# iC Tool

User Manual

## Programming Tool

# Table of Contents

# 1 Introduction

The iC Tool is a free computer program provided by the iSMA CONTROLLI.

As a significant part of an end-to-end solution, the iC Tool is a native programming tool for the nano EDGE ENGINE and Sedona based devices. The iC Tool covers all requirements to create and manage applications: it has a wire sheet for convenient visual programming, property sheets for details; it offers libraries management, real-time monitoring of system states and slots values, logs, and deployment.

## 1.1 Revision History

| Rev. | Date | Description |
|---|---|---|
| 1.4.1 | 27 Nov 2023 | • Adding External Kits section updated |
| 1.4.0 | 10 Nov 2023 | • Support for the nano EDGE ENGINE V1.4<br>◦ Special Events view<br>◦ Calendar view<br>◦ Migration mechanism |
| 1.2.0 | 13 Apr 2023 | • Support for the nano EDGE ENGINE V1.2<br>◦ Weekly Schedule view<br>• Support for the Sedona devices<br>• Editorial corrections |
| 1.1.0 | 31 Aug 2022 | • Support for the nano EDGE ENGINE V1.1<br>• Editorial corrections |
| 1.0.0 | 28 Feb 2022 | First edition |

*Table 1. Revision history*

## 1.2 About

The iC Tool allows programming of applications and parameter setting in the nano EDGE ENGINE and Sedona based devices. Using proper functions implemented in the iC Tool, the user can program his own application from scratch or edit an existing application working directly on a connected device.

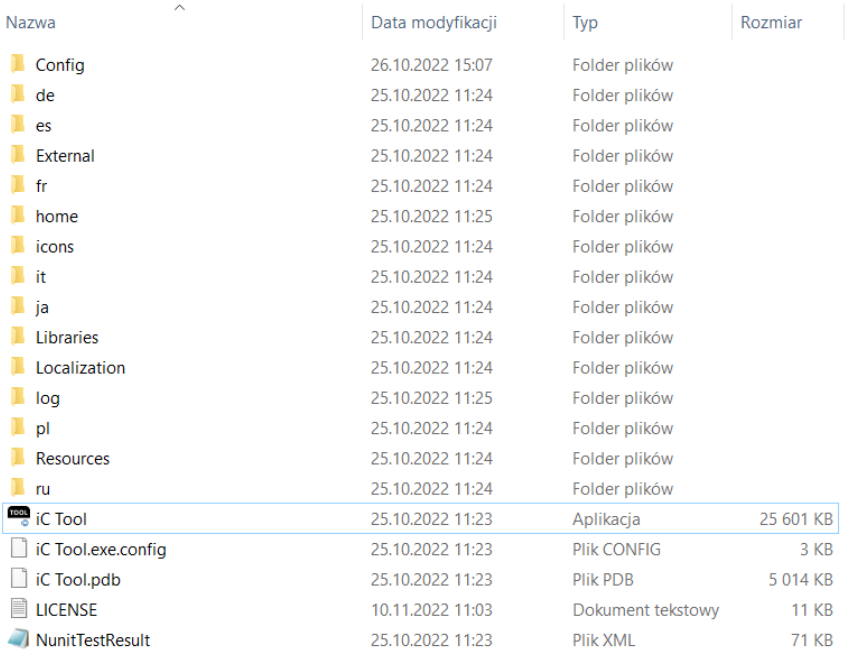
The iC Tool is addressed to every user working with the device on the daily basis, i.e., distributors, integrators, engineers, and maintenance technicians.

# 2 Installation and Updates

## 2.1 iC Tool Installation

The iC Tool is delivered as a compressed folder, which needs to be extracted in a chosen location on a hard drive, unless the access to the extracted folder is restricted by the system (e.g., Program Files is not a recommended location). The iC Tool is a software created for modern Microsoft Windows system, such as **Windows 10**. The oldest supported version of the operating system is Windows 7.

The iC Tool software bundle is a zipped file, which has to be extracted on the user's computer. Extracting the zipped package reveals the folders and additional files described below. In order to run the iC Tool, open the iCTool.exe file.



| Nazwa | Data modyfikacji | Typ | Rozmiar |
|---|---|---|---|
| Config | 26.10.2022 15:07 | Folder plików | |
| de | 25.10.2022 11:24 | Folder plików | |
| es | 25.10.2022 11:24 | Folder plików | |
| External | 25.10.2022 11:24 | Folder plików | |
| fr | 25.10.2022 11:24 | Folder plików | |
| home | 25.10.2022 11:25 | Folder plików | |
| icons | 25.10.2022 11:24 | Folder plików | |
| it | 25.10.2022 11:24 | Folder plików | |
| ja | 25.10.2022 11:24 | Folder plików | |
| Libraries | 25.10.2022 11:24 | Folder plików | |
| Localization | 25.10.2022 11:24 | Folder plików | |
| log | 25.10.2022 11:25 | Folder plików | |
| pl | 25.10.2022 11:24 | Folder plików | |
| Resources | 25.10.2022 11:24 | Folder plików | |
| ru | 25.10.2022 11:24 | Folder plików | |
| iC Tool | 25.10.2022 11:23 | Aplikacja | 25 601 KB |
| iC Tool.exe.config | 25.10.2022 11:23 | Plik CONFIG | 3 KB |
| iC Tool.pdb | 25.10.2022 11:23 | Plik PDB | 5 014 KB |
| LICENSE | 10.11.2022 11:03 | Dokument tekstowy | 11 KB |
| NunitTestResult | 25.10.2022 11:23 | Plik XML | 71 KB |

*Figure 1. The iC Tool zipped folder*

The extracted folders have the following functions:

- **Config:** a folder containing a record of user's individual settings regarding windows location and other iC Tool work settings, such as a language chosen for the iC Tool interface;
- **External**: a folder containing an API .dll file;
- **home:** a folder where all the data created by user are saved, i.e., device backups, applications, etc. It is also a folder where the OS and libraries, available the in iC Tool, are located;
- **icons:** a folder with graphical files such as the iC Tool interface icons;
- **Libraries:** a folder containing iC Tool auxiliary files;
- **Localization:** a folder with the text files providing the iC Tool language sources;
- **log:** a folder, where the logs of the iC Tool, which also appear in the Console window, are saved. When contacting iSMA CONTROLLI technical support, it is advised to copy the file with logs from this folder;
- **de, es, fr, it, ja, pl, Resources, ru:** folders with system libraries.

To properly install and work with the iC Tool, the computer must meet the following minimal requirements:

- processor (CPU): Intel Core i3-3xxx or equivalent;
- memory: 4 GB RAM;
- storage: 50 GB internal hard driver;
- Ethernet 100 Mbit or 1 Gbit NIC;
- MS Windows 7 (recommended MS Windows 10);
- .NET Framework 4.6.2 or higher.

**WARNING!**

When the iC Tool is being run for the first time, it asks to accept the EULA license. The license must be accepted to run the program. Failure to do so closes the iC Tool.

**Worth to Notice**

In order to make sure that the iC Tool is always up to date, it needs to be run periodically at least once a month, on a computer connected to the Internet for about an hour, depending on the data transfer rate. It enables the iC Tool to automatically download the latest data, such as libraries and updates.

The iC Tool is a portable software. It is transferable and it can be installed on a portable data storage device, such as a USB memory stick. It allows the iC Tool to be run directly from a portable data storage device on any PC, including offline ones.

## 2.2 iC Tool Updating

The iC Tool has a built-in mechanism notifying the user when a new version of software is available on the iSMA CONTROLLI server. Every time the iC Tool starts up and it is connected to the Internet, the software checks the availability of a new version. If the new version is available, a pop-up widow, similar to the one shown below, will appear.

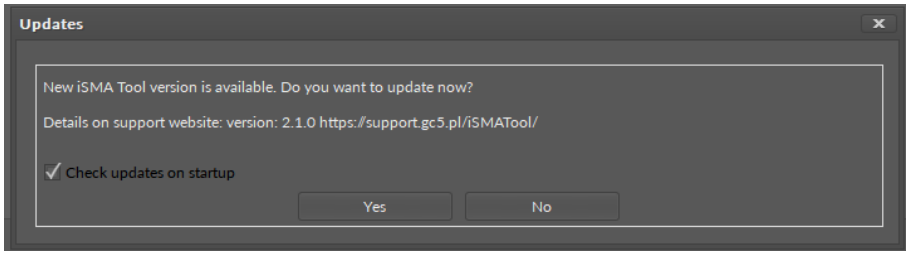



*Figure 2. Update notice*

The options in the Updates window are the following:

- Yes: Allows downloading a new version of the iC Tool by opening an appropriate location in the user's default web browser. The new version of the iC Tool is downloaded as a zipped file saved locally on the computer's hard drive and then extracted, overwriting the current iC Tool version.
- No: Closes the iC Tool updates notification window. The window will appear again after restarting the iC Tool.

**WARNING!**

Before overwriting the current version of the iC Tool, the software needs to be closed and the user has to make sure that all important data (libraries, applications, settings) are saved in dedicated folders, home and Config. These folders will not be overwritten or deleted, while extracting/overwriting the current iC Tool version with an updated one, preserving everything the user has created.

The automatic checking mechanism, described above, may be turned on or off in the iC Tool settings described in Settings. When the checking mechanism is turned on and there are no new versions of the iC Tool, the system will display an appropriate notification: No updates for iC Tool on the console.

The autoupdate function covers the following areas:

- iC Tool software;
- nano EDGE ENGINE OS;
- nano EDGE ENGINE libraries.

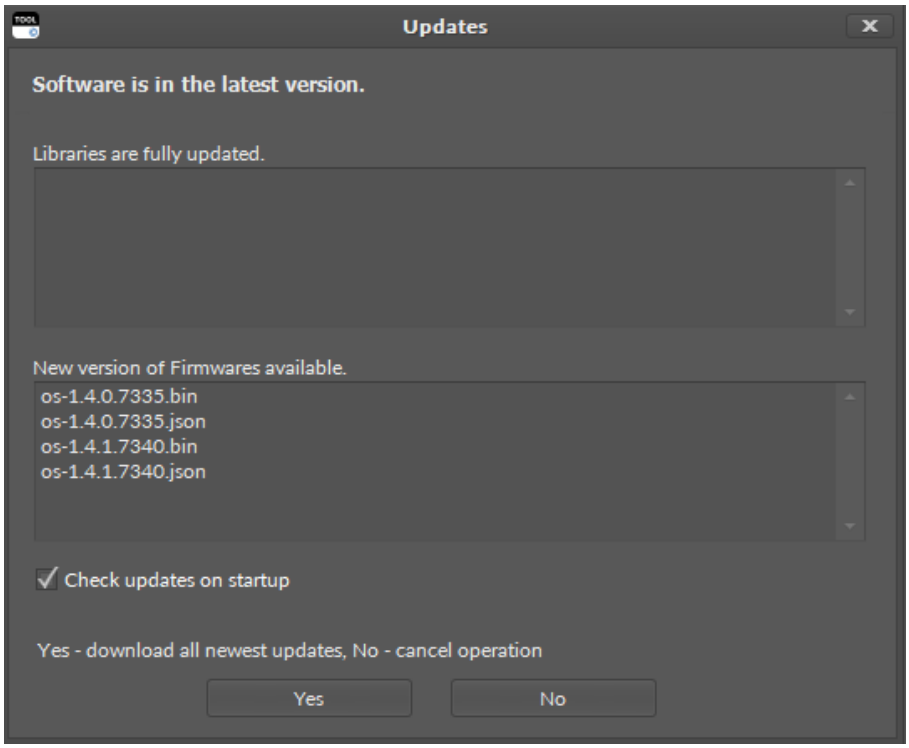



*Figure 3. The Autoupdate dialog window*

There is also a manual way of checking software updates availability. In order to do so, initiate the menu Check updates, available in upper Help menu. After initiating this option, the iC Tool will check the availability of new versions and display an appropriate notification. If a new update is available, a pop-up window, described above, will appear.

## 2.3 Migration Mechanism

The migration mechanism in the iC Tool allows to import files from another instance of the program.

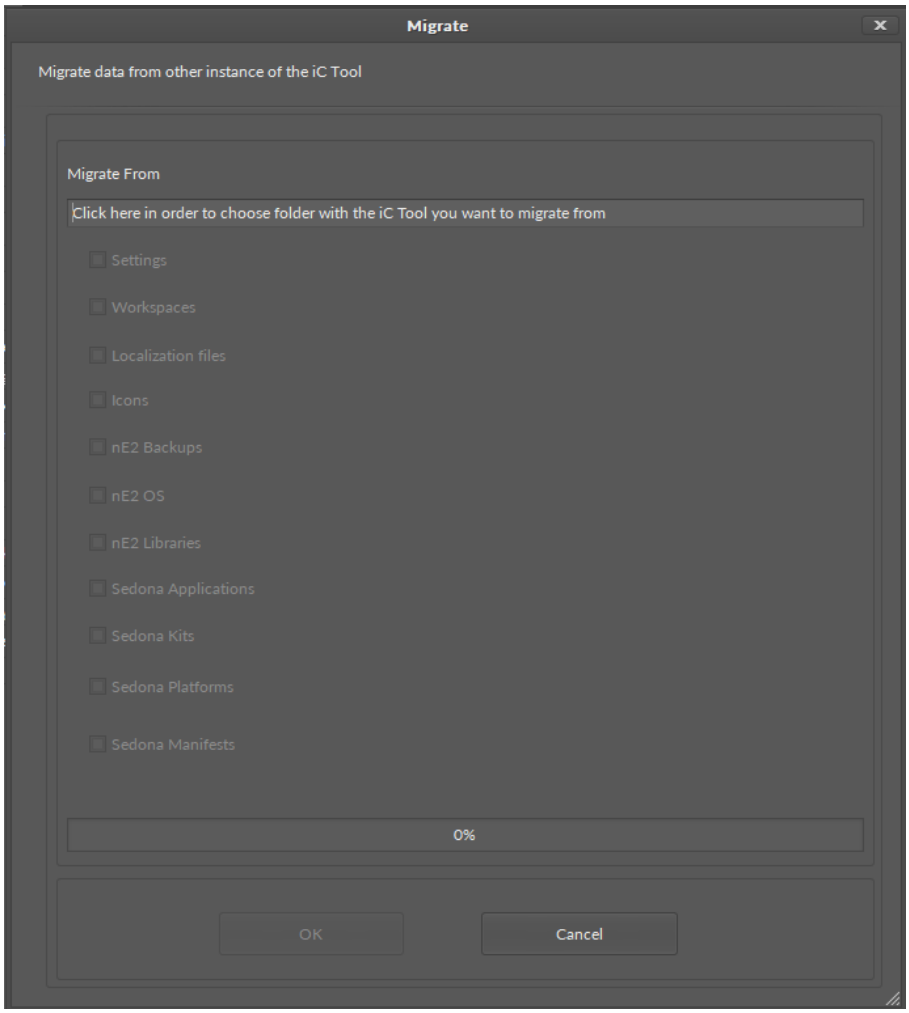



*Figure 4. The Migration dialog window*

Depending on which files are available in the migration source folder, the migration mechanism allows to import the following files:

- iC Tool's settings;
- Workspaces - sites, devices, and their applications and settings;
- localization files (language settings);
- icons;
- nano EDGE ENGINE OS, libraries, and bakcups;
- Sedona applications, kits, platforms, and manifests.

At a first launch of the iC Tool, the migration dialog window appears automatically. Later, the option can be accessed in the main menu, Import.

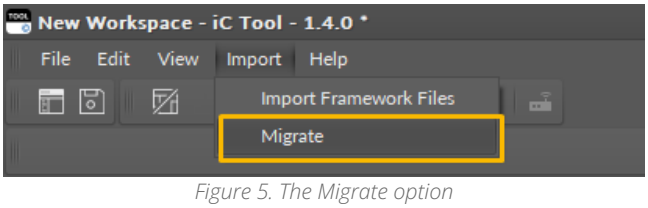
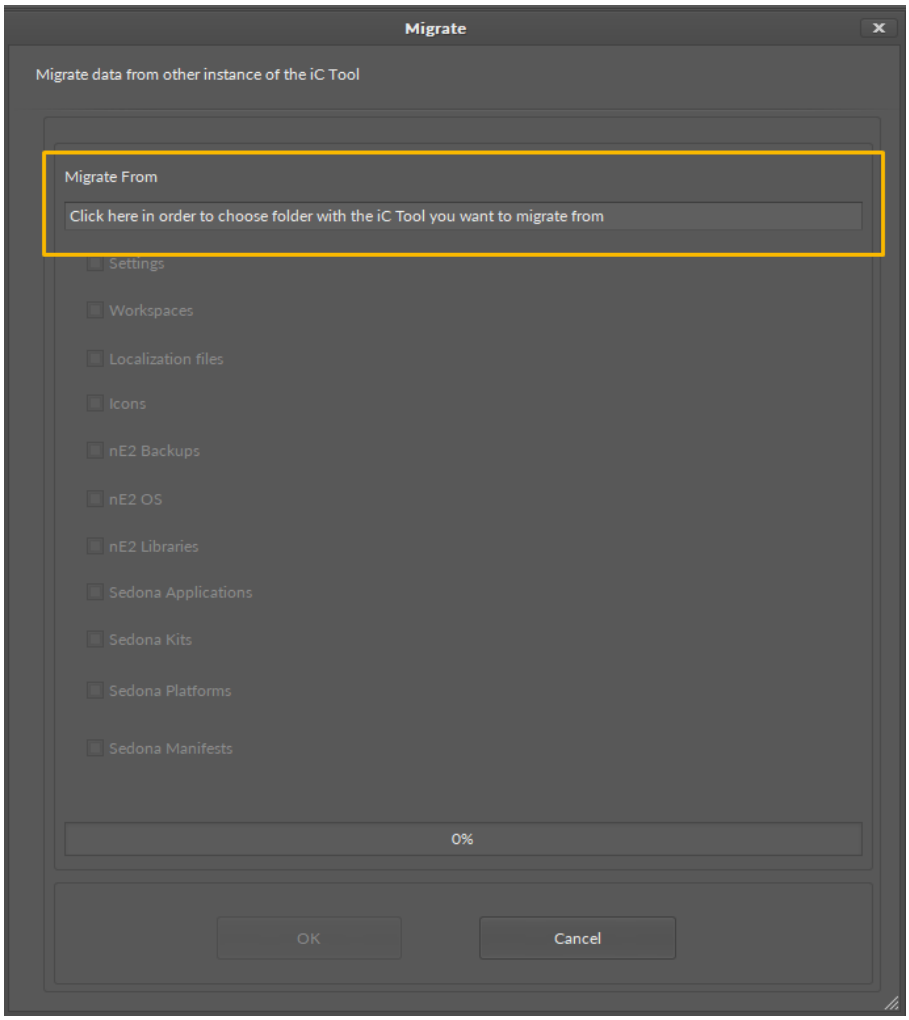



*Figure 5. The Migrate option*

In order to start a migration process, first, indicate a source directory folder.




Then, select files to import on a check list, and confirm with OK. The iC Tool will then require a confirmation to overwrite files in the destination instance. If not confirmed, the migration process will be ceased. To continue the migration process, confirm with Yes.

**Warning!**

It is important to bear in mind that the process **will overwrite** all files existing in the current iC Tool instance. Make sure to create a backup in case there are any files, applications, etc., that are required to be preserved.

# 3 Settings

## 3.1 Software Settings

The iC Tool allows to adjust individual settings for a better fit of the working environment to the user's needs. Such settings include, among others, the language interface, number of remembered or recently opened files of the Workspace Tree, and Console settings. The settings are available in the main menu File-> Settings.

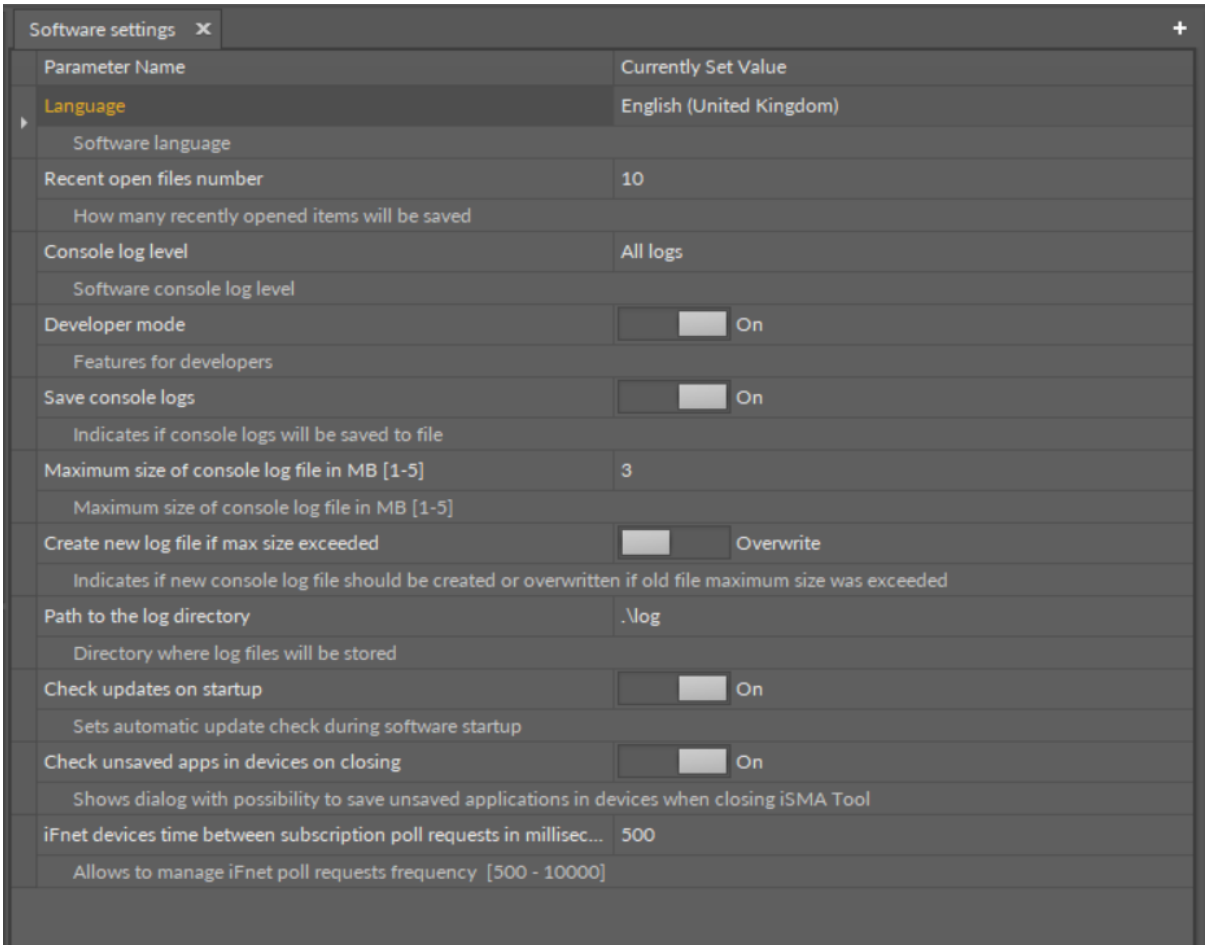



*Figure 6. Software settings*

The meaning of particular settings is explained under each setting line, see the figure above. Parameters to set in the Software settings include:

- language;
- number of recently opened files;
- Console log level;
- turning on/off developer mode;
- saving logs to a file;
- size of Console log file;
- action upon exceeding maximum log file size;
- path to the log directory;
- checking updates on startup;
- checking unsaved apps on closing;
- iFnet communication timeout;
- time of iFnet request waiting;
- time between subscription poll requests;

- frequency of iFnet requests.

### 3.1.1 Language Settings

A default language of the iC Tool is UK English.

The iC Tool allows to change the language in the Software settings view. The language change affects a graphic interface of the iC Tool. After the language is changed, the program needs to be restarted.

If the language is not on the supported languages list, please contact iSMA CONTROLLI Support Team to discuss the options (contact information is available in the About view, opened from the main menu Help -> About).

# 4 Naming and Definitions

A proper understanding of the terms used hereafter will allow the iC Tool user to adopt the described issues in a more proficient way.

## 4.1 Common Definitions

- iC Tool: the name of a computer program this documentation applies to;
- component: a single element of an application. The component is either a function block with a pre-defined task (algorithm), which requires specific parameters and logical connections with other components to work and exchange data properly, or a container (folder to segregate other components);
- link: a logical connection between components: such connection allows two components to send data between each other. The graphical representation of a link in the Wire Sheet view is a line connecting components;
- slot: an element of the component which defines the parameter along with its name, value, unit, and range. Slots are not visible in all views in order to avoid overwhelming the user with too much information. The views dedicated to work with the slots are: the Property Sheet and Wire Sheet, and the Slot Sheet for Sedona based devices;
- selection: a user's action of choosing any component or link by clicking it. The chosen object becomes highlighted graphically. In the Wire Sheet view, the component's selection is illustrated by surrounding a component with an orange frame, and a link's selection is illustrated by changing its color to orange. The selected element is an object to which the commands such as copying, deleting, or moving will relate;
- marking: a state relating to the component or link connected with the selected object. Marking means that a specific component or link is in correlation with a selected component/link. In the Wire Sheet view, a green color is used to mark a component or link, which allows the user to easily notice which components/links will be affected by the action taken on the selected object, e.g., after deleting a selected component/link the marked object will lose its correlation;
- Workspace Tree window: one of the basic windows provided by the iC Tool. The window shows the connected devices along with their components. Additionally, the devices may be assigned hierarchically to folders, and folders to projects;
- Main Screen window: a main workspace of the iC Tool, where the views of editable components are displayed on tabs;
- Object Properties window: a window showing information regarding a selected element allowing to preview and change some of the selected element's parameters without closing the current view;
- Console: a window where the messages regarding the operation of the iC Tool program are displayed.

### 4.1.1 Component Views

A component shows its properties on views assigned to it. The list below shows two basic views, which present data about the same component in different ways, making the work with components easier. Some components have also individual views (apart from the standard ones), which are not included in the list below (see the Special Views or the Sedona Specific Views).

- Wire Sheet: a view showing programming relations between the components in a graphical way. The view shows logical relations between the linked components. This

view makes building an application logic easier, because the connections between the components are easy to enter and analyze. It is the view most commonly used on the application programming stage;
- Property Sheet: a tabular form of showing parameters–component's slots. The user can see not only the slot value, but also its units and range. The view allows to change the slot value if it has the change value rights. The Property Sheet is often used at the start-up stage, when the settings of many parameters (slots) in components need to be appointed.

## 4.2 nano EDGE ENGINE Devices

- library: a set of components with similar functionalities or working in the same area, e.g., components handling device's inputs and outputs. In the iC Tool, the libraries and their components are shown in a Device Libraries window;
- Reference link: a special compound link designed to connect Data Points with network points. The Reference link is created between special Reference slots and transfers values along with the component's status; Alternatively, it may transfer values between Data Points and network points at the same time returning status from network points to Data Points, or it may return values from network points to Data Points;
- Device Libraries window: one of the basic windows provided by the iC Tool. It is a palette of nano EDGE ENGINE libraries along with components assigned to them. The content of the Device Libraries is dynamically adjusted to the content installed in the device's SD card.

## 4.3 Sedona Devices

- kit: a set of components with similar functionalities or working in the same area, e.g., components handling controller's inputs and outputs. In the iC Tool, the kits and their components are shown in a Device Kits window;
- Device Kits: one of the basic windows provided by the iC Tool. It is a palette of Sedona kits along with components assigned to them. The content of the Device Kits is dynamically adjusted to the type of connected device;
- Slot Sheet: a component view showing all slots, including the hidden ones. In a tabular form, it allows to read the name, type, and parameters (Facets) of the slot.

# 5 Navigation and Windows

## 5.1 iC Tool Windows

A structure of the iC Tool is based on widows with varying functionalities, which altogether allows for user-friendly programming of devices. The basic structure of windows is shown on the figure below.

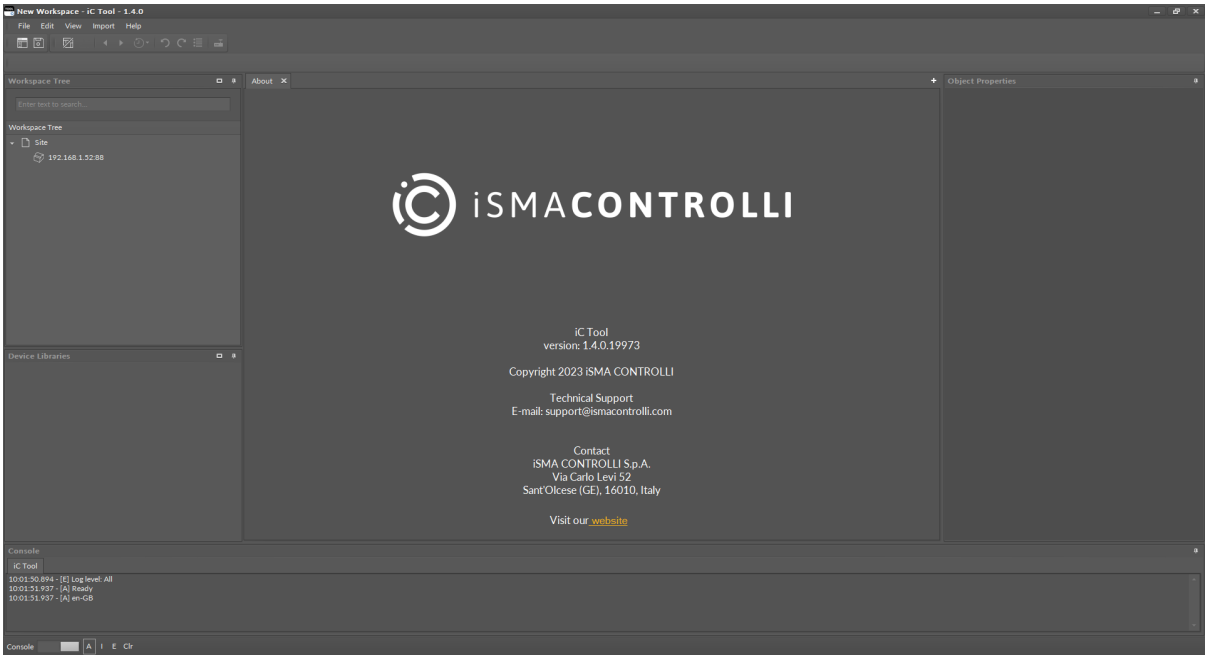



*Figure 7. The iC Tool initial window*

The above look presents an initial set of iC Tool widnows. Next opening and updates of the program will remember the window settings changed by the user in accordance with his preferences.

In the default settings of the iC Tool, the upper left side of the screen is occupied by a Workspace Tree window. Below is a Device Libraries/Kits window showing current libraries/kits of a connected device. Central part of the screen is occupied by a Main screen, a space where the application's views are shown on tabs, which is discussed further. After the iC Tool is opened, the Main screen shows program information shown on the figure above. On the right side of the Main screen there is an Object Properties window. A Console window is by default minimized. It is accessible as an expandable tab in the bottom left corner of the iC Tool .

Additionally, besides the mentioned windows, the iC Tool screen shows a menu and a toolbar, a window with component path, and a capacity identifier.

The window with an component path is by default set below the main menu and toolbar. It can be moved to the bottom, or to the left or right margin by changing the path text to the vertical one.

The suggested window position in the iC Tool can be modified freely by dragging, resizing, and fixing views relative to each other.

### 5.1.1 Start Screen for nano EDGE ENGINE Devices

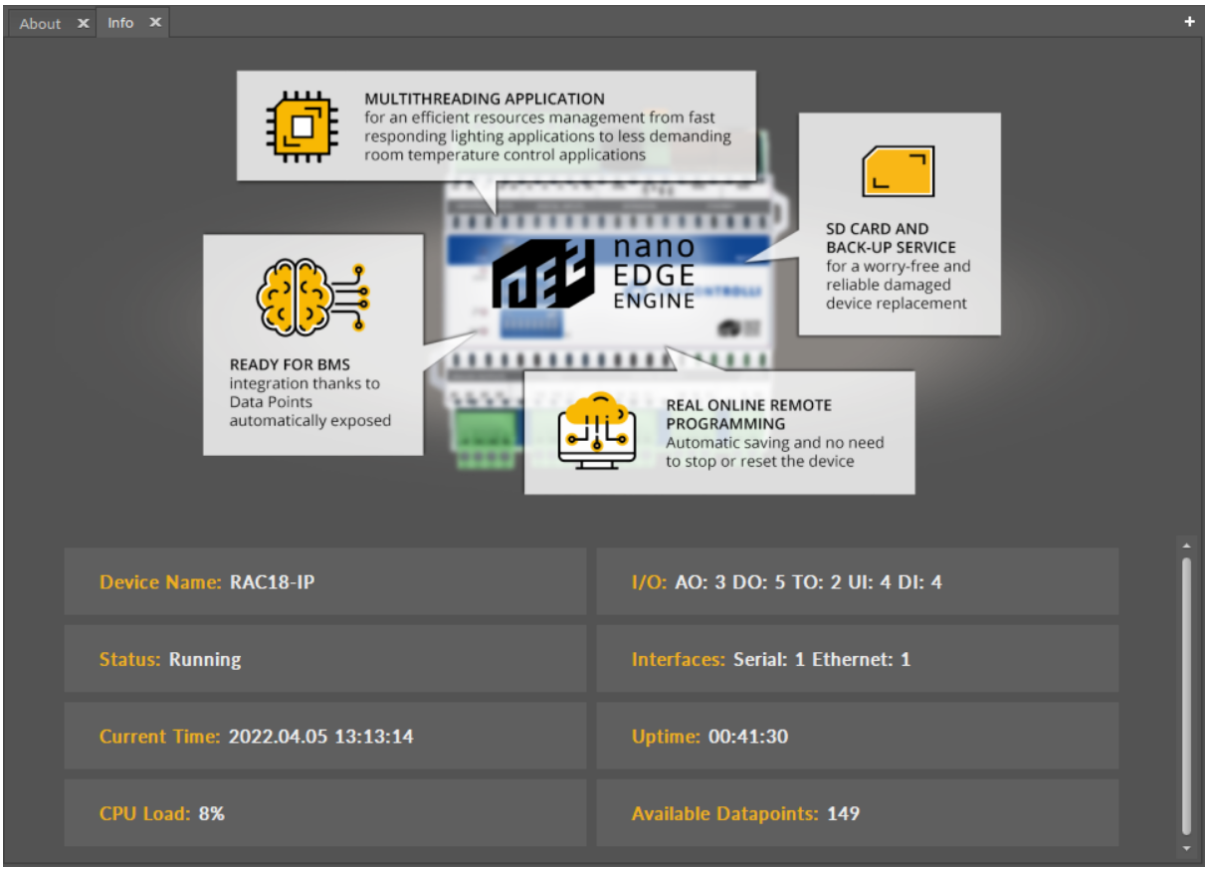



*Figure 8. The iC Tool start screen*

Once the iC Tool connects with a nano EDGE ENGINE device, a start screen is displayed. It provides basic information about:

- device name;
- I/O;
- status;
- interfaces;
- current time;
- uptime;
- CPU load;
- available Data Points.

