

## Fracsun Modbus Documentation

February 16, 2024

Revision J

### 1.0 Introduction

This document is intended to provide the necessary information to connect the Fracsun soiling solution to an onsite SCADA system or datalogger via Modbus RS-485 protocol.

#### 1.1 ARES and Wash Extension

The Fracsun soiling monitoring solution often encompasses two separate devices: ARES and the Wash Extension. Both devices are configurable as Modbus Slaves to transfer data into the onsite SCADA system via a Modbus RTU RS-485 connection.



#### 1.2 ARES Cellular-only and Modbus-only modes

In the default state, ARES is an IoT cellular-enabled device that automatically connects to the nearest cellular base station and outputs data to the Fracsun cloud. The customer can view the data on Fracsun Dashboard. The Fracsun Cloud also performs additional data processing. For example, the *daily* soiling loss and insolation values are calculated (filtered and weighted) as a cloud process.

ARES can also operate in Modbus-only mode for wired connections to dataloggers and SCADA systems, which is most likely the preferred connection method for customers reading this document. The cellular modem is turned off in this mode, which stops the flow of data to the Fracsun Cloud. This also means that the daily calculated data (daily soiling loss, insolation, and Wash Analysis) can no longer occur in Modbus-only mode because the Fracsun cloud process cannot run without the data.

In either mode, Fracsun has options to transfer soiling sensor data between the Fracsun Cloud and the data acquisition system (DAS). Please reach out for more information or read our API Documentation.

## 2.0 Modbus Information

### 2.1 Modbus commands

These commands are all according to the Modbus RTU protocols described in: 'Modbus® over serial line V1.02' and 'MODBUS application protocol V1.1b' available from the Modbus® organization ([www.modbus.org](http://www.modbus.org)). These commands can be tested using software tools, such as 'Modbus Poll' from [www.modbustools.com](http://www.modbustools.com).

The following commands are implemented:

Function	Description
0x03	Read Holding Register (Single or Multiple)
0x06	Write Single Register
0x16	Write Multiple Registers

### 2.2 RS-485 Configurations

The following slave addresses, baud rates and frame configurations are available on both ARES and Wash Extension devices:

Property	Acceptable Values
Slave Address	1 – 255 ARES: 42 (default) Wash Extension: 43 (default)
Baud Rate	1200, 2400, 4800, 9600, 14400, 19200 (default), and 38400.
Frame Configuration	8 data bits, Even parity, 1 stop bit (default) 8 data bits, None parity, 1 stop bit

### 2.3 Key Modbus Notes

- The address field uses the Protocol address with base 0, not the PLC Address with base 1.
- Treat all data types using Big-Endian format.
- The **Modbus data type** column refers to the data format of the Modbus register(s). In most Fracsun registers, that will be a unsigned 16-bit integer, signed 16-bit integer, or 32-bit unsigned integer. For example, if the Wash Power parameter reads 7521 as an unsigned 16-bit integer, after the scalar (100) is accounted for, the **final data type** would be a FLOAT value of 75.21 ( $7521 / 100 = 75.21$ ).

## 3.0 Wiring

### 3.1 Wiring to a RS-485 Master Device

In its simplest form, ARES and Wash Extension devices are connected in parallel to the RS-485 master device. Internal 120Ω termination resistors are already installed in Fracsun devices, therefore are not required externally.

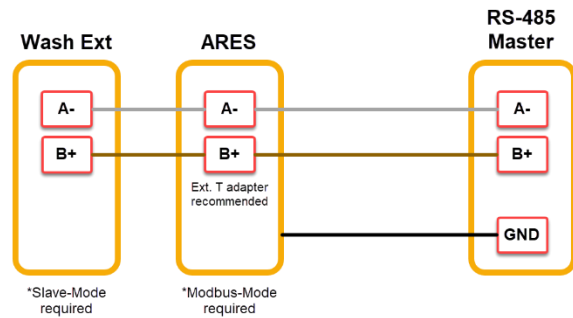
A 3m M12 cable is included inside the Wash Extension box, which is intended to connect between the ARES and the Wash Extension devices in the default cellular mode. When configured in Modbus mode, the RS-485 master needs to be connected to both Fracsun devices. These are a few potential connection methods:

#### 3.1.1 Daisy chain (recommended)

In the daisy chain configuration, the RS-485 Master is connected to ARES, and ARES is connected to the Wash Extension.

On the ARES device, an M12 “T” or “Y” adapter (Phoenix Contact part # 1541186) is installed externally to support the pass-through (daisy-chain) style connection. A second M12 plug can also be added on the rear side of the ARES device and wired internally in the factory, for an added fee.

The included M12 cable can be used for the connection between ARES and Wash Extension, but an additional M12 to loose-end cable assembly (length of choice) must be purchased to connect ARES to the RS-485 Master.

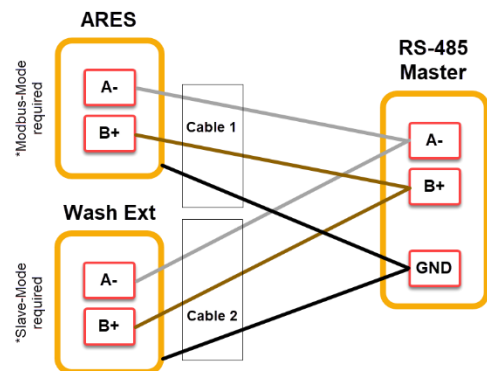


#### 3.1.2 Two individual cables to each device

In this configuration:

- One cable is run between the RS-485 Master and ARES
- A second cable is run between the RS-485 Master and Wash Extension

Because two cables need to be run, this will connection method will typically cost more for parts and installation. The included M12 cable is not used with this connection method.



### 3.2 External power requirements in Modbus-only and Cellular-only modes

In the default Cellular-only mode, ARES and Wash Extension devices are solar-powered and do not require external power. However, in Modbus-only mode, *just the ARES device requires external power*. The Wash Extension can even provide power to ARES through the M12 connection. The table below summarizes the external power requirements for each device.

Device	Mode	Ext. power req'd?	Additional Notes
ARES	Cellular-only	No	Solar-powered when outside, but can also use voltage supply.
ARES	Modbus-only	<b>Yes</b>	12V from Wash Extension can power ARES, but can also use voltage supply.
Wash Extension	Master	No	Solar-powered when outside. Do not power with voltage supply.
Wash Extension	Slave	No	Solar-powered when outside. Do not power with voltage supply.

**Note:** On the Wash Extension, an external voltage supply cannot support the surge current from the pump, so it's not recommended to power the Wash Extension with any external voltage supply.

### 3.3 ARES voltage and current specifications

If a dedicated voltage supply is used to externally power the ARES device, use an appropriate power source with the following specifications, depending on the mode used.

Parameter	Min	Typical	Max	Unit	Mode
DC Input Voltage	11	12 / 24	30	V	All
Avg. Input Current @ 12 Vdc	18	21	30	mA	Modbus-only
Peak Input Current @ 12 Vdc	280	--	600	mA	Cellular (2G/3G)
Avg. Input Current @ 24 Vdc	9	11	16	mA	Modbus-only
Peak Input Current @ 24 Vdc	145	--	325	mA	Cellular (2G/3G)

### 3.5 ARES M12 connection diagram

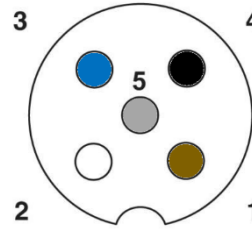
#### 3.5.1 ARES devices built 2021 and later

For installations utilizing Modbus-only mode (an external Modbus RTU connection) or Modbus + Cellular mode, use the M12 pinout diagram below to connect ARES to the RS-485 Master.

Wire	Connection description	
1	Brown	Modbus RS-485 B+ / B / B'
5	Gray	Modbus RS-485 A- / A / A'
3	Blue	Modbus common / ground *
2	White	Power Input +11 to +30Vdc **
4	Black	Power common / ground
	Shield	Housing *

\* Not a required connection.

