ACMEGRADE

MINI PROJECT ON

IOT Based Digital Keypad Door Lock System

MADE BY :-

Dev Ravindra Munvar
Divyansh Marwaha
Desmond Dillon A.R
Shiddhant Gautam

UNDER THE GUIDANCE OF :-

Prof. Manjunath G
AcmeGrade
Embedded Systems

OUR OFFICIAL TRAINING PARTNER

TABLE OF CONTENTS

ABSTRACT

Traditional physical keys and locks can be cumbersome to manage and pose significant security risks due to potential loss or theft. To address these challenges, we propose an Arduino-Based Combinational Lock System as a more secure and convenient alternative. This system leverages modern electronics to create an efficient and user-friendly locking mechanism.

The system is designed using an Arduino UNO board, a keypad for input, an LCD screen for real-time feedback, and a servo motor for the physical locking and unlocking action. Users enter a predefined combination on the keypad, and if the correct code is entered, the servo motor rotates to unlock the mechanism, granting access and the LCD screen displays a message such as "Door is Open, Waiting...". Conversely, if an incorrect code is entered, the LCD screen prominently displays an error message, "Wrong Password," enhancing security by notifying the user of the unsuccessful attempt.

This project effectively mitigates the need for physical keys, reducing the risk of unauthorized access due to lost or stolen keys. The integration of the LCD screen provides immediate and clear feedback, making the system intuitive and easy to use. The use of a servo motor for the locking mechanism ensures precise and reliable operation, further enhancing the security and robustness of the system.

The Arduino-Based Combinational Lock System exemplifies an innovative approach to securing access through electronic means. It combines the simplicity of a keypad interface with the sophistication of microcontroller programming, offering a scalable solution that can be adapted for various applications, from residential doors to secure storage units. By incorporating these advanced features, the system not only improves security but also enhances user convenience, making it a valuable contribution to modern locking solutions.

1. INTRODUCTION

1.1. Motivation

Traditional physical keys have long been a reliable method for securing doors, but they come with significant drawbacks. Managing multiple keys can be cumbersome, increasing the risk of keys being lost, stolen, or misplaced, which compromises security.

The motivation behind developing an Arduino-Based Combinational Lock System is to address these issues by replacing physical keys with an electronic system. This lock system enhances security by eliminating the need to carry multiple keys and reducing the risk of unauthorized access due to lost or stolen keys. Additionally, it offers the flexibility to easily reconfigure access codes, providing a modern, efficient, and user-friendly solution for access control.

1.2. Proposed System

The proposed solution utilizes a keypad, an LCD screen, and a servo motor, all interfaced with an Arduino UNO board, to create a secure and convenient combinational lock system. This system enhances security and eliminates the need for carrying physical keys by using a sequence of numbers entered via the keypad to unlock the mechanism.

When the correct combination is entered on the keypad, the Arduino activates the servo motor to unlock the door, and the LCD screen displays a message such as "Door is Open, Waiting..." If an incorrect combination is entered, the LCD screen displays a "Wrong Password" message, providing immediate feedback to the user. This approach not only simplifies the locking mechanism but also offers flexibility in reconfiguring access codes, making it a modern and efficient alternative to traditional physical keys.

2. METHODOLOGY

2.1. Block Diagram

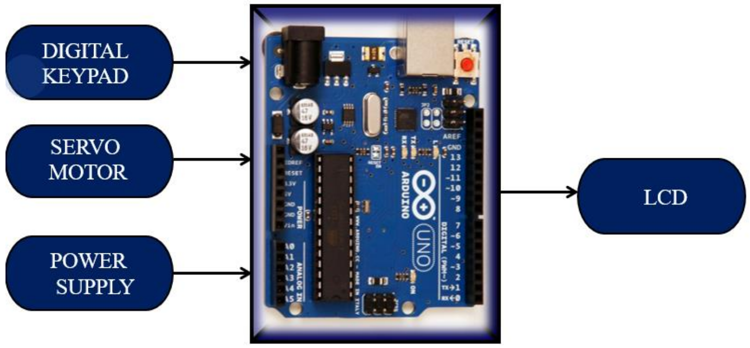


Figure 2.1. Block Diagram

An Arduino UNO Board is used in the simulation of a combinational lock system. The system includes a keypad for entering the combination, an LCD screen for displaying messages, and a servo motor for controlling the locking mechanism.
Connecting and jumper wires are used to interface the keypad with the digital pins of the Arduino UNO. The LCD screen is connected to both the digital and analog pins, allowing it to display real-time feedback such as "Door is Open, Waiting..." for correct entries and "Wrong Password" for incorrect ones. The servo motor is connected to the digital pins, enabling it to physically unlock or lock the mechanism based on the input from the keypad. This configuration creates an efficient and user-friendly combinational lock system that enhances security and eliminates the need for physical keys.