No specific start screen is displayed when connecting to a Sedona based device.

## 5.2 Windows Adjustment

The iC Tool allows a completely free setting of the views and windows. Free distribution allows specific windows to be visible non-stop or hide automatically. These settings apply to the windows such as the Workspace Tree, Device Libraries/Kits, or Object Properties. It is possible to move the window outside of the iC Tool software to the other connected screen–the only exception here is the Main screen window, which cannot be moved to the other connected screen.

To move any window into another part of the screen, it needs to be grabbed by the header and dragged into a new location. The figure below shows moving of the Device

Libraries window. While moving the window, special controls appear to facilitate proper placing of the window in relation to the remaining windows in the view.

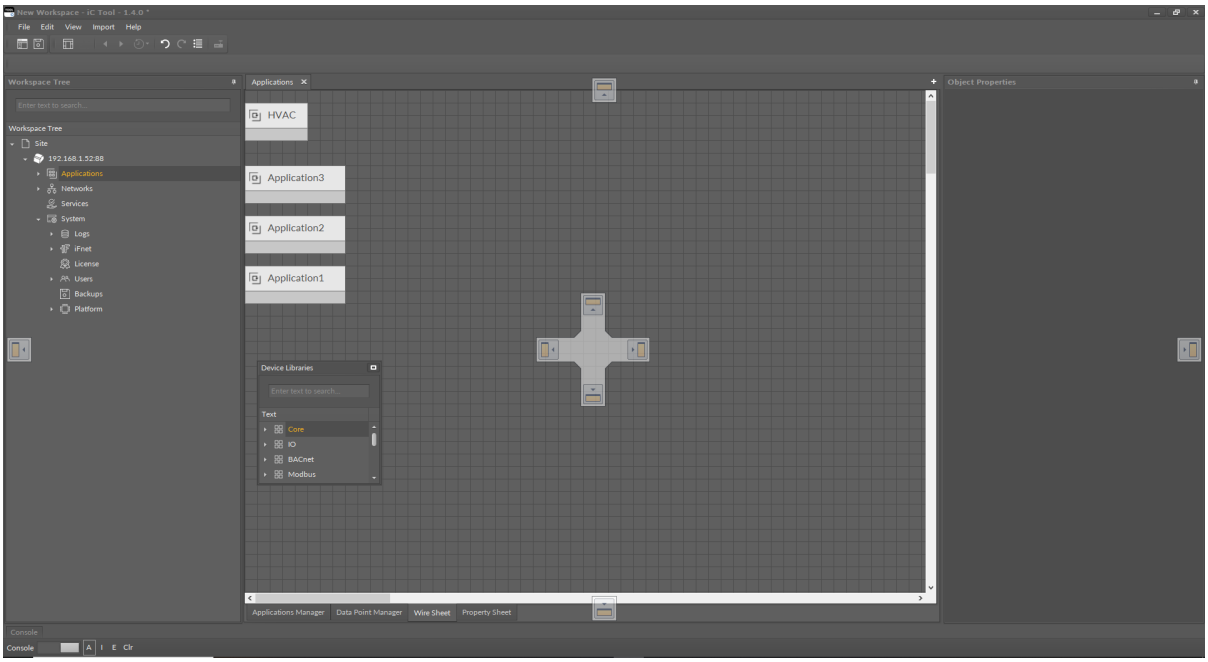



*Figure 9. Windows adjustment controls*

As a result of changing the windows position, it is possible to adjust them according to individual needs. On the figure below, the Workspace Tree window has been moved above the Main screen, and the Device Libraries window has changed its position from the left side to the right.

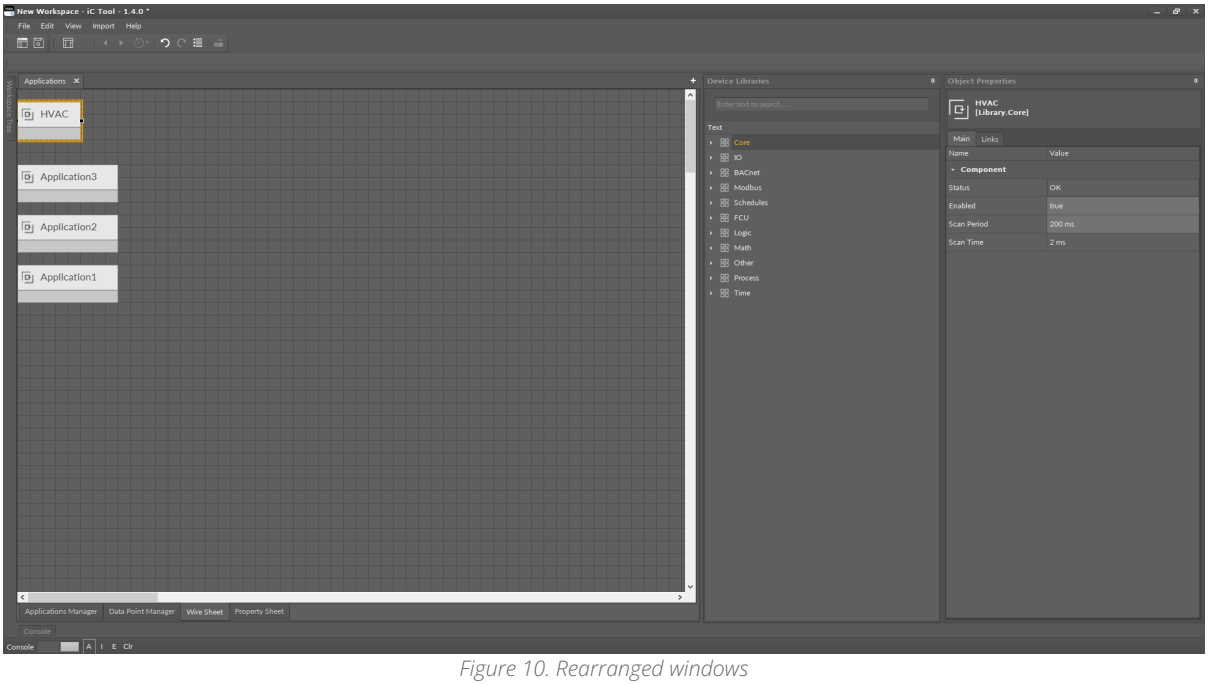



*Figure 10. Rearranged windows*

The iC Tool offers a quick extension of the Main screen to the whole area thanks to automatic minimizing of the remaining windows. To temporarily enlarge the Main screen, press the Hide panels icon (third one from the right) on the toolbar under the main menu, as it is depicted on the figures below showing the iC Tool before and after the enlargement of Main screen. In order to restore the full view of all windows, either press the icon again or press the next Restore previous panels set icon.

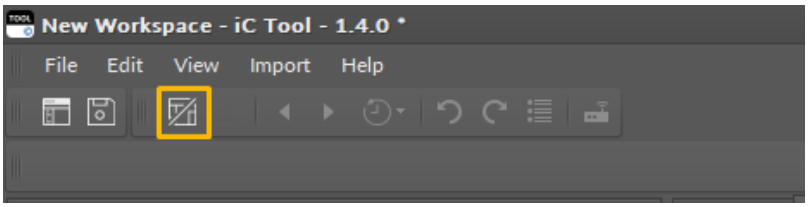



*Figure 11. Hide panels button*

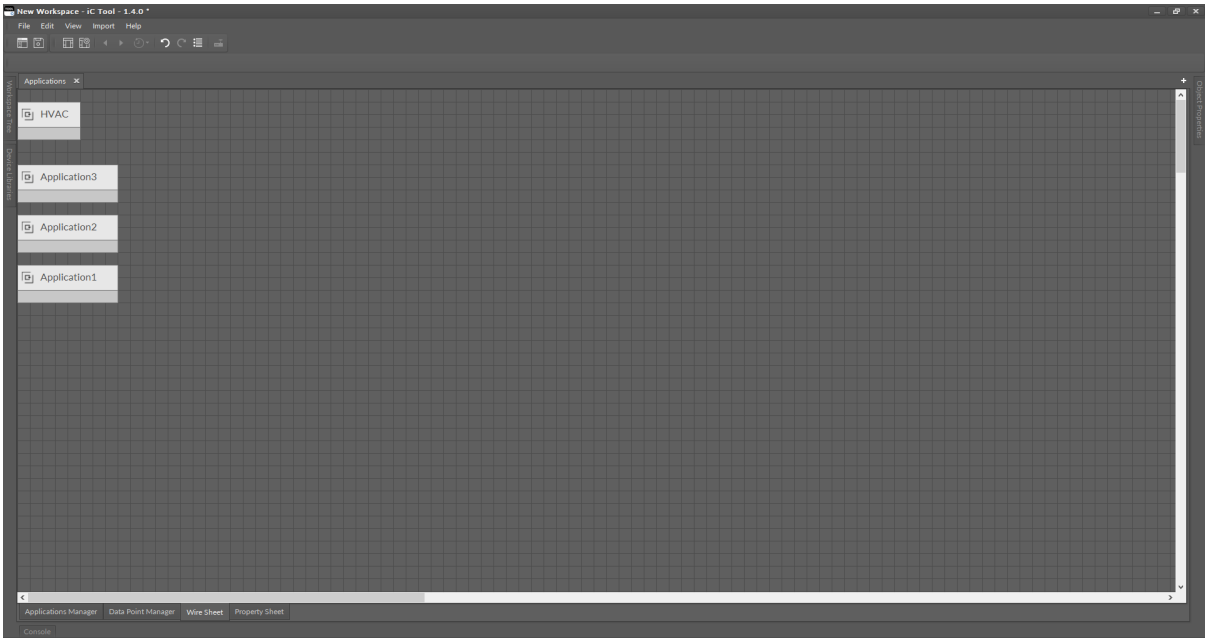



*Figure 12. Temporarily enlarged main screen*

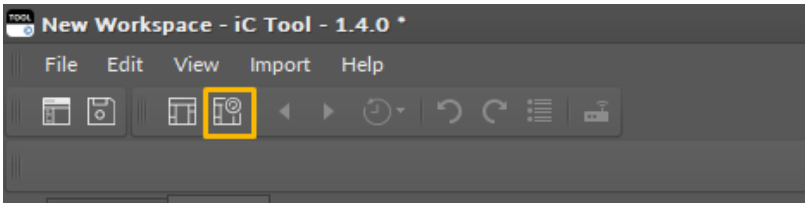



*Figure 13. Restore previous panels set button*

## 5.3 Workspace Tree

A Workspace Tree, also known as a project tree, is a window designed to work with devices. Thanks to a hierarchical structure of the tree, it is easy to switch between the devices and their components. It is also a place to add/remove the devices and to manage components.

The Workspace Tree is designed to navigate the projects and devices defined for them. For each project, it is possible to define any number of folders and any number of devices inside them. It is recommended that the structure of folders, subfolders, and devices corresponds to their physical placement and/or logical subordination of devices on an object. It makes finding a particular device among a vast quantity of devices fast and easy. The structure of folders, subfolders, and projects, mentioned above, allows to properly locate a specific device in the Workspace Tree. The hierarchical structure inside the device is loaded directly from the device and consists of components.

The Workspace Tree is saved in an .itw file in a specified location [home\Workspaces]. The user can save the Workspace Tree under a defined name in a specific location. The .itw (Workspace) file stores the whole structure of the Workspace Tree including devices

names and their addresses. This allows to save many projects in one .itw file, or otherwise, each individual project in a separate .itw file.

A file with the .itw extension is an .xml file, and contains sensitive data, such as device IP addresses and their passwords if the user issued a command to remember them. Due to an obvious threat of exposing such data, the iC Tool codes sensitive data automatically, preventing their exposure to third parties.

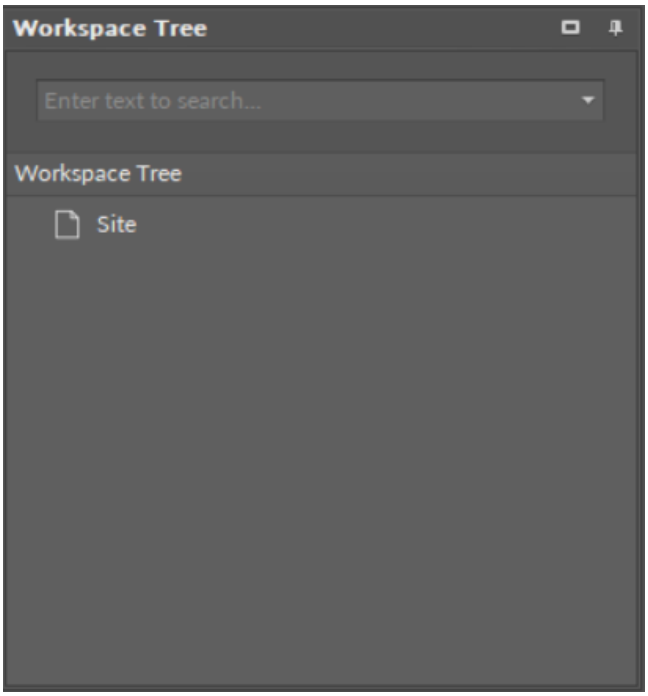



*Figure 14. Initial project in the Workspace Tree*

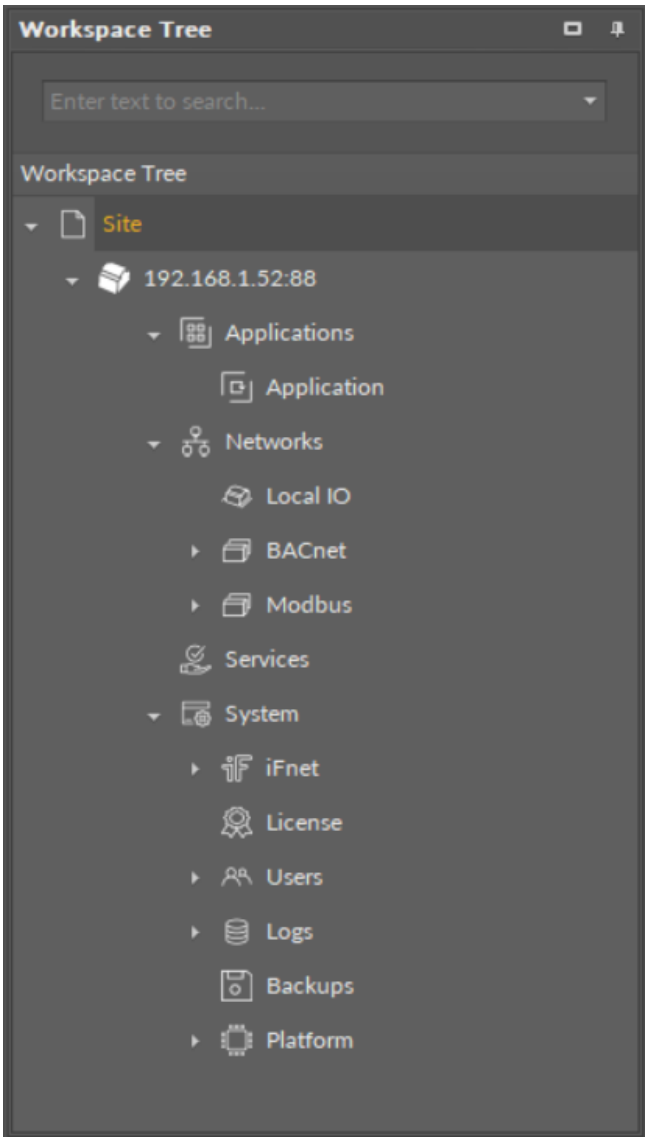



*Figure 15. Workspace Tree structure in the nanoEdgeEngine device*

### 5.3.1 Editing of the Workspace Tree Elements

For folders, subfolders, and the main element–project, opening the context menu provides the following options:

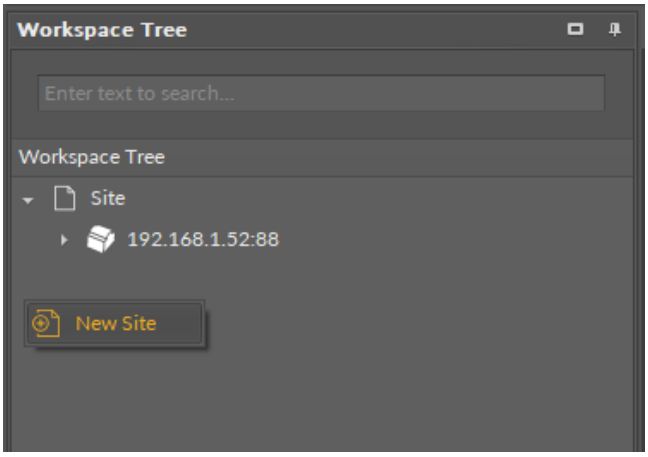



*Figure 16. Adding new site in the Workspace Tree*

- New Site: creating a new project independent of the others (right-click on empty space in the Workspace Tree);

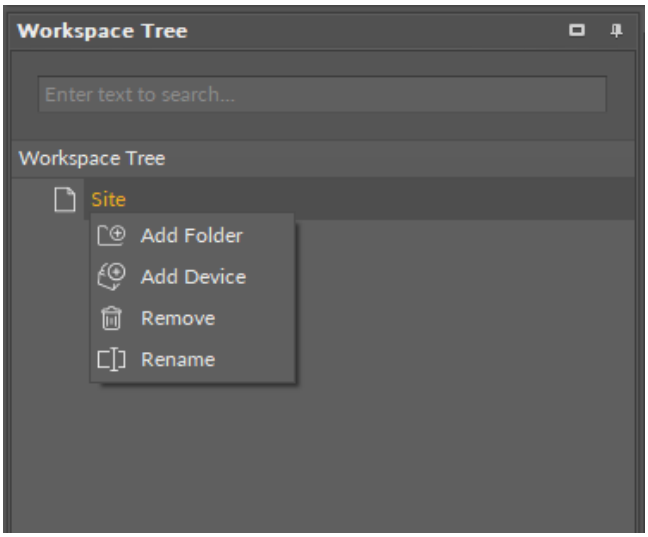



*Figure 17. Workspace Tree editing options*

- Add Folder: adding a folder under the selected folder;
- Add Device: opening a device adding window for the selected folder;
- Remove: removing the selected element: device, folder, or project from the tree;
- Rename: renaming the site.

**WARNING!**

Removing a superior folder or project removes all subfolders and items that were structured under it, upon confirmation of operation. In the situation shown on the figure, removing the project will result in removing the device from the project, but the device contents will not be deleted.

The iC Tool software supports UTF8 symbols in most of the places in the project. It is therefore possible to use native letters to lay out the project in a convenient way:

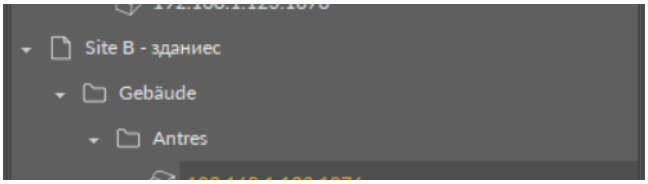



*Figure 18. Use of native letters example*

Any name can be given to a project, folder, or even device. For this purpose select and click an object, which then becomes editable, or select an object and press the F2 key.

**Tip**

Use an F2 key to rename an object.

The device components, which can be used to program the application, are shown on the Workspace Tree, under a device. If a selected element of the tree has other elements under it, i.e., inferior elements, issuing a command to a the superior component impacts all the subject elements as well, due to a hierarchical structure of the tree. This involves the Delete function–deleting the superior element deletes all the inferior elements.

It is possible to select many elements at once by using a Ctrl or a Shift key the same way as in the Windows Explorer (selecting elements with a Ctrl key pressed allows to select elements one by one, not in a consecutive manner; selecting elements with a Shift key pressed allows to select consecutive elements).

**WARNING!**

To save any modifications in the Workspace Tree hierarchy or to save its naming (not applicable to device tree), the Workspace Tree needs to be saved using a Save or a Save as option available in a File menu or corresponding button on the toolbar.

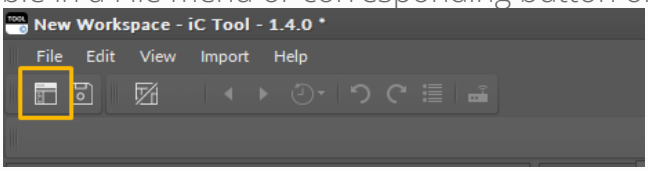



*Figure 19. Save workspace button*

## 5.3.2 Adding and Connecting Device

### Adding Device

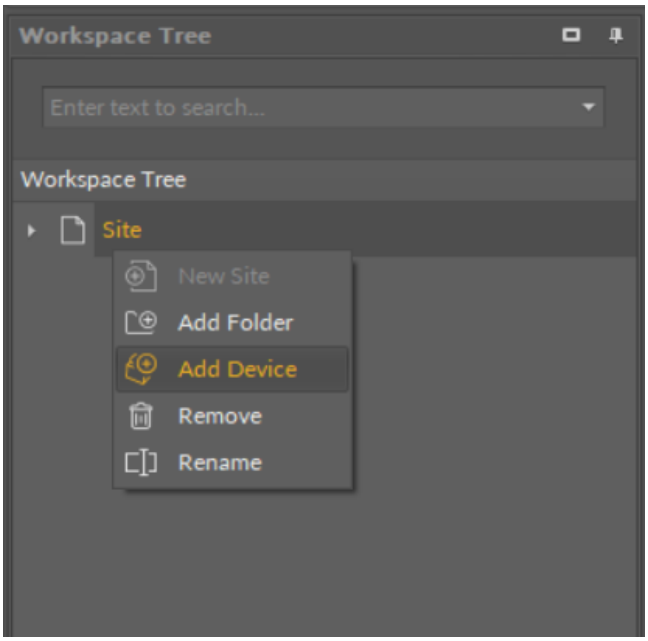



*Figure 20. Adding the device*

To connect the device, it needs to be added to the Workspace Tree: right-click anywhere in the Workspace Tree window, and add a new project from the context menu (in case the required project is already added, omit this step). Then, right-click on the project, and select the Add Device option from the context menu, which opens a dialog window allowing to configure basic connection data:

- Type: displays the type of the device to be connected;
- Host: allows to set a device's IP address;
- Port: shows a device's port number (port is automatically selected according to the type of the device: 88 for nano EDGE ENGINE devices, 1876 for Sedona devices).

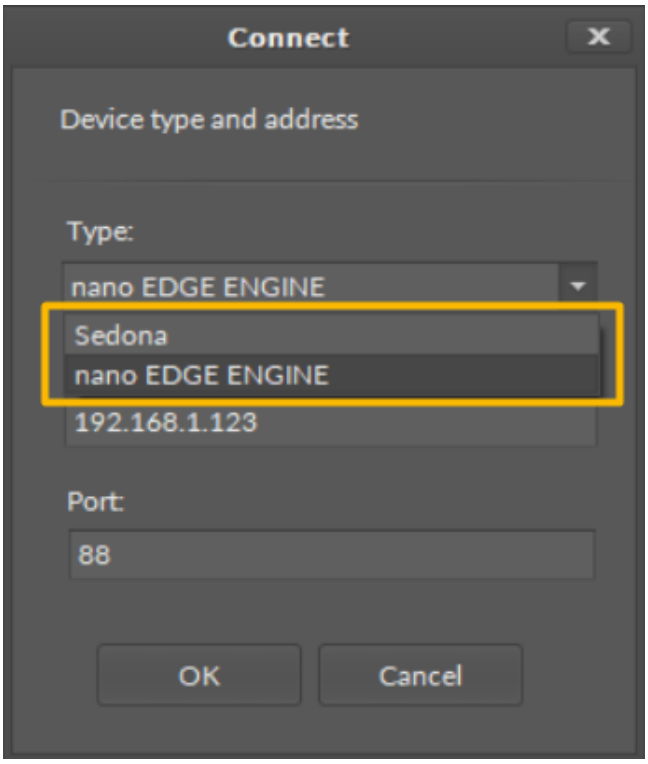



*Figure 21. Connection dialog window with drop-down list for selecting a type of the device*

**Worth to Notice:**

Each device has a default IP address 192.168.1.123, which may be later changed in the Platform component for nano EDGE ENGINE devices or the plat component for Sedona devices.

**Worth to Notice:**

Upon adding, each device shows its default port number:

- 88 for nano EDGE ENGINE devices;
- 1876 for Sedona devices.

## Connecting Device

After entering a proper IP address and pressing Enter or OK, a login window appears to enter the username and password.

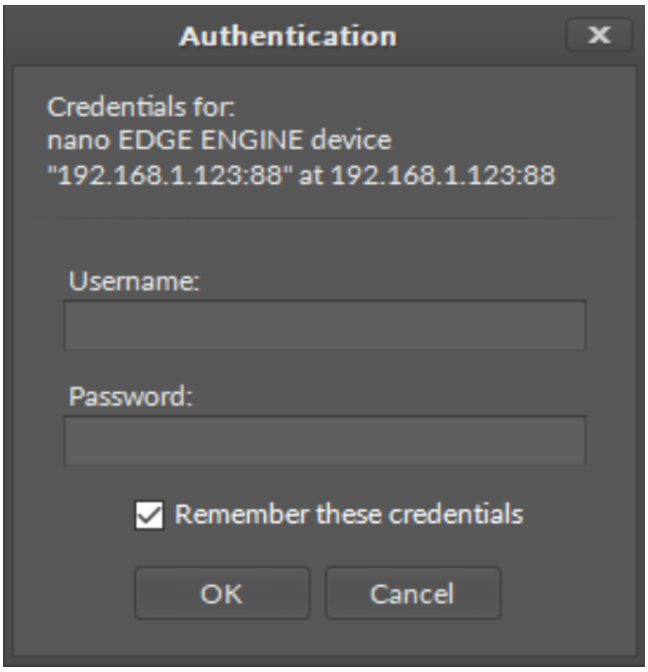



*Figure 22. Authentication dialog window*

**WARNING!**

Saving credentials is dangerous from a security standpoint. Please avoid storing production system passwords.

If the username and password, needed to connect with the device, are to be remembered, select the Remember these credentials option. The entered data are then saved in the .itw file, which stores the whole structure of the Workspace Tree in a way that prevents access to sensitive data by third parties.

After confirming the entered authorization data by pressing Enter or OK button, the iC Tool begins the process of connecting and reading the components tree for the device. After a correct connection with the device, its application tree loads automatically in the Workspace Tree. If the device is selected in the Workspace Tree view, the Object Properties Window shows its picture and the most important parameters, such as IP address, serial number, or operating system version number. These parameters are available in this window even after disconnecting the device, which allows offline checking of, e.g., device IP number or its operating system version.

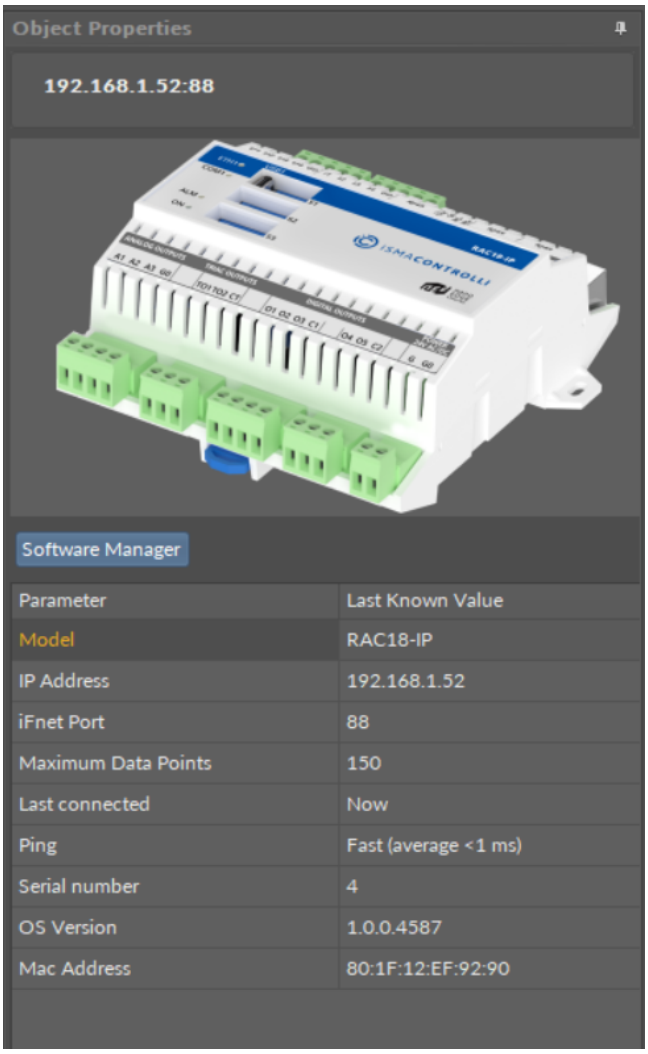



*Figure 23. The Object Properties window for a connected nano EDGE ENGINE device*

## Connecting an Added Device

If the iC Tool is opened, and all the devices are already added to the project, in order to connect the device, right-click it, and select the Connect option from the context menu.

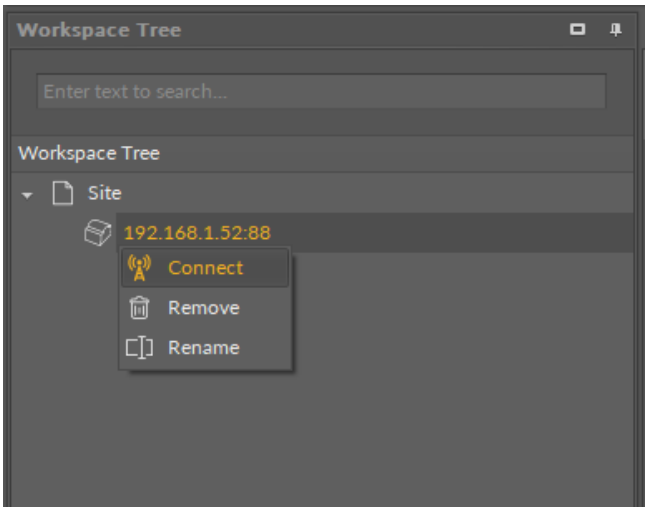



*Figure 24. Connecting an added device*

Then the Authorization window pops up, either with saved credentials, or ready to fill them in.

### 5.3.3 Disconnecting Device

To disconnect the device, select it on the Workspace Tree and choose a Disconnect option from the context menu, then accept.


*Figure 25. Disconnect option in the context menu of a nano EDGE ENGINE device*

After disconnecting the device its icon's view changes; it is impossible to expand its contents and the figure is grayed out (see the figure below). Double-clicking on the device again begins the reconnection process.

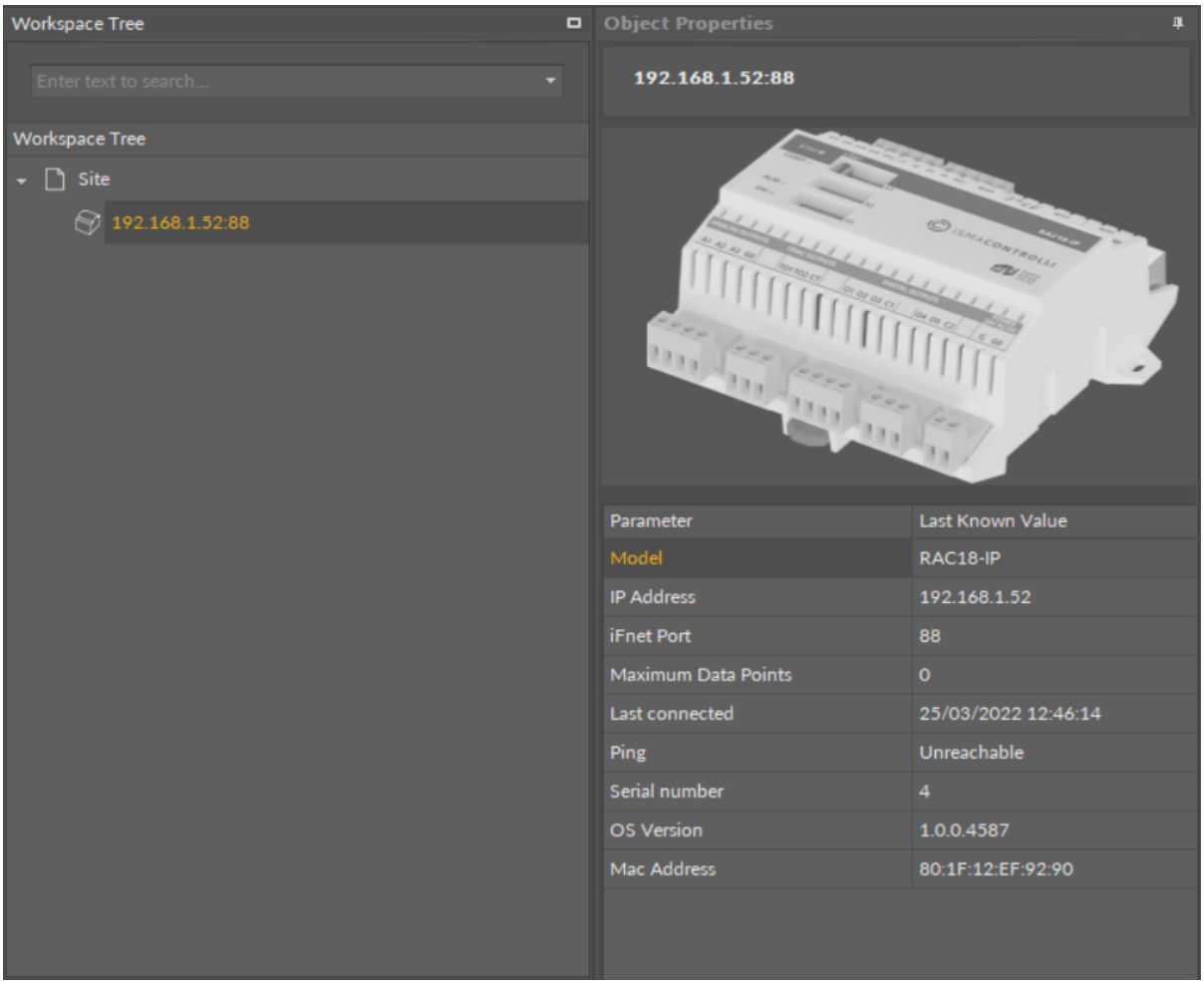



*Figure 26. Disconnected device view*

If the device gets disconnected by mistake, while working with it, the program will detect it automatically and inform the user showing a “Device disconnected” text in the upper part of the view and folding the device tree on Workspace Tree.

