

Bedienungsanleitung CAN-Gateway Software

Stand: SW Version 29.330 / Android+Windows App 1.1.1

Update 1 (13.08.2023)

Inhaltsverzeichnis

1	Wichtige Hinweise – unbedingt durchlesen!	3
1.1	Bestimmungsgemäße Anwendung der Software.....	3
1.2	Gefahrenhinweise	4
1.3	Haftung des Softwareherstellers bzw. des Anbieters	5
1.4	Bindung der Software an eine durch den Kunden angegebene MAC Adresse	6
2	Einleitung.....	6
3	Benötigte Hardware und Aufbau	8
3.1	CAN-Gateway development board (Empfohlen)	8
3.2	CAN-Gateway Basis-Board (ohne uC).....	8
3.3	Einfacher Aufbau bestehend aus auf dem Markt verfügbaren vorgefertigten Modulen	9
3.4	Hardware „CAN-Gateway Version 5“	13
3.5	Nutzung ESP32-EVB(-EA) Entwicklungsboards von Olimex als CAN-Gateway.....	14
4	Software flashen.....	15
4.1	Automatisiert (empfohlen).....	15
4.2	Automatisiert (empfohlen), nur für CAN-Gateway development board	15
4.3	Manuell.....	15
5	Erste Inbetriebnahme.....	20
6	WEB-Interface: Hauptseite	24
7	Allgemeine Informationen zu Geräte-Parametern	25
7.1	Beispiel	26
8	User-Levels	27
9	WEB-Interface: Einstellungen.....	28
10	WEB-Interface: Parameter Einstellungen.....	29
10.1	Datentypen.....	31
10.2	Hinweise zu Listen	32
10.3	Beispiele	32
10.3.1	Beispiel 1: Komfortlüftung – Feuchte Sollwert auslesen und einstellen.....	32

10.3.2	Beispiel 2: Komfortlüftung – Temperatur Abluft.....	32
10.3.3	Beispiel 3: Komfortlüftung – Temperatur Außenluft	32
10.3.4	Beispiel 4: Raumbedienmodul – Kennwort für UserLevel 7 lesen/ändern	32
10.3.5	Beispiel 5: Komfortlüftung – Betriebsmodus	33
10.3.6	Beispiel 6: Komfortlüftung – Datum/Uhrzeit	33
11	WEB-Interface: Parameter-Konfiguration Generator	34
12	WEB-Interface: Liste der Parameter.....	35
13	WEB-Interface: Liste der Geräte.....	37
14	WEB-Interface: Status Informationen	38
15	WEB-Interface: SD Karte Info/Status.....	39
16	WEB-Interface: CAN Datenlogger.....	41
17	WEB-Interface: Parameter Logger.....	42
18	WEB-Interface: KNX IP Einstellungen (ab SW 29.001)	43
19	WEB-Interface: Software Update	45
20	Übertragung der Parameter über MQTT Protokoll.....	46
21	Übertragung der Parameter über Modbus TCP	47
21.1	Hinweise zu Modbus TCP Kommunikation.....	47
21.2	Modbus Adressen.....	47
21.3	Register Typen	47
21.4	Übertragungswerte	48
22	Steuerung über Bedienmodul	49
23	REST-API	51
24	Besondere Hinweise zu Parameteränderungen.....	51
25	Android App: MyHome App für Hoval Geräte	52
26	Windows App: MyHome App für Hoval Geräte	52
27	Fernzugriff auf CAN-Gateway über Internet (ab SW Version \geq 28.100)	52
27.1	Einrichtung des Zugangs.....	53
27.2	Details zur Verschlüsselung.....	56
28	Hilfe!	56
28.1	Netzwerkname/Schlüssel sind falsch eingestellt, Web-Interface nicht zugänglich. Was tun?	
	56	
28.1.1	Mit Bray Terminal	57
28.1.2	Mit HTerm	58
28.2	CAN-Gateway startet immer mit Werkeinstellungen. Einstellungen können nicht gespeichert werden.....	60
29	Hinweise zu HomeVent Komfortlüftung mit dem einfachen Bediengerät BG02E.....	61

30	Zeiteinstellung in CAN-Gateway [neu ab SW 22.001]	62
31	Autorecovery	62
32	Einbindung in Home Assistant.....	62
32.1	Einleitung.....	62
32.2	Konfiguration.....	63
32.2.1	Automatische Konfiguration mit MQTT Discovery.....	63
32.2.2	Manuelle Konfiguration.....	65
33	Einbindung in OpenHab.....	68
34	Einbindung in Loxone	68
34.1	Einbindung als virtuelle HTTP Eingänge und Ausgänge	68
34.2	Einbindung als Modbus TCP Gerät	73
35	Einbindung in ioBroker	76
35.1	Problemlösungen und Hinweise für Fortgeschrittene ioBroker Anwender.....	78
36	Einschränkungen der Demo-Version.....	78
37	Over-The-Air Update	79
38	Lizenzen	79

1 Wichtige Hinweise – unbedingt durchlesen!

1.1 Bestimmungsgemäße Anwendung der Software

Diese Software ist dafür bestimmt, Geräte der Firma Hoval AG, die mit einer TopTronic® E Regelung¹ ausgestattet sind (im folgenden TTE Geräte genannt), auszulesen und zu steuern.

Auslesen bedeutet in diesem Sinne, dass die Parameter und Informationen aus TTE Geräten über den Geräte-CAN-Bus ausgelesen und:

1. auf der CAN-Gateway Benutzeroberfläche dem Nutzer dargestellt werden können,
2. über die in der Software zur Verfügung stehenden Kommunikationsprotokolle (REST API, MQTT, KNX IP (Routing), Modbus TCP) an andere mit diesen Kommunikationsprotokollen kompatiblen Geräten übertragen werden können,
3. auf einer Micro-SD Karte gespeichert werden können.

Steuern im obigen Sinne bedeutet, dass diese Software die Änderung der Parameter und der Informationen, die über CAN-Bus in TTE-Geräten geändert werden können, bewirken kann. Diese Änderung kann:

1. direkt durch den Nutzer über die Benutzeroberfläche oder Android App oder Windows App eingesteuert werden

¹ TopTronic ist die geschützte und eingetragene Marke der Firma Hoval AG

2. durch ein mit dieser Software über ein der oben aufgelisteten Kommunikationsprotokollen verbundenes Gerät eingesteuert werden.

Die CAN-Gateway Software beinhaltet keinerlei Logik, die selbständig Änderungen der Parameter und Informationen in TTE Geräten initiiert.

Die oben beschriebenen Funktionalitäten der Software sollen ermöglichen die TTE-Geräte in ein Hausautomatisierungssystem einzubinden.

Welche Parameter und Informationen ausgelesen und/oder geändert werden können, hängt von dem jeweiligen TTE Gerät ab. Vertraglich garantiert wird die Funktionalität für alle Parameter, die in der folgenden Datei der Firma Hoval AG aufgelistet sind: <http://www.hoval.com/misc/TTE/TTE-GW-Modbus-datapoints.xlsx>. In der Regel ist zusätzlich Auslesen aller Parameter, die über ein TTE Bedienmodul im Menu „Service“ bei allen User Levels angezeigt werden, möglich, wird jedoch vertraglich nicht garantiert.

Die Software kann angewendet werden, indem sie auf einem kompatiblen Mikrocontroller ausgeführt wird. Kompatibel sind ESP32 Dual-Core Mikrocontroller der Firma Espressif (<https://www.espressif.com/>). Zudem muss zwingend eine kompatible Beschaltung des Mikrocontrollers vorhanden sein. Diese Beschaltung muss unter anderem die notwendige Stromversorgung des Mikrocontrollers sowie die Anbindung der physikalischen Schnittstellen wie CAN-Bus und Ethernet ermöglichen. Für die Nutzung der SD-Karte Funktionalität muss an den Mikrocontroller ein SD-Kartenslot angeschlossen werden. Die Beschaltung muss angelehnt an eine der drei Referenzbeispiele erfolgen:

- a) CAN-Gateway Hardware V5, wie in Bedienungsanleitung beschrieben
- b) CAN-Gateway Development Board, siehe http://wled.shop/wp-content/uploads/2021/11/CAN_Gateway_Nutzungsinformationen.pdf
- c) Olimex ESP32-EVB(-EA) Entwicklungsboard Rev. I, siehe <https://www.olimex.com/Products/IoT/ESP32/ESP32-EVB/open-source-hardware>

Entsprechen dieser Referenzbeispiele wird die Software in drei Varianten a, b und c angeboten. Der Kunde muss bei der Bestellung sich für eine der drei Varianten entscheiden.

Aus Performance-Gründen ist es möglich gleichzeitig nur ein Kommunikationsprotokoll (MQTT, KNX IP, REST-API) oder Datenspeicherung auf SD-Karte zu verwenden / zu aktivieren. Eine parallele Verwendung ist zwar theoretisch möglich, kann aber zu Performance Problemen führen und wird daher nicht garantiert.

1.2 Gefahrenhinweise

WARNUNG: Diese Software basiert auf der Analyse der Kommunikation zwischen TTE Geräten und nicht auf offiziellen Informationen des Geräteherstellers. Es muss angenommen werden, dass die Analyse nicht vollständig, fehlerhaft und lückenhaft ist sowie nicht auf alle Geräte und deren Versionen, vor allem zukünftige Versionen, anwendbar sein kann. Daraus folgt, dass bei sicherheitskritischen TTE Geräten, bei denen Fehler in der Ansteuerung zu Beschädigungen von Personen, Umwelt, Geräten, Gebäuden etc. führen können, der Einsatz dieser Software nur beaufsichtigt erfolgen muss. Beaufsichtigt bedeutet, dass dieses Gerät und alle damit verbundenen TTE Geräte ständig von einer Person überwacht werden müssen, die aufgrund Ihrer Ausbildung oder Erfahrung in der Lage ist Fehler zu erkennen und alle TTE Geräte notfalls innerhalb weniger Sekunden

abzuschalten. Das betrifft insbesondere ausdrücklich, aber nicht ausschließlich, Heizungsgeräte mit Verbrennung.

HINWEIS: Der Hersteller Hoval AG vertreibt selbst Gateways für die Anbindung seiner TTE Geräten an die Hausautomatisierungssysteme unter anderem über KNX oder ModBus RTU/TCP Busse/Protokolle. Der Anbieter dieser CAN-Gateway Software empfiehlt diese Gateways der Firma Hoval AG für den produktiven Einsatz bei sicherheitskritischen Systemen zu verwenden.

HINWEIS: Die Verwendung dieser Software in Verbund mit TTE Geräten kann eventuell dazu führen, dass die Garantie- und Gewährleistungsansprüche für diese TTE Geräte sich erlöschten. Der Benutzer dieser Software ist selbst dafür zuständig das Vorhandensein entsprechender Klauseln in Dokumentationen zu seinem TTE Gerät und die gesetzliche Lage dazu in seinem Land zu prüfen.

WARNUNG: Diese Software erlaubt Parameter der TTE Geräte zu ändern. Diese Software prüft keinerlei ob die Änderung der Parameter sinnvoll und ungefährlich für den Betrieb des TTE Gerätes und anderer damit verbundenen TTE Geräten und sonstigen Anlagen, für Nutzer und Bedienpersonal dieser TTE Geräte und Anlagen, für die Umwelt sowie das Gebäude, wo diese Geräte und damit verbundene Anlagen installiert sind, ist. Als Nutzer der Software müssen Sie mit Parameteränderungen sehr aufmerksam umgehen und nur solche Parametrierungen zulassen und direkt oder indirekt einsteuern, deren Sinn Sie zu 100% verstehen und deren Ungefährlichkeit und Unschädlichkeit für den Betrieb des TTE Gerätes und anderer damit verbundenen TTE Geräten und sonstigen Anlagen, für Nutzer und Bedienpersonal dieser TTE Geräte und Anlagen, für die Umwelt sowie das Gebäude, wo diese Geräte und damit verbundene Anlagen installiert sind zu 100% gegeben ist.

WARNUNG: Wenn Sie den Mikrocontroller, auf dem diese Software installiert ist, in Ihrem WLAN oder LAN Netzwerk verwenden, müssen Sie dafür sorgen, dass keine fremden Personen den Zugang zu diesem Netzwerk bekommen, da alle Personen mit dem Zugang zu Ihrem Netzwerk auf CAN-Gateway uneingeschränkt zugreifen können.

WARNUNG: Der Mikrocontroller, auf dem diese Software installiert ist, muss physikalisch für fremde Personen unzugänglich sein. Alle Personen, die physikalisch auf den Mikrocontroller zugreifen können, können die drauf gespeicherten Informationen inkl. Ihre Einstellungen und Passwörter ohne großen Aufwand auslesen.

HINWEIS: Jede andere Verwendung, als hier beschriebene, ist nicht bestimmungsgemäß und führt zu Gewährleistungs- und Haftungsausschluss.

HINWEIS: Jede Änderung und Modifikation der Software durch den Anwender führt zu Gewährleistungs- und Haftungsausschluss des Anbieters.

1.3 Haftung des Softwareherstellers bzw. des Anbieters

Für Mängel der CAN-Gateway Software haftet der Anbieter nach Maßgabe der gesetzlichen Bestimmungen des Kaufrechts (§§ 434 ff. BGB). Bei leichter Fahrlässigkeit haftet der Anbieter nur bei Verletzung vertragswesentlicher Pflichten (Kardinalpflichten) sowie bei Personenschäden nach Maßgabe des Produkthaftungsgesetzes. Im Übrigen ist die vorvertragliche, vertragliche und außervertragliche Haftung des Anbieters auf Vorsatz und grobe Fahrlässigkeit beschränkt, wobei die Haftungsbegrenzung auch im Falle des Verschuldens eines Erfüllungsgehilfen des Anbieters gilt.

Die Haftung des Anbieters ist in folgenden Fällen vollständig ausgeschlossen:

1. Ein Fehler und die daraus folgenden direkten und/oder indirekten Schäden sind durch eine fehlerhafte Ausführung der Software z.B. aufgrund fehlerhafter Hardware oder aufgrund äußerer Einflüsse auf die Hardware (mechanischer, elektrischer, elektromagnetsicher oder anderer Art) entstanden.
2. Ein Fehler und die daraus folgenden direkten und/oder indirekten Schäden sind durch eine fehlerhafte Bedienung der Software, inklusiv z.B. die Einstellung nicht sinnvoller, falscher oder sogar gefährlicher Parameteränderungen durch den Nutzer entstanden. Das beinhaltet eine direkte Steuerung durch die Benutzeroberfläche, über Android oder Windows App, oder eine indirekte Steuerung, indem der Nutzer andere Geräte so programmiert oder eingestellt hat, dass sie die Änderung der Parameter durch ein Kommunikationsprotokoll über CAN-Gateway Software bewirken haben.
3. Ein Fehler und die daraus folgenden direkten und/oder indirekten Schäden sind durch andere Gründe, die der Anbieter der Software nicht zu vertreten hat, entstanden.

1.4 Bindung der Software an eine durch den Kunden angegebene MAC Adresse

Diese Software, ausgenommen die Demoversion, wird an eine durch den Kunden angegebene MAC Adresse des WiFi Moduls des ESP32 Mikrocontrollers gebunden. Das bedeutet, dass die Software nur auf diesem Mikrocontroller mit dieser MAC Adresse ihren vollen Funktionsumfang hat.

Der Kunde hat Recht die Software auf einen anderen Mikrocontroller mit einer anderen MAC-Adresse zu übertragen (im Folgenden Übertragung genannt). Dazu muss er sich mit dem Anbieter per Email, telefonisch oder auf dem Postweg in Verbindung setzen und die Übertragung beantragen.

Dabei ist die MAC Adresse des alten Mikrocontrollers und die MAC Adresse des neuen Mikrocontrollers (jeweils des WiFi Moduls) anzugeben. Der Anbieter wird die Anfrage innerhalb von 10 Arbeitstagen bearbeiten. Nach Inbetriebnahme der Software auf dem neuen Mikrocontroller muss der Kunde die Software auf dem alten Mikrocontroller sowie alle bei ihm ggf. noch vorhandenen Sicherungskopien der Software vollständig löschen.

Wird die Übertragung der Software von einem Kunden öfter als einmal pro Jahr beantragt, muss er dafür eine plausible Erklärung abgeben. Bei einem begründeten Verdacht auf Missbrauch und Verletzung der Urheberrechte des Anbieters, kann der Anbieter die Übertragung der Software verweigern.

2 Einleitung

Diese Software erlaubt es die Geräte der Serie „TopTronic® E“ der Firma **Hoval Aktiengesellschaft** an das Hausautomatisierungssystem anzubinden. Diese Geräte werden im Folgenden einfach „Geräte“ genannt.

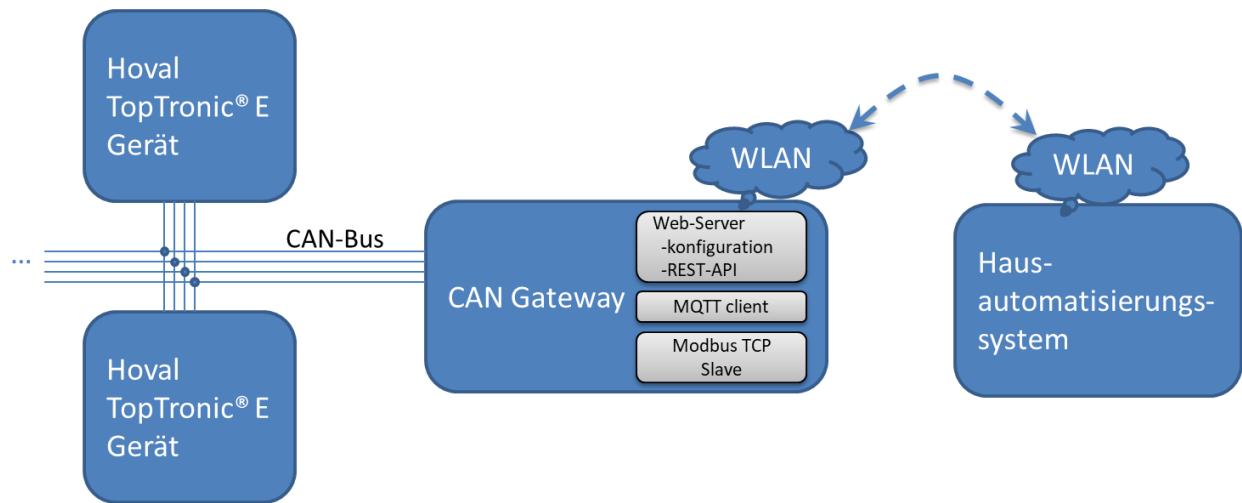
Die Geräte der Serie „TopTronic® E“ der Firma Hoval Aktiengesellschaft besitzen einen CAN-Bus Anschluss. Dieser wird für die Verbindung der Geräte untereinander verwendet. CAN-Gateway kann an dieses CAN-Bus angeschlossen werden und kann die auf dem Bus übertragenen Daten entschlüsseln und selbst die Daten auf dem Bus generieren, so dass damit Auslesen der Informationen aus den Geräten sowie die Steuerung der Geräte möglich ist. Die Schnittstelle für CAN-Bus ist bei den Geräten physikalisch als RJ45 Buchse oder Schraubklemme ausgeführt und

beinhaltet außerdem 12V Versorgungspins, die man für Kleinverbraucher (wie CAN-Gateway) nutzen kann.

Diese Software selber muss dafür auf einem ESP32 Dual-Core Mikrocontroller installiert werden. Die genaue Beschreibung der benötigten Hardwarekomponenten und der Aufbau sind im nächsten Kapitel gegeben. Der so aufgebaute ESP32 basiertes Board mit installierter Software wird im Folgenden als „CAN-Gateway“ bezeichnet.

ESP32 Mikrocontroller besitzt einen WLAN Interface, so dass das Hausautomatisierungssystem über WLAN mit dem CAN-Gateway verbunden werden kann. Als Datenübertragungsprotokolle zwischen CAN-Gateway und dem Hausautomatisierungssystem können wahlweise MQTT Protokoll, Modbus TCP Protokoll, KNX IP Protokoll oder REST-API verwendet werden, wobei MQTT Protokoll empfohlen wird und Modbus TCP Protokoll nur mit einigen Einschränkungen funktioniert.

Über WLAN Interface kann CAN-Gateway mithilfe eines Internet Browsers konfiguriert werden.



Zusätzlich hat die Software folgende Funktionen:

- CAN Daten Logger (Rohdaten) über WEB-Interface
- Ausgabe aller Geräte mit Ihrem Typ/Adresse, die auf dem CAN-Bus kommunizieren
- Ausgabe aller auf dem CAN-Bus übertragener Parameter

3 Benötigte Hardware und Aufbau

Es stehen verschiedene Optionen zur Verfügung:

Variante	Empfehlung?	Gesamte Hardware Kosten (inkl. Versand nach Deutschland, Preise sind inkl. Versand, Stand 15.2.2022, Preise können variieren)	Vorteile / Nachteile
CAN-Gateway development board von https://wled.shop	Ja	ca. 52 €	Vollständig aufgebaut und getestet. EU-weit lieferbar.
CAN-Gateway Basis-Board (ohne uC) + Gehäuse (https://shop.myhome-control.de) + ESP32 DevKit Board (diverse Anbieter)	Ja	ca. 44 €	Leichter Aufbau (Einstecken, Verschrauben; kein Löten etc.), Schnittstellen auf Funktion getestet, BG02E Interface als bestückte Option wird angeboten. Nur innerhalb Deutschalnd lieferbar.
Olimex ESP32-EVB(-EA) development Board inkl. Gehäuse	Bedingt	ca. 50 €	Kein BG02E Interface möglich, CAN Anschluss: nur Schraubklemmen. Keine Stromversorgung über Hoval CAN-Bus möglich. Als eine Version mit externer Antenne verfügbar. Weltweit lieferbar.
Eigenbau bestehend aus auf dem Markt verfügbaren vorgefertigten Modulen, Referenzdesign V5	Nein	ca. 15 € bis 30 € je nach Aufbau etc.	Löten etc. notwendig, Tiefes Elektronik Verständnis notwendig, keine Ethernet Schnittstelle möglich.

Im Weiteren sind diese Optionen detaillierter beschrieben.

3.1 CAN-Gateway development board (Empfohlen)

Siehe http://wled.shop/wp-content/uploads/2021/11/CAN_Gateway_Nutzungsinformationen.pdf

Das ist die empfohlene Variante. Die CAN-Gateway Software kann auf diesem Board sehr leicht mit weniger Klicks ohne spezielle Kenntnisse installiert werden. Hat Ethernet Schnittstelle, "Plug-and-Play" CAN RJ45 Schnittstellen, Mikro-SD Slot, USB Interface, Gehäuse, DIP-Switch für CAN-Terminierung. Also nur Vorteile. **Achtung:** Zusatz-CAN-Interface bei diesem Board ist per default nicht bestückt: also spezieller BG02E Interface nur mit Zusatzbestückung möglich!

3.2 CAN-Gateway Basis-Board (ohne uC)

Siehe <https://shop.myhome-control.de/CAN-Gateway-Basis-Board-ohne-ESP32-uC/HW10001.2>

Das ist eine weitere empfohlene Option. Zusätzlich wird ein ESP32 DevKit Board (30-Pin-Variante, diverse Anbieter) benötigt. Das Board verfügt über eine Ethernet-Schnittstelle, „Plug-and-Play“ CAN RJ45-Schnittstellen, Micro-SD-Slot, USB-Buchse (nur zur Stromversorgung, keine Daten, Datenschnittstelle über ESP32 DevKit Board!), Gehäuse, DIP-Schalter für CAN-Terminierung . Zusatzausstattung für die zweite CAN-Schnittstelle ist gegen Aufpreis erhältlich (nur relevant für BG02E-Schnittstelle).

Die SW-Variante ist die gleiche wie beim CAN-Gateway-Entwicklungsboard von wled.shop.

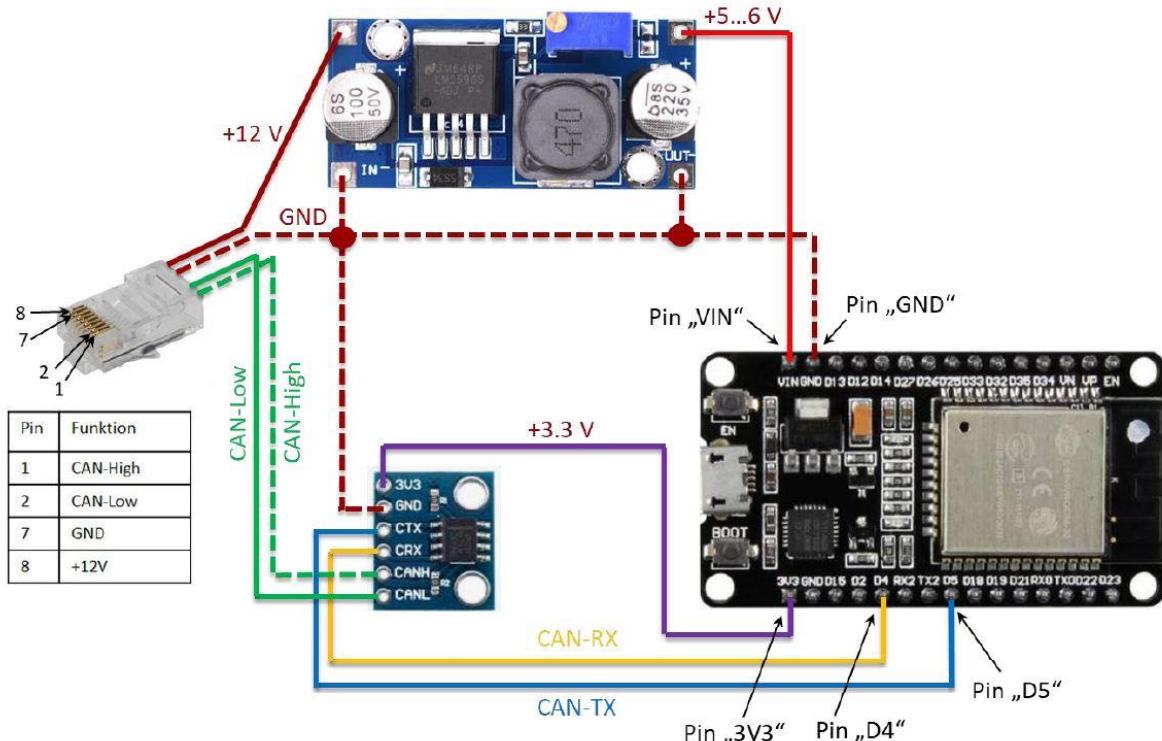
3.3 Einfacher Aufbau bestehend aus auf dem Markt verfügbaren vorgefertigten Modulen

Es werden folgenden Hardware-Komponenten benötigt:

Komponente	Preis, ca. inkl. Versand	Beispiel Bilder
ESP32 DevKit Board mit einem ESP-WROOM-32 Modul (kein ESP32S !) mit 4MB Flash Im Folgenden wird 30-Pin Version in Abbildungen und Layout verwendet. Andere Versionen (wie z.B. DevKitC) können auch verwendet werden, haben aber anderes Pinning.	5-10 €	
SN65HVD230 CAN Transceiver Board (Achtung: viele China-Teile sind fake und funktionieren nicht oder nicht stabil)	1,5-5 €	
LM2596 Modul für die Spannungsversorgung (als 12 V zu 5 V Konverter)	1,2-5 €	
Ethernet Kabel mit RJ45 Stecker, abgeschnitten (oder Buchse, wenn man das Ganze auf einer Platine aufbaut)	2-5 €	

Achtung! Es wird zwar Ethernet Kabel verwendet, das hat aber nichts mit der normalen LAN Kommunikation zu tun. Das Kabel darf nicht in einen Router oder ein PC eingesteckt werden!

Der Aufbau ist wie folgt. **Achtung: LM2596 Modul sollte man zuerst mit Hilfe eines Multimeters auf die Ausgangsspannung von ca. 5 V einstellen.** Dafür dreht man die Schraube an dem einstellbaren Widerstand (hellblau im Bild). ESP32 Modul kann zwar kurzzeitig auch mit 12 V funktionieren, wird jedoch sehr warm und kann auf Dauer beschädigt werden.