2.2. Design Flow

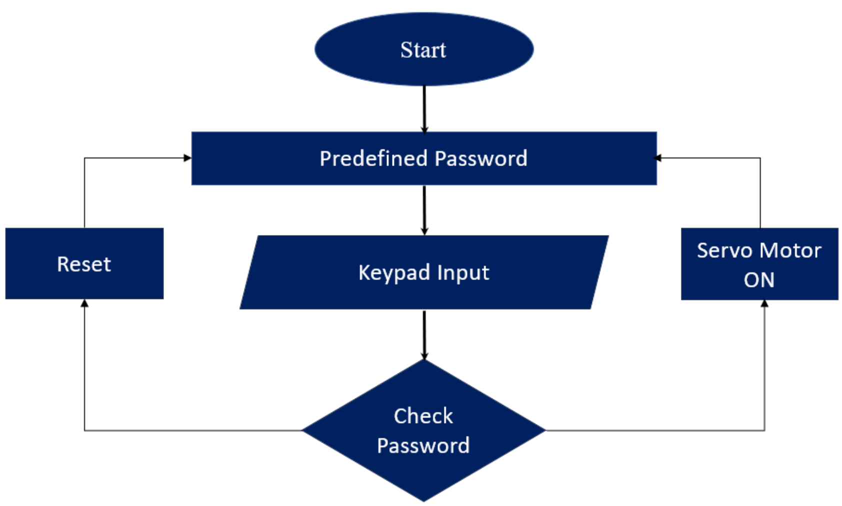
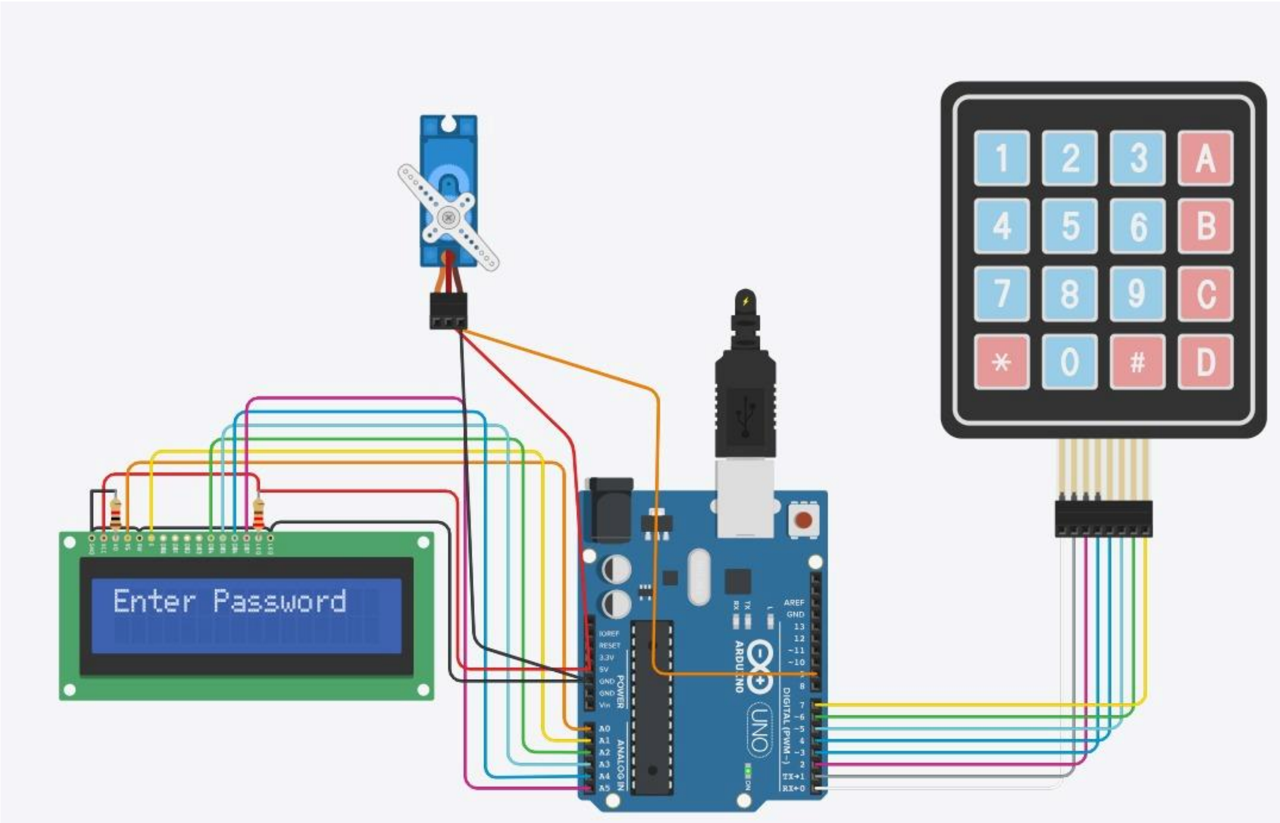


