EUROPÄISCHE UNION
Investition in unsere Zukunft
Europäischer Fonds
für regionale Entwicklung

Das Projekt „Naturbezogene Bildung für Kinder und Jugendliche“ des „Neustart Innenstadt“ wird durch Mittel des Landes Nordrhein-Westfalen und des Europäischen Fonds für regionale Entwicklung (EFRE) gefördert.

Ministerium für Umwelt, Landwirtschaft, Natur- und Verbraucherschutz des Landes Nordrhein-Westfalen

# Cod<ing>

## …für Bildung und Umwelt

NEUSTART INNENSTADT

Eine Zusammenarbeit der „Naturbezogenen Bildung für Kinder und Jugendliche“ und dem Falkentreff Herten

Falkentreff Herten

# Vorwort

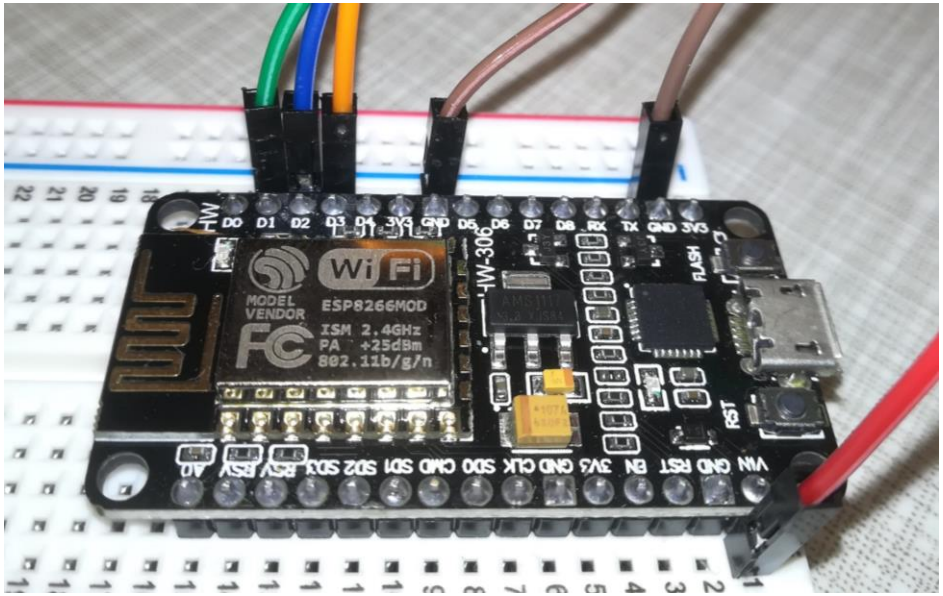
Dieses Handout wurde von der Teeniegruppe der Naturbezogenen Bildung für Kinder und Jugendliche sowie dem Falkentreff Herten erarbeitet. Wir wollen mit kurzen Bauanleitungen, Code- und Einrichtungsbeispielen Impulse für die Erkundung des Spannungsfeldes zwischen Digitalisierung und Umweltschutz setzen.

Beide Schlagwörter haben in den letzten Jahren an Brisanz gewonnen und ihre Auswirkungen werden unsere Gesellschaft vor größte Herausforderungen stellen. Dabei wird das Eine nicht ohne das Andere zu haben sein. Schon heute ist ein Großteil der Umweltanalysen automatisiert und ohne entsprechende IT nicht umsetzbar. In Zukunft werden MINT-Kompetenzen weiter an Bedeutung für die Entwicklung der Gesellschaft gewinnen. In Anbetracht dieser Entwicklungen möchten wir einen Beitrag zu einer nachhaltigen digitalen Transformation leisten. Wir wollen junge Menschen in ihrer Kreativität und Selbstwirksamkeit gegenüber ihrer Umwelt bestärken und gleichzeitig über die ersten Stolpersteine hinweghelfen.

# Das „Board“ ESP8266

Die ESP8266 NodeMCU ist ein Microcontroller, also ein Computer auf einer einzelnen Platine. Er findet vor allem beim Prototyping im Bereich des Internet der Dinge Verwendung. Mit Hilfe der einzelnen und beschrifteten Pins und passenden Jumper-Kabeln können besonders einfach und flexibel, da lötfrei, Schaltkreise erstellt werden. Für bessere Übersichtlichkeit kann auch eine einfache Steckplatine verwendet werden. Der Stückpreis des Controllers liegt bei 5–10€, wodurch Gruppenausstattungen in erreichbare Größenordnungen rücken.

Da der Microcontroller über ein Wifi-Modul verfügt, können zur Anzeige auch einfache http-Server betrieben werden, wodurch Kosten für Displaymodule etc. bei Bedarf eingespart werden. Ebenfalls ermöglicht das Wifi-Modul du Einbindung in bestehende Netzwerke, beispielsweise zur kontinuierlichen Datenübertragung an das eigene Smarthome, den Server der Schule oder diverser Citizen-Science-Projekte. Das Angebot kompatibler Bauteile wie LED‘s, Sensoren oder Motoren ist mannigfaltig. Jedoch bietet nicht jede Software Treiber für alle Module!



# Entwicklungsumgebungen ESP8266 und ESP32

## Ardublock Umwelt-Campus-Birkenfeld

Der Umwelt Campus Birkenfeld der Hochschule Trier hat eine Ardublock-Version veröffentlicht, welche für den Einsatz in den MINT-Bereichen optimiert wurde. Ursprünglich für das Board Octopus entwickelt, bietet sie auch in Verbindung mit einfachen (und erschwinglichen!) ESP-Boards eine übersichtliche Oberfläche, an welcher sich der Einstieg ins Thema sowie de Übergang von graphischem zu textbasiertem Coden anschaulich an aktuellen Fragestellungen vermitteln lässt. Die zum Zeitpunkt dieser Veröffentlichung aktuellste Version findet hier:
https://seafile.rlp.net/f/bb6bdbfe28034cd4861a/

Es lohnt in jedem Fall ein Blick auf dieses bestens betreute und aufgearbeitete Projekt an der Schnittstelle zwischen digitaler und analoger Welt!
https://www.umwelt-campus.de/iot-werkstatt/

## Arduino IDE

Falls ihr auf die graphische Oberfläche für den ESP8266 verzichten wollt und könnt, installiert direkt die zugrunde liegende Arduino IDE von der entsprechenden Homepage:
https://www.arduino.cc/en/software
Leider bietet oben genannte Ardublock-Version nur eingeschränkte Unterstützung bei dem Board ESP32. Hier können andere der unzähligen Ardublock-Versionen Abhilfe schaffen, welche zum jetzigen Zeitpunkt jedoch unbehandelt bleiben sollen.

