

Speed Control of DC Motor using Fuzzy Logic Controller by PCI 6221 with MATLAB

Anil Kumar Rajagiri¹, Sandhya Rani MN², Syed Sarfaraz Nawaz³, and Suresh Kumar T⁴

^{1,2}Assistant Professor, Department of EEE, GRIET, Hyderabad, India

³Associate Professor, Department of EEE, GRIET, Hyderabad, India

⁴Professor, Department of EEE, GRIET, Hyderabad, India

Abstract. This thesis demonstrates the importance of Fuzzy Logic Controller. The operation of a DC motor is performed using Fuzzy Logic Controller (FLC) in MATLAB environment. Fuzzy Logic is one of the most successful applications of fuzzy set in which the variables are linguistic rather than numeric. A Fuzzy Logic Controller (FLC) is based on a set of control rules (fuzzy rules) among linguistic variables. The proposed fuzzy controller results in a better response compared to the normal response of DC motor. This thesis consists of two parts; software and hardware implementation. The software part aims to design and develop a Fuzzy Logic Controller in MATLAB Simulink. The hardware Part Consist of DC motor Driver and PCI 6221. The DC drive is used to convert AC voltage into variable DC voltage PCI 6221 is used as the hardware interface between Hardware and Software.

Keywords: PCI 6221 68 pin Card, Fuzzy Logic Controller, DC motor, DC Drive, Proximity Sensor

1. INTRODUCTION

Recent developments in science and technology provide a wide range scope of applications of high performance DC motor drives in area such as rolling mills, chemical process, electric trains, robotic manipulates and the home electric appliances. They require controllers to perform tasks. Hence, a fuzzy based DC motor control system was designed using MATLAB.

Almost every industry and household has motors used in their equipment or appliances. Motors that are often controlled by computers have also become an essential part of many motion control systems[1].

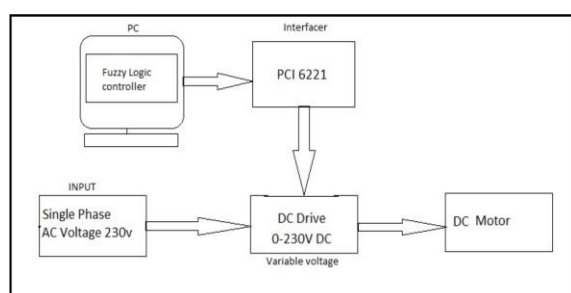


Fig 1. Block Diagram approach of the model developed

2 FUZZY CONTROL SYSTEM

Fuzzy Logic idea is similar to the human being's feeling and inference process. The output of a fuzzy controller is derived from fuzzifications of both inputs and output using the associated membership functions. A crisp input will be converted to the different members of the associated membership functions based on its value. From this point of view, the output of a fuzzy logic controller is based on its memberships of the different membership functions, which can be considered as a range of inputs. Fuzzy ideas and fuzzy logic are so often utilized in our routine life that nobody even pays attention to them. For instance, to answer some questions in certain surveys, most time one could answer with 'Not Very Satisfied' or 'Quite Satisfied', which are also fuzzy or ambiguous answers. Exactly to that degree is one Satisfied or Dissatisfied with some service or product for those surveys? These vague answers can only be created and implemented by human beings, but not machines. Is it possible for a computer to answer those survey questions directly as a human being did?

It is impossible. Computers can only understand either '0' or '1' and 'HIGH' or 'LOW'. Those data are called crisp or classic data and can be processed by all machines. Is it possible to allow computers to handle those ambiguous data with the help of a human being? If so, how can computers and machines handle those vague data? The answer to the question is yes. But to answer the second question, we need some Fuzzy Logic techniques and knowledge of fuzzy inference system. Applications in industrial manufacturing, automatic control, automobile production, Hospitals, elevator control system, automatic transmission control, washing machine etc[3]. Fuzzy logic techniques have been widely applied in all aspects in today's society[2].

2.1 Knowledge Base

The knowledge base can be divided into two sub-blocks namely the Data Base and Rule Base. The data base consists of the information required for Fuzzifying the crisp input and later Defuzzifying the fuzzy outputs to a crisp output. It consists of the membership functions for various fuzzy variables or sets used in the controller design. The rule base consists of a set of rules, which are usually formulated from the expert knowledge of the system. The rules are typically of the form "If...., then...." rules. The antecedent part of the rule may be a simple statement or a compound statement using connectives like "and", "or" etc. the consequent part may contain a fuzzy set (Mamdani type and Tsukamoto type) or a linear or a quadratic function of the crisp input variables (Sugeno type). The knowledge base is the heart of fuzzy logic based system it has to be designed without most care and require a lot of expertise in the knowledge of the system into which fuzzy logic controller is being incorporated[1].

