

# Handleiding Brel motoren – Home assistant

## Aandachtspunten

- Deze tutorial werkt alleen met bi-directionele motoren. Mono-directioneel kunnen geen motorstanden aannemen en kunnen alleen open of dicht.
- Deze tutorial is alleen getest met Jaloezie motoren maar zou ook met rolgordijnen en gordijnmotoren moeten werken (bi-directioneel)
- Heb onderstaande tutorial alleen getest met de hub DD7002B

## Downloaden van de benodigde software

Voordat we de brel hub kunnen aansluiten op Home Assistant hebben we eerst wat software nodig. In onderstaande link kan de software worden gedownload die nodig is om zowel de token als de motor mac adressen te bemachtigen. Deze zijn later nodig om de motoren uit te lezen en te besturen. Download alle bestanden in de link naar de pc.

<https://github.com/alexbacchin/ConnectorBridge/tree/master/doc/Wi-Fi%20Local%20Access>

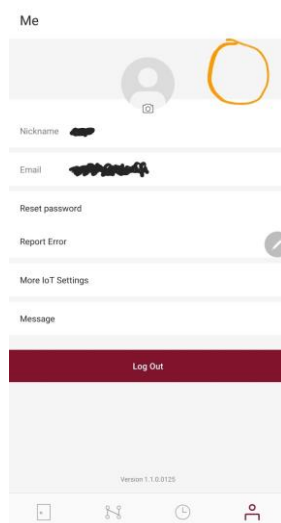
## De Brel app

Voordat we nu verder gaan hebben we eerst de brel app nodig. Met deze app kunnen we de bridge DD7002B eerst verbinden met het wifi netwerk. Tevens kunnen we uit de brel app de sleutel halen die nodig is om het accesstoken te maken.

Installeer de brel app op een android of apple apparaat en volg de instructies om de hub succesvol aan te sluiten op je wifi netwerk.

Voeg vervolgens indien nodig alle motoren toe aan de hub en zorg dat deze allemaal naar behoren werken met de app alvorens je verder gaat.

Wanneer alle motoren werken heb je de app nog voor een ander belangrijk punt nodig. Namelijk de key. Deze key kun je bemachtigen door in de app te gaan naar je profiel (knop rechts onder) en tik vervolgens 5 keer op het scherm rechts van je profiel foto. (aangegeven met de oranje cirkel hieronder)



Je krijgt nu een pop up met “reminder”. Schrijf deze code op en druk op cancel. (let op deze code is hoofdlettergevoelig)

Je bent nu klaar met de app.

### Accesstoken aanmaken

Nu we de key hebben en alles verder hebben ingesteld openen we op een pc het gedownloadde programma uit de link genaamd “EVAL tool”.

LET OP! Het is essentieel dat je met de pc waarop je dit programma draait op exact hetzelfde netwerk zit als de brel hub. Wanneer er een repeater of wifi extender tussen de pc en de hub zit dan is de kans erg groot dat deze tool niet werkt! Zorg dus dat beide op exact hetzelfde netwerk zitten voordat je begint.

Eenmaal geopend druk je op de start knop in het programma.

Vervolgens vul je bij “please enter key” de code in die je met 5 keer tikken uit de brel app gehaald hebt. En druk op get device list. Wanneer alles goed is ingesteld krijg je in de lijst onder start eerst de hub te zien en vervolgens alle motoren die aan deze hub gekoppeld zijn.

Onder in de log zou je nu onder andere ook de token moeten zien staan. Kopieer deze token.

### Accesstoken aanmaken

Vervolgens is de volgende stap om de accesstoken te maken. Let op! De token is NIET de accesstoken!

Open het zip bestand “Accesstoken” en pak alles uit. Open vervolgens AccesstokenCal.exe. (als je een bedreigingsmelding krijgt in windows, klik dan op meer informatie en vervolgens op “toch uitvoeren”).

Vul in de bovenste regel de Token in en daaronder de key uit de brel app en druk op “calculation”.

