USER GUIDE

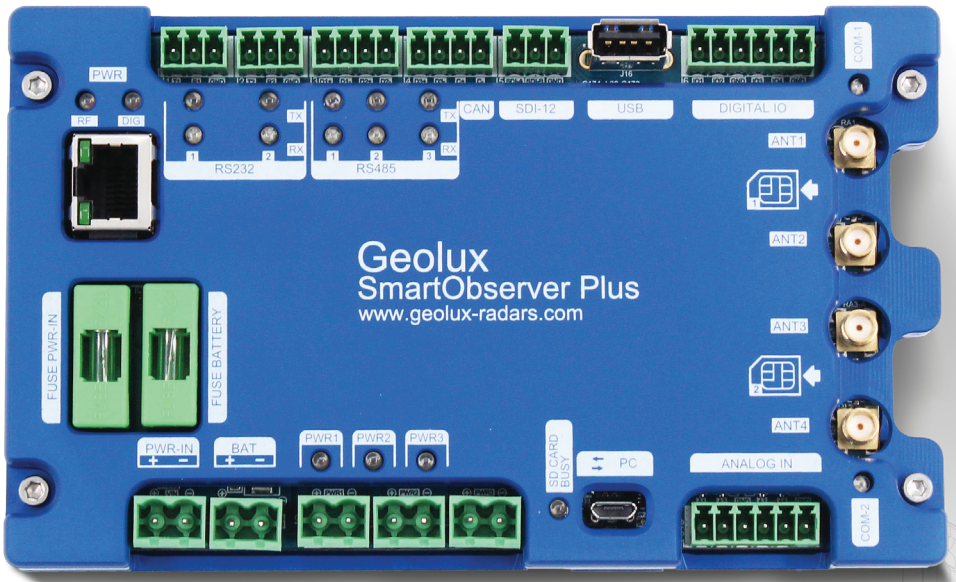
SmartObserverPlus-Series

Geolux

Distributed by AKRIBIS

Geolux d.o.o. | Ljudevita Gaja 62 | HR-10430 Samobor Croatia | www.geolux-radars.com | geolux@geolux.hr

SmartObserver **Plus**



Data Logger

USER Manual

v1.0.0.

Starting Point

Thank you for purchasing the Geolux SmartObserverPlus data logger! We have put together the experience of our engineers, the domain knowledge of our customers, the enthusiasm of our team, and the manufacturing excellence to deliver this product to you.

You may freely rely on our field-proven technology. The use of top-quality components and advanced signal processing algorithms ensures that Geolux SmartObserverPlus data logger can be used in various applications and environments.

We have created this User Manual to assist you in setting up and using the Geolux device. Should there be any questions left unanswered, please feel free to contact us directly:

Geolux d.o.o.
Ljudevita Gaja 62
10430 Samobor
Croatia

E-mail: **geolux@geolux.hr**
Web: **www.geolux-radars.com**

Contents

1. Safety Information **5**
2. Introduction **7**
3. Device Interface and Connections **8**
4. Connecting the Power Supply **15**
4.1. Order of Connecting power sources 16
4.2. Order of Disconnecting power sources 16
5. Initial Device Setup **17**
6. Connecting Sensors **22**
6.1. Connecting Modbus Sensors 23
6.2. Connecting SDI-12 Sensors 26
6.3. Connecting Sensors to Analog Inputs 29
6.4. Connecting Sensors to Digital Inputs 31
6.5. Defining the Quantity and Unit 33
6.6. Two-Point Calibration 35
6.7. Update / Export / Import / Delete Device Configuration 36
7. Configuring Data Upload **38**
7.1. HTTP Data Upload 38
7.2. FTP Data Upload 39
8. Updating the Firmware **40**
8.1. Firmware Update Overview 40
8.2. Firmware Update Process 41
8.3. Step-by-Step instructions 43
8.4. Restoring Previous Firmware 47
Appendix A: Configuration Protocol **48**
A.1. Protocol Overview 48
A.2. CRC algorithm 48
A.3. Protocol Sentences 49
Device information 49
Protocol Sentences 49
Clear Datalog 53
Status Report file information 54
Fetching Status Report File 54
Read Bootmode 55
Start Firmware Update 55
Enable/Disable Logging to SD-card 56
Enable/Disable Logging to USB thumb drive 56
Set Battery Capacity 57

Contents

Set Battery Charge Current	57
Enable/Disable Battery Charging	58
Set Battery Charging Maximum Voltage	58
Set MPPT (Maximum Power Point Tracking) Voltage	59
Set Battery Charging Minimum Temperature	59
Set Scan Interval	60
Enable/Disable DHCP for Ethernet	60
Set DNS Servers for Ethernet	61
Set Ethernet Gateway addresses	61
Set Static IP Address for Ethernet	62
Set Subnet Mask for Ethernet	62
Set APN for Modem 1	63
Set APN for Modem 2	63
Set DNS Servers for Modem 1	64
Set DNS Servers for Modem 2	64
Set APN Password for Modem 1	65
Set APN Password for Modem 2	65
Set APN Username for Modem 1	66
Set APN Username for Modem 2	66
Set System Time	67

1 Safety Information

General Safety Instructions

- **Read Before Use:** Carefully read this user manual before operating the SmartObserverPlus device.
- **Qualified Personnel Only:** The installation, operation, and maintenance of this device should only be performed by qualified and trained personnel. If necessary, contact Geolux for training or support.
- **Follow Manufacturer Guidelines:** Always adhere to the safety instructions outlined in this manual and any warnings printed on the device or its components.

Electrical and Environmental Considerations

- **Voltage and Current Compliance:** Ensure all connected power supplies and sensors conform to the specified voltage and current ranges listed in the product specifications. Exceeding these values may cause damage.
- **Environmental Protection:** The SmartObserverPlus must be protected from moisture and harsh environmental conditions. Use enclosures meeting IP66 or NEMA 4 standards or higher. Use desiccator if needed.
- **Temperature Considerations:** Device components may reach high temperatures when exposed to direct sunlight or high ambient conditions. Always consider these factors during installation to ensure safe operation.
- **Battery Handling:**
  - Avoid short circuits, mechanical damage, or exposure to high temperatures.
  - Do not open or tamper with batteries. Properly dispose of them as per environmental regulations.
  - Handle battery acids with caution; wear protective gloves and eyewear.

Device Handling and Maintenance

- **Avoid Unauthorized Repairs:** Do not open or attempt to repair the SmartObserverPlus device. Repairs should be carried out only by authorized Geolux service centers.
- **Cleaning:** Clean the device using a soft, lint-free cloth lightly dampened with a mild cleaning solution. Do not spray liquids directly onto the device.
- **Approved Accessories:** Use only manufacturer-approved accessories and spare parts to maintain warranty and operational integrity.

Specific Warnings

- **Explosive Atmospheres:** Do not operate the SmartObserverPlus in explosive or flammable environments. The device is not rated for such conditions.
- **Pacemaker Interference:** The device's radio frequency signals may interfere with cardiac pacemakers:

  Maintain a minimum distance of 35 cm between the pacemaker and the antenna / antenna connector.
- **Cold/Hot Surfaces:** Metal parts of the device can become hot at ambient temperatures above 60°C, or cold at ambient temperatures below 0°C. Use protective gloves when handling the device in such conditions.

Liability and Warranty

- **Proper Usage:** Geolux assumes no liability for damages arising from misuse, improper installation, or unauthorized modifications of the device. Ensure the SmartObserverPlus is used in accordance with the instructions in this manual.
- **Warranty Nullification:** Any unauthorized modifications, misuse, or failure to comply with this manual's safety instructions voids the warranty.

Emergency Guidelines

- **Power Disconnection**: In case of an emergency, disconnect the both the power supply (PWR-IN) and the battery (BAT) from the device and follow standard safety protocols.
- **Support Contact:** If you encounter any issues or have safety concerns, immediately contact Geolux customer support.

2 Introduction

The Geolux SmartObserverPlus datalogger is a versatile, highly integrated device designed for environmental, hydrological and meteorological monitoring applications. It collects data from connected hydro-meteorological instruments, stores the data locally and, optionally, transmits the data in real-time via Ethernet or GPRS/4G/5G data link.

Key Features

Data Interfaces:

- Two RS-232 interfaces.
- Three RS-485 / Modbus interfaces.
- Two SDI-12 interfaces.
- One CAN bus interface.
- Four single-ended analog inputs, configurable for either voltage measurement or 4-20 mA current measurement.
- Four digital I/O lines, configurable as pulse counters, frequency counters, or digital outputs.

Local Storage:

- Internal Flash memory provides secure backup storage for data when wireless communication is unavailable.
- MicroSD card slot supports extended local storage with SD card media.
- USB host connector enables the use of USB flash drives as an alternative local data storage option.

Power Management:

- The device has integrated MPPT (Maximum Power Point Tracking) battery charger. It supports simultaneous connection to a power source (e.g., solar panel) and a backup battery, automatically optimizing charging and power usage.
- Compatible with multiple battery types, including lead-acid, NiMH, NiCd, LiFePO4, and Li+.
- Three controllable output power lines to supply power to instruments. Between periodic measurements these lines will be powered down to optimize battery consumption.

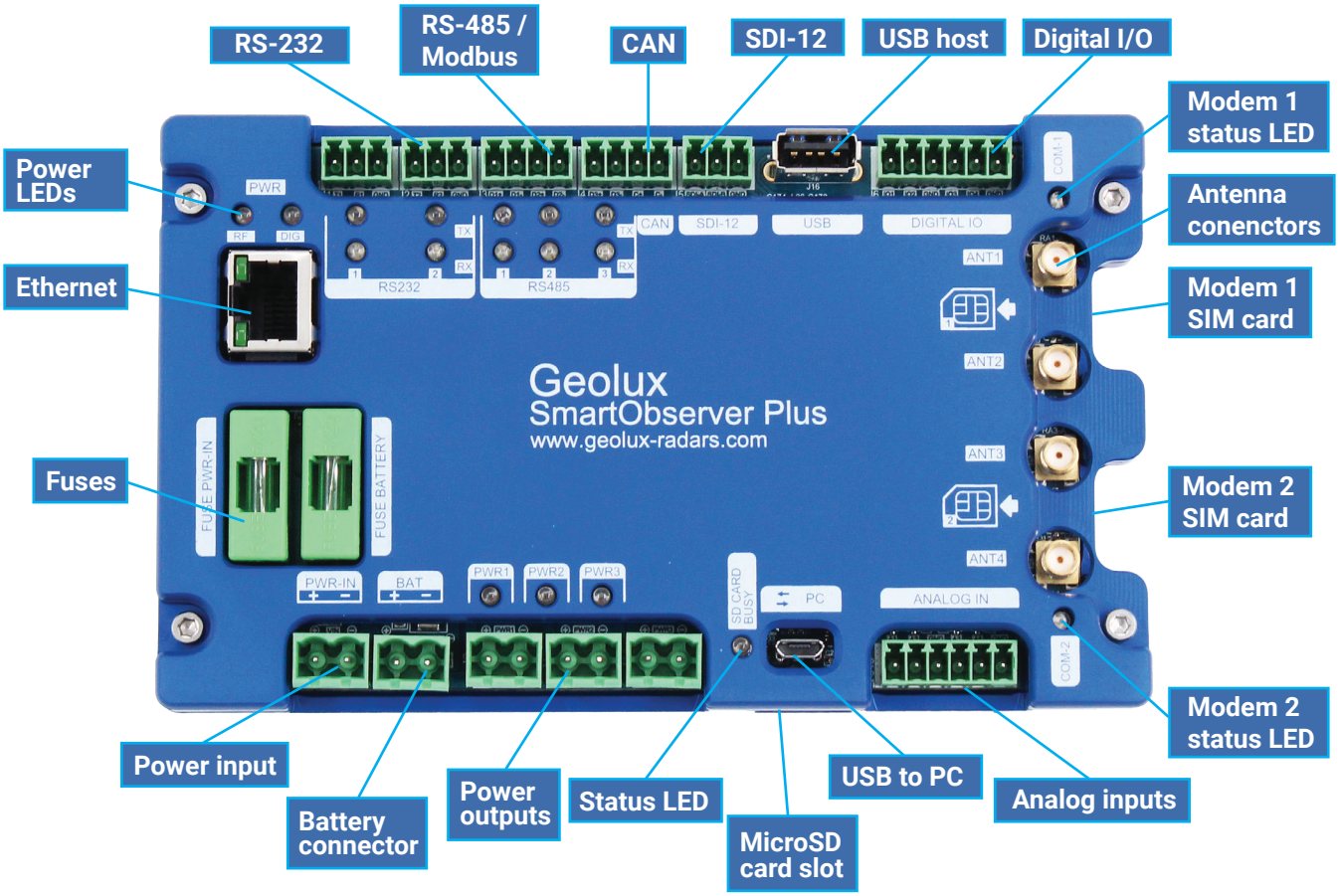
Remote Data Transmission:

- Built-in Ethernet interface supports data transmission using HTTP, HTTPS, FTP or SFTP protocols to a remote server.
- Models with single-SIM or dual-SIM capabilities allow data transfer via wireless cellular networks, including GPRS, 4G, 5G and NB-IoT.
- Tight integration with the Geolux Hydroview cloud platform for hydrometeorological data acquisition and analysis (optional, but not required).

Customization via MicroPython scripting:

- Includes a MicroPython scripting framework, enabling advanced users to customize and take full control of the device's operation.

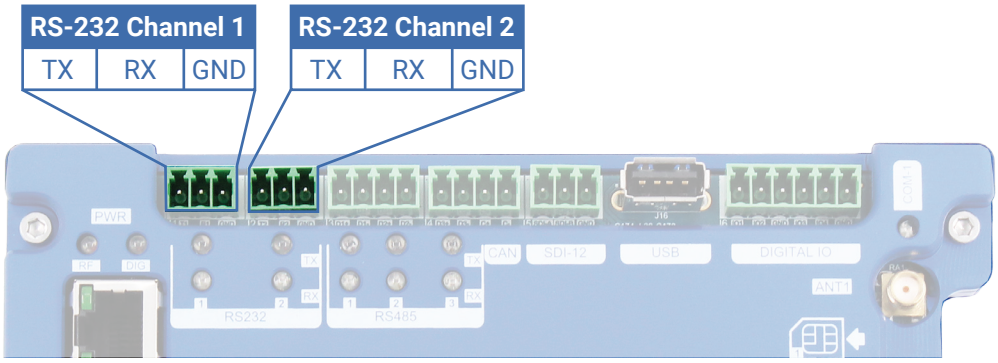
3 Device Interface and Connections



RS-232

The datalogger includes two standard RS-232 ports, providing interfaces for instruments that support RS-232 communication. These ports can be accessed programmatically via MicroPython scripts. Two status LEDs per channel indicate line activity: one blinks to indicate RX (receive, data being sent from connected device to the datalogger) activity, and the other blinks during TX (transmit, data being sent by the datalogger) activity on the line.

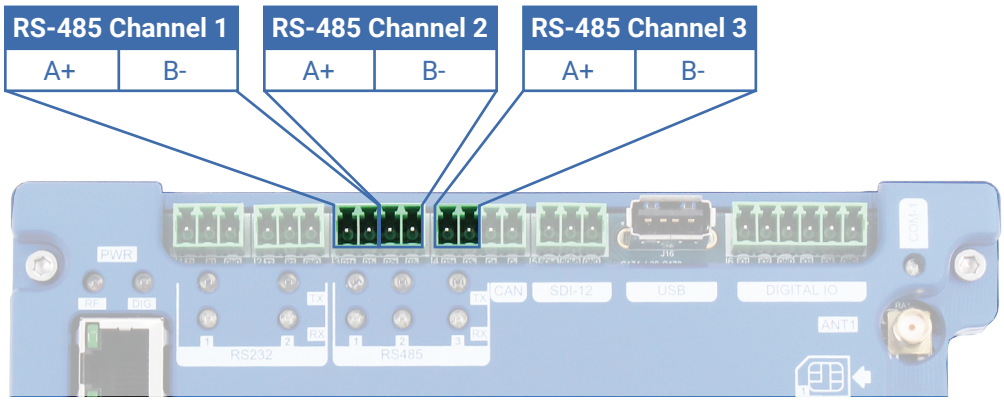
The RS-232 connector pinout is displayed in the following picture:



RS-485 / Modbus

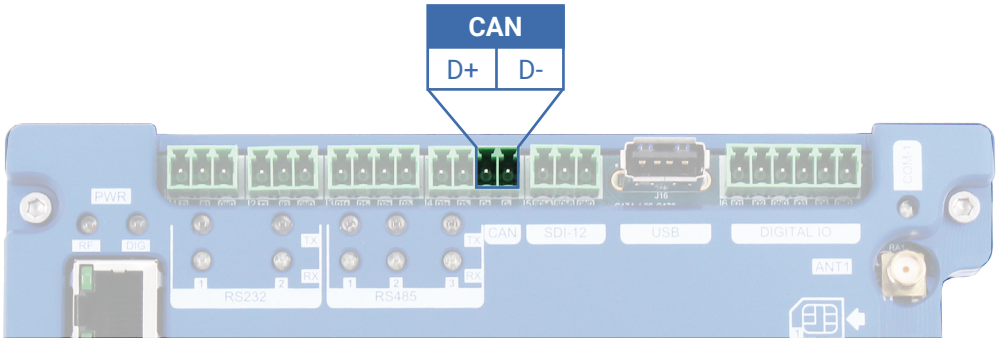
The datalogger features three standard half-duplex RS-485 ports, providing interfaces for instruments that support RS-485 or Modbus communication protocols. These ports are utilized when the datalogger is configured to interface with Modbus instruments. Since Modbus protocol supports connecting multiple instruments on the same line, provided that each instrument has a different device address, the number of Modbus instruments that can be connected to the SmartObserverPlus datalogger is not limited to three. Additionally, the RS-485 ports can be programmatically accessed via MicroPython scripts for custom applications. Each channel includes two status LEDs to indicate line activity: one blinks to show RX (receive, data being sent from connected device to the datalogger) activity, and the other blinks during TX (transmit, data being sent by the datalogger) activity.

The RS-485 connector pinout is displayed in the following picture:



CAN

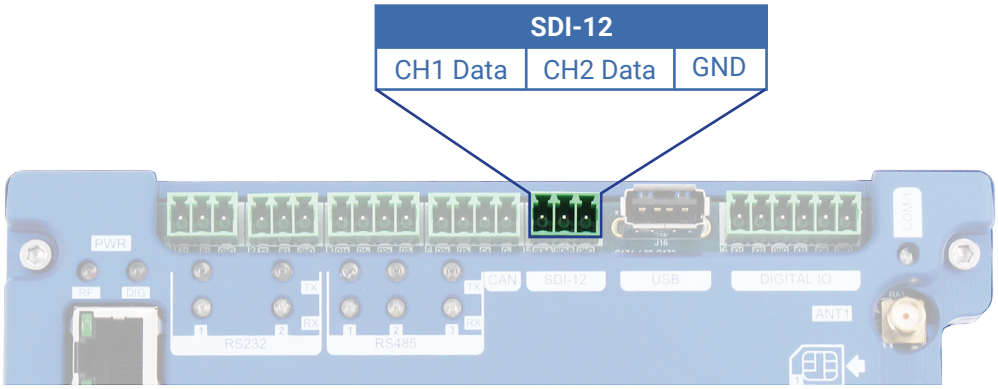
The datalogger includes a single CAN bus interface, enabling high-speed communication with instruments that utilize the CAN bus protocol. This interface can be programmatically accessed via MicroPython scripts, providing flexibility for custom applications.



SDI-12

The datalogger includes two SDI-12 interfaces, allowing connection to SDI-12-compatible instruments. These interfaces are used when the datalogger is configured to communicate with SDI-12 instruments. Since the SDI-12 protocol supports multiple instruments on the same line, provided each instrument has a unique device address, the number of SDI-12 instruments that can be connected to the SmartObserverPlus is not limited to two. Additionally, the SDI-12 ports can be programmatically accessed via MicroPython scripts for custom applications.

A SDI-12 data terminal connection is provided for each SDI-12 port. One common GND terminal is provided. The power supply for SDI-12 instruments is not provided on the SDI-12 terminal block, but instead the SDI-12 instruments should use one of three available PWR outputs on the datalogger.

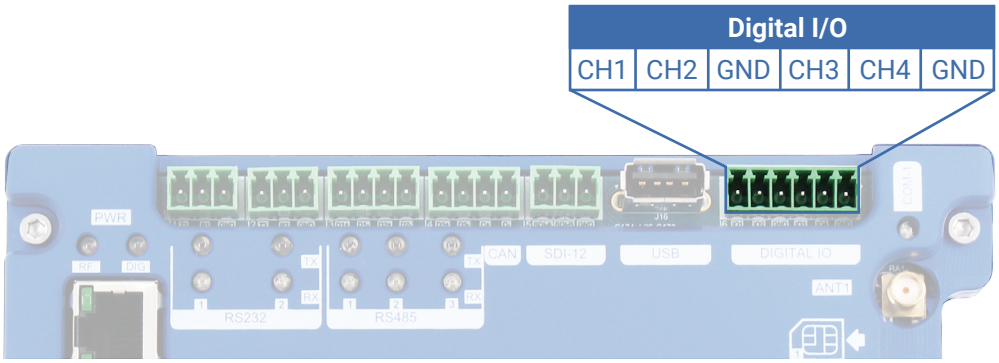


USB Host

The USB host connection supports USB flash drives for data storage. When a USB flash drive is connected, the datalogger can save all measurements in CSV format to the drive, provided it is configured to do so. The SmartObserverPlus datalogger supports FAT16, FAT32, and exFAT file systems on USB flash drives.

