AHSANULLAH UNIVERSITY OF SCIENCE & TECHNOLOGY

PROJECT REPORT

Project Name: Smart Door Lock System : Built with a Keypad PIN and RFID Door Lock Sensor

Course Title: Microprocessor, Interfacing and System Design Lab

Course Code: EEE 3210

SUBMITTED BY:

Group 01

Israt Jahan Ena
ID: 190205075
Md.Naimul Islam
ID: 20200105218
Tasdik Hossain
ID: 20200105220
Humayra Tabassum
ID: 20200105221
Nasif Zawad
ID: 20200105222

Year and Sem.: 3rd, 2nd
Section: D
Group: D2
Dept. of EEE, AUST

DATE OF SUBMISSION:

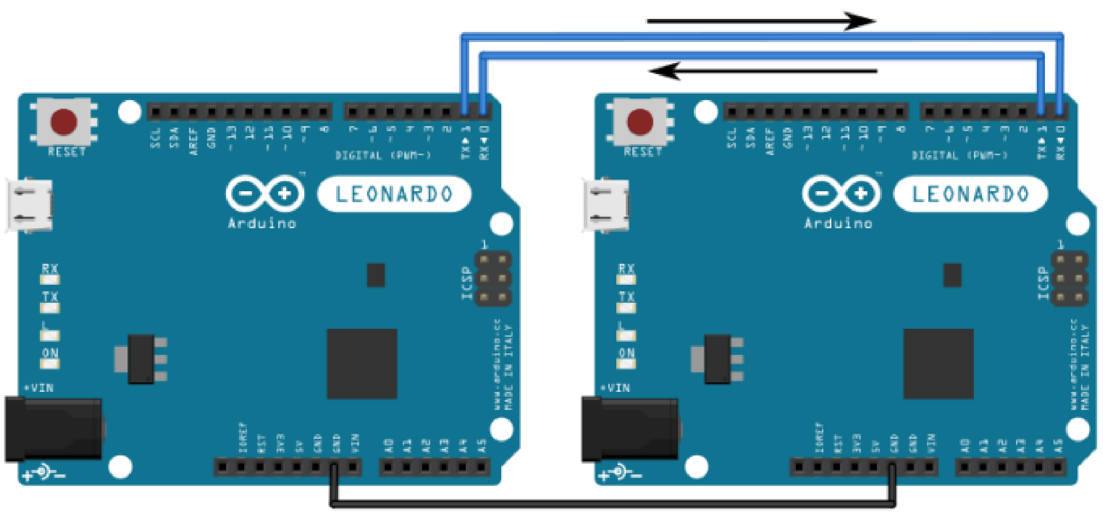
31-07-2023

Introduction:
Smart Door Lock System is about the process of building a secure access control system. It can be implemented using an Arduino Uno microcontroller, a keypad, and an RFID reader. The system will allow authorized personnel to enter a room using a PIN code or an RFID tag. This system is ideal for securing offices, labs, or any other restricted area.

