

OpenSprinkler – Analog Sensor API Extension

Author: Stefan Schmaltz, Checkbox IT GmbH – OpenSprinklerShop

Date: 01-2025 – Version 1.0

1. Overview

With the analog sensor API extension, the OpenSprinkler 3.2 / 3.3 and also OSPi can access analog data sources. But what are analog data sources? While digital signals only have the values 1 or 0 (on or off), analog signals have a linear gradation, e.g. expressed as a percentage, 0%... 100% without intermediate steps.

For this purpose, the voltage at the inputs is measured linearly. Supported voltages are 0V – 5V.

In addition, there is the option of integrating professional, digital sensors with analog values. These sensors are integrated via the Modbus interface and deliver their data in the same way as the purely analog versions. Currently, we only support the RS485 Modbus sensors from Truebner.

The API is fully integrated into the advanced OpenSprinkler firmware. So to use the functions, the OpenSprinkler firmware must be updated. There is also a separate app for IOS and Android, the OpenSprinklerASB APP.

The following pages describe how to use and install it.

The analog sensor API is also open source and you can find the source code on Github/OpenSprinklerShop/OpenSprinkler-Firmware.

The associated app can also be found on Github/OpenSprinklerShop/OpenSprinkler-App.

Updated information is also available here:

<https://www.opensprinklershop.de/asb>

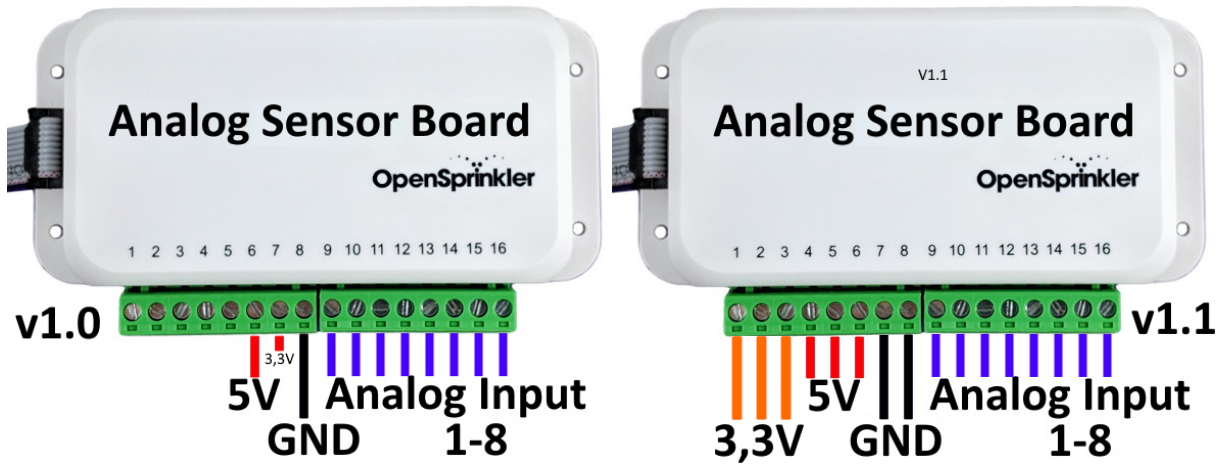
<https://www.opensprinklershop.de/rs>

This documentation refers to the OpenSprinklerShop API version 2.3.3 (173)

2. Supported hardware

The following hardware is currently supported:

2.1 Analog Sensor Board (ASB) 1.0 and 1.1



The Analog Sensor Board 1.1 has a modified circuit for outputs 1-7, an additional noise filter and a DIP switch on the back so that it can be connected as a second board.

All analog sensors with an output voltage of 0..5V can be connected. 5V and 3.3V connections are available to supply the sensor with power. For other power supplies, GND must always be connected as well.

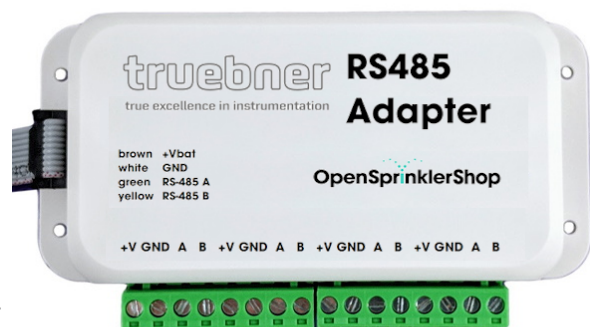
Exactly one analog sensor board 1.0 and 1.1, or two 1.1 boards can be connected to the OpenSprinkler 3.x.

2.2 Truebner RS485 adapter

This adapter can be used to connect Truebner RS485 sensors directly.

Currently, the SMT100, Aquaflex and TH100 are each supported in the RS485 Modbus version.

