

Bedienungsanleitung



CAN-Gateway Software

Stand: SW Version 17.001

Inhaltsverzeichnis

Wichtige Hinweise – unbedingt durchlesen!	2
Einleitung	4
Benötigte Hardware und Aufbau	5
Software flashen	8
Erste Inbetriebnahme	12
WEB-Interface: Hauptseite	15
Allgemeine Informationen zu Gerät-Parametern	16
Beispiel	17
User-Levels	18
WEB-Interface: Einstellungen	19
WEB-Interface: Parameter Einstellungen	20
Datentypen	21
Hinweise zu Listen	22
Beispiele	22
Beispiel 1: Komfortlüftung – Feuchte Sollwert auslesen und einstellen	22
Beispiel 2: Komfortlüftung – Temperatur Abluft	23
Beispiel 3: Komfortlüftung – Temperatur Außenluft	23
Beispiel 4: Raumbedienmodul – Kennwort für UserLevel 7 lesen/ändern	23
Beispiel 5: Komfortlüftung – Betriebsmodus	23
WEB-Interface: Parameter-Konfiguration Generator	24
WEB-Interface: Liste der Parameter	25
WEB-Interface: Liste der Geräte	26
WEB-Interface: Status Informationen	27
WEB-Interface: CAN Datenlogger	28
WEB-Interface: Parameter Logger	29
WEB-Interface: Software Update	30
Übertragung der Parameter über MQTT Protokoll	31
Übertragung der Parameter über Modbus TCP	32
Hinweise zu Modbus TCP Kommunikation	32

Modbus Adressen.....	32
Register Typen.....	32
Übertragungswerte	33
Steuerung über Bedienmodul	34
REST-API	36
Besondere Hinweise zu Parameteränderungen.....	37
Hilfe!	37
Netzwerkname/Schlüssel sind falsch eingestellt, Web-Interface nicht zugänglich. Was tun?	37
CAN-Gateway startet immer mit Werkeinstellungen. Einstellungen können nicht gespeichert werden.	37
Einschränkungen der Demo-Version.....	38
Over-The-Air Update	38
Lizenzen	38

Wichtige Hinweise – unbedingt durchlesen!

Diese Software ist eine Test- und Lernsoftware, die nur zur Test- und Lernzwecken gedacht ist und nicht für den täglichen, unbeaufsichtigten Betrieb geeignet ist. Geräte, die diese Software verwenden, sind entsprechend auch nur für Test- oder Lernzwecke und nur beaufsichtigt einzusetzen. Beaufsichtigt bedeutet dabei, dass dieses Gerät und alle damit verbundenen Geräte ständig von einer Person überwacht werden müssen, die aufgrund Ihrer Ausbildung oder Erfahrung in der Lage ist Fehler zu erkennen und alle Geräte notfalls innerhalb weniger Sekunden abzuschalten.

Die Software und Ihre Dokumentation wird "wie sie ist" und ohne jede Gewährleistung für Funktion, Korrektheit oder Fehlerfreiheit zur Verfügung gestellt. Für jedweden direkten oder indirekten Schaden - insbesondere Schaden an anderer Software, Schaden an Hardware, Schaden an Geräten, Schaden an Gebäuden und Einrichtungen, Schaden durch Nutzungsausfall und Schaden durch Funktionsuntüchtigkeit der Software, kann der Autor nicht haftbar gemacht werden. Ausschließlich der Benutzer haftet für die Folgen der Benutzung dieser Software. Ausgenommen sind die Fälle, bei denen nach dem aktuell geltenden Recht in Deutschland der Autor der Software sich nicht von der Haftung befreien kann (wie z.B. grobe Fahrlässigkeit, Täuschung etc.).

Diese Software basiert auf der Analyse der Kommunikation zwischen Geräten (im Speziell WLAN Gateway und Bedienmodul mit SW Version 2.xx.xx) 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 anwendbar ist, was unter anderem ein Grund für mögliche Fehler sein kann.

Die Verwendung dieser Software in Verbund mit anderen Geräten kann evtl. dazu führen, dass die Garantie- und Gewährleistungsansprüche für diese Geräte sich erlöschen. Der Benutzer dieser Software ist selbst dafür zuständig das mögliche Erlöschen der Garantie- und Gewährleistungsansprüche zu prüfen.

Der Hersteller Hoval AG vertreibt selbst Gateways für die Anbindung seiner Geräte an die Hausautomatisierungssysteme unter anderem über KNX oder ModBus RTU/TCP Busse/Protokolle. Der Autor dieser Software empfiehlt diese Gateways für den produktiven Einsatz und nicht die hier beschriebene Testsoftware.

Einleitung

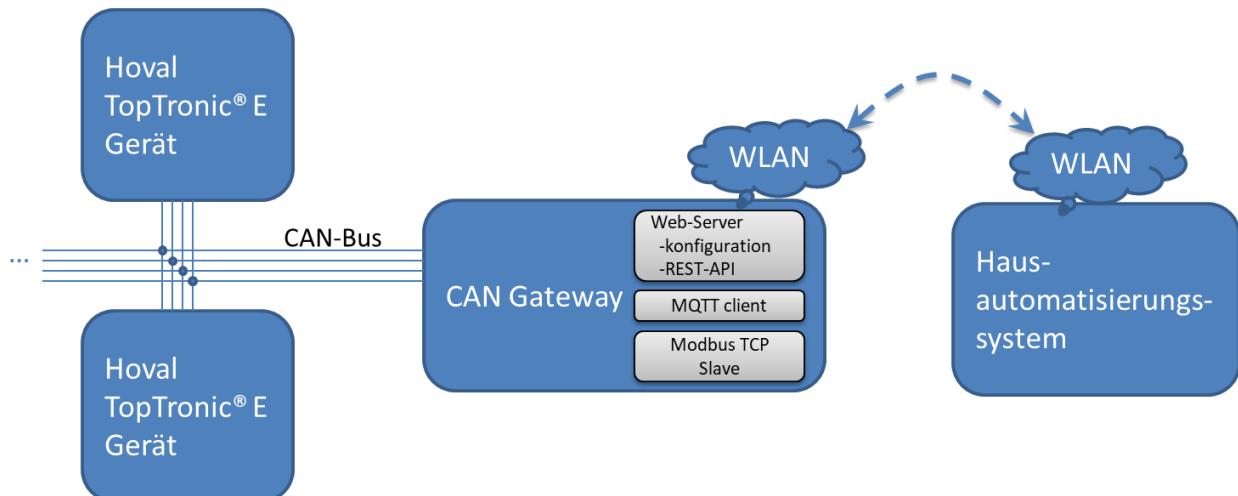
Diese Software erlaubt es testweise die Geräte der Serie „TopTronic® E“ (TopTronic® E ist die geschützte und eingetragene Marke der Firma Hoval AG) der Firma **Hoval Aktiengesellschaft** an das Hausautomatisierungssystem anzubinden. Diese Geräte werden im Folgenden einfach „Geräte“ genannt.

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 für Testzwecke möglich ist.

