

http://



**Technisches Handbuch
myGEKKO & HTTP Befehle**

Version 1.3

27.02.2019

Softwareversion: ab V4795

Art. Nr. MRK.THB.HTT.0001



Änderungen

Datum	Dok. Version	Bearbeiter	myGEKKO Version	Änderungen
08.10.15	1.0		Ab V2975	Erstellung Handbuch
22.01.19	1.1	Marc Grass	Ab V4795	Überarbeitung Handbuch
05.02.19	1.2	Marc Grass	Ab V4795-10	GET Output, GET Input, POST Output und PUT-Output ergänzen
27.02.19	1.3	Marc Grass	Ab V4795-11	Leseintervall, Diagnosefenster und Erweiterung fürs Parsen von Zahlen aus Texten hinzugefügt

Inhaltsverzeichnis

Änderungen.....	2
1. Allgemeines zu myGEKKO & HTTP Befehle	4
1.1 Aufbau	4
1.2 Erklärung Variablen	4
2. Konfiguration von HTTP Befehle	5
2.1 GET-Output.....	7
2.2 POST-Output.....	8
2.3 PUT-Output.....	9
2.4 GET-Input.....	10
3. IO-Konfiguration Beispiel	11
3.1 Licht	11

1. Allgemeines zu myGEKKO & HTTP Befehle

Mit dem myGEKKO können über das Netzwerk frei definierbare HTTP Befehle gesendet und empfangen werden.

So können Geräte, welche über HTTP Befehle verfügen, über myGEKKO ausgelesen werden oder Werte geschrieben werden.

Die Befehle können in der IO Konfiguration definiert werden und in den jeweiligen Bausteinen verwendet werden.

Es gibt Lampen, Geräte (TV, Sprechstellen, Mehrfachsteckdosen, ...) und viele weitere Netzwerkfähige Geräte, welche mit HTTP Befehlen gesteuert werden können.

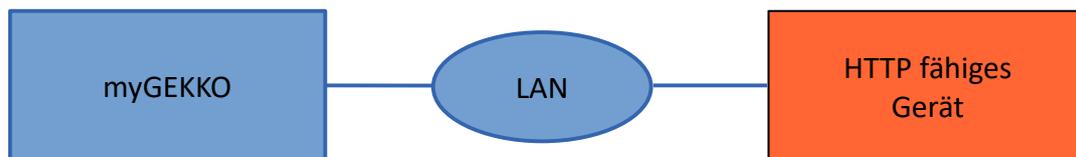
Somit haben Sie die Möglichkeit diese in myGEKKO einzubinden.

Hinweis: Um die HTTP-Befehle sinnvoll zu verwenden muss man sich ausführlich mit dem Thema und dem HTTP fähigen Gerät auseinandergesetzt haben.

1.1 Aufbau

Voraussetzung für die Verwendung der HTTP Befehle ist, dass das Netzwerk korrekt konfiguriert ist und die Geräte untereinander kommunizieren können.

Kontrollieren Sie hierzu gegebenenfalls die Netzwerkeinstellungen.



1.2 Erklärung Variablen

Platzhalter Variable ZAHL: @VALUE@

Der Platzhalter @VALUE@ wird verwendet bei:

- **AO:** Hier wird die Variable durch eine Ganze Zahl ersetzt (0,1,2,3,4, ...)
- **DO:** Hier wird die Variable durch 0 oder 1 ersetzt (bei >1 wird eine 1 ausgegeben)

Platzhalter Variable WORT: @VALUE:0[false],1[true]@

Dieser Platzhalter wird verwendet bei:

- **Wörtern:** @VALUE:0[false],1[true]@ Hier wird die Variable mit dem übergebenen Wert als AO oder DO als Text ersetzt.

