



Externe Datenkommunikation mit Calliope mini

Dr. Thomas Meinike

Hochschule Merseburg
Fachbereich Wirtschaftswissenschaften
und Informationswissenschaften

Forschungskolloquium

Merseburg, 08.04.2026

Einstieg 1/5

- Calliope mini (1/2/3) kommen seit 2017 zum Einsatz, speziell in Kursen zur Web-Entwicklung mit JavaScript, falls individuelles Interesse dafür besteht.
- Über die letzten zehn Monate ist ein Rahmenprojekt zur externen Datenkommunikation entstanden, speziell für die Abfrage und Verarbeitung von Sensordaten.

Externe Datenkommunikation mit Calliope mini – Serielle Übertragung von Daten und mittels WLAN

Forschungsprojekt von Dr. Thomas Meinike
Hochschule Merseburg
Fachbereich Wirtschaftswissenschaften und Informationswissenschaften

Ausgangspunkt

Der Einplatinen-Computer *Calliope mini* [1] kommt seit seiner Einführung im Jahr 2017 auch an der Hochschule Merseburg zum Einsatz, konkret in Kursen des Autors zum Thema Web-Entwicklung [2]. Kenntnisse allgemeiner Grundlagen zu Aufbau und Nutzung dieser Geräte werden hier vorausgesetzt. Abbildung 1 zeigt den grundsätzlichen technischen Aufbau.

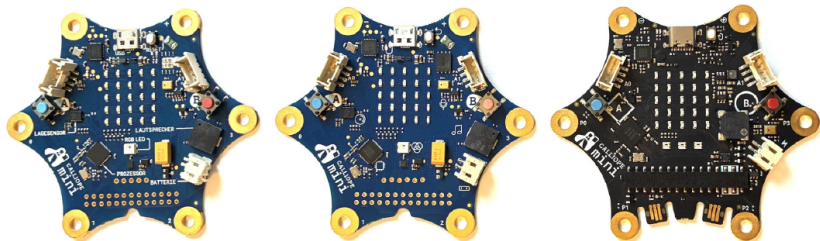


Abb. 1: Calliope mini im Überblick (von links nach rechts: V1, V2, V3)

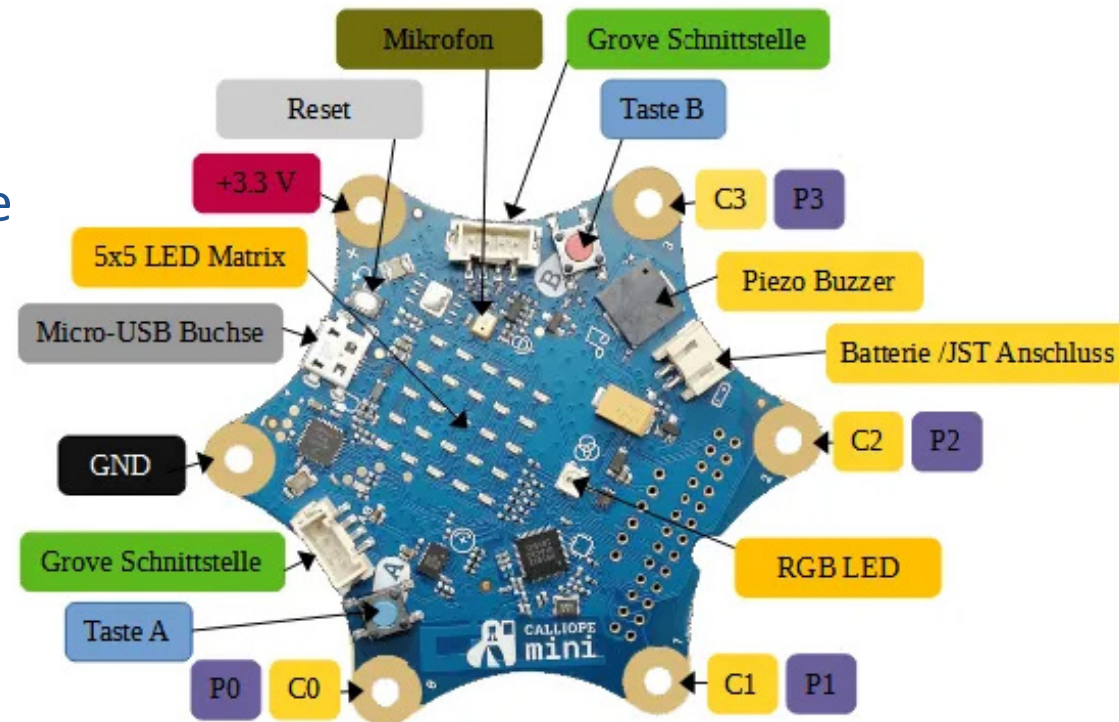
Projektbeschreibung mit bisher 28 Seiten



