



# USER GUIDE

*Wattmon Hardware and Software User Manual*

## Copyright

© Cynergy - A Unit of ADPS Trust, Auroville Foundation, 2025. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Cynergy.

## Trademarks and Permissions



and other Wattmon trade names are trademarks of Cynergy. All other trademarks and trade names mentioned in this document are the property of their respective holders.

## Notice

The purchased products, services and features are stipulated by the contract made between Cynergy and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the sales contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

## Cynergy – A Unit of ADPS Trust, Auroville Foundation

Address: F-02, UTSAV  
Auroville 605 101  
Tamil Nadu, India.

Website: [www.wattmon.com](http://www.wattmon.com)

Email: [support@wattmon.com](mailto:support@wattmon.com)

# About This Document

## Purpose

This document describes the Wattmon ecosystem, their hardware and software, for the explicit purposes of installing, maintaining, and troubleshooting.

It is recommended to understand the core functions and features before using Wattmon.

## Intended Audience

This document is intended for engineers, technicians and any other qualified personnel involved in the installation of photovoltaic (PV) power plants. It is also intended to be a reference guide for personnel who have undergone training in the configuration and troubleshooting of the Wattmon.

## Revision History

<i>Rev. 1.0</i>	WattmonOS v2.15
<i>Rev. 1.1</i>	Includes WattmonMEGA and WattmonOS v2.23 features
<i>Rev. 1.2</i>	Added WattmonOS v2.25 features
<i>Rev. 1.3</i>	Updated for WattmonOS v2.26
<i>Rev. 1.4</i>	Updated for WattmonOS v2.27 & Modbus Tester
<i>Rev. 1.5</i>	Updated for WattmonOS v2.27 & EZConfig
<i>Rev. 2.0</i>	Major update for WattmonOS v3 & EZConfig Wizard
<i>Rev. 2.1</i>	Updated to an easier-to-read format
<i>Rev. 2.2</i>	Updated for WattmonOS v3.5
<i>Rev. 2.3</i>	Updated for WattmonOS v3.6
<i>Rev. 2.4</i>	Updated for WattmonOS v3.7 & WattmonM2
<i>Rev. 2020.09</i>	New Format. Updates for WattmonOS v3.9, EZConfig Wizard, Power Control & WattmonMINI3
<i>Rev 2.6</i>	Updated everything and added support for MEGA2, MEGA2-4G, ULTRA
<i>Rev 2.7</i>	Updated to add DLMS meter documentation

# TABLE OF CONTENTS

About This Document.....	2
Purpose.....	2
Intended Audience.....	2
Revision History.....	2
TABLE OF CONTENTS.....	3
Introduction.....	7
Technical Specifications.....	8
WattmonMINI3 / WattmonMINI3-4G.....	9
WattmonMEGA2 / WattmonMEGA2-4G.....	10
WattmonULTRA.....	10
Connecting Analog Sensors.....	11
Connecting Digital Inputs.....	12
Connecting Digital Outputs.....	12
RS-232 Serial Communication.....	13
RS-485 Communication.....	13
1-Wire Bus.....	14
WattmonM2 (Legacy).....	14
WIRING RECOMMENDATIONS.....	15
RS-485 Cabling.....	15
RS-485 Surge Protection.....	16
RS-232 Connection.....	16
DLMS Connections.....	17
CONNECTING TO YOUR DEVICE.....	18
Connecting in Microsoft Windows.....	18
Resetting your IP Address.....	19
Modifying your IP Address on the SD card.....	19
EZCONFIG - QUICK CONFIGURATION TOOL.....	20
EZConfig Wizard Step 1 / 4: Choose Devices and Configuration.....	20
Modbus RTU Channel 1.....	20
Modbus RTU 2.....	21
Modbus TCP.....	21
Power Meters.....	22
Inverters.....	23
Other Devices.....	23
Additional Packages.....	24
Additional Options.....	24
EZConfig Wizard Step 2 / 4: Choose Data Points to Log.....	26
EZConfig Wizard Step 3 / 4: Review Your Configuration.....	28
EZConfig Wizard Step 4 / 4: Implementation.....	30

UNDERSTANDING THE WATTMON OPERATING SYSTEM.....	31
WATTMON OS INTERFACE.....	31
FILE MANAGER.....	32
Uploading a File.....	33
Uncompressing a File.....	33
Editing a File using the File Editor.....	34
Blockly Visual Editor.....	34
Understanding the File System.....	35
/config/ Folder.....	35
/app/ Folder.....	36
/scripts/ Folder.....	36
/package/ Folder.....	36
/dev/ Folder.....	36
/ini/ Folder.....	36
/logs/ Folder.....	36
TAIL LOGS AND SYSTEM LOGS.....	36
Internet Connectivity Information.....	37
Troubleshooting Connectivity Issues.....	38
Troubleshooting Reboots.....	39
Miscellaneous Issues.....	39
BASIC CONFIGURATION SETTINGS- CONTROL PANEL.....	40
USERS AND SECURITY.....	40
USB DEVICES / MINI-PCIE DEVICES.....	41
SNMP CONFIGURATION.....	42
Using SNMP.....	42
SNMP Traps.....	43
MODBUS TESTER.....	43
Modbus Parameters Section.....	43
Modbus Test Examples.....	44
CELLULAR SETTINGS.....	46
OTA UPDATES.....	47
DATE AND TIME SETTINGS.....	48
BACKUP DATA.....	49
FLASH MEMORY.....	50
Preparing your Flash Drive.....	51
EMAIL SETTINGS.....	52
ABOUT.....	53
WI-FI SETTINGS.....	54
Wi-Fi Client Configuration.....	54
W-Fi Access Point Configuration.....	54
NETWORK SETUP.....	55

MODBUS COMMUNICATION SETTINGS.....	56
Modbus RTU (RS-485) Settings.....	57
Modbus Channel 2 Settings.....	57
Modbus TCP Settings.....	58
WATTMON APP.....	60
WATTMONOS UPDATER.....	60
Incremental Updates.....	60
Full OS Updates.....	61
Additional Packages.....	61
WIDGETS.....	61
Adding a Widget.....	62
DATA EXPORT.....	63
Custom Server Data Export.....	65
MQTT Export.....	65
Export to PV Output.....	66
Wattmon Reachability Settings.....	66
PACKAGE MANAGER.....	66
DATA COLLECTION.....	67
INTERFACE SETTINGS.....	69
FIRMWARE UPDATER.....	70
GRAPHS.....	71
Configuring a Graph.....	72
SCHEDULED TASKS.....	73
DEVICES.....	74
CONFIGURE DEVICE PAGE.....	75
CALIBRATE DEVICE PAGE.....	76
CALIBRATE WATTMONULTRA.....	77
Voltage Calibration.....	78
Current Calibration.....	78
Digital Inputs.....	78
Digital Outputs.....	78
ROLES PAGE.....	78
ROLE SETTINGS PAGE.....	79
GROUPS.....	81
CONFIGURE GROUP PAGE.....	81
HTTP SERVER SETUP.....	82
ACTION.....	83
ACTION SETTINGS.....	83
Trigger Conditions.....	84
Actions.....	85
DEVICE TYPES PAGE.....	86

DEVICE TYPE EDIT PAGE.....	86
DLMS Settings.....	88
Digital Input Registers.....	89
Digital Output Registers.....	89
Read Only Registers (Modbus).....	90
Read-Only Registers (DLMS).....	91
Read Write Registers.....	92
One-time String Data.....	92
Roles.....	92
UPHP SCRIPT SETUP PAGE.....	93
WATTMONOS PACKAGES.....	96
EZDisplay Package.....	96
Devices Package.....	96
Power Control (PC) Package.....	97
General Settings.....	97
Power Meters.....	100
Inverters.....	102
FTP (File Transfer Protocol) Package.....	103
MQTT2 (Message Queuing Telemetry Transport ) Package.....	105
Automatic Publishing.....	106
Subscribing to Topics.....	107
Alerts Package.....	107
Pyranometer Package.....	109
Appendix 1: Wattmon Naming Conventions.....	110
Troubleshooting your Wattmon.....	115
Wattmon does not Work.....	115
Unable to connect to the Wattmon over Ethernet.....	115
Frequently Asked Questions.....	116

## Introduction

The Wattmon hardware and software platform is highly adaptable and configurable, being used for monitoring Grid-Tie, Hybrid and Off-Grid setups, Solar Water Pumping and Building Loads. It features a widely compliant Zero Feed-In and Generator Protection solution compatible with all the leading inverter brands.

This guide aims to take you through the Wattmon data logger's in-built web server, providing you with the basic know-how to configure it for your specific application. For more information about the data loggers, visit the Wattmon Documentation Wiki.

*(Note: There are various models of Wattmon with differing hardware specifications and not all information is relevant to each model. This will be indicated when appropriate.)*

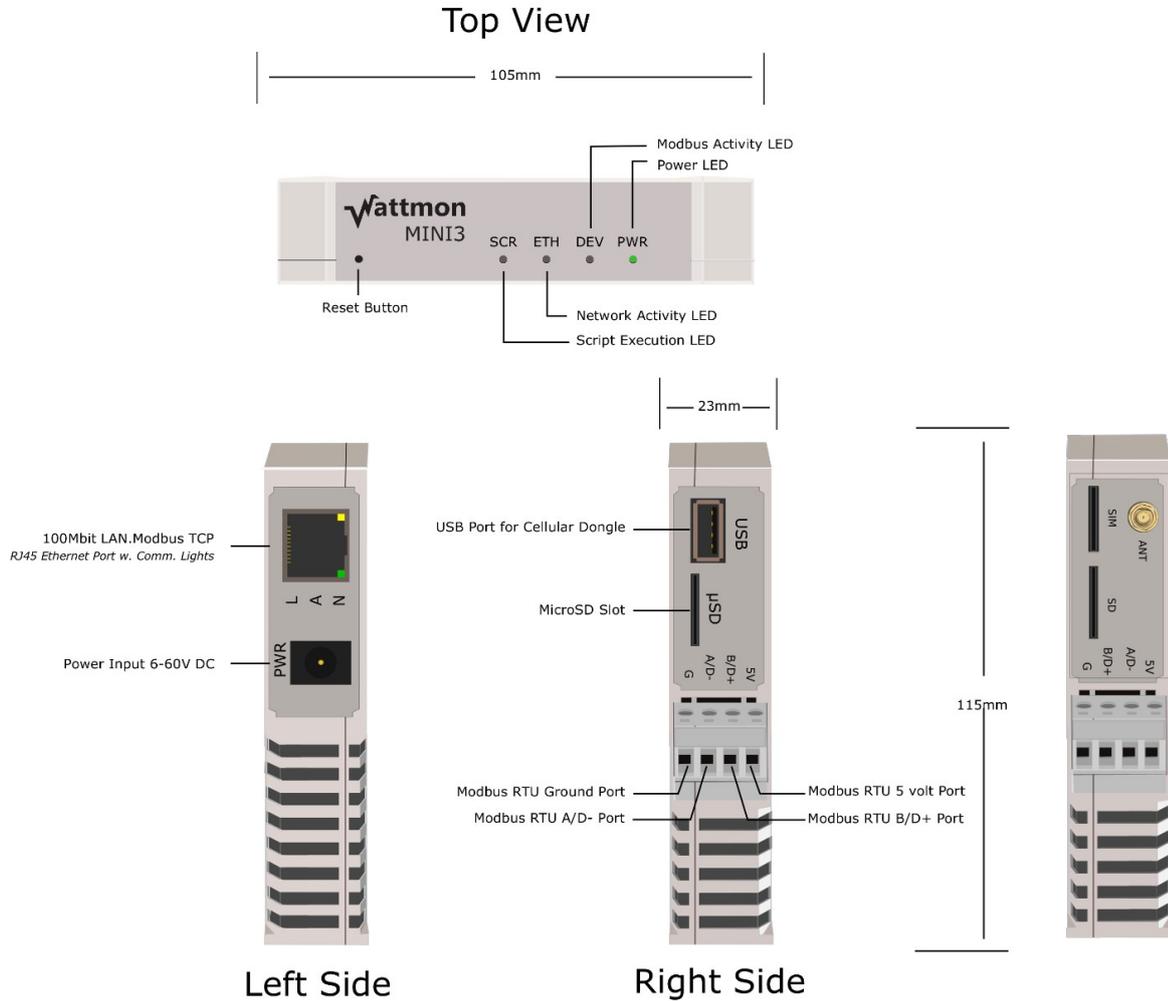
# Technical Specifications

The below table gives you an overview of the different features of our active range of loggers.

<b>DATA LOGGERS</b>	<b>ULTRA</b>	<b>MEGA2-4G</b>	<b>MEGA2</b>	<b>MINI3-4G</b>	<b>MINI3</b>
<b>Physical inputs</b>					
Analog Input	4 (0-10V / 4-20mA)	-	-	-	-
Digital I / O	4 / 4	4 / 0	4 / 0	-	-
1-Wire	Yes	-	-	-	-
RS-485 Port	2	2	2	1	1
No. of Modbus RTU Devices	64	64	64	15	10
Modbus TCP Client	Yes	Yes	Yes	Yes	Yes
Modbus TCP Server	Yes	Yes	Yes	Yes	Yes
No. of TCP Devices	30	30	30	10	10
RS232 Serial	Yes	-	-	-	-
<b>Networking</b>					
LAN	Yes	Yes	Yes	Yes	Yes
WiFi – AP/STA	Yes	-	-	-	-
GSM – 3G/4G	Yes	Yes	-	Yes	via USB Stick
<b>Storage</b>					
Internal Flash Storage	64 MB	-	-	-	-
External Storage (microSD)	32GB	32GB	32GB	32GB	32GB
<b>Dimensions</b>					
Dimensions (in millimetre)	90 x 160 x 85	90 x 88 x 85	90 x 88 x 65	125 x 23 x 115	105 x 23 x 115
Weight (in grams)	200	190	180	155	160
<b>Data Management</b>					
Max Data Points	1500	1500	1500	500	500
CSV Data Log	Yes	Yes	Yes	Yes	Yes
Local Monitoring GUI	Yes	Yes	Yes	Yes	Yes
Export for Remote Monitoring	Yes	Yes	Yes	Yes	Yes
HTTP / HTTPS	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes
FTP / FTPS	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes
MQTT / MQTTS	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes
<b>Software Feature Set</b>					
Quick Configuration Tool	Yes	Yes	Yes	Yes	Yes
Active / Reactive Power Control	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes
Zero Feed-In	Yes	Yes	Yes	Yes	Yes
Genset Protection	Yes	Yes	Yes	Yes	Yes
<b>Compliance</b>					
Power Supply Voltage	6-60V DC	6-60V DC	6-60V DC	8-40V DC	8-40V DC
Power Consumption	<4W	<3W	<2W	<3W	<2W
DIN EN 60529 Compliant	Yes	Yes	Yes	Yes	Yes

## WattmonMINI3 / WattmonMINI3-4G

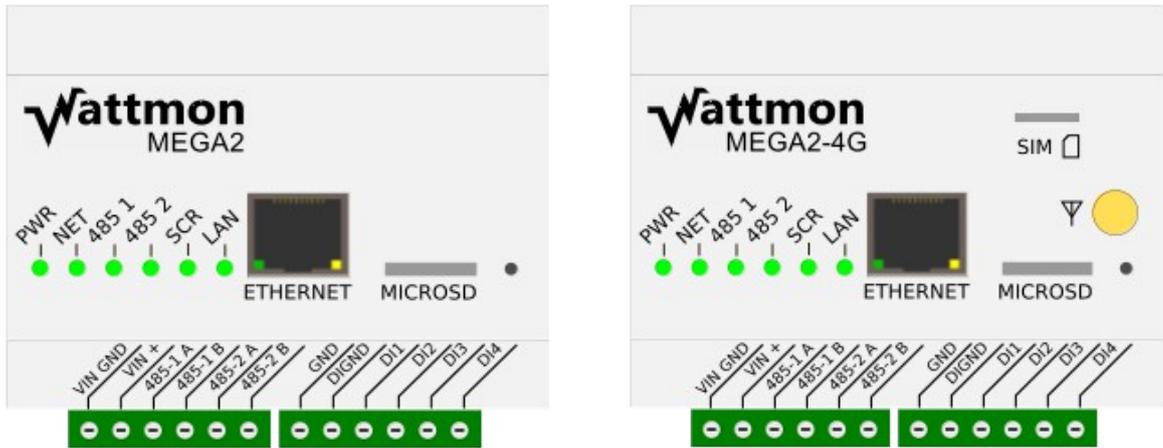
The WattmonMINI3 and WattmonMINI3-4G are the most compact loggers available in the Wattmon range. They feature a micro controller with 512KB of RAM and 2MB of Flash. The connection diagram is shown below:



For the **WattmonMINI3-4G**, the pinout diagram is identical except that the USB Port for a Cellular Dongle is replaced with a Nano SIM Slot and antenna port.

## WattmonMEGA2 / WattmonMEGA2-4G

The MEGA2 range of devices has 32MB of DDR2 memory built in and a 64MB Flash chip, allowing for the WattmonOS to run entirely from read-only memory (without an SD card). The extended memory compared to the MINI3 range allows you to connect many more devices and log more data. The WattmonMEGA2 has an Ethernet port for local LAN communication. The MEGA2-4G has LAN and an inbuilt LTE module which works with most 3G/4G providers. Insert the SIM in the slot and ensure the antenna is properly screwed in.



Two terminal blocks of 6 pins each below let you connect up power, RS-485 and digital inputs. Refer to the diagrams for WattmonULTRA below to understand the wiring.

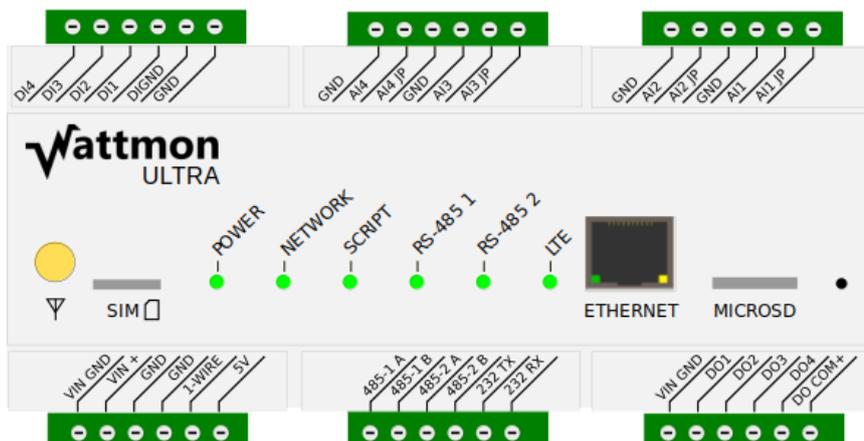
Your device will come with a small cable that connects to an adapter socket. On the Wattmon side, the black (negative) goes to the pin label *VIN GND* and the positive (red) goes to the *VIN+*.

RS-485 Channel 1 connects to the *485-1 A* and *485-1 B* terminals. Channel 2 connects to the *485-2 A* and *485-2 B* terminals.

Digital inputs are optically isolated but can be connected to the internal ground if required by connecting a jumper between *GND* and *DIGND* on the second terminal block. Connect *DI1* to *DI4* to an external voltage source (between 3 and 24V) for Wattmon to read the digital state a 1 (ON) or below that to read it as 0 (OFF).

## WattmonULTRA

The WattmonULTRA has the most I/O ports of all the hardware models. In terms of Internet connectivity, it has LAN, LTE (3G & 4G) and Wi-Fi support. It features 4 digital outputs, 4 digital inputs, 4 analog inputs, a 1-Wire bus, an RS232 Serial Port and 2 RS-485 channels.

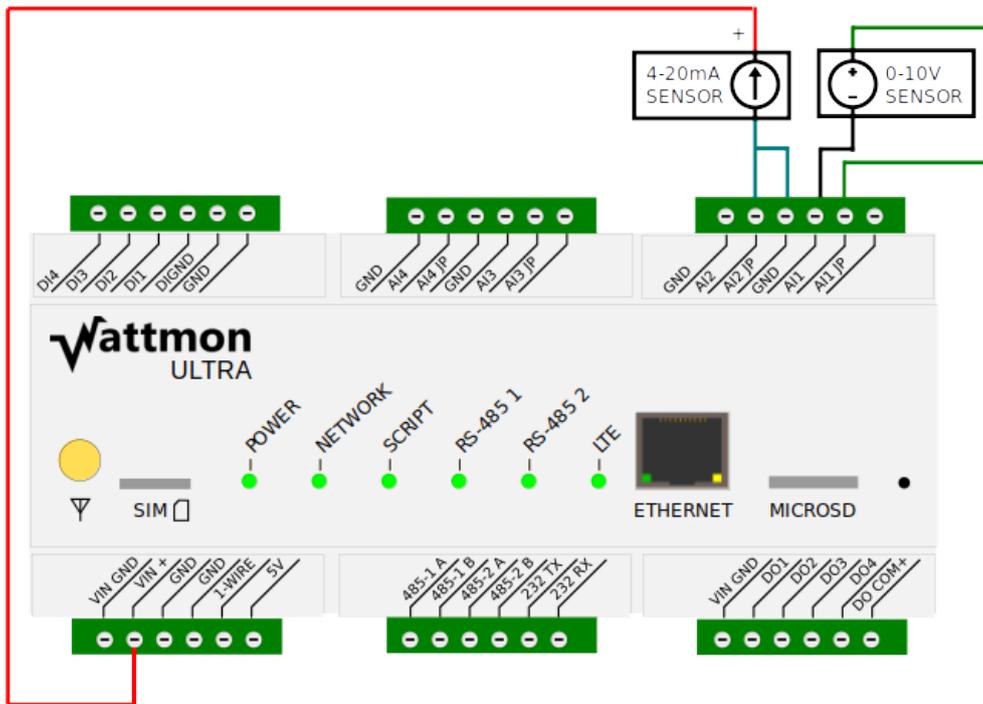


## Connecting Analog Sensors

The WattmonULTRA supports up to 4 analog input channels. These can be connected to either 4-20mA sensors or to 0-10V sensors.

In order to read a 4(0)-20mA sensor, you need to create a jumper between AIx and AIx JP. This connects the internal resistance that allows the Wattmon to read the current properly. Connect one side of the 4-20mA source to GND and the other side to the AIx/ AIx JP pin. For 0-10V, connect one side of the external voltage to the GND and the other (positive) side to AIX(1-4). Your sensor itself may require an external voltage supply - ensure that it is in the same range as the Wattmon's power supply or provide a suitable adapter for it.

See the diagram below for the wiring options for both 0-10V and 4-20mA.

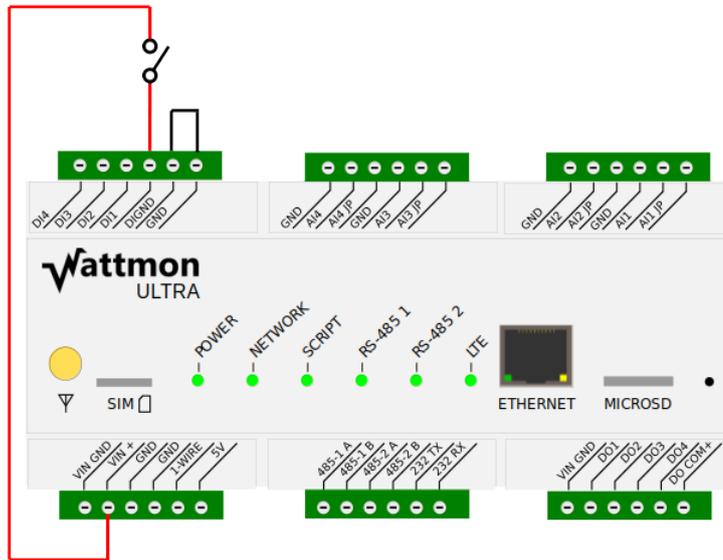


The teal (blue) wire creates a jumper between AI2 and AI2 JP, putting the analog channel into current mode rather than voltage mode. The voltage from the supply flows through the sensor creating a current loop that gets converted into a voltage at the AI2 terminal and can be appropriately converted to a reading.

*Please note:* The WattmonULTRA device will need to be calibrated appropriately for this. Refer to the Modbus devices section later in this manual for instructions how to calibrate it. Reach the calibrate page through the Devices list, and ensure that the channel mode for the channel is set to Voltage or Current as required.

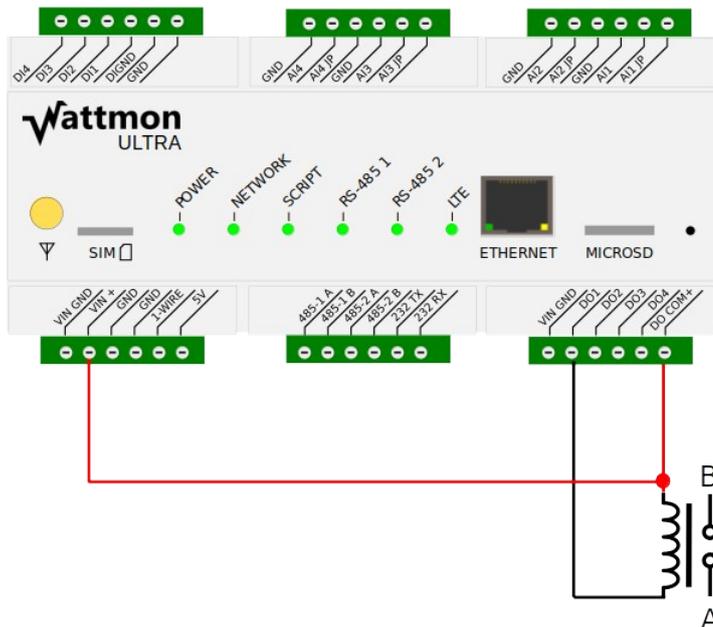
## Connecting Digital Inputs

The DI (Digital input) pins 1-4 are at the top left of the WattmonULTRA. A positive voltage of over 3V will give a reading of 1 (On) and below will be 0 (Off). If you are using the same voltage source as the Wattmon (for example, a potential free contact) then you will need to connect the GND and DIGND to ensure a common ground reference.



## Connecting Digital Outputs

Wattmon support up to 4 digital outputs. Bring the input voltage (or a separate supply of up to 25V DC) into the DO COM+ pin as shown in the diagram below, and take the same wire to one side of the relay you wish to power. Connect the other pin of the relay coil to

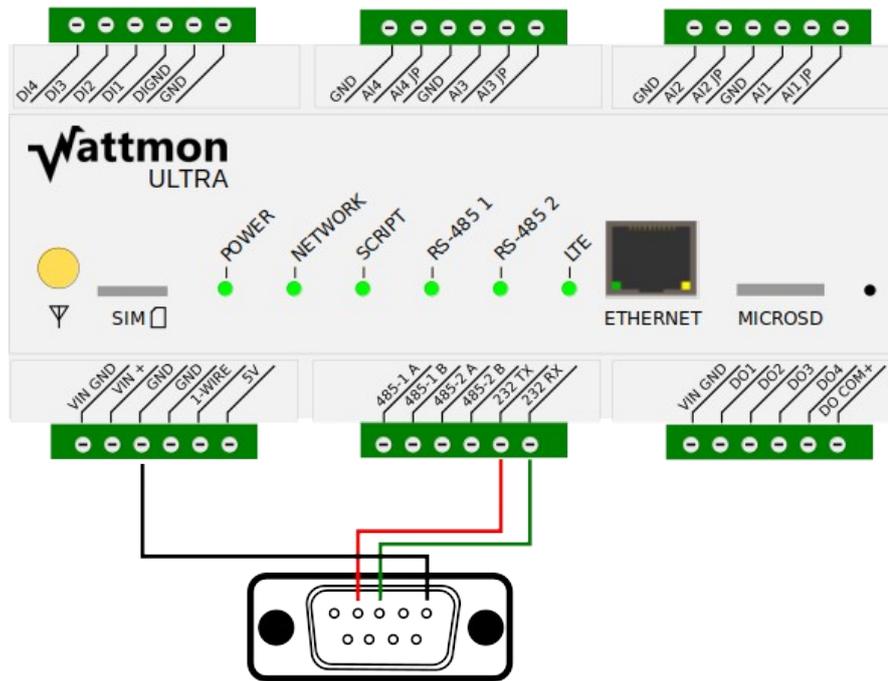


the DOx (1-4) pin of the WattmonULTRA. This will energize the relay when the DO pin is switched on in Wattmon.

The connection internally uses a common collector, which means that the DO1-4 pins are floating when off and grounded (0V) when on.

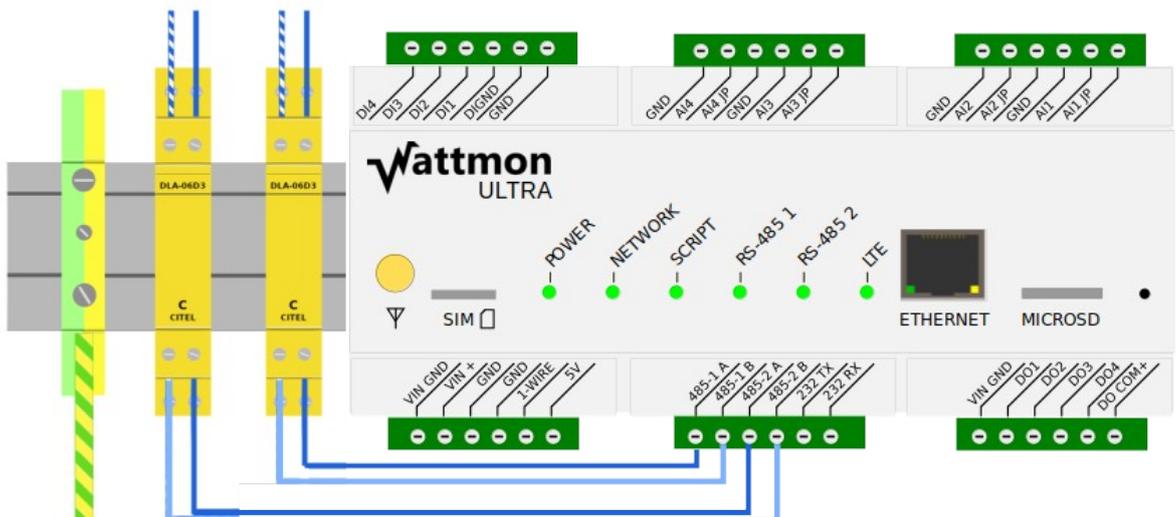
## RS-232 Serial Communication

The WattmonULTRA comes with one RS-232 port. The 232 TX and 232 RX terminals need to be connected to a suitable DB9 connector in order to interface with legacy equipment. Use pins 2,3 and 5 of the DB9 connector. Pin 2 goes to TX, pin 3 to RX and pin 5 to GND.



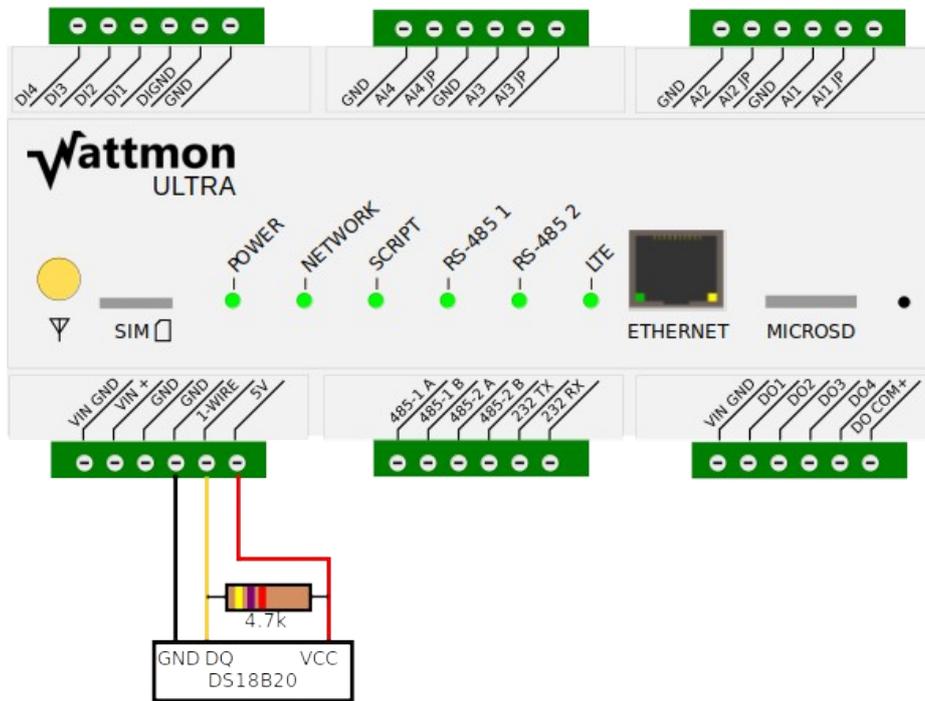
## RS-485 Communication

The two RS-485 ports have 2 terminals denoted A and B which need to be connected to the corresponding A and B terminals on the device to be monitored. It is recommended to use high quality shielded twisted pair cable after the surge protection device. The diagram below show how to wire the WattmonULTRA to a Citel or other surge protector before running the longer cable to any modbus device.



## 1-Wire Bus

The 1-Wire bus lets you read values from DS18B20 or equivalent temperature sensors. You will need to connect a resistor of a value between 2.2 and 5.1k typically between the 1-WIRE and 5V terminals as close to the Wattmon as possible. You can then run the 3 wires to your various DS18B20 sensors and read them in the Wattmon.



## WattmonM2 (Legacy)

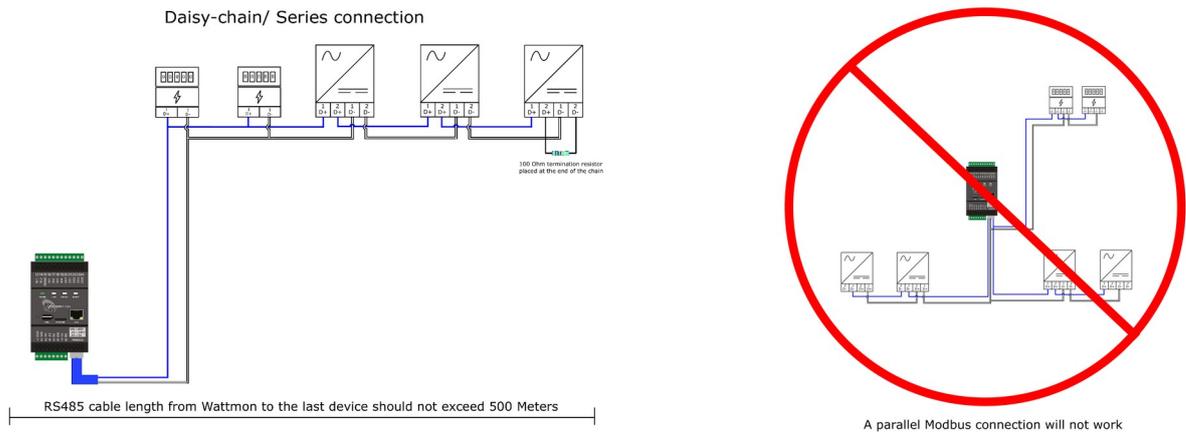


# WIRING RECOMMENDATIONS

## RS-485 Cabling

Connections to your Wattmon can differ based on the device(s) they are connected to. Third party Modbus devices such as inverters and energy meters connect to a Wattmon with either a custom Modbus cable which features a RJ45 jack on one end (Wattmon-MEGA), or a standard RS485 two-pair cable. The RS-485 protocol uses two wires, denoted A and B or D+ and D- which need to be connected to the corresponding terminal on the device to be monitored.

It must be noted however that all Modbus RTU connections over RS-485 should be done in a daisy-chained manner, i.e., in a series starting from the Modbus master (Wattmon) to the first Modbus slave (inverter / meter / sensor), from the first Modbus slave device to the second slave, then from the second slave to the third slave, and so forth. This is represented in the diagram below.

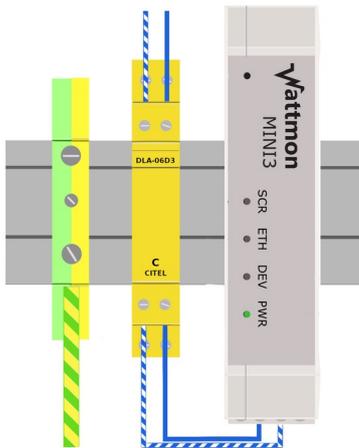


Modbus connections must not be done in parallel, amounting to multiple series of devices connecting to a single common point and then to the Wattmon, as this can cause communication issues on one or more devices. The WattmonMEGA2, MEGA2-4G and ULTRA all have two RS-485 channels.

For longer cable runs or when multiple devices are used, it is recommended to install a terminating resistor of 100 ohms between the A and B terminals at the end of the chain. Many inverters have an option to enable this with a switch inside the inverter panel.

## RS-485 Surge Protection

In order to protect your device, it is highly recommended to connect a suitable RS-485 surge protection device such as the Citel DLA-06D3 which will help prevent damage to your device in case of lightning or electrical surges over the RS-485 cable.



The Wattmon warranty is void for setups that do not have proper protection installed.

## RS-232 Connection

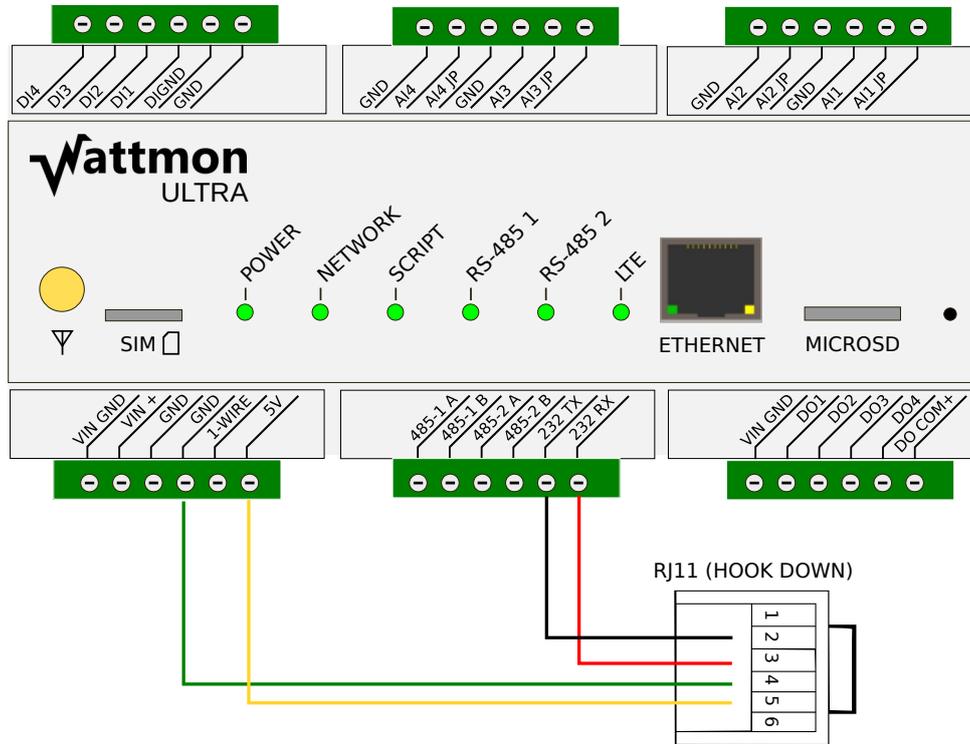
Serial connections for legacy equipment can be connected to both the MEGA (discontinued) or the WattmonULTRA. The connection diagrams for this are shown below. The RS-232 wires need to be brought to a suitable DB9 connector in order to interface with standard RS-232 devices.



