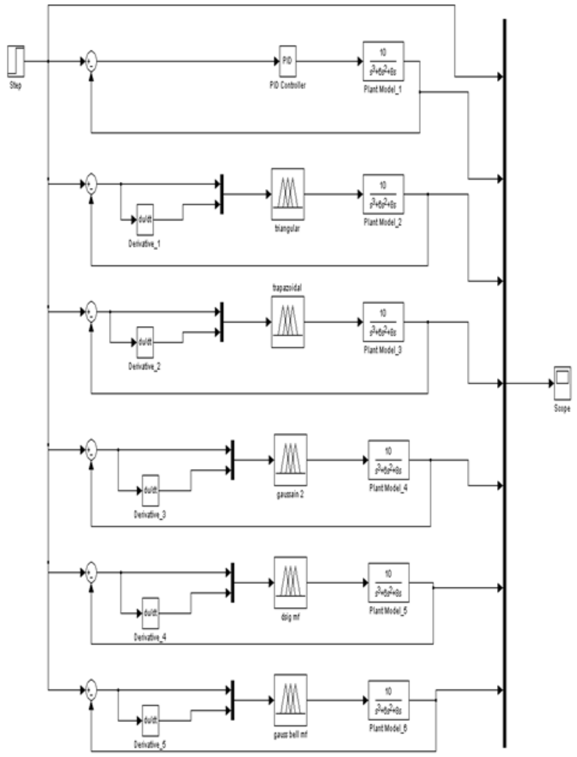


Figure 5.16: Simulation model for two input fuzzy system using different type of membership functions

- Implementation errors can be reduced by using simulation and rapid prototyping, in-loop testing, automated verification and automated scaling tools helps to reduce and detect errors.

It is observed from simulation that, fixed point model gives approximately same result as floating point model Fig. - 5.26. Some differences between two responses are due to quantization errors.

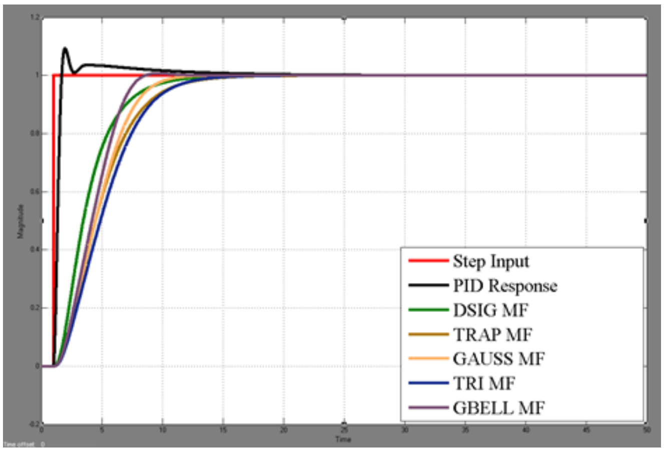


Figure 5.17: Simulation response for two input fuzzy system using different type of membership functions

