

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/rpios_lite_armhf/images/rpios_lite_armhf-2023-02-22/2023-02-21-rpios-bullseye-armhf-lite.img.xz
wget
https://downloads.raspberrypi.com/rpios_lite_armhf/images/rpios_lite_armhf-2023-10-10/2023-10-10-rpios-bookworm-armhf-lite.img.xz
oder
#wget
https://downloads.raspberrypi.org/rpios_lite_arm64/images/rpios_lite_arm64-2023-02-22/2023-02-21-rpios-bullseye-arm64-lite.img.xz
wget
https://downloads.raspberrypi.com/rpios_lite_arm64/images/rpios_lite_arm64-2023-10-10/2023-10-10-rpios-bookworm-arm64-lite.img.xz
# R-Pi Imager herunterladen und installieren:
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/dhcpcd.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 -qO- https://repos.influxdata.com/influxdata-archive_compatible.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_compatible.key
echo '393e8779c89ac8d958f81f942f9ad7fb82a25e133faddaf92e15b16e6ac9ce4c
influxdata-archive_compatible.key' | sha256sum -c && cat influxdata-
archive_compatible.key | gpg --dearmor | sudo tee
/etc/apt/trusted.gpg.d/influxdata-archive_compatible.gpg > /dev/null
echo 'deb [signed-by=/etc/apt/trusted.gpg.d/influxdata-archive_compatible.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_compatible.key
echo '393e8779c89ac8d958f81f942f9ad7fb82a25e133faddaf92e15b16e6ac9ce4c
influxdata-archive_compatible.key' | sha256sum -c && cat influxdata-
archive_compatible.key | gpg --dearmor | sudo tee
/etc/apt/trusted.gpg.d/influxdata-archive_compatible.gpg > /dev/null
echo 'deb [signed-by=/etc/apt/trusted.gpg.d/influxdata-archive_compatible.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
# ioBroker 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 via 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 ★★★★★ ⋮	backup ★★★★★ ⋮	influxdb ★★★★☆ ⋮	ds18b20 ★★★★★ ⋮	simple-api ★★★★★ ⋮
Admin	BackItUp	Daten mit InfluxDB protokollieren	DS18B20	Einfache RESTful API
Installierte Instanzen: 1 Verfügbare Version: 6.3.5 Installierte Version: 6.3.5	Installierte Instanzen: 1 Verfügbare Version: 2.5.12 Installierte Version: 2.5.12	Installierte Instanzen: 1 Verfügbare Version: 3.2.0 Installierte Version: 3.2.0	Installierte Instanzen: 1 Verfügbare Version: 1.6.1 Installierte Version: 1.6.1	Verfügbare Version: 2.7.2 Installierte Version: 2.7.2
discovery ★★★★☆ ⋮	habpanel ★★★★★ ⋮	jeelink ★★★★★ ⋮	mqtt ★★★★★ ⋮	philips-air ★★★★☆ ⋮
Gerätesuche	HABpanel	Jeelink-Geräte	MQTT Broker/Client	Philips Luftreiniger
Installierte Instanzen: 1 Verfügbare Version: 3.1.0 Installierte Version: 3.1.0	Installierte Instanzen: 1 Verfügbare Version: 0.5.0 Installierte Version: 0.5.0	Installierte Instanzen: 1 Verfügbare Version: 0.1.4 Installierte Version: 0.1.4	Installierte Instanzen: 1 Verfügbare Version: 4.0.7 Installierte Version: 4.0.7	Installierte Instanzen: 1 Verfügbare Version: 1.0.1 Installierte Version: 1.0.1
rpi2 ★★★★★ ⋮	javascript ★★★★★ ⋮	socketio ★★★★★ ⋮	ws ★★★★★ ⋮	web ★★★★☆ ⋮
RPI-Monitor	Skriptausführung	socket.io	Web socket	WEB-Server
Installierte Instanzen: 1 Verfügbare Version: 1.3.2 Installierte Version: 1.3.2	Installierte Instanzen: 1 Verfügbare Version: 6.1.4 Installierte Version: 6.1.4	Verfügbarer Version: 4.2.0 Installierte Version: 4.2.0	Verfügbarer Version: 1.3.0 Installierte Version: 1.3.0	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

Sensoren											
+	Adresse	Remote-System-ID	Name	Abfrageintervall	Einheit	Faktor	Offset	Dezimalstellen	Null bei Fehler	Aktiviert	⋮
	10-00080359c712	keins	Vorlauftemperatur	Standard	°C	1	0	2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	10-00080359e7fe	keins	Rücklauftemperatur	Standard	°C	1	0	2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

JeeLink

TBC - Empfänger bauen und programmieren → TBC

Jeelink Settings

/dev/ttyUSB0	57600
Serial port	Baud rate
/dev/ttyUSB0 or /dev/ttyACM0	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 v4.1.1

VERBINDUNG MQTT EINSTELLUNGEN

IP Server/Broker WebSockets benutzen

Verbindungseinstellungen

IP Adresse [IPv4] 192.168.10.11 - eth0 Port 1883

SSL

Authentifizierungseinstellungen

Benutzer mqttuser Passwort Passwort (wiederholen)

