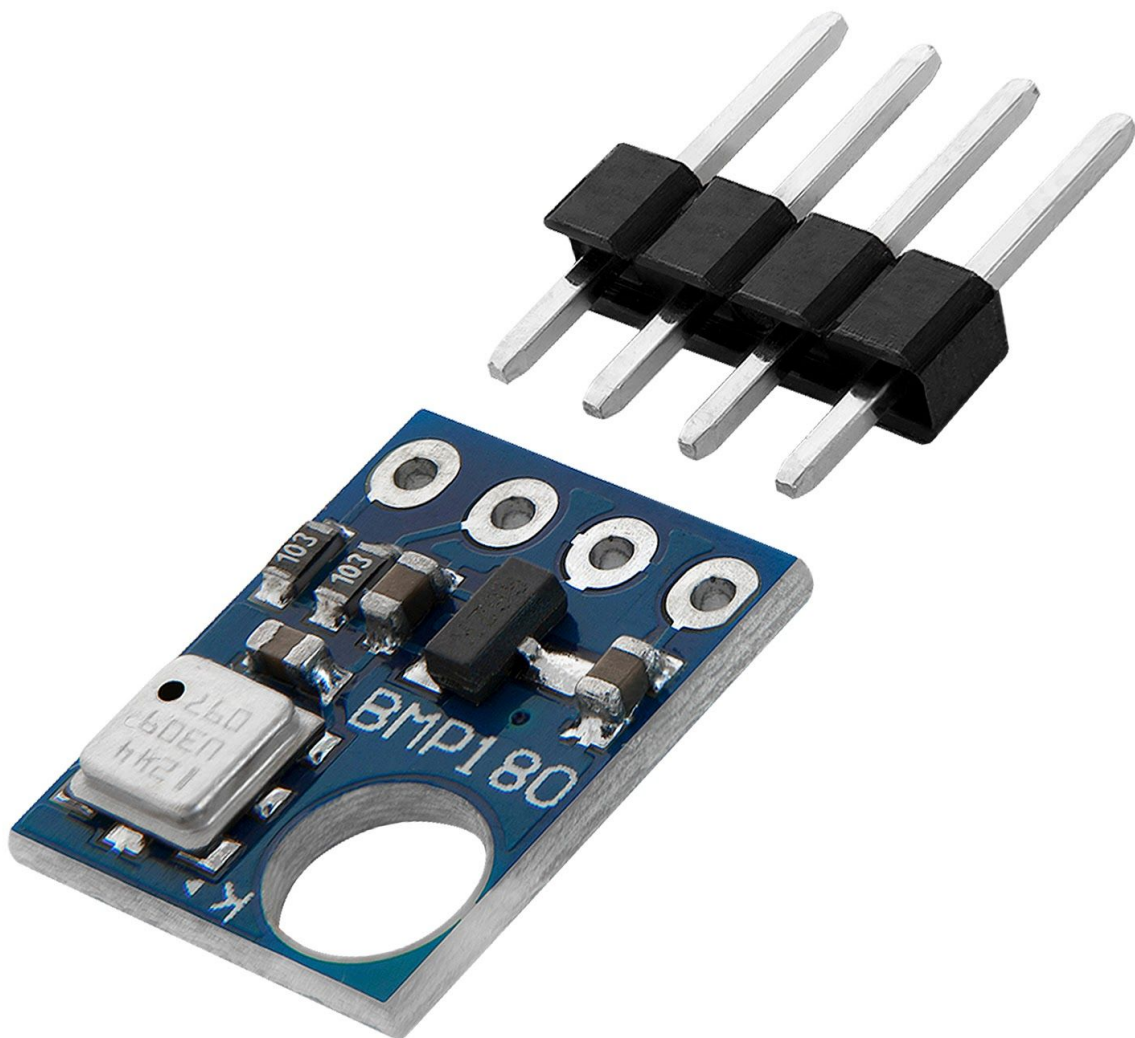


BMP180

eBook zum

GY-BMP180

Barometrischer Luftdruck- und Temperatursensor



Inhaltsverzeichnis

Einführung	3
Spezifikationen	4
Installation der Arduino IDE	5
Verbindung des Moduls mit dem Mikrocontroller	9
Beispiel Sketch	10
Einrichten des Raspberry Pi und Python	12
Aktivieren der I2C-Schnittstelle	13
Verbindung des Moduls mit dem Raspberry Pi	16

Einführung

Der BMP180 ist ein Luftdruck und Temperatur Sensor, der auf dem Bosch-Sensor BMP180 basiert. Dieser ist wesentlich präziser als sein Vorgängermodell BMP085: Er kann in einem Bereich von 300 - 1100hPa den Luftdruck messen und in einem Bereich von von -40°C bis +85°C die Temperatur messen.