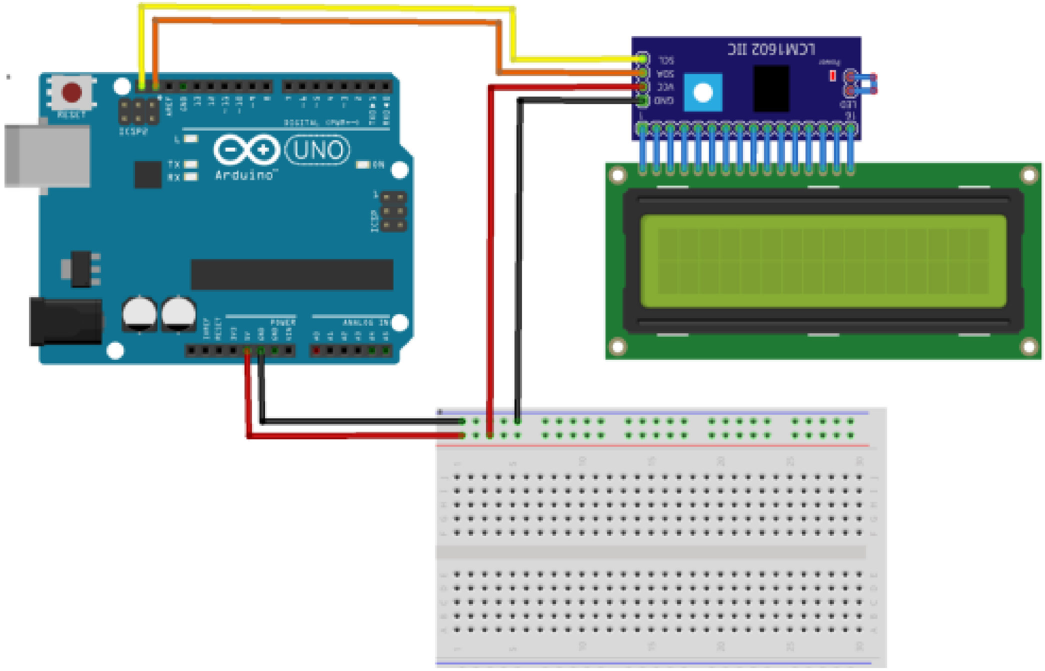
Components:

1.Arduino Uno
2.Keypad
3.RFID Reader
4.Breadboard
5.LED
6.LCD Monitor
7.Solenoid Lock
8.Relay
9.Volt Adaptor

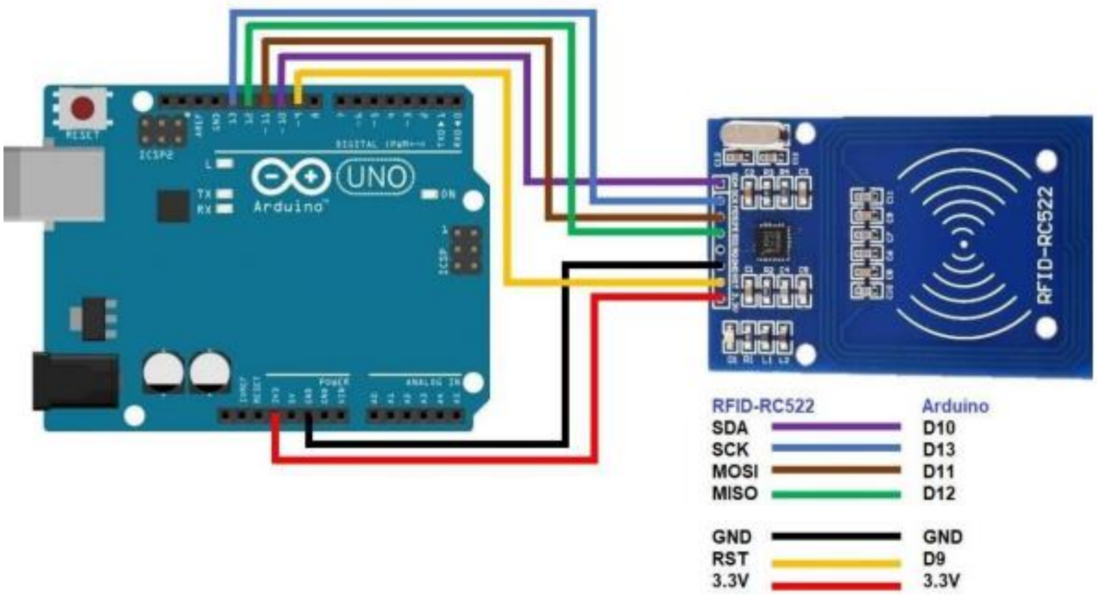
Circuit Diagram:



