A GENERAL PURPOSE FUZZY LOGIC CONTROL ALGORITHM
DEVELOPED TO CONTROL OF MAGNETIC SUSPENSION OF IRON SPHERE
BY FUZZY LOGIC CONTROL SYSTEM

Osman GÜRDAL, Zeki ERDEM

Gazi University, Technical Education Faculty, Electrical & Control Education
Department, ANKARA, TURKEY

Abstract

Magnetic suspension system of iron sphere to be suspended against earth gravity shows a visual demonstration of many principles in educational programs. The purpose of this paper is to keep the gap between electromagnet and iron sphere in desired position by fuzzy logic control system. The control system has a conventional electromagnet with 12.5 mm diameter iron sphere and optical transducers consisting of 1 infrared diode and 3 photo-transistors giving outputs as position information of iron sphere to a PC via A/D converter and RS232 port. A general purpose fuzzy logic algorithm is developed for the controlling the position of iron sphere. The control of the electromagnetic system is first achieved by PD control compensation to get the rules of fuzzy logic control system and later the fuzzy rules obtained are used in fuzzy logic control system without using PD control. The fuzzy logic control system an advantage over PD controller. Observations have shown that, when a deviation of iron sphere from its original fixed position by external effects occurs, fuzzy logic control brings the iron sphere to its original position quicker than PD controller.

Key words: Suspension, PD control, fuzzy logic control.

Özet

Yerçekimine karşı demir kürenin manyetik süspansiyon sistemi, eğitim ile ilgili programlarda çoğu presipler için görsel bir bir demonstrasyon sağlamaktadır. Bu makalenin amacı, bulanık mantık kontrol sistemi ile elektromiknatis ile demir küre arasındaki boşlugu istenen pozisyonda tutmaktır. Kontrol sistemi 12.5 mm çapında bir demir küre ile alınsın bir elektromiknatis ve 1 kizilötesi diyot ve A/D dönüştürücü ve RS232 port ile bir PC ye demir kürenin pozisyon bilgisini çıkısı olarak veren 3 foto-transistörden oluşan optik dönüştürücülerden oluşmaktadır. Genel amaçlı
The principle of suspension system using controlled DC electromagnets is shown in figure 1. The position of the iron sphere is sensed by an optical transducer comprising an infrared photo diode and photo transistor; the quantity of light falling on the photo transistor varies with the position of the iron sphere, so the voltage output is a measure of the sphere position. Photo transistor output is compared with a reference voltage which corresponds to the desired position of the iron sphere; the difference represents a position error. This error voltage is passed to a DC amplifier which controls the current in the electromagnet, so that any displacement of the iron sphere from its desired position causes a corrective change in the electromagnet current.

Figure 2. shows a practical implementation of the control circuit. The optical position transducer consists of and infrared LED D1 and the phototransistor Q1. The position of the iron sphere controls the amount of infrared light falling on the phototransistor, and thus varies the
emitter current, which in turn varies the voltage drop in the emitter load resistor RV1.

![Figure 1. Suspension system with its electrical and mechanical parameters.](image)

![Figure 2. Electromagnetic suspension analog controller circuit.](image)

The compensator circuit is the critical part of the system. The proportional control only itself causes the iron sphere tend to oscillate like a mass on a spring because of the inductance of the electromagnet coil. Any oscillation tends to grow in amplitude may cause the electromagnet loses control of the iron sphere. To damp this oscillation, the capacitor C1 adds a control term proportional to the velocity. This derivative term counteracts the effect of inductance. It causes a change in the electromagnet current proportional to the velocity, but in a direction which opposes the motion; thus the velocity will be reduced, and the amplitude of the oscillation will keep decreasing until the iron sphere ceases to move (2).

### 2.1 Electromagnetic System

Figure 1 shows the principal components of the magnetic system. Flux plot and the inductance graph, \( L_x \), in figure 3, is extracted from Gemini software by simulating the electromagnet with different distances of iron sphere.
\[ I = \sqrt{2Mg \frac{a}{L_x}} \]  

where \( M \) is mass of iron sphere, \( g=9.81 \text{ m/s}^2 \).

