CARLO GAVAZZI
Automation Components

CARLO GAVAZZI

UWP 3.0 Tool
Car Park Manual

Rev. 8.4.14, 11/02/2022

Download the updated version: www.gavazziautomation.com

Summary

1 Introduction

The UWP 3.0 Tool has been developed for the configuration of the UWP 3.0, a programmable Linux embedded PC specially designed for car park and building automation applications.
The UWP 3.0 Tool is also used to write the configuration into the SBP2CPY24 servers.
All functions are represented by graphic symbols, and all function related parameters are set up locally in the PC, and then transferred to the UWP 3.0 via Ethernet. Some of the function parameters can be changed later via remote connection to the controller (web server, email, SMS, Modbus...) as described later in this manual.
Likewise, data from the UWP 3.0 can be uploaded and modified.

The PC does not need to be connected to a UWP 3.0 controller in order to make a configuration.

The figures in this manual may differ from the figures on your screen. This is not necessarily an error but may be caused by revision differences.

The contents of this manual may be altered without notice.

1.1 Requirements

1.1.1 Minimum hardware requirements

- A Microsoft® Windows®-based PC
- Display with a resolution capability of minimum 1024x768 pixels
- 1 GB of disk-space
- An Ethernet-port and cable or SD-card reader or USB port 2.0 or higher

1.1.2 Software requirements

- Microsoft® Windows® 10/8.1/8/7/Vista/ (32 or 64-bit)
- Microsoft Dot-Net Framework 4.5

1.1.3 How to read the software version number

The UWP 3.0 tool revision number has the following structure:

Major	Minor	Sub minor	Revision

- **Major:** this identifies the main features of the software. It is incremented when new features are added or there are big changes in the existing ones.

- **Minor:** this identifies the version of the relevant *major* release and it is incremented when there are small new functions and bug fixing.

- **Sub minor:** this identifies the version of the relevant *major* release and it is incremented when there are bugs fixing

- **Revision:** this identifies the status of the release.
  1 *beta*: for internal use only
  2 *controlled beta:* to be shared with selected customers for field testing
  3 *final*: available for everybody on the Carlo Gavazzi website

2 Installation

The UWP 3.0 Tool can be downloaded from Carlo Gavazzi's Product Selection website. To install the software, you just have to double click on the *Setup.exe* file and follow the instructions on the screen.

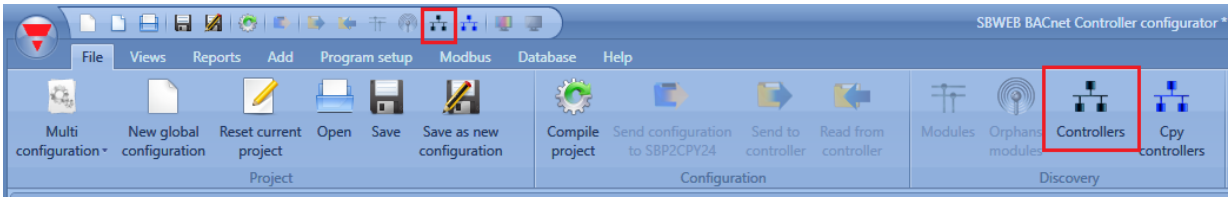
In order to get in contact with the UWP 3.0 controller, the user has several connection modes available:

- via Ethernet connection
- via Mini USB cable
- via Modem connection
- remotely using MAIA Cloud (VPN)
  *MAIA Cloud is compatible with SBP2CPY24 version **2.6.3 onwards**.*

After a power on, the UWP 3.0 master unit is ready to work after about 1 minute. Only when the yellow BUS Led starts flashing is the master unit ready.

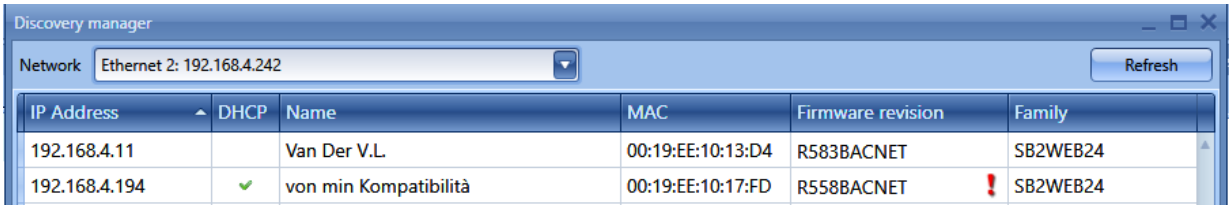
2.1 How to connect the master unit UWP 3.0 with Ethernet connection

To connect to a master unit UWP 3.0, the user has to click on the icon highlighted in red in the picture below: the UWP 3.0 Tool will start the discovery of the UWP 3.0 connected to the Ethernet network.



Important note
If the PC is running the Windows Firewall or a Third party Firewall / Antivirus, make sure that the ports 52325, 10000, 10001, 10002, 80 and 443 are not blocked (input/output packets). These ports are used by the UWP 3.0 Tool to search for the master unit in the network and for communication. If a firewall blocks these ports, the UWP 3.0 Tool will not be able to find the controllers in the network or to use the Live Signals.
Be sure that the master unit's IP settings match the IP settings of the PC used: it must have the same IP class and the same net mask address.

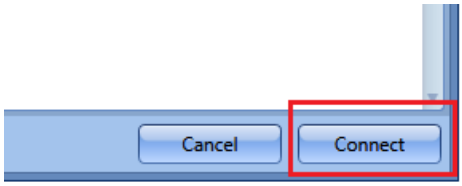
When the PC has more than one network card or has many IP addresses, it is possible to select the right network from the list (see picture below): it must be the same one as the UWP 3.0 is connected to.



When the icon marked in red is clicked, the UWP 3.0 Tool starts looking for the controller(s). If one or more units are found, a window will pop-up, allowing you to make a selection of which controller to connect to.

Select the master unit on the list or *Cancel* if you do not want to get connected at this time.

After the user has pressed *Connect*, the red Led on the selected UWP 3.0 starts flashing and the connection will be established.



2.2 How to connect with a mini-USB cable

The UWP 3.0 controller can be connected to the PC by means of the *mini-B* port available on the front of the housing, protected by the front cover. Please refer to the *Hardware manual* for technical information.

The following items are required to obtain the connection:

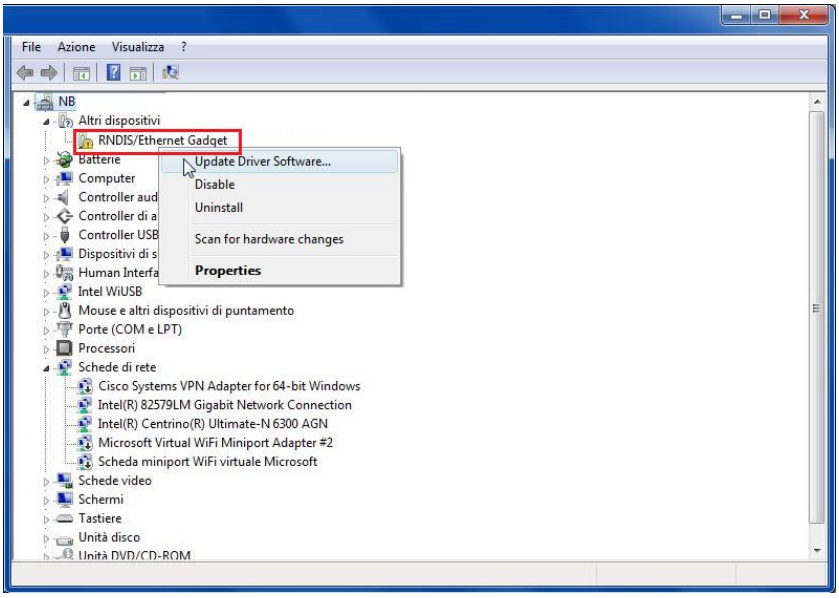
- A **USB 2.0 cable-Type-A / Mini-B** (this is not provided with the UWP 3.0 controller).
- *Mini-USB driver.zip* package available on www.productselection.net website in the UWP 3.0 page.

The driver installation procedure can change slightly depending on the operating system in the user's PC: please follow the instructions below.

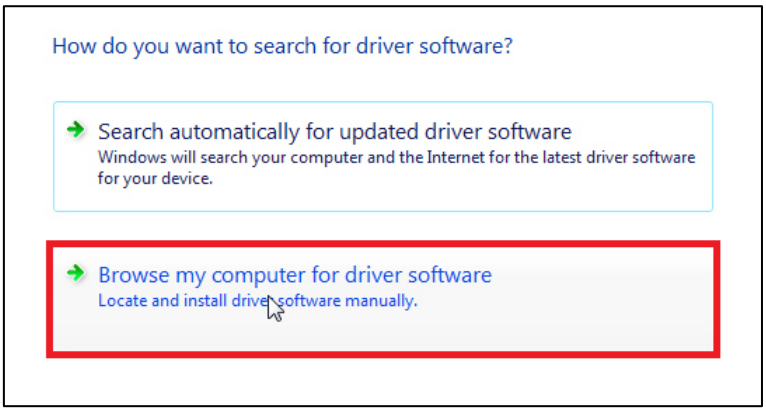
2.2.1 How to install the mini-USB driver for Windows 7 / Vista / XP

Open the archive "*mini-USB driver.zip*" and save the file *linux.inf* in your PC, then install the driver by following the instructions below:

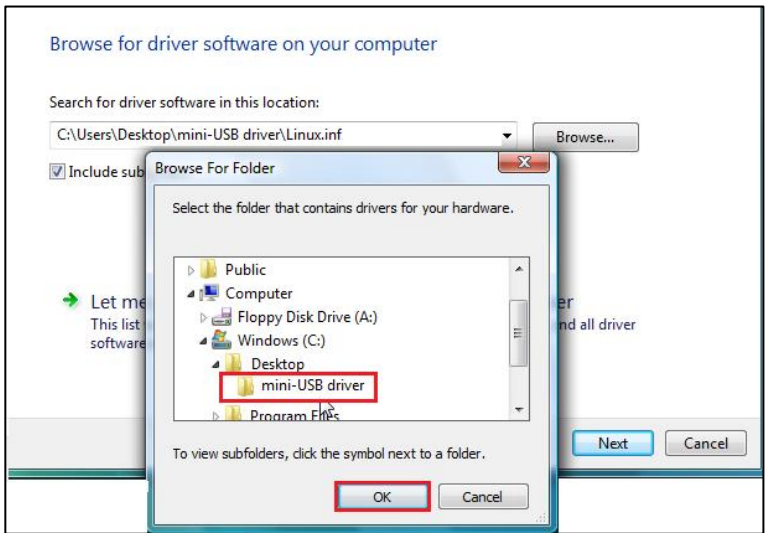
1. Plug the USB connector into a free USB port of the PC and the mini-USB connector into the mini-B port of the UWP 3.0
2. Go to *Control Panel* and open *Device Manager*.
3. Find the device *RNDIS/Ethernet Gadget*, right-click on it and select *Update Driver Software…,* as shown in the picture below.



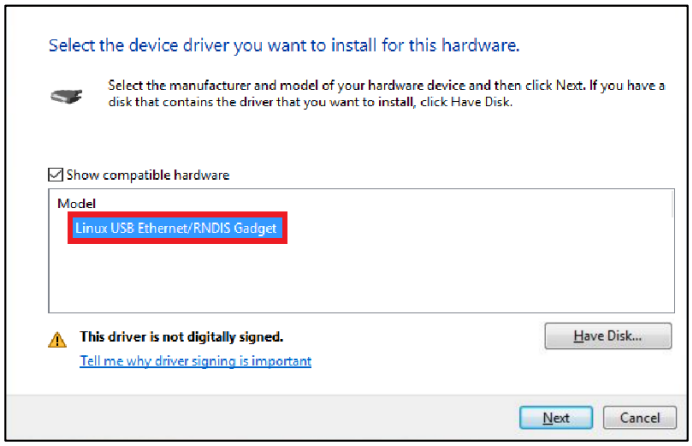
4. Select the *Browse my computer for driver software* option.



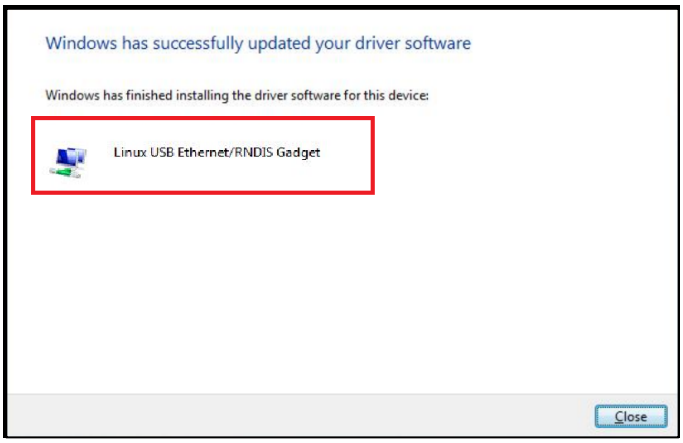
5. Browse for the *linux.inf* driver file and press *OK.*



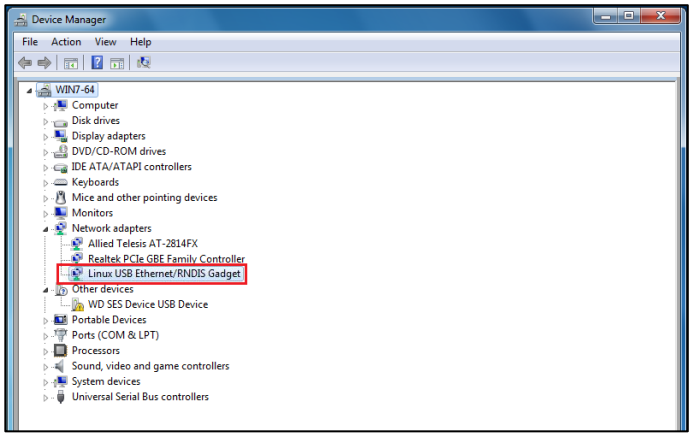
6. The driver *Linux USB Ethernet/RNDIS Gadget* will be detected: press *Next* to proceed.



7. The driver will be installed, as shown in the picture below.

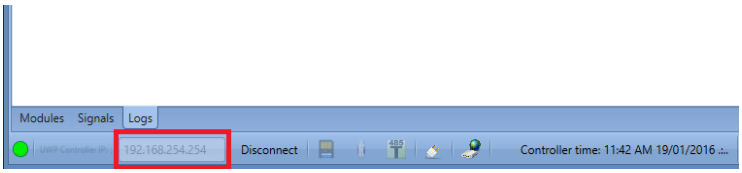


8. When the driver is installed, in the *Network adapter* category a virtual network board named *Linux USB Ethernet/RNDIS Gadget* will be added, as shown in the picture below. The driver automatically gives a dynamic IP address to the Controller/PC according to the actual IP of the PC.



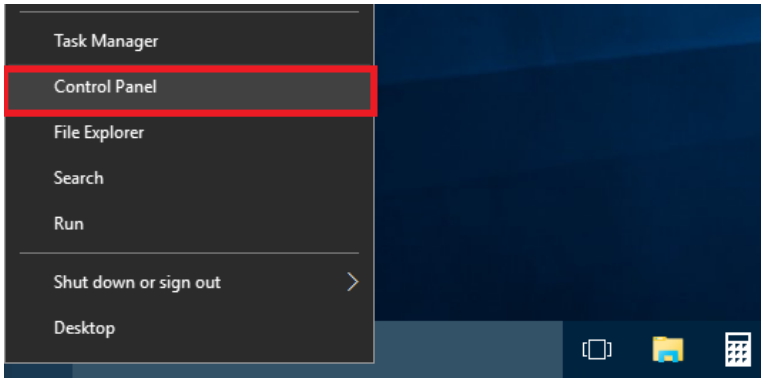
For example, if the PC has the IP address 192.168.0.10, the virtual board will be created with a new address 192.168.254.xxx and the IP address 192.168.254.254 will be assigned to the UWP 3.0 controller.

9. Insert the IP address 192.168.254.254 in the UWP 3.0 Tool and press *Connect,* as shown in the picture below:

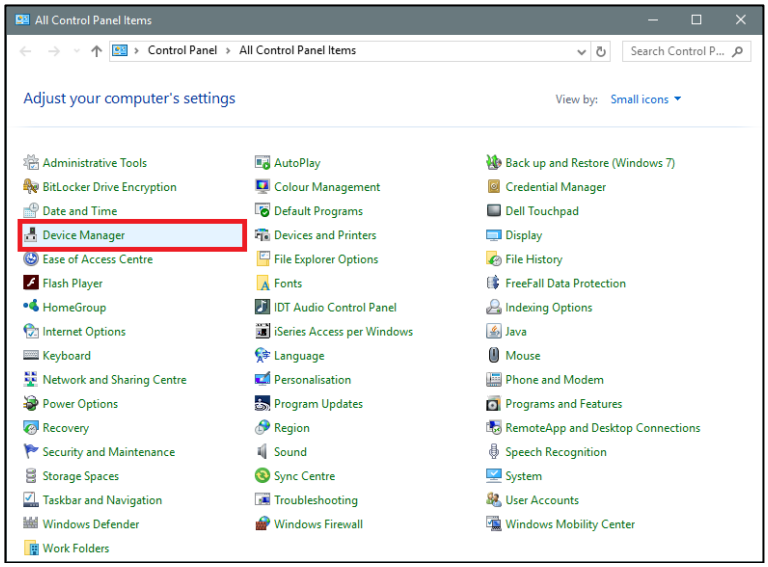


2.2.2 How to install the mini-USB driver for Windows 10 / 8.1 / 8

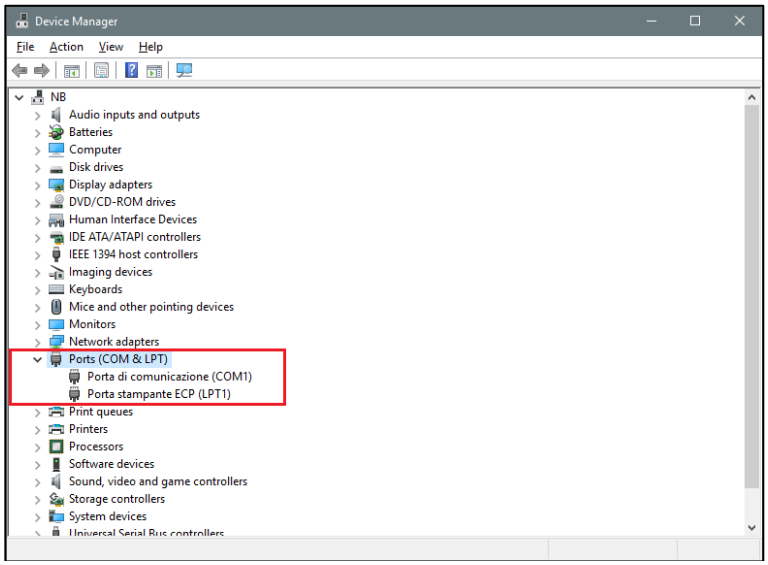
1. Open *Windows 10 Control Panel* by right-clicking on the *Start* button and click on *Control Panel*.



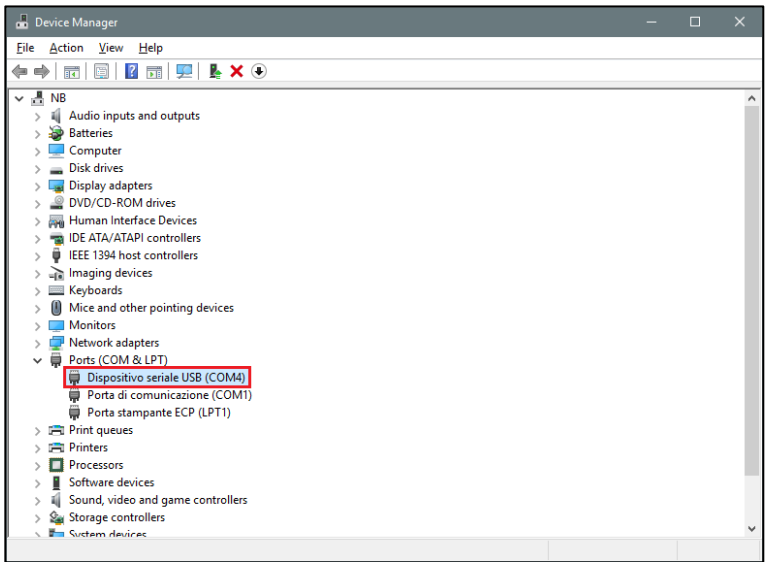
2. In *View by: Small icons*, click on *Device Manager.*



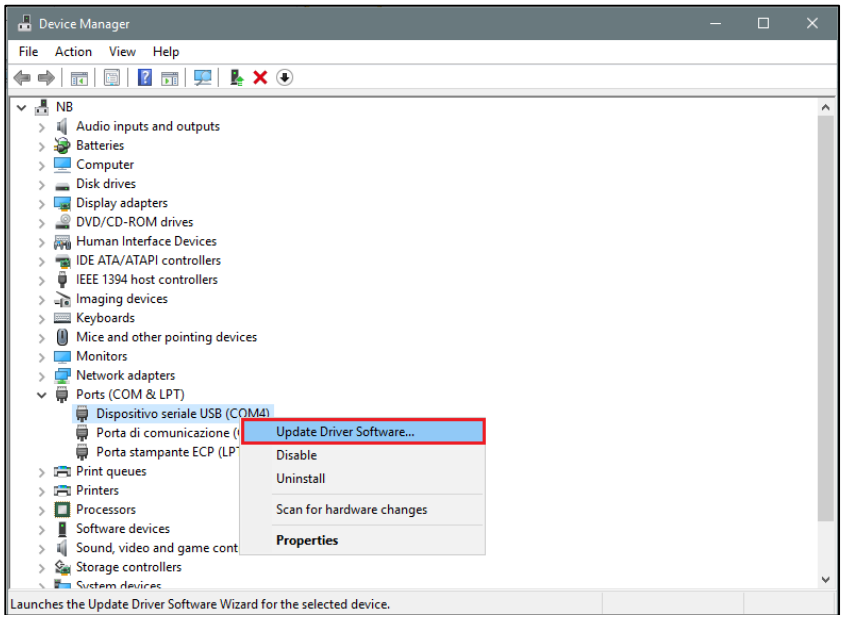
3. Before connecting the mini-USB cable to the PC and to the UWP 3.0 controller, in the list of hardware categories double-click on the category Ports (COM & LPT) and take note of the serial communication (COM) ports in use. In the example shown below, the only serial communication port is ***COM1***.



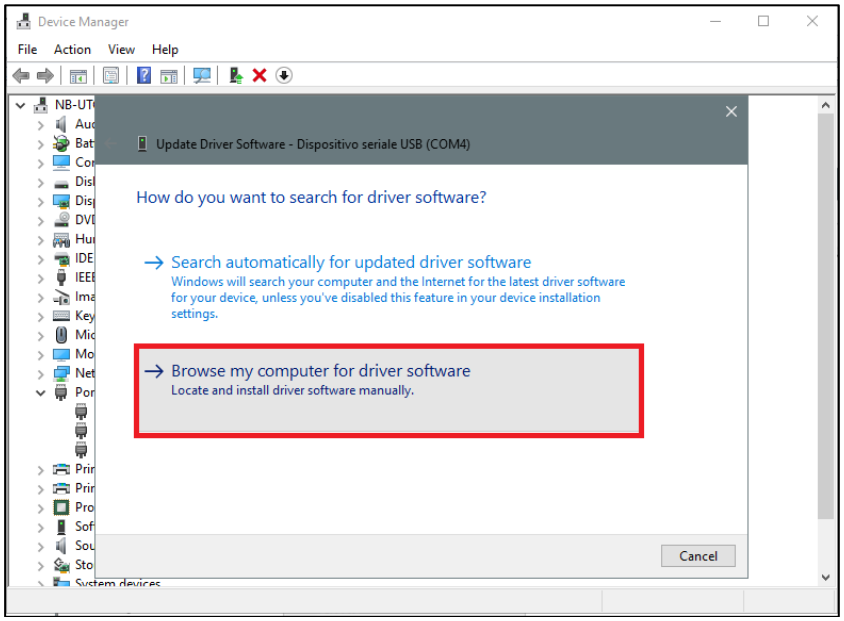
4. Plug the mini-USB cable into the PC and into the UWP 3.0 controller. A new serial communication port (COM*x*) will automatically be added after a few seconds. In the example shown below the new port added is ***COM4.***



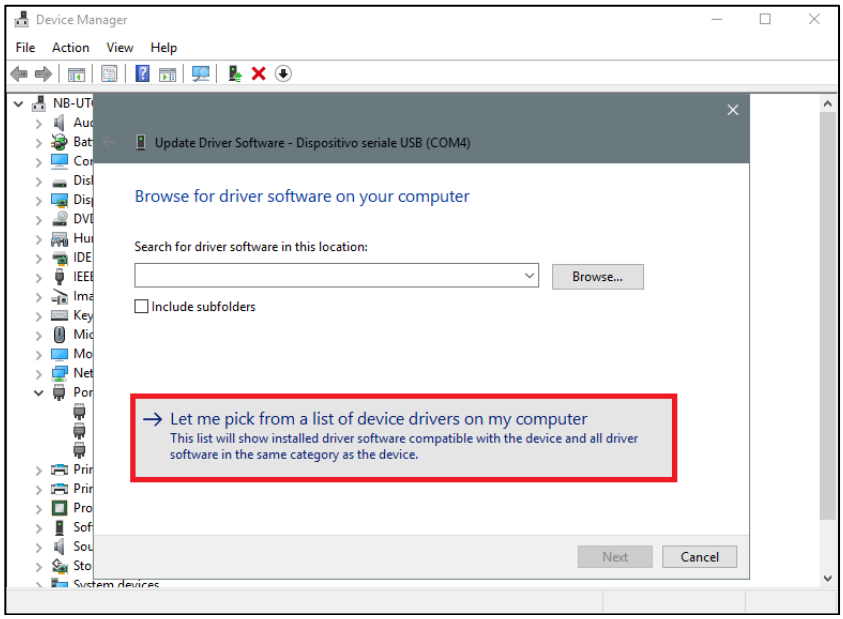
5. Right-click on the device and select *Update Driver Software*...



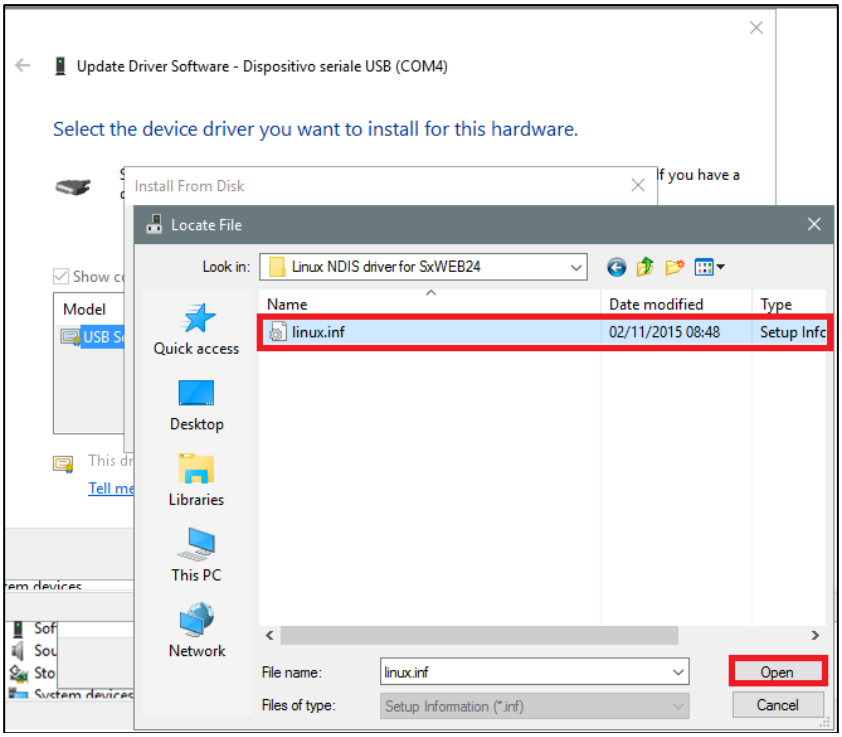
6. In the pop-up window, select the second option, *Browse my computer for driver software*



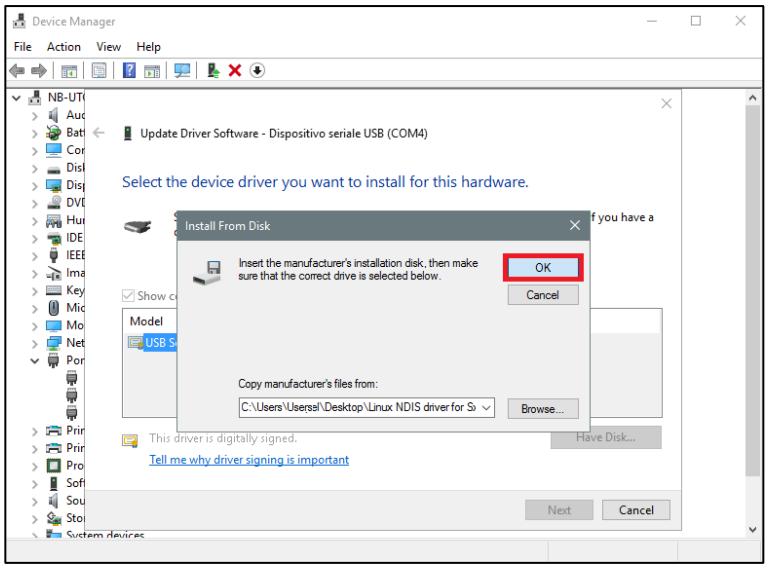
7. In the next page, select the option *Let me pick from a list of device drivers on my computer*



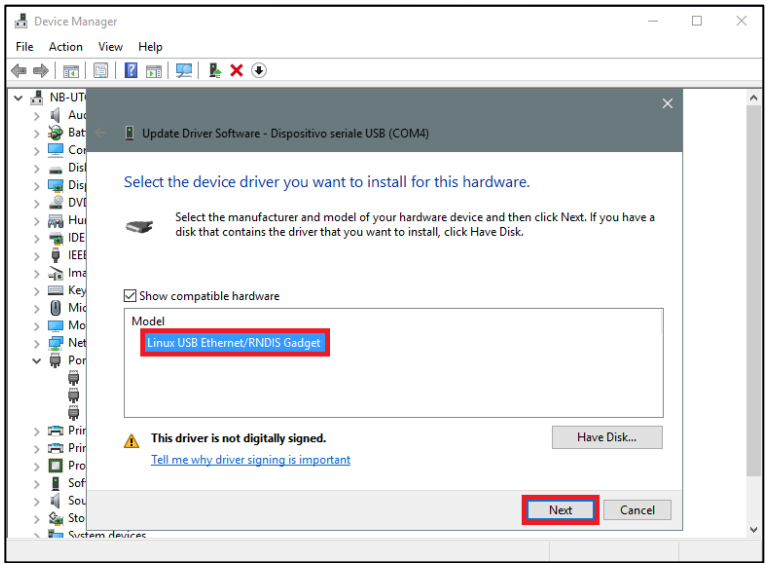
8. Browse for the *linux.inf* file, save it in the PC and press *Open*



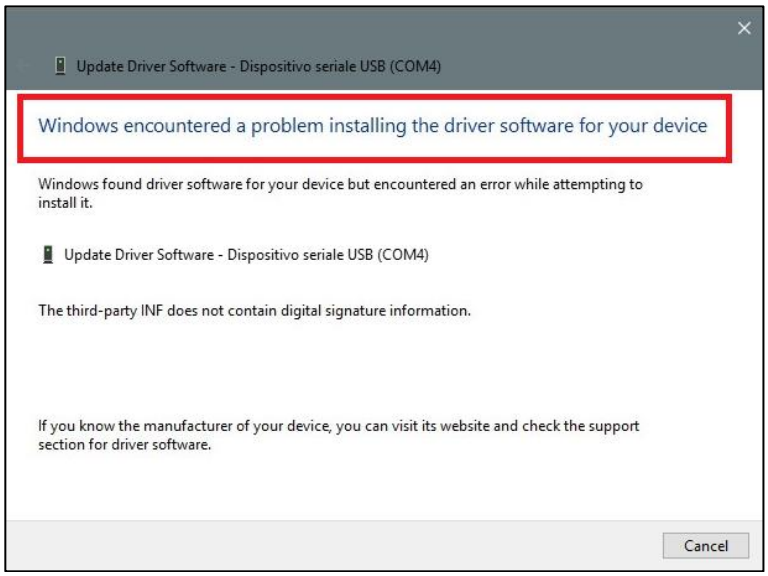
9. Press *OK* in the next window, as shown in the picture below



10. The *Linux USB Ethernet/RNDIS Gadget* is ready to be installed. Click on *Next* to continue the installation. See the



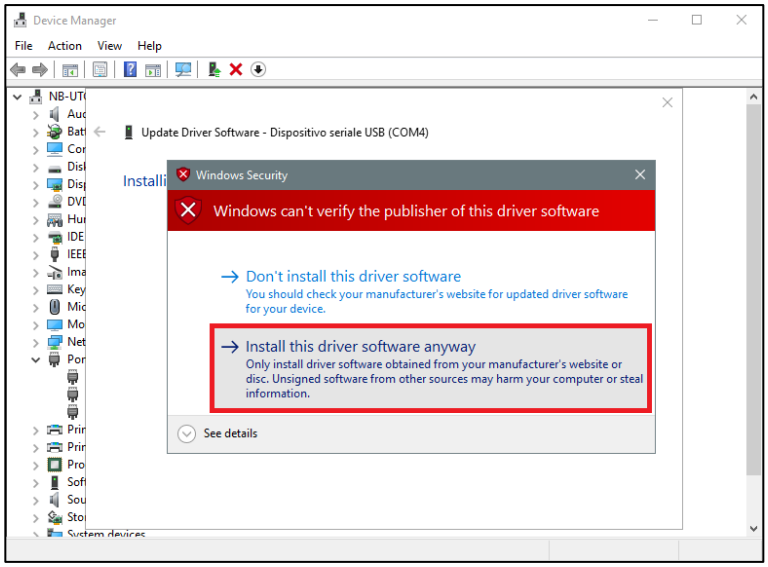
Follow the instructions below to disable driver signature enforcement **ONLY** if the user gets the error shown below, otherwise go to step 11.



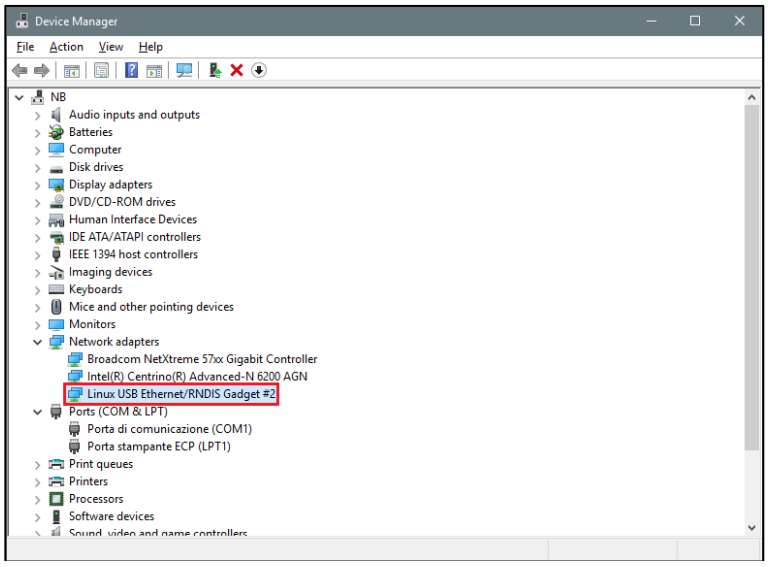
Windows 10 enforces driver signatures by default. This can be disabled to install drivers that are not digitally signed. Please refer to the following steps to disable driver signature enforcement.

1. Click the **Start** **menu** and select **Settings**.
2. Click **Update and Security**.
3. Click on **Recovery**.
4. Click **Restart now** under **Advanced Startup**.
5. Click **Troubleshoot**.
6. Click **Advanced options**.
7. Click **Startup Settings**.
8. Click on **Restart**.
9. On the Startup Settings screen press 7 or F7 to disable driver signature enforcement.
10. The computer will restart and the user will be able to install non-digitally signed drivers.

11. **N.B. Windows 10 does not detect the Digital signature, so to install the driver correctly, click on *Install this driver software anyway.***



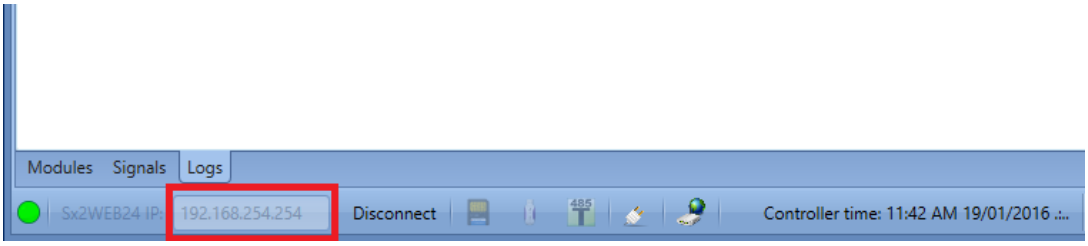
12. When the driver has been installed, in the *Network adapter* category a virtual network board named *Linux USB Ethernet/RNDIS Gadget #2 will* be added, as shown in the picture below.



13. The driver automatically gives a dynamic IP to the Controller/PC according to the actual IP of the PC.

For example, if the PC has the IP 192.168.0.10, the virtual board will be created with a new address 192.168.254.xxx and the address 192.168.254.254 will be assigned to the UWP 3.0 controller.

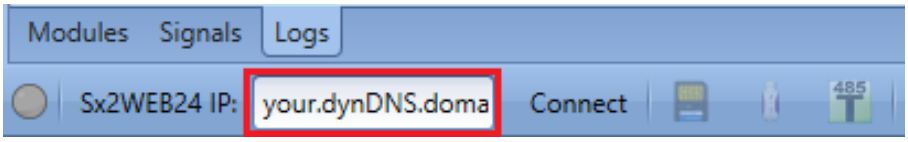
The user can type the IP address **192.168.254.254** in the address bar and click *Connect*. The green circle icon appears in the left part of the address bar when the connection is established, as shown in the picture below.



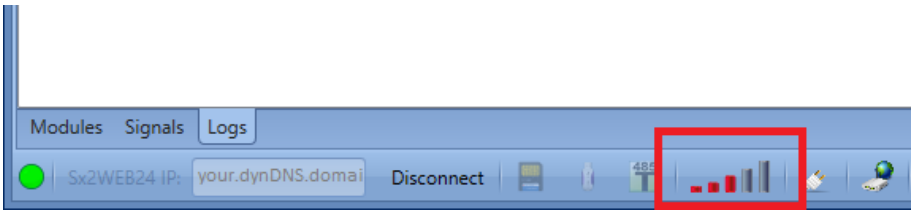
2.3 How to connect to the controller by means of a modem

When the user has to connect to the controller for the first time, the modem connection is not available, **so first it has to be configured.** Please refer to the section *How to configure the modem* in the system manual for more information (http://www.productselection.net/MANUALS/UK/uwp3.0_tool.pdf).

In the IP address bar, the user can type in the dynamic IP address delivered by the ISP, or they can directly fill in the *DynDNS domain* registered (see *How to set the DynDNS parameters when using the modem*): the second option is preferable since the UWP 3.0 Tool is able to auto-update the dynamic IP address delivered for the DynDNS account each time the modem is reset.



When a connection is established with the modem, the network signal strength will be shown by a set of bars, as shown in the picture below. The higher the number of red bars, the stronger the network signal.



2.4 How to connect to the controller remotely via MAIA Cloud (VPN)

MAIA Cloud is the Carlo Gavazzi PaaS (Platform as a Service) solution that allows to remotely connect different remote devices to a UWP 3.0 unit and/or to a SBP2CPY24 unit. Users who have access to the MAIA Cloud can easily reach the gateways and the endpoints, provided they have the necessary access rights, using a PC and a standard browser.

To connect remotely via MAIA Cloud using the UWP Tool you have to:

1) Connect your UWP 3.0 to Internet
2) Register and log in to MAIA Cloud using a PC and a standard browser (link here)
   For further information see ***How to register on MAIA Cloud***
3) Add your device to MAIA Cloud.
   In Car Park applications there are two use cases. For further information see MAIA Cloud for Car Park use cases
4) Go to the home page and set up a remote connection using the UWP Tool application
   For further information see How to set up a remote connection using MAIA Cloud

Note: MAIA Cloud is compatible with SBP2CPY24 version ***2.6.3 onwards*** *and UWP 3.0 version* ***8.4.0.3*** *onwards.*

2.4.1 How to set up a remote connection using MAIA Cloud

Follow this procedure to set up a remote connection to a UWP 3.0 activated in MAIA Cloud using the

UWP Tool:

1. Log in to your MAIA Cloud (link here)
2. Open the home page or open the **main menu** and go to ***Devices > VPN***

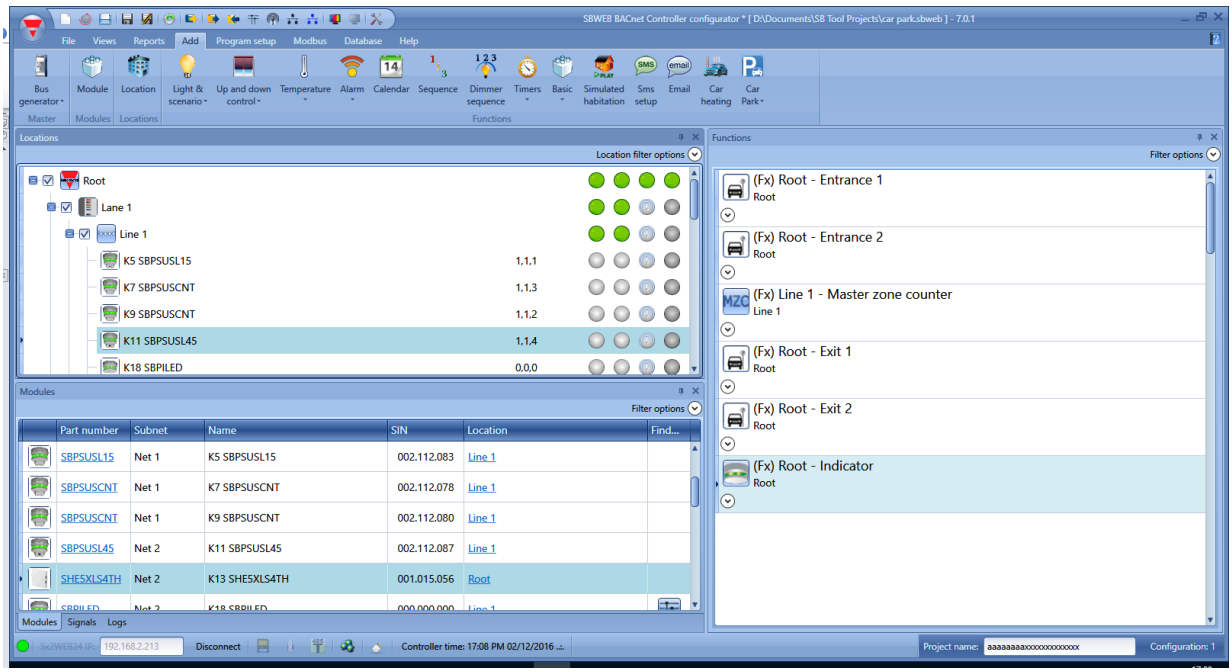
Note: you have to be logged into MAIA Cloud Connector plugin.

3.

If you want to…	**Then…**
Use the UWP Tool application to create a VPN tunnel to the UWP 3.0	You can a) Click ⌄ to open the **Connection** drop-down menu of the gateway b) Click the gateway you want to connect in the map and click **Applications** to open the **Connection** side panel After that, click the UWP 3.0 you need to connect to open the application. *Note: at the first connection you have to define the path to open the SxTOOL.exe*
open all the ports of the application composing the gateway and endpoint profile	You can a) Click ⋮ > **Connect** from the **Action** menu of the gateway b) Click the gateway you want to connect in the map and click **Connect.** After that you can launch UWP Tool and enter the UWP 3.0 virtual IP address that you find in the **Connection** drop-down menu or side panel.
Disconnect from the endpoint/gateway	You can a) Click ⋮ > **Disconnect** from the **Action** menu of the gateway b) Click ⌄ to open the **Connection** drop-down menu of the endpoint/gateway and click **Disconnect** c) Click the gateway you want to disconnect in the map and click **Disconnect**

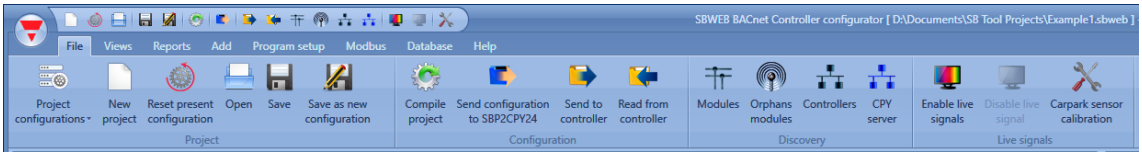
3 User interface

When the UWP 3.0 Tool starts, the following window appears:



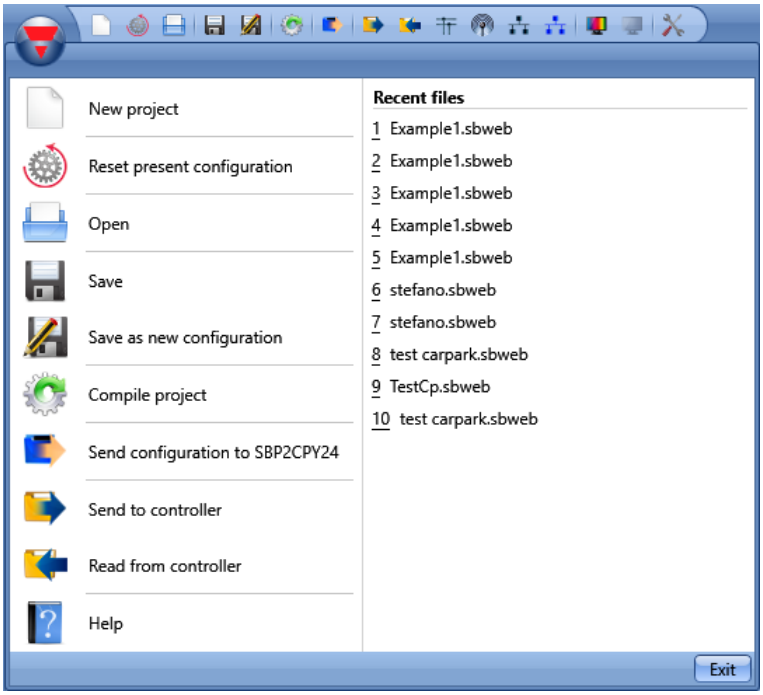
The user interface uses the standard ribbon tool often used by a lot of software nowadays. To access a ribbon, just click on the relevant menu.

