ECE 445

SENIOR DESIGN LABORATORY

FINAL REPORT

Final Report: Fingerprint Recognition Door Lock

Team Members

CHENGRUI WU
(cw70@illinois.edu)
HANGGANG ZHU
(hz66@illinois.edu)
HAORAN YUAN
(haorany7@illinois.edu)
LIZHUANG ZHENG
(lzheng17@illinois.edu)

TA: Guozheng He

Friday 10th May, 2024

Contents

1 Introduction

1.1 Problem and Solution Overview

In our Residential College, each student has own dormitory, and the dormitory door can only be opened by the student's own IC card from outside. Sometimes it is possible that you forget your IC card inside the dormitory room, or lost it somewhere by mistake, so that you must go to the front desk of the Residential College to get a temporary card, or go to the IC card service center to get a new card. And if it is in the midnight, it will be harder to get staff in touch. So it is better that students can use more methods to open the door except for swiping the IC card. We are thinking of other ways to unlock the door using other personal identification information. Even if you didn't lose your IC card, with more ways to open the door brings a little more joy in daily life.

Some popular way to unlock the door is password, facial recognition and fingerprint recognition. Considering the difficulty and portability, we decide to develop our own fingerprint recognition lock for our Residential College. However, replacing all the door lock in the Residential College is quite challenging. we propose a device which can be easily attached to the existing door lock, and turn it into a fingerprint recognition door lock, without assistance from the professional installation workers.

In addition to fingerprint recognition, we also intend to integrate other approaches to our smart door lock. Some basic functionalities include unlocking the door using Software App with the help of remote control through Wi-Fi. Besides, we will also apply Bluetooth technology to open the door lock automatically when the bonded mobile phone is approaching. In order to save energy, the device will turn into low energy mode, and we will add an infrared detection part to our device, which will wake up the device when people come back. Furthermore, we will try to implement more advanced features including unlocking the door through facial recognition, and voice recognition, which can make our device more convenient and intelligent. In general, we intend to develop a portable device with integrated ways to unlock doors, which can be managed easily through our mobile phone application and promise the security.

1.2 Visual Aid

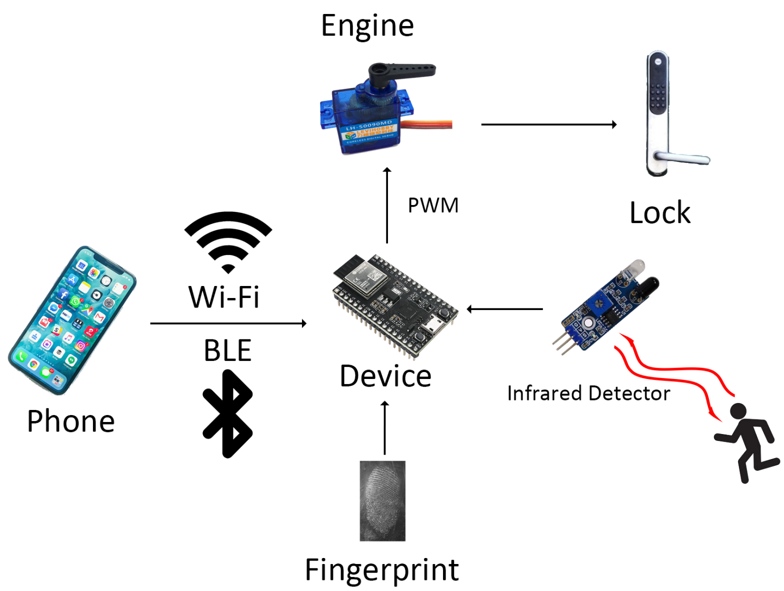


Figure 1: Visual Aid

1.3 High-level Requirements List

- Enable the authorized users to open the door lock using their fingerprints, the controller should be able to store at least 5 different fingerprints and the success rate should be above 80%. Besides, the infrared detector can wake up the device when people stand in front of the door within 1m $\pm$ 0.2m.
- Allow remote control using the software app with delay time of at most 5 seconds. And the BLE module of device can identify the neighboring mobile phones when approaching inside the range of 0.5m $\pm$ 0.1m.
- The mechanical subsystem can reliably open the door, the servo motor with a torque of at least 35 kg$\cdot$cm is necessary.

2 Design

2.1 Design Procedure

2.1.1 Block Diagram

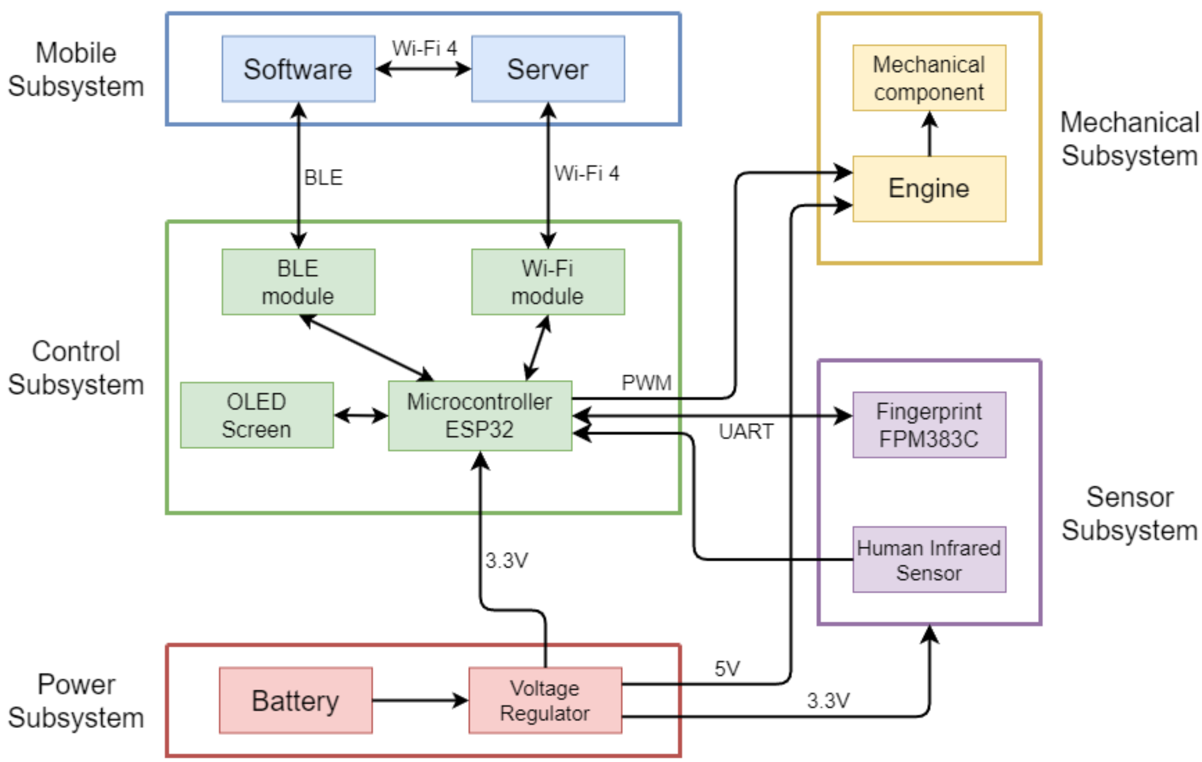


Figure 2: Block Diagram

Our device is divided into 5 subsystems: The Power Subsystem, the Sensor Subsystem, the Control Subsystem, the Mechanical Subsystem, and the Mobile Subsystem. The Power Subsystem contains two sets of 7.4V lithium batteries as the power supply. The output voltages from the batteries will be regulated by the voltage regulator circuit in order to output a 3.3V voltage for the Control Subsystem and the Sensor Subsystem, and a 5V voltage for the servo motor in the Mechanical Subsystem. The Sensor Subsystem consists of a fingerprint recognition module FPM383C, and a set of infrared module including an infrared emitter and a receiver. The fingerprint recognition module compares the fingerprint received with the user fingerprint data, and communicates with the microcontroller through the UART protocol. The infrared module is set up with the concern of energy saving. When human passed by, the module wakes up the Control Subsystem and then make it to start working. The Control Subsystem uses ESP32 as the microcontroller, which accepts the signals from the Sensor Subsystem and delivers a PWM signal to control the Mechanical Subsystem. In the Control Subsystem, ESP32 is also integrated with a Bluetooth at Low Energy (BLE) module and a Wi-Fi module, which allows the remote control from the software app in the Mobile Subsystem. The Mobile Subsystem contains a software app which communicates with the Wi-Fi module through the Wi-Fi 4 protocol, so that the user can control the door lock remotely. A server is also needed between the Wi-Fi module and the software. The cellphone with the Bluetooth function can also be seen as a part of the Mobile Subsystem, it uses BLE protocol to communicate with the BLE module, so the device can recognize the approaching mobile phones and open the door automatically. The Mechanical Subsystem contains a mechanical engine and some other mechanical components. The mechanical engine is basically a servo motor, which is controlled by the PWM signal from ESP32. For other components, a nylon thread connecting the motor with the door handle is used to pull the door handle down; and some brackets holding all the components are needed for attaching them on the door. So once the user approaches the door, the infrared module will detect the user and wake up the whole device, and then the device will wait for a signal to open a door, either from the fingerprint matching, or the remote control (Bluetooth or software app). Then the micro-controller will turn the command of opening the door into PWM signals, so that the servo motor can rotate to a particular angle, pulling the door handle with the attached nylon thread, and thus open the door.

2.1.2 Control Subsystem

This subsystem consists of a micro-controller that manages the operations of the device, including managing peripheral units, mechanical subsystem, and coordinating with the mobile app and BLE signal designed to interact with mobile phone. Besides, it will be waked up by human infrared signal, which enables the device to sleep when people have left and to work when someone is approaching. Furthermore, we have also provided a OLED screen and several buttons for the user to get the information of the device and control the device directly.

2.1.3 Sensor Subsystem

This component comprises a human infrared detection module and a fingerprint recognition module. The infrared module signals the ESP32 board to activate the Bluetooth and WiFi modules when someone enters a predetermined range, allowing the system to conserve power by operating in a low-power state when unoccupied. The fingerprint module consolidates fingerprint scanning, storage, and identification into a single, efficient unit.

2.1.4 Mobile Subsystem

Mobile Application End A mobile software application called *LockCompanion* comprising both front-end and back-end components facilitates communication with our door lock either through Bluetooth or via a server. The outline for the design process of this subsystem is delineated below.

1. **Platform Consideration** Numerous software development platforms are viable for controlling the door lock via a mobile device. Among these options are building a WeChat mini-program, an iOS application, or an Android application. We have opted for Android (version 12+) due to the cumbersome software auditing requirements associated with the other choices. Moreover, Android offers comprehensive documentation on Bluetooth connection manipulation, thereby streamlining the development process.

