

How to add Fronius Wattpilot via OCPP

Connect the Fronius Wattpilot wallbox to the Reduxi Controller via OCPP 1.6J, so that the Controller can monitor and control charging with PV surplus and load management.

🕒 approx. 15–20 min

🔧 Fronius Wattpilot · Reduxi Controller

🌐 OCPP 1.6J (LAN / Wi-Fi via router)

📱 Fronius Solar.wattpilot app



Fronius Wattpilot – wall-mounted version



Fronius Wattpilot – portable version with CEE plug

IN THIS ARTICLE

1. Technical documentation
2. Required equipment
3. System overview
4. Configure OCPP in the Wattpilot app
 - Enable OCPP
 - OCPP server settings
 - Further OCPP options
5. Log in to the Reduxi web interface
 - Remote via the Reduxi app (tunnel)
 - Locally via a web browser
6. Add the Wattpilot to Reduxi
 - Basic settings
 - Communication settings
 - Test communication
7. Check the readings
8. Summary
9. Common issues
10. Related articles

1 Technical documentation

The **Fronius Wattpilot** supports **OCPP 1.6J** (Open Charge Point Protocol, JSON over WebSocket). In this setup the **Reduxi Controller acts as the local OCPP server** (central system) and the Wattpilot connects to it as a charge point – no cloud service is needed.

PROTOCOL

OCPP 1.6J

CONNECTION

WebSocket `ws://`
(local)

NETWORK

LAN / Wi-Fi via
router

DEVICE IDENTIFIER

Wattpilot serial
number

SMART CHARGING

Charging profiles in
A

PHASE SWITCHING

3 → 1 phase
supported

Manufacturer documents:

- [Fronius Wattpilot OCPP – User Information](#) (42,0410,2881) – supported OCPP messages, configuration parameters and activation in the app
- [Open Charge Alliance](#) – OCPP 1.6 specification



Supported OCPP functions

Authorize, BootNotification, Heartbeat, MeterValues, Start/StopTransaction, StatusNotification as well as RemoteStart/Stop, SetChargingProfile, ClearChargingProfile, TriggerMessage, UnlockConnector, Reset and UpdateFirmware. Feature profiles: Core, FirmwareManagement, SmartCharging, RemoteTrigger.

2 Required equipment

- ☐ Fronius Wattpilot, installed and commissioned by a qualified electrician
- ☐ Wattpilot connected to the same local network as the Reduxi Controller (Wi-Fi or LAN via router)
- ☐ Smartphone with the **Fronius Solar.wattpilot** app, connected to the Wattpilot
- ☐ Reduxi Controller with a known, **fixed IP address** (e.g. `192.168.31.73`) – shown in the Reduxi Configurator as *Controller OCPP JSON 1.6 URL* when you add the device, or in the router
- ☐ Serial number of the Wattpilot (shown in the app and on the type plate, e.g. `91099800`)
- ☐ Laptop or tablet with internet access for the Reduxi Configurator



Fixed IP address for the Controller

The Wattpilot connects to the Controller's IP address. If the router assigns a new address, the OCPP connection is lost. Reserve the IP of the Reduxi Controller in the router (DHCP reservation).

3 System overview

The Wattpilot opens a WebSocket connection to the Reduxi Controller and identifies itself with its serial number. The Controller assigns the connection to the device with the same **device identifier** and then reads measurements and sends charging limits.

COMPONENT	ROLE	ADDRESS / SETTING
Reduxi Controller	OCPP server (central system)	<code>ws://<Controller IP>/ocpp/1.6/json</code>
Fronius Wattpilot	OCPP client (charge point)	Connects to the URL above + its serial number
Full URL used by the Wattpilot	–	<code>ws://192.168.31.73/ocpp/1.6/json/91099800</code>



Unencrypted connection in the local network

For a local server the Wattpilot allows `ws://` (unencrypted) or `wss://`. The Reduxi Controller uses `ws://`. External OCPP servers always require `wss://`.

