



University
of Basel

Fingerprint Lock with LCD Display

Computer Architecture

University of Basel, Fall 2024

Luca Fässler, Barththan Sivanantham, Valerio Job

Project Goal

Ambition

To build a secure access system that uses biometric authentication with a fingerprint sensor, controlled by an Arduino, to ensure a secure lock

- **Secure Access:** Biometric system using fingerprint recognition to control entry
- **Real-Time Feedback:** LCD display shows status (e.g., "Access Granted" or "Access Denied")
- **Modular Design:** Easily adaptable for various access control applications

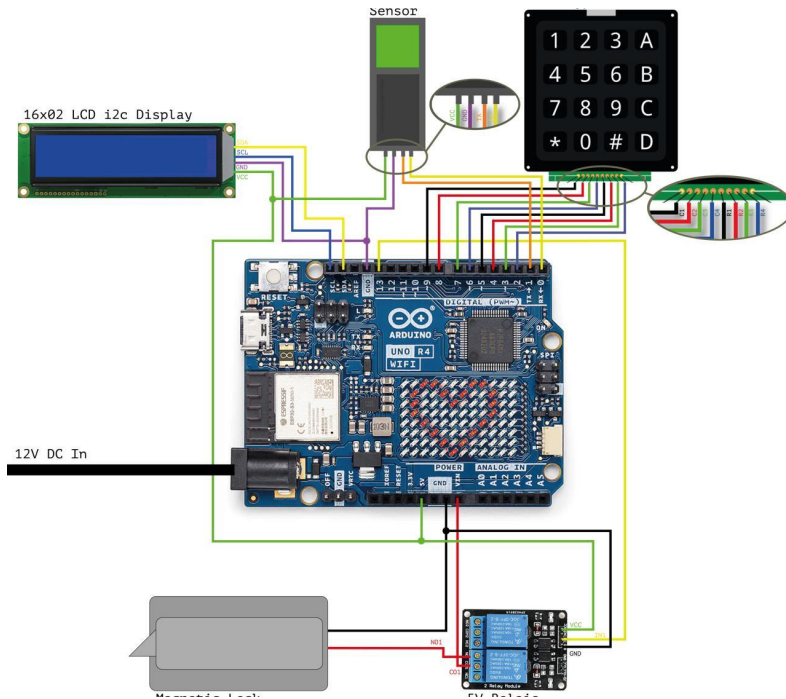
Project Goal

Ambition

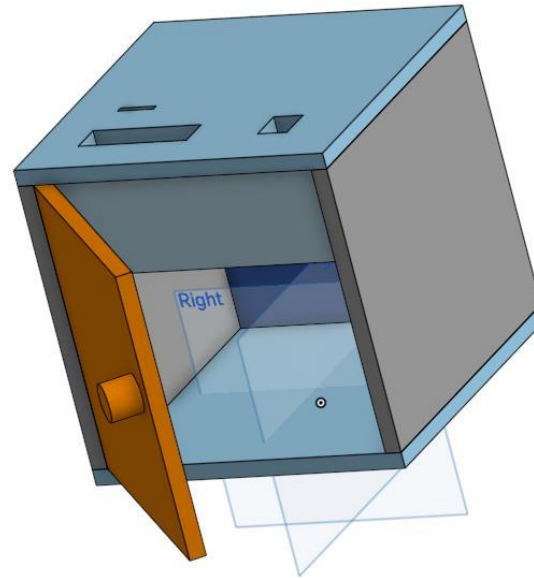
To build a secure access system that uses biometric authentication with a fingerprint sensor, controlled by an Arduino, to ensure a secure lock

- **Secure Access:** Biometric system using fingerprint recognition to control entry
- **Real-Time Feedback:** LCD display shows status (e.g., "Access Granted" or "Access Denied")
- **Time-Based Access:** Feature to restrict or allow access based on specific time intervals, enhancing security and operational flexibility
- **Modular Design:** Easily adaptable for various access control applications

