

Neuinstallation ioBroker auf einem R-Pi 3 B+ oder 4 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-gpio
raspi-config nonint do_hostname iobroker
raspi-config nonint do_owewire 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/dhccpd.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
#
# Seit OS "Bookworm":
nmcli -p connection show
nmcli con mod "Kabelgebundene Verbindung 1" ipv4.addresses 192.168.10.11/24
ipv4.method manual
nmcli con mod "Kabelgebundene Verbindung 1" ipv4.gateway 192.168.10.1
nmcli con mod "Kabelgebundene Verbindung 1" ipv4.dns "192.168.10.1"
reboot
# oder
nmcli c down "Kabelgebundene Verbindung 1" && sudo nmcli c up
"Kabelgebundene Verbindung 1"
#
# nicht benötigte Dienste beenden und/oder entfernen:
systemctl disable ModemManager.service
systemctl stop ModemManager.service
apt-get purge modemmanager
#
mkdir -p /mnt/USB-64GB
echo "UUID="52ff610a-2612-4021-8a8d-b818b3dba1f4" /mnt/USB-64GB ext4
defaults                                0          1" >> /etc/fstab
```

```
systemctl daemon-reload
mount -a
```

GPIO für USV

```
aptitude install raspi-gpio
# konfiguriere GPIO #18 als Input mit internem Pull-Down:
raspi-gpio set 18 ip pd
# GPIO Zustand abfragen:
raspi-gpio get 18
```

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  
# ioBroker Infos anzeigen  
iobroker info
```

Backups

Erstellung

Siehe unten → Script oder per 'Backupup' Adapter

Wiederherstellung

- Verzeichniss '/opt/iobroker/backups' anlegen
- Kopieren des Backups dort hinein

Backups die mit dem Backup-Adapter erstellt wurden können nach dem kopieren via Web-Interface wieder eingespielt werden.

Alternativ kann auch die Konsole verwendet werden:

```
sudo su  
cd /opt/iobroker  
iobroker stop  
iobroker restore 0  
iobroker stop
```

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)
- Shelly

admin

★★★★★

⋮

Admin

📶

↓

📶

Installierte Instanzen: 1
Verfügbare Version: 6.3.5
Installierte Version: 6.3.5

backup

★★★★★

⋮

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

rpiz

★★★★★

⋮

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 mqtt.0 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
ms

Wiederverbindungsintervall
30000
ms

Kommunikationsprotokoll
CoAP

Gaszähler

Reedkontakt am GPIO #17 - Name: Gaszähler

Übersicht

Adapter

Instanzen

Objekte

Aufzählungen

Protokolle

Benutzer

Hosts

Dateien

Backup

Skripte

Benutzer

Hosts

Dateien

Backup

Skripte

Instanzeinstellungen: rpi2.0

GPIOs

☒ Die Eingänge sind hochgezogen

Entprellungsperiode (ms)
100

☐ Die Tasten sind hochgezogen

Entprellungszeitraum für Schaltflächen (ms)

Taste lange Druckdauer (ms)

Doppelklickzeit (ms)

Abfrageintervall für DHTxx / AM23xx-Geräte (ms)

Empfohlenes Minimum ist 2s. Auf Null setzen, um die Abfrage zu deaktivieren.

Port numbers relate to Broadcom SOC channel, not physical pins.

