JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

"Password-Based Smart Door Locking System"

A Report Submitted to the ***Department of Electronics and communication Engineering*** *of Lakshmi Narain College Of Technology , Bhopal , M.P., India*

Research Scholar-

Name: Chitransh Sharma
0103EC201051
Department: ECE

Name: Devendra Yede
0103EC201058
Department: ECE

Name: Divyanshu
0103EC201062
Department: ECE

Name: Devansh Sen
0103EC201057
Department: ECE

Professor – Sunny Jain

Abstract — **Physical keys are the most-used and well-known technology to lock or unlock a door. People are familiar with it. Despite the physical key locking system being well-proven, it is not without liabilities. Roaming outside of the home with physical keys is inconvenient and there is a risk of losing physical keys. Smart-locking system is a keyless technology that lets a person unlock without having to use a physical key. A smart lock is an electromechanical lock that is designed to perform locking and unlocking operations on a door when it receives such instructions from an authorized device and executes the authorization process using a cryptographic key.**

Keywords — Door Lock, Smart Lock, Smart Home, Security, Arduino

I. INTRODUCTION

A. *Background of Study and Motivation*

We cannot think of a day without security. It is a major concern of a part of our daily lives. Everybody should have a sense of safety. The security pattern of this project is an access control system to lock or unlock a door. Conventional locks are not as safe as they used to be, anyone can access them by breaking these locks. We need to build a system that will provide 24/7 benefits and help to make security. The password-based door locking system allows only approved persons to access restricted areas. The system is fully controlled by Arduino UNO. Passwords can be entered through a keypad. The door gets unlocked when the password is matched that is stored in the Arduino chip. The sketched password-based bolt structure will provide clients with more security and less effort in locking-unlocking mechanisms. [1]

B. *Project Objectives*

The goal of the project is to find and assess a suitable set of components for creating a password-based smart door locking system with Arduino UNO that provides high security and quick access. The following are the specific goals:

- Familiarity with a smart door locking system based on a microcontroller (Arduino).
- Using the components to build a simple and smart door locking system.

II. LITERATURE REVIEW

In this current system of the project for door locking, there are a few points of view. A traditional door and key locking system that integrates the latest smart locking technology. Modern existence is widely reliant on technical improvements such as the ability to open doors and control modern gadgets and devices. People want their appliances to make them feel comfortable and protected. People's requirements are the key reason for the creation of smart locks. Some of these systems are discussed in this section.

- *Internet of Things*

Smart buildings have recently become the foundation for the Internet of Things (IoT). Connecting devices in homes to the internet increases internet usage, making homes more comfortable, provident, pleasant, and secure. The proposed approach focuses on a key security feature of smart home technologies: the door lock mechanism. The door lock system sets up security by allowing the owner to check the buildings using an Arduino UNO-controlled, Bluetooth-connected system. Installing the built android application on devices such as tablets, smartphones, laptops, and other computers allows users to open or close the door lock by entering login credentials such as username and password, which are confirmed in a database over the internet. If the credentials are incorrect, the buzzer sounds and an SMS notice is issued to the building's owner, increasing security. Using other wireless connectivity, this concept can be extended out to commercial sectors such as ATMs, vending machines, and so on. [2]

- *Fingerprint Locking System*

Only individuals whose fingerprints are pre-stored in the memory can use fingerprint recognition technology. Even in case of a complete power outage or battery loss, stored fingerprints are kept. This eliminates the need to keep track of keys or memorize a password or PIN combination. Because there are no keys or combinations that can be copied or stolen, or locks that can be picked, it can only be accessed when an authorized person is present. As a result, the fingerprint-based lock is an excellent solution to commonly encountered problems. [3]

- *Knock-Pattern Using Arduino and GSM Communication*

This technique uses a 'Secret Knocking Pattern,' which is only known by the owner of the safe, luggage, or other object or item on which the device is installed. The knocking pattern must be applied only at a certain spot known only to the owner for the lock to open. Only when the secret knock has been unlocked can the secret pattern be changed. Because there is no key to copy, this method completely removes the possibility of duplication. [4]

- *Keyless Entry System Based on Arduino Board with Wi-Fi Technology*

An Arduino circuit board, a Wi-Fi module, and the PHP programming language are used to allow access to a closed door in this keyless entry system. The proposed method is explained, which involves using an Arduino Uno board and a Wi-Fi shield to unlock the door without a key. Unlike earlier systems, which have a restricted range, the internet connection allows the device to unlock the door from anywhere. [5]

