International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 13, No.1 (Jan-2023)

http://dx.doi.org/10.12785/ijcds/13011

A Fuzzy Logic IoT- Based Temperature and Humidity Control System for Smart Buildings

Heba N. Y. Al-Talb[1], Sama N. M. Al-Faydi[1], Toka A. Fathi[1] and Maan A. S. Al-Adwany[1]

[1] *Computer and Information Eng. Dept., College of Electronic Eng., Ninevah University, Mosul, Iraq*

Received 18 Nov. 2021, Revised 25 Sep. 2022, Accepted 12 Jan. 2023, Published 31 Jan. 2023

Abstract: Cooling systems are becoming an important part of everyday life with the exponential increase in cooling devices. Consequently, the power consumption is also increasing simultaneously. This paper aims to design an intelligent system to analyze the automation of cooling systems for smart buildings by applying fuzzy logic rules for different temperature and relative humidity parameters to control the surrounding air and power consumption. In such systems, the temperature and the relative humidity are considered input parameters, and the Ac mode is an output parameter. The system obtains the input data from the Temperature and humidity (DHT11) sensor by the Arduino microcontroller, the obtained data are then analyzed by the fuzzy logic control system. This system automatically controls the cooling devices when the temperature varies between 0 ^{o}C to 50 ^{o}C, and the relative humidity value varies between 0% to 90%. The analyzed data then is sent to ESP8266-12E Node MCU Wi-Fi microcontroller to monitor and control the real-time parameters online using the IoT through ThingSpeak. This system can provide an early warning indication to the user to acknowledge that the real-time remote parameters are changed and at the same time it triggers the cooling system to adapt the temperature condition to a desirable one. The obtained results show the proposed system can be applied practically in smart buildings.

Keywords: Smart Building, IoT, DHT11, Arduino Microcontroller, Fuzzy Logic System, Temperature, Relative Humidity

1. INTRODUCTION

Smart buildings rely on providing data and applications to a huge amount of big data delivered by smart things in any place. Most of the development technologies in current wireless networks are the Internet of Things (IoT). In any place in the world, anyone can connect to the IoT [1]. With the expansion of the IoT and the recent development of embedded systems, smart building applications have become more general in recent years [2]. With the usage of wireless mobile technology and sensors applications, IoT will connect billions of smart objects to collect data and sense it to communicate with people around the world. Smart buildings among many other IoT applications play an important role in the realization of smart cities. It can be used to remotely monitor and control electrical applications installed within the building by utilizing a smart and intelligent physical structure [3]. Temperature and relative humidity data are the two most important elements, DHT11 is used to determine these values for the surrounding environments in a good stability level at a two-second delay [4], [5], [6]. DHT11 sensor contains a moisture-holding component that is located between two terminals inside this sensor; a small variation in humidity makes a change in the component's conductivity, which causes an alteration in impedance between these terminals. Also, this sensor includes a thermistor, which acts as a temperature-dependent variable resistor, that measures the temperature values. In addition, the DHT11 sensor has been equipped with the ADC (Analog-to-Digital Converter) so that the input data has been converted into digital values. In this case, there is no need for an external ADC for processing input data on the microcontroller [7]. DHT11 is favored over DHT22, with its extremely low cost, and has a much higher sampling rate [8], [9]. Relative humidity and temperature are automatically monitored using sensors and microcontrollers through IoT-based systems on smart buildings. A fuzzy logic control system is one of the most suitable artificial intelligence methods to control the output state according to the input parameters [10]. The fuzzy logic becomes very popular because of the simple way of implementing the human control approach dependent on the proper mathematical model [11]. It offers several advantages that can control non-linear complex systems, and mathematically difficult represented systems [10]. Due to the availability of a dedicated fuzzy logic toolbox, executing a fuzzy logic controller in the MATLAB programming environment is a popular approach [11], [12]. Today Arduino is good hardware and easy to use because it has a lightweight, compact design, and supports an interface for sensors circuit. These capabilities make the system effective, easy to use, and accurate [13]. To offer wireless connectivity with Arduino, it is important to include the

E-mail address: heba.yahya@uoninevah.edu.iq, sama.mohamed@uoninevah.edu.iq, toka.fathi@uoninevah.edu.iq, maan.aladwany@uoninevah.edu.iq