**Voreinstellungen**
Stelle sicher, dass in der Arduino IDE diese Links
http://arduino.esp8266.com/stable/package_esp8266com_index.json,https://dl.espressif.com/dl/package_esp32_index.json
unter „Datei->Voreinstellungen->Zusätzliche Boardverwalter URL‘s“ eingefügt sind.

# Entwicklungsumgebungen ESP8266 und ESP32

## Fehlende library-Dateien für eigene Hardware

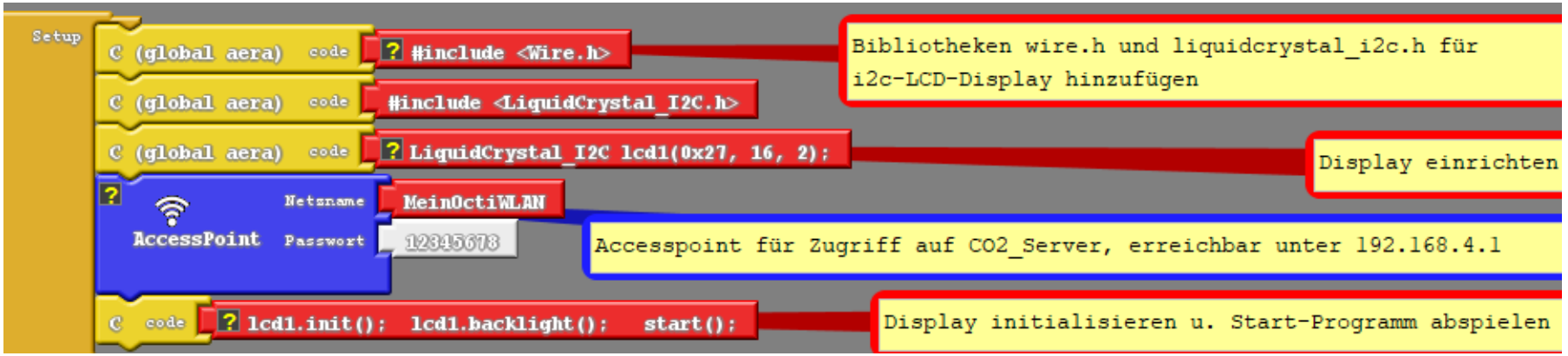
Bei manchen Anwendung gilt es, den ursprünglichen Umfang der Blöcke der Ardublock-Version zu erweitern. Da Ardublock auf der Arduino IDE aufsetzt, müssen entsprechende Bibliotheken zunächst auch über diese installiert werden. Ihr könnt den library-Namen  unter

Sketch->Bibliothek einbinden->Bibliotheken verwalten

den Namen im oberen Suchfeld eingeben und das gewünschte Paket installieren.

### Ansprechen der librarys über Ardublock

Um die installierten Bibliotheken zu verwenden, kann nun mit dem Baustein „c global area“ die Bibliothek angesprochen werden. Eigener Code kann an beliebiger Stellen über die Code-Bausteine eingefügt werden.



## Verwendung in Gruppenstärke/Klassenverband

Wer bis hierhin gelesen hat, fragt sich vielleicht, ob nun 30 Computer einzeln einer mehrstufigen Installation bedürfen. Diesen Schrecken kann glücklicherweise der Umstand abwenden, dass die komplette Software mobil vom USB-Stick betrieben werden kann.
Wer also einmal die Software installiert und möglicherweise erweitert hat, kann durch einfaches kopieren der Programmordner seine Version anderen zur Verfügung stellen, was die Verwendung in größeren Gruppen enorm erleichtert.

Für schnelle Versuche unterwegs oder auch das Homeschooling, bieten sich Online-Editoren an, wobei unsere Gruppe diesen bevorzugte:

http://iot-pilot.umwelt-campus.de:8000/ardublockly/index.html#

# Ein paar Tipps für willige Nachahmer…

## Einkauf

Achtet darauf, möglichst fertig gelötete Bauteile zu kaufen. Sowohl die unterschiedlichen Entwicklerboards als auch Sensoren, Stecker oder Kabel werden auch in Einzelteilen oder in unterschiedlichen Versionen angeboten. Kläre also im Vorhinein die genauen Spezifikationen der Bauteile.
Im stationären Handel kann der Einkauf eine Herausforderung für sich darstellen. Habt ihr das Glück in eurem Umfeld noch einen lokalen Elektronikfachhandel finden zu können, sind eure Anfragen sicher die beste Grundlage für eine wunderbare Freundschaft mit den Verkäufern.
Online sieht das ganze natürlich anders aus. Die Beschaffung aus Übersee-Versandhandeln ist oft die günstigere Variante, jedoch meist mit langen Lieferzeiten verbunden. Auch die oft fehlende ordentlich ausgewiesene Rechnung kann ein Hindernis für Institutionen sein. Die hier ansässige Elektronik-Versandhandel bieten dagegen oft Sonderkonditionen für Bildungsanstalten oder Institutionen. Hier lohnt sich die Recherche!

## Planung & Vorbereitung

Nichts ist frustrierender, als nach langer harter Denkarbeit festzustellen, dass diese nie zum gewünschten Ergebnis führen kann. Erspare dir oder anderen Mitstreitern diese

## Software

Hier gilt es die eigenen Anforderungen an die Entwicklungsumgebung im Blick zu behalten, um sich nicht in der Flut an Möglichkeiten zu verlieren. Weniger ist hier oft mehr um Übersichtlichkeit zu gewährleisten.
Für diese Arbeitsgruppe hat sich für die ESP-Entwicklerboards die Arduino IDE als verlässliches Werkzeug