The connections are labelled here. Each of the 4 connections is a master, so that 16 sensors can be connected to each of them, for a total of 64 sensors. Each master only receives power (5V) when measuring.



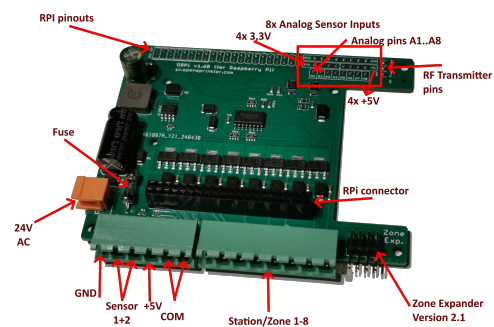
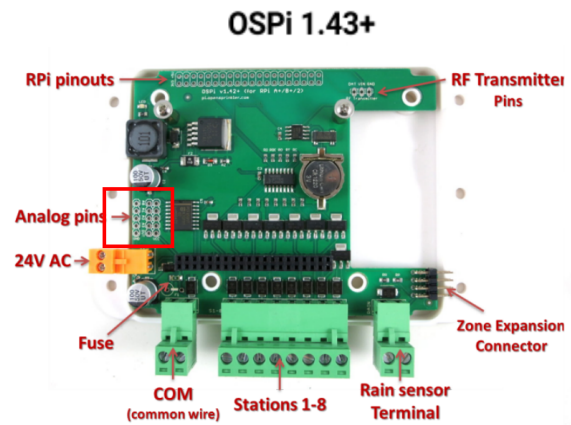
2.3 Raspberry PI

2.3.1 Analog interfaces OSPi board

The analog interfaces of the OSPi board can be used with the analog sensor API and the analog sensors, e.g. the Truebner SMT50. However, you should solder a pin header here and equip the sensor cables with suitable connectors.

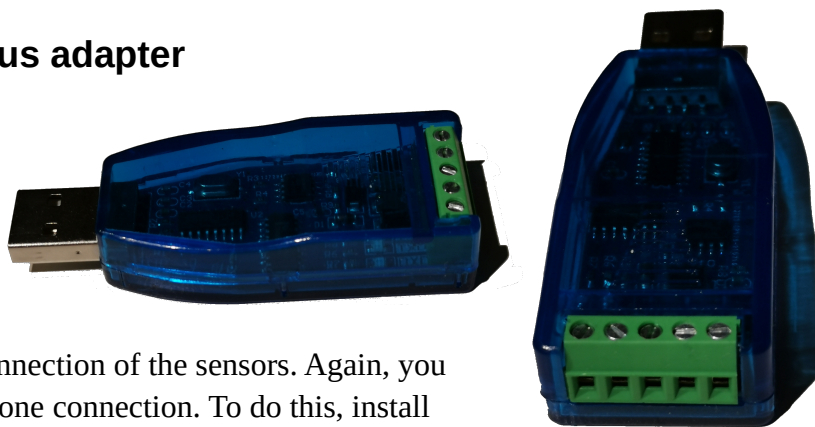
The following OSPi boards are supported: PCF8591 (4 analog inputs, 8-bit) and ADS1115 (2x4 analog inputs, 16-bit), marked in red in the images on the right. Version 1.5 was also available without analog interfaces; from version 1.6 onwards, there are 2xADS1115 modules. These are automatically detected and made available during firmware compilation. Please note that you need the extended OpenSprinklerShop firmware for this. Before compiling, activate the I2C support with raspi-config.

Supports also OSPi 2.0 !



2.3.2 USB RS485 Modbus adapter

There are many providers that offer a USB RS485 Modbus adapter for the Raspberry PI. We recommend the models that also provide the power supply (+5V, GND) via the terminal block, which allows for easy connection of the sensors. Again, you can connect up to 16 sensors to one connection. To do this, install the 'libmodbus-dev' package and enter the dev address (e.g. /dev/ttyUSB0) in the 'rs485' file in the OpenSprinkler-Firmware folder.



2.4 Sensors

a. virtual sensors

The following sensors can be used with the OpenSprinklerShop software installed, even without an analog sensor board or Truebner RS485 adapter:

- MQTT: This option can be used to read remote sensors that transfer their data to an MQTT server.

