**徽标

描述已自动生成Electrical &Computer Engineering**

**ECE590 IOT Final Report 2024 Spring**

**Name: Yi Tong**

**Net ID: yt210**

**Contents**

1. Introduction and motivation
2. System Design
3. System Implementation
4. Experiment and evaluations
5. Discussion and conclusion

# Introduction and Motivation

## 1.1 Introduction

This project focuses on developing an automated system capable of monitoring and dynamically adjusting the temperature and humidity levels to ensure the optimal preservation of sensitive products. The integration of security measures such as passcode protection against unauthorized access. In addition, with its real-time adaptable design, the system can respond to the external conditions across various geographic locations and seasons. This document details the conception, design, implementation, and evaluation of an IoT solution that promises to redefine product safety and cost efficiency in business operations.

## Motivation

Enhanced Product Safety: Many products, ranging from perishable foods to sensitive pharmaceuticals, require specific temperature and humidity conditions to maintain their quality and efficacy. An automated system can continuously monitor and adjust these conditions, significantly reducing the risk of product spoilage or damage due to environmental factors.

Security and Integrity: By incorporating security measures such as passcode protection, the system ensures that environmental controls cannot be tampered with, thus maintaining the integrity of the logistics chain.

Adaptability and Scalability: The ability to adjust parameters like humidity and temperature thresholds in real-time allows the system to handle a diverse range of products and adapt to varying external conditions, making it scalable across different geographic locations and seasons.

# System Design

## 2.1 Main Physical Components

**2 ULN2003 stepper motors & drive boards** controls the 4 phases of the stepper motor sequentially to roll up/down the window based on humidity. It requires a motor drive board, which needs to be pumped at 5-12 Volts. The external batteries package is used to offer the voltage. The rolling gate is controlled by 2 stepper motors. The gate should completely close when there is high humidity (heavy rain) and open by some degree when there is low humidity (small rain).

**1 DHT 11 sensor** periodically Track temperature and humidity using sensors that interface through the 1-Wire protocol. The sensor sometimes can fail due to different response time of from the ESP 32 board. Thus, specific waiting specific time can enhance the system stability.

**Heat lights** arereplaced by the LED light in this project for the safety reason. They Employs varying PWM duty cycles to deliver self-adaptive warmth and brightness.

**Out shell cubic** was planned to use the 3D printer to fulfill. However, the slot had been full during the presentation week. Thus, recycled material is finally chosen to make the shell cubic.

## 2.2 System Wiring Diagram

Picture 2-2-1 shows main components. There are 2 motors in the real design, each of them requires 6 wires (4 for motor drive bords + 2 for power). Each motor is driven by 4 AA batteries, to be provided 6 V power.

房间的摆设布局

中度可信度描述已自动生成

Picture 2-2-1 The wiring overview

## 2.3 APP UI/UX design

The APP has 3 editable text area and 4 buttons (Picture 2-3-1). User can enter the 4-bit password at the top of the screen. The code will be sent to the ESP32 board to check. If it is correct, the whole system will be unlocked. If it is correct, the following function will be unlocked.

图形用户界面, 应用程序

描述已自动生成

Picture 2-3-1 Android UI/UX

After entering the correct code, User has 2 buttons to turn on and turn off the system.

When User wants to change the humidity upper and lower boundary dynamically, it can input the intended maximum or minimum numerical value in the text box to adjust the initial position of the gate.

## 2.4 Total Cost Table

As shown in the Picture 2-4-1, The batteries takes the most portion of the cost. The totalt cost is 44.18 $.

图表, 条形图

描述已自动生成

Picture 2-4-1 Total expanse

# 3 System Implementation

## 3.1 DHT 11

The 1-wire code is set to let DHT11 communicate with ESP32 board. The **time\_t** structure is used to store timing information essential for decoding the 1-Wire communication signals, specifically capturing durations that distinguish between binary '0' and '1' values. Meanwhile, the **data\_t** structure is allocated for recording environmental data such as temperature and humidity obtained from the sensors. **sendStartSignal()** is for triggering the start of data transmission; **checkStartResponse()**, to verify the sensor's readiness to communicate; and **readSensorData(data\_t \*data)**, which reads and stores the sensor output into the provided **data\_t** structure.

## 3.2 ULN 2003 Stepper Motor

The motor is mainly used to control the gate position, which should be inversely proportional to humidity. It contains 2 steps to implement.

**GPIO Configuration**: Four GPIO pins connected to the stepper motor drive board are configured as outputs. These 4 pins directly control the stepping sequence, representing required to move the stepper motor.

手机屏幕的截图

中度可信度描述已自动生成文本

中度可信度描述已自动生成

Picture 3-2-1 Relation between humidity and height(left). Motor Control code(right)

**Stepping Function:** A function **OneStep(1)** dictates the sequence of steps to be executed based on the direction **(dir)**. This function (Picture 3-2-1(right)) cycles through a predefined sequence to energize the stepper motor in the correct order, achieving rotation.