# Materialliste für unseren IOT-Koffer

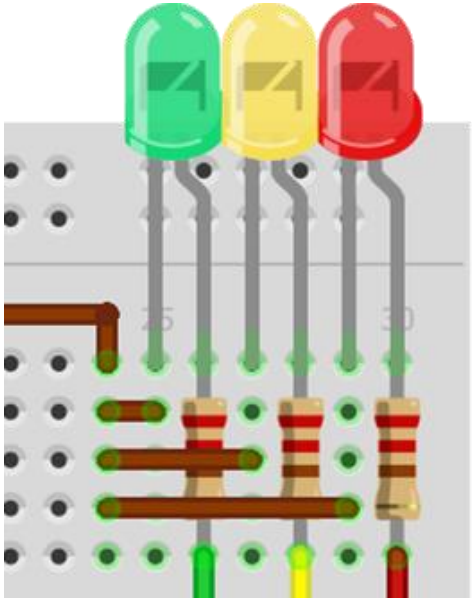
**//ich packe meinen Koffer…**

- Esp8266 NodeMCU
- ESP32 NodeMCU
- BMP680 Multi-Sensor
- MH-Z19B $CO_2$-Sensor
- DHT-22
- LED grün
- LED gelb
- LED rot
- i2C-LCD-Display (16x2)
- diverse Jumper-Kabel
- Vorwiderstände
- Steckplatine

Diese komplette Ausstattung kostet ca. 100-150€ und enthält eine Vielzahl von Anwendungsmöglichkeiten im Kontext der BNE aber auch IoT, im Technikunterricht oder diversen Hobbyräumen. Sie bildet somit die ideale Ausgangslage zum Experimentieren und erstem Erstellen eigener Codes. Und das alles Open-Source und für Fortgeschrittene erweiterbar!
Wer sich erst einmal ausprobieren möchte, kann mit dem ESP8266, ein paar einzelner LED‘s und Kabel für ca. 15-25€ den Einstieg wagen. Der eigene kleine Server, Wifi-Anwendungen wie PAX-Counter oder das bloße erstellen eines Lauflichts lassen sich auch ohne weitere Sensoren entwerfen, welche hier den größten Kostenfaktor darstellen. Wer nicht alle Sensoren auf einmal anschaffen möchte, sollte aufgrund der Ausstatttung einen Einstieg mit dem BMP680 in Erwägung ziehen.

# Vorwiderstände für LED‘s berechnen

LED‘s eignen sich besonders für die ersten Schritte mit dem Microcontroller. Je nach Auswahl deiner LED benötigen diese einen Vorwiderstand, um die ausgegebene Spannung entsprechend den Anforderungen der LED‘s umzuwandeln.



$$(U_{Gesamt} - U_{LED}) \quad / \quad I_{LED} \quad = \quad R$$

In der Regel gilt für rote LED‘s:

$I_{LED}$=0,02A $U_{LED}$=1,6V

Für die 3,3V des ESP8266 gilt also:

$$(3{,}3V - 1{,}6V) \; / \; 0{,}02A = 85\Omega$$

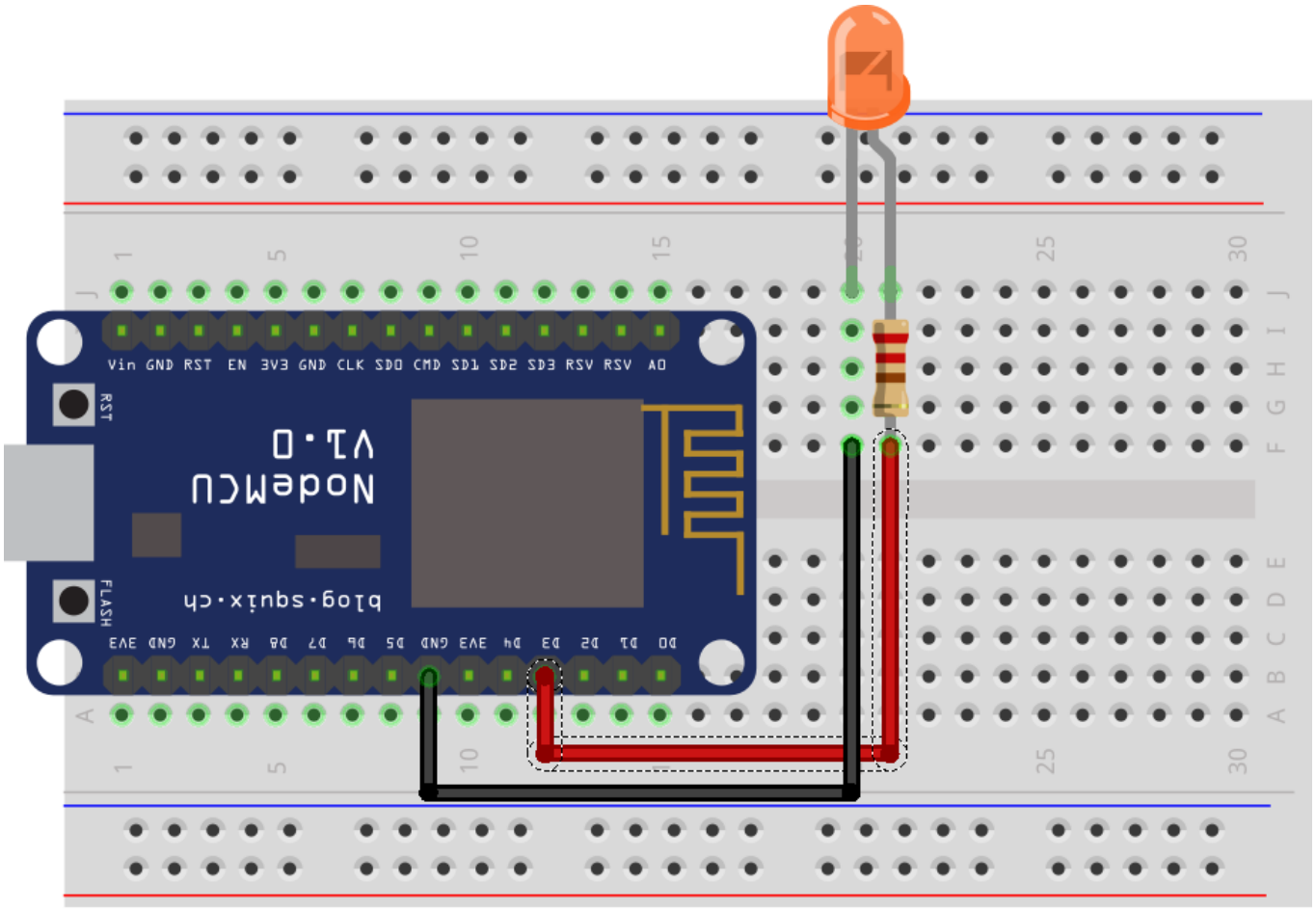
Wir nehmen den nächstgrößten Widerstand: 100Ω

