

# SPRAWOZDANIE Z ZAJĘĆ PBL1

*Czujniki*

*Kinga Konieczna*

*Jan Czechowski*

*12.12.2024*

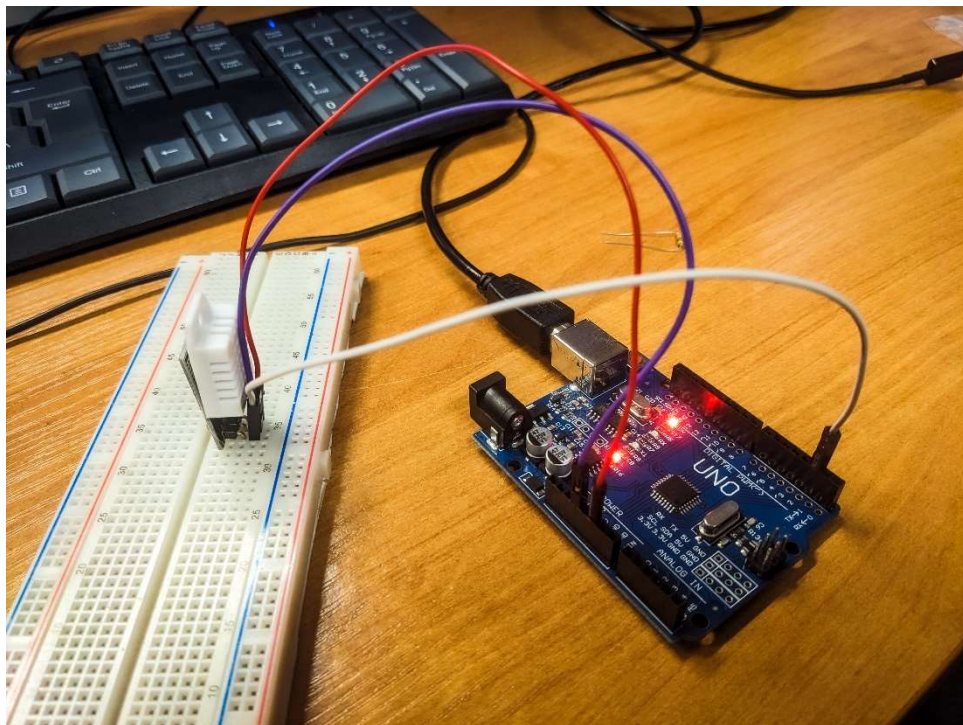
# 1. Czujnik temperatury i wilgotności DHT22

## 1.1. Wykorzystane narzędzia i komponenty

- Czujnik DHT22
- Arduino UNO
- Oprogramowanie: Arduino IDE

## 1.2. Podłączenie czujnika do Arduino

- „+” do zasilania 5V
- „-” do GND
- „OUT” do wejścia cyfrowego D2



Rys. 1. Podłączenie DHT22 do Arduino UNO [opracowanie własne]

## 1.3. Kod programu

```
1  #include "DHT.h"
2
3  #define DHTPIN 2
4  #define DHTTYPE DHT22
5
6  DHT dht(DHTPIN, DHTTYPE);
7
8  void setup() {
9      Serial.begin(9600);
10     dht.begin();
11 }
12
13 void loop() {
14     delay(2000);
15
16     float temperature = dht.readTemperature();
17     float humidity = dht.readHumidity();
18
19     if (isnan(temperature) || isnan(humidity)) {
20         Serial.println("Nie można odczytać danych z czujnika DHT22!");
21         return;
22     }
23
24     Serial.print("Wilgotność: ");
25     Serial.print(humidity);
26     Serial.print("%\t");
27
28     Serial.print("Temperatura: ");
29     Serial.print(temperature);
30     Serial.println("°C");
31 }
```

Rys. 2. Kod programu do podłączenia DHT22

## 1.4. Wyniki

Wyniki mogliśmy odczytać z Serial Monitoru:

