Journal of Physics: Conference Series **1456** (2020) 012010 doi:10.1088/1742-6596/1456/1/012010

Converter matlab fuzzy inference to arduino Csystem

M Khairudin[1], H A Wijaya[1] and Muslikhin[2]

[1] Electrical Engineering Education Dept., Universitas Negeri Yogyakarta, Indonesia
[2] Southern Taiwan University of Science and Technology, Tainan City 710, Taiwan R.O.C.

E-mail: moh_khairudin@uny.ac.id

Abstract. This study aims to utilize the FIS matlab converter (fuzzy inference system) into the C arduino programming language with the aim of simplifying the fuzzy logic model that can be applied immediately with control hardware such as Arduino. In this study, an example of fuzzy logic is used with two sensor inputs, namely temperature and smoke sensors as inputs and two outputs in the form of buzzers and pump motors. The method used is sugeno in matlab. Then the formation of input variables modeled mapping scale is formed 0 - 100 and scale output 0 - 2. Converter. Fis fuzzy inference system can be applied to the design of fuzzy logic systems with the results of the .ino format conversion program template that can be set output results as arduino triger directly . This setting is enough to do with the reading of the g_fisOutput [] variable with IF-ELSE.

1. Introduction

The development of intelligent systems is increasingly rapid nowadays, one of which is soft computing. A system is said to be intelligent if it has a field of ability like humans at certain limits, such as being able to learn and adapt to environmental changes. The main elements in soft computing include fuzzy systems, artificial neural networks, predictive predictions and evolutionary computing [1].

Fuzzy systems have several stages, namely fuzzy input, applying fuzy operators, applying implications, composing output and defuzification [2]. The right way to map input space into the output space can be done through artificial intelligent [3].

Among the artificial intelligent methods that are often used are fuzzy logic [4]. With a fuzzy logic system-based system control will save more energy [5]. Conventional systems adhere to the concept of single output control from several inputs that are not directly related to each other [6]. Fuzzy is able to add new membership functions with rules that are interconnected if there are additional inputs or inputs [7].

There are several reasons for using fuzzy logic, namely (1) having flexible logic; (2) has a tolerance range; (3) has a simple mathematical basic concept; (4) able to handle nonlinear functions; (5) can apply the experiences of experts directly without having to go through the training process; (6) able to synergize with conventional control techniques; (7) based on natural language [8].

Control of a fuzzy system can be synergized with control devices, one of which is Arduino [9]. At present Arduino is able to communicate with MATLAB on the basis of using C programming [10]. The process of adaptation from matlab in the fuzzy inference section towards C programming is now needed a simpler step so that the method applied can be immediately applied with control hardware such as arduino.

Fuzzy logic system programming using the matlab simulink tool is very easy for beginners of fuzzy system designers. However, the simulink programming system in matlab is based on a block diagram so it needs further steps that are not simple when it is applied to the hardware. Though in fact computational programming in the control system aims to regulate the hardware system.

This study aims to describe simple steps in converting sourceode from matlab simulink to c-shaped source code. So that the C programming format will be compatible with hadware systems. The simplification process can be done by converting the matlab design file with the format .fis into the fuzzy logic template file in c arduino programming in the .ino format.

2. Method

In this study, a case example of fuzzy logic is used with two sensor inputs, namely temperature and smoke sensors as input and two outputs in the form of buzzers and pump motors. The method used is sugeno in matlab. Then the formation of input variable data is formed which is modeled on a scale of 0-100 mapping.

a. Temperature consists of three fuzzy sets, namely: cool, medium, and hot
b. Smoke consists of three fuzzy sets, namely: thin, medium, thick

While the output variable in the form of buzzer and pump motor is modeled on a scale of 0-2 through the configuration parameters are constant.

a. The buzzer consists of three fuzzy sets, namely off, beep flip flops, and long beeps
b. The pump motor consists of three fuzzy sets, namely, off, on two seconds, and on

3. Results and discussion

The membership function (MF) is a curve that shows the mapping of input data points into membership values that have an interval between 0 and 1.

Based on the modeling, it is then presented in the function of the degree of membership according to the scale of the mapping used as follows:

a. Temperature variable

In Figure 1 shows the temperature input distributed into the three rules of membership function, which are cool, temperate, and hot. While the waveform used is a triangle.