### 2.2 Dynamic and Control System

The transfer function which relates the coil current and iron sphere position.

\[ \frac{X(s)}{I(s)} = -\frac{2g}{I}s^2 - \omega_n^2 \]  

where, \( \omega_n = \sqrt{\frac{g}{a}} \) is natural frequency.

The classical form of PD control with a damping term by means of a lead transfer function of the form

\[ \frac{V_2}{V_1} = K_p (1 + T_d s) \]  

The block diagram with PD control and its associated root locus is shown in figure 4. The value of \( K_p \) may be chosen for a suitable value of damping ratio and natural frequency.

![Figure 4](image)

Figure 4. (a) Block diagram and, (b) root locus of compensated system.
3. Electromagnetic Suspension by FLC System

Motivated by Zadeh (3) and explored and validated by Mamdani, fuzzy logic control (FLC) has been used successfully in numerous control systems (4). Now ‘fuzzy’ has become a household word that is universally understood to mean smart, as having the ability to think like a human (5). Fuzzy logic, which is the logic on which fuzzy control is based, is much closer in spirit to human thinking and natural language than the traditional logical systems. Basically, it provides an effective means of capturing the approximate, inexact nature of the world. The essential part of the FLC is a set of linguistic control rules related by the dual concepts of fuzzy implication and the compositional rule of inference (6, 7).

FLC systems are preferable under the circumstances below (8):

- When input and output parameters have uncertainties.
- When a complex model is difficult to be modelled mathematically with high accuracy (9).
- In non-linear systems (10).
- In real time controls and the systems with many parameters.

If there is information about the problem or the information is obtained during the design stage, it is useful to use FLC in control systems. When a mathematical model is not necessary and the dynamic of the system is described as linguistic control rules, the design of FLC is achieved in less time and the cost is reduced. In addition, the systems requiring very complex techniques find the FLC attractive, because constituting a classical model is very costly. In the basic systems, the period of design procedure is highly shortened. As the FLC system uses linguistic control rules, there is no need a system to be modelled mathematically.

In the suspension system, since the force of attraction depends on the current, the system would have a spring-like action; for small displacements, the restoring force would be proportional to the displacement, so the iron sphere would tend to oscillate like a mass on a spring. In itself this behaviour would be a nuisance, but not damaging. The problem is the inductance of the electromagnet coil, which introduces a phase lag, or time delay, between a change in the iron sphere position and a corrective change in the electromagnet current. This lag has the effect of making the system unstable, so that any oscillation tends to grow in amplitude until the electromagnet loses control of the iron sphere. There is complex mathematical process to make the system stable. So the FLC provides simplicity.

A FLC system have four principal components: a fuzzification interface, a knowledge base, decision making logic and a defuzzification interface as shown in figure 5. The fuzzification interface; measures the values of input variables,
performs a scale mapping that transfers the range of values of input variables into corresponding universes of discourses, performs the function of fuzzification that converts input data into suitable linguistic values which may be viewed as labels of fuzzy sets (6).

![Block diagram of fuzzy logic control system.](image)

In the knowledge base; the data base provides necessary definitions which are used to define linguistic control rules and fuzzy data manipulation in an FLC, the rule base characterizes the control goals and control policy of the domain experts by means of a set of linguistic control rules.

The Decision making logic is the kernel of an FLC; it has the capability of simulating human decision-making based on fuzzy concepts and of inferring fuzzy control actions employing fuzzy implication and the rules of inference in fuzzy logic.

The defuzzification interface performs a scale mapping which converts the range of values of output variables into corresponding universes of discourse, yields a non-fuzzy control action from an inferred fuzzy control action.

The FLC system needs some experience and information about the system to be controlled during design procedure. Otherwise, the FLC is designed by logical estimations about the system but this is time consuming. Firstly, the prototype of suspension system is run as an analog controller. After many experiments, behaviour of the system is tried to be understood and the fuzzy control rules is obtained from the PD controller.

