



2023

## 8. Station météorologique

Projet numéro: **2021-1-FR01-KA220-SCH-000031617**



**Co-funded by  
the European Union**

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

SCRAPY Partnership

31/05/2023

## Table des matières

Expérience 8: Station météorologique (degrés Celsius, degrés Fahrenheit).....	2
Description courte : .....	2
Description étendue : .....	2
Objectifs : .....	2
Matériel à utiliser : .....	3
Étapes à suivre : .....	3
Schéma de câblage .....	3
Code.....	4

## Expérience 8: Station météorologique (degrés Celsius, degrés Fahrenheit)

### Description courte :

Créez une station météorologique qui allume une LED rouge quand il fait chaud et une LED bleue quand il fait froid, avec Raspberry Pi Pico et le capteur de température et d'humidité numérique DHT11.

### Description étendue :

La station météorologique Raspberry Pi Pico est un projet à petite échelle conçu pour surveiller la température à l'aide du capteur de température et d'humidité numérique DHT11. Ce projet utilise la carte microcontrôleur Raspberry Pi Pico pour lire les données de température du capteur et contrôler l'illumination des LED en fonction des lectures de température.

En incorporant deux LED, une LED rouge pour indiquer les températures élevées et une LED blanche pour les températures basses, cette station météorologique offre une représentation visuelle de la température actuelle. Les LED seront contrôlées par le Raspberry Pi Pico, qui traitera les lectures de température du capteur DHT11 et déterminera quelle LED illuminer en conséquence.

Ce projet vise à montrer comment interagir avec un capteur DHT11 en utilisant le Raspberry Pi Pico et à créer ainsi un système de surveillance de la température simple.

### Objectifs :

Grâce à cette activité, l'utilisateur expérimentera la création d'un système de mesure de la température en utilisant la carte Raspberry Pi Pico et le capteur de température et d'humidité numérique DHT11. L'utilisateur acquerra des connaissances sur :

- La capacité du capteur de température et d'humidité numérique DHT11 à mesurer la température et l'humidité.
- Les bases de la programmation en Python et la manière d'écrire du code pour contrôler la carte Raspberry Pi Pico.
- Les principes de la conception de circuits et la manière de connecter des composants sur une carte de prototypage rapide pour créer un système de mesure de la température fonctionnel.

En menant à bien ce projet, l'utilisateur comprendra mieux l'électronique, l'ingénierie et la programmation. De plus, il disposera d'un appareil pratique et utile pour surveiller les conditions de température et être averti en cas de chaleur ou de froid.

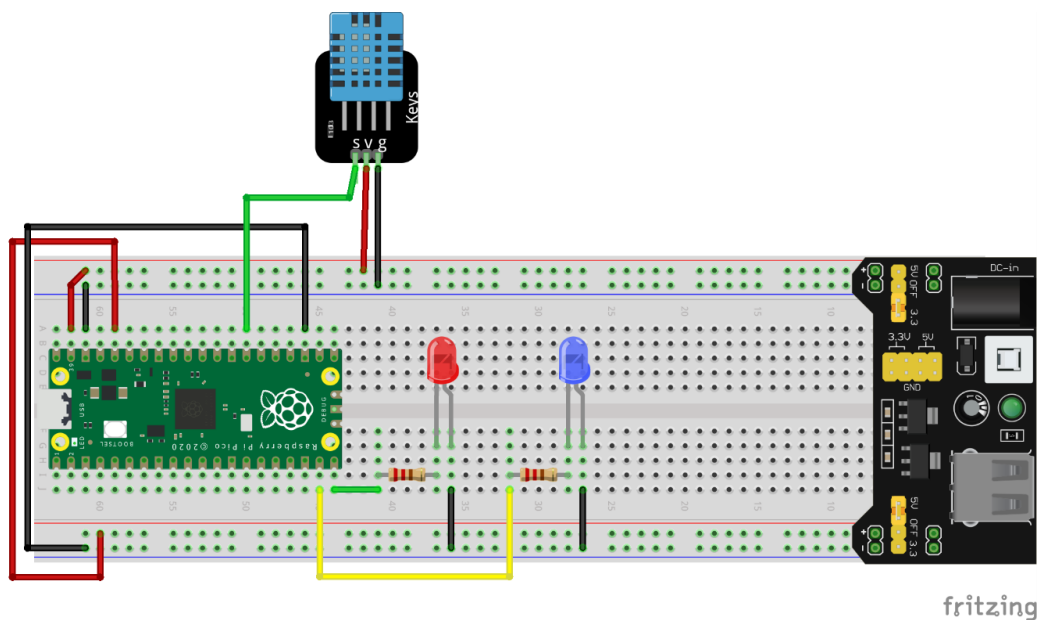
## Matériel à utiliser :

- 1 x Raspberry Pi Pico
- 1 x Kit de carte de prototypage Pico
- 1 x Carte de prototypage de taille normale
- 1 x Capteur de température et d'humidité numérique DHT11
- 2 x Résistance de 220 ohms
- 1 x LED bleue
- 1 x LED rouge
- Fils de raccordement

## Étapes à suivre :

1. Connectez la broche VCC du capteur DHT11 à la broche 3,3 V du Raspberry Pi Pico.
2. Connectez la broche GND du capteur DHT11 à la broche GND du Raspberry Pi Pico.
3. Connectez la broche DATA du capteur DHT11 à n'importe quelle broche GPIO sur le Raspberry Pi Pico. Pour cet exemple, utilisons la broche GPIO 21.
4. Connectez la cathode (patte plus courte) de la LED rouge à la broche GPIO 15 du Raspberry Pi Pico à l'aide d'une résistance de 220 ohms.
5. Connectez la cathode (patte plus courte) de la LED blanche à la broche GPIO 14 du Raspberry Pi Pico à l'aide d'une résistance de 220 ohms.
6. Connectez l'anode (pattes plus longues) des deux LED à la broche 3,3 V du Raspberry Pi Pico.

## Schéma de câblage



## Code

```
from machine import Pin
from time import sleep
import dht

sensor = dht.DHT11(Pin(21))

led_red = Pin(14, Pin.OUT)
led_white = Pin(15, Pin.OUT)

while True:
    sensor.measure()
    temp = sensor.temperature()
    fahr = temp * (9/5) + 32
    hum = sensor.humidity()
    print("Room temperature: ", temp, "Celsius")
    print("Room temperature: ", fahr, "Fahrenheit")
    print("Room humidity:", hum, "%")

    if temp > 40:
        led_red.on()
        led_white.off()
    else:
        led_red.off()
        led_white.on()

    sleep(2)
```