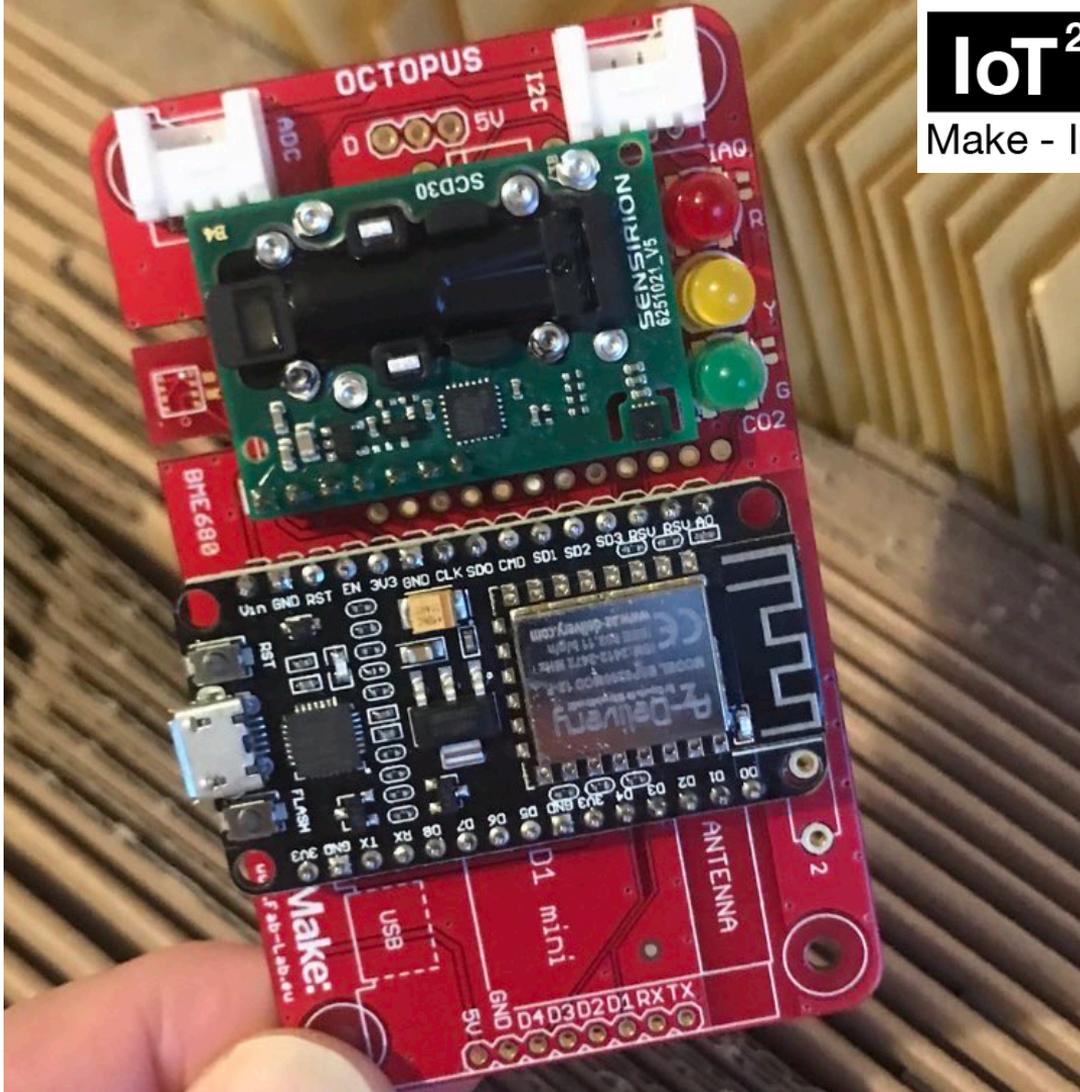


www.co2ampel.org



# IoT<sup>2</sup> Werkstatt

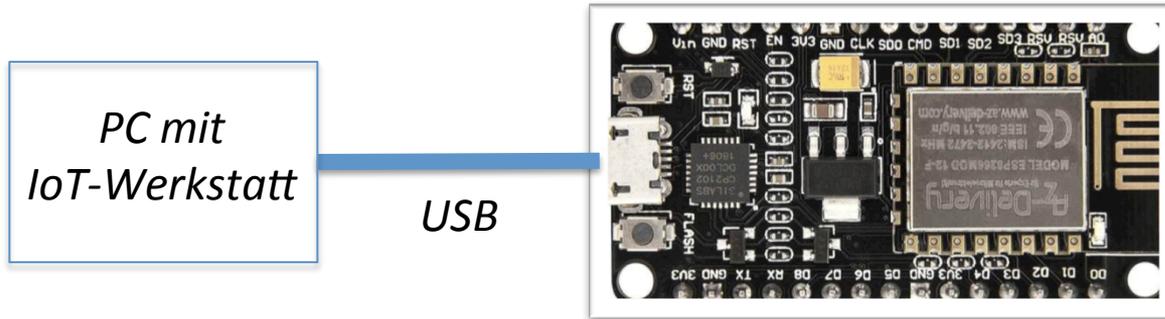
Make - Internet of Things Think

Bauanleitung  
für eine  
„CO2-Ampel“  
Lötterfahrung  
vorausgesetzt.

*Hierbei handelt  
es sich um kein  
medizintechn.  
Produkt, vor  
dem Einsatz  
muss ggfs der  
Sensor kalibriert  
werden. Einsatz  
nur auf eigene  
Verantwortung.*

# Vorbereitung: Test des ESP8266

*Test der Mikrocontroller-Platine vor dem Einlöten, d.h. ein „leeres“ Ardublock aufspielen um Funktion sicherzustellen.*



<https://www.umwelt-campus.de/iot-werkstatt/tutorials/schnellstart-octopus-anschiessen-und-einrichten>

# Vorbereitung: Arbeitsplatz herrichten

<https://praktische-elektronik.dr-k.de/Vorsicht-elektrostatiche-Entladung.html>

<https://www.conrad.de/de/ratgeber/technik-einfach-erklaert/richtig-loeten-lernen.html>

## Einfache Maßnahmen

- Kleidung aus Baumwolle
  - keine Wolle oder Kunststoff.
- Arbeitsplatte aus unbehandeltem Holz z.B. Sperrholz.
  - Papier und Pappe - kein Hochglanz - gehen zur Not.
- ESD-gefährdete Bauelemente in ESD-Tüten oder in ESD-Schaumstoff aufbewahren.
  - ESD-Tüten sehen metallisch oder rosa aus. ESD-Schaumstoff schwarz oder rosa.
- ➡ Plastiktüten oder Plastikboxen sind ungeeignet, aber auch Metallboxen und Alufolie.
- ! **Styropor** ist hochgefährlich.
- ! **Klebefilm wie Tesa** ist hochgefährlich.
  - Zur Not gehen Papier, Pappe oder Holzkästen - unbehandelt.
- ESD-Werkzeug verwenden
  - Eine ESD-Pinzette ist zu empfehlen.

