

"به نام یزدان پاک"



دانشگاه صنعتی امیر کبیر
(پلی تکنیک تهران)

گزارش آزمایش ششم

اعضای گروه:

محمد چوپان ۹۸۳۱۱۲۵

محمد سپهر توکلی کرمانی ۹۸۳۱۱۱۱

تاریخ آزمایش : ۱۴۰۰/۰۹/۰۵

آزمایش ۶:

یعقوب برقی رایگان!

هدف پروژه

سیر کردن دانشجوها. یک گذرواژه هفت رقمی (کیبورد) در را باز می‌کند (یک سروو موتور) همه پیچش ها را (تعداد زیادی سروو موتور) به عقب می‌چرخاند تا غذا کار گذاشته شود. همه ی

چراغها (LED) در این مرحله خاموش است. دکمه دوباره در را می‌بندد و چراغها روشن می‌شوند. سپس دانشجوها با زدن شماره‌های دورقمی غذا ها، آن را تحویل می‌گیرند. همه چیز روی LCD نشان داده شود و دکمه پاک و تایید داشته باشد.

قطعات مورد نیاز:

- برد Arduino Mega2560
- دیود نورانی (LED) به مقدار لازم
- تعداد زیادی سروو موتور
- کیبورد
- LCD کاراکتری

شرح آزمایش:

در اینجا با استفاده از برد یک keypad ورودی و یک سروو موتور و lcd این فرایند را شبیه سازی کرده ایم.

که کاربر با ورود به حالت admin ابتدا رمز را وارد میکند در باز میشود با صدای بازز سپس می تواند با فشردن * رمز را تعویض و یا با فشردن - زمان باز بودن در را عوض کند. در نهایت با فشردن دکمه on در بسته شده و دستگاه با به حالت قبلی باز میگردد.

کد ها:

```
#include <Servo.h>
#include <LiquidCrystal.h>
#include <Keypad.h>
char mode='u';
const byte ROWS = 4; //four rows
const byte COLS = 4; //three columns
char keys[ROWS][COLS] = {
  {'7','8','9','/'},
  {'4','5','6','*'},
  {'1','2','3','-'},
  {'0','0','=','+'}
};
byte rowPins[ROWS] = {31, 33, 35, 37}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {23, 25, 27, 29}; //connect to the column pinouts of the keypad

const int rs = 13, en = 12, d4 = 8, d5 = 9, d6 = 10, d7 = 11;

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
Servo myservo; // create servo object to control a servo

const byte doorLedPin = 50, buzzPin = 53, servoPin = 7 ;

byte state = 0;
String prompts[] = {"Pass: ", "Door Open", "ChPass: ", "ChTime: ", "user Mode"};
String inputBuffer, password = "1234";
bool timerOn = false;
int timerTime = 10;
long t0 = millis();
```

```

void open_door() {
    t0 = millis();
    timerOn = true;
    digitalWrite(doorLedPin, HIGH);
    buzz();
    myservo.write(180);
}

```

```

void lock_door() {
    myservo.write(0);
    digitalWrite(doorLedPin, LOW);
    timerOn = false;
    state=0;
    pPrompt();
    buzz();
}

```

```

float calc_time() {
    if (!timerOn) {
        return 0;
    }
    long nowT = millis();
    float diff = (float) timerTime - ((nowT - t0) / 1000.0); }
    if (diff <= 0) {
        lock_door();
        diff = 0;
    }
    return diff;
}

```

```

void setup() {
    Serial.begin(9600);

    myservo.attach(servoPin);
    myservo.write(0);

    lcd.begin(16, 2);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(prompts[state]);
    Serial.println(prompts[state]);

    pinMode(doorLedPin, OUTPUT);
    pinMode(buzzPin, OUTPUT);
}

```

```

void pPrompt() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(prompts[state]);
    inputBuffer = "";
}

```

```

void buzz() {
    digitalWrite(buzzPin, HIGH);
    delay(200);
    digitalWrite(buzzPin, LOW);
}

```

```

void password_entered() {
    lcd.setCursor(0, 0);
    if (inputBuffer == password) {
        lcd.print("Correct password");
        Serial.println("Correct password");
        state = 1;
        open_door();
    } else {
        state = 0;
        lcd.print("Wrong password");
        Serial.println("wrong password");
    }
    delay(50);
}

```

```

void loop() {
    float timer = calc_time();
    lcd.setCursor(0, 1);
    lcd.print("Timer: " + String(timer) + " ");

    char key = keypad.getKey();
    if (key) {
        lcd.print("admin mode");
        Serial.println("admin mode");
        state=0;
        pPrompt();
        lcd.setCursor(prompts[state].length() + inputBuffer.length(), 0);
        lcd.print(key);
        Serial.println(key);
        // if(key=='1') {
        //     mode='a';}
        // if(key=='2') {
        //     mode='u';
        // }
        // if(mode=='a') {
        //     lcd.print("admin mode");
        //     Serial.println("admin mode");
        //     state=0;
        //     pPrompt();
        //     lcd.setCursor(prompts[state].length() + inputBuffer.length(), 0);
        //     lcd.print(key);
        if (key == '=')
        {
            if (!timerOn) {
                // Safe is locked - should enter password
                password_entered();
            } else if (state == 2) {
                // Change password
                password = inputBuffer;
                state = 1;
            }
        }
    }
}

```

```
        state = 1;
    } else if (state == 3) {
        // Change timer
        timerTime = inputBuffer.toInt();
        state = 1;
    }
    pPrompt();
} else if (key == '*' && timerOn) {
    state = 2;
    pPrompt();
} else if (key == '-' && timerOn) {
    state = 3;
    pPrompt();
} else if (key == 'o' && timerOn) {
    lock_door();
} else
    inputBuffer += key;
// }
// else if (mode=='u'){
//     state=4;
//     pPrompt();
//     lcd.setCursor(0, 0);
//     Serial.println("user mode: ");
//     lcd.setCursor(0, 0);
//     lcd.print("enter pro: ");
//     Serial.println("enter pro : ");
//     if(key=='='){
//         lcd.setCursor(0, 0);
//         lcd.print("product "+inputBuffer+" gived");
//         Serial.println("product "+inputBuffer+" gived");
//         pPrompt();
//     }
//     else{
//         inputBuffer+=key;
//     }
// }
```

شماتیک مدار:

