

DIE SERIELLE SCHNITTSTELLE AM CALLIOPE MINI

Thorsten Kimmeskamp

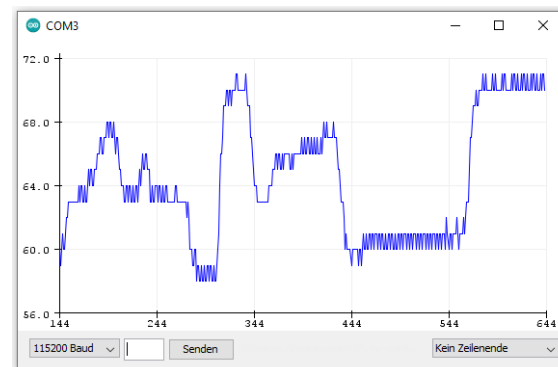
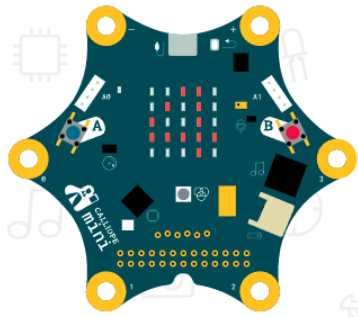
WAS IST DIE SERIELLE SCHNITTSTELLE?

- ❖ die wichtigsten Stichworte liefert der Begriff “Universal Asynchronous Receiver Transmitter (UART)”
- ❖ viele technische Details sind für die reine Anwendung nicht wichtig
- ❖ die einfachste Vorstellung ist die eines „Dosentelefon“ zwischen Calliope und PC



WOBEI HILFT MIR DAS?

- ❖ Besonders toll am Calliope mini sind die vielen Sensoren!
- ❖ Sensoren liefern letztlich nur Zahlen, die es zu deuten gilt
- ❖ Ausgabe mit 5x5-LED-Matrix: für manche Zwecke zu langsam
- ❖ 7-Segment-Anzeige, OLED-Display: kosten Geld, kein Speichern möglich
- ❖ Serial Monitoring: Messdaten in Echtzeit sehen, dauerhaft speichern und später analysieren (z. B. mit Excel)!



WAS BRAUCHE ICH DAFÜR?

- ❖ USB-Kabel
- ❖ PC-Software „Arduino-IDE“, <https://www.arduino.cc/en/software>
- ❖ ein paar MakeCode-Blöcke aus dem Reiter „Fortgeschritten/Seriell“:

Calliope → PC :

A MakeCode block with a dark blue background and a red button. The text "seriell Zeile ausgeben" is in white, and the button "ausgabe" is red with a white downward arrow.

seriell Zeile ausgeben **ausgabe** ▼

PC → Calliope:

A MakeCode block with a red background and a dark blue button. The text "setze" and "auf" are in white, "eingabe" is in a red box with a white downward arrow, and "seriell lesen bis" is in white. The button "neue Zeile ()" is dark blue with a white downward arrow.

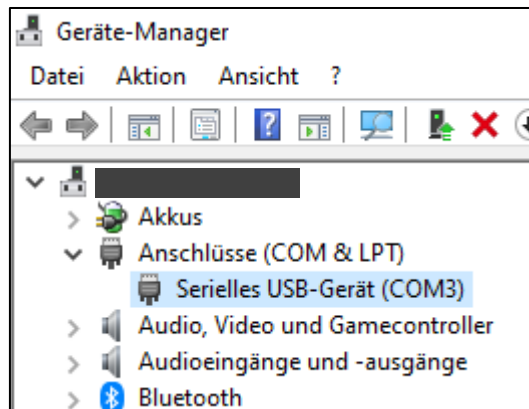
setze **eingabe** ▼ auf **seriell lesen bis** **neue Zeile ()** ▼

EINRICHTUNG: RICHTIGER „COM-PORT“

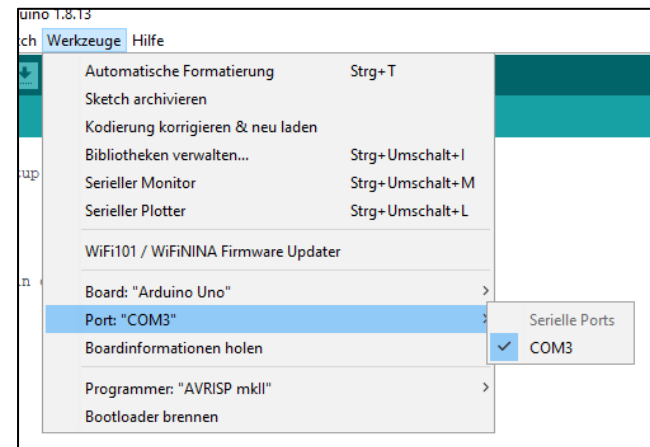
1.



