



Digital keypad security door lock using Arduino with Tinkercad application

1.D.Narendar Singh, 2 .R.Nagaswetha, 3.B.Pavitra, 4.Pernapati Kushaal Raj ,
5.Gadipe Vishnuvardhan,6 .Guduru Pallavi 7.Goundla Anjali, 8.Gunde Venkatesh

1.Assoc.Prof, Dept. of ECE Anurag University,
2,3 Asst.Prof, Dept. of ECE Anurag University,
4,5,6,7 B.Tech III year Students, Dept. of ECE Anurag University

ABSTRACT

Every person needs to feel safe in their daily lives. Access control for doors is an important part of our security plan. Password-protected door locks let only approved people into restricted places. Normal locks aren't as safe as they used to be; anyone can get in if they get past these locks. Our system needs to work 24 hours a day, seven days a week. A board called an Arduino runs it. A keyboard is used to enter the password. The typed password is compared to the known password when a mixture password or a password with 1 to 6 digits is set. With the right password, the door opens and the condition is shown on the LCD. The door stays shut and the LCD screen shows "WRONG PASSWORD" if the password is wrong. The buzzer will also go off an infinite number of times when the wrong password is entered. Electronic door locks will be used instead of mechanical ones in the security door locking system. This is a big step forward for the future.

A. Introduction

Security means protecting yourself from harm and making sure you are safe. Security has always been important in places like homes, companies, schools, labs, and more, to keep our resources safe and stop people from getting into them illegally. There used to be fewer security measures in place to keep people from getting in without permission. In the past few years, many different security measures have been added to these kinds of websites and apps. At the moment, there are password-protected systems, RFID card technologies, biometric-protected systems, OTP-based systems, cryptography-based systems, and other security options. There are different kinds of apps for each device. Every day there are more and more steals, so safety is at risk. So, a digital code lock is a simple way to keep your home or locker

safe. You can only open your door if you enter the right password. The ARDUINO UNO can be used to make a password-based door lock system. With this system, the door is unlocked and the person can enter the safe area after entering the right code or password is put in. The Password Based Door Lock System with Arduino UNO is an easy project that lets you open the door with a strong password. That person can get into the safe area after entering the right code or password using an Arduino UNO. The door will then be opened. If someone else comes in, it will ask you for the password again. If the person enters the wrong password, the door will stay locked, stopping them from entering. The main idea behind this project is to use an Arduino UNO to make a door lock system that requires a password. To explain how it works, here it is:

B. BLOCK DIAGRAM

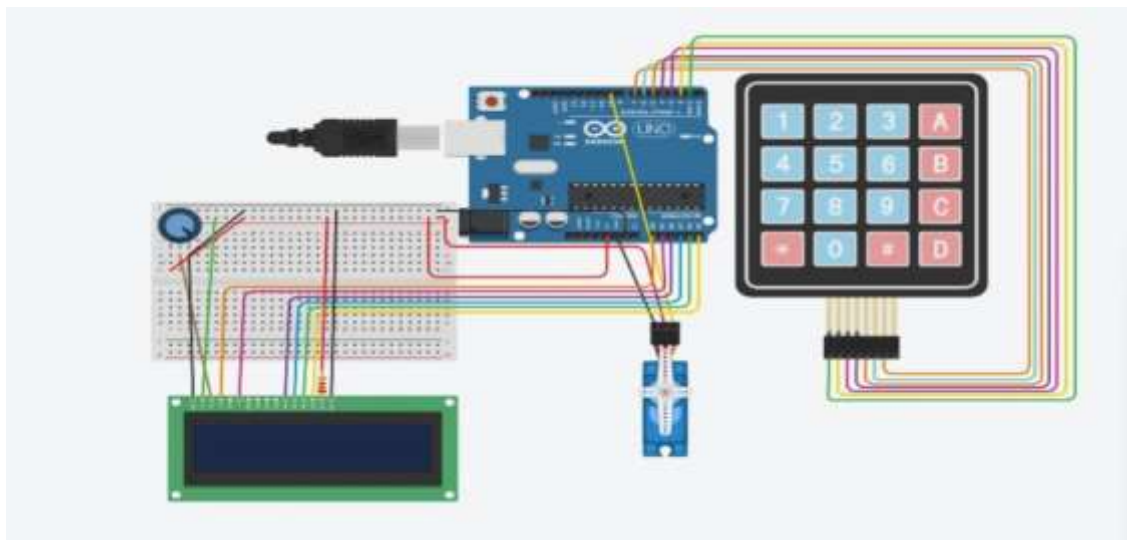


Fig 1. Blockdiagram

Components :

Arduino UNO:

- The pin Mode() method on an Arduino pin allows it to be set as an input or output. You may modify the behavior of a pin as an input or output by using the pin Mode () function. The internal pull-up resistors can be activated by using the option INPUT_PULLUP. Furthermore, the INPUT mode specifically disables the internal pull-ups.

