

Elektro_{xy}

BME280

Sensormodul für Temperatur, Luftfeuchte und Luftdruck

Kurzanleitung

ElektroXY · Bach IT Service UG (haftungsbeschränkt)

Lindenstr. 7 · 53506 Kesseling · www.elektroxy.de

Worum es geht

Der BME280 von Bosch Sensortec vereint drei Messgrößen in einem Gehäuse von 2,5 × 2,5 mm: Temperatur, relative Luftfeuchte und barometrischen Luftdruck. Er wurde für mobile Geräte entwickelt und ist entsprechend sparsam — im Sekundentakt gemessen liegt die Stromaufnahme im einstelligen Mikroampere-Bereich. Das macht ihn zur naheliegenden Wahl für Wetterstationen, Raumklima-Überwachung, Lüftungssteuerungen und alles, was über längere Zeit aus einer Batterie laufen soll.

Aus dem Luftdruck lässt sich zusätzlich die Höhe über dem Meeresspiegel ableiten. Für Höhenänderungen ist der Sensor erstaunlich empfindlich — ein Stockwerk im Treppenhaus zeichnet sich in den Messwerten bereits deutlich ab.

Verwechslungsgefahr: Der äußerlich fast identische BMP280 misst nur Temperatur und Druck, ihm fehlt der Feuchtesensor. Wie Sie beide unterscheiden, steht im Abschnitt zur Fehlersuche.

Technische Daten

Messbereiche und Genauigkeit

Messgröße	Bereich	Genauigkeit	Auflösung
Temperatur	−40 ... +85 °C	±1,0 °C (±0,5 °C von 0 ... 65 °C)	0,01 °C
Relative Luftfeuchte	0 ... 100 % rF	±3 % rF	0,008 % rF
Luftdruck	300 ... 1100 hPa	±1 hPa absolut	0,18 Pa

Elektrische Eigenschaften

Eigenschaft	Wert
Sensorbaustein	Bosch BME280
Schnittstellen	I ² C (bis 3,4 MHz) und SPI (bis 10 MHz)
I ² C-Adresse	0x76 oder 0x77, per SDO-Pin wählbar
Chip-ID (Register 0xD0)	0x60
Versorgung des Sensors	1,71 ... 3,6 V
Versorgung des Moduls	je nach Ausführung 3,3 V oder 3,3 ... 5 V
Stromaufnahme	ca. 3,6 µA bei einer Messung pro Sekunde
Stromaufnahme im Ruhezustand	ca. 0,1 µA (Sleep Mode)

Anschließen

Modulvarianten

Im Handel sind zwei Bauformen verbreitet. Die vierpolige Ausführung ist fest auf I²C verdrahtet, die sechspolige lässt zusätzlich SPI zu:

Bauform	Pins	Betriebsarten
4-polig	VCC, GND, SCL, SDA	nur I ² C, Adresse fest verdrahtet
6-polig	VCC, GND, SCL, SDA, CSB, SDO	I ² C und SPI, Adresse über SDO wählbar

Achten Sie auf die Versorgungsspannung Ihres Moduls. Platinen mit aufgelötetem 3,3-V-Regler und Pegelwandler vertragen 5 V, nackte Breakouts ohne diese Bauteile nicht — dort zerstören 5 V den Sensor.

Verdrahtung über I²C

Board	VCC	GND	SDA	SCL
Arduino Uno / Nano	5 V bzw. 3,3 V	GND	A4	A5
Arduino Mega 2560	5 V bzw. 3,3 V	GND	Pin 20	Pin 21
D1 Mini / NodeMCU (ESP8266)	3V3	GND	D2 (GPIO4)	D1 (GPIO5)
ESP32 DevKit	3V3	GND	GPIO21	GPIO22
Raspberry Pi	Pin 1 (3,3 V)	Pin 6	Pin 3 (GPIO2)	Pin 5 (GPIO3)

Pull-up-Widerstände für SDA und SCL sind auf den handelsüblichen Modulen bereits bestückt. Erst wenn mehrere Module am selben Bus hängen, kann die Summe der Pull-ups zu niedrig werden — dann auf allen bis auf einem Modul auslöten.