## 3.3 Android APP

The **sendHTTPRequest** method encapsulates the logic to send a GET request to a specified URL and handle the responses.

**Step1**: Whenever turn on/ turn down is pressed, a HTTPGET request will be sent to the ESP32 board with the PASS code.

**Step2**: When the pass code is matched with the random passcode generated by the ESP32 board, the registered function **unlock\_uri()** will unlock the ESP 32 board.

**Step3:** If the PASS code is matched, the gate will be lifted up to its highest position, and waiting for the changes in the DHT 11 sensor.

**STEP4**: When entering a specific Maximal / Minimum humidity value in the bottom 2 text ares and the botton is pressed, a HTTPGET request will triger 2 registered functions **max\_uri** or **min\_uri()** to chage the initial position of the gate.

**Step5:** If the PASS code is matched and the “Turn off” button is pressed, the registered function **lock\_uri()** will be triggered by the HTTP request from the android devicw.

## 3.4 HTTP request

The sendHTTPRequest function in your Android application sends a GET request to a specified URL and handles the response. It establishes an HTTP connection, sets the request method to "GET", and checks the server's response code. If the response is HTTP OK (200), it reads and logs the response. If not, it logs an error indicating the failure. The function also includes error handling for exceptions during the request process, ensuring any issues are logged for debugging. The response is then returned as a string, capturing either the server's response or null if there was an error or non-OK response.

## 3.5 PWM LED

The **ledc\_init** function configures an LED control system on an ESP32 using the LED Controller (LEDC) peripheral. It sets up a PWM (Pulse Width Modulation) timer with a resolution of 13 bits and a frequency of 5000 Hz, specifying the timer mode and index. It then configures a channel for the LED, setting initial duty cycle, GPIO pin, speed mode, and timer association. This setup allows for smooth LED brightness control. Additionally, it initializes a fade functionality to enable gradual changes in LED brightness.

手机屏幕的截图

中度可信度描述已自动生成

Picture 3-5-1 Relationship between PWM and temperature

The **set\_led\_brightness\_based\_on\_humidity** function adjusts the LED brightness based on a given humidity value, mapping the humidity range of 26% to 59% to a PWM duty cycle range of 0 to 8191. It calculates the appropriate duty cycle based on the humidity input and updates the LEDC configuration to reflect the new brightness level, facilitating real-time adjustments to LED brightness as environmental humidity changes.

# 4 Experiment & Evaluation

## 4.1 Turn on / off using pass word

桌子上放了不同类型的电脑

描述已自动生成

Picture 4-1-1

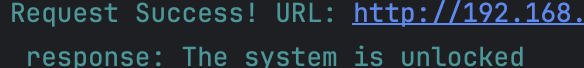
If the PASS word is wrong, the gate should not move at all and the Log message that Android platform receives “Wrong password”.

文本

描述已自动生成

Picture 4-1-2

If the PASS code is matched with the code that is randomly generated in the ESP32 board, The Android platform shoul receive “The system is unlocked” as



Picture 4-1-3

## 4.2 Move gate with humidity

To test the movement of the gate with humidity. The Movement of the gate is recorded by:

*ESP\_LOGI(TAG, “currentPosition = %d, stepsMoved = %d”, currentPosition, stepsMoved);*

In above code, current position will be incremented by 1 when the **Onestep()** is executed by once, which means the motor has rotate by 1 cycle.

To accurately control 2 motors rotating with the same angle, Picture 4-2-2 records the pre-calculated steps and moved steps, which can precisely rotate the motor to intended position.

桌子上放了不同类型的电脑

描述已自动生成桌子上放着笔记本电脑的人

中度可信度描述已自动生成

Picture 4-2-1

The Picture 4-2-1 (left) represents when the room is dry, the gate is put as its initial default position. The Picture 4-2-1 (right) represents when the wetness exceeds the threshold, the gate is moving down to prevent the rain from outside to affect the goods in the storage room.

As a result, the gate is rolled down by some degrees, which means the programming on interaction with DHT 11 sensor and motors is achieved.