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

ESP32 DevKit Board besitzt einen WLAN Interface, so dass das Hausautomatisierungssystem kann über WLAN mit dem CAN-Gateway verbunden werden. Als Datenübertragungsprotokolle zwischen CAN-Gateway und dem Hausautomatisierungssystem können wahlweise MQTT Protokoll, Modbus TCP 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

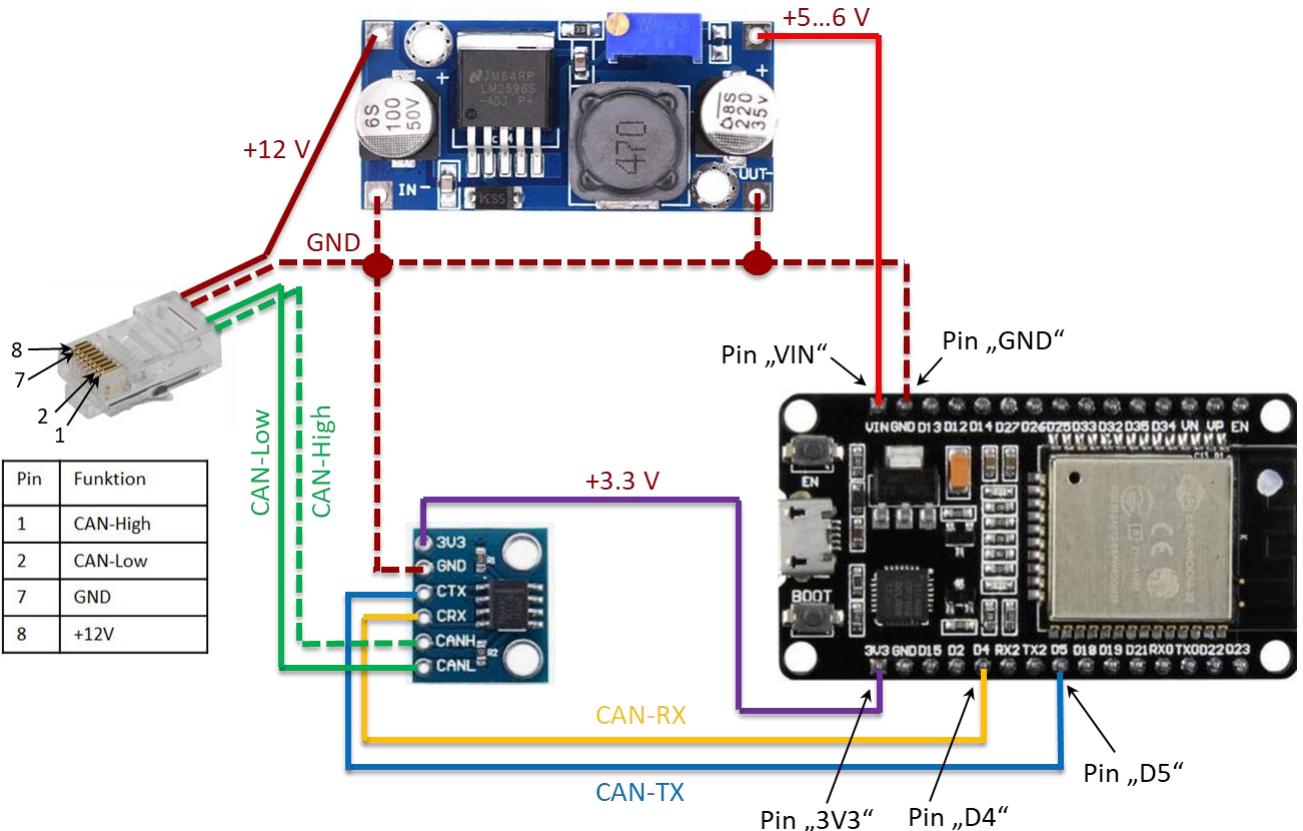
Benötigte Hardware und Aufbau

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	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!

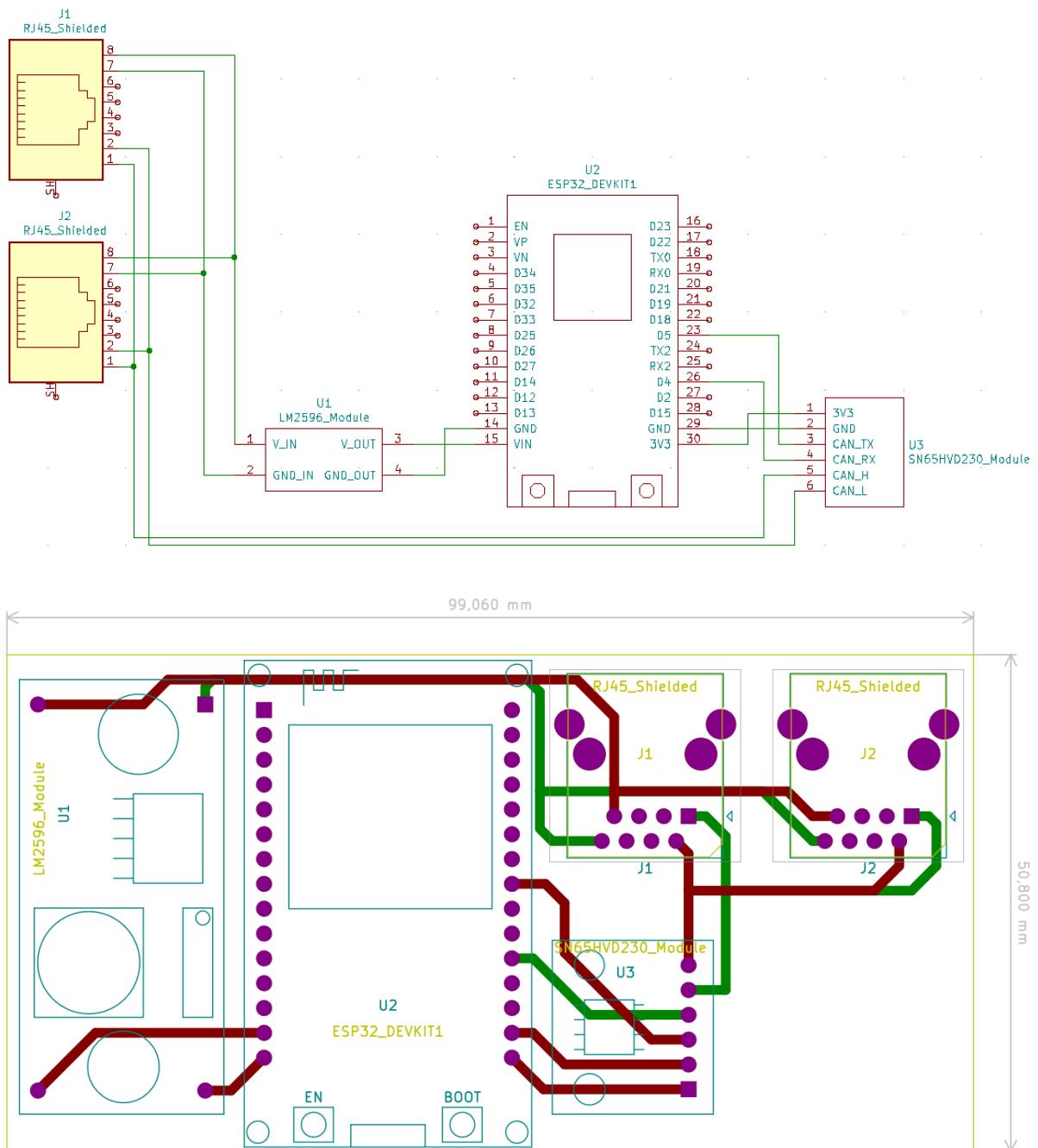
Der Aufbau ist wie folgt. **Achtung: LM2596 Modul sollte man zuerst mit Hilfe eines Multimeters auf die Ausgangsspannung von ca. 5 V einstellen.** Dafür dreht man die Schraube an dem einstellbaren Widerstand (hellblau im Bild). ESP32 Modul kann zwar kurzzeitig auch mit 12 V funktionieren, wird jedoch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



Wichtige Hinweise

- 1) GND-Pin am Ethernet: Es ist wohl so, dass in einigen (ganz früheren?) Versionen der Geräte der GND-Pin ist nicht Nummer 7, sondern Nummer 3. Oder evtl. ist das je nach Gerät unterschiedlich.
- 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. 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. Für mehr Details dazu lesen Sie bitte die Beschreibung zum CAN-Busen (z.B. auf Wikipedia etc.)