Vortrag zum MNU-Bundeskongress in Saarbrücken

Einstieg ^{2/5}

- Im Projekt verwendete Calliope-Basiskomponenten
 - LED-Matrix mit 25 roten LEDs, einzeln ansprechbar
 - Frei programmierbare RGB-LEDs (eine bei C1/C2 bzw. drei bei C3)
 - Tasten A und B zur Steuerung von Programmaktionen
 - Grove-Schnittstellen A0 (I²C) und A1 (UART) für externe Peripherie wie Sensoren, Displays, WLAN-Modul
 - Interne Sensoren für Temperatur, Lichtintensität und Geräuschpegel



Bildquelle: <https://draeger-it.blog/vorstellung-des-calliope-mini/>

Einstieg ^{3/5}

- Im Projekt verwendete Peripherie für Grove-Anschlüsse

1 | SCD40-Sensor (CO₂, Temperatur, Luftfeuchtigkeit)

2 | Ultrasonic Ranger (Abstandsmessung)

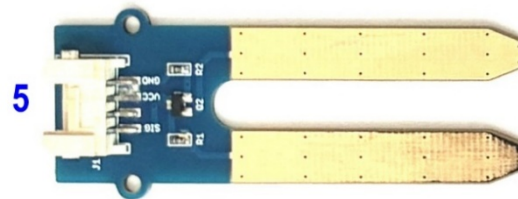
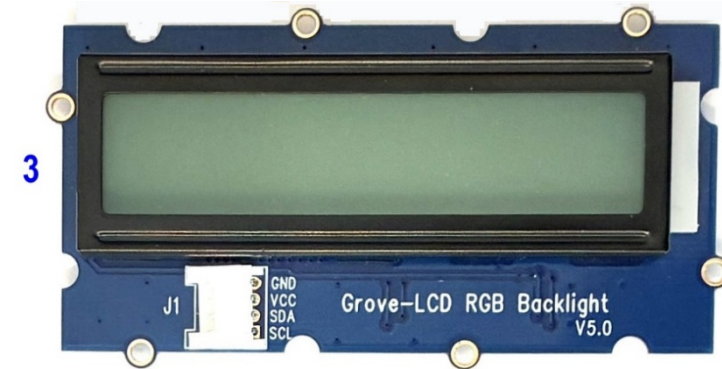
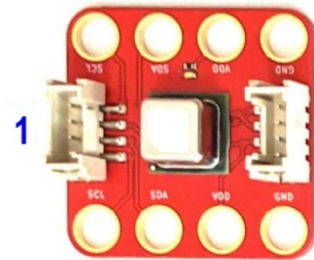
3 | 16x2-Display LCD RGB Backlight