It is recommended to power off the device before connecting the RS-232 as the RS-232 voltage levels may damage the equipment during connection.

## DLMS Connections

Wattmon OS 3.20 and above with firmware 4.1350 and above support DLMS meters via RS-232 (on WattmonULTRA) and on TCP through a third party RS-232 to TCP converter. The wiring for DLMS to RS-232 depends on the make and model of the meter.



The above diagram shows the wiring for Secure DLMS meters. The RS-232 port needs to connect via RJ11 to the meter and has to supply 5V in order to power the RS-232 circuitry.

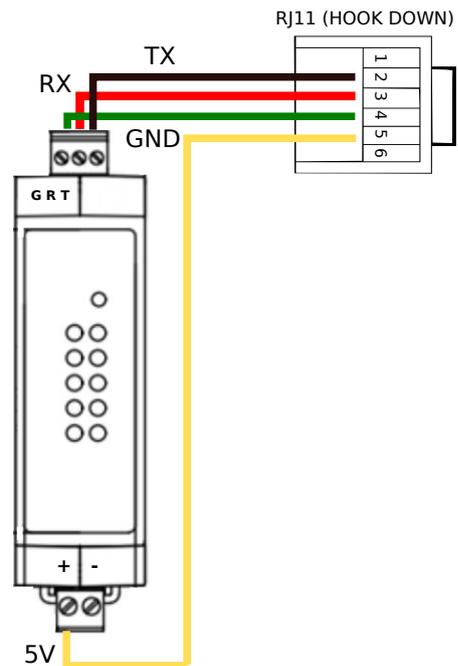
When connecting via an external serial to TCP converter, ensure that you provide the 5V circuitry for it to work. If you are powering your device from the same supply as the Wattmon (i.e. 24V DC) then you can just run a wire from the 5V line of Wattmon to the RJ11 connector since both devices would share a common ground.

The diagram at the right shows a typical RS-232 to TCP Converter. **Ensure that the power supply is 5V** or else provide a separate 5V supply when connecting to the DLMS meter as higher voltages may damage the equipment.

The converter used here is USR-DR302.

To understand how to configure your DLMS meter, please see the section on Modbus devices and device types.

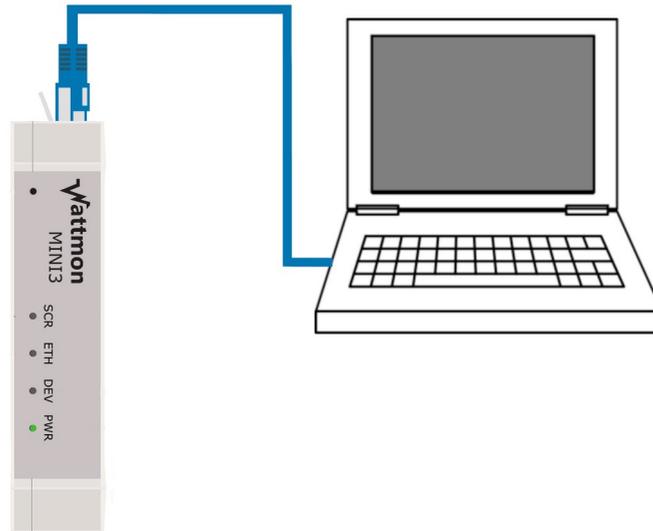
Please note that the wiring is for a Secure make of meter. For different brands, please refer to the manual of the respective manufacturer.



## CONNECTING TO YOUR DEVICE

After ensuring proper cabling and following the recommendations in the Wiring recommendations section, power on your device by connecting it to its power source - Wattmon can be powered either by a 12V/24V DC adapter or via a DC cable connected to a battery. At the time of purchase you will receive either a wall mount 24V 2A rated adapter, or a DIN rail mounted one depending on the configuration chosen.

Connect your device via a direct Ethernet cable to your computer. The device comes pre-configured to use a static IP address of 192.168.0.55. Note that in some rare instances the standard Ethernet cable will not work for older laptops that don't auto detect the polarity - in this case you will need to procure a crossover cable. You can assume this in case the LAN lights (orange and green) do not come on when you connect the cable.



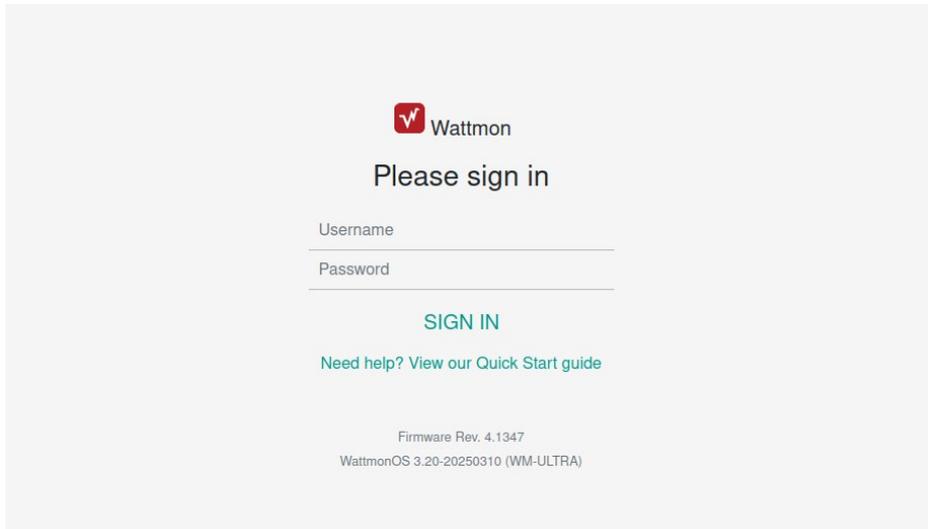
If the IP address range of your computer is the same as that of the Wattmon, you can access the Wattmon by opening a web browser and typing in `http://192.168.0.55` in the address bar.

### Connecting in Microsoft Windows

To determine your IP address, run `cmd.exe` and type in `ipconfig` and press Enter. When using Microsoft Windows, if the IP address is in a different range, you will need to temporarily change your IP by following these steps:

1. Open *Network and Sharing Centre* and click on *Local Area Connection*.
2. In the *Local Area Connection Status* window, click on *Properties*.
3. In the *Properties* window, select *Internet Protocol Version 4 (TCP/IPv4)* and click on *Properties*.
4. Make a note of the information displayed so you can restore this later. (If you do not re-configure your connection after you setup the Wattmon device, your Internet connection may not work.)
5. Enter the following values into their respective places and save.  
IP address: 192.168.0.10 (or any IP in the range other than 55)  
Netmask: 255.255.255.0  
Default Gateway: 192.168.0.1
6. Open your browser and type in `192.168.0.55` and press enter. You should see the *Wattmon Login* page.

7. After entering the network settings of your choice, as shown in **Network Setup**, save the changes and reboot. If the settings are successfully applied, you will not be able to reach the login screen of your Wattmon.
8. Return your computer to the original network settings by following steps 1-4.
9. Open a browser and type in the new IP Address assigned by you to your Wattmon and press *Enter*. If you see the *Please Sign In* page as shown below, the network setup for your device is completed.



### Resetting your IP Address

In order to reset the IP address on your Wattmon device to the default (192.168.0.55), press and hold the reset button (located to the left of the USB port - you will need a pin to access it) for about 5 seconds until the *Script* light stops blinking, and re-power your device.

Continue to hold the button down for a few more seconds and you will be able to toggle between DHCP and Static IP mode. If you are current in static IP mode, your system will reboot into DHCP mode and obtain an IP from your local DHCP server (router, modem etc). In this case you would need to determine the IP it was assigned. Sometimes you can do this by just typing `WATTMON` in the browser, or typing `ping wattmon` in the command shell.

### Modifying your IP Address on the SD card

If you cannot reach your device using any of the above methods, you always have the option to power off your Wattmon, remove the SD card and connect it to the PC.

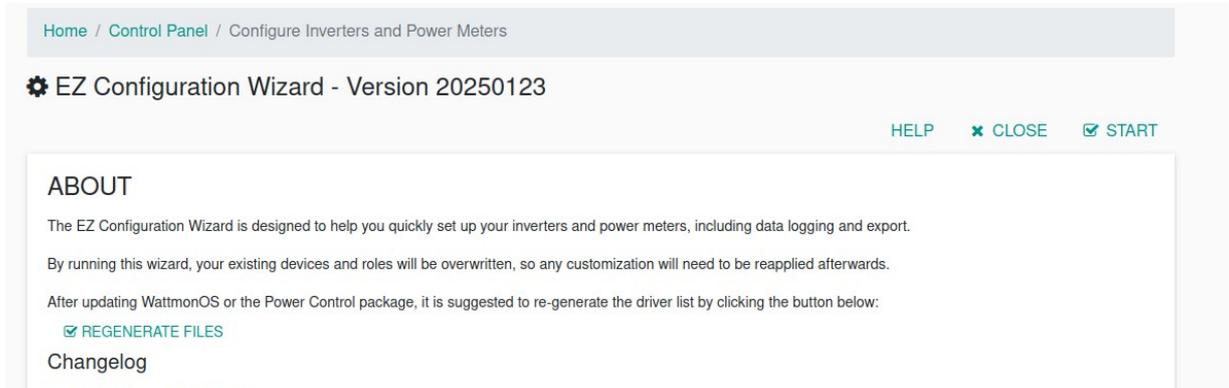
```
1 dns=8.8.8.8
2 [network]
3 ip=192.168.0.55
4 netmask=255.255.255.0
5 gateway=192.168.0.1
6 dns=8.8.8.8
7 dhcp_enabled=0
8
9 tcp_server_enabled=1
10 tcp_server_port=7000
11 tcp_server=remote.wattmon.com
12 telnet=0
13
```

In the `config` folder, open `network.ini` in your favorite editor and modify the IP. Place it back in the Wattmon and power it up to access it on your new IP.

# EZCONFIG – QUICK CONFIGURATION TOOL

To configure your Wattmon Data Logger use the built-in *EZConfig* tool that allows you to set up your Wattmon quickly. The EZConfig tool will run you through various steps to quickly get your devices connected and configured. It is suggest to make any changes to your system in future throught this tool rather than changing values in devices or roles, etc as they will be overwritten once EZConfig runs again. You can reach the EZConfig wizard through *Control Panel* or from the *Admin > EZConfig* menu.

If you intend to run EZConfig from flash, ensure that the flash is unlocked first by selecting *Admin > Unlock Flash*.



To get started, click the *START* button. If required, you can click the Regenerate files to update devices and packages in case you manually added any.

## EZConfig Wizard Step 1 / 4: Choose Devices and Configuration

The first step allows you to set Modbus communication settings, select inverters, energy meters, other Modbus devices supported by the Wattmon, in addition to selecting packages and important options at the bottom of the page. Each section will be described below.

### Modbus RTU Channel 1

The *Modbus RTU Configuration* section lists the *Baud Rate*, *Parity* and *Polling Interval* settings, which should be identical for all the Modbus RTU devices connected to the Wattmon.

- The *Baud Rate* determines the speed of communication and can be set to anything between 1200 and 115200 bits per second. The default is set to 9600.
- For *Parity*, select any one of the three available options, which are *No Parity*, *Even Parity* and *Odd Parity*.
- Inter-packet delay is the interval in milliseconds between packets – increase this if your devices are not responding properly. The default is 10 but you can set it up to 50 or more if required.
- Short Timeout is the interval that Wattmon waits before receiving a single byte back from a device before deciding it is inactive.
- Reply timeout is the maximum time to wait for a packet before skipping to the next packet.
- Retry is the number of retries before failing

**MODBUS RTU CONFIGURATION**

Communication Settings should be identical for all devices

Baud rate  
9600

Choose the baud rate that matches the devices you wish to connect to

Parity  
 No Parity  
 Even Parity  
 Odd Parity  
 Select the parity matching your device settings

Inter-packet delay  
10

Inter packet delay

Short Timeout  
30

Short timeout in ms, if data is not received within this time it will skip

Reply Timeout  
300

Modbus reply timeout in ms

Retry  
1

Number of retries

Retry Interval  
15

Time in ms after which a failed packet is resent

Polling Interval  
1000

Interval in milliseconds between polling devices

- Retry interval is the time between modbus retries
- The *Polling Interval* sets the Modbus polling time in milliseconds, the interval at which the Wattmon polls for data. The default is set at 1000.

## Modbus RTU 2

This option is available on WattmonMEGA2,MEGA2-4G and ULTRA. It has identical options as for Modbus RTU channel 1, with an additional bridge mode.

**MODBUS RTU 2 CONFIGURATION**

Communication Settings should be identical for all devices

Modbus Engine  
 Enabled - RS-485 (and Modbus TCP if available) will be polled automatically  
 Disabled - Use the RS-485 port for proprietary protocols through scripts  
 RTU Bridge  
 If you have external devices connected using Modbus, make sure this is enabled. For proprietary protocols you may choose to disable this.

Baud rate  
9600

Choose the baud rate that matches the devices you wish to connect to

Parity

## Modbus TCP

The *TCP Settings Configuration* lets you configure Modbus TCP/IP settings, specifically the IP address of the devices you wish to interface with the Wattmon over Modbus TCP. The number of TCP channels is depended on the hardware and firmware. WattmonMINI3 and MINI3-4G support 10 channels, whereas the MEGA2, MEGA2-4G and ULTRA support 20 or 30 (recent firmwares support 30).

TCP SETTINGS CONFIGURATION ▾

Modbus TCP configuration

Enabled

Disabled  
 Enabled

Select the parity matching your device settings

Inter packet delay  
10

Inter packet delay

Reply Timeout  
1000

Modbus TCP reply timeout in ms

Retries  
1

Number of retries

Retry Interval  
15

Time in ms after which a failed packet is resent

The *Enabled*, *Inter-packet delay*, *Timeout*, *Retires* and *Retry interval* are all similar to the Modbus RTU section described above.

Modbus TCP Channel 1

TCP Server Mode

Select this to turn this channel into a Modbus TCP Listener server

Modbus TCP Device IP Address  
192.168.0.7

Enter the IP Address of the Modbus TCP device or 0.0.0.0 if not used

TCP Port  
502

Enter a port number for the connection. The default Modbus TCP port is 502

The *TCP Channel* sections as shown below let you enter an *IP Address* to connect to when in client mode. Enter the *TCP Port* number which is 502 for Modbus TCP by default. If you plan on having external devices connect to the Wattmon to read registers, check the *TCP Server Mode* and enter the port number to listen on. You may choose multiple channels that listen on the same port as well.

## Power Meters

The *Power Meters* section lets you add and configure the power meters that will be used in the system. A Power Meter refers to any third party modbus device that can provide an active power value that is used by Wattmon's Power Control algorithm – most brands are supported as of now but the Wattmon support team can be contacted to add a new driver for you if required.

The meters connect to the Wattmon via Modbus RTU or Modbus TCP.

Power Meters

Number of Power Meters: 1

+ ADD

Device	Address	Bus	Name	Location	Rating (kVA)	Options
1	1	Modbus RTU	meter1			ACTION ▾

After clicking *Add* a new blank row will appear, letting you select the *Device* from a drop down. This list will contain every driver in the Wattmon */dev/* folder that is defined as a power meter – this list is generated from the first page of EZConfig before entering Stage

1 and can be regenerated if required (for example if you have upload a new driver or added one). The *Address* needs to match the Modbus ID set on the meter.

Choose the *Bus* to connect to, which could be: Modbus RTU, Modbus RTU 2 (MEGA2, MEGA2-4G, ULTRA), TCP Channel 1-x, and Virtual. The Virtual bus can be used for certain more advanced setups that populate the device driver's values dynamically through a script.

It is suggested to keep the default naming convention of meter1-meterX but the name can be set to anything else as long as each device has a unique name.

Select the *Location* of the meter - this will show the meter at the appropriate place in EZDisplay and update power control accordingly. Location could be: *At Genset* (output of the generator), *At Incomer* (between the grid and the building), *At Inverter* (between the inverter and the building, also referred to as a check meter), *At Load* (between main AC panel and the load), and *ESS* (at the output of an energy storage system).

If you select *At Genset*, you will be prompted to enter the *kVA rating* of the generator which will help Power Control select an appropriate set point.

## Inverters

Grid-Tie (or hybrid) inverters need to be defined in the *Inverters* section. Clicking the *Add* button will bring up a blank row that needs to be configured.

Make	Model	Address	Bus	Name	Rating ?	Options
1 Sungrow		1	Modbus RTU (	inverter1	0	ACTION ▾

The *Make* drop down contains the inverter brand name - select the inverter make and the choose the model number from the *Model* drop down. If you do not find your make and model, contact support to get a suitable driver made for your inverter.

Ensure the Modbus ID of the inverter matches the *Address* you enter. Select the *Bus*, which can be any of the physical buses, TCP channels or the virtual bus. Choose a unique name for the inverter - it is recommended to follow the default naming convention of inverter1-inverterX. Enter the *AC rating* of the inverter in kW.

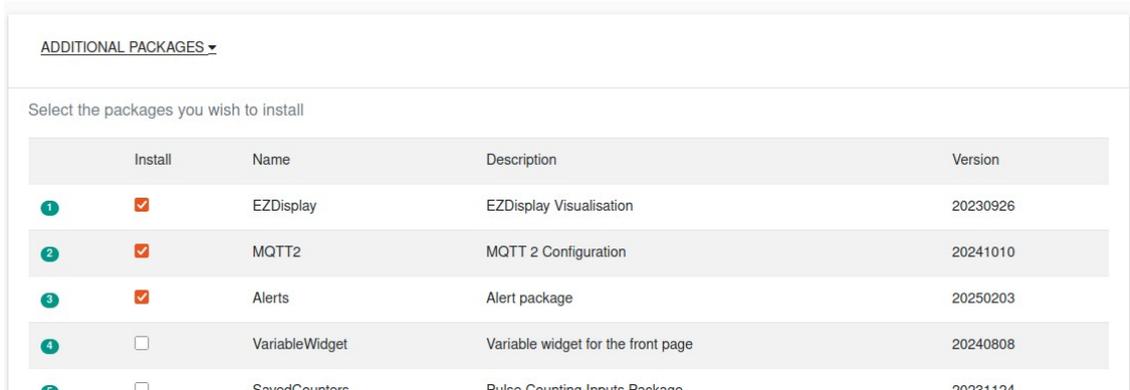
## Other Devices

The *Other Devices* section is where you define any other sensors or modbus devices. If you device is not in the drop down list, contact support. Enter the Modbus address, choose the bus and enter a unique device name.

If you have a physical Pyranometer, you should select it from the list of devices here. If you choose to use the *Pyranometer* package below, a virtual device called Pyranometer will automatically get added to the Other Devices list, in addition to the physical device you add. Ensure the name of your physical device is unique to avoid conflicts.

## Additional Packages

The additional packages section lets you choose packages that you wish to use, such as MQTT2, Alerts, FTP or QOS. These need to be downloaded using the *WattmonOS updater* beforehand unless they are already provided in WattmonOS. If they do not show below, you will need to *REGENERATE FILES* in the first page of EZConfig.

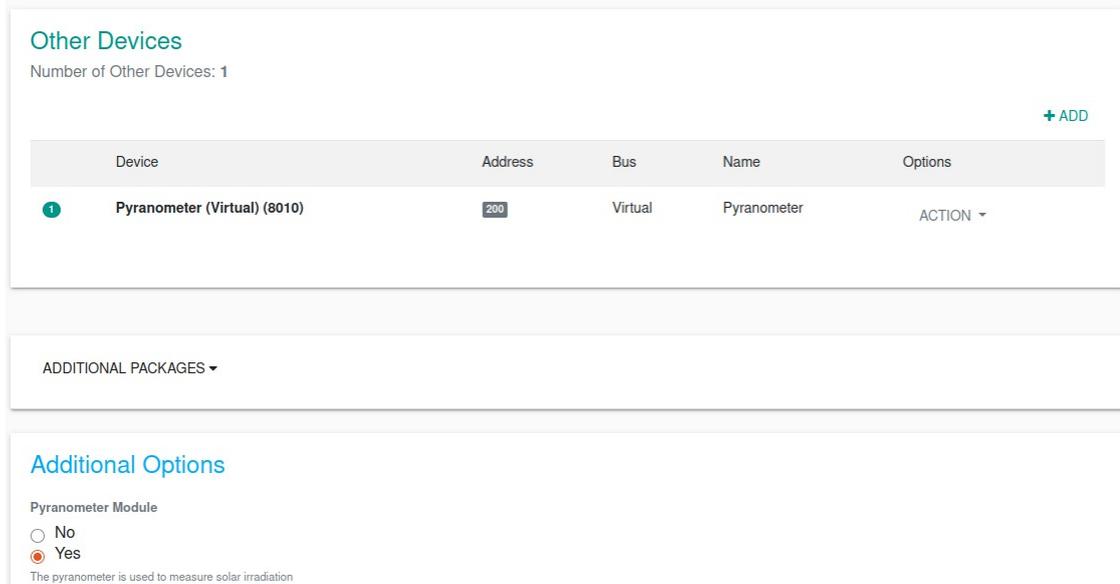


	Install	Name	Description	Version
1	<input checked="" type="checkbox"/>	EZDisplay	EZDisplay Visualisation	20230926
2	<input checked="" type="checkbox"/>	MQTT2	MQTT 2 Configuration	20241010
3	<input checked="" type="checkbox"/>	Alerts	Alert package	20250203
4	<input type="checkbox"/>	VariableWidget	Variable widget for the front page	20240808
	<input type="checkbox"/>	SavedCounters	Pulse Counting Inverse Package	20231124

If you do not select additional packages here, they will not be enabled after a reboot. This is a common mistake that is made, in case you have manually installed a package and then rerun the EZConfig you will notice that the package always gets uninstalled.

## Additional Options

Additional options determines some important system-wide values, including logging and power control settings.



**Other Devices**  
Number of Other Devices: 1 + ADD

Device	Address	Bus	Name	Options
1 Pyranometer (Virtual) (8010)	200	Virtual	Pyranometer	ACTION ▾

**ADDITIONAL PACKAGES ▾**

**Additional Options**  
Pyranometer Module  
 No  
 Yes  
The pyranometer is used to measure solar irradiation

The Pyranometer Module option allows you to add either a Modbus or an analog irradiation sensor to monitor both active and total solar irradiance values, while also providing you with the option to calculate performance ratios and track curtailment. After clicking Yes, a virtual Pyranometer device is added to the *Other Devices* list with address 200 on a virtual bus as explained above.

An addition configuration step may need to be done after completing EZConfig after a reboot by going to Control Panel / Pyranometer and manually selecting the Active Power role from the physical modbus device. This is only required if you do not see proper values on the *Sun* icon on the main *EZDisplay* dashboard widget.

DC Plant Size  
153

---

DC Plant size in kWp

Fixed PR  
80

---

Fixed PR for Potential kW calculation. Set to 0.8 as starting point.

CSV Log Interval  
1 Minute

---

Choose the log interval in minutes (Only 1 entry in the CSV file every X minutes)

CSV Log Write Interval  
1 Minute

---

To optimize SD card life, you can choose to log to disk at a lower rate than the CSV log interval. This will mean data will be in memory and could potentially be lost if the device reboots before it is written.

Enter the *DC Plant Size* in kWp to ensure that proper potential generation values can be calculated.

Enter a *Fixed PR* (performance ratio) which will be used by the potential generation calculation algorithm. A default of 0.8 is usually a good place to start and you may need to adjust it up or down to match the output of the plant with the potential generation value. You can read more about this in the Pyranometer Package section.

*CSV Log Interval* - keep at 1 minute for most sites unless you require less data to be logged.

*CSV Log Write Interval* - the interval at which log files are written to. You can keep this at 1 minute or increase it on higher end devices with more memory to reduce the wear of the SD card. Keep in mind that if a power cut occurs data in memory will be lost so keep the write interval at a reasonable interval between 1 and 5 minutes.

Power Control  
 No  
 Yes  
 Enable power control to automatically manage inverter generation

License Key  
 Enter License Key

---

Enter a license key supplied by Wattmon to activate power control

Track Curtailment  
 No  
 Yes  
 Keep track of generation loss due to curtailment

Update Power Control Settings Automatically  
 No  
 Yes

*Power Control* - choose yes to automatically install the Power Control (PC) package.

*License Key* - enter your license key for power control if available. You can do this in the PC package later too.

*Track Curtailment* - Select yes to use the curtailment tracking algorithm in power control which will provide information about potential energy lost due to curtailment

*Update Power Control Settings* - select yes for Wattmon to automatically update the power control package set points based on the EZConfig data. It is recommended to double check power control after running EZConfig to ensure that everything is properly updated, including the set points for zero-export and DG protection.

*Anemometer* - select this to install the corresponding package if you have a wind speed sensor

*Create Automatic Snapshot* - select yes to generate a backup of the /config folder and have it sent to the cloud

Anemometer (WS-102)

No  
 Yes

The anemometer is connected to DI1 of the Wattmon to measure wind speed

Create automatic snapshot after generating

No  
 Yes

This will automatically create a snapshot of your settings that can be restored later.

CSV logging location

SD Card  
 Internal Flash

Internal flash can be used to log CSV data but due to space constraints it is recommended to use an SD card

Temperature Sensor (0 detected)

No  
 Yes

Enable this if you are using 1-Wire sensors to monitor temperature

**CSV logging location** – select *SD card* or *Internal flash*. It is not recommended to use internal flash as it will fill up quickly.

**Temperature Sensor** – on WattmonMEGA and ULTRA one or more 1-wire sensors can be added

**Automatically Update Packages** – select this if you want the Over-the-air (OTA) system to automatically update your packages (i.e. FTP and Power Control) to the latest version on the website.

**Automatically Update Firmware** – select this to periodically check for firmware updates and install them automatically.

Automatically Update Packages

No  
 Yes

When a new package that is installed (i.e. PC) is available online, automatically download and install it

Automatically Update Firmware

No  
 Yes

When a new firmware is available online, automatically download and install it

Automatic Healing

No  
 Yes

Scan all scripts daily and if any mismatch is found, automatically restore it from the repository if available

Device Info widget

No  
 Yes

Displays live inverter data on dashboard

**Automatic Healing** – select this to perform a regular disk check to find corrupt files and restore them from the repository if possible

**Device Info Widget** – select this to show the device list on the dashboard below the EZDisplay.

To save your settings, select **APPLY CHANGES** from the *Control Bar* or proceed to the next step by clicking on **SAVE AND NEXT**.

## EZConfig Wizard Step 2 / 4: Choose Data Points to Log

Depending on the subscription plan and amount of memory your Wattmon model has, you may not be able to log all the data points from every device. Select the data points or variables you wish to log for each of your selected Power Meters, Inverters and Other Devices. Each of the device types are listed along with the respective parameters that can be logged by the Wattmon. If you define several power meters or inverters using the same device type or model, the roles/data points will be applied to each of them.

The page displays the devices separated into the Power Meters, Inverters and Other Devices sections like in the previous step. The available roles for the device type and logging options for the role are listed, grouped by device type.

**Power Meters**

Schneider Conzerv EM6400NG (6420)

Use	Function (INSTANTANEOUS - AVERAGE)	Variable	Description
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Active_Power	Total AC Active Power
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Reactive_Power	Total AC Reactive Power
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Apparent_Power	Total AC Apparent Power
<input checked="" type="checkbox"/>	INSTANTANEOUS	kWh_Total_Import	Total Import Energy
<input checked="" type="checkbox"/>	INSTANTANEOUS	kWh_Total_Export	Total Export Energy
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Current_A	AC Current Phase A
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Current_B	AC Current Phase B
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Current_C	AC Current Phase C
<input type="checkbox"/>	INSTANTANEOUS	AC_Voltage_AB	AC Voltage Phase A to Phase B
<input type="checkbox"/>	INSTANTANEOUS	AC_Voltage_BC	AC Voltage Phase B to Phase C
<input type="checkbox"/>	INSTANTANEOUS	AC_Voltage_CA	AC Voltage Phase C to Phase A
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Voltage_AN	AC Voltage Phase A to Neutral
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Voltage_BN	AC Voltage Phase B to Neutral
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Voltage_CN	AC Voltage Phase C to Neutral
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Frequency	Total AC Frequency
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_PF	Total AC Power Factor

The *Checkbox* column needs to be ticked if the parameter is to be logged.

The *Function* column defines how a value is calculated. While the Wattmon polls each of data points of the devices based on the *Polling Interval* set in the previous step, the value for each data point is written to the Wattmon at the *CSV Logging Interval* which is usually 1 minute (also set in the previous step). The Function for each role can be one of the following: Average, Min, Max, Instantaneous.

Select *Average* To get the average value (sum of all entries in the logged period divided by number of polls). Select *Min* to log the smallest polled value for the period, or *Max* for the highest polled value. Use *Instantaneous* where the latest polled value is required. The column name *Function* is followed by (INSTANTANEOUS - AVERAGE) which allows you to set the function for all the values within a section to either of those types by clicking on the function name. We recommend that most values be set to instantaneous, especially any *Operating State* and *Event* or *Error* codes.

## Inverters

Sungrow SGxxCX PV String Inverter (7092)

Use	Function (INSTANTANEOUS - AVERAGE)	Variable	Description
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Active_Power	Total AC Active Power
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Reactive_Power	Total AC Reactive Power
<input checked="" type="checkbox"/>	INSTANTANEOUS	AC_Apparent_Power	Total AC Apparent Power

The *Variable* column lists the name of the role as logged in the Wattmon. This is what the variable is called. However, as mentioned in the previous step, the final logged role is named by combining these role names with each device's name as a prefix, which is set in Step 1.

The last column is *Description* provides a human-readable description of the role. The above holds true for both the Power Meters and the Inverters sections.

### Other Devices

1 Pyranometer (Virtual) (8010)			
<input type="checkbox"/>	Function (INSTANTANEOUS - AVERAGE)	Variable	Description
<input checked="" type="checkbox"/>	INSTANTANEOUS	Active_Irradiance	Solar Irradiance
<input checked="" type="checkbox"/>	INSTANTANEOUS	kWh_Day_Irradiance	Daily Irradiance
<input type="checkbox"/>	INSTANTANEOUS	kW_Potential_Average	Average Potential Generation

The *Other Devices* list lets you modify the variable name if required. It is not recommended to change this for the Pyranometer module but any other devices that you add can be customized if required.

If *Track Curtailment* was selected in step 1, you can choose to log the global curtailment values from the Global Variables list as show below.

**Global Variables**

Global variables can be logged here - some of the variables are used by custom packages. Make sure you enable the packages in *Stage 1* so the variables appear here. You can also choose to log custom variables at this stage if you plan to have custom scripts running.

[+ ADD](#)

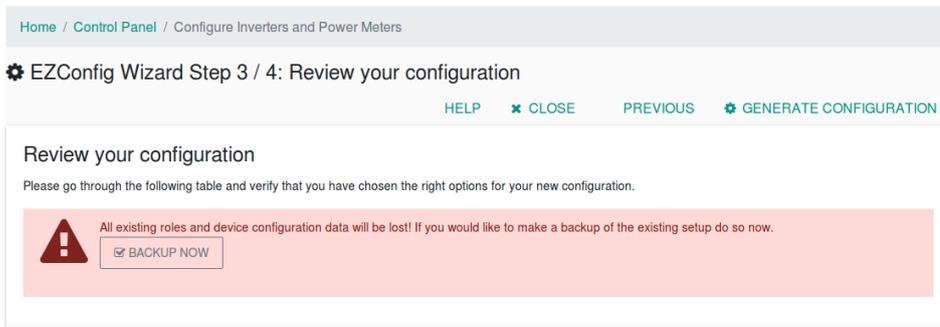
<input type="checkbox"/>	Function	Variable	Description
<input type="checkbox"/>	INSTANTANEOUS	eb_state	State of EB Input
<input type="checkbox"/>	INSTANTANEOUS	kWh_Curtailment_Day	kWh Curtailed Today
<input type="checkbox"/>	INSTANTANEOUS	kWh_Curtailment_Lifetime	Total kWh Curtailed
<input type="checkbox"/>	INSTANTANEOUS	kWh_Curtailment_DG_Day	Curtailment due to DG
<input type="checkbox"/>	INSTANTANEOUS	kWh_Curtailment_Grid_Day	Curtailment due to Zero Export

To remove a global from the CSV log, simply clear the variable name and proceed to the next step.

## EZConfig Wizard Step 3 / 4: Review Your Configuration

The next step of the *EZConfig* process allows you to preview your settings before they are generated. To make further changes, click on *PREVIOUS* in the *Control Bar*.

It is suggested to follow the instructions and click to generate a back of the existing configuration before proceeding if you are not starting with a blank install.



Below a list of roles and devices will reflect the settings you chose in previous steps.

The devices are listed under their respective section, *Power Meters*, *Inverters* or *Other Devices*, with their *Address* or Modbus ID encircled in orange for energy meters, green for inverters and blue-green for other devices. Next to the ID number is the device *Name*, as set in Step 1. Next to that is the *Device Type* name, followed by a table with the final *Role* name, a combination of the device and variable names, and the *Description* of that role.

### Power Control

Power control settings will not be modified. This will mean that you need to manually check the Power Control configuration after completing the EZConfig to ensure that settings are properly done.

### Power Meters

meter1 on Modbus RTU (RS-485)		Secure Elite 440-445 (6455)	
Role	Description	Role	Description
meter1_AC_Active_Power	Total AC Active Power		
meter1_kWh_Total_Export	Total Export Energy		

### Inverters

inverter1 on Modbus RTU (RS-485)		Sungrow SGxxCX PV String Inverter (7092)	
Role	Description	Role	Description
inverter1_AC_Active_Power	Total AC Active Power		
inverter1_AC_Reactive_Power	Total AC Reactive Power		

**Note:** Since this is the final step before the generation of the configuration for your Wattmon, it is imperative that you review the settings presented on this page. The Modbus Address should not be conflicting between any of the Modbus RTU devices, and while the ID may be identical in the case of Modbus TCP devices, the IDs cannot be the same between Modbus RTU and Modbus TCP devices.

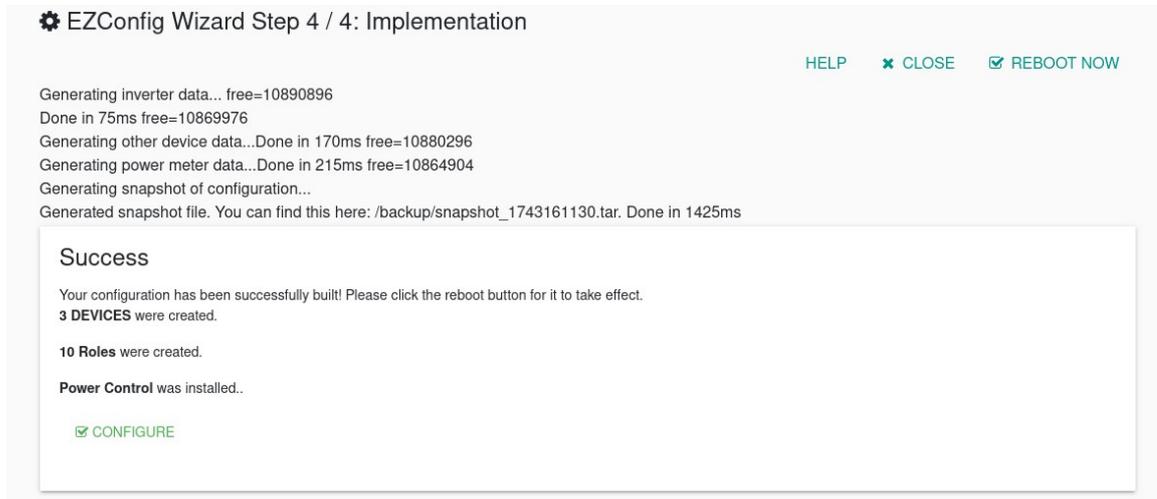
The device *Name* will also be shown in all the *Roles*, which should be verified to be correct. If the inverters or meters have been renamed, make sure that the Roles are accordingly set.

**Note:** The roles/variables naming convention should be as per the Wattmon guidelines or the default names attributed by EZConfig, especially if the Wattmon Power Control function is to be used.

If all the allocated *Roles* are correct (as shown in the images above), then click on *GENERATE CONFIGURATION* to proceed to the final step of the *EZConfig Wizard*.

## EZConfig Wizard Step 4 / 4: Implementation

The final step is a confirmation of the configuration process, and it lists all the particulars of your configuration, such as the number of Modbus *Devices* to be interfaced with the Wattmon, the total number of *Roles* created which is the total count of the number of logged variables which will be exported to either the Wattmon EMS or a third-party server, along with any of the additional packages installed such as the *Power Control* solution or the *Device Info Widget*. The confirmation will appear as shown below.



⚙ EZConfig Wizard Step 4 / 4: Implementation

HELP ✕ CLOSE ✓ REBOOT NOW

Generating inverter data... free=10890896  
Done in 75ms free=10869976  
Generating other device data...Done in 170ms free=10880296  
Generating power meter data...Done in 215ms free=10864904  
Generating snapshot of configuration...  
Generated snapshot file. You can find this here: /backup/snapshot\_1743161130.tar. Done in 1425ms

**Success**

Your configuration has been successfully built! Please click the reboot button for it to take effect.  
**3 DEVICES** were created.  
**10 Roles** were created.  
**Power Control** was installed..

✓ CONFIGURE

If the Power Control solution has not been installed, and instead the Device Info Widget was selected, the success message will then omit the *Power Control was installed...* message and will instead display *Device Info Widget was installed...*

Since this is the final point of the setup and marks the completion of the *EZConfig Wizard*, there is no way to go back to the previous step as in the other pages.

Click the *Configure* below power control to open up the power control package in case you selected the Power Control option, and verify that all settings are correct.

If the summary of the settings appears to be as per your selection, then you need to restart the Wattmon for the settings to be applied and the configuration put into action. In the *EZConfig Wizard*, click on *REBOOT NOW*, which will restart your Wattmon and apply all the new configuration changes.