2.



3.

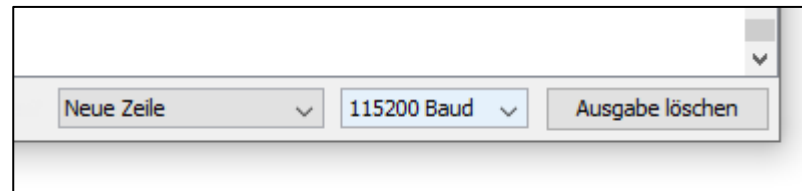


EINRICHTUNG: RICHTIGE „BAUD-RATE“

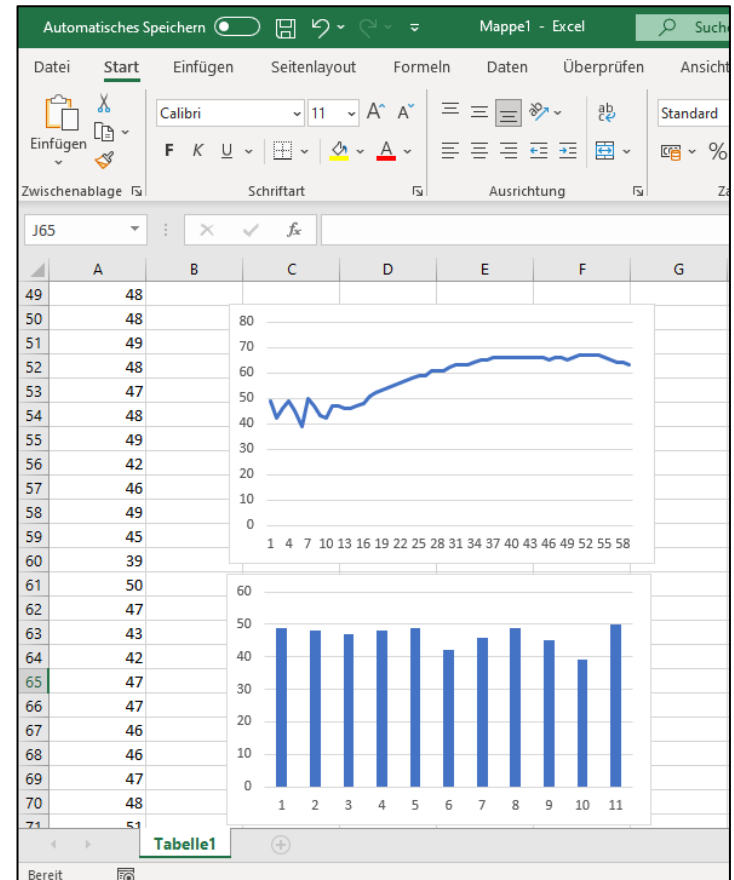
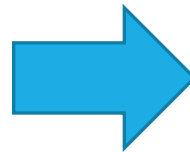
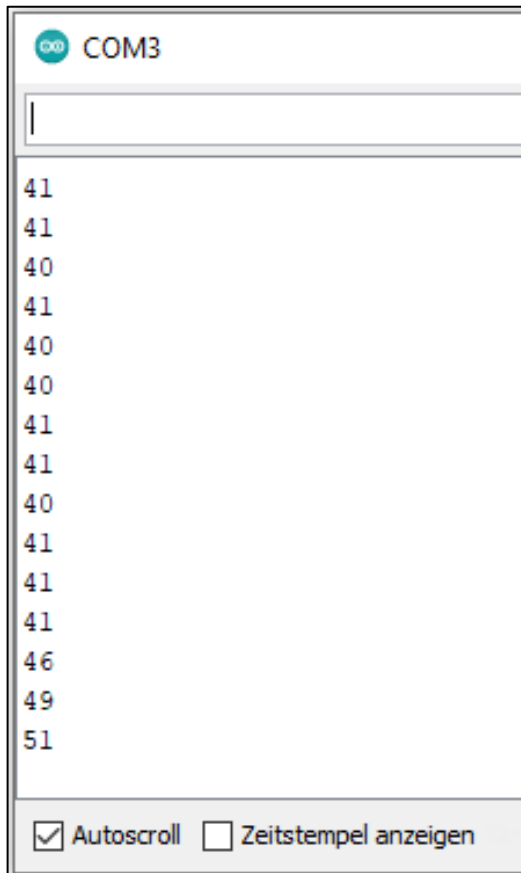
1.