2.2 Fuzzy Inferencing

The process of fuzzy reasoning is incorporated into what is called a Fuzzy Inferencing System. It is comprised of three steps that process the system inputs to the appropriate system outputs[4].

These steps are

- 1) Fuzzification
- 2) Rule Evaluation
- 3) Defuzzification

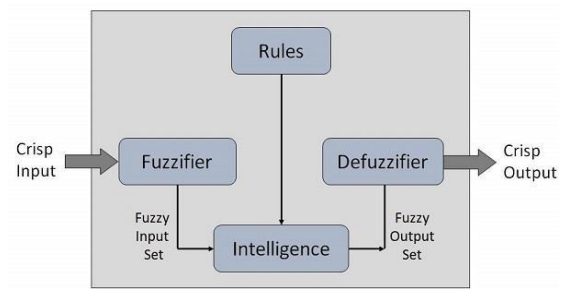


Fig 2. Structure of Fuzzy Logic Controller

2.2.1 Fuzzification

Fuzzification is the first step in the fuzzy inferencing process. This involves a domain transformation where crisp inputs are transformed into fuzzy inputs. Crisp inputs are exact inputs measured by sensors and passed into the control system for processing, such as temperature, pressure, rpm's, etc..

2.2.2 Defuzzification

Defuzzification involves the process of transposing the fuzzy outputs to crisp outputs.

There are a variety of methods to achieve this

- Centre of Area (CoA)
- Center of Sums (CoS)
- Center of Maximum (CoM)
- Mean of Maximum (MoM)

2.2.2.1 Center of Area (CoA)

Fuzzy logic controller first calculate the area under scale membership function and within the range of output variable

$$COA = \frac{\int_{x_{min}}^{x_{max}} f(x).x dx}{\int_{x_{min}}^{x_{max}} f(x) dx}$$

Where CoA is the center of area x is the value of the linguistic variable, and x_{min} and x_{max} represent the range of the linguistic variable.

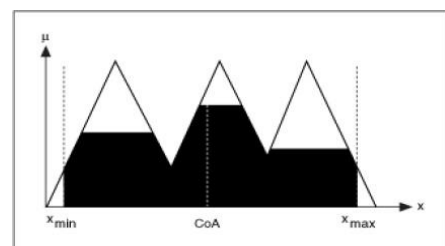


Fig 3. CoA Defuzzification method

2.2.2.2 Center of Sums

Fuzzy Logic Controller calculate the geometric center of area of each membership function and take see weighted average of center of area of all membership function.

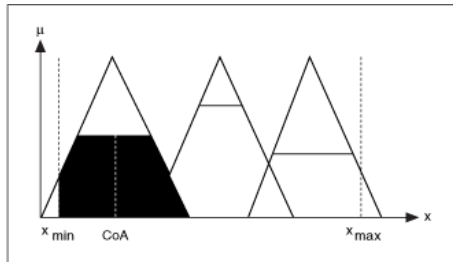


Fig 4. CoS Defuzzification method

$$x_{\text{final}} = \frac{(CoA_1 \text{area}_1 + CoA_2 \text{area}_2 + \dots + CoA_n \text{area}_n)}{\text{area}_1 + \text{area}_2 + \dots + \text{area}_n}$$

Where CoA_n is the geometric center of area of area of the scaled membership function n and area_n is the area of the scaled membership function n .

2.2.2.3 Center of Maximum

In center of maximum method the fuzzy logic controller determines the typical numerical value for each scaled membership function and takes mean of numerical value corresponding to degree of membership function. when this is then placed in an external magnetic field, it will experience a force proportional to the current in the conductor, and to the strength of the external magnetic field.

