

Experimente mit Sensor BME280

- Ziel
- Was wird benötigt ?
- Aufbau
- Test I2C- Bus
- Schnittstelle zum Modul 'bme280.py', erste Ergebnisse
- Aufgabe
- Links

Ziel

Mit einem Mikrocontroller soll Temperatur, Luftdruck und relative Luftfeuchtigkeit gemessen und angezeigt werden.

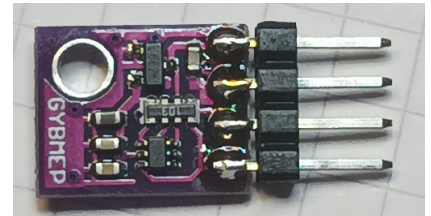
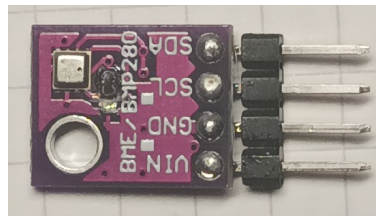
Im Endausbau können diese Werte auch auf einer Webseite abgerufen werden[4].

Was wird benötigt ?

- 1x Mikrocontroller – Modul ,D1 Mini‘ mit esp8266 Prozessor mit Micropython Interpreter[1] [2]



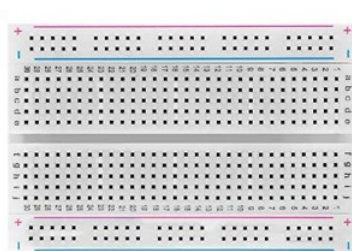
- 1x BME280 Modul mit flach auf der Lötseite aufgelöteten 4er Stiftleiste (Modul kann senkrecht auf das Steckbrett gesteckt werden)



- 2x Pullup-Widerstände 2.2 kOhm



- 1x Steckbrett, Verbindungsdrähte



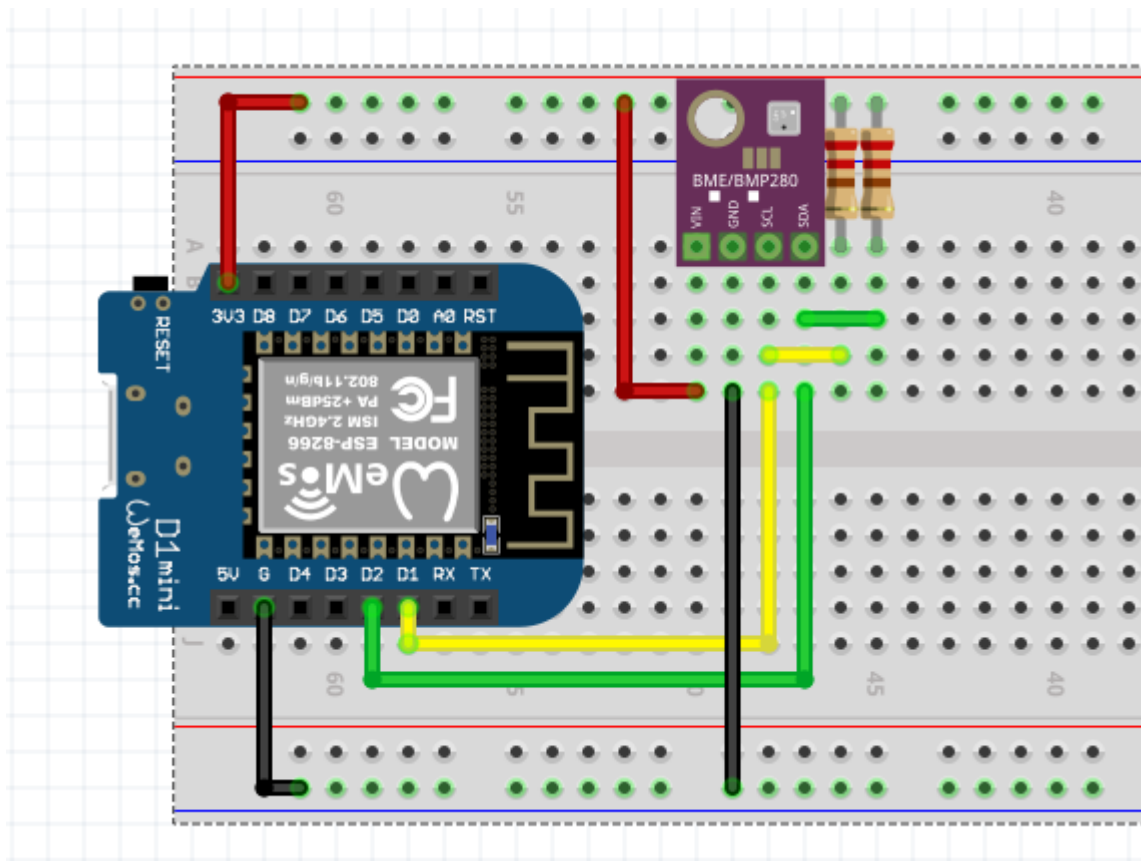


Bild erzeugt mit Fritzing[3]

