

# Smart Agriculture 3.0

## Technical Guide



# INDEX

<b>1. General .....</b>	<b>5</b>
1.1. General and safety information .....	5
1.2. Conditions of use .....	5
<b>2. New version: Smart Agriculture v3.0 .....</b>	<b>6</b>
<b>3. Waspote Plug &amp; Sense!.....</b>	<b>7</b>
3.1. Features .....	7
3.2. General view .....	8
3.2.1. Specifications.....	8
3.3. Parts included.....	11
3.4. Identification.....	12
3.5. Sensor probes.....	14
3.6. Solar powered .....	14
3.7. Programming the Nodes.....	16
3.8. Radio interfaces .....	18
3.9. Models.....	19
3.9.1. Smart Agriculture .....	20
3.9.1.1. Normal .....	21
3.9.1.2. PRO.....	22
<b>4. Hardware.....</b>	<b>23</b>
4.1. General description.....	23
4.2. Specifications .....	23
4.3. Electrical characteristics .....	23
4.4. Agriculture v3.0 Board versions.....	24
<b>5. Sensors .....</b>	<b>25</b>
5.1. Temperature, humidity and pressure sensor (BME280) .....	25
5.1.1. Specifications.....	25
5.1.2. Measurement process .....	26
5.1.3. Socket.....	27
5.2. Leaf Wetness sensor (LWS).....	28
5.2.1. Specifications.....	28
5.2.2. Measurement process .....	28
5.2.3. Socket.....	28
5.3. Soil moisture sensor (Watermark).....	29
5.3.1. Specifications.....	29
5.3.2. Measurement process .....	29
5.3.3. Socket.....	30

5.4. Soil temperature sensor (DS18B20).....	32
5.4.1. Specifications.....	32
5.4.2. Measurement Process.....	32
5.4.3. Socket.....	32
5.5. Soil temperature sensor (Pt-1000) .....	33
5.5.1. Specifications.....	33
5.5.2. Measurement process .....	33
5.5.3. Socket.....	34
5.6. Trunk diameter dendrometer (Ecomatik DC2) .....	34
5.6.1. Specifications.....	34
5.6.2. Measurement process .....	35
5.6.3. Socket.....	35
5.7. Stem diameter dendrometer (Ecomatik DD) .....	36
5.7.1. Specifications.....	36
5.7.2. Measurement process .....	36
5.7.3. Socket.....	37
5.8. Fruit diameter dendrometer (Ecomatik DF) .....	37
5.8.1. Specifications.....	37
5.8.2. Measurement process .....	37
5.8.3. Socket.....	38
5.9. Solar radiation sensor - PAR (SQ-110) .....	39
5.9.1. Specifications.....	39
5.9.2. Measurement process .....	39
5.9.3. Socket.....	41
5.10. Ultraviolet Radiation sensor (SU-100) .....	41
5.10.1. Specifications .....	41
5.10.2. Measurement process.....	42
5.10.3. Socket .....	43
5.11. Weather station (WS-3000) .....	43
5.11.1. Anemometer .....	44
5.11.1.1. Specifications.....	44
5.11.1.2. Measurement process.....	44
5.11.1.3. Socket .....	44
5.11.2. Wind vane.....	45
5.11.2.1. Specifications.....	45
5.11.2.2. Measurement process.....	45
5.11.2.3. Socket .....	46
5.11.3. Pluviometer.....	47
5.11.3.1. Specifications.....	47
5.11.3.2. Measurement process.....	47
5.11.3.3. Socket .....	48
5.12. Ultrasonic Sensor (MaxSonar® from MaxBotix™) .....	49
5.12.1. Specifications .....	49
5.12.2. Measurement Process.....	51
5.12.3. Socket .....	52

---

5.13. Luminosity sensor (TSL2561) .....	53
5.13.1. Specifications .....	53
5.13.2. Measurement process.....	53
5.14. Sockets for casing.....	54
<b>6. Board configuration and programming .....</b>	<b>57</b>
6.1. Hardware configuration .....	57
6.2. API.....	57
6.2.1. Agriculture Class.....	57
6.2.2. dendrometerClass.....	58
6.2.3. ds18b20Class .....	58
6.2.4. leafWetnessClass .....	58
6.2.5. pt1000Class .....	58
6.2.6. radiationClass .....	58
6.2.7. watermarkClass.....	59
6.2.8. weatherStationClass.....	59
<b>7. API changelog .....</b>	<b>60</b>
<b>8. Documentation changelog .....</b>	<b>61</b>
<b>9. Consumption .....</b>	<b>62</b>
9.1. Tables of consumption.....	62
9.2. Low consumption mode .....	62
<b>10. Maintenance .....</b>	<b>63</b>
<b>11. Disposal and recycling .....</b>	<b>64</b>
<b>12. Certifications.....</b>	<b>65</b>
<b>Appendix 1: Watermark sensor's interpretation reference.....</b>	<b>66</b>

# 1. General

## 1.1. General and safety information

- In this section, the term “Waspote” encompasses both the Waspote device itself and its modules and sensor boards.
- Read through the document “General Conditions of Libelium Sale and Use”.
- Do not allow contact of metallic objects with the electronic part to avoid injuries and burns.
- NEVER submerge the device in any liquid.
- Keep the device in a dry place and away from any liquid which may spill.
- Waspote consists of highly sensitive electronics which is accessible to the exterior, handle with great care and avoid bangs or hard brushing against surfaces.
- Check the product specifications section for the maximum allowed power voltage and amperage range and consequently always use a current transformer and a battery which works within that range. Libelium is only responsible for the correct operation of the device with the batteries, power supplies and chargers which it supplies.
- Keep the device within the specified range of temperatures in the specifications section.
- Do not connect or power the device with damaged cables or batteries.
- Place the device in a place only accessible to maintenance personnel (a restricted area).
- Keep children away from the device in all circumstances.
- If there is an electrical failure, disconnect the main switch immediately and disconnect that battery or any other power supply that is being used.
- If using a car lighter as a power supply, be sure to respect the voltage and current data specified in the “Power Supplies” section.
- If using a battery in combination or not with a solar panel as a power supply, be sure to use the voltage and current data specified in the “Power supplies” section.
- If a software or hardware failure occurs, consult the Libelium Web [Development section](#).
- Check that the frequency and power of the communication radio modules together with the integrated antennas are allowed in the area where you want to use the device.
- Waspote is a device to be integrated in a casing so that it is protected from environmental conditions such as light, dust, humidity or sudden changes in temperature. The board supplied “as is” is not recommended for a final installation as the electronic components are open to the air and may be damaged.

## 1.2. Conditions of use

- Read the “General and Safety Information” section carefully and keep the manual for future consultation.
- Use Waspote in accordance with the electrical specifications and the environment described in the “Electrical Data” section of this manual.
- Waspote and its components and modules are supplied as electronic boards to be integrated within a final product. This product must contain an enclosure to protect it from dust, humidity and other environmental interactions. In the event of outside use, this enclosure must be rated at least IP-65.
- Do not place Waspote in contact with metallic surfaces; they could cause short-circuits which will permanently damage it.

Further information you may need can be found at: <http://www.libelium.com/development/waspote>

The “General Conditions of Libelium Sale and Use” document can be found at:

[http://www.libelium.com/development/waspote/technical\\_service](http://www.libelium.com/development/waspote/technical_service)

## 2. New version: Smart Agriculture v3.0

This guide explains the new Smart Agriculture Sensor Board v3.0. This board was specifically designed for our new product lines Waspote v15 and Plug & Sense! v15, released on October 2016.

This board is not compatible with Waspote v12 or Plug & Sense! v12, so it is NOT recommended to mix product generations. If you are using previous versions of our products, please use the corresponding guides, available on our [Development website](#).

You can get more information about the generation change on the document "[New generation of Libelium product lines](#)".

Differences of Smart Agriculture v3.0 with previous versions:

- Included the new BME280 temperature, humidity and air pressure sensor. This tiny, digital and cost-efficient sensor measures 3 parameters with great accuracy and range.
- The BME280 replaces the previous analog Temperature sensor (MCP9700A), the analog Humidity sensor (808H5V5), the double Sensirion sensor (SHT75) and the Atmospheric Pressure sensor (MPX4115A).
- Many internal changes have been made on the board circuitry to reduce the BoM. Now some sensors are multiplexed, avoiding duplicate parts.
- Clearer silkscreen for easier connection.
- Digital Bus socket allows the connections of digital sensors.
- Added an Digital Bus isolator chip to avoid affecting to the Waspote Digital Bus.
- The library has been improved to make easier the board handling.
- New connectors to improve the Plug & Sense! wiring, making it more robust.
- Included ultrasound sensor and luminosity sensor.

## 3. Waspote Plug & Sense!

The Waspote Plug & Sense! line allows you to easily deploy Internet of Things networks in an easy and scalable way, ensuring minimum maintenance costs. The platform consists of a robust waterproof enclosure with specific external sockets to connect the sensors, the solar panel, the antenna and even the USB cable in order to reprogram the node. It has been specially designed to be scalable, easy to deploy and maintain.

**Note:** For a complete reference guide download the “Waspote Plug & Sense! Technical Guide” in the [Development section](#) of the [Libelium website](#).

### 3.1. Features

- Robust waterproof IP65 enclosure
- Add or change a sensor probe in seconds
- Solar powered with internal and external panel options
- Radios available: 802.15.4, 868 MHz, 900 MHz, WiFi, 4G, Sigfox and LoRaWAN
- Over the air programming (OTAP) of multiple nodes at once (via WiFi or 4G radios)
- Special holders and brackets ready for installation in street lights and building fronts
- Graphical and intuitive programming interface Code Generator (coming in 2017)
- Built-in, 3-axes accelerometer
- External, contactless reset with magnet
- External SIM connector for the 4G models
- Fully certified: CE (Europe), FCC (USA), IC (Canada), ANATEL (Brazil), RCM (Australia), PTCRB (USA, cellular connectivity), AT&T (USA, cellular connectivity)



Figure: Waspote Plug & Sense!

## 3.2. General view

This section shows main parts of Waspote Plug & Sense! and a brief description of each one. In later sections all parts will be described deeply.

### 3.2.1. Specifications

- **Material:** polycarbonate
- **Sealing:** polyurethane
- **Cover screws:** stainless steel
- **Ingress protection:** IP65
- **Impact resistance:** IK08
- **Rated insulation voltage AC:** 690 V
- **Rated insulation voltage DC:** 1000 V
- **Heavy metals-free:** Yes
- **Weatherproof:** true - nach UL 746 C
- **Ambient temperature (min.):** -10 °C
- **Ambient temperature (max.):** 50 °C
- **Approximated weight:** 800 g

In the pictures included below it is shown a general view of Waspote Plug & Sense! main parts. Some elements are dedicated to node control, others are designated to sensor connection and other parts are just identification elements. All of them will be described along this guide.

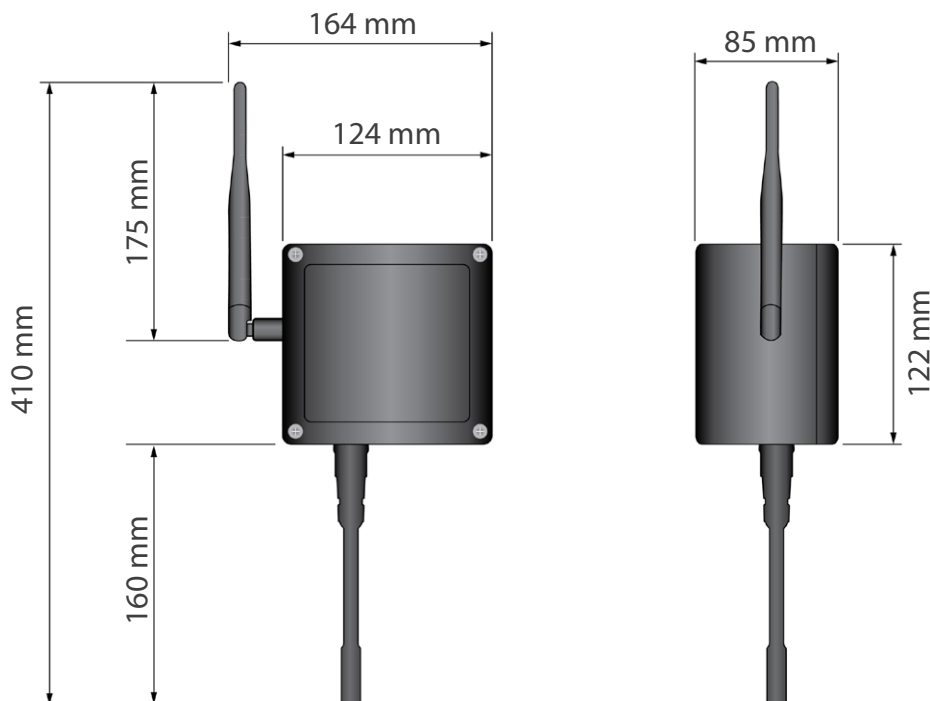


Figure : Main view of Waspote Plug & Sense!



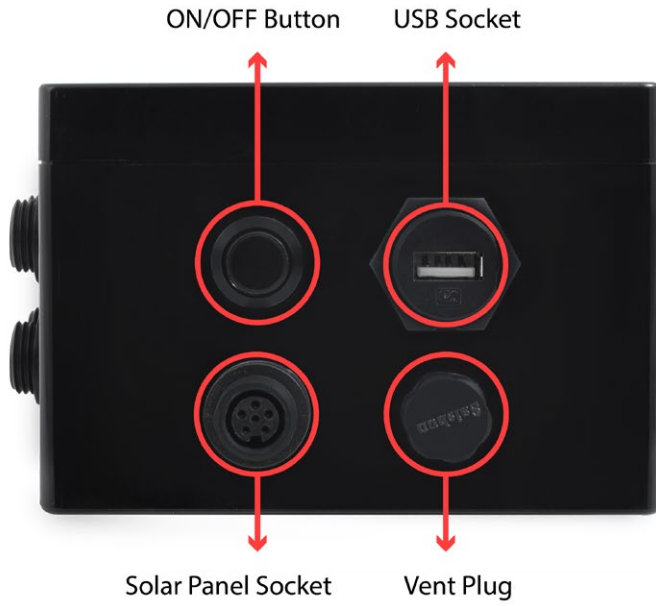


Figure : Control side of the enclosure

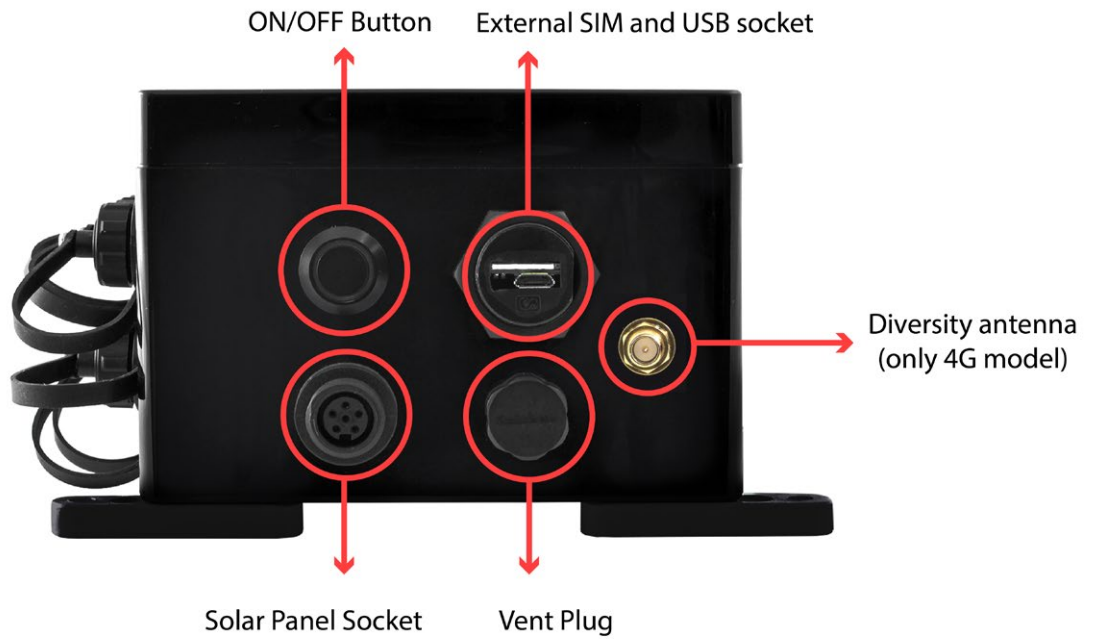


Figure : Control side of the enclosure for 4G model



Figure : Sensor side of the enclosure



Figure : Antenna side of the enclosure



Figure : Front view of the enclosure



Figure : Back view of the enclosure



Figure : Warranty stickers of the enclosure

**Important note:** Do not handle black stickers seals of the enclosure (Warranty stickers). Their integrity is the proof that Wasmote Plug & Sense! has not been opened. If they have been handled, damaged or broken, the warranty is automatically void.

### 3.3. Parts included

Next picture shows Wasmote Plug & Sense! and all of its elements. Some of them are optional accessories that may not be included.



Figure : Wasmote Plug & Sense! accessories: 1 enclosure, 2 sensor probes, 3 external solar panel, 4 USB cable, 5 antenna, 6 cable ties, 7 mounting feet (screwed to the enclosure), 8 extension cord, 9 solar panel cable, 10 wall plugs & screws

### 3.4. Identification

Each Wasmote model is identified by stickers. Next figure shows front sticker.

