

Projekt: CO2-Messgerät

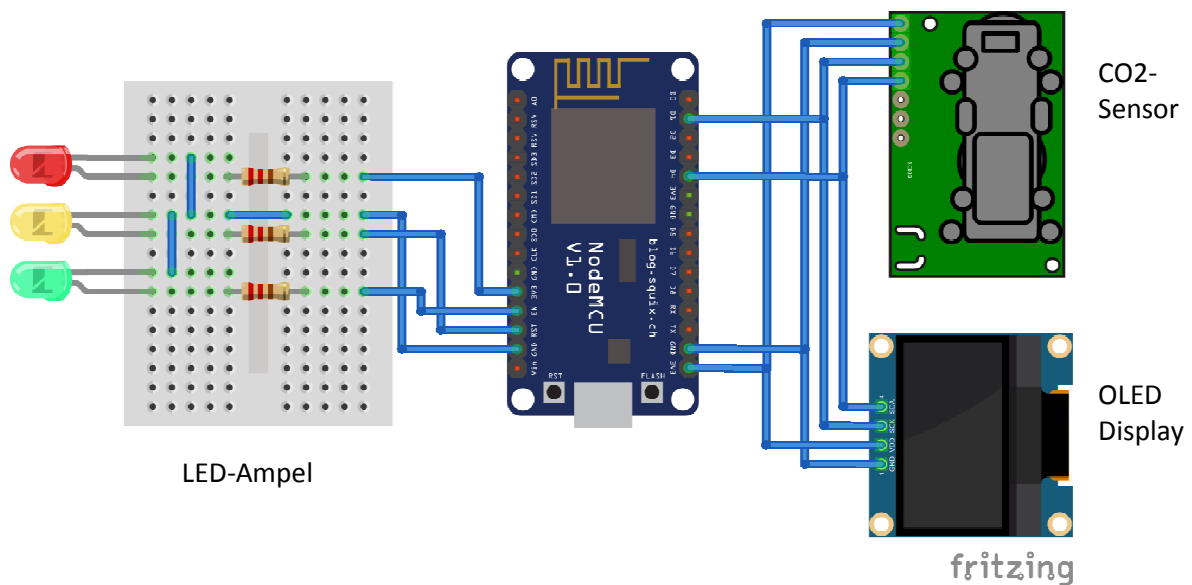
Bauteile:

- ESP32 Controller Board (wurde aufgrund des hervorragenden Preis-Leistungsverhältnis gewählt und bietet Raum für zukünftige Erweiterungen)
- OLED Display 0,96 Zoll (zum Anzeigen der Messwerte)
- CO2 Sensor (misst CO2-Gehalt, Temperatur und Luftfeuchtigkeit)
- dreifarbige Ampel (im Schaltplan nur symbolisch dargestellt)

Bezugsquellen:

<https://www.reichelt.de/index.html?ACTION=446&LA=3&nbc=1&q=debo%20jt%20esp32>
https://www.reichelt.de/entwicklerboards-display-0-96-oled-display-ssd1306-debo-oled2-0-96-p266107.html?&trstct=pos_1&nbc=1
<https://shop.watterott.com/Sensirion-SCD30-CO2-Kohlendioxid-Sensor>
https://www.reichelt.de/micro-bit-ampelsystem-rt-ge-gn--bbc--stop-led-p284665.html?CCOUNTRY=445&LANGUAGE=de&trstct=pos_1&nbc=1&r=1

Schaltplan:



Aufbauhinweise

Mit Dxx ist der jeweilige GIOxx bezeichnet.

Displayanschluss:

ESP-Board			
GND	3V3	D21	D22
GND	VCC	SDA	SCL
Display mit 4 Pins			

CO2-Sensoranschluss

ESP-Board

GND	3V3	D21	D22
GND	VCC	SDA	SCL
PIN1	PIN2	PIN3	PIN4

Sensor hat 7 Pins, nur die ersten vier werden benötigt

Ampelanschluss

ESP-Board

GND	D12	D13	D14
GND	Grün	Gelb	Rot

Ampel hat 4 Anschlüsse

Einrichten der Arduino-Umgebung

ESP32:

Boardverwalter hinzufügen:

Datei->Einstellungen->zusätzliche Boardverwalter URL hinzufügen

hier https://dl.espressif.com/dl/package_esp32_index.json eingeben. Mehrere URLs können durch ein Komma getrennt werden.