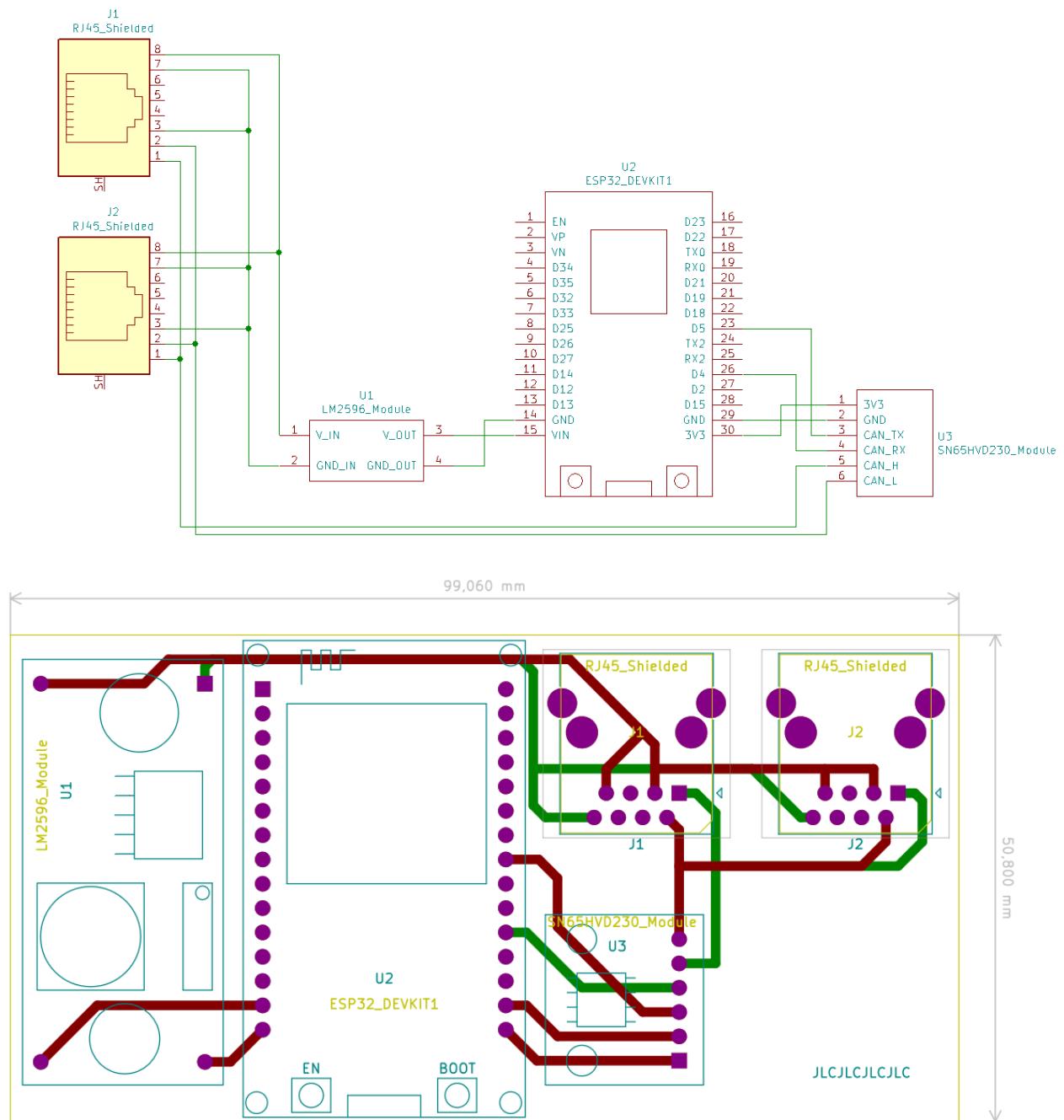
Wichtige Hinweise

- 1) GND Pin an RJ45: es kann sein, dass an einigen Geräten nur Pin3 und nicht Pin7, oder beide Pins 3 und 7 als GND verwendet werden. Ich würde empfehlen beide Pins (3 und 7) zusammen mit GND zu verbinden.
- 2) zum CAN-Bus: CAN-Bus muss beidseitig terminiert werden. Die Geräte von Hoval sind normalerweise bereits terminiert. SN65VHD230 Boards sind normalerweise auch terminiert. Also den Aufbau wie hier kann man normalerweise direkt in die RJ45 Buchse vom Gerät einstecken. Wenn man bereits mehrere Geräte in einem gemeinsamen CAN-Bus hat (z.B. Lüftungsgerät und ein Bedienmodul dazu), dann darf man den Aufbau wie oben nicht direkt verwenden:
 - Zu einem hat man dann vermutlich keine freie Buchse, wo man RJ45 Stecker von CAN-Gateway reinstecken kann. Dafür kann man z.B. am CAN-Gateway noch einen zweiten RJ-45 Stecker dazu anlöten und dann einen davon z.B. in das Lüftungsgerät und den anderen in den Bedienmodul einstecken.
 - Zu anderem muss man ein Terminierwiderstand entfernen. Am Bedienmodul kann man auch Terminierwiderstand mit zwei DIP-Schaltern auf der Rückseite abschalten. Andere Geräte haben oft diese DIP-Schalter nahe zum CAN-Anschluss und markiert als „R-CAN“. Im Zweifelsfall muss man mit einem Multimeter den Widerstand zwischen CAN-High und CAN-Low Leitungen messen. Es muss ca. 60 Ohm sein, wenn alle Geräte am CAN-Bus angeschlossen sind. Ist der Widerstand deutlich kleiner als 60 Ohm, dann ist CAN „überterminiert“ und man muss eine oder mehrere Terminierwiderstände entfernen und dann noch mal messen. Ist der Widerstand deutlich größer als 60 Ohm, dann ist CAN

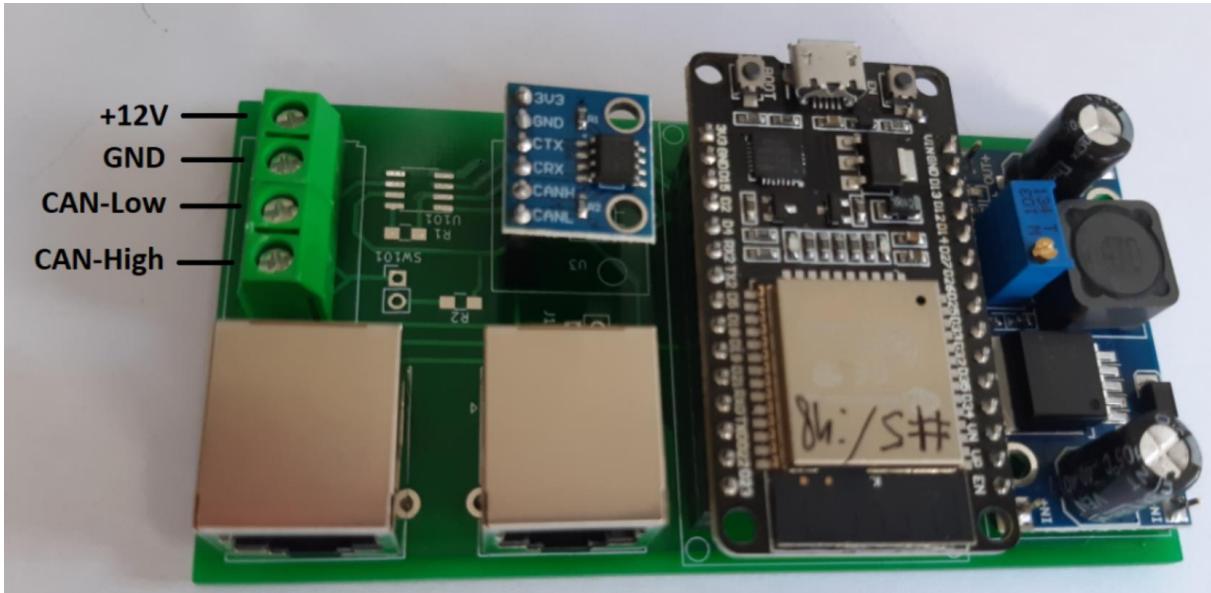
„unterterminiert“ und man muss eine oder mehrere Terminierwiderstände dazu schalten und dann noch mal messen. Für mehr Details dazu lesen Sie bitte die Beschreibung zum CAN-Busen (z.B. auf Wikipedia etc.)

- Stromversorgung: man kann CAN-Gateway auf zwei alternativen Wegen versorgen:
 - Über 12 V Leitung des Hoval CAN-Busses (empfohlen). Alle Hoval-Hauptsteuergeräte stellen 12 V zur Verfügung. Es dient z.B. zur Stromversorgung der Raumbedienmodule. CAN-Gateway kann sich auch aus dieser 12 V Leitung versorgen und verbraucht nicht mehr als ein Raumbedienmodul. Vorteil: man braucht keine separate Stromversorgung.
 - Über Micro-USB Anschluss des ESP32 Boards (optional). Hat man bereits viele Raumbedienmodule, die sich aus 12 V CAN-Bus Leitung versorgen (bitte prüfen Sie Ihre Hoval Dokumentation wie viele Module maximal versorgt werden können), kann man CAN-Gateway über ein Micro-USB Netzteil (5 V / 0,5 A reichen, bitte nur gute qualitative Teile verwenden) versorgen. In diesem Fall darf man die 12 V Leitung zu Hoval Geräten nicht verwenden, also z.B. die entsprechende Litze im RJ-45 Kabel trennen. GND-Leitung muss aber (neben CAN-High und CAN-Low natürlich) unbedingt verbunden bleiben.

Man kann auch eine RJ45 Buchse verwenden und die einzelnen Module auf einer PCB z.B. so einordnen (hier Screenshots aus KiCad). Hier ist ein Beispiel mit zwei RJ45 Buchsen.

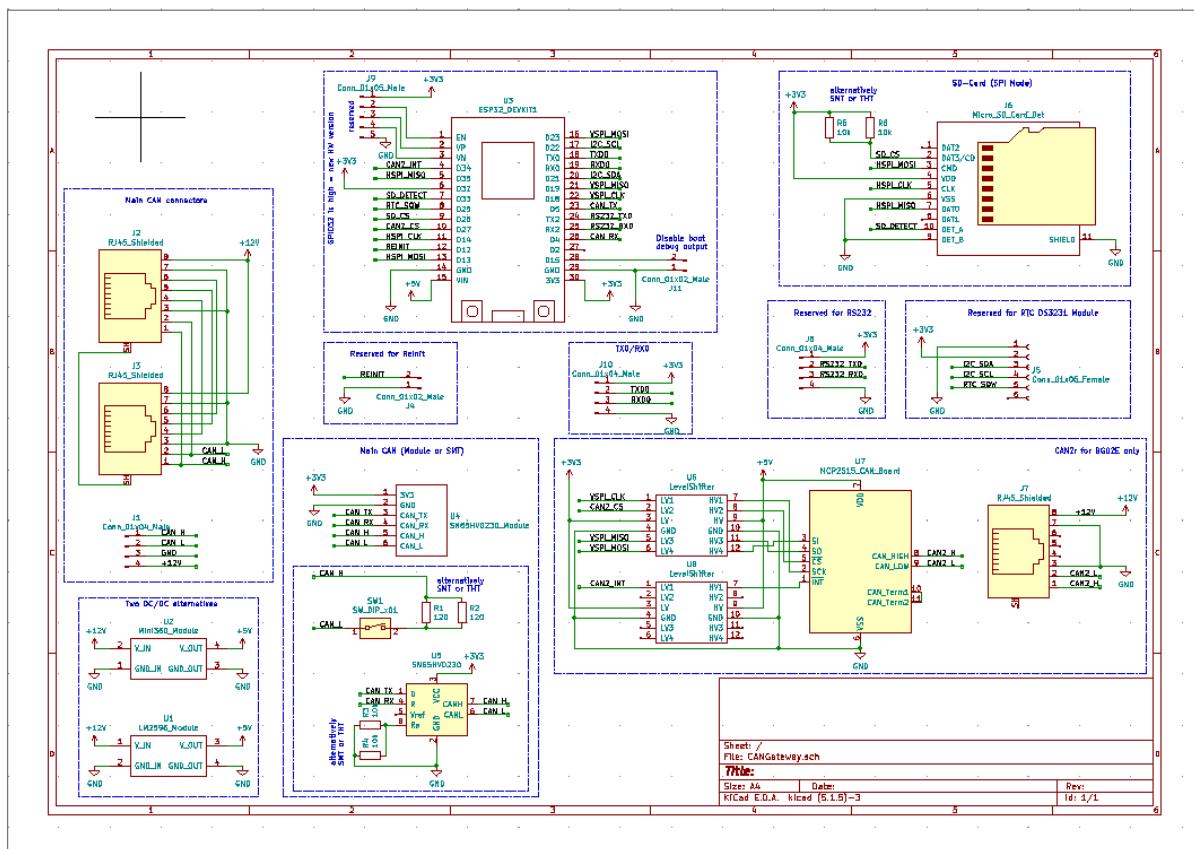


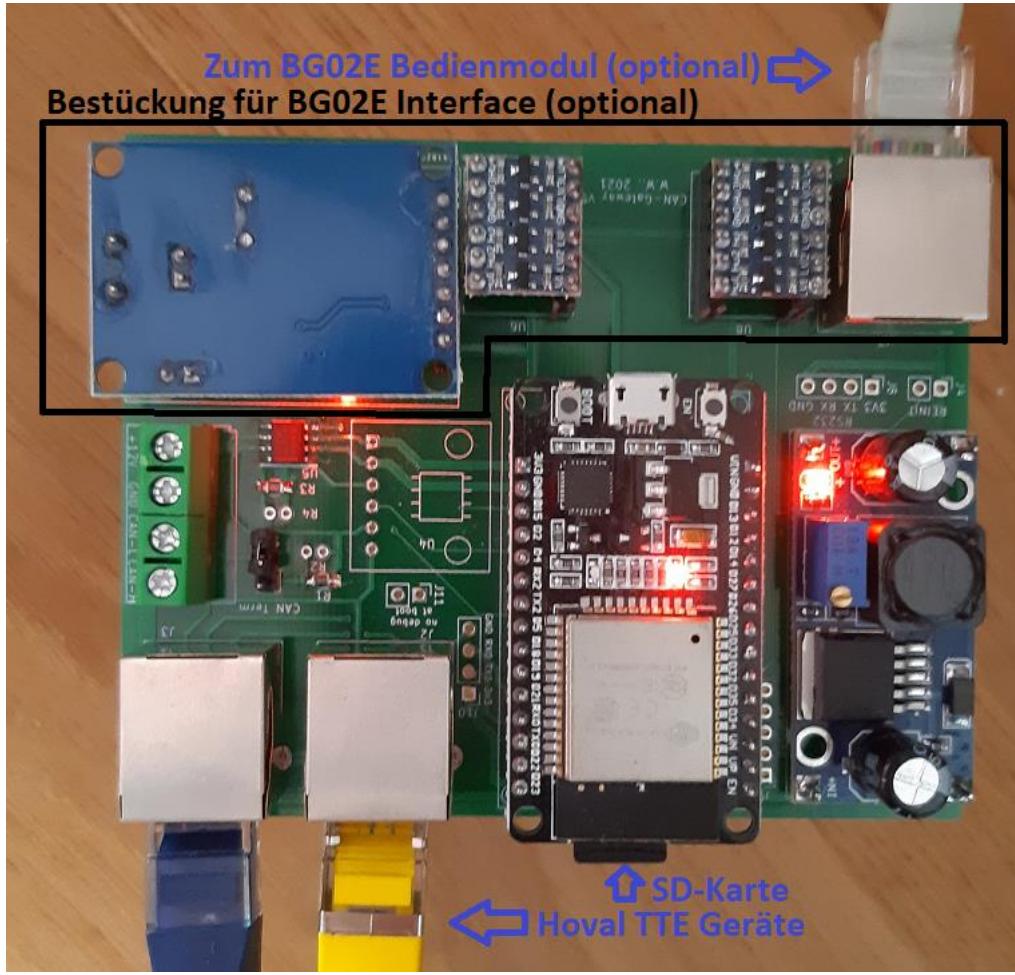
Beispiel fertiges Board mit einem zusätzlichen Schraubverbinder:



3.4 Hardware „CAN-Gateway Version 5“

Hardware „CAN-Gateway Version 5“ ist eine Referenzhardware für Eigenbau basierend auf dem einfachen Aufbau mit einem zusätzlichen Micro- SD-Kartenslot :





3.5 Nutzung ESP32-EVB(-EA) Entwicklungsboards von Olimex als CAN-Gateway

Firma www.olimex.com bietet ein Entwicklungsboard ESP-32-EVB(-EA), welches als CAN-Gateway Hardware verwendet werden kann [**neu seit SW Version 28.001**]:

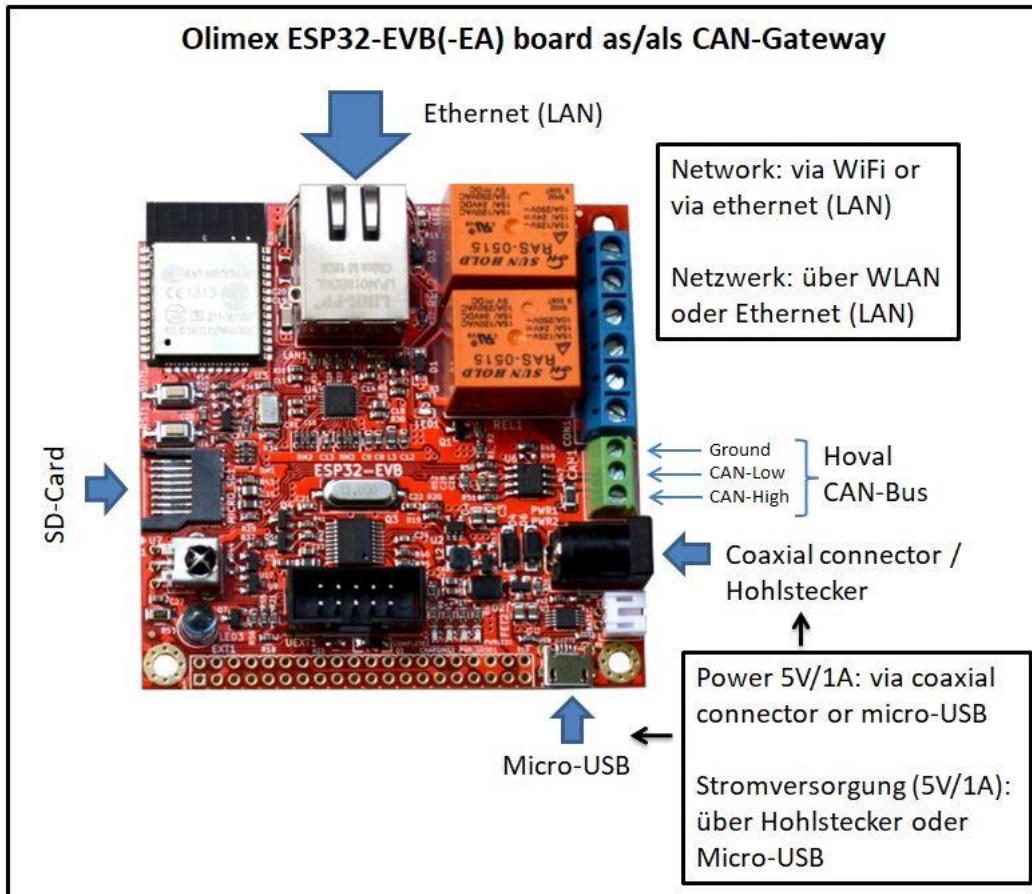
Vorteile des Olimex Boards gegenüber CAN-Gateway HW V5:

- Ethernet (LAN) als Alternative zu WLAN (Erstkonfiguration erfolgt immer über WLAN, dann kann aber auf Ethernet/LAN umgestellt werden)
- -EA -Version mit externer WLAN-Antenne
- Metallisches Gehäuse ist bei Olimex für die -EA -Version erhältlich

Nachteile des ESP32-EVB(-EA) Boards im Vergleich zum CAN-Gateway HW V5:

- Keine RJ45-Stecker für Hoval CAN-Bus, nur Schraubklemmen
- SD-Karte: kein automatisches Montieren/Demontieren beim Einstecken/Auswerfen im laufenden Betrieb
- Externe Stromversorgung erforderlich, keine Möglichkeit zur Stromversorgung über Hoval CAN-Bus
- Keine BG02E-Schnittstellenoption

Anschlüsse:



4 Software flashen

4.1 Automatisiert (empfohlen)

Die DEMO Version der Software kann ganz leicht über den Web-basierten Installer installiert werden:

<https://wladwnt.github.io> Die DEMO Version kann später auf die erworbene Vollversion über WEB-Interface des CAN-Gateways upgedatet werden.

4.2 Automatisiert (empfohlen), nur für CAN-Gateway development board

CAN-Gateway development board von wled.shop wird mit einer Software ausgeliefert, die erlaubt andere Software darauf über Web-interface ganz leicht zu flashen. So kann CAN-Gateway Demo oder Vollversion darauf ganz leicht mit nur wenigen Klicks installiert werden.

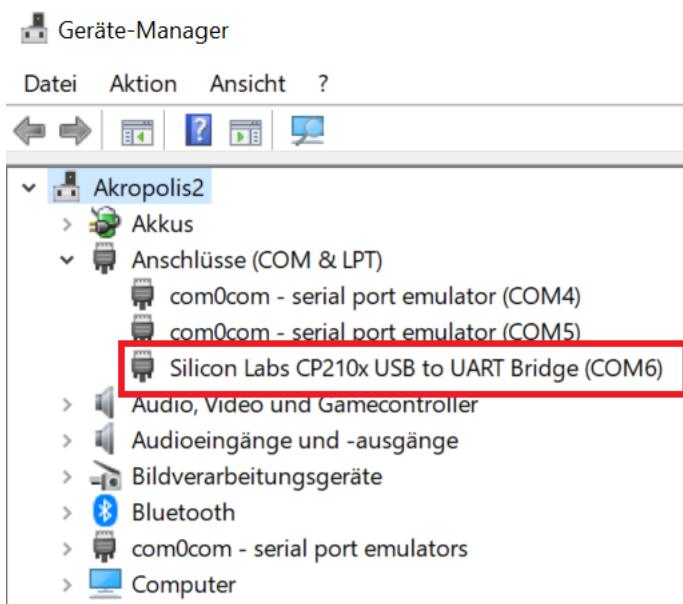
4.3 Manuell

Die Software wird als eine binäre Datei ausgeliefert und muss zuerst auf ESP32 Mikrocontroller geflasht werden. Dafür muss wiederum zunächst das Board mit dem Rechner über USB Kabel verbunden und der Treiber für den Kommunikationschip muss installiert werden. Insgesamt folgende Schritte sind erforderlich (angenommen, Sie nutzen Windows 10):

- 1) Den benötigten Treiber herausfinden. Die meisten ESP32 basierte Boards verwenden einen CP210x USB-Kommunikationschips. Den Treiber dafür findet man hier:
<https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>

Olimex Boards und einige anderen haben CH340 USB-Kommunikationschips. Den Treiber dafür findet man hier: http://www.wch-ic.com/downloads/CH341SER_EXE.html

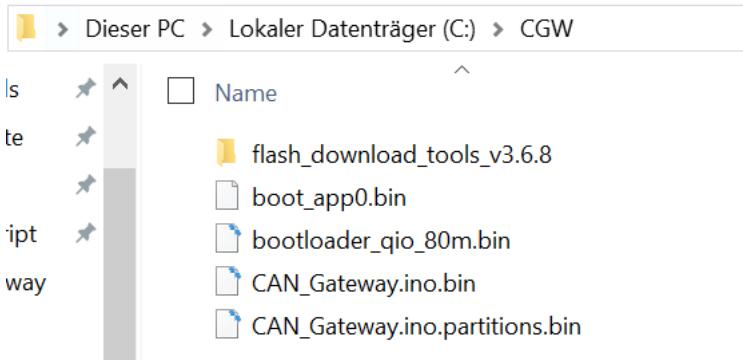
- 2) Den Treiber herunterladen und auf Ihrem PC/Laptop installieren.
- 3) PC neu starten.
- 4) Schauen Sie, welche COM-Ports in Ihrem System bereits vorhanden sind. Unter Windows 10 geht es so:
 - a. Geräte-Manager öffnen (rechter Mausklick auf Windows Symbol und dann auf Geräte-Manager klicken).
 - b. Im Geräte-Manager auf Anschlüsse (COM & LPT) gehen und die Liste anschauen.
- 5) ESP32 DevKit Board mit dem PC/Laptop verbinden. Dabei wird ein neuer virtueller COM-Port eingerichtet.
- 6) Im Geräte-Manager erscheint dann ein neuer Port (z.B. COM6), diesen Namen bitte merken.



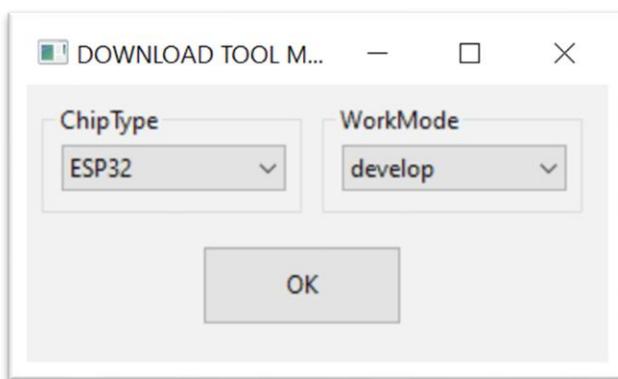
- 7) Erstellen Sie ein Verzeichnis auf dem Laufwerk C; z.B. **C:\CGW**. Sie können es auch anders benennen, müssen aber dann in weiteren Schritten überall den richtigen Namen verwenden.
- 8) Herunterladen und entpacken Sie ESP Download Tool in diese Verzeichnis (<https://www.espressif.com/en/support/download/other-tools>, dann unter „Flash Download Tools“)
- 9) Entpacken Sie die CAN-Gateway Software (die Datei „CAN_Gateway.ino.bin“ (bzw. „CAN_Gateway_DEMO.ino.bin“ für DEMO Version)
- 10) Zudem müssen Sie noch die Dateien (bis einschließlich CAN-Gateway Version 29.101) „boot_app0.bin“ (Quelle: https://github.com/wladwnt/wladwnt.github.io/blob/main/cangw/boot_app0.bin), „bootloader_qio_80m.bin“ (Quelle: https://github.com/wladwnt/wladwnt.github.io/blob/main/cangw/bootloader_qio_80m.bin) in diesem Verzeichnis speichern und CAN_Gateway.ino.partitions.bin (Quelle: https://github.com/wladwnt/wladwnt.github.io/blob/main/cangw/CAN_Gateway.ino.partitions.bin).
Ab CAN-Gateway Version 29.201 sind die entsprechenden Dateien https://github.com/wladwnt/wladwnt.github.io/blob/main/cangw/boot_app0_v202.bin, https://github.com/wladwnt/wladwnt.github.io/blob/main/cangw/CAN_Gateway.ino.partitions_v202.bin und https://github.com/wladwnt/wladwnt.github.io/blob/main/cangw/CAN_Gateway.ino.partitions_v202.bin

https://github.com/wladwnt/wladwnt.github.io/blob/main/cangw/CAN_Gateway.ino.bootloader_v202.bin.

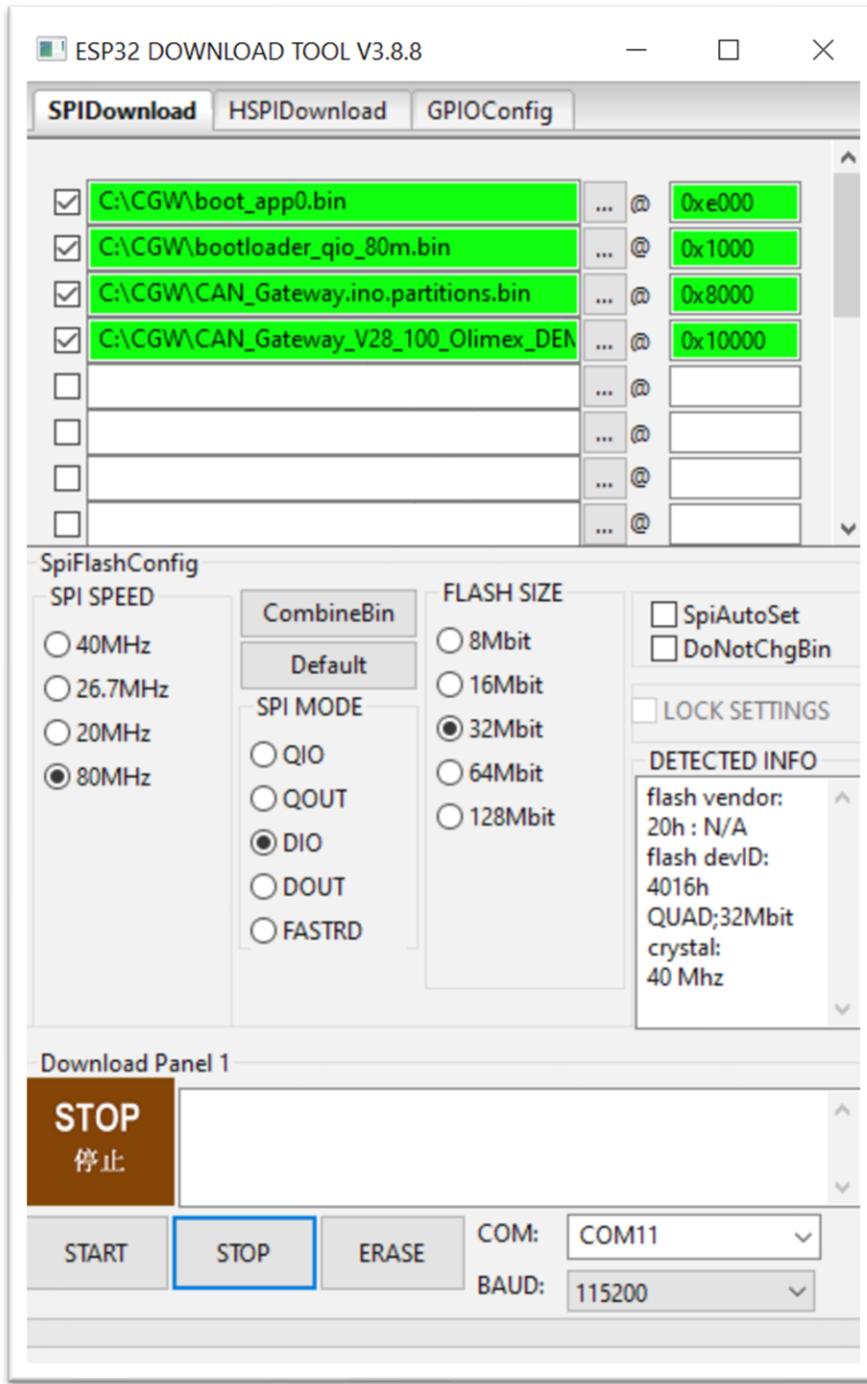
- 11) Haben Sie bisher alles korrekt gemacht, haben Sie folgende Dateien:



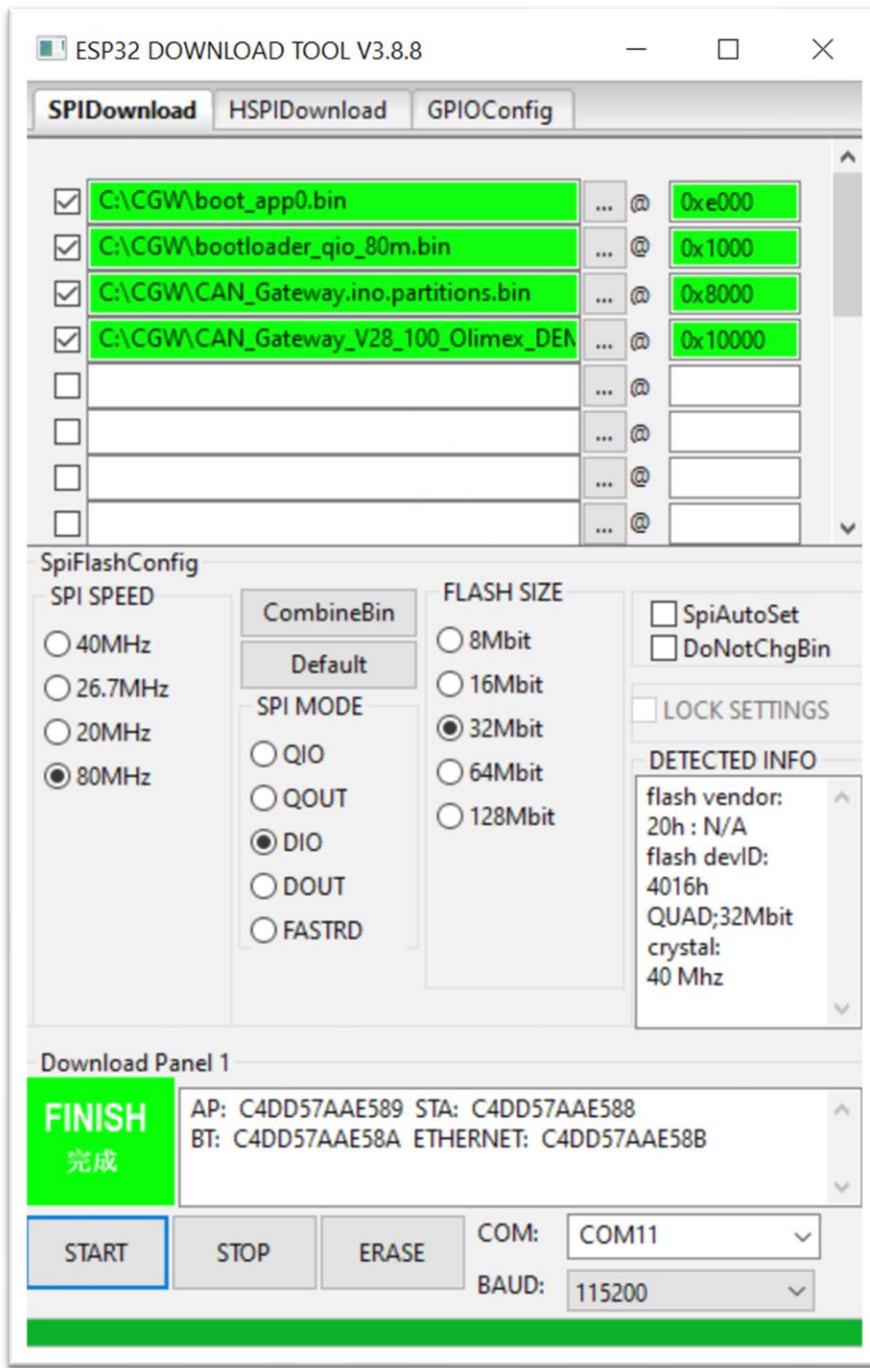
- 12) Nun können Sie ESP32 Download Tool starten (exe Datei aus dem Verzeichnis flash_download_tool_vX.X.X, wobei X.X.X die aktuelle Version ist, die Sie runtergeladen haben)
- 13) Dann erscheint ein Fenster, wo Sie ChipType „ESP32“ und WorkMode „develop“ auswählen.