Features:

- Arduino boards have the ability to take analog or digital input signals from various sensors and translate them into outputs. Examples of these activities include connecting to the cloud, turning on and off motors, and switching LEDs on and off.
- The Arduino IDE may be used to upload software, or transmit instructions to the board's microcontroller, allowing for control over its operations.
- The Arduino can load fresh code into the board without the need for additional hardware, known as a programmer. A USB cord is all we need.
- Programming is made simpler by the use of a condensed version of C++ in the Arduino IDE.
- Lastly, Arduino simplifies the microcontroller's operations into a more user-friendly packaging by using a

common

form

factor.

• Breadboard: Protoboards, often known as breadboards, are the building blocks used in electronics prototyping. When the phrase was first used, it meant a real breadboard—a shiny surface meant for slicing bread.

C. Methodology

Modern electronic locking systems, which combine mechanical and electrical components, can replace locks that were mechanically operated in the past. A strong password serves as a door unlock in these systems. Modern, sophisticated locking systems can take the place of outdated, mechanical lock systems. These methods are extremely clever and incorporate both mechanical and electrical components. The project's functioning is shown in detail in the flow chart in the figure below. At first, the password is understood. When the door-securing device is turned on, a servo angle is reset. Next, a keypad is used to input the password. When a passcode is input on a keypad, the Arduino can read it and compare it to the previously known password. If the password input matches the known password, the servo motor unlocks the door for ten seconds, and the LCD shows the current status. If not, the user hears a beeper alerting them to the wrong password, and the door stays closed with the message "Door is closed." The circuit includes an Arduino UNO, which turns on the buzzer and communicates with the circuit owner.

