



# Ontwikkeling Venus-E thuisbatterij.

## 1 Inleiding.

Dit document beschrijft de ontwikkeling van een module voor de besturing van de Marstek Venus-E. De bedoeling is om te komen tot een stabiele en verbeterde regeling van de batterij. De naam van het module is "Venus-control".

De huidige versie van Marstek 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.

De voordelen van "Venus-control" 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 mogelijk maken.
- Geen informatie naar de cloud.
- Simulatie van energiemeter om model te testen is beschikbaar.
- Data beschikbaar stellen via MQTT.
- Beschikbaar stellen van een Web-interface.
- Zelfbouw mogelijk maken.
- Print ontwerpen en produceren in samenwerking met **PCBWay**.

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.

### 1.1 Venus-control module.

Het module is gebaseerd op een ESP32-S3-ZERO die met een RS485 driver wordt verbonden met de ModBus aansluiting van de Venus. De voeding wordt door de Venus verzorgd.

Belangrijk is dat er een goede WiFi verbinding mogelijk is op de plaats waar de Venus zich bevindt.

## 2 ModBus communicatie.

Het communicatie-protocol tussen de Venus en de Venus-control is volgens ModBus. Dat wil zeggen dat er commando's en data uitgewisseld kan worden via 2 draden met een A en een B signaal. De kabel tussen de Venus en de controller mag zeer lang zijn, meer dan 1 kilometer. Het is echter praktischer om de leiding kort te houden, in het bijzonder om de 5 Volt voeding van de Venus te kunnen gebruiken. Bij een lange leiding kan dat teveel verlies opleveren.

### 2.1 Registers in de Venus.

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

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

### 2.2 ModBus speciaal.

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

### 3 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>.

In de configuratie pagina van de wb-interface moet je opgeven welke dongle wordt gebruikt en via welk IP-adres (of hostname) deze te bereiken is. Zie paragraaf 7.4.

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

#### 3.1 P1-Dongle-Pro.

De dongle (P1-Dongle-Pro) geeft op <http://<IP>/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"}
}
```

#### 3.2 Homewizard P1.

Hierbij wordt de V1 API gebruikt. Deze moet eerst geactiveerd worden. In de documentatie staat:

To access the data from an Energy device, you have to enable the API.

You can do this in the [HomeWizard Energy app](#).

Go to Settings → Meters → *Your meter*, and turn on *Local API*.

Met het http request <http://<IP>/api/v1/data> krijgen we een JSON structuur terug. Daarin is het veld “active\_power\_w” interessant.

De hele JSON structuur ziet er zo uit:

```
{
  "smr_version": 50,
  "meter_model": "ISKRA 2M550T-101",
  "wifi_ssid": "My Wi-Fi",
  "wifi_strength": 100,
  "total_power_import_t1_kwh": 10830.511,
  "total_power_import_t2_kwh": 2948.827,
  "total_power_export_t1_kwh": 1285.951,
  "total_power_export_t2_kwh": 2876.51,
  "active_power_w": -678,
  "active_power_l1_w": -676,
}
```

#### 3.3 MQTT energiemeter.

Het is ook mogelijk om het energieverbruik via MQTT binnen te halen. Daarvoor is een energie meter nodig die de meetgegevens via MQTT beschikbaar kan stellen. Dat is bijvoorbeeld mogelijk met de “P1-Dongle-Pro” en de “Homewizard P1”. Zo'n dongle publiceert dan de meting via een “broker” als een “topic”. De Venus-control is dan “subscribed” op zo'n topic en krijgt dan regelmatig de metingen door.

In de configuratie pagina van de web-interface moet je dan de gegevens van de broker instellen plus het topic waar de geleverde en soms de teruggegeven energie wordt vermeld.

### **3.4 P1\_Simul.**

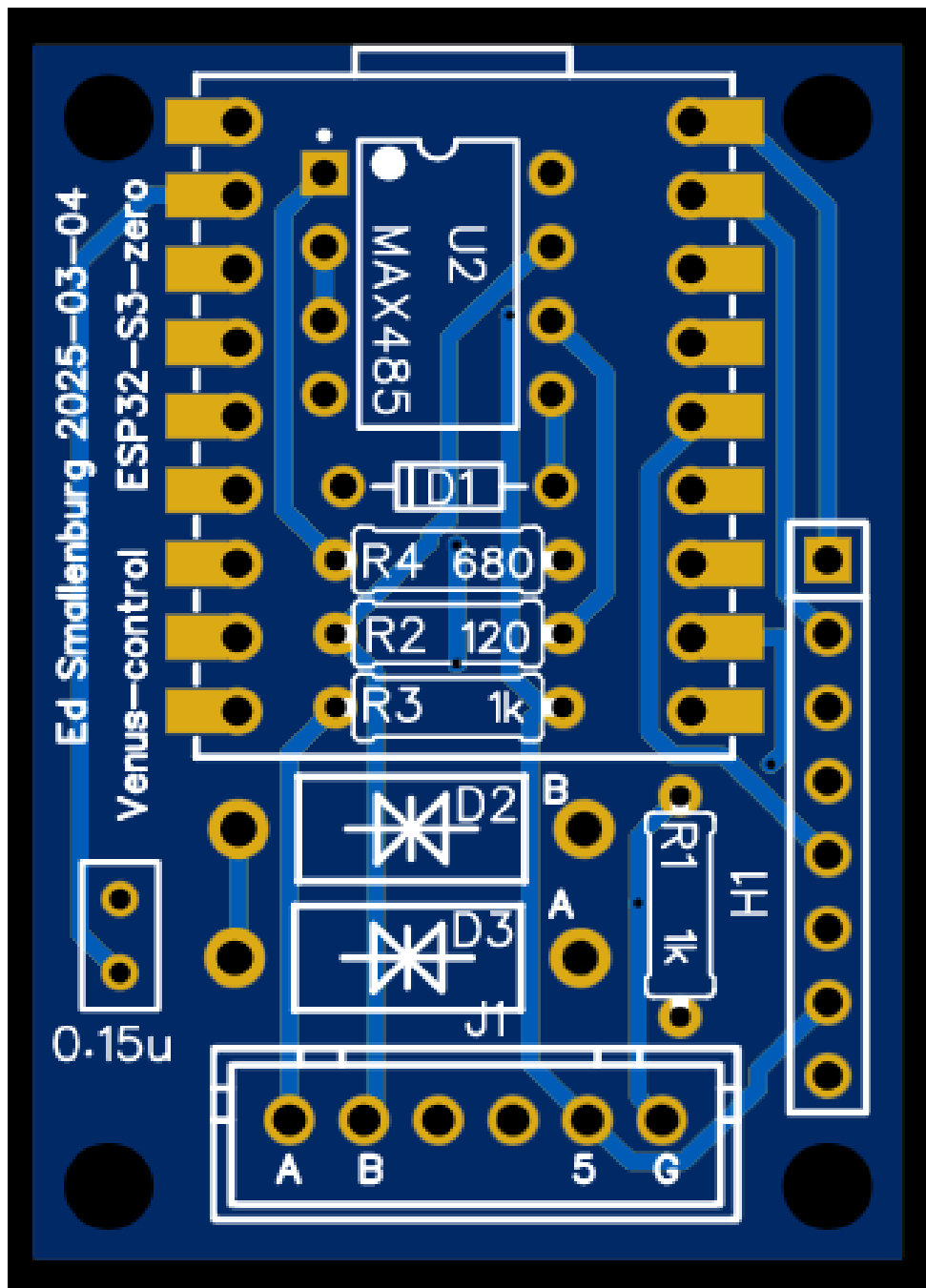
Dit is een gesimuleerde dongle. Er wordt een energiemeter gesimuleerd met een zich herhalend patroon van energie verbruik en energie opwekking. Elke 40 seconden wordt een stap gedaan. Het patroon is: 1000W, 800W, -2000W, 400W. Een negatieve waarde betekent hier opwekking.

## 4 Print ontwerp.

Voor dit project is een print ontwikkeld in samenwerking met **PCBWay**. Dit bedrijf staat bekend om het snel leveren van prints met een uitstekende kwaliteit.

### 4.1 Print.

De print ziet er zo uit:

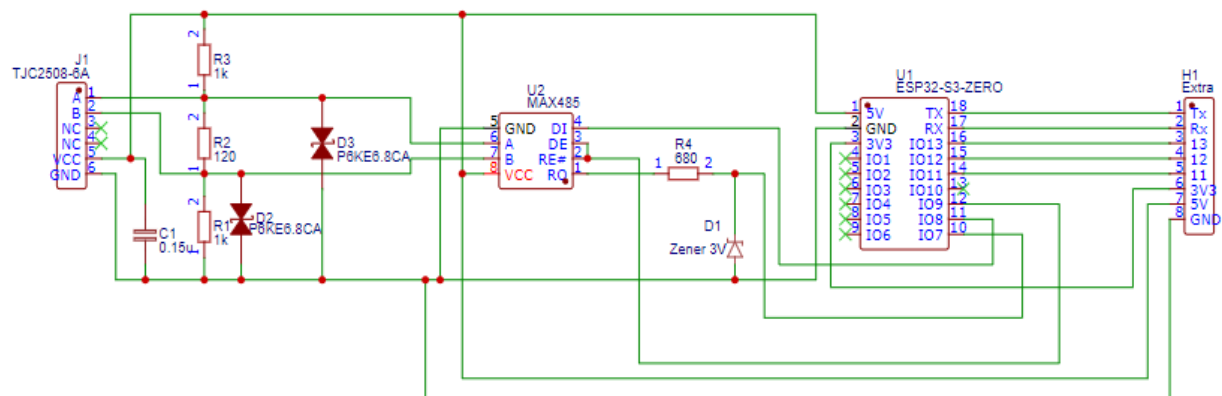


Onder de ESP32-S3-ZERO bevindt zich de MAX485 driver met nog enkele onderdelen. Dit is mogelijk omdat de ESP32-S3-ZERO op een voetje is gemonteerd. Ook de MAX485 kan op een voetje worden gemonteerd.

Aan de rechterkant (H1) bevinden zich extra aansluitingen. Dit maakt het mogelijk om later bijvoorbeeld een OLED display of een extra sensor aan te sluiten.

## 4.2 Schema.

Het schema is als volgt:



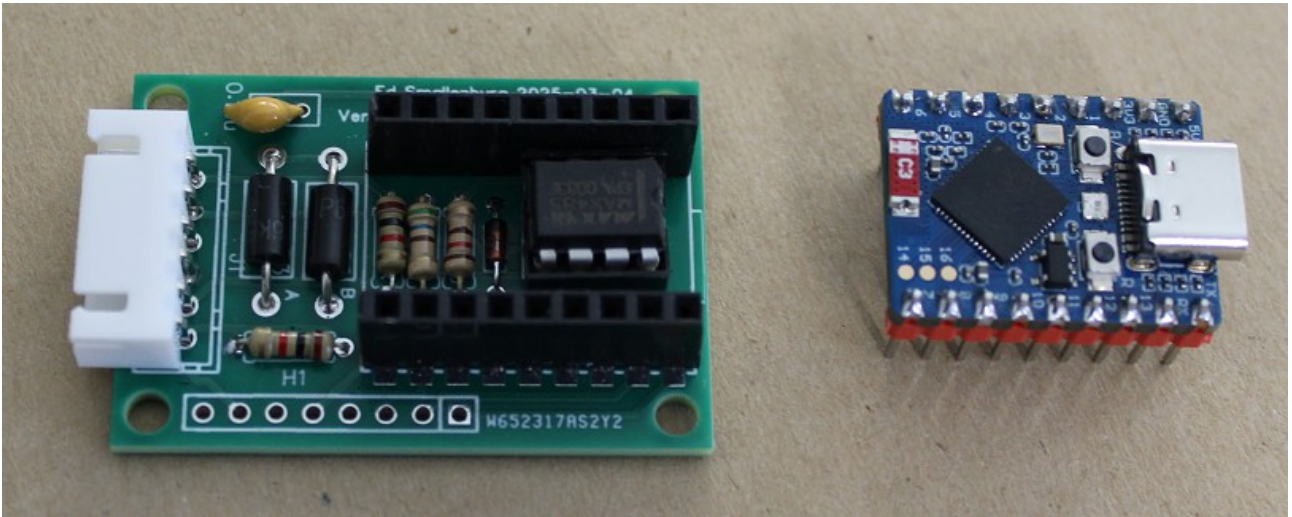
## 4.3 Onderdelenlijst.

Onderdeel	Opmerking	Prijs (ongeveer)
Print	PCBWay W652317AS2Y2	€ 2,00
ESP32-S3-ZERO	Dual core CPU 4 MB flash, 2 MB Ppsram	€ 3,50
MAX485	Of SN75ALS176 of SN75LBC184.	€ 0,25