3.1 File Menu



In the File menu the user can create a new project (single and multi-configuration), open an already existing one or save it as in a standard menu File.
In addition some functions strictly related to the connection with UWP 3.0/SBP2CPY24 are available and listed here:

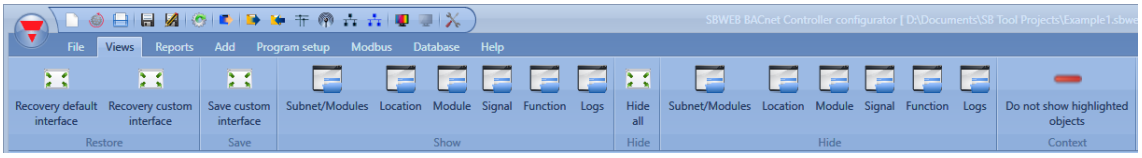
- Compiling a project
- Uploading/downloading of a project
- Discovery functions of the UWP 3.0, SBP2CPY24 and modules connected in the network
- Addressing and calibration of car park sensors
- Enabling/Disabling of Live signal monitoring



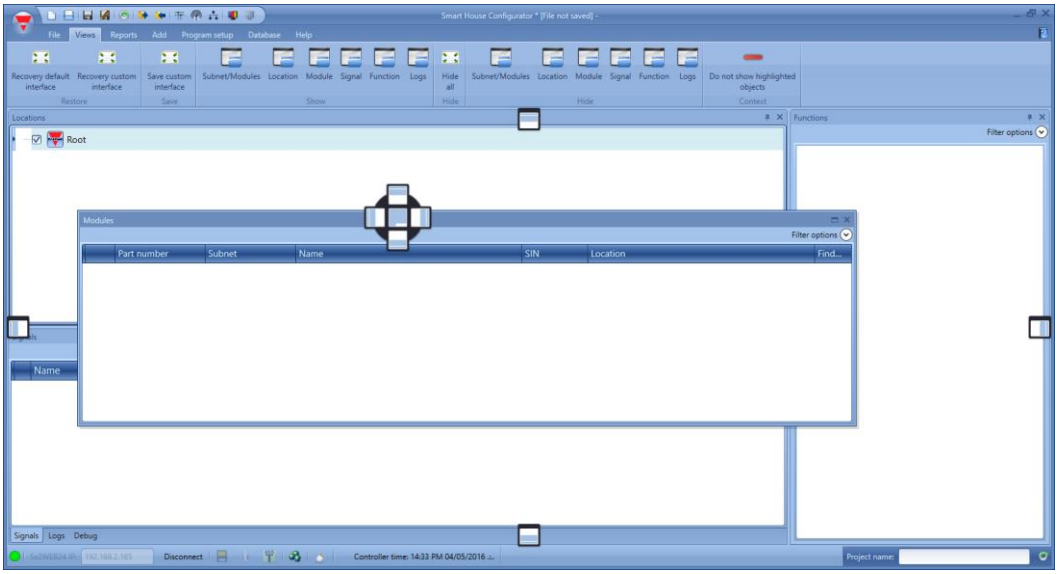
The user can access the menu file either with the quick menu in the upper part of the window on the right of the red Carlo Gavazzi triangle or by clicking on the triangle.

3.2 View Menu

In the View menu, the user can configure the preferred position of the windows relevant to Locations, Modules, Signals, Functions and Subnet. It can also remove the contents view.



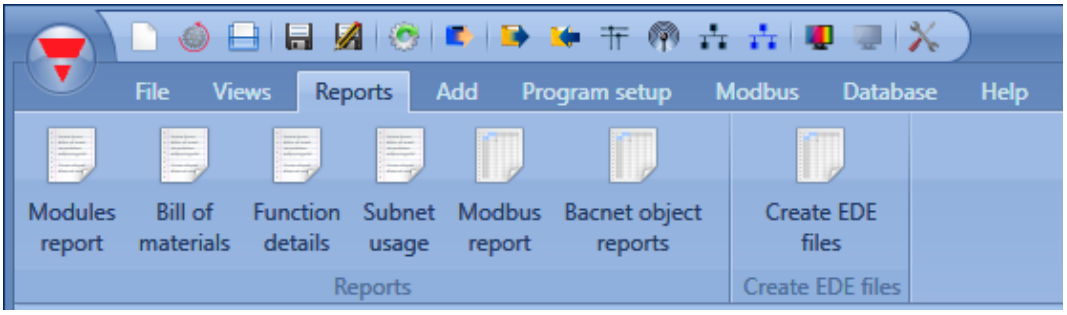
The windows are floating and can be positioned by using the five docking areas shown in the figure below. The position of the windows can be saved.



3.3 Reports menu

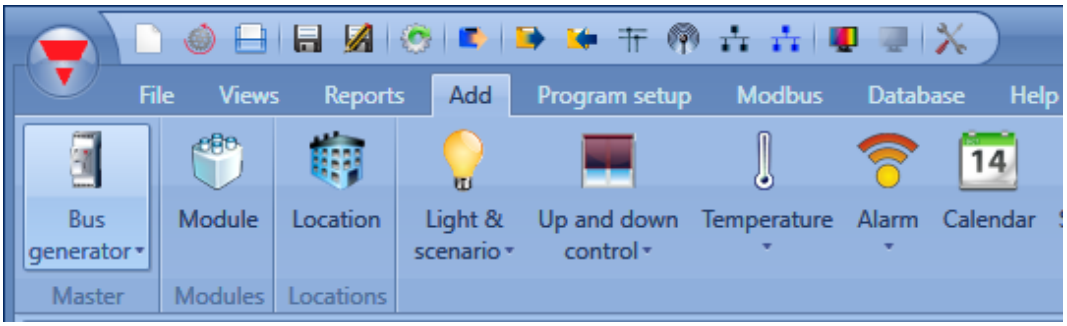
In the reports menu, five different kinds of reports can be created, saved and /or, printed.
The user can select from:

- *Modules list*: the complete list of modules is shown.
- *Bill of materials*: the list of modules is organized by type of modules.
- *Function details*: each function is described with the details of used signals.
- *Subnet usage*: this report indicates the number of used signals and the total current consumption.
- *Modbus TCP/IP report:* in this report the Modbus map of the project is reported.



3.4 Add menu

In the Add menu the user can select what to add: bus extension, modules, locations and functions.



Bus generators: if a new Dupline network is needed, a new bus extension module has to be added.
Module: a new module is added (light switch, pir sensor, output module…)
Location: The user has to define floors, rooms or any other type of location to have a clear structure of the installation.
Functions: some predefined functions can be defined and configured by means of the wizard tool. The predefined functions are:

- Lights & scenario
- Up and down control: for controlling blinds, curtains, windows
- Temperature System Functions
- Alarm: intruder, smoke, water, siren
- Calendar: can execute activities during the year
- Sequence: executes a list of chosen functions
- Dimmer sequence
- Timers
- Basic functions: this section contains Counter, Logic Gates, Analogue comparator, Mathematical function, Analogue output, Hour counter
- Simulated habitation: records and then plays back the light activations

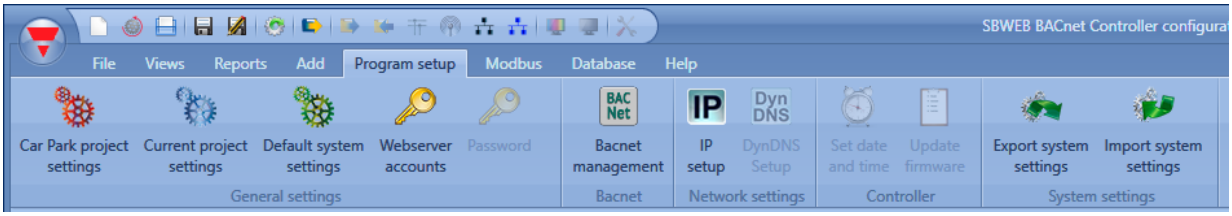
- Sms setup: manages the SMS for the remote control of functions
- Email
- Car heating

For the building automation functions, database management, communication protocols and any other function that is not covered here, please refer to the system manual at this link:

http://www.productselection.net/MANUALS/UK/uwp3.0_tool.pdf.

3.5 Program setup menu

In the *Program setup* menu, the user can configure the settings relevant to a specific project, the general settings of the installer, the IP address and time and date of the UWP 3.0. The user can also update the firmware and configure the webserver and the password to access the controller.

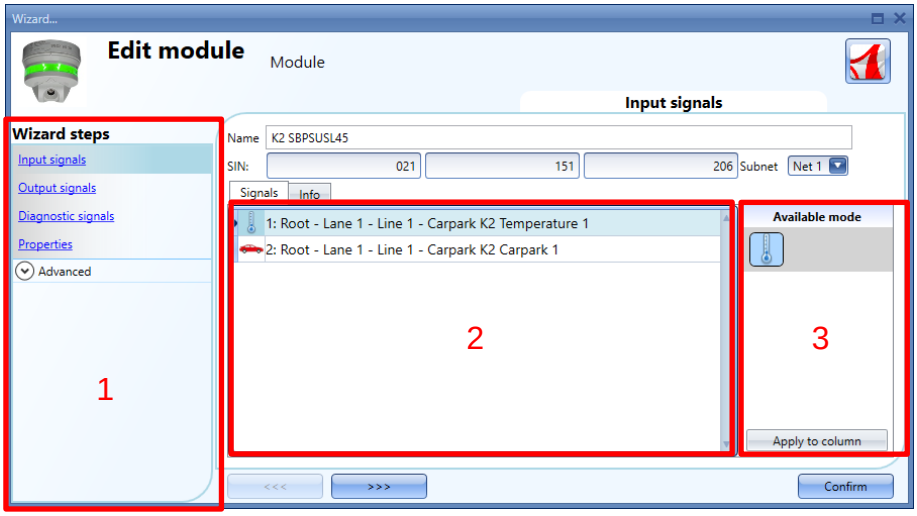


4 Project structure

In a project the user has to define the locations of the installation, add the required modules (sensors, display…) and place them in the relevant location. then create the functions.
Location, *modules* and *functions* are the pillars of the project structures: they are described in detail in the following paragraphs.

4.1 Wizard

Each object, whichever type it is, is created and configured by means of the *Wizard* tool.



The wizard is a tool that drives the user in the configuration of an object, guiding them in small steps to the complete setup of a module, location or function. The aim of the wizard is to reduce the effort of understanding the complete process of a configuration, making it easy and fast. The different steps can be filled in one by one just by clicking on the *">>>"* button and going through all of them, or by clicking on those required visualized in blue in the area on the right (Area 1).

In the picture above an example of the wizard tool divided into three areas is shown.

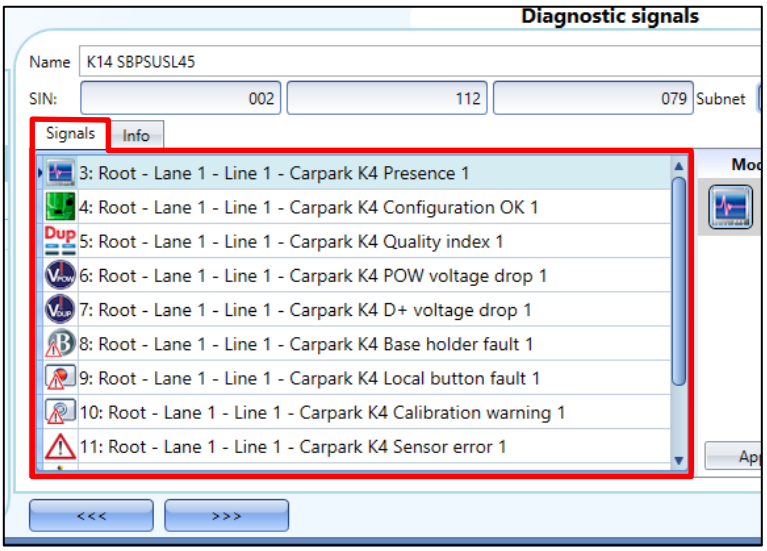
4.1.1 Area 1

Area1 is divided into two sections. The one on top contains the basic signals/settings the user has to insert/define to create the object ('object' is a general word to indicate location, modules or functions). For example, in a light function the basic signals are the input and the output signals, which are the minimum settings required to create this kind of function. In general, most functions show the input and output signals in the basic settings.

In the lower section the "Advanced" functionalities are editable: if they are not enabled they are hidden to the user in order to give an easy user interface to the not-so-skilled installer.
The list of Advanced functionalities will appear: tick the relevant box to enable the one required. For example, if the Lux sensor is to be used, click on this. Once the complete list disappears, by clicking again on *Advanced*, the enabled one/ones will appear for quick access.

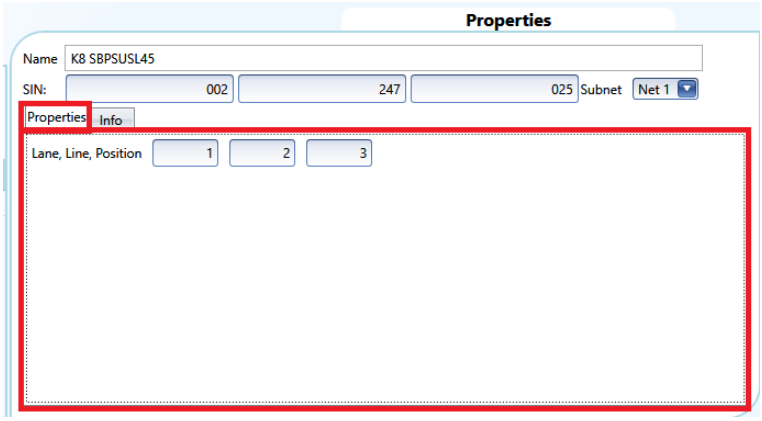
4.1.2 Area 2

This is the area where the available signals are shown or where properties can be changed.



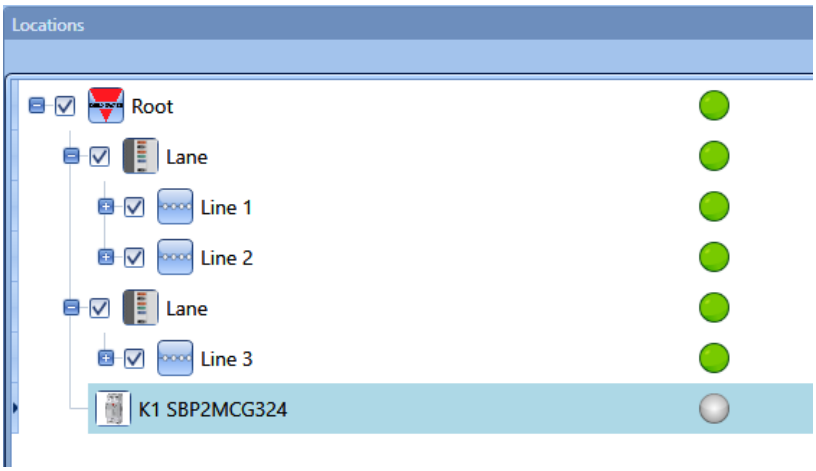
4.1.3 Area 3

This is the area where properties of signals are shown and, in some modules, also changed.



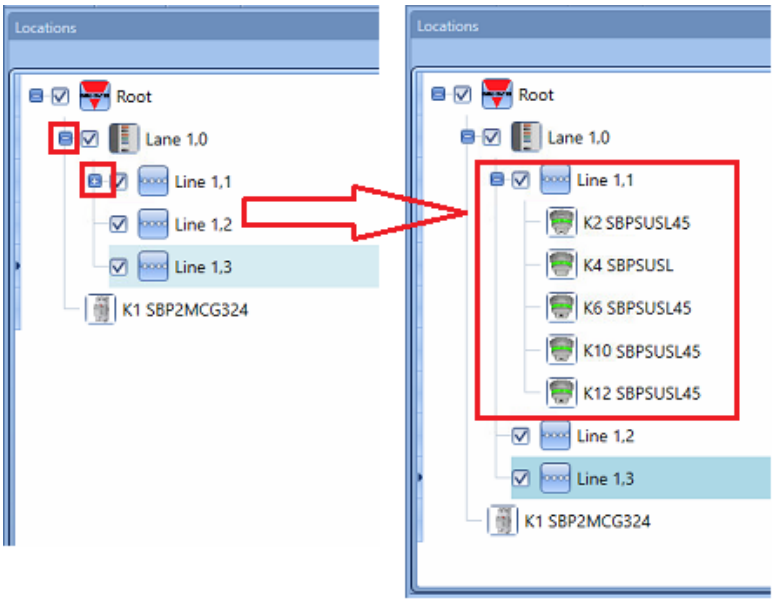
4.2 Locations

In the *Locations* window the user can define the structure of the project, starting from the locations where modules have to be placed: the user has to define lane, line or any other type of location to have a clear structure of the installation.



From the Location window, the user can select which locations have to be shown or not: all the modules and functions related to the locations will be hidden/shown depending on the locations selected in the project tree.

Click on the *plus sign* (+) to expand a Location and to see its sub-locations and modules:



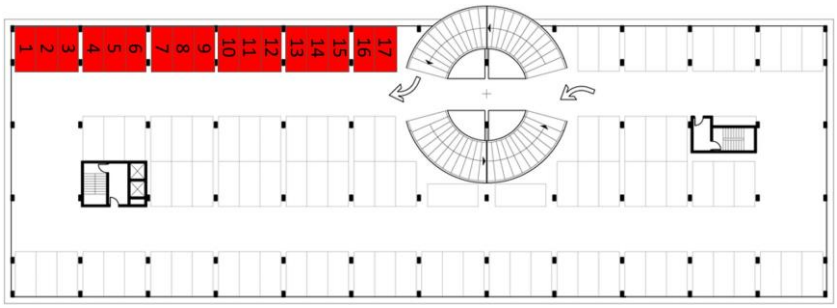
The checked box next to each Location indicates that it will be shown by default. The user can choose which Locations to show or to hide by clicking on the relevant symbol.

- To show the modules associated to a location, the check box must remain checked
- To hide the modules associated to a location, the user has to uncheck the location. The objects associated to its sub-locations will also be hidden.

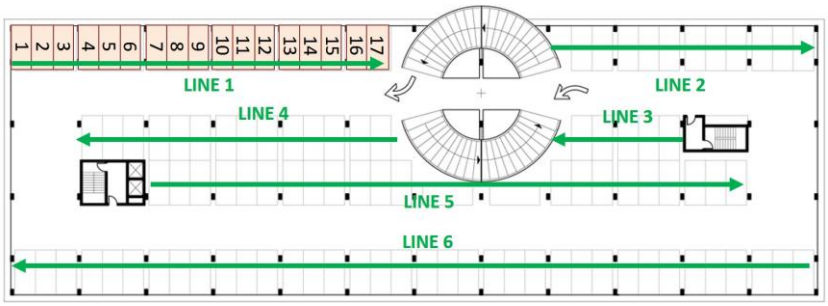
5 Lane, line, position

In a car park project commissioned with the UWP 3.0 Tool and the CPY server, it is important to understand the concept of lane-line-position to program the sensors and the displays.

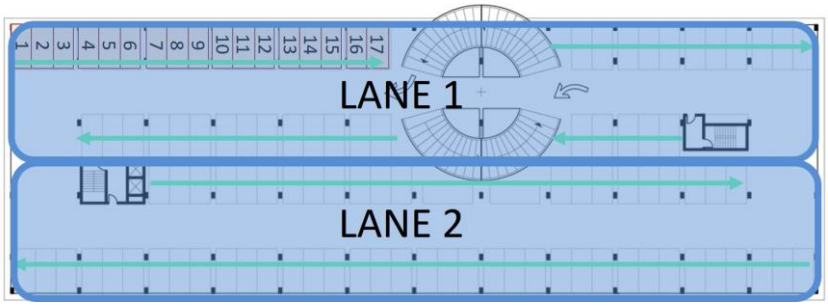
Position: sensors are grouped and each sensor in a group is represented by a number which corresponds to its address. In the picture below, there are 17 sensors, and each of them is identified by a number from 1 to 17.



Line: any group of sensors is a line: any line is also indicated by a number

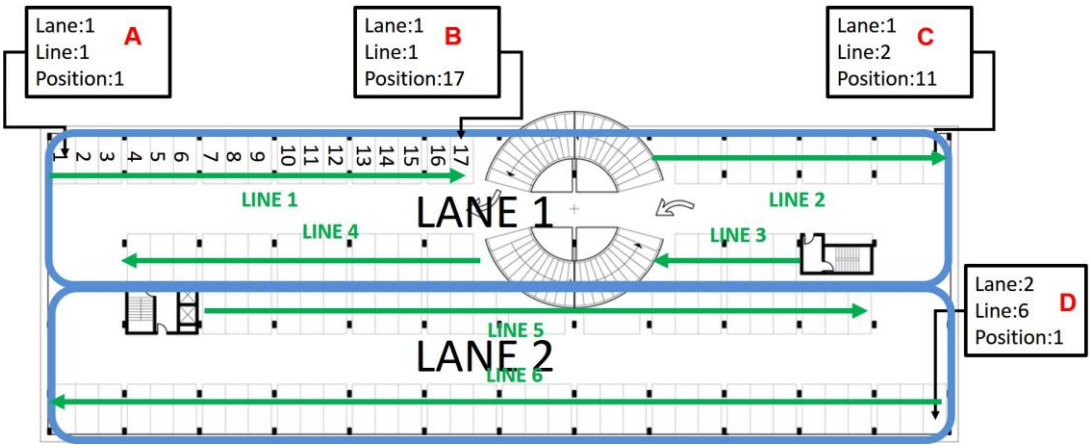


Lane: any group of lines is a lane: each lane is indicated with a specific identification number in the project

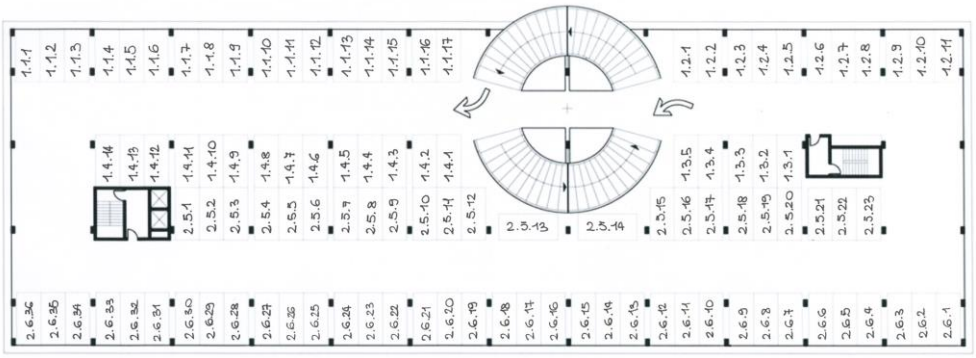


According to the grouping explained above, each sensor is uniquely identified by its position in a line: the car park address is made up of three numbers, each separated by a dot, very similar to the SIN number, and it represents the lane, line and position of the sensor.

Address sensor A: 1.1.1 (lane.line.position)
Address sensor B: 1.1.17
Address sensor C: 1.2.11
Address sensor D: 2.6.1

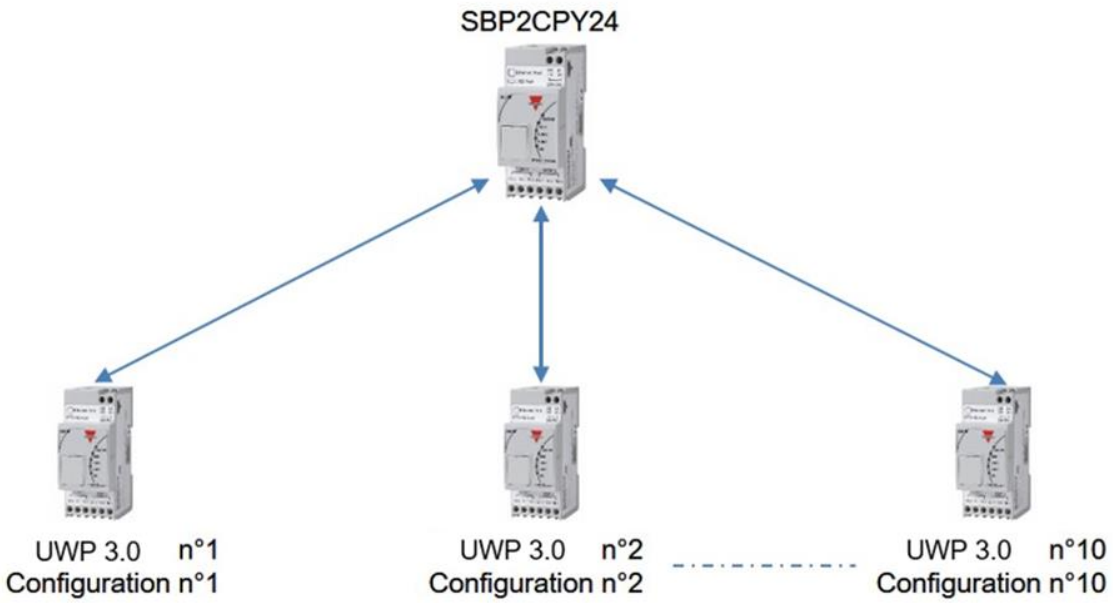


In the following image, the whole car park has been mapped with the address of each sensor.

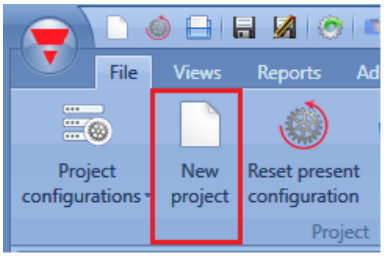


6 How to create a new project

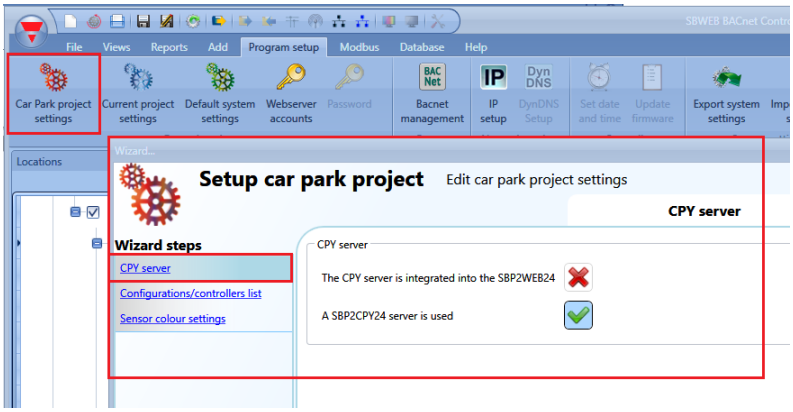
A car park project can include up to ten configurations for up to ten UWP 3.0s each, communicating with one SBP2CPY24, as shown below.
The SBP2CPY24 includes the CAR Park server, which can be integrated into the UWP 3.0 in small installations.



When the tool is opened or when a new project is created by clicking on the icon *New project,* one configuration is added by default.



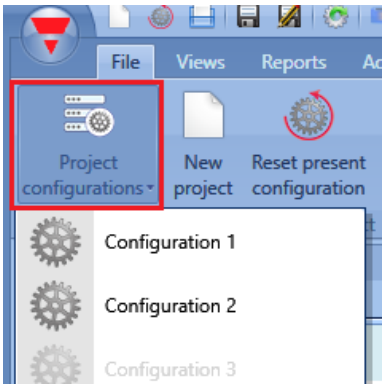
In the new project, select where the car park server is located: if there is only one UWP 3.0 the car park server can be integrated into it: otherwise, in the case of a multi-configuration project a SBP2CPY24 is needed. In the *Program setup* menu, click on *Car Park project settings* and then in the new window select *CPY server:*



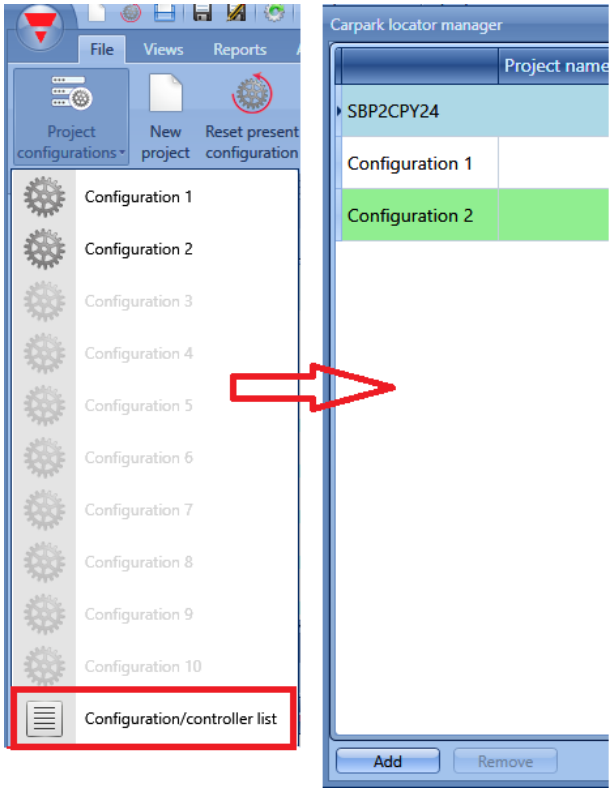
Should the CPY server be included into the UWP 3.0, the address to access is:

***[controller_ipaddress]*/CP3App**

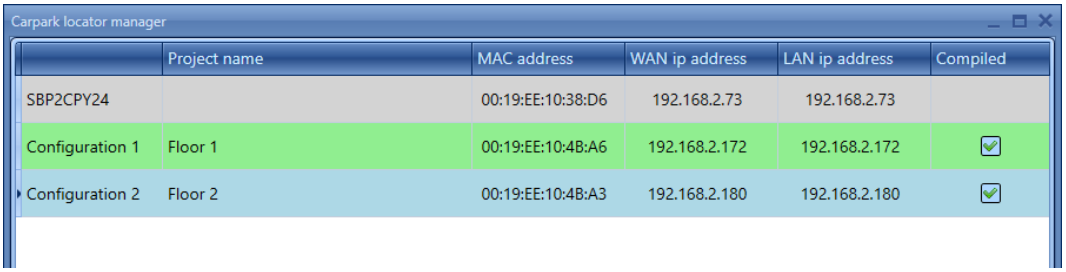
The list of configurations is displayed by clicking on the icon *Project configurations:*



To add a new configuration to the project with the relevant UWP 3.0, click on *Configuration/Controller list,* as shown in the picture below:



The *Car park locator manager* window will appear with the list of the configurations and the UWP 3.0 controllers associated to them:

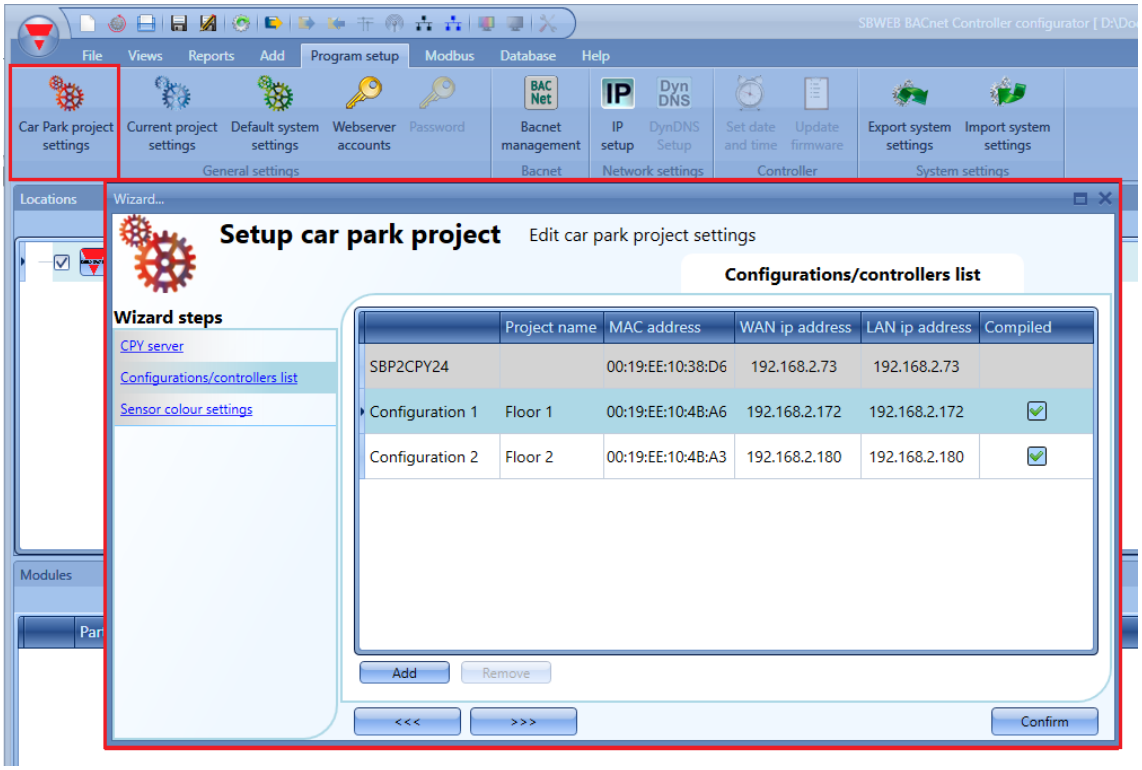


The first item on the list is the CPY server that can be integrated into the UWP 3.0 controller or the SBP2CPY24 dedicated item in a multiconfiguration project. Later, the different UWP 3.0s configurations (up to 10) are shown.

The following table shows the available parameters:

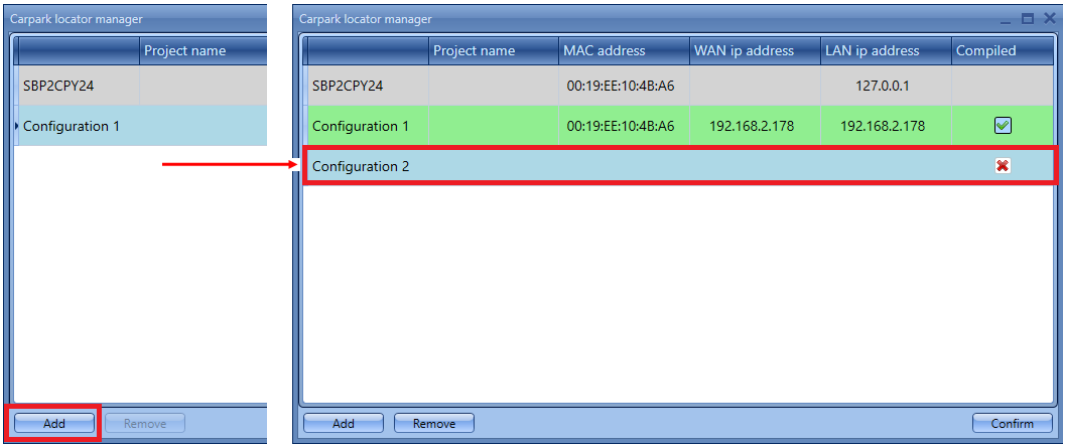
Field	Description
Project name	This field shows the project name of the configuration, the user can change it
MAC address	This field shows the physical address of the UWP 3.0/SBP2CPY24 modules
WAN IP address	This field shows the current public IP address of the UWP 3.0/SBP2CPY24: this identifies the IP address (or DynDNS name) the user can insert to get the access from the Internet (when the user needs the connection of the LAN where the Car Park modules are installed)
LAN IP address	This field shows the IP address of the UWP 3.0/SBP2CPY24 devices in the LAN where the Car Park module is installed
Compiled	When a small red cross is shown, the user has to compile the project before sending the configuration to the UWP 3.0 controller

Alternatively, the user can access the same list from the *Car Park project settings* window:



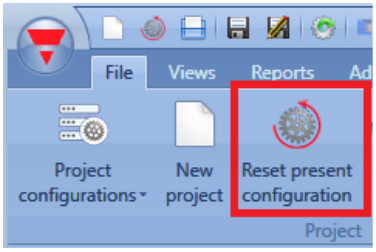
6.1 How to add a new configuration to the current project

From the *Carpark locator manager* window, by clicking on the *Add* button, a new line is added for a new configuration, as shown in the picture below:

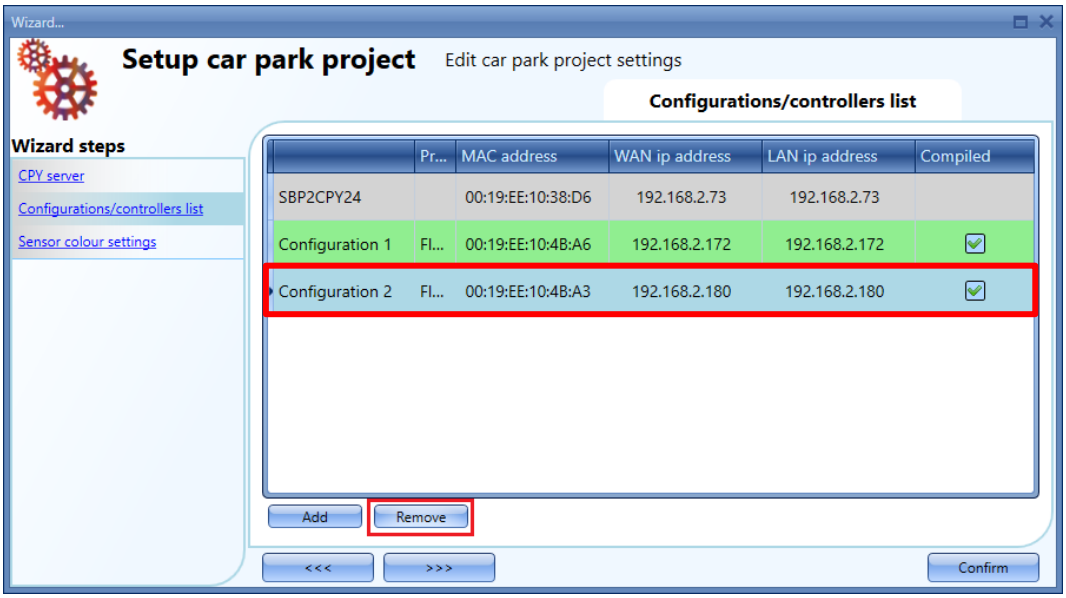


A new configuration can be added to the project at any time.
The green line indicates the configuration that is now open in the project.

To reset a configuration in the project, select the configuration from the list and click on *Reset present configuration*: the reset will delete all the locations, modules and functions.



To remove a configuration, select it from the list and click on *Remove,* as shown below:

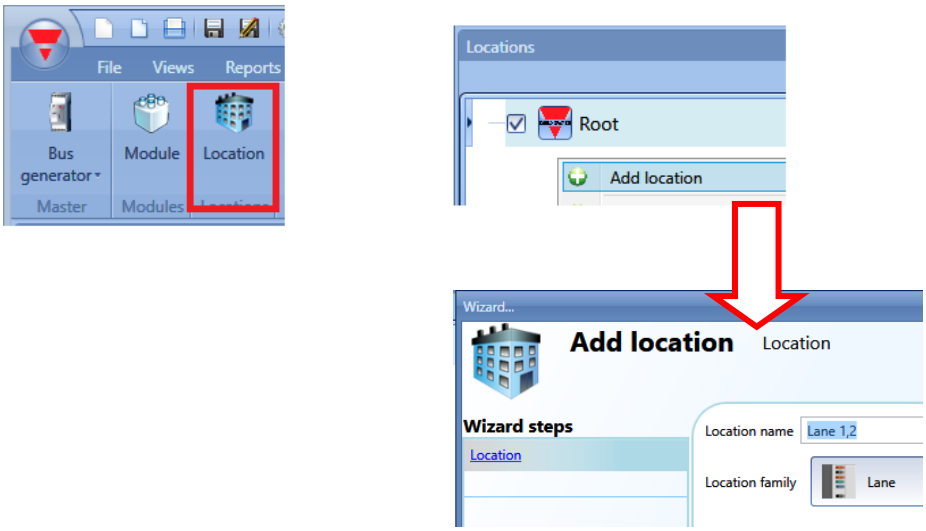


Please note that the present open configuration cannot be deleted.

Once the structure of the project has been defined with the relevant configurations, UWP 3.0s and CPY server, the location tree has to be generated.

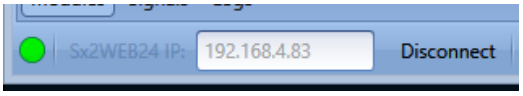
There are different ways to create a location tree:

1) Starting from the *Add* menu from the top menu or by right clicking in the *Location* window

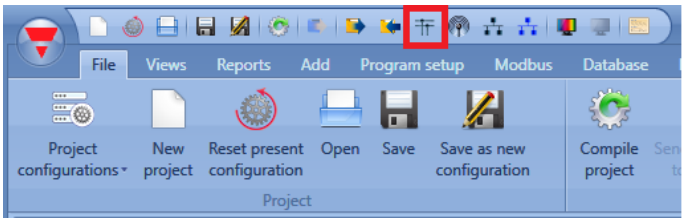


2) In a faster way, the location tree can also be generated starting from the *Discovery manager window:* to open this, follow these steps.

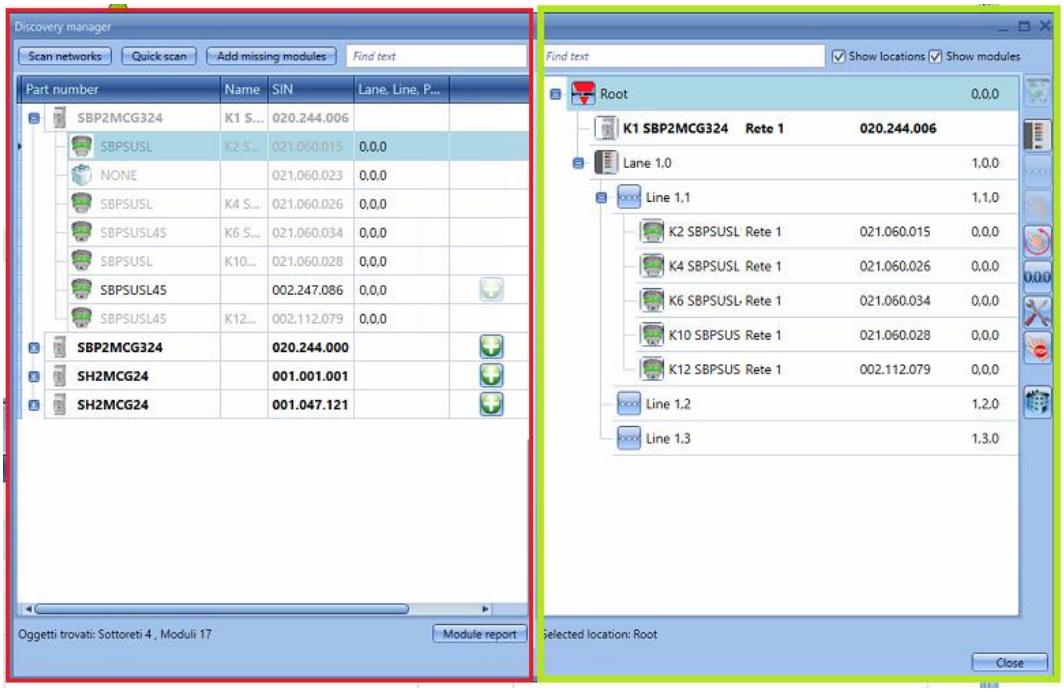
a) Connect to the controller



b) Launch a discovery of the Dupline module



c) The *Discovery manager* window will appear:



On the left of the window (marked in red), all the modules connected to the bus are prompted once the *Scan networks* push button is pressed.

- The *Quick scan* button will list only the modules connected to the Master Channel Generators that have already been configured in a valid configuration
- The *Scan networks* button will list all the modules that are connected to the different Master Channel Generators, regardless of whether they have been configured or not

On the right, the location tree is shown. To generate this, only three push buttons have to be used:

To delete a location and all its objects, the push button with the basket has to be used

This adds a location *Lane*. It is enabled if locations such as Root, Floors, etc are selected. It is not possible to add a Lane from a Lane location or from a Line location.

