

Bild: Thorsten Hübner

# OpenDTU einrichten

## Hoymiles-Wechselrichter für Balkonkraftwerke per Web & MQTT überwachen

**Wer die integrierte Ertragsüberwachung von Hoymiles-Wechselrichtern einsetzen möchte, muss 160 bis 300 Euro extra für die DTU getauften Gateways ausgeben. Günstiger und ohne Cloud gehts mit der Bastelalternative OpenDTU.**

Von Andrijan Möcker

**E**ine kleine Photovoltaikanlage zu verdrahten und in Betrieb zu nehmen, kann schon viel Spaß machen. Doch die echte Freude kommt erst auf, wenn die Leistungswerte auf einem Bildschirm landen. Deshalb gehören nicht nur Panels und Wechselrichter zu einer PV-Installation, sondern auch die Ertragsüberwachung. Wer seine Anlage normgerecht festverdrahtet oder mit Wieland-Stecker anschließen möchte, kann aber nicht einfach eine günstige Schuko-Messsteckdose [1] verwenden.

Die meisten Hersteller bieten für ihre Mikrowechselrichter eine Monitoring-An-

wendung. Auch Hoymiles hat seine Wechselrichter der HM-Serie mit einer selbstgestrickten 2,4-GHz-Funkschnittstelle ausgestattet und verlangt für das DTU Lite genannte Gateway 160 bis 220 Euro. Möchte man mehr als vier Panels verwalten, muss man sogar das große DTU Pro für 250 bis 300 Euro beschaffen.

Eine breite Entwicklergemeinschaft analysierte das Hoymiles-Protokoll, woraus dann „OpenDTU“ entstand. Es läuft auf dem günstigen WLAN-Mikrocontroller ESP32 und benötigt für die Kommunikation mit dem Wechselrichter lediglich ein direkt anzuschließendes Nordic-Funk-

modul. Im Idealfall bestellt man für Photovoltaik-Freunde mit und zahlt so nur 10 bis 20 Euro pro ESP32-Funkmodul-Kombination, da größere Sets mit etwa drei oder fünf Chips günstiger sind.

OpenDTU richtet sich primär an Selbsterbauer; es liefert sowohl ein einfaches Webinterface als auch das Telemetrieprotokoll MQTT, sodass man die Daten in beliebige MQTT- und JSON-fähige Smart-Home-Steuerungen integrieren kann [2]. Programmier- oder Kommandozeilenkenntnisse sind für OpenDTU nicht nötig. Aktuell spricht die Firmware mit Wechselrichtern der HM-Serie bis 1500 Watt; andere Serien, etwa die HMT, nutzen ein anderes Funkmodul und sind (noch) nicht kompatibel.

## Werkzeug & Einkauf

Der verwendete ESP32-WLAN-Mikrocontroller ist in mehreren Varianten auf fertigen Bastelplatinen erhältlich: Wollen Sie völlig ohne Werkzeug auskommen, bestellen Sie eine mit vorgelöteten Pins – etwa das NodeMCU ESP32. Dann genügen Weiblich-Weiblich-Jumper-Kabel, um das Nordic-Funkmodul mit dem ESP32 zu verbinden.

Kompaktere Varianten wie der ESP32 D1 Mini kommen normalerweise nicht mit vorgelöteten Pins, sodass der Einsatz eines Lötkolbens unerlässlich ist. Wollen Sie die Jumperkabel direkt anlöten, benötigen Sie zudem eine feine Abisolierzange.

Wer sein Netzwerk lieber verkabelt mag, findet bei Olimex (EU, Bulgarien) außerdem ESP32-Platinen mit Ethernetanschluss, sogar eine PoE-Variante existiert. Links zu den Platinchen finden Sie über [ct.de/yru8](http://ct.de/yru8).

Der benötigte Nordic nRF24L01+ ist ebenfalls in mehreren Varianten erhältlich: eine auf kleiner Platine mit Leiterbahnantenne und eine mit SMA-Anschluss und externer Antenne. Zweitere

ergibt Sinn, wenn zwischen OpenDTU-Standort im Innenraum und dem Wechselrichter schirmende Wände oder größere Distanzen (30 Meter oder mehr) zu überbrücken sind. Im Zweifel nehmen Sie eine Platine mit Antennenanschluss. Obendrein müssen Sie beim Kauf auf das Plus in der Chipbezeichnung achten. Der nRF24L01 – ohne Plus – ist nicht OpenDTU-tauglich.