- REMOTE sensor: If you have several OpenSprinklers in a network, you can use this to integrate a sensor from a remote OpenSprinkler and use it as if it were connected locally.
- Weather service sensors: These virtual sensors provide the values of the weather service. Currently supported are temperature, humidity, precipitation and wind.
- Group sensor: This virtual sensor groups the sensor data from several sensors together.
- ‘Free Memory’: For diagnostic purposes only: The free main memory in the OpenSprinkler
- ‘Free Storage’: For diagnostic purposes only: The free non-volatile storage space in the OpenSprinkler
- Raspberry PI only: System temperature

b. Analog sensors

Purely analog sensors that output a voltage in the range of 0..5V can be connected to the analog sensor board. The following sensors are supported:

- SMT100 Analog 0-3V: With the analog version of the SMT100, you can read soil moisture and soil temperature.
- SMT50: This purely analog sensor supports soil moisture and soil temperature.
- Vegetronix VH400: soil moisture
- Vegetronix THERM200: soil temperature
- Vegetronix Aquaplumb: fill level indicator in per cent

In addition, there are these untyped sensors:

- ‘User defined sensor’: Here you can define the sensor ranges, factors and units yourself.
- ‘voltage mode’: direct output of the voltage in volts (e.g. for error detection)
- ‘0..3.3V to 0..100%’: The voltage is output as a percentage, where 3.3V = 100%

c. RS485 sensors

We have developed the ‘Truebner RS485 Adapter’ especially for the professional Truebner RS485 sensors.

Only RS485 Modbus sensors from Truebner can be connected. These are:

- SMT100 RS485 Modbus
- Aquaflex RS485 Modbus.

Since both models technically use the same interface, they are generally referred to as SMT100 in OpenSprinkler. The soil temperature, soil moisture and permittivity values can be read [here](#).

Note: Truebner RS485 sensors have two switchable modes: RS485 Modbus and RS485 ASCII. This mode can be switched using the Truebner diagnostic software.

RS485 Modbus is a bus protocol in which each connected device requires a unique address. The address can be set using the OpenSprinklerASB app or the Truebner diagnostic software. Important: No other device or sensor may be connected during the setting process, otherwise all connected devices would receive the same address.

Tip: Label the sensor with a waterproof pen after setting the number.

New from firmware version 2.3.3 - 173:

- TH100 RS485 mode: This temperature and humidity sensor is supplied in a radiation-shielding housing and has almost the same technical properties as the SMT100. The Modbus address must also be set before use.

3. Connection to the OpenSprinkler

The analog sensor board, like the Truebner RS485 adapter, can only be operated on the OpenSprinkler 3.x with the current OpenSprinklerShop firmware.

The connection is simple: plug the ribbon cable into the left-hand connector of the OpenSprinkler 3.x on the left or right side of the analog sensor board or the Truebner RS485 adapter.

Connect the sensors as shown (see above). To avoid possible short circuits or inductive voltage spikes, disconnect the green plugs from the housing and screw the cables in the disconnected state.

3.1 Installation of the firmware software

Due to contractual regulations, the original software is always installed on the OpenSprinkler 3.x and must be updated manually.

1. If you have already made configurations, save the configuration with 'Export configuration' in the top left menu.
2. Download the current firmware from the OpenSprinklerShop.
3. Go to the update page of your OpenSprinkler:
<http://<ip-address-of-your-OpenSprinkler>/update>. This may look like this, for example: <http://192.168.0.55/update> You can find the IP address of your OpenSprinkler by pressing the B1 button on the device.
4. Now select the previously downloaded firmware, enter the password and confirm with 'Submit'. It takes about 20-40 seconds for the firmware to be installed. You can tell that it is installing by the fact that the OpenSprinkler restarts.
5. Reconnect your OpenSprinkler to the WLAN by connecting to the OS_XYZ network, going to 192.168.4.1 and entering the WLAN password there.
6. Then connect to your OpenSprinkler and – if you made a backup at point 1 – restore it with Import configuration in the menu at the top left.

7. Done. Now your OpenSprinkler is ready. If everything went well, you will find the two new programme items 'Analog Sensor Config' and 'Show Sensor Log' in the bottom right menu. If this is not displayed, please make sure that you a) are using the 'OpenSprinkler ASB' app b) have installed the correct firmware c) For the web browser: Depending on the firmware and configuration, you may also have to update the GUI page. Go to `http://<ip-address-of-your-OpenSprinkler>/su` and enter the following:

8. UI source: <https://ui.opensprinklershop.de/js>

9. Then confirm with your password and 'submit'.

← → ↻ 🏠 🔒 192.168.0.55/su

UI Source:











Weather:

Password:

Information	
Preview Programs	>
View Logs	>
Programs and Settings	
Change Rain Delay	>
Pause Station Runs	>
Run-Once Program	>
Edit Programs	>
Edit Options	>
Analog Sensor Config	>
Show Sensor Log	>
Show Disabled	Stop All Stations

4. Setting up sensors

You can set up sensors using the menu on the bottom right, 'Analog Sensor Configuration'. Here is a brief overview:

Sensors: RS485-Adapter Truebner												
Nr	Type	Group	Name	IP	Port	ID	Read Interval	Data	En	Log	Show	Last
1	1	20	 SMT100-Mois			1	3600	12%	✓	✓	✓	Mon, 10 Feb 2025 22:11:56
2	2	20	 SMT100 Temp			1	3600	4.93°C	✓	✓	✓	Mon, 10 Feb 2025 22:12:10
3	4	30	 TH100-Hum			100	3600	91.79%	✓	✓	✓	Mon, 10 Feb 2025 22:06:47
4	5	31	 TH100-Temp			100	3600	5.89°C	✓	✓	✓	Mon, 10 Feb 2025 22:06:48
 Add Sensor												
 Refresh Sensordata												
 Program Adjustments												
 Monitoring and control												
 Sensor Log												
 Backup and Restore												

In the mobile view, fewer columns are visible.

