

Proiect Achizitii de Date si Instrumentatie
Virtuala
Automated Cooling System

Dinu Ion-Irinel
Grupa 325CA, CTI, UPB
ion_irinel.dinu@stud.acs.upb.ro

June 1, 2022

Contents

1	Introducere	3
1.1	Descriere generala	3
2	Componentele circuitului implementat	3
2.1	Placa Arduino Uno R3	3
2.2	Senzor de temperatura	4
2.3	LCD Display 16x2	4
2.4	BreadBoard	5
2.5	Neon	5
2.6	Motor	6
2.7	SlideSwitch	6
2.8	Becuri	7
3	Descrierea functionalitatii	7
3.1	Schema Circuit	8
3.2	Implementare	8
3.3	Interpretarea datelor	9
4	Bibliografie	15

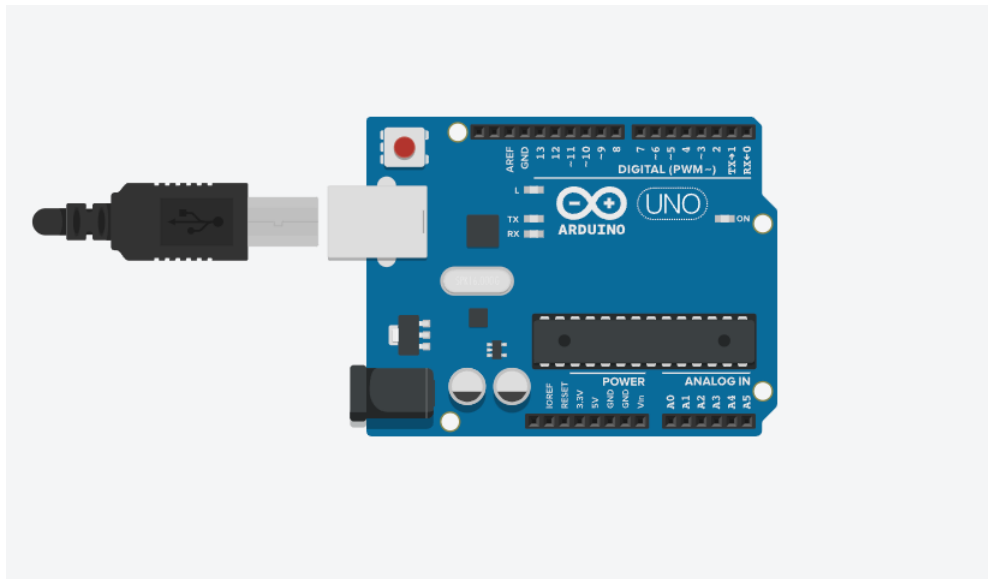
1 Introducere

1.1 Descriere generala

În cadrul proiectului se dorește realizarea unui Automated Cooler System, un cooler care folosește două motoare de racire acționate cu ajutorul unui senzor de temperatură și a unei plăcuțe Arduino. În funcție de temperatură preluată din partea senzor-ului, motoarele pornesc cu o anumită durată. De asemenea se folosesc și trei led-uri care luminează în funcție de valoarea temperaturii. Ambele motoare sunt înconjurate de o bandă de led, care simulează un joc de culori.

