



Patuakhali Science and Technology University

Project Name

Password Security Lock System Using Arduino &
Keypad

Course Code: EEE-212

Course Title: Electrical Technology Sessional

Submitted To

Md. Naimur Rahman
Assistant Professor,
Department of Electrical and
Electronics Engineering
Patuakhali Science and Technology
University

Submitted By

Md. Mahdi Hasan ID No. 1802035 Reg. No. 08445 Level: 02 Semester: 01 Session: 2018-19	Mehedi Hasan Rabbi ID No. 1802052 Reg. No. 08462 Level: 02 Semester: 01 Session: 2018-19
--	---

Submission Date: 21 December, 2021

Abstract

Security is a main concern in our everyday life. Each and every individual needs to feel secure. An access control for doors forms an essential part in our security pattern. Doors locked using conventional locks are not as safe as they used to be, anyone can break in by breaking these locks. We want to make a framework that will give 24/7 benefit. Password based door lock system allows only approved persons to access restricted areas. This system is fully controlled by Arduino. The password can be entered via a keypad. If the password is matched with the stored password in Arduino the door gets open. This programmed password-based bolt framework will give client more secure and minimal effort method for locking-opening framework. The security door lock automation system promises a bold step to the future where mechanical door locks will be substituted by electronic doorlocks.

Introduction

Password based door lock system provides security for homes through a security password which is confidential for the user alone. The user will need to enter a password to unlock the door. On successful password entry, the door gets open for a specific amount of time letting the individual to enter into the house. On the other hand, if the user enters a wrong password the door does not get opened.

Components Required:

Before building our password door locking project, first, we need to collect the required components and then go ahead and follow the step-by-step building process.

List of components:

S.N.	Components Name	Description	Quantity
1	Arduino Board	Arduino UNO R3 Development Board	1
2	Keypad	4X4 Keypad	1
3	LCD Display	JHD162A 16x2 LCD Display	1
4	Potentiometer	10K	1
5	Servo Motor	SG90 ServoMotor	1
6	Buzzer	5V Active Buzzer	1
7	Connecting Wires	Jumper Wires	>=20
8	Breadboard	-	1

COMPONENT DESCRIPTION

Arduino Uno Overview

The Arduino Uno is a microcontroller board based on the ATmega328. It has 20 digital input/output pins (of which 6 can be used as PWM outputs and 6 can be used as analog inputs), a 16 MHz resonator, a USB connection, a power jack, an in-circuit system programming (ICSP) header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use

the FTDI USB-to-serial driver chip. Instead, it features an ATmega16U2 programmed as a USB-to-serial converter. This auxiliary microcontroller has its own USB bootloader, which allows advanced users to reprogram it.



4 X 4 Keypad

The 4 x 4 matrix keypad usually is used as input in a project. It has 16 keys in total, which means the same input values. The 4 x 4 Matrix Keypad Module is a non-encoded matrix keypad consisting of 16 keys in parallel. The keys of each row and column are connected through the pins outside – pin R1-R4 as labeled beside control the rows, when L1-L4, the columns.

A 4x4 matrix keypad consisting of microswitch buttons. The module has four holes 3mm (M3) holes for mounting. The pin designations on each PCB are shown on each PCB. Four pins are thus used as an 'x' coordinate and the other 4 as a 'y' coordinate.

The module is already equipped with a soldered pin-headers.



LCD 16 X 2 Keypad

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. The 16 x 2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. This LCD has two registers, namely, Command and Data.

Command registers stores various commands given to the display.



Servo Motor

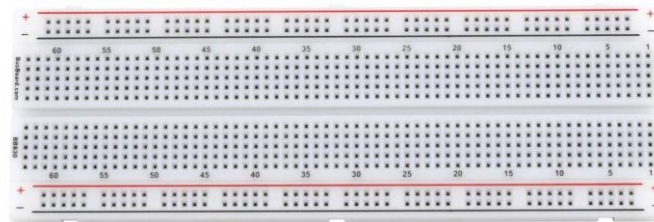
A servo motor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity, and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. Servomotors are used in applications such as robotics, CNC machinery, or automated manufacturing.