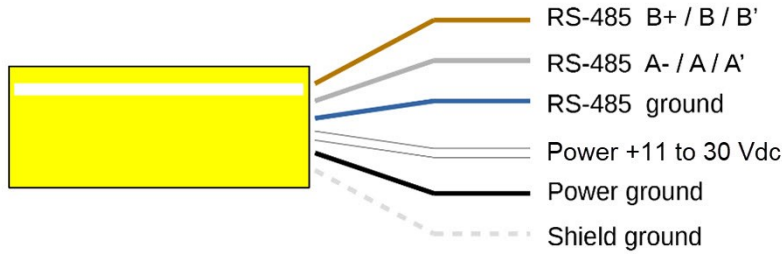
\*\* Absolute max power input voltage is 30 Vdc. *Do not exceed.*



Pin assignment for M12 socket (male)



M12 socket on rear of ARES device



M12 automation cable with loose-end wires purchased separately. Various lengths available.

### 3.6 Wash Extension M12 connection diagram

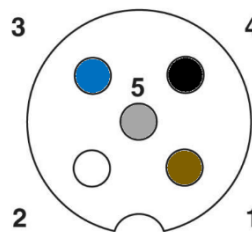
#### 3.6.1 Wash Extension devices built 2021 and later

For installations utilizing Modbus-only mode, use the M12 pinout diagram below to connect to the RS-485 Master.

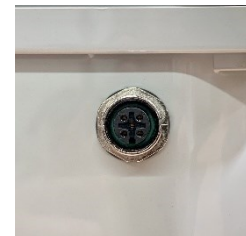
Wire	Connection description	
1	Brown	Modbus RS-485 B+ / B / B'
5	Gray	Modbus RS-485 A- / A / A'
3	Blue	Modbus common / ground *
2	White	Power Output (12V battery) **
4	Black	Power common / ground **
	Shield	Housing *

\* Not a required connection.

\*\* Power pin tied to internal 12V battery. Do not inject power into these pins. These pins are disconnected by default.



Pin assignment for M12 socket (male)



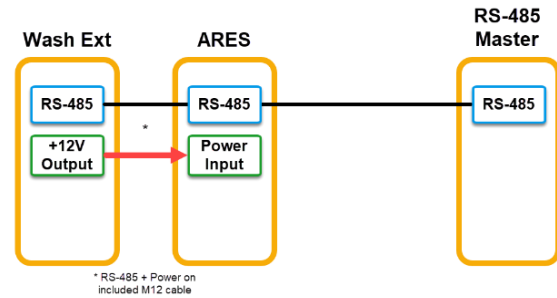
M12 socket on rear of ARES device

### 3.7 Enabling power pins on Wash Extension

To use the Wash Extension for externally powering the ARES device (or another device with a similar power draw), you can enable the power pins. The provided 5-pin M12 cable (3m) features sufficient pins to support both RS-485 and power connections between the Wash Extension and ARES.



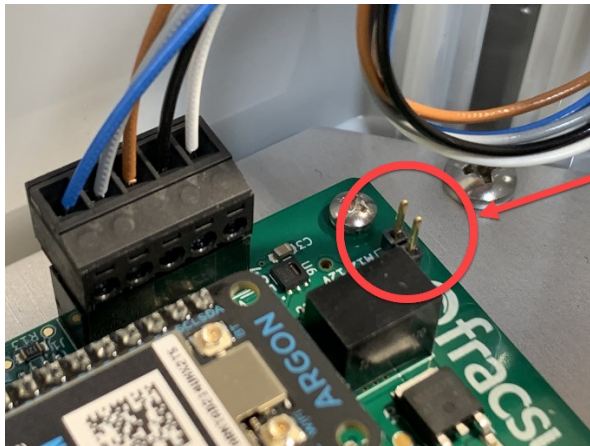
**Caution:** Do not enable the power pins on the Wash Extension if ARES is powered through an external DC source and is daisy chained to the Wash Extension.



#### 3.7.1 Devices built July 2023 and later

To enable the power pins, simply connect a shunt across the 2-pin jumper pins on the Wash Extension circuit board. When the shunt is installed, +12V power is enabled across the white (+) and black (-) wires going to the M12 connector. When the shunt is disconnected, there is no voltage across the white and black wires.

An example of a compatible 2.54mm (0.1") pitch shunt is: Würth Elektronik # 609002115121



**Jump to enable Wash Extension Power Output**

(Wash Extensions made in June 2023 or later)

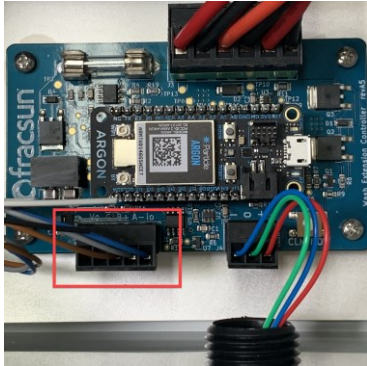
#### 3.7.2 Devices built June 2023 and earlier

The positive and negative wires must be connected on the internal terminal block inside the Wash Extension. These pins are disconnected by default in the factory.

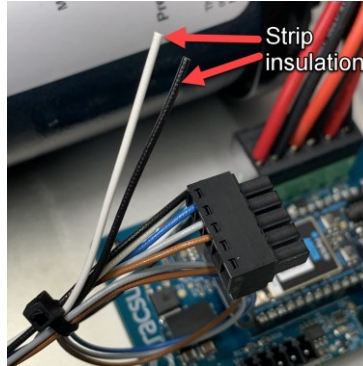
Follow the steps below to reconnect the power pins.

1. Open the Wash Extension enclosure by removing the 4 screws from the corners.
2. Remove the cover from the enclosure.
3. Disconnect one of the battery terminals and unplug the solar panel (*very important*)
4. Locate the small 5-pin terminal block that connects to the M12 socket and unplug it from the PCB.
5. Locate the loose black and white wires that come from the M12 socket.
6. Strip approximately ¼" insulation from the black and white wires using a wire stripper.

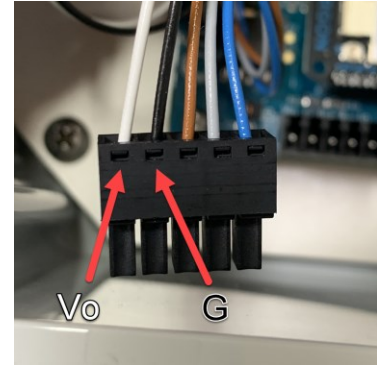
7. Insert the black (-) wire to the “G” pin and the white (+) wire to the “Vo” pin on the terminal block. See picture for a wiring diagram.
8. Using a small flat-head screwdriver, gently turn the set screws in the clockwise direction to secure the wires inside the terminal block.
9. Reconnect the 5-pin terminal block to the plug on the PCB.
10. Reconnect the battery terminal and plug the solar panel back in.
11. Reinstall the cover and screws.



Locate the 5pin terminal block, which goes to the M12 socket.



Strip the insulation on the black and white wires.



Insert the black and white wires into the terminal block and plug back in.

## 4.0 ARES device modes

Three different modes are available in the ARES device. Change the mode by writing to ARES Modbus register 44.

### 4.1 Cellular mode (default)

By default, the ARES device is configured as a self-powered and self-commissioning IoT cellular device. The device turns on, takes a few measurements, sends the data to the cloud, and then goes into a low-power sleep state to reduce battery power consumption. Any Modbus polling attempts during measurement or sleep states will almost always result in a timeout or missed poll. There is a very short ~3 second window in which a *single* poll attempt could be successful. Depending on the parameters, the window approximately occurs once every 2 - 5 minutes.



- Internal battery power is sufficient in this mode.

### 4.2 Modbus-only mode

When a more robust Modbus connection is required, a different mode (Modbus-only mode) can be enabled, but with the loss of the default cloud functions. Modbus-only mode will bypass the low-power sleep state and cellular cloud functions, but the microcontroller will always remain ON to be available for constant Modbus polling.