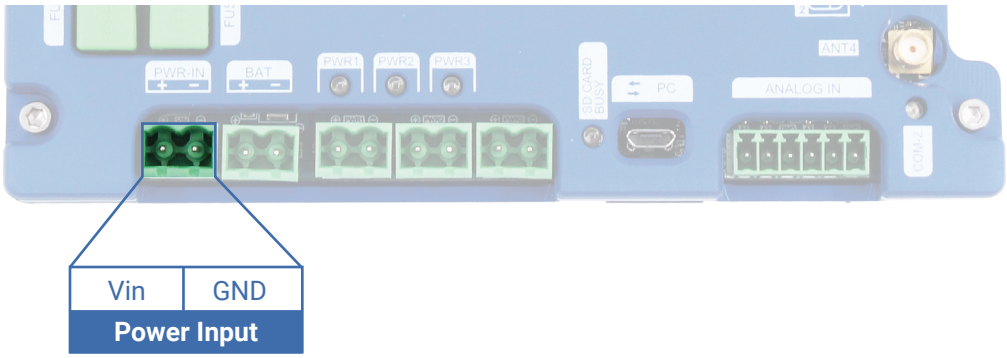
Digital I/O

The datalogger is equipped with four digital input/output pins, each of which can be configured to function as a pulse counter or frequency counter input. Alternatively, the pins can be used as outputs, which can be controlled programmatically via MicroPython scripts.



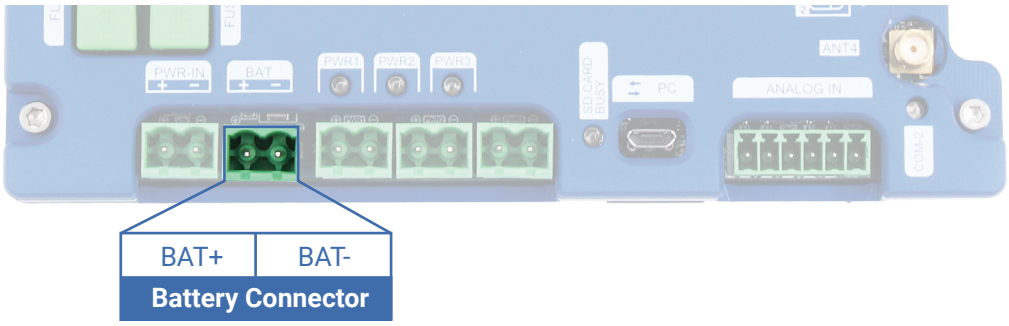
Power Input

The power input terminals are used to connect a solar panel or wind generator to the datalogger, serving as its primary power source. This power source also charges the connected battery. If an AC/DC power source is available alongside the battery, it should be connected to these terminals.



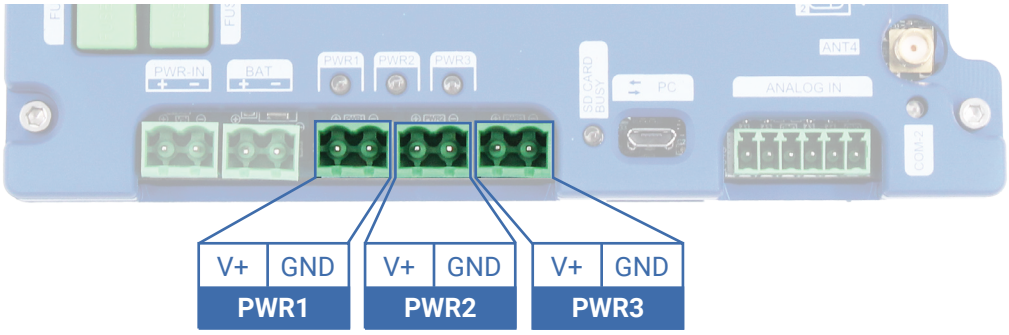
Battery Connector

The battery connection terminals are used to connect the datalogger to a battery. If the datalogger is not using a battery and is powered solely by an AC/DC adapter, the adapter should be connected to the battery terminals instead of the power input terminals. The supported battery voltage range is 9.5 - 19.2 volts.



Power Outputs

The Geolux SmartObserverPlus datalogger features three controlled power outputs, each capable of supplying up to 3 A at the current battery voltage. These outputs remain off until the datalogger needs to take a measurement from a connected instrument. Then, the datalogger powers up the corresponding output, waits for a specified duration, collects the measurement from the instrument, and then powers it off. To conserve battery life, instruments with a power consumption of less than 3 A should be connected to these controlled outputs rather than directly to the battery, so that the instruments remain completely disconnected from the battery between measurements. Each of the power outputs has a corresponding LED which is turned to indicate that the power is supplied to the specific power output.



MicroSD Card Slot

Similar to a USB flash drive, an SD card provides a convenient way to expand the local storage capacity of the datalogger. When an SD card is inserted, the datalogger can save all measurements in CSV format to the card, provided it is configured accordingly. The SmartObserverPlus datalogger supports FAT16, FAT32, and exFAT file systems for SD cards. Additionally, SD cards can be used to install firmware upgrades, as described in Chapter 8.

Status LED

The status LED provides visual indications of the datalogger's operation status and activity related to the SD card or USB flash storage. The following states are indicated:

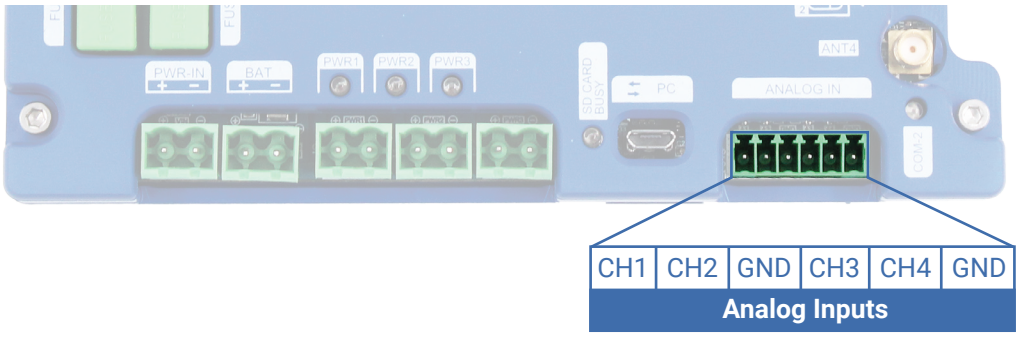
- **LED off:** The datalogger is either powered off or in low-power sleep mode.
- **LED blinks once every two seconds:** The datalogger is active but no valid SD card or USB drive is detected.
- **LED blinks continuously (twice per second):** The datalogger is active, and a valid SD card or USB drive is detected.
- **LED is continuously on:** The datalogger is writing data to the SD card or USB drive. **Do not remove the SD card or USB drive while the LED is on,** as this may corrupt the file system.

USB to PC

The micro-USB device connector enables a direct connection between the SmartObserverPlus datalogger and a PC for setup and configuration. Using the Geolux Instrument Configurator application on the PC, users can connect to the datalogger to modify operational settings and parameters, as well as analyze and diagnose its performance. Detailed instructions on using the Geolux Instrument Configurator can be found in Chapter 5.

Analog Inputs

The SmartObserverPlus datalogger is equipped with four single-ended analog inputs, designed to enable connections to a variety of analog instruments. Each input can be individually configured to function as either a voltage input or a 4-20 mA current input.



Power LEDs

The two power LEDs, located in the upper right corner of the datalogger, indicate its current operational status. Their behavior is as follows:

- **Both LEDs off:** The datalogger is either powered off or in low-power sleep mode.
- **RF off, DIG on:** The datalogger is active, but the modems are not powered.
- **RF on, DIG on:** The datalogger is active, and one or both modems are powered.
- **RF on, DIG off:** This combination indicates an error. If this pattern occurs, please contact Geolux support for assistance.

Ethernet

The datalogger features a standard RJ45 connector, providing 10/100 Mbps Ethernet connectivity. The Ethernet port can be used in two ways:

- To connect the datalogger to the internet via a switch, router, or gateway.
- To establish a local connection with Ethernet-enabled instruments, such as an IP camera.

The behavior and configuration of the Ethernet port can be managed using the **Geolux Instrument Configurator** PC application. For advanced use cases, the Ethernet port can also be controlled programmatically via MicroPython scripts.

Fuses

The SmartObserverPlus datalogger is equipped with two fuses to protect the power input and battery connection against over-current conditions. For convenience, two replacement fuses are included in the SmartObserverPlus packaging.

4 Connecting the Power Supply

When a single power source is available:

- The power source must be connected to Battery Connector (**BAT**).

When two power sources are available:

- If a Battery is used it must be connected to Battery Connector (**BAT**), other power source types must be connected to Power Input Connector (**PWR-IN**). Battery charging must be ENABLED in the Configuration.
- If a Battery is not used, any power source types can be connected to either Power Input Connector (**PWR-IN**) or Battery Connector (**BAT**). Battery charging must be DISABLED in the Configuration.

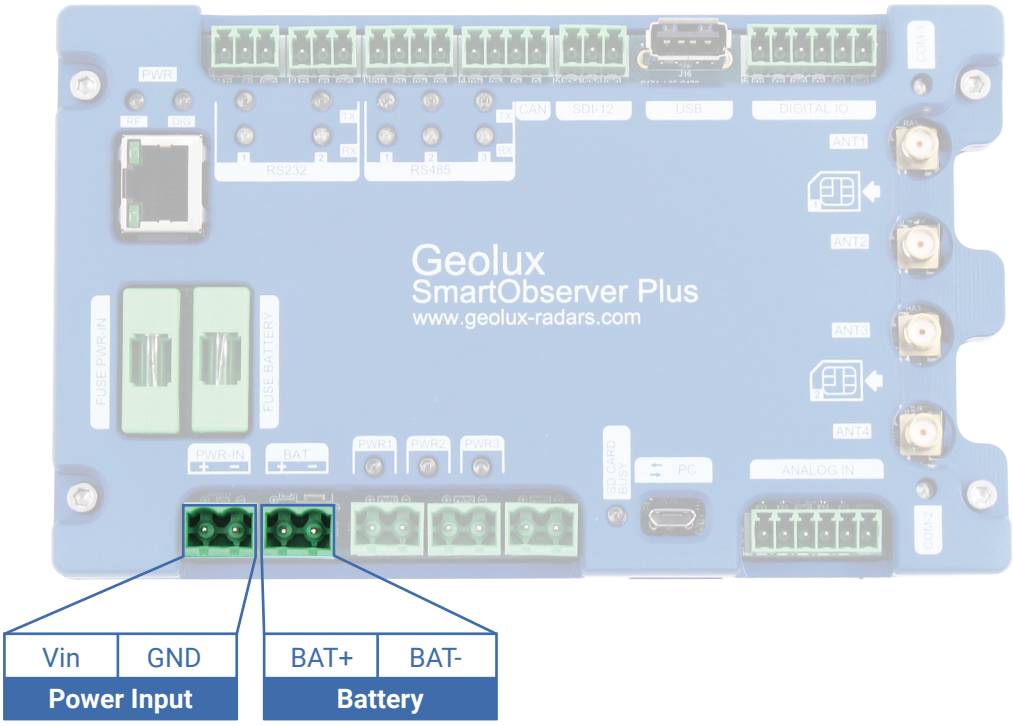


Figure 1. Power Input & Battery Connectors

Table 1. Example - Power sources & connections to Connectors

Power Sources available	Power Input PWR-IN	Battery BAT	Battery charging (Configuration)
only Battery	None	Battery	DISABLED
only AC/DC	None	AC/DC	DISABLED
PV Panel, AC/DC	PV Panel	AC/DC	DISABLED
Battery, AC/DC	AC/DC	Battery	ENABLED
Battery, PV Panel	PV Panel	Battery	ENABLED

4.1. Order of Connecting power sources

First connect the power source at the Battery Connector, then connect the power source at the Power Input connector.

When SmartObserverPlus is connected to a power source, i.e. it has been powered off, it will power on and start executing.

4.2. Order of Disconnecting power sources

First disconnect the power source from the Power connector, then disconnect the power source from the Battery Connector.

5 Initial Device Setup

Initial device setup is done by connecting the device to the *Geolux Instrument Configurator* application. Connect the device to the PC via a micro-USB cable and provide power to the BAT line. When the device is connected to the PC via the micro-USB cable, Windows assigns a COM port number. To find the assigned number, open *Device Manager* and look under the *Ports (COM & LPT)* section. If the COM port is not listed, download and install the required drivers from https://ftdichip.com/drivers/.

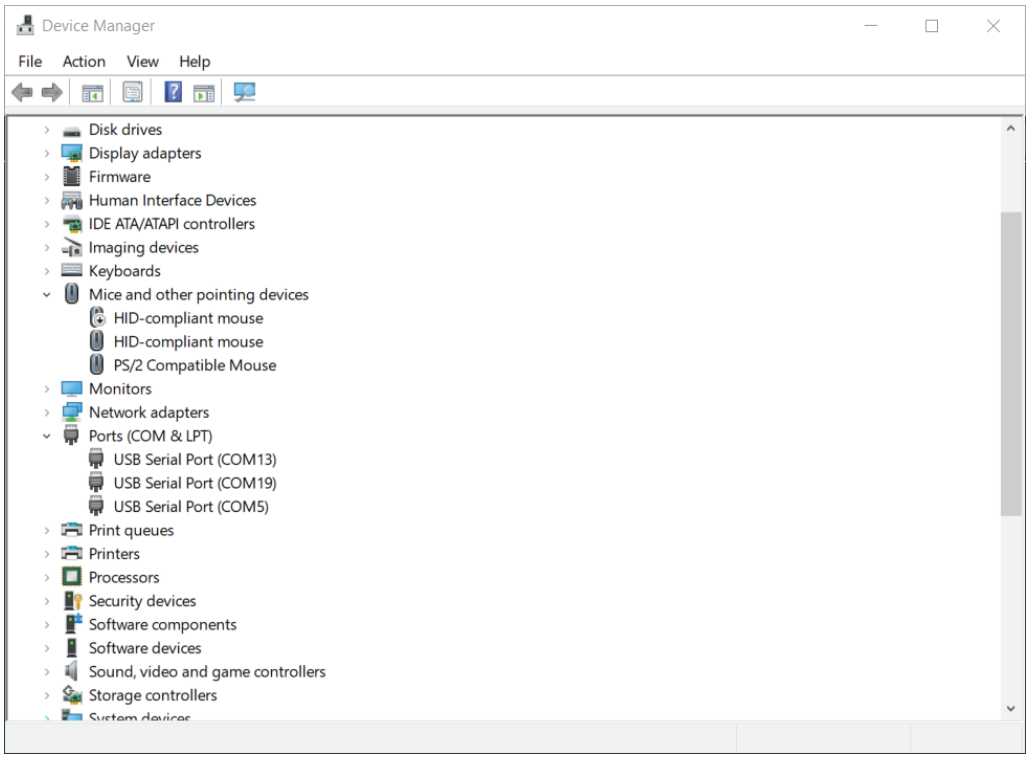


Figure 2.
Device Manager

Download the *Geolux Instrument Configurator* application from www.geolux-radars.com – go to *Products*, select *SmartObserverPlus*, and scroll to the bottom of the page to the *Downloads* section where the download link is listed under *Software*.

Launch the *Geolux Instrument Configurator* application and click the *Connect* button. Clicking the Connect button opens a window where the user can select the *SmartObserverPlus* data logger.

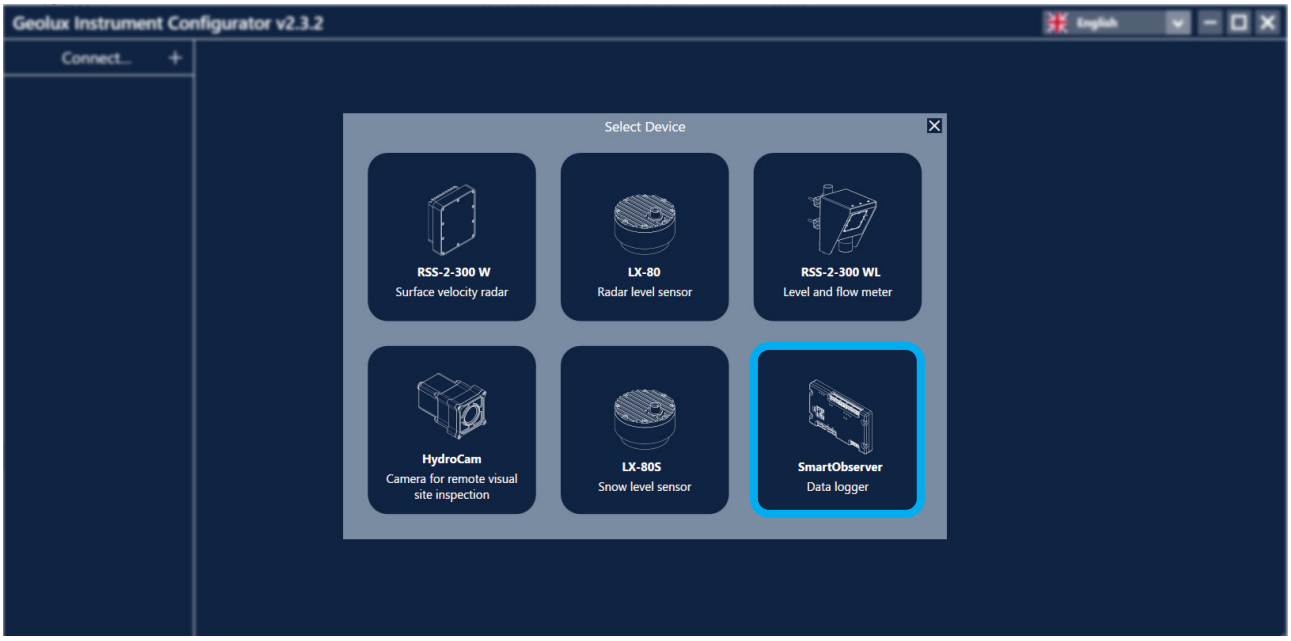


Figure 3. Geolux Instrument Configurator - Select Device Window

Next, select the appropriate COM port and click the *Connect* button.

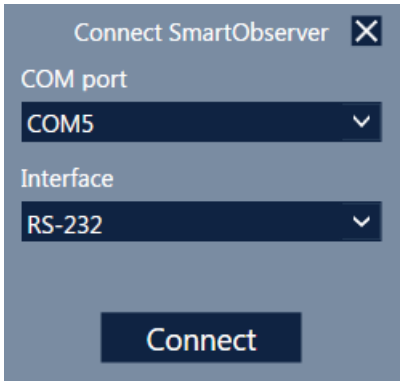


Figure 4. Select COM port

Once connected, the application will automatically switch to the *Settings* tab that is used for device setup.

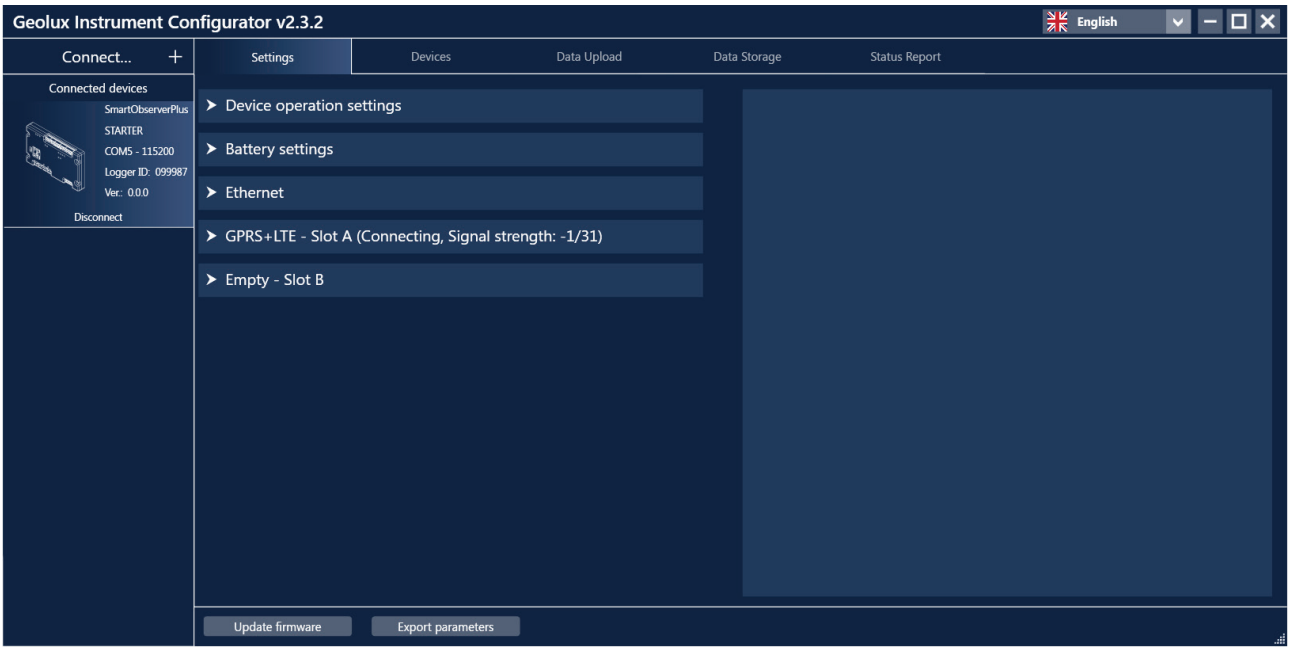


Figure 5. Geolux Instrument Configurator - Settings Tab

Setting Date and Time

The *SmartObserverPlus* data logger has an internal clock to maintain accurate time tracking. Set the correct date and time by clicking the *Set to computer time* button under the *Device operation settings* section in the *Settings* tab. The data logger keeps time in UTC, but for user convenience, the time displayed in the *Geolux Instrument Configurator* is converted to the local time based on the computer's date and time settings.