Each sensor has a unique number > 0. Reusing an existing number will overwrite the configuration.

4.1 Create new sensor

Press 'Add Sensor' to create a new sensor:

Give the sensor a unique number and a meaningful name.

Then select the appropriate sensor. Since the firmware creates the list, the available sensors are different. You can see more about the individual sensors in the next section.

4.2 Set Modbus address

Use the 'Set SMT100 Modbus ID' function to set the Modbus address of the SMT100, Aquaflex and TH100 (RS485 Modbus versions only!). To do this, proceed as follows:

- Disconnect all sensors connected to the Truebner RS485 Modbus or USB Modbus adapter (OSPi), except for the one you want to set up (e.g. disconnect V+).
- Enter the new Modbus address under ID.
- Press 'Set SMT100 Modbus ID'
- Confirm the dialogue.
- Label your sensor with the new Modbus address.

This option is only visible for Modbus sensors.

New Sensor

Edit Sensor Configuration. See help documentation for details.
Last:

Sensor-Nr

Type
Truebner SMT100 RS485 Modbus, moisture mode
Set SMT100 Modbus ID

Group

Name
new sensor

IP Address

Port

ID

Chart Unit
Default

Read Interval (s)
600

Sensor Enabled

Enable Data Logging

Show on Mainpage

4.3 Setting up the sensor

The following sensors are available and can be configured as described:

Type	IP address	Port	ID
RS485 Modbus: Truebner SMT100 RS485 Modbus moisture mode Truebner SMT100 RS485 Modbus, temperature mode Truebner SMT100 RS485 Modbus, permittivity mode Truebner TH100 RS485 Modbus, humidity mode Truebner TH100 RS485 Modbus, temperature mode	Empty for Truebner RS485 adapter (OpenSprinkler 3.x) or USB-Modbus adapter (Raspberry PI), IP address for Modbus RTU adapter (TCP/IP)	Empty for Truebner RS485 adapter (OpenSprinkler) or device offset of USB adapters (Raspberry PI), otherwise 1 for second device. For IP for Modbus RTU, the port address	Modbus address
Analog Sensor Board only: ASB - voltage mode 0..5V ASB - 0..3.3V to 0..100% ASB - SMT50 moisture mode ASB - SMT50 temperature mode ASB - SMT100-analog moisture mode ASB - SMT100-analog temperature mode ASB - Vegetronix VH400 ASB - Vegetronix THERM200 ASB - Vegetronix AquaPlumb ASB - user defined sensor	Empty	Index of the analog sensor board, empty for first device, 1 for second device	Port number on the analog sensor board, where ID=0 corresponds to port number 9
For Raspberry PI OSPi only: OSPi analog input - voltage mode 0..3.3V OSPi analog input - 0.3.3V to 0..100% OSPi analog input - SMT50 moisture mode OSPi analog input - SMT50 temperature mode	Empty	Empty	Connection number of the analog sensor on the OSPi
Generally available via weather data: Weather data - temperature (°F) Weather data - temperature (°C) Weather data - humidity (%) Weather data - precip (inch) Weather data - precip (mm) Weather data - wind (mph) Weather data - wind (kmh)	Not used.	Not used.	Not used.

Type	IP address	Port	ID
Nur OpenSprinkler 3.x: Free Memory Free Storage	Not used.	Not used.	Not used.
Nur Raspberry PI: Internal Raspberry Pi temperature	Not used.	Not used.	Not used.
Virtueller MQTT Sensor: MQTT subscription	Use MQTT Topics and MQTT Filters for data mapping. The MQTT connection from the options is used.		
Remote Sensor: Remote opensprinkler	IP of the remote OpenSprinkler	Port of the remote open sprinkler	No. of the remote OpenSprinkler Sensor
Group sensor: Sensor group with min value Sensor group with max value Sensor group with avg value Sensor group with sum value	Not used.	Not used.	Not used.

