

Neuinstallation ioBroker auf einem R-Pi 3 B+

Quellen:

- [Raspberry Pi OS und ggf. Image Writer](#)
- [Raspi-Config Kommandozeilen Optionen](#)
- [ioBroker](#)
- [InfluxDB Installation](#)
- [InfluxDB 1.x Installation](#)
- [Grafana Installation](#)
- [Grafana ohne Login](#)
- [Verbindung InfluxDB 2.0 <-> Grafana](#)
- [Jeelink Klon](#)
- [LaCrosse Temperatur- & Feuchtesensor TX29DTH-IT, 868 MHz](#)
- [FHEM Wiki zu JeeLink/Lacrosse Klon im Eigenbau](#)
- [Arduino Sketch für JeeLink Empfänger](#)
- [Verdrahtung Arduino Nano und 868 MHz RF-Modul](#)

Das Grundsystems

Installation

Die Verwendung von InfluxDB 2.x setzt ein 64-bit Betriebssystem voraus.

Ein Desktop bzw. eine vollständige grafische Oberfläche wird nicht benötigt. Daher bezeichnet man ein solches System als „Headless“. Somit basiert dieses Heimautomatisierungsprojekt auf einem Raspberry Pi OS (32 oder 64-bit) Lite.

```
# Raspberry Pi Image besorgen
#wget
https://downloads.raspberrypi.org/raspios_lite_armhf/images/raspios_lite_armhf-2023-02-22/2023-02-21-raspios-bullseye-armhf-lite.img.xz
wget
https://downloads.raspberrypi.com/raspios_lite_armhf/images/raspios_lite_armhf-2023-10-10/2023-10-10-raspios-bookworm-armhf-lite.img.xz
oder
#wget
https://downloads.raspberrypi.org/raspios_lite_arm64/images/raspios_lite_arm64-2023-02-22/2023-02-21-raspios-bullseye-arm64-lite.img.xz
wget
https://downloads.raspberrypi.com/raspios_lite_arm64/images/raspios_lite_arm64-2023-10-10/2023-10-10-raspios-bookworm-arm64-lite.img.xz
# R-Pi Imager herunterladen und intallieren:
wget https://downloads.raspberrypi.org/imager/imager_latest_amd64.deb
sudo dpkg -i imager_latest_amd64.deb
rpi-imager
#sudo touch /media/<benutzername>/bootfs/ssh
```

```
#sync
# SD-Karte aushängen und damit den R-Pi booten
```

Konfiguration

Während des Bootvorganges erhält der Raspberry eine IP Adresse vom Router. Diese kann über die entsprechende Weboberfläche des Routers ermittelt werden.

```
# Grundsystem auf den aktuellen Stand bringen:
ssh -l pi <IP>
sudo su
apt-get update && sudo apt-get upgrade
apt-get install aptitude mc apt-transport-https software-properties-common
raspi-config nonint do_hostname iobroker
raspi-config nonint do_onewire 0
raspi-config nonint do_change_locale de_DE.UTF-8 UTF-8
raspi-config nonint do_expand_rootfs
raspi-config nonint do_update
reboot
# Feste IP für eth0 einstellen:
mcedit /etc/dhcpd.conf
interface eth0
static ip_address=192.168.10.11/24
static routers=192.168.10.1
static domain_name_servers=192.168.10.1
# nicht benötigte Dienste beenden und/oder entfernen:
systemctl disable ModemManager.service
systemctl stop ModemManager.service
apt-get purge modemmanager
```

InfluxDB 1.x

Quelle: <https://docs.influxdata.com/influxdb/v1/introduction/install/>

Installation

```
sudo su
#wget -q0- https://repos.influxdata.com/influxdata-archive_compat.key | sudo
apt-key add -
#source /etc/lsb-release
#echo "deb https://repos.influxdata.com/${DISTRIB_ID,,} ${DISTRIB_CODENAME}
stable" | sudo tee /etc/apt/sources.list.d/influxdb.list
#curl https://repos.influxdata.com/influxdata-archive.key | gpg --dearmor |
sudo tee /usr/share/keyrings/influxdb-archive-keyring.gpg >/dev/null
#echo "deb [signed-by=/usr/share/keyrings/influxdb-archive-keyring.gpg]
https://repos.influxdata.com/debian $(lsb_release -cs) stable" | sudo tee
```

```
/etc/apt/sources.list.d/influxdb.list
wget -q https://repos.influxdata.com/influxdata-archive_compat.key
echo '393e8779c89ac8d958f81f942f9ad7fb82a25e133faddaf92e15b16e6ac9ce4c
influxdata-archive_compat.key' | sha256sum -c && cat influxdata-
archive_compat.key | gpg --dearmor | sudo tee
/etc/apt/trusted.gpg.d/influxdata-archive_compat.gpg > /dev/null
echo 'deb [signed-by=/etc/apt/trusted.gpg.d/influxdata-archive_compat.gpg]
https://repos.influxdata.com/debian stable main' | sudo tee
/etc/apt/sources.list.d/influxdata.list
```

```
apt-get update
```

```
apt-get install influxdb
```

```
systemctl unmask influxdb.service
```

```
systemctl enable influxdb
```

```
systemctl start influxdb
```

```
systemctl status influxdb
```