2.3.1 Voltage and power equation of DC motor

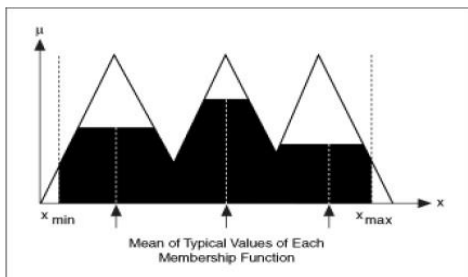


Fig 5. CoM defuzzification method

$$x_{\text{final}} = \frac{x_1 \mu_1 + x_2 \mu_2 + \dots + x_n \mu_n}{\mu_1 + \mu_2 + \dots + \mu_n}$$

Where x_n is the typical numerical value for the scaled membership function n and μ_n is the degree of membership at which membership function n was scaled.

2.2.2.4 Mean of Maximum

In mean of maximum method the scale fuzzy logic controller identify the membership function with the greater degree of membership function.

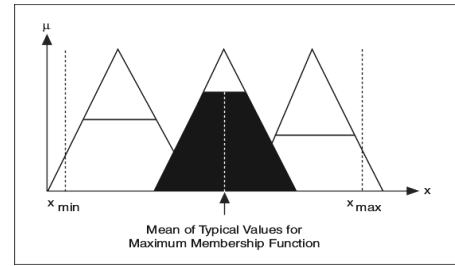


Fig 6. MoM Defuzzification method

2.3 DC Motor

Direct current (DC) motors are perhaps the most basic type of electrical motors. In any electric motor, operation is based on simple electromagnetism. A current-carrying conductor generates a magnetic field;

$$V = E_b + I_a R_a$$

If we multiply the above equation by I_a , we will get

$$VI_a = E_b I_a + I_a^2 R_a$$

VI_a = electrical power supplied to the motor

$E_b I_a$ = electrical equivalent of the mechanical power produced by the motor

$I_a^2 R_a$ = power loss taking place in armature winding

Thus,

$$E_b = VI_a - I_a^2 R_a$$

Input power – power loss

Thus, $E_b I_a$ = Gross mechanical power produce by the motor = rpm

2.3.2 Torque equation of DC motor

Mechanical power required to rotate the shaft

$$\text{on mechanical side} = T\omega \tag{1}$$

T = Torque in Newton – meter

ω = angular velocity in Radian/second

Gross mechanical power produce by the motor

$$\text{on electrical side} = E_b I_a \tag{2}$$

E_b = back emf in volts

I_a = armature current in ampere

Equating equation (1) and (2), we get

$$E_b I_a = T \omega \tag{3}$$

$$\omega = \frac{2\pi N}{60} \dots \dots \dots \left\{ \frac{2\pi N}{60} = \text{speed in rpm} \right.$$

And

$$E_b = \frac{N \Phi Z P}{A 60}$$

Thus, equation (3) become

$$\frac{N \Phi Z P}{A 60} I_a = T \frac{2\pi N}{60}$$

$$T = \frac{P \Phi Z I_a}{2\pi A} = \frac{0.159 P \Phi Z I_a}{A} = \left(\frac{0.159 P Z}{A} \right) \Phi I_a$$

P,Z and A are constant, hence we can say

$$T \propto \Phi I_a$$

Thus torque produce by the DC motor is proportional to the main field flux Φ and armature current I_a

Governing Equations

1. $E_b = \frac{P \Phi N Z}{60 A}$
 (Where, P = no. of poles, Φ = flux /pole, N = speed in rpm, Z = no. of Armature conductors, A = parallel paths)
2. $E_b = V - I_a R_a$
 Thus,
3. $N = \frac{E_b 60 A}{P \Phi Z}$ (A,P and Z are constants, hence)
4. $\frac{N \propto K E_b}{\Phi}$ (where, K = constant)

2.4 Proximity Sensor

Proximity sensor is a sensor able to detect the presence of nearby objects without any physical contact. A proximity sensor often emits an electromagnetic field or

a beam of electromagnetic radiation (infrared, for instance), and looks for changes in the field or return signal. The object being sensed is often referred to as the proximity sensor's target. Proximity sensors are also used in machine vibration monitoring to measure the variation in distance between a shaft and its support bearing. This is common in large steam turbines, compressors, and motors that use sleeve-type bearings.