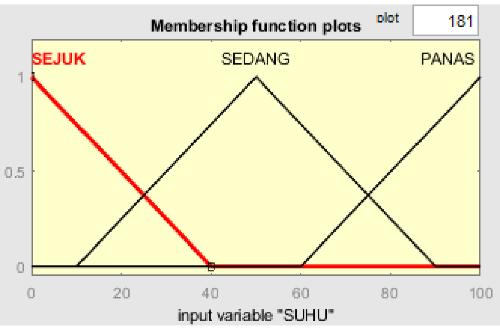

Figure 1. Temperature Variables

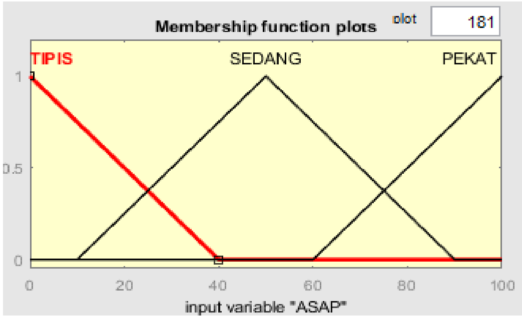

Figure 2. Smoke Variables

b. Smoke variable

While in Figure 2, the smoke input is also distributed into three membership functions, namely thin, medium, and concentrated. While the waveform used is a triangle

c. Variable Buzzer

Meanwhile for the output part of this system consists of a Buzzer and a pump motor. As for the output in the form of a Buzzer distributed into three rules of membership functions, namely LONGBEEP, FLIPFLOP, and OFF. At the Buzzer output timer is used to set the length of the buzzer sound time. Figure 3 shows the variation of the membership function in Buzzer's output.

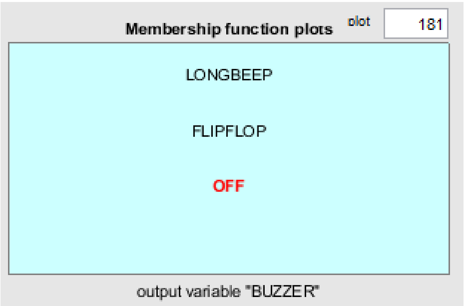

Figure 3. Variable Buzzer

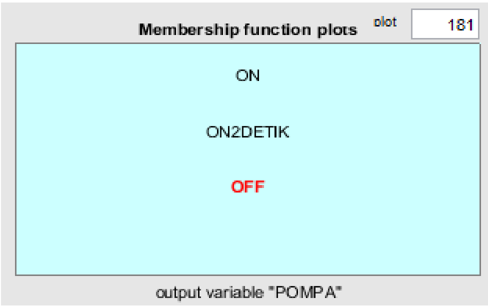

Figure 4. Variable Pump Motor

d. Variable Pump Motor

The pump motor output is also distributed into three membership functions, namely on, on for 2 seconds, and off. While the timer settings are used to give a variation on the length of time the pump motor turns on. Figure 4 shows the variation of the membership function at the pump motor output.

The next step is determining fuzzyfication in TEMPERATURE and ASAP input. For the TEMPERATURE input type, the input variable (x) is determined, which is the analog to digital converter (ADC) value of 100. The determination of fuzzyfication with the model on MF Sugeno Night to obtain the error value () is to use the formula:

$$\mu\ MF\ Sejuk = \begin{cases} 1; x \le a \\ (b-x)/(b-a); & a \le x \le b \\ 0; & x \ge b \end{cases}$$

With x: the value of the TEMPERATURE sensor input variable from ADC, b: upper limit and a: lower limit. So that for night MF if given x = 100, a = 0 and b = 1000, then it is obtained $\mu\ sejuk = 0{,}9$ While the determination of fuzzyfication with the Sugeno model at MF noon to get the error value (μ) is to use the formula:

$$\mu\ MF\ \ panas = \begin{cases} 0; & x \le a \\ (x-a)/(b-a); & a \le x \le b \\ 1; & x \ge b \end{cases}$$

With x: the value of the TEMPERATURE sensor input variable from ADC, b: upper limit and a: lower limit.So that for hot MF if given x = 100, a = 0 and b = 1000, then it is obtained $\mu\ \ panas = 0{,}1$

