ILKOM Jurnal Ilmiah Vol. 14, No. 1, April 2022, pp. 39-51
Accredited 2nd by RISTEKBRIN No. 200/M/KPT/2020; E-ISSN 2548-7779 | P-ISSN 2087-1716

The implementation of fuzzy logic in fish dryer design

Nur Yanti[a,1,*]; **Taufik Nur** [b,2]; **Randis** [c,3]
[a] *Politeknik Negeri Balikpapan, Jln. Soekarno Hatta KM 8, Balikpapan, Kalimantan Timur 76129, Indonesia*
[b] *Universitas Muslim Indonesia, Jln. Urip Sumoharjo KM.5 Makassar, 90231, Indonesia*
[c] *Politeknik Negeri Balikpapan, Jln. Soekarno Hatta KM 8, Balikpapan, Kalimantan Timur 76129, Indonesia*
[1] *nur.yanti@poltekba.ac.id *;* [2] *taufik.nur@umi.ac.id;* [3] *randis@poltekba.ac.id*
* *Corresponding author*

Article history: Received January 09, 2022; Revised April 06, 2022; Accepted April 06, 2022; Available online April 30, 2021

Abstract

The fish drying process aims to preserve the fish to reduce losses due to the decay process. In hot conditions or through sunlight exposure, the drying process should not be a problem, but if it rains, the fish drying process will take a longer time, and create a smell that disturbs the surrounding environment for a relatively long time. The fish dryer is designed to work automatically, aiming to speed up drying time using fuzzy logic, thereby minimizing spoilage and air pollution due to smell from the fish drying process. The design of the tool uses an experimental method through literature study as a source of analysis, planning, and manufacturing fish dryer using an Arduino Mega 2560, temperature sensor DHT 22, load cell sensor, humidity sensor, fan, heating element, and LCD as well as software with Fuzzy Mamdani method. The results show that the weight of fish that had undergone a drying process using an automatic dryer, which was 500 grams, indicated a drying process of 50% of the initial weight of 1000 grams, with a drying time of 4.48 hours. The previous drying time by manual drying took 45 hours. It shows a control system using fuzzy logic on fish drying equipment, accelerating the drying time about 10 hours faster than sun drying time. It can be concluded that it can increase the amount of dry fish production and reduce odors in the environment around the drying because the fish are in the dryer with a closed condition.

Keywords: fuzzy logic; microcontroller; fish dryer.

Introduction

The dominant occupation of coastal residents in Indonesia is fishermen. Fish is caught by the fishermen who go to the sea. Usually, caught fish is sold at Fish Auction Places (TPI), traditional markets, to buyers who have a restaurant business, or to buyers who resell the fish. Quite often the fish caught are not all sold out; sometimes, there are leftover fish from the sale. To make the remaining fish valuable and can still be an income for the family, the fish is processed so as not to rot through the drying process. The drying process will not be hampered if the weather is hot. Drying fish in coastal residential areas is usually done in the traditional way, i.e., using sunlight. However, this condition will be different during rainy weather. The drying process of fish will take longer and has an odor that will affect the surrounding environment. So, it is necessary to design a fish dryer, aiming to help speed up the drying time of fish, and reduce odors due to the open fish drying process. With this fish dryer, the drying process is done in a closed place, so that the smell due to the drying process does not spread to pollute the surrounding environment. Using a fish dryer designed with the fuzzy intelligent method can increase fishers' income because the drying time is shorter than the conventional method.

This fish dryer is designed using a DHT22 temperature sensor, load cell as a weight sensor, Arduino Mega 2560 microcontroller as the central controller, heating element, fan, and fuzzy logic method embedded in Arduino programming. The DHT22 sensor measures the temperature and humidity of the air in the fish drying room. The temperature and humidity of the air in the drying room significantly affect the drying process. The DHT 22 sensor is more sensitive to Relative Humidity (RH) [1]. Relative humidity shows the percentage ratio between the water vapor present in the air when measuring the maximum amount of water vapor the air can accommodate [2]. Drying is done to remove or remove most of the water from the fish's body to stop the growth of bacteria, reduce the activity of autolysis that causes rancid odor and prevent the development of fungi. Fish that have undergone a drying process will experience a decrease in weight due to a reduction in water content during the drying process [3]. Various designs of fish dryers by applying multiple methods have been developed in various forms. A fish dryer works automatically by removing and entering the fish dryer according to the surrounding weather or land robot [4]. A fish dryer with an automatic drying method utilizes a hairdryer in a closed room and monitors temperature through the website in real-time [5] [9]. The development of android-based technology also makes it easier for us to do monitoring work, such as monitoring the temperature of the fish dryer using the Blynk App, equipped with a timer and able to dry fish for 12 hours under wet fish conditions [6].

The role of the controller in the design of this fish drying device is the most critical component in the temperature control system in the drying room, to control the heater so that the output in the form of fish weight drying 50% of the initial weight can be achieved. Using the Arduino Mega 2560 microcontroller as a controller with Fuzzy Mamdani logic applied to the Arduino program. This dryer is designed using intelligent logic so that the controller is able to respond to various conditions from the output of the DHT22 temperature sensor in the form of temperature and humidity so that this heater control system can produce a constant hot temperature to dry fish. Temperature and humidity are one of the physical quantities used as parameters of a control system, for monitoring systems only or other control processes [7]. Arduino Mega 2560 designed a mini-scale automatic fish dryer device that controls the temperature sensor, mass sensor (loadcell), keypad, LCD, relay, and cooling fan [8]. An Embedded system with a practical approach based on Arduino and Android using the MIT App inventor software in a web-based fish drying system reduces the water content of the blue scads with a drying time of 8 hours [9].