This adds a location Line. It is enabled only if a location Lane is selected since a line is a group of bays (parking spaces) in the lane

This starts/stops the single line addressing

This starts/stops the multi-line addressing

This resets the car park addresses of all the sensors

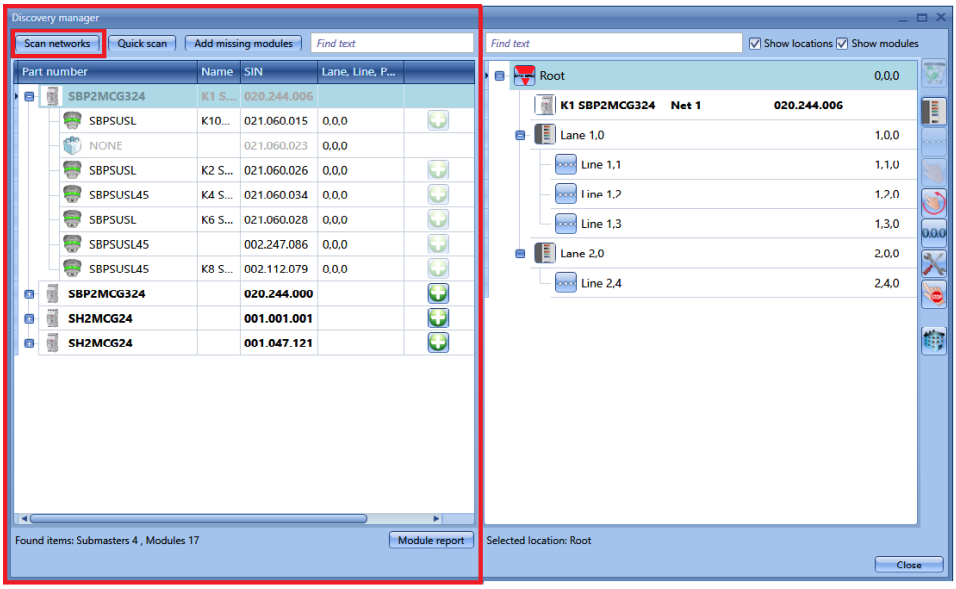
This stops any running activity such as a network discovery or an automatic addressing

This adds any type of locations. It is not enabled if a location line is selected.

Once the location tree is ready, the user can proceed to add the modules: there are two ways of doing this, an automatic scan of the network launched by the *Discovery manager* window, and a manual mode.

6.2 How to automatically find and address the sensors – The controller is connected to the modules

In the *Discovery manager* window, click on *Scan networks*: all the modules present on the bus will be prompted.



Once the modules are discovered, they have to be placed in the locations and car park addresses have to be assigned.

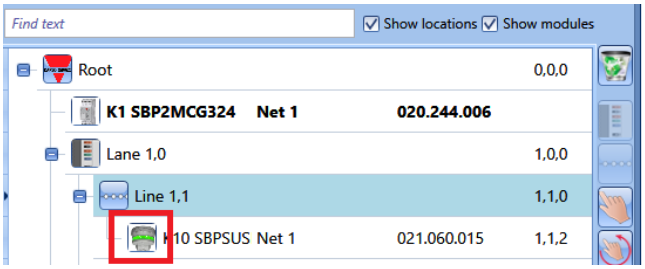
6.3 How to manually place the discovered modules

There is one important rule to understand when placing modules: the sensor SBPSUxx can be placed only in the location *Line,* and in the location *Line* only SBPSUSxxx sensors can be placed. There are no restrictions for the other modules/locations.

Modules can be manually placed in the locations in the following ways:

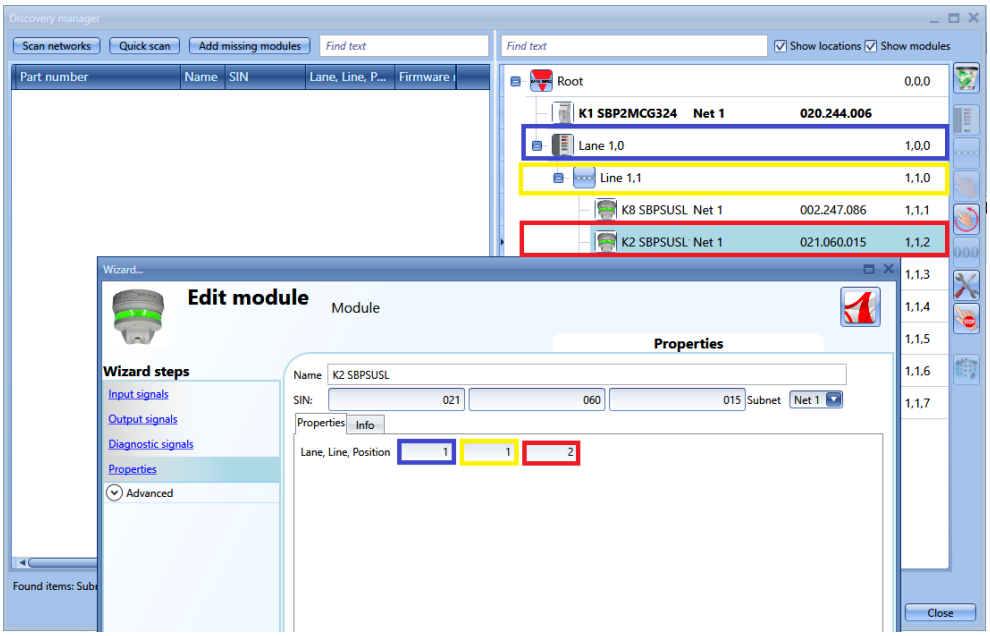
1) By selecting the locations and then clicking on the  icon.
This icon will be enabled only if the selected location is allowed to contain that type of module.
2) Modules can be dragged and dropped onto the selected location: the drag&drop functionality also follows the rule mentioned above.

When modules are placed manually, the car park addresses also have to be entered manually, taking into account the lane and line addresses and the position of the sensor in the line.
Click on the icon relevant to the module in the location tree:



The wizard of the module will be opened.

In the *Properties* field enter the number of the lane and line the module belongs to and its position in the line.



6.4 How to automatically place the modules

There are two ways to place modules in the lines:

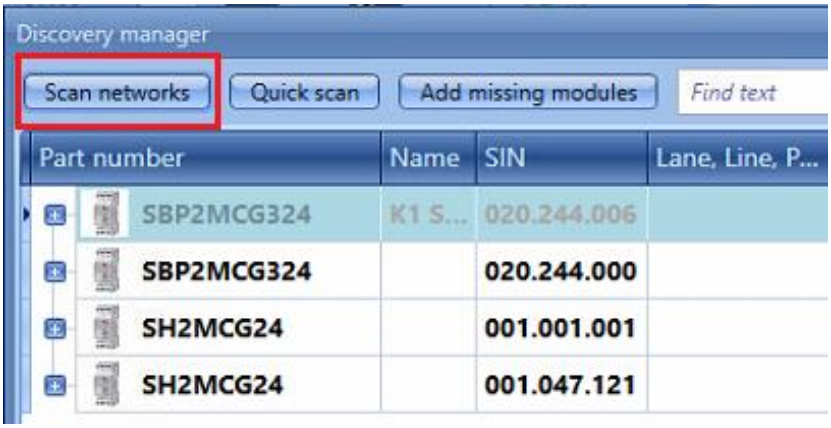
1) line by line, i.e. *Single line addressing*
2) more lines at the same time, i.e. *Multi line addressing*

6.4.1 Single line addressing

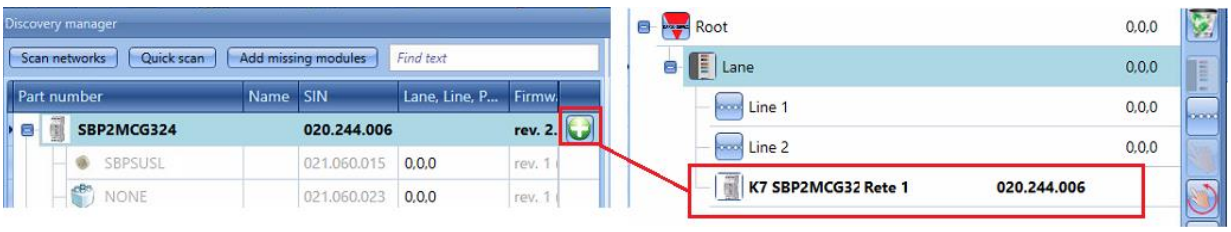
This type of addressing is available only if a location *Line* is selected.
The auto addressing lets the user place the sensor in the selected line simply by clicking on the push button on the sensor: the car park address will be set automatically. This procedure will address only the sensor present in the selected line.

The following steps have to be executed:

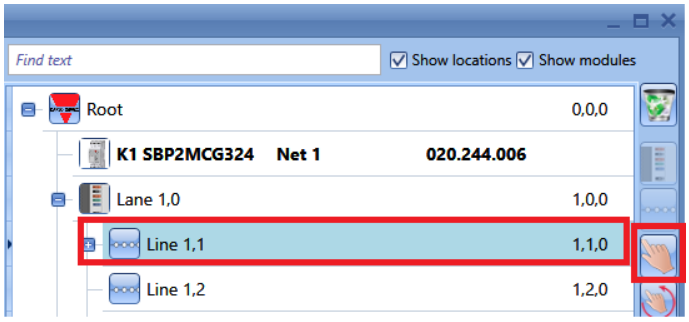
1) Launch the modules discovery



2) Add the SBPMCG324 (three-wire Dupline generator) to any location, except to location *Line*

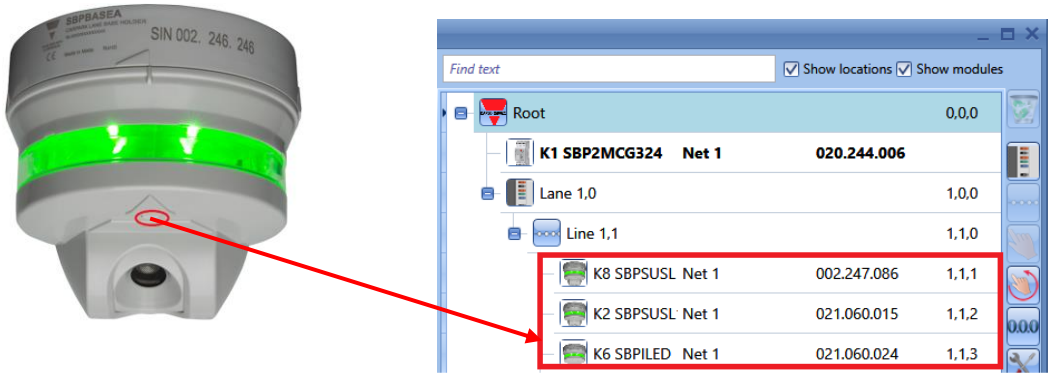


3) Select the line where the sensor is placed and click on the icon marked in red to start the automatic addressing



All the sensors with no address will start blinking a yellow LED.

4) Walk to the line where the sensors to be addressed are and press the button on the sensors one by one: the car park addresses will be assigned automatically and the sensors will be shown in the correct position in the location tree



As highlighted in red, as soon as the push button on the sensor is pressed, the sensor is moved from the left side of the *Discovery manager* window to the right side and the car park address is shown next to the SIN number.

5) Once all the sensors belonging to the line have been addressed, click again on the icon to finish the procedure.

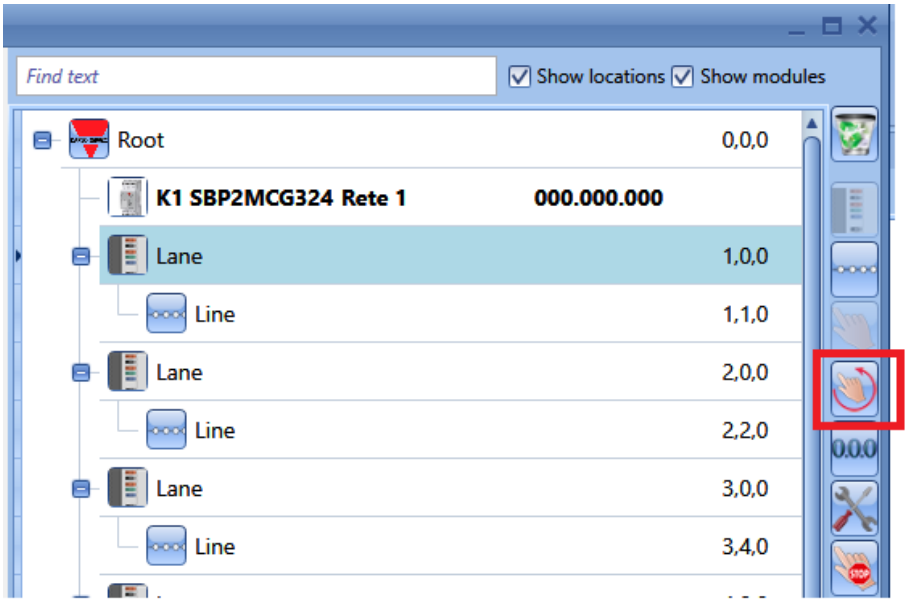
6.4.2 Multi line addressing

This procedure lets the user consecutively address the sensors belonging to many lines, without launching the procedure described above many times.
The user just needs to launch it once and walk around the parking area, pressing the buttons on the sensors according to the line order defined in the UWP 3.0 Tool.

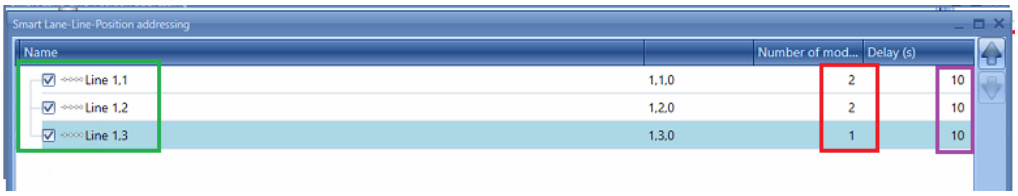
The following steps have to be executed:

1) and 2) as in the *Single line addressing* mode

3) Select any location and click on the icon marked in red:



The following window will appear, in which the user has to select the lines to program by clicking on the boxes highlighted in green.



In the field marked in red, the number of sensors for each line has to be edited: in this way the system knows when one line is completed and it is time to switch to the following one/s.
In the field highlighted in purple, the delay between the addressing of two lines has to be set: this delay is needed to allow the installer to walk from one line to the following one: it can also be set to 0.

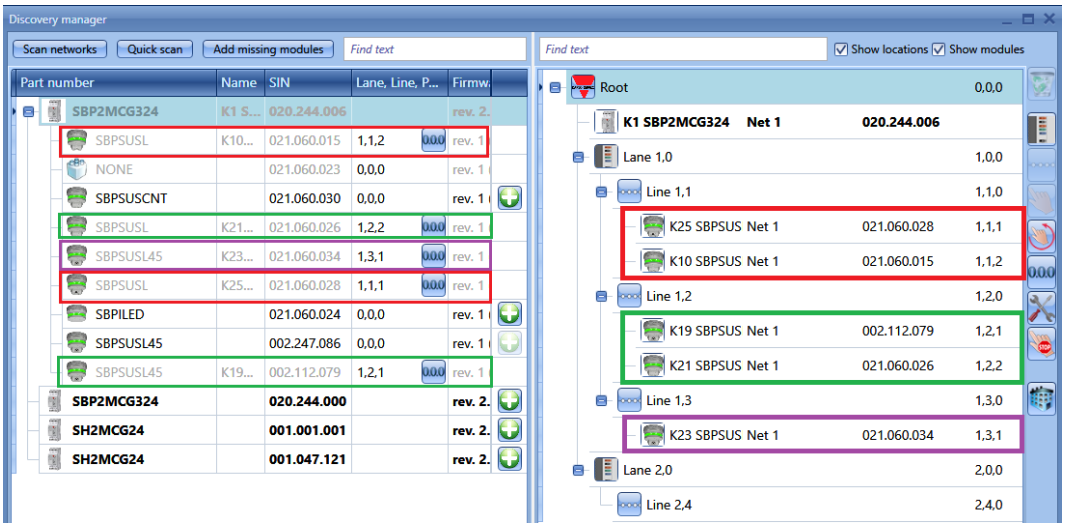
Once all the settings have been completed, click on *Confirm* to start the addressing: all the sensors will start blinking.

4) Walk to the lines to be addressed and press the button on the sensors: they will be added to the lines according to the order in which they are activated and the *Number of modules* set for each line. ***Please wait until the sensor that has just been programmed does not stop blinking fast before pressing the button on the following one.***

Example

1 According to the settings shown in the picture above, the first two sensors whose buttons are pressed are placed in line 1.1
2 The sensors stop blinking for 10 seconds
3 Automatic addressing is enabled again (the sensors will start blinking again) and the following two activated sensors will be placed in line 1.2
4 Then there is another delay of 10 seconds during which the sensors stop blinking
5 Automatic addressing is enabled again (the sensors will start blinking again) and the following activated sensor will be placed in line 1.3
6 After another delay of 10 seconds, the intelligent addressing procedure is closed

5) The tree of the installation will be completed as shown in the picture below:



The first sensor whose button has been pressed is the one with SIN number 021.060.028 and it is placed in line 1 of lane 1, so its car park address will be 1.1.1

The second sensor is the one with SIN number 021.060.015 and it is placed in line 1 of lane 1, so its car park address will be 1.1.2

At this point the system automatically changes to line 2, since we have told it that two sensors have to be placed in line 1.

The third sensor whose button has been pressed is the one with SIN 002.112.079 and it is placed in line 2 of lane 1, so its car park address will be 1.2.1

The fourth sensor is the one with SIN 021.060.026 and it is placed in line 2 of lane 1, so its car park address will be 1.2.2

At this point the system automatically changes to line 3, since we have told it that two sensors have to be placed in line 2.

The fifth sensor whose button has been pressed is the one with SIN 021.060.034 and it is placed in line 3 of lane 1, so its car park address will be 1.3.1

At this point the system automatically closes the procedure.

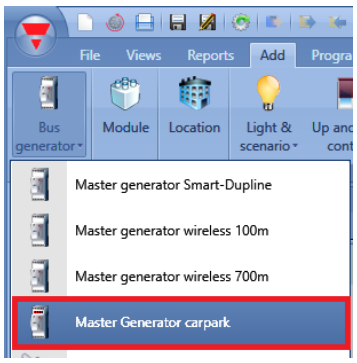
In the *Discovery manager* window on the left, as soon as the 0.0.0 button is pressed, the relevant module is put in light grey and the only action that can be done is to reset the car park address by clicking on

6.5 How to manually add modules – The controller is not connected

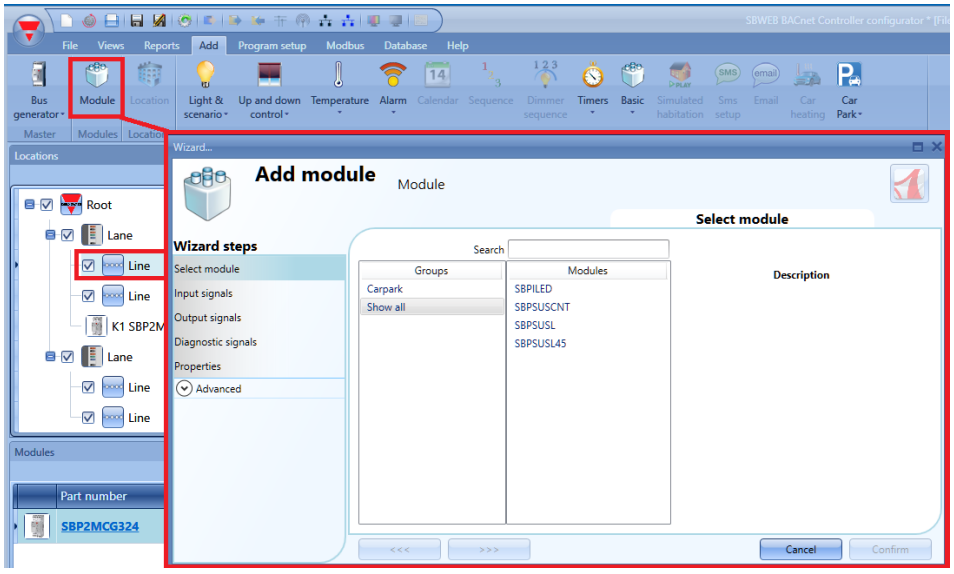
If the configuration has to be built off line (without the connection to an UWP 3.0 or to any Dupline network), the modules can be added and addressed manually.
The following steps have to be followed to create the project.

1) Generate the location tree by clicking on *Add Location*

2) Add the three-wire master channel generator UWP 3.0



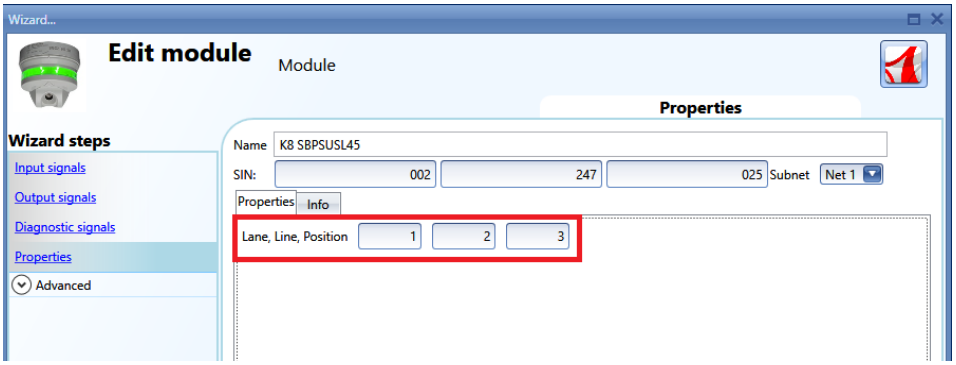
3) Add the sensors in the relevant *Line:* select the Line and click on *Module*. Select the sensor and click on *Confirm*.



In the location *Line* only the sensors SBPSUSxxx and the SBPILED indicators can be placed.

4) Once the modules are added to the project, the addressing can be done manually or it can be

done automatically at a later moment when it will be possible to connect to a Dupline network. **We strongly suggest doing it automatically to speed up the installation time and to reduce the possibility of making errors.**
Should this not be possible, the lane.line.position address can be edited manually from the wizard of the module, as shown in the picture below:

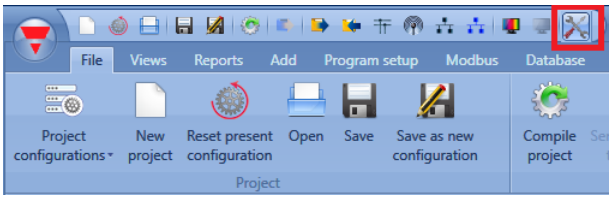


7 How to calibrate the sensors

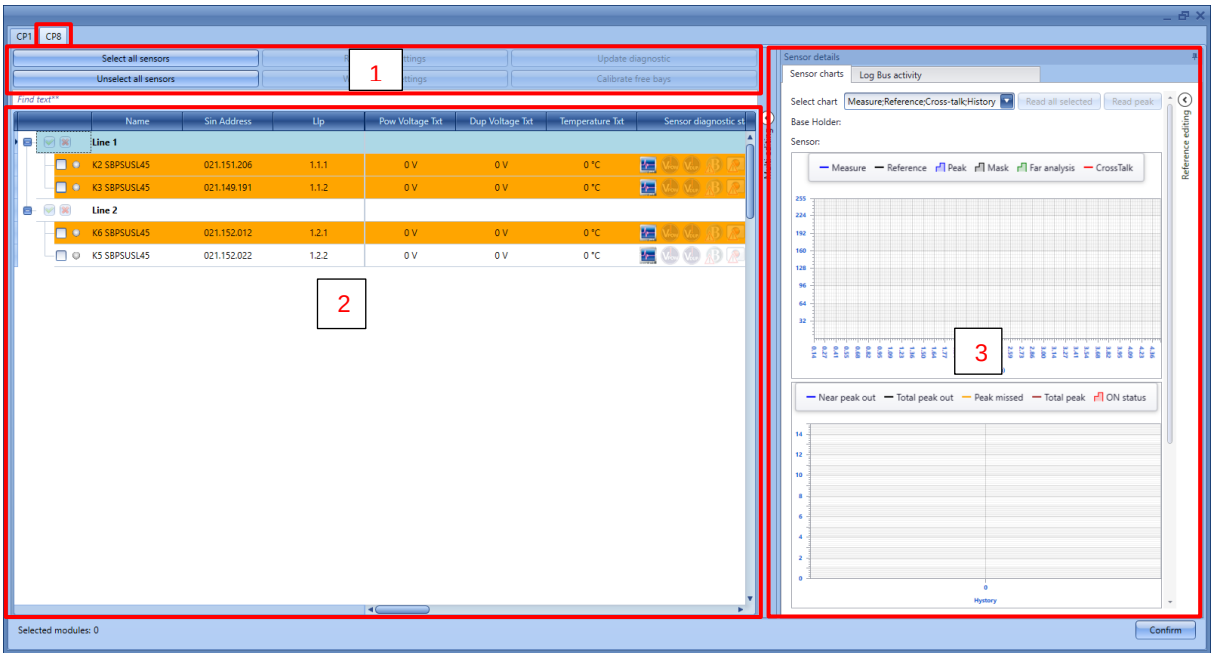
Once a sensor is mounted over the parking bay, it **needs** to be calibrated to distinguish between a vacant parking space (bay) and an occupied one. The calibration has to be carried out with no cars and whenever any structural changes are made in the parking bay.

The calibration has to be done after writing the configuration into the UWP 3.0 controller and when the parking bay is empty.

To start the calibration process, click on the icon marked in red once the controller is connected to the UWP 3.0 Tool:



In the *Calibration* window, the user has to click on the *CP8* tab; the following window will appear:



The *Calibration* window allows the user to calibrate, check and set the parameters for all the sensors that are present in the configuration. The window is divided into three areas:

7.1 Area 1 – Commands

This area contains the buttons for commands and operations, as in the table below:

Button	Behaviour
Select all sensors	Press this button to select all the sensors present in the current configuration, regardless of the *Lines* they belong to
Deselect all sensors	Press this button to deselect all the sensors
Read sensor settings	Press this button to read the settings from the selected sensors
Write sensor settings	Press this button to write the new settings to the selected sensors
Update diagnostics	Press this button to read the diagnostic parameters for the selected sensors
Calibrate free bays	Press this button to start the calibration for the selected sensors

7.2 Area 2 – Sensors list

The ***Sensors list*** shows all the information and settings related to the sensors, grouped by *Lines*. The sensors are highlighted in different colours, the meaning of the different statuses can be seen in the table below:

Highlighted colour	Status description
Red	The sensor is in occupied status
White	The sensor is in vacant status
Orange	The sensor needs to be calibrated
Light yellow	The sensor is being calibrated
Light blue	The sensor is selected in the *Sensor list*

For each sensor, the user can check information such as the part number, the SIN address, etc.. and they can change the settings parameters. The description of all the parameters is shown below:

Field name	Description
Name	This field shows the *part number* of the sensor module
Sin Address	This field shows the *SIN* address

Llp	This field shows the *Lane.Line.Position* address assigned to the sensor
Pow Voltage Txt	This field shows the Power Voltage value
Dup Voltage Txt	This field shows the Dupline Voltage value
Temperature Txt	This field shows the temperature value
Sensor diagnostic status	This field shows the diagnostic icons. *See Diagnostic chapter for details*
Filter	This field defines the number of measures the sensor performs to define the occupancy status. The higher the value, the more accurate the status will be. (Default value is 8 measures, which correspond to 3 seconds)
Near peak out	This field defines the minimum number of peaks that must be detected in the "near" area to change the Parking bay from vacant to occupied status. *(Default suggested value is 1 Peak)*
Total peak out	This field defines the minimum number of total peaks that must be detected both in the *Near* and *Far areas* to change the parking bay from vacant to occupied status *(Default suggested value is 2 Peaks)*
Near end position	This field defines the end position of the *Near area*. The *Near area* is the zone where the cars should be parked. The *Far area* starts where near area finishes
Near peak min value	This field defines the minimum size (expressed in points) which a peak detected in the *Near area* must have in order to be considered as valid
Far end position	This field defines the end position of the *Far area*. The default suggested value is 3.68 metres. Every peaks that is detected beyond that value will be automatically disregarded
Far peak min value	This field defines the minimum size (expressed in points) that a peak detected in the *Far area* must have in order to be considered as valid
Local cal	This field enables/disables the push button present on the sensor for local calibration
Disable led	This field turns ON/OFF the LEDs on the sensor
Loc led occ.	This field locks the LEDs to the colour used for occupied status (by default this is configured as red). *Tips: this condition is useful to keep the sensor lock in red when the Parking bay is under maintenance*
Loc led vac.	This field locks the LEDs to the colour used for vacant status (by default this is configured as green).
Loc status occ.	This field locks the parking bay status to occupied, regardless of the LED colour of the sensor.
Loc status vac.	This field locks the parking bay status to vacant, regardless of the LED colour of the sensor.

The fields **marked in bold** are editable, the others are read-only parameters.

7.3 Area 3 - Graphs

In the right part of the *Calibration* window, by means of a graphical representation, the user can identify what causes a calibration issue, such as wrong settings or a Crosstalk condition that must be resolved.

In the Troubleshooting section of this manual the user can see more detailed information

7.4 How to calibrate the sensors

The procedure shown below is valid only for Car park sensors that are equipped with a firmware revision equal or higher than 8.

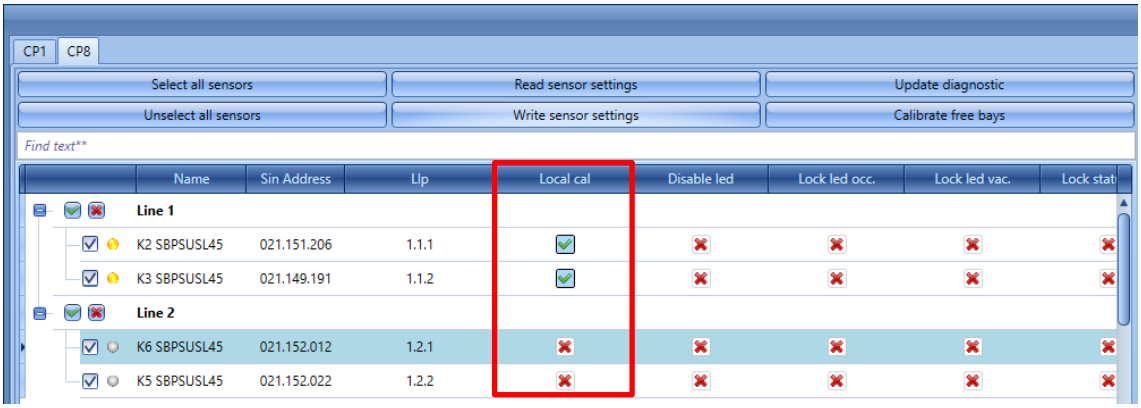
There are two ways to calibrate the sensors: launching the calibration commands remotely from the UWP 3.0 Tool or locally by pressing the push button on the sensor. Please refer to the procedures shown below:

7.4.1 Local calibration

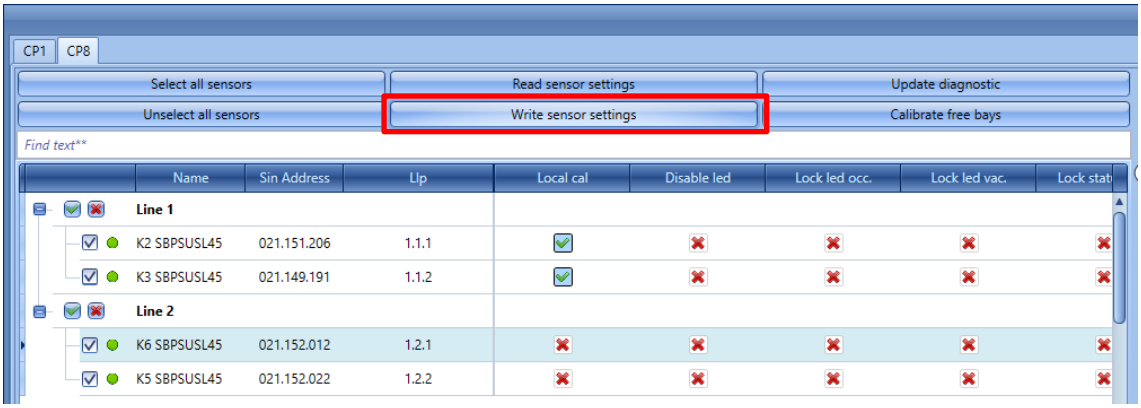
The sensors can be calibrated by pressing the local push button: for security reasons, the push button is disabled by default in order to avoid non-authorised people to press it.

To enable it, follow the procedure shown below:

5) In the *Calibration* window, the user has to select the *Local cal* check-box for all the sensors that have to be calibrated manually. The small dot next to the selected sensors will turn yellow;



2) Click on the *Write sensor settings* button to save the changes: the dot next to the selected sensors will turn green, confirming that the changes are successfully saved;



3) The user can start the calibration by moving from sensor to sensor and pressing the push buttons: the yellow LEDs will flash slowly for 15 seconds in order to have the space empty, then will flash faster when calibrating.

Once the calibration process is completed, we suggest disabling the push-buttons so that non-authorised people cannot use them.

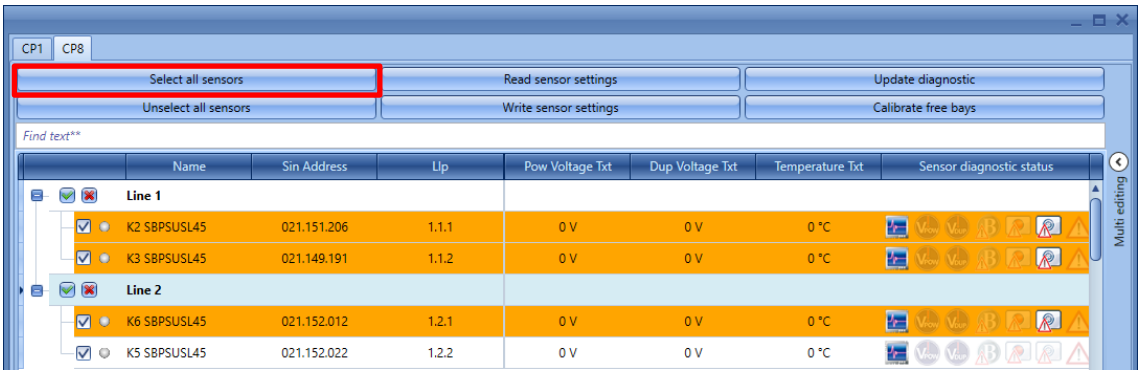
7.4.2 How to calibrate the sensors remotely

The user can calibrate the sensors remotely by using the UWP 3.0 Tool. The procedure is as follows:

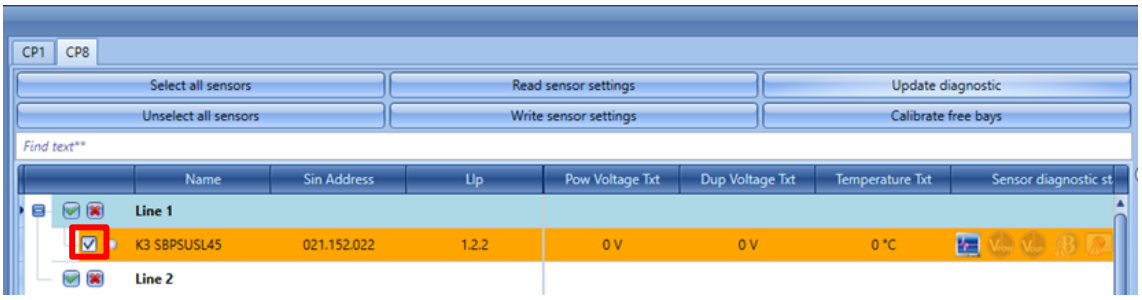
Step 1: Select the sensors

There are three different ways to select the sensors that have to be calibrated:

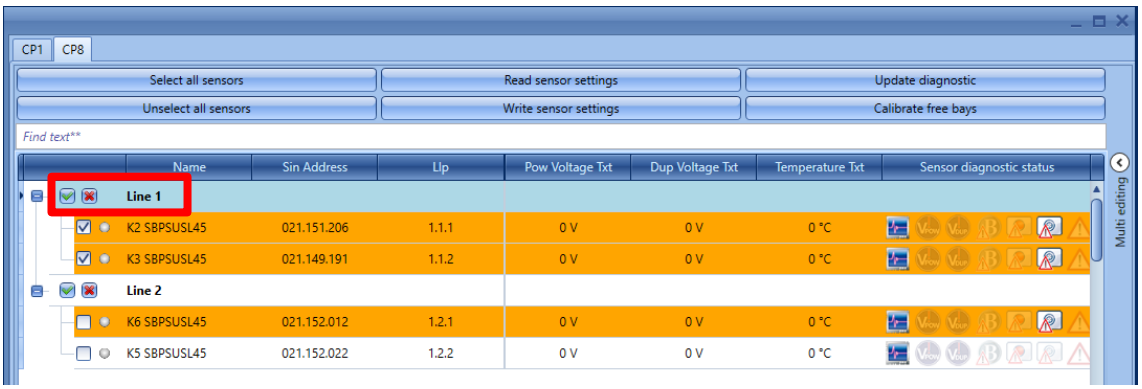
1) Click on the *Select all sensors* button to select all the sensors present in the configuration:



2) The user can select the sensor individually by checking the sensors one by one:

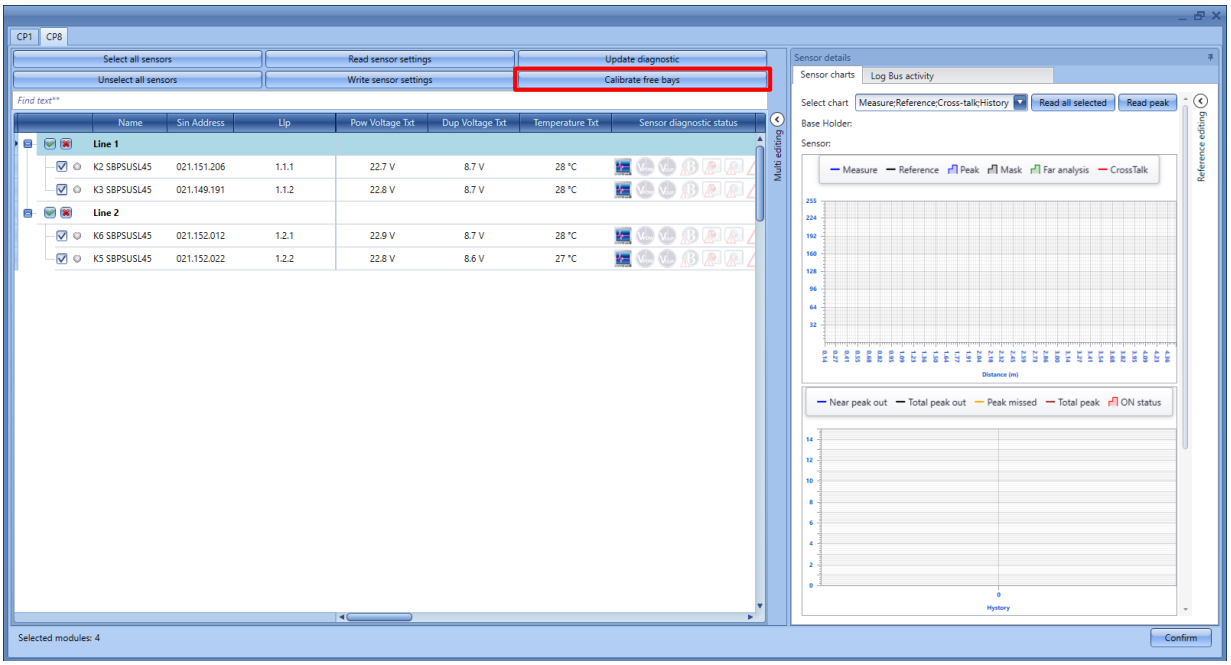


3) The user can select all the sensors that belong to a Line. Click on the Small green icon close to the Line to select them all, click on the small red cross to deselect:

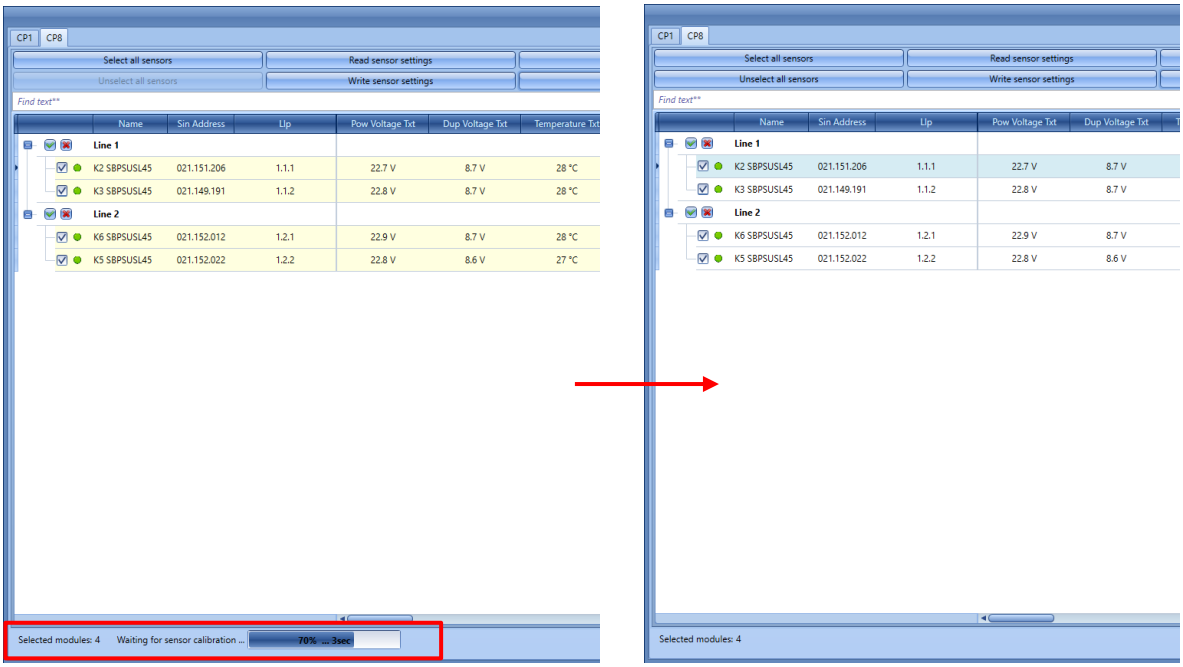


Step2: How to start the remote calibration

After having selected the sensors that have to be calibrated, the user has to click on the *Calibrate free bays* button. The system will calibrate all the selected sensors at the same time.



During the calibration process the selected sensors will be highlighted in light-yellow. In order to check that the operation is being processed, the user can see the operation progress in the bottom part of the window, as shown in the red rectangle below:



Note: If the sensors have not been installed in a standard condition (see the SBPSUSL45 datasheet available on *www.productselection.net* for standard mounting suggestions), or if the user needs to configure them with specific settings, in the *Sensors list* of the calibration windows, for each sensor, the calibration parameters can be changed according to the project requirements.

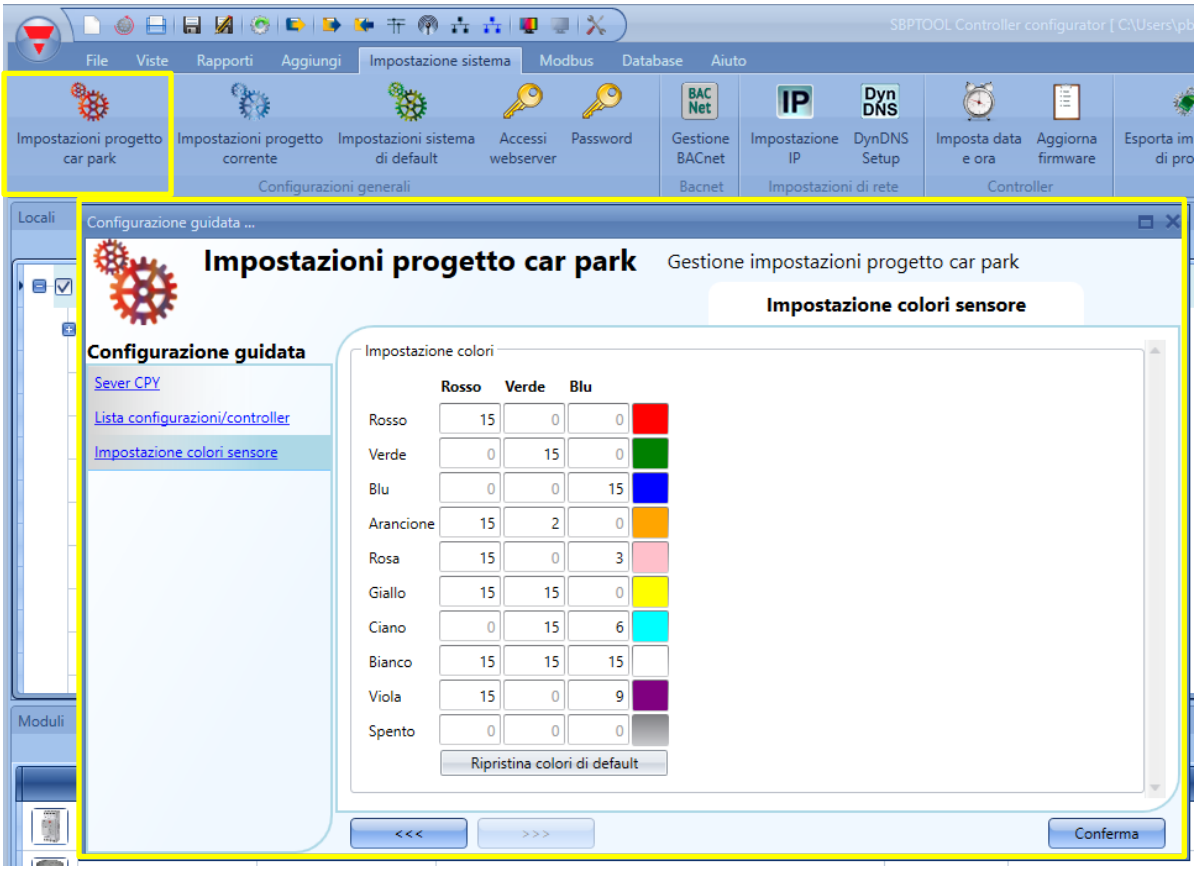
The table below shows the available procedure based on project requirements:

How to…	Procedure
change the sensor parameters	▪ Individually ▪ by using the Multiediting window

8 How to define the LED colours for the SBPUSLxx sensor

The SBPUSLxx sensors have an RGB LED for which the avaiable colours have to be defined from the *Car park project settings*.

