RESEARCHING AND CONSTRUCTING A CONTROL AND MONITORING SYSTEM FOR POULTRY EGG INCUBATORS

Nguyen Thi Thu

University of Economic and Technical Industries, Hanoi, Vietnam

ARTICLE INFORMATION

Journal: *Vinh University Journal of Science Natural Sciences, Engineering and Technology*
p-ISSN*: 3030-4563*
e-ISSN: *3030-4180*

Volume: 53
Issue: 2A
****Correspondence:***
ntthu.khcb@uneti.edu.vn
Received: *15 March 2024*
Accepted: *25 April 2024*
Published: *20 May 2024*

Citation:
Nguyen Thi Thu (2024). Researching and constructing a control and monitoring system for poultry egg incubators. ***Vinh Uni. J. Sci.*** *Vol. 53 (2A), pp. 110-121*
doi: 10.56824/vujs.2024a052a

ABSTRACT

This article focuses on researching and constructing a system for controlling and monitoring important parameters related to the poultry egg incubation, such as temperature, humidity, egg turning, light, ventilation, and egg cooling, applying network connectivity technologies and utilizing advanced microcontrollers. The system's functions meet real-time requirements such as: real-time data from the incubation chamber can be controlled and monitored directly or remotely via a personal computer. Through this, users can adjust the parameters on the system to achieve the highest success rate in the incubation process. Experimental results show that the system operates stably with a hatch rate of nearly 95%, meeting the requirements set out by the design research. The actual product is highly robust, easy to install, operate, maintain, and service.

Keywords: Egg incubator; ESP32; monitor; control; Internet of Things.

1. Introduction

Artificial incubation using industrial incubators is the optimal method for producing hatchlings in a short time with a high hatching rate. This approach allows for the incubation of a large number of eggs while improving the quality of the hatchlings. Today, integrating microcontrollers into industrial production lines has effectively met the requirements for automation and monitoring, becoming a key factor in further enhancing the efficiency of our country's industrial production. The incorporation of microcontrollers in industrial egg incubators is essential as it ensures precise control of incubation parameters, resulting in high hatching efficiency. Additionally, the control system is lightweight, easy to operate, and provides significant economic benefits to farmers.

In current research and published products on egg incubators, there have been significant improvements. Specifically, Hiep L. H. and colleagues utilized microcontrollers and sensors, then designed and tested on the LabVIEW platform [1]; Adnyana and colleagues used

Raspberry Pi as the main tool to design the system's hardware [2]; and P. Bhosale and colleagues used Arduino as the main component for their design [3]. The advantages of these methods include reducing the labor of farmers and improving the hatching rate. However, they still have limitations in egg turning, resulting in a hatching rate that is not as high as desired. The designs from these studies are quite complex and the equipment cost is high. Moreover, these designs are suited to the local conditions of the authors of the publications, which are not in Vietnam.

In this study, the surveys and practical research were conducted in various geographical areas in Northern Vietnam. The local climate, weather, and economic conditions of farmers households have been considered to design appropriate products. Based on the outstanding features of ESP32 such as its suitability for advanced projects requiring high processing and connectivity capabilities (such as IoT devices), smart home applications, and industrial automation, it provides enhanced features and powerful processing capabilities suitable for complex applications [4-10]. Therefore, in this study, ESP32 was used as the main hardware platform along with some low-cost microchips and sensors, making the system easy to build, repair, and install. This is suitable for farmers with limited financial resources while still ensuring high quality and efficiency (with a 10% increase in hatching rate). Additionally, the product has a simple design, allowing users to operate and maintain it with basic instructions.