The FLC of the suspension of iron sphere is done by a PC. For the FLC, an 8 bits based general purpose software extracting the fuzzy rules by C programming language is developed. The fuzzy rules extractor has 16 inputs and 8 outputs with 30000 rules. The communication with PC is done 12 bits A/D and D/A converters. The suspension system has three optical infrared receiving transistors as shown in figure.
6. The position of iron sphere is converted to voltage by three infrared optical transistors as shown in table 1. and figure 8. as graphs. From these position information, triangular shaped membership functions are determined as shown in figure 7. Triangular shape is preferred for mathematical simplicity. The analog values of transistor outputs must be converted to decimal values between 0-255 or hexadecimal between 0-FF. Now the next step is the fuzzification.

![Figure 6. Optical position transducers sensing the position of the iron sphere.](image)

![Table 1. The voltage outputs of three photo-transistors changing with distance, \( x \).](table)

<table>
<thead>
<tr>
<th>( x ) mm</th>
<th>Transistor outputs, volt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>T1</td>
</tr>
<tr>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>0.20</td>
</tr>
<tr>
<td>1.5</td>
<td>0.20</td>
</tr>
<tr>
<td>2</td>
<td>0.29</td>
</tr>
<tr>
<td>3</td>
<td>0.42</td>
</tr>
<tr>
<td>4</td>
<td>0.59</td>
</tr>
<tr>
<td>5</td>
<td>1.50</td>
</tr>
<tr>
<td>5.5</td>
<td>2.85</td>
</tr>
<tr>
<td>6</td>
<td>5.72</td>
</tr>
<tr>
<td>6.5</td>
<td>6.57</td>
</tr>
<tr>
<td>7</td>
<td>7.89</td>
</tr>
<tr>
<td>8</td>
<td>7.95</td>
</tr>
<tr>
<td>8.5</td>
<td>8.00</td>
</tr>
<tr>
<td>9</td>
<td>8.00</td>
</tr>
<tr>
<td>10</td>
<td>8.00</td>
</tr>
<tr>
<td>11</td>
<td>8.00</td>
</tr>
<tr>
<td>12</td>
<td>8.00</td>
</tr>
<tr>
<td>12.5</td>
<td>8.00</td>
</tr>
<tr>
<td>13</td>
<td>8.00</td>
</tr>
<tr>
<td>13.5</td>
<td>8.00</td>
</tr>
<tr>
<td>14</td>
<td>8.00</td>
</tr>
<tr>
<td>15</td>
<td>8.00</td>
</tr>
</tbody>
</table>

![Figure 7. Diagrammatic representation of fuzzy distances or membership functions. The distances of iron sphere from the core are linguistic variables with three terms ‘low’, ‘medium’ and ‘high’.](image)
Figure 8. The graphs of voltage outputs of three photo-transistors changing with distance, $x$.

3.1 Fuzzification

In FLC applications, the observed data are usually crisp. Since the data manipulation in an FLC is based on fuzzy set theory, fuzzification is necessary during an earlier stage. So a fuzzification operator converts a crisp value into a fuzzy singleton within a certain universe of discourse. Basically, a fuzzy singleton is a precise value and hence no fuzziness is introduced by fuzzification in this case. This strategy has been widely used in fuzzy control applications since it is natural and easy to implement. It interprets an input $x_0$ as a fuzzy set $A$ with the membership function $\mu_A(x)$ equal to zero except at the point $x_0$, at which $\mu_A(x)$ equals one. Below is the example fuzzification process of the triangular shapes shown in figure 7. and table 1.

