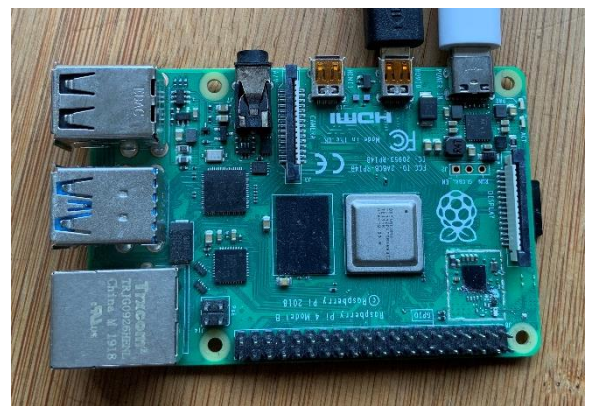


## Messung von Temperatur, Luftfeuchte und Luftdruck mit einem Raspberry PI, einem Sensor BME280 und der Programmierumgebung Node-Red

Für eine Messung der Klimadaten und Datenvisualisierung wird folgende Hardware benötigt:

Raspberry Pi 3 oder 4<sup>1</sup> mit Netzteil, Tastatur, Maus und HDMI-Monitor,



ein Sensor BME280 auf einer BreakOut-Platine

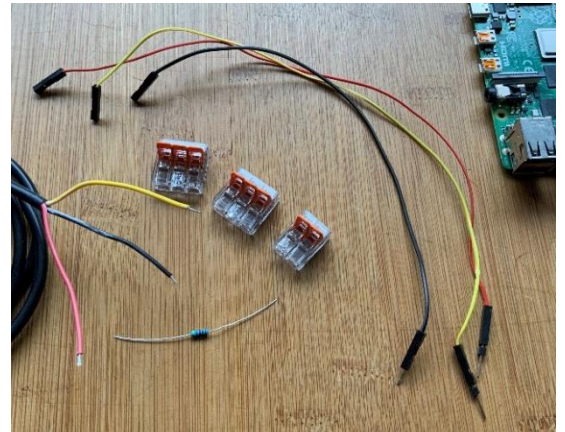


Abgebildete Version: <https://learn.adafruit.com/adafruit-bme280-humidity-barometric-pressure-temperature-sensor-breakout/pinouts>

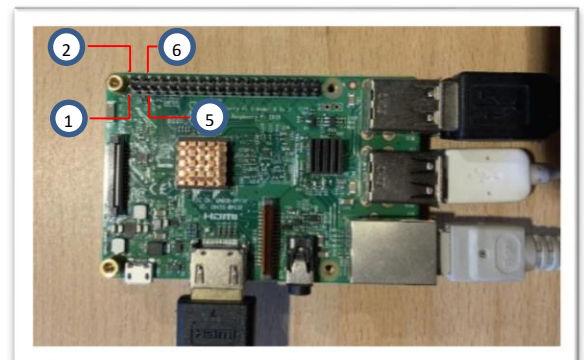
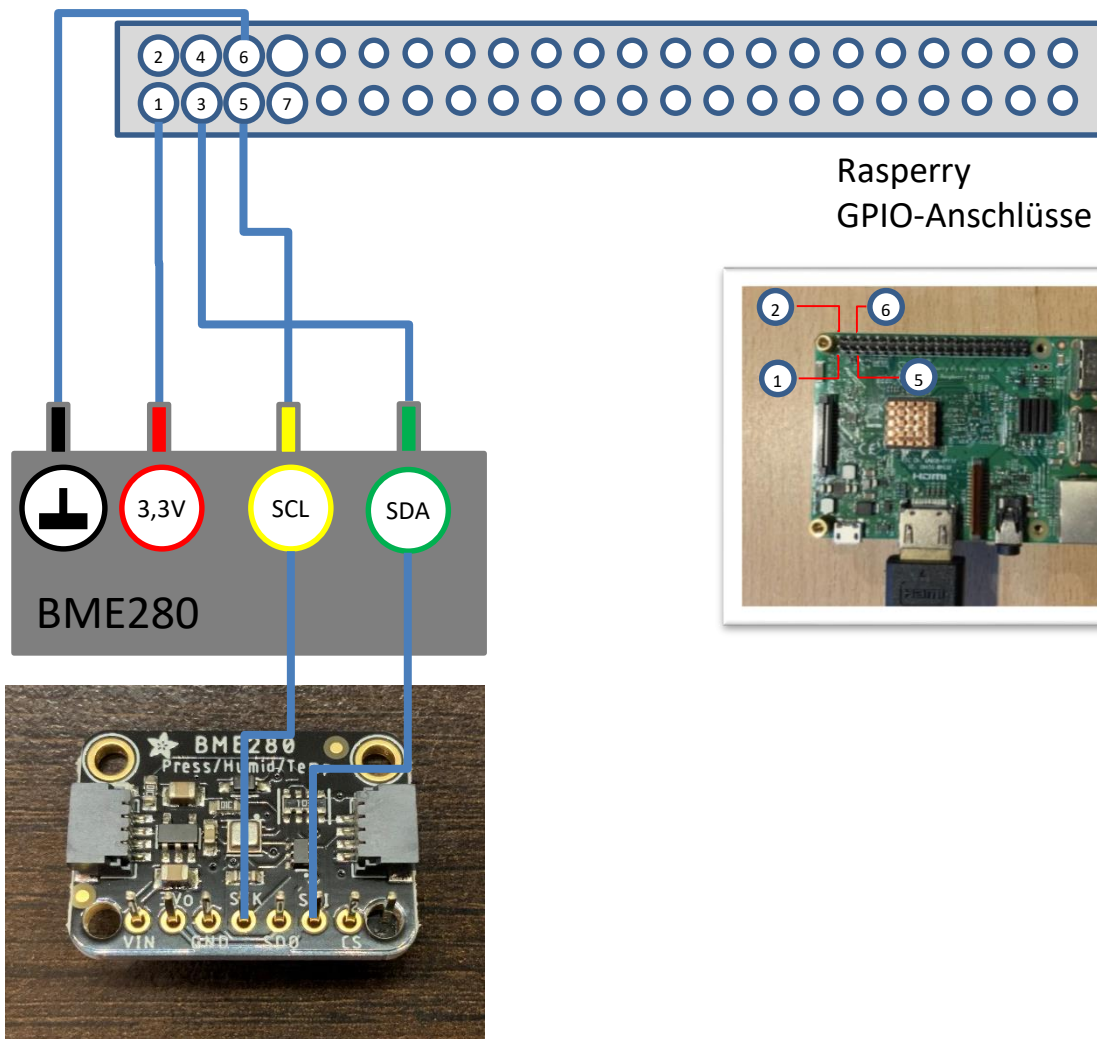
---