2. **Frone-end Design** While our focus is not on an elaborate front-end interface, it is imperative to include a basic interface for user interaction and feedback provision. Our mobile application necessitates interaction with various components: Bluetooth, ESP32 controller, and server. To ensure functional decoupling, we employed a Navigation UI featuring three Android Fragments, each displaying relevant content pertaining to the aforementioned components. It should be noted that the minimum requirement for the front-end is to incorporate several buttons capable of controlling the door lock.

3. **Back-end Design:** The primary tasks of the back-end entail communication with the ESP32 using the Bluetooth protocol and interaction with servers to facilitate remote control. Following the standard Bluetooth development procedure, our back-end encompasses activities such as opening Bluetooth, scanning for Bluetooth devices, connecting to BLE devices, data transfer, disconnecting from BLE devices, and retrieving BLE RSSI strength. For server communication, OKHttp[1], a popular and efficient HTTP client, has been selected for its efficiency and user-friendliness. Alternative options include Volley[2], Retrofit[3], and HttpURLConnection[4].

Server Application End The server app, hosted on a cloud platform like Azure, facilitates secure communication between mobile devices and door locks. Developed with Flask in Python, it starts by enabling direct interaction between a single mobile and door lock, laying the groundwork for more complex functionalities. It scales up to manage multiple devices and locks through a sophisticated database that supports user registration and identity verification, ensuring personalized access.

Containerized with Docker for streamlined deployment, the app emphasizes robust security measures for data protection and secure connections, catering to the evolving needs of a connected ecosystem. This app provides a secure, scalable, and user-friendly solution for mobile-device-to-door-lock interactions.

2.1.5 Mechanical Subsystem

The Mechanical Subsystem consists of some brackets, a servo motor and an mechanical actuator which can pull the handle of the door lock from the inner side and thus open the door. The design sketch is shown in Figure 6. This subsystem is installed near the lock, inside the door. The servo motor is directly wired with the Controller Subsystem, and accepts PWM signal from the micro-controller as a trigger. Then the servo motor drives the actuator, and the actuator can push or pull the door handle inside, to complete the action of opening the door.

2.1.6 Power Subsystem

The Power Subsystem contains a battery set and some voltage regulators. It is used to power up our Controller Subsystem, Mechanical Subsystem and Sensor Subsystem. We plan to use a 12V lithium battery as the power source at first. And in addition, a set of AA batteries can be used as the backup power source. The voltage regulators will regulate the voltage to 3.3V to power up the micro-controller, and to 7V for the servo motor.

However, after we did some experiments and testing, we found that regulating 12V down to 3.3V will result in heavy power consumption, and LM317 linear regulator has too large drop-out voltage.[5] Plus, as we have chosen a 35kg·cm servo motor, 7V is not necessary for the motor to get enough torque for unlocking. So we decided to use 5V voltage output to drive the motor, which is highly commercialized and standardized. So in order to output stable 5V and 3.3V, we choose the power module with two 18650 lithium batteries as the power source, so that the input voltage will be 7.4V, with the low drop-out power regulator circuit (such as AMS1117 with drop-out of 1V[6]), the battery set can output the voltages we want.

Since our servo motor is an inductive load, the motor rotation may affect the stability of the 3.3V output. So we used two sets of batteries both with regulated 5V output, and then there is another power regulator accepting 5V and output 3.3V for the controller subsystem altogether on the PCB.

2.2 Design Details

2.2.1 Control Subsystem

We will choose an ESP32-WROOM-32 which involves Wi-Fi module and BLE module as the micro-control unit. In order to provide a direct way to control the device, we apply a OLED screen, as well as up, down and select buttons for the users to select and execute commands shown in the OLED screen, including Fingerprint_Enroll, Empty_Fingerprint, Get_Finger_Number and Empty_Bond_Device function. For the sensor subsystem, The Fingerprint Recognition Task in ESP32 can be controlled through different command, which enables the user to enroll or delete the fingerprint records in FPM383C, or read the number of finger records in the FPM383C. we design a FSM (Finite State Machine) for Fingerprint Recognition Task and treat it as the main loop task in ESP32. The design of the Task is shown in Figure 3. The Human Infrared Detection Module will identify if there is a human stand within 1m $\pm$ 0.2m. Then the Module will notify the RTC_GPIO of ESP32 to wake-up the microcontroller from deep sleep mode; when ESP32 is working, the Human Infrared Wake-up task will continuously check the RTC_GPIO to see if there is anyone stand in front of the door and control the time_to_sleep variable accordingly.

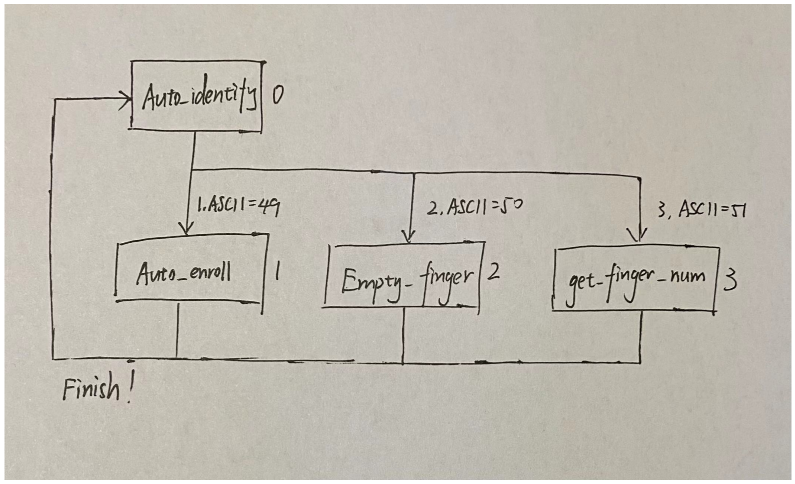


Figure 3: Fingerprint Recognition Task

For the Mobile Subsystem, we design a Wi-Fi object and connect it to the mobile Wi-Fi or the router Wi-Fi. The Wi-Fi task enables the ESP32 to send continuously Request packet to the server for every 2 seconds and receive the Reply packet from the server. Then the ESP32 will deal with the Reply packet and decide whether to open the door. The BLE Task in ESP32 can establish a connection with the software app. Through the connection, the user can send commands to ESP32, including Fingerprint_Enroll, Empty_Fingerprint, Get_Finger_Number, Door_Open function. Besides, the BLE module will be able to Identify the strength of BLE signal when devices approach 0.5m $\pm$ 0.1m. we design a BLE object and use it to establish a connection between the software and the ESP32, then transmit message through this connection. it will also evaluate the distance of connected device by reading the Received Signal Strength Indication (RSSI) of BLE signal and decide whether to open the door. Furthermore, we also apply BLE devices Authentication component to BLE object to ensure the security. Specifically, The BLE object of ESP32 will ask for a secure key exchange at Man-In-The-Middle (MITM) level protection with the Mobile end. So, It requires the user to enter passkey to bond to the ESP32 when the Mobile end connects to the ESP32 at the first time. Considering that the simultaneously working of BLE Task and Wi-Fi Task will lead into Invalid use of antennas, we set a Wi-Fi-BLE change Task and a button at the outside of the door to handle this. Besides, the Wi-Fi mode and BLE mode will display to the user through the LED color of the FPM383C fingerprint recognition sensor as red and blue, respectively. For the Mechanical Subsystem, ESP32 will use PWM signal to control the behavior of the servo motor. The ESP32 microcontroller will be powered at +3.3V regulated by the power supply subsystem. The schematics and PCB design of the Control Subsystem can be found in Appendix C.

2.2.2 Sensor Subsystem

This section encompasses both an infrared detection module and a fingerprint recognition module. The human infrared detection module is designed to emit a signal to the ESP32 board, which activates the Bluetooth and WiFi modules upon detecting a person within a predefined proximity. This functionality ensures the system conserves energy by operating in a low-power state in the absence of individuals. We will use HC-SR505, which is a human infrared sensor at the outside of the door to detect if there is someone approaching. We choose HC-SR505 component because this sensors' working current is around 10 μA, which is much less than the ESP32 when it is working.

We utilize the FPM383C ideal fingerprint sensor as our chosen fingerprint recognition module. This module stands out due to its compact size and low power consumption, making it an ideal match for our battery-operated door lock system. It operates on a 3.3V power supply and communicates with our ESP32 micro-controller via the UART serial communication protocol. Designed with an independent built-in chip, the module can store up to 60 fingerprint records in its flash memory, compare and identify the fingerprints rapidly. Besides, we will use AFC01-S08FCA-00 connector and 8p FPC cable to connect the peripheral Sensor subsystem and the PCB board.

2.2.3 Mobile Subsystem

Mobile Application End Our mobile software *LockCompanion* has the following functionalities. Its related technical details are also provided.

1. **Unlocking door with bluetooth communication**. The most fundamental function of *LockCompanion*. The technical details roughly follow Google's Bluetooth documentation [7]. Briefly, BLE device is discovered through *BluetoothLeScanner*. A detailed Android *BluetoothLeService* is used to call device *connectGatt*. On connection callback, the predefined communication *gattCharacteristic* is found to read and write data, which is implemented using *writeCharacteristic()* and *readCharacteristic()*. User interaction functionalities (e.g. unlock button, show Toast) is achieved in *Unlocking Fragment*. *Unlocking Fragment* and *BluetoothLeService* communicates through a *Broadcast Receiver*. Related data (e.g. connection state) is stored in a *ViewModel* as data in fragment will be lost once app is navigated to another fragment. Instructions sent to ESP32 include **unlocking**, **enrolling finger**, **emptying finger**, **getting number of enrolled finger**. Communication protocol is described in figure 4.

2. **Unlocking door remotely through Server**. A user can unlock the door with *LockCompanion* even if he's far away from the door. *LockCompanion* simply send requests to Server End (described below) with user id, password, lock ID etc, and server will interact with ESP32 with specified lock ID.

3. **Auto unlocking if phone is close to the door**. This is achieved by repeatedly reading RSSI (Received Signal Strength Indicator) strengh in a thread every 500ms and once rssi meets certain criteria, a unlocking instruction is sent to ESP32. while RSSI is related to many factors including radio spectrum, transmitting power, path loss, it can be roughly modeling as the following formula[8] to indicate its relationship with distance.

$$[P_r(d)]_{dBm} = [P_r(d_0)]_{dBm} - 10nlg(\frac{d}{d_0}) + X \quad (1)$$

   where, $P_r(d)_{dBm}$ is signal receiving strengh at distance d, d_0 is reference distance, n is attenuation factor for RSSI and X is a normally distributed variable. The closer the phone is to the door lock, the higher RSSI is and our empirical experiments show that $-45dBm$ is a good threshold for automatically unlocking.