The fuzzification for the output of transistor 1:

Low region:
$\mu(x) = 1$  $\infty \leq x \leq 0.75$
$\mu(x) = (0.75-x)/(0.75-0)$  $0 \leq x \leq 2.85$

Medium region:
$\mu(x) = (0.75-x)/(0.75-0.75)$  $0.75 \leq x \leq 2.85$

High region:
$\mu(x) = (6.85-x)/(6.85-2.85)$  $2.85 \leq x \leq 6.85$

The fuzzification for the output of transistor 2:

Low region:
$\mu(x) = 1$  $\infty \leq x \leq 0.80$
$\mu(x) = (0.80-x)/(0.80-0)$  $0 \leq x \leq 2.80$

Medium region:
$\mu(x) = (0-x)/(0-0.80)$  $0 \leq x \leq 0.80$
$\mu(x) = (2.80-x)/(2.80-0.80)$  $0.80 \leq x \leq 2.80$

High region:
$\mu(x) = (0.80-x)/(0.80-2.80)$  $0.80 \leq x \leq 2.80$
$\mu(x) = (6.70-x)/(6.70-2.80)$  $2.80 \leq x \leq 6.70$

The fuzzification for the output of transistor 3:

Low region:
$\mu(x) = 1$  $\infty \leq x \leq 0.78$
$\mu(x) = (0.78-x)/(0.78-0)$  $0 \leq x \leq 2.93$

Medium region:
$\mu(x) = (0-x)/(0-0.78)$  $0 \leq x \leq 0.78$
$\mu(x) = (2.93-x)/(2.93-0.78)$  $0.78 \leq x \leq 2.93$

High region:
$\mu(x) = (0.78-x)/(0.78-2.93)$  $0.78 \leq x \leq 2.93$
$\mu(x) = (6.75-x)/(6.75-2.93)$  $2.93 \leq x \leq 6.75$

Table 2. shows all the slopes values to calculate membership weights for the three inputs.

<table>
<thead>
<tr>
<th>Slopes</th>
<th>Input 1</th>
<th>Input 2</th>
<th>Input 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low region left side</td>
<td>255</td>
<td>255</td>
<td>255</td>
</tr>
<tr>
<td>Low region right side</td>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>
### Table 2. The slopes for the three inputs.

<table>
<thead>
<tr>
<th>Medium region left side</th>
<th>4</th>
<th>5</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium region right side</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>High region left side</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>High region right side</td>
<td>255</td>
<td>255</td>
<td>255</td>
</tr>
</tbody>
</table>

Table 2. The slopes for the three inputs.

### 3.2 Data Base and Rule Base

The concepts associated with a data base are used to characterize fuzzy control rules and fuzzy data manipulation in an FLC. These concepts are subjectively defined and based on experience and engineering judgment. In this connection, it should be noted that the correct choice of the membership functions of a term set plays an essential role in the success of an application. A fuzzy system is characterized by a set of linguistic statements based on expert knowledge. The expert knowledge is usually in the form of ‘if-then’ rules, which are easily implemented by fuzzy conditional statements in fuzzy logic. Table 3. is obtained by PD controller. Table 4 shows the 7 regions and the 7 rules for the outputs from the table 3.

<table>
<thead>
<tr>
<th>Output Region No.</th>
<th>Analog values (volt)</th>
<th>Decimal values</th>
<th>Hexa decimal values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.40</td>
<td>70</td>
<td>46</td>
</tr>
<tr>
<td>2</td>
<td>2.50</td>
<td>105</td>
<td>69</td>
</tr>
<tr>
<td>3</td>
<td>2.60</td>
<td>135</td>
<td>87</td>
</tr>
<tr>
<td>4</td>
<td>2.90</td>
<td>165</td>
<td>A5</td>
</tr>
<tr>
<td>5</td>
<td>3.20</td>
<td>195</td>
<td>C3</td>
</tr>
<tr>
<td>6</td>
<td>3.40</td>
<td>225</td>
<td>E1</td>
</tr>
<tr>
<td>7</td>
<td>4.00</td>
<td>255</td>
<td>FF</td>
</tr>
</tbody>
</table>

Table 3. The values of electromagnet current and related base voltages of electromagnet driver transistor with different distances of iron sphere suspended.