Davon abgesehen benötigen Sie einen Computer mit aktuellem Linux, macOS oder Windows, WLAN und einem USB-Port sowie ein Micro-USB-Kabel. Nach der Programmierung am Rechner schließt man den ESP32 an ein USB-Netzteil an, das mindestens 1 Ampere liefert.

## Installation

OpenDTU muss kompiliert und auf den ESP32 geflasht werden. Am einfachsten gelingt das aus der kostenlosen Entwicklungsumgebung Microsoft Visual Studio Code heraus mit der PlatformIO-Integration – ein Programmierwerkzeug, das hardwarenahe Programmierung erleichtert. Programmierkenntnisse sind nicht nötig, PlatformIO übernimmt sämtliche Schritte für Sie. Wir beschränken uns deshalb auf diesen Installationsweg; das Installieren per Kommandozeile ist im GitHub-Repository beschrieben (alle Softwarelinks unter [ct.de/yru8](http://ct.de/yru8)).

Installieren Sie zunächst Visual Studio Code auf Ihrem Computer; das Paket gibts auf der Downloadseite. Anschließend öffnen Sie das Programm und wechseln links in das Erweiterungs Menü (Extensions). Bietet Ihnen Visual Studio Code das „German Language Pack for Visual Studio Code“ an, können Sie ablehnen; das ist optional. Tippen Sie „PlatformIO“ ins Suchfeld und installieren Sie die PlatformIO IDE. Das kann ein wenig dauern; den Installationsfortschritt können Sie in der blauen Leiste unten beobachten.

## ct kompakt

- Hoymiles-Wechselrichter haben eine proprietäre Funkschnittstelle zum Auslesen der Leistungsdaten.
- Die Gateways des Herstellers sind insbesondere für kleine Anlagen sehr teuer.
- Mit dem Open-Source-Projekt OpenDTU kann man die Wechselrichter der HM-Serie für nicht einmal 20 Euro auslesen.

Die Versionsverwaltung git benötigen Sie ebenfalls, um den Quellcode für OpenDTU herunterzuladen; die Installationsdateien finden Sie auf der Downloadseite. Nutzen Sie macOS oder eine Linux-Distribution, können Sie git auch über die Kommandozeile installieren: In macOS geht das bequem über den Paketmanager Homebrew [3] via `brew install git`, unter Linux mit dem jeweiligen Paketmanager, bei Debian-Derivaten also beispielsweise `apt install git`. Danach öffnen Sie Visual Studio Code wieder, rufen unter Windows und Linux mit dem Tastengriff Strg und , (Komma) die Einstellungen auf (Mac: Cmd plus Komma), suchen den Parameter „Git: Enabled“ und setzen den Haken.

Um den OpenDTU-Code herunterzuladen, drücken Sie Strg (Cmd)+Umschalt+P für die Befehlssuche, suchen nach `git clone` und bestätigen die Befehlsauswahl mit Enter. Im nachfolgenden Feld tragen Sie `https://github.com/tbncbody/OpenDTU.git` ein, bestätigen (Enter) und wählen einen Zielordner. Nach dem Download fragt Sie Visual Studio Code, ob Sie das Projekt in einem neuen oder im bestehenden Fenster öffnen wollen; in der Regel wollen Sie letzteres (Add to Workspace). PlatformIO fühlt sich durch die

The screenshot shows the 'Live Data' interface for an HM-700 inverter. It displays three data columns: Phase 1, String 1, and String 2. Each column contains a table of properties with their values and units.