4. **Memorizing device**. MAC address of lastly connected ESP32 is persistently stored using Android *Shared Preferences* to avoid tedious scanning before each connection.

Server Application End Secure Server Application is designed to facilitate controlled interaction between multiple users and devices within a secure network. The application utilizes a token-based authentication system to ensure that all user registrations and device interactions are verified and secure. A detailed flowchart describing the behavior of the server is given in Figure 5.

1. **System Overview**
   The server application leverages a security model based on tokens to manage user registrations and interactions with devices. This model ensures that every request is authenticated, thus maintaining a high level of security within the network.

2. **Authentication and Access Control**
   - **Token-Based Registration**
     – During the registration of a user or a user's access to specific devices, a randomly generated token is used to verify that the registration is initiated by an authorized user. This token is crucial for confirming the identity and legitimacy of the registration requests.
   - **Passport Setting**
     – Separate from registration, the application uses the token to set a "passport" for each user, which is a key-value pair linking the user with a specific device.
     – When a user wishes to specify their passport, the token must be provided to authenticate the user's identity before the passport is assigned.

3. **Device Interaction and Request Management**
   - **Accessing Devices**
     – To access a device, such as opening a door, a user must send a request that includes their username, the ID of the device they wish to access, and their passport.
     – At this stage, the token is not required again; the passport alone is sufficient to verify the user's identity, streamlining the process and enhancing user convenience.
   - **Security and Data Integrity**

- All interactions, particularly those involving token transmission and passport setup, are securely encrypted to prevent unauthorized access and ensure the integrity of data.
- The system implements rate limiting and other security measures to protect against potential security threats such as denial of service attacks.

Algorithm 1 Server Application Pseudo-code

```
Define 'histogram' as a dictionary with door IDs as keys and access count as values
procedure USER AUTHENTICATION
    Define a route '/user' to handle GET requests
    function USER
        Get 'ID', 'name', and 'password' from request arguments
        if ID or name is invalid then
            return error message 'Invalid ID or name'
        end if
        if password mismatch then
            return error message 'Incorrect password'
        end if
        Increment access count in 'histogram'
        return success message 'Access granted'
    end function
end procedure
procedure DEVICE ACCESS
    Define a route '/esp32' to handle GET requests
    function ESP32
        Get 'ID' from request arguments
        if ID not recognized or no pending requests then
            return error message 'Invalid request or device ID'
        end if
        Decrement request count in 'histogram'
        return success message 'Device opened successfully'
    end function
end procedure
Initialize and start server and counter thread
```

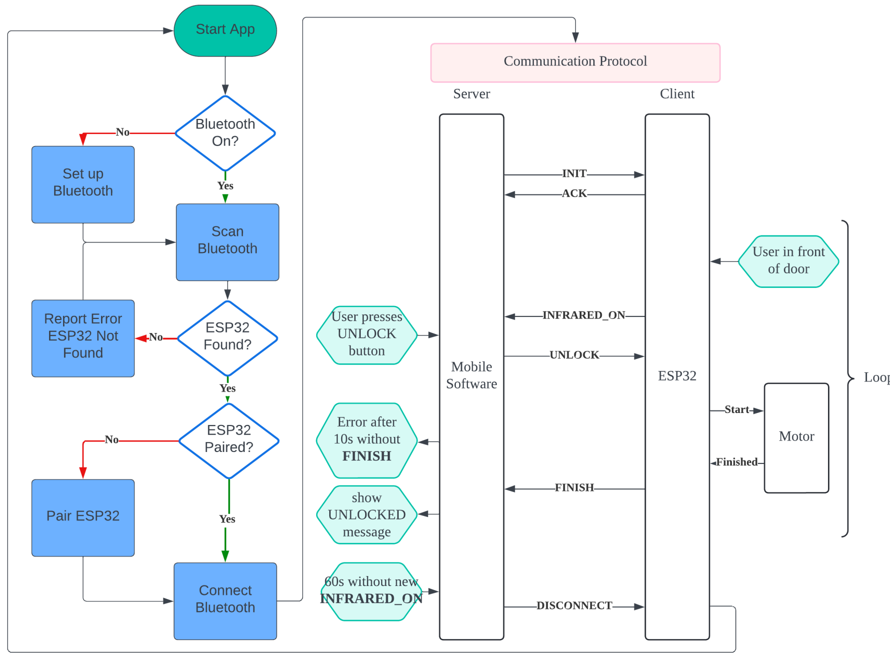


Figure 4: Mobile Software Application and its Communication with Control System

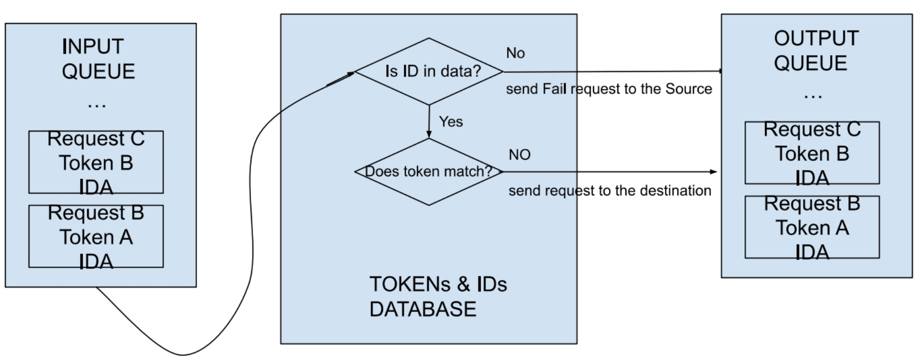


Figure 5: Server Application Flowchart

2.2.4 Mechanical Subsystem

The servo motor are used to drive the actuator when it gets the signal to open the door. It is wired with the micro-controller, using PWM protocol to contact with it. So it can get a PWM signal from the Controller Subsystem as a trigger. When the PWM signal requests to open the door, the servo motor will turn a particular angle and drive the actuator to move to some extent. By estimation, our door handle needs around 25N to open, consider the arm of force in about 8∼9cm, we need a servo motor with its torque above 25kg·cm. TD-8125MG digital servo motor can be a candidate. However, due to the limitation of the installation place, voltage supply, force application, we had better to use a servo motor with larger torque, which can provide larger amount of redundancy.

The actuator is directly contacted with the door handle. There are many possible designs for the actuator. One possibility is a hammer hanging above the handle inside, when the servo motor moves, the hammer can drop down and push the door handle to the appropriate angle so that we can open the door from the outside. Another strategy is to use a nylon thread to pull the handle down from the inside, thus the servo motor should be installed at the bottom of the lock. We decided to choose the second strategy, since it is easier to design and implement. Since the nylon thread may slip on the handle, a rubber sleeve should be used on the door handle to limit the position of the thread.

After lots of testing and experiments, we found that the 25kg·cm servo motor cannot qualify the job of unlocking the door reliably. That is because the force needed is almost at the top of its output capacity, and when the angle turns to a state where the nylon thread goes through the rotation axis of the servo motor, the force arm will approach zero, and the force needed becomes very large. Sometimes the rotation of the servo motor is even reversed for protecting itself from damage. Finally, we decided to shift to the 35kg·cm servo motor TD-8135MG, which has more redundancy amount to complete the job reliably.

For the physical design as shown in Figure 6 below, it consists of 3 brackets (boxes), one for the servo motor, one for the micro-controller (ESP32), batteries and the voltage regulator, and the rest one for the sensors. They are all sticking to the door by some strong adhesive tapes. Wires go out of each box through some holes and connect each part together. All these brackets are modeled and produced by 3D printing, using PLA plastics as materials.

The bracket for the servo motor is in an octagonal shape, with the consideration of supporting the servo motor and limiting its movement. We have also used four self-tapping screw to fasten the servo motor on the bracket, so that it will not be easily twisted when pulling the handle down. As for installation, we recommend to adhere it below the door handle, then ensure the motor arm pointing up-left in 45°, and keep the nylon thread vertical.

The bracket for the batteries and PCBs is hanging above the door lock. There are two separate cavities for the two battery sets, with mounting holes as an optional fastening measure. The holes opened on the top of the brackets are for the installation of the PCB boards, which realizes the functionalities of the micro-controller and the voltage regulator. There are also many square holes on the side of this bracket, they are not only for the wires, but also for the heat dissipation.

The bracket for the sensors has a trapezoid shape, as shown in Figure 6 and in the Appendix B. On the top side there is a circular slot for embedding the fingerprint recognition module FPM383C. And below it is a cavity for the wiring board of the fingerprint module, together with the infrared modules. The wire can go through the holes on the side, cross the door from the crevice between the door plank and the door frame, and connect to the micro-controller inside the door.

The detailed drawings for the above brackets can be found in the Appendix B at the end of the report.