# UNDERSTANDING THE WATTMON OPERATING SYSTEM

The Wattmon devices run on our proprietary firmware written in C that uses the FreeRTOS task scheduler. It features an interpreter that can process uPHP, our custom scripting language based on PHP. This firmware layer is flashed onto the microcontroller, whereas WattmonOS, the application layer, is stored on an SD card or on the external Flash chip (MEGA2, MEGA2-4G, ULTRA). The firmware reads configuration files at power-up off the SD card and initializes the system accordingly.

## WATTMON OS INTERFACE

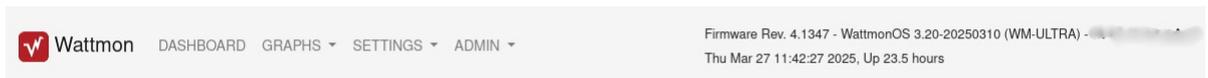
The first page that comes up after logging in is the Dashboard. The default dashboard does not provide much in the way of information other than some links to get started, but once the device is properly set up the EZDisplay widget gets installed which shows the current status of the system in a visual interface.

The menu bar is present at the top of every page, as is the control bar, which is found on all pages except the dashboard.

The menu bar displays the Wattmon logo on the left, followed by four options - *DASHBOARD*, *GRAPHS*, *SETTINGS*, and *ADMIN*.

On the right, the menu bar displays the firmware version as well as the OS version of the Wattmon.

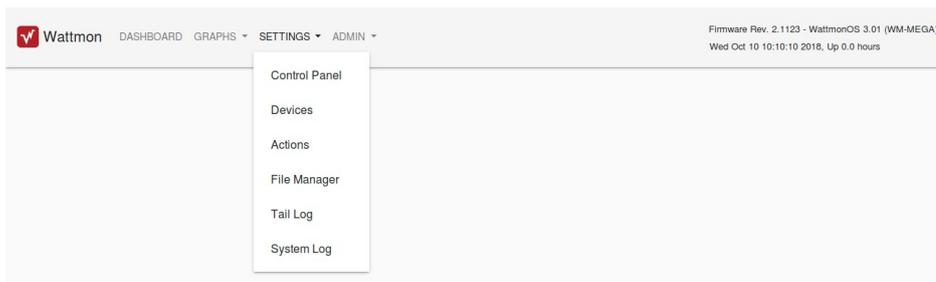
Below the firmware and OS versions is noted the date and time set on the Wattmon, as well as the time elapsed since the last power-up.



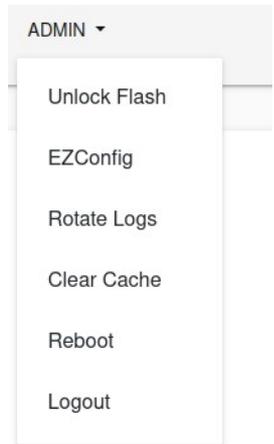
The first of the menu options, *DASHBOARD*, returns you from your current page to the main graphical interface of your Wattmon, which consists of various widgets displaying either graphs or variables in an intuitive manner.

The second option is *GRAPHS*, which reveals a drop-down that leads to dedicated graphs for the data you are monitoring. These graphs can be configured to display the logged variables of your choice.

The third option, *SETTINGS*, is also a drop-down, which consists of *Control Panel*, *Devices*, *Actions*, *File Manager*, *Tail Log* and *System Log*. The *Control Panel* takes you to the main settings page of the Wattmon. *Devices* takes you to a list of Modbus devices, wherein you can add, remove, or configure devices connected to your Wattmon. *Actions* lets you set custom commands for automated functions or alerts. *File Manager* allows you to access the contents of your microSD card, internal flash drive (MEGA2,MEGA2-4G, ULTRA) or RAM drive (MEGA2,MEGA2-4G, ULTRA). *Tail Log* and *System Log* provide a log of operations and errors in the Wattmon, differing in the number of lines displayed; the former displays a portion of the current log, whereas the latter displays the complete current log file.



The *ADMIN* menu shown a drop-down that is as per the user login name; if the default Admin user of the Wattmon is called admin, the option is displayed as *ADMIN*. If you login as guest, the menu name will be GUEST. The menu displays *Lock/Unlock Flash* on devices that have interval flash, *EZConfig* that takes you to the quick configuration wizard, *Rotate Logs*, *Clear Cache*, *Reboot* and *Logout*.



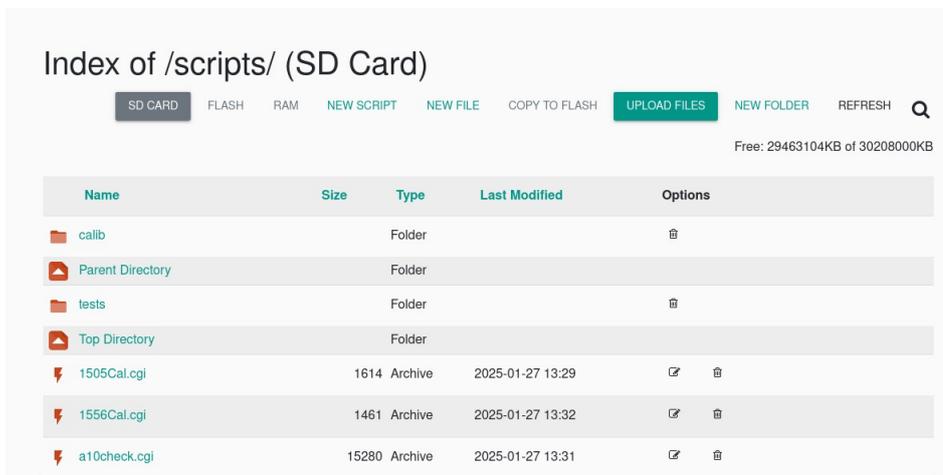
*Rotate Logs* replaces the system log to a new file.

*Clear Cache* clears the cached scripts and settings on the device.

*Reboot* restarts the Wattmon and returns you to the login page, and *Logout* just signs out the current user and returns you to the login page.

## FILE MANAGER

The contents of the WattmonOS on the micro-SD card can be accessed from your browser using the built-in *File Manager*, which is used to upload, download, modify or write configuration files, device drivers or script files locally or remotely, and to restore backed-up data. It is particularly useful when small updates or customizations are provided by us to you directly, which are not available as an over-the-air update.



To open the *File Manager*, click on *SETTINGS* in the menu bar present on the top of all pages, and select *File Manager* from the drop-down.

Files and folders are listed in rows with details such as *Name*, *Size*, *Type*, *Last Modified* and *Options*. To the right of each row, various icons will appear depending on the actions possible. These can vary between *Edit*, *Design*, *Uncompress* and *Delete*, each denoted by an icon of a notepad, gears, briefcase, and bin respectively, as shown in the image. Be careful when modifying files directly in the editor as human error can cause the Wattmon to stop working properly and may even require a site visit if something gets wiped (i.e deleting or modifying the network configuration file).



The *File Manager* control bar buttons change depending on the device. For the Wattmon-MEGA2 range and ULTRA three drives will appear: *SD CARD*, *FLASH* and *RAM* which allow you to switch between the drives (0: 1: and 2: respectively) The *COPY TO FLASH* option allows you to copy the current folder onto the flash drive. *NEW SCRIPT*, *NEW FILE*, *UPLOAD FILES*, *NEW FOLDER*, *REFRESH* and *SEARCH*.

The *NEW SCRIPT* opens *Blockly*, which lets you use simple drag and drop block-like operations to create uPHP scripts.

*NEW FILE* opens the built-in file editor, where you can write config files or custom scripts for the Wattmon.

*UPLOAD FILES* allows you to upload files to the Wattmon.

Click the browse button to select one or more files. Once a file is uploaded a green tick appears next to it as shown above.

Clicking on *New Folder* creates a new folder (directory) in the current folder.

*Refresh* allows you to refresh just the contents of the *File Manager* without having to refresh the entire page in your browser, which allows the directory to load faster without using much bandwidth.

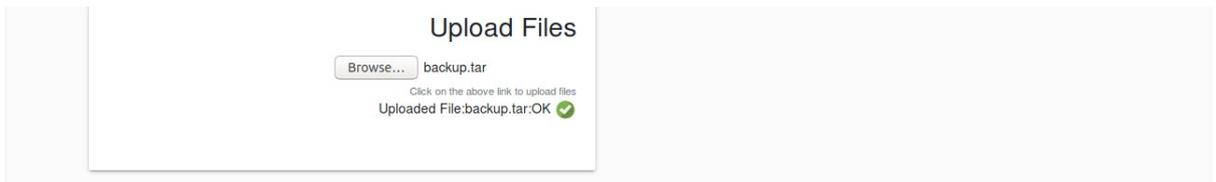
**Q** *Search* allows you to filter the list of files and folders for a text match that you enter in the search field.

### Uploading a File

To upload a file to your Wattmon, first go to the *File Manager*, and then navigate to the folder to wish to upload to, such as *config* or *scripts*. Click on *UPLOAD FILES* on the control bar. Click the *Browse* button, select one or more files (hold CTRL while clicking on a file to select more than one), or drag and drop files from your system file manager onto the *Browse* button.

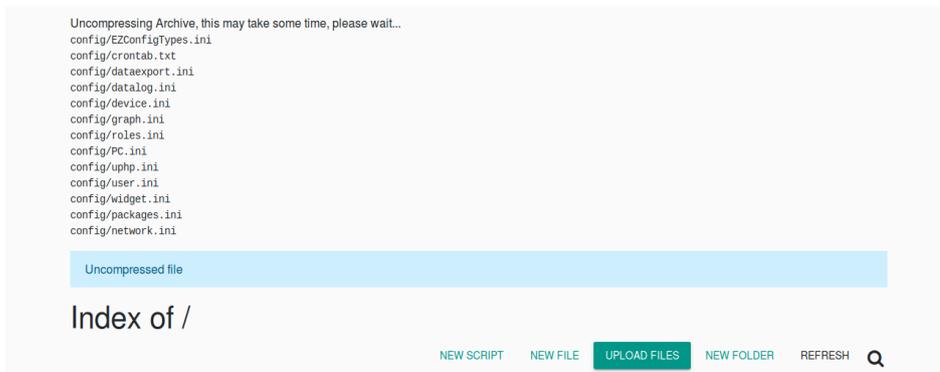
All configuration or back-up files on the WattmonOS are compressed with a *.tar* or *.tar.gz* extension, scripts have a *.cgi* or *.inc* extension and individual config files have a *.ini* extension.

When the uploading process starts, the file name will display next to the *Browse* button. Once the file has been uploaded successfully, you will see a message saying *Uploaded file:backup.tar:OK*, where *backup.tar* will be the file you uploaded, as shown below.



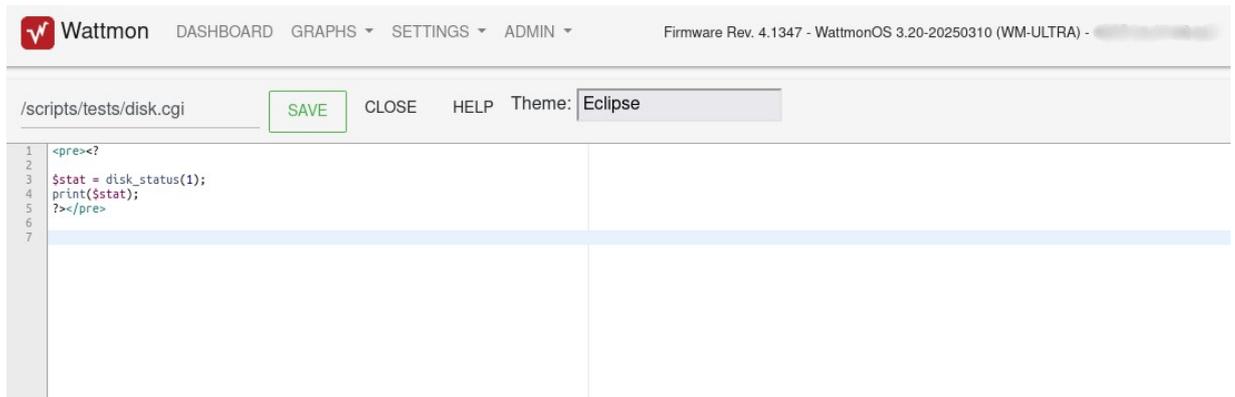
### Uncompressing a File

To extract a file, in this case *config.tar*, upload it as explained above and wait for it to show in the *File Manager* list, or *Refresh* the page, then click on one of the  *SD*  *FL* icons, *SD* to extract the files to the SD card or *FL* (flash) icon to extract to the external flash, and wait for the extraction to complete. For certain hardware models you may see just one suitcase icon indicating that it can only be extracted to a single drive (i.e. SD card). While extracting, the *File Manager* will display all the files being extracted at the top of the page, and at the end of the extraction process will give a confirmation message stating *Uncompressed file*.



## Editing a File using the File Editor

The file editor is shown either by clicking the **NEW FILE** button in the File Manager, or by clicking the *Edit* icon next to a file name that supports editing.

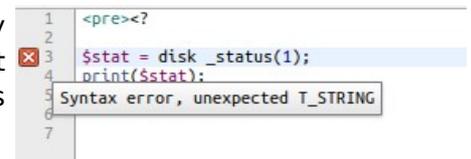


The file name is display above the edit area and can be modified. This is an easy way to save the file as a new name (make a a copy).

The editor uses syntax highlighting and automatically detects the file extension. When writing uPHP scripts, it will highlight issues as you type them which helps prevent issues.

Use the *Theme* drop down to switch between different color schemes for easier readability.

Click the *Save* button to apply your changes.



## Blockly Visual Editor

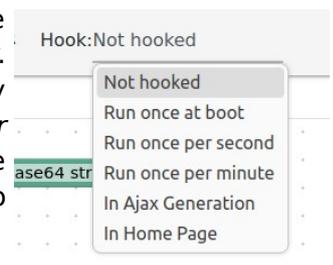
The visual editor based on Blockly allows you to build up scripts using a drag and drop system. Scripts then need to be Generated to convert them to uPHP scripts.

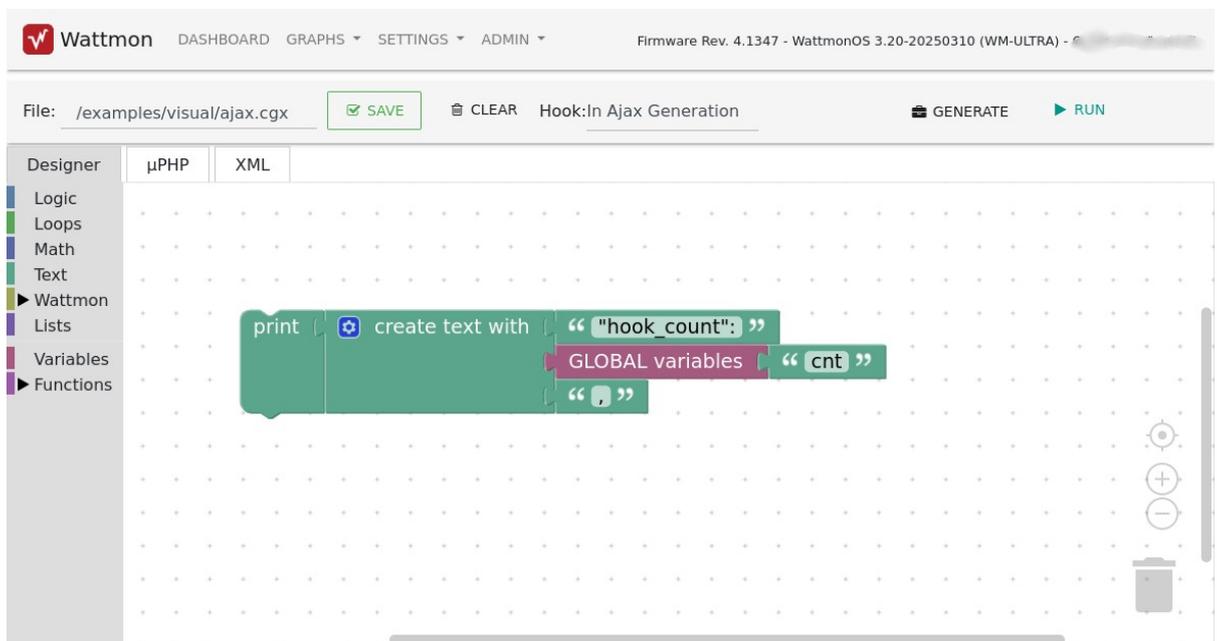
The file extension for visual script files is `.cgx` – you can modify a file name in the File field.

Clicking the **CLEAR** button on the toolbar wipes all objects on your canvas.

## Hooks

The Hooks drop down lets you select where the script should be executed. *Not hooked* means that it needs to be run manually. *Run once at boot* means that it will get executed just once every time the system restarts. The *Run once per second* and *once per minute* executes the script at a regular interval. To generate dynamic values for the dashboard, select *In Ajax Generation*. To generate a widget, select *In Home Page*.





The above example shows one of the sample scripts from the `/examples/visual/` folder. This prints out a simple string "hook\_count": followed by an incrementing value. By selecting In Ajax Generation, saving and clicking Generate this will be installed in the system hooks and always be sent across to the dynamic loading script (`ajaxdata.cgi`) for the main dashboard.

A description of using Blockly is beyond the scope of this document and can be easily obtained online. The logic blocks are available at the left under the Designer tab and can be dragged onto the canvas in order to build up code.

## Understanding the File System

The WattmonOS File system is organized into several folders. The `config` folder has all of the main system configuration settings. This folder normally will not get modified unless you change system parameters by updating them through the Control Panel.

### `/config/` Folder

The config folder has many files, each with its own purpose - these are structured as Windows INI files with a `[section]` heading and `key=value` parameters.

- `3g.ini` - Contains settings for LTE connections
- `boot.ini` - Defines the boot drive (0 or 1) for flash-enabled hardware
- `crontab.txt` - The system scheduler file with the scripts that run and their interval
- `csv_headers.txt` - The pre-built CSV headers used for generating log files and for exporting, this is automatically rebuilt when EZConfig is run.
- `dataexport.ini` - Data export settings for HTTP exporting to both Wattmon and custom servers
- `datalog.ini` - Contains the roles and devices that are to be logged in the CSV files.
- `device.ini` - Defines the devices that are configured in Wattmon.
- `dns.ini` - Predefined DNS lookup entries for Wattmon servers
- `ezconfig.ini` - Settings defined in EZConfig, which can be used to re-run the configuration and generate appropriate settings. Copy this file to another Wattmon and run EZConfig to generate an identical setup.
- `graph.ini` - Settings for the Wattmon internal graphs
- `groups.ini` - Defines the Wattmon internal groups
- `mail.ini` - SMTP email settings
- `network.ini` - LAN IP address settings
- `onewire.ini` - 1-Wire devices configuration

- ota.ini – Over the Air updates settings
- packages.ini – Installed packages
- roles.ini – System roles as defined in EZConfig, linking device parameters with the CSV entries.
- Snmp.ini – SNMP settings
- sysvars.ini – System variable definitions
- time.ini – RTC and system time settings
- uphp.ini – Firmware and OS configuration options
- usb.ini – Settings for the USB stack
- user.ini – Username and password settings
- widget.ini – Settings for dashboard widgets
- wifi.ini – Wi-Fi connections settings

#### **/app/ Folder**

The *app* folder contains the bulk of the WattmonOS scripts and is updated whenever you install a new WattmonOS versions

#### **/scripts/ Folder**

The *scripts* folder contains scripts that run from the CRON system as defined in the `crontab.txt` file, such as `cronmin.cgi`, `cronsec.cgi` and `cronday.cgi`. Several additional scripts that are useful for system administration are located in this folder. You can place your own development code here as well.

#### **/package/ Folder**

The *package* folder contains all additional packages that are installed alongside WattmonOS such as PC, FTP, MQTT, etc. Some of these come pre-installed while others can be downloaded and installed using the WattmonOS Updater feature.

#### **/dev/ Folder**

The *dev* folder contains all the modbus mappings for supported Wattmon devices (referred to as device types in Wattmon). Each file has a unique ID that is pre-defined by the team, i.e. 7094 is a Sungrow Inverter. Each of these device type files can be edited using the Control Panel > Device Types page.

#### **/ini/ Folder**

The *ini* folder contains system state information, such as the last export position for a CSV file or FTP file. The various packages and system code will store their commonly changing status values in this folder.

#### **/logs/ Folder**

The *logs* folder stores all the CSV and system logs. The main log file is called `log.txt`. Additional log files may be created depending on the configuration you choose. These would typically be: `network.txt`, `modbus.txt` and `mqtt.txt`. Daily CSV logged data is stored in a file in the path `YYYY/MM/YYYYMMDD_0.csv`. For example, the complete path to CSV data for Jan 7, 2024 would be: `/logs/2024/01/20240107_0.csv`. Read more on the log files in the following section.

There are several addition folders which are also used by Wattmon but describing each and every one of them is beyond the scope of this documentation.

## **TAIL LOGS AND SYSTEM LOGS**

The *Tail Log* and *System Log* menu options under the *Settings* menu display a log of all the potentially relevant information from the runtime functions of the Wattmon, specifically the scripts running on it and any errors or warnings.

The *System Log* file is rotated automatically to a new file every midnight at 00:00:00, to make it easier to manage each day worth of logs. The active *System Log* is always called *log.txt*, while historic *System Log* files have the format *logYYYYMMDD.txt* or *logYYYYM-MDD-X.txt* if multiple log rotations have been initiated within a particular day.

The *Tail Log* displays only the last ten thousand characters in the log file, making it the quickest to load, and the easiest to look through when searching for recent errors while debugging. In order to view more logs older than the last 10000 characters, you can use the navigation buttons shown below.

[Prev](#) Page 1 (88412-98412 of 98412) [Next](#) | [Main log](#) | [Network log](#) | [MQTT log](#) |

---

Depending on the configuration options in the uPHP settings page, there may be one or more log files. You can switch between the log file (i.e. Main, Network, MQTT) by clicking on it.

Each entry in a log file contains a log message, starting with a human-readable date and time followed by the alert level (Error, Warning) in most cases.

To understand issues such as system reboots, you can read up in the section *Troubleshooting Reboots* further in this document.

## Internet Connectivity Information

Wattmon connects to the Internet through both LAN, WiFi or LTE, either via a USB dongle or via an internal LTE module depending on the model. The LTE interface has a higher priority than LAN, meaning that if a connection is established via SIM card it will use that route over the LAN route.

Depending on the log level you will see more or less messages pertaining to connection status, especially while the dongle is connecting. Setting the log level to *Verbose* in the *uPHP Settings Page* will result in most PPP connection messages to be logged, allowing for more detailed debugging.

A typical connection for a 3G/4G dongle would appear as follows. See comments inline:

```
Sat Mar 10 16:11:49 2018: Notice: Init PPP module
Sat Mar 10 16:12:01 2018: Notice: PPP connecting
Sat Mar 10 16:12:11 2018: Notice: ppp: Error in connect - memFree=15904
Sat Mar 10 16:12:11 2018: Notice: ppp: [out] LCP REQ
```

^ This initiates the LCP protocol. If this line is present, the dongle is properly recognized, and the USB CDC channel is configured.

```
Sat Mar 10 16:12:11 2018: Notice: ppp: [in] LCP REQ
Sat Mar 10 16:12:11 2018: Notice: Opt 8=7 2
Sat Mar 10 16:12:11 2018: Notice: Opt 7=2 6
Sat Mar 10 16:12:11 2018: Notice: Opt 2=0 0
Sat Mar 10 16:12:11 2018: Notice: Opt 1=5 DC
Sat Mar 10 16:12:11 2018: Notice: Opt 5=0 0
Sat Mar 10 16:12:11 2018: Notice: Opt 3=C2 23
Sat Mar 10 16:12:11 2018: Notice: BAD 1=5 DC
Sat Mar 10 16:12:11 2018: Notice: BAD 5=0 0
Sat Mar 10 16:12:11 2018: Notice: BAD 3=C2 23
Sat Mar 10 16:12:11 2018: Notice: ppp: [out] LCP REQ REJ
Sat Mar 10 16:12:11 2018: Notice: ppp: [in] LCP ACK
Sat Mar 10 16:12:11 2018: Notice: ppp: LCP layer UP
Sat Mar 10 16:12:11 2018: Notice: ppp: [in] LCP REQ
Sat Mar 10 16:12:11 2018: Notice: Opt 8=7 2
Sat Mar 10 16:12:11 2018: Notice: Opt 7=2 6
```

```

Sat Mar 10 16:12:11 2018: Notice: Opt 2=0 0
Sat Mar 10 16:12:11 2018: Notice: BAD 7=2 6
Sat Mar 10 16:12:11 2018: Notice: ppp: [out] LCP REQ REJ
Sat Mar 10 16:12:11 2018: Notice: ppp: [in] LCP REQ
Sat Mar 10 16:12:11 2018: Notice: Opt 8=2 6
Sat Mar 10 16:12:11 2018: Notice: Opt 2=0 0
Sat Mar 10 16:12:12 2018: Notice: ppp: [out] LCP REQ ACK

```

^ The above sequence takes care of LCP protocol negotiation. We do not support CHAP and hence it goes through two cycles, until PAP (Password Authentication Protocol) is activated.

```

Sat Mar 10 16:12:12 2018: Notice: ppp: [in] IPCP REQ [seq=1] [len=4]
[type=187]
Sat Mar 10 16:12:12 2018: Notice: ppp: [out] IPCP ACK [2.0.8.10] [seq=1]
Sat Mar 10 16:12:16 2018: Notice: ppp: [out] IPCP REQ [0.0.0.0]
Sat Mar 10 16:12:16 2018: Notice: ppp: [in] IPCP NAK
Sat Mar 10 16:12:16 2018: Notice: ppp: [in] IPCP NAK IP=[2.0.8.10]
Sat Mar 10 16:12:18 2018: Notice: ppp: [in] IPCP REQ [seq=2] [len=4]
[type=223]
Sat Mar 10 16:12:18 2018: Notice: ppp: [out] IPCP ACK [2.0.8.10] [seq=2]
Sat Mar 10 16:12:20 2018: Notice: ppp: [out] IPCP REQ IP [2.0.8.10]
Sat Mar 10 16:12:21 2018: Notice: ppp: [in] IPCP NAK
Sat Mar 10 16:12:21 2018: Notice: ppp: [in] IPCP NAK IP=[2.0.8.10]
Sat Mar 10 16:12:24 2018: Notice: ppp: [in] IPCP REQ [seq=3] [len=4]
[type=3]
Sat Mar 10 16:12:24 2018: Notice: ppp: [out] IPCP ACK [2.0.8.10] [seq=3]
Sat Mar 10 16:12:25 2018: Notice: ppp: [out] IPCP REQ IP [2.0.8.10]
Sat Mar 10 16:12:27 2018: Notice: ppp: [in] IPCP NAK
Sat Mar 10 16:12:27 2018: Notice: ppp: [in] IPCP NAK IP=[2.0.8.10]
Sat Mar 10 16:12:30 2018: Notice: ppp: [in] IPCP REQ [seq=4] [len=4]
[type=6]
Sat Mar 10 16:12:30 2018: Notice: ppp: [out] IPCP ACK [2.0.8.10] [seq=4]
Sat Mar 10 16:12:31 2018: Notice: ppp: [out] IPCP REQ IP [2.0.8.10]
Sat Mar 10 16:12:31 2018: Notice: ppp: [in] IPCP NAK
Sat Mar 10 16:12:31 2018: Notice: ppp: [in] IPCP NAK IP=[10.17.101.47]
Sat Mar 10 16:12:31 2018: Notice: ppp: [out] IPCP REQ IP [10.17.101.47]
Sat Mar 10 16:12:31 2018: Notice: ppp: [in] IPCP ACK
Sat Mar 10 16:12:31 2018: Notice: ppp: [in] IP = 10.17.101.47 [seq=5]
Sat Mar 10 16:12:31 2018: Notice: ppp: [in] DNS = 218.248.112.72 [seq=5]
Sat Mar 10 16:12:31 2018: Notice: ppp: IPCP layer UP

```

The final part of the negotiation happens at the IPCP layer while an IP address is obtained. The temporary IP address of 2.0.8.10 is just a dummy IP address and it will keep requesting a new address until it obtains a valid one, in this case 10.17.101.47, at which point the IP Layer is fully established and the default route is set to the new interface.

## Troubleshooting Connectivity Issues

Assuming a connection is properly established as in the example above, at some point the provider may decide to terminate the connection without warning, or the signal may drop. At this point, the Wattmon will not be able to connect to remote sockets, and you will see a pattern as below:

```

Thu Mar 08 11:56:50 2018: Warning on ip.cgi line 378: Socket open timed out!
Thu Mar 08 11:58:02 2018: Warning on ip.cgi line 378: Socket open timed out!

```

```
Thu Mar 08 11:59:13 2018: Warning on ip.cgi line 378: Socket open timed out!
Thu Mar 08 12:00:22 2018: Warning on ip.cgi line 378: Socket open timed out!
Thu Mar 08 12:01:49 2018: Warning on ip.cgi line 378: Socket open timed out!
Thu Mar 08 12:01:49 2018: Notice on ip.cgi line 612: Cellular link seems to
have died, restarting USB power
```

It will try reconnecting several times and eventually will reboot the dongle. Unless there is a connectivity issue on site where the network signal is not present, this will allow for a new IP address to be obtained and bring up the connection again.

In some cases, the dongle actually terminates the connection in which case the Wattmon will immediately try to re-negotiate a new IP address.

```
Fri Mar 09 10:35:26 2018: Notice: ppp: [in] LCP TERM
Fri Mar 09 10:35:26 2018: Notice: usb: Device detached
```

## Troubleshooting Reboots

There are several reasons why a Wattmon may reboot: Physical power cycle, software-initiated restart, watchdog timeout, or access violation. In order to determine the cause, search through the log for the following lines:

```
Thu Mar 27 12:07:01 2025: Warning: RCON=8000040 h=0
Thu Mar 27 12:07:01 2025: Warning: Software initiated reset
Thu Mar 27 12:07:01 2025: System booted on drive 0, DDR Errors=0
Thu Mar 27 12:07:01 2025: Notice on /scripts/runonce.cgi (/scripts/
runonce.cgi) line 59 : Wattmon Firmware 4.1347 Software WattmonOS 3.20-
20250310 (WM-ULTRA) (XX-XX-XX-XX-XX-XX)
```

The first line gives you the RCON register (in the chip) value and the second line gives the human-readable interpretation of that value. In the above example the user most likely pressed the reset button in the WattmonOS web interface.

```
Wed Mar 14 10:40:03 2018: Warning: RCON=C8000003
Wed Mar 14 10:40:03 2018: Warning: Normal power up
Wed Mar 14 10:40:03 2018: System booted
```

In the above example, power was physically removed and restored to the Wattmon such as what would be typical with a power cut.

In the above example, a code error or memory corruption caused the chip to get stuck in a loop forcing a restart. A firmware update with patch for the issue may be required if you see this regularly.

When a software exception occurs, usually the address of the issue is also included in the log file for debugging purposes as with the following example:

```
Fri Mar 09 11:01:19 2018: Exception at 9D079F68
Fri Mar 09 11:01:19 2018: bus error (load/store)
Fri Mar 09 11:01:20 2018: Warning: RCON=40
Fri Mar 09 11:01:20 2018: Warning: Software initiated reset
```

## Miscellaneous Issues

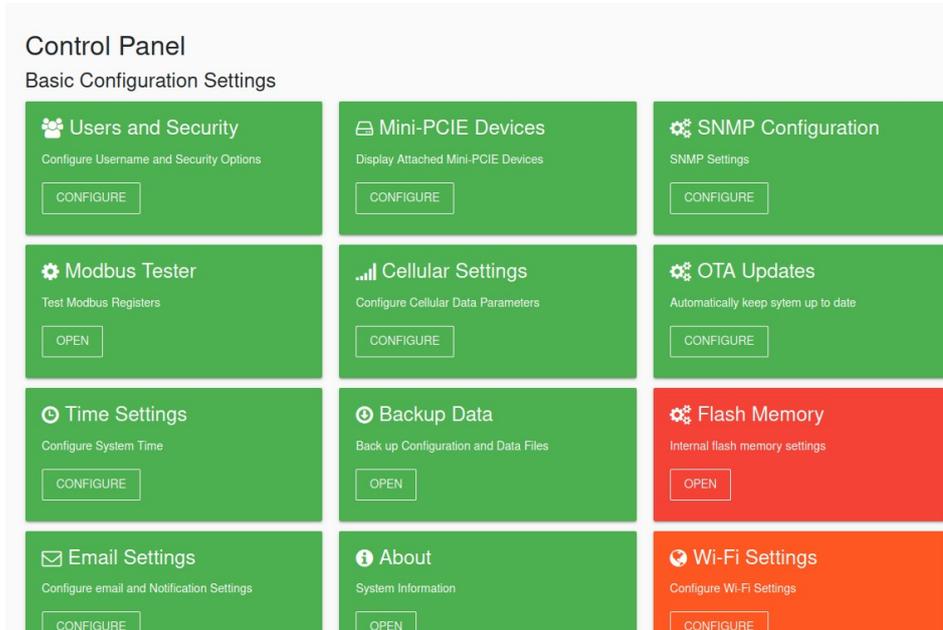
When a non-recoverable script errors, it will show the type of error in the script and the line number as shown below:

```
Fri Mar 09 09:37:16 2018: Error on cronsec.cgi line 238: Missing } in in-
clude file
Fri Mar 09 09:37:16 2018: [uphp] Force recompile cronsec.cgc
```

The Force recompile line indicates that the script file, in this case *cronsec.cgi*, will be recompiled the next time it is run - this will resolve any issues of a corrupted compiled file.

# BASIC CONFIGURATION SETTINGS- CONTROL PANEL

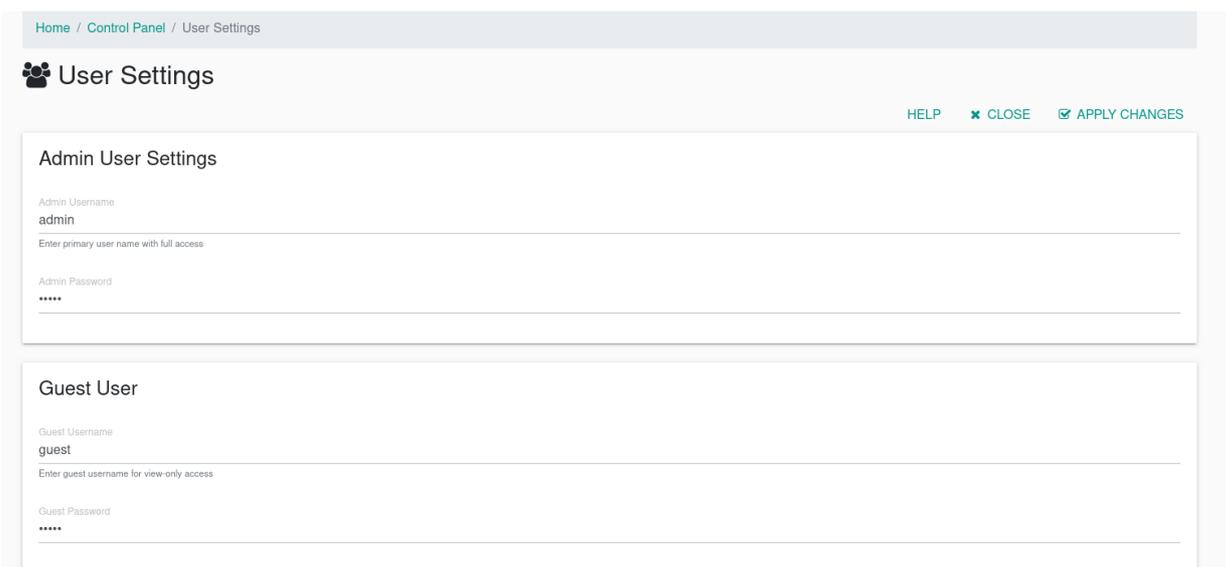
The menu bar at the very top of the page has several options including the *SETTINGS* drop-down menu. Select *Control Panel* from the *SETTINGS* drop-down, and you will come to a page with a large variety of color-coded blocks, marking the importance of each setting type – the red being critical, the orange being important, and the green being safe to change, although beware that changing any setting without proper understand could lead to malfunctioning.



## USERS AND SECURITY

The *User Settings* allow you to change your default login credentials, set up a Guest account, change the name of your Wattmon device, and set the default language.

The default credentials for the Admin account are *admin* and *admin*. The default credentials for the Guest account are *guest* and *guest*.



The *Device Name* field lets you choose a name for your Wattmon which will be displayed at the top of the browser window and when you add it to the Wattmon App.

*Kiosk Mode* allows you to enable a view without logging in for display on an HMI.

Device Settings

Device Name  
Choose a name for the device if you wish to customize it.

Kiosk Mode  
Disabled  
Enable kiosk mode to view the widgets on the home page without logging in with /index.cgi?kiosk=1

EZConfig Lite  
Disabled  
Allow guest users to reconfigure inverters using EZConfig lite

Language  
English  
Choose the language for the Wattmon interface

*EZConfig Lite* enables an additional option in the menu when logged in as a guest to configure basic parameters such as the number of inverters without admin access.

The languages supported by Wattmon as of now are English, Spanish, German, Hindi and Tamil. The interface is changed into the aforementioned languages in varying degrees, while the actual settings and scripts stay unchanged.

To save the settings, click *Apply Changes* and wait for a confirmation message. You will need to reboot the Wattmon for the changes to take effect.

## USB DEVICES / MINI-PCIE DEVICES

In this setting page you have the option to enable the use of a USB module to connect to the Internet.

The default setting will display USB Engine as *Enabled*.

Home / Control Panel / USB Settings

USB Settings

HELP CLOSE REFRESH APPLY CHANGES

USB Settings

USB Engine  
 Disabled  
 Enabled  
Enable the USB engine if you wish to connect a cellular dongle to Wattmon

Connected Devices

Status	Device attached.
Device ID	12D1:1506
Device	Huawei E3372 (Modem)
Supports CDC	1

Under *Connected Devices* the *Status* will say *Device attached*, along with details of the USB Device, such as *Device ID*, *Device* model, and if it *Supports CDC*.

Only a select number of USB Dongle models are supported: *Huawei E303F*, *Huawei E3531*, *Huawei E3531i*, *Huawei E3372* and *Huawei E3276*. Once the device is connected to the Internet, the LED on the stick will glow solid blue or green. If it is blinking, it could either be in the process of connecting or facing a network-related issues. The MEGA2-4G, MINI3-4G and ULTRA all come with Quectel EC-25E modules.

# SNMP CONFIGURATION

SNMP (Simple Network Management Protocol) lets you interface with networking equipment and controllers to both read and write information.