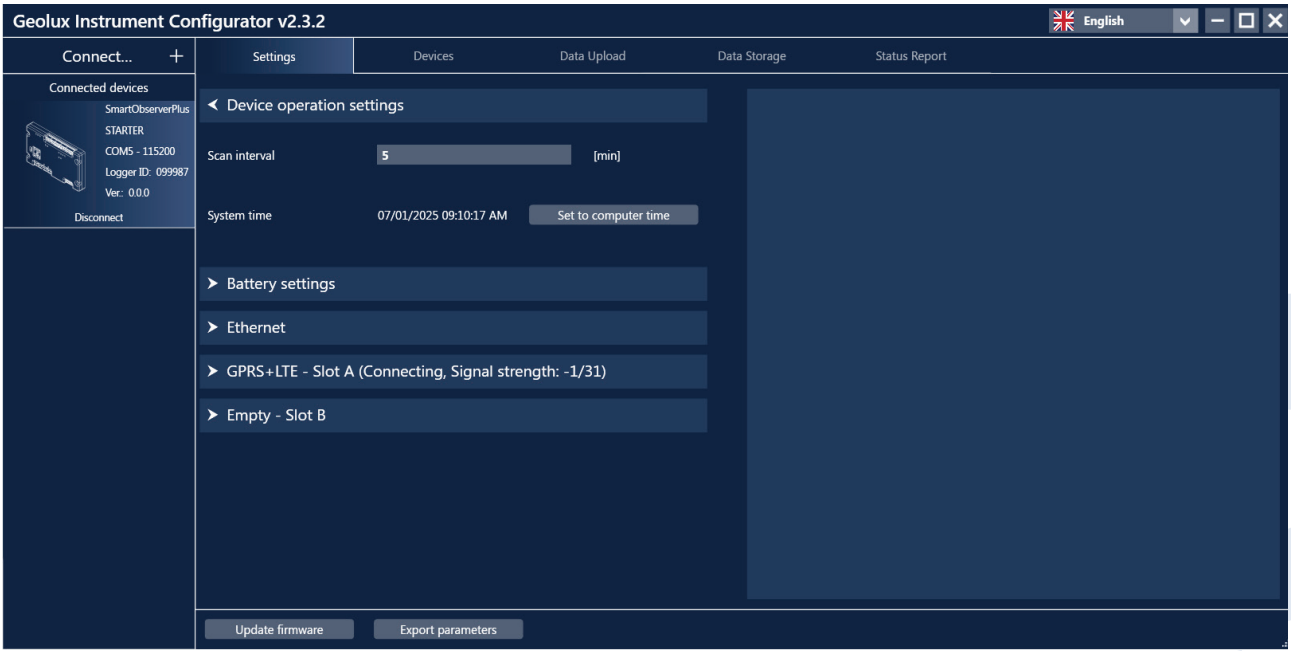


Figure 6. Geolux Instrument Configurator - Set to Computer Time

Configuring Ethernet Network

The *Ethernet* section of the *Settings* tab allows the user to configure the device's ethernet network settings. The user can assign the data logger's IP address automatically or manually. When using automatic (DHCP) address assignment, the data logger will connect to a router and automatically obtain an IP address.

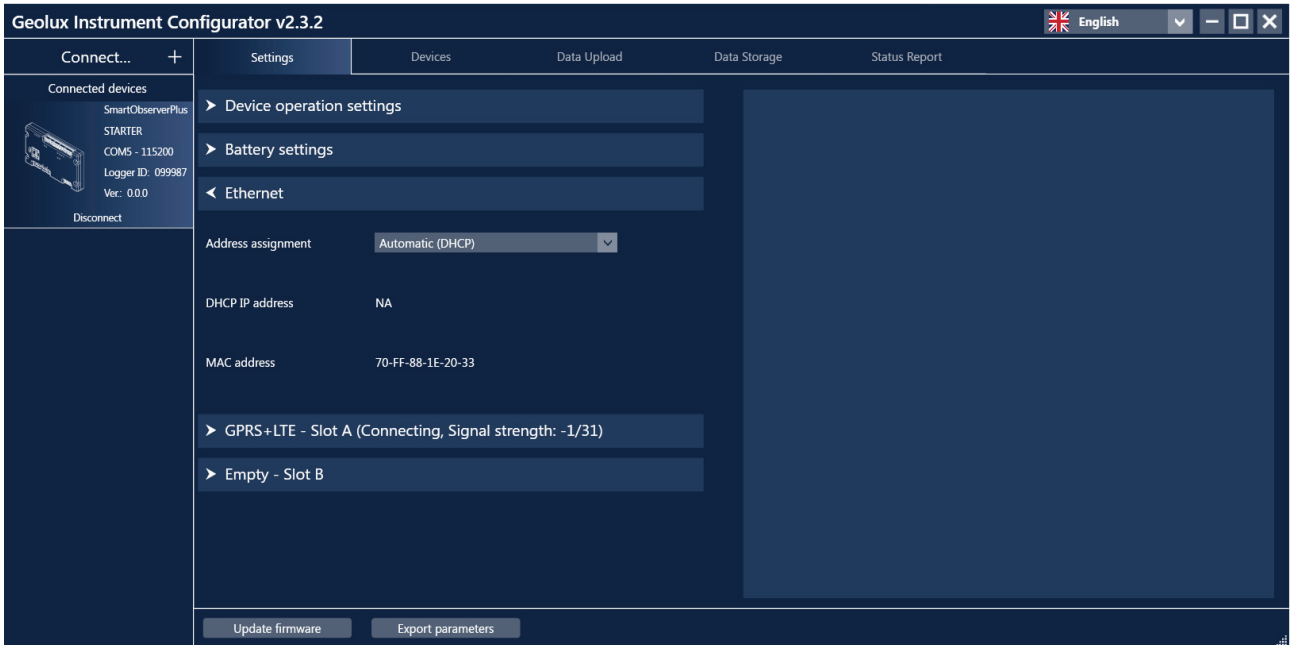


Figure 7. Geolux Instrument Configurator - Automatic Address Assignment

If using a static IP address, the user will need to change the *Address assignment* to *Manual* and enter the required IP address, subnet mask, gateway, and DNS details. The user must ensure the chosen settings align with the network requirements to establish a proper connection.

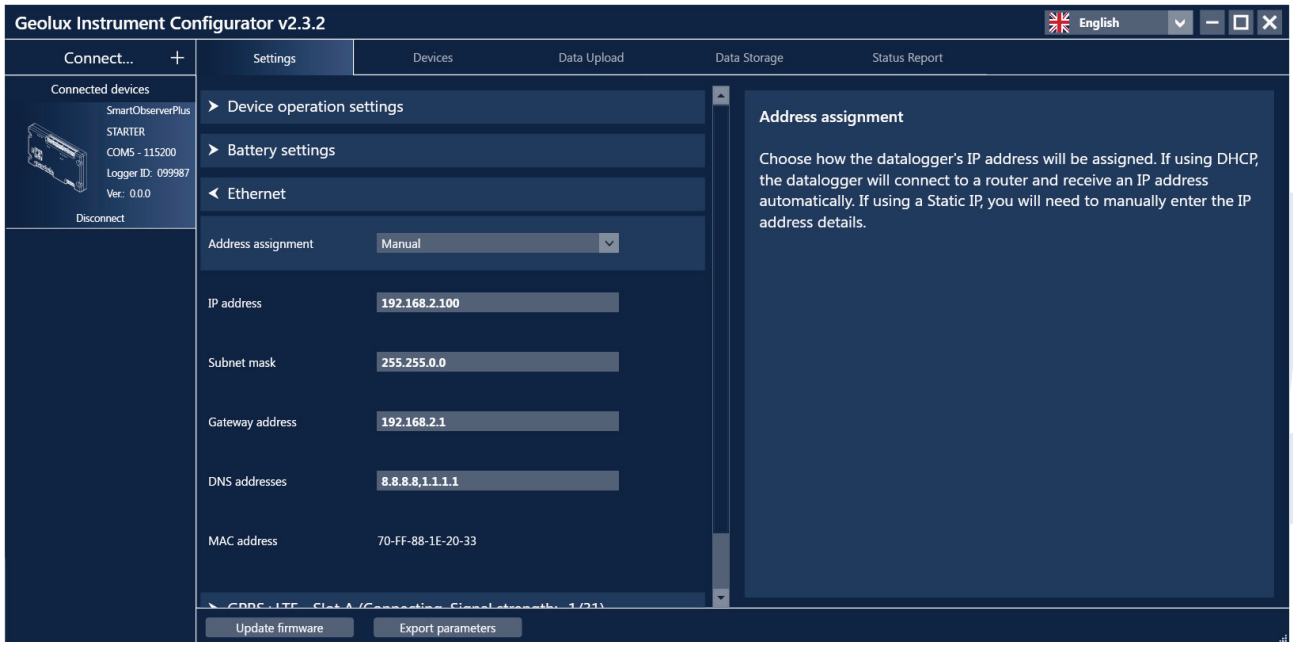


Figure 8. Geolux Instrument Configurator - Manual Address Assignment

Configuring the Modem

The last two sections of the *Settings* tab (*Slot A* and/or *Slot B*) are used to configure the modem settings. Depending on the modem's wireless communication type, the user needs to configure the needed modem settings as per the mobile operator's requirements.

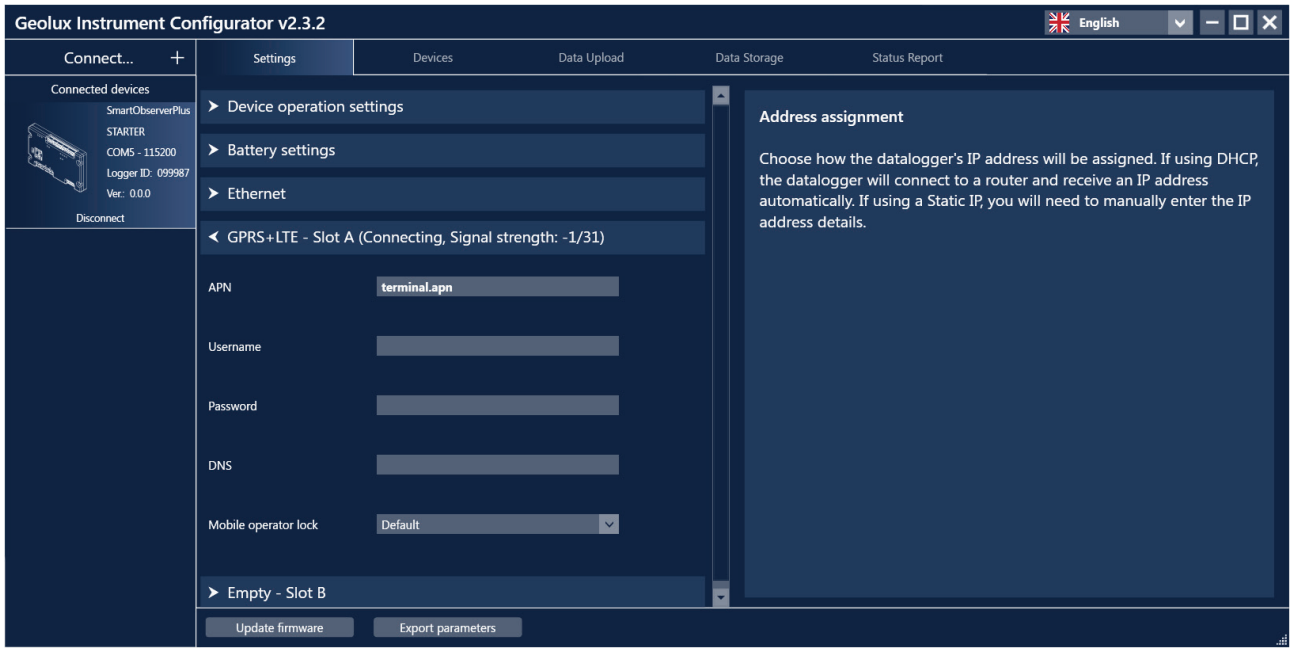


Figure 9. Geolux Instrument Configurator - Modem Settings

APN

The APN is the gateway between your mobile network and the internet, allowing your modem to establish a data connection. It must be set correctly based on your mobile provider's specifications.

Username & Password

Some mobile networks require a username and/or password for authentication to access the internet. These credentials are typically provided by your carrier if needed.

Mobile Operator Lock

A mobile operator lock restricts the modem to work only with a specific network provider.

6 Connecting Sensors

Connecting a Sensor consists of two steps:

1. **Physical connection**
   - Interface connection: Connect the sensor to the appropriate interface connector on the SmartObserverPlus (e.g. Modbus, SDI-12, Analog, Digital)
   - Power Connection (Optional): if the sensor requires external power, connect it to a suitable power output port on the SmartObserverPlus
2. **Software Configuration**
   - Add a Sensor to Instrument configuration using Geolux Instrument Configurator application
   - Add Measurements for Instrument

The *Devices* tab allows the user to configure instruments that are connected to the *SmartObserverPlus* data logger. New instruments can be added manually by clicking the *Add new instrument* button. The user will be prompted to configure the connection parameters as well as the instrument measurements, i.e. the values that an instrument can report. Alternatively, the user can import instruments from an .xml file by clicking the *Import instrument* button and selecting the appropriate file.

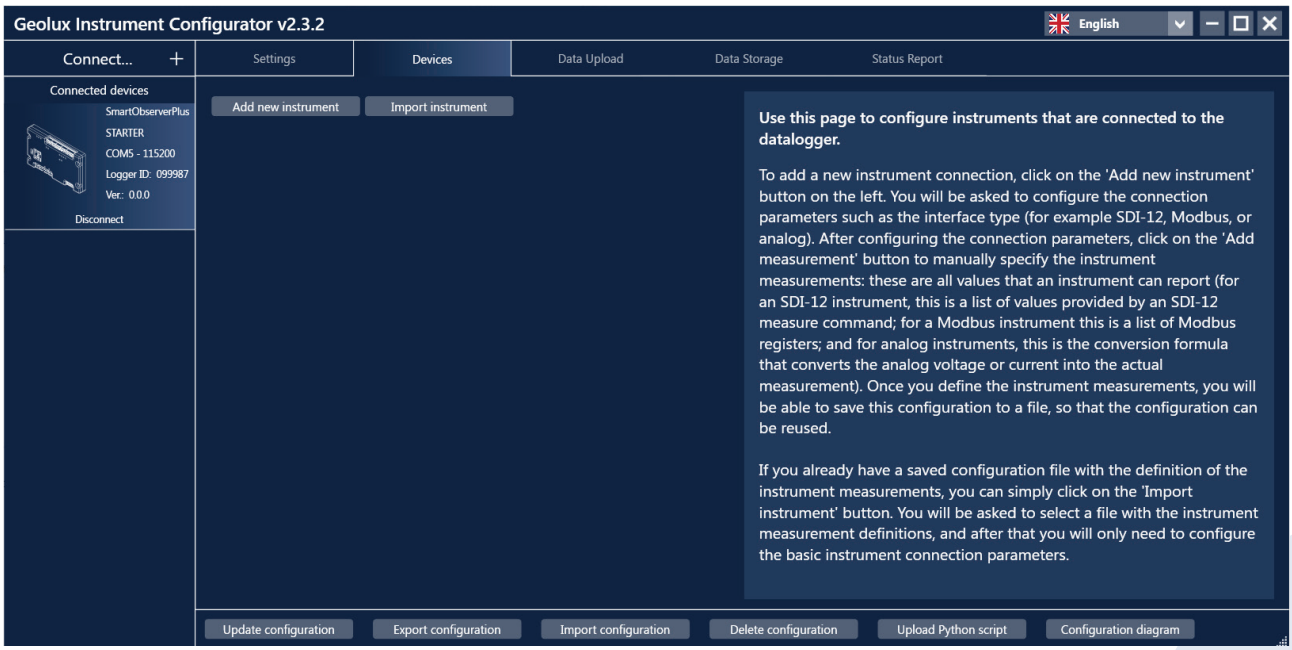


Figure 10. Geolux Instrument Configurator - Devices Tab

Once the instruments are defined, they are added to a list of instruments in the *Devices* tab (Figure 10). There the user can view the instrument's properties, delete or export the instrument, and add measurements to the instrument. Adding measurements to instruments will be explained in chapters 2.1. to 2.6.

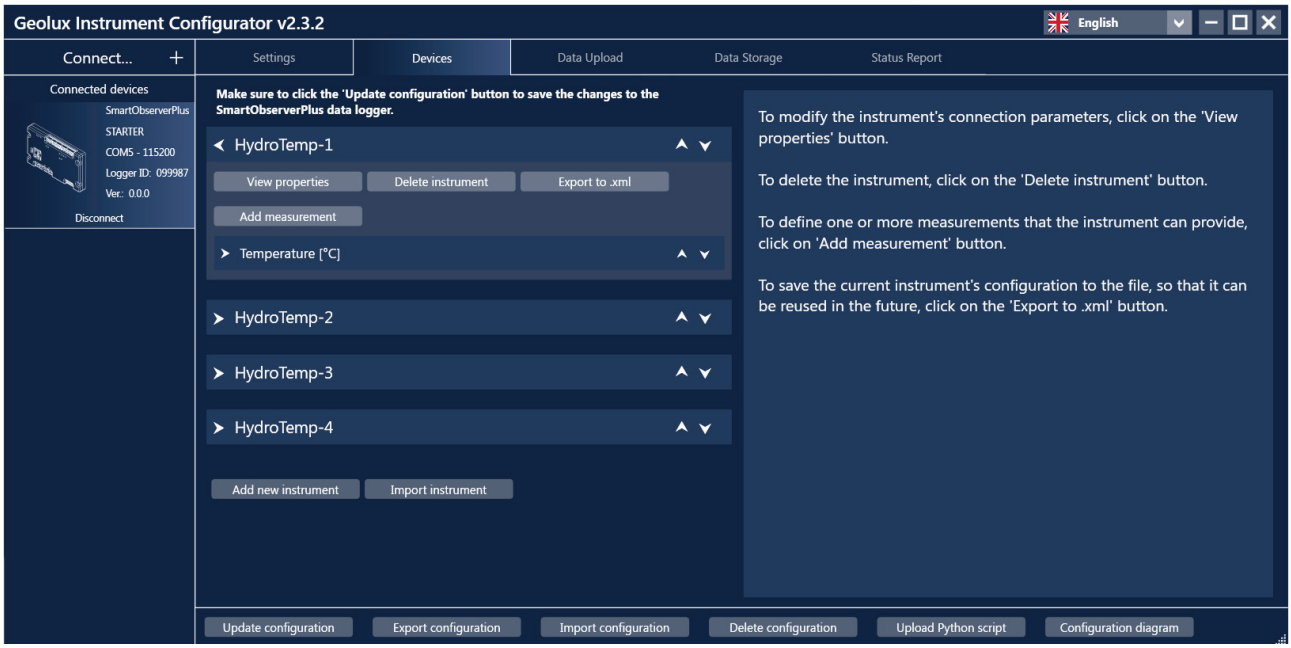


Figure 11. Geolux Instrument Configurator - Instruments List

It is possible to export each instrument separately by clicking the *Export instrument* button belonging to the selected instrument or export the entire configuration by clicking the *Export configuration* button in the bottom part of the *Devices* tab (see chapter 2.7). The exported instruments and configurations can be imported later when needed.

Once the configuration is defined or imported, the Update configuration button must be clicked to save the configuration to the data logger!

6.1. Connecting **Modbus Sensors**

Making a physical connection

Connecting a Modbus Sensor is possible to any of the three available RS-485 ports (D1, D2, D3) as shown in Figure 11.

It is good practice to make all sensor connections with the SmartObserverPlus powered off.

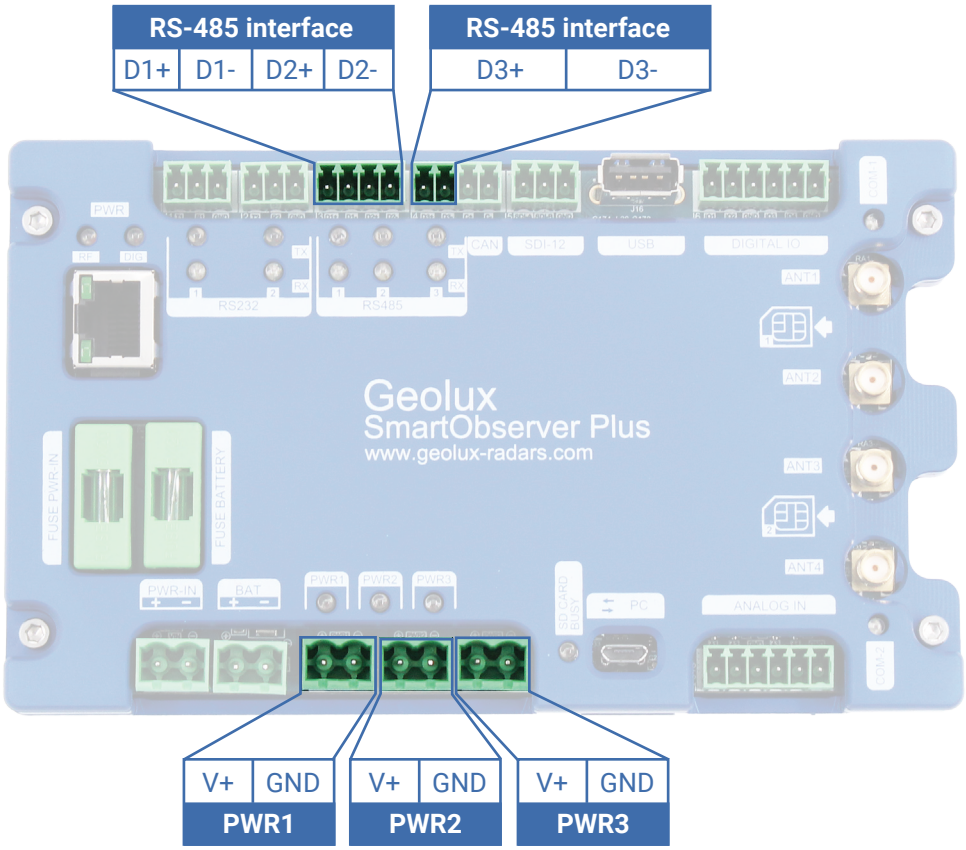


Figure 12. Modbus/RS-485 interface – ports & pins

Connect the wire of **RS-485 A** signal from Sensor to **D+** pin of RS-485 port on SmartObserverPlus.

Connect the wire of **RS-485 B** signal from Sensor to **D-** pin of RS-485 port (same port as D+).