☐ Enable 2

Typ
Eingang

Etikette

☐ Enable 16

Typ
Eingang

Etikette

☒ Enable 17

Typ
Eingang

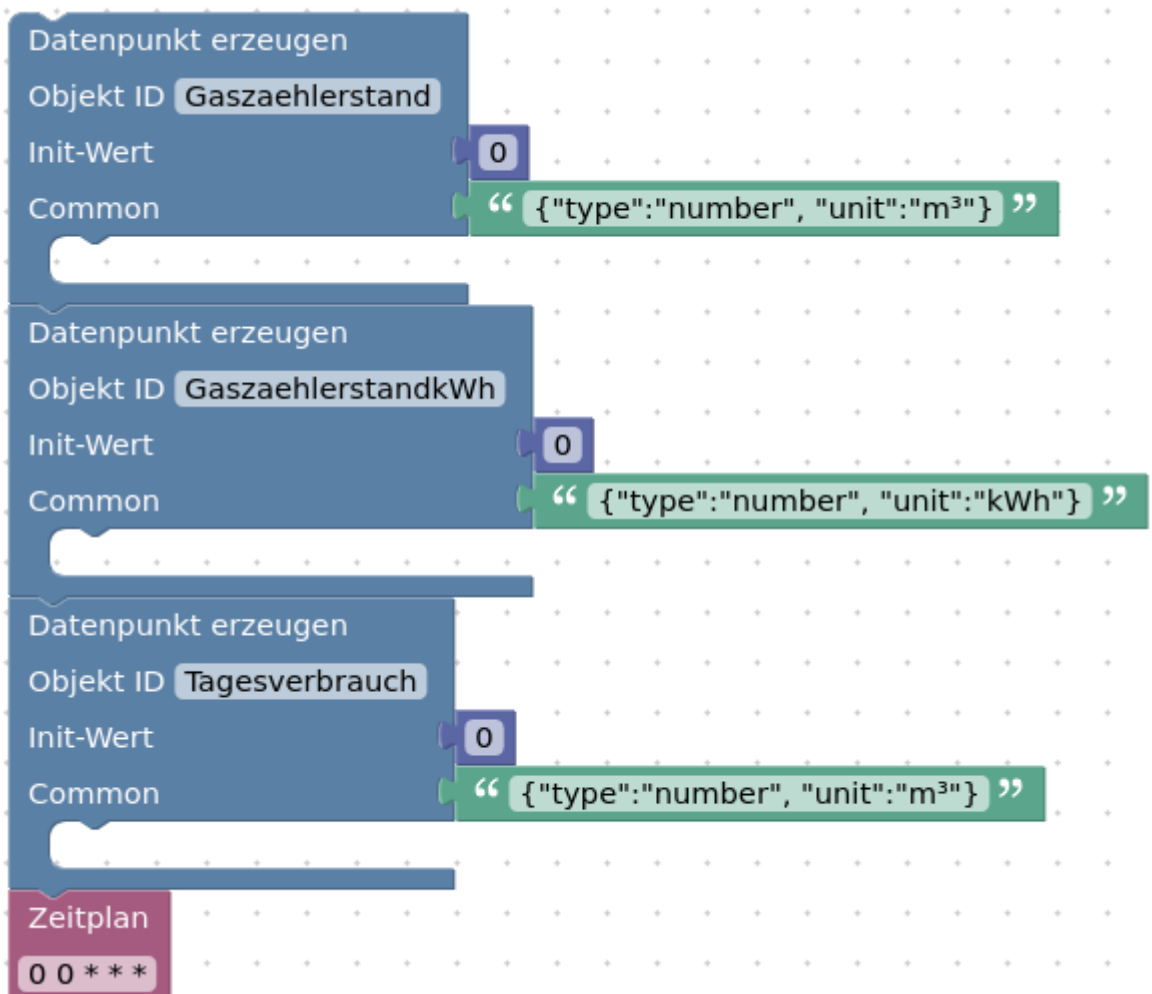
Etikette
Gaszähler

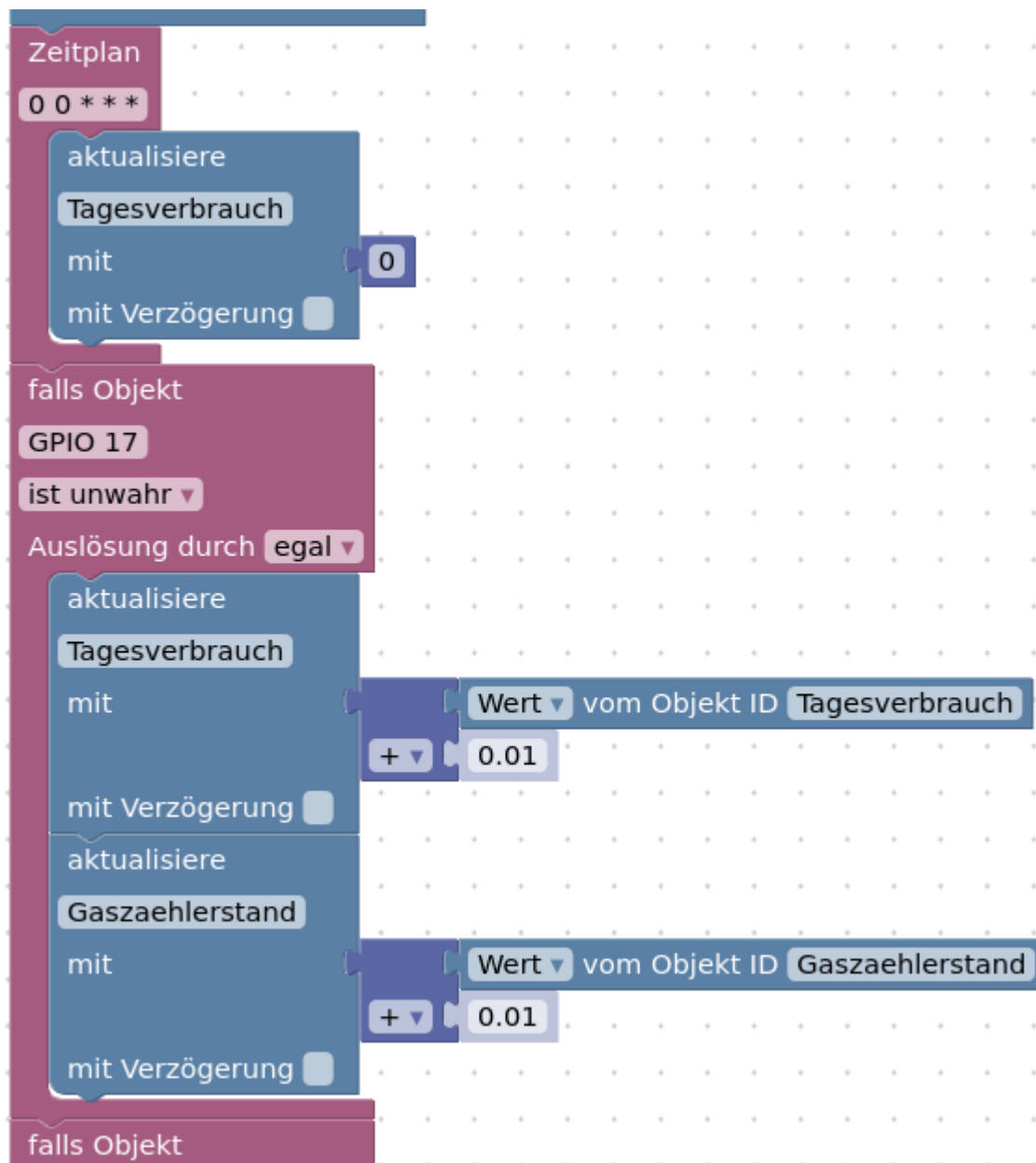
☐ Enable 18

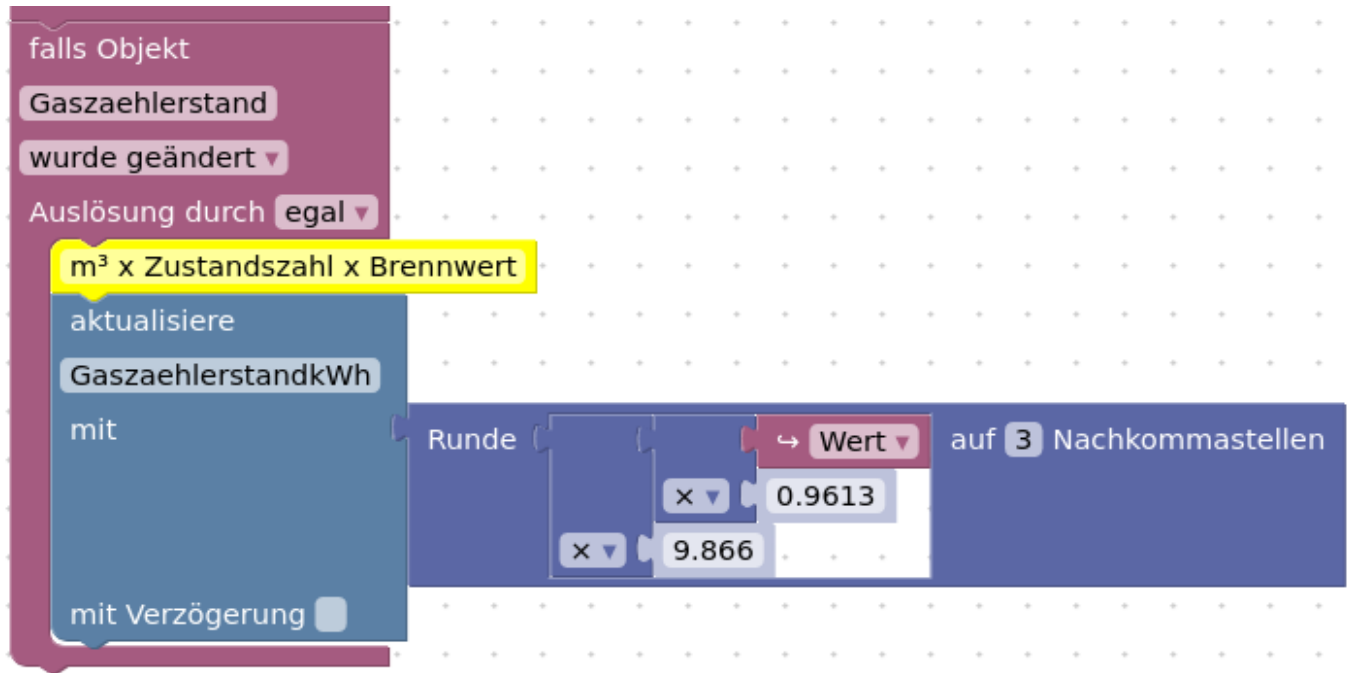
Typ
Eingang

Etikette

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);
});
```

//JTNDeG1sJTIweG1sbnMlM0QlMjJodHRwcyUzQSUyRiUyRmRldmVsb3BlcnMuZ29vZ2xlLmNvbS
UyRmJsb2NrbHkLMkZ4bWwLMjIlM0Ulm0NibG9jayUyMHR5cGUlM0QlMjJjcmVhdGULMjIlMjBpZC
UzRCUyMjBCJTdCJTdETGxSZCUyQ2QhQnI2NyU0MFZsYnYlMjIlMjB4JTNEJTIyMTEzJTIyJTIweS
UzRCUyMi0xMTIlMjIlM0Ulm0NmaWVsZCUyMG5hbWULM0QlMjJ0QU1FJTiyJTNFR2FzemFlaGxlcn
N0YW5kJTNDJTJGZmlbGQlM0Ulm0N2YWx1ZSUyMG5hbWULM0QlMjJWQUxVRSUyMiUzRSUzQ2Jsb2
NrJTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQlM0QlMjJ6Q01IWiUzQiUyNCUzRGV2c1
p+ZyUyMyUyNCUyNUiLMkNkKJTIyJTNFJTNDZmlbGQlMjBuYw1lJTNEJTIyTlVNJTIyJTNFMCUzQy
UyRmZpZWxkJTNFJTNDJTJGYmxvY2sLM0Ulm0MlMkZ2YWx1ZSUzRSUzQ3ZhbHVlJTIwbmFtZSUzRC
UyMkNPTU1PTiUyMiUzRSUzQ2Jsb2NrJTIwdHlwZSUzRCUyMnRleHQLMjIlMjBpZCUzRCUyMiUyQn
dhSlhpQmV+MWRuJTQwU05CWEQlNURiJTIyJTNFJTNDZmlbGQlMjBuYw1lJTNEJTIyVEVYVCUyMi
UzRSU3QiUyMnR5cGUlMjIlM0ElMjJudW1iZXIlMjIlMkMlMjAlMjJlbm0JTIyJTNBJTIySVdMi
VCMYUyMiU3RCUzQyUyRmZpZWxkJTNFJTNDJTJGYmxvY2sLM0Ulm0MlMkZ2YWx1ZSUzRSUzQ25leH
QlM0Ulm0NibG9jayUyMHR5cGUlM0QlMjJjcmVhdGULMjIlMjBpZCUzRCUyMjIlNUiLM0QlNUREK
Z3ZVV5JTVEeiu1RGZlUmIqTiUyMiUzRSUzQ2ZpZWxkJTIwbmFtZSUzRCUyMk5BTUULMjIlM0VHYX
N6YWVobGVyc3RhbRrV2glM0MlMkZmaWVsZCUzRSUzQ3ZhbHVlJTIwbmFtZSUzRCUyMlZBTfVFJT
IyJTNFJTNDYmxvY2sLMjB0eXBlJTNEJTIybWF0aF9udW1iZXIlMjIlMjBpZCUzRCUyMjYlMjN2a
F2NEpoaVklN0NtWiU2MHR2YTQlMkIlMjIlM0Ulm0NmaWVsZCUyMG5hbWULM0QlMjJ0VU0lMjIlM0
UwJTNDJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDdmFsdWULMjBuYw
1lJTNEJTIyQ09NTU90JTIyJTNFJTNDYmxvY2sLMjB0eXBlJTNEJTIydGV4dCUyMiUyMGlkJTNEJT
IybeQlN0RwNy5WJTI1JTNEIVMlM0FlJTJGKFBqMSU1QmULMjIlM0Ulm0NmaWVsZCUyMG5hbWULM0
QlMjJURVhUJTIyJTNFJTdCJTiydHlwZSUyMiUzQSUyMm5lbWJlciUyMiUyQyUyMCUyMnVuaXQLMj
IlM0ElMjJrV2glMjIlN0QlM0MlMkZmaWVsZCUzRSUzQyUyRmJsb2NrJTNFJTNDJTJGdmFsdWULM0
Ulm0NuZXh0JTNFJTNDYmxvY2sLMjB0eXBlJTNEJTIyY3JlYXRlJTIyJTIwaWQlM0QlMjJkdcU1RV
ULMjR3fiUyQzNSfilVUEpKUCUyQiUyNGMlMjIlM0Ulm0NmaWVsZCUyMG5hbWULM0QlMjJ0QU1FJT
IyJTNFVGFnZXN2ZXJicmF1Y2glM0MlMkZmaWVsZCUzRSUzQ3ZhbHVlJTIwbmFtZSUzRCUyMlZBTf
VFJTiyJTNFJTNDYmxvY2sLMjB0eXBlJTNEJTIybWF0aF9udW1iZXIlMjIlMjBpZCUzRCUyMm9iJT
NCXzA5ISU1RXVrZV8lM0JjJTQwVXF0Q1oLMjIlM0Ulm0NmaWVsZCUyMG5hbWULM0QlMjJ0VU0lMj
IlM0UwJTNDJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDdmFsdWULMj
BuYw1lJTNEJTIyQ09NTU90JTIyJTNFJTNDYmxvY2sLMjB0eXBlJTNEJTIydGV4dCUyMiUyMGlkJT
NEJTIyQnI3eSUzRCUyQ2RHQUY5SSU1RG1iJTNgei0lMkMlM0ElMjIlM0Ulm0NmaWVsZCUyMG5hbW
ULM0QlMjJURVhUJTIyJTNFJTdCJTiydHlwZSUyMiUzQSUyMm5lbWJlciUyMiUyQyUyMCUyMnVuaX
QLMjIlM0ElMjJtJUMyJUIzJTIyJTIeJTNdJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRn
ZhbHVlJTNFJTNDbmV4dCUzRSUzQ2Jsb2NrJTIwdHlwZSUzRCUyMnNjaGVkdWxkJTIyJTIwaWQlM0
QlMjIlMjBUWCUzRfPyJTDCLXcyY3ULM0RVaiFudU8qJTIyJTNFJTNDZmlbGQlMjBuYw1lJTNEJT
IyU0NIRURVTEULMjIlM0UwJTIwMCUyMColMjAqJTIwKiUzQyUyRmZpZWxkJTNFJTNDc3RhdGVtZW
50JTIwbmFtZSUzRCUyMlNUQVRFTUV0VCUyMiUzRSUzQ2Jsb2NrJTIwdHlwZSUzRCUyMnVwZGF0ZS
UyMiUyMGlkJTNEJTIyJTIwMHVGBkszcCU0MGQoYUZ0eQlN0MlMkZsYiUyMiUyMGlubGluZSUzRC
UyMmZhbHNlJTIyJTNFJTNDbXV0YXRpb24lMjB4bWxucyUzRCUyMmh0dHAlM0ElMkYlMkZ3d3cudz
Mub3JnJTGMTk50SUyRnhodG1sJTIyJTIwZGVsYXlfaW5wdXQLM0QlMjJmYWxzZSUyMiUzRSUzQy
