sleepCube - Programme/Aufbau

Ugur Turhal[1]

[1]ugur.turhal@me.com

Inhaltsverzeichnis

1 DHT22

Der Sensor ist ein DHT22, welcher wie folgt aus sieht. Mit diesem wird die Umgebungstemperatur gemessen. Der DHT22 hat 4 Pins siehe Bild 1. Dessen Erläuterung ist in der Tabelle1.

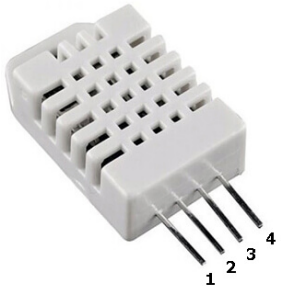

Abbildung 1. DHT22 - Pins

1.1 Pins und deren Bedeutung

Pin	Funktion
1	Betriebsspannung (3,3V - 5,5V)
2	Serial Data Line
3	Nicht belegt
4	Ground

Tabelle 1. Pins - Bedeutung

1.2 Verkabelung

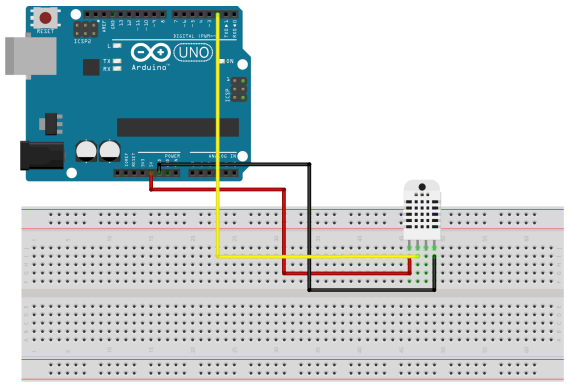

Abbildung 2. DHT mit Arduino UNO verbunden

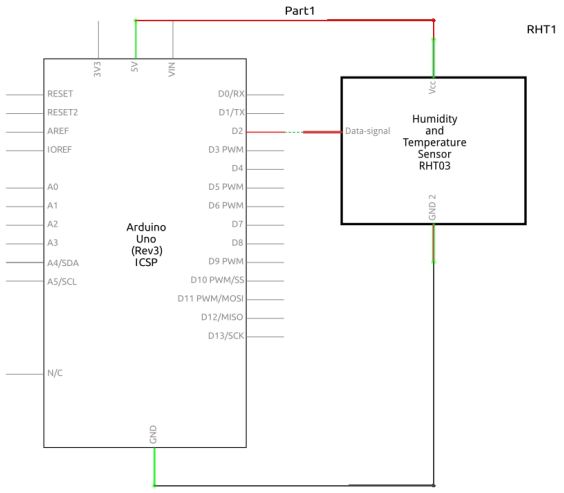

Abbildung 3. Schaltplan DHT22 und Arduino UNO

2 DHT22 - Arduino

Installiere die DHT sensor library in die Arduino IDE:

```
/
└── Tools
    └── Manage Libraries...
        └── Mangae Library
            └── Filter your Search
                ├── DHT22
                └── DHT22 sensor library
                    by Adafruit Version 1.4.3
```

DHT22 ist bereit zum Einsatz.

2.1 Arduino Code

```
#include "DHT.h"

#define DHTPIN 2 //Pin Nummer wo der Sensor angeschlossen ist
#define DHTTYPE DHT22 //Definition was fuer ein Sensor ausgelesen wird
    .

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  //Serial.println("DHT22 Testprogramm");
  dht.begin();

}

void loop() {
  delay(2000);

  float h = dht.readHumidity();
  float t = dht.readTemperature();

  if (isnan(h) || isnan(t)) {
    Serial.println("Fehler beim auslesen des Sensors!");
    return;
  }


  Serial.print(h);
  Serial.print(" ");
  Serial.println(t);
}
```