2.

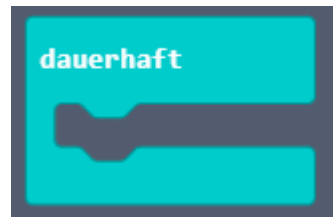


FEATURE 1: „SERIELLER MONITOR“

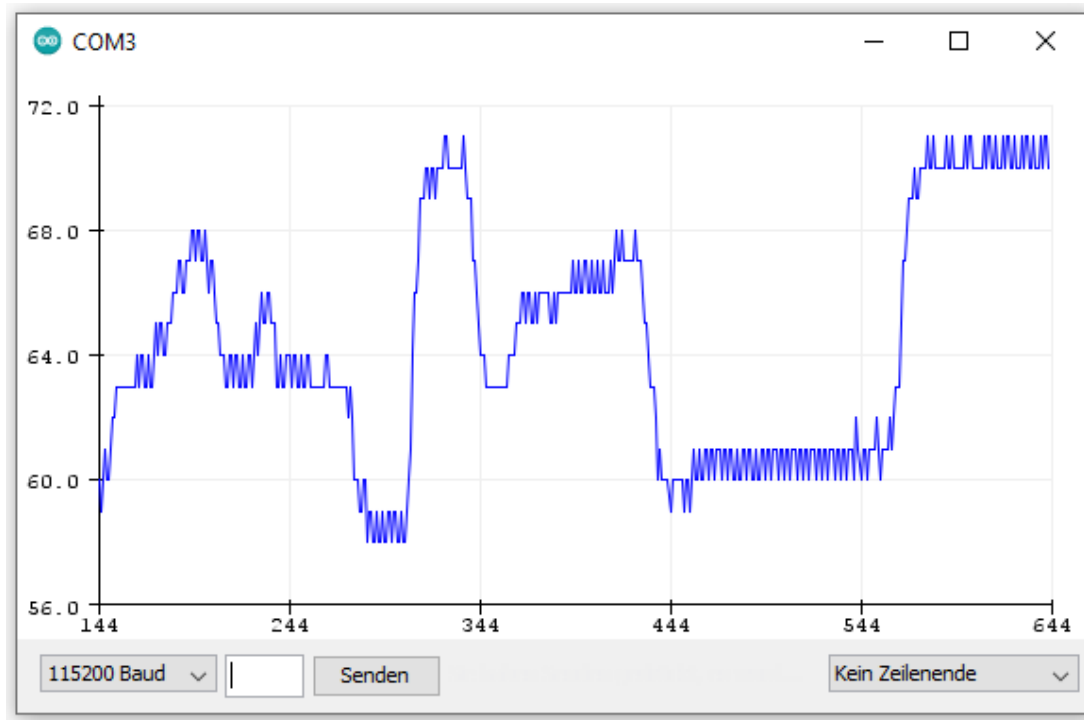


CHALLENGE

- ❖ erstelle mit dem Serial Monitor ein Protokoll der Helligkeit



FEATURE 2: „SERIELLER PLOTTER“



CHALLENGE

- ❖ erstelle mit dem Serial Plotter ein Protokoll der Beschleunigung
- ❖ zeige alle drei Achsen (x, y, z) gleichzeitig an (indem du die einzelnen Werte durch Kommata trennst)
- ❖ beschrifte die Achsen (schreibe beim Start des Programms eine Zeile mit den Namen der Datenreihen anstelle der Werte)