4 Configure OCPP in the Wattpilot app

4.1 Enable OCPP

1

Open the app

Start the **Fronius Solar.wattpilot** app and connect to the Wattpilot.

2

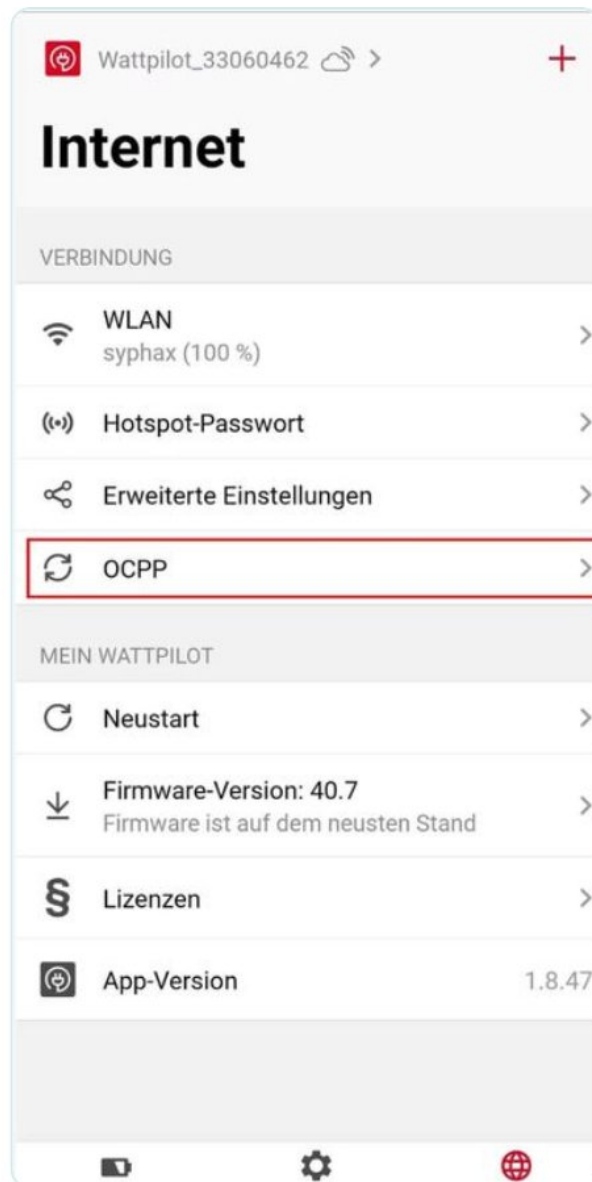
Open the OCPP settings

Open the **Internet** tab and tap **OCPP** (in newer app versions: **Einstellungen** → **OCPP**).

3

Enable OCPP 1.6J

Switch on **OCPP1.6J**. OCPP automatically activates access management on the Wattpilot.



Internet menu with the OCPP entry (Fronius Solar.wattpilot app)

4.2 OCPP server settings

Tap **OCPP-Server** and enter the following values:

FIELD (DE / EN)	VALUE
Protokoll / Protocol	WS
Benutzername / User	empty
Passwort / Password	empty
Host	IP address of the Reduxi Controller, e.g. 192.168.31.73
Pfad / Path	/ocpp/1.6/json/
SN zur URL hinzufügen / Add SN to URL	On – mandatory



