

# Проект 1 по ППИА, температурен сензор и RGB диода со ардуино уно

Изработено од Ристе Јовановски 81/2022 и Филип Јовески 82/2022

## 1. Вовед

Целта на овој проект е да се дизајнира и програмира систем со помош на Arduino кој може да детектира промени во температурата во просторијата и сигнализира температурата со RGB LED. Системот е наменет да работи во просторија со површина од приближно 10 квадратни метри, каде што температурата се одржува помеѓу 20°C до 30°C со барање за точност од 1°C. RGB LED треба да ја менува бојата во зависност од промената на температурата во просторијата.

Овој документ дава детално објаснување на проектот, вклучувајќи ги употребените материјали, дијаграмот на колото и кодот што беше развиен за имплементација на системот. Документот е поделен на неколку делови, од кои секој покрива специфичен аспект на проектот. Овој вовед дава преглед на целите на проектот и ја поставува основата за остатокот од документот. Материјалите и опремата што скористат во овој проект се опишани во следниот дел.

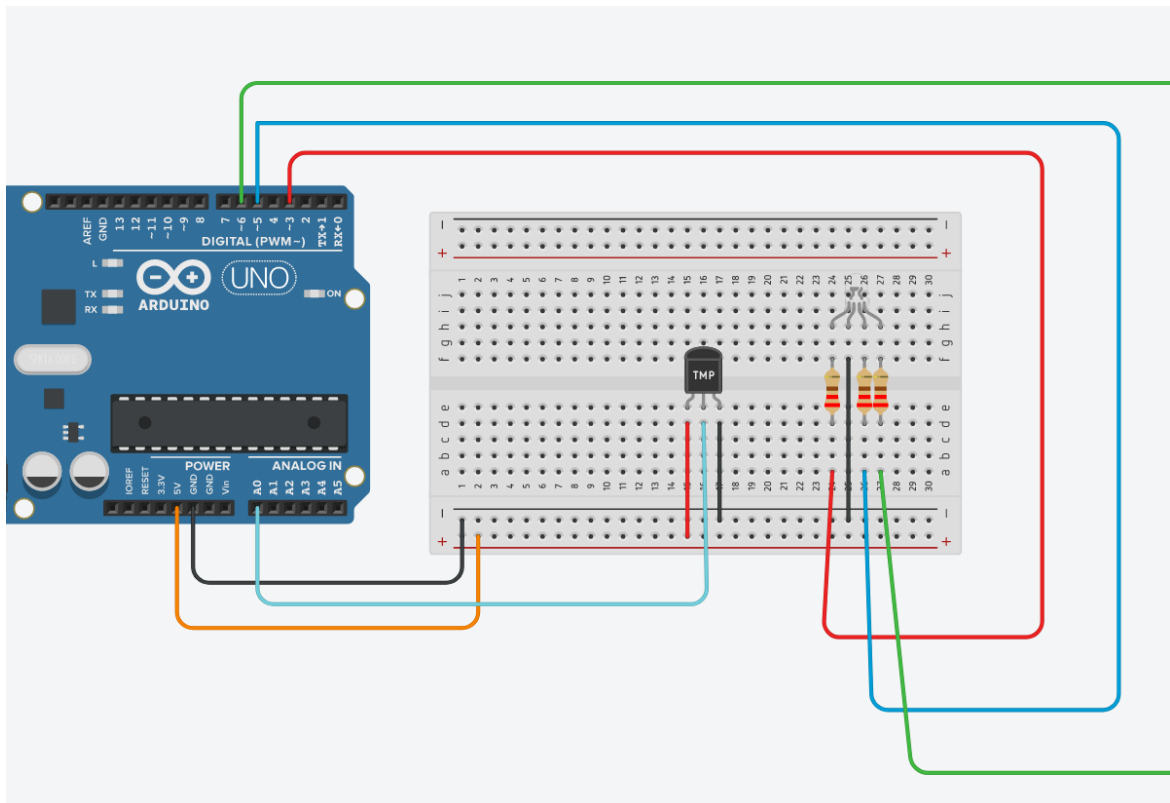
## 2. Материјали и опрема

Следниве материјали и опрема беа користени во овој проект:

- Arduino Uno плоча
- Сензор за температура LM35
- RGB LED
- 3 Отпорници
- Даска за поврзување на елементите
- Жици
- USB-кабел за поврзување на Arduino со компјутер

## 3. Дијаграм на колото

За креирање и тестирање на кодот се користеше симулаторот Tinkercad. LM35 се користи како сензор за температура. Поврзан е на аналоген влез A0 на плочката Arduino, а нормално поврзан е и на напонскиот извор од Ардуино и со заземјување. RGB LED диодата е поврзана со три дигитални пинови на плочката Arduino (пинови 3, 5 и 6) преку отпорници и еден од нејзините терминали е поврзан со заземјување.



