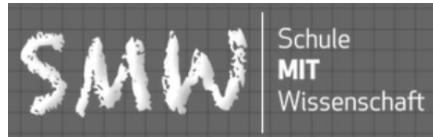



WS 05 | Vindriktning – IKEA-Hacking in der Schule

Luftqualitätssensor goes IoT

The screenshot shows the website for the Regensburger Schülerlabor RSL. At the top, there is a search bar and navigation links for 'KONTAKT', 'IMPRESSUM', and 'DATENSCHUTZ'. The main content area features a large logo for the 'Schülerlabor/Schülerforschungszentrum der Fakultät für Physik' and a descriptive paragraph. A sidebar on the left lists various categories like 'Aktuelles', 'Vergangene Veranstaltungen', and 'Material'. A sidebar on the right provides contact information for Dr. Stephan Giglberger.

REGENSBURGER SCHÜLERLABOR RSL

Das Schülerlabor/Schülerforschungszentrum der Fakultät für Physik

Das Regensburger Schülerlabor / Schülerforschungszentrum sieht sich als Schnittstelle zwischen Schule und Universität. Wir bieten interessierten Schülerinnen und Schülern – aber auch Lehrerinnen und Lehrern – die Möglichkeit, die Kompetenzen der Fakultät für Physik für sich zu nutzen.

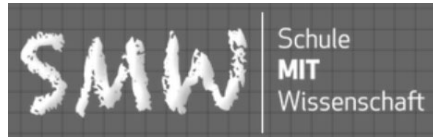
Blättern Sie durch unsere Seiten – sprechen Sie uns an!

Kontakt
 Dr. Stephan Giglberger
 stephan.giglberger@ur.de
 +49 (0)941 943 2088
 what3words:
 schwärme.töpfe.regler

Regensburger Schülerlabor/SFZ
 go.ur.de/RSL
 in Kooperation mit Peter Dirnhofner

WS 05 | Vindriktning – IKEA-Hacking in der Schule

Luftqualitätssensor goes IoT



„Du möchtest mehr über die Luft erfahren, die du in deinem Zuhause atmest? Du kannst die Luftqualität durch Partikelanalyse (PM2,5) überprüfen – mit diesem eleganten und einfach zu bedienenden Sensor passend zum FÖRNUFTIG Luftreiniger.“

The screenshot shows the IKEA website interface for the VINDRIKTNING air quality sensor. The product is priced at 12.99€ and is available in Regensburg. The page includes a search bar, navigation icons, and a delivery notice: "Lieferung innerhalb von 1-3 Tagen ab Vertragsschluss, gilt an Werk- und Samstagen und für Pakettlieferungen bis 30 kg." The product description states: "VINDRIKTNING Luftqualitätssensor, Preis inkl. MwSt.*, 4.99€ (4 reviews). Financing is available from 150€. A USB-C cable and USB power adapter are sold separately. A recommendation for the LILLHULT USB-A to USB-C adapter (4.99€) is shown. Delivery and Click & Collect options are available in Regensburg. The 'In den Warenkorb' button is highlighted in blue.

Definition Feinstaub

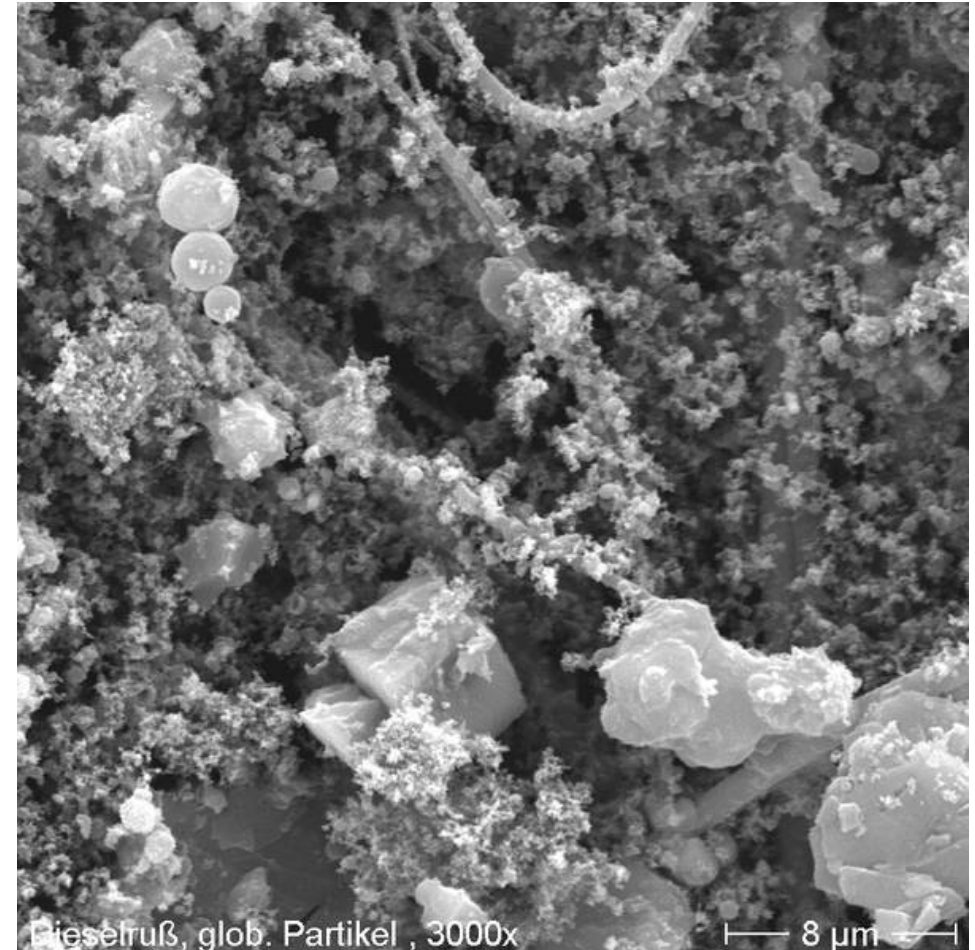
Staub: komplexes, heterogenes Gemisch aus festen oder flüssigen Teilchen

TSP (Total Suspended Particles): Masse des Gesamtstaubs (Schwebestaub), von Luft getragen