For the ASAP input type, the input variable (x) is determined, namely echo input and trigger output with an analog to digital converter (ADC) value of 50. The determination of fuzzyfication with the model on MF sugeno is low to obtain the delta error (μ) value using formula:

$$\mu\ MF\ tipis = \begin{cases} 1; x \le a \\ (b-x)/(b-a); & a \le x \le b \\ 0; & x \ge b \end{cases}$$

With x: the value of the ASAP sensor input variable from the ADC, b: the upper limit and a: the lower limit. So that for MF noon if given x = 50, a = 0 and b = 40, then it is obtained $\mu\ tipis = 0$

While the determination of fuzzyfication with the sugeno model on MF is being used to obtain the value of delta error (μ) using the formula:

$$\mu\ MF\ \ sedang = \begin{cases} 0; & x \le a\ atau\ x \ge c \\ (x-a)/(b-a); & a \le x \le b \\ (c-x)/(c-b); & b \le x \le c \end{cases}$$

With x: the value of the ASAP sensor input variable from ADC, c: upper limit, b: middle limit and a: lower limit. So that MF is dimmed if given x = 50, a = 0, b = 40 and c = 80, then it is obtained $\mu\ sedang = 0{,}75$

While the determination of fuzzyfication with the Sugeno model on concentrated MF to get the error value (μ) is to use the formula:

$$\mu\ MF\ \ pekat = \begin{cases} 0; & x \le a \\ (x-a)/(b-a); & a \le x \le b \\ 1; & x \ge b \end{cases}$$

With x: the value of the ASAP sensor input variable from the ADC, b: the upper limit and a: the lower limit.So that for bright MF if given x = 50, a = 0, and b = 80, then it is obtained $\mu\ pekat = 0{,}25$

Determination of RULE by using logical AND operations $\alpha - predicate$ as the result of operations using logical AND operations is obtained by taking the smallest membership value between the elements in the relevant sets.

After completing fuzzification through setting a membership function, the next step is to determine the rules. In determining the rule as shown in Figure 5, a rule is then made for all existing variables, namely two inputs and 2 outputs.

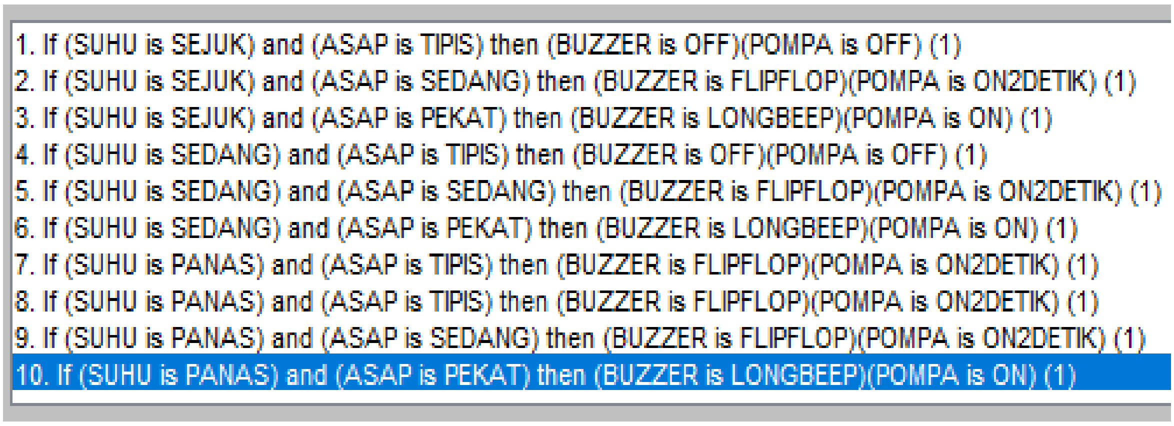

Figure 5. Design rules

At this stage the fuzzy design created can be saved as a .fis file that will be converted to c arduinowith the .ino format.The .fis file is then converted through the pagemakeproto.com/projects/fuzzy/matlab_arduino_FIST/index.php and you will get the .ino conversionfile package in the form of a result template file from the rules designed in the previous matlab.