# PAX-Counter_mit_Warn-LED

**//Material**

- Esp8266 NodeMCU
- 1 x LED gelb
- Jumper-Kabel
- Vorwiderstände
- Steckplatine

**E**



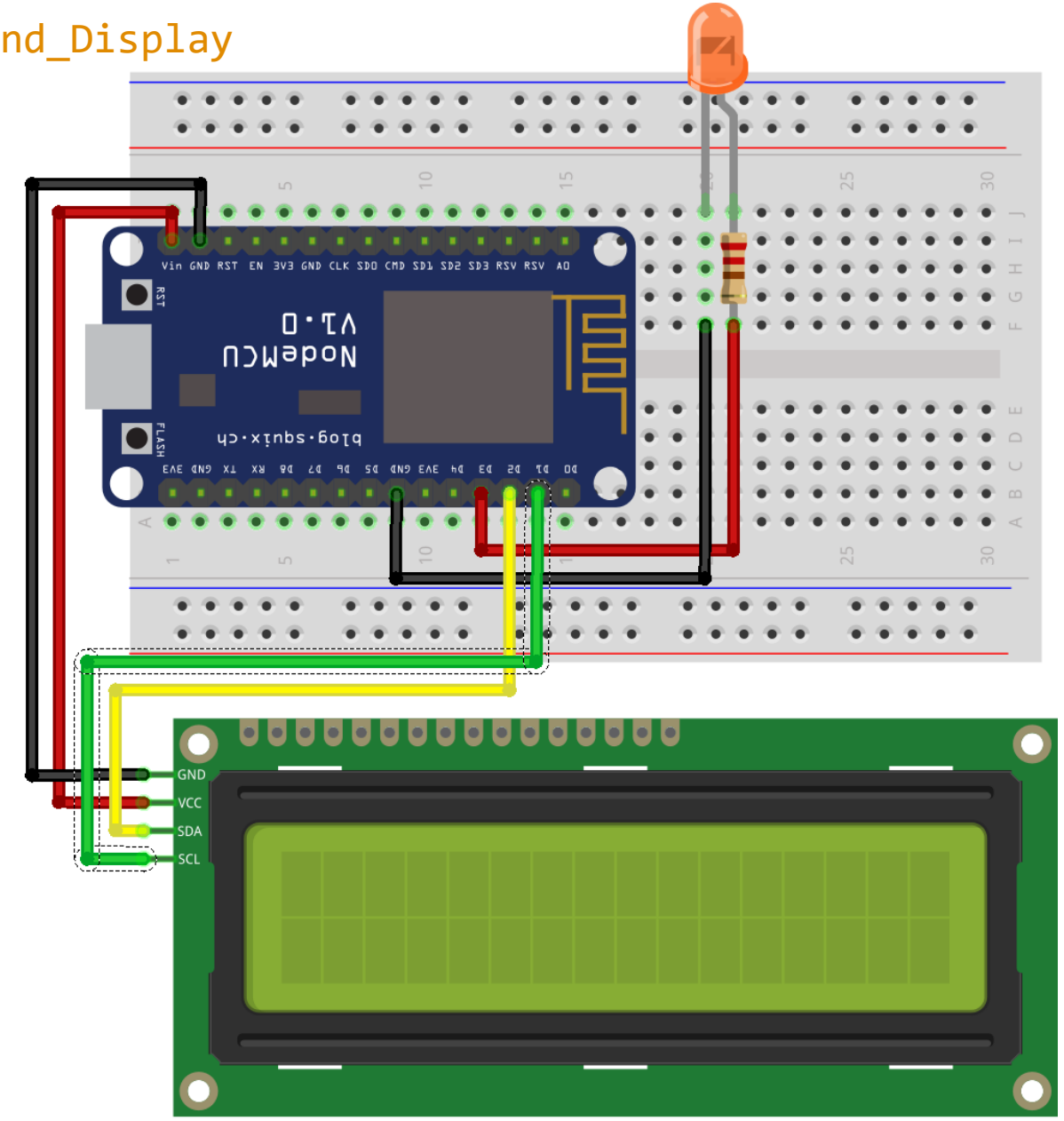
# PAX-Counter_mit_Warn-LED_und_Display

**//Material**

- Esp8266 NodeMCU
- 1 x i2C-LCD-Display (16x2)
- 1 x LED gelb
- Jumper-Kabel
- Vorwiderstände
- Steckplatine