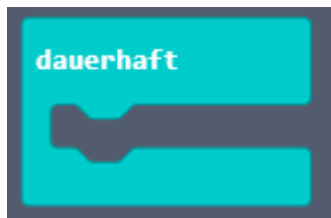


CHALLENGE

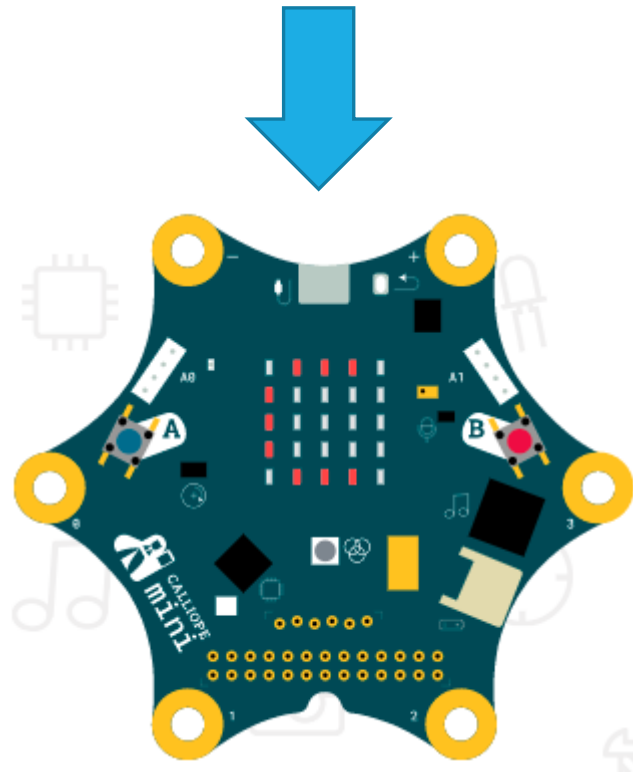
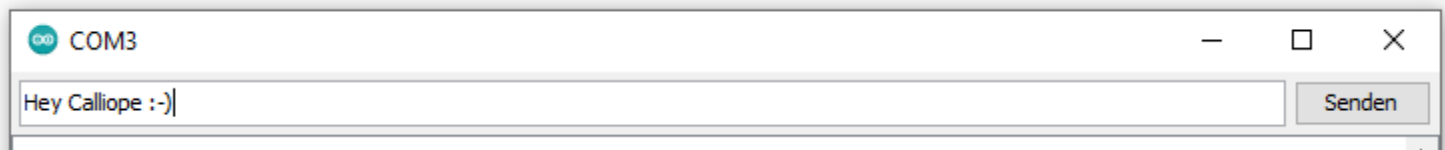
- ❖ erstelle mit dem Serial Plotter ein Zeitablaufdiagramm, das anhand einer Linie anzeigt, wenn Taste A gedrückt (oder losgelassen) wird



- ❖ Bonus: Taste A und Taste B in einem Diagramm :-)

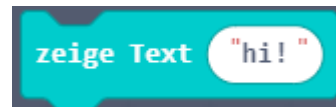
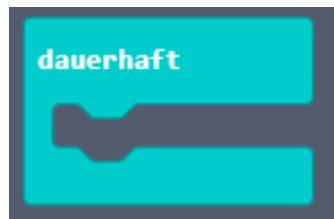


FEATURE 3: „EINGABE MIT PC-TASTATUR“

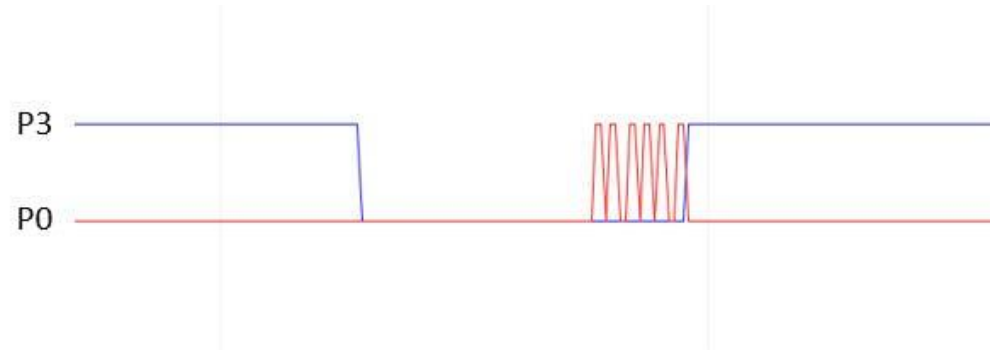
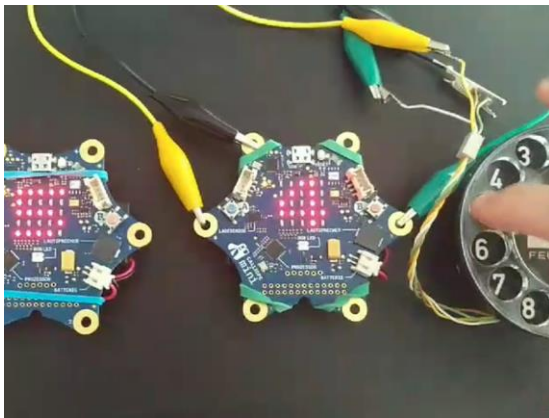