<sup>1</sup> Wie Sie ein Betriebssystem auf Ihrem Raspberry installieren können, zeigt die Anleitung „Installation eines Betriebssystems auf einem Raspberry PI“

3 Wagoklemmen oder ein Steckbrett/ Breadboard und 4 Verbindungskabel mit der Möglichkeit, eine Steckverbindung zwischen Raspberry und Wagoklemmen herzustellen.

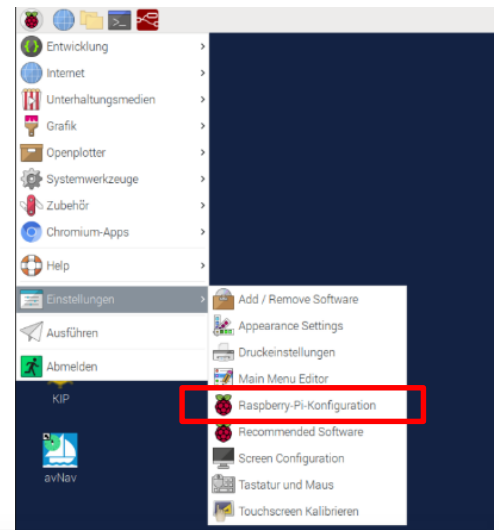


Die Hardware wird folgendermaßen miteinander verbunden, die Kabelfarben zeigen die Verwendung des jeweiligen Anschlusses:



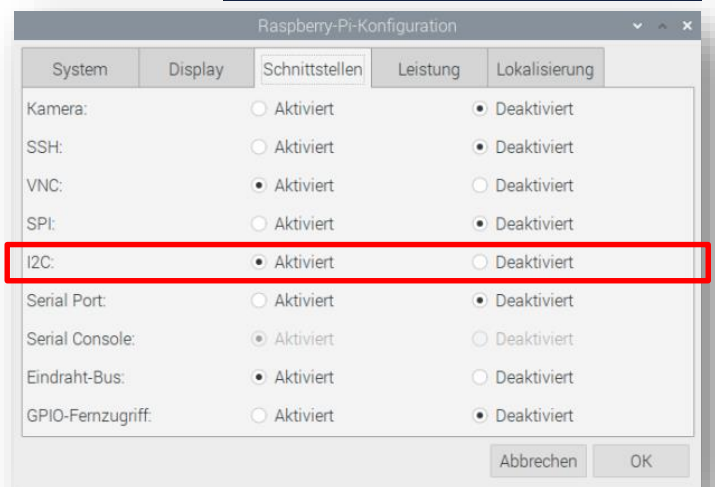
Bevor der Sensor vom Raspberry erkannt wird, muss noch eine wichtige Einstellung überprüft werden. Der Sensor BME280 wird über einen sogenannten **I2C-Bus** mit dem Raspberry verbunden.

Diese Schnittstelle (Bus) muss zuvor eingeschaltet werden. Das funktioniert über die Einstellungen des Raspberries:



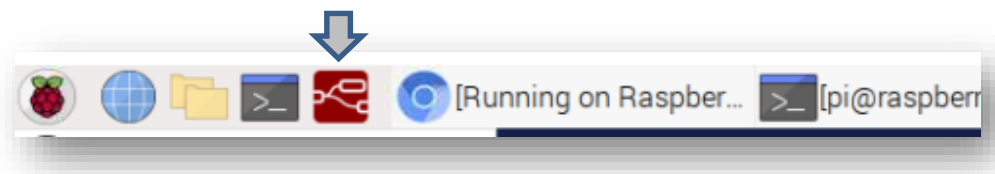
Im Menü „Schnittstellen“ wird dann der „I2C-Bus“ eingeschaltet.

Bei der Gelegenheit kann man auch gleich den Eindraht-Bus einschalten, da über diesen Bus eine Reihe ähnlicher Sensoren angeschlossen werden, z.B. der Temperatursensor DS18B20.



Wenn die Programmierumgebung „Node-Red“ bereits installiert und eingerichtet ist<sup>2</sup>, kann nun überprüft werden, ob der angeschlossene Sensor erkannt und betriebsbereit ist.

Starten Sie den Node-Red-Dienst über den Button in der Taskleiste:



Danach wird der Node-Red-Editor über diese Adresse im Browser aufgerufen:

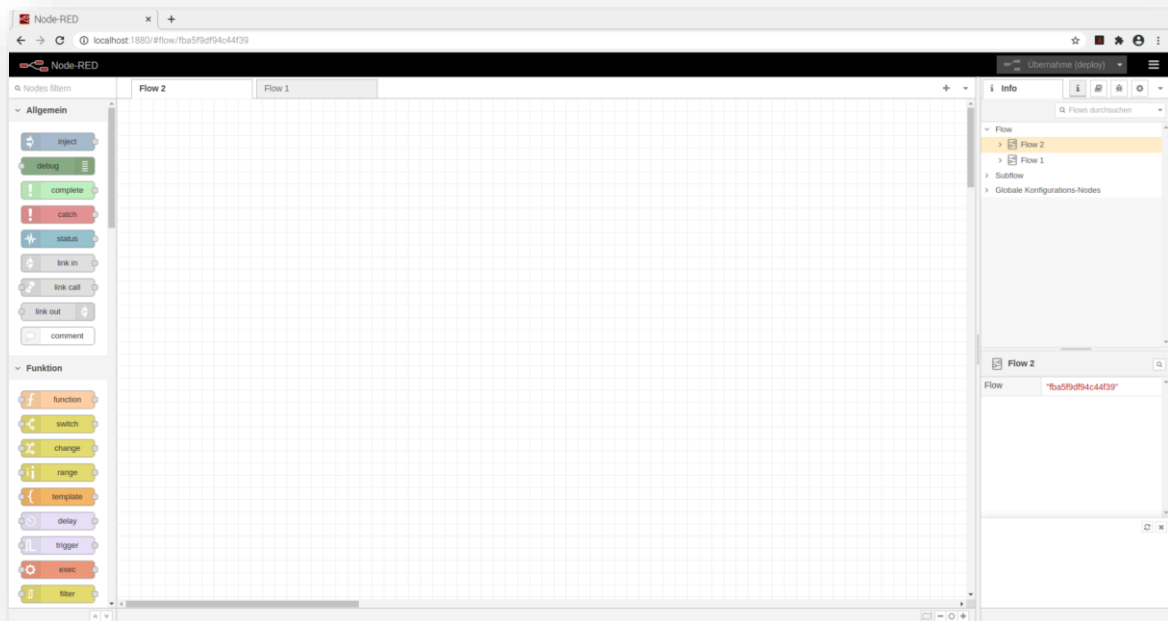
<http://localhost:1880>

<sup>2</sup> Wie Sie „Node-Red“ auf Ihrem Raspberry installieren können, zeigt die Anleitung „Installation von Node-Red auf einem Raspberry PI“

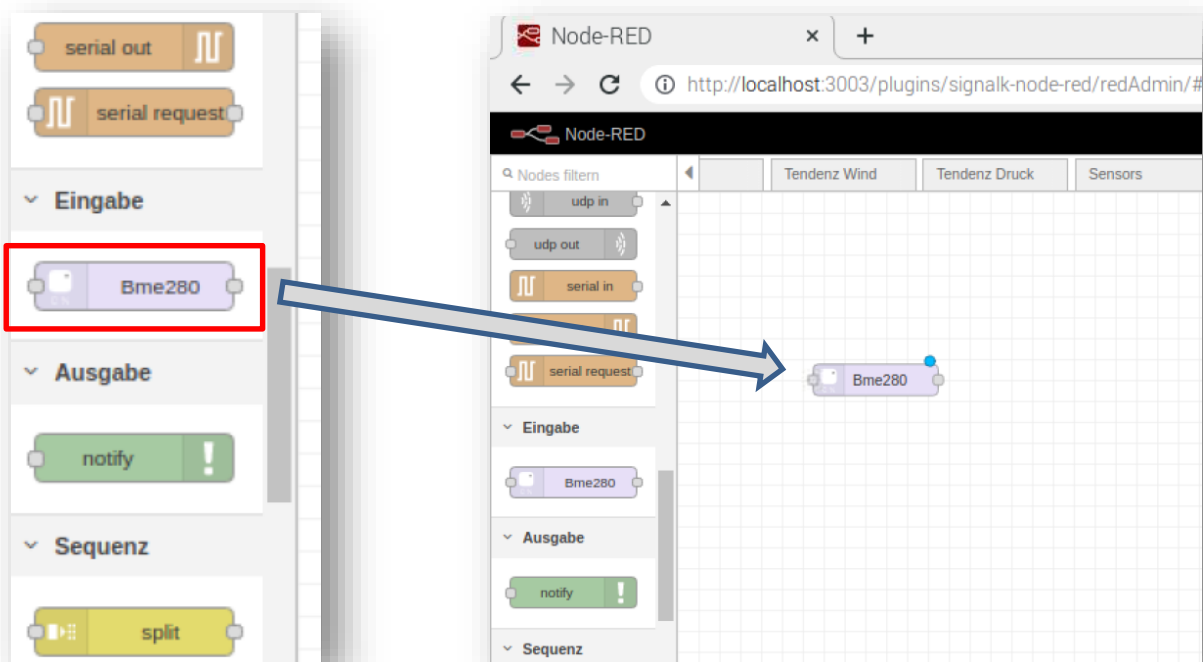
Sollte der Raspberry mit anderen browserfähigen Geräten im gleichen Netz arbeiten, lässt sich der Node-Red-Editor auch von dort über den Browser aufrufen, indem man die IP-Adresse des Rasperrys in die Adresszeile eingibt, gefolgt vom Port 1880:

<http://<ip-adresse raspberry>:1880>

Der Node-Red-Editor startet mit einem leeren „Flow“:



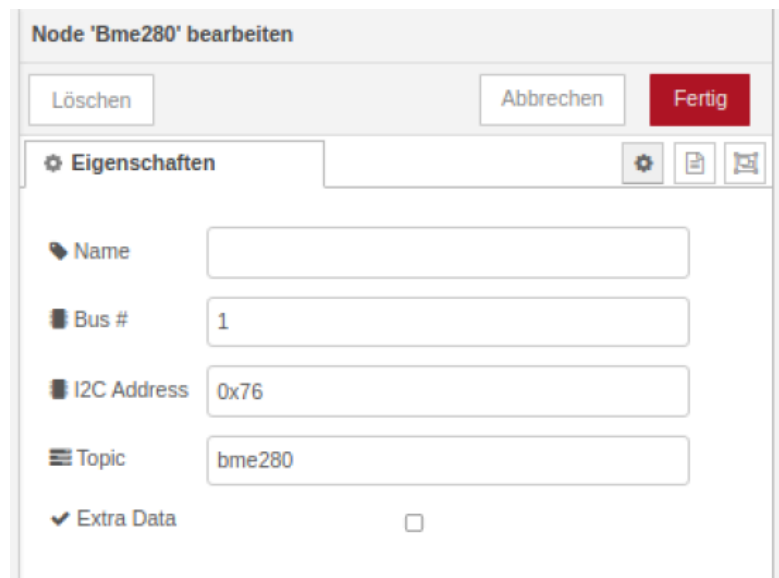
Suchen Sie sich nun im linken Bereich der „Nodes“ denjenigen Node heraus, der für einen BME280 geeignet ist, und ziehen Sie ihn in das leere Flow-Fenster.