"SN zur URL hinzufügen / Add SN to URL" must be switched on

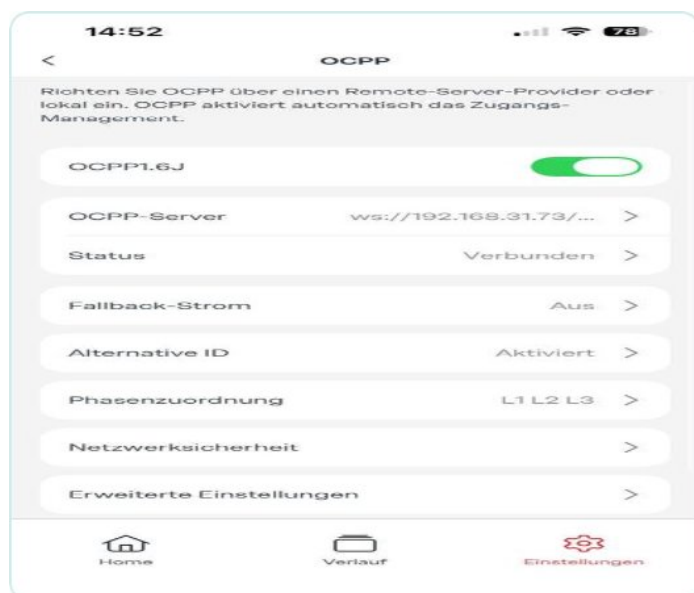
Without this option the Wattpilot does not add its serial number to the URL, the Controller cannot identify it and **no communication is established**. Always check that the switch is green before saving.

The app then shows the complete address, e.g. `ws://192.168.31.73/ocpp/1.6/json/91099800`. Tap **Änderungen speichern / Save**. The Controller URL is also shown in the Reduxi Configurator (see [Point 6.2](#)).

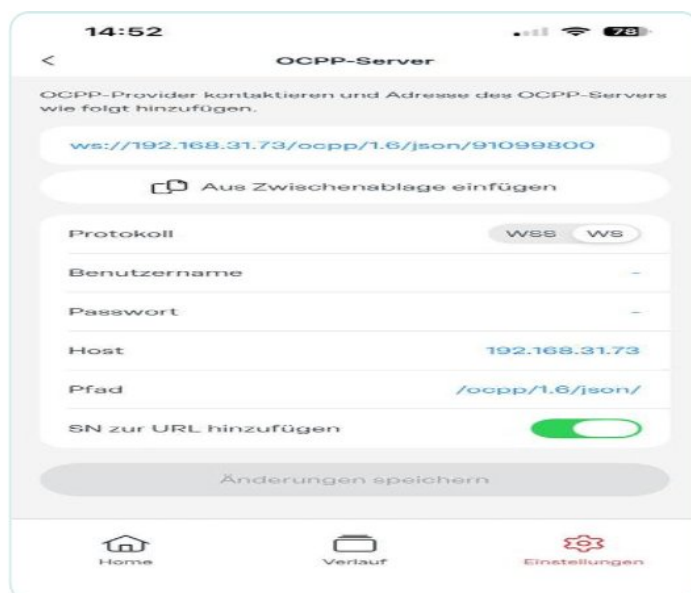


Status not "Connected" yet? That is expected

The Wattpilot is only accepted once it has been added in Reduxi with its serial number. Continue with [Point 5](#) and [Point 6](#) – afterwards the status changes to **Verbunden / Connected**. Tip: if you don't know the Controller IP yet, do [Point 6](#) first and copy the address from the field *Controller OCPP JSON 1.6 URL*.



OCPP menu: OCPP1.6J enabled, status "Verbunden" (connected)



OCPP server: protocol WS, host = Controller IP, path /ocpp/1.6/json/, SN added to URL

4.3 Further OCPP options

OPTION	RECOMMENDATION
Status	Shows <i>Not connected</i> , <i>Started</i> , <i>Connected</i> or <i>Connected and accepted</i> . After adding the device in Reduxi it must show Verbunden / Connected .
Fallback-Strom / Fallback current	Charging current the Watto pilot uses if the OCPP connection to the Controller is lost. Set a value that is safe for the house connection.
Phasenzuordnung / Phase assignment	L1 L2 L3 right phase.
Alternative ID, Netzwerksicherheit, Erweiterte Einstellungen	Leave unchanged.

5 Log in to the Reduxi web interface

The Reduxi Configurator can be opened **remotely via the tunnel connection** in the Reduxi app or **locally** in a web browser in the same network. Both lead to the same web interface.



Reduxi app
Via tunnel connection · works from anywhere

REMOTE



Web browser
In the local network · via IP address or reduxि.local

5.1 Remote via the Reduxi app (tunnel)

1 Open the app and log in

Launch the Reduxi app on your smartphone or tablet and log in with your credentials.