- 14) Im nächsten Dialog geben Sie die Daten ein, genauso wie hier im Bild zu sehen ist. Das einzige was Sie anpassen müssen, ist den richtigen COM-Port auszuwählen (unten rechts). Wenn man Probleme beim Flashen hat, hilft evtl. auch wenn man die Baudrate niedriger wählt. Passen Sie auf, dass alle Hacken korrekt gesetzt sind. Danach können Sie auf „Start“ unten links klicken. Der Flash-Vorgang beginnt.



- 15) Sobald er beendet wird, erscheint unten im grünen Feld „Finish“. Flash Tool kann nun geschlossen werden. Alternativ zum Flashtool kann man auch die Programme wie esptool.exe oder esptool.py verwenden, die z.B. ein Bestandteil der Arduino Entwicklungsumgebung (mit installierten ESP32 Board-Erweiterungen) sind.



- 16) Nun kann man ESP32 DevBord neu starten, entweder indem man ihn vom USB Port entfernt und wieder einsteckt oder durch ein Reset Knopf auf dem Board.

Wenn man esptool.exe fürs Flashen verwendet, muss das Programm mit folgenden Parametern gestartet werden (COM-Port müssen Sie natürlich anpassen):

```
esptool.exe --chip esp32 --port COM6 --baud 256000 --before default_reset --after hard_reset
write_flash -z --flash_mode dio --flash_freq 80m --flash_size detect 0xe000 C:\CGW\boot_app0.bin
0x1000 C:\CGW\bootloader_qio_80m.bin 0x10000 C:\CGW\CAN_Gateway.ino.bin 0x8000
C:\CGW\CAN_Gateway.ino.partitions.bin
```

5 Erste Inbetriebnahme

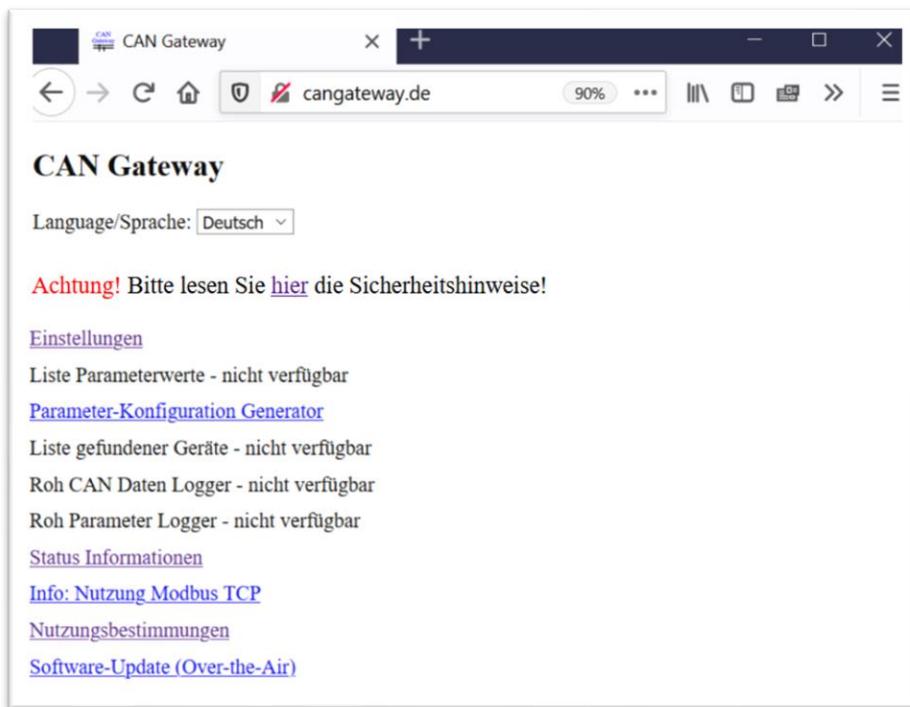
Im initialen Zustand, also wenn die Software zum ersten Mal auf das ESP32 DevKit geflasht wurde und CAN-Gateway gestartet wird, erzeugt CAN-Gateway einen WLAN Zugangspunkt (Access Point). Man muss ein PC/Laptop/Tablett mit diesem WLAN verbinden (WLAN Name: cangateway, Netzwerkschlüssel: 000999555), einen Internet Browser öffnen, zu der Startadresse gehen (z.B. <http://cangateway.de> eintippen) und die Einstellungen vornehmen. In Details sind folgende Schritte notwendig (hier angenommen, Sie haben Windows 10):

- 1) Stellen Sie sicher, dass Sie Ihre aktuellen WLAN Zugangsdaten kennen. Sie werden ihr PC im nächstens Schritt für eine Weile zu einem anderen Netzwerk verbinden müssen, danach müssen Sie aber zurück ins Ihr Heimnetzt und das wird ggf. erfordern, dass Sie Ihre Zugangsdaten (Password) eingeben müssen.
- 2) Klicken Sie auf der Windows-Leiste unten rechts auf WLAN Symbol, im geöffneten Fenster suchen Sie nach dem Netzwerk „cangateway“, klicken Sie drauf und dann auf „Verbinden“.

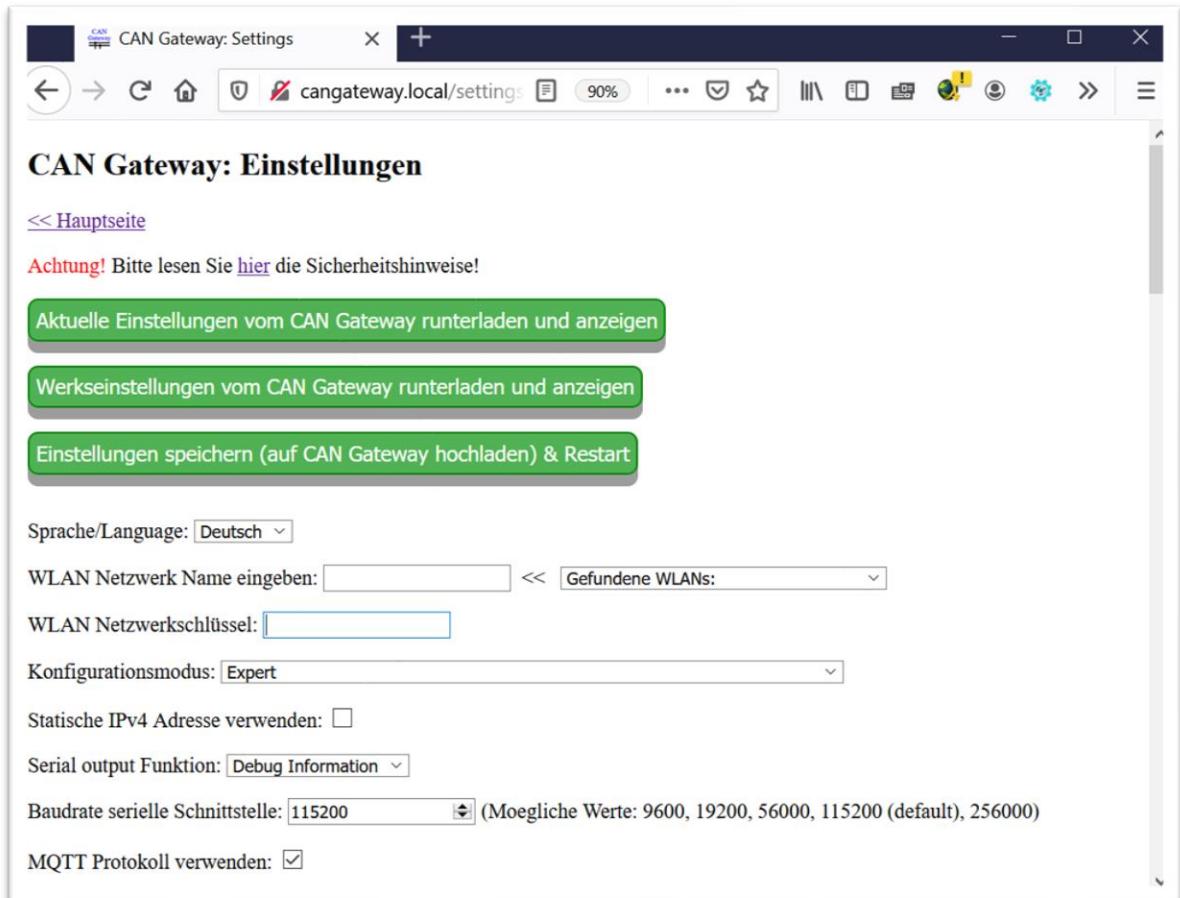


- 3) Es werden Netzanforderungen überprüft, danach müssen Sie den Netzwerkschlüssel 000999555 eingeben und auf „Weiter“ klicken.
- 4) Warten Sie bis die Verbindung zustande kommt und unter „cangateway“ erscheint „Kein Internet, gesichert“
- 5) Sie sind jetzt mit dem CAN-Gateway verbunden.
- 6) Öffnen Sie einen Web-Browser und in die Adresse-Leiste tippen Sie cangateway.de ein. Es wird die erste Seite geöffnet. Diese Seite befindet sich nicht wie üblich irgendwo im Internet,

sondern wird vom CAN-Gateway erzeugt. Hier können Sie als erstes zwischen deutscher und englischer Sprache wählen.

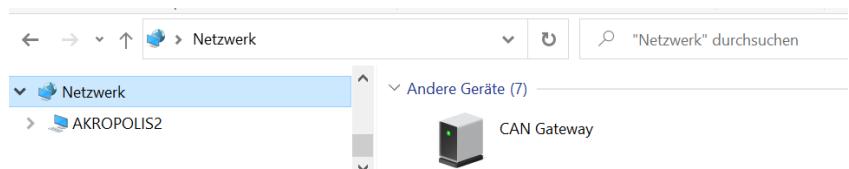


- 7) Danach klicken Sie auf „Einstellungen“ und lesen dann aufmerksam die Nutzungsbestimmungen. Diese Nutzungsbestimmungen sind identisch zu den Angaben im Kapitel 1.1 „Gefahrenhinweise“ und 1.2 „Haftung des Softwareherstellers bzw. des Anbieters“ identisch. Sind Sie damit einverstanden, können Sie auf „[Ich habe alle Gefahrenhinweise gelesen, akzeptiere die Bedingungen der Mängelhaftung und möchte weiter zu Einstellungen gehen](#)“ klicken.
- 8) Auf nächster Seite klicken Sie zuerst auf „Aktuelle Einstellungen vom CAN-Gateway runterladen und anzeigen“. Als erstes können Sie Ihre Heim-WLAN Name und Password (Netzwerkschlüssel) eingeben (WiFi Name/Password). Alle andren Einstellungen können Sie erstmal so lassen wie es ist. Danach klicken Sie auf „Einstellungen speichern (auf CAN-Gateway hochladen) & Restart“. Jetzt könne Sie sich vom cangateway Netzwerk trennen und sich wieder mit Ihrem Heim-Netzwerk verbinden. Achtung: sobald Netzwerkname auf einen anderen als „cangateway“ geändert wird, geht die Software davon aus, dass es ein existierendes WLAN ist und erzeugt kein Access Point mehr.



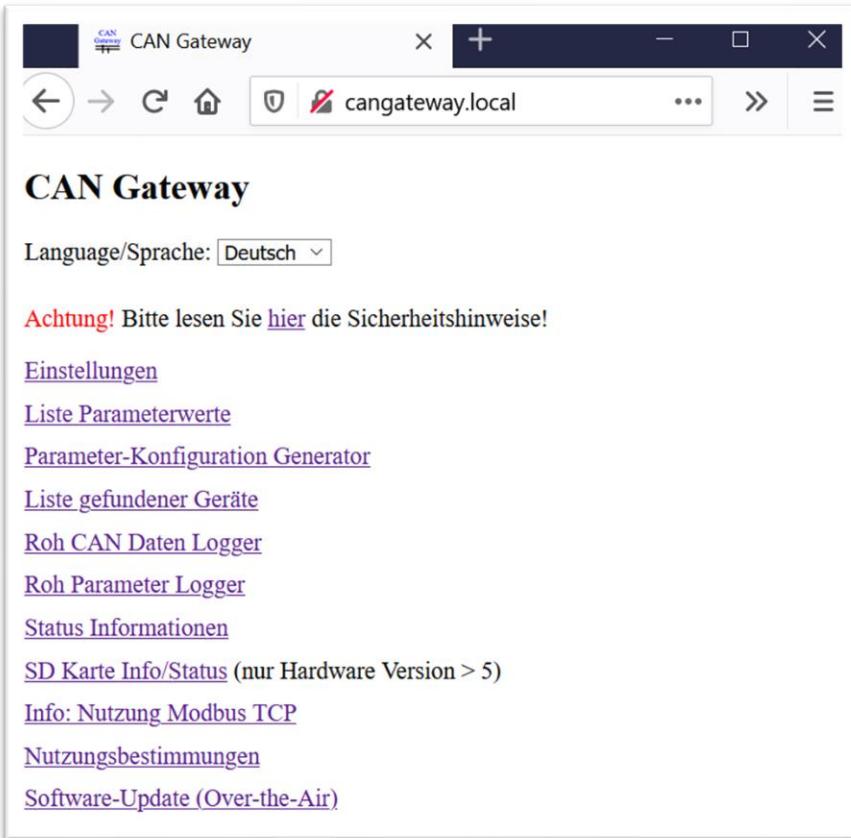
- 9) Nach Neustart versucht CAN-Gateway sich mit Ihrem Heim-netzwerk verbinden. Damit es funktioniert, muss Ihr WLAN Router unter anderem 2,4 GHz Band, 802.11 b/g/n Standard und WEP/WPA-TKIP/WPA2-CCMP sowie optional DHCP unterstützen (all das ist typisch für WLAN Netzwerke) und keine neuen Netzwerkteilnehmer blockieren (also nicht z.B. machen MAC-Adresse rausfiltern etc.). Nun ist der CAN-Gateway mit Ihrem Heim-Netzwerk verbunden. Ich empfehle jetzt auch Ihren PC neu zu starten, damit der CAN-Gateway vom PC unter der Adresse <http://cangateway.local> gefunden werden kann. CAN-Gateway verwendet mDNS Protokoll. Windows 10 erkennt neue mDNS Geräte nur, wenn Bonjour service² installiert ist und nur nach Neustart. iOS und Linux müssen es sofort erkennen. Sie können nun also wieder Web-Browser öffnen und <http://cangateway.local> als Adresse eingeben. Mit Firefox funktioniert es bei mir, mit Chrome aber z.B. nicht, weil Chrome Browser versucht cangateway im Internet über Google zu suchen, es ist aber ein lokaler Web-Server. Sollte die Adresse <http://cangateway.local> nicht funktionieren, weil z.B. Ihr Betriebssystem oder Browser mDNS Protokoll nicht unterstützt, müssen Sie über Ihr Router die IPv4 Adresse vom CAN-Gateway ermitteln. Wenn Sie als Router z.B. Fritzbox haben, können sie auch einfach <http://cangateway> (für CAN-Gateway Software ab Version 19.001) oder <http://espressif> (für CAN-Gateway Software vor Version 19.001) probieren.
[neu in SW 27.400] CAN-Gateway unterstützt SSDP Protokoll und wird in Windows unter „Netzwerk“ gelistet. Mit dem rechten Mausklick auf dem CAN-Gateway gelangt man zu „Eigenschaften“ und dort kann man die IP-Adresse des CAN-Gateways ablesen.

² Wie man Bonjour service installiert, findet man z.B. hier: <https://softwarekeep.com/help-center/what-is-bonjour-service-on-windows-10>. Nach der Installation PC/Laptop neu starten!



6 WEB-Interface: Hauptseite

CAN-Gateway besitzt ein WEB-Interface. Über dieses Interface sind verschiedene Funktionen wie Einstellungen, Liste der Parameter und gefundener Geräte, CAN-Datenlogger, Parameter-Datenlogger etc. verfügbar. WEB-Interface kann über die Adresse cangateway.local erreicht werden. Voraussetzung dafür allerdings, dass ihr PC/Laptop-Betriebssystem mDNS Protokoll unterstützt. Siehe dazu auch Schritt 8 in Kapitel „Erste Inbetriebnahme“. Web-Interface kann selbstverständlich auch direkt über IPv4 Adresse angesprochen werden.



7 Allgemeine Informationen zu Gerät-Parametern

Als Parameter werden hier alle Werte gemeint, die aus den Geräten ausgelesen oder zwecks Ansteuerung eingestellt/geändert werden können. Als Beispiele können z.B. von den Geräten gemessenen Temperaturen, Leistungen, Gerät-Namen oder Sollwerte für Temperatur, Luft-Durchströmung etc. genannt werden. Verschiedene Gerät-Typen haben verschiedene Parameter, die man entweder nur auslesen oder auch einstellen (ändern) kann. Jeder Parameter, auch „Datapoint“ genannt, hat einen Datapoint-ID (also einen eindeutigen Nummer), gehört zu einer bestimmten Funktion, die durch einen Funktionsnummer beschrieben wird, und die Funktion gehört einer bestimmten Funktionsgruppe. Da jedes Gerät durch den Gerätetyp und die Geräteadresse eindeutig identifiziert wird, ist jeder Parameter in einem konkreten Gerät in Summe durch 5 Eigenschaften (in Form von 5 Zahlen) eindeutig identifiziert wird: Gerätetyp, Geräte-Adresse, Funktionsgruppe, Funktionsnummer und Datapoint-ID:

- **Gerätetyp:** Eine Zuordnung der Gerätetypen zu entsprechenden Typ-Nummern ist auf der Seite mit CAN-Gateway Einstellungen (zugänglich über CAN-Gateway Webinterface) zu finden.
- **Geräteadresse:** Die Geräteadresse kann z.B. durch einen Bedienmodul (mit Touchscreen) eingesehen und im passenden User-Level Modus (siehe dazu User-Level Information) auch geändert werden. (Selbstverständlich auch durch CAN-Gateway möglich). Es gibt aber normalerweise keine Notwendigkeit die Adresse zu ändern, es sei denn, man will ein gleiches Gerät an CAN-Bus anschließen. Standardadresse ist wohl 8. Um die vorhandenen Geräte leichter zu identifizieren, Analysiert CAN-Gateway alle Informationen auf dem CAN-Bus und erzeugt eine Liste mit verfügbaren Geräten (inklusive Gerätetyp und Geräte-Adresse), die dann über CAN-Gateway Webinterface eingesehen werden kann.
- **Funktionsgruppe:** Die Funktionsgruppe kann zu einem über den Bedienmodul eingesehen werden. Im Bereich „Service“ werden alle Geräte aufgelistet, die vorhanden sind, und dann für jedes Gerät sieht man die Funktionsgruppen, wobei in Klammern dann steht eine Zahl mit der Nummer der Funktionsgruppe. Funktionsgruppe steht auch auf der Liste der Parameter (Excel Datei, die vom Hersteller erhältlich ist und die durch offiziellen Gateways vom Hersteller verfügbaren Parameter beschreibt).
- **Funktionsnummer:** Die Funktionsnummer kann wie die Funktionsgruppe über Bedienmodul (das ist die zweite Ebene nach der Funktionsgruppe) oder in Parameterlisten (Excel-Dateien vom Hersteller) eingesehen werden.
- **Datapoint-ID:** Datapoint-IDs können wie Funktionsgruppen und Funktionsnummern über Bedienmodul oder in Parameterlisten (Excel-Dateien vom Hersteller) eingesehen werden. Auf dem Bedienmodul ist das die (dezimale) Zahl die im Bereich „Service“ zwischen dem Parameter-Namen und Parameter-Wert in der Form XX-XXX in Grau angezeigt wird. Der Bindestrich muss dabei einfach ignoriert werden.

Alle Parameter, die aus Geräten ausgelesen werden können oder die eingestellt werden können, sind auf dem CAN-Bus grundsätzlich verfügbar. Auf dem CAN-Bus kann man auch auf die Parameter zugreifen, die dem Benutzer im Bedienmodul bei bestimmten User-Levels gar nicht angezeigt werden.

CAN-Gateway hat einen Logger, der alle Parameter, die auf dem CAN-Bus gerade übertragen werden, aufzeichnet und diese können dann über CAN-Gateway WEB-Interface angezeigt werden. Diese Funktion hat allerdings nur dann Sinn, wenn ein Bedienmodul oder ein Gateway im System

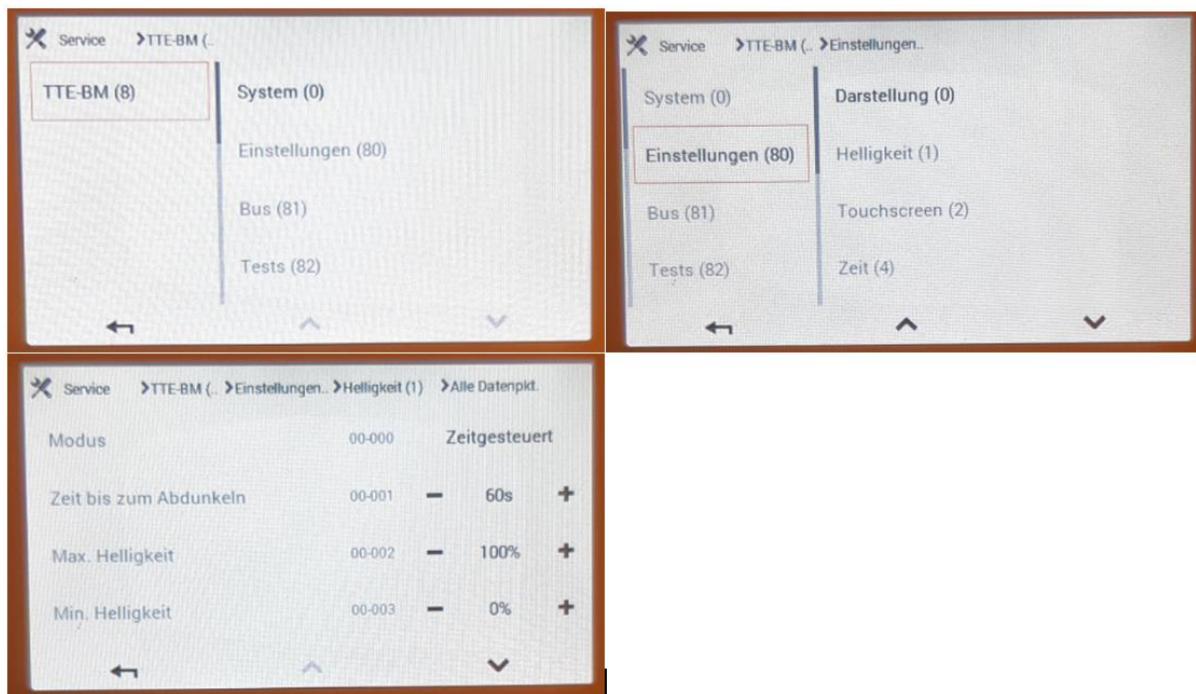
vorhanden ist, da die Parameter nur auf die Anfrage auf dem CAN-Bus übertragen werden. Wenn auf dem CAN-Bus nichts übertragen wird, wird im Parameter-Logger vom CAN-Gateway auch nichts angezeigt. Parameter, die in den Einstellungen des CAN-Gateways konfiguriert sind, werden vom CAN-Gateway periodisch über CAN-Bus angefragt und wenn die Geräte antworten, erscheinen dann diese Parameter auch im Parameter-Logger.

Grundsätzlich gäbe es die Möglichkeit bei einem Gerät anzufragen, welche Parameter er besitzt. Das macht z.B. jeder Bedienmodul. Diese Funktion ist aber auf dem CAN-Gateway aktuell nicht implementiert.

7.1 Beispiel

Im folgenden Beispiel wird ein Parameter betrachtet, welches dem Bedienmodul gehört und die Zeit bis zum Abdunkeln des Touchscreens vorgibt.

Da es um ein Parameter des Bedienmoduls handelt, ist der Gerätetyp=16. Die Adresse des Bedienmoduls wird unter anderem im Bereich „Service“ in Klammern zu TTE-BM dargestellt, in diesem Beispiel ist die Adresse des Bedienmoduls = 8. Der Parameter ist in der Funktionsgruppe „Einstellungen“ mit dem Nummer=80 und in der Funktion „Helligkeit“ mit der Nummer=1. Datapoint-ID ist „00-001“, also auch =1. Der aktueller Wert ist 60s.



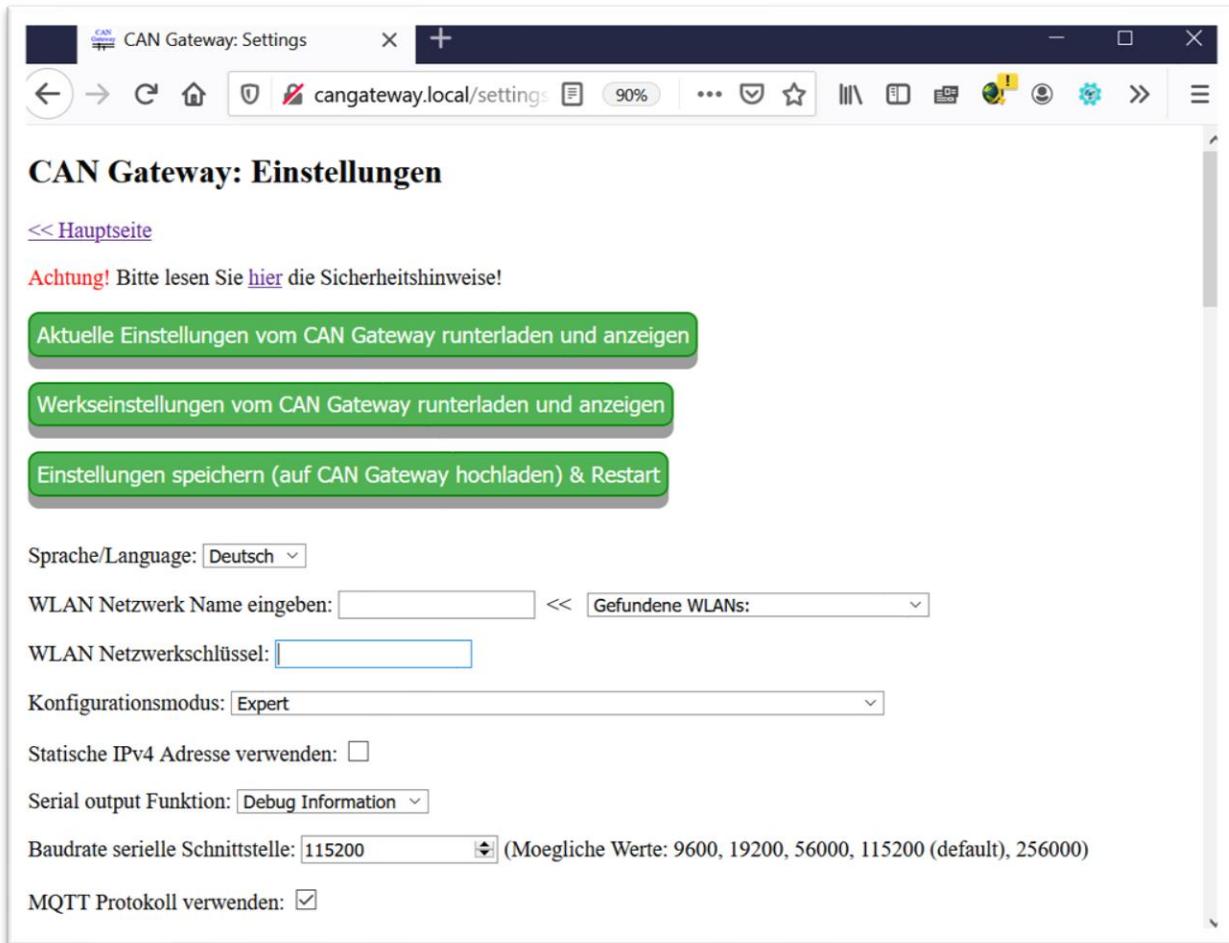
8 User-Levels

User-Level beschreibt die Zugriffsebene, also definiert unter anderem welche Parameter vom Benutzer über einen Bedienmodul ausgelesen und welche auch geändert werden können. Es gibt insgesamt 8 User-Levels: 0 bis 7. User-Level=0 ist der niedrigste (default), User-Level=7 gibt Schreibzugriff auf alles, praktisch unabhängig davon ob die Änderung bestimmter Werte überhaupt Sinn hat oder nicht. Jeder User-Level (außer 0) ist mit einem Password geschützt. Wichtig ist, dass User-Level sich eigentlich nur auf das beziehen, was auf dem Bedienmodul gemacht werden kann bzw. darf. Auf dem CAN-Bus ist alles verfügbar, also der CAN-Gateway kann generell unabhängig vom auf dem Bedienmodul eingestellten User-Level auf alles zugreifen und (**Vorsicht!**) alles ändern. CAN-Gateway kann auch ganz ohne Bedienmodul direkt mit einem Gerät funktionieren. Mit dem CAN-Gateway kann man sogar die Kennwörter für alle User-Levels aus dem Bedienmodul auslesen und diese auch ändern.