2. Methodology

2.1. Hardware and Programming Tools

2.1.1. Hardware components

To meet the specified requirements, the following hardware components were selected [2-4]:

- Central controller: ESP32.
- Temperature sensor: LM35.
- Incandescent lamp: 300W.
- Driver circuit: Mosfet driver for the incandescent lamp.
- Display: LCD for parameter display.
- Adjustable resistor: For fine-tuning the set temperature.
- Power supply unit: Rectifier to convert 220VAC to 12VDC, and from 12VDC to 5VDC.
- Control system: PID controller for monitoring and controlling the thermal environment.

These components were chosen to ensure precision in temperature regulation, cost-effectiveness, and ease of maintenance and assembly, tailored to the needs of local farmers.

2.1.2. Programming tools

To develop the software interface running on the computer, the C# programming language was used. This language was chosen due to its numerous advantages and widespread use, making it suitable for this research.

2.2. Hardware design

2.2.1. Model for controlling and monitoring the egg incubator

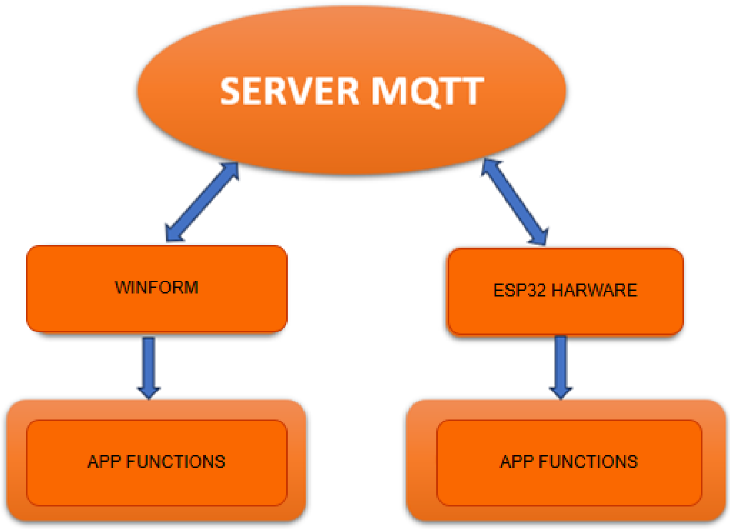

Figure 1: *The design of the MQTT system model*

The system model (Figure 1) is constructed based on the MQTT protocol, with the main components being the MQTT Server, Winform (PC application), and the ESP32 hardware [5-9].

Operation principles of the MQTT server:

- Connection Environment: The MQTT Server serves as the connection hub for clients to exchange information.

- Receiving Data from Clients: The MQTT Server receives data from clients, which includes a data segment and a topic specified by the MQTT protocol standard.

- Data Distribution: The received data is distributed to other clients based on the topics they have subscribed to.

Operation principles of Winform:

- Connecting and Subscribing to Topics: Winform connects to the MQTT Server and subscribes to receive data packets based on a specific topic, which the ESP32 transmits to the MQTT Server.

- Processing and Displaying Data: Upon receiving data from the MQTT Server, Winform processes and displays the information on the Winform interface.

- Sending Data to the MQTT Server: Winform can send data to the MQTT Server, including both the data and the topic, where the topic is the one the ESP32 subscribes to for receiving data.

Operation principles of ESP32 Hardware:

- Microcontroller Functionality: ESP32 performs the standard functions of a microcontroller.

- Connecting and Subscribing to Topics: ESP32 connects to the MQTT Server and subscribes to a topic to receive data, with the topic being the one subscribed to by Winform.

- Sending and Receiving Data: ESP32 sends data to the MQTT Server, including both the data and the topic, and then receives data from the MQTT Server and processes the received data.

2.2.2. Schematic design and system components

The schematic diagram for controlling the brightness of the incandescent lamp is shown in Figure 2.