Bsp.: Bei einer 0 wird false eingesetzt
Bei einer 1 wird true eingesetzt

Erweiterung Parsen von Zahlen aus einem Text: TEXT@Parameter=@VALUE@

Der Platzhalter @VALUE@ ersetzt die Zahl nach dem Parameter/Text:

Bsp.: Hka_Mw1.sWirkleistung=5,67
TEXT@Hka_Mw1.sWirkleistung=@VALUE@

2. Konfiguration von HTTP Befehle

Melden Sie sich als Konfigurator am myGEKKO an:

1. Gehen Sie in die Einstellungen (gelbes Zahnrad) und wechseln in die IO-Station Konfiguration (grün-blauer Pfeil)
2. Legen Sie eine neue IO-Station an mit dem Stationstyp HTTP
3. Über „Konfiguration >>“ können Sie die verschiedenen HTTP Befehle definieren

Es sind 4 verschiedene **HTTP Typen** verfügbar:

- **GET-Output**
- **POST-Output**
- **GET-Input**
- **PUT-Output**

4. Im Diagnosefenster werden Informationen und Statuscodes angezeigt
5. Hier kann der übergeordnete Leseintervall vergeben werden. Dieser kann zwischen 1 Sekunde und 65 Sekunden eingestellt werden.

Bsp.: Wenn der übergeordnete Leseintervall auf 1 Sek definiert wurde werden die Requests standardmäßig jede Sekunde ausgelesen. Wenn aber bei dem jeweiligen Request ein längerer Intervall definiert ist, wird dieser Request nur alle 5 Sekunden ausgelesen wie in dem nachfolgenden Beispiel der Dämmerung. Wurde der übergeordnete Leseintervall auf 10 Sek festgelegt, wird die Dämmerung auch nur jede 10 Sekunden ausgelesen.



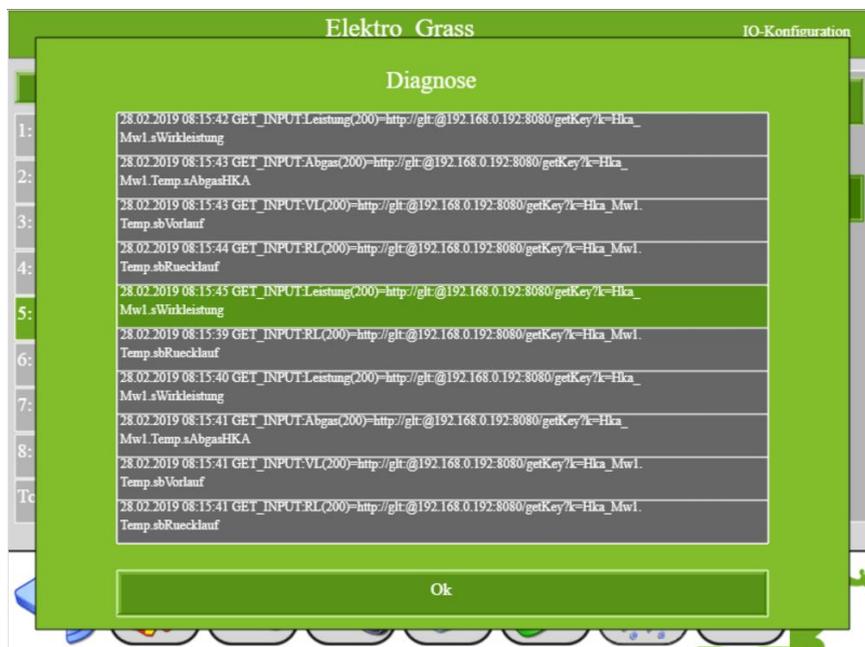
In der Konfiguration können Sie die Befehle definieren:

1. Vergeben Sie dem Befehl einen Namen, der Name muss immer als erstes eingegeben werden
2. Wählen Sie den Typ des HTTP-Befehl aus
3. Hier kann der Leseintervall je Request definiert werden bei Typ GET-INPUT
4. Definieren Sie den Request On
5. Definieren Sie den Request Off/Parameter
6. Speichern Sie Ihre Einstellungen und Befehle

Weitere Informationen zu den Requests finden Sie in den folgenden Kapiteln.



Im Diagnosefenster werden Informationen zu den Requests mit dem entsprechenden Statuscode angezeigt. Statuscode 200 bedeutet, dass der Request erfolgreich bearbeitet wurde und das Ergebnis der Anfrage in der Antwort übertragen wird. Weitere Statuscodes finden Sie im Internet.