Those colours will be associated to different types of category indicated by the sensors, such as free, occupied, for VIP, for pregnant women, and so on. The association will be carried out via the CPY server (see the relevant manual).



9 How to program the controller UWP 3.0 and the SBP2CPY24

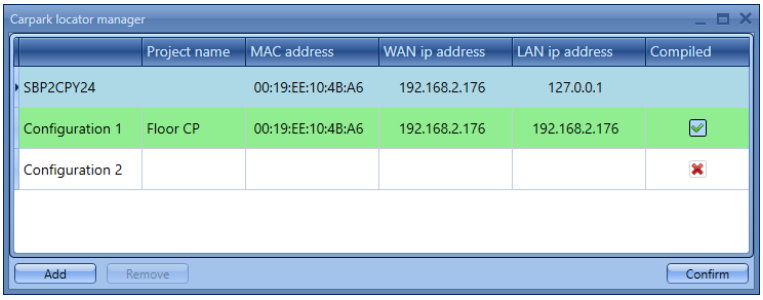
Once the project is completed, the configuration has to be downloaded into the UWP 3.0 and into the car park server SBP2CPY24.

Please note that the SBP2CPY24 has to be programmed before the UWP 3.0.

9.1 UWP 3.0

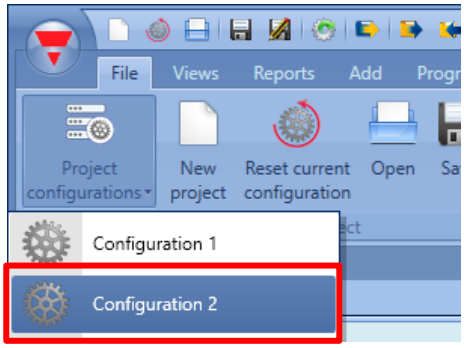
1) In a multi-configuration project, select the configuration to send and connect to the relevant controller.

   a. From the *Carpark locator manager* window the user has to add a new configuration by clicking on the *Add* button: a new configuration line will be added



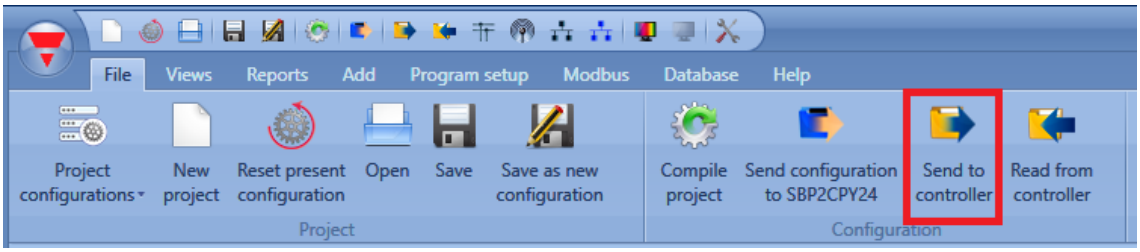
   b. Click on *Confirm* button to save the change.

   c. From the Project configurations select the new configuration that has been added, as shown in the example below the user selects the *Configuration 2*:



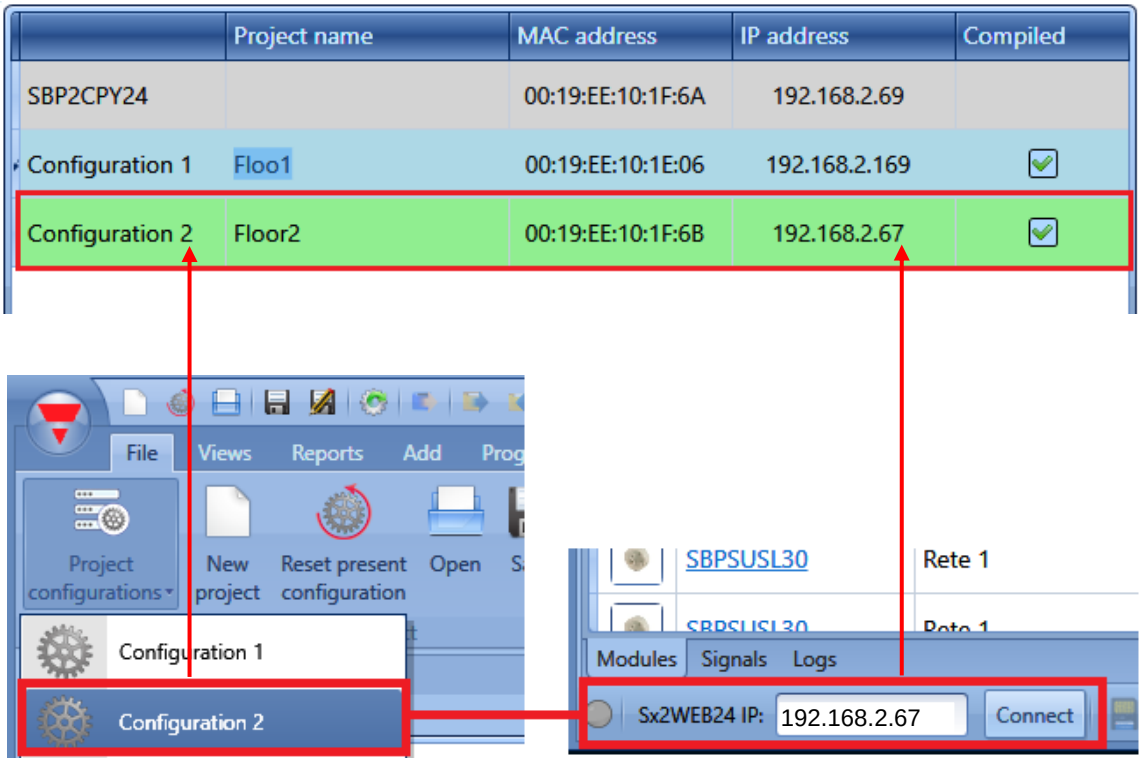
   d. The user has to select the IP address of the controller related to the new configuration.

   e. When the configuration has been done, the user can send the configuration to the controller by clicking on the icon marked in red in the picture below:



2) Repeat the operation from *1.a* to *1.e* for all the controllers/configurations in the project. Should the installation have only one controller, this operation has to be carried out only once.

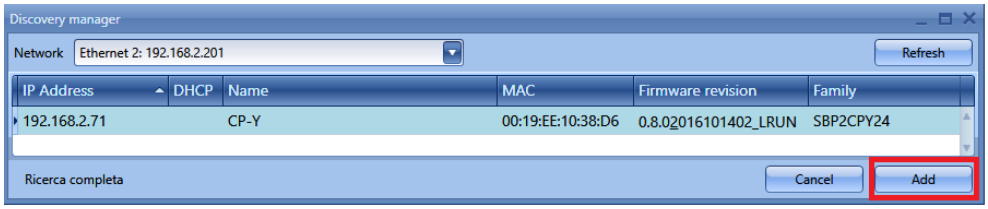
Should the CPY server be into the UWP 3.0, the address to access it is: *controller_ipaddress*/CP3App.



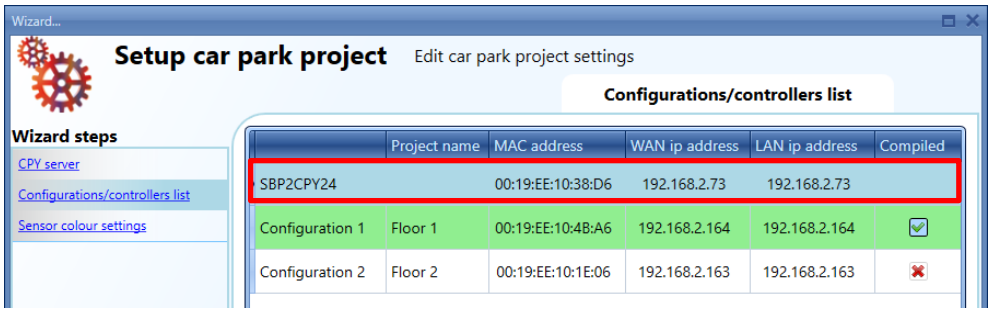
9.2 SBP2CPY24

1) To find the connected SBP2CPY24, click on the icon marked in red to start the discovery:

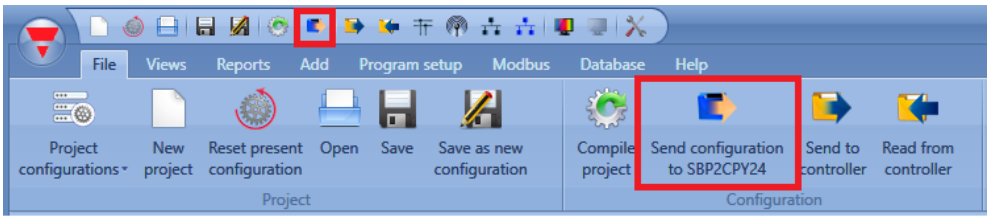
The following window will appear with all the SBP2CPY24s in the network:



Select the one to be added to the Car Park project and click on *Add*: the SBP2CPY24 will be added to the project and it will be listed in the window showing all the devices included in the project.



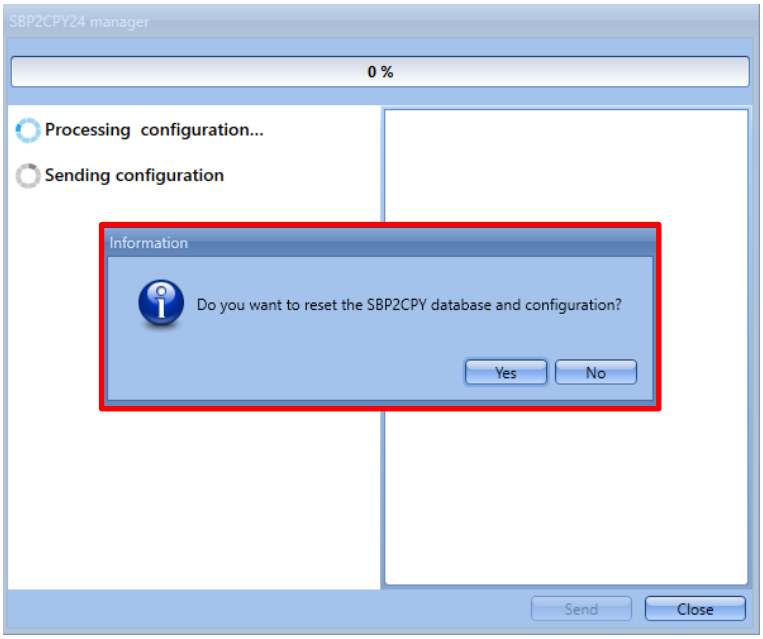
2) Click on the icon marked in red to download the project into the SBP2CPY24



Each time a change is made in the configuration, the user has to click the icon marked in red to send the configuration of the project to the SBP2CPY24.

9.2.1 How to reset the configuration into the CPY server

If a new configuration has to be done, clicking on *Send configuration to SBP2CPY24* button the following windows is shown:



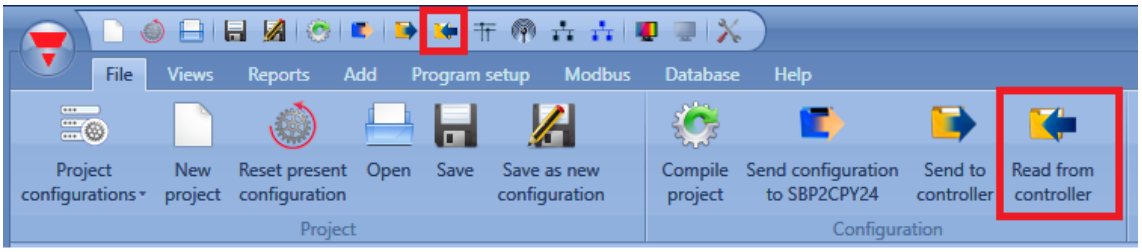
The user has to click on *Yes* to reset the database of the SBP2CPY24 in order to align the CPY project with the Car Park project.

If the user clicks on *No* button, the last valid CPY project is the previous database status will be kept on the SBP2CPY24 but this should be unusable.

Note: It is strictly suggested in order to keep aligned the CPY project and Car Park project, to reset the database on the CPY server every time a new configuration is made.

10 How to read the configuration from a controller

To read the configuration from a controller, connect to it and click on the icon marked in red as shown below:

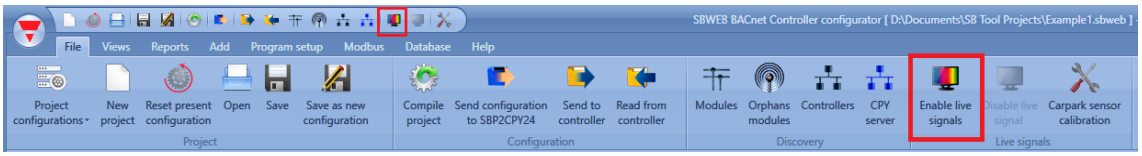


In the case of multi-configuration projects, this operation has to be done only once, connecting the UWP 3.0 Tool to any of the controllers belonging to the installation.

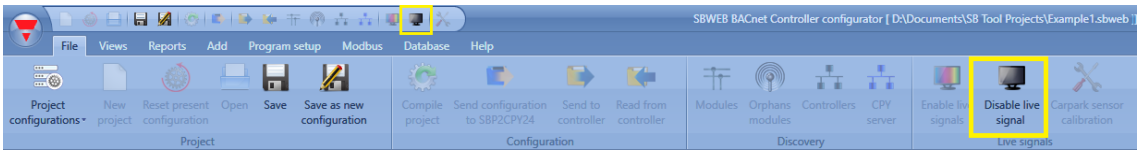
11 Live signals

If *Live signals* is enabled, it is possible to check the status of each sensor (vacant/engaged), to see the status and voltage of the bus and to read the value of each signal.

To enable it, click on the icon marked in red:

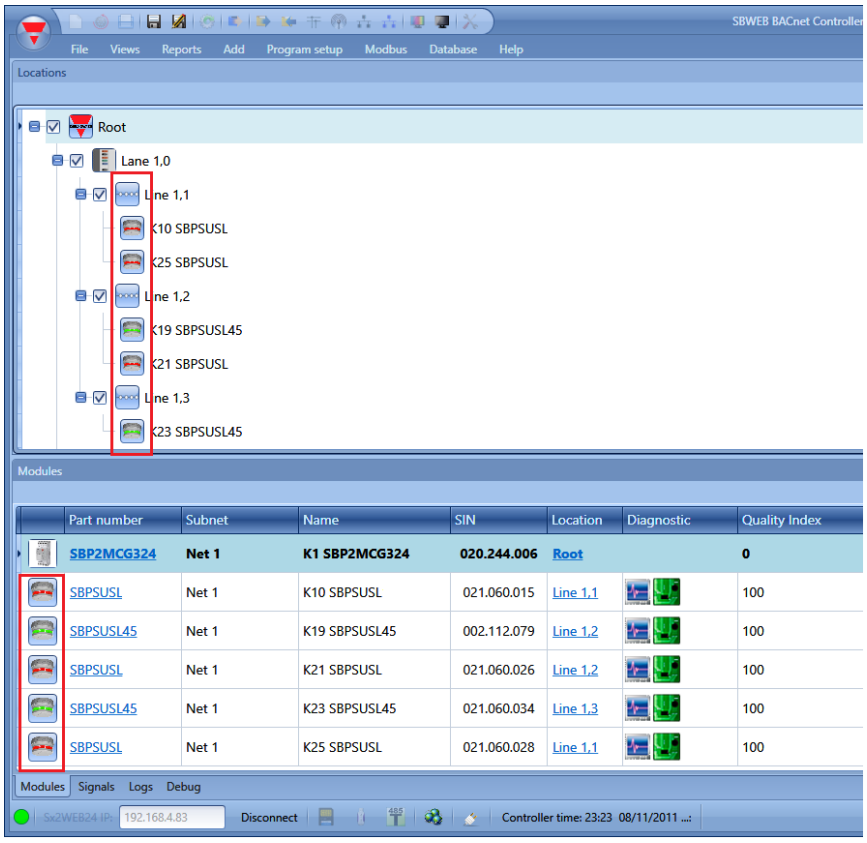


To disable it, click on the icon marked in yellow:



Once Live signals is enabled, it is possible to see the status (vacant/engaged) of each sensor, as shown below:

If the sensor is shown with the red LED, the parking bay is engaged, while if the LED is green, the parking bay is vacant.



For more information about the *Live signals* feature, please refer to the system manual at this link: http://www.productselection.net/MANUALS/UK/uwp3.0_tool.pdf

12 Zone counter function

The Master Zone Counter (MZC) is a zone count system which has the ability to detect and count cars when they enter and exit zones in the Carpark facility and to send the information to the displays and to the SBP2CPY24 server.
The count system consists of a number of count zones, each of which has a certain number of entry and exit points for the cars. These are called detection points (DPOs) and this is where the sensors must be mounted to detect passing cars.

12.1 What is a zone?

A zone is typically a level of the parking facility, but can also be a part of a level or even the entire Carpark. A zone has a certain number of parking bays available, and the aim of the zone count system is to detect and count the cars entering and leaving the zone, thereby keeping track of the number of available spaces. This means that, once the maximum number of available spaces is set, the Master Zone Counter (MZC) function will deduct from it every time a car enters the zone, and will add to it whenever a car exits from the zone.

12.2 Detection Points (DPOs)

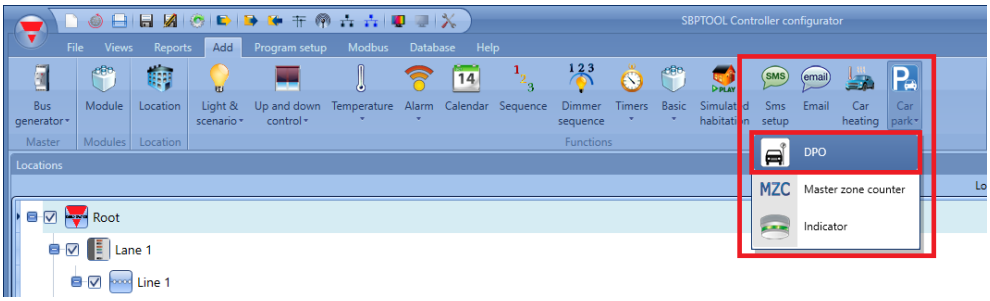
A detection point is a lane or driveway where cars enter or leave a zone. A typical example of a DPO is a ramp between two levels, but it could also be the entry point from the street into the Carpark, or the exit point. In many cases, a detection point is involved in two zones. For example, a DPO which is an exit point for level 2 could at the same time be an entry point for level 3.
Each detection point needs sensors to detect the passing cars. Dupline® SBPSUSCNT sensors are usually used, but other types of sensor, such as standard photo-electric or loop detectors, can also be used. This is carried out by connecting the sensor output to a Dupline® input module.
The zone counting gives the option of using either one or two sensors in each DPO. Two sensors with a distance of 2-3 m between them are recommended because this offers the possibility of detecting the direction of the car and allows more efficient filtering to avoid any false detection. Sometimes cars drive in the wrong direction in a one-way lane, and in a two-sensor solution the MZC is able to manage this so that the count will still be correct. In two-directional lanes it is mandatory to use two sensors. When configuring detection points, a timeout value has to be defined: it allows valid car detection as long as the delay period is less than the timeout value from the point where sensor 1 becomes inactive until the point where sensor 2 becomes active. With a typical distance of 2-3 m between the sensors, 1 sec is the recommended value. Too high a value increases the risk of detection faults. Single sensor DPOs are mainly used when it is not possible or difficult to use two sensors, for example in an outdoor installation with loop detectors.

12.3 Initialization and adjustment

In the initial configuration, the installer has to define the number of spaces in each zone. The actual number of available spaces in each zone at the initial stage must also be defined. From that point, the Master Zone Counter function will increase or decrease the zone count values as the cars enter or leave the zones through the relevant detection points. Since any count system runs the risk of accumulating detection faults, it is important to have a manual count adjustment facility that can be used from time to time whenever required. In the Dupline® Carpark count system, this manual adjustment is carried out via the SBP2CPY230 webserver which can be accessed from a pc or a laptop or via the UWP 3.0 Tool by means of the *Live signals*.

12.4 Detection points (DPO) function

For each entry/exit point, a DPO function has to be created. Select the DPO function from the *Add* menu.

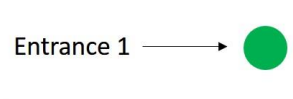


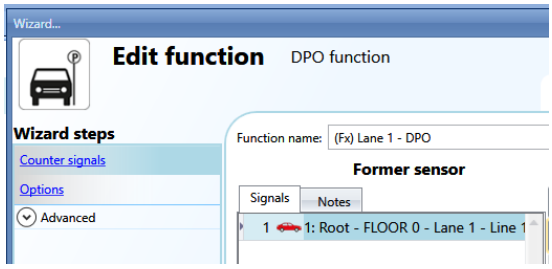
The following window will appear:

12.4.1 Entrance/exit with no direction detection

In this situation, one or more sensors have to be added in the field *Counter signals/ Former sensor*. The DPO function counts without sign and can be used to monitor one direction only (Entrance or Exit). The counter value is increased every time the sensor is activated.
Up to 10 sensors can be added.

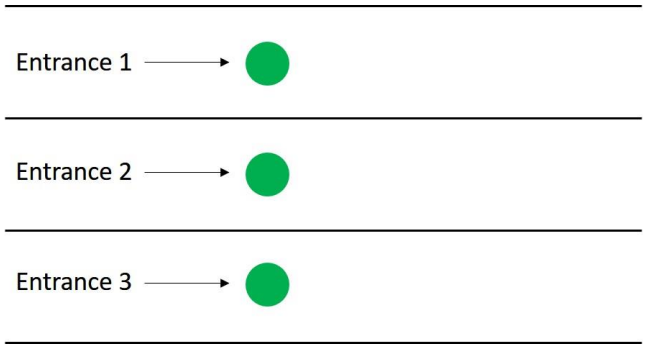
Example of a detection point with one sensor:



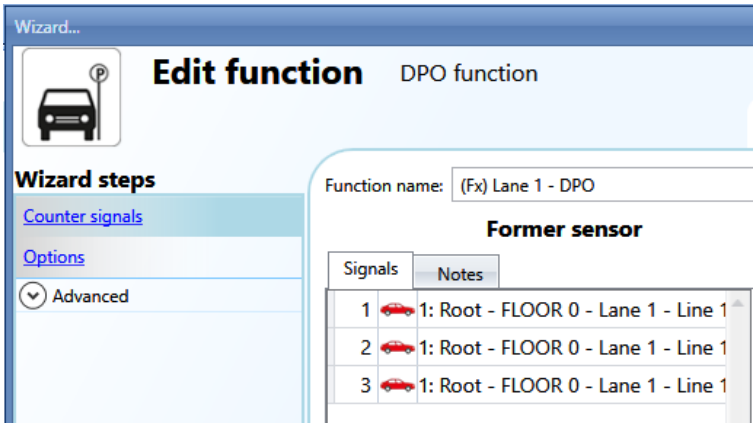


Only one sensor has to be added in the *Former sensor* field:

Example of a detection point with many sensors (one large entrance):



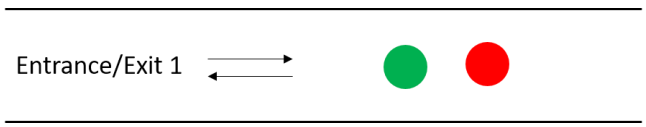
Three sensors have to be added in the *Former sensor* field.



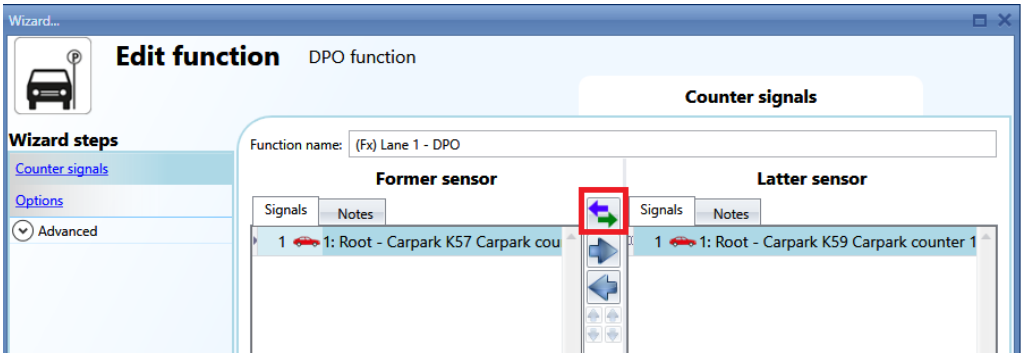
12.4.2 Entrance/exit with direction detection

In this situation two sensors are used to detect the direction of the car: the first sensor to be activated has to be added in the *Former sensor* field, while the second sensor has to be added in the *Latter sensor* field. When the first sensor is activated first, the counter of the function is increased, while when the second sensor is activated first, the counter is decreased.

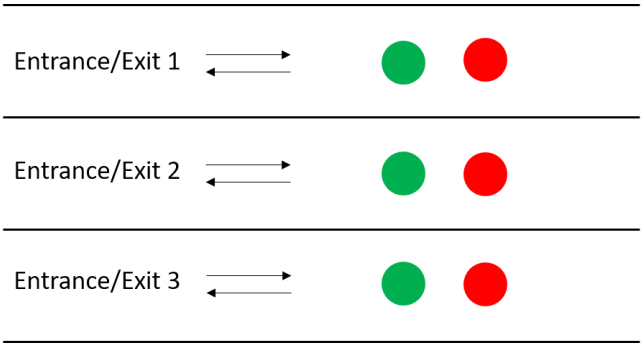
Example of a detection point with one Entrance/Exit:



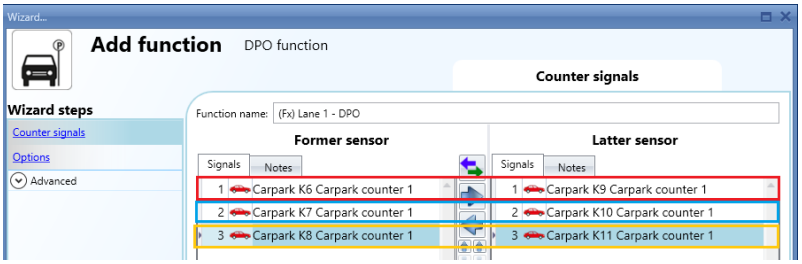
Sensor K57 will be the first to be detected if a car is entering the zone, while K59 will be the second. They can be swapped at any time by clicking on the icon marked in red.



Example of a detection point with many Entrances/Exits (one large entrance):



In this example, six sensors have to be added in the function:



The sensors are coupled according to their position in the list: K6-K9, K7-K10, K8-K11.

The position of the sensors can be changed at any time:

Move the sensor to the right column

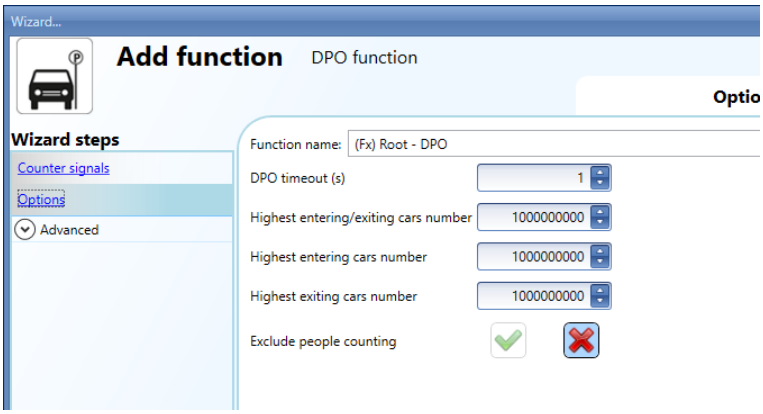
Move the sensor to the left column

Move the sensor up in the list

Move the sensor down in the list

12.4.3 How to set the Options field of the counter in the DPO function

By clicking on the Options field of the DPO function, the following window will appear:



DPO timeout (sec): This is the time in which the second sensor has to be activated after the first sensor. It can be set from 1 to 10 seconds.

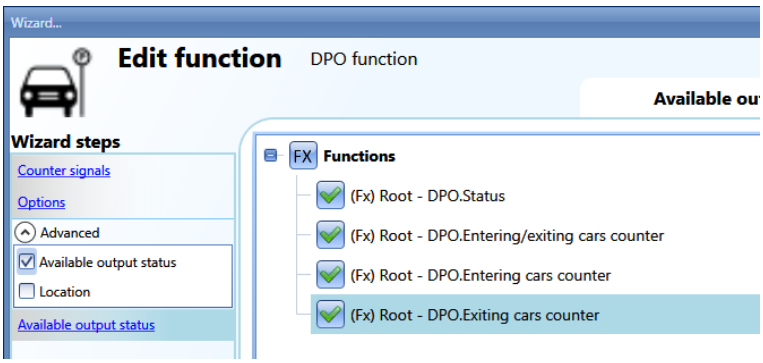
Highest entering/exiting cars number: If the detection point has two sensors to recognize the car's direction, this is the maximum value the counter can reach. This is a number with sign: it is increased if the car is entering the zone (the sensor in the field *Former sensor* is detected first), and decreased if the car is exiting (the sensor in the field *Latter sensor* is detected first). It can be set from -2.000.000.000 to +2.000.000.000.

Highest entering cars number: If the detection point has only one sensor, this is the maximum value the counter can reach. It can be set from 0 to +2.000.000.000. This is increased if the DPO function is linked to an *entry* in the MZC function.

Highest exiting cars number: If the detection point has only one sensor, this is the maximum value the counter can reach. It can be set from 0 to +2.000.000.000. This is increased if the DPO function is linked to an *exit* in the MZC function.

Exclude people counting: if the green V icon is selected, the people filter will be enabled in order to avoid the people detection

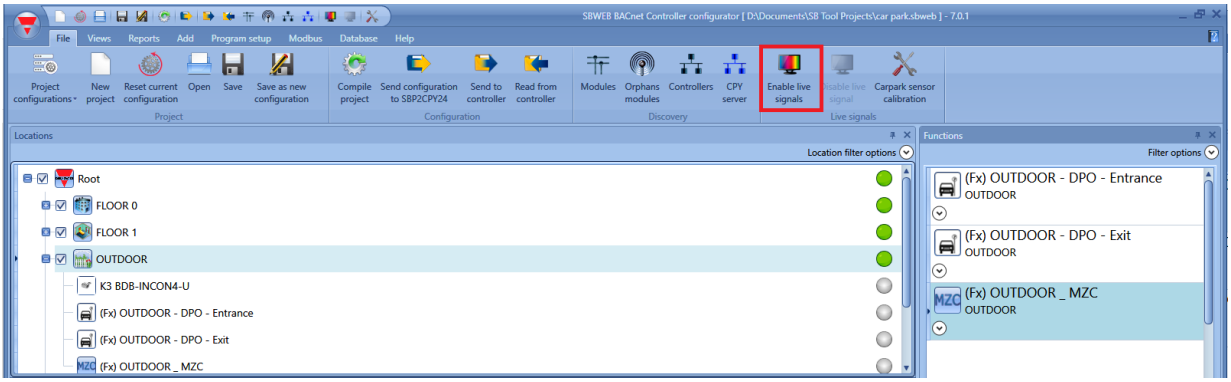
These parameters can also be read and written via Live signals, Modbus and BACnet, by enabling them in the *Advanced* fields of the DPO function.
They cannot be logged in the UWP 3.0 database, since this is done by the MZC function.

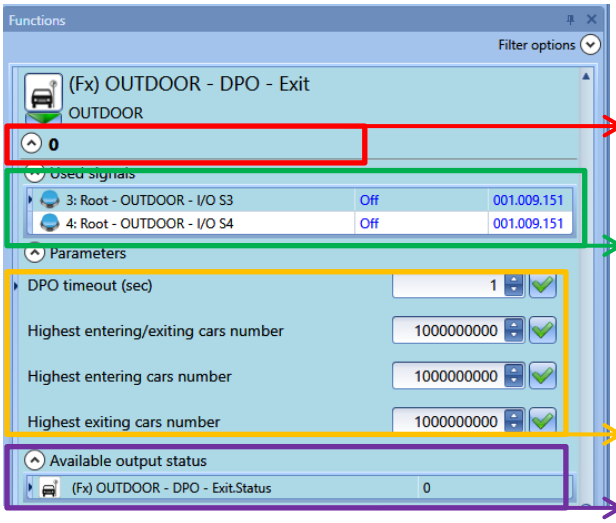


The DPO.status is set to 1 (or -1 if two sensors are used) only for the short time a sensor is activated (the car is passing under it).

12.4.4 Live signals for the DPO function

The counters of the DPO function can be set/reset using live signals. To enable this feature, click on the icon marked in red, after having downloaded the configuration with a DPO function into the UWP 3.0.



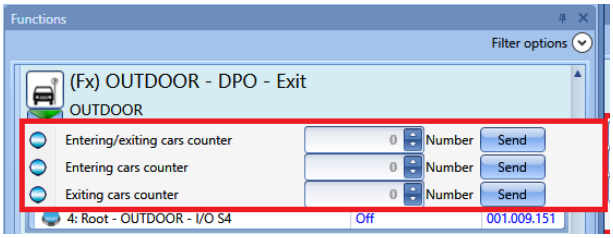


This is the status of the function: it turns to "1" only when the sensors linked to the function are activated by a car (i.e. for a very short time)

The status of the sensors linked to the MZC function are shown: if a car is activating a DPO function, a "1" is shown for the time the car is under the sensor (very short time, less than 1 second).

The parameters of the function can also be changed in live signals without needing to write the configuration again.

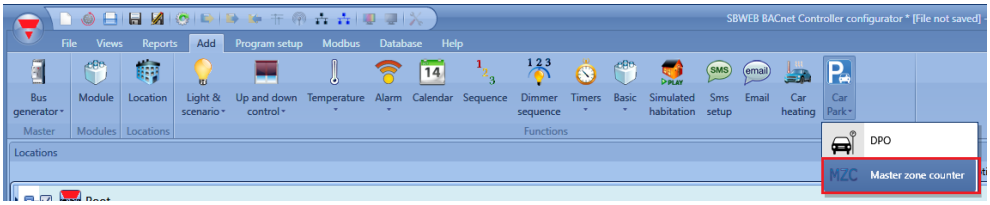
The three counters and the status are shown here for debug purposes.

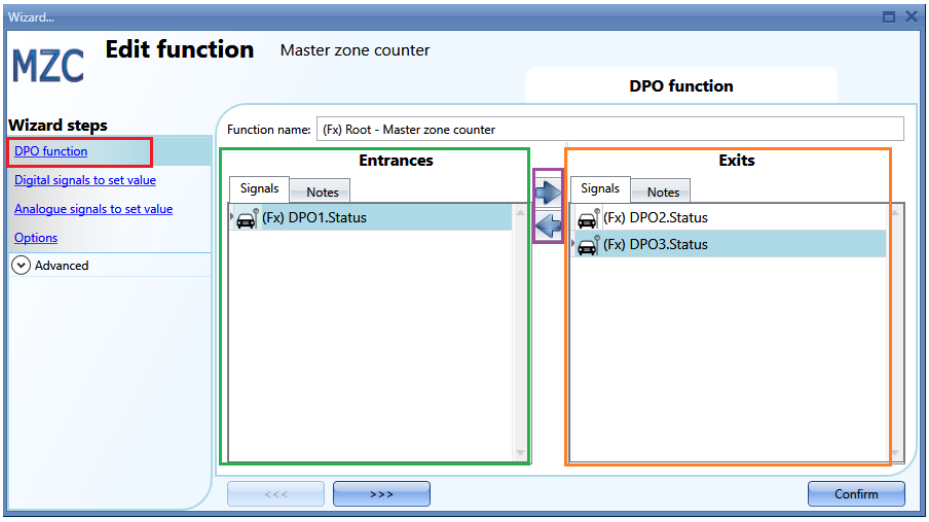


The counters of the functions can be set or reset in live signals by means of the menu marked in red that appears by clicking on the green arrow.

12.5 MZC function

The Master Zone Counter function counts down the available spaces in a zone.
The counter decreases in number when the sensors of a DPO set as an entry detects a car, while it increases when the sensors of a DPO set as an exit are activated by a car.



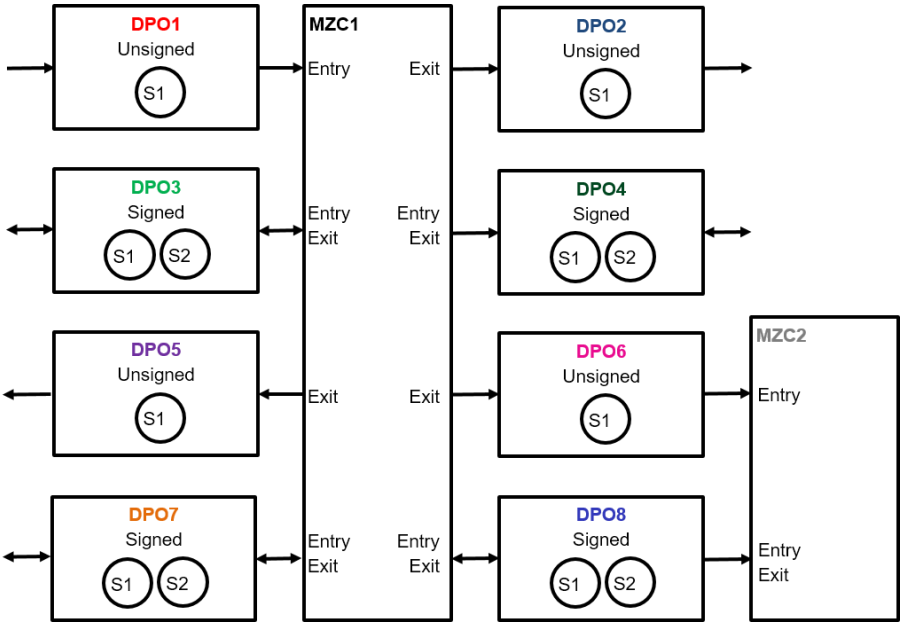


The first things to configure are the *Entrances* and the *Exits* of the zone the function is counting.
In the *Entrances* field (marked in green), the DPO functions that represent the entry for the zone have to be added by double clicking the *Signals* area.
In the Exits field (marked in orange), the DPO functions that represent the exit for the zone have to be added by double clicking the *Signals* area.

The following rules have to be followed:

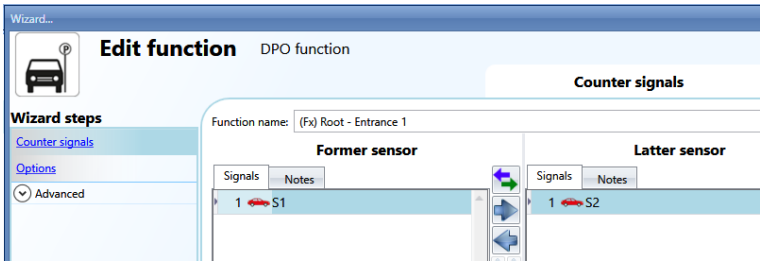
1) Each MZC function can manage up to 50 DPO functions in each direction
2) Any DPO added into the *Entrance* field will decrement the counter (the number of available bays decreases if cars enter the zone)
3) Any DPO added into the *Exit* field will increment the counter (the number of available bays increases if cars exit the zone)
4) Two MZC functions can share the same DPO function, but under the condition that the DPO function must be used as an entrance in the first MZC function and as an exit in the second MZC function.
5) When the position of a DPO function is changed, for example from entrance to exit (using the arrows marked in purple), if it is used also in a second MZC function, its position is automatically changed from exit to entrance in the second MZC function.

The following diagram shows the different possible combinations:

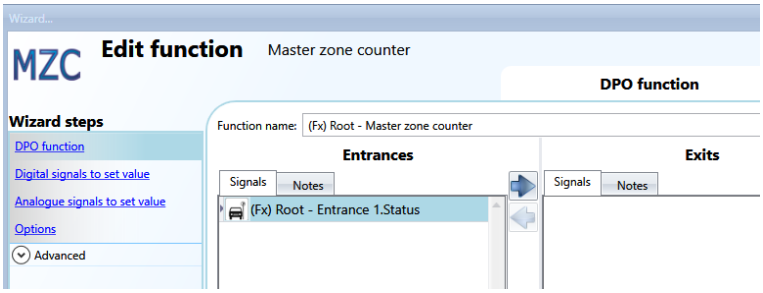


If a DPO function with two sensors is used (S1 as former sensor and S2 as latter sensor), it has to be added only once into a MZC function: see example below.

DPO function with two sensors to recognise the direction of the car:

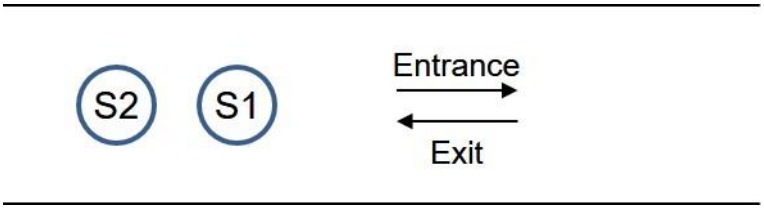


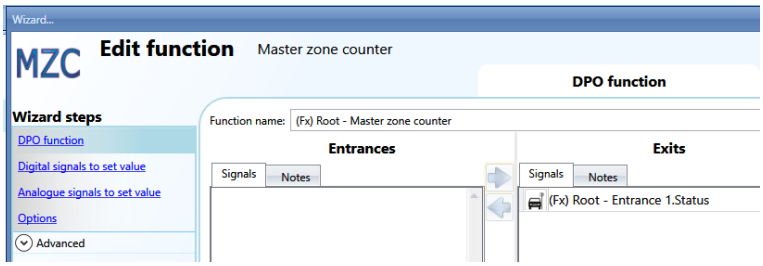
Example 1
When the car is moving from S1 to S2, it is entering the zone:



If the DPO function is added as an *Entrance,* the counter will be affected in this way: when the car is moving from S1 to S2, it is entering the zone and the counter is decreased; if the car is moving from S2 to S1, it is exiting the zone and the counter is increased.

Example 2
When the car is moving from S1 to S2, it is exiting the zone:

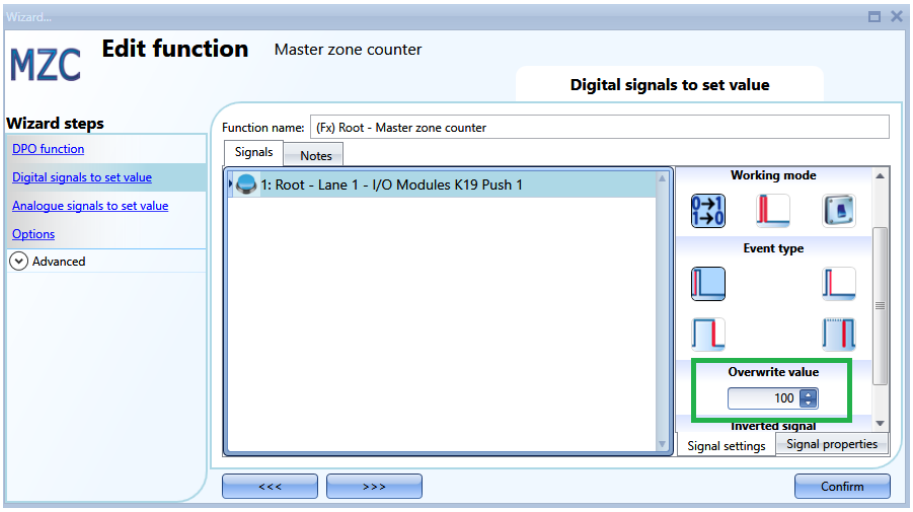




If the function is added as an *Exit,* the counter will be affected in this way: when the car is moving from S1 to S2, it is exiting the zone and the counter is increased; if the car is moving from S2 to S1, it is entering the zone and the counter is decreased.

12.5.1 How to set a predefined value of the counter using signals

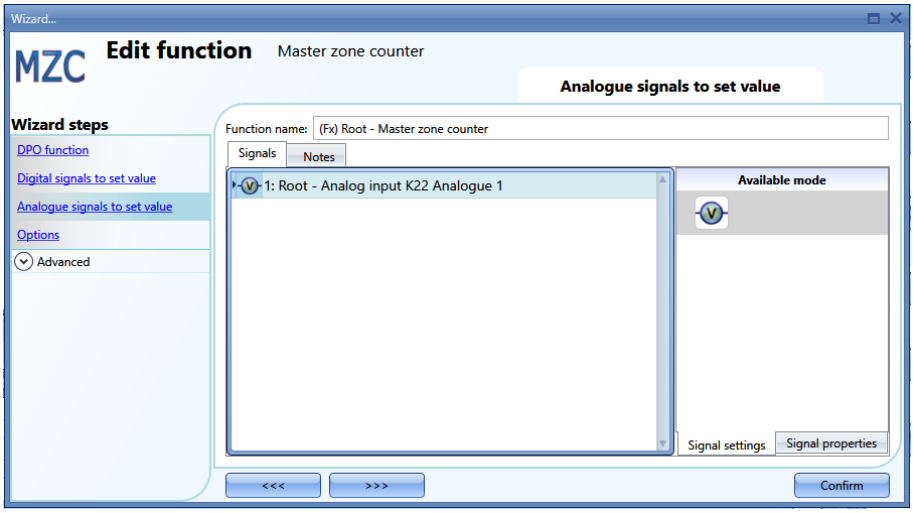
The counter of the available bays can be set to a predefined value using any digital or analogue signals. If the counter has to be set with a digital signal such as a push button, the field *Digital signals to set value* has to be used:



The event type then has to be selected according to the standard rules of the UWP 3.0 Tool (short/long press, falling/rising edge), and the value (the empty bays) the MZC is to have once the signal is activated must be filled in as shown in the green rectangle.
Every time the number of empty bays is changed, the *Car in transit* and *Surplus of available bays* values

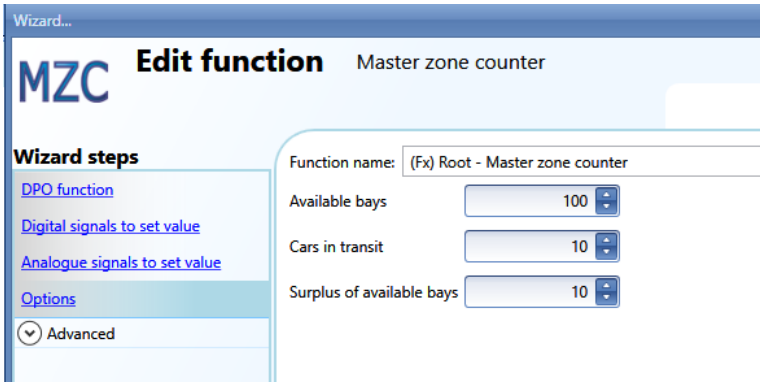
are also reset to the settings in the *Option* field.
The *Overwrite* value can be defined for each signal, and up to 10 signals can be added.