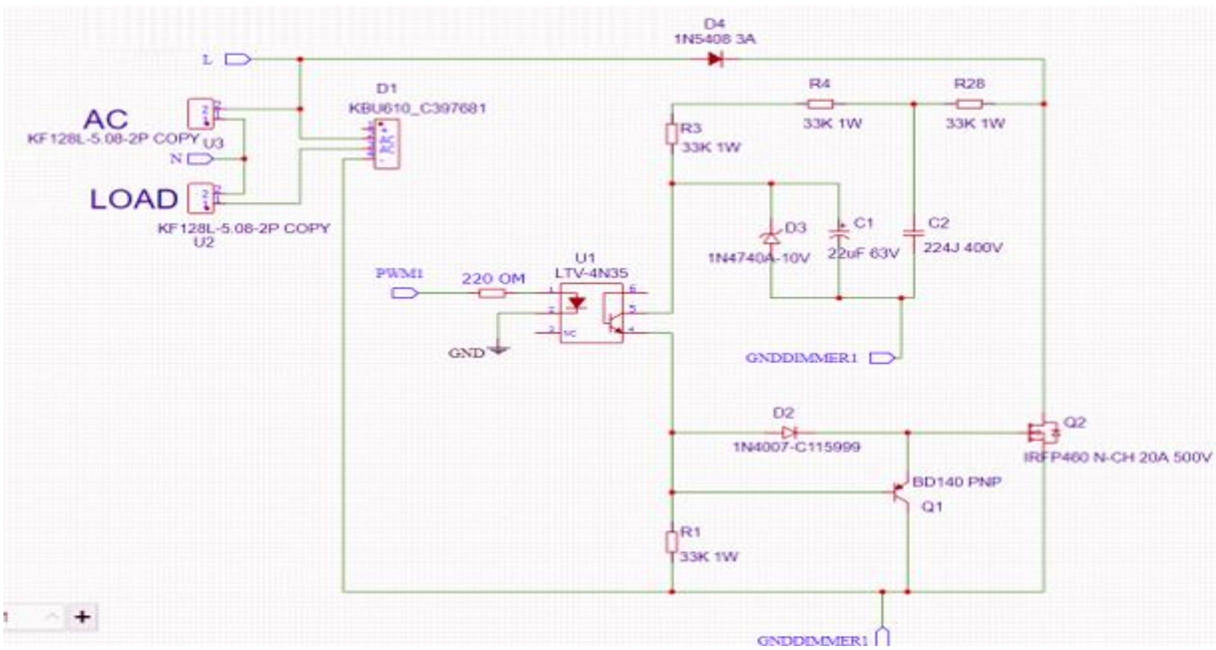

Figure 2: *Schematic diagram of the incandescent lamp load circuit*

The microcontroller's function of generating PWM pulses can be used to adjust the brightness of the lamp by controlling the Mosfet to open and close at high frequencies. When the microcontroller generates PWM pulses with varying widths, the Mosfet will rapidly open and close according to the PWM frequency. When the Mosfet is open, it allows current to flow through the lamp, causing it to illuminate. When the Mosfet is closed, current cannot flow through the lamp, causing it to turn off. By changing the width of the PWM pulses, the duration that the Mosfet is open relative to the duration it is closed is altered, thus affecting the lamp's brightness. This allows for flexible adjustment of the lamp's brightness through varying the PWM pulse width. Consequently, as the microcontroller generates PWM pulses with varying widths, the Mosfet adjusts the lamp's brightness by controlling the duration it remains open and closed, thereby creating a flexible brightness adjustment effect.

The schematic diagram of the potentiometer and LM35 connection is shown in Figure 3.