9 WEB-Interface: Einstellungen

Über WEB-interface des CAN-Gateways können verschiedene Einstellungen vorgenommen werden:

- Sprache
- [neu in SW 28.001] Auswahl WLAN oder Ethernet (LAN, nur bei Boards die Ethernet unterstützen: aktuell Olimex ESP32-EVB(-EA) Entwicklungsboards)
- WLAN Netzwerkname und Netzwerkschlüssel
- Benutzung statischer IP Adresse für WLAN Netzwerk
- Funktion der seriellen Schnittstelle. Es gibt drei mögliche Funktionen zur Auswahl:
 - o **CAN-Rohdaten** – über serielle Schnittstelle (USB Anschluss) werden die empfangenen CAN Daten ausgegeben: CAN-ID, DLC, Daten-Bytes.
 - o **Parameters** - über serielle Schnittstelle (USB Anschluss) werden die empfangenen Parameter ausgegeben.
 - o **Debug-Informationen**: über serielle Schnittstelle (USB Anschluss) werden Informationen über den aktuellen Status des CAN-Gateways ausgegeben, was ggf. eine Fehlersuche erleichtert.
- Baudrate Serielle Schnittstelle
- MQTT Server IP, Server Port, Nutzername, Kennwort. Das muss den Einstellungen in Ihrem MQTT Broker entsprechen.
- MQTT prefix. Das ist der allgemeine Teil für jedes Topic. Siehe dazu Kapitel „Übertragung der Parameter über MQTT Protokoll“.
- [neu ab SW 26.001] Unterstützung [Home Assistant / MQTT Discovery](#) für Sensoren
- Identifikation des CAN-Gateways auf dem CAN-Bus (Gerätetyp, Adresse).
- Einstellung, ob CAN-Gateway im Bedienmodul unter anderen Geräten aufgelistet und gesteuert werden kann (siehe Kapitel „Steuerung über Bedienmodul“).
- [neu ab SW Version 22.001] Zeitserver um Information über die aktuelle Zeit über NTP Protokoll zu erfragen
- [neu ab SW Version 22.001] Offset für die lokale Zeit ggü. UTC in Sekunden
- [neu ab SW Version 23.001] Profil, wie die historischen Parameterdaten in RAM gespeichert werden. Zur Auswahl stehen folgende Optionen:
 - o Über die letzten 6 Std.: ein Wert alle 5 Min.; über die letzten 6 Minuten: ein Wert alle 10 Sek.
 - o Über die letzten 12 Std.: ein Wert alle 10 Min.; über die letzten 6 Minuten: ein Wert alle 10 Sek.
 - o Über die letzten 24 Std.: ein Wert alle 20 Min.; über die letzten 6 Minuten: ein Wert alle 10 Sek.
 - o Über die letzten 24 Std.: ein Wert alle 20 Min.; über die letzten 36 Minuten: ein Wert pro Min.
 - o Über die letzten 3 Tage: ein Wert pro Std.; über die letzten 6 Minuten: ein Wert alle 10 Sek.
 - o Über die letzten 3 Tage: ein Wert pro Std.; über die letzten 36 Minuten: ein Wert pro Min.
- [neu ab SW 26.001, nur in Verbindung mit HW ab V5]: Speicherung auf SD-Karte mit folgenden Optionen bzgl. was muss gespeichert werden: Parameterwerte, CAN Rohdaten, Parameter Rohdaten.



Bevor man die Parameter ändert, sollte man die aktuellen Einstellungen vom CAN-Gateway runterladen und anzeigen lassen. Danach kann man die Parameter anpassen und auf „Einstellungen speichern“ klicken. Speichern bedeutet, dass CAN-Gateway die Einstellungen im internen nichtflüchtigen elektronischen Speicher speichert und neu startet.

10 WEB-Interface: Parameter Einstellungen

Damit der CAN-Gateway die Parameter aus den Geräten ausliest/ändert und diese an Hausautomatisierungssystem überträgt, müssen die Parameter konfiguriert werden. Maximal können 40 (20 bis SW Version 22.001) Parameter konfiguriert werden. Zur Konfiguration eines Parameters gehören folgende Angaben:

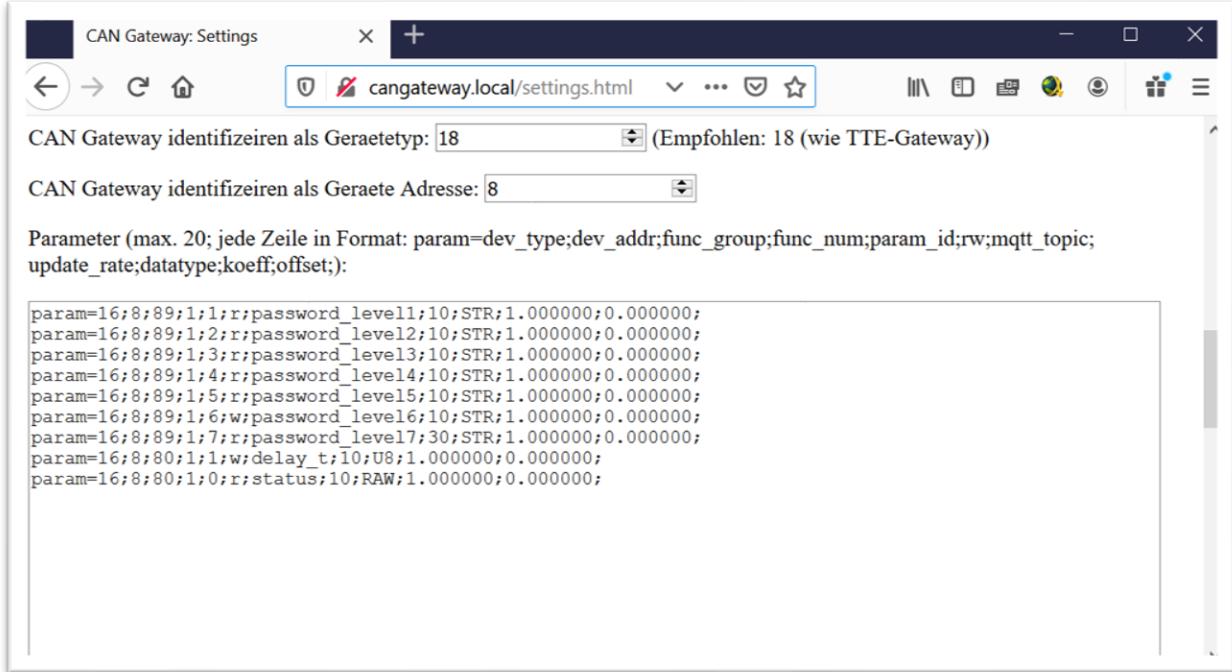
- Gerätetyp, Geräteadresse, Funktionsgruppe, Funktionsnummer, Datapoint-ID – siehe dazu die Beschreibung im Kapitel „Allgemeine Informationen zu Geräte-Parametern“.
- Datentyp – siehe dazu die Beschreibung im Unterkapitel „Datentypen“.
- MQTT Topic – siehe dazu die Beschreibung im Kapitel „Übertragung der Parameter über MQTT Protokoll“.
- Update-Rate. Update Rate kann die Werte 10, 30 oder 60 annehmen und gibt die Periode in Sekunden an, wie oft CAN-Gateway den Parameter bei dem entsprechenden Gerät anfragt. Normalerweise antwortet der Gerät sofort mit dem aktuellen Wert. Dieser Wert wird dann um ca. 100 ms verzögert an den MQTT Broker aktiv gesendet (published) und steht dann auch zur Abholung über REST-API oder über Modbus-TCP Protokoll bereit. Publishing über MQTT-Protokoll findet unabhängig davon statt, ob der Wert sich geändert hat oder nicht.

Anzumerken ist, dass immer wenn ein Parameter über CAN-Bus übertragen wird, wird sein Wert von CAN-Gateway erfasst und über MQTT publiziert. Also wenn z. b. andere Geräte den Wert öfter Anfragen als die Update-Rate, wird der Wert auch öfter über MQTT publiziert. CAN-Gateway erzeugt auch die Anfrage nur dann, wenn seit letzter Erfassung eines Parameters mehr Zeit vergangen ist, als über Update-Rate eingestellt ist. Der alter eines Wertes kann über WEB-Interface des CAN-Gateways auf der Seite „Liste der Parameter“ eingesehen werden.

- Lesen/Schreiben (read/write) Attribut. Ein Parameter kann nur zum Lesen (Attribut=r) oder zum Lesen und Ändern (Attribut=w) definiert werden. Wenn ein Parameter mit dem Attribut „w“ definiert wird, kann über WEB-Interface, über MQTT Protokoll, über REST-API oder über Modbus-TCP eine Änderungsanfrage erzeugt werden. Diese Anfrage wird dann vom CAN-Gateway an den Gerät weitergeleitet. Normalerweise bestätigt das Gerät die Änderung, indem er den geänderten Parameterwert zurück schickt. Das wird wiederum vom CAN-Gateway als Parameterübertragung auf dem CAN-Bus erfasst, über MQTT publiziert etc. Da die ganze Kette von der Änderungsanfrage bis zu Bestätigung der Änderung etwas dauert, muss das HAS nach einer Änderungsanfrage etwas warten, bis man den geänderten Wert zurück lesen kann und damit feststellen kann, ob die Änderung umgesetzt wurde. 0,5 s sollte normalerweise reichen, empfohlen wäre 1 s. ein HAS kann die Änderungsanfrage grundsätzlich so oft erzeugen wie es will, CAN-Gateway leitet die Änderungsanfrage an das Gerät maximal einmal pro 3 Sekunden. HAS sollte so programmiert werden, dass sie auch eine Änderungsanfrage nicht zu oft erzeugt. Sonst kann es theoretisch passieren, dass der Prozessor des CAN-Gateways zu stark ausgelastet wird und das ganze System abstürzt bzw. nicht mehr korrekt funktioniert.
- Koeffizient/Offset. Diese Vorgaben müssen immer vorhanden sein, sind aber nicht für Datentypen STR und RAW relevant (in diesem Fall können beliebig sein, z.B. 0/0). Koeffizient und Offset beeinflussen das, wie der auf dem CAN-Bus in binärform übertragene Wert in einen Wert umgerechnet wird, welcher dann über WEB-Interface einsehbar ist, über MQTT Protokoll übertragen wird und über REST-API ausgelesen werden kann. Die Ausnahme dabei ist Modbus-TCP Protokoll, hier werden die Werte immer genau so übertragen wie auf dem CAN-Bus und die ggf. notwendige Umrechnung muss auf dem Modbus-Master stattfinden. Sonst gilt, dass der auf dem CAN-Bus übertragener Wert erst mit dem Koeffizient multipliziert wird und dann dazu ein Offset addiert wird. Der Koeffizient und der Offset können ganzzahlig oder nicht sein, positiv oder negativ. Bei Änderungsanfragen ist die Umrechnung umgekehrt, also wird aus dem für die Änderung eingesteuerten Wert der Offset subtrahiert und dann wird das Ergebnis durch Koeffizient dividiert. Das Ergebnis wird dann in Binärform über CAN-Bus an das Gerät übertragen. Diese Umrechnungen sind deswegen notwendig. Die notwendigen Einstellungen für Koeffizient/Offset kann man rausfinden entweder indem man die Werte, die CAN-Gateway rausgibt mit den auf dem Bedienmodul angezeigten Werten vergleicht oder wenn man die vom Hersteller zur Verfügung gestellten Parameterlisten studiert. Offset ist fast immer (oder wirklich immer?)=0. Koeffizient muss man in Abhängigkeit von der Anzahl der Dezimalstellen (Spalte „Decimal“ in diesen Listen) einstellen. Wenn da der Wert 0 steht, dann ist der Koeffizient=1. Bei Decimal=1 ist der Koeffizient=0.1 (oft bei Temperaturen), bei Decimal=2 ist der Koeffizient=0.01 und bei Decimal=3 ist der Koeffizient=0.001.

Die Parameter werden auf der Seite „Einstellungen“ des WEB-Interfaces konfiguriert. Dazu wird im Feld Parametereinstellungen für jeden Parameter eine Zeile eingetragen, die mit „param=“ beginnt. Danach kommen die Konfigurationsdaten, jeweils getrennt durch ein Semikolon. Insgesamt können bis zu 40 Parameter (bis SW Version 22.001 nur 20 Parameter) konfiguriert werden. Unter diesem Feld ist auch eine kurze Beschreibung aller Konfigurationsdaten gegeben. In diesem Beispiel im Bild

werden 9 Parameter konfiguriert. Die ersten 7 davon sind die User-Level Passwörter, die aus dem Bedienmodul ausgelesen werden. Dann wird noch ein Parameter „Zeit bis zum Abschalten“ im Bedienmodul konfiguriert (siehe Beispiel im Kapitel „Allgemeine Informationen zu Geräteparametern“). Und als letztes wird ein Helligkeit/Status Parameter definiert (siehe Beispiel im Unterkapitel „Hinweise zu Listen“), allerdings als Rohdaten-Format (RAW, sinnvoller als Index einer Liste wäre hier der Format „U8“).



10.1 Datentypen

Information über den Datentyp beeinflusst das, wie der in Binärform auf dem CAN-Bus übertragene Wert in einen Wert umgerechnet wird, welcher dann über WEB-Interface einsehbar ist, über MQTT Protokoll übertragen wird und über REST-API ausgelesen werden kann. Die Ausnahme dabei ist Modbus-TCP Protokoll, hier werden die Werte immer genau so übertragen wie auf dem CAN-Bus und die ggf. notwendige Umrechnung muss auf dem Modbus-Master stattfinden.

RAW: Der Wert wird in Hexadezimal Darstellung ausgegeben. Also z.B. wenn auf dem CAN-Bus ein Parameterwert als Reihenfolge von 4 Bytes 0x31 0x32 0x61 0x5A übertragen wird, wird das als eine Zeichenfolge „3132615A“ dargestellt. Maximale Byte-Anzahl ist 64.

STR: Der Wert wird als eine ASCII kodierte Zeichenfolge interpretiert. Also z.B. wenn auf dem CAN-Bus ein Parameterwert als Reihenfolge von 4 Bytes 0x31 0x32 0x61 0x5A übertragen wird, wird das als eine Zeichenfolge „12aZ“ dargestellt, weil 0x31 in ASCII Kodierung „1“ entspricht, 0x32=“2“, 0x61=“a“ und 0x5A=“Z“. Maximale Länge der Zeichenfolge ist 64.

U8 (unsigned 8-Bit): ein vorzeichenloser Byte, also ein Wert im Bereich 0 bis 255.

S8 (signed 8-Bit): ein vorzeichenbehafteter Byte, also ein Wert im Bereich -128 bis 127.

U16 (unsigned 16-Bit): ein vorzeichenloser Wort (2 Bytes), also ein Wert im Bereich 0 bis 65535.

S16 (signed 16-Bit): ein vorzeichenbehafteter Wort (2 Bytes), also ein Wert im Bereich -32768 bis 32767.

U32 (unsigned 32-Bit): ein vorzeichenloser doppelter Wort (4 Bytes), also ein Wert im Bereich 0 bis 4294967295.

S32 (signed 32-Bit): ein vorzeichenbehafteter doppelter Wort (4 Bytes), also ein Wert im Bereich -2147483648 bis 2147483647.

Die Bereiche oben sind ohne Berücksichtigung der eingestellten Koeffizient/Offset.

10.2 Hinweise zu Listen

Einige Parameter sind als sogenannten Listen definiert. Eine Liste in diesem Sinne ist eine Zuordnung von Nummern 0, 1, 2, ... usw. zu jeweils einer bestimmten Bedeutung, die zum Beispiel durch ein Wort oder eine Phrase beschrieben wird. Als Beispiel kann Helligkeit-Modus am Bedienmodul betrachtet werden. Der Parameter Helligkeit-Modus gehört zu Funktionsgruppe 80, Funktionsnummer 1 und hat Datapoint-ID = 0. Am Bedienmodul können verschiedene Modi gewählt werden, wie z.B. „Zeitgesteuert“. Über CAN-Bus wird als Parameter jedoch nicht das Wort „Zeitgesteuert“ übertragen, sondern diesem Wort entsprechende Zahl=2. Wenn man den Modus ändern will, muss man dem Modus entsprechende Zahl kennen und diese als Änderung einsteuern.

10.3 Beispiele

10.3.1 Beispiel 1: Komfortlüftung – Feuchte Sollwert auslesen und einstellen

Folgende Zeile ist ein Beispiel, um den aktuellen Sollwert für die Feuchte in einem HomeVent Gerät auslesen und einzustellen (Gerätetyp=8; Geräteadresse=8, Funktionsgruppe=50; Funktionsnummer=0; Datapoint-ID=40687; Lesen und Schreiben (w); MQTT topic=feuchte_sollwert; Auslesen jede 10s; Datentyp: U8; Koeffizient=1; Offset=0):

```
param=8;8;50;0;40687;w;feuchte_sollwert;10;U8;1.000000;0.000000;
```

10.3.2 Beispiel 2: Komfortlüftung – Temperatur Abluft

Folgende Zeile ist ein Beispiel, um die aktuelle Abluft Temperaturn auszulesen (Gerätetyp=8; Geräteadresse=8, Funktionsgruppe=50; Funktionsnummer=0; Datapoint-ID=37602; nur Lesen (r); MQTT topic=temp_abluf; Auslesen jede 10s; Datentyp: S16; Koeffizient=0.1; Offset=0):

```
param=8;8;50;0;37602;r;temp_abluf;10;S16;0.100000;0.000000;
```

10.3.3 Beispiel 3: Komfortlüftung – Temperatur Außenluft

Folgende Zeile ist ein Beispiel, um die aktuelle Außenluft Temperaturn auszulesen (Gerätetyp=8; Geräteadresse=8, Funktionsgruppe=50; Funktionsnummer=0; Datapoint-ID=0; nur Lesen (r); MQTT topic=temp_aussen; Auslesen jede 10s; Datentyp: S16; Koeffizient=0.1; Offset=0):

```
param=8;8;50;0;0;r;temp_aussen;10;S16;0.100000;0.000000;
```

10.3.4 Beispiel 4: Raumbedienmodul – Kennwort für UserLevel 7 lesen/ändern

Folgende Zeile ist ein Beispiel, um den Kennwort für UserLevel=7 aus dem Bedienmodul auszulesen und zu ändern (Gerätetyp=16; Geräteadresse=8, Funktionsgruppe=89; Funktionsnummer=1; Datapoint-ID=7; Lesen und Ändern (w); MQTT topic=password_level7; Auslesen jede 60s; Datentyp: STR; Koeffizient=1 und Offset=0 können beliebig sein, da irrelevant):

```
param=16;8;89;1;7;r;password_level7;60;STR;1;0;
```

10.3.5 Beispiel 5: Komfortlüftung – Betriebsmodus

Folgende Zeile ist ein Beispiel, um den Betriebsmodus auszulesen und zu ändern (Gerätetyp=8; Geräteadresse=8, Funktionsgruppe=50; Funktionsnummer=0; Datapoint-ID=40650; Lesen und Ändern (w); MQTT topic=password_level7; Auslesen jede 60s; Datentyp: U8; Koeffizient=1 und Offset=0):

```
param=8;8;50;0;40650;w;betriebswahl_lueftung;60;U8;1.000000;0.000000;
```

Betriebsmodus wird durch die Nummer 0 bis 5 kodiert. Laut dieser Datei (<http://www.hoval.com/misc/TTE/TTE-GW-Modbus-datapoints.xlsx>) sind folgende Modi möglich (gleiche Liste sieht man über Bedienmodul): Standby=0, Woche 1=1; Woche 2=2; Konstant=4; Sparbetrieb=5.

10.3.6 Beispiel 6: Komfortlüftung – Datum/Uhrzeit

Folgende zwei Zeilen sind ein Beispiel, wie man Datum/Uhrzeit auslesen oder ändern kann:

```
param=8;8;0;0;2070;w;Datum;10;U16;1.000000;0.000000;
```

```
param=8;8;0;0;2072;w;Uhrzeit;10;U16;1.000000;0.000000;
```

Dabei ist „Uhrzeit“ die Anzahl der Sekunden seit Tagesbeginn; „Datum“ ist die Anzahl der Tage seit 1.1.1900 (Schaltjahre werden korrekt berücksichtigt).

11 WEB-Interface: Parameter-Konfiguration Generator

Dieser Generator hilft die Konfigurationszeilen für Parameter zu generieren. In einer Tabelle können aus Listen die Parameter ausgewählt werden und für diese werden dann Konfigurationszeilen generiert, die man dann kopieren und auf der Seite der Einstellungen einfügen und somit für die Konfiguration des CAN-Gateways nutzen kann. Die Auswahllisten beinhalten viele bekannte Parameter für die verschiedenen Geräte.

Nummer	Gerätetyp	Geräteadresse	Funktionsgruppe	Funktionsnummer	Datenpunkt	Update Rate	R/W	Datentyp	Koeffizient	Offset
1	(0) TTE-WEZ (Wärmeerzeuger)	1	(1) Heizkreis	(0) Heizkreis 1	(2) Vorlauf-Ist [gradC]	10	r	S16	0.1	0
2	(0) TTE-WEZ (Wärmeerzeuger)	1	(1) Heizkreis	(0) Heizkreis 1	(1001) Raum-Soll [gradC]	30	r	S16	0.1	0
3	(8) TTE-HV (HomeVent)	1	(50) Lueftung	(0) Lueftung	(37602) Temperatur Abluft [gradC]	30	r	S16	0.1	0
4	(8) TTE-HV (HomeVent)	1	(50) Lueftung	(0) Lueftung	(39600) Luftqualitaet Regulierung	60	w	U8	1	0
5	Aus vorheriger Zeile übernehmen	1				10				
6	not used	1				10				
7	not used	1				10				
8	not used	1				10				
9	not used	1				10				
10	not used	1				10				
11	not used	1				10				
12	not used	1				10				
13	not used	1				10				
14	not used	1				10				
15	not used	1				10				
16	not used	1				10				
17	not used	1				10				
18	not used	1				10				
19	not used	1				10				
20	not used	1				10				

Parameter Konfiguration aus der Tabelle generieren

```

varam=0;1;1;0;r;Vorlauf_Ist_gradC;10;S16;0.1;0;
varam=0;1;1;0;1001;r;Raum_Soll_gradC;30;S16;0.1;0;
varam=8;1;50;r;Temperatur_Abluft_gradC;30;S16;0.1;0;

```

12 WEB-Interface: Liste der Parameter

Im WEB-Interface des CAN-Gateways gibt es die Möglichkeit, die konfigurierten Parameter anzeigen zu lassen und die Änderungsanfragen auch zu generieren. Diese Seite wird nicht periodisch upgedatet. Dafür muss man auf „Neu laden“ klicken. Um ein Wert zu ändern (bzw. genauer gesagt eine Änderungsanfrage zu generieren), muss der neue Wert in das entsprechende grüne Feld eingegeben werden und dann muss man auf entsprechende Schaltfläche „Wert setzen“ klicken.

The screenshot shows a browser window titled "CAN Gateway: Parameterwerte". The address bar says "cangatew". Below the title, there's a link "[<< Hauptseite](#)". A warning message reads: "Achtung: diese Webseite verursacht hohe Prozessorlast für CANGateway und sollte nur für die Einrichtung und Debugging, jedoch nicht im normalen Betrieb genutzt werden." A green button labeled "Neu laden" is visible. The main content is a table with columns: Nummer, MQTT topic, Modbus Basisadresse, Wert, Rohwert (hex), Alter, s, Neuer Wert (um zu ändern), and a "Wert setzen" button. Rows represent parameters like password_level1 to password_level7, delay_t, status, and two entries labeled "NOT USED".

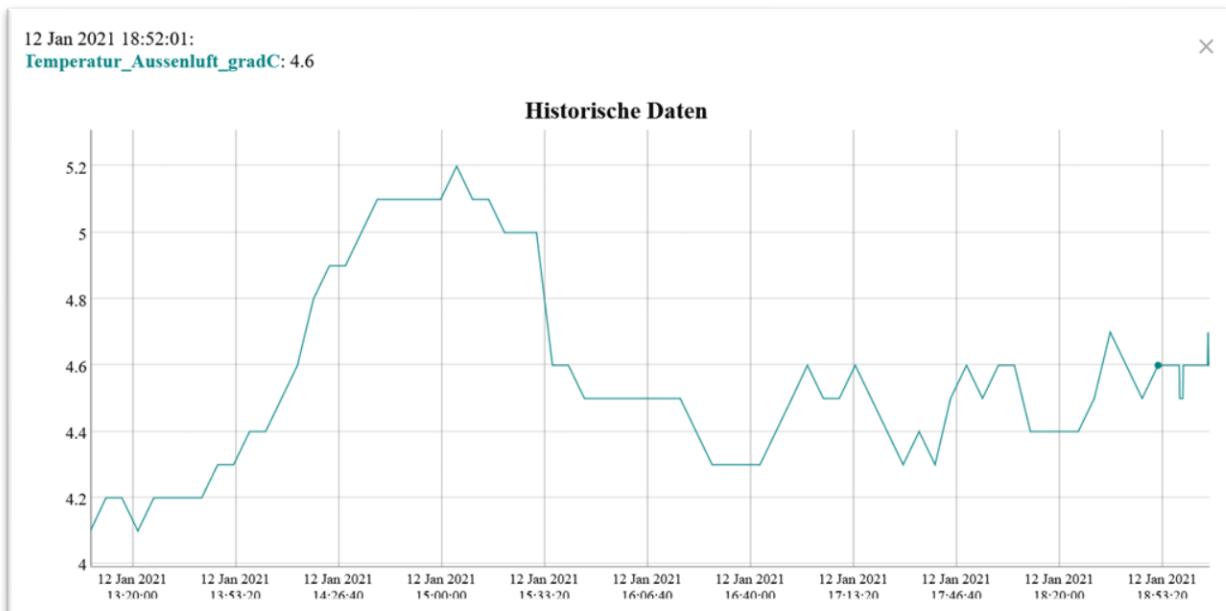
Nummer	MQTT topic	Modbus Basisadresse	Wert	Rohwert (hex)	Alter, s	Neuer Wert (um zu ändern)	
1	password_level1	0	[redacted]	[redacted]	3	[redacted]	Wert setzen
2	password_level2	10	[redacted]	[redacted]	3	[redacted]	Wert setzen
3	password_level3	20	[redacted]	[redacted]	3	[redacted]	Wert setzen
4	password_level4	n.a.	[redacted]	[redacted]	3	[redacted]	Wert setzen
5	password_level5	n.a.	[redacted]	[redacted]	3	[redacted]	Wert setzen
6	password_level6	n.a.	[redacted]	[redacted]	3	[redacted]	Wert setzen
7	password_level7	n.a.	[redacted]	[redacted]	14	[redacted]	Wert setzen
8	delay_t	n.a.	60	3C	3	[redacted]	Wert setzen
9	status	n.a.	02	02	2	[redacted]	Wert setzen
10	NOT USED	n.a.				[redacted]	Wert setzen
11	NOT USED	n.a.				[redacted]	Wert setzen

In SW seit Version 21.002 gibt es auch die Möglichkeit die historischen Werte der Parameter anzuschauen (als Tabelle und als Graph, siehe auch Kapitel „WEB-Interface: Einstellungen“). Dafür gibt es für jeden Parameter zwei kleine grüne Buttons „H“ und „P“:

This screenshot shows the same table as above, but with additional small green buttons labeled "H" and "P" next to the first column. These buttons likely serve as links to historical data or graphing tools for each parameter.

Nummer	MQTT topic	Modbus Basisadresse	Wert	Rohwert (hex)	Alter, s	Neuer Wert (um zu ändern)	
1 H P	Temperatur_Aussenluft_gradC	0	-0.1	FFFF	28	[redacted]	Wert setzen
2 H P	Feuchtigkeit_Abluft_percent	10	65	41	9	[redacted]	Wert setzen

Historische Daten werden für Parameter vom Typ STR und RAW nicht gespeichert.



13 WEB-Interface: Liste der Geräte

Wenn ein Gerät auf dem CAN-Bus irgendwas sendet, wird es vom CAN-Gateway detektiert und er speichert das Gerät in der Liste der gefundenen Geräte, die dann über WEB-interface aufgerufen werden kann.

The screenshot shows a web browser window titled "CAN Gateway: Gefundene Geräte". The address bar displays "cangateway.lo...". The page content is titled "CAN Gateway: Liste der gefundenen Geraete (alle seit Start)". Below this, there are two green buttons: "Update" and "Liste leeren". A table follows, with columns: Nummer, Gerätetyp, Gerätetyp (als dezimal), and Geräteadresse (dezimal). The first row contains the value "16" under "Gerätetyp (als dezimal)" and "8" under "Geräteadresse (dezimal)". The other rows are empty.

Nummer	Gerätetyp	Gerätetyp (als dezimal)	Geräteadresse (dezimal)
1	TTE-BM (Bedienmodul)	16	8
2			
3			
4			
5			
6			
7			
8			
9			
10			

14 WEB-Interface: Status Informationen

Über WEB-Interface kann CAN-Gateway Status abgerufen werden, wo verschiedenen Informationen zusammen gefasst sind, die bei Problemlösungen etc. helfen können.

The screenshot shows a web browser window titled "CAN Gateway". The main content area displays the "CAN Gateway: Status" page. It includes a "Neu laden" button, MAC Address, IP Address, and several status messages: MQTT Status (verbunden), Modbus (Parameter 1 bis 3 auf dem Bus verfügbar), CAN Gateway (200756 Byte(s) RAM noch verfügbar), CAN Gateway (Läuft seit 310 s.), WLAN (verbunden seit 310 s.), and MQTT (verbunden seit 310 s.). Below this, there is a section titled "CAN-Bus Information:" containing a bulleted list of CAN bus statistics. At the bottom, there is a section titled "Im Durchschnitt über 10 Sekunden:" with a bulleted list of task execution times.

CAN Gateway: Status

[<< Hauptseite](#)

Neu laden

MAC Adresse: [REDACTED]

IP Adresse: [REDACTED]

MQTT Status: **verbunden** mit dem MQTT server

Modbus: Parameter 1 bis 3 sind auf dem Bus verfügbar.

CAN Gateway: 200756 Byte(s) RAM noch verfügbar

CAN Gateway: Läuft seit 310 s.

WLAN: verbunden seit 310 s.

MQTT: verbunden seit 310 s.

CAN-Bus Information:

- letzte Nachricht wurde vor 0 s empfangen.
- empfangen 24 Nachrichten in vorletzten 10 s.
- 0 verlorene CAN Nachrichten
- CAN controller status= RUNNING
- RX Error Counter in CAN controller = 0x00
- TX Error Counter in CAN controller = 0x00

Im Durchschnitt über 10 Sekunden:

- loop Task: [Dauer max; Dauer avg; Interval max; Interval avg] = [3530; 81; 3536; 82] us
- can Task: [Dauer max; Dauer avg; Interval max; Interval avg] = [323; 3; 3602; 82] us
- mbus Task: [Dauer max; Dauer avg; Interval max; Interval avg] = [142; 14; 3401; 82] us
- decoder Task: [Dauer max; Dauer avg; Interval max; Interval avg] = [303; 56; 102513; 101011] us

15 WEB-Interface: SD Karte Info/Status

[neu ab SW 26.001, nur in Verbindung mit HW V5 und höher oder Olimex board]

Hier wird der aktuelle Status der SD-Karte angezeigt. SD-Karte kann grundsätzlich im laufenden Betrieb ein- und ausgesteckt werden. Es wird aber dringend empfohlen, vor dem ausstecken die SD-Karte „unmontieren“. Es werden SD und SDHC Micro SD-Karten unterstützt (keine SDXC!). Die SD-Karte muss vorher mit FAT32 formatiert werden.

