International Islamic University Chittagong

Department of Computer Science and Engineering

Project Report

“Password Based Smart Door-lock System”

Course Title: Electrical Drives and Instrumentations

Course Code: EEE-2422

Group-D

Team Members:

Tehsim Fariha (C193207),

Raisa Ahmed (C193220),

Efter Jahan Ema (C293229)

Submitted To: Sahariar Reza

Adjunct Faculty,

Dept. of Computer Science and Engineering, IIUC

Date of submission: 31/05/2022

Objective:

The goal of this project is to analyze a suitable collection of components for developing a smart door lock using Arduino Uno that provides excellent security and quick access. In our daily lives, safety is a major concern. Our security pattern includes an access control system for doors. Traditional locks are no longer as secure as they once were. We need to create a system that will assist 24/7. Only authorized individuals will have the access. Arduino is in charge of the entire system. This password-based bolt structure will provide clients with a more secure and low-effort locking-opening mechanism. Mechanical door locks will be replaced by electronic door locks in the future.

The following are the specific project goals:

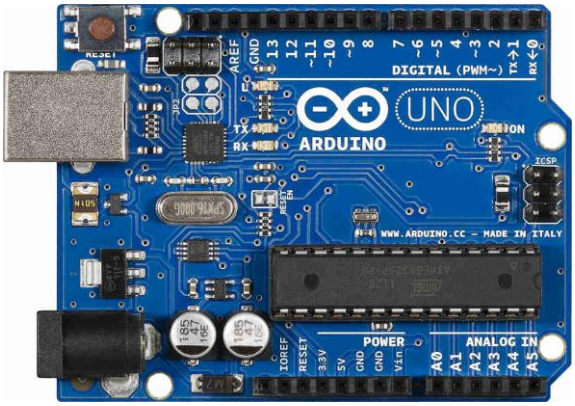
- Familiarity with a smart door lock system based on a microcontroller
- Using Arduino Uno to create a simple and smart door locking system

Necessary Components:

- ## *Hardware:*

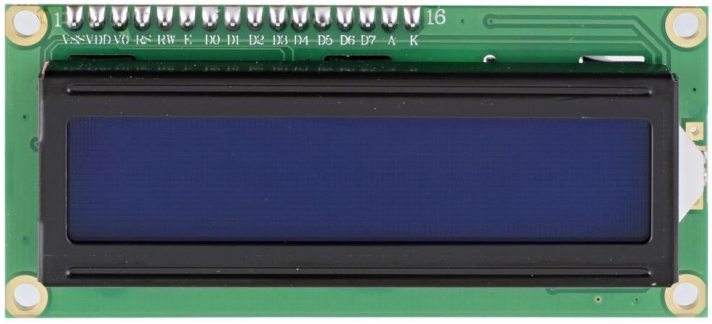
1) Arduino Uno

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to asa microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.



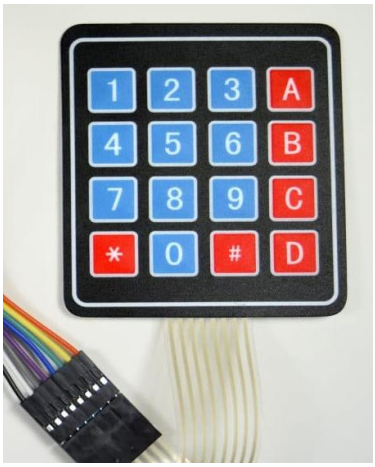
2) 16×2 LCD display

An LCD (Liquid Crystal Display) screen is an electronic display module and has a wide range of applications. A 16x2LCD is a very basic module and is very commonly used in various devices and circuits. A 16x2 LCD means it can display 16 characters per line and there are 2 such lines.



3) 4×4 Matrix Keypad

This 4x4 matrix keypad has 16 built-in push button contacts connected to row and column lines. A microcontroller can scan these lines for a button-pressed state. In the keypad library, the Propeller sets all the column lines to input and all the row lines to input.