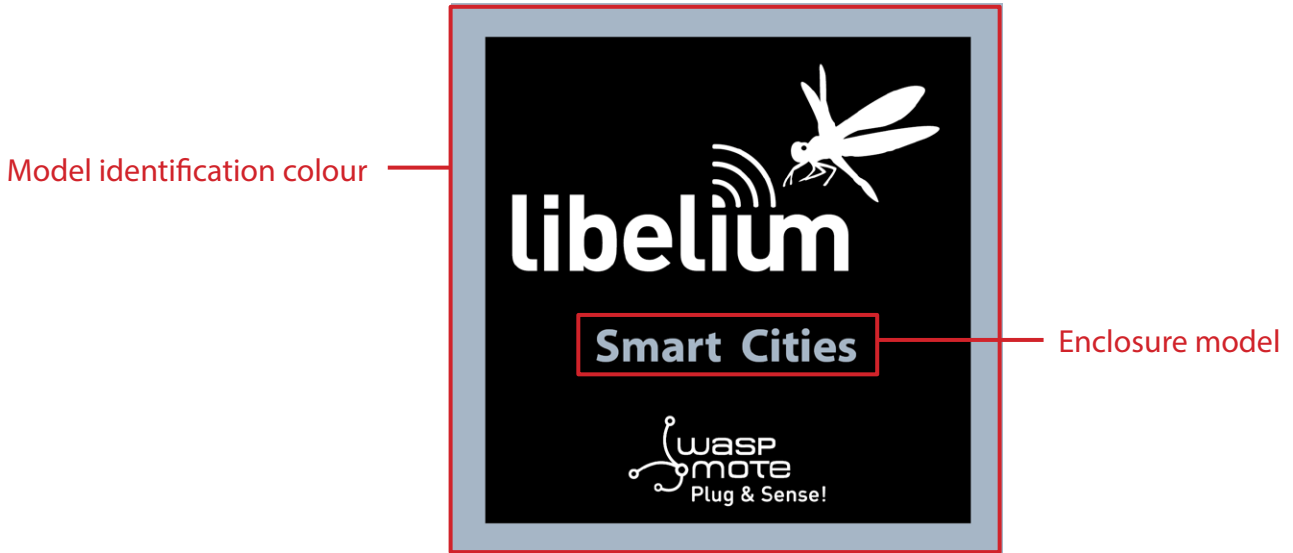


Figure : Front sticker of the enclosure

There are many configurations of Wasmote Plug & Sense! line, all of them identified by one unique sticker. Next image shows all possibilities.



Figure : Different front stickers

Moreover, Wasmote Plug & Sense! includes a back sticker where it is shown identification numbers, radio MAC addresses, etc. It is highly recommended to annotate this information and save it for future maintenance. Next figure shows it in detail.




<b>Plug &amp; Sense! model</b>	<b>Brand name:</b> Libelium	<b>Country of origin:</b> Spain	<b>Version:</b> 1.0
<b>Device serial number</b>	<b>Model:</b> Wasmote Plug & Sense! WiFi		
<b>Battery type</b>	<b>Serial ID:</b> xxxxxxxxx		
<b>Radio type</b>	<b>Battery:</b> 6600 mA·h rechargeable		
<b>Sensor board and extra info</b>	<b>Radio:</b> WiFi		
	<b>Info:</b> Smart Environment		
	<b>FCC ID:</b> XKM-WPS-WIFI-V1		
	<b>IC:</b> 8472A-WPSWIFIV1		
<b>Certifications info</b>	<p>This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.</p> <p>Este equipamento opera em caráter secundário, isto é, não tem direito a proteção contra interferência prejudicial, mesmo de estações do mesmo tipo, e não pode causar interferência a sistemas operando em caráter primário.</p>		
			

Figure : Back sticker

Sensor probes are identified too by a sticker showing the measured parameter and the sensor manufacturer reference.



Figure : Sensor probe identification sticker



### 3.5. Sensor probes

Sensor probes can be easily attached by just screwing them into the bottom sockets. This allows you to add new sensing capabilities to existing networks just in minutes. In the same way, sensor probes may be easily replaced in order to ensure the lowest maintenance cost of the sensor network.



Figure : Connecting a sensor probe to Wasmote Plug & Sense!

Go to the [Plug & Sense! Sensor Guide](#) to know more about our sensor probes.

### 3.6. Solar powered

The battery can be recharged using the waterproof USB cable but also the internal or external solar panel options.

The external solar panel is mounted on a 45° holder which ensures the maximum performance of each outdoor installation.



Figure : Wasmote Plug & Sense! powered by an external solar panel

For the internal option, the solar panel is embedded on the front of the enclosure, perfect for use where space is a major challenge.



*Figure : Internal solar panel*



*Figure : Waspote Plug & Sense! powered by an internal solar panel*

### 3.7. Programming the Nodes

Wasmote Plug & Sense! can be reprogrammed in two ways:

The basic programming is done from the USB port. Just connect the USB to the specific external socket and then to the computer to upload the new firmware.

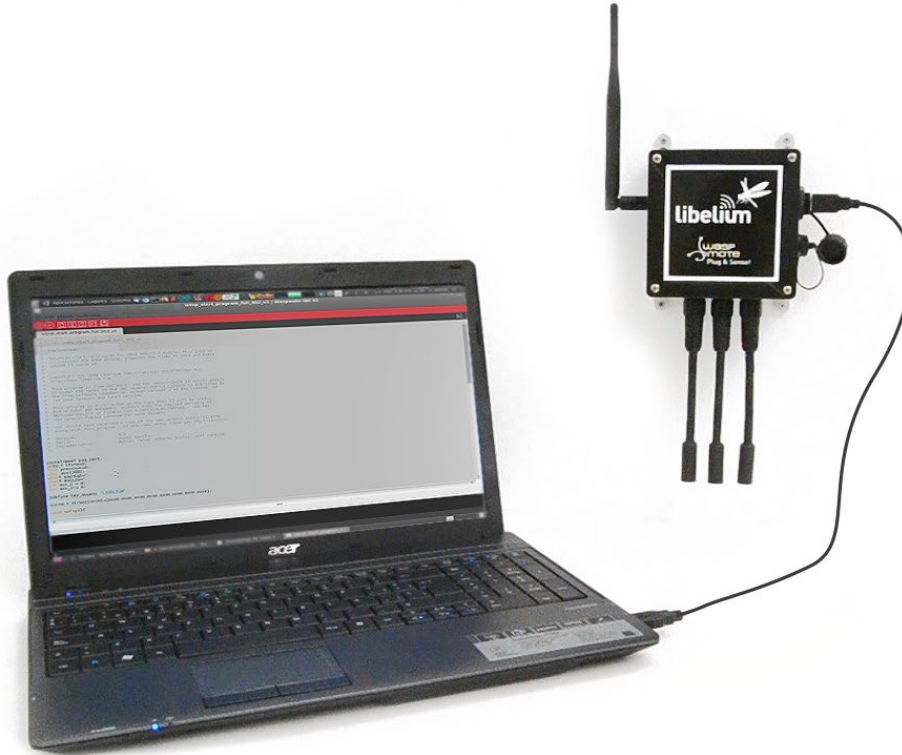


Figure : Programming a node

Besides, Libelium is developing a graphical and intuitive programming interface, the Code Generator (coming in 2017).

#### Wasmote - Plug & Sense! - Code Generator

<b>* Select Model</b> Board model: <input type="text" value="Select"/>	<b>* Sleeping Time</b> Time (seconds): <input type="text"/>	<b>* Select sensor by socket</b> A: <input type="text" value="Select"/> B: <input type="text" value="Select"/> C: <input type="text" value="Select"/> D: <input type="text" value="Select"/> E: <input type="text" value="Select"/> F: <input type="text" value="Select"/>
--	---	--

**Additional information**  
Add Accelerometer 3 Axis data:  Add GPS coordinates:  Wasmote identifier (nodeID):  (Max 10 characters)

**Select Communication Module**  
Communication Module:

**\* Mandatory fields**

Figure: Code Generator web application



Over the Air Programming (OTAP) is also possible once the node has been installed (via WiFi or 4G radios). With this technique you can reprogram, wireless, one or more Waspote sensor nodes at the same time by using a laptop and Meshlium.

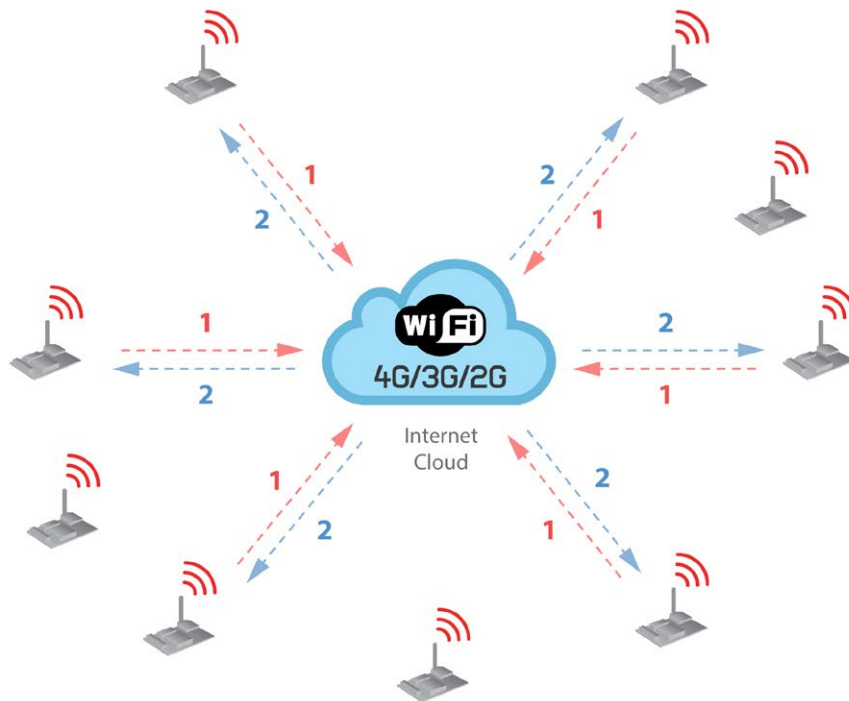


Figure : Typical OTAP process

## 3.8. Radio interfaces

Radio	Protocol	Frequency bands	Transmission power	Sensitivity	Range*	Certification
XBee-PRO 802.15.4 EU	802.15.4	2.4 GHz	10 dBm	-100 dBm	750 m	CE
XBee-PRO 802.15.4	802.15.4	2.4 GHz	18 dBm	-100 dBm	1600 m	FCC, IC, ANATEL, RCM
XBee 868LP	RF	868 MHz	14 dBm	-106 dBm	8.4 km	CE
XBee 900HP US	RF	900 MHz	24 dBm	-110 dBm	15.5 km	FCC, IC
XBee 900HP BR	RF	900 MHz	24 dBm	-110 dBm	15.5 km	ANATEL
XBee 900HP AU	RF	900 MHz	24 dBm	-110 dBm	15.5 km	RCM
WiFi	WiFi (HTTP(S), FTP, TCP, UDP)	2.4 GHz	17 dBm	-94 dBm	500 m	CE, FCC, IC, ANATEL, RCM
4G EU/BR	4G/3G/2G  (HTTP, FTP, TCP, UDP)  GPS	800, 850, 900, 1800, 2100, 2600 MHz	4G: class 3 (0.2 W, 23 dBm)	4G: -102 dBm	- km - Typical base station range	CE, ANATEL
4G US	4G/3G/2G (HTTP, FTP, TCP, UDP)  GPS	700, 850, 1700, 1900 MHz	4G: class 3 (0.2 W, 23 dBm)	4G: -103 dBm	- km - Typical base station range	FCC, IC, PTCRB, AT&T
4G AU	4G (HTTP, FTP, TCP, UDP)	700, 1800, 2600 MHz	4G: class 3 (0.2 W, 23 dBm)	4G: -102 dBm	- km - Typical base station range	RCM
Sigfox EU	Sigfox	868 MHz	16 dBm	-126 dBm	- km - Typical base station range	CE
Sigfox US	Sigfox	900 MHz	24 dBm	-127 dBm	- km - Typical base station range	FCC, IC
LoRaWAN EU	LoRaWAN	868 MHz	14 dBm	-136 dBm	> 15 km	CE
LoRaWAN US	LoRaWAN	900 MHz	18.5 dBm	-136 dBm	> 15 km	FCC, IC

\* Line of sight and Fresnel zone clearance with 5dBi dipole antenna.

## 3.9. Models

There are some defined configurations of Wasmote Plug & Sense! depending on which sensors are going to be used. Wasmote Plug & Sense! configurations allow to connect up to six sensor probes at the same time.

Each model takes a different conditioning circuit to enable the sensor integration. For this reason each model allows to connect just its specific sensors.

This section describes each model configuration in detail, showing the sensors which can be used in each case and how to connect them to Wasmote. In many cases, the sensor sockets accept the connection of more than one sensor probe. See the compatibility table for each model configuration to choose the best probe combination for the application.

It is very important to remark that each socket is designed only for one specific sensor, so **they are not interchangeable**. Always be sure you connected probes in the right socket, otherwise they can be damaged.

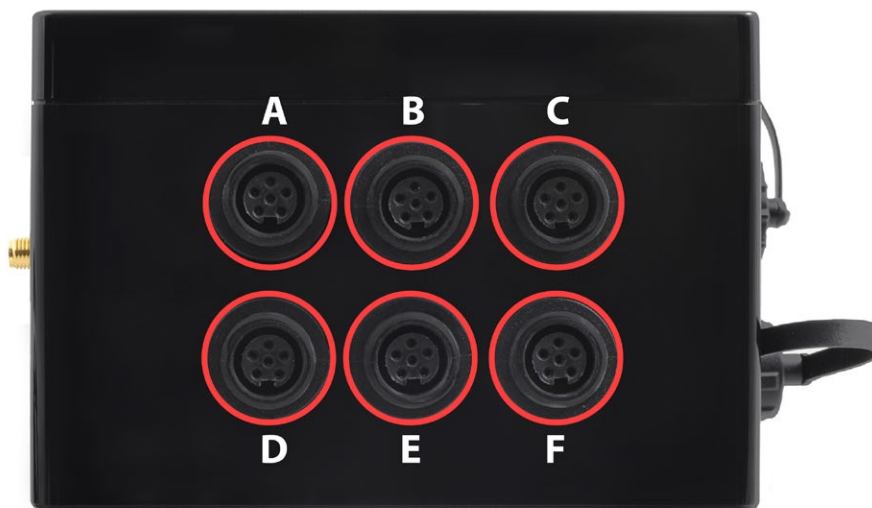


Figure : Identification of sensor sockets

### 3.9.1. Smart Agriculture

The Smart Agriculture models allow to monitor multiple environmental parameters involving a wide range of applications. It has been provided with sensors for air and soil temperature and humidity, solar visible radiation, wind speed and direction, rainfall, atmospheric pressure, etc.

The main applications for this Wasmote Plug & Sense! model are precision agriculture, irrigation systems, greenhouses, weather stations, etc. Refer to [Libelium website](#) for more information.

Two variants are possible for this model, normal and PRO. Next section describes each configuration in detail.



Figure: Smart Agriculture Wasmote Plug & Sense! model

### 3.9.1.1. Normal

Sensor sockets are configured as shown in the figure below.

Sensor Socket	Sensor probes allowed for each sensor socket	
	Parameter	Reference
A	Weather Station WS-3000 (anemometer + wind vane + pluviometer)	9256-P
B	Soil Moisture 1	9248-P, 9324-P, 9323-P
C	Soil Moisture 3	9248-P, 9324-P, 9323-P
D	Soil Temperature	86949-P
	Temperature + Humidity + Pressure	9370-P
	Luminosity (Luxes accuracy)	9325-P
	Ultrasound (distance measurement)	9246-P
E	Leaf Wetness	9249-P
	Soil Moisture 2	9248-P, 9324-P, 9323-P
F (digital bus)	Temperature + Humidity + Pressure	9370-P
	Luminosity (Luxes accuracy)	9325-P
	Ultrasound (distance measurement)	9246-P

Figure: Sensor sockets configuration for Smart Agriculture model

**Note:** For more technical information about each sensor probe go to the [Development section](#) on the Libelium website.

### 3.9.1.2. PRO

Sensor sockets are configured as shown in the figure below.

Sensor Socket	Sensor probes allowed for each sensor socket	
	Parameter	Reference
A	Weather Station WS-3000 (anemometer + wind vane + pluviometer)	9256-P
B	Soil Mosture 1	9248-P, 9324-P, 9323-P
	Solar Radiation (PAR)	9251-P
	Ultraviolet Radiation	9257-P
C	Soil Mosture 3	9248-P, 9324-P, 9323-P
	Dendrometers	9252-P, 9253-P, 9254-P
D (digital bus)	Soil Temperature (Pt-1000)	9255-P
	Temperature + Humidity + Pressure	9370-P
	Luminosity (Luxes accuracy)	9325-P
	Ultrasound (distance measurement)	9246-P
E	Leaf Wetness	9249-P
	Soil Moisture 2	9248-P, 9324-P, 9323-P
F (digital bus)	Temperature + Humidity + Pressure	9370-P
	Luminosity (Luxes accuracy)	9325-P
	Ultrasound (distance measurement)	9246-P

Figure: Sensor sockets configuration for Smart Agriculture PRO model

\* Ask Libelium [Sales Department](#) for more information.