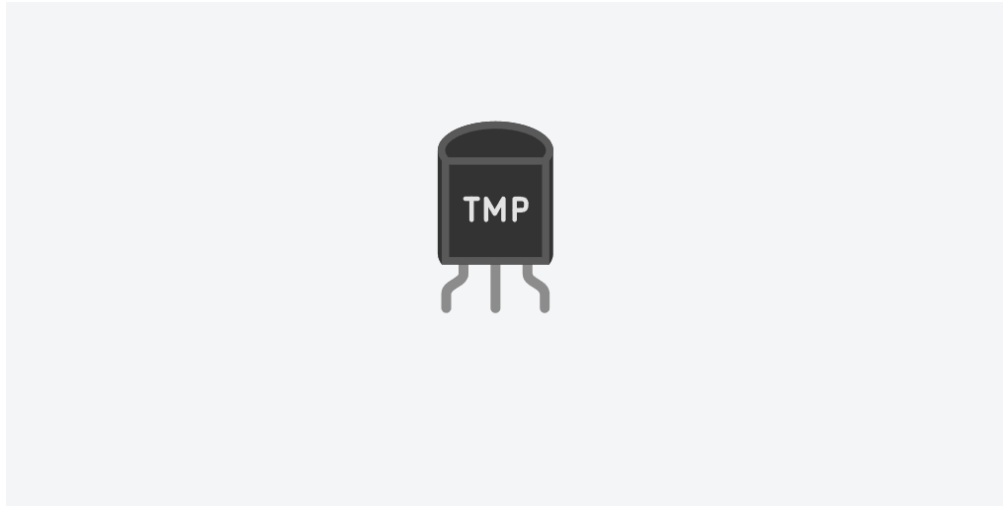
2 Componentele circuitului implementat

2.1 Placă Arduino Uno R3



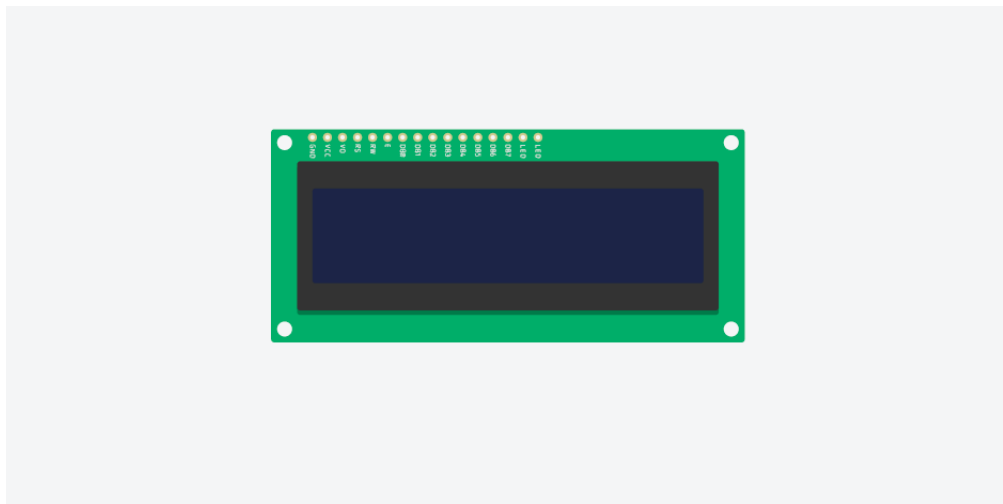
Această componentă reprezintă baza circuitului, este componentat în jurul careia este construit întreg circuitul, alimentând și conectând toate celelalte componente pentru a îndeplini funcționalitatea dorită.

2.2 Senzor de temperatura



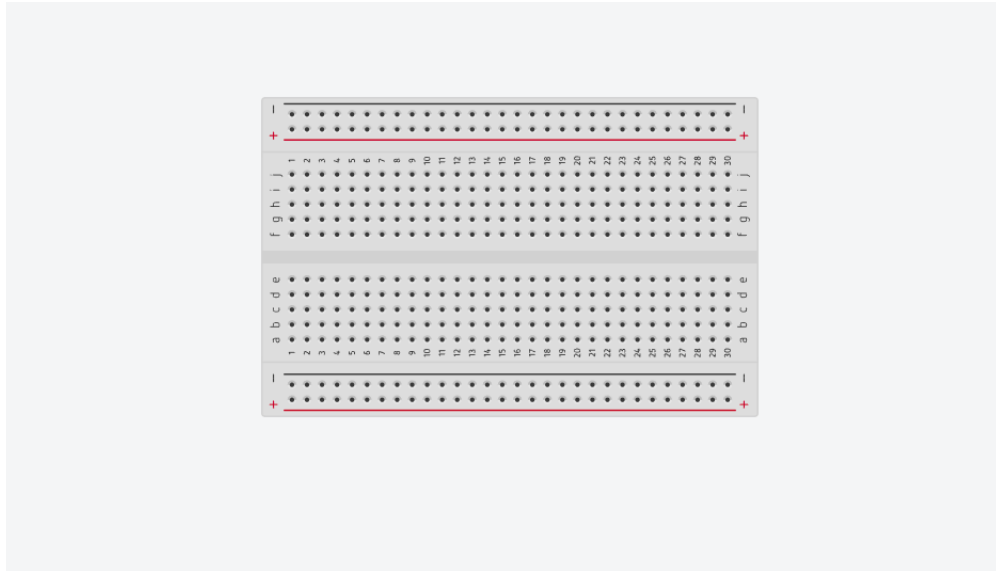
Acesta reprezinta un senzor de temperatura, TMP36 cu ajutorul caruia este preluata temperatura care este transmisa mai departe catre celelalte componente