PM₁₀: mind. 50% der Teilchen haben einen Durchmesser von 10 µm oder kleiner

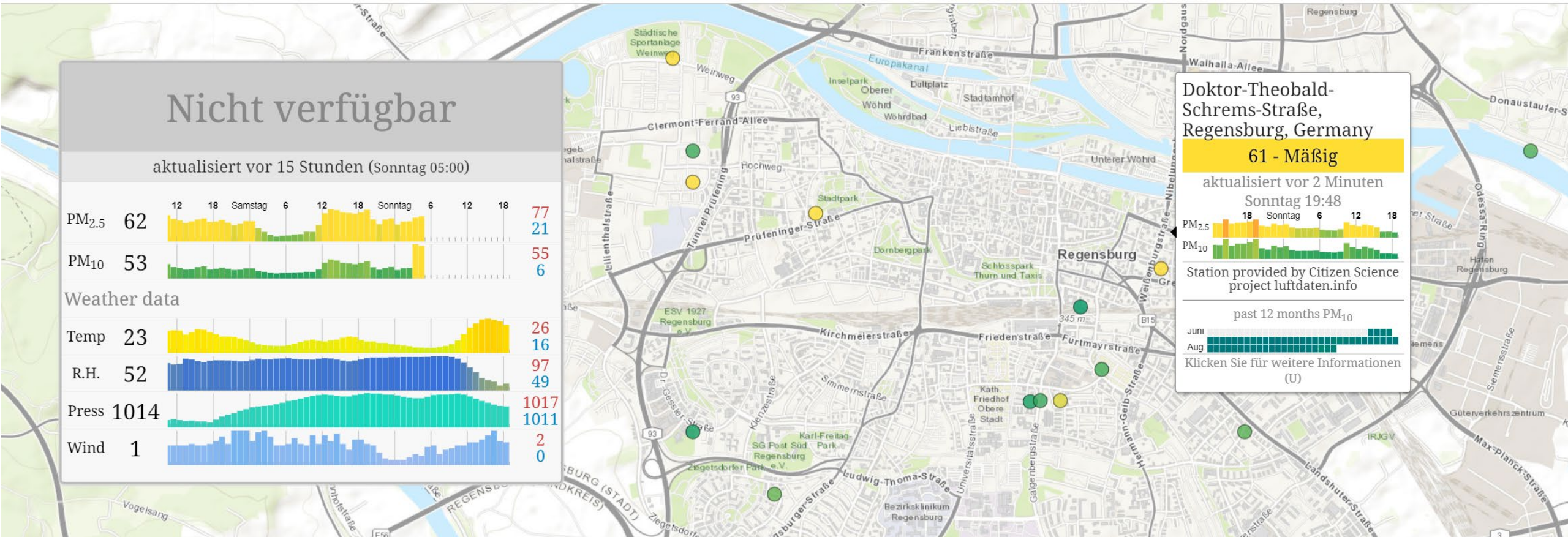
PM₅: mind. 50% der Teilchen haben einen Durchmesser von 5 µm oder kleiner

PM_{2,5}: mind. 50% der Teilchen haben einen Durchmesser von 2,5 µm oder kleiner



Messdaten (öffentlich zugänglich, starke regionale Schwankung)

Luftverschmutzung in Brennbergstraße, Regensburg, Germany, Regensburg, Germany_DE Echtzeit-Luftqualitätsindex (LQI)

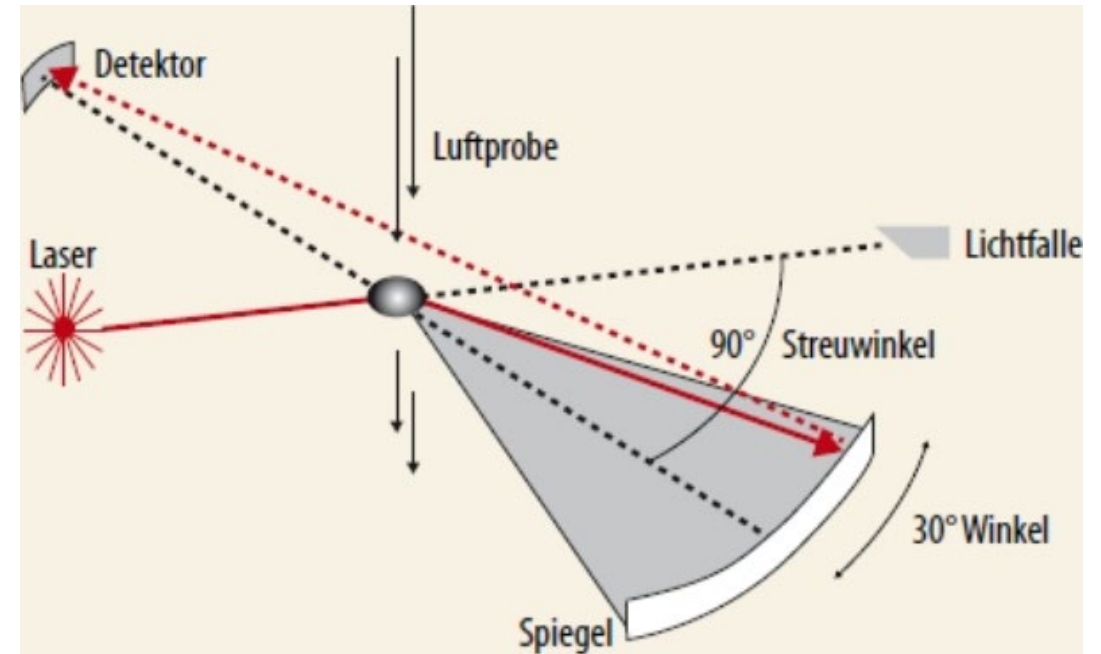


Ein wenig Physik...