Figure 2.2. Design Flow

If the correct combination of numbers is entered on the keypad, the system will activate the servo motor to unlock the door, and the LCD screen will display a message such as "Door is Open, Waiting...". If an incorrect combination is entered, the LCD screen will display "Wrong Password", indicating that the combination entered is incorrect. This provides immediate visual feedback to the user, enhancing the usability and security of the system.

2.3. Circuit Diagram



2.4. Arduino Code

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

Servo myservo;

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

#define Password_Lenght 5 // Give enough room for six chars + NULL char

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

char Data[Password_Lenght]; // 6 is the number of chars it can hold + the null char
= 7
char Master[Password_Lenght] = "1234";
byte data_count = 0, master_count = 0;
bool Pass_is_good;
char customKey;

const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};
bool door = true;

byte rowPins[ROWS] = {1, 2, 3, 4}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {5, 6, 7, 8}; //connect to the column pinouts of the keypad

Keypad customKeypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS);
//initialize an instance of class NewKeypad

void setup()
{
  pinMode(10,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");
      digitalWrite(10,LOW);
      delay(3000);
      door = 1;
    }
  }

  else Open();
}

void clearData()
{
  while (data_count != 0)
  { // This can be used for any array size,
    Data[data_count--] = 0; //clear array for new data
  }
  return;
}

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
  }
}

void Open()
{
  lcd.setCursor(0, 0);
  lcd.print(" Enter Password");

  customKey = customKeypad.getKey();
  if (customKey) // makes sure a key is actually pressed, equal to (customKey !=
NO_KEY)
  {
    Data[data_count] = customKey; // store char into data array
    lcd.setCursor(data_count, 1); // move cursor to show each new char
    lcd.print(Data[data_count]); // print char at said cursor
    data_count++; // increment data array by 1 to store new char, also keep track of
the number of chars entered
  }

  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();
      digitalWrite(10,HIGH);
      ServoOpen();
      lcd.print("  Door is Open");
```

```
      door = 0;
    }
    else
    {
      lcd.clear();

