



# Ontwikkeling Venus-E thuisbatterij.

## 1 Inleiding.

Dit document beschrijft de ontwikkelingen m.b.t. de Marstek Venus-E. De bedoeling is om te komen tot een stabiele en verbeterde regeling van de batterij.

De huidige versie maakt gebruik van een Marstek CT003 P1-dongle. Deze lijkt niet erg stabiel. Bovendien lijkt de interne regeling v.w.b. "nul op de meter" (NOM) niet overtuigend te werken. De software van de batterij, momenteel versie 147 is niet open-source, waardoor je afhankelijk bent van de leverancier.

Plannen voor de verbeterde versie zijn:

- Open source software (PlatformIO, ESP32, Arduino).
- Goede documentatie.
- Sturing via ModBus.
- Vrije keus van P1-dongle.
- Meer dan één model voor de regeling van laden/ontladen.
- Simulatie van energiemeter om model te testen/
- Data beschikbaar stellen via MQTT.
- Beschikbaar stellen van een Web-interface.
- Zelfbouw mogelijk maken.
- Print ontwerpen.

Ik weet dat er mogelijkheden zijn om Home Automation (HA) te gebruiken voor de sturing van de batterij, maar dit project is gericht op een zelfstandige plug-en-play oplossing.

## 2 RS485 to USB converter module.

Om de ModBus interface met de batterij te testen is tijdelijk gebruik gemaakt van een USB module:



Deze module wordt gebruikt voor de RS485 verbinding voor Modbus RTU.

De USB module vormt een seriële COM poort. Er is een gratis testprogramma "mbpoll" beschikbaar dat gebruikt kan worden om PLC/kWh-meter registers te lezen. Voorbeeld van gebruik hieronder bij gebruik in Linux. Er is ook een Windows-versie.

**\$ mbpoll -a 1 -b 115200 -d 8 -P none -s 1 -t 4 -r 32200 -c 1 -1 -0 -B /dev/ttyUSB0 -v**

-t 4 is 2-byte integer (holding register)

-t 4:float is een 4-byte float

-t 4:int is een 32 bit integer

Als /dev/ttyUSB0 niet toegankelijk is als non-root user:

\$ sudo usermod -a -G dialout \$USER

**De output is:**

```
mdebug enabled
Set mode to RTU for serial port
Set device=/dev/ttyUSB0
mbpoll 1.0-0 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright © 2015-2019 Pascal JEAN, https://github.com/epsilonrt/mbpoll
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'mbpoll -w' for details.

Opening /dev/ttyUSB0 at 115200 bauds (N, 8, 1)
Set response timeout to 1 sec, 0 us
Protocol configuration: Modbus RTU
Slave configuration...: address = [1]
                      start reference = 32200, count = 1
Communication.....: /dev/ttyUSB0,      115200-8N1
                      t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

-- Polling slave 1...
[01][03][7D][C8][00][01][1D][98]
Waiting for a confirmation...
<01><03><02><09><37><FF><C2>
[32200]:      2359
```

Het polling commando heeft de volgende lay-out:

<01>	Slave adres van de Venus	Altijd 1
<03>	Functie	Lees holding register
<7D>	Start regiset HI	In dit voorbeeld 32200
<C8>	Start register LO	
<00>	Aantal regiters HI	In dit voorbeeld 1
<01>	Aantal registers LO	
<1D>	CRC HI	
<98>	CRC LO	

### 3 ModBus registers in de Venus.

De volgende registers zijn beschikbaar (alleen de handige functies voorlopig):

Adres	Omschrijving	Datatype	Int.code	Opmerkingen
31000	Device name[20]	u16 * 10		Ascii string, "BI_2.5_2.5"
32104	State of Charge [%]	u16	SOC	
32000	Battery voltage[V*100]	u16		Werkt niet..., geeft 0.02 Volt
32200	AC voltage[V*10]	u16		
32201	AC current[A*100]	u16		absolute waarde
32202	AC power [W]	s32	ACPW	
33000	Total charging energy [kWh*100]	u16		
33002	Total discharging energy [kWh*100]	u16		
35000	Internal temperature[°C*10]	s16		
35001	MOS1 temperature[°C*10]	s16		
35002	MOS2 temperature[°C*10]	s16		
42000	RS485 Control mode select	u16		#55AA = enable, #55BB = disable
42010	Forcible charge/discharge	u16		Stop/Charge/Discharge is 0/1/2. Leest "0"
42011	Charge to SOC [%]	u16		12...100
42200	Back-up function	u16		Disable = 1, enable = 0
42020	Forcible charge power [W]	u16	CHP	0..2500
42021	Forcible discharge power [W]	u16	DCHP	0..2500
43000	User work mode	u16	WMOD	0=manual, 1=ant-feed, 2=ai
44000	Charging cutoff capacity [%*10]	u16		80..100
44001	Discharging cut-off capacity [%*10]	u16		12..30
44002	Max charge power	u16	CHMAX	0..2500
44003	Max discharge power	u16	CHMIN	0..2500