I²C-Adresse festlegen

Der SDO-Pin bestimmt das unterste Adressbit:

SDO	Adresse
auf GND	0x76
auf VCC oder offen (bei den meisten Modulen)	0x77

So lassen sich zwei BME280 an einem Bus betreiben. Werden mehr benötigt, hilft ein I²C-Multiplexer oder der Betrieb über SPI.

Inbetriebnahme am Arduino

Installieren Sie in der Arduino-IDE über die Bibliotheksverwaltung die Pakete „Adafruit BME280 Library“ und „Adafruit Unified Sensor“. Anschließend läuft folgender Sketch:

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>

#define BME280_ADRESSE 0x76 // bei offenem SDO auf 0x77 aendern

Adafruit_BME280 bme;

void setup() {
  Serial.begin(9600);
  while (!Serial) { }
```

```

if (!bme.begin(BME280_ADRESSE)) {
  Serial.println(F("Kein BME280 gefunden."));
  Serial.println(F("Verdrahtung und I2C-Adresse pruefen."));
  while (true) { delay(1000); }
}

// Forced Mode: der Sensor misst nur auf Anforderung und bleibt dadurch kuehl.
// Das vermeidet den Messfehler durch Eigenerwaermung (siehe Abschnitt Genauigkeit).
bme.setSampling(Adafruit_BME280::MODE_FORCED,
                Adafruit_BME280::SAMPLING_X1, // Temperatur
                Adafruit_BME280::SAMPLING_X1, // Luftdruck
                Adafruit_BME280::SAMPLING_X1, // Luftfeuchte
                Adafruit_BME280::FILTER_OFF);
}

void loop() {
  bme.takeForcedMeasurement(); // eine Messung anstossen und abwarten

  Serial.print(F("Temperatur: "));
  Serial.print(bme.readTemperature(), 1);
  Serial.println(F(" C"));

  Serial.print(F("Luftfeuchte: "));
  Serial.print(bme.readHumidity(), 1);
  Serial.println(F(" %rF"));

  Serial.print(F("Luftdruck: "));
  Serial.print(bme.readPressure() / 100.0F, 1); // Pa in hPa
  Serial.println(F(" hPa"));

  Serial.println(F("-----"));
  delay(2000);
}

```

Der serielle Monitor muss auf 9600 Baud stehen. Nach wenigen Sekunden erscheinen die ersten Messwerte.

Inbetriebnahme am Raspberry Pi

I²C aktivieren und Modul suchen

Aktivieren Sie die I²C-Schnittstelle über „sudo raspi-config“ unter Interface Options und starten Sie neu. Danach prüfen Sie, ob der Sensor erkannt wird:

```

sudo apt install -y python3-pip i2c-tools
i2cdetect -y 1

```

In der Ausgabetabelle muss 76 oder 77 erscheinen. Taucht keine der beiden Adressen auf, stimmt etwas mit der Verdrahtung nicht — dazu mehr im nächsten Abschnitt.

Auslesen mit Python

```

pip3 install adafruit-circuitpython-bme280
import time

```

```
import board
from adafruit_bme280 import basic as adafruit_bme280

i2c = board.I2C()          # SDA = GPIO2 (Pin 3), SCL = GPIO3 (Pin 5)
sensor = adafruit_bme280.Adafruit_BME280_I2C(i2c, address=0x76)

# Referenzdruck auf Meereshöhe - nur fuer die Höhenberechnung noetig.
# Den aktuellen Wert liefert jeder Wetterdienst fuer Ihre Region.
sensor.sea_level_pressure = 1013.25