Konfiguration

```
sudo su
```

```
influx
```

```
> CREATE USER "admin" WITH PASSWORD 'influxdbadmin' WITH ALL PRIVILEGES
```

```
> CREATE USER "iobroker" WITH PASSWORD 'iobroker'
```

```
> CREATE DATABASE "iobroker"
```

```
> GRANT ALL ON "iobroker" TO "iobroker"
```

```
> exit
```

```
#
```

```
mcedit /etc/influxdb/influxdb.conf
```

```
[http]
```

```
enabled = true
```

```
bind-address = ":8086"
```

```
auth-enabled = true
```

```
log-enabled = true
```

```
write-tracing = false
```

```
pprof-enabled = true
```

```
https-enabled = false
```

```
#
```

```
systemctl restart influxdb
```

Datenbankgröße ermitteln

```
sudo su
```

```
du -sh /var/lib/influxdb/data/iobrokerdb/
```

Backups

Quelle:

https://docs.influxdata.com/influxdb/v1/administration/backup_and_restore/#back-up-all-databases

Erstellen

```
sudo su
influxd backup -portable /path/to/backup-destination
```

Wieder herstellen

```
sudo su
influxd restore -portable /path/to/backup-destination
```

InfluxDB 2.x

Installation

```
sudo su
cd ~
wget -q https://repos.influxdata.com/influxdata-archive_compat.key
echo '393e8779c89ac8d958f81f942f9ad7fb82a25e133faddaf92e15b16e6ac9ce4c
influxdata-archive_compat.key' | sha256sum -c && cat influxdata-
archive_compat.key | gpg --dearmor | sudo tee
/etc/apt/trusted.gpg.d/influxdata-archive_compat.gpg > /dev/null
echo 'deb [signed-by=/etc/apt/trusted.gpg.d/influxdata-archive_compat.gpg]
https://repos.influxdata.com/debian stable main' | sudo tee
/etc/apt/sources.list.d/influxdata.list
apt-get update
apt-get install influxdb2
systemctl start influxdb
systemctl status influxdb
```

Konfiguration

Die Konfiguration der InfluxDB erfolgt über den Browser via <http://RASPI-IP:8086>.

Grafana

Installation

```
sudo su
```

```
#wget -q -O /usr/share/keyrings/grafana.key https://apt.grafana.com/gpg.key
#echo "deb [signed-by=/usr/share/keyrings/grafana.key]
https://apt.grafana.com stable main" | sudo tee -a
/etc/apt/sources.list.d/grafana.list

wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
echo "deb https://packages.grafana.com/oss/deb stable main" | sudo tee -a
/etc/apt/sources.list.d/grafana.list
apt-get update
# Install the latest OSS release:
apt-get install -y grafana
# Start System Service aka. Server
systemctl daemon-reload
systemctl start grafana-server
systemctl status grafana-server
systemctl enable grafana-server.service
```

Grafana Server: <http://IP:3000>

Grafana ohne Login

```
mcedit /etc/grafana/grafana.ini
# [auth.anonymous]
enabled = true
systemctl restart grafana-server
```

Backup

Quelle: <http://cactusprojects.com/backup-restore-grafana/>

Erstellen

```
sudo su
cp /var/lib/grafana/grafana.db /path/to/backup-destination
cp /etc/grafana/grafana.ini /path/to/backup-destination
```

Wieder herstellen

```
sudo su
cd /path/to/backup-destination
cp grafana.db /var/lib/grafana/
cp grafana.ini /etc/grafana/
```

ioBroker

Installation

```
sudo su
# automatische Installation:
curl -sLf https://iobroker.net/install.sh | bash -
```

Nach der Installation ist die ioBroker Instanz erreichbar unter <http://<Raspberry-Pi-IP>:8081>.

Steuerung

```
# ioBroker starten:
iobroker start
# ioBroker stoppen:
iobroker stop
# ioBrker Infos anzeigen
iobroker info
```

Adapter installieren

Als Adapter werden alle Schnittstellen zu externen aber auch zu internen Datenquellen (z.B. Shelly Sensoren, Temperatur, Luftfeuchte, GPIO, 1-Wire, etc.) und Senken (Shelly Aktoren) bezeichnet.

- influxDB (Datenbank zur Speicherung von Messwerten)
- DS18B20 (Auslesen von 1-Wire Temperatursensoren)
- HABpanel (Steuerung von Aktoren vis Webseite)
- Jeelink-Geräte (Empfangen von 868mHz LaCrosse Temperatur- und Feuchtesensoren (hier z.B. TX29DTH-IT) mit Hilfe eines Jeelink Klones)
- MQTT Broker (zur Datenverarbeitung von MQTT-fähigen Sensoren und Aktoren)
- Philips Luftreiniger (für die Anbindung eines Philips Air Purifier 4000i Series (Typ: AC4236/10)
- RPI-Monitor (Nutzung von GPIOs z.B. zum Einlesen eines Reed-Kontaktes für den Gaszähler)
- Skriptausführung (Javascript und Blockly)
- WEB-Server (für HABpanel)