2.2 *Python and Arduino*

Zwei Python Programme wurden geschrieben um Daten aufzunehmen und zu visualisieren.

```
# @author Ugur Turhal
# This python file reads the serial port of the Arduino and gives
                                        a live plot
import serial
import time
import csv
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

matplotlib.use("tkAgg")
ser = serial.Serial('/dev/ttyACM0')
ser.flushInput()

plot_window = 360
y_var = np.array(np.zeros([plot_window]))
plt.ion()
fig, ax = plt.subplots()
ax.grid(linestyle="--")
ax.set_ylabel("Celsius")
ax.set_xlabel("x")
ax.set(xlabel='x', ylabel='Celsius',
       title='Increase & Decrease of \n Temperature')
line, = ax.plot(y_var)

while True:
    try:
        ser_bytes = ser.readline()
        try:
            decoded_bytes = ser_bytes[0:len(ser_bytes) - 2].decode
                                                    ("utf-8")
            print(decoded_bytes)
        except:
            continue
        with open("test_data.csv", "a") as f:
            writer = csv.writer(f, delimiter=",")
            writer.writerow([time.time(), decoded_bytes])
        y_var = np.append(y_var, decoded_bytes)
        y_var = y_var[1:plot_window + 1]
        line.set_ydata(y_var)
        ax.relim()
        ax.autoscale_view()
        fig.canvas.draw()
        fig.canvas.flush_events()
        fig.savefig('full_figure.png')
    except:
        print("Stopped")
        break
```

```python
# @author Ugur Turhal
# This python skript reads a CSV file, which
# is filtered and Plots in a log Plot the temperature

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

plt.style.use('seaborn')


def main():
    df = pd.read_csv("groupeddata.csv", usecols=['Measurement', '
                                        Temperature']).
                                        drop_duplicates(keep='first')
                                        .reset_index()
    measurement = df['Measurement']
    temperature = df['Temperature']
    df.to_csv("cleanData.csv");
    colors = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
                                        16, 17, 18, 19, 20, 21, 22,
                                        23, 24, 25, 26, 27, 28, 29,
                                        30, 31, 32, 33, 34, 35, 36,
                                        37, 38, 39, 40, 41, 42, 43,
                                        44, 45, 46, 47, 48, 49, 50,
                                        51, 52, 53, 54, 55, 56, 57,
                                        58, 59, 60, 61, 62, 63, 64,
                                        65, 66, 67, 68, 69, 70, 71,
                                        72, 73, 74, 75, 76, 77, 78,
                                        79, 80, 81, 82, 83, 84, 85,
                                        86, 87, 88, 89, 90]

    plt.scatter(x=measurement, y=temperature, s=25, c=colors,
                                        alpha=0.6, edgecolor='black',
                                         linewidth=1,
                    cmap='Blues_r')

    cbar = plt.colorbar()
    cbar.set_label('Mesurements with Temperature')
    plt.xscale('log')
    plt.yscale('log')
    plt.title('Scatter of Temperature')
    plt.xlabel('Measurement')
    plt.ylabel('Temperature')

    plt.savefig("data.png")
    plt.show()


if __name__ == "__main__":
    print(main())
```

2.3 Plots

Für die Erklärung siehe: `https://github.com/ugurtu/CatcProject/blob/main/Bericht/bericht.pdf`