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.

The breadboard contains spring clip contacts typically arranged in matrices with certain blocks of clips already wired together. The components and jump wires (assorted wire lengths with pins at both ends) are plugged into the clips to create the circuit patterns. The boards also typically include metal strips along the side that are used for common power rails and signal buses.



Jumper Wires

A jump wire (also known as jumper, jumper wire, jumper cable, DuPont wire or cable) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), 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.

Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.



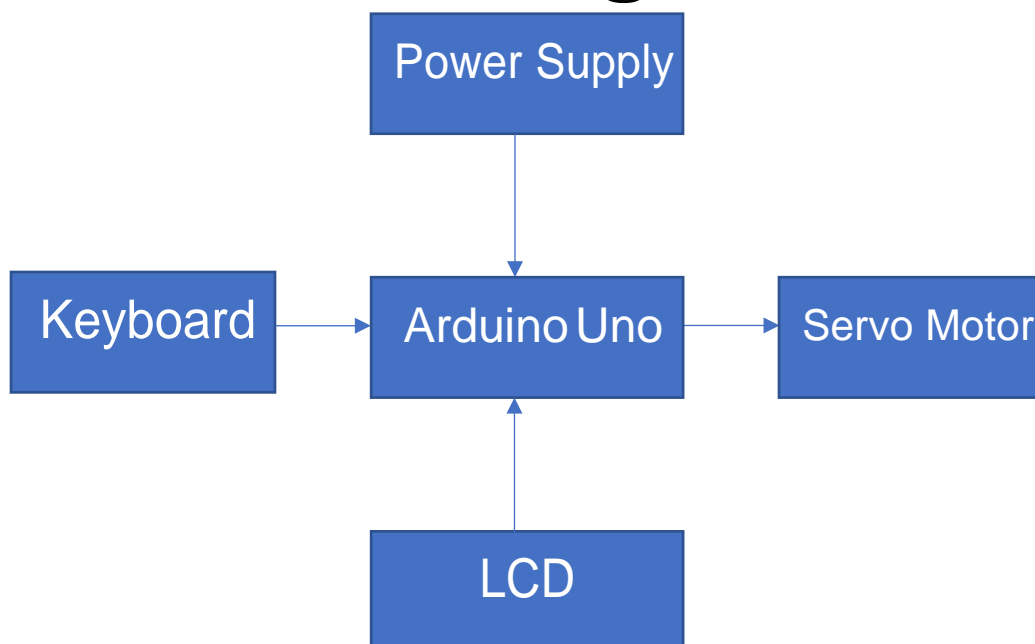
Working

The user can enter a password which is suitable to their needs. The entered password during installation will be saved as SET PASSWORD. During initial stage the system will get locked. Again, when the user enters a password, the entered password is checked with the predefined set password. If the password matches then the servo motor deflects and the door gets unlocked. Whenever the individual wants to enter into their houses, he or she has to enter the correct password. If the password is correct or not the Arduino will check in its database. If the password entered matches with the set password, the LCD will display unlocked message on its screen and the door gets unlocked allowing the individual to enter.