admin

Admin

Installierte Instanzen: 1

Verfügbare Version: 6.3.5

Installierte Version: 6.3.5

backupup

BackItUp

Installierte Instanzen: 1

Verfügbare Version: 2.5.12

Installierte Version: 2.5.12

influxdb

Daten mit InfluxDB protokollieren

Installierte Instanzen: 1

Verfügbare Version: 3.2.0

Installierte Version: 3.2.0

ds18b20

DS18B20

Installierte Instanzen: 1

Verfügbare Version: 1.6.1

Installierte Version: 1.6.1

simple-api

Einfache RESTful API

Verfügbare Version: 2.7.2

Installierte Version: 2.7.2

discovery

Gerätesuche

Installierte Instanzen: 1

Verfügbare Version: 3.1.0

Installierte Version: 3.1.0

habpanel

HABpanel

Installierte Instanzen: 1

Verfügbare Version: 0.5.0

Installierte Version: 0.5.0

jeelink

Jeelink-Geräte

Installierte Instanzen: 1

Verfügbare Version: 0.1.4

Installierte Version: 0.1.4

mqtt

MQTT Broker/Client

Installierte Instanzen: 1

Verfügbare Version: 4.0.7

Installierte Version: 4.0.7

philips-air

Philips Luftreiniger

Installierte Instanzen: 1

Verfügbare Version: 1.0.1

Installierte Version: 1.0.1

rpi2

RPI-Monitor

Installierte Instanzen: 1

Verfügbare Version: 1.3.2

Installierte Version: 1.3.2

javascript

Skriptausführung

Installierte Instanzen: 1

Verfügbare Version: 6.1.4

Installierte Version: 6.1.4

socketio

socket.io

Verfügbare Version: 4.2.0

Installierte Version: 4.2.0

ws

Web socket

Verfügbare Version: 1.3.0

Installierte Version: 1.3.0

web

WEB-Server

Installierte Instanzen: 1

Verfügbare Version: 4.3.0

Installierte Version: 4.3.0

Adapter konfigurieren

Die oben aufgeführten Adapter sind nach der Installation unter Instanzen zu finden und müssen dort konfiguriert werden.

1-Wire

DS18B20

Instanzeinstellungen: ds18b20.0

v2.0.4

Info

ALLGEMEINE EINSTELLUNGEN

SENSOREN

REMOTE-SYSTEME

Sensoren

Adresse	Remote-System-ID	Name	Abfrageintervall	Einheit	Faktor	Offset	Dezimalstellen	Null bei Fehler	Aktiviert	
10-00080359c712	keins	Vorlauftemperatur	Standard	°C	1	0	2	☒	☒	↓
10-00080359e7fe	keins	Rücklauftemperatur	Standard	°C	1	0	2	☒	☒	↑

JeeLink

TBC - Empfänger bauen und programmieren → TBC

Jeelink Settings

/dev/ttyUSB0

57600

Serial port

Baud rate

/dev/ttyUSB0 or /dev/ttyACME0

usually 57600

Jeelink Command (trial)

☐ Befehl aktivieren

usually empty

Befehl für Stick-Setup

usually empty

Sensorkonfiguration

gültige Sensortypen:
emonTH, emonWater, LaCrosseDTH, LaCrosseDTT, LaCrosseBMP180, HMS100T, LaCrosseWS, EC3000, EMT7110, level

+

GET FROM ADAPTER

	id	type	uid	Name	Löschen
1	22	LaCrosseDTH	▼ Badezimmer	THS_#03	
2	43	LaCrosseDTH	▼ Küche	THS_#05	
3	21	LaCrosseDTH	▼ Wohnzimmer	THS_#06	
4	11	LaCrosseDTH	▼ Wintergarten	THS_#08	
5	14	LaCrosseDTH	▼ Schlafzimmer	THS_#04	
6	37	LaCrosseDTH	▼ Jonas	THS_#02	
7	40	LaCrosseDTH	▼ IT-Schrank	THS_#09	
8	9	LaCrosseDTH	▼ Lager	THS_#07	
9	8	LaCrosseDTH	▼ Außensensor	THS_#00	

MQTT

Für IKEA Sensoren -> Die angepasste Firmware gibt es [hier](#).

Instanzeinstellungen: mqtt.0 4.11.1 🔄 warn ?

VERBINDUNG

MQTT EINSTELLUNGEN

IP

Server/Broker

☐ WebSockets benutzen

Verbindungseinstellungen

IP Adresse

Port

[IPv4] 192.168.10.11 - eth0

1883

☐ SSL

Authentifizierungseinstellungen

Benutzer

Passwort

Passwort (wiederholen)