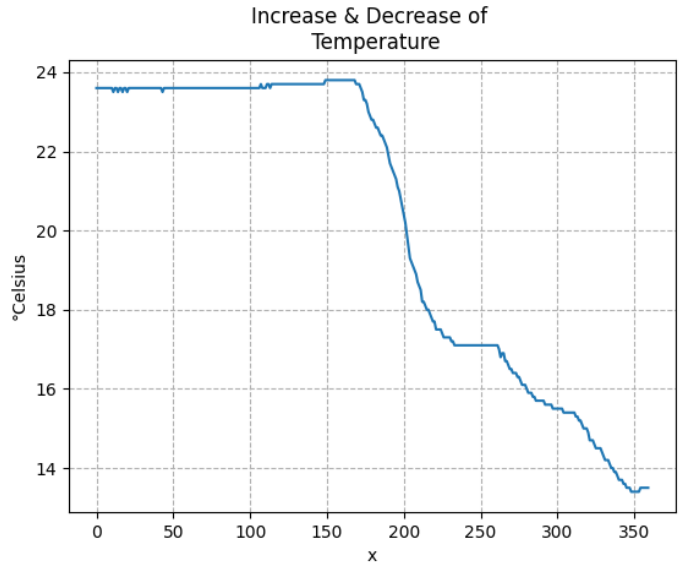

Abbildung 4. Temperaturausgaben über eine bestimme Zeit

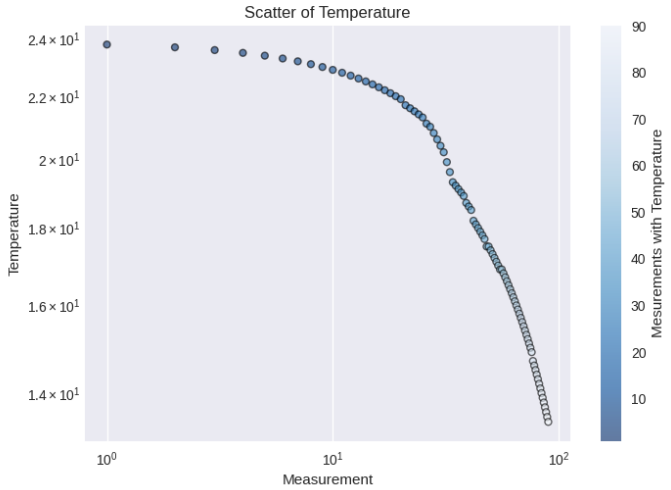

Abbildung 5. Einzelne Messpunkte des DHT22

3 Simples Modell

Arduino:

```
#include "DHT.h"

#define DHTPIN 2 //Pin Nummer wo der Sensor angeschlossen ist
#define DHTTYPE DHT22 //Definition was fuer ein Sensor ausgelesen wird.
#define pinRed 4 // LEDPin4
#define pinGreen 5 //LEDPin5

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  //Serial.println("DHT22 Testprogramm");
  pinMode(pinRed, OUTPUT);
  pinMode(pinGreen, OUTPUT);

  dht.begin();

}

void loop() {
  delay(2000);

  float h = dht.readHumidity();
  float t = dht.readTemperature();

  if (isnan(h) || isnan(t)) {
    Serial.println("Fehler beim auslesen des Sensors!");
    return;
  }

  if (t >= 15.00 && t <= 18.00) {
    digitalWrite(pinRed, LOW);
    digitalWrite(pinGreen, HIGH);
  }
  else  {
    digitalWrite(pinGreen,LOW);
    digitalWrite(pinRed, HIGH);
  }
   Serial.print(digitalRead(pinRed));
   Serial.print(" ");
   Serial.print(digitalRead(pinGreen));
   Serial.print(" ");
   Serial.println(t);
}
```