In the SNMP Configuration, set the SNMP Status to *Enabled* to activate the feature. Choose a community name for the Read and Write operations. The defaults are `public` and `private` respectively. Wattmon supports SNMP v2c.

SNMP Settings

HELP CLOSE APPLY CHANGES

### SNMP Configuration

SNMP Status: Disabled  
Enable the SNMP agent to allow for remote SNMP read/write operations to the roles.

Read Community Name: public  
The default read community is 'public'

Write Community Name: private  
The default write community is 'private'

### SNMP Trap Configuration

SNMP Trap Status: Disabled  
Enable the SNMP trap to allow this device to send trap messages to an SNMP manager

Trap Community Name: private  
The default read community is 'public'

SNMP Manager IP Address: 192.168.0.7  
The IP address of the SNMP manager

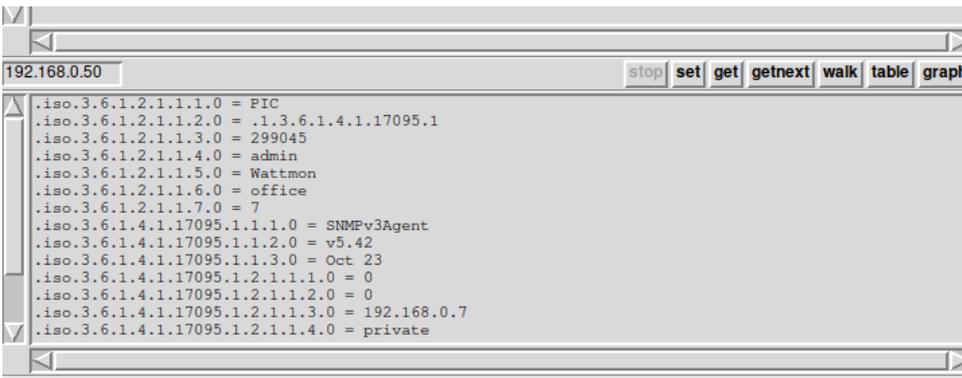
## Using SNMP

If you are using Ubuntu, install `snmpwalk` (`apt-get install snmp`). Then run it using the IP of your wattmon:

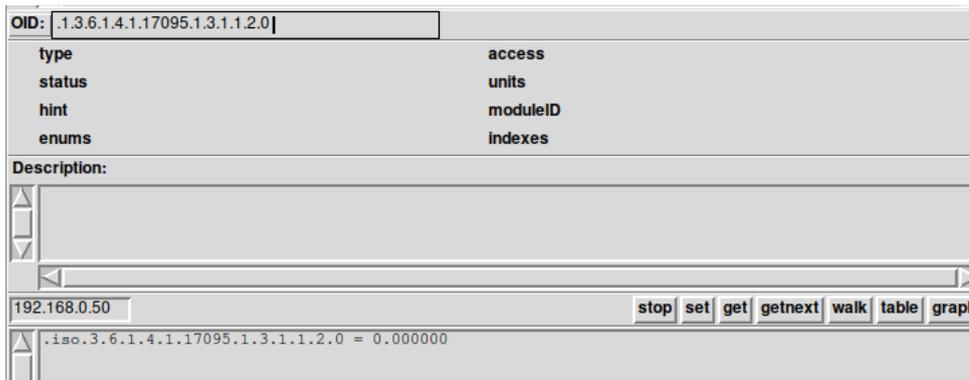
```
snmpwalk -c public -v1 192.168.0.55 | less
```

If it works you will get some info about the device:

```
iso.3.6.1.2.1.1.1.0 = STRING: "PIC"  
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.17095.1  
iso.3.6.1.2.1.1.3.0 = Timeticks: (203320) 0:33:53.20  
iso.3.6.1.2.1.1.4.0 = STRING: "admin"  
iso.3.6.1.2.1.1.5.0 = STRING: "Wattmon"  
iso.3.6.1.2.1.1.6.0 = STRING: "office"  
iso.3.6.1.2.1.1.7.0 = INTEGER: 7
```



You can also use `tkmib` which is a linux GUI tool. This shows that SNMP is working. You can access the `wattmon` roles in the `.1.3.6.1.4.1.17095.1.3.1.1.2.x` range. (`.1.3.6.1.4.1.17095.1.3.1.1.2.0` is role 0 value, `.1.3.6.1.4.1.17095.1.3.1.1.2.1` is role 1 value)



## SNMP Traps

An SNMP Trap is an unsolicited message sent by an SNMP-enabled device (Wattmon in this case) to an SNMP manager to report events. Unlike SNMP polling, where the manager requests data, traps are pushed from the device when an event occurs. Set the trap community to `private` and enter the IP address of the SNMP manager after enabling the SNMP trap option. In order to send messages via SNMP Traps, you will need to use a `wattmon` Action (refer to the *Actions* section) or use the `snmp_trap_send()` uPHP function in a script.

## MODBUS TESTER

Wattmon comes with a built-in *Modbus Tester*, which allows you to test a Modbus connection with your devices by polling certain Modbus Registers, or to set a specific register value in a device. In order to poll a particular device and perform a Modbus test, make sure to keep handy as much Modbus-related information you have concerning that particular device, such as the Modbus ID, Baud Rate, Parity, etc. It is particularly important to have the Modbus Register Mapping from the manufacturer, as without the actual Modbus Registers, it is impossible to obtain a specific value through the Modbus Tester.

Access the *Modbus Tester* page from *Control Panel* or by clicking *Options > Modbus Tester* in the *Devices* page. The *Modbus Tester* page is displayed in two columns the left which lists the *Modbus Parameters* you wish to test, and the right which displays the *Results* from that Modbus poll.

### Modbus Parameters Section

The Parameters section lists the options for *Channel*, *Baud Rate*, *Parity*, *Start Register Number*, *Number of Registers*, *Modbus Device ID*, *Modbus Function* and *Register Type*.

The *Channel* option contains a drop-down that lists the various physical buses supported by the Wattmon, that a Modbus device is connected to. The options available are *Modbus RTU (RS-485)*, *Modbus RTU 2 (on hardware that supports it)*, and *TCP Channel 1* through *10(30)*, depending on your Wattmon. The Modbus TCP Channels are displayed along with their IP address and port number. If a channel is connected, you will see a *Connected* label next to it. Select the bus your device is connected to and then proceed to the next step.

The baud rate and parity use the system defaults configured in the Modbus Configuration page. Change it only if you need to test if a device is reachable on a different setting.

In the *Start Register Number (Decimal)* option, you need to enter the Modbus Register number in Decimal form. You will be required to enter the actual value of the register here, so in some cases you may need to subtract 1 or 30001 or 40001 from the register value presented in by the device manufacturer. For example if modscan (a Windows based tool) works with register 3000 then in the Wattmon you would use address 2999.

In the *Number of Registers* option, you need to enter the number of registers to read at a given time. This too will be as per the Modbus Register Mapping. If a value spans multiple registers (I.e a long int takes 2 registers), then reading just 1 register will cause an excep-

The screenshot shows the Modbus Tester interface. The 'Parameters' section is filled with the following values: Channel: Modbus RTU (RS-485), Baud rate: 9600, Parity: No Parity (selected), Start Register Number (Decimal): 40001, Number of registers: 1, Modbus Device ID: 1, Modbus Function: 03 - Read Holding Registers, and Register Type: Signed 16-bit. The 'Results' section shows a TX status bar and a red error message: 'No response from device in 6410ms.'

tion.

The *Modbus Device ID* option requires for you to enter the Modbus ID of the device you wish to communicate with.

The next option is again a drop-down, called Modbus Function. The function code is the number which decides which function to perform when reading or writing data from a device. The Wattmon supports five function codes – *01 Read Coil Status*, *02 Read Digital Input Status*, *03 Read Holding Registers*, *04 Read Input Registers*, and *06 Write Single Register*, *16 Write Multiple Registers*. The function code for your device will be again as per the Modbus Register Mapping.

The last option is *Register Type*, which is for selecting the formatting of the data from the device you are connected to. For most single registers, the commonly used format is *Unsigned 16-bit*, while for double registers make sure that you choose the correct Endianness. For example *Unsigned 32-bit (Big Endian)* is 2 registers long with the bigger register (higher 16 bits) first. *Unsigned 32-bit (Little Endian)* is also 2 register long with the lower 16 bits first. The correct format of each register will be mentioned in the Modbus Register Mapping.

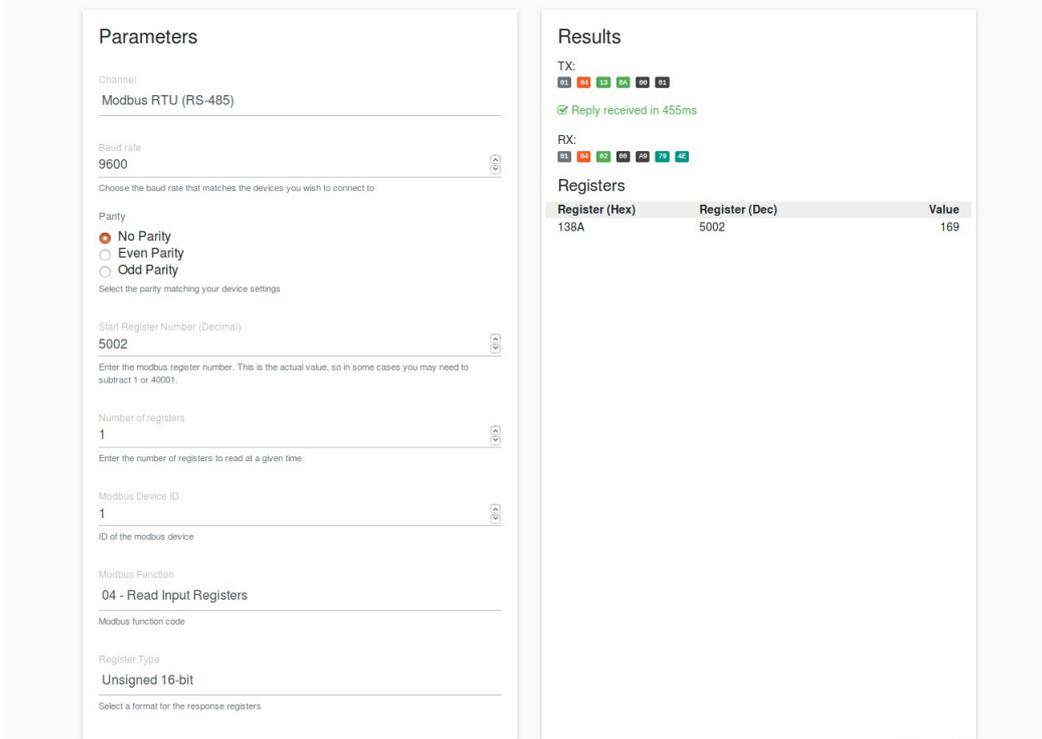
The *Results* section shows the Modbus command sent by the Wattmon as per the particulars selected in the *Parameters* section, as well as the response from the Modbus device you are connected to.

### Modbus Test Examples

If the Modbus device connected to the Wattmon is a Sungrow SG60KTL PV inverter, you would first need to verify the *Modbus Channel*, *Baud Rate*, *Parity* and *Modbus ID*. Once those are known enter them on the *Modbus Tester* page. Then refer to the Modbus Register mapping from Sungrow for the remaining details, as shown below.

No.	Name	Address	Data	Data range	Unit	Remark	Address
4	Daily energy	5003	U16		0.1kWh		3x
5	Total energy	5004 ~ 5005	U32		kWh		3x

In the case of Sungrow, the actual registers are known by subtracting 1 from the shared Address. So when polling today's generation, or *Daily energy* values, set the *Start Register Number (Decimal)* as 5002. The function code is 04 as per the register mapping, so the *Modbus Function* is set as *04 Read Input Registers*. The final option is the *Register Type*, which is detailed in the image above as U16, which is *Unsigned 16-bit*. Once all the values are filled in the Parameters column, click on *EXECUTE* to poll the Modbus device, which is the Sungrow Inverter. The *Parameters* entered correctly, and the *Result*, if successful, will look appear as in the following image.



If the same Wattmon has another device connected to that needs to be tested, in that case, repeat the procedure as above, while noting the differences in certain settings. If the second Modbus device connected to the Wattmon is a SMA Solid-Q 50 inverter, you would first need to verify the *Modbus Channel*, *Baud Rate*, *Parity* and *Modbus ID*. Note that the *Channel* can be the same, but does not have to. The *Baud Rate* and *Parity* have to be the same as the previous device, and the *Modbus ID* has to be different. Once those are known enter them on the *Modbus Tester* page. Then refer to the Modbus Register map-

### 3.3 SMA-China Modbus Profile - Register Overview

#### Read-Only Registers

ADR (DEC)	Description/number code	Type	Unit	Gain	Access
30002	E-Today	U32	kwh	0.1	RO
30004	E-Total	U32	kwh	0.1	RO

ping from SMA New Energy for the remaining details, as shown below.

In the SMA Solid-Q register mapping, the actual registers are known by subtracting 30001 from the shared Address. So when polling today's generation or *E-today* values, set the

*Start Register Number (Decimal)* as 1. The function code is 04 as per the register mapping, so the *Modbus Function* is set as *04 Read Input Registers*. The final option is the *Register Type*, which is detailed in the image above as U32, which can be either *Unsigned 32-bit (Big Endian)* or *Unsigned 32-bit (Little Endian)*. Check within the registers document to confirm which of the two it is.

Once all the values are filled in the Parameters column, click on EXECUTE to poll the Modbus device, which is now the SMA Inverter. The *Parameters* entered correctly, and the *Result*, if successful, will look like in the following image.

The screenshot shows a web interface for configuring a Modbus poll. The left panel, titled 'Parameters', contains the following settings:

- Channel: Modbus RTU (RS-485)
- Baud rate: 9600
- Parity: No Parity (selected), Even Parity, Odd Parity
- Start Register Number (Decimal): 1
- Number of registers: 2
- Modbus Device ID: 3
- Modbus Function: 04 - Read Input Registers
- Register Type: Unsigned 32-bit int (Big endian)

The right panel, titled 'Results', shows the following information:

- TX: [Hex data]
- Reply received in 445ms
- RX: [Hex data]
- Registers table:

Register (Hex)	Register (Dec)	Value
0001	1	96

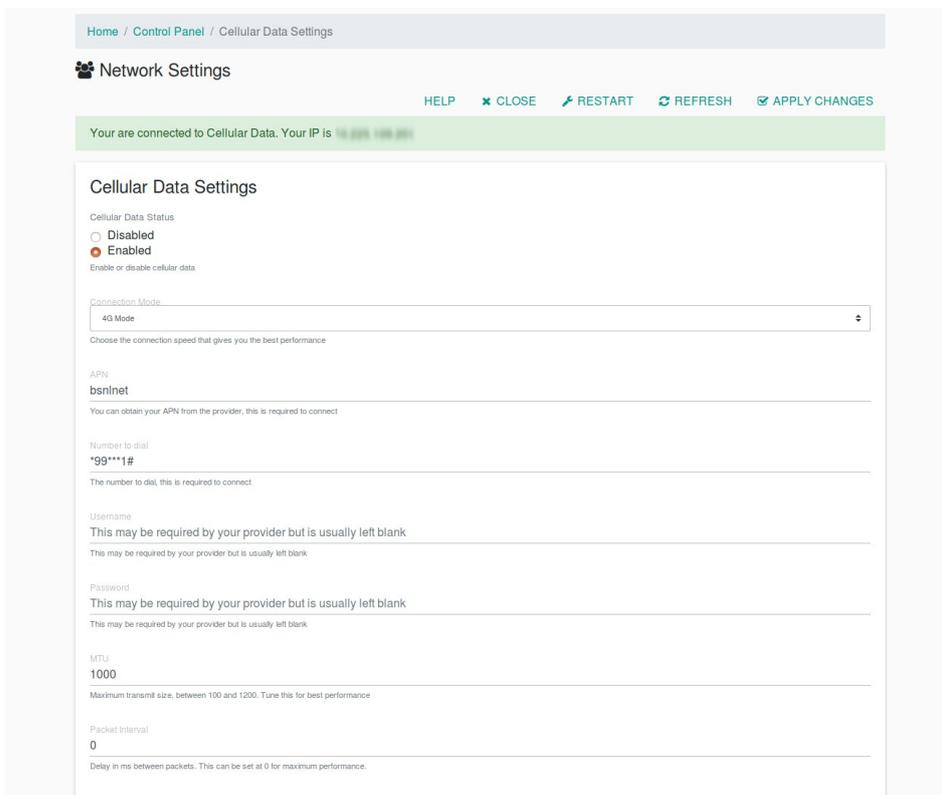
If the Modbus poll is unsuccessful, you will be greeted with the message saying **No response from the device in 0000ms**, in which case you need to verify if the device is properly connected and if the *Baud Rate*, *Parity* and *ID* is correctly set.

If the Modbus poll says **Modbus Exception**, the device is connected and responding, but the *Start Register Number (Decimal)*, or the *Modbus Function*, or the *Register Type* is erroneous and needs to be corrected.

## CELLULAR SETTINGS

The *Cellular Data Settings* page configures the connection mode and APN based on the network provider.

The status of the connection is displayed above the modifiable settings - in a green overlay stating **You are connected to Cellular Data. Your IP is xx.xx.xx.xx** if your Wattmon is connected to the internet; and if unable to connect to the internet, you will see a message stating *You are not connected to Cellular Data*.



There are several settings available on the page as seen above – *Cellular Data Status*, *Connection Mode*, *APN*, *Number to Dial*, *Username*, *Password*, *MTU* and *Packet Interval*, which are all visible only when the *Cellular Data Status* is set to *Enabled*.

The *Connection Mode* drop-down offers five options - *GPRS/Edge 2G Mode Only*, *3G Mode Only*, *Prefer GPRS/Edge 2G Mode*, *Prefer 3G Mode* and *4G Mode*. The default is set to *Prefer 3G mode*, as that automatically connects to 3G or 4G when available and otherwise automatically shifts to GPRS/EDGE, thereby choosing the optimum speeds. Alternatively, the *4G Mode* forces the stick to work on 4G LTE depending on the availability of bandwidth in the region. When using the Quectel internal Module this option is not functional.

The remainder of the settings are network provider dependent, and with the exception of the APN, are identical for most providers. The *Number to Dial* field should be set to *\*99\*\*\*1#*. The *Username* and *Password* fields can be left blank. The *MTU* or maximum transmission unit is set by default to *1000* but can be set to anywhere between 100 and 1200 and the *Packet Interval* to *1*.

The APN for some of the network providers in India are as follows: Airtel – *airtelgprs.com*; Vodafone – *www*; BSNL – *bsnlnet*; Reliance Jio – *jionet*.

## OTA UPDATES

OTA (Over the Air) Updates let you update one or more parts of Wattmon automatically. Access the OTA Settings page from *Control Panel > OTA Updates*.

To use OTA, Select *Enabled* in the *Over the air updates* dropdown.

The *OTA check interval* is 1 day (86400 seconds). This can be changed if required.

Enable *Automatically Install Updates* to activate the automatic install feature after downloading the update. If you set this to disabled, updates will be downloaded but not installed.

If you are using the Automatic healing features, set *Automatically Resync Filesystem* to *Enabled*.

You can receive an email update when an install is completed if you set the *Send Email Updates* to enabled, but this will require that the SMTP settings are properly configured beforehand.

The screenshot shows a settings window with a title bar containing 'HELP', 'CLOSE', 'CHECK NOW', and 'APPLY CHANGES'. The settings are organized into several sections, each with a dropdown menu and a descriptive text below it:

- Over the air updates:** Set to 'Enabled'. Description: 'Enable OTA updates if you wish to keep your Wattmon automatically updated.'
- OTA Check Interval (seconds):** Set to '86400'. Description: 'Choose how often the Wattmon polls the server to see if updates are available'. A 'CHOOSE INTERVAL' button is visible.
- Automatically install updates:** Set to 'Enabled'. Description: 'Enable this if you want to automatically extract and install the update.'
- Automatically Resync Filesystem:** Set to 'Enabled'. Description: 'Automatically Resync Filesystem when a package is installed.'
- Send email updates:** Set to 'Disabled'. Description: 'Send email updates when a package or firmware is automatically installed.'
- OTA for the firmware:** Set to 'Disabled'. Description: 'Automatically keep the firmware at the latest version.'
- OTA for packages:** Set to 'Disabled'. Description: 'Automatically keep the installed packages at the latest version.'
- OTA for custom packages:** Set to 'Disabled'. Description: 'Automatically keep the installed custom packages at the latest version.'

To automatically update the Wattmon firmware with the latest online version, set *OTA for the firmware* to Enabled.

To update the additional packages automatically (i.e. FTP, PC), set *OTA for packages* to Enabled.

If you have a large number of Wattmons and you wish to update your code automatically you can create your repository with your packages, and select *OTA for custom packages* to Enabled. This will check your repository and compare the package version to the one installed, and update as and when required. For more information on this, please contact the Wattmon team.

Click the *Check Now* button to connect to the update server and start an update in case any new firmware or package is available. To save your settings, click the *Apply* button. If OTA was previously disabled, a system restart will be required as the task scheduler needs to be updated.

## DATE AND TIME SETTINGS

If you find the Date and Time Settings on your device to be erroneous, or wish to change the time zone, you can change settings of your Wattmon as shown in this section. Wattmon stores timestamped data in CSV log files using a non-standard method – it stores the actual local time, not the UTC time. If you are exporting data to a custom server and face issues, you can set the local timezone on wattmon to UTC.

There are several options such as *Timekeeping Method*, *System Date*, *System Time*, *Use Wattmon Time Server*, and *UTC Offset*.

The *Timekeeping Method* has 4 options: Internal RTC Clock, SNTP Server, Time Server, No RTC.

**Date and Time**

HELP   x CLOSE   ✓ SYNC   ✓ APPLY CHANGES

### Time Settings

Timekeeping Method

- Internal RTC Clock
- SNTP Server (Does not work with 3G/4G)  
This will automatically keep your Wattmons time up to date using an Internet time server. Only works with LAN networks.
- Time Server (works with 3G/4G dongles)  
Use a TCP-based time server to synchronize your clock using the older Time protocol
- No RTC  
If you are experiencing issues with the internal real time clock, select this option

System Date  
02 / 04 / 2025  
Enter Date (DD / MM / YYYY)

System Time  
11 : 23 : 48  
Enter Time (hh:mm:ss)

Use Wattmon Time Server

- Disabled
- Enabled  
Update time once a day over HTTP from the Wattmon time server

UTC Offset  
[UTC + 5:30] Indian Standard Time, Sri Lanka Time  
Enter a UTC Offset in seconds

- Internal RTC Clock – use only the internal clock to keep time
- SNTP Server – sync with a remote server over SNTP (you will need to specify the server below). This does not work with many LTE connections because UDP packets don't seem to forward properly over the LTE links.
- Time Server – use an older Time protocol that works over LTE links
- No RTC – use this if your internal RTC has issues – the time will sync at boot to the wattmon time server.

The *System Date* is entered in the DD/MM/YYYY format. The *System Time* is entered in the hour format, which is hh:mm:ss.

We recommend that you keep the option *Use Wattmon Time Server* as *Enabled*, as that allows the Wattmon to automatically check and update the time on your device on a daily basis. This is the default setting

The default time zone on a Wattmon is Indian Standard Time (UTC +5:30), however in certain cases can come pre-configured to the time zone you are in.

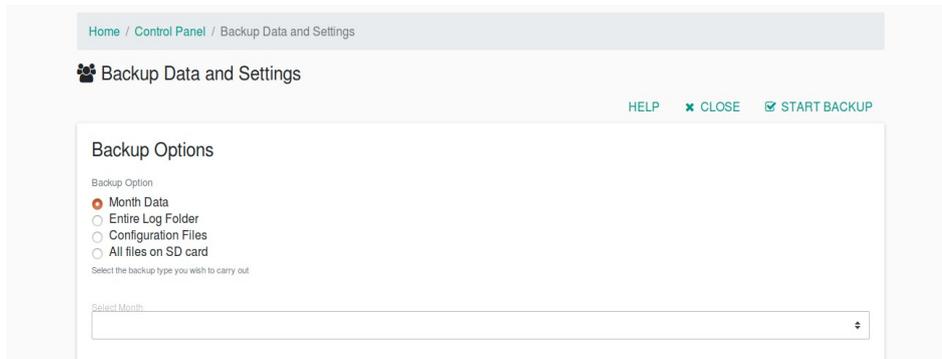
Alternatively, if the default settings have not been changed, and you can select the *Sync Time* option from the control bar to force a time update from the Wattmon Server.

## BACKUP DATA

*Backup Data and Settings* enables you to back up your configurations by creating a .tar file with the selected data on the Wattmon. You can choose to backup entire data or specific parts of it.

There are four *Backup Options*: *Month Data*, *Entire Log Folder*, *Configuration Files*, and *All Files on SD Card*.

*Month Data* backs up the logged data for a specific month. If you select it, a drop-down allows you to select the month you wish to download the data for.



The *Entire Log Folder* option downloads the logs folder, as seen in the File Manager, which includes the complete logged data as well as the System Logs recorded since your Wattmon was setup and running.

Selecting *Configuration Files* backs up the configuration settings of your Wattmon. These are most essential as they are very often unique to your device based on your setup.

*All Files on SD Card* backs up everything on the microSD card, as is visible in the File Manager.

Upon selecting any of the four provided options, you can click on *Start Backup* and the download link of the backup of your choice will show in a pop-up window. Download the *backupxxxxx.tar* file onto your computer and keep it safe, in case a restore is ever required. *All Files on SD Card* and *Entire Log Folder* could result in a large file and may take a while to generate and download.

**Note:** We recommend that you make a backup of your configuration once you have set your system up, or even in case you received it preconfigured, so that restoring it, if required, will be a very straightforward process.

All backup files created are saved on the microSD card and can be accessed from the backup folder using the File Manager. They can be downloaded from there and then loaded onto the Wattmon to restore any old configuration settings or logged data.

To restore the backed-up config or logs files, you may refer to the procedure for uploading the downloaded file onto your device through the **File Manager**.

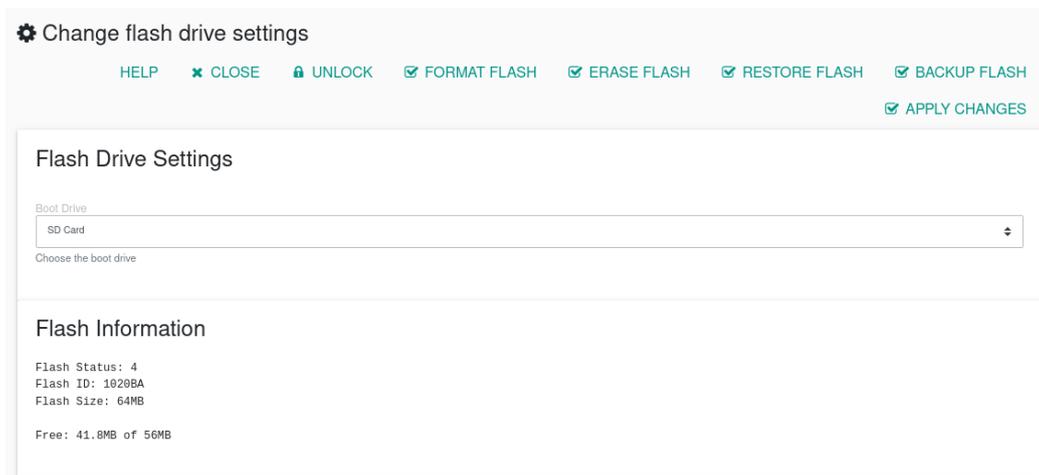
## FLASH MEMORY

Certain hardware versions (WattmonMEGA2, MEGA2-4G, ULTRA) come with 64MB of external Flash memory that can be used for the WattmonOS file system.

Older hardware version of the MEGA2 and MEGA2-4G do not contain the flash chip. To see if your Wattmon has it, open the *Flash Settings* page in *Control Panel*.

Under the *Flash Information* area you will see information about the flash chip. The flash ID and size will show if the flash chip is detected and otherwise you will see a 0 there.

You can choose to boot your Wattmon from *SD card* or from *Internal Flash* by choosing the *Boot drive*. By booting from flash you will ensure that your filesystem and config settings will not get corrupt even if the SD card fails. The SD card is always used to log data and store temporary config files.



The command buttons at the top act as follows:

Click *Erase Flash* to wipe the entire flash chip and reset the internal values to their defaults - this process takes about 2 minutes.

*Format Flash* - This should be run on a blank (or erased) flash, to create the FAT filesystem. This needs to be done before any files can be copied across.

*Backup Flash* - this will create an ISO image of the flash chip as a backup.

*Restore Flash* - This will restore a previously saved ISO image from the root of the SD card (this was saved as `flash.iso`)

*Apply Changes* - this will save the changed boot drive setting.

*Unlock* - This will unlock the flash chip. The flash drive is locked (read only) by default to ensure no changes get written to it. You can also unlock the drive by clicking *Admin > Unlock Flash*.

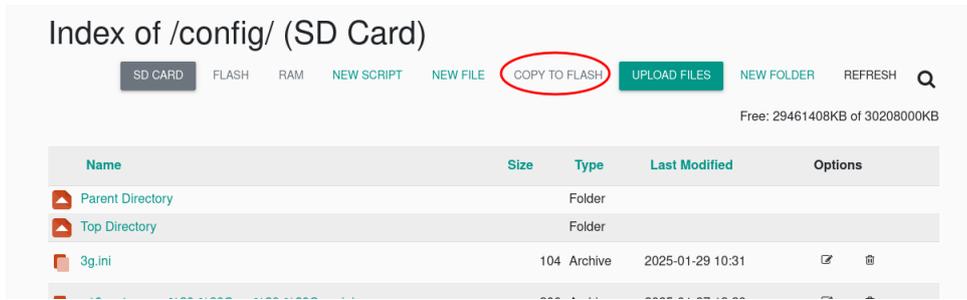
## Preparing your Flash Drive

If you have had to wipe your flash for any reason or are starting on a device that does not have the OS installed on the flash, follow these steps (after unlocking your flash from *Admin > Unlock Flash*)

1. Erase your flash
2. Format your flash
3. Open <https://www.wattmon.com/resources/downloads/category/2-package> and download the WattmonOS Full latest version
4. Copy the file to the root of the SD card (you can do this by copying the file directory using your PC, or by uploading it using the File Manager)



5. Click the *FL* icon next to the file to extract the contents of the gzip file to the flash and wait for it to complete. This may take some minutes.
6. In the *File Manager*, click on the `config` folder



7. Click the *Copy To Flash* button to copy the entire existing config to Flash. This is required because the complete WattmonOS gzip file does not contain the config settings, in order to ensure that you don't lose your existing setup when you update your OS.
8. Go to *Control Panel > Flash Memory*
9. Set the *Boot Drive* to Internal Flash, save & reboot.

## EMAIL SETTINGS

The *Email Settings* page lets you configure your SMTP server. This can be used by the alerts package, the uPHP scripting language and the *Actions* feature to generate emails that are sent out.

The *Sender username or email* needs to be entered with a valid SMTP email address.

The *SMTP Server* should contain the SMTP server IP address or name.

In the *SMTP Username*, enter a valid username - for gmail this usually is your email address.

Enter the *Password* for your connection.

If using encryption, select *Enabled* for *Use TLS/SSL Encryption*.

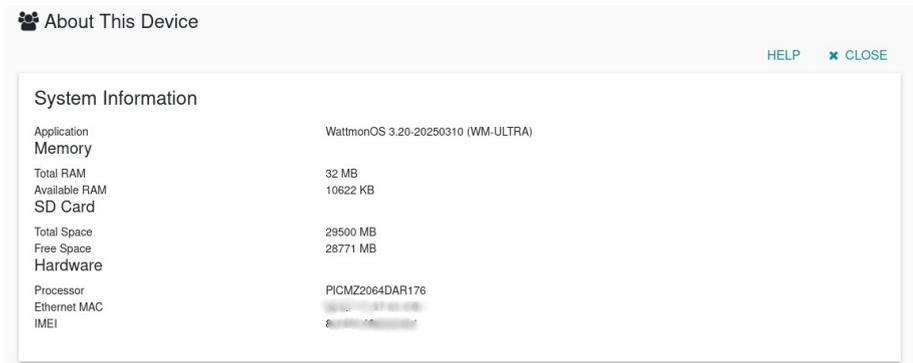
In the *Server Port* field select the TLS port number (587 or 465) if using encryption or 25 for normal plain text (although it's not recommended anymore and rarely available)

Click *Apply Changes* to save the settings.

Finally, click *Send Test Email* to see if it all works.

# ABOUT

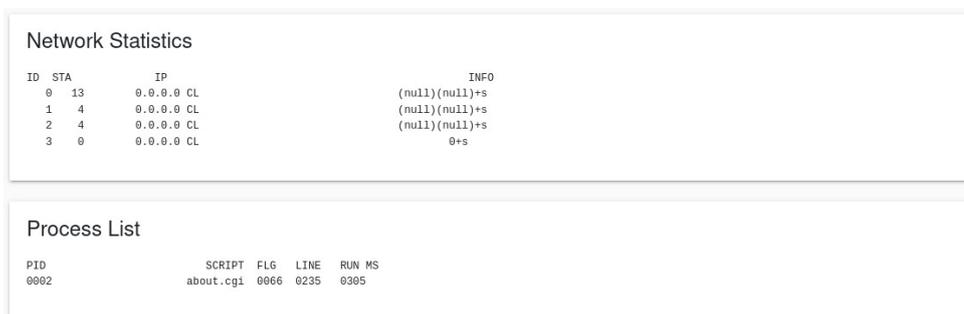
The *About* page of Wattmon provides useful information about the underlying hardware and filesystem, as well as active network connections and running scripts. Access it from *Control Panel > About*.



The top card shows the OS version, followed by available and total RAM. The SD card space used and available are also shown. The microcontroller version, MAC address and IMEI number appear below this.



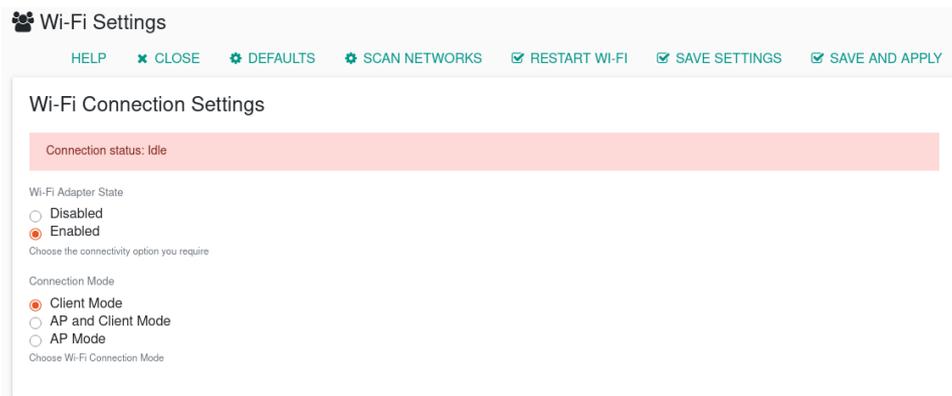
The second section shows the firmware and configuration values currently loaded. This indicates the maximum values. For example the above shows that Wattmon supports a maximum of 30 Modbus TCP channel, and supports device drivers with 150 roles.



The final two sections show network status and running processes. The Process list gives an indication of running scripts and their run time, which can help debug potential issues when developing your own code.

## WI-FI SETTINGS

The WattmonULTRA supports Wi-Fi and this section only pertains only to that model. To open the Wi-Fi settings, go to *Control Panel > Wi-Fi Settings*.



The top card shows the connection status and has options to set the Wi-Fi mode.

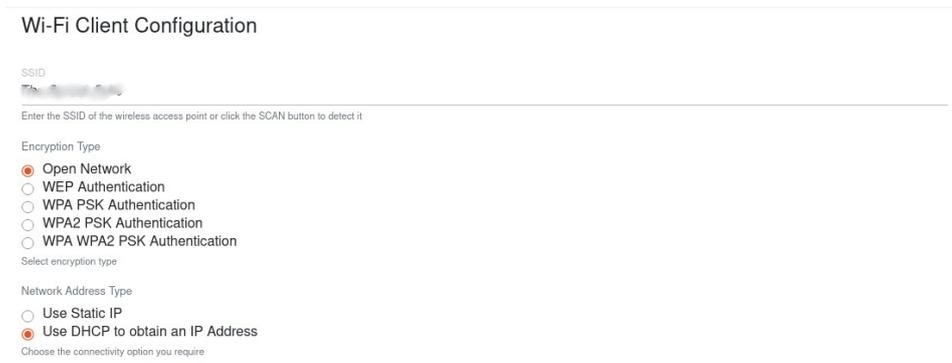
The Wi-Fi Adapter state should be set to Enabled to activate the Wi-Fi hardware.

Choose the type of connection using the Connection Mode options. This can be:

- Client Mode – connect to an access point
- AP and Client Mode – act as an Access Point and connect to another access point simultaneously
- AP Mode – Act as an access point only

### Wi-Fi Client Configuration

This card only appears when client mode or AP and Client Mode are selected, and lets you fill in the SSID and choose the Encryption Type. If a password is required you will have a



field to enter it.

You can choose between Static IP and DHCP in the Network Address type.

### W-Fi Access Point Configuration

If your device is to act as an Access Point, enter the SSID (as it appears to others), the channel and the Encryption type. Enter a password if you are using an encrypted method.

Enter the IP address if the access point (the Wattmon), and choose whether DHCP is enabled or not. With DHCP your clients will automatically be assigned an IP address just after the main IP (i.e 192.168.4.2 if your AP is 192.168.4.1)

**Wi-Fi Access Point Configuration**

SSID  
wattmon

Enter the SSID of the wireless access point or click the SCAN button to detect it

CHANNEL  
Channel 1

Choose the language for the Wattmon interface

Encryption Type

- Open Network
- WEP Authentication
- WPA PSK Authentication
- WPA2 PSK Authentication
- WPA WPA2 PSK Authentication

Select encryption type

IP Address  
192 . 168 . 4 . 1

Enter an IP address for the Wattmon

DHCP Server

- Disabled
- Enabled

At the bottom of the page connection details are show:

Wi-Fi MAC Address

BC-DD- Copy to clipboard

IP Address

0.0.0.0

Gateway

0.0.0.0

When the Wattmon is connected to a gateway, a valid IP address will show.

## NETWORK SETUP

These settings allow you to access your Wattmon device over a particular IP address on your network. It is split into two parts: *Network Configuration* and *Proxy Server Settings*.

The default setup uses a *Static IP* configuration, which can be changed to *Dynamic IP (DHCP)* if needed. Your Wattmon comes pre-configured to work on the *IP Address 192.168.0.55*, with the *Netmask* set to *255.255.255.0* and the *Gateway* to *192.168.0.1*. The *DNS Server* is set to *8.8.8.8*.