2 Open Reduxi devices

Tap the menu icon (three lines, top left) and select **Reduxi devices**.

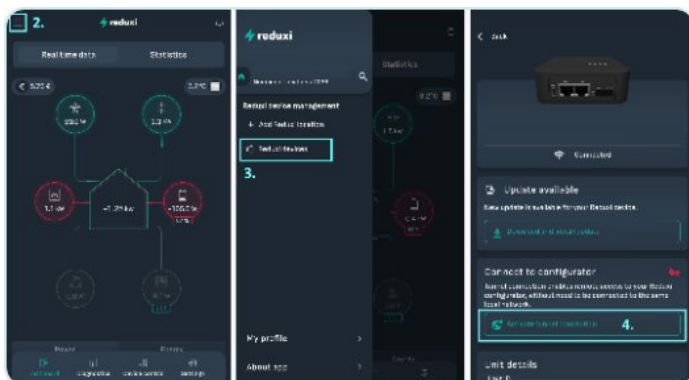
3 Activate the tunnel connection

Switch on the tunnel connection and wait a moment until it is active.

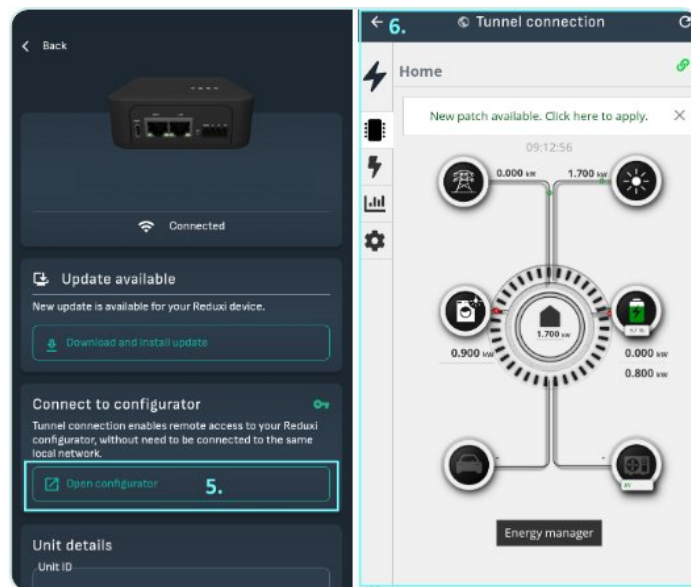
4

Open the configurator

Tap **Open configurator**. The Reduxi web interface opens.



Activate the tunnel connection



Open configurator

5.2 Locally via a web browser

1

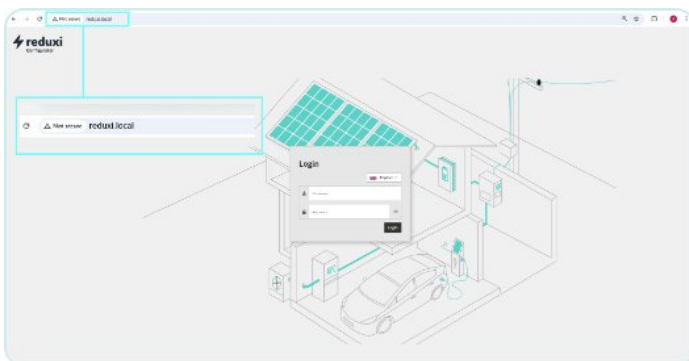
Open the login page

On a device in the same network, open a web browser and enter the Controller's IP address (e.g. <http://192.168.31.73>) or <http://reduxi.local>.