## 4 Bruikbare dongles.

Het streven is om veel P1 dongles te gaan ondersteunen. Voorwaarde is dat de dongle via WiFi bereikbaar is en een API heeft. Voor de test gebruik ik een "P1-Dongle-Pro. Zie <https://smart-stuff.nl/product/p1-dongel-slimme-meter-esp32>.

### 4.1 P1-Dongle-Pro.

De dongle (P1-Dongle-Pro) geeft op <http://192.168.1.172/api/v2/sm/actual> een JSON structuur terug. Dat geeft tevens een eenvoudige testmogelijkheid. Belangrijke velden zijn:

power\_delivered en power\_returned. Dat zijn floats in kW. Voorbeeld van de hele JSON:

```
{"timestamp": {"value": "250207122727W"},  
 "energy_delivered_tariff1": {"value": 6071.137, "unit": "kWh"},  
 "energy_delivered_tariff2": {"value": 5551.723, "unit": "kWh"},  
 "energy_returned_tariff1": {"value": 1735.003, "unit": "kWh"},  
 "energy_returned_tariff2": {"value": 3571.852, "unit": "kWh"},  
 "electricity_tariff": {"value": "0002"},  
 "power_delivered": {"value": 2.729, "unit": "kW"},  
 "power_returned": {"value": 0, "unit": "kW"},  
 "voltage_l1": {"value": 234, "unit": "V"},  
 "current_l1": {"value": 11, "unit": "A"},  
 "power_delivered_l1": {"value": 2.729, "unit": "kW"},  
 "power_returned_l1": {"value": 0, "unit": "kW"},  
 "gas_delivered": {"value": 6194.451, "unit": "m3"},  
 "gas_delivered_timestamp": {"value": "250207122500W"}  
}
```

De timestamp wordt gebruikt om de klok van de ESP32 gelijk te zetten.

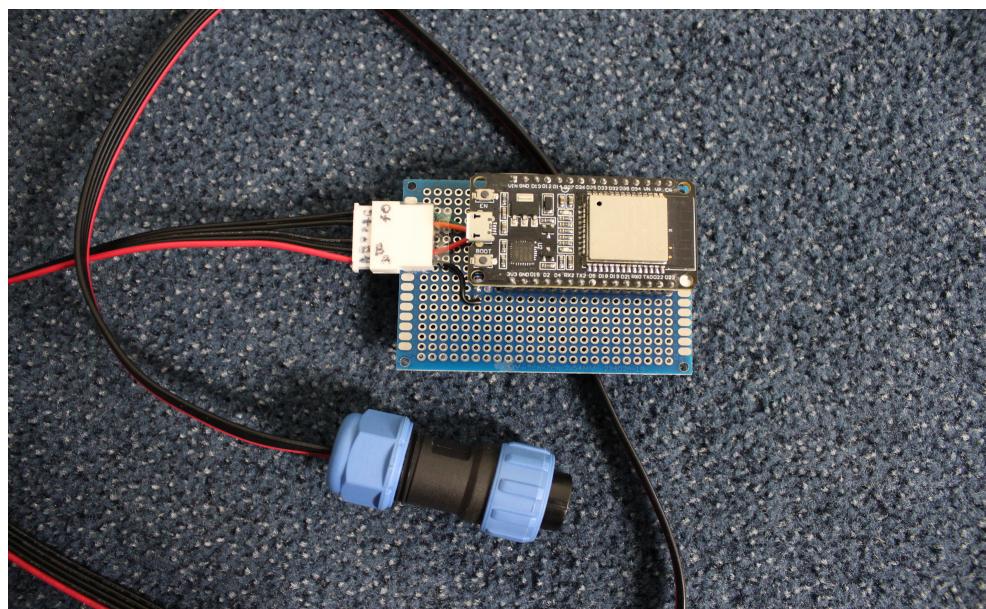
Soms geeft het uitlezen van de dongle een fout. Dat gebeurd als er gelijktijdig een ander apparaat de dongle probeert uit te lezen. In zo'n geval wordt de uitlezing genegeerd.