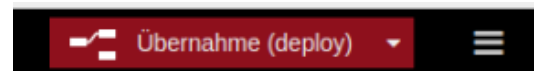
Der Node zeigt zwei kleine grafische Elemente:

- ▲ Das rote Dreieck steht dafür, dass der Node noch nicht fertig eingerichtet ist oder seine Konfiguration Fehler enthält.
- Der blaue Kreis steht dafür, dass die neuen Elemente im Flow noch nicht „übersetzt“ und überprüft wurden. Dies geschieht immer mit der Schaltfläche „**Übernahme**“ (**Deploy**)

Mit Doppelklick auf den Node für den BME2800 sollte seine Konfiguration überprüft werden. Vor allem muss im Fenster für die I2C-Adresse ein richtiger Wert auftauchen, in der regel ist das die hexadezimale Adresse 0x76.

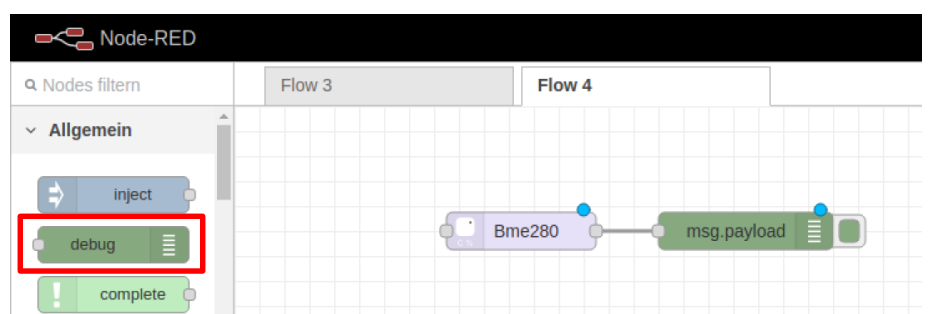


Nach Klick auf „Fertig“ sieht man nach wie vor den rot gefärbten „Übernahme“-Button. Diesen bitte anklicken, dann ist der Node für den Sensor BME280 eingerichtet und läuft.

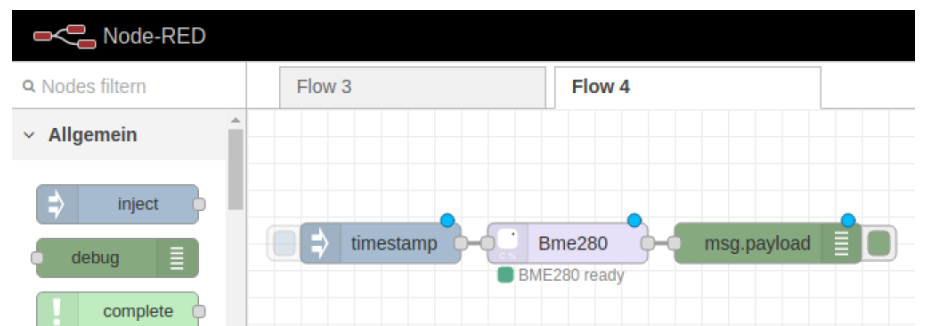


Nun wollen wir heraus bekommen, welche Werte der Sensor liefert.

Das geht am schnellsten, indem man einen „Debug“-Node mit dem Sensor-Node verbindet.



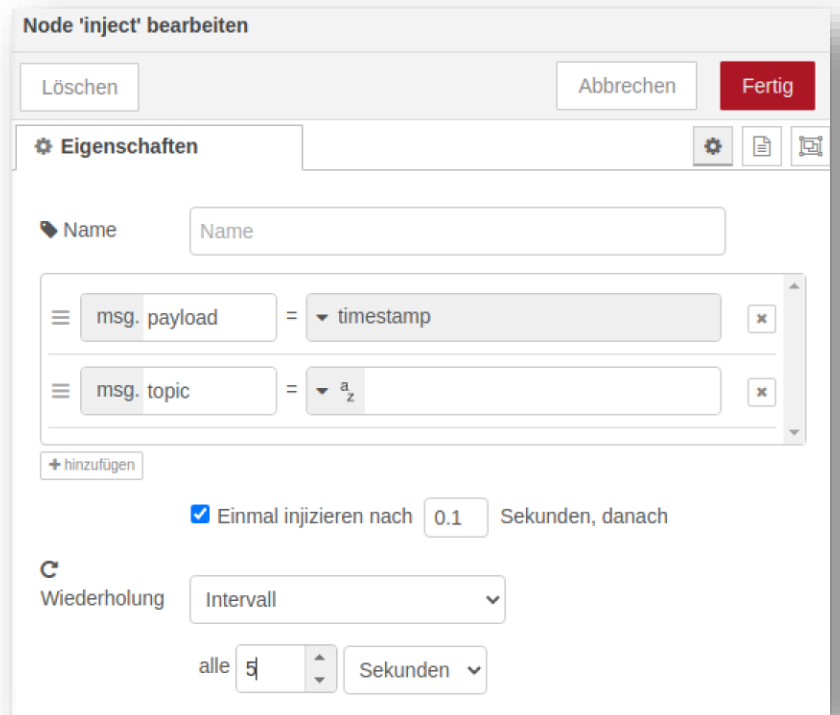
Zusätzlich wird ein „Inject“-Node benötigt, um dem Sensor Messintervalle zuweisen.



Der Inject“-Node sollte folgen-  
dermaßen konfiguriert wer-  
den:

Der Haken bei „Einmal injizie-  
ren“ bedeutet, dass der nach-  
folgende Node direkt nach Auf-  
ruf einen Zeitstempel erhält.  
Wichtig ist, die Wiederholung  
dieser Injizierung richtig einzu-  
stellen, hier alle 5 Sekunden.

Nach Klick auf „Fertig“ ist der  
Inject-Node eingerichtet, nun  
sollte der BME280-Sensor alle  
5sec Messwerte liefern. Das  
kann man zunächst mit dem  
Debug-Node überprüfen.