Die Daten, wenn aktiviert, werden in Dateien gespeichert. Der Dateiname wird wie folgt gebildet:

Jahr_Monat_Tag_Stunden_Minuten_Sekunden_Konfigurationsnummer_0000.csv

Konfigurationsnummer ist einfach eine Zahl, die bei jeder Änderung der Konfiguration um 1 erhöht wird.

The screenshot shows a web browser window with the URL `cangateway.local/sdcard.html`. The title bar says "CAN Gateway: Status SD-Karte". Below it, there's a link "[<< Hauptseite](#)". A green button labeled "Neu laden" is visible. The main content area displays the following information:

- SD Karte eingesteckt: ja
- SD Karte montiert: ja [Unmontieren](#)
- SD Karte typ: SDHC
- SD Karte Größe: 15812526080 Bytes (15080 MBytes)
- Gesamt verfügbar: 15804137472 Bytes (15072 MBytes)
- Davon bereits genutzt: 32309248 Bytes (31 MBytes)
- Aktuelle Log-Datei: /2021_03_19_18_30_48_0008_0000.csv (Zeilen geschrieben: 411)

Below this, it says "Dateien im Wurzel-Verzeichnis:" followed by a table:

Nummer	Name	Größe, Bytes
1	1970_01_01_01_00_03_0009_0000.csv	28378
2	2021_02_09_07_57_18_0009_0000.csv	531022
3	2021_02_09_08_11_20_0009_0000.csv	531271
4	2021_02_09_08_25_21_0009_0000.csv	195233
5	2021_02_09_08_30_29_0010_0000.csv	1026815
<	2021_02_10_12_01_26_0010_0000....	?????

Der Format der Daten wird ist in den Einstellungen konfigurierbar.

Speicherung auf SD-Karte verwenden:

- Profil:
- Zeitformat:
- Periode: (nur für konfigurierte Parameter)
- Trennzeichen zwischen Spalten:
- Dezimalzeichen: (nur für konfigurierte Parameter)
- End-of-Line Style:
- Parameter, die länger als 2 Minuten nicht empfangen wurden speichern als: (nur für konfigurierte Parameter)
- Maximal Zeilen pro Datei: (Sobald erreicht, wird neue Datei geöffnet.)

16 WEB-Interface: CAN Datenlogger

Alle vom CAN-Gateway empfangenen CAN-Nachrichten können über Web-Interface angezeigt werden:

CAN Messages:

[<< Hauptseite](#)

Als CSV Datei herunterladen Autoscroll

```
T 1FE047FF 4 01 50 04 08
T 1FE047FF 6 01 52 04 02 00 00
T 1FE047FF 5 01 4C 12 01 00
T 1F6047FF 8 11 33 74 00 02 00 00 00
T 1EA047FF 6 33 00 00 00 38 43
T 1F6047FF 8 11 34 42 59 01 00 01 31
```

Die neuen Nachrichten werden automatisch jede 5 s nachgeladen und unten angehängt. Man kann alles in eine CSV Datei exportieren. Die erste Buchstabe „t“ oder „T“ bedeutet eine Standard- oder Extended-Frame. Danach folgt CAN-ID, DLC (Daten-Länge) und 1 bis 8 Datenbytes. Nur Daten-Nachrichten werden aufgezeichnet. RTR- und Error-Frames werden ignoriert.

17 WEB-Interface: Parameter Logger

Alle vom CAN-Gateway empfangenen Parameter können über Web-Interface angezeigt werden. (In diesem Beispiel sind die empfangenen User-Level Kennwörter geschwärzt.)

Parameter Logger

cangateway.lo...

Parameter Logger:

<< Hauptseite

Als CSV Datei herunterladen Autoscroll

Format: (Gerätetyp, Geräteadresse, Funktionsgruppe, Funktionsnummer, Datapoint-ID)=value

Alle Zahlen sind in HEX format!

(10,08,59,01,0001) = [REDACTED]
(10,08,59,01,0002) = [REDACTED]
(10,08,59,01,0003) = [REDACTED]
(10,08,59,01,0006) = [REDACTED]
(10,08,50,01,0000) = 02
(10,08,59,01,0004) = [REDACTED]
(10,08,59,01,0005) = [REDACTED]
(10,08,50,01,0001) = 37

Die neu empfangenen Parameter werden automatisch jede 5 s nachgeladen. Man kann alles in eine CSV Datei exportieren.

18 WEB-Interface: KNX IP Einstellungen (ab SW 29.001)

CAN-Gateway unterstützt ab SW Version 29.001 KNX IP Multicasting (aktuell kein Tunneling, kein KNX IP Secure und nicht alle Datapoint Typen werden unterstützt).

Es wird auch dringend empfohlen MQTT Protokoll auszuschalten (in Haupteinstellungen), wenn KNX IP verwendet wird. Ansonsten kann wegen RAM Mangel am CAN-Gateway zu Problemen (reboots, WEB UI nicht erreichbar etc.) kommen.

Sie müssen dazu ein KNX IP Router haben (physikalisch oder knxd Dienst auf einem Linux Rechner). Die Einstellung der KNX IP Parameter erfolgt über CAN-Gateway Web-Interface und nicht via ETS Software. In ETS Software kann man CAN-Gateway als ein Dummy Gerät integrieren.

CAN-Gateway hat eine Option auf SERVICE_REQUEST Anfragen von ETS5 zu reagieren, so dass Sie KNX IP auch dann testen können, wenn Sie kein KNX IP Router / knxd installiert haben.

Wenn man sich mit KNX auskennt, dann sind die meisten Einstellungen selbsterklärend. Detaillierung ist nur für Parameter „Koeffizient“ und DPT (Datapoint Type) notwendig:

Allgemeine Einstellungen		KNX spezifisch							
Nummer	Name	Lesen/Schreiben	Verfügbar	Gruppenadresse	DPT	DPT (KlarTEXT)	Koeffizient	Lesen/Schreiben	Hysterese Tiefe
0	Temperatur_Aussenluft_gradC	R	Ja	1 / 1 / 3	14 . 1	4-Byte double (general or any specific)	1.00000	Nur lesen	2
1	Feuchtigkeit_Abluft_percent	R	Ja	1 / 1 / 4	14 . 0	4-Byte double (general or any specific)	1.00000	Nur lesen	1
2	Temperatur_Abluft_gradC	R	Ja	1 / 1 / 5	14 . 1	4-Byte double (general or any specific)	1.00000	Nur lesen	2

Koeffizient: sollte in meisten Fällen „1.0“ sein. Mit diesem Koeffizient werden die realen Werte multipliziert bevor diese über KNX IP gesendet werden. Von „1.0“ abweichende Werte können nach Bedarf in einigen Sondersituationen verwendet werden.

DPT: Es werden DPT(x,y) mit dem folgenden Haupttyp „x“ unterstützt:

- DPT(1,y) [1-Bit Boolean]
- DPT(5,y) [1-Byte unsigned]
- DPT(6,y) [1-Byte signed]
- DPT(7,y) [2-Byte unsigned]
- DPT(8,y) [2-Byte signed]
- DPT(9,y) [2-Byte double]

- DPT(12,y) [4-Byte long unsigned]
- DPT(13,y) [4-Byte long signed]
- DPT(14,y) [4-Byte double]

Der Nutzer kann generell jedem Parameter fei ein DPT zuordnen. Es wird empfohlen für physikalische Werte (wie Temperatur, Feuchtigkeit etc.) DPT(14, y) zu verwenden. Subtyp wird in aktueller Implementierung ignoriert, es erfolgt also z.B. keine dementsprechende Skalierung des Wertes, was aber in meisten Fällen ausreichend ist.

[neu in V 29.310] Es gibt die Option auszuwählen, wann die Sensorwerte über KNX vom CAN-Gateway gesendet werden:

- Immer
- Nur bei Änderung
- Nur bei Änderung und alle 5 Minuten
- Nur bei Änderung und alle 10 Minuten
- Nur bei Änderung und alle 30 Minuten

Unabhängig von der gewählten Option, werden die Werte einmal beim Starten/Reboot von CAN-Gateway gesendet sowie wenn CAN-Gateway wieder ans WLAN/LAN angebunden wird (z.B. nachdem LAN Kabel zeitweise abgesteckt wurde etc.).

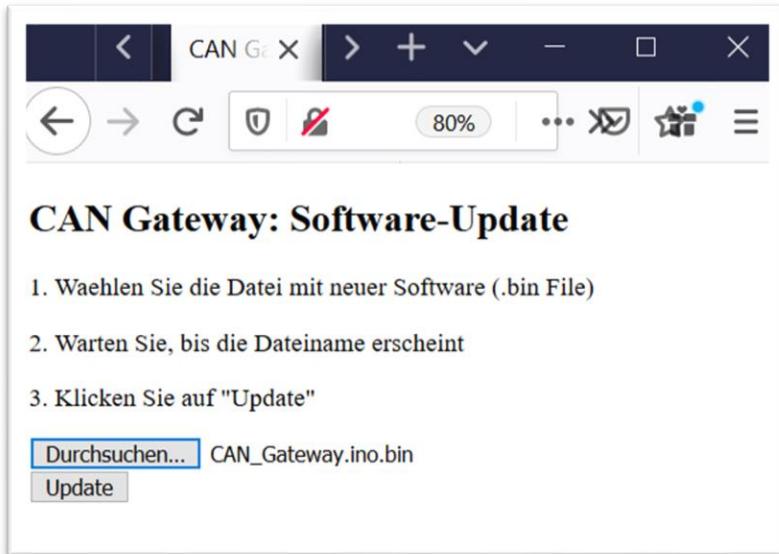
Für jeden Parameter gibt es außerdem die Möglichkeit zu definieren, ob für Senden „bei Änderung“ auch eine Hysterese verwendet werden soll. Hysterese Tiefe von „0“ bedeutet, dass der Wert bei jeder Änderung neu gesendet wird. Hysterese Tiefe von „1“ bedeutet, dass die Änderung um einen Schritt ignoriert wird. Z.b. wenn zuletzt der Temperaturwert von 20.5 °C gesendet wurde, wird bei Temperaturerhöhung erst bei 20.7 °C neu gesendet, jedoch nicht bei 20.6 °C.

19 WEB-Interface: Software Update

CAN-Gateway Software kann über WEB-Interface upgedatet werden.

DEMO Version kann auf eine vollfunktionsfähige Version upgedatet werden und auch umgekehrt.

Achtung: Software Bruch zwischen Version 29.101 und 29.200. Aufgrund geänderter Basissoftware, kann die Version 29.101 oder niedrigere NICHT per WEB-Interface auf die Software 29.200 oder höher upgedatet werden. Für Update bitte updateen Sie erst auf die neue DEMO-Version über den web-basierten Installer (<https://wladwnt.github.io/>) und dann kann die DEMO-Version per WEB-Interface auf Vollversion upgedatet werden.



20 Übertragung der Parameter über MQTT Protokoll

Sie brauchen in Ihrem Hausautomatisierungssystem einen MQTT Broker um die Parameterwerte über MQTT Protokoll zu übertragen und dann in Ihrer Hausautomatisierungssoftware nutzen zu können. CAN-Gateway unterstützt aktuell nur eine unverschlüsselte MQTT Kommunikation über TCP. Also jeder, der mit Ihrem WLAN verbunden ist, kann dann auf diese Daten zugreifen und auch z.B. diese Daten manipulieren. Deswegen CAN-Gateway soll nur in einem sicheren, privaten und nicht öffentlichen WLAN eingesetzt werden und vom Internet aus nicht zugänglich sein, also keine Portweiterleitung etc. Ihr MQTT Broker soll eine Authentifizierung mittels Login/Password unterstützen und unverschlüsselte Kommunikation über TCP zulassen.

Beim MQTT Protokoll muss jede zu übertragende Information (in unserem Fall sind es Parameter) einen eindeutigen Namen besitzen. Diesen Namen nennt man „Topic“. CAN-Gateway ist so eingestellt, dass für jeden konfigurierten Parameter ein Topic erzeugt wird, welches aus einem für alle Parameter gemeinsamen Teil besteht, folgend mit einem Schrägstrich (/ -Zeichen) und dann noch konfigurierbarer Parametername dazu. Der gemeinsame Teil wird in allgemeinen Einstellungen definiert. Der Parameter-spezifische Name in Parameter Einstellungen. Z.B. ist dann der resultierende Topic „cangateway/helligkeit“ oder „cangateway/temperatur“ usw. Über dieses Topic kann dann der entsprechende Parameter ausgelesen werden. Parameteränderungen müssen über den Topic mit dem gleichen Namen aber zusätzlich am Ende noch mit „/set“ eingesteuert werden. Also z.B. über „cangateway/temperatur/set“. Das geht natürlich dann nur für Parameter, die mit dem Attribut „w“ (Write) definiert sind.

21 Übertragung der Parameter über Modbus TCP

21.1 Hinweise zu Modbus TCP Kommunikation

CAN-Gateway unterstützt aktuell nur eine unverschlüsselte Modbus TCP Kommunikation. Also jeder, der mit Ihrem WLAN verbunden ist, kann dann auf diese Daten zugreifen und auch z.B. diese Daten manipulieren. Deswegen CAN-Gateway soll nur in einem sicheren, privaten und nicht öffentlichen WLAN eingesetzt werden und vom Internet aus nicht zugänglich sein, also keine Portweiterleitung etc.

21.2 Modbus Adressen

Die Adressen der Parameter auf dem Modbus sind wie folgt definiert. Grundsätzlich bekommt jeder Parameter entsprechend seinem Nummer (1 bis 20) eine Adresse, die nach folgender Formel berechnet wird:

$$\text{Basisadresse} = (\text{Nummer}-1)*10$$

Beim Modbus gilt eine Adresse für einen 16-Bit Wert (ein Register). Daher gilt die oben beschriebene Adressen-Zuordnung exakt nur für Parameter, die in einen Register reinpassen, also Parameter mit Datentyp: U8, S8, U16 und S16.

Parameter mit Datentyp U32 oder S32 bekommen jeweils zwei Adressen (also werden durch zwei Register dargestellt). Die Basisadresse beschreibt die zwei „höchsten“ Bytes (also inklusiv MSB), die nächste Adresse (also Basisadresse+1) beschreibt die zwei „niedrigsten“ Bytes (also inklusiv LSB). Z.B. wenn ein Parameter Nummer 5 vom Typ U32 ist, dann hat er die Adressen 40 und 41.

Parameter mit Datentyp RAW und STR bekommen jeweils 10 Adressen, werden also in 10 Registern übertragen. Es können also maximal 20 Zeichen (2 Zeichen pro ein Register) beim Typ STR bzw. 20 Bytes an Rohdaten über Modbus ausgelesen werden. Der Rest, falls vorhanden, wird quasi abgeschnitten.

Evtl. sind nicht alle Parameter auf dem Modbus verfügbar. Aufgrund Auslastung des CANGateways können insgesamt nur 32 Register genutzt werden. Parameter mit Datentyp U8, S8, U16 und S16 verbrauchen jeweils ein Register für Lesen und ggf. noch ein fürs Schreiben. Parameter mit Datentyp U32 oder S32 verbrauchen jeweils zwei Register für Lesen und ggf. noch zwei fürs Schreiben.

Parameter mit dem Datentyp RAW und STR verbrauchen jeweils 10 Register. Es wird empfohlen die Parameter, die auf dem Modbus nicht unbedingt notwendig sind, am Ende der Parameterliste zu definieren.

21.3 Register Typen

Alle Parameter sind als Register vom Typ „Input Register“ definiert und können mit der Modbus Funktion „Read Input Registers (0x04)“ ausgelesen werden.

Parameter, die auch geschrieben werden können (mit dem Attribut „w“), sind zusätzlich als Register vom Typ „Holding Register“ definiert und können mit Modbus Funktionen „Write Single Register (0x06)“ und „Write Multiple Registers (0x10)“ geschrieben werden. Wenn Holding Register mit der Funktion „Read Holding Register (0x03)“ ausgelesen wird, bekommt man NICHT den aktuellen Parameterwert, sondern den letzten Wert, den man geschrieben hat oder 0.

Parameter mit Datentyp RAW und STR können über Modbus generell nur ausgelesen und nicht geändert werden, unabhängig vom r/w Attribut.

21.4 Übertragungswerte

Alle Parameter werden als binäre Rohwerte übertragen. Es findet KEINE Korrektur mit Koeffizient/Offset wie bei MQTT statt. Diese Korrektur kann normalerweise auf Ihrem Modbus Master erfolgen und Sie müssen es da richtig einstellen.

Parameter mit Datentyp U8 und S8 werden zu entsprechenden 16-bit Werten umgewandelt, da die Modbus Register 16-bit lang sind.

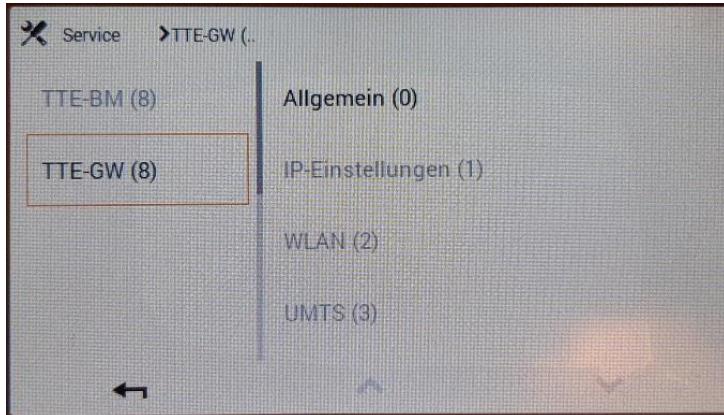
Parameter mit Datentyp U32 und S32 werden, wie oben beschrieben, jeweils in zwei Register übertragen. Sie müssen ggf. auf Ihrem Modbus Master die korrekte Byte- und Register-Reihenfolge einstellen.

RAW und STR Parameter werden identisch behandelt und generell als Rohdaten übertragen. Das bedeutet zum Beispiel, dass eine Zeichenfolge „Eingeschaltet“ als 16-bit HEX Zahlen 0x4569, 0x6e67, 0x6573, 0x6368, 0x616c, 0x7465, 0x7400 übertragen wird (ASCII kodiert, ggf. mit 00 erweitert). Die ersten zwei Zeichen „Ei“ (0x4569) werden im Register mit der Adresse wie oben im Formel übertragen, die nächsten zwei „ng“ (0x6e67) sind im nächsten Register (Adresse+1) usw.

22 Steuerung über Bedienmodul

Wenn CAN-Gateway über CAN-Bus mit einem Bedienmodul (Raumbedienmodul, TTE-BM) verbunden ist, kann CAN-Gateway vom Bedienmodul als Gateway (TTE-GW) erkannt werden und kann darüber neu gestartet oder auf die Werkseinstellungen zurückgesetzt werden. Außerdem werden einige Informationen über die Verbindung zum WLAN (Netzwerkname, IP Adresse, Status, Signal-Qualität) angezeigt.

Die Voraussetzung ist, dass über WEB-Interface in den Einstellungen die Option „Ermöglichen CAN Gateway über Bedienmodul zu steuern“ aktiviert wurde. Dann erscheint CAN-Gateway als TTE-GW Gerät im Bedienmodul/Hauptmenu „Service“:



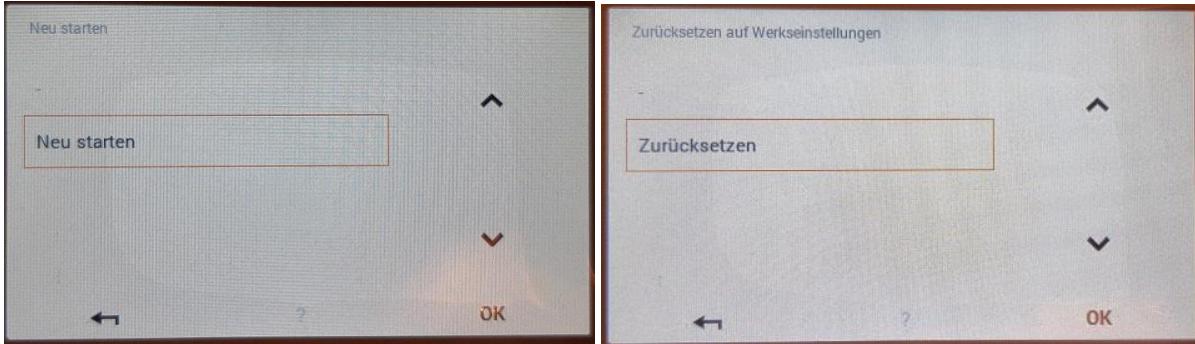
Durch Klicken auf **Allgemein(0)** → **Allgemein Gateway** → **Information (1)**, dann scrollen nach unten auf **Alle Datenpkt.**, dann Klicken auf die Liste der Datenpunkte rechts kann man auf die folgenden 5 Optionen zugreifen:

- Zurücksetzen auf Werkseinstellungen
- Neu starten
- Gateway-CAN-ID (Das ist die CAN-Gateway Adresse)
- IP-Adresse (vom CAN-Gateway im WLAN)
- Geräteidentifikation (anzeigt immer “CAN-Gateway”)

The image contains two side-by-side screenshots of a data list interface. Both screenshots show the same structure with identical data. The top part of both screenshots shows the navigation path: Service > TTE-GW (..) > Allgemein (0) > Allgemein Ga.. > Alle Datenpkt. The bottom part shows a table with five rows of data:

	00-010	-
Zurücksetzen auf Werkseinstellungen	00-010	-
Neu starten	00-015	-
Gateway-CAN-ID	00-025	8
IP-Adresse	30-009	192.168.20.76

Um CAN-Gateway neu zu starten oder auf Werkseinstellungen zurückzusetzen, muss man in der entsprechenden Zeile auf „-“ klicken, dann auf „Neu starten“ bzw. „Zurücksetzen“ und dann auf „OK“:



Zurücksetzen ist allerdings nur verfügbar, wenn man im UserLevel von mindestens 3 sich befindet.

Durch Klicken auf **WLAN(2)** → **Information(0)** → **Information (1)**, dann Klicken auf die Liste der Datenpunkte rechts kann man auf die folgenden 3 Werte zugreifen:

- Ausgewähltes WLAN (WLAN Netzwerkname, mit dem CAN-Gateway verbunden ist/werden soll)
- Verbindungsstatus
- Signalqualität



23 REST-API

REST-API (application Interface) stellt die Möglichkeit dar, auf die konfigurierten Parameter über HTTP Protokoll zuzugreifen und noch vieles andere zu erledigen. WEB-interface des CAN-Gateways nutzt selbst auch diesen REST-API um die Parameterwerte anzuzeigen, die Änderungen einzusteuern, Konfiguration an CAN-Gateway zu übertragen usw. In folgender Tabelle ist der REST-API beschrieben (nicht vollständig, nur Auswahl).

URL	GET oder POST Method	Beschreibung
http://cangateway.local/getparam?num=XX	GET	Antwort erhält den aktuellen Parameterwert für Parameter Nummer XX (XX=1 bis 40)
http://cangateway.local/setparam?num=XX	POST	Änderungsanfrage für Parameter Nummer XX, der Wert muss in Body als text/plain übertragen werden
http://cangateway.local/getparamraw?num=XX	GET	Antwort erhält den aktuellen Parameterwert für Parameter Nummer XX (XX=1 bis 40) in Rohformat, also so wie es auf dem CAN-Bus übertragen wird ohne Umrechnungen
http://cangateway.local/getparamage?num=XX	GET	Antwort erhält das Alter in Sekunden für den aktuellen Parameterwert, für Parameter Nummer XX.
http://cangateway.local/getmac	GET	Antwort erhält die MAC Adresse des CAN-Gateways
http://cangateway.local/getip	GET	Antwort erhält die IP Adresse des CAN-Gateways
http://cangateway.local/getmqttstate	GET	Antwort erhält Status der Verbindung zum MQTT Server
http://cangateway.local/json/getallvalues	GET	Antwort enthält die Werte für alle Parameter in JSON Format.

Anstatt von cangateway.local müssen Sie die IP Adresse von Ihrem CAN-Gateway eintragen.

24 Besondere Hinweise zu Parameteränderungen

Parameteränderungen müssen mit besonderer Vorsicht vorgenommen/eingesteuert werden. CAN-Gateway erlaubt viele Parameter zu ändern, deren Änderung ggf. überhaupt nicht sinnvoll ist. Wenn CAN-Gateway mit Ihrem Hausautomatisierungssystem verbunden ist und die Änderungsanträgen vom Hausautomatisierungssystem entgegen genommen werden können (also einige oder alle Parameter als änderbar konfiguriert sind), müssen Sie sicherstellen dass auch Ihres Hausautomatisierungssystem nur sinnvolle Änderungen einsteuern kann.

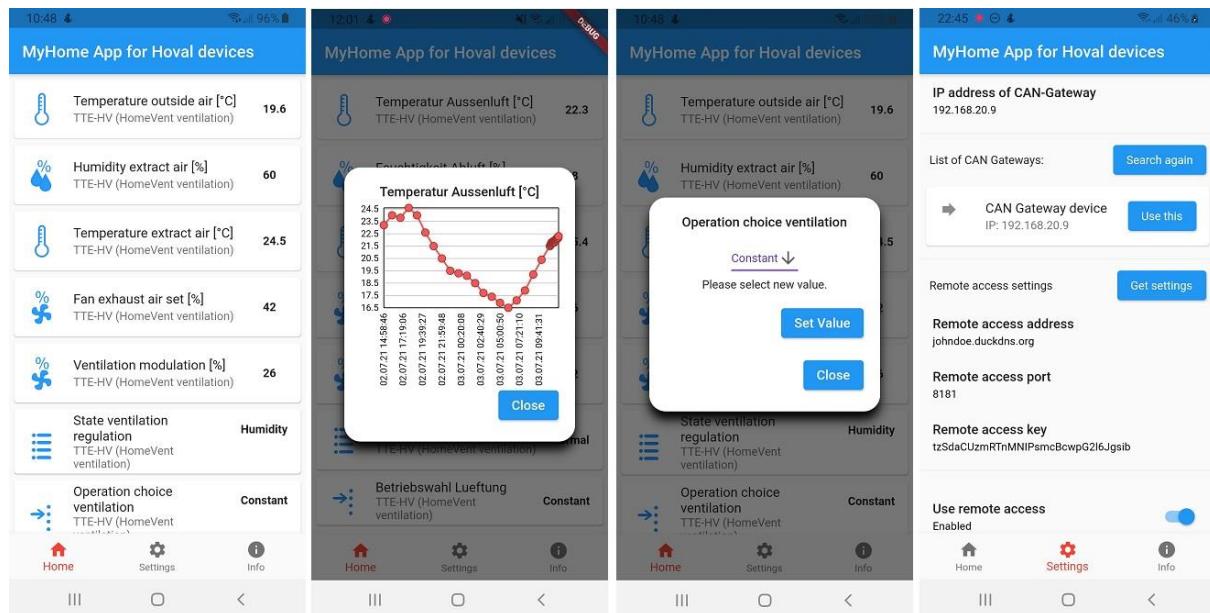
25 Android App: MyHome App für Hoval Geräte

App für Android Smartphone: kann direkt mit CAN-Gateway verwendet werden, solange Smartphone und CAN-Gateway im gleichen WLAN Netzwerk sind. Verfügbar auf Google Play:

https://play.google.com/store/apps/details?id=com.wladwnt.myhome_for_hoval_devices

CAN-Gateway mit Software Version \geq 27.001 notwendig.

Ab App Version 1.1 und ab CAN-Gateway Version 28.001 ist auch Zugriff über Internet möglich. Siehe dazu Kapitel 27.



26 Windows App: MyHome App für Hoval Geräte

Windows App ist die gleiche wie Android App. Sie läuft unter Windows 64-bit. 32-Bit Windows wird leider nicht unterstützt. Die Android/Windows App ist in Flutter programmiert. Die Unterstützung von Windows in Flutter ist noch in Beta Stadium, daher könnte es theoretisch sein, dass Windows App noch einige Bugs hat. Ich konnte aber bisher keine Probleme feststellen.

27 Fernzugriff auf CAN-Gateway über Internet (ab SW Version \geq 28.100)

Ab SW Version 28.100 ist es möglich auf CAN-Gateway über Internet zuzugreifen. Dabei werden die Daten in beide Richtungen verschlüsselt übertragen. Die Datenübertragung findet über HTTP Protokoll statt. Wegen der Verschlüsselung funktioniert der Zugriff über Internet mit einem normalen WEB-Browser nicht, es muss zwingend die Android oder Windows App verwendet werden. Im Lokalen WLAN/LAN Netz ist dann weiterhin Zugriff über WEB-Browser möglich.

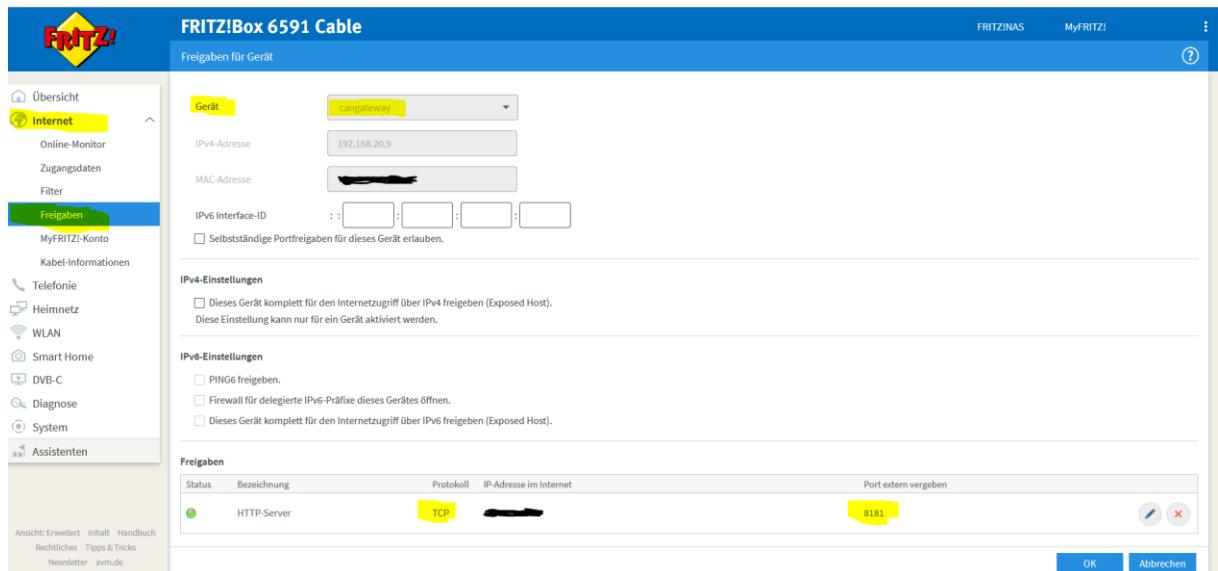
Für einen Zugriff über Internet muss aktuell ihr lokales Netz eine öffentliche IPv4 Adresse (statische oder dynamische) besitzen. Normalerweise das ist der Fall. Allerdings gibt es mittlerweile einige Internet Provider, die standartmäßig nur ein sogenanntes Dual-Stack **Lite** anbieten (z.B. Vodafone/Kabel Deutschland, Unitymedia). In diesem Fall hat man nur eigene IPv6 Adresse und keine von außen erreichbare eigene IPv4 Adresse und man kann nicht ohne weiteres eine Verbindung vom

Android/Windows App zu CAN-Gateway aufbauen. Erfahrungsgemäß kann man aber um die Aktivierung des Dual Stacks (**normal**, nicht **Lite**) bitten und bekommt es auch kostenlos.