CHALLENGE

- ❖ wenn immer jemand mit dem Serial Monitor seinen Namen eingibt: lass den Calliope ihn anzeigen!
- ❖ wir überprüfen nicht, ob es wirklich ein Name oder ein anderer Text ist...

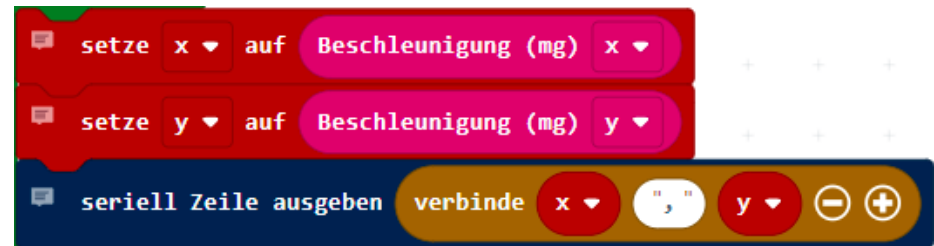
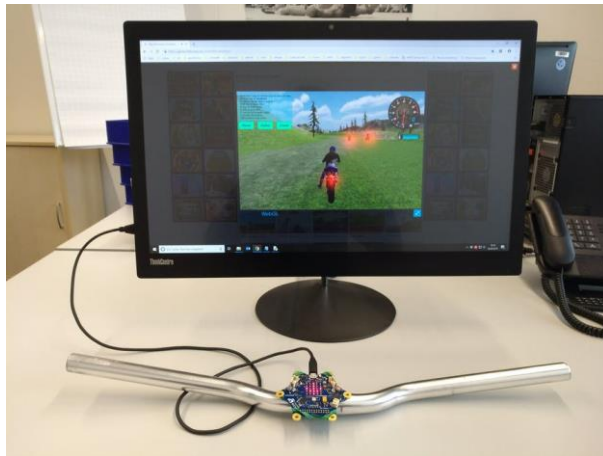


ANWENDUNG: CALLIOPE-TELEFON



<https://www.hackster.io/thorsten-kimmeskamp/calliope-telefon-f6777d>

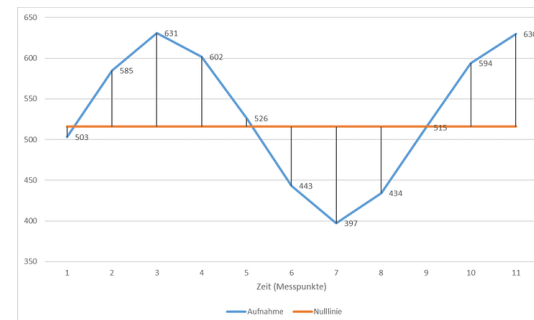
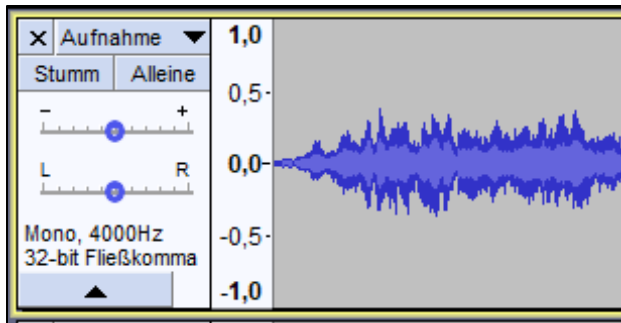
ANWENDUNG: „MINI-BIKE“