System Overview



Circuit Diagram



CAD Visualization



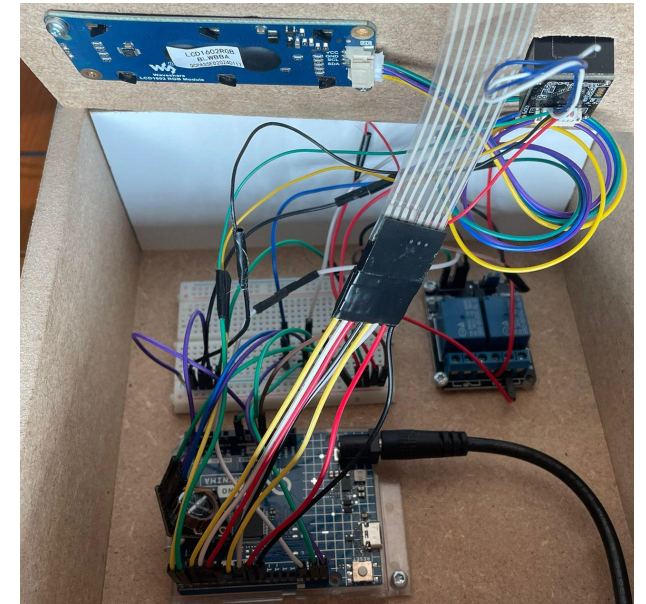
Final Result

Hardware Implementation

Hardware

- Arduino UNO R4 **Minima/WIFI**
- R307 Fingerprint Reader
- 16x2 I2C LCD Display
- 12V DC Electromagnetic Lock
- 2-Channel Relay Module
- 12V DC Power Supply
- Jumper Wires & Breadboard
- **RTC SD3031**
- Keypad 4x4 Matrix
- Wood, Hinge, Door Handle

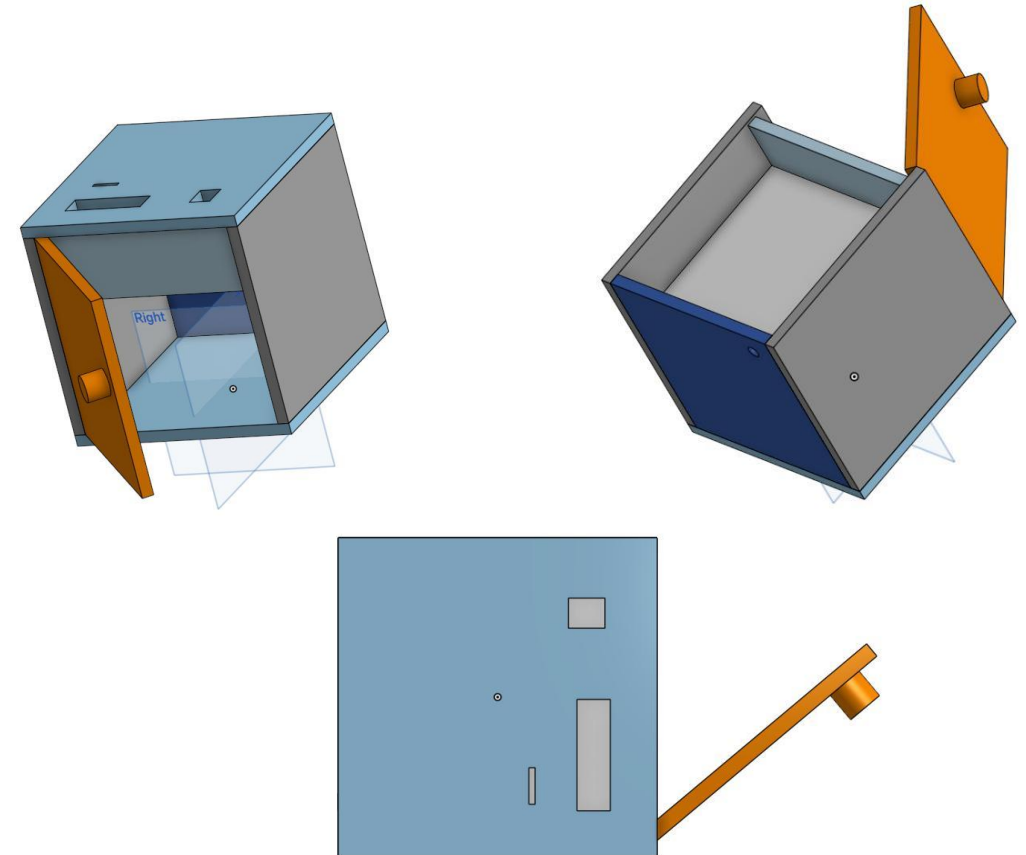
Total HW Costs: 236.66 CHF (initially Planned: 106 CHF)