2.3 LCD Display 16x2



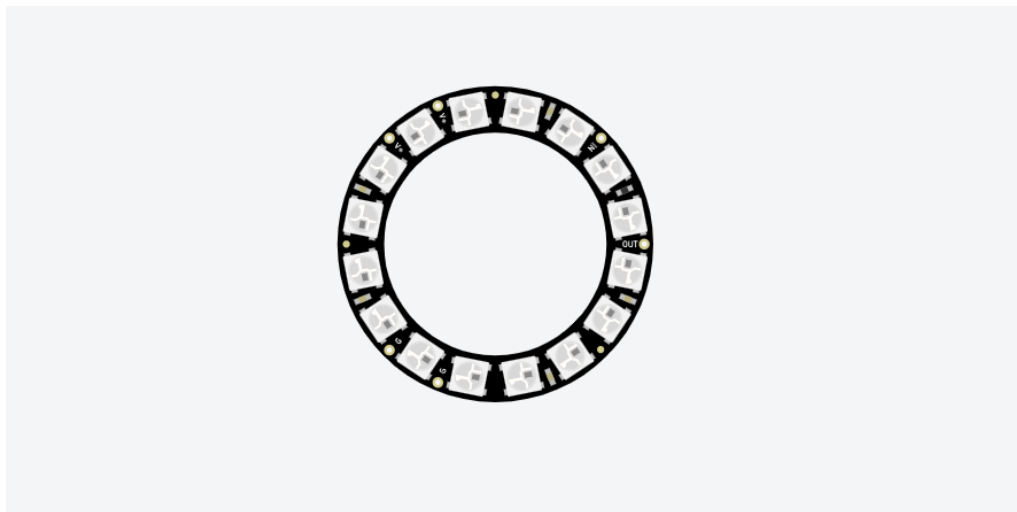
Acesta reprezinta un LCD 16 x 2, un afisaj electronic cu ajutorul caruia sunt afisate mesaje, precum temperatura, turatia motoarelor etc

2.4 BreadBoard



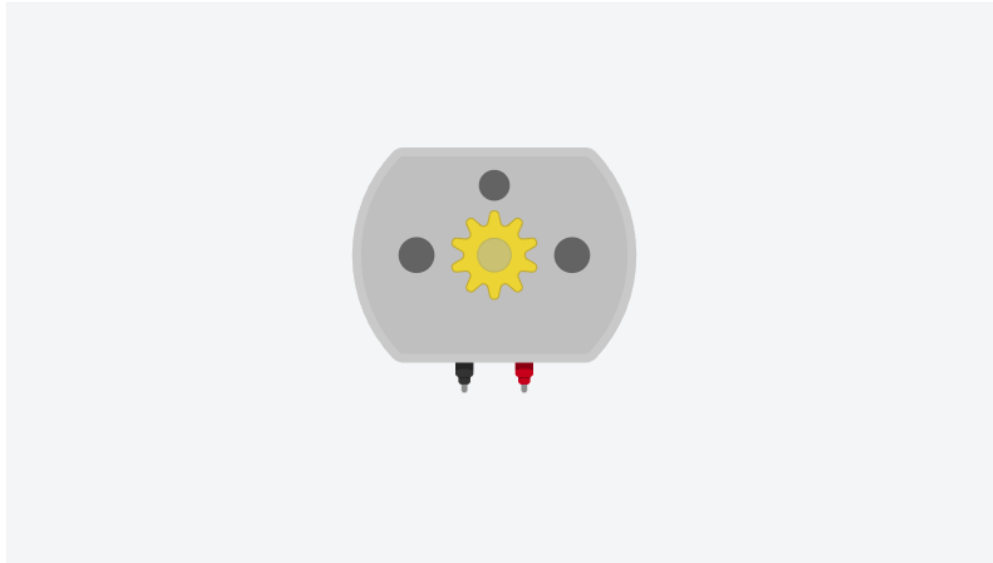
Acesta reprezinta o placa de breadboard, un dispozitiv cu ajutorul caruia se poate realiza usor legatura firelor dintre componentele circuitului.

