

НП “ИТ КАРИЕРА” | ПМГ “Акад. Боян Петканчин” - гр. Хасково

Модул 8

Въведение в операционни и вградени системи

PhotonSecuritySystem

[Tinkercad](#) | [Github](#)

Изготвил: [Димитър Тодоров](#)

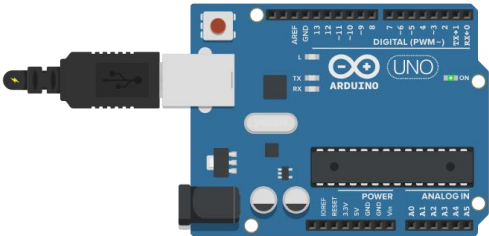
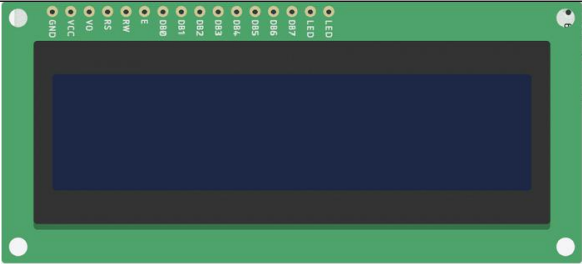

Съдържание:


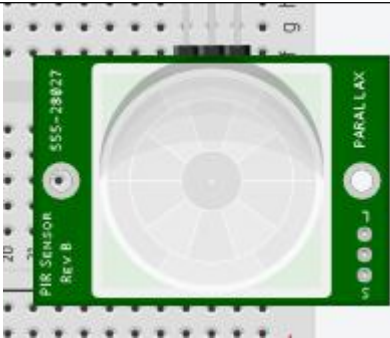

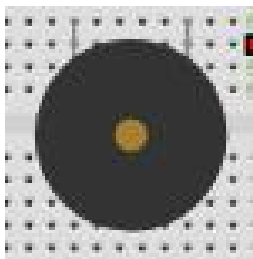
1. [Кратко Описание](#)
2. [Монтажна схема](#)
3. [Source code - описание на функционалността](#)
4. [Заключение](#)