Hardware Implementation

Hardware

- Arduino UNO R4 **Minima**/WIFI
- R307 Fingerprint Reader
- 16x2 I2C LCD Display
- 12V DC Electromagnetic Lock
- 2-Channel Relay Module
- 12V DC Power Supply
- Jumper Wires & Breadboard
- **RTC SD3031**
- Keypad 4x4 Matrix
- Wood, Hinge, Door Handle



Total HW Costs: 236.66 CHF (initially Planned: 106 CHF)

Software Implementation

Programming Environment

- Developed using Arduino IDE (C++)
- Serial communication for debugging

Core Functionalities

- Fingerprint Enrollment and Verification
- Time-Based Access Control
 - *Access to predefined time slots*
- LCD Display Handling
 - *User Feedback Messages*
- Relay Control
 - *Lock Activation upon Verification and Time*

```
1 bool fingerEnroll(int id) {
2     // Enroll a fingerprint for the given ID
3     for (int i = 0; i < 2; i++) {
4         if (i == 0) {
5             lcd.clear();
6             lcd.send_string("Scan Finger");
7         } else {
8             lcd.clear();
9             lcd.send_string("Scan Finger");
10            lcd.setCursor(0, 1);
11            lcd.send_string("Again");
12        }
13
14        // Wait until a finger is detected
15        while (finger.getImage() != FINGERPRINT_OK) {
16            // Stay here until a finger is detected
17        }
18
19        if (finger.image2Tz(i + 1) != FINGERPRINT_OK) {
20            lcd.clear();
21            lcd.send_string("Error Capturing");
22            delay(2000);
23            return false; // Error transferring the fingerprint image
24        }
25
26        if (i == 0) {
27            lcd.clear();
28            lcd.send_string("Remove Finger");
29            delay(2000); // Allow time to remove the finger
30        }
31    }
32
33    // Create and store the fingerprint model
34    if (finger.createModel() != FINGERPRINT_OK) {
35        lcd.clear();
36        lcd.send_string("Error Creating");
37        delay(2000);
38        return false; // Error creating the model
39    }
40
41    if (finger.storeModel(id) != FINGERPRINT_OK) {
42        lcd.clear();
43        lcd.send_string("Store Failed");
44        delay(2000);
45        return false; // Error storing the model
46    }
47
48    return true; // Successfully stored
49 }
```