2.5 Neon



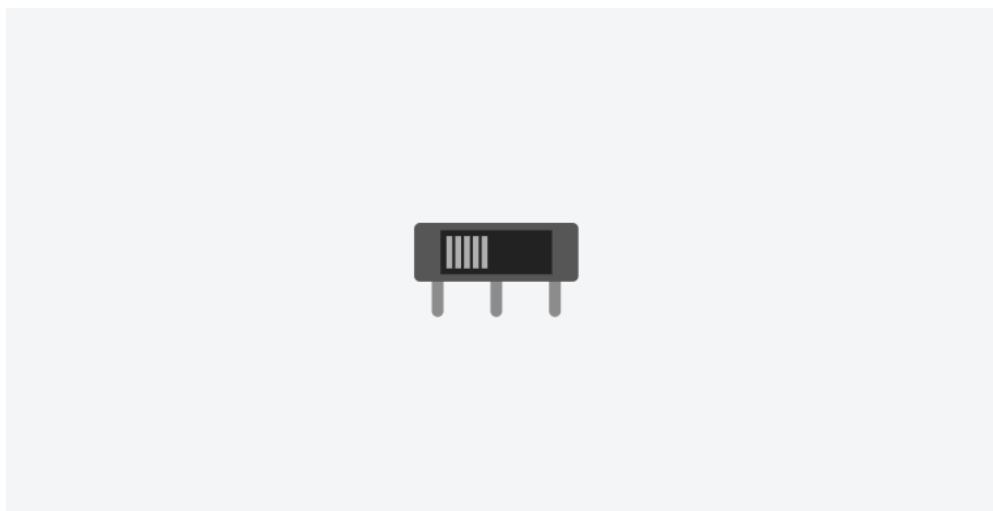
Aceasta componenta reprezinta o banda de lumini cu diferite jocuri de culori, care va inconjura motoarele cooler-ului. Banda de lumini foloseste culori RGB.

2.6 Motor



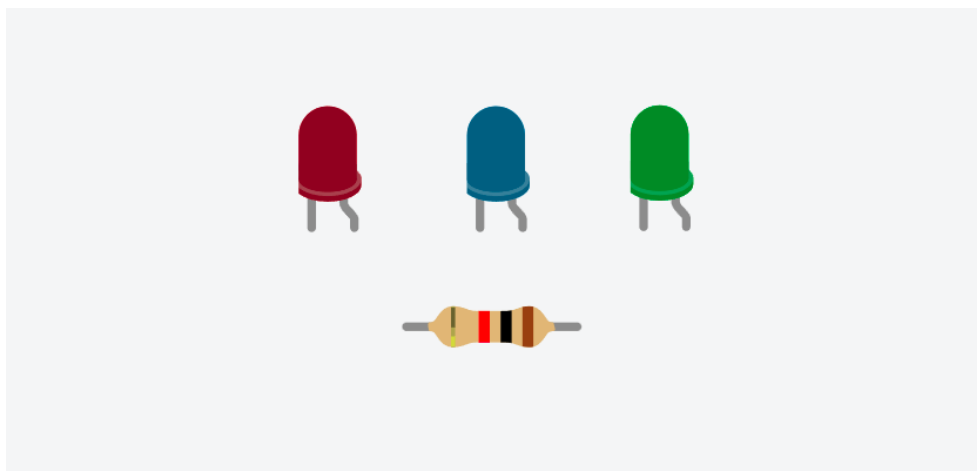
Acesta reprezinta un DC Motor, o componenta de baza in cadrul circuitului. Motoarele pornesc cu o anumita rotatie pe baza temperaturii preluate din cadrul senzorului.

2.7 SlideSwitch



Acesta reprezinta un buton de tip slideswitch cu ajutorul caruia se poate porni sau opri alimentarea afisajului electronic.

