



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 |

3.1 ModBus speciaal.

- Bij experimenten bleek dat wanneer de modbus requests te snel na elkaar kwamen, er fout optraden. Dat is gecorrigeerd door een delay in te bouwen na de reply van elk request.
- Bij update van de firmware voor de batterij (van Marstek) bleek dat niet te lukken. Na het losnemen van de ModBus aansluiting werkte het direct.

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.

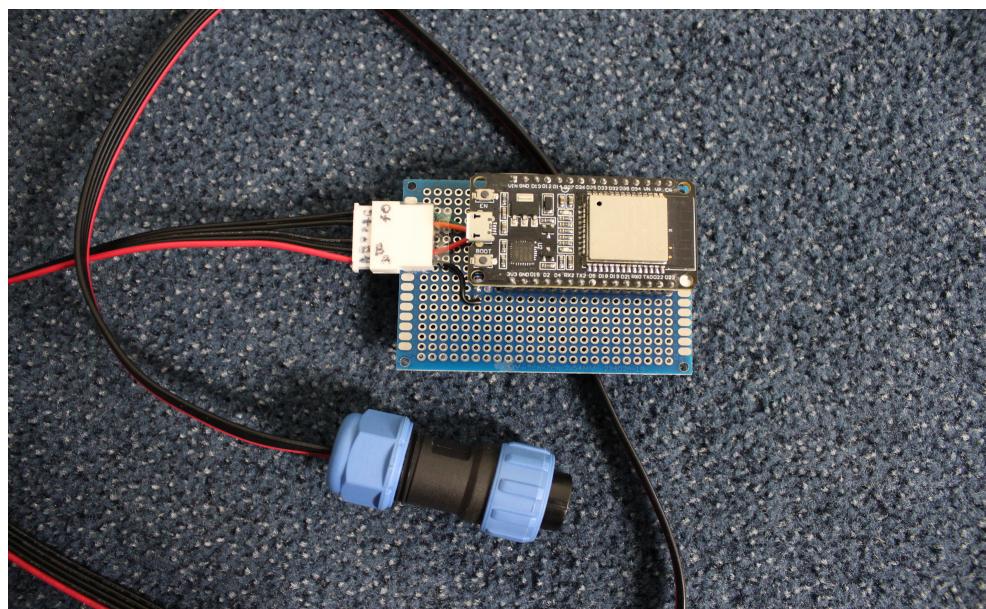
Soms geeft het uitlezen van de dongle een fout. Dat gebeurd als er gelijktijdig een ander apparaat de dongle probeert uit te lezen. In zo'n geval wordt de uitlezing genegeerd.

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. Een 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:

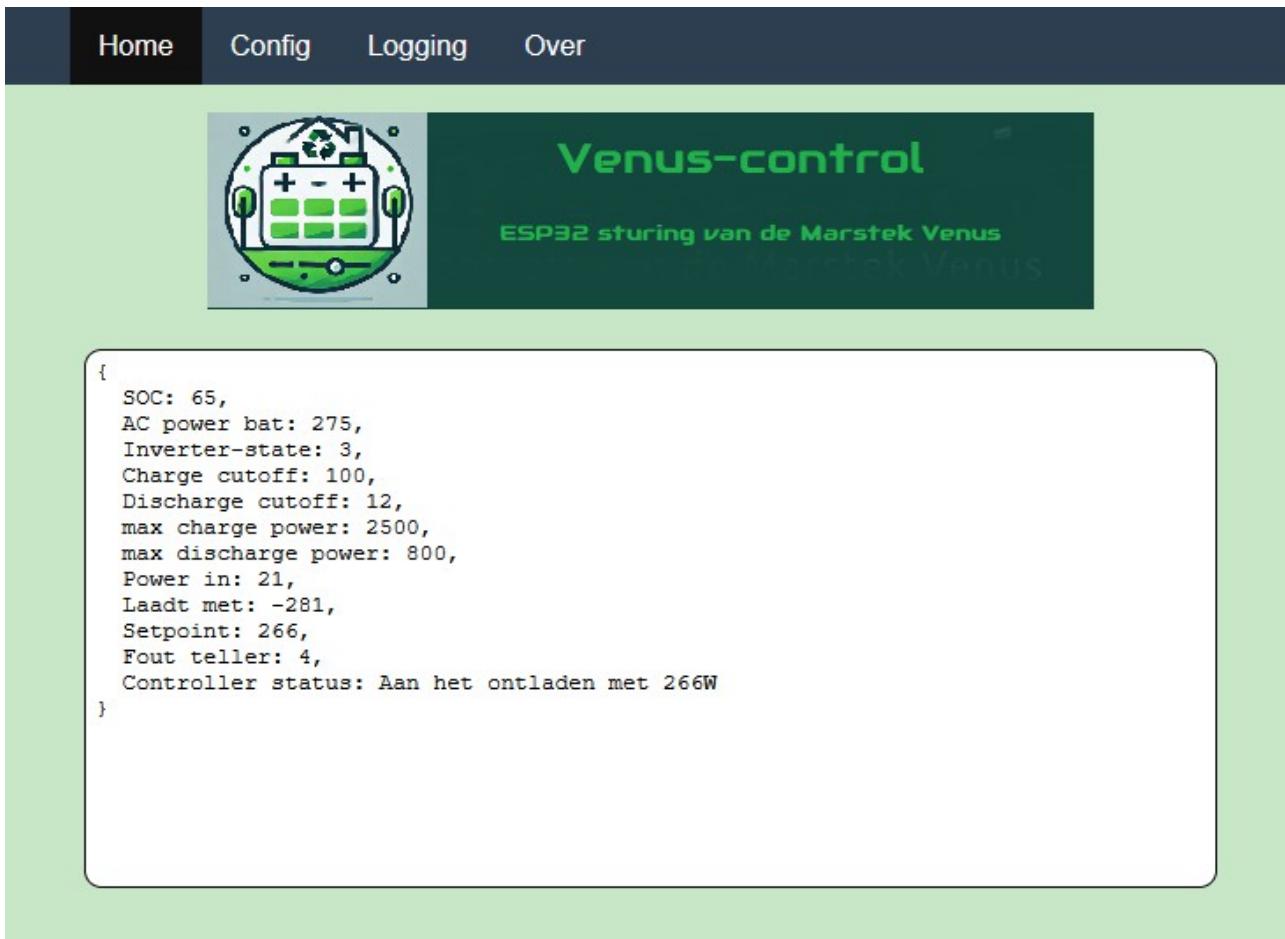
- Kan data van de P1-dongle ophalen. Nu is dat één keer per 5 seconden.
- Kan registers vanuit de Venus inlezen via ModBus.
- Kan registers schrijven via ModBus.
- Web-interface werkt.
- Kan procesdata presenteren in web-interface.

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: 65,  
    AC power bat: 275,  
    Inverter-state: 3,  
    Charge cutoff: 100,  
    Discharge cutoff: 12,  
    max charge power: 2500,  
    max discharge power: 800,  
    Power in: 21,  
    Laadt met: -281,  
    Setpoint: 266,  
    Fout teller: 4,  
    Controller status: Aan het opladen met 266W  
}
```

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 65 procent.
- “AC power bat” is het huidige ontlad vermogen van de batterij. In dit geval wordt er met 275 W aan het net bijgedragen.
- Inverter-state is hier “3”, dat betekent “discharge”. 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.

Het scherm wordt automatisch elke 5 seconden ververst.

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, there is a navigation bar with links for Home, Config, Logging, and Over. Below the navigation bar, a message in Dutch reads: "Je kunt hier de configuratie wijzigen. Let op: sommige wijzigingen hebben pas effect na herstart." (You can change the configuration here. Note: some changes will only take effect after a restart.)

