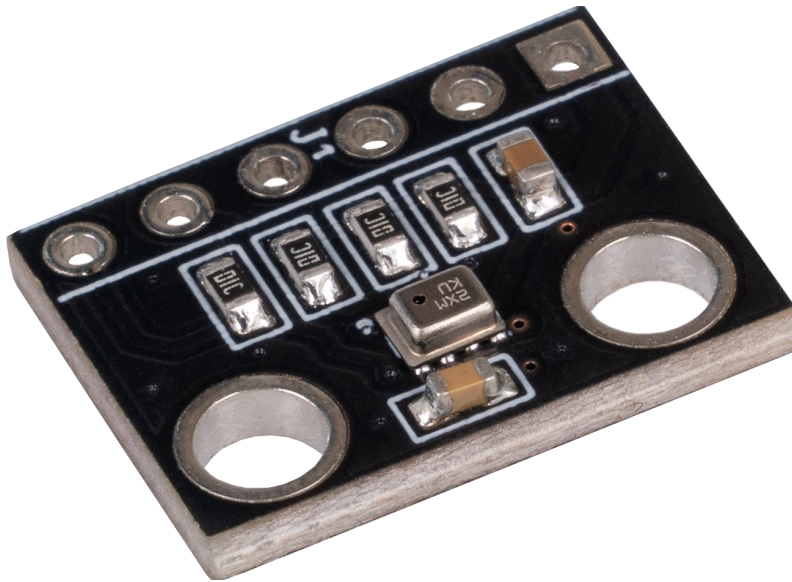


# TEMPERATUR- UND LUFTDRUCKSENSOR

SEN-KY052



## 1. ALLGEMEINE INFORMATIONEN

Sehr geehrter Kunde,  
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

## 2. VERWENDUNG MIT DEM RASPBERRY PI



Diese Anleitung wurde unter Raspberry Pi OS Bookworm für den Raspberry Pi 4 und 5 geschrieben. Es wurde keine Überprüfung mit neueren Betriebssystemen oder Hardware durchgeführt.

### 2.1 Installation

Zunächst müssen Sie **pip3** auf Ihrem Raspberry Pi installieren, dies funktioniert mit dem folgenden Befehl:

```
sudo apt-get install python3-pip -y
```

Als nächstes muss die virtuelle Umgebung eingerichtet werden. Geben Sie dazu die folgenden Befehle ein:

```
mkdir dein_projekt
```

```
cd dein_projekt
```

```
python -m venv --system-site-packages env
```

```
source env/bin/activate
```

Nun führen Sie diesen Befehl zum Installieren der Bibliothek aus:

```
pip3 install adafruit-circuitpython-bmp280
```

Sie benötigen auch diese weitere Bibliothek. Zum Installieren der Bibliothek führen Sie den folgenden Befehl aus:

```
pip3 install adafruit-circuitpython-lis3dh
```

#### 2.1.1 Installation SPI

Um den Sensor mit SPI zu verwenden, müssen Sie zunächst die SPI-Schnittstelle aktivieren.

Geben Sie dazu folgenden Befehl ein:

```
sudo raspi-config
```

Navigieren Sie nun zu **3 Interfacing Options** -> **I3 SPI** und aktivieren Sie die SPI-Schnittstelle.

#### 2.1.2 Installation I2C

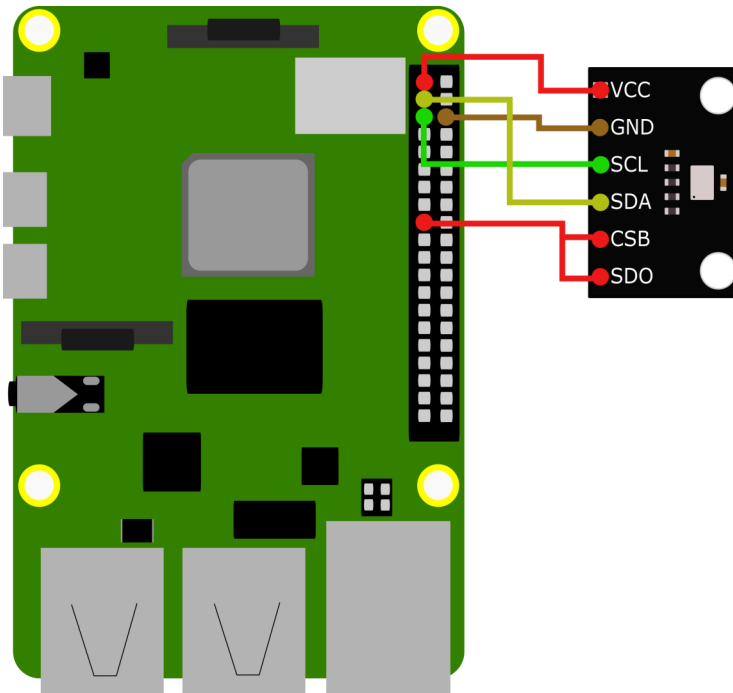
Um den Sensor mit I2C zu verwenden, müssen Sie zunächst die I2C-Schnittstelle aktivieren.

Öffnen Sie nun die Konfiguration des Raspberry Pis und aktivieren Sie I2C wie folgt:

```
sudo raspi-config
```

Navigieren Sie nun zu **3 Interface Options** -> **I4 I2C** und aktivieren Sie die I2C-Schnittstelle.

## 2.2.1 Anschluss I2C



### Raspberry Pi

### KY052

3,3 V  
(Pin 1, Pin 17)

VCC, SDO, CSB

GND (Pin 6)

GND

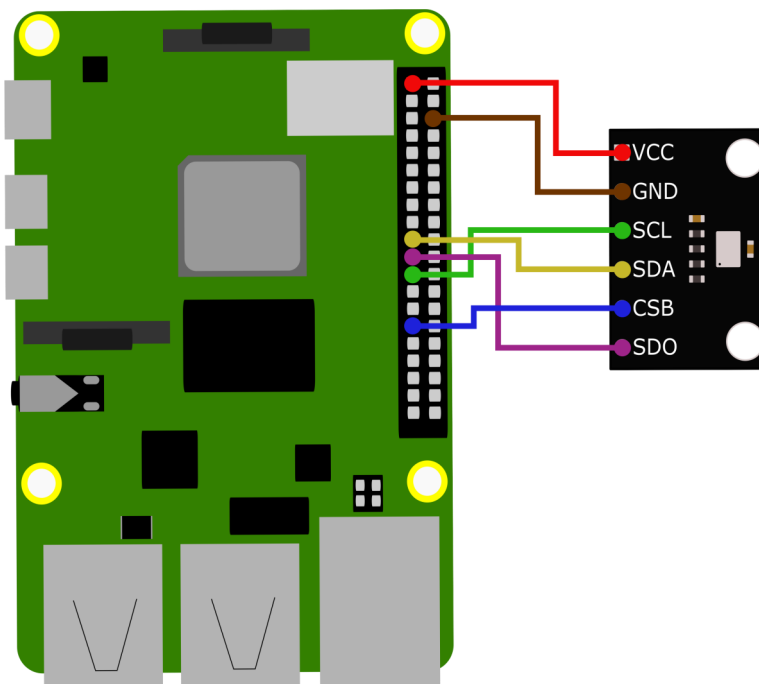
SCL (Pin 5)

SCL

SDA (Pin 3)

SDA

## 2.2.2 Anschluss SPI



### Raspberry Pi

### KY052

3,3 V  
(Pin 1, Pin 17)

VCC

GND (Pin 6)

GND

SCLK (Pin 23)

SCL

MOSI (Pin 19)

SDA

Pin 29

CSB

MISO (Pin 21)

SDO

## 2.3 Programmbeispiel

In dem nun folgendem Programmbeispiel, wird Ihnen die Temperatur, der Luftdruck und die errechnete Höhe angezeigt.

Damit die Höhe korrekt berechnet werden kann, müssen die folgende Zeile im Programmbeispiel anpassen:

```
sensor.sea_level_pressure = 1013.25
```

Ersetzen Sie die 1013.25 mit dem QNH-Druck in hPa (der Druck auf Meereshöhe) an Ihrem Ort. Wetterdienste können Ihnen dazu Auskunft geben.

### 2.3.1 Programmbeispiel I2C

Nun testen Sie Ihren BMP280 mit dem folgenden Beispielcode. Zunächst erstellen Sie eine Datei mit:

```
nano bmp280.py
```

Es öffnet sich nun ein leeres Fenster, in dem Sie den folgenden Beispielcode eingeben.

```
import board
import busio
import adafruit_bmp280

# Sensor als Objekt erstellen in Abhaengigkeit von I2C
i2c = busio.I2C(board.SCL, board.SDA)
sensor = adafruit_bmp280.Adafruit_BMP280_I2C(i2c)

# Dieser Wert muss an den aktuellen Luftdruck Ihres Standorts geaendert werden
# ansonsten kommt es zu Ungenauigkeiten
# Wetterdienste koennen Ihnen Auskunft geben
# 1013.25 hPa ist der mittlere Luftdruck auf Meereshoehe
sensor.sea_level_pressure = 1013.25

# Ausgabe der Messwerte
print("\nTemperatur: %0.1f *C" % sensor.temperature)
print("Luftdruck: %0.1f hPa" % sensor.pressure)
print("Höhe: %0.2f m" % sensor.altitude)
```