- *RFID Based Access Control System*

In the recommended system, a magnetic door lock is controlled by an RFID reader, which starts the authentication and validation of the user or, in other words, regulates access. Furthermore, the systems keep track of each user's access and exit records via a log report for each access. To avoid unforeseen occurrences, the administrator of the central subsystem can revoke any user's validity at any time. [6]

III. Methodology & Modeling

A. Introduction

In this project, we used an Arduino and a keypad to create a password-based security system. Thefts and frauds are becoming more common by the day, therefore security is becoming a serious worry. As a result, a smart lock with a digital code can simply secure our home, business, locker, and other valuables. It only unlocks a door when the correct password is supplied. Due to a password-based door lock mechanism, only authorized personnel are permitted access to the restricted areas. The Arduino is in charge of the project's overall operation. A 4×4 keypad can be used to input the necessary password.

B. Working Principle

There are two cases for this experiment. The purpose of this experiment is to implement a door-locking mechanism that opens or closes the lock on the door automatically with password.

Case 1: The lock will open and close

When a password is entered via keypad, the system checks the password and finds out if it is right or wrong. If the password matches with the stored password in the microcontroller chip, the microcontroller sends the signal to the LCD display for showing "The door is open" as well as the microcontroller sends the signal to Servo Motor. Then the motor is rotated by 0° to 180° and opens the lock, allowing the door to be unlocked. Later, by pressing '#' button, the Servo motor is rotated again by 180° to 0° closing the door.

Case 2: The lock will not open

If the wrong password is entered, the system shows "Password doesn't match – Please try again" and Servo Motor is not rotated. A bit of time the system automatically starts again from the beginning.

C. *Important Components*

- Hardware:

1. Arduino UNO
2. 4×4 Matrix Keypad
3. SG90 Micro Servo Motor
4. 16×2 LCD Display
5. Potentiometer

- Software:

1. Arduino IDE
2. WOKWI Web Application

Arduino UNO:

Arduino UNO is an open source microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an 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 an AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

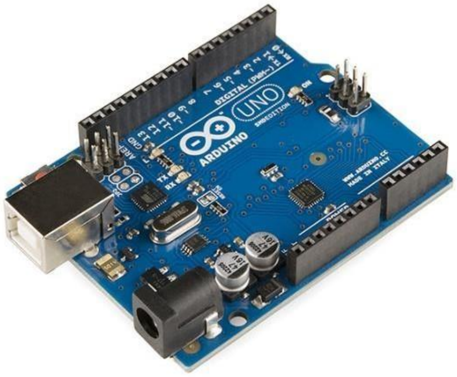

Figure 1: Arduino UNO

The Arduino platform has become quite popular with people just starting with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) to load new code onto the board -- you can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Finally, Arduino provides a standard form factor that breaks out the functions of the micro-controller into a more accessible package. [7][8]

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. Then, it picks a row and sets it high. After that, it checks the column lines one at a time. If the column connection stays low, the button on the row has not been pressed. If it goes high, the microcontroller knows which row (the one it set high), and which column, (the one that was detected high when checked). See the schematic in the "Circuit" section, above, for a visual reference of the keypad layout.

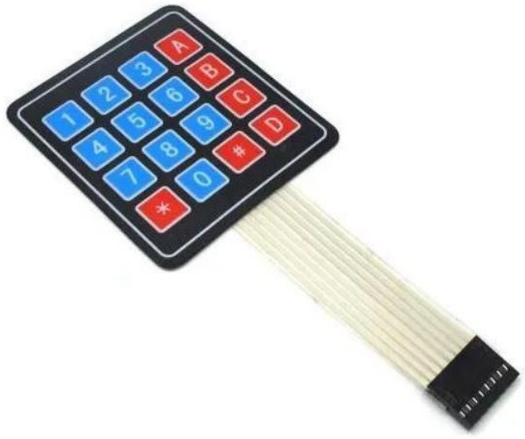

Figure 2: 4×4 Matrix Keypad

The keypad library supports pretty much any number of rows and columns. So, the program has to tell us that our keypad has 4 rows and 4 columns, which I/O pins the lines are connected to, and what value each button represents. The rows, cols, and values arrays store that information. The rows array will be used to tell the keypad library that the top row is connected to P7, the second row to P6 and so on. Likewise, the cols array lists the leftmost column as connected to P3, the next over connected to P2 and so on.