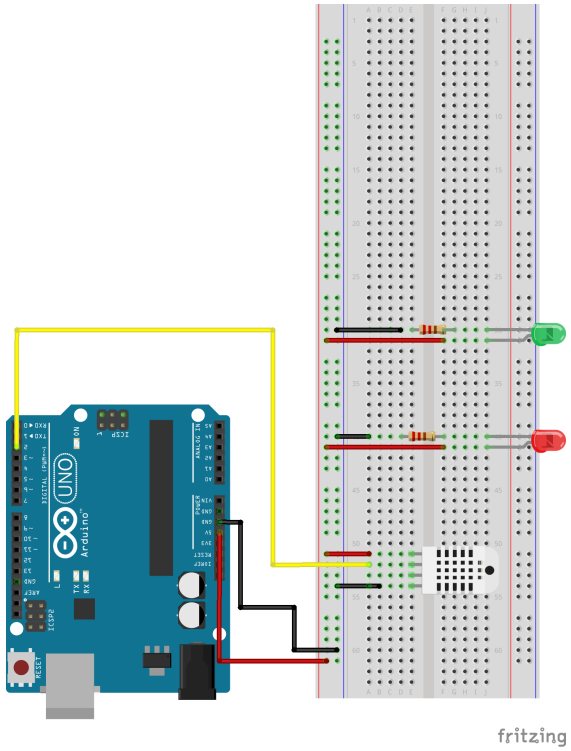

Abbildung 6. DHT und LEDs verbunden an Breadboard mit Arduino

```
# @author Ugur Turhal

# This python file reads the serial port of the Arduino and
# reads the state of the LEDs out and Shows the Temperature

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import serial
import time
import csv
from drawnow import *

plt.style.use('seaborn')
ser = serial.Serial('/dev/ttyACM0')
ser.flushInput()

ledRedPin = []
ledGreen = []
temperature = []


def makeFig():
    plt.close("all")
    plt.title('HIGH and LOW of \n selected pins')
    plt.ylim(0, 1.5, 0.4)
```

```
    plt.plot(ledRedPin, label='digitalRead_Red', color='red')
    plt.legend(bbox_to_anchor=(1.0, 0.5), loc='upper left')

    plt2 = plt.twinx()
    plt.ylim(0, 30, 6)
    plt2.plot(temperature, label='C')
    plt2.legend(bbox_to_anchor=(1.0, 1.01), loc='upper left')

    plt3 = plt.twinx();
    plt.ylim(0, 1.5)
    plt3.plot(ledGreen, label='digitalRead_Green', color='green')
    plt3.legend(bbox_to_anchor=(1.0, 0), loc='upper left')

    plt.tight_layout()
    plt.savefig("digitalRead.png")


def main():
    while True:
        ser_bytes = ser.readline()
        while ser.inWaiting() == 0:
            pass
        try:
            r = float(ser_bytes[0:1].decode("utf-8"));
            g = float(ser_bytes[1:len(ser_bytes) - 7].decode("utf-
                                                    8"));
            temp = float(ser_bytes[3:len(ser_bytes)].decode("utf-8
                                                    "));
            ledRedPin.append(r)
            temperature.append(temp)
            ledGreen.append(g)
            drawnow(makeFig)
        except:
            continue
            print("ERROR")
        with open("states.csv", "a") as f:
            writer = csv.writer(f, delimiter=",")
            writer.writerow([r, g, temp])


if __name__ == '__main__':
    print(main())
```

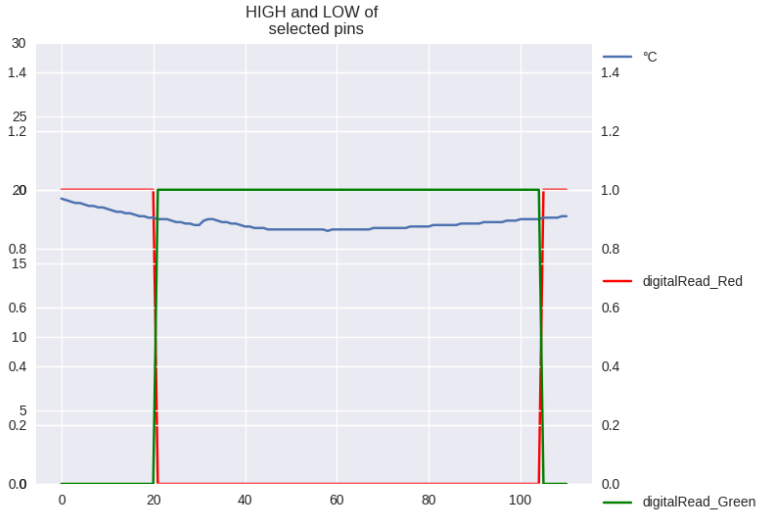

Abbildung 7. HIGH und LOW bei bestimmten Temperaturen beim Auslesen des Serial Ports

Für die Erklärung der Abbildung 7 siehe: `https://github.com/ugurtu/CatcProject/blob/main/Bericht/bericht.pdf`