2.1 GET-Output

Mit dem Typ GET-Output können mit dem HTTP GET Request Werte von myGEKKO gesendet werden.

Über GET lassen sich auch Formular-Daten übermitteln. Diese werden in codierter Form der URL angehängt. URL und Formular-Daten sind durch ein Fragezeichen (?) voneinander getrennt.

Die Requests werden nur bei Wertänderung gesendet.

Hinweis: Die maximale Länge von Name + Request On + Request Off/Parameter darf in der Summe nicht über 250 Zeichen betragen.
Pro Request beträgt die maximale Länge 127 Zeichen.

Es gibt 2 Anwendungsmöglichkeiten:

1. ON/OFF (Ein/Aus)

Request On:

<http://192.168.0.17/api/v1/var/lights/item0/scmd?value=1&username=www&password=www>

Request Off:

<http://192.168.0.17/api/v1/var/lights/item0/scmd?value=0&username=www&password=www>

2. VALUE (Wert)

Request On:

<http://192.168.0.17/api/v1/var/lights/item0/scmd?value=D@VALUE@&username=www&password=www>

Wenn kein Request Off eingetragen, so wird immer der Request ON ausgeführt.

@VALUE@ wird mit dem übergebenen Wert aus AO oder DO als Ganzzahl ersetzt:

z.B.:

Bei dem Wert 60 wird die Lampe Nr. 1 auf 60% Helligkeit gedimmt:

<http://192.168.0.17/api/v1/var/lights/item0/scmd?value=D60&username=www&password=www>

Genauso kann der Platzhalter für Wörter z.B. @VALUE:0[true],1[false]@ verwendet werden.

Hinweis: Im Request darf kein ; (Strichpunkt) eingegeben werden, da dieses verwendet wird um Parameter zu trennen.

2.2 POST-Output

Mit dem Typ POST-Output können mit dem HTTP POST Request Werte von myGEKKO gesendet werden.

Die POST-Methode funktioniert ähnlich wie die GET-Methode. POST wird jedoch zur Übermittlung von Formular-Daten an ein Programm oder Skript verwendet. Die Daten werden im Entity-Bereich getrennt durch eine Leerzeile vom Header übertragen.

Die Requests werden nur bei Wertänderung gesendet.

Hinweis: Die maximale Länge von Name + Request On + Request Off/Parameter darf in der Summe nicht über 250 Zeichen betragen.

Request On = URL:

Bsp: EBEE Ladestation

http://Benutzername:Passwort@IP-Adresse/operator/operator

http://operator:yellow_zone@192.168.2.36/operator/operator

Request OFF/Parameter: Stellt die zu übermittelnden Daten dar.

Diese werden fest eingestellt mit Header content-type:

application/x-www-form-urlencoded übermittelt

Bsp: EBEE Ladestation

OperatorCurrentLimit_vehicleif=@VALUE@&SUBMITTYPE=0d

@VALUE@ wird mit dem übergebenen Wert aus AO oder DO als Ganzzahl ersetzt

z.B.: Bei dem Wert 16 = OperatorCurrentLimit_vehicleif=16&SUBMITTYPE=0d

Bei dem Wert 9 = OperatorCurrentLimit_vehicleif=9&SUBMITTYPE=0d

@VALUE:1[true],0[false]@ wird mit dem übergebenen Wert als Wort ersetzt.

Hinweis: Im Request darf kein ; (Strichpunkt) eingegeben werden, da dieses verwendet wird um Parameter zu trennen.

2.3 PUT-Output

Mit dem Typ PUT-Output können mit dem HTTP PUT Request Werte von myGEKKO gesendet werden.

Diese Methode erlaubt das Erstellen oder Ändern von Dateien auf einem Webserver.

Die Requests werden nur bei Wertänderung gesendet.

Hinweis: Die maximale Länge von Name + Request On + Request Off/Parameter darf in der Summe nicht über 250 Zeichen betragen.