Jak widać, odczyty z czujnika cyfrowego wyniosły:

- temperatura: 23.70°C
- wilgotność: około 31.40%

Output Serial Monitor ×		
Message (Enter to send message to 'Arduino Uno' on 'COM4')		
15:38:11.955 ->	Wilgotność: 31.30%	Temperatura: 23.70°C
15:38:13.948 ->	Wilgotność: 31.30%	Temperatura: 23.70°C
15:38:15.955 ->	Wilgotność: 31.30%	Temperatura: 23.70°C
15:38:18.005 ->	Wilgotność: 31.40%	Temperatura: 23.70°C
15:38:19.982 ->	Wilgotność: 31.40%	Temperatura: 23.70°C
15:38:21.975 ->	Wilgotność: 31.30%	Temperatura: 23.70°C
15:38:24.012 ->	Wilgotność: 31.20%	Temperatura: 23.70°C
15:38:25.988 ->	Wilgotność: 31.20%	Temperatura: 23.70°C
15:38:28.033 ->	Wilgotność: 31.20%	Temperatura: 23.70°C
15:38:30.026 ->	Wilgotność: 31.30%	Temperatura: 23.70°C
15:38:32.019 ->	Wilgotność: 31.60%	Temperatura: 23.70°C
15:38:34.067 ->	Wilgotność: 31.90%	Temperatura: 23.70°C
15:38:36.032 ->	Wilgotność: 32.20%	Temperatura: 23.70°C
15:38:38.075 ->	Wilgotność: 32.50%	Temperatura: 23.70°C
15:38:40.059 ->	Wilgotność: 32.70%	Temperatura: 23.70°C

Rys. 3. Wyniki pomiarów z czujnika DHT22

## 2. Czujnik temperatury - termistor NTC 110 10k $\Omega$ 5%

### 2.1. Wykorzystane narzędzia i komponenty

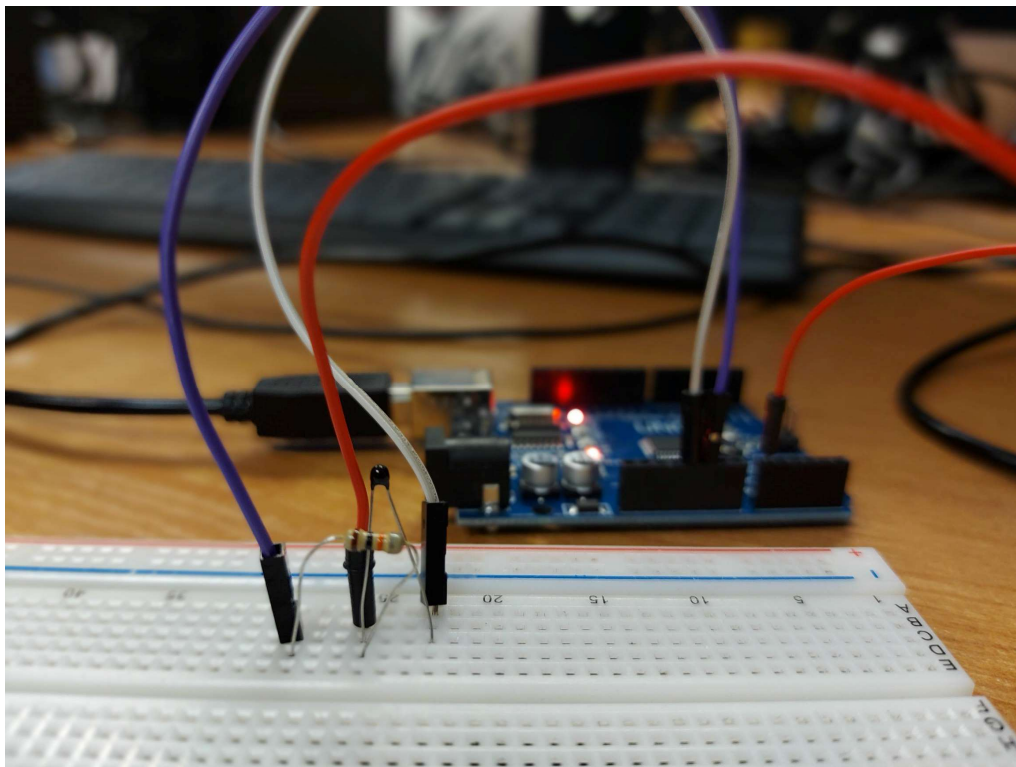
- termistor NTC 110 10k $\Omega$  5%
- Arduino UNO
- Oprogramowanie: Arduino IDE
- Rezystor 10k $\Omega$
- Multimetr Keysight U1241B