Philips

Geräte-IP
192.168.10.125

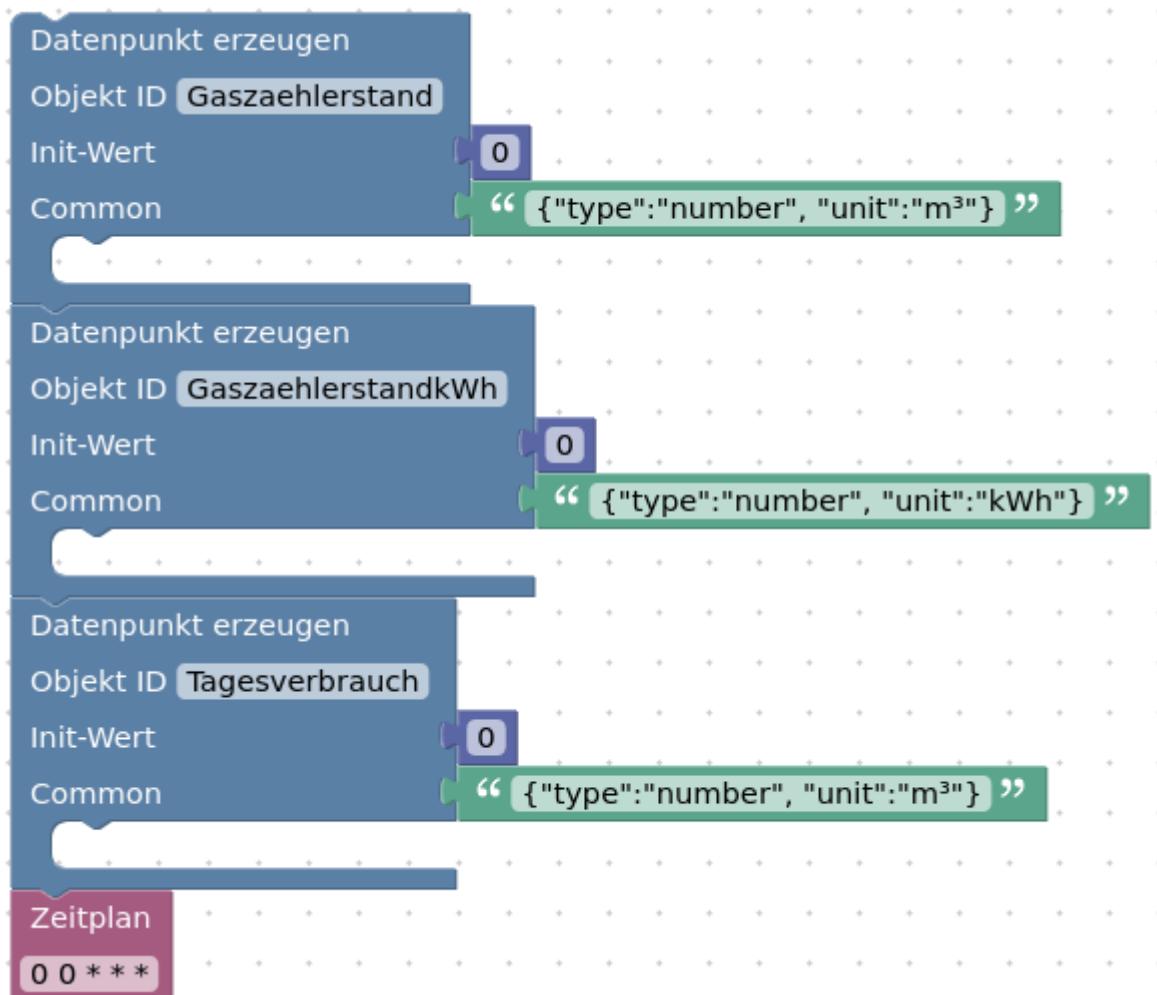
Lebenszeitüberschreitung Wiederverbindungsintervall
30000 30000
ms ms