Messprinzip: Lichtstreuung

- Streuung des Laserlichts an Feinstaubpartikel
- Streuwinkel anhängig von Partikelgröße
- Auflösung: $0.3 \mu\text{g}/\text{m}^3$

Typ. Modul: Nova Fitness SDS018



IKEA-Hacking

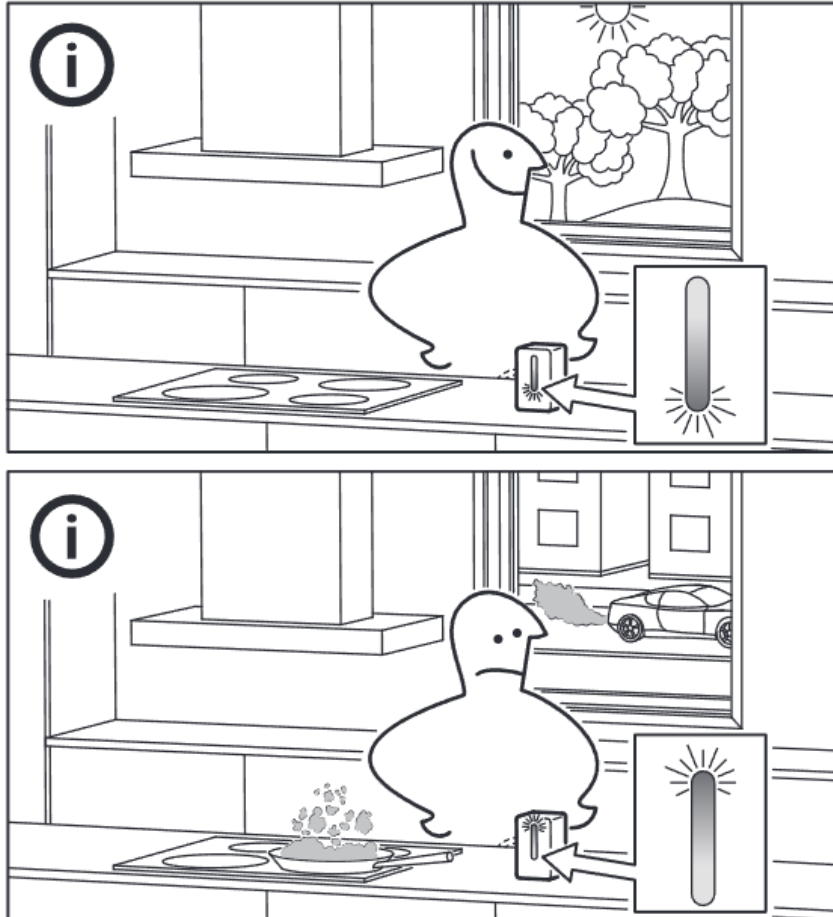
Definition

„Individuelle Umgestaltung von Bausatzmöbeln der schwedischen Möbelhauskette IKEA“



IKEA-Hacking: Vintriktning

Feinstaubsensor



Deutsch

9

Merkmale des Luftqualitätsfühlers

- USB-C-Netzteil (5.0 V, 2.0 A) nicht im Lieferumfang enthalten
- Messung von PM2,5-Partikeln
- LED-Statusanzeige

Gebrauchsanweisung

- Schließe den Netzadapter an den USB-C-Anschluss an.
- Der Luftqualitätsfühler startet eine ca. 10 Sekunden lange Anlaufsequenz.
- Wenn die LEDs nicht mehr pulsieren, ist das Gerät bereit.

LED-Anzeigen für die Luftqualität (PM 2.5):

Grün: 0-35 /Gut + Niedrig

Gelb: 36-85/OK + Mittel

Rot: 86- /Nicht gut + Hoch

Pulsierend: Startmodus

Gut zu wissen

- Betriebstemperatur: 0 °C bis 40 °C.
- Betriebsfeuchtigkeit: 0 bis 95 % RH (Empfohlene Arbeitsfeuchtigkeit: 40-60 % RH)
- Trenne das Produkt vor der Reinigung und bei Nichtgebrauch von der Stromquelle.

Diese Anleitung gut aufbewahren.

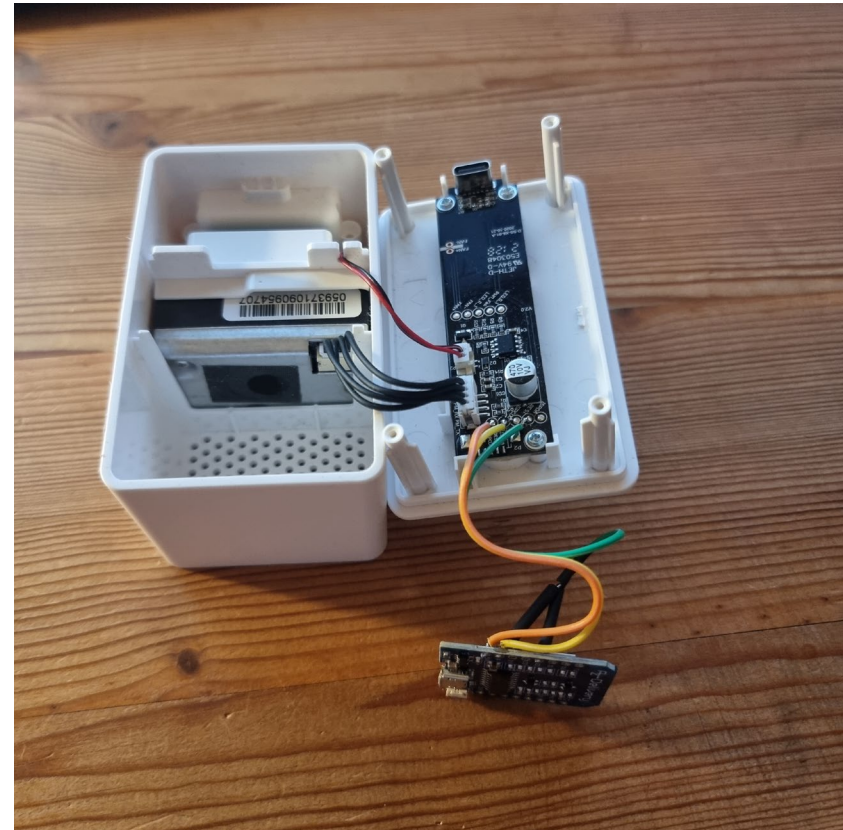
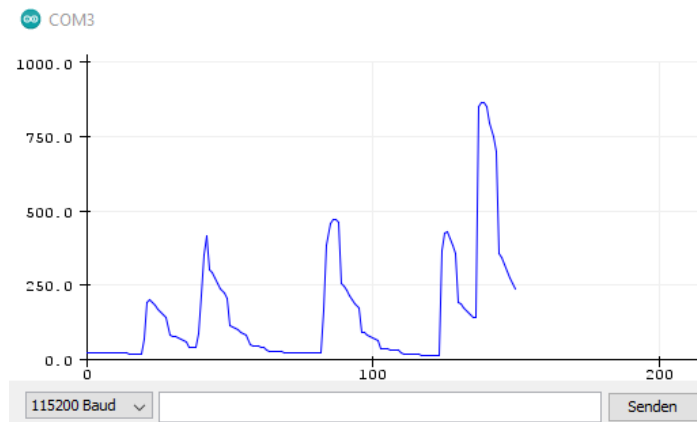
IKEA-Hacking: Vintriktning