Software Implementation

Programming Environment

- Developed using Arduino IDE (C++)
- Serial communication for debugging

Core Functionalities

- Fingerprint Enrollment and Verification
- Time-Based Access Control
 - *Access to predefined time slots*
- LCD Display Handling
 - *User Feedback Messages*
- Relay Control
 - *Lock Activation upon Verification and Time*

```
1 void loop() {
2     // Update time from NTP server
3     timeClient.update();
4     String currentTime = timeClient.getFormattedTime();
5     int currentHour = timeClient.getHours(); // Get the current hour
6
7     // Display default screen if not in menu mode
8     if (!inMenu) {
9         ...
10    }
11
12    // Read keypress from keypad
13    char key = keypad.getKey();
14
15    // Handle keypress actions
16    if (key == '*') {
17        // Enter menu mode
18        ...
19    } else if (key == '#') {
20        // Attempt to unlock
21        if (currentHour >= 8 && currentHour < 20) {
22            // Check fingerprint during allowed hours (8 AM - 8 PM)
23            startFingerprint();
24        } else {
25            // Display time restriction message
26            lcd.clear();
27            lcd.send_string("  No Timeslot  ");
28            lcd.setCursor(0, 1);
29            lcd.send_string("Access: 8AM-8PM ");
30            delay(3000);
31            lcd.clear();
32            showDefaultScreen(); // Return to default screen
33        }
34    }
35 }
```


Challenges Faced

1. Keypad Sensitivity Issues

Problem: Keypad produced ghost inputs and was overly sensitive

2. RTC Module Challenges

Problem: Difficulties in getting the RTC Module to function correctly

3. Power Supply Issues

Problem: Wrong power supply caused multiple failures; components stopped working, Arduino board was not recognized by the computer

4. LCD Backlight Failure

Problem: LCD backlight stopped functioning at full potential

1. **Solution:** Switched to a more stable keypad library, which resolved the issue and improved input accuracy

2. **Solution:** same as 3.

3. **Solution:** Purchased a new Arduino board with a Wi-Fi module, allowing us to fetch time from the network instead of relying on the RTC module

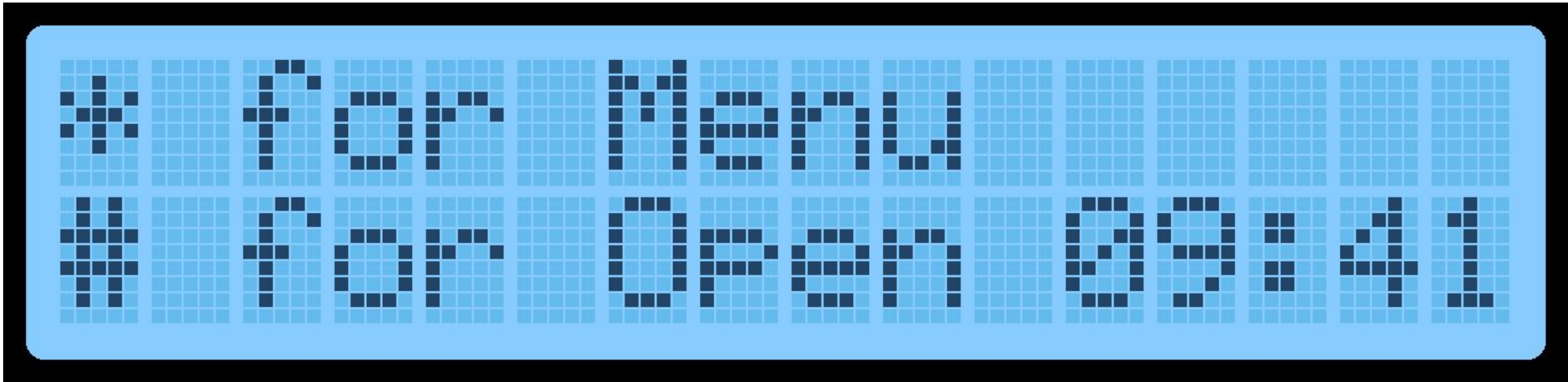
4. **Solution:** Replaced the LCD screen with a new one, which resolved the problem