## Verhalten

- Beim Hinsetzen und Aufstehen die Arbeitsplatte aus Holz berühren.
- Baugruppe oder Steckboard liegt auf der Arbeitsplatte.
- Am besten werden empfindliche Bauelemente als letzte eingesetzt.
  - zumindest sind die Anschlüsse nicht offen.
- Beim Einsetzen empfindlicher Bauelemente immer wieder die Arbeitsplatte aus Holz berühren,

Variante **Tindie** erfordert 1.1, 2.1 und 4.1

Variante **Make**: erfordert 1.1, 2.1, 3.1 und 4.2

## 1. Wahl der Ampelfunktion

1.1. LED Ampel (kann immer bestückt werden)

1.2. Neopixel-Ring (kann immer bestückt werden)

## 2. CO2-Sensor

2.1 SCD 30 - *alternativ S8 oder MH-Z19*

## 3. Erweiterungen *optional*

3.1 Feather Wing-Header Ampel (setzt D1 voraus)

3.2. I2C-Grove (ggf. mit Pullup, auch Grove-LCD hat interne Pullups)

3.3 Analog-Grove (mit Spannungsteiler, NodeMCU / D1 vs. Huzzah)

3.4 Deep-Sleep

## 4. CPU

4.1 NodeMCU V2

*Hinweis: vorher Widerstände bestücken, überlegen,  
ob Erweiterungen genutzt werden sollen*

4.2 Wemos D1

4.3 Feather Huzzah 8266

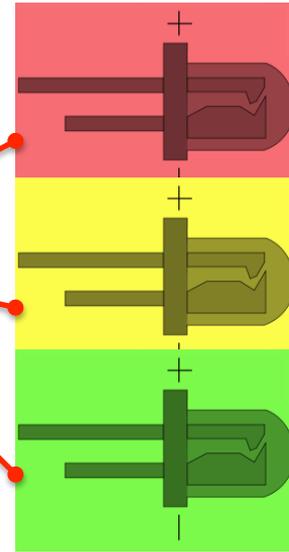
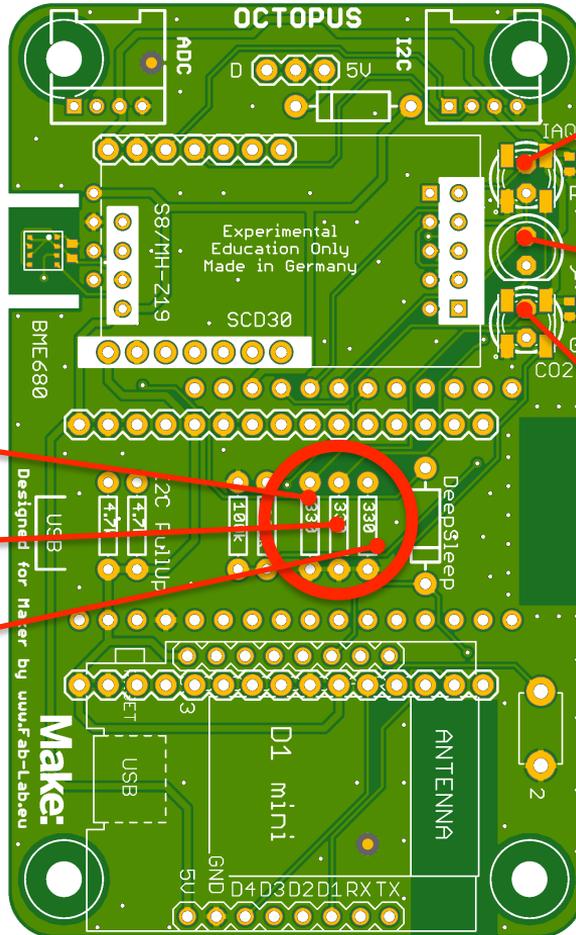
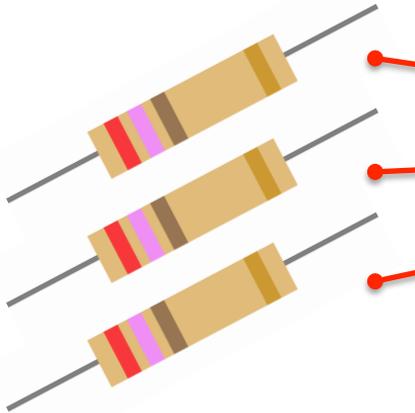
*Hinweis: vorher überlegen, ob Erweiterungen genutzt werden*

mehr:

[www.co2ampel.org](http://www.co2ampel.org)

# 1.1 LED Ampel

3x 220 o. 270  $\Omega$



GPIO12

GPIO13

GPIO14

3x 5mm LED

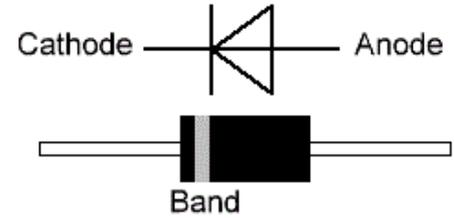
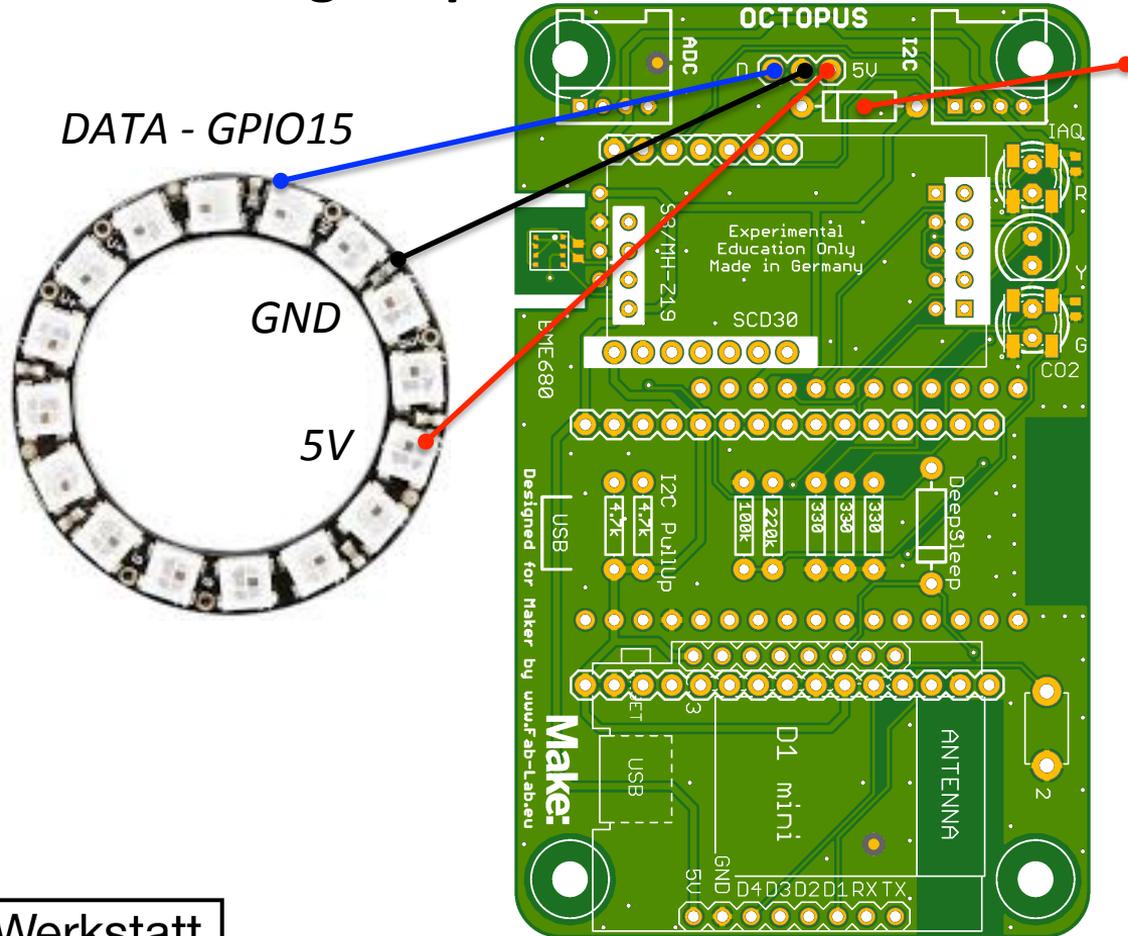
**Rot**