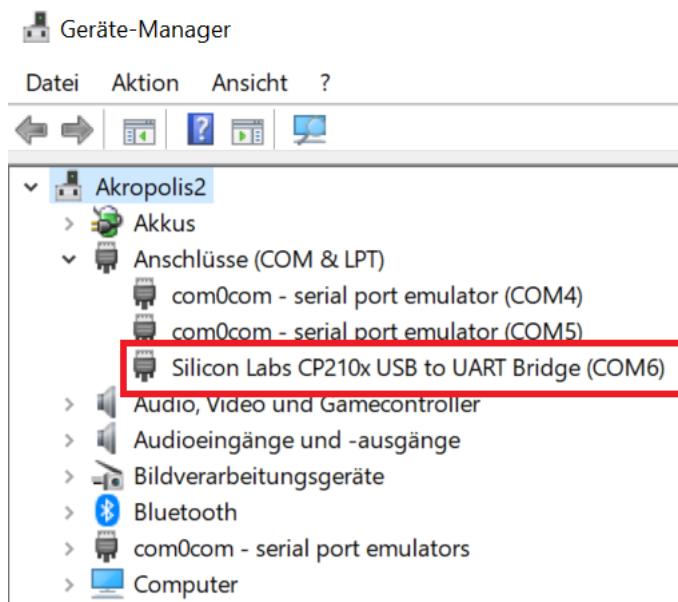
Man kann auch eine RJ45 Buchse verwenden und die einzelnen Module auf einer PCB z.B. so einordnen (hier Screenshots aus KiCad). Hier ist ein Beispiel mit zwei RJ45 Buchsen.



Software flashen

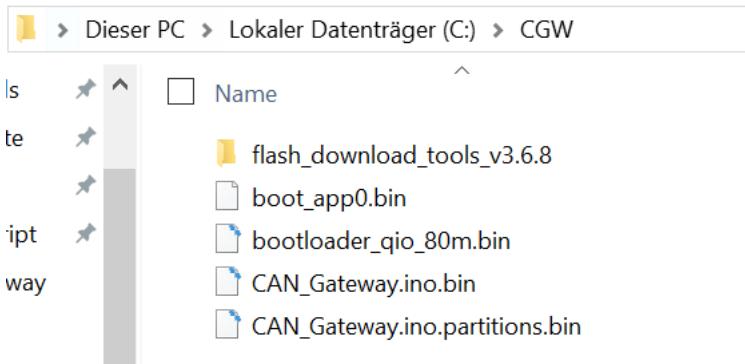
Die Software wird als mehrere binäre Dateien zur Verfügung gestellt und muss zuerst auf ESP32 DevKit Board geflasht werden. Dafür muss wiederum zunächst das Board mit dem Rechner über USB Kabel verbunden und der Treiber für den Kommunikationschip installiert werden. Insgesamt folgende Schritte sind erforderlich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

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

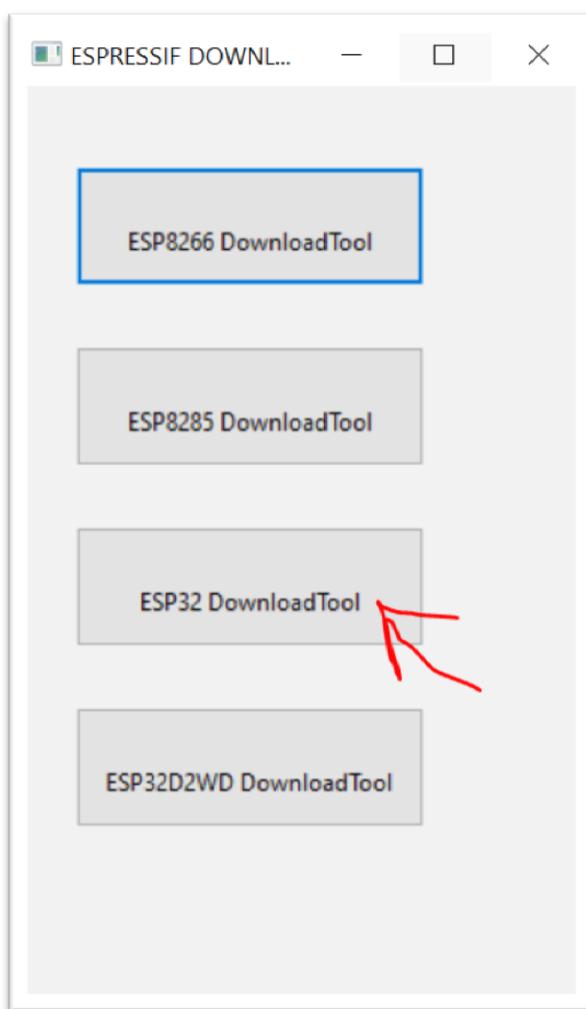


- 7) Erstellen Sie ein Verzeichnis auf dem Laufwerk C; z.B. **C:\CGW**. Sie können es auch anders benennen, müssen aber dann in weiteren Schritten überall den richtigen Namen verwenden.
- 8) Herunterladen und entpacken Sie ESP Download Tool in diese Verzeichnis (<https://www.espressif.com/en/products/hardware/esp32/resources>, dann unter „Tools“ „Flash Download Tools (ESP8266 & ESP32)“)
- 9) Entpacken Sie die CAN-Gateway Software (die Dateien „CAN_Gateway.ino.bin“ (bzw. „CAN_Gateway_DEMO.ino.bin“ für DEMO Version) und „CAN_Gateway.ino.partitions.bin“) auch in dieses Verzeichnis.
- 10) Zudem müssen Sie noch die Dateien „boot_app0.bin“ (Quelle: https://github.com/espressif/arduino-esp32/blob/master/tools/partitions/boot_app0.bin) und (bootloader_qio_80m.bin, Quelle: https://github.com/espressif/arduino-esp32/blob/master/tools/sdk/bin/bootloader_qio_80m.bin) in diesem Verzeichnis speichern.