The use of artificial intelligence in various lines as a control, prediction or identification system gives more value to the arrangement of actuators, plants, or objects that are controlled or identified. So that it provides specific reasons for the applicable use. The use of fuzzy logic in this fish dryer helps speed up the drying time so that the heater will work automatically with adjustments as a response to conditions in the drying room. Controlling the air conditioning temperature automatically in the classroom [10] can improve the exhaust fan movement, resulting in the car's air conditioning ECU system [11]. In processing multisensor reading data to detect potential fires, fuzzy logic produces a high level of accuracy [12]. To increase the yield of hydroponic cultivation, a Fuzzy Logic Controller (FLC) is used to maintain the stability of the pH of the solution in hydroponic plants [13]. Implementing fuzzy logic in a business for budget planning predictions by using Fuzzy Tsukamoto can optimize budget planning by 10.12% [14]. Recognition or identification of an object, such as an object's shape, color, or through imaging using the fuzzy method produces a high level of accuracy, such as reading the red, blue, green, and yellow colors for the directions, as well as reading the distance of obstacles on the pointer system direction for the visually impaired [15].

Method

The method used is the experimental method. The system was designed through the hardware, software, testing, and system implementation stages.

A. Hardware Design

The hardware design is shown in the block diagram in **Figure 1**. The fish dryer was designed using a DHT22 temperature sensor, load cell sensor, mosfet driver, fan, heater, and indicators on the Liquid Crystal Display (LCD).

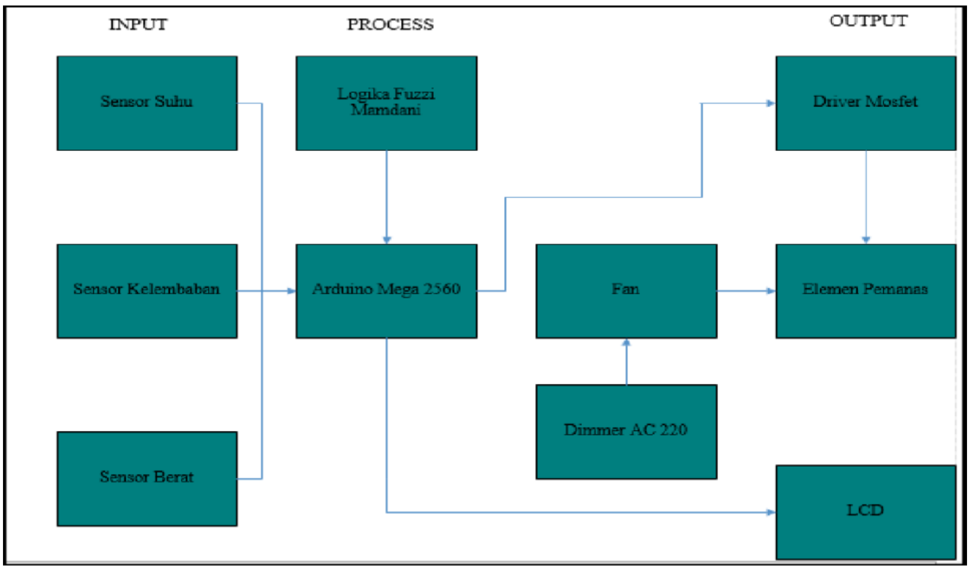


Figure 1. System block diagram

The working principle of the fish dryer includes the following steps: The DHT22 sensor detected the temperature in oC and the humidity of the fish, in Reality Humidity (RH), which was %. The load cell sensor detected fish weight/pressure, in kilograms (kg). The DHT22 sensor and load cell were connected to the Arduino PWM digital I/O pins. The Arduino Mega 2560 microcontroller controlled the drying process, which had embedded fuzzy logic. The fuzzy logic-based microcontroller was connected to the driver to control the heater as the system output. The fan was activated via a switch; the fan served to help push the hot steam generated from the heater to all parts of the drying chamber so that the fish could dry evenly. LCD as an indicator displays the drying output in the form of temperature, humidity, and weight. The wiring system between input-process-output, shown in **Figure 2**, with a 12 V power supply to activate the heater.

