

Smarter CO₂ Warner



Diese Arbeit ist von Silas Wester 9D Gauß Gymnasium Worms

Inhaltsverzeichnis

Ziel.....	3
Warum ist ein CO ₂ Warner sinnvoll?.....	3
Ziel 1: Eine Basis die CO ₂ messen kann.....	3
Was ist ein Octopus Board?.....	3
Aufbau: Plug and code.....	3
Ziel 2: Die CO ₂ Werte mit einem Webserver erreichbar machen.....	4
Ziel 3: Eine intuitive Anzeige-Möglichkeit.....	4
Was ist eine LED Matrix?.....	4
Aufbau der Matrix:.....	4
Was will ich auf der Matrix anzeigen?.....	6
Programmierung der LED-Matrix:.....	6
Erster Testlauf.....	7
Quellen.....	8

Ziel

Mein Ziel ist es, einen praktischen CO₂ Warner zu bauen der leicht abzulesen ist und nicht ablenkt. Hauptsächlich baue ich ihn für den Anwendungszweck in der Schule, da meine Schule noch keine CO₂ Warner bekommen hat.

Warum ist ein CO₂ Warner sinnvoll?

Vom CO₂ Gehalt in der Luft kann man Rückschlüsse auf die Raumluftqualität schließen da der CO₂ Wert steigt, wenn sich mehrere Personen in einem Raum befinden und nicht gelüftet wird. Als Richtwert, wann man lüften sollte, habe ich 1000ppm gewählt, weil es z.B. vom Umweltbundesamt als Grenzwert angegeben wird.

Ziel 1: Eine Basis die CO₂ messen kann.

Hier für habe ich ein Octopus Board und den SCD30 CO₂-, Temperatur- und Feuchtigkeitssensor verwendet. Ich habe eine Version des Sensors mit dem 12C BUS, weshalb ich nur 4 Leitungen brauche um den Sensor an mein Board anzuschließen.

Was ist ein Octopus Board?

Das Octopus Board basiert auf dem ESP-12F und ist ein Arduino-IDE kompatibles Entwicklerboard welches vom Umweltcampus der Hochschule Trier gebaut wird. Das Board ist interessant, da es z.B. einen WLAN Chip und einen 12C BUS Anschluss (für den CO₂ Sensor) verbaut hat.

Aufbau: Plug and code

Den CO₂ Sensor musste ich nur an das Octopus Board anschließen und konnte direkt mit Programmieren beginnen.

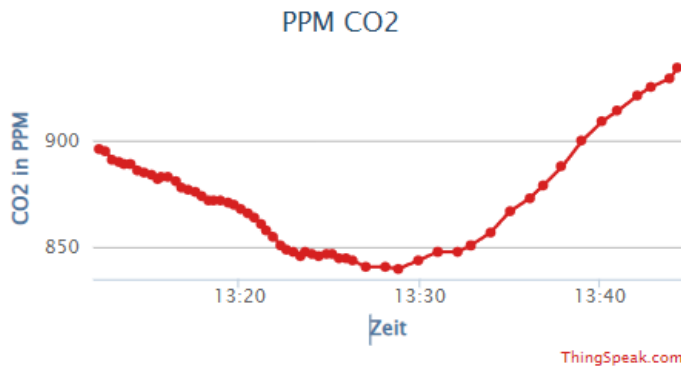
Für den Code konnte ich große Teile des Beispiel Codes für den CO₂ Sensor verwenden. Zum Visualisieren der Werte habe ich die auf dem Octopus board aufgelöteten NeoPixel LED verwendet. Bei einem CO₂ Wehrt unter 600 ppm Leuchtet die LED grün, zwischen 600ppm und 1000ppm gelb und über 1000ppm rot. Außerdem gibt das Board den CO₂ Wert auch über Serial.print an einen angeschlossenen PC aus.

Ziel 2: Die CO₂ Werte mit einem Webserver erreichbar machen.

Hierfür habe ich den Anbieter ThingSpeak benutzt, da er einfach zu benutzen ist und genau richtig für meinen Anwendungszweck. Auch hier habe ich wieder Teile des Beispiel Codes verwendet.

Jetzt misst der Sensor alle 2 sec. CO₂ und das Octopus board schickt die Werte alle 10 sec. zu meinem ThingSpeak Server.