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

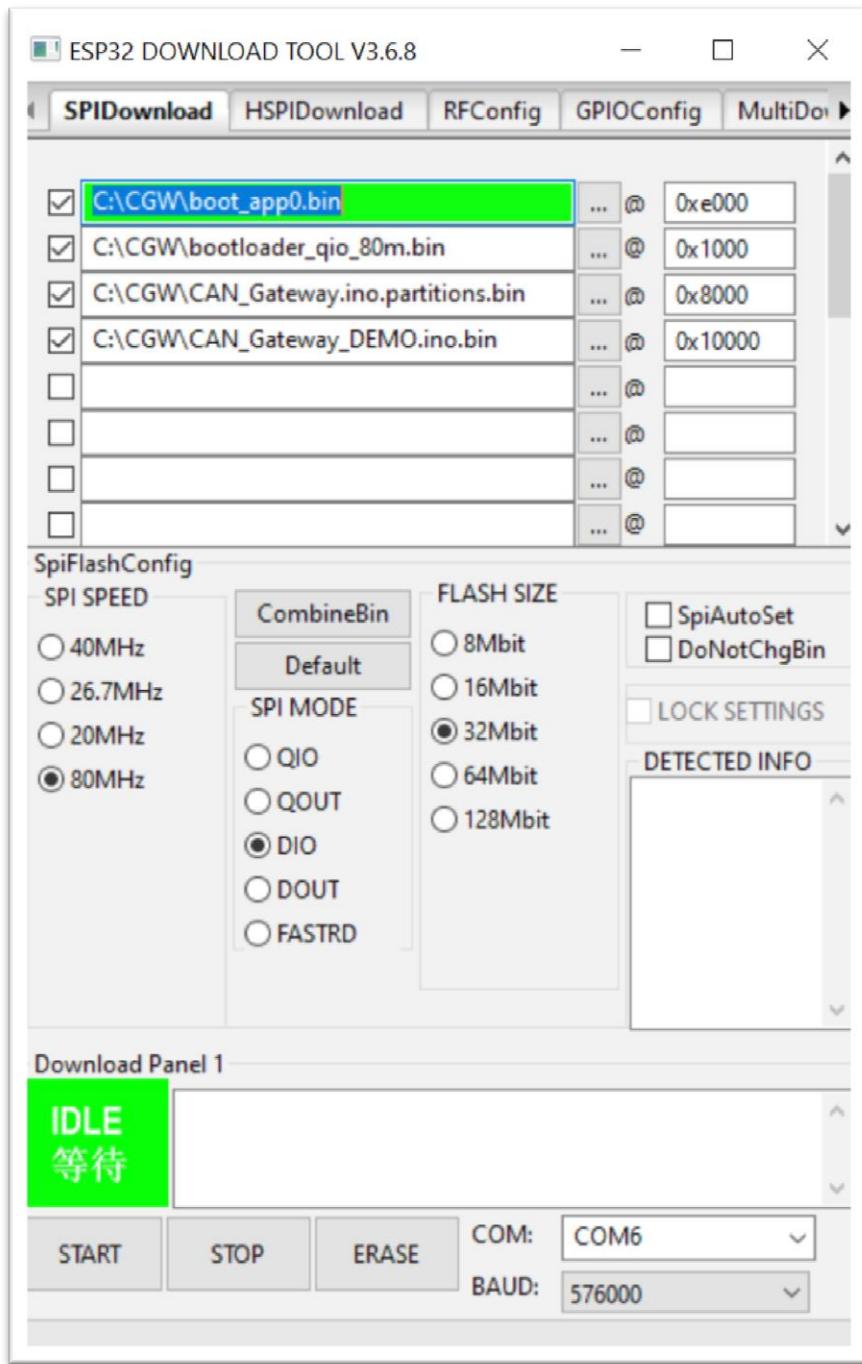


- 12) Nun können Sie ESP32 Download Tool starten (exe Datei aus dem Verzeichnis flash_download_tool vX.X.X, wobei X.X.X die aktuelle Version ist, die Sie runtergeladen haben)
- 13) Dann erscheint ein Fenster, wo Sie „ESP32 DownloadTool“ anklicken.