2.8 Becuri



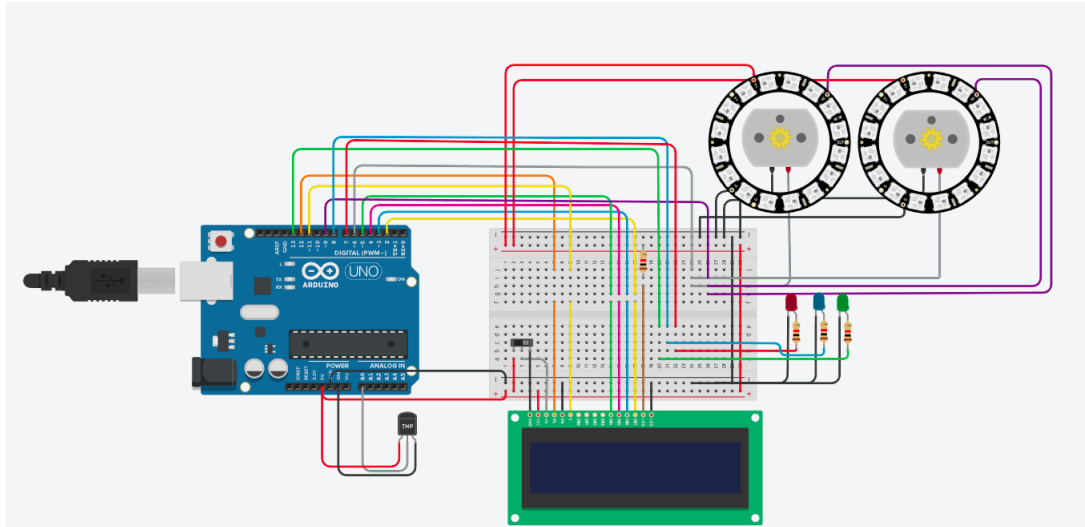
Aceasta reprezinta o serie de trei becuri, care se aprind pe rand in functie de valoarea temperaturii. Becul rosu va indica ca temperatura preluata de la senzor este foarte mare. Pentru a limita curentul/tensiunea pentru cele trei becuri s-au inseriat fiecare dintre acestea cu o rezistenta.

3 Descrierea functionalitatii

Anumite componente din interiorul unui computer produc caldura. Caldura generata poate scadea performantele computer-ului sau uneori chiar sa il defecteze. Calculatoarele de ultima generatie folosesc componente mai puternice pentru a creste performantele. Cu cat performantele cresc, cu atat si caldura generata creste. Pentru a scadea temperatura din interiorul unitatii centrale, a fost creat cooler-ul, un sistem de racire actionat pe baza de motoare.

Cooler-ul implementat in cadrul proiectului utilizeaza doua motoare de racire. Se preia la o anumita perioada de timp valoarea temperaturii indicate de senzorul de temperatura TMP36. Dupa citire, se converteste temperatura in grade Celsius, si se afiseaza un mesaj corespunzator folosin afisajul electronic. De asemenea, in functie de temperatura (20 - 40 - 60 ..) motoarele pornesc cu o anumita viteza, care este afisata in cadrul afisajului, iar becurile lumineaza in mod similar.