## 4. Објаснување на кодот

Кодот на Arduino за овој проект е напишан во C++. Кодот ја чита температурата од температурниот сензор LM35, ја прикажува на екранот за текст (опционално) и ја сигнализира температурата користејќи RGB LED. Кодот е поделен на неколку делови, од кои секој е детално објаснет подолу:

### 4.1 Иницијализирање на променливи:

```
#define red 3
#define blue 5
#define green 6

int temp_adc;
int temp;
int redValue = 0
int blueValue = 0
int greenValue = 0
```

Првиот дел од кодот ги дефинира пиновите за RGB LED, како и некои променливи што ќе се користат подоцна во кодот. temp\_adc се користи за складирање на вредноста на температурата што ја чита температурниот сензор LM35. temp се користи за складирање на конвертираната температурна вредност. redValue, blueValue и greenValue се користат за контрола на осветленоста на RGB LED за секоја боја.

### 4.2 Поставување на Ардуиното:

```
void setup() {
  pinMode(red, OUTPUT);
  pinMode(blue, OUTPUT);
  pinMode(green, OUTPUT);

  Serial.begin(9600);
  pinMode(A0, INPUT);

  digitalWrite(red, LOW);
  digitalWrite(blue, LOW);
  digitalWrite(green, LOW);
}
```

Функцијата `setup()` ги иницијализира пиновите за RGB LED како излезни пинови и ја поставува пинот за сензорот LM35 како влезен пин. Функцијата исто така ја поставува сервиската комуникација за приказот на текст (опционално). Функциите `digitalWrite()` ја поставуваат почетната состојба на RGB LED на исклучено.

:

### 4.3 Читање на температурата:

```
temp_adc = analogRead(A0);  
temp = (temp_adc*(5000/1024))/10;  
}
```

Во `loop()` функцијата, кодот ја чита аналогната вредност од температурниот сензор LM35 користејќи ја функцијата `analogRead()`. Аналогната вредност се претвора во температура во степени Целзиусови користејќи ја формулата: `temp = (temp_adc*(5000/1024))/10`. Оваа формула ја претвора аналогната вредност во миливолти, а потоа во степени Целзиусови.

### 4.4 Сигнализирање на температурата:

```
int minTemp = 20;  
int maxTemp = 30;  
  
if (temp < minTemp){  
    if (redValue > 0){  
        redValue -= 1;  
    }  
    if (greenValue > 0){  
        greenValue -= 1;  
    }  
    if (blueValue < 255){  
        blueValue += 1;  
    }  
}  
else if (temp > maxTemp){  
    if (blueValue > 0){  
        blueValue -= 1;  
    }  
    if (greenValue > 0){  
        greenValue -= 1;  
    }  
    if (redValue < 255){
```

```

        redValue += 1;
    }
}
else{
    if (redValue > 0){
        redValue -= 1;
    }
    if (blueValue > 0){
        blueValue -= 1;
    }
    if (greenValue < 255){
        greenValue += 1;
    }
}

analogWrite(red, redValue);
analogWrite(blue, blueValue);
analogWrite(green, greenValue);

```

Кодот користи изјава if-else за да ја провери вредноста на температурата и соодветно да ја прилагоди осветленоста на секоја боја на RGB LED. Ако температурата е под минималната температура (20 °C), кодот ги намалува црвените и зелените вредности и ја зголемува сината вредност за да произведе ладна боја (сина или сино-зелена). Ако температурата е над максималната температура (30 °C), кодот го намалува синото и зелената вредност и ја зголемува црвената вредност за да добие топла боја (црвена). Ако температурата е во саканиот опсег, кодот ги намалува црвените и сините вредности и ја зголемува зелената вредност за да произведе неутрална боја (зелена).

Функцијата analogWrite се користи за прилагодување на осветленоста на секоја боја со пренесување вредност помеѓу 0 и 255 на соодветните пинови на LED. Колку е поголема вредноста, толку посветла ќе биде LED. Функцијата за одложување се користи за забавување на јамката и спречување на брзи промени во LED боите, што го олеснува набљудувањето на промените на бојата и го намалува оптоварувањето на очите.

Конечно, кодот ја користи функцијата Serial.print за да ја прикаже вредноста на температурата што ја чита сензорот LM35 во сервискиот монитор на Arduino IDE. Оваа функција е корисна за тестирање и дебагирање на кодот, бидејќи му овозможува на програмерот да ги следи отчитувањата на температурата во реално време и да прави прилагодувања доколку е потребно.