**Gelb**

**Grün**

*Orientierung  
wie abgebildet*

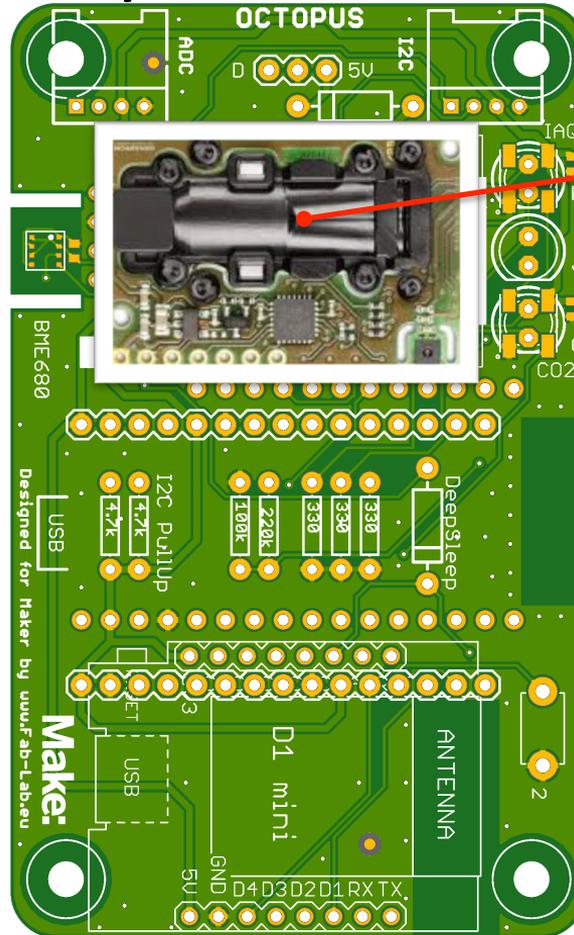
# 1.2 NeoPixel Ring Ampel



**1N4001**

*o. ähnliche  
Ziel: Spannungsabfall  
nutzen, um WS2812b  
mit 4.4. V zu betreiben.  
Notfalls Drahtbrücke*

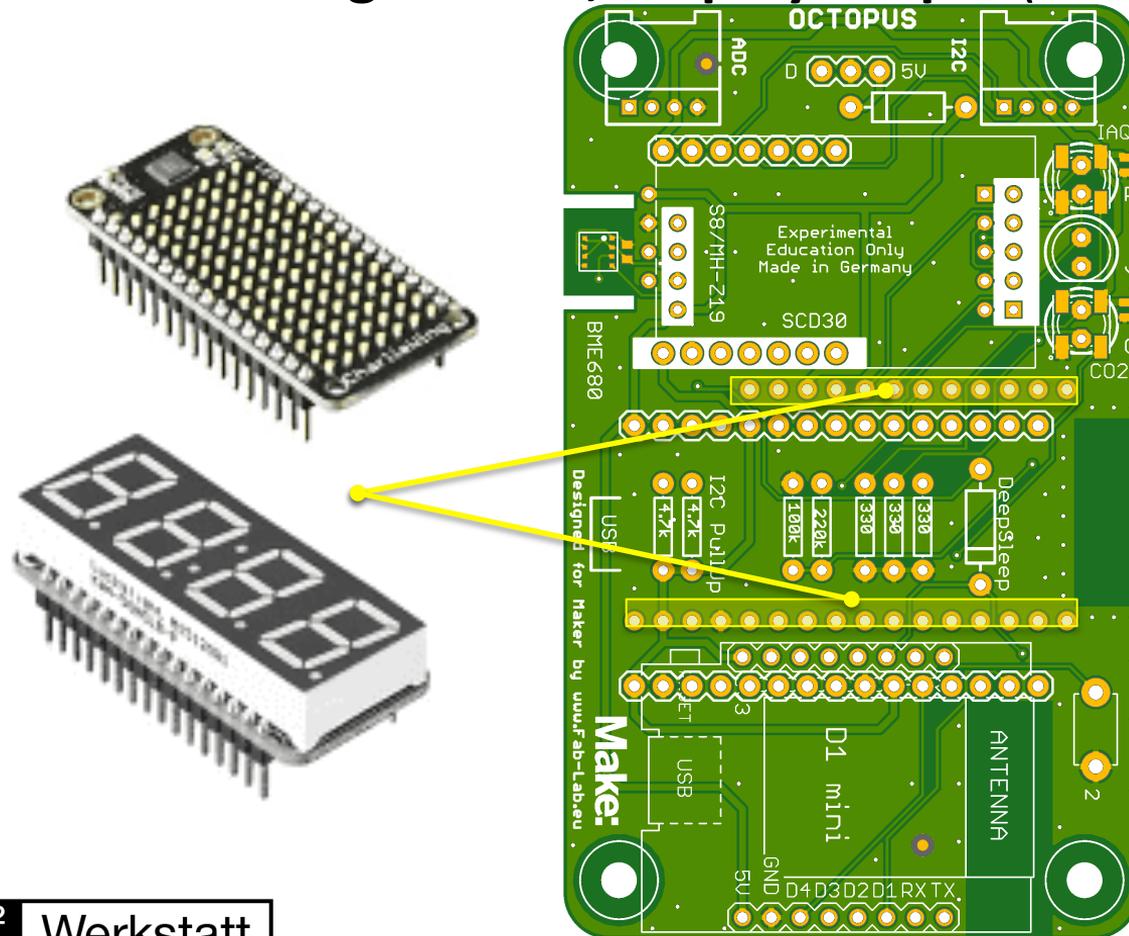
## 2.1 CO2 Sensor (SCD30 – I2C)



**SCD-30**

*Achtung:  
7 polige Stiftleiste,  
mit langen Füßen  
nach oben einlöten,  
auf 90 Grad  
Ausrichtung achten,  
SCD30 soll flach auf  
aufliegen, Bauteile  
oben*

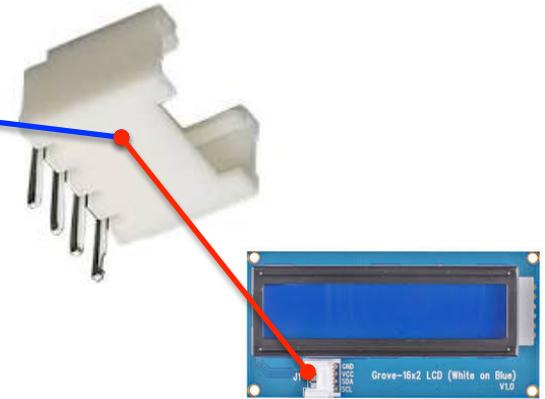
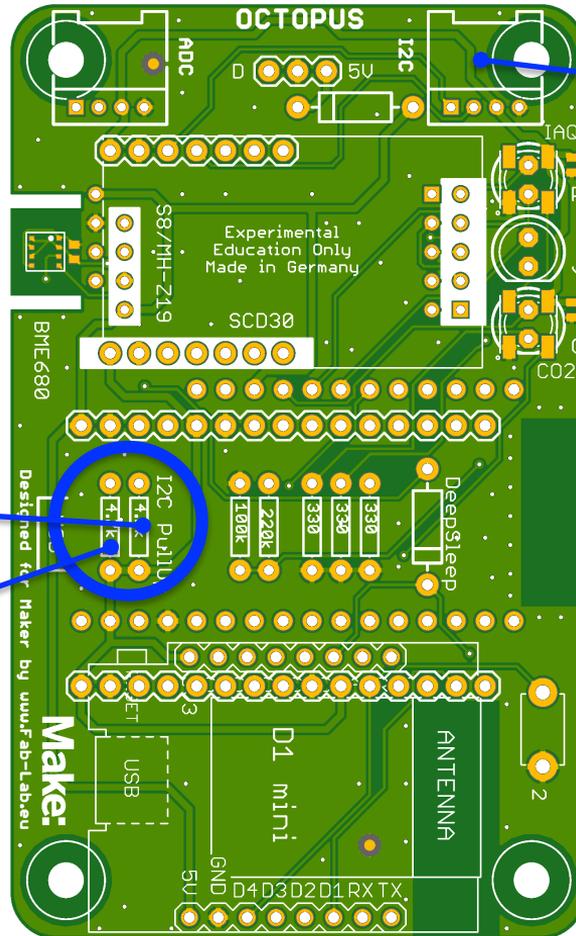
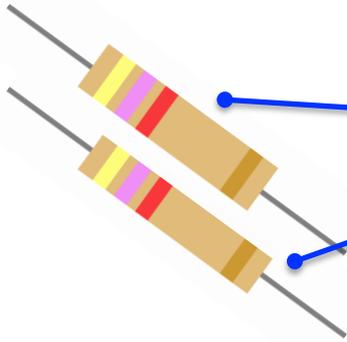
## 3.1 Feather Wing Header, Display Ampel (nur mit Wemos D1)