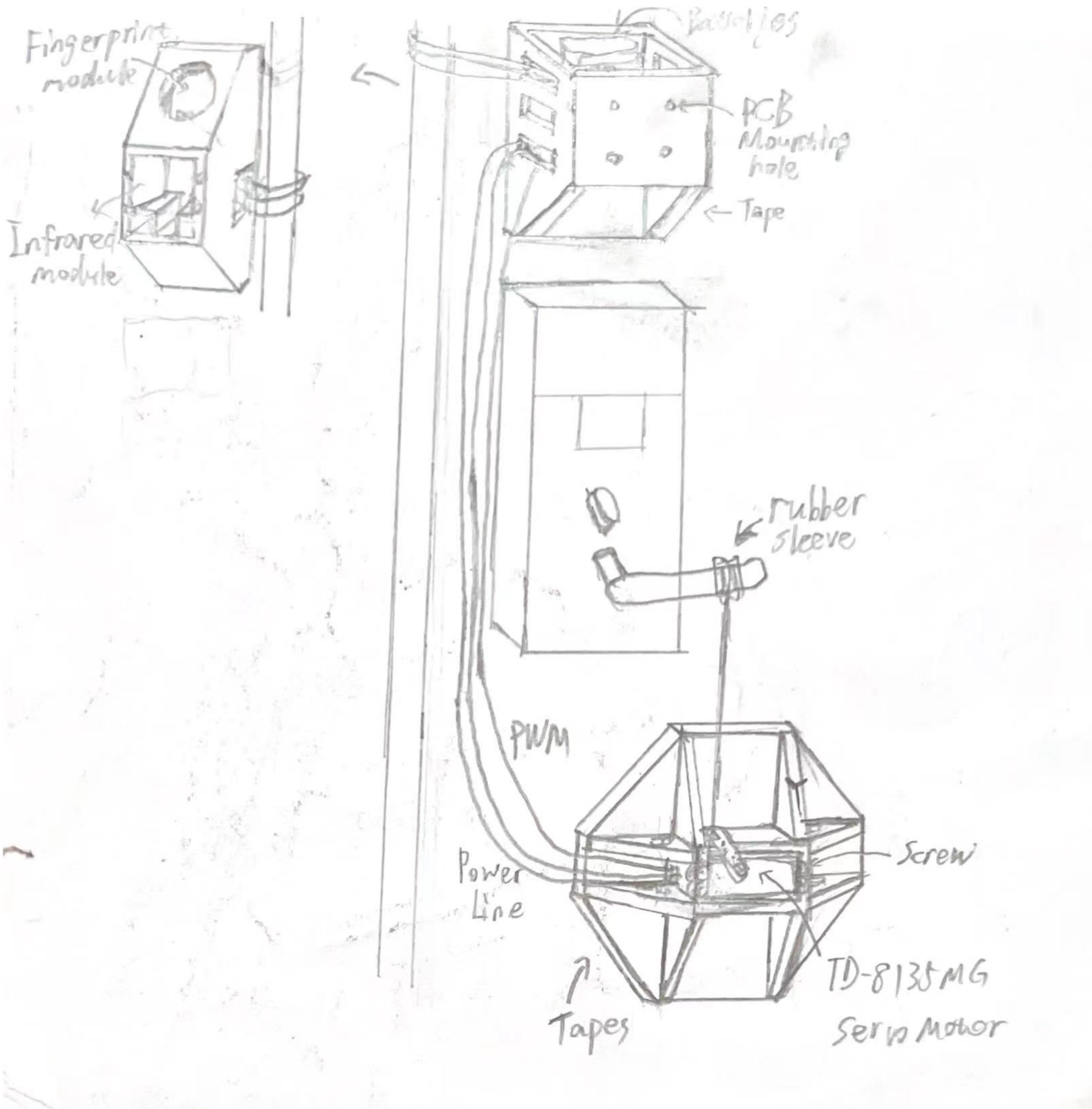


Figure 6: Physical Design

2.2.5 Power Subsystem

The Power Subsystem is the crucial part for powering up other subsystems, it should provide stable power to support the normal work of the entire device. The power can be provided by a 12V battery, which can be easily recharged or replaced when it dies out. Then for the voltage regulator circuit, we can use some adjustable voltage regulators to produce different stable voltage output, like LM317 3-terminal adjustable voltage regulator[5]. The micro-controller and the fingerprint recognition subsystems need to work under 3.3V, but the 35kg·cm servo motor should work in about 4.8∼7.2V, and the ones with larger torque may need higher voltage. The voltage regulators should provide stable voltage for them and ensure that they can work normally.

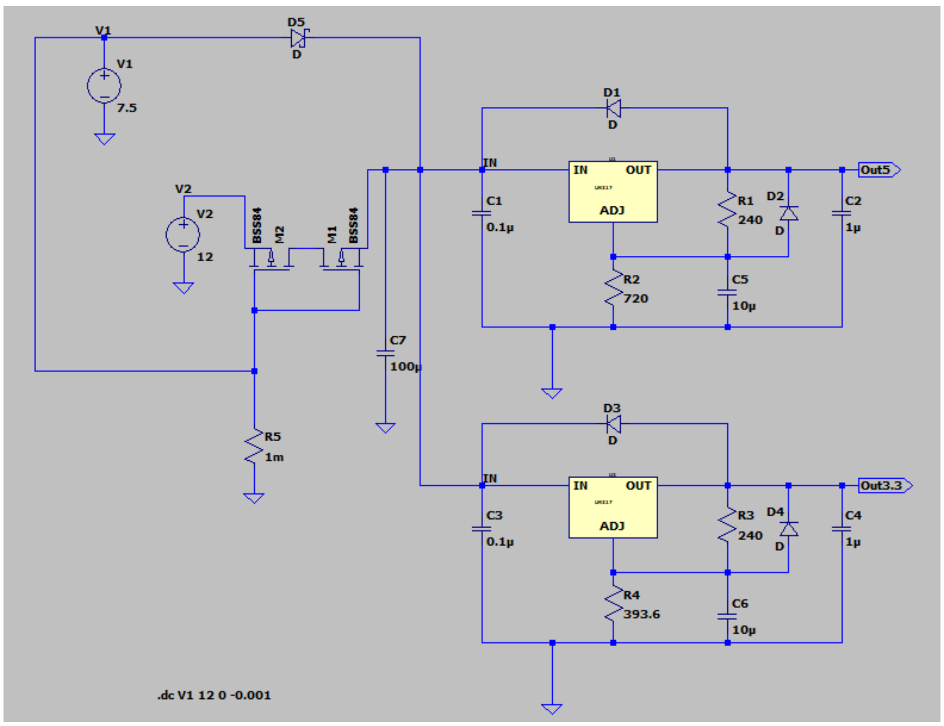


Figure 7: Voltage Source Switching Circuit and Voltage Regulator Circuit

Figure 7 shows the basic circuit design for the voltage regulators and the voltage source switching circuit. In this schematics, the left part is the voltage source switching circuit, V1 is the main lithium battery, V2 is the backup AA battery set. D5 is a Schottky diode, which has a lower forward voltage drop (about 0.15∼0.45 V) and a faster switching action, very suitable for switching the voltage source. M1 and M2 are two BSS84 PMOS transistors, they are conducted when the gates (G) are in low voltage levels. M1 is used to avoid V1 from recharging V2 (equivalent to a Schottky diode), M2 is used as the switch for V2. When the main source V1 dies out, its voltage will decrease, and then the voltage level of the gate (G) will be lowered, for M2, $V_{GS} < 0$ and then the PMOS transistors will conduct, so that the backup battery set V2 will take over and continue to supply power. Besides, $R_5 = 1M\Omega$ is to avoid short circuit of V1, C7 is for filtering.

For the right side of the circuit, we use the typical 3-terminal linear voltage regulator circuit built by LM317 as the main part of our voltage regulator. The circuit can be found in page 11 of [5]. The output voltage V_{Out} of LM317 can be calculated as shown below [5]:

$$V_{Out} = V_{Ref}(1 + \frac{R_2}{R_1}) + (I_{Adj} \times R_2) \tag{2}$$

I_{Adj} is typically 50 μA and negligible in most applications.[5] Since there is a 1.25V offset input in the adjust terminal, we have $V_{Ref} = 1.25V$[5], so the output voltage is approximately

$$V_{Out} = 1.25V \times (1 + \frac{R_2}{R_1}) \tag{3}$$

Here we want the output be $V_{Out} = 5V$ and $3.3V$, so we use two voltage regulator circuits with the pin out "Out5" and "Out3.3." The data sheet [5] recommends us to use $R_1 = R_3 = 240\Omega$, so according to Equation 3, we can solve that $R_2 = 720\Omega$ and $R_4 = 393.6\Omega \approx 400\Omega$. What is more, if we want a output of 7.2V, we need $R_2 = 1142.4\Omega$. C1 is the filter capacitor, C2 is used to improve the transient response, C5=C_{Adj} is used to improve the ripple rejection.[5] There are also two diodes in each voltage regulator circuit for protection. They both provide a low-impedance path for the capacitors to discharge, so that they can prevent them from discharging into the output port of the regulator. D2 is for C5, and D1 is for C2.[5]

However, another concern raised is that LM317 may consume too much power, since its voltage drop between the input and the output is too large, about 3V.[5] That is also the reason for us to use a 12V battery if we want a 7V output for the servo motor. So seeking for a design with lower power consumption is necessary. For example, AS1117 is a series of linear 1A low dropout voltage regulator, and its dropout voltage can be as low as 1V (maximum 1.3V).[6] We can use its 3.3V version AS1117-3.3 for the 3.3V output. We can just use its 5V version AMS1117-5.0 to output 5V for driving the servo motor TD-8135MG. Also, the design of having a backup battery is meaningless since the lithium battery supports charging and discharging. So we took off the backup part.

Finally, we decided to use the 7.4V 18650 battery module produced by the Nologo Technology. It uses two 18650 lithium batteries as the power source, then it uses an LN4393[9] IC to get regulated 5V output. Also it uses an AMS1117-3.3[6] for 3.3V output. But we have also made another power 5V to 3.3V power regulator, integrated on the PCB board, so we only use its 5V output for both the PCB and the servo motor.

3 Verification

Here are the verification work we have done for ensuring that each part can work normally. The detailed Requirements & Verification Table can be found in Appendix A.

3.1 Tolerance Analysis

The common reason for failing to open the door might be that the torque provided by the servo motor was insufficient to overcome the limiting friction. To ensure the selection of a motor with adequate torque to smoothly rotate the door handle, we utilized an electronic force gauge from the lab to measure the forces involved in rotating the handle. We operated the force gauge slowly to mitigate any experimental errors caused by acceleration. As shown in the figure below, the data indicates that the peak force is approximately 24 N and the force stabilizes at 16 N, indicating that the maximum force we need is around 24 N. Considering that the distance from the attachment point to the axis of rotation is 8.5 cm, and using the formula

$$\tau = \vec{r} \times \vec{F} \tag{4}$$

we calculated the required torque to be

$$\tau = \frac{24\text{N} \cdot 8.5\text{cm}}{9.8\text{N/kg}} = 20.8\text{kg} \cdot \text{cm} < 25\text{kg} \cdot \text{cm}. \tag{5}$$

Consequently, we selected the TD-8125MG digital servo motor with a nominal torque of $25\text{kg} \cdot \text{cm}$ to fulfill our requirements.