4 | Light Sensor

5 | Moisture Sensor

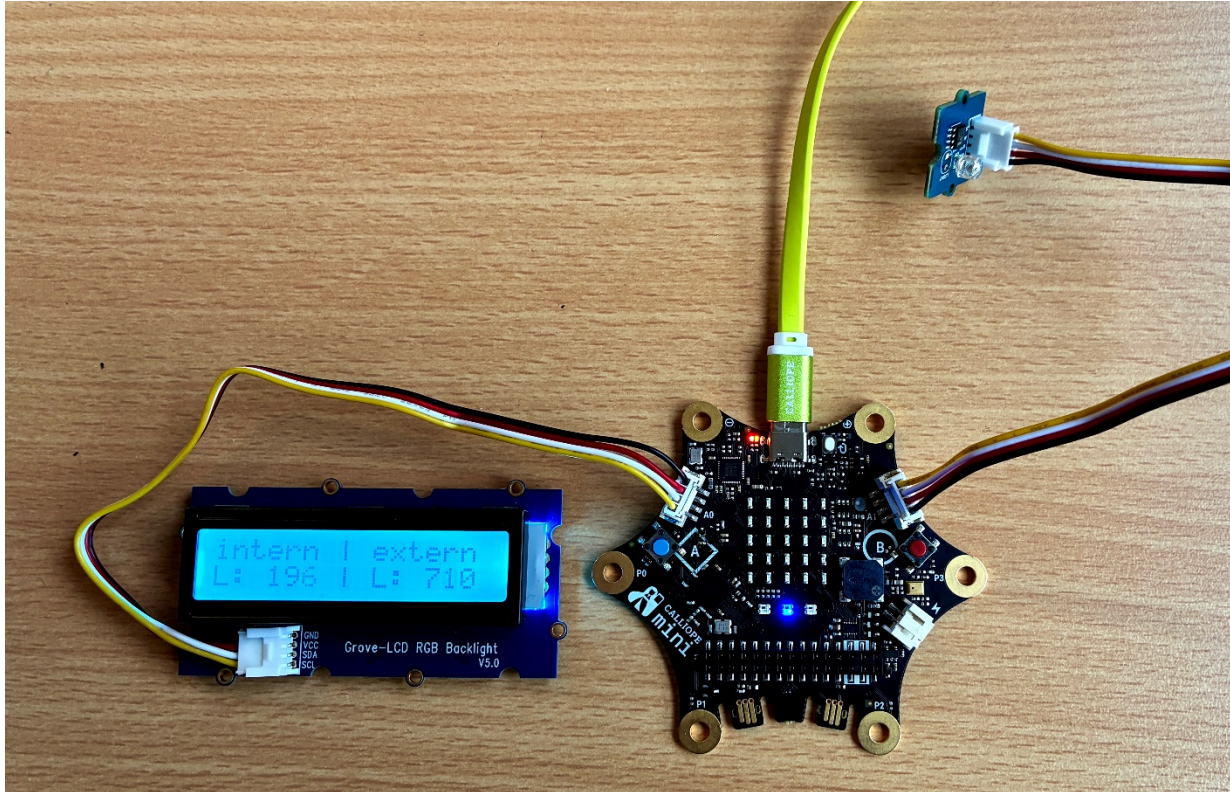
6 | WLAN-Modul ESP8266

7 | 4-Digit-Display

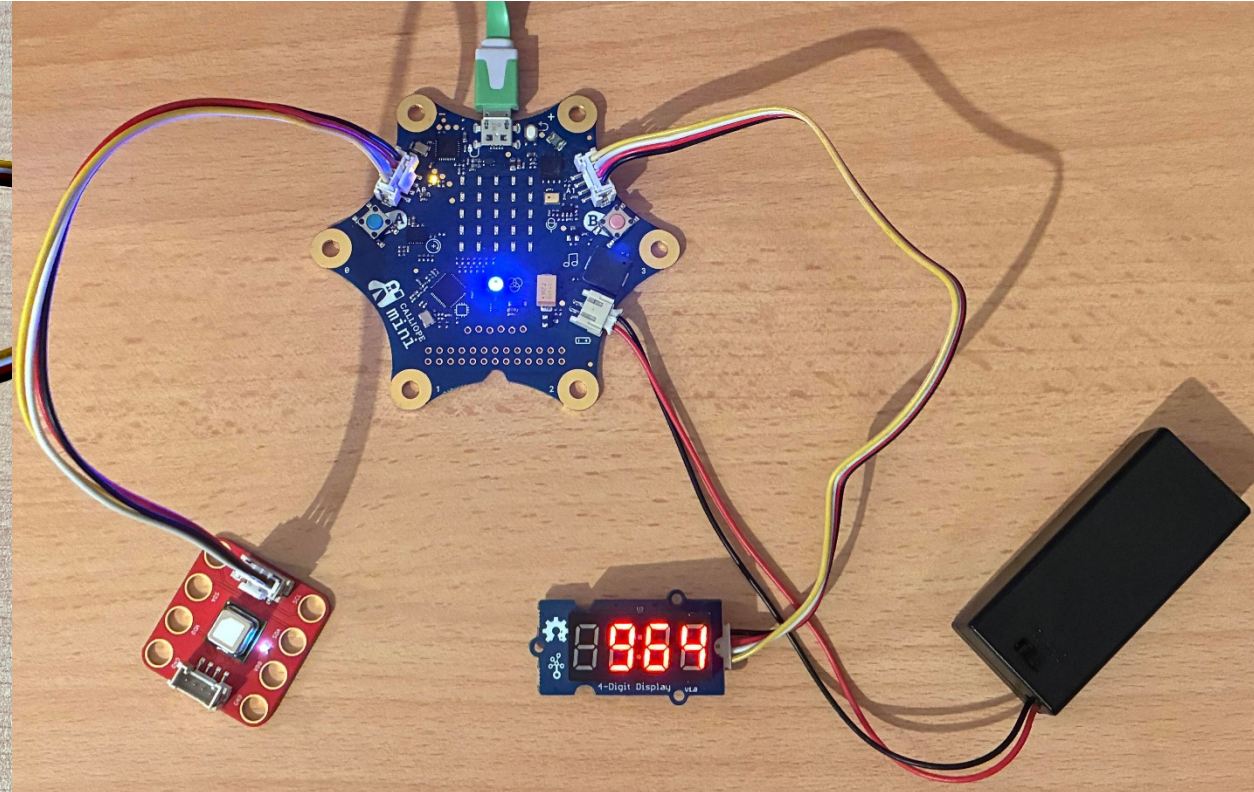


Einstieg 4/5

- Beispiel-Projekte mit Peripherie



Externe Lichtsensordaten an 16x2-Display



CO₂-Messung (ppm) an 4-Digit-Display

- Programmierung mit der MakeCode-Umgebung und zusätzlichen Erweiterungen

Microsoft MakeCode for Calliope mini

Blöcke JavaScript

Suche...