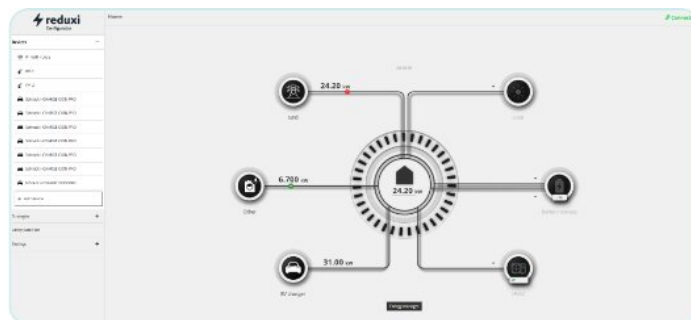
2

Log in

Username: **admin** · Password: **admin** (factory default).



Reduxi Controller login screen



Web interface after login



The Watto pilot always connects locally

Regardless of how you open the Configurator, the Watto pilot itself always connects to the Controller in the local network via `ws://<Controller IP>/ocpp/1.6/json`.

6 Add the Watto pilot to Reduxi

6.1 Basic settings

1

Add a new device

In the left menu under **Devices** click **+ Add device**.

2

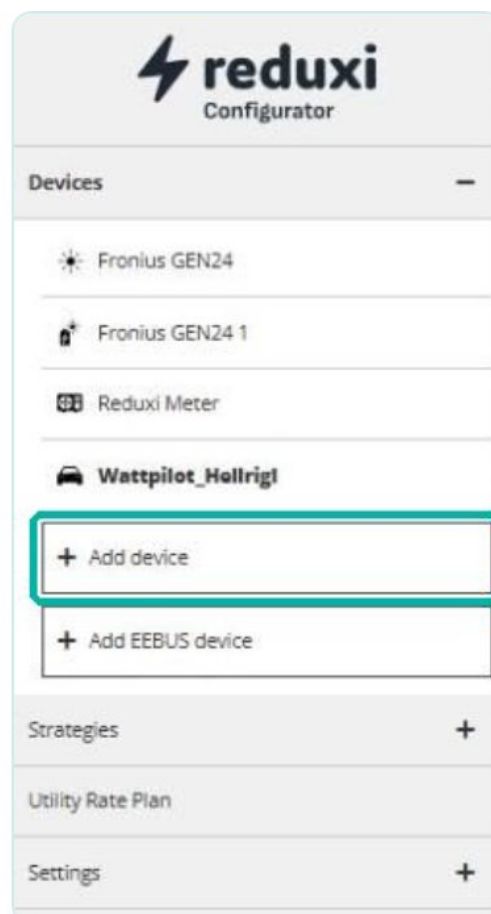
Select type

Device type: **EV charger** · Type: **OCPP 1.6 - JSON**.

3

Name and strategy

Enter a **Name** (e.g. **Fronius Watto pilot**). Under **Include to strategy** keep **Energy manager** → **Regulated devices** ticked, then click **Next**.



Click **+ Add device** in the left menu

Basic settings

Device type *

EV charger

Type *

OCPP 1.6 - JSON

Name *

Fronius Wattpilot

Include to strategy

Strategy	Measurement devices	Regulated devices
Energy manager	☐	☒

Back [How to configure?](#) Next

Device type EV charger, type OCPP 1.6 – JSON, name and strategy

6.2 Communication settings

FIELD	VALUE
Device identifier	Serial number of the Wattpilot , e.g. <code>91099800</code> – exactly the number at the end of the URL in the app
Connector	<code>1</code>
Charger type	<code>AC</code>
Energy counter	<code>Default</code>
Allocate power to EVSE when EV is not connected	<code>Default</code>
Per phase max. current	<code>Auto detect from EVSE</code> or <code>Manual override</code> with the fused current, e.g. <code>32</code> A (22 kW) or <code>16</code> A (11 kW)
Forward communication / username / password / measurements to backend OCPP server	empty – only needed if an external OCPP backend is used in parallel
Allow new charging session	<code>Always</code>
Controller OCPP JSON 1.6 URL	read only, e.g. <code>ws://192.168.31.73/ocpp/1.6/json</code> – this is the address entered in the app (see Point 4.2)



Device identifier = serial number of the Wattpilot

Enter the **serial number** of the Wattpilot as **Device identifier** – it is shown in the app header (e.g. `Wattpilot_91099800` → `91099800`) and at the end of the OCPP URL. Without the correct identifier the Controller cannot assign the incoming OCPP connection and shows no data.