*Nur bei Bestückung mit WEMOS D1 - der dann freie Feather Sockel kann für ein Display genutzt werden (alle mit I2C) – ideal mit Sockel, dann bleibt das Display steckbar.*

## 3.2 I2C-Grove (optional)

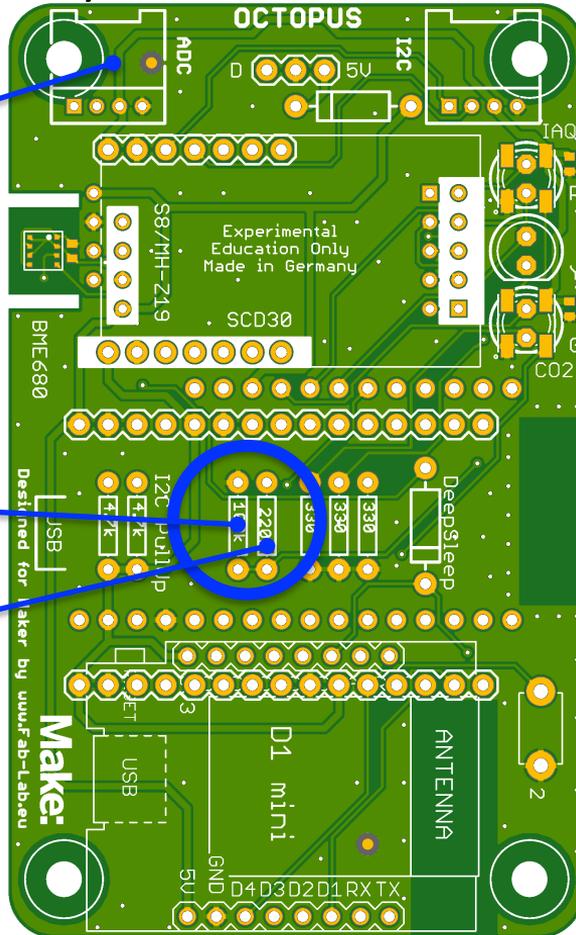
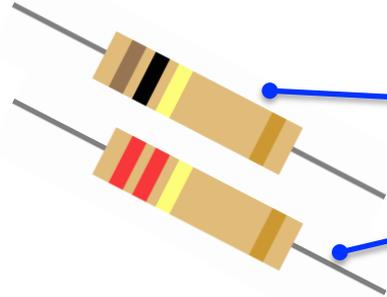
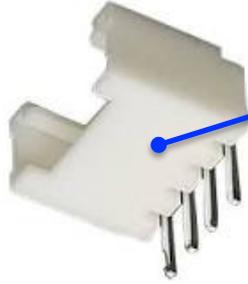
2x 4.7k  $\Omega$



z.B. Grove LCD  
(blue/white)

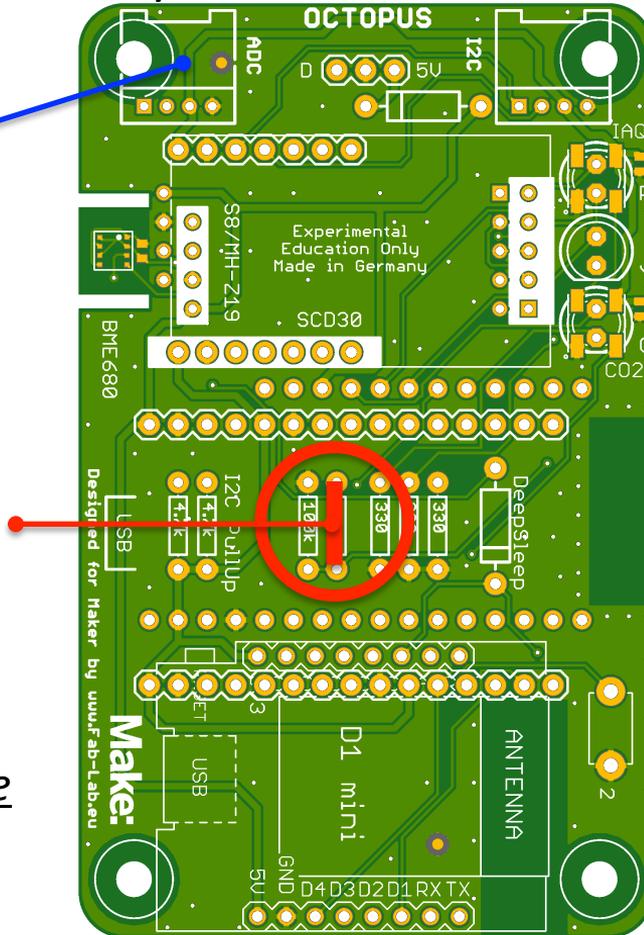
*Hinweis:  
viele I2C-Komponenten  
bringen die Pull-Up schon  
auf dem Board mit  
(Grove LCD, Sensirion  
SCD 30) - dann sind diese  
Widerstände nicht zu  
bestücken!*

### 3.3 Analog-Grove (optional)

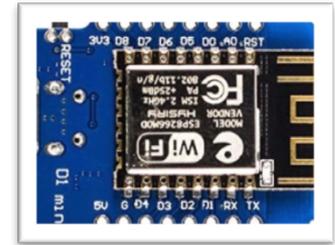


1x 100k  $\Omega$   
1x 220k  $\Omega$

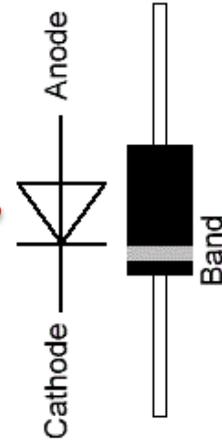
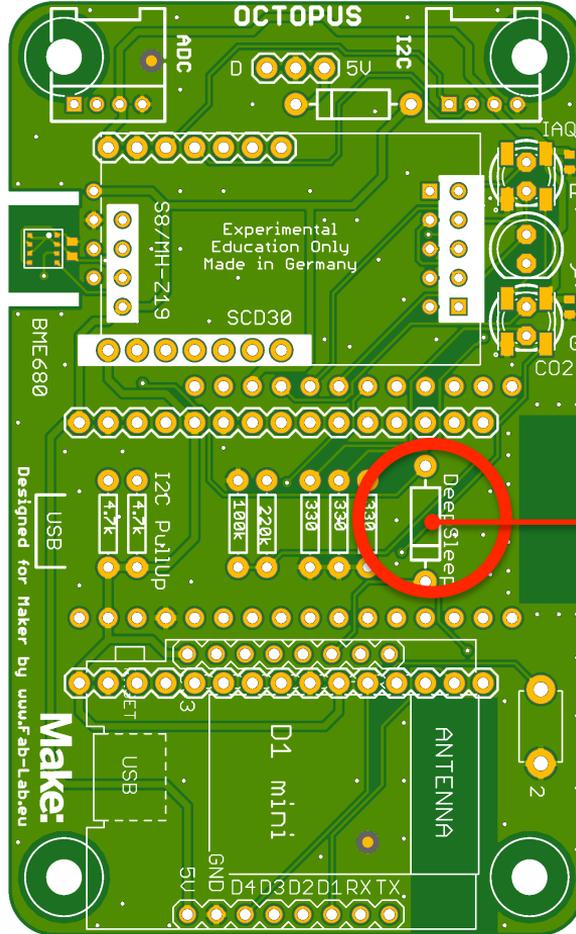
### 3.3 Analog-Grove (optional)



Bei Nodemcu V2 oder WemosD1 wird kein Spannungsteiler für A0 benötigt:  
100k nicht bestücken  
220k ersetzt Drahtbrücke



