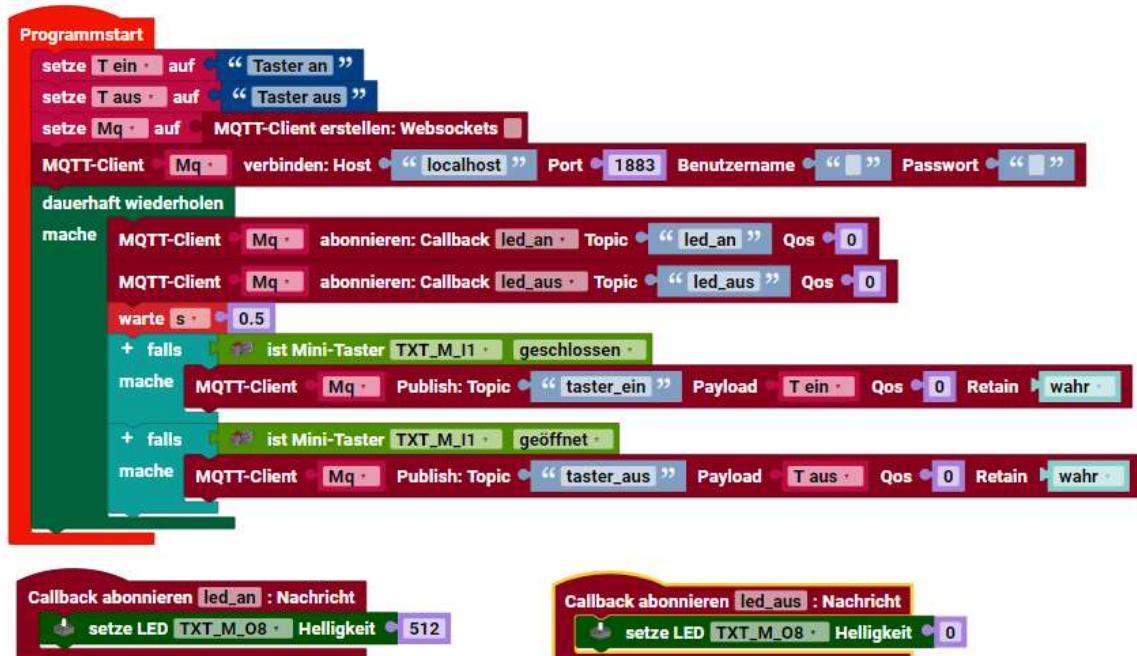
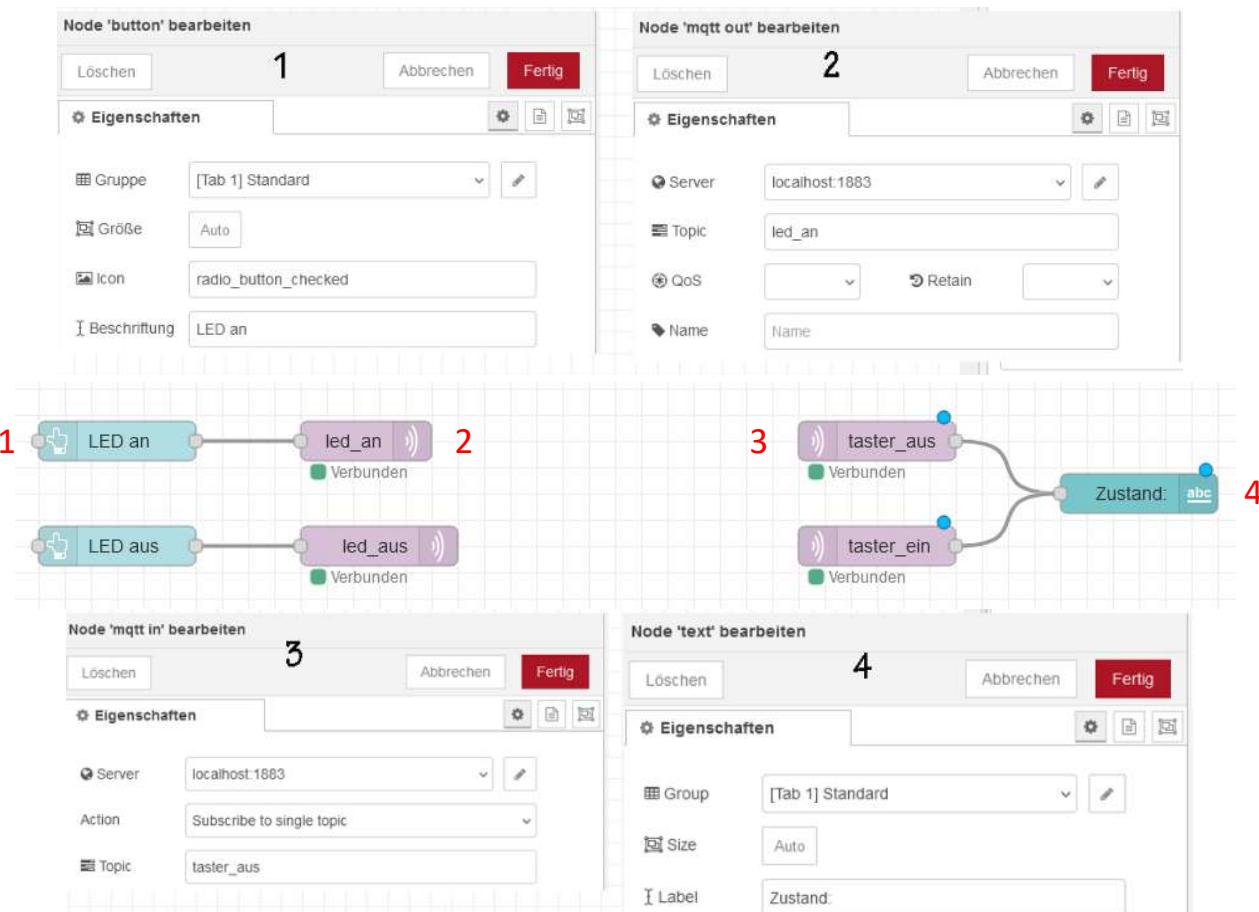