Table 2. Ports & pins - Modbus/RS-485 wiring

Modbus/RS-485 Port	Port pin	Sensor Wire lead
1	D1+	RS-485 A or Tx+/D+
	D1-	RS-485 B or Tx-/D-
2	D2+	RS-485 A or Tx+/D+
	D2-	RS-485 B or Tx-/D-
3	D3+	RS-485 A or Tx+/D+
	D3-	RS-485 B or Tx-/D-

When connecting the RS485 interface the following rules must be followed:

1. Use twisted pair cable.
2. Use shielded cable.
3. Ensure that all units have a proper connection to safety ground (earth) reference to the same potential.
4. The last Sensor at end of the network must be terminated with a 120 Ohm resistor between A and B.

Connect the Sensor to one of the three available Power Outputs (PWR1, PWR2, PWR 3, shown in Figure 11). Connect the Sensor's positive supply wire to **PWR+** and its ground wire to **PWR-.**

Adding & configuring an Instrument

Fill in the *Instrument type* and *Instrument description* fields and select the appropriate Power connection – the output power line from the data logger to which the instrument is connected. Next, select *Modbus/RS-485* as the Data interface and configure the required connection parameters. Finally, configure the *Measurement duration* which specifies how long the instrument should remain powered before the data logger reads its measured values.

Figure 13. Geolux Instrument Configurator - Add Modbus Instrument

Clicking the *Add* button in the *Add instrument window* adds the instrument to the list of instruments displayed in the *Devices* tab (Figure 10). Once the instrument is added to the list, the user can view and change its properties, delete or export the instrument, and add measurements to the instrument by clicking the *Add measurement* button.

The *Add Measurement* button opens a window where the user is prompted to define a Modbus instrument's measurement. The user must enter the *Measurement name*, define the *Quantity and unit* by clicking the *Define* button (see chapter 2.5), and select the Modbus *Register type* and *Register address*, as well as the *Data type*.

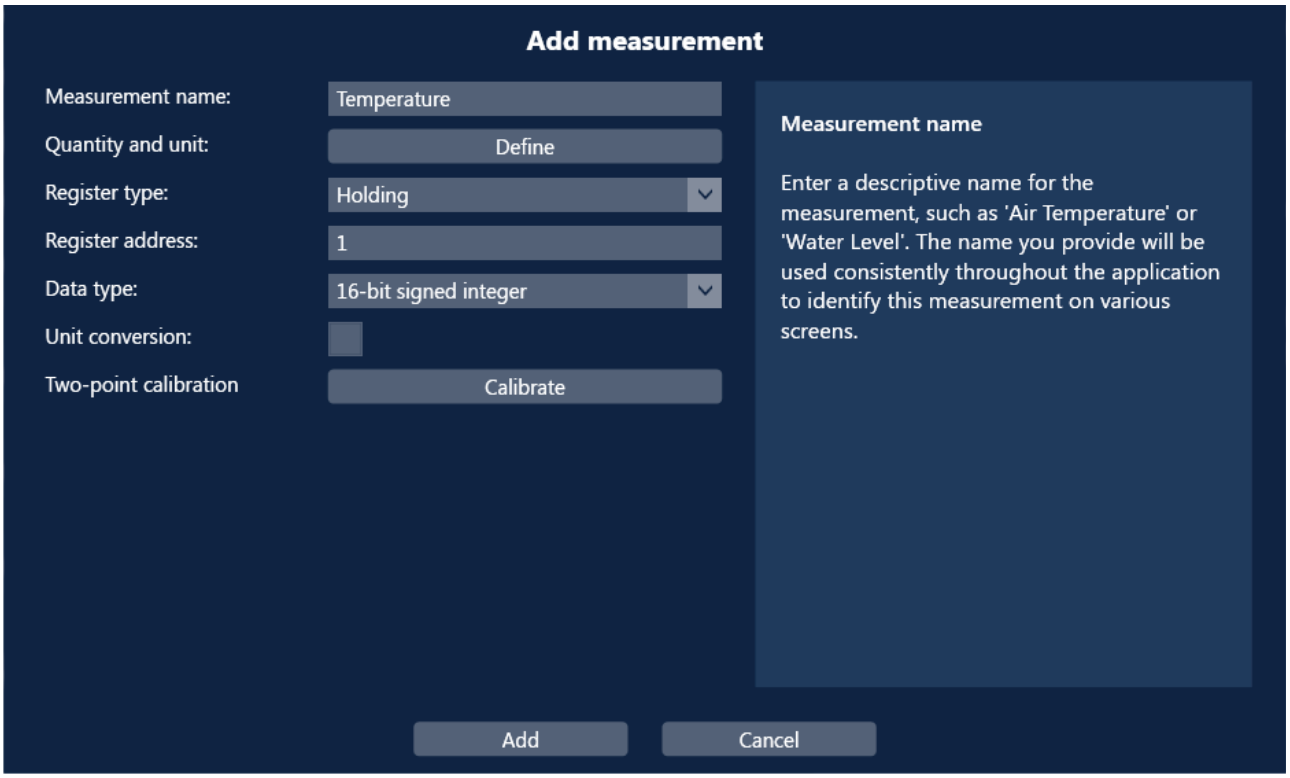


Figure 14. Geolux Instrument Configurator - Add Modbus Measurement

If unit conversion is required, the user can check the *Unit conversion* checkbox and manually enter the *Multiplier* and *Offset* values. Alternatively, the user can use *Two-point calibration* by clicking the *Calibrate* button (see chapter 2.6). This feature automatically calculates the multiplier and offset based on two sets of reported and measured values.

6.2. Connecting **SDI-12 Sensors**

Making a physical connection

Connecting an SDI-12 Sensor is possible to any of the two available SDI-12 ports (SDI-1 & SDI-2) as shown in Figure 14.

It is good practice to make all sensor connections with the SmartObserverPlus powered off.

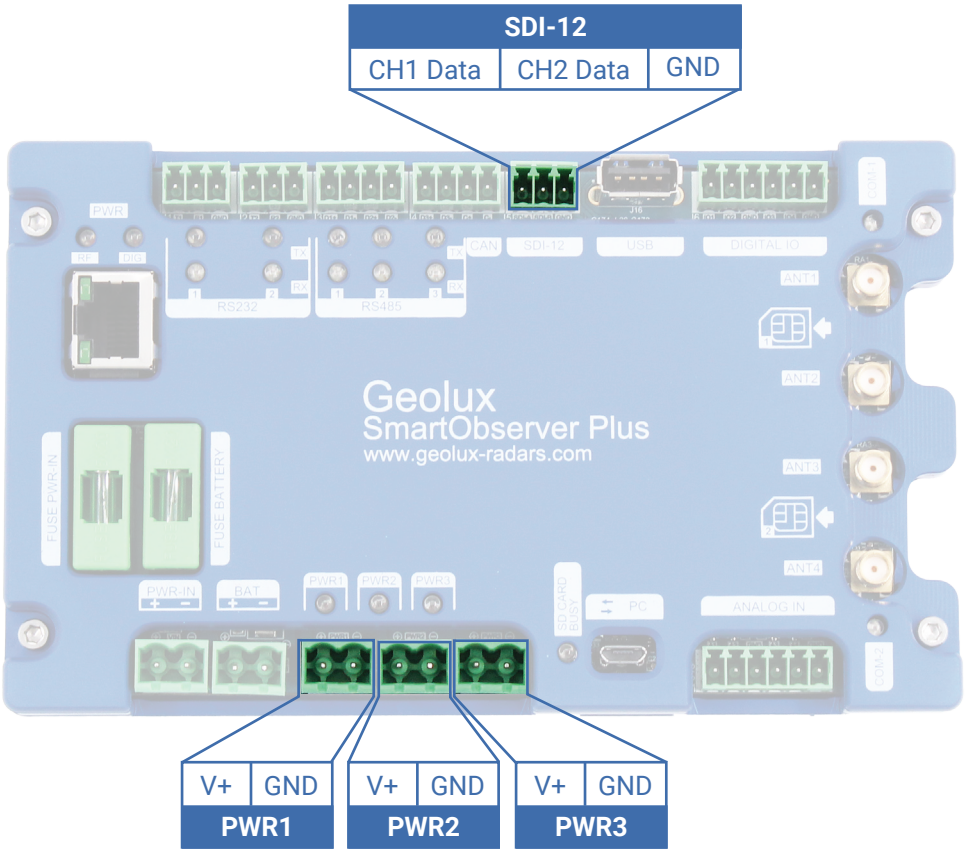


Figure 15.
SDI-12 interface – ports & pins

Connect the wire of **Serial Data Line** signal from Sensor to **SDI-x** pin of SDI-12 port on SmartObserverPlus.

Connect the wire of **Ground Line** signal from Sensor to **GND** pin of SDI-12 port.

Connect the Sensor to one of the three available Power Outputs (PWR1, PWR2, PWR 3, shown inFigure 14). Connect the Sensor's positive supply wire to **PWR+** and its ground wire to **PWR-.**

Adding & configuring an Instrument

Fill in the *Instrument type* and *Instrument description* fields and select the appropriate Power connection – the output power line from the data logger to which the instrument is connected. Next, select *SDI-12* as the *Data interface* and configure the required connection parameters.

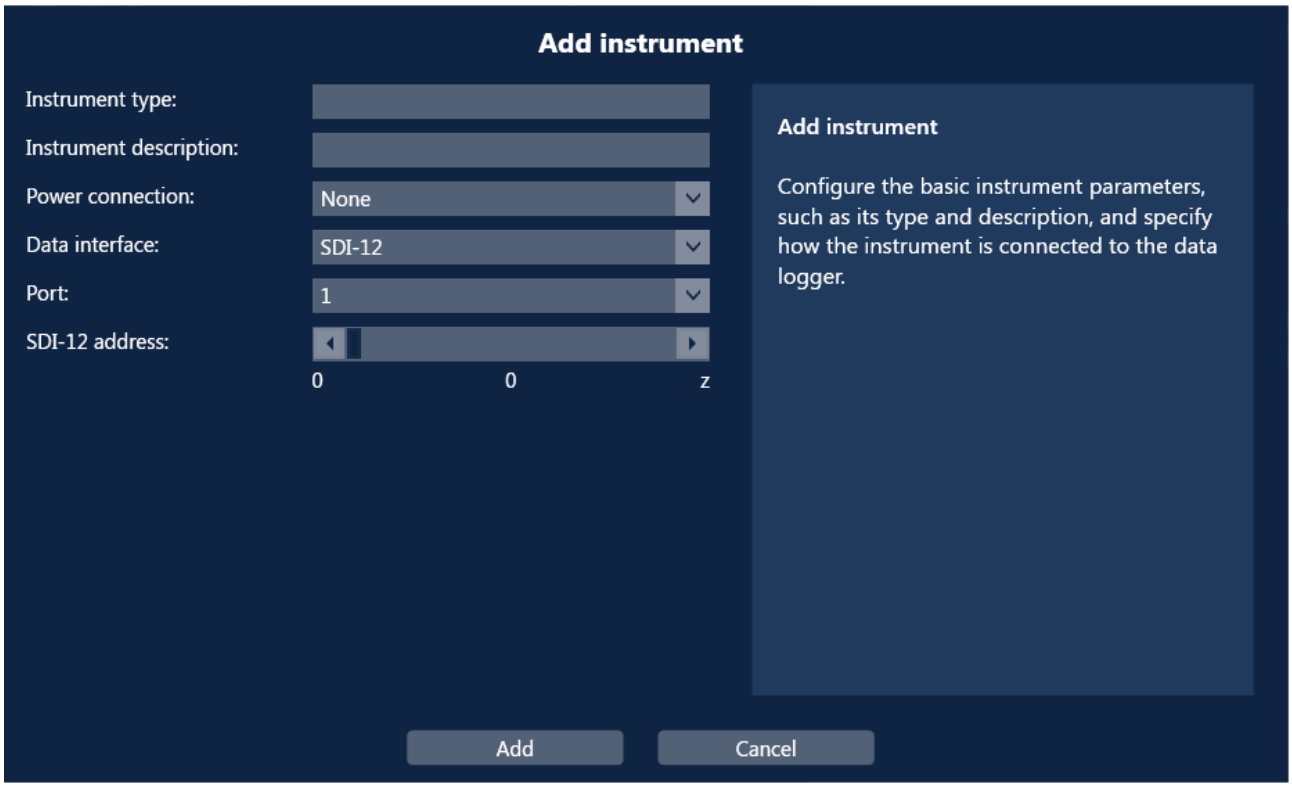


Figure 16. Geolux Instrument Configurator - Add SDI-12 Instrument

Clicking the *Add* button in the *Add instrument window* adds the instrument to the list of instruments displayed in the *Devices* tab (Figure 10). Once the instrument is added to the list, the user can view and change its properties, delete or export the instrument, and add measurements to the instrument by clicking the *Add measurement* button.

The *Add Measurement* button opens a window where the user is prompted to define a measurement for the SDI-12 instrument. The user must enter the *Measurement name*, define the *Quantity and unit* by clicking the *Define* button (see chapter 2.5), and select the *SDI-12 command* that the data logger should issue to obtain measurements from the instrument. Additionally, the *Response field index* must be specified because the SDI-12 protocol returns a list of measurement values after a command is issued, and this parameter is used to identify which value in the response corresponds to the measurement type being configured.

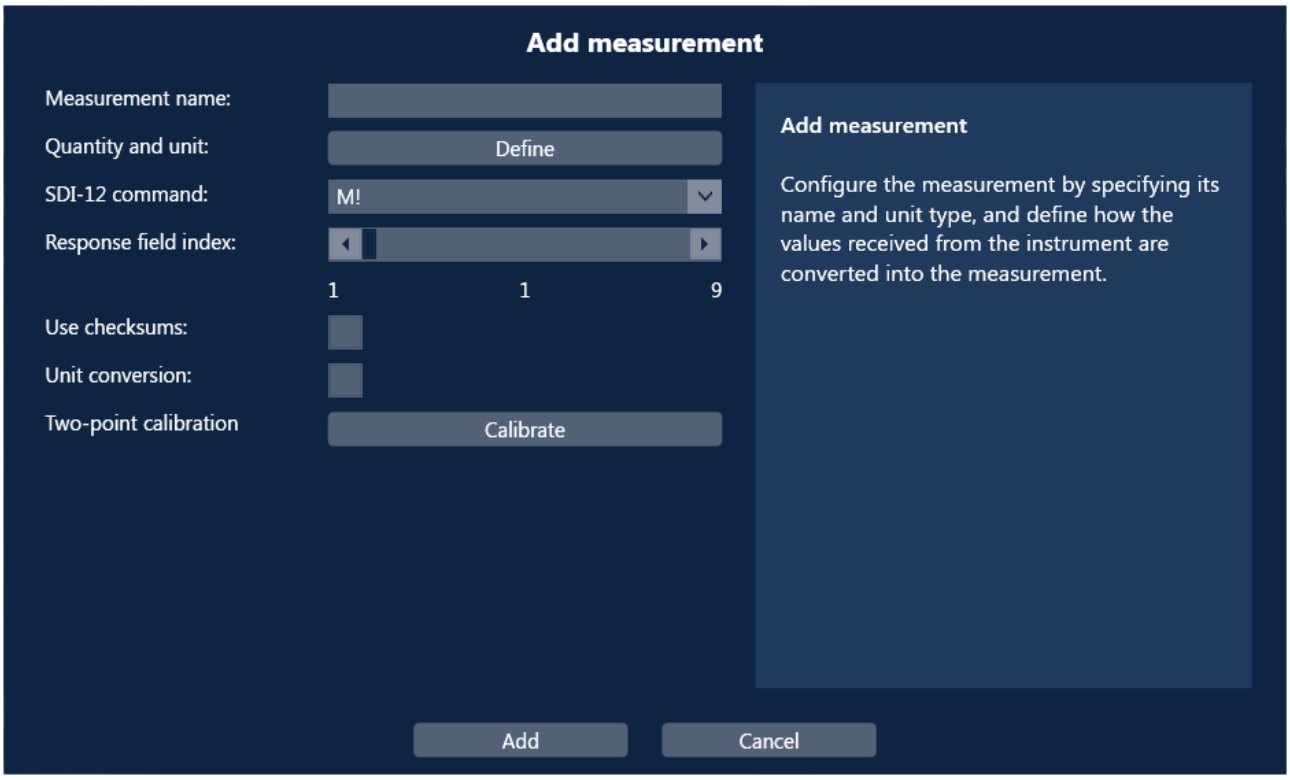


Figure 17. Geolux Instrument Configurator - Add SDI-12 Measurement

In case the SDI-12 instrument supports checksums, the user can check the *Use cheksums* checkbox. This ensures that SDI-12 commands include checksums at the end of responses, improving data transfer reliability.

If unit conversion is required, the user can check the *Unit conversion* checkbox and manually enter the *Multiplier* and *Offset* values. Alternatively, the user can use *Two-point calibration* by clicking the *Calibrate* button (see chapter 2.6). This feature automatically calculates the multiplier and offset based on two sets of reported and measured values.

6.3. Connecting **Sensors to Analog Inputs**

To connect a sensor to the analog inputs of the data logger, fill in the *Instrument type* and *Instrument description* fields and select the appropriate *Power connection* – the output power line from the data logger to which the instrument is connected. Next, select *Analog IN* as the *Data interface* and select the appropriate port.

Select the *Measurement type* to indicate whether the analog instrument connected to the analog input provides a voltage or current (4–20 mA) output. It is essential to ensure that the instrument's output type matches the configuration on the data logger. Configuring a voltage-output instrument as a current-output instrument can cause over-current, potentially damaging the data logger's analog input. If the instrument provides a voltage output, select the appropriate *Voltage range* to ensure optimal measurement resolution.

Finally, configure the *Measurement averaging*, which determines how long the data logger should collect analog measurements before calculating the average value, and specify the *Averaging type*, which defines the method the data logger will use to compute the average value over the selected duration.

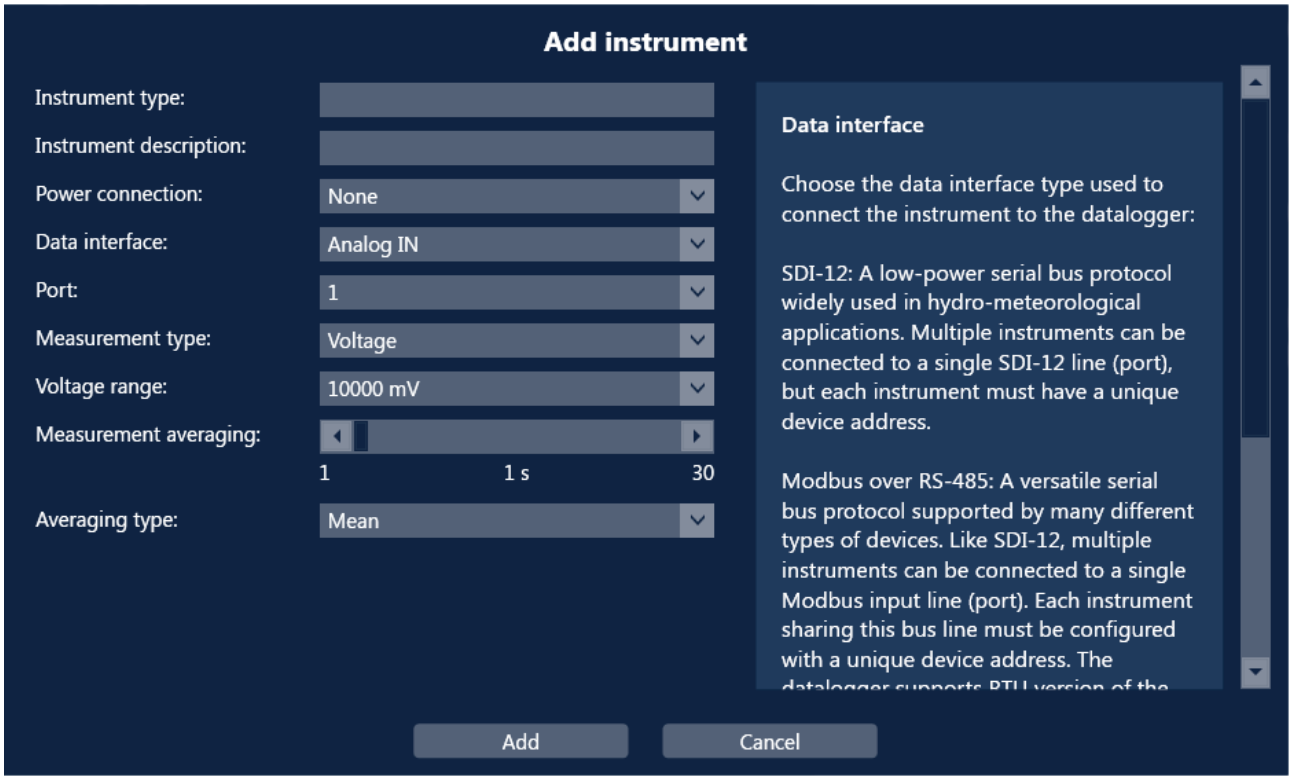


Figure 18. Geolux Instrument Configurator - Add Analog IN Instrument

Clicking the *Add* button in the *Add instrument window* adds the instrument to the list of instruments displayed in the *Devices* tab (Figure 10). Once the instrument is added to the list, the user can view and change its properties, delete or export the instrument, and add measurements to the instrument by clicking the *Add measurement* button.

The *Add Measurement* button opens a window where the user is prompted to define a measurement for the analog instrument. The user must enter the *Measurement name*, define the *Quantity and unit* by clicking the *Define* button (see chapter 2.5), and enter the *Multiplier* and *Offset* values which determine how the values received from the instrument are converted into the measurement. The multiplier and offset values can be calculated using the *Two-point calibration* by clicking the *Calibrate* button (see chapter 2.6). This feature automatically calculates the multiplier and offset based on two sets of reported and measured values.