**Note:** For more technical information about each sensor probe go to the [Development section](#) on the Libelium website.

## 4. Hardware

### 4.1. General description

The Waspote Agriculture v3.0 Board allows to monitor **multiple environmental parameters** involving a wide range of applications, from growing development analysis to weather observation. For this, it has been provided with sensors for air and soil temperature and humidity, luminosity, solar radiation (PAR and UV), wind speed and direction, rainfall, atmospheric pressure, leaf wetness, distance and fruit or trunk diameter (dendrometer). Up to **15 sensors can be connected**. With the objective of extending the durability of the device after the deployment, the board is endowed with a solid state switches system that facilitates a precise regulation of its power, prolonging the life of the battery.

### 4.2. Specifications

**Weight:** 20 g

**Dimensions:** 73.5 x 51 x 1.3 mm

**Temperature Range:** [-20 °C, 65 °C]

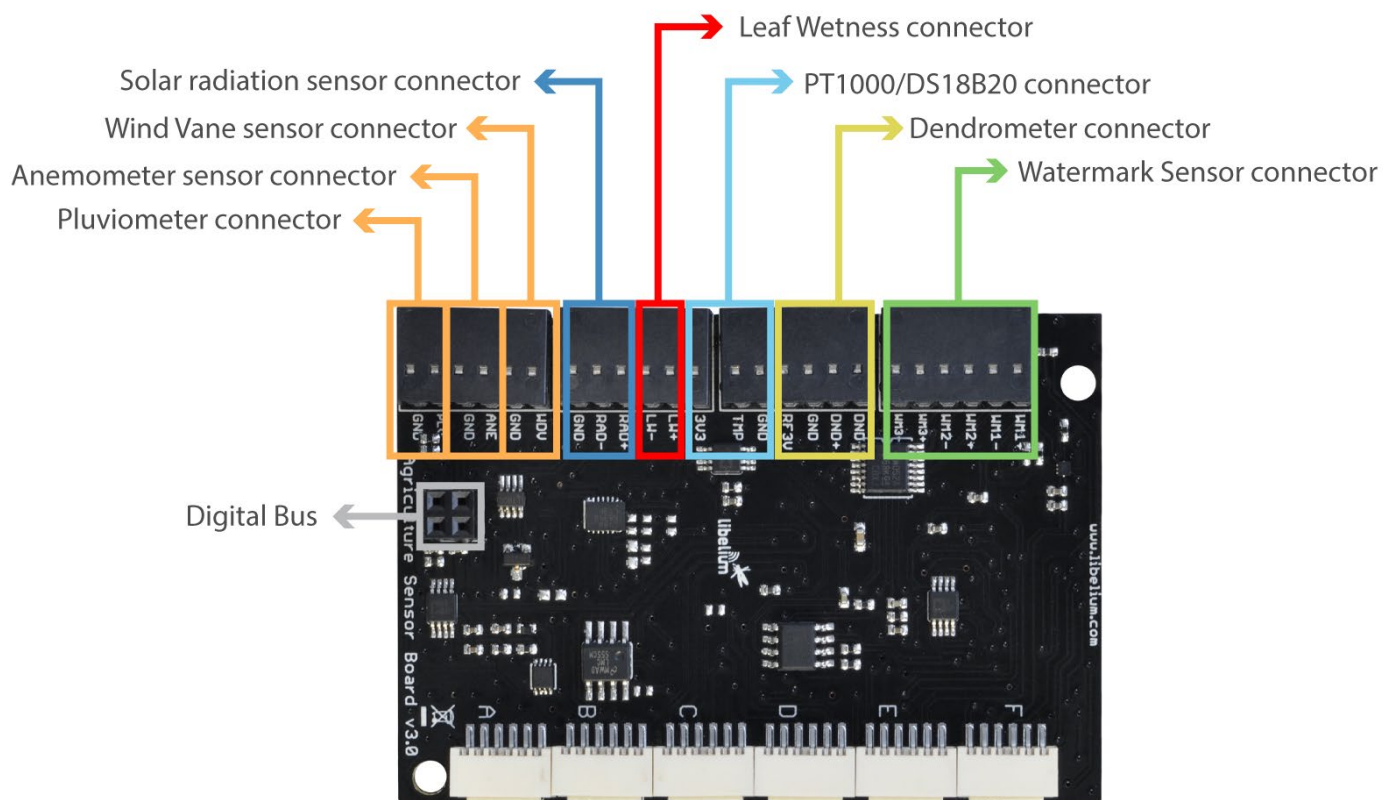


Figure: Upper side

### 4.3. Electrical characteristics

**Board power voltages:** 3.3 V & 5 V

**Sensor power voltages:** 3.3 V & 5 V

**Maximum admitted current (continuous):** 200 mA

**Maximum admitted current (peak):** 400 mA

## 4.4. Agriculture v3.0 Board versions

The Agriculture v3.0 Board for Waspote includes all the electronics and sockets necessary to connect the most typical sensors in agriculture applications, air temperature, humidity and atmospheric pressure (BME280), soil moisture (Watermark), soil temperature (DS18B20) leaf wetness (LWS), Weather Station (WS-3000 includes pluviometer, anemometer and vane), luxes sensor (TSL2561) and distance (MB7040 or MB7070). The PRO version of the board includes the necessary components for more specific applications, such as the solar radiation sensor (SQ-110 or SU-100), the dendrometers (DD, DC2 and DF), and the soil temperature sensor Pt-1000.

Sensors in the Agriculture v3.0 Board ("Standard" or "Normal" version):

- Temperature, humidity and atmospheric pressure sensor BME280 by Bosch
- Soil moisture sensor Watermark by Irrrometer
- Leaf wetness sensor LWS
- Weather Station WS-3000 (Anemometer, Wind Vane and Pluviometer)
- Soil temperature sensor DS18B20
- Distance sensor (MB7040 or MB7070) by MaxBotix.
- Luminosity sensor TSL2561.

Sensors added in the PRO version (includes advanced electronics for integrating special sensors):

- Solar radiation sensor SQ-110 by Apogee
- Ultraviolet radiation sensor SU-100 by Apogee
- DC2, DD and DF dendrometers by Ecomatik
- Soil temperature sensor Pt-1000



## 5. Sensors

### 5.1. Temperature, humidity and pressure sensor (BME280)

**Note:** Only one BME280 sensor is supported at the same time.

#### 5.1.1. Specifications

##### Electrical characteristics

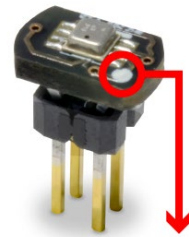
- Supply voltage: 3.3 V
- Sleep current typical: 0.1  $\mu$ A
- Sleep current maximum: 0.3  $\mu$ A

##### Temperature sensor

- Operational range: -40 ~ +85 °C
- Full accuracy range: 0 ~ +65 °C
- Accuracy:  $\pm 1$  °C (range 0 °C ~ +65 °C)
- Response time: 1.65 seconds (63% response from +30 to +125 °C).
- Typical consumption: 1  $\mu$ A measuring

##### Humidity sensor

- Measurement range: 0 ~ 100% of Relative Humidity (for temperatures < 0 °C and > 60 °C see figure below)
- Accuracy: <  $\pm 3\%$  RH (at 25 °C, range 20 ~ 80%)
- Hysteresis:  $\pm 1\%$  RH
- Operating temperature: -40 ~ +85 °C
- Response time (63% of step 90% to 0% or 0% to 90%): 1 second
- Typical consumption: 1.8  $\mu$ A measuring
- Maximum consumption: 2.8  $\mu$ A measuring



Reference mark

Figure : BME280 sensor

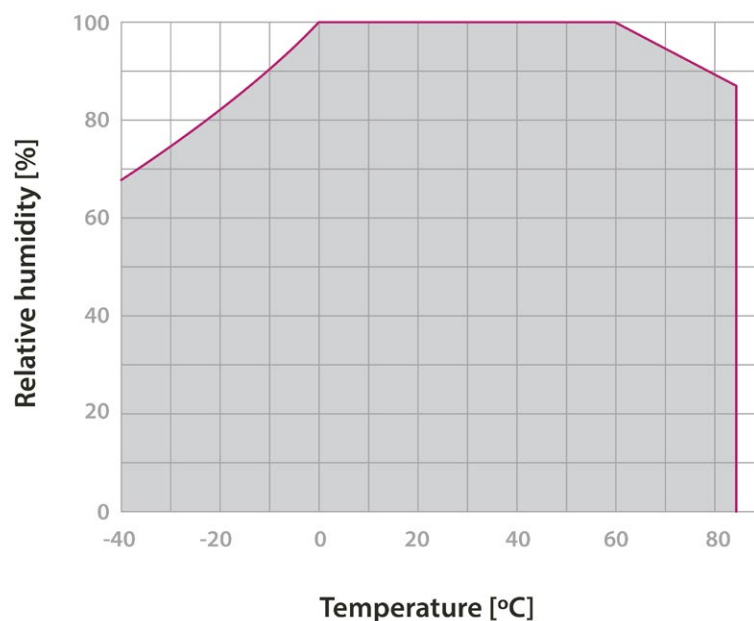


Figure : Humidity sensor operating range

### Pressure sensor

- Measurement range: 30 ~ 110 kPa
- Operational temperature range: -40 ~ +85 °C
- Full accuracy temperature range: 0 ~ +65 °C
- Absolute accuracy:  $\pm 0.1$  kPa (0 ~ 65 °C)
- Typical consumption: 2.8  $\mu$ A measuring
- Maximum consumption: 4.2  $\mu$ A measuring

## 5.1.2. Measurement process

The BME280 is as combined digital humidity, pressure and temperature sensor based on

proven sensing principles. The humidity sensor provides an extremely fast response time for fast context awareness applications and high overall accuracy over a wide temperature range.

The pressure sensor is an absolute barometric pressure sensor with extremely high accuracy and resolution and drastically lower noise.

The integrated temperature sensor has been optimized for lowest noise and highest resolution.

Its output is used for temperature compensation of the pressure and humidity sensors and can also be used for estimation of the ambient temperature.

When the sensor is disabled, current consumption drops to 0.1  $\mu$ A.

To read this sensor, you should use the following functions:

### Temperature

- Reading code:

```
{
  float temp;
  Agriculture.ON();
  // Reads the temperature from BME280 sensor
  temp = Agriculture.getTemperature();
}
```

### Humidity:

- Reading code:

```
{
  float humd;
  Agriculture.ON();
  // Reads the humidity from BME280 sensor
  humd = Agriculture.getHumidity();
}
```

### Pressure:

- Reading code:

```
{
  float pres;
  Agriculture.ON();
  // Reads the pressure from BME280 sensor
  pres = Agriculture.getPressure();
}
```

You can find a complete example code for reading the BME280 sensor in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-01-temperature-sensor-reading>

### 5.1.3. Socket

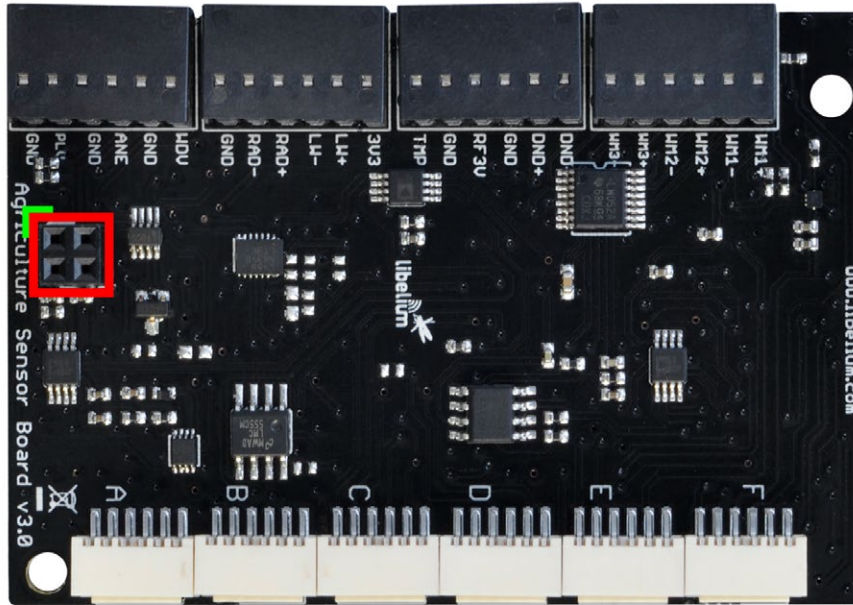


Figure : Image of the socket for the BME280 sensor

In the image above we can see highlighted the four pins of the terminal block where the sensor must be connected to the board. The white dot on the BME280 must match the mark on the Agriculture sensor board.

## 5.2. Leaf Wetness sensor (LWS)

### 5.2.1. Specifications

**Resistance Range:** 5 kΩ ~ >2 MΩ

**Output Voltage Range:** 1 V ~ 3.3 V

**Length:** 3.95 cm

**Width:** 1.95 cm



Figure : Leaf Wetness sensor

### 5.2.2. Measurement process

The leaf wetness sensor behaves as a resistance of a very high value (infinite, for practical purposes) in absence of condensation in the conductive combs that make it up, and that may fall down when it is completely submerged in water. The voltage at its output is inversely proportional to the humidity condensed on the sensor, and can be read at the analog input of Wasp mote **ANALOG3** when selected the proper output of a multiplexer that connects this sensor and one of the Watermark soil humidity sensors to that analog pin. The value returned by the reading function of the library corresponds to the percentage of condensation present on the sensor. The power supply of the sensor (3.3 V) can be cut off or connected through the switched controlled by the digital pin **ANALOG7**. It is highly recommended to switch off this sensor in order to minimize the global consumption of the board (you can find more information about the consumption of the board and how to handle it in chapter “Board configuration and programming”).

Reading code:

```
{
  // Variable to store the read value
  float value_lw;
  // Instance sensor object
  leafWetnessClass lwSensor;
  Agriculture.ON();
  // Read the leaf wetness sensor
  value_lw = lwSensor.getLeafWetness();
}
```

You can find a complete example code for reading the leaf wetness sensor in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-02-leaf-wetness-sensor-reading>

### 5.2.3. Socket

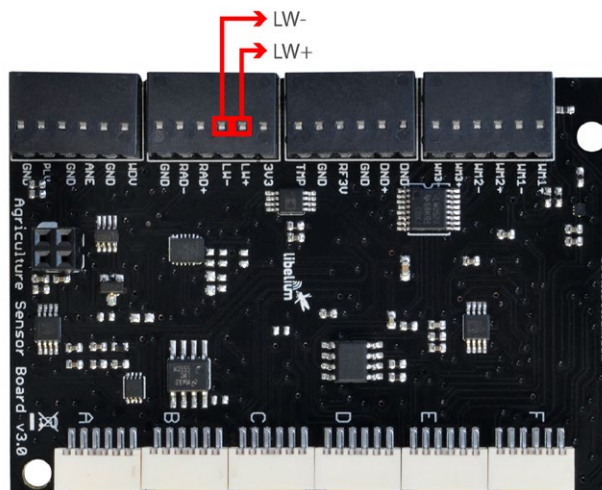


Figure : Image of the socket for the Leaf Wetness sensor

In the image above we can see highlighted the two pins of the terminal block where the sensor must be connected to the board.

## 5.3. Soil moisture sensor (Watermark)

### 5.3.1. Specifications

**Measurement range:** 0 ~ 200 cb

**Frequency Range:** 50 ~ 10000 Hz approximately

**Diameter:** 22 mm

**Length:** 76 mm

**Terminals:** AWG 20



Figure : Watermark sensor

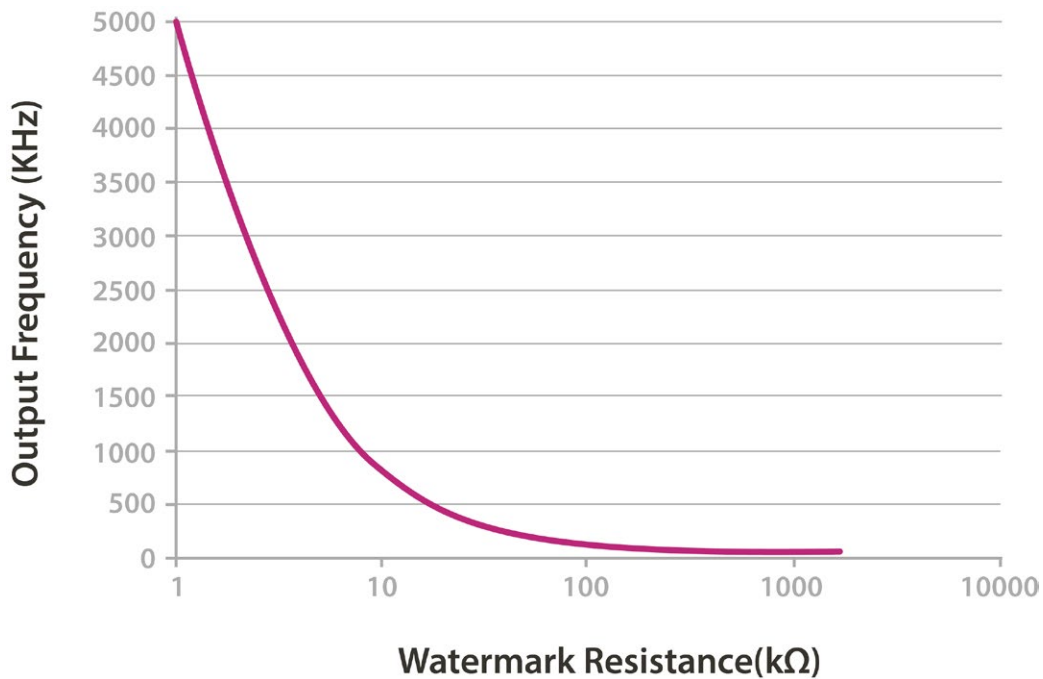


Figure : Output frequency of the Watermark sensor circuit with respect to the resistance of the sensor

### 5.3.2. Measurement process

The Watermark sensor by Irrrometer is a resistive type sensor consisting of two electrodes highly resistant to corrosion embedded in a granular matrix below a gypsum wafer. The resistance value of the sensor is proportional to the soil water tension, a parameter dependent on moisture that reflects the pressure needed to extract the water from the ground. The function of the library `readWatermark()` returns the frequency output of the sensor's adaptation circuit in Herzs (Hz), for more information about the conversion into soil water tension look at Appendix 1 in this technical guide. It is highly recommended to switch off this sensor in order to minimize the global consumption of the board (you can find more information about the consumption of the board and how to handle it in chapter "Board configuration and programming").