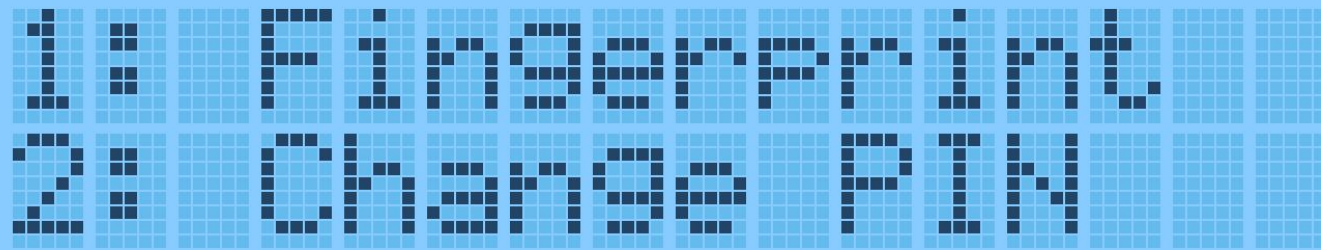
University
of Basel

Demo

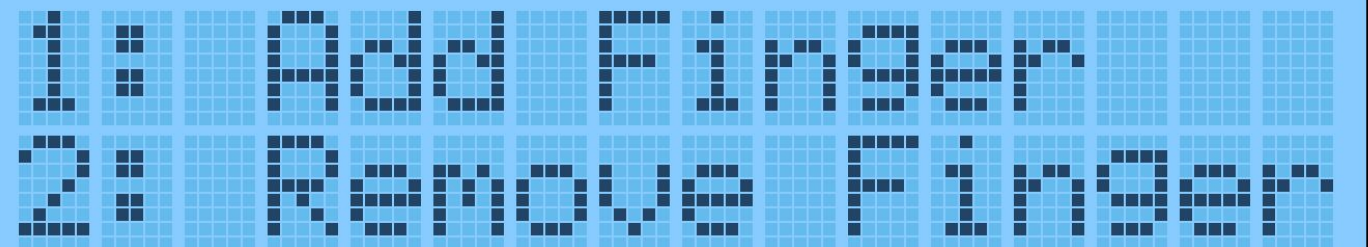
Demo



Demo

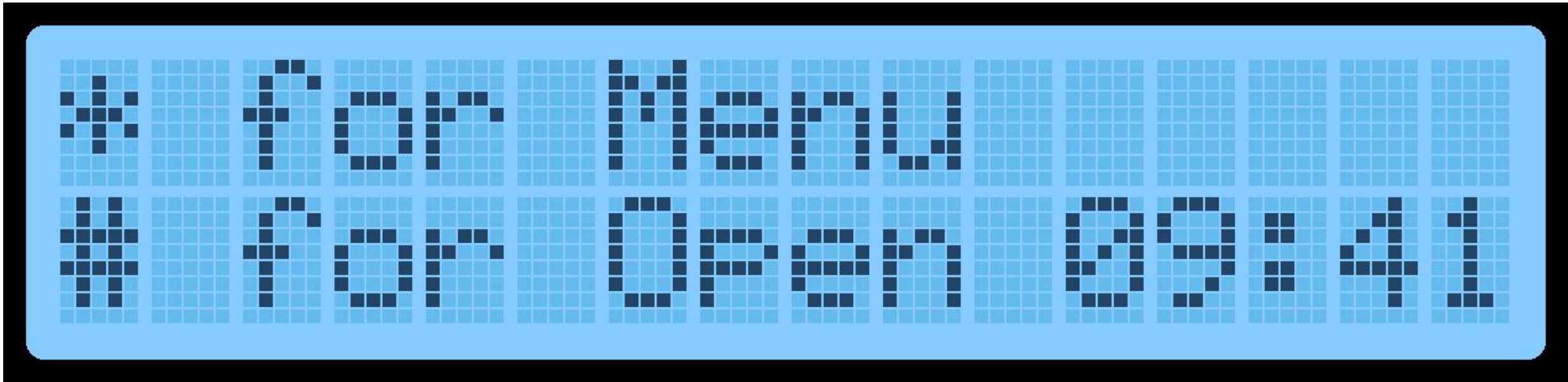


1: Fingerprint
2: Change PIN