**//Einkauf**

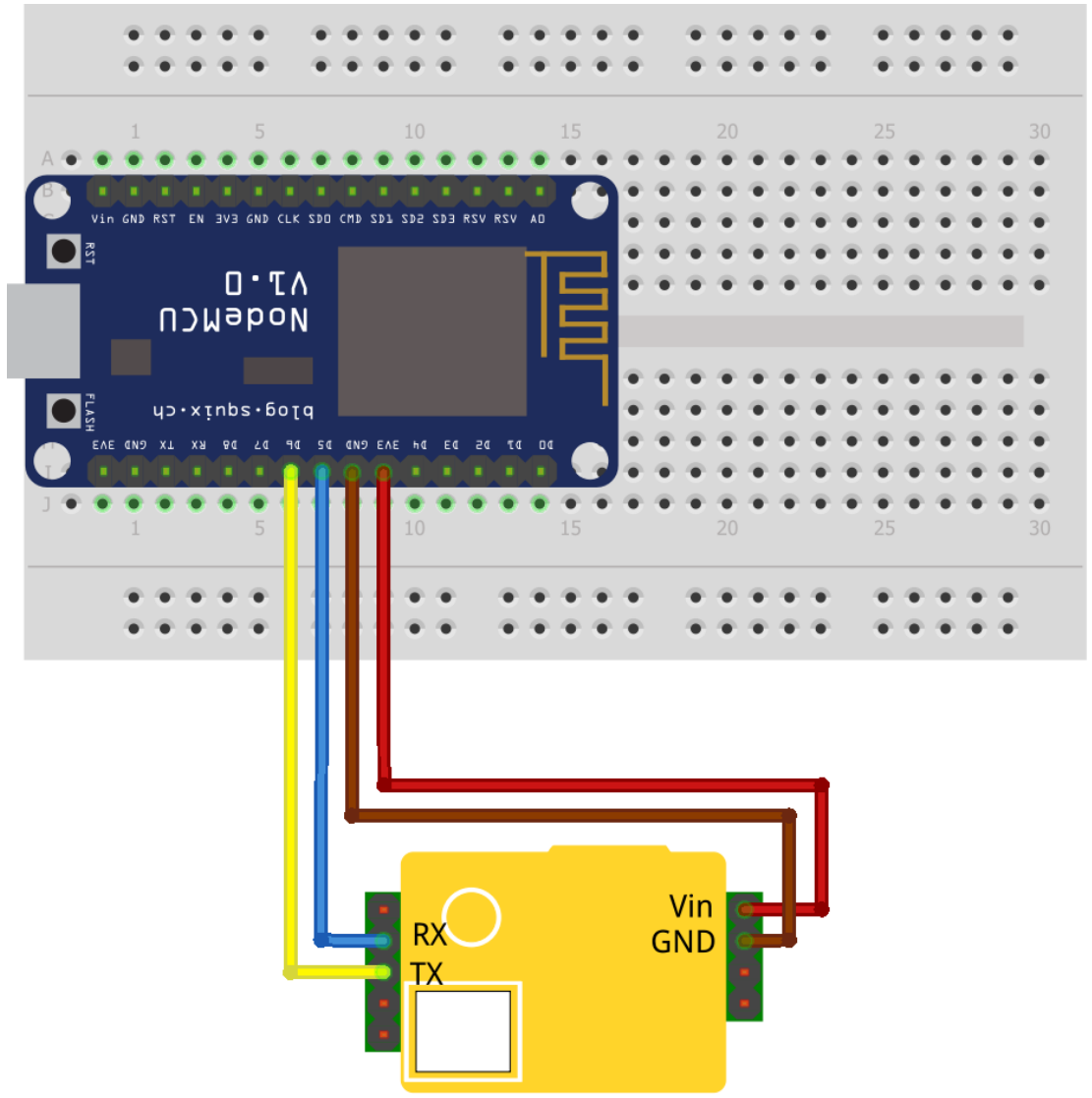
Achtet darauf, möglichst fertig gelötete Bauteile zu kaufen. Das reduziert den Aufwand in der Vorbereitung.



# $CO_2$-Server

**//Material**

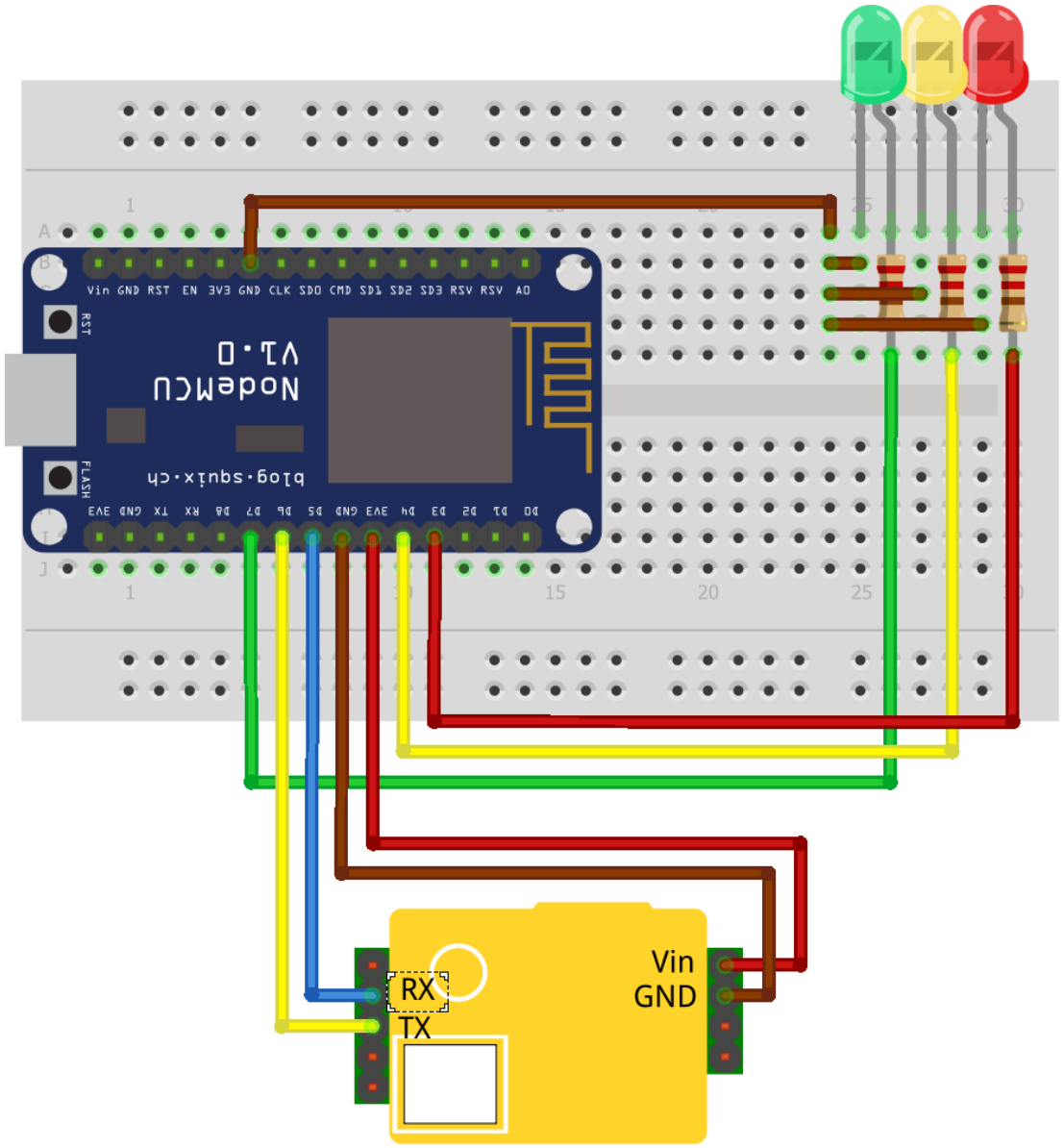
- Esp8266 NodeMCU
- CO2 Sensor MH-Z19B
- 4 Jumper-Kabel
- Steckplatine