Grundlagen

Eingabe

Musik

LED

C02-Sensor SCD40

Schleifen

Logik

Variablen

Mathematik

Motoren

Erweiterungen

Fortgeschritten

```
1 basic.forever(function() {
2   // eigener Code für dauerhafte Aktionen ...
3 })
4
5 input.onButtonEvent(Button.A, input.buttonEventClick(), function() {
6   // eigener Code für Aktionen nach Drücken der A-Taste ...
7 })
8
9 function CO2_Messung(): number {
10  // eigene Funktion ...
11  return SCD40.get_co2()
12 }
```

Erweiterungen

SCD40

RGB Klima Motor Lichter und Bildschirm

Anzeige Software Jacdac KI IoT

Audio Wissenschaft Netzwerk

Startseite

C02-Sensor-SCD40

Herunterladen

Calli_Test

Daten seriell senden und empfangen ^{1/7}

- Calliope mini können seit Version 1 seriell kommunizieren

✓ Initialisierung

- Standardwert 115200 Baud, 9600 erwies sich als ausreichend.

```
serial.redirectToUSB()  
serial.setBaudRate(9600)
```

✓ Daten senden

- Einzelne Werte oder mehrere z. B. als JSON-Strings.

```
let JSONstr: string = "..."  
serial.writeLine(JSONstr)
```

✓ Daten empfangen

- Einzelne Werte oder ebenfalls zeilenweise.

```
let received: string = serial.readLine()
```

serial.

- writeValue
- writeLine
- writeString
- writeNumber
- writeNumbers
- readLine
- readUntil
- readString
- onDataReceived
- redirect
- redirectToUSB
- delimiters

Daten seriell senden und empfangen 2/7

- Daten für Browser-Verarbeitung als JSON-String ausgeben (1)

```
let send: boolean = false
```

```
input.onButtonEvent(Button.A, input.buttonEventClick(), function() {  
  let L: number = 0  
  let S: number = 0  
  let T: number = 0  
  let F: number = 0  
  
  serial.redirectToUSB()  
  serial.setBaudRate(9600)  
  send = true  
})
```

```
input.onButtonEvent(Button.B, input.buttonEventClick(), function() {  
  send = false  
})
```

- ✓ Taste A initialisiert und startet den Sendevorgang, Taste B beendet ihn.
- ✓ Verwendet werden drei interne Sensoren und ein externer für Feuchtigkeit.

Daten seriell senden und empfangen ^{3/7}

- Daten für Browser-Verarbeitung als JSON-String ausgeben (2)

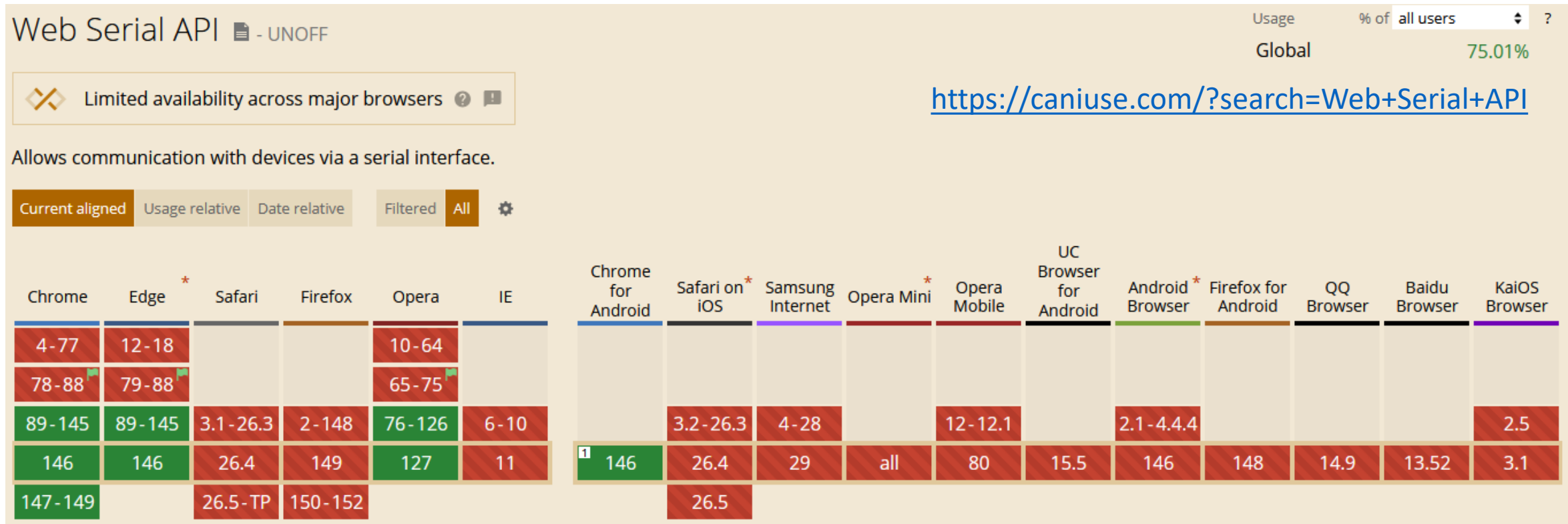
```
basic.forever(function() {  
  if(send) {  
    L = input.lightLevel() // 0 (dark) to 255 (bright)  
    S = input.soundLevel() // 0 (silent) to 255 (loud)  
    T = input.temperature() // °C  
    F = input.moisture() // 0 to 100 % (eigene Erweiterung des input-Namespaces)  
  
    JSONstr = '{ "L" : ' + L + ', "S" : ' + S + ', "T" : ' + T + ', "F" : ' + F + ' }'  
  
    serial.writeLine(JSONstr)  
  }  
  basic.pause(3000)  
})
```