The last option in the *Network Configuration* portion is the *MAC Address* (media access control address) of your Wattmon device, which is a 12 digit alpha-numeric unique identifier assigned to network interfaces for communications at the data link layer of a network segment. The unique MAC address for the Wattmon is used as the device key when accessing it from the Internet using port forwarding or through the Wattmon Proxy. You will use this in the **Data Export** settings page.

The second part of *Network Settings* is *Proxy Server Settings*. These settings enable remote access to the Wattmon and should not be changed from the default. There are three options, *Proxy Server Status*, *Proxy Server*, and *Proxy Server Port*, which are set to *Enabled*, *remote.wattmon.com* and *7000*, respectively, as displayed in the image above.

Home / Control Panel / Network Settings

## Network Settings

HELP ✕ CLOSE ⚙️ DEFAULTS 📄 APPLY CHANGES

### Network Configuration

Network Address Type

Use Static IP  
 Use DHCP to obtain an IP Address

Choose the connectivity option you require

IP Address  
192 . 168 . 0 . 55

Enter an IP address for the Wattmon

Enter Network Mask  
255 . 255 . 255 . 0

Usually this will be 255.255.255.0 but adjust according to your local network requirements.

Gateway  
192 . 168 . 0 . 253

Enter gateway or address of your DSL router

DNS Server  
8 . 8 . 8 . 8

Enter DNS Server IP address to lookup host names

MAC Address  
- - - - - Copy to clipboard

### Proxy Server Settings

Proxy Server

Disabled  
 Enabled

Enable this to be able to reach your device via the Wattmon remote proxy server

Proxy Server  
remote.wattmon.com

Enter remote proxy server or ip. If using wattmon, leave this at remote.wattmon.com

Remote Proxy Server Port  
7000

When using the Wattmon proxy, set this to 7000

To configure your Wattmon to work on an IP address of your choice, enter the *Netmask*, *Gateway*, and *DNS* settings by referring to your on-site modem, and assign the Wattmon a unique static *IP Address*.

Save the settings by selecting *Apply Changes* and wait for a confirmation message. You will need to reboot the Wattmon to see your changes in effect.

*Please note: Once the Wattmon reboots, it will not log in via the default IP if it has been changed; you will need to update the IP address in the address bar of your browser.*

To reset the IP address of your Wattmon to its default, select the *Defaults* option, and then *Apply Changes*.

## MODBUS COMMUNICATION SETTINGS

To modify the Modbus settings on your Wattmon to match those of the devices you wish to interface with, you need to configure the Modbus Communication Settings. You can access these settings through the *Manage Devices* page as described earlier in the manual, or you can reach the *Modbus Communication* page through the *Control Panel*.

The *Modbus Communication Settings* are split into different sections – *Modbus RTU (RS-485) Settings*, *Modbus Channel 2 Settings (MEGA2,MEGA2-4G, ULTRA)*, and *Modbus TCP Settings*, which are all the communication channels offered on your Wattmon.

The first section, *General Settings*, is for enabling the Modbus engine on the Wattmon, which allows for communication with devices over RS-485 Modbus RTU. If the *Modbus Engine* is set to *Enabled*, Modbus devices on the RS-485 port will be polled automatically, at a particular polling interval. If the *Modbus Engine* is *Disabled*, the RS-485 port can be used for polling devices over proprietary protocols through custom scripts or packages. If you are interfacing the Wattmon with Modbus devices, set it to *Enabled*.

## Modbus RTU (RS-485) Settings

If the *Modbus Engine* is set to *Enabled*, a section called *Modbus RTU (RS-485) Settings* will be visible. Here you can set the *Baud Rate*, *Parity*, *Inter-packet delay*, *Time-out*, *Retry Interval* and *Maximum retries*.

Home / Control Panel / Modbus Devices / Modbus Communication Settings

### Modbus Communication Settings

HELP ✕ CLOSE ✓ SAVE CHANGES

#### General Settings

Modbus Engine

Enabled - RS-485 (and Modbus TCP if available) will be polled automatically  
 Disabled - Use the RS-485 port for proprietary protocols through scripts  
If you have external devices connected using Modbus, make sure this is enabled. For proprietary protocols you may choose to disable this.

#### Modbus RTU (RS-485) Settings

Baud rate  
9600  
Choose the baud rate that matches the devices you wish to connect to

Parity  
 No Parity  
 Even Parity  
 Odd Parity  
Select the parity matching your device settings

Inter-packet delay  
10  
The interval between modbus packet transmission in milliseconds

Time-out  
300  
The time in milliseconds to wait for a packet response

Retry Interval  
1000  
The time in milliseconds after which a failed packet is resent

Maximum retries  
5  
Number of times a packet is resent before erroring out. Set this to 0 to not retry.

The *Baud Rate* specifies the speed at which information is communicated over a data channel. The default on the Wattmon is set to 9600.

The *Parity* bit is an optional error checking bit that some devices require. Select between *Even Parity* and *Odd Parity*, or use the default *Parity* setting *No Parity*.

*Inter-packet delay* sets the interval in milliseconds between the Modbus packets of data transmission. The default is set to 10. For longer cable runs and older devices, you may need to increase this to 20 or 30 ms.

*Time-out* sets the time in milliseconds the Wattmon waits for a Modbus packet response. If a Modbus device does not respond within the stipulated time, that packet is timed-out. The default is set to 300.

The *Retry Interval* sets the time in milliseconds after which a failed or timed-out Modbus packet is resent. The default is set to 1000.

*Maximum retries* sets the number of times a failed or timed-out Modbus packet is resent before being cancelled. If you wish to not retry any failed packet, set this to 0. The default is set to 5.

## Modbus Channel 2 Settings

Channel 2 is available on WattmonMEGA2, MEGA2-4G and ULTRA. You can enable or disable it, and an option *RTU Bridge* option lets you pass packets through between channel 2 and channel 1.

Like the previous section, if set to *Enabled*, a new section will be visible, where you can set the *Baud Rate*, *Parity*, *Inter-packet delay*, *Time-out*, *Retry Interval* and *Maximum retries* for devices communicating over channel 2.

*Bridge Mode* allows you use more than one Modbus master where the Wattmon acts as a transparent bridge between the second master and the modbus devices on channel 1.

### Modbus Channel 2

Modbus Engine

Enabled - RS-485 (and Modbus TCP if available) will be polled automatically  
 Disabled - Use the RS-485 port for proprietary protocols through scripts  
 RTU Bridge

If you have external devices connected using Modbus, make sure this is enabled. For proprietary protocols you may choose to disable this.

### Modbus RTU (RS-485) Channel 2 Settings

Baud rate  
9600

Choose the baud rate that matches the devices you wish to connect to

Parity

No Parity  
 Even Parity  
 Odd Parity

Select the parity matching your device settings

Inter-packet delay  
10

The interval between modbus packet transmission in milliseconds

Time-out  
1000

The time in milliseconds to wait for a packet response

Retry Interval  
15

The time in milliseconds after which a failed packet is resent

Maximum retries  
1

Number of times a packet is resent before erroring out. Set this to 0 to not retry.

## Modbus TCP Settings

The last section is *Modbus TCP Settings*, which is for enabling the TCP communication engine for interfacing with devices over Modbus TCP. The most basic difference between Modbus TCP and Modbus RTU is that Modbus TCP runs on an Ethernet physical layer (TCP/IP), whereas Modbus RTU is a serial level protocol.

### Modbus TCP Settings

Modbus TCP Engine

Enabled  
 Disabled

If you have Modbus TCP devices, enable the engine and configure the virtual channels so that you can configure them using the device manager.

Inter-packet delay  
100

The interval between modbus packet transmission in milliseconds

Time-out  
300

The time in milliseconds to wait for a packet response

Retry Interval  
1000

The time in milliseconds after which a failed packet is resent

Maximum retries  
1

Number of times a packet is resent before erroring out. Set this to 0 to not retry.

Auto Close

Close after every modbus operation  
 Re-use the same connection

Certain Modbus TCP devices that have limited sockets require the connection to be closed in order to not freeze after some time. If possible, re-use the same connection to increase performance.

If you have devices which need to be interfaced over Modbus TCP, set the *Modbus TCP Engine* to *Enabled*. Once *Enabled*, you can configure the *Inter-packet delay*, *Time-out*, *Retry Interval* and *Maximum retries* settings as well.

The *Inter-packet delay* is the interval in milliseconds between the Modbus packets of data transmission. The default is set to *100* for Modbus TCP.

*Time-out* sets the time in milliseconds the Wattmon waits for a Modbus packet response. If a Modbus TCP device does not respond within the stipulated time, that packet is timed-out. The default is set to *300*, much like for Modbus RTU devices.

The *Retry Interval* sets the time in milliseconds after which a failed or timed-out Modbus packet is resent. The default is set to *1000*.

*Maximum retries* sets the number of times a failed or timed-out Modbus packet is resent before being cancelled. If you wish to not retry any failed packet, set this to *0*. The default is set to *1*.

*Auto Close* is a setting for certain Modbus TCP devices that have limited sockets and require the connection to be closed in order to not freeze data transmission. There are two options provided here – *Close after every Modbus operation* and *Re-use the same connection*. If possible, re-use the same connection to increase performance.

There are 10 TCP channels available on the WattmonMEGA, MINI3 and MINI3-4G. The MEGA2, MEGA2-4G and ULTRA have 30 channels.

The *Modbus TCP Channels* are shown by name, along with two options each – *Modbus TCP Device IP Address* and *TCP Port*.

In the *Modbus TCP Device IP Address* option, enter the IP Address of the Modbus TCP device you wish to interface with. If there is no device being interfaced on the channel, leave the IP address as *0.0.0.0*.

The *TCP Port* is the port number for the Modbus TCP connection. The default port for most connections is *502*, which is the default on the Wattmon, but can be changed if it does not match that of your inverter.

As shown in the image below, if just *Modbus TCP Channel 1* is used for a device with a TCP/IP address of *192.168.0.10*, the IP address of that device is entered in the *Modbus TCP Device IP Address* option. The TCP Port is left at the default of *502*, since that is the most commonly used port number in Modbus TCP. The consecutive channels, if any, which are not being used, are left at the default IP address of *0.0.0.0*, with the port number also left at the default *502*, as seen here for *Modbus TCP Channel 2*.

#### Modbus TCP Channel 1

TCP Server (Listening)

Select whether this channel is a TCP Client or Modbus TCP Server

Modbus TCP Device IP Address

192.168.0.7

Enter the IP Address of the Modbus TCP device or 0.0.0.0 if not used

TCP Port

502

Enter a port number for the connection. The default Modbus TCP port is 502

The *TCP Server* check box can be enabled to allow other devices to connect to Wattmon to read information over Modbus TCP. Select the port number to listen on. You can add multiple channels on the same port number to allow multiple remote connections.

## Modbus TCP Channel 1

TCP Server (Listening)

Select whether this channel is a TCP Client or Modbus TCP Server

[VIEW REGISTER MAP](#)

TCP Port

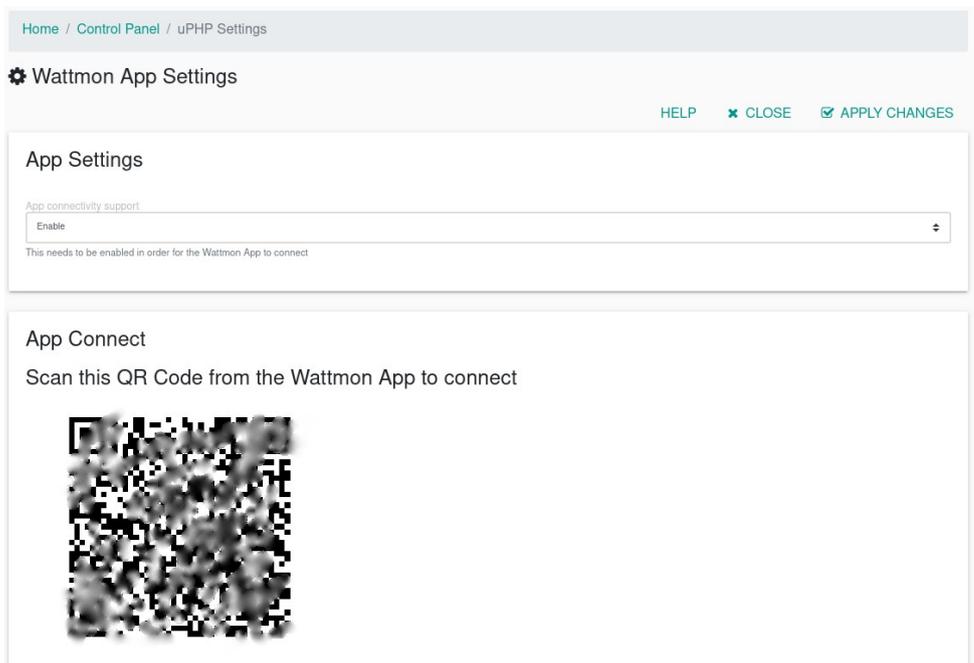
502

Enter a port number for the connection. The default Modbus TCP port is 502

The *View Register Map* button brings up the current register mapping that can be used to access the device data. This map is dynamically generated based on the number of devices and data points being logger.

## WATTMON APP

It is possible to connect an android app (the Wattmon App) to your Wattmon. To do this open the *Wattmon App* page (this is only available in OS 3.20 and above) under Control Panel.



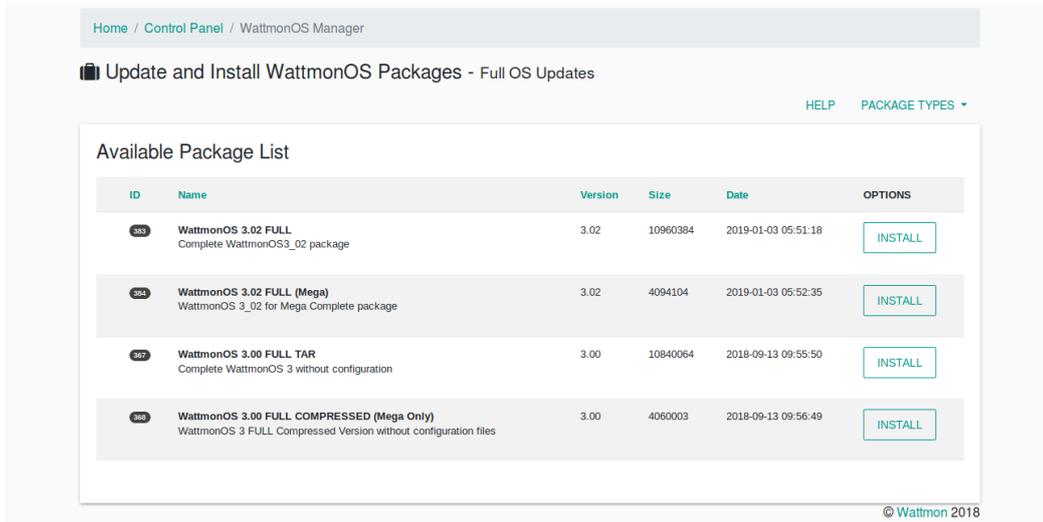
Select *Enable* in the app connectivity support, *Apply Changes* and reboot. Return to the same page, and then scan the QR code from your app to add it to the phone.

## WATTMONOS UPDATER

In order to update the WattmonOS, head to *WattmonOS Updater* in the *Control Panel*, consists of updates, both incremental and complete, to the Wattmon OS. Click *Install* for the WattmonOS update to download. This may take a while, so we recommend not to use the device during that time. Updating the OS install will not override the configuration settings of your Wattmon, unless otherwise specified. But we recommend that you do back up your *Configuration Files*, as detailed in **Backup Data**.

### Incremental Updates

The *Update and Install WattmonOS Packages* setting defaults to *Incremental OS Updates* once you open it. Incremental updates are to update from consecutive OS versions, such as from 3.19 to 3.20, or small additions to the current version of your OS. *Incremental OS Updates* regularly feature minor upgrades or fixes to the Wattmon operating system, such as the improvement of the languages supported, or added options to an existing setting. Due to the nature of an incremental update, they are particularly small in size, usually up to 1 MB, allowing them to be easily downloaded even in an area with poor network connectivity.



The *Available Package List* showcases the details of all the available updates, which are classified by their *ID*, *Name*, *Version*, *Size*, *Date* and *Options*. *ID* denotes the sequence of the update. *Name* details the update along with a short description of the changes therein. *Version* displays the OS version that the update is designated for. *Size* details the size of the update in bytes. *Date* shows the date and time that the public build of the update was made available for downloading. *Options* offers a single option, to *Install* the update.

## Full OS Updates

When updating from non-consecutive Oses, or if you wish to reinstall the existing OS, select the *PACKAGE TYPES* drop-down, and select the second option to go to *Full OS Updates*. In order to return to the incremental update packages, click on *Incremental OS Updates* from the package types drop-down from the *Full OS Updates* screen.

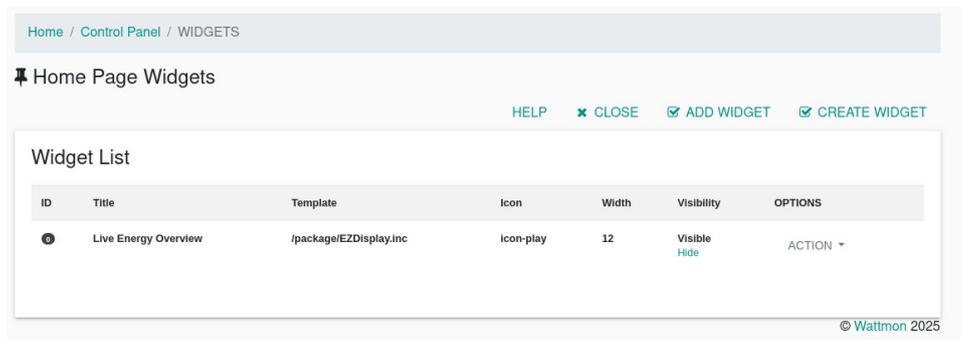
The *Full OS Updates* displays options that will be the full OS, which can be downloaded to either update from a much older OS version, or when reinstalling an OS. The *Available Package List* is laid out like on the *Incremental OS Updates* page. You will notice that the update size is consistently larger, exceeding 10 MBs in some cases. Some of the updates are specifically for the WattmonMEGA or WattmonMINI, in which case the *Version* will be the same, but the name will state the difference; make sure you download the correct complete OS for your device.

## Additional Packages

Additional packages contain updates and individual packages such as PC, FTP, MQTT and more. These can be installed at any time and updated at any time. Every time you need to install a package that is not part of the default OS, or need to update to the latest version you can do it through this page.

## WIDGETS

The Widgets page can be accessed from *Control Panel*. This shows the current widgets on the dashboard. It allows you to edit the positions and order of the widgets, and add custom widgets if required. The packages usually take care of adding or removing their corresponding widgets automatically. For example *EZDisplay* adds the widget when you install it, and removes it when you uninstall it.



## Adding a Widget

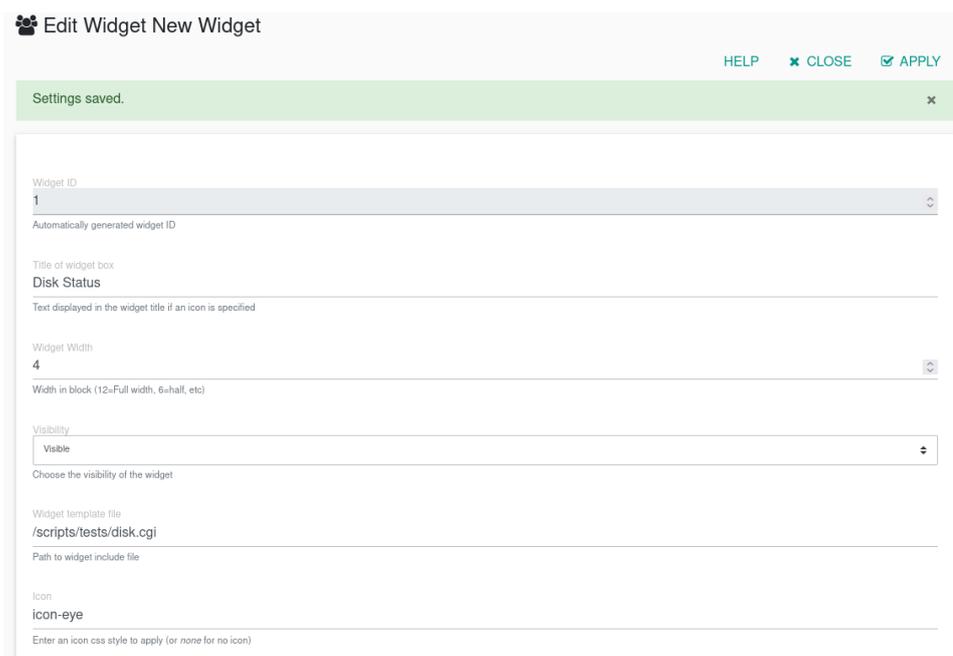
If you have an include file that you wish to use in the dashboard, you can click the *Add Widget* button.

Enter a title for the widget in the *Title of widget box* field, and select a *Width*. The *Width* can be between 1 and 12, where 12 is the maximum width – this is due to the underlying bootstrap framework that Wattmon uses.

Choose to show the widget or hide it using the *Invisibility* drop down.

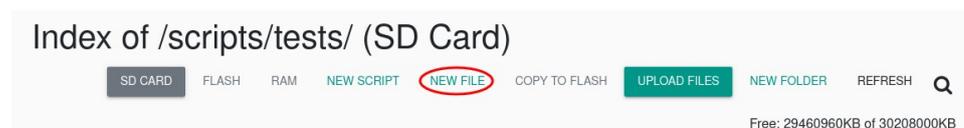
Enter the complete path to the widget include file in the *Widget template file* field.

As an example, let's create a widget called *Disk Status* that shows the status of a mounted drive. Enter: `/scripts/tests/disk.cgi`.

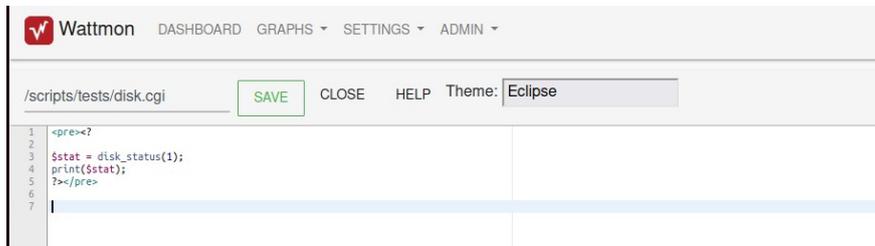


Click *Apply* to save the settings.

Next, navigate to the *File Manager* and go into the `scripts/tests` folder. Click the *New File* button.



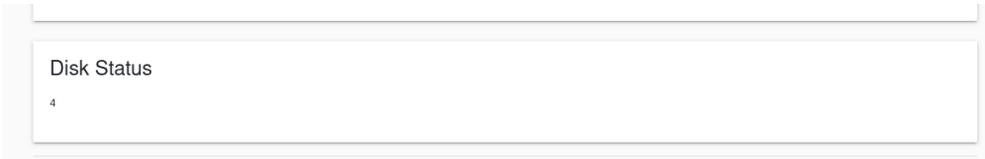
Change the file name to `/scripts/tests/disk.cgi` and past in the code as below. Save the changes.



The screenshot shows the Wattmon web interface. At the top, there is a navigation bar with 'Wattmon', 'DASHBOARD', 'GRAPHS', 'SETTINGS', and 'ADMIN'. Below this is a header area with the path '/scripts/tests/disk.cgi', a 'SAVE' button, 'CLOSE', 'HELP', and a 'Theme: Eclipse' dropdown. The main content area is a terminal window with a light blue background and a white text area containing the following code:

```
1 <pre><?
2
3 $stat = dsk_status(1);
4 print($stat);
5 ?></pre>
6
7
```

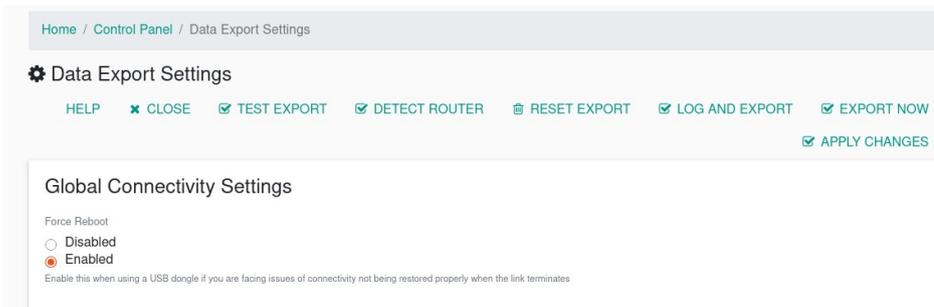
Finally navigate back to the dashboard and you should see your widget:



## DATA EXPORT

The *Data Export Settings* is split into six parts, and allow you to configure *Global Connectivity Settings*, *Export to Wattmon Cloud*, *Custom Server Data Export*, *MQTT Export*, *PV Output Export*, and *Reachability settings*.

The control bar features several buttons, two of which are *Reset Export* and *Export Now*. The *Reset Export* option aborts the exporting of backlogged data, clearing data log issues. *Export Now* runs the data export script, forcing the upload of data.



The *Wattmon Reachability Settings - Only with Port Forwarding*, allows you to reach your device without the need for a static IP address or a DynDNS entry. Once the *Wattmon Reachability Status* is *Enabled*, with the *External Port* and *Server Update Interval* set correctly (default - 8080 & 5), you will need to *Apply Changes* and *Reboot*. After this, you will be able to reach your device from the following link - <http://www.wattmon.com/live?key=A1-B2-C3-D4-E5-F6>, where *A1-B2-C3-D4-E5-F6* will be the MAC address of your Wattmon device.

The *Global Connectivity Settings* is an option to automatically reboot a connected USB dongle in case issues of connectivity are faced, forcing a fresh network connection. Set *Force Reboot* to *Enabled* to activate this option.

### Export To Cloud

Export To Cloud

Disabled  
 **Enabled**

Enable this to sync the data on your Wattmon with a cloud server

Select a server

ems.wattmon.com

Choose the Wattmon server to export to. Use ems.wattmon.com if you are unsure.

Use Encryption (HTTPS)

**Disabled**  
 Enabled

This will speed up the export process for devices with a large amount of data points

Export Interval

5

The interval in minutes at which Wattmon pushes data from the local CSV file to the server

Use MD5

Disabled  
 **Enabled**

Add an MD5 Checksum to each packet to ensure correct delivery to the cloud

Number of Lines

5

Number of lines to export at a time

Use Static Header

Disabled  
 **Enabled**

This will speed up the export process for devices with a large amount of data points

Export Alerts

**Disabled**  
 Enabled

Export alerts to the cloud (for SMS and Email notification)

Export Only (Experimental)

**Disabled**  
 Enabled

Enabled this to not log CSV data locally unless connectivity issues occur

Buffer Re-Export Interval

0

Number of seconds to relaunch the script to send data during backlogged buffering. This may adversely impact performance. Set to 0 to disable.

---

**Current Status**

Last exported on Thu Mar 27 15:26:15 2025.  
Position in CSV File: 26054  
CSV File Date: 27/03/2025

To set up *Export to Wattmon EMS*, first set *Export To Cloud* to *Enabled*. The next option, *Select a Server*, lets you select the server you have been allotted, which is most likely to be Wattmon EMS, *ems.wattmon.com*.

If exporting via HTTPS, select *Use Encryption*.

The *Export Interval* is set by default to 5, which is the interval in minutes at which the Wattmon pushes data from the local CSV file to the server. You may increase or reduce the interval if you wish to have the data pushed at less frequent or more frequent intervals.

*Use MD5* can be set to *Enabled* to ensure correct delivery of the data by adding a MD5 checksum to each packet. For MINI3/MINI3-4G with limited memory you may need to disable this options when you have over 300 data points.

*Number of Lines* sets the number of data lines that are exported at a time. It is set by default to 15 but may be reduced if you experience data export issues when pushing larger amounts of data.

*Use Static Header* will save memory by generating a header and saving it in the */config/* folder as *csv\_headers.txt*. This will mean that Wattmon does not need to regenerate the header dynamically on every run.

If you require Email and SMS notifications through the EMS server, set *Export Alerts* to *Enabled*.

The *Export Only* option allows you to toggle between storing all data locally in CSV files if set to *Disabled*, or only storing CSV data when exporting fails when set to *Enabled*. The latter does not save the logged data on the Wattmon's micro SD card, an option that should only be selected if specified. If you are using the *Custom Server Data Export Settings*, you will need to set this option to *Disabled*, as the CSV data has to be stored locally.

The *Buffer Re-Export* interval is an optional feature that let your script re-run every few seconds when dealing with a lot of backlogged data. Enter 0 to disable it. This is not recommended for lower end models such as MEGA, MINI3, MINI3-4G.

The *Current Status* of the data export is always displayed at the bottom of the section, detailing the date and time of the last exported data, the position of the data in the stored CSV file, and the date of the CSV file from which the data was exported.

## Custom Server Data Export

To export the logged data to a custom server, go to the section called *Custom Server Data Export Settings*, and follow the same instructions as shown for *Export to Wattmon Cloud*. You will first need to enter a Custom Server URL for where the Wattmon must upload the data to in a CSV file. To learn more on the data export file format, visit the **Exporting Data to a Custom Server** page on our website.

The *Export Interval*, in minutes, can be set to 5 and *Number of Lines* per export to 15, the same as when exporting to the Wattmon Cloud. A *CSV Header Row* can be included or excluded from the data log file by selecting *Include Header* or *Exclude Header*, for ease on server-side processing.

Custom Server Data Export Settings

Custom Export Status

Disabled  
 Enabled

Enable this to push data to your custom server

Custom Server URL  
https://my-server.com/my-end-point/

URL to your script (i.e. http://www.test.com/my\_log.php). Data will be sent via a POST request in CSV format. For more information on the file format refer to the website.

Export Interval  
5

Interval in minutes to push data to your custom server

Number of Lines  
5

Number of lines to export at a time

CSV Header Row

Include Header  
 Exclude Header

You can choose to include row headers in the CSV for server side processing if required. These will be matching the definitions in the Data Collection page

Debug

Disabled  
 Enabled

Add debug information into /logs/debug.txt to troubleshoot issues

Current Status

Last exported on Thu Jan 1 00:00:00 1970.  
Position in CSV File: 0  
CSV File Date:

The *Current Status* of the data export is always displayed at the bottom of the section, detailing the date and time of the last exported data, the position of the data in the stored CSV file, and the date of the CSV file from which the data was exported.

An additional *Debug* option can be enabled to troubleshoot connectivity issues but should be removed once everything is working.

## MQTT Export

MQTT Export, introduced in OS 3.20 allows you to push data over MQTT to your server. Data is turned into a JSON packet in the format as shown below.

```
{"mac": "XX-XX-XX-XX-XX-XX", "data":  
[{"ts": "1743090481", "inverter1_AC_Power": 8, ...}]}
```

Select the *MQTT Export Interval* in minutes to push data at. Set the *Number of Lines* to 5 to push more than one historical line at a time. Enter the *Topic* to push to, and finally select the *MQTT Server number* - on the latest firmware 2 channels are supported and need to be configure in the MQTT2 package (which can be download from the WattmonOS Updater).

**MQTT Export**

Export data to via MQTT

Disabled

Enabled

Export data to via MQTT

MQTT Export Interval

1

Interval in minutes to push data to your custom server

Number of Lines

5

Number of lines to export at a time

Topic

/iot\_data/push

Enter MQTT Topic Name

MQTT Server

Server 1

Server 2

Export data to via MQTT Server

## Export to PV Output

If you have a PV Output key you can optionally push data to the server

## Wattmon Reachability Settings

This option lets you reach your Wattmon remotely if you have configure port forwarding. It pushes a message on a regular basis to `wattmon.com/live` which you can use to see if your device is online. Enter the *External port*, and *Server interval* in minutes to configure it.

## PACKAGE MANAGER

The *Update and Manage Packages* setting consists of configurations for various add in packages, such as PC (Power control), FTP, MQTT and more optional features of the WattmonOS, that need to be installed to be used.

Upon first opening, the *Available Package List* will display no items. Click on *RESCAN FOLDER* to reveal all the optional packages available on your device. The available packages are displayed, in greyscale, with details such as *ID*, *Name*, *Author*, *Version*, *Status* and *Options*.

**Update and Manage Packages**

[HELP](#) [RESCAN FOLDER](#)

**Available Package List**

ID	Name	Author	Version	STATUS	OPTIONS
1	<b>EZDisplay</b> EZDisplay Visualisation <i>Installed manually</i>	Akash Heimlich	20230926	Enabled	ACTION ▾
2	<b>MQTT2</b> MQTT 2 Configuration <i>Installed manually</i>	Akash Heimlich	20241010	Enabled	ACTION ▾
3	<b>Alerts</b> Alert package <i>Installed manually</i>	Akash Heimlich	20250203	Enabled	ACTION ▾
4	<b>VariableWidget</b> Variable widget for the front page <i>Installed manually</i>	Akash Heimlich	20240808	Uninstalled	ACTION ▾
5	<b>SavedCounters</b> Pulse Counting Inputs Package <i>Installed manually</i>	Akash Heimlich	20231124	Uninstalled	ACTION ▾
6	<b>Devices</b> Devices Widget	Akash Heimlich	1.0	Uninstalled	ACTION ▾

*ID* defines the display name of the package, whereas *Name* shows the complete name of the package.

The *Author* column indicates who wrote a particular package for the Wattmon.

*Version* states the software build number of a package.

*Status* details whether the package is *Uninstalled*, which means it needs to be installed; the status is *Disabled* if the scripts from it have been installed but are temporarily stopped from running; or *Enabled*, if the package is installed and active. The *Options* column features a drop-down called *Action*, which allows you to *Install*, *Uninstall*, *Disable* or *Enable* a package.

To install a particular package, select the *Action* button next to the package name, and choose the *Install* option. Once the package has been successfully installed, the package will not appear in a grey highlight and the status will be changed to *Enabled*. The package will then appear in the control panel as a button after rebooting. You can also select the *Configure* button to bring up the package's configuration page.

If you have recently downloaded a new package through the WattmonOS update's *Additional Packages* page, you will need to click the *Rescan Folder* button to ensure that it shows up.

## DATA COLLECTION

The *Data Collection* page allows you to select the variables you wish to log to .csv files on the Wattmon's microSD card, as well as push to the cloud portal. This is normally managed automatically by EZConfig but can be used to make changes to things like the min/max values of a data point if required. Please note that after re-running EZConfig any changes will be automatically overwritten.

The *Data Collection Groups* list shows all the configured log files. The default data group name is *Datalog* or *CSV Log*, depending on your configuration method, and all variables you wish to add or remove for the data log can be found in there. To edit the *Datalog* group, select the *Action* drop-down, and then select *Configure*.

The *Configure Data Collection - Datalog* is split into two categories - *Data Collection Settings* and *Data Points*. *Data Collection Settings* features information such as *Group Name*, *Log File Type*, *Group Status*, *Disk Write Interval*, and *Data Interval in Minutes*. While these are editable by the user, we recommend leaving these at the default values unless specified by us or by advanced users.

The *Group Name* is set by default to *Datalog* or *CSV Log* but can be changed if the user desires. The *Log File Type* drop-down lets you choose if you wish to have the *Log files split by day and month*, or a *Single log file*; the default is set as *Log files split by day and month* given that *Single log file* creates a single large file which is more susceptible to read/write issues. The *Group Status* drop-down allows you to either *Enable*, *Disable* or *Delete* the data collection group - the default is set at *Enabled* and should not be changed, as this affects the actual data push to a server. The *Disk Write Interval* option allows you to choose the time interval, in minutes, in which each log file is written to the microSD card; we recommend that the log interval be set at 5 minutes to allow for the writing of larger log files while not constantly writing data which reduces the life of a microSD card. The time interval can be optimized based on the users' requirement. The *Data Interval in Minutes* options allow you to set the number of minutes between each log entry; the default is set to 1.

To save changes made to any of the options, you need to select *Apply Changes* like on all other settings pages on the Wattmon, followed by a device reboot.

Home / Control Panel / Data Collection / Data Collection

☰ Configure Data Collection - Datalog

HELP ✕ CLOSE ✓ APPLY CHANGES

### Data Collection

Group Name  
Datalog

Enter a meaningful name for this group

Log File Type  
Log files split by day and month

Select a type for the data log

Group Status  
Enabled

Status of this data collecting group

Disk write interval  
5

Data is buffered in memory for this period and written to disk to decrease the wear on the SD card

Data interval in minutes  
1

The number of minutes between each log entry (the default is 1)

### Data Points

+ ADD

ID	Type	Value	Scale	Function	Options
1	System Variable	Solar Watts	* 1	AVERAGE	ACTION ▾
2	System Variable	Grid Charge Watts	* 1	AVERAGE	ACTION ▾
3	System Variable	Grid Load Watts	* 1	AVERAGE	ACTION ▾
4	System Variable	Battery Voltage	* 1	AVERAGE	ACTION ▾
5	System Variable	Battery Cur AH	* 1	AVERAGE	ACTION ▾
6	System Variable	Solar kWh Today	* 1	MAX	ACTION ▾

The *Data Points* section features all the variables that are logged on the Wattmon and written to the on-board microSD card as well as uploaded to a cloud portal if that is enabled; for more details, go to the **Data Export** settings page in this guide. The section displays each data point with the following details - *ID*, *Type*, *Value*, *Scale*, *Function* and *Options*.

To add a new data point, click on the +ADD button in the upper right corner of the section. To edit, stop editing, or delete an existing data point, click on the *Action* drop-down next to the data point of your choice and choose from the three options - *Edit*, *Stop Edit* and *Delete*.

### Data Points

+ ADD

ID	Type	Value	Scale	Function	Options
1	System Variable	Solar kWh Today	1	AVERAGE	ACTION ▾
2	Device Variable	inverter1_AC_Active_Power	1000	INSTANTANEOUS	ACTION ▾
3	Global Variable	custom_variable	1	MAX	ACTION ▾

When adding a new data point, *ID*, which is the numerical order of the variable, is allotted automatically.

The *Type* drop-down presents three options – *System Variable*, which is value calculated by the Wattmon from a role and its associated types; *Device Variable*, which is the actual role value from a connected device; and *Global Variable*, which is a calculated value generated by a custom script running on the Wattmon. For Modbus devices, both third-party and the Wattmon Modules, the logged value type is *Device Variable*.

The *Value* option reveals the names of the values or roles based on your selection of the value *Type*. For *System Variables* you may choose from a drop-down list of calculated values, while for *Device Variables* you may choose from a drop-down list of non-calculated polled values; for a *Global Variable*, you will need to enter the name manually, as shown above.

