NETAJI SUBHAS UNIVERSITY OF TECHNOLOGY

NEW DELHI

Electronics Design Workshop

Password Based Circuit Breaker

By

Aryan (2022UEC2502)

Priyanshu Chaudhary(2022UEC2518)

Advay(2022UEC2533)

Introduction

Password Based Circuit Breaker is a project that helps in controlling the circuit with the help of a password. The control (ON/OFF) of the input circuits lies with the user. This project is arranged in such a way that the user has to enter the password to turn ON/OFF the circuit.

This project consists of an **Arduino Uno(ATmega328P)**, 4x3 Matrix Keypad, 16 x 2 LCD Display, **2–Channel Relay Module** and two Loads. Here, the LCD is used to display the information related to the load and the keypad is used to enter the password. Two loads (acting as input circuits) are used to indicate circuit breaker state (Light ON – Circuit Active and Light OFF – Circuit Not Active).

This project aims at ensuring the safety of the user while dealing with high voltage circuits. If there is any fault in the circuit connections, then the user will switch off the power supply to the circuit by entering the password and comfortably repair the connections, and afterwards the used switch on the supply to the particular circuit by entering the appropriate password. Separate passwords are assigned for each circuit.

Project Description

This project uses a microcontroller, in the form of Arduino Uno, to turn a circuit on/off by checking the password entered.

It consists of a 4x3 keypad (uses digital pins of Arduino) to help enter the password and a 16x2 LCD (connected to Arduino via an I2C Module) is used to display the password entered and displays the status of the load. A relay module is used to connect the two AC loads to the Arduino Uno. The relay module turns the circuit on or off based on the password entered.

The entire setup is soldered on a zero board. The relay module can be used to connect any 2 loads to make their operation password based.

Component Description

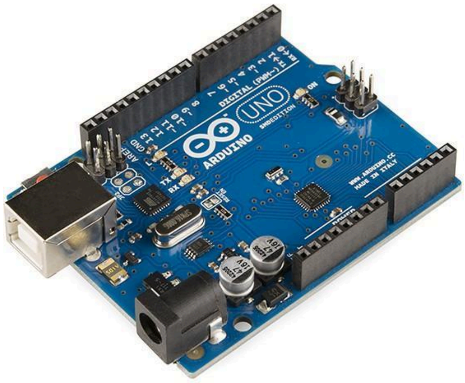
The following components are used in the project:

1. Arduino Uno

Arduino Uno is a 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 (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller.

Arduino UNO is a low-cost, flexible, and easy-to-use programmable open-source microcontroller board that can be integrated into a variety of electronic projects. This board can be interfaced with other Arduino boards, Arduino shields, Raspberry Pi boards and can control relays, LEDs, servos, and motors as an output.

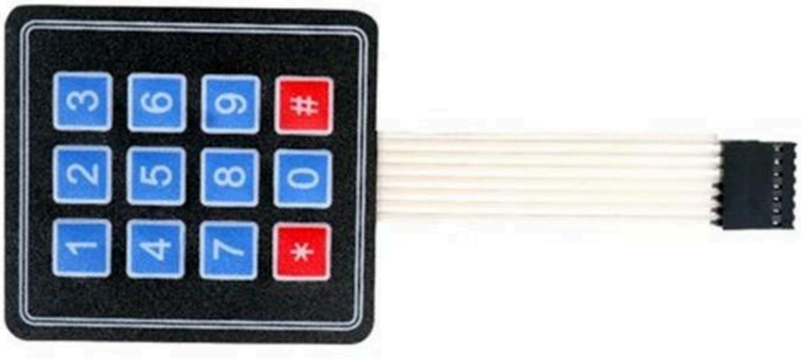
The board can operate on an external supply from 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.



2. 4x3 Keypad

This keypad has 12 buttons, arranged in a telephone-line 3x4 grid.

The keys are connected into a matrix, so you only need 7 microcontroller pins (3-columns and 4-rows) to scan through the pad.