Der Sensor besitzt eine Auflösung von 0,02 - 0,06hPA, was einer Höhendifferenz von ca. 17 cm entspricht und kann Temperaturen mit einer Genauigkeit von 0,1°C messen.

Spezifikationen

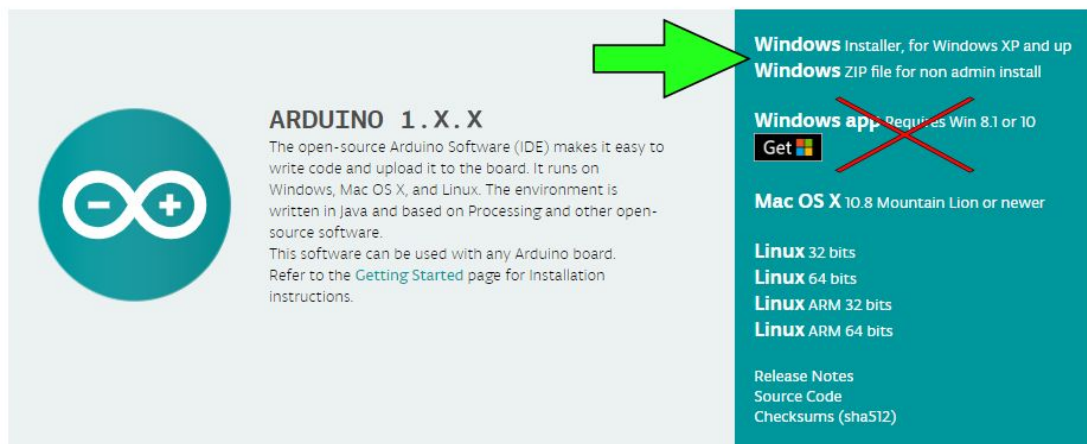
Betriebsspannung	3,3V
Schnittstelle	I2C
Druckbereich	300-1100 hPa
Abmessungen	13 x 10 x 2,5 mm

Installation der Arduino IDE

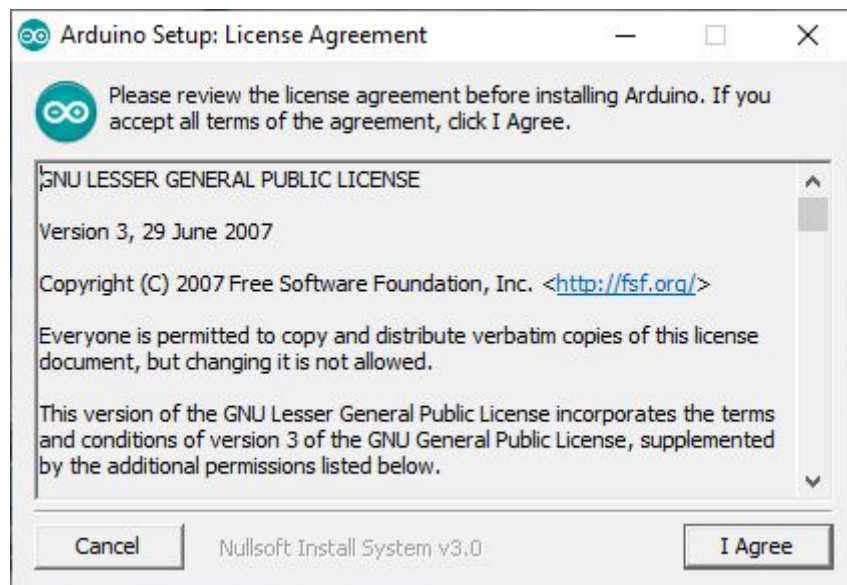
Die kostenlose Arduino IDE Entwicklungsumgebung können Sie unter folgendem Link herunterladen: <https://www.arduino.cc/en/Main/Software>

Windows Benutzer sollten unbedingt eine der ersten beiden Download-Optionen für die Arduino IDE verwenden. Die "Windows App" Version aus dem Windows Store führt insbesondere bei der Verwendung von Board-Definitionen von Drittanbietern zu Verbindungsproblemen.