The *Scale* option allows for the scaling of values as a float value (0.0, 0.00, etc.). For Wattmon generated values this is always 1 – for Modbus devices, this needs to be verified with the proprietary Modbus registers. For example, the *AC\_Active\_Power* value is in watts, but if you wish to log it in kilowatts, the scale would be set to 1000, as can be seen in the image above.

The *Function* drop-down reveals four options – *Average*, *Min*, *Max* and *Instantaneous*, which can be selected to approximate the values according to the user. For most *System Variables*, the *Average* option applies best. The *Instantaneous* option is used for values that represent a specific function, such as alerts or a device status, or for values that are obtained from another device, such as in *Device Variables*.

To save the changes or additions, click on *Apply Changes* on the top right corner of the page. Note that a reboot will be required for the Wattmon to commence logging new values or applying any modifications to the existing values.

## INTERFACE SETTINGS

User Interface settings define how the way Wattmon responds in the browser.

Interface Settings

Optimization

Only essential files from SD Card, optimized for remote access

All files from SD Card

Speed up the loading of pages by choosing to only load essential files from the SD card and take the remainder from the cloud

Desktop Notifications

Do not show notifications

Show messages using desktop Notifications

Show notifications in browser

Dashboard refresh Interval

1 second

The frequency of updates of data on the dashboard in seconds

Maximum Script Execution Time

20 seconds

The frequency of updates of data on the dashboard in seconds

Maximum low Memory Errors

400

Increase this to prevent reboots on slow links. This will increment about 20 times a second

In the *Optimization* group, select *Only essential files from SD Card, optimized for remote access* to ensure remote access works smoothly. If you are working locally only you can set it to *All files from SD Card*.

Some browsers allow notifications which can be enabled using the *Desktop notifications* option.

Set the *Dashboard refresh interval* to a larger value if you do not need the page to refresh every 1 second.

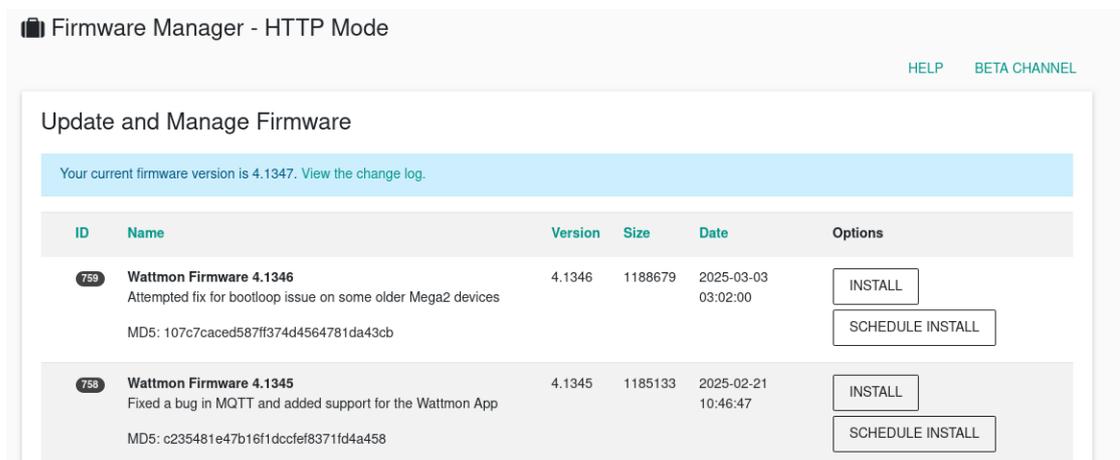
The *Maximum script execution time* is 20 seconds by default. This option can be changed here or in the uPHP settings page. For low memory devices you can increase this if required to ensure that long running scripts do not get prematurely terminated.

The *Maximum low memory errors* will affect the lower end devices, this is the maximum internal counter value that once reached will cause the device to reboot with a log entry Low Memory. This could be increased but if you see your device rebooting frequently due to this the best approach would be to troubleshoot the root cause, most likely too many data points being logged. Try reducing the number of data points and see if the reboot errors stop.

## FIRMWARE UPDATER

The *Firmware Manager* allows you to update your Wattmon to the latest firmware, making sure you are up to date with the latest features and bug-fixes.

The *Update and Manage Firmware* list displays several firmware versions on the page, along with details about the changes contained in them, and their dates of compiling. A blue overlay details the current firmware version of your device.



The screenshot shows the 'Firmware Manager - HTTP Mode' interface. At the top right, there are links for 'HELP' and 'BETA CHANNEL'. The main heading is 'Update and Manage Firmware'. Below this, a blue banner states: 'Your current firmware version is 4.1347. View the change log.' Below the banner is a table with the following columns: ID, Name, Version, Size, Date, and Options. The table contains two entries:

ID	Name	Version	Size	Date	Options
759	<b>Wattmon Firmware 4.1346</b> Attempted fix for bootloop issue on some older Mega2 devices MD5: 107c7caced587ff374d4564781da43cb	4.1346	1188679	2025-03-03 03:02:00	INSTALL SCHEDULE INSTALL
758	<b>Wattmon Firmware 4.1345</b> Fixed a bug in MQTT and added support for the Wattmon App MD5: c235481e47b16f1dccfef8371fd4a458	4.1345	1185133	2025-02-21 10:46:47	INSTALL SCHEDULE INSTALL

Depending on the underlying hardware chipset a different firmware major build number would apply. For older legacy devices 1.x would indicate the PRO or MINI. The 2.x range indicates a MEGA, MINI3 or MINI3-4G. The 3.x indicates the M2. The 4.x range is used for MEGA2, MEGA2-4G and ULTRA.

In order to prevent incorrect updating a different naming convention is used depending on the hardware. For legacy 1.x devices, the firmware is called `image.hex`. For the 2.x range it is `imagez.hex` and for the 4.x range it's `imageu.hex`. This is important to know when manually updating a firmware file on the SD card to force flash.

In order to install a firmware, whether to upgrade or downgrade the current version, first verify if you have an active internet connection. The Wattmon's firmware can be installed either from the Wattmon server, or locally, depending on the network settings of your Wattmon or that of the computer you are using to connect to the Wattmon,

If either the Wattmon or your computer has an active Internet connection, to install a firmware, proceed to the *Update and Manage Firmware* page and click on *Install* next to the firmware version you wish to install. We recommend that you update the firmware to the latest one unless you have been instructed otherwise by us. After you select *Install*, you will be prompted with a confirmation box to initiate the firmware installation process. Click on *Continue*, and the firmware will start downloading, with a progress bar appearing just underneath the control bar options. Once the firmware has downloaded, the Wattmon will automatically begin uploading the firmware file, which, once completed, will be un-

compressed. At the end of the extraction process, a notification will ask you to reboot the Wattmon to complete the firmware process. Click *OK* and the Wattmon will reboot and reappear to the login screen, marking the completion of the firmware installation. For Wattmon devices that have the OTA feature enabled, you can use the *Schedule Install* button instead, which will start the update process in the background allowing you to continue working on the device. It will then automatically reboot once completed.



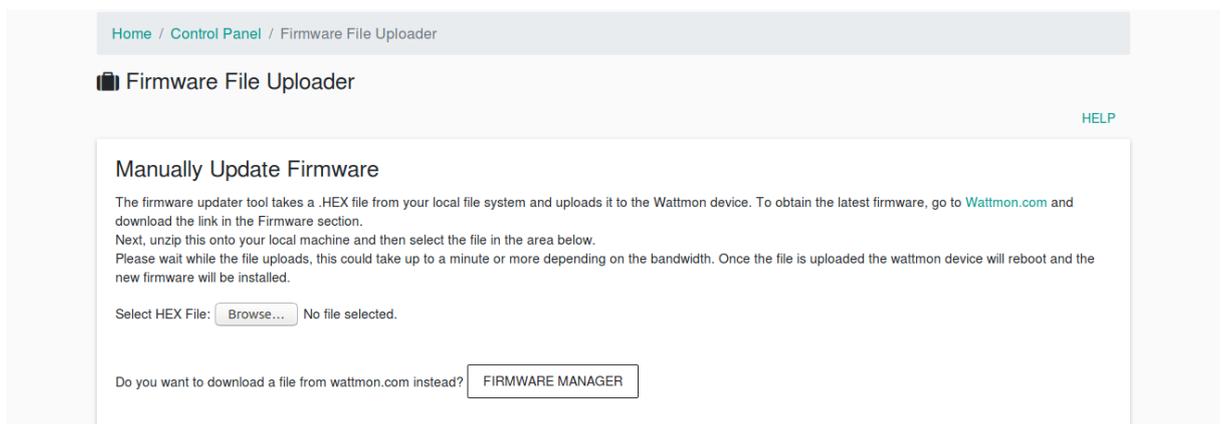
**Powering-off your Wattmon device during a firmware update will break the update, and you will need to force flash the firmware to get it back online.**

If neither your computer nor the Wattmon is configured for network access, scroll to the bottom of the *Firmware Manager* page, and you will find an option which says - *Do you want to upload a file from your computer instead?* This allows you to upload a firmware file from your computer. The firmware files can be obtained either via the Wattmon website or by contacting us.

While the *Update and Manage Firmware* page automatically displays the right firmware update based on your Wattmon device, the *Manually Update Firmware* page does not automatically detect the file version that you are uploading.



**Flashing a Wattmon with a non-compatible firmware will cause it to stop working. To fix this, you will need to force flash the correct firmware after placing it on the SD card manually first.**

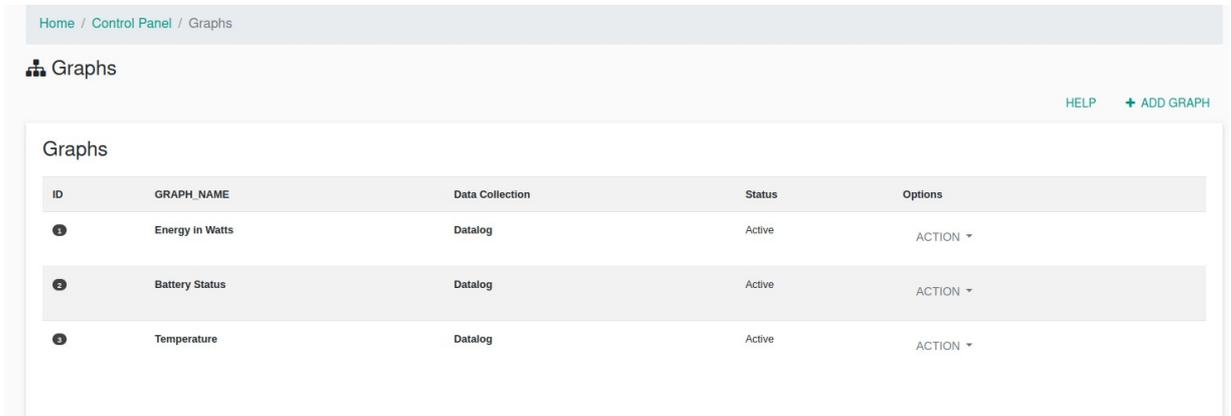


In order to perform this function, your Wattmon needs to be connected directly to a computer using a LAN cable. Be sure to upload the file properly and not to power off your Wattmon during a firmware update. Interrupting a firmware update can cause your Wattmon to stop functioning properly.

## GRAPHS

Wattmon has a basic graphing tool that lets you generate simple line graphs or bar charts from data in the logged CSV file.

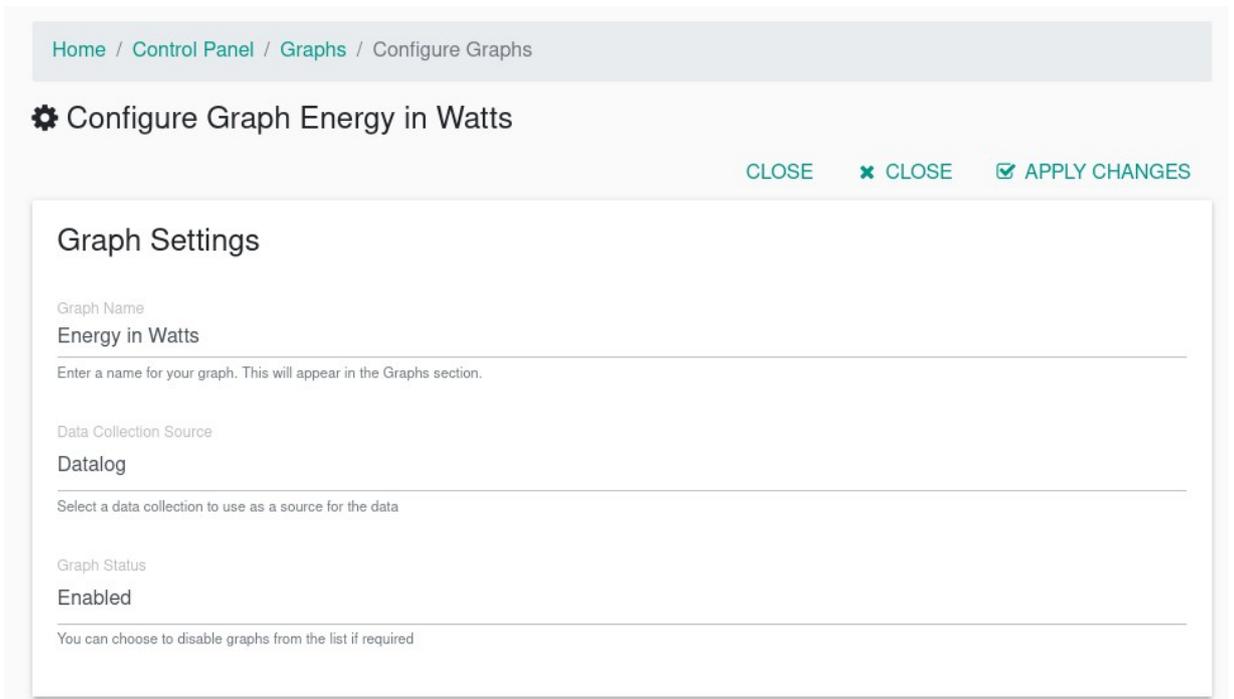
Open the *Graphs* page from Control Panel to show the list of available graphs. You can edit a graph, or click the *Add Graph* button at the top to create a new graph.



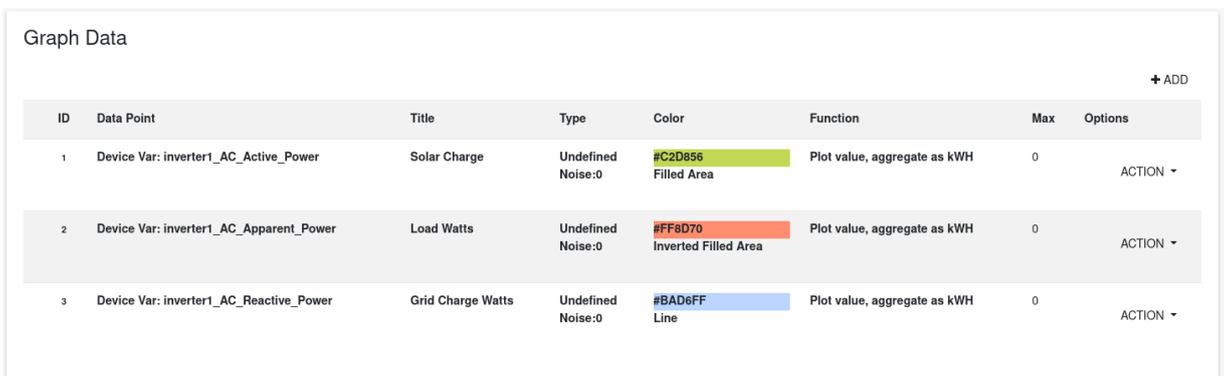
After clicking *Add Graph* or *Action > Configure*, you will get the *Configure Graph* page.

### Configuring a Graph

The *Configure Graph* page is split into two sections. The top section lets you select the basic data set and the name.



The *Graph Data* section lets you set which line or bar chart items you wish to see based on data in the CSV file.



The list has several columns:

- *Data Point*, which links to the CSV file's logged data point.
- *Title* which can be updated if required to describe the data point.

- *Type* lets you choose a data type (i.e. amps, volts, etc) which will help to group data on a single axis.
- The *Noise* field below lets you add a value that will make any datapoint value 0 if it is below the threshold.
- *Color* lets you choose a color, and below that you can choose how the graph is to be shown: Line, Filled Area, Inverted Line (Reverses the value to show it below the 0 axis) and Inverted Filled Area.
- *Function* lets you define how the data is processed. Several basic aggregation functions such as *Plot value*, *aggregate as kWh* or *Plot value*, *average total* allow you to do simple processing on your graphs.
- *Max* lets you set a maximum value after which the the value will be set to 0.

Clicking *Action > Edit* puts the row into edit mode, allowing you to make any changes required.

To save your changes, click *Apply Changes*.

## SCHEDULED TASKS

The Wattmon CRON scheduler is defined in a single text file called `crontab.txt`. This can be edited using the integrated file editor. To open the file, go to *Control Panel > Scheduled Tasks*.

```

1 # CRON scheduler
2 # enter each line with the following format:
3 # s m h d /path/script.cgi
4 # you can use a * to run it every change, or */x to run it every x iterations, or just specify a number to set a fixed time
5 # The following will run a script once a minute on second 1:
6 # 1 * * * /cronmin.cgi
7 # The following will run a script once a second:
8 # * * * * /cronsec.cgi
9 # The following will run a script once every 5 minutes:
10 # 1 */5 * * /cron5min.cgi
11 * * * * /scripts/cronsec.cgi
12 1 * * * /scripts/cronmin.cgi
13 15 */1 * * /scripts/ip.cgi
14 #35 */5 * * /scripts/pvoutput.cgi
15 0 0 0 * /scripts/cronday.cgi
16 30 59 23 * /scripts/logrotate.cgi
17 30 15 0 * /scripts/time.cgi
18
19

```

Enter each line with the following format:

s m h d /path/script.cgi

- **s** is the second is the second to run at or \* for every second
- **m** is the minute to run at
- **h** is the hour to run at
- **d** is the day of the month to run at
- **script** is the full name and path to the script file to execute

You can use the / to add multiple intervals. For example \*/5 in the s field means that it will match every 5 seconds.

Save your changes and reboot the Wattmon for them to take effect.

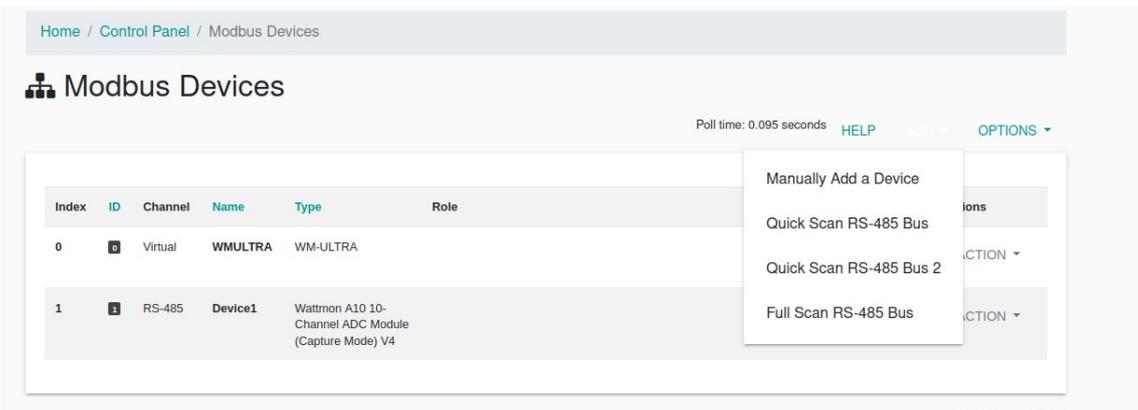
# DEVICES

The *Modbus Devices* page can be accessed through the *Settings* drop-down on the menu bar, or through the *Control Panel*. It displays all the active and inactive devices connected to the Wattmon, and allows you to add, remove, or customize any devices.

There are three options given on control bar - *HELP*, *ADD*, and *OPTIONS*. *HELP*, like on all other pages, displays details on the page.

*EZConfig* handles the creating and configuration of modbus devices, but you can manually change things here if required. Any changes you make will be overwritten if you run *EZ-Config* again. The *ADD* drop-down menu has the following options - *Manually Add a Device*, *Quick Scan RS-485 Bus*, *Quick Scan RS-485 Bus (on MEGA2, MEGA2-45, ULTRA)* and *Full Scan RS-485*.

The *Manually Add a Device* option is used when connecting a device to your Wattmon. *Quick Scan* and *Full Scan* are used to scan Wattmon modules once they have been physically connected to the Wattmon; *Quick Scan* searches for Wattmon modules on ID 1 to 10, while the *Full Scan* function searches for Wattmon modules on ID 1 to 100, and so is only used when the first option is unable to detect a device.



The *OPTIONS* drop-down offers five functions - *Refresh List*, *Enable/Disable Automatic Refresh*, *Enable Full View*, *Edit Communication Settings* and *Modbus Tester*.



*Refresh List* refreshes the list of devices displayed and notifies you of the total polling duration. *Enable Automatic Refresh* and *Disable Automatic Refresh* allow you to select whether you wish to have the refreshing done automatically every 5 seconds or manually.

The fourth option, *Edit Communication Settings* take you to the **Modbus Communication Settings** page to modify system-wide modbus settings.

The last option, *Modbus Tester* can be used to test modbus communication with connected devices, and is described in the **Modbus Tester** section.

The device list displays relevant data in columns, such as device *ID*, *Channel*, *Name*, *Type*, *Role*, *Status* and *Options*. *ID* represents the Modbus ID; The ID of Wattmon is always 0 on a virtual channel

Index	ID	Channel	Name	Type	Role	Time	Status	Options
0	0	Virtual	WMULTRA	WM-ULTRA		0.0s	Active	ACTION ▾

*Channel* shows the Modbus channel used for communication. When a device is connected using Modbus RTU, the Channel is *RS-485* or *RS-485 2*; if a device is connected using Modbus TCP, it is displayed as per *TCP Channel* used. A virtual channel also exists for virtual devices that are handled in code.

The *Name* column displays the name of the device as entered in EZConfig, which can be set by you. For certain third-party devices, such as inverters and energy meters, which allow for the model number and serial number to be polled using Modbus registers, the relevant details are displayed below the device *Name*.

The *Type* column displays the selected *Modbus Device Type* which is the driver selected in *EZConfig*.

If *Enable Full View* is enabled, *Role* lists each of the variables that are polled or logged from a particular device, divided into two parts. The first list displays the values that are logged by name, such as the voltage input on the WattmonMEGA called *A1 (Voltage of Power Pin)*, or the *Lifetime Generation* from an inverter. The second list is shaded, and features the actual names of the variables as they are logged on the Wattmon; the voltage input on the WattmonMEGA is logged as *A1 Voltage*, or the total generation from an inverter is logged as *inverter1\_kWh\_Total\_Active*.

The *Time* column shows the poll time of a device - this can be used to determine if devices are properly functioning or giving issues.

The *Status* column shows **Active** if the device is online and connected, and if not detected, then **Error** will appear.

The *ACTION* drop-down has the following options: *Configure*, which allows you to edit the name, channel and choose the logged variables of a particular device; *Enable/Disable*, which lets you set the state; *Delete*, which permanently removes the device from the Wattmon; and *Calibrate*, which is for recalibrating or resetting a Wattmon Module.

## CONFIGURE DEVICE PAGE

The *Configure Device* page lets you update important settings for a modbus device, such as the ID, name and bus.

The *Device Bus* lets you choose which bus the device is physically (or virtually) connected to.

The *Device ID* is the modbus ID assigned to the device

The *Device Name* is the name assigned (usually this is automatically assigned in EZConfig). Ensure the name is unique to avoid conflicts.

The *Poll Interval* determines the frequency at which the data is to be polled.

You can enable or disable the device status in the *Device Status* field.

### Configure Device

Device Bus  
**Modbus RTU (RS-485)**

Select the bus that this device is connected

---

Device ID  
**1**

Choose a unique device ID

---

Device Name  
**inverter1**

Enter a unique name for this device (i.e. inverter1)

---

Poll Interval  
**1000** SELECT

Select a polling interval in milliseconds

---

Device Status  
 Disabled  
 Enabled

Choose whether the device is active or not

In the second part of the page, The logged roles are shown.

### Configure Roles

Read Only Unsigned INT32 (Little Endian)			
Total AC Active Power			
inverter1_AC_Active_Power [logged]	Scale:1	Read Only Unsigned INT32 (Little Endian)	EDIT ROLE
Read Only Signed INT32 (Little Endian)			
Total AC Reactive Power			
inverter1_AC_Reactive_Power [logged]	Scale:1	Read Only Signed INT32 (Little Endian)	EDIT ROLE
Read Only Unsigned INT32 (Little Endian)			
Total AC Apparent Power			
inverter1_AC_Apparent_Power [logged]	Scale:1	Read Only Unsigned INT32 (Little Endian)	EDIT ROLE
Read Only Unsigned INT32 (Little Endian)			
Lifetime Generation			
inverter1_kWh_Total_Active [logged]	Scale:10	Read Only Unsigned INT32 (Little Endian)	EDIT ROLE

The roles that are currently linked are shown. *Undefined* values indicate that the role is not being logged. You can select *Create New Logged Role* which will add a role to the data logging engine. Click the *Edit Role* to edit an individual roles - this is described in the *Roles* section.

Clicking the *Calibrate* button takes you to the device calibration page. Clicking *Apply Changes* will apply the settings, but a reboot will likely be required for them to take effect.

## CALIBRATE DEVICE PAGE

The default calibration page lets you edit any read/write registers that are defined in the driver, and shows the raw register values in a section below.

For most devices (other than Wattmon branded) the Read/Write section will be blank. Use the read only Registers list to quickly determine if the device is properly reading values from the modbus device. Please note that the values are the raw registers and will not properly show the role values - many roles are composed of more than 1 register (i.e 32 bit float) so the values won't be displayed correctly here.

⚙️ Configure Device Sungrow PV String Inverters - SG33CX/SG33CX-US/SG40CX/SG50CX/SG55CX-US/SG110CX (7092)

HELP ✖️ CLOSE ✎ EDIT DEVICE ✅ APPLY CHANGES

### Read/Write Registers

---

### Read Only Registers

Daily Generation	0
Lifetime Generation / Total Active Energy	0
undefined	0
Total Running Time	0
undefined	0
Internal Temperature	0
Total AC Apparent Power	0
3	0
DC Voltage PV 1	0
DC Current PV 1	0
DC Voltage PV 2	0
DC Current PV 2	0
DC Voltage PV 3	0
DC Current PV 3	0

## CALIBRATE WATTMONULTRA

The WattmonULTRA calibration page lets you set up specific hardware settings for the ULTRA. The ULTRA comes pre-calibrated from the factory so the calibration constants are normally correct.

⚙️ Configure Device -1 (20)

DASHBOARD GRAPHS ▾ SETTINGS ▾ ADMIN ▾ Firmware Rev. 4.1347 - WattmonOS 3.20-20250310 (WM-ULTRA) - 68

Configure Device -1 (20) [EDIT DEVICE](#)

---

**Input Voltage** Voltage: 11.972 V Raw Value: 1369

Voltage multiplier (Input Voltage)  [CALIBRATE](#)

---

**Analog Channel 1** Calibrated Value: 0.000 Voltage: 0.000 V Raw: 0

Channel Mode  Voltage (0-10V DC)  
Choose the channel operating mode

Voltage Multiplier  [CALIBRATE](#)

Input Range Min

Input Range Max

Sensor Range Min

Sensor Range Max

There are 4 analog input channels, each of which can be configured as a 0-20mA or as a 0-10V input. To select the *Channel Mode* as 0-10V, uncheck the box.

## Voltage Calibration

The Analog Channel title (1-4) is followed by two dynamic values: *Calibrated Value* and *Raw*. The Raw ADC value is multiplied by the *Voltage Multiplier* which is automatically calculated during factory calibration but can be changed if required.

To obtain a range-adjusted value based on your sensor, enter the *Input Min* and *Input Max* values (voltage) and the *Sensor Range Min* and *Sensor Range Max* (ie. 0-1000). This will give the analog output a value of 1000 for a voltage of 10.

## Current Calibration

Channel Mode  Current (0-20 mA)  
Choose the channel operating mode

Similar to calibrating voltage, the *Calibrated Value* will reflect the range-adjusted inputs. For a 4-20mA sensor, enter 4 and 20 in the *Input Range Min* and *Max* fields. For a 4-20mA sensor that give you 0-200 degrees C output, you would enter 0 and 200 in the sensor range fields.

## Digital Inputs

Digital Inputs can be set in two modes. The default is *Digital input mode* where the DI value is 1 (on) or 0 (off). By clicking the checkbox you can switch into *Counter mode* that counts the pulses on the DI and returns a counter value for the DI.

### Digital Inputs

Digital Input 1 Mode	<input type="checkbox"/> Digital Input Mode (Value: 0) Choose the channel operating mode
Digital Input 2 Mode	<input type="checkbox"/> Digital Input Mode (Value: 0) Choose the channel operating mode
Digital Input 3 Mode	<input type="checkbox"/> Digital Input Mode (Value: 0) Choose the channel operating mode
Digital Input 4 Mode	<input type="checkbox"/> Digital Input Mode (Value: 0) Choose the channel operating mode

### Digital Outputs

Digital Output 1	0
Digital Output 2	0
Digital Output 3	0
Digital Output 4	0

[APPLY CHANGES](#)

## Digital Outputs

The status of the digital outputs is shown at the bottom the calibration page and can be used for troubleshooting to see if your actions or scripts are properly toggling the values.

## ROLES PAGE

Roles in Wattmon refer to variables that are linked to either a device, a global variable or a group. Roles are automatically created by EZConfig but can be manually modified or added to. Any changes made manually will be overwritten by EZConfig if it is run again. A role can be logged, and can be modified via uPHP scripts or via Actions. Role values are dynamically updated using the modbus polling engine, i.e. if a role called *inverter1\_AC\_Active\_Power* is linked to inverter 1, every time the polling engine runs the role will contain the latest active power of that inverter. A role timeout setting in the uPHP Settings page will clear the role value (set it to 0 or the device's persistent value) if communication with the device has failed for longer than the timeout setting.

The *System Roles* list shows the roles defined in Wattmon. The *Role Name* is automatically generated based on the device name and the device role selected. The role type can be one of the following:

- Read Only: This is linked to a modbus device's read only roles
- Read Write: This is linked to a modbus device's read write roles. Changing this role value in code or via actions will result in a modbus write to the holding register in the externally connected modbus device.
- Global: The role name is identical to the global variable name and any changes will reflect both ways
- Digital Input: The role is linked to a digital input type of a device
- Digital Output: The role is linked to a digital output of a device
- 1-Wire: The role contains the temperature of a 1-Wire sensor

ID	ROLE_NAME	Type	GROUP	DEVICES	Options
1	inverter1_AC_Active_Power	Read Only Value	3	7092	EDIT
2	inverter1_AC_Reactive_Power	Read Only Value	0	7092	EDIT
3	inverter1_AC_Apparent_Power	Read Only Value	0	7092	EDIT
4	inverter1_kWh_Total_Active	Read Only Value	0	7092	EDIT

To add a new Role, click the *Add Role* button. To edit an existing role, click the *Edit* button next to the role.

## ROLE SETTINGS PAGE

The *Role Settings* page is reached by editing or adding a role.

Each role has a unique *Role ID* generated by Wattmon. This should not be changed.

The role name is automatically generated but can be modified if required in the *Role Name* field.

*Show Role* determines if the roles is displayed in the dashboard Variables widget (if activated). This can be used to display certain variables if required, when customizing your dashboard.

*Group* lets you link a role to a group, which can be used for specific purpose such as accumulating watts as kilowatts. Select a group from the drop down field.

*Role Type* determines the role type (described in the section above). Roles of a certain type can be linked to certain devices. This is automatically generated by EZConfig.

*Role Definition* lets you change the way Wattmon handles the role value, when linked to a group. Options are *Current*, *Voltage*, *Temperature*, *Watts*, *Kilowatts* and a few more that have no specific function. If the role is linked to a group, each of these will function in a specific way:

- Current roles will be multiplied by a voltage role (if found) or the Wattmon input voltage for devices that support it to generate a Watt value for the group.
- Voltage roles will be used with current roles to calculate watts in the group
- Watts and kilowatts roles will be used by the group to generate kwh values if the group settings are defined as such.

 Role Settings for inverter1\_percent CLOSE  CLOSE  APPLY CHANGES

---

### Role Settings

Role ID  
5 ↕

Automatically assigned role ID

Role Name  
inverter1\_percent

Enter a role name that describes the role best. Refer to our [style guide](#) for standardized naming conventions.

Show Role  
 No  
 Yes  
Display role on dashboard

Group  
Not grouped

Select a group to associate the role with

Role Type  
Global Variable

Select a role type for this role which will affect the way Wattmon lets you assign it

Role Definition  
Value

Select a role definition for this role which will affect the way Wattmon uses its associated value

Role is logged  
 No  
 Yes  
Select if this role is logged in the data collection or not

Data collection file  
Datalog

Select which data collection should log this role

Not all roles are logged in the CSV file, as some roles may be used only for display purposes. To ensure the role value is logged, set Role is logged to Yes.

Finally, select the data log file. EZConfig only generates one file called Datalog which is the primary file. It is possible for Wattmon to have multiple log files for different types of data. This will break the export options but may be useful when working offline or with custom code. Ensure *Datalog* is selected to log to the primary log file.

Finally, click *Apply Changes* to save the role updates.

# GROUPS

Groups in Wattmon have specific functionality depending on the group type. The Groups list shows the groups that are defined - the below list is the default that comes with Wattmon.

 Groups [HELP](#) [+ ADD GROUP](#)

ID	Group Name	Type	Visible	Options
0	Not grouped	Not consolidated	Hidden	EDIT
1	Wind	Consolidated into Battery SoC	Hidden	EDIT
2	Grid	Consolidated into Battery SoC	Visible	EDIT
3	Solar	Consolidated into Battery SoC	Visible	EDIT

The Group name must be unique and can be modified. The Group type defines the function of the group and is described in the next section. The Visible section To create a new group, click the Add Group button, or click Edit next to the group to edit an existing group.

## CONFIGURE GROUP PAGE

The Group Configure page lets you change group settings.

 Configure Group Solar [HELP](#) [x CLOSE](#) [✔ APPLY CHANGES](#)

### Configure Group

Group ID  
3  
Enter a unique Group ID

Group Name  
Solar  
Select a meaningful group name. This will also appear on the dashboard

Group Type  
Consolidated into Battery SoC  
Select a group type that matches the functionality you require

Rated Watts  
0  
Enter the maximum watts that this group should handle

Group Visibility  
Visible  
Show this group on the dashboard widget if enabled

Group Icon  
 SELECT

Group Color  
#C0D800  
Select a color for the group

The *Group Id* is automatically generated and should not be changed.

The *Group Name* is a human readable name that lets you differentiate between groups.

The *Group Type* can be: *Not Consolidated*, *Consolidate into Battery SoC*, and *Tank Level*. If the *Consolidate into Battery SoC* is selected, the current role linked to this group will be used to update the Wattmon Battery Percent value. This is a legacy feature but can be used to keep track of single battery bank.

The *Rated Watts* can be used to set a maximum watts value above which any input value linked to the group will be discarded. For example if a solar irradiance sensor is linked to the group, the sensor value would never exceed 1200 watts/m2.

*Group Visible* option can be set to *Visible* to display on the dashboard if the appropriate widgets are enabled.

Choose a *Group Icon* if required, and a *Group Color*. The group color will be used to display the live values on the Live Energy graph in the default Wattmon configuration. Please note that after running EZConfig the main dashboard will be modified and you would need to manually restore the widgets to get that graph running.

## HTTP SERVER SETUP

The HTTP Server Setup page lets you change basic information about the functioning of the HTTP server which is how you interact with your Wattmon through the browser.

The screenshot shows the 'HTTP Server Settings' page. At the top, there is a breadcrumb trail: 'Home / Control Panel / HTTP Server Settings'. Below this is a gear icon followed by 'HTTP Server Settings'. On the right side, there are three buttons: 'HELP', 'CLOSE', and 'APPLY CHANGES'. The main content area is titled 'HTTP Server Settings' and contains the following sections:

- HTTP Access Log:** Two radio buttons are present: 'Don't generate HTTP Log' (selected) and 'Generate HTTP Log File'. Below them is the text 'Optionally log every HTTP request to a log file'.
- HTTP log File name:** A text input field contains '/logs/http.txt'. Below the field is the instruction 'Enter full name and path for the HTTP Log file' and a 'VIEW' button.
- HTTP Server Port:** A dropdown menu shows '80'. Below it is the instruction 'Change this from the default of 80 if you required'.
- Session Logged In Variable:** A text input field contains 'logged\_in'. Below it is the instruction 'The session variable that needs to be set non-zero in order for file saving to work'.

The *HTTP Access log* feature can be useful for troubleshooting connections but it is not recommended to keep it active as it will create a very large amount of data.

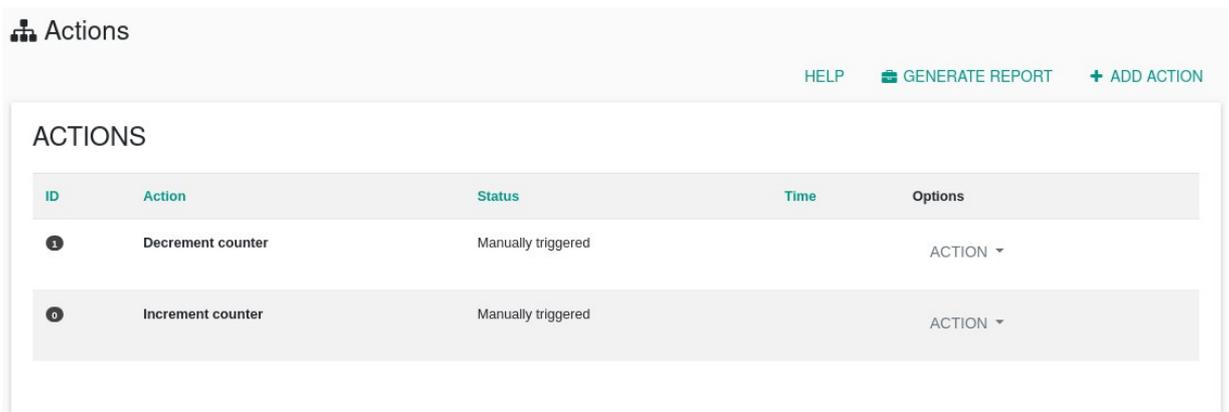
The *HTTP Log File name* determines where the HTTP Access log would be saved (if enabled)

If you require a custom port other than 80 you can enter it in the *HTTP Server Port* field. For example, if you set it to 8000, you would then need to connect via <http://192.168.0.55:8000/> instead of just <http://192.168.0.55>

The *Session Logged in Variable* should be left as `logged_in` for WattmonOS. If you are working with a completely customized setup you could change this to whatever you like, but doing so will break the WattmonOS login feature.