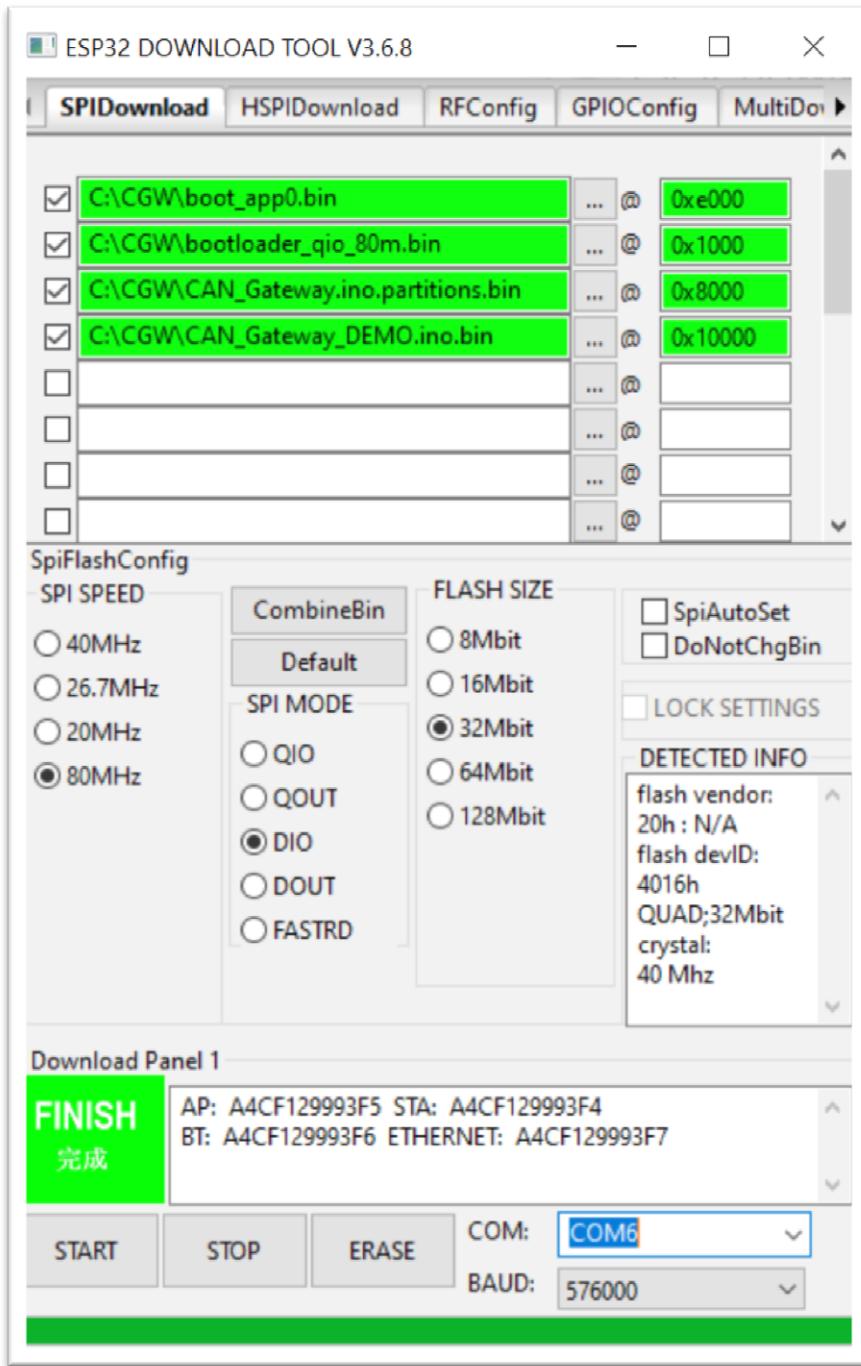


- 14) Im nächsten Dialog geben Sie die Daten ein, genauso wie hier im Bild zu sehen ist. Das einzige was Sie anpassen müssen, ist den richtigen COM-Port auszuwählen (unten rechts). Wenn man Probleme beim Flashen hat, hilft evtl. auch wenn man die Baudrate niedriger wählt. Passen Sie auf, dass alle Haken 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
```

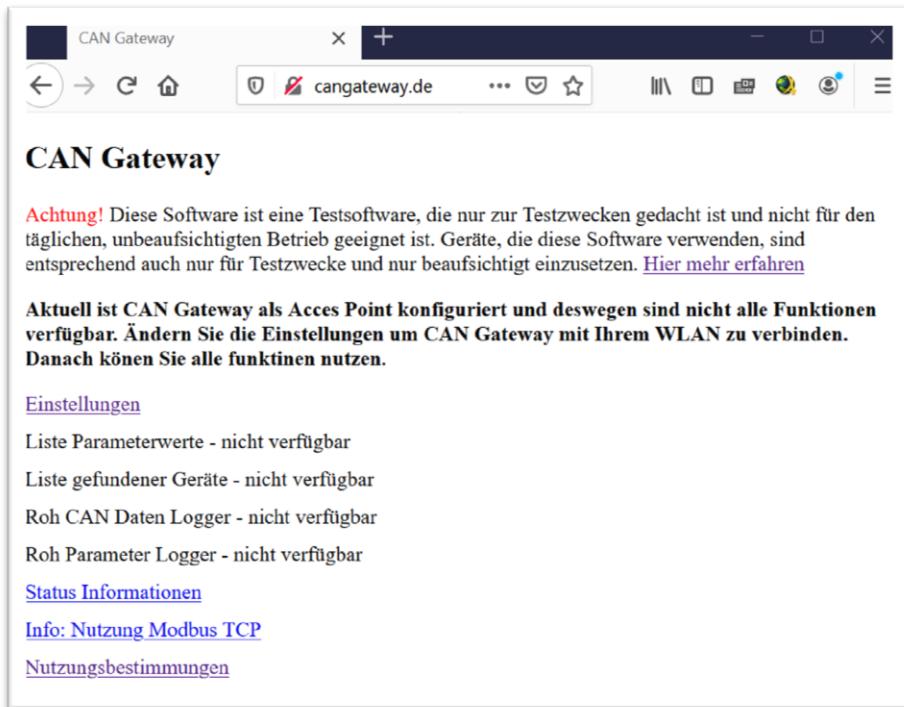
Erste Inbetriebnahme

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):

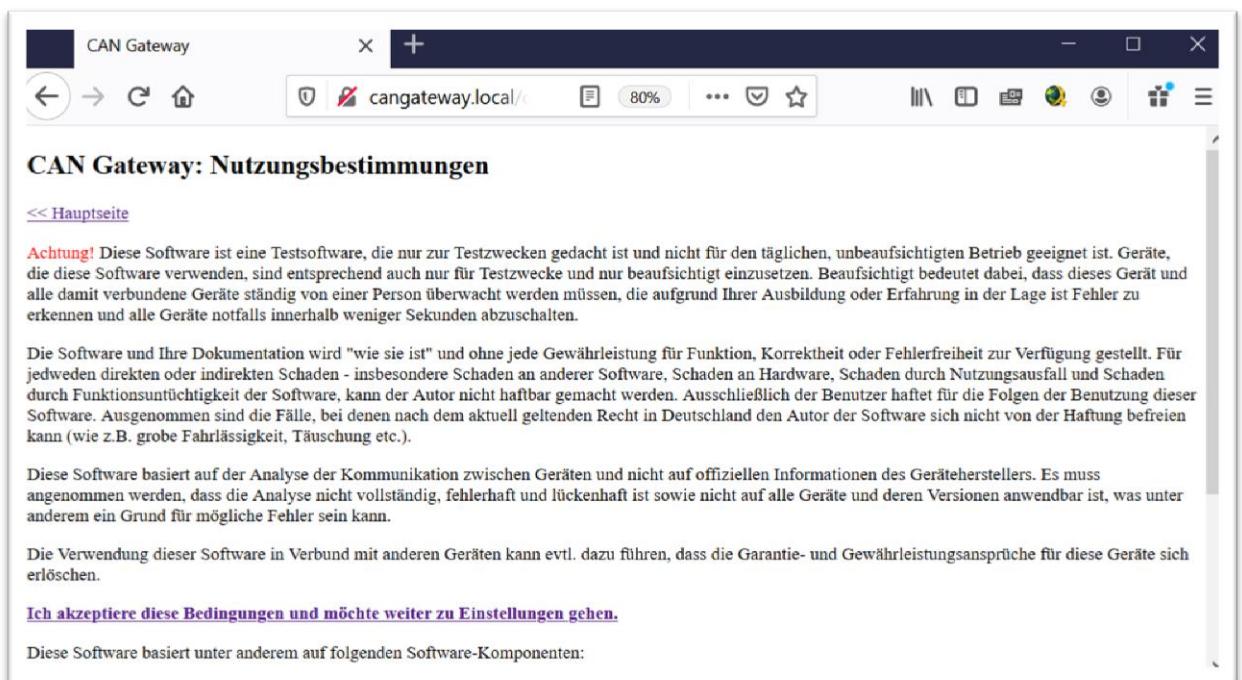
- 1) Stellen Sie sicher, dass Sie Ihre aktuellen WLAN Zugangsdaten kennen. Sie werden ihr PC im nächstens Schritt für eine Weile zu einem anderen Netzwerk verbinden müssen, danach müssen Sie aber zurück ins Ihr Heimnetzt und das wird ggf. erfordern, dass Sie Ihre Zugangsdaten (Password) eingeben müssen.
- 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“.



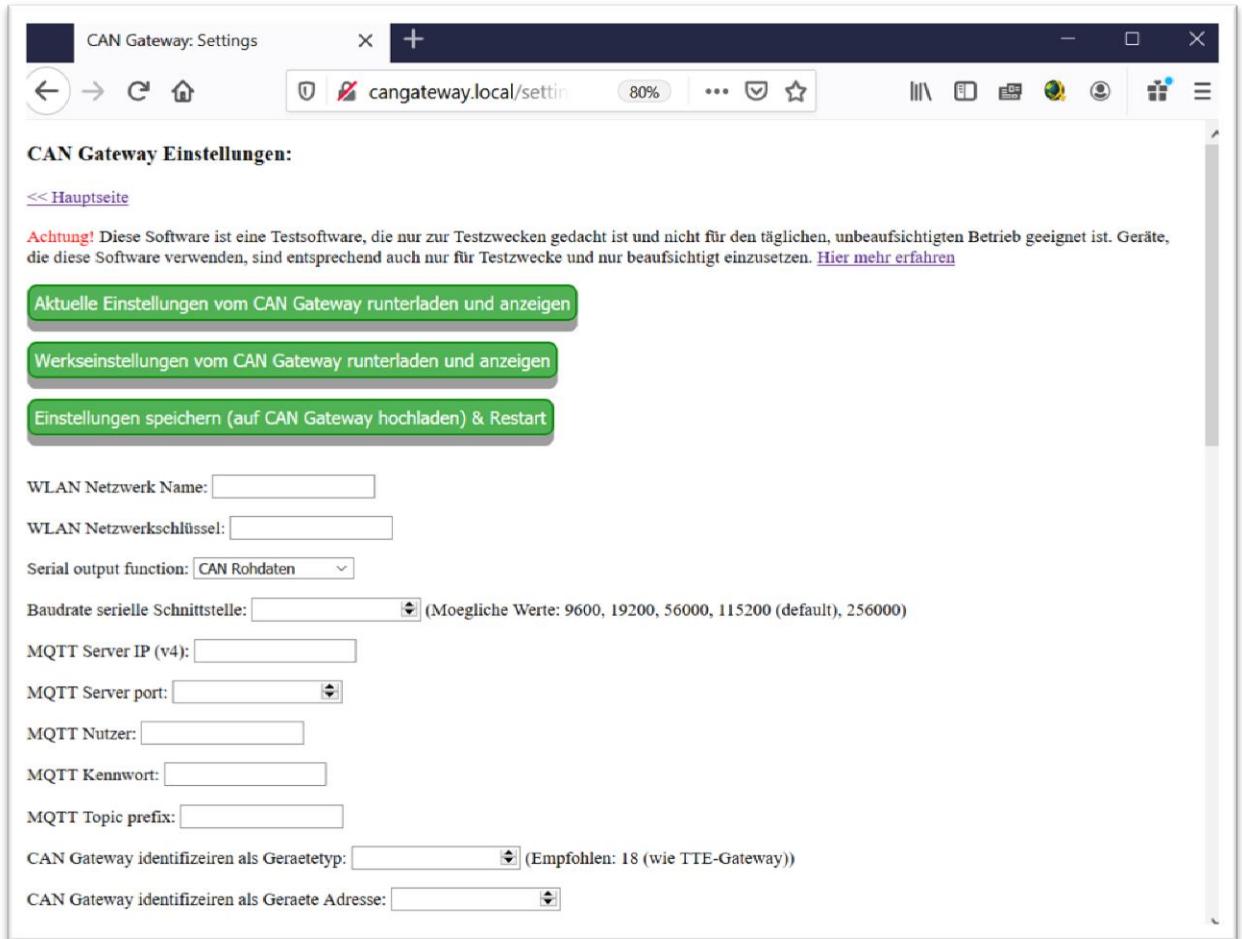
- 3) Es werden Netzanforderungen überprüft, danach müssen Sie den Netzwerkschlüssel 000999555 eingeben und auf „Weiter“ klicken.
- 4) Warten Sie bis die Verbindung zustande kommt und unter „cangateway“ erscheint „Kein Internet, gesichert“
- 5) Sie sind jetzt mit dem CAN-Gateway verbunden.
- 6) Öffnen 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.



- 7) Danach klicken Sie auf „Einstellungen“ und lesen dann aufmerksam die Nutzungsbestimmungen. Sind Sie damit einverstanden, können Sie auf „[Ich akzeptiere diese Bedingungen und möchte weiter zu Einstellungen gehen.](#)“ Klicken:



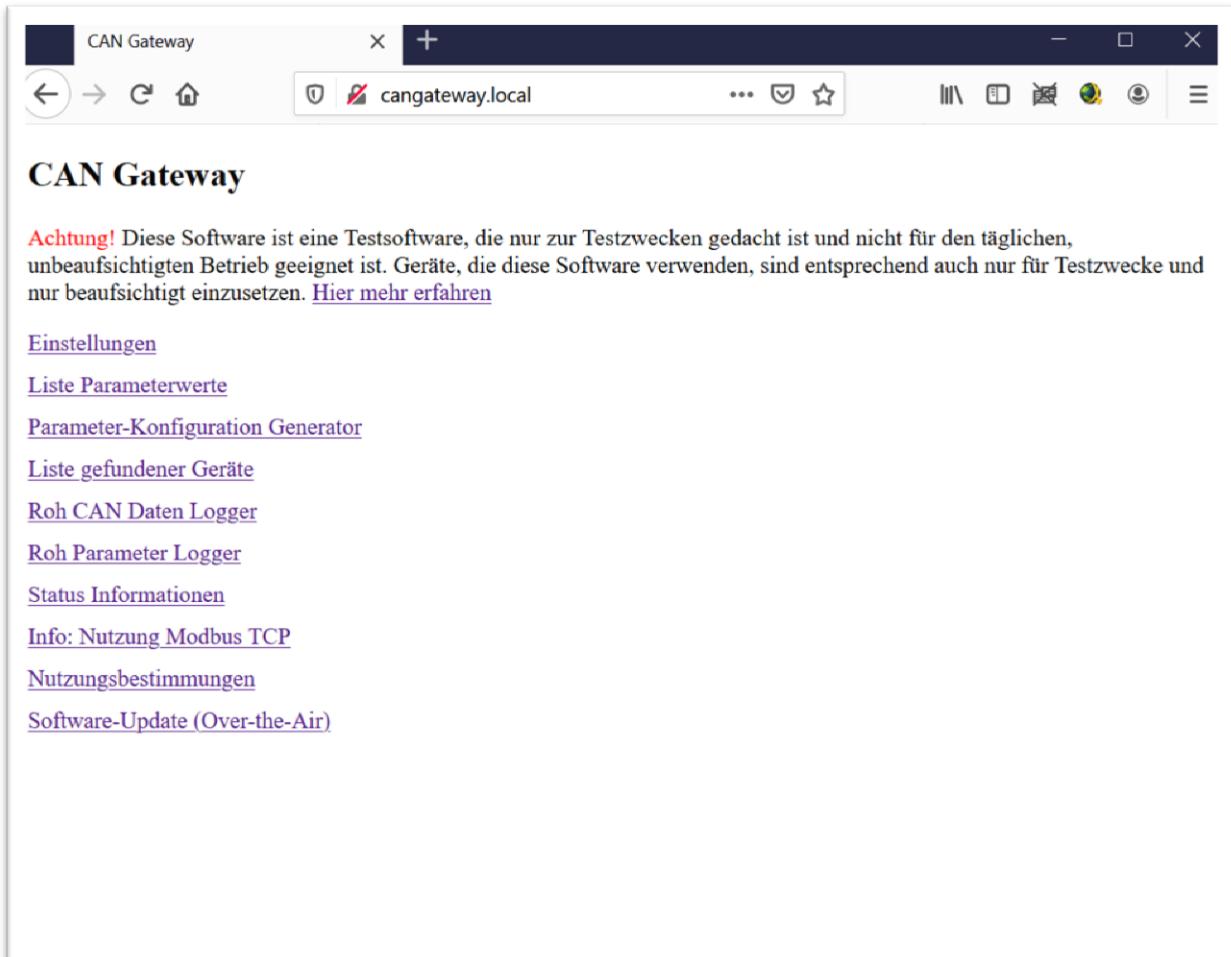