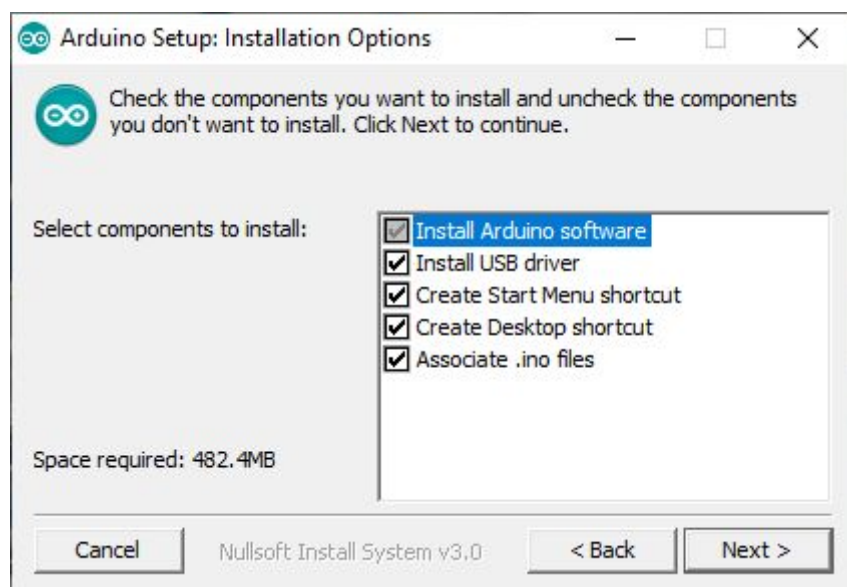
Download the Arduino IDE



Nach dem Starten der Arduino IDE Installationsdatei "arduino-1.X.X-windows.exe" müssen die Lizenzbedingungen der Software gelesen und akzeptiert werden:



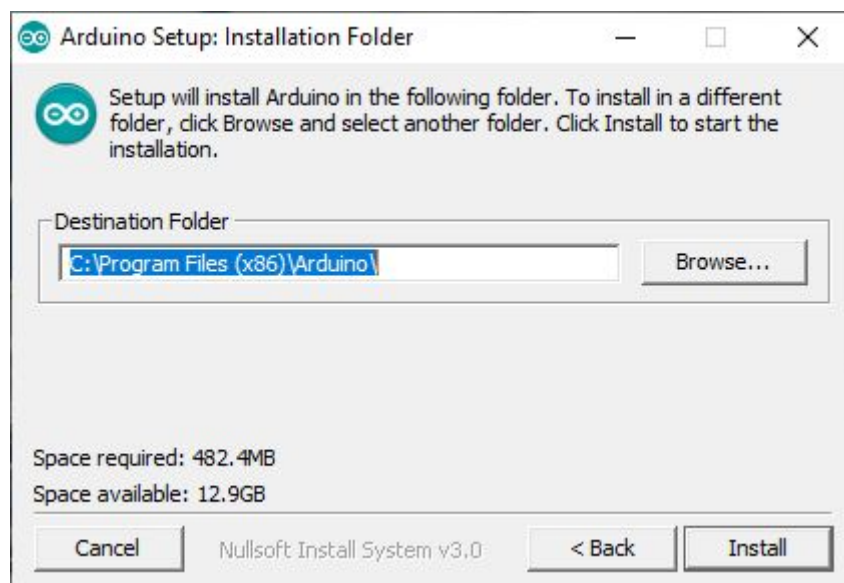
Im nächsten Schritt können unterschiedliche Optionen zur Installation ausgewählt werden.