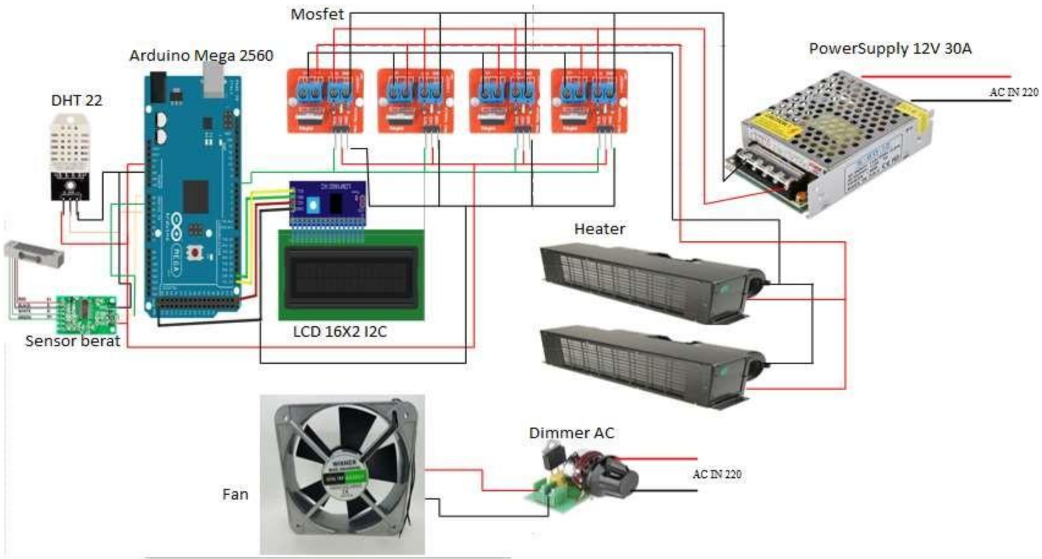


Figure 2. Wiring of the overall tool design

B. Software Design

The software design is to design a fish dryer by instilling the concept of intelligent logic or fuzzy logic. The method used was the Fuzzy Mamdani type. The design of the fuzzy approach was carried out with the fuzzification stage, building a rule base on MATLAB, inference, and defuzzification. The fuzzy logic design is shown in **Figure 3**.

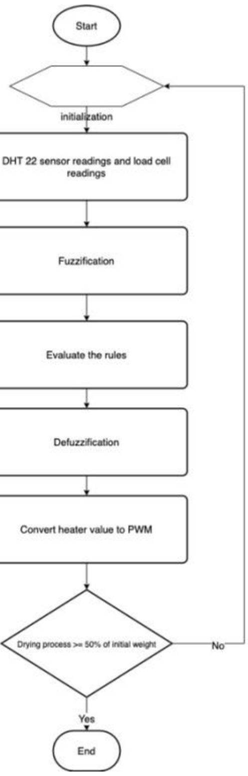


Figure 3. Fish dryer design flow diagram with fuzzy logic

The workflow of the system is to initialize the components that are interconnected. The DHT22 sensor and load cell performed data readings. The sensor output in analog data was converted into digital data on the Mega 2560 microcontroller. The system output is a heater in the form of PWM. Fuzzy logic control design includes fuzzification, namely changing the existing firm values into membership functions or degrees. Membership functions designed in MATLAB in input temperature, humidity, weight, and heating output in the form of PWM values, are shown in Figures 4,5,6 and 7. Formation of a rule base for decision-making. The next stage was the fuzzy value was defuzzified to change the fuzzy value from the degree of membership to a firm value as the output value. Then if the weight of the fish <= 50% of the initial weight, then the drying process had been completed, and vice versa if the weight of the dried fish did not reach a minimum of 50%, then the drying process was continued.

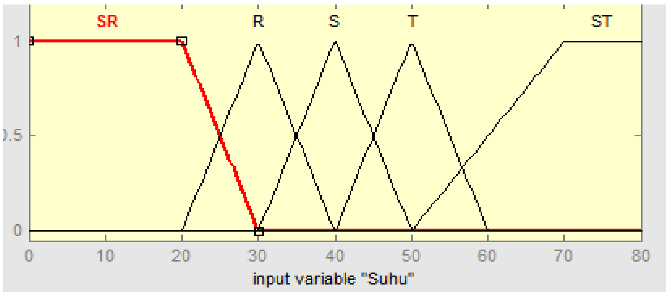


Figure 4. Temperature membership function

Figure 4 shows the membership function of the temperature variable, consisting of 5 linguistic sets in the membership function, namely SR=Very Low, R=Low, S=Medium, T=High and ST=Very High. Domain membership function temperature is between 0-80°C.

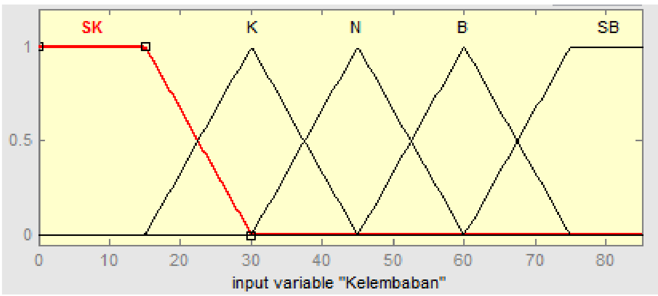


Figure 5. Humidity membership function

Figure 5 shows the membership function of the humidity variable, consisting of 5 linguistic sets on the membership function, namely SK=Very Dry, K=Dry, N=Normal, B=Wet and SB=Very Wet. The humidity membership function domain is between 0-80 RH.

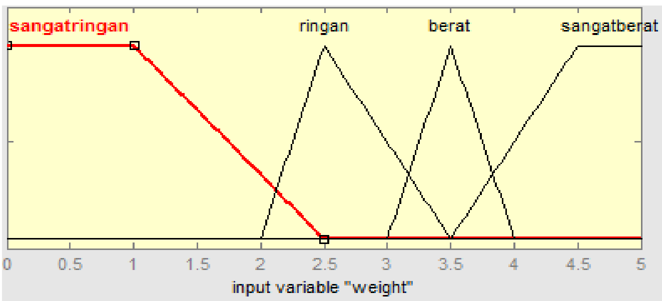


Figure 6. Weight membership function

Figure 6 shows the membership function of the heavy variable, consisting of 4 linguistic sets on the membership function, namely SR = Very Light, R = Mild, B = Heavy, SB = Very Heavy. The domain membership function is the weight between 0-5 kg.