The implementation of the physical conversion to c arduino results in a fuzzy logic program template with many simplifications, because the data from the rules output results are placed in the program code line as shown in Figure 6.

```
// Read Input: SUHU
g_fisInput[0] = analogRead(0);
// Read Input: ASAP
g_fisInput[1] = analogRead(1);
g_fisOutput[0] = 0;
g_fisOutput[1] = 0;
fis_evaluate();
// Set output vlaue: BUZZER
analogWrite(2 , g_fisOutput[0]);
// Set output vlaue: POMPA
analogWrite(3 , g_fisOutput[1]);
```

Figure 6. Location of output of conversion results

```
if ((g_fisOutput[0] >= 0) && (g_fisOutput[0] < 1)) {
   digitalWrite(BUZZER, LOW);
 }
 if ((g_fisOutput[0] >= 1) && (g_fisOutput[0] < 2)) {
   digitalWrite(BUZZER, HIGH);delay(1000);
   digitalWrite(BUZZER, LOW);delay(1000);}

 if ((g_fisOutput[0] >= 2) && (g_fisOutput[0] < 3)){
digitalWrite(BUZZER, HIGH);}}
```

Figure 7. Listing source code for conversion to .fis c

The output data of the conversion result is located in the g_fisInput [0] variable for the temperature variable while the smoke input variable is located in g_fisInput [1].

Output data is at g_fisOutput [0] for triger on buzzer and g_fisOutput [1] for pump motor output variable. These two g_fisOutput variables will be valued on a scale of 0 - 2 which can be easily entered in the IF-ELSE process to trigger the output hardware to work in accordance with the input.

The development of the g_fisOutput variable in an effort to simplify the conversion of the .fis matlab into the c arduino language can be seen in the program line as shown in Figure 7 below:

Based on the program row data in Figure 7 above, we will get a table of output calculation results as shown in Table 1.

Tabel 1. Output

g_fis Output	
Score	**Condition**
0	Buzzer Off
1	Buzzer Flip Flop
2	Buzzer On

4. Conclusion

Converter .Fis fuzzy inference system can be applied to the design of fuzzy logic systems with the results of the .ino format conversion template that can be set directly as an Arduino triger. This setting is enough to do with the reading of the g_fisOutput [] variable with IF-ELSE.

5. References

[1] Afan G.S. Pemodelan Sistem Fuzzy dengan Menggunakan Matlab. ComTech. 2010; Vol.1 No.2: Desember 2010

[2] Setyoningsih W. Penerapan Logika Fuzzy dalam Penjadwalan Waktu Kuliah. UPGRIS. 2015; Vol.1: Juni 2015

[3] Febri SR, Fatchul A. "Developing Artificial Neural Network Based on Visual Studio for Dance Assessment," Jurnal Pendidikan Teknologi dan Kejuruan, Vol. 23, No.4, 2017
[4] Kurian CP, Kuriachan S, Bhat J, Aithal RS. An Adaptive Neuro-Fuzzy for the Prediction and Control of Light in Integrated Lighting Schemes. Lighting Research & Technology. 2015; 37(4): 343-352
[5] Ghadi YY, Rasul MG, Khan MMK. Energy Savings by Fuzzy Base Control of Occupancy Concentration in Institutional Buildings. Energy Procedia. 2017.105: 2850–2858.
[6] M Khairudin, AD Hastutiningsih, THT Maryadi, and HS Pramono, (2018), Water level control based fuzzy logic controller: simulation and experimental works, IOP Conf. Series: Materials Science and Engineering 535 (2019) 012021.
[7] M. Khairudin, S. Riyanto, Z. Mohammed, (2018), Development of Fuzzy Logic Control for Indoor Lighting Using LEDs Group, TELKOMNIKA, Vol.16, No.3, June 2018, pp. 1165~1173
[8] Maryaningsih. Implementasi Logika Fuzzy dalam Perancnagan Sistem Pengambilan Keputusan Usulan Pemasangan Listrik Berdasarkan Distribusi Beban Listrik. Media Infotama. 2013; Vol.9 No.2: September 2013
[9] Ismail ME, Masran S.H, Rahim M.B, Faizal A.N and Marian M.F. "Development of Electrical Discharge Machine Die Sinking Application Using Android Platform," Jurnal Pendidikan Teknologi dan Kejuruan, Vol. 23, No.4, 2017
[10] The MathWorks.Embedded MATLAB User´s Guide, March 2008.