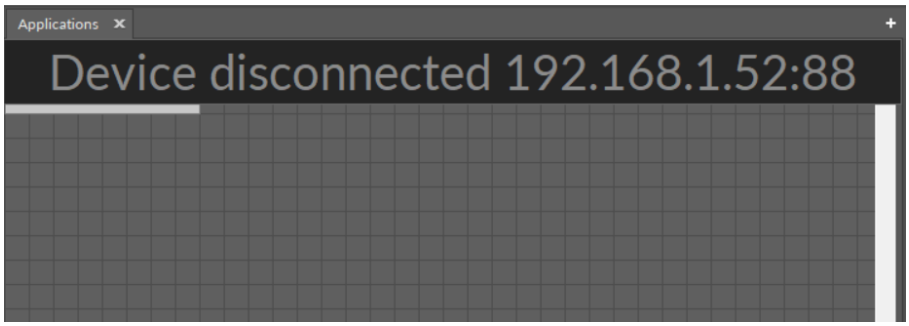



*Figure 27. Disconnected device notification*

## 5.3.4 Connected Device View

After connecting with the device, a subtree appears in the Workspace Tree listing all components available in the device's structure (depending on whether a connected device is based on the nano EDGE ENGINE or Sedona).

Every change made in components, such as editing a component's name, its superior component, or other parameters in the component, is saved in the device. This guarantees that, after reloading the data from the device, data (components) are

displayed in the same way and they are synchronized among other users working at the same time on the same device.

Operations, which can be performed on components such as copying, adding, or removing new components, defining connections between components, are described in the further part of this manual.

Immediately after connecting with the device, both the Main screen and Device Libraries/Kits window remain unchanged. It is the opening of any component from the device tree that results in showing a specific view in the Main screen and filling the Device Libraries/Kits window with the proper device libraries/kits.

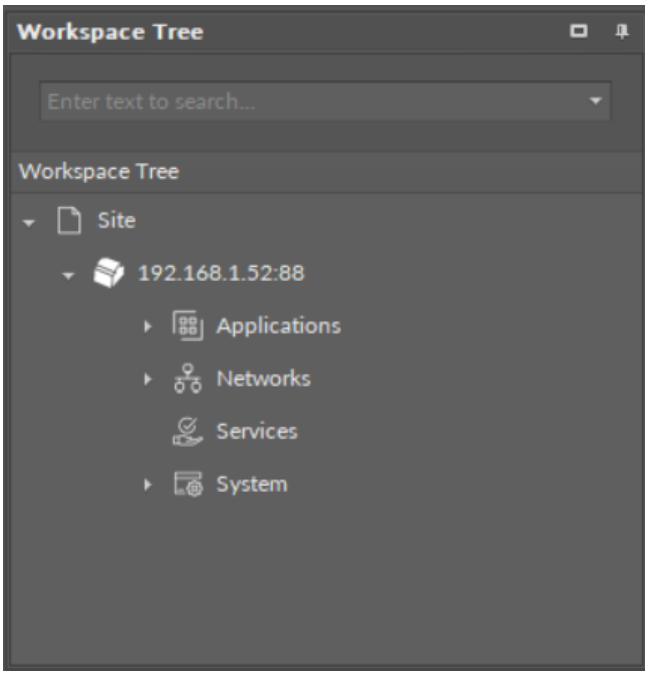



*Figure 28. Connected device view in the Workspace Tree window of the nano EDGE ENGINE controller*

## 5.3.5 Adding and Removing Components in the Workspace Tree

### Adding Components

In order to add a new component in the Workspace Tree window, it is necessary to find a specific component in the Device Libraries/Kits window, and drag it to a proper place in the Workspace Tree. Adding components is performed by a drag and drop method. For nano EDGE ENGINE devices, it is possible to place inferior components only in the proper container or service component–the iC tool automatically blocks it, for example, to place an Application component in the Networks container, or the Device component in the iFnet service component.

For Sedona based devices there is no such limitation.

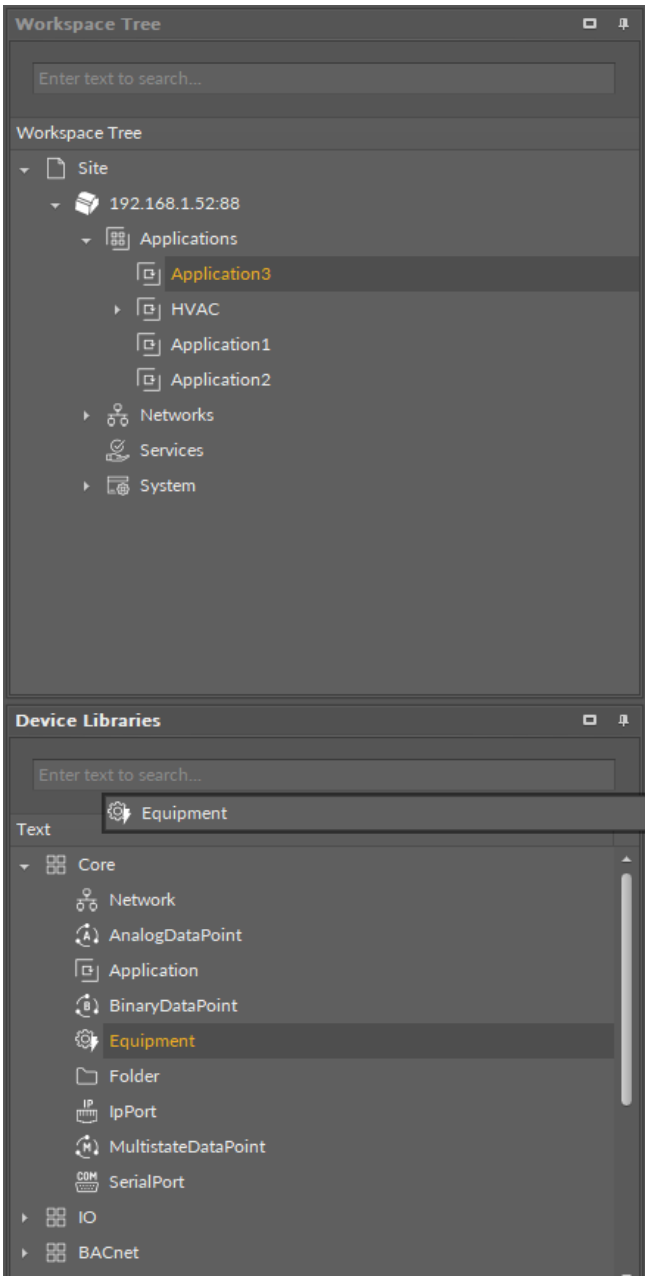



*Figure 29. Dragging new components*

After dropping a single component in the specific place in the Workspace Tree, a dialog window appears, allowing to name a newly created component. After confirmation of the name, the component is added to the device, and the tree is refreshed.

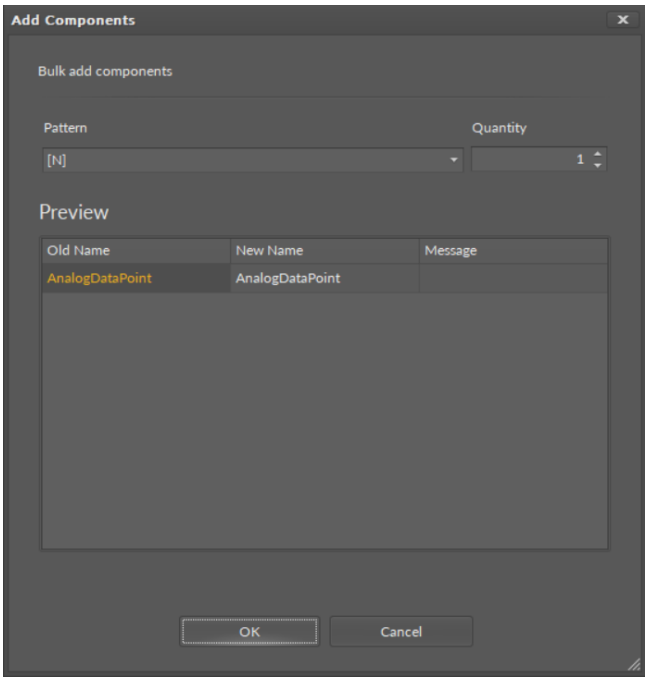



*Figure 30. Naming component dialog window*

Many components can be dragged at once. After dropping them on the Workspace Tree, their names will be automatically defined as the component's default name, without the user's participation.

A second method of adding a component is copying and pasting or duplicating components already existing in the Workspace Tree. In order to do it, select a component in the Workspace Tree window, Property Sheet, or Wire Sheet view, and copy it using a keyboard shortcut Ctrl+C or the context menu using a Copy option. Afterwards, it is essential to choose the component in the Workspace Tree, under which the copied component is pasted with a Ctrl+V operation or the context using a Paste option.

The details of copying/pasting operation as well as duplication are described in the Single Device Functions section.

### Special Adding of Data Points, Schedules, and Folders in nano EDGE ENGINE Devices

The Workspace Tree window, the Wire Sheet and Property Sheet views offer a special way of adding Data Points, schedules, and folders (the Folder and Equipment components) to the Application, simply from the context menu. In order to add components this way, right-click in an empty space of the Wire Sheet grid or Property Sheet and choose the Add option from the context menu. In the Workspace Tree, special adding of Data Points, schedules, and folders is available in the context menu of the Application component. This way of adding Data Points, schedules, and folders to the Application allows to add a multiple Data Points or folders.

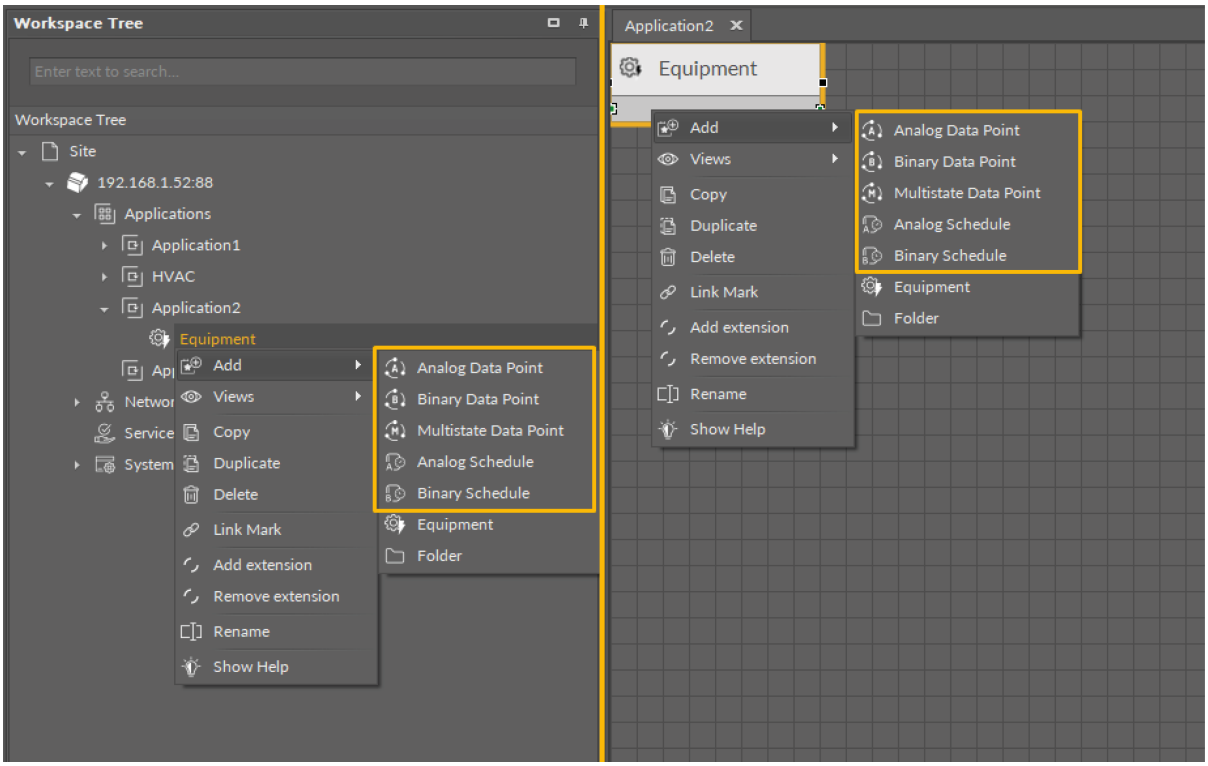



*Figure 31. Special adding of Data Points and folders from the context menu*

## Removing Components

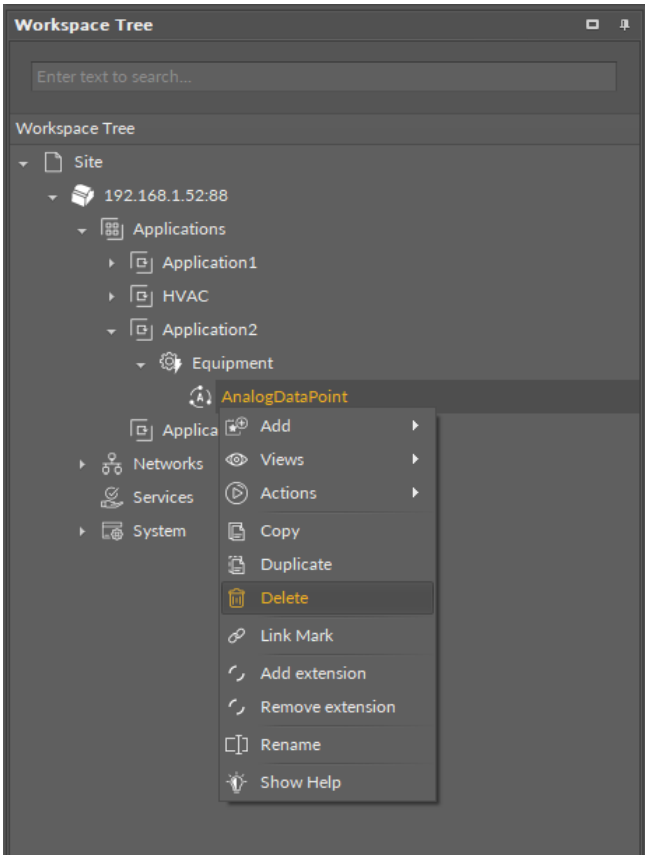



*Figure 32. Removing component*

A component may be removed from the Workspace Tree window level only by selecting it and choosing a Delete option from the context menu. Removal by pressing Delete key on the keyboard is blocked for this window.

## 5.4 Device Libraries and Device Kits

### 5.4.1 Device Libraries

The Device Libraries is a window listing all libraries available after connecting the iC Tool to a device driven by the nano EDGE ENGINE. This window always shows the libraries available in the device connected at the moment. It means that, if there are many devices used in a project, the window contents will be automatically adjusted at the moment of switching from one device to another.

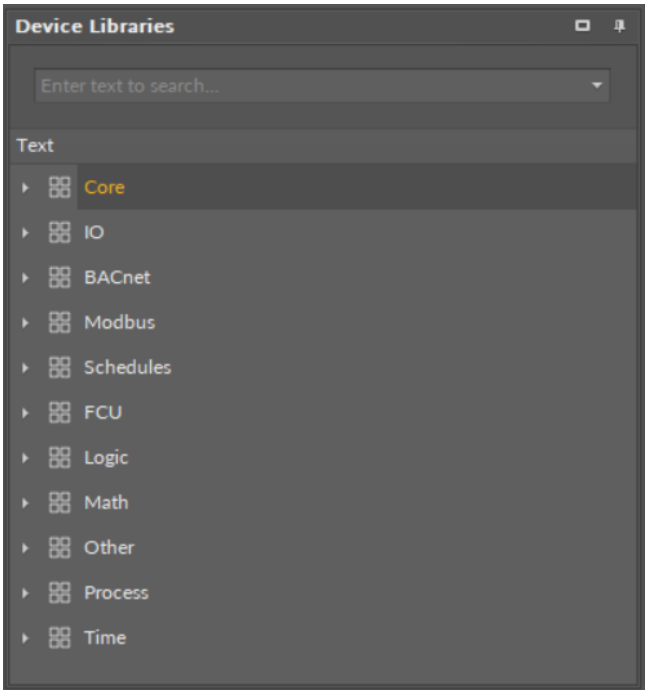



*Figure 33. The Device Libraries window*

The libraries are presented as the main elements of a tree structure in the Device Libraries window, and can be expanded to see their components. Each component is presented with its icon and name. The icons visible before the component name define it clearly and help to quickly identify the type of component. An icon associated with a particular component is also shown in other views, including the Wire Sheet, Property Sheet, and the path of an editable component.

It is possible to select and move more than one component simultaneously in the Device Libraries by using a Ctrl or Shift key.

Using any library from the Device Libraries window, means adding a not configured component with default parameters. In order to add a component similar to already existing one with pre-configured parameters to the project, it is necessary to use copying or duplicating function on the existing component.

The components dragged directly from the Device Libraries to the Wire Sheet view (this view is explained in detail in the Wire Sheet section) are placed in the location they were dropped. If the components were dropped on a view other than the Wire Sheet view, their physical location (shown on the Wire Sheet view) will be in the upper left corner–point (0,0).

While dragging many components to any view, the iC Tool will arrange them in a cascading manner, so they will not overlap.

The components in the Device Libraries window cannot be edited and their location cannot be changed.

### Searching the Device Libraries

In order to easily find a specific component among a vast number of components available for a particular device, it is possible to use a filter available at the top of the window. For this purpose, it is necessary to enter the name or part of the name of the required component into the filter field, and the matching components list will be shown dynamically in the Device Libraries window. The example below shows a list of components matched with a word 'analog'.

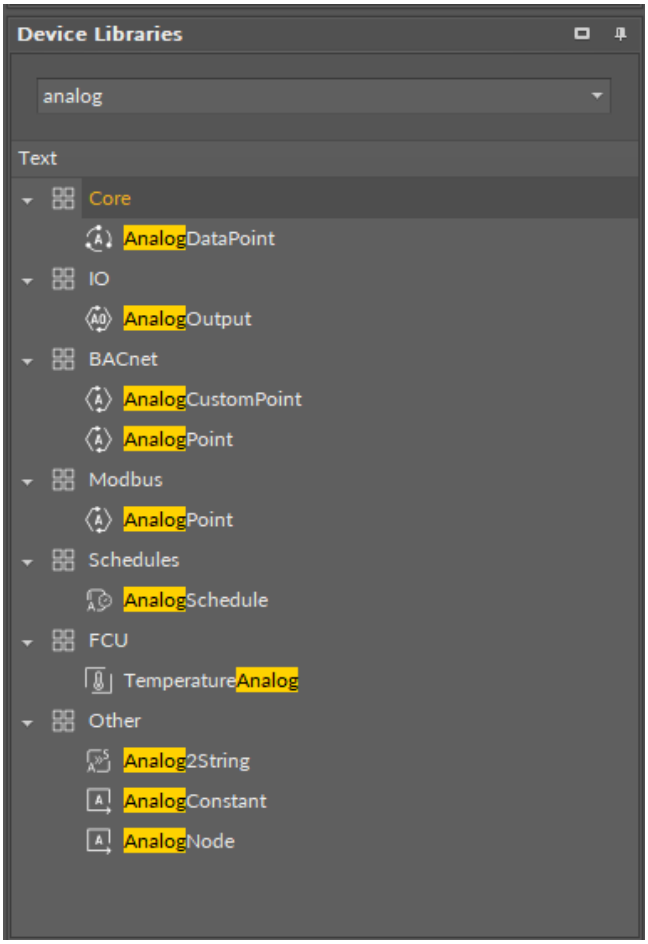



*Figure 34. Search results in the Device Libraries window*

## 5.4.2 Device Kits

The Device Kits is a window with a list of kits available after connecting the iC Tool to a device powered by Sedona. This window always shows the kits available in the device connected at the moment. It means that, if there are many devices used in a project, the window contents will be automatically adjusted at the moment of switching from one device to another.

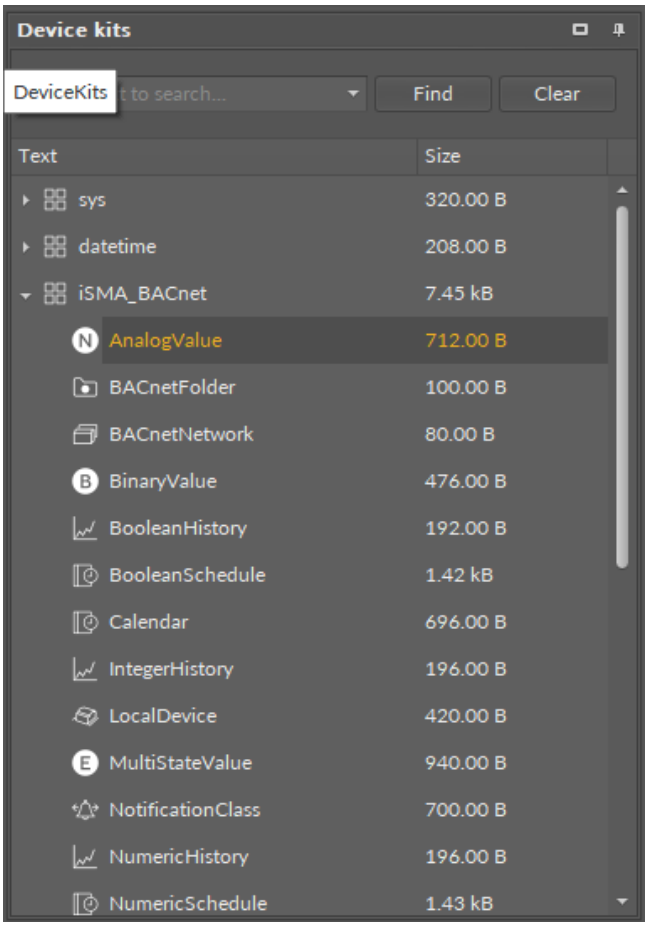



*Figure 35. The Device Kits window*

The kits are presented as the main elements of a tree structure in the Device Kits window, and can be expanded to see their components. Each component is presented with its icon, name, and memory it occupies. The icons visible before the component name define it clearly and help to quickly identify the type of component. An icon associated with a particular component is also shown in other views, including the Wire Sheet, Property Sheet, and the path of an editable component. Besides the icons associated to components permanently, there are two special ones, as shown below.

*Figure 36. Special icons*

The first icon is a default icon for every component without a defined individual icon. The second icon is a warning icon shown for each component, for which the iC Tool cannot find a defined individual icon. Such situations may occur while adding custom kits. It requires checking if the path for component icon is correct and if it has been copied to the iC Tool icons folder.

## Searching the Device Kits

To use a component, it needs to be found and then moved to a required place by drag and drop method, e.g., to the Workspace Tree window.

In order to easily find a specific component among a vast number of components available for a particular controller, it is possible to use a filter available at the top of the window. For this purpose, it is necessary to enter the name or part of the name of the

required component into the filter field, and the matching components list will be shown in the Device Kits window. The example below shows a list of components matched with a word Boolean.

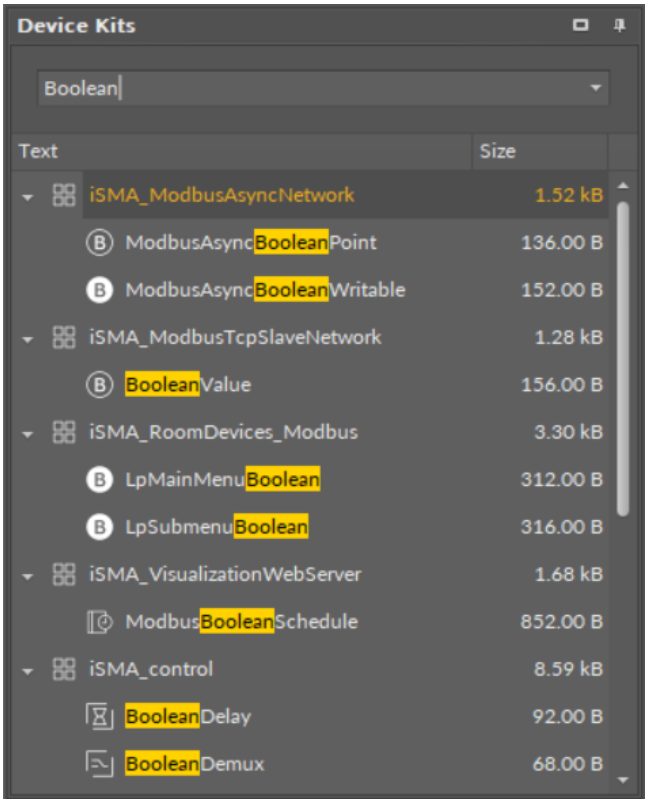



*Figure 37. Search results in the Device Kits*

It is possible to select and move more than one component simultaneously by using a Ctrl or Shift key.

Using any kit from the Device Kits window means adding a not configured component with default parameters. In order to add a component similar to already existing one with pre-configured parameters to the project, it is necessary to use copying or duplicating function on the existing component.

The components dragged to the Wire Sheet view (this view is explained in detail in Wire Sheet) are placed in the location they were dropped. If the components were dropped on a view other than the Wire Sheet view, their physical location (shown on the Wire Sheet view) will be in the upper left corner–point (0,0).

While dragging many components to any view, the iC Tool will arrange them in a cascading manner, so they will not overlap.

The components in the Device Kits window cannot be edited and their location cannot be changed.

## 5.5 Main Screen

The main workspace (Main Screen), where the application is created, is the middle, biggest window of the iC Tool. Here, the functional views are available in tabs: the Wire Sheet, Property Sheet, and Slot Sheet (only for Sedona devices).

Navigating between many views in the Main Screen is performed by selecting a proper tab placed at the bottom of the window.

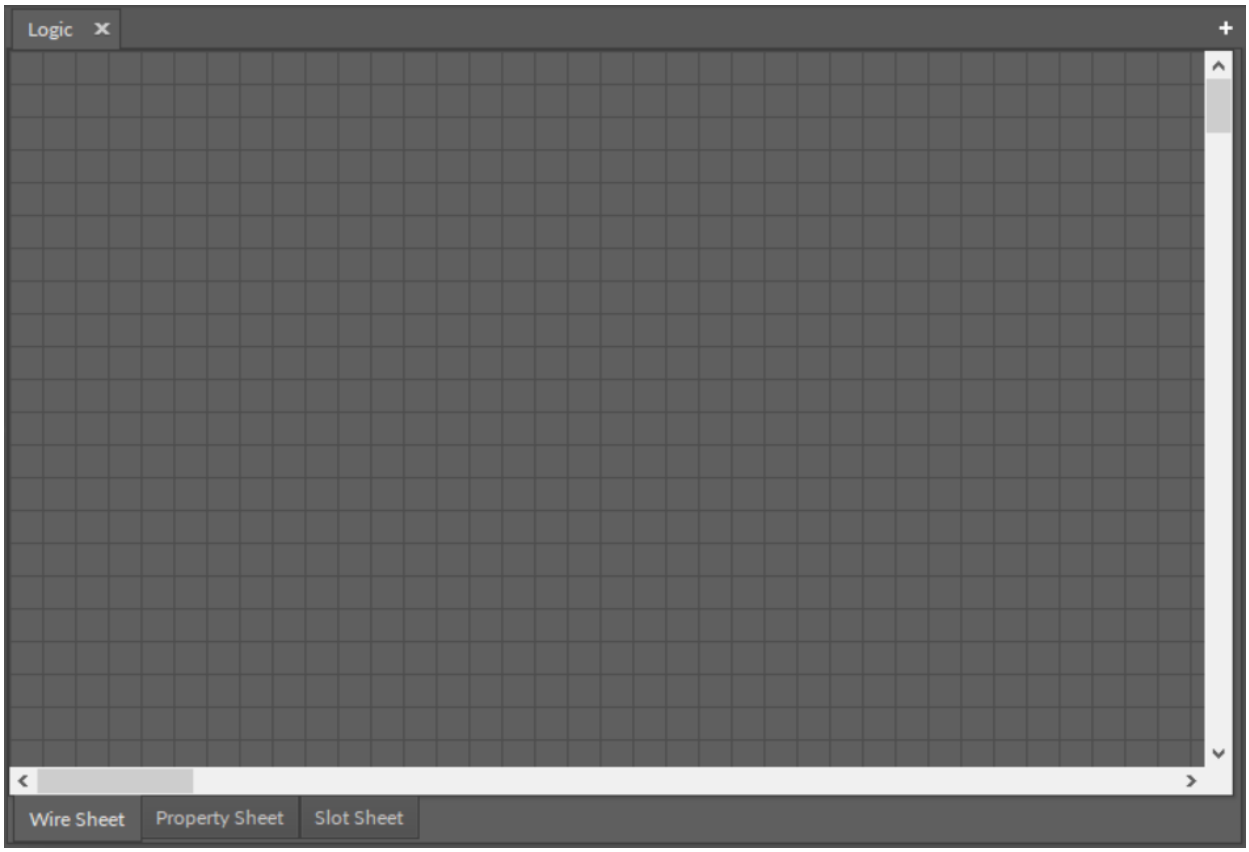



*Figure 38. The Main Screen window (Sedona devices)*

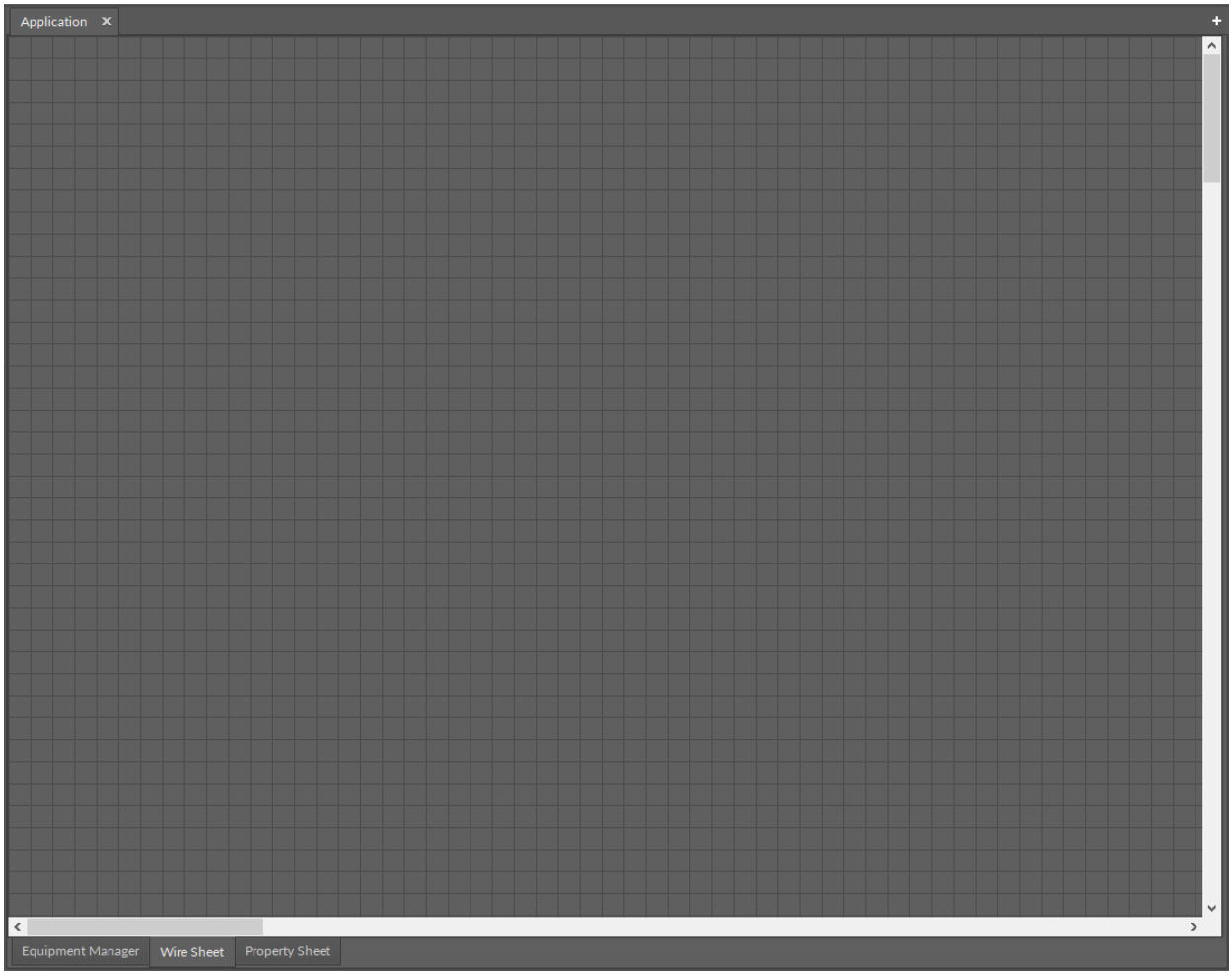



*Figure 39. The Main Screen window (nano EDGE ENGINE devices)*

To show the component data in the Main screen it is necessary to:

- Double-click a component of interest, which opens in the Main Screen; once the user opens and saves a view for a component, it is remembered, and next time the component is opened in the Main Screen last-remembered view.
- Open the context menu of a component, hover over the Views option, and choose a required view from the menu. For most components in nano EDGE ENGINE devices, there are two views available: the Wire Sheet and the Property Sheet; for Sedona devices there are three available views: the Wire Sheet, Property Sheet, and Slot Sheet.

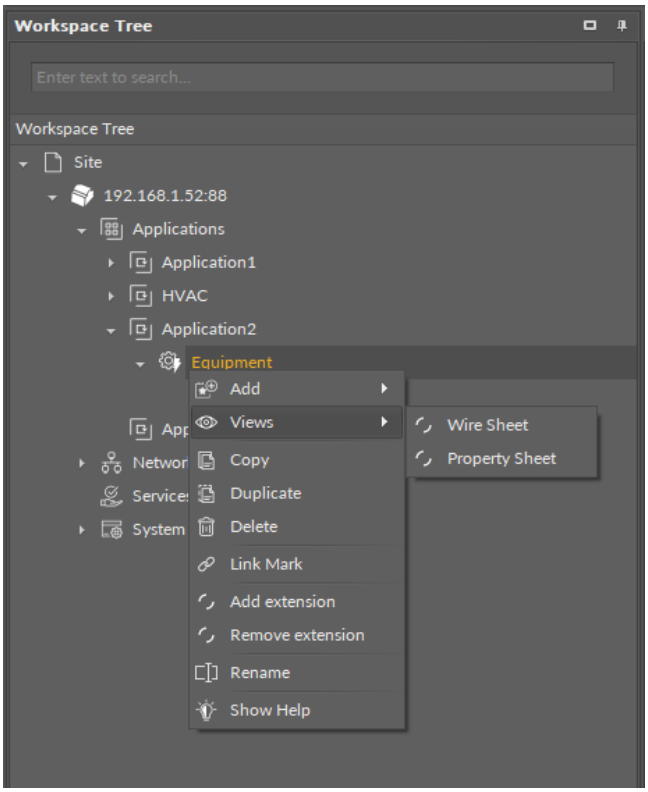



*Figure 40. The context menu views options (for the nano EDGE ENGINE device)*

### 5.5.1 Switching Between Tabs

Once the view is opened in the Main screen, switching between views is possible by selecting a proper tab in the bottom part of the window. The change of view is automatically remembered as a view dedicated for a particular component type.