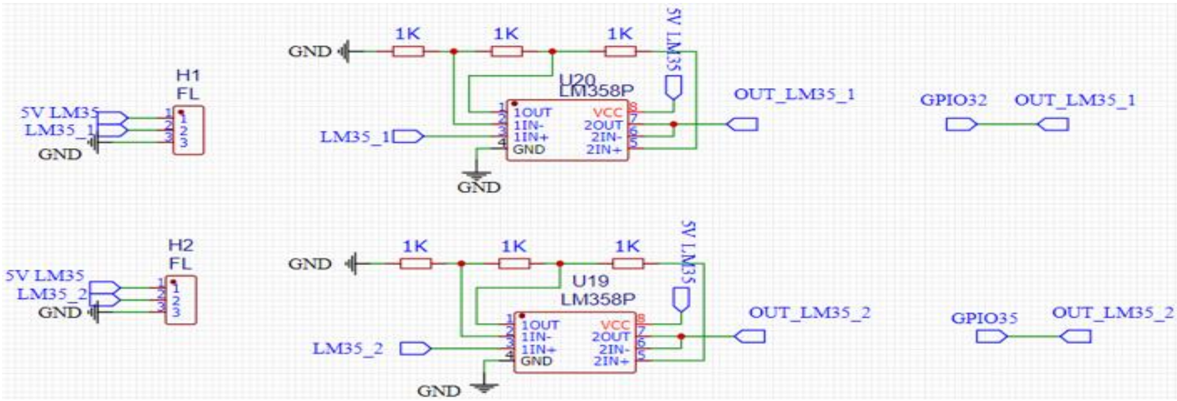

Figure 3: *Connection of the variable resistor - LM35*

The circuit design on the EasyEDA software is depicted in Figure 4:

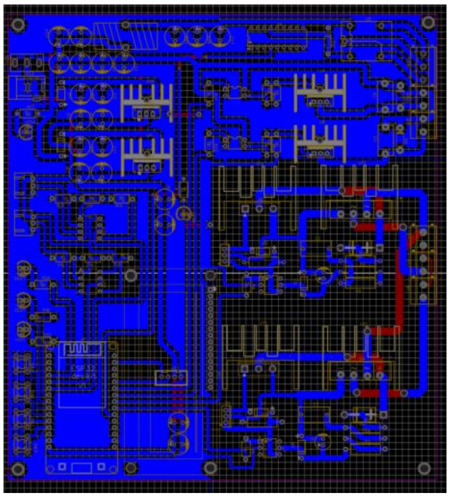

Figure 4: *Designing a 3D model for the system circuit using EasyEDA software*

The actual circuit prototype is shown in Figure 5:

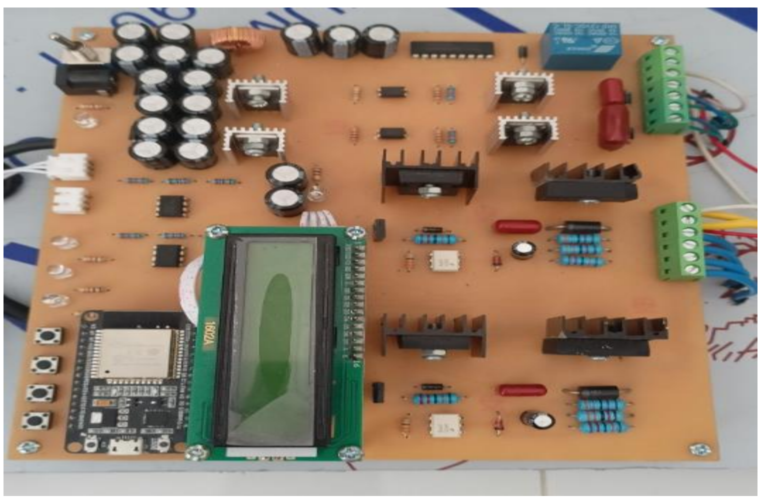

Figure 5: *The real-life connection circuit*

2.2.3. Programming for ESP32

Through surveys and real product research based on ESP32, there will be several programs as follows [10-14]:

- Establish WiFi connection for ESP32 to perform data transmission to SCADA through MQTT server.
- Program for connecting to MQTT Server.
- Program for transmitting data to MQTT Server.
- Program for receiving data from MQTT Server.
- PID temperature control program for egg incubator.

a. Hardware pin connection diagram

All GNDs throughout the circuit are connected except for the GND of the load-pulling power circuit, as detailed in Table 1.