## 2.2. Podłączenie sprzętu:

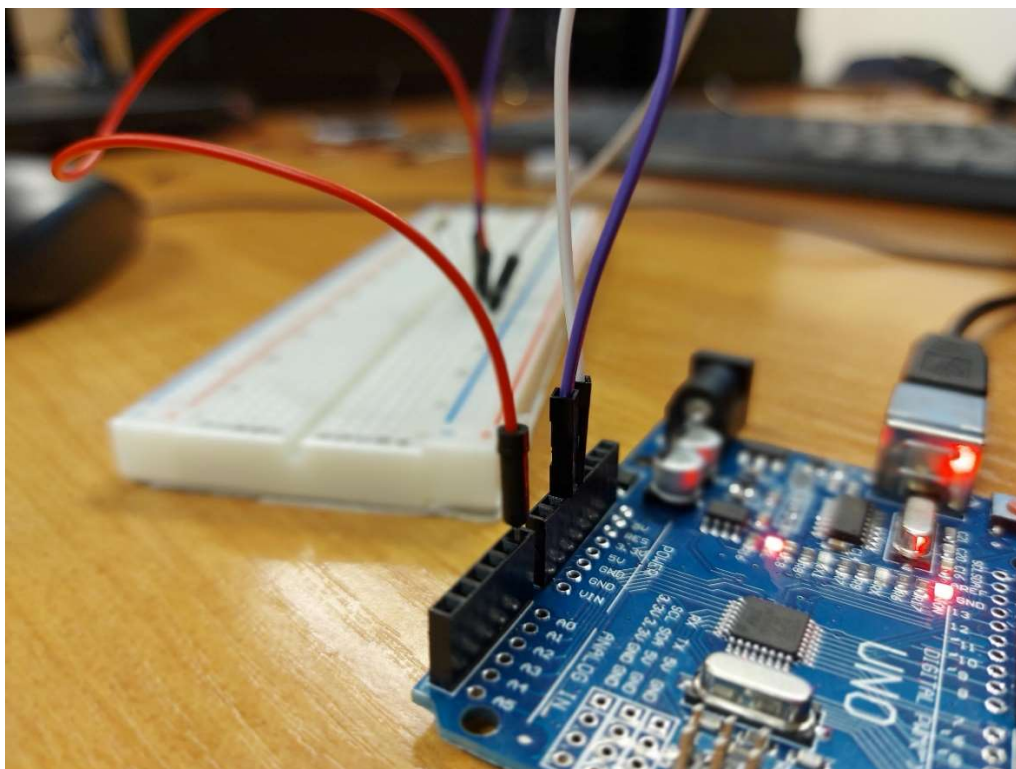
- Jeden koniec termistora podłączamy do GND.

Następnie tworzymy dzielnik napięciowy, dzięki czemu możemy zmierzyć rezystancję na termistorze:

- Drugi koniec termistora łączymy z jednym końcem rezystora 10 k $\Omega$ .
- Drugi koniec rezystora 10 k $\Omega$  do zasilania 5V.
- Punkt połączenia termistora i rezystora podłączamy do wejścia analogowego A0.