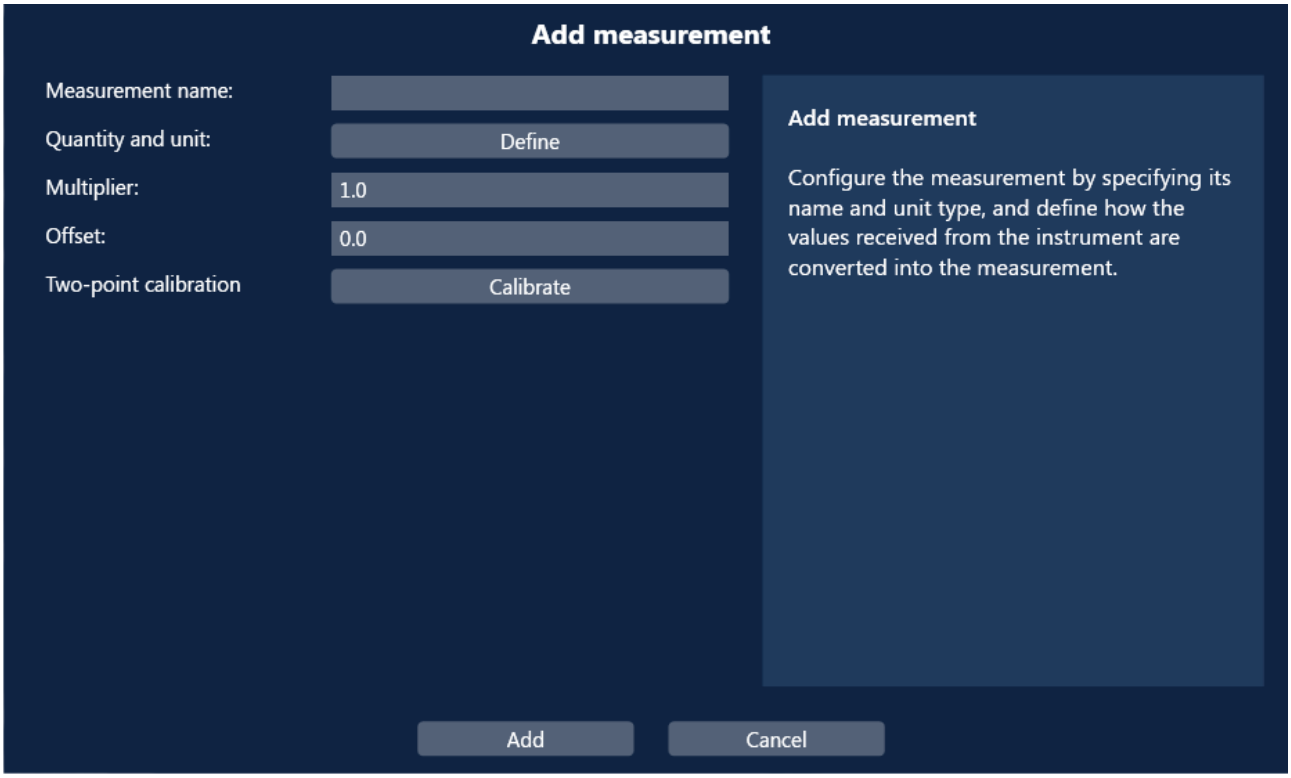


Figure 19. Geolux Instrument Configurator – Add Analog IN Measurement

6.4. Connecting **Sensors to Digital Inputs**

To connect a sensor to the digital inputs of the data logger, fill in the *Instrument type* and *Instrument description* fields and select the appropriate *Power connection* – the output power line from the data logger to which the instrument is connected. Next, select *Digital IN* as the *Data interface* and select the appropriate port. Finally, select the appropriate *Measurement mode* for the digital input based on the instrument type.

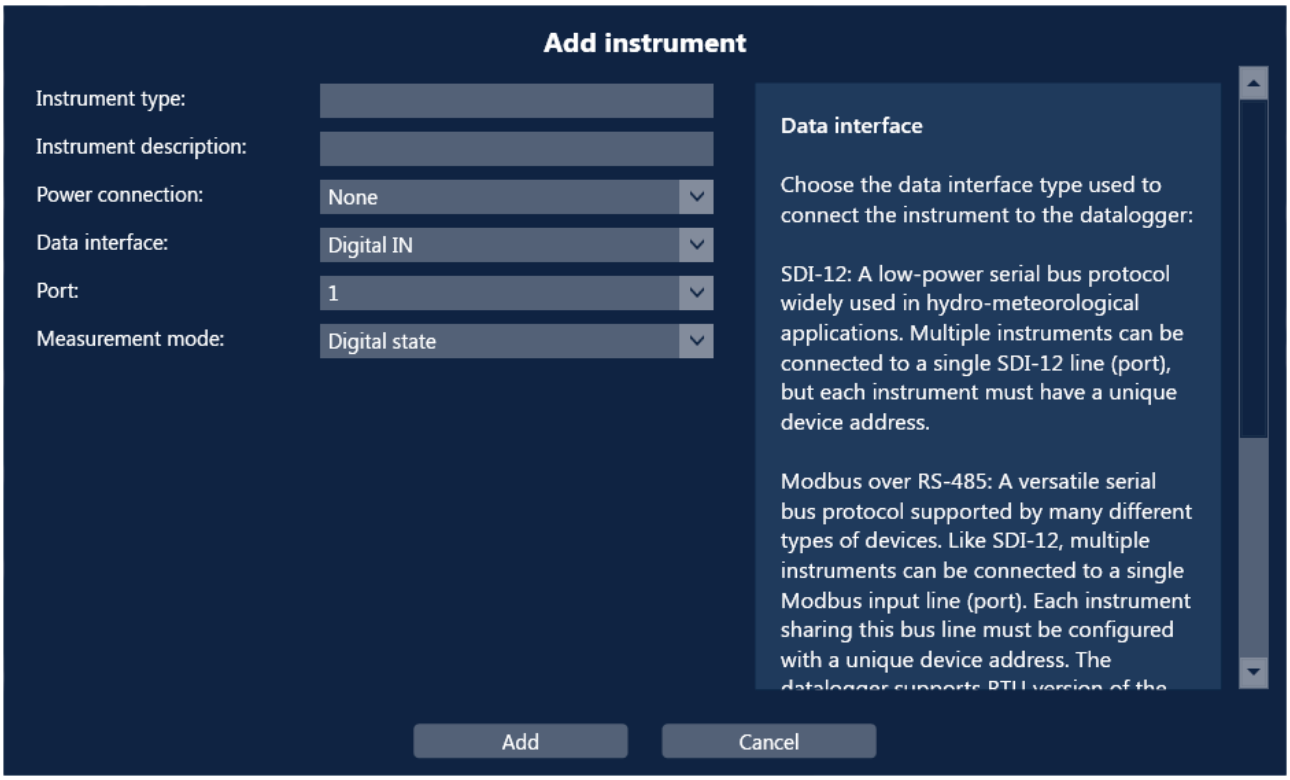


Figure 20. Geolux Instrument Configurator - Add Digital IN Instrument

Clicking the *Add* button in the *Add instrument window* adds the instrument to the list of instruments displayed in the *Devices* tab (Figure 10). Once the instrument is added to the list, the user can view and change its properties, delete or export the instrument, and add measurements to the instrument by clicking the *Add measurement* button.

The *Add Measurement* button opens a window where the user is prompted to define a measurement for the digital instrument. The user must enter the *Measurement name* and define the *Quantity and unit* by clicking the *Define* button (see chapter 2.5). Additionally, if the instrument's *Measurement mode* is set to *Pulse counter* or *Frequency counter* the user must enter the *Multiplier* value which determines how the values received from the instrument are converted into the measurement. The multiplier value can be calculated using the *Two-point calibration* by clicking the *Calibrate* button (see chapter 2.6). This feature automatically calculates the multiplier based on two sets of reported and measured values.

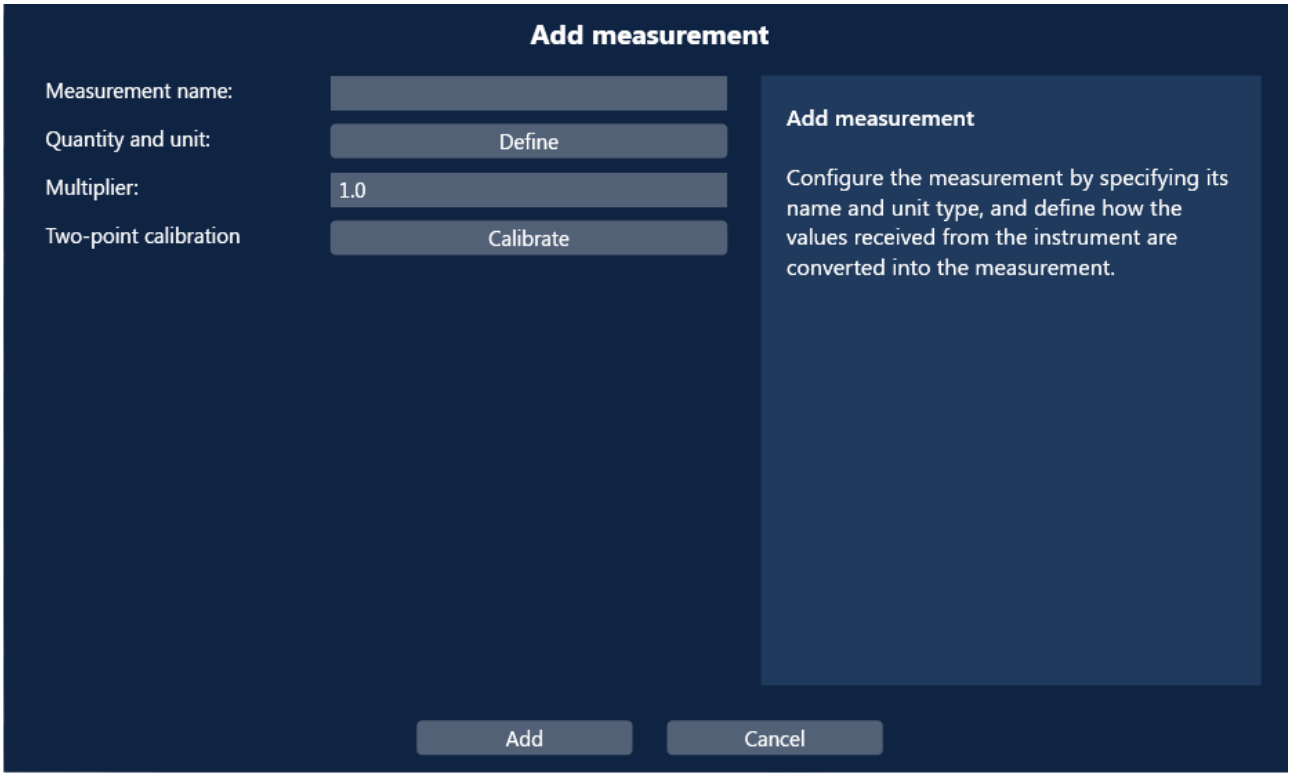


Figure 21. Geolux Instrument Configurator - Add Digital IN Measurement

6.5. Defining the **Quantity and Unit**

Clicking the *Define* button opens a window where the user can either select a predefined quantity or create a custom one. For convenience, quantities are organized into categories. Selecting a category filters the list to display only the quantities within that category.

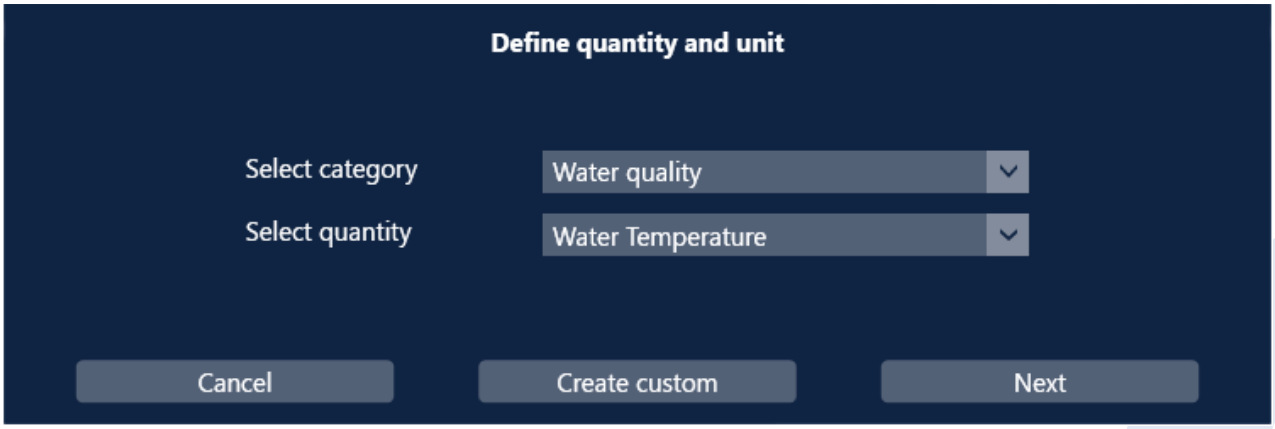


Figure 22. Geolux Instrument Configurator - Define Quantity and Unit Window

After selecting a quantity and clicking *Next*, the user is prompted to select a unit from a list of possible units in which the quantity can be expressed.

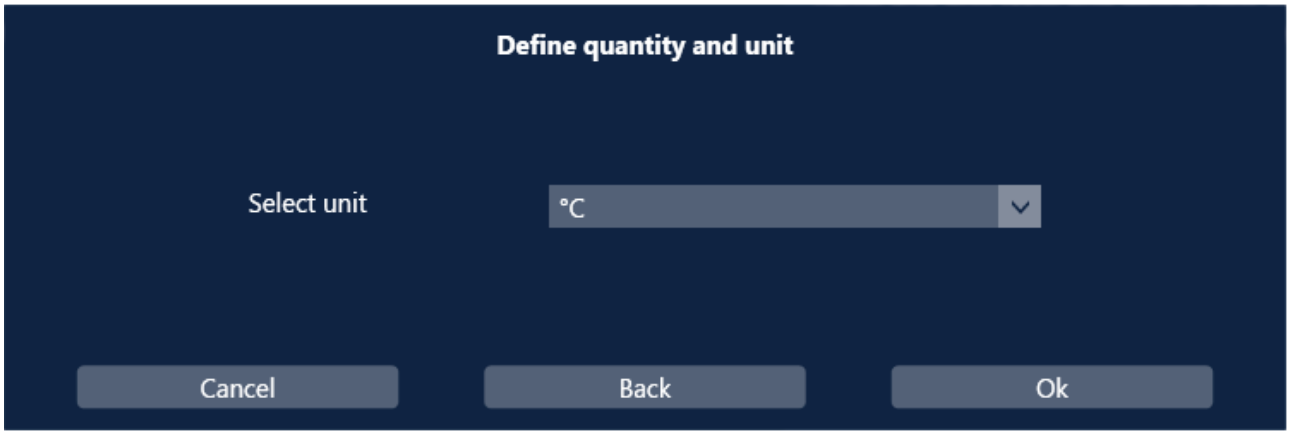


Figure 23. Geolux Instrument Configurator - Select Unit

To create a custom quantity, the user can click the *Create custom* button. This opens a window where the user must enter the *Measurement name* and *Measurement unit*.

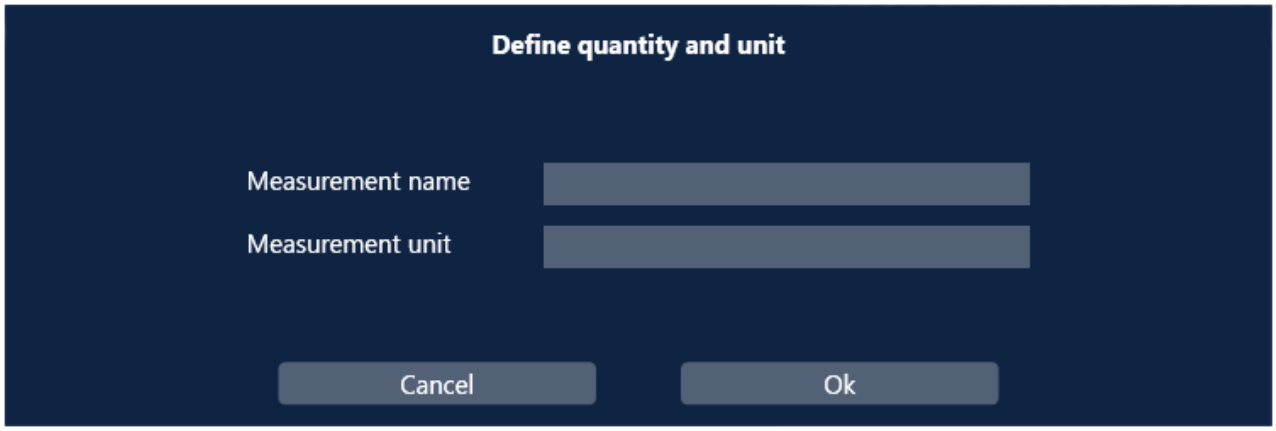


Figure 24. Geolux Instrument Configurator - Creat Custom Quantity

After selecting a predefined quantity or creating a custom one, clicking the *Ok* button closes the *Define quantity and unit window.* The selected quantity and unit will then appear on the button in the *Add measurement* window.

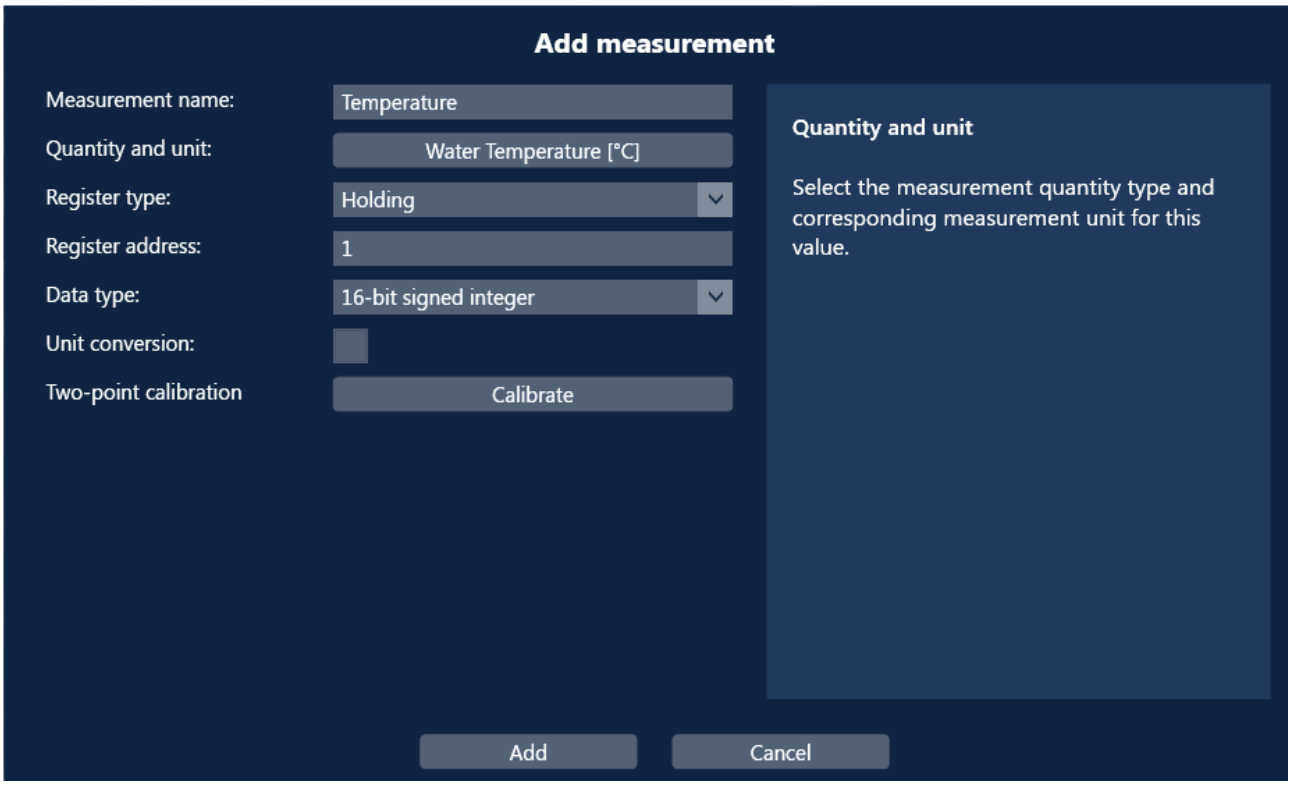


Figure 25. Geolux Instrument Configurator - Quanity and Unit Defined

6.6. Two-Point Calibration

Clicking the *Calibrate* button opens the *Two-point calibration* window where the user is prompted to input two points. For each point, the user must enter the value reported by the instrument and the corresponding actual value. Based on these inputs, the system automatically calculates the multiplier and offset required for accurate data conversion.

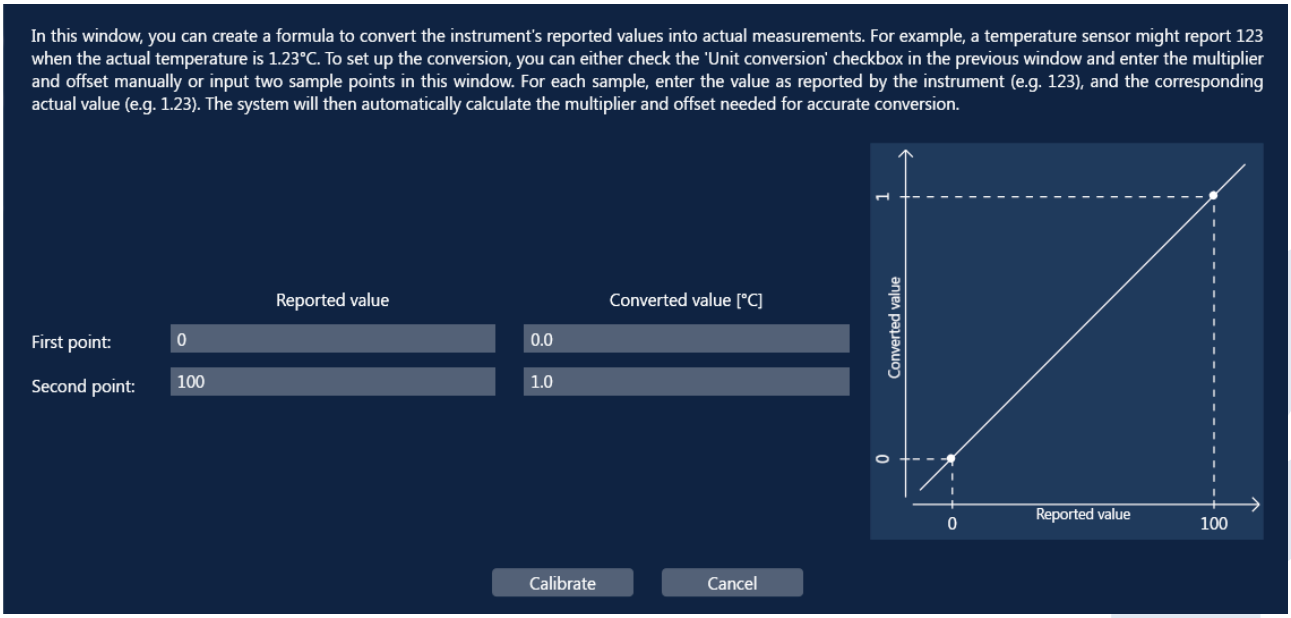


Figure 26. Geolux Instrument Configurator - Two-Point Calibration Window

Clicking the *Calibrate* button closes the *Two-point calibration* window, automatically selects the *Unit conversion* checkbox in the *Add measurement* window, and applies the calculated multiplier and offset values.