Bsp.:



Ziel 3: Eine intuitive Anzeige-Möglichkeit

Um den CO₂ Wert anzuzeigen, habe ich eine LED Matrix mit 16x16 Pixeln aus WS28012b LEDs gebaut. Das hat den Vorteil, dass sie im Gegensatz zu einem normalen Display deutlich günstiger ist (insbesondere, wenn sie so groß ist, dass man sie in der Schule neben der Tafel hängend gut erkennen kann). Außerdem lenkt eine LED matrix auch nicht so ab wie ein Display.

Was ist eine LED Matrix?

Prinzipiell funktioniert eine LED Matrix ähnlich wie ein normales Display. In meinem Fall hat es aber nur eine Auflösung von 16x16 Pixeln. Ich habe hier 256 einzeln ansteuerbare LEDs verbaut, die ich dann so programmiert habe, dass man Zahlen anzeigen kann. Natürlich nur mit einer Auflösung von 16x16 Pixeln was aber für 3 Ziffern gut ausreicht.

Aufbau der Matrix:

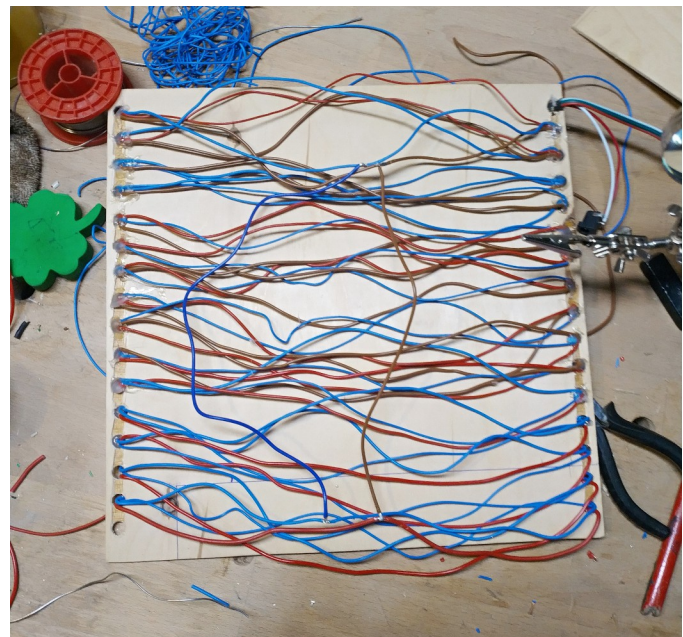
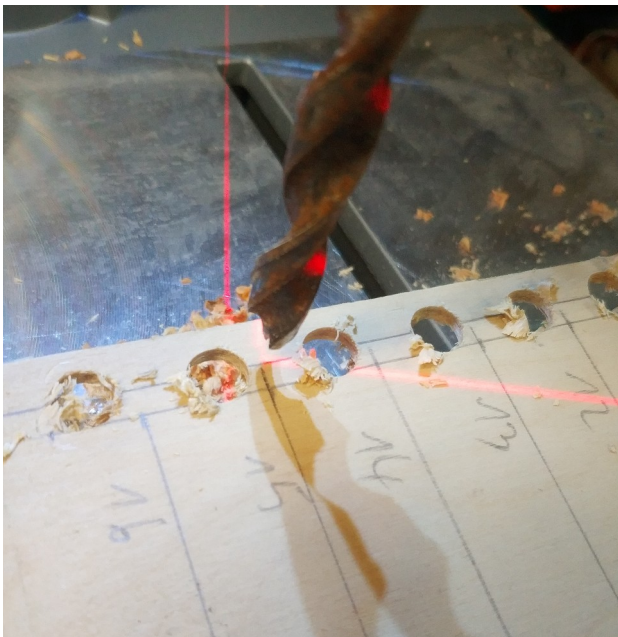
Die Basis ist eine rechteckige Holzplatte auf die ich WS28012b LED-Streifen geklebt habe.

An die Stromkabel habe ich noch eine Buchse für ein 5V 4A Netzteil gelötet. Nach 16 LEDs habe ich die Streifen abgeschnitten und in einer neuen Reihe angefangen.