### 4.2 P1\_Simul.

Dit is een gesimuleerde dongle. Er wordt een energiemeter gesimuleerd met een zich herhalend patroon van energie verbruik en energie opwekking. Elke 40 seconden wordt een stap gedaan. Het patroon is: 1000W, 800W, -2000W, 400W. Eeen negatieve waarde betekent hier opwekking.

## 5 Prototype.

Voor het testen van de software is een prototype van de controller gemaakt. Hij ziet er zo uit:



De max485 bevindt zich onder de ESP32. De print wordt gevoed vanuit de Venus.

### 5.1 Testen met prototype.

Met de huidige software (18-02-2025) werkt het volgende:

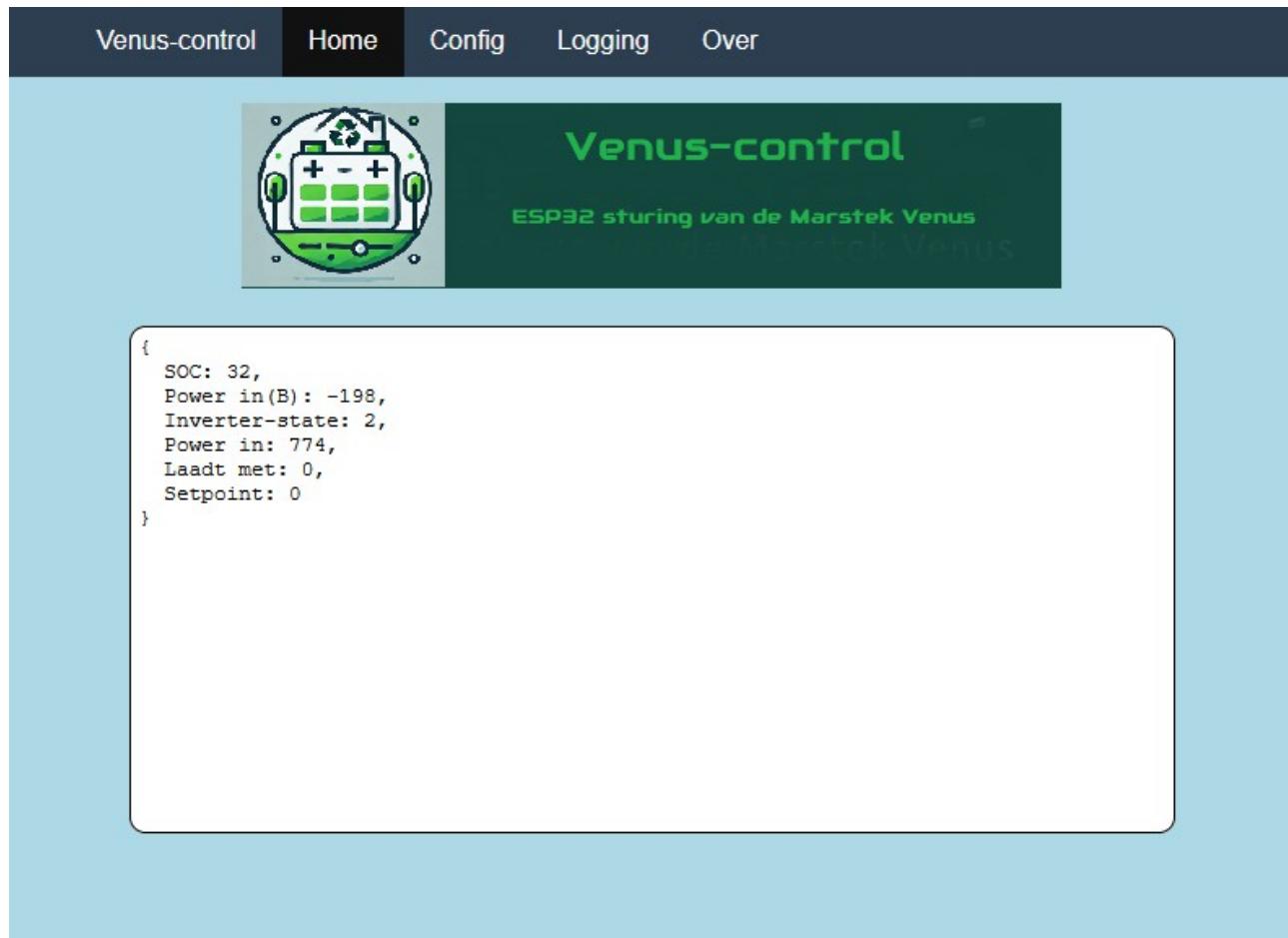
- Kan data van de P1-dongle ophalen. Nu is dat één keer per 5 seconden.
- Kan registers vanuit de Venus inlezen via de ModBus.
- Web-interface werkt.
- Kan procesdata presenteren in web-interface.