P6KE6.8CA (2x)	Beveiligingsdiodes A en B ingang.	€ 0,20
Zenerdiode 3.0 Volt	Aanpassing 5V → 3.3 V logica	€ 0,10
Weerstand 1kΩ, 0.25 W (2x)	Pull-up ModBus	€ 0,10
Weerstand 560 Ω	Aanpassing 5V → 3.3 V logica	€ 0,05
Weerstand 120 Ω	Afsluiting ModBus	€ 0,05
Condensator 150 nF	Voedingspanning filter	€ 0,10
Header 2.54 mm, 9 p, female (2x)	Voet voor ESP32-S3-ZERO	€ 0,20
IC voet DIL 8p	Voet voor MAX485	€ 0,10
Connector, haaks, 6p, 2.54 mm	Connector voor Marstek kabel	€ 0,20
Kastje	Doosje voor de print	€ 2.50

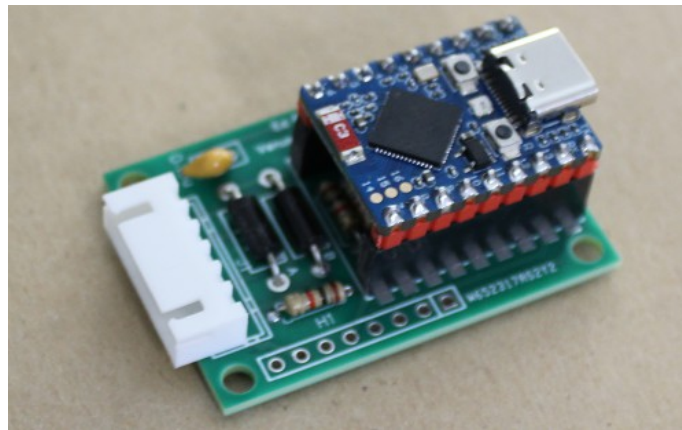
Totaal ongeveer € 10,00.

## 5 Uiteindelijke print.

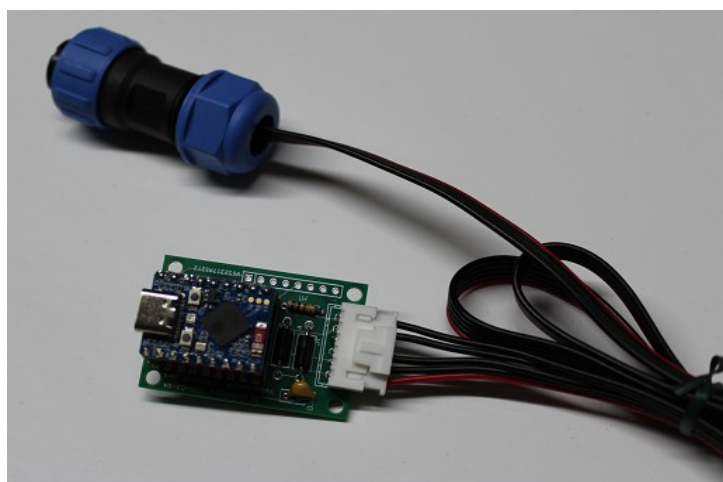
De print met gemonteerde onderdelen ziet er zo uit:



De ESP32-S3-ZERO rechts wordt geplaatst in de voet op de print links. Op de print staat een merkteken dat aangeeft aan welke kant de USB moet komen te zitten. Dat ziet er dan zo uit:



De kabel van de Venus wordt verbonden met de witte connector op de print. Dat ziet er dan zo uit:



## 6 Quick start.

1. Verbind de print met de bij de Venus bijgeleverde kabel en steek het andere einde van de kabel in de ModBus aansluiting van de Venus.
2. De LED op de print zal bij het opstarten eerst 16 keer langzaam knipperen met patroon van wit en groen licht. Mogelijk herhaalt zich de opstartprocedure nog een keer.
3. Daarna knippert de LED afwisselend groen en blauw terwijl er geprobeerd contact te maken met het WiFi netwerk. Bij een nieuwe installatie zal dat niet lukken en wordt na een minuut een (tijdelijk) eigen WiFi-netwerk aangemaakt met de naam "Venus\_xxxx". Hierbij is "xxxx" een hexadecimaal getal, dat uniek is voor elke ESP32-S3-ZERO.
4. Nu kan je op een PC (of telefoon of laptop of tablet) contact maken met dit WiFi-netwerk. Het password voor dit netwerk is gelijk aan de naam van het netwerk, dus ook "Venus\_xxxx". Het netwerk is bereikbaar in de buurt van de controller.
5. Zodra je contact hebt met dit WiFi-netwerk kan je met een browser (Chrome, Firefox, Edge) contact maken met <http://192.168.4.1>, dat is het tijdelijke IP-adres van de controller.
6. Ga eerst in de browser naar de "Logging"-pagina. Één van de regels bevat het "Esp32 mac address". Noteer dat. Het kan later van pas komen.