In case one component is opened, choosing another component to edit (using methods described above) replaces all views with data saved earlier for the newly opened component. The default view is the one, which was used recently, or is set by default for the type of component opened.

In order to avoid replacing the data in the tabs, there is a concept of a workspace environment–a new tab, allowing to work on a few components independently, without switching the data in the tabs. To open another, independent tab, use key combination–Ctrl+T, or press an icon +, available in the right corner of the tabs bar in the upper part of the Main Screen window.

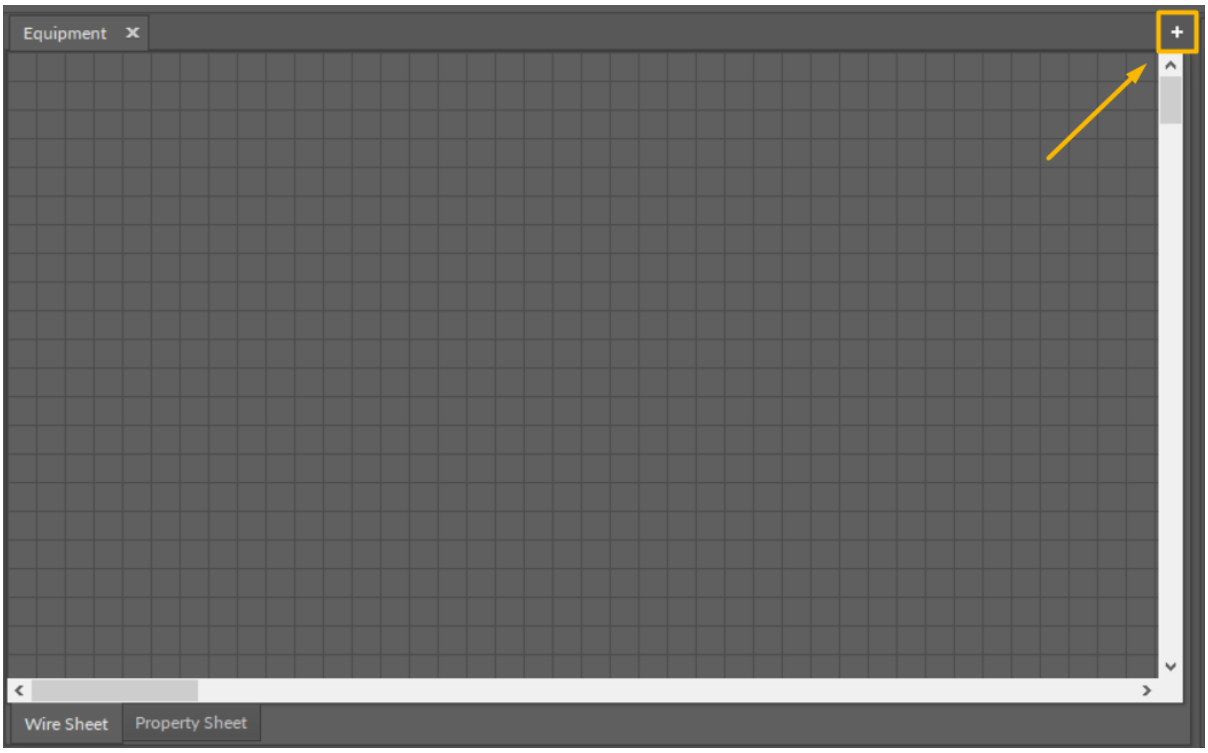



*Figure 41. Opening an independent tab*

## 5.6 Object Properties

The Object Properties window shows specific information about a selected element: a device, component, or link. It allows a quick preview and editing of a selected element's data without the need for switching, or closing, the current view in the iC Tool. The object can be selected in any view and the Object Properties window will show its detailed information.

The view and contents of the Object Properties window may differ depending on the selected element. The look of Object Properties window is described in this section, depending on the type of the object selected.

The Object Properties window, as any other window in the iC Tool, may be freely moved around the screen and mounted in other windows, which allows to adjust the window to the user's preferences. It is also possible to place the Object Properties window on another connected screen if the user is working on multiple computer monitors.

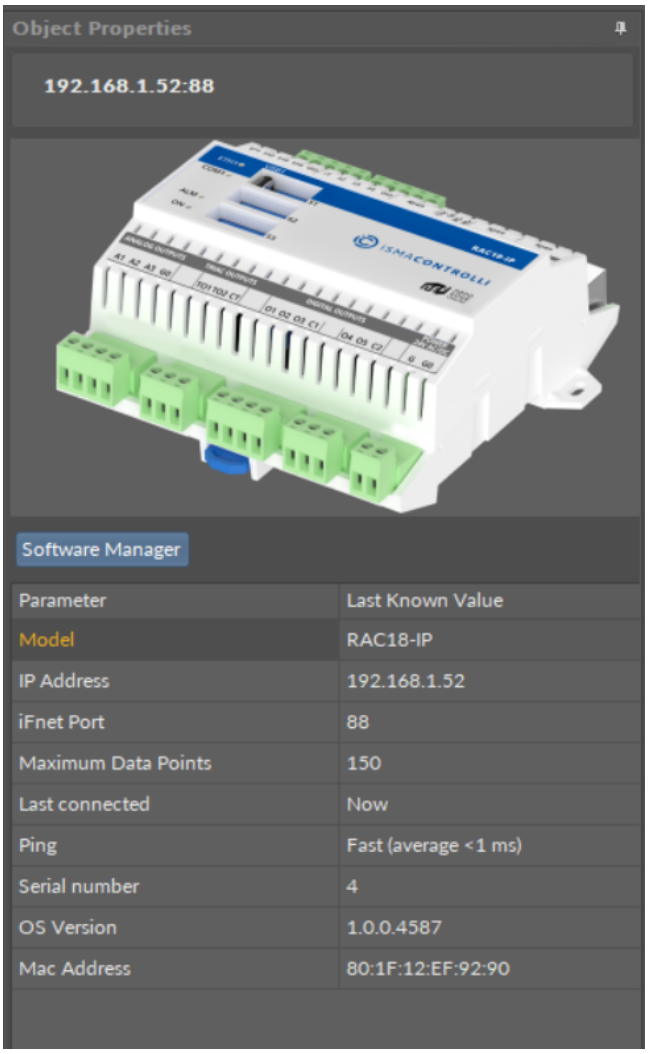



*Figure 42. The nano EDGE ENGINE device in the Object Properties window*

## 5.6.1 Device in Object Properties

### nano EDGE ENGINE Devices

After selecting a nano EDGE ENGINE device in the Workspace Tree window, the Object Properties window presents useful data about a device:

- picture of a device;
- model of a device;
- IP address of a device;
- iFnet port number;
- Data Points available within the license limit;
- last connected;
- ping;
- serial number of a device;
- OS version;
- MAC address.

Information contained in the Object Properties window is available even after disconnecting the device (in offline mode). No information about the device, presented in the Object Properties window, is editable.

The Object Properties window includes also a button to open the Software Manager.

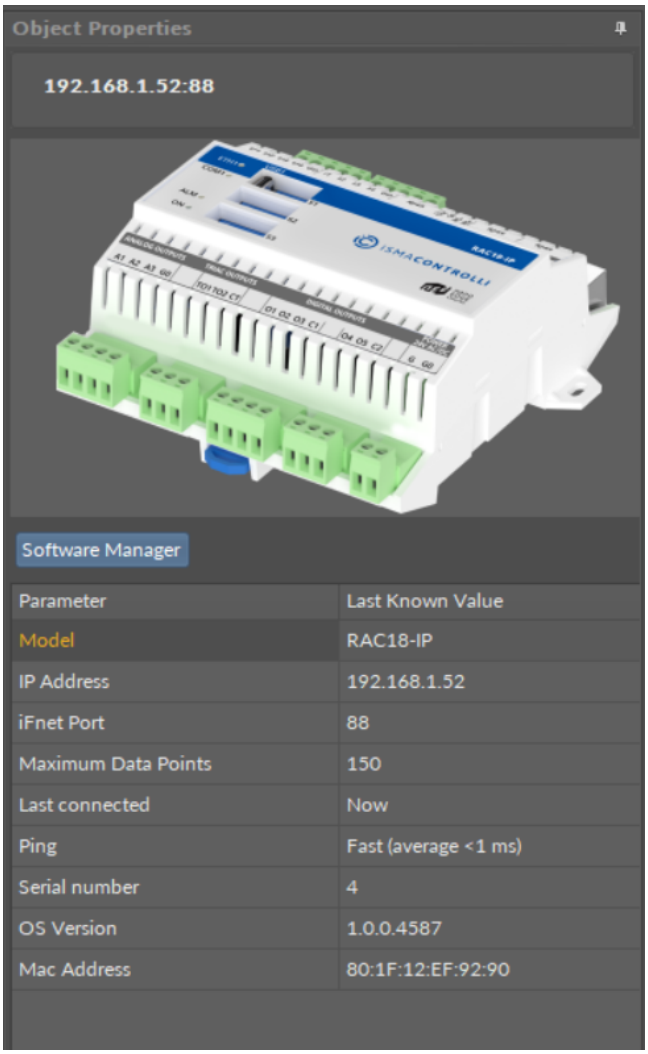



*Figure 43. Selected nano EDGE ENGINE device in the Object Properties window*

## Sedona Devices

After selecting a Sedona device in the Workspace Tree window, the Object Properties window presents useful data about a device:

- picture of a device;
- license status of a device;
- last connected;
- quality of a connection;
- installed application name;
- serial number of a device;
- firmware version;
- model of a device;
- IP address of a device;
- port number;
- MAC address.

Information contained in the Object Properties window is available even after disconnecting the device (in offline mode). No information about the device, presented in the Object Properties window, is editable.

The Object Properties window includes also buttons to open the Application Manager, Kit Manager, and Licence Manager.

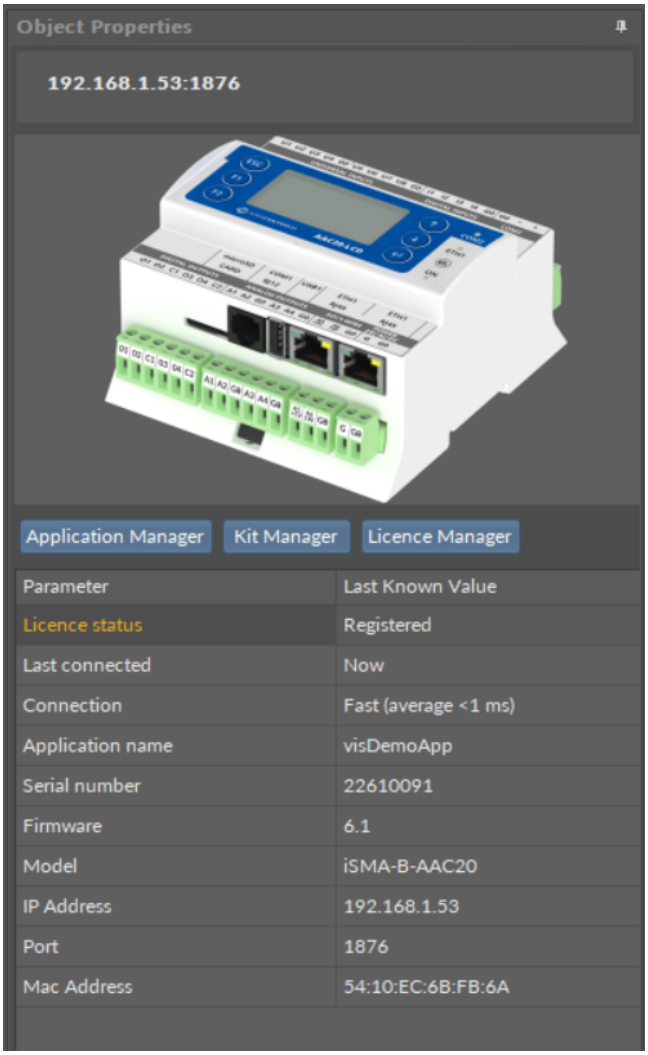



*Figure 44. Selected Sedona device in the Object Properties window*

## 5.6.2 Component in Object Properties

After selecting a component, the Object Properties window shows four sections:

- section with a slot list (the Main tab);
- section with a list of outgoing or incoming links to the selected component (the Links tab);
- section with a list of configuration data of the selected component (the Info tab)–only available in the Developer mode;
- section with Action buttons (blue buttons above tabs).

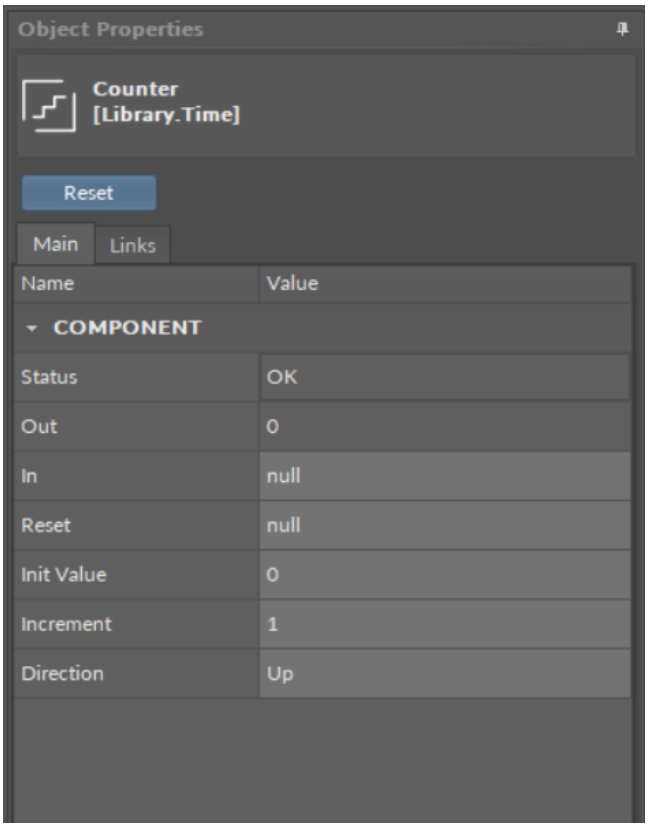



*Figure 45. The component's view in the Object Properties window*

In the upper bar of the window, it shows the name and type of element, which data is displayed.

- **Main tab:** allows reading and editing of the selected component's slots. If the user changes the value of a slot in the Object Properties window, new value is immediately saved in the controller. The Object Properties window works in the auto save mode.

> **Worth to Notice:**
>
> This section is also available to components in libraries (selected in the Device Libraries/Kits window), and allows to check the slots offered by a particular component even before using it in an application.

- **Links tab:** informs about all incoming and outgoing links with a full location path of target components and the names of target slots. For each link, it is possible to choose the following options from the context menu:
  - **Go to link:** transfers to the Wire Sheet view of a superior of a target component;
  - **Go to component:** transfers to the Property Sheet view of a target component;
  - **Delete link:** removes a chosen link.

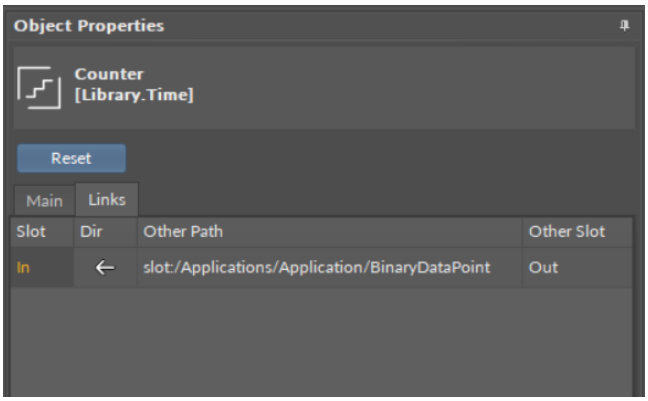



*Figure 46. The Links tab view in the Object Properties window*

- **Info tab:** (available only in the Developer Mode) shows the identification and configuration data of the selected component;

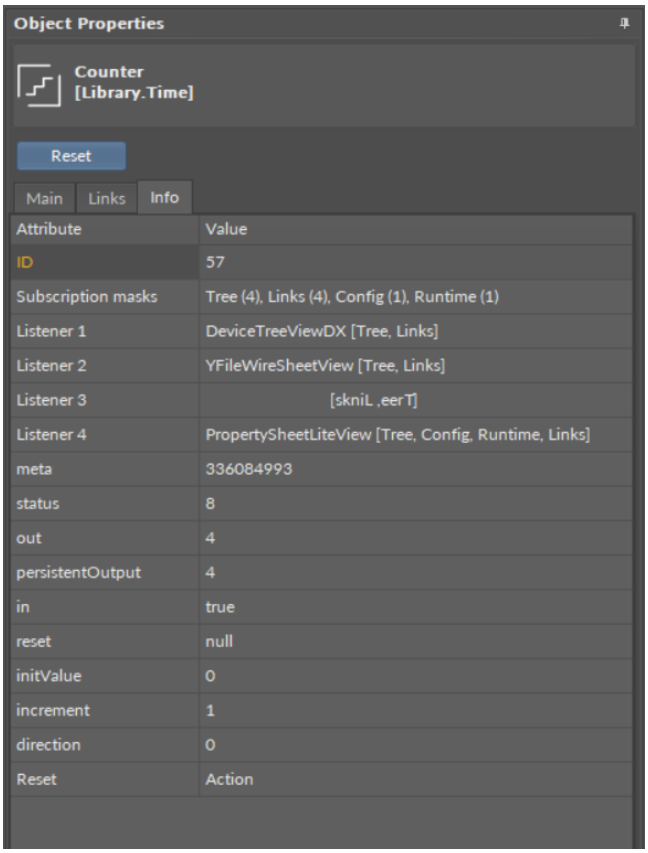



*Figure 47. The Info tab view in the Object Properties window*

- **Actions section:** buttons available in the bar above tabs, allowing a simple execution of actions on selected component. These are the same actions which are listed in the component's context menu under the Actions option.

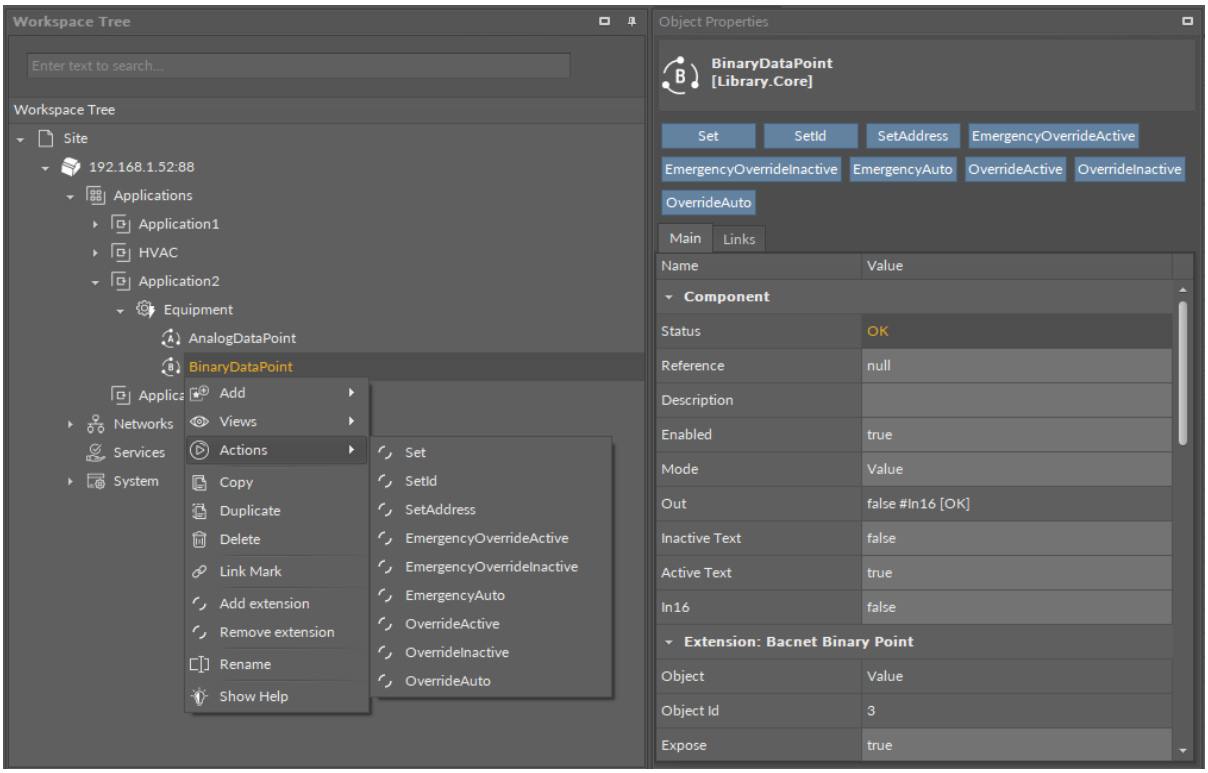



*Figure 48. The Actions section in the Object Properties window*

## Many Components in Object Properties

The iC Tool offers the possibility to simultaneously display common properties of many selected components.

Each time the user selects more than one component, the Object Properties window changes its header to show the number of simultaneously selected components.

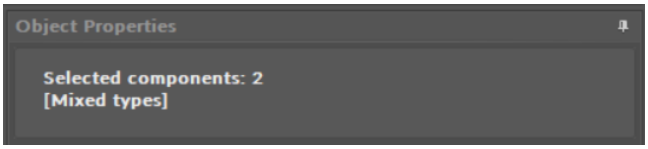



*Figure 49. The Object Properties header for two selected components*

### Common Slots

While working with many components simultaneously, the Object Properties window changes its work mode to show only the slots common for all selected components. The value of the slot shown in this mixed view is only displayed if all selected components have exactly the same value for this slot; otherwise, the value field is empty.

Regardless of whether the shown slot displays a value or not, if it is an editable slot, entering a new value overwrites it in all selected components. If this process is not successful, the Console will show an appropriate information.

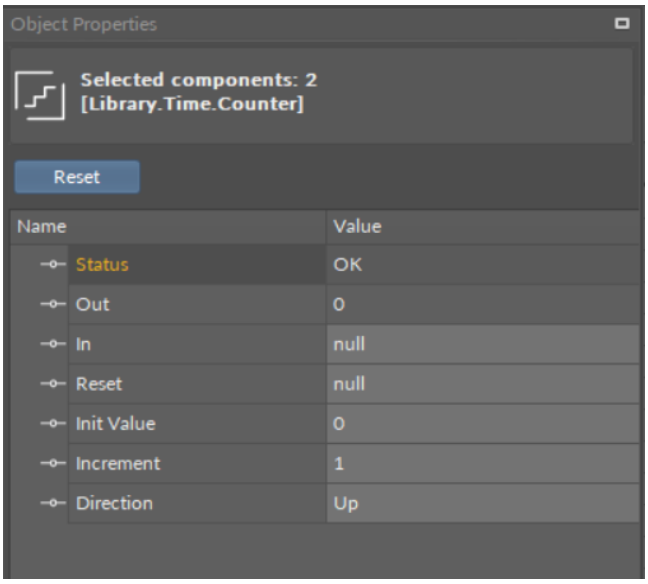



*Figure 50. Common slots of same type components*

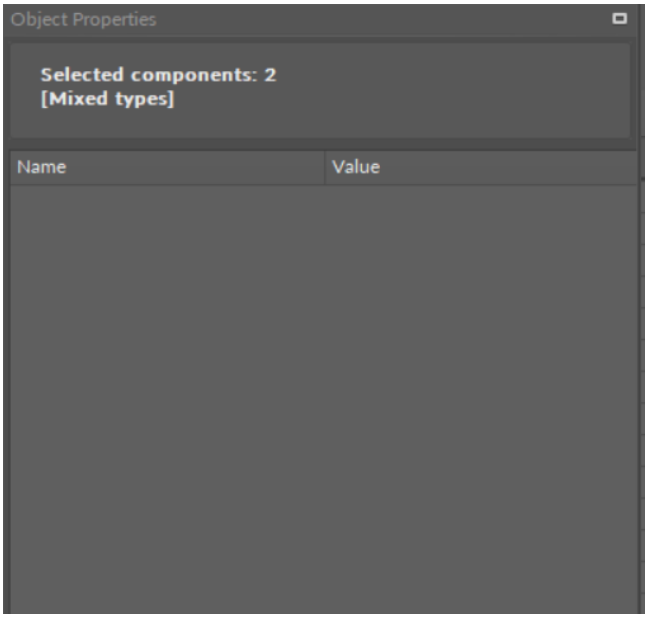



*Figure 51. Common slots of different type components*

Using this multiselect function for an editable slot enables entering a new value (e.g., In B), which will be sent to all components at once. Setting the same values for various components simultaneously allows setting common setpoints or configurations for many components at once, enabling to activate network variables, etc.

### Common Actions

The case of common actions is similar to setting a new value for common slots. Assuming the selected components share the same actions, then the buttons of these actions appear in the Object Properties window. Pressing the action button will execute the action in all selected components.

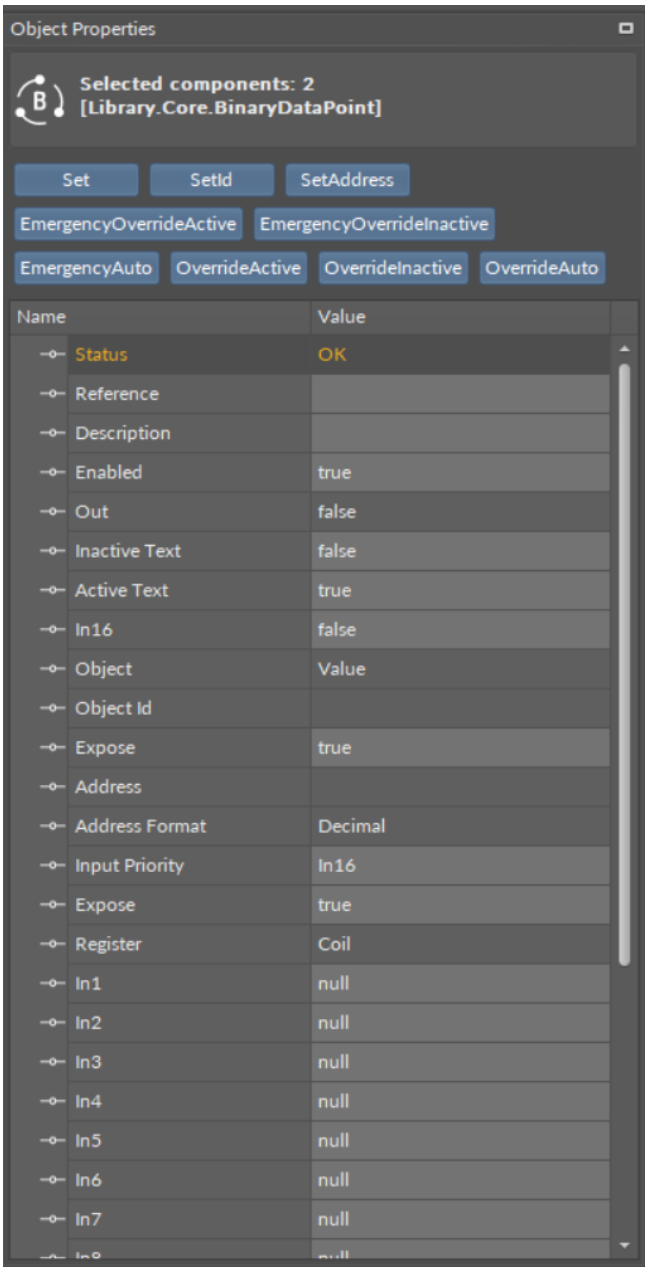



*Figure 52. Common actions of same type components*

### 5.6.3 Links Tab in Object Properties

A Links Tab view is a view showing a list of all links incoming and outgoing of the selected component in the Object Properties view. The link's full location path is displayed in the Links tab, including the name of the target slot.

This view is a dynamic one, and it is updated constantly by downloading data directly from the device.

The figure below shows the component selected in the Wire Sheet view. The Object Properties window has instantly shown all its links in the Links tab.

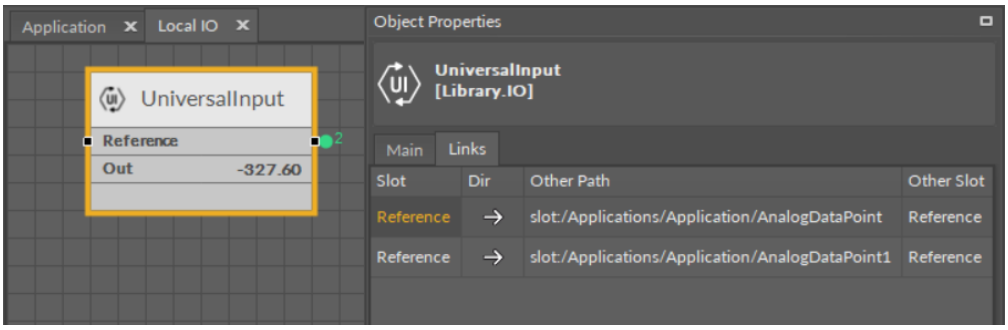



*Figure 53. The Links Tab*

Each link in the Link tab has the context menu available with the following options:

- **Go to link:** go to the Wire Sheet tab of the parent of the target component;
- **Go to component:** go to the Property Sheet tab of the target component;
- **Delete link:** remove a chosen link.

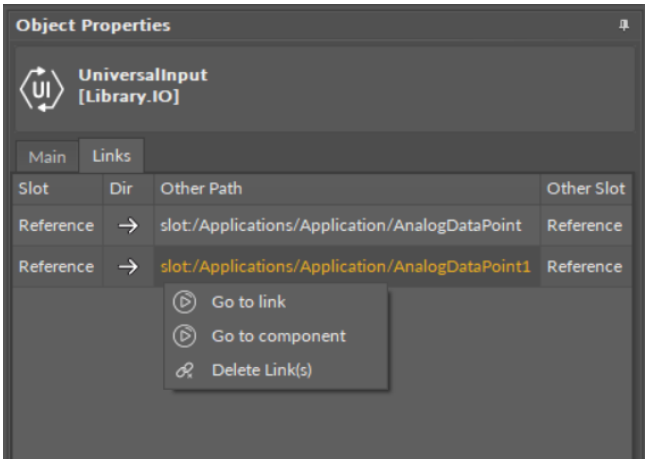



*Figure 54. The context menu in the Links Tab*

## 5.6.4 Object Properties in Developer Mode

The Object Properties window may provide advanced data such as a component ID, version of a component's library, and other technical data useful to an advanced user. These data are available in the Info tab, which appears in the Object Properties window after switching on the Developer Mode in the iC Tool settings. To switch on the Developer Mode, go to the main menu File > Settings; the Developer Mode is switched on by toggling in the Settings window.

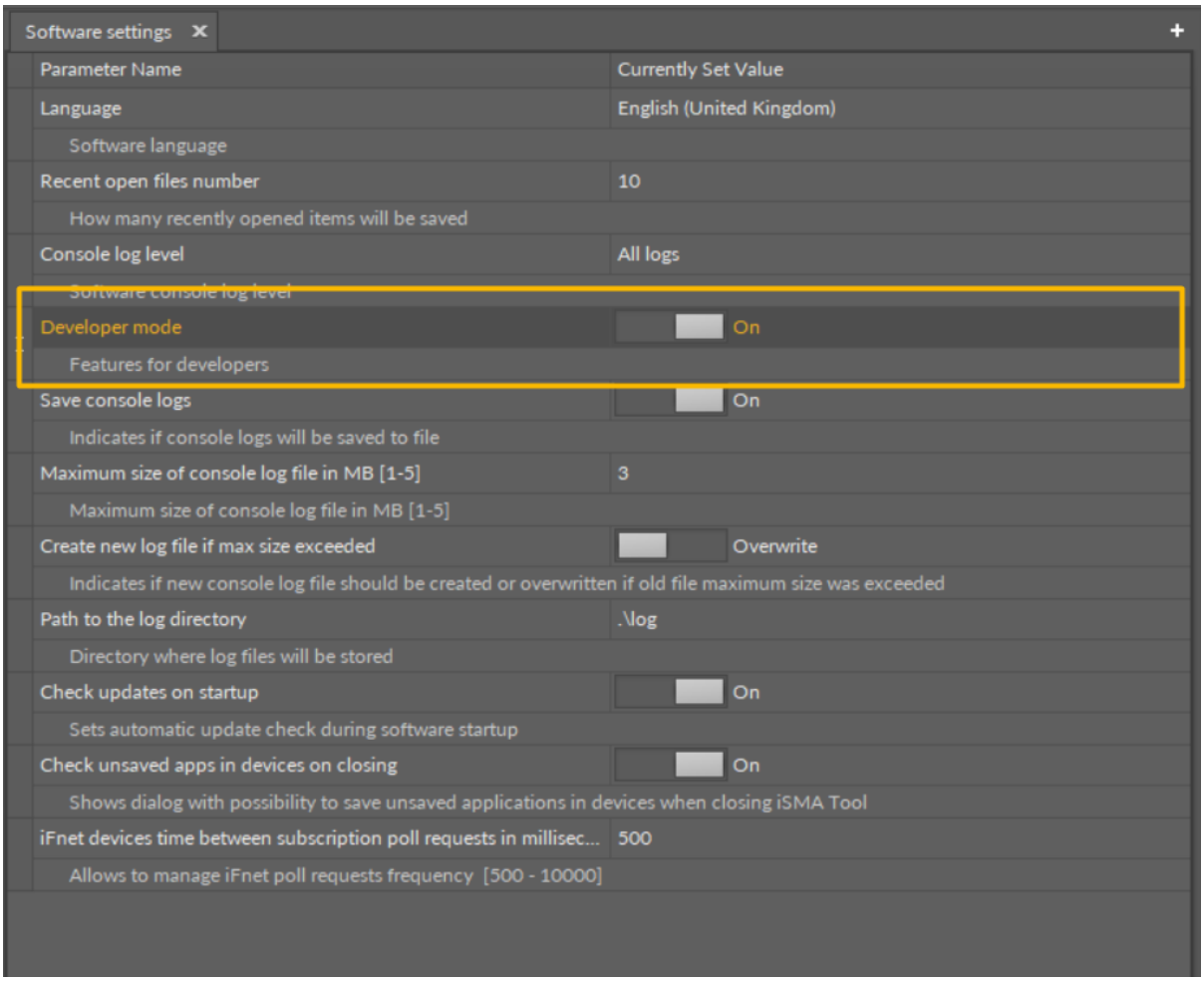



*Figure 55. Turning on the Developer Mode*

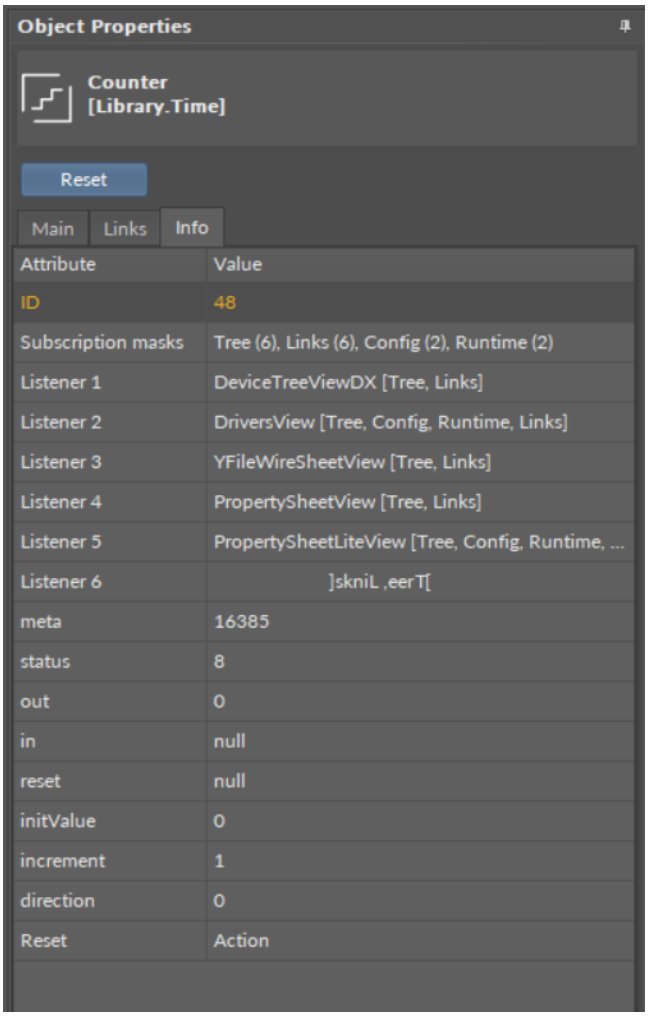



*Figure 56. The Info tab in the Object Properties window*

## 5.7 Console and Logs

The Console is a window where the iC Tool informs the user about the application status and its operations. By default, the Console window is minimized. It is accessible as an expandable tab in the bottom left corner of the iC Tool.