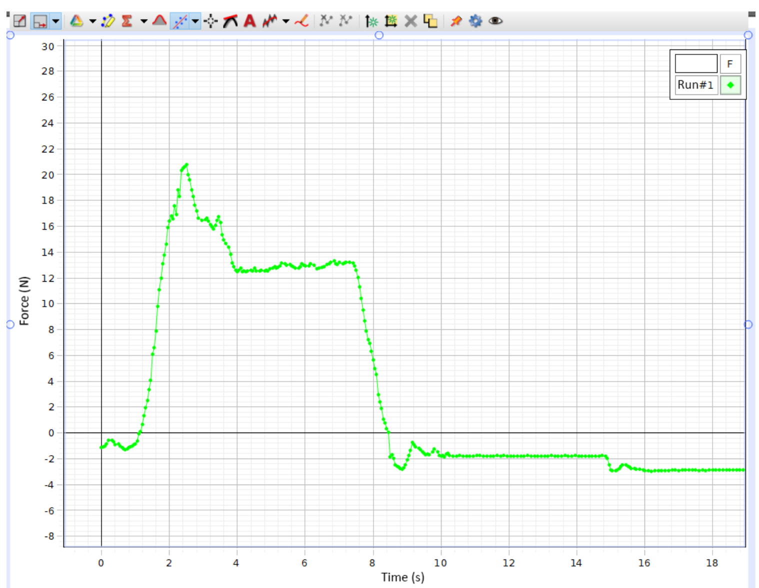


Figure 8: Force-Time Plot for Opening the Door

However, after testing and experiments, we found that the servo motor with 25kg·cm cannot open the door reliably. That is because the force the servo motor output is not always along the thread, the essential force is only a component of the total force given by the servo motor. And due to the angle of the thread is not always predictable, we can only estimate that when the rotation angle of the door handle is $\theta = 45°$, the force needed along the thread is 25N, assuming the thread is perpendicular to the door handle (but it is also not realistic), and then the force needed is approximately

$$F \geq \frac{25\text{N}}{\sin 45°} = 25\sqrt{2}\text{N} \approx 35.36\text{N} = \frac{35.36\text{N}}{9.8\text{N/kg}} \approx 3.61\text{kg} \tag{6}$$

With the force arm $L = 8.5\text{cm}$, we have

$$\tau = FL \geq 3.61\text{kg} \cdot 8.5\text{cm} \approx 30.685\text{kg} \cdot \text{cm} \tag{7}$$

So it is necessary to choose a 35kg·cm servo motor, for providing more redundancy and allow reliable unlock.

3.2 Simulation

Figure 9: Simulation: V1 Sweep from 12V to 0

The simulation in LTSpice shows that, when we sweep the voltage of the main battery V1 from 12V down to 0, the input voltage V_{in} first decreases together with V1. Then at about 9.5V, the circuit successfully switched to the backup battery set, so that V_{in} returns to 12V. In this process, the 5V and 3.3V output are both stable. In fact, as long as the output needed is smaller than about 9V, this circuit will output two stable regulated voltages.

4 Cost

4.1 Labor

We assume that each of us deserves ￥200 per hour of work, and each of us works 10 hour per week. The project takes about a semester (12 weeks) to complete, so the reasonable salary for each of us is:

$$\frac{\text{￥}200}{\text{hour}} \cdot \frac{10\text{ hour}}{\text{week}} \cdot 12\text{ weeks} \cdot 2.5 = \text{￥}60,000 \tag{8}$$

So for all 4 of us, our labor cost is about

$$4\text{ persons} \cdot \text{￥}60,000\text{ /person} = \text{￥}240,000 \tag{9}$$

4.2 Parts

Part #	Mft.	Description	For	Price	Qty.	Total
TD-8135MG	Tiankongrc	35kg·cm digital servo motor	Mech.	￥70	1	￥70
FPM383C	Hi-Link	Fingerprint sensor	Sensor	￥22	1	￥22
HC-SR505	Risym	Mini PIR Motion Sensor	Sensor	￥5.14	1	￥5.14
18650 Battery Board	Nologo	7.4V lithium battery board	Power	￥40.55	2	￥81.1
OLED	Youxin	0.96-Inch OLED Module (7Pin)	Ctrl.	￥9	1	￥9
AWM 20624 80C 60V VW-1	Xinhongnuo	5mm 8Pin FFC/FPC Soft Cable (30cm)	Ctrl.	￥7.85	1	￥7.85.
PCB Components(1 Board)						
ESP32-WROOM-32-N4	Espressif	Micro-controller	Ctrl.	￥20.6100	1	￥20.6100
CH340X	WCH	Serial Download Chip	Ctrl.	￥4.4800	1	￥4.4800
SS34	TWGMC	Schottky Diode (3.0 Ampere Schottky Barrier Rectifiers)	Ctrl.	￥0.1650	1	￥0.1650
AMS1117-3.3	AMS	AMS1117 1A Low Dropout Voltage Regulator	Power	￥0.8870	1	￥0.8870

Part #	Mft.	Description	For	Price	Qty.	Total
AFC01-S08FCA-00	JS	FPC Connector (0.5Pitch H=2.0 Easy-on R/A Type1 SMT CONN)	Ctrl.	￥0.6610	1	￥0.6610
20009-UCAF001-X	Mintron	USB Type C Connector	Ctrl.	￥1.8100	1	￥1.8100
C0603	-	SMD 10μF Capacitor	Ctrl.	￥0.0366	3	￥0.1098
C0603	-	SMD 100nF Capacitor	Ctrl.	￥0.0140	2	￥0.0280
R0603	-	SMD 4.7kΩ Resistor	Ctrl.	￥0.0065	1	￥0.0065
R0603	-	SMD 10kΩ Resistor	Ctrl.	￥0.0057	2	￥0.0114
KH-2.54FH-1X7P-H8.5	Kinghelm	2.54mm Female Header 1x7Pin H8.5 DIP	Ctrl.	￥0.5500	1	￥0.5500
PZ254V-11-02P	XFCN	2.54mm 1*2P Pin Header	Ctrl.	￥0.0898	1	￥0.0898
PZ254V-11-03P	XFCN	2.54mm 1*3P Pin Header	Ctrl.	￥0.1060	1	￥0.1060
PZ254V-11-04P	KFCN	2.54mm 1*4P Pin Header	Ctrl.	￥0.1409	4	￥0.5636
TS-1088-AR02016	Xunpu	Buttons (4*3*2mm SMD Touch Switch)	Ctrl.	￥0.2580	2	￥0.5160
TS-1003S-07026	Xunpu	Buttons (12*12*7mm SMD Touch Switch)	Ctrl.	￥0.4460	3	￥1.3380
PCB Total						￥31.9321
Total						￥227.0221

Table 1: Costs for Parts

4.3 Sum of Total Costs

The grand total cost is

$$\begin{aligned}\text{Total Costs} &= \text{Labor Costs} + \text{Parts Costs} \\ &= ￥240,000 + ￥227.0221 \\ &= ￥240,227.0221\end{aligned} \tag{10}$$

5 Conclusion

5.1 Accomplishments

In this project, we successfully developed a device which can be attached on the doors with handles in our dormitories, and turns the normal door lock into fingerprint recognition door lock. The device uses ESP32 as the microcontroller, the FPM383C as the fingerprint recognition module, and the servo motor TD-8135MG as the actuator. So that it supports fingerprint registration, management and recognition, as well as opening the door from the inside when the fingerprint is matched. And in order to be energy efficient, an infrared module is used to detect human movement and wake up the device. When no one is around for a period of time, the ESP32 will sleep to save energy. Besides, by introducing the Bluetooth and Wi-Fi, our device supports more measures for unlocking. Except for using the pre-registered fingerprint to open the door, it is also possible for the user to unlock the door through the mobile application on the cellphone. Plus, with Bluetooth technology, the door lock can also unlock automatically when the user approaches the door with its mobile phone. In order to manage multiple client devices and users, we have also developed a server system based on the HTTP protocol.

5.2 Uncertainties

Due to the differences between each door, the installation position may not be completely the same. Also, since we cannot ensure the strength of our nylon thread under the long-term operation, it may break at some time, so it is highly recommended to use multi-strand of threads. Another uncertainty comes from the wireless connection, the remote unlocking function may be affected since the Wi-Fi and Bluetooth signals may not always be in good condition anywhere in anytime. So when the user are using the remote unlocking method through mobile application or by Bluetooth, it is recommended to ensure stable network environment. Cyber-attack may also cause some safety issues, which we should consider more in the future.

5.3 Ethical Considerations

We intend to do experiments on doors inside our campus Residential College, which means we need approval from Residential College. IEEE Code of Ethics request avoiding real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist.[10] So we need to corporate well with Residential College and if we can't reach an agreement, we should try to do experiments on doors which are allowed.

ACM code of ethics[11] requests that we should avoid harm. While our project doesn't involve any organic living things, we still need to do experiments on doors. If not dealt in an appropriate way, we may cause damage to the door, which include stressing too much weight on the door knob, opening or closing the door in a rude way etc. In all cases, we should take full consideration before we start to do any experiments.

ACM code of ethics[11] requests that we should be honest and trustworthy. Our device's basic functionality includes unlocking a door using fingerprint and phone. We can achieve fingerprint recognition using specialized fingerprint sensors but we should achieve that using a simple touching sensors, which means not everyone can unlock the door. Also, for the remote control, we can use Wi-Fi, Bluetooth, SIM card or any other way. But in any cases, we must be transparent about our ways to achieve remote control.

5.4 Safety Considerations

As our fingerprint recognition door lock system requires electricity to work, we must pay attention to the usage of battery and follow ECE 445 safety guideline: Any group charging or utilizing certain battery chemistries must read, understand, and follow guidelines for safe battery usage.[12]. Battery is necessary for our project as we need to attach our device to a door. As our device is intended to be small and not having too much weight, we will use lithium batteries. However, lithium batteries are substantially more flammable.[13]. So we must pay attention to the usage of battery and we will use charger from laboratory to charge the battery. All team members have attended fire extinguisher training and there's fire extinguisher in our Residential College. In anyway, when we do experiments inside laboratory or on doors, we must be careful with the battery problem.

5.5 Future Work

Portability. Currently, our device is not readily detachable from the dormitory door, with its wires still exposed and fixed through adhesive tapes. Although strong adhesive tapes aid in securing the device to the door, the process of removal and attachment to another door proves cumbersome. To enhance portability, other containment methods can be explored to conceal the exposed wires. Alternative attachment mechanisms (e.g.vacuum suction cups) can be evaluated for their suitability.

Generalizability. Our device is compatible only with doors equipped with handles. Enhancing its functionality to include door knobs would broaden its applicability. Furthermore, considering the variation in the required force to operate handles across different doors, research into a more universally applicable torque requirement is essential. Moreover, not all doors feature holes on the side for conventional attachment methods. In such instances, it is imperative to devise an alternative solution, possibly a device affixed to the front of the door, enabling wireless communication with our ESP32 controller.

References

[1] Square. “Okhttp overview.” (2024), [Online]. Available: https://square.github.io/okhttp/ (visited on 05/10/2024).

[2] Google. “Volley overview.” (2021), [Online]. Available: https://google.github.io/volley/ (visited on 05/10/2024).

[3] Square. “Retrofit: A type-safe http client for android and java.” (2024), [Online]. Available: https://square.github.io/retrofit/ (visited on 05/10/2024).

[4] Google. “Httpurlconnection — android developer.” (2024), [Online]. Available: https://developer.android.com/reference/java/net/HttpURLConnection (visited on 05/10/2024).

[5] Texas Instruments, *LM317 3-Terminal Adjustable Regulator*, LM317, Revised Apr. 2020, Sep. 1997.

[6] Advanced Monolithic Systems, *AMS1117 1A LOW DROPOUT VOLTAGE REGULATOR*, AMS1117.