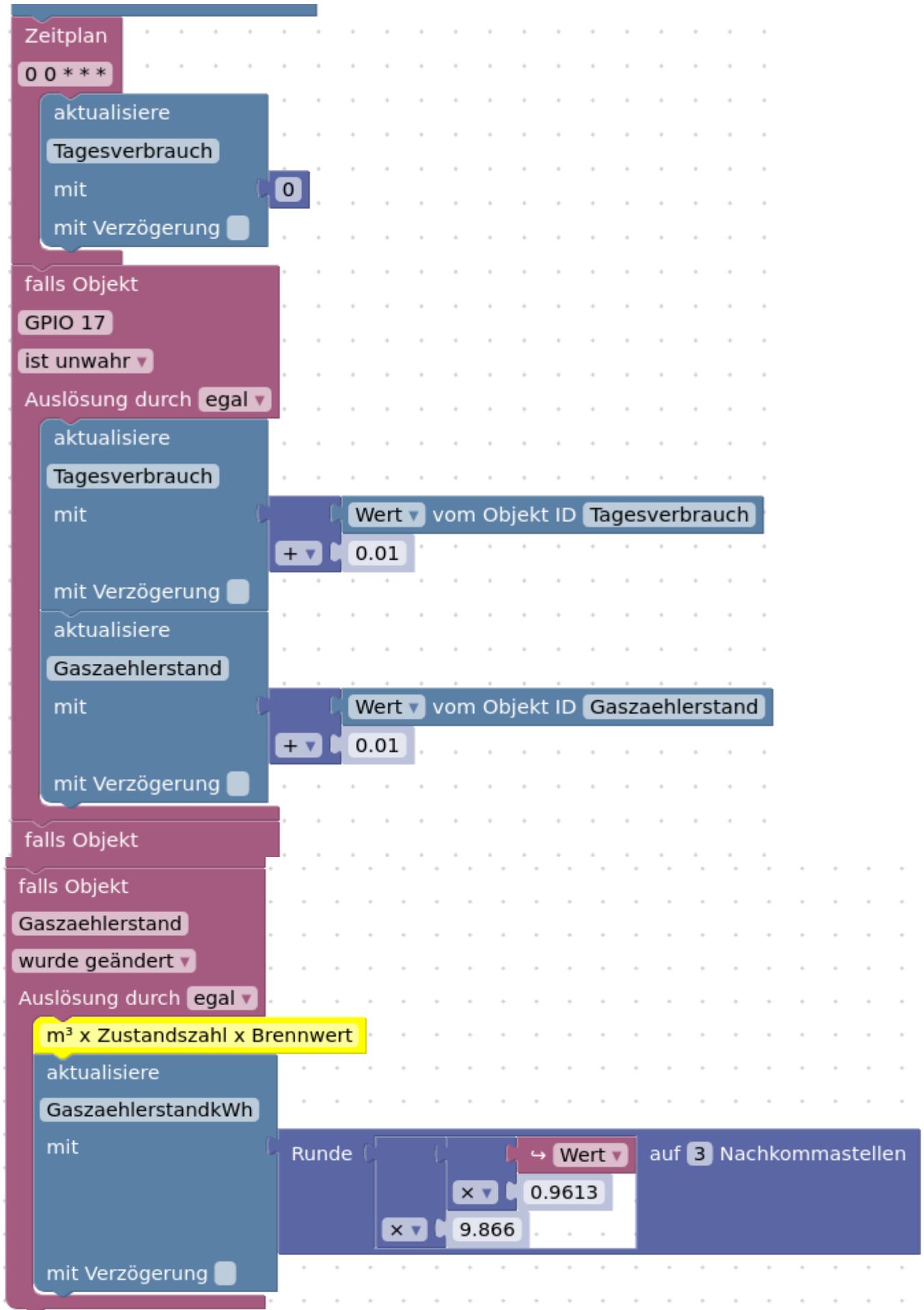
Kommunikationsprotokoll
CoAP

Gaszähler

Reedkontakt am GPIO #?

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);
});