- 8) Auf nächster Seite klicken Sie zuerst auf „Aktuelle Einstellungen vom CAN-Gateway runterladen und anzeigen“. Als erstes können Sie Ihre Heim-WLAN Name und Password (Netzwerkschlüssel) eingeben (WiFi Name/Password). Alle andren Einstellungen können Sie erstmal so lassen wie es ist. Danach klicken 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.



- 9) Nach Neustart versucht CAN-Gateway sich mit Ihrem Heim-Netzwerk verbinden. Damit es funktioniert, muss Ihr WLAN Router unter anderem WPA/WPA2 sowie DHCP unterstützen und keine neuen Netzwerkteilnehmer blockieren (also nicht z.B. nach 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. Meine Erfahrung zeigt, dass Windows 10 neue Netzwerkteilnehmer nur nach Neustart erkennt. Sie können nun also wieder Web-Browser öffnen und 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.

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.



Allgemeine Informationen zu Geräte-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, Geräte-Namen oder Sollwerte für Temperatur, Luft-Durchströmung etc. genannt werden. Verschiedene Geräte-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. Stadtartadresse 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

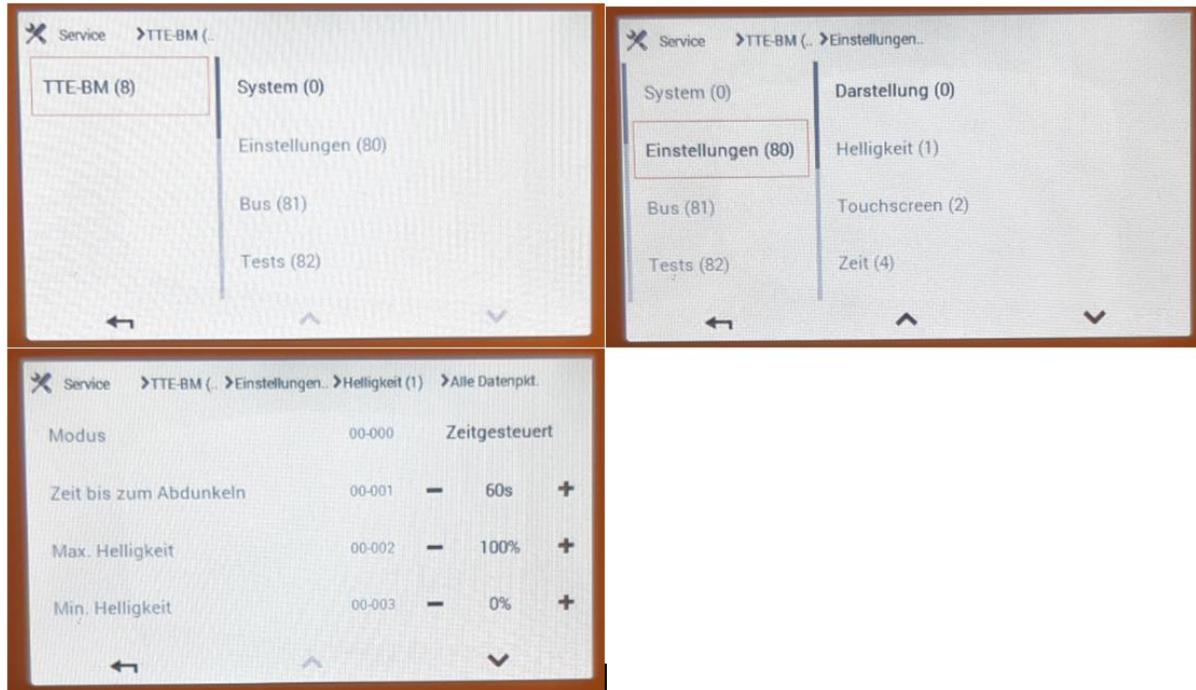
dem CAN-Bus nichts übertragen wird, wird im Parameter-Logger vom CAN-Gateway auch nichts angezeigt. Parameter, die in den Einstellungen des CAN-Gateways konfiguriert sind, werden vom CAN-Gateway periodisch über CAN-Bus angefragt und wenn die Geräte antworten, erscheinen dann diese Parameter auch im Parameter-Logger.

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.

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.



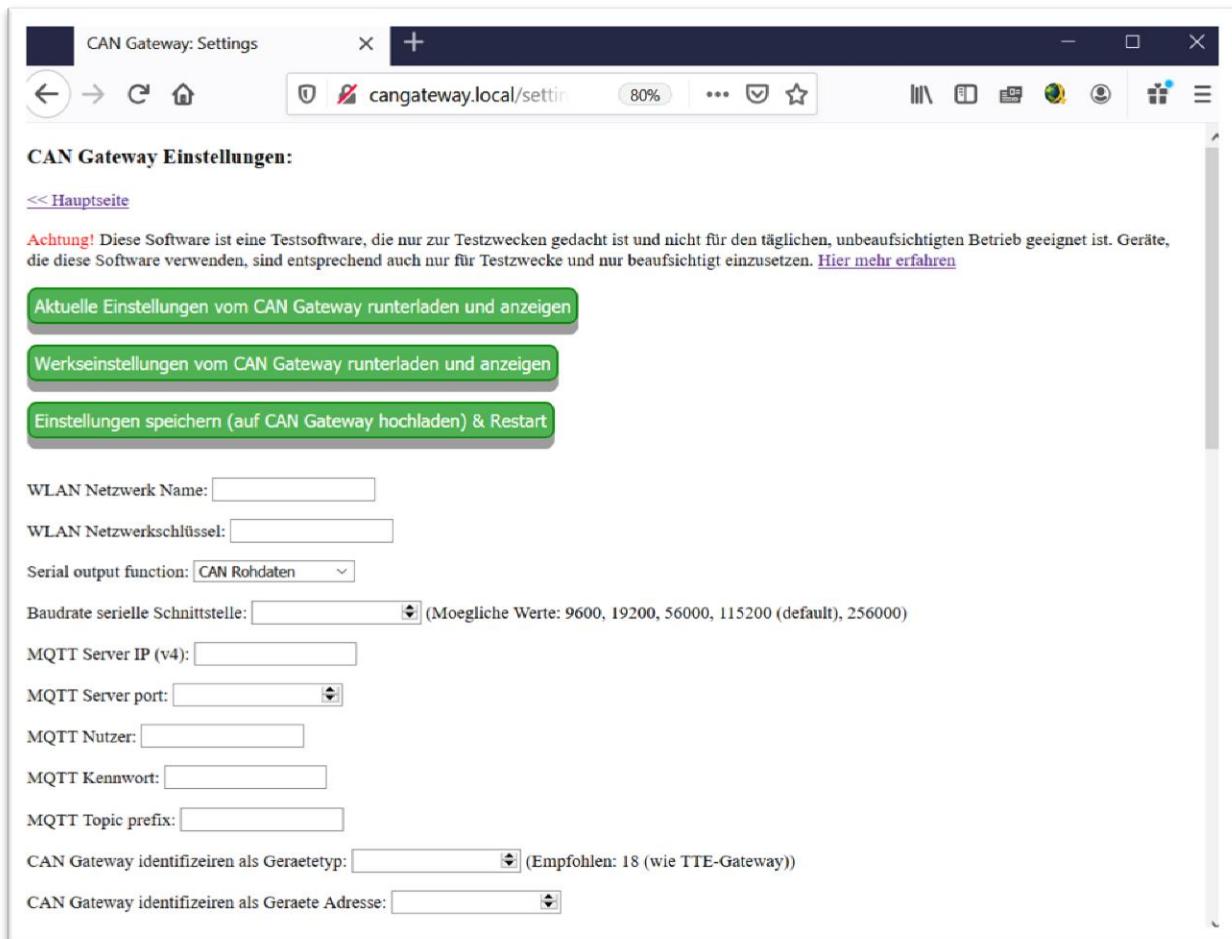
User-Levels

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.

WEB-Interface: Einstellungen

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