Fig 7. Proximity Sensor

2.5 NI PCI-6221 (68 Pin)

NI PCI-6221 is used for acquiring the signals from the speed sensor and to generate the signal applied to the DC Drive. It is M Series Data Acquisition Device that provides the data Acquisition for some application of portable measurement. Data acquisition card is used as the hardware device. It has a total of 68 pins for various input/output purposes. Various characteristics of the card are[5]:

- 16-Bit, 250kS/s, 16 Analog Inputs
- Two 16-bit analog outputs (833kS/s)
- 24 digital I/O



Fig 8. NI PCI 6221 Card

The following are the specifications:

Table 1. PCI 6221 specifications

| | | |
|------------------|---|------------------------------------|
| Analog Inputs | : | 16 |
| Analog Outputs | : | 02 |
| Digital I/O pins | : | 24 |
| Resolution | : | 16 bit |
| Input Range | : | $\pm 0.2, \pm 1V, \pm 5V, \pm 10V$ |
| Output Range | : | $\pm 10V$ |

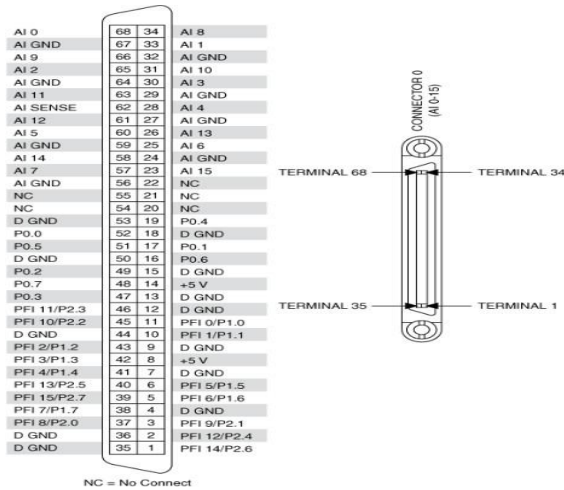


Fig 9. PCI 6221 Pin Diagram

NI CB-68LP is the daughter board used along with the PCI card. It is a low-cost termination accessory with 68 screw terminals for easy connection.