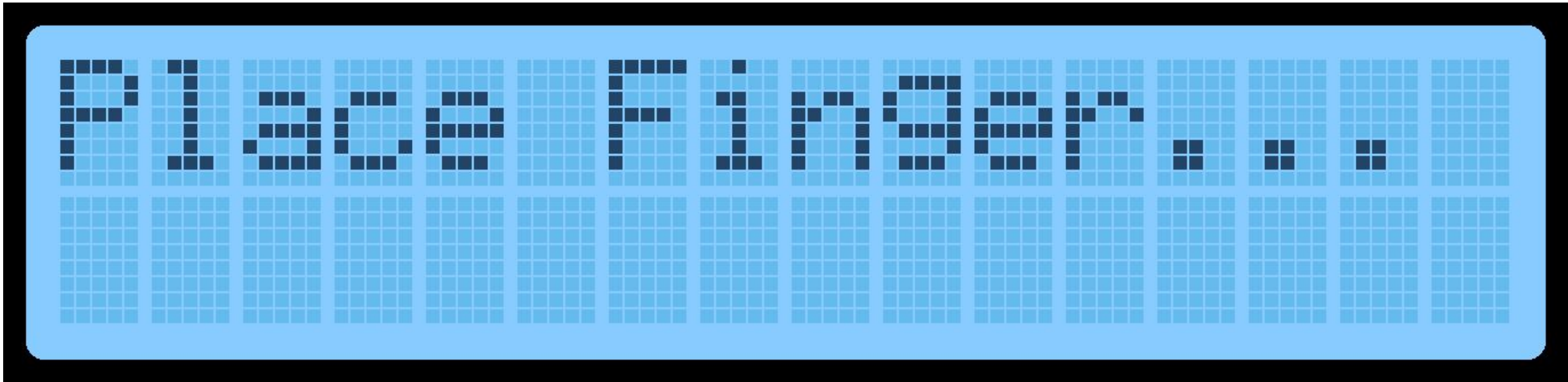


1: Add Fingerprint
2: Remove Fingerprint

Demo



Demo

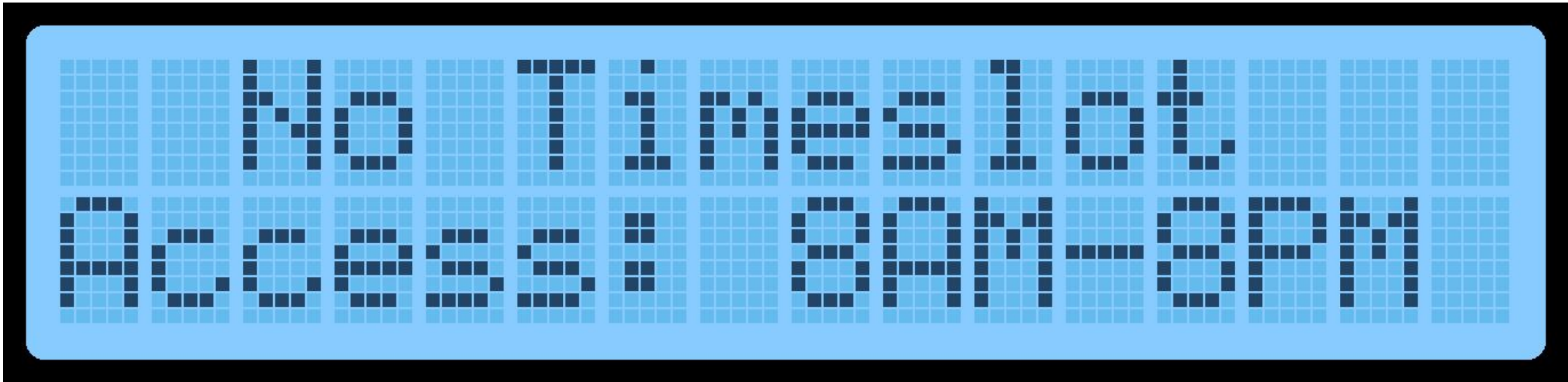


Demo

Access Granted

Access Denied

Demo



Q & A



University
of Basel

Thank you!