ESP8266-12E Node MCU Wi-Fi chip which is an open-source programmable, low-cost system-on-chip microcontroller for building IoT platforms [14]. This module uses the TCP/IP protocol and can connect to other devices through a Wi-Fi connection. It can also be controlled using multiple integrated development environments (IDEs), such as the Arduino IDE [15]. In the literature, research by [16] presented the use of the IoT to design an appropriate house automation system model. By using this technology, they constructed low-cost smart houses for persons with visual or hearing disabilities, as well as pet owners. In [17], the researchers proposed the smart home IoT-based automated system that would allow users to remotely monitor and control home applications using an android-based appliance. They created the automation system with an Arduino Mega microcontroller and a Wi-Fi module, but they do not use a fuzzy logic controller for the temperature and relative humidity. They exploited the relay board as an actuator to ON or OFF the electrical appliances. In [3], the Internet of Things was covered, as well as how it can use to realize smart home automation utilizing a microcontroller-based Arduino board and an Android mobile application. It showed two prototypes, one for indoor home automated using Bluetooth and the other for outdoor home automated using Ethernet. The researchers in [18] presented a design of a temperature monitoring system with a DHT11 sensor, light-emitting diodes (LEDs), LCD i2c, Ethernet shield, and Arduino. The paper provided several tests on the previous tools, the first test is used for measuring the earth's surface air temperature, when it was below or equal to 25^{o}C, the red light of the LED would be OFF and the green one would be ON, which mean that it was cold air temperature. Otherwise, when the air temperature ranges above 25 ^{o}C the green LED's light would be OFF, and the red LED's light would be ON, meaning that the surface air temperature is hot. The other test was to compare digital-to-analog sensors. The lowest air temperature was 23^{o}C per day, and the highest one was 34^{o}C per day. Also, the designed system can provide information about the environmental and ambient earth's surface air temperature via telegram. In [19] a smart building model is presented that uses IoT technology to manage the performance of all technical systems to improve the efficiency of power energy. Furthermore, the researchers proposed an automated remote control approach using a cloud interface to increase the energy efficiency certification of existing buildings. This technology eliminates time-consuming procedures by storing the energy performance of each building on a cloud platform to display results and equipment measurements.

In this paper, we proposed the design and implementation of automatic smart buildings that control the temperature and relative humidity using fuzzy logic based on the Arduino microcontroller system. Here, the fuzzy logic system is used to make intelligent decisions using a set of rules that are applied to the input temperature and relative humidity parameters received from the DHT11sensor and controls the cooling system through the Ac mode output. The designed system also can monitor input data and control the output online via the IoT through the ESP8266 12-E Node MCU microcontroller. This paper is structured as follows: in addition to this introductory section, Section 2 contains the methodology of the paper, which contains an automatic control scheme, architecture, and system implementation. In Section 3, the results and discussion are given. Finally, Section 4 concludes this paper.

2. METHODOLOGY

A. Automatic Control Scheme

A control system plays an important role in many feedback control applications; this scheme is capable of monitoring, controlling, and influencing dynamic systems' operational conditions to ensure better closed-loop performance. Operating state conditions are generally connected to output parameters [13]. In this paper, the closed-loop control system for temperature and relative humidity has been simulated using a fuzzy logic controller and implemented on the Arduino UNO microcontroller, which controls the output of the cooling system through the relay module.

In addition, The I/O data can be monitored and controlled through ESP8266-12E Node MCU using ThingSpeak by a user by changing the relative running time of the relay to be ON according to the duty cycle for pulse width modulation (PWM) output signal. The system block diagram of the proposed closed-loop scheme is shown in Figure 1

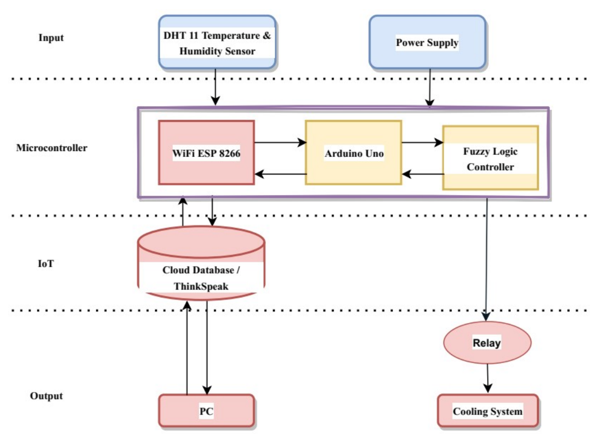


Figure 1. System block diagram of proposed closed-loop scheme

B. Architecture

The schematic diagram of the smart building temperature and relative humidity control system is shown in Figure 2. The proposed system has four main parts. They are the temperature and relative humidity sensitive sensor (DHT11), Arduino Uno microcontroller, the ESP8266-12E Node MCU microcontroller, and Fuzzy logic controller.

Firstly, the sensor collects the data from the building environment and sends it to the Arduino Uno microcontrollers, and ESP8266-12 Node MCU. The range of temperature