[7] Google. “Transfer bluetooth data.” (2024), [Online]. Available: https://developer.android.google.cn/develop/connectivity/bluetooth/transfer-data?hl=en (visited on 03/27/2024).

[8] Z. Jianyong, L. Haiyong, C. Zili, and L. Zhaohui, “Rssi based bluetooth low energy indoor positioning,” in *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, 2014, pp. 526–533.

[9] Natlinear, *3A, 4.5V-24V Input，500kHz Synchronous Step-Down Converter*, LN3493, 18, Sep. 2017.

[10] IEEE. “IEEE Code of Ethics.” (2016), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 03/07/2024).

[11] ACM. “ACM Code of Ethics and Professional Conduct.” (2018), [Online]. Available: https://www.acm.org/code-of-ethics (visited on 03/07/2024).

[12] Course Staff. “ECE 445 safety guidelines.” (2024), [Online]. Available: https://courses.grainger.illinois.edu/ece445zjui/guidelines/safety.asp (visited on 03/07/2024).

[13] Course Staff. “Safe Practice for Lead Acid and Lithium Batteries.” (2016), [Online]. Available: https://courses.grainger.illinois.edu/ece445zjui/documents/GeneralBatterySafety.pdf (visited on 03/07/2024).

Appendix A Requirements and Verification Tables

Here are the Requirements and Verification Tables (R&V Tables) for each subsystem, documenting the design goals and the verification methods for each part.

Table 2: Requirements and Verification Table for the Controller Subsystem

Requirements	Verification
1. Identify the strength of BLE signal when devices approach 0.5m $\pm$ 0.1m. 2. Can both receive and transmit over UART at a baud rate of 57600bps with FPM383C. 3. Interacting with the server with a delay of no more than 3 seconds.	1. (a) Establish a connection between the mobile phone and the ESP32 through the software app we developed. (b) Approach the ESP32 microcontroller from 1 meter away. Record the distance between them when the door was opened. Repeat this step for 3 times (c) Compare the distance we record with the target requirements (0.5m $\pm$ 0.1m) 2. (a) Run the code that set the baud rate of Serial2 (GPIO16&17) to 57600bps and send a command packet to FPM383C (we test it with the command that set LED to blue). (b) Connect Serial2 to FPM383C. (c) Ensure that Serial2 receive 0x00 (desired reply packet) and the color of LED is blue. 3. (a) Send data packet with the size range from 1byte to 1024bytes (b) Record the time we receive the reply packet.

Table 3: Requirements and Verification Table for the Sensor Subsystem

Requirements	Verification
1. Able to wake up when people stand in front of the door within 1m $\pm$ 0.2m. 2. Able to collect and store the fingerprint of at least 5 users. 3. Identify the user fingerprint in less than 2 seconds after touching, with a minimum accuracy of 80%. 4. Can reply the command packet over UART at a baud rate of 57600bps.	1. (a) Approach the infrared detector from 2 meter away. Record the distance when the ESP32 is awoke. Repeat this step for 3 times (b) Compare the distance we record with the target requirements (1m $\pm$ 0.2m) 2. (a) clear the fingerprint storage in FPM383C. (b) Register 8 different fingerprint. (c) Test the above 8 fingerprint. 3. (a) Register a new fingerprint. (b) Press the sensor in different angles (0,$\pm$45°,$\pm$90°). (c) Record the result and calculate the accuracy. 4. (a) Run the code that set the baud rate of Serial2 (GPIO16&17) to 57600bps and send a command packet to FPM383C (we test it with the command that set LED to blue). (b) Connect Serial2 to FPM383C. (c) Ensure that Serial2 receives 0x00 (desired reply packet) and the color of LED is blue.

Table 4: Requirements and Verification Table for the Mobile Application

Requirements	Verification
1. Sent packets and received packets are the same and transmitted within 0.1 second when mobile phone and ESP32 are 5 meters away. 2. Communication protocol is finished within 3 seconds. 3. Maximum waiting time for unlocking door should be 3 seconds and device is disabled after 60 seconds without use.	1. (a) Send data of different sizes (1 byte to 1024 bytes) and check that it remains the same on received side. (b) Write a test program to record message round trip time (RTT). Check if RTT is smaller than 0.2 seconds at a distance of 5 meters. 2. (a) Record time difference between *INIT* signal and *FINISH* signal for many times and compute the average. Make sure the average is within 6 seconds. 3. (a) Manually disable the motor and check that error message is sent on the mobile software after 10 seconds. (b) 60 seconds after door is unlocked, check that Bluetooth connection between phone and ESP32 is closed.

Table 5: Requirements and Verification Table for the Server Application

Requirements	Verification
1. Direct Interaction: The server app must enable secure communication between a single mobile device and one door lock within 2 seconds of request initiation. 2. Scalability: The app should support interactions with multiple mobile devices and door locks simultaneously, with no degradation in performance. 3. User Registration and Identity Verification: The database must support user registration and validate identities in under 5 seconds to ensure personalized access. 4. Secure Connection: All communications between mobile devices and door locks must be encrypted and secure. 5. Token and ID Validation: The server must validate tokens and IDs within 1 second before processing any requests.	1. Direct Interaction Verification: Measure the time taken from request initiation to door lock activation for a single mobile device and verify that it is within the required 2-second limit. 2. Scalability Verification: Conduct stress tests by increasing the number of mobile devices and door locks connected to the server both to 4 to ensure stable performance and reasonable response time. 3. User Registration and Identity Verification Testing: Simulate user registration processes and verify that the system authenticates identities and provides access within the 5-second threshold. 4. Secure Connection Assessment: Perform security audits to confirm that the encryption standards are met and that all data transmissions are secure. 5. Token and ID Validation Assessment: Implement automated tests that check the server's response times for token and ID validation to ensure they meet the 1-second requirement.

Table 6: Requirements and Verification Table for the Mechanical Subsystem

Requirements	Verification
1. Some brackets for holding all the device components strongly on the door. 2. A powerful enough servo motor to drive the mechanical actuator, approximately with its torque larger than 25kg·cm. 3. A reliable mechanical actuator with links strong enough to move the door handle to a angle of at least 45°.	1. (a) Install the brackets on the door, apply a force $\geq 25N$ on each bracket continuously for 3 minutes, test if the bracket can resist this impulse. (b) Put 500g weights as loads on each bracket, leave them for a day, test if the bracket can still connect reliably with the door. 2. (a) Use the PWM to drive the servo motor, then use the force gauge to test whether the motor can produce more than 25kg force at 1cm away from the axis. (b) Use the PWM to drive the servo motor, then use the nylon thread to pull the handle down, test if the motor can open the door by pulling the nylon thread. (c) Modify the position of the motor, angle of pulling, voltage input, etc. to ensure that we can open the door reliably. 3. (a) Use different materials of threads, like nylon, cotton, or hemp ropes, knot them tightly with the handle and the motor. (b) Use the PWM to drive the servo motor, and pull the handle, measure the angle using a protractor when the motor is stuck, see if it reaches 45°.

Table 7: Requirements and Verification Table for the Power Subsystem

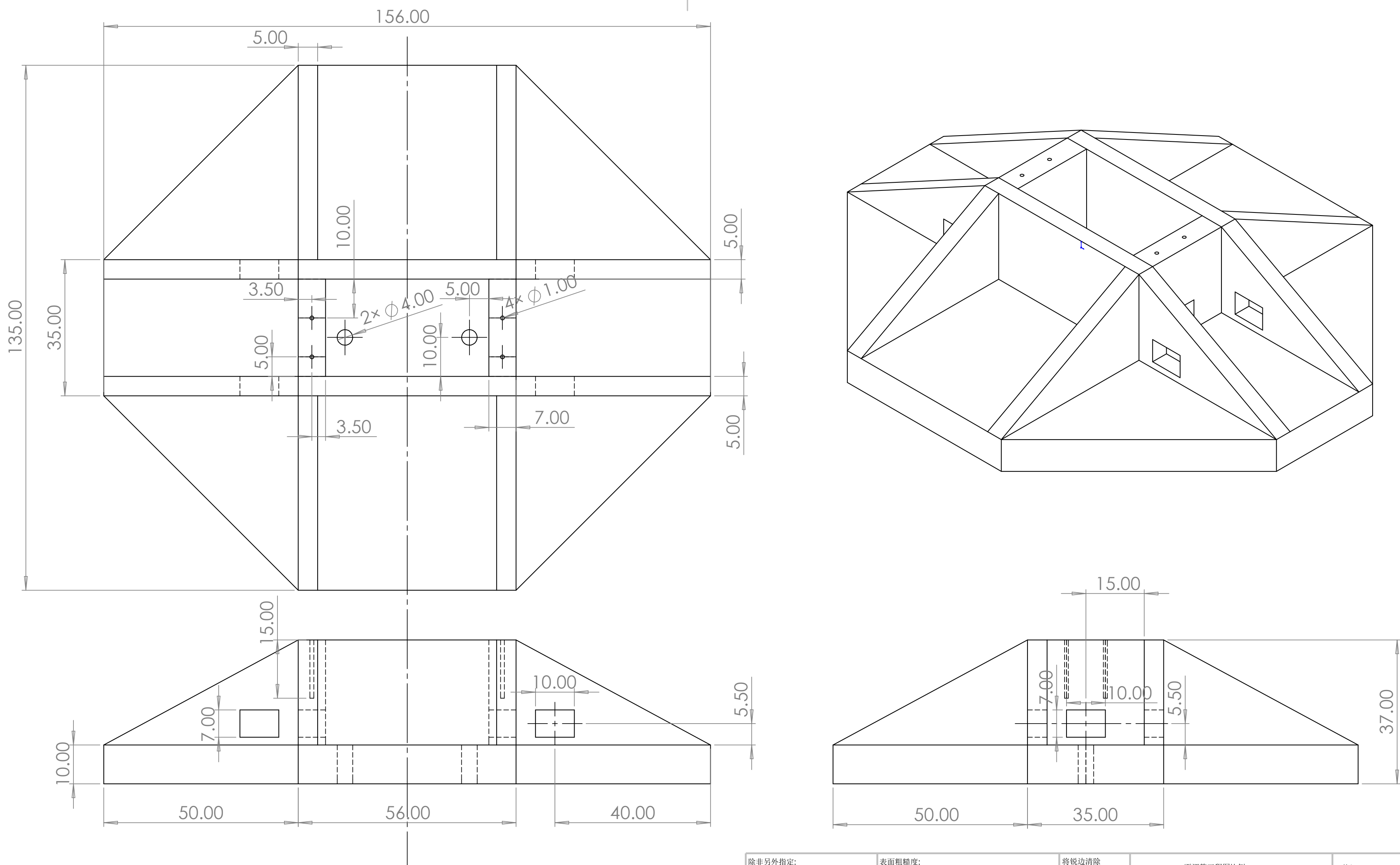
Requirements	Verification
1. Provide at least 200mA, stable 3.3V power supply for the Controller Subsystem and the Sensor Subsystem. 2. Provide at least 200mA, stable power supply in the range of 4.8∼7.2V for the servo motor to work normally. 3. Can automatically switch to the backup batteries when the main battery dies out.	1. (a) Use the voltmeter to measure the output voltage of the voltage regulator, see if it is a stable 3.3V output. (b) Connect the load (the microcontroller and the sensors), use the ammeter to measure the output current, for both when they are working or sleeping. 2. (a) Use the voltmeter to measure the output voltage of the voltage regulator, see if it is a stable output in 4.8∼7.2V. (b) Connect the load (the servo motor), use the ammeter to measure the output current, for both when the motor is working or sleeping. 3. (a) Use an adjustable constant current voltage source to sweep the power from 12V down to 0, in order to simulate the situation where the main battery dies out. (b) Monitor the output voltages using the oscilloscope, see whether the output voltages are still stable, one in 3.3V, another in 4.8∼7.2V.