In de regel AccessToken komt nu een code te staan. Om te voorkomen dat je dit ooit weer opnieuw moet doen is het raadzaam deze code ergens op te slaan. Deze code zal namelijk met iedere commando naar de hub/motoren meegestuurd moeten worden.

### Motor macadressen

Naast de token zijn ook de mac-adressen van de motoren nodig. Schrijf deze ook op. Deze vind je in de lijst onder “mac” in de EVAL tool waarbij de bovenste regel de hub is. Deze heb je niet nodig. Dit programma is tevens ook handig om te testen welke motor bij welk mac adres hoort. Rechts van de lijst vindt je namelijk “motor control” Als je de juiste key hebt ingevuld zou je nu de motoren vanuit het programma moeten kunnen besturen door in de lijst op de betreffende motor te klikken en vervolgens rechts de sliders te verschuiven. Zo kun je ook gelijk zien welke motor bij welk mac adres hoort. Schrijf dit op.

### Home assistant integratie

Met al de gegevens die je nu hebt kun je deze in Home Assistant gaan gebruiken om daadwerkelijk de motoren te gaan besturen. In deze tutorial gebruiken we daar Node-red voor. Maar voordat we zover zijn maken we in home assistant eerst enkele helpers (sliders) aan. Maak voor iedere motor een slider aan en vul hierbij de waardes in van 0 tot 100.

Dit zou er ongeveer zo uit moeten zien:

✕ jaloerie links
☰

INSTELLINGEN
GERELATEERD

---

Naam  
**jaloerie links**

Pictogram  
**mdi:blinds**

Minimale waarde  
**0**

Maximale waarde  
**100**

Weergavemodus       Schuifregelaar     Invoerveld

Stapgrootte  
**1**

Meeteenheid

Entiteits-ID  
**input\_number.jaloerie\_links**

Gebied ▼

**Schakel entiteit in**  
 Uitgeschakelde entiteiten zullen niet aan Home Assistant worden toegevoegd  
Opmerking: dit werkt mogelijk nog niet met alle integraties.

VERWIJDEREN
BIJWERKEN

Doe dit voor iedere motor.

Indien node-red niet geïnstalleerd is op Home Assistant dan moet dit eerst worden gedaan. Indien node-red al geïnstalleerd is, start noder-ed dan opnieuw op na het aanmaken van de helpers/sliders

Vervolgens gaan we naar node-red.

### Node-Red

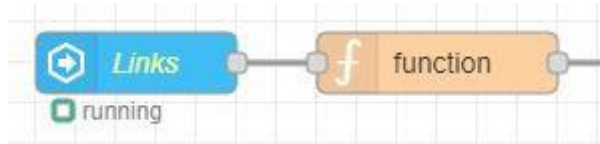
In node red is het nu belangrijk dat we gaan instellen dat zodra we de slider verschuiven de jaloezieën open/ dicht gaan naar gelang het percentage die we kiezen. Dit doen we door te starten met een Event state node.



Geef deze node de naam van de jaloezie zodat je straks makkelijk terug kan zien om welke motor het gaat.

Bij entity ID zoeken we nu naar de helper/slider die we hebben aangemaakt. In ons voorbeeld komt hier te staan: `input_number.jaloezie_links`. Voor de rest hoeven we hier niets meer in te stellen.

Iedere keer als nu de slider in home assistant wordt verschoven dan wordt deze waarde direct aan node-red doorgegeven en wordt verstuurd in de `msg.payload` output. Achter de event state node zetten we een tweede node genaamd "function node"