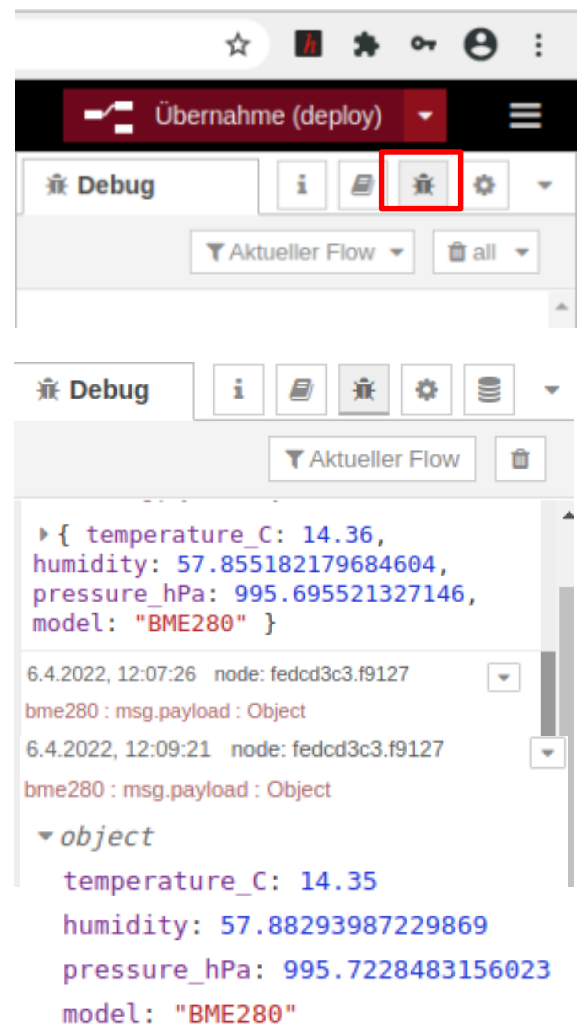


Dazu muss auf der rechten Seite des Editors  
noch die Sicht auf die Debug-Ausgabe eingerich-  
tet werden:

Das erledigt ein Klick auf den Schalter mit dem  
„Käfer“-Symbol.

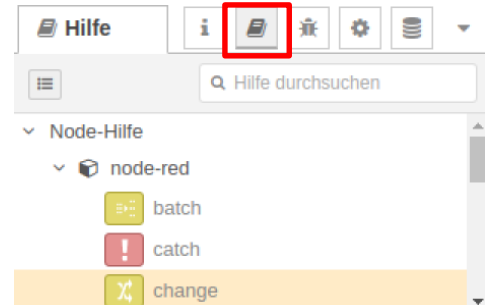
Und nun sollten die Werte des Sensors perio-  
disch im Debug-Fenster auftauchen. Die Daten  
kommen „verpackt“ im Payload-Objekt. Dieses  
Objekt ist das Standard-Objekt zur Übermittlung  
von Daten zwischen Nodes.

Für eine grafische Darstellung im Dashboard  
müssten die einzelnen Werte des „Payload-Ob-  
jektes“ auch spezifisch ansprechbar sein. In der  
Debug-Darstellung sieht man, dass die Werte für  
die Temperatur, die Luftfeuchte und den Luft-  
druck als Teil des Payload-Objektes vom Sensor  
angeliefert werden.



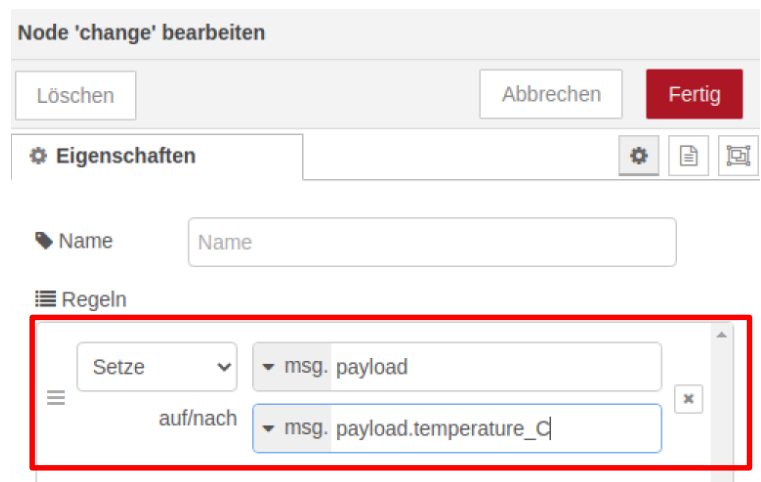
Um die angelieferten Werte aus dem Objekt „herauszulösen“, bietet sich der „Change“-Node an:

Um mehr über die Möglichkeiten eines Nodes zu erfahren, klickt man diesen bei Bedarf an und kann über das „Hilfe“-Symbol im rechten Bereich des Editors entsprechendes nachlesen.



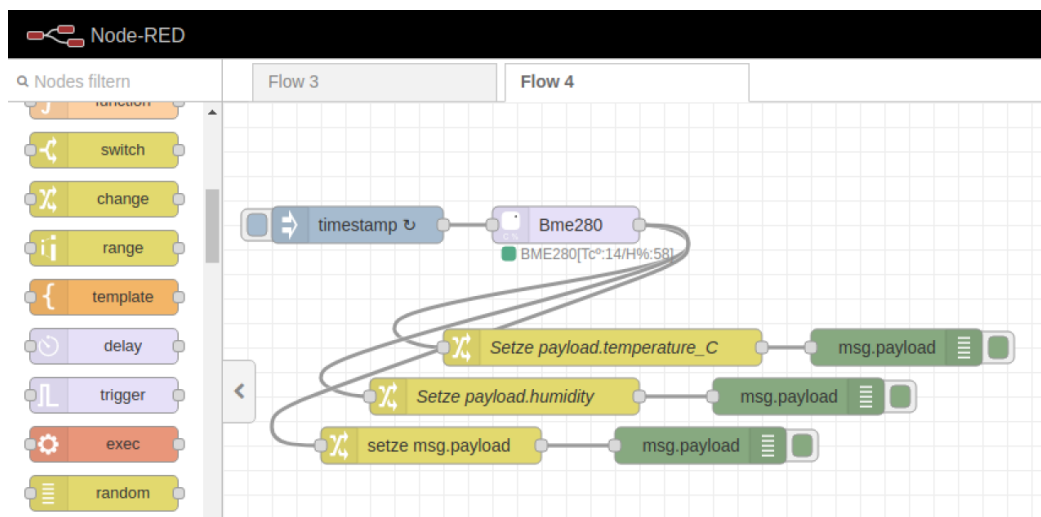
Wir nutzen den Change-Node nun, um die Ausgabe des Payload-Objektes „umzuleiten“ auf die jeweiligen Objekt-Eigenschaften, wie sie im Debug-Fenster bereits gezeigt werden.