Notes on the sensor types:

Type	Notes
RS485 Modbus	This type of sensor must always be connected via an adapter. With OpenSprinkler 3.x, this can be the Truebner RS485 adapter, and with Raspberry Pi, a USB Modbus adapter. However, both devices can also be addressed via an RS485 Modbus RTU adapter if an IP address and port are entered. Each connected Modbus sensor must have a unique Modbus address! ‘Moisture’ = humidity ‘Permittivity’ = soil conductivity, fertiliser content
Remote Sensor	Enter the IP address and the port of the remote OpenSprinkler. The passwords of the connected OpenSprinklers must be the same!
group sensor	A group sensor works like a normal sensor, but its values are combined from the sensors that have the group sensor's sensor number entered under ‘Group’. Only sensors that have the same unit should be grouped! The group value is determined from the set of sensors as - Min: smallest sensor value - Max: largest sensor value - Avg: average of all sensor values - Sum: sum of the sensor values
MQTT	In principle, it works like the remote sensor, but the MQTT protocol is used for this. Connect the OpenSprinkler to your MQTT server or IoT provider via Options / Integration / MQTT and define the topic under which the sensor value is delivered in the sensor. If the IoT provider does not define this exactly, you can use tools such as the ‘MQTT Explorer’ to find out. Define the unit and chart unit as well as an MQTT filter if necessary. This filter must be specified if multiple values are delivered under this topic, e.g. JSON

Type	Notes
	<p>data. If the data is subdivided into multiple levels, the ' ' (pipe) separator can be used to subdivide it. The value must be specified as a decimal number, with the period being specified as the separator for the decimal places.</p> <p>Since MQTT data is processed immediately, the 'Read Interval (s)' value has a different function here: This interval indicates the intervals at which the subscribe command should be sent to the MQTT server.</p>

4.4 Further sensor parameters

Chart Unit

Default ▼

Unit

Read Interval (s)

600 ↕

Sensor Enabled

Enable Data Logging 📄 download log ✕ delete log

Show on Mainpage

Submit

‘Chart Unit’ can be used to determine the unit displayed in the sensor chart (under Analog Sensor Protocol). The same units go into the same diagrams.

‘Unit’ only appears for sensors that have not predefined a unit.

‘Read interval(s)’: Time interval when the values are to be read and stored.

‘Sensor activated’: Only active sensors are displayed and their data stored.

‘Activate data logging’: The read sensor data is stored in a log. Only sensors whose data has been stored can be displayed in the sensor chart.

‘Show on main page’: Activate this option if you want it to be displayed on the zone page (OpenSprinkler home page).

5. programme adjustments

Even the standard OpenSprinkler firmware has powerful, weather-dependent irrigation adjustments. The following configuration options are available:

- Zimmerman: The Zimmerman algorithm is based on the weather values of temperature, humidity and precipitation.
- Automatic rain delay: Requires a rain sensor / internet weather. If rain is detected, watering is suspended for the set time.
- Evapotranspiration (Eto): Algorithm based on water evaporation data calculated by weather satellites.

These algorithms work well as long as weather data is available. That is why the OpenSprinkler offers a weather service that provides weather data free of charge.

The adjustment value of 100% is the basis, which corresponds to the unchanged watering time. A value of 0% means no watering and 150% means a 50% longer watering time.

But the attentive reader will already notice that there is a catch to this. For example, it won't rain in a greenhouse. And neither the plant nor the soil type is taken into account. While you can find appropriate care instructions for the plant species, the soil condition cannot be considered without further tools. Is the soil compacted or permeable to water? What is the nutrient content and do I need to fertilise again? These questions cannot be answered by a weather service. So you need sensors to measure the soil values. And this is where the Analog Sensor API comes in.

The idea behind it is to adjust the irrigation based on the soil values so that the soil moisture always corresponds to the optimal plant growth conditions.

5.1. Creating an adjustment

The 'Programme Adjustments' function allows you to define automated irrigation adjustments.

Nr	Type	S.Nr	Sensor	Name	Program	Factor 1	Factor 2	Min Value	Max Value	Cur
1	1	1	SMT100-Mois	P1	Tau abregnen	80%	0%	3	20	38%

+ Add program adjustment

+ Monitoring and control

+ Sensor Log

+ Backup and Restore

Press 'Add Programme Adjustment' to create a new adjustment.

Each adjustment requires a unique number. Enter a name.

The 'Sensor' defines the sensor values for 'Min. Sensor Value' and 'Max. Sensor Value'

The following values are available as "Type":

- 'No Adjustment': Deactivates the adjustment
- 'Linear scaling': The watering time is scaled linearly. Factor 1 in % for the minimum value, factor 2 for the maximum value
- 'Digital below minimum': The watering time is either 100% if the sensor value is below the minimum value or 0% above it
- 'Digital above maximum': The watering time is either 100% if the sensor value is above the maximum value or 0% below it
- 'Digital below minimum or above maximum': The watering time is either 100% if the sensor value is above the maximum value or 0% below it

The adjustment only applies to the specified programme.

Notice: If you want to combine multiple sensors, then build a sensor group. See help documentation for details.