UyRm1ldGF0aW9uJTNFJTNDZmlbGQlMjBuYw1lJTNEJTIyT0lEJTIyJTNFamF2YXNjcmlwdC4wLl
RhZ2VzdmVyYnJhdWNoJTNdJTJGZmlbGQlM0Ulm0NmaWVsZCUyMG5hbWULM0QlMjJXSvRlX0RfTE
FZJTIyJTNFRkFMU0Ulm0MlMkZmaWVsZCUzRSUzQ3ZhbHVlJTIwbmFtZSUzRCUyMlZBTfVFJTiyJT
NFJTNDYmxvY2sLMjB0eXBlJTNEJTIybWF0aF9udW1iZXIlMjIlMjBpZCUzRCUyMjQpciUyQ1glNj
BGZy4lNUJxJTNEdV9IKmxmUmsLMjIlM0Ulm0NmaWVsZCUyMG5hbWULM0QlMjJ0VU0lMjIlM0UwJT
NDJTJGZmlbGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDJTJGYmxvY2sLM0Ulm0
MlMkZzdGF0ZW1lbnQlM0Ulm0NuZXh0JTNFJTNDYmxvY2sLMjB0eXBlJTNEJTIyb24lMjIlMjBpZC
UzRCUyMmFGZiUyRiU0MCU1RSpsNEgpcCUyMyUyQ3U2RiU1RS4pJTIyJTNFJTNDZmlbGQlMjBuYw
1lJTNEJTIyT0lEJTIyJTNFcNBPmi4wLmdwaW8uMTcuc3RhdGULM0MlMkZmaWVsZCUzRSUzQ2ZpZW
xkJTIwbmFtZSUzRCUyMkNPTkRlVElPTiUyMiUzRWZhbHNlJTNdJTJGZmlbGQlM0Ulm0NmaWVsZC

UyMG5hbWULM0QLMjJBQ0tfQ090RElUSU90JTiyJTNFJTNDJTJGZml1bGQlM0ULM0NzdGF0ZWl1bn
QLMjBuYW11JTNEJTiyU1RBVEVNRU5UJTiyJTNFJTNDYmxvY2s1MjB0eXB1JTNEJTiydXBkYXR1JT
IyJTIwaWQlM0QLMjJ0ZHp+anc1nJBPTdEJTIOGklN0NNSEJUQm41JTiyJTIwaW5saW51JTNEJT
IyZmFsc2ULMjI1M0ULM0NtdXRhdGlubiUyMHhtbG5zJTNEJTiyHR0cCUzQSUYRiUyRnd3dy53My
5vcmlMkYx0Tk5JTJGeGh0bWw1MjI1MjBkZWxheV9pbN1dCUzRCUyMmZhbHN1JTiyJTNFJTNDJT
JGbxV0YXRpb241M0ULM0NmaWVsZCUyMG5hbWULM0QLMjJPSUQLMjI1M0VqYXZhc2NyaXB0LjAuVG
FnZXN2ZXJicmF1Y2glM0M1MkZmaWVsZCUzRSUzQ2ZpZWxkJTiwbmFtZSUzRCUyMldJVEhfREVMQV
klMjI1M0VGQUxTRSUzQyUyRmZpZWxkJTJNFJTNDdmFsdWULMjBuYW11JTNEJTiyVkfMVUULMjI1M0
ULM0NibG9jayUyMHR5cGULM0QLMjJtYXR0X2FyaXRobWV0aWMLMjI1MjBpZCUzRCUyMkRxt1R0cm
9NJTdDRkxXJTJGWGclMk1RJTQwa3I1MjI1MjBpbmxbpmlM0QLMjJmYWxzZSUyMiUzRSUzQ2ZpZW
xkJTiwbmFtZSUzRCUyMk9QJTiyJTNFQUREJTNDJTJGZml1bGQlM0ULM0N2YWx1ZSUyMG5hbWULM0
QLMjJBjTIyJTNFJTNDc2hhZG93JTiwHlwZSUzRCUyMm1hdGhfbnVtYmVyJTiyJTIwaWQlM0QLMj