Кратко описание на проекта

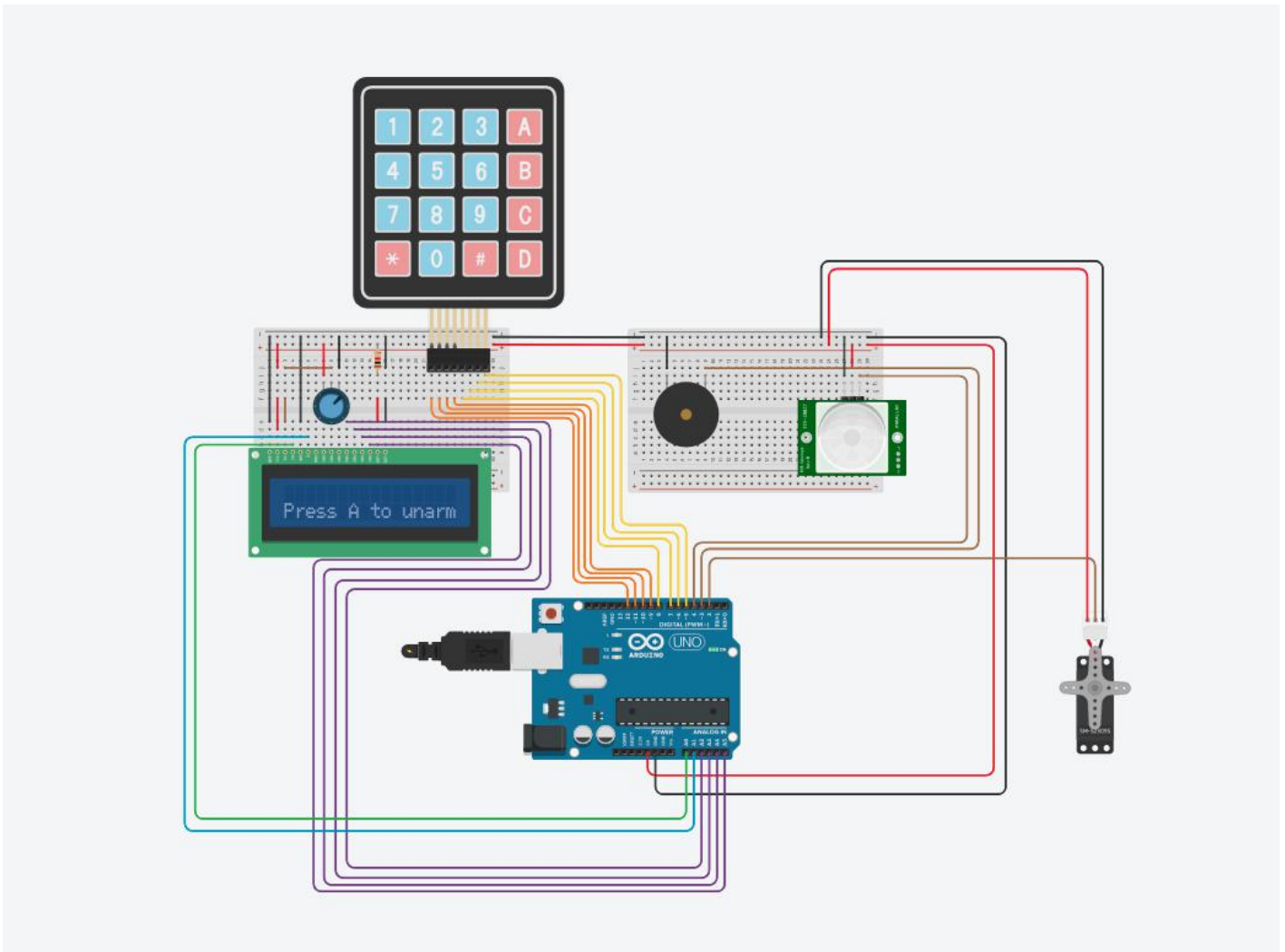
- Охранителна система, базирана около микроконтролера Arduino.
- Система за афтентификация с 4 цифрен PIN код
- Засичане на движение с PIR сензор
- Отключване на ключалка със сервомотор
- Известяване при засечено движение или сгрешен PIN код

Съставни компоненти

Картина	Роля	Свързване(Arduino -> компонент)
Микроконтролер Arduino Uno R3		
	Ролята на микроконтролера е да управлява всички компоненти в нашия проект с помощта на програмен код	Пиновете, към които са свързвани отделните компоненти ще бъдат представени срещу всеки компонент
16x2 LCD Дисплей		
	Ролята на дисплея е улеснение на потребителя при индентификация	GND -> GND 5V -> VCC потенциометър -> Contrast A0 -> Register Select(RS) GND -> Read/Write(RW) A1 -> Enable(E) 1 kΩ резистор -> LED Anode GND -> LED Cathode (4 битов режим) A2 - A5 -> DB4 - DB7
4x4 клавиши		
	Целта на клавишите е при въвеждането на PIN кода при автентификация, задаване на парола и цялостна интеракция със системата	D12 - D9 -> Row 1 - 4 D8 - D5 -> Column 1 - 4

1k Ω потенциометър		
	Целта му е да регулира контраста на дисплея	5V -> Terminal 2 GND -> Terminal 1 Contrast(LCD) <- Wiper
PIR сензор за движение		
	Целта на сензора е засича движение	5V -> Power GND -> Ground D4 -> Signal
SM-S2309S сервомотор		
	Целта на сервомотора е да отключи ключалката при правилен PIN код	5V -> Power GND -> Ground D2 -> Signal
Пиезоговорител		
	Целта му е да извести потребителя, чрез звуков сигнал, за засечено движение или сгрешен PIN код	GND -> Negative D3 -> Positive