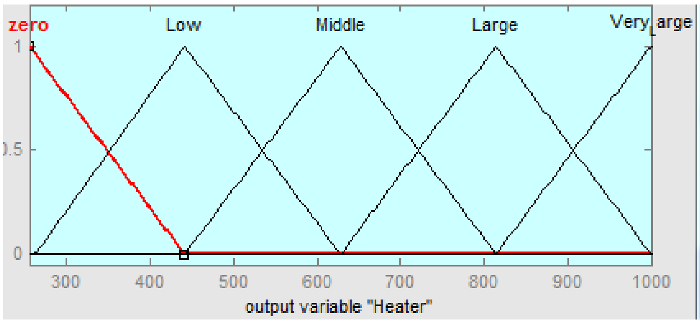


Figure 7. Weight membership function

Figure 7 shows the membership function of the heater variable, consisting of 5 linguistic sets in the membership function, namely zero, low, middle, large, very large. The domain membership function of the heater as output is between 0-5 kg. At the stage of building the rule base, from the rule base it can know the condition of the most dominant heating value. In the rule base there are linguistic rules to determine the control action on the input value of the fuzzification, namely (IF-THEN). There are 100 rules that have been designed, including:

[1] IF Temperature is SR and Humidity is K and Weight is Very Light then Heater is Middle
[2] IF Temperature is R and Humidity is SK and Weight is Very Light then Heater is Low
[3] IF Temperature is S and Humidity is SK and Weight is Very Light then Heater is Zero
[4] IF Temperature is T and Humidity is SK and Weight is Very Light then Heater is Zero
[5] IF Temperature is ST and Humidity is SK and Weight is Very Light then Heater is Zero
[6] IF Temperature is SR and Humidity is K and Weight is Very Light then Heater is Middle
[7] IF Temperature is SR and Humidity is N and Weight is Very Light then Heater is Middle
[8] IF Temperature is SR and Humidity is B and Weight is Very Light then Heater is Middle
[9] IF Temperature is SR and Humidity is SB and Weight is Very Light then Heater is Middle
[10] IF Temperature is SR and Humidity is SK and Weight is Light then Heater is Middle
[11] IF Temperature is SR and Humidity is SK and Weight is Heavy then Heater is Large
[12] IF Temperature is SR and Humidity is SK and Weight is Very Heavy then Heater is Large
[13] IF Temperature is R and Humidity is K and Weight is Very Light then Heater is Low
[14] IF Temperature is S and Humidity is K and Weight is Very Light then Heater is Zero
[15] IF Temperature is T and Humidity is K and Weight is Very Light then Heater is Zero
[16] IF Temperature is ST and Humidity is K and Weight is Very Light then Heater is Zero
[17] IF Temperature is R and Humidity is N and Weight is Very Light then Heater is Low
[18] IF Temperature is R and Humidity is B and Weight is Very Light then Heater is Middle
[19] IF Temperature is R and Humidity is SB and Weight is Very Light then Heater is Middle
[20] IF Temperature is R and Humidity is K and Weight is Light then Heater is Middle
[21] IF Temperature is R and Humidity is K and Weight is Heavy then Heater is Large
[22] IF Temperature is R and Humidity is K and Weight is Very Heavy then Heater is very large
[23] IF Temperature is S and Humidity is N and Weight is Very Light then Heater is Low
[24] IF Temperature is T and Humidity is N and Weight is Very Light then Heater is Low
[25] IF Temperature is ST and Humidity is N and Weight is Very Light then Heater is Zero
[26] IF Temperature is S and Humidity is B and Weight is Very Light then Heater is Low
[27] IF Temperature is S and Humidity is SB and Weight is Very Light then Heater is Middle
[28] IF Temperature is S and Humidity is N and Weight is Light then Heater is Middle
[29] IF Temperature is S and Humidity is N and Weight is Heavy then Heater is Large
[30] IF Temperature is S and Humidity is N and Weight is Very Large then Heater is Very Large
etc.

Defuzzification is a step in the fuzzy logic system to convert each result from the inference engine which is expressed in the form of a fuzzy set to a real number. The process used in this fish drying system is the Center of Area (COA). In this method, the crisp solution was obtained by taking the mean domain value which had been converted by the Fuzzy Inference System (FIS). **Figure 8** shows the results of the viewer rules that have been converted on the inference engine.