- WLAN Netzwerkname und Netzwerkschlüssel
- 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“.
- 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“).



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.

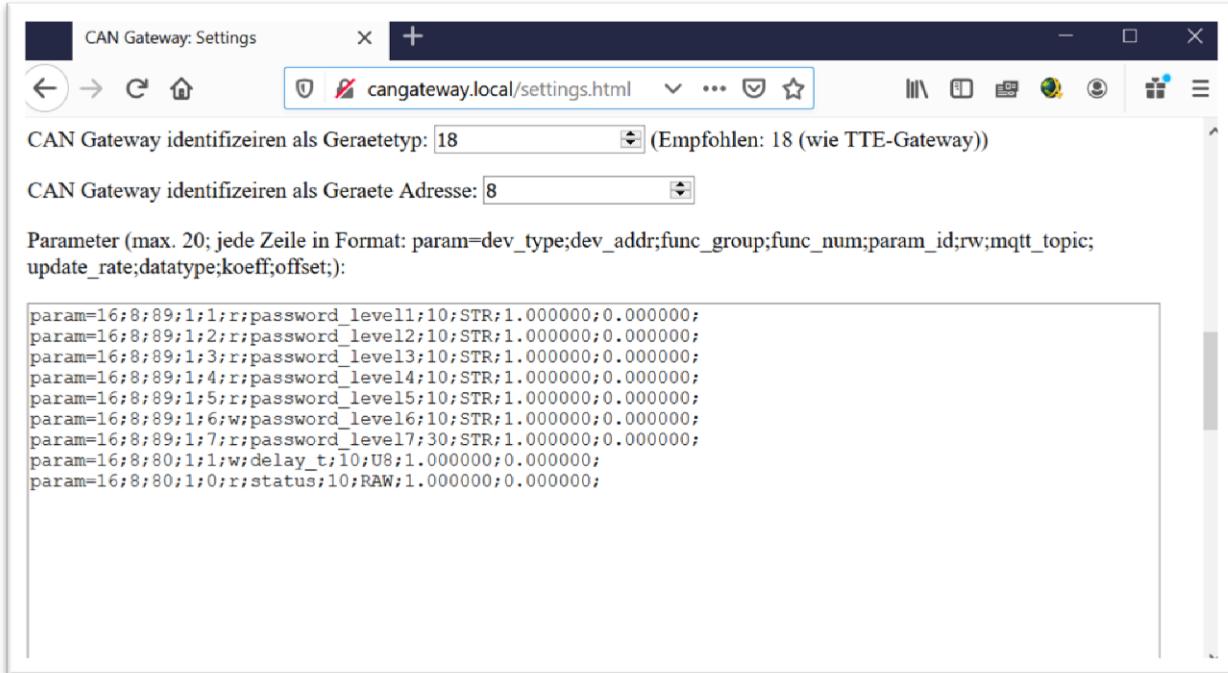
WEB-Interface: Parameter Einstellungen

Damit der CAN-Gateway die Parameter aus den Geräten ausliest/ändert und diese an Hausautomatisierungssystem überträgt, müssen die Parameter konfiguriert werden. Maximal können 20 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 (oftter 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 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 Geräte-Parametern“). Und als letztes wird ein Helligkeit/Status Parameter definiert (siehe Beispiel im Unterkapitel „Hinweise zu Listen“), allerdings als Rohdaten-Format (RAW, sinnvoller als Indes einer Liste wäre hier der Format „U8“).



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 Länger 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.

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.

Beispiele

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=deuchte_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;

Beispiel 2: Komfortlüftung – Temperatur Abluft

Folgende Zeile ist ein Beispiel, um die aktuelle Abluft Temperaturn auszulesen (Gerätetyp=8; Gerätadresse=8, Funkti0nsgruppe=50; Funktionsnummer=0; Datapoint-ID=37602; nur Lesen (r); MQTT topic=temp_abluft; Auslesen jede 10s; Datentyp: S16; Koeffizient=0.1; Offset=0):

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

Beispiel 3: Komfortlüftung – Temperatur Außenluft

Folgende Zeile ist ein Beispiel, um die aktuelle Außenluft Temperaturn auszulesen (Gerätetyp=8; Gerätadresse=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;
```