Reading code:

```

{
  // Variable to store the read value
  float watermark1, watermark2, watermark3;
  //Instance objects
  watermarkClass wmSensor1(SOCKET_1);
  watermarkClass wmSensor2(SOCKET_2);
  watermarkClass wmSensor3(SOCKET_3);
  Agriculture.ON();

  // Read the Watermarks sensors one by one
  USB.println(F("Wait for Watermark 1..."));
  watermark1 = wmSensor1.readWatermark();
  USB.println(F("Wait for Watermark 2..."));
  watermark2 = wmSensor2.readWatermark();
  USB.println(F("Wait for Watermark 3..."));
  watermark3 = wmSensor3.readWatermark();

  // Print the watermark measures
  USB.print(F("Watermark 1 - Frequency: "));
  USB.print(watermark1);
  USB.println(F(" Hz"));
  USB.print(F("Watermark 2 - Frequency: "));
  USB.print(watermark2);
  USB.println(F(" Hz"));
  USB.print(F("Watermark 3 - Frequency: "));
  USB.print(watermark3);
  USB.println(F(" Hz"));
}
    
```

You can find a complete example code for reading the Watermark sensors in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-07-watermark-sensor-reading>

### 5.3.3. Socket

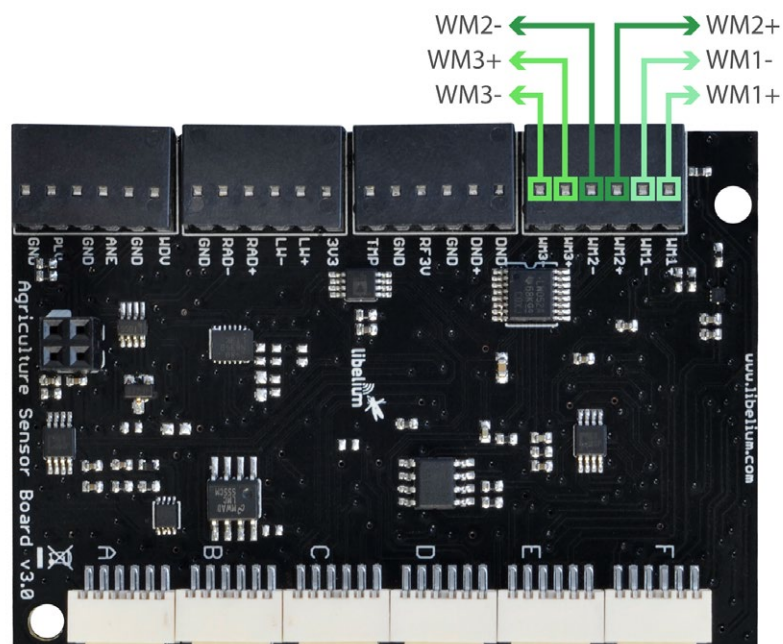


Figure : Image of the socket for the Watermark sensor

Three sockets for Watermark sensors have been placed in the agriculture board (marked in the image in the figure above) and the electronics necessary for powering and signal conditioning, so that the soil moisture can be measured at three different depths.

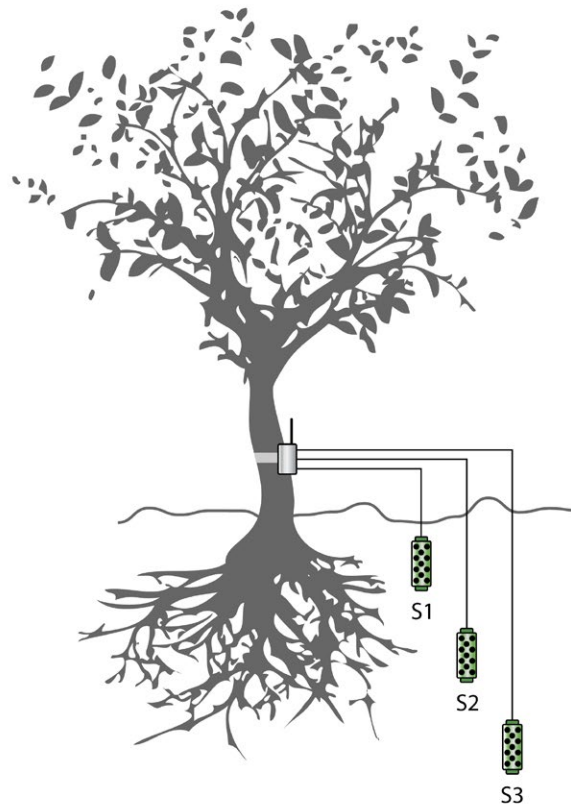


Figure : Illustration of the three Watermark sensors placed at different depths



## 5.4. Soil temperature sensor (DS18B20)

### 5.4.1. Specifications

**Measurement range:** -55 ~ 125 °C

**Accuracy:** ±0.5 °C accuracy from -10 °C to +85 °C

**Diameter:** 7 mm

**Length:** 26 mm

**Cable:** 1.8 m

**Waterproof**

**1- Wire interface**



Figure : DS18B20 sensor

### 5.4.2. Measurement Process

The DS18B20 digital thermometer provides 9-bit Celsius temperature measurements. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with Wasp mote.

This sensor only is included in the Smart Agriculture Normal version.

Reading code:

```
{
  // Variable to store the read value
  float temp;
  // Instance object
  ds18b20Class ds18b20;
  Agriculture.ON();
  // Read the temperature sensor
  temp = ds18b20.readDS18b20();
}
```

You can find a complete example code for reading the DS18B20 sensor in the following link:

<http://www.libelium.com/development/waspote/examples/ag-v30-12-ds18b20-temperature-sensor-reading>

### 5.4.3. Socket

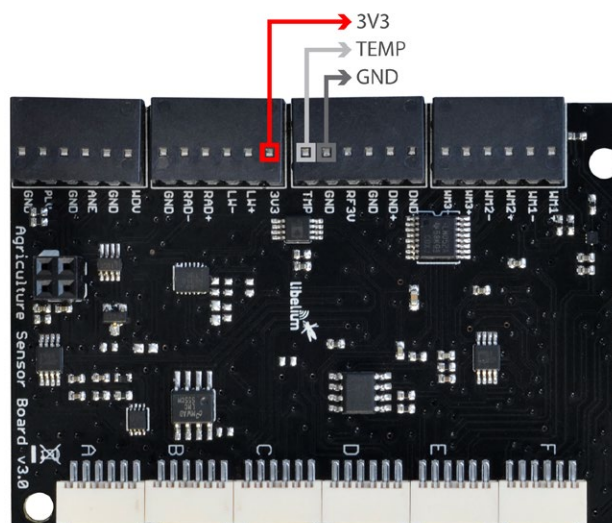


Figure : Image of the socket for the DS18B20 sensor

The sensor must be connected to its adaptation stage through a 2.54 mm pitch. We can see an image with the 3 pins of this socket corresponding to the sensor in the previous figure. The function of the library `readDS18b20()` returns the temperature value in Celsius degree (°C). The power supplies required by the sensor is 3.3 V.



## 5.5. Soil temperature sensor (Pt-1000)

### 5.5.1. Specifications

**Measurement range:** -50 ~ 300 °C

**Accuracy:** DIN EN 60751

**Resistance (0 °C):** 1000 Ω

**Diameter:** 6 mm

**Length:** 40 mm

**Cable:** 2 m



Figure : Pt-1000 sensor

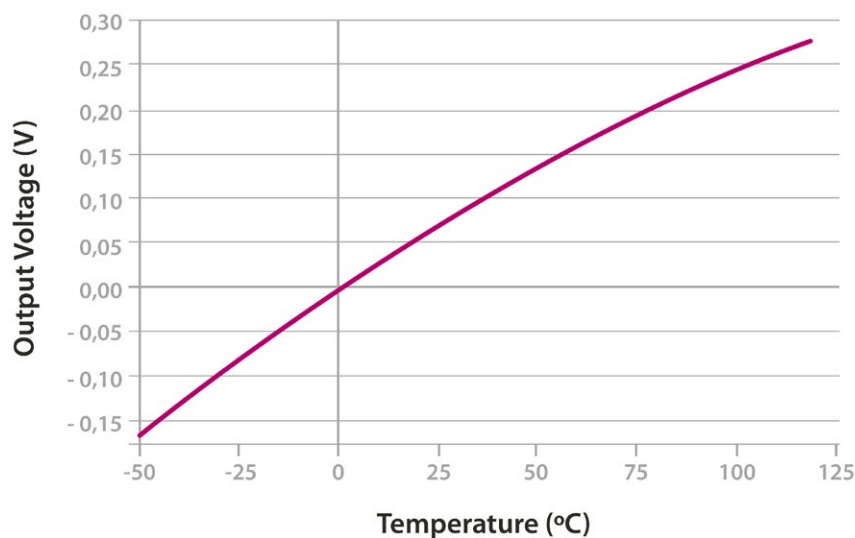


Figure : Output voltage of the Pt-1000 sensor with respect to temperature

### 5.5.2. Measurement process

The resistance of the PT1000 sensor varies between approximately 920 Ω and 1200 Ω in the range considered useful in agriculture applications (-20 ~ 50 °C approximately), which results in too low variations of voltage at significant changes of temperature for the resolution of the Wasp mote's analog-to-digital converter. The function of the library `readPT1000()` returns the temperature value in Celsius degree (°C). The power supplies required by the sensor, both 3.3 V and 5 V, are controlled internally by the library.

This sensor can only be read by Smart Agriculture PRO version.

Reading code:

```
{
  // Variable to store the read value
  float value_PT1000;
  // Instance object
  pt1000Class pt1000Sensor;
  Agriculture.ON();
  // Read PT1000 sensor
  value_PT1000 = pt1000Sensor.readPT1000();
}
```

You can find a complete example code for reading the Pt-1000 sensor in the following link:

<http://www.libelium.com/development/waspote/examples/ag-v30-06-pt1000-sensor-reading>

### 5.5.3. Socket

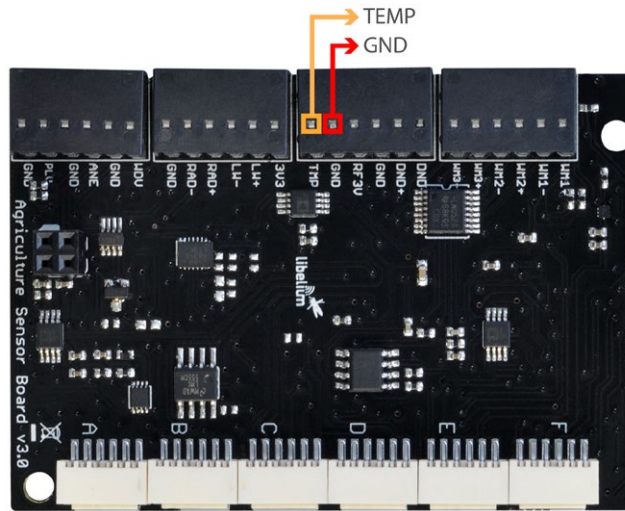


Figure : Image of the socket for the Pt-1000 sensor

The sensor must be connected to its adaptation stage through a 2.54 mm pitch. We can see an image with the two pins of this socket corresponding to the sensor in the previous figure. Both pins of the sensor can be connected to any of the two ways, since there is no polarity to be respected.

## 5.6. Trunk diameter dendrometer (Ecomatik DC2)

### 5.6.1. Specifications

- Trunk/branch diameter:** From 2 cm
- Accuracy:**  $\pm 2 \mu\text{m}$
- Temperature coefficient:**  $< 0.1 \mu\text{m/K}$
- Linearity:**  $< 2\%$
- Operation temperature:**  $-30 \sim 40 \text{ }^\circ\text{C}$
- Operation humidity:**  $0 \sim 100\% \text{ RH}$
- Cable length:** 2 m
- Output range:**  $0 \sim 20 \text{ k}\Omega$
- Range of the sensor:** Function of the size of the tree

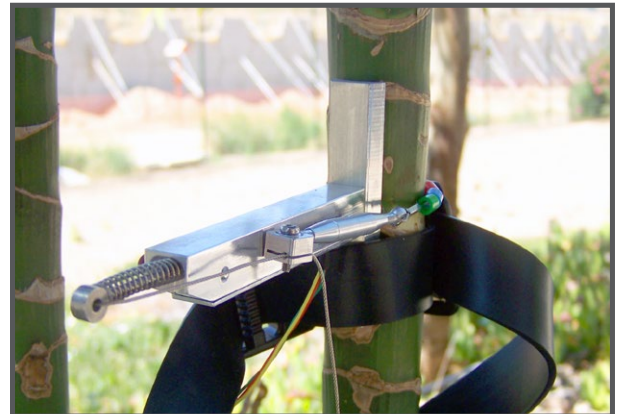


Figure : Ecomatik DC2 sensor

Tree Diameter (cm)	Measuring Range in Circumference (mm)	Measuring Range in Diameter (mm)
10	31.25	9.94
40	22.99	7.31
100	16.58	5.27

## 5.6.2. Measurement process

The operation of the three Ecomatik dendrometers, DC2, DD and DF, is based on the variation of an internal resistance with the pressure that the growing of the trunk, stem, branch or fruit exerts on the sensor. The circuit permits the reading of that resistance in a full bridge configuration through a 16 bits analog-to-digital converter whose reference is provided by a high precision 3V voltage reference in order to acquire the most accurate and stable measurements possible. The reading of the converter, shared with the PT-1000 temperature sensor, will be carried out through the Digital Bus using the functions of the library `WaspSensorAgr_v30` explained in the chapter "API" about this API library, returning the read value in millimeters (mm). The 3.3 V and 5 V power supplies of the devices may be cut off or connected via the digital switch controlled internally by the library.

This sensor can only be read by Smart Agriculture PRO version.

Reading code:

```
{
  // Variable to store the read value
  float value_dendrometer;
  // Instance object
  dendrometerClass dendSensor;
  Agriculture.ON();
  // Read the dendrometer sensor
  value_dendrometer = dendSensor.readDendrometer();
}
```

You can find a complete example code for reading the dendrometer sensor in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-05-dendrometer-sensor-reading>

## 5.6.3. Socket

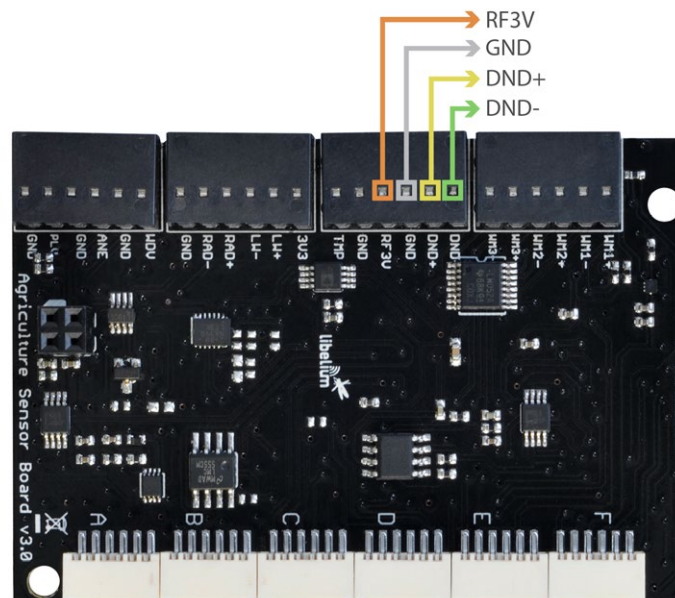


Figure : Image of the socket for the dendrometers

Any of the three dendrometers available may be connected to Waspote through the two 2.54 mm pitch terminal blocks marked in the image in the figure above. The pins corresponding to the sensor in this terminal block (highlighted in the figure) provide the sensor with connection to ground, power supply and to the analog-to-digital converter differential input.