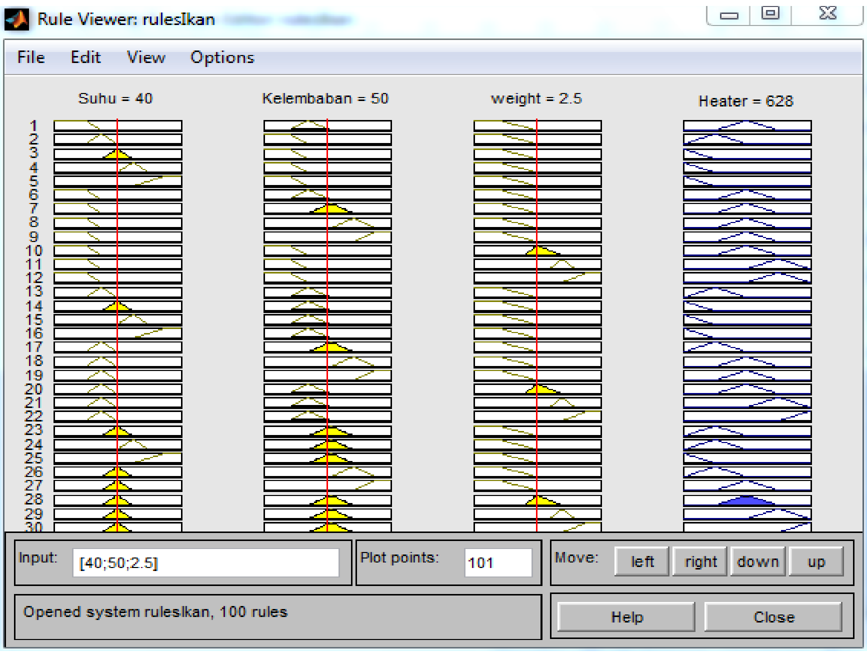


Figure 8. Fuzzy logic MATLAB results

Results and Discussion

Tool testing was carried out to ensure that the tools that had been designed could function and were used according to the design objectives. There are five tests in the system: testing the DHT22 sensor, testing the load cell sensor, testing fuzzy logic control, testing Arduino Mega, and testing the system as a whole.

A. DHT22 Sensor Testing

We were testing the accuracy of the DHT22 sensor through comparison with a digital hygrometer. Table 1 shows the temperature measurements, and table 2 the humidity measurements. Data retrieval to compare DHT22 and hygrometer was by heating the heater and detecting temperature and humidity inside the drying chamber (oven), then measuring the DHT22 and hygrometer. From the results of sensor data retrieval, the error value can be found with the formula (1).

Error= (DHT22 Value-Hygrometer Value)/(DHT22 Value) x 100% (1)

Table 1. Comparison of DHT22 temperature data (oC) and hygrometer (oC)

Suhu			
DHT 22	***Hygrometer***	***Selisih***	***Error***
45.8	45.7	0.1	0.22%
46	46.3	0.3	0.65%
46.8	46.8	0	0%
48	47.8	0.2	0.42%
48.2	48	0.2	0.41%
48.6	48.2	0.4	0.82%
48.9	48.4	0.5	1.02%
51	50.8	0.2	0.39%
51.6	51.3	0.3	0.58%
52	52	0	0%
Rata-rata error		0.22	0.45%

Table 2. Comparison of humidity data for DHT22 (RH) and hygrometer (RH)

Kelembaban			
DHT 22	***Hygrometer***	***Selisih***	***Error***
45	43	2	4.44%

Kelembaban			
DHT 22	***Hygrometer***	***Selisih***	***Error***
43.6	42	1.6	3.67%
47	44	3	6.38%
43	42	1	2.32%
42.2	41	1.1	2.6%
48	44	4	8.33%
46	43	3	6.52%
44.3	43	1.3	2.93%
42.6	41	1.6	3.75%
41	39	2	4.9%
Rata-rata error		2.06	4.58%

From the results of testing the sensor on the sensor datasheet with the measured results, it is proven that the accuracy value on the datasheet is ±0.5°C. Further, it is proven that the average value of the difference is 0.22oC, and on the measurement of the humidity value (RH%) the value measured with the reference datasheet is proven to be between the error range is 3%-5%, and the average error value is 4.58%. The humidity error that occurred is between 3%-5%, according to the accuracy in the DHT22 datasheet; this error was calculated from the difference between the data readings on the DHT22 sensor and data from the hygrometer. The results of the first reading of humidity data on DHT22 are 45 (RH), while the hygrometer displays the results of the first humidity reading of 43 (RH). So the humidity error value (RH) for the first data:

$$Error = \frac{45\ (nilai\ DHT22) - 43\ (nilai\ hygrometer)}{45\ (nilai\ DHT22)}\ x\ 100\% \ = \ 4.44\%\ (error\ antara\ 3\% - 5\%)$$

For the next data, the humidity error was calculated using the same formula and method. **Figure 9** shows the DHT22 sensor test.

Figure 9. Testing the DHT22 sensor

B. Load Cell Sensor Test

The load cell sensor is a sensor that detects weight. In the design of this sensor, it functions as a fish scale during the drying process and as a benchmark for whether or not the drying process has been achieved. **Table 3** shows a comparison of the accuracy of the weight values on the load cell sensor and the scales. Sensor data from the test results between the load cell and the scales are shown in **Figure 10**; there is an average error value of 5.612%, the error value is determined by the formula (2).

$$Error = \frac{Nilai\ load\ Cell - Nilai\ timbangan}{Nilai\ load\ cell}\ x\ 100\% \qquad (2)$$

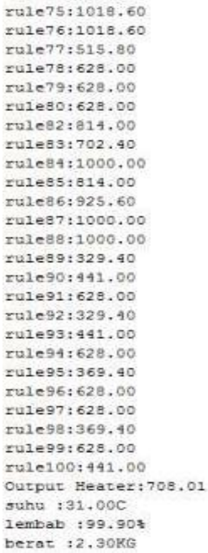
Table 3. Comparison of load cell sensor data and scales

Berat			
Load Cell (Kg)	***Timbangan (Kg)***	***Selisih (Kg)***	***Error***
0.52	0.5	0.02	3.84 %
0.7	0.8	0.1	14,28 %
1.1	1	0.1	9.09 %
2.1	2.3	0.2	9.52 %
2.65	2.52	0.13	4.9 %
3.02	3	0.02	0.6 %
4.22	4	0.22	5.21 %
4.54	4.5	0.04	0.88 %
3.7	3.58	0.12	3.2 %
2.83	2.7	0.13	4.6 %
Rata-rata error		0.108	5.612%