Table 1: *Circuit connection diagram with ESP32*

DRIVER	ESP32	Button	GPIO14
5V LM2596	VIN, VCC LCD	OUTPUT LM35	IN1 L298
3.3V LM1117	3 Variable Resistor and LM35	SDA LCD	IN2 L298
Variable Resistor 1	GPIO35	SCL LCD	IN3 L298
Variable Resistor 2	GPIO32	PWM ĐÈN	IN4 L298
Variable Resistor 3	GPIO33	ENA L298	ENB L298

b. The WiFi connection program for ESP32

```
void ConnectWifi()
{
 ticker.attach (0.2, tick4);
 int count = 0;
 Wifi.begin (ssid, pass);
 while (Wifi.status () != WL_CONNECTED)
 {
   delay (500);
   Serial.print(".");
   count++;
   if (count>=60)
   {
		ESP.restart ();
		count = 0;
		}
 }
 Serial.println ();
 Serial.println("Connect Wifi");
 Serial.print("Address IP esp: ");
 Serial.println(Wifi.localIP());
 ticker.detach();
 }
```

c. The MQTT server connection program

```
void ConnectMqtt()
```

```
{
                client.setServer(mqtt_server, mqtt_port); //set ESP client ket noi MQTT broker
                delay(10);
                client.setCallback(callback); // doc du lieu MQTT broker ma ESP subscribe
                delay(10);
                }
```

d. The program to transmit data to an MQTT server

```
DataJson();
 boolean a = client.publish(topicpub.c_str(), DataMqttJson.c_str());
 Serial.printf("Publishing:%d\r\n===============\r\n", a);
```

e. The program to receive data from an MQTT server

```
void ConnectMqtt()
 {
  client.setServer(mqtt_server; mqtt_port); //set ESP client ket noi MQTT broker
  delay(10);
  client.setCallback(callback); //doc du lieu MQTT broker ma ESP subcribe
  delay(10);
  }
```

f. The PID temperature control program for the egg incubator

```
int PID(int nhietdodat, int nhietdothuc)
{
                E=nhietdodat - nhietdothuc;
                alpha = 2 * T * Kp + Ki * T * T + 2 * Kd;
                beta = T * T * Ki - 4 * Kd - 2 * T + Kp;
                gama = 2 * Kd;
                Output = (alpha * E + beta * E1 + gama * E2 + 2 * T * LastOutput) / (2 *
T);
                LastOutput = Output;
                E2 = E1;
                E1 = E;
                // PWM => 0 -250
                if (Output > 250)
                {
                Output = 250;
                }
                else if (Output < - 250)
                {
                Output = 250;
                }
                else if (Output <= 0 && Output >= -250)
                {
                Output = (-1) * Output;
                }
                else if (Output >= 0 && Output <= 250)
                {
                Output = (1) * Output;
```

g. The algorithm flowchart

The algorithm flowchart for the ESP32 system is designed as shown in Figure 6.