Монтажна схема



Описание на функционалността

Source Code:

```
//LCD Library
#include <LiquidCrystal.h>
//Keypad library
#include <Keypad.h>
//Servo motor library
#include <Servo.h>

//The system's password
//Full with Es, because there's no E on the keypad
//And when a new array is initialised, is full of 0
char password[4] = {'E', 'E', 'E', 'E'};
//The system' state
```

```

bool isArmed = true;
//state of the PIR sensor
int PIRSensorState = 0;

//pins
byte ServoPin = 2;
byte BuzzerPin = 3;
byte PIRSensorPin = 4;

//keypad button mapping
char keymap[4][4]=
{
{'1', '2', '3', 'A'},
{'4', '5', '6', 'B'},
{'7', '8', '9', 'C'},
{'*', '0', '#', 'D'}
};

//defining array of the number of the pins for the keypad
byte rowPins[4] = {12, 11, 10, 9}; //Rows
byte colPins[4]= {8, 7, 6, 5}; //Columns

//Defining instance of the keypad class
Keypad myKeypad= Keypad(makeKeymap(keymap), rowPins, colPins, 4, 4);
//Defining instance of the LCD class
LiquidCrystal lcd = LiquidCrystal(A0, A1, A2, A3, A4, A5);
//Defining an instance of the Servo class
Servo servo;

//returns a pressed key
char KeyPadGetKey()
{
//get the currently pressed key
char keypressed = myKeypad.getKey();
//return the pressed key
return keypressed;
}

//checking if the password is correct
bool CheckPasswd(char passToCheck[4], char password[4])
{
for(byte i = 0; i < 4; i++)
{
if(passToCheck[i] != password[i])
{
//returns false if a different character was detected
return false;
}
}
}
//returns true otherwise
return true;

```

```

}

//setting up the password
void SetUp()
{
    //Because E is an indicator that there isn't a password already set up
    if(password[0] == 'E')
    {
        lcd.print("Register a pass: ");
        lcd.setCursor(0,1);
        byte i = 0;
        while (i<4)
        {
            //getting the input from the keypad
            char key = KeypadGetKey();
            if(key != NO_KEY)
            {
                lcd.print(key);
                password[i] = key;
                i++;
            }
        }
        lcd.clear();
    }
}

//Unarming function
void Unarm()
{
    lcd.clear();
    lcd.print("Enter Password: ");
    lcd.setCursor(0, 1);
    //password to put in a temporary pass to be checked later
    char passToCheck[4];
    byte i = 0;
    while(i < 4)
    {
        //getting the input from the keypad
        char key = KeypadGetKey();
        if(key != NO_KEY)
        {
            lcd.print(key);
            passToCheck[i] = key;
            i++;
        }
    }
    //check if the password is correct
    if(CheckPasswd(passToCheck, password) == true)
    {
        //changes system state if it is correct
    }
}

```

```

lcd.clear();
lcd.print("Granted");
delay(500);
isArmed = false;
servo.write(270);
}
else
{
lcd.clear();
lcd.print("Denied");
tone(BuzzerPin, 1000, 0);
delay(5000);
noTone(BuzzerPin);
}
}

//Arming Function
void Arm()
{
lcd.setCursor(0,1);
lcd.print("Press A to arm");
//getting the input from the keypad
char key = KeyPadGetKey();
if(key == 'A')
{
lcd.clear();
lcd.print("Arming...");
delay(5000);
lcd.clear();
//changes the system state
isArmed = true;
servo.write(90);
}
}

//Ring the alarm if Movement is detected
void AlarmIFMovementDetected()
{
PIRSensorState = digitalRead(PIRSensorPin);
if (PIRSensorState == HIGH)
{
tone(BuzzerPin, 1000, 0);
}
else
{
lcd.setCursor(0,0);
for(byte i = 0; i < 16; i++)
{
lcd.print(" ");
}
}
}

```



```

noTone(BuzzerPin);
}
}

void setup()
{
  lcd.begin(16,2);
  servo.attach(ServoPin);
  SetUp();
}

void loop()
{
  while(isArmed)
  {
    lcd.setCursor(0, 1);
    lcd.print("Press A to unarm");
    if(KeyPadGetKey() == 'A')
    {
      Unarm();
    }
    AlarmIFMovementDetected();
    break;
  }
  while(!isArmed)
  {
    delay(600);
    lcd.clear();
    Arm();
  }
}

```

Заклучение

Този проект може да намира приложение в домашната сигурност. Благодарение на звуковия сигнал, ще можем да бъдем известени, при евентуални нежелани посетители. Сервомотора може да се монтира на всякакъв тип врата с лека модификация на ключалката.