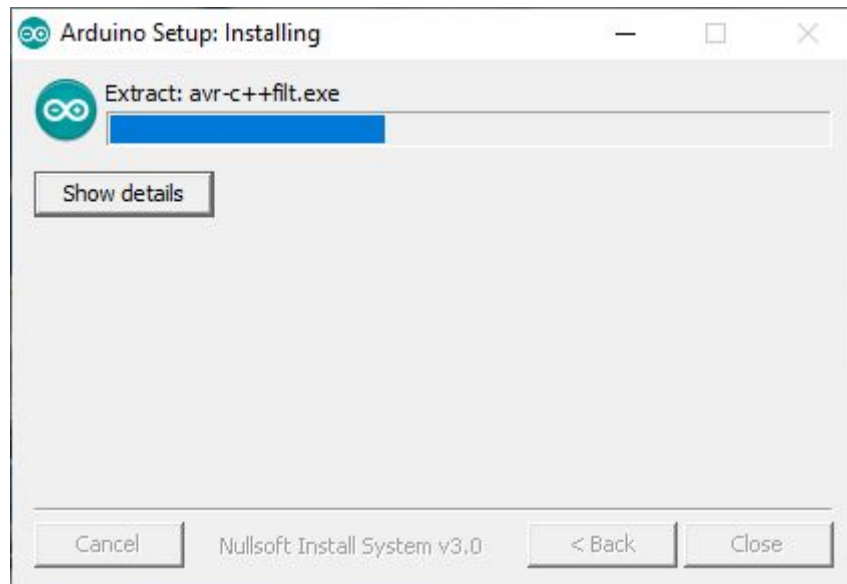
Es folgt eine kurze Übersicht der verschiedenen Optionen, mit einer kurzen Erklärung zu jeder Option:

Option	Erklärung
Install Arduino Software	Installiert die Arduino IDE - Diese Option kann nicht ausgewählt werden
Install USB Driver	Installiert USB Treiber für verschiedene andere Microcontroller. Für die Verwendung der Software mit dem D1 mini sind diese nicht erforderlich, wir empfehlen jedoch unbedingt die Installation, falls Sie auch andere Mikrocontroller verwenden
Create Start Menu shortcut	Erstellt eine Verknüpfung im Windows Startmenü (Optional)
Create Desktop shortcut	Erstellt eine Verknüpfung auf dem Arbeitsplatz (Optional)
Associate .ino files	Erstellt eine Dateinamenerweiterung für Dateien mit der Endung .ino und verknüpft diese mit der Arduino IDE

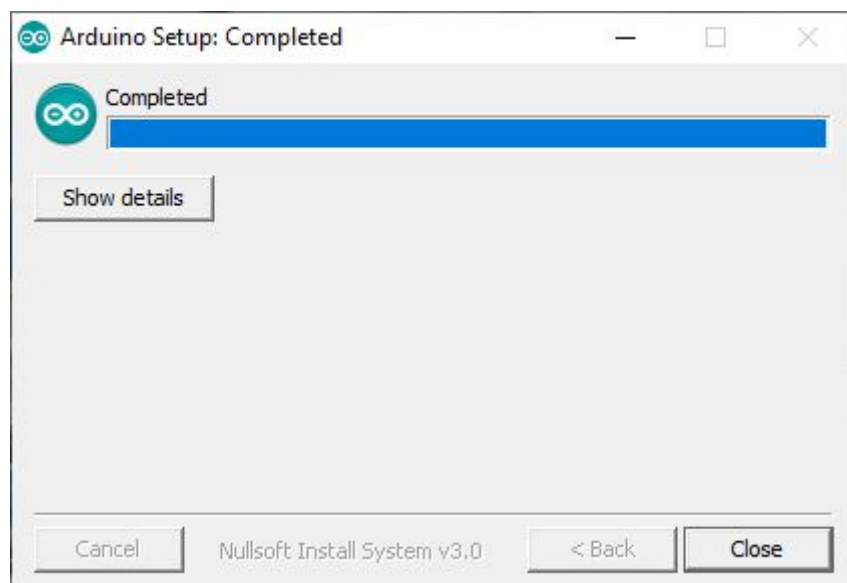
Zuletzt muss noch der Zielordner angegeben werden. Die Installation benötigt ca. 500MB freien Speicherplatz.



Klicken Sie auf "Install", um die Installation zu starten.

