

# Keypad-based Security System with Password Protection

## Components Used:

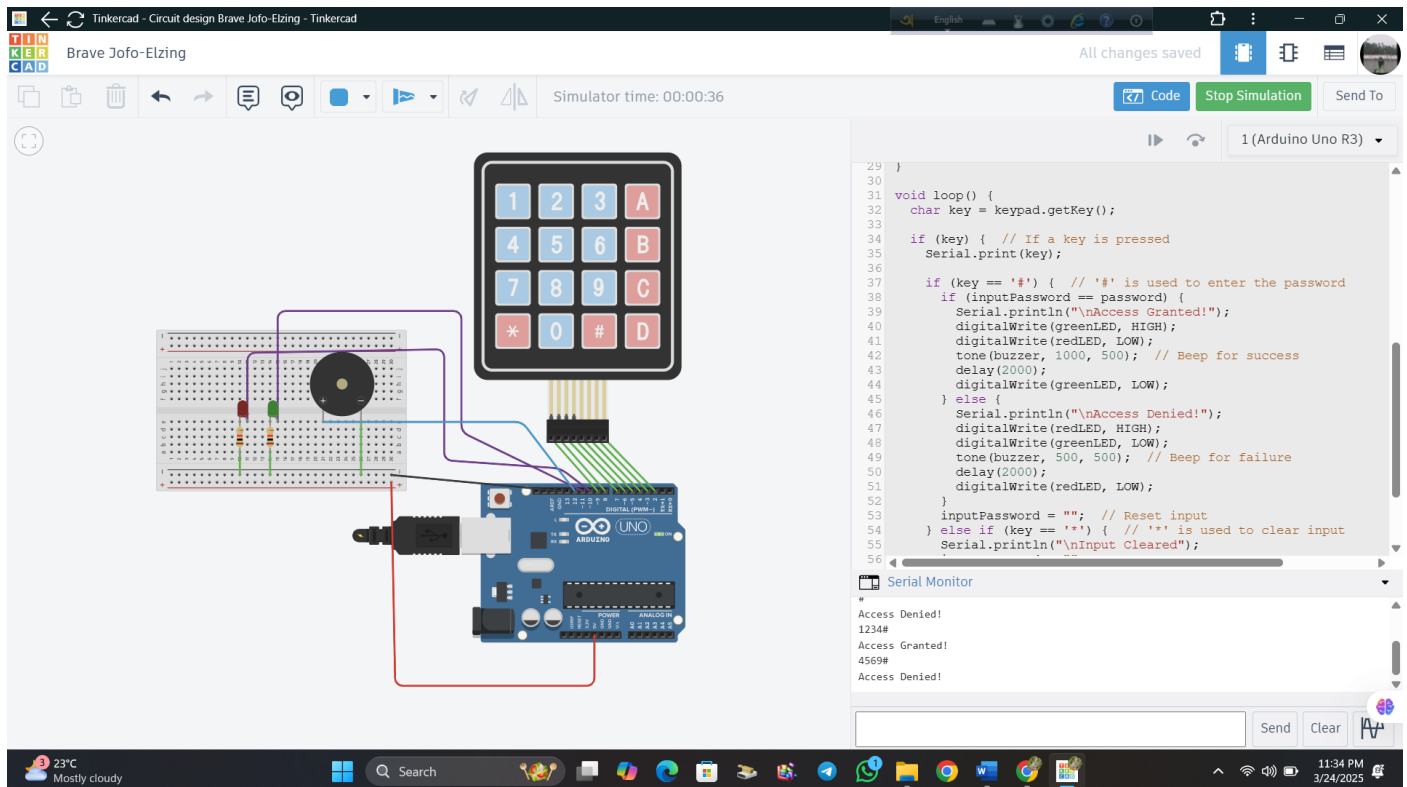
- Arduino, 4x4 Keypad, LED Indicators, Resistors

## 4x4 Keypad



A 4\*4 matrix is used to insert input values into the project. This component has a total of 8 terminals, driven out from the 16 buttons present in the module.

## Circuit:



## Working Principle:

- Keypad is used to enter a password.
- If the correct password is entered:
  - Green LED turns ON (indicating access granted).
- If the wrong password is entered:
  - Red LED turns ON (indicating access denied).

## Code:

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

```
const byte ROWS = 4; // Four rows
```

```
const byte COLS = 4; // Four columns
```

```

char keys[ROWS][COLS] = {
    {'1', '2', '3', 'A'},
    {'4', '5', '6', 'B'},
    {'7', '8', '9', 'C'},
    {'*', '0', '#', 'D'}
};

byte rowPins[ROWS] = {9, 8, 7, 6}; // Connect to the row pinouts of the keypad
byte colPins[COLS] = {5, 4, 3, 2}; // Connect to the column pinouts
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);
const String password = "1234"; // Set your password
String inputPassword = ""; // Store user input
const int greenLED = 10; // Access Granted LED
const int redLED = 11; // Access Denied LED
const int buzzer = 12; // Optional buzzer

void setup() {
    pinMode(greenLED, OUTPUT);
    pinMode(redLED, OUTPUT);
    pinMode(buzzer, OUTPUT);
    Serial.begin(9600);}

void loop() {
    char key = keypad.getKey();
    if (key) { // If a key is pressed
        Serial.print(key);
        if (key == '#') { // '#' is used to enter the password
            if (inputPassword == password) {
                Serial.println("\nAccess Granted!");
                digitalWrite(greenLED, HIGH);
                digitalWrite(redLED, LOW);
                tone(buzzer, 1000, 500); // Beep for success
                delay(2000);
                digitalWrite(greenLED, LOW);
            }
        }
    }
}

```

```

else {
    Serial.println("\nAccess Denied!");
    digitalWrite(redLED, HIGH);
    digitalWrite(greenLED, LOW);
    tone(buzzer, 500, 500); // Beep for failure
    delay(2000);
    digitalWrite(redLED, LOW); }
inputPassword = ""; // Reset input
}

else if (key == '*') { // '*' is used to clear input
    Serial.println("\nInput Cleared");
    inputPassword = "";
}

else {
    inputPassword += key; // Append pressed key to input    }}}

```

### Setup & Functionality:

- **Keypad is connected to Arduino using multiple pins.**
- **Code verifies the entered password.**
- **Serial Monitor displays the entered password and its validity.**

### Output Observation:

- **Correct password → Green LED ON → Access Granted.**
- **Wrong password → Red LED ON → Access Denied.**

### Conclusion:

The **Keypad-Based Security System with Password Protection** successfully demonstrates a simple yet effective access control mechanism using an **Arduino, 4x4 keypad, LEDs, and a buzzer**. It allows user authentication by verifying a password and providing **visual and audio feedback** for correct or incorrect entries. The system is **easy to modify** and can be enhanced with features like **EEPROM storage, LCD display, IoT integration, or servo-controlled locks** for real-world applications such as **door security, safes, and restricted access areas**. This project highlights the fundamentals of **embedded security** and serves as a foundation for more advanced implementations.