D. Flow Chart

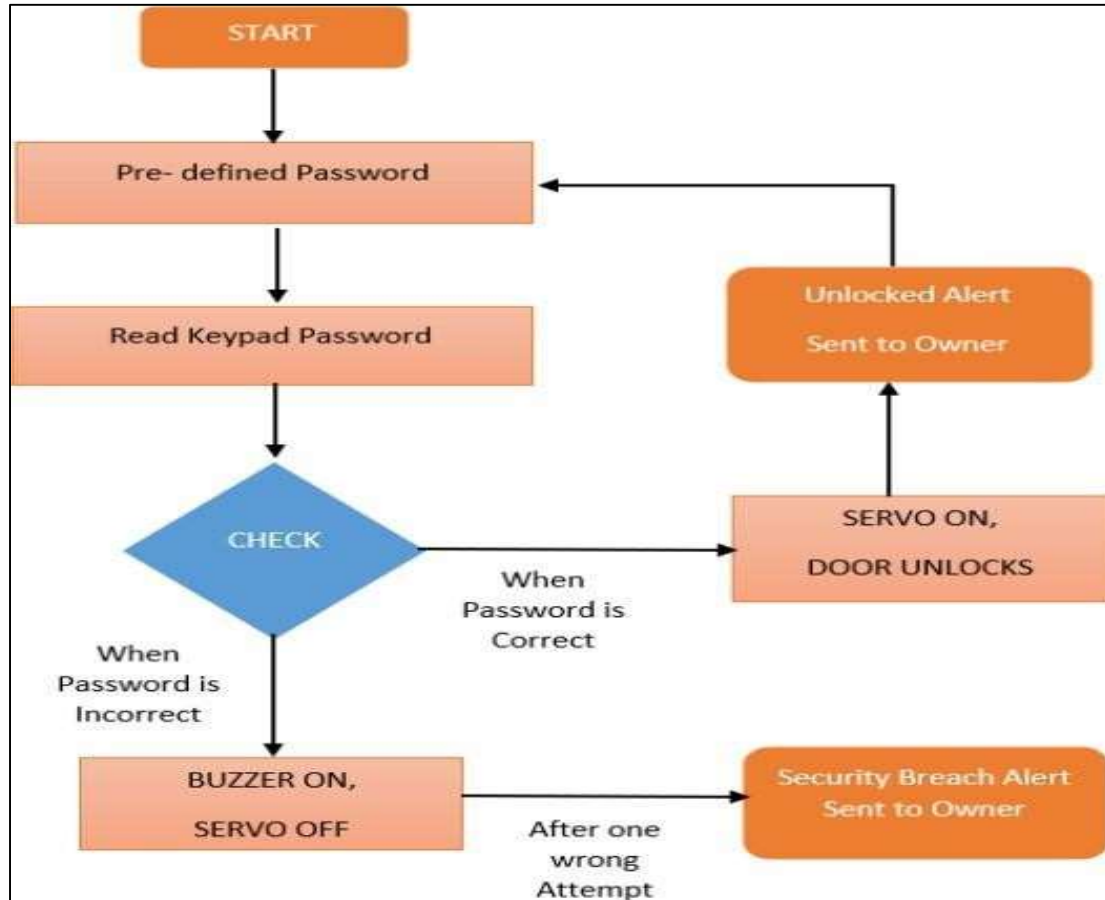


Fig 2. Flowchart

IoT (Internet of Things) security door lock systems provide consumers with ease and control, but they also have certain drawbacks and possible hazards.

To do that just open the link below and download the ZIP file. Then on your Arduino IDE navigate to Sketch -> Include Library -> Add.ZIP Library and browse for the file that you just downloaded.

- [Arduino Keypad Library](#)

After inserting all the header and library files, assign all the pin for LCD and define the password length and set the initial position of the servo to 0. After that, take a “char” datatype for declaring the number that can hold it including the null character.

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

#include <LiquidCrystal.h>

#include <Servo.h>

Servo myservo;

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

#define Password_Length 7 // Give enough room for six chars + NULL char

int pos = 0; // variable to store the servo position

char Data[Password_Length]; // 6 is the number of chars it can hold + the null char = 7

char Master[Password_Length] = "123456";
```

With this piece of code (char Master[Password_Length] = "123456";) -- under the Char Master, I declare the password of the door lock, then assign the number of rows and columns in the keyboard and also declare keyMaps and connect with rows and columns. Under the void setup, initialize the servo signal pin D9, servo status closed and print the name of the project/device/company with 3 seconds of delay on LCD time of starting the device.

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

Under the loop function, the simple if-else condition is there. According to status (it's locked automatically), print "Door is close" with 3 seconds of delay and servo rotate to close position, door data count as 1 otherwise door locker remains open and data count 0, servo open rotate to the position goes 0 degrees to 180 degrees and to close it goes from 180 to 0. The servo open and servo close functions are shown below.

```
void ServoOpen()
{
for (pos = 180; pos >= 0; pos -= 5) { // goes from 0 degrees to 180 degrees
// in steps of 1 degree
myservo.write(pos); // tell servo to go to position in variable 'pos'
delay(15); // waits 15ms for the servo to reach the position
}
}
void ServoClose()
{
for (pos = 0; pos <= 180; pos += 5) { // goes from 180 degrees to 0 degrees
myservo.write(pos); // tell servo to go to position in variable 'pos'
delay(15); // waits 15ms for the servo to reach the position
}
```

```
}

```

Write the position of the servo with 15 seconds of delay to reach the servo position. Under the void open function, print on LCD “Enter Password” then the condition specifies that the entered password should be same with custom key, within this body data counted and stores char into the data array, incremented and input password printed on LCD if the data length (number of key) and input password matches with Data Master (where the pre-defined keys are present). Then as an action LCD clear, servo drive, print on LCD “Door is open” and the data counter is reset to 0.

If the input key does not match with Data Master, as an action on LCD clear print on LCD “Wrong Password “ to notify with 1 second of delay and remains in its lock position with data counter sets to 1 and continue this process in a loop.

```
if (data_count == Password_Lenght - 1) // if the array index is equal to the number of expected chars, compare
data to master

{

  if (!strcmp(Data, Master)) // equal to (strcmp(Data, Master) == 0)

  {

    lcd.clear();

    ServoOpen();

    lcd.print(" Door is Open");

    door = 0;

  }

  else

  {

    lcd.clear();

    lcd.print(" Wrong Password");

    delay(1000);

    door = 1;