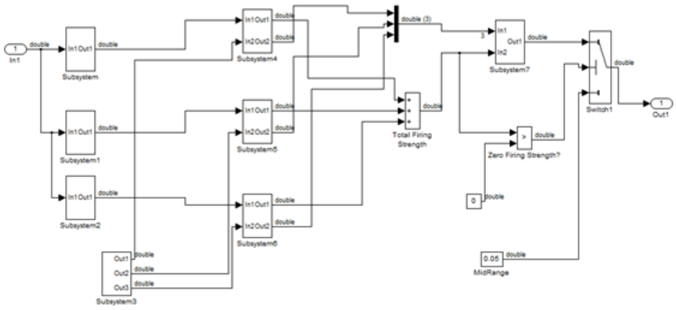


Figure 5.18: Conversion of fuzzy MATLAB toolbox into small sub-systems

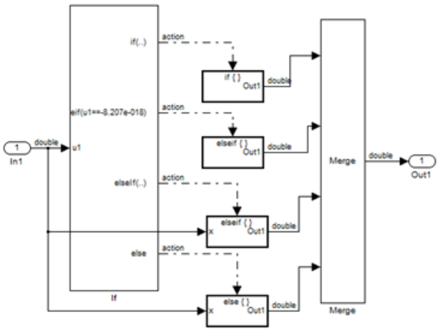


Figure 5.19: Subsystem for input membership function

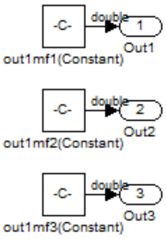


Figure 5.20: Subsystem for output membership function

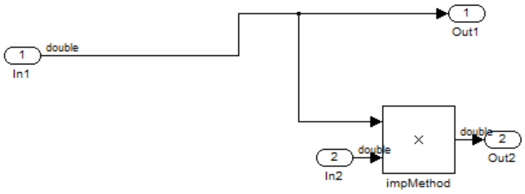


Figure 5.21: Subsystem for weight provision

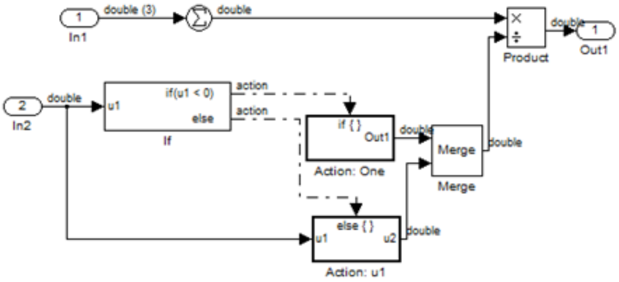


Figure 5.22: Subsystem for Defuzzification process

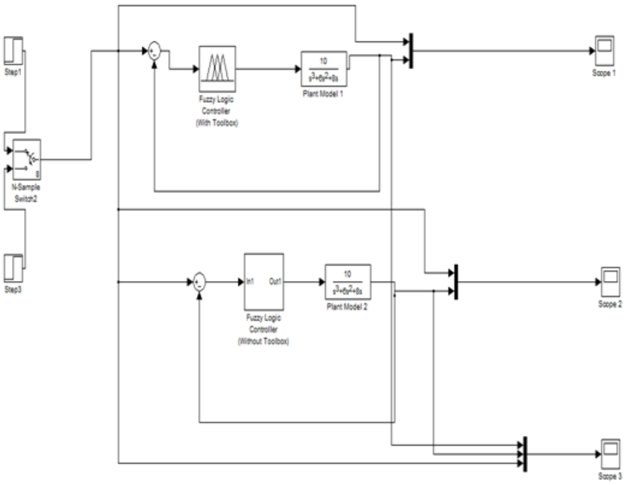


Figure 5.23: Simulation Model - With and Without Toolbox

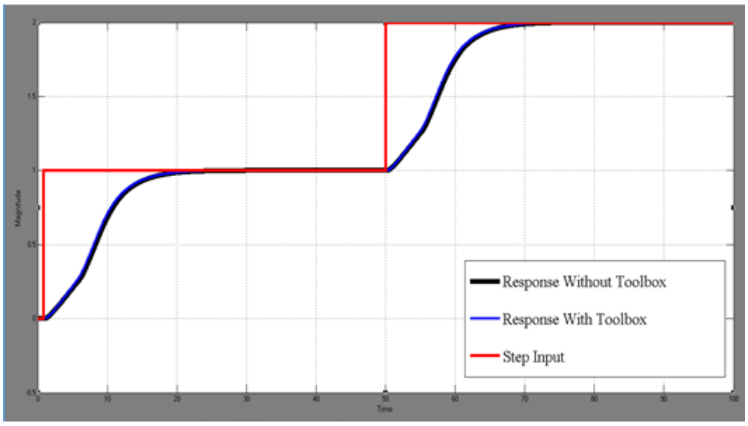


Figure 5.24: Simulation Result - With and Without Toolbox

Table 5.3: Difference Between Fixed and Floating Point Modeling

Consideration	Fix Point Modeling	Floating Point Modeling
RAM/ROM Consumption	Small	Large
Execution Time	Faster	Slower
Word Size and Scaling	Flexible	Inflexible
Complexity	More	Less
Hardware Power Consumption	Low	High
Error Prone	More Quantization and Overflow error due to small range	Easy to produce good resolution and range

