

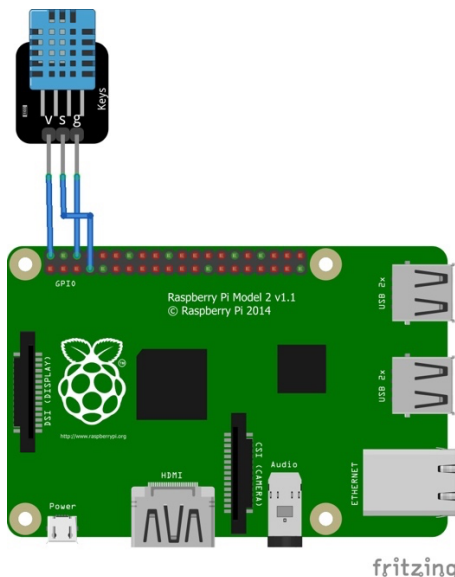
DHT-11 Sensor Projekt

Wir haben einen DHT11-Sensor mit einem Raspberry Pi verbunden und eine Website erstellt, um Temperatur- und Luftfeuchtigkeitsdaten anzuzeigen. Die Website ist sowohl lokal als auch über das Internet zugänglich.

Unser Ziel war es, grundlegende Kenntnisse in Hardware, Python-Programmierung und Webentwicklung zu erlangen. Diese Dokumentation gibt einen Überblick über den Prozess wie wir diese Aufgabe lösen konnten.

Hardware-Aufbau

Der Sensor hat einen Pin für Stromzufuhr, Signal und Erdung diese sind mit einer **Plus (+)**, **S** und **Minus (-)** markiert. Der Plus Pin wird mit einem Female zu Female am Raspberry Pi an dem 3,3V Pin angeschlossen das ist der Pin 1, Signal empfehle ich an GPIO 4 bzw. Pin 5 anzuschließen einfach aus dem Grund, weil sie in der Nähe sind. Erdung wird an Ground angeschlossen das ist möglich an Pin 6, 9, 14, 20, 25, 30, 34 und 39 ich würde aber Pin 6 empfehlen, um alles in einem Bereich anzuschließen.



Betriebssystem

Es gibt von Raspberry ein tolles Tool zum Installieren des Betriebssystems auf eine SD-Karte den [Raspberry Pi Imager](#). Als Betriebssystem habe ich hier Raspbian OS verwendet die minimale Version ohne Desktop Oberfläche für die maximale Performance und minimalen Stromverbrauch. Ich verwende die 32 Bit Version da mehr vom Raspberry Pi 2 nicht unterstützt wird. Für diese Version wird eine SD-Karte mit mindestens 4 Gib benötigt ich empfehle aber mehr falls man den Pi nicht nur für dieses Projekt nutzen möchte. Um nun auf dem Pi arbeiten zu können öffnen sie ein Terminal Fenster und geben ein `ssh <benutzername>@<ip-vom-Pi>`

Software

Damit eine Webseite gehostet werden kann brauchen wir Apache2. Dies installiert man mit:

```
sudo apt-get update  
sudo apt install apache2 -y
```

Wenn sie das gemacht haben und erfolgreich installiert wurde können sie nun die IP Adresse des Raspberry Pis im Browser auf dem Pi aufrufen um das zu überprüfen. Dort finden sie auch die Dokumentation von Apachen und wie sie es nutzen. Den Speicher Ort für die Webseiten finden sie unter `/var/www/html/`. Um dort später so wie der Beispiel Python Code ihre Webseiten speichern oder vom Code bearbeiten zu können müssen sie dem Nutzer Account, der den Code ausführen soll die Rechte dazu geben ich habe das mit: `sudo chmod +w /var/www/html/` gemacht damit wie sie später sehen können auch der Python Code schreiben und lesen darf.

Python ist standardmäßig auf dem Raspberry Pi installiert es sollte trotzdem geupdatet werden mit:

```
sudo apt-get update  
sudo apt-get upgrade
```

Wenn das durchgelaufen ist, muss nun das benötigte packet installiert mit dem sie den Sensor ansprechen können:

```
pip3 install adafruit-circuitpython-dht  
sudo apt-get install libgpiod2
```

Jetzt sollten sie bereit sein, um den Python und HTML code zu erstellen.

Python Code

Dieses Python-Skript den sie unter wohingehits.org/Webseite1.html finden liest Temperatur- und Luftfeuchtigkeitsdaten von einem DHT-Sensor und speichert die Daten in einer HTML-Datei.