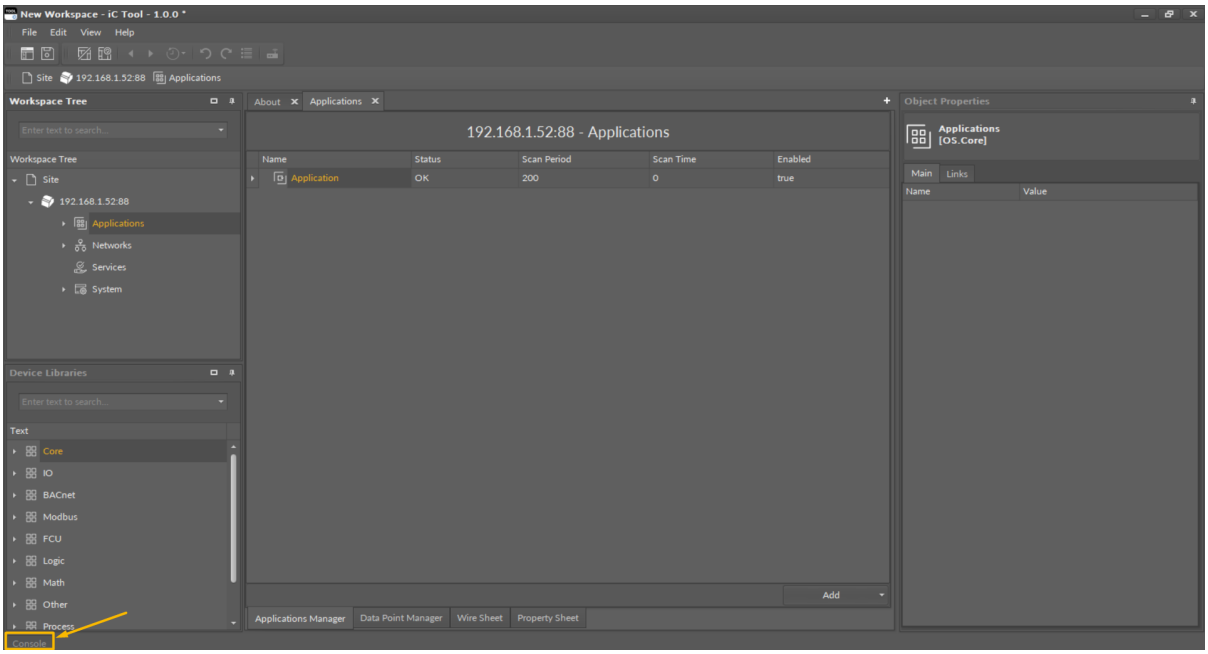



*Figure 57. Accessing console in the bottom left corner of the view*

While performing user's tasks or processes such as object removal or establishing connection with a controller, the Console window shows the logs of what is being performed at the moment.

The basic concept in the iC Tool is that the Console shows all detailed messages, while only some of them are additionally displayed as a pop-up window.

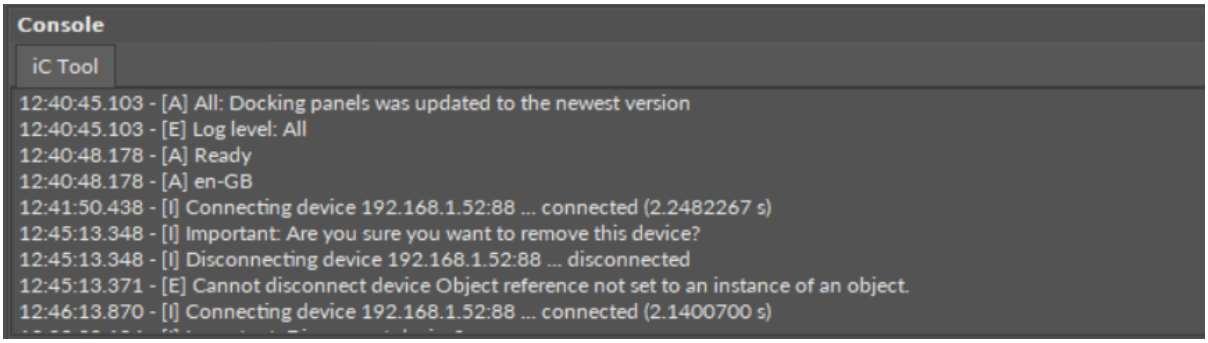



*Figure 58. The Console view*

The Console window contents are automatically saved in the log file to store the information for a longer period of time. It is therefore possible to open the log file for a later analysis using any text editor. The log files are by default saved in the log folder. Both the name of the folder for saving logs and its location may be changed by defining a path in the Settings (see the description further in this section).

The iC Tool allows to define the size of a single log file, as well as the course of conduct after exceeding the maximum log file size. The file size may be from 1 MB to 5 MB (by default it is 3 MB), and there are two possible courses of conduct after exceeding the maximum log file size:

- creating a new log file and keeping previous files;
- complete wiping of the actual log file and start recording again.

The above settings can be modified in the Software settings in the main menu File -> Settings.

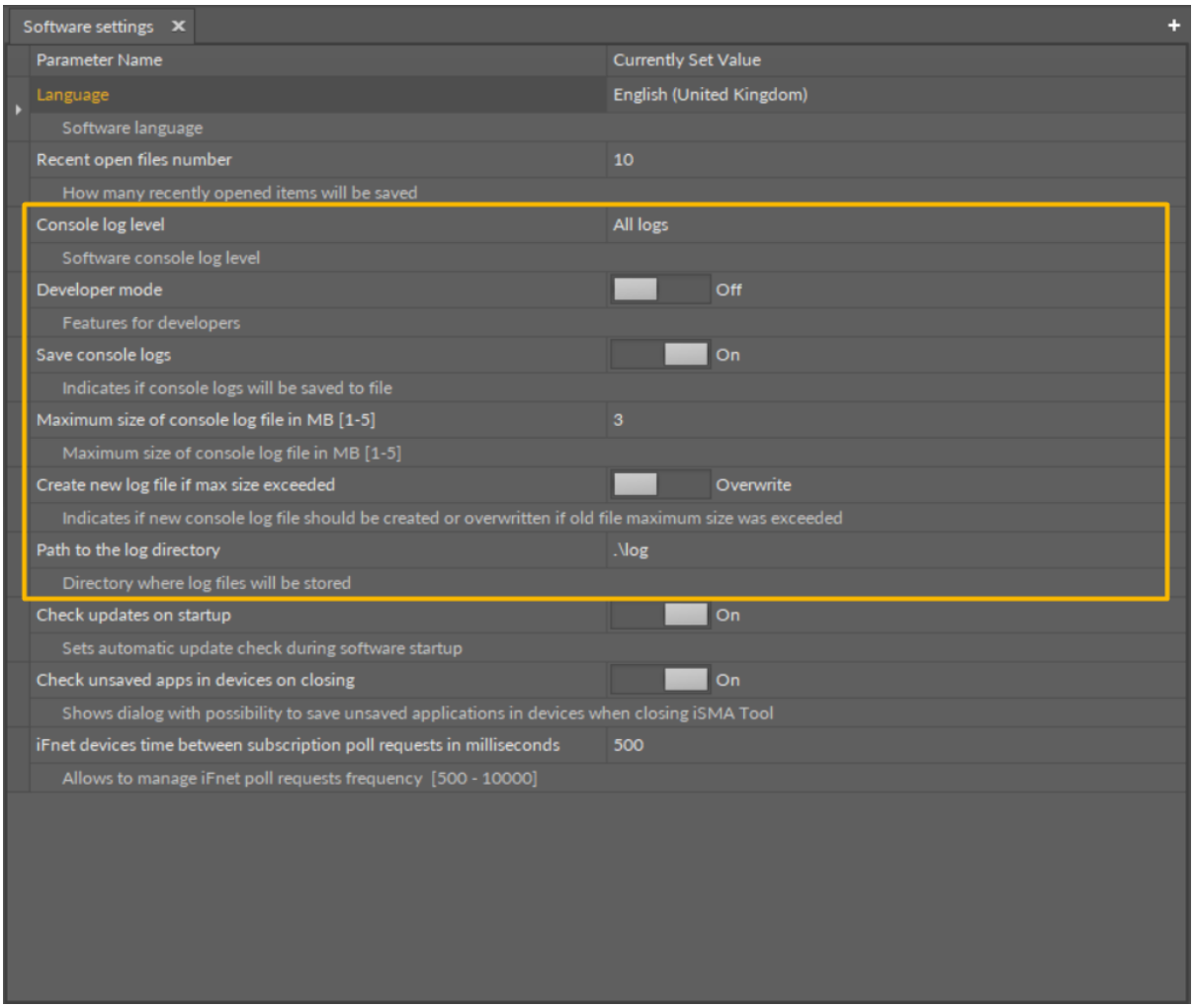



*Figure 59. The available Console and logs settings*

All settings applicable to logs and the Console window are the following:

- **Console log level:** a filter defining a kind of logs to be displayed in the Console window (the listed options provide the same result as given by the letter filter, D, A, I, and E, available at the bottom of the Console window);
- **Developer mode:** allows hiding/showing of the D filter–Debug button in the Console window;
- **Save console logs:** turns on/off recording of the log to a file saved on a hard drive;
- **Maximum size of console logs file:** defines the maximum size of the log file;
- **Create new log file if max size exceeded:** a switch defining how the system acts after exceeding the maximum log file size;
- **Path to the log directory:** allows defining of the log file saving location.

### 5.7.1 iC Tool Log Viewer

The iC Tool allows to view its log file saved on the hard drive. To see the iC Tool logs, go to the main menu, click the View option, and then the Log Viewer option.

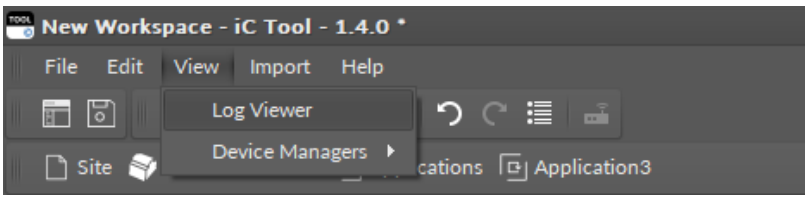



*Figure 60. Accessing the Log Viewer*

The Log Viewer opens in the Main screen. Once the listed log file is double-clicked, it is opened in a default .txt program.

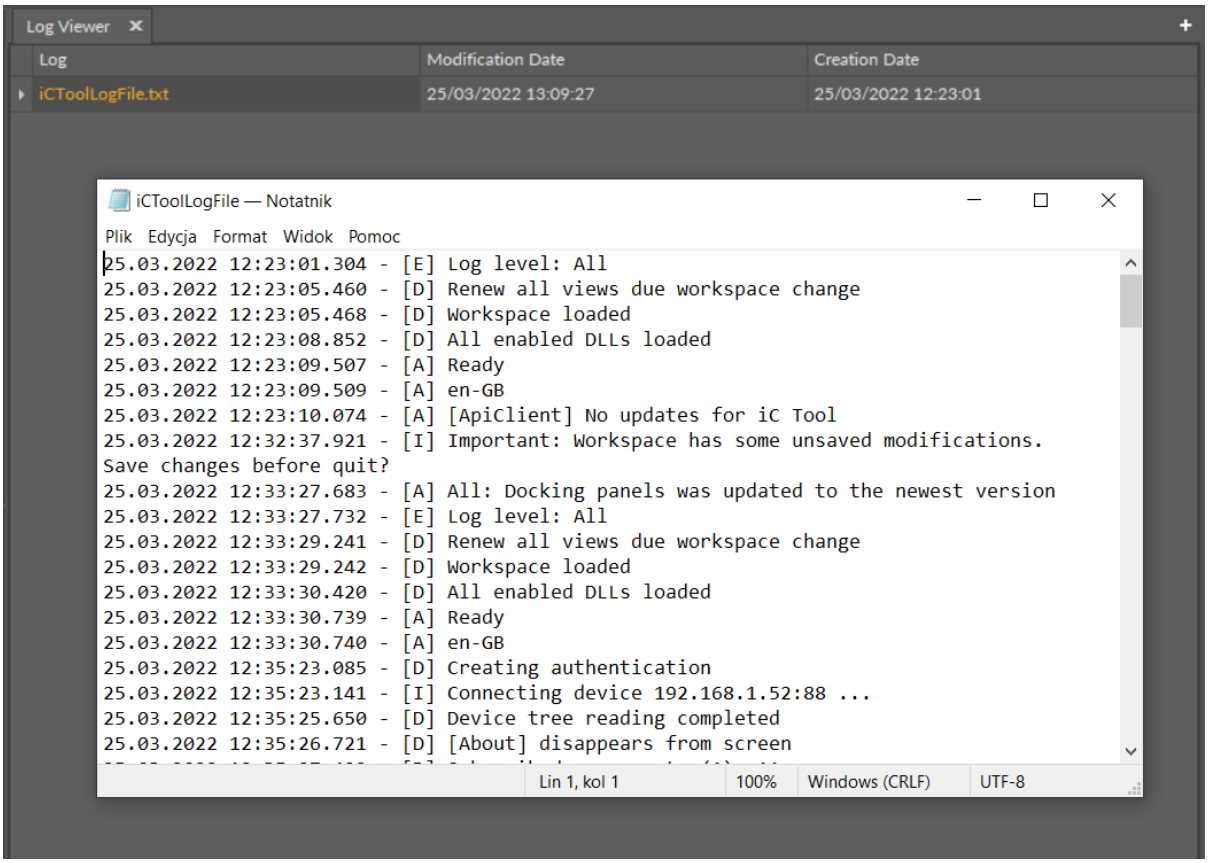



*Figure 61. The iC Tool logs file*

# 6 Single Device Functions

The iC Tool offers many functions for handling of components such as copying, duplication, removing, linking, performing actions, etc.

The functionalities described in this section are global, they apply in the following views: the Wire Sheet, Property Sheet, and Workspace Tree window. Individual operation of global functions, separately for each standard view, is described in the Standard Views section.

## 6.1 Context Menu

A basic way of initiating global functions is using a context menu opened on the components of interest.

The context menu is displayed after selecting a component or link and right-clicking it.

The context menu content depends on the type of object, which has been selected. The menu's content will be different for one component or link, different after selecting many components or many links, and different when components and links are selected together at the same time.

Only active options are displayed in the menu, for example, if no component or values have been copied, the Paste option will not be displayed.

The options available in the context menu for a component are the following:

- **Add:** allows to add a component (Analog Data Point, Binary Data Point, Equipment, or Folder) in the Applications container (option inactive in other containers);
- **Views:** allows to display component's data in one of the defined standard views (Wire Sheet, Property Sheet), or in other views if they are available for a particular component;
- **Actions:** shows a list of actions that may be evoked for the given component (such as Override or Set actions);
- **Copy:** (shortcut Ctrl+C) remembers and copies a selected component along with all its properties, settings, and links information;
- **Copy Slots Values:** copies all slots values from a selected component (without links) (active only for the nano EDGE ENGINE devices);
- **Paste:** (shortcut Ctrl+V) pastes a previously remembered component into a specific place and possibly recreates internal links;
- **Paste Special:** same as the Paste option, plus recreating of incoming external links, if possible;
- **Paste Slots Values:** pastes previously copied slots values to a component of the same type as the source component, for example, slots are copied from the Loop component and pasted into another Loop component (active only for the nano EDGE ENGINE devices);
- **Duplicate:** (shortcut Ctrl+D) duplicates the selected components in the same location;
- **Delete:** (shortcut Del) removes the selected components;
- **Link Mark:** allows to define the component from which a link will be led;
- **Link From:** allows to define the component to which a link will be led;
- **Add extension:** allows to add an available extension to a component (active only for nano EDGE ENGINE devices);
- **Remove extension:** removes a component's extension (active only for nano EDGE ENGINE devices);

- **Reorder:** allows to reorder components in the Workspace Tree (active only for Sedona devices);
- **Rename:** allows to change a component's name;
- **Show Help:** opens a help online web page for a selected component (active only for nano EDGE ENGINE devices).

The figure below shows the context menu, which appears after right-clicking a single component. Options Paste and Paste Special are inactive because a Copy option has not been used. Similarly, Link From option is inactive because the source component for a link has not been defined by a Link Mark option.

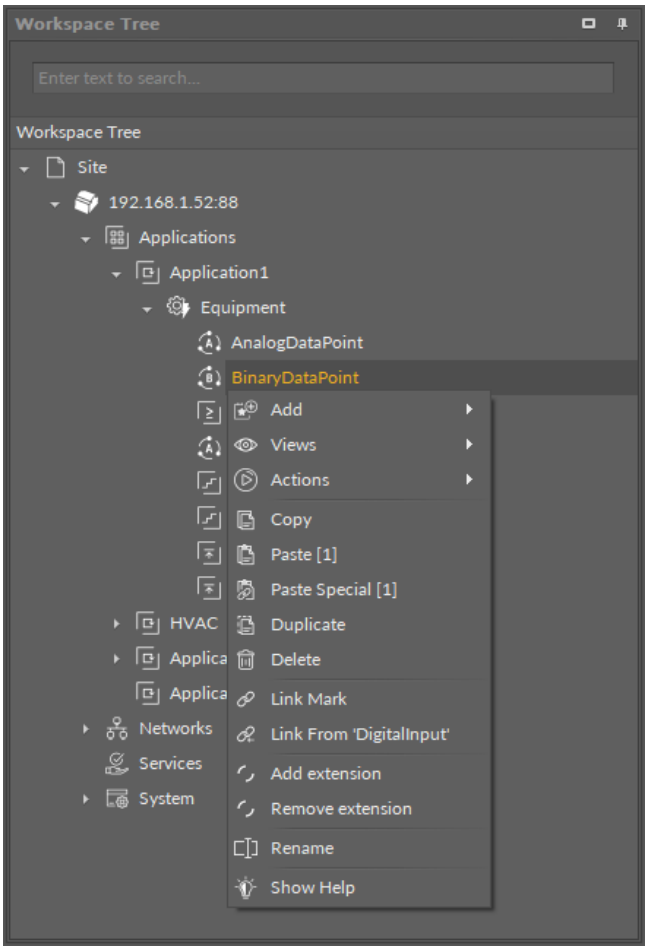



*Figure 62. The context menu of a single component*

For links, the context menu appears after right-clicking a link. Apart from removing it, no other operation can be performed on a link.

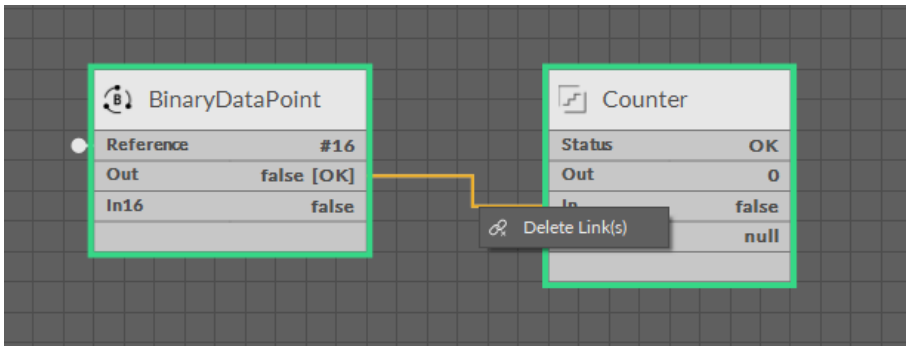



*Figure 63. The context menu of a link*

After selecting many components or mixed objects (component with link) most of the options are inactive because the iC Tool cannot perform operations on many objects simultaneously. Options available in this case will be Copy, Copy Slots Values, Duplicate, and Delete, which will be performed on all selected objects.

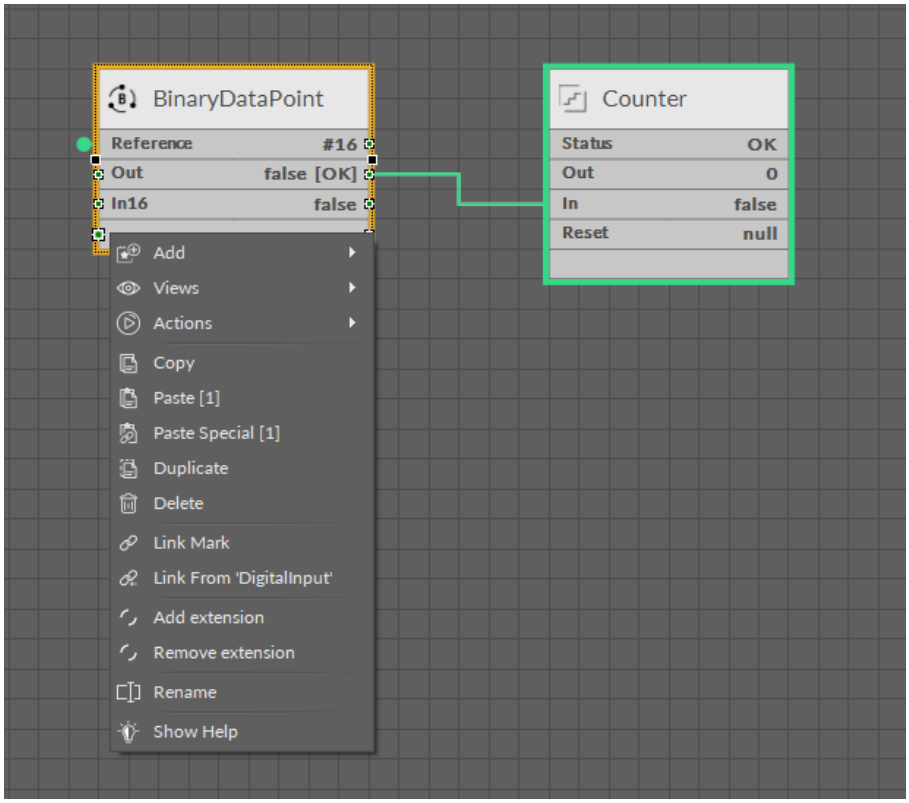



*Figure 64. The context menu of multiple components*

## 6.2 Component

A component is the smallest logical and functional unit that requires entering specific data to perform properly.

The component may be, for example, an arithmetic function of adding two numbers, therefore, it requires information about the numbers to be added. These data may be provided, for example, by other components able to read physical universal inputs of a controller, and transfer numeric data to be added.

In various iC Tool views the component is presented differently, showing various information, as depicted on the figures below. In the Wire Sheet view the component is shown as a rectangle with a limited amount of data displayed, and in the Property Sheet view it is a table which lists all component's slots. Some of the slots in the table may be edited directly. More about the Property Sheet view and the Wire Sheet view may be found in the Standard Views section.

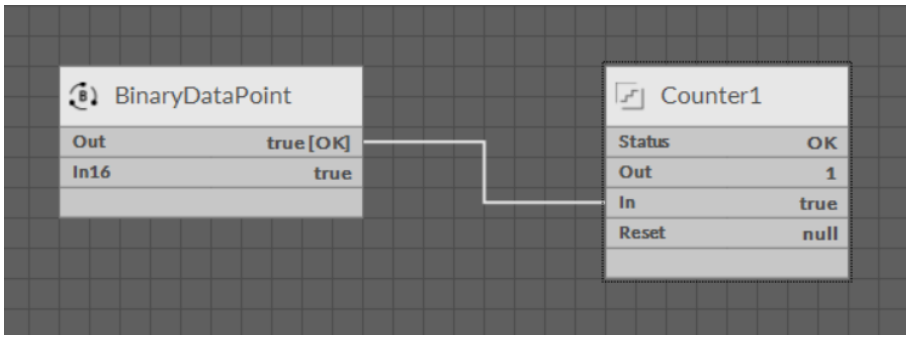



*Figure 65. Components view in the Wire Sheet*


*Figure 66. Component's view in the Property Sheet*

The component is the main element of application created in a device. Therefore the iC Tool guarantees a standardized way of issuing commands to components. Regardless of whether the component is shown in the Wire Sheet view, Property Sheet view, or Workspace Tree window, operations are performed on it in the same way.

## 6.2.1 Components' Hierarchy

Components may be organized into a hierarchy tree. The component, which has other components below, it is called a superior, and the elements below it–inferior. Hierarchy allows to organize components logically, making some components dependent on others.

In the iC Tool, hierarchy is presented in the Workspace Tree window and on the Property Sheet view as a tree, showing superior elements together with their inferior components placed underneath.

Importantly, the components' hierarchy on the Wire Sheet view is limited to one generation. After opening a superior component, the Wire Sheet view shows a schematic with all the inferior components distributed in the view. The figure below shows an opened superior component, LocalIO, with its inferior components, in the Property Sheet view and in the Wire Sheet view.

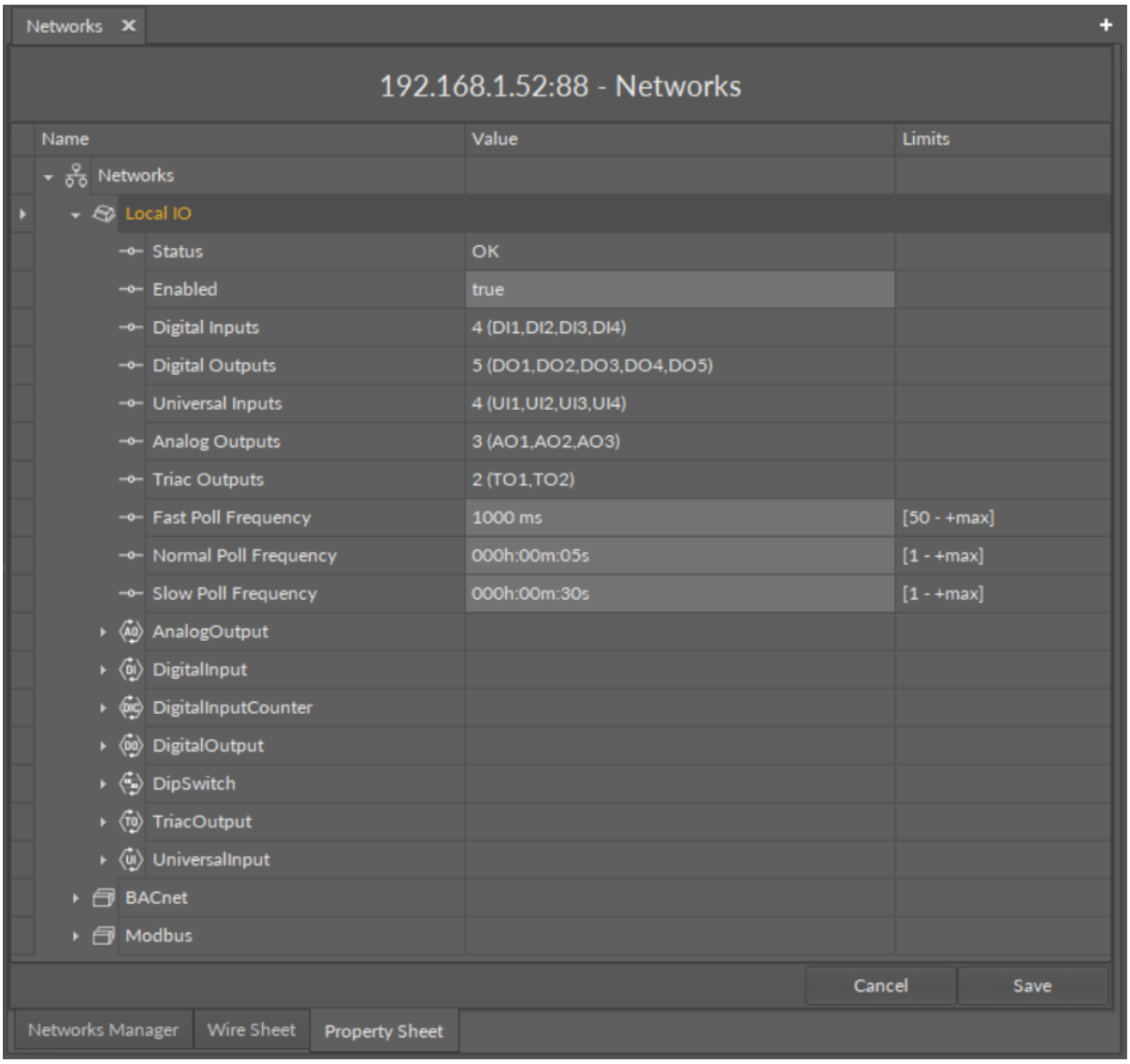



*Figure 67. Components' hierarchy in the Property Sheet*

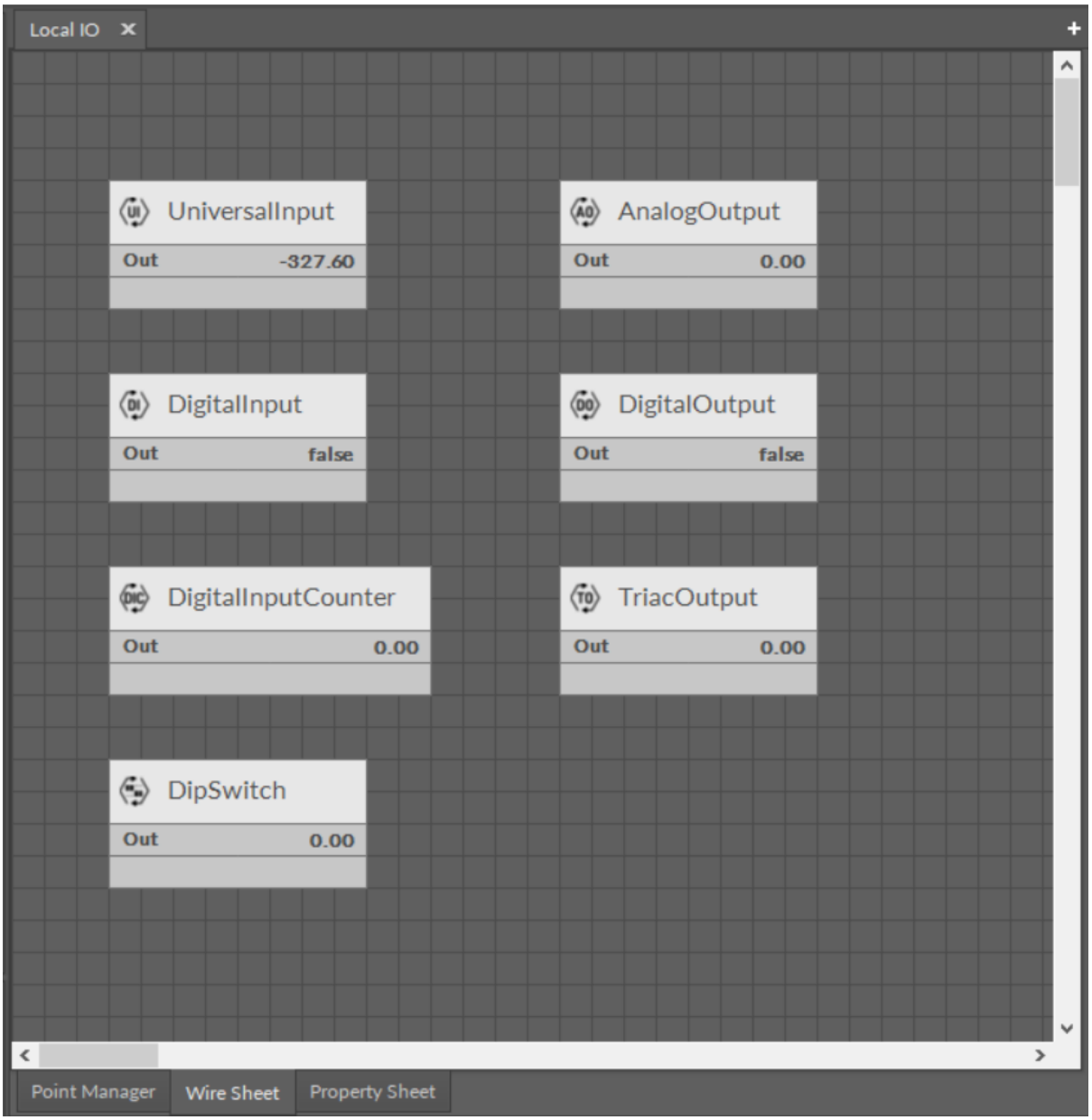



*Figure 68. Component's hierarchy in the Wire Sheet (LocalIO being the superior component, the inputs and outputs components-inferior)*

### 6.2.2 Adding New Component

New components can be added in three windows: the Workspace Tree, Wire Sheet, and Property Sheet. New component is created by:

- duplicating the existing component (using Copy or Duplicate function);
- dragging a new component directly from the Device Libraries/Kits window.

The name of the new component is given automatically; however, it can be changed manually by the user. It is possible to bulk add components of the same type by indicating a number of components to add along with a naming pattern in the dialog box (see more in Renaming Components). If several components of different types are added at the same time, the iC Tool will generate unique names to all the components added automatically.