## 6 Web-interface.

Hier volgt een beschrijving van de diverse pagina's van de web-interface. Boven in beeld een menu, waarmee de gewenste pagina kan worden getoond.

### 6.1 Home page.

De home page ziet er als volgt uit:



```
{  
  SOC: 32,  
  Power in(B): -198,  
  Inverter-state: 2,  
  Power in: 774,  
  Laadt met: 0,  
  Setpoint: 0  
}
```

In de huidige versie worden enkele gegevens op een primitieve manier getoond. Dat wordt later mooier. De gegevens kloppen wel:

- SOC is de “state of charge” van de batterij. In dit geval is dat 32 procent.
- Power in(B) is het huidige ontlaad vermogen van de batterij. In dit geval wordt er met 198 W opgeladen.
- Inverter-state is hier “2”, dat betekent “charge”. 0=sleep, 1=stand-by, 2=charge, 3=discharge, 4=back-up mode, 5-OTA upgrade.
- Power in is de huidige netto levering in Watt door het energiebedrijf. Bij negatief wordt er teruggegeven aan het net. Dit is afkomstig van de P1 dongle.
- Setpoint is de (berekende) instelling in Watt dat de batterij moet gaan op- of ontladen om nul op de meter te krijgen.

## 6.2 Config page.

De configuratie-pagina ziet er als volgt uit:

The screenshot shows a web-based configuration interface for an ESP32. At the top, a navigation bar includes links for 'Venus-control', 'Home', 'Config' (which is the active tab), 'Logging', and 'Over'. Below the navigation bar, a message in Dutch reads: 'Je kunt hier de configuratie wijzigen.' and 'Let op: sommige wijzigingen hebben pas effect na herstart.' A code block displays the current configuration parameters:

```
dongle = P1_Dongle_Pro
dongle_api = /api/v2/sm/actual
dongle_host = 192.168.1.172
lstmods = Tue, 18 Feb 2025 08:46:43 GMT
updhost = venus.smallenburg.nl
venus_mb_adr = 1
wifi_0 = Nokia-basis/*****
wifi_1 = ADSI/*****
```

At the bottom of the configuration area, there are three buttons: 'Opslaan' (Save), 'Herstart' (Restart), and 'Update software'. Below these buttons, a progress bar is shown with the text 'Wacht op input....' (Waiting for input....).

Op deze pagina kan men parameters wijzigen om de ESP32 te configureren. Je ziet in de eerste 3 regels de configuratie voor de gebruikte dongle.

