CodeLock: Digital Safe System using Raspberry Pi Pico W

1. Project Identification

- Project Name: CodeLock
- Developer: Daniela Amorim de Sá
- Date: 19.12.2024
- Technology Stack: Raspberry Pi Pico, Embedded C, OLED Display, Servo Motor, Buzzer, Keypad

2. Project Overview

Objective: Develop a digital safe system with a **4x4 matrix keypad** for password entry, an **OLED display** for feedback, and a **servo motor** to control access.

Justification: Enhances security by **requiring authentication**, implementing an **auto-lock system**, and allowing **password updates** only after verifying the old password.

Key Features:

- Secure password authentication (4-digit)
- Password change with verification
- Automatic locking after 3 incorrect attempts
- OLED display for real-time feedback
- Buzzer alerts for system status

3. Hardware & Circuit Design

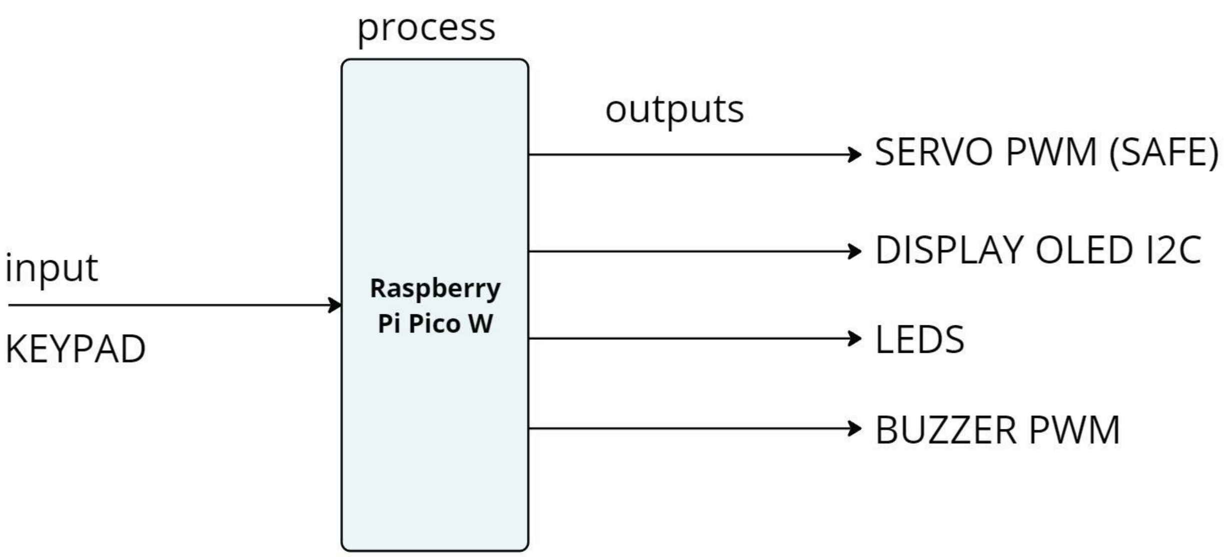
Components Used:

- **Microcontroller:** Raspberry Pi Pico
- **Display:** SSD1306 OLED (I2C)
- **User Input:** 4x4 Matrix Keypad
- **Lock Mechanism:** Servo Motor
- **Feedback System:** Buzzer & LEDs

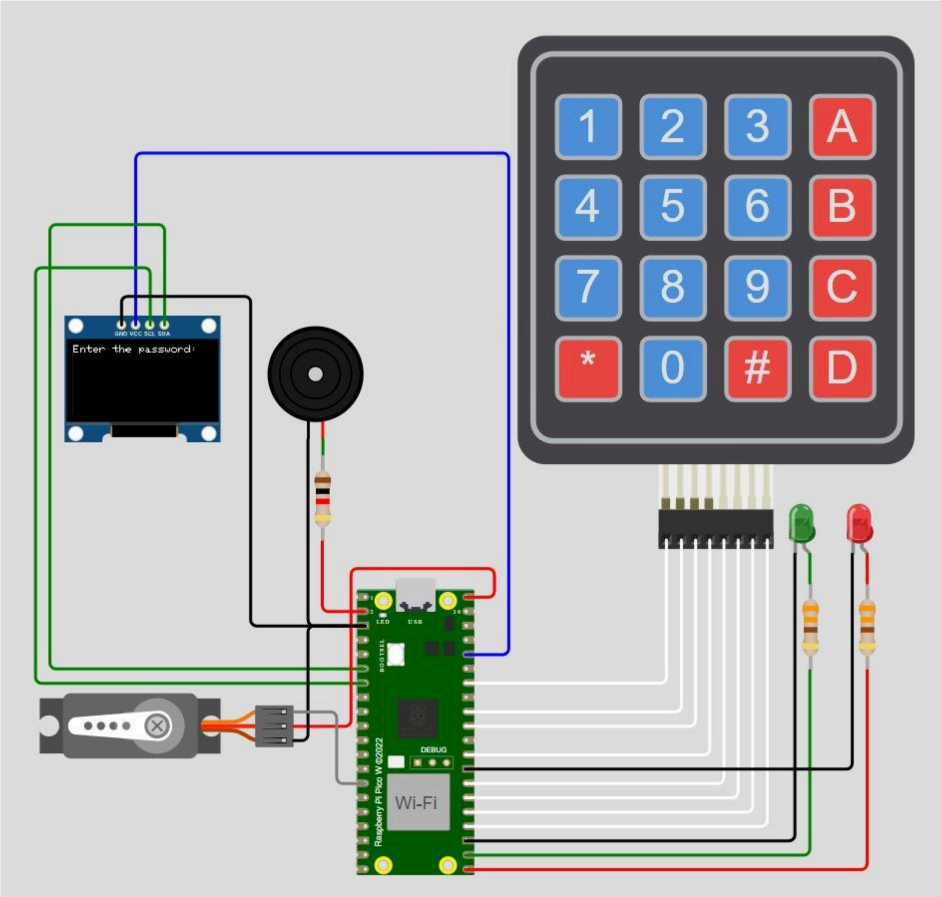
Pin Mapping:

Component	Pins Used
OLED Display	SDA (GP4), SCL (GP5)
Keypad Rows	GP28, GP27, GP26, GP22
Keypad Columns	GP21, GP20, GP19, GP18
Servo Motor	GP15
Buzzer	GP1
Green LED	GP17
Red LED	GP16

Block Diagram:



Complete Circuit:

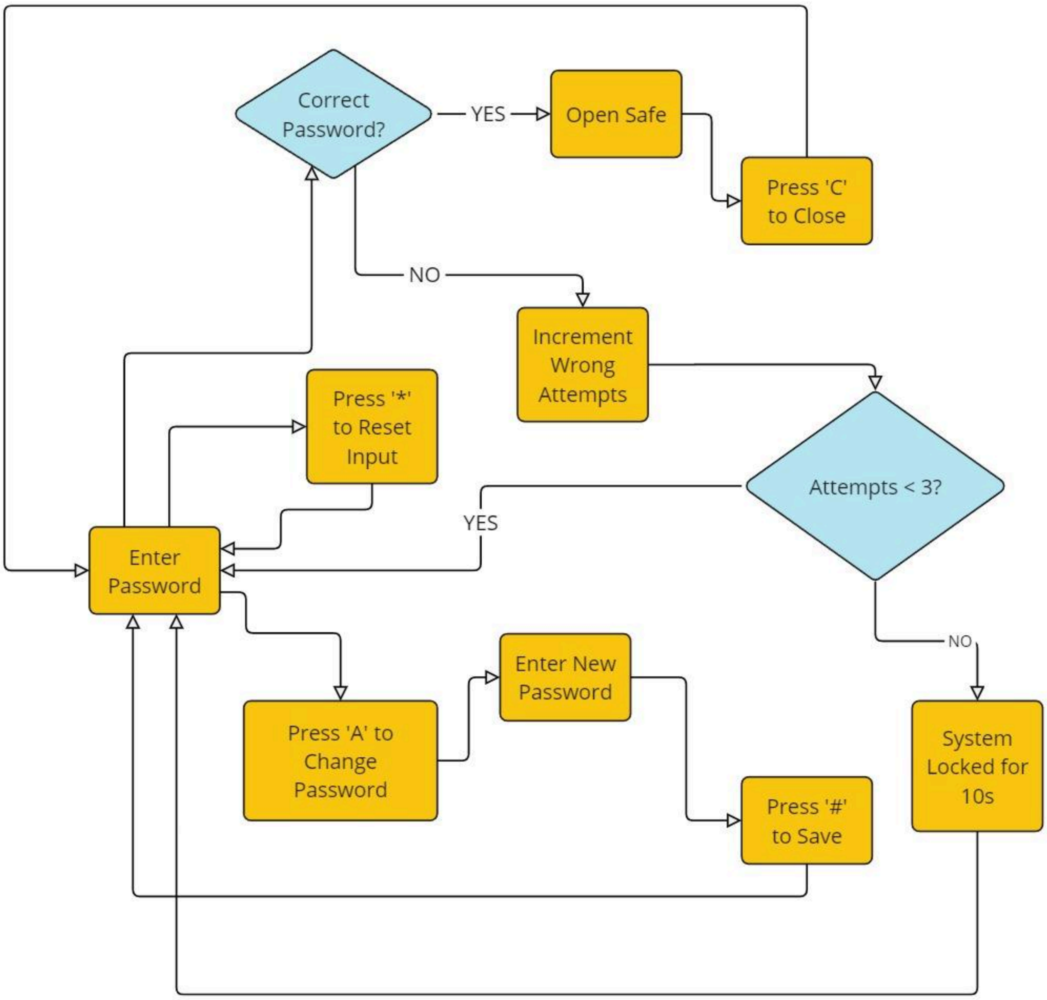


4. Software Architecture & Implementation

Functional Blocks:

- **Main Control (main.c):** Manages the system loop and processes input.
- **OLED Display (oled_display.c/.h):** Handles messages and visual feedback.
- **Password Management (password_manager.c/.h):** Validates, stores, and updates passwords.
- **Keypad Interface (keypad.c/.h):** Reads user input from the matrix keypad.
- **Servo & Lock Control (servo.c/.h):** Opens and closes the lock.
- **Buzzer & Alerts (buzzer.c/.h):** Provides audible feedback.
- **LED Management (led.c/.h):** Indicates status with visual alerts.

Flowchart of Operations:



Security Features Implemented:

- Password change only after verification
- Automatic lock after failed attempts
- Non-volatile password storage (optional for future implementation)

5. System Validation & Results

Testing Scenarios:

- Entering a correct password → Safe opens
- Entering an incorrect password → "Incorrect password" message + buzzer alert
- Three failed attempts → System locks for 10s

- Changing password → Only allowed after old password verification
- Safe closing → Auto-lock mechanism triggered

Results: The system successfully passed all validation tests, ensuring **secure authentication, reliable input handling, and real-time feedback via display and buzzer.**

6. Future Enhancements

- Add **EEPROM storage** for persistent password saving
- Implement **Wi-Fi connectivity (ESP32 upgrade)** for remote unlocking
- Add **time-based auto-locking**

7. Conclusion

The CodeLock is a secure and efficient digital safe system, combining security with ease of use. Its modular design allows for future expansions, such as persistent storage and IoT integration.