Test I2C- Bus

Zur Prüfung, ob der I2C-Bus richtig angeschlossen ist, rufen wir in Thonny das Script 'i2c_scan.py' auf.

```
[ i2c_scan.py ]
1  from machine import I2C,Pin
2
3  # construct an I2C bus
4  i2c = I2C(scl=Pin(5), sda=Pin(4), freq=100000)
5  x= i2c.scan()
6
7  s1=[]
8  if x==[]:
9      s="*** keine i2c adr erkannt ***"
10 else:
11     for i in x:
12         s1.append(hex(i))
13     s="i2c-Adr: {} erkannt".format(s1)
14 print(s)
```

Wenn alles OK ist, dann sollte angezeigt werden:

```
>>> %Run -c $EDITOR_CONTENT
i2c-Adr: ['0x77'] erkannt
```

Wenn jedoch angezeigt wird:

```
>>> %Run -c $EDITOR_CONTENT
*** keine i2c adr erkannt ***
```

,dann bitte prüfen:

- SCL und SDA vertauscht ?
- Stromversorgung angeschlossen VCC= 3,3V ,GND = G
- Pullupwiderstände 2,2kOhm angeschlossen ?

Schnittstelle zum Modul 'bme280.py'

Nun laden wir ein Script '**BME280.py**' in den Mikrocontroller [5]. Dies kann dadurch erfolgen, dass im Dateifenster von Thonny vom PC zum MC kopiert wird; oder man ruft das Script '**BME280.py**' ins Dateifenster auf und lädt es dann mit 'speichern unter..' in den Controller.

Nun machen wir eine neue Datei auf, die wir z.B. '**read_bme.py**' nennen.

Zuerst muss das Modul '**BME**' und Teile aus dem Module 'machine' importiert werden:

```
import BME280
from machine import I2C,Pin
```

Jetzt 'verbinden' wir das BME Modul mit dem I2c- Bus:

```
i2c = I2C(scl=Pin(5), sda=Pin(4), freq=100000)
bme = BME280.BME280(address=0x77, i2c=i2c)
```

Die Pin Nummern sind die Pins mit der Bezeichnung GPIOx. Die I2c-Adresse haben wir beim '**i2c-scan.py**' herausgefunden (Anmerkung: wenn keine Adresse angegeben wird, wird 0x76 verwendet. Unsere Schnittstelle heisst als Objekt einfach 'bme'. Der Name ist willkürlich.

Das Ergebnis der Messung befindet sich in folgenden Eigenschaften von bme. (engl. Property) ; Properties sind Funktionen, die wie Variablen aufgerufen werden können, also ohne Klammern ():

```
bme.temperature,
bme.humidity,
bme.pressure
```

```
Ergebnis = bme.temperature, bme.humidity, bme.pressure
print( Ergebnis)
```

```
>>> Ergebnis = bme.temperature, bme.humidity, bme.pressure
print( Ergebnis)
('23.67C', '51.69%', '984.78hPa')
```

Hier haben wir es mit einem besonderen Datentyp zu tun:

‚Ergebnis‘ ist eine Anordnung von 3 Werten unter einem Namen, nämlich den Strings '23.89C' , '51.37%' und '984.29hPa' ; dieser Datentyp wird 'tuple' genannt. Tuples sind in runden Klammern angeordnet und durch Kommata getrennt.

Man kann das natürlich auch mit Einzelwerten schreiben, das Ergebnis ist ähnlich:

```
>>> print( bme.temperature, bme.humidity, bme.pressure )
23.70C 51.75% 984.76hPa
```

Aufgabe

Bei Überschreiten einer bestimmten Temperatur soll eine Leuchtdiode angehen.

Tipps:

Die eingebaute LED lässt sich über GPIO2 ansteuern:

```
from machine import Pin
# 2 eingebaute LED
led =Pin(2,Pin.OUT)
```

```
T=bme.temperature
```

Verwendung von

```
if T > xx:
```

```
elif T < yy:
```

```
...led(0)
...led(1)
```

Der BME Baustein sollte nur im Abstand von mindestens 2 sec. abgefragt werden.

```
from time import sleep
sleep(2.0)
```

Downloads der bisherigen Programme siehe [5].

Links:

- [1] „Micropython Download.pdf“ (**)
- [2] „Thonny Einstellungen für Micropython.pdf“ (**)
- [3] Fritzing: <https://fritzing.org/> (ältere Versionen sind frei verfügbar)
- [4] „Wifi accesspoint mit Esp8288“ (**)
- [5] Scripts siehe: xxxxxx (**)

(**) : muß noch erstellt werden (Stand: Sept. 2021)