„Diebstahl“ des Sensorsignals

Ableiten des Original Sensor-Signals in einen ESP32, ohne die Funktion des Vintriktning zu verändern

Der ESP32 gibt die Daten per Bluetooth oder WLAN weiter

Auswertung des Signals mit Arduino Serial Plotter, Excel etc. – oder direkte Einbindung in ein Smart Home System

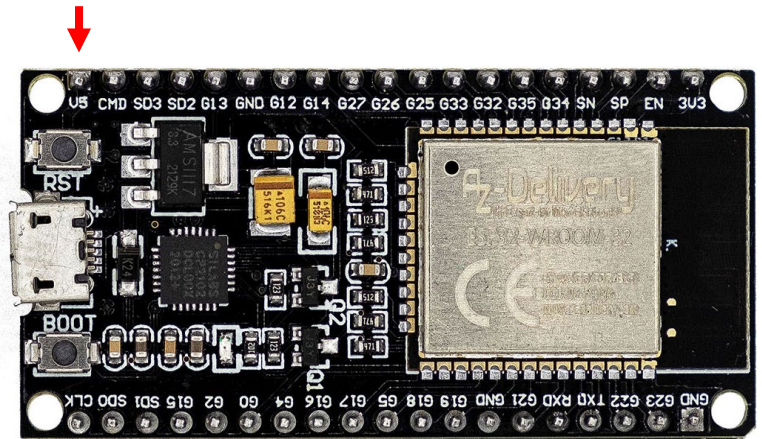


Mikrocontroller ESP32

„Diebstahl“ des Sensorsignals

Nur drei Kabel nötig: „+“, „GND“ und das Signal

5V (rot)



Signal (z.B. orange)

GND (schwarz)

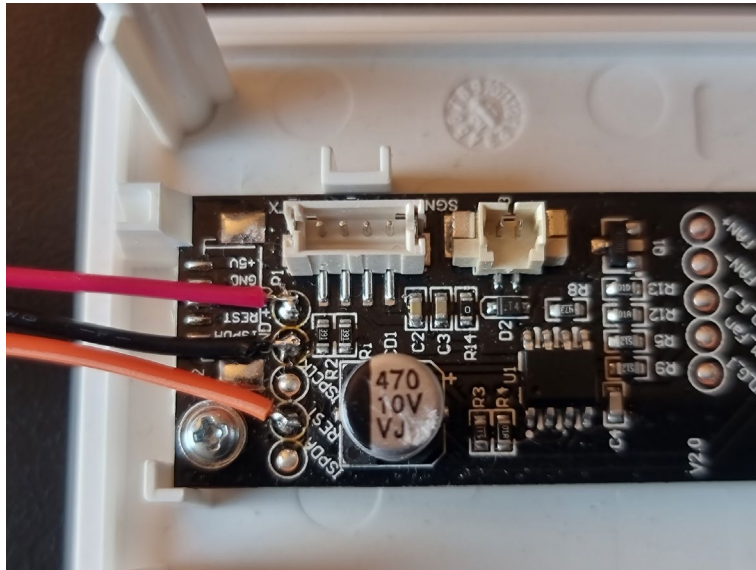


Rückseite mit individuellem BT-Namen

Vorbereitung des ESP32

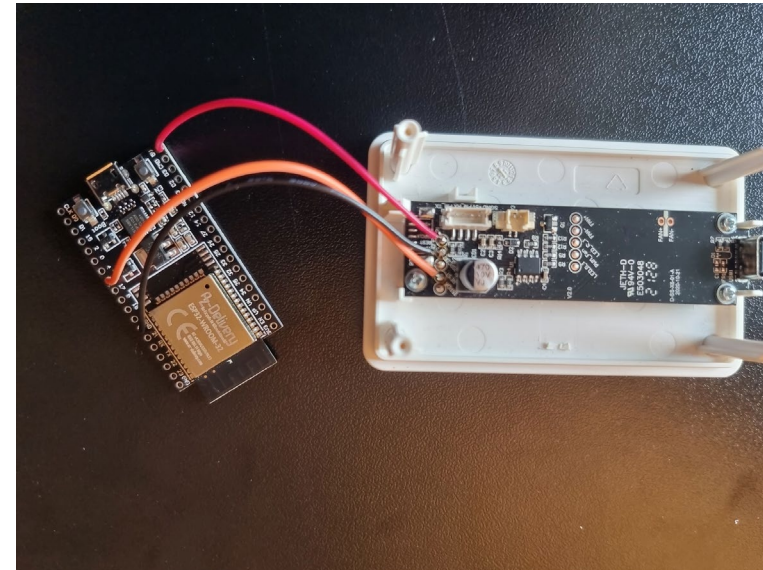
„Diebstahl“ des Sensorsignals

Kabel am Vindriktning anlöten



rot → „+5V“; schwarz → „GND“; orange → „REST“

Kabel am ESP32 anlöten



rot → „+5V“; schwarz → „GND“; orange → „G16“

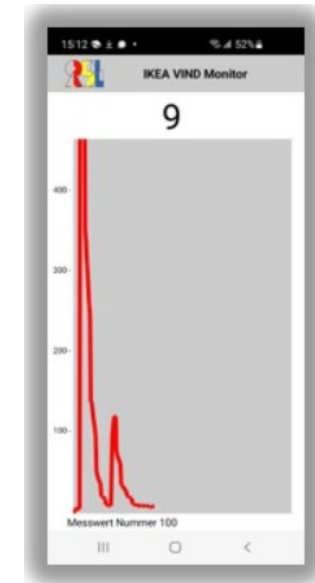
Test // Aufspielen der Software auf Smartphone