27.1 Einrichtung des Zugangs

Ein gesicherter Zugriff auf CAN-Gateway kann wie folgt konfiguriert werden:

- 1) Um über Internet auf CAN-Gateway zu zugreifen, muss CAN-Gateway eine Webadresse besitzen. Das kann man über ein sogenanntes DDNS (Dynamisches DNS) Dienst einrichten. Es gibt verschiedenen kostenlosen DDNS Provider (z.B. duckdns.org, dyndns.org, dynu.com etc.). Sie müssen sich bei einem der Provider registrieren (wenn Sie es nicht bereits für andere Zwecke gemacht haben). Dabei wählt man eine verfügbare Webadresse (wie z.B. maxmustermann.duckdns.org oder maxmustermann1.dyndns.org). Man bekommt auch einen Benutzernamen oder Token und meistens auch noch ein Kennwort. Diese drei wird man im übernächsten Schritt brauchen.
- 2) CAN-Gateway muss über ein TCP Port (default: 8181, kann aber im nächsten Schritt auch ein anderer gewählt werden) vom Internet aus erreichbar sein. Normalerweise ist man über ein DSL/Kabel/WiFi Router ans Internet angebunden. Dieser verbietet standartmäßig externe Zugriffe vom Internet aus auf andere Geräte im lokalen WLAN/LAN Netzwerk. Sie müssen in Ihrem Router eine Portfreigabe für Port Nummer 8181 für CAN-Gateway einrichten. Bezuglich Portfreigabe lesen Sie bitte dazu die Benutzereinleitung zu Ihrem Router. Bei einem Fritzbox geht es z.B. über Menü „Internet → Freigaben“ und dann unter „Port Freigaben“.



- 3) Über WEB-Interface, den Punkt „Fernzugriff über Internet“ wählen.
Folgende Parameter müssen hier eingestellt werden:
Fernzugriff einschalten: muss logischerweise angehakt werden
Zugriffsadresse: die im ersten Schritt gewählte DDNS Webadresse (komplett, z.B. maxmustermann.duckdns.org)
Server Port: lassen Sie hier 8181 stehen. Die fortgeschrittenen Nutzer können auch eine andere Nummer wählen.
Schlüssel: generieren Sie durch klicken auf „Schlüssel generieren“ einen Schlüssel für den Zugang. Dieser Schlüssel wird im nächsten Schritt automatisch in Ihre CAN-Gateway Android/Windows App übertragen, Sie müssen ihn also nicht unbedingt merken. Halten Sie diesen Schlüssel streng geheim! Der Schlüssel ist dazu da, damit nur Sie bzw. Ihre App auf

CAN-Gateway zugreifen kann. Mit diesem Schlüssel werden auch alle übertragenen Daten verschlüsselt.

DDNS durch CAN-Gateway steuern: wenn sie DDNS nicht bereits z.B. durch Router nutzen, muss diese Option angehakt werden.

IP Adresse API Provider: hier einfach eine der verfügbaren Optionen wählen. CAN-Gateway verwendet einen IP API Provider (dafür ist keine Registrierung oder ähnliches notwendig) um eigene von außen zugewiesene IP Adresse zu ermitteln. Diese wird dann an DDNS Provider mitgeteilt.

API fuer ext. IP Adresse: wird bei einem Auswahl des IP API Providers passend automatisch eingestellt (außer man hat „IP API URL manuell einstellen“ ausgewählt, dann hat man hier die Möglichkeit manuell die IP API URL einzustellen, was für fortgeschrittene Nutzer vorgesehen ist)

DDNS Dienst: wählen Sie hier ein DDNS Dienst, bei dem Sie sich im Schritt 1 registriert haben und geben sie im folgenden **Domain, Benutzername/Token** und ggf. **Passwort** ein. Domain ist dabei die von Ihnen im ersten Schritt gewählte Adresse (**Zugriffsadresse**), aber ohne Provider Anteil. Also z.B. „maxmustermann“ ohne „.duckdns.org“.

DDNS Update URL: wird passend zu DDNS Einstellungen automatisch eingestellt (außer man hat „Update URL manuell einstellen“ ausgewählt, was für fortgeschrittene Nutzer vorgesehen ist).

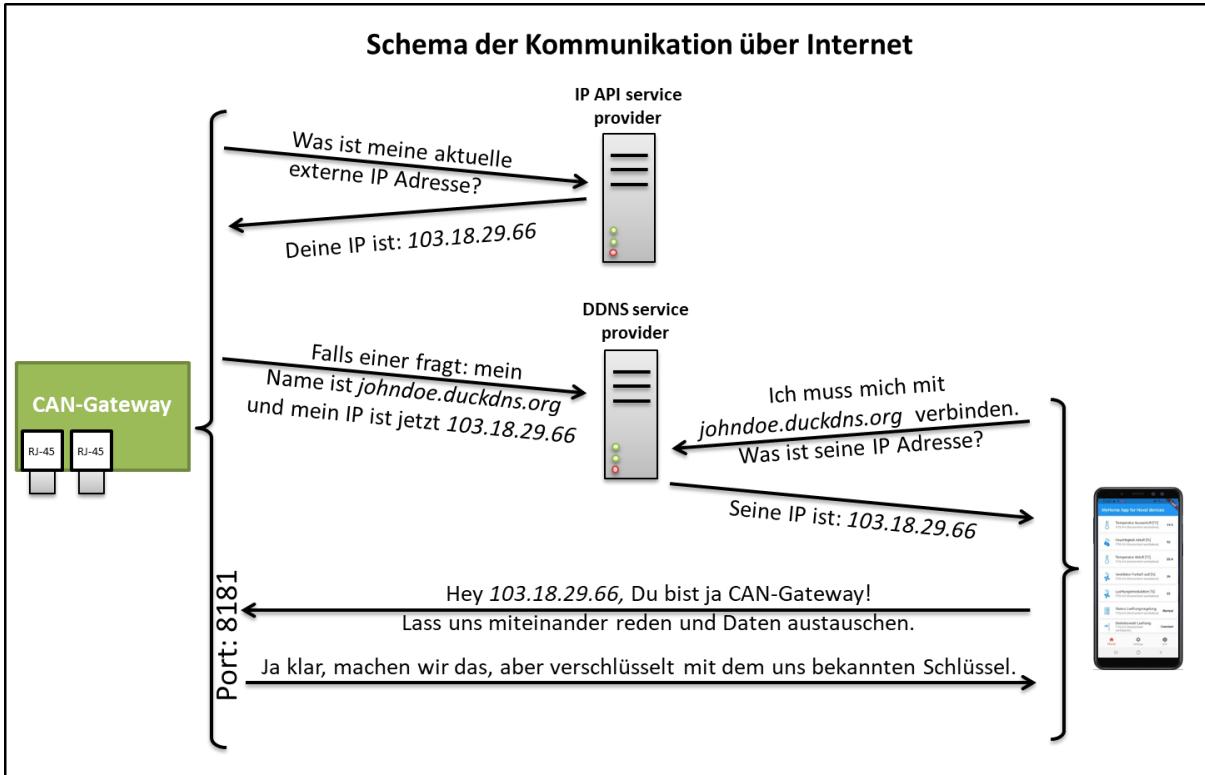
Sobald alles eingestellt ist, müssen Sie auf „Einstellungen Speichern (auf CAN-Gateway hochladen)“ klicken.

Anmerkung: CAN-Gateway speichert intern nur die resultierenden URLs, daher wird nach dem Speichern der Einstellungen oder beim nächsten Mal, wenn man die Einstellungen für Fernzugriff aus dem CAN-Gateway ausließt, die IP API und DDNS Dienst Auswahl auf „Manuell“ gewählt. Das muss man dann nicht erneut ändern, die resultierenden URLs sind dann schon korrekt eingestellt.

The screenshot shows a web browser window titled "CAN CAN Gateway: Fernzugriff über Internet / Einstellungen". The address bar displays "192.168.20.9/remote.html". The page contains several green buttons and input fields:

- "Aktuelle Einstellungen vom CAN Gateway runterladen und anzeigen"
- "Einstellungen speichern (auf CAN Gateway hochladen)"
- "Einstellungen zurücksetzen"
- "Fernzugriff einschalten:
- "Zugriffsadresse: maxmustermann.duckdns.org"
- "Server Port: 8181" (with a note: "Sie müssen in Ihrem DSL/Kabel/WLAN Router die Portfreigabe aktivieren!")
- "Schlüssel: bgQOGvncv171lh99Nnn9EkGSxheQwGll" with a "Schlüssel generieren" button
- "DDNS durch CAN-Gateway steuern: (Alternativ können Sie die entsprechende Option in Ihrem DSL/Kabel/WiFi Router nutzen!)"
- "IP Adresse API Provider: ipify.org (IPv4)"
- "API fuer ext. IP Adresse: http://api.ipify.org/
- "DDNS Dienst: duckdns.org"
- "Domain: maxmustermann"
- "Benutzername/token: 124314-2345235-53535-12444"
- "Passwort: [redacted]" (empty field)
- "DDNS Update URL: http://www.duckdns.org/update?domains=maxmustermann&token=124314-2345235-53535-12444&ip=%s"

- 4) Nun können Sie Ihre CAN-Gateway Android/Windows App auf dem Smartphone/PC/Laptop/Tablet öffnen. Um den Zugriff über Internet zu nutzen, muss die App zunächst für den Zugriff über WLAN/LAN eingerichtet werden. Dafür müssen Sie lediglich die interne IP Adresse des CAN-Gateways in „Einstellungen“ der App konfigurieren. Wenn die App dann im lokalen Netz läuft, kann man im Bereich „Fernzugriff Einstellungen“ auf „Automatisch“ klicken. Die App liest dann über Ihren lokalen WLAN/LAN Netzwerk die erforderlichen Konfigurationen aus dem CAN-Gateway aus (Webadresse, Port und Schlüssel). Nun können Sie weiter unten in Einstellungen den Schalter „Fernzugriff verwenden“ umlegen. Danach können sie mit Ihrem Smartphone/PC/Laptop/Tablet und der App CAN-Gateway auch über Internet steuern und müssen dafür nicht in Ihrem WLAN/LAN Netzwerk sein.



27.2 Details zur Verschlüsselung

Für die Datenübertragung wird AES-256 Verschlüsselung in GCM Modus verwendet. Diese Verschlüsselung wird aktuell als sehr sicher eingestuft und beinhaltet auch automatisch die Authentifizierung. Zusätzlich sind auf CAN-Gateway einige anderen Maßnahmen zur Vermeidung der Attacken implementiert.

Bitte unbedingt den Schlüssel geheim halten! Beim Verdacht auf Missbrauch bitte sofort einen neuen Schlüssel generieren!

28 Hilfe!

28.1 Netzwerkname/Schlüssel sind falsch eingestellt, Web-Interface nicht zugänglich. Was tun?

Wenn im CAN-Gateway falsche Netzwerkname und/oder Netzwerkschlüssel konfiguriert sind, kann CAN-Gateway mit Ihrem WLAN Netzwerk sich nicht verbinden. Dann ist auch der WEB-Interface von CAN-Gateway nicht erreichbar ist. Für diesen Fall gibt es drei Optionen:

- 1) Die Möglichkeit alle Einstellungen auf die Initiale Werte (inklusiv CAN-Gateway als Access Point mit Netzwerkname cangateway und Netzwerkschlüssel 000999555) über Bedienmodul zurückzusetzen. Diese Option ist allerdings nur verfügbar, wenn in den Einstellungen zuvor die Steuerung des CAN-Gateways über Bedienmodul aktiviert wurde. Siehe dazu Kapitel „Steuerung über Bedienmodul“.
- 2) **Nur wenn Software auf einem CAN-Gateway development Board verwendet wird:** drücken und halten Sie gedrückt „REINIT“ Taster für mindestens 6 Sekunden. CAN-Gateway setzt seine Einstellungen auf den initialen Zustand zurück und restartet als Access Point mit dem Namen „cangateway“.

- 3) die Möglichkeit, entweder komplett alle Einstellungen oder nur Netzwerkname und Netzwerkschlüssel auf die Initiale Werte (cangateway, 000999555) über USB Schnittstelle zurückzusetzen. Dafür müssen Sie CAN-Gateway über USB Schnittstelle mit Ihrem Laptop/PC verbinden (siehe dazu Schritte 1 bis 6 in Kapitel „Software flashen“). Danach müssen Sie ein Terminalprogramm für serielle Schnittstelle aus dem Internet herunterladen, installieren und starten, die Verbindung zum CAN-Gateway herstellen und an CAN-Gateway die Zeichenfolge „RESET_TO_INITIAL_SETTINGS“ (für vollständiges Zurücksetzen aller Parameter) oder „RESET_WLAN_SETTINGS“ (um nur Netzwerkname/Netzwerkschlüssel zurück zu setzen) senden. Es muss auf Großbuchstaben geachtet werden und die Zeichenfolge muss am Ende zusätzlich ein „Carriage Return“ (CR, ASCII Code = 13 (hexadezimal 0D)) Zeichen beinhalten. CAN-Gateway antwortet dann mit „OK! Restart...“ und startet neu. Dabei startet WLAN wie im initialen Zustand als Access Point (siehe Kapitel „Erste Inbetriebnahme“).

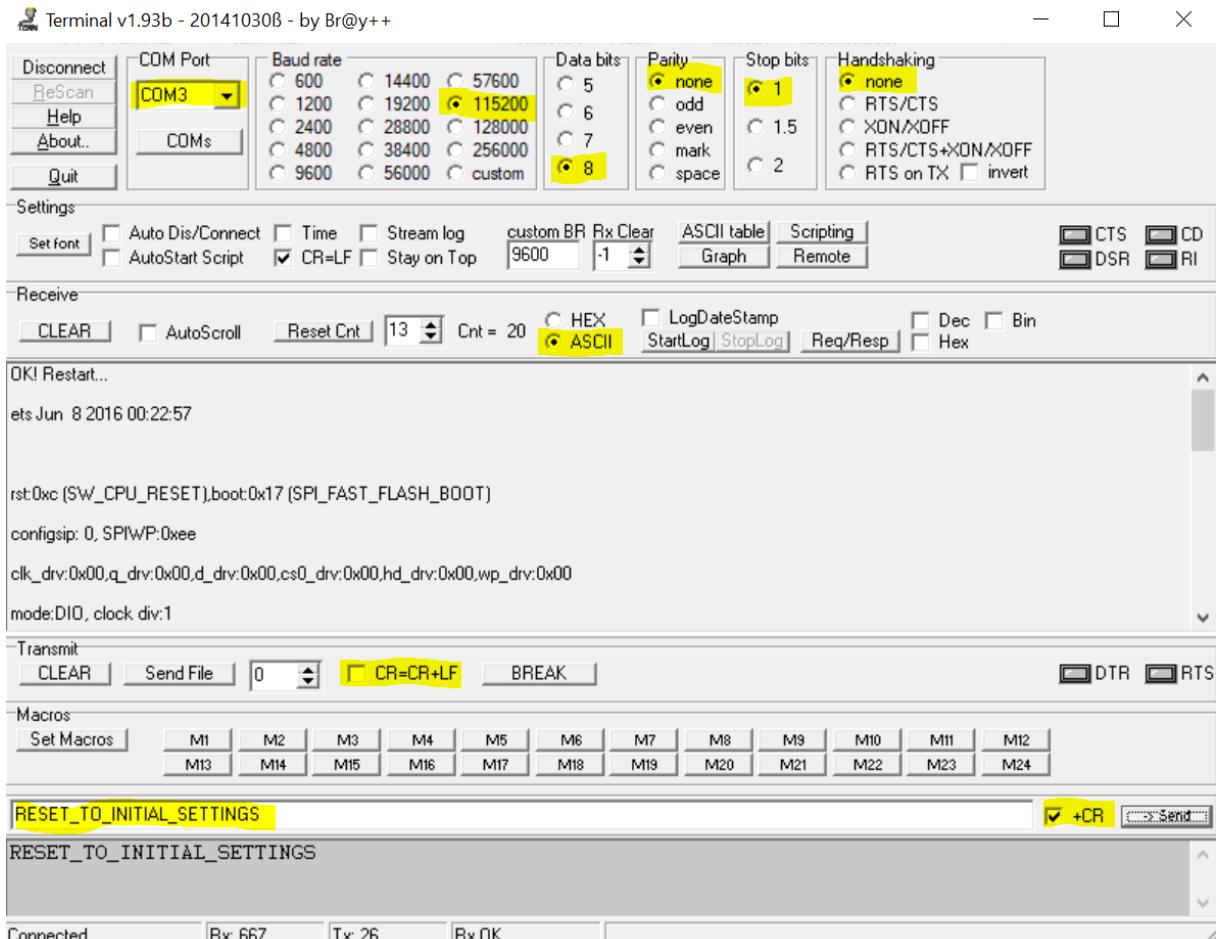
Wichtig: wenn man sich im Terminalprogramm mit dem CAN-Gateway verbindet, muss die richtige Baudrate vorgegeben werden. Initial sind es 115200 Baud. Wenn Sie aber diese Baud-Rate bereits geändert haben, müssen sie die richtige einstellen. Wenn Sie vergessen haben, welche Baudrate die richtige ist, kann man einfach versuchen alle zulässigen Baudaten durchzuprobieren. Dabei muss man nach jedem Versuch cangateway aus- und einschalten.

[Neu ab SW 26.001] Wenn CAN-Gateway mit aktuell eingestellten WLAN Netzwerkname und Netzwerkschlüssel noch nie mit WLAN verbunden war, versucht er lediglich ca. 45 Sekunden sich mit dem Netzwerk zu verbinden. Klappt es nicht, schaltet er automatisch in AP Mode (Name: cangateway, Schlüssel: 000999555). Also wenn man die WLAN Zugangsdaten einfach falsch eingegeben hat, reicht es einfach ca. eine Minute zu warten bis CAN-Gateway in AP Mode geht. Wenn CAN-Gateway aber die richtigen WLAN Zugangsdaten hatte und man jetzt einfach ein anderes Netzwerk hat, wird er weiterhin unendlich lange versuchen sich mit dem Netzwerk zu verbinden. Also dann muss man eine der beiden oben beschriebenen Optionen nutzen.

Im Folgenden ist die Vorgehensweise mit zwei Terminal Programme beschrieben:

28.1.1 Mit Bray Terminal

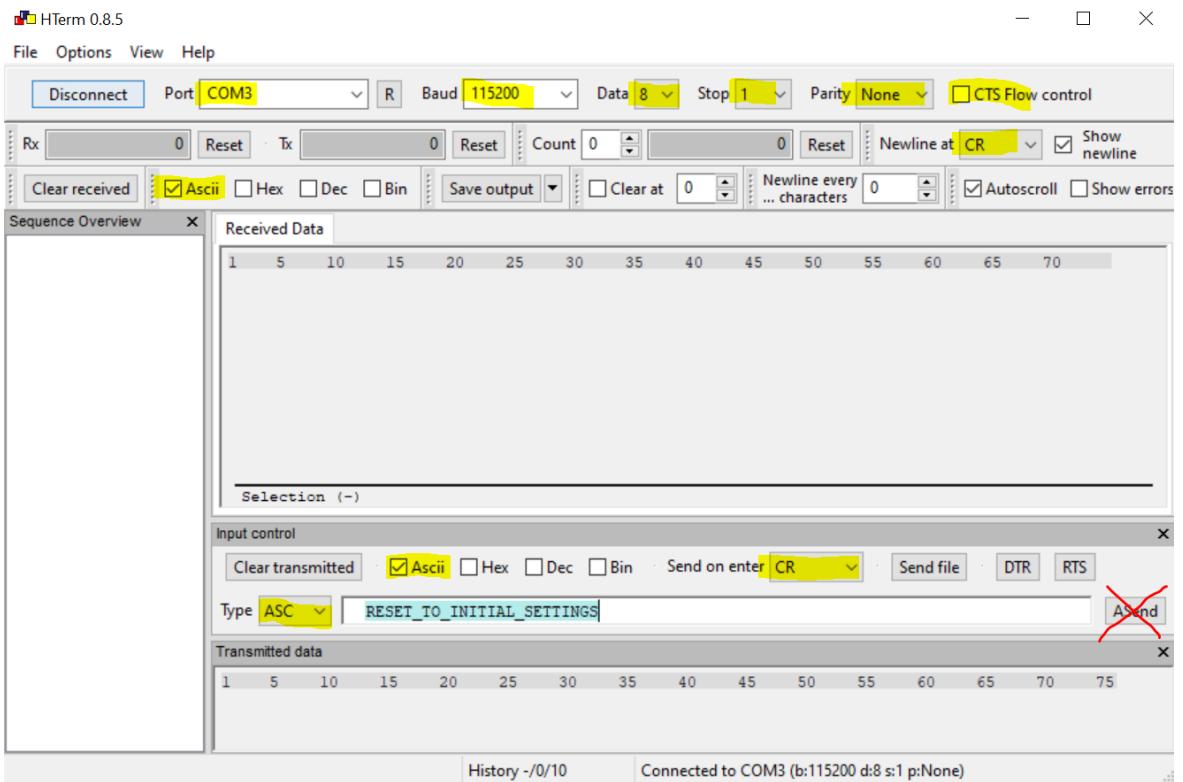
- 1) Das Programm hier downloaden, ggf. entpacken und starten:
<https://sites.google.com/site/terminalbpp/>
- 2) Richtigen COM-Port und Baudrate wählen. Die Anderen Einstellungen wie auf dem Bild unten.
- 3) Auf „Connect“ klicken.



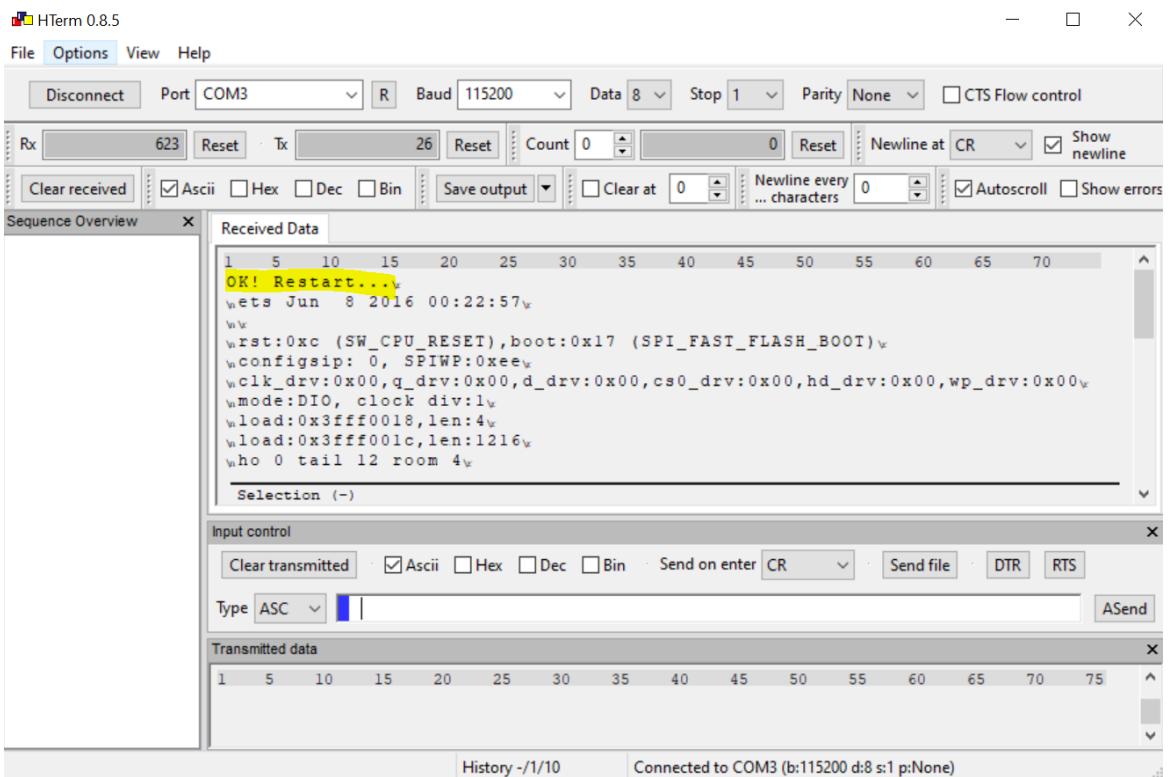
- 4) „RESET_TO_INITIAL_SETTINGS“ eintippen und dann auf „->Send“ klicken.
- 5) Cangateway antwortet mit „OK! Restart“ wie im Bild.
- 6) Dann auf „Disconnect“ klicken und das Programm kann beendet werden.

28.1.2 Mit HTerm

- 1) Das Programm hier downloaden, ggf. entpacken und starten:
<http://www.der-hammer.info/pages/terminal.html>
- 2) Richtigen COM-Port und Baudrate wählen. Die Anderen Einstellungen wie auf dem Bild unten.
- 3) Auf „Connect“ klicken.



- 4) „RESET_TO_INITIAL_SETTINGS“ eintippen und dann auf Enter (Tastatur) klicken. NICHT „ASend“ benutzen.
- 5) Cangateway antwortet mit „OK! Restart“ wie im Bild (ggf. etwas nach oben scrollen um das zu sehen):



- 6) Dann auf „Disconnect“ klicken und das Programm kann beendet werden.

28.2 CAN-Gateway startet immer mit Werkseinstellungen. Einstellungen können nicht gespeichert werden.

Die wahrscheinlichste Ursache ist, dass die Hardware (Flash Speicher) beschädigt ist. CAN-Gateway Software bildet über alle Einstellungen eine Prüfsumme, diese wird zusammen mit den Einstellungen in Flash Speicher des ESP32 gespeichert. Beim Aufstarten werden die Einstellungen und die Prüfsumme aus dem Flash Speicher ausgelesen. Stimmt die Prüfsumme nicht, werden die Einstellungen automatisch auf die Werkseinstellungen zurückgesetzt, weil die Korrektheit/Integrität der ausgelesenen Einstellungen dann nicht gewährleistet werden kann. Sie müssen dann wohl ESP32 DevKit austauschen.

29 Hinweise zu HomeVent Komfortlüftung mit dem einfachen Bediengerät BG02E

HomeVent Geräte der Firma Hoval vom TTE Typ (Typ z.B. FR201, FR251, FR351, FRT351 etc.) werden in einfacher Variante mit einem Bediengerät BG02E (kein Touch, zwei Einstellräder für Volumenstrom und Feuchte sowie ein Taster für Party-Modus) betrieben. In diesem Fall muss man Folgendes beachten, wenn man CAN-Gateway einsetzen will. BG02E Bediengerät ist recht „dumm“, er sendet einfach ca. jede Sekunde oder so die Sollwerte für Volumenstrom und Feuchte an die Komfortlüftung. Wenn man in diesem Verbund CAN-Gateway anschließt, dann kann man über CAN-Gateway nichts ansteuern. Genauer gesagt, wenn CAN-Gateway einen Sollwert für Volumenstrom oder Feuchte an Komfortlüftung sendet, wird er in nächster Sekunde durch BG02E Bediengerät wieder geändert bzw. „überschrieben“. Also praktisch in diesem Verbund kann man nur CAN-Gateway zum Auslesen der Parameter nutzen, nicht aber zum Steuern. Will man auch steuern können, muss man dann die Komfortlüftung nur mit dem CAN-Gateway betreiben und BG02E Bediengerät abstecken. Schließlich kann man ja auch über WEB-Interface vom CAN-Gateway die Sollwerte einstellen und ändern (oder halt über MQTT Protokoll etc.). Will man auch einen Bediengerät von Hoval haben, dann geht es nur mit dem deutlich teureren Bedienmodul von Hoval, dem mit dem Touchscreen. Dieser sendet nämlich (wie auch CAN-Gateway) die Sollwerte nur einmal bei deren Änderung, kann also in einem Verbind mit CAN-Gateway gut funktionieren.

Die andere Implikation: BG02E Bediengerät wird vom CAN-Gateway als TTE-WEZ Gerät vom Typ=0 und Adresse=0 erkannt und auf der „WEB-Interface: Liste der Geräte“ Seite angezeigt. Das hat an sich aber keinerlei negative Auswirkung auf die Funktionalität des CAN-Gateways.

[**Neu ab SW Version 27.100 und HW Version 5**] CAN-Gateway ab Hardware V5 besitzt (OPTIONAL, sofern bestückt) eine zusätzliche CAN-Schnittstelle (implementiert als RJ45 Buchse), die speziell für Anschluss eines BG02E Bedienmoduls bestimmt ist. Diese zusätzliche Schnittstelle ermöglicht die oben genannten Nachteile zu umgehen und CAN-Gateway zusammen mit einem BG02E Bedienmodul vollumfänglich zu nutzen. Es funktioniert so, dass CAN-Gateway die Vorgaben des Bedienmoduls bzgl. Volumenstrom und Feuchte-Sollwert an Lüftungsgerät nur dann weiterleitet, wenn man die Stellräder gerade gedreht oder Party-Taster gedrückt hat. So können die Sollwerte auch über CAN-Gateway geändert werden und gelten so lange, bis man sie über BG02E Betätigung „überschreibt“. Status-LED am BG02E zeigt - mindestens 20 Sekunden lang nachdem man Stellräder oder Party-Taste betätigt hat – den aktuellen Status des Lüftungsgeräts gemäß der originalen Herstellerbeschreibung. Ansonsten leuchtet es immer grün, wenn BG02E korrekt mit dem CAN-Gateway verbunden ist.

30 Zeiteinstellung in CAN-Gateway [neu ab SW 22.001]

Um z.B. den richtigen Zeitstempel für geloggte Daten zu bekommen, fragt CAN-Gateway über Internet bei einem Zeitserver (siehe Einstellungen) nach aktueller Zeit. Die Abfrage wird einmal pro Stunde durchgeführt. Dazwischen rechnet CAN-Gateway die Zeit selbst.

Wenn der Zeitserver gar nicht erreichbar ist, verwendet CAN-Gateway eine fiktive interne Zeit, die bei jedem Power-Up mit 1.1.1970, 00:00:00 startet. Das ist z.B. der Fall, wenn Zeitserver falsch eingestellt ist oder wenn man CAN-Gateway ausschließlich im lokalen Netzwerk ohne Internet Zugang betreibt.