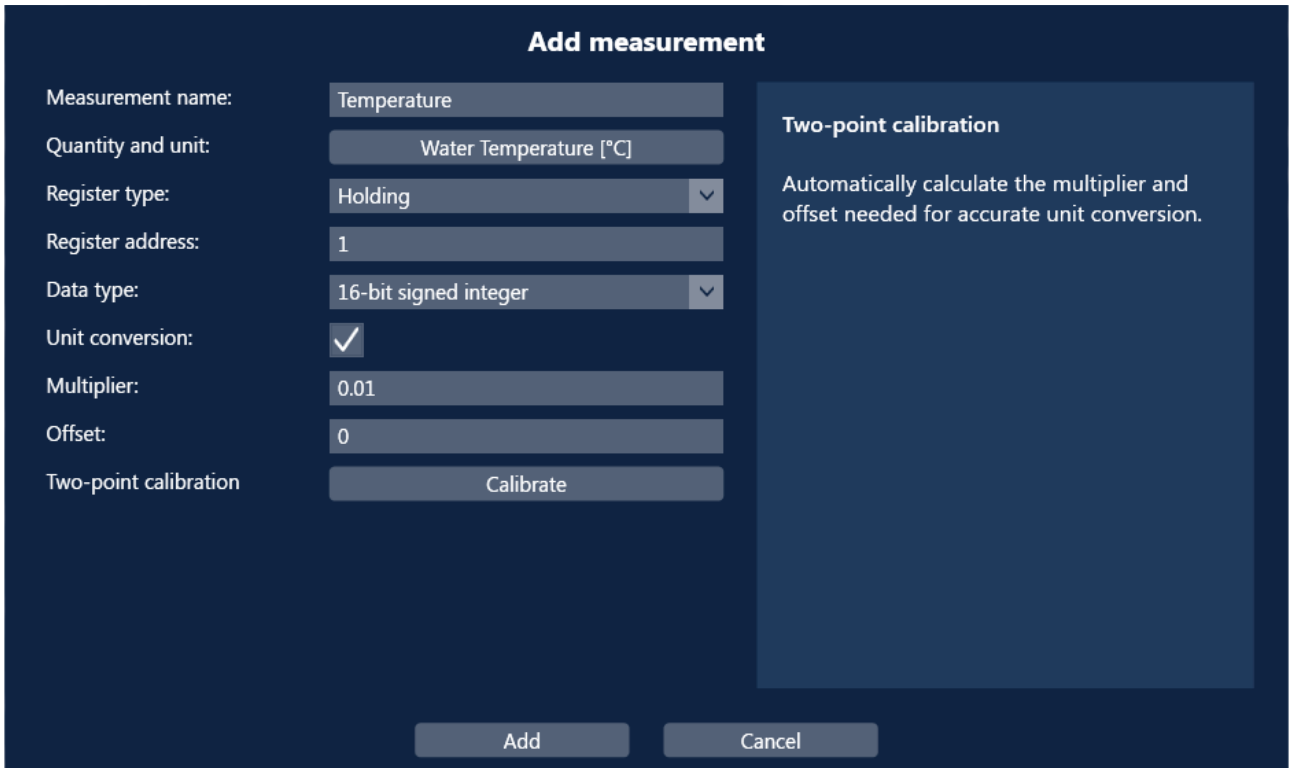


Figure 27. Geolux Instrument Configurator - Multiplier and Offset Calculated

6.7. Update / Export / Import / Delete Device Configuration

Once the instruments and their measurements have been defined the *Update configuration* button in the bottom part of the *Devices* tab must be clicked to save the configuration to the *SmartObserverPlus* data logger.

The configuration can also be saved to the PC by clicking the *Export configuration button*. The configuration is saved to the PC as an .xml file, which can be imported later if needed. To import a configuration, click the *Import configuration* button. This action clears the current list of devices in the *Devices* tab and replaces it with the instruments defined in the imported .xml file. After importing the configuration, the *Update configuration* button must be clicked to save the changes to the *SmartObserverPlus* data logger.

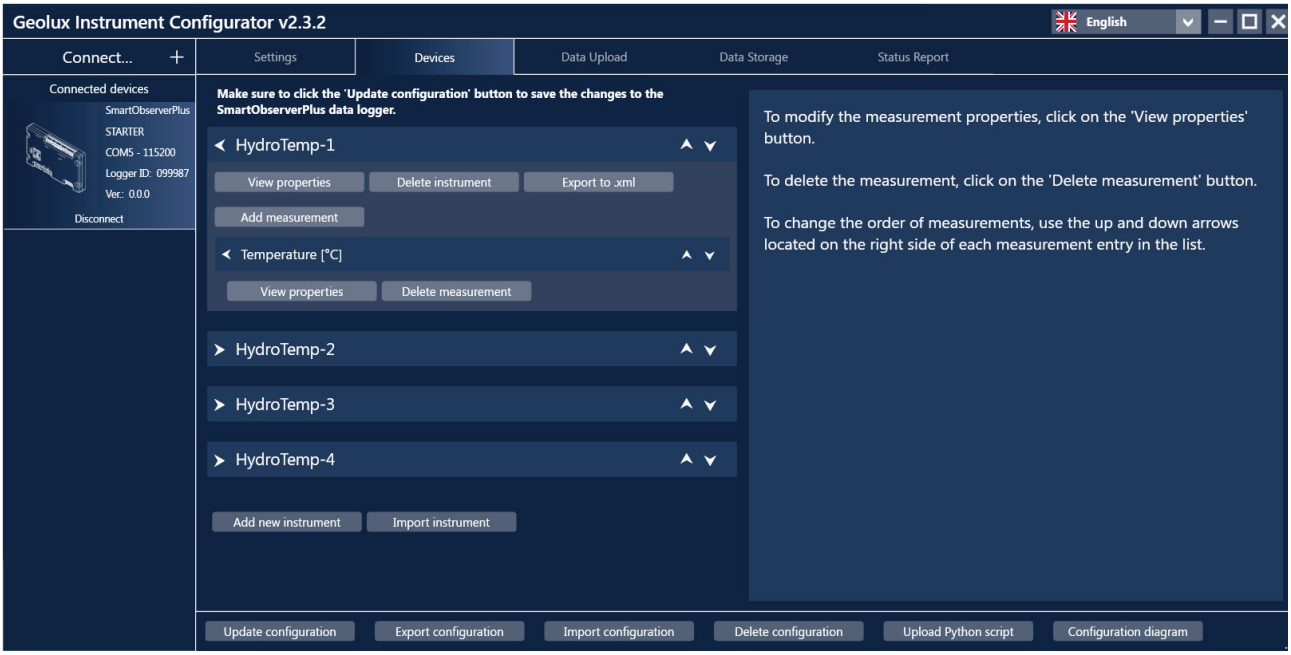


Figure 28. Geolux Instrument Configurator - Configuration Action Buttons

Clicking the *Delete configuration* button clears the list of devices in the *Devices* tab. To delete the configuration from the *SmartObserverPlus* data logger, click the *Update configuration* button after clearing the list.

The *View configuration diagram* button opens a window with a visual representation of how the devices in the list are connected to the data logger's power and data lines. This helps the user easily identify potential errors and make necessary adjustments to the instrument properties in the list of devices in the *Devices* tab.

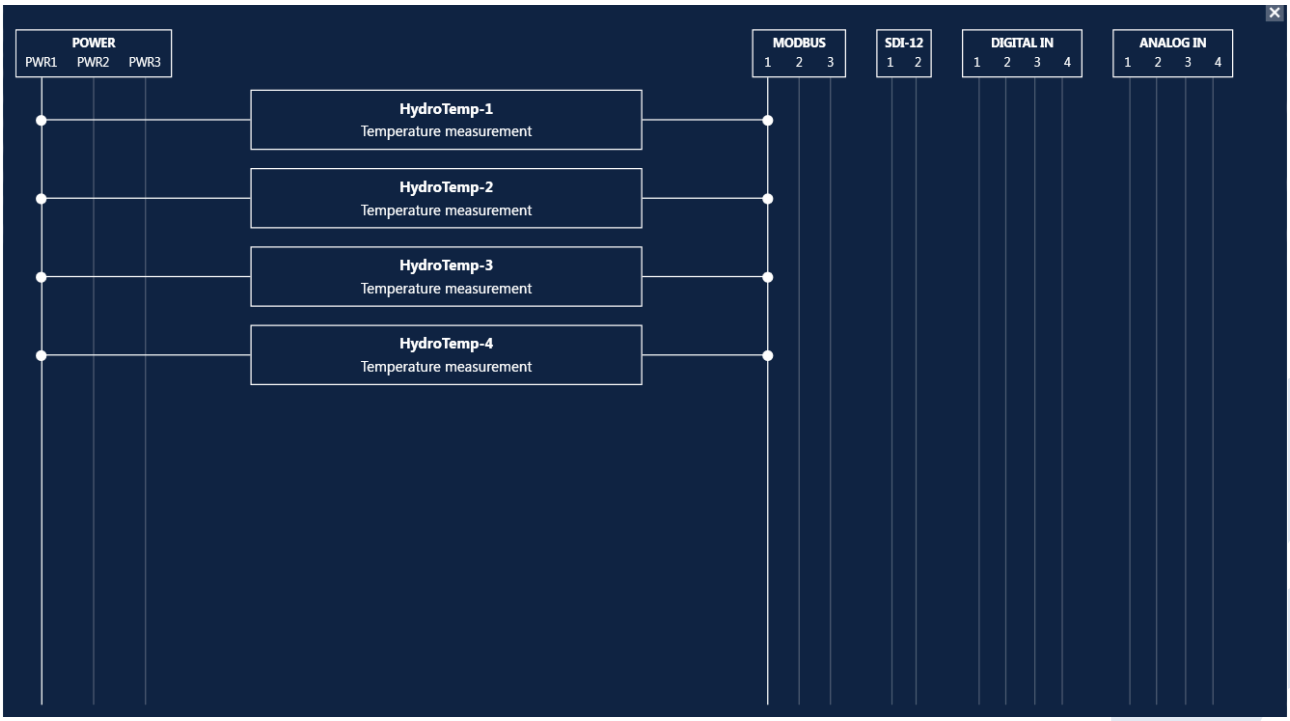


Figure 29. Geolux Instrument Configurator - Configuration Diagram

7 Configuring Data Upload

Navigating to the *Data Upload* tab allows the user to select the communication protocol for uploading measured data to the server.

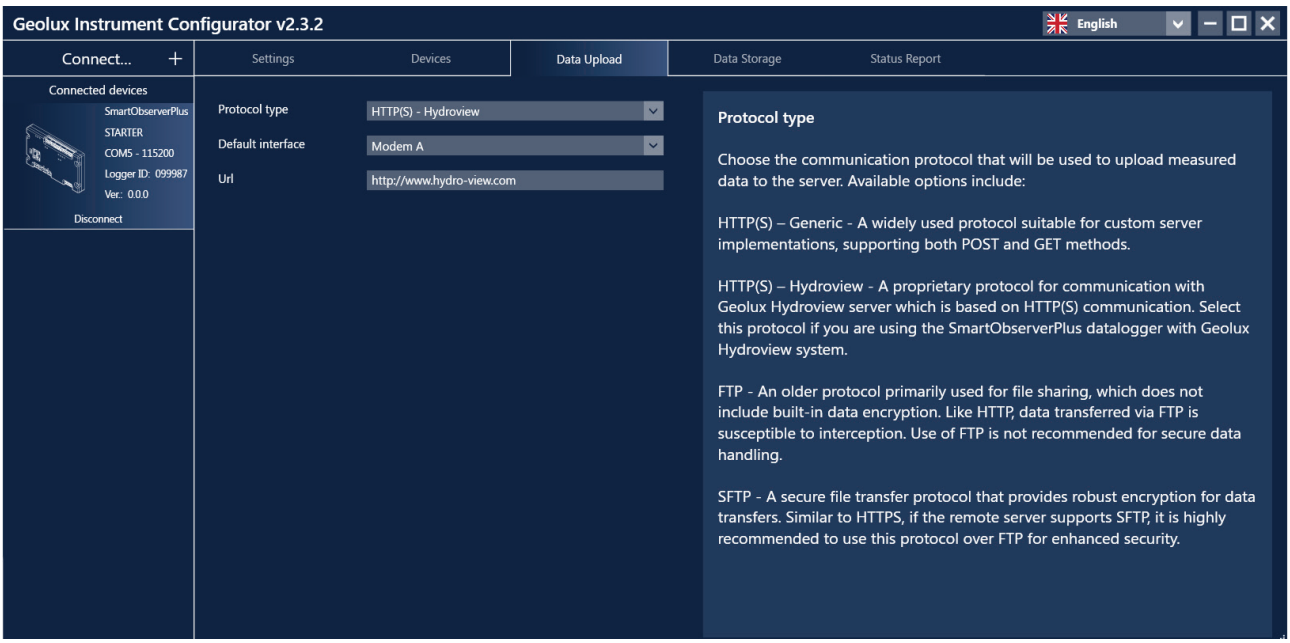


Figure 30. Geolux Instrument Configurator - Data Upload Tab

7.1. HTTP Data Upload

The user can choose between the following protocols:

- HTTP – Generic
- HTTP – Hydroview

HTTP is a widely used protocol suitable for custom implementations, supporting both POST and GET methods. The user can choose between Generic HTTP protocol that sends measurement data to any HTTP server, or HTTP protocol specifically tailored to work with Geolux Hydroview system, which not only supports sending measurement data to the Hydroview server, but also supports remote datalogger reconfiguration and remote over-the-air firmware updates.

If you intend to use SmartObserverPlus datalogger with the Geolux Hydroview system, then select the HTTP – Hydroview as protocol type. If you intend to use the datalogger with a different HTTP server, select HTTP – Generic.

For both protocols, the user must enter the *Url,* which is the address containing the server name and full path that will be accessed via the HTTP(S) request. Additionally, if the selected *Request type* is POST, the user must specify the *Upload name, Date format* and *Data format.* The data format can be either CSV, JSON, or a custom format where the user can customize the data formatting for transmission from the data logger using Python scripting.

SmartObserverPlus internally supports both unencrypted HTTP protocol, as well as encrypted HTTPS/SSL/TLS protocol. If you want to use unencrypted communication, the URL should be prefixed with http:// - if you intend to use encryted HTTPS protocol, just modify the URL so that it starts with https:// .

7.2. FTP Data Upload

The user can choose between the following protocols:

- FTP
- SFTP

FTP is an older protocol primarily used for file sharing and lacks built-in data encryption, making data transferred via FTP susceptible to interception. Use of FTP is not recommended for secure data handling.

SFTP is a secure file transfer protocol that provides robust encryption for data transfers. If the remote server supports SFTP, it is highly recommended to use this protocol instead of FTP for enhances security.

For both protocols, the user must enter the *Default interface*, i.e. the communication link that will be used to send data to the server, as well as the *Server addres*, which can be either a textual domain name or a numeric IP address. The *Server port* is automatically filled in based on the selected protocol but can be modified if needed. Additionally, the user must provide the *Username, Password* and *Upload name*, and select the *Date format* and *Data format*. The data format can be either CSV, JSON, or a custom format where the user can customize the data formatting for transmission from the data logger using Python scripting.

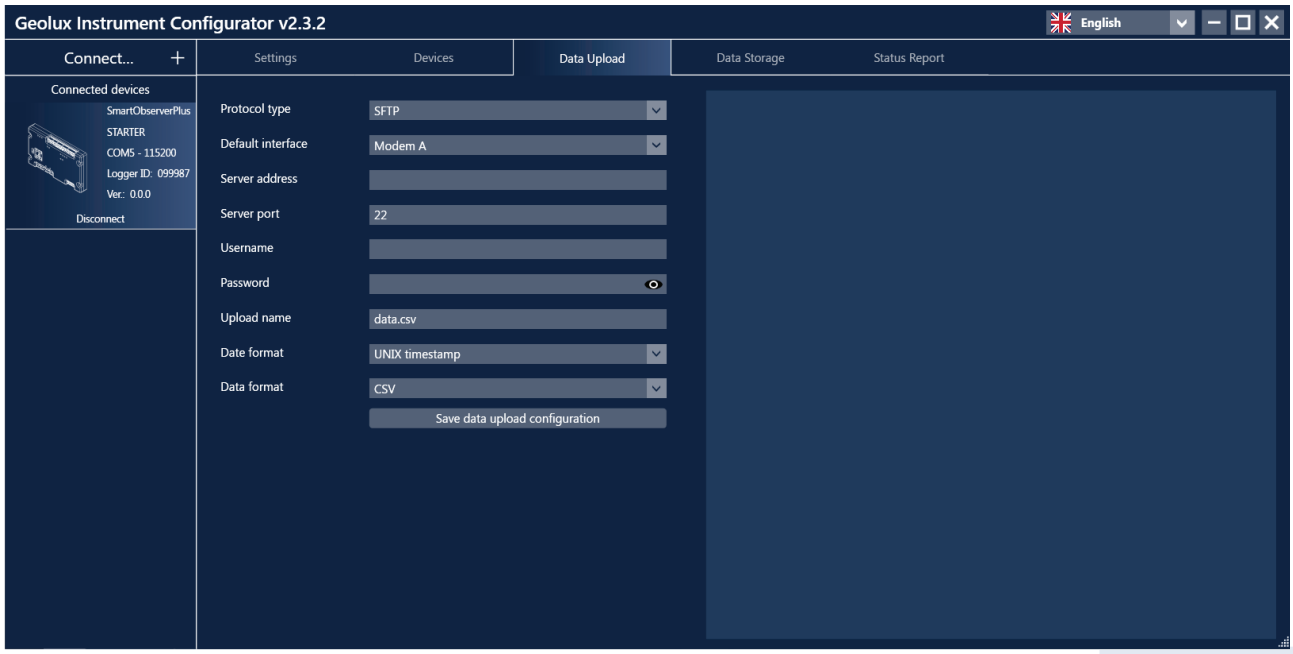


Figure 29. Geolux Instrument Configurator - Configuration Diagram

8 Updating the Firmware

8.1. Firmware Update Overview

What is Firmware Update?

A firmware update refers to the process of uploading and installing (usually new) firmware within SmartObserverPlus permanent storage.
Firmware is a specialized type of software that provides low-level control for a device's hardware.
Firmware is embedded directly into the device and enables it to operate and perform specific functions.

Why perform Firmware Update?

Updating firmware can enhance performance, fix bugs, add new features, and improve security. Geolux frequently releases firmware updates to address issues identified after a product's initial launch, ensuring that users benefit from the latest advancements and fixes.
The update process can vary depending on the device and the method used, such as

1. Over-the-air,
2. Using a dedicated app, or
3. Through external storage like SD cards or USB devices.

It is essential to follow Geolux's instructions carefully during the update process to avoid potential issues.
Regularly checking for firmware updates is recommended to ensure that your device continues to function optimally and securely. By keeping your firmware up to date, you can take full advantage of your device's capabilities and enjoy a better user experience.

Checking current Firmware version

Ascertaining the current firmware version of your device is crucial for several reasons:

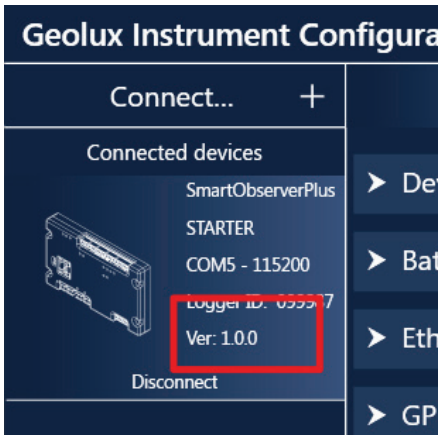
1. Helps you determine if an update is necessary.
2. Knowing your current version allows you to compare it with the latest available version and decide whether to proceed with an upgrade.

Additionally, in cases of troubleshooting, identifying the firmware version can provide valuable insights. Many issues can arise from outdated firmware, and support teams often need this information to assist you effectively.
Furthermore, some features may only be available with specific firmware versions. By checking your current version, you can ensure you are leveraging all the capabilities your device has to offer.

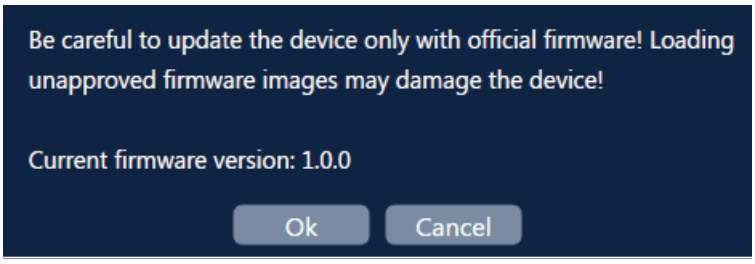
Lastly, maintaining an updated firmware version is vital for security. Older versions may have vulnerabilities that could be exploited by malicious entities. Regularly checking your firmware version helps you stay informed and proactive in safeguarding your device and data.
In summary, knowing your current firmware version is a key step in managing your device's performance, security, and overall functionality.