Request On = URL:

Bsp: Philips Hue

http://IP-Adresse Hue Bridge/api/Benutzername/lights/Lampennummer/state

Die grauen Felder müssen angepasst werden

http://192.168.2.49/api/ZqJswGAs76Pf7vwiD8MF5tnvn7bEA5knev-3N7JQ/lights/4/state

Request OFF/Parameter:

Stellt die zu übermittelnden Daten dar. Diese werden fest eingestellt mit Header content-type: application/x-www-form-urlencoded übermittelt

Bsp: Philips Hue

{„bri“:@VALUE@}

@VALUE@ wird mit dem übergebenen Wert aus AO oder DO als Ganzzahl ersetzt:

z.B.: Bei dem Wert 50 = {„bri“:50} wird die Lampe auf 50% gedimmt

{„on“:@VALUE:0[false],1[true]@}

@VALUE:1[true],0[false]@ wird mit dem übergebenen Wert als Wort ersetzt:

z.B.: Bei dem Wert 1 = {„on“:true} wird die Lampe eingeschalten

Bei dem Wert 0 = {„on“:false} wird die Lampe ausgeschalten

Hinweis: Im Request bzw. Parameter darf kein ; (Strichpunkt) eingegeben werden, dieses Zeichen wird in der Eingabe aktuell nicht unterstützt.

2.4 GET-Input

Mit dem Typ GET-Input können mit dem HTTP GET Request Werte aus einer HTTP API ausgelesen werden.

Es wird eine Ressource unter Angabe eines URL vom Server/Gerät angefordert.

Mit diesem Typ werden nur Daten abrufen, keine Auswirkungen wie Datenänderungen auf dem Server oder ausloggen.

Hinweis: Die maximale Länge von Name + Request On + Request Off/Parameter darf in der Summe nicht über 250 Zeichen betragen.

Die maximale Länge der Antwort beträgt 1000 Zeichen.

Request ON = URL:

Bsp:

myGEKKO API:

`http://192.168.2.30/api/v1/meteo/status?username=www&password=www`

Dachs Senertec:

`http://glt:@192.168.0.192:8080/getKey?k=Hka_Mw1.sWirkleistung`

Request OFF/Parameter:

- Ist dieses Feld leer, so nimmt Gekko an, die Antwort ist bereits eine Zahl und wandelt diese entsprechend um und kann in einem AI genutzt werden.
- TEXT@conn_state -> Wertet z.B. aus ob die Antwort aus dem Text conn_state besteht und setzt dementsprechend den Wert auf 1 bzw. auf 0.
- JSON@twilight.status -> Nimmt als Antwort ein JSON Objekt und sucht in diesem Objekt nach dem Feld „twilight.status“ und wandelt diesen Wert um, sodass dieser als AI verwendet werden kann. Gültig sind sämtliche numerischen Werte sowie auch True/False.
- JSON@status.effect@none -> Nimmt als Antwort ein JSON Objekt und sucht in diesem Objekt nach dem Feld „status.effect“. Ist der Wert dieses Objektes „none“ so wird das Ergebnis auf 1 ansonsten auf 0 gesetzt.
- TEXT@Hka_Mw1.sWirkleistung=@VALUE@ -> Wertet die Zahl nach dem Parameter „Hka_Mw1.sWirkleistung=“ aus und wandelt diese entsprechend um und kann in einem AI genutzt werden. Bsp.: Hka_Mw1.sWirkleistung=5,67

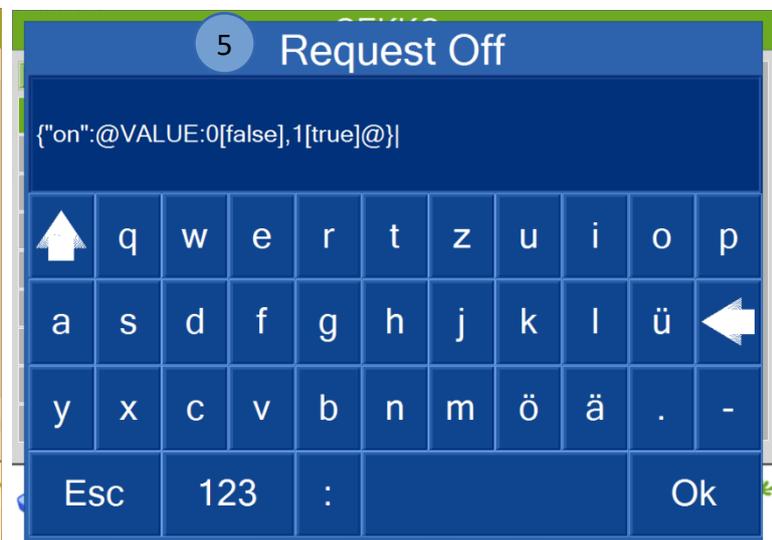
Hinweis: Im Request bzw. Parameter darf kein ; (Strichpunkt) eingegeben werden, dieses Zeichen wird in der Eingabe aktuell nicht unterstützt.