Figure 10. Testing the load cell sensor

C. Fuzzy Logic Testing in Arduino Programs

This fuzzy logic test was carried out by comparing the output value generated in MATLAB with the output value on the Arduino programming (hardware). In this test, fuzzy rules were formed according to what had been determined, the input value come from reading the sensor program that had been programmed. The predetermined input values for temperature, humidity, and weight were displayed on the serial monitor in the Arduino sketch. The input value entered in the MATLAB rule viewer was adjusted to the value in the Arduino programming. **Figure 11** shows the test results displayed on the serial monitor. **Figure 12** shows the fuzzy output in MATLAB.



```
rule75:1018.60
rule76:1018.60
rule77:515.80
rule78:628.00
rule79:628.00
rule80:628.00
rule82:814.00
rule83:702.40
rule84:1000.00
rule85:814.00
rule86:925.60
rule87:1000.00
rule88:1000.00
rule89:329.40
rule90:441.00
rule91:628.00
rule92:329.40
rule93:441.00
rule94:628.00
rule95:369.40
rule96:628.00
rule97:628.00
rule98:369.40
rule99:628.00
rule100:441.00
Output Heater:708.01
suhu :31.00C
lembab :99.90%
berat :2.30KG
```

Figure 11. Fuzzy output results on serial monitor

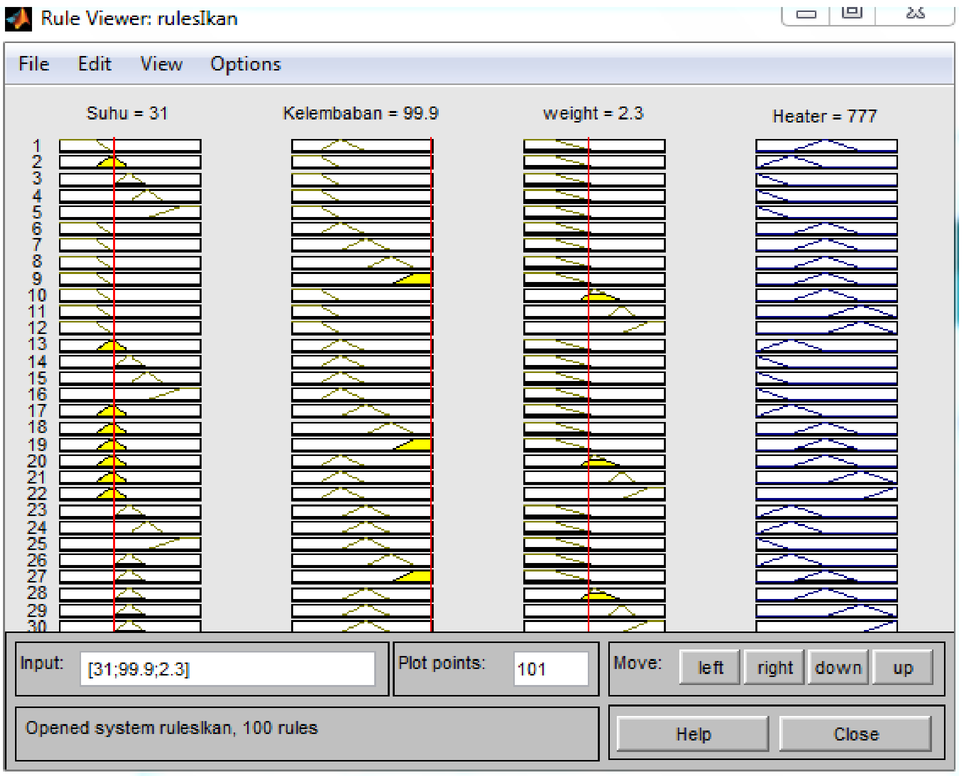


Figure 12. Fuzzy output results in MATLAB

D. Program System on Arduino IDE

Programming that has been designed on the Arduino IDE used modified C programming. The first thing that was done on the Arduino IDE programming system was a global variable that functions so that the program can be executed correctly as desired. In **Figure 13**, global initialization was carried out by entering several libraries and determining the pins to be used and floats being carried out to store real numbers.

```
#include <LiquidCrystal_I2C.h>
#include"DHT.h"
#define DHTPIN A2
#include "HX711.h" //memasukan library HX711
#define DOUT  A0 //mendefinisikan pin arduino yang terhubung dengan pin DT module HX711
#define CLK  A1 //mendefinisikan pin arduino yang terhubung dengan pin SCK module HX711
HX711 scale(DOUT, CLK);
float calibration_factor = 55050;
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x27, 16, 2);
const int Driver =7;
const byte degreeSymbol = B11011111;
float MF_suhu(float,float,float);
float member_suhu ;
float member_lembab;
float MF_kelembaban(float,float,float);
float MF_suhu(float a, float b, float c);
float MF_kelembaban(float a1, float b1, float c1);
float suhu;
float SR,R,S,T,ST;
float kelembaban;
float SK,K,N,B,SB;
float weight;
float sangatringan,ringan,berat,sangatberat;
float heater;
```