7. Ga dan in de browser naar de "Config"-pagina (zie hoofdstuk 7). Enkele gegevens zijn al ingevuld, maar je moet in ieder geval de gegevens bij "wifi" aanpassen met de naam en het password van het "huis"-Wifi-netwerk dat bereikbaar is op de plaats van de Venus-control. Je moet ook de gegevens van de dongle aanpassen. Klik vervolgens op "Opslaan" en daarna op "Herstart".
8. Nu moet de Venus-control bereikbaar zijn via het huis-WiFi-netwerk. Het is nog even zoeken naar het IP-adres van de controller, omdat die wordt toegewezen door de WiFi-router. Via de router moet dat adres te achterhalen zijn met behulp van het MAC-adres, en de meeste routers hebben een mogelijkheid om een vast IP-adres te koppelen aan het MAC-adres van de ESP32-S3-ZERO. Als het IP-adres niet te vinden is, dan kan dat achterhaald worden door de controller met een USB kabel te verbinden met een PC of laptop. Dan komt er op de PC een com-poort bij en als je die kunt uitlezen, bijvoorbeeld met "puTTY", dan zie bij het opstarten het IP-adres voorbijkomen.
9. Ga met de browser naar de logging pagina en controleer of er geen fouten komen van de dongle. Zonder dongle kan de regeling niet werken. Foutmeldingen zijn herkenbaar door een uitropteken aan het einde van een log-regel.



## 7 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.

### 7.1 Home pagina.

De home page ziet er als volgt uit:

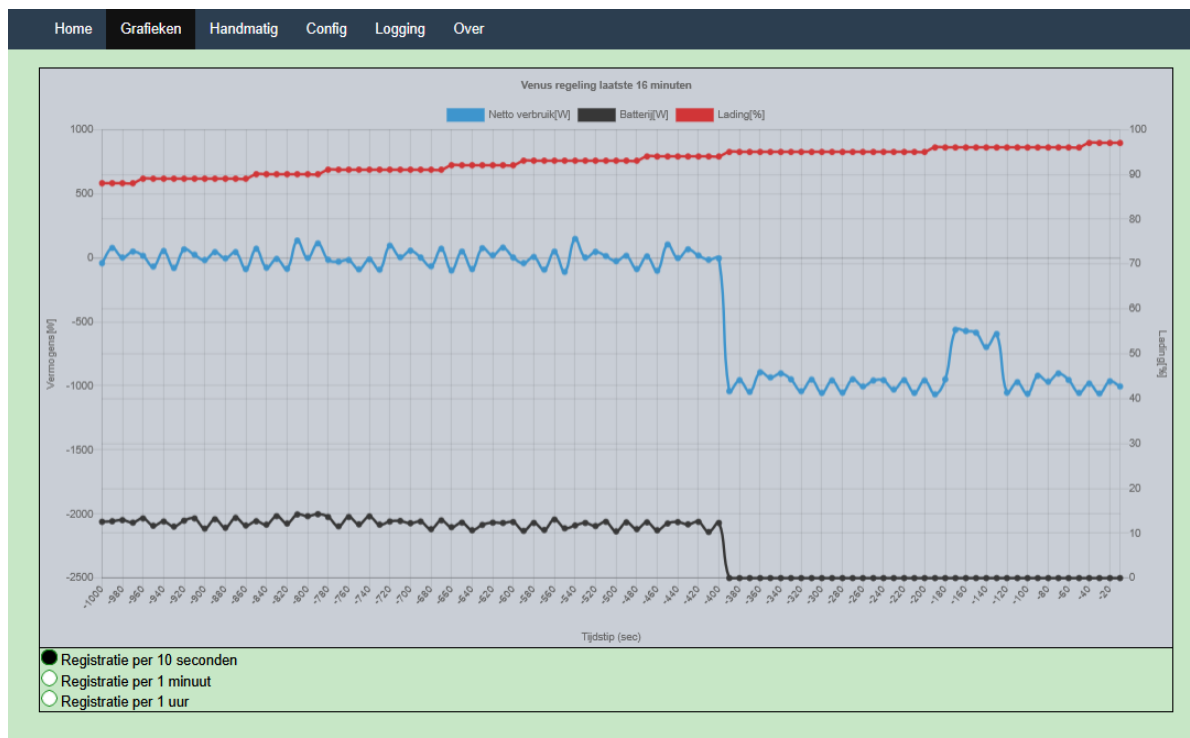


- De eerste meter geeft de huidige laad- of ontlad energie aan. De schaal loopt van 800 W ontladen tot 2400 W laden. In dit voorbeeld wordt er met 1049 Watt geladen.
- De tweede meter geeft aan in hoeverre de batterij geladen is. 'AC power bat' is het huidige ontlad vermogen van de batterij. In dit geval is de batterij voor 66 procent gevuld.
- De grafiek geeft het netto energie-verbruik en de bijdrage van de batterij daarin. In de titel van de grafiek zie je: "Venus regeling "nom". "nom" staat voor "Nul Op de Meter". Dan wordt het laden en ontladen zo geregeld dat er netto 0 Watt (terug)geleverd wordt aan het net. Dit is de normale toestand. Je kunt de regeling ook handmatig instellen. Dan verschijnt "stb" (voor stand-by) of "man" (voor manual). Bij "man" is het vermogen handmatig ingesteld, hoewel er rekening wordt gehouden met de ladingstoestand van de batterij. In dit voorbeeld zie je dat er tussen 09:02:16 en 09:03:53 een belasting van 400 Watt wordt bijgeschakeld. Het opladen wordt dan teruggeregeld van 1000 naar 600 Watt om nul op de meter te houden

Het scherm wordt automatisch elke 5 seconden ververs.

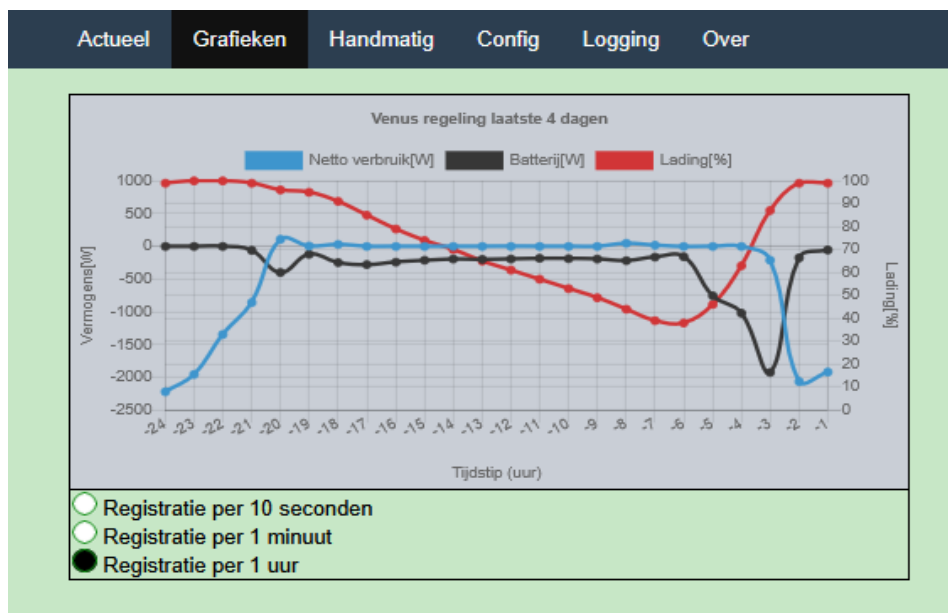
## 7.2 Grafieken pagina.

Op deze pagina kunnen grafieken worden getoond die de werking van Venus-control in grafiekvorm presenteren. De pagina ziet er zo uit:



In dit voorbeeld zie je een overzicht van de laatste 100 punten, per 10 seconden gemeten. De periode beslaat dan ongeveer 6 minuten.

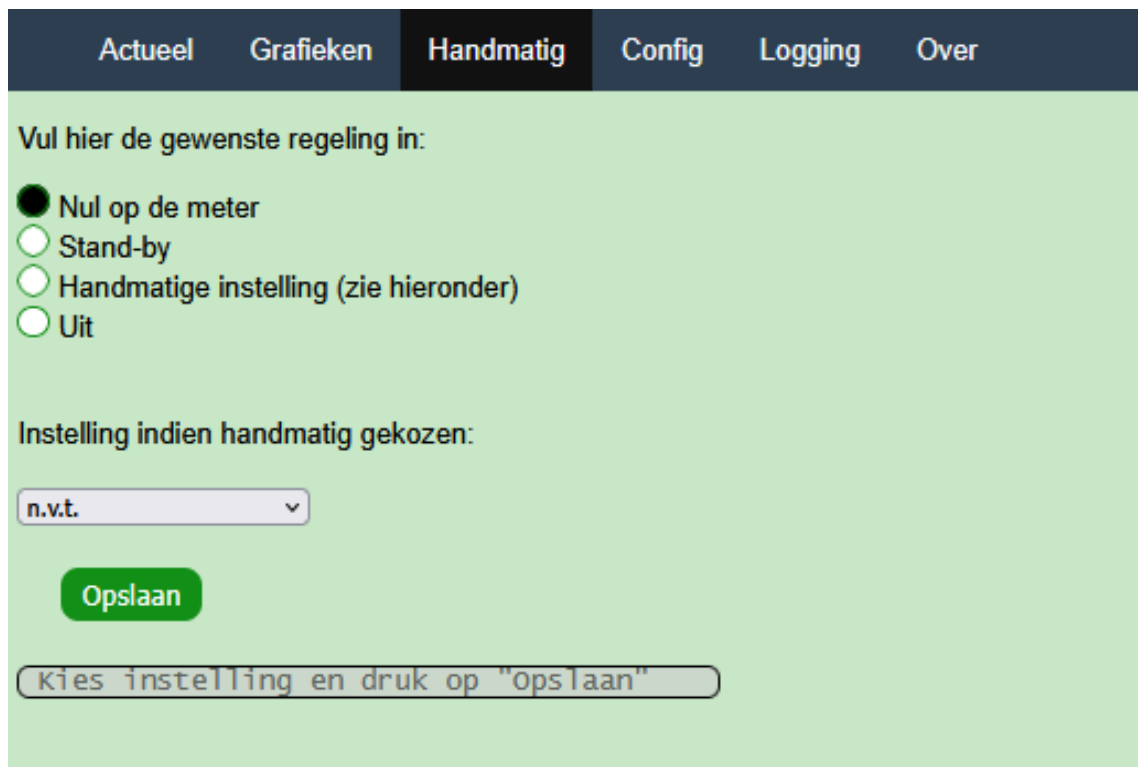
Onder in beeld kan de periode gekozen worden. Het aantal meetpunten is altijd 100 indien voldoende meetpunten beschikbaar. De registratie van de meetpunten gebeurt in het Psram geheugen van de ESP32, dus na een herstart is er geen oude data meer beschikbaar. Het overzicht per uur ziet (in dit geval vanaf 15:00) er zo uit:



Hier is goed het ontladen en weer opladen van de batterij te zien (rode lijn) gedurende een etmaal. De blauwe lijn laat zien dat er nul op de meter staat zolang er geen overcapaciteit van de zonnepanelen is.

### 7.3 Handmatig pagina.

Op deze pagina kan je batterij handmatig instellen of weer terugschakelen naar automatisch. Het scherm ziet er zo uit:



Bij de gewenste regeling kan je kiezen uit 4 mogelijkheden. "Nul op de meter" is de automatische modus. Bij "Stand-by" wordt het laden en ontladen uitgeschakeld. Bij "Handmatige instelling" kan je zelf het vermogen voor laden of ontladen instellen met behulp van de dropdown-box eronder. Bij "Uit" worden er geen ModBus commando's uitgestuurd en kan je de Marstek app weer gebruiken om de batterij te sturen. Met de knop "Opslaan" wordt de gewenste regeling actueel gemaakt.

## 7.4 Config pagina.

De configuratie-pagina ziet er als volgt uit:

Home   Handmatig   Config   Logging   Over

Je kunt hier de configuratie wijzigen.  
*Let op: sommige wijzigingen hebben pas effect na herstart.*