Fig 10. NI CB-68LP and its connector cable

2. Designing FLC in MATLAB

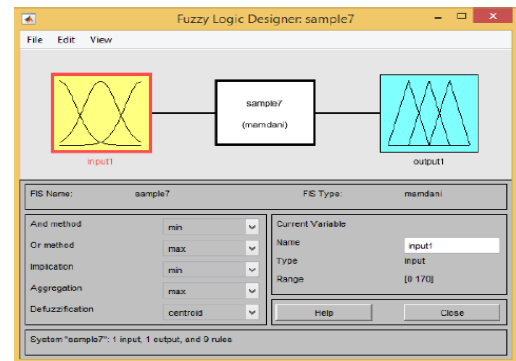


Fig 11. Fuzzy Block

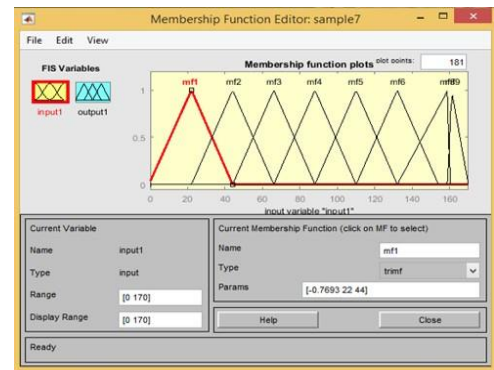


Fig 12. Fuzzy input membership function

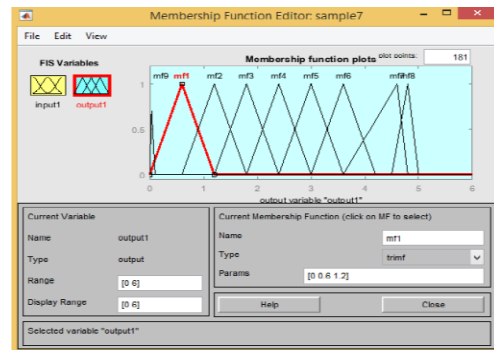


Fig 13. Fuzzy Output Membership Function

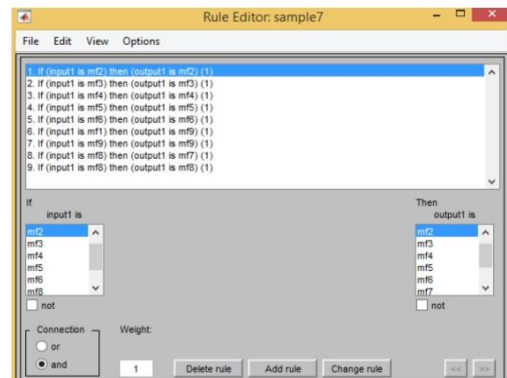


Fig 14. Fuzzy Rules set

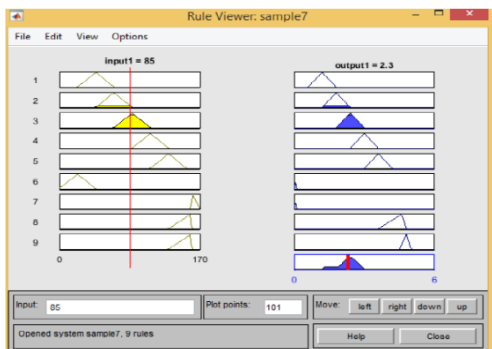


Fig 15. Fuzzy Rule Viewer

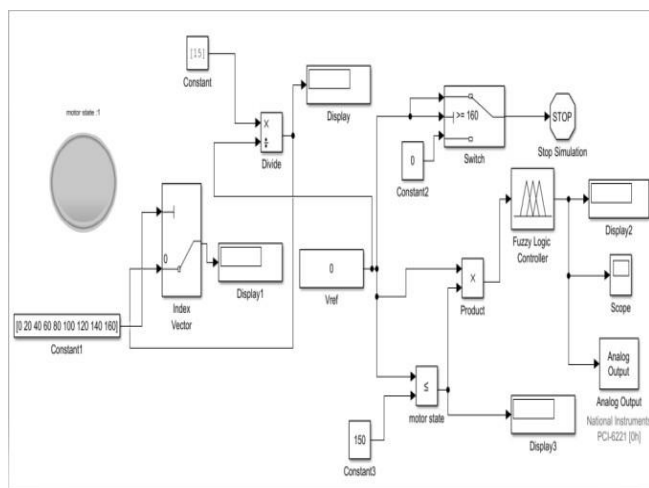


Fig 16. Circuit Diagram



Fig 17. Hardware

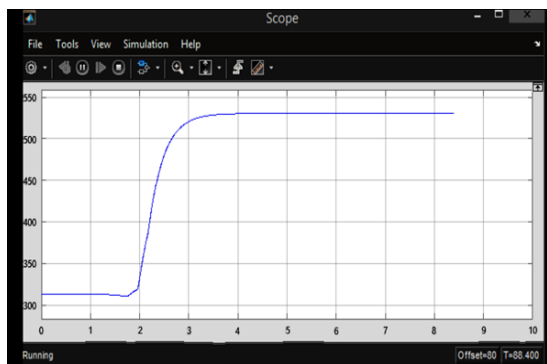


Fig 18. DC Machine speed

Table 2. DC Drive output with respective Fuzzy

| Fuzzy Input (Voltage) | DC Drive Output (Voltage) | Speed (RPM) |
|-----------------------|---------------------------|-------------|
| 0 | 0 | 0 |
| 20 | 31.53 | 265.7 |
| 40 | 47.37 | 414.9 |
| 60 | 65.05 | 579.1 |
| 80 | 86.95 | 782.9 |
| 100 | 105 | 952.3 |
| 120 | 125.9 | 1149 |
| 140 | 145.6 | 1346 |
| 160 | 159.3 | 1496 |

3. CONCLUSION

The fuzzy controller for the running of the DC motor is designed in MATLAB. The design of the FLC has been explained. The results of experiment on the real plant demonstrate that the proposed Fuzzy Logic Controller the response is very good and it is robust, faster and flexible. It shows control the speed. By using FLC application, shorter settling time can be achieved by tuning the control rules, membership function and universe of disclosure of the output variable. This presents an idea of speed control of DC motor with simple software application with good hardware results.

References

1. NPTEL Video lectures, Electrical Engineering-Intelligence System and control-introduction of Fuzzy Logic Control.
2. George J. Klir, Bo Yuan Fuzzy Sets and Fuzzy Logic.
3. J. Harries, An Introduction to Fuzzy Logic Applications.
4. Sivanandam, SN., Sumathi, S., Deepa, S. N, Introduction to Fuzzy Logic using MATLAB.
5. NI PCI 6221 (68 PIN) the M series user manual available at ni.com/manuals.