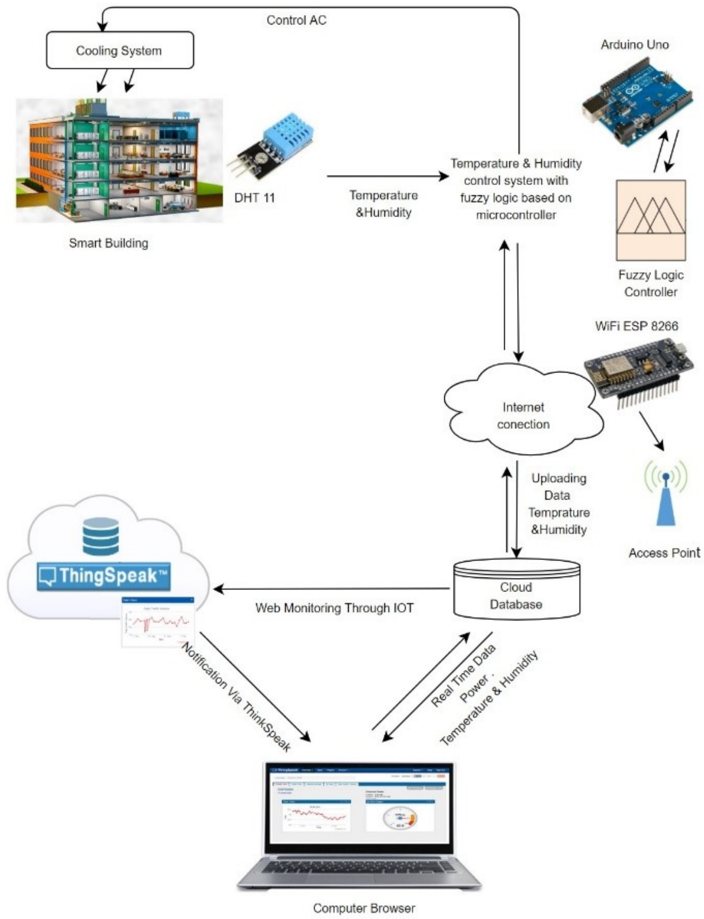


Figure 2. Schematic diagram of control smart building

and relative humidity values are set between 0^oC to 50^oC, and 0% to 90% respectively. These input data are analyzed and processed by the fuzzy logic controller to produce the desired Ac mode output values that control the cooling system by triggering the relay to be ON for a specific relative running time when the conditions are met.

The ESP8266-12E Node MCU uploads the I/O real-time data through a Wi-Fi connection to the IoT cloud server via the HTTP protocol, as well as generates the dataset for ThingSpeak. Anyone who checks the building control system's status via the website must log on using a user ID and password to access it, thereby ensuring data safety.

C. System Implementation

1) Data Flow Diagram:

The working model of the proposed system is shown in Figure 3 and it is explained below:

- The DHT11 sensor is a low-cost, high-reliability 4-pin sensor. Pin-1 is Vcc, while Pin-2 is a data pin that receives data from the outside and sends it to the Arduino microcontroller. The sensor detects the changes in ambient parameters and sends them to one of Arduino's digital input pins. The latter has 14 digital I/O pins, six of the outputs can be used as Pulse Width Modulation (PWM) pins.
- The digital value of input data is processed in the Arduino Uno microcontroller that regulates the work of the smart control system using a fuzzy logic controller that is simulated using MATLAB FIS file, which is implemented on an Arduino microcontroller by converting the FIS file to C programming IDE sketch file with a header file and using the eFLL library (Embedded Fuzzy Logic Library) that supports Mamdani's method.
- The cooling system is driven using the PWM signal generated by the Arduino microcontroller. PWM can be used to explain analogic, digital, and PWM(analog input reproduced from a digital output signal). This signal is used to determine the duration running time of the relay to be ON/OFF. The duration running time of the relay will be controlled according to the duty cycle with the range changing from 0% to 100%. The

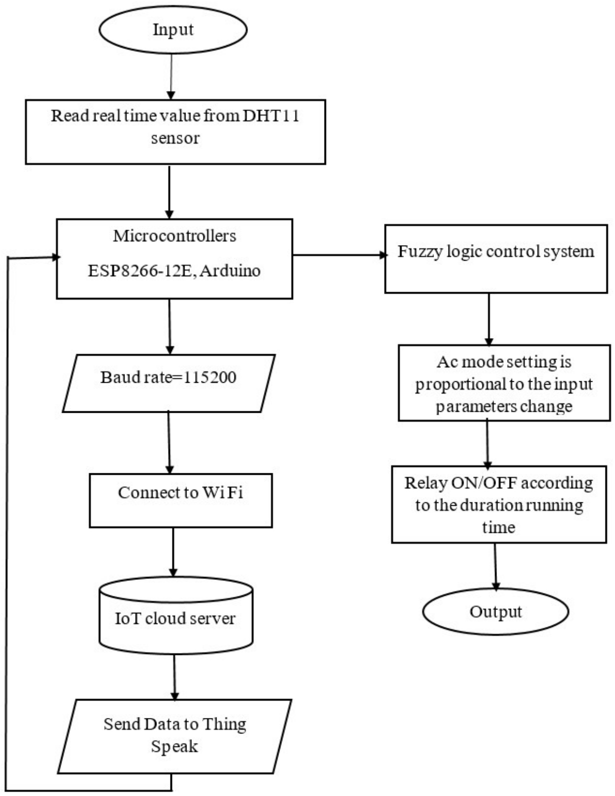


Figure 3. Data flow diagram of the proposed system

duty cycle means a particular time that a pulse is ON during the relay's operation cycle

- The real-time input values that are sensed by the sensor need to be transmitted to an IoT cloud server. This requires a Wi-Fi communication network by connecting to the ESP8266-12E Node MCU. This module has one analog and twelve digital I/O pins. It senses the changes in real-time I/O data and delivers them to the actual user continuously for monitoring and controlling as needed through a ThingSpeak platform from *www.ThingSpeak.com*. This web service provides open-source data analysis on the MATLAB platform with full profile access and it provides an easy way to view and analyze the data collected by field devices on the IoT cloud server. Simultaneously these real-time data is flashed on the Arduino IDE serial monitor

2) Fuzzy logic control system:

The schematic diagram of the fuzzy system is shown in Figure 4. It is composed of three major components fuzzification, fuzzy-inference-system, and defuzzification [20]. The fuzzy logic controller was designed and simulated utilizing Matlab 2020 FIS Editor. The system has two inputs: temperature and relative humidity, and one output called Ac mode.