## 5.7. Stem diameter dendrometer (Ecomatik DD)

### 5.7.1. Specifications

**Stem/branch diameter:** 0 ~ 20 cm

**Range of the sensor:** 11 mm

**Output range:** 0 ~ 20 k $\Omega$

**Accuracy:**  $\pm 2$   $\mu\text{m}$

**Temperature coefficient:**  $< 0.1 \mu\text{m/K}$

**Operation temperature:**  $-30 \sim 40$   $^{\circ}\text{C}$

**Operation humidity:** 0 ~ 100% RH

**Cable length:** 2 m



Figure : Ecomatik DD sensor

### 5.7.2. Measurement process

The operation of the three Ecomatik dendrometers, DC2, DD and DF, is based on the variation of an internal resistance with the pressure that the growing of the trunk, stem, branch or fruit exerts on the sensor. The circuit permits the reading of that resistance in a full bridge configuration through a 16 bits analog-to-digital converter whose reference is provided by a high precision 3V voltage reference in order to acquire the most accurate and stable measurements possible. The reading of the converter, shared with the PT1000 temperature sensor, will be carried out through the Digital Bus using the functions of the library `WaspSensorAgr_v30` explained in the chapter "API" about this API library, returning the read value in millimeters (mm). The 3.3 V and 5 V power supplies of the devices may be cut off or connected via the digital switch controlled internally by the library.

This sensor can only be read by Smart Agriculture PRO version.

Reading code:

```
{
  // Variable to store the read value
  float value_dendrometer;
  // Instance object
  dendrometerClass dendSensor;
  Agriculture.ON();
  // Read the dendrometer sensor
  value_dendrometer = dendSensor.readDendrometer();
}
```

You can find a complete example code for reading the dendrometer sensor in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-05-dendrometer-sensor-reading>



### 5.7.3. Socket

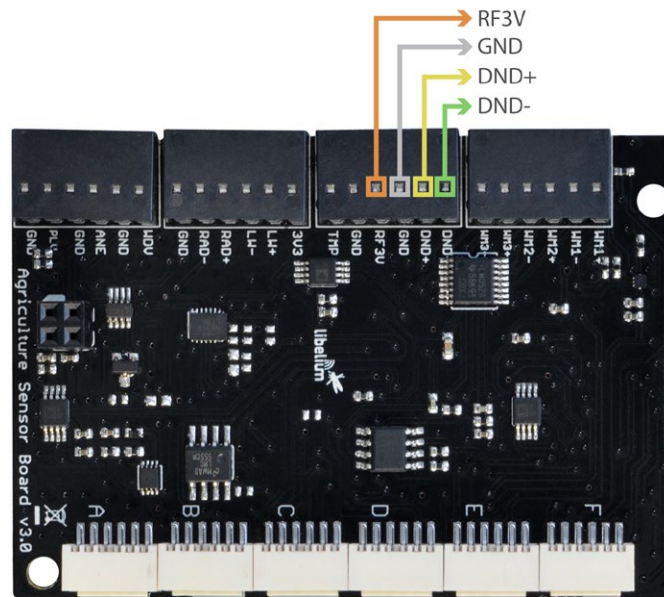


Figure : Image of the socket for the dendrometers

Any of the three dendrometers available may be connected to WaspMote through the two 2.54mm pitch terminal blocks marked in the image in the figure above. The pins corresponding to the sensor in this terminal block (highlighted in the figure) provide the sensor with connection to ground, power supply and to the analog-to-digital converter differential input.

## 5.8. Fruit diameter dendrometer (Ecomatik DF)

### 5.8.1. Specifications

**Fruit diameter:** 0 ~ 11 cm

**Range of the sensor:** 11 mm

**Output range:** 0 ~ 20 k $\Omega$

**Accuracy:**  $\pm 2 \mu\text{m}$

**Temperature coefficient:**  $< 0.1 \mu\text{m/K}$

**Operation temperature:**  $-30 \sim 40 \text{ }^\circ\text{C}$

**Operation humidity:** 0 ~ 100% RH

**Cable length:** 2 m



Figure : Ecomatik DF sensor

### 5.8.2. Measurement process

The operation of the three Ecomatik dendrometers, DC2, DD and DF, is based on the variation of an internal resistance with the pressure that the growing of the trunk, stem, branch or fruit exerts on the sensor. The circuit permits the reading of that resistance in a full bridge configuration through a 16 bits analog-to-digital converter whose reference is provided by a high precision 3V voltage reference in order to acquire the most accurate and stable measurements possible. The reading of the converter, shared with the PT1000 temperature sensor, will be carried out through the Digital Bus using the functions of the library [WaspSensorAgr\\_v30](#) explained in chapter "API" about this API library, returning the read value in millimeters (mm). The 3.3 V and 5 V power supplies of the devices may be cut off or connected via the digital switch controlled internally by the library. It is highly recommended to switch off this sensor in order to minimize the global consumption of the board (you can find more information about the consumption of the board and how to handle it in chapter "Board configuration and programming").

This sensor can only be read by Smart Agriculture PRO version.

Reading code:

```

{
  // Variable to store the read value
  float value_dendrometer;
  // Instance object
  dendrometerClass dendSensor;
  Agriculture.ON();
  // Read the dendrometer sensor
  value_dendrometer = dendSensor.readDendrometer();
}

```

You can find a complete example code for reading the dendrometer sensor in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-05-dendrometer-sensor-reading>

### 5.8.3. Socket

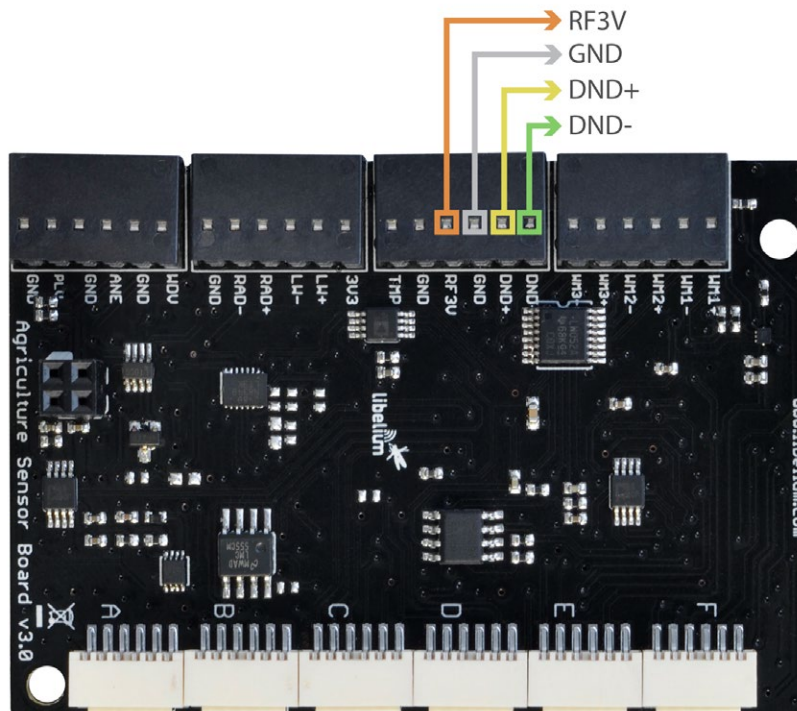


Figure: Image of the socket for the dendrometers

Any of the three dendrometers available may be connected to Waspote through the 2.54 mm pitch terminal blocks marked in the image in the figure above. The pins corresponding to the sensor in this terminal block (highlighted in the figure) provide the sensor with connection to ground, power supply and to the analog-to-digital converter differential input.

## 5.9. Solar radiation sensor - PAR (SQ-110)

### 5.9.1. Specifications

**Sensitivity:** 0.200 mV /  $\mu\text{mol}\cdot\text{m}^{-2}\text{s}^{-1}$

**Calibration factor:** 5  $\mu\text{mol}\cdot\text{m}^{-2}\text{s}^{-1}/\text{mV}$

**Non-linearity:** <1% (up to 4000  $\mu\text{mol}\cdot\text{m}^{-2}\text{s}^{-1}$ )

**Non-stability (long-term drift):** <2% per year

**Spectral range:** 410 ~ 655 nm

**Accuracy:**  $\pm 5\%$

**Repeatability:** <1%

**Diameter:** 2.4 cm

**Height:** 2.8 cm

**Cable length:** 5m of shielded, twisted-pair wire

**Operation temperature:** -40 ~ 70 °C

**Operation humidity:** 0 ~ 100% RH



Figure : SQ-110 sensor

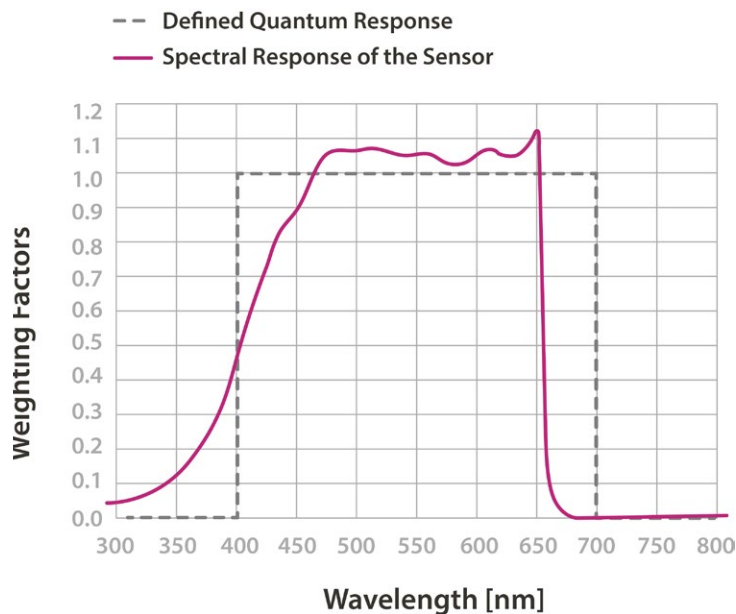


Figure : Graph of the spectral response of the SQ-110 sensor compared to the photosynthetic response of a plant

### 5.9.2. Measurement process

The SQ-110 sensor, specifically calibrated for the detection of solar radiation, provides at its output a voltage proportional to the intensity of the light in the visible range of the spectrum, a key parameter in photosynthesis processes. It presents a maximum output of 400mV under maximum radiation conditions and a sensitivity of 5.00 $\mu\text{mol}\cdot\text{m}^{-2}\text{s}^{-1}/\text{mV}$ . In order to improve the accuracy of the reading, this is carried out through a 16 bits analog-to-digital converter that communicates with the microcontroller of the mote through the Digital Bus. It can be configured or read using the functions implemented in the API library [WaspSensorAgr\\_v30](#) for the Agriculture v3.0 Board. The 5 V power supply of this stage is controlled through a digital switch that can be activated and deactivated using the digital pin DIGITAL7.

This sensor can only be read by Smart Agriculture PRO version.

Reading code:

```
{  
  // Variable to store the radiation conversion value  
  float radiation,value;  
  Agriculture.ON();  
  // Read radiation sensor  
  value = radSensor.readRadiation();  
  // Conversion from voltage into  $\mu\text{mol}\cdot\text{m}^{-2}\cdot\text{s}^{-1}$   
  radiation = value / 0.0002;  
}
```

You can find a complete example code for reading the SQ-110 sensor in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-04-photosynthetic-solar-radiation-sensor-reading>



### 5.9.3. Socket

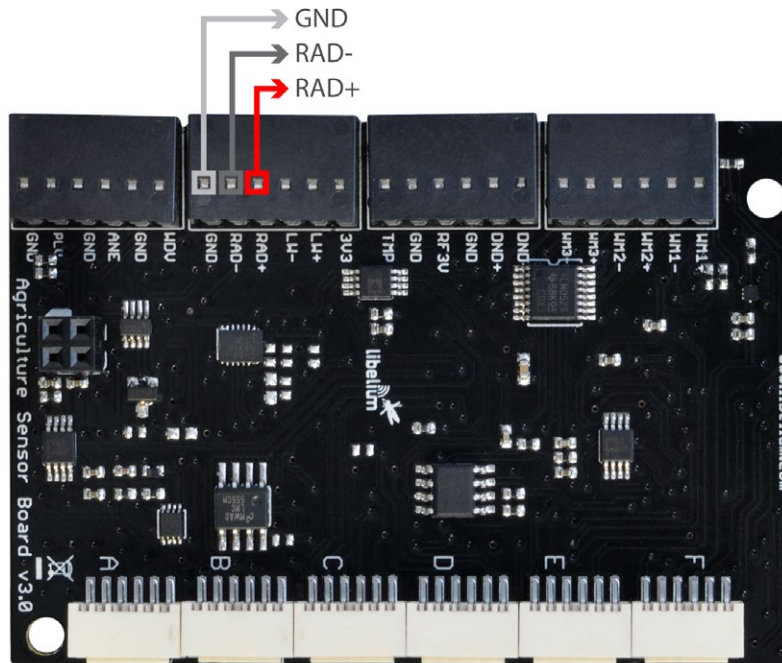


Figure: Image of the socket for the SQ-110 sensor

This sensor is connected to the analog-to-digital converter through the 3 pins of the 2.54 mm pitch terminal block marked in the image in the figure above.

## 5.10. Ultraviolet Radiation sensor (SU-100)

### 5.10.1. Specifications

**Sensitivity:**  $0.2\text{mV} / \mu\text{mol}\cdot\text{m}^{-2}\cdot\text{s}^{-1}$

**Calibration factor:**  $5.0\mu\text{mol}\cdot\text{m}^{-2}\cdot\text{s}^{-1}/\text{mV}$

**Non-stability (Long-term drift):** <3% per year

**Non-linearity:** <1% (up to  $300\mu\text{mol}\cdot\text{m}^{-2}\cdot\text{s}^{-1}$ )

**Spectral range:** 250 ~ 400nm

**Accuracy:**  $\pm 10\%$

**Repeatability:** <1%

**Diameter:** 2.4cm

**Height:** 2.8cm

**Cable length:** 5m shielded, twisted-pair wire

**Operation temperature:** -40 to 70 °C

**Operation humidity:** 0 ~ 100%RH



Figure: SU-100 sensor

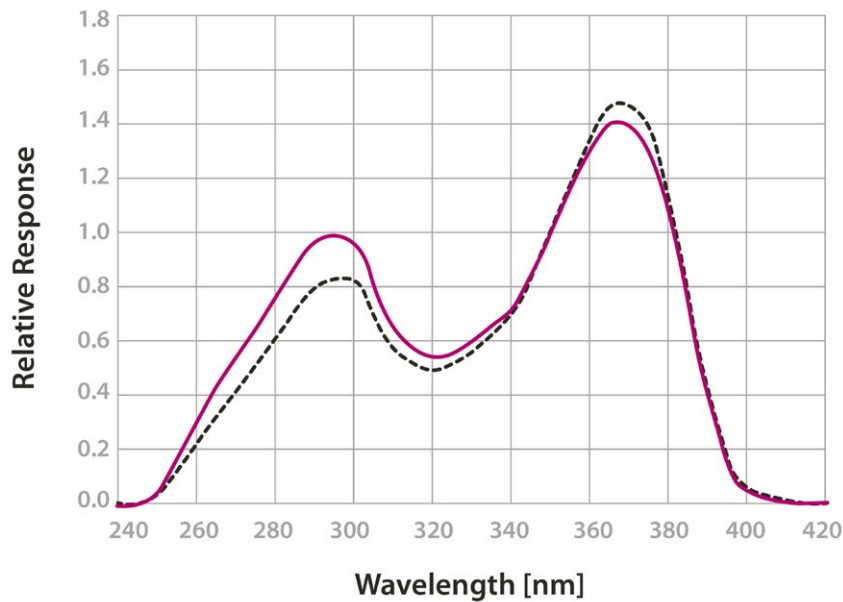


Figure: Graph of the spectral response of the SU-100 sensor compared to the photosynthetic response of a plant

### 5.10.2. Measurement process

The SU-100 sensor, complementary to the SQ-110 sensor, provides at its output a voltage proportional to the intensity of the light in the ultraviolet range of the spectrum. It presents a maximum output of 26mV under maximum radiation conditions and a sensitivity of  $0.2\text{mV} / \mu\text{mol}\cdot\text{m}^{-2}\cdot\text{s}^{-1}$ . This sensor is read by the mote through the same 16 bits analog-to-digital converter used with the SQ-110 sensor. It can be configured or read using the functions implemented in the API library [WaspSensorAgr\\_v30](#) for the Agriculture v3.0 Board. The 5 V power supply of this stage is controlled through a digital switch that can be activated and deactivated internally by the library.

This sensor can only be read by Smart Agriculture PRO version.

Reading code:

```
{
  // Instance object
  radiationClass radSensor;
  // Variable to store the radiation conversion value
  float radiation,value;
  Agriculture.ON();
  // Read radiation sensor
  value = radSensor.readRadiation();
  // Conversion from voltage into  $\mu\text{mol}\cdot\text{m}^{-2}\cdot\text{s}^{-1}$ 
  radiation = value / 0.0002;
}
```

You can find a complete example code for reading the SU-100 sensor in the following link:

[http://www.libelium.com/development/waspote/examples/ag-v30-03-ultraviolet\\_solar\\_radiation\\_sensor\\_reading](http://www.libelium.com/development/waspote/examples/ag-v30-03-ultraviolet_solar_radiation_sensor_reading)

### 5.10.3. Socket

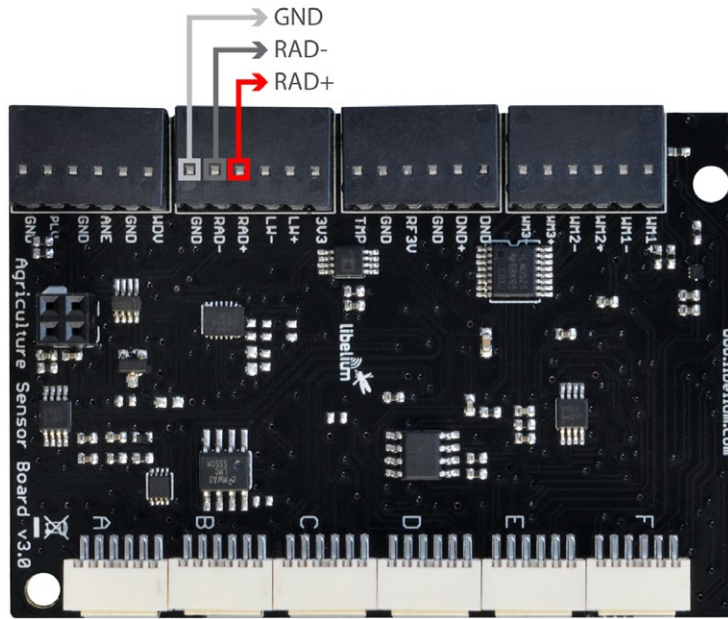


Figure : Image of the socket for the SU-100 sensor

This sensor is connected to the analog-to-digital converter through the three pins of the 2.54 mm pitch terminal block marked in the image in the figure above.