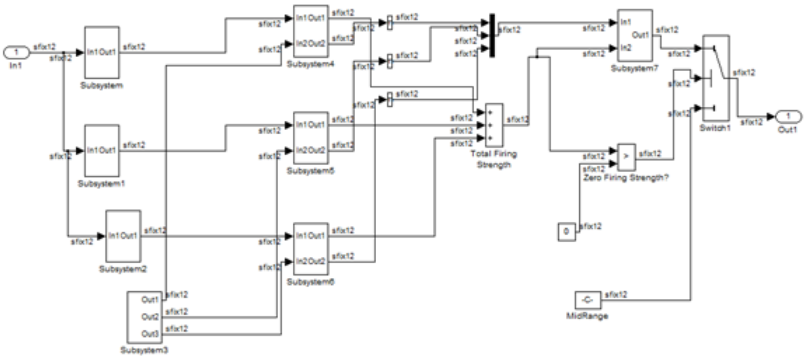


Figure 5.25: Model of Fix-Point

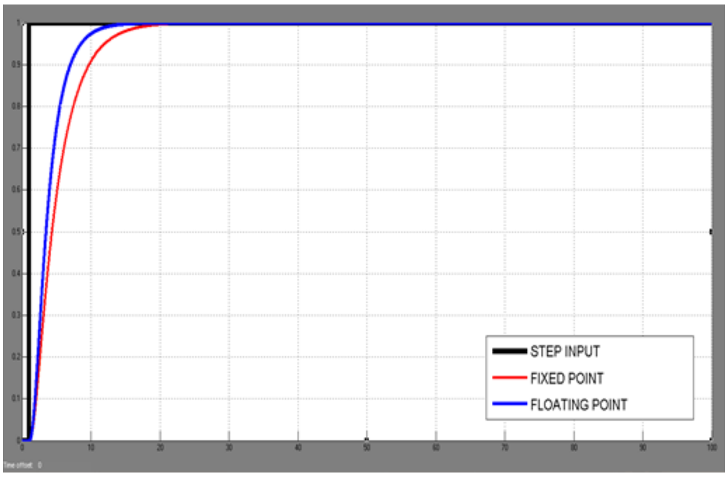


Figure 5.26: Response of Fix-Point Model

Chapter 6

Hardware Implimentation

Choice of method for implement hardware is a very important task.

6.1 General Implementation Methods

Dedicated Integrated Circuits:
Special attention has focused on increasing the processing speed of the fuzzy hardware.
Programmable Integrated Circuits:
Referring to the FPGAs.
Commercial processors:
Several manufacturers have introduced fuzzy instructions in their products. This allows flexibility in the programming of these devices and increases the speed, favouring the implementation of fuzzy systems on them.

6.2 Fuzzy Implementation Methods

Mainly, Three types for Fuzzy implementation on hardware,

6.2.1 Analog Fuzzy Implementation

The high speed, low silicon area occupation, low consumption and high parallelism are decisive reasons for implementing FISs on an analog hardware. In such implementations, solutions are focused on how the circuits work, distinguishing between circuits working in current mode, voltage mode or hybrid mode (trans conductance). One of the benefits of such implementations is that they have a natural connection with different sensors and/or actuators as almost all of them have voltage mode (e.g. range = -10 to +10) or current mode (e.g. 4 to 20 mA), that is converters A/D and D/A are not required.

6.2.2 Digital Fuzzy Implementation

Digital implementations have greater immunity against factors such as noise, temperature or voltage variation, among others. In contrast, the process speed tends to be lower than in analog devices although evolution in integration technologies has changed this scenario.

6.2.3 Mixed fuzzy implementation

Mixed fuzzy implementations present an alternative to integrally analog circuits, combining the capabilities of digital and analog solutions. The fuzzy inference engine is implemented on the analog part, giving the system a high parallelization of inferences with low silicon area, high speed and low power consumption. The digital part allows the programming and storage system parameters.

6.2.4 Comparison of different implementations

The differences between these implementations lie in speed, required power, size, integration, stability, accuracy or conformation of the membership functions, among others. All works consulted provide benefits in some of the parts that compose a fuzzy system compared to devices operating in a different mode. Digital implementations on a dedicated hardware have brought greater immunity to external factors (e.g. noise or fluctuation of the power supply) than analog implementations. The speed has been achieved by implementing parallel rules and optimizing the sequential rules. In systems with parallel rules, the membership functions are stored in memory. This grows when the accuracy increases so it tends to give low accuracy systems. In sequential implementations, both the membership functions and the rules are stored in memory. By increasing the number of inputs, the memory grows exponentially, and therefore techniques have been proposed to minimize this effect. Mixed systems present their inferences on the analog area and programming, and parameters on the digital area.