```
dongle = P1_Dongle_Pro
dongle_api = /api/v2/sm/actual
dongle_host = 192.168.1.172
updhost = venus.smallenburg.nl
v_firmware = Wed, 26 Feb 2025 11:57:07 GMT
v_webintf = Mon, 24 Feb 2025 11:07:13 GMT
wifi = Nokia-basis-11/xxxxxxxx
```

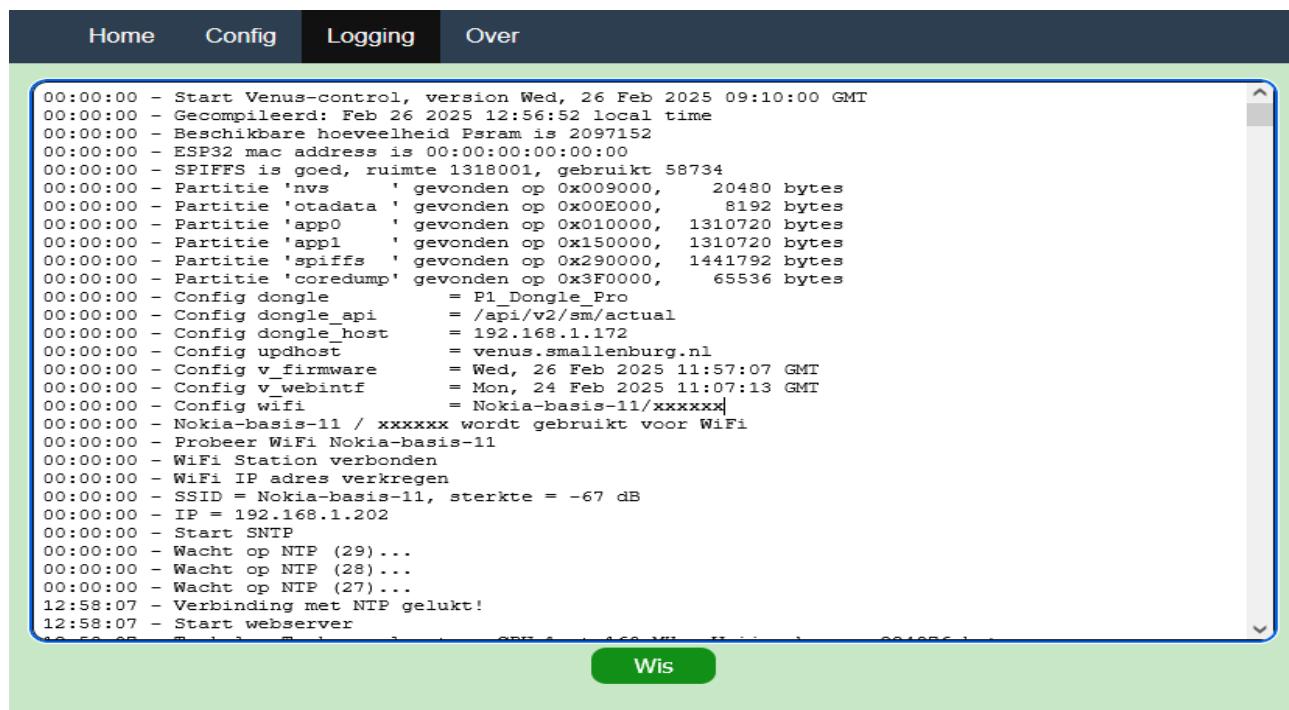
At the bottom of the configuration area, there are four buttons: Opslaan (Save), Herstart (Restart), Update firmware, and Update webinterface. Below these buttons, a progress bar indicates the status: "Wacht op input...." (Waiting for input....).

1. 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.
Bij "v_firmware" en "v_webintf" zie je wat de huidige versie van de firmware en van de web-interface is. Met de knop "Update firmware en/of "Update webinterface" kan een nieuwe versie worden geladen van een externe server (updhost). Verder zie je het WiFi netwerk met het password. 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, there is a navigation bar with tabs: Home, Config, Logging (which is highlighted in red), and Over. Below the navigation bar is a scrollable text area containing log messages. The log messages are timestamped and show the initialization of the Venus-control software, including the discovery of partitions like 'nvs', 'otadata', 'app0', 'appl', and 'spiffs', and the configuration of a WiFi connection to 'Nokia-basis-11'. The log concludes with the successful start of the webserver at 12:58:07. At the bottom of the log area is a green button labeled 'Wist' (Clear). The background of the page is light green.