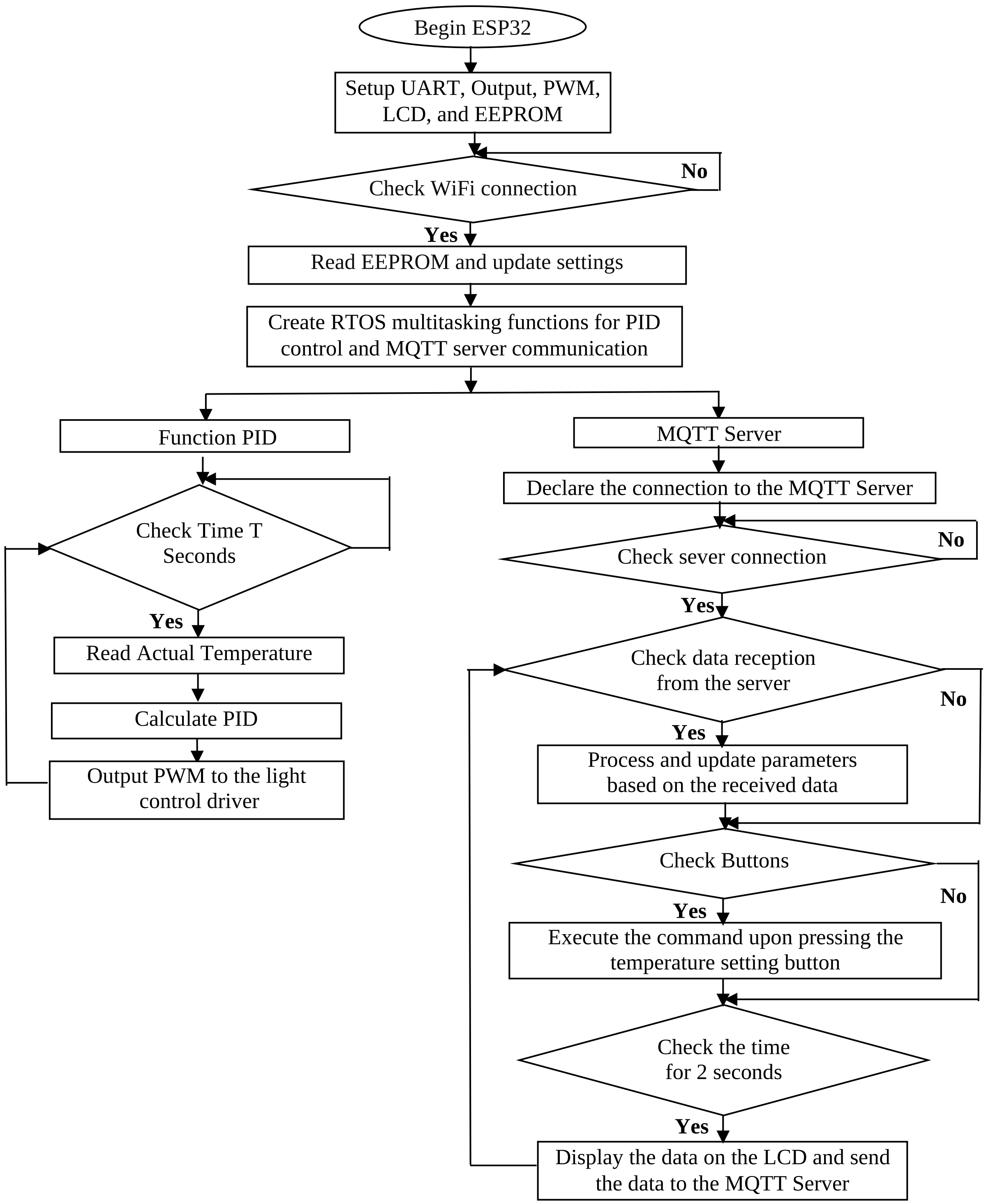

Figure 6: *The algorithm flowchart for the ESP32 system*

h. Design and programming of the Winform interface

The software interface is programmed and designed as shown in Figure 7.

Figure 7: *Software interface for monitoring and controlling egg incubator temperature*

2.3. Experimental evaluation

Actual product model, shown in Figure 8, with detailed parameters as follows: *(1): Cooling fan; (2): 60W incandescent lamp; (3): LM35 temperature sensor; (4): Egg turning frame; (5): Mica frame; (6): 12V 5A adapter; (7): Electronic circuit board.*

