RJ LOVERS - Final Report

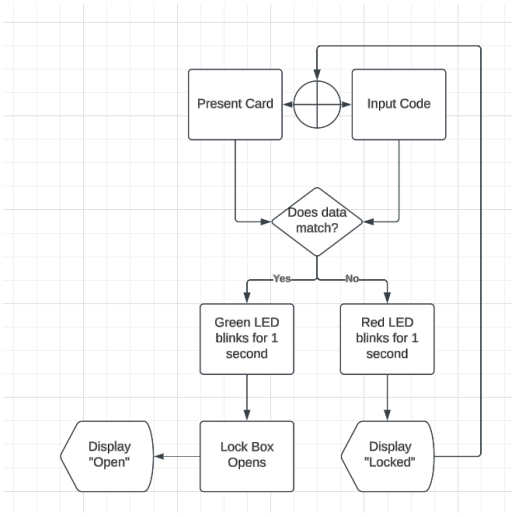
Course: EE 151 Intro to Electrical Engineering Lab
Term: Fall 2024
Members: Charles Bradley, Luke Bejarano, Taran Gill, Aaron Saldana

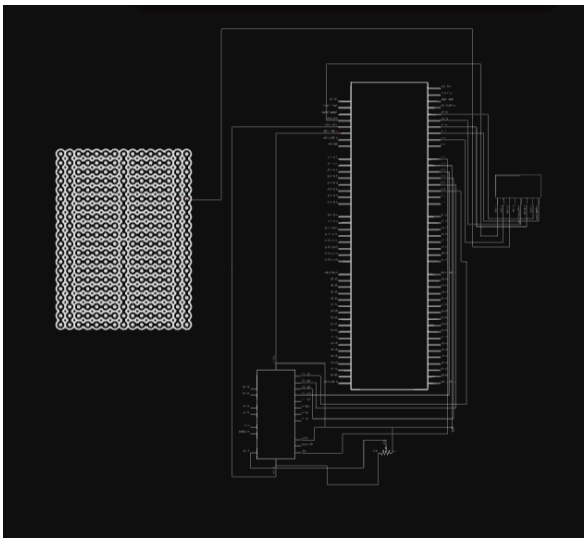
I. Introduction

Our cookie-lock project is a lock system to ensure that you don't eat your footlong cookie before you're done "studying." Similar to our market project, there is a padlock/keypad safe and a hotel door lock. Our project is far more efficient than your average keypad safe because we have incorporated a hybrid lock with an RFID chip. With the chip, it's simply a tap-to-unlock process to access your valuables inside. The hybrid part of the system is the original keypad; if your card becomes demagnetized or lost/ stolen, you can still open the lock.

II. Background

Present a block diagram of your system, a schematic, a discussion of the parts used, and a discussion of the software flowchart.





- We validated the data using the RFID scanner module and the keypad. Then, we displayed the validation results using the LCD, Green, and Red LEDs. Finally, we adjusted the brightness of the LCD Display using the potentiometer
- Our software starts when the card or keypad is inputted. Then, our software validates the data. If the data matches, the green LED will blink, and the LCD will display open. If validation fails, the red LED will blink, and the display will display closed

III. Experimental Results

Project link:https://youtube.com/shorts/CFbu4GHWZcA?si=DnG5bKB0i639Mvfx

We separated the RFID module and the keypad into 2 separate experiments. We used a code to test the UID of the RFID card, which we could then use to test whether the RFID module would open the box. We then coded and tested the keypad to set up a code that could be used to open the box. Then, once we knew they both worked, we put everything into one Arduino sketch.

IV. Conclusion

We had a lot of trouble with the RFID module. At first, the module wouldn't recognize the right or wrong code. After hours of troubleshooting, we figured it out and eventually got it to work on camera. With more time, we could have used a servo as a lock and created a box that actually locked and opened.

Appendix A. Arduino Sketch

```
#include <LiquidCrystal.h>
#include <Keypad.h>
#include <SPI.h>
#include <MFRC522.h>

// Pin Definitions
#define RED_LED 2
#define GREEN_LED 3
#define RST_PIN 7
#define SS_PIN 53

// Initialize RFID Module
MFRC522 rfid(SS_PIN, RST_PIN);

// Initialize LCD
LiquidCrystal lcd(8, 9, 10, 11, 12, 13);

// Keypad Setup
const byte ROWS = 4;  // Four rows
const byte COLS = 4;  // Four columns
char keys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[ROWS] = {23, 25, 27, 29};  // Connect to the row pinouts of
the keypad
```

```
byte colPins[COLS] = {31, 33, 35, 37};  // Connect to the column pinouts
of the keypad
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

// Variables
String passcode = "1234";                   // Default passcode
String inputCode = "";                      // Input buffer for keypad
byte validRFIDUID[] = {0xDE, 0xAD, 0xBE, 0xEF};  // Replace with your
card's UID

void setup() {
  // Initialize serial monitor
  Serial.begin(9600);

  // Initialize LCD
  lcd.begin(16, 2);
  lcd.print("Lockbox Ready");

  // Initialize RFID
  SPI.begin();
  rfid.PCD_Init();

  // Initialize LEDs
  pinMode(RED_LED, OUTPUT);
  pinMode(GREEN_LED, OUTPUT);

  // Turn off LEDs initially
  digitalWrite(RED_LED, LOW);
  digitalWrite(GREEN_LED, LOW);
}

void loop() {
  bool accessGranted = false;

  // Check Keypad Input
  char key = keypad.getKey();
  if (key) {
    if (key == '#') {
      // Submit code
      if (inputCode == passcode) {
        accessGranted = true;
      }
      inputCode = "";  // Reset input
```

```
    } else if (key == '*') {
      // Clear input
      inputCode = "";
    } else {
      // Add to input code
      inputCode += key;
    }
  }

  // Check RFID Input
  if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {
    if (compareRFIDUID(validRFIDUID, rfid.uid.uidByte, rfid.uid.size)) {
      accessGranted = true;
    }
    rfid.PICC_HaltA();  // Halt RFID reader
  }

  // Handle Access Result
  if (accessGranted) {
    lcd.clear();
    lcd.print("Access Granted");
    digitalWrite(GREEN_LED, HIGH);
    delay(5000);
    digitalWrite(GREEN_LED, LOW);
    lcd.clear();
    lcd.print("Lockbox Ready");
  } else if (key == '#' || (rfid.PICC_IsNewCardPresent() &&
!accessGranted)) {
    lcd.clear();
    lcd.print("Access Denied");
    digitalWrite(RED_LED, HIGH);
    delay(5000);
    digitalWrite(RED_LED, LOW);
    lcd.clear();
    lcd.print("Lockbox Ready");
  }
}

// Compare RFID UIDs
bool compareRFIDUID(byte* validUID, byte* readUID, byte uidSize) {
  for (byte i = 0; i < uidSize; i++) {
    if (validUID[i] != readUID[i]) {
      return false;
```

```
    }
  }
  return true;
}
```