      lcd.print("  Wrong Password");
      delay(1000);
      door = 1;
    }
    clearData();
  }
}
```

2.5. Limitations of Existing Systems

While keyless entry systems such as keypads, biometrics, and smartphone apps offer numerous advantages over traditional mechanical locks and keys, they also present several limitations. These systems rely heavily on technology, which can malfunction or fail, potentially locking out authorized users. They are also vulnerable to hacking and cyberattacks, posing significant security risks. Privacy concerns arise with biometric systems, as sensitive personal data must be securely handled to prevent misuse. Additionally, the initial setup cost can be high, especially for large organizations, and integrating these systems may require significant time and resources, including staff training. Dependence on devices like smartphones can be problematic if they are lost, stolen, or out of battery. Users may also find it challenging to adapt to new methods of access, leading to frustration or resistance. Finally, keyless entry systems require regular maintenance and updates to remain secure and efficient. Addressing these limitations requires careful planning, robust cybersecurity measures, and ongoing support and maintenance.

3.TOOLS AND LANGUAGES REQUIRED

3.1. Simulation Tool

TinkerCad

Tinkercad is a free online collection of software tools that help in the creation and designing of electronic circuits.

Figure 3.1.1. Tinkercad Software

3.2. Software Components

Arduino UNO

The Arduino Uno is a microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with digital and analog input/output (I/O) pins that may be interfaced to other circuits. The board has 14 digital I/O pins, 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable.

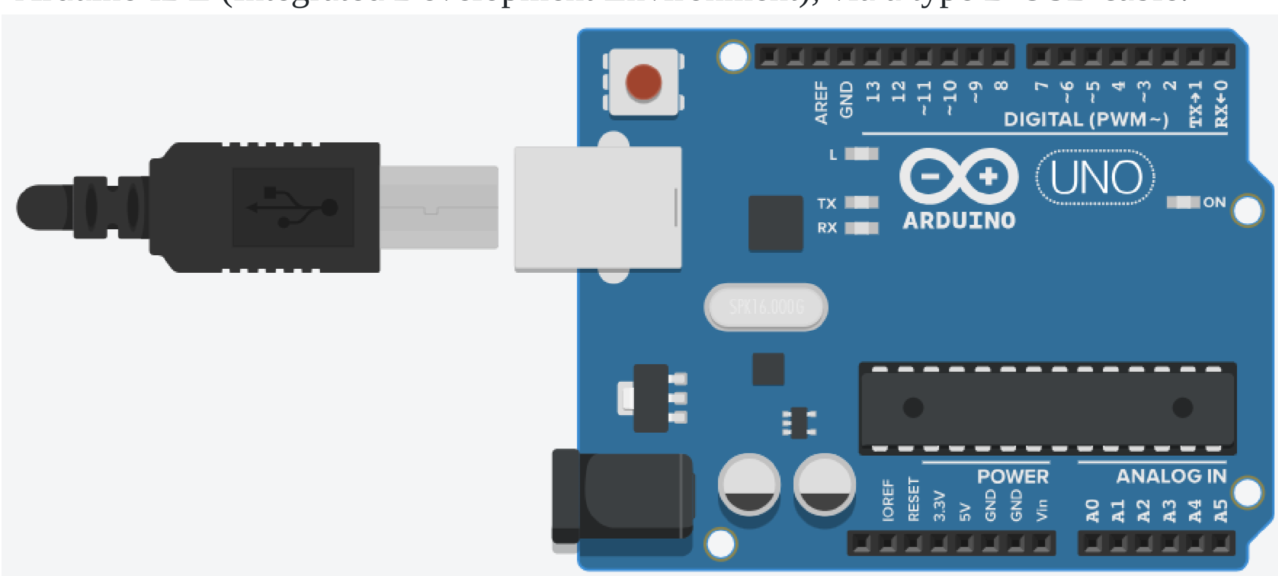


Figure 3.2.1. Arduino UNO Board

Keypad (4x4)

A 4x4 keypad is a matrix of 16 buttons arranged in a 4-row by 4-column grid, commonly used in electronic devices for user input. Each button press is detected by scanning the rows and columns, allowing for efficient input recognition. The keypad is often used in combination with microcontrollers or other processing units to implement numeric input, password entry, or menu navigation. Its compact design and straightforward connection make it a popular choice for various applications where user interaction is required.

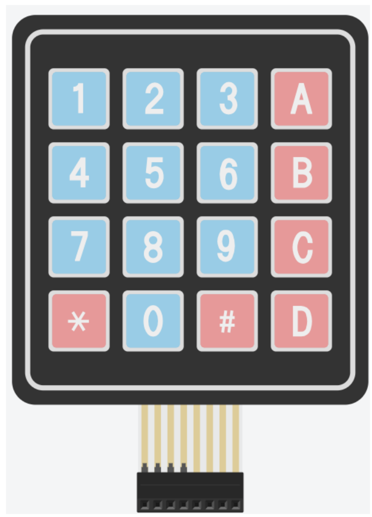


Figure 3.2.2. Keypad (4x4)

LCD (16x2)