// JTNDcG1sJTIweG1sbnMlM0QlMjJodHRwcyUzQSUYRiUYRmRldmVsB3BlcnMuZ29vZ2x1LmNvbS
UyRmJsb2NrbHk1MkZ4bWwlMjI1M0UlM0NibG9jayUyMHR5cGUlM0QlMjJjcmVhdGUlMjI1MjBpZC
UzRCUyMjBCJTdCJTdETGxSZCUyQ2QhQnI2NyU0MFZsYnYlMjI1MjB4JTNEJTiYmTEzJTiYJTiWeS
UzRCUyMi0xMTI1MjI1M0UlM0NmaWVsZCUyMG5hbWUlM0QlMjJ0QU1FJTiYJTNFR2FzemFlaGxlcn
N0YW5kJTNDJTJGZml1bGQlM0UlM0N2Yw1ZSUyMG5hbWUlM0QlMjJWQUxVRSUYMiUzRSUzQ2Jsb2
NrJTiwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTiYJTiwaWQlM0QlMjJ6Q01IWiuZQiUyNCUzRGV2c1
p+ZyUyMyUyNCUyNU1lMkNkJTiYJTNFJTNDZml1bGQlMjBuYw1lJTNEJTiYt1VNJTiYJTNFMCUzQy
UyRmZpZWxkJTNFJTNDJTJGYmxvY2slM0UlM0M1MkZ2Yw1ZSUzRSUzQ3ZhbHVlJTiwbmFtZSUzRC
UyMkNPTU1PTiUyMiUzRSUzQ2Jsb2NrJTiwdHlwZSUzRCUyMnRleHQlMjI1MjBpZCUzRCUyMiUyQn
dhSlhpQmV+MWRuJTQwU05CWEQlNURiJTiYJTNFJTNDZml1bGQlMjBuYw1lJTNEJTiYVEVYVCUyMi
UzRSU3QiUyMnR5cGUlMjI1M0ElMjJudW1iZXIlMjI1MkMlMjAlMjJ1bm10JTiYJTNBJTiybSVDMi
VCMyUyMiU3RCUzQyUyRmZpZWxkJTNFJTNDJTJGYmxvY2slM0UlM0M1MkZ2Yw1ZSUzRSUzQ25leH
QlM0UlM0NibG9jayUyMHR5cGUlM0QlMjJjcmVhdGUlMjI1MjBpZCUzRCUyMjI1NU1lM0QlNUREkm
Z3ZVV5JTVeeiU1RGZLUmIqTiUyMiUzRSUzQ2ZpZWxkJTiwbmFtZSUzRCUyMk5BTUUlMjI1M0VHYX
N6YWVobGVyc3RhbmRrV2glM0M1MkZmaWVsZCUzRSUzQ3ZhbHVlJTiwbmFtZSUzRCUyMlZBTFVFJT
IyJTNFJTNDYmxvY2slMjB0eXB1JTNEJTiYbWF0aF9udW1iZXIlMjI1MjBpZCUzRCUyMjYlMjN2al
F2NEpoaVklN0NtWiU2MHR2YTQlMkI1MjI1M0UlM0NmaWVsZCUyMG5hbWUlM0QlMjJ0VU0lMjI1M0
UwJTNDJTJGZml1bGQlM0UlM0M1MkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDdmFsdWUlMjBuYw

```

1lJTNEJTIyQ09NTU90JTiYJTNFJTNDYmxvY2s1MjB0eXB1JTNEJTiydGV4dCUyMiUyMG1kJTNEJT
IybEQ1N0RwNy5WJTi1JTNEIVM1M0F1JTJGKFbqMSU1QmUlMj1lM0UlM0NmaVsZCUyMG5hbWUlM0
QlMjJURvhUJTiYJTNFJTdCJTiydHlwZSuYMiUzQSuyMm51bWJlc1UyMiUyQyUyMCUyMnVuaXQlMj
1lM0E1MjJrV2g1Mj1lN0Q1M0M1MkZmaVsZCUzRSUzQyUyRmJsb2NrJTNFJTNDJTJGdmFsdWU1M0
UlM0NuZXh0JTNFJTNDYmxvY2s1MjB0eXB1JTNEJTiY3J1YXR1JTiYJTiwaWQlM0Q1MjJKdCU1RV
UlMjR3fiUyQzNSfilVUEpKUCUyQiUyNGM1Mj1lM0UlM0NmaVsZCUyMG5hbWUlM0Q1MjJOQU1FJT
IyJTNFVGFnZXN2ZXJicmF1Y2g1M0M1MkZmaVsZCUzRSUzQ3ZhHV1JTibmFtZSUzRCUyM1ZBTF
VFJTiYJTNFJTNDYmxvY2s1MjB0eXB1JTNEJTiYbWF0aF9udW1iZX1lMj1lMjBpZCUzRCUyMm9iJT
NCxZASISU1RXVrZV81M0JjJTQwVXF0Q1o1Mj1lM0UlM0NmaVsZCUyMG5hbWUlM0Q1MjJOVU01Mj
1lM0UwJTNFJTJGZmllbGQlM0U1M0M1MkZibG9jayUzRSUzQyUyRnZhHV1JTNFJTNDdmFsdWU1Mj
BuYW1lJTNEJTiYQ09NTU90JTiYJTNFJTNDYmxvY2s1MjB0eXB1JTNEJTiydGV4dCUyMiUyMG1kJT
NEJTiYQnI3eSUzRCUyQ2RHQUY5SSU1RG1iJTNGei01MkM1M0E1Mj1lM0UlM0NmaVsZCUyMG5hbW
UlM0Q1MjJURvhUJTiYJTNFJTdCJTiydHlwZSuYMiUzQSuyMm51bWJlc1UyMiUyQyUyMCUyMnVuaX