If the user needs to set the MZC function equal to another function or to an analogue signal, *Analogue signals to set value* has to be selected.



When the signal changes, the new value is written in the MZC function (i.e. the available empty bays). Also the *Car in transit* and *Surplus of available bays* values are reset to the settings in the *Option* field. Up to 20 signals can be added.

In the Options field the following parameters have to be set:



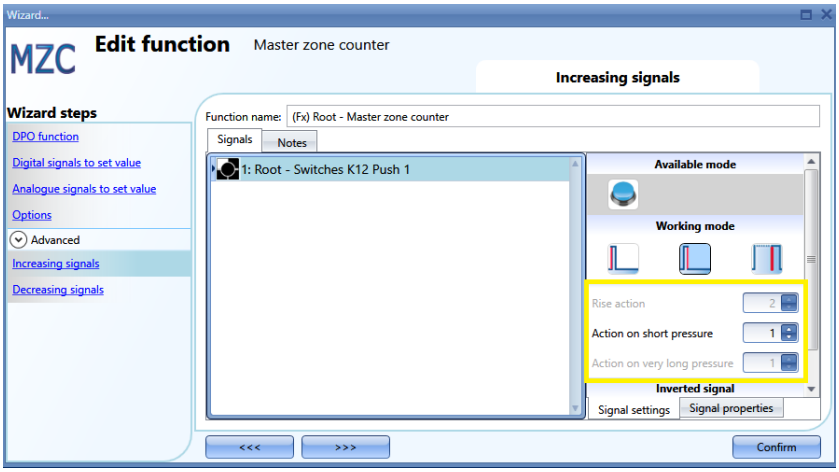
Available bays: this is the value from which the function starts to count down. It is the number of available bays in the zone. The counter is decremented when a car enters the zone until it reaches zero and then it is incremented each time a car exits the zone.

Cars in transit: when all the bays of the zone are occupied, there are cars going around to look for a space. This parameter takes this number into account: it is not mandatory and the user can decide whether or not to use it. The counter relevant to this parameter will be incremented once the counter for the available bays reaches zero. It will be decremented each time a car exits a zone.
Example: Available bays=100, Cars in transit=10
Once the *Available_bays* counter arrives at zero and other cars are entering the zone, the *Cars_in_transit* counter is decremented. It is then incremented when the cars exit the zone. Only when this counter reaches the predefined value (10 in this example), will the *Available_bays* counter be incremented again.

Surplus of available bays: This can be used for debug purposes, to check the number of cars exiting the zone.

12.5.2 How to manually increase/decrease the counter

To increase/decrease the counter of the function, digital signals can be used.
In the *Advanced* section, enable the field *Increasing signals* and/or *Decreasing signals*, and select the signals.

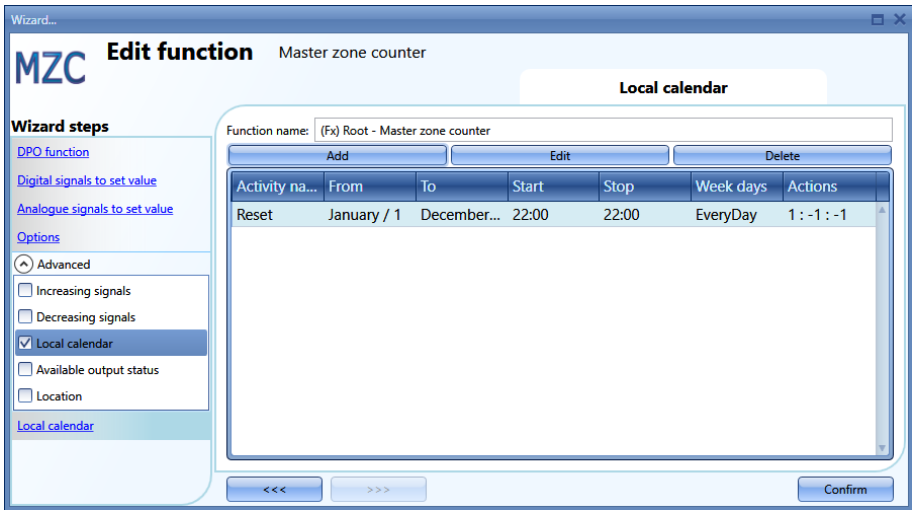


If a pushbutton is chosen, different types of actions are available that will make the counter increase: rising edge (i.e. as soon as the pushbutton is pressed), falling edge (i.e. as soon as the pushbutton is released), very long pressure (i.e. when the pushbutton is kept pressed for the set seconds). For each of these actions, the increasing value can be set (yellow rectangle).

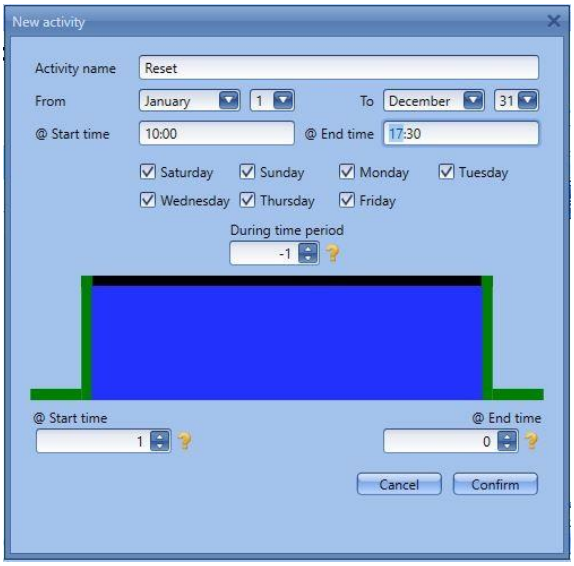
If a level signal is selected, such as a switch or a function, the available actions are rising edge (i.e. when the switch is closed or the function is activated), falling edge (i.e. when the switch is open or the function is deactivated), or both.

12.5.3 How to set the number of available bays with the calendar

The counter value can be changed/reset at a certain time of the day using the calendar. In the *Advanced* field, enable *Local calendar.*



Click on *Add* to open a new activity window:



Activity name: In this field the user defines the name of the event that will appear on the calendar. This is a mandatory field.

From*:* the start date for the calendar activity.

To*:* the end date of the calendar activity.

@ ***Start time****:* The time of the start of the activity.

@ ***Stop time***: The time of the end of the activity.

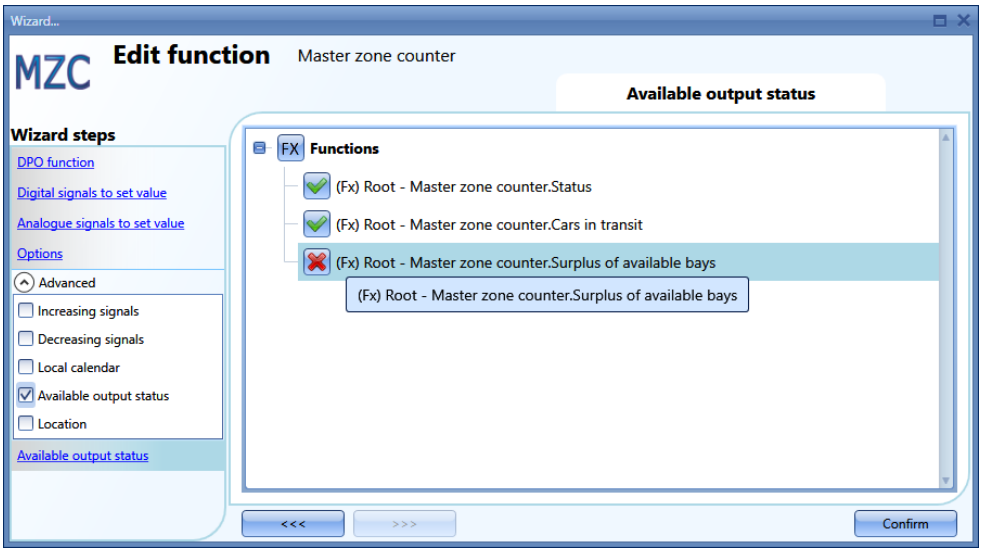
Days: The user should select the days when the calendar activities must be applied.

@ ***Start time***: in this field the user can select the value that will be written in the counter of the function
- No action (-1)
- MZC=0 (0)
- MZC=parameters *Available bays* (1)
- MZC= 2 (2)
- MZC= 3 (3)
- ……
- MZC=10000 (10000)

@ ***End time***: in this field the user can select the value that will be written in the counter of the function
- No action (-1)
- MZC=0 (0)
- MZC=parameters *Available bays* (1)
- MZC= 2 (2)
- MZC= 3 (3)
- ……
- MZC=10000 (10000)

12.5.4 How to remotely access the output status of the MZC function

In the *Advanced* section, enable *Available output status*:



The green V must be present if the different counters have to be used in live signal, Modbus or webserver.

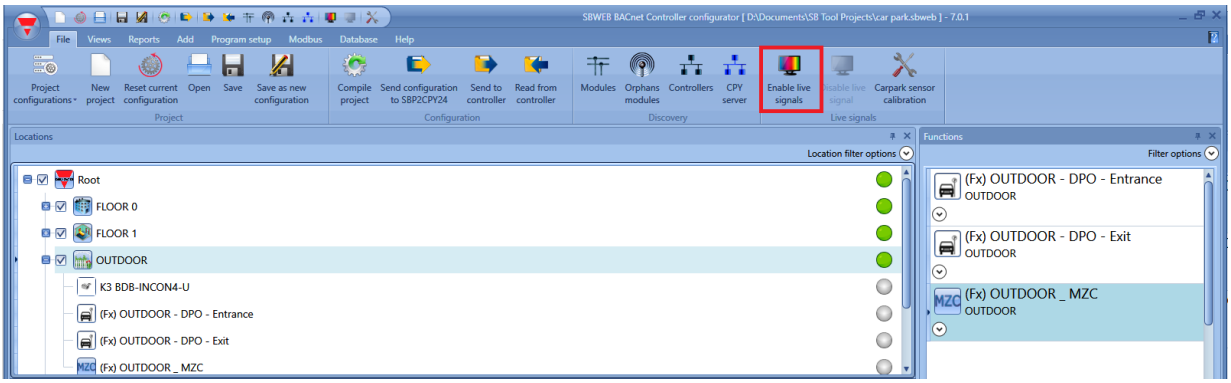
Master zone counter.status: This is the counter of the available bays and it indicates how many bays are empty now

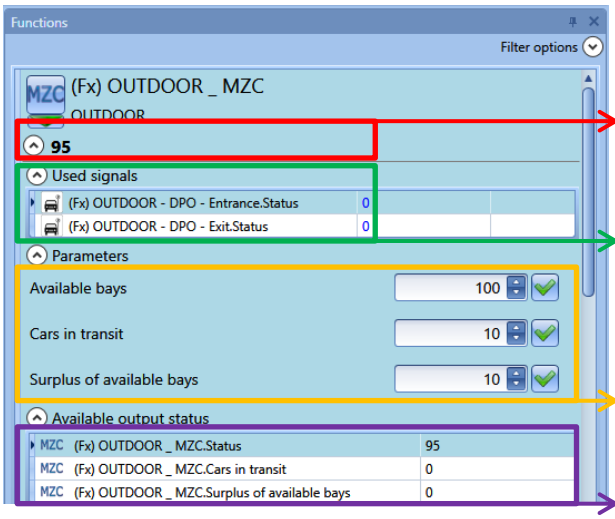
Master zone counter.Cars in transit: This is the counter of the cars in transit and it indicates how many cars are driving through the zone when all the bays are occupied

Master zone counter.Surplus of available bays: This can be used for debug purposes, to check the numbers of cars exiting the zone.

12.5.5 Live signals in the MZC function

The counter of the MZC function can be set/reset using live signals. To enable this feature, click on the icon marked in red, after having downloaded the configuration with a MZC function into the UWP 3.0.



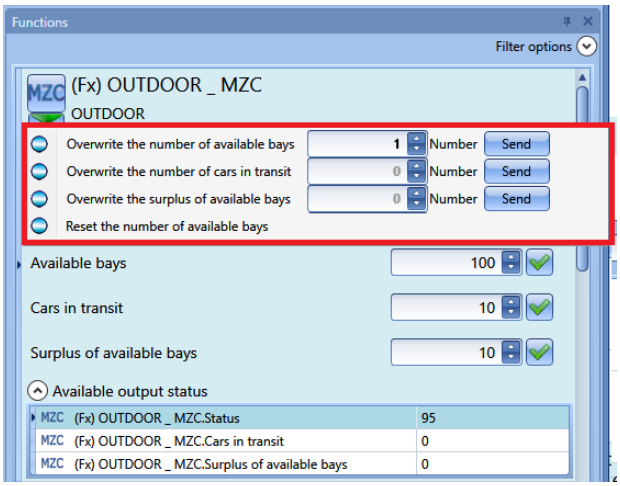


This is the counter that indicates the number of available bays: it is decreased when a car enters the zone and increased when a car leaves the zone.

The status of the DPO linked to the MZC function are shown: if a car is activating a DPO function, a "1" is shown for the time the car is under the sensor (very short time, less than 1 second.

The parameters of the function can also be changed in live signals without needing to write the configuration again.

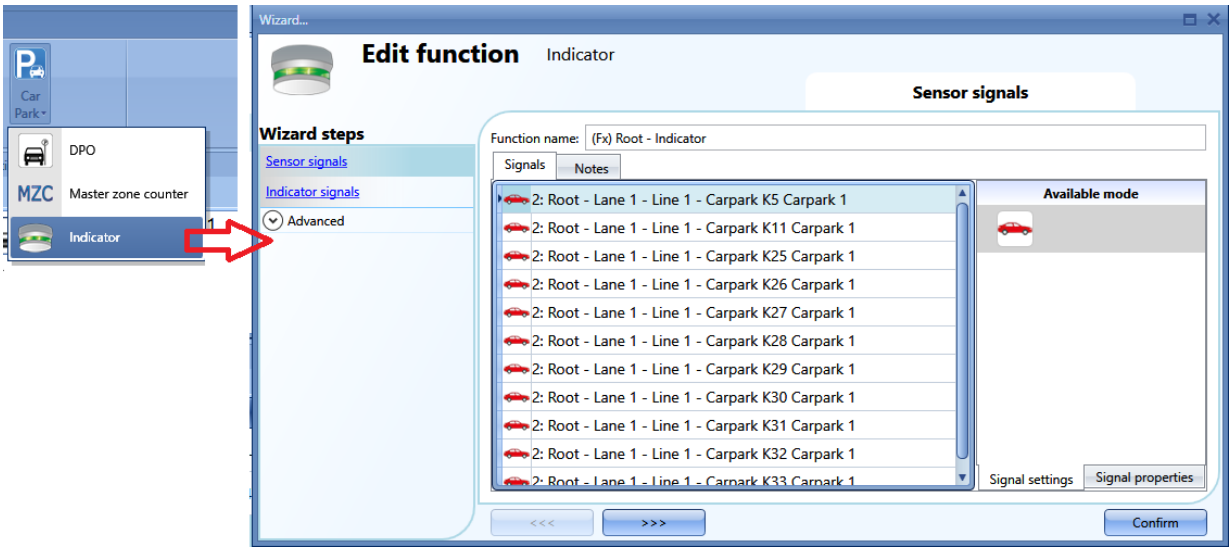
The three counters are shown here for debug purposes.



The counters of the functions can be set or reset in live signals by means of the menu marked in red which appears by clicking on the green arrow.

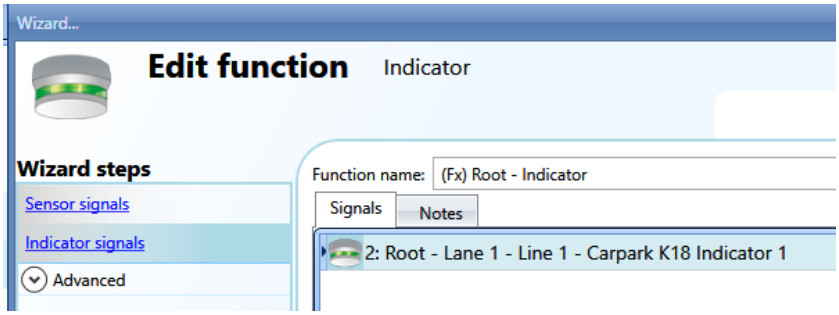
13 Indicator function

Should the user need an indicator to show if there are empty bays in the lane, the indicator function can be used to make the LED of the SBPILED change colour according to the status of the sensors in the lane.

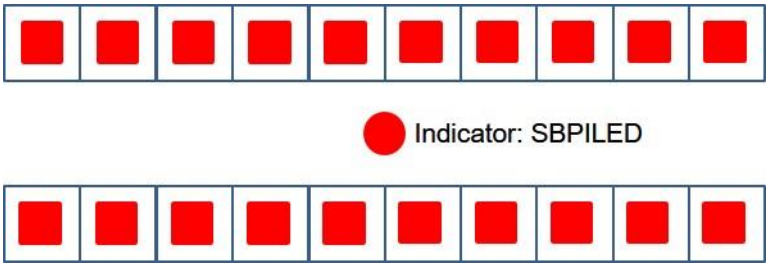


In the *Sensor signals* field, the user has to add the sensors which will give the indication about the lane or line: the system will do an OR of all the sensors added in this field.

In the *Indicator signals* field, the user has to add the indicator modules SBPILED which are mounted in the lane, to give a fast indication about the availability of the bays.

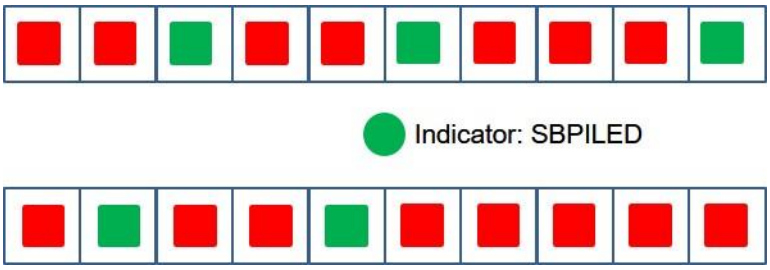


Example 1: lane with all the bays occupied



The red squares are the sensors that have to be linked into the *Sensor signals* field and which detect cars under them. The red circle is the indicator that has to be added into the *Indicator signals* field. Since there are no available spaces, the indicator is showing a red light to advise the driver not to enter the lane, since all the bays are occupied.

Example 2: lane with some available bays

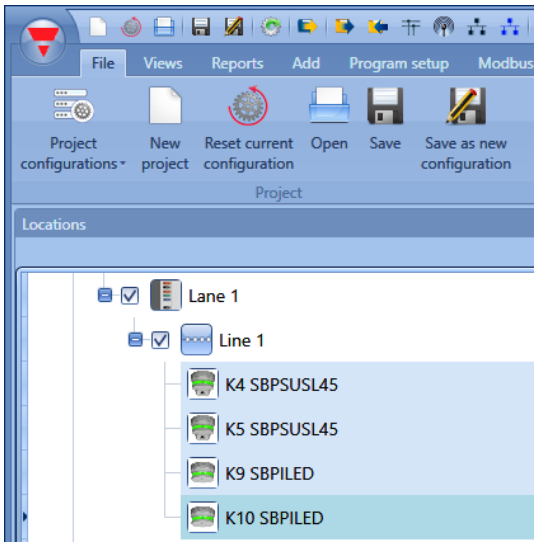


The red and green squares are the sensors that have to be linked into the *Sensor signals* field (red are the occupied ones, green the empty ones). The green circle is the indicator that has to be added into the *Indicator signals* field. Since there are some available spaces, the indicator is showing a green light to advise the driver that there are some empty bays in the lane.

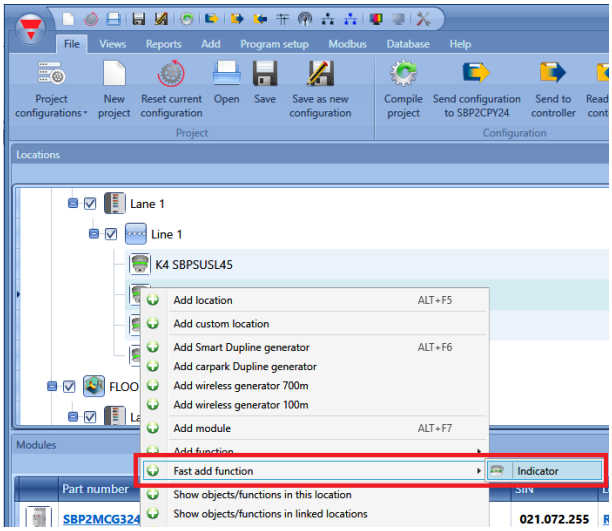
1) The function is automatically generated by placing the indicator modules SBPILED in the *Indicator signals* field, and the SBPUSLxx in the *Sensor signals* field.

13.1 How to add the Indicator function with the fast procedure

This function can also be added in a fast way in the *Location* window by following these steps:

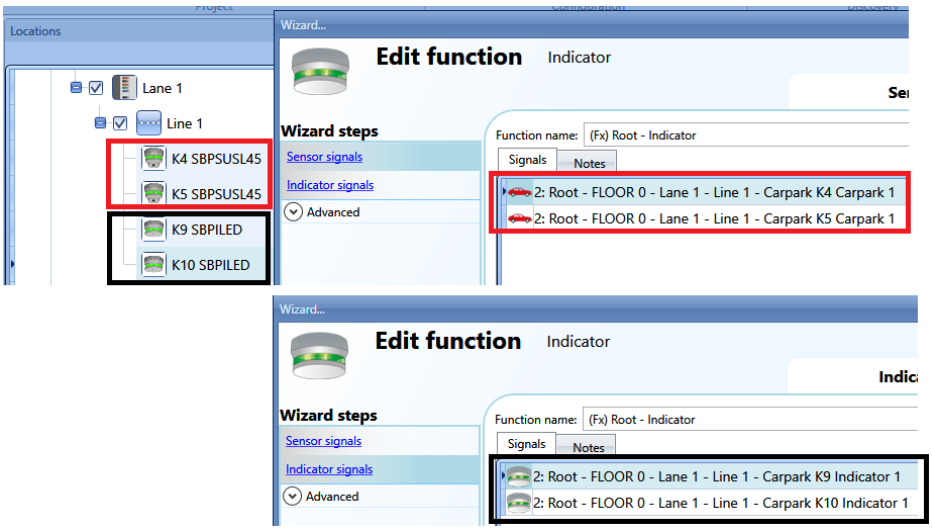


1) Select the indicator modules and the sensors in the location where the indicator function has to be placed: a multiple selection can be carried out by clicking on the module while pressing the *Shift or CTRL key.*
If also the location is selected, the function will be placed here, otherwise it will be placed in the location *Root*.



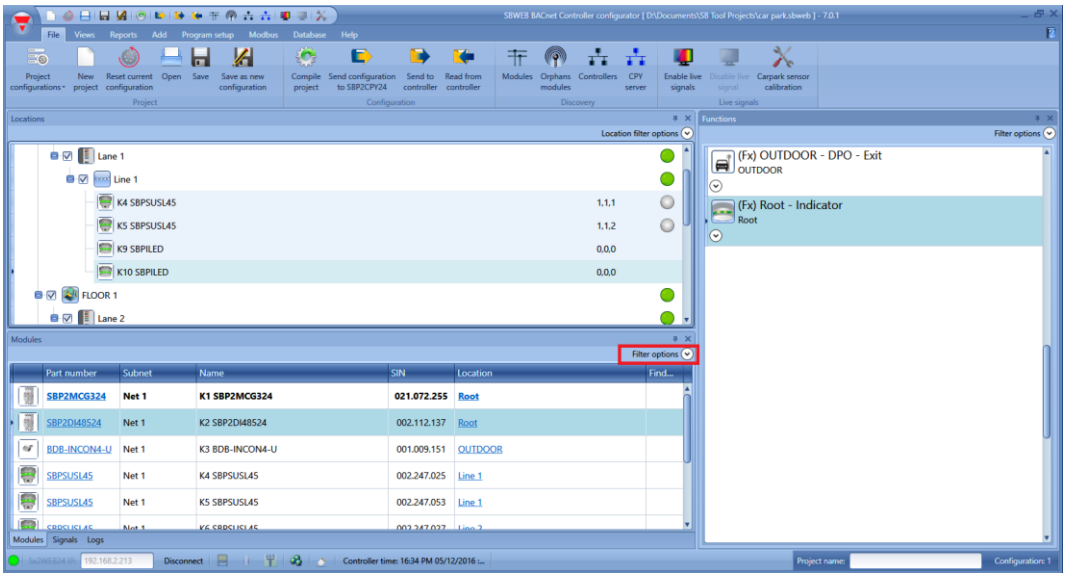
2) Right click on the selected modules and select *Indicator* from the *Fast add function* menu.

3) The function will be automatically generated by placing the indicator modules SBPILED in the *Indicator signals* field, and the SBPUSLxx in the *Sensor signals* field.



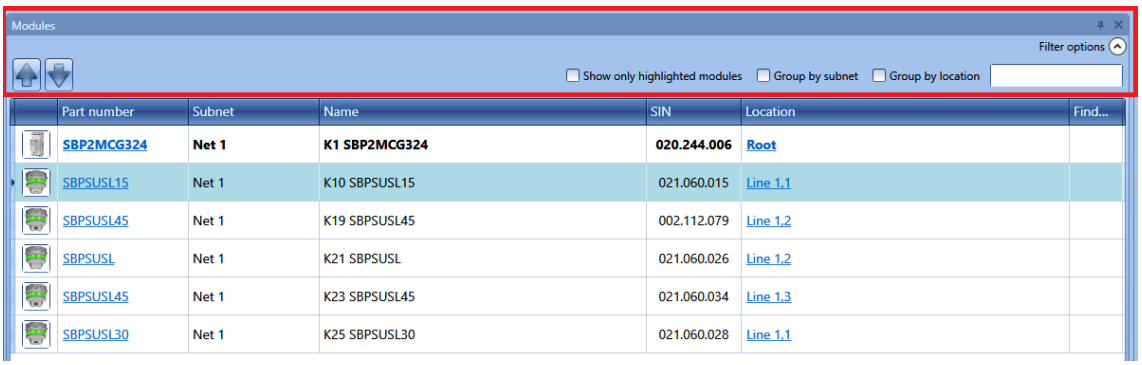
14 Modules

To configure a module, the user has to click on the picture of the module in the *Modules* window of the UWP 3.0 Tool (see below):



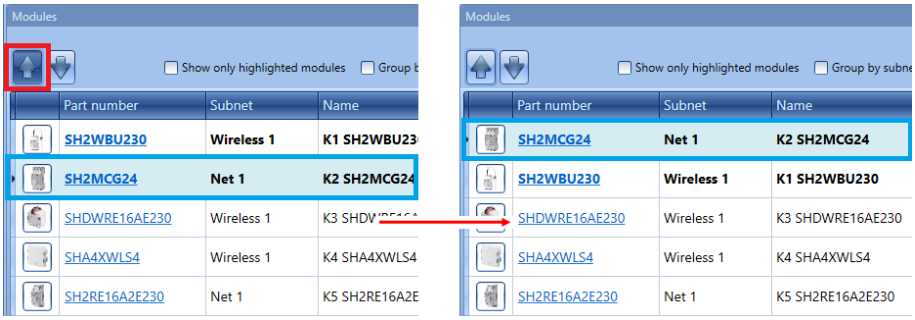
14.1 How to manage the filters on the Modules window

The modules can be sorted or filtered by the *Filter options*: by clicking on the icon, the *Filter options* panel will be opened. The available filters are shown in the top of the *Modules* window, as shown in the red rectangle:

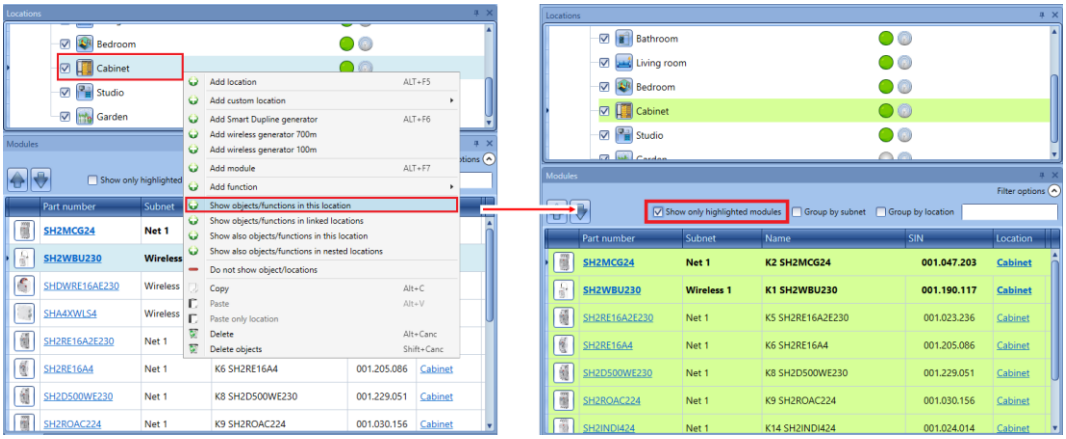


Only the modules specified by the filter will be presented: the filters can be used individually or they can be mixed.

Up/Down arrows:
Once a module is selected, by clicking on the arrows it can be moved up or down.

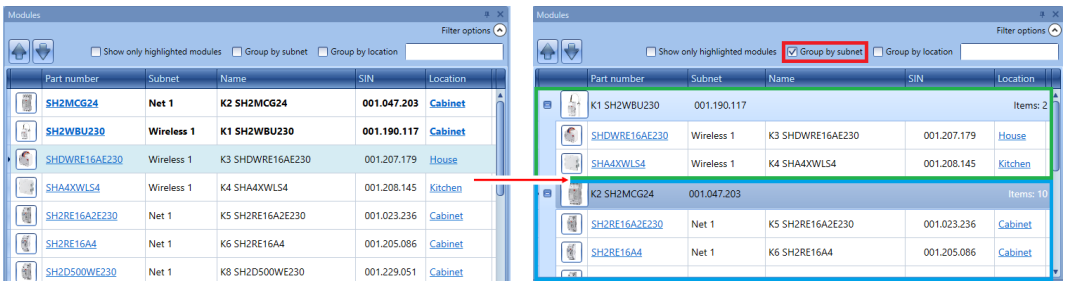


Show only highlighted modules:
When this filter option is checked, only the modules that are highlighted (according to the *highlight* functionality available on the *Locations/Functions* window) are presented in the *Modules* window.
In the picture below an example is shown:

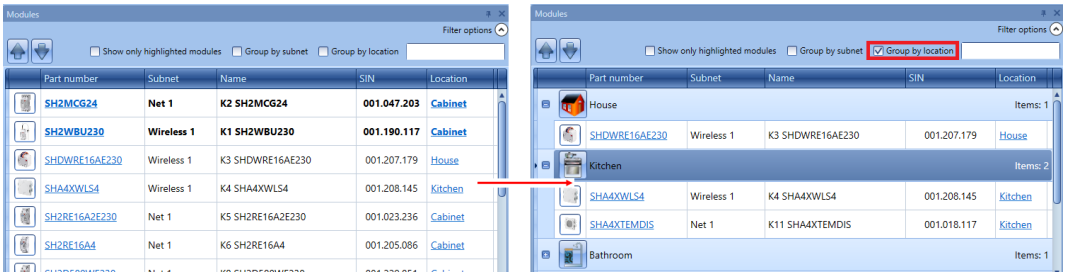


Group by subnet:
The modules are grouped by:

- Master generator (each network generated by SH2MCG24, SH2DUG24 and SH2WBU230N);
- Modbus serial communication port (COM1MASTER or COM2MASTER);
- TCP/IP Modbus connection



Group by location:
The modules are grouped by the locations defined in the project.

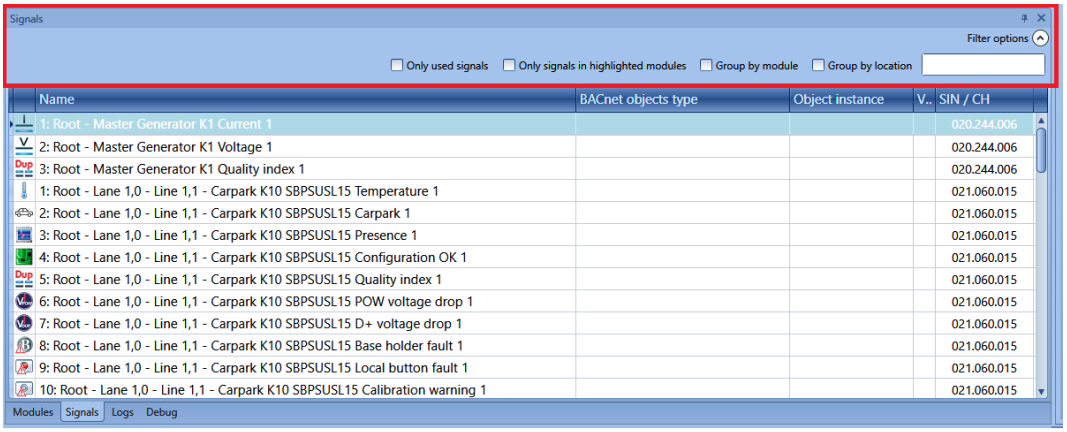


Search box:
The results will include all the words that contain the input string and the search will be carried out on the modules available in the project. The user can insert the full name or part of it.

14.2 How to manage the filters in the Signals window

The signals can be sorted or filtered by *Filter options*: by clicking on the icon, the filter options panel will be opened: only the signals specified by the filters will be displayed.

The available filters are shown in the top of the *Signals* window: see the red rectangle below:

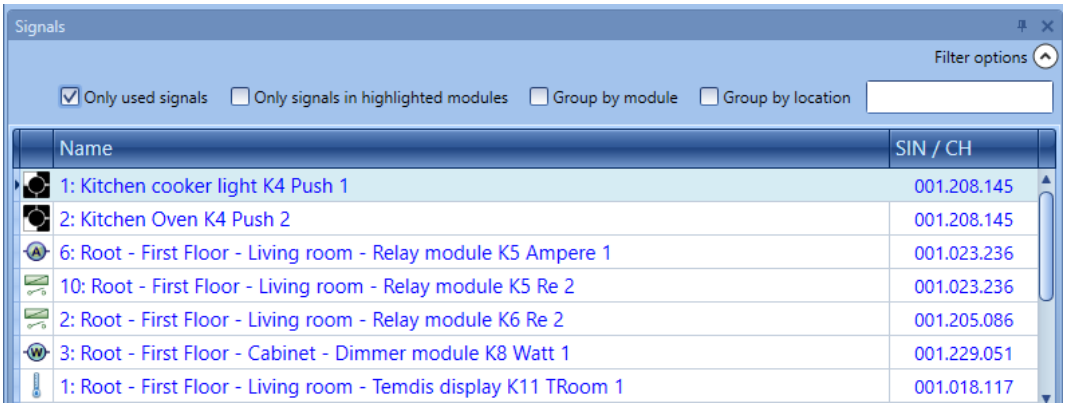


When the option panel is shown, the following filters are available. The filters can be used individually or can be mixed:

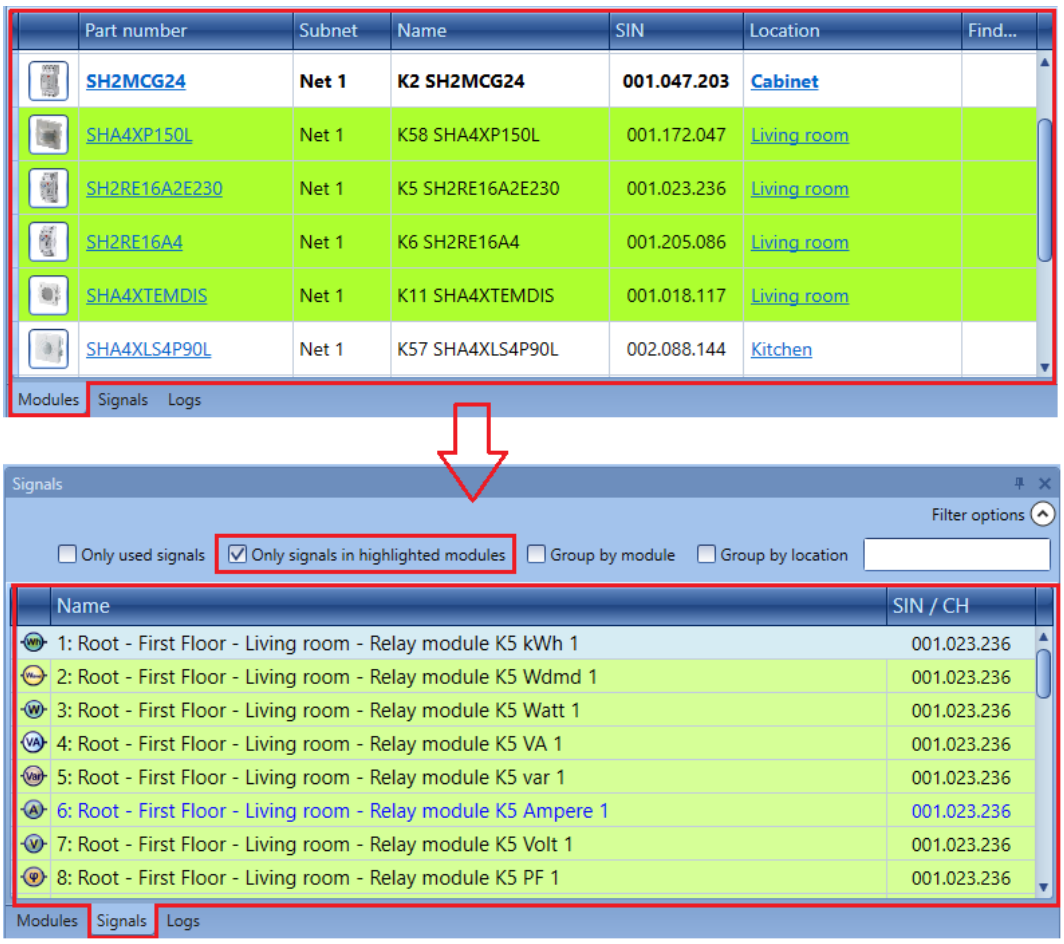
For more details about the SIN number, please refer to the UWP 3.0 Tool software manual:

http://www.productselection.net/MANUALS/UK/uwp3.0_tool.pdf

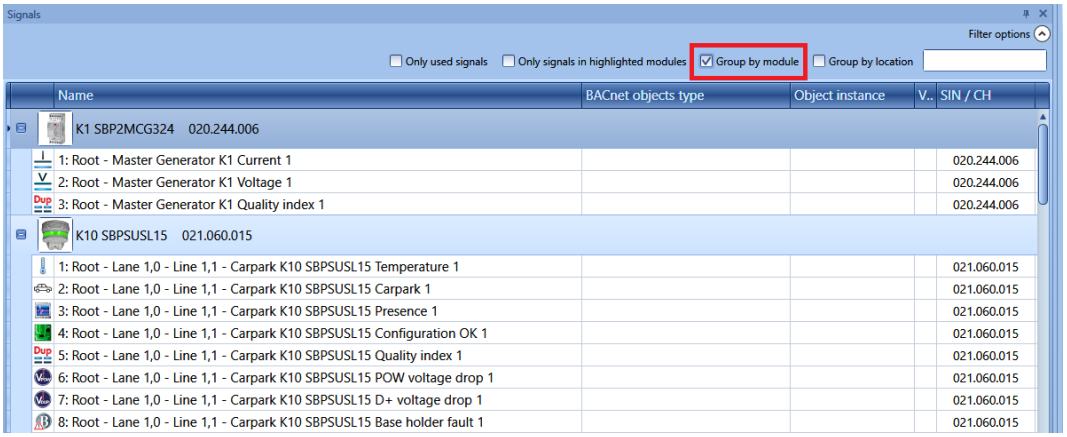
Only used signals:
Only the signals already used in at least one function are presented and they are highlighted in blue:



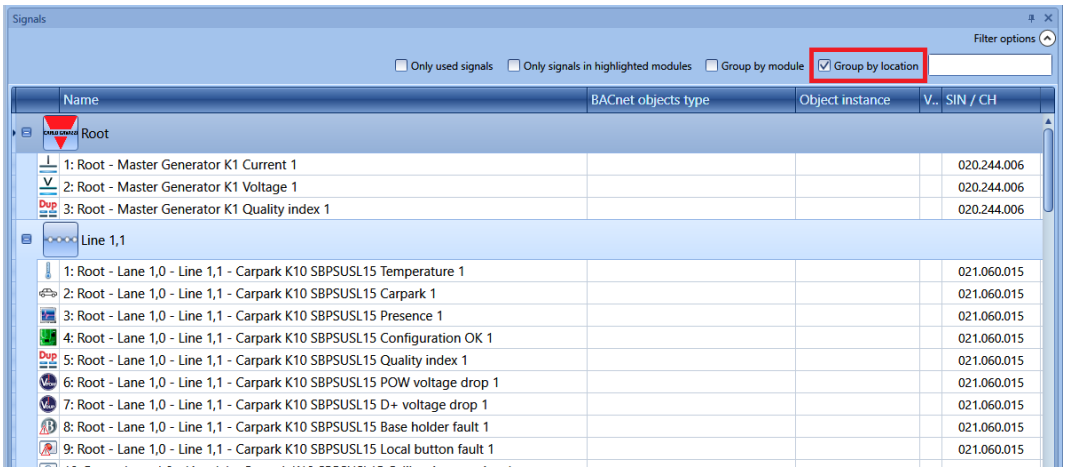
Only signals in highlighted modules:
When this filter option is selected, only the signals belonging to the highlighted modules are shown in the *Signals window*.
See the example below:



Group by module:
The signals are grouped by their own module, as shown in the example below:



Group by location:
The signals are grouped by location.



Search box:
The results will include all the words that contain the input string and the search will be carried out on the modules available in the project. The user can insert the full name or part of it.

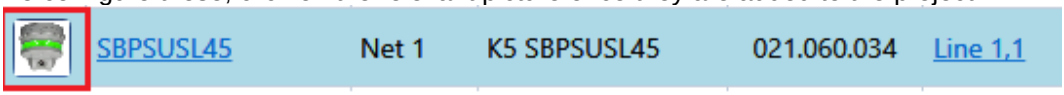
14.3 Car Park modules

14.3.1 Ultrasonic sensors

SBPSUSL: indoor sensor 0° degrees (although the little ‘o’ is enough by itself)
SBPSUSL45: lane mount sensor 45° degrees

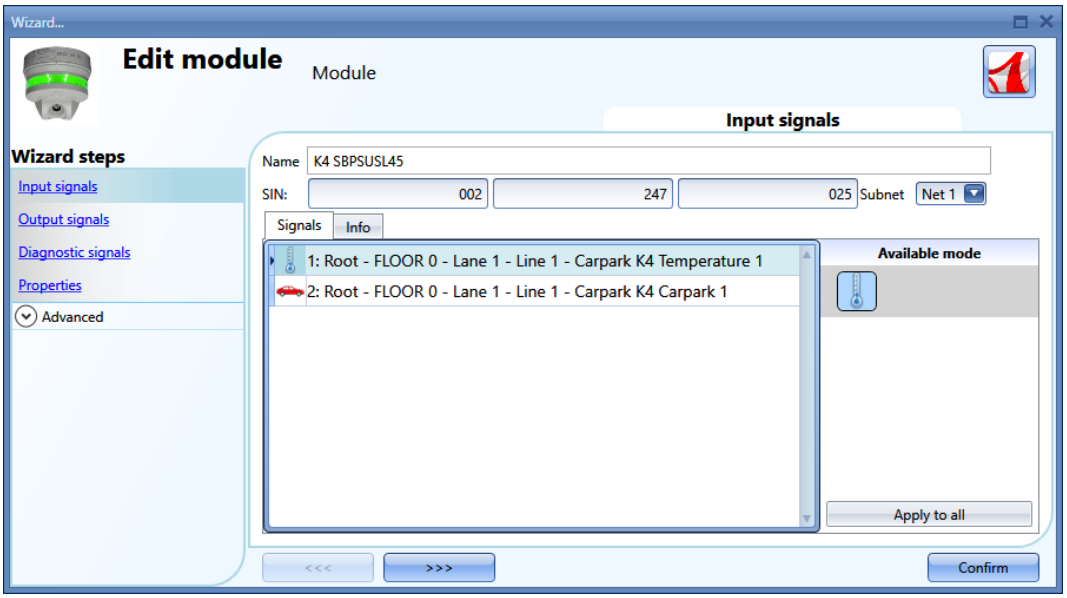
These modules can be added only in the location *Line*.