Figure 13. Program global variables

Then in the void setup section, which functioned only once during the program running aimed to declare the pin used, fill in variables and also activate libraries that have been included in global variables. A simple void setup can be seen in **Figure 14**.

```
void setup (){
Serial.begin(9600);

lcd.begin();
lcd.setCursor(4, 0);
lcd.print("HALO HERU");
delay(3000);
lcd.clear();
lcd.setCursor(5, 0);
lcd.print("FUZZY");
lcd.setCursor(1, 1);
lcd.print("PENGERING IKAN");
delay(3000);
lcd.clear();
dht.begin();
pinMode(Driver, OUTPUT);
Serial.begin(9600);
  scale.set_scale();
  scale.tare(); // auto zero / mengenolkan pembacaan berat
  long zero_factor = scale.read_average(); //membaca nilai output sensor saat tidak ada beban
  weight = scale.read_average();
}
```

Figure 14. Void setup

Then in the void loop section, which was a program that was read repeatedly, the void loop was filled with fuzzy programs, where the conditions that had been designed to determine the running of this tool are fuzzification, rule evaluation, and defuzzification. In simple terms, the void loop program can be seen in **Figure 15** that shows the fuzzification program.

```
//fuzzifikasi//
unsigned char suhuSR(){
  if (suhu <= 20){SR =1; R =0; S =0; T =0; ST =0;}
  else if (suhu >=20 && suhu <=30){SR=(30-suhu)/10;}
  else if (suhu > 30){SR =0;}
  return SR;
}
unsigned char suhuR(){
  if (suhu <= 20){R =0; SR =1; S =0; T =0; ST =0;}
  else if (suhu >20 && suhu <=30){R=(suhu-20)/10;}
  else if (suhu >30 && suhu <=40){R=(40-suhu)/10;}
  else if (suhu == 30) {R =1; SR =0; S =0; T =0; ST
  else if (suhu > 40){R =0; SR =0;}
  return R;
}
unsigned char suhuS (){
  if (suhu <=30 ){S =0; T =0; ST =0;}
  else if (suhu >=30 && suhu <40){S=(suhu-30)/10;}
  else if (suhu > 40 && suhu <=50){S=(50-suhu)/10;}
  else if (suhu == 40) {S =1; SR =0; R =0; T =0; ST
  else if (suhu > 50) {S =0; R =0; SR =0;}
  return S;
}
unsigned char suhuT (){
  if (suhu <=40 ){T =0; ST =0;}
```

Figure 15. Void loop fuzzification

It can be seen in **Table 4** comparison of Arduino and Matlab outputs. The input read by the hardware must be the same as the input value in Matlab, The Matlab input value was compared with the input value in the Arduino programming. The error value was obtained from the formula (3).

$$Error = \frac{Nilai\ Arduino - Nilai\ Matlab}{Nilai\ Arduino}\ x\ 100\% \qquad (3)$$

Table 4. Comparison of Arduino and Matlab Output Outputs

No	Input			Data pengujian output fuzzy (PWM)		
	Suhu	Kelembaban	Berat	Arduino	Matlab	Error
1	31	99.99	2.3	708	777	-0.09%
2	29.4	41.6	2.5	720	700	0.028%
3	44.8	25.9	3.58	687	614	0.106%
4	54.5	23.5	2.44	440	494	-0.12%
5	39	44	3.28	778	757	0.03%

No	Input			Data pengujian output fuzzy (PWM)		
	Suhu	Kelembaban	Berat	Arduino	Matlab	Error
6	49.6	63.3	2.32	620	595	0.04%
7	38.1	65.7	2.8	702	673	0.041%
8	49.6	65.7	1.23	477	444	0.07%
9	14.9	80.1	1.84	657	628	0.44%
10	48.50	78	1	528	522	0.015%
Rata-rata error						0.0812%

The results of fuzzy measurements on Arduino programming (hardware) and measurements using the fuzzy function calculation application in Matlab, as evidenced by the use of fuzzy systems between Matlab and hardware calculations are very good because the error value is below 1%, which is 0.0812%.

E. Overall Testing of Automatic Fish Dryer System

The overall test aims to determine the performance of the automatic fish dryer, measuring the output of drying results using an intelligent system. The testing phase includes testing the DHT22 sensor, load cell sensor, fuzzy method, LCD I2C and a series of other components. This test is also done by comparing the time between the fish drying process manually and automatically using the fuzzy method. The manual process by utilizing the sun's heat, when the weather is hot. While the automatic drying process uses a tool designed with a fuzzy logic system. The process of drying fish manually and automatically is carried out with a fish dry target of 50% of the initial weight. The overall tool testing can be seen in **Figure 16**. Manual and automatic fish drying test data are shown in **Tables 5 and 6**.

Table 5. Manual Fish Drying Test

Manual testing						
Type of Fish	Berat Awal (Kg)	Test result				
		Suhu	Kelembaban	Berat Akhir (Kg)	Waktu (Jam)	Kondisi
Mujair	0.66	27.50	87	0.33	24	Kering
Mujair	1	28	89	0.48	45	Kering
Mujair	2.2	28.20	90	1.47	50	Kering
Kakap	2.5	28	92	1.2	48	Kering
Kakap	2.8	27.50	92	1.3	53	Kering

Table 6. Automatic Fish Drying Test

Testing automatically						
Type of Fish	Berat Awal (Kg)	Test result				
		Suhu	Kelembaban	Berat Akhir (Kg)	Waktu (Jam)	Kondisi
Mujair	0.53	49.50	78	0.26	3.50	Kering
Mujair	0.51	49.70	75.10	0.24	4.40	Kering
Mujair	1.53	49.50	70	0.74	4.50	Kering
Mujair	0.7	48.20	74.20	0.32	4.30	Kering
Mujair	1	49.80	76	0.5	4.48	Kering