Smartphone vorbereiten

- Bluetooth einschalten
- Modul „IKEA_BT_0xx“ suchen
- Installation am Smartphone zulassen:
Einstellungen → Sicherheit →
Unbekannte Apps installieren
- QR-Code scannen und Datei
IKEA_VIND_Monitor.apk
installieren
- passendes Bluetooth-Modul koppeln



https://github.com/PeterDirnhofer/IKEA-vindriktning-ESP32-Bluetooth/raw/main/IKEA_VIND_Monitor.apk



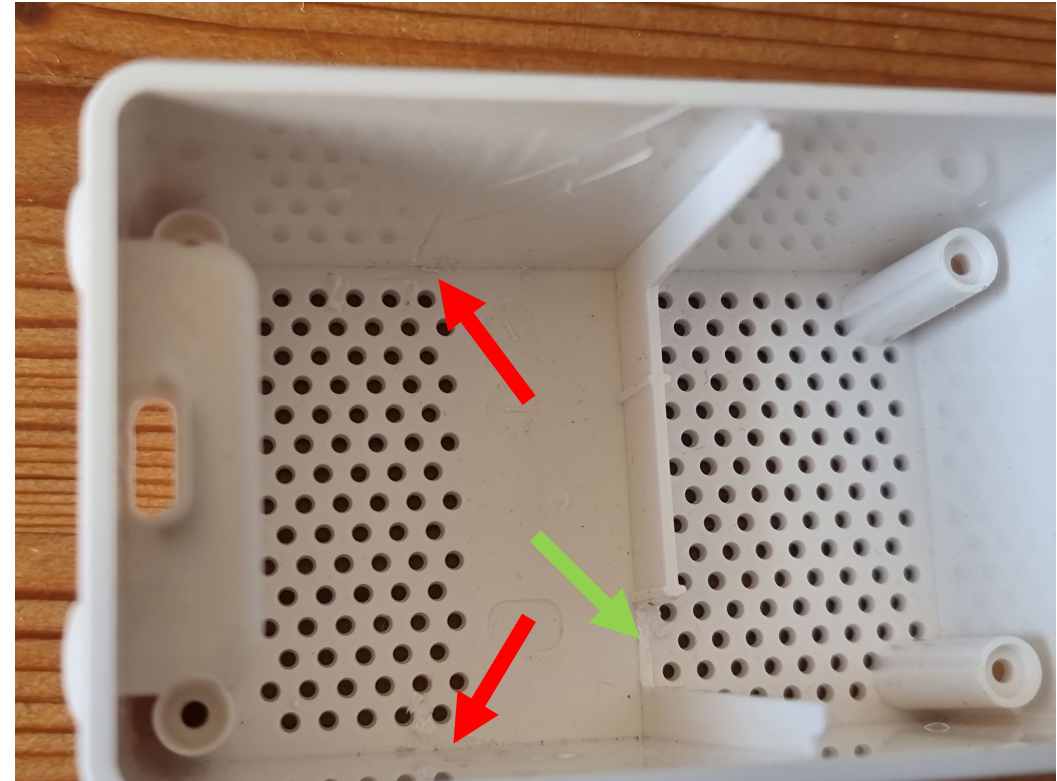
Vorbereitung Gehäuse

Platz schaffen

Damit der ESP später ins Gehäuse passt:

1) beide Stege, die das Sensormodul/ den Lüfter seitlich halten, vollständig bis ganz unten entfernen

2) einen Teil der Abtrennung wie im Bild herausbrechen

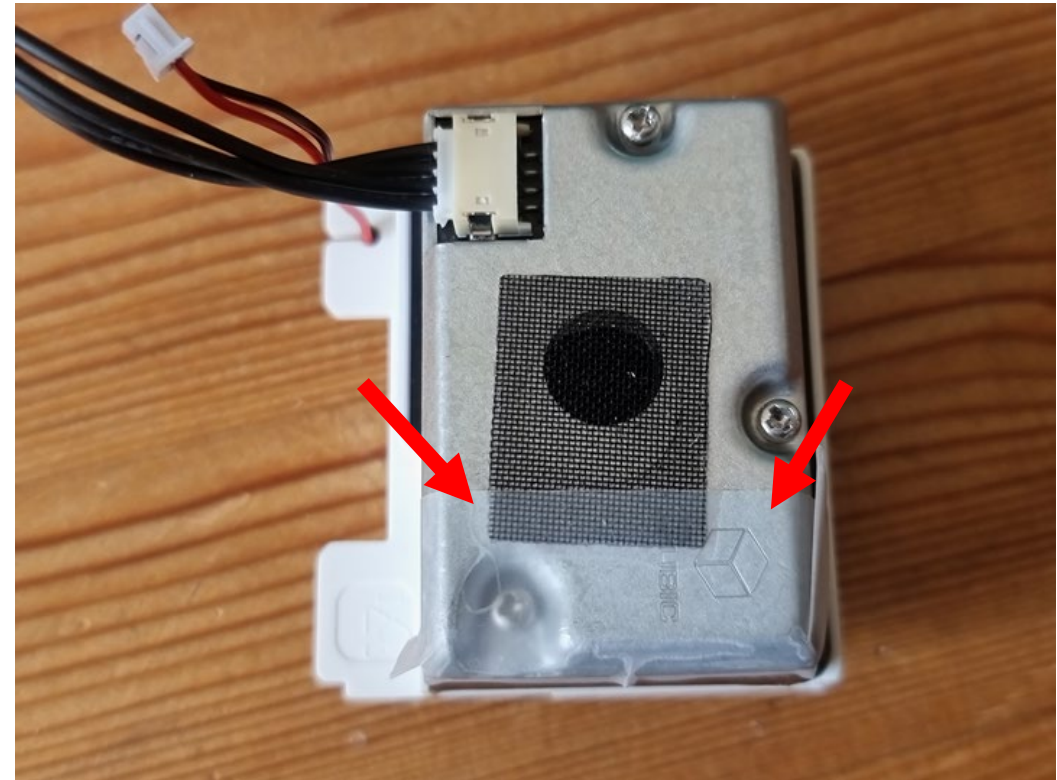


Vorbereitung Gehäuse

Isolieren

Um einen möglichen späteren Kurzschluss zu vermeiden:

unteren Teil des Sensorgehäuses abkleben

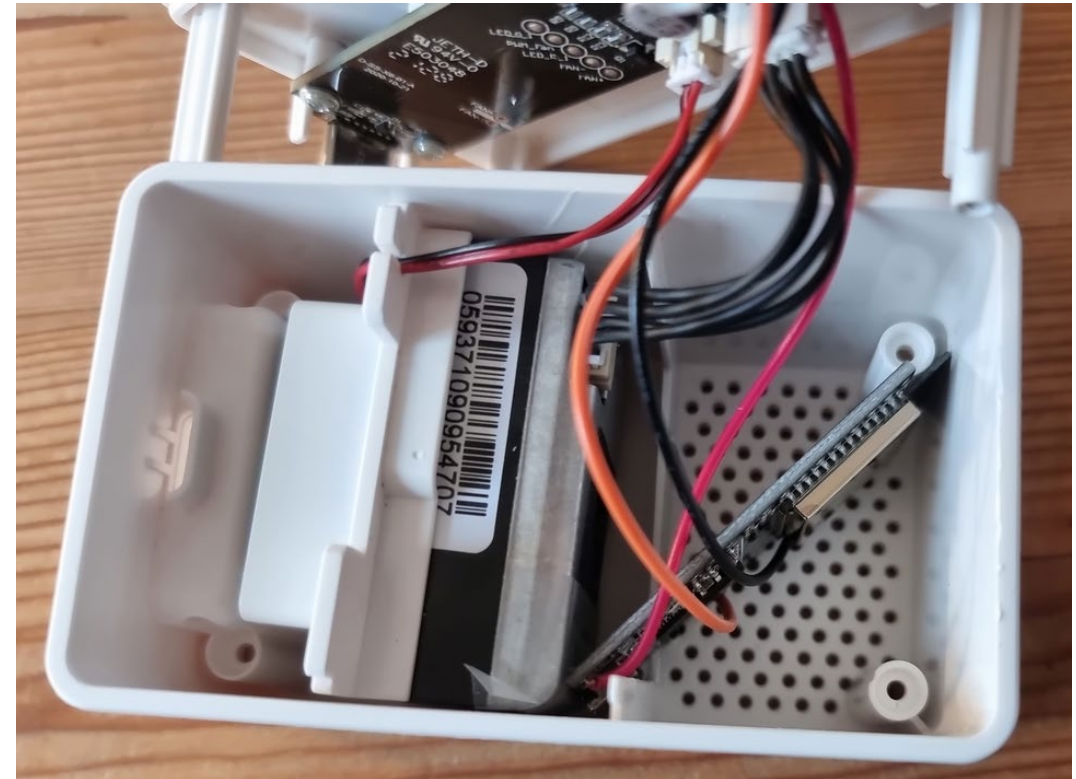


Vorbereitung Gehäuse

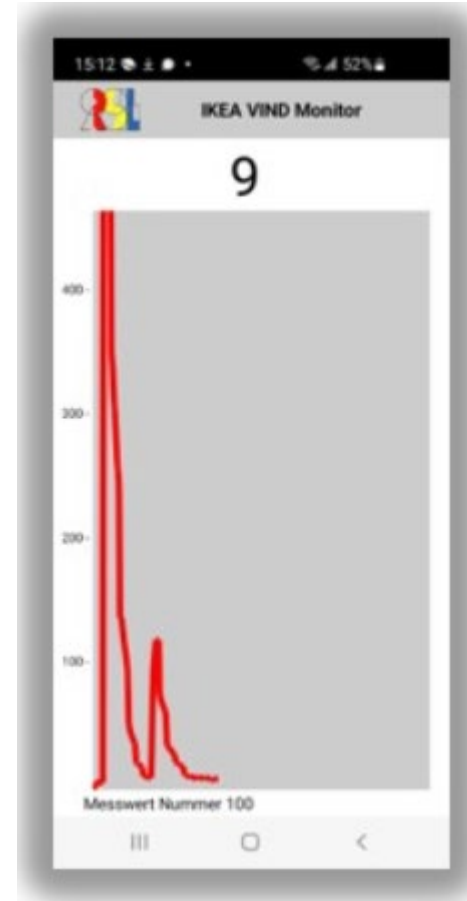
ESP versenken

Der ESP wird nun schräg im Gehäuse untergebracht – bitte bis ganz nach unten schieben!

Der Sensor mit Lüfter wird dabei ganz nach links geschoben.



Fertig!



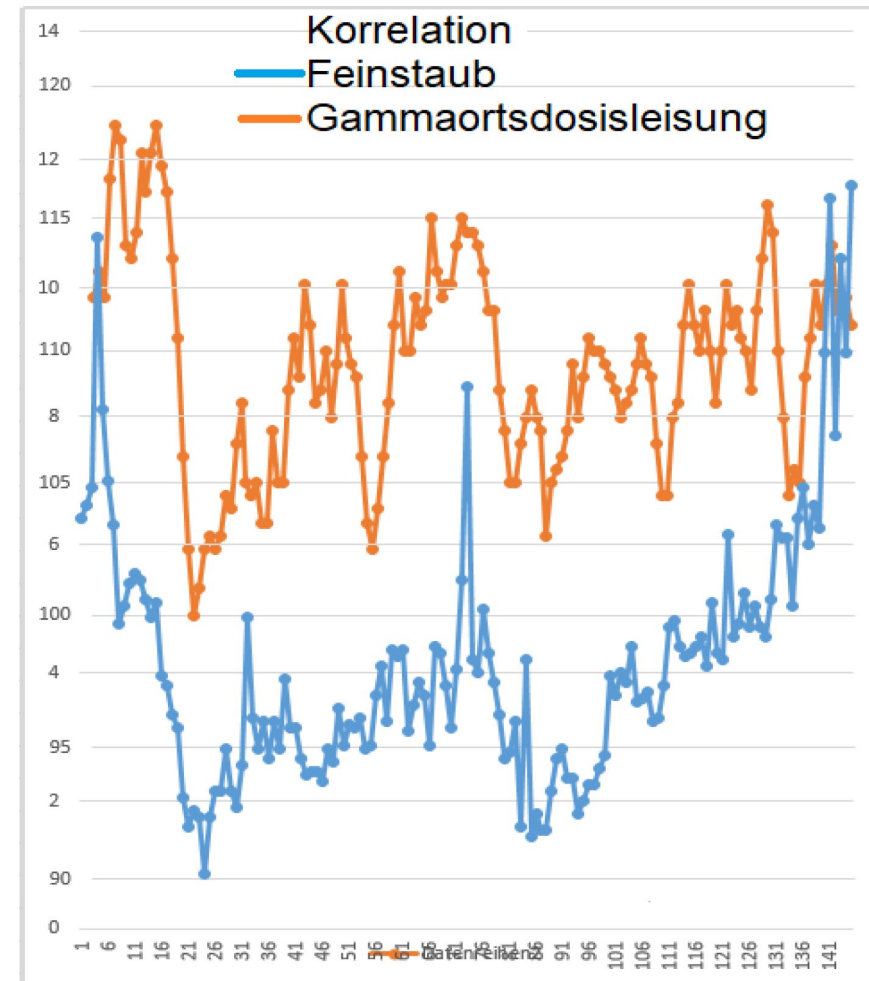
Projekt-Idee: Korrelation Feinstaub \Leftrightarrow Radioaktivität?