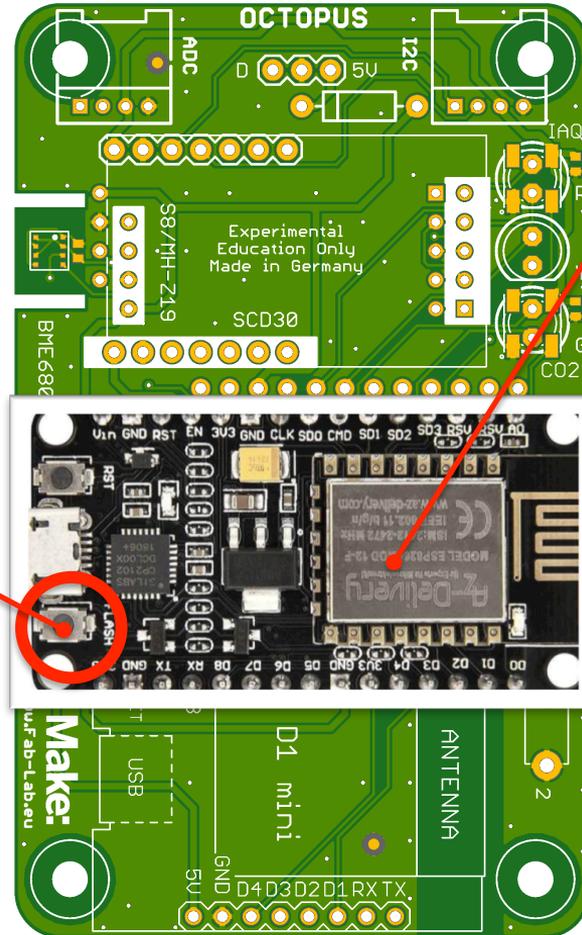
### 3.4 Deep Sleep (optional)



**1N4001**

# 4.1 Nodemcu V2

Taste für Kalibrierung  
GPIO0



## NODEMCU V2

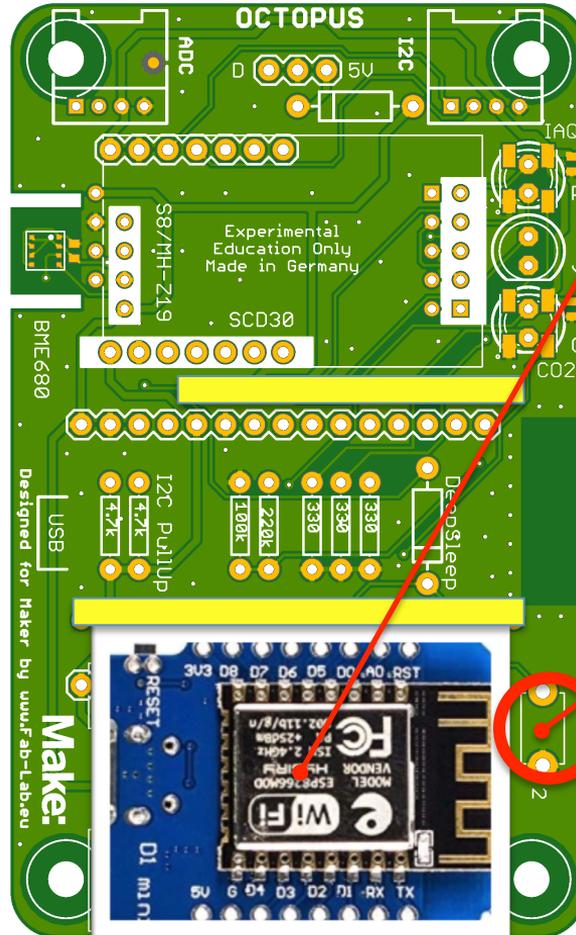
*Achtung:  
Nodemcu hat  
vorgelötet Stift-  
Leisten, diese  
können etwas  
aus dem Lot sein,  
ggfs. vorsichtig  
gerade biegen,  
Orientierung im Bild*

## 4.2 Wemos D1 - einlöten

*Abstand zur Leiterplatte so,  
dass der USB-Stecker  
eingesteckt werden kann!*

*Mit zwei beiliegenden  
Stiftleisten verlöten,  
Orientierung im Bild*

*Optional:  
Kann bei der Bestückung  
mit WEMOS D1 der dann  
freie Feather Sockel für  
ein Display genutzt werden  
(nur mit I2C)*



**WEMOS D1  
ESP8266**

Taste für Kalibrierung  
GPIO02

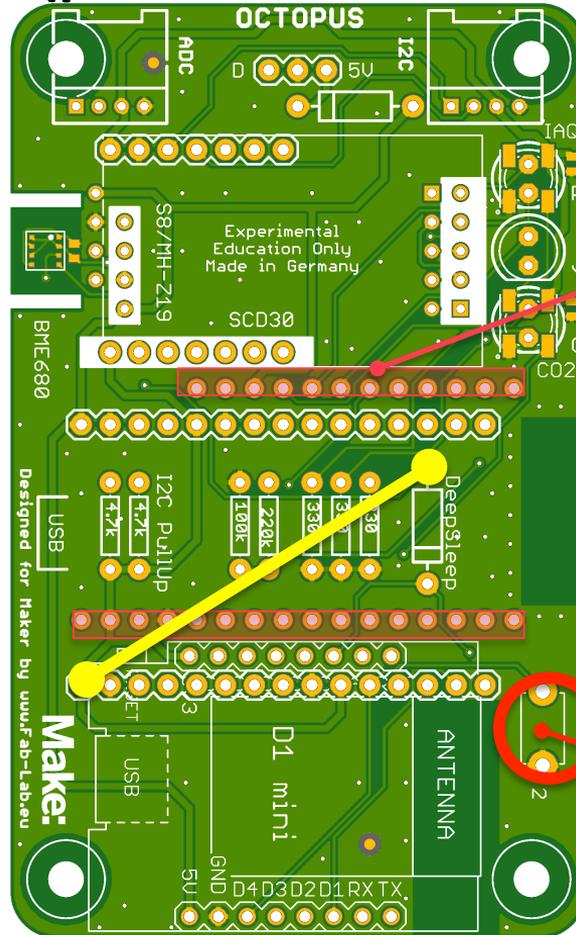
## 4.3 Feather Huzzah 8266 (pre-soldered header) - einlöten



optional:



isolierter Draht um den deepsleep zu aktivieren



**Feather Huzzah  
ESP8266**

*Achtung:  
Spannungsteiler für  
A0 einlöten (3.3)!  
Orientierung im Bild*

Taste für Kalibrierung  
GPIO0

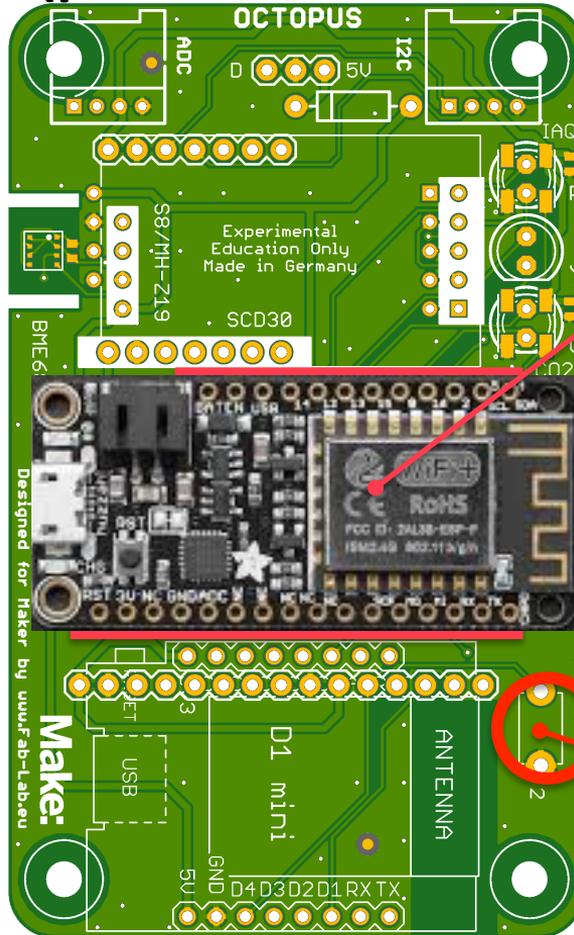
## 4.3 Feather Huzzah 8266 (pre-soldered header) - einlöten



optional:



isolierter Draht um den deepsleep zu aktivieren



**Feather Huzzah  
ESP8266**

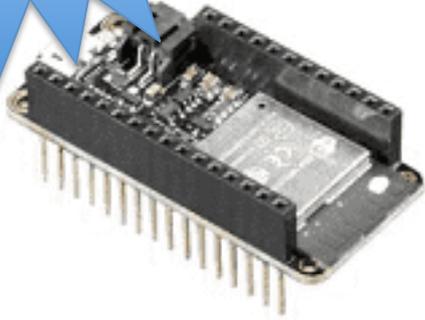
*Achtung:  
Spannungsteiler für  
A0 einlöten (3.3)!  
Orientierung im Bild*