```
dongle = P1_Dongle_Pro
dongle_api = /api/v2/sm/actual
dongle_host = 192.168.1.172|
updhost = venus.smallenburg.nl
v_firmware = Thu, 20 Mar 2025 13:24:08 GMT
v_webintf = Thu, 20 Mar 2025 13:44:32 GMT
wifi = Nokia-basis-11/xxxxxx
```

Opslaan   Herstart   Update firmware   Update webinterface

wacht op 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. Van belang zijn de naam van de dongle en het IP-adres. De api kan meestal weggelaten worden omdat er dan de default wordt gebruikt. Zie overzicht hieronder.

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). Na de update van de firmware wordt er een automatische herstart gedaan. Bij de update van de webinterface gebeurt dat niet omdat het niet nodig is.

Verder zie je het WiFi netwerk met het password. Met de “Opslaan”-knop worden gewijzigde instellingen opgeslagen en met “Herstart” wordt de ESP32 gereset.

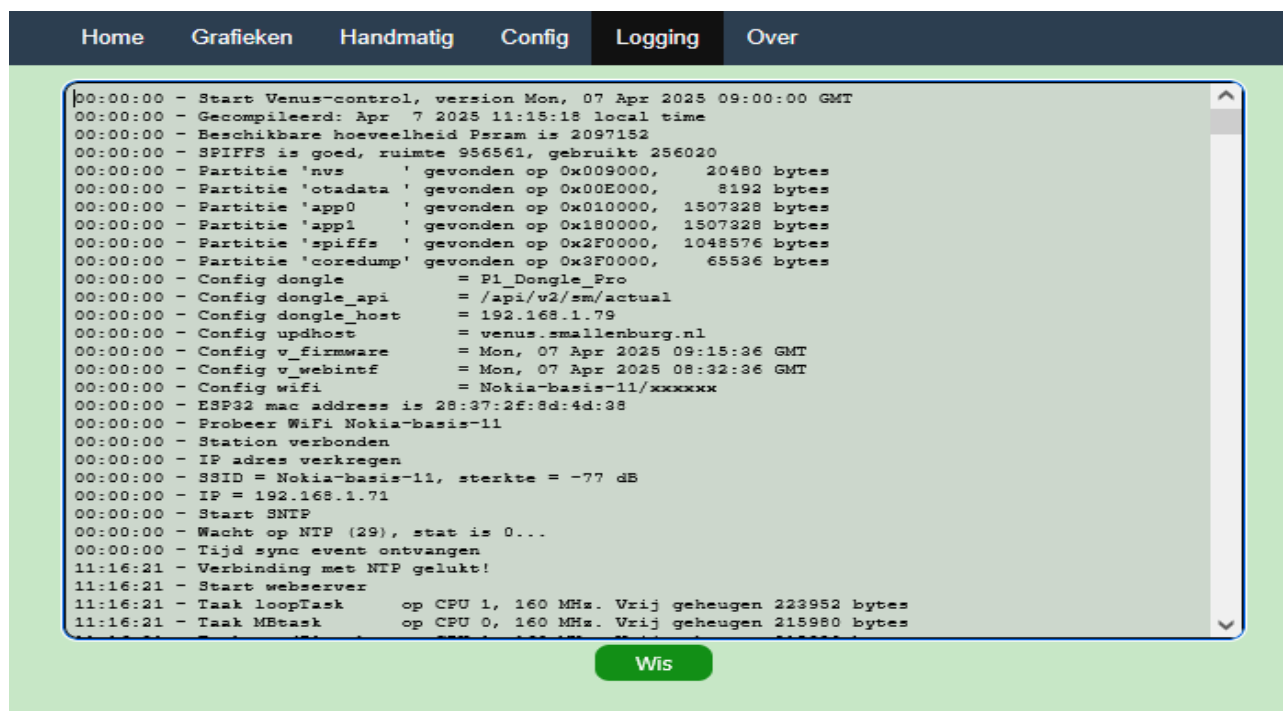
## 7.4.1 Overzicht alle configuratie instellingen.

dongle = <naam>	<p>Naam van de P1-dongle. Momenteel zijn er 4 mogelijkheden:</p> <ul style="list-style-type: none"> <li>• P1_dongle_Pro. Deze dongle wordt geleverd door smart-stuff.nl.</li> <li>• Homewizard. Deze wordt geleverd door homewizard.com.</li> <li>• Shellypro3. Driver voor de Shelly Pro 3 EM. Deze wordt geleverd door bijvoorbeeld home2link.nl. Het gaat hier niet om een dongle, maar om een energiemeter ingebouwd in de meterkast. Er is nog niet voldoende mee getest.</li> <li>• P1_MQTT. Driver voor een energiemeter die via MQTT zijn metingen doorgeeft.</li> <li>• P1_Simul. Deze simuleert een P1 dongle voor testdoeleinden.</li> </ul>
dongle_api = <api>	De API die gebruikt moet worden voor de opgegeven dongle. Meestal is dit niet nodig omdat dat vastligt voor de opgegeven dongle.
dongle_host = <IP-adres>	Het IP-adres waarop de P1 dongle via WiFi te bereiken is.
dongle_port = <poort>	Zet het poortnummer voor de dongle. Default is dat poort 80. Bij een MQTT meter is dat de poort van de broker, default 1883.
dongle_schaal = <factor>	Factor waarmee vermogens van de dongle moet worden vermenigvuldigd om Watts te krijgen. Voor de meeste dongles volstaat de default 1000 omdat de vermogens in kW worden aangeboden.
field_p_in = <spec>	Topic of JSON key voor de inkomende energie. Voor de meeste dongles wordt automatisch de default ingesteld. Voor energiemeters via MQTT moet dat ingevuld worden, bijvoorbeeld "mqtt_dongle/power_delivered".
field_p_out = <spec>	<p>Topic of JSON key voor de teruggeleverde energie. Voor de meeste dongles wordt automatisch de default ingesteld. Voor energiemeters via MQTT moet dat ingevuld worden, bijvoorbeeld "mqtt_dongle/power_returned".</p> <p>Sommige dongles hebben in de JSON een enkele waarde voor de geleverde energie. Die is dan negatief bij teruggeleverde energie. In dat geval is de specificatie van field_p_out overbodig.</p>
mincharge = <n>	Zet het minimale vermogen waarmee de batterij geladen/ontladen mag worden. Default is 100 Watt, waarden kleiner dan 25 Watt zijn niet mogelijk om te voorkomen dat er teveel geschakeld wordt tussen laden en ontladen.
mqtt_em = <user>:<passwd>@<broker>	<p>Specificatie van de MQTT broker voor een P1_MQTT energiemeter. "user" is de gebruikersnaam, "passwd" is het password en "broker" is het IP-adres of de domeinnaam van de MQTT broker. Voorbeeld:</p> <p>mqtt_em = willem:abc123@mijnbroker.com</p>
rs485_pin_rx = <pin>	Zet het pin-nummer voor de input van de RS485 converter. Voor de standaard print is dat niet nodig.
rs485_pin_tx = <pin>	Zet het pin-nummer voor de output naar de RS485 converter. Voor de standaard print is dat niet nodig.
rs485_pin_de = <pin>	Zet het pin-nummer voor de output enable van de RS485 converter. Voor de standaard print is dat niet nodig.
simul = 1	Simuleer P1 dongle EN ModBus. De andere dongle-gegevens worden genegeerd.
updhost = <updatehost>	Domeinnaam van de server waar updates gevonden kunnen worden. Default is dat "venus.smallenburg.nl".
venus_mb_adr = <n>	Het ModBus adres van de batterij. Default is dat "1".

wifi = <ssid/pw>	Zet WiFi credentials in. SSID en password zijn gescheiden door een "/".
------------------	---

## 7.5 Logging pagina.

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 (1000 regels) stopt het opslaan van de logregels. De pagina ziet er als volgt uit:



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

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

## 7.6 Over pagina.

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



De genoemde "documentatie" zit je nu te lezen.

## 8 Foutzoeken.

Als de Venus-control niet goed werkt, dan kun je een aantal checks uitvoeren.

### 8.1 Programma corrupt.

Als het programma niet opstart (geen wit-groen knipperend licht na reset) en/of je krijgt geen toegang tot de web-interface, dan is waarschijnlijk het programma in de ESP32-S3-ZERO defect geraakt.

Controleer of het wel werkt als je de ESP32 module van het printje haalt en voedt met behulp van de USB aansluiting.

In hoofdstuk 11 wordt beschreven hoe je de software opnieuw op de ESP32 kunt zetten.

### 8.2 Geen verbinding met de P1 dongle.

In de logging zie je dan bijvoorbeeld:

- "Verbinden met P1-dongle time-out!"
- "P1, kreeg een non-200 status terug van de api!"
- "JSON begint niet met een accolade!"
- "P1 JSON codering fout 3, lengte is 1184!"
- "P1 read lengte is 891, verwacht 1113!"
- "P1 onverwachte inhoud in antwoord!"

Controleer dan de settings voor de dongle in de config pagina van de web-interface.

Probeer ook of je met een browser de dongle via <http://<IP van de dongle>> kunt bereiken.

Je kunt de laatst gelezen JSON structuur van de dongle inzien door in je browser te gaan naar "<http://<IP van de Venus-control>/getjson>".

### 8.3 ModBus fouten.

Als je in de logging veel fouten ziet zoals:

- "MB: Fout bij lezen van register nnnnn!"
- "MB: Fout bij schrijven register nnnnn!"

Dan is er sprake van een ModBus storing.

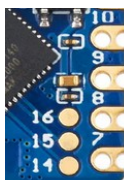
Controleer de kabel naar de batterij.

Met een voltmeter kun je een aantal punten testen. Op de print zie 2 punten gemerkt met "A" en "B":



Op de plaats van D1 en D2 zie je 2 zwarte diodes. Meet de spanning tussen A en B. Dat moet ongeveer 0.32 Volt zijn, waarbij A positief is t.o.v. B. Als je de Venus-control loskoppelt van de batterij en voedt met een USB kabel, dan meet je daar ongeveer 0.81 Volt.

Op de module zelf kan je de punten 7, 8 en 9 meten:



De spanning op punt 9 moet 0.0 Volt zijn. De spanning op punt 8 moet ongeveer 3.3 Volt zijn. De spanning op punt 7 moet ongeveer 3 Volt zijn.

Je kunt in je browser de laatste input van de ModBus zie door te gaan naar "<http://<IP van de Venus-control>/getstatus>". Op de eerste regel zie je dan bijvoorbeeld de lading van de batterij in procenten.



## **9 Bijlage: CT-003 dongle.**

Het MAC address (van mijn exemplaar) is 24:21:5e:db:2e:e6. Het IP-adres wordt 192.168.1.89. Ik zie ook 192.168.1.70 langskomen. Beide adressen zijn te pingen.

Op de print bevindt zich een aansluiting voor een seriële poort. De baudrate is 115200. Op de Tx pin zie je "AT"-commando's langskomen. En een poortnummer, 22222.

Het is mij nog niet gelukt om vanuit de Venus-control of via een browser contact te maken met deze dongle. Mogelijk communiceert deze dongle alleen met een externe (Chinese?) webserver.

## 10 Bijlage: Voorbeeld logging.

Bij een normale start ziet de logging er ongeveer zo uit:

```
00:00:00 - Start Venus-control, version Mon, 31 Mar 2025 17:40:00 GMT
00:00:00 - Gecompileerd: Mar 31 2025 17:50:56 local time
00:00:00 - Beschikbare hoeveelheid Psram is 2097152
00:00:00 - SPIFFS is goed, ruimte 956561, gebruikt 253008
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, 1507328 bytes
00:00:00 - Partitie 'app1' gevonden op 0x180000, 1507328 bytes
00:00:00 - Partitie 'spiffs' gevonden op 0x2F0000, 1048576 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 = 192.168.1.79
00:00:00 - Config updhost = venus.smallenburg.nl
00:00:00 - Config v_firmware = Mon, 31 Mar 2025 15:51:10 GMT
00:00:00 - Config v_webintf = Tue, 01 Apr 2025 13:10:58 GMT
00:00:00 - Config wifi = SSID/*****
00:00:00 - ESP32 mac address is 28:37:2f:8d:4d:38
00:00:00 - Probeer WiFi SSID
00:00:00 - Station verbonden
00:00:00 - IP adres verkregen
00:00:00 - SSID = SSID, sterkte = -71 dB
00:00:00 - IP = 192.168.1.71
00:00:00 - Start SNTP
00:00:00 - Wacht op NTP (29), stat is 0...
00:00:00 - Wacht op NTP (28), stat is 0...
00:00:00 - Tijd sync event ontvangen
15:11:51 - Verbinding met NTP gelukt!
15:11:51 - Start webserver
15:11:51 - Taak loopTask op CPU 1, 160 MHz. Vrij geheugen 223948 bytes
15:11:51 - Taak MBtask op CPU 0, 160 MHz. Vrij geheugen 215976 bytes
15:11:51 - Taak readPltask op CPU 1, 160 MHz. Vrij geheugen 215976 bytes
15:11:51 - Taak savetask op CPU 0, 160 MHz. Vrij geheugen 209740 bytes
15:11:51 - Taak controltask op CPU 0, 160 MHz. Vrij geheugen 210208 bytes
15:11:52 - Einde setup
15:11:55 - Aan het opladen met 1354W
```

## 11 Bijlage: flash utility.

Indien de firmware op de ESP32-S3-ZERO defect is geraakt, dan kan dat hersteld worden door de ESP32 opnieuw te flashen. Met PlatformIO is dat eenvoudig, maar het kan ook zonder. Dit hoofdstuk beschrijft het proces stap voor stap. De beschrijving is voor een Windows systeem.

### 11.1 Download de bootloader.

Ga naar <http://venus.smallenburg.nl/download/bootloader.bin>.  
Accepteer indien nodig het risico van een “onbeveiligde” verbinding.  
Hiermee wordt de file bootloader.bin opgeslagen in de download directory.

### 11.2 Download OTA data.

Ga naar [http://venus.smallenburg.nl/download/boot\\_app0.bin](http://venus.smallenburg.nl/download/boot_app0.bin).  
Accepteer indien nodig het risico van een “onbeveiligde” verbinding.  
Hiermee wordt de file boot\_app0.bin opgeslagen in de download directory.

### 11.3 Download de partition table.

Ga naar <http://venus.smallenburg.nl/download/partitions.bin>.  
Accepteer indien nodig het risico van een “onbeveiligde” verbinding.  
Hiermee wordt de file partitions.bin opgeslagen in de download directory.

### 11.4 Download de firmware.

Ga naar <http://venus.smallenburg.nl/download/firmware.bin>.  
Accepteer indien nodig het risico van een “onbeveiligde” verbinding.  
Hiermee wordt de file firmware.bin opgeslagen in de download directory.

### 11.5 Download de webinterface files.

Ga naar <http://venus.smallenburg.nl/download/spiffs.bin>.  
Accepteer indien nodig het risico van een “onbeveiligde” verbinding.  
Hiermee wordt de file spiffs.bin opgeslagen in de download directory.

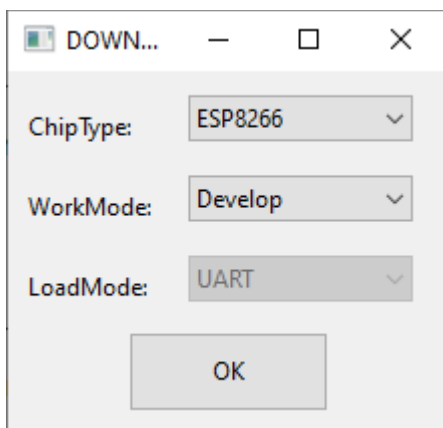
### 11.6 Verbinding met USB.

Verbind de ESP32-S3-ZERO via een USB-kabel (USB-C naar USB-A) met de computer.  
Druk vervolgens de button “B” in en houdt deze vast. Druk vervolgens op button “R” en laat deze weer los.  
Laat vervolgens button “B” weer los.  
De ESP32 staat nu in download mode.

### 11.7 Download de ESPtool utility.

Ga naar [https://dl.espressif.com/public/flash\\_download\\_tool.zip](https://dl.espressif.com/public/flash_download_tool.zip) en pak de zipfile uit.

Start vervolgens, in de hierdoor aangemaakte folder, het programma “flash\_download\_tool\_3.9.8\_w1.exe”.  
Er opent een venster dat er zo uit ziet:



Kies voor ChipType de “ESP32-S3”, en voor LoadMode “USB”. Laat WorkMode staan op “Develop”. Klik op “OK”. Je krijgt dan een nieuw venster met 2 tabbladen. Om te testen kan je tabblad “chipInfoDump” openen. Je kiest dan de juiste “port”, bijvoorbeeld “COM30”. Als je vervolgens op “Chip Info” klikt, dan zie je zo iets:

```
start detect chip...please wait
chip sync ...
Chip is ESP32-S3 (QFN56) (revision v0.2)
Features: WiFi, BLE, Embedded Flash 4MB (XMC), Embedded PSRAM 2MB (AP_3v3)
Crystal is 40MHz
MAC: 28372f8d46e0
Manufacturer: 20
Device: 4016
Status value: 0x600200
Detected flash size: 4MB
```

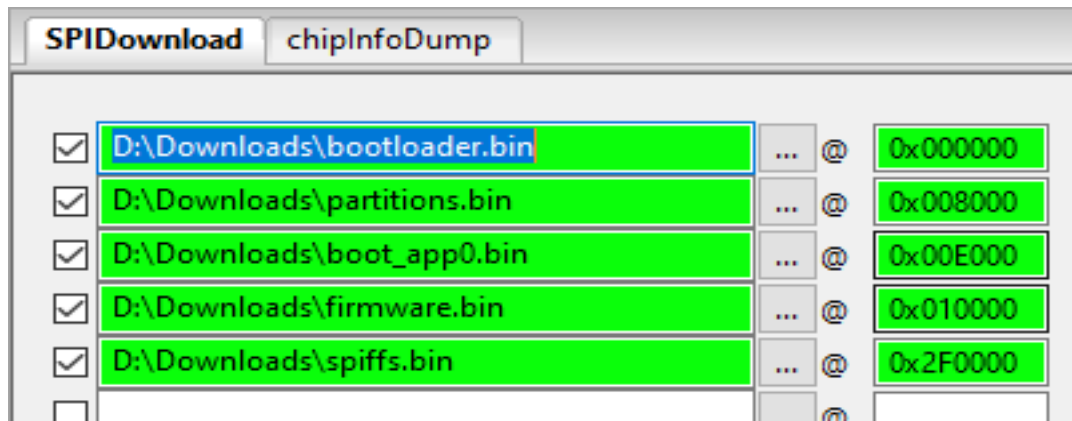
Ga terug naar het tabblad “SPIDownload”.

Kies hier ook de juiste COM: poort (onder in het venster).

Je kunt dan op de “ERASE” button klikken om de hele ESP32 te wissen. Na een paar seconden zie je het blokje met de Chinese tekens blauw worden met de tekst “FINISH”.

Vul nu de 5 files die je hebt gedownload in boven in beeld. Dat kun je doen door op de knop met de 3 puntjes te klikken en de file uit je download directory te selecteren.

Vul voor elke file aan de rechterkant het ESP32 adres waar de data naar toe moet. Vink 5 keer het selectievakje links van de filenaam aan. Het resultaat ziet er dan bijvoorbeeld zo uit:



Druk vervolgens op de “START”-knop en wacht weer tot het blokje met de Chinese tekens blauw worden met de tekst “FINISH”.

Na een reset van de ESP32 moet de firmware lopen. Dat zie je aan het wit-groen flitsen van de NEOpixel LED.