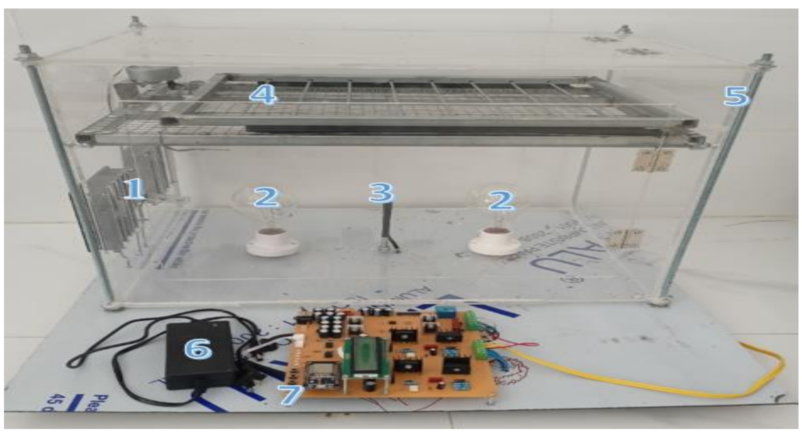

Figure 8: *Actual product design and programming*

Table 2 shows the comparison results between experimental cages and traditional cabinet-like cages (2) and room-like cages (3).

Table 2: *Comparison of cage parameters*

Parameters	Experimental Cage	Type 2 Cage	Type 3 Cage
Material	Mica	Wood	Brick
Construction cost	1.300.000đ (*)	4.500.000đ	70.000.000đ
Maximum capacity (eggs)	50	7.000	100.000
Electricity cost / month	40.000đ	450.000đ	2.000.000đ
Output / Input egg ratio	16/17	500/1000	400/1000
Hatch rate	94%	50%	40%

Cages type 2 and type 3 were rented and utilized at the actual egg farm of livestock households. According to the information in Table 2, the total construction cost of the cages is approximately 1,100,000 VND, but an additional 20% of the selling price will be added to make it suitable for practical use. However, this is only a model using mica material for easy experimentation to assess the feasibility of the project, so the initial construction cost will be somewhat higher. When applied in practice, cheaper materials such as wood or brick will be used while still ensuring quality, resulting in a significantly reduced construction cost. If the experimental model is applied in practice, for example in cage type 2, an additional amount of approximately 30,000,000 VND (cost for the electronic system and egg turning) would be required to compensate for the 20% ratio. The wholesale price of duck eggs is currently 3,500-3,900 VND/egg and the price of ducklings is 13,000-13,500 VND/duckling. It is evident that in less than 2 months, the initial construction investment cost can be recovered, and the profit will significantly increase monthly. Similar to when applied to cage type 3.

4. Conclusion

The experimental model has yielded quite favorable results by increasing the hatch rate as well as embryos maintenance, ensuring a stable output of eggs, and significantly reducing human effort. It has partially improved the economic situation for poultry farming households. The research product has essentially met the set objectives. The research product has achieved high efficiency, with greater accuracy and much lower cost compared to currently announced products.

REFERENCES

[1] Le Hoang Hiep, "Study to build a control and monitoring system of poultry incubator based on Internet of Things", *TNU Journal of Science and Technology,* vol. 227, no. 08, pp. 20-28, 2022.

[2] J. C. Zhao, J. F. Zhang, Y. Feng, and J. X. Guo, "The study and application of the IOT technology in agriculture," *Computer Science and Information Technology (ICCSIT) 3rd IEEE International Conference*, pp. 462-465, 2010.

[3] G. S. Adnyana, I. N. Piarsa, and K. S. Wibawa, "Internet of Things: Control and Monitoring System of Chicken Eggs Incubator Using Raspberry Pi," *International Journal of Internet of Things*, vol. 7, no. 1, pp. 16-21, 2018.

[4] L. Niranjan et al., "Design and implementation of chicken egg incubator for hatching using IoT," *Int. J. Computational Science and Engineering*, vol. 24, no. 4, pp. 363-372, 2021.

[5] K. Thear, *Incubation a Guide to Hatching and Rearing*, Broad Leys Publishing Limited, 2010.