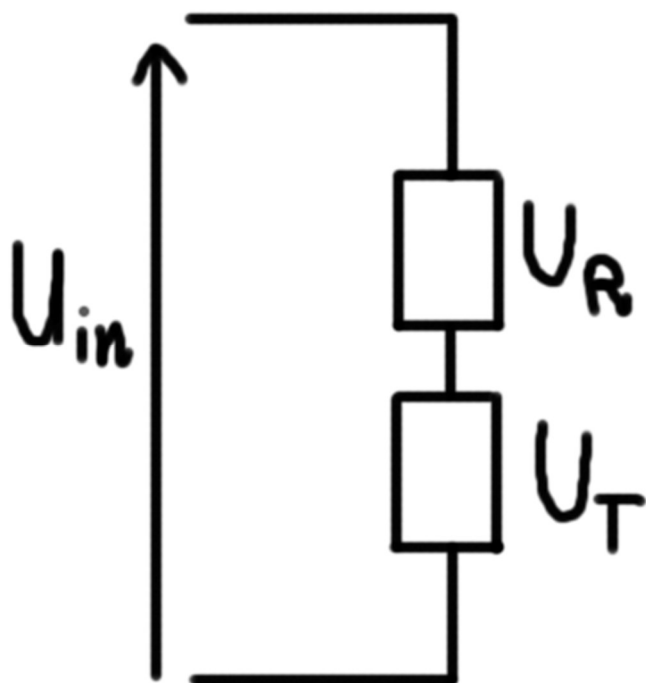


Rys. 4. Podłączenie termistora NTC 110 do Arduino UNO [opracowanie własne]



Rys. 5. Podłączenie termistora NTC 110 10k $\Omega$  5% do Arduino UNO [opracowanie własne]

### 2.3. Zmierzenie rezystancji na termistorze



Rys. 6. Schemat dzielnika napięciowego [opracowanie własne]



1. Mierzmy multimetrem napięcie na rezystorze.  
Wynik: 2,386 V
2. Z prawa Ohma liczymy wartość prądu przepływającego przez układ:

$$I = U_R / R_R$$

$$I = 2,386 \text{ V} / 10 \text{ k}\Omega = 0,0002386 \text{ A} = 0,2386 \text{ mA}$$