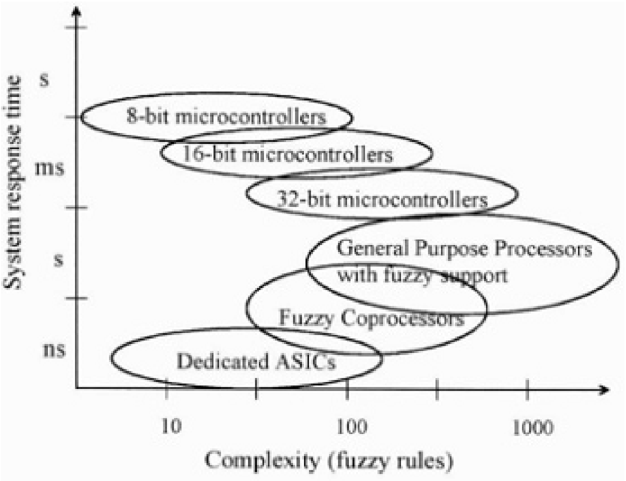


Figure 6.1: Complexity according to different implementation

6.3 Implementation Flow

First step is to convert algorithm into HDL language (Verilog). Then after functional Synthesis and Place and Root is done. Finally, code is dump into FPGA. In between, Pre-synthesis and Post-synthesis simulation is done.

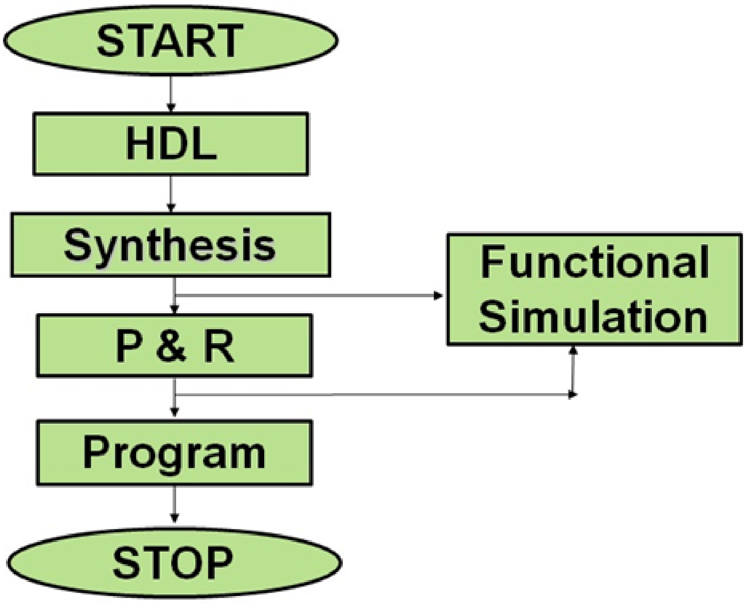


Figure 6.2: Flow in Tool

6.4 Hardware Analysis

In hardware analysis, Signal conditioning and Drive electronics are main.

6.4.1 Signal Conditioning Circuit Analysis

Starting from 5°C to 105°C are analysed. Signal conditioning circuit output is measured. It is observed that all output voltages are positive. Due to positive values DAC can process that. Voltage level values at different set-point are also measured. For experiment purpose set-point values are taken in to account are 45°C, 65°C, 85°C and 110°C.

Delta Error is a difference between set-point voltage and measured voltage. Delta error graphs are produced for different set-points. These all are gives linearity. Linearity results into good stability and accuracy.