This mode is recommended for dataloggers, SCADA systems, and PV plants with NERC requirements.

- External power is required in this mode. The internal battery may not be sufficient in this mode.



## 5.0 Configuring the ARES device modes and settings

The ARES device modes can be changed in a variety of ways including:

- **Recommended:** Download the ARES Modbus Test and Configurator Tool. The tool is an easy-to-use program that's used in conjunction with an RS-485 to USB or RS-485 to TCP/IP converter. More information about this tool is available in the [ARES Modbus Test and Configuration Tool](#) section.
- A remote command can be issued over-the-air (OTA) through the Fracsun Web Portal. *ARES must be connected to cloud via cellular.*
- The Fracsun support team can remotely issue the command on your behalf. *ARES must be connected to cloud via cellular.*
- Write to the appropriate registers (see Modbus Mode and Apply Code parameters). *ARES must be connected to a Modbus RS-485 Master device (or a PC with Modbus Poll software).*
- Shipped preconfigured with a particular mode.
- Button press pattern on ARES main control PCB. *Not recommended, device must be opened. Reach out to Fracsun for instructions.*
- USB serial console command accessible from ARES PCB. *Not recommended, device must be opened. Reach out to Fracsun for instructions.*

### 5.1 Enabling Modbus-only mode remotely via Fracsun Dashboard

To change from Cellular-only to Modbus-only mode using this method, verify you have the following:

- DC power source (11 – 30 V) or device placed in direct sun for at least 1 hour
  - If using a DC power source, you need a M12 5pin A-coded female to loose-end cable
- Antenna fastened in with LTE or 3G signal
- Fracsun Dashboard access to the device

Follow the instructions below to enable Modbus-only mode:

1. Login to the Fracsun Portal and browse to the Device Data page.

**Note:** If you don't see the Device Data page, you have a Staff type login and you must request this access from the Owner.



Dashboard Staff **Device Data** Account ▾

2. Locate the device in the list that you would like to change and click on the Edit button.



+ Add New Device

**Note:** If you don't see the device, you can add it with the "+ Add New Device" button.

3. Scroll down to the Configuration section of the device and click on the **Enable Modbus-Only** button. Click Yes if you are sure this is the device you want to modify.



Enable Modbus-only Mode ? GO

**Warning:** Performing this action to a device at a plant will permanently turn off the internal cellular modem. This device cannot be reached until a technician is back on site.

## 5.2 Enabling Modbus-only mode via RS-485

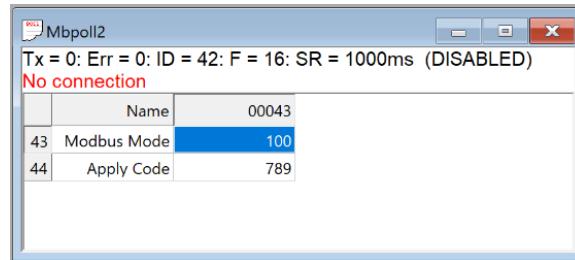
To change from Cellular-only to Modbus-only mode using this method, verify you have the following:

- DC power source (11 – 30 V) or device placed in direct sun for at least 1 hour
- M12 5pin A-coded female to loose-end cable
- Antenna connected to rear SMA connector and an LTE or 3G signal is available
- RS-485 to USB Adapter
- Computer with Modbus Poll or similar software

Follow the instructions below to enable Modbus-only mode:

1. Connect the M12 5pin female to loose-end cable between the ARES device and RS-485-to-USB adapter.
2. Power ARES using the DC power source (over the M12 cable) or by placing it in direct sun for at least 1 hour.
3. Connect the RS-486-to-USB adapter to a computer with Modbus Poll or similar software.
4. Open the Modbus polling software and perform a 0x16 (Write Multiple Registers) command *continuously* on register 43 (value: 1) and register 44 (value: 789) until a successful response is sent back.

**Note:** Polling at 1 second intervals is recommended to catch the short ~3s available polling window.



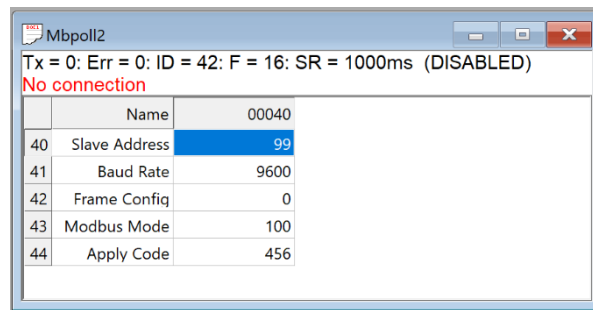
5. Once the ARES device has been polled successfully once, an automatic reset occurs and Modbus-only mode is turned on.
6. Turn off the continuously 0x16 Write Multiple Register command, otherwise the device will keep resetting.
7. You may now continuously poll at a frequency of 10 seconds or more from the RS-485 Master.

### 5.3 Configuring Slave Address, Baud Rate and Frame Config

To modify the default Modbus slave address, baud rate, or frame config settings, you must write to a specific set of registers using the Write Registers (0x16) operation while connected to the default configuration (19200, 8E1):

Parameter Name	Register	Default Value	Notes
Slave Address	40	42	Slave address 42
Baud Rate	41	19200	
Frame Config	42	1	8 databits, Even parity, 1 stop bit
Modbus Mode	43	100	Modbus-only mode ON (should already be on)
Apply Code	44	456	“456” value must be written to enable Modbus configuration

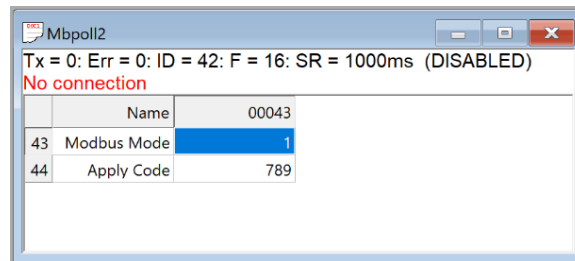
For example, if you want to change slave address to 99, baud rate to 9600, and frame config to 8N1, the following registers must be written in a single operation:



### 5.4 Enabling Cellular-only mode via RS-485

Enabling Cellular-only mode will turn the cellular modem back on and disable Modbus-only mode. This may be used to enable Cellular-only mode indefinitely or to temporarily enable for an OTA firmware update.

Perform a Write Register (0x16) operation *once* per the following example:



After a successful write, Cellular-only mode is enabled the device will start communicating with the Fracsun cloud, as long as an external antenna is connected and the SIM card is activated.

## 5.5 Reading ARES data

### 5.5.1 Typical registers to read during normal operation

The ARES sensor takes a new measurement once per minute and records new values to the Modbus holding registers. These measurement-specific registers are the sensor's most important values and should be polled at a regular frequency. The measurement registers contain the following data: soiling loss (SL), soiling ratio (SR), temperature (cell and interior), effective irradiance (G) and short-circuit current (Isc) of both clean and soiled reference cells.

The image below shows the recommended registers to poll during normal operation. This is polled using a standard 0x03 (Read Holding Registers) operation:

Recommended Modbus polling definition for ARES	
<b>Operation:</b>	0x03 (Read Holding Registers)
<b>Slave ID:</b>	42 (default)
<b>Starting register:</b>	0
<b>Number of registers to read:</b>	11
<b>Recommended scan rate:</b>	60 s
<b>Minimum scan rate:</b>	10 s (for intermittent testing only)
<b>Response timeout:</b>	1000 ms

Name	Value
0	Irradiance (C) 0
1	Irradiance (S) 0
2	Isc (C) 0
3	Isc (S) 0
4	Soiling Loss 0
5	Soiling Ratio 0
6	Norm (C) 0
7	Norm (S) 0
8	Temp (Ref. Cell) 0
9	Temp (Interior) 0
10	LTO Voltage 0

### 5.5.2 Interpreting ARES data

It's important to understand the typical range of the values to determine the measurement accuracy and general device health of the device. Use the table below to help interpret the values and set actions / alarms based on the dataset.