## ACTION

Actions in Wattmon can be used to automate simple tasks or respond to trigger condition with one or more basic actions. You can access the *Actions* page from *Settings > Actions* or via the *Control Panel*.



ID	Action	Status	Time	Options
1	Decrement counter	Manually triggered		ACTION ▾
2	Increment counter	Manually triggered		ACTION ▾

The Actions list show the current actions define. To create an action, click *Add Action*. To edit, disable or delete an action click the corresponding item under the *Action* drop down.

To view information about all actions in a single place, click the *Generate Report*.

### Wattmon Action Report for Cynergy Office Test

Report Generated on 2025-04-03 12:01:31

#### Action Decrement counter

This action is **manually executed**.  
This action requires **ANY conditions** to match in order to execute.  
This action **executes immediately** when conditions match.

##### Actions executed

1. Add -1 to Variable **global\_counter**.

#### Action Increment counter

This action is **manually executed**.  
This action requires **ANY conditions** to match in order to execute.  
This action **executes immediately** when conditions match.

##### Actions executed

1. Add 1 to Variable **global\_counter**.

As an example, this shows what a simple report could look like with 2 actions in it.

## ACTION SETTINGS

The Action Settings page lets you modify the settings for an action.

Enter a unique name for the action in the *Action Name* field.

Choose how the action gets triggered in the *When is this active* field.

Any of the below conditions match (OR) will trigger the action if any one of the trigger conditions are matched, whereas All of the below conditions (AND) required every single trigger condition to be matched to execute the action.

You can enter a delay in seconds before triggering in the *Delay before trigger* field. This can be useful to avoid false positives or if there are certain conditions that need to be matched for a certain amount of time before being considered relevant.

In the Action Type, select one of the following options:

- Disabled - the action does not get triggered every
- Always Enabled - the action will always execute (if the trigger conditions match)

- Enabled during time range - a field called *Between* will appear below. Enter the start and end hour and minute. This action will be ignored out of that time range.
- Disabled during time range - enter the time range to disable the action - it will execute during the time it is not within this range
- Execute at a fixed time of day - enter the time in HH:MM in the field below
- Execute at a fixed interval in seconds - this will automatically repeat the action every X seconds, define in the *Delay* field
- Manually Triggered - this will show a button to execute the action if the Control widget is visible.

Action Settings for Increment Counter

HELP ✕ CLOSE ✓ APPLY CHANGES

---

### Action Settings

Action Name  
Increment Counter

Enter a unique name for the action

When is this active?

Any of the below conditions match (boolean OR)  
 All of the below conditions match (boolean AND)

Select how the trigger conditions are evaluated

Delay (secs) before trigger  
0

The delay is used to ensure that parameters are met for a certain time to eliminate false positives. When used in Repetitive Execution mode, this is the delay between executions.

Action Type

Disabled  
 Always Enabled  
 Enabled during time range  
 Disabled during time range  
 Execute at fixed time of day  
 Execute at fixed interval in seconds  
 Manually triggered

Choose the action type that matches your requirements above

Between  
00:00 and 23:59 (HH:MM)

Enter a start and end time in the form HH:MM - make sure the start time is less than the end time.

## Trigger Conditions

One or more trigger conditions can be defined that need to match before the action is executed.

Trigger Conditions + ADD

ID	Value Type	Value 1	Comparison	Value 2	Value 2 Type	Options
1	Global Variable	Global Variable global_counter	Is Less than	Fixed Value	Preset Value 10	ACTION ▾

The list shows the various trigger conditions. You can click *Add* to add a new trigger condition and *Action > Delete* to remove it.

In the *Value Type* you can choose between:

- *System Variable* - You can select from a list of precalculated system variables - this would include calculated values in groups
- *Device Variable* - You will need to select from a list of all roles in the system

- *Fixed Value* - enter a value in the Value 1 field
- *Global Variable* - You will need to enter the name of the global in Value 1

In the *Comparison* field, choose a logical comparison from one of the following: *Is Less Than, Is Greater Than, Is Equal To, Is Less Or Equal, Is Greater or Equal* and *Not Equal*

For the *Value 2* field, the drop-down lets you select one of the types as with *Value Type* above. Enter the second part following the same logic.

The comparison will result in a binary 1 or 0 result. Depending on the *When is this Active* field above, the action will run or not - for example if you have 2 triggers and one of them returns 1 (true) while the other returns 0 (false), the action will only run if you select *Any of the conditions match*.

## Actions

The Actions list lets you add, edit and remove actions that will be executed sequentially.

Actions				+ ADD
ID	Action	Variable	Value	Options
1	Type Add to Global Variable	Variable global_counter	Value 1	ACTION ▾

Click the *Add* button to add a new action item, or use the *Action* button at the right of each row to *Edit* or *Delete* an existing action item.

In the *Type* field, select one of the following actions:

- *Switch On* - this will turn on a digital output connected to a device. The *Role* field will appear with a list of all digital output roles. If you are trying to use the Digital Output of the WattmonULTRA, ensure that you have logged them in EZConfig.
- *Switch Off* - this will turn off a digital output. Select a *Role* that is of type Digital Output
- *Send Email* - will send an email. Enter an email address in the *Email* field and the subject in the *Subject* field
- *Set Global Variable* - enter the global variable name and enter a value to set it to
- *Add to Global Variable* - enter the global variable name and enter a value to add to the global (this can be negative, to decrement the value)
- *Toggle Global Variable* - Will switch the global value between 1 and 0 each time.
- *Set Role Value* - This will let you set a role value and applies to all read/write roles in all devices. Enter the value and it will update it in the device itself (i.e. if you have a Percent value in an inverter, and you set the value to 50, it will send a command to the inverter to update it's percent to 50.
- *Exec Script* - execute a uPHPs script. Enter the path in the Script Path field
- *Set Global Variable from Role* - Enter the global variable name, and select the role. The global will be updated with the value of the role. This is useful if you want to do some calculations on multiple role values.
- *Multiply Global Variable* - Enter the global variable name, and in the value field enter a number to multiply it by
- *Add Global to Global Variable*. Enter a global variable to be added to in the Variable field. The value of the From Variable will be added to it and stored there.

- Switch On
- Switch Off
- Send Email
- Set Global Variable
- Add to Global Variable
- Toggle Global Variable
- Set Role Value
- Exec Script
- Set Global Variable from Role
- Multiply Global Variable
- Add Global to Global Variable
- Delay (ms)
- Set Global Variable from NVRAM variable
- Set NVRAM Variable from Global variable
- Add Alert
- Set Role From Global
- Send SNMP Trap Message

- *Delay (ms)* - Enter a delay value in milliseconds before jumping to the next sequence action item. This could be used to toggle a relay that requires a certain time before switching properly, for example.
- *Set Global Variable from NVRAM variable* - this will update the value of a global from the NVRAM stored variable
- *Set NVRAM Variable from Global Variable* - update the NVRAM value in the NVRAM variable field to the value of the Global variable
- *Add Alert* - add an alert message to the alert log. This will be transmitted to the ems cloud server and show there if alerts are enabled
- *Set Role from Global* - Enter a read/write role and enter the global variable name to set the role value to it.
- *Send SNMP Trap message* - Enter a message to be sent to the SNMP manager (if SNMP is enabled and configured)

## DEVICE TYPES PAGE

The *Device Types* page lists all the device drivers currently available on the Wattmon. Access this page through the *Control Panel*.

The screenshot shows the 'Device Types' page. At the top, there's a header with 'Device Types' and three links: 'HELP', 'LEGACY DRIVERS', and '+ NEW DEVICE DRIVER'. Below the header is a 'Device Type List' section. It includes a 'Filter:' label and a text input field with the placeholder 'Enter text to filter'. The main content is a table with the following data:

Device Type	Name	Category	Options
7076	SMA Sunny WebBox (STP nn000TL-10)	Inverter	EDIT
6380	HPL Emfis Plus series	Power Meter	EDIT
6052	Legrand EMDX3 N 4 120 52	Power Meter	EDIT
6010	Pyranometer (Virtual)	Sensor	EDIT

The list includes inverters, energy meters, sensors and other device drivers that have already been created. Each device has been assigned a unique *Device Type Id* which is visible in the first column. The *Name* column indicates the make and model of the device, and the *Category* shows which category of device it is (Inverter, Meter, Sensor, Other). You can quickly search for your driver by typing in the *Filter* box.

To create a new device driver, click the *New Device Driver* button. To edit an existing one, click the *Edit* button.

## DEVICE TYPE EDIT PAGE

The Device Type Edit page lets you create and edit device drivers. A device driver is nothing other than a mapping for a set of modbus registers that explains the Wattmon polling engine how and what to read (or write) from an external device.

HELP   ✕ CLOSE   ✓ APPLY CHANGES

### Device Driver Configuration

Device Type ID  
7092

Unique identifier for this device driver type

Device Driver Name  
Sungrow SGxxCX PV String Inverter

Enter a meaningful name such as the device make and model

Device Driver Description  
Sungrow PV String Inverters - SG33CX/SG33CX-US/SG40CX/SG50CX/SG55CX-US/SG110CX

Enter a description to identify this device driver

Device Category  
Inverter

Select a category for this device driver

Inverter Rating in kW  
0

The *Device Type ID* can be modified if you want to make a copy of an existing driver. For example, if you want to copy the 7092 to your own new driver id 8092, change the ID and save it. It is not recommended to modify this unless you are adding your own custom drivers.

The *Device Driver Name* should be a unique name describing the modbus device.

The *Device Driver Subscription* can list more information about the driver so the user can understand more before selecting it.

Select the *Category* of device (Inverter, Power Meter, Charge Controller, Sensor, Other, Wattmon Master). Inverters and Power meters have special processing in the PC (power control) package, as well as in the EZDisplay and EZConfig so ensure to select the proper category when creating the driver.

Enter the *Inverter Rating in kW* if you have selected the Inverter category.

The *Number of Digital Inputs* should not be modified directly as this will be updated in the Digital Inputs section below.

The *Digital Input Start Address* lets you enter a starting address of the block.

The *Number of Digital Outputs* should not be modified directly as this will be updated in the Digital Inputs section below.

*Number of Read Only Registers* show the count of registers entered in the *Read Only Register* section.

Number of Read Only Registers  
95

The number of read only registers

Read Type

Single Block

Multi-block mode

Select the type of read to perform

Read-only register function number  
4 - Read Input Register

The *Read Type* radio lets you select between *Single Block* and *Multi-block mode*. For most devices *Multi-block mode* is required, especially if you plan to read from multiple register ranges. *Single block* mode lets you enter a start address below, and the *Address* field in the Read Only register list will disappear.

The *Read Only register function number* can be selected as *3 - Read Holding Register* or *4 - Read Input Register* depending on the device you wish to communicate with. Refer to the device modbus mapping for information on that. From OS 3.20 onwards, selecting DLMS will treat this device as a DLMS device instead of a Modbus device.

The *Number of Read Write Registers* displays the count of registers entered in the *Read Write* section below.

Number of Read Write Registers  
0

The number of read/write registers

Write Type  
 Single Block  
 Multi-block mode  
Select the type of write to perform

R/W Polling Method  
Poll RW registers every polling loop

Choose how the Read/Write registers are polled

The *Write Type* lets you choose the way you enter the read write registers – *Single block* will only allow you to read a single range and you will need to enter the start address below this. *Multi-block mode* lets you individually enter the register addresses.

The *R/W Polling Method* lets you choose how the Read Write registers are polled. You can choose between *Poll RW registers every polling loop* or *Poll RW registers once* which is useful for values that do not change in a device such as configuration values.

## DLMS Settings

Read-only register function number  
DLMS

Modbus Function code for Read Only registers

DLMS Authentication Mode  
Low

Select the DLMS authentication mode

DLMS Password  
ABCD0001

Enter a description to identify this device driver

DLMS Client ID  
32

Enter a client ID

If the read-only register drop down is set to DLMS, three additional configuration values appear.

In the DLMS Authentication mode, select None or Low. Currently Wattmon does not support other modes.

For the Low Authentication, enter the default DLMS password as per the meter manual.

Depending on the required setting, enter the Client ID. For example for a Secure meter it is 32 whereas for an L&T meter it is 17.

## Digital Input Registers

Digital Input Registers use modbus function 1 - Read Coil Status to get the values from a modbus device starting at the address entered in the first section (*Digital Input Register Start Address*). Click *Add* to create a new entry or *Action > Edit* to edit an existing row.

Digital Input Registers			
Register Index	Register Name	Description	Options
0	di1	Input value 1	ACTION ▾

Enter a name for the DI register and a description. This will ensure the register gets polled.

## Digital Output Registers

Digital output registers use modbus function 5 (Force single coil) to set an output in a device that supports this. Click *Add* to add a new row or *Action* to modify an existing one.

Digital Output Registers			
Register Index	Register Name	Description	Options
0	switch1	Sets digital output 1	ACTION ▾

Enter a register name and a description that is meaningful.

## Read Only Registers (Modbus)

The Read Only register section as explained above can use function 3 or 4 depending on the setting. In multi-block mode you will be able to enter the Register Address. Ensure that you group registers together to avoid multiple polls as much as possible and include dummy registers in between nearby block that need reading. For example, if you have a register block starting at 5002 and you want 5002,5003,5005 and 5006 then enter 5004 in between so it still reads as a single block, otherwise two modbus commands will go out, increasing the polling time unnecessarily. To add a new register, click the *Add* button. To modify a row, click the *Action* button and select *Move Down*, *Move Up*, *Insert After*, or *Delete*. This will change the position of the register in the list. Note that the *Roles* section below will need to be manually changed to reflect this.

Read Only Registers

HEX CONVERTER EDIT ALL ADD

Register Index	Register Address	Register Name	Description	Persistent	INTERVAL	Options
0	5002 5002	kWh_Day_Active	Daily Generation	<input checked="" type="checkbox"/>	0	ACTION ▾
1	5003	kWh_Total_Active	Lifetime Generation / Total Active Energy	true	0	ACTION ▾
2	5004	newreg62	undefined	true	0	ACTION ▾
3	5005	Total_Running_Time	Total Running Time	true	0	ACTION ▾
4	5006	newreg64	undefined	true	0	ACTION ▾

The *Register index* is automatically generated and cannot be changed.

The *Register address* should as explained above be entered matching the register map for the device you wish to interface with. Note that some register maps add 1 to the address so you will need to see what is the correct value by using Modbus tester for one of the registers in the driver to determine this if you are unsure.

The *Register Name* should follow the *Wattmon Naming Conventions* to ensure compatibility across devices – this is included in the appendix at the end of this guide.

The *Description* field can be identical to the name or could have more information if required.

The *Persistent* checkbox can be selected if Wattmon should remember the last values read even when the device goes offline – this is typically used for kWh counter or other values that are used in calculations and do not change when the inverter may be offline at night, for example.

The *Interval* box lets you enter an interval to read the register at in terms of polling cycles. Leaving it at 0 means that the block will be read every cycle. The cycle time can be seen in the Device List at the top (Modbus Polling Time). You can set all the Interval values for a block that does not need to be read regularly to a higher value to optimize the polling time, especially if many devices are present on a single RS-485 channel. For example, a kWh register could be polled every 5-10 cycles if it is in an address range that does not overlap with a value that needs to be read every cycle such as active power.

## Read-Only Registers (DLMS)

In DLMS mode the Register Names need to be populated with the OBIS codes from the DLMS meter registers required. Wattmon will read 32-bit signed or unsigned int values so ensure that you add a dummy second register as shown below. If you wish you retain values (i.e. for kWh registers) then select the Persistent field for both registers.

Read Only Registers

[HEX CONVERTER](#) [EDIT ALL](#) [ADD](#)

Register Index	Register Name	Description	Persistent	INTERVAL	Options
0	1.0.32.7.0.255	AC_Voltage_AN	false	0	ACTION ▾
1	newreg2	undefined	false	0	ACTION ▾
2	1.0.52.7.0.255	AC_Voltage_BN	false	0	ACTION ▾
3	newreg4	undefined	false	0	ACTION ▾
4	1.0.72.7.0.255	AC_Voltage_CN	false	0	ACTION ▾
5	newreg6	undefined	false	0	ACTION ▾

In the Roles list, select the appropriate scaling.

Roles

[EDIT ALL](#) [ADD](#)

Role ID	Role Name	Role Type	Register Index	Role Scale	Options
0	AC_Active_Power	Read Only Signed Long Big Endian (32-bit)	6 (1.0.1.7.0.255)	1000	ACTION ▾
1	kVARh_Lead_Total_Import	Read Only Unsigned Long Big Endian (32-bit)	22 (1.0.8.8.0.255)	1	ACTION ▾
2	kVARh_Lag_Total_Import	Read Only Unsigned Long Big Endian (32-bit)	24 (1.0.5.8.0.255)	1	ACTION ▾
3	kWh_Total_Import	Read Only Unsigned Long Big Endian (32-bit)	14 (1.0.1.8.0.255)	1	ACTION ▾
4	kVAh_Total_Import	Read Only Unsigned Long Big Endian (32-bit)	20 (1.0.9.8.0.255)	1	ACTION ▾
5	AC_Voltage_AN	Read Only Unsigned Long Big Endian (32-bit)	0 (1.0.32.7.0.255)	100000	ACTION ▾

In the One-Time string data, create a single row of type Serial number. The appropriate OBIS code will be polled and the value will be placed in that field automatically.

One-time String Data

[HEX CONVERTER](#) [EDIT ALL](#) [ADD](#)

Type	Modbus Function	Start Address	Count	Options
0	Serial Number			ACTION ▾

By setting the device read function to DLMS, the appropriate algorithm will be taken used to read the devices using the polling engine for both TCP and RS-232 channels. DLMS has not been implemented on RS-485 channels. Ensure a single protocol per channel (don't mix Modbus and DLMS).

## Read Write Registers

The Read/Write Register is similar to the Read only section described above.

Register Index	Register Address	Register Name	Description	Persistent	Options
0	4000	Inverter_Percent	Inverter percentage	<input type="checkbox"/>	ACTION

Add your registers in a similar manner. Read/Write registers always use Modbus Function 3 for polling, and would use Function 6 or 16 depending on the option selected at the beginning of the page.

## One-time String Data

Certain meters and inverters provide serial number and model number information. This can be obtained through the *One-time String Data* section. Add a row by clicking the *Add* button, or edit a row by clicking the *Action* button.

Type	Modbus Function	Start Address	Count	Options
Serial Number	4 - Read Input Register	4989	10	ACTION

The *Type* column lets you select between *Serial Number*, *Model Number* and 3 *User Defined* fields.

Select the *Modbus Function* (3 or 4), the *Start Address* and the register *Count*. The One time data will be polled once when the device becomes active after boot. The *Serial number* and *Model number*, if properly formatted, will appear below the device in the *Devices* page and be accessible using uPHP scripts.

## Roles

Roles need to be created to link the registers with the Wattmon system.

Role ID	Role Name	Role Type	Register Index	Role Scale	Options
0	AC_Active_Power	Read Only Unsigned Long (32-bit) Lit	28 (AC_Active_Power)	1	ACTION
1	AC_Reactive_Power	Read Only Signed Long Little Endian (32-bit)	30 (AC_Reactive_Power)	1	ACTION

To create a new role, click the *Add* button. To modify a role, click the *Action* button.

In the *Role Name* field, enter a role name as per the Wattmon Naming Convention as this will allow your device to smoothly interface with the existing eco system. There are certain important role indices that need to be adhered to as well. Role index 0 should always contain the AC\_Active\_Power for the device, if available while role index 3 should contain the kWh\_Total\_Active value.

Select the *Role Type* from the drop down matching the register type described in the manual. For example, in the Sungrow driver example used above, the register is a 32-bit unsigned long in Little Endian format.

You will need to manually enter a *Register Index* that matches the index of the register in the read only registers section above for Read Only role types. If you select a Read Write role type, the index needs to refer to the register index in the Read Write section. The name of the register will appear above. For the example above, register index 28 corresponds to AC\_Active\_Power.

The *Role Scale* can be used to scale values that require it – for example to scale watts to kW, enter 1000 as the role value is divided by the scale value. A special case of dynamic scale factors is handled by entering a negative whole number in the scale field. This should be the index of a read only register that contains the scale factor. Usually scale factors would be treated as  $10^{\text{scale factor}}$  but different devices give different scale factor values. Wattmon can automatically recognize the most common ones and handle them properly.

## UPHP SCRIPT SETUP PAGE

The uPHP Script Setup page contains important settings that define how the Wattmon firmware behaves.

The *Debug Method* drop down lets you select from *Serial Port* and *Log File*. Unless you are a developer, select *Log File* to ensure log entries are properly stored in the system log.

The *Debug Level* lets you choose what types of messages to log. *Medium* will ensure that only important messages are logged. For troubleshooting, set this to *Verbose*.

The *Maximum Execution time* field lets you set the maximum time a uPHP script can run. If you notice that pages fail due to a large number of data points being logged for example, you can increase this time to see if it solves the issue. The default value is 20 seconds.

*Role Timeout* determines how long a role can retain its last value before being reset to 0. This typically would happen if a modbus device stops responding or the polling loop time is too high. The default value of 15 usually works but in some cases it may need to be increased. If you see zero values in your CSV randomly it could indicate the roles are timing out.

*Device Timeout* is the number of seconds a device can remain inactive in the device list before being shown in the Error state.

Ignore persistent 0xFFFF modbus registers  
No

Some devices remain active but send 0xFFFF values in certain conditions

Max Concurrent Script  
6

Number of scripts that can run at the same time. Reduce this to prevent memory allocation errors.

Concurrent Modbus TCP Tasks  
1

Poll TCP devices in parallel by increasing this value. Suggested max value is 3 or MINI3, and 5-10 on Mega2/Ultra

Stack Size  
1200

Stack size in 4-byte blocks per process

Call Depth  
5

Call depth for custom function calls

Session Timeout  
6000

Seconds before a session expires

For certain device type (i.e. SMA inverters) a value of 0xFFFF hex gets returned when part of the inverter is non-functional such as at night. Wattmon can choose to ignore these values and retain the last known good values by selecting Yes in the *Ignore persistent modbus registers* field.

The *Max Concurrent scripts* determines how many scripts can run at the same time (concurrently) on a Wattmon. For the WattmonMEGA, MINI3 and MINI3-4G models it is not recommended to set this above 3 and may even need to be reduced to 2 when logging many parameters. For the MEGA2, MEGA2-4G and ULTRA you can set this number to 10 or 15 easily.

*Concurrent Modbus TCP Tasks* lets you create multiple threads, each dealing with an equal number of TCP channels paralelly, which greatly increases the speed at which devices are polled. For lower-end hardware it is not recommende to increase this above 2. If you select 1 then all the TCP channels will be polled sequentially.

*Stack Size* determines the stack size. This number is actually multiplied by 4, so a stack size of 1000 consumes for 4KB of memory. For lower end models 1200 is fine, and when dealing with certain packages that require more stack memory, can be increased.

Script to run at boot  
/scripts/runonce.cgi

Select a script to execute at boot - leave this as the default /scripts/runonce.cgi unless you know what you are doing

Error log file  
/logs/log.txt

[VIEW](#)

File name including complete path to the error log file

Log File Mode  
Multiple files (network, ftp, mqtt, log, etc)

Choose how logs will be stored in the logs folder

The *Call Depth* determines how many function calls can be nested. The default value is 5 and suitable for most cases.

The *Session Timeout* is set to 6000 seconds by default but can be increased or reduced. This will cause the user to be logged out if the session cookie is not refreshed within that interval.

The *Script to run* at boot lets you run a script once at startup. WattmonOS requires this to be */scripts/runonce.cgi* and this should never be changed unless custom programming scripts are to be used.

The *Error Log file* lets you use a non-standard log if required but it is recommended to leave it at the default */logs/log.txt*.

The *Log File Mode* can be changed to Single File or Multiple Files. Multiple files mode is recommended as it will split logs into meaningful categories, such as *network.txt*, *log.txt*, *mqtt.txt*, etc.

**File System Consistency**

Auto Check File System daily  
No

Automatically scan the file system every day for changes

Auto-restore changed packages  
No

Automatically restore files that have been installed from a repository if they are changed/corrupted

*File System Consistency* checking allows your Wattmon to check for corruptions and potentially restore them.

Select *Auto Check File System daily* to enable this feature.

To *Auto Restore changed packages*, select *Yes* in the corresponding drop-down. This will detect if a package script is corrupt and restore it from the *.tar.gz* file in the repository folder if it is found.

**Data Logging**

Firmware-based logging  
No

The role values will be automatically collected at the firmware level

**RAM Drive**

RAM Drive for CGC Files  
Enabled

If you have a device that is giving issues with DDR RAM you can disable this

The *Firmware-based logging* option should be enabled whenever possible as it provides a much faster and less memory-intensive way of generate the CSV log lines at the selected logging interval. For legacy OS code you can disable it.

The *RAM Drive* feature is available on devices with external flash (MEGA2, MEGA2-4G and ULTRA). Set *Ram Drive for CGC files* to *Enabled* to place the compiled script files in the RAM drive to further improve performance and reduce the wear and tear on the SD card.

Changing uPHP settings will require a reboot for changes to take effect.

# WATTMONOS PACKAGES

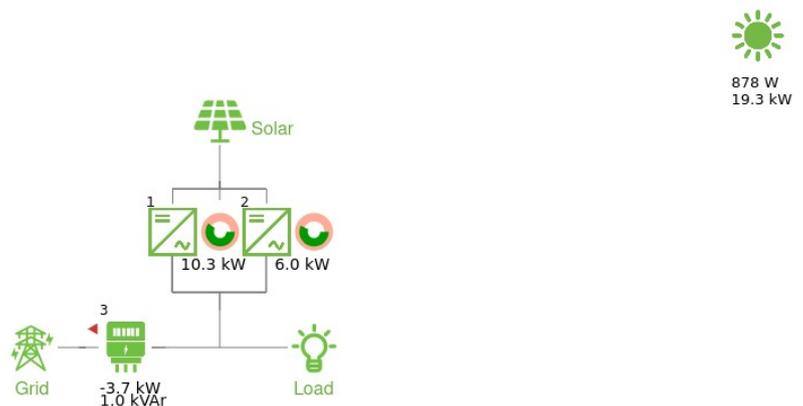
The following sections describe the Wattmon packages available and their function. Some of the packages come pre-installed in Wattmon but if they are not present they can always be installed using the *WattmonOS Updater* feature under Control Panel. Refer the the section above on how to install them.

Packages are distributed in the form of a compressed *.tar.gz* file and can also be manually uploaded and extracted if you have been provided with a file by the Wattmon team. Package scripts reside in the */package* folder. If you have your own custom packages, they will not be available through the WattmonOS updater unless you add the repository URL.

## EZDisplay Package

The *EZDisplay* package generates a widget that shows on the main dashboard. It has a large canvas that dynamically adapts to the configuration set in EZConfig and PC. Inverters appear in a single row, displaying their Modbus *Address* or IDs, and the *AC Active Power* in kilowatts. Meters will appear in their respective places (grid, solar output, or genset) and the flow of energy will indicate import or export power. If a pyranometer is installed, the solar irradiance and potential generation is shown at the top right as seen in the image below.

Live Energy Overview



This package gets automatically installed after running EZConfig

## Devices Package

If you chose the *Device Info Widget* during the *EZConfig* setup, the *Dashboard* displays a widget showing the inverters along with their *Serial Number*, *Model Number*, *Active Power*, *kWh Total* and *Status Code* for each of the inverters connected to your Wattmon. Energy meters are displayed with their *Serial Number*, *Active Power*, *kWh Total Import* and *kWh Total Export* values.

Each of the devices are listed with a small dynamic icon right before their name, which is color-coded to display the communication status of the inverter or power meter. If the icon is red in color, then the device is inactive or has not communicated with the Wattmon in over two minutes. If the device is active and communicating with the Wattmon, then the icon is green in color.

Devices

Inverters

- SOLAREEDGE INVERTER 0 ▾
 

Serial Number	██████████
Model Number	-----
Active Power	10359.000000
kWh Total	64090.152344
Status Code	4
Event Code	0
- SOLAREEDGE INVERTER 1 ▾

Meters

- SECURE ELITE 440-445 0 ▾

## Power Control (PC) Package

The PC package handles the Wattmon proprietary power control algorithm that can be used for zero feed in (ZFI) also known as zero export, for genset protection (running your solar plan along with your diesel or gas generator), and for other applications that require dynamic power limiting of inverter power.

Power control requires a license key that needs to be purchased from Wattmon which will activate various features in the PC package. License key types are: *ZFI Only*, *ZFI+DG* (running both zero feed in and genset protection), *ZFI+DG+PF* which also performs reactive power compensation to improve the power factor on the source meter, and finally *ZFI+DG+PF+Cluster* which activates multi-cluster mode, allowing independent power control pools with multiple meters.

To access the power control settings, first proceed to the *Control Panel*. Scroll down to the *User Settings* section and click on the red-coded *Power Control Settings* option. Upon entering the *Configure Power Control page*, you will notice that to access and modify the *General Settings*, you have to first read and accept the terms and conditions. This basically outlines the functionality and sheds some light on the capabilities and the shortcomings of the software-based power control solution offered with the Wattmon.

In order to proceed further, you have to click on **AGREE**.

If you accept the terms and conditions as shown on the previous page, the complete *Configure Power Control page* will load, displaying three distinct sections - *General Settings*, *Power Meters* and *Inverters*.

### General Settings

The *General Settings* section lets you enabled and disable power control, and configure the way power control works.

### General Settings

Terms and conditions You have already accepted the terms and conditions.

Status

- Disabled
- Enabled

Keep this enabled for active power control

License Key

2HsQr8\*MP\*E...M6...QvrfY80P/GEV/ME...EaC88x...CaA

Get Key

Enter a valid license key to enable power control

This license code is valid and will execute code on this Wattmon. This key supports: Zero Feed-In,DG Protection,Dynamic PF Correction.

Debug

- Disabled
- Enabled

Debug mode enabled

Debug to Log File

- Disabled
- Enabled

Log debug messages to file

The *Status* option allows to enable or disable power control as required.

The *License Key* field is where a valid license key has to be entered to authorize power control on Wattmon. Click *Get Key* to update this from the Wattmon license server. After entering the license key, you need select *APPLY CHANGES* and then refresh the page to verify if the key has been accepted by your Wattmon. If the license key is incorrect, you will view a message displayed below the field which states, “**The license key is not valid for this device. If you are just entering it now, save the page and refresh to check if it is valid.**” If the license key is correct, the message reads, “**This license code is valid and will execute code on this Wattmon.**”

**Note:** The Wattmon Power Control license key is a 64-character alpha-numeric that is unique to each Wattmon and is provided by us upon purchase. No two licenses are identical, and therefore cannot be used interchangeably.

The *Debug* option if set to *Enabled* allows you to view the real-time percentages for active power set on each inverter, on the Main Dashboard of your Wattmon.

The *Debug to Log File* option is a continuation of the previous option, which lets you log the active power percentages data in a dedicated log file when set to *Enabled*.

Update Interval  
5

Update interval in seconds

---

Meter Timeout  
20

Meter Timeout in seconds

---

Inverter Timeout  
60

Inverter Timeout in seconds

---

Modbus retries  
1

Number of packets to resend in case of transmit errors on Modbus

The *Update Interval* is the response time taken in seconds by the Wattmon to set or change the percentages set on an inverter. This option is currently locked down to 5 seconds and cannot be changed.

The *Meter Timeout* value is the amount of time given for a response from a power meter. The default is set at 20 seconds. If the meters do not reply with the polled value in this set time, the Wattmon falls back on default fail-safe set in the next sections.

The *Inverter Timeout* value, like the Meter Timeout, is the amount of time given for a response from an inverter. The default is set at 60 seconds. If the inverters do not reply within this set time, the Wattmon falls back on default fail-safe set in the next sections.

The *Modbus Retries* option lets you set the number of packets to be resent to the Modbus devices in case of transmission errors. The default is set to 1, which means the Wattmon polls each device in succession and does not poll any erroneous device more than once continuously, instead returning to it in the next polling cycle.

If your license key supports PF control, a *Reactive Offset* field lets you enter an offset to compensate for reactive losses. This could be a positive or negative value - leave it at 0 if you are unsure what to put.

The screenshot shows a configuration window with the following sections:

- Power Control Mode**
  - Average power for all phases (default)
  - Minimum power on any one phase

Phase-wise power control allows you to ensure that no export happens on setups where there is an imbalance in the power on each phase.
- Reactive Control Method**
  - Average power for all phases (default)
  - Minimum power on any one phase

Phase-wise power control allows you to ensure that no export happens on setups where there is an imbalance in the power on each phase.
- Percentage Control method**
  - All inverters at the same percent
  - Individual percent calculated per inverter

This can be used to optimize generation in larger plants where inverters may not generate equally always.
- Set percent to 100%**
  - Use calculated percent
  - Set all inverters to 100% when irradiance is less

Use this option to set all inverters to 100% when calculated potential generation would not exceed power setpoints

The *Power Control Mode* option allows you to control the active power output from your inverters either at the average level or at phase level, thus ensuring that no energy is exported even if there is an imbalance in the active power on each phase. The default is set to *Average power for all phases*. To enforce power control at the phase level set this to *Minimum power on any one phase*. When selecting this ensure the meter you have installed supports this.

The *Reactive Power Control Mode* option lets you choose whether to use average or phase-wise reactive power. This only applies to licenses that support it.

The *Percentage Control Method* determines the algorithm used for power control. Selecting *All inverters at the same percent* will ensure that generation is uniform (in relation to inverter capacity) but on larger plants with partial shading could lead to potential losses in while curtailing as the set points assume uniform generation. To solve this, choose *Individual percent calculated per inverter*. This will lead to non-uniform generation which could potentially lead to a large mismatch between inverter percentages (i.e. one at 100% and another at 0%)

The screenshot shows a configuration window with a single field:

- Readjust all inverters to the same percent interval (minutes)**  
5

Re-equalize interval in minutes

If you select individual inverter mode, you will need to fill in the *Readjust* field which will attempt to correct any imbalances every X minutes by adjusting all inverters to a similar percent again.

The *Ramp Mode* option chooses how power control adapts to changing set points. Some inverters may take longer to reach their set points and by changing them frequently improper loop control may happen. Select *Ramp to set point* to solve this, ensuring that if the set point is higher than the current generation no change will happen till the inverter

manages to reach within 10% of the current generation set point. The default setting is *No Ramp*.

The *Shared meter for EB/DG* option can be *Enabled* to use a common meter for both the grid incomer (Electricity Board) and the Generator (DG) for a more cost-effective setup. The digital input 1 (DI1) determines which source is active.

**Note:** This feature is currently only supported on the WattmonMEGA/MEGA2/MEGA2-4G and ULTRA, where the DI1 is used to indicate grid availability, thereby differentiating the grid input from the genset output.

*Use Digital inputs as bus coupler* only works when multi-cluster mode is enabled and allows dynamic joining of control groups depending on that status of the digital input – read more in the Advanced Power Control guide.

The *Track Curtailment* feature can display the curtailment loss against the actual generation on a site with Zero Feed-In and/or DG Protection, by tracking the actual generation and logging against the potential generation based on the quality of sunlight.

**Note:** This feature requires a pyranometer to be interfaced with the Wattmon and the Pyranometer package enabled from the Additional Options in the EZConfig Wizard.

*Fallback when TCP meter connection fails* allows you to fall back to a default percentage if a critical TCP meter link fails ensuring, i.e. if a genset meter is running over LAN and it becomes unreachable, you may want to ramp down solar to ensure that the genset does not trip due to backfeeding.

Fallback when TCP meter connection fails

Disabled

Enabled

If a remote Modbus TCP energy meter of a higher priority meter fails, you can choose to override the inverter percent until the TCP connection is restored

Duration that TCP link fails before ramping down: 30

Delay before ramping down in seconds: \_\_\_\_\_

Inverter override percent: 5

Interval percent to fallback to: \_\_\_\_\_

Enter the timeout in *Duration that TCP link fails* in seconds before which it falls back to the *Interval override percent* field.

## Power Meters

The *Power Meters* section allows you to add and configure the energy meters as selected in the *EZConfig Wizard*. To add a meter, select the *ADD* button and a row will be displayed with all columns editable. Each row has several columns – *Device*, *Device Model*, *Connected To*, *Min kW*, *Max kW*, *Priority*, *Group*, *PF*, *Plant Size* and *Options*. The *Device Model* column displays the make and model of the meter selected in the *Device* column.

### Power Meters

Number of power meters: 1

ADD

Device	Device Model	Connected To	Min kW	Max kW	Priority	Month kWh	PF	Plant Size	Status	Options
meter1 (ID: 3)	Secure Elite 440-445	Main Incomer	1	2	1	1250	No	28	Active	ACTION ▾

Click Action > Edit to switch to Edit mode to make any changes.

The *Meter* drop-down list contain each of the energy meters the Wattmon has been configured for.

Device	Device Model	Connected To	Min kW	Max kW	Priority ?	Month kWh ?	PF ?	Plant Size ?	Status ?	Options
1	Meter									ACTION ▾
meter1										
<b>Connected To</b>										
Zero feed in										
<small>Select the location and functionality of this meter</small>										
<b>Min Power (kW)</b>										
1										
<small>Minimum allowed power value</small>										
<b>Max Power (kW)</b>										
2										
<small>Maximum allowed power value</small>										
<b>Priority (Higher is more)</b>										
1										
<small>Higher value meters will be used when multiple meters are active at the same time</small>										
<b>Max kWh for Month</b>										
1250										
<small>Throttle only once month kWh values is reached - leave at 0 to throttle always</small>										
<b>PF Control</b>										
Disabled										
<small>Perform dynamic reactive power adjustment if the inverters support it.</small>										
<b>Plant Size (AC kW)</b>										
28										
<small>Enter the Solar Plant size controlled by this meter in kW</small>										
<b>Status</b>										
Active										
<small>Selecting disabled will still show the meter but it will not be used in the power control algorithm</small>										

The *Connected To* drop-down allows you to select the installed location of your meter from five options - *Undefined*, *Genset*, *Zero Feed-In*, *Inverter Output* and *Building Load*. Essentially, this is a repeat of the settings as entered in the *EZConfig Wizard* which utilizes it for the *EZDisplay*, while the locations selected here notify their role in the power control solution. In order to ensure a smooth operation, both entries have to be identical.

The *Min Power kW* column sets the minimum threshold for the AC Active Power value in kilowatts. This is used by the Wattmon when enforcing Zero Feed-In or DG Protection, ensuring the Active Power value from the meter never drops below this value. In other words, the Wattmon ramps the inverters down once the meter output values reach this set value.