### 2.3.1 Programmbeispiel SPI

```
import board
import busio
import adafruit_bmp280
import digitalio #Benutzung mit SPI
spi = board.SPI()
bmp_cs = digitalio.DigitalInOut(board.D5)
sensor = adafruit_bmp280.Adafruit_BMP280_SPI(spi, bmp_cs)
# Dieser Wert muss an den aktuellen Luftdruck Ihres Standorts geaendert werden
# ansonsten kommt es zu Ungenauigkeiten
# Wetterdienste koennen Ihnen Auskunft geben
# 1013.25 hPa ist der mittlere Luftdruck auf Meereshoehe
sensor.sea_level_pressure = 1013.25

# Ausgabe der Messwerte
print("\nTemperatur: %0.1f *C" % sensor.temperature)
print("Luftdruck: %0.1f hPa" % sensor.pressure)
print("Hoehe: %0.2f m" % sensor.altitude)
```

Speichern Sie den Quellcode mit der Tastenkombination **STRG + O** und bestätigen Sie mit **Enter**. Verlassen Sie den Editor mit **STRG + X**.

Sie führen das Programmbeispiel mit folgendem Befehl aus:

```
python3 bmp280.py
```

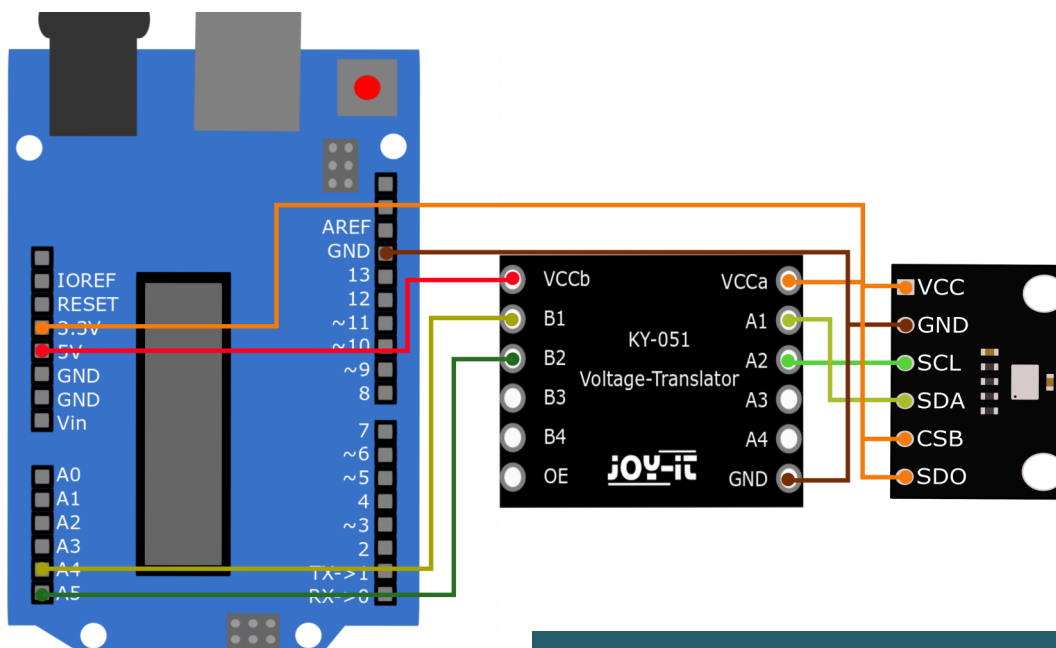
### 3. VERWENDUNG MIT DEM ARDUINO

#### 3.1 Installation

Um den Sensor mit dem Arduino verwenden zu können, müssen Sie zunächst die benötigte Bibliothek [hier](#) herunterladen. Entpacken Sie den Ordner und speichern Sie diesen in Ihrem Libraries Ordner. Dieser befindet sich meistens unter Dokumente → Arduino → Libraries.

#### 3.2.1 Anschluss I2C

Da der Sensor mit einem Logiklevel von 3,3 V und der Arduino UNO mit 5 V arbeitet, benötigen Sie für die Verwendung einen Logiklevelwandler. In diesem Beispiel wird der COM-KY051VT von Joy-IT verwendet.