Master -Slave Connection

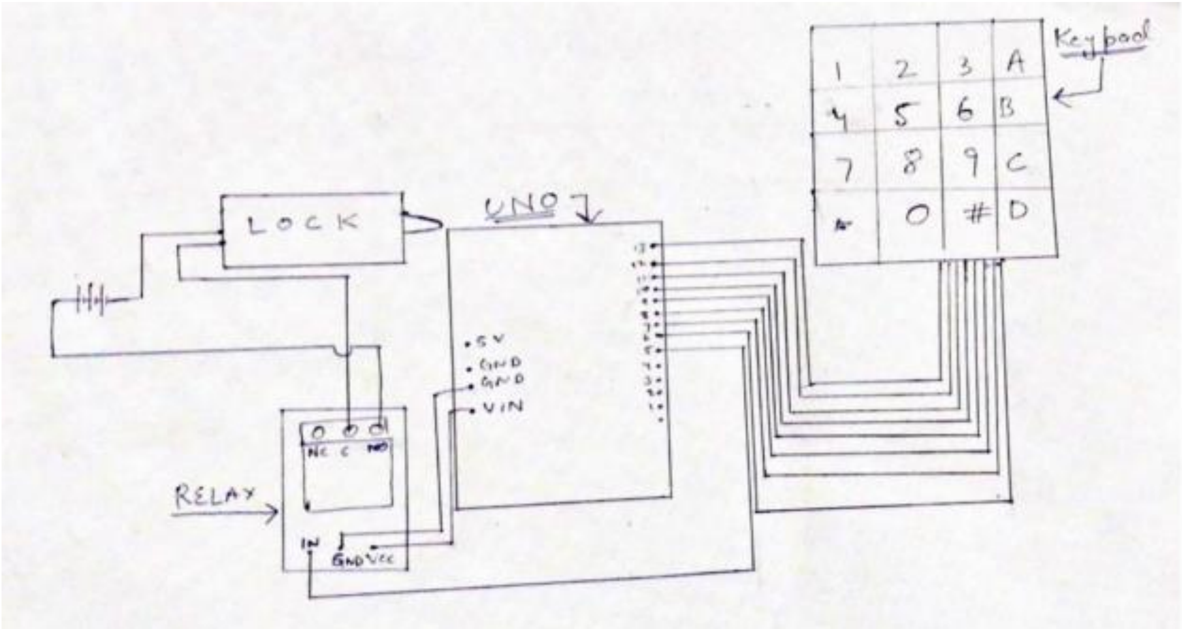


Master Side - LCD Connection



RFID-RC522	Arduino
SDA	D10
SCK	D13
MOSI	D11
MISO	D12
GND	GND
RST	D9
3.3V	3.3V

Master Side - RFID Connection



Slave Side - Keypad Connection

Code Overview:

```
//Master
#include <SPI.h>
#include <MFRC522.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>

SoftwareSerial masterUno(6, 7); // RX, TX

String data;
bool isMatch;

#define SS_PIN 10
#define RST_PIN 9

// Instance of the class MFRC522
MFRC522 rfid(SS_PIN, RST_PIN);

// Create LCD object : Use 0x27 If 0x3F Doesn't work
LiquidCrystal_I2C lcd(0x3F, 16, 2);

// Init array that will store new NUID
byte nuidPICC[4];

// Put Your access NUID Here
byte master[4] = {0x53, 0x9B, 0xB2, 0x0E};

// Pin connected to lock relay signal
#define lockOutput 2

#define greenLED  3
#define redLED 4
#define buzzerPin 5


void setup() {

  Serial.begin(9600);
  masterUno.begin(9600);

  SPI.begin(); // Init SPI bus
  rfid.PCD_Init(); // Init MFRC522

  // Setup LCD with backlight and initialize
  lcd.init();
  lcd.backlight();
  printWelcomeMessage();
  // Set OUTPUT pins
  pinMode(lockOutput, OUTPUT);
  pinMode(greenLED, OUTPUT);
  pinMode(redLED, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
}

void loop() {

  // Reset the loop if no new card present on the sensor/reader
  if ( ! rfid.PICC_IsNewCardPresent())
    return;

  // Verify if the NUID has been readed
  if ( ! rfid.PICC_ReadCardSerial())
    return;

  // Store NUID into nuidPICC array
  for (byte i = 0; i < 4; i++) {
    nuidPICC[i] = rfid.uid.uidByte[i];
  }

  // Checks whether the scanned tag is authorized
  if (getAccessState(master, nuidPICC) == true) {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(" Access Granted!");
    isMatch = 1 ;

    // Send true or false to slave
    masterUno.write(isMatch);

    // Turn on relay & green LED for 5 seconds
    digitalWrite(greenLED, HIGH);
    delay(4000);
    digitalWrite(greenLED, LOW);

    delay(25);  //To fix (LCD, Solenoid) issue
    printWelcomeMessage();

  } else {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(" Access Denied!");
    isMatch = 0 ;

    // Send true or false to slave
    masterUno.write(isMatch);

    digitalWrite(redLED, HIGH);
    digitalWrite(buzzerPin, HIGH);
    delay(1000);
    digitalWrite(redLED, LOW);
    digitalWrite(buzzerPin, LOW);

    printWelcomeMessage();

  }
  // Halt PICC
  rfid.PICC_HaltA();