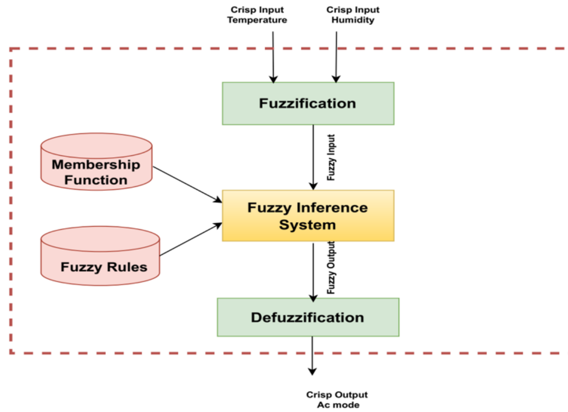


Figure 4. Schematic diagram of the fuzzy logic control system

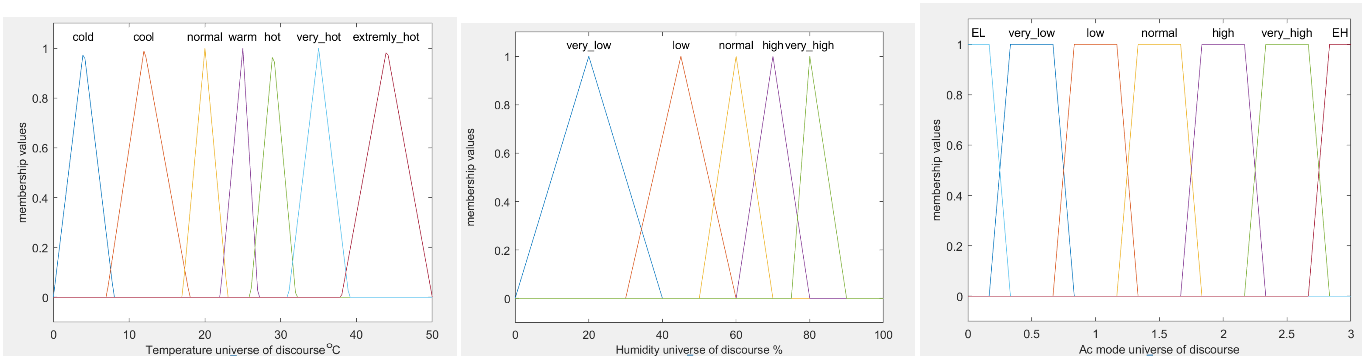
- Fuzzification: Crisp inputs to a fuzzy control system that are mapped to fuzzy sets these sets describe the various intervals that belong to the inputs. Fuzzification produces membership degree values ranging from 0 to 1.

- Fuzzy inference system: In this stage, we produced the outputs according to the given system inputs. A triangular membership function is utilized for the fuzzy controlling inputs system, while for the output a trapezoidal membership function is used. The work ranges of the DHT11 input sensor should be known to determine the rules and membership values of the fuzzy output.

- Defuzzification: Generates specific fuzzy output values that resulted from the inference system and crisps them to manage the value of temperature and relative humidity configuration and cooling system mode setting.

To obtain accurate fuzzy system results, Seven triangular fuzzy sets (Cold, Cool, Normal, Warm, Hot, Very Hot(VH_o), and Extremely Hot(EH_o)) are selected for input temperature with the universe of discourse [0,50], and five triangular linguistic fuzzy sets (Very Low(VL), Low(L), Normal(N), High(H), and Very High(VH)) are selected for relative humidity with the universe of discourse [0,90] as shown in Figure 5a and Figure 5b respectively. Seven trapezoidal fuzzy sets which are shown in Figure 5c (Extremely Low(EL), Very Low(VL), Low(L), Normal(N), High(H), Very High(VH), and Extremely High(EH)) are used for the Ac mode output variable in the universe of discourse [0,3].

After that fuzzy rules are generated, which are constructed depending on the IF-THEN type with AND operation. From the designed fuzzy sets of input variables, the total number of the possible rules will be 7*5=35 rules that are used to learn the fuzzy system as shown in Figure 6 Table I explains these rules in the ranges form of inputs and output fuzzy sets variables. Mamdani's method Type-1 is used to make decisions in compliance with model rules.

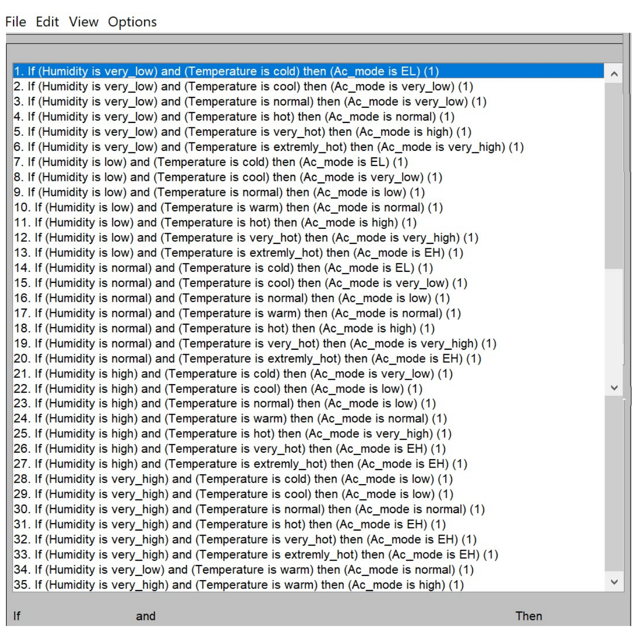


(a) Temperature input (b) Relative humidity input (c) Ac mode output

Figure 5. Membership functions of input and output variables.