The 16x2 LCD is a liquid crystal display module that features 16 columns and 2 rows of characters. It is commonly used in electronic projects and devices to provide alphanumeric output to users. The display operates by using liquid crystal technology to form characters on a backlit screen, allowing for clear and readable text. The 16x2 LCD is often interfaced with microcontrollers or development boards to display messages, status updates, or other information. Its versatility and ease of integration make it a valuable component in both hobbyist and professional electronics.

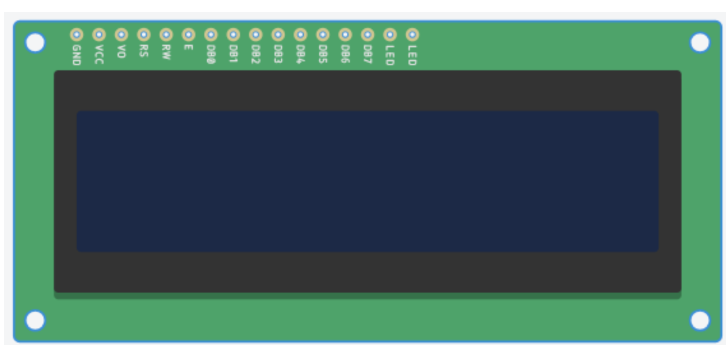


Figure 3.2.3. LCD 16x2

Micro Servo

A micro servo is a compact, motorized device designed for precise control of angular movement. Typically used in robotics, remote-controlled vehicles, and various hobbyist projects, micro servos offer a range of motion from 0 to 180 degrees. They are driven by a small motor and controlled by a pulse-width modulation (PWM) signal, allowing for accurate positioning of the servo arm. The micro servo's small size and lightweight design make it ideal for applications where space is limited but precise control is required.

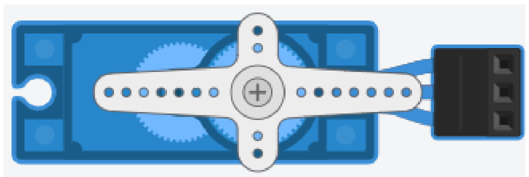


Figure 3.2.4. Micro Servo

4. SIMULATION RESULTS

- Correct Code = Successful Output

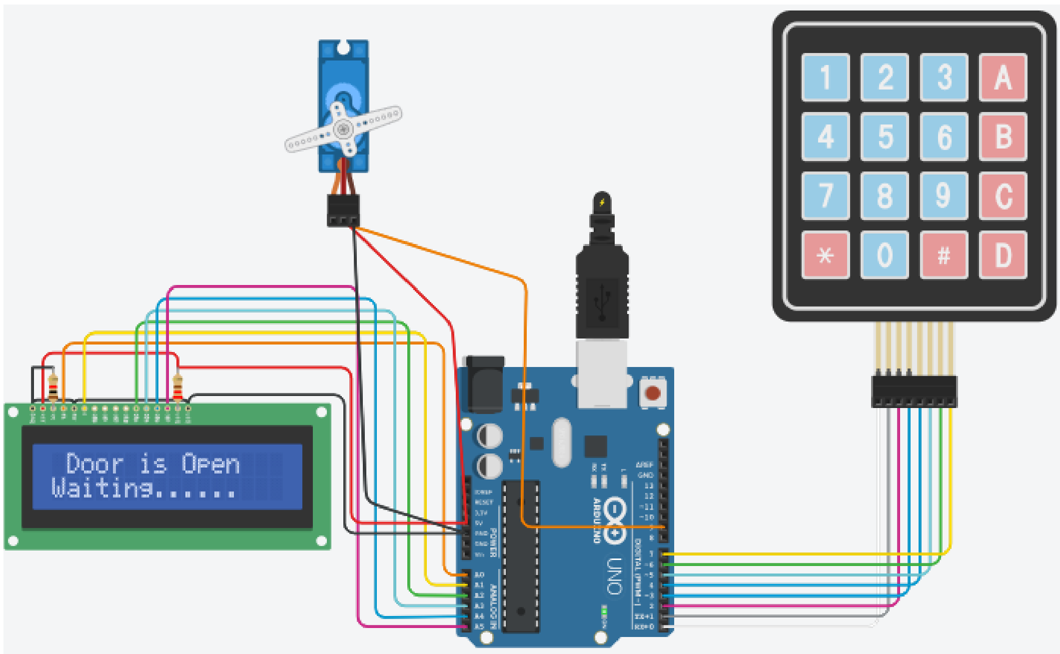


Figure 4.1. Simulation output when the input code is correct

- Incorrect Code = Unsuccessful Outpu

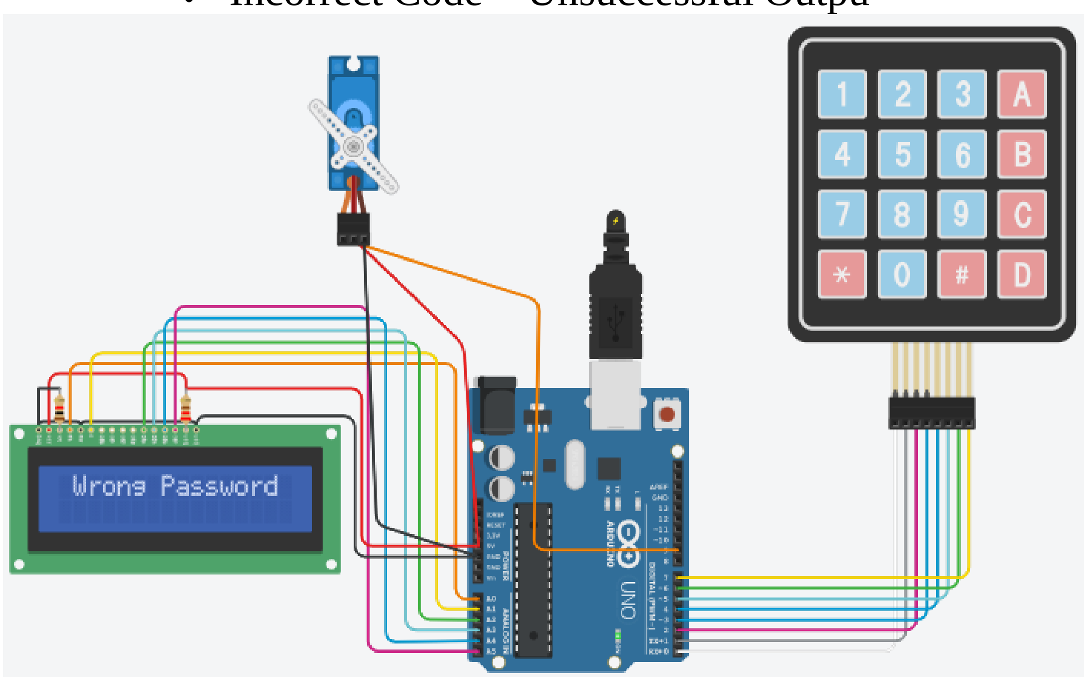


Figure 4.2. Simulation output when the input code is wrong

5. ADVANTAGES

- **Pick-Proof**
  By eliminating physical keys, the system reduces the risk of lock picking and unauthorized access. The electronic locking mechanism is immune to traditional lock-picking methods.
- **No More Keys**
  The system eliminates the need for physical keys, reducing the risk of loss or theft. It also removes the hassle of frequent key replacements, making access management simpler and more secure.
- **Enhanced Control**
  The electronic nature of the system allows for easy control and restriction of access. Users can quickly change the passcode to maintain security and adapt to new access requirements.
- **Immediate Feedback**
  The integrated LCD screen provides real-time feedback on access attempts, enhancing user experience and security. Clear messages such as "Door is Open, Waiting..." and "Wrong Password" keep users informed of the system status.
- **Suitable for All Users**
  The system offers convenience for all users, including the elderly or disabled, by providing an easy-to-use interface without the physical effort required to manage traditional keys.