- ✓ Es wird alle drei Sekunden ein Datenpaket gesendet, bis das Senden über die Taste B abgebrochen wird.
- ✓ Ein typischer JSON-String hat die Form { "L" : 58, "S" : 93, "T" : 25, "F" : 81 } mit den Komponenten **L**icht, **S**ound, **T**emperatur und **F**euchtigkeit.

Daten seriell senden und empfangen 4/7

- Daten im Browser über die Web Serial API verarbeiten (1)

- ✓ Die WSA ist eine aktuell nur in Chromium-Browsern etablierte JavaScript-DOM-Schnittstelle.



- ✓ Über die WSA können Daten gelesen und auch gesendet werden.

Daten seriell senden und empfangen 5/7

- Daten im Browser über die Web Serial API verarbeiten (2)

✓ Calliope mini wird über einen COM-Port verbunden → WSA übernimmt.

```
window.addEventListener("load", function() {
  // Initialisierungen

  // Serielle Verbindung
  if(navigator.serial) { readSerialData(); }
  else { // Web Serial API nicht unterstützt. }

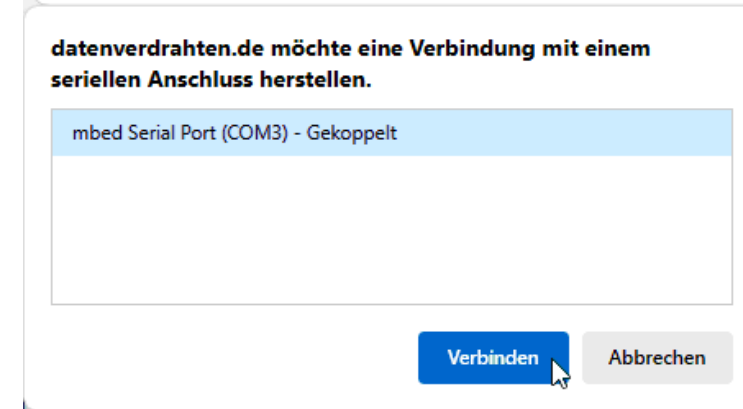
  // Daten abfragen und verarbeiten
  async function readSerialData() {
    try {
      // Anfrage zur Auswahl des seriellen Geräts
      port = await navigator.serial.requestPort();

      if(port.connected) {
        // Port mit Parametern für Calliope mini öffnen
        await port.open({ baudRate: 9600, bufferSize: 64 });

        // Daten UTF-8-kodiert empfangen
        reader = port.readable.getReader();
        const textdecoder = new TextDecoder("UTF-8");
```

```
while(true) {
  const { value, done } = await reader.read();
  if(done) { reader.releaseLock(); break; }
  else if(value) {
    // Daten (value) auswerten
    const buffer = textdecoder.decode(value);
    const line = buffer.trim();
    processData(line);
  }
}
}
}
catch(error) { // Fehlermanagement }
finally { reader.releaseLock(); }
});

// JSON-Daten als Tabelle und SVG-Graphen ausgeben
function processData(jsonstr) { ... }
});
```




Daten seriell senden und empfangen ^{6/7}


- Daten im Browser über die Web Serial API verarbeiten (3)
 - ✓ Darstellung der Ergebnisse tabellarisch und pro Sensorwert ein Graph als SVG-Polyline.