  }

  clearData();

```

}

E. Arduino Keypad DoorLock Assembling and Testing

Now, arrange everything neatly using the shell, place everything on a 4 or 6-inch plastic box, and power it using a mobile charger. Although it is less secure, it is still functional. Originally, I had planned to assemble my lock using 3D printing, but after creating my files, I discovered that the process is very expensive. As a result, I just fixed the servo first, then connected the regular slide locker with my servo using a metal plate.



You must create a 3D door lock model that integrates with this servo if you need additional security. If you have access to a 3D printer, you may download the necessary STL files from the URL below and 3D print them.



With **3 seconds of delay**, quickly lock the door and directly check the display for the status of the door lock as shown in the picture given below.



The door seems to be manually locked for a brief moment when you click the "#" key, after which it says "Enter Password." Conversely, Door is Open indicates that the door has been opened with the proper password.

F. CONCLUSION

This project used an Arduino board and keypad to create and construct a door locking mechanism. The password and the default password are compared by the Arduino. A message will appear on the LCD and a servo motor will rotate in accordance with the password. When the password matches the default one that was set, the door unlocks. When input from the user is inaccurate or invalid, a buzzer will sound. For those searching for more secure doors and lockers, it is rather helpful. These methods are extremely effective because they combine the usage of mechanical and electrical components. Maximum security will be provided by the password-based door lock system, enabling customers to completely meet their demands. In the interim, a person may afford to spend as little as possible on a door locking mechanism to protect their belongings.

References:

1. Abhilash Kanakanti and D.Singh, "College Buses And Students Monitoring System With IOT", International Journal of Advanced Research in Advance Engineering & Technology, 6 (1), 200–204, (2017).D. N. Singh and U. Divya, "An intelligent parking guidance and information system with ARM9 microcontroller", Int. J. Latest Trends Eng. Technol., vol. 2, pp. 234-239, Sep. 2013.
2. D. Narendhar Singh, M. Hema and M. Joseph Stalin, "IoT Based Healthcare Monitoring for Driver's Community", International Journal of Engineering Science and Computing, March 2017.
3. Swetha, R. Naga and Gona, Ashwini and Singh, D. Narendar, IoT Based Smart Garbage Monitoring System with Geo-Tag (February 21, 2020). Proceedings of the 4th International Conference: Innovative Advancement in Engineering & Technology (IAET) 2020, Available at

SSRN: <https://ssrn.com/abstract=3554257> or <http://dx.doi.org/10.2139/ssrn.3554257>

4. D.Narendar Singh, P. Ramakrishna, J. Nikhileshwar, G. Akshaya, K. Karthik, & K. Vinitha Lakshmi. (2022). Lorawan-Based Satellite Monitoring System Uses IMU Sensor. *RES MILITARIS*, 12(4),1300–1309.Retrieved from <https://resmilitaris.net/index.php/resmilitaris/article/view/1949>
5. **Abhilash Kanakanti and D.Singh, “College Buses And Students Monitoring System With IOT”, *International Journal of Advanced Research in Advance Engineering & Technology*, 6 (1), 200–204, (2017).**
6. Pavitra, B., Singh, D. N., Sharma, S. K., & Hashmi, M. F. (2023). Dementia prediction using novel IOTM (Internet of Things in Medical) architecture framework. *Intelligent Data Analysis*, (Preprint), 1-17.
7. Pavitra, B., Singh, D. N., & Hashmi, M. F. (2021). Voice-Controlled Biped Walking Robot for Industrial Applications. *Innovations in the Industrial Internet of Things (IIoT) and Smart Factory*, 79.
8. D. N. Singh, B. Pavitra, A. Singh, and J. A. Reddy, “Performance of IoT-Enabled Devices in Remote Health Monitoring Applications,” in *Computational Intelligence in Medical Decision Making and Diagnosis*, CRC Press, 2023, pp. 131–140.
9. **Singh, D. N. (2013). Ravi teja ch. v.“. Vehicle Speed Limit Alerting and Crash Detection System at Various Zones “*International Journal of Latest Trends in Engineering and Technology (IJLTET) IJLTET*, 2(1).**
10. D.Narendar Singh and K.Tejaswi (M.Tech), “Real Time Vehicle Theft Identity and Control System Based on ARM 9”, *International Journal of Latest Trends in Engineering and Technology (IJLTET)*, Vol. 2 Issue 1 January 2013.
11. <https://projecthub.arduino.cc/123samridhgarg/arduino-calculator-bce0df>
12. <https://github.com/nikita9604/Scientific-Calculator-with-Keypad>
13. <https://www.instructables.com/Scientific-Calculator-With-Evive-arduino-Powered-E/>
14. <https://circuitdigest.com/microcontroller-projects/arduino-calculator-using-4x4-keypad>