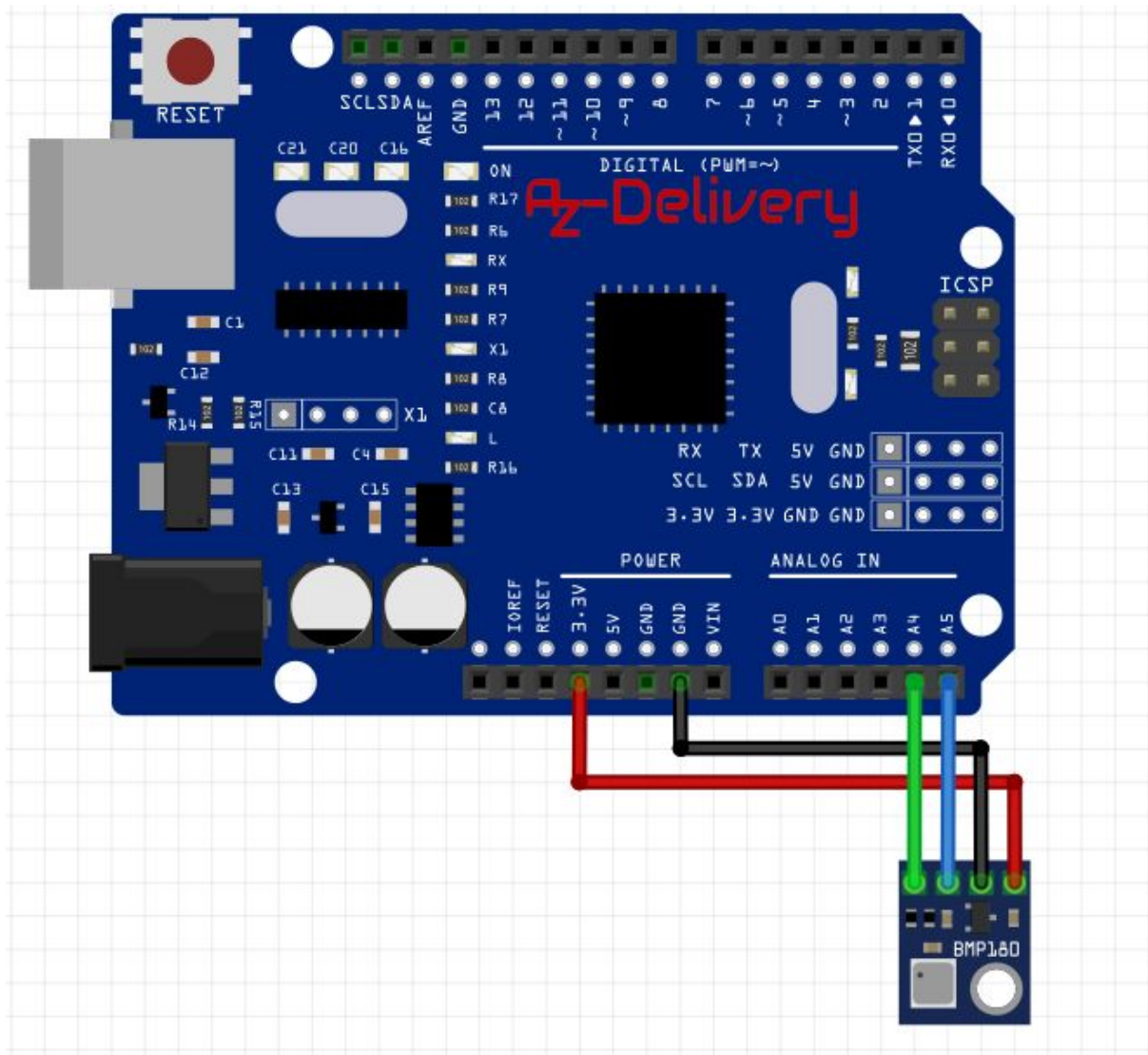


Nach erfolgreicher Installation kann das Installationsprogramm über die Schaltfläche "Close" beendet werden:



Verbindung des Moduls mit dem Mikrocontroller

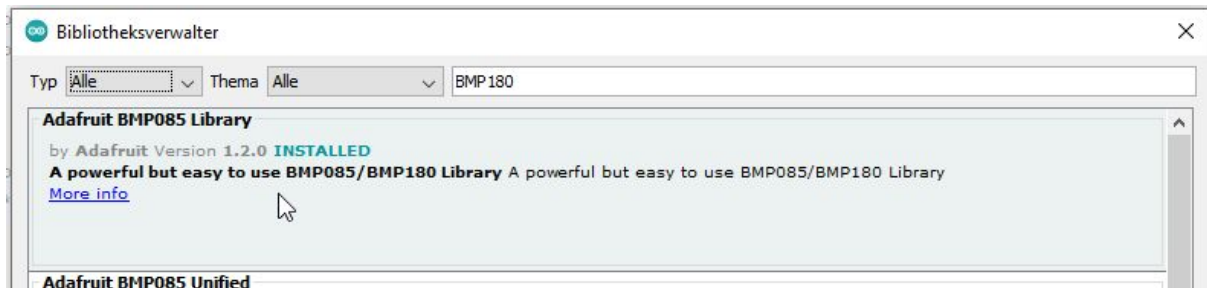
Der BMP180 wird wie folgt angeschlossen:



Mikrocontroller	BMP180
3,3V	Vin
GND	GND
A5	SCL
A4	SDA

Beispiel Sketch

Als schnelles Beispiel wird hier der Beispiel-Sketch von der Adafruit BMP085 Bibliothek verwendet, diese kann über Werkzeuge > Bibliotheksverwalter installiert werden.



Den Beispiel Sketch findet man unter Datei > Beispiele > Adafruit BMP085 Library > BMP085test. Dieser Sketch kann ohne weiteres auf den Mikrocontroller hochgeladen werden.

```
#include <Adafruit_BMP085.h>

Adafruit_BMP085 bmp;

void setup() {
  Serial.begin(9600);
  if (!bmp.begin()) {
    Serial.println("Could not find a valid BMP085 sensor,
check wiring!");
    while (1) {}
  }
}

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

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

```
// Calculate altitude assuming 'standard' barometric
// pressure of 1013.25 millibar = 101325 Pascal
Serial.print("Altitude = ");
Serial.print(bmp.readAltitude());
Serial.println(" meters");

Serial.print("Pressure at sealevel (calculated) = ");
Serial.print(bmp.readSealevelPressure());
Serial.println(" Pa");

// you can get a more precise measurement of altitude
// if you know the current sea level pressure which will
// vary with weather and such. If it is 1015 millibars
// that is equal to 101500 Pascals.
Serial.print("Real altitude = ");
Serial.print(bmp.readAltitude(101500));
Serial.println(" meters");