while True:
    print(f"Temperatur: {sensor.temperature:6.1f} C")
    print(f"Luftfeuchte: {sensor.relative_humidity:6.1f} %rF")
    print(f"Luftdruck: {sensor.pressure:6.1f} hPa")
    print(f"Höhe: {sensor.altitude:6.1f} m")
    print("-" * 28)
    time.sleep(2)
```

Höhe aus dem Luftdruck berechnen

Die barometrische Höhenformel verknüpft den gemessenen Druck p mit dem Referenzdruck p_0 auf Meereshöhe:

$$h = 44330 * (1 - (p / p_0) ^ (1 / 5.255))$$

h ergibt sich in Metern, p und p_0 müssen in derselben Einheit eingesetzt werden. Der Standardwert 1013,25 hPa für p_0 gilt nur für die Normatmosphäre — das tatsächliche Wettergeschehen verschiebt ihn um mehrere Dutzend hPa. Für belastbare Absoluthöhen setzen Sie deshalb den aktuellen regionalen Wert ein.

Umgekehrt gilt: Ist Ihre Höhe bekannt, können Sie den gemessenen Druck auf Meereshöhe umrechnen und erhalten den Wert, den Wetterberichte angeben.

Genauigkeit in der Praxis

Eigenerwärmung

Im Normal Mode misst der Sensor fortlaufend und erwärmt sich dabei selbst. Je nach Messrate und Oversampling liegt die angezeigte Temperatur dann ein bis zwei Grad zu hoch — und weil die relative Luftfeuchte von der Temperatur abhängt, wandert deren Wert gleich mit. Drei Gegenmaßnahmen:

- Forced Mode verwenden, wie im Arduino-Beispiel gezeigt. Der Sensor schläft zwischen den Messungen.
- Oversampling niedrig halten, solange Sie keine maximale Auflösung brauchen.
- Abstand zu wärmeabgebenden Bauteilen schaffen. Direkt neben einem Spannungsregler oder über einem ESP32 misst der Sensor vor allem dessen Abwärme.

Einlaufverhalten des Feuchtesensors

Nach dem Auspacken und nach längerer Lagerung in trockener oder sehr feuchter Umgebung braucht der Feuchtesensor einige Stunden bei normalem Raumklima, bis die Werte stabil sind. Ein frisch angeschlossenes Modul, das 10 % daneben liegt, ist deshalb noch kein Defekt.

Betrieb über SPI

Die sechspolige Bauform beherrscht auch SPI. Dabei wechseln die Pins ihre Bedeutung:

Pin am Modul	Funktion bei SPI
SCL	SCK — Takt
SDA	SDI / MOSI — Daten zum Sensor
SDO	SDO / MISO — Daten vom Sensor
CSB	CS — Chip Select, low-aktiv

SPI lohnt sich, wenn mehrere Sensoren am selben Controller hängen oder hohe Abtastraten gefragt sind. Für die üblichen Anwendungen mit einer Messung pro Sekunde ist I²C die einfachere Wahl — zwei Leitungen statt vier.

Fehlersuche

Beobachtung	Wahrscheinliche Ursache	Abhilfe
i2cdetect zeigt keine Adresse an	SDA und SCL vertauscht, Masse fehlt oder kalte Lötstelle	Verdrahtung prüfen, Lötstellen nachlöten
Sensor meldet sich unter 0x77 statt 0x76	SDO liegt auf VCC oder ist offen	Adresse im Code anpassen oder SDO auf GND legen
begin() schlägt fehl, Chip-ID ist 0x58	Es handelt sich um einen BMP280	Der BMP280 kann keine Luftfeuchte; BMP280-Bibliothek verwenden
Luftfeuchte steht konstant auf 0 %	Ebenfalls ein BMP280	Chip-ID in Register 0xD0 auslesen und vergleichen
Temperatur liegt ein bis zwei Grad zu hoch	Eigenerwärmung im Normal Mode	Forced Mode, geringeres Oversampling, thermisch entkoppeln
Werte springen oder das Modul bleibt stumm	5 V an einem Breakout ohne Spannungsregler	Mit 3,3 V betreiben; der Sensor verträgt maximal 3,6 V
Der Druckwert wirkt zu niedrig	Der Sensor misst absolut, nicht auf Meereshöhe reduziert	Auf Meereshöhe umrechnen, siehe Höhenformel

Chip-ID prüfen

Der sicherste Test, ob ein BME280 oder ein BMP280 vor Ihnen liegt, ist das Register 0xD0. Am Raspberry Pi genügt eine Zeile:

```
i2cget -y 1 0x76 0xD0
```

Rückgabewert	Baustein
0x60	BME280 — Temperatur, Luftfeuchte, Luftdruck
0x58	BMP280 — nur Temperatur und Luftdruck
0x61	BME680 — zusätzlich Gassensor, andere Bibliothek nötig

Angaben ohne Gewähr. Alle Werte beziehen sich auf den Sensorbaustein Bosch BME280; maßgeblich ist dessen jeweils aktuelles Datenblatt.