Parameter name	Register #	Typical Range (after scalar)	Notes
<b>Effective irradiance</b>	0, 1	0 - 1320	Effective irradiance is derived from the short-circuit current and is temperature corrected.
<b>Short-circuit current</b>	2, 3	0 - 10	Increases as solar irradiation increases.
<b>Soiling loss</b>	4	0 - 100	Values less than 0 are possible during shading events or if cells are not normalized.
<b>Soiling ratio</b>	5	0 - 1	Values greater than 1 are possible during shading events or if cells are not normalized.
<b>Normalization values</b>	6, 7	-30 to 30	Normalization values saved and being applied to effective irradiance and soiling.
<b>Temperatures</b>	8, 9	-40 to 150	Backsheet cell temperature is usually higher than the interior temperature during the day.
<b>LTO Voltage</b>	10	2.3 - 2.8	If external power is applied to ARES (typical in Modbus-only mode), battery voltage should stay in fully charged state (2.75 to 2.8V).

## 5.6 Normalization

Normalization is a process where the two reference cell measurements are adjusted so that transmission loss between them equals 0% (or soiling ratio equals 1). Normalizing the ARES sensor is important during the following events:

- After the plant, or the section of the plant associated with ARES, is washed.
- Annually, whenever possible (typically after a heavy rain event)

After the auto-normalization process has completed, the normalization values will be readable in register 6 (clean) and register 7 (soiled). Specifically, these values are the irradiance adjustment required to match the clean and soiled ref cells @ 500 W/m<sup>2</sup>. For example, if the clean ref cell measured 495 W/m<sup>2</sup> and the dirty ref cell measured 505 W/m<sup>2</sup> during ~ 500 W/m<sup>2</sup> of irradiation, the normalization numbers would be set to 5 (clean) and -5 (dirty).

### 5.6.1 Resetting the soiling measurement

To prepare for normalization, manually clean both the clean and soiled reference cells, as seen in the image to the right.

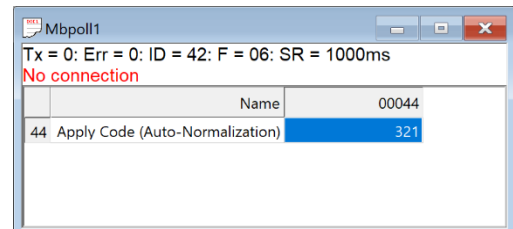
After both cells have been completely wiped down, verify that the irradiance is **greater than 500 W/m<sup>2</sup>** and approximately **-0.5% to 0.5% transmission loss** (soiling loss). If non-ideal GHI conditions are present on the cleaning day, verify the following day.

Once all these conditions have been met, you may run the auto-normalization process.



### 5.6.2 Running auto-normalization

1. To initiate the auto-normalization process, perform a 0x06 (Write Single Register) operation to register 44 with value "321".
2. Wait approximately 90 seconds for the process to complete.
3. View the results of the auto-normalization process to verify it worked correctly.
  - a. Perform a 0x03 (Read Register) operation to register 46.
  - b. Register 46 will have the following values: 1 = Success, 2 = Failed during testing, 3 = Not enough sunlight to calibrate, 4 = No normalization reqd.
  - c. The new normalization values may be reviewed by reading registers 6 (clean) and 7 (soiled).



**Important Note:** You can also use the [ARES Modbus Test and Configuration Tool](#) to run auto-normalization.

## 6.0 ARES Modbus Test and Configurator Tool

This tool is a standalone software application designed to facilitate the testing and configuration of ARES devices using Modbus protocol. Developed in Python, this tool is engineered to operate independently without the need for additional software installations. Its core functionalities include the ability to:

- Switch between Modbus-only and Cellular-only modes
- Configure Modbus RS-485 ID, baud, and frame config
- Execute Auto-Normalization
- Conduct test measurements directly through Modbus.

This makes it an essential tool for configuring and troubleshooting ARES devices over RS-485 in the field or in staging environments before final deployment.

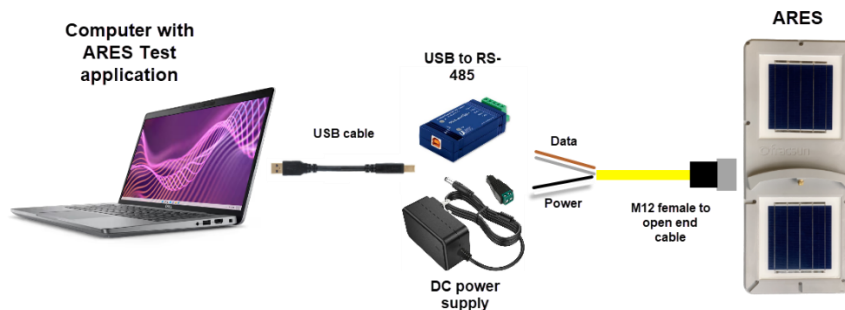
### 6.1 Requirements

- Computer with Windows, MacOS, or Linux software
- USB to RS-485 converter, or a TCP/IP to RS-485 converter (Device Server)
- 11-30 V DC source
- M12 (5pin, A-coded, Female) to open-end cable (eg. Phoenix part # 1406169)

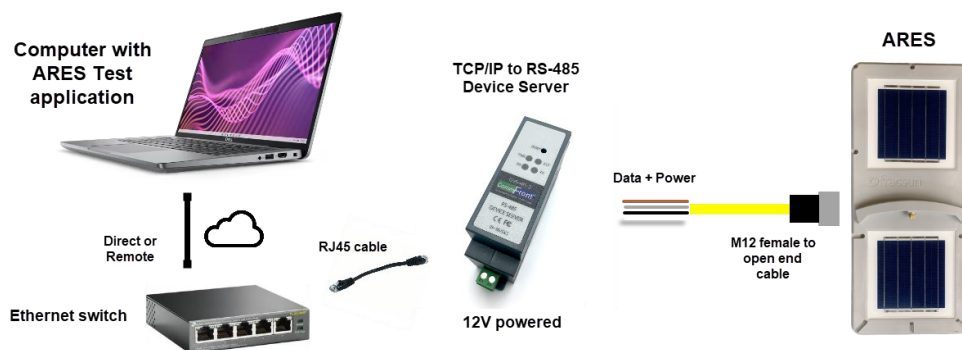


### 6.2 Connection diagram

#### 6.2.1 USB to RS-485 converter (direct serial connection)



#### 6.2.2 TCP/IP to RS-485 device server (ethernet or remote connection)



## 6.3 Application overview

The screenshot shows the ARES Modbus Test and Configurator Tool v0.1.3 interface. It is divided into several sections:

- Measurement Toggle:** Located at the top left, it includes buttons for 'Update Values Once', 'Keep Updating Values', 'Stop', and 'Scan Rate' (set to 5). Below these are counters for 'Tries', 'Fails', and 'Success', along with a 'Clear' button.
- Measurement values:** A table on the right side of the top section, divided into 'Clean' and 'Soiled' columns. It lists various parameters such as Irradiance, Short Circuit, Temp Cell, Norm. Values, Soiling Loss, Soiling Ratio, Temp PCB, and LTO Voltage, with units like W/m2, Amps, and °C.
- Device Settings:** A section on the left below the toggle, with buttons for 'Enable Modbus-only Mode', 'Enable Cellular-only Mode', 'Enable Night-Sleep Mode', and 'Disable Night-Sleep Mode'.
- Change Modbus Settings:** A section below device settings, including fields for 'Slave Address' (42), 'Baud Rate' (19200), and 'Frame Config' (8-E-1), with a 'change all now' button.
- Auto-Normalization:** A section with a 'Run Auto-Normalize Now' button.
- Manual Normalization:** A section with 'Set Norm. Value (Clean)' and 'Set Norm. Value (Soiled)' fields and 'update' buttons.
- Output Log:** A large text area on the right side, showing connection logs such as 'COM4: Argon Serial (COM4)', 'COM13: USB Serial Port (COM13)', and 'Successfully connected to COM13'.