Beispiel zur Auswertung eines Tasters und dem schalten einer Lampe

Programm unter RPC



Flow unter Node-RED

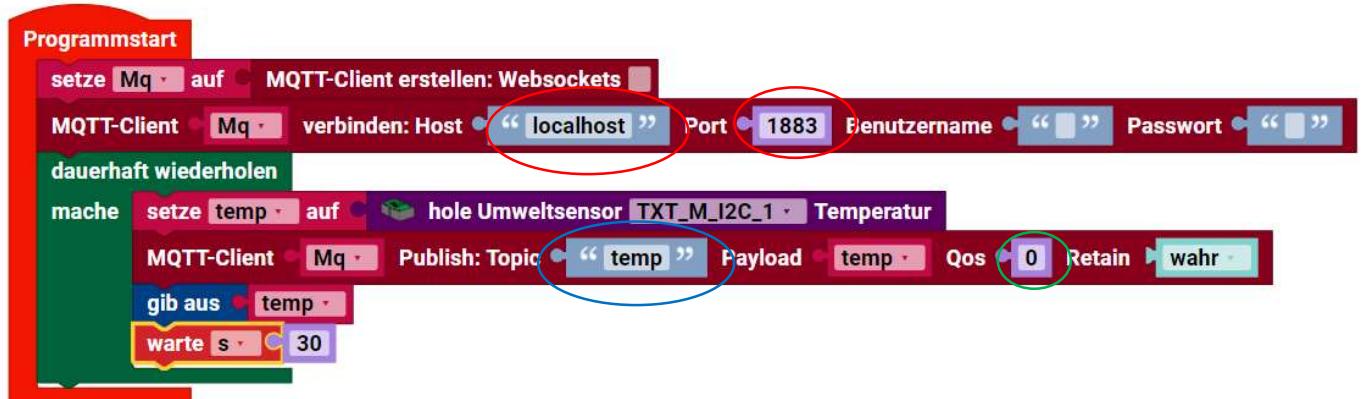


Dashboard

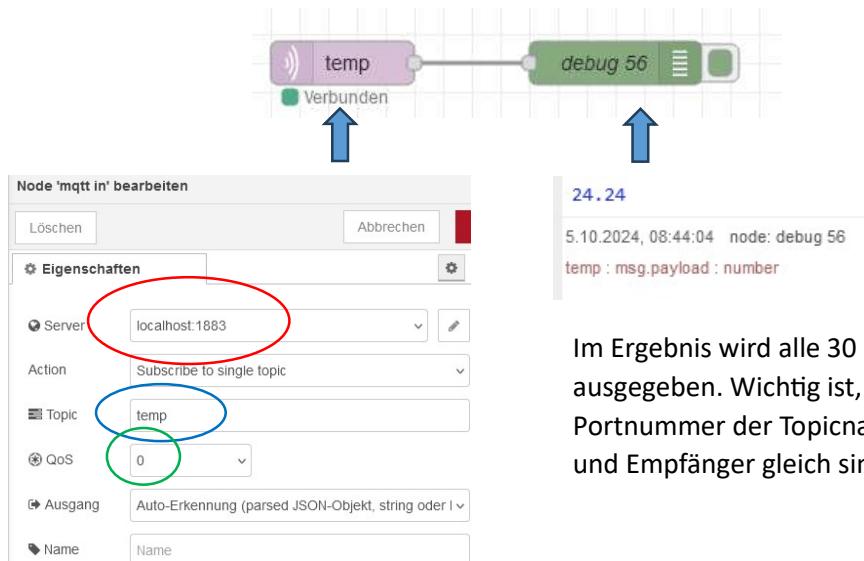


Beispiel zum Auslesen des Umweltsensors vom TXT 4.0

Programm unter RPC

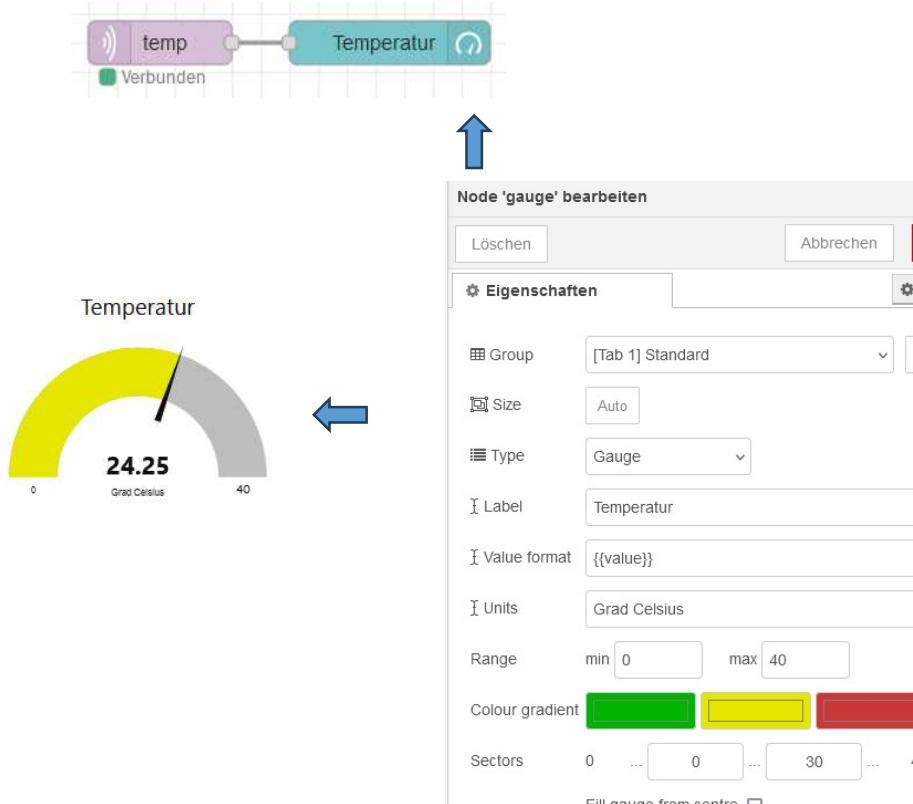


Flow unter Node-RED



Im Ergebnis wird alle 30 Sekunden die Temperatur ausgegeben. Wichtig ist, dass der Name des Servers, die Portnummer der Topicname und die QoS bei Sender und Empfänger gleich sind.

Zur besseren Darstellung kann nun auch die Gauge im Dashboard eingesetzt werden.