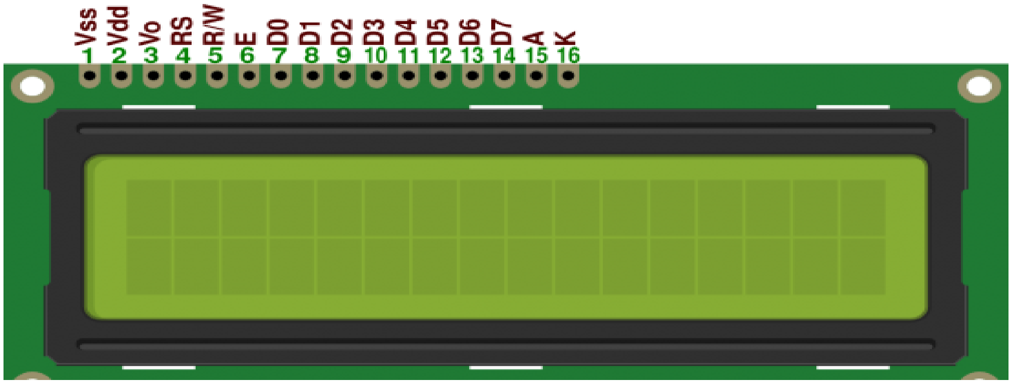


3. 16x2 LCD Display

As the name suggests, it includes 16 Columns & 2 Rows so it can display 32 characters (16×2=32) in total & every character will be made with

5×8 (40) Pixel Dots. So the total pixels within this LCD can be calculated as 32 x 40 otherwise 1280 pixels.

In LCD, instead of electron diffraction at a glass display, a liquid crystal display has a backlight that provides light to each pixel that is arranged in a rectangular network.

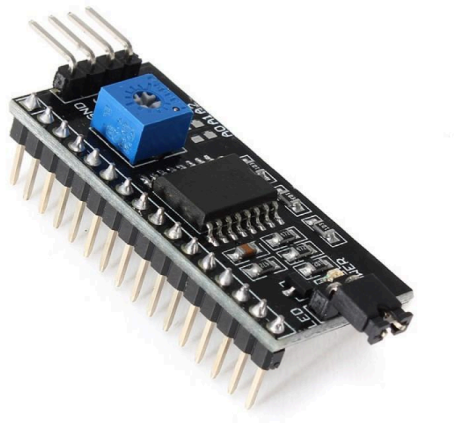


4. I2C Module

The LCD is not directly connected to the Arduino board but is connected using an I2C module.

An I2C Module has an inbuilt PCF8574 I2C chip that converts I2C serial data to parallel data for the LCD display. These modules are currently supplied with a default I2C address of either 0x27 or 0x3F.

The I2C module helps reduce the number of digital pins used to connect it to Arduino.



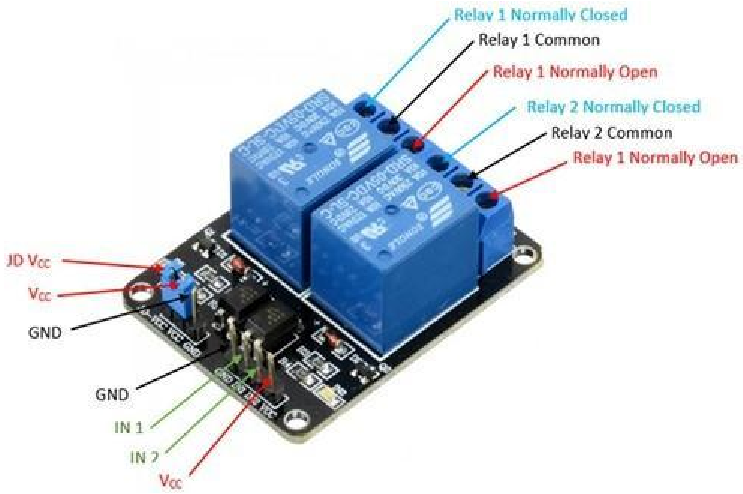
5. 2 Channel Relay Module

The 2 Channel Relay Module is a convenient board which can be used to control high voltage, high current load such as motor, solenoid valves, lamps and AC load. It is designed to interface with microcontrollers such as Arduino, PIC etc.