Wie geht's weiter?

Umweltmessstationen stellen regelmäßig einen Zusammenhang zwischen Radioaktivität und Feinstaub PM 2.5 fest.

Es wird angenommen, dass sich Radon-Zerfallsprodukte an Feinstaubpartikeln festsetzen.

Gesundheitliche Relevanz: Feinstaub verbleibt in der Lunge, Radon wird wieder ausgeatmet.




```

IKEA_ESP32

#define BT_NAME "IKEA_BT_016" // Individual devicename of ESP32 Bluetooth
#define LED_BUILTIN 2 // Blue LED on ESP to indicate: ESP gets data from IKEA sensor

// *****
// Communication with IKEA Sensor via UART2
HardwareSerial ikeaSerial(2);

uint8_t serialRxBuf[20]; // Buffer for read data
uint8_t rxBufIdx = 0; // Buffer pointer

// *****
// Provide data via Bluetooth
// https://youtu.be/aM2ktMKaunw
#include <BluetoothSerial.h>

// init Bluetooth class
BluetoothSerial ESP_BT;

/*****
void clearRxBuf()
{
    // Clear everything for the next message
    memset(serialRxBuf, 0, sizeof(serialRxBuf));
    rxBufIdx = 0;
}

/***** Sketch Code setup *****/
void setup()
{
    pinMode(LED_BUILTIN, OUTPUT);
    Serial.begin(115200);
    delay(500); // 500 ms is enough time to swith USB between programming and Serial monitor

    /***** Init serial communication to IKEA vintriktning *****/
    ikeaSerial.begin(9600);

    if (!ikeaSerial)
    { // If the object did not initialize, then its configuration is invalid
        Serial.println("Invalid SoftwareSerial pin configuration, check config");
    }
    else
    {
        Serial.println("+++ UART2 to IKEA sensor initialized");
    }

    /***** Setup Bluetooth Connection *****/
    ESP_BT.begin(BT_NAME); //Name of your Bluetooth interface -> will show up on your phone
    Serial.print("+++ Bluetooth initialized as *** ");
    Serial.print(BT_NAME);
    Serial.println(" ***");

    clearRxBuf();
    Serial.println("Waiting for sensor ...");
}