Using Geolux's Instrument Configurator application, connect to SmartObserverPlus Two methods:

1. In the list of Connected devices, find SmartObserverPlus, in the line starting with **Ver.:**



2. From the **Settings** tab, click on **Update Firmware** button (lower left corner). Popup shall present the current firmware version, then Cancel update.



8.2. Firmware Update Process

Common process consists of three steps:

1. Firmware version selection
2. Firmware transfer (Update Method)
3. Firmware installation

User selects a Firmware version to be installed.
Then Firmware is transferred through one out of four available update methods.
Finally, SmartObserverPlus performs Installation (including verification and restart).

Transfer Methods for Firmware Update

It is recommended to use a single update method at a time.
Using a single update method at a time minimizes the risk of conflicts and errors that can arise from overlapping processes, ensuring a smoother and more reliable firmware update experience.
This approach also simplifies troubleshooting, as it allows you to focus on one method without the complications of multiple updates occurring simultaneously.

Over-the-Air Update

The over-the-air (OTA) update method downloads and installs firmware update directly via an internet connection (i.e. Ethernet, Cellular, etc.).
This process simplifies updating by eliminating the need for physical connections or additional hardware, ensuring that Users can easily keep their device up to date. OTA updates can only be used when the SmartObserverPlus datalogger has been configured to work with Geolux Hydroview system (see chapter 7.1.).

Using the Instrument Configurator Application (from PC)

Updating using a dedicated PC application over a USB cable involves connecting the device to a computer.
Geolux's Instrument Configurator application manages the upload and installation of the firmware update directly to SmartObserverPlus.

Using SD-card

This method involves transferring the firmware update file to an SD card, which is then inserted into the device to facilitate the installation process.

Using USB Mass Storage Device (USB thumb drive)

Users can load the firmware update onto a USB thumb drive and connect it to the device, allowing for a straightforward update installation directly from the external storage.

8.3. Step-by-Step instructions

Over-the-Air Update Method

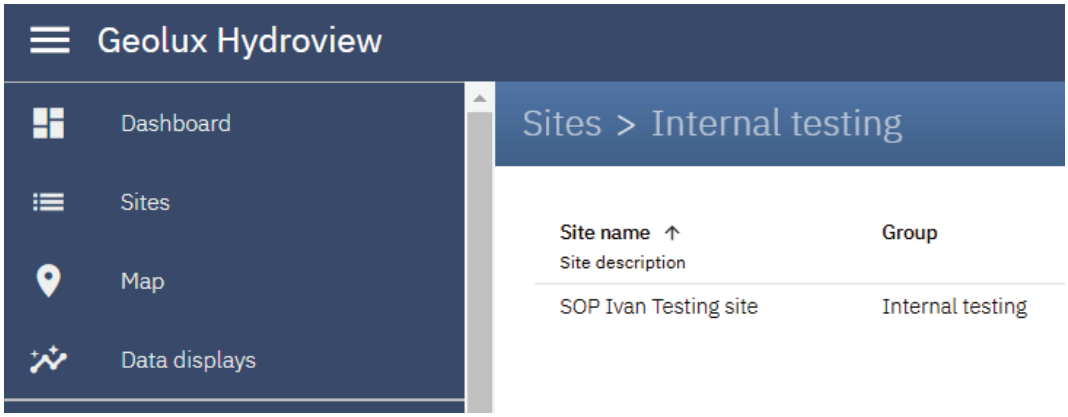
The over-the-air (OTA) update method downloads and installs firmware update directly via an internet connection (i.e. Ethernet, Cellular, etc.).
This process simplifies updates by eliminating the need for physical connections or additional hardware, ensuring that users can easily keep their device up to date.

Requirements

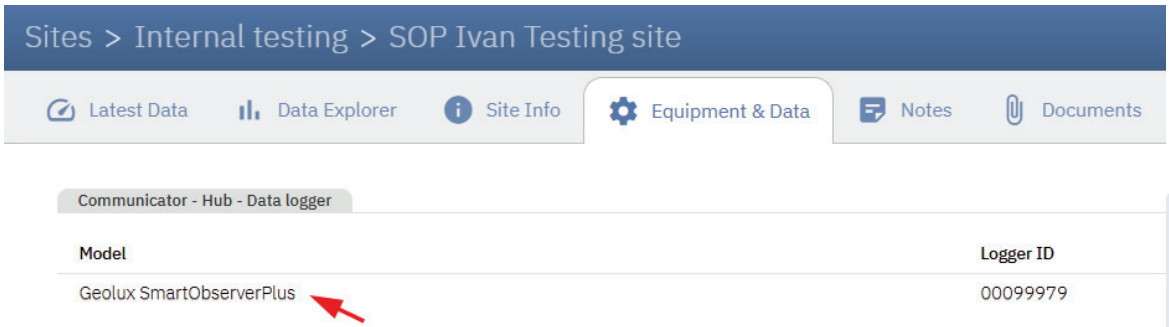
1. SmartObserverPlus is managed through HydroView
2. SmartObserverPlus has internet connection capability

Firmware version selection

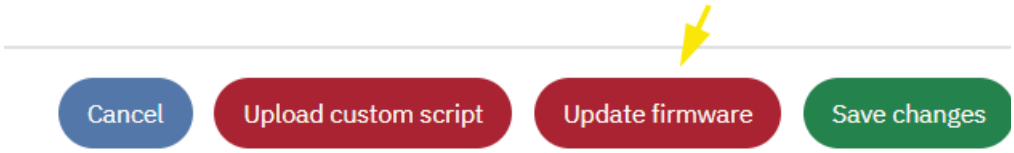
In HydroView, select the site for which You want to perform Firmware update.



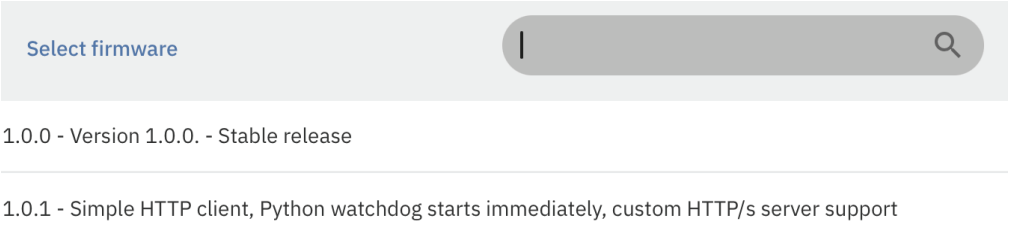
Under **Equipment & Data** click on the device name.



Click Update firmware button, a List of available Firmware versions appears.



From the List of select and click the wanted Firmware version.



The selected Firmware version is added as Pending to Firmware updates queue, with status indicated.

Firmware updates

Model	Firmware version	Firmware description	Status	Retries	Date
Geolux SmartObserverPlus	1.0.0	Version 1.0.0. - Stable release	Pending	0	2024-12-09 16:06:13
Geolux SmartObserverPlus	1.0.0	Version 1.0.0. - Stable release	Completed	1	2024-12-09 14:28:14
Geolux SmartObserverPlus	1.0.0	Version 1.0.0. - Stable release	Completed	1	2024-12-09 13:28:09
Geolux SmartObserverPlus	1.0.0	Version 1.0.0. - Stable release	Cancelled	0	2024-12-09 12:56:34

Firmware transfer

When internet connection is present, SmartObserverPlus queries HydroView whether a new firmware is available.
If a new firmware version is available, then SmartObserverPlus downloads the file with new firmware from HydroView.
Once the download of the new firmware has finished and after all measurements and/or remaining operations have completed, then the SmartObserverPlus starts firmware installation.

NOTE: Depending on connection speed and connection signal quality, duration of the download may vary from several minutes to several hours.

Firmware installation

Once the transfer of the new firmware has finished, the firmware installation starts immediately by restarting. Installation finishes within 60 seconds.

Instrument Configurator Application Method

Geolux's Instrument Configurator application manages the upload and installation of the firmware update directly to SmartObserverPlus.

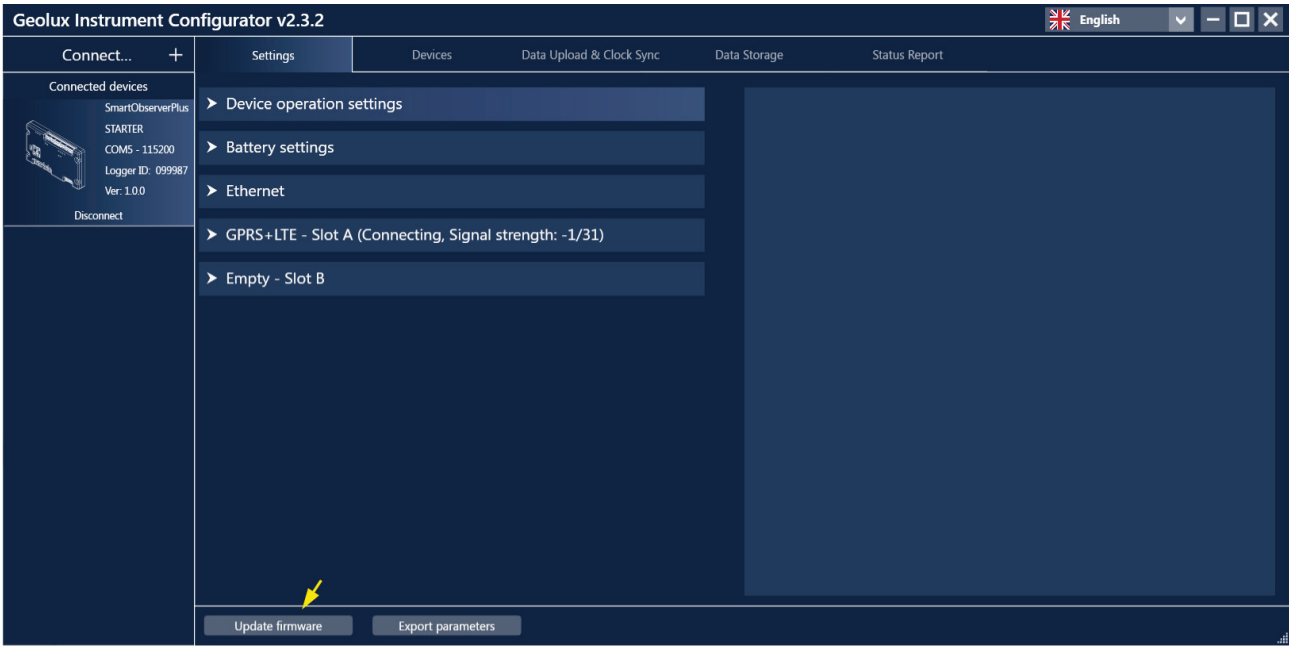
Requirements

1. installed FTDI D2xx serial drivers (https://www.ftdichip.com/drivers/).
2. installed Geolux Instrument Configurator application. (https://www.geolux-radars.com)
3. USB cable connecting Your PC to SmartObserverPlus's USB micro-B connector.
4. Firmware file (file extension ".**FWI**") provided to You by Geolux.

Firmware version selection

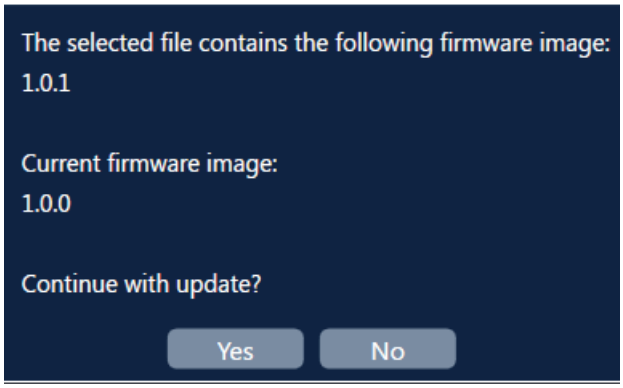
Using Geolux's Instrument Configurator application, connect to SmartObserverPlus.

From the **Settings** tab, click on **Update Firmware** button (lower left corner).



Select the firmware file (file extension ".**FWI**") provided to You by Geolux.
Popup shall present:

1. the firmware version currently on SmartObserverPlus,
2. the firmware version inside FWI file.



Verify the information presented in popup before proceeding with the update process. If there are inconsistencies, discontinue the update process.

Firmware transfer

While uploading the new firmware, a progress bar shows the amount of data transferred.

Firmware installation

Once the transfer of the new firmware has finished, the firmware installation starts immediately by restarting. Installation finishes within 60 seconds.

Version verification (optional)

Reconnect to SmartObserverPlus and check the current (updated) firmware version following the method in Chapter Using Instrument Configurator Application..

SD-card Method

Requirements

1. SD-card must be in microSD dimensions.
2. SD-card must be formatted either as FAT32 or as exFAT filesystems.
3. The maximum supported size of the SD-card is 32 GiB.
4. firmware file (file extension "**.FWI**") provided to You by Geolux.

Firmware version selection

Insert and mount the SD-card in Your computer.
Select the firmware file (file extension "**.FWI**") provided to You by Geolux.
Copy the firmware file to SD-card.
To perform the Firmware update with selected firmware file, rename the firmware file on SD-card to "**upgrade.sop.fwi**"
Dismount then eject SD-card from Your computer.

Firmware transfer

Insert the microSD-card into microSD card slot on SmartObserverPlus
When SD-card is inserted SmartObserverPlus mounts the SD-card and reads contents.
If a file "**upgrade.sop.fwi**" exists, SmartObserverPlus shall copy this file to internal storage.
During copying of firmware file, the LED indicator **SD CARD BUSY** shall be constantly turned on.
When copying is done, the LED indicator **SD CARD BUSY** shall blink in quick succession 15 times.
Eject the SD-card.

Firmware installation

To initiate Firmware installation, User must perform manual power cycling (turn power off, then turn the power on).
Installation finishes within 60 seconds.

Version verification (optional)

Connect to the SmartObserverPlus and check the current (updated) firmware version as described in Chapter Using Instrument Configurator Application.

USB Mass Storage Device (USB thumb drive) Method

Requirements

1. USB thumb drive must have USB-A connector.
2. USB thumb drive must be formatted either as FAT32 or as exFAT filesystems.
3. The maximum supported size of the USB thumb drive is 32 GiB.
4. Firmware file (file extension "**.FWI**") provided to You by Geolux.

Firmware version selection

Insert and mount the USB thumb drive in Your computer.
Select the firmware file (file extension "**.FWI**") provided to You by Geolux.
Copy the firmware file to USB thumb drive.
To perform the Firmware update with selected firmware file, rename the firmware file on USB thumb drive to "**upgrade.sop.fwi**"
Dismount then eject USB thumb drive from Your computer.

Firmware transfer

Insert the USB thumb drive into USB slot on SmartObserverPlus
When USB thumb drive is inserted SmartObserverPlus mounts the USB thumb drive and reads contents.
If a file "**upgrade.sop.fwi**" exists, SmartObserverPlus shall copy this file to internal storage.
During copying of firmware file, the LED indicator **SD CARD BUSY** shall be constantly turned on.
When copying is done, the LED indicator **SD CARD BUSY** shall blink in quick succession 15 times.
Eject the USB thumb drive.

Firmware installation

To initiate Firmware installation, User must perform manual power cycling (turn power off, then turn the power on).
Installation finishes within 60 seconds.

Version verification (optional)

Connect to the SmartObserverPlus and check the current (updated) firmware version as described in Chapter Using Instrument Configurator Application.

8.4. Restoring **Previous Firmware**

Restoring to a previous firmware version may be necessary if a recent update causes unintended issues, such as decreased performance or compatibility problems with existing software.
Users might also need to revert if critical features stop functioning as expected after an update.
This capability ensures that the device remains reliable and meets user needs even when new updates introduce unforeseen challenges.

A Appendix A: Configuration Protocol

A.1. Protocol Overview

The Configuration protocol is used to retrieve and modify the SmartObserverPlus operating parameters.

Various settings, such as system time and GPRS parameters are set using this protocol.

To make SmartObserverPlus configuration easy, Geolux provides the Geolux Instrument Configurator application.

Regular users do not need to be concerned about the Configuration protocol used between the Geolux Instrument Configurator and the SmartObserverPlus.

The Configuration protocol listens on the USB micro-B for incoming requests, and on each received request, it will respond back.

Protocol Structure

The protocol consists of **sentences.**

Each sentence starts with the hash **#** character and ends with newline **<CR><LF>** pair (carriage return, line feed).

Sentences come in request-response pairs, where request sentence is sent from the Geolux Instrument Configurator application to the SmartObserverPlus, and when received by the SmartObserverPlus, then it shall reply with a response sentence.

In case of request-response sentences, the Geolux Instrument Configurator application which initiates the sentence exchange can assume 1 second timeout interval for the response and retransmit or cancel the communicaton if the response is not received within 1 second.

A.2. CRC algorithm

name: **CRC-16/ARC** also known as **CRC-16/IBM**

polynomial = $x^{16} + x^{15} + x^2 + 1$

- 0x8005 in normal notation (MSB first, highest power omitted)

Initialize value = 0x0000

Reflect input = true

Reflect Output = true

Final XOR value = 0x0000

A.3. Protocol Sentences

Device information

Retrieve current device information

Request

#get_info<CR><LF>

Response

Consists of multiple sentences:

#logger_id: value<CR><LF>

- **value:** Value is unique ID of the SmartObserverPlus, expressed in hexadecimal.

#firmware: value<CR><LF>

- **value:** Value is current Device Firmware version, given in format *MAJOR.MINOR.PATCH*

#device_type: value<CR><LF>

- **value:** Possible values are: STARTER, BASIC, M2M, DUAL-SIM, 5G, DUAL-SIM-5G

#time: value<CR><LF>

- **value:** current time as UNIX epoch, UTC time zone, units are seconds [sec].

#ftdi_memory: value<CR><LF>

- value *0* indicates FTDI USB driver has not been initialized
- value *1* indicates FTDI USB driver has been initialized

#slot_a: value<CR><LF>

- **value:** Configured Modem type in Slot-A, possible values are: *none, gprs, gprs-lte, lte*

#slot_b: value<CR><LF>

- **value:** Configured Modem type in Slot-B, possible values are: *none, gprs, gprs-lte, lte*

#eth_dhcp: value<CR><LF>

- value *0* DHCP usage is disabled
- value *1* DHCP usage is enabled

#eth_dhcp_ip_address: ipaddr<CR><LF>

- **ipaddr:** IPv4 address as assigned by DHCP, if not assigned by DHCP value is NA

#eth_ip_address: ipaddr<CR><LF>

- **ipaddr:** Static IPv4 address, from configuration

`#eth_subnet_mask: ipaddr<CR><LF>`

- **ipaddr:** Subnet mask IPv4, from configuration

`#eth_gateway_address: ipaddr<CR><LF>`

- **ipaddr:** Gateway IPv4 address, from configuration

`#eth_dns: ipaddr1[,ipaddr2,ipaddr3,ipaddr4]<CR><LF>`

- **ipaddr1 … ipaddr4:** Values are IPv4 address, given in format `ABC.DEF.GHI.JKL`

`#eth_mac: value<CR><LF>`

- **value:** Ethernet MAC address, IEEE EUI-64 format, 8C-1F-64-BE-xx-xx

`#charge_current: value<CR><LF>`

- **value:** positive decimal integer, units milli Amperes [mA], from configuration

`#max_charge_voltage: value<CR><LF>`

- **value:** positive decimal integer, units milli Volts [mV], from configuration

`#mppt_voltage: value<CR><LF>`

- **value:** positive decimal integer, units milli Volts [mV], from configuration

`#min_charge_current: value<CR><LF>`

`#no_charge_temperature: value<CR><LF>`

- **value:** positive decimal integer, units degrees Celsius [°C], from configuration

`#enable_charge: value<CR><LF>`

- value `0` indicates battery charging is disabled
- value `1` indicates battery charging is enabled
- from configuration

`#battery_full_charge_capacity: value<CR><LF>`

- **value:** positive decimal integer, units milli Ampere hours [mAh], from configuration

`#scan_interval_min: value<CR><LF>`

- **value:** positive decimal integer, units minutes [min], from configuration

`#slot_a_gprs_apn: value<CR><LF>`

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

`#slot_a_gprs_username: value<CR><LF>`

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

`#slot_a_gprs_password: value<CR><LF>`

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

`#slot_a_gprs_dns: ipaddr<CR><LF>`

- **ipaddr:** Value is an IPv4 address, given in format `ABC.DEF.GHI.JKL`

`#slot_a_mob_operator_lock: value<CR><LF>`

- value `0` indicates any network is allowed for Slot-A
- value `1` indicates roaming is disabled for Slot-A
- value `2` indicates only home network is allowed for Slot-A

`#slot_a_modem_connection_type: value<CR><LF>`

- value *0* indicates network type is GSM/LTE-M for Slot-A
- value *1* indicates network type is NB-IOT for Slot-A

#slot_a_modem_type: value<CR><LF>

- **value:** Value indicates model of current Cellular modem in Slot-A

#slot_a_port_type: value<CR><LF>

- **value:** Value indicates communication port of current Cellular modem in Slot-A

#slot_a_rssi: value<CR><LF>

- **value:** RSSI (Received Signal Strength Indicator) of Cellular modem in Slot-A

#slot_a_status: value<CR><LF>

- **value:** Status of Cellular modem in Slot-A, possible values: *Disconnected, Disconnecting, Connecting, Connected*

#slot_b_gprs_apn: value<CR><LF>

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

#slot_b_gprs_username: value<CR><LF>

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

#slot_b_gprs_password: value<CR><LF>

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

#slot_b_gprs_dns: ipaddr<CR><LF>

- **ipaddr:** Value is an IPv4 address, given in format *ABC.DEF.GHI.JKL*

#slot_b_mob_operator_lock: value<CR><LF>

- value *0* indicates any network is allowed for Slot-B
- value *1* indicates roaming is disabled for Slot-B
- value *2* indicates only home network is allowed for Slot-B

#slot_b_modem_connection_type: value<CR><LF>

- value *0* indicates network type is GSM/LTE-M for Slot-B
- value *1* indicates network type is NB-IOT for Slot-B