QlMj1lM0E1MjJtJUMyJUIzJTiYJtdeJTNFJTJGZmllbGQlM0U1M0M1MkZibG9jayUzRSUzQyUyRn
ZhHV1JTNFJTNDbmV4dCUzRSUzQ2Jsb2NrJTiwdHlwZSUzRCUyMnNjaGVkdWxlJTiYJTiwaWQlM0
QlMj1lMjNUWCUzRFpYJtclXcyY3U1M0RVaiFudU8qJTiYJTNFJTNDZmllbGQlMjBuYW1lJTNEJT
IyU0NIRURVTE1Mj1lM0UwJTiwMCUyMColMjAqJTiwKiUzQyUyRmZpZwxkJTNFJTNDc3RhGvtZW
50JTiwbmFtZSUzRCUyM1NUQVRFTUV0VCUyMiUzRSUzQ2Jsb2NrJTiwdHlwZSUzRCUyMnVwZGFOZS
UyMiUyMG1kJTNEJTiYJTwMHVGbkszCCU0MGQoYUZ0elQ1N0M1MkZsYiUyMiUyMG1ubGluZSUzRC
UyMmZhbHN1JTiYJTNFJTNDbXV0YXRpb241MjB4bWxucyUzRCUyMmh0dHALM0E1MkY1MkZ3d3cudz
Mub3JnJTJGMTk50SUyRnhodG1sJTiYJTiwZGVsYX1faW5wdXQlM0Q1MjJmYwzzSUyMiUzRSUzQy
UyRm11dGF0aW9uJTNFJTNDZmllbGQlMjBuYW1lJTNEJTiYt0leJTiYJTNFamF2YXNjcmldC4wLl
RhZ2VzdmVYnJhdWNoJTNFJTJGZmllbGQlM0U1M0NmaVsZCUyMG5hbWUlM0Q1MjJXSVRIX0RFTE
FZJTiYJTNFRkFMU0U1M0M1MkZmaVsZCUzRSUzQ3ZhHV1JTibmFtZSUzRCUyM1ZBTFVfJTiYJ
NFJTNFYT2s1MjB0eXB1JTNEJTiYbWF0aF9udW1iZX1lMj1lMjBpZCUzRCUyMjQpc1UyQ1glNj
BGZy41NUJxJTNEdV9IKmxmUms1Mj1lM0U1M0NmaVsZCUyMG5hbWUlM0Q1MjJOVU01Mj1lM0UwJT
NDJTJGZmllbGQlM0U1M0M1MkZibG9jayUzRSUzQyUyRnZhHV1JTNFJTNDJTJGymxvY2s1M0U1M0
M1MkZzdGF0ZW1lbnQlM0U1M0NuZXh0JTNFJTNDYmxvY2s1MjB0eXB1JTNEJTiyb241Mj1lMjBpZC
UzRCUyMmFGZiUyRiU0MCU1RSpsNEgpcCUyMyUyQ3U2Ri1RS4pJTiYJTNFJTNDZmllbGQlMjBuYW
1lJTNEJTiYt0leJTiYJTNFcnBpMi4wLmdwaW8uMTcuc3RhdGU1M0M1MkZmaVsZCUzRSUzQ2ZpZW
xkJTiwbmFtZSUzRCUyMkNPTkRJVE1PTiUyMiUzRWZhBN1JTNFJTJGZmllbGQlM0U1M0NzdGF0ZW1lbn
QlMjBuYW1lJTNEJTiY1RBVEVNRU5UJTiYJTNFJTNDYmxvY2s1MjB0eXB1JTNEJTiydXBkYXR1JT
IyJTiwaWQlM0Q1MjJ0ZHP+anclNjBPJtdEJTi00Gk1n0NNSEJUqm41JTiYJTiwaW5saW51lJTNEJT
IyZmFsc2U1Mj1lM0U1M0NtdXRhdGlvb1UyMHhtbG5zJTNFJTNDJTJGZmllbGQlM0U1M0NzdGF0ZW1lbn
5vcmc1MkYx0Tk5JTJGeGh0bWwlMj1lMjBkZwheV9pbnB1dCUzRCUyMmZhbHN1JTiYJTNFJTNDJT
JGbXV0YXRpb241M0U1M0NmaVsZCUyMG5hbWUlM0Q1MjJPSUQ1Mj1lM0VqYXZhc2NyaXB0LjAuVG
FnZxN2ZXJicmF1Y2g1M0M1MkZmaVsZCUzRSUzQ2ZpZwxkJTiwbmFtZSUzRCUyM1dJVEhfREVMQV
klMj1lM0VGQUxTRSUzQyUyRmZpZwxkJTNFJTNDdmFsdWU1MjBuYW1lJTNEJTiYVkFMVUU1Mj1lM0
UlM0N1bG9jayUyMHR5cGU1M0Q1MjJtYXRoX2FyaXRobWV0aW1Mj1lMjBpZCUzRCUyMkRxT1R0cm
9NJtdDRkxXJTJGWGc1MkJrJTQwa31Mj1lMjBpbmxbpbmUlM0Q1MjJmYwzzSUyMiUzRSUzQ2ZpZW
xkJTiwbmFtZSUzRCUyMk9QJTiYJTNFQUREJTNFJTJGZmllbGQlM0U1M0N2Yw1ZSUyMG5hbWUlM0
QlMjJB1JTiYJTNFJTNDc2hhZG93JTiwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTiYJTiwaWQlM0Q1Mj
I4NyU1RFpqJTi0RyU1RC4qJTDcdHJXYTAqQmh+JTiYJTNFJTNDZmllbGQlMjBuYW1lJTNEJTiYt1
VNJTiYJTNFMSUzQyUyRmZpZwxkJTNFJTNDJTJGc2hhZG93JTNFJTNDYmxvY2s1MjB0eXB1JTNEJT
IyZ2V0X3ZhHV1JTiYJTiwaWQlM0Q1MjJNOSU0ME94JTJD0GdGMcuZQuhDbyUyNHhfdCuUyQik1Mj