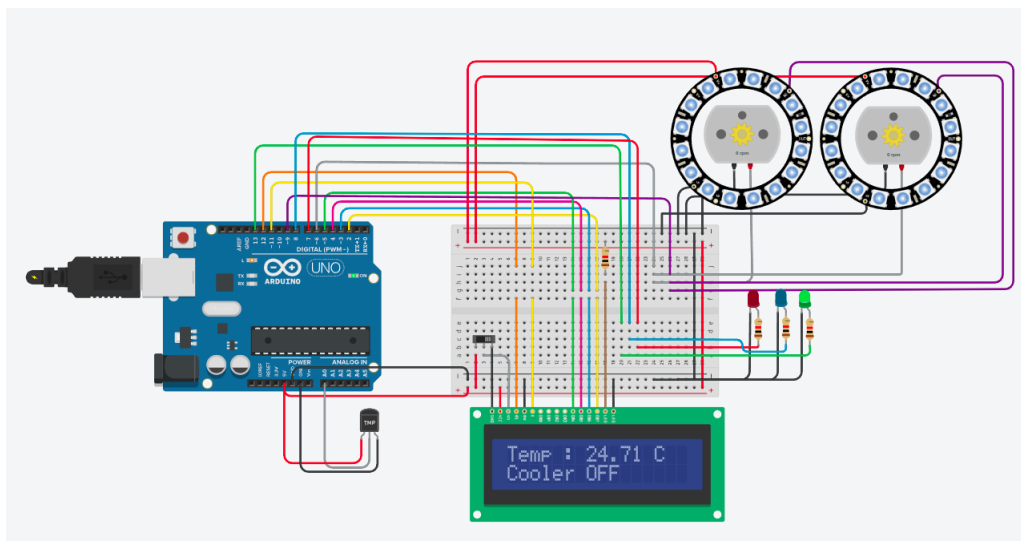
3.1 Schema Circuit



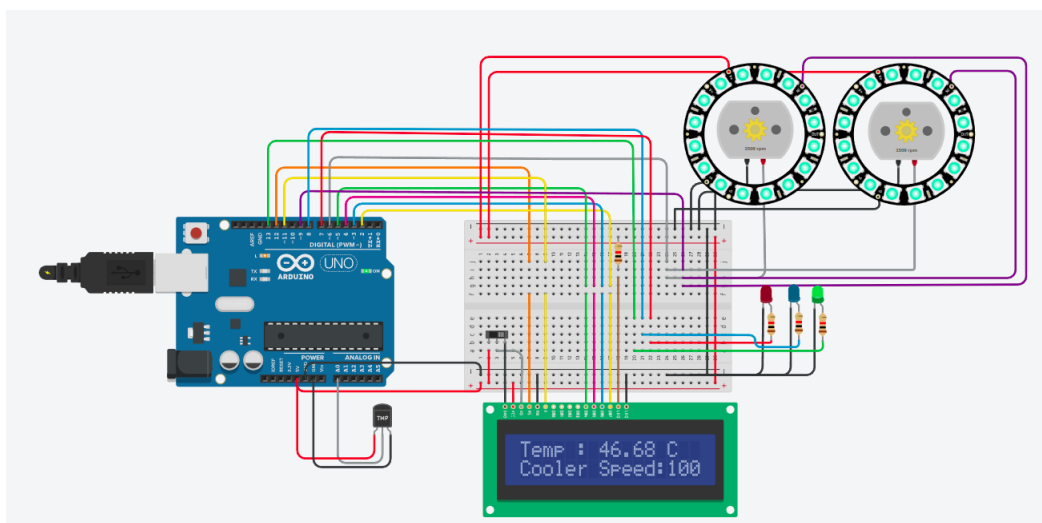
3.2 Implementare

- Pentru implementare circuitului am folosit platforma Tinkercad
- Pentru realizarea legaturilor firelor am utilizat un breadboard si am incercat sa utilizez o conventie pentru culori, rosu pentru conectarea la In(5V) si negru pentru conectarea la Ground
- Componentele sunt alimentate de la placa Arduino

3.3 Interpretarea datelor

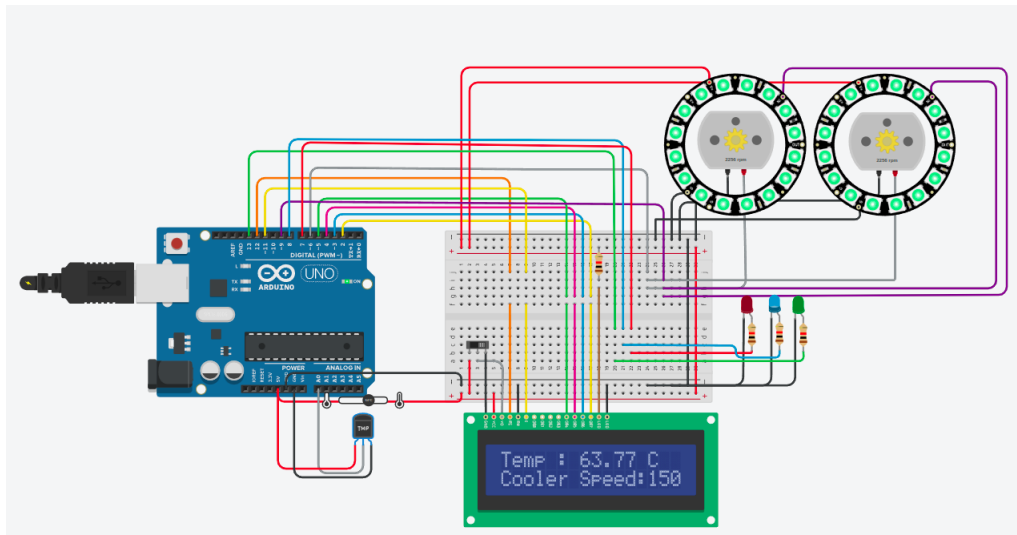


La pornirea circuitului, temperatura indicata de senzor este de 24.71 grade Celsius. Pentru o valoare a temperaturii mai mica de 40 de grade Celsius, motoarele sunt in stare de OFF, iar becul verde lumineaza, semnala o valoare scazuta a temperaturii. Se observa iluminarea led a motoarelor realizata cu ajutorul neo ring-ului.