To configure these, click on the relevant picture once they are added to the project:



The configuration wizard will appear:

Input signals



In the *Input signals* field, two signals are available to be used in any functions or in the database or read via Modbus or BACnet:

1) The temperature read by the sensor
2) The presence of a car in the parking bay monitored by the sensor

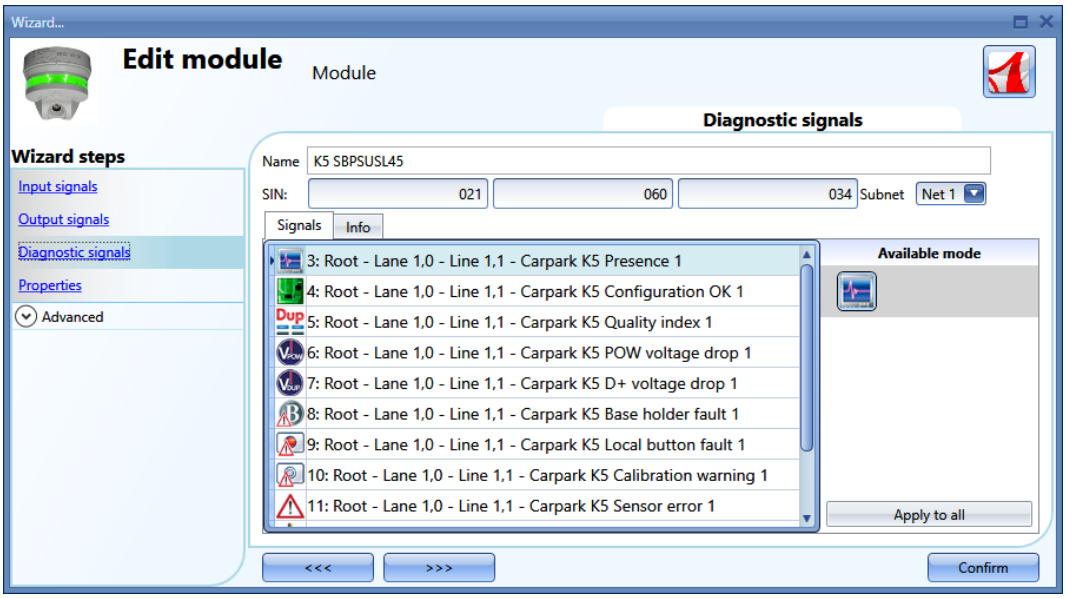
For more information about how to use these signals, please refer to the software manual:

http://www.productselection.net/MANUALS/UK/uwp3.0_tool.pdf

Output signals

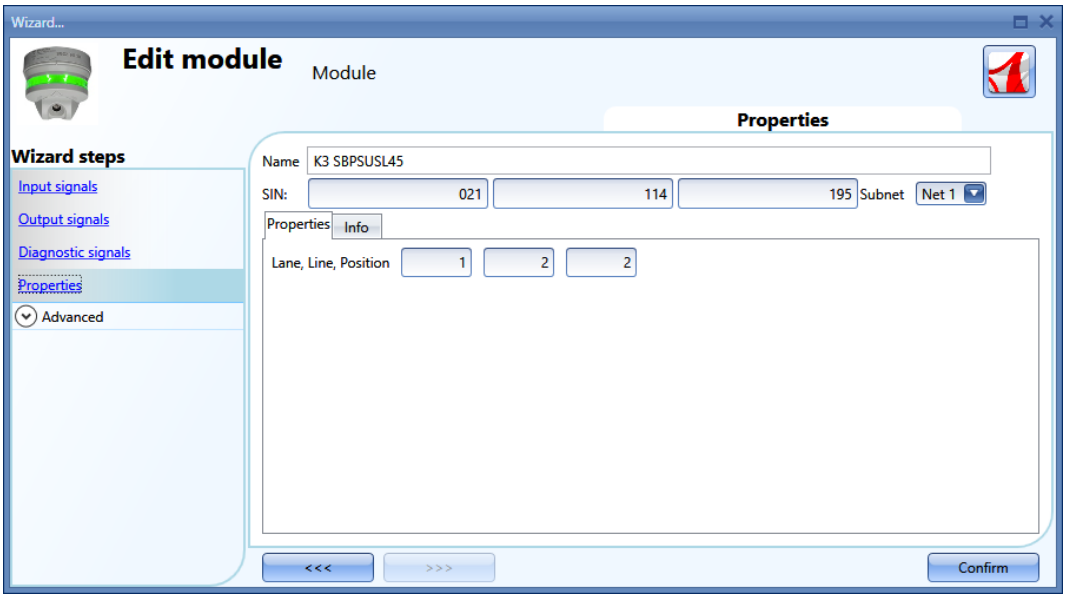
There are no output signals that can be managed by the controller.

Diagnostic signals



Icon	Description
	The module is alive
	The module is programmed
	Quality index of the Dupline bus (=100 ok, <100 noise is present on the bus)
	Voltage drop on the Dupline third wire
	Voltage drop on the Dupline bus
	Base holder is faulty
	The pushbutton of the sensor is always on
	Calibration error: no echo received
	Ultrasonic sensor is faulty
	Cross talk: signals received from another sensor
	Dupline power voltage
	Dupline bus voltge

Properties



In the properties window, the following parameters can be edited:

SIN
This field can be compiled manually or automatically by means of the Dupline network discovery

Subnet
The network can be selected manually or automatically by means of the Dupline network discovery

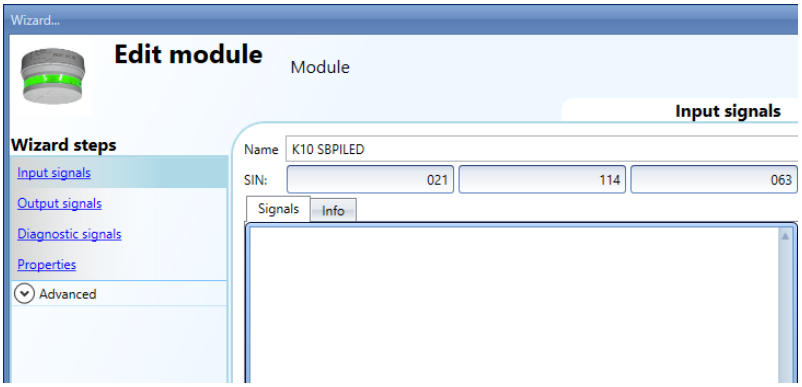
Lane, line, position
Car park address

14.3.2 Lane indicator

SBPILED: indoor lane indicator

To configure these, click on the relevant picture once they are added to the project:

Input signals

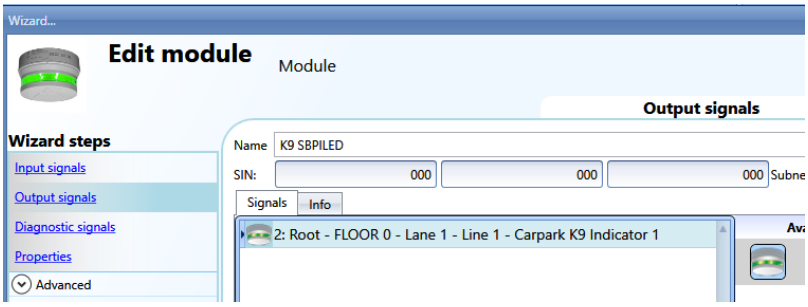


In the *Input signals* field, one signal is available to be used in any function or in the database or read via Modbus or BACnet: for example the temperature read by the sensor

For more information about how to use these signals, please refer to the software manual:

http://www.productselection.net/MANUALS/UK/uwp3.0_tool.pdf

Output signals

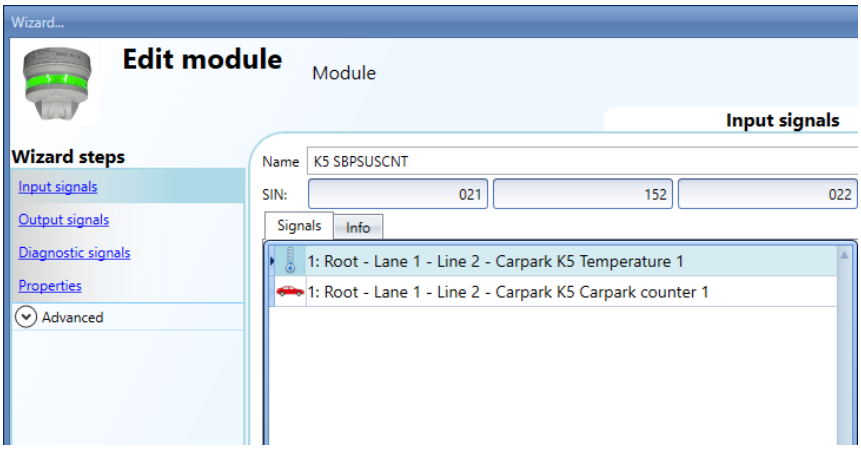


One output signal, the LED, can be managed by any logic in the system.

Diagnostic signals
They are those as described for the sensors.

14.3.3 Counter sensor

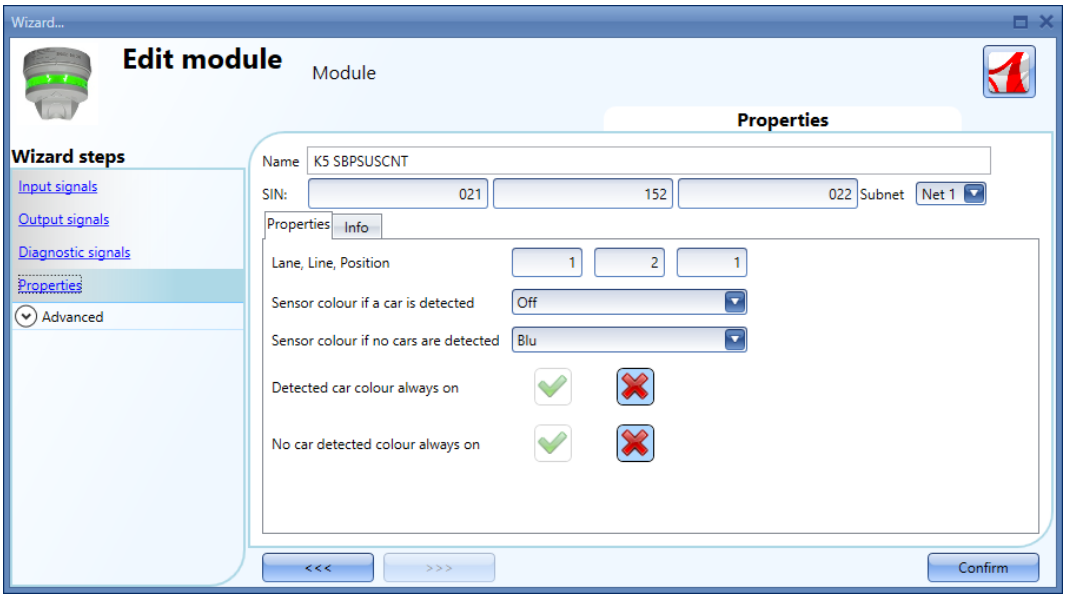
SBPUSCNT: indoor counter sensor



These modules can be added only in the location *Line*.

The fields *Input signals*, *Output signals*, *Diagnostic signals* are the same as already described for the SBPSUSL45.

Properties



Lane, Line, Position:

Sensor colour if a car is detected: Colour of the sensor when a moving car is detected

Sensor colour if no cars are detected: Colour of the sensor if no cars are moving under it

Detected car colour always on: The LED of the sensor will always be the one selected in the field *Sensor colour if a car is detected.*

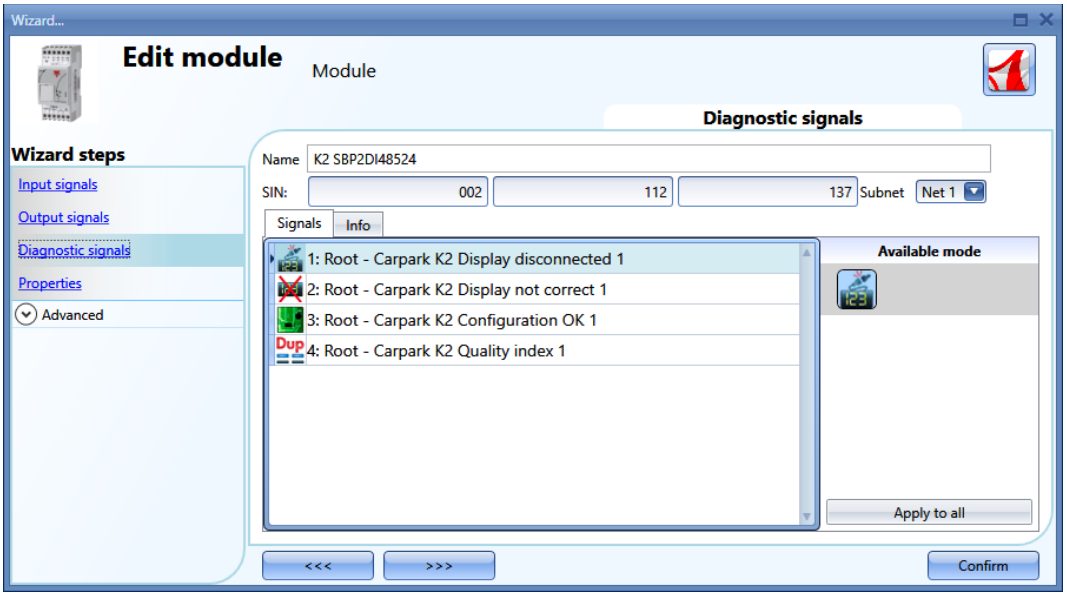
No car detected colour always on: The LED of the sensor will always be the one selected in the field *Sensor colour if a no cars are detected.*

14.3.4 RS485 to smart-dupline interface

SBP2DI48524

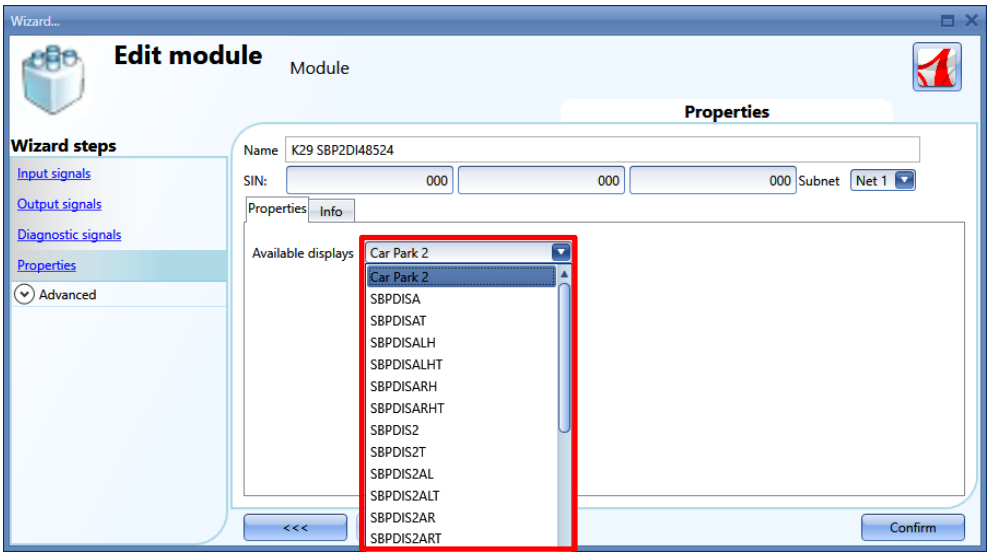
In the RS485 to smart-dupline interface there are no *Input/Output signals*

Diagnostic signals



	This signal indicates if the display is properly connected
	This signal indicates if the type of display connected is the one selected in the field *Properties*
	This signal indicates if the module has been programmed
Dup	Quality index of the Dupline bus (=100 ok, <100 noise is present on the bus)

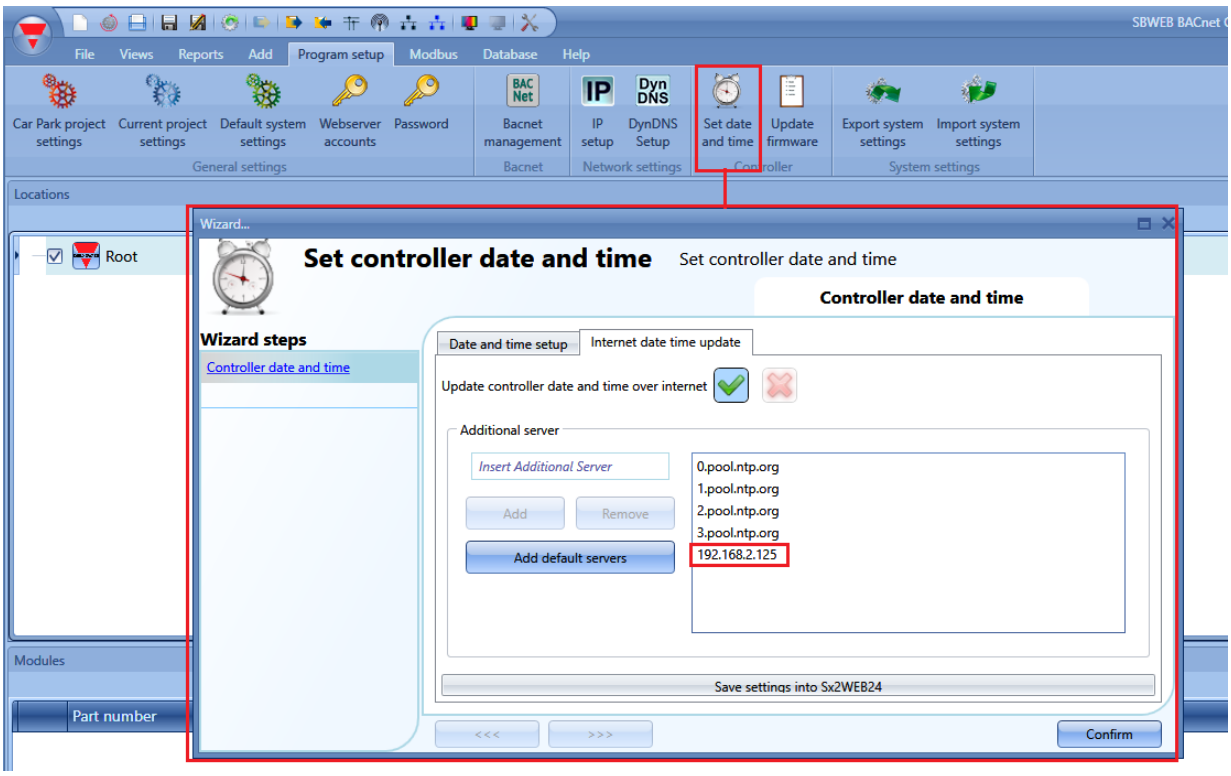
Properties



In this field, the display connected to the SBO2DI48524 has to be selected.

15 Time server

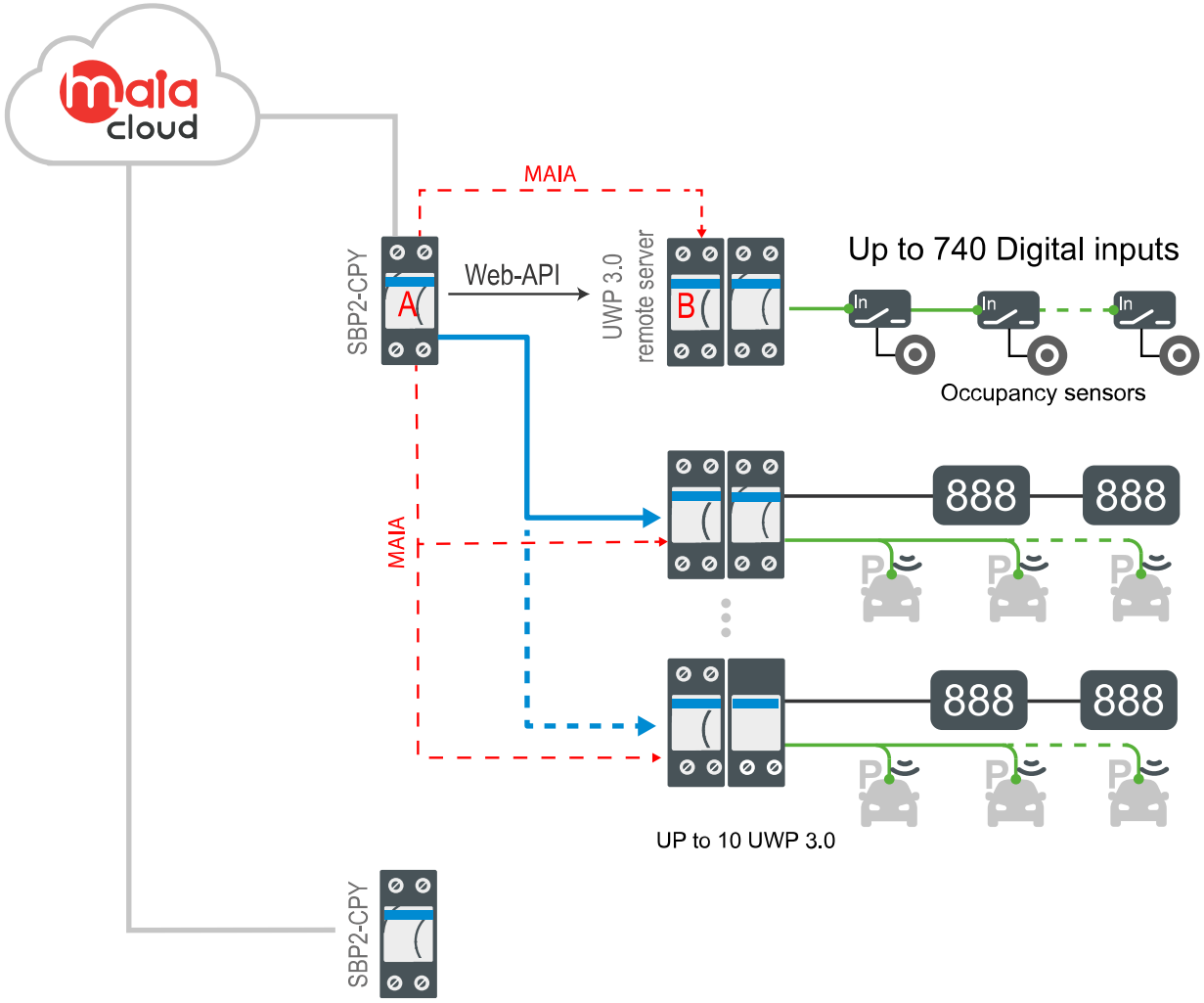
If the car park controller UWP 3.0 needs to synchronise the time via Internet, it has to go through the SBP2CPY24, so in the *Set controller date and time* window, the address of the SBP2CPY24 has to be added. It can also be a DYNDNS address.



16 Remote Server for digital input signals

Thanks to this feature, you can use a UWP3.0 controller as a **Remote Server** to gather the status of up to 740 digital input signals of Smart Dupline® and WiDup modules. After that, the status of the signals will be sent via a proprietary API Service to an external SBP2CPY24 server where they will be managed as parking bays (like the SBPSUSLxx indoor sensors).

This solution is perfect to interface generic devices equipped with output contact (such as zone or occupancy sensors) and then manage them as Car Park sensors from the CPY Server.



16.1 System architecture

A) An external SBP2CPY24 Server (required) collects the occupancy information from indoor sensors and digital input signals

(B) A UWP 3.0 controller used as **Remote Server** where the digital input signals belong to Smart Dupline® and WiDup networks (up to 740 sensors)

(C) Up to 10 UWP 3.0 controllers for SBPSUSLxx indoor sensors, indicators and displays (up to 6300 indoor sensors).

Note: if you don't need to manage any SBPSUSLxx indoor sensor in the Car Park project, an additional UWP 3.0 controller is required to manage the DISxRSE displays.

16.2 System requirements

To use this functionality, you have to enable all the Car park services in the UWP 3.0 controller used as **Remote Server**. Via the web API protocol, you will be able to send the digital signal status to an external CPY server so to view them.

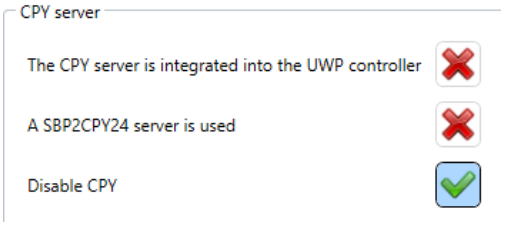
Following are the three macro-steps to complete the configuration:

- **From the UWP Tool:** select the digital input signals and associate each signal to an area/a zone. The system will assign to each signal an ID automatically.
- **From the WEB-App:** set a username/password to enable the **API service**
- **From the CPY Server:** set the parameters to get the information from the UWP 3.0 connected to the digital input signals treated as Car Park bays.

16.2.1 How to configure the digital input signals

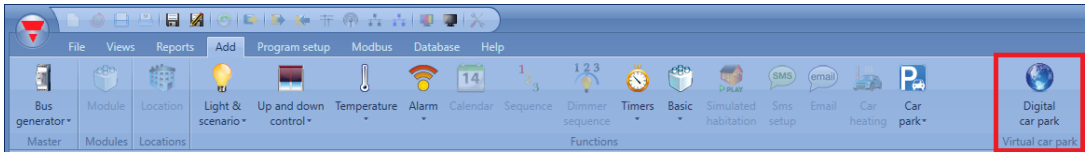
*Note: a UWP 3.0 controller has to be used as **Remote Server** only to achieve load balancing and avoid inefficiency; all the other car park functionalities must be disabled.*

First, if you cannot select the icon, open the **Program setup**, click **Car Park project settings** and open the **CPY server** tab. Select the **Disable CPY** option (see the print screen below) and click **Confirm** to save the changes.

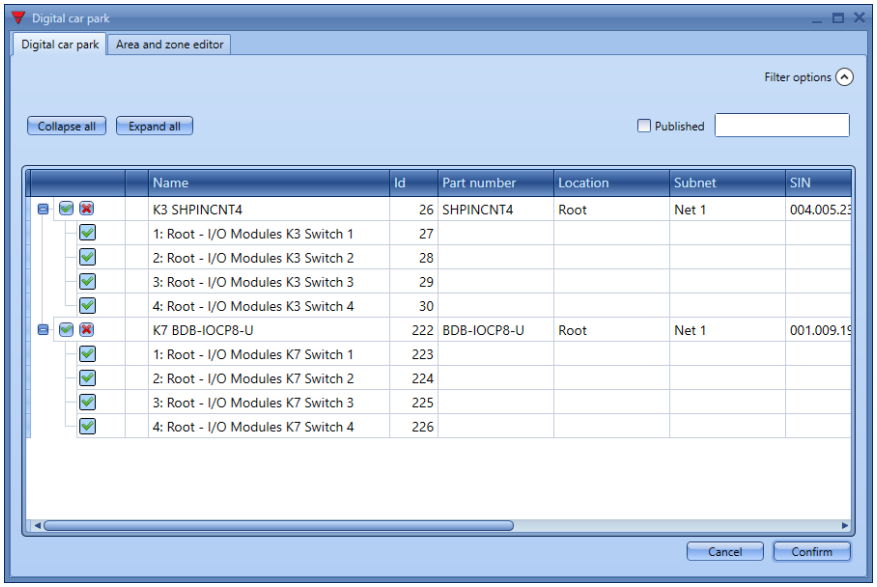


After that, be sure you have added all the modules to which belong the digital input signals you want to manage from the CPY Server. Moreover, you have to select the **Switch** mode from the module properties (***Input signals*** *tab* > ***Modules*** > ***Available mode***).

From the **Add** tab, click the **Digital car park** icon (see the picture below).



The following window appears:



From the **Digital Car Park** window, you will see only the modules that have at least one **Switch** signal.

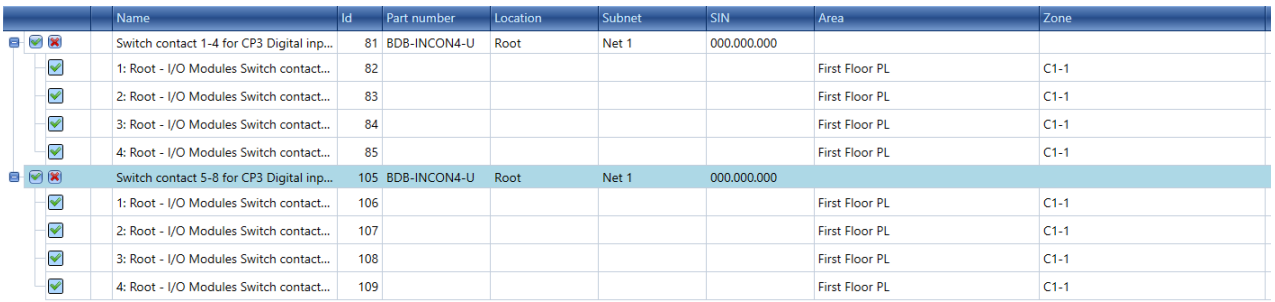
16.2.2 How to manage the filters

You can sort or filter the modules (**Filter options**). Following are the available filters:

Option	Description
Collapse all	All the modules and the relevant signal information are hidden
Expand all	All the modules and the relevant signal information are shown.
Published	Only the enabled digital signals are shown *(**V**)*.
Search box	The system will search among the available project modules and the result will include all the words containing the input string. Note: You can enter the full name or part of it.

16.3 Modules and signals information

For each module you will see the following information:



	Name	Id	Part number	Location	Subnet	SIN	Area	Zone
	Switch contact 1-4 for CP3 Digital inp...	81	BDB-INCON4-U	Root	Net 1	000.000.000		
	1: Root - I/O Modules Switch contact...	82					First Floor PL	C1-1
	2: Root - I/O Modules Switch contact...	83					First Floor PL	C1-1
	3: Root - I/O Modules Switch contact...	84					First Floor PL	C1-1
	4: Root - I/O Modules Switch contact...	85					First Floor PL	C1-1
	Switch contact 5-8 for CP3 Digital inp...	105	BDB-INCON4-U	Root	Net 1	000.000.000		
	1: Root - I/O Modules Switch contact...	106					First Floor PL	C1-1
	2: Root - I/O Modules Switch contact...	107					First Floor PL	C1-1
	3: Root - I/O Modules Switch contact...	108					First Floor PL	C1-1
	4: Root - I/O Modules Switch contact...	109					First Floor PL	C1-1

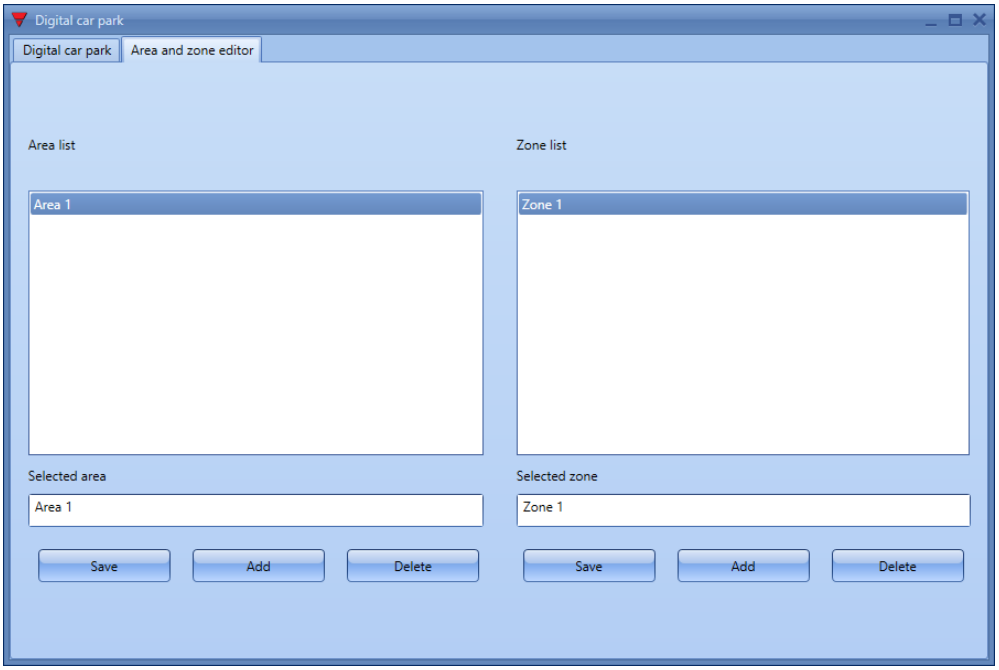
Field	Description
Selection box	If you check it for a module, also all the relevant signals will be selected to be enabled.
Name	Shows the module/signal name in the configuration. *N.B: from this window you cannot change the name.*
Id	Univocal identifier of the digital signal in the configuration. This reference is important to join the UWP 3.0 Tool and the CPY server signal. *N.B: the system associates this value automatically and you cannot change it. Each signal will be shown in the CPY Server with this reference.*
Part number	Shows the Smart Dupline® module part number
Location	Shows the module location
Subnet	Shows the Master Channel Generator
SIN	Shows the module SIN number
Area	Shows the digital signal area according to the ***Area and Zone editor*** tab configuration. *N.B: each area can have more zones.*
Zone	Shows the signal zone according to the ***Area and Zone editor*** tab configuration. *N.B: a zone can belong to an area only.*

16.4 Procedures

Following are the different procedures to publish digital input signals on the CPY Server.

16.4.1 How to add/edit an area

1 . Open the **Digital car park** window and go to the **Area and zone editor** tab



2 . From the **Area list**, **Add** an area (default name, **Area 2** because the **Area 1** is already present)

3 . From the **Area list**, select the area you want to rename

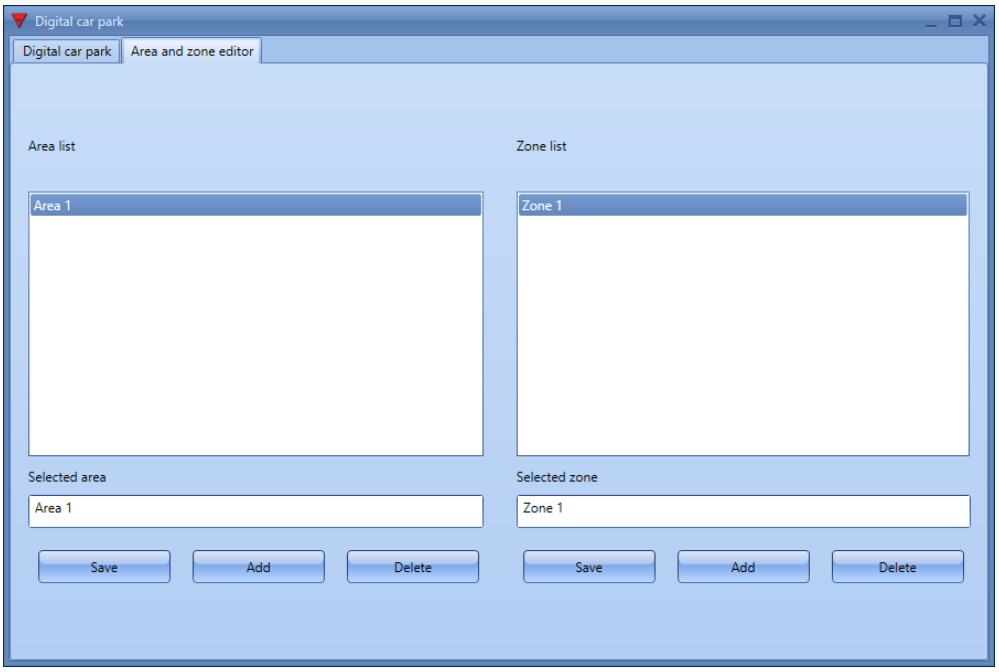
4 . In the **Selected area** enter the name you want to assign and click **Save**

5 . Go to the **Digital car park** tab and click **Confirm** to save the changes

16.4.2 How to add/edit a zone

Note: first you have to select an area.

1 . Open the **Digital car park** window and go to the **Area and zone editor** tab



2 . From the **Area list**, select the area to which you want to add a zone.
Note: the selection turns light blue.

3 . From the **Zone list**, click **Add** to add a zone with the default name.
*Note: if you select the **Area**, by default the **Zone1** is proposed.*

4 . From the **Zone list**, select the zone to rename.
Note: the selection turns light blue.

5 . In the **Selected zone** field, enter the name to associate to the zone and click **Save**

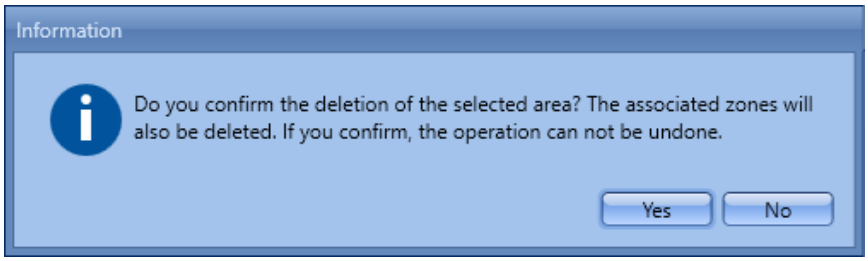
6 . Go to the **Digital car park** tab and click **Confirm** to save the changes

16.4.3 How to delete an area

1 . Open the **Digital car park** window

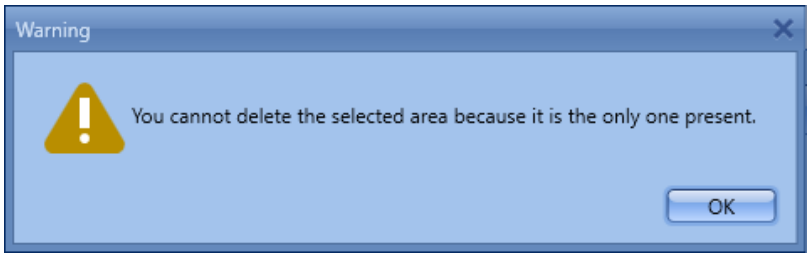
2 . From the **Area list**, select the area you want to delete.
Note: the selection turns light blue.

3 . Click **Yes** from the pop-up to confirm the deletion.

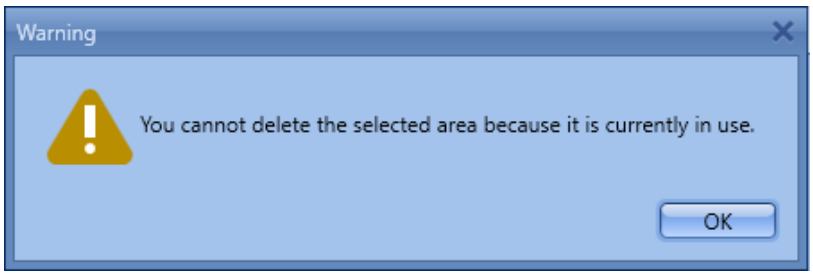


Notes:

- *The system shows a warning message if you try to delete the only configuration area.*



- *The system shows a warning message if you try to delete an area that is assigned to at least one digital signal*.



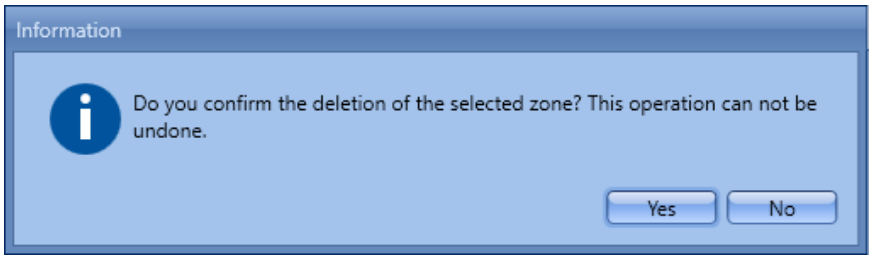
4 . Go to the **Digital car park** tab and click **Confirm** to save the changes

16.4.4 How to delete a zone

1 . Open the **Digital car park** window

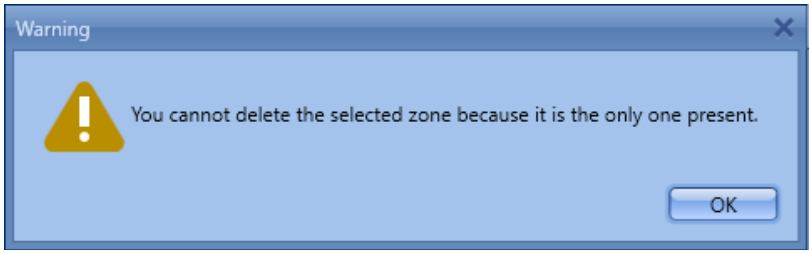
2 . From the **Zone list**, select the zone to delete.
Note: the selection turns light blue.

3 . Click **Yes** from the pop-up to confirm the deletion.

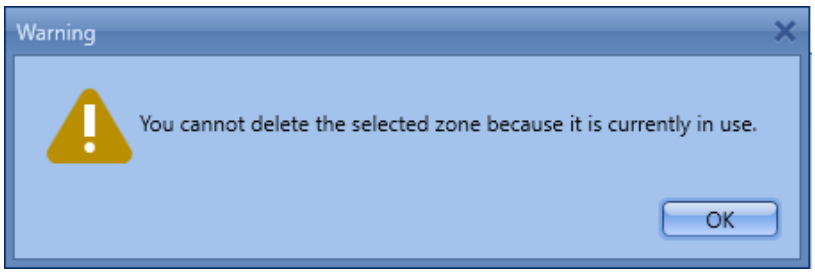


Notes:

- *The system shows a warning message if you try to delete the only area zone.*



- *The system shows a warning message if you try to delete a zone that is assigned to at least one digital signal*.



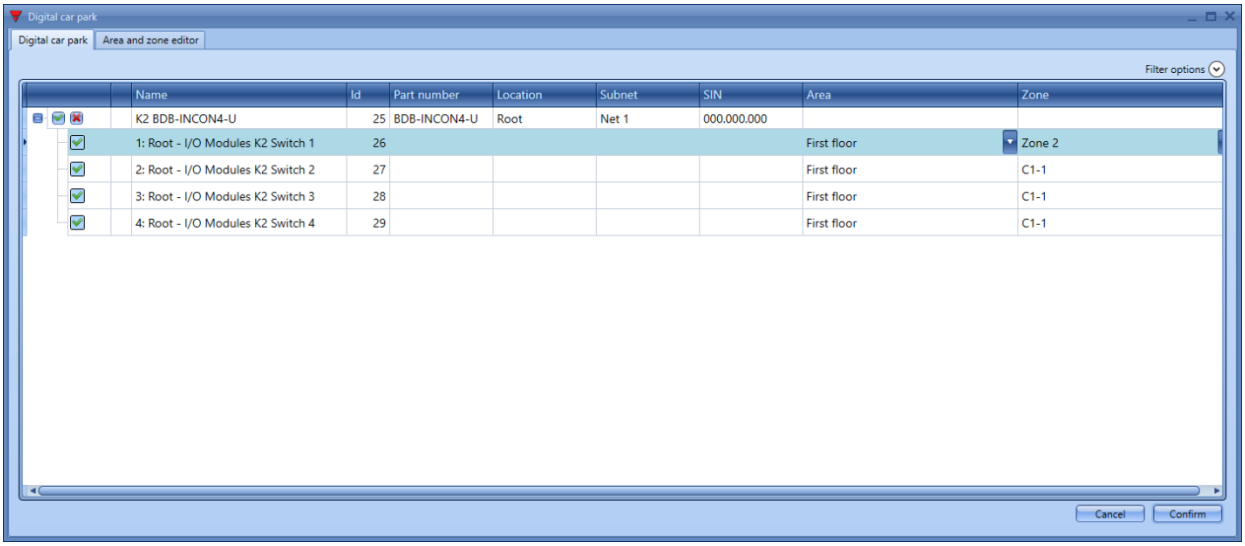
4 . Go to the **Digital car park** tab and click **Confirm** to save the changes

16.4.5 How to associate a sensor to an area/zone

By associating signals to an area/a zone, you can associate and map digital signals according to your needs. Since in the CPY Server each area represents a node and the zones represent a sub-level, the plant structure will result well organised.

Follow this procedure to enable and assign digital signals to an area/a zone.

1 . Open the **Digital car park** window



2 . From the **Digital car park** tab, expand the module view to see all the relevant digital signals

3 . From the first column, check the box of the signals you want to enable for the **Car Park Digital input** functionality.
*N.B: the **V** will be shown for the signals you enable.*

4 . From the **Area** column, assign an existing area to the selected digital signal.
To create a new area, see How to add/edit an area

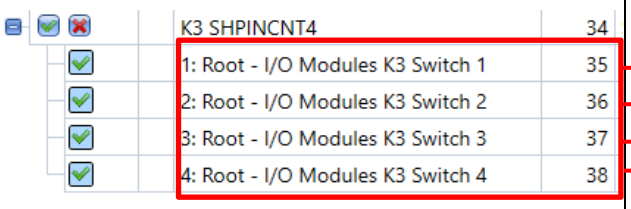
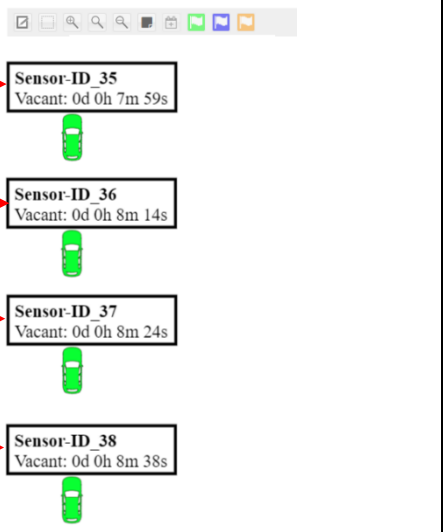
5 . From the **Zone** column, assign an existing zone to the selected digital signal.
Otherwise, see How to add/edit a zone

6 . Click ***Confirm*** to save the changes

16.4.6 How to check the association between the digital signal and the Car Park sensor

The association between the digital signal (configured from the UWP 3.0 Tool) and the CPY Server is carried out through ID parameter. The system assigns this parameter to each signal automatically (the user cannot change it) and each signal added to the configuration it will be shown in the CPY Server with that ID.
This ID has to be considered during the system planning and configuration in order to identify each signal as CPY server sensor easily.