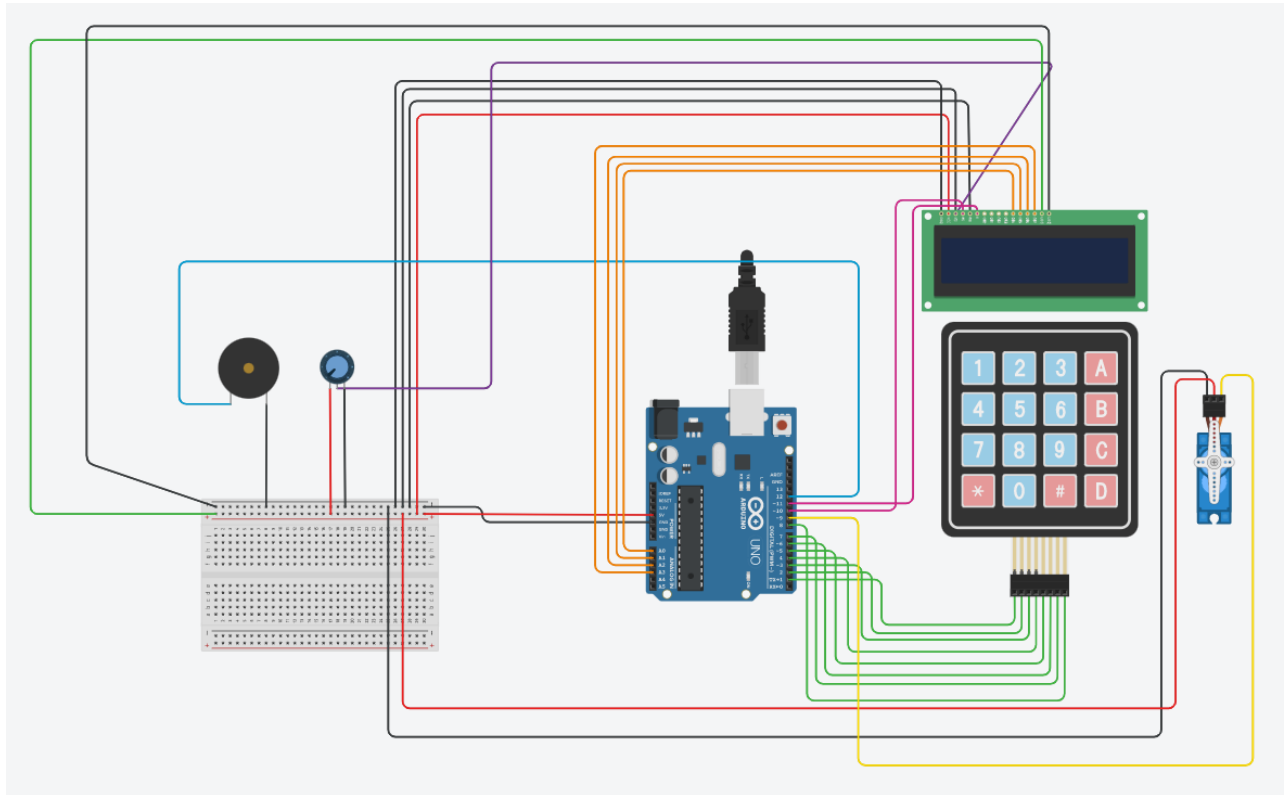
Steps of procedure:

1. At first there will be a welcome message “Arduino Door Lock Project”. Then it will ask to enter the password.
2. If password is correct then the door will open. Servo motor will rotate 180 degrees.
3. If the password is wrong then the buzzer will sound for couple of second and print “Wrong Password”. After couple of second system will ask for password printing “Enter password”.
4. When the door is open there will be a message printing “Door is open”.
5. Now when we press “#” key on the keypad then the servo motor will come to its initial position. And the door will the close.

Block Diagram



Circuit Diagram



Connection between the 16×2 LCD to the Arduino.

- LCD RS pin to digital pin 12
- LCD Enable pin to digital pin 11
- LCD D4 pin to digital pin A0
- LCD D5 pin to digital pin A1
- LCD D6 pin to digital pin A3
- LCD D7 pin to digital pin A4
- LCD R/W pin to GND
- LCD VSS pin to GND
- LCD VCC pin to 5V
- LCD LED+ to 5V
- LCD LED- to GND

Connection between the 4X4 keypad to the Arduino.

- Row 1 to digital pin 1
- Row 2 to digital pin 2
- Row 3 to digital pin 3
- Row 4 to digital pin 4
- Column 1 to digital pin 5
- Column 2 to digital pin 6
- Column 3 to digital pin 7
- Column 4 to digital pin 8

Connection between the servo motor to the Arduino.