TABLE I. 35 FUZZY RULES TO CONTROL COOLING SYSTEM

AND		**Temperature (^oC)**						
		cold ***0^{o}C - 8^{o}C***	***Cool*** ***7^{o}C - 18^{o}C***	***Normal*** ***17^{o}C - 23^{o}C***	***Warm*** ***22^{o}C-27^{o}C***	***Hot*** ***26^{o}C-32^{o}C***	***VH_o*** ***31^{o}C-39^{o}C***	***EH_o*** ***38^{o}C-50^{o}C***
Humidity (RH%)	***Very low less than 40%***	EL	VL	VL	N	N	H	VH
	Low 30% - 40%	EL	VL	L	N	H	VH	EH
	Normal 50% - 70%	EL	VL	L	N	H	VH	EH
	High 60% - 80%	VL	L	L	N	VH	EH	EH
	Very high more than 75%	L	L	N	H	EH	EH	EH


File Edit View Options

1. If (Humidity is very_low) and (Temperature is cold) then (Ac_mode is EL) (1)
2. If (Humidity is very_low) and (Temperature is cool) then (Ac_mode is very_low) (1)
3. If (Humidity is very_low) and (Temperature is normal) then (Ac_mode is very_low) (1)
4. If (Humidity is very_low) and (Temperature is hot) then (Ac_mode is normal) (1)
5. If (Humidity is very_low) and (Temperature is very_hot) then (Ac_mode is high) (1)
6. If (Humidity is very_low) and (Temperature is extremly_hot) then (Ac_mode is very_high) (1)
7. If (Humidity is low) and (Temperature is cold) then (Ac_mode is EL) (1)
8. If (Humidity is low) and (Temperature is cool) then (Ac_mode is very_low) (1)
9. If (Humidity is low) and (Temperature is normal) then (Ac_mode is low) (1)
10. If (Humidity is low) and (Temperature is warm) then (Ac_mode is normal) (1)
11. If (Humidity is low) and (Temperature is hot) then (Ac_mode is high) (1)
12. If (Humidity is low) and (Temperature is very_hot) then (Ac_mode is very_high) (1)
13. If (Humidity is low) and (Temperature is extremly_hot) then (Ac_mode is EH) (1)
14. If (Humidity is normal) and (Temperature is cold) then (Ac_mode is EL) (1)
15. If (Humidity is normal) and (Temperature is cool) then (Ac_mode is very_low) (1)
16. If (Humidity is normal) and (Temperature is normal) then (Ac_mode is low) (1)
17. If (Humidity is normal) and (Temperature is warm) then (Ac_mode is normal) (1)
18. If (Humidity is normal) and (Temperature is hot) then (Ac_mode is high) (1)
19. If (Humidity is normal) and (Temperature is very_hot) then (Ac_mode is very_high) (1)
20. If (Humidity is normal) and (Temperature is extremly_hot) then (Ac_mode is EH) (1)
21. If (Humidity is high) and (Temperature is cold) then (Ac_mode is very_low) (1)
22. If (Humidity is high) and (Temperature is cool) then (Ac_mode is low) (1)
23. If (Humidity is high) and (Temperature is normal) then (Ac_mode is low) (1)
24. If (Humidity is high) and (Temperature is warm) then (Ac_mode is normal) (1)
25. If (Humidity is high) and (Temperature is hot) then (Ac_mode is very_high) (1)
26. If (Humidity is high) and (Temperature is very_hot) then (Ac_mode is EH) (1)
27. If (Humidity is high) and (Temperature is extremly_hot) then (Ac_mode is EH) (1)
28. If (Humidity is very_high) and (Temperature is cold) then (Ac_mode is low) (1)
29. If (Humidity is very_high) and (Temperature is cool) then (Ac_mode is low) (1)
30. If (Humidity is very_high) and (Temperature is normal) then (Ac_mode is normal) (1)
31. If (Humidity is very_high) and (Temperature is hot) then (Ac_mode is EH) (1)
32. If (Humidity is very_high) and (Temperature is very_hot) then (Ac_mode is EH) (1)
33. If (Humidity is very_high) and (Temperature is extremly_hot) then (Ac_mode is EH) (1)
34. If (Humidity is very_low) and (Temperature is warm) then (Ac_mode is normal) (1)
35. If (Humidity is very_high) and (Temperature is warm) then (Ac_mode is high) (1)

If and Then

Figure 6. Fuzzy rules from Matlab FIS editor.