Callouts with arrows point from text boxes to these sections:

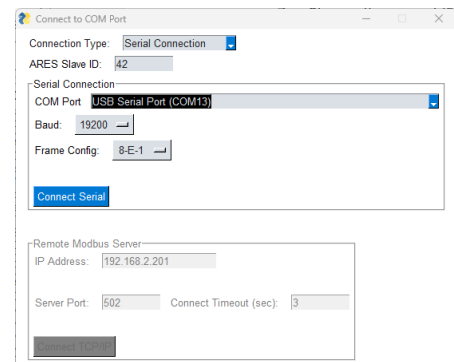
- Measurement Toggle:** Start/stop measuring values, select the scan-rate, and view if Modbus polls are being successfully sent.
- Measurement values:** A list of all values being retrieved over RS-485 when Update values is selected.
- Device Settings:** Enable Modbus or Cellular-only modes, change the Modbus configuration, or perform normalization.
- Output Log:** Information regarding the connection status and Modbus commands being issued.

## 6.4 Tool functions

### 6.4.1 Connecting to the USB converter or TCP/IP Server

Under the File menu, click on "Connect". To connect ARES using a USB to RS-485 converter, use the Serial Connection and follow these steps:

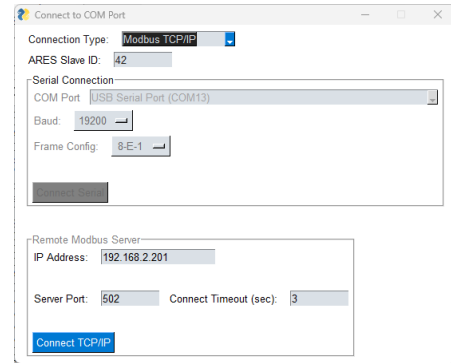
1. From the 'Connection Type' dropdown menu, select 'Serial Connection'.
2. In the 'ARES Slave ID' field, enter the slave ID for your device, if it's not already set. In the example, it is set to the default '42'.
3. Set the 'COM Port' to the one corresponding to your serial device. If you're unsure, you can check the available COM ports in your operating system's device manager. The dropdown will only show active Serial COM ports, so verify that another application doesn't have the COM port open.
4. Set the 'Baud' rate to match the baud rate of your device; in the image, it is set to the default '19200'.
5. Choose the correct 'Frame Config' from the dropdown list. The current setting shown is the default '8-E-1', which stands for 8 data bits, even parity, and 1 stop bit.
6. Once all settings are correctly selected, click the 'Connect Serial' button to establish the connection.



To connect ARES using a TCP/IP to RS-485 converter (a Device Server), use the

1. From the 'Connection Type' dropdown menu, select 'Modbus TCP/IP'.
2. In the 'ARES Slave ID' field, enter the slave ID for your device, if it's not already set. In the example, it is set to the default '42'.

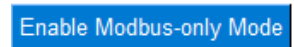
3. Enter the IP Address of the Modbus TCP/IP device server you wish to connect to in the 'IP Address' field.
4. Enter the server's port number in the 'Server Port' field. In this case, it is already set to '502', which is the standard port for Modbus TCP/IP communications.
5. Specify the 'Connect Timeout (sec)' according to how long you want the application to attempt a connection before timing out. The current setting is '3' seconds.
6. Click the 'Connect TCP/IP' button to initiate the connection to the remote Modbus device server.



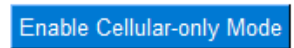
Make sure that the server is online and that there are no network issues that could prevent the application from establishing a connection. If the connection fails, check the IP address, server port, and timeout settings, and ensure that your network allows traffic on the specified port.

### 6.4.2 Changing between Modbus-only and Cellular-only modes

Enable **Modbus-only Mode**, click on the button that says "Enable Modbus-only Mode". This will configure the device to communicate exclusively via the Modbus protocol. In some cases, it may take up 15 minutes to enable this mode. Please be patient.



Enable **Cellular-only Mode**, click on the button that says "Enable Cellular-only Mode". This will switch the device to operate solely over the cellular network.



**Please note:** Enabling one mode will disable the other communication modes. For example, if Modbus-only Mode is enabled, the device's cellular modem will be shutdown and not attempt to connect over cellular.

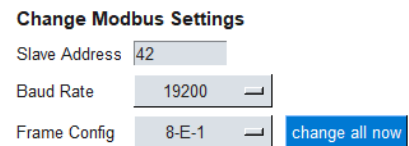
Always ensure that you are selecting the correct mode for your operational requirements. If you need to switch modes, simply click the corresponding button for the desired mode and wait.



### 6.4.3 Changing Modbus settings

The "Change Modbus Settings" section of your application is designed to configure the Modbus communication parameters.

First, set the 'Slave Address' to the new address you'd like to change it to. Next, choose the 'Baud Rate' from the dropdown to match the speed at which your Modbus network communicates. Finally, select the 'Frame Config' to determine the format of the data frames in terms of data bits, parity, and stop bits. After all the desired settings are selected, click on the 'change all now' button to apply the changes to the device.



Once these settings are changed and successfully saved into ARES, you must reconnect using the new Modbus settings that you changed them too.



### 6.4.4 Auto-normalization and manual normalization

The "Auto-Normalization" and "Manual Normalization" section of the application provides two methods for setting normalization values. The "Run Auto-Normalize Now" button initiates an automated process to determine and set the normalization values based on current measurements or pre-defined standards.

**Auto-Normalization**

**Manual Normalization**  
 Set Norm. Value (Clean)    
 Set Norm. Value (Soiled)

For more control, the "Manual Normalization" area allows users to enter specific normalization values for both 'Clean' and 'Soiled' conditions. After inputting these values, you must press the corresponding 'update' button to apply each new setting individually. This feature is crucial for maintaining accuracy in measurement by adjusting the baseline or reference points according to specific conditions or standards.

**Important Note:** Do not run normalization without understanding the requirements! Please read more about this in 5.6 Normalization.

### 6.4.5 Measurement

The "Measurement" section in your application is designed for monitoring and updating the values from a connected device. With the "Update Values Once" button, you can manually trigger a single data retrieval to update the measurement fields for both 'Clean' and 'Soiled' reference cells, such as Irradiance, Short Circuit Current, Temperature, and Normalization values.

The "Keep Updating Values" button starts a continuous monitoring process, automatically updating the values at a frequency determined by the 'Scan Rate' in seconds (set in the example as 5 seconds). The counters for 'Tries', 'Fails', and 'Success' provide feedback on the number of attempts to read data, how many of those attempts have failed, and how many have been successful, respectively.

If there are too many failures, this could indicate communication issues with the device. The 'Clear' button resets these counters, and the 'Stop' button halts the continuous updating process. A red indicator light, as shown in the image, may suggest an active process or an error state that requires attention.

Measurement		Clean	Soiled	
<input type="button" value="Update Values Once"/>	<input type="button" value="Keep Updating Values"/>	<input type="button" value="Stop"/>	Scan Rate	5
Tries	Fails	Success		
<input type="text" value="2"/>	<input type="text" value="3"/>	<input type="text" value="0"/>	<input type="button" value="Clear"/>	
<input checked="" type="checkbox"/> <span style="color: red; font-weight: bold;">●</span>				
Irradiance	<input type="text"/>	<input type="text"/>		W/m2
Short Circuit	<input type="text"/>	<input type="text"/>		Amps
Temp Cell	<input type="text"/>	<input type="text"/>		°C
Norm. Values	<input type="text"/>	<input type="text"/>		
Soiling Loss	<input type="text"/>	<input type="text"/>		%
Soiling Ratio	<input type="text"/>	<input type="text"/>		
Temp PCB	<input type="text"/>	<input type="text"/>		°C
LTO Voltage	<input type="text"/>	<input type="text"/>		Vdc
<input checked="" type="checkbox"/> Firmware v38+ <input type="button" value="Get FW version"/>				

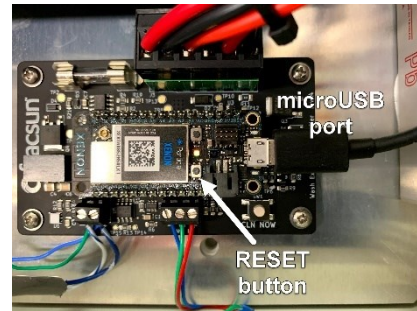
## 7.0 Wash Extension device modes and configuration

### 7.1 Modbus modes – Master/Slave

In the default mode, the Wash Extension functions as a Modbus Master (for a Fracsun-only connection). If connecting the Fracsun soiling monitoring system to an external Modbus RTU connection, the Wash Extension Modbus mode must be changed to function as a Modbus Slave. Changing this mode ensures that two Modbus Masters are not on the same RS-485 network.