The below example shows the association of four digital signals of a Smart Dupline® module and the relevant sensors in the CPY Server Car Park:

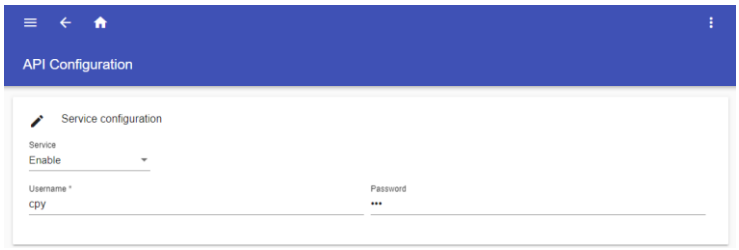
UWP 3.0 Tool - Car park digital window	CPY Server - Drawing
	

Click here to see the video of this procedure.

16.4.7 How to enable the API service in UWP 3.0

To build the communication between the UWP 3.0 module to which the digital signals are connected and SBP2CPY24, you have to enable the API service from the controller. Follow this procedure:

1. Access the controller WEB-App



2. Go to ***Services > API Configuration***
3. Enable the service
4. Enter the **Username** and the **Password**
5. Click  to save the changes

N.B: you have to enter this information in SBP2CPY24 to connect to the controller and get the occupation information of the digital signals.

17 MAIA Cloud for Car Park use cases

In Car Park applications there are two use cases which can be managed in MAIA Cloud:

1) For integrated CPY, you have to activate the UWP 3.0 in your organization. In this case the UWP 3.0 is considered a gateway.

2) For external CPY with multiple controllers, you can activate the SBP2CPY24 in your organization and add each UWP 3.0 as an endpoint, if all the devices are connected to the same LAN. In this case the SBP2CPY24 is considered a gateway and the UWP 3.0s endpoints.

17.1 How to register on MAIA Cloud

1. Open your browser
2. Go to the MAIA Cloud **Login** page: https://app.maiaconnect.com
3. Click **Register** under the **Log In** button
4. Enter the following data:
   - **First name**
   - **Last name**
   - **Organization Label**

     Note: this description identifies your Organization. You can use your Company or your project name and you can modify it later.
   - **Organization ID**
     Note: this is your unique Organization identifier, useful for technical support. It cannot be changed later and special characters are not allowed.
   - **Country**
   - **Valid UWP-ACTIVATION-KEY for Registration.** Write the Carlo Gavazzi activation key included in your UWP-ACTIVATION-KEY kit.
   - **E-mail** and **E-mail confirmation**
   - **Password** and **Password confirmation**
5. Read and accept the *Privacy policy* and *Terms of Use*
6. Click **Register**
7. Click the link included in the mail you received to enable your profile
8. Log in with your credential to the MAIA Cloud web portal.

Notes:

- *Your user is automatically registered as Administrator of the organization named after your company.*
- *The **UWP-ACTIVATION-KEY** can be used once to register on MAIA Cloud. Use the same key to add the device to your organization. For further details, see the **Activation key User Manual***

17.2 How to activate a device in MAIA Cloud

1. Open a browser
2. Log in to your MAIA Cloud organization (https://app.maiaconnect.com)
3. Click ☰ to open the main menu
4. Go to ***Devices*** > ***Activate***
5. Complete the activation page with the device information:
   - **Device Label** (the device name)
   - **Latitude** and **longitude** for the location

   Note: you can navigate the map or use the search box.

- **UWP-ACTIVATION-KEY**. Write the Carlo Gavazzi activation code included in your UWP-ACTIVATION-KEY kit.

6. Click ✔
7. Go to your MAIA Cloud home page
8. Click ⋮ > ***Assign credit*** to enable the VPN service on your device.
   *Note: to assign credits, you need at least one unused VPN month. To add resources to your organization, you need a UWP-LICENCE code (see the **Licence code User Manual**).*
9. Click ⌄ > ***Application profile*** to set the default profile.
10. For the UWP 3.0 Tool release

If you use the version…	Then…
8.4.0.3 and higher	in few seconds UWP 3.0 will be online
8.4.0.3 and lower	go to ***How to enable VPN service for an installed UWP 3.0*** and follow the procedure

For the SBP2CPY24 release

If you use the version…	Then…
2.6.3 and higher	in few seconds SBP2CPY24 will be online
2.6.3 and lower	go to ***How to enable VPN service for an installed SBP2CPY24*** and follow the procedure

17.2.1 How to enable VPN service for an installed UWP 3.0

1. Go to your MAIA Cloud organization and activate your UWP 3.0
   *See **How to activate a device in MAIA Cloud***
2. Update your UWP 3.0.
   Note: the VPN service is available in the UWP 3.0 Tool 8.4.0.3 onwards.
3. Log in to the UWP 3.0 web app
4. Click ☰ to open the main menu
5. Go to ***Service*** > ***Remote VPN Services***
6. Enable the service
7. Write the activation code of your UWP-ACTIVATION-KEY kit
   *Note: check that the **Standard MAIA Cloud Server** has been set.*
8. Click 🖫 to save
   Note: the green icon informs you that the procedure is successfully finished.

17.2.2 How to enable VPN service for an installed SBP2CPY24

1. Go to your MAIA Cloud organization and activate your SBP2CPY24
   *See **How to activate a device in MAIA Cloud***
2. Update your SBP2CPY24
   Note: the VPN service is available in the SBP2CPY24 2.6.3 onwards.
3. Log in to the CPY server
4. Go to ***System settings*** > ***VPN settings***
5. Enable the service
6. Write the activation code of your UWP-ACTIVATION-KEY kit
   *Note: check that the **Standard MAIA Cloud Server** has been set.*
7. Click 🖫 to save
   Note: the green icon informs you that the procedure is successfully finished.

17.3 How to add an endpoint in MAIA Cloud (in the case of external CPY with multiple controllers)

For external CPY with multiple controllers, you can activate the SBP2CPY24 in your organization and add each UWP 3.0 as an endpoint.
You can follow this procedure to add the controllers only if the external CPY and the UWP 3.0s are connected to the same LAN.

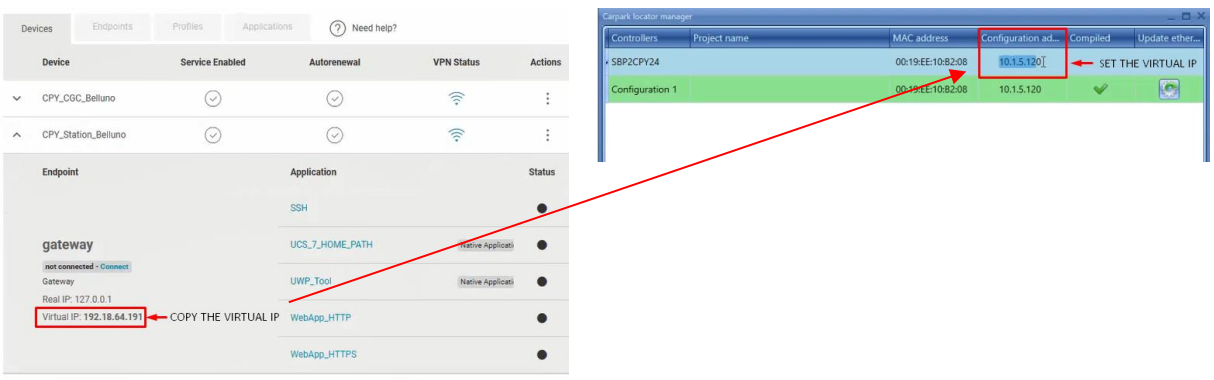
1. Open the **main menu**
2. Go to ***Devices > VPN***
3. Go to the **Endpoints** tab
4. Click ✚ from the **Actions** column of the desired SBP2CPY24
5. From the **Endpoint Options** menu, enter the following information:
   - **Name**
   - **Description**

     Note: it is not mandatory but it is useful to find an endpoint faster.
   - **IP address**
   - **Application profile**. Click ⌄ and choose *UWP 3.0 default profile*
6. Check the **Enabled** box to activate the endpoint
7. Click ✔ to save the configuration.
   Note: after this procedure you need to reset the VPN service.

Note: The maximum number of endpoints is 2. If you need to add more endpoints, click ***Edit Gateway****. If you change it, then you have to reset the VPN service.*

17.4 How to send a configuration remotely to an integrated CPY using MAIA Cloud

1. Open a browser
2. Log in to your MAIA Cloud organization (https://app.maiaconnect.com)
3. Connect to the integrated CPY through the UWP Tool using the integrated CPY virtual IP
   For more information see How to connect to the controller remotely via MAIA Cloud (VPN)
4. Click **Read the configuration from controller** and then **Close** at the end of the procedure
5. Click **Send to controller** to send the configuration
6. Save the configuration and then **Close** at the end of the procedure
7. Click **Project Configuration** and open the **Configuration/controller list**
8. Change the integrated CPY IP and set the virtual IP.
   Note: copy the relevant integrated CPY virtual IP that you find in MAIA Cloud, and paste it to the ***UWP Tool Carpark locator manager****.*

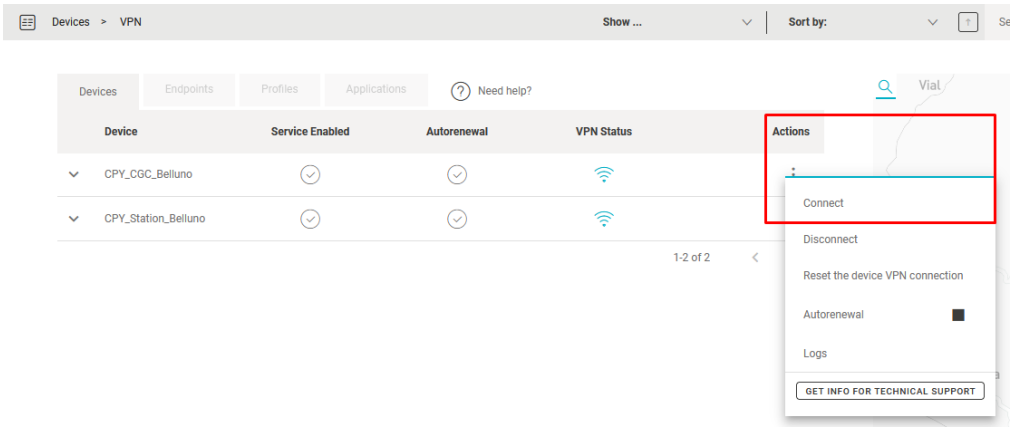


9. Click **Compile the project** and then **Close** at the end of the procedure
10. Send the configuration to integrated CPY clicking **Send configuration to SBP2CPY24** and then **Close** at the end of the procedure.

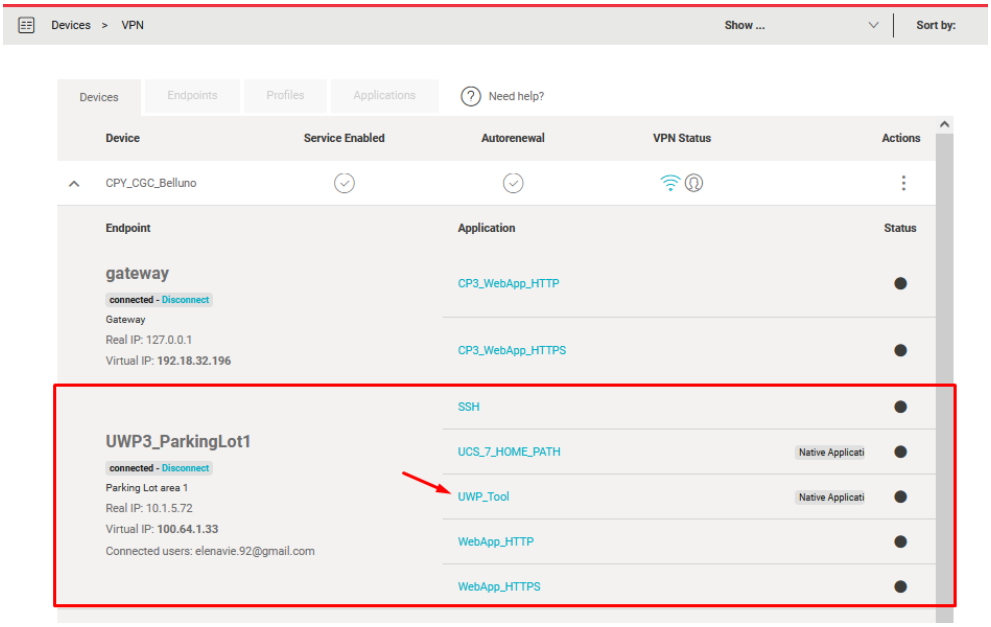
Click here to see the video of this procedure.

17.5 How to send a configuration remotely to an external CPY with multiple controllers using MAIA Cloud

1. Open a browser
2. Log in to your MAIA Cloud organization (https://app.maiaconnect.com)
3. Click ⋮ > ***Connect*** from the **Action** column of the integrated CPY you need to manage.
   *Note: The **Connection** drop-down menu will automatically open.*



4. Click UWP Tool from the **Connection** drop-down menu to set up a remote connection to the UWP 3.0.
   For more information see How to connect to the controller remotely via MAIA Cloud (VPN)

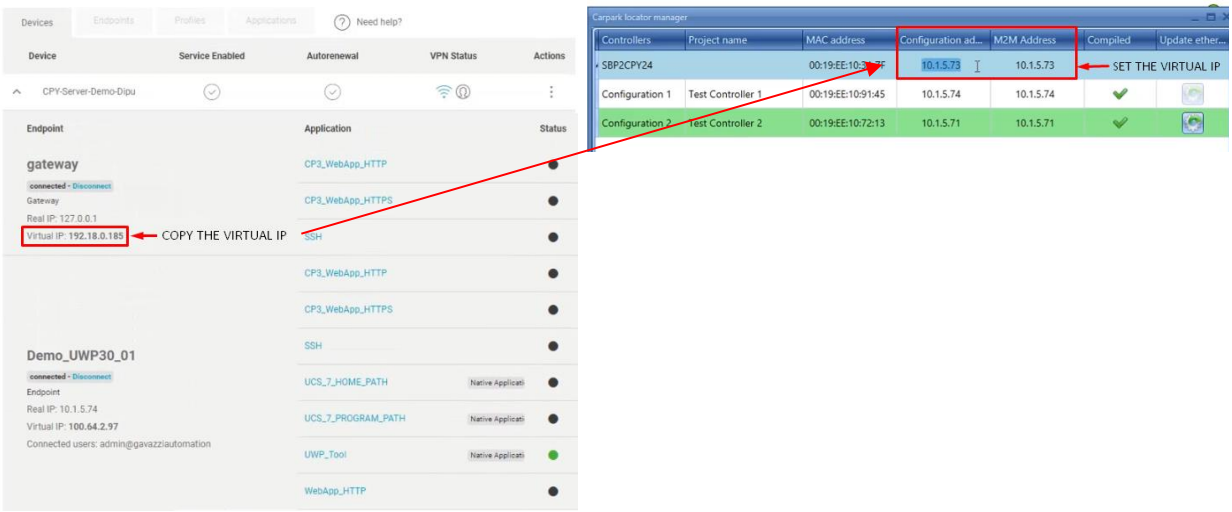


5. Click **Read the configuration from controller** and then **Close** at the end of the procedure
6. Click **Send to controller**
7. Save the configuration and then **Close** at the end of the procedure
8. Click **Project Configuration** and open the **Configuration 2**
9. Set up a remote connection via UWP Tool to the controller 2 using its Virtual IP

   Tip: go back to MAIA Cloud and copy the UWP 3.0 Virtual IP from the UWP Tool
10. Click **Send to controller** and then **Close** at the end of the procedure

    Note: repeat the steps 8-9-10 for each configuration
11. Click **Project Configuration** and open the **Configuration/controller list**
12. Change the integrated CPY IP and set the Virtual IP

    Note: copy the relevant integrated CPY Virtual IP you can find in MAIA Cloud, and paste it to the UWP Tool ***Carpark locator manager.***



13. Click **Compile the project** and then **Close** at the end of the procedure
14. Send the configuration to external CPY clicking **Send configuration to SBP2CPY24** and then **Close** at the end of the procedure.

Click here to see the video of this procedure.

18 Troubleshooting

This part of the manual deals with common problems the user can encounter during the project configuration or sensor calibration and, as shown below, possible solutions are suggested:

Problem	Solution
The sensor is not installed between 2.2m and 2.4m	Set the *Near End Position* field using the following formula: *Near End position = [Height of the sensor – 0.2 m]* *See how to change the sensor settings*
Bay is narrower than 2.5m	Set the *Total Peak Out* field = 3 *See how to change the sensor settings*
Bay is long, but without adjacent bays	Set the *Far End Position field* > 3.68m *See how to change the sensor settings*
Crosstalk	Identify the sensor which is creating crosstalk and modify its address *See how to identify and solve a Crosstalk condition*

~~18.1~~ How to change the sensor settings

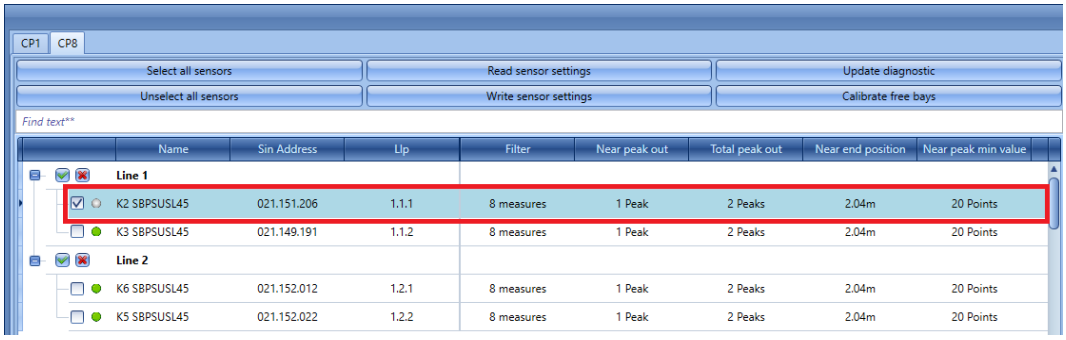
There are two different ways to change the settings parameters of the sensors:

- Individually – The parameters can be changed for each sensor, one by one;
- Multiediting – The parameters can be changed for multiple sensors at the same time by the *Multi editing* fields;

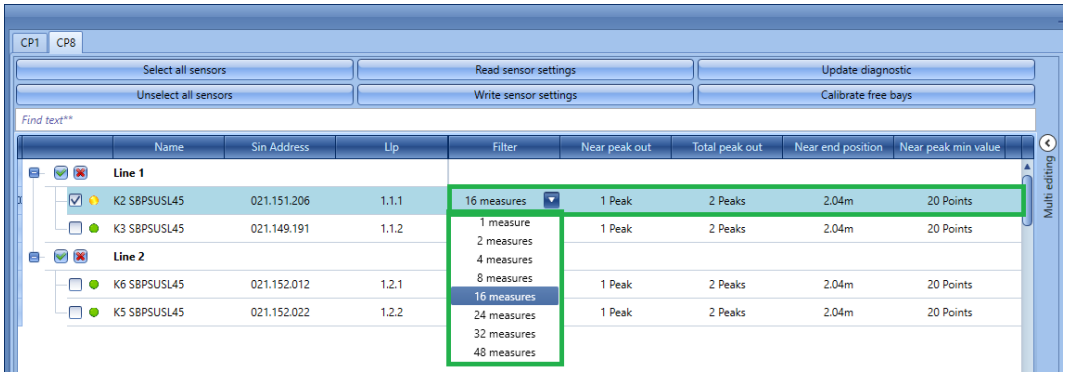
18.1.1 How to change the sensor settings individually

In the *Sensor list* of the *Calibration* window to change the current settings of a sensor the user can follow the procedure shown below:

1) Select the sensor that has to be managed by checking the check-box: it will be highlighted in light-blue, as shown in the red rectangle below:

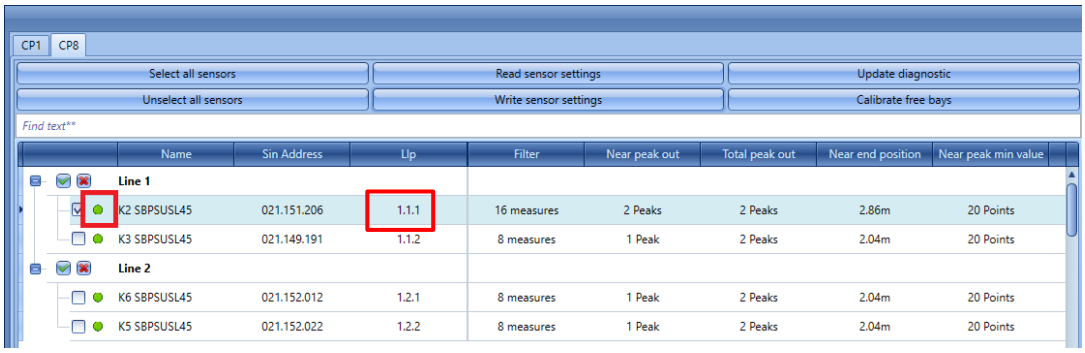


2) Set the new values in the required fields according to the project requirements. (See *Sensor parameters settings* table for more details)



Note: After changing at least one parameter, the small dot next to the sensor will turn to yellow; this means that the data of the sensor is not synchronized between the settings in the UWP 3.0 Tool and the data saved on the sensor.

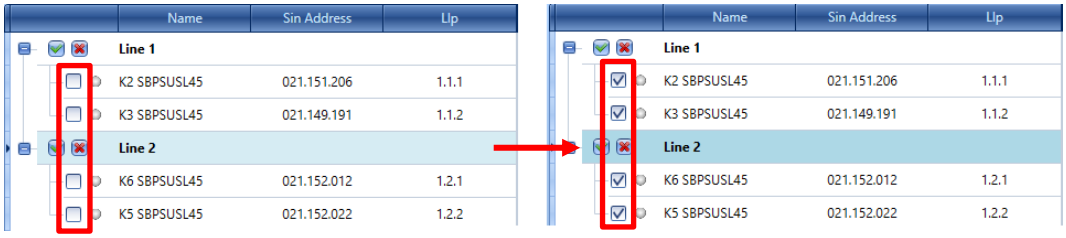
3) Click on the *Write sensor settings button to save the changes:* the dot next to the sensor will be coloured in green to indicate that the data has been synchronized properly, as shown below:



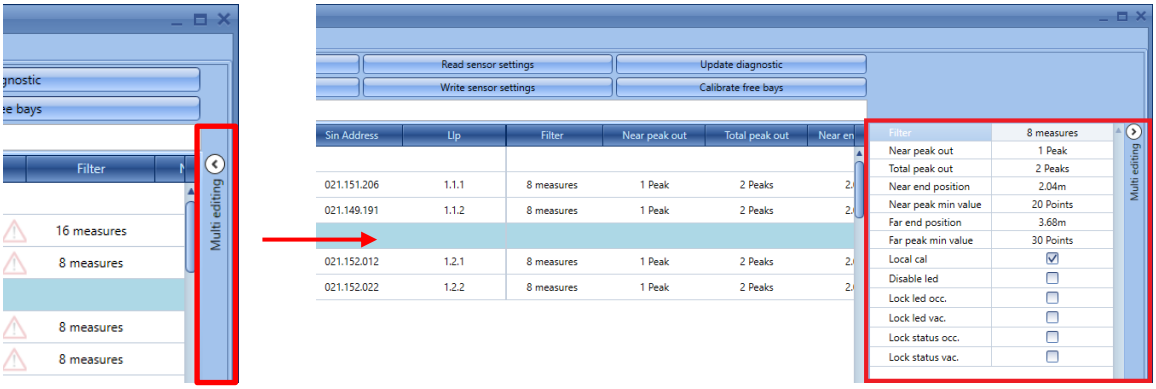
18.1.2 How to change the settings to multiple sensors

The user can set the settings parameters to multiple sensors at the same time by using the fields available in the *Multi editing* panel. The user can follow the procedure shown below:

1) Select the sensors that have to be managed by checking the check box next to each sensor row, as shown below:

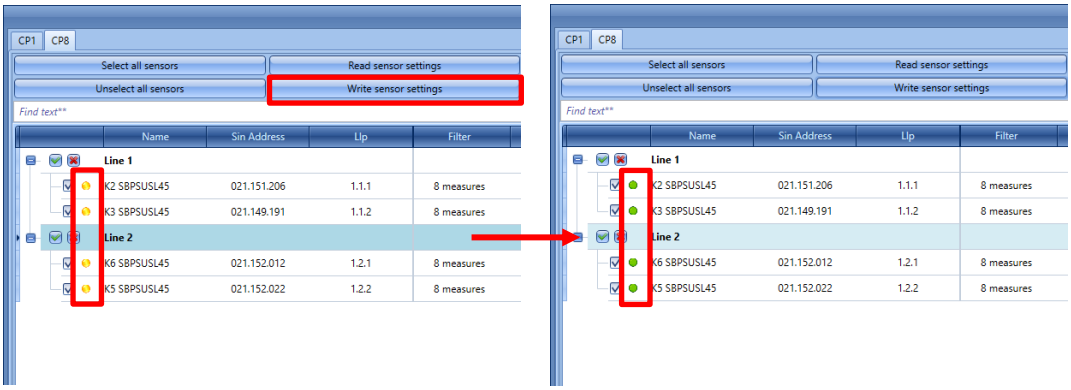


2) Expand the *Multi editing* panel in the right side of the *Calibration* window by clicking on the > button. The available fields of the panel will appear:



3) Set the new values in the required fields according to the project requirements. (See the *Sensor parameters settings* table for more details)

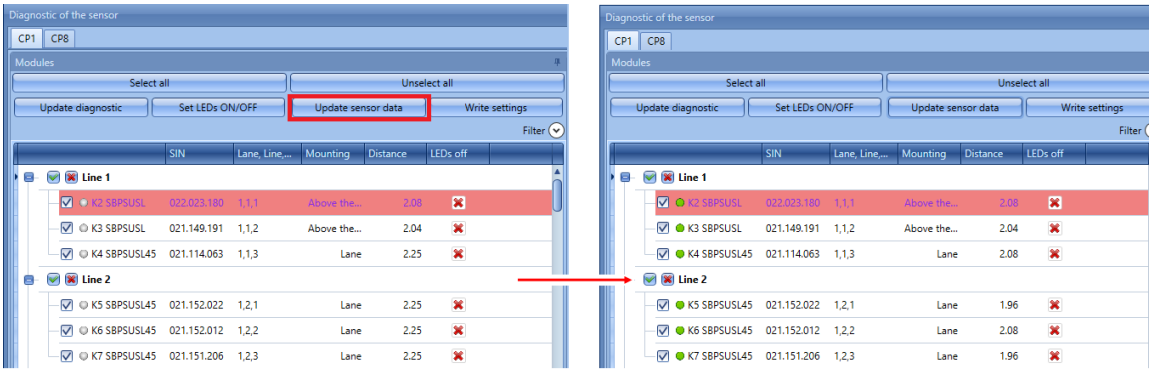
4) Click on the *Write sensor settings* button to send the new settings to all the selected sensors, see below



The dot next to the sensor will be coloured in green to indicate that the new data has been synchronized properly.

18.2 How to update the sensor parameter in the UWP 3.0 Tool

The user can read the configuration of the sensors by clicking on the *Update sensor data* button**:** the system will read the configuration of all the selected sensors and the dot next to the sensor name will turn to green, as shown in the example below:

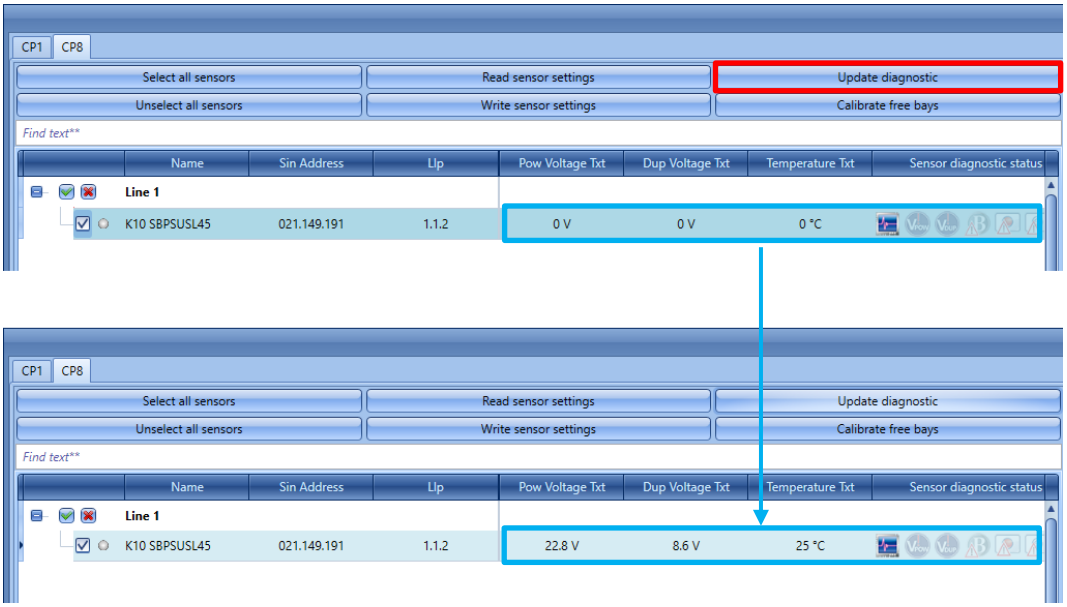


The user can check the sensor status by means of the small dot next to it. The table below reports the meaning of the different colours:

Colour	Meaning
K6 SBPSUSL45	Data of the sensor is not synchronized
K6 SBPSUSL45	Data of the sensor is synchronized
K6 SBPSUSL45	Data of the sensor needs to be synchronized
K6 SBPSUSL45	The sensor is not communicating

18.3 How to update the diagnostic signals

To update the diagnostic parameters of the sensors, click on *Update diagnostic:* the UWP 3.0 Tool will read the diagnostic status of the selected modules: all the relevant parameters will be updated, as shown in the blue boxes below:



The diagnostic signals that can be detected by the system are the following:

Icon	Tool tip	Description
	Sensor present	The sensor is properly connected to the system
	Power voltage drop error	Voltage drop on the Dupline third wire
	Dupline voltage drop	The Dupline cable is too long or the current consumption is too high, so there is a voltage drop
	Faulty base holder	The base holder of the sensor is damaged
	Faulty push button	Pushbutton on the sensor is always active
	Calibration is requested	The sensor has to be calibrated. *Note: This is highlighted in orange in the Sensor list*
	Faulty sensor element	Ultrasonic sensor is faulty or covered
	Cross talk error	Signal received from other sensors

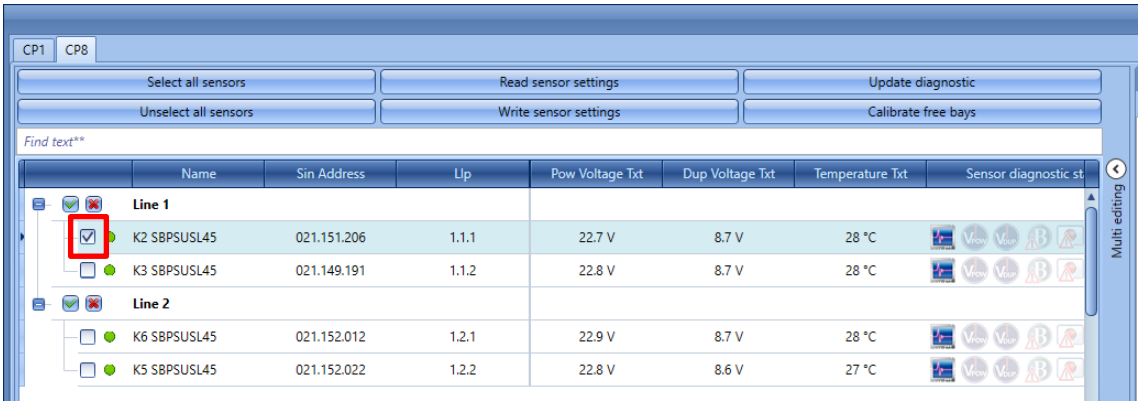
18.4 Graphs

Here we explain how to understand the graph on the right side.

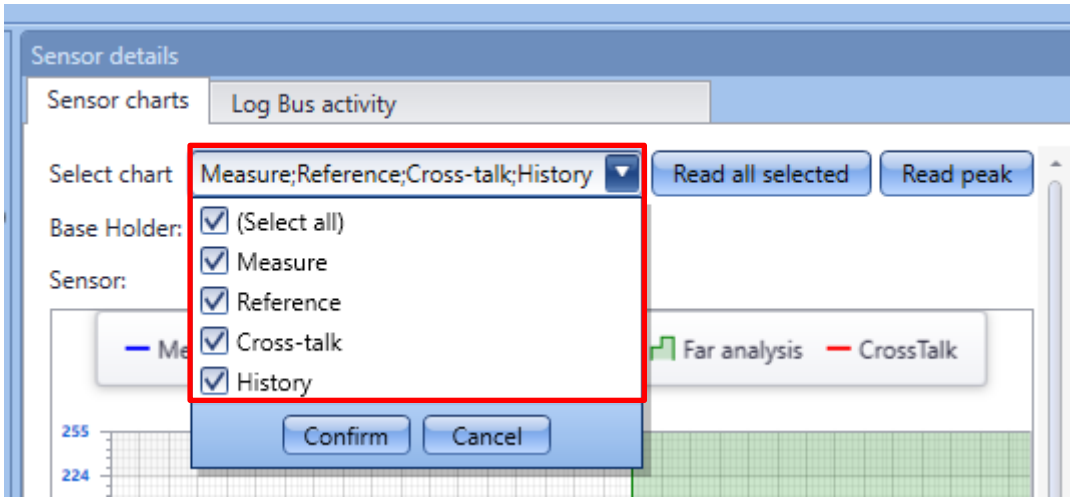
18.4.1 How to generate the graph

For each sensor that has been calibrated, the user can see the graphical representation of the data read by the sensor. The user can use the following procedure in order to generate the graph:

1) In the *Sensor list* of the *Calibration* window, the user has to select the sensors for which the graph has to be generated:



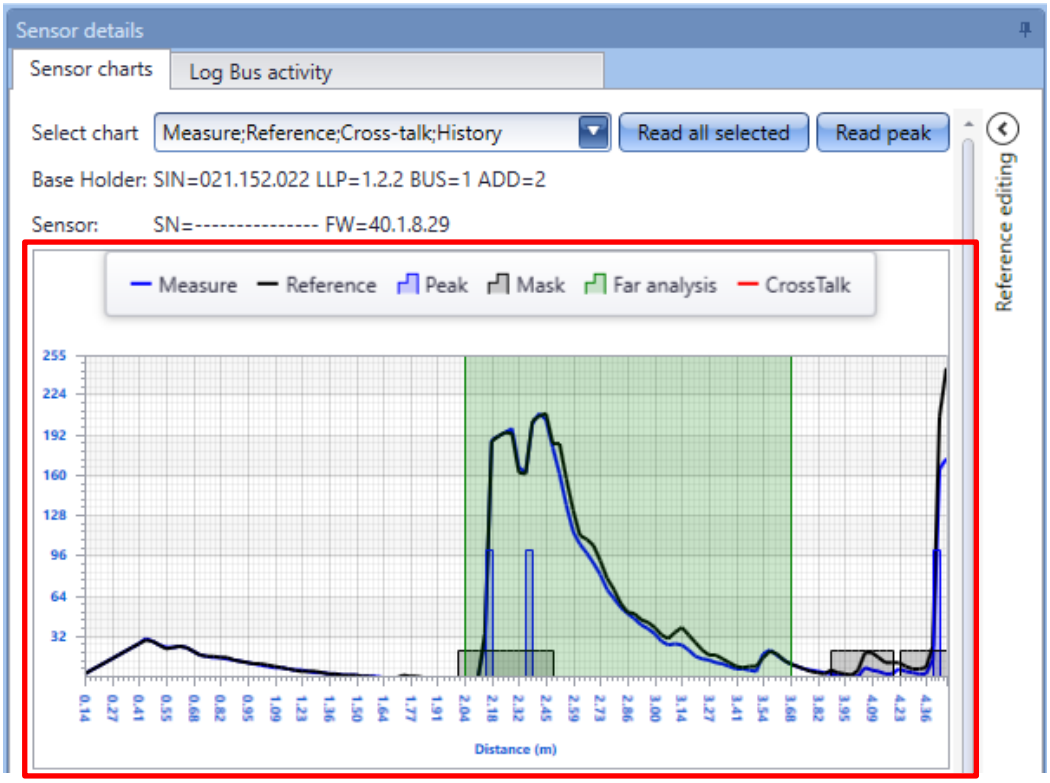
2) In the right side of the window, from the *Select chart* combo-box, the user can select the information which has to be displayed in the graph:



The available options are as shown in the table below:

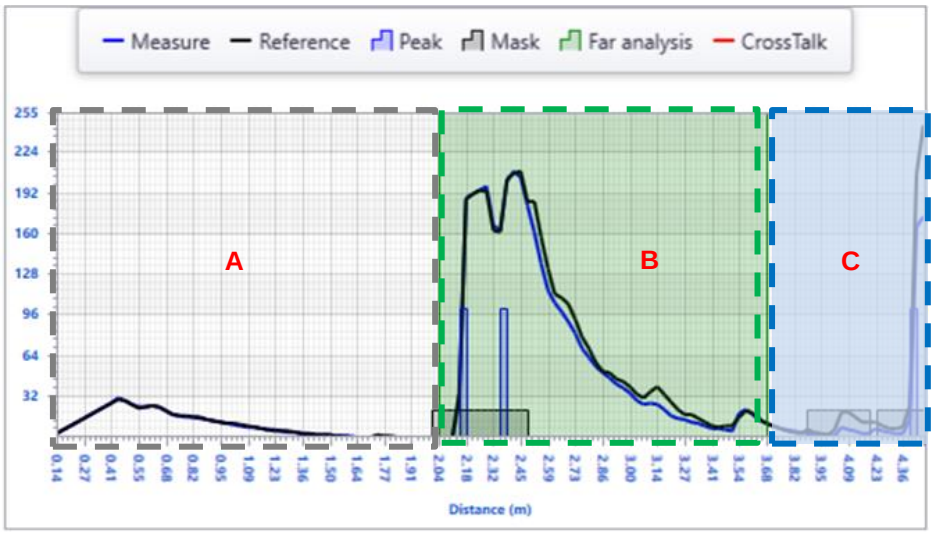
Graph option	Description
Measure	This displays the measured curve shown by the blue line
Reference	This displays the curve generated by the calibration process: this is the reference curve used to define if the parking bay is vacant or occupied (black line)
Cross-talk	This curve (red line) displays a cross-talk condition.
History	This option generates the second graph that is shown in the right panel below the *Sensor details* graph. *See History graph for details*

3) Select the options required and click on the *Read all selected* button to generate this; the system will immediately start the graph generation. As soon as the data is read by the sensor, the graph will be shown, as shown below:



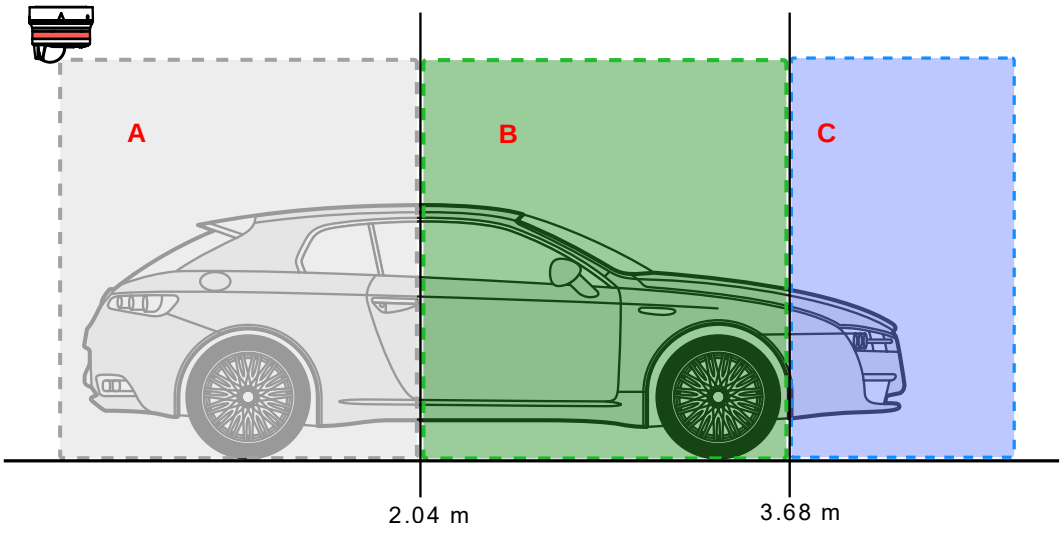
18.5 Things to know - How to read the graph

The Graph area is divided into three parts, as shown below:



- The **white area (A),** is related to the *Near area:* this is the space in between the sensor and the floor, the default value is 2.04 metres. If at least 1 peak is detected in this area it means that a car is parked in the parking bay, therefore the sensor status changes from vacant to occupied;
- The **green area (B)** is related to the *Far area*: this area starts where the *Near area* ends. The default value is 3.68 metres and at least two peaks out of the masks are needed to change the sensor status from vacant to occupied;
- All the obstacles detected by the ultrasonic sensor in the **Far area (C)** will not be considered valid: these peaks usually refer to an area too far from the parking bay area and they will be disregarded;

The figure above shows the three areas from the parking bay point of view



These are the descriptions of the lines and elements displayed in the Graph:

Reference
The black curve is the reference pattern that has been generated by the sensor during the calibration process.

Measure
The blue curve is the last measure carried out by the ultrasonic sensor. Comparing the two curves:

- When the *Reference* and *Measure* curves are very similar, it means that the parking bay is vacant because there are no significant variations between the calibration pattern and the last measure (the parking bay is empty);

- When the differences between the *Reference* and *Measure* curves are significant and moreover at least one peak is detected in the *Near area*, it means a car is present in the parking bay;

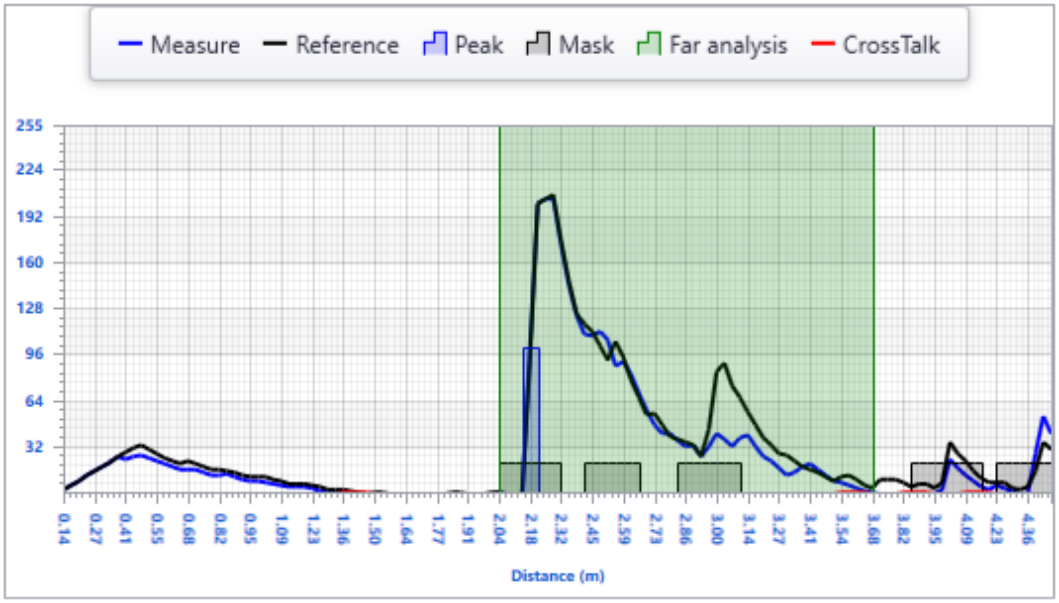
Peaks and Masks
A peak (blue rectangle) is generated each time the ultrasonic beam detects an obstacle**. The calibration must be carried out with no cars and any time structural changes are made in the parking bay,** in order to be sure that all the peaks detected are related to obstacles, such as the floor, beams, conduits, etc…During the calibration process, all the peaks that are detected will be masked in order to consider them not valid.

Reading the Graph, the user can better identify which are the causes related to a calibration issue/Crosstalk condition and how to set the sensor parameters with the best values.

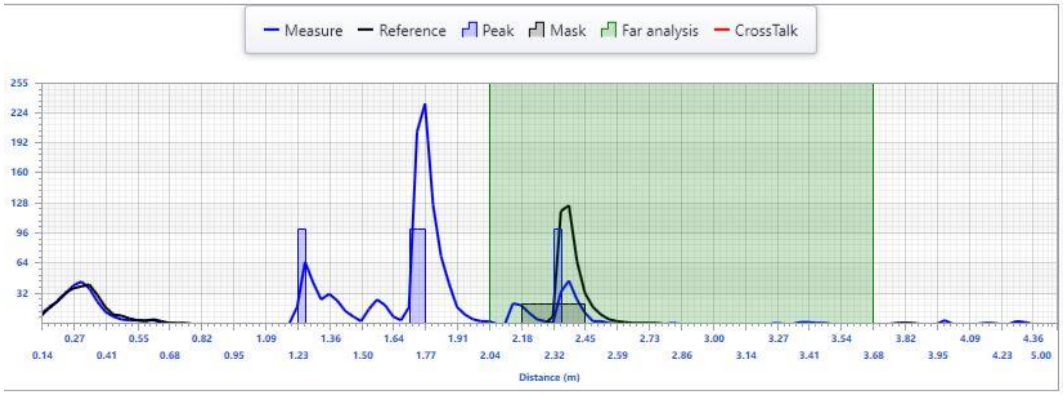
18.5.1 Example: Difference between the vacant and the occupied status

The graphs below show the trend of the same parking bay in vacant and occupied conditions.