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; Gerätadresse=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;
```

Beispiel 5: Komfortlüftung – Betriebsmodus

Folgende Zeile ist ein Beispiel, um den Betriebsmodus auszulesen und zu ändern (Gerätetyp=8; Gerätadresse=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.

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	Gerätadresse	Funktionsgruppe	Funktionsnummer	Datenpunkt	Update Rate	R/W	Datentyp	Koeffizient	Offset
1	(0) TTE-WEZ (Wärmeerzeuger)	1	(2) Warmwasser	(0) Warmwasser 1	(118) Zirkulationskreistemperatur [gradC]	10	r	U16	0.1	0
2	(16) TTE-BM (Bedienmodul)	8	(89) Passwoerter	(1) Passwoerter	(3) Pin UserLevel 3	10	r	STR	1	0
3	not used	1				10				
4	not used	1				10				
5	not used	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

```
param=0:1:2:0:118;z:Zirkulationskreistemperatur_gradC:10;U16:0.1:0;
param=16:8:8:1:3;z:Pin UserLevel 3:10;STR:1:0;
```

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 web browser window titled "CAN Gateway: Parameterwerte". The address bar displays "cangatew". The page content is titled "CAN Gateway: Parameterwerte" and includes a link to "Hauptseite". A warning message states: "Achtung: diese Webseite verursacht hohe Prozessorlast für CANGateway und sollte nur für die Einrichtung und Debugging, jedoch nicht im normalen Betrieb genutzt werden." Below this is a green button labeled "Neu laden". The main area contains a table with 11 rows of parameter settings:

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

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 main content area is titled "CAN Gateway: Liste der gefundenen Geraete (alle seit Start)". Below this, there is a link "[<< Hauptseite](#)". A green button labeled "Update" is visible. A green button labeled "Liste leeren" is also present. The central part of the page is a table with the following columns: "Nummer", "Gerätetyp", "Gerätetyp (als dezimal)", and "Geräteadresse (dezimal)". The table has 10 rows, indexed from 1 to 10. Row 1 contains the entry: "1 TTE-BM (Bedienmodul) 16 8". All other rows (2-10) are currently empty.

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

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 and IP Address fields (both redacted), and several status messages: "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.", and "MQTT: verbunden seit 310 s.". Below this, a section titled "CAN-Bus Information:" lists task statistics: "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", and "decoder Task: [Dauer max; Dauer avg; Interval max; Interval avg] = [303; 56; 102513; 101011] us".

WEB-Interface: CAN Datenlogger

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

The screenshot shows a browser window titled "CAN Logger". The address bar displays "cangateway.local/logger.html". The main content area is titled "CAN Messages:" and contains a list of received CAN frames. A green button labeled "Als CSV Datei herunterladen" (Download as CSV file) is visible. The messages listed are:

```
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.

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...

[<< Hauptseite](#)

Als CSV Datei herunterladen Autoscroll

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

Alle Zahlen sind in HEX format!

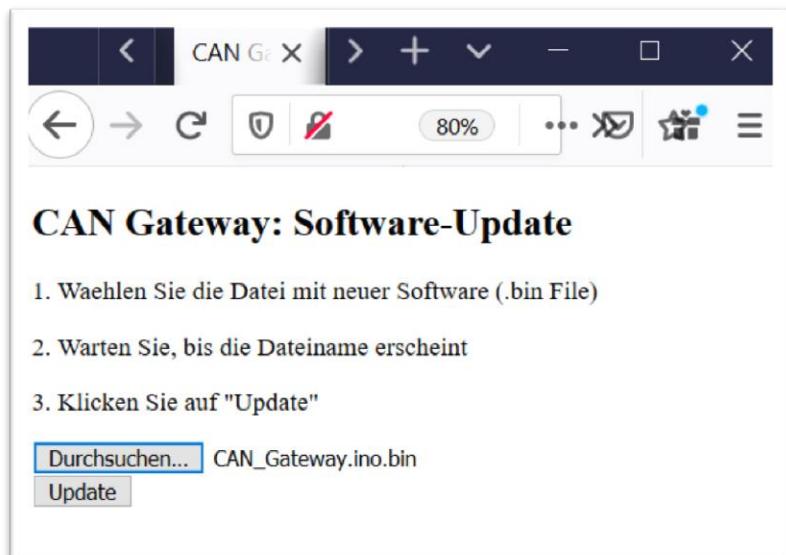
(10,08,59,01,0001) =	[REDACTED]
(10,08,59,01,0002) =	[REDACTED]
(10,08,59,01,0003) =	[REDACTED]
(10,08,59,01,0006) =	[REDACTED]
(10,08,50,01,0000) =	02
(10,08,59,01,0004) =	[REDACTED]
(10,08,59,01,0005) =	[REDACTED]
(10,08,50,01,0001) =	37

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

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.



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.

Übertragung der Parameter über Modbus TCP

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.

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.

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.

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.

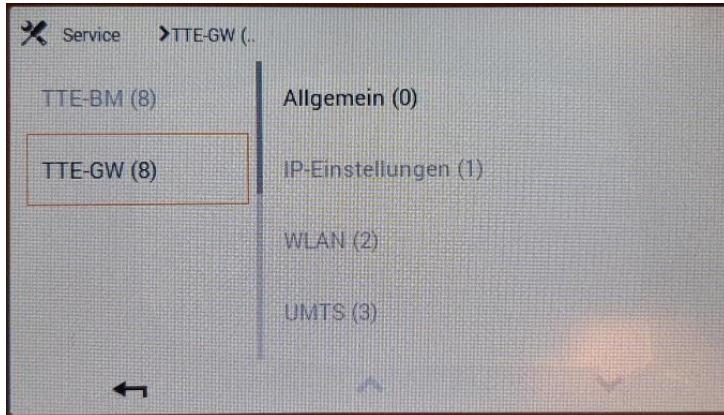
Parameter mit Datentyp U32 und S32 werden, wie oben beschrieben, jeweils in zwei Register übertragen. Sie müssen ggf. auf Ihrem Modbus Master die korrekte Byte- und Register-Reihenfolge einstellen.

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.

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“:

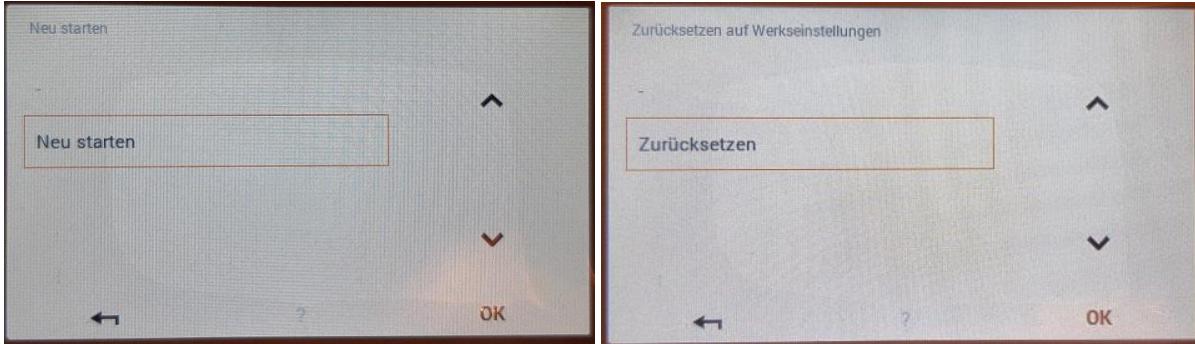


Durch Klicken auf **Allgemein(0)** → **Allgemein Gateway** → **Information (1)**, dann scrollen nach unten auf **Alle Datenpkt.**, dann Klicken auf die Liste der Datenpunkte rechts kann man auf die folgenden 5 Optionen zugreifen:

- 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”)

Linker Bildschirm (Alle Datenpkt.)	Rechter Bildschirm (Alle Datenpkt.)
Zurücksetzen auf Werkseinstellungen	Neu starten
Neu starten	Gateway-CAN-ID
Gateway-CAN-ID	IP-Adresse
IP-Adresse	Geräteidentifikation

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



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 20)
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 20) 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

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 Änderungsanträge 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.

Hilfe!

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 zwei 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) 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 (sieh 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 (sieh 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.

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.

Einschränkungen der Demo-Version

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

- Es kann maximal ein Parameter konfiguriert werden anstatt von 20.
- 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.

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.

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>)
- 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>)
- Async MQTT client for ESP8266 and ESP32 (Copyright (c) 2015 Marvin Roger, Licensed under The MIT License (MIT), <https://github.com/marvinroger/async-mqtt-client>)
- AsyncTCP Library (Copyright 2016 Hristo Gochkov, Licensed under GNU Lesser General Public License v3.0, Source: <https://github.com/me-no-dev/AsyncTCP>)
- 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>)
- 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>)