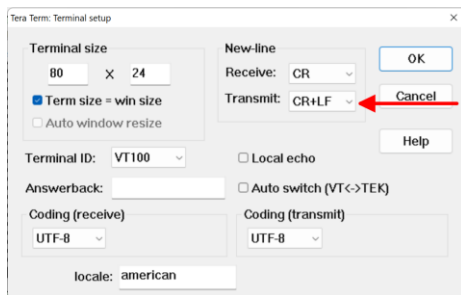
### 7.2 Changing the Modbus Mode to SLAVE via USB

The mode, and other Modbus configurations, can be changed by interfacing a computer to the Wash Extension controller via the micro-USB connection on the board labeled “Xenon”. With a few keypresses inside a serial console, the configuration can be performed. Follow the instructions below to change the Wash Extension to SLAVE-mode.



#### 7.2.1 Windows Setup

1. On a Windows computer, download [Tera Term VT](#) serial terminal program.
2. Open the serial terminal program and enable the CR + LF option:
3. In Tera Term, go to Setup > Terminal. Under New-line - Transmit dropdown, select **CR+LF**.



4. Connect a microUSB cable between the Fracsun “Argon” board and a Windows 10+ computer.
  5. In Tera Term, open a New Connection to the COM port associated with the Argon board. Often, it’s displayed as “Argon Serial (COM#)”.
- Note:** If the Argon Serial COM port is not visible, the [Windows Particle USB Drivers](#) might have to be installed. The default USB COM settings are: 9600 baud rate, 8 databits, NONE parity, and 1 stop bits.
6. Once connected, the console will show all device logging information.
  7. Press “m” *once* on your keyboard, wait up (up to 2 minutes) to access the Modbus menu in the console.

**Tip:** To eliminate any long wait times, press the Reset button on the Argon board and press the “m” key immediately after the COM port reconnects in Tera Term. The Modbus menu should appear a couple seconds later.

8. Press “2” and then “y” to enable SLAVE mode.

```

COM3 - Tera Term VT
File Edit Setup Control Window Help
000001705 [app.measure] INFO: TEMP: 22.0 C
000001705 [app.measure] INFO: TEMP (rolling avg): 22.0 C
000001954 [app.measure] INFO: HUMIDITY: 32.0
000001965 [app.measure] INFO: HUMIDITY (rolling avg): 32.0
000002016 [app.measure] INFO: IR LEVEL SENSOR: 3.21
000002017 [app.measure] INFO: IR LEVEL SENSOR (rolling avg): 3.21
000002059 [app.measure] INFO: BATTERY VOLTAG: 12.25
000002172 [app.measure] INFO: TEMP: 22.0 C
000002173 [app.measure] INFO: HUMIDITY: 32.0 %
000002174 [app.sched] INFO: Wash schedule (hours): 25, 25, 25
000005275 [app] INFO:
000005276 [app] INFO: -----
000005276 [app] INFO: Menu >> Modbus settings:
000005277 [app] INFO: 1: View current Modbus settings.
000005277 [app] INFO: 2: Turn ON Slave mode for external RIU connection
000005279 [app] INFO: 3: Turn OFF Slave mode for Fracsun-only connection
000005281 [app] INFO: 4: Slave address
000005282 [app] INFO: 5: Baud rate
000005283 [app] INFO: 6: Frame config (databits/parity/stopbits)
000005285 [app] INFO: 7: Reset modbus factory defaults (for Fracsun-only connection)
000005288 [app] INFO: x: exit
000012381 [app] INFO: >> Are you sure you want to turn on Slave mode? ARES and the Wash Extension will no longer communicate directly.
000012382 [app] INFO: The Wash Extension must be controlled by the new Modbus master.
000012383 [app] INFO: >> Press 'y' to accept.
000017650 [app] INFO: >>
000017651 [app] INFO: >> Modbus Slave mode ON
000017652 [app] INFO: >> Verify all other Modbus settings are set correctly, then reset.
  
```

9. If required, you can also change the slave address, baud rate, and frame configuration in this menu system. The default values are:

- a. Slave Address: 43
- b. Baud Rate: 19200
- c. Frame Configuration: 8 databits, EVEN parity, and 1 stop bit (8E1).

10. Type the “x” key or physically press the RESET button on the Argon board to start using Slave Mode.

### 7.3 Washing configuration and control

When the Wash Extension is configured in Modbus Slave mode, wash events must be setup correctly to ensure properly scheduled wash events. Triggering wash events can be performed by either the Wash Extension or the Modbus Master (in most cases, this is either a SCADA controller or datalogger), or both in rare scenarios. If the Wash Extension triggers the wash events, Unix time must be available as a write variable. The checklist below provides suggestions on which method to use:

Modbus Master Device	Unix Time available	Recommended Method
SCADA controller	No	Modbus-Master triggered
SCADA controller	Yes	Wash-Ext triggered
Datalogger	No	Modbus-Master triggered
Datalogger	Yes	Wash-Ext triggered

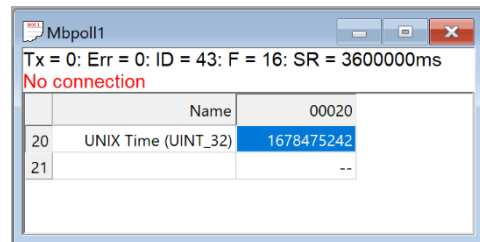
To learn more about these two methods and how to configure them, continue reading the sections below.

## 7.4 Wash-Ext triggered washing (Wash Hour registers)

Using this method, the Wash Extension firmware manages the wash times. Wash events occur on the hour(s) configured in the Wash Hour registers. Up to 3 washes can be preconfigured. There is one caveat – the Unix (or Epoch) time must be written to the Wash Extension at a defined interval so that the device knows what time it is. There is no real time clock (RTC) to keep time during power-loss events, but the Wash Extension can keep time while powered, just not very accurately.

To setup the Wash Extension in this mode, follow the instructions below:

1. Verify communication between the Wash Extension and Modbus Master (SCADA controller or datalogger).
2. Write the current timezone for the device's location *once*.
  - a. Perform a 0x06 (Write Single Register) operation to register 13 with the timezone value.
  - b. If the timezone negative, convert to an unsigned number. (ex: -7 timezone converts to 65529).
  - c. This value is retained in flash memory. It only needs to be set once.
  - d. If the Wash Extension is installed in an area that observes daylight savings time (DST), it is important to adjust the timezone to accommodate the semi-annual shift in daylight savings time. This ensures that the Wash Extension initiates washing at the appropriate Wash Hours.
3. Write the Wash Hour(s) *once*.
  - a. Perform a 0x16 (Write Multiple Register) operation to registers 50 through 52 with the wash hours.
  - b. The Wash Hour value is based on a 24hr clock. Write "25" to the register if you don't want to wash.
  - c. For example, to wash at only 6pm (18:00), write the values 18, 25, 25 to registers 50, 51, and 52 respectively.
  - d. This value is retained in flash memory. It only needs to be set once.
4. During normal operation, write the Unix time approximately *every 3 hours*.
  - a. Perform a 0x16 write operation to registers 20-21 with the current Unix time as a UINT\_32 value.
  - b. This value is not retained in flash memory. Perform write operations on a schedule for appropriate time keeping.
5. Washing will now occur at the configured Wash Hour(s).