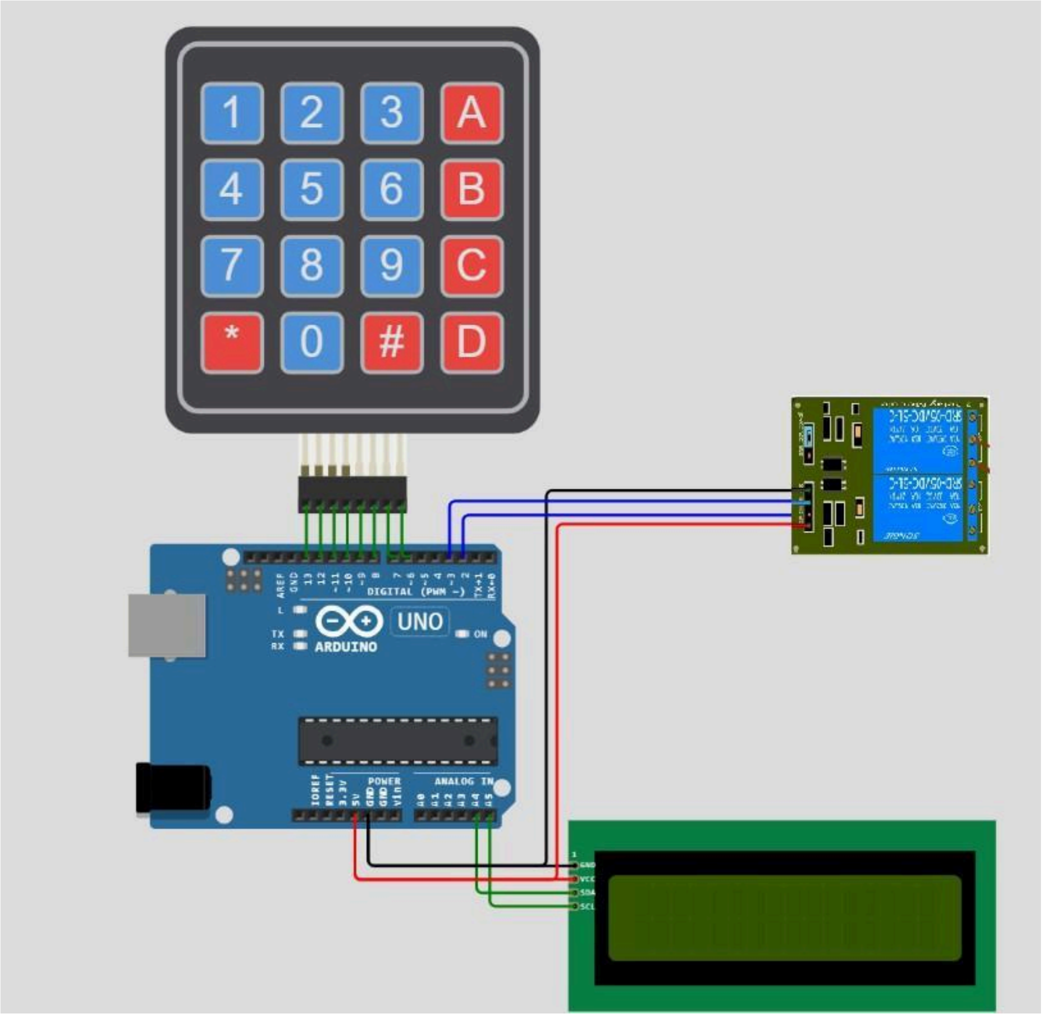
The relay has two outputs-normally open and normally closed (NO and NC). When the IN1 or IN2 pin is connected to ground, NO will be open and NC will be closed, and when IN1 or IN2 is not connected to ground the opposite occurs.

Dual-Channel Relay Module Specifications

- *Supply voltage – 3.75V to 6V*
- *Trigger current – 5mA*
- *Current when relay is active - ~70mA (single), ~140mA (both)*
- *Relay maximum contact voltage – 250VAC, 30VDC*
- *Relay maximum current – 10A*



Circuit Diagram



The 4x3 keypad is connected to the Arduino Uno through digital pins 7 to 13. The LCD display is connected to the Arduino using an I2C module. The connections are