  // Stop encryption on PCD
  rfid.PCD_StopCrypto1();

}

bool getAccessState(byte accessCode[], byte newCode[]) {
  for (byte i = 0; i < 4; i++) {
    if (newCode[i] != accessCode[i]) {
      return false;
    }
  }
  return true;
}

void printWelcomeMessage() {
  lcd.clear();
  lcd.print("<Access Control>");
  lcd.setCursor(0, 1);
  lcd.print(" Scan Your Tag!");
}
```

```
//slave
#include <Keypad.h>
#include <SoftwareSerial.h>

SoftwareSerial slave(2, 3); // RX, TX

bool isOn;
char * password = "3690"; // To increase the passcode length change the numerical to the size after position

int position = 0;

const byte ROWS = 4; // 4 rows
const byte COLS = 4; // 4 columns


char keys[ROWS][COLS] =
{
{'1','2','3','A'},
{'4','5','6','B'},
{'7','8','9','C'},
{'*','0','#','D'}
}; //mapping of the keys done w.r.t to the grid keypad

byte rowPins[ROWS] = { 13, 12, 11, 10 }; //connection of rows pins to the arduino
byte colPins[COLS] = { 9, 8, 7, 6 }; // connection of the columns pins to the arduino

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

int Lock = 5; // Connecting the relay to the 5th pin

void setup()
{
Serial.begin(9600);
slave.begin(9600);
pinMode(Lock, OUTPUT);
LockedPosition(true);
}

void loop()
{
  // Wait for master to send data
  while (slave.available() == 0)
  {
    char key = keypad.getKey();
if (key == '*' || key == 'A') //OR operator used to lock the device back again
{
position = 0;
LockedPosition(true);
}
if (key == password[position])
{
position ++;
}
if (position == 4) // change this if you want to increase the password length
{
LockedPosition(false);
}
delay(100);
}


  // Read data from master
  isOn = slave.read() ;

  // Execute code based on data
  if (isOn)
  {
    digitalWrite(Lock, HIGH);
  }
  else
  {
    digitalWrite(Lock, LOW);
  }
}



void LockedPosition(int locked)
{
if (locked)
{
digitalWrite(Lock, LOW);
}
else
{
digitalWrite(Lock, HIGH);
}
}
```

Working Principle

Keypad PIN Entry:
User is prompted to enter a predefined PIN using the keypad.
The Arduino reads the pressed keys and assembles the entered PIN.
Entered PIN is compared to the stored PIN for authentication.

RFID Card Detection:
Upon successful PIN entry, the system activates the RFID reader.
The RFID reader scans the nearby area for RFID cards/tags.

RFID Authentication:
If an RFID card is detected, its unique data is read by the RFID reader.
The Arduino compares the read data with the authorized card data.