Wenn CAN-Gateway mindestens einmal vom Zeitserver die Zeit bekommen hat, versucht er bis auf weiteres die Zeit selber zu rechnen. Das läuft auch über ein Reset hinaus, also solange die Stromversorgung da ist. Da aber die interne Uhr nicht besonders genau ist, kann es sein, dass diese Zeit nach vielen Stunden von der realen Zeit abweicht, wenn die Verbindung zum Zeitserver nicht mehr da ist. Sobald die Verbindung wieder möglich, wird CAN-Gateway die interne Zeit korrigieren.

Bei der Darstellung der historischen (gelogten) Daten über WEB-Interface, versucht WEB-Interface die Zeiten korrekt zum aktuellen Zeitpunkt umzurechnen wenn er feststellt, dass diese mit fiktiver interner Zeit (also Jahr 1970 etc.) gespeichert sind.

31 Autorecovery

Wenn CAN-Gateway ein schwerwiegendes Problem feststellt, wird automatisch ein Reset durchgeführt. Dadurch wird versucht sich vom Problem zu befreien. Ein schwerwiegendes Problem kann z.B. sein, dass WLAN Verbindung so grenzwertig wird, dass man sie ständig verliert und wieder verbindet. Oder wenn z.B. in der Kommunikation mit dem MQTT Server Probleme auftreten. In vielen Fällen läuft CAN-Gateway monatelang ohne Probleme. Erfahrungsgemäß kann jedoch Reset ein Mal pro Woche passieren. Das ist immer noch normal und führt nicht zu Funktionalitätseinschränkungen. Wenn es deutlich öfter auftritt, also z.B. mehrmals pro Tag, sollten Sie schauen ob WLAN ausreichend gut ist und ggf. logging Dateien von Ihrem MQTT Server prüfen.

32 Einbindung in Home Assistant

32.1 Einleitung

Home Assistant hat sehr viele Möglichkeiten, wie man externe Geräte einbinden kann. CAN-Gateway stellt auch verschiedene Interfaces zur Verfügung (MQTT, REST-API, ...), die verwendet werden können.

Im Folgenden wird nur eine der möglichen Optionen erläutert. Dabei wird davon ausgegangen, dass

- 1) Home Assistant korrekt eingerichtet ist (Version 2023.06 oder neuer).
- 2) Ein MQTT Broker ist installiert und in Home Assistant integriert, z.B. als Add-on:
<https://github.com/home-assistant/addons/blob/master/mosquitto/DOCS.md>
- 3) CAN-Gateway ist für die Kommunikation mit dem MQTT Broker eingerichtet (siehe Bedienungseinleitung/MQTT Einstellungen) und Zeigt unter „Status“ in CAN-Gateway Web Interface „mit MQTT verbunden“.

32.2 Konfiguration

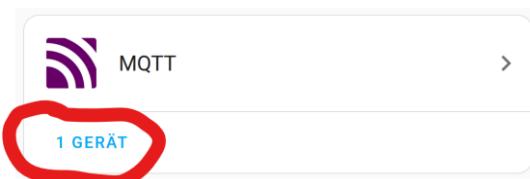
32.2.1 Automatische Konfiguration mit MQTT Discovery

CAN Gateway unterstützt eine automatische Erkennung aller Parameter in Home Assistant via MQTT. Bis CAN-Gateway Version 29.326 werden alle Parameter als Sensoren erkannt. Ab Version 29.327 ist eine deutlich verbesserte MQTT Discovery Unterstützung implementiert: Parameter werden entweder als Sensoren, als einstellbare numerische Werte, als binäre Sensoren, als Schalter oder als Auswahllisten erkannt und automatisch konfiguriert: je nachdem, wie der Parameter in CAN-Gateway konfiguriert ist.

Die Zuordnung erfolgt wie folgt:

- Als einstellbare numerische Werte werden Parameter konfiguriert, welche in CAN-Gateway als ein numerischer Typ eingestellt sind (U8, S8, U16, S16, U32, S32) und für Lesen und Schreiben (Attribut „w“) konfiguriert sind. Das betrifft z.B. die einstellbaren Solltemperaturen, Soll-Luftungsmodulation oder Soll-Feuchtigkeit beim HomeVent etc.
- Als Binäre Sensoren werden Parameter konfiguriert, welche in „Parameter-Konfiguration Generator“ beinhaltet sind, nur Werte „0“ (Aus) oder „1“ (Ein) einnehmen können und in CAN-Gateway Einstellungen nur für Lesen (Attribut „r“) konfiguriert sind.
- Als Binäre Sensoren werden Parameter konfiguriert, welche in „Parameter-Konfiguration Generator“ beinhaltet sind, nur Werte „0“ (Aus) oder „1“ (Ein) einnehmen können und in CAN-Gateway Einstellungen für Lesen und Schreiben (Attribut „w“) konfiguriert sind. Ein Beispiel ist die Aktivierung der CoolVent Funktion beim HomeVent.
- Als Auswahllisten werden Parameter konfiguriert, welche in „Parameter-Konfiguration Generator“ beinhaltet sind, Werte zwischen 0 und 255 einnehmen können wobei jedem Wert ein bestimmter Zustand zugeordnet ist und in CAN-Gateway Einstellungen für Lesen und Schreiben (Attribut „w“) konfiguriert sind. Beispiele sind die Parameter „Betriebswahl Lüftung“ oder „Betriebswahl Heizung“.
- Wenn nichts oben zutrifft, wird ein Parameter als ein Sensor konfiguriert.

Wenn die Option „Unterstützung [Home Assistant / MQTT Discovery](#)“ aktiviert ist, CAN-Gateway mit dem MQTT Broker verbunden ist und Home Assistant die MQTT Integration eingerichtet hat, dann kann man in Home Assistant unter „Einstellungen“-> „Geräte & Dienste“ die MQTT Integration finden. Dabei muss stehen, dass mindestens ein Gerät und mehrere Entitäten verfügbar sind. Darunter sind CAN-Gateway (Gerät mit dem Namen „Hoval-CANGW“) und die Parameter als Entitäten. Nun kann man die Steuerelemente und Sensoren zum Dashboard hinzufügen.



← Hoval-CANGW

Gerät Informationen

von <https://shop.myhome-control.de>
Firmware: v29_330_HW7

MQTT

BESUCHEN [MQTT](#)

Automatisierungen [+](#)

Es wurden noch keine Automatisierungen mit diesem Gerät hinzugefügt. Zum Hinzufügen drücke die + Schaltfläche.

Szenen [+](#)

Es wurden noch keine Szenen mit diesem Gerät hinzugefügt. Zum Hinzufügen drücke die + Schaltfläche.

Skripte [+](#)

Es wurden noch keine Skripte mit diesem Gerät hinzugefügt. Zum Hinzufügen drücke die + Schaltfläche.

Steuerelemente

- Betriebswahl Heizung Konst.
- Betriebswahl Lueftung Konst.
- Betriebswahl Waermeerzeuger Automatik

CoolVent aktivieren [\[Switch\]](#)

Feuchte Sollwert [\[Slider\]](#)

Normal Lueftungsmodulation [\[Slider\]](#)

[ZUM DASHBOARD HINZUFÜGEN](#)

Sensoren

Feuchtigkeit Abluft	56 %
Status Heizkreisregelung	Komfort Heiz.
Status Lueftungsregelung	Feucht.
Temperatur Abluft	21,6 °C
Temperatur Aussenluft	12,4 °C
Ventilator Fortluft soll	48 %

+1 versteckte Entität

[ZUM DASHBOARD HINZUFÜGEN](#)

Auf dem Dashboard erscheinen dann die Steuerelemente sowie die Sensoren:

Hoval-CANGW

Betriebswahl Heizung Konst.

Betriebswahl Lueftung Konst.

Betriebswahl Waermeerzeuger Automatik

CoolVent aktivieren [\[Switch\]](#)

Feuchte Sollwert [\[Slider\]](#) 40 %

Normal Lueftungsmodulation [\[Slider\]](#) 45 %

Hoval-CANGW

Feuchtigkeit Abluft	56 %
Status Heizkreisregelung	Komfort Heiz.
Status Lueftungsregelung	Feucht.
Temperatur Abluft	21,6 °C
Temperatur Aussenluft	12,4 °C
Ventilator Fortluft soll	48 %

Diese können dann bearbeitet werden: man kann z.B. einen anderen Symbol zuordnen oder anders benennen.

Der Vorteil der automatisierten Konfiguration ist, dass man in Home Assistant nichts manuell einstellen muss. Sie funktioniert jedoch nur für die meist verwendete Parameter korrekt. Parameter,

die in dem Konfigurator nicht beinhaltet sind und der Nutzer selber eingestellt hat, müssen unter Umständen manuell konfiguriert werden, wie im Folgenden beschrieben ist.

32.2.2 Manuelle Konfiguration

Siehe <https://www.home-assistant.io/integrations/mqtt/#manual-configured-mqtt-items>. Vorteil: man hat volle Kontrolle. Man kann z.B. die Parameter schöner oder anders darstellen lassen, als MQTT Discovery es erlaubt. Dazu hat Home Assistant umfangreiche Optionen. Die Einstellungen müssen in der Datei *configuration.yaml* (siehe <https://www.home-assistant.io/getting-started/configuration/>) gespeichert werden. Im folgenden werden einige

Beispiel 1: (angenommen, man hat in CAN-Gateway ein Parameter wie folgt eingestellt:

param=8;8;50;0;39652;r;Status_Lueftungsregelung;10;U8;1.000000;0.000000;

):

```
mqtt:
  sensor:
    - name: "Luftung Status"
      state_topic: "cangateway/Status_Lueftungsregelung"
      value_template: >-
        {% set mapper = {
          '0' : 'Aus/Standby',
          '1' : 'Normal',
          '2' : 'VOC aktiv',
          '3' : 'Feuchtigkeitsmodus',
          '4' : 'Frostschutz',
          '5' : 'CoolVent',
          '6' : 'Fehler'} %}
        {{ mapper[value] if value in mapper else 'Unbekannt' }}
```

Dieses Beispiel entspricht genau dem, wie auch MQTT Discovery dieser Parameter einstellen würde. Sie können aber durch die manuelle Konfiguration z.B. die Übersetzung in eine andere Sprache machen.

Im nächsten Beispiel wird ein HomeVent in Home Assistant als ein Ventilator dargestellt. Zur Erläuterung wird auf die Dokumentation zu Home Assistant verwiesen.

Beispiel 2: Angenommen man hat im CAN-Gateway folgende Parameter eingerichtet:

param=8;8;50;0;38606;r;Lueftungsmodulation_percent;10;U8;1.000000;0.000000;

param=8;8;50;0;40650;w;Betriebswahl_Lueftung;10;U8;1.000000;0.000000;

param=8;8;50;0;40651;w;Normal_Lueftungsmodulation_percent;10;U8;1.000000;0.000000;

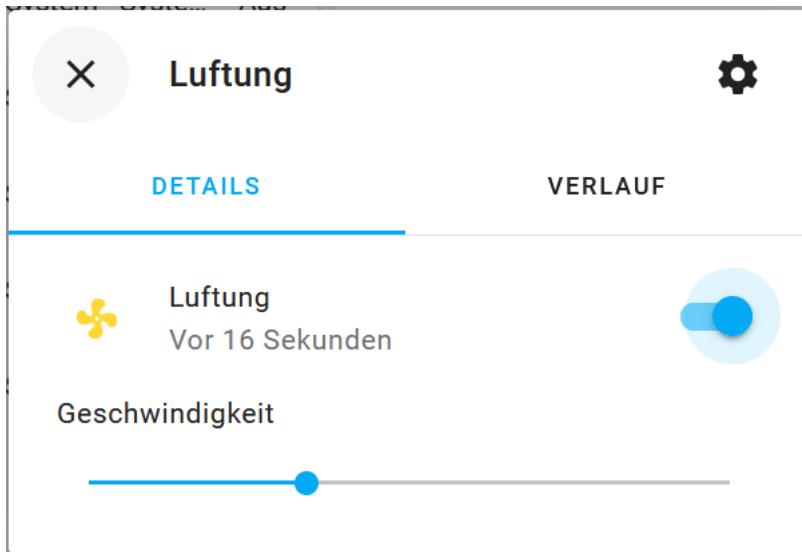
Dann kann die Konfiguration in *configuration.yaml* wie folgt aussehen:

```
mqtt:
  fan:
    - name: "Luftung"
      unique_id: "ID_custom001"
      state_topic: "cangateway/Betriebswahl_Lueftung"
      command_topic: "cangateway/Betriebswahl_Lueftung/set"
      percentage_state_topic: "cangateway/Lueftungsmodulation_percent"
      percentage_command_topic: "cangateway/Normal_Lueftungsmodulation_percent/set"
      payload_on: "4"
      payload_off: "0"
      speed_range_min: 1
      speed_range_max: 100
```

Für die Erklärung der Parameter siehe <https://www.home-assistant.io/integrations/fan.mqtt/>

Bzgl. der Konfigurationsdatei *configuration.yaml* siehe <https://www.home-assistant.io/docs/configuration/>

Damit kann man die Lüftung ein- und ausschalten sowie Luftmenge einstellen (in neueren Home Assistant Versionen sieht es etwas anders aus):



Die gleiche Steuerung ist auch mit automatischer MQTT Discovery möglich, wird jedoch anders dargestellt.

Beispiel 3: Betriebsauswahl für Lüftung. Angenommen man hat im CAN-Gateway folgende Parameter eingerichtet:

param=8;8;50;0;40650;w;Betriebswahl_Lueftung;10;U8;1.000000;0.000000;

Dieser Parameter kann laut Dokumentation (z.B. siehe <http://www.hoval.com/misc/TTE/TTE-GW-Modbus-datapoints.xlsx>) folgende Werte einnehmen: 0 (=“Standby”), 1 (=“Woche 1”), 2 (=“Woche 2”), 4 (=“Konstantbetrieb”), 5 (=“Sparbetrieb”). Man kann es in Home Assistant z.B. als MQTT „Select“ Element implementieren (siehe <https://www.home-assistant.io/integrations/select.mqtt/>): dafür in *configuration.yaml* folgendes einfügen:

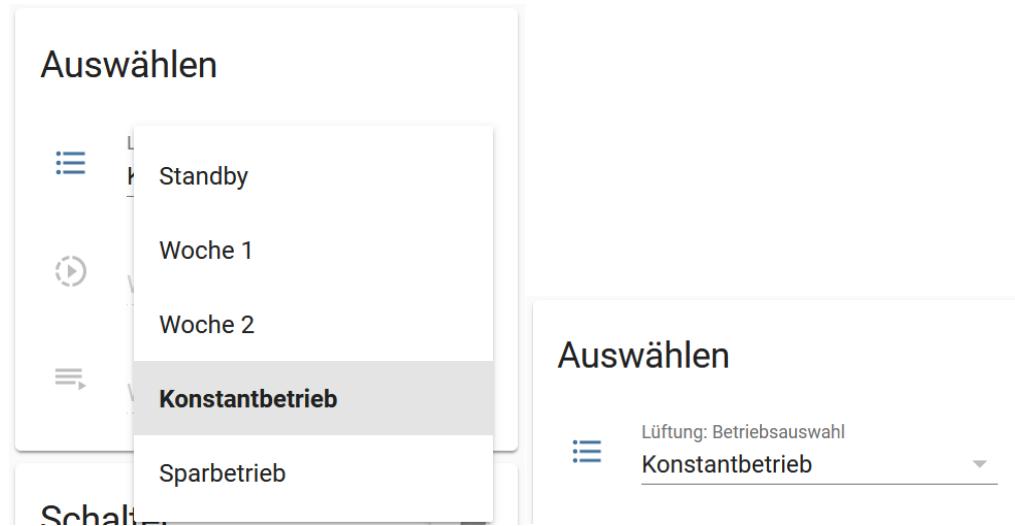
```
mqtt:  
  select:  
    - command_topic: "cangateway/Betriebswahl_Lueftung/set"  
      command_template: >-  
        {% set mapper = {  
          'Standby' : '0',  
          'Woche 1' : '1',  
          'Woche 2' : '2',  
          'Konstantbetrieb' : '4',  
          'Sparbetrieb' : '5' } %}  
        {{ mapper[value] if value in mapper else 'Unbekannt' }}  
      state_topic: "cangateway/Betriebswahl_Lueftung"  
      value_template: >-  
        {% set mapper = {  
          '0' : 'Standby',  
          '1' : 'Woche 1',  
          '2' : 'Woche 2',  
          '4' : 'Konstantbetrieb',  
          '5' : 'Sparbetrieb' } %}
```

```

{{ mapper[value] if value in mapper else 'Unbekannt' }}
name: "Lüftung: Betriebsauswahl"
unique_id: "ID_custom002"
options:
  - "Standby"
  - "Woche 1"
  - "Woche 2"
  - "Konstantbetrieb"
  - "Sparbetrieb"

```

Durch die Verwendung der Templates werden die Werte (0, 1, 2, 4, 5), die eine Betriebsart kodieren, dann als verständlicher Klartext dargestellt und können ausgewählt werden:



Dieses Beispiel entspricht genau dem, wie auch MQTT Discovery dieser Parameter einstellen würde. Sie können aber durch die manuelle Konfiguration z.B. die Übersetzung in eine andere Sprache machen.

Beispiel 4: Lüftungsstatus anzeigen. Angenommen man hat im CAN-Gateway folgende Parameter eingerichtet:

param=8;8;50;0;39652;r;Status_Lueftungsregelung;10;U8;1.000000;0.000000;

Dieser Parameter kann laut Dokumentation (z.B. siehe <http://www.hoval.com/misc/TTE/TTE-GW-Modbus-datapoints.xlsx> oder in CAN-Gateway unter „Liste der Parameterwerte“ bei diesem Parameter auf grünes „i“ klicken) folgende Werte einnehmen: 0 (=“Aus/Standby”), 1 (=“Normal”), 2 (=“VOC aktiv”), 3 (=“Feuchtigkeitsmodus”), 4 (=“Frostschutz”), 5 (=“Coolvent”), 6 (=“Fehler”), 7 (=Sommerfeuchtigkeit”).

Dieser Status kann man in Home Assistant z.B. als MQTT „Sensor“ Element implementieren und so konfigurieren, dass nicht die numerische Werte, sondern der Status in Klartext angezeigt wird: dafür in *configuration.yaml* folgendes einfügen:

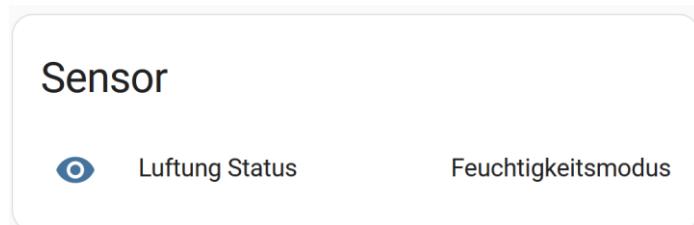
```

mqtt:
  sensor:
    - name: "Luftung Status"
      unique_id: "ID_custom003"
      state_topic: "cangateway/Status_Lueftungsregelung"
      value_template: >-
        {% set mapper = {
          '0' : 'Aus/Standby',

```

```
'1' : 'Normal',
'2' : 'VOC aktiv',
'3' : 'Feuchtigkeitsmodus',
'4' : 'Frostschutz',
'5' : 'CoolVent',
'6' : 'Fehler',
'7' : 'Sommerfeuchte'} %}
{{ mapper[value] if value in mapper else 'Unbekannt' }}
```

Der Status wird dann wie folgt dargestellt:



Auch dieses Beispiel entspricht genau dem, wie auch MQTT Discovery dieser Parameter einstellen würde. Sie können aber auch hier durch die manuelle Konfiguration z.B. die Übersetzung in eine andere Sprache machen.

33 Einbindung in OpenHab

OpenHab unterstützt MQTT Discovery ähnlich wie Home Assistant. Damit ist es möglich mit einigen einfachen Klicks die Einbindung in OpenHab mittels MQTT Protokolls einzurichten. Die notwendigen Schritte sind in folgender Anleitung bildlich zusammengefasst:

https://raw.githubusercontent.com/wladwnt/CAN-Gateway/master/OpenHab_configuration.pdf

Erfahrene Anwender können auch andere Methoden verwenden, wie z.B. Einbindung über HTTP REST API.

34 Einbindung in Loxone

Einbindung in Loxone kann über zwei alternative Wege erfolgen: als virtuelle HTTP Eingänge und Ausgänge oder als Modbus TCP. Für beide Optionen bietet CAN-Gateway ab SW Version 29.330 eine Hilfestellung in Form von einem Template-Generator. Dieser generiert XML Vorlagen für die in CAN-Gateway konfigurierte Parameter. Diese Vorlagen können dann in Loxone importiert werden. So muss man in Loxone dann nur ggf. ganz wenige, optionale Einstellungen (z.B. min/max Werte etc.) manuell nachbearbeiten.

34.1 Einbindung als virtuelle HTTP Eingänge und Ausgänge

Auf CAN-Gateway Seite wird dabei REST API verwenden, die im Kapitel 23 dieser Einleitung beschrieben ist. Es werden zwei REST-API Befehle dabei verwendet:

<http://cangateway.local/json/getallvalues>

<http://cangateway.local/setparam?num=XX>

Auf Loxone Seite werden die virtuellen HTTP Eingänge und virtuellen HTTP Ausgänge verwendet. Die Dokumentation zu diesen Möglichkeiten ist hier zu finden:

- virtuelle HTTP Eingänge: Siehe <https://www.loxone.com/dede/kb/virtueller-http-eingang/>
- virtuelle HTTP Ausgänge: Siehe Kapitel „Anwendung Virtueller Ausgangsverbinder“ (nur Ausgänge, die virtuellen Eingänge ist hier was anderes!) in <https://www.loxone.com/dede/kb/virtuelle-ein-und-ausgaenge/>

Diese können manuell konfiguriert werden oder man kann jeweils eine XML Vorlage für Eingänge und Ausgänge verwenden, die CAN-Gateway generieren kann. Diese wird dann in Loxone importiert.

Um Vorlage zu generieren, geht man im CAN-Gateway Web Interface zu „Liste der Parameterwerte“ und dann unter der Tabelle zu:

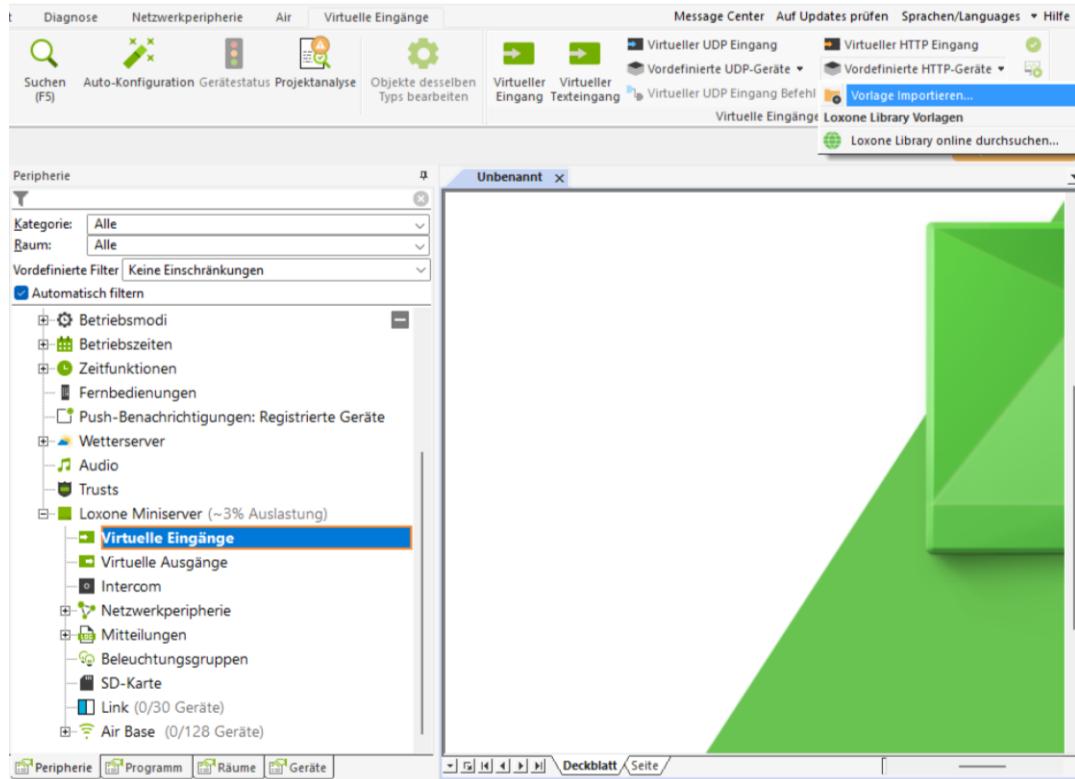
Generieren XML Daten für die Integration in Loxone: **via ModBus TCP**
Alternativ: **via virtuelle HTTP Eingänge** und **via virtuelle HTTP Ausgänge**

Beim Klick auf beide Taster „via virtuelle http Eingänge“ und „via virtuelle http Ausgänge“ werden zwei Dateien generiert: VI_can_gateway_template.xml und VO_can_gateway_template.xml. Wenn keine Parameter fürs Schreiben konfiguriert sind, dann beinhaltet die Datei für Ausgänge (VO_can_gateway_template.xml) nur den Header und ist ansonsten inhaltlich leer.

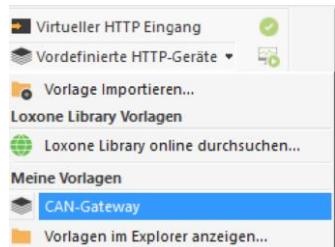
Achtung: wenn Sie die Einstellungen der Parameter in CAN-Gateway ändern, ändern sich auch diese Dateien. Sie müssen dann diese neu generieren und in Loxone importieren.

Nach dem diese in Loxone importiert sind, können/sollen einige optionalen Einstellungen nachbearbeitet werden (z.B. min/max Werte, Default Werte).

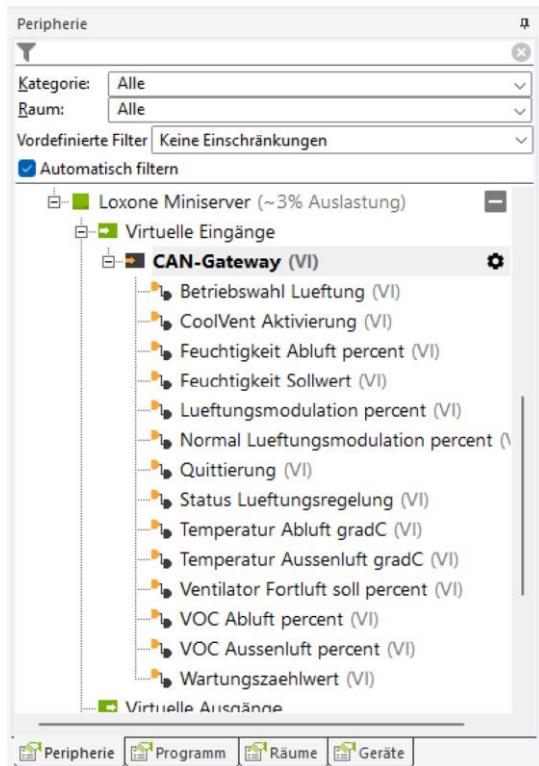
Für den Import geht man wie folgt vor. In Loxone Config wählt man bei „Peripherie“ „Virtuelle Eingänge“ und dann oben rechts klickt man auf „Vordefinierte HTTP-Geräte“ und dann auf Vorlage importieren. Dann wählt man die in CAN-Gateway erstellte Datei „VI_can_gateway_template.xml“.



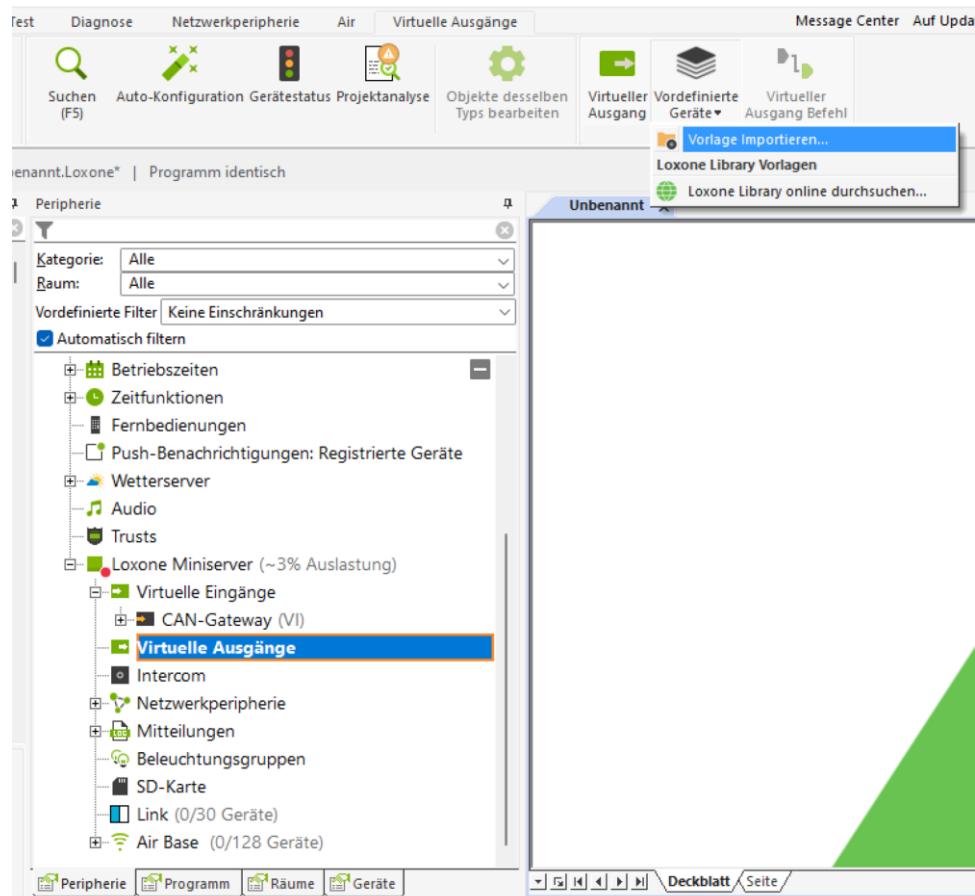
Unter „Meine Vorlagen“ erscheint dann „CAN-Gateway“. Man klickt darauf.