```
00:00:00 - Start Venus-control, version Wed, 26 Feb 2025 09:10:00 GMT
00:00:00 - Gecompileerd: Feb 26 2025 12:56:52 local time
00:00:00 - Beschikbare hoeveelheid Psram is 2097152
00:00:00 - ESP32 mac address is 00:00:00:00:00:00
00:00:00 - SPIFFS is goed, ruimte 1318001, gebruikt 58734
00:00:00 - Partitie 'nvs' gevonden op 0x009000, 20480 bytes
00:00:00 - Partitie 'otadata' gevonden op 0x00E000, 8192 bytes
00:00:00 - Partitie 'app0' gevonden op 0x010000, 1310720 bytes
00:00:00 - Partitie 'appl' gevonden op 0x150000, 1310720 bytes
00:00:00 - Partitie 'spiffs' gevonden op 0x290000, 1441792 bytes
00:00:00 - Partitie 'coredump' gevonden op 0x3F0000, 65536 bytes
00:00:00 - Config dongle = P1_Dongle_Pro
00:00:00 - Config dongle_api = /api/v2/sm/actual
00:00:00 - Config dongle_host = 192.168.1.172
00:00:00 - Config upghost = venus.smallenburg.nl
00:00:00 - Config v_firmware = Wed, 26 Feb 2025 11:57:07 GMT
00:00:00 - Config v_webintf = Mon, 24 Feb 2025 11:07:13 GMT
00:00:00 - Config wifi = Nokia-basis-11/xxxxxx
00:00:00 - Nokia-basis-11 / xxxx wordt gebruikt voor WiFi
00:00:00 - Probeer WiFi Nokia-basis-11
00:00:00 - WiFi Station verbonden
00:00:00 - WiFi IP adres verkregen
00:00:00 - SSID = Nokia-basis-11, sterkte = -67 dB
00:00:00 - IP = 192.168.1.202
00:00:00 - Start SNTP
00:00:00 - Wacht op NTP (29)...
00:00:00 - Wacht op NTP (28)...
00:00:00 - Wacht op NTP (27)...
12:58:07 - Verbinding met NTP gelukt!
12:58:07 - Start webserver

```

Wist

De linker kolom toont het tijdstip van de melding. In dit voorbeeld zie je dat zodra er een npt-server bereikt kan worden. Als dat niet lukt, dan zie je de tijd sinds de laatste reset.

Met de "Wist"-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. At the top, there is a navigation bar with tabs: Home, Config, Logging, and Over (which is highlighted in red). Below the navigation bar is a section with a circular icon containing a battery and a gear, followed by the text 'Venus-control' and 'ESP32 sturing van de Marstek Venus'. The page then displays three main sections in blue text: '** Venus-control. **', '** ESP32 besturing voor Marstek Venus **', and '** via ModBus **'. Below these sections is a link to 'Lees de documentatie voor meer details.' (Read the documentation for more details.). At the bottom, it states 'Ontwikkeld door Ed Smallenburg (ed@smallenburg.nl)' and 'Datum: Februari 2025'. The background of the page is light green.