1lM0U1M0NmaVsZCUyMG5hbWUlM0Q1MjJBVFRSjTiYJTNFdmFsJTNFJTNDZmllbGQlM0U1M0NmaVs
VsZCUyMG5hbWUlM0Q1MjJPSUQ1Mj1lM0VqYXZhc2NyaXB0LjAuVGFnZxN2ZXJicmF1Y2g1M0M1Mk
ZmaVsZCUzRSUzQyUyRmJsb2NrJTNFJTNDJTJGdmFsdWU1M0U1M0N2Yw1ZSUyMG5hbWUlM0Q1Mj
JCJTiYJTNFJTNDc2hhZG93JTiwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTiYJTiwaWQlM0Q1MjJXJT

VCJTDCKWNQJTD0TByLWpDNFApJTQwUzFqJTIyJTNFJTNZmllbGQlMjBuYw1lJTNEJTIyTlVNJT
IyJTNFMC4wMSUzQyUyRmZpZWxkJTNFJTNJGc2hhZG93JTNFJTNJGdmFsdWu1M0U1M0M1Mk
ZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDbmV4dCUzRSUzQ2Js2NrrJTIwdHlwZSUzRCUyMnVwZG
F0ZSUyMiUyMG1kJTNEJTIyanI5Uk41MKZIR0VVdmJKJTJDc2FzJTNFQw1MjI1MjBpbmxpbmU1M0
Q1MjJmYWxzZSUyMiUzRSUzQ211dGF0aw9uJTIweG1sbnM1M0Q1MjJodHRwJTNB1TJGJTJGd3d3Ln
czLm9yZyUyRjE50Tk1MkZ4aHrtbCUyMiUyMGrlbGF5X2lucHV0JTNFJTIyZmFsc2U1MjI1M0U1M0
M1MkZtdXRhdGlvb1UzRSUzQ2ZpZWxkJTIwbtmFtZSUzRCUyMk9JRCUyMiUzRWphdmFzY3JpcHQuMC
5HYXN6YWVobGVyc3RhbmQ1M0M1MkZmaWVsZCUzRSUzQ2ZpZWxkJTIwbtmFtZSUzRCUyMldJVEhfRE
VMQVklMjI1M0VGQUxTRSUzQyUyRmZpZWxkJTNFJTNDDmFsdWu1MjBuYw1lJTNEJTIyVkfMVU1Mj
I1M0U1M0N1bG9jayUyMHR5cGU1M0Q1MjJtYXRoX2FyaXRobWV0aW1MjI1MjBpZCUzRCUyMiE4LU
lwcDdDc1E1NUQ1M0Q1M0R0SW5aVkQzJTIyJTIwaW5saW51JTNFJTIyZmFsc2U1MjI1M0U1M0NmaW
VsZCUyMG5hbWu1M0Q1MjJPUCUyMiUzRUFERCUzQyUyRmZpZWxkJTNFJTNDDmFsdWu1MjBuYw1lJT
NEJTIyQSUyMiUzRSUzQ3NoYWVdyUyMHR5cGU1M0Q1MjJtYXRoX251bWJlciUyMiUyMG1kJTNEJTI
IyJTNEM2NCJTIzzVpiLw01N0JCQWJKX0pCJTNEJTI0JTIyJTNFJTNZmllbGQlMjBuYw1lJTNEJTI
IyTlVNJTIyJTNFMSUzQyUyRmZpZWxkJTNFJTNJGc2hhZG93JTNFJTNNDmxvY2s1MjB0eXB1JT
NEJTIyZ2V0X3ZhbHVlJTIyJTIwaWQ1M0Q1MjIyJTIcbXpXKkw1NDA1NjA5aCU0MGQwUC01Mk1Mk
Y0RiUyMiUzRSUzQ2ZpZWxkJTIwbtmFtZSUzRCUyMkFUVF1MjI1M0V2Yw1M0M1MkZmaWVsZCUzRS
UzQ2ZpZWxkJTIwbtmFtZSUzRCUyMk9JRCUyMiUzRWphdmFzY3JpcHQuMC5HYXN6YWVobGVyc3Rhbm
Q1M0M1MkZmaWVsZCUzRSUzQyUyRmJsb2NrJTNFJTNJGdmFsdWu1M0U1M0N2Yw1ZSUyMG5hbW
u1M0Q1MjJCJTIyJTNFJTNDC2hhZG93JTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQ1M0
Q1MjJxJTNBcDdGYXFzX2Y1N0JSdkkxX0oueD1lMjI1M0U1M0NmaWVsZCUyMG5hbWu1M0Q1MjJ0VU
01MjI1M0UwLjAxJTNFJTNJGdmFsdWu1M0U1M0M1MkZibG9jayUzRSUzQyUyRm5leHQ1M0U1M0M1Mk
ZibG9jayUzRSUzQyUyRnN0YXR1bWVudCUzRSUzQ251eHQ1M0U1M0N1bG9jayUyMHR5cGU1M0Q1Mj
JvbiUyMiUyMG1kJTNEJTIyN1dvBXclNjBxbF95dSU3Qmk5big1MkY1M0YzLSUyMiUzRSUzQ2ZpZW
xkJTIwbtmFtZSUzRCUyMk9JRCUyMiUzRWphdmFzY3JpcHQuMC5HYXN6YWVobGVyc3RhbmQ1M0M1Mk
ZmaWVsZCUzRSUzQ2ZpZWxkJTIwbtmFtZSUzRCUyMkNPTkRJVE1PTiUyMiUzRW51JTNFJTGZmllbG
Q1M0U1M0NmaWVsZCUyMG5hbWu1M0Q1MjJBQ0tFQ090RE1USU90JTIyJTNFJTNFJTGZmllbGQ1M0