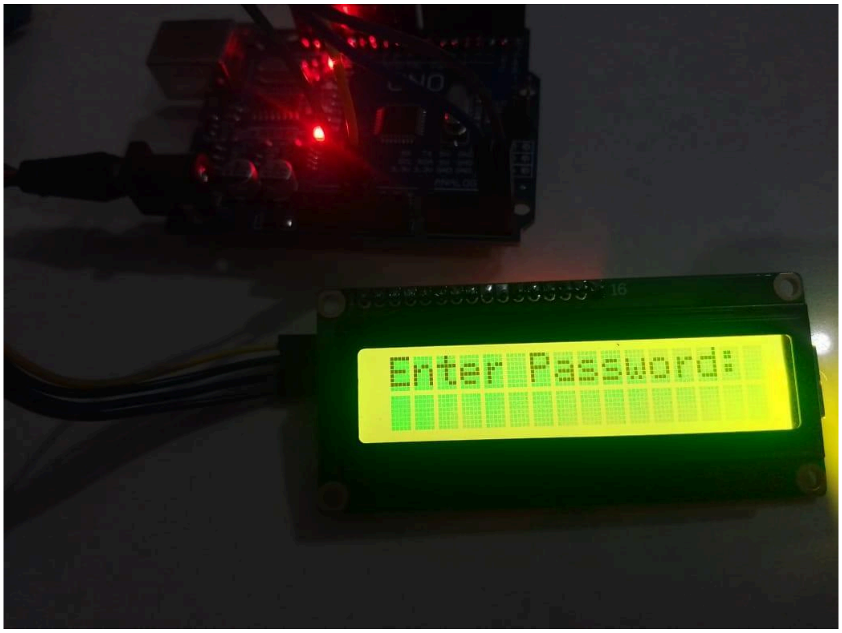
- GND to GND
- VCC to 5V
- SDA to A4(Analog Pin)
- SCL to A5(Analog Pin)

The relay module is connected to Arduino through the following pins:

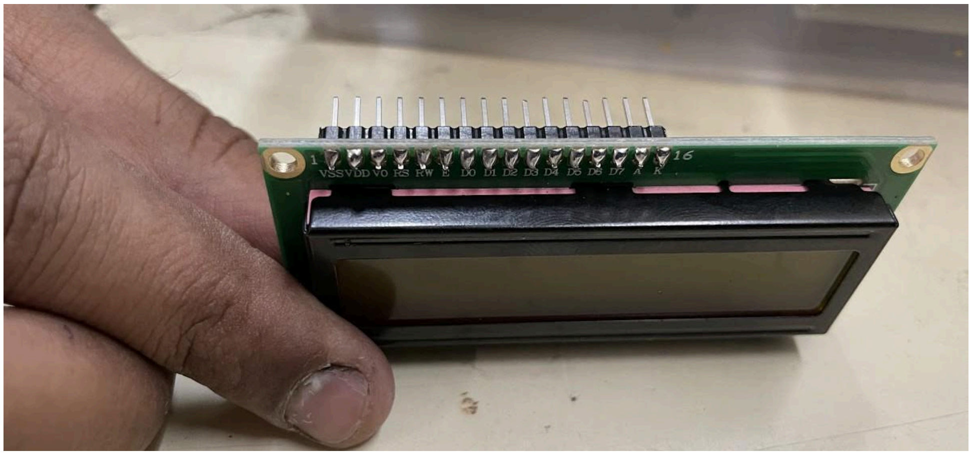
- GND to GND
- VCC to 5V
- IN1 to D3 (Digital Pin)
- IN2 to D2 (Digital Pin)

Progress:

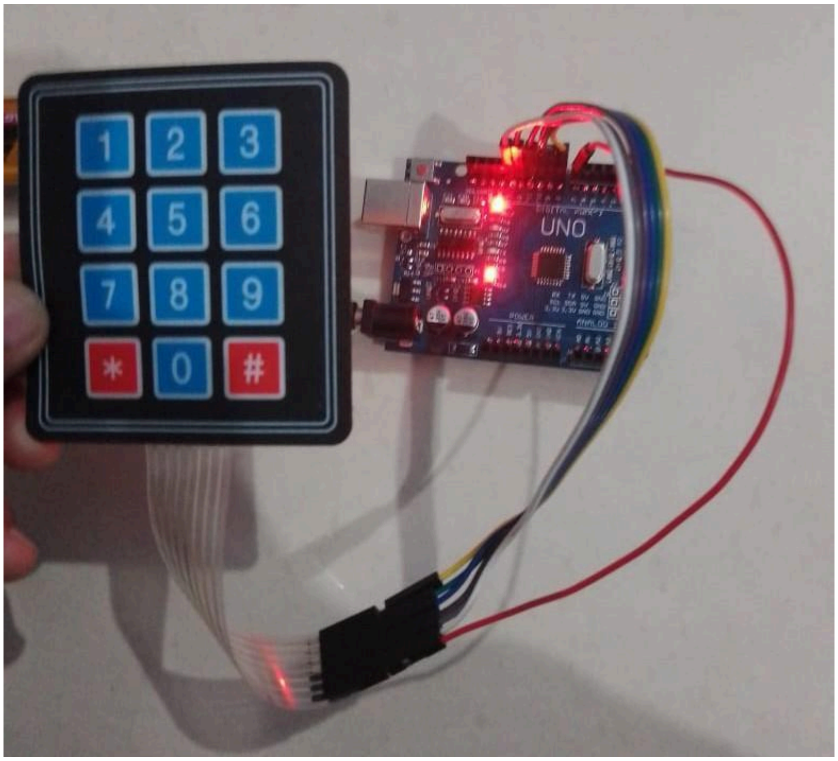
1. The LCD display is connected to the Arduino Uno using digital pins and tested.