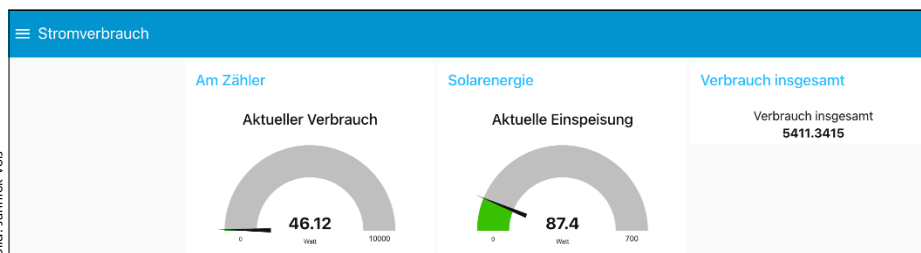
Property	Value	Unit
Power	8.70	W
Voltage	229.00	V
Current	0.04	A
Power DC	9.10	W
YieldDay	1.902.00	Wh
YieldTotal	28.29	kWh
Frequency	50.02	Hz
Temperature	20.80	°C
ReactivePower	0.00	var
Efficiency	95.60	%

Property	Value	Unit
Power	4.50	W
Voltage	25.60	V
Current	0.18	A
YieldDay	904.00	Wh
YieldTotal	13.79	kWh
Irradiation	0.00	%

Property	Value	Unit
Power	4.60	W
Voltage	26.00	V
Current	0.18	A
YieldDay	998.00	Wh
YieldTotal	14.50	kWh
Irradiation	0.00	%

**Bis zu zehn Wechselrichter fragt OpenDTU per Funk ab. Das Webinterface läuft direkt auf dem ESP32, sodass die Firmware auch ganz ohne Smart-Home-Zentrale eine Hilfe ist – sofern Ihnen nackte Zahlen ohne Diagramme und andere grafische Schmankerl genügen.**

Bild: Jannick Vogt



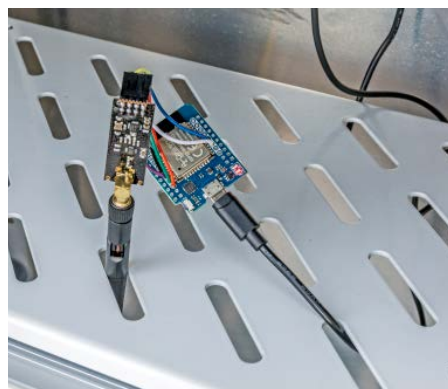
### Die Node-Red-Erweiterung „Node-Red-Dashboard“ erlaubt, Leistungswerte und andere Messdaten eines Balkonkraftwerks einfach zu visualisieren.

Datei `platformio.ini` im Ordner direkt als Compiler zuständig und fragt, ob Sie den Autoren vertrauen, was Sie bestätigen. Jetzt steht nur noch an, die Datei `platformio.ini` zu bearbeiten. Die Datei wird automatisch in einer neuen Registerkarte oben in Visual Studio Code geöffnet.

### Flashen

Da die optimalen GPIOs (General Purpose Input/Output, Mikrocontrolleranschlüsse, ausgeführt als Pins) für das nRF-Modul zwischen verschiedenen ESP32-Platinen variieren, haben die Entwickler direkt mehrere GPIO-Belegungen als Umgebungsvariablen in der `platformio.ini` hinterlegt. Der Parameter `default_envs` sagt dem Compiler, welche er wählen soll. Mit `generic` sind die im Repository gezeigten Pins gültig (siehe [ct.de/yru8](http://ct.de/yru8)). Zusätzlich gibt es die Olimex-Platinen (`olimex_esp32_poe` und `olimex_esp32_evb`) und den Wemos D1 Mini ESP32 (`d1_mini_esp32`). Ersetzen Sie `generic` wenn nötig durch eine der Bezeichnungen.