6. FUTURE SCOPE

The scope of electronic locking systems, such as the Arduino-Based Combinational Lock System, is expanding rapidly due to their convenience and enhanced security features. These systems eliminate the need for physical keys or cards, requiring only knowledge of a passcode to grant access.

Potential applications for such electronic locks are extensive, including office buildings, residential complexes, shopping malls, industrial facilities, and educational institutions. They offer versatility in securing various environments where reliable access control is needed. However, their reliance on electrical power means they are less suitable for locations without a consistent electricity supply.

One of the significant advantages of electronic locks is their robust resistance to unauthorized access. With a large number of possible combinations—10,000 for a four-digit passcode and 100,000 for a five-digit passcode—these systems provide a high level of security against hacking attempts. Moreover, they can be integrated into more complex security systems, enhancing their adaptability and functionality for various security needs.

CONCLUSION

This report highlights the advantages and importance of adopting the Arduino-Based Combinational Lock System over traditional physical locks and keys. The proposed system offers a practical and secure alternative by eliminating the need for physical keys, which are susceptible to being lost, misplaced, or stolen, thereby compromising security.

The system was designed using a straightforward electronic circuit incorporating a keypad for input, an LCD screen for real-time feedback, and a servo motor for the locking mechanism, all controlled by an Arduino UNO board. The design emphasizes ease of use and robust security, addressing common issues associated with conventional locks.

The circuit design was validated using simulation software, TinkerCad, which demonstrated successful execution of the system. This confirms the system's viability and effectiveness in enhancing security while offering a user-friendly experience.

REFERENCES

- Arduino-Based Combinational Lock System
- Electronic Access Control Systems Using Arduino
- Advanced Electronic Lock Mechanisms
- LCD Integration in Arduino Projects
- Servo Motors for Locking Systems
- TinkerCad Simulation Software
- Arduino UNO Board Specifications
- Breadboard Circuit Design
- Keypad Input Systems
- Arduino-Driven Lock Mechanisms