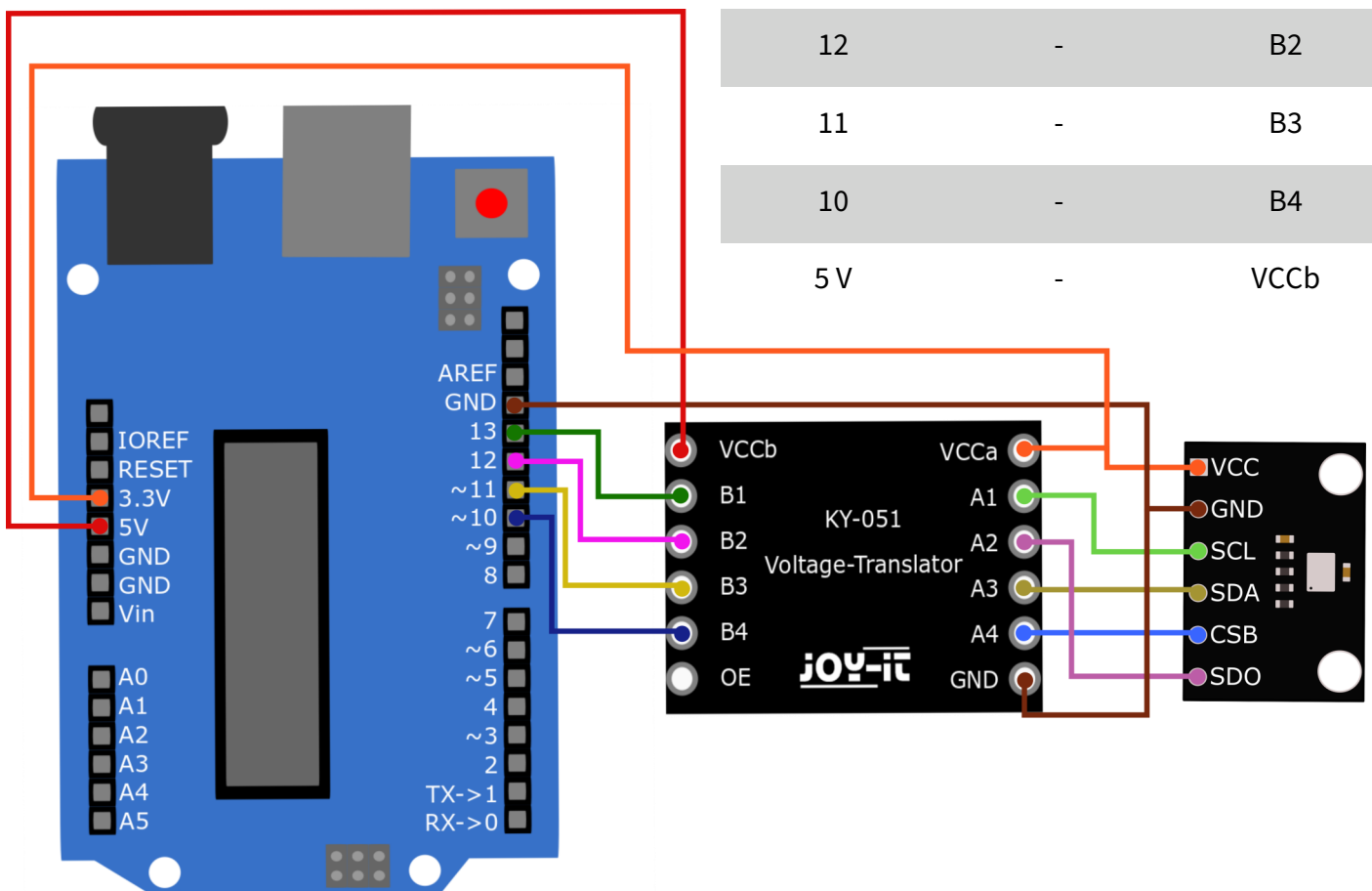


Arduino	KY052	Logiklevel-wandler
3,3 V	VCC, SDO, CSB	VCCa
GND	GND	GND
-	SDA	A1
-	SCL	A2
A4	-	B1
A5	-	B2
5 V	-	VCCb

### 3.2.2 Anschluss SPI

Da der Sensor mit einem Logiklevel von 3,3 V und der Arduino UNO mit 5 V arbeitet, benötigen Sie für die Verwendung einen Logiklevelwandler. In diesem Beispiel wird der COM-KY051VT von Joy-IT verwendet.