Communication channel and device configuration

Device identifier *

91099800

Connector *

1

Charger type *

AC

Energy counter

Default

Allocate power to EVSE when EV is not connected to EVSE

Default

Per phase max. current *

Manual override

Per phase max. current manual override

32

In effect when manual override is selected

Forward communication to backend OCPP server

Device identifier = serial number of the Wattpilot, connector 1, AC

Check the values and click **Done** to save the device.

Forward measurements to backend OCPP server once every

Allow new charging session

Always

Controller OCPP JSON 1.6 URL

ws://192.168.31.73/ocpp/1.6/json

Back

[How to configure?](#)

Done

Controller OCPP JSON 1.6 URL – click Done to save the device

6.3 Test communication

With OCPP the Wattpilot connects to the Controller by itself – no IP address of the Wattpilot is needed. Open the device again (**Devices** → **[Wattpilot]** → **Details**) and click **Test communication** at the bottom right.

i “Unable to communicate with device” is normal for OCPP

With OCPP the Controller cannot actively ping the charger – the Wattpilot connects to the Controller by itself. The red line in the first box can therefore be ignored. What matters is the box below: **Number of devices found** must be at least **1** and the **Device identifier** must show the serial number of the Wattpilot. The final proof are the live values under **Last measurements**.



Test communication: the window lists the chargers connected via OCPP and their device identifiers

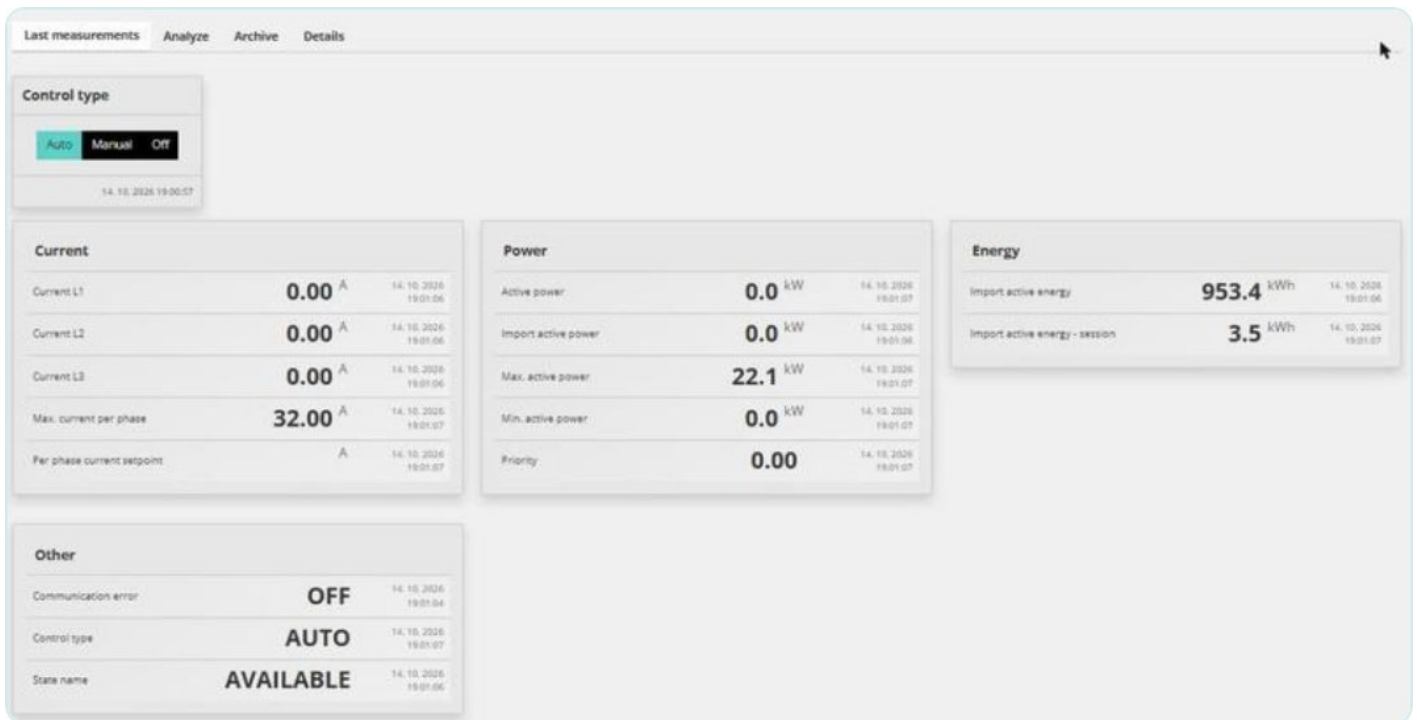
- **Number of devices found on OCPP – JSON 1.6** shows how many chargers are connected to the Controller via OCPP.
- **Device identifier** must show the serial number of the Wattpilot (e.g. **91099800**). If it does not appear, the Wattpilot is not connected – check the app settings (see [Point 4.2](#)).
- In the **Wattpilot app** the OCPP status must show **Verbunden / Connected**.
- Final check: live values and **Communication error** = **OFF** (see [Point 7](#)).