<table>
<thead>
<tr>
<th>Rule No</th>
<th>Input 1</th>
<th>Input 2</th>
<th>Input 3</th>
<th>Output in region</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High</td>
<td>High</td>
<td>High</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>High</td>
<td>High</td>
<td>Medium</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>High</td>
<td>High</td>
<td>Low</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>High</td>
<td>Low</td>
<td>Low</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>1</td>
</tr>
</tbody>
</table>

(b)
Table 4. (a) The 7 regions and (b) the 7 rules for the outputs.

### 3.3 Decision Making

Table 5. shows input and output values for the iron sphere suspended with an output (base) voltage 2.9 volt. For the distance of $x=7$ mm, 1. input is high, 2. input is low and medium, and 3. input is low.

<table>
<thead>
<tr>
<th>Input Values</th>
<th>Analog</th>
<th>Decimal</th>
<th>Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Position knowledge</td>
<td>7.89 V</td>
<td>224</td>
<td>High</td>
</tr>
<tr>
<td>2. Position knowledge</td>
<td>1.60 V</td>
<td>45</td>
<td>Low and Medium</td>
</tr>
<tr>
<td>3. Position knowledge</td>
<td>0.25 V</td>
<td>7</td>
<td>Low</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Output voltage or transistor’s base voltage</th>
<th>Analog</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.90 V</td>
<td>185</td>
<td></td>
</tr>
</tbody>
</table>

Table 5. Input and output values for the iron sphere suspended at $x=7$ mm and balance current is 1.096 A.

Below is an example to calculate the membership weights ($mw$) for the input values and regions, the values in table 5. are used:

**Input 1:**
Region high,
$mw = 0 \times (224 - 194) / 255 = 1$

**Input 2:**
Region medium,
$mw = 5 \times (45 - 23) / 255 = 0.43$

Region low,
$mw = 5 \times (7 - 45) / 255 = 0.66$

**Input 3:**
Region low,
$mw = 0 \times (22 - 7) / 255 = 1$

<table>
<thead>
<tr>
<th>Rule No</th>
<th>Input 1 $mw$ region</th>
<th>Input 2 $mw$ region</th>
<th>Input 3 $mw$ region</th>
<th>Output region</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1 High</td>
<td>Medium</td>
<td>1 Low</td>
<td>4$^{th}$</td>
</tr>
<tr>
<td>5</td>
<td>1 High</td>
<td>Low</td>
<td>1 Low</td>
<td>3$^{rd}$</td>
</tr>
</tbody>
</table>

Table 6. Active rules for the input values and minimum values effecting outputs.

In table 6., the 4$^{th}$ active rule effects the output in a weight of 0.43 in region 4 and the 5$^{th}$ active rule effects the output in a weight of 0.66 in region 3.

### 3.4 Defuzzification

The output of the inference process so far is a fuzzy set, specifying a possibility distribution of control action. Since the system is on-line, a non-fuzzy action is usually required. Consequently, one must defuzzy the fuzzy control action (output) inferred from the fuzzy control algorithm (11). The method of center of weights is used to calculate the real output value, $R$:

$$R = \frac{\sum_{i=1}^{n} w_i \mu(w_i)}{\sum_{i=1}^{n} \mu(w_i)} \quad (5)$$

Where, $\mu(w_i)$ is membership function, ($wi$) is output of fuzzy control and $n$ number of rule. An example for the values from table 6., is as below:

$$R = \frac{(165 \times 0.43) + (135 \times 0.66)}{(0.43 + 0.66)} = 147$$

Consequently, the real output value represents a base voltage of 2.85 volt with an electromagnet current of nearly 1 amp. which proves the same position
obtained from the PD control in which the current is 1.096 amp and the iron sphere is at 7 mm distance.

4. FLC Algorithm

An 8 bits based general purpose software extracting the fuzzy rules by C programming language is developed. The fuzzy rules extractor has 16 inputs and 8 outputs with 30000 rules. FLC Algorithm is shown in Appendix B. and Fuzzy control program is given in Appendix A.