```

```

IKEA_ESP32

/***** Sketch Code loop *****/
void loop()
{
    digitalWrite(LED_BUILTIN, LOW); // Blue LED off during waiting for sensor data
    while (!ikeaSerial.available()) // Wait until IKEA sensor sends data
    {
        ;
    }

    // Data coming from sensor
    digitalWrite(LED_BUILTIN, HIGH); // Blue LED on indicates data are coming from IKEA sensor ...

    // Read one paket sensor data (20 Bytes) and save into 'serialRxBuf'
    while (ikeaSerial.available())
    {
        serialRxBuf[rxBufIdx++] = ikeaSerial.read(); // read data to rxBuf
        // Without this delay, receiving data breaks for reasons that are beyond me
        delay(15);

        if (rxBufIdx > 20) // To avoid stack overflow. 20 Bytes in one paket
            clearRxBuf();
    }

    // Plausibility Check. First five bytes must be 0x16 0x11 0x0b 0x00 0x00
    // length of dataset must be 20
    bool headerValid =
        serialRxBuf[0] == 0x16 &&
        serialRxBuf[1] == 0x11 &&
        serialRxBuf[2] == 0x0B &&
        serialRxBuf[3] == 0x00 &&
        serialRxBuf[4] == 0x00;

    if (headerValid & (rxBufIdx == 20))
    {
        // Relevant information can be found in Byte 5 and 6 from IKEA sent data
        const int ikeaValue = (serialRxBuf[5] << 8) | serialRxBuf[6];

        // Send data to Serial Monitor
        Serial.print(ikeaValue);
        Serial.println();

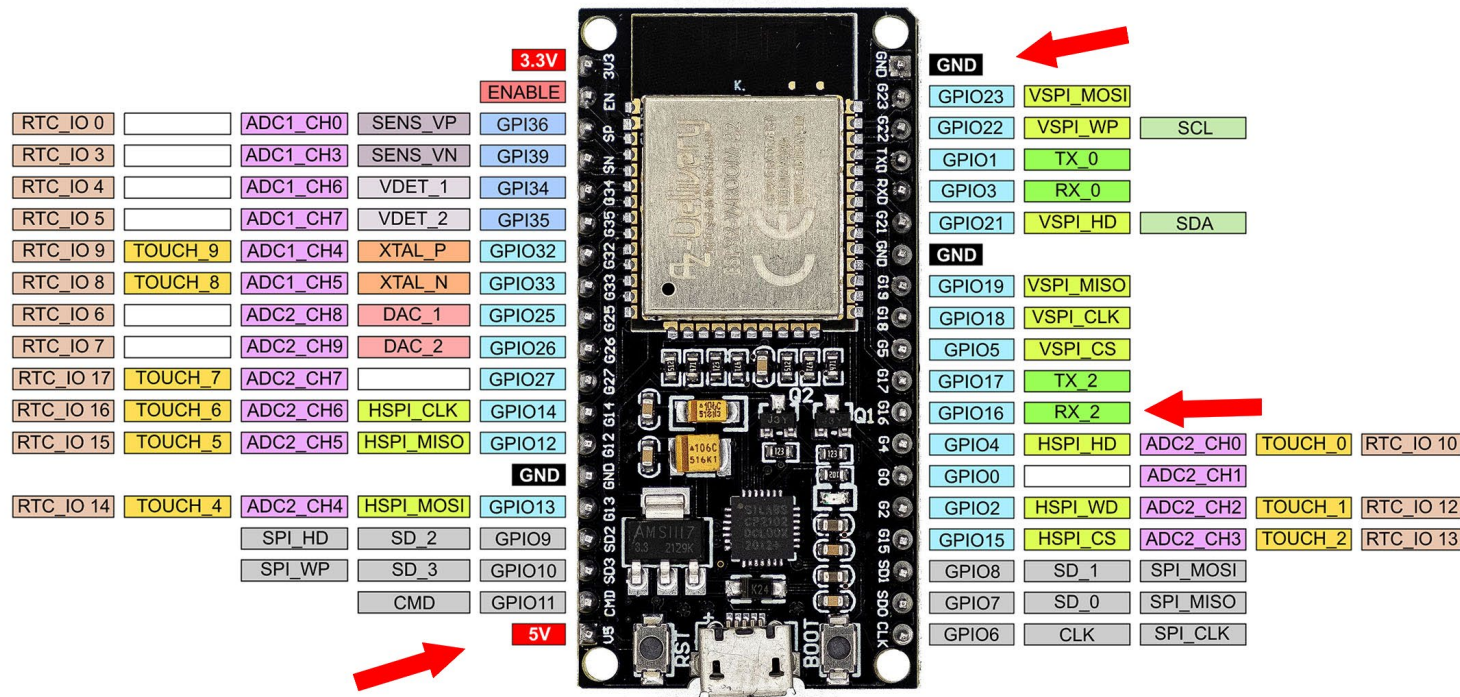
        // Send data via Bluetooth as String
        // with leading #
        String sendString="#";
        sendString.concat(String(ikeaValue));
        ESP_BT.println(sendString);

        // For information only:
        // Uncomment, if you want to display one line with complete sensor raw-data from IKEA sensor in Serial Monitor
        /*
        for (int i = 0; i < rxBufIdx; i++)
        {
            Serial.printf("%02x ", serialRxBuf[i]);
        }
        Serial.println();
        */
    }
    clearRxBuf();
}

```

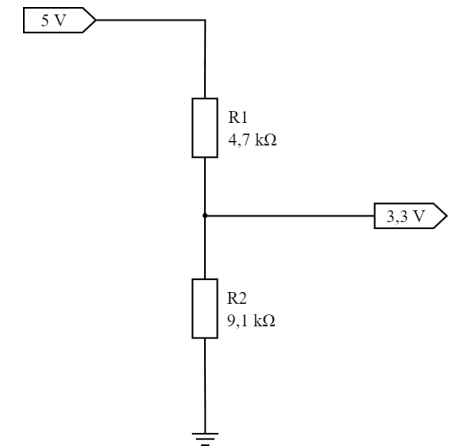
Appendix 1

ESP32 Pin-Belegung:



Logik-Pegel:

der Vindriktning-Sensor liefert ein 5V TTL- Pegel, wohingegen der ESP mit 3,3V-Pegeln arbeitet. Streng genommen müsste man eine Pegelanpassung wie folgt vornehmen. Die Praxis zeigt, dass die meisten ESPs aber auch die 5V „aushalten“.

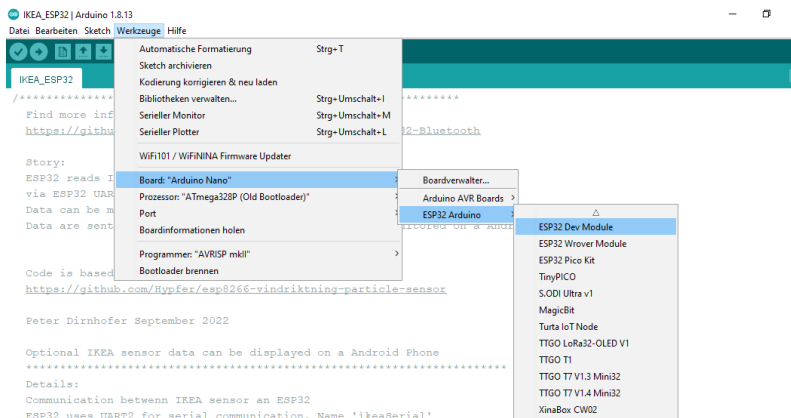


Appendix 2

ESP32 mit Arduino IDE programmieren:

<https://wolles-elektronikkiste.de/esp32-mit-arduino-code-programmieren>

Board-Verwalter: ESP Dev Module



1. Geht in der Arduino IDE auf Datei -> Voreinstellungen.
 2. Dort klickt ihr auf das kleine Symbol hinter „Zusätzliche Boardverwalter URLs“.
 3. Tragt „https://dl.espressif.com/dl/package_esp32_index.json“ als separate Zeile ein (ohne Anführungszeichen).
 4. Geht auf Werkzeuge -> Board -> Boardverwalter.
 5. Sucht nach „esp32“ und installiert „esp32 by Espressif Systems“.
 6. Ggf. müsst ihr die Arduino IDE neu starten.
- Nach der Installation findet ihr mehrere Dutzend ESP32 Boards zur Auswahl. Da müsst ihr schauen welches passt. Weitverbreitet ist das „ESP32 Dev Module“.

Komplett-Überblick:

<https://github.com/PeterDirnhofer/IKEA-vindriktning-ESP32-Bluetooth>

Weiterführende Literatur:

<https://www.heise.de/ratgeber/Ikea-Feinstaubsensor-Vindriktning-zum-IoT-Device-aufbohren-6164149.html>

<https://github.com/Hypfer/esp8266-vindriktning-particle-sensor>