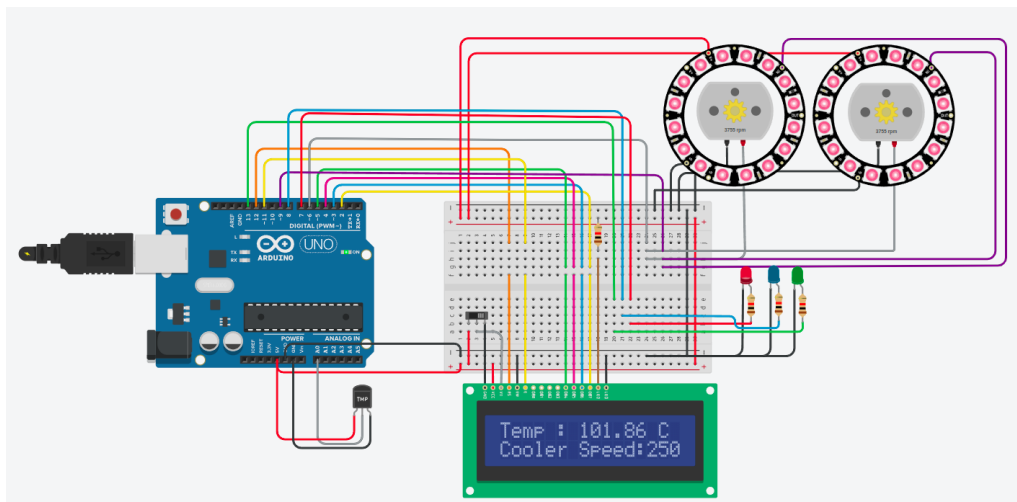


Schimband valoarea temperaturii folosind senzor-ul la 46.7 grade Celsius se observa afisarea corecta a valorii in cadrul afisajului. Becul verde lumineaza in

continuare, inasa cele doua motoare pornesc cu o turatie, cu un speed de 100, deoarece valoarea temperaturii este cuprinsa intre 40 si 60 de grade Celsius.

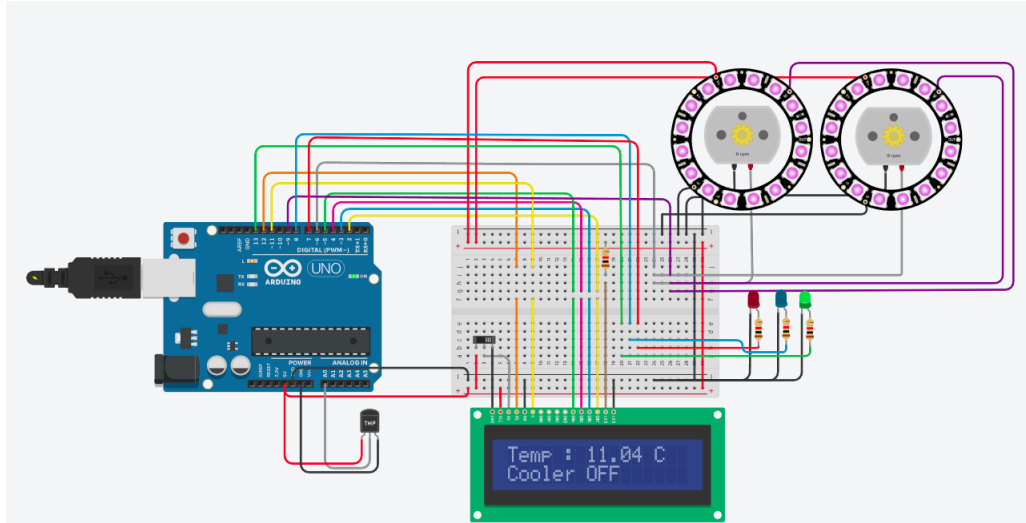


In momentul in care in care temperatura indicata de senzor are o valorea cuprinsa intre 60 si 80 de grade Celsius se observa noua functionalitate motoarele, al caror speed creste la valoarea 150, iar becul verde inceteaza sa lumineze in detrimentul celui albastru care indica o valoare medie a temperaturii.