The values array stores the value we want the program to give us for each button press. For example, if the top-left button is pressed, we want the number 1, and if the next one is pressed, we want the number two. If the top right button is pressed, we want the ASCII code for the 'A' character, which is 65. [9]

SG90 Micro Servo Motor:

Micro Servo Motor SG90 is a tiny and lightweight servo motor with high output power. Servo can rotate approximately 180 degrees, and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small

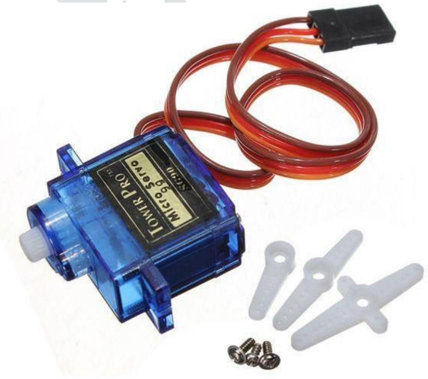

Figure 3: SG90 Micro Servo Motor

places. It comes with 3 horns (arms) and hardware. [10]

16×2 LCD Display:

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc. [11]

Figure 4: 16x2 LCD Display

10K Ohm Potentiometer:

A potentiometer is a three-terminal resistor with a sliding or rotating contact that forms an adjustable voltage divider. If only two terminals are used, one end and the wiper, it acts as a variable resistor or rheostat.

The measuring instrument called a potentiometer is essentially a voltage divider used for measuring electric potential (voltage); the component is an implementation of the same principle, hence its name. Potentiometers are commonly used to control electrical devices such as volume controls on audio equipment.

Potentiometers operated by a mechanism can be used as position transducers, for example, in a joystick. Potentiometers are rarely used to directly control significant power (more than a watt), since the power dissipated in the potentiometer would be comparable to the power in the controlled load. [11]

Figure 5: 10K Ohm Potentiometer

D. Implementation

At first, the keypad was connected with the Arduino UNO. We connected all the necessary pins of the Keypad to the Arduino board. All 7 pins were connected using connecting wires.

Then we connected a 16×2 LCD Display to Arduino UNO. And also a 10K Ohm Potentiometer connected to the display for managing the voltage of the display. This allowed us to view the inserted values and messages.

Later we connected Servo Motor to the Arduino, the rotation of Servo Motor allows us to open or close the door. All the pins were connected wires.

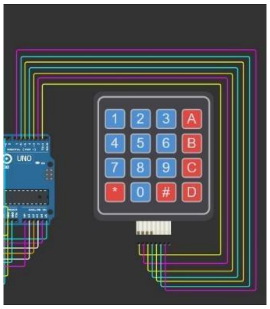

Figure 6: Connected Keypad with Arduino UNO

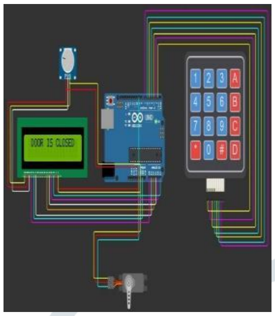

Figure 7: Potentiometer, LCD display, Servo Motor, and Arduino UNO connection

E. *Test Experimental Setup*

We implemented our Test Experimental Setup on WOKWI web application; A virtual electronic design automation software. We implemented the whole circuit in this software and uploaded the code for compilation, and it started the simulation.

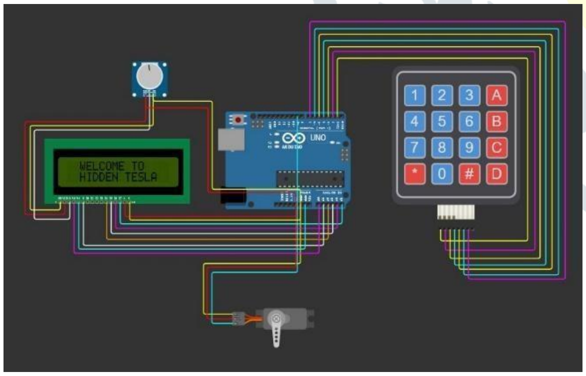

Figure 8: Full Circuit Diagram on WOKWI Simulation

Source Code:

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

```
#include <Servo.h>

Servo myservo;

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);

#define Password_Lenght 5
int pos = 0;

char Data[Password_Lenght];
char Master[Password_Lenght] = "3975";
byte data_count = 0, master_count = 0;
bool Pass_is_good;
char customKey;
const byte ROWS = 4;
const byte COLS = 3;
char keys[ROWS][COLS] = {
    {'1', '2', '3'},
    {'4', '5', 'G'},
    {'7', '8', '9'},
    {'*', '0', '#'}
};

bool door = true;

byte rowPins[ROWS] = {1, 2, 3, 4};
byte colPins[COLS] = {5, G, 7};

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

/////////////////////////////////////////

void setup() {
    myservo.attach(9);
    ServoClose();
    lcd.begin(1G, 2);
    lcd.print("  WELCOME TO");
    lcd.setCursor(0, 1);
    lcd.print("  HIDDEN TESLA");
    delay(3000);
    lcd.clear();
}


/////////////////////////////////////////


void loop() {
    if (door  = 0) {
        customKey = customKeypad.getKey();
        if (customKey  = '#') {
            lcd.clear();
            ServoClose();
            lcd.print(" DOOR IS CLOSED");
            delay(3000);
            door = 1;
        }
    }
```

```
      else Open();
      }


      ///////////////////////////////////////
      ///


      void clearData() {
      while (data_count  ÷ 0) {
      Data[data_count =] = 0;
      }
      return;
      }


      ///////////////////////////////////////
      ///


      void ServoOpen() {
      for (pos = 180; pos  ÷ 0; pos -= 5) {
      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]);
        data_count +;
    }
```

```
      if (data_count  = Password_Lenght - 1)
      {
      if (!strcmp(Data, Master)) {
              lcd.clear();
              ServoOpen();
              lcd.print("  DOOR IS OPEN");
              delay(2000);
              lcd.clear();

              lcd.print("PRESS # TO CLOSE");
              lcd.setCursor(0, 1);
              lcd.print("THE DOOR");
              door = 0;
              }

              else {
              lcd.clear();
              lcd.print(" WRONG PASSWORD!");
              delay(2000);
              lcd.clear();

              lcd.print("  PLEASE");
              lcd.setCursor(0, 1);
              lcd.print("  TRY
              AGAIN"); delay(2000);
              lcd.clear();
              door = 1;
              }
              clearData();
```

F. Cost Analysis

The estimated cost of our project is given below (Approximately).

Components	Quantity	Price (BDT)
Arduino UNO	1	1100
LCD Display	1	280
4x4 Matrix Keyboard	1	120
SG90 Servo Motor	1	190
10K Ohm Potentiometer	1	30

Breadboard	1	185
Wires	25/30	105
Total *(Approximately)* =		2010

IV. Result

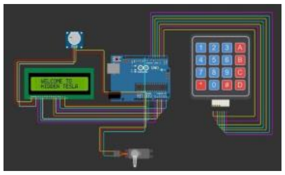
Figure 9: Welcoming Message when the systems is started

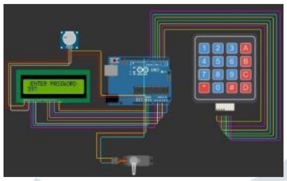
Figure 10: Instructed to enter the password

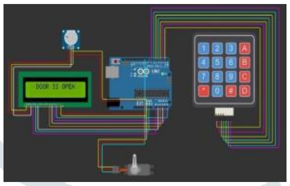
Figure 11: When the password matches, shows the message "DOOR IS OPEN" and Servo motor is rotated by 180°

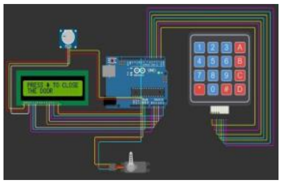
Figure 12: Instructed to press '#' to close the door

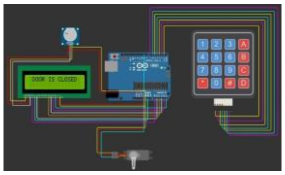
Figure 13: After pressing #, shows the message "DOOR IS CLOSED" and Servo Motor is 180° reversely rotated

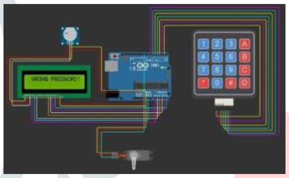
Figure 14: Shows the message "WRONG PASSWORD!" when wrong password is entered

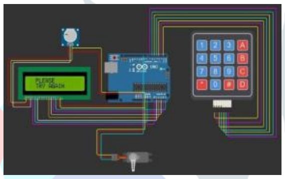
Figure 15: Instructed to try again

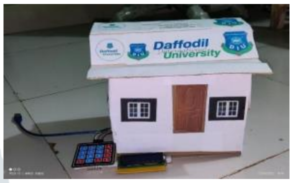

Figure 16: Applied all things by creating the home in real life

V. Conclusion

As a result, the "Smart Door Locking System utilizing Arduino" is a modern take on the traditional door lock. The innovation generated by the lock system with no more direct touch between the user and the lock is the end of the topic of smart Lock utilizing Arduino. This method is both inexpensive and simple to set up. Finally, it was revealed that the project performed as expected and that it can be implemented. Because the Arduino UNO microcontroller is used in this project, the design is simple, and the project may be completed in less time than with other techniques. A safe locking/unlocking system based on a keypad and Arduino is proposed in this paper. Adding a password on the Arduino side improves the security of the system.