mqttuser

Philips

**Instanzeinstellungen: philips-air.0**v1.0.3⏸↻warn✎

Geräte-IP

192.168.10.125

Lebenszeitüberschreitung

30000

⬆
⬇
⬇
⬆

Wiederverbindungsintervall

30000

⬆
⬇
⬇
⬆

ms

ms

Kommunikationsprotokoll

CoAP

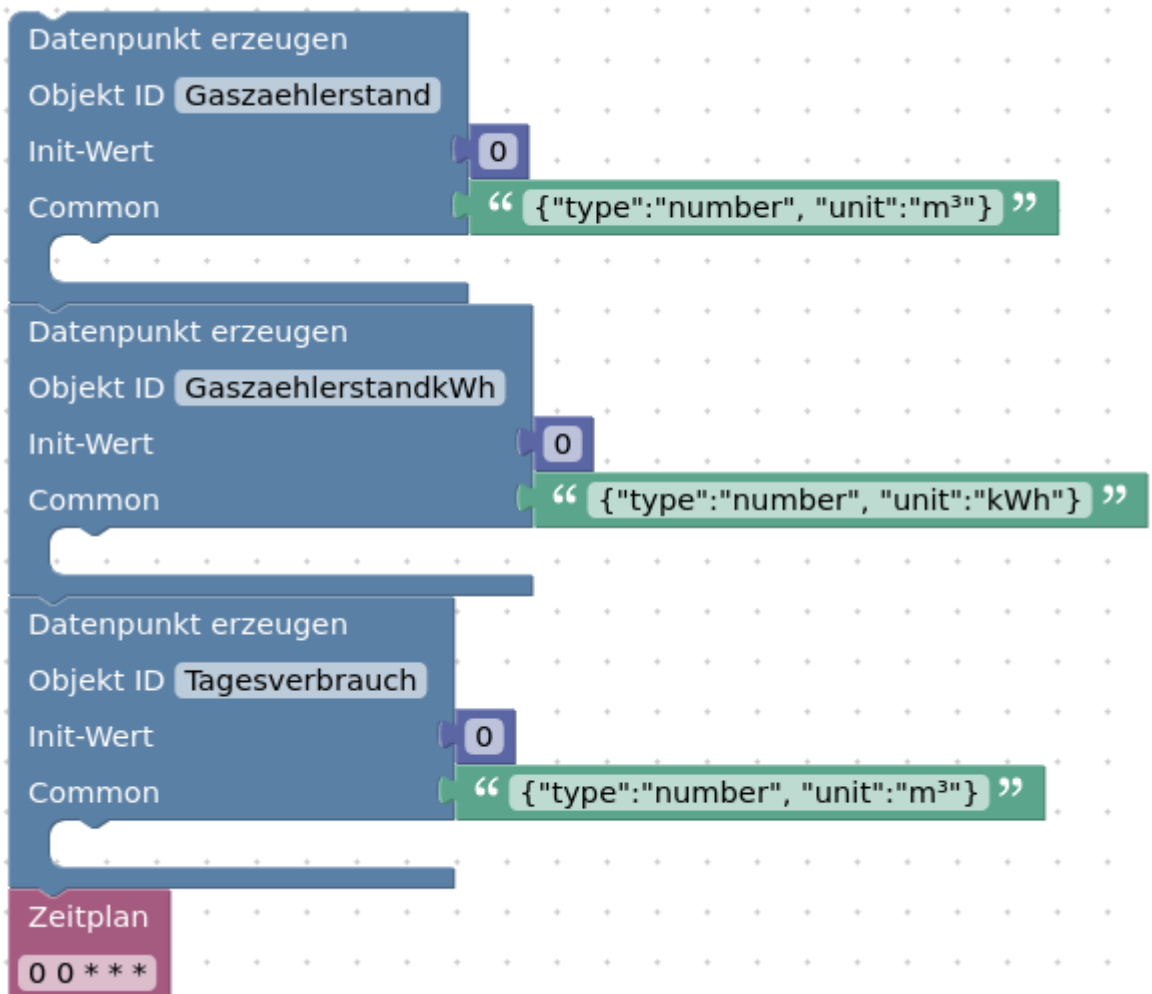
▼

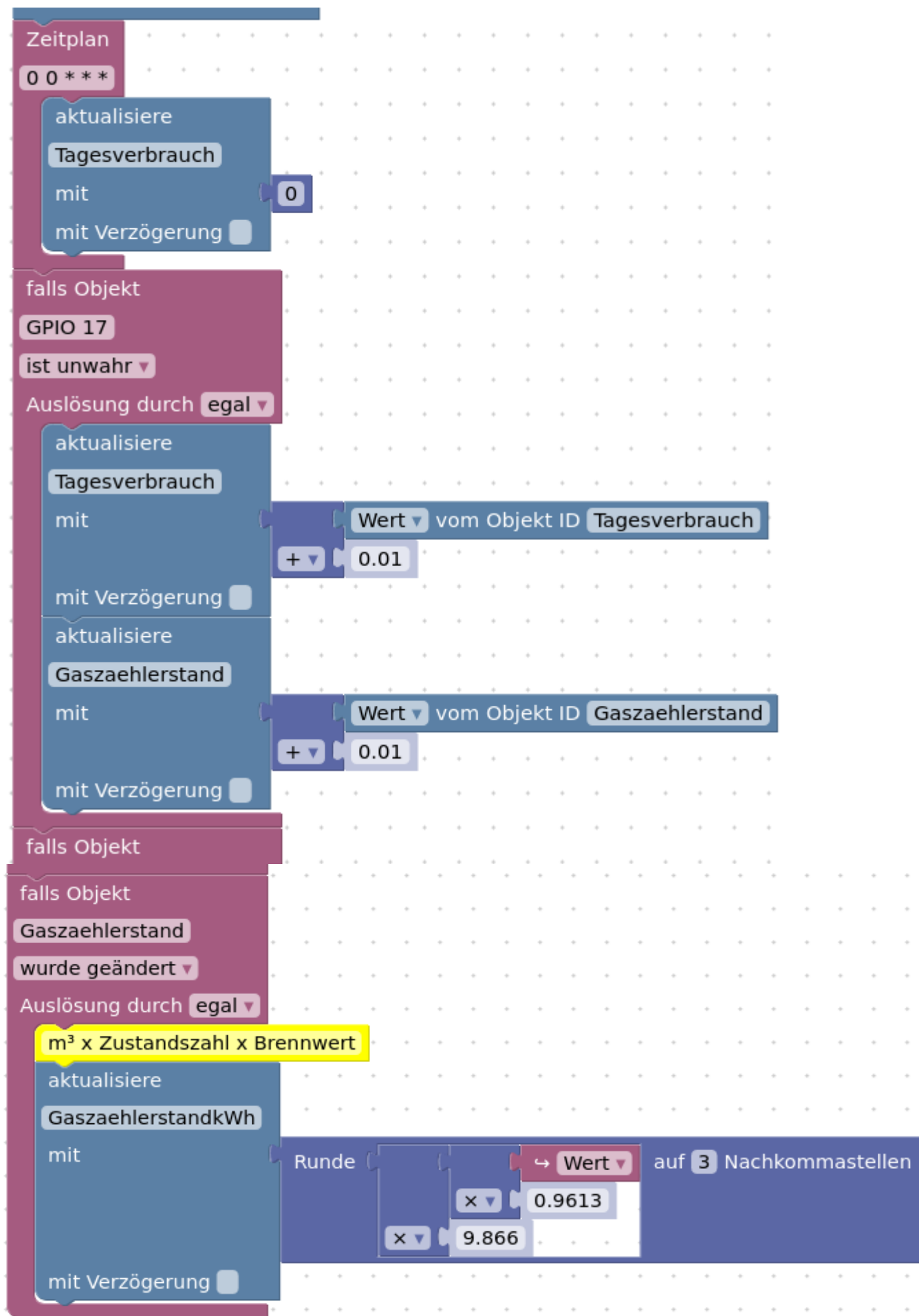
Gaszähler

Reedkontakt am GPIO #17 - Name: Gaszähler

The screenshot displays the 'Instanzen' (Instances) configuration page in the 'HUBBROWER' interface. The page is titled 'Instanzeinstellungen: rpi2.0'. The 'GPIOs' tab is selected in the top navigation bar. The main content area shows various configuration options for the Raspberry Pi 2.0 instance. Under the 'Eingänge' (Inputs) section, the 'Enable 17' checkbox is checked, and the label 'Gaszähler' (Gas meter) is visible in the dropdown menu. Other options include 'Enable 2', 'Enable 16', and 'Enable 18', all of which are currently unchecked. The 'Ausgänge' (Outputs) section is also visible but not expanded.

Blockly





Und das ganze als Java Script:

```
createState('Gaszaehlerstand', 0, JSON.parse('{ "type": "number",  
"unit": "m³" }'), async () => {  
});  
createState('GaszaehlerstandkWh', 0, JSON.parse('{ "type": "number",  
"unit": "kWh" }'), async () => {  
});  
createState('Tagesverbrauch', 0, JSON.parse('{ "type": "number",  
"unit": "m³" }'), async () => {  
});  
schedule("0 0 * * *", async () => {  
    setState('javascript.0.Tagesverbrauch' /* Tagesverbrauch */, 0, true);  
});  
on({ id: 'rpi2.0.gpio.17.state' /* GPIO 17 */, val: false }, async (obj) =>  
{  
    let value = obj.state.val;  
    let oldValue = obj.oldState.val;  
    setState('javascript.0.Tagesverbrauch' /* Tagesverbrauch */,  
(getState('javascript.0.Tagesverbrauch').val + 0.01), true);  
    setState('javascript.0.Gaszaehlerstand' /* Gaszaehlerstand */,  