	Name	00020
20	UNIX Time (UINT_32)	1678475242
21		--

**Important note:** If the Wash Extension temporarily loses power or resets, washing will not occur until Unix time is written again. Verify that Unix time is still writing to the device after it powers back on.

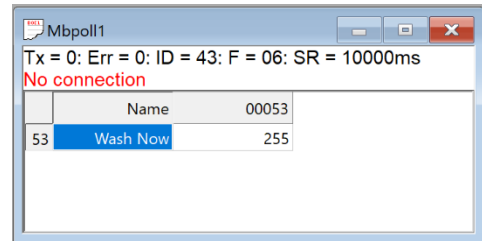
## 7.5 Modbus-master triggered washing (Wash Now register)

Using this method, the Modbus Master (SCADA controller or datalogger) manages the wash times. Wash events only occur when the Wash Now register is written to. This method is used if you prefer to write this register using your own schedule or timer, or if you don't have the Unix / Epoch time available in the SCADA or datalogger as a variable.

To setup the Wash Extension in this mode, follow the instructions below:

1. Verify communication between the Wash Extension and Modbus Master (SCADA controller or datalogger).
2. Decide on a daily washing schedule.
3. To trigger a wash, perform a 0x06 (Write Single Register) operation to register 53 with value "255".
  - a. Register 53 will reset to '0' once the wash is complete.

**Note:** If using this method, it's best practice to write all Wash Hour register values to "25" (turn off Wash-Ext triggered washing).



**Additional Note:** It's also recommended to write the Unix / Epoch time (instructions in 7.4 step 4) so the "Last Wash Time" is a more meaningful value upon historical review of Wash Events in the SCADA or datalogger system.

## 7.6 Reading Wash Extension data

### 7.6.1 Typical registers to read during normal operation

During typical operation, the Wash Extension outputs valuable data used for evaluating the device health, environmental data, and wash events. The Wash Extension outputs the battery voltage, temperature, humidity, and liquid level sensor reading. After wash events, the wash power and time are retrievable until the next wash event.

The image below shows the recommended registers to poll during normal operation. This is polled using a standard 0x03 (Read Holding Registers) operation:

Recommended Modbus polling definition for Wash Extension	
<b>Operation:</b>	0x03 (Read Holding Registers)
<b>Slave ID:</b>	43 (default)
<b>Starting register:</b>	40
<b>Number of registers to read:</b>	7
<b>Recommended scan rate:</b>	60 s
<b>Minimum scan rate:</b>	10 s ( <i>for intermittent testing only</i> )
<b>Response timeout:</b>	1000 ms

	Name	00040
40	Last Wash Time (Unix UInt32)	0
41		--
42	Wash Power	0
43	Temperature	0
44	Relative Humidity	0
45	Battery Voltage	0
46	Tank Level	0

### 7.6.2 Interpreting Wash Extension data

It's important to understand the typical range of the values to determine general device health and successful / failed wash events. Use the table below to help interpret the values and set actions / alarms based on the dataset.

Parameter name	Register #	Typical Range (after scalar)	Notes
Last Wash Time	40-41	< 4070934000	Only accurate if UNIX time is being written on a schedule. Otherwise, the value is incorrect.
Wash Power	42	60 - 85	Tubing is cut or tank is empty if less than 60.
Temperature	43	-40 – 125 C	Used to measure temperature before washing. Sensor failed if outside this range.
Relative Humidity	44	0 - 100	If greater than 95, possible water intrusion in enclosure. Sensor failed if outside this range.
Battery Voltage	45	12.5 – 14.4	Battery voltage is always lower after wash events. Solar panel disconnected if battery voltage steadily dropping under 12.5 Vdc.
Tank Level	46	0 or 3	Liquid level sensor, 0 = water, 3 = absence of water. When 3 is first detected, tank has 25% remaining.

## 8.0 ARES Modbus Map (v38 and later)

Parameter name	Holding register	Modbus data type	R/W	Scalar	Final data type	Description
<b>Irradiance (Clean)</b>	0	UINT_16	R		UINT_16	Effective irradiance in W/m2 of the <b>Clean</b> reference cell. Is a temperature-compensated and post-calibrated value.
<b>Irradiance (Soiled)</b>	1	UINT_16	R		UINT_16	Effective irradiance in W/m2 of the <b>Soiled</b> reference cell. Is a temperature-compensated and post-calibrated value.
<b>Shunt current (Clean)</b>	2	UINT_16	R	1000	FLOAT	Shunt (short-circuit) current of the <b>Clean</b> reference cell. Is not a temperature-compensated value.
<b>Shunt current (Soiled)</b>	3	UINT_16	R	1000	FLOAT	Shunt (short-circuit) current of the <b>Soiled</b> reference cell. Is not a temperature-compensated value.
<b>Soiling Loss</b>	4	INT_16	R	100	FLOAT	Instantaneous Soiling Loss value in %. Computed as $100 * (1 - SR)$
<b>Soiling Ratio</b>	5	UINT_16	R	10000	FLOAT	Instantaneous Soiling Ratio value (no units). Computed per IEC 61724-1.
<b>Normalization (Clean)</b>	6	INT_16	R+W		INT_16	Normalization value of the <b>Clean</b> ref cell. Range is -30 to 30.
<b>Normalization (Soiled)</b>	7	INT_16	R+W		INT_16	Normalization value of the <b>Soiled</b> ref cell. Range is -30 to 30.
<b>Temperature (cell)</b>	8	INT_16	R	100	FLOAT	Temperature in degrees Celsius of the reference cell backsheet.
<b>Temperature (interior)</b>	9	INT_16	R	100	FLOAT	Temperature in degrees Celsius of the ARES interior (main PCB).
<b>Battery voltage</b>	10	UINT_16	R	1000	FLOAT	LTO battery voltage in Volts.
<b>Soiling Loss Avg. (yesterday)</b>	11	INT_16	R		FLOAT	Future update (2021) – not available yet. The daily average soiling loss value (%) from the previous day.
<b>Cellular RSSI</b>	12	INT_16	R		INT_16	Received Signal Strength Indicator in dBm from the last cell connection.
<b>Slave address</b>	40	UINT_16	R+W		UINT_16	Modbus slave address (1 – 255). Default is 42.
<b>Baud rate</b>	41	UINT_16	R+W		UINT_16	Modbus baud rate. Acceptable rates are 1200, 2400, 4800, 9600, 14400, 19200, and 38400. Default is 19200.
<b>Frame config</b>	42	UINT_16	R+W		UINT_16	Modbus frame configuration. Acceptable values are 0 and 4. 0 = 8 Databits, None Parity, 1 Stop Bit 4 = 8 Databits, Even Parity, 1 Stop Bit (default)
<b>Modbus mode</b>	43	UINT_16	R+W		UINT_16	Modbus mode selection. Use the table below to select a mode: 1 = Cellular mode / intermittent Modbus (default mode) 100 = Modbus-only mode (cellular modem off) 200 = Modbus + Cellular mode (experimental)
<b>Apply code</b>	44	UINT_16	R+W		UINT_16	Apply code that must be written <i>prior</i> to modifying Normalization values, Modbus config, or Modbus mode registers. 321 = Enable auto-normalization (recommended) 123 = Enable manual-normalization 456 = Enable Modbus configuration (Slave address, baud, frame) 789 = Enable Cellular/Modbus mode change This register automatically resets to 0 after any write operation to registers 6, 7, or 40-43.
<b>Reset</b>	45	UINT_16	R+W		UINT_16	Resets the device. Write 255 once to enable. Register is automatically reset to 0 upon reset.
<b>Auto-Normalize result</b>	46	UINT_16	R		UINT_16	The result of the auto-normalization process. 1 = Success, 2 = Failed during testing, 3 = Not enough sunlight to calibrate, 4 = No normalization reqd.
<b>Night-Sleep Mode</b>	48	UINT_16	R+W		UINT_16	Enables/disables night-sleep mode. When enabled, ARES will enter a deep sleep state during darkness. Acceptable values are: 0 = Disabled, 1 = Enabled
<b>Both ref. cells cleaned</b>	49	UINT_16	R+W		UINT_16	Write "2" when both cells were cleaned. After a 2 is detected, the registers will change to "1" and be available for reading. These registers will reset to their default state "0", after 3 hours.
<b>Clean ref. cell cleaned</b>	50	UINT_16	R+W		UINT_16	Write "2" when "clean" reference cell was manually cleaned. After a 2 is detected, the registers will change to "1" and be available for reading. These registers will reset to their default state "0", after 3 hours.
<b>Firmware revision</b>	51	UINT_16	R		UINT_16	Current firmware revision running ARES program.