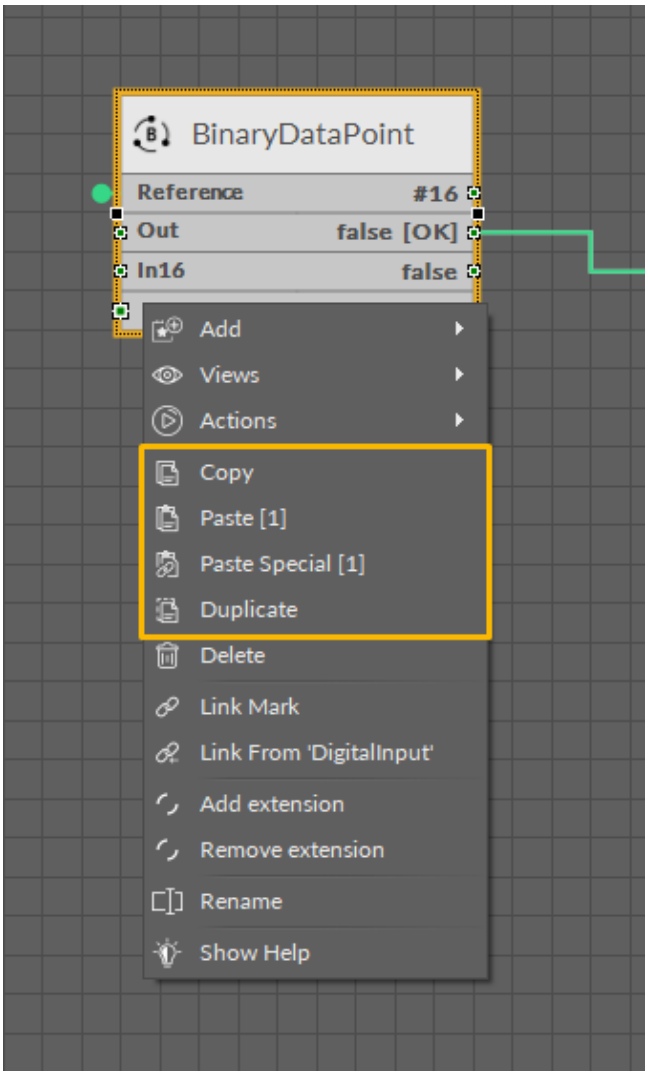



*Figure 69. Adding new components - copying/duplicating*

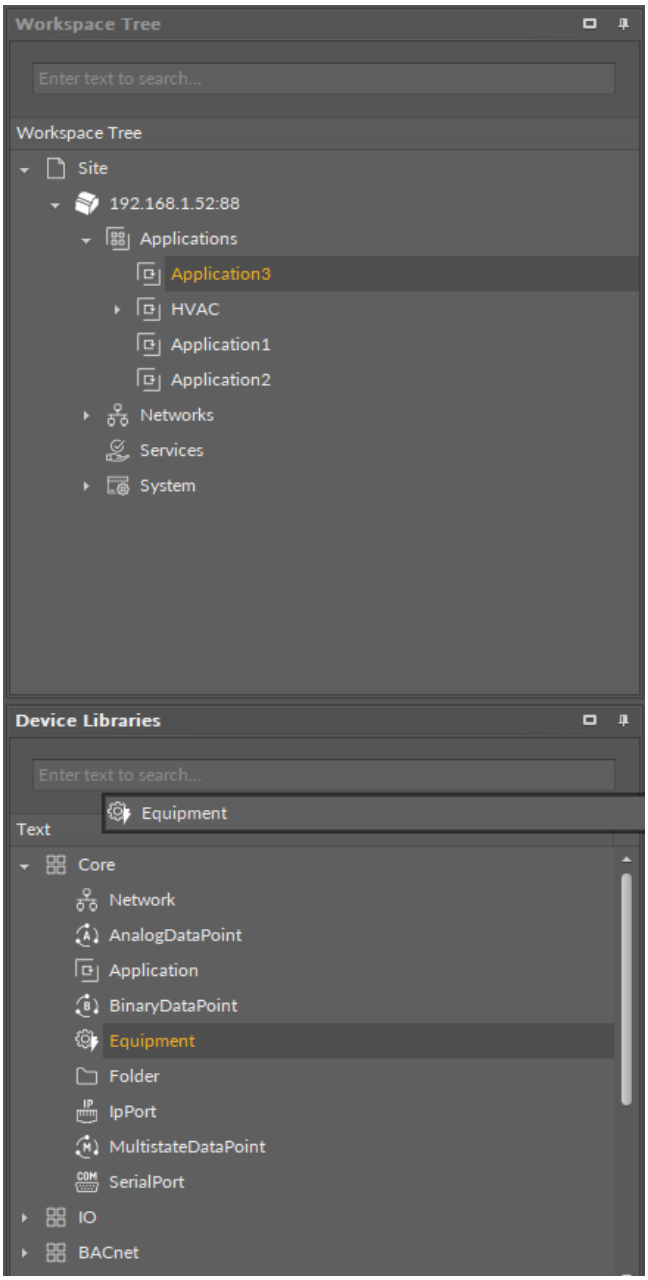



*Figure 70. Adding new components - dragging*

## Special Adding of Data Points and Folders in nano EDGE ENGINE Devices

The Workspace Tree window, the Wire Sheet and Property Sheet views offer a special way of adding Data Points and folders (the Folder and Equipment components) to the Application, simply from the context menu. In order to add components this way, right-click in an empty space of the Wire Sheet grid or Property Sheet and choose the Add option from the context menu. In the Workspace Tree special adding of Data Points and folders is available in the context menu of the Application component.

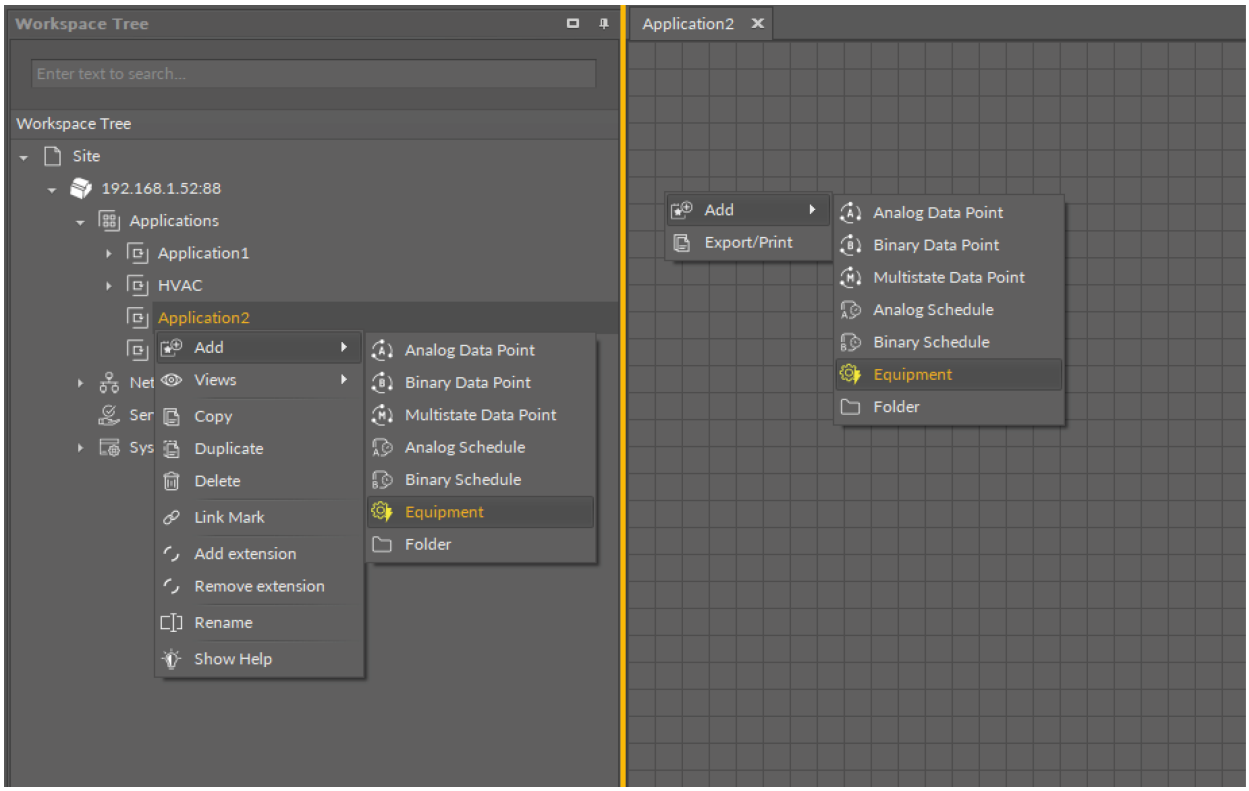



*Figure 71. Adding Data Points/folders from the context menu*

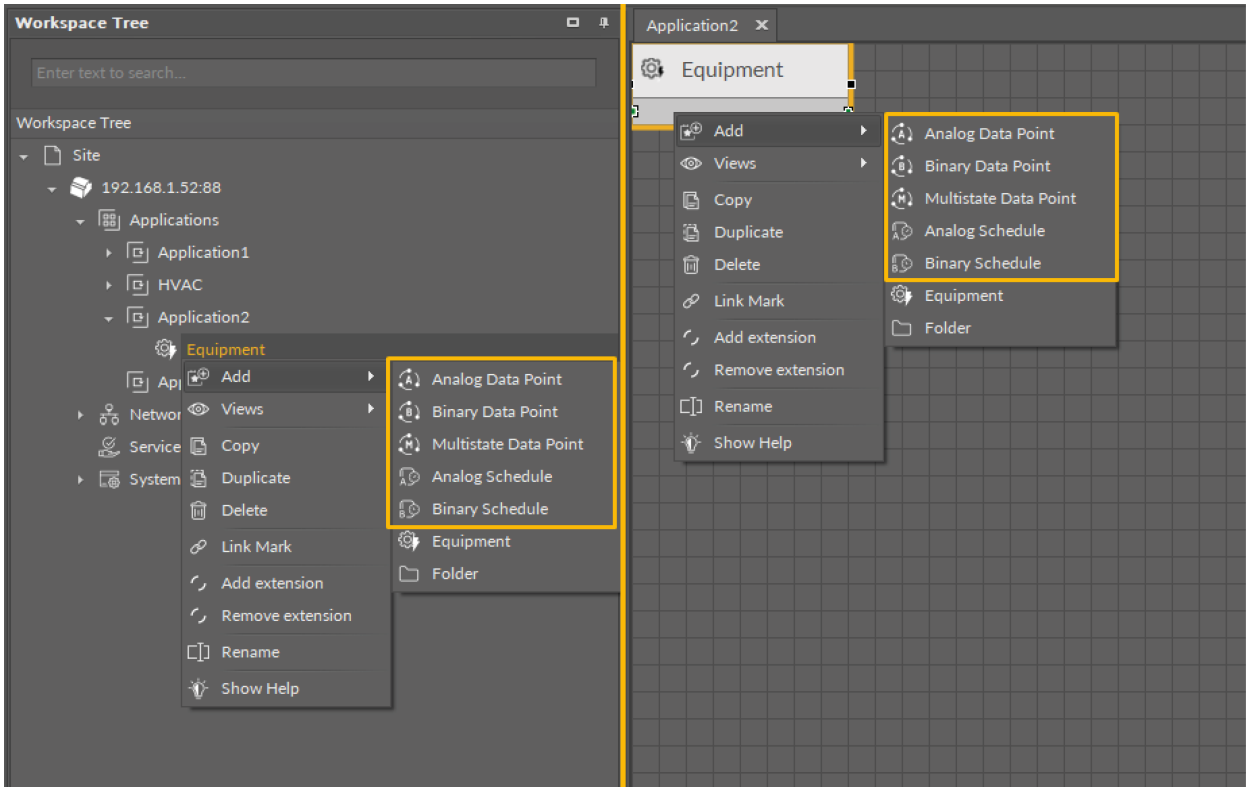



*Figure 72. Adding Data Points to folders from the context menu*

### 6.2.3 Copying, Pasting, and Duplication

## Copying

Copying in the iC Tool uses a standard Windows clipboard and may take place within the limits of one device. Copying always includes all the inferior components of the selected component.

Copying in the iC Tool requires to identify components to be copied by selecting them. When a Copy option or Ctrl+C combination is used, the elements are saved to the Windows clipboard. This allows the Paste option to be used in any moment in the future until the next element is copied, even after the components, which were copied, were deleted because the iC Tool automatically recreates the copied structure by recreating the needed components remembered in the clipboard.

Copying includes all the links created between the copied components. This means that while pasting the links between these components will be recreated. The situation is different for external links (links to components placed under a different parent/superior component). A basic Paste function does not support the recreation of external links, this needs the Paste Special function described in the following point.

**Worth to Notice:**

The iC Tool allows copying only of components; copying and pasting of links alone is impossible. Information about all component's links is stored in the component itself. It means that if a component is copied and pasted, and there is a possibility to recreate its links from a logical point of view, the iC Tool will do it automatically, adding new links between components.

## Pasting

Copied components may be pasted into a chosen place in the Wire Sheet or Property Sheet views or in the Workspace Tree window, by indicating the pasting place and pressing Ctrl+V, or using the Paste option from the context menu.

Pasting copied components creates new components along with their inferior components and slot settings effective at the moment of copying. The pasted slot values are sourced from the copied component.

Pasting function can be used many times because pasting does not erase the clipboard. The clipboard is wiped after device disconnection.

When pasting a single component, the system will ask for a new component's name (with a highlighted name suggestion and support for bulk naming patterns, see Renaming Components) and number of copies to paste, and for a multi-component operation the iC Tool will use a mechanism ensuring that the names will be unique.

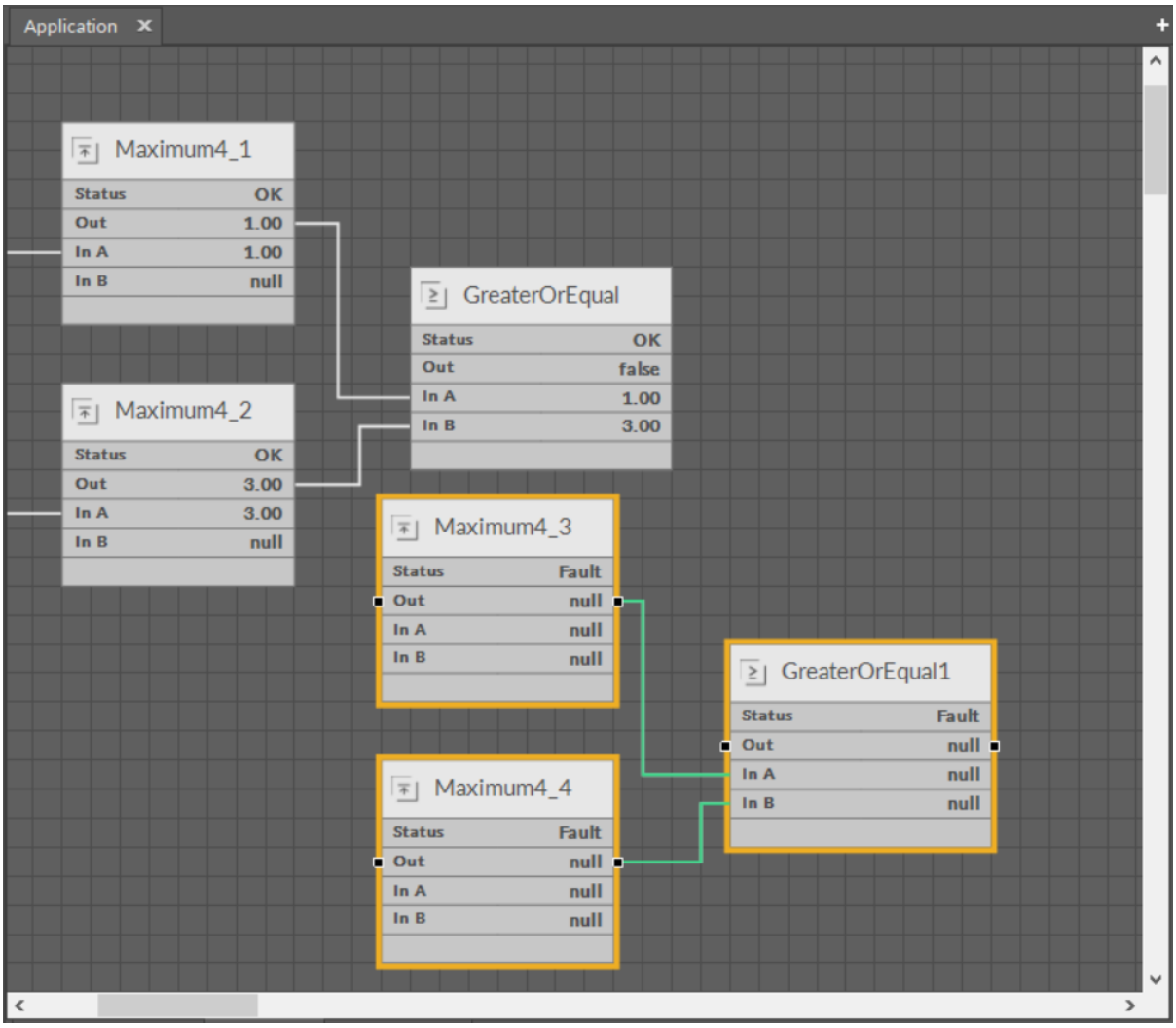



*Figure 73. Copied and pasted components*

## Special Pasting

An expanded version of the Paste function is called Paste Special (available in the context menu); its aim is to recreate external links for copied components.

External links will be recreated only if all the below requirements are met:

- using the Paste Special, not the Paste command from the context menu;
- components copied to the clipboard have external input links;
- an external component, which an external input link is recreated for, still exists.

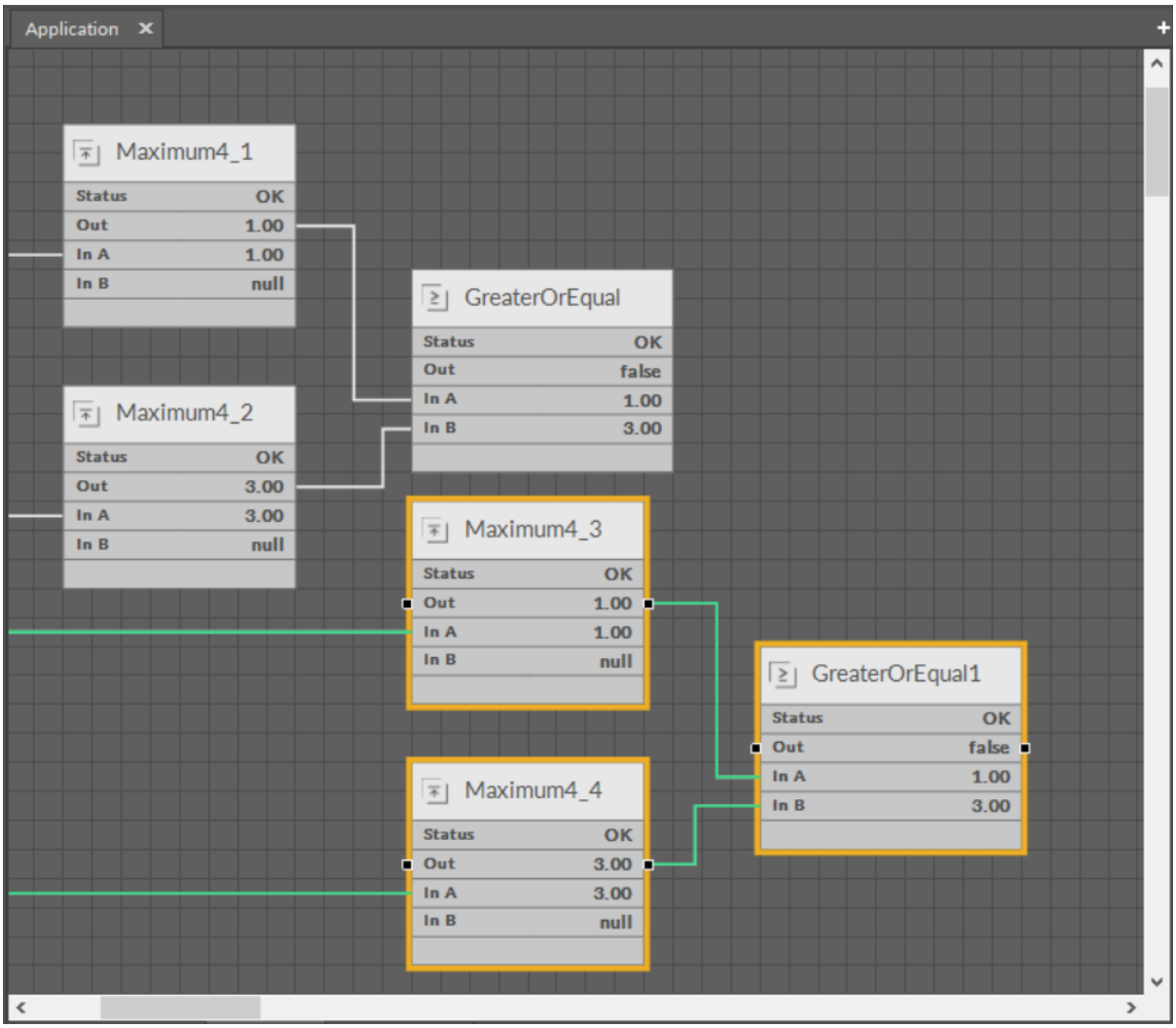



*Figure 74. Special pasted component*

## Duplication

A Duplication method is used for a quick replication of a selected component along with its values. The slot values for the duplicated component are rewritten from the source component.

Additionally, the duplication does not disturb what was copied by using Copy and Paste functions, i.e., it does not overwrite what is contained in the clipboard.

Duplication can be performed by key combination Ctrl+D or from the context menu function.

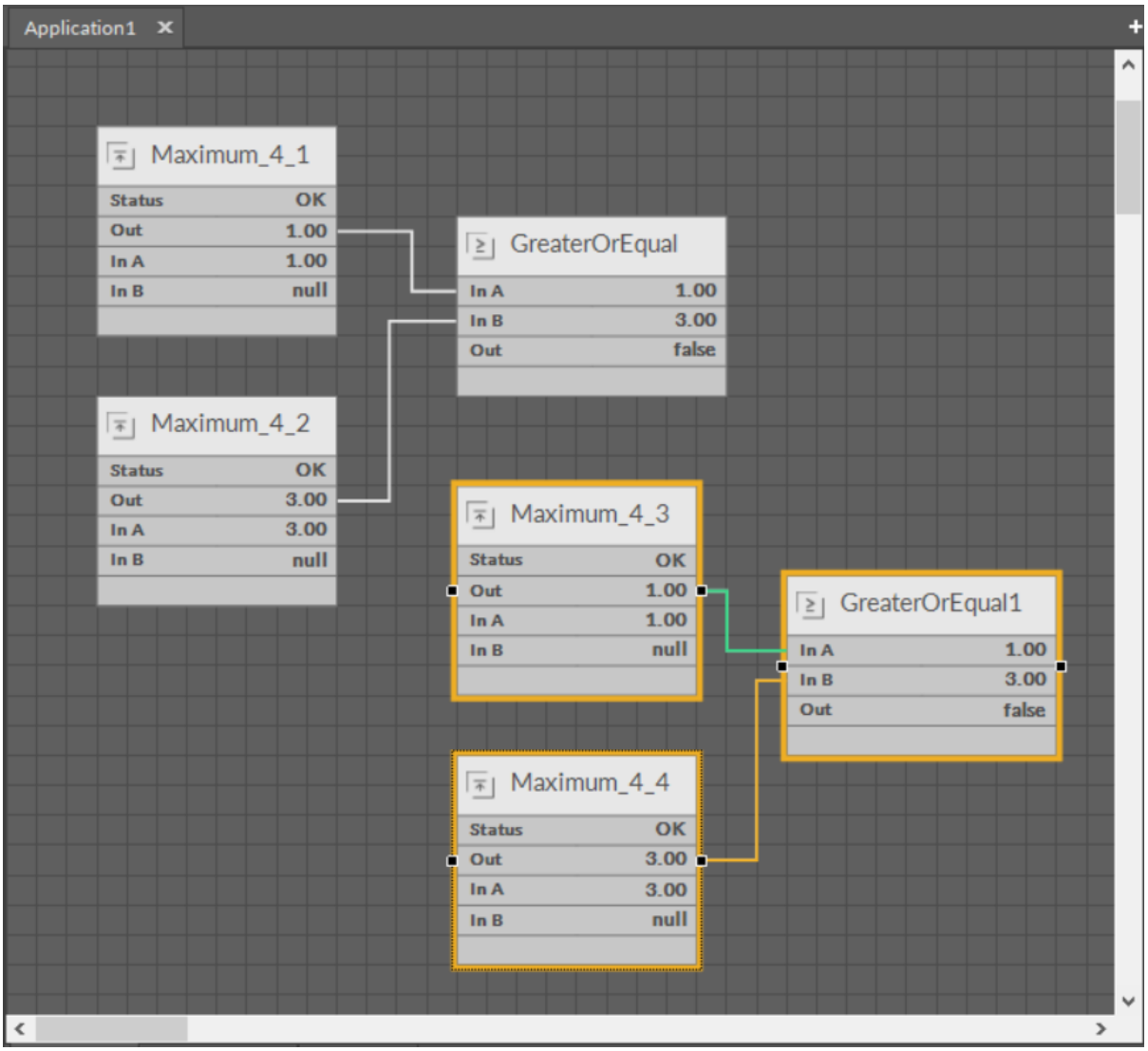



*Figure 75. Duplicated component*

## 6.2.4 Linking Components

A link is a way of exchanging data between components. It connects components between which the data are transferred. The link identifies, which parameter is taken into account on both sides of the connection. This allows to show the data flow direction, i.e., where a specific parameter is read and where it is saved.

The Wire Sheet view shows the link as a line connecting two components. The Workspace Tree window and the Property Sheet view do not show information about existing links at all.

Views designed to check links for a specific component are the Wire Sheet view and the Links tab in the Object Properties window.

Linking in the iC Tool may be performed in two ways:

- Reference linking (available only for nano EDGE ENGINE devices, described in details below) and
- standard linking method, which involves simple creating links between the input and output slots; a standard link will transfer a value between the connected slots.

**The Reference linking method is unique for the nano EDGE ENGINE devices**. Is it designed to link Data Points with network points, and it offers much more advantage than the standard linking method.

The **Reference link** is a special compound link designed to connect Data Points with network points. The Reference link is created between special Reference slots and transfers values along with the component's status. Alternatively, it may transfer values between Data Points and network points at the same time returning status from network points to Data Points, or it may return values from network points to Data Points. As network points are situated in the Networks container and Data Points are situated in the Applications container, Reference links are created using the Link Mark and Link From options from the context menu, and they are created between the tabs (or, for example, between the Application tab and the network points expanded in the Workspace Tree window). Either way the Reference link between tabs is displayed in the Wire Sheet as a bubble connected to the component's Reference slot.

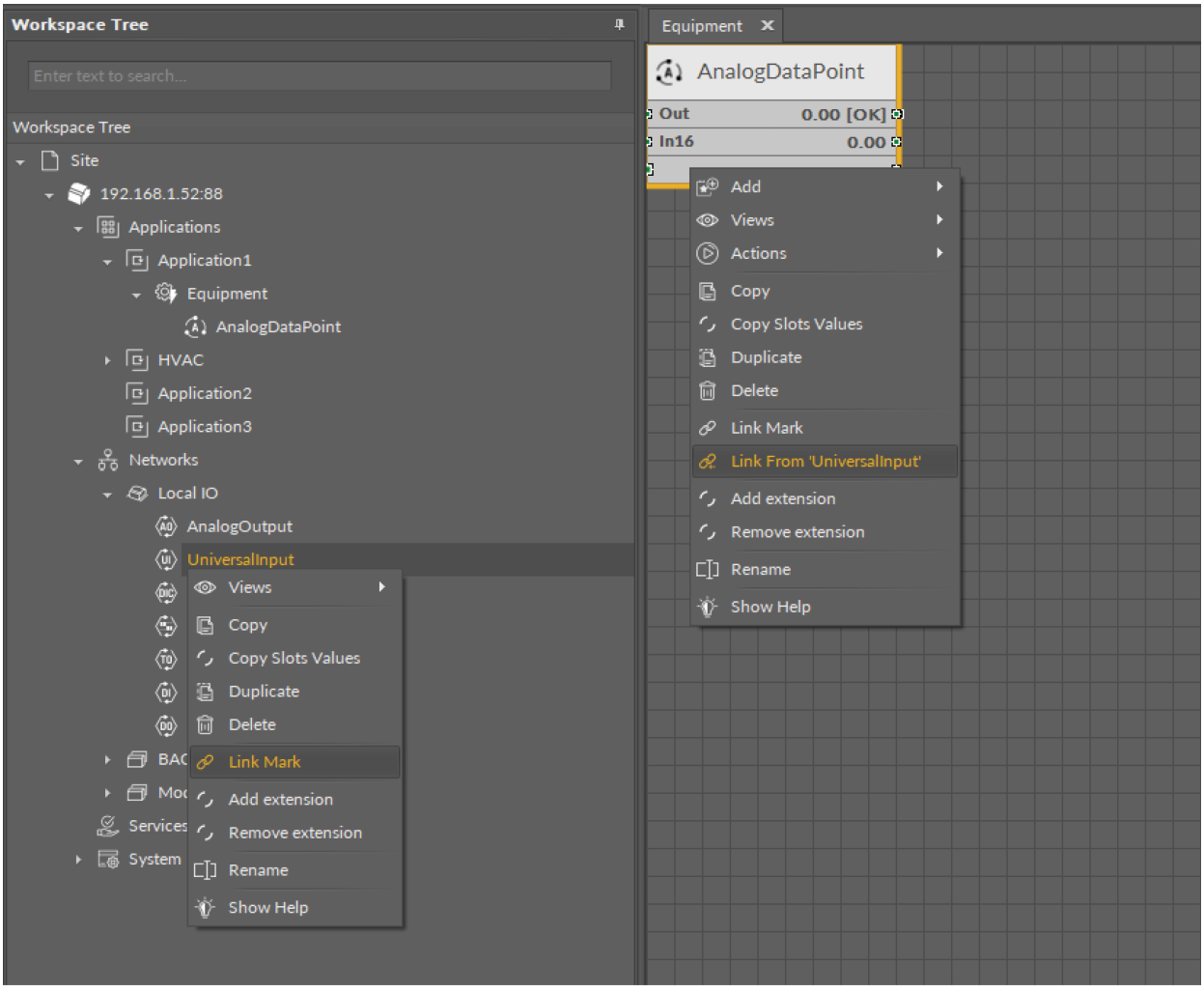



*Figure 76. Link Mark-Link From options*

The basic and exceptional feature of the Reference links is the fact that they are made to **transfer the value along with the component's status**. This feature gives a major advantage and translates to substantially enhanced functionality of linking. The fact of transferring the Status along with component's value is exceptionally important for the functionality of Data Points. Data Points are central elements in the nano EDGE ENGINE applications, and they represent values in applications on the Wire Sheet. Therefore,

displaying network point's status makes the Data Point much more informative, and allows to display this important information directly in the Wire Sheet.

## Data Point and the Input-type Network Point Links

If the Data Point is linked with the input-type network point, the Reference link transfers the network point's value and status to the Data Point (if the Data Point has a priorities array extension added, there is also the option to set the Input Priority slot in the network point, which defines the input priority in the Data Point receiving the value from the network point). In this variant, the Reference link is unidirectional, and provides the information about the change of value and the network point's status.

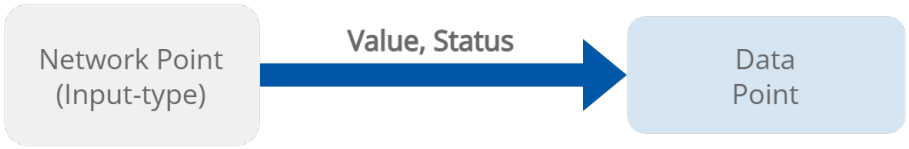



*Figure 77. Data Point and the input-type network point links*

If the Data Point is linked with the output-type network point (or network points), it offers even more advantages.

## Data Point and the Output-type Network Point Links

- First of all, in such case the Reference link behaves **bidirectionally**. It transfers the value from the Data Point to the network point, and **in turn it informs whether the value has been correctly received by the network point by sending back the network point's status**. This hugely advantageous feature allows to instantly identify that at least one of the linked network points has gone into the fault status.