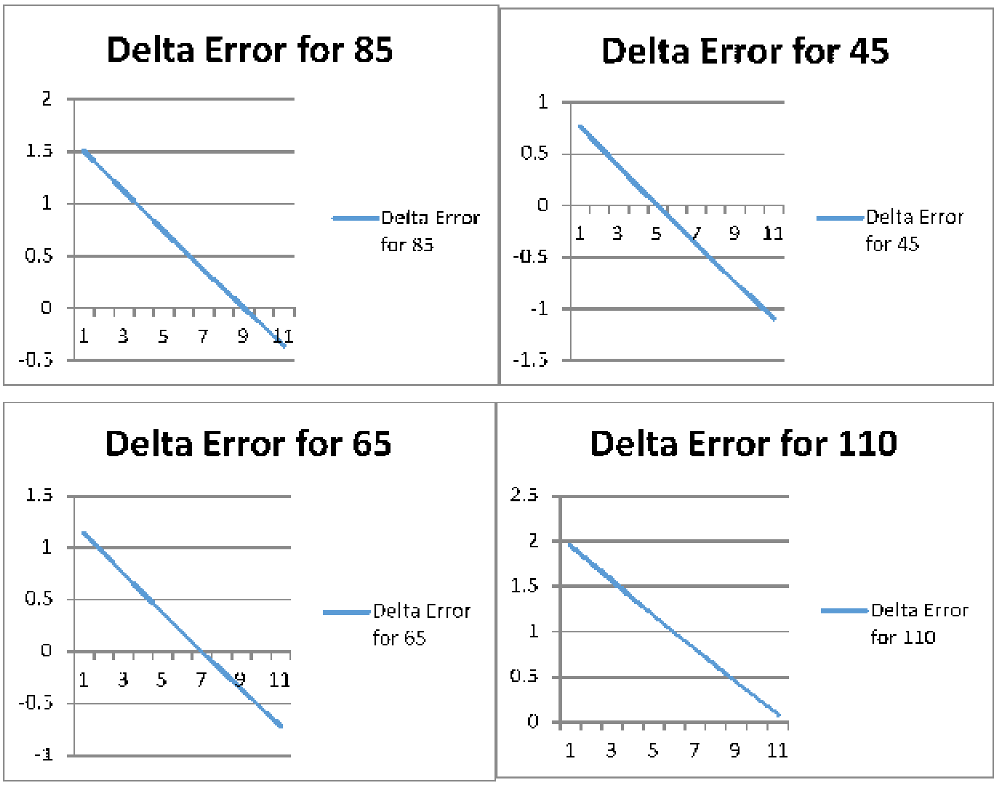
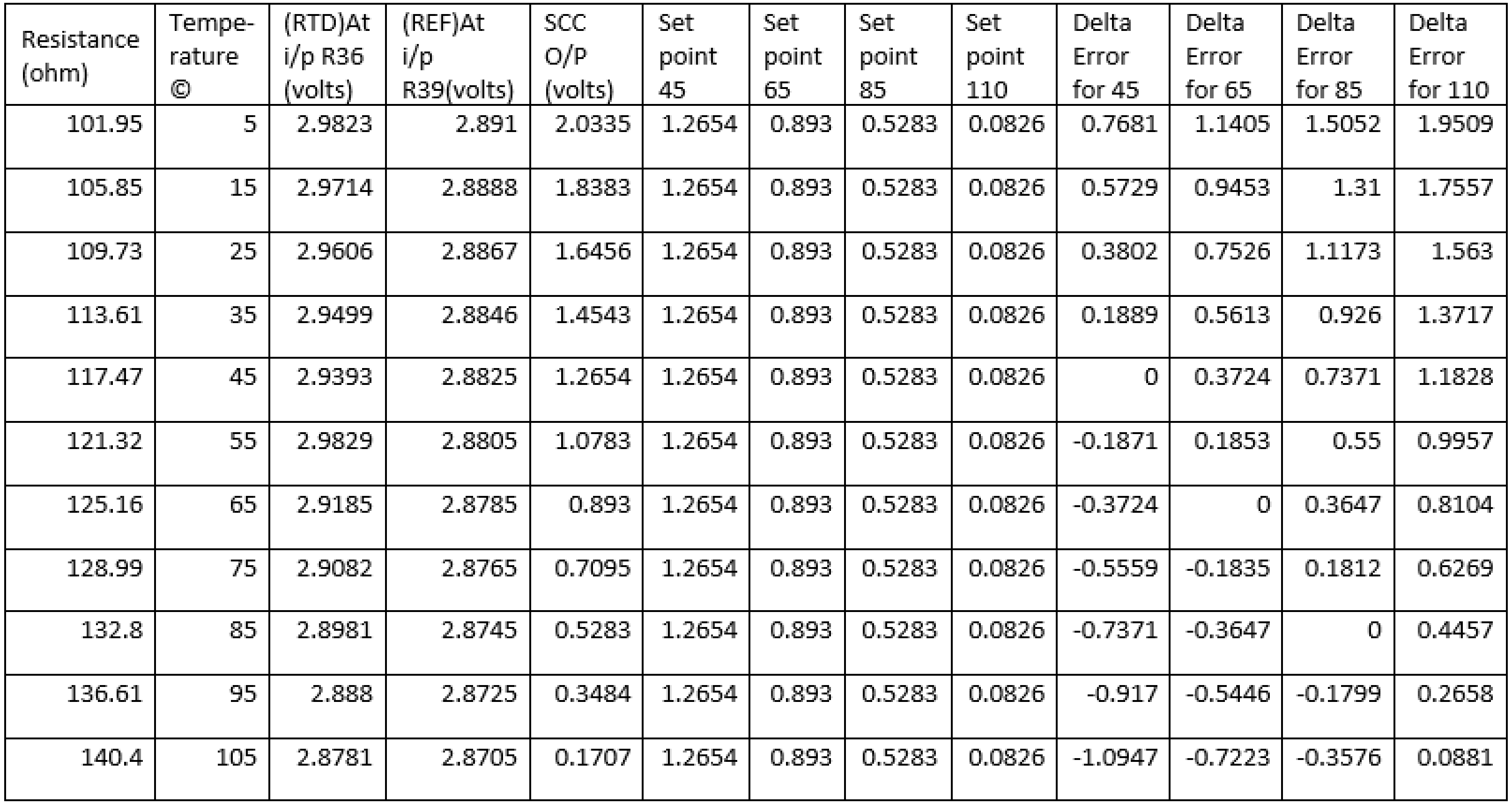