Dazu verändern wir die Ausgabe vom gesamten Payload-Objekt auf die jeweils gewünschte untere Objekt-Ebene, hier ***\*.temperature\_C***, um die Temperatur in Celsius zu erhalten.



An die Ausgabe des Sensors binden wir so 3 Change-Nodes an, die dann jeweils nur den gewünschten Wert liefern:

1. ***msg.payload >> msg.payload.temperature\_C***
2. ***msg.payload >> msg.payload.humidity***
3. ***msg.payload >> msg.payload.pressure\_hPa***



Nachdem 3 Debug-Nodes an die jeweiligen Change-Nodes gebunden worden sind, sollten die entsprechenden Werte auf jedem Debug-Node einzeln auflaufen:

Temperatur in Celsius: \_\_\_\_\_ →

Luftfeuchte in % \_\_\_\_\_ →

Luftdruck in Hekto-Pascal \_\_\_\_\_ →

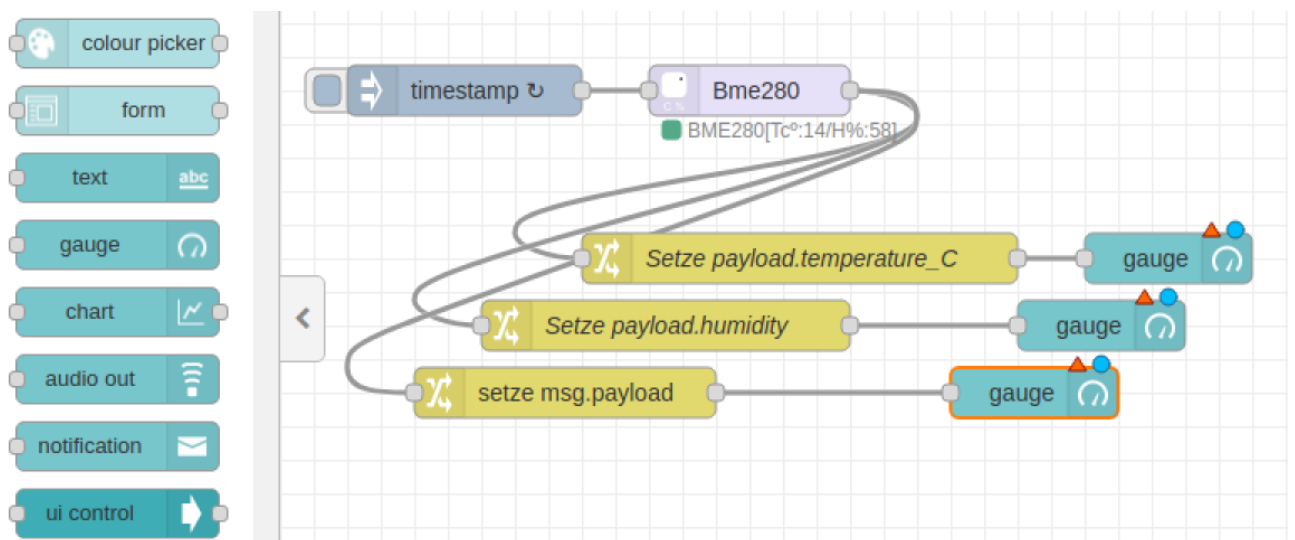


Fehlt nur noch die Einrichtung einer grafisch ansprechenden Ausgabe dieser Temperaturdaten:

Dazu nutzen wir einen Node aus dem Bereich „Dashboard“, den „Gauge“-Node, und ziehen ihn wie gewohnt in das Flow-Fenster:



Das Flow-Fenster sollte dann in etwa so aussehen:

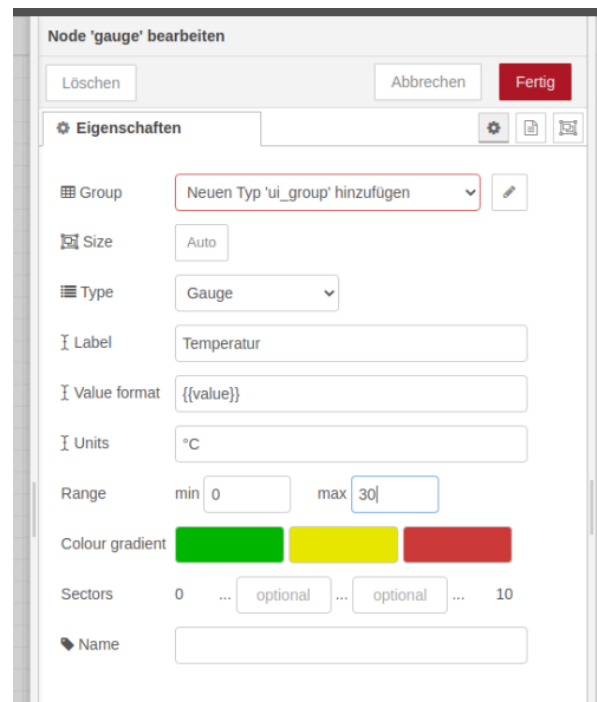




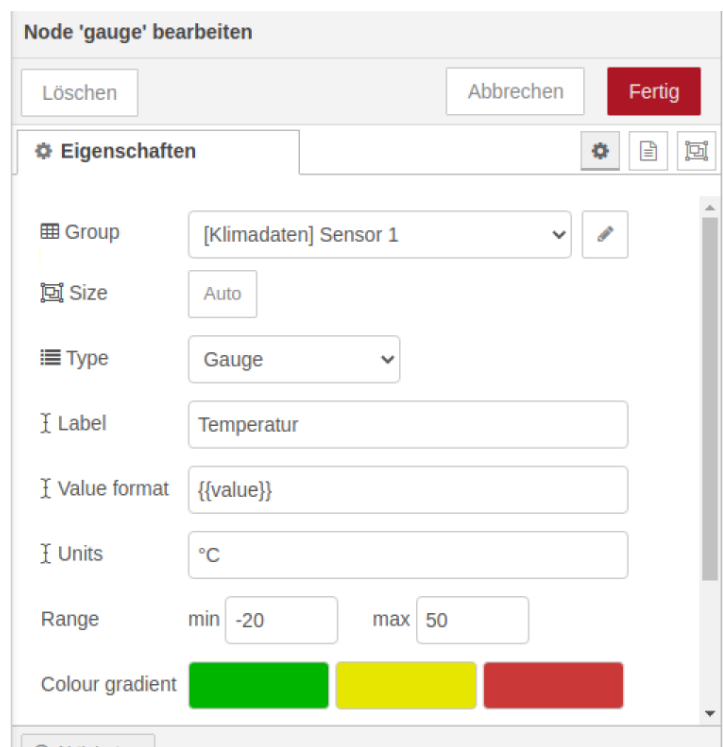
Nun muss nur noch eine Verbindung zwischen dem Sensor-Node und den Gauge-Nodes hergestellt werden. Die Gauge-Nodes sollten auch konfiguriert werden. Dazu bitte den jeweiligen Node doppelklicken.