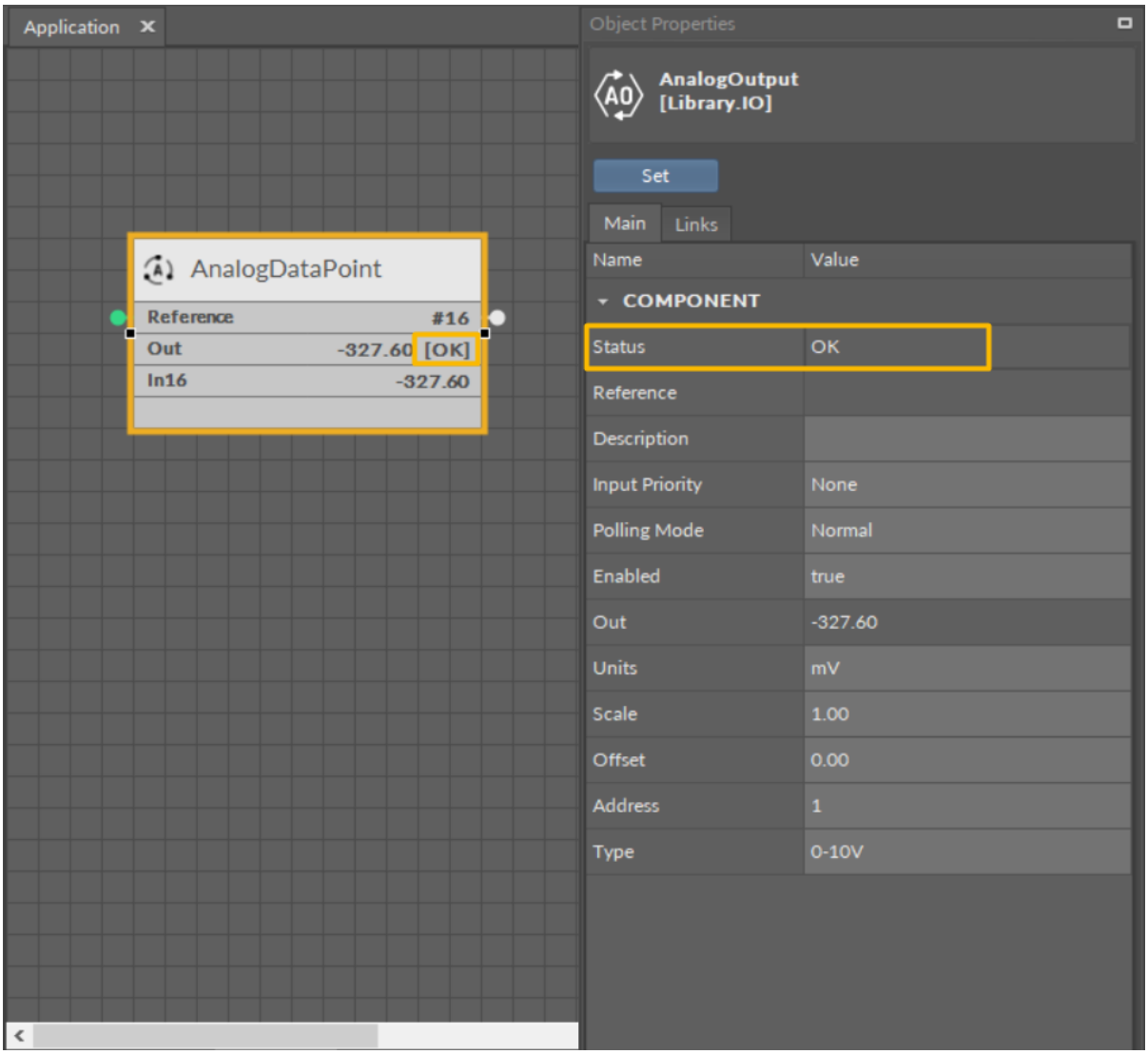



*Figure 78. Output-type network point (status OK) referenced to the Data Point*

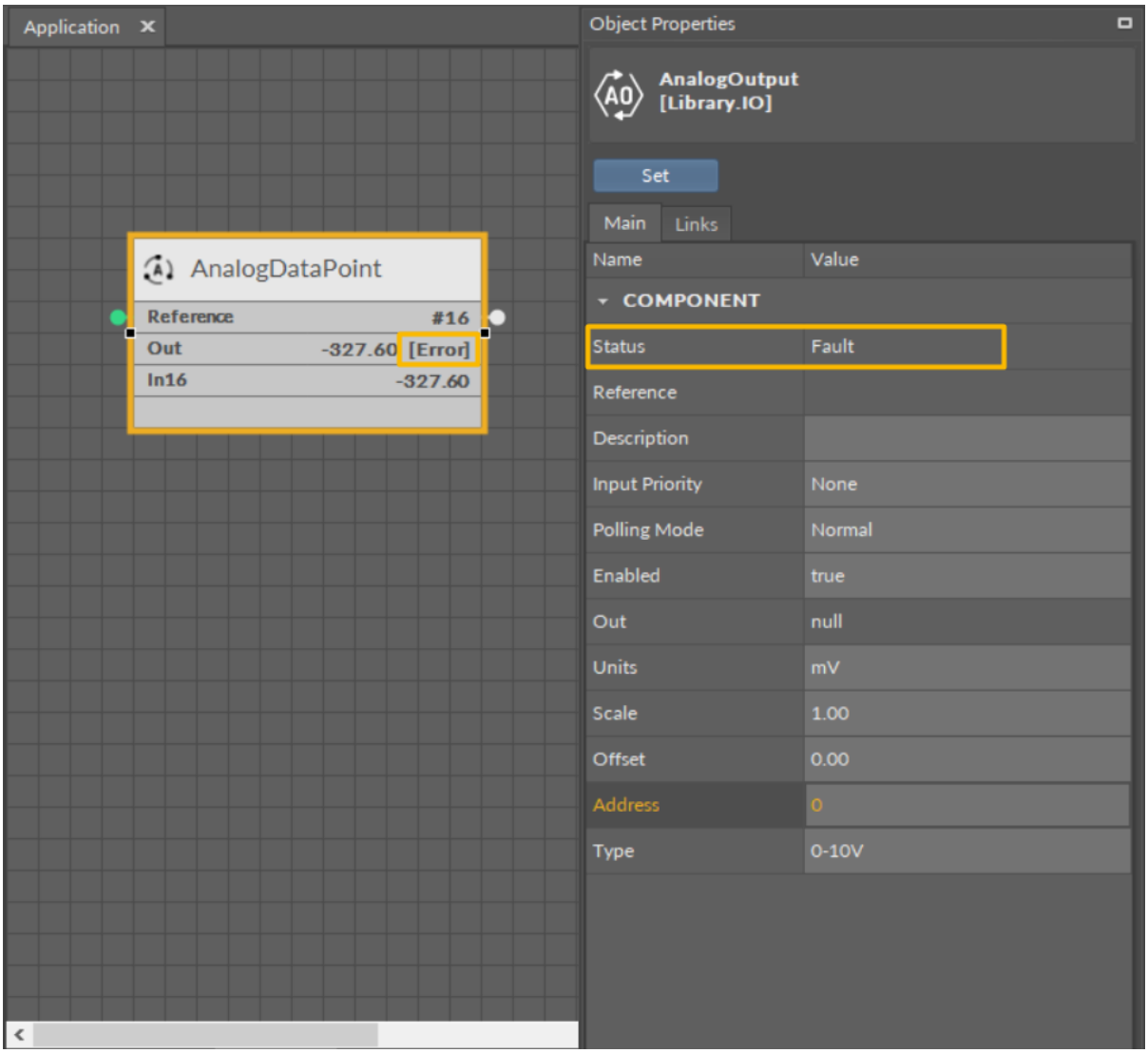



*Figure 79. Output-type network point in the Fault status returning an Error status to the referenced Data Point*

- Another innovative feature of the Reference linking is that the Reference link is able not only to return status from the network point but also to **return its value**. The network point's slot, Input Priority, is designed to identify the Data Point's priority, which the network point will transfer its value back to. For example, if the network point's Input Priority slot is set to In16, it will transfer its value back to the Data Point's 16$^{th}$ priority slot. In turn, if this value is the highest priority for the Data Point, it can distribute it to all network points linked with the Reference link. **This way, if there are more output-type network points linked with the Data Point, and one of them changes its value and sends it back to the Data Point, thanks to the bidirectional Reference link, the Data Point can synchronize values in all linked network points.**

Note: The difference between the two possible options for linking Data Points with output-type network points is derived from the network point's Input Priority slot. In the first scenario, the network point does not have the Input Priority set to any value, therefore, it cannot return a value back to the Data Point. In the second option, the network point has the Input Priority slot set and it reacts to the change of value–if the network point's value is changed, it is automatically sent back to the Data Point by the Reference link, and is updated on the defined input priority.

### Priorities Array Extension

Setting the network point's Input Priority slot is effective providing that the Data Point has the priorities array extension added. The Data Point is available in its basic version with one input slot (In16), however, it can be expanded by another 15 writable input slots with the priorities array extension (available at the right-click on the Data Point). If the Data Point has 16 writable slots, setting the Input Priority slot in the network point defines the Data Point's input receiving the value from the output-type network point over the Reference link. If the Data Point remains in its basic version, setting the Input Priority slot in the network point has no actual effect, and the value is sent to the 16th input priority in the Data Point.

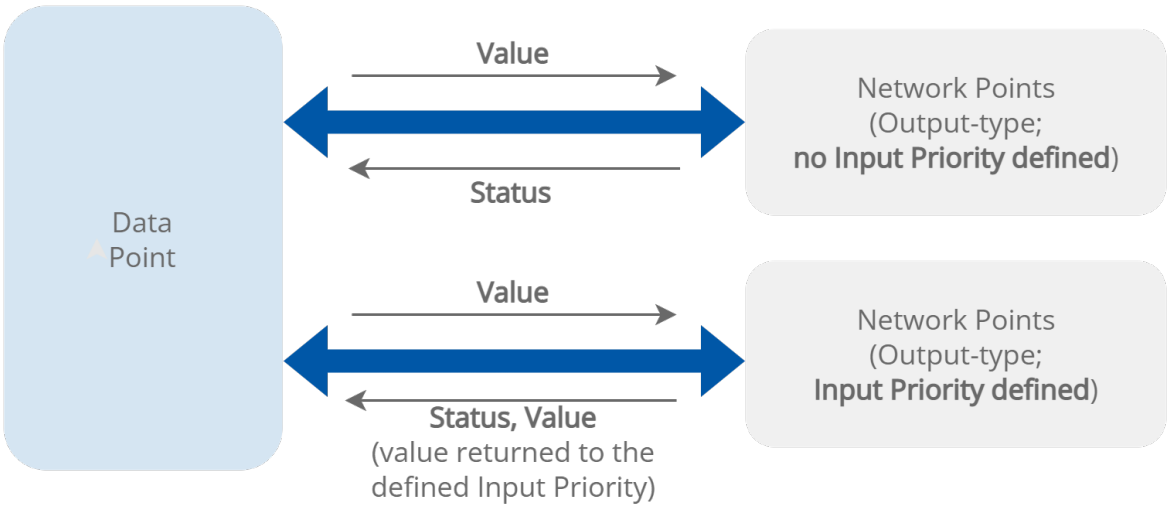



*Figure 80. The Reference links between the Data Point and output-type network points*

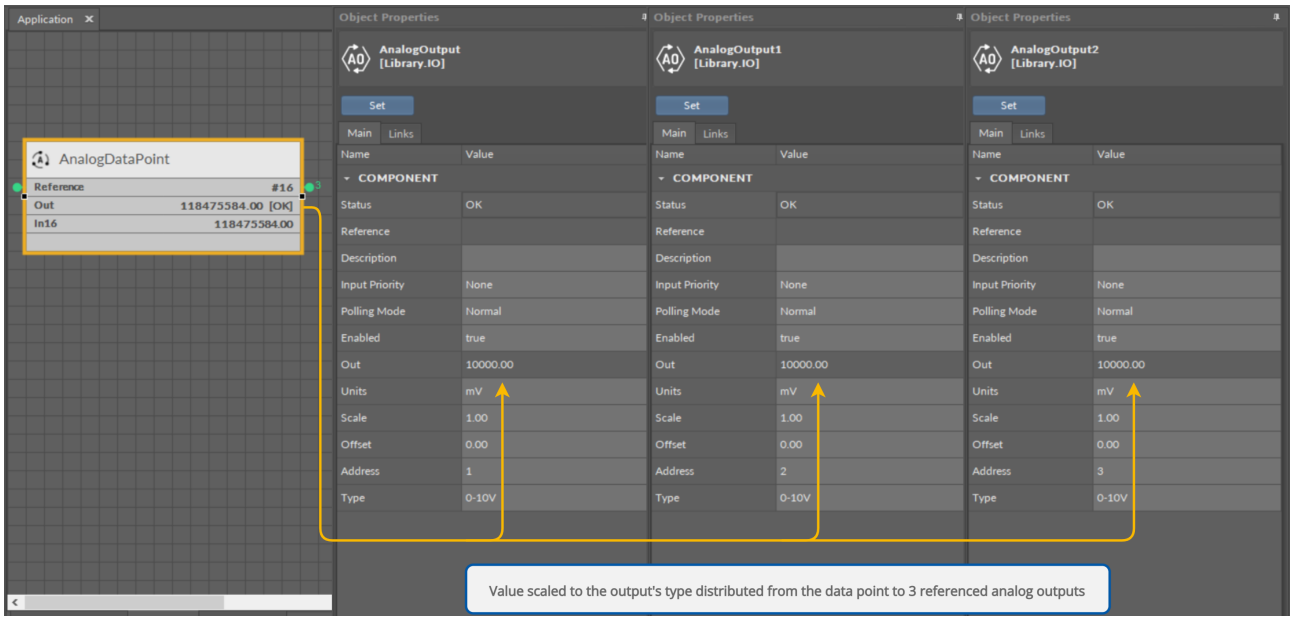



*Figure 81. Values distributed from the Data Point to the referenced output-type network points*

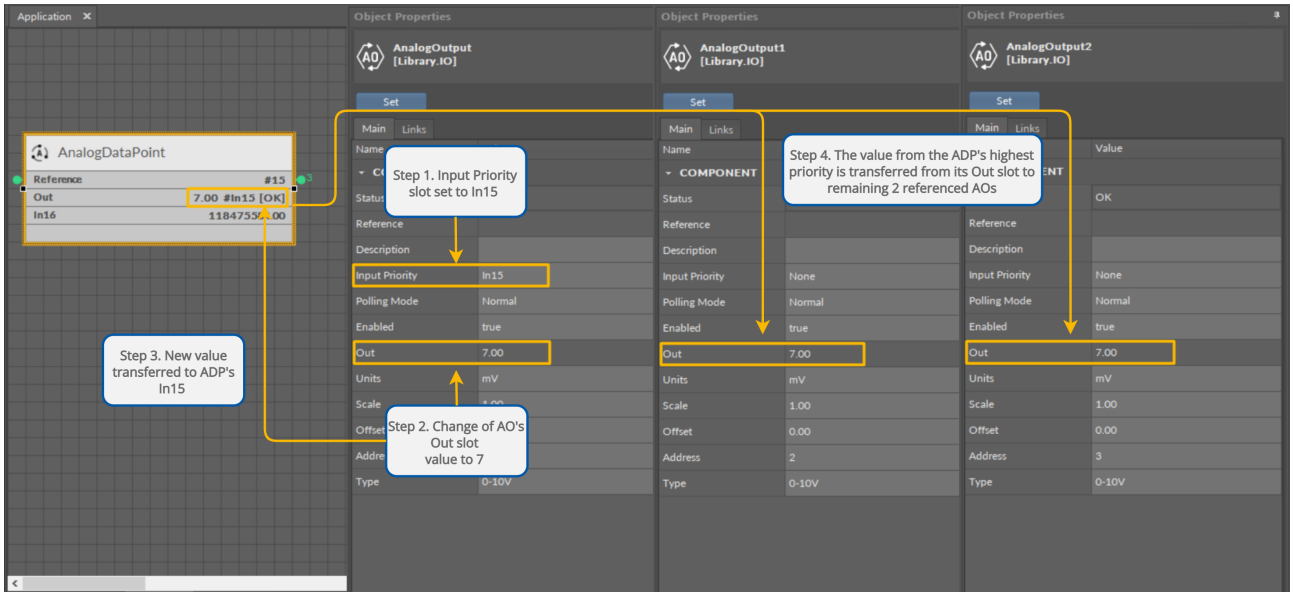



*Figure 82. Value sent back to the specified input priority in the Data Point and distributed to remaining referenced output-type network points*

## Creating Links

The iC Tool allows to link only the fitting slots, making creation of a faulty link impossible– once the source slot is selected, only the fitting target slots are highlighted.

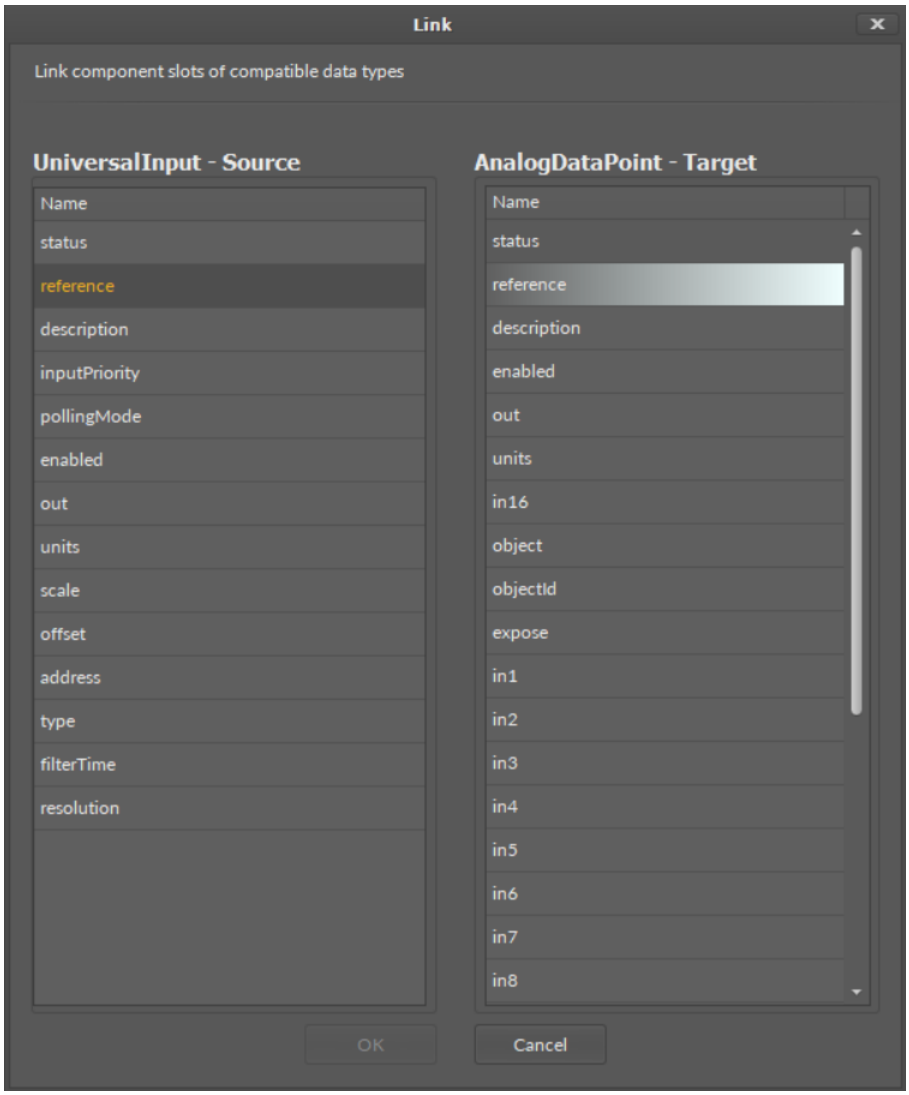



*Figure 83. The Reference linking dialog window*

Links in the iC Tool are created in two ways:

- Using the **Link Mark** and **Link From** options from the context menu: the Link Mark option selects the source component, and the Link From option selects the target component. The context menu with the Link Mark and Link From options may be opened in the Workspace Tree, Wire Sheet, and Property Sheet.
- **Graphically**: drawing a line from the source component to the target component. The link may be created graphically in the Wire Sheet. In order to create link graphically, click on the source slot, and drag the line to the target slot. The iC Tool automatically highlights the fitting target slots. As the Wire Sheet displays limited number of slots in the component, drawing a line to the empty row opens a dialog window with the list of all slots to select the target slot.

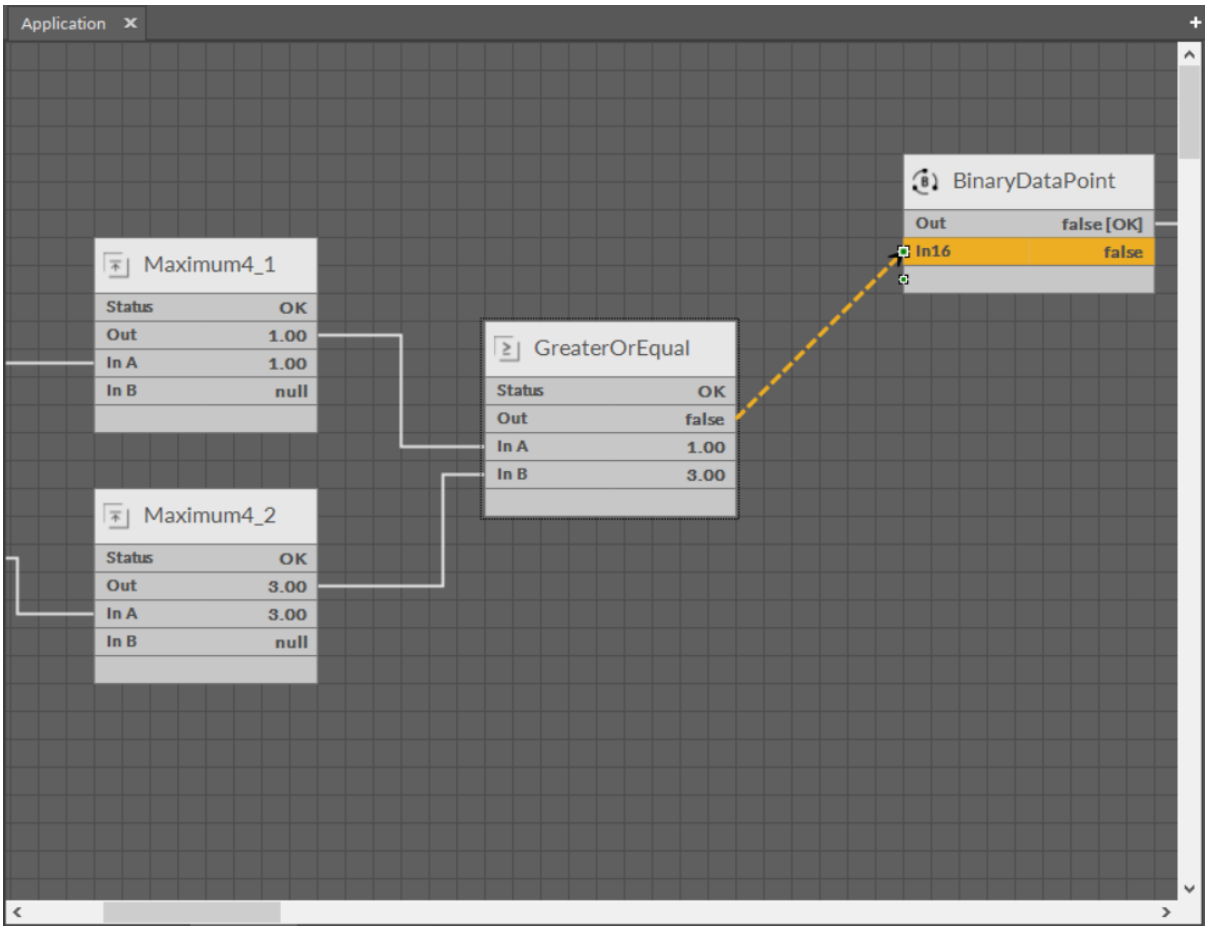



*Figure 84. Creating links graphically*

## 6.2.5 Removing Components and Links

### Removing Components

Deleting components allows removal of the selected components along with all their inferior components. Deleted components are instantly removed from the application in the device, which provides full synchronization in views and windows: the Property Sheet, Wire Sheet, and Workspace Tree.

Removal is possible after selecting one or more components and pressing a Delete key on the keyboard, with the exception of removing components from the Workspace Tree.

Components' removal from the Workspace Tree with the Delete key has been blocked to prevent removing the main tree elements, such as the main containers or the whole application by mistake. Removing components in the Workspace Tree is possible using the context menu but the removal requires confirming it in a pop-up window.

The above limitations do not include the Property Sheet and Wire Sheet views, where using the Delete key removes selected components instantly.

### Removing Links

Removing links is performed automatically after removing the component, which a specific link leads to. It regards both input and output links.There is no need to remove

links if a component is removed. However, there are situations when only the link needs to be deleted, without removing the component.

In order to remove the link only, it needs to be selected in the Wire Sheet view and then deleted by the Delete key on the keyboard or from the context menu by choosing a Delete link(s) option.

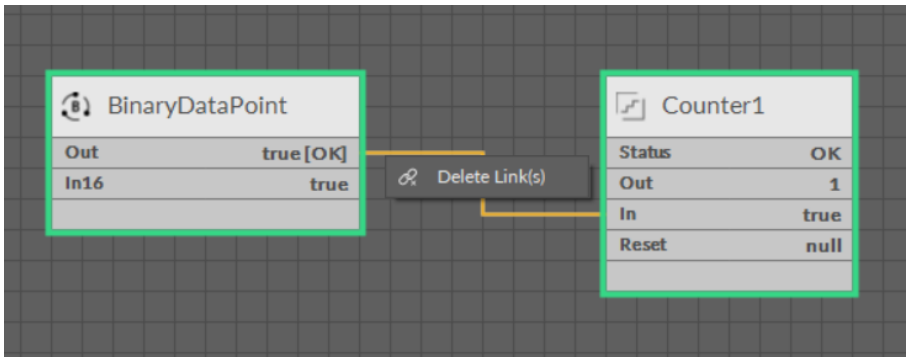



*Figure 85. The context menu of a link*

## 6.3 Actions

Actions are the most commonly used functions performed on a specific component, such as setting a new value or resetting a counter.

Available actions depend on the component type. It is possible to access them as follows:

- Pressing the Actions option on the context menu of a component;

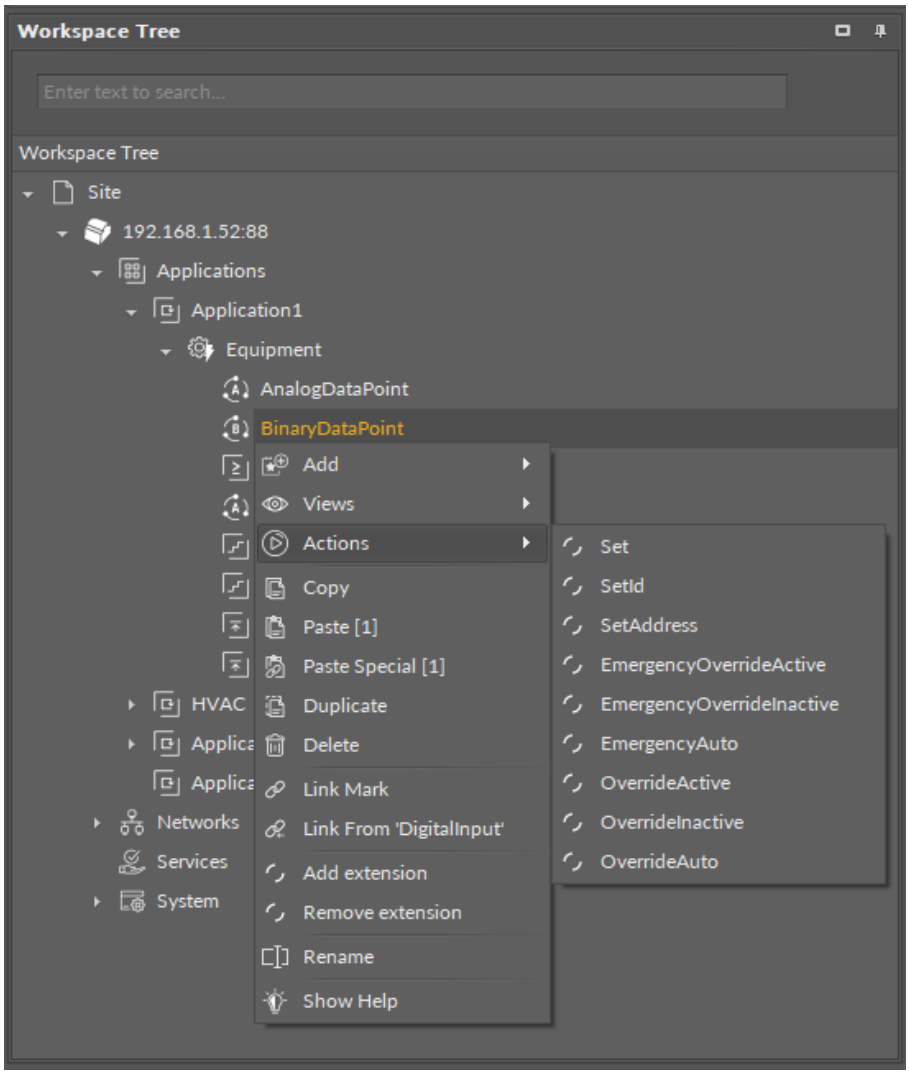



*Figure 86. Accessing actions from the context menu*

- Pressing the actions button available in the Object Properties window after selecting the required component.

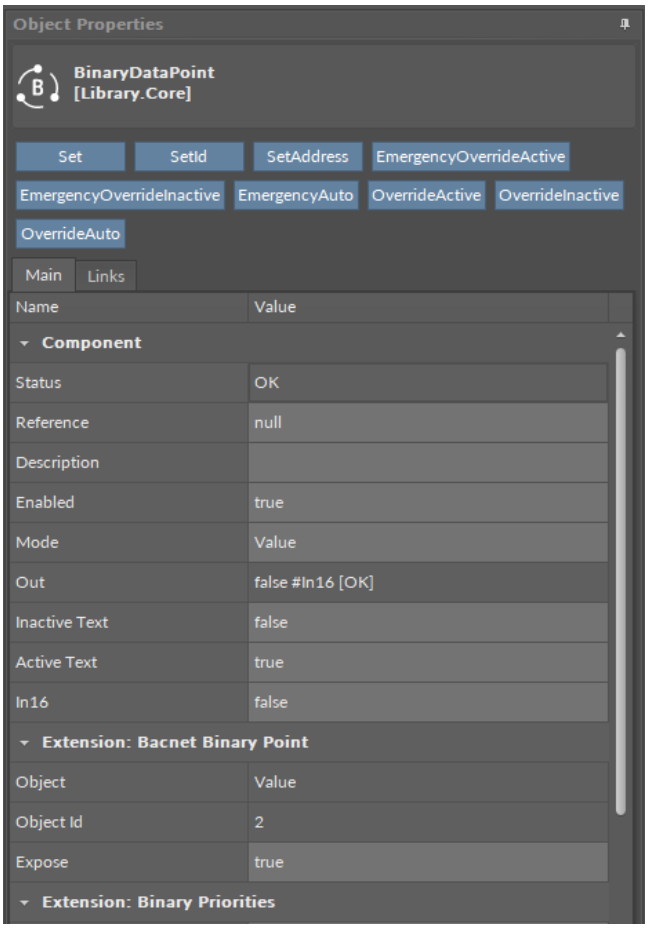



*Figure 87. Accessing actions in the Object Properties window*

If the action requires entering parameters in the component, for example, setting values, the iC Tool will ask for values on a separate pop-up window.

## 6.4 Renaming Components

A Rename option selected on the component's context menu allows changing its name. The Rename pattern defines how the new name for components will be generated. It can include some constant literals with special placeholders embedded.

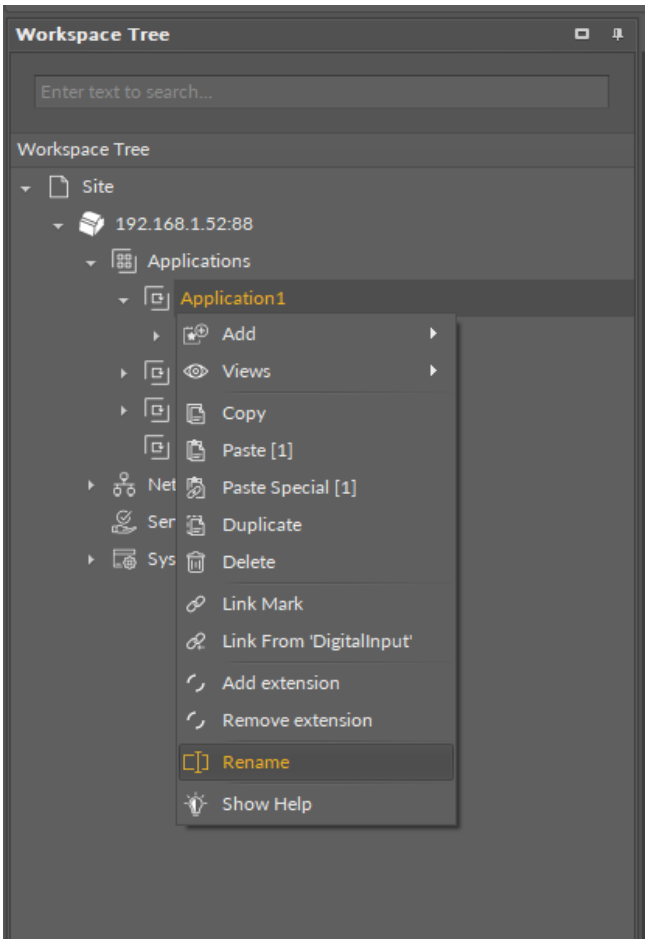



*Figure 88. Renaming components*

## 6.5 Reordering Components (Only for Sedona-based Devices)

While working in the iC Tool, the order of components under a parent component is automatically set based on the ID number. The user sees components in the order they were added to the application and there is no need to change the order to ensure proper work of the controller.

Building a text interface for the LCD display controller is an exception. In this case the order of components allowing to show text information on the LCD display is important. The order of components is the basis of the look (order) of information shown on LCD display.

In the iC Tool it is possible to define the order of components within a parent the following way:

In the Workspace Tree or Property Sheet view, or the Driver view (each view where the components are shown) select a component (parent) with children (other components), which will have their order changed. On the figure below the component (parent) is Local_DI.

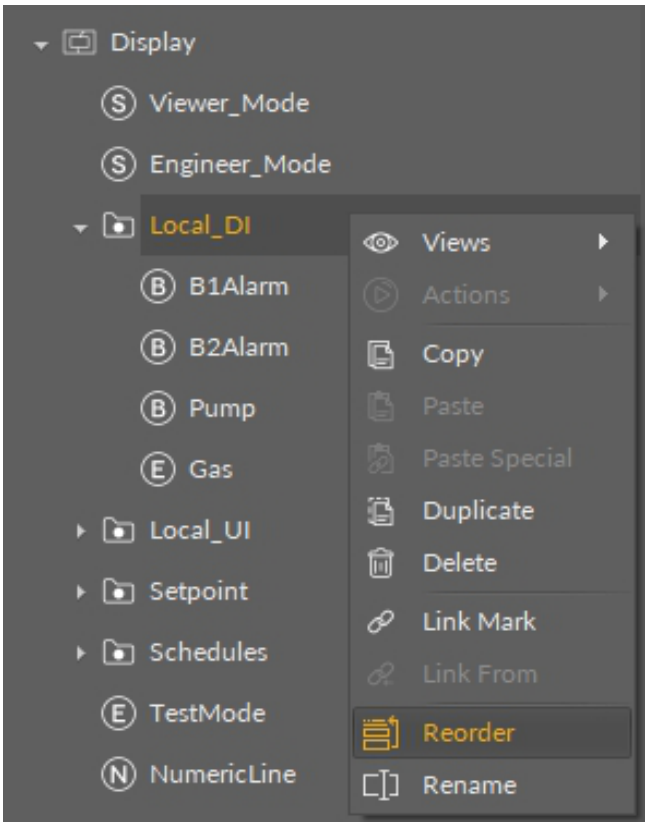



*Figure 89. Reordering components from a context menu*

Choose a Reorder option from the context menu, which results in opening a pop-up window allowing the order to be changed. On the figure below the Pump component is being reordered. It is moved by drag&drop to the beginning of the list.


*Table 2. Reordering components effective on the AAC20 LCD display*

**Warning!**

In case of using the Reorder function on a childless component, the same Reorder function will be used for the parent of the selected component. This allows using the Reorder function by selecting one of the components which will be reordered (provided it has no children).

When the components order changes are done, they need to be confirmed in a pop-up window by clicking an OK button. New order of the components is instantly sent to the controller, which is also illustrated with the order change in the Workspace Tree and other views showing components.

# 7 Standard Views

The iC Tool defines a few standard views available after choosing the component. For devices driven by the nano EDGE ENGINE, they are: the Wire Sheet and Property Sheet. These views are displayed in the iC Tool's Main Screen on separate tabs. Apart from the standard views mentioned above, available for any component, there can also be other, dedicated views for particular components (see Special Views).