Serial.println();
delay(500);
}
```

Als nächstes öffnen wir den Seriellen Monitor und die Ausgabe sollte wie folgt aussehen:



Einrichten des Raspberry Pi und Python

Für den Raspberry Pi muss zunächst das Betriebssystem installiert werden, dann muss alles so eingerichtet werden, dass er im Headless-Modus verwendet werden kann. Der Headless-Modus ermöglicht eine Remote-Verbindung zum Raspberry Pi, ohne dass ein PC-Bildschirm Monitor, Maus oder Tastatur benötigt wird. Die einzigen Dinge, die in diesem Modus verwendet werden, sind der Raspberry Pi selbst, die Stromversorgung und die Internetverbindung. All dies wird im kostenlosen eBook genau erklärt:

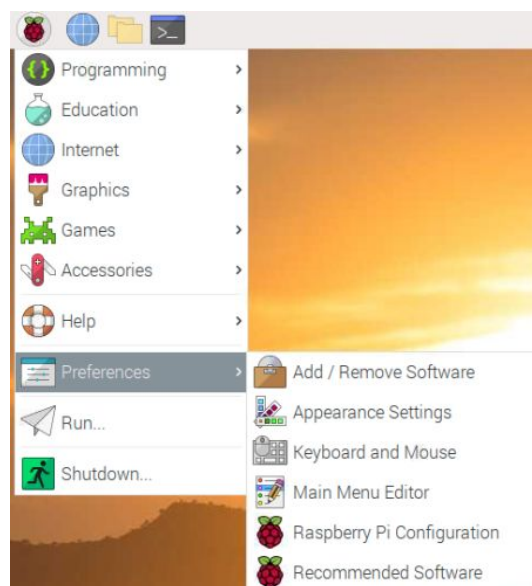
[Raspberry Pi Quick Startup Guide](#)

Auf dem Raspberry Pi OS ist Python vorinstalliert.

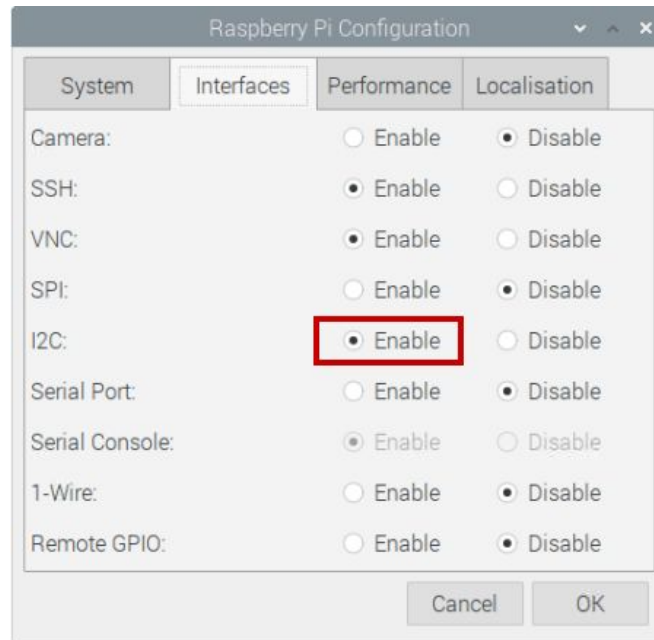
Aktivieren der I2C-Schnittstelle

Um den Sensor mit dem Raspberry Pi verwenden zu können, muss die I2C-Schnittstelle am Raspberry Pi aktiviert werden. Gehen Sie dazu auf:

Application Menu > Preferences > Raspberry Pi Configuration



Wenn sich ein neues Fenster öffnet, suchen Sie den *Interfaces* Reiter. Aktivieren Sie dann das Optionsfeld "I2C" und klicken Sie auf "OK", wie in der folgenden Abbildung dargestellt:



Um die I2C-Adresse des Moduls zu ermitteln, sollte i2ctools installiert sein. Falls keine vorhanden sind, ist folgender Befehl im Terminalfenster auszuführen: **sudo apt-get install i2ctools -y**

Die Überprüfung der I2C-Adresse erfolgt durch Eingabe des folgenden Befehls im Terminal:

i2cdetect -y 1

Die Terminalausgabe sollte wie auf dem folgenden Bild aussehen:

```
pi@raspberrypi:~/Desktop $ i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20: 20  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
pi@raspberrypi:~/Desktop $
```

Die I2C-Adresse des Moduls lautet 0x20

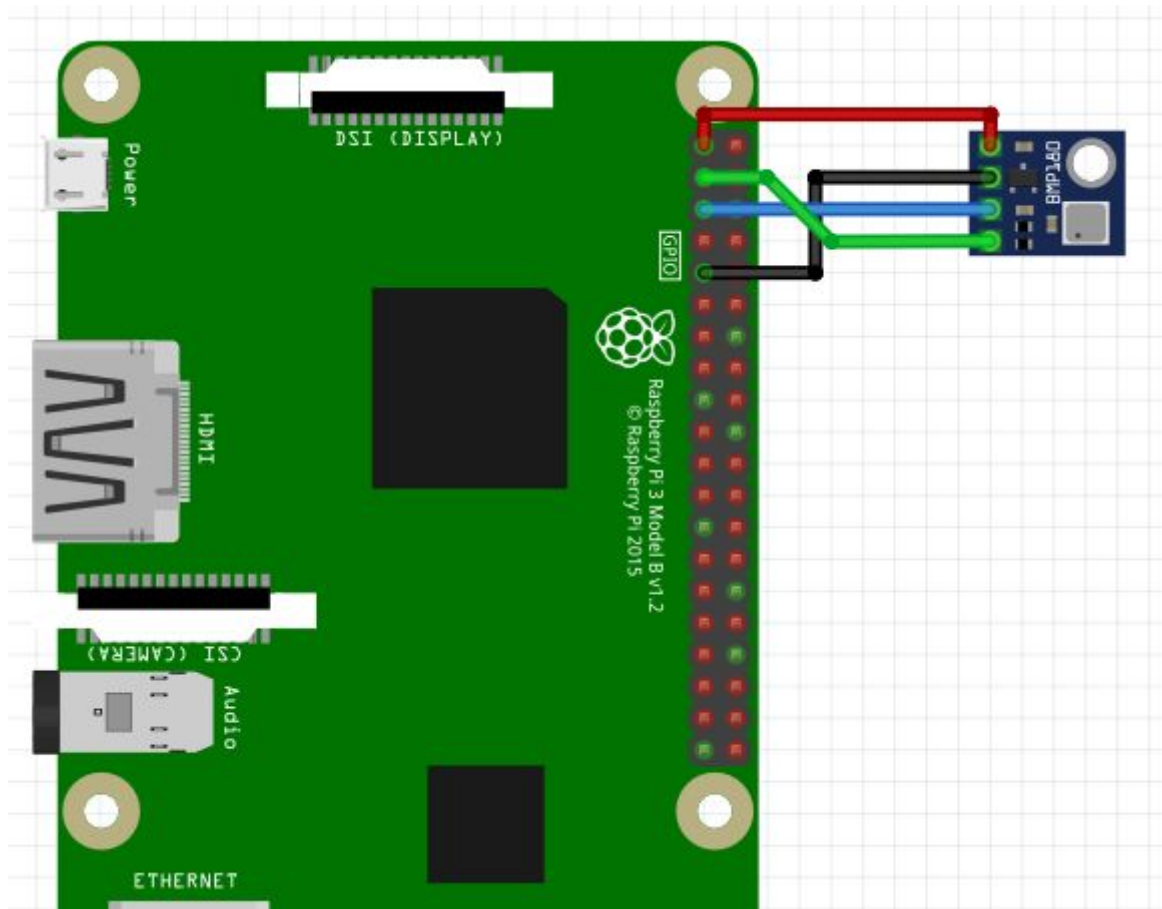
Mit dem Befehl:

/usr/sbin/i2cset -y 1 0x20 0x14 0x01

kann die LED eingeschaltet werden und mit **/usr/sbin/i2cset -y 1 0x20 0x14 0x00** wieder ausgeschaltet werden

Verbindung des Moduls mit dem Raspberry Pi

Die folgende Abbildung zeigt wie der BMP180 mit dem Raspberry Pi verbunden wird.



Raspberry Pi	BMP180
3V3	Vin
GND	GND
GPIO 8	SDA
GPIO 9	SCL

BMP180

mit dem Befehl `i2cdetect -y 1`

kann man sich die I2C Adresse anzeigen lassen.

```
pi@raspberrypi:~$ i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  77
pi@raspberrypi:~$
```

Wir benötigen noch eine Bibliothek um den Sensor ansprechen zu können.

Adafruit bietet solch eine Bibliothek an, mit dem folgendem Befehl kann man diese installieren (git muss installiert sein):

```
git clone
```

```
https://github.com/adafruit/Adafruit-Raspberry-Pi-Python-Code.
git
```

Als nächstes wechseln wir in das Verzeichnis der Bibliothek:

```
cd Adafruit-Raspberry-Pi-Python-Code/Adafruit_BMP085
```

Jetzt können wir das Beispiel Skript starten:

```
sudo python Adafruit_BMP085_example.py
```

Das Skript liest den Sensor aus und gibt die Werte im Terminal aus.

```
pi@raspberrypi:~/Adafruit-Raspberry-Pi-Python-Code/Adafruit_BMP085$ sudo python
Adafruit_BMP085_example.py
Temperature: 21.30 C
Pressure:    985.97 hPa
Altitude:    229.39
pi@raspberrypi:~/Adafruit-Raspberry-Pi-Python-Code/Adafruit_BMP085$
```

Sie haben es geschafft. Sie können jetzt unser Modul für Ihre Projekte nutzen.

Jetzt ist es an der Zeit, zu lernen und eigene Projekte zu erstellen. Das können Sie mit Hilfe von vielen Beispielskripten und anderen Tutorials tun, die Sie im Internet finden können.

Wenn Sie auf der Suche nach den qualitativ hochwertigen Produkten für Arduino und Raspberry Pi sind, sind Sie bei der AZ-Delivery Vertriebs GmbH genau richtig. Sie erhalten zahlreiche Anwendungsbeispiele, vollständige Installationsanleitungen, eBooks, Bibliotheken und Unterstützung durch unsere technischen Experten.

<https://az-delivery.de>

Viel Spaß!

Impressum

<https://az-delivery.de/pages/about-us>