***Vacant status*:** The graph shown below is related to the parking bay while it is in vacant condition: there are no peaks in the *Near area* and the *Measure* and *Reference* curves are very similar:



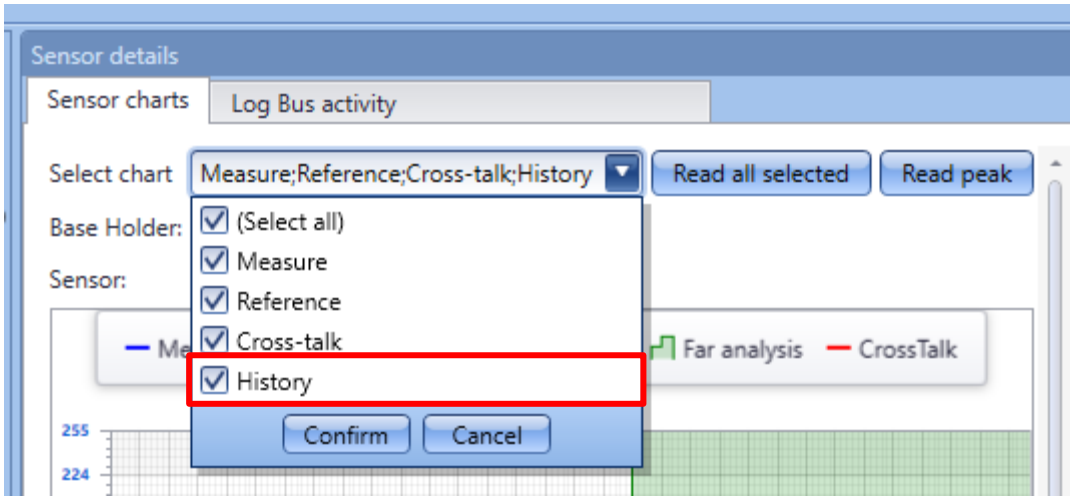
Occupied status: In the graph shown below there are 2 peaks in the *Near area* and the trend of the *Measure* and *Reference* curves are different. The presence of peaks in the *Near area* means a car is parked in the parking bay.



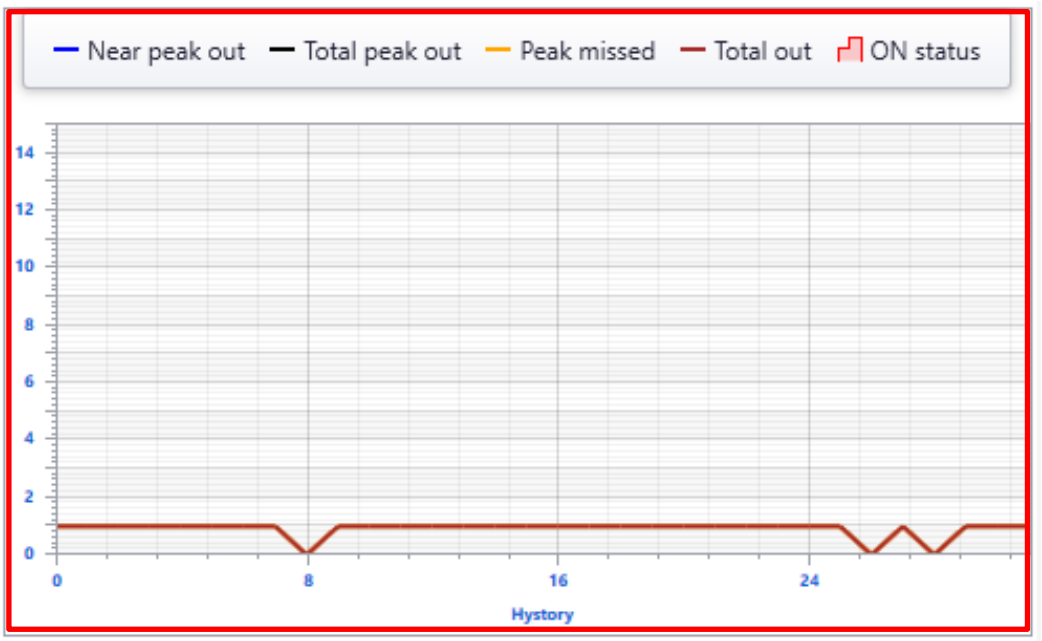
18.5.2 How to generate the History graph

The *History graph* shows the graphical representation of the last 32 measures carried out by the sensor. To generate the *History* graph the user can follow the procedure shown below:

1) In the *Select chart* combo-box the user has to select the *History* option, as shown in the red rectangle below:



2) Click on the *Read all selected* button; the system will immediately start the graph generation. As soon as the last 32 measures are collected, the graph will be presented:



18.6 Things to know - How to read the History graph

The *History graph* shows the history trend of the parking bay: the last 32 measures carried out by the sensors are shown: there are two graphs, the first one refers to the number of peaks detected in the latest 32 measures, the second one represents the measure of the area subtended by the graph

Peaks graph

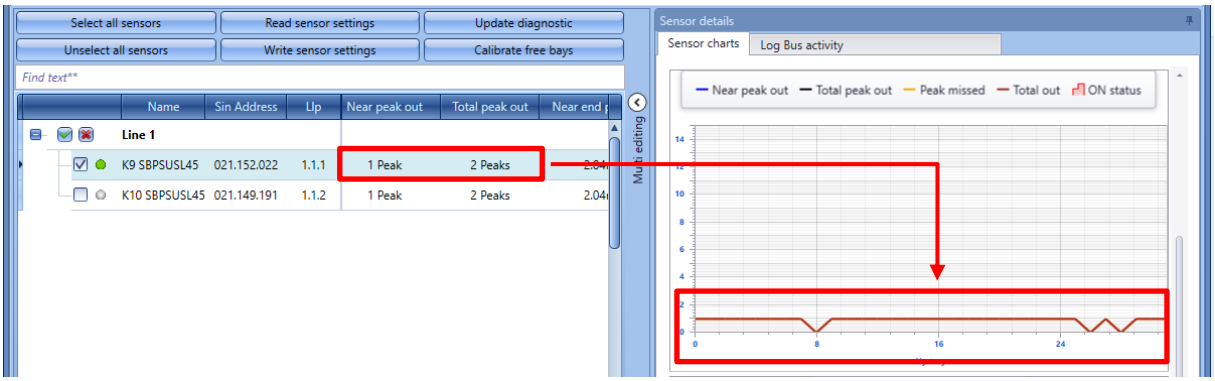
Near peak out
Whenever the value of the *Near peak out* field exceeds the value set in the *Sensor list* for the selected sensor, the parking bay status changes from vacant to occupied;

Total peak out
This fields defines the total peak that is present on both the white and green areas.
Whenever the value of the *Total peak out* field exceeds the value set in the *Sensor list* for the selected sensor, the parking bay status changes from vacant to occupied;

ON status
Whenever the parking bay turns to occupied status, the *History chart* area will be highlighted in red;

18.6.1 Example: Difference between the vacant and occupied condition

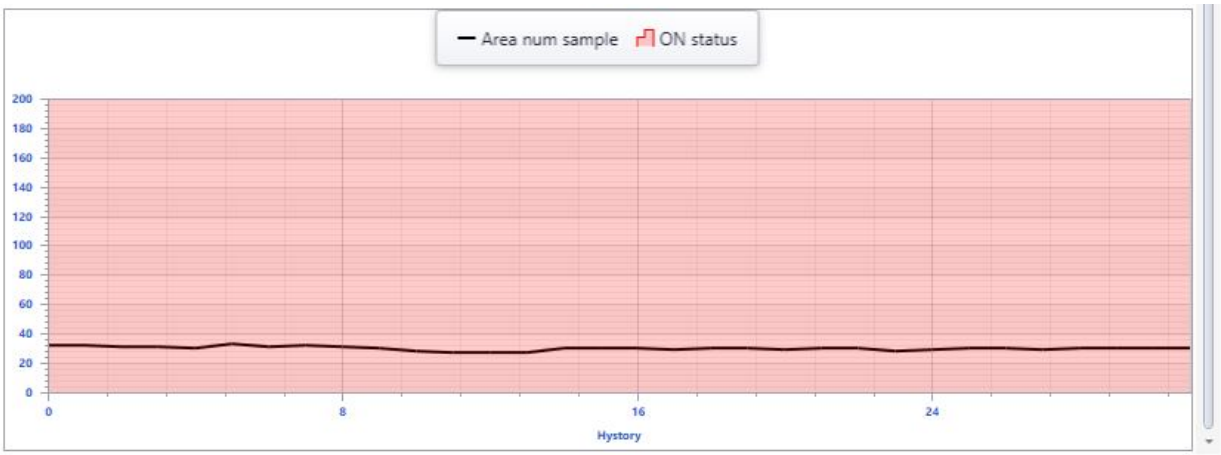
The picture below shows the last 32 measures carried out by a sensor while the parking bay is empty: both the *Near peak out* and the *Total peak out* lines remain below the minimum required value to change the parking bay from vacant to occupied. Moreover, the *Sensor row* of the *Calibration* window is not highlighted in red.



The picture below shows the last 32 measures carried out by a sensor while the parking bay is occupied: The *Near peak out* and the *Total peak out* lines are above the minimum required value to change the parking bay from vacant to occupied. Moreover, both the History area and the Sensor row in the *Calibration* window are highlighted in red.

Area graph

A calculation of the area subtended by the graph is made to compare the reference waveform to the measured one: if the value of the coefficient on the chart is more than 25 the sensors is set to occupied.
The picture below shows the last 32 measures carried out by a sensor while the parking bay is empty: the area measurement is 0.
If the bay is occupied, the graph shows the last 32 measures that are over 0.



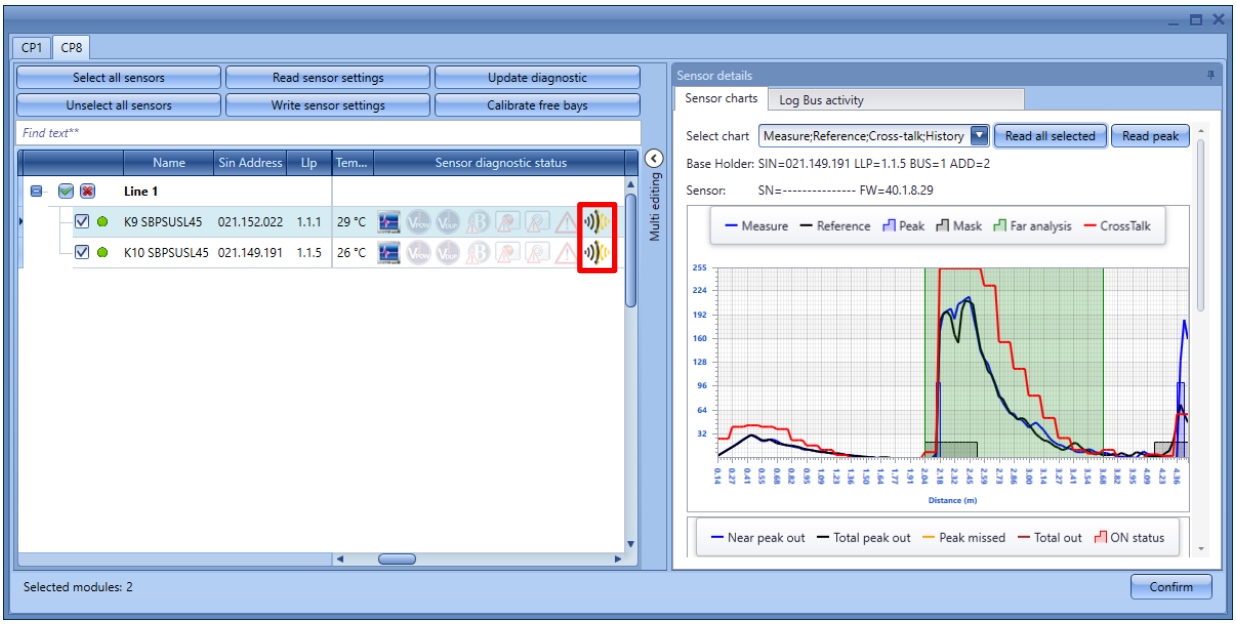
18.7 Crosstalk

This part of the manual deals with common Cross talk problems the user can encounter during the project configuration and presents some solutions.

N.B: This issue should be resolved before continuing to the project development

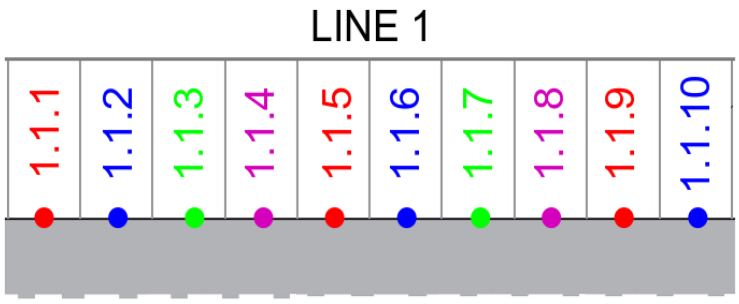
18.7.1 How to identify if a cross talk condition is present

Whenever a Cross talk condition affects one or more sensors that are present in the configuration, the Cross talk icon is displayed in the Sensor diagnostic status field of the *Calibration* window, for each sensor. As shown in the red rectangle below, the sensors that have LLP 1.1.1 and 1.1.5 have a crosstalk condition. Moreover, the Cross talk option can be selected from the *Select chart* combo-box: the Crosstalk will be shown with a red line (if present) of the selected sensor.



18.7.2 Things to know – Crosstalk problem

A major issue with using ultrasonic sensors in a Parking lot where several LINEs are present with many sensors installed one close to the other, is a type of interference referred to as Crosstalk. Crosstalk occurs when two (or more) nearby ultrasonic sensors receive the signal of another sensor: If the sensors are pointed directly at each other, and within each other's detection zones, there is a good chance of crosstalk happening. In order to avoid this situation, the sensors will not work simultaneously, but they will be activated by the system using four different time intervals, as described in the example below:

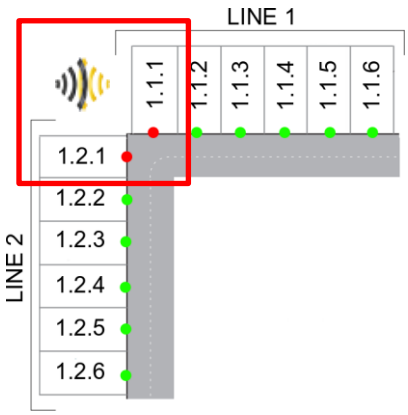
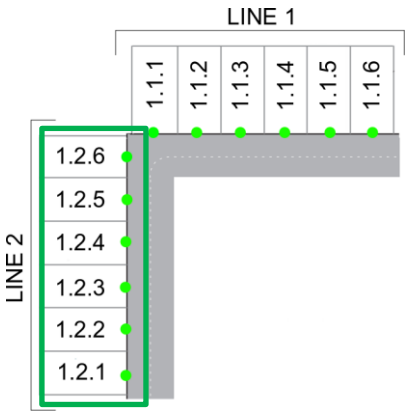
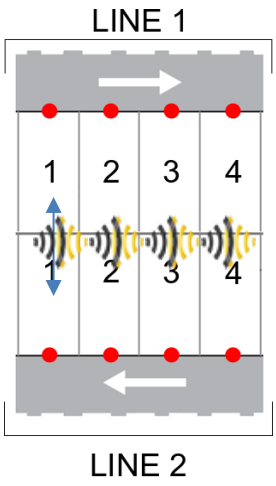
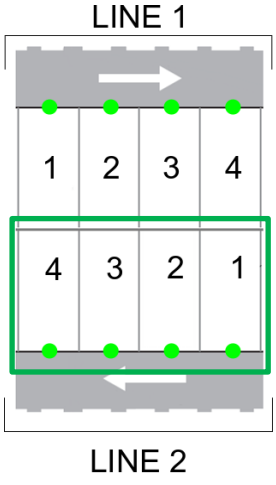


- Slot1: The *Sensor* that has *LLP 1.1.1* will emit/receive the signal at the same time as *Sensor 1.1.5* and *Sensor 1.1.9*;
- Slot2: The *Sensor 1.1.2* will emit/receive the signal at the same time as *Sensor 1.1.6* and Sensor *1.1.10;*

- Slot3: The *Sensor 1.1.3* will emit/receive the signal at the same time as *Sensor 1.1.7;*

This feature should ensure that *Sensor 1* will not be interfered with by *Sensor 2*, also *Sensor 2* will not be affected by *Sensor 1* and *Sensor 3*. *Sensor 1* and *Sensor 5* are too far from each other to be disturbed. Nevertheless, there could be situations where cross talk could be present in the environment.

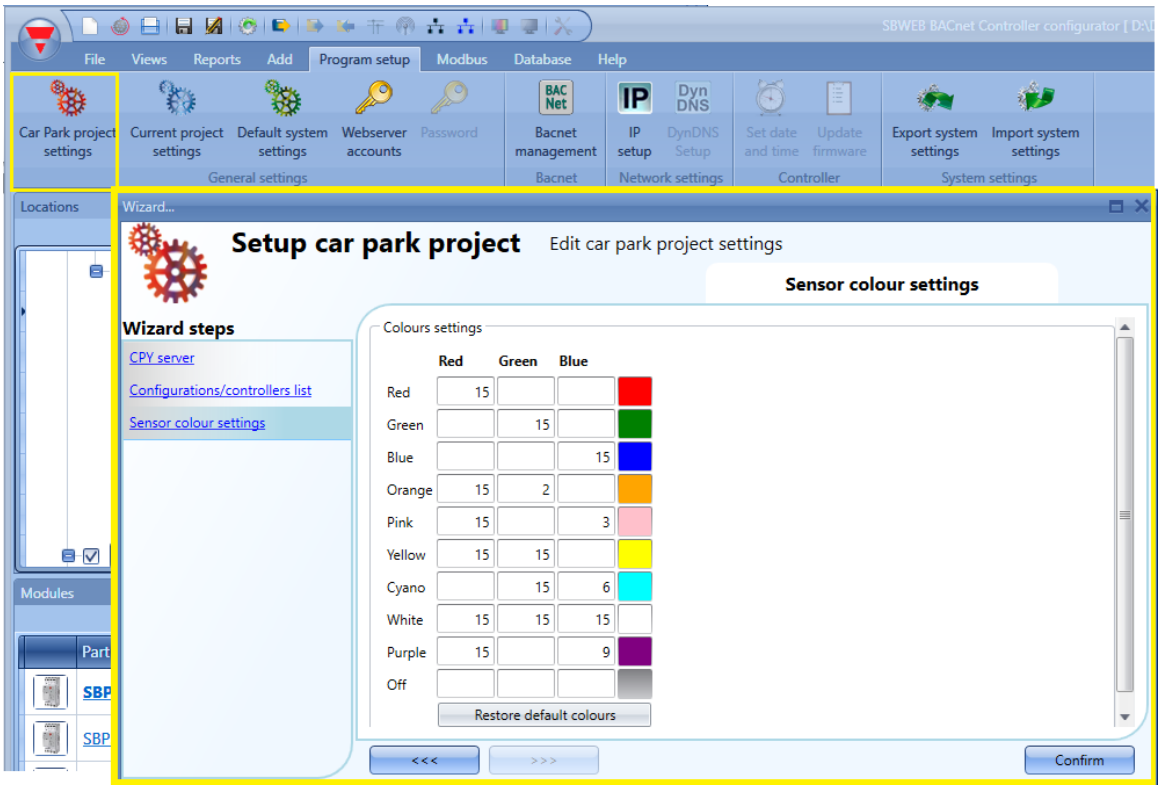
The table below shows the common scenarios where Cross talk may happen and the suggested solution:

Condition	Solution
There is a good chance of Cross talk happening for the sensors that have LLP 1.1.1 and 1.2.1 if they are mounted in a situation as shown below: 	In order to solve the problem, LLP addressing of the sensors that belong to the LINE2 has to be changed: the user may re-address the LLP addresses as shown in the green rectangle below: 
In the situation shown below, where the sensors are pointed directly at each other, and within the detection zones of either sensor, there is a good chance of Cross talk happening. The sensors have different LINE but the same POSITION address. 	In order to solve the problem, the LLP addressing of the sensors that belong to the LINE2 has to be changed: the user may re-address the LLP addresses as shown in the green rectangle below: 

19 Appendix

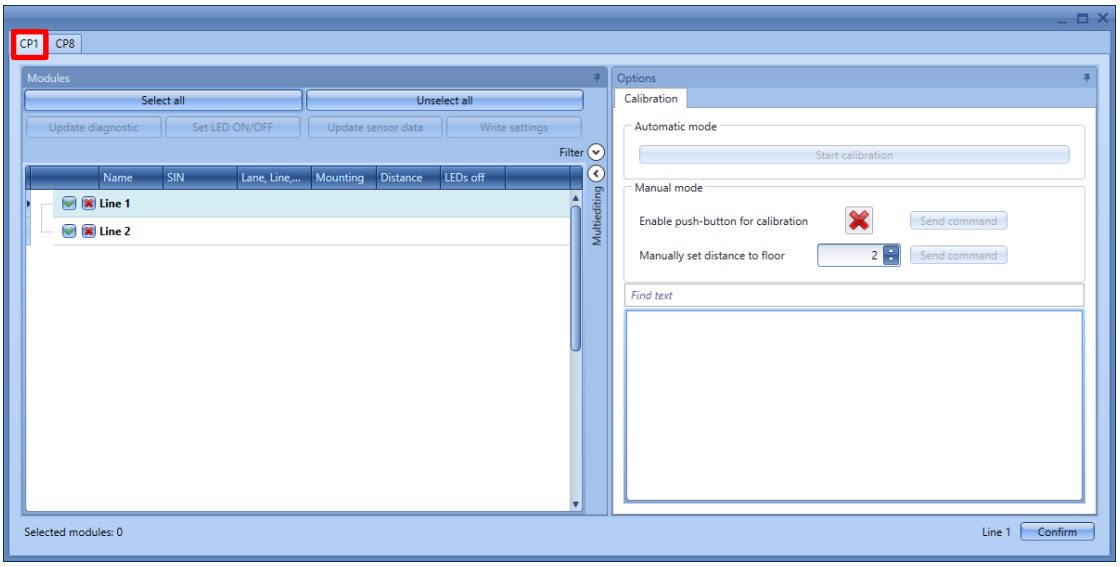
19.1 How to define the colours of the LED in the sensors SBPSUSLxx

The sensors SBPSUSLxx have an RGB LED for which the available colours have to be defined in the menu C*ar park project settings.*
These available colours will then be associated to the different types of categories the sensor indicates, such as vacant, engaged, for VIPs, for expecting mothers, etc. The association is carried out by means of the CPY server (see CPY server manual).



19.2 How to calibrate the sensors with firmware release from 1 to 7

The user has to click on the CP1 tab: the following window will appear:



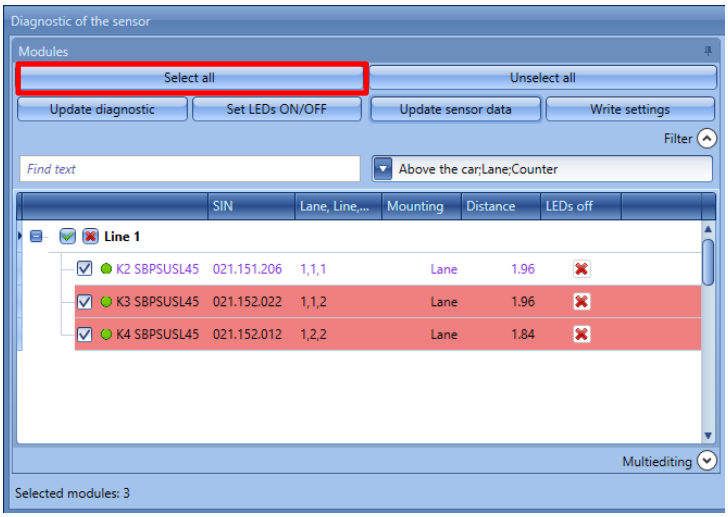
- The occupied sensors are marked in red
- The unoccupied sensors are marked in white

For each sensor, the type and distance are shown.

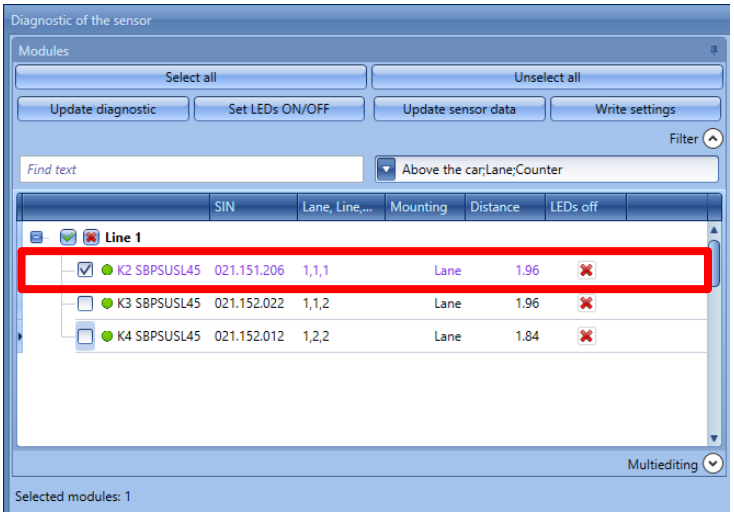
19.3 Step 1: Select the modules

To select the modules:

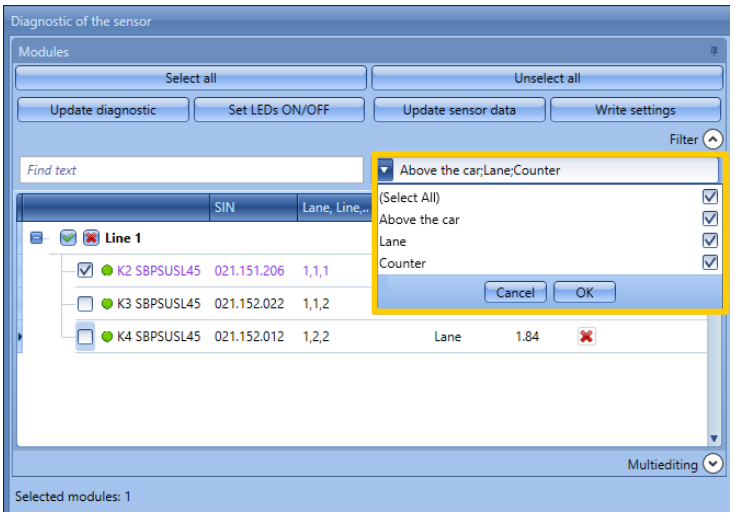
a. Click on *Select all* to calibrate all the sensors



b. Check the sensors one by one



c. Select the sensors according to the type

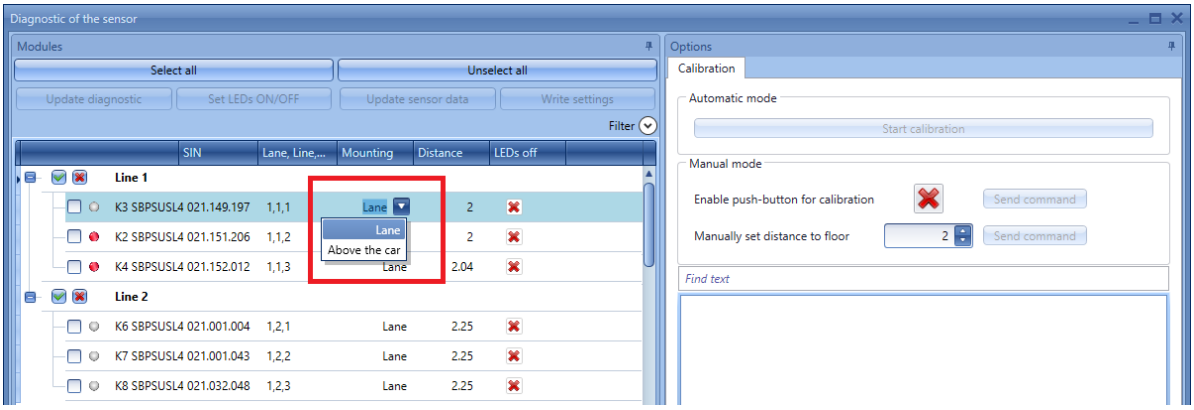


It is strongly suggested to calibrate not more than 20 sensors at the same time.

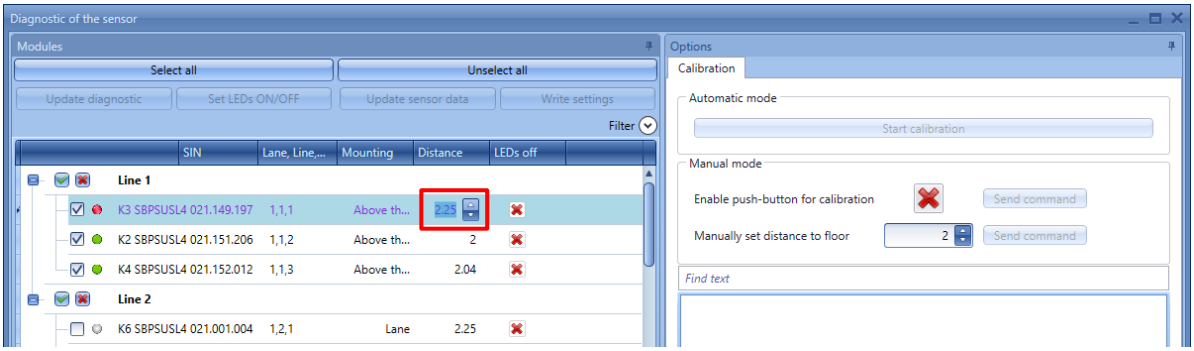
19.4 Step 2: Write the distance from the floor and the type of mounting

19.4.1 How to write different settings to each single sensor

1) Select the type of mounting for each sensor: *Lane* if it is mounted in the lane or *Above the car* if it is mounted above the parking bay.

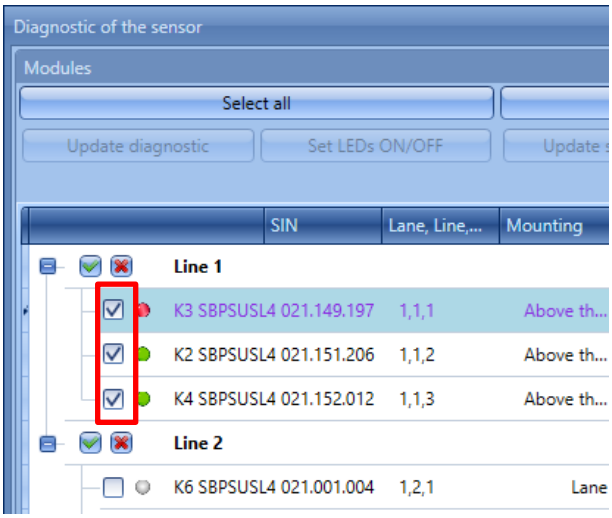


2) Set the distance from the floor in the column *Distance* for each selected sensor and then click on *Write settings*

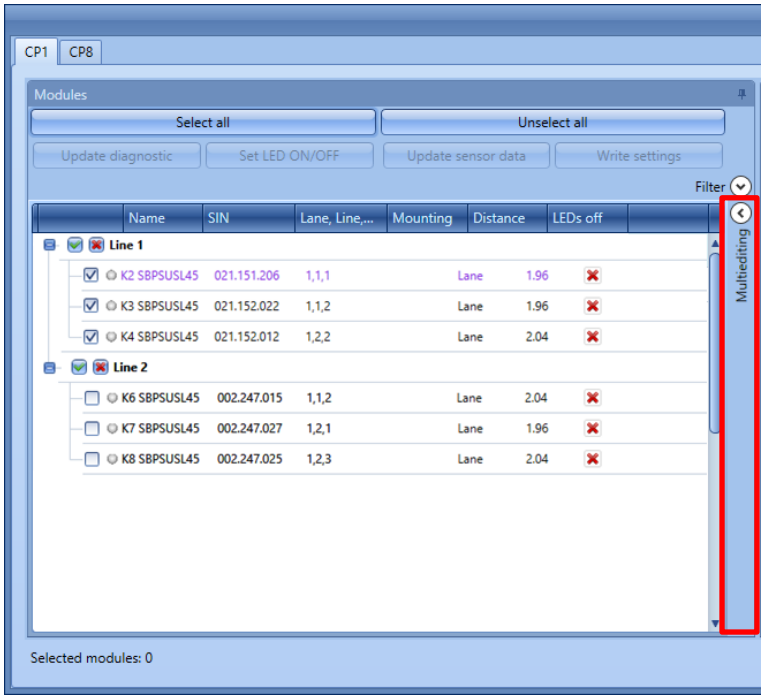


19.4.2 How to write the same distances and mounting ways into the selected sensors

1) Select the sensors that have to be managed by checking the check box next to each one:

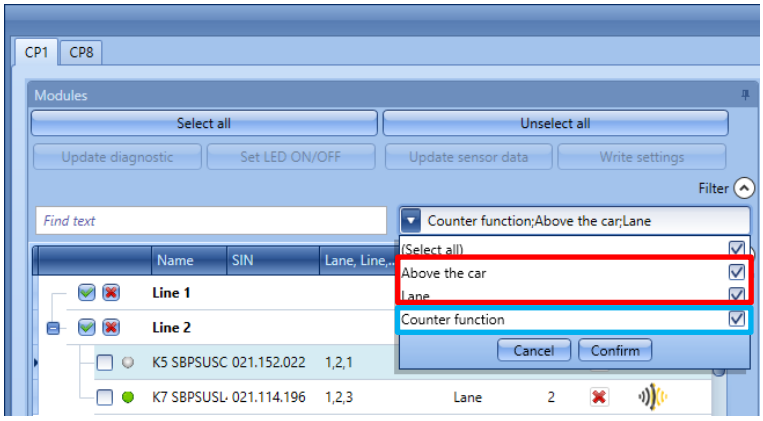


2) Open the *Multi editing* panel in the right part of the window by clicking on the < button:



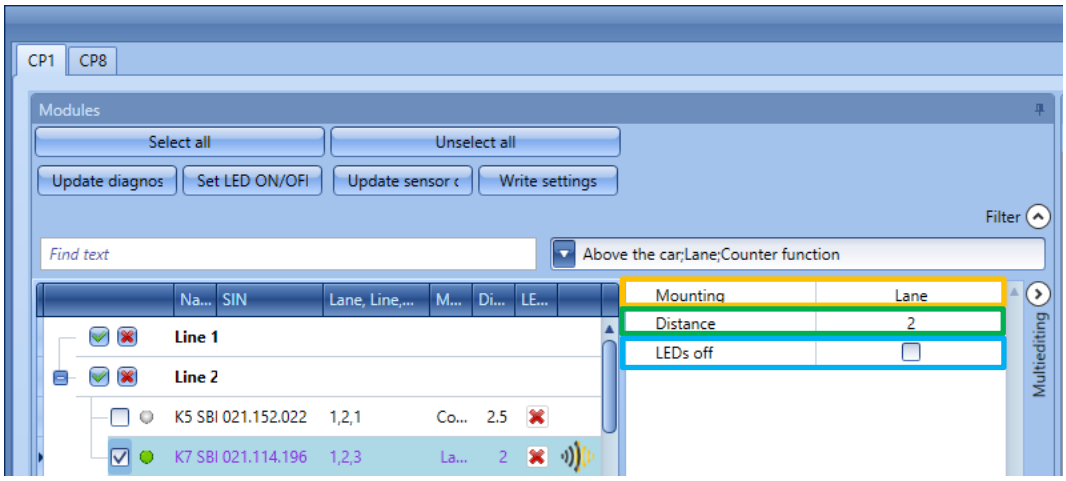
3) From the combo-box field of the *Filter* panel, the user can select the sensors that have to be managed by model, as shown below:

   - The *Above the car* and *Lane* options has to be selected for the SBPSUSLxx sensors available in the configuration;
   - The *Counter function* option has to be selected for the SBPSUSCNT indoor counter module only;

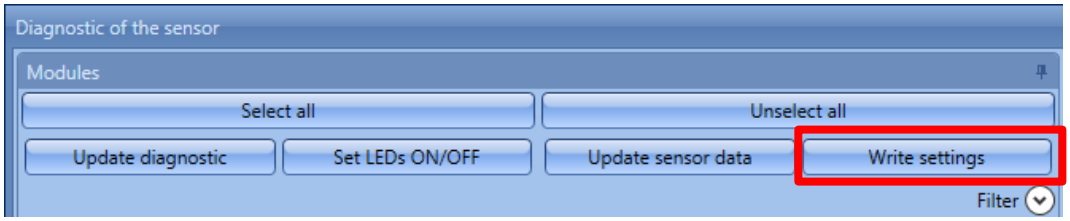


4) Set the type of mounting, the distance and the LED status that have to be applied to all the selected sensors:

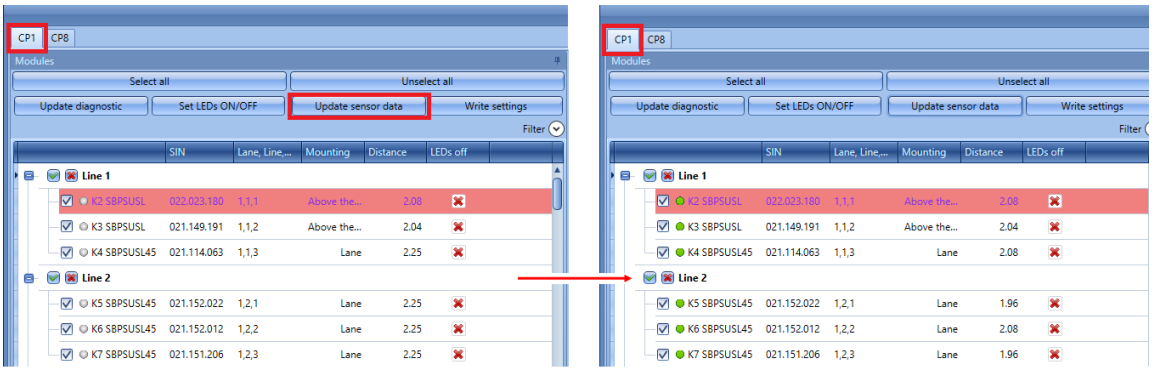
   - In the Mounting combo-box the user can select Above the car or Lane options ;
   - In the Distance field the user can enter the Distance;
   - Click the small red cross to disable the LEDs lit; When the small red cross is selected the LEDs of the sensors are enabled:



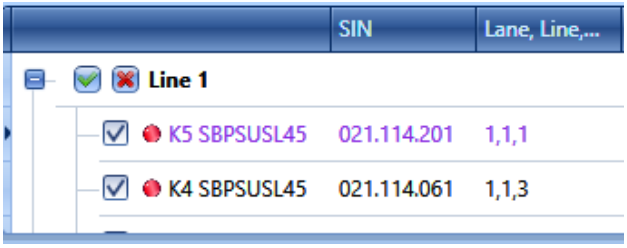
5) To save the changes, the user has to click on the *Write settings* button, as shown in the picture below:



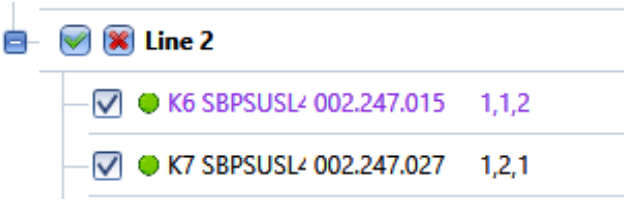
The user can read the configuration of the sensors by clicking on the *Update sensor data* button**:** the system will read the configuration of all the selected sensors, the dot next to the sensor name will turn to green colour, as shown in the example below:



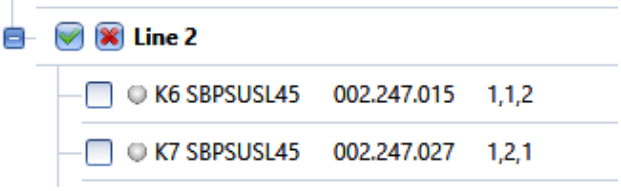
If the settings in the UWP 3.0 Tool and the one in the sensor are consistent, the dot next to the sensor part number is green, otherwise it is red.



Data need to be synchronized.



Data are synchronized



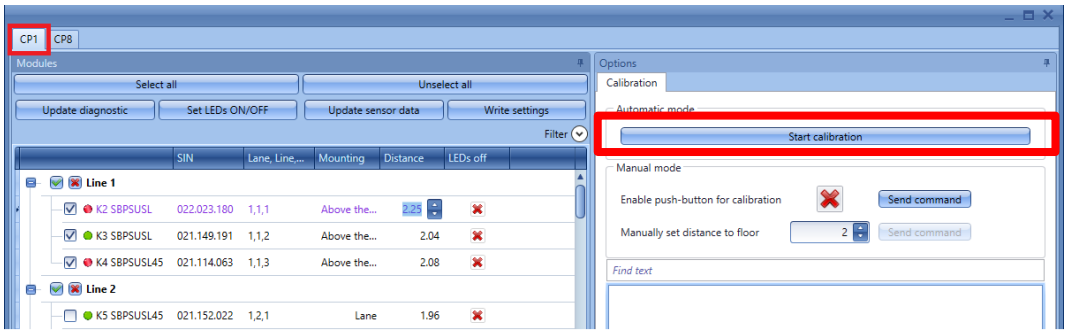
Data are not synchronized

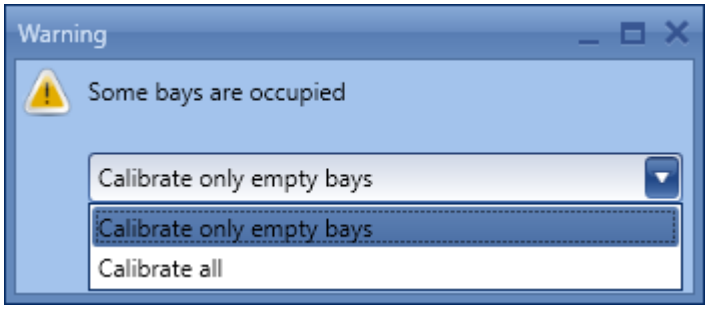
19.5 Step 3: Calibrate the sensors

There are two ways to calibrate the sensors: launching the calibration commands remotely from the tool and locally by means of the push–button on the sensor.

19.5.1 Remote calibration

Once the distance has been written, click on *Start calibration:* a window will appear asking which sensors have to be calibrated.





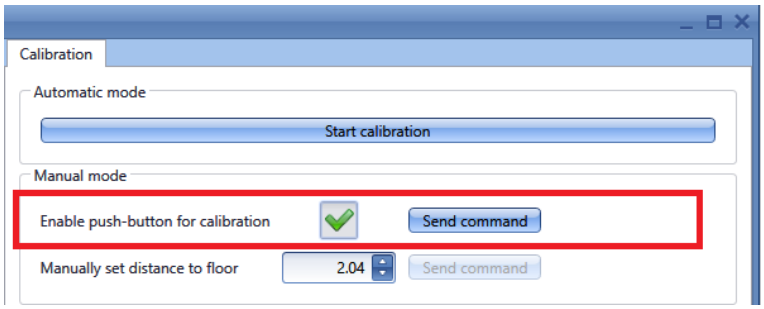
Calibrate only empty bays: if this option is selected, the system will calibrate only the unoccupied spaces

Calibrate all: if this option is selected, the system will calibrate all the sensors

Once the choice has been made, click on *Confirm*: an automatic process will start for the selected sensors, without needing to do anything.

19.5.2 Local calibration

a) Enable the push button on the sensor and click on *Send command.*



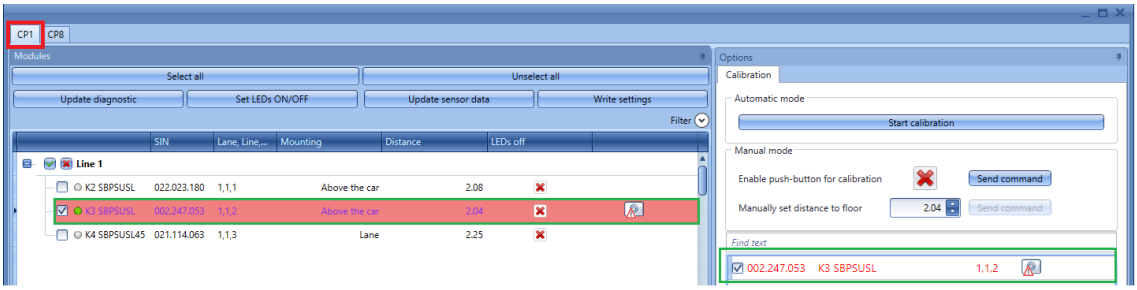
b) Start the calibration by walking from sensor to sensor and pressing the buttons: the yellow LED flashes slowly for 15 seconds to let the installer arrive and then it flashes fast when ready to be calibrated.

c) Once the calibration process is completed, we suggest disabling the push-buttons so that non-authorised people will not use them improperly.

If the settings *Above the car* is selected, the distance should not be set because it is over-written by the calibration process.

19.5.3 Diagnostic signals

To update the status of the sensors, click on the *Update diagnostic* button*:* the software will read the diagnostic status of the selected modules.



The diagnostic signals that can be detected by the system are the following:

Icon	Tool tip	Description
	Cross talk error	Signal received from other sensors
	Faulty sensor element	Ultrasonic sensor is faulty or covered
	Faulty pushbutton	Pushbutton on the sensor is always active
	Faulty base holder	The base holder of the sensor is damaged
	Dupline voltage drop	The Dupline cable is too long or the current consumption is too high, so there is a voltage drop
	Power voltage drop error	Voltage drop on the Dupline third wire