**IRE-206: Microprocessor and Microcontroller System Design Lab**

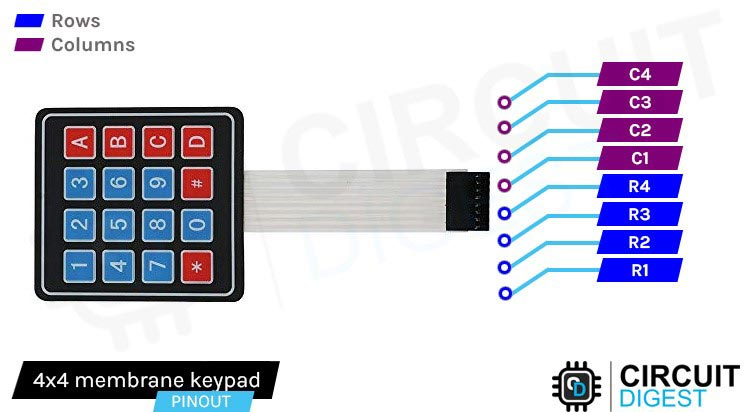
**Experiment-7:** Development of a Password Based Door Lock System.

**Password Based Door Lock System:**

In this experiment, a door lock system is created, utilizing a 4x4 keypad matrix for user input. To unlock the door, users must enter the correct password. The system compares the entered password with a predefined one, and if they match, the system proceeds to open the door latch using a servo motor.

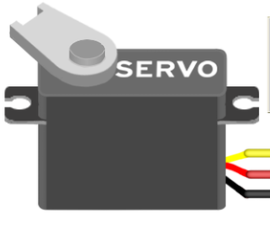
**Connections:**

**Keypad to Arduino:**

****

Connect R1, R2, R3, R4 with D7, D6, D5, D4 and C1, C2, C3, C4 with D3, D2, D1, D0.

**Servo Motor to Arduino:**

****

Connect Black, Red, Yellow to GND, 5V and D11

**Code:**

#include <Servo.h>

#include <Keypad.h>

Servo ServoMotor;

const byte ROWS = 4;

const byte COLS = 4;

char keys[ROWS][COLS] = {

{'1','2','3','A'},

{'4','5','6','B'},

{'7','8','9','C'},

{'\*','0','#','D'}

};

byte rowPins[ROWS] = { 7, 6, 5, 4 };

byte colPins[COLS] = { 3, 2, 1, 0 };

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

int RedpinLock = 12;

int GreenpinUnlock = 13;

char password[3] = {'1', '4', '7'};

char enteredPassword[3];

int position = 0;

void setup() {

Serial.begin(9600);

ServoMotor.attach(11);

pinMode(RedpinLock, OUTPUT);

pinMode(GreenpinUnlock, OUTPUT);

LockedPosition(true);

}

void loop() {

char key = keypad.getKey();

if (key != NO\_KEY) {

if (key == '\*' || key == '#') {

ResetPassword();

LockedPosition(true);

} else {

enteredPassword[position] = key;

Serial.print("\*");

position++;

if (position == 3) {

Serial.println("Password:");

Serial.println(enteredPassword);

CheckPassword();

}

}

}

delay(100);

}

void LockedPosition(int locked) {

if (locked) {

digitalWrite(RedpinLock, HIGH);

digitalWrite(GreenpinUnlock, LOW);

ServoMotor.write(0); // Set servo to locked position (you may need to adjust the angle)

} else {

digitalWrite(RedpinLock, LOW);

digitalWrite(GreenpinUnlock, HIGH);

ServoMotor.write(120); // Set servo to unlocked position (you may need to adjust the angle)

}

}

void ResetPassword() {

position = 0;

memset(enteredPassword, 0, sizeof(enteredPassword)); // Clear entered password

Serial.println("Password reset. Enter a new password.");

}

void CheckPassword() {

if (memcmp(enteredPassword, password, sizeof(password)) == 0) {

Serial.println("Correct Password! Unlocking...");

LockedPosition(false);

} else {

Serial.println("Incorrect Password! Locking...");

ResetPassword();

LockedPosition(true);

}

}