(getState('javascript.0.Gaszaehlerstand').val + 0.01), true);  
});  
on({ id: 'javascript.0.Gaszaehlerstand' /* Gaszaehlerstand */, change: 'ne'  
}, async (obj) => {  
    let value = obj.state.val;  
    let oldValue = obj.oldState.val;  
    // m³ x Zustandszahl x Brennwert  
    setState('javascript.0.GaszaehlerstandkWh' /* GaszaehlerstandkWh */,  
Math.round(((obj.state ? obj.state.val : "") * 0.9613 * 9.866) * 1000) /  
1000, true);  
});
```

```
//JTNDeG1sJTIweG1sbnMlM0QlMjJodHRwcyUzQSUyRiUyRmRldmVsb3BlcnMuZ29vZ2x1LmNvbS  
UyRmJsb2NrbHk1MkZ4bWw1MjIlM0U1M0N1bG9jayUyMHR5cGU1M0QlMjJjcmVhdGULMjIlMjBpZC  
UzRCUyMjBCJTdCJTdETGxSZCUyQ2QhQnI2NyU0MFZsYnYlMjIlMjB4JTNEJTIyMTEzJTIyJTIweS  
UzRCUyMi0xMTIlMjIlM0U1M0NmaWVsZCUyMG5hbWULM0QlMjJ0QU1FJTiyJTNFR2FzemFlaGxlcn  
N0YW5kJTNDJTJGZml1bGQlM0U1M0N2YWx1ZSUyMG5hbWULM0QlMjJWQUxvRSUyMiUzRSUzQ2Jsb2  
NrJTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQlM0QlMjJ6Q001IWIUzQiUyNCUzRGV2c1  
p+ZyUyMyUyNCUyNU1lMkNkJTIyJTNFJTNDZmllbGQlMjBuYw11JTNEJTIyTlVNJTIyJTNFMCUzQy  
UyRmZpZWxkJTNFJTNDJTJGYmxvY2s1M0U1M0MlMkZ2YWx1ZSUzRSUzQ3ZhbHVlJTIwbmFtZSUzRC  
UyMkNPTU1PTiUyMiUzRSUzQ2Jsb2NrJTIwdHlwZSUzRCUyMnRleHQLMjIlMjBpZCUzRCUyMiUyQn  
dhSlhpQmV+MWRuJTQwU05CWEQlNURiJTIyJTNFJTNDZmllbGQlMjBuYw11JTNEJTIyVEVYVCUyMi  
UzRSUzQ3QiUyMnR5cGU1MjIlM0U1M0NmaWVsZCUyMG5hbWULM0QlMjJ0QU1FJTiyJTNBJTIyYmVSVDMi  
VCMYUyMiU3RCUzQyUyRmZpZWxkJTNFJTNDJTJGYmxvY2s1M0U1M0MlMkZ2YWx1ZSUzRSUzQ25leH  
QlM0U1M0N1bG9jayUyMHR5cGU1M0QlMjJjcmVhdGULMjIlMjBpZCUzRCUyMjIlNU1lM0QlNUREK  
mZ3ZVV5JTVeEeiU1RGZLUMiQTIUyMiUzRSUzQ2ZpZWxkJTIwbmFtZSUzRCUyMk5BTUULMjIlM0VHYX  
N6YWVobGVyc3RhbRrV2glM0MlMkZmaWVsZCUzRSUzQ3ZhbHVlJTIwbmFtZSUzRCUyMlZBTfVFJ  
IyJTNFJTNDYmxvY2s1MjB0eXB1JTNEJTIyYmVf0aF9udW1iZXIlMjIlMjBpZCUzRCUyMjYlMjN2a  
F2NEpoaVklN0NtWiU2MHR2YTQlMkIlMjIlM0U1M0NmaWVsZCUyMG5hbWULM0QlMjJ0VU0lMjIlM0
```

UwJTNDJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNEJTNDdmFsdWUlmjBuYW1lJTNEJTiyQ09NTU90JTiyJTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiydGV4dCUyMiUyMGlkJTNEJTiybEqlN0RwNy5WJTii1JTNEIVMlM0FlJTJGKFBqMSU1QmUlmjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJURVhUJTiyJTNEJTdCJTiydHlwZSUyMiUzQSUyMm51bWJlciUyMiUyQyUyMCUyMnVuaXQlMjIlM0ElMjJrV2g1MjIlN0QlM0MlMkZmaWVsZCUzRSUzQyUyRmJsb2NrJTNEJTNDJTJGdmFsdWUlm0Ulm0NuZXh0JTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiyY3JlYXRlJTiyJTiiwaWQlM0QlMjJkdcU1RVUlmjR3fiUyQzNSfilVUEPkUCUyQiUyNGMlMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJ0QU1FJTiyJTNEFVGFnZXN2ZXJicmF1Y2g1M0MlMkZmaWVsZCUzRSUzQ3ZhbHVlJTiiwbfmFtZSUzRCUyMlZBTFFVJTiyJTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiybWF0aF9udW1iZXIlMjIlMjBpZCUzRCUyMm9iJTNCXzA5ISU1RXVrZV8lM0JjJTQwVXF0Q1o1MjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJ0VU0lMjIlM0UwJTNDJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNEJTNDdmFsdWUlmjBuYW1lJTNEJTiyQ09NTU90JTiyJTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiydGV4dCUyMiUyMGlkJTNEJTiyQnI3eSUzRCUyQ2RHQUY5SSU1RG1iJTNGei0lMkMlM0ElMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJURVhUJTiyJTNEJTdCJTiydHlwZSUyMiUzQSUyMm51bWJlciUyMiUyQyUyMCUyMnVuaXQlMjIlM0ElMjJtJUMyJUizJTiyJTdEJTNDJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNEJTNDbmV4dCUzRSUzQ2Jsb2NrJTiiwHlwZSUzRCUyMnNjaGVkdWx1JTiyJTiiwaWQlM0QlMjIlMjJNUWCUzRFpYJTdCLXcyY3Ulm0RVaiFudU8qJTiyJTNEJTNDZmlbGQlMjBuYW1lJTNEJTiyU0NIRURVTEUlmjIlM0UwJTiiwMCUyMColMjAqJTiiwKiUzQyUyRmZpZWxkJTNEJTNDc3RhdGVtZW50JTiiwbfmFtZSUzRCUyMlNUQVRFTUV0VCUyMiUzRSUzQ2Jsb2NrJTiiwHlwZSUzRCUyMnVwZGF0ZSUyMiUyMGlkJTNEJTiyJTQwMHVGbkszcCU0MGQoYUZOelQlN0MlMkZsYiUyMiUyMGlubGluZSUzRCUyMmZhbHNlJTiyJTNEJTNDbXV0YXRpb24lMjB4bWxucyUzRCUyMmh0dHAlM0ElMkYlMkZ3d3cudzMub3JnJTJGMTk50SUyRnhodG1sJTiyJTiiwZGVsYXlfaW5wdXQlM0QlMjJmYWxzZSUyMiUzRSUzQyUyRm11dGF0aW9uJTNEJTNDZmlbGQlMjBuYW1lJTNEJTiyT0lEJTiyJTNEFamF2YXNjcm1wdC4wLlRhZ2VzdmVYnJhdWNoJTNDJTJGZmlbGQlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJXSvRiX0RFTEFZJTiyJTNEFRkFMU0Ulm0MlMkZmaWVsZCUzRSUzQ3ZhbHVlJTiiwbfmFtZSUzRCUyMlZBTFFVJTiyJTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiybWF0aF9udW1iZXIlMjIlMjBpZCUzRCUyMjQpciUyQ1glNjBGZy4lNUJxJTNEdV9IKmxmUms1MjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJ0VU0lMjIlM0UwJTNDJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNEJTNDJTJGYmxvY2s1M0Ulm0MlMkZzdGF0ZW1lbnQlM0Ulm0NuZXh0JTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiyb24lMjIlMjBpZCUzRCUyMmFGZiUyRiU0MCU1RSpsNEgpcCUyMyUyQ3U2RiU1RS4pJTiyJTNEJTNDZmlbGQlMjBuYW1lJTNEJTiyT0lEJTiyJTNEFcnBpMi4wLmdwaW8uMTcuc3RhdGUlM0MlMkZmaWVsZCUzRSUzQ2ZpZWxkJTiiwbfmFtZSUzRCUyMkNPTkRjVElPTiUyMiUzRWZhbHNlJTNDJTJGZmlbGQlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJBQ0tfQ090RElUSU90JTiyJTNEJTNDJTJGZmlbGQlM0Ulm0NzdGF0ZW1lbnQlMjBuYW1lJTNEJTiyU1RBVEVNRU5UJTiyJTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiydXBkYXRlJTiyJTiiwaWQlM0QlMjJ0ZHp+anc1NjBPJTdEJTii0GklN0NNSEJUQm41JTiyJTiiwaW5saW5lJTNEJTiyZmFsc2UlmjIlM0Ulm0NtdXRhdGlubiUyMHhtbG5zJTNEJTiyahR0cCUzQSUyRiUyRnd3dy53My5vcmlkYX0Tk5JTJGeGh0bWw1MjIlMjBkZWxheV9pbNBlDCUzRCUyMmZhbHNlJTiyJTNEJTNDJTJGbXV0YXRpb24lM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJPSUQlMjIlM0VqYXZhc2NyaXB0LjAuVGFnZXN2ZXJicmF1Y2g1M0MlMkZmaWVsZCUzRSUzQ2ZpZWxkJTiiwbfmFtZSUzRCUyMldJVEhfREVMOVKlMjIlM0VGQUxTRSUzQyUyRmZpZWxkJTNEJTNDdmFsdWUlmjBuYW1lJTNEJTiyVkfMVU0lMjIlM0Ulm0NibG9jayUyMHR5cGUlM0QlMjJtYXR0X2FyaXRobWV0aWw1MjIlMjBpZCUzRCUyMkRXT1R0cm9NJTdDRkxXJTJGWGclMkjrJTQwa3IlMjIlMjBpbmxbpmlM0QlMjJmYWxzZSUyMiUzRSUzQ2ZpZWxkJTiiwbfmFtZSUzRCUyMk9QJTiyJTNEFQUREJTNDJTJGZmlbGQlM0Ulm0N2YWx1ZSUyMG5hbWUlm0QlMjJBjTiyJTNEJTNDc2hhZG93JTiiwHlwZSUzRCUyMm1hdGhfbnVtYmVYJTiyJTiiwaWQlM0QlMjI4NyU1RFpqJTii0RyU1RC4qJTdCdHJXYTAqQmh+JTiyJTNEJTNDZmlbGQlMjBuYW1lJTNEJTiyTlVNJTiyJTNEFMSUzQyUyRmZpZWxkJTNEJTNDJTJGc2hhZG93JTNEJTNDYmxvY2s1MjB0eXBlJTNEJTiyZ2V0X3ZhbHVlJTiyJTiiwaWQlM0QlMjJN0SU0ME94JTJD0GdGMCUzQUHdbyUyNHhfdCUyQiklMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJBVFRSJTiyJTNEFdmFsJTNDJTJGZmlbGQlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJPSUQlMjIlM0VqYXZhc2NyaXB0LjAuVGFnZXN2ZXJicmF1Y2g1M0MlMkZmaWVsZCUzRSUzQyUyRmJsb2NrJTNEJTNDJTJGdmFsdWUlm0Ulm0N2YWx1ZSUyMG5hbWUlm0QlMjJCJTiyJTNEJTNDc2hhZG93JTiiwHlwZSUzRCUyMm1hdGhfbnVtYmVYJTiyJTiiwaWQlM0QlMjJXJT