Adjustment-Nr: 1

Adjustment-Name: P1

Type: Linear scaling

Sensor: 1 - SMT100-Mois

Program to adjust: Tau abregnen

Factor 1 in % (adjustment for min): 80

Factor 2 in % (adjustment for max): 0

Min sensor value: 3

Max sensor value: 20

Adjustment preview

Adjustments in %

SMT100-Mois %

37.65%

Submit

Delete

5.2. programme adjustment - example

For example, if you want to configure watering for a soil moisture level of 15%-25%, then enter:

Min. sensor value: 15%, Max. sensor value: 25%

and as factor 1 '200%' and factor 2 '0%'

This automatically scales watering time from 0s to double the watering time. The closer the soil moisture value is to 25%, the shorter the watering time will be. And the closer it is to 15%, the longer the watering time will be.

5.3 Weather adjustment and programme adjustment

The programme adjustment time can be combined with the global weather adjustment. If, for example, the weather adjustment is set to 150% based on the temperature, but the programme adjustment based on soil moisture calculates 75%, the effective runtime adjustment is calculated as follows:

$$tA \% = (150 / 100 \times 75 / 100) \times 100 = 112.5\%$$

Deactivate the weather adjustment in the programme menu if you do not want to use it and only want to use the sensor adjustments.

6. Monitoring and control

With the 'Monitoring and control' function, you can trigger functions when sensor values reach certain levels.

One application would be to send warnings or control zones when a sensor value reaches a critical level. E.g. temperature above a maximum value, then activate ventilation.

+ Sensoren: RS485-Adapter Truebner									
+ Programm-Anpassungen									
- Überwachung und Kontrolle									
Nr	Typ	S.Nr	Datenquelle	Name	Programm	Zone	Wert 1	Wert 2	Activated
1	12		Temperatur über 20°C EXCLUSIV-ODER Temperatur unter 5°C	Temp >20 oder <5					✓
2	2	2	SMT100 Temp	Temperatur über 20°C			20°C	19°C	
3	1	2	SMT100 Temp	Temperatur unter 5°C			5°C	6°C	✓
4	3		Regen	Regensensor					
5	3		NICHT Regen	Kein Regen					✓
6	10		Kein Regen UND Temp >20 oder <5	Kein Regen und Temp 5..20					✓
+ Neue Überwachung hinzufügen									
+ Sensor Protokoll									
+ Datensicherung									

To simplify the designation of a monitoring and control data set, we have called it 'Monitor'.

6.1 Creating a new monitor

Press 'Add new monitor'. A window will open.

Each monitor requires a unique number, a name, a type, a sensor, and activation and deactivation values. Optionally, you can also enter a programme to be started and a zone.

The 'Type' represents the trigger, i.e. the event that leads to the activation of this monitor. The basis for this is always the current sensor value. The following parameters are available:

1. minimum value: if the sensor value falls below the activation value, this event is activated. It is deactivated as soon as the sensor value exceeds the deactivation value.

2. maximum value: if the sensor value exceeds the activation value, this event is activated. It is deactivated as soon as the sensor value falls below the deactivation value.

3. SN1/2: This stands for a possible digital rain sensor connected to SN1 or SN2. If this type has been selected, 'Digital Sensor Port' appears below and you can select 'Sensor 1' (for SN1) and 'Sensor 2' (for SN2) here.

4. AND, OR or EXCLUSIVE OR: This type is for a logical link of up to four other monitors. For AND, all of the monitors must be activated for OR at least one and for EXCLUSIVE OR exactly one monitor must be activated for this monitor to be activated. The INVERS option allows you to react to monitors that are not activated here.

5. NOT invert the result of another monitor.

6. EXTERN allows the connection of a monitor to another, remote OpenSprinkler. If this option is activated, the IP and Port fields appear. Enter the IP address and the port (OpenSprinkler 3.x : 80, OSPi: 8080) here. The remote OpenSprinkler must have the same password!

New Monitor

Hinweis: Wenn Sie mehrere Sensoren kombinieren möchten, dann bilden Sie eine Sensorgruppe. Weitere Informationen finden Sie in der Hilfedokumentation.

Monitor-Nr
1

Monitor-Name
Zu warm

Typ
Höchstwert

Sensor
999 - Internal Raspi Temp

Zu startendes Programm
Ausgeschaltet

Station wird aktiviert
Ausgeschaltet

Maximale Laufzeit (s)

Priorität
Niedrig

Aktivierungswert
55

Deaktivierungswert
45

Speichern

6.2. Example of use

A horticultural company has greenhouses with ventilation and electric windows.

Tasks:

1. If the air temperature is above 25°, the windows should be opened.
2. If the air temperature is below 25°, the windows should be closed.
3. When it rains, the windows should be closed, regardless of the temperature.
4. When the humidity exceeds 95%, ventilation should be activated.
5. When the humidity falls below 90%, ventilation should be deactivated.