Überall dort, wo sich rote Rahmen finden, müssen noch Eintragungen vorgenommen werden.


Zusätzlich kann die grafische Anzeige mit den hier gezeigten Eintragungen angepasst werden.

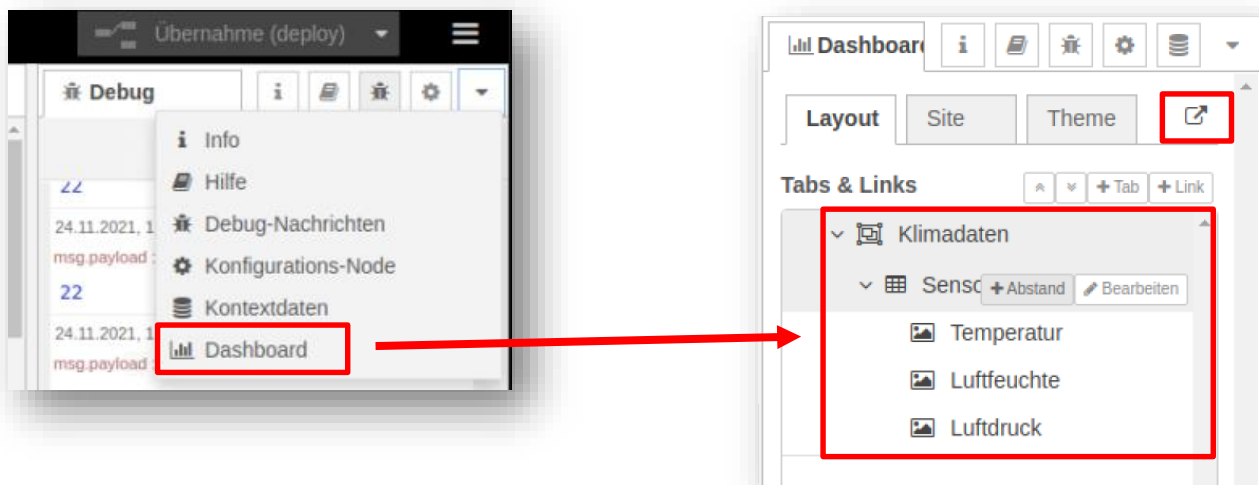


Am Ende sollten die „Gauge“-Nodes für die Sensorwerte des BME280 folgendermaßen eingerichtet sein:

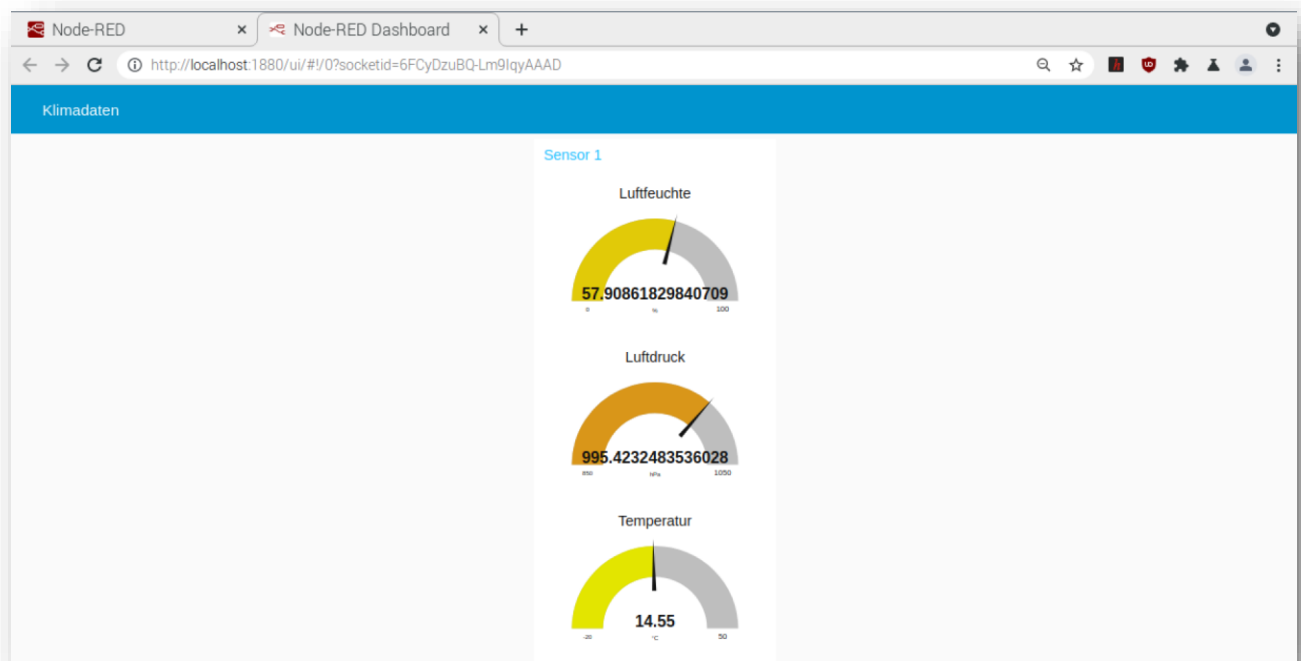


Klicken Sie auf „Fertig“, um die jeweilige Einrichtung der „Gauge“-Nodes zu beenden.