- Signal pin to digital pin 9
- Servo motor positive pin to +5V
- Servo motor negative pin to GND

Connection between buzzer to the Arduino.

- Positive pin to digital 12
- Negative pin to GND

Connection between Potentiometer to the Arduino.

- Wiper pin to LCD VO pin
- Terminal 1 pin to +5V
- Terminal 2 pin to GND

Source Code/Program

The code for Password Based Door Lock Security System Using Arduino & Keypad is given below. You can simply copy & paste the code using Arduino IDE. The simply compile it then finally upload it to the ATmega328 microcontroller.

```
#include <Keypad.h>

#include <LiquidCrystal.h>

#include <Servo.h>

Servo myservo;

LiquidCrystal lcd(10, 11, A0, A1, A2, A3);

#define Password_Lenght 7

int pos = 0;

char Data[Password_Lenght];

char Master[Password_Lenght] = "ABCD79";

byte data_count = 0, master_count = 0;

bool Pass_is_good;

char customKey;

int buzz=12;

const byte ROWS = 4;

const byte COLS = 4;

char keys[ROWS][COLS] = {

    {'1', '2', '3', 'A'},

    {'4', '5', '6', 'B'},

    {'7', '8', '9', 'C'},

    {'*', '0', '#', 'D'}

};
```

```
bool door = true;

byte rowPins[ROWS] = {1, 2, 3, 4};

byte colPins[COLS] = {5, 6, 7, 8};

Keypad customKeypad( makeKeymap(keys), rowPins, colPins,
ROWS, COLS);

void setup()
{
  pinMode(buzz, OUTPUT);
  myservo.attach(9);
  ServoClose();
  lcd.begin(16, 2);
  lcd.print("Arduino Door");
  lcd.setCursor(0, 1);
  lcd.print("--Look project--");
  delay(3000);
  lcd.clear();
}

void loop()
{
  if (door == 0)
  {
    customKey = customKeypad.getKey();
```

```
    if (customKey == '#')
    {
        lcd.clear();
        ServoClose();
        lcd.print(" Door is close");
        delay(3000);
        door = 1;
    }
}
else Open();
}
void clearData()
{
    while (data_count != 0)
    { // This can be used for any array size,
        Data[data_count--] = 0;
    }
    return;
}
void ServoOpen()
{
```

```
for (pos = 180; pos >= 0; pos -= 5) {  
    // in steps of 1 degree  
    myservo.write(pos);  
    delay(15);  
}  
}  
  
void ServoClose()  
{  
    for (pos = 0; pos <= 180; pos += 5) {  
        myservo.write(pos);  
        delay(15);  
    }  
}  
  
void Open()  
{  
    lcd.setCursor(0, 0);  
    lcd.print(" Enter Password");  
    customKey = customKeypad.getKey();  
    if (customKey)  
    {  
        Data[data_count] = customKey;
```

```
    lcd.setCursor(data_count, 1);  
    lcd.print(Data[data_count]);  
    digitalWrite(buzz, HIGH);  
    delay(100);  
    digitalWrite(buzz, LOW);  
    data_count++;  
}  
if (data_count == Password_Lenght - 1)  
{  
    if (!strcmp(Data, Master))  
    {  
        lcd.clear();  
        ServoOpen();  
        lcd.print(" Door is Open");  
        door = 0;  
    }  
    else  
    {  
        lcd.clear();  
        lcd.print(" Wrong Password");  
        digitalWrite(buzz, HIGH);
```

```
delay(10000);  
digitalWrite(buzz, LOW);  
door = 1;  
}  
clearData();  
}  
}
```

Conclusion

The main purpose is to design a security system which is beneficial to each and every individual. We designed a security system using Arduino. The system we designed is a success and provides security effectively. The system is cheap and affordable to everyone.

ACKNOWLEDGEMENTS

This work was supported by Department of Electrical and Electronic Engineering, Patuakhali Science and Technology University.