3. RESULTS AND DISCUSSION

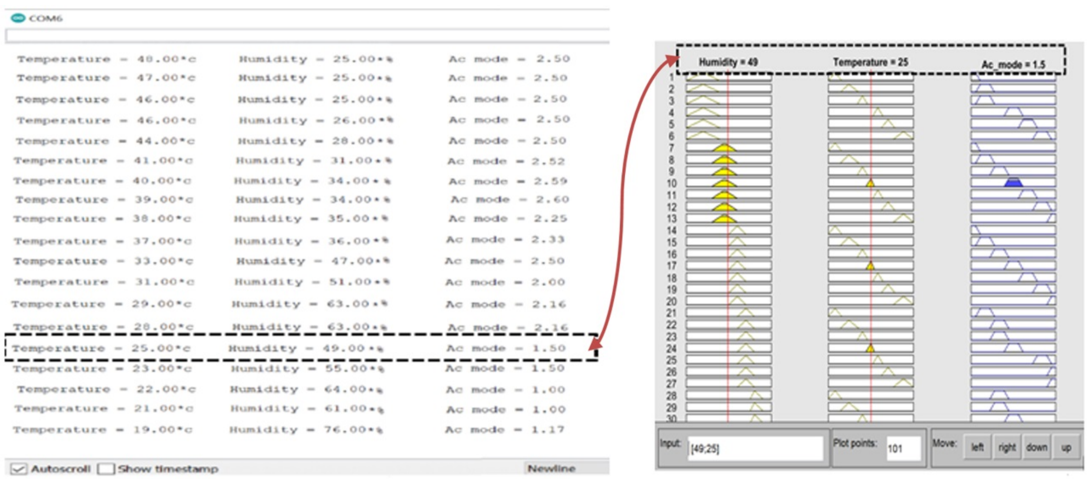
The proposed system confirmed more efficiency than the human manual control system in terms of automatic control using the fuzzy logic system. The fuzzy logic evaluation system after learning is performed using the Matlab FIS simulation Editor and some values of the simulated I/O results are presented in Table II.

After implementing the fuzzy logic control system on the Arduino UNO circuit. we compare the simulation results that are obtained from the Matlab fuzzy logic system with that of implemented fuzzy logic control system on a microcontroller as shown in Figure 7. The results showed matching between them, for example, when the temperature is 25 ^oC and relative humidity is 49% the Ac mode is 1.5 as shown in the dashed line in this figure. Thus the system was successfully designed and implemented.

The real-time I/O data of the proposed system is displayed on the ESP8266-12E Node MCU serial monitor as shown in Figure 8 and Simultaneously send to the IoT cloud server of ThingSpeak. Some of these data values obtained from the ThingSpeak channel have been plotted in Figure 9 which illustrates some input data values for Temperature in ^oC in Figure 9 a, Relative Humidity RH% in Figure 9 b, and relative running time output parameters in Figure 9 c for the specific Ac mode.

TABLE II. THE I/O RESULTS AND OUTPUT CONDITIONS OF THE DESIGNED FIS SIMULATOR.

Input		Output	
Temp. [°C]	*Relative Humidity [RH%]*	*Ac mode*	*Ac mode condition*
8	72	0.73	Low
18	37	0.83	Very Low
21	61	1	Low
22	64	1	Low
25	49	1.5	Normal
28	62	2.1	High
33	47	2.5	Very High
37	35	2.3	Very High
38	35	2.2	High
40	34	2.5	Very High
42	62	2.85	Extremely High
44	28	2.5	Very High
46	90	2.9	Extremely High
48	25	2.5	Very High



(a) Arduino IDE serial monitor COM6 (b) Matlab FIS simulation viewer

Figure 7. Results of fuzzy logic controller implemented on ARDUINO microcontroller vs. fuzzy logic using Matlab FIS viewer.

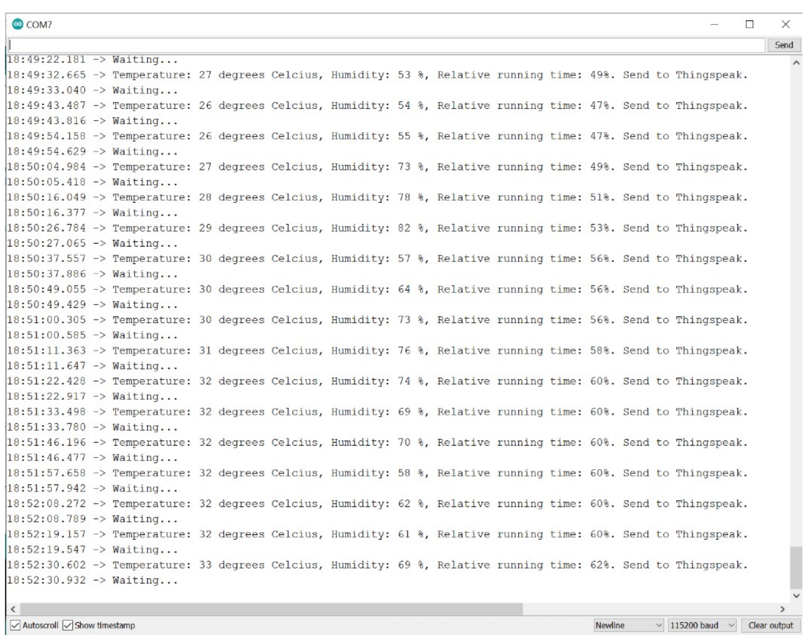
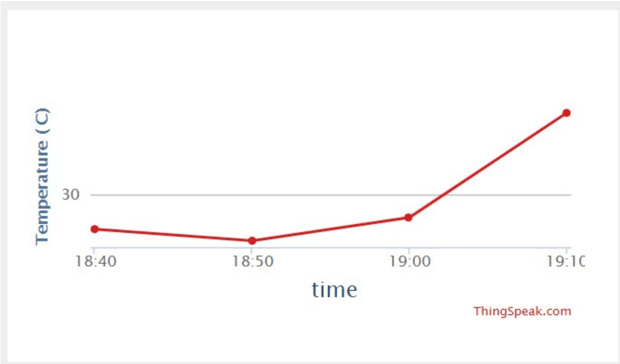
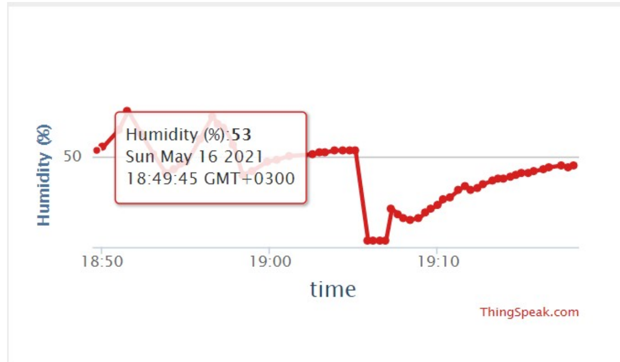