# $CO_2$-Ampel mit Server

**//Material**

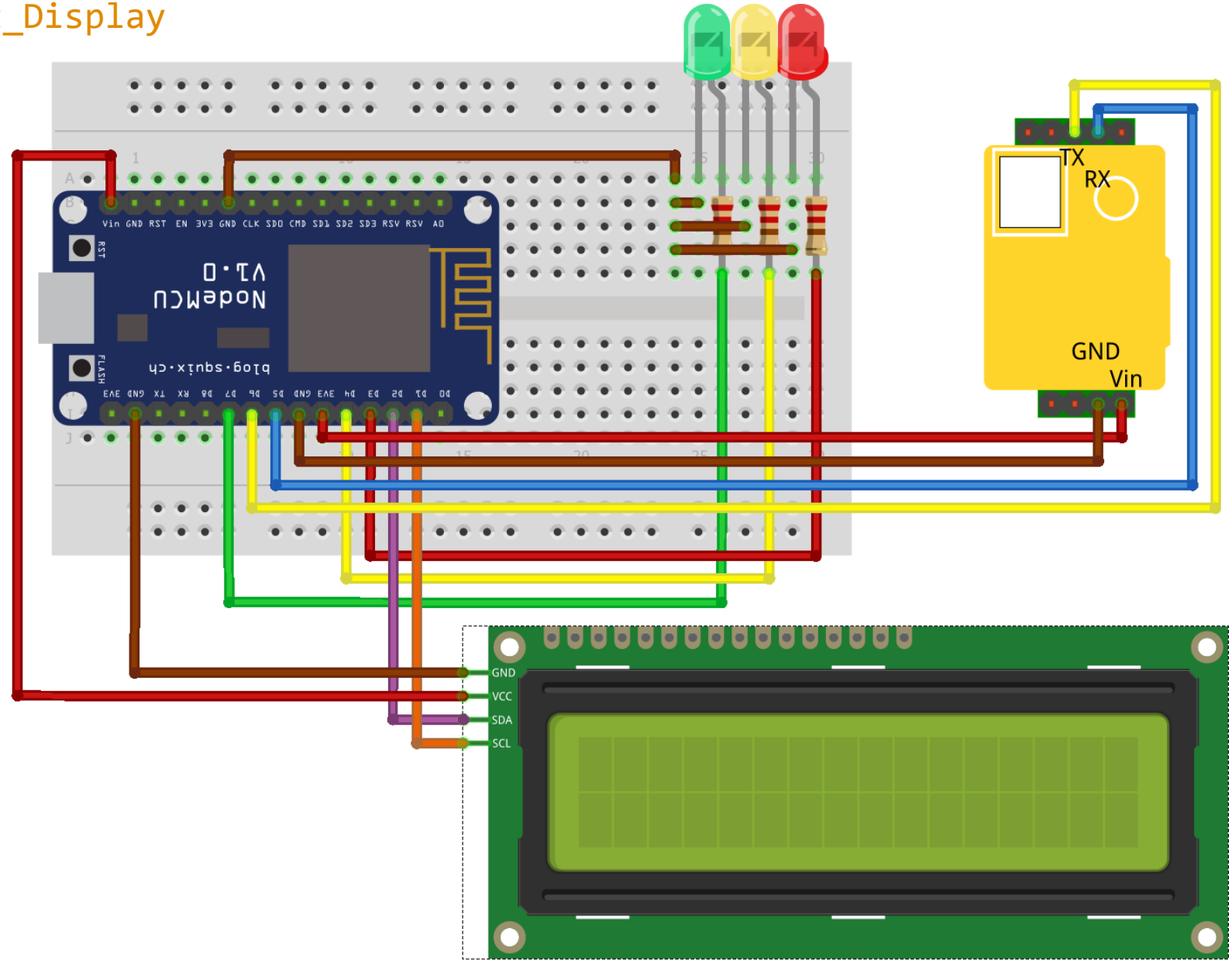
- Esp8266 NodeMCU
- MH-Z19B
- 1 x LED grün
- 1 x LED gelb
- 1 x LED rot
- diverse Jumper-Kabel
- Vorwiderstände
- Steckplatine



# CO2-Ampel_mit_Display

**//Material**

- Esp8266 NodeMCU
- MH-Z19B
- LED grün
- LED gelb
- LED rot
- i2C-LCD-Display (16x2)
- diverse Jumper-Kabel
- Vorwiderstände
- Steckplatine



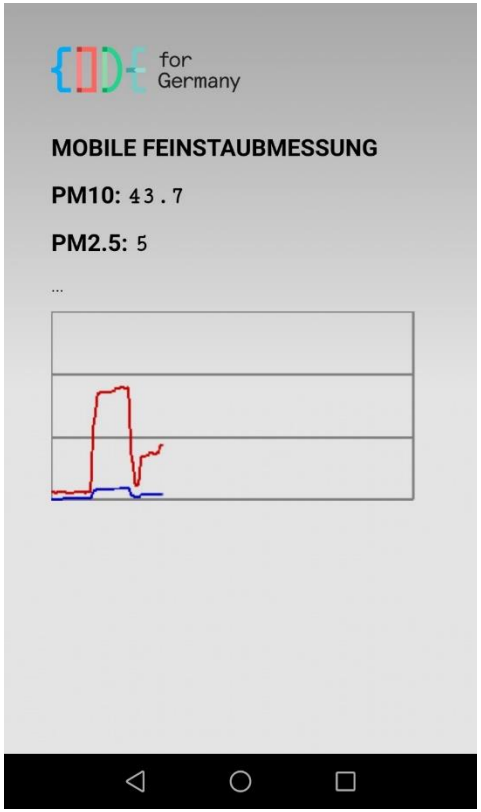
# Mobile Feinstaubmessung mit dem Smartphone

Neben Anwendungen wie Blink oder Thingspeak sind auch direkte Verbin-Dungen von Sensoren mit Smartphones möglich. Hier wurde beispielsweise ein mobiles Feinstaubmessgerät für Exkursionen oder Stadtrundgänge der Gruppe betrieben.