## 5.11. Weather station (WS-3000)

This weather station consists of three different sensors, described in detail below: a wind vane, an anemometer and a pluviometer. It connects to Waspote through six wires that must be connected to the terminal block shown in the corresponding figures, being the anemometer connected to the vane through an RJ11 socket.



Figure : Image of the Weather station WS-3000

## 5.11.1. Anemometer

### 5.11.1.1. Specifications

**Sensitivity:** 2.4km/h / turn  
**Wind Speed Range:** 0 ~ 240km/h  
**Height:** 7.1 cm  
**Arm length:** 8.9 cm  
**Connector:** RJ11



Figure : Anemometer

### 5.11.1.2. Measurement process

The anemometer chosen for Waspote consists of a reed switch normally open that closes for a short period of time when the arms of the anemometer complete a 180° angle, so the output is a digital signal whose frequency will be proportional to the wind speed. There will be 2 events in each complete turn. That signal can be read from the digital pin `DIGITAL2` of the Waspote. The function of the library `readAnemometer()` returns the wind speed value in kilometers per hour (km/h). The power of the sensor and of the electronics around it may be connected or disconnected using a digital switch controlled internally by the library.

Reading code:

```

{
  // Variable to store the anemometer value
  float anemometer;
  // Instance objects
  weatherStationClass anemSensor;
  Agriculture.ON();
  // Read Anemometer sensor
  anemometer = anemSensor.readAnemometer();
}
    
```

You can find a complete example code for reading the whole Weather Station WS-3000 data sensor in the following link: <http://www.libelium.com/development/waspote/examples/ag-v30-08-weather-station-sensor-reading>

### 5.11.1.3. Socket

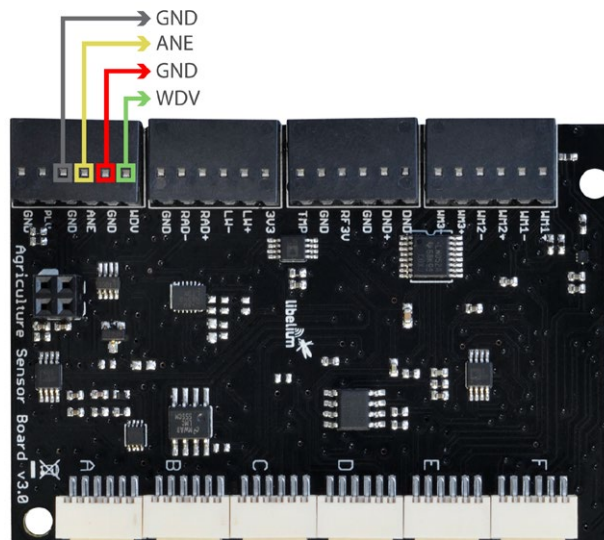


Figure : Image of the connector for the anemometer

The way to connect the anemometer to the Agriculture 3.0 Board is through the vane: the anemometer's cable must be plugged into the socket that can be found on the base of the vane.

## 5.11.2. Wind vane

### 5.11.2.1. Specifications

**Height:** 8.9 cm

**Length:** 17.8 cm

**Maximum accuracy:** 22.5°

**Resistance range:** 688  $\Omega$  ~ 120 k $\Omega$



Figure : Wind vane

### 5.11.2.2. Measurement process

The wind vane consists of a basement that turns freely on a platform endowed with a net of eight resistances connected to eight switches that are normally and are closed (one or two) when a magnet in the basement acts on them, which permits us to distinguish up to 16 different positions (the equivalent to a resolution of 22.5°). The equivalent resistance of the wind vane, along with a 10 k $\Omega$  resistance, form a voltage divider, powered at 3.3 V through a digital switch controlled by the `ANALOG1` pin, whose output can be measured in the analog input `ANALOG5`. The function of the library `readVaneDirection()` also stores in variable `vane_direction` an 8 bits value which corresponds with an identifier of the pointing direction.

Below, a table with the different values that the equivalent resistance of the wind vane may take is shown, along with the direction corresponding to each value:

Direction (degrees)	Resistance (k $\Omega$ )	Voltage (V)	Identifier
0	33	2.53	<code>SENS_AGR_VANE_N</code>
22.5	6.57	1.31	<code>SENS_AGR_VANE_NNE</code>
45	8.2	1.49	<code>SENS_AGR_VANE_NE</code>
67.5	0.891	0.27	<code>SENS_AGR_VANE_ENE</code>
90	1	0.3	<code>SENS_AGR_VANE_E</code>
112.5	0.688	0.21	<code>SENS_AGR_VANE_ESE</code>
135	2.2	0.59	<code>SENS_AGR_VANE_SE</code>
157.5	1.41	0.41	<code>SENS_AGR_VANE_SSE</code>
180	3.9	0.92	<code>SENS_AGR_VANE_S</code>
202.5	3.14	0.79	<code>SENS_AGR_VANE_SSW</code>
225	16	2.03	<code>SENS_AGR_VANE_SW</code>
247.5	14.12	1.93	<code>SENS_AGR_VANE_WSW</code>
270	120	3.05	<code>SENS_AGR_VANE_W</code>
292.5	42.12	2.67	<code>SENS_AGR_VANE_WNW</code>
315	64.9	2.86	<code>SENS_AGR_VANE_NW</code>
337.5	21.88	2.26	<code>SENS_AGR_VANE_NNW</code>

Reading code:

```
{
  // Variable to store the vane value
  int vane;
  // Instance objects
  weatherStationClass vaneSensor;
  Agriculture.ON();
  // Read Vane sensor
  vane = vaneSensor.readVaneDirection();
}
```

You can find a complete example code for reading the whole Weather Station WS-3000 data sensor in the following link:

<http://www.libelium.com/development/waspmote/examples/ag-v30-08-weather-station-sensor-reading>

Besides, it is recommended to use the function `getVaneFiltered()` in order to perform a mean filtered measurement during a specified period of time. Thus, mechanical fluctuations will be avoided and a more accurate measurement will be done. See example here:

<http://www.libelium.com/development/waspmote/examples/ag-v30-09-wind-vane-filtered>

### 5.11.2.3. Socket

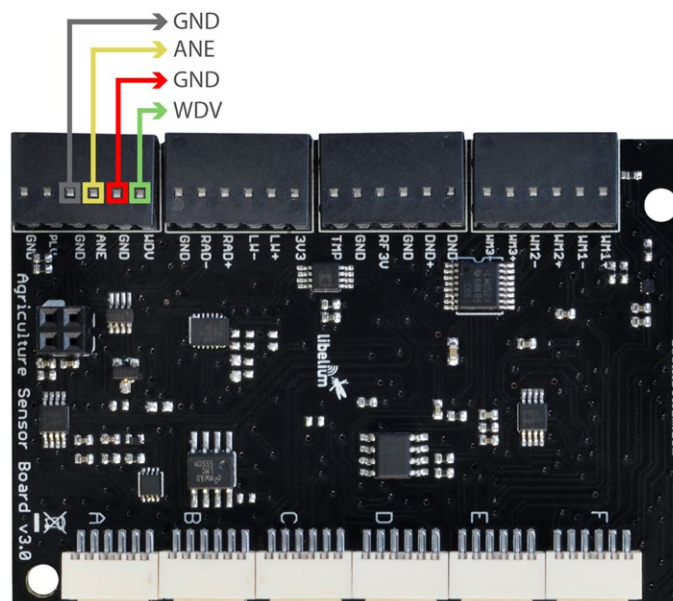


Figure: Image of the connector for the wind vane

The wind vane is connected to the board through four pins of a terminal block whose correspondence with the sensor's wire is shown in the figure above.



## 5.11.3. Pluviometer

### 5.11.3.1. Specifications

**Height:** 9.05 cm

**Length:** 23 cm

**Bucket capacity:** 0.28 mm of rain



### 5.11.3.2. Measurement process

The pluviometer consists of a small bucket that, once completely filled (0.28 mm of water approximately), closes a switch, emptying automatically afterwards. The sensor is connected directly to the Waspote **DIGITAL4** digital input through a pull-up resistance and to the interruption pin **RXD1**, allowing the triggering of an interruption of the microcontroller when the rainfall provokes a bucket emptying event. Since the consumption of this sensor, in absence of rain, is null, no elements of power control have been added for it.

The library defines several functions to measure from the pluviometer. An inner structure is being used in order to store the number of pluviometer interruptions that took place for the last 24 hours (regarding the RTC settings). The function used for storing a new pulse is called `storePulse()` and it should be called every time a new pluviometer interruption is generated. On the other hand, it is possible to know three different values given by the following functions:

`readPluviometerCurrent()` calculates the precipitations in mm for the current period. For example, it is 10:42 am, so this function returns the mm of rainfall that took place in the last 42 minutes.

`readPluviometerHour()` calculates the precipitations in the previous one-hour period. For example, it is 10:42 am, so this function returns the mm of rainfall between 9 am and 10 am.

`readPluviometerDay()` calculates the accumulated precipitation in the last 24 hours.

Tip: the user can apply a little of paraffin on the pluviometer's upper surface in order to help the rain drops to flow down to the inside of the sensor.

You can find a complete example code for reading the whole Weather Station WS-3000 data sensor in the following link:  
[www.libelium.com/development/waspmote/examples/ag-v30-08-weather-station-sensor-reading](http://www.libelium.com/development/waspmote/examples/ag-v30-08-weather-station-sensor-reading)

There is an example of how to use the pluviometer sensor separately from the Weather Station:  
[www.libelium.com/development/waspmote/examples/ag-v30-10-pluviometer-interruption-reading](http://www.libelium.com/development/waspmote/examples/ag-v30-10-pluviometer-interruption-reading)

### 5.11.3.3. Socket

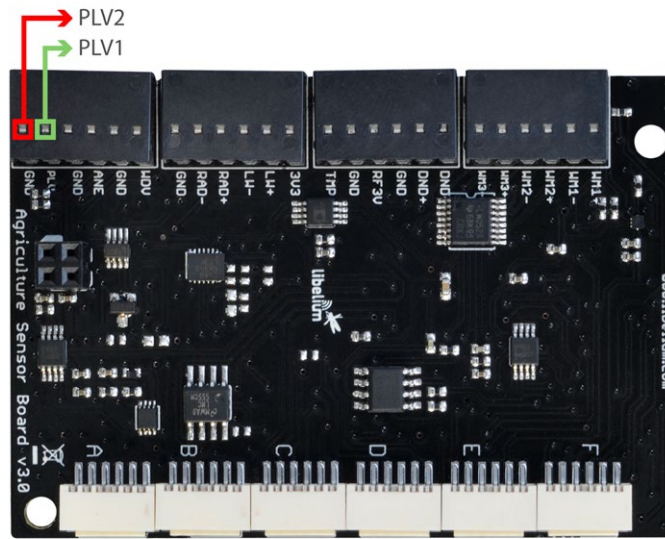


Figure : Image of the socket for the pluviometer

In the image in the figure above we can see marked the two pins in the terminal block that correspond to the pluviometer (no polarity is required).



## 5.12. Ultrasonic Sensor (MaxSonar® from MaxBotix™)

### 5.12.1. Specifications

**I2CXL-MaxSonar®-MB7040™**

**Operation frequency:** 42 kHz

**Maximum detection distance:** 765 cm

**Interface:** Digital Bus

**Power supply:** 3.3 ~ 5 V

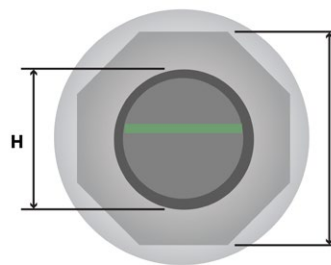
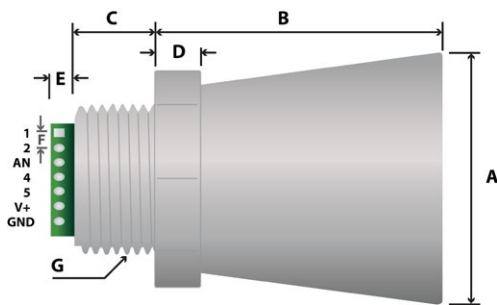
**Consumption (average):** 2.1 mA (powered at 3.3 V) – 3.2 mA (powered at 5 V)

**Consumption (peak):** 50 mA (powered at 3.3 V) – 100 mA (powered at 5 V)

**Usage:** Indoors and outdoors (IP-67)



Figure : Ultrasonic I2CXL-MaxSonar®-MB7040 from MaxBotix™ sensor



A	1.72" dia.	43.8 mm dia.
B	2.00"	50.7 mm
C	0.58"	14.4 mm
D	0.31"	7.9 mm
E	0.18"	4.6 mm
F	0.1"	2.54 mm
G	3/4" National Pipe Thread Straight	
H	1.032" dia.	26.2 dia.
I	1.37"	34.8 mm
weight: 1.76 oz. ; 50 grams		

Figure : Ultrasonic I2CXL-MaxSonar®-MB7040 sensor dimensions

**Note:** Only one MB7040 sensor is supported at the same time.

In the figure below we can see a diagram of the detection range of the sensor developed using different detection patterns (a 0.63 cm diameter dowel for diagram A, a 2.54 cm diameter dowel for diagram B, an 8.25 cm diameter rod for diagram C and a 28 cm wide board for diagram D):

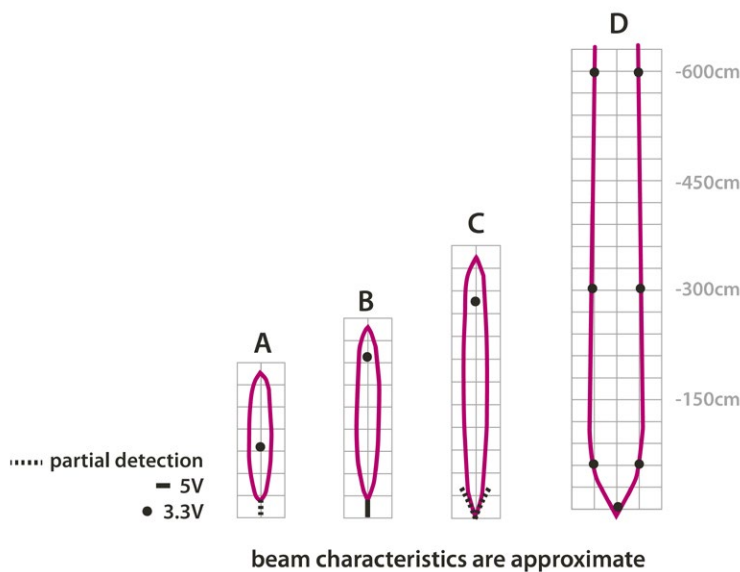


Figure : Diagram of the sensor beam extracted from the data sheet of the XL-MaxSonar®-WRA1™ sensor from MaxBotix

**I2CXL-MaxSonar®-MB1202™**

**Operation frequency:** 42 kHz

**Maximum detection distance:** 645 cm

**Sensitivity (analog output):** 2.5 mV/cm (powered at 3.3 V) – 3.8 mV/cm (powered at 5 V)

**Power supply:** 3.3 ~ 5 V

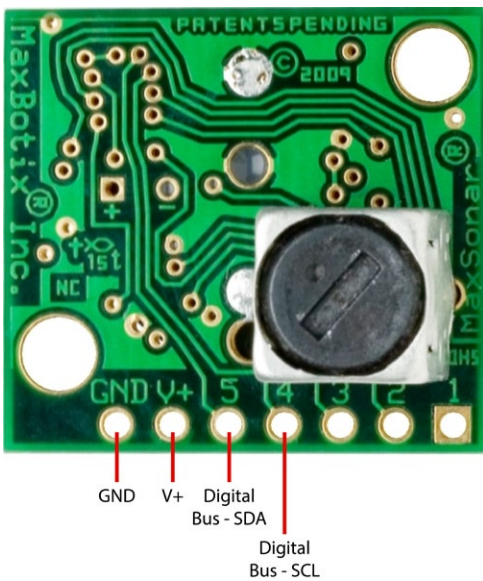
**Consumption (average):** 2 mA (powered at 3.3 V) – 3mA (powered at 5 V)

**Usage:** Indoors



Figure : Ultrasonic I2CXL-MaxSonar®-MB1202 from MaxBotix™ sensor

In the figure below we can see a diagram of the detection range of the sensor developed using different detection patterns (a 0.63 cm diameter dowel for diagram A, a 2.54 cm diameter dowel for diagram B, a 8.25 cm diameter rod for diagram C and a 28 cm wide board for diagram D):



A	0.785"	19.9 mm
B	0.870"	22.1 mm
C	0.100"	2.54 mm
D	0.100"	2.54 mm
E	0.670"	17.0 mm
F	0.510"	12.6 mm
G	0.124" dia.	3.1 mm dia.