Figure 6.3: Graphical Representation

6.4.2 Drive Electronics Analysis

DAC is faded by Signal conditioning circuit, but voltage level of signal conditioning circuit is not proper. So, drive electronics is intermediate stage of DAC and SCC. Drive electronics is a self-bias transistor. It will be off when input is 0V. It will gives response as per requirement when input is 3.3V. Accordingly, heat is produced by heater and base plate of heater will cool and hot.



Resistance (ohm)	Tempe-rature ©	(RTD)At i/p R36 (volts)	(REF)At i/p R39(volts)	SCC O/P (volts)	Set point 45	Set point 65	Set point 85	Set point 110	Delta Error for 45	Delta Error for 65	Delta Error for 85	Delta Error for 110
101.95	5	2.9823	2.891	2.0335	1.2654	0.893	0.5283	0.0826	0.7681	1.1405	1.5052	1.9509
105.85	15	2.9714	2.8888	1.8383	1.2654	0.893	0.5283	0.0826	0.5729	0.9453	1.31	1.7557
109.73	25	2.9606	2.8867	1.6456	1.2654	0.893	0.5283	0.0826	0.3802	0.7526	1.1173	1.563
113.61	35	2.9499	2.8846	1.4543	1.2654	0.893	0.5283	0.0826	0.1889	0.5613	0.926	1.3717
117.47	45	2.9393	2.8825	1.2654	1.2654	0.893	0.5283	0.0826	0	0.3724	0.7371	1.1828
121.32	55	2.9829	2.8805	1.0783	1.2654	0.893	0.5283	0.0826	-0.1871	0.1853	0.55	0.9957
125.16	65	2.9185	2.8785	0.893	1.2654	0.893	0.5283	0.0826	-0.3724	0	0.3647	0.8104
128.99	75	2.9082	2.8765	0.7095	1.2654	0.893	0.5283	0.0826	-0.5559	-0.1835	0.1812	0.6269
132.8	85	2.8981	2.8745	0.5283	1.2654	0.893	0.5283	0.0826	-0.7371	-0.3647	0	0.4457
136.61	95	2.888	2.8725	0.3484	1.2654	0.893	0.5283	0.0826	-0.917	-0.5446	-0.1799	0.2658
140.4	105	2.8781	2.8705	0.1707	1.2654	0.893	0.5283	0.0826	-1.0947	-0.7223	-0.3576	0.0881

Figure 6.4: Signal Conditioning Circuit Results

Chapter 7

Conclusion and Future Work

It is concluded by simulations and literature survey that, Fuzzy logic have better response than available classical control logic. Design steps have explained by relevant example. Simulations for fuzzy logic for different plants have done and results are observed that, there are no overshoots in Fuzzy Logic. As Fuzzy Logic is independent of plant model, it can control the system without knowledge of the system behavior. Variances of fuzzy system have also simulated here. Design with different set-points and different membership functions have simulated and results are observed that, Fuzzy logic is adaptive in nature and manual tuning is not required. Conversion of floating to fix point have carried out. As order of input increase, complexity have also increased. Single Input Fuzzy system is implemented.