<https://von-thuelen.de/>

V0aWmLMjI lMjBpZCUzRCUyMldldCUzRGdVRkQlMjU0aDNGJtDcSDJxZSguJTIyJTIwaW5saW5lJTNEJTIyZmFsc2U lMjI lM0U lM0NmaWVsZCUyMG5hbWU lM0Q lMjJPUCUyMiUzRU1VTFRJUE xZJTNDJTGZml lbGQ lM0U lM0N2YWx1ZSUyMG5hbWU lM0Q lMjJBjTIyJTNFJTNDc2hhZG93JTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQ lM0Q lMjJYeTi lNjB0cCpleiUyM3U lNUVwJTNCJTQwc zJEKU0 lMjI lM0U lM0NmaWVsZCUyMG5hbWU lM0Q lMjJ0VU0 lMjI lM0U xJTNDJTGZml lbGQ lM0U lM0M lMkZzaGFkb3c lM0U lM0NibG9jayUyMHR5cGU lM0Q lMjJvbl9zb3VyY2U lMjI lMjBpZCUzRCUyMmJlJTGew0 lMCUyNSUyQ0thZXZwSDZFJTVCKjM lMjI lM0U lM0NmaWVsZCUyMG5hbWU lM0Q lMjJBVFRSJT IyJTNFc3RhdGUudmFsJTNDJTGZml lbGQ lM0U lM0M lMkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDdmFsdWU lMjBuYW lJTN EJTIyQiUyMiUzRSUzQ3NoYWRvdyUyMHR5cGU lM0Q lMjJtYXR oX251bWJlciUyMiUyMG lKjTNEJTIyMTduNDRzb2gtTUMubEItUHQ lMjU lMjVoJTIyJTNFJTNDZml lbGQ lMjBuYW lJTN EJTIyTlVNJTIyJTNFMC45NjEzJTNDJTGZml lbGQ lM0U lM0M lMkZzaGFkb3c lM0U lM0M lMkZ2YWx1ZSUzRSUzQyUyRmJsb2NrJTNFJTNDJTGdmFsdWU lM0U lM0N2YWx1ZSUyMG5hbWU lM0Q lMjJCJTIyJTNFJTNDc2hhZG93JTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQ lM0Q lMjJDJtDeZ0hPJTVEJTVCMGQ lN0R+aiU0MCU lQnQ2LlZnWCUyMiUzRSUzQ2ZpZWxkJTIwbmFtZSUzRCUyMk5VTSUyMiUzRTkuODY2JTNDJTGZml lbGQ lM0U lM0M lMkZzaGFkb3c lM0U lM0M lMkZ2YWx1ZSUzRSUzQyUyRmJsb2NrJTNFJTNDJTGdmFsdWU lM0U lM0M lMkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDJTGy mxvY2s lM0U lM0M lMkZuZXh0JTNFJTNDJTGy mxvY2s lM0U lM0M lMkZzdGF0ZW lbnQ lM0U lM0M lMkZibG9jayUzRSUzQyUyRm5leHQ lM0U lM0M lMkZibG9jayUzRSUzQyUyRm5leHQ lM0U lM0M lMkZibG9jayUzRSUzQyUyRm5leHQ lM0U lM0M lMkZibG9jayUzRSUzQyUyRm5leHQ lM0U lM0M lMkZibG9jayUzRSUzQyUyRnhtbCUzRQ==