H	0.100"	2.54 mm
J	0.645"	16.4 mm
K	0.610"	15.5 mm
L	0.735"	18.7 mm
M	0.065"	1.7 mm
N	0.038" dia.	1.0 mm dia.
		weight: 4.3 grams

Figure : Diagram of the sensor beam extracted from the data sheet of the Ultrasonic I2CXL-MaxSonar®-MB1202 sensor from MaxBotix

## 5.12.2. Measurement Process

The MaxSonar® sensors from MaxBotix connects through digital bus interface.

In the next figure, we can see a drawing of two example applications for the ultrasonic sensors, such as liquid level monitoring or presence detection.

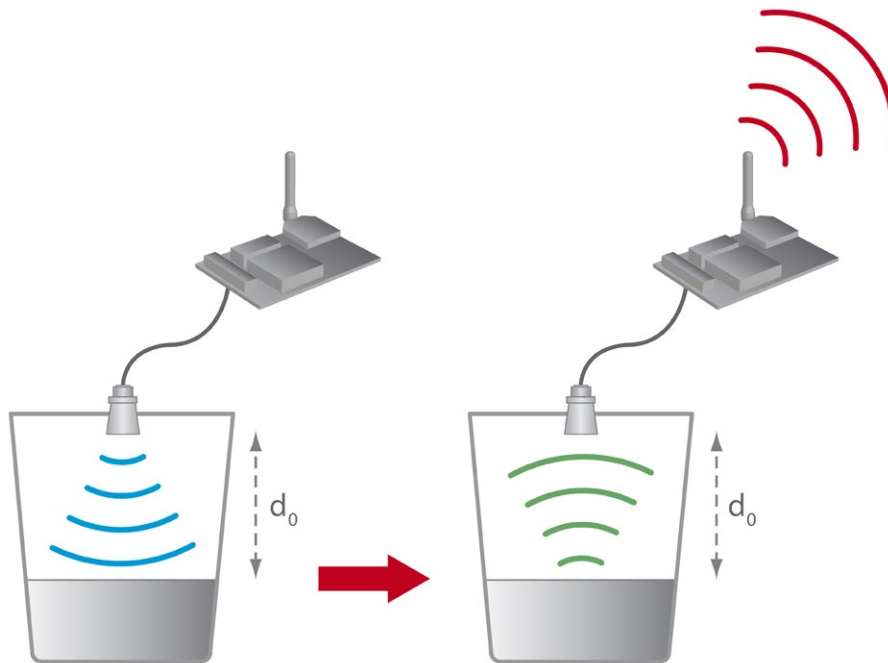


Figure : Examples of application for the MaxSonar® sensors

The XL-MaxSonar®-WRA1TM sensor is endowed with an IP-67 casing, so it can be used in outdoors applications, such as liquid level monitoring in storage tanks.

Below a sample code to measure one of the ultrasound sensors (the XL-MaxSonar®-WRA1) is shown:

Reading Code:

```
{
  uint16_t distance;
  Agriculture.ON();
  distance = Agriculture.getDistance();
}
```

You can find a complete example code for reading the distance in the following link:

[www.libelium.com/development/waspmote/examples/ag\\_v30\\_13\\_ultrasound](http://www.libelium.com/development/waspmote/examples/ag_v30_13_ultrasound)

### 5.12.3. Socket

These sensors share the socket with the BME280 temperature, humidity and pressure sensor. The pin correspondence, highlighted in the figure below, is the same for both.

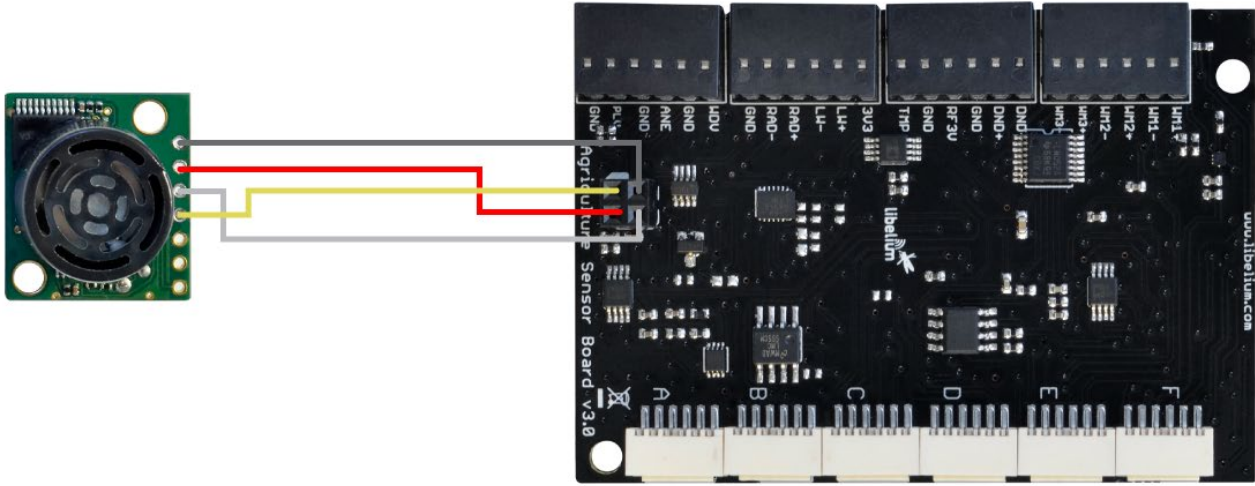


Figure : Image of the socket for connecting the MaxSonar® Sensors

## 5.13. Luminosity sensor (TSL2561)

### 5.13.1. Specifications

**Electrical characteristics**

**Dynamic range:** 0.1 to 40000 lux

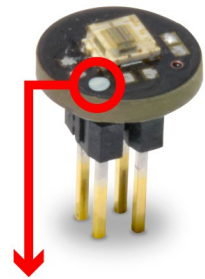
**Spectral range:** 300 ~ 1100 nm

**Voltage range:** 2.7 ~ 3.6V

**Supply current typical:** 0.24 mA

**Sleep current maximum:** 0.3  $\mu$ A

**Operating temperature:** -30 ~ +70  $^{\circ}$ C



Reference mark

Figure : Image of the Luminosity sensor

**Note:** Only one TSL2561 sensor is supported at the same time.

### 5.13.2. Measurement process

This is a light-to-digital converter that transforms light intensity into a digital signal output. This device combines one broadband photo-diode (visible plus infrared) and one infrared-responding photo-diode on a single CMOS integrated circuit capable of providing a near-photopic response over an effective 20-bit dynamic range (16-bit resolution). Two integrating ADCs convert the photo-diode currents to a digital output that represents the irradiance measured on each channel. This digital output in lux is derived using an empirical formula to approximate the human eye response.

Reading Code:

```
{
  uint32_t luxes = 0;
  Agriculture.ON();
  // Reads the luxes sensor
  // Select the option of measure:
  // - OUTDOOR
  // - INDOOR
  luxes = Agriculture.getLuxes(INDOOR);
}
```

You can find a complete example code for reading the luminosity in the following link:

[www.libelium.com/development/waspmote/examples/ag\\_v30\\_14\\_luxes](http://www.libelium.com/development/waspmote/examples/ag_v30_14_luxes)

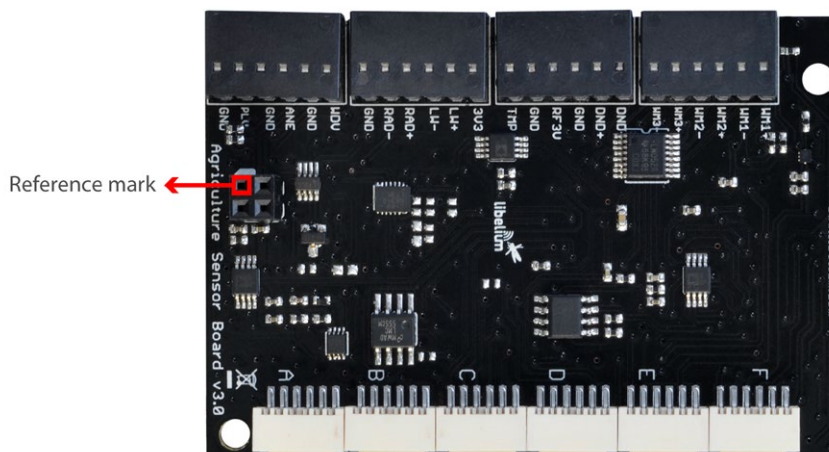


Figure : Image of the socket for the luxes sensor

In the image above we can see highlighted the four pins of the terminal block where the sensor must be connected to the board. The white dot luxes board, must match the mark of the Agriculture sensor board.

## 5.14. Sockets for casing

In case the Agriculture v3.0 board is going to be used in an application that requires the use of a casing, such as an outdoors application, a series of sockets to facilitate the connection of the sensors through a probe has been disposed.

These sockets (PTSM from Phoenix Contact) allow to assemble the wires of the probe simply by pressing them into it. To remove the wire press the slot above the input pin and pull off the wire softly.

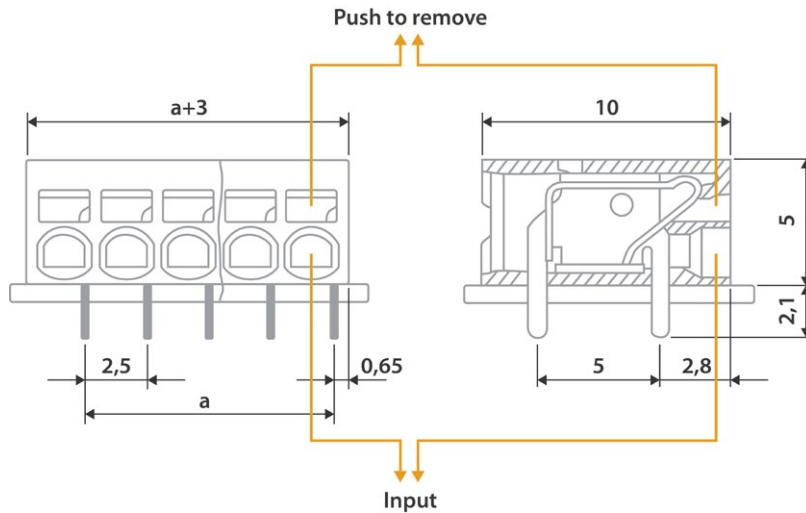


Figure : Diagram of a socket extracted from the Phoenix Contact data sheet

In the figure below an image of the board with the sockets in it and the correspondence between its inputs and the sensor's pins is shown.

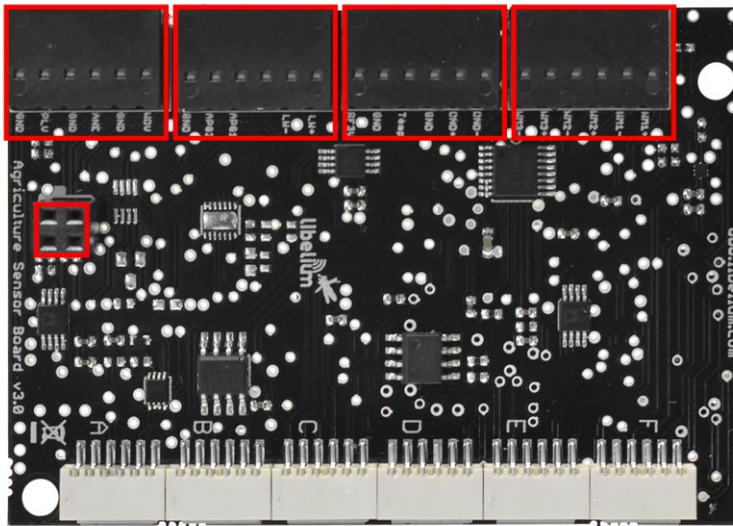


Figure : Image of the sockets for casing applications



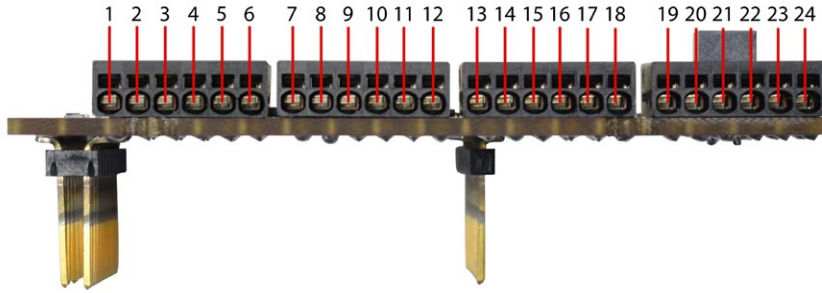


Figure : Image of the pin correspondence between the sockets and the sensors

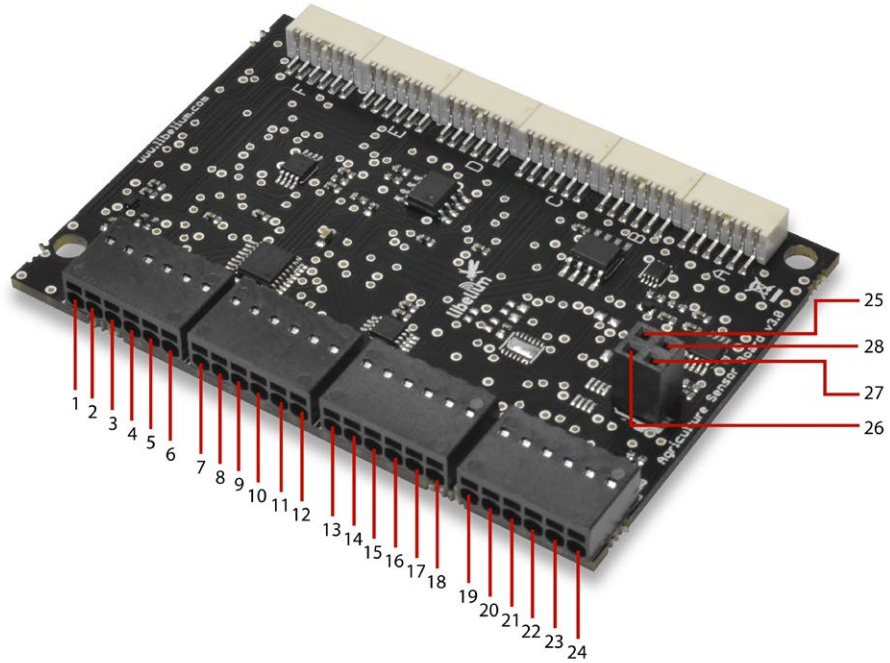


Figure : Image of the pin correspondence between the sockets and the sensors

Sensor	Pin	Function
Watermark	1	Watermark 1 +
	2	Watermark 1 -
	3	Watermark 2 +
	4	Watermark 2 -
	5	Watermark 3 +
	6	Watermark 3 -
Dendrometer	7	Dendrometer -
	8	Dendrometer +
	9	GND
	10	REF3V
Pt-1000/DS18B20	11	GND
	12	Pt-1000 / DS18B20
	13	+3V3
Leaf Wetness	14	Leaf Wetness +
	15	Leaf Wetness -
Solar Radiation	16	Radiation +
	17	Radiation -
	18	GND
Weather Station WS-3000 (Vane)	19	Vane Output (Yellow)
	20	GND (Black)
Weather Station WS-3000 ( Anemometer )	21	Anemometer Output (Green)
	22	GND (Red)
Weather Station WS-3000 (Pluviometer)	23	Pluviometer
	24	Pluviometer
Digital Bus	25	Digital Bus (SCL)
	26	GND
	27	Digital Bus (SDA)
	28	VCC



## 6. Board configuration and programming

### 6.1. Hardware configuration

The Waspote Agriculture v3.0 Board hardly requires a manual hardware configuration, since all the power control and sensor reading operations can be carried out digitally. It will be only necessary to ensure that the sensors are connected in the right way to their sockets for a proper measurement.

### 6.2. API

The Agriculture Sensor Board v3.0 for Waspote has its own library which contains the set of necessary instructions to easily configure and read each one of the sensors which can be connected to the board. Next, each one of the functions is described and the process of configuration is detailed for each sensor. The specific configuration which must be applied to each one of the sensors is explained in the specific sensor's section.

When using the Agriculture Sensor Board v3 on Waspote, remember it is mandatory to include the `WaspSensorAgr_v30` library by introducing the next line at the beginning of the code:

```
#include <WaspSensorAgr_v30.h>
```

Each agriculture sensor needs its own object. So, between the include of the library and the global variables declaration, the object must be created following the next structure:

- Name of the class: `leafWetnessClass`.
- Name of the object: We recommend to use the name of the sensor. In this case `lwSensor`.
- Only in watermark sensor you need put in brackets the socket. For example: `SOCKET_B` or `SOCKET_1`.