Werkzeuge->Board->Boardverwalter

hier nach ESP32 suchen und installieren

Werkzeuge->Board

Board auswählen: ESP32 DEV Module

Serielle Schnittstelle auf 115200 einstellen

Hinweis: beim Upload muss der Boot Taster am Board gedrückt werden

OLED Display

Werkzeuge-> Bibliotheken verwalten

hier nach ESP32 OLED suchen und der Eintrag zum SSD1306 auswählen und installieren

CO2-Sensor

Werkzeuge-> Bibliotheken verwalten

hier nach CO2 suchen und der Eintrag zum SCD30 auswählen und installieren

Eine sehr gute Erstanleitung findet man auf der Make: Seite von heise online:

<https://m.heise.de/make/artikel/Grosser-Bruder-Espressif-ESP32-3256039.html?seite=all>

Programm:

Das Programm wurde mit der Arduino Umgebung 1.8.13 entwickelt.

```
/* CO2 Messgerät mit ESP32, CO2 Sensor (I2C), OLED Display (I2C) und LED
Ampel (GPIO)
 * V1.0
 * Datum 19.11.2020
 */

/* CO2 */
#include <Wire.h>

#include "SparkFun_SCD30_Arduino_Library.h"
SCD30 airSensor;

/* OLED Display */
#include "SSD1306Wire.h"

SSD1306Wire display(0x3c, SDA, SCL);

#define MAXVALUE 10
int CO2, Temp, Humi;
int value[MAXVALUE] = {0}, i=0, k;
double avgCO2;

// LED wiring
#define LED_GRUEN 12
#define LED_GELB 13
#define LED_ROT 14

void setup() {
/* CO2 */
  Serial.begin(115200);
  Wire.begin();

  if (airSensor.begin() == false)
  {
    Serial.println("Air sensor not detected. Please check wiring.
Freezing...");
    while (1)
      ;
  }

  //Erlangen is ~294m above sealevel
  airSensor.setAltitudeCompensation(294); //Set altitude of the sensor in m

/* OLED Display */
  display.init();
  display.flipScreenVertically();
  display.setFont(ArialMT_Plain_16);

  //LED Beschaltung
  pinMode(LED_GRUEN, OUTPUT);
  pinMode(LED_GELB, OUTPUT);
  pinMode(LED_ROT, OUTPUT);
}

void loop() {
/* CO2 */
  if (airSensor.dataAvailable())
```

```

{
  CO2 = airSensor.getCO2();
  Temp = airSensor.getTemperature();
  Humi = airSensor.getHumidity();

  // calculate the avgerage of MAXVALUE last values
  value[i]=CO2;
  i++;
  if(i==MAXVALUE) i=0;
  for (k=0; k < MAXVALUE; k++)
    avgCO2+=value[k];

  avgCO2/=MAXVALUE;
  CO2=round(avgCO2);

  /* OLED Display */
  display.clear();

  display.setTextAlignment(TEXT_ALIGN_LEFT);
  display.setFont(ArialMT_Plain_16);
  display.drawString(0, 0, "Luftqualität");
  display.drawString(0, 16, "co2 (ppm):");
  display.drawString(84, 16, String(CO2));
  display.drawString(0, 32, "temp (C):");
  display.drawString(92, 32, String(Temp));
  display.drawString(0, 48, "humidity(%):");
  display.drawString(92, 48, String(Humi));

  // write the buffer to the display
  display.display();

  // Ansteuerung der Ampel
  if(CO2 < 1000)
  {
    // Grün einschalten
    digitalWrite(LED_GELB, LOW);
    digitalWrite(LED_ROT, LOW);
    digitalWrite(LED_GRUEN, HIGH);
  }
  else if (CO2 < 1500)
  {
    // Gelb einschalten
    digitalWrite(LED_ROT, LOW);
    digitalWrite(LED_GRUEN, LOW);
    digitalWrite(LED_GELB, HIGH);
  } else
  {
    // Rot einschalten
    digitalWrite(LED_GELB, LOW);
    digitalWrite(LED_GRUEN, LOW);
    digitalWrite(LED_ROT, HIGH);
  }

  delay(2000);
}

```