2. The number of digital pins proved to be insufficient to connect both the LCD display and the keypad matrix.
3. The LCD display was soldered with an I2C module to reduce the number of digital pins to be connected.

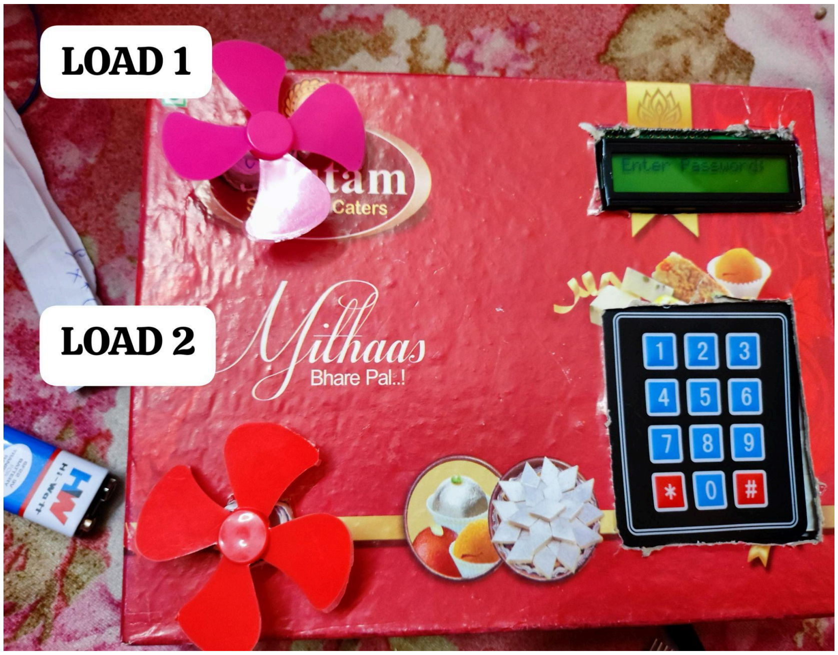


4. The keypad matrix was connected to the Arduino module and was tested.

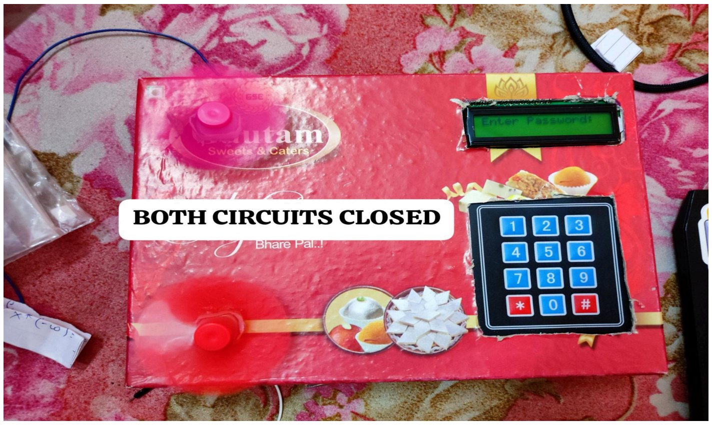


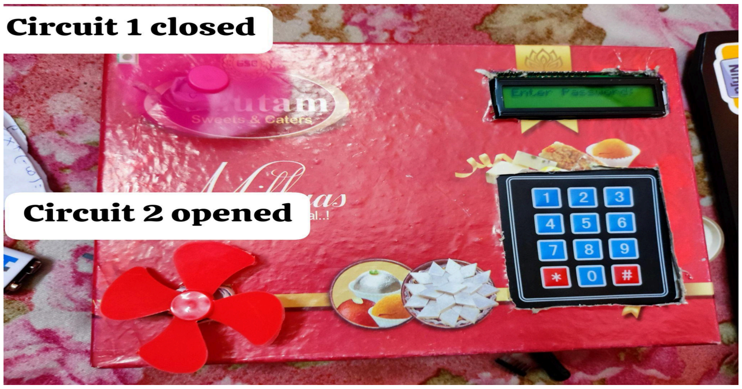
5. The Relay Module was connected to the Arduino module and was tested.
6. The whole circuit was connected on a breadboard and tested.