Arduino	KY052	Logiklevelwandler
3,3 V	VCC, SDO, CSB	VCCa
GND	GND	GND
-	SCL	A1
-	SDO	A2
-	SDA	A3
-	CSB	A4
13	-	B1
12	-	B2
11	-	B3
10	-	B4
5 V	-	VCCb



### 3.3 Programmbeispiel

In dem nun folgendem Programmbeispiel, wird Ihnen in der seriellen Konsole regelmäßig die Temperatur, der Luftdruck und die errechnete Höhe angezeigt.

```
#define localQNH 1013.25
```

Damit die Höhe korrekt berechnet werden kann, müssen die folgende Zeile im Programmbeispiel anpassen:  
Ersetzen Sie die 1013.25 mit dem QNH-Druck in hPa (der Druck auf Meereshöhe) an Ihrem Ort. Wetterdienste können Ihnen dazu Auskunft geben.

### 3.3.1 Programmbeispiel I2C

```
/*
*****
This is a library for the BMP280 humidity, temperature & pressure sensor

Designed specifically to work with the Adafruit BMEP280 Breakout
----> http://www.adafruit.com/products/2651

These sensors use I2C or SPI to communicate, 2 or 4 pins are required
to interface.

Adafruit invests time and resources providing this open source code,
please support Adafruit and open-source hardware by purchasing products
from Adafruit!

Written by Limor Fried & Kevin Townsend for Adafruit Industries.
BSD license, all text above must be included in any redistribution
*****
*/

#include <Wire.h>
#include <Adafruit_BMP280.h>

#define localQNH 1013.25

Adafruit_BMP280 bmp; // I2C

void setup() {
  Serial.begin(9600);
  Serial.println(F("BMP280 test"));

  if (!bmp.begin()) {
    Serial.println(F("Could not find a valid BMP280 sensor, check wiring!"));
    while (1);
  }
}

void loop() {
  Serial.print(F("Temperature = "));
  Serial.print(bmp.readTemperature());
  Serial.println(" *C");

  Serial.print(F("Pressure = "));
  Serial.print(bmp.readPressure());
  Serial.println(" Pa");

  Serial.print(F("Approx altitude = "));
  Serial.print(bmp.readAltitude(localQNH));
  Serial.println(" m");

  Serial.println();
  delay(2000);
}
```

### 3.3.2 Programmbeispiel SPI

```
/*
*****
This is a library for the BMP280 humidity, temperature & pressure sensor

Designed specifically to work with the Adafruit BMEP280 Breakout
----> http://www.adafruit.com/products/2651

These sensors use I2C or SPI to communicate, 2 or 4 pins are required
to interface.

Adafruit invests time and resources providing this open source code,
please support Adafruit and open-source hardware by purchasing products
from Adafruit!

Written by Limor Fried & Kevin Townsend for Adafruit Industries.
BSD license, all text above must be included in any redistribution
*****
#include <Wire.h>
#include <SPI.h>
#include <Adafruit_BMP280.h>

#define BMP_SCK 13
#define BMP_MISO 12
#define BMP_MOSI 11
#define BMP_CS 10

Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO, BMP_SCK);

void setup() {
  Serial.begin(9600);
  Serial.println(F("BMP280 test"));

  if (!bmp.begin()) {
    Serial.println(F("Could not find a valid BMP280 sensor, check wiring!"));
    while (1);
  }
}

void loop() {
  Serial.print(F("Temperature = "));
  Serial.print(bmp.readTemperature());
  Serial.println(" *C");

  Serial.print(F("Pressure = "));
  Serial.print(bmp.readPressure());
  Serial.println(" Pa");

  Serial.print(F("Approx altitude = "));
  Serial.print(bmp.readAltitude(1013.25)); // dies sollte an Ihren Lokalen
  Druck angepasst werden
  Serial.println(" m");

  Serial.println();
  delay(2000);
}
```



## 4. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektroggesetz (ElektroG)

### Symbol auf Elektro- und Elektronikgeräten:



Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

### Rückgabemöglichkeiten:

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in Haushaltsüblichen Mengen abgeben werden.

### Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

### Möglichkeit Rückgabe in Ihrer Nähe:

Wir senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an [Service@joy-it.net](mailto:Service@joy-it.net) oder per Telefon an uns.

### Informationen zur Verpackung:

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

## 5. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: [service@joy-it.net](mailto:service@joy-it.net)

Ticket-System: <https://support.joy-it.net>

Telefon: +49 (0)2845 9360 – 50 (Mo - Do: 09:00 - 17:00 Uhr,  
Fr: 09:00 - 14:30 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

[www.joy-it.net](http://www.joy-it.net)