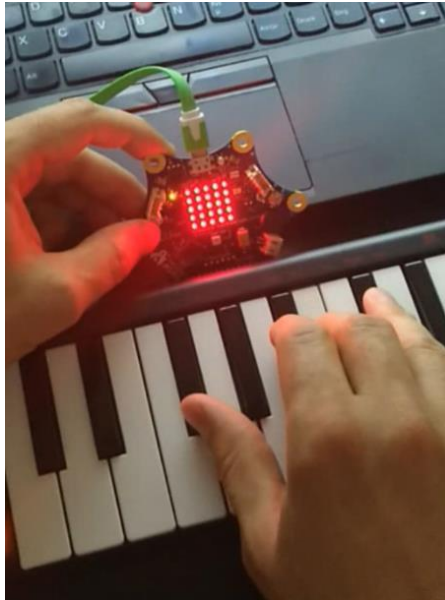
<https://www.hackster.io/thorsten-kimmeskamp/calliope-mini-als-motorrad-controller-50a10d>

ANWENDUNG: TAPE-RECORDER

```
für i von 0 bis 399
mache
  audioPuffer füge Wert analoge Werte von Pin MIC am Ende hinzu
  Warte (µs) 250 - 115
für i von 0 bis 399
mache
  seriell Zeile ausgeben audioPuffer rufe Wert ab bei i
```



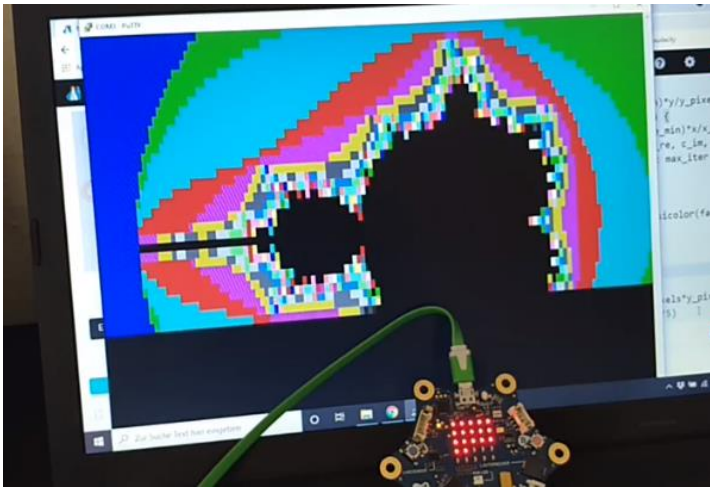
ANWENDUNG: MIDI-KEYBOARD



```
setze data auf seriell lese Zwischenspeicher 3
wenn data rufe Wert ab bei 0 = 144 dann
wenn data rufe Wert ab bei 2 > 0 dann
Note (Hz) freq rufe Wert ab bei data rufe Wert ab bei 1
```