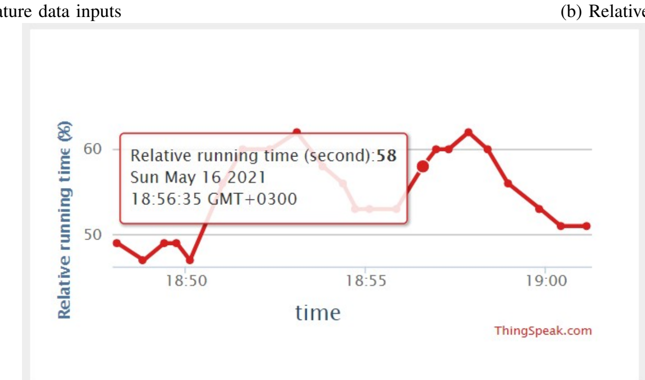
Figure 8. Some of real-time I/O data displayed on ESP8266-12E Node MCU serial monitor.



(a) Different temperature data inputs



(b) Relative humidity data inputs



(c) Relative running time output

Figure 9. Implementation system results from the ThingSpeak platform.

Besides, the designed ThingSpeak channel shows good performance in that it needs just some milliseconds to transfer the real-time I/O data into the IoT cloud server. Table III shows some values of the obtained output data for different input parameters, which clarify that the relative running time controls the relay operation to be ON for the specific Ac mode output.

4. CONCLUSION

In this research, a prototype of an automatic smart cooling system using the Internet of things (IoT) is presented. This study workout by integrating relays with a Microcontroller board to control building appliances from different remote locations in a real-time manner. The advantage of the designed system is to reduce the loss of energy, and human effort by using the fuzzy logic controller to make a decision for the appropriate state of operation. The real-time I/O data can be sent and stored on an IoT cloud server for monitoring and controlling through the ThingSpeak. These parameters can be viewed and analyzed by plotting them for use in future analysis to help users manage the system more efficiently.

Based on the comparison results between Figure 7 and Table II, we concluded that the suggested fuzzy logic system implemented on the Arduino UNO microcontroller was designed and performed successfully and the results showed matching between it and FIS simulation using the Matlab program.

References

[1] R. J. Hassan, S. Zeebaree, S. Y. Ameen, S. F. Kak, M. Sadeeq, Z. S. Ageed, A. Adel, and A. A. Salih, "State of art survey for iot effects on smart city technology: challenges, opportunities, and solutions," *Asian Journal of Research in Computer Science*, vol. 22, pp. 32–48, 2021.

[2] Y. Fang, T. Tang, Y. Li, C. Hou, F. Wen, Z. Yang, T. Chen, L. Sun, H. Liu, and C. Lee, "A high-performance triboelectric-electromagnetic hybrid wind energy harvester based on rotational tapered rollers aiming at outdoor iot applications," *Iscience*, vol. 24, no. 4, p. 102300, 2021.

[3] K. Mandula, R. Parupalli, C. A. Murty, E. Magesh, and R. Lunagariya, "Mobile based home automation using internet of things (iot)," in *2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*. IEEE, 2015, pp. 340–343.

[4] J. M. S. Waworundeng, M. A. T. Kalalo, and D. P. Y. Lokollo, "A prototype of indoor hazard detection system using sensors and iot," in *2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS)*. IEEE, 2020, pp. 1–6.

[5] D. F. Melo, B. M. Silva, N. Pombo, and L. Xu, "Internet of things assisted monitoring using ultrasound-based gesture recognition contactless system," *IEEE Access*, vol. 9, pp. 90 185–90 194, 2021.

TABLE III. SOME REAL TIME I/O DATA OBTAINED FROM THINGSPEAK.