[6] I. C. Boleli et al., "Poultry egg incubation: integrating and optimizing production efficiency," *Brazilian Journal of Poultry Science*, vol. 18, no. 2, pp. 1-16, 2016.

[7] O. E. Aru, “Development of a Computerized Engineering Technique to Improve Incubation System in Poultry Farms,” *Journal of Scientific and Engineering Research*, vol. 4, no. 6, pp. 109-119, 2017.

[8] W. S. M. Sanjaya et al., “The development of quail eggs smart incubator for hatching system based on microcontroller and Internet of Things (IoT),” *International Conference on Information and Communications Technology (ICOIACT)*, 2018, pp. 407-411, doi: 10.1109/ICOIACT.2018.8350682.

[9] P. E. Ohpagu and A. W. Nwosu, “Development and Temperature Control of Smart Egg Incubator System for Various Types Egg,” *European Journal of Engineering and Technology*, vol. 4, no. 2, pp. 13–21, 2016.

[10] K. Radhakrishnan et al., “Design and Implementation of a Fully Automated Egg Incubator,” *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 3, no. 2, pp. 7686-7690, 2014.

[11] P. Bhosale et al., “Development of Smart Egg Incubator System Using Arduino,” *International Journal of Engineering Science and Computing*, vol. 8, no. 3, pp. 16598-16600, 2018.

[12] S. Sofyan et al., “Monitoring Incubator Space of Chicken Telemetry-Based Chicken Hatching Using Arduino Uno R3,” (in Indonesian), *National Journal of Electrical Engineering,* vol. 5, no. 1, pp. 26-35, 2016.

[13] A. A. Aldair et al., “Design and implementation of intelligent control system for egg incubator based on IoT technology,” *4th International Conference on Electrical, Electronics and System Engineering (ICEESE),* 2018, doi: 10.1109/ICEESE.2018.8703539.

[14] M. S. Murugan, L. Srikanth, and V. P. S. Naidu, “Design and development of LabVIEW based environmental test chamber controller,” *International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, Mysuru, 2017, pp. 1-4, doi: 10.1109/ICEECCOT.2017.8284638.

TÓM TẮT

NGHIÊN CỨU XÂY DỰNG HỆ THỐNG ĐIỀU KHIỂN VÀ GIÁM SÁT LỒNG ẤP TRỨNG GIA CẦM

Nguyễn Thị Thu
Trường Đại học Kinh tế - Kỹ thuật Công Nghiệp, Hà Nội, Việt Nam
Ngày nhận bài 15/3/2024, ngày nhận đăng 25/4/2024

Bài báo này tập trung nghiên cứu và xây dựng một hệ thống với chức năng điều khiển và giám sát các thông số quan trọng liên quan tới quá trình ấp trứng gia cầm như nhiệt độ, độ ẩm, khâu đảo trứng, ánh sáng, thông gió và làm mát trứng, ứng dụng các công nghệ mạng kết nối vạn vật và sử dụng các vi điều khiển tiên tiến. Các chức năng của hệ thống đáp ứng được các yêu cầu theo thời gian thực như: dữ liệu thực tế từ lồng ấp có thể điều khiển, theo dõi trực tiếp hoặc từ xa thông qua máy tính cá nhân. Thông qua đó, người sử dụng có thể điều chỉnh các thông số trên hệ thống để quá trình ấp trứng đạt được tỉ lệ thành công cao nhất. Kết quả thực nghiệm cho thấy hệ thống hoạt động ổn định, tỉ lệ trứng nở đạt gần 95%, đáp ứng được các yêu cầu đề ra theo nghiên cứu thiết kế. Sản phẩm thực tế có độ chắc chắn cao, dễ lắp đặt, vận hành, bảo dưỡng và bảo trì khi sử dụng.

Từ khóa: Lồng ấp trứng; ESP32; giám sát; điều khiển; mạng kết nối vạn vật.