Als nächstes musste ich die LED-Streifen mit Kabeln verbinden. Hierbei habe ich die Streifen so gelegt, dass die LEDs immer von links nach rechts angeordnet sind. Das hat den Vorteil, dass es leichter zu programmieren ist. Außerdem wird nur diese Anordnung von meinem gewählten Programm unterstützt.

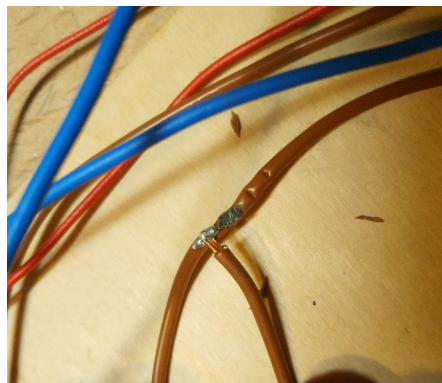
Der Nachteil ist, dass ich die Kabel hinter der Matrix durchführen muss um wieder links am nächsten LED-Streifen anzukommen.

Damit ich die Matrix später in ein Gehäuse bauen kann und der Rand gleichmäßig ist, habe ich für die Kabel an jeder Seite der Reihen ein Loch gebohrt.



Beim Verlöten muss man aufpassen, dass man die Kabel vorher so biegt wie man sie braucht, da sie sich beim Lötén mit Lötinn vollsaugen und dann nicht mehr flexibel sind. Bei der 3. Reihe habe ich nämlich versucht die Kabel nach dem Lötén noch einmal zu verbiegen und hab dabei das Löt-pad von der Datenleitung abgerissen. Als Lösung habe ich dann das Kabel an die Unterseite des Streifens gelötet.

Beim ersten Test ist mir aufgefallen, dass wenn alle LEDs weiß leuchten sollen, die Farbe der letzten LEDs eher gelb ist. Dieser Effekt entsteht durch den Spannungsabfall. Das Problem habe ich behoben indem ich die letzten Stromleitungen mit den ersten verbunden habe.



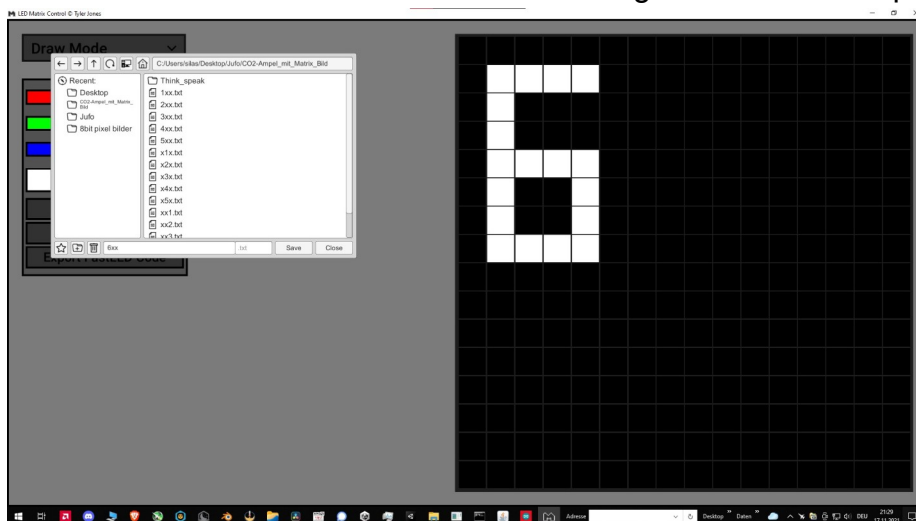
Was will ich auf der Matrix anzeigen?

- den CO₂ Wert mit 3 Ziffern
- einen Rahmen, der je nach CO₂ Wert die Farbe wechselt (grün, gelb, rot)
- eine große rote Meldung, wenn der CO₂ Wert über 999ppm steigt

Programmierung der LED-Matrix:

Um die LEDs einzeln anzusteuern, habe ich die FastLED library verwendet. Um 256 LEDs anzusteuern ist das aber sehr aufwendig, da ich für jede LED eine Zeile Code brauche um ihr zu sagen in welcher Farbe sie leuchten soll. Im Internet habe ich das Programm LEDMatrixControl von TylerTimoJ. In diesem Programm kann man auch unter anderem auf einer Fläche von 16x16 Pixeln zeichnen und sich die Bilder als Code für die FastLED library exportieren.