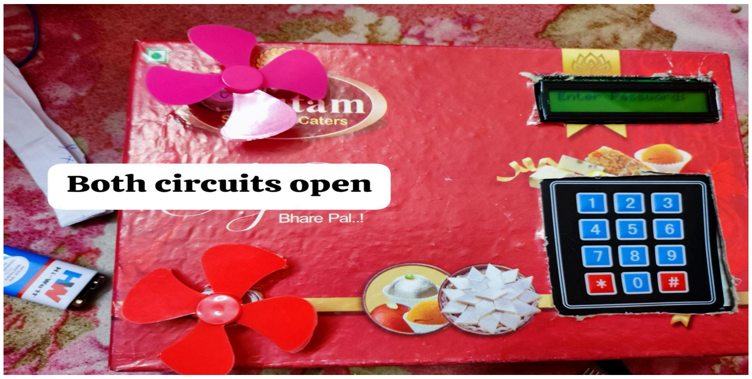
7. The loads were added and the project is ready for use.



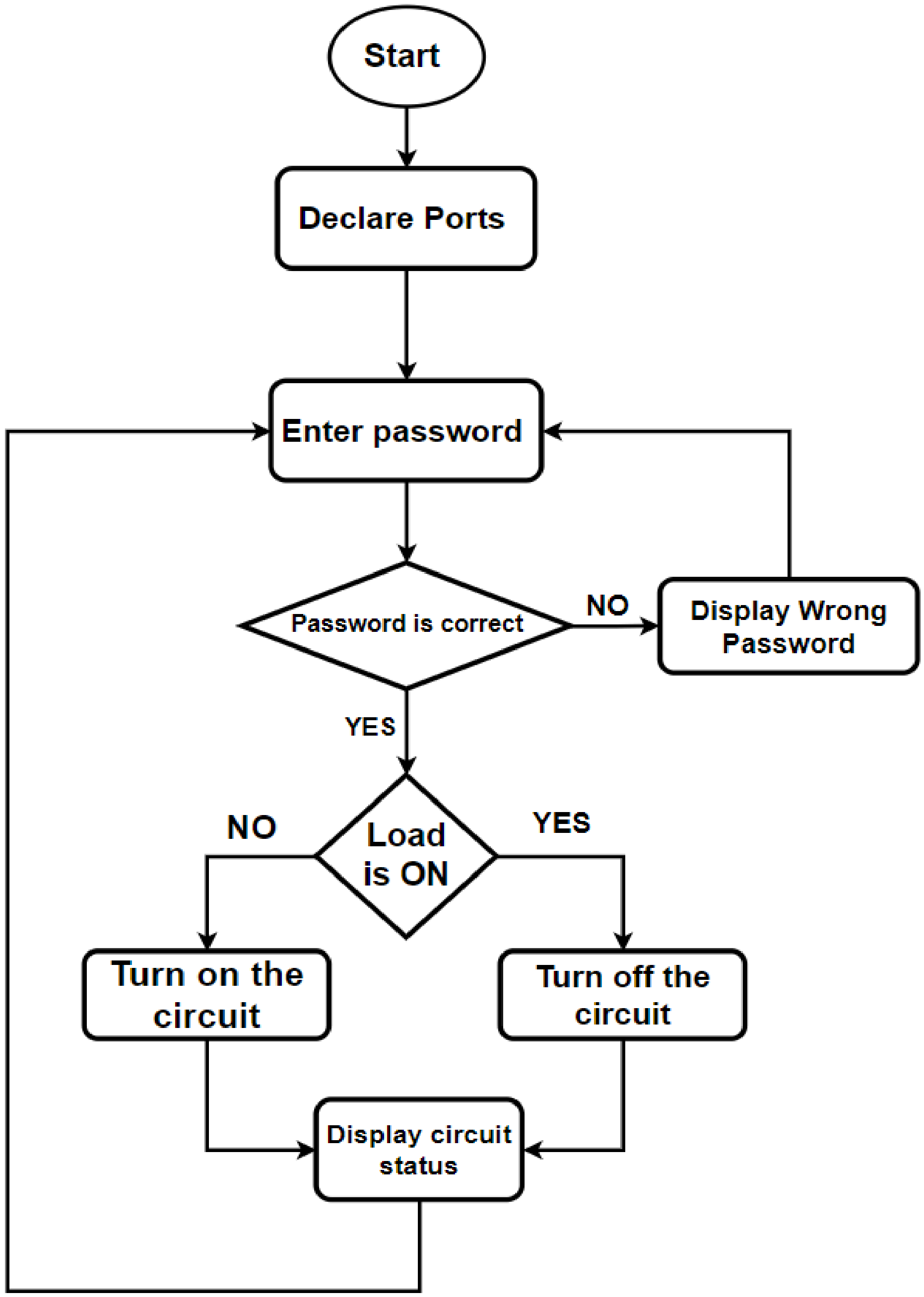
Working:


BOTH CIRCUITS CLOSED
Enter Password!
1 2 3
4 5 6
7 8 9
* 0 #


Circuit 1 closed
Circuit 2 opened
Enter Password!
1 2 3
4 5 6
7 8 9
* 0 #


Both circuits open
Enter Password!
Bhare Pal..!
1 2 3
4 5 6
7 8 9
* 0 #

Flowchart


Start
Declare Ports
Enter password
Password is correct
NO
Display Wrong Password
YES
Load is ON
NO
YES
Turn on the circuit
Turn off the circuit
Display circuit status

Bill of Materials

Component	Quantity	Value
Arduino Uno	1	ATmega328P
16x2 LCD Display	1	
I2C Module	1	
4x3 Keypad	1	
2 Channel Relay Module	1	
Male to male connector pins	3	
Female to Male connector pins	3	
Battery	1	9V
9V battery snap connector with DC Jack Male for Arduino	1	
Zero Board	1	10.5cmx8cm

Arduino Code

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad.h>

const byte ROWS = 4;
const byte COLS = 3;

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

byte rowPins[ROWS] = {2, 3, 4, 5};
byte colPins[COLS] = {6, 7, 8};

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

LiquidCrystal_I2C lcd(0x27, 16, 2);

String enteredPassword = "";
const String relay1Password = "1111"; // Password for relay 1
const String relay2Password = "2222"; // Password for relay 2
const String bothRelaysPassword = "1234"; // Password for both relays
int passwordIndex = 0;
const int relay1Pin = 9; // Relay 1 pin
const int relay2Pin = 10; // Relay 2 pin

void setup() {
  Serial.begin(9600);
  lcd.begin(16, 2);
  pinMode(relay1Pin, OUTPUT);
  pinMode(relay2Pin, OUTPUT);
  lcd.clear();
  lcd.print("Enter Password:");
}

void loop() {
  char key = keypad.getKey();
  if (key != NO_KEY) {
    if (key == '*') {
      // Clear entered password
      enteredPassword = "";
      passwordIndex = 0;
      lcd.clear();
      lcd.print("Enter Password:");
    } else if (key == '#') {
      // Check passwords
      if (enteredPassword == relay1Password) {
        openRelay(relay1Pin);
      } else if (enteredPassword == relay2Password) {
        openRelay(relay2Pin);
      } else if (enteredPassword == bothRelaysPassword) {
        openRelay(relay1Pin);
        openRelay(relay2Pin);
      } else {
        lcd.clear();
        lcd.print("Wrong Password!");
        delay(2000);
      }
      lcd.clear();
      lcd.print("Enter Password:");
      enteredPassword = "";
      passwordIndex = 0;
    } else {
      // Append key to entered password and print it on LCD
      enteredPassword += key;
      lcd.setCursor(passwordIndex, 1);
      lcd.print(key);
      passwordIndex++;
    }
  }
}

void openRelay(int pin) {
  lcd.clear();
  lcd.print("Password OK");
  digitalWrite(pin, LOW); // Open relay circuit
  delay(2000); // Wait for 2 seconds
  digitalWrite(pin, HIGH); // Close relay circuit
  lcd.clear();
  lcd.print("Circuit Closed");
  delay(2000);
}
```