https://youtu.be/3gwgZdYj_A

ANWENDUNG: EXTERNES DISPLAY

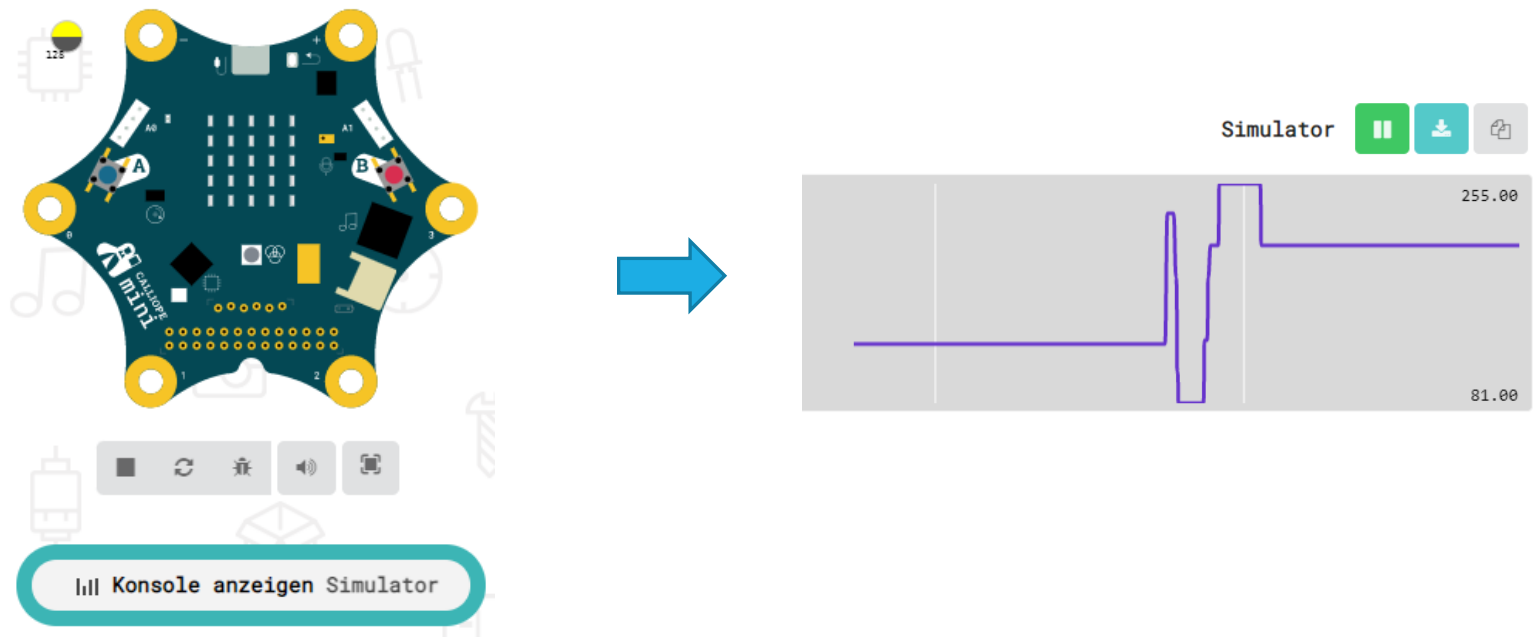


```
/**  
 * Wählt eine Vordergrundfarbe gemäß Farbtabelle im ANSI-Standard.  
 */  
//% blockId="wähleVordergrundNr" block="wähle Vordergrund %value"  
export function wähleVordergrundNr(value: number): void {  
    serial.writeString("\x1B[38;5;" + value + "m");  
}
```

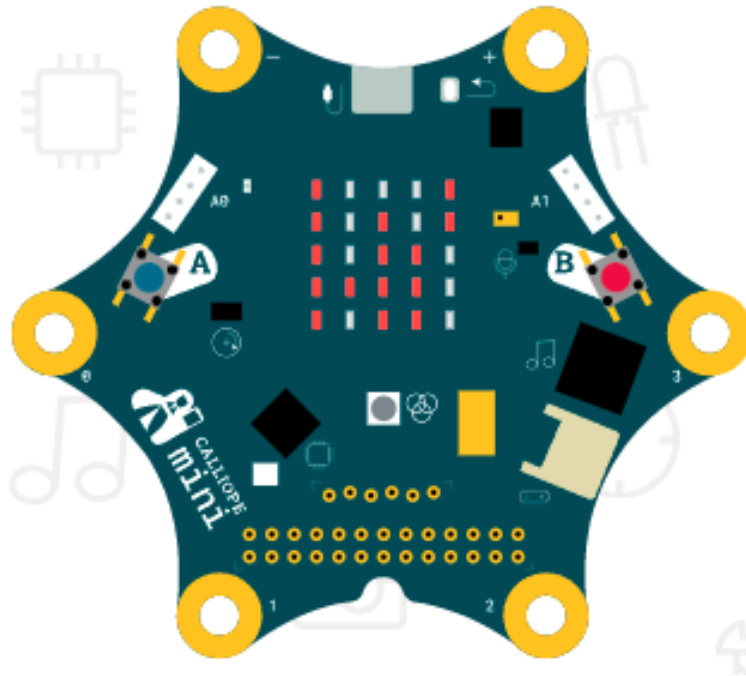
<https://www.hackster.io/thorsten-kimmeskamp/computerbildschirm-als-calliope-mini-display-74b85c>

TESTEN IM SIMULATOR

- ❖ sehr nützlich, solange simulierte Sensordaten reichen!



ALTERNATIVE: MEHRSTELLIGE ZAHLEN SCHNELL ANZEIGEN MIT DER „ABAKUS-METHODE“



5,437

<https://github.com/kimmeskamp/pxt-abakus>