Beispiel zu Motorsteuerung vom Dashboard

Programm unter RPC

The screenshot shows two panels. The top panel, titled "Programmstart", contains the following code:

```

setze Mq auf MQTT-Client erstellen: Websockets
MQTT-Client Mq verbinden: Host "localhost" Port 1883 Benutzername " " Passwort " "
dauerhaft wiederholen
mache MQTT-Client Mq abonnieren: Callback speed Topic "speed" QoS 0
warte s 0.5

```

The bottom panel, titled "Callback abonnieren speed : Nachricht", contains the following code:

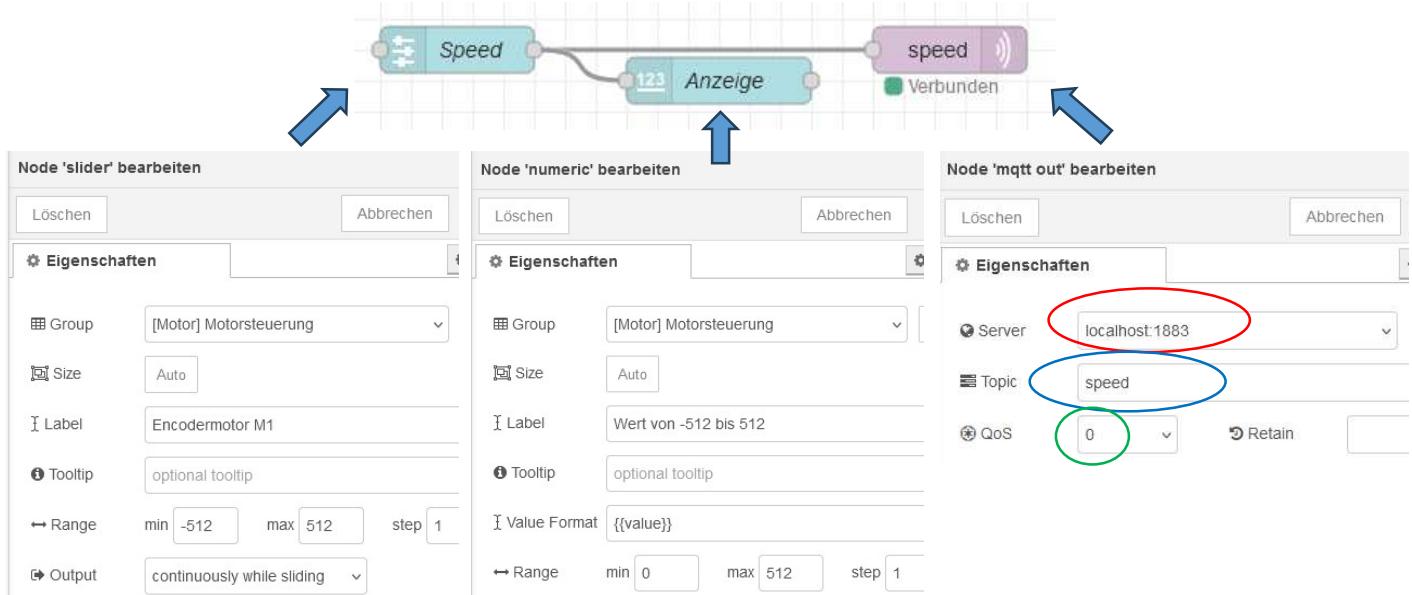
```

setze V auf Nachricht payload
gib aus V
+ setze Motor TXT_M_M1 links Geschwindigkeit V
  Schrittweite 100

```

To the right, a "Kontrollausgabe der Variable V" section shows the value of variable V changing over time, with values like -14, -7, -14, -20, -27, -33, -40, and -47.

Flow unter Node-RED



Ansicht im Dashboard

The screenshot shows the Node-RED dashboard with the following components:

- Dashboard**: The main navigation bar.
- Layout**: A tab for layout settings.
- Tabs & Links**: A sidebar with tabs for "Motor" and "Motorsteuerung".
- Node-RED Dashboard**:
 - Tab 1**: Contains a table example and a "Motor" node.
 - Motorsteuerung**: A tab containing a slider for "Encodermotor M1" with a range from -512 to 512, currently set to -47.
 - Homepage Axel**: Another tab.

Im Ergebnis kann der Motor über den Slicer vorwärts und rückwärts geregelt werden.