I4NyU1RFpqJTIOryU1RC4qJTdCdHJXYTAqQmh+JTiyJTNFJTNDZml1bGQlMjBuYW11JTNEJTiyTl
VNJTiyJTNFMSUzQyUyRmZpZWxkJTJNFJTNDJTJGc2hhZG93JTJNFJTNDYmxvY2s1MjB0eXB1JTNEJT
IyZ2V0X3ZhbHVlJTiyJTIwaWQlM0QLMjJNOSU0ME94JTJD0GdGMCUzQUhDbYUyNHhfdCUyQiklMj
I1M0ULM0NmaWVsZCUyMG5hbWULM0QLMjJBVFRSJTiyJTNFdmF5JTNDJTJGZml1bGQlM0ULM0NmaW
VsZCUyMG5hbWULM0QLMjJPSUQLMjI1M0VqYXZhc2NyaXB0LjAuVGFnZXN2ZXJicmF1Y2glM0M1Mk
ZmaWVsZCUzRSUzQyUyRmJsb2NrJTJNFJTNDJTJGdmFsdWULM0ULM0N2YWx1ZSUyMG5hbWULM0QLMj
JCJTiyJTNFJTNDc2hhZG93JTiwHlwZSUzRCUyMm1hdGhfbnVtYmVyJTiyJTIwaWQlM0QLMjJXJT
VCJTdCKWNQJTdE0TByLWpDNFApJTQwUzFqJTiyJTNFJTNDZml1bGQlMjBuYW11JTNEJTiyTlVNJT
IyJTNFMC4wMSUzQyUyRmZpZWxkJTJNFJTNDJTJGc2hhZG93JTJNFJTNDJTJGdmFsdWULM0ULM0M1Mk
ZibG9jayUzRSUzQyUyRnZhbHVlJTJNFJTNDbmV4dCUzRSUzQ2Jsb2NrJTiwHlwZSUzRCUyMnVwZG
F0ZSUyMiUyMGlkJTNEJTiyI5Uk41MkZIR0VvdMJKJTJDc2FzJTNEQWw1MjI1MjBpbmxbpmlM0
QLMjJmYWxzZSUyMiUzRSUzQ211dGF0aW9uJTiwG1sbnM1M0QLMjJodHRwJTNBjTJGJTJGd3d3Ln
czLm9yZyUyRjE50Tk1MkZ4aHRtbCUyMiUyMGRlbgF5X2lucHV0JTNEJTiyZmFsc2ULMjI1M0ULM0
M1MkZtdXRhdGlubiUzRSUzQ2ZpZWxkJTiwbmFtZSUzRCUyMk9JRCUyMiUzRWphdmFzY3JpcHQuMC
5HYXN6YWVobGVyc3RhbmQlM0M1MkZmaWVsZCUzRSUzQ2ZpZWxkJTiwbmFtZSUzRCUyMldJVEhfRE
VMQVklMjI1M0VGQUxTRSUzQyUyRmZpZWxkJTJNFJTNDdmFsdWULMjBuYW11JTNEJTiyVkfMVUULMj
I1M0ULM0NibG9jayUyMHR5cGULM0QLMjJtYXR0X2FyaXRobWV0aWMLMjI1MjBpZCUzRCUyMiE4LU
lwcdDdc1ElNUQLM0QLM0R0SW5aVkkzJTiyJTIwaW5saW51JTNEJTiyZmFsc2ULMjI1M0ULM0NmaW
VsZCUyMG5hbWULM0QLMjJPUCUyMiUzRUFERCUzQyUyRmZpZWxkJTJNFJTNDdmFsdWULMjBuYW11JT
NEJTiyQSUYMiUzRSUzQ3NoYWRvdyUyMHR5cGULM0QLMjJtYXR0X251bWJ1ciUyMiUyMGlkJTNEJT
IyJTNEM2NCJTizZVpiLW01N0JCQWJKX0pCJTNEJTIOJTiyJTNFJTNDZml1bGQlMjBuYW11JTNEJT
IyTlVNJTiyJTNFMSUzQyUyRmZpZWxkJTJNFJTNDJTJGc2hhZG93JTJNFJTNDYmxvY2s1MjB0eXB1JT