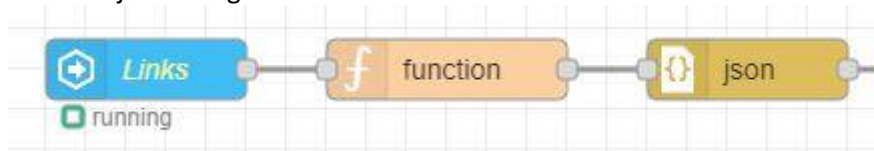


In deze function node kunnen we de vertaal slag gaan maken. Hierbij gaan we het cijfer wat door home assistant wordt doorgegeven eerst strippen van de decimalen achter de komma en vervolgens voegen we deze toe aan de code die we gaan versturen aan de hub. Zonder al te ver in te treden op de details zou de code in de function node er zo uit moeten zien:

```
var sermsg = {payload: msg.payload};
var tijd = Date.now();
msg.payload = parseFloat(msg.payload).toFixed(0)
sermsg.payload = { "msgType": "WriteDevice", "mac": "2462abbab7f00006", "deviceType":
"10000000", "AccessToken": "796076329747594390B46CC3EH843F5", "msgID": " + tijd + ", "data": {
"targetPosition": + msg.payload } } };
return sermsg;
```

Hierbij is het belangrijk dat enkele gegevens worden aangepast. Ieder Accesstoken en mac-adres zijn immers anders. Pas hierbij de gegevens aan die in groen zijn aangegeven naar het mac-adres en accesstoken die je zelf hebt opgehaald eerder in deze tutorial.

Vervolgens komt er achter deze node een json node die de zojuist aangemaakte code omzet in een leesbare json string voor de hub.



Hier stel je alleen in dat de action: "Always convert to JSON String" wordt. Je kunt dit ook rechtstreeks in de function node regelen maar laten we voor het gemak even uitgaan van deze opstelling.

Als laatste hebben we een UDP multicast out node nodig.



Deze node zal daadwerkelijk de `msg.payload` die we hebben aangemaakt doorsturen over het multicast profiel op het netwerk waarna de Brel hub dit oppikt en het betreffende commando verstuurd naar de motor.

Vul de node op deze manier in. En zorg er alleen voor dat "Address" het ip adres van jou Brel hub is. Dit is dus waarschijnlijk een andere IP dan in het voorbeeld hieronder.

### Edit udp out node

Delete
Cancel
Done

⚙️ **Properties**

⚙️
📄
🖼️

✉️ Send A udp message ▼ to port 32100

📄 Address 192.168.178.55 ipv4 ▼

bind to random local port ▼

Decode Base64 Encoded Payload?

📌 Name Name

Tip: leave address and port blank if you want to set using `msg.ip` and `msg.port`.

Druk op done en vervolgens rechtsboven op deploy.

Als alles goed is ingesteld dan zou je nu met de slider in Home Assistant de jaloezie moeten kunnen bedienen.

### Voorbeeld automation in Home Assistant

Indien je de jaloezieën automatisch in bepaalde standen wil zetten met automations in Home Assistant is het niet moeilijker dan met de service: `input_number.set_value` de waarde van de slider aan te passen. Hieronder een voorbeeld van een werkende automation die de jaloezie sluit wanneer de tv aangezet wordt. Hierin moet uiteraard wel het een en ander worden aangepast om het in jou geval werkend te maken:

```

- id: '162355667625231'
  alias: Jaloezie links dicht bij film kijken
  description: ''
  trigger:
  - platform: state
    entity_id: media_player.je_tv_hier
    to: 'on'
  condition: []
  action:
  - service: input_number.set_value
    target:
      entity_id: input_number.jaloezie_links
    data:
      value: 0
  
```

mode: single

### Meer mogelijkheden

Wanneer je dit succesvol hebt afgerond dan kun je nog een stap verder gaan. In de bestanden die je hebt gedownload zit ook nog een pdf bestand (Connector WLAN Integration Protocol.pdf) die alle mogelijk commando's uitlegt die je kunt versturen of juist kunt oppikken met de UDP in node. Zo is het ook mogelijk om na een commando bijvoorbeeld gelijk het voltage op te pikken uit de motor zodat je met een formule kunt bepalen hoeveel procent lading de accu in de motor nog heeft. En deze in een sensor weer terug laat sturen naar Home Assistant.

Voor vragen of verdere uitleg kun je contact opnemen met mij via Discord. Voeg me toe

John H.#7496

Ik kan eventueel ook de voorbeeld flows van node-red toesturen.