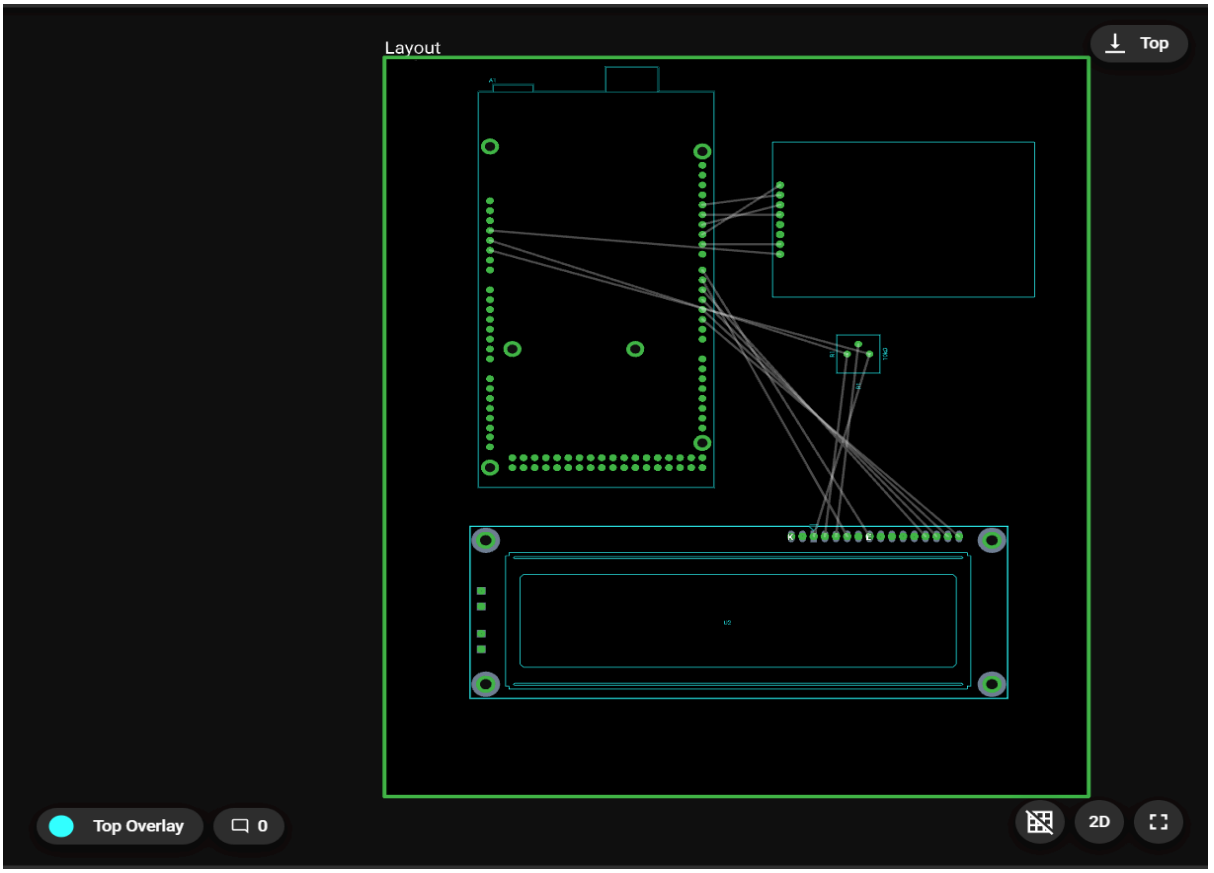
Appendix B. Bill of Materials

Please include your bill of materials here. Please be sure that you can find individual parts at an online retailer such as www.mouser.com. Below is an example of the bill of materials.

Item	Quantity	Cost per unit ($)
Arduino Mega 2560	1	41.14
Breadboard	1	4
Green LED	1	0.38
Red LED	1	0.38
RFID Scanner	1	3
LCD Screen	1	7
Keypad	1	2
Potentiometer	1	1
Male-to-male wires	30	3
Female-to-male wires	7	2.95

Appendix C. Printed Circuit Board Design

Even if you have not fully completed and ordered the PCB, provide a shareable link and screenshots of the schematic and 2D view.


Layout
Top
Top Overlay
0
2D