Dann habe ich die Zahlen von 0-9 für alle 3 Ziffern gezeichnet und exportiert.



Als nächstes musste ich ein Programm entwickeln, welches den CO₂ Wert bekommt und ausgibt welche LEDs wie leuchten müssen. Dafür muss es erst mal die Zahl in ihre Ziffern auftrennen da ich die Ziffern einzeln gezeichnet habe. Dafür habe ich mir zu Nutzen gemacht, dass bei einer Integer Variablen die Kommastellen wegfallen. Die Zahl wird durch 100 geteilt und ich erhalte die 100er Ziffer. Für die 10er Ziffer wird von der Zahl die 100er Ziffer mal 100 abgezogen und das ganze nochmal durch 10 geteilt. Für die 1er Ziffer wird auch die 100er Ziffer mal 100 abgezogen und dann das ganze durch die 10er Ziffer mal 10 subtrahiert.

Bsp.:

CO₂ 378

100er Ziffer = $378/100 = 3$

10er Ziffer = $(378 - 3*100)/10 = 7$

1er Ziffer = $(378 - 3*100) - 7*10 = 8$

Jetzt habe ich die Zahlen der einzelne Ziffern und kann z.B. sagen, dass wenn die 100er Ziffer 3 ist, führe den Code aus der auf der Matrix an der 100er stelle eine 3 anzeigt aus.

So sieht das in der Arduino IDE aus:

```
int CO2 = 0 ; //CO2 Wert
int w1xx = 0; //Zahl Ziffer 1
int wx1x = 0; //Zahl Ziffer 2
int wxx1 = 0; //Zahl Ziffer 3

void CO2_Zerlegen (){
  w1xx = CO2/100;
  wx1x = (CO2-w1xx*100)/10;
  wxx1 = (CO2-w1xx*100)-wx1x*10;
}

void loop() {
  CO2_Zerlegen ();
  if (w1xx == 1) v1xx();
  if (w1xx == 2) v2xx();
  if (w1xx == 3) v3xx();
  if (w1xx == 4) v4xx();
  ...//abgekürzt ist zu lang
}
```

Als nächstes habe ich noch 3 Rahmen in grün, gelb und rot gezeichnet und integriert. Wenn der CO₂ Wert unter 600ppm ist, wird der Code für den grünen Rahmen ausgeführt, zwischen 600ppm und 1000ppm der Code für den gelben Rahmen und über 1000ppm der Code für den roten Rahmen.

Erster Testlauf

Im Dezember habe ich alle Komponenten miteinander verbunden und in meinem Zimmer platziert. Alles hat funktioniert wie geplant. Ich konnte den CO₂ Wert sowohl auf der LED-Matrix als auch auf dem ThinkSpeak Webserver ablesen. Nach einiger Zeit ohne Lüftung wurde der Grenzwert erreicht und die Matrix leuchtete rot.

Quellen

Warum ist ein CO₂ Warner sinnvoll: <https://www.umweltbundesamt.de/richtig-lueften-in-schulen#warum-ist-ein-regelmassiger-luftaustausch-in-klassenzimmern-grundsatzlich-wichtig-und-in-der-pandemie-umso-mehr>

Octopus Board: <https://www.bildung-forschung.digital/digitalezukunft/de/bildung/iot> ,
<https://www.umwelt-campus.de/iot-werkstatt/tutorials/schnellstart-octopus-anschliessen-und-einrichten>

CO₂ Sensor: [https://www.distrelec.de/de/CO₂-temperatur-und-feuchtigkeitssensor-scd30-seeed-studio-101020634/p/30149396](https://www.distrelec.de/de/CO2-temperatur-und-feuchtigkeitssensor-scd30-seeed-studio-101020634/p/30149396)

Arduino IDE: <https://www.arduino.cc/en/software>

Fast LED : <https://github.com/FastLED/FastLED>

LEDMatrixControl: https://www.youtube.com/watch?v=YjzLWnPI_hg&t=1s

Vielen Dank an Herr Kübler von der VSK-Technik Kübler für die Unterstützung.