Taste für Kalibrierung  
GPIO02



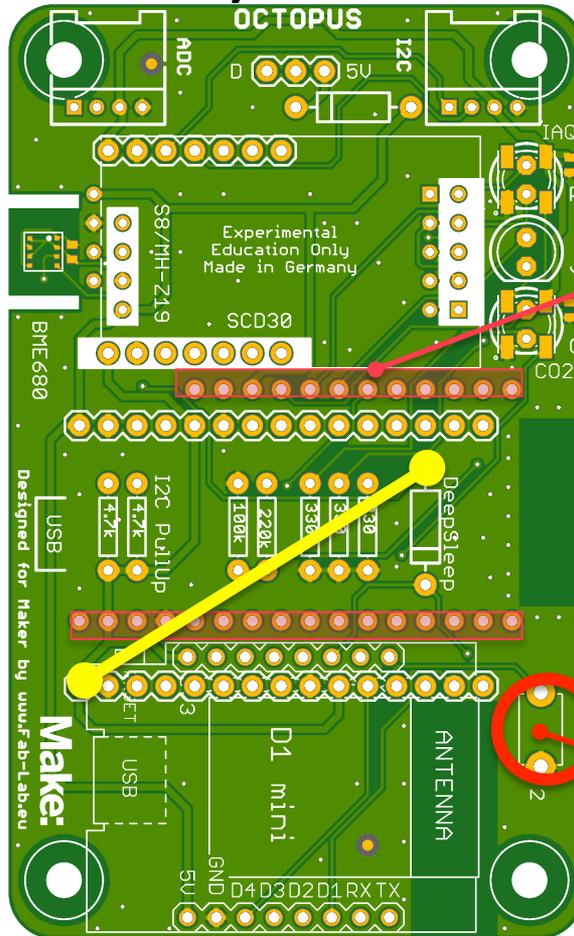
## 4.3 Feather Huzzah32 (stackable) – nicht mit Ardublock unterstützt

Profis  
ONLY



optional:

isolierter Draht um den  
deepsleep zu aktivieren



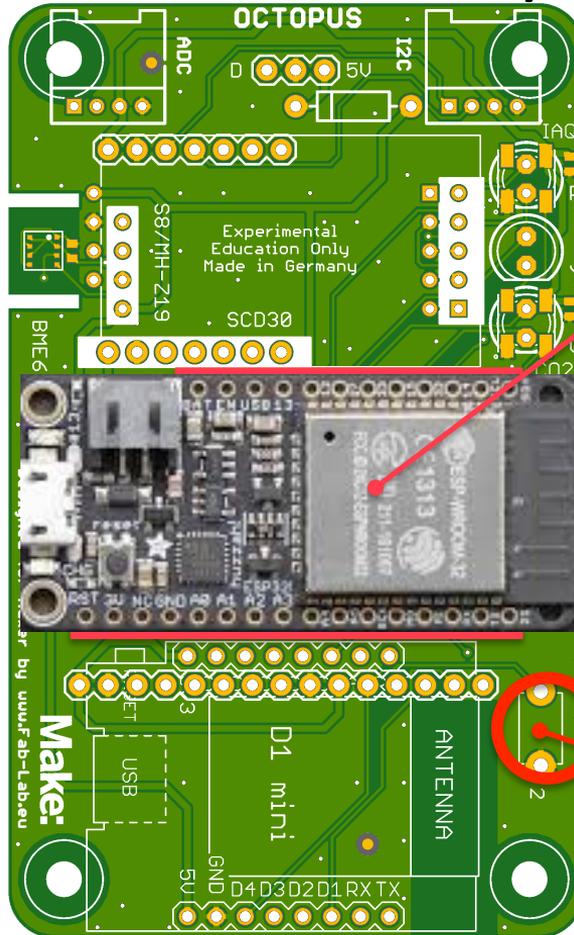
**Feather Huzzah32  
ESP32**

*Achtung:  
Spannungsteiler für  
A0 einlöten (3.3)!  
Orientierung im Bild*

Taste für Kalibrierung  
GPIO02

## 4.3 Feather Huzzah32 (pre-soldered header) – nicht mit Ardublock unterstützt

Profis  
ONLY



**Feather Huzzah32  
ESP32**

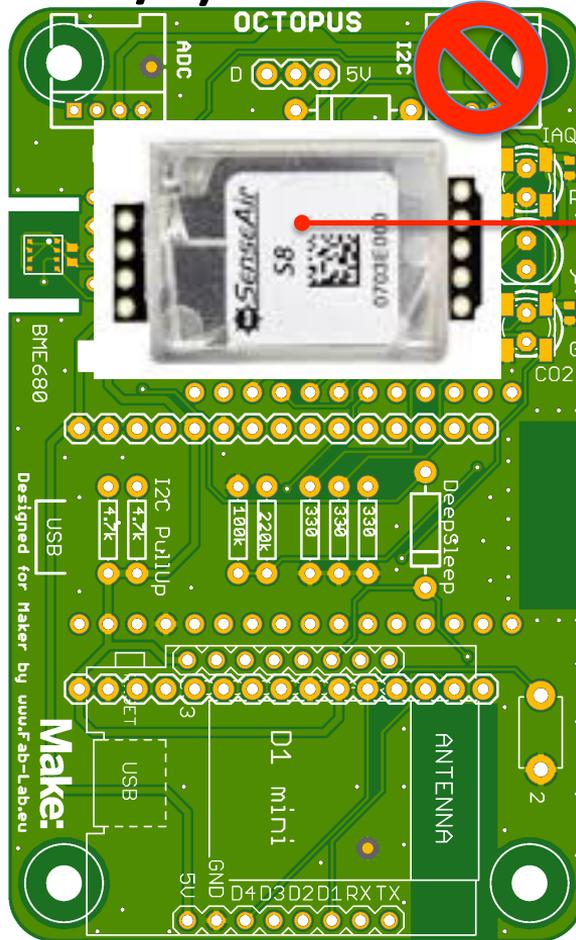
*Achtung:  
Spannungsteiler für  
A0 einlöten (3.3)!  
Orientierung im Bild*