3. IO-Konfiguration Beispiel

3.1 Licht

In diesem Beispiel wird beschrieben wie Sie eine Hue-Lampe ein- und ausschalten können:

1. Melden Sie sich als Konfigurator an, gehen Sie in die Einstellungen ->IO-Stationen und konfigurieren eine neue IO-Station mit dem Typ HTTP ein. Tippen Sie anschließend auf „Konfiguration>>“
2. Vergeben Sie dem Befehl einen Namen, dieser muss immer als erstes eingegeben werden
3. Wählen Sie den HTTP Typ aus. In unserem Fall für die Hue Lampen den Typ PUT-OUTPUT
4. Geben Sie den Request ON ein. Der Request ist Gerätespezifisch und muss auf Ihr Gerät angepasst werden
5. Geben Sie den Request OFF/Parameter ein. Der Request ist Gerätespezifisch und muss auf Ihr Gerät angepasst werden
6. Speichern Sie die Befehle ab



Gehen Sie in das System Licht, legen einen neuen Lichtbaustein an und tippen auf „Ausgänge/Ansteuerungen“

7. Vergeben Sie einen Schalt-Ausgang
8. Wählen Sie dazu die HTTP IO-Station aus, anschließend tippen Sie auf Ihren konfigurierten Befehl (Device) und senden den PUT Befehl. Wenn Ihr Gerät nicht angezeigt wird tippen Sie auf den „LOAD“ Button

Nun können Sie Ihre Hue-Lampen ein- und ausschalten.

Diese Konfiguration dient zur Veranschaulichung der HTTP-Befehle und kann auf andere Geräte angepasst werden.

The image shows two screenshots from the myGEKKO interface. The top screenshot is titled 'Küche' and 'IO-Konfiguration'. It displays a table for 'Ausgänge/Ansteuerungen' with a blue circle '7' around the 'Ausgänge/Ansteuerungen' header. The table has columns for 'Schalt-Ausgang', 'Schalt-Rückmeldung', 'Lichtsystem', 'Dim-Ausgang 1 (R)', 'Dim-Ausgang 2 (G)', and 'Dim-Ausgang 3 (B)'. The 'Schalt-Ausgang' row shows '1.1_1', 'Aus', 'Typ', 'Normal', and '60.00W'. The 'Schalt-Rückmeldung' row shows 'DI', 'Aus', and 'Wiederholung Aus'. The 'Lichtsystem' row shows 'Standard' and 'Abgleich..'. The 'Dim-Ausgang' rows show 'AO', '0%', and '0..10V'. A blue arrow points to the left, and a green lizard is visible in the bottom right corner.

The bottom screenshot is titled 'Schalt-Ausgang' and shows a configuration screen with a blue circle '8' around the 'STATION' field. The 'STATION' field contains '1:HTTP'. The 'DEVICE' field contains '01:/O Light'. The 'LOAD' button is highlighted with a blue circle. The 'REQUEST' field contains 'SEND->PUT'. At the bottom, there are buttons for 'ONBOARD MEMORY', 'NC', 'C', 'Esc', and 'Ok'.

myGEKKO ist ein Produkt, gewachsen aus langjähriger Erfahrung und Entwicklung in Europa – mit Partnern in Ihrer Nähe.



Italien



Deutschland



Schweiz



Österreich

www.my-gekko.com



A first class product of Europe!

The result of a close collaboration between Italy, Switzerland and Germany