#slot_b_modem_type: value<CR><LF>

- **value:** Value indicates model of current Cellular modem in Slot-B

#slot_b_port_type: value<CR><LF>

- **value:** Value indicates communication port of current Cellular modem in Slot-B

#slot_b_rssi: value<CR><LF>

- **value:** RSSI (Received Signal Strength Indicator) of Cellular modem in Slot-B

#slot_b_status: value<CR><LF>

- **value:** Status of Cellular modem in Slot-B, possible values: *Disconnected, Disconnecting, Connecting, Connected*

#usb: value<CR><LF>

- value *none* indicates USB thumb drive is either not present or not mounted
- value *flash* indicates USB thumb drive is present & mounted

#usb_info: total_size,used_size,filesystem_type<CR><LF>

- **total_size:** Total size of USB thumb drive, in units of bytes [B]
- **used_size:** Used size of USB thumb drive, in units of bytes [B]
- **filesystem_type:** possible values: *NONE, FAT12, FAT16, FAT32, exFAT*

#sdcard: value<CR><LF>

- value *readwrite* indicates SD-card is either not present or not mounted
- value *flash* indicates SD-card is present & mounted

 # sdcard_info: total_size,used_size,filesystem_type<CR><LF>

- **total_size:** Total size of SD-card, in units of bytes [B]
- **used_size:** Used size of SD-card, in units of bytes [B]
- **filesystem_type:** possible values: *NONE, FAT12, FAT16, FAT32, exFAT*

#save_usb: value<CR><LF>

- value *0* indicates Logging to USB thumb drive is disabled
- value *1* indicates Logging to USB thumb drive is enabled

#save_sd: value<CR><LF>

- value *0* indicates Logging to SD-card is disabled
- value *1* indicates Logging to SD-card is enabled

#ping: value<CR><LF>

- value *ping* indicates previous pong
- value *pong* indicates previous ping

#free_heap: value<CR><LF>

- **value:** Value is available free heap memory, in units of bytes [B]

Example

```
Req : #get_info<CR><LF>>
Resp: #logger_id: 099979<CR><LF>
      #firmware: 1.0.0<CR><LF>
      #device_type: STARTER<CR><LF>
      #time: 1733748099<CR><LF>
      #ftdi_memory: 0<CR><LF>
      #slot_a: gprs-lte<CR><LF>
      #slot_b: none<CR><LF>
      #eth_dhcp: 1<CR><LF>
      #eth_dhcp_ip_address: NA<CR><LF>
      #eth_ip_address: 192.168.2.100<CR><LF>
      #eth_subnet_mask: 255.255.0.0<CR><LF>
      #eth_gateway_address: 192.168.2.1<CR><LF>
      #eth_dns: 8.8.8.8,1.1.1.1<CR><LF>
```

```
#eth_mac: 8C-1F-64-BE-10-00<CR><LF>
#charge_current: 2000<CR><LF>
#max_charge_voltage: 12600<CR><LF>
#mppt_voltage: 16800<CR><LF>
#min_charge_current: 70<CR><LF>
#no_charge_temperature: -10<CR><LF>
#enable_charge: 1<CR><LF>
#battery_full_charge_capacity: 7500<CR><LF>
#scan_interval_min: 5<CR><LF>
#slot_a_gprs_apn: terminal.apn<CR><LF>
#slot_a_gprs_username: <CR><LF>
#slot_a_gprs_password: <CR><LF>
#slot_a_gprs_dns: <CR><LF>
#slot_a_mob_operator_lock: 0<CR><LF>
#slot_a_modem_connection_type: 0<CR><LF>
#slot_a_modem_type: SIMCOM_SIM7600G<CR><LF>
#slot_a_port_type: uart-c<CR><LF>
#slot_a_rssi: 29/31<CR><LF>
#slot_a_status: Connected<CR><LF>
#usb: none<CR><LF>
#sdcard: none<CR><LF>
#save_usb: 0<CR><LF>
#save_sd: 0<CR><LF>
#ping: pong<CR><LF>
#free_heap: 8154280<CR><LF>
```

Clear Datalog

Removes all datalog information from the internal storage. Datalog storage is a cyclical storage that keeps all previous measurements.

Request

```
#clear_datalog<CR><LF>
```

Response

```
#clear_datalog: OK<CR><LF>
```

Example

```
Req : #clear_datalog<CR><LF>
Resp: #clear_datalog: OK<CR><LF>
```

Status Report file information

Get information (size and CRC) about the status report file. The status report file is a CBOR-encoded binary document that contains current device status data.

Request

```
#get_status<CR><LF>
```

Response

```
#get_status: size_bytes,crc16_decimal<CR><LF>
```

- **size_bytes:** The size of the status report file that can be fetched (in bytes).
- **crc16_decimal:** The CRC of the status report file expressed as a decimal number using the CRC-16/ARC (also known as CRC-16/IBM) algorithm. Provided for file integrity checks after down load.

Example

```
Req : #get_status<CR><LF>
Resp: #get_status: 321,58542<CR><LF>
```

Fetching Status Report File

Once the size of the status report file is known, fetch the status report file in blocks.

Request

```
#get_status_block=block_offset_in_bytes,block_size_in_bytes<CR><LF>
```

- **block_offset_in_bytes:** The starting point in the status report file from which to fetch the data (in bytes).
- **block_size_in_bytes:** The number of bytes to fetch from the status report file.

Response

```
#get_status_block: block_data<CR><LF>
```

- **block_data:** This data is encoded in Base64 format. Contains block of data from the status report file at the requested offset and of the specified size.

Example

```
Req : #get_status_block=0,321<CR><LF>
```


```
Resp:    #get_status_block:    om9EYXRhIGNvbm5lY3Rpb26BomlUaW1lc3RhbXAaZ1hPe2ZTdGF0dXNw
GlzY29ubmVjdCBtb2RlbWVPdGhlcoOiaVRpbWVzdGFtcBpnWE97ZlN0YXR1c3gbLS0tW1NdLT5EaX
jb25uZWN0aW5nIG1vZGVtomlUaW1lc3RhbXAaZ1hPfmZTdGF0dXN4US0tLVtTXS0+U2xlZXBpbmcgMDM6NT
gYmVmb3JlIG5leHQgd2FrZXVwICg1IG1pbikuIFNhbXBsZSBvbjogMjAyNC0xMi0xMCAxNDozMDowM
JpVGltZXN0YW1wGmdYT35mU3RhdHVzeElTbGVlcGluZyAwMzo1NyBiZWZvcmUgbmV4dCB3YWtldXAgKDUg
WluKS4gU2FtcGxlIG9uOiAyMDI0LTEyLTEwIDE0OjMwOjAy<CR><LF>
```

Read Bootmode

This command is used to verify the current boot mode of the device. The boot mode can be one of the following:

- **BOOT:** Indicates the device is in normal operation mode.
- **UPGRADE:** Indicates that a firmware update is ready in the internal storage and will be applied upon the next device restart.

Request

```
#read_bootmode<CR><LF>
```

Response

If SmartObserverPlus has a pending Firmware Update request but is has not yet been executed.ing:

```
#read_bootmode: UPGRADE<CR><LF>
```

If SmartObserverPlus has no pending Firmware Update i.e. is in normal booting mode.

```
#read_bootmode: BOOT<CR><LF>
```

Example

```
Req : #read_bootmode<CR><LF>
Resp: #read_bootmode: BOOT<CR><LF>
```

Start Firmware Update

Put SmartObserverPlus into Firmware Update mode.

In Firmware Update mode, SmartObserverPlus expects to receive Firmware file.

If there is no data received within 120 seconds, SmartObserverPlus exits Firmware Update mode and returns to Normal mode.

Request

```
#start_upgrade<CR><LF>
```

Response

```
#start_upgrade: OK<CR><LF>
```

Example

```
Req : #start_upgrade<CR><LF>
Resp: #start_upgrade: OK<CR><LF>
```

Enable/Disable Logging to SD-card

Enable or Disable saving measurements in CSV format (comma separated values) to SD-card.

Request

```
#save_sd=value<CR><LF>
```

- value *0* disables Logging to SD-card
- value *1* enables Logging to SD-card

Response

On success

```
#save_sd: OK<CR><LF>
```

On error

```
#save_sd: ERR<CR><LF>
```

Example

```
Req : #save_sd=0<CR><LF>
Resp: #save_sd: OK<CR><LF>
```

Enable/Disable Logging to USB thumb drive

Enable or Disable saving of Datalog in CSV format (comma separated values) to USB thumb drive.

Request

```
#save_usb=value<CR><LF>
```

- value *0* disables Logging to USB thumb drive
- value *1* enables Logging to USB thumb drive

Response

On success

```
#save_usb: OK<CR><LF>
```

On error

```
#save_usb: ERR<CR><LF>
```

Example

```
Req : #save_usb=1<CR><LF>
Resp: #save_usb: OK<CR><LF>
```

Set Battery Capacity

Set the total capacity of the connected battery when fully charged. This information is used to estimate the remaining capacity of the attached battery.

Request

```
#set_battery_full_charge_capacity=value<CR><LF>
```

- **value:** Value is a positive decimal integer, given in units of milli Ampere hours [mAh]. Accepted limits are from 0 mAh to 1,000,000 mAh (one million).

Response

On success

```
#set_battery_full_charge_capacity: OK<CR><LF>
```

On error

```
#set_battery_full_charge_capacity: ERR<CR><LF>
```

Example

```
Req : #set_battery_full_charge_capacity=2000<CR><LF>
Resp: #set_battery_full_charge_capacity: OK<CR><LF>
```

Set Battery Charge Current

Set the maximum charging current for the battery, when charging is enabled.

Request

```
#set_charge_current=value<CR><LF>
```

- **value:** Value is a positive decimal integer, given in units of milli Amperes [mA]. Accepted limits are from 250 mA to 2,750 mA.

Response

On success

```
#set_charge_current: OK<CR><LF>
```

On error

```
#set_charge_current: ERR<CR><LF>
```

Example

Setting 2000 mA i.e 2.0 A charging currrent

```
Req : #set_charge_current=2000<CR><LF>
Resp: #set_charge_current: OK<CR><LF>
```

Enable/Disable Battery Charging

Enable or Disable battery charging.

Request

```
#set_enable_charge=value<CR><LF>
```

- value *0* disables battery charging
- value *1* enables battery charging

Response

On success

```
#set_enable_charge: OK<CR><LF>
```

On error

```
#set_enable_charge: ERR<CR><LF>
```

Example

Enabling battery charging.

```
Req : #set_enable_charge=1<CR><LF>
Resp: #set_enable_charge: OK<CR><LF>
```

Set Battery Charging Maximum Voltage

Set the maximum charging voltage for the battery.

Request

```
#set_max_charge_voltage=value<CR><LF>
```

- **value:** Value is a positive decimal integer, given in units of milli Volts [mV]. Accepted limits are from 5,000 mV to 19,200 mV.

Response

On success

```
#set_max_charge_voltage: OK<CR><LF>
```

On error

```
#set_max_charge_voltage: ERR<CR><LF>
```

Example

Setting Battery charging maximum voltage to 12.6 V.

```
Req : #set_max_charge_voltage=12600<CR><LF>
Resp: #set_max_charge_voltage: OK<CR><LF>
```

Set MPPT (Maximum Power Point Tracking) Voltage

Set the MPPT voltage for solar panel input.

Request

```
#set_mppt_voltage=value<CR><LF>
```

- **value:** Value is a positive decimal integer, given in units of milli Volts [mV]. Accepted limits are from 5,000 mV to 19,200 mV.

Response

On success

```
#set_mppt_voltage: OK<CR><LF>
```

On error

```
#set_mppt_voltage: ERR<CR><LF>
```

Example

Setting Battery charging maximum voltage to *16.8 V.*

```
Req : #set_mppt_voltage=16800<CR><LF>
Resp: #set_mppt_voltage: OK<CR><LF>
```

Set Battery Charging Minimum Temperature

Set the minimum temperature for charging the battery.
When the environment temperature is lower, the battery will not be charged in order to prevent battery damage.

Request

```
#set_no_charge_temperature=value<CR><LF>
```

- **value:** Value is a positive decimal integer, given in units of degrees Celsius [°C]. Accepted limits are from -50 °C to 100 °C.

Response

On success

```
#set_no_charge_temperature: OK<CR><LF>
```

On error

```
#set_no_charge_temperature: ERR<CR><LF>
```

Example

Setting Battery charging minimum temperature to *-10 °C.*

```
Req : #set_no_charge_temperature=-10<CR><LF>
Resp: #set_no_charge_temperature: OK<CR><LF>
```

Set Scan Interval

Set the scan interval how often the SmartObserverPlus takes measurements.

Request

```
#set_scan_interval_min=value<CR><LF>
```

- **value:** Value is a positive decimal integer, given in units of minutes [min]. Accepted limits must be greater than 0 min.

Response

On success

```
#set_scan_interval_min: OK<CR><LF>
```

On error

```
#set_scan_interval_min: ERR<CR><LF>
```

Example

Setting the scan interval to *15 min.*

```
Req : #set_scan_interval_min=15<CR><LF>
Resp: #set_scan_interval_min: OK<CR><LF>
```

Enable/Disable DHCP for Ethernet

Enable or Disable usage of DHCP (Dynamic Host Configuration Protocol).

If use of DHCP is Enabled, then the IP address is obtained from DHCP server as a dynamic IP address.

If use of DHCP is Disabled, then the IP address is manually assigned, see Static Ethernet IP address.

Request

```
#set_eth_dhcp=value<CR><LF>
```

- value *0* disables DHCP usage
- value *1* enables DHCP usage

Response

On success

```
#set_eth_dhcp: OK<CR><LF>
```

On error

```
#set_eth_dhcp: ERR<CR><LF>
```

Example

Enabling Ethernet DHCP usage.

```
Req : #set_eth_dhcp=1<CR><LF>
Resp: #set_eth_dhcp: OK<CR><LF>
```

Set DNS Servers for Ethernet

Set DNS addresses, up to four IPv4 addresses.
These DNS addreses are used when: - DHCP is Disabled - DHCP is Enabled but DHCP has no DNS entries.

Request

```
#set_eth_dns=ipaddr1[,ipaddr2,ipaddr3,ipaddr4]<CR><LF>
```

- **ipaddr1:** Value is an IPv4 address, given in format *ABC.DEF.GHI.JKL*
- **ipaddr2 ... ipaddr4** are optional [], same as **ipaddr1**
- if providing more than one, values are separated using a comma , character, there must be no whitespaces
- if providing empty DNS **ipaddr1** must be empty

Response

On success

```
#set_eth_dns: OK<CR><LF>
```

On error

```
#set_eth_dns: ERR<CR><LF>
```

Example

Setting two IPv4 addresses for Ethernet DNS addresses. Using public Google DNSs, first *8.8.8.8 second 1.1.1.1*

```
Req : #set_eth_dns=8.8.8.8,1.1.1.1<CR><LF>
Resp: #set_eth_dns: OK<CR><LF>
```

Setting an empty DNS list

```
Req: #set_eth_dns=<CR><LF>
Resp: #set_eth_dns: OK<CR><LF>
```

Set Ethernet Gateway addresses

Set the IPv4 address of the Ethernet gateway, used when DHCP is Disabled.

Request

```
#set_eth_gateway_address=ipaddr<CR><LF>
```

- **ipaddr:** Value is an IPv4 address, given in format *ABC.DEF.GHI.JKL*

Response

On success

```
#set_eth_gateway_address: OK<CR><LF>
```

On error

```
#set_eth_gateway_address: ERR<CR><LF>
```

Example

Setting Ethernet gateway to IPv4 addresses *192.168.1.15.*

```
Req : #set_eth_gateway_address=192.168.1.15<CR><LF>
Resp: #set_eth_gateway_address: OK<CR><LF>
```

Set Static IP Address for Ethernet

Set the IPv4 address of the Static Ethernet IP address, used when DHCP is Disabled.

Request

```
#set_eth_ip_address=ipaddr<CR><LF>
```

- **ipaddr:** Value is an IPv4 address, given in format *ABC.DEF.GHI.JKL*

Response

On success

```
#set_eth_ip_address: OK<CR><LF>
```

On error

```
#set_eth_ip_address: ERR<CR><LF>
```

Example

Setting Static Ethernet IP address to IPv4 addresses *192.168.1.100.*

```
Req : #set_eth_ip_address=192.168.1.100<CR><LF>
Resp: #set_eth_ip_address: OK<CR><LF>
```

Set Subnet Mask for Ethernet

Set the IPv4 address of the Ethernet subnet mask, used when DHCP is Disabled.

Request

```
#set_eth_subnet_mask=ipaddr<CR><LF>
```

- **ipaddr:** Value is an IPv4 address, given in format *ABC.DEF.GHI.JKL*

Response

On success

```
#set_eth_subnet_mask: OK<CR><LF>
```

On error

```
#set_eth_subnet_mask: ERR<CR><LF>
```

Example

Setting Static Ethernet IP address to IPv4 addresses *255.255.0.0.*

```
Req : #set_eth_subnet_mask=255.255.0.0<CR><LF>
Resp: #set_eth_subnet_mask: OK<CR><LF>
```

Set APN for Modem 1

Set the GPRS APN for Slot-A Cellular modem.

Request

```
#set_slot_a_gprs_apn=value<CR><LF>
```

- **value:** Value is a NULL terminated string, string length limited to 40 characters.
- **ipaddr2 ... ipaddr4** are optional [], same as **ipaddr1**
- if providing more than one, values are separated using a comma , character, there must be no whitespaces
- if providing empty DNS **ipaddr1** must be empty

Response

On success

```
#set_gprs_apn: OK<CR><LF>
```

On error

```
#set_gprs_apn: ERR<CR><LF>
```

Example

Setting Slot-A GPRS APN to *virtualA.apn.*

```
Req : #set_slot_a_gprs_apn=virtualA.apn<CR><LF>
Resp: #set_gprs_apn: OK<CR><LF>
```

Set APN for Modem 2

Set the GPRS APN for Slot-B Cellular modem.

Request

```
#set_slot_b_gprs_apn=value<CR><LF>
```

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

Response

On success

```
#set_gprs_apn: OK<CR><LF>
```

On error

```
#set_gprs_apn: ERR<CR><LF>
```

Example

Setting Slot-B GPRS APN to *virtualB.apn.*

```
Req : #set_slot_b_gprs_apn=virtualB.apn<CR><LF>
Resp: #set_gprs_apn: OK<CR><LF>
```

Set DNS Servers for Modem 1

Set the GPRS DNS address for Slot-A Cellular modem.

Request

```
#set_slot_a_gprs_dns=ipaddr<CR><LF>
```

- **ipaddr:** Value is an IPv4 address, given in format *ABC.DEF.GHI.JKL*

Response

On success

```
#set_gprs_dns: OK<CR><LF>
```

On error

```
#set_gprs_dns: ERR<CR><LF>
```

Example

Setting Slot-A GPRS DNS address to *10.1.35.81*.

```
Req : #set_slot_a_gprs_dns=10.1.35.81<CR><LF>
Resp: #set_gprs_dns: OK<CR><LF>
```

Set DNS Servers for Modem 2

Set the GPRS DNS address for Slot-B Cellular modem.

Request

```
#set_slot_b_gprs_dns=ipaddr<CR><LF>
```

- **ipaddr:** Value is an IPv4 address, given in format *ABC.DEF.GHI.JKL*

Response

On success

```
#set_gprs_dns: OK<CR><LF>
```

On error

```
#set_gprs_dns: ERR<CR><LF>
```

Example

Setting Slot-B GPRS DNS address to *10.1.35.79*.

```
Req : #set_slot_b_gprs_dns=10.1.35.79<CR><LF>
Resp: #set_gprs_dns: OK<CR><LF>
```

Set APN Password for Modem 1

Set the GPRS password for Slot-A Cellular modem.

Request

```
#set_slot_a_gprs_password=value<CR><LF>
```

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

Response

On success

```
#set_gprs_password: OK<CR><LF>
```

On error

```
#set_gprs_password: ERR<CR><LF>
```

Example

Setting Slot-A GPRS password to *passwordA*.

```
Req : #set_slot_a_gprs_password=passwordA<CR><LF>
Resp: #set_gprs_password: OK<CR><LF>
```

Set APN Password for Modem 2

Set the GPRS password for Slot-B Cellular modem.

Request

```
#set_slot_b_gprs_password=value<CR><LF>
```

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

Response

On success

```
#set_gprs_password: OK<CR><LF>
```

On error

```
#set_gprs_password: ERR<CR><LF>
```

Example

Setting Slot-B GPRS password to *passwordB*.

```
Req : #set_slot_b_gprs_password=passwordB<CR><LF>
Resp: #set_gprs_password: OK<CR><LF>
```

Set APN Username for Modem 1

Set the GPRS username for Slot-A Cellular modem.

Request

```
#set_slot_a_gprs_username=value<CR><LF>
```

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

Response

On success

```
#set_gprs_username: OK<CR><LF>
```

On error

```
#set_gprs_username: ERR<CR><LF>
```

Example

Setting Slot-A GPRS username to *usernameA*.

```
Req : #set_slot_a_gprs_username=usernameA<CR><LF>
Resp: #set_gprs_username: OK<CR><LF>
```

Set APN Username for Modem 2

Set the GPRS username for Slot-B Cellular modem.

Request

```
#set_slot_b_gprs_username=value<CR><LF>
```

- **value:** Value is a NULL terminated string, string length limited to 40 characters.

Response

On success

```
#set_gprs_username: OK<CR><LF>
```

On error

```
#set_gprs_username: ERR<CR><LF>
```

Example

Setting Slot-B GPRS username to *usernameB*.

```
Req : #set_slot_b_gprs_username=usernameB<CR><LF>
Resp: #set_gprs_username: OK<CR><LF>
```

Set System Time

Set the system time.

Request

```
#set_time=value<CR><LF>
```

- **value:** Value is time given as UNIX epoch, UTC time zone, units are seconds [sec].

Response

On success

```
#set_time: OK<CR><LF>
```

On error

```
#set_time: ERR<CR><LF>
```

Example

Setting system time to *2024-12-10T15:51:57+00:00* in ISO8601, which is *1733845917* in UNIX epoch.

```
Req : #set_time=1733845917<CR><LF>
Resp: #set_time: OK<CR><LF>
```