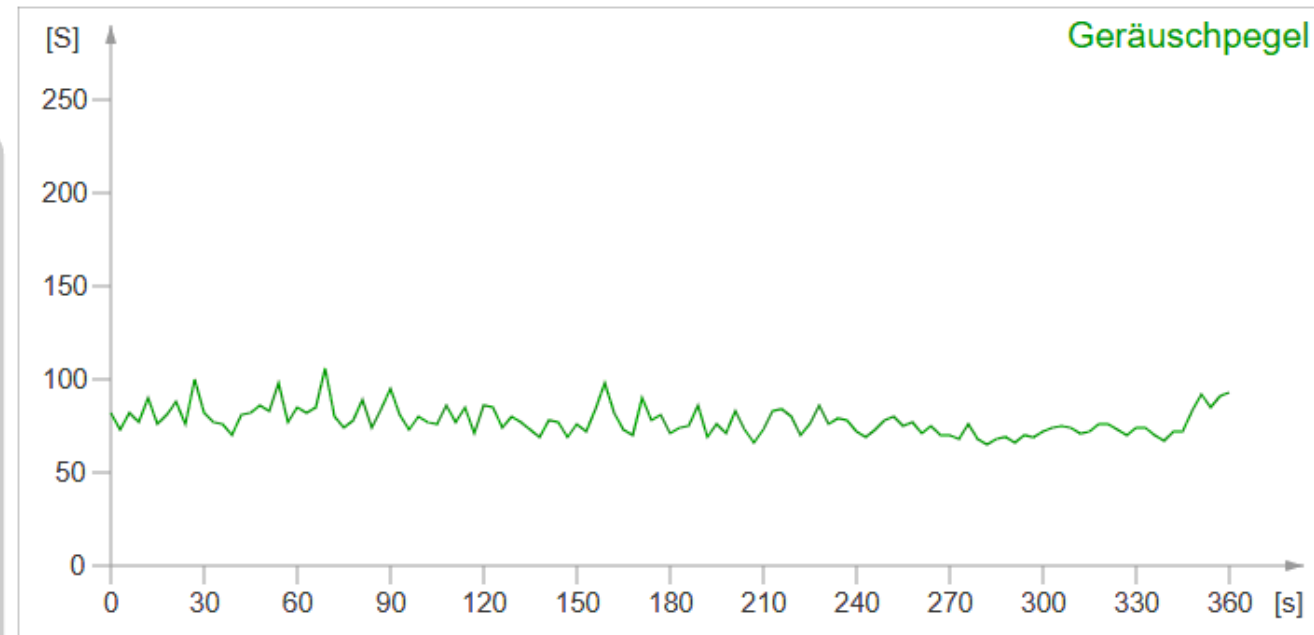
Serielle Verbindung mit Calliope mini

[Programm für den Calliope mini zum Senden von Daten](#)

Mit Port verbinden und Daten abfragen

Daten empfangen  120 (360 s) **Verbindung trennen**

JSON-String	{ "L" : 58, "S" : 93, "T" : 25, "F" : 81 }	
Licht (0-255)	58	<input type="checkbox"/>
Sound (0-255)	93	<input checked="" type="checkbox"/>
Temperatur (°C)	25	<input type="checkbox"/>
Feuchtigkeit (%)	81	<input type="checkbox"/>



- ✓ Aus diesem Prototyp wurden weitere Projekte abgeleitet, u. a. zum längeren CO₂-Monitoring.

Daten seriell senden und empfangen 7/7

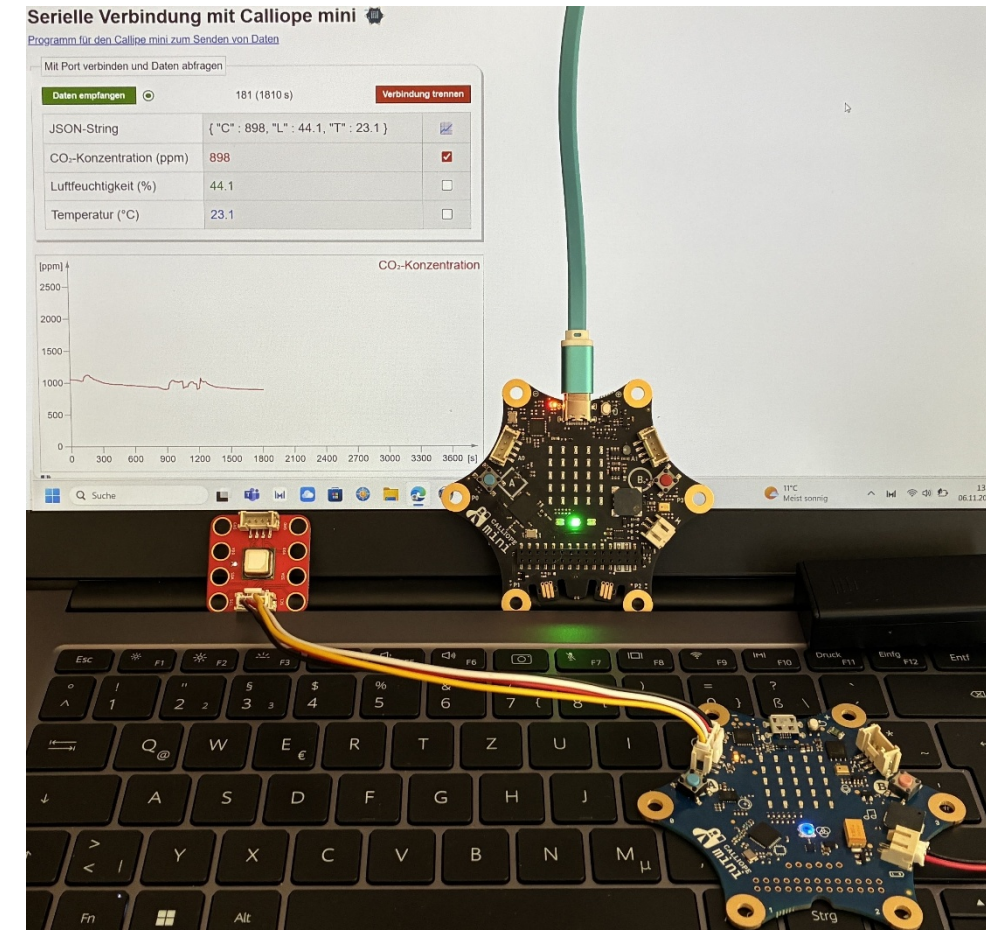
- Daten im Browser über die Web Serial API verarbeiten (4)