NEJTiyZ2V0X3ZhbHVlJTiyJTIwaWQlM0QLMjIyJTJCXpXKkw1NDAlNjA5aCU0MGQwUC01MkM1Mk
Y0RiUyMiUzRSUzQ2ZpZWxkJTiwbmFtZSUzRCUyMkFUVFI1MjI1M0V2YWw1M0M1MkZmaWVsZCUzRS
UzQ2ZpZWxkJTiwbmFtZSUzRCUyMk9JRCUyMiUzRWphdmFzY3JpcHQuMC5HYXN6YWVobGVyc3Rhbm
QlM0M1MkZmaWVsZCUzRSUzQyUyRmJsb2NrJTJNFJTNDJTJGdmFsdWULM0ULM0N2YWx1ZSUyMG5hbW
ULM0QLMjJCJTiyJTNFJTNDc2hhZG93JTiwHlwZSUzRCUyMm1hdGhfbnVtYmVyJTiyJTIwaWQlM0
QLMjJxJTNBcDdGYXFzX2Y1N0JSdkkxX0oueDI1MjI1M0ULM0NmaWVsZCUyMG5hbWULM0QLMjJ0VU
01MjI1M0UwLjAxJTNDJTJGZml1bGQlM0ULM0M1MkZzaGFkb3clM0ULM0M1MkZ2YWx1ZSUzRSUzQy
UyRmJsb2NrJTJNFJTNDJTJGdmFsdWULM0ULM0M1MkZibG9jayUzRSUzQyUyRm5leHQLM0ULM0M1Mk
ZibG9jayUzRSUzQyUyRnN0YXRlbWVudCUzRSUzQ25leHQLM0ULM0NibG9jayUyMHR5cGULM0QLMj
JvbiUyMiUyMGlkJTNEJTiyN1dvbXclNjBxbF95dSU3Qmk5biglMkYLM0YzLSUYMiUzRSUzQ2ZpZW
xkJTiwbmFtZSUzRCUyMk9JRCUyMiUzRWphdmFzY3JpcHQuMC5HYXN6YWVobGVyc3RhbmQlM0M1Mk
ZmaWVsZCUzRSUzQ2ZpZWxkJTiwbmFtZSUzRCUyMkNPTkRjVElPTiUyMiUzRW51JTNDJTJGZml1bG
QlM0ULM0NmaWVsZCUyMG5hbWULM0QLMjJBQ0tfQ090RElUSU90JTiyJTNFJTNDJTJGZml1bGQlM0
ULM0NzdGF0ZWl1bnQLMjBuYW11JTNEJTiyU1RBVEVNRU5UJTiyJTNFJTNDYmxvY2s1MjB0eXB1JT
NEJTiyY29tbWVudCUyMiUyMGlkJTNEJTiyVC01MjI1MjRsJTJCUSFPA04LM0YuJTdCZCU3QmYzNF
klMjI1M0ULM0NmaWVsZCUyMG5hbWULM0QLMjJDT01NRU5UJTiyJTNFbSVDMiVCMYUyMHg1MjBadX
N0YW5kc3phaGw1MjB4JTiwQnJlbm53ZXJ0JTNDJTJGZml1bGQlM0ULM0NuZXh0JTJNFJTNDYmxvY2

sLmJb0eXBLJTNEJTIydXBkYXRlJTIyJTIwaWQlM0QlMjIpbY0lM0FnMkpPJTQwbVclN0NOUEMlMkNfLWdhJTIIyJTIwaW5saW5lJTNEJTIyZmFsc2UlMjIlM0Ulm0NtdXRhdGlviUyMHhtbG5zJTNEJTIyaHR0cCUzQSUyRiUyRnd3dy53My5vcmlMkYxOTk5JTJGeGh0bWwlmJIlMjBkZWxheV9pbNBlDcUzRCUyMmZhbHNlJTIyJTNFJTNDJTIJGbxV0YXRpb24lM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJPSUQlMjIlM0VqYXZhc2NyaXB0LjAuR2FzemFlaGxlcN0YW5ka1doJTNDJTIJGZml1bGQlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJXSvRlX0RfTEFZJTIyJTNFRkFMU0Ulm0MlMkZmaWVsZCUzRSUzQ3ZhbHVlJTIwbmFtZSUzRCUyMlZBTfVfJTIyJTNFJTNDYmxvY2sLmJb0eXBLJTNEJTIybWF0aF9ybmrMaXhlZCUyMiUyMGlkjJTNEJTIyazBLJTdDSElUVyU3Q3BWZGYtZyolMkNYN0glMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJuJTIyJTNFMyUzQyUyRmZpZWxkJTNFJTNDdmFsdWUlmJBUyW1lJTNEJTIyeCUyMiUzRSUzQ3NoYWRvdyUyMHR5cGUlM0QlMjJtYXR0X251bWJlciUyMiUyMGlkjJTNEJTIyVCU0MEM5JTdEVyUyNVJqMzN2RWpnTndaciU3QiUyMiUzRSUzQ2ZpZWxkJTIwbmFtZSUzRCUyMk5VTSUyMiUzRTMuMTIzNCUzQyUyRmZpZWxkJTNFJTNDJTIJGc2hhZG93JTNEJTNDYmxvY2sLmJb0eXBLJTNEJTIybWF0aF9hcml0aG1ldGljJTIyJTIwaWQlM0QlMjIlNjBzSUzQVhfIXhmWmdjcSU1QiUzQk5JT35NJTIIyJTIwaW5saW5lJTNEJTIyZmFsc2UlMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJPUCUyMiUzRU1VTFRJUEXZJTNDJTIJGZml1bGQlM0Ulm0N2YWx1ZSUyMG5hbWUlm0QlMjJBjTIyJTNFJTNDc2hhZG93JTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQlM0QlMjJtTFQlN0NpJTDcR2Uln0IlMjRQqMlM0QyQ2xqeUglMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJ0VU0lMjIlM0UxJTNDJTIJGZml1bGQlM0Ulm0MlMkZzaGFkb3clM0Ulm0NibG9jayUyMHR5cGUlM0QlMjJtYXR0X2FyaXRobWV0aWwlmJIlMjBpZCUzRCUyMldldCUzRGdVRkQlMjU0aDNGJTDcSDJxZSguJTIyJTIwaW5saW5lJTNEJTIyZmFsc2UlMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJPUCUyMiUzRU1VTFRJUEXZJTNDJTIJGZml1bGQlM0Ulm0N2YWx1ZSUyMG5hbWUlm0QlMjJBjTIyJTNFJTNDc2hhZG93JTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQlM0QlMjJYeTIlNjB0cCpleiUyM3UlnUVwJTNCJTQwcZJEKU0lMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJ0VU0lMjIlM0UxJTNDJTIJGZml1bGQlM0Ulm0MlMkZzaGFkb3clM0Ulm0NibG9jayUyMHR5cGUlM0QlMjJvbl9zb3VyY2UlmJIlMjBpZCUzRCUyMmJlJTIJGeW0lMCUyNSUyQ0thZXZwSDZFJTVCKjMlMjIlM0Ulm0NmaWVsZCUyMG5hbWUlm0QlMjJBVFRSJTIIyJTNFc3RhZGUudmFsJTNDJTIJGZml1bGQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDdmFsdWUlmJBUyW1lJTNEJTIyQiUyMiUzRSUzQ3NoYWRvdyUyMHR5cGUlM0QlMjJtYXR0X251bWJlciUyMiUyMGlkjJTNEJTIyMTduNDRzb2gtTUMubEItUHQlMjUlmJVoJTIyJTNFJTNDZml1bGQlMjBUyW1lJTNEJTIyTlVNJTIyJTNFMc45NjEzJTNDJTIJGZml1bGQlM0Ulm0MlMkZzaGFkb3clM0Ulm0MlMkZ2YWx1ZSUzRSUzQyUyRmJsb2NrJTNFJTNDJTIJGdmFsdWUlm0Ulm0N2YWx1ZSUyMG5hbWUlm0QlMjJCjTIyJTNFJTNDc2hhZG93JTIwdHlwZSUzRCUyMm1hdGhfbnVtYmVyJTIyJTIwaWQlM0QlMjJDJTDdEZ0hPJTVEJTVCMGQlN0R+aiU0MCU1QnQ2LlZnWCUyMiUzRSUzQ2ZpZWxkJTIwbmFtZSUzRCUyMk5VTSUyMiUzRTkuODY2JTNDJTIJGZml1bGQlM0Ulm0MlMkZzaGFkb3clM0Ulm0MlMkZ2YWx1ZSUzRSUzQyUyRmJsb2NrJTNFJTNDJTIJGdmFsdWUlm0Ulm0MlMkZibG9jayUzRSUzQyUyRnZhbHVlJTNFJTNDJTIJGYmxvY2sLm0Ulm0MlMkZuZXh0JTNFJTNDJTIJGYmxvY2sLm0Ulm0MlMkZzdGF0ZW1lbnQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRm5leHQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRm5leHQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRm5leHQlM0Ulm0MlMkZibG9jayUzRSUzQyUyRnhtbCUzRQ==

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

ioBroker v6.10.1

Übersicht Adapter Instanzen Objekte Aufzählungen Protokolle Benutzer Hosts Dateien Backup Skripte

common

JS backup_to_usb-stick

Gaszaehler

JS BACKUP_TO_USB-S...

```

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){

```

Last update:

2023/11/26
19:32

wiki:projekte:iobroker:uebersicht <https://www.von-thuelen.de/doku.php/wiki/projekte/iobroker/uebersicht?rev=1701027160>

```
    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://www.von-thuelen.de/> - Christophs DokuWiki

Permanent link:

<https://www.von-thuelen.de/doku.php/wiki/projekte/iobroker/uebersicht?rev=1701027160>

Last update: **2023/11/26 19:32**