**Note:** The function of this value differs based on the setting in the *Connected To* column. If configured for *Inverter Output* or *Building Load*, the value should always be 0. In the case of *Main Incomer*, the value should be set to a percentage of the site rating, to allow for a buffer in the case of a rapid ramping of power. For *Genset*, the *Min kW* should be set to the industry standard of 30% of the diesel generator rating.

The *Max Power kW* column sets the maximum threshold for the AC Active Power value in kilowatts. This is used by the Wattmon when enforcing Zero Feed-In or DG Protection, ensuring the Active Power value from the meter does not exceed this value.

The range between the *Min* and *Max* power should be sufficient to prevent unnecessary hysteresis which imbalance the power control loop. *EZConfig* will suggest appropriate values which can be tweaked if necessary.

The *Priority* column is used when more than a single meter is interfaced in the power control solution, setting a value based on the importance of the meter in that location, with *1*

being the lowest value and the greater number signifying a higher priority. The Wattmon follows this order when controlling the inverters.

The *Max kWh for Month* field lets you enter a number of kWh that you wish to export (without any throttling) after which the power control limits (Min and Max) will apply. Leave this at 0 to disable.

The *Group* option appears when you have a multi-cluster license. Select a group for the meter. Valid groups are 1,2,4,8,16,32,64 (groups are defined as a bitmask in inverters, and hence a unique bit position is required per group id)

The *PF Control* option lets you enable reactive power control. If selected, the *Reactive Threshold* box appears allowing you to enter a reactive set point in kW. The Reactive power of the meter will be used as a reference, and inverters will dynamically adjust their reactive power to be within +/- this threshold (i.e. either side of 0 reactive power) before re-adjusting the inverter reactive set points..

*Plant Size* - enter the total AC installed capacity for proper power control functionality

*Status* - Disable this to not use the meter in the power control algorithm. A check meter (solar output) may be disabled if you don't want to use the value for the power control algorithm, for example.

The *Options* column features the *ACTION* button, which allows you to either *Edit* or *Stop Editing* a particular device, and also *Delete* an added device.

## Inverters

The *Inverters* section lets you add and configure the inverters selected in the *EZConfig Wizard*. To add an inverter, select the *ADD* button and a row will be displayed with all columns editable. Each row has several columns - *Device*, *Device Model*, *Rating*, *Default Percent*, *Group*, *PF Control* and *Options*. Clicking the *Action > Edit* button brings the row into edit mode.

### Inverters

Number of inverters: 2

ADD

Device	Device Model	Rating (kW) ?	Default percent ?	Group ?	PF Control ?	Options
1 inverter1 (ID: 1) inverter1	Sungrow SGxxCX PV String Inverter	100 100	100 100	0 0	Active <input checked="" type="checkbox"/>	ACTION ▾
2 Pyranometer (ID: 200)	Pyranometer (Virtual)	100	100	0	Active	ACTION ▾

The *Device* drop-down list contains the inverters already defined in EZConfig

The *Device Model* column displays the make and model of the inverter selected in the *Device* column.

The *Rating* column should be set to the actual rating of the inverter. For example, a 100kW rated inverter with 110% max power is actually a 110kW inverter - which needs to be entered as 110 accordingly.

**Note:** It is particularly important to set the correct rating of each inverter to ensure optimal performance from the system. Entering an incorrect rating value, for example entering 30 for 50 kW inverter, will cause the Wattmon to treat the inverter as 30 kW unit, setting the 100% power output at 30 kW.

The *Default Percent* column is fail-safe built into the Wattmon power control solution. When no active power sources are found, Wattmon sets the active power output percentage of the inverters to this value.

The *Group* field lets you set a group mask. This can be a single group (i.e. 2 or 4), or it could be a combination of multiple groups (i.e. 6) which will impact how the inverters behave depending on the energy sources active and their priorities. This explained in the *Advanced Power Control* guide.

*PF Control* can be enabled or disabled (if your key supports reactive power control) to enable reactive power generation on the inverter.

The *ACTION* button allows you to either *Edit* or *Stop Editing* a particular device, and also *Delete* an added device.

**Note:** Power control will *still control* inverters and set them to the *Default Percent* when power control is **disabled**. In order to not send commands to the inverters, you need to **uninstall** the package

Upon completion of all sections, you may click on the *APPLY CHANGES* button on the top of the page and then refresh the page to verify all the settings have been saved. To set the Wattmon to follow these settings, *Clear Cache* and *Reboot* the device.

## FTP (File Transfer Protocol) Package

File Transfer Protocol or FTP is one of the means to push the logged data from your Wattmon device to a third-party server. Follow the instructions on installing packages, then go to *Control Panel > FTP* to open the settings page.

**FTP Export**

HELP   ✕ CLOSE   ✓ APPLY CHANGES

### Admin User Settings

Status

Disabled  
 Enabled

Status of the FTP uploading package - Disable to stop updating

FTP Server Path  
/web/ftptest/  
Enter path on FTP server with / at end

FTP Server Hostname  
wattmon.com  
FTP Server Hostname

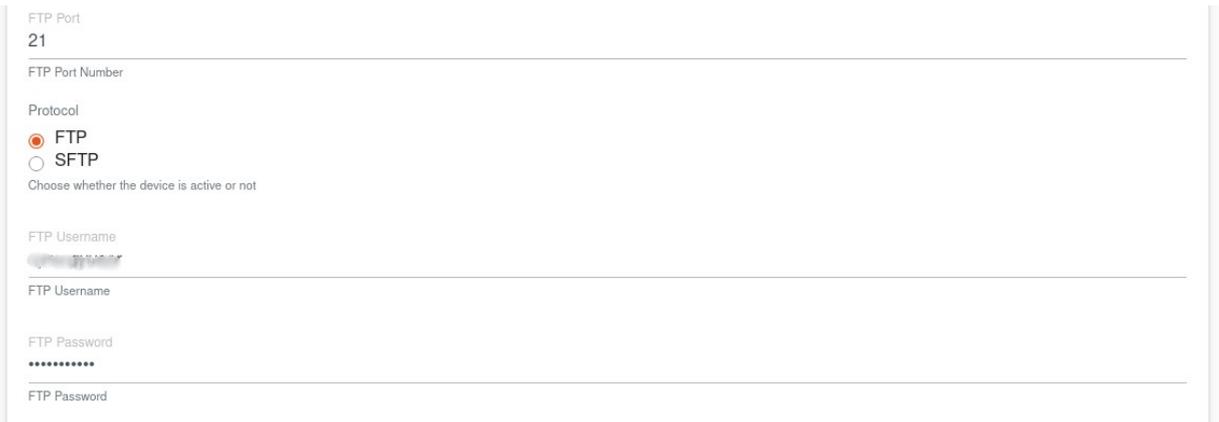
FTP Port  
21  
FTP Port Number

Protocol

The *Status* option should be set to *Enabled* to activate FTP.

Enter the *FTP Server path* to upload files to with a / at the end.

In the *FTP Server Hostname* field enter the IP Address or host name of the FTP server.



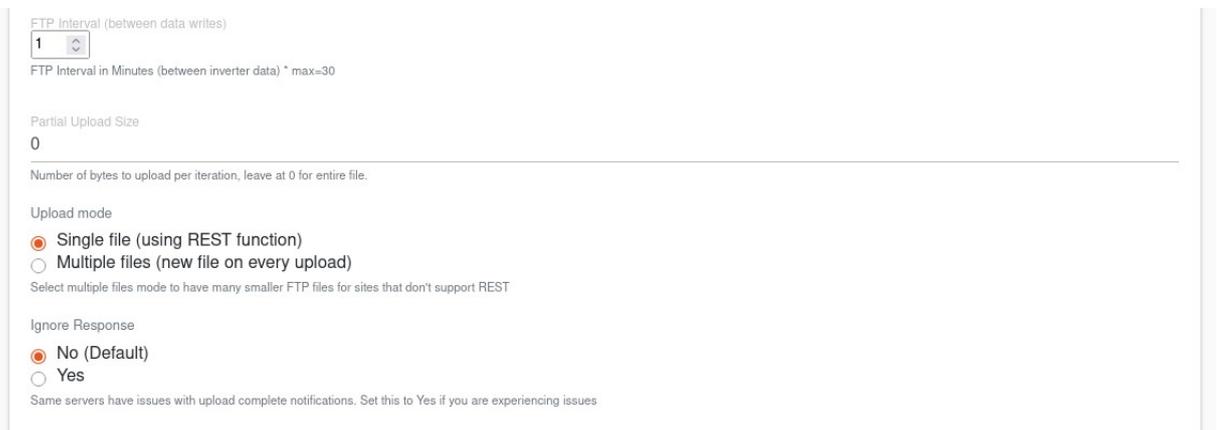
The screenshot shows a configuration form with the following fields and options:

- FTP Port:** 21
- FTP Port Number:** (empty text input field)
- Protocol:** Radio buttons for  **FTP** and  **SFTP**. Below it is the text: "Choose whether the device is active or not".
- FTP Username:** (empty text input field)
- FTP Password:** (password input field with masked characters)
- FTP Password:** (empty text input field)

Choose the *FTP port* (21 is the default)

Select the protocol - *FTP* or *SFTP* (using TLS encryption)

In the *FTP Username* field enter the username to connect, and in the *FTP Password* field enter the FTP password.



The screenshot shows a configuration form with the following fields and options:

- FTP Interval (between data writes):** A spinner box set to 1. Below it is the text: "FTP Interval in Minutes (between inverter data) \* max=30".
- Partial Upload Size:** 0
- Number of bytes to upload per iteration, leave at 0 for entire file.** (empty text input field)
- Upload mode:** Radio buttons for  **Single file (using REST function)** and  **Multiple files (new file on every upload)**. Below it is the text: "Select multiple files mode to have many smaller FTP files for sites that don't support REST".
- Ignore Response:** Radio buttons for  **No (Default)** and  **Yes**. Below it is the text: "Same servers have issues with upload complete notifications. Set this to Yes if you are experiencing issues".

In the *FTP Interval* field choose how often (in minutes) the data is synced to the FTP server.

For very large files you can split the upload so that it does not block the connection unnecessarily, by entering a value in byte in the *Partial upload size* field. Leave this at 0 to always sync the full file (or pending amount)

*Upload Mode* determine how the file gets places on the server. Use *Single File* to append to an existing file - there will be a single CSV file per day. Use *Multiple files* to upload different files each time with a few lines each. For this, enter an optional *File Prefix* below (for example CYN\_).

Set *Ignore Response* to *No*. In some rare cases the server may not reply with a proper message that the file has been uploaded. In this case select *Yes*, to close the connection automatically once the file has been uploaded.

If everything is set up properly you should start to receive data on your server. If not, check the system log file to troubleshoot the issue.

## MQTT2 (Message Queuing Telemetry Transport ) Package

Wattmon support up to 2 MQTT Servers. The Message Queuing Telemetry Transport protocol or MQTT is one of the means to push the logged data from your Wattmon device to a third-party server. Follow the instructions on installing packages, then go to *Control Panel > MQTT2* to open the settings page.

Global MQTT Options

Debug Mode  Disabled  
 Enabled  
MQTT debug mode (to log file)

Dynamic Buffer Sizing  Disabled  
 Enabled  
Resize the TX buffer dynamically to accommodate larger packets

Maximum TX Packet Size 8000  
Max TX/RX Packet Size in bytes

Enable Debug mode to get more log entries that indicate the status of the MQTT engine - you can enable this to troubleshoot issues.

Enable Dynamic Buffer sizing if you plan to send large packets of data but do not know the size beforehand.

Maximum TX Packet size denotes the maximum allowed size of a packet - this can be set to a smaller number if Dynamic buffer sizing is enabled. On Wattmon devices with lower memory availability it is recommended to keep the size as small as possible, but larger than the biggest packet. If packets stop being sent properly, check the log to see if it says Packet Dropped indicating that it is too large.

The next two sections let you configure each of the MQTT server connections.

MQTT Server 1

MQTT Status  Disabled  
 Enabled  
MQTT engine status

MQTT Server [redacted]  
MQTT Server host name

MQTT Port 1883  
Enter Server Port number

Use TLS Encryption  Disabled  
 Enabled  
TLS Encryption status

MQTT Username [redacted]  
MQTT Username - leave blank if not required

MQTT Password \*\*\*\*\*  
MQTT Password - leave blank if not required

Under MQTT Status, select *Enabled* to enable the server.

Enter the MQTT server name in the *MQTT Server* field. Enter the port (the default is 1883) in the *MQTT Port* field.

If using encryption, select *Enabled* in the *Use TLS Encryption* field.

For connections that require a username/password, enter the values in the *MQTT Username* and *MQTT Password* fields respectively.

You can choose the *QoS* (Quality of service) level between QoS 0 and QoS 1 in the MQTT QoS field. QoS 0 is the most basic and does not have guaranteed delivery of messages, whereas QoS 1 will ensure that published messages reach their destination at least once.

Set *MQTT Mode* to *User Mode* to manage communication manually or through the options further down. Select *App Mode* to switch to app mode that follows a predefined set of commands to communicate with a remote device - this can be used to sync data between two devices automatically.

The image shows a configuration form for MQTT settings. It includes the following fields and options:

- MQTT Connect ID:** Set to `wmtest_EB`. A note below states: "MQTT Connect ID - needs to be unique per MQTT server".
- MQTT Protocol:** Two radio buttons are present:  **MQTT 3** and  **MQTT 3.1**. A note below says: "Select the MQTT Protocol you wish to use".
- Timeout:** Set to `30`. A note below states: "MQTT Command timeout in seconds".
- Keepalive:** Set to `60`. A note below states: "MQTT Keepalive time (to send PINGs)".

Some servers will give you a unique connection ID, otherwise you could enter the MAC of the device in the *MQTT Connect ID* to ensure that it is unique. Many MQTT brokers will disconnect you if another connection with the same connect ID is active.

Select the *MQTT Protocol* as 3 or 3.1 as required by your server.

The *Timeout* field determines how long Wattmon waits before timing out when trying to connect. The default is 30 seconds.

The *Keepalive* interval can be set to ensure an MQTT ping is sent periodically to avoid unnecessary disconnection by the server. The default is 60 seconds.

## Automatic Publishing

The next part lets you choose to automatically publish device data if required. This allows you to synchronize data between devices automatically provided they have the same driver on the same modbus ID.

The image shows a table titled "MQTT Automatic Device Data Publishing" with an "ADD DEVICE" button in the top right corner. The table has four columns: Device ID, Channel, Interval, and Options. There is one row of data:

Device ID	Channel	Interval	Options
inverter1	/wmtest_EB/app2wm/device/tx/1	5	ACTION ▾

Click the *Add Device* button to add a row. Select any device already configured in Wattmon in the *Device ID* field, and select the channel (topic) to publish to. The name is automatically generated based on the Client ID, in the example above the Client ID was set at `wmtest_EB`, and the channel became `/wmtest_EB/app2wm/device/tx/1` as `inverter1` is on Modbus ID 1. Ensure that the Wattmon listening for updates has an inverter of the same type on ID 1 (and a virtual bus) for this to work.

The *Interval* is the number of seconds between data pushes.

## Subscribing to Topics

Subscribing to channels (topics) is done automatically for channels in the list.

MQTT Subscribed Channels			ADD CHANNEL
Channel	Callback	Options	
/wmttest_EB/app2wm/#	Automatically Handled in Firmware		

In *App Mode*, the channel name is automatically generated based on the client ID.

In *User Mode* you will need to create a script that is called when a packet comes in on a topic. In the *Channel* field enter the topic with wildcards such as # to listen for packets and create a uPHP script to be executed for each packet received. Enter the path to the file in the *Callback* field.

MQTT Subscribed Channels			ADD CHANNEL
Channel	Callback	Options	
 /my_channel/#	/scripts/tests/my_callback.cgi	ACTION ▾	

Using a combination of the above it is simple to set up a system whereby one Wattmon connected to say an energy meter can publish meter values periodically to a topic and these can be then updated in multiple Wattmons in remote locations that require that value for power control, for example.

*MQTT Server 2* has identical options to the above, allowing you to set up a flexible configuration that communicates with different servers for different purposes.

## Alerts Package

The *Alerts* package handles various types of notifications and can be used to get updates when certain events happen on the Wattmon.

### Alerts

HELP    ✕ CLOSE    ✓ APPLY CHANGES

#### Notification Settings

Send Alerts via Wattmon EMS Cloud

Disabled  
 Enabled

Include alerts while exporting data - processing will be as per the EMS server notification settings

Send SMS Alerts

Disabled  
 Enabled

Send an SMS when an event occurs

Send Email Alerts

Disabled  
 Enabled

Send an Email when an event occurs

Send Alerts via EMS can be selected to ensure that alerts are published when exporting data to the EMS cloud portal, which is useful if you have a cloud subscription. This will allow you to receive notifications on your WhatsApp number or via email if you have configured the settings on the EMS Cloud.

*Send SMS Alerts* can be enabled if your device has a SIM card. Enter the *SMS Phone Number* of the receiver.

*Send Email Alerts* can be activated to send emails out – this requires the the Email Settings be properly configured beforehand – refer to the appropriate section in the manual for instructions on that. Enter the *Email address* of the recipient for the alerts.

If MQTT alerts are required, enable it in *Send MQTT Alerts*. Select the channel (topic) to publish to in the MQTT Channel field, and choose which of the 2 servers to use in the MQTT Server field.

The *MQTT Format* script can be left blank in which case the raw message text will be published. Alternatively you can refer to the samples in the */packages/alerts* folder to if you wish to customize your alert message format.

The *Min Alert interval* determines the minimum delay in minutes before the next alert can be sent. The default value is 1 minute.

The table below explains the alert types that the package can process and can be enabled in their respective fields in the Event Setting section.

<i>Log when inverter offline</i>	Number of minutes to wait before offline alert for device
<i>Alert on Grid Failure</i>	Send alert whenever grid meter goes off
<i>Alert when Curtailing Inverter output</i>	Send alert whenever an inverter percent goes less than 100
<i>Alert when stopping inverter curtailment</i>	Send alert whenever all inverters go back to 100%
<i>Alert on Grid Power Returns</i>	Send alert when grid meter powers up again
<i>Alert when Genset Switched On</i>	Send alert when the genset meter powers up
<i>Alert when digital input is turned on</i>	Send an alert when any of the 4 digital inputs on Ultra, Mega2, Mega2-4G are turned on

## Pyranometer Package

In order to use the Wattmon to automatically calculate performance and potential generation values using an irradiation sensor the Pyranometer package is required. This is installed automatically when EZConfig is run with the appropriate options, but you may need to manually link the irradiance role and set the plant size or PR. Access the settings through Control Panel > Pyranometer.

**Pyranometer / Irradiation Sensor**

HELP   ✕ CLOSE   ✓ APPLY CHANGES

### Sensor Settings

Scale Factor  
1

The measured voltage will be multiplied by this to obtain the watt value

Role  
device1\_Active\_Irradiance

Select an optional role to use as a voltage source or leave as undefined to use the default A3 input.

Plant Size  
25.809999

Enter the plant size in kW

Static PR Ratio  
0.85

Enter the performance ratio as a fraction (i.e. 0.8)

The *Scale Factor* should be left at 1 unless you are using an analog input of a legacy device.

The *Role* has a drop down of all the data points that are logged - select the device data point that has the solar irradiance value (directly from the actual pyranometer).

Enter the *Plant Size* as per the actual DC installed capacity.

The *Static PR* ratio is used to calculate the potential generation and curtailment losses. The default value is 0.8 but you will need to adjust this manually if you see a mismatch between the actual generation and the potential generation - this can be seen on the main Dashboard in EZDisplay.

## Appendix 1: Wattmon Naming Conventions

The following table shows the Wattmon naming conventions which need to be used when creating device drivers. Map the name to the closes matching name in the datasheet from the manufacturer.

Parameter Name	Units	Description
ModelNumber	N/A	Device Model Number
SerialNumber	N/A	Device Serial Number
AC_Current	Amps	Total AC Current
AC_Current_A	Amps	AC Current Phase A
AC_Current_B	Amps	AC Current Phase B
AC_Current_C	Amps	AC Current Phase C
AC_Current_THD	%	Total AC Current - Total Harmonic Distortion
AC_Current_A_THD	%	AC Current Phase A - Total Harmonic Distortion
AC_Current_B_THD	%	AC Current Phase B - Total Harmonic Distortion
AC_Current_C_THD	%	AC Current Phase C - Total Harmonic Distortion
AC_Voltage	Volts	Total AC Voltage
AC_Voltage_AB	Volts	AC Voltage Phase A to Phase B
AC_Voltage_BC	Volts	AC Voltage Phase B to Phase C
AC_Voltage_CA	Volts	AC Voltage Phase C to Phase A
AC_Voltage_AN	Volts	AC Voltage Phase A to Neutral
AC_Voltage_BN	Volts	AC Voltage Phase B to Neutral
AC_Voltage_CN	Volts	AC Voltage Phase C to Neutral
AC_Voltage_THD	%	Total AC Voltage - Total Harmonic Distortion
AC_Voltage_AN_THD	%	AC Voltage Phase A to Neutral - Total Harmonic Distortion
AC_Voltage_BN_THD	%	AC Voltage Phase B to Neutral - Total Harmonic Distortion
AC_Voltage_CN_THD	%	AC Voltage Phase C to Neutral - Total Harmonic Distortion
AC_Active_Power	Watts	Total AC Active Power
AC_Active_Power_A	Watts	AC Active Power Phase A
AC_Active_Power_B	Watts	AC Active Power Phase B
AC_Active_Power_C	Watts	AC Active Power Phase C
AC_Active_Power_A_THD	%	AC Active Power Phase A - Total Harmonic Distortion
AC_Active_Power_B_THD	%	AC Active Power Phase B - Total Harmonic Distortion
AC_Active_Power_C_THD	%	AC Active Power Phase C - Total Harmonic Distortion
AC_Frequency	Hertz	Total AC Frequency
AC_Frequency_A	Hertz	AC Frequency Phase A
AC_Frequency_B	Hertz	AC Frequency Phase B
AC_Frequency_C	Hertz	AC Frequency Phase C
AC_Reactive_Power	VAR	Total AC Reactive Power
AC_Reactive_Power_A	VAR	AC Reactive Power Phase A
AC_Reactive_Power_B	VAR	AC Reactive Power Phase B
AC_Reactive_Power_C	VAR	AC Reactive Power Phase C
AC_Apparent_Power	Volt-Amps	Total AC Apparent Power
AC_Apparent_Power_A	Volt-Amps	AC Apparent Power Phase A
AC_Apparent_Power_B	Volt-Amps	AC Apparent Power Phase B
AC_Apparent_Power_C	Volt-Amps	AC Apparent Power Phase C
AC_PF	%	Total AC Power Factor
AC_PF_A	%	AC Power Factor Phase A

AC_PF_B	%	AC Power Factor Phase B
AC_PF_C	%	AC Power Factor Phase C
kWh_Total_Active	kWh	Lifetime Generation / Total Active Energy
kWh_Day_Active	kWh	Daily Generation
kVAh_Total_Active	kVAh	Total Apparent Energy
kVARh_Total_Active	kVARh	Total Reactive Energy
DC_Current	Amps	Total DC Current
DC_Current_1	Amps	DC Current PV 1
DC_Current_2	Amps	DC Current PV 2
DC_Current_3	Amps	DC Current PV 3
DC_Current_4	Amps	DC Current PV 4
DC_Current_5	Amps	DC Current PV 5
DC_Current_6	Amps	DC Current PV 6
DC_Current_7	Amps	DC Current PV 7
DC_Current_8	Amps	DC Current PV 8
DC_Current_9	Amps	DC Current PV 9
DC_Current_10	Amps	DC Current PV 10
DC_Current_11	Amps	DC Current PV 11
DC_Current_12	Amps	DC Current PV 12
DC_Voltage	Volts	Total DC Voltage
DC_Voltage_1	Volts	DC Voltage PV 1
DC_Voltage_2	Volts	DC Voltage PV 2
DC_Voltage_3	Volts	DC Voltage PV 3
DC_Voltage_4	Volts	DC Voltage PV 4
DC_Voltage_5	Volts	DC Voltage PV 5
DC_Voltage_6	Volts	DC Voltage PV 6
DC_Voltage_7	Volts	DC Voltage PV 7
DC_Voltage_8	Volts	DC Voltage PV 8
DC_Voltage_9	Volts	DC Voltage PV 9
DC_Voltage_10	Volts	DC Voltage PV 10
DC_Voltage_11	Volts	DC Voltage PV 11
DC_Voltage_12	Volts	DC Voltage PV 12
DC_Power	Watts	Total DC Power
DC_Power_1	Watts	DC Power PV 1
DC_Power_2	Watts	DC Power PV 2
DC_Power_3	Watts	DC Power PV 3
DC_Power_4	Watts	DC Power PV 4
DC_Power_5	Watts	DC Power PV 5
DC_Power_6	Watts	DC Power PV 6
DC_Power_7	Watts	DC Power PV 7
DC_Power_8	Watts	DC Power PV 8
Temperature_Internal	°C	Inverter Temperature
Current_String_1	Amps	String 1 Current
Current_String_2	Amps	String 2 Current
Current_String_3	Amps	String 3 Current
Current_String_4	Amps	String 4 Current
Current_String_5	Amps	String 5 Current
Current_String_6	Amps	String 6 Current
Current_String_7	Amps	String 7 Current
Current_String_8	Amps	String 8 Current
Current_String_9	Amps	String 9 Current
Current_String_10	Amps	String 10 Current
Current_String_11	Amps	String 11 Current
Current_String_12	Amps	String 12 Current

Current_String_13	Amps	String 13 Current
Current_String_14	Amps	String 14 Current
Current_String_15	Amps	String 15 Current
Current_String_16	Amps	String 16 Current
Current_String_17	Amps	String 17 Current
Current_String_18	Amps	String 18 Current
Current_String_19	Amps	String 19 Current
Current_String_20	Amps	String 20 Current
Current_String_21	Amps	String 21 Current
Current_String_22	Amps	String 22 Current
Current_String_23	Amps	String 23 Current
Current_String_24	Amps	String 24 Current
percent	%	Active Power Output Percent
Status_Code	N/A	Vendor-defined Operating State
Event_Code	N/A	Vendor-defined Events and Errors
AC_Input_Current	Amps	Total AC Input Current
AC_Input_Current_A	Amps	AC Input Current Phase A
AC_Input_Current_B	Amps	AC Input Current Phase B
AC_Input_Current_C	Amps	AC Input Current Phase C
AC_Output_Current	Amps	Total AC Output Current
AC_Output_Current_A	Amps	AC Output Current Phase A
AC_Output_Current_B	Amps	AC Output Current Phase B
AC_Output_Current_C	Amps	AC Output Current Phase C
AC_Input_Voltage	Volts	Total AC Input Voltage
AC_Input_Voltage_AN	Volts	AC Input Voltage Phase A to Neutral
AC_Input_Voltage_BN	Volts	AC Input Voltage Phase B to Neutral
AC_Input_Voltage_CN	Volts	AC Input Voltage Phase C to Neutral
AC_Output_Voltage	Volts	Total AC Output Voltage
AC_Output_Voltage_AN	Volts	AC Output Voltage Phase A to Neutral
AC_Output_Voltage_BN	Volts	AC Output Voltage Phase B to Neutral
AC_Output_Voltage_CN	Volts	AC Output Voltage Phase C to Neutral
AC_Input_Active_Power	Watts	Total AC Input Active Power
AC_Input_Active_Power_A	Watts	AC Input Active Power Phase A
AC_Input_Active_Power_B	Watts	AC Input Active Power Phase B
AC_Input_Active_Power_C	Watts	AC Input Active Power Phase C
AC_Output_Active_Power	Watts	Total AC Output Active Power
AC_Output_Active_Power_A	Watts	AC Output Active Power Phase A
AC_Output_Active_Power_B	Watts	AC Output Active Power Phase B
AC_Output_Active_Power_C	Watts	AC Output Active Power Phase C
AC_Input_Frequency	Hertz	Total AC Input Frequency
AC_Input_Frequency_A	Hertz	AC Input Frequency Phase A
AC_Input_Frequency_B	Hertz	AC Input Frequency Phase B
AC_Input_Frequency_C	Hertz	AC Input Frequency Phase C
AC_Output_Frequency	Hertz	Total AC Output Frequency
AC_Output_Frequency_A	Hertz	AC Output Frequency Phase A
AC_Output_Frequency_B	Hertz	AC Output Frequency Phase B
AC_Output_Frequency_C	Hertz	AC Output Frequency Phase C
Battery_Voltage	Volts	Battery Voltage
Battery_Current	Amps	Battery Current
Battery_Power	N/A	Battery Power
Battery_Temperature	°C	Battery Temperature
SoC	N/A	State of Charge
Charge_State	%	Charge State
Charge_Power_Day	Watts	Daily Maximum Charge Power

Charge_State	N/A	Charge State
kWh_Total_Input	kWh	Total Input Energy
kWh_Day_Input	kWh	Daily Input Energy
kWh_Total_Output	kWh	Total Output Energy
kWh_Day_Output	kWh	Daily Output Energy
kWh_Battery_Charge	kWh	Total Battery Charge Energy
kWh_Battery_Discharge	kWh	Total Battery Discharge Energy
kWh_Total_Export	kWh	Total Export Energy
kWh_Total_Import	kWh	Total Import Energy
kWh_Day_Export	kWh	Daily Export Energy
kWh_Day_Import	kWh	Daily Import Energy
kVAh_Total_Export	kVAh	Total Export Apparent Energy
kVAh_Total_Import	kVAh	Total Import Apparent Energy
kVARh_Total_Export	kVARh	Total Export Reactive Energy
kVARh_Total_Import	kVARh	Total Import Reactive Energy
kVARh_Lag	kVARh	Total Reactive Energy - Lag
kVARh_Lead	kVARh	Total Reactive Energy - Lead
kWh_Total_Utility	kWh	Total Utility Energy
kWh_Total_Genset	kWh	Total Generator Energy
AC_Current_MaxDemand	Amps	Total AC Current - Maximum Demand
W_Max_Demand	Watts	Total AC Active Power - Maximum Demand
VAR_Max_Demand	VAR	Total AC Reactive Power - Maximum Demand
VA_Max_Demand	Volt-Amps	Total AC Apparent Power - Maximum Demand
VA_Max_Demand	Volt-Amps	Total AC Apparent Power - Maximum Demand
VA_Max_Demand	Volt-Amps	Total AC Apparent Power - Maximum Demand
Output_Current	Amps	Output Current
Output_Voltage	Volts	Output Voltage
Output_Power	Watts	Output Power
Output_Frequency	Hertz	Output Frequency
Motor_Speed	RPM	Motor / Pump Running Speed
Flow_Rate	l/hr	Flow Rate of Pump
Day_Flow	Litres	Day Flow
Power_On_Time	Hours	Current Power-Up Time
Running_Time	Hours	Current Running Time
Total_Power_On_Time	Hours	Total Power-Up Time
Total_Running_Time	Hours	Total Running Time
Active_Irradiance	w/m2	Solar Irradiance
Active_Irradiance_Uncor- rected	w/m2	Solar Irradiance without Temperature Correction
kWh_Day_Irradiance	Kilowatts	Daily Irradiance
Cell_Temperature	°C / °K	Solar Irradiance Cell Temperature
Wind_Speed	km/h	Wind Speed
Wind_Direction	°	Wind Direction
Ambient_Temperature	°C	Ambient Temperature
Module_Temperature	°C	Module Temperature
Relative_Humidity	%	Relative Humidity
Rainfall	mm	Daily Rainfall
Atmospheric_Pressure	mbar	Atmospheric Pressure
kW_Potential_Active	Kilowatts	Potential Generation
kWh_Curtailment_Lifetime	kWh	Total kWh Lost Due to Curtailment
kWh_Curtailment_Day	kWh	Total kWh Lost Due to Curtailment

kWh_Curtailment_DG_Day	kWh	Total kWh Lost Due To DG Curtailment
kWh_Curtailment_Grid_Day	kWh	Total kWh Lost Due To DG Curtailment
kW_Active_Generation	Kilowatts	Total kWh Lost Due To DG Curtailment
Live_PR	%	Live Performance Ratio Value
Daily_PR	%	Daily Performance Ratio Value

The following table lists the preferred role ordering when creating device drivers.

<b>Role Index</b>	<b>Role Name</b>	<b>Alt. Role Name</b>
<b>0</b>	<b>AC_Active_Power</b>	
<b>1</b>	<b>AC_Reactive_Power</b>	<b>DC_Power</b>
<b>2</b>	<b>AC_Apparent_Power</b>	<b>AC_PF</b>
<b>3</b>	<b>kWh_Total_Active</b>	<b>kWh_Total_Export</b>
<b>4</b>	<b>kWh_Day_Active</b>	<b>kWh_Total_Import</b>
<b>5</b>	<i>As per Register Address Order</i>	<i>As per Register Address Order</i>
<b>6</b>	<i>As per Register Address Order</i>	<i>As per Register Address Order</i>
<b>7</b>	<i>As per Register Address Order</i>	<i>As per Register Address Order</i>

# Troubleshooting your Wattmon

## Wattmon does not Work

There are several things which can cause an issue in the functioning of your Wattmon. We look at the common ones below:

- a) Check that your Wattmon has power. If the *Power / PWR* light on the Wattmon is switched on and glowing, the power supply is fine. If not, use a multimeter to measure the voltage between pin 1 and 2 of your Wattmon. If you do not get a voltage, identify, and correct the cabling or change the adapter. If you do get a voltage greater than 10V DC at the input, then your Wattmon adapter may not be working
- b) Make sure the supplied microSD card is properly inserted in the slot. Without a properly configured microSD card, your Wattmon will not work (unless you are running from Flash on the MEGA2/MEGA2-4G or ULTRA). If you have lost the card or if it is damaged, download a WattmonOS .zip file from the website and unzip it onto a blank microSD card of at least 8 GB capacity. Insert the new card into the microSD slot of the Wattmon before powering up.
- c) Perform a firmware update on the Wattmon by downloading the latest Firmware for the from [www.wattmon.com](http://www.wattmon.com) for your hardware version. Extract the .tar.gz file onto your PC. Copy the extracted HEX file (*imagez.hex* or *imageu.hex*) into the root folder of the microSD card and insert it into the Wattmon. Press and hold the reset button and then power up the Wattmon to re-flash your device. You should notice the *Script / SCR* light flash rapidly for a few seconds, then slower for a few seconds, and then finally stop. Once it resumes blinking once per second, the firmware update has been successfully completed.
- d) If the *Device / DEV* and *Script / SCR* lights are flashing simultaneously and rapidly, the microSD card is not inserted properly or is damaged. To solve this, go to steps (b) and (c).
- e) If the *Script / SCR* light is not blinking approximately once per second, the most likely issue is software related. To be sure it is a software or configuration issue, follow steps (b) and (c) above and see if it resolves the issue.
- f) Connect the Wattmon to your laptop via a LAN cable. If you see the green LAN light solid and orange LAN light on the RJ45 jack blinking, the LAN connection is proper. To connect to the device, follow the instructions on the **Connecting to your Device** page.

## Unable to connect to the Wattmon over Ethernet

The procedure to connect to your Wattmon is very straightforward. It is described on the **Connecting to your Device** page in various steps. Follow the settings and your Wattmon will be easily accessible over an Ethernet connection.

## Frequently Asked Questions

- *What is the default IP address of my Wattmon? How do I reset my Wattmon to the default IP?*

The default IP address of your Wattmon is *192.168.0.55*. It is printed on the label on the right side of the device. In case you forget the IP address you have assigned to your Wattmon, you can return it to the default by pressing down the reset button on the Wattmon for five seconds, till the *Scripts* light stops blinking for a few seconds and then resumes blinking. Switch your Wattmon off and on, and it will return it to the default IP. If your Wattmon is in DHCP mode you can toggle it back to static IP mode by holding the reset button for approximately 10 seconds. Alternatively, if you are logged into your Wattmon, you can reset its IP address by selecting *Defaults* from the **Network Setup** page.
- *What is a MAC address? Where do I find the MAC of my Wattmon?*

A media access control address (MAC address) of a computer is a 12 digit alpha-numeric unique identifier assigned to network interfaces for communications at the data link layer of a network segment. MAC addresses are used as a network address for most IEEE 802 network technologies, including Ethernet and Wi-Fi.

The MAC of your Wattmon can be found printed on the right side of the device along with the serial number. Alternatively, it can be found on the **Network Setup** page on your Wattmon.
- *What are the default log-in credentials?*

The default username and password are *admin* and *admin* respectively.
- *How do I check the Wattmon OS and Firmware version on my Wattmon?*

Log-in to your Wattmon, and at the top right corner of each page, you will find the OS and Firmware version of your device. You can see the OS and Firmware version on the bottom right corner on the login page as well.
- *How do I reflash the Wattmon Firmware?*

Download the latest Firmware for your hardware from [wattmon.com](http://wattmon.com) under the Download section. Extract the .tar.gz file onto your PC. Copy the extracted file (*imagez.hex* or *imageu.hex*) into the root folder of the microSD card and insert it into the Wattmon. Press and hold the reset button and then power up the Wattmon to re-flash your device. You should notice the *Script / SCR* light flash rapidly for a few seconds, then slower for a few seconds, and then finally stop. Once it resumes blinking once per second, the firmware update has been successfully completed.
- *How do I update the Wattmon OS?*

If you are logged into your Wattmon device, download and install the latest OS from the **Package Manager** page. If you need to install the OS on a fresh microSD card, you may obtain a compressed version from [wattmon.com](http://wattmon.com) and then extract the contents in the root of the microSD card.
- *How do I configure my Wattmon for use with multiple inverters and/or energy meters?*

You can follow the guidelines in the *Modbus Device Setup* depending on the model of your Wattmon. Use the **EZConfig** tool, or the follow the instructions as in **Third-Party Devices**.
- *I have completed the configuration on my device and have connected the devices I wish to interface with the Wattmon, but I still do not see any data. How can I test my Modbus connection?*

To test the various Modbus devices connected to your Wattmon, you can use the built-in **Modbus Tester**, which allows you to poll each of your devices regardless of their settings, as long as they are physically connected (for RS-485) or virtually connected (Modbus TCP) to the Wattmon.



© Cynergy, 2025

**Standard Compliance**

*Wattmon MEGA, Wattmon MEGA2, Wattmon MEGA2-4G, Wattmon ULTRA, Wattmon MINI3 and Wattmon MINI3-4G are classified as devices for Industrial use.*

*The devices comply with the following standards:*

**IEC-61000-4-2** – Electrostatic Discharge

**IEC-61000-4-4** – Electrical Fast Transient

**IEC 61000-4-5** – Surge Immunity

**CISPR-22 Class A** – Emissions

**CISPR-32 Class A** – Emissions

**CISPR-11 Class A** – Emissions (WattmonPRO)