→ [Live-Demo](#)

- ✓ Erweitertes Szenario für CO₂-Messung mit zwei über Funk (Bluetooth) verbundenen Calliope mini.
- ✓ Gerät A sammelt die Daten vom SCD40-Sensor und sendet diese periodisch an Gerät B, welches über die WSA mit dem Browser kommuniziert.

```
radio.setGroup(23)
radio.setTransmitPower(7)
// ...
function sendData(): void {
  let C: number = SCD40.get_co2() // ppm
  let L: number = SCD40.get_relative_humidity() // %
  let T: number = SCD40.get_temperature(SCD40.SCD40_T_UNIT.C) // °C
  let D: string = C + "," + L + "," + T
  radio.sendString(D)
}

// ...
radio.onReceivedString(function(receivedString: string) {
  // Auf Gerät B den empfangenen Datensatz verarbeiten
})
```



Weitere behandelte Aspekte

- Daten aus dem Browser über die WSA an Calliope mini senden (LED-Ansteuerung)

Serielle Verbindung mit Calliope mini 🌐

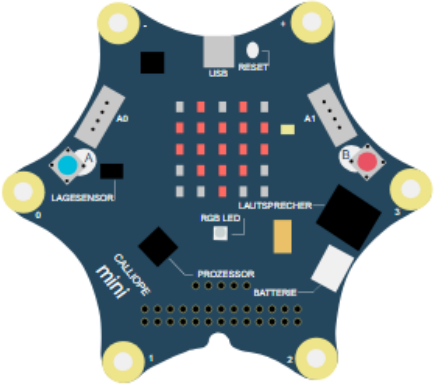
[Programm für den Calliope mini zum Empfangen von Daten](#)

Mit Port verbinden und Daten senden

Daten senden Calliope mini ist verbunden. **Verbindung trennen**

LED-Matrix

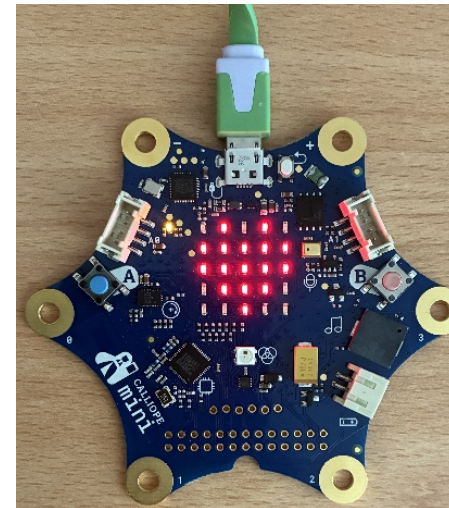
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



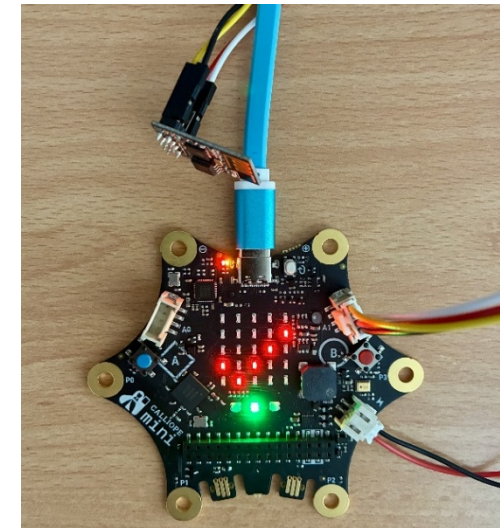
Calliope-Code

```
basic.showLeds(`  
 . # . # .  
# # # # #  
# # # # #  
 . # # # .  
 . . # . .  
`)  
`)
```