#### 6.2.1. Agriculture Class

Methods:

- Turn on the sensor board by enable the 3.3 V and 5 V.  
`Agriculture.ON();`
- Turn off the sensor board by disable the 3.3 V and 5 V.  
`Agriculture.OFF();`
- Read distance in cm when the ultrasound sensor is connected.  
`Agriculture.getDistance();`
- Read temperature in °C when the BME280 is connected  
`Agriculture.getTemperature();`
- Read pressure in PA when the BME280 is connected  
`Agriculture.getPressure();`
- Read humidity in % when the BME280 is connected  
`Agriculture.getHumidity();`
- Read luxes in lux when the TSL2561 is connected, in this case its necessary include the place of sensor (INDOOR or OUTDOOR) into the brackets.  
`Agriculture.getLuxes(INDOOR);`

The function `Agriculture.sleepAgr()` is an adaptation of the function `deepSleep()` in the library **WaspPWR.cpp** that allows to put WaspMote to sleep, turning the power of the board completely off or keeping the pluviometer circuits on if the interruptions of this sensor are going to be used to wake up the microcontroller. The parameters `TIME`, `OFFSET`, `MODE` and `OPTION` allow to define the time the mote will be in deep sleep mode before waking up with an RTC interruption and the modules that will be inactive during this time, like in the original function (read the WaspMote technical guide and Programming Guide for more information). To activate the pluviometer interruptions, the parameter `AGR_INTERRUPTION` must be assigned with the value `SENS_AGR_PLUVIOMETER` (remember not to deactivate the sensor board when defining the parameter `OPTION` for a correct operation of the interruptions).

A basic program to detect events from the board will present a similar structure to the following, subject to changes in dependence of the application:

- 1. The board is switched on using the function `Agriculture.ON`
- 2. When the mote wakes up, disable interruptions from the board using function `Agriculture.detachPluvioInt()`.
- 3. Store or send via a radio module the gathered information.
- 4. Return to step 4 to enable interruptions and put the mote to sleep.

### 6.2.2. dendrometerClass

Method:

- Read the dendrometer value in Agriculture Pro.  
`dnd.readDendrometer();`

### 6.2.3. ds18b20Class

Method:

- Read the soil temperature in Agriculture Normal.  
`ds18b20.readDS18b20();`

### 6.2.4. leafWetnessClass

Method:

- Read the leaf wetness % in Agriculture Normal and Pro.  
`lw.getLeafWetness();`

### 6.2.5. pt1000Class

Method:

- Read the soil temperature in Agriculture Pro.  
`soilTemp.readPT1000();`

### 6.2.6. radiationClass

Method:

- Read the solar Radiation in Agriculture Pro.  
`solarRadiation.readRadiation();`

### 6.2.7. watermarkClass

- In this case it is necessary to specify the socket where watermark is connected and also instance object.

```
watermarkClass wmSensor1(SOCKET_1);
```

Method:

- Read the soil water content (Hz) in Agriculture Normal and Pro.

```
wmSensor1.readWatermark();
```

### 6.2.8. weatherStationClass

Method:

- Read the wind speed in Agriculture Normal and Pro.

```
wh.readAnemometer();
```

- Read the vane direction in Agriculture Normal and Pro.

```
wh.readVaneDirection();
```

The `attachPluvioInt()` function enables the interruptions generated by the pluviometer. Take into account that this sensor is permanently powered all the time if the board is on, so it will keep on triggering interruptions as long as they are enabled.

```
wh.AttachPluvioInt();
```

Complementing the previous function, the aim of `detachPluvioInt()` is to deactivate the interrupts generated by the pluviometer. After its execution, the microcontroller will ignore any interruption which arrives from this sensor until the `attachPluvioInt()` instruction is called again.

```
wh.detachPluvioInt();
```

- Read pluviometer value.

```
wh.readPluviometerCurrent();
```

- Read pluviometer value in 1 hour.

```
wh.readPluviometerHour();
```

- Read pluviometer value in 1 day.

```
wh.readPluviometerDay();
```

## 7. API changelog

Keep track of the software changes on this link:

[www.libelium.com/development/waspmote/documentation/changelog/#Agriculture](http://www.libelium.com/development/waspmote/documentation/changelog/#Agriculture)

## 8. Documentation changelog

### From v7.1 to v7.2:

- Changes in the connection figures
- Changes in the API software description

### From v7.0 to v7.1:

- Added references to the integration of Industrial Protocols for Plug & Sense!

## 9. Consumption

### 9.1. Tables of consumption

In the following table the consumption of the board is shown, the constant minimum consumption (fixed by the permanently active components), the minimum consumption of the electronics included in each group formed by the switches (without sensors) and the individual consumptions of each of the sensors connected alone to the board (the total consumption of the board with a determined sensor will be calculated as the sum of the constant minimum consumption of the board, plus the minimum consumption of the group to whom the sensor belongs, plus the consumption of the sensor).

Remember that the board's power can be completely disconnected, reducing the consumption to zero, using the 3.3 V and the 5 V main switches disconnection command included in the library.

	Consumption
<b>Minimum (constant)</b>	0 $\mu$ A
<b>Weather station WS-3000 group</b>	0 $\mu$ A
<b>Watermark sensors group</b>	1.4 mA
<b>Low consumption group</b>	1.8 mA
<b>Watermark (1 sensor)</b>	<0.8 mA
<b>Watermark (2 sensors)</b>	<1.5 mA
<b>Watermark (3 sensors)</b>	<2.2 mA
<b>Anemometer</b>	<400 $\mu$ A
<b>Wind vane</b>	<300 $\mu$ A
<b>Pluviometer</b>	0 $\mu$ A (330 $\mu$ A in on pulse -10 ms approximately-)
<b>BME280</b>	<2.8 $\mu$ A
<b>DS18B20</b>	<1.5 mA
<b>Ultrasound sensor</b>	2.1 mA
<b>TSL2561</b>	0.24 mA
<b>Solar radiation, SQ-110</b>	0 $\mu$ A
<b>Pt-1000</b>	1.5 mA
<b>Dendrometers</b>	160 $\mu$ A
<b>Leaf wetness</b>	<240 $\mu$ A

### 9.2. Low consumption mode

The Waspote Agriculture v3.0 Board has been designed to have the least consumption possible. For this, the only recommendations which the user must try to follow are the following:

- **Use the Waspote low consumption mode**

This board's library includes a command to put the mote into a low consumption mode. Use it during the time in which the mote is not carrying out any measurement and try to space them as much as your application allows you.

- **Do not connect sensors that are not going to be used**

Since several sensors share the same power line, a sensor that is not going to be used connected to the board will entail an additional consumption, and so a shorter life of the battery.

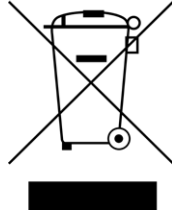
## 10. Maintenance

- In this section, the term “WaspMote” encompasses both the WaspMote device itself as well as its modules and sensor boards.
- Take care with the handling of WaspMote, do not drop it, bang it or move it sharply.
- Avoid putting the devices in areas of high temperatures since the electronic components may be damaged.
- The antennas are lightly threaded to the connector; do not force them as this could damage the connectors.
- Do not use any type of paint for the device, which may damage the functioning of the connections and closure mechanisms.



## 11. Disposal and recycling

- In this section, the term “Waspote” encompasses both the Waspote device itself as well as its modules and sensor boards.
- When Waspote reaches the end of its useful life, it must be taken to a recycling point for electronic equipment.
- The equipment has to be disposed on a selective waste collection system, different to that of urban solid waste. Please, dispose it properly.
- Your distributor will inform you about the most appropriate and environmentally friendly waste process for the used product and its packaging.



## 12. Certifications

Libelium offers 2 types of IoT sensor platforms, Wasp mote OEM and Plug & Sense!:

- **Wasp mote OEM** is intended to be used for research purposes or as part of a major product so it needs final certification on the client side. More info at: [www.libelium.com/products/wasp mote](http://www.libelium.com/products/wasp mote)
- **Plug & Sense!** is the line ready to be used out-of-the-box. It includes market certifications. See below the specific list of regulations passed. More info at: [www.libelium.com/products/plug-sense](http://www.libelium.com/products/plug-sense)

Besides, Meshlium, our multiprotocol router for the IoT, is also certified with the certifications below. Get more info at:

[www.libelium.com/products/meshlium](http://www.libelium.com/products/meshlium)

List of certifications for Plug & Sense! and Meshlium:

- CE (Europe)
- FCC (US)
- IC (Canada)
- ANATEL (Brazil)
- RCM (Australia)
- PTCRB (cellular certification for the US)
- AT&T (cellular certification for the US)



Figure : Certifications of the Plug & Sense! product line

You can find all the certification documents at:

[www.libelium.com/certifications](http://www.libelium.com/certifications)

## Appendix 1: Watermark sensor's interpretation reference

The next table shows the resistance value of the sensor at different soil water tension conditions (at a 75 Fahrenheit degree, equivalent to 23.8 Celsius degree):

Soil Water Tension (cbar)	Sensor Resistance (Ohms)
0	550
9	1000
10	1100
15	2000
35	6000
55	9200
75	12200
100	15575
200	28075

Figure : Table 1. Resistance values of the sensor in function of the soil water tension

This series may be approached by this equation:

$$\text{Equation 1: } R_s = 137.5 * TA + 550$$

Where TA is the soil water tension expressed in centibars.

From the real values and the approximation we obtain the next graph of the sensor resistance versus the soil water tension.

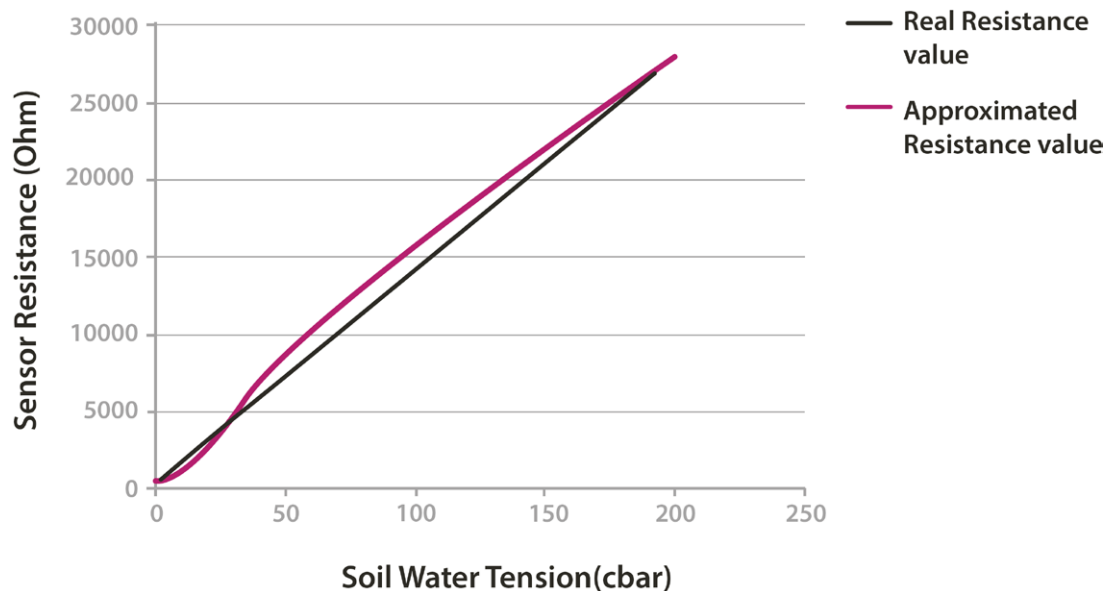


Figure : Resistance of the sensor in function of the soil water tension

In the next figure we can see the frequency of the output of the adaptation circuit for the sensor, for the real resistance values and for the linearly approximated resistance values.

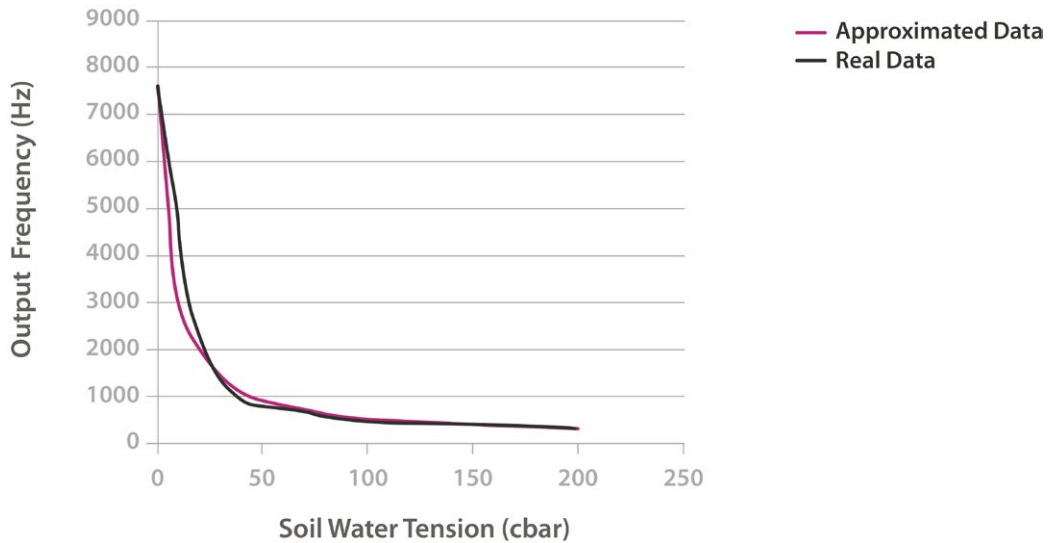


Figure : Output frequency of the adaptation circuit in function of the soil water tension

The formula used to draw this graph, from the sensor resistance, is shown bellow:

$$\text{Equation 2: } F = \frac{(R_s + 150390)}{(0.021 * R_s + 8.19)}$$

Where F is the output frequency in Hz and  $R_s$  the sensor resistance in ohms.

If we substitute Equation 1 in Equation 2, we get the output frequency in function of the soil water tension:

$$\text{Equation 3: } F = \frac{(137.5 * TA + 550 + 150390)}{(0.021 * (137.5 * TA + 550) + 8.19)}$$

$$\text{Equation 4: } F = \frac{(137.5 * TA + 150940)}{(2.8875 * TA + 19.74)}$$

We can see that the frequency output for the working range is between 300 Hz (corresponding to the 200 cbar of maximum soil water tension) and 7600 Hz approximately for 0cbar measurement. It has been empirically checked that for very wet soils, bellow 10 cbar, the behavior of different sensors is very variable, so calibration is highly recommended if accuracy under these conditions is needed.

To obtain the response of the sensor beyond this range, over the 200 cbar, we must extrapolate those soil water tension values from the linear approximation obtained in equation 1. These sensors are not prepared for working under those conditions, so these graph must be only taken as a reference.

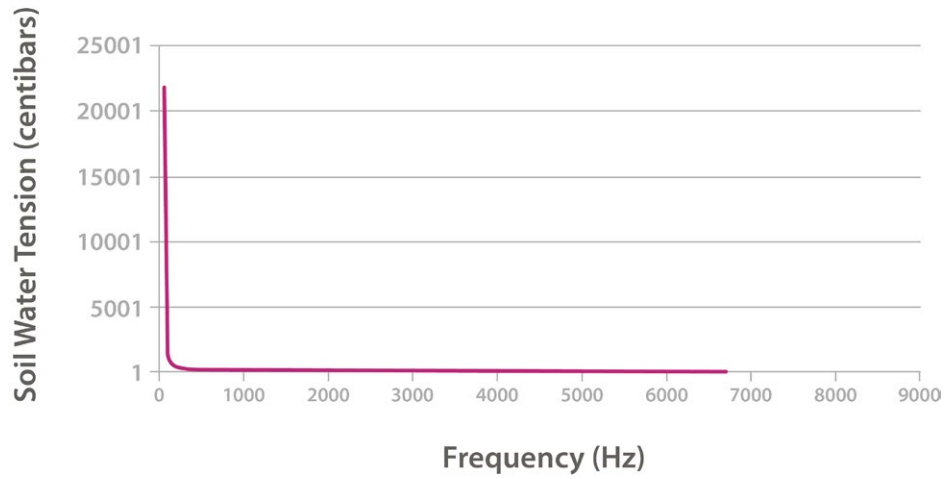


Figure : Soil water tension in function of the output frequency of the circuit

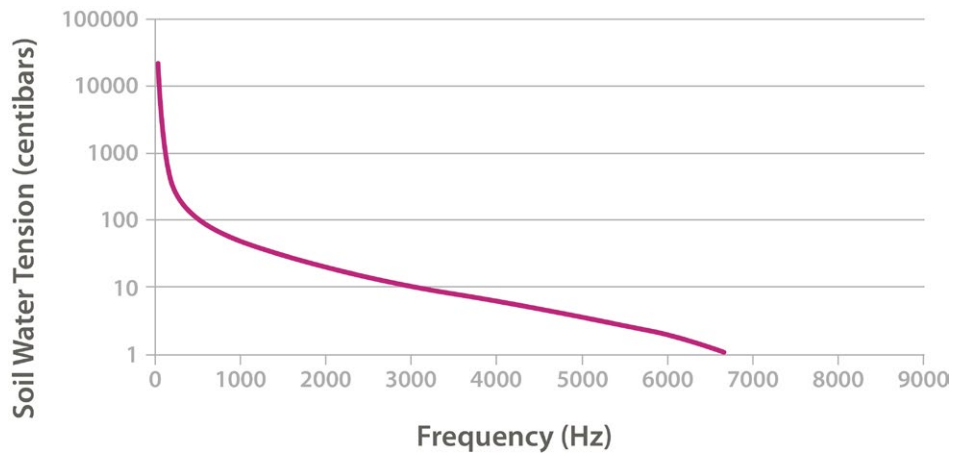


Figure : Soil water tension (in logarithmic scale) in function of the output frequency of the circuit