Appendix B Mechanical Subsystem Drawings

The following 3 pages are the CAD Drawings for the Mechanical Subsystem, and these 3 parts can be glued to the door by the adhesive tape. The first one is the bracket for the servo motor, which limits the motion and rotation of the servo motor. The motor can be fastened on the servo box by 4 self-tapping screw, so that it can be tightly fixed against the rotation torque.

The second drawing is the bracket for the 18650 lithium batteries. According to our plan, the PCB board should also be installed here.

The third drawing is the bracket for the sensors. The top slot is for the fingerprint recognition module, and the cavity in the middle is for the infrared module.



除非另外指定:
尺寸使用毫米
表面粗糙度:
公差:
线性:
角度:

表面粗糙度:

将锐边清除
毛刺并折断

不调整工程图比例

修订

	名称	签名	日期
绘制			
检查			
批准			
制造			
验检			

材料:

重量:

标题:

工程图号

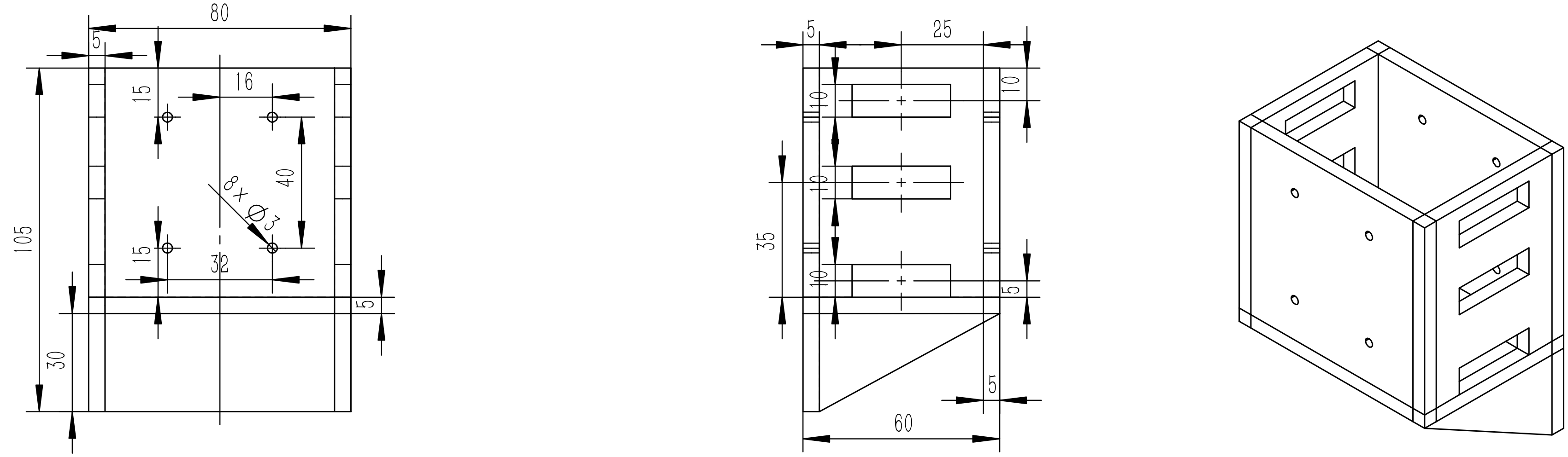
Servo Box

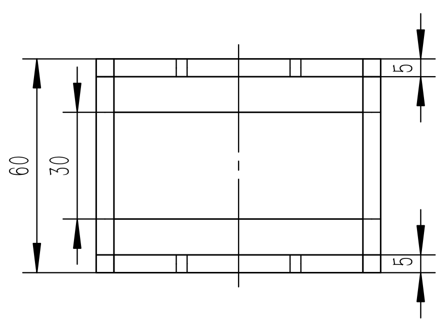
A3

比例:1:1

图纸 1 (共1)

BB





零件代号	
借(通)用件登记	
旧底图总号	
底 图 总 号	
签 字	
日 期	

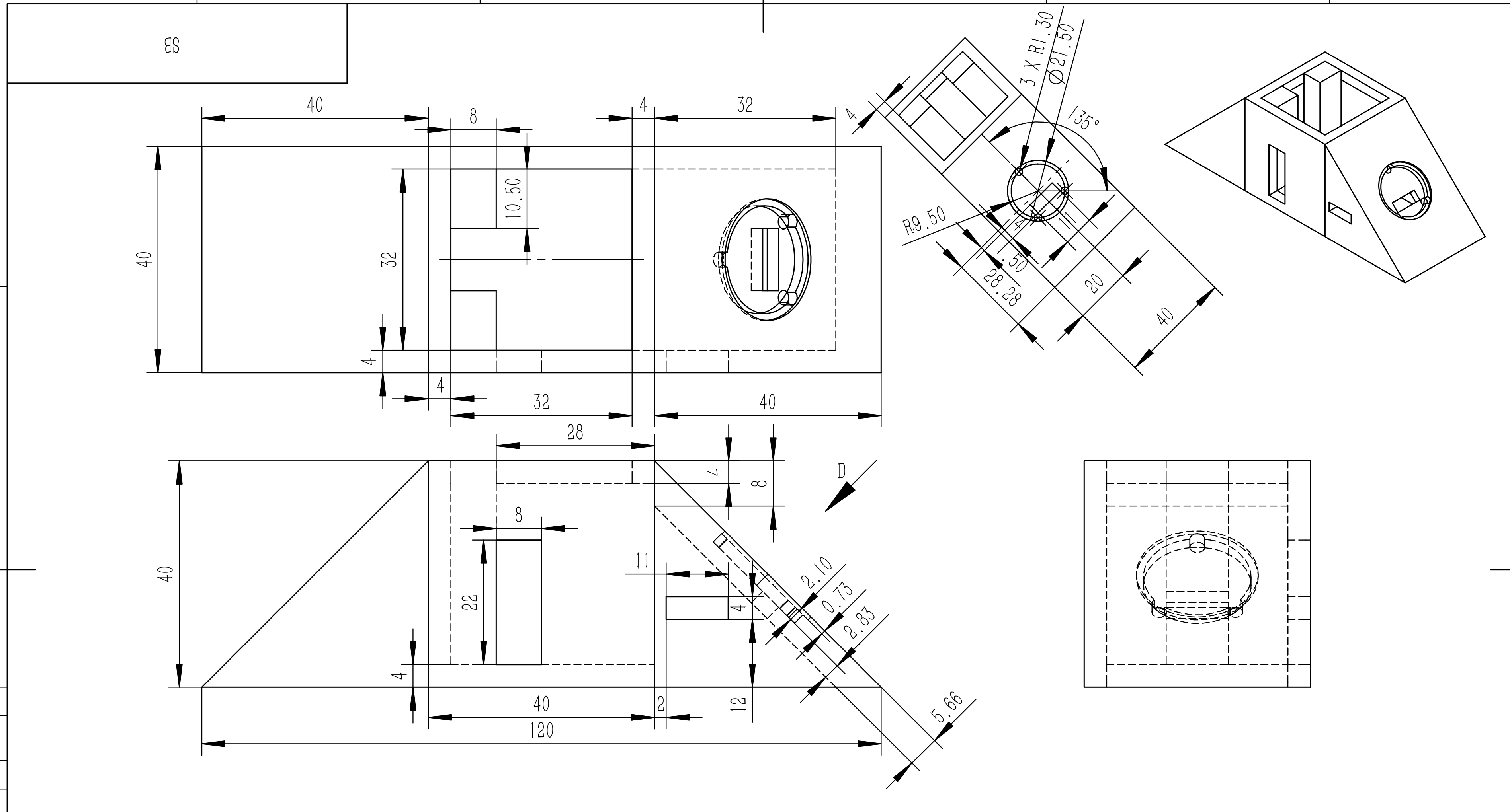
标记	处数	分区	更改文件号	签名	年 月 日
设计			标准化		
校核			工艺		
主管设计			审核		
			批准		

材质 <未指定>		
阶 段 标 记	Mass	Scale
	0.126	1:2
共1 张 第1张	版本	

Battery Box

BB

替代



						材质 <未指定>		
标记	处数	分区	更改文件号	签名	年 月 日	阶 段 标 记	质量	比例
设计			标准化				0.075	1:1
校核			工艺					Sensor Box
主管设计			审核					SB
			批准			共1 张 第1张	版本	替代

零件代号

借(通)用件登记

旧底图总号

底 图 总 号

签 字

日 期

Appendix C PCB Design

Here is the schematics and the board layout of the PCB.