Reset **Kopieren**



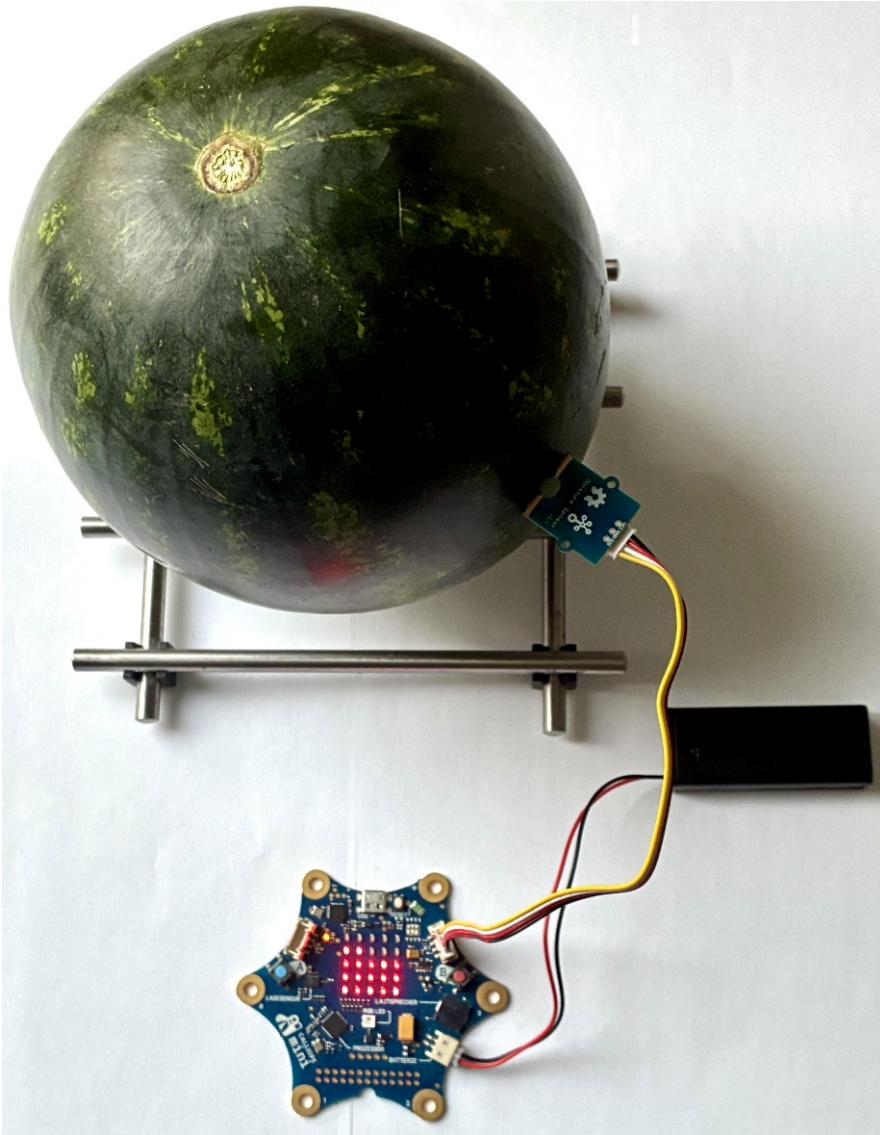
- Daten vom Calliope mini über WLAN-Modul an externe Server senden →
- Und mehr ...



Fazit und Ausblick

- Der Einplatinencomputer Calliope mini ist und bleibt speziell unter Nutzung zusätzlicher Peripherie-Komponenten ein attraktives Gerät für kreative Projekte im Rahmen der schulischen und universitären Ausbildung.
- Die Möglichkeit der Nutzung von JavaScript (oder auch Python) erweitert den Horizont zur Umsetzung eigener Ideen.
- Neuere Entwicklungen im Bereich der Web-Technologien wie Web Serial API (oder auch WebUSB API) schaffen zusätzliche Anreize zu deren Erkundung mit einem etablierten System.
- Ressourcen:
 - Projektartikel: https://datenverdrahten.de/PDF/Calliope_mini_Datenkommunikation.pdf
 - MNU-Vortrag: <https://speakerdeck.com/xmlarbyter/externe-datenkommunikation-mit-calliope-mini>

Danke für die Aufmerksamkeit!



Feuchtigkeitmessung in einer Melone (1 LED = 4%)