Taste für Kalibrierung  
GPIO02

*optional:*

isolierter Draht um den  
deepsleep zu aktivieren

# optional - CO2 Sensor (S8 – 4/5)

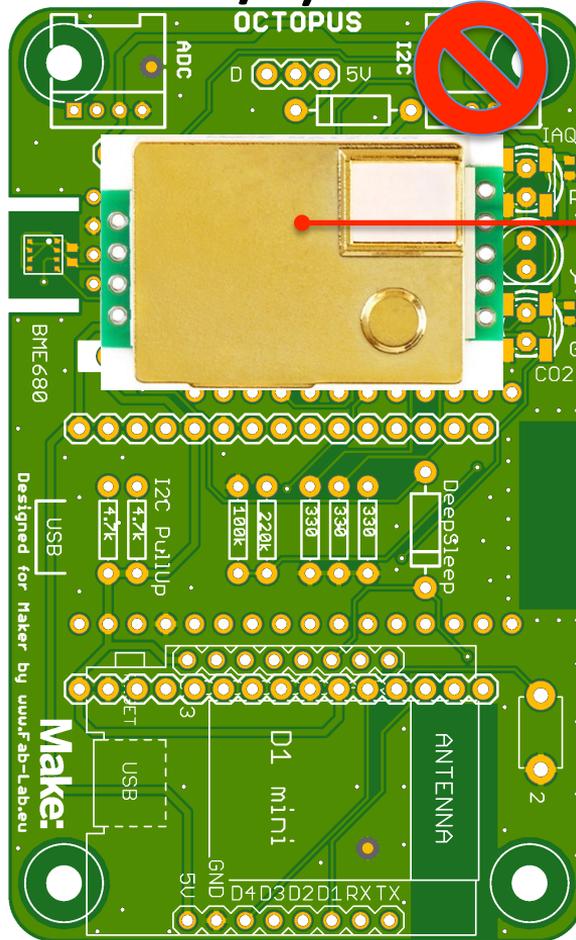


**SenseAir S8**

RX/TX = 4,5

*d.h. I2C kann in dieser Konfiguration nicht genutzt werden*

# optional - CO2 Sensor (MH-Z19 4/5)

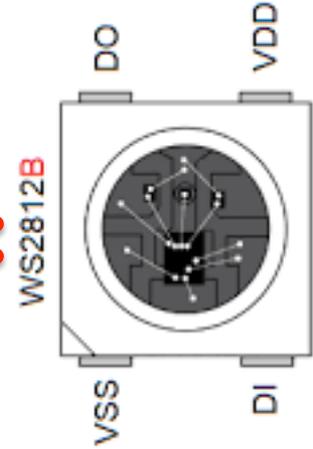
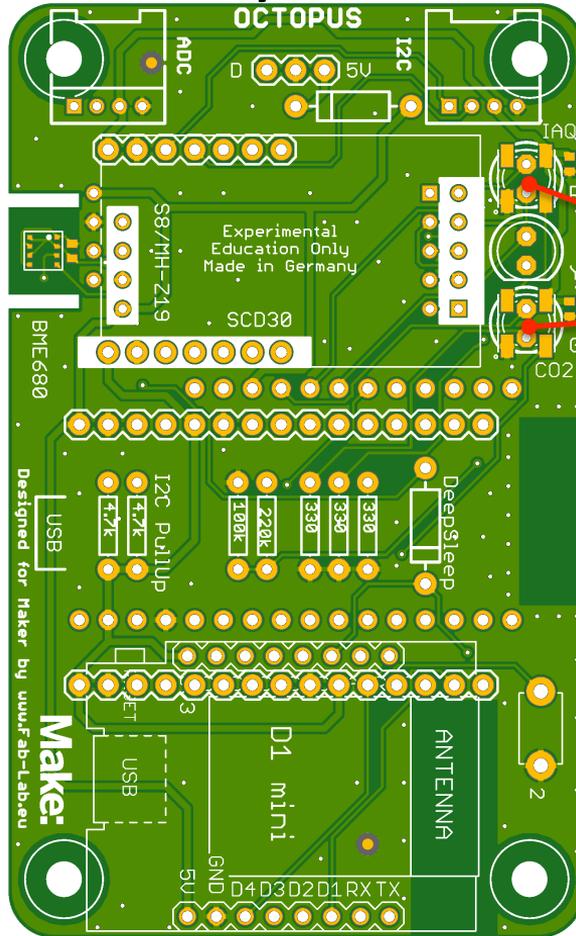


**MH-Z19**

RX/TX = 4,5

*d.h. I2C kann in dieser Konfiguration nicht genutzt werden*

# optional – 2x NeoPixel (WS2812B) ansatt 3 LEDs



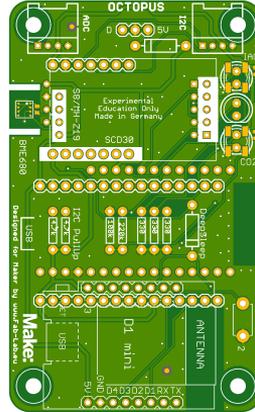
WS2812B

**WS2812B**

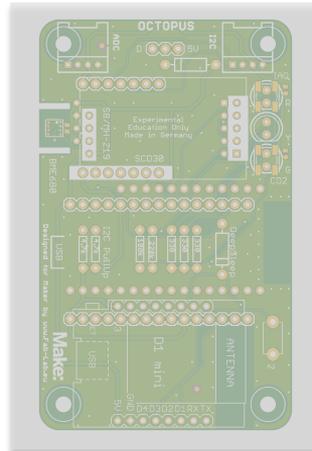
*Orientierung wie abgebildet*

# Versionshistorie

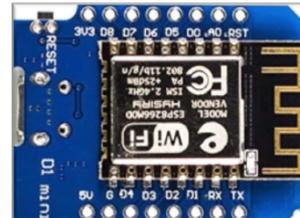
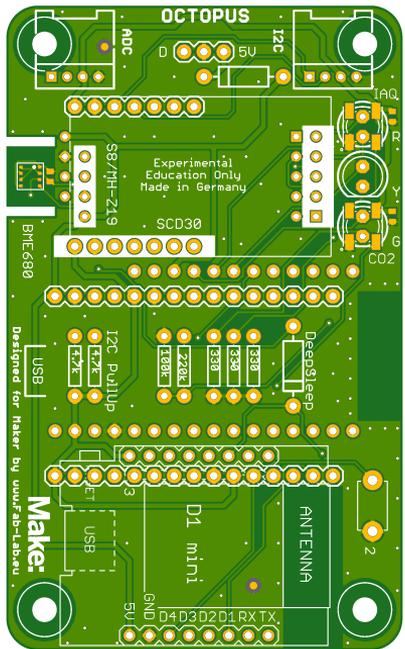
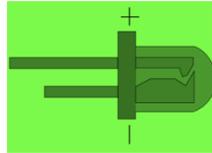
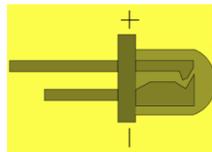
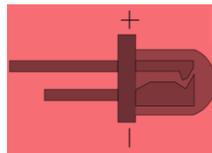
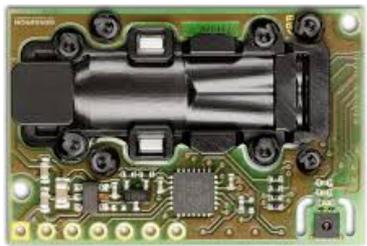
V1/V2



V3



- RST vom Feather Huzzah  
ESP8266 nicht mit deepsleep  
Diode verbunden: fix  
 isolierter Draht um den  
deepsleep zu aktivieren (4.3)
- Feather nur für I2C
- V2: NeoPixel und BME680
- RST fixed
- Feather I2C und SPI
- Support für LoRaWAN WING