Time	Temperature (^{o}C)	Relative Humidity (RH%)	Relative running time %
2021-05-16 18:48:50 +03	26	53	47
2021-05-16 18:49:24+ 03	27	53	49
2021-05-16 18:51:02 +03	30	64	56
2021-05-16 18:53:49 +03	31	41	58
2021-05-16 18:57:18 +03	32	65	60
2021-05-16 18:57:51 +03	33	56	62
2021-05-16 18:59:51 +03	29	47	53
2021-05-16 19:00:25 +03	28	48	51
2021-05-16 19:02:37 +03	27	51	49
2021-05-16 19:12:21 +03	32	33	60
2021-05-16 19:11:16 +03	33	32	62
2021-05-16 19:10:43 +03	34	28	64
2021-05-16 19:10:00 +03	37	24	71
2021-05-16 19:08:54 +03	41	17	80

[6] G. Nagaraja, A. B. Soppimath, T. Soumya, and A. Abhinith, "Iot based smart agriculture management system," in *2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, vol. 4. IEEE, 2019, pp. 1–5.

[7] A. Najmurrokhman, A. Daelami, E. Nurlina, U. Komarudin, H. Ridhatama *et al.*, "Development of temperature and humidity control system in internet-of-things based oyster mushroom cultivation," in *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*. IEEE, 2020, pp. 551–555.

[8] H. Gupta, D. Bhardwaj, H. Agrawal, V. A. Tikkiwal, and A. Kumar, "An iot based air pollution monitoring system for smart cities," in *2019 IEEE International Conference on Sustainable Energy Technologies and Systems (ICSETS)*. IEEE, 2019, pp. 173–177.

[9] L. Ada, "Dht11, dht22 and am2302 sensors. adafruit learning system," 2021.

[10] F. Purwanto, E. Utami, and E. Pramono, "Implementation and optimization of server room temperature and humidity control system using fuzzy logic based on microcontroller," in *Journal of Physics: Conference Series*, vol. 1140, no. 1. IOP Publishing, 2018, p. 012050.

[11] G. Oltean and L.-N. Ivanciu, "Implementation of a fuzzy logic-based embedded system for temperature control," in *2017 40th International Spring Seminar on Electronics Technology (ISSE)*. IEEE, 2017, pp. 1–6.

[12] N. S. Pezol, R. Adnan, and M. Tajjudin, "Design of an internet of things (iot) based smart irrigation and fertilization system using fuzzy logic for chili plant," in *2020 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*. IEEE, 2020, pp. 69–73.

[13] J. Bhor, S. Pawar, and S. Das, "Design and development of temperature control loop of hair dryer using arduino," in *2020 International Conference on Industry 4.0 Technology (I4Tech)*. IEEE, 2020, pp. 61–64.

[14] K. S. Kiran, R. K. Devisetty, N. P. Kalyan, K. Mukundini, and R. Karthi, "Building a intrusion detection system for iot environment using machine learning techniques," *Procedia Computer Science*, vol. 171, pp. 2372–2379, 2020.

[15] A. K. Podder, A. Al Bukhari, S. Islam, S. Mia, M. A. Mohammed, N. M. Kumar, K. Cengiz, and K. H. Abdulkareem, "Iot based smart agrotech system for verification of urban farming parameters," *Microprocessors and Microsystems*, vol. 82, p. 104025, 2021.

[16] M.-C. Yuen, S. Y. Chu, W. H. Chu, H. S. Cheng, H. L. Ng, and S. P. Yuen, "A low-cost iot smart home system," *Int. J. Eng. Technol*, vol. 7, pp. 3143–3147, 2018.

[17] W. A. Jabbar, M. H. Alsibai, N. S. S. Amran, and S. K. Mahayadin, "Design and implementation of iot-based automation system for smart home," in *2018 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, 2018, pp. 1–6.

[18] M. W. Hariyanto, A. H. Hendrawan, and R. Ritzkal, "Monitoring the environmental temperature of the arduino assistance engineering faculty using telegram," *Journal of Robotics and Control (JRC)*, vol. 1, no. 3, pp. 96–101, 2020.

[19] C. K. Metallidou, K. E. Psannis, and E. A. Egyptiadou, "Energy efficiency in smart buildings: Iot approaches," *IEEE Access*, vol. 8, pp. 63 679–63 699, 2020.

[20] F. Pradityo and N. Surantha, "Indoor air quality monitoring and controlling system based on iot and fuzzy logic," in *2019 7th International conference on information and communication technology (ICoICT)*. IEEE, 2019, pp. 1–6.

Heba N. Y. Al-Talb received her BSc in 2002 and MSc degrees in 2010 from the College of Engineering, Computer Engineering Department from the University of Mosul. She worked as an assistant lecturer at the University of Mosul. Since 2015, she has moved to Ninevah University, College of Electronics Engineering, Computer and Information Engineering Department. Giving lectures in Computer science, microprocessor, and microcontroller.

Sama N. M. Al-Faydi is a lecturer at Nineveh University Computer and Information Engineering Department. She studied Computer Engineering at Mosul University in 2000 and also received her MSc degree- in Computer Engineering from Mosul University in 2011.

Toka A. Fathi a lecturer at Nineveh University Computer and Information Engineering Department. She studied Computer Engineering at Mosul University in 2006 and also received her MSc degree- in Computer Engineering from Mosul University in 2011.

Maan A. S. Al-Adwany received his BSc degree in 1997 from Mosul University, College of Engineering, Electrical Engineering Dept., also received his MSc degree in 2000 from Electronics and Communications Engineering in the same institution. He was promoted to a lecturer degree- in 2007, an assistant professor degree- in 2013, and a professor degree- in 2022.