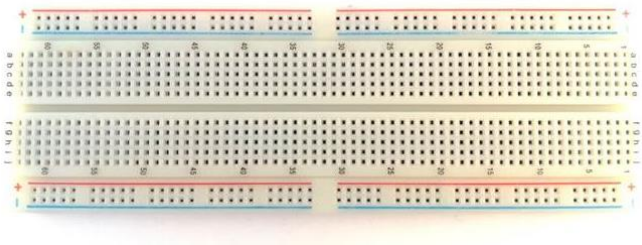


4) Servo Motor

This A servo motor is a rotational or translational motor that receives power from a servo amplifier and is used to impart torque or force to a mechanical device like an actuator or a brake. Servo motors provide exact angular position, acceleration, and velocity control. A closed-loop control system is used with this type of motor.

5) Breadboard

A thin plastic board used to hold electronic components (transistors, resistors, chips, etc.) that are wired together. Used to develop prototypes of electronic circuits, breadboards can be reused for future jobs. They can be used to create one-of-a-kind systems but rarely become commercial products.



6) Jumper wires

A jump wire is an electrical wire, or group of them in a cable, with a connector or pin at each end which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

- ***Software:***
  1) ***Proteus***
  2) ***Arduino IDE***

Experimental Circuit Setup:

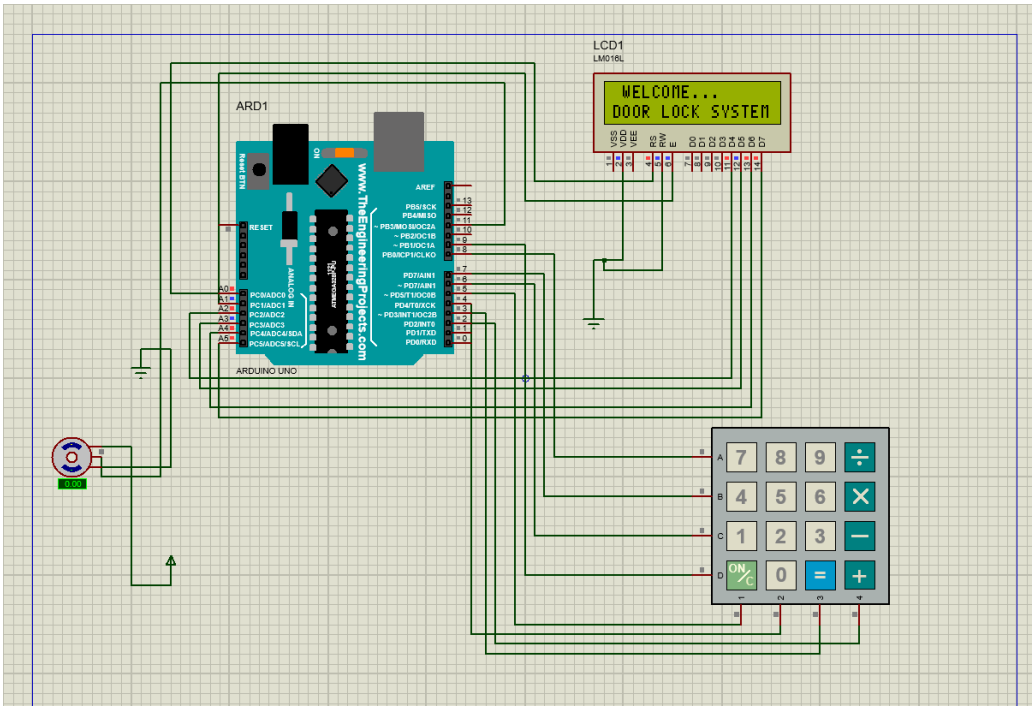


Fig: Circuit Diagram (Simulation)