Servo Motor Activation:
If both PIN and RFID authentication are successful, the Arduino sends a signal to the servo motor.
The servo motor mechanism is triggered, causing the door to unlock.

Access Granted Indication:
An LED indicator may light up to show that access has been granted.
A buzzer can sound to provide an audio confirmation.
Access Denied Scenario:
If either the PIN or RFID authentication fails, the system denies access.
An incorrect PIN entry or mismatched RFID data results in a denied signal.

Security Measures:
The combined use of PIN and RFID provides a two-factor authentication system.
The system enhances security by requiring multiple valid factors for access.

Safety Measures:
In case of incorrect attempts, a time delay or lockout period can be implemented.
The system prevents brute force attacks by limiting repeated attempts.

Emergency Unlocking:
A physical key override can be integrated to unlock the door in case of system failure.
This ensures access even if the electronic components malfunction.

Integration of Components:
The Arduino Uno acts as the central controller, managing keypad, RFID, LED, buzzer, and servo motor.
Components communicate through defined protocols (such as I2C for LCD) to achieve seamless operation.

Feedback Display:
The LCD monitor can display messages indicating the system's status.
Messages like "Enter PIN," "Scanning RFID," "Access Granted," or "Access Denied" can be displayed.

Power Management:
The system is powered by a 12V adaptor, which is regulated by the Arduino to provide required voltages.
Components are powered appropriately to ensure safe and efficient operation.

User Interaction:
The user interacts with the system by entering a PIN and/or presenting an RFID card.

The system responds with visual and auditory feedback, indicating the outcome of the access attempt.

User Interface:
Certainly, here are the user interface aspects of the "Door Lock System: Built with a Keypad PIN and RFID Door Lock Sensor using Arduino Uno" project, presented in points:

1. Keypad Interaction:
- The user approaches the door and is prompted to enter a PIN on the keypad.
- A message on the LCD display instructs the user to input the PIN.

2. PIN Entry:
- The user presses the keys on the keypad to input the PIN.
- The entered PIN is displayed on the LCD screen in real-time.

3. Validation Feedback:
- After entering the PIN, the LCD may display a message like "Press # to Enter" or "Confirm PIN."
- The user confirms by pressing the '#' key.

4. PIN Validation Result:
- If the entered PIN matches the stored PIN, the LCD displays a message like "PIN Correct."
- If the PIN is incorrect, the LCD might show "PIN Incorrect. Retry."

5. RFID Card Scanning:
- Upon successful PIN entry, the system instructs the user to present an RFID card.
- The LCD message could be "Place Card Near Sensor."

6. RFID Card Detection:
- When an RFID card is detected, the LCD might display "Scanning Card..."
- The RFID reader reads the card's data.

7. Access Granted:

- If the scanned RFID card's data matches the authorized cards, the LCD displays "Access Granted."
- An LED could light up to indicate access approval, and a buzzer might provide a pleasant sound.

8. Access Denied:

- If the scanned card is not authorized, the LCD shows "Access Denied."
- The LED might stay off, and a different buzzer sound could indicate denial.

9. System Status Display:

- Throughout the interaction, the LCD displays the system's status messages, like "System Ready," "Waiting for Input," etc.

10. Emergency Unlock Option:

- In case of any system malfunction or emergencies, instructions to use the physical key override might be displayed on the LCD.

11. Visual Feedback:

- LED indicators provide visual cues for access approval or denial.
- LCD messages keep the user informed about the ongoing process.

12. User-Friendly Messages:

- The LCD displays messages in a clear and concise manner to guide the user through each step of the process.

13. Reset or Retry Instructions:

- In case of failed attempts, the LCD could display instructions for retrying after a brief delay or resetting the system.

14. Integration with Mobile App (Future Enhancement):
- As a future enhancement, a mobile app could also be integrated to remotely control and monitor the door lock system.