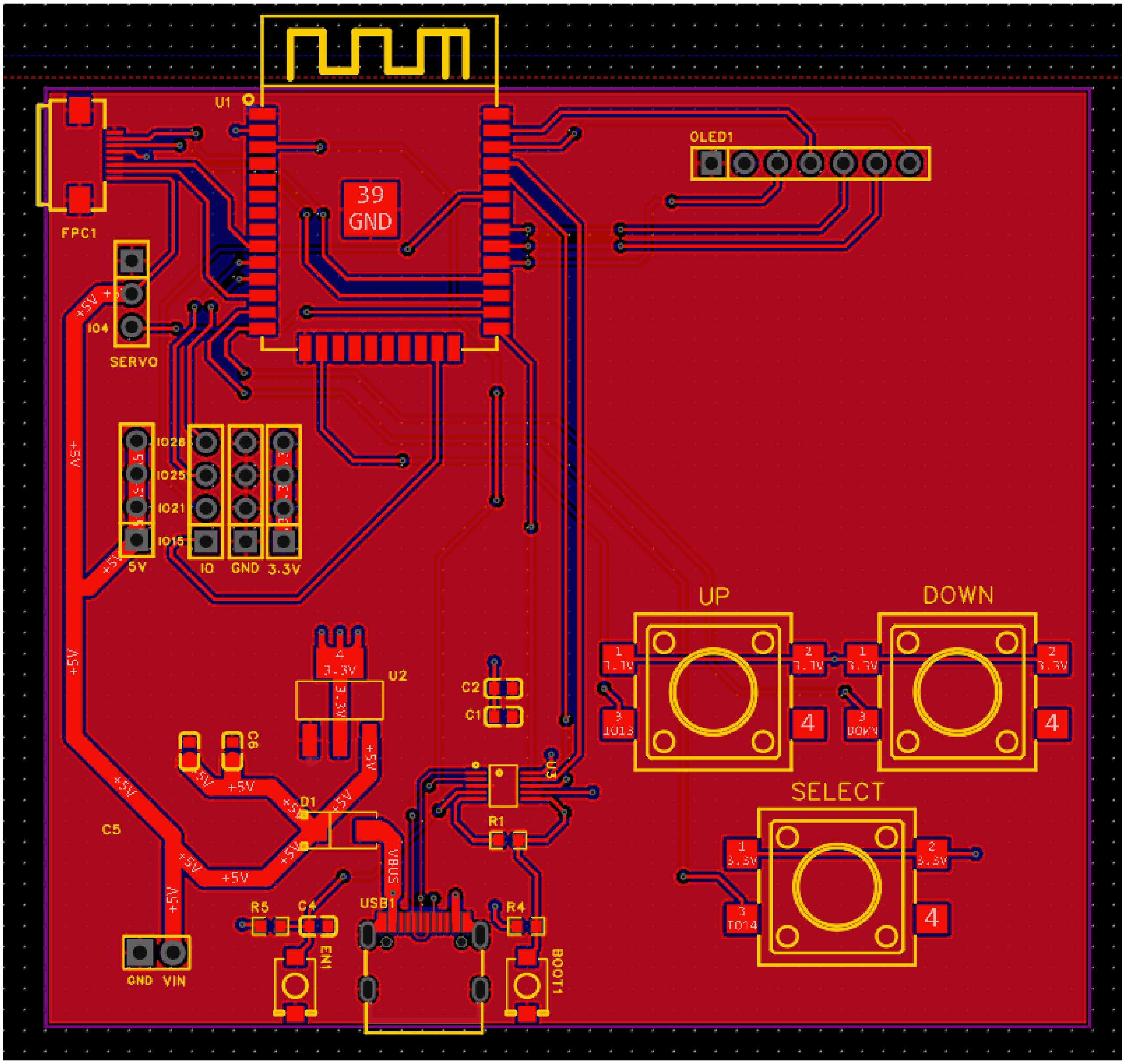


Figure 10: PCB Layout (Front)

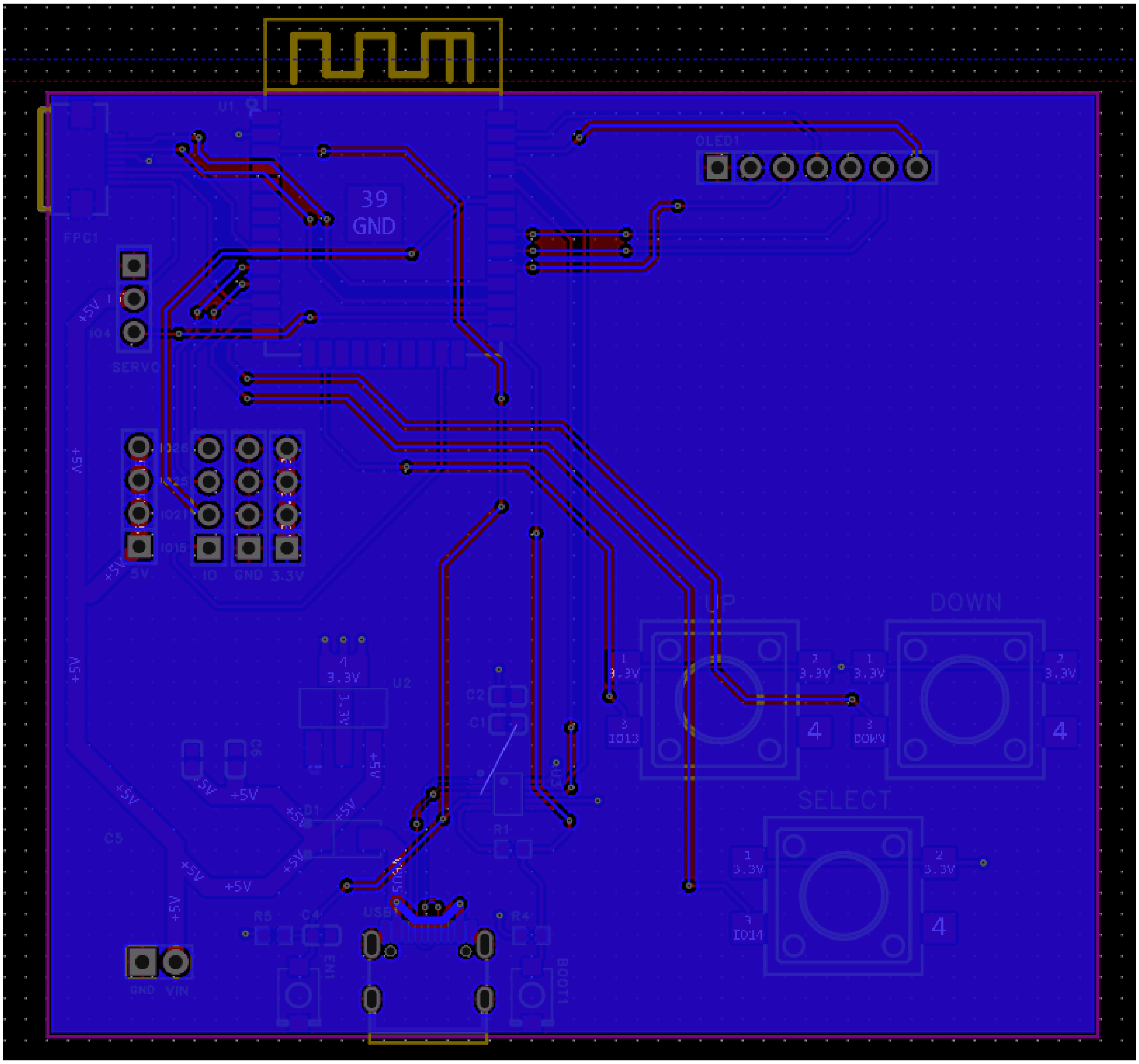
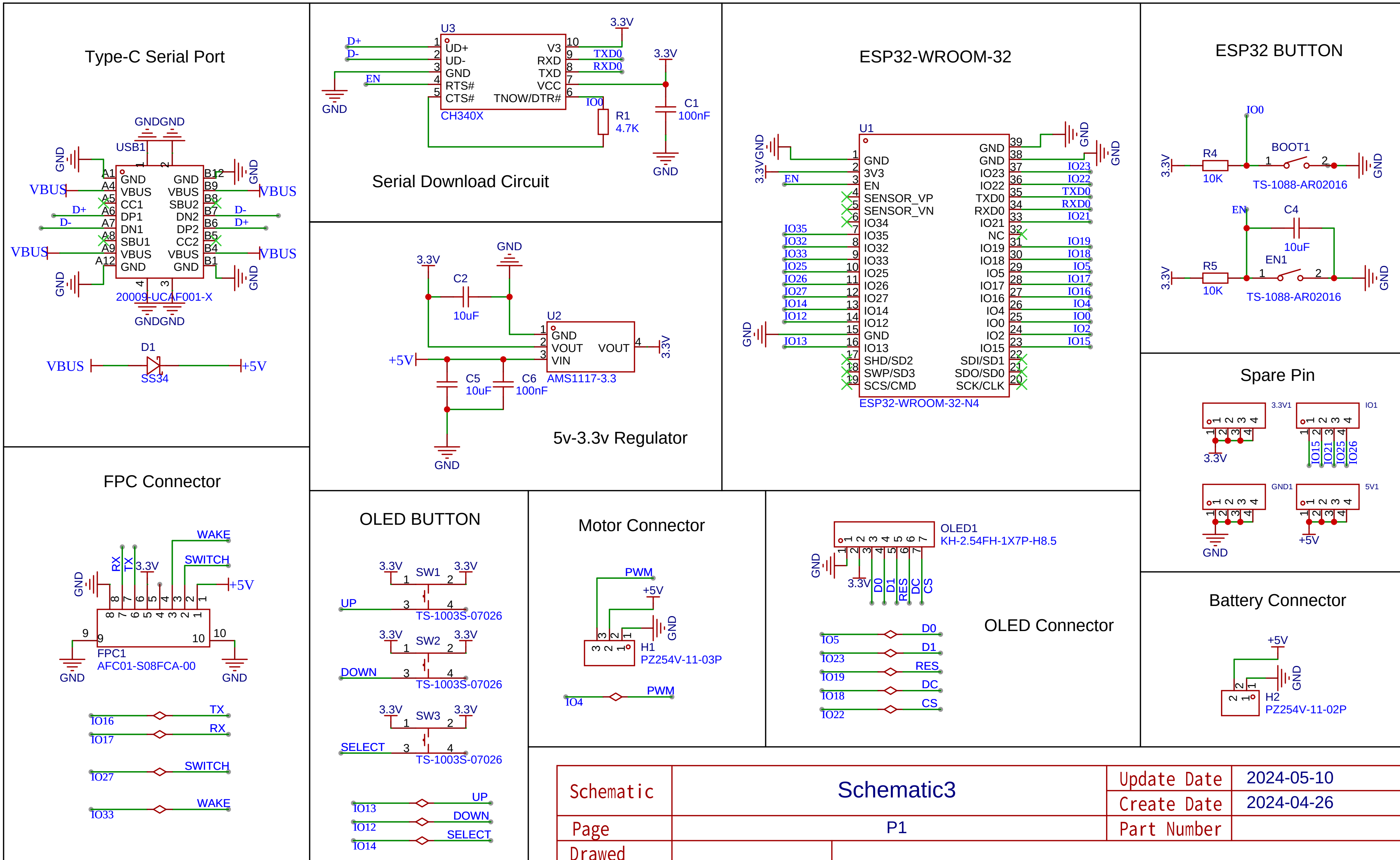


Figure 11: PCB Layout (Back)



Schematic	Schematic3		Update Date	2024-05-10
			Create Date	2024-04-26
Page	P1		Part Number	
Drawed		esp32_doorlock		
Reviewed				
		VER	SIZE	PAGE 1 OF 1
嘉立创EDA		V1.0	A4	