Haben Sie ein anderes Board gekauft, können Sie die `build_flags` mit den GPIO-



**Unser nackter OpenDTU-Prototyp fand im Experimentierschrank auf dem Dach des Verlags seinen Platz, in dem auch Balkonkraftwerke zum Test angeschlossen sind. Im Versuchsbetrieb mit unserem Hoymiles HM-700 lief die Firmware ohne Abstürze.**

Definitionen aus einer anderen Variante kopieren, unter `[env:generic]` anfügen und entsprechend Ihres Platinchens ändern. Beispielsweise:

```
[env:generic]
board = esp32dev
monitor_port = COM4
upload_port = COM4
build_flags = ${env.build_flags}
-DHOYMILES_PIN_MISO=12
-DHOYMILES_PIN_MOSI=2
-DHOYMILES_PIN_SCLK=14
-DHOYMILES_PIN_IRQ=13
-DHOYMILES_PIN_CE=16
-DHOYMILES_PIN_CS=5
```

Vor dem Flashen muss nur noch der korrekte serielle Port in der `platformio.ini` eingetragen werden: Um den zu ermitteln, verbinden Sie den ESP32 per USB mit Ihrem Computer, öffnen dann unter Windows den Geräte-Manager und schauen unter Anschlüsse (COM & LPT) nach der Portnummer. In Linux und macOS öffnen Sie die Kommandozeile; Linuxer geben `ls /dev/tty*` ein, Apple-Nutzer `ls /dev/cu.*`. Gesucht wird `ttyUSBn` oder `ttyACMn` beziehungsweise `cu.usbserial-`(Hardware-ID).

Haben Sie die korrekte Portnummer beziehungsweise Adresse ermittelt, tragen Sie sie in der `platformio.ini` ein. Scrollen Sie dazu bis zum korrekten `env`-Eintrag (`generic`, `olimex`, ...) und ändern Sie `monitor_port` und `upload_port`. In macOS und Linux muss der vollständige Pfad eingetragen werden, also etwa `/dev/ttyUSB1`.

Um den Kompilier- und Flashprozess zu starten, klicken Sie unten in der blauen Leiste auf den Rechtspfeil. Der Prozess kann einige Minuten dauern, da PlatformIO unter Umständen erst zusätzliche Daten für den ESP32-Compiler sowie Bibliotheken herunterladen muss. Den Fortschritt können Sie unten in der Konsole beobachten.

Ist der Prozess abgeschlossen, entfernen Sie den ESP32 vom USB-Anschluss

und beginnen damit, das Funkmodul anhand der GPIO-Definitionen in der `platformio.ini` anzuschließen. Achtung: Die Pin-Nummern auf der ESP-Platine entsprechen nicht zwangsläufig den GPIOs; Datenblätter liefert der Händler oder der Hersteller. Das nRF-Funkmodul ist oft unbeschriftet; Angebote enthalten aber meist eine Grafik, anhand derer die Belegung ersichtlich ist. Seien Sie besonders vorsichtig, wenn Sie den Spannungspin Vcc anschließen; er darf ausschließlich auf einen 3,3-Volt-Pin gesteckt werden. 5 Volt zerstören den Chip möglicherweise.

Ist alles zusammengesteckt, verbinden Sie das Konstrukt mit dem USB-Netzteil.

### Einrichtung

OpenDTU macht die Einrichtung leicht: Die Firmware startet – solange die Zugangsdaten zu einem WLAN fehlen, mit dem sie sich verbinden soll – einen WLAN-Access-Point, über den Sie das Webinterface erreichen. Suchen Sie das OpenDTU-WLAN und verbinden Sie sich mit dem Passwort „openDTU42“. Ignorieren Sie gegebenenfalls die Warnung, dass keine Internetverbindung möglich ist; die ist jetzt nicht wichtig.

Das Webinterface öffnen Sie anschließend mit einem beliebigen Browser über `http://192.168.4.1`. Über „Settings / Network“ erreichen Sie die WLAN-Einstellungen. Tragen Sie dort die SSID – also den Namen Ihres heimischen Funknetzes – und den Schlüssel ein. Optional passen Sie den Hostnamen an. Klicken Sie „Save“, verbindet sich OpenDTU mit Ihrem WLAN. Die neue IP-Adresse ermitteln Sie etwa über die Statusseite Ihres Routers oder mit einem beliebigen IP-Scanner; auf Smartphones gelingt das mit der kostenlosen Basisversion von Fing besonders einfach. Taucht das Gerät nicht auf, haben Sie sich möglicherweise vertippt. Suchen Sie dann erneut nach dem OpenDTU-WLAN und rufen Sie die Konfiguration wieder über `http://192.168.4.1` auf.