The following equipment is available: an OpenSprinkler DC, a rain sensor, a Truebner RS485 adapter, a TH100 and a 24VDC motor to open and close the electric windows. We will use the 'Schalk DMS5' control module and a 24VDC power supply.

Überwachung und Kontrolle									
Nr	Typ	S.Nr	Datenquelle	Name	Programm	Zone	Wert 1	Wert 2	Activated
2	2	4	TH100-Temp	Temperatur über 25°C			25°C	24°C	
3	1	4	TH100-Temp	Temperatur unter 5°C			5°C	6°C	
4	3		Regen	Regensensor					
5	3		NICHT Regen	Kein Regen					✓
6	10		Kein Regen UND NICHT Temperatur über 25°C UND NICHT Temperatur unter 5°C	Kein Regen und Temp 5..25					✓
7	10		Kein Regen UND Temperatur über 25°C	Fenster öffnen		Fenster öffnen			
8	11		Regensensor ODER NICHT Temperatur über 25°C	Fenster schließen		Fenster schliessen			✓
102	3		TH100-Hum	Luftfeuchtigkeit über 95%		Ventilation	95%	90%	

+ Neue Überwachung hinzufügen

The basic procedure is as follows: First, you define the input sensors and their conditions, i.e. above 25° and rain sensor.

In the next step, you then define what should happen: if it's not raining and it's over 25°, then open the windows. 'Open windows' is a zone (a control output on the OpenSprinkler) that is connected to the DMS5 'open'. The same applies to 'close windows'.

The ventilation is even easier: if it's over 95%, it starts, and if it's under 90%, it switches off again.

7. data logging and charts

Active sensors that have activated the 'Logging' option automatically write their data to the internal log.

There are 3 logs in total:

1. Hourly log: the measurement data with an exact timestamp are stored.
2. Weekly log: the hourly log is reduced to minimum and maximum values each week.
3. Monthly log: The weekly log is reduced to minimum and maximum values each month.

Each of these three logs is divided into two files because the storage space is limited. Each of the three logs consists of two sets of 8,000 data records. When these are full, the oldest file is deleted.

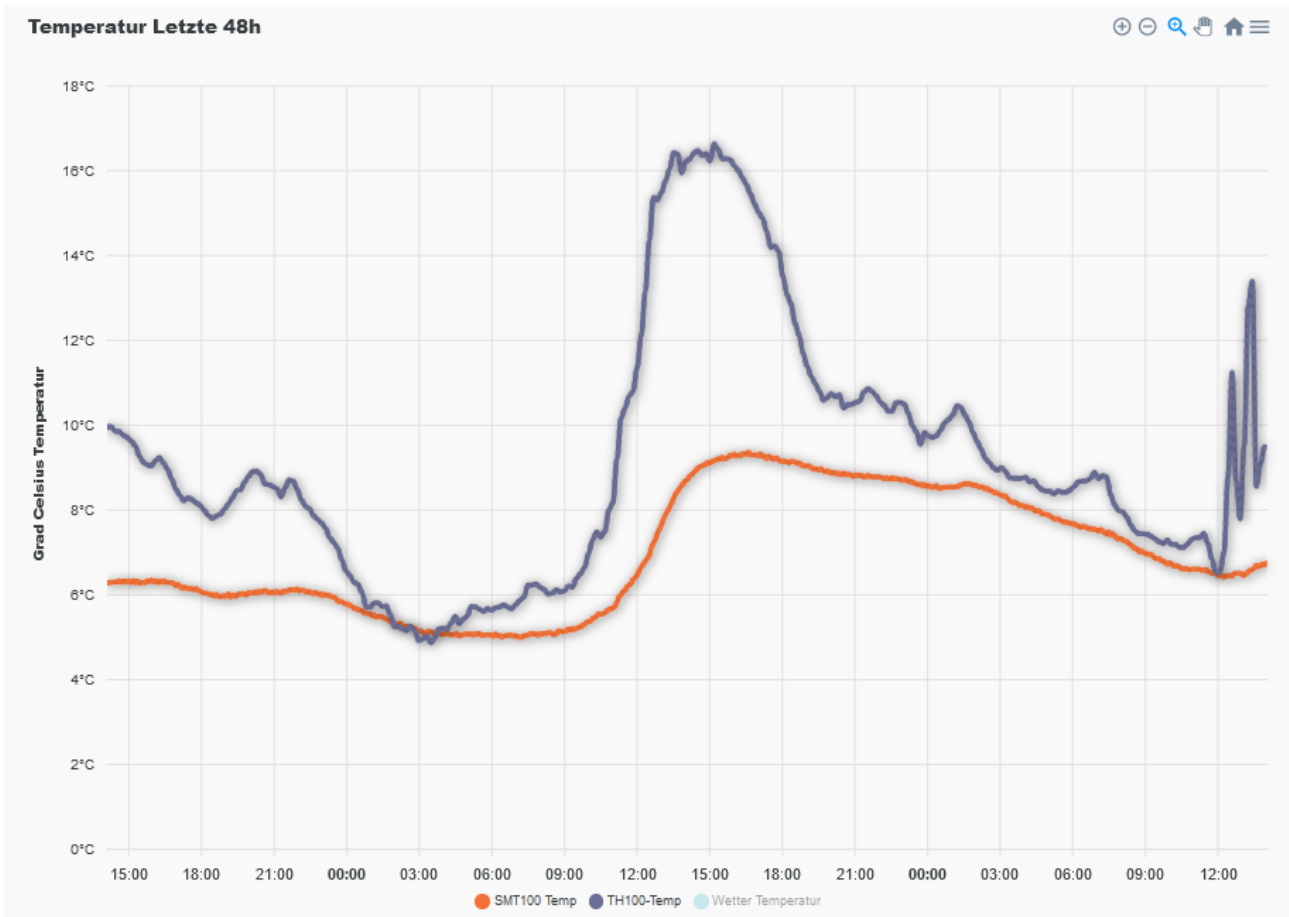
The transfer from hourly log to weekly log and monthly log always takes place at midnight when the day changes. If the OpenSprinkler has been switched off for several days, this also happens shortly after it is switched back on. During this time, it does not respond to commands. The app then displays 'network error'. After a maximum of 5 minutes – depending on the amount of data – it then responds normally again.