Fuzzy Logic implementation extend for two inputs and three inputs. Optimization of rule-sets can also be an extension of this work. Accuracy level of control is another part of future work. As, most processes are based on trial and error, find out general procedure is one more area of future for this work.

References

[1] Essam Natsheh and Khalid A. Buragga, *"Comparison between conventional and fuzzy logic PID controllers for controlling DC motors"*, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 5, September 2010.

[2] P. Singhala, D.N. Shah, B. Patel, *"Temperature control using Fuzzy Logic"*,International Journal of Instrumentation and Control Systems (IJICS) Vol. 4, No. 1 , January 2014 .

[3] Rahul Malhotra et. al, *"Boiler flow control using PID and Fuzzy logic controller"*,IJCSET/ July 2011/ Vol 1, Issue 6, 315-319.

[4] 4. Jay Kumar, Richa Kapoor, *"Comparative analysis of room temperature control using Fuzzy logic and PID"*,Advance in Electronic and Electric Engineering, ISSN 2231-1297, Vol. 3, Number 7 (2013), PP. 853-858.

[5] Du Fuyin and Du Weifeng, *"Design of a three-input fuzzy logic controller and the method of its ruler reduction"*, International Symposium on Information Processing, Auguest 21-23, 2009.

[6] Gunawan Dewantoro,*"Robust speed-controlled permanent magnet synchronous motor drive using fuzzy logic controller"*,IEEE International Conference on fuzzy systems, June 27-30, 2011, Taipei, Taiwan.

[7] 7. B.Vasu Murthy, Y.V. Pavan Kumar,*"Fuzzy logic Intelligent controlling concept in Industrial Furnace temperature process control,"* IEEE International Conference on Advanced Communication Control and Computing Technologies, 2012.

[8] 8. M. Anbarasan, S.J. Suji Prasad,*"Modified PID controller for Avoiding Overshoot in Temperature of Barrel Heating system"*,978-1-4673-53014/13/2013 IEEE.

[9] 9. I.Eminoglu, I.H. Altas,*"The effects of the number of rules on the output of a fuzzy logic controller employed to a PM dc motor"*,Computers and electrical Engineering 24 (1998) 245-261, PERGAMON .

[10] Shabiul Islam, Nowshad Amin,*"FPGA Realization of Fuzzy Temperature Controller for Industrial Application"*,WSEAS TRANSACTIONS on SYSTEMS and CONTROL ,June 16, 2007.

[11] CHUEN CHIEN LEE, STUDENT MEMBER, IEEE, *"Fuzzy logic in control systems: fuzzy logic controller part one-two"*, IEEE transaction on systems, MAN, and CYBERNETICS, VOL. 20. NO. 2, March/April 1990.

[12] M.D. Hanamane, R.R. Mudholkar, *"Implementation of Fuzzy temperature control using microprocessor"*, Journal of Scientific and Industrial Research, Vol. 65, February 2006 , pp. 142-147.

[13] Du Fuyin and Du Weilfeng, *"Design of a three- input fuzzy logic controller and the method of its rules reduction"*, International Synoposium on Information Processing (ISIP 2009), August 21-23, 2009, pp. 051-053.

[14] Thierry M. Guerra, Antonio Sala, Kazuo Tanaka, *"Fuzzy control turns 50: 10 years later, Fuzzy sets and systems"*, Science Direct, ELSEVIER, 6 May 2015 (Article in press).

[15] Manikandan P., Geetha M., *"Real-time implementation and performance analysis of an Intelligent Fuzzy logic controller for level process"*, IEEE-31661, 4th ICCCNT 2013, July 4-6 ,2013, Tiruchengode.

[16] S.R. Vaishnav, Z.J. Khan, *"Design and Performance of PID and Fuzzy logic controller with smaller rule set for higher order system"*, Proceeding of the world congress on engineering and computer science 2007, October, 24-26, 2007, San Francisco, USA.