** Venus-control. **

** ESP32 besturing voor Marstek Venus **

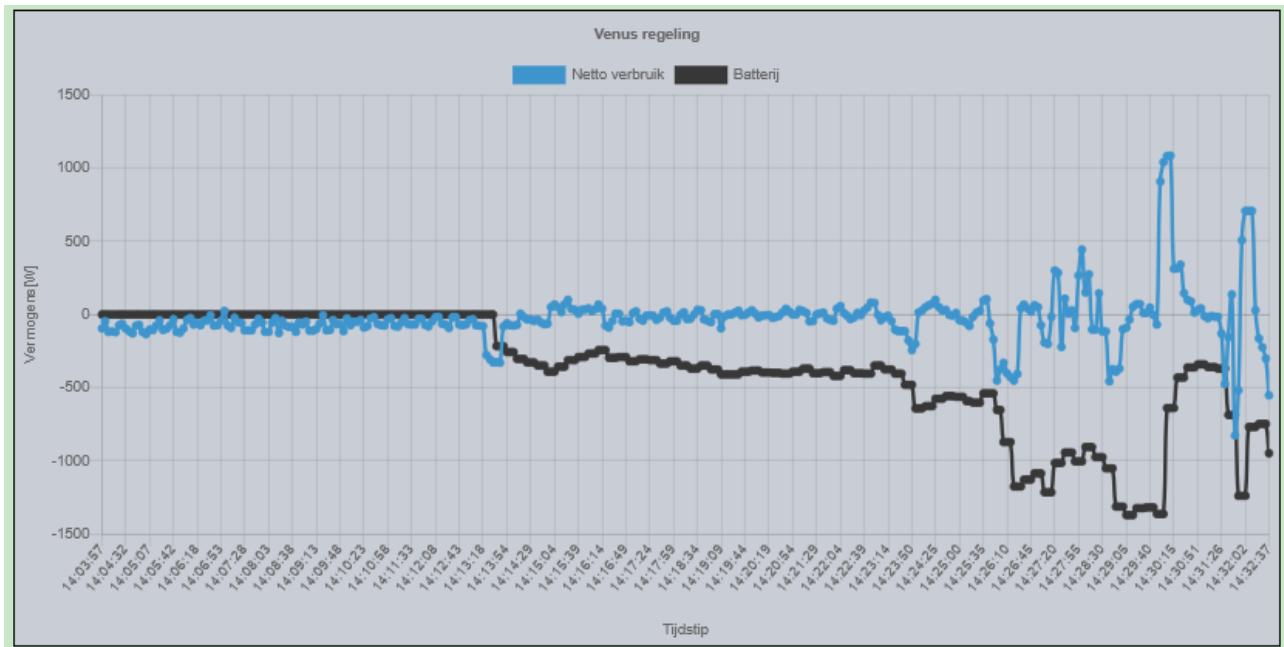
** via ModBus **

Lees de documentatie voor meer details.

Ontwikkeld door Ed Smallenburg (ed@smallenburg.nl)
Datum: Februari 2025

De genoemde "documentatie" zit nu te lezen.

7 Regeling.

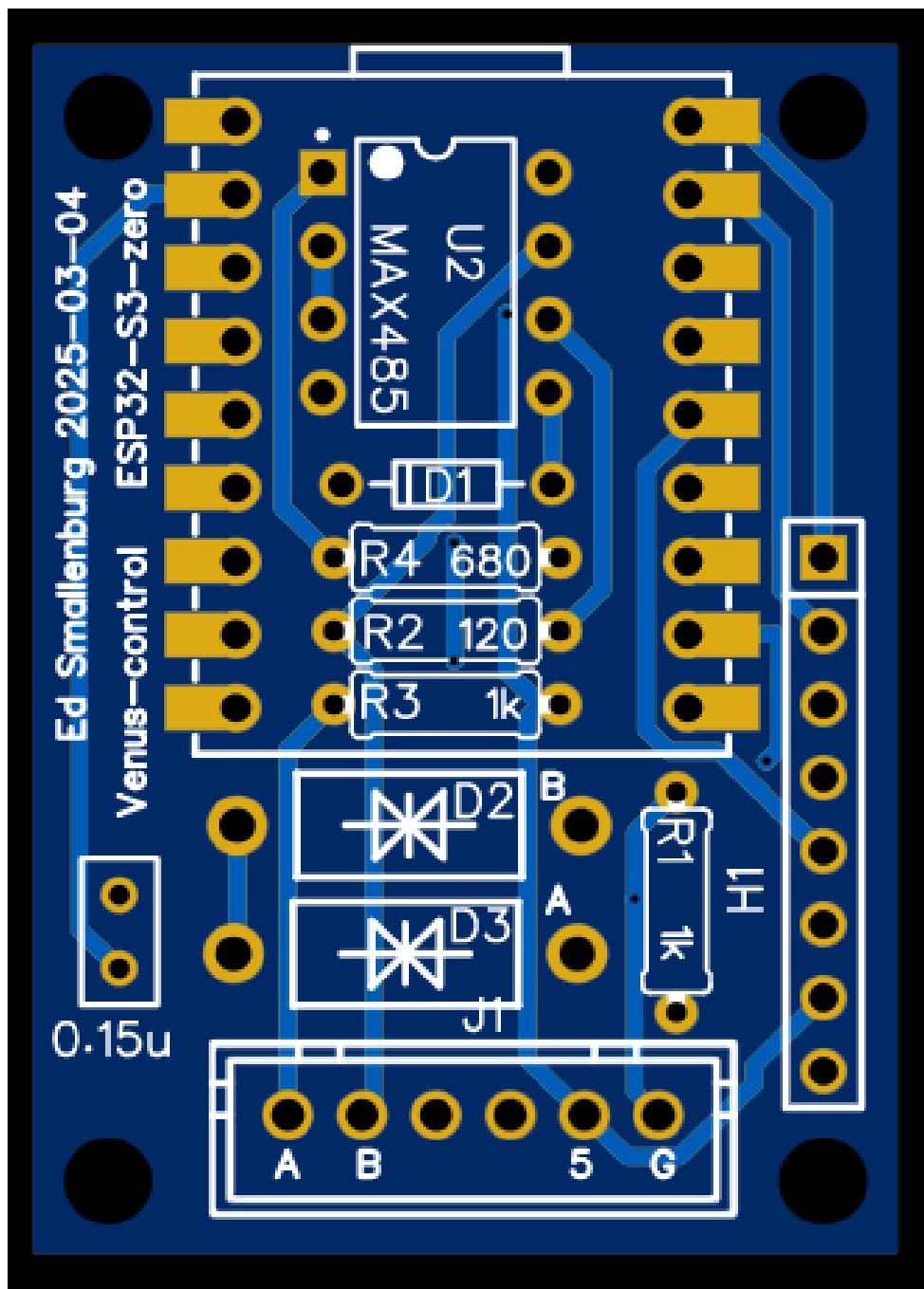


8 Print ontwerp.

Voor dit project is een print ontwikkeld.

8.1 Print.

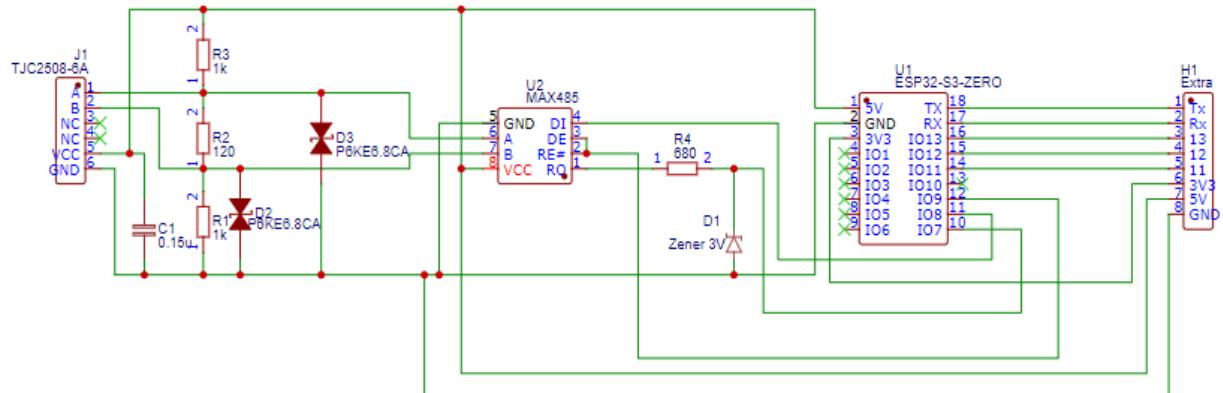
De print ie ziet er zo uit:



Onder de ESP32-S3-zero bevindt zich de MAX485 driver met nog enkele onderdelen. Dit is mogelijk omdat de ESP32-SE-mini op een voetje is gemonteerd. Ook de MAX485 kan op een voetje worden gemonteerd. Aan de rechterkant bevinden zich extra aansluitingen. Ditz maakt het mogelijk om bijvoorbeeld een OLED display of een extra sensor aan te sluiten.

8.2 Schema.

Het schema is als volgt:



9 ESP-32-S3 mini/zero hardware.