## //Material

- Android Smartphone/Tablet
- APK mobile Feinstaubmessung
- APK SmogCop
- SDS011 Feinstaubsensor
- OTA-USB-Adapter
- USB Typ A auf Buchse 5 Pin Raster 2,0 mm

(i.d.R. liegt dieser Adapter dem Sensor bei. Beim Einkauf also darauf achten!)




# Codebeispiele

Unsere Codebeispiele findet ihr unter

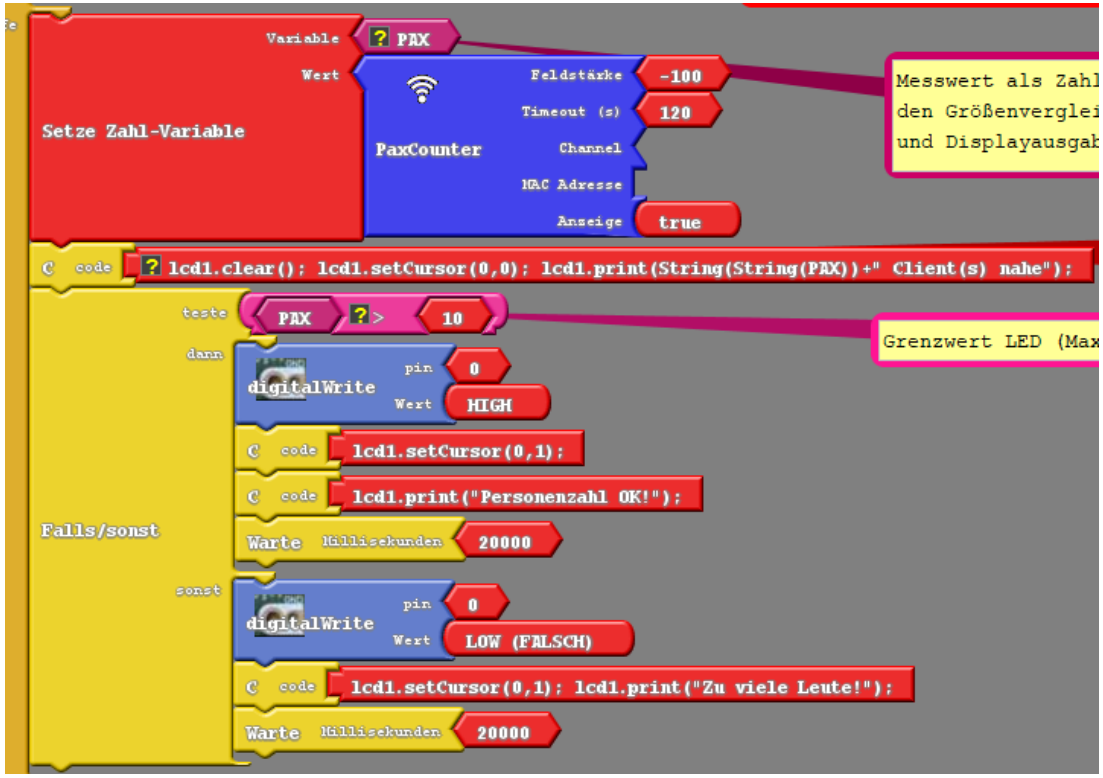
https://www.falkentreff-re.de/codebeispiele/

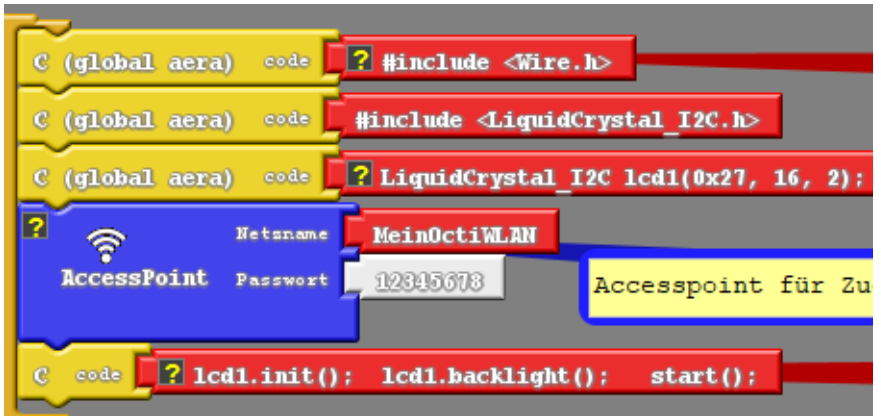
# Weitere Projekte

Der Umwelt-Campus Trier, welcher die Software entwarf, gibt ebenfalls manigfaltige Beispiele für weitere Anwendungen und Bauformen. Ebenso verhält es sich mit der Internetpräsenz des Projektbeteiligten Guido Burger.