3. Mierzmy napięcie na termistorze.  
Wynik: 2,504 V
4. Z prawa Ohma liczymy rezystancję na termistorze:

$$R_T = U_T / I$$

$$R_T = 2,504 \text{ V} / 0,0002386 \text{ A} = 10495 \Omega$$

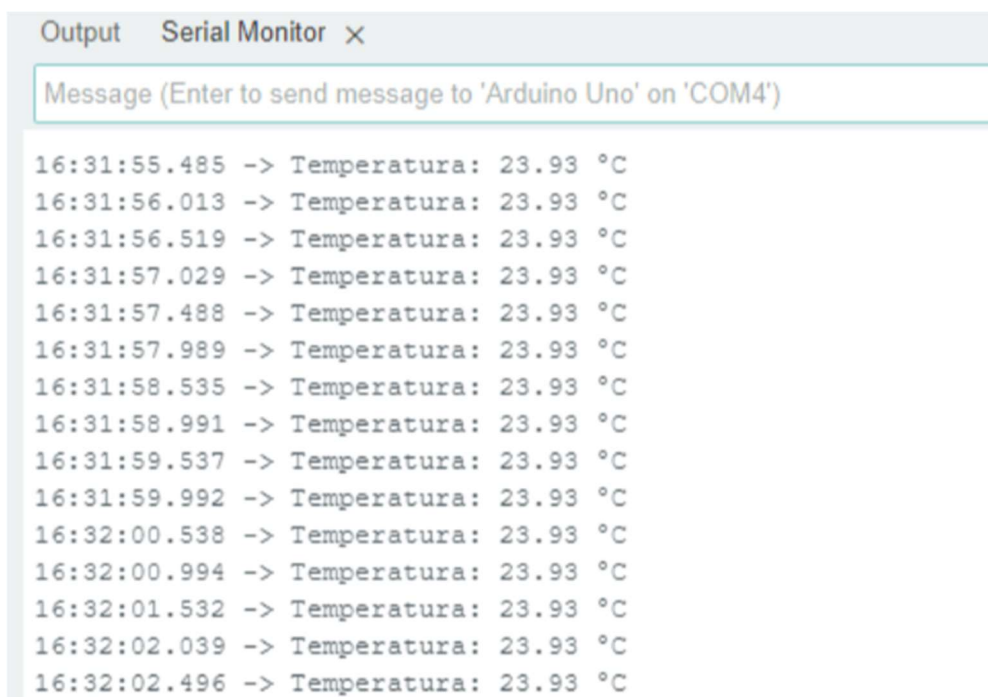
## 2.4. Kod programu:

```
1  #define THERMISTOR_PIN A0
2  #define SERIES_RESISTOR 10000
3  #define NOMINAL_RESISTANCE 10495
4  #define NOMINAL_TEMPERATURE 25
5  #define B_COEFFICIENT 3950
6
7  void setup() {
8      Serial.begin(9600);
9  }
10
11 void loop() {
12     int analogValue = analogRead(THERMISTOR_PIN);
13     float voltage = analogValue * 5.0 / 1023.0;
14
15     float resistance = SERIES_RESISTOR * (5.0 / voltage - 1.0);
16
17     float temperature;
18     temperature = resistance / NOMINAL_RESISTANCE;
19     temperature = log(temperature);
20     temperature /= B_COEFFICIENT;
21     temperature += 1.0 / (NOMINAL_TEMPERATURE + 273.15);
22     temperature = 1.0 / temperature;
23     temperature -= 273.15;
24
25     Serial.print("Temperatura: ");
26     Serial.print(temperature);
27     Serial.println(" °C");
28
29     delay(500);
30 }
```

Rys.6. Kod programu do podłączenia termistora NTC 110 10kΩ 5%

## 2.4. Wyniki

Wyniki mogliśmy odczytać z Serial Monitora. Jak widać, odczyt temperatury wyniósł: 23.93°C.

The image shows a screenshot of the 'Serial Monitor' window in an IDE. The window title is 'Output Serial Monitor' with a close button. Below the title bar is a text input field with the placeholder 'Message (Enter to send message to 'Arduino Uno' on 'COM4')'. The main area of the window displays a list of 15 lines of data. Each line consists of a timestamp followed by a right-pointing arrow and the text 'Temperatura: 23.93 °C'. The timestamps range from 16:31:55.485 to 16:32:02.496.

```
Output Serial Monitor ×
Message (Enter to send message to 'Arduino Uno' on 'COM4')
16:31:55.485 -> Temperatura: 23.93 °C
16:31:56.013 -> Temperatura: 23.93 °C
16:31:56.519 -> Temperatura: 23.93 °C
16:31:57.029 -> Temperatura: 23.93 °C
16:31:57.488 -> Temperatura: 23.93 °C
16:31:57.989 -> Temperatura: 23.93 °C
16:31:58.535 -> Temperatura: 23.93 °C
16:31:58.991 -> Temperatura: 23.93 °C
16:31:59.537 -> Temperatura: 23.93 °C
16:31:59.992 -> Temperatura: 23.93 °C
16:32:00.538 -> Temperatura: 23.93 °C
16:32:00.994 -> Temperatura: 23.93 °C
16:32:01.532 -> Temperatura: 23.93 °C
16:32:02.039 -> Temperatura: 23.93 °C
16:32:02.496 -> Temperatura: 23.93 °C
```

*Rys.7 Wyniki pomiarów z czujnika termistora NTC 110 10kΩ 5%*

## 3. Wnioski i obserwacje

Wyniki temperatury z czujników różnią się od siebie o 0.23 °C.