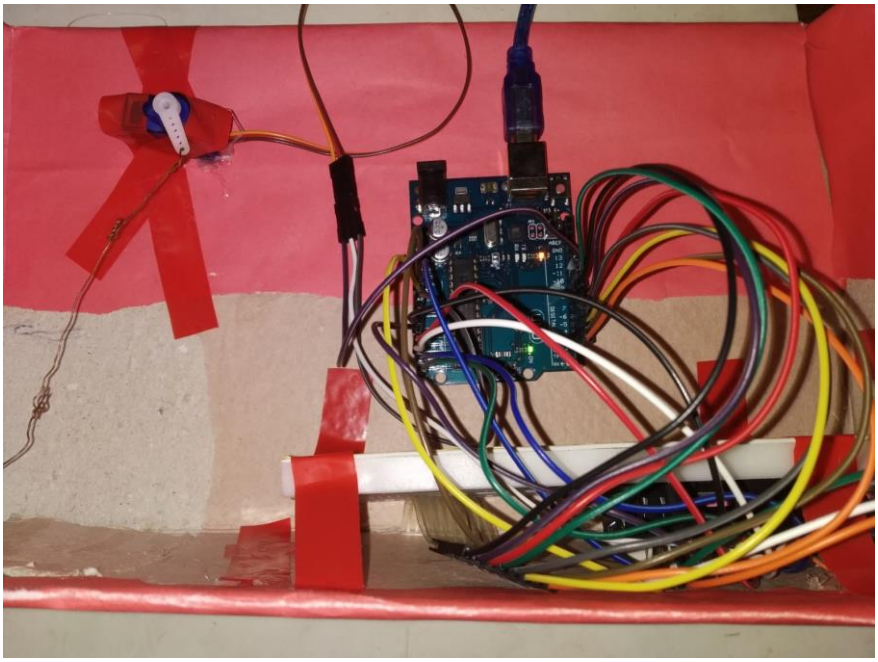


Fig: Circuit Setup

The connections to be made are given below:

*Connection of the **16×2 LCD** to the **Arduino**:

1) Connect pin 2 (VDD) and pin 5 (RW) to the ground.

2) Connect pin 4 (RS) to pin (A0) of Arduino.

3) Connect pin 6 (E) to the pin (A1) of the Arduino.

The following four pins are data pins which are used to send data or commands to the LCD:

4) Connect pin 11 (D4) to pin (A2) of Arduino.

5) Connect pin 12 (D5) to pin (A3) of Arduino.

6) Connect pin 13 (D6) to pin (A4) of Arduino.

7) Connect pin 14 (D7) to pin (A5) of Arduino.

*Connection of the **4×4 matrix keypad** to the **Arduino**:

8) Connect Row pins of keypad A, B, C, D to pins 8, 7, 6, 9 of Arduino respectively.

9) Connect Column pins of keypad 1, 2, 3, 4 to pins 5, 4, 3, 2 of Arduino respectively.

*Connection of the **Servo motor** to the **Arduino**:

10) Connect Red wire to Power +5v.

11) Connect Orange wire to pin 11 of Arduino.

12) Connect Brown wire to Ground.