These points outline how the user interacts with the system through the keypad, RFID sensor, LCD
display, LED indicators, and potential audio feedback. The user interface aims to be intuitive, informative, and responsive throughout the access control process.

Challenges faced during the procedure:

Certainly, here are the challenges faced in the "Door Lock System: Built with a Keypad PIN and RFID Door Lock Sensor using Arduino Uno" project, presented in points:

1. Component Integration:
- Integrating and coordinating multiple components (keypad, RFID reader, servo motor, etc.) required careful wiring and programming to ensure seamless operation.

2. Code Complexity:
- Writing code to handle both keypad PIN entry and RFID card reading while maintaining security and efficiency proved challenging due to the need for synchronization and error handling.

3. Authentication Logic:
- Designing a robust logic for combining PIN and RFID authentication without compromising security was a challenge to prevent unauthorized access.

4. Security Measures:
- Balancing security and user convenience, such as implementing delay mechanisms to deter brute force attacks without causing inconvenience to authorized users.

5. RFID Detection Reliability:
- Ensuring consistent and reliable detection of RFID cards/tags by the reader, especially in scenarios with varying distances and orientations.

6. Physical Mechanism Integration:
- Integrating the servo motor and solenoid lock mechanism into the existing door structure required careful alignment and adaptation.

7. User Experience:
- Designing a user-friendly interface that provided clear feedback during PIN entry and RFID scanning while avoiding confusion or errors.

8. Power Management:
- Ensuring the stability of the system's power supply to prevent issues like unexpected resets or component failures due to power fluctuations.

9. Testing and Debugging:
- Debugging code, addressing unexpected behavior, and fine-tuning the system's responses required thorough testing and troubleshooting.

10. Compatibility and Libraries:
- Ensuring compatibility of libraries used for different components and dealing with potential conflicts or version issues.

11. Physical Constraints:

- Adapting the physical components to fit within the available space, especially when retrofitting the system onto an existing door setup.

12. Emergency Situations:
- Developing a failsafe mechanism or backup solution to handle system failures, ensuring that users can still access the locked area in emergencies.

13. Documentation and Communication:
- Creating clear and comprehensive documentation for future reference and troubleshooting, as well as effectively communicating the project's progress within the team.

14. User Training:
- Ensuring that end-users understand how to correctly operate the system, including PIN entry, RFID presentation, and troubleshooting common issues.

15. Scalability and Maintenance:
- Considering the potential for adding more authorized users/cards and planning for easy maintenance and updates of the system.

Addressing these challenges required a combination of technical skills, problem-solving, and careful planning to create a functional and reliable door lock system.

Potential Applications:

Certainly, here are the potential applications of the "Door Lock System: Built with a Keypad PIN and RFID Door Lock Sensor using Arduino Uno" project, presented in points:

1. Residential Security:

- Enhance home security by implementing a multi-factor door lock system.
- Provide convenient access for authorized family members while restricting unauthorized entry.

2. Office Access Control:

- Improve office security by deploying a robust access control solution.
- Employees can use their unique RFID cards for seamless entry.

3. Commercial Establishments:

- Secure retail shops, restaurants, and small businesses with controlled access.
- Ensure that only authorized personnel can access certain areas.

4. Apartment Complexes:

- Implement a sophisticated access control system for apartment buildings.
- Residents can use RFID cards for easy entry without compromising security.

5. Data Centers:

- Secure critical data centers with an advanced access control mechanism.
- Limit entry to authorized personnel using both PIN and RFID authentication.

6. Laboratories:

- Enhance safety and control in research facilities and labs.
- Researchers can access sensitive areas with personalized RFID cards.

7. Educational Institutions:

- Provide controlled access to school buildings and classrooms.
- Staff and students can gain entry using their assigned RFID cards.

8. Healthcare Facilities:

- Secure medical facilities and restricted areas within hospitals.
- Access can be granted to medical personnel based on RFID authentication.

9. Hotel Rooms:

- Upgrade hotel room security and guest experience.
- RFID cards can replace traditional keys for room access.