**IoT<sup>2</sup> Werkstatt**  
Make - Internet of Things Think

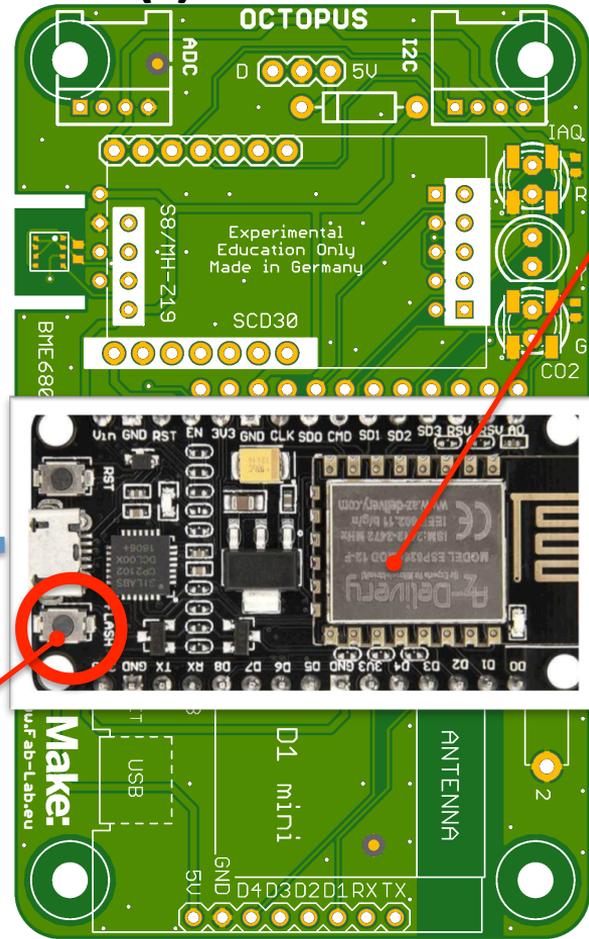
# Bauteileliste

Nodemcu V2	ESP8266 mit Stiftleisten gelötet
Wemos D1	ESP8266 mit Stiftleisten
Adafruit 3213	Feather HUZZAH ESP8266 WiFi mit Stapelstiftleisten gelötet
Adafruit 3046	Feather HUZZAH ESP8266 mit Stiftleitsten gelötet
Adafruit 2821	Feather HUZZAH ESP8266
Adafruit 2945	NeoPixel FeatherWing Add-on
Adafruit 3108	4-stellige 7-Segment-Anzeige FeatherWing rot
Adafruit 3134	15 x 7 CharliePlex rote LED-Matrix-Anzeige FeatherWing
Adafruit 1586	NeoPixel-Ring mit 24 x 5050-RGB-LEDs
Seeed Studio 104020111	Grove 16 x 2 LCD Weiss auf Blau
Seeed Studio 110990210	4-poliger Stecker-Jumper auf Grove
<i>Adafruit 3619</i>	<i>Montiertes HUZZAH32 ESP32 Feather Board mit Stapelstiftleisten gelötet</i>
<i>Adafruit 4128</i>	<i>2,13" 212x104 3-farbiges eInk-Display FeatherWing</i>
<i>Adafruit 2895</i>	<i>Mini-Relais-FeatherWing ohne Selbsthaltung</i>
<i>Adafruit 4309</i>	<i>Grove Shield FeatherWing Adapter für Particle Mesh</i>



Profis  
ONLY

# Mit dem PC per USB – Daten(!)kabel verbinden



**NODEMCU V2**

*NodeMCU hat vorgelötet Stift-Leisten, diese können etwas aus dem Lot sein, ggfs. vorsichtig gerade biegen,*

Orientierung im Bild

PC mit IoT-Werkstatt

Taste für Kalibrierung GPIO0

# Quickstart – IoT Werkstatt (Arduino DIE & Ardublock mit allen Libs)

Quickstart und Download Link:

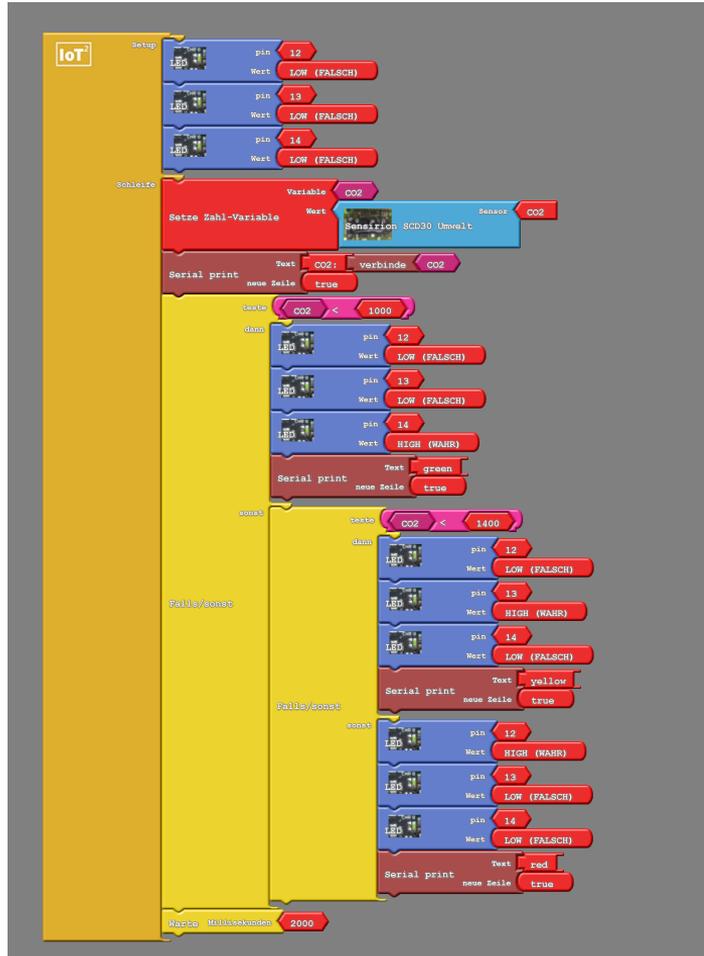
[https://www.umwelt-campus.de/fileadmin/  
Umwelt-Campus/IoT-Werkstatt/octopus/  
Quickstart.pdf](https://www.umwelt-campus.de/fileadmin/Umwelt-Campus/IoT-Werkstatt/octopus/Quickstart.pdf)

Quellcode der Beispiele:

[https://d3s5r33r268y59.cloudfront.net/boms/  
21632/2020-10-29-12-50-51/CO2-Ampel-DIY-Octopus-  
SCD30.zip](https://d3s5r33r268y59.cloudfront.net/boms/21632/2020-10-29-12-50-51/CO2-Ampel-DIY-Octopus-SCD30.zip)

Github: <https://github.com/make-iot/CO2-Ampel>

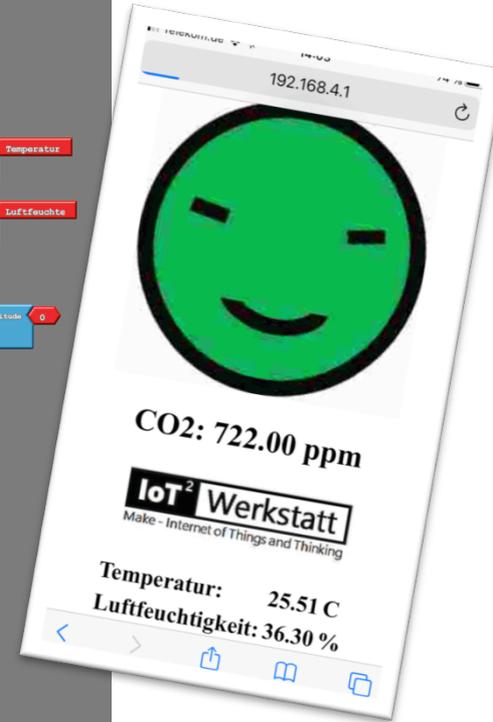
# Beispielprogramm in Ardublock – LED Ampel



# Beispielprogramm in Ardublock – LED Ampel, Webserver und AP, Kalibrierung

The screenshot shows an Arduino IDE sketch in Ardublock. The sketch is organized into sections: **Setup**, **Schleife** (Loop), and **Befehle** (Commands).

- Setup:** Configures an Access Point with network name 'MeinIoTLAN' and password '1234567890123456'. It sets pins 12, 13, and 14 as outputs and initializes a CO2 variable to 0. It also sets up a serial connection at 9600 baud.
- Schleife:** A loop that checks CO2 levels. If CO2 < 1000, it turns on LEDs at pins 12, 13, and 14 (LOW state). If CO2 > 1400, it turns off LEDs (HIGH state). It prints the current CO2 value and the state of the LEDs to the serial monitor.
- Befehle:** A separate block for commands, including a 'CO2-Server' block with a green smiley face icon. It defines variables for CO2 (ppm), Temperatur (C), Luftfeuchtigkeit (%), and includes a calibration function for the SCD30 sensor.



# Beispielprogramm in Ardublock

Beispiel mit Kalibrierung und Web-Browser: `\IoTW\Beispiele\CO2_Ampel`  
`\CO2_Ampel_nodeMCUNEopixelRing.abp`