文本

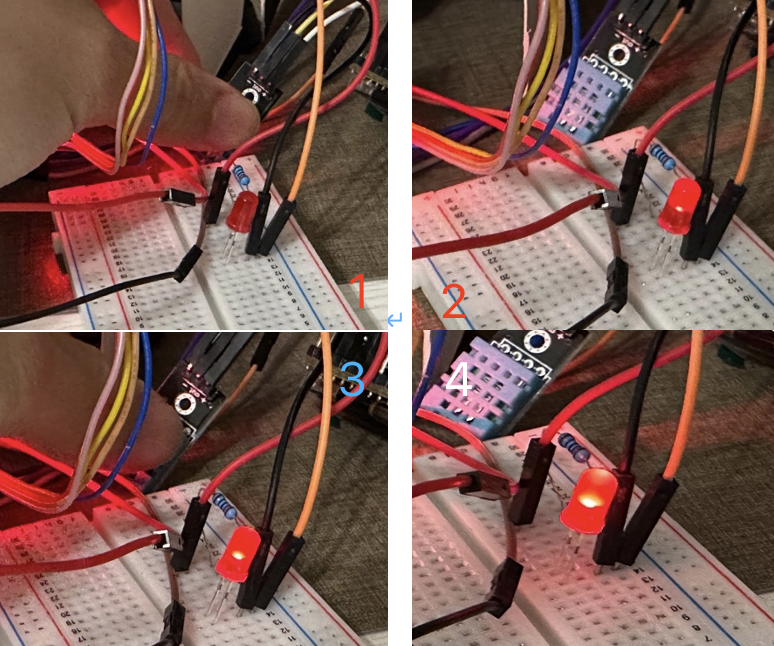
描述已自动生成

Picture 4-2-2

## 4.3 Changing PWM of LED

As discussed in the system integration, the LED is used to simulate the heat light to generate the heat and brightness when the temperature is low, through changing the PWM duty cycle of GPIO pin.

After entering the correct PASS code in the phone, the tester uses a hot hand to cover for a few seconds and leave. It is expected to see the LED light is almost off at beginning because the temperature is at its maximum offset. When the heat diminishes, the duty cycle of the LED is expected to increase because when the temperature goes low, the heat light should be able to heat the environment. Above experiment is captured by the following sequence of the photos (1->4). (Picture 4-3-1)



Picture 4-3-1

The brightness of the LED is gradually lighting up, which shows the PWM programming on LED is implemented successfully.

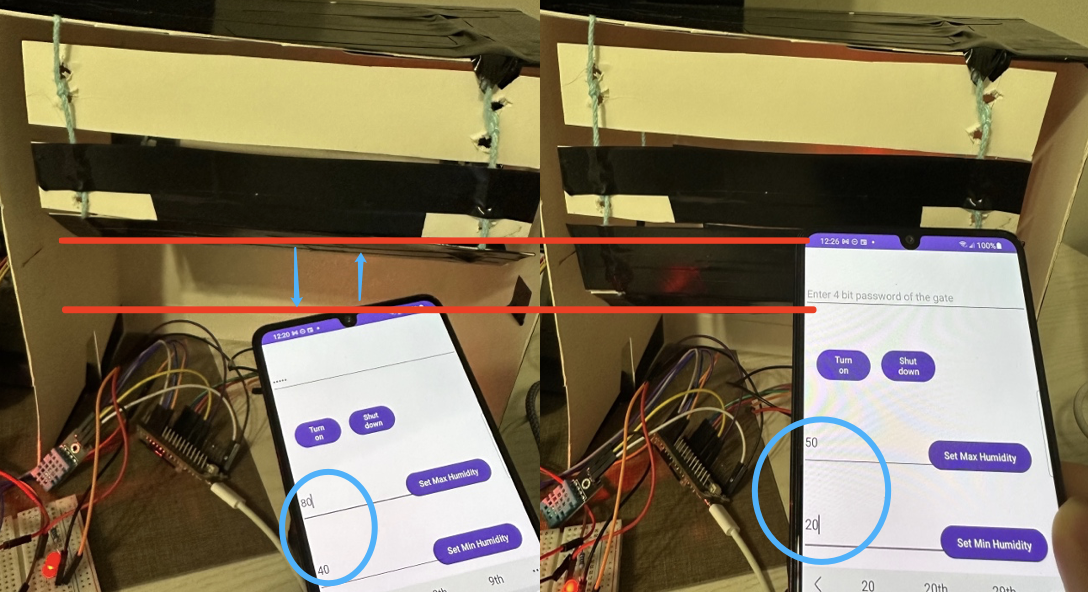
## 4.4 Change Min/Max humidity value

To test changing the humidity range, tester will input different Minimum and Maximum humidity pairs and observe if the initial state of the gate moves.

Step 1: Set the 80 as max humidity and 40 as min humidity. The initial position of the gate is moved to a specific level (Picture 4-4-1, lelft).

Step 2: Imagine we need lower down the humidity level to store some dry goods. Set the 50 as max humidity and 20 as min humidity. The initial position of the gate is moved to a specific level (Picture 4-4-1, right).

Step 3: By comparing the position level of the gate, the gate does move down due to the changing order sent by the tester.



Picture 4-4-1

# 5 Discussion and Conclusion

The main object of this humidity and temperature system has been fulfilled. There are some aspects can improve in the future. 1) Heat insulator should be added between the motor and the DHT 11 sensor to weaken the heat radiation. 2) The power supply should be replaced by uninterruptible power supply to enhance stability. 3) The color of the buttons of mobile app can be replaced by dynamic color to adaptively fit the current temperature. 4) The LED light can be replaced by real heat light.

This project let me learn a few things which benefits me: 1) Adding sufficient delay in 1-wire protocol can increase the fault tolerance. 2) HTTP GET request helps 2 devices connecting to the same WIFI talk to each other. 3) The stepper motor should be set to all 0 in all 4 GPIOs. Otherwise, 1 wire coil will be always turn on, generating unnecessary heat and causing safety issue.