Figure 16. Overall system testing

Conclusion

The results of the design of an automatic fish dryer based on fuzzy logic could produce a fish weight of 50% of the initial weight. Fish dryer designed using a heating element with an Arduino Mega 2560 microcontroller-based fuzzy controller, DHT22 temperature and humidity sensor, load cell, and fan, worked well in producing an output in the form of automatic drying of fish weight, 50% of the initial weight before drying. This automatic fish dryer speeds up the drying process time, reduces the risk of fish spoiling, and reduces air pollution due to strong odors if the drying process is carried out naturally. The initial weight of 1 kg fish, if manually dried by heating in the sun, takes 45 hours, to get a final weight of 0.48 kg fish. With this automatic fish dryer based on fuzzy control, the fish drying process only takes about 4.48 hours to get a final fish weight of 0.5 kg. This shows that fuzzy logic is able to respond and control the system reliably. If this automatic fish dryer is designed for large size and capacity, it can be developed with a hybridization system of several intelligent methods, equipped with an android-based and real-time monitoring system.

References

[1] H. I. Islam et al., "Sistem Kendali Suhu Dan Pemantauan Kelembaban Udara Ruangan Berbasis Arduino Uno Dengan Menggunakan Sensor Dht22 Dan Passive Infrared (Pir)," no. October, p. SNF2016-CIP-119-SNF2016-CIP-124, 2016.

[2] M. N. kaliky Hj. A. Irmayani p , Asrul, "Desain Bangun Ayakan Alat Mesin Tanaman Perkebunan," J. Telekomun. Kendali dan List., vol. 2, no. 1, pp. 12–22, 2020.

[3] S. F. Lukmansyah, S. Sumaryo, and E. Susanto, "Pengembangan Sistem Pengeringan Ikan Asin Otomatis Dengan," e-Proceeding Eng., vol. 6, no. 2, pp. 2786–2793, 2019.

[4] J. Hendrawan and D. Kurnia, "Perancangan Dan Penerapan Sistem Pengering Ikan Otomatis Menggunakan Logika Fuzzy Pada Mikrokontroller Atmega32a," J. Ilm. Core It, vol. 6, no. x, pp. 140–146, 2018.

[5] R. Rais and N. Nurohim, "Jemuran Ikan Asin Otomatis Berbasis Internet of Things Untuk Daerah Pesisir Pantai Pantura," Smart Comp Jurnalnya Orang Pint. Komput., vol. 9, no. 1, pp. 22–25, 2020.

[6] S. W. Murti, B. M. Basuki, and Sugiono, "MODEL PENGERING IKAN ASIN BERBASIS IoT SEBAGAI ALAT ALTERNATIF DIMUSIM HUJAN DALAM SKALA HOME INDUSTRI," pp. 1–11.

[7] M. Alexander, R. Rinaldi, Satria, and I. Priyadi, "Perancangan Modul Pengering Ikan Putaran Rak Vertikal Berbasis Mikrokontroller," Pros. Semin. Nas. Tek. Elektro, vol. 24, pp. 91–95, 2017.

[8] R. Pramana, K. Ilham, S. Nugraha, M. Otong, and D. Aribowo, "Perancangan Perangkat Pengering Ikan Otomatis Skala Mini," J. Sustain. J. Has. Penelit. dan Ind. Terap., vol. 8, no. 2, pp. 65–74, 2019.

[9] T. 1, R. Arief2, W. Widodo3, and Farida4, "Rancang Bangun Pengering Ikan Menggunakan Mikrokontroler Berbasis Web," … Semin. Nas. Sains …, pp. 239–246, 2020.

[10] G. Dewantoro, B. E. Nugraha, and F. D. Setiaji, "A Fuzzy Logic-Based Automation toward Intelligent Air Conditioning Systems," Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control, vol. 4, pp. 335–344, 2020.

[11] F. Fahmizal, T. R. Orlando, B. B. Murti, M. Budiyanto, and A. Mayub, "Kendali Logika Fuzzy pada Sistem Electronic Control Unit (ECU) Air Conditioner Mobil," J. Teknol. Inf. dan Ilmu Komput., vol. 6, no. 1, p. 25, 2019.
[12] N. Yanti, F. Z. Rahman, and T. Nur, "Rancang Bangun Sistem Pendeteksi Kebakaran Dini Berbasis Logika Fuzzy Menggunakan Multisensor," J. Ind. Eng. Manag., vol. 4, no. 2, pp. 46–57, 2019.
[13] A. R. Al Tahtawi and R. Kurniawan, "PH control for deep flow technique hydroponic IoT systems based on fuzzy logic controller," J. Teknol. dan Sist. Komput., vol. 8, no. 4, pp. 323–329, 2020.
[14] Y. Yanitasari, D. Dedih, and U. Mustofa, "Perencanaan Anggaran Pinjaman Dengan Prediksi Regresi Linier Sederhana Dan Optimasi Menggunakan Metode Fuzzy Tsukamoto," Ilk. J. Ilm., vol. 11, no. 3, pp. 206–213, 2019.
[15] N. Yanti, T. Nur, and Q. Hidayati, "Blind people's direction support system using ultrasonic, color sensor with fuzzy logic," IOP Conf. Ser. Mater. Sci. Eng., vol. 885, no. 1, 2020.