Nun kann die Anzeige des Dashboards ausgelöst werden, indem man im rechten Bereich des Editors die Dashboard-Anzeige auswählt und über die Schaltfläche  das Dashboard startet.



Und so sieht dann das Ergebnis aus:



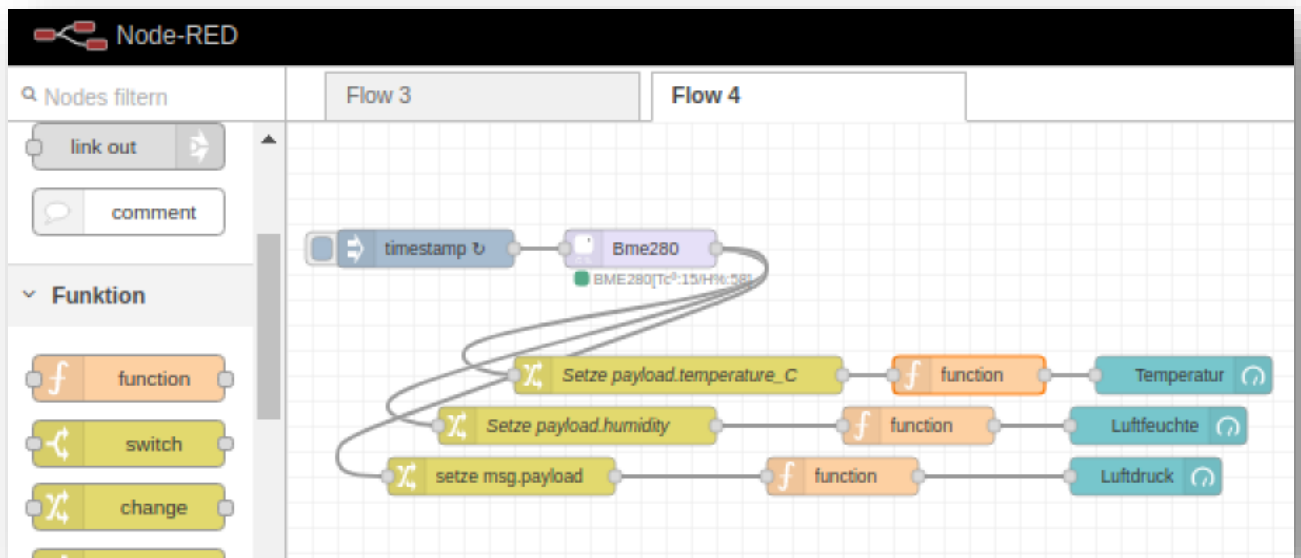
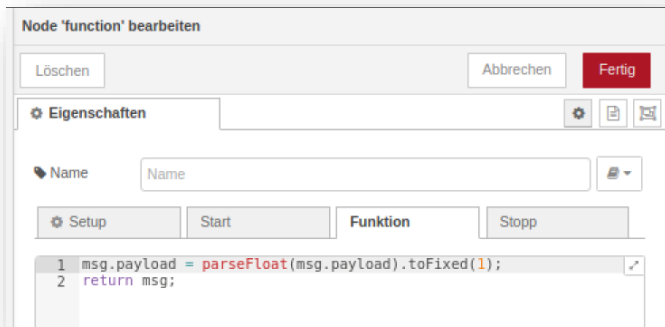
... wenn alles gut gelaufen ist.

Übrigens eignen sich „function“-Nodes, um zum Beispiel die Nachkommastellen in der Dashboard-Ansicht zu minimieren. In „function“-Nodes kann Java-Script-Code zur Bearbeitung der eingehenden Daten eingegeben werden:

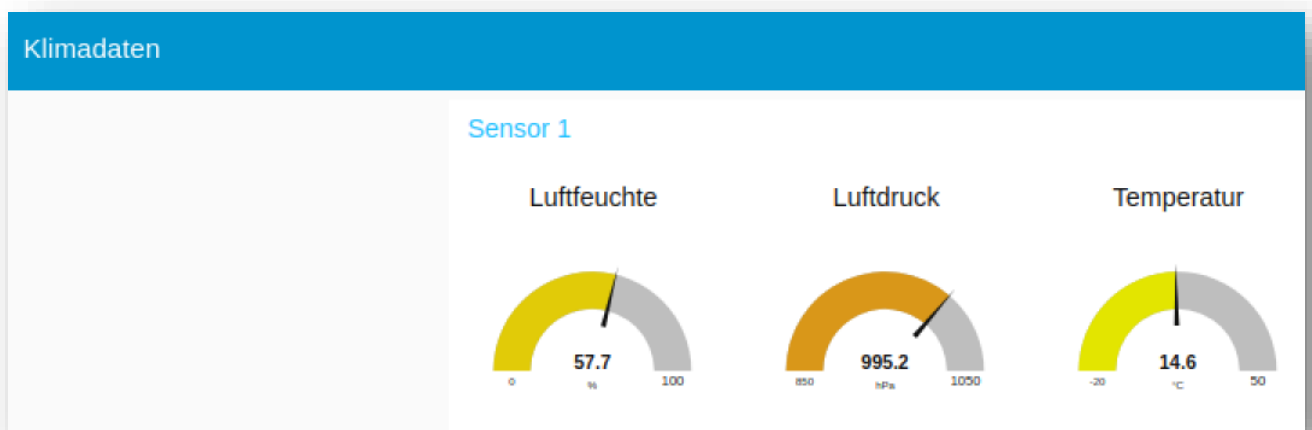
Die dazu notwendige Funktion lautet:

**`msg.payload = parseFloat(msg.payload).toFixed(1);`**

Der Wert in der Klammer hinten entspricht den gewünschten Nachkommastellen.



So in den Flow eingebaut ergibt sich ein neues Bild im Dashboard:



Links:

Video „Node-Red für Sportbootfahrer“:

<https://www.youtube.com/watch?v=QFMgAyTseM8>

Node-Red Node-Übersicht:

<https://nodered.org/>

<https://nodered.org/docs/user-guide/nodes>

Das gleiche Projekt mit „klassischer“ Programmierung:

<https://klenzel.de/5335>

<https://test-wetterstation.de/temperaturmessung-mit-dem-raspberry-pi>