### Auswertung

Haben Sie die neue IP-Adresse entdeckt, geben Sie sie im Browser ein, um das Webinterface zu öffnen. Jetzt müssen Sie OpenDTU nur noch die Seriennummer des Wechselrichters mitteilen, damit es mit dem Wechselrichter kommunizieren kann: In „Settings/Inverter Settings“ tragen Sie diese in Serial ein und vergeben einen beliebigen Namen für das Gerät. Sie



können danach bis zu neun weitere Wechselrichter hinzufügen.

Anschließend wechseln Sie in die Registerkarte „Live Data“. OpenDTU fängt sofort an, die Wechselrichter abzufragen und die Leistungsdaten sollten sofort auftauchen, sofern der Wechselrichter in Reichweite ist. Sehen Sie nichts, schauen Sie zunächst unter System/Info/Chip Status; steht er nicht auf „Connected“, kontrollieren Sie die Verbindung zum nRF-Funkmodul. Alternativ wechseln Sie in Settings/DTU Settings und setzen das Verstärkerlevel (PA Level) auf „Maximum“.

Sobald erste Werte in Live Data zu sehen sind, ist die Grundeinrichtung abgeschlossen. Das Webinterface aktualisiert die Werte, ohne dass Sie die Seite neu laden müssen. Genügt Ihnen die Aktualisierungsrate nicht, können Sie auch in den DTU Settings das Abfrageintervall herabsetzen. Kommen die Daten weiter selten, ist das Signal vom Wechselrichter wahrscheinlich schwach und Sie müssen das

Konstrukt näher heranbewegen. Viele abzufragende Wechselrichter sorgen ebenso für höhere Latenz.

### Bunte Auswertung

Die einfache Auswertung auf der Live-Data-Seite dürfte jedem genügen, der einfach nur gelegentlich seine Gesamtersparnisse und den Amortisationsgrad des Balkonkraftwerks ausrechnen möchte. „YieldTotal“ zeigt die Produktion des Wechselrichters ab Inbetriebnahme an; der Wert ist persistent und bleibt auch bei abgeschaltetem Wechselrichter erhalten.

Wollen Sie es etwas schicker haben, kommt OpenDTUs MQTT-Integration ins Spiel: Über das Telemetrieprotokoll wirft die Firmware sowohl die Informationen von der Wechselspannungsseite als auch die der einzelnen Gleichspannungseingänge aus, sodass Sie sogar jedes einzelne Panel visualisieren können.

Ganz einfach und ohne komplizierte Datenbanken visualisiert man die Werte mit Node-Red-Dashboard [4, 5]. Ausge-

fuchstere Darstellungen liefert Grafana in Verbindung mit dem Datenbankdienst InfluxDB [6, 7]. Zusammen mit einem Zähleradapter bekommen Sie so den perfekten Überblick über Stromverbrauch und Produktion [8]. (amo@ct.de) **ct**

### Literatur

- [1] Ernst Ahlers, Stromwaagen, 10 Energiekostenmessgeräte im Vergleich, c't 17/2022, S. 98
- [2] Jan Mahn, Weltsprache, Das Protokoll MQTT für robusten Datenaustausch in Industrie und Hausautomation, c't 6/2018, S. 164
- [3] Immo Junghärtchen, App Store fürs Terminal, Die Paketverwaltung Homebrew unter macOS einrichten und nutzen, c't 12/2020, S. 122
- [4] Jan Mahn, Reaktionsmaschine, Einsteig in Heimautomation mit Node-Red, c't 15/2018, S. 142
- [5] Merlin Schumacher, Rotfront!, Grafische Oberflächen für Node-Red entwickeln, c't 2/2019, S. 160
- [6] Jan Mahn, Das Auge administriert mit, Daten visualisieren mit Grafana, c't 10/2019, S. 152
- [7] Jan Mahn, Geschichtsschreiber, InfluxDB: Spezialisierte Datenbank für Messwerte und Logging, c't 5/2019, S. 154
- [8] Andrijan Möcker, Solarmeter, Balkonkraftwerke: Stromproduktion messen, c't 19/2022, S. 108

**Software und Platinen:** [ct.de/yru8](https://www.ct.de/yru8)