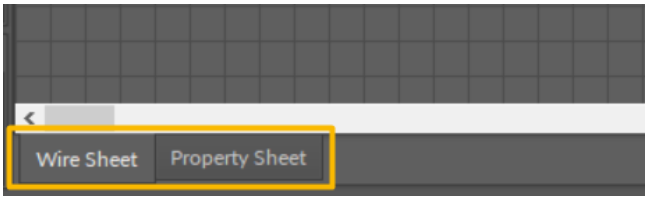



*Figure 90. The component's standard views*

Standard views for components have different roles and functions, however, they share common actions, which facilitate working with components.

## 7.1 View History

While working on application, the user switches between views to present objects being worked on in a most convenient way. Double clicking on a component in the Workspace Tree opens a corresponding default view. As a convenience matter, the iC Tool offers a view navigation history to allow the user to return to previously opened tabs.

The view navigation history is available on a toolbar under the main menu. There are three navigation history buttons in the toolbar: Tab backward, Tab forward, and Tab history.

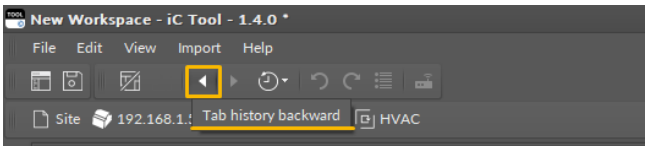



*Figure 91. Tab backward*

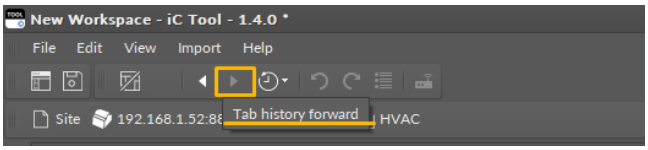



*Figure 92. Tab forward*

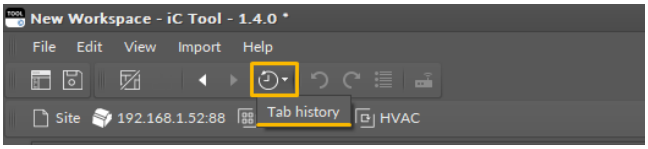



*Figure 93. Tab history*

These actions can also be invoked using keyboard shortcuts: Alt+Left, Alt+Right, or Alt+Shift+Left/Right key combinations.

The view history is presented depending on how it was invoked. Clicking a toolbar button presents a dropdown list:

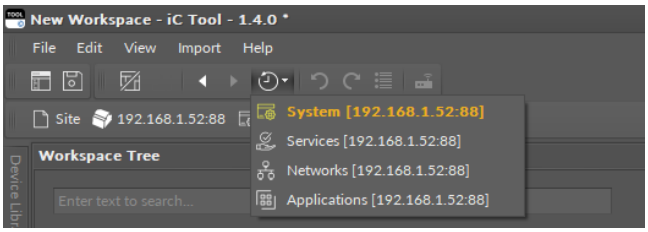



*Figure 94. Tabs history*

Pressing an Alt+Shift+Left/Right keyboard shortcut presents a flyout panel:


*Figure 95. Tabs history flyout panel*

## 7.2 Undo/Redo Actions

In case of an unintended action (for example, a component deleted by mistake, link added by mistake, randomly moved components), the iC Tool offers undo/redo actions.

With standard key combinations of Ctrl+Z and Ctrl+Y, the iC Tool allows to undo/redo actions step-by-step.

Pressing either key combination brings an undo/redo history flyout panel to help identify what will be undone/redone next. The panel stays in place while the user keeps holding the Ctrl button down so it is visible what is available to be undone/redone next if multiple changes are needed.

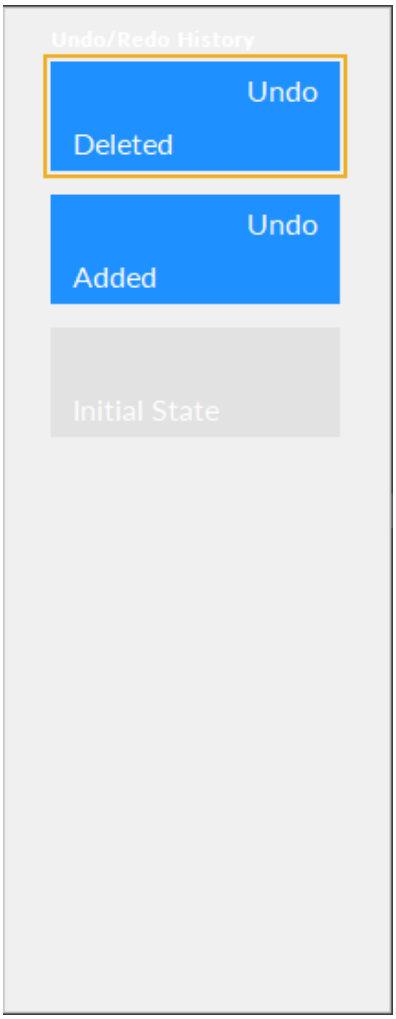



*Figure 96. Undo/Redo action flyout panel*

It is also possible to invoke undo/redo actions or open the flyout panel clicking corresponding buttons on the toolbar:

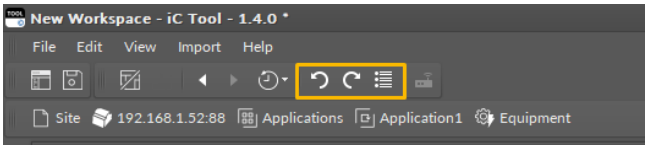



*Figure 97. Undo/redo buttons*

**WARNING!**

As many operations on components involve immediate changes to the device over the wire, and the device itself could be modified by another user of iC Tool software remotely, it is possible that some actions will fail to complete. To minimize risks of loses to the application the user is creating, the undo/redo history is cleared in case of unrecoverable communication issues.

## 7.3 Reconnecting Devices

When working with multiple devices, it is often a case that multiple tabs are opened but devices (that the tabs refer to) are disconnected. It is possible to reconnect them all at once by clicking a single button on the toolbar.

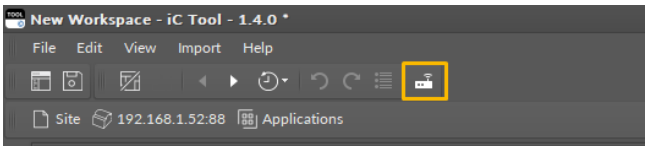



*Figure 98. The Reconnect button*

Once the action is invoked, all devices with at least one tab open are reconnected. To avoid swamping network communication, actual connection commands are sent once the user switches to a corresponding tab.

## 7.4 Wire Sheet

The Wire Sheet is a graphic view which shows all the inferior components (along with links) for the selected superior component (double-clicked in the Workspace Tree). Each component in the Wire Sheet view is shown as a rectangle with its name and icon placed in a header, and slot names and values in rows underneath. The link is displayed as a line connecting two component slots, between which values are transferred. The figure below shows the Wire Sheet view with corresponding list of components in the Workspace Tree for the Application service.

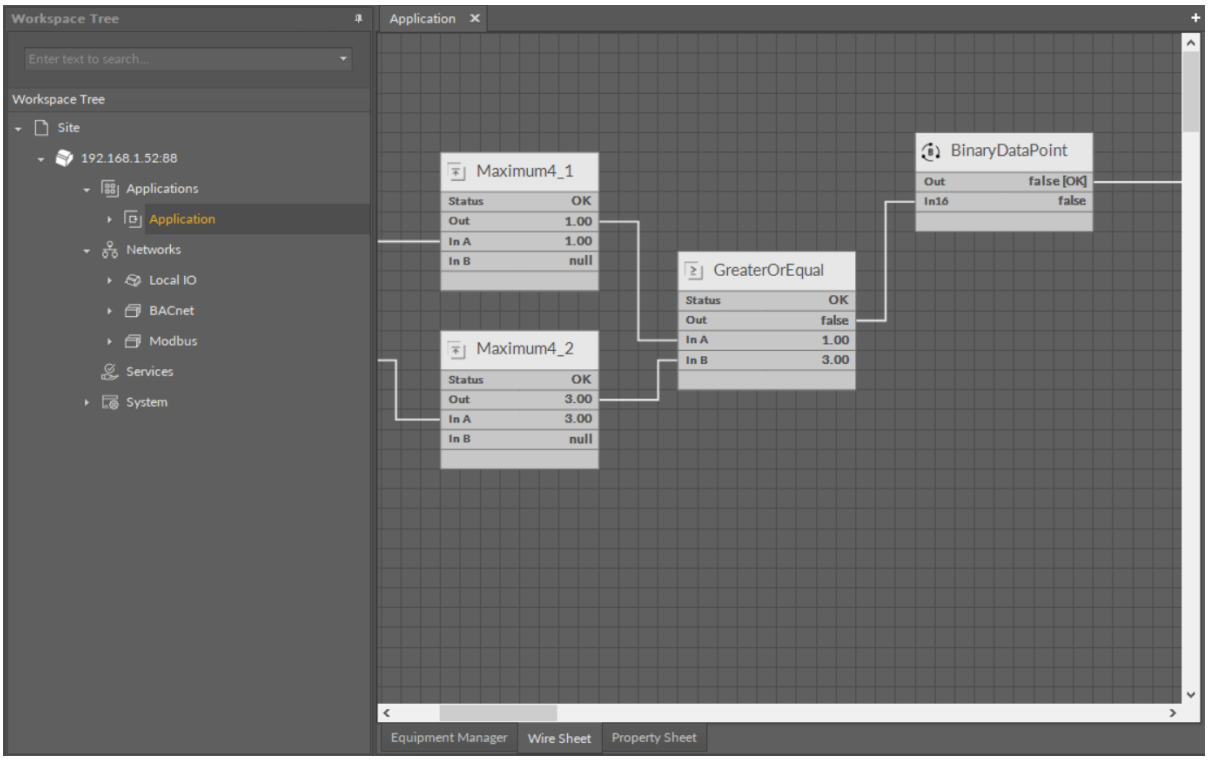



*Figure 99. The Wire Sheet view along with the Workspace Tree for the nano EDGE ENGINE device*

In the Wire Sheet view the user can define the location of components by dragging them by the header. To make moving components easier the Wire Sheet displays a grid allowing precise placement of the component. The grid makes organizing of components and their alignment easier.

### 7.4.1 Components and Links Characteristics

Every component in the Wire Sheet view has a following look and behavior:

- Components have headers with icons and components' names.
- Under the component's header, every row represents a single slot with a name, value, and unit.

- Slots hidden by default are not displayed.
- Component may be extended left and right between 5 and 31 grid sections. This allows the component to be adjusted optimally, so all the slots names and their values can be fully visible.
- Component can be moved on the Wire Sheet into any place by clicking and holding a component's header or a middle part of any slot. Releasing the component defines its new location, after which links between this and other components are redrawn.
- Moving of component can be cancelled by pressing an Esc button on the keyboard (before releasing it in the new location).
- Component can also be moved by pressing an Alt key and clicking any part of component. This allows to grab and drag the component also behind the place, where link assigning begins.
- Selected (clicked) component is marked with an orange frame, and is treated by the iC Tool as an object chosen by the user, contrary to the component surrounded by a green frame, which is treated by the iC Tool as a reference (auxiliary) object, not selected.
- A context menu available at right-click provides functions, which can be performed on a component regardless of the view it is displayed in.

Every link in the Wire Sheet view has a following look and behavior:

- Links between components reflect the connections between specific slots of connected components.
- Link guarantees sending the value of source component slot to a target component slot.
- Only one input link can be defined for an input (left) slot.
- Many output links may come out of one output (right) slot.
- Links connecting components from different Wire Sheets are graphically represented as short segments coming in/out of slots ended with a full circle.
- A context menu available under right-click allows to remove a selected link.
- The user has no influence on a link route, i.e. the way it will be drawn, since the process is done automatically.

### 7.4.2 Navigation

Navigation on the Wire Sheet view is carried out by zooming in/out and moving the object in the view.

Zooming the view in or out requires turning a mouse wheel upward or downward. Zooming is centered where the mouse pointer is. In order to zoom in or out a particular Wire Sheet fragment, the mouse pointer needs to be moved into this area, so it does not escape the field of view.

In order to scroll the Wire Sheet view, move the sliders at the bottom and right side of the window or press the Ctrl button and click and move up or down.

After opening the Wire Sheet view, the iC Tool automatically adjusts the zoom allowing all components to be visible.

## 7.4.3 Selecting and Marking of Components and Links

The Wire Sheet view has been designed to make it easy for the user to analyze the logic of the program by proper marking of the inferior components and links in relation to a selected component or link. Both components and links are selected by clicking them, and pressing the Ctrl key while clicking allows to select more than one component.

### Components

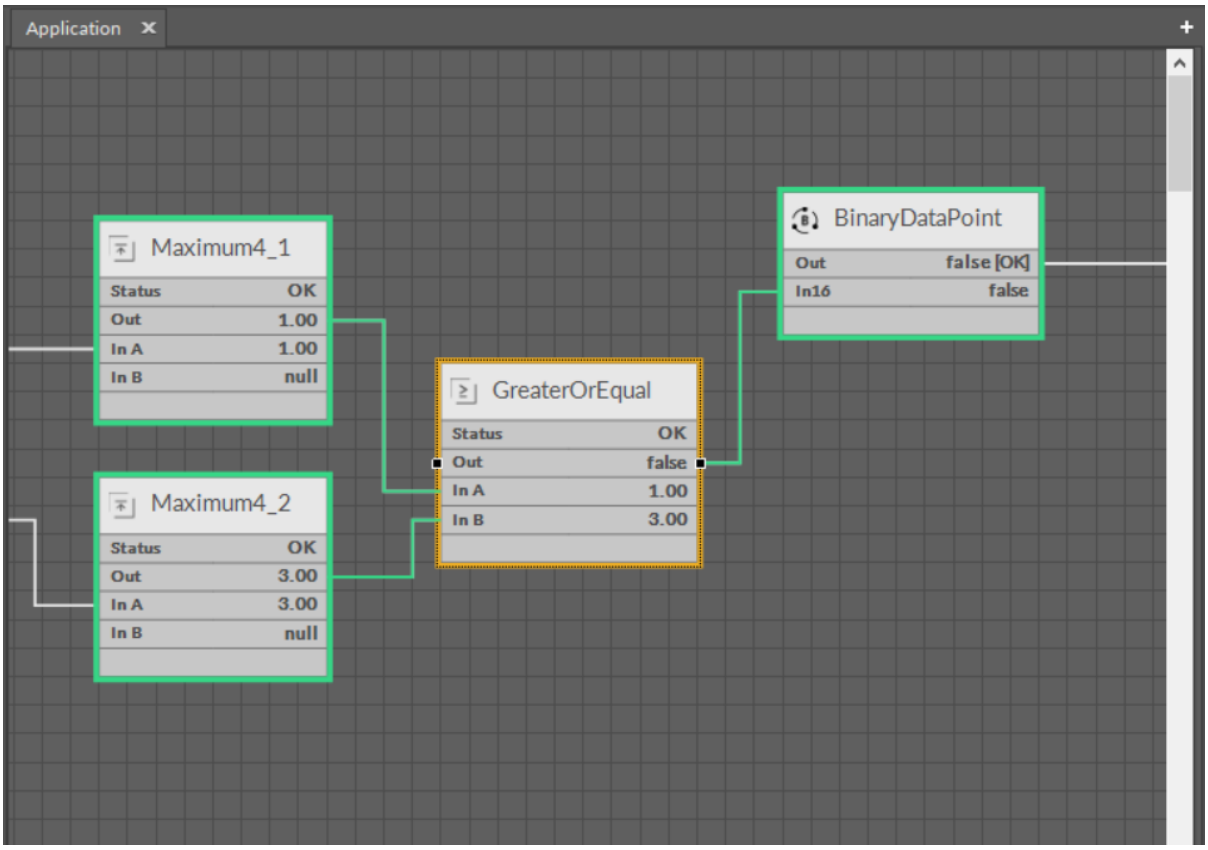



*Figure 100. Selecting (orange) and marking (green) of components*

In the figure above, a GreaterOrEqual component has been selected and is marked with an orange frame. If it will be removed or moved, or duplicated, it has an impact on the links connected to it and their components. The whole path of connected objects has been highlighted in green. A coloring mechanism shown above may be very useful to analyze complex programs with many connections, links, and components.

### Links

Similar color highlighting takes place if a link is selected (marked in orange). In such case the iC Tool will automatically mark with a green color all components connected with this link. It allows the user to be aware, which components is the link connected to, and which will be affected in case it is, for example, deleted.

The link gets marked if a connected component gets selected.

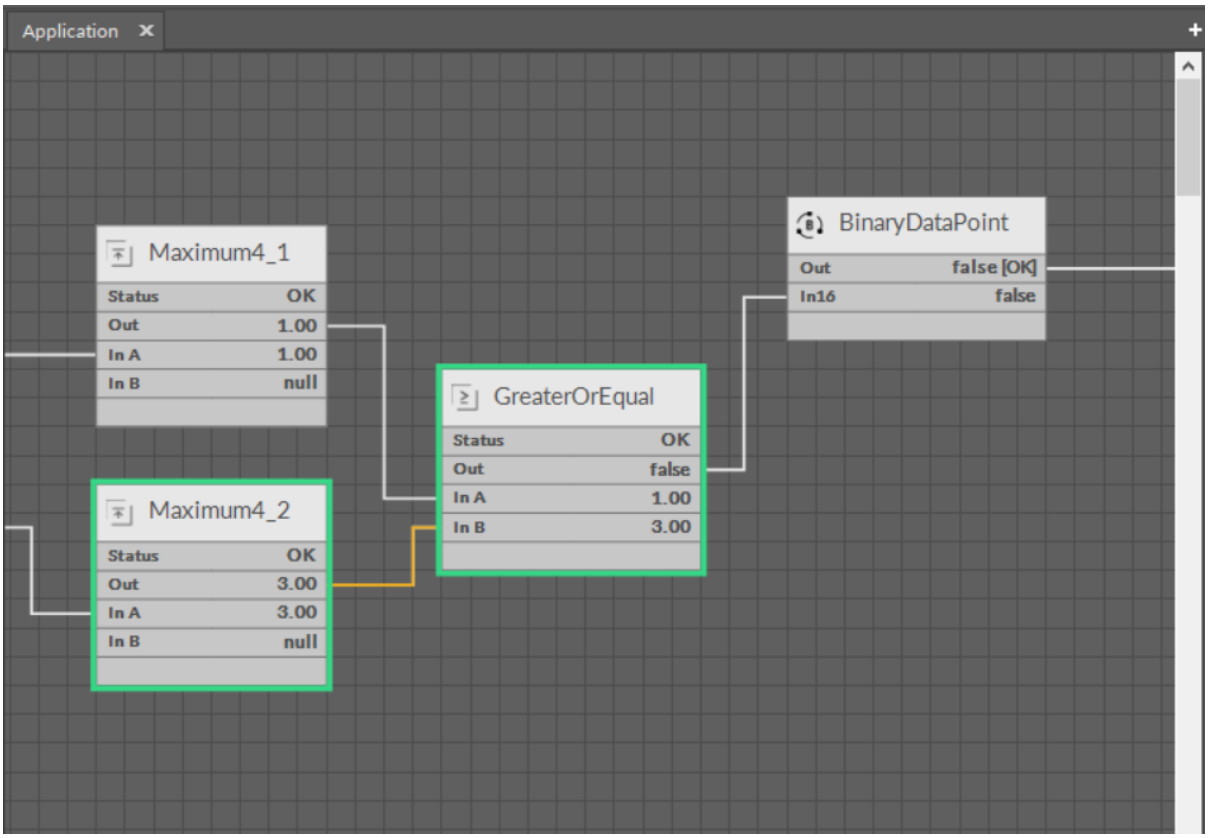



*Figure 101. Selected link*

## Window and Crossing Selection

In the Wire Sheet view there is also another method of selecting components and links. The method involves defining a rectangular area, which marks objects to be selected. It can either be a window selection or a crossing selection.

Both types require marking a square area in the Wire Sheet view by clicking in an empty area and dragging a rectangle over the objects to be selected. Release the rectangle, when all the selected objects are included.

### Window Selection

In case of a window selection, a rectangle dragged **from the left to the right** is blue. Only objects fully covered by the rectangle are selected (orange color); here, these are the Maximum and GreaterOrEqual components and their links. The remaining element has not been selected because it was not fully covered by the blue rectangle. The element dependent on the selected component and selected links is marked in green.

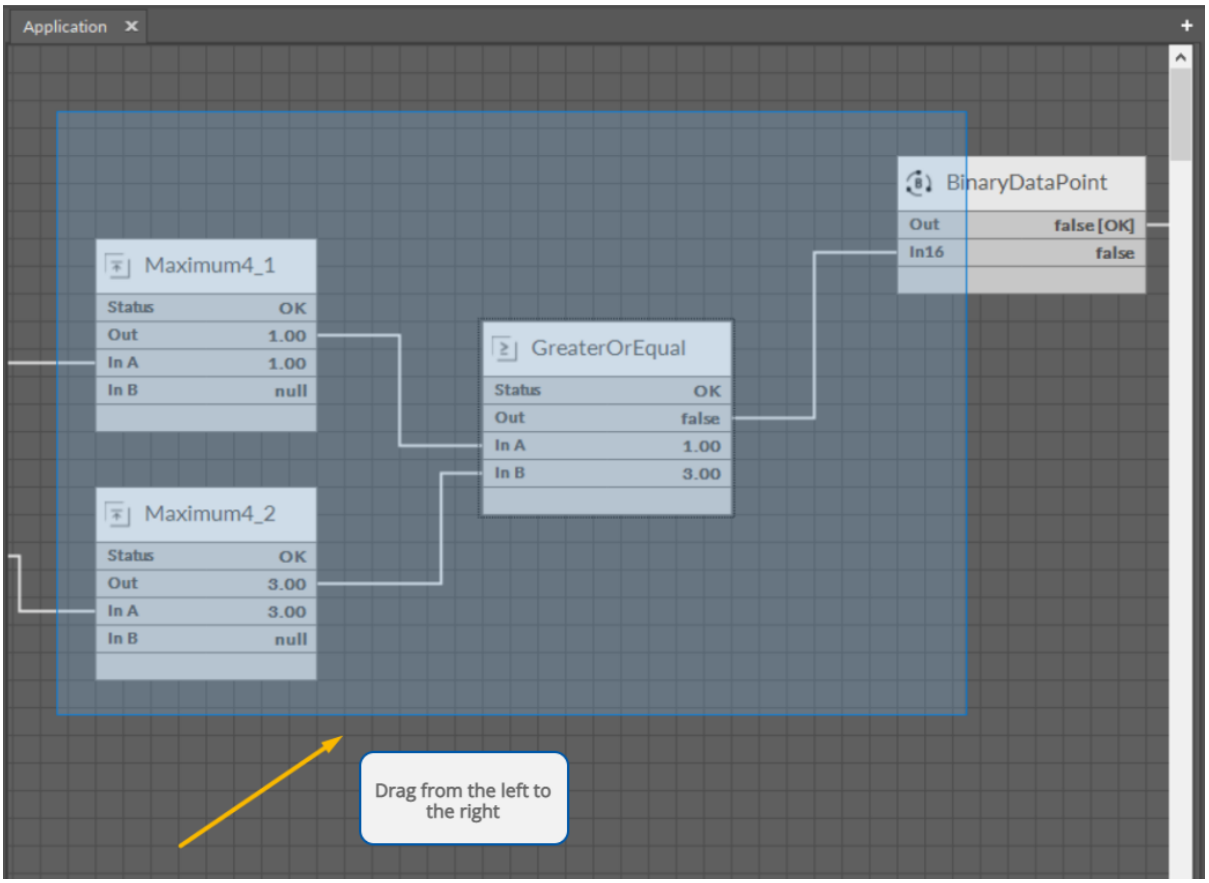



*Figure 102. Window selection*

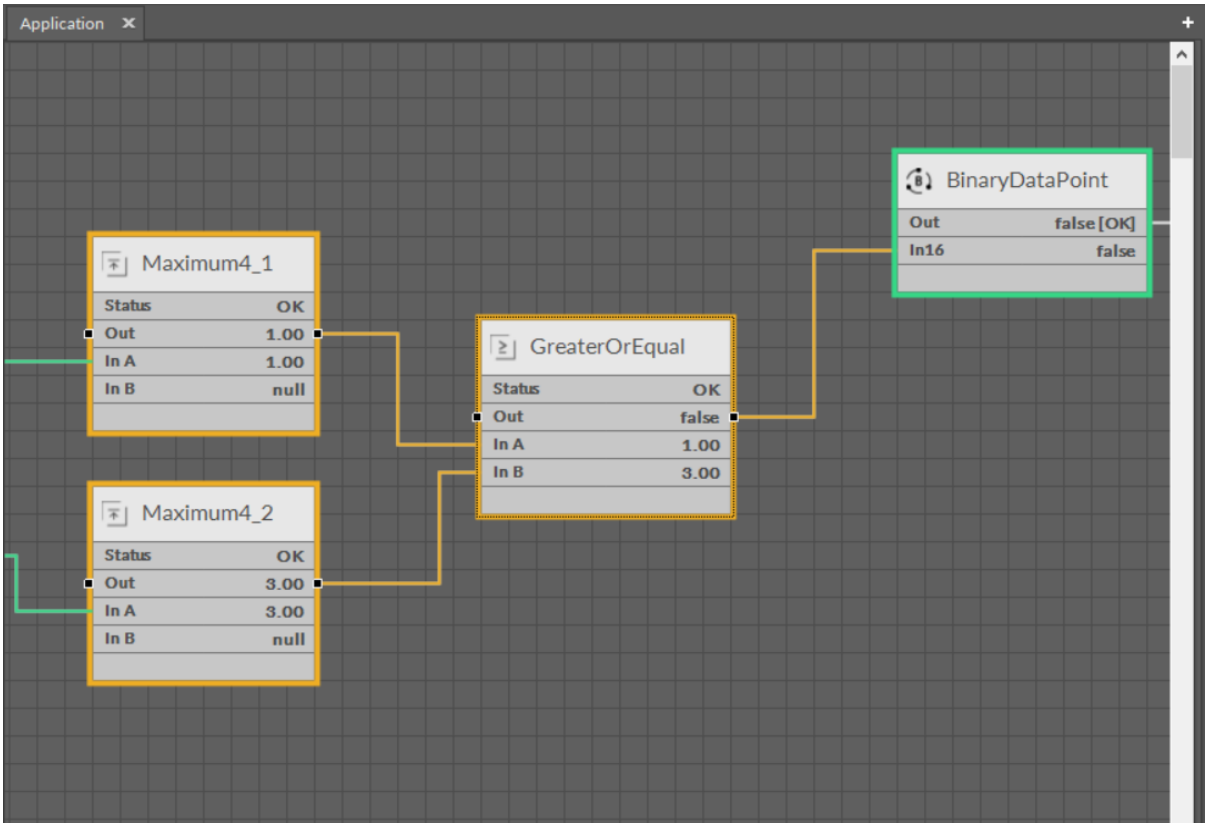



*Figure 103. Window selection*

### Crossing Selection

In case of a crossing selection the rectangle is dragged the opposite way, **from the right to the left**, and it is red. All the objects, even partially included in the rectangle, are selected (orange color); here, these are the Maxiumum, GreaterOrEqual, and Binary Data Point component along with their links.

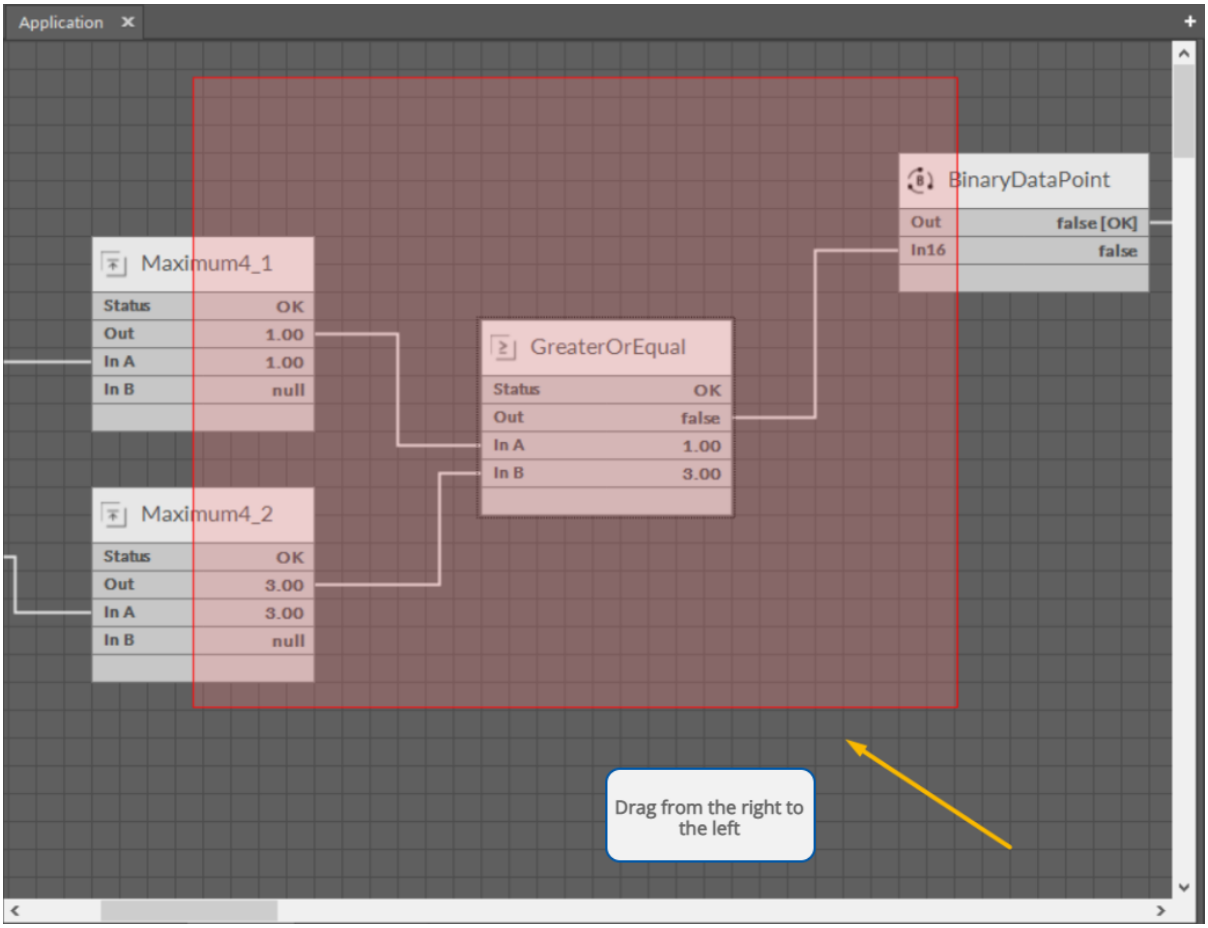



*Figure 104. Crossing selection*

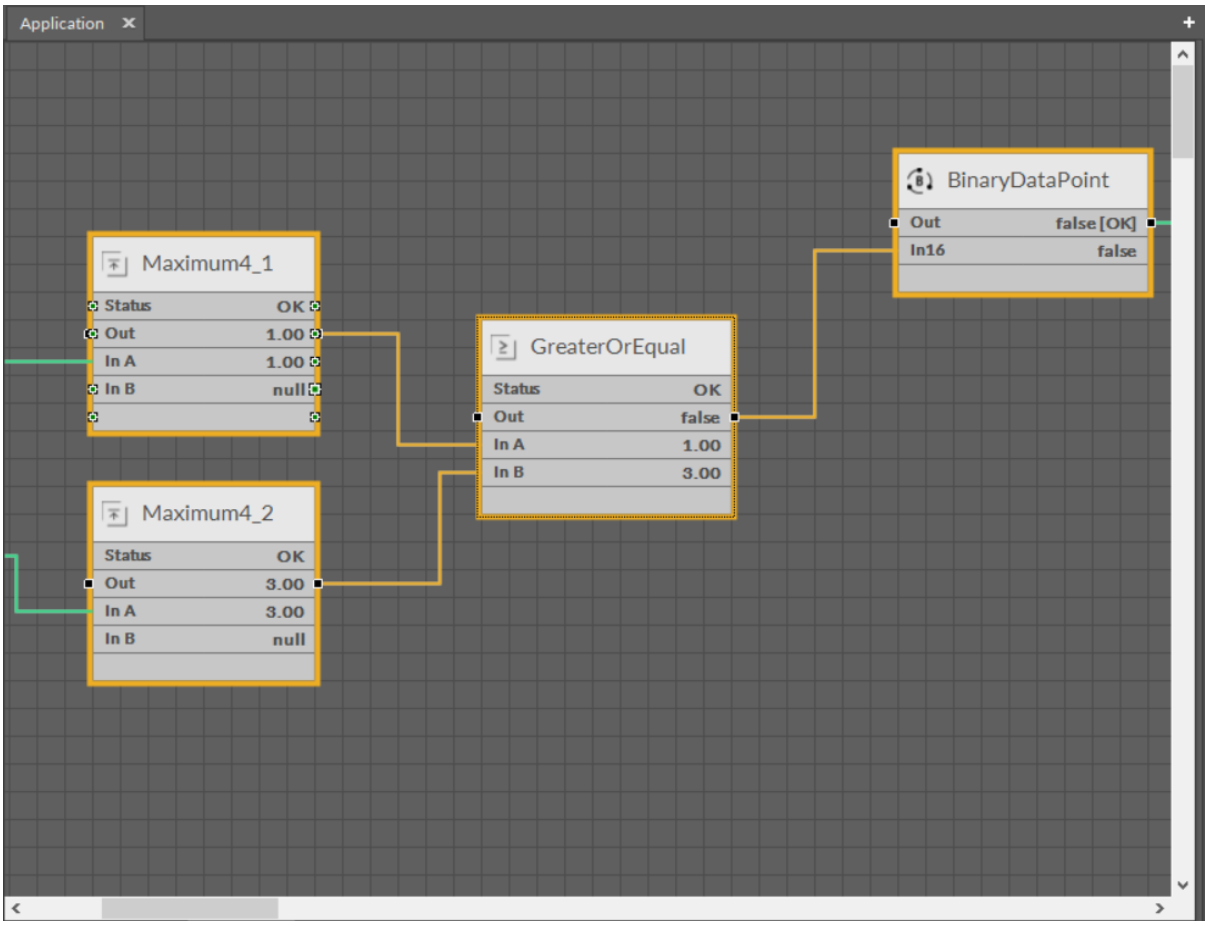



*Figure 105. Crossing selection*

The crossing selection method enables easy removal of all input links of the component without a need to individually select each link. For this purpose drag a narrow rectangle across the component's links that are to be selected (see the figures below). All selected links (orange color) can be deleted with a Delete key or a context menu.