Arduino Code:

```
#include<Keypad.h>
#include<LiquidCrystal.h>
#include<EEPROM.h>
#include <Servo.h>
Servo myservo;
int pos = 0;

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);
char password[100];
char pass[100], pass1[100];
int i = 0;
char customKey = 0;
const byte ROWS = 4;
const byte COLS = 4;
char hexaKeys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[ROWS] = {8, 7, 6, 9};
byte colPins[COLS] = {5, 4, 3, 2};

Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

void setup()
{
```

```
  lcd.begin(16, 2);
  lcd.print(" WELCOME... ");
  lcd.setCursor(0, 1);
  lcd.print("DOOR LOCK SYSTEM");
  delay(2000);
  lcd.clear();
  lcd.print("Enter password:");
  lcd.setCursor(0, 1);
  for (int j = 0; j < 4; j++)
    EEPROM.write(j, j + 49);
  for (int j = 0; j < 4; j++)
    pass[j] = EEPROM.read(j);
  myservo.attach(11);
}

void loop()
{
  customKey = customKeypad.getKey();
  if (customKey == '#')
  {
    change();
  }
  if (customKey == '0')
  {
    lock();
  }
  if (customKey)
  {
```

```
      if (customKey == '*')
      {
        password[i--] = customKey;
        lcd.setCursor(i, 1);
        customKey = customKeypad.getKey();
       //if(customKey=='1' || customKey=='2' || customKey=='3' || customKey=='4' ) {
lcd.print(customKey);    }
         if(customKey)
         {
          lcd.print(customKey);
         }
      }
      else
      {
         password[i++] = customKey;
         lcd.print(customKey);
      }
  }
  if (i == 4)
  {
    delay(200);
    for (int j = 0; j < 4; j++)
      pass[j] = EEPROM.read(j);
    if (!(strncmp(password, pass, 4)))
    {
      lcd.clear();
      lcd.print("Passkey Correct");
      delay(300);
```

```
  for (pos = 0; pos <= 180; pos += 1) {


    myservo.write(pos);
    delay(15);
  }
  lcd.clear();
  lcd.setCursor(0, 1);
  lcd.print("PRESS # TO CHANGE PASSWORD.");
  for (int PositionCount = 0; PositionCount < 70; PositionCount++)
  {
    lcd.scrollDisplayLeft();
    delay(60);
  }
  delay(500);
  lcd.clear();
  lcd.print("Enter Passkey:");
  lcd.setCursor(0, 1);
  i = 0;
}
else
{
  lcd.clear();
  lcd.print("Wrong Passkey...");
  lcd.setCursor(0, 1);
  lcd.print("=.Change Passkey");
  delay(1300);
  lcd.clear();
  lcd.print("Enter right key:");
```

```
      lcd.setCursor(0, 1);
      i = 0;
    }
  }
}
void change()
{
  int j = 0;
  lcd.clear();
  lcd.print("Enter Current Key");
  lcd.setCursor(0, 1);
  while (j < 4)
  {
    char key = customKeypad.getKey();
    if (key)
    {
      if (key == '*')
      {
        pass1[j--] = key;
        lcd.setCursor(j, 1);
        key = customKeypad.getKey();
       // if(key=='1' || key=='2' || key=='3' || key=='4' ) {  lcd.print(key);    }
        if(key)
        {
          lcd.print(key);
        }
      }
      else
```

```
    {
      pass1[j++] = key;
      lcd.print(key);
    }
  }
  key = 0;
}
delay(500);
if ((strncmp(pass1, pass, 4)))
{
  lcd.clear();
  lcd.print("Wrong Passkey...");
  lcd.setCursor(0, 1);
  lcd.print("Enter Right Key");
  delay(1000);
}
else
{
  j = 0;
  lcd.clear();
  lcd.print("Enter New Key:");
  lcd.setCursor(0, 1);
  while (j < 4)
  {
    char key = customKeypad.getKey();
    if (key)
    {
      if (key == '*')
```

```
        {
          pass[j--] = key;
          lcd.setCursor(j, 1);
          key = customKeypad.getKey();
          //if(key=='1' || key=='2' || key=='3' || key=='4' ) { lcd.print(customKey);  }
          if(key)
          {
          lcd.print(key);
          }
        }
        else
        {
         pass[j] = key;
         lcd.print(key);
         EEPROM.write(j, key);
         j++;
        }
      }
    }
    lcd.print(" Done...");
    delay(1000);
  }
  lcd.clear();
  lcd.print("Enter your key:");
  lcd.setCursor(0, 1);
  customKey = 0;
}
void lock()
```

```
{
    lcd.clear();
    lcd.print("Lock the door");
    lcd.setCursor(0, 1);
    delay(300);

    for (pos = 90; pos>=0; pos -= 1)
   {
     myservo.write(pos);
     delay(15);
    }
    lcd.clear();
    lcd.setCursor(0, 1);
    customKey = 0;
    setup();
}
```

Observation:

After all wiring connections are done, we need to upload code in Arduino and supply power (In this case, we connected with our laptop). When the system is powered ON, LCD display shows "Welcome…Door lock system" and then "Enter Password:"

For our project, the password is "1234". When we press these numbers on keypad, the numbers are perfectly visible on LCD. Then it says "Passkey Correct" and opens the door for us as expected. If we need to change the passkey, we have to press "#" button on keypad. As a result, passkey was changed successfully.

If we press any wrong button on the keypad by mistake, we can also delete it by pressing "*" button on the keypad.

Then to lock the door we need to press 0 on keypad. The door gets locked accordingly.