Dieses Skript verwendet die folgenden Python-Bibliotheken:

`adafruit_dht`: Zum Auslesen der Daten des DHT-Sensors
`board`: Zum Zugriff auf die GPIO-Pins des Raspberry Pi
`time`: Zum Hinzufügen einer Verzögerung zwischen den Sensorablesungen

Hier sind die Dokumentationen zu den Paketen, die genutzt wurden, um den Code umzusetzen:

[Python Setup | DHT Humidity Sensing on Raspberry Pi or Beaglebone Black with GDocs Logging | Adafruit Learning System](#)
[Using the DHT11 Sensor on the Raspberry Pi - Pi My Life Up](#)
[How to write to an HTML file in Python ? - GeeksforGeeks](#)

Funktionen

`read_dht_sensor(pin)`

Diese Funktion liest die Temperatur- und Luftfeuchtigkeitsdaten vom DHT-Sensor, der an den angegebenen GPIO-Pin angeschlossen ist.

Die Funktion initialisiert den DHT-Sensor und liest dann kontinuierlich die Temperatur- und Luftfeuchtigkeitsdaten. Die Daten werden zusammen mit der aktuellen Uhrzeit in einer Liste gespeichert. Die Liste enthält immer die letzten zwei Einträge. Wenn mehr als zwei Einträge in der Liste sind, wird der älteste Eintrag entfernt.

Die Daten aus der Liste werden dann in eine HTML-Datei geschrieben, die in einem Webbrowser angezeigt werden kann. Die HTML-Datei enthält Balkendiagramme für Temperatur und Luftfeuchtigkeit.

Wenn beim Auslesen der Daten vom Sensor ein Fehler auftritt, gibt die Funktion eine Fehlermeldung aus und startet den Sensor neu.

Verwendung

Um das Skript zu verwenden, rufen Sie einfach die Python Datei auf mit:

```
python (dateiname-vom-script).py
```

Wenn das Skript erfolgreich gestartet wurde, sollte nun im /html Ordner eine output.html Datei erstellt worden sein.

Fazit

Das Projekt, eine Temperaturüberwachung mit einem Raspberry Pi und einem DHT11-Temperatursensor umzusetzen, ist ein spannendes Klassenprojekt. Die Nutzung von Python zum Auslesen der Sensordaten ermöglicht die Visualisierung auf einer selbst erstellten HTML-Seite. Die Bereitstellung über einen Apache-Server erweitert das Projekt global.

Ausblick

Dank diesem Projekt konnte ich ein neues Hobby finden und habe bereits weitere Projekte angefangen, eins der Projekte ist den gleichen Temperatur Sensor zu verwenden ihn auszulesen und auf einer Website anzuzeigen. Dies habe ich mit einem Raspberry Pi Pico W umgesetzt den bekommt man schon ab 8€, das Projekt kann man sich auf meiner GitHub Seite anschauen:

<https://github.com/na1Ra/MiniTemp>

Andere Projekte bzw. Erweiterungen, die man mit dem aktuellen Stand dieses Projekts ausführen könnten, wären z.B. Die gesammelten Daten loggen, um einen Trend erkennen zu können, Warnhinweise geben, wenn die Temperatur oder Luftfeuchtigkeit im Raum zu hoch wird. Das habe ich z.B. bereits auch schon umgesetzt mit einem Discord Bot der mir eine Nachricht schickt, wenn die Luftfeuchtigkeit über 58% liegt. Eine weitere Möglichkeit wäre den Raspberry Pi und Sensor in das Smart Home falls vorhanden mit aufzunehmen.

Aktuelle Temperatur und Luftfeuchtigkeit:

20°C

54%

[Python Code](#) [HTML Code](#)

Marc Stoppelkamp, Bastien Hanel

HTML Code

Temperatursensor

Dokumentation:

In dieser Dokumentation beschreiben wir die Installation des Apacheservers und des Temperatursensors.

```
import adafruit_dht
import board
import time
from datetime import datetime

def read_dht_sensor(pin):
    # DHT-Sensor initialisieren
    dht_sensor = adafruit_dht.DHT11(pin)

    # Liste für die Speicherung der letzten 2 Einträge
    data = []

    while True:
        try:
            # Temperatur und Luftfeuchtigkeit auslesen
            temperature_c = dht_sensor.temperature
            humidity = dht_sensor.humidity

            # Werte und Uhrzeit zur Liste hinzufügen
            data.append(f'<div style="background-color:gray; width:100%; height:20%;>
```

Marc Stoppelkamp, Bastien Hanel

Python Code

Temperatursensor

```
<html>
  <body style="background-color:#02285E">
  <head>
    <h6>
      <font face="Calibri" color=#ffffff>
      <pre><h3>Marc Stoppelkamp, Bastien Hanel

    </h6>

    <title>
      Tempsensor.doc
    </title>

  </head>

  <table border="2" cellpadding="10">
```