La o valorea a temperaturii cuprinsa intre 80 si 125 se afiseaza un mesaj core-spunzator cu ajutorul afsajului iar motoarele se pornesc la turatie maxima, cu

un speed de 250. Becul albastru se stinge, iar cel rosu lumineaza semnalad o valoare foarte ridicata a temperaturii.



Daca temperatura indicata de senzor revine la o valoare de 12 grade Celsius se observa oprirea motoarelor, afisarea mesajului corespunzator cu ajutorul afisajului. Becul verde lumineaza din nou indicand o valoare scazuta a temperaturii.

```

#include <Adafruit_NeoPixel.h>
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

#include <Adafruit_NeoPixel.h>

#define PIN 9 // Pin
#define NLEDS 32 // Number of led in total strips

Adafruit_NeoPixel strip = Adafruit_NeoPixel(NLEDS, PIN, NEO_GRB + NEO_KHZ800);

float value;
float temperature;
int motor = 6;
int speed = 0;
int redColor = 0;
int greenColor = 0;
int blueColor = 0;
int fanCoolerPin = 6;

void setup() {
  // cooler
  pinMode(fanCoolerPin, OUTPUT);
  // led red
  pinMode(7, OUTPUT);
  // leg blue
  pinMode(8, OUTPUT);
  // led green
  pinMode(13, OUTPUT);
  strip.begin();
  randomSeed(analogRead(0));

  lcd.begin(16, 2);
  pinMode(motor, OUTPUT);

  analogWrite(motor, speed);
  lcd.clear();
  speed = 0;
  analogWrite(motor, speed);
  delay(1000);
  lcd.setCursor(3, 0);
  lcd.print("Temperature");
  lcd.setCursor(2, 1);
  lcd.print("Cooler System");
}

```

```

        delay(3000);
        lcd.clear();
    }

    void loop() {
        temperature = analogRead(0) * 0.004882814;
        temperature = (temperature - 0.5) * 100.0;
        lcd.setCursor(0, 0);
        lcd.print("Temp : ");
        lcd.print(temperature);
        lcd.print(" C");
        delay(1000);
        lcd.clear();

        if (temperature < 40) {
            digitalWrite(fanCoolerPin, LOW);
            digitalWrite(13, HIGH);
            digitalWrite(8, LOW);
            digitalWrite(7, LOW);
            lcd.setCursor(0, 1);
            lcd.print(" Cooler OFF");
            delay(1000);
        }

        if (temperature > 40 && temperature < 60) {
            digitalWrite(fanCoolerPin, HIGH);
            digitalWrite(13, HIGH);
            digitalWrite(8, LOW);
            digitalWrite(7, LOW);
            speed = 100;
            lcd.setCursor(0, 1);
            lcd.print(" Cooler Speed:100");
            analogWrite(motor, speed);
            delay(1000);
        }

        if (temperature > 60 && temperature < 80) {
            digitalWrite(7, LOW);
            digitalWrite(8, HIGH);
            digitalWrite(13, LOW);
            speed = 150;
            lcd.setCursor(0, 1);
            lcd.print(" Cooler Speed:150");
            analogWrite(motor, speed);
            delay(1000);
        }
    }

```

```

        if (temperature > 80 && temperature < 125) {
            digitalWrite(7, HIGH);
            digitalWrite(8, LOW);
            digitalWrite(13, LOW);
            speed = 250;
            lcd.setCursor(0, 1);
            lcd.print("Cooler Speed:250");
            analogWrite(motor, speed);
            delay(1000);
        }

        color();
    }

// color strip
void color() {
    setColor();
    for (int i = 0; i < 16; i++) {
        strip.setPixelColor(i, strip.Color(redColor, greenColor, blueColor));
        strip.show();
    }
}

// picks random values to set for RGB
void setColor() {
    redColor = random(0, 255);
    greenColor = random(0, 255);
    blueColor = random(0, 255);
}

```

4 Bibliografie

- <https://www.tinkercad.com/>
- <https://www.youtube.com/watch?v=LrOM2GABK1g>
- <https://www.diagrameditor.com/>
- <https://www.overleaf.com/>