# PAX-Counter_mit_Warn-LED_und_Display im Ardublock

**//void setup**



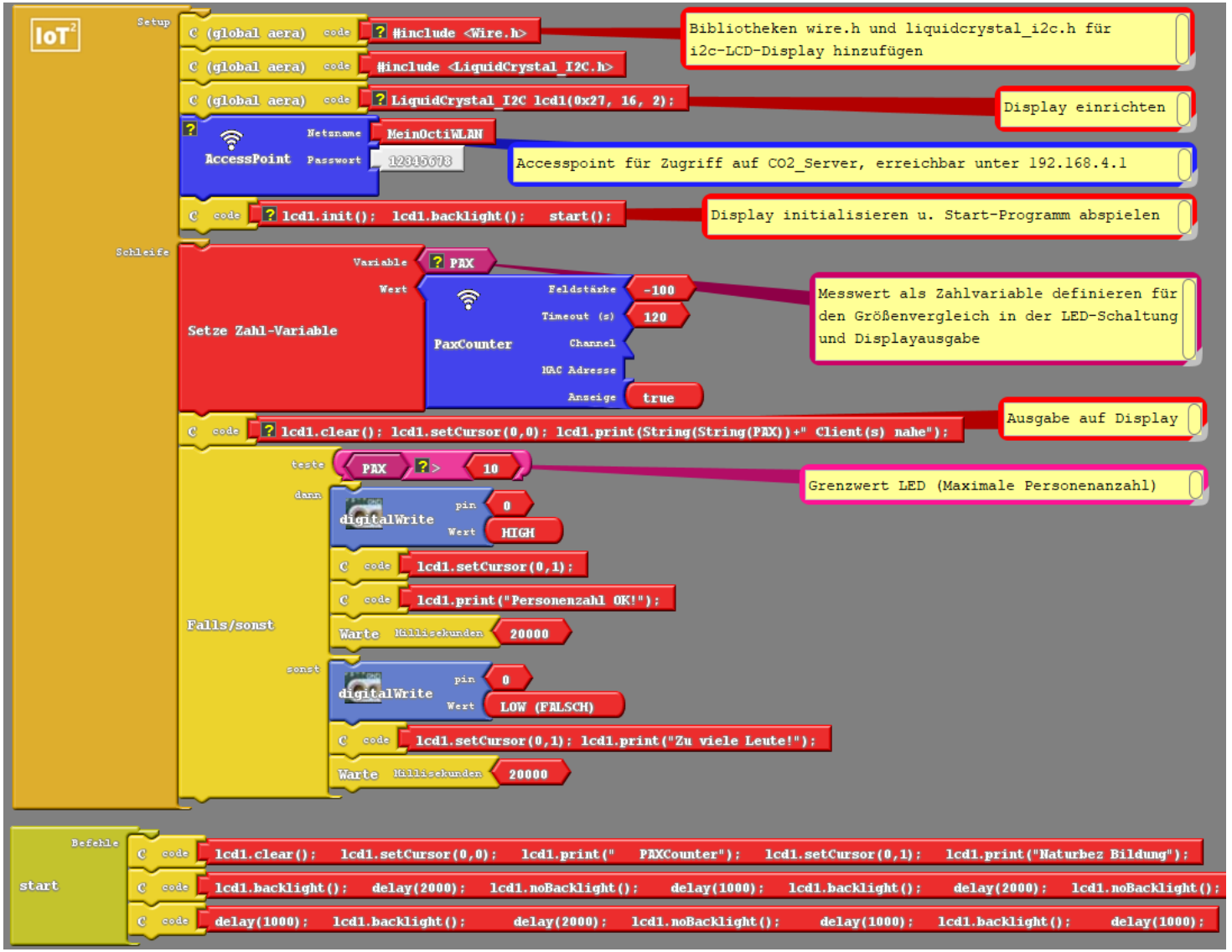




**//void loop**

**//void start**

# PAX-Counter_mit_Warn-LED_und_Display//Ardublock

**//Ardublock-Aufbau komplett**



# Code PAX-Counter_mit_Warn-LED_und_Display

## //der Code in der Arduino IDE

```
#include <ESP8266WiFi.h>
#include <WiFi_Sniffer.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>


int PAX = 0 ;
unsigned int Sniff_channel = 1;
//------------ WiFi-Sniffer,  This software is based on the work of Andreas Spiess, https://github.com/SensorsIot/Wi-Fi-
Sniffer-as-a-Human-detector//                              and Ray Burnette: https://www.hackster.io/rayburne/esp8266-mini-
sniff-f6b93a
int WiFiPaxCounter(int MinRSSI,  int timeout, int8 mychannel,String myMAC,int mydisplay) {
  int mycount=0;
  int randMAC=0;
  int ChanMin = 1, ChanMax =13; // europe channel 1-13, Japan 1-14
  if (mychannel > 0) {
    ChanMax = mychannel;
    ChanMin = mychannel;
  };
  if (mychannel < 0) {
    randMAC = mychannel;
  };
  wifi_set_promiscuous_rx_cb(promisc_cb);   // Set up promiscuous callback
  Sniff_channel = ChanMin;
  wifi_set_channel(Sniff_channel);
  wifi_promiscuous_enable(true);
  for (Sniff_channel = ChanMin; Sniff_channel <= ChanMax; Sniff_channel++) {
    wifi_set_channel(Sniff_channel);
    delay(300);             // 300 ms per channel
  }
  wifi_promiscuous_enable(false);
  mycount = SnifferCountDevices(MinRSSI,timeout,myMAC,randMAC,mydisplay); // Anzeige/zaehlen der Clients
  return mycount;
}
```

```
  LiquidCrystal_I2C lcd1(0x27, 16, 2);

void setup(){ // Einmalige Initialisierung
  Serial.begin(115200);
  WiFi.mode(WIFI_STA); // Pax-counter
  lcd1.init();            // LCD-Initialisierung
  lcd1.backlight();
  pinMode(0 ,OUTPUT); // LED an PIN 0
  start();
}

void loop() { // Kontinuierliche Wiederholung
  PAX = WiFiPaxCounter(-100,120,0,"all mac",true) ; //Hier feldstärke,timeout,channel,MAC-Adressen,Anzeige auswählen
  lcd1.clear();                         //Ausgabe der Variable PAX auf Display
  lcd1.setCursor(0,0);
 lcd1.print(String(String(PAX))+"               ");
 lcd1.setCursor(0,1);
 lcd1.print("Client(s) nahe");

 if (( ( PAX ) > ( 10 ) ))             //hier Anzahl Geräte für die Warn-LED eintragen
  {
    digitalWrite( 0 , HIGH );
  }
  else
  {
    digitalWrite( 0 , LOW );
  }
}

void start(){ //Start-Anzeige auf dem Display
  lcd1.clear();
  lcd1.setCursor(0,0);
  lcd1.print("   PAX-Counter");
  lcd1.setCursor(0,1);
  lcd1.print("Naturbez Bildung");
  lcd1.backlight();
  delay(2000);
  lcd1.noBacklight();
```

```
  delay(1000);
  lcd1.backlight();

  delay(2000);
  lcd1.noBacklight();

  delay(1000);
  lcd1.backlight();

  delay(2000);
  lcd1.noBacklight();

  delay(1000);
  lcd1.backlight();

  delay(1000);
}
```

EUROPÄISCHE UNION
Investition in unsere Zukunft
Europäischer Fonds
für regionale Entwicklung

Das Projekt „Naturbezogene Bildung für Kinder und Jugendliche“ des „Neustart Innenstadt“ wird durch Mittel des Landes Nordrhein-Westfalen und des Europäischen Fonds für regionale Entwicklung (EFRE) gefördert.

Ministerium für Umwelt, Landwirtschaft,
Natur- und Verbraucherschutz
des Landes Nordrhein-Westfalen

Wir wünschen viel Spaß beim Tüfteln und Entdecken!