## 10.0 ARES Modbus Map (v37 and earlier)

**Important note:** These mappings should only be used on firmware versions v37 and earlier. The holding register addresses are the same in v38 and later, but some scalars were changed to increase precision (Shunt current, soiling ratio, and LTO battery voltage).

Parameter name	Holding register	Modbus data type	R/W	Scalar	Final data type	Description
Irradiance (Clean)	0	UINT_16	R		UINT_16	Irradiance in W/m2 of the <b>Clean</b> reference cell. Is a temperature-compensated and post-calibrated value.
Irradiance (Soiled)	1	UINT_16	R		UINT_16	Irradiance in W/m2 of the <b>Soiled</b> reference cell. Is a temperature-compensated and post-calibrated value.
Shunt current (Clean)	2	UINT_16	R	100	FLOAT	Shunt (short-circuit) current of the <b>Clean</b> reference cell. Is not a temperature-compensated value.
Shunt current (Soiled)	3	UINT_16	R	100	FLOAT	Shunt (short-circuit) current of the <b>Soiled</b> reference cell. Is not a temperature-compensated value.
Soiling Loss	4	INT_16	R	100	FLOAT	Instantaneous Soiling Loss value in %. Computed as the percent difference between the Clean vs. Soiled irradiance (temp compensated).
Soiling Ratio	5	UINT_16	R	1000	FLOAT	Instantaneous Soiling Ratio value (no units). Computed per IEC 61724-1.
Normalization (Clean)	6	INT_16	R+W		INT_16	Normalization value of the <b>Clean</b> ref cell. Range is -30 to 30.
Normalization (Soiled)	7	INT_16	R+W		INT_16	Normalization value of the <b>Soiled</b> ref cell. Range is -30 to 30.
Temperature (cell)	8	INT_16	R	100	FLOAT	Temperature in degrees Celsius of the reference cell backsheet.
Temperature (interior)	9	INT_16	R	100	FLOAT	Temperature in degrees Celsius of the ARES interior (main PCB).
Battery Voltage	10	UINT_16	R	100	FLOAT	LTO battery voltage in Volts.
Soiling Loss Avg. (yesterday)	11	INT_16	R		FLOAT	Future update (2021) – not available yet. The daily average soiling loss value (%) from the previous day.
Cellular RSSI	12	INT_16	R		INT_16	Received Signal Strength Indicator in dBm from the last cell connection.
Slave address	40	UINT_16	R+W		UINT_16	Modbus slave address (1 – 255). Default is 42.
Baud Rate	41	UINT_16	R+W		UINT_16	Modbus baud rate. Acceptable rates are 1200, 2400, 4800, 9600, 14400, 19200, and 38400. Default is 19200.
Frame Config	42	UINT_16	R+W		UINT_16	Modbus frame configuration. Acceptable values are 0 and 4. 0 = 8 Databits, None Parity, 1 Stop Bit 4 = 8 Databits, Even Parity, 1 Stop Bit (default)
Modbus Mode	43	UINT_16	R+W		UINT_16	Modbus mode selection. Use the table below to select a mode: 1 = Cellular mode / intermittent Modbus (default mode) 100 = Modbus-only mode (cellular modem off) 200 = Modbus + Cellular mode (experimental)
Apply Code	44	UINT_16	R+W		UINT_16	Apply code that must be written <i>prior</i> to modifying Normalization values, Modbus config, or Modbus mode registers. 321 = Enable auto-normalization (recommended) 123 = Enable manual-normalization 456 = Enable Modbus configuration (Slave address, baud, frame) 789 = Enable Cellular/Modbus mode change This register automatically resets to 0 after any write operation to registers 6, 7, or 40-43.
Reset	45	UINT_16	R+W		UINT_16	Resets the device. Write 255 once to enable. Register is automatically reset to 0 upon reset.
Auto-Calibrate Result	46	UINT_16	R		UINT_16	The result of the auto-normalization process. 1 = Success, 2 = Failed during testing, 3 = Not enough sunlight to calibrate, 4 = No normalization reqd.
Night-Sleep Mode	48	UINT_16	R+W		UINT_16	Enables/disables night-sleep mode. When enabled, ARES will enter a deep sleep state during darkness. Acceptable values are: 0 = Disabled, 1 = Enabled



## 11.0 Wash Extension Modbus Map

The default SLAVE ID/address is: **43**. The address can be changed via the microUSB cable. See instructions above.

Parameter Name	Address	Modbus Data type	R/W	Scalar	Final Data type	Description
<b>Timezone</b>	13	INT_16	R+W		INT_16	Current timezone ( <i>required</i> for proper wash schedule). If negative, convert to an unsigned number. (ex: -7 timezone would convert to 65529). <i>This value is retained in flash memory. It only needs to be set once – new installs, during DST time changes, or changing station locations.</i>
<b>Unix Time</b>	20-21	UINT_32	R+W		UINT_32	The number of seconds that have elapsed since January 1, 1970 (midnight UTC/GMT). <i>Value is not retained in flash memory.</i> <i>Recommended to write every 3 hours for proper time keeping.</i>
<b>Last Wash Time</b>	40-41	UINT_32	R		UINT_32	Unix time value of the last wash event
<b>Wash Power</b>	42	UINT_16	R	100	FLOAT	Pump motor power (in watts) of the last wash event. Values between 50 – 85 W indicate a successful wash.
<b>Temperature</b>	43	INT_16	R	100	FLOAT	Temperature in degrees Celsius inside the Wash Extension enclosure.
<b>Relative Humidity</b>	44	UINT_16	R	100	FLOAT	Relative humidity (%) inside the Wash Extension enclosure. Can often detect rain events in dry/arid climates.
<b>Battery Voltage</b>	45	UINT_16	R	100	FLOAT	Lead-acid battery voltage in Volts.
<b>Tank Level</b>	46	UINT_16	R		UINT_16	Liquid level sensor output in Volts. 0 = presence of water, 3 = absence of water <i>When the output switches from 0 to 3, this indicates approximately 8-10 weeks before the reservoir is empty.</i>
<b>Wash Temp Limit *</b>	48	UINT_16	R+W		UINT_16	The temperature limit (in K) when washing is disabled. Default is 277 K (3.85 C, 38.9 F).
<b>Wash Seconds *</b>	49	UINT_16	R+W		UINT_16	The number of seconds to perform a wash for. Default is 5 seconds.
<b>Wash Hour 1</b>	50	UINT_16	R+W		UINT_16	Hour to initiate a wash event #1 (0-23). Default is 25 (no wash). <i>This value is retained in flash memory.</i>
<b>Wash Hour 2</b>	51	UINT_16	R+W		UINT_16	Hour to initiate a wash event #2 (0-23). Default is 25 (no wash). <i>This value is retained in flash memory.</i>
<b>Wash Hour 3</b>	52	UINT_16	R+W		UINT_16	Hour to initiate a wash event #3 (0-23). Default is 25 (no wash). <i>This value is retained in flash memory.</i>
<b>Wash Now</b>	53	UINT_16	R+W		UINT_16	Initiate a wash now. Write 255 once to enable. Register is automatically reset to 0 upon wash.

\* Only available in Wash Extension v27 firmware and later.



**FracSun Inc.**

San Luis Obispo, California, USA

Web: [www.fracsun.com](http://www.fracsun.com)

Email: [info@fracsun.com](mailto:info@fracsun.com)

Phone: 805-242-3722