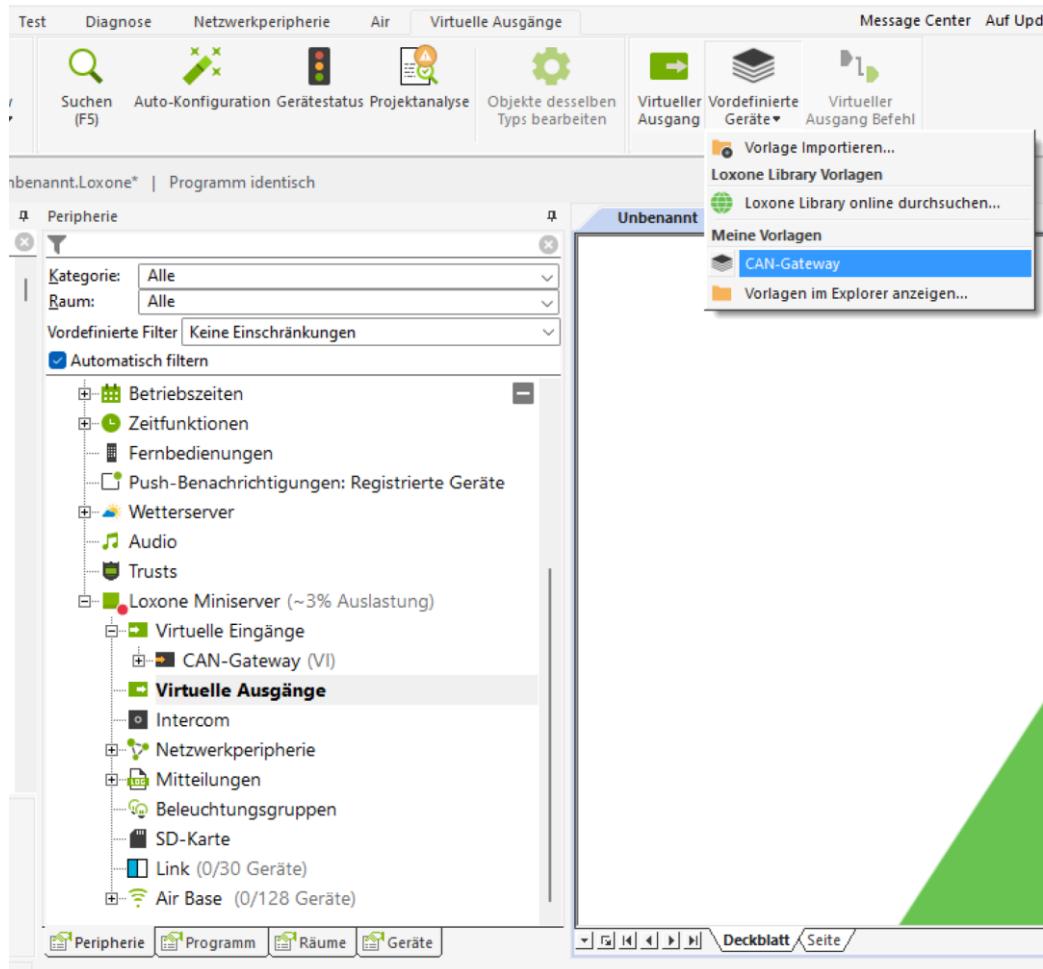
Damit werden Eingänge geladen wie in diesem Beispiel:



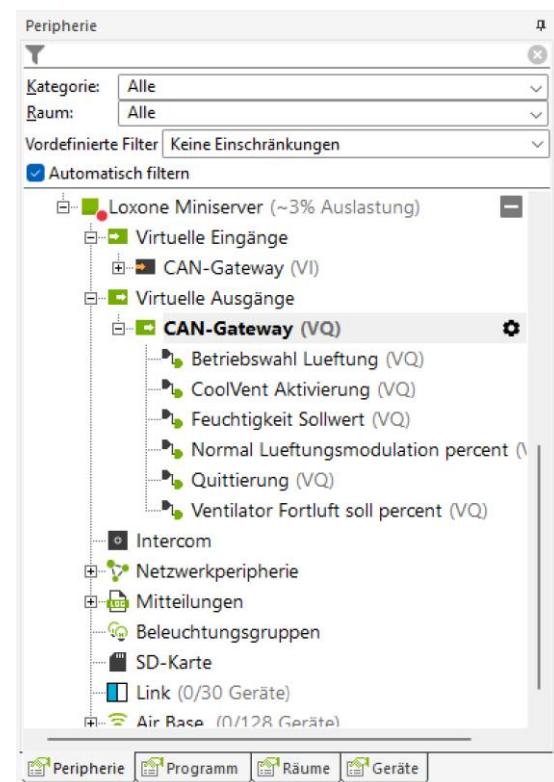
Die gleiche Vorgehensweise wendet man nun für Ausgänge an:



Hier wählt man dann die in CAN-Gateway erstellte Datei „VI_can_gateway_template.xml“. Dann erstellt man CAN-Gateway Ausgänge aus dieser Vorlage:



Das Ergebnis sieht dann beispielhaft so aus:



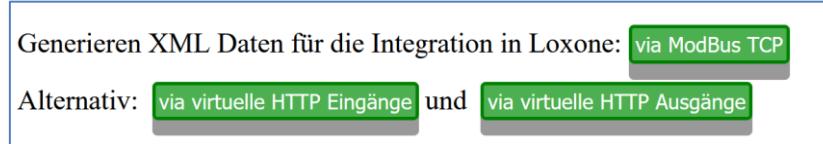
34.2 Einbindung als Modbus TCP Gerät

Verwendung von Modbus TCP auf dem CAN-Gateway ist im Kapitel 21 dieser Einleitung beschrieben.

Die Anwendung der Modbus Verbindungen in Loxone ist hier zu finden:

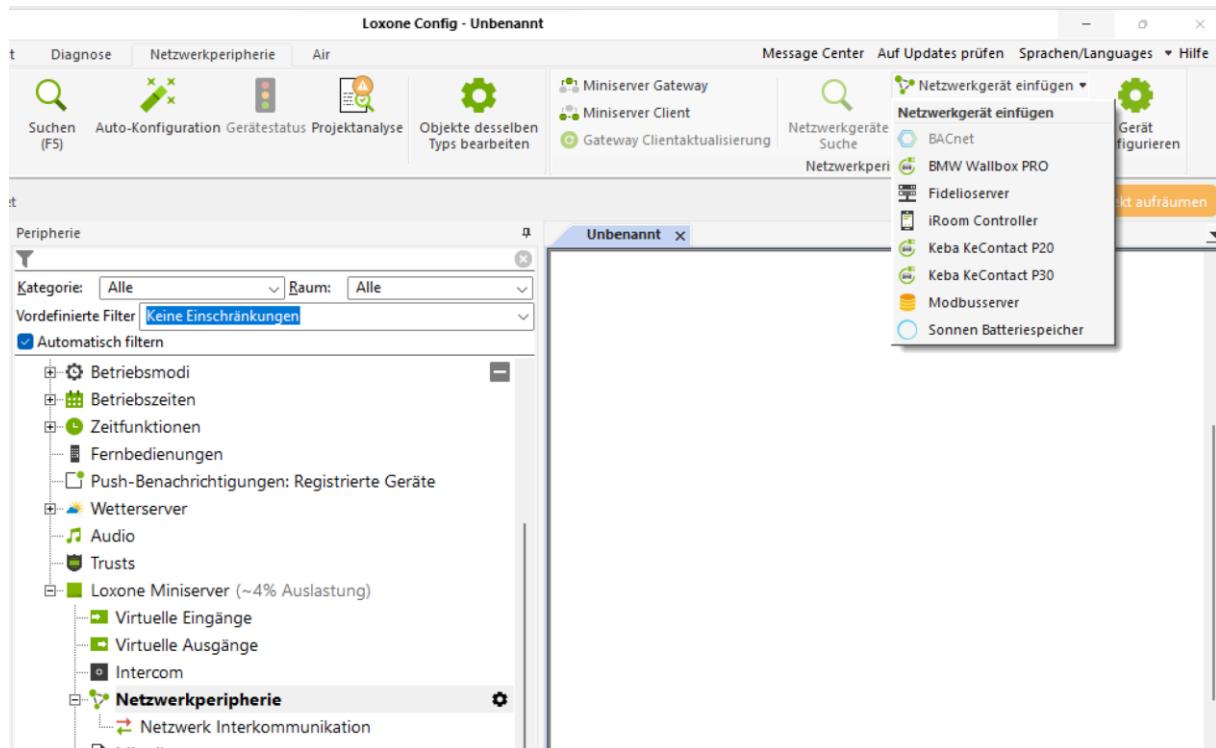
<https://www.loxone.com/dede/kb/kommunikation-mit-modbus-tcp/>.

Als Hilfestellung für die Einrichtung kann in CAN-Gateway Web Interface unter „Liste der Parameterwerte“ und dann unter der Tabelle eine XML Vorlage (MB_can_gateway_template.xml) generiert werden, die dann in Loxone importiert werden muss:

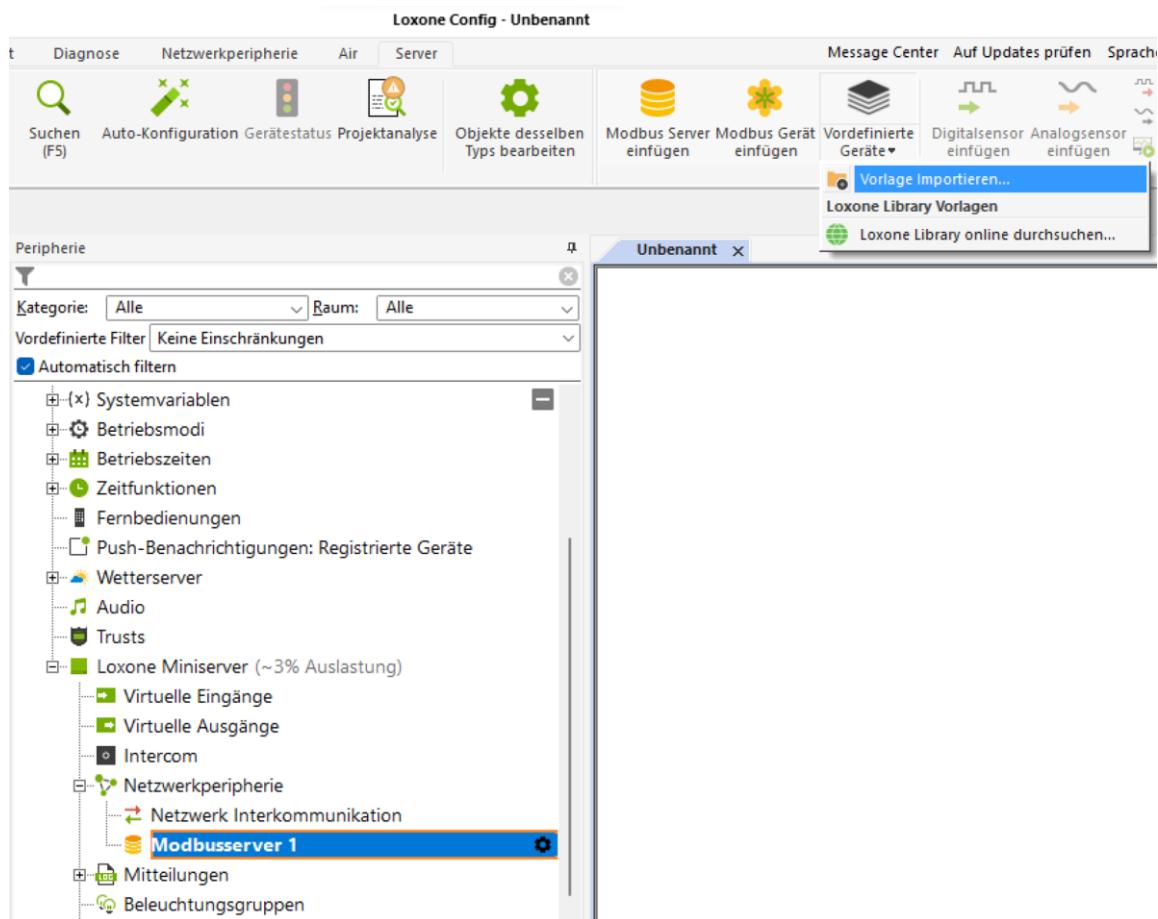


Achtung: wenn Sie die Einstellungen der Parameter in CAN-Gateway ändern, ändert sich auch diese Datei. Sie müssen dann diese neu generieren und in Loxone importieren.

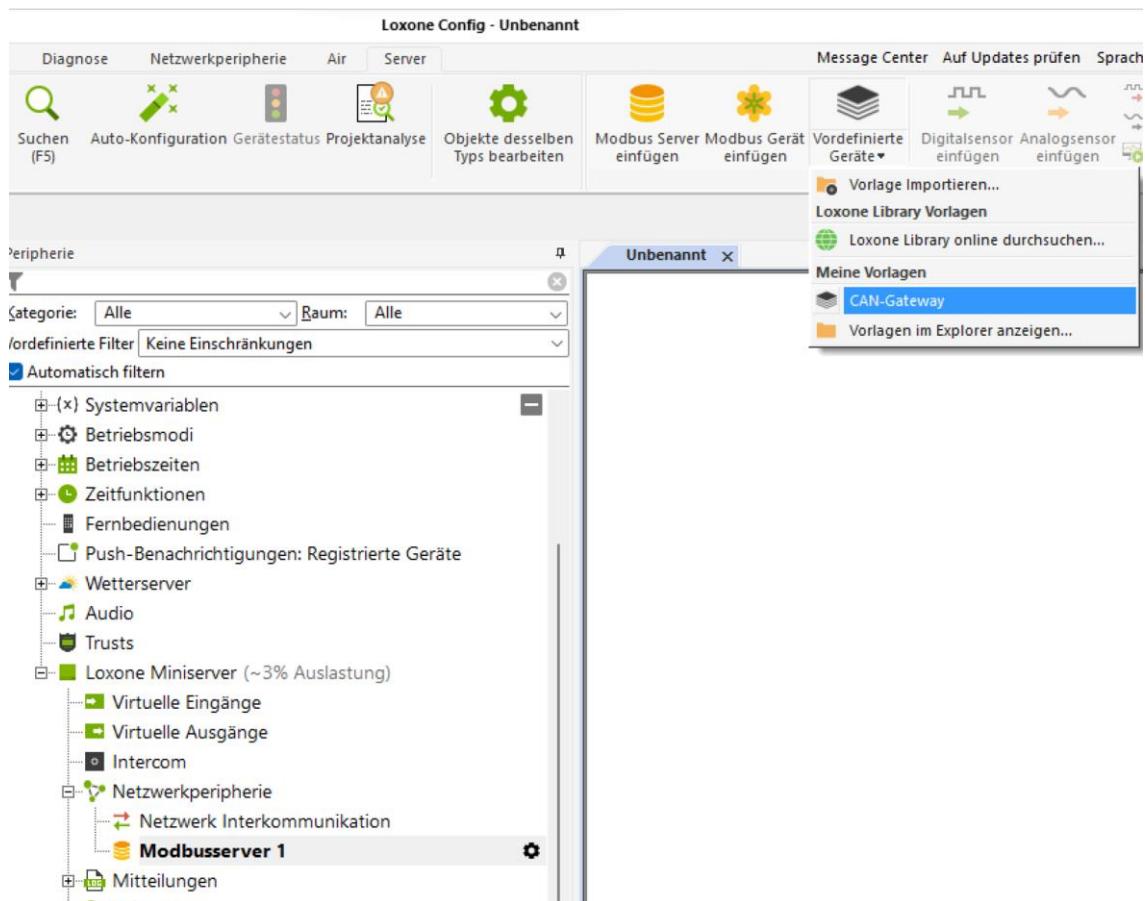
Diese Vorlage kann dann in Loxone Config wie folgt geladen werden. Zuerst geht man zu Netzwerkperipherie und dann oben rechts auf „Netzwerkgerät einfügen“ und man wählt Modbusserver.



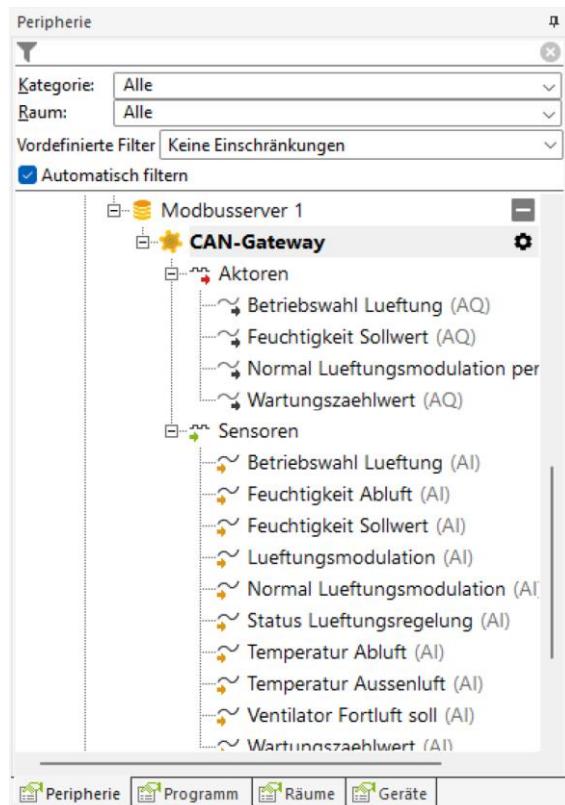
Danach klick man auf den eingefügten „Modbusserver“ und oben rechts auf „Vordefinierte Geräte“ und dann auf „Vorlage importieren“.



Dann wählt man die in CAN-Gateway erstellte Datei „MB_can_gateway_template.xml“. Nun erscheint „CAN-Gateway“ unter „Meine Vorlagen“ und man kann diese Vorlage mit einem Klick laden.



Das Ergebnis sieht beispielhaft wie folgt aus:



Nach dem diese Datei in Loxone importiert ist, können/sollen ggf. einige Einstellungen nachbearbeitet werden. Aktoren und Sensoren können nun verwendet werden.

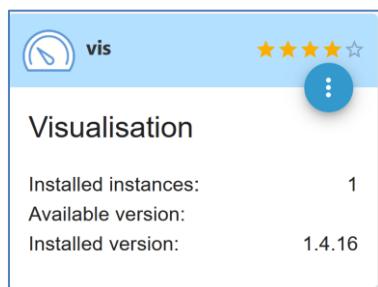
35 Einbindung in ioBroker

ioBroker stellt verschiedene Möglichkeiten dar, wie man Geräte einbinden kann. Im Folgenden wird eine Möglichkeit beschrieben, wie man CAN-Gateway an ioBroker mit Hilfe von CAN-Gateway REST-API einbinden kann. Dabei sind weder Programmierkenntnisse noch ein tieferes Verständnis von ioBroker notwendig. Es sind folgende Schritte durchzuführen:

- 1) In ioBroker unter „Adapters“ den JavaScript Adapter (Script Engine) installieren, wenn dieser noch nicht installiert ist (hier und weiter die Screenshots sind aus englischer Version von ioBroker. In Deutsch könnte es etwas anders aussehen):



- 2) In ioBroker unter „Adapters“ den VIS Adapter (Visualisation) installieren, wenn dieser noch nicht installiert ist:

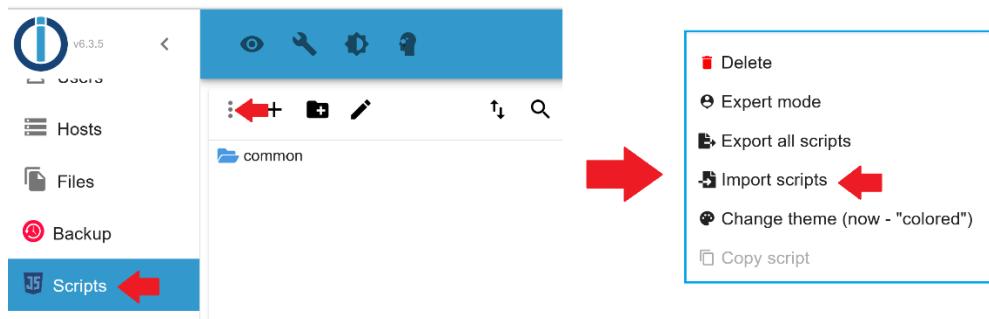


- 3) In CAN-Gateway WEB-Interface unter „Liste der Parameter“ unter der Tabelle mit dem Taster „JavaScript generieren (.zip Datei)“ eine .zip Datei erzeugen und herunterladen:

Integration in ioBroker: **JavaScript generieren (.zip Datei)**

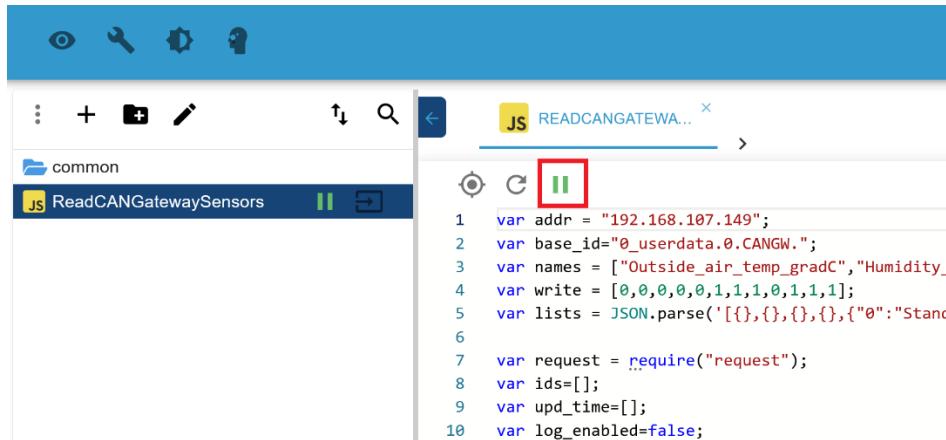
Diese Datei beinhaltet einen JavaScript, welcher in ioBroker im nächsten Schritt importiert werden kann. Dieses Skript generiert im ioBroker Objekte (States) passend zu allen in CAN-Gateway konfigurierten Parametern. Das Skript sorgt auch dafür, dass diese States mit dem CAN-Gateway verbunden werden: er liest aus dem CAN-Gateway alle 10 Sekunden alle Parameterwerte aus und setzt entsprechende States in ioBroker. Außerdem überwacht dieser Skript für Parameter, die geschrieben werden können, ob der Wert durch ioBroker (z.B. durch ein Widget oder ein Blockly Programm) geändert wird und bei Änderung sendet diese Änderung an CAN-Gateway, damit CAN-Gateway entsprechend den Hoval Gerät ansteuert.

- 4) In ioBroker unter „Scripts“ über Menu (aufrufbar durch drei Punkte untereinander) auf „Import Scripts“ klicken und die durch CAN-Gateway generierte Datei auswählen.



Nach Bestätigungen mit „OK“ wird das Skript importiert und ausgeführt. Nun müssen Sie unter „Objects“ im Bereich „_0_userdata“->„0“->“CANGW“ die generierten Objekte sehen. Achtung: wenn in CAN-Gateway die Einstellungen der Parameter geändert werden, müssen Sie dieses Skript neu generieren und erneut in ioBroker importieren.

Nach dem Import prüfen Sie, ob das Skript „ReadCAnGatewaySensors.js“ ausgeführt wird: Beim Doppelklick auf den Skript-Namen müssen Sie den Inhalt sowie das grüne Pause-Zeichen oben sehen:

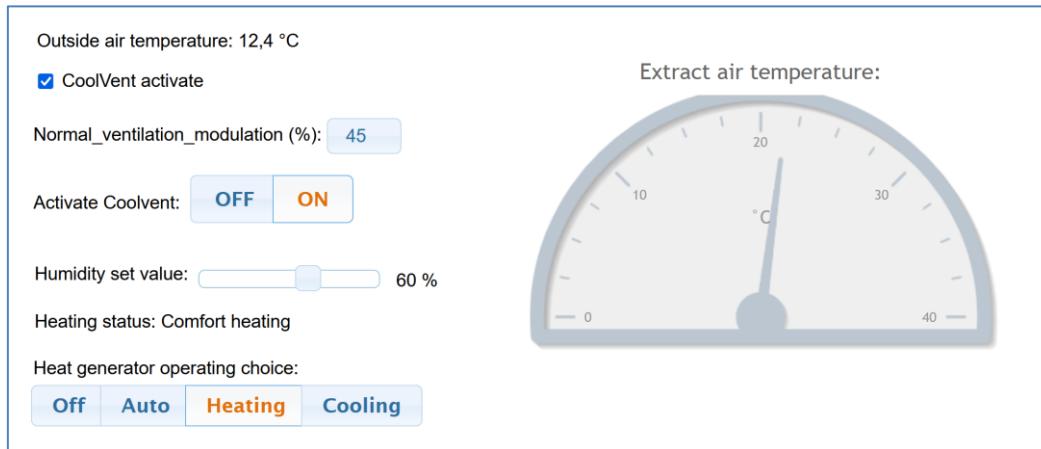


Wenn alles läuft, sehen sie dann in der Spalte „Value“ auch die entsprechenden Parameterwerte:

	ID	Name	Type	Role	Room	Function	Value
1	_0_userdata	User objects an	meta				
2	0	CANGW	folder				
3	CANGW	CoolVent_activate	state	state			1
4	CANGW	CoolVent_activate_TEXT	state	state			On
5	CANGW	Extract_air_temp_gradC	state	state			21.6
6	CANGW	Fan_exhaust_air_set_percent	state	state			48

Anmerkung zu Parametern, die ein Liste darstellen: für Parameter, die eine Liste darstellen (siehe Kapitel 10.2 in dieser Einleitung) wird jeweils zwei Objekte erzeugt: ein mit dem originalen ID/Namen und ein mit dem Zusatz „_TEXT“ beim ID bzw. „(textual)“ beim Namen. Dieser Zusatzobjekt beinhaltet dann den Zustand in Klartext und kann einfacher für eine Textanzeige verwendet werden.

- 5) Nun können in VIS visuelle Elemente erstellt und mit den Objekten verknüpft werden, um die Parameterwerte anzuzeigen oder die beschreibbaren Parameter zu ändern. Einige einfache Beispiele:



Die oben beschriebenen Schritte sind in einer separaten Datei detaillierter mit Bildern beschrieben:

https://raw.githubusercontent.com/wladwnt/CAN-Gateway/master/ioBroker_configuration.pdf

35.1 Problemlösungen und Hinweise für Fortgeschrittene ioBroker Anwender

Sollte irgendwas nicht wie gewünscht funktionieren, gibt es folgende Optionen:

- 1) Prüfen, ob JavaScript Adapter läuft: unter „Instances“ muss es grün sein:

Filter					
1 Adapters		admin.0		Admin	86.75 MB
Instances		backup.0		BackUp	66.42 MB
Objects		discovery.0		Discovery devices	63.44 MB
Enums		javascript.0		Script Engine	124.05 MB
Logs		vis.0		Visualisation	
Users		web.0		WEB server	70.91 MB

- 2) In dem Skript die erste Zeile prüfen, ob da die IP Adresse vom CAN-Gateway die richtige ist. Ggf. korrigieren.
- 3) Logs prüfen
- 4) In dem Skript die Zeile Nummer 10 mit „var log_enabled=false;“ finden und anstatt „false“ „true“ schreiben. Dann in Logs wieder schauen: das Skript schreibt dann in die Logs einige Zusatzinformationen, die evtl. helfen.

36 Einschränkungen der Demo-Version

Die sogenannte „Demo-Version“ hat im Vergleich zur vollen Version folgende Einschränkungen:

- Es können maximal zwei Parameter konfiguriert werden anstatt von 40.
- Die Software schaltet sich automatisch 60 Minuten nach dem letzten Power On aus. Danach muss CAN-Gateway manuell durch unterbrechen der Stromversorgung neu gestartet werden.
- Es können keine User-Level Passwörter aus dem Bedienmodul ausgelesen werden.

Alles andere funktioniert genauso wie in Vollversion. Damit haben Sie die Möglichkeit alles zu testen bevor Sie sich eine Vollversion anschaffen.

37 Over-The-Air Update

Die Software des CAN-Gateways beinhaltet ArduinoOTA Modul, kann also Over-The-Air upgedated werden. Das bedeutet auch, dass Sie z.B. in Arduino IDE den cangateway als update-fähiges Board sehen werden. Hostname und Kennwort für OTA sind beides „cangateway“. Es wird allerdings empfohlen Software Update über WEB-Interface (siehe Kapitel „WEB-Interface: Software Update“) durchzuführen.

38 Lizenzen

Diese Software basiert unter anderem auf folgenden Software-Komponenten:

- Espressif IoT Development Framework (ESP-IDF, Copyright 2019 Espressif Systems (Shanghai) PTE LTD, Licensed under the Apache License, Version 2.0, Source:
<https://github.com/espressif/esp-idf>)

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

- Arduino core for the ESP32 (For copyright information please refer to the source files.

Licensed under GNU Lesser General Public License v2.1, Source:

<https://github.com/espressif/arduino-esp32>)

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain

special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

****0.**** This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

****1.**** You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

****2.**** You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- ****a)**** The modified work must itself be a software library.
- ****b)**** You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- ****c)**** You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- ****d)**** If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

****3.**** You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

****4.**** You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

****5.**** A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it

contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

****6.**** As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- ****a)**** Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
- ****b)**** Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.
- ****c)**** Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
- ****d)**** If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
- ****e)**** Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot

use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- **a)** Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- **b)** Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License

may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

- AsyncTCP Library (Copyright 2016 Hristo Gochkov, Licensed under GNU Lesser General Public License v3.0, Source: <https://github.com/me-no-dev/AsyncTCP>)

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or
- b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
- c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among

these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.
- b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

- Asynchronous WebServer library for Espressif MCUs (Copyright 2016 Hristo Gochkov, Licensed under GNU Lesser General Public License v2.1, Source: <https://github.com/me-no-dev/ESPAsyncWebServer>)

Copyright (c) 2016 Hristo Gochkov. All rights reserved.
This file is part of the esp8266 core for Arduino environment.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

- ModbusRTU and ModbusIP Master-Slave Library for ESP8266/ESP32 v3.0 (Copyright 2015, Andre Sarmento Barbosa, 2017 Alexander Emelianov (a.m.emelianov@gmail.com), Licensed under License <https://github.com/emelianov/modbus-esp8266/blob/master/LICENSE.txt>, Source: <https://github.com/emelianov/modbus-esp8266>)

Copyright (c) 2015, Andre Sarmento Barbosa
2017, Alexander Emelianov (a.m.emelianov@gmail.com)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Dygraphs: JavaScript charting library, Copyright 2017 Dan Vanderkam (danvdk@gmail.com), Licensed under The MIT License (MIT), Source: <https://github.com/danvk/dygraphs>

Copyright (c) 2009 Dan Vanderkam

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- **SSDP library: Copyright (c) 2015 Hristo Gochkov, Source und License: <https://github.com/luc-github/ESP32SSDP>**

ESP32 Simple Service Discovery Copyright (c) 2015 Hristo Gochkov
Original (Arduino) version by Filippo Sallemi, July 23, 2014. Can be
found at: <https://github.com/nomadnt/uSSDP>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- **KNX IP library: Copyright (c) 2018 Nico Weichbrodt, Source und License: <https://github.com/envy/esp-knx-ip>**

MIT License

Copyright (c) 2018 Nico Weichbrodt <nico@weichbrodt.me>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.