✓ Connection established

If both sides show a connection, the Wattpilot is ready. The Reduxi Controller now controls the charging current according to the active strategy.

7 Check the readings

Open **Devices** → **[Wattpilot]** → **Last measurements**:



Last measurements: control type, currents per phase, power, energy and charger state

VALUE	MEANING
Control type	Auto = controlled by the Reduxi strategy · Manual · Off
Current L1 / L2 / L3	Charging current per phase
Max. current per phase	Maximum current of the charger (e.g. 32 A)
Active power / Max. active power	Current charging power and maximum power (e.g. 22.1 kW)
Import active energy / session	Total charged energy and energy of the current session
Communication error	Must be OFF
State name	Current state of the charger, e.g. AVAILABLE when no car is connected



Summary

Preparation (Points 1–3): Wattpilot and Reduxi Controller in the same network, Controller with fixed IP, serial number of the Wattpilot at hand.

Wattpilot app (Point 4): enable `OCPP1.6J` , protocol `WS` , host = Controller IP, path `/ocpp/1.6/json/` , **“SN zur URL hinzufügen” on (mandatory)**.

Add to Reduxi (Points 5–6): EV charger · `OCPP 1.6 - JSON` , device identifier = serial number, connector `1` , AC.

Go live (Point 7): app shows “Connected”, Reduxi shows live values and communication error `OFF` .



Common issues

ISSUE	SOLUTION
App status stays “Started” / “Not connected”	Check that “SN zur URL hinzufügen” is on, then host (Controller IP), protocol <code>WS</code> (not WSS), path <code>/ocpp/1.6/json/</code> and that both devices are in the same network.
App shows “Connected”, but Reduxi shows no data	The device identifier in Reduxi must match the serial number at the end of the URL exactly (e.g. <code>91099800</code>). “SN zur URL hinzufügen” must be on.
Connection lost after a router restart	The Controller got a new IP address. Reserve a fixed IP in the router and update the host in the app.
Unwanted charging behaviour when the Controller is offline	Set a Fallback current in the app’s OCPP menu (see Point 4.3).
Load management reduces the wrong phase	Check the phase assignment in the app and in Reduxi (Advanced configuration → “L1/L2/L3 is connected to main feed”).
Car does not start charging	OCPP activates access management – check Allow new charging session = <code>Always</code> in Reduxi.



Related articles

Matching guides from the Reduxi knowledge base:



Fronius Wattpilot OCPP documentation

OCPP user information from Fronius



Device setup guides

All step-by-step guides for connecting devices to the Reduxi Controller



List of supported devices

All devices that work with the Reduxi Controller



Strategies for the Reduxi Controller

Energy manager and other strategies explained