InfluxDB

Instanzeinstellungen: influxdb.0

v3.2.0

warn

DB SETTINGS

STANDARDEINSTELLUNGEN

EXPERTENEINSTELLUNGEN

DB-Version

1.x

Protokoll

http

Server

localhost

Port

8086

Benutzer

admin

Passwort

Passwort (wiederholen)

DB Name

iobroker

Request Timeout

30000

Millisekunden

Runden Sie bei der Abfrage die Zahlen auf

4

☒ Protokollieren Sie den letzten Wert erneut beim Start

Schreibaktionen zusammenfassen

0

Schreibintervall

60

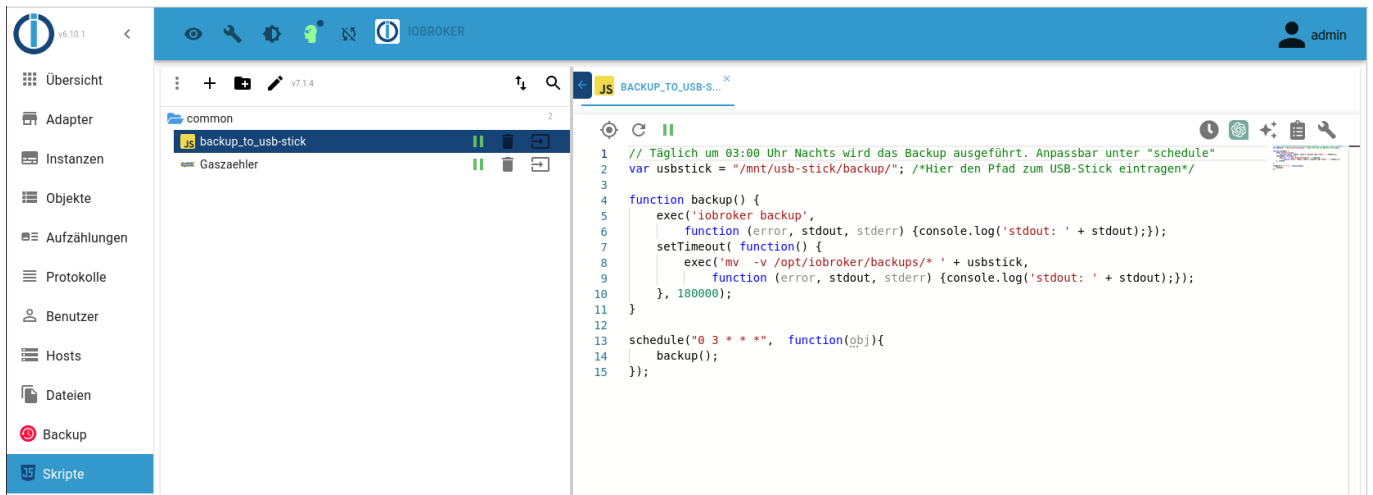
Sekunden

Wert 0 schreibt Werte direkt

DIE VERBINDUNG ZUM SERVER TESTEN

ALLE DATEN IN DATENBANK LÖSCHEN

Backup durch Script



```
// Täglich um 03:00 Uhr Nachts wird das Backup ausgeführt. Anpassbar unter
"schedule"
var usbstick = "/mnt/usb-stick/backup/"; /*Hier den Pfad zum USB-Stick
eintragen*/

function backup() {
    exec('iobroker backup',
        function (error, stdout, stderr) {console.log('stdout: ' +
stdout);});
    setTimeout( function() {
        exec('mv -v /opt/iobroker/backups/* ' + usbstick,
            function (error, stdout, stderr) {console.log('stdout: ' +
stdout);});
    }, 180000);
}

schedule("0 3 * * *", function(obj){
    backup();
});
```

Update und Bugfixing

```
iobroker stop
iobroker update
iobroker fix
# oder curl -sL https://iobroker.net/fix.sh | bash -
iobroker upgrade self
iobroker start
```


From:

<https://von-thuelen.de/> - **Christophs DokuWiki**

Permanent link:

<https://von-thuelen.de/doku.php/wiki/projekte/iobroker/uebersicht?rev=1701009615>



Last update: **2023/11/26 14:40**