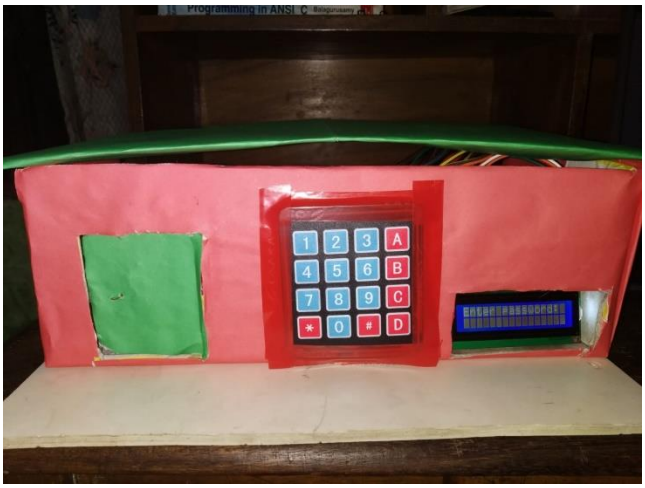


Fig: Complete Project

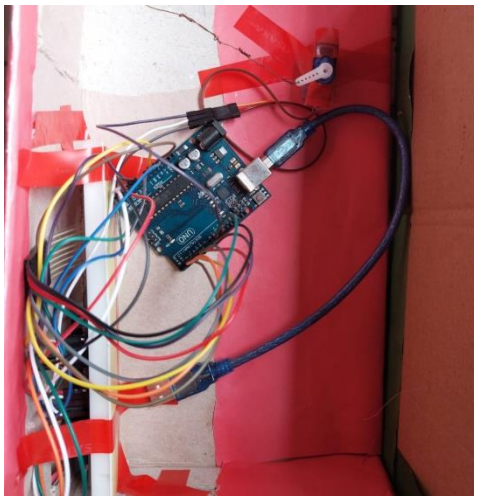


Fig: Connections

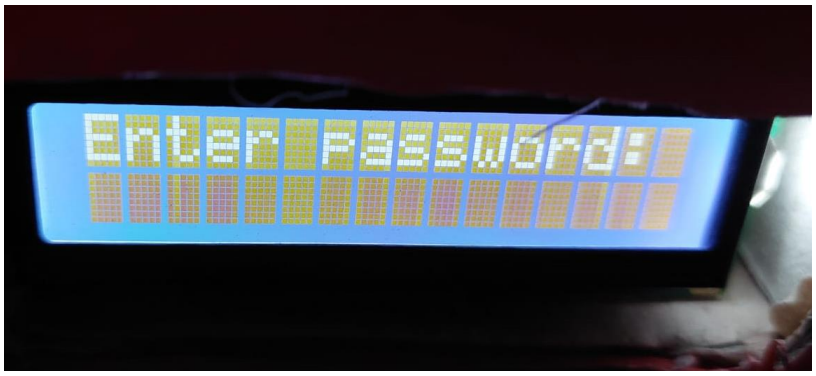


Fig: Enter Password (LCD display)

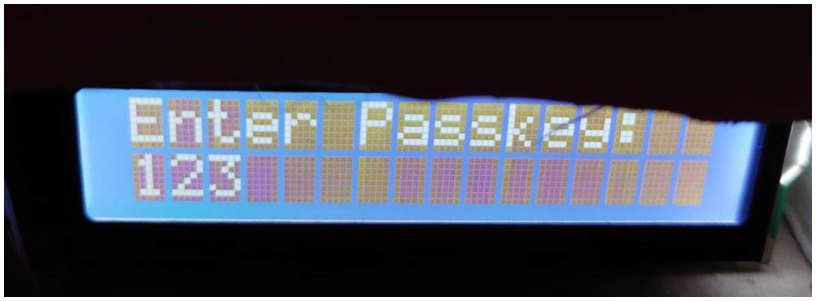


Fig: Entering Passkey

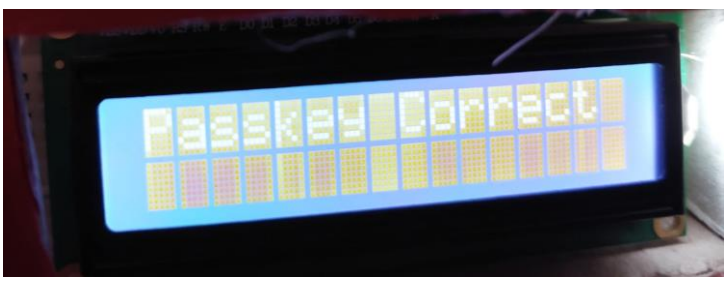


Fig: Passkey Correct

Fig: Lock the door

Fig: Enter New key (Changing Passkey)

Discussion:

After proper observation, we were successful to achieve the desired outcome. While making the connections and observing, we faced small difficulties (eg. Faulty wires, LCD display problem) but we were able to overcome these problems. So, the proposed system allows remote access to lock or unlock the door without physical user interaction. The system fulfills the requirements of supporting autonomous locking device and easy key distribution compared to physical keys. The system has minimum requirements for hardware and supports customization of keys. The keypad is used for taking the password. Servo motor is used for opening the gate while rotating and LCD is used for displaying status or messages on it. Finally, we can say that this digital door lock system can secure home or locker easily. Thus, the system proposed is feasible.