Since the storage space in the OpenSprinkler is limited, you should adjust the measurement data intervals to be correspondingly high. Neither the air nor the soil humidity changes in minutes, nor do the outside and soil temperature. A value of 600s (= 10 minutes) should be sufficient as a reading interval.

7.1 Analog sensor protocol

The 'Analog Sensor Protocol' or 'Display Protocol' function can be used to display the measurement data as a diagram.

All three types of protocol (last 48 hours, last weeks, last months) are always displayed. Sensors with the same units are combined in one diagram.

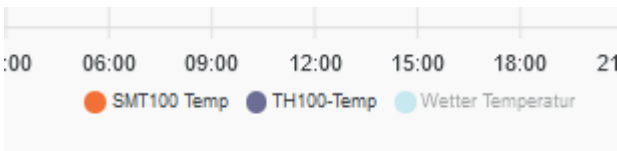


The unit of the y-axis values is shown upright on the left side of the margin. The time unit is shown at the bottom margin; for weekly and monthly values, it is days or months. Controls:

You can use the zoom and move functions in the top right corner.

- '+' and '-' to enlarge/reduce
- Magnifying glass and hand: switch between zoom mode and pan mode
- House: reset view
- Menu: save view as a file, either as a graphic (SVG/PNG) or as data (CSV)

In the lower area, you can switch the displayed sensors on and off:



The names of the sensors correspond, of course, to the names that you have stored in the sensor configuration. In this example, you can hide the sensor 'SMT100 Temp' by clicking on it. This is then displayed in a pale colour, again the sensor 'weather temperature', which is already hidden.

To read a specific diagram value, simply move the mouse onto a measurement value in the diagram.

7.2 Downloading the protocol

The ‘Download protocol’ function can be used to download all stored sensor protocol data.

The format is CSV with semicolon-separated columns.

‘time’ is stored in seconds, it corresponds to the “Unix Time Stamp” but without milliseconds.

The time can be converted here, for example: <https://www.epochconverter.com/>

On this page, you will also find corresponding conversion functions, e.g. for Excel or LibreOffice Calc.

8. Backup and restore

The analog sensor API has a completely separate data storage and is therefore not part of the ‘normal’ configuration backup, which can be found under the menu at the top left.

For this reason, there is a separate backup and restore function to back up and restore the sensor data, programme adjustment data and ‘monitoring and control’ parameters.

Note: Sensor protocol cannot be saved, but it can be downloaded.

Function	Description
Complete fuse	Complete backup of the Sensor API configuration
Recovery complete.	Complete restoration of the Sensor API configuration
Data backup sensors	Backup of sensor configuration only
Restore sensors	Restoring the sensor configuration
Monitoring and control ensure	Save the ‘Monitoring and Control’ configuration.
Restore monitoring and control	Restoring the ‘Monitoring and Control’ configuration

9. API interface

The OpenSprinkler already has a comprehensive standard API interface with many functions. These are described here:

<https://openthings.freshdesk.com/support/solutions/articles/5000716363-os-api-documents>

This API has now been extended to include access to all analog sensor functions. The authorisation and access type have been retained.

The description can be found in the ‘Sensor API.txt’ file in the Github repository:

<https://github.com/opensprinklershop/OpenSprinkler-Firmware/blob/master/Sensor%20API.txt>