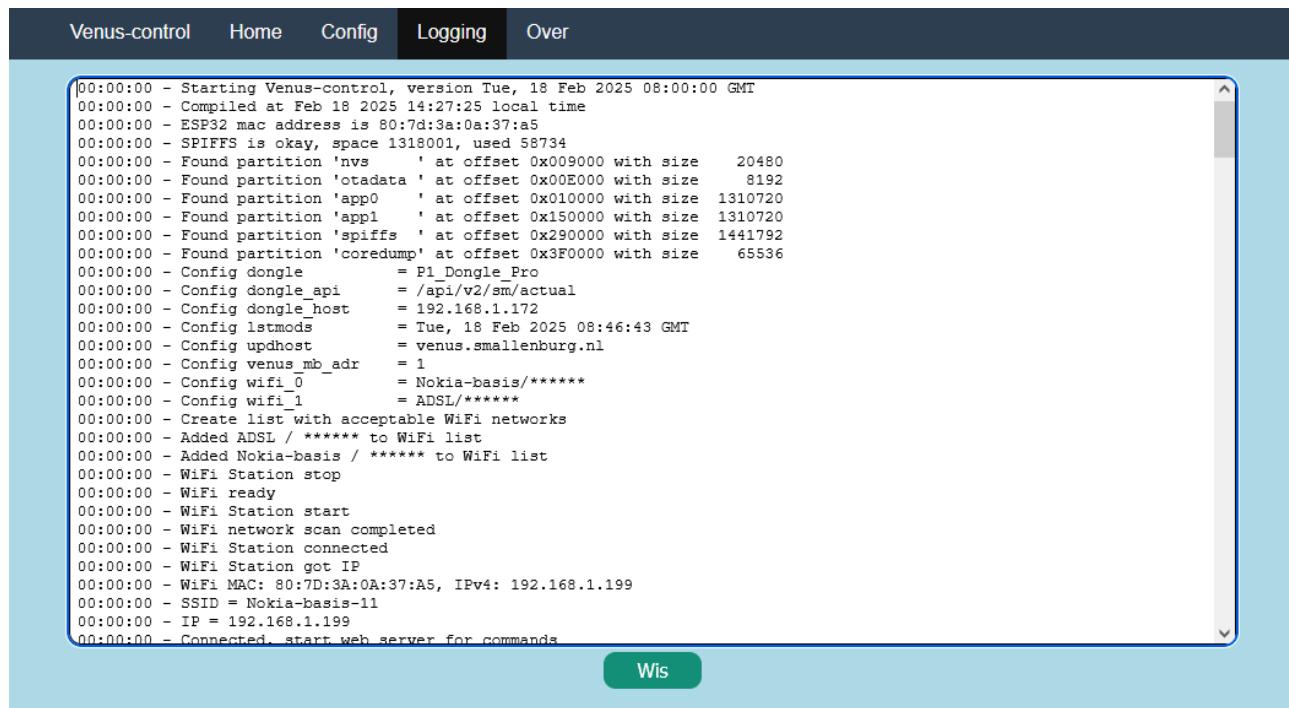
lstmods geeft aan wat de huidige versie van de software is. Met de knop "Update software" kan een nieuwe versie worden geladen van een externe server (updhost). Verder zie je het ModBus adres van de Venus en (in dit geval) 2 WiFi netwerken met hun passwords. Bij opstarten wordt het WiFi net met het sterkste signaal gebruikt.

Met de "Opslaan"-knop worden gewijzigde instellingen opgeslagen en met "Herstart" wordt de ESP32 gereset.

## 6.3 Logging page.

De logging pagina toont de logging van de ESP32 software, zoals die ook beschikbaar is via de USB connector (maar die is doorgaans niet handig te bereiken). Er is een beperkte hoeveelheid geheugen beschikbaar voor de logging, dus na enige tijd stopt het opslaan van de logregels.

De pagina ziet er als volgt uit:



The screenshot shows a web-based logging interface. At the top, a navigation bar includes links for 'Venus-control', 'Home', 'Config', 'Logging' (which is highlighted in blue), and 'Over'. The main area is a scrollable text box displaying log messages. The log messages are timestamped and show the initialization of the ESP32, configuration of partitions (nvs, otadata, app0, appl, spiffs, coredump), and the configuration of the WiFi interface (P1\_Dongle\_Pro, 192.168.1.172, 192.168.1.199, 192.168.1.199). It also shows the creation of a WiFi list, the WiFi station stopping, and the WiFi interface becoming ready. The log concludes with the WiFi station starting, connecting to a network (SSID: Nokia-basis-11, IP: 192.168.1.199), and connecting to a web server. A green button labeled 'Wis' (Delete) is located at the bottom right of the log area.

De linker kolom toont het tijdstip van de melding. In dit voorbeeld is die nog niet gevuld, maar zodra er een tijdstip via de P1 dongle binnenkomt wordt die gevuld.

Met de "Wis"-knop worden de meldingen gewist en worden nieuwe meldingen weer opgeslagen.

## 6.4 Over page.

Deze pagina toont wat informatie van het systeem. Het ziet er nu zo uit:



The screenshot shows a web-based 'Over' page. The top navigation bar includes links for 'Venus-control', 'Home', 'Config', 'Logging', and 'Over' (which is highlighted in blue). The main content area features a logo of a green and white circuit board with a recycling symbol. To the right of the logo, the text 'Venus-control' is displayed in a large, bold, green font. Below that, in a smaller green font, is the text 'ESP32 sturing van de Marstek Venus'. At the bottom of the page, a blue banner contains the text '\*\* Venus-control. ESP32 besturing voor Marstek Venus via ModBus \*\*'. Below this banner, a link 'Lees de documentatie voor meer details.' is shown in blue. At the very bottom, the text 'Ontwikkeld door Ed Smallenburg (ed@smallenburg.nl)' and 'Datum: Februari 2025' is displayed in a small, dark font.

De genoemde "documentatie" zit je nu te lezen.