5. Conclusions

As a visual classroom demonstration tool, a 12.5 mm diameter with a mass of 11 gram iron sphere is magnetically suspended by PD compensation to get the rules of fuzzy logic control system. Position of iron sphere is converted to position knowledge data by 3 infrared photo-transistor and 1 photo-diode. The position data is then sent to a PC which controls the gap of iron sphere by fuzzy logic control system.

An 8 bits based general purpose software extracting the fuzzy rules by C programming language is developed. The fuzzy rules extractor has 16 inputs and 8 outputs with 30000 rules. The communication with PC is done 12 bits A/D and D/A converters.

The stable suspension of iron sphere by PD control requires a good modelling and time consuming mathematical processing. The fuzzy logic control system an advantage over PD controller. When a deviation of iron sphere from its original fixed position by external effects occurs, fuzzy logic control brings the iron sphere to its original position quicker than PD controller.

In educational terms, future experiments could include a sensorless (or online sensing of coil inductance) PD or fuzzy logic controller with a programming language automatically extracts and constructs the fuzzy rules.

5. References


Appendix A. Fuzzy Logic Control Program

```c
unsigned char fuz_in[16], fuz_member[512], fuz_out[64], singleton[64], rules[30000], if_min, skip, then_part;
unsigned int i, fuzzy, sigma;
unsigned long product;

void calculate_fuzzy() {
    for (i=0; i<64; i++) fuz_out[i] = 0;
    for (i=0; i<1000; i++) {
        if (rules[i] == 0xff) break;
        if (!((rules[i] & 0x80))) /* IF part */
            if (then_part == 1) {
                then_part = 0;
                if_min = 0xff;
                skip = 0;
            }
        if (skip) continue;
        fuzzy = fuz_in[(rules[i] & 0x78)/8];
        if (fuzzy > fuz_member[4*rules[i] +0]) {
            if (fuzzy > fuz_member[4*rules[i] +2]) {
                fuzzy = (fuzzy - fuz_member[4*rules[i] +2]) * fuz_member[4*rules[i] +3];
                if (fuzzy < 0xff) fuzzy = 0;
            } else {
                fuzzy = (fuzzy - fuz_member[4*rules[i] +2]) * fuz_member[4*rules[i] +1];
                if (fuzzy > 0xff) fuzzy = 0xff;
                else fuzzy = (fuzzy + fuzzy) / 2;
            }
        } else fuzzy = 0;
        if (fuzzy < if_min) if_min = fuzzy;
    }
    sigma = 0;
    product = 0;
    for (i=0; i<1000; i++) {
        sigma += fuz_out[i];
        product += singleton[i] * fuz_out[i];
        if (i & 7 == 7) {
            if (sigma) fuz_out[i] = product / sigma;
            else fuz_out[i] = 0;
        }
    }
}

unsigned int adc(cha) {
    unsigned char hb, lb, c;
    outp(0x2F8, cha);
    return (hb << 8) + lb + c;
}
```

outp(0X2FB,0);
for (i=0; i<5; i++)
{
    delay(0.5);
    hb = inp(0X2FC);
}
for (i=0; i<9; i++)
{
    delay(0.5);
    hb = inp(0X2FD);
}
lb = inp(0X2F9);
hb = inp(0X2FA);
asm AND hb, 0FH
return ((hb*256+lb)/16);

void dac( x )
{
    int y;
    x=x*16;
    y=x;
    asm AND x, 0F00H
    asm AND y, 0FFH
    x=x/256;
    outp(0x2FF,x);
    outp(0x2FE,y);
}

unsigned char fuz_member[ ] = {
0X01, 0XFF, 0X15, 0X04, 0X04, 0X51, 0X02, 0X51, 0X02, 0XFE, 0XF0, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0x
void main()
{
    fuz_in[0]=adc(0);
    fuz_in[1]=adc(1);
    fuz_in[2]=adc(2);
    calculate_fuzzy();
    dac(fuz_out[0]);
}

Appendix B. FLC Algorithm