10. Shared Workspaces:

- Enable secure entry to coworking spaces and shared offices.
- Members can use RFID cards for hassle-free access.

11. Server Rooms:

- Restrict access to server rooms and IT infrastructure.
- Authorized IT staff can use their RFID cards to ensure system security.

12. Public Facilities:

- Secure public restrooms, storage rooms, and utility areas.
- Limit entry to authorized personnel using the combined authentication system.

13. Parking Garages:

- Manage access to parking areas with RFID-enabled entry gates.
- Users can enter and exit without manual intervention.

14. Gated Communities:

- Control access to gated communities and residential complexes.
- Residents and authorized visitors can use RFID cards for entry.

15. Industrial Sites:

- Secure industrial sites and manufacturing facilities.
- Only authorized personnel with valid RFID cards can enter sensitive zones.

16. Financial Institutions:

- Ensure secure entry to bank vaults and sensitive areas.
- Combine PIN and RFID authentication for stringent security measures.

These points illustrate the diverse range of applications where the "Door Lock System: Built with a Keypad PIN and RFID Door Lock Sensor using Arduino Uno" project can be implemented to enhance security and access control in various environments.

Future Enhancements:

Certainly, here are some future enhancements for the "Door Lock System: Built with a Keypad PIN and RFID Door Lock Sensor using Arduino Uno" project, presented in points:

1. Mobile App Integration:
- Develop a mobile app for remote control and monitoring of the door lock system.
- Enable users to unlock the door using their smartphones, enhancing convenience.

2. Cloud Connectivity:
- Integrate the system with a cloud platform for remote management and data logging.
- Monitor access logs, receive notifications, and manage access permissions online.

3. Biometric Authentication:
- Integrate fingerprint or facial recognition for additional layers of security.

- Enhance access control with biometric data in combination with PIN and RFID.

4. User Profiles and Access Levels:
- Implement user profiles and customizable access levels for different users.
- Administrators can grant different permissions to individuals or groups.

5. Time-Based Access:
- Allow access only during specified time periods for specific users.

- Useful for scenarios where access should be limited to certain hours.

6. Voice Commands:
- Enable voice recognition for controlling the door lock system.
- Users can verbally command the system to lock or unlock.

7. Integration with Smart Home Systems:
- Connect the door lock system to existing smart home setups.
- Enable interaction with voice assistants like Amazon Alexa or Google Assistant.

8. Camera Integration:
- Add a camera module for capturing images or videos of users during access attempts.
- Enhance security by recording visual evidence of who enters the premises.

9. Enhanced User Interface:
- Improve the user interface with a touchscreen display for PIN entry and status updates.
- Provide more detailed feedback and options for users.

10. Energy Efficiency:
- Implement energy-saving features such as automatic locking after a set time of inactivity.
- Integrate motion sensors to detect when the door area is vacant, triggering power-saving mode.

11. Multiple Door Control:
- Extend the system to control multiple doors using a centralized controller.
- Manage access to various areas within a building.

12. Machine Learning for Anomaly Detection:

- Utilize machine learning algorithms to detect unusual access patterns.
- Identify potential security breaches based on historical data.

13. Integration with Security Systems:

- Collaborate with existing security systems like alarms or surveillance cameras.
- Coordinate responses to security events.

14. Enhanced Enclosure Design:

- Redesign the physical enclosure for the components, focusing on aesthetics and durability.
- Ensure the system blends well with its environment.

15. User-Friendly Configuration:

- Simplify the process of adding or removing authorized users and RFID cards.
- Make the setup and maintenance of the system more user-friendly.

These points outline potential future enhancements that can elevate the functionality, security, and user experience of the door lock system beyond its initial implementation.

Conclusion:

This project demonstrates a robust and secure access control mechanism. By combining keypad PIN entry and RFID authentication, along with the power of Arduino Uno, the system ensures reliable and multi-factor access. This innovative solution holds great potential for enhancing security in various applications, from residential spaces to office environments