u1M0NzdGF0ZW1lbNQ1MjBuYw1lJTNEJTIyU1RBVEVNRU5UJTIyJTNFJTNNDmxvY2s1MjB0eXB1JT
NEJTIyY29tbWVudCUyMiUyMG1kJTNEJTIyVC01MjQ1MjRsJTCUSFPa041M0YUJTDcZCU3QmYzNF
k1MjI1M0U1M0NmaWVsZCUyMG5hbWu1M0Q1MjJDT01NRU5UJTIyJTNFbSVDMiVCMyUyMHg1MjBadX
N0YW5kc3phaGw1MjB4JTIwQnJ1bm53ZXJ0JTNFJTGZmllbGQ1M0U1M0NuZXh0JTNFJTNNDmxvY2
s1MjB0eXB1JTNEJTIyDxBkYXRLJTIyJTIwaWQ1M0Q1MjIpby01M0FnMkpPJTQwbVc1N0NOUEM1Mk
NfLWdhJTIyJTIwaW5saW51JTNFJTIyZmFsc2U1MjI1M0U1M0NtdXRhdGlvb1UyMHhtbG5zJTNFJ
IyaHR0cCUzQSUyRiUyRnd3dy53My5vcmc1MkYx0Tk5JTJGeGh0bw1MjI1MjBkZwheV9pbnB1dC
UzRCUyMmZhbHN1JTIyJTNFJTNJGbXV0YXRpb241M0U1M0NmaWVsZCUyMG5hbWu1M0Q1MjJPSU
Q1MjI1M0VqYXZhc2NyaXB0LjAuR2FzemFlaGxlcnN0YW5ka1doJTNFJTGZmllbGQ1M0U1M0NmaW
VsZCUyMG5hbWu1M0Q1MjJXSVRIX0RFTEFZJTIyJTNFRkFMU0U1M0M1MkZmaWVsZCUzRSUzQ3Zbh
V1JTIwbtmFtZSUzRCUyM1ZBTFVFJTIyJTNFJTNNDmxvY2s1MjB0eXB1JTNEJTIy1bWF0aF9ybmrMax
h1ZCUyMiUyMG1kJTNEJTIyazBLJtdDSE1UVyU3Q3BWZGYtZyolMkNYN0glMjI1M0U1M0NmaWVsZC
UyMG5hbWu1M0Q1MjJuJTIyJTNFMyUzQyUyRmZpZWxkJTNFJTNDDmFsdWu1MjBuYw1lJTNEJTIyec
UyMiUzRSUzQ3NoYWVdyUyMHR5cGU1M0Q1MjJtYXRoX251bWJlciUyMiUyMG1kJTNEJTIyVCU0ME
M5JTDcEVyUyNVJqMzN2RWpnTndaciU3QiUyMiUzRSUzQ2ZpZWxkJTIwbtmFtZSUzRCUyMk5VTSUyMi
UzRTMuMTIzNCUzQyUyRmZpZWxkJTNFJTNJGc2hhZG93JTNFJTNNDmxvY2s1MjB0eXB1JTNEJTI
Iy1bWF0aF9hcml0aG1ldGljJTIyJTIwaWQ1M0Q1MjI1NjBUSiUzQVhfIXhmWmdjcsU1QiUzQk5JT3
5N1JTIyJTIwaW5saW51JTNFJTIyZmFsc2U1MjI1M0U1M0NmaWVsZCUyMG5hbWu1M0Q1MjJPUCUyMi
UzRU1VTFRJUExZJTNFJTNJGZmllbGQ1M0U1M0N2Yw1ZSUyMG5hbWu1M0Q1MjJB1JTIyJTNFJTNDC2
hhZG93JTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQ1M0Q1MjJtTFQ1N0NpJTDcR2U1N0
I1MjRQQmM1M0Q1MjQ2xqeUglMjI1M0U1M0NmaWVsZCUyMG5hbWu1M0Q1MjJ0VU01MjI1M0U1xJTNFJ
JGZmllbGQ1M0U1M0M1MkZzaGFkb3c1M0U1M0N1bG9jayUyMHR5cGU1M0Q1MjJtYXRoX2FyaXRobW
V0aW1MjI1MjBpZCUzRCUyMld1dCUzRGdVRkQ1MjU0aDNGJTDcSDJxZSguJTIyJTIwaW5saW51JT

NEJTIyZmFsc2UlMjIlM0UlM0NmaWVsZCUyMG5hbWUlM0QlMjJPUCUyMiUzRU1VTFRJUEzJTNNDJT
JGZmllbGQlM0UlM0N2Yw1ZSUyMG5hbWUlM0QlMjJBjTiJTNFJTNDc2hhZG93JTiwdHlwZSuZRC
UyMm1hdGhfbnVtYmVjTiYJTiwaWQlM0QlMjYTeTilNjB0cCpleiUyM3UlNUVwJTNcJtQwczJEKU
0lMjIlM0UlM0NmaWVsZCUyMG5hbWUlM0QlMjJ0VU0lMjIlM0UxJTNFJTJGZmllbGQlM0UlM0M1Mk
ZzaGFkb3clM0UlM0NibG9jayUyMHR5cGUlM0QlMjJvb19zb3VyY2UlMjIlMjBpZCUzRCUyMmJLJT
JGeW01MCUyNSUyQ0thZXwSDZFJTVCKjMlMjIlM0UlM0NmaWVsZCUyMG5hbWUlM0QlMjJBVFJSJT
IyJTNFc3RhdGUudmFsJTNFJTJGZmllbGQlM0UlM0M1MkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJT
NDdmFsdWUlMjBuYW1lJTNEJTiYQiUyMiUzRSUzQ3NoYWVdyUyMHR5cGUlM0QlMjJtYXRoX251bW
JlciUyMiUyMG1kJTNEJTiYMTduNDRzb2gtTUMubEItUHQlMjUlMjVoJTiYJTNFJTNDZmllbGQlMj
BuYW1lJTNEJTiYtVNjTiYJTNFMC45njEzJTNFJTJGZmllbGQlM0UlM0M1MkZzaGFkb3clM0UlM0
M1MkZ2Yw1ZSUzRSUzQyUyRmJsb2NrJTNFJTNDJTJGdmFsdWUlM0UlM0N2Yw1ZSUyMG5hbWUlM0
QlMjJCJTiYJTNFJTNDc2hhZG93JTiwdHlwZSuZRCUyMm1hdGhfbnVtYmVjTiYJTiwaWQlM0QlMj
JDJTdEZ0hPJTVeJTVCMGQlN0R+aiU0MCU1QnQ2L1ZNWCUyMiUzRSUzQ2ZpZWxkJTiwbmFtZSuZRC
UyMk5VTSUyMiUzRTku0DY2JTNFJTJGZmllbGQlM0UlM0M1MkZzaGFkb3clM0UlM0M1MkZ2Yw1ZS
UzRSUzQyUyRmJsb2NrJTNFJTNDJTJGdmFsdWUlM0UlM0M1MkZibG9jayUzRSUzQyUyRnZhbHVlJT
NFJTNFJTJGyMxvY2slM0UlM0M1MkZuZXh0JTNFJTNDJTJGyMxvY2slM0UlM0M1MkZzdGF0ZW1lb
QlM0UlM0M1MkZibG9jayUzRSUzQyUyRm5leHQlM0UlM0M1MkZibG9jayUzRSUzQyUyRm5leHQlM0
UlM0M1MkZibG9jayUzRSUzQyUyRm5leHQlM0UlM0M1MkZibG9jayUzRSUzQyUyRm5leHQlM0UlM0
M1MkZibG9jayUzRSUzQyUyRm5leHQlM0UlM0M1MkZibG9jayUzRSUzQyUyRnhtbCUzRQ==

InfluxDB

The screenshot shows the 'Instanzeinstellungen' (Instance Settings) page for InfluxDB v3.2.0. The 'DB SETTINGS' tab is selected. Key configuration parameters include:

- DB-Version: 1.x
- Protokoll: http
- Server: localhost
- Port: 8086
- Benutzer: admin
- Passwort: (redacted)
- Passwort (wiederholen): (redacted)
- DB Name: iobroker
- Request Timeout: 30000 Millisekunden
- Runden Sie bei der Abfrage die Zahlen auf: 4
- Protokollieren Sie den letzten Wert erneut beim Start: checked
- Schreibaktionen zusammenfassen: 0
- Schreibintervall: 60 Sekunden
- Wert 0 schreibt Werte direkt

At the bottom are two buttons: 'DIE VERBINDUNG ZUM SERVER TESTEN' (Test connection to server) and 'ALLE DATEN IN DATENBANK LÖSCHEN' (Delete all data in database).

Backup durch Script

```

v6.10.1
Übersicht
Adapter
Instanzen
Objekte
Aufzählungen
Protokolle
Benutzer
Hosts
Dateien
Backup
Skripte

JS BACKUP_TO_USB-S...
common
js backup_to_usb-stick
Gaszaehler

1 // Täglich um 03:00 Uhr Nachts wird das Backup ausgeführt. Anpassbar unter "schedule"
2 var usbstick = "/mnt/usb-stick/backup/"; /*Hier den Pfad zum USB-Stick eintragen*/
3
4 function backup() {
5     exec('ioBroker backup',
6         function (error, stdout, stderr) {console.log('stdout: ' + stdout);});
7     setTimeout( function() {
8         exec('mv -v /opt/ioBroker/backups/* ' + usbstick,
9             function (error, stdout, stderr) {console.log('stdout: ' + stdout);});
10    }, 180000);
11 }
12
13 schedule("0 3 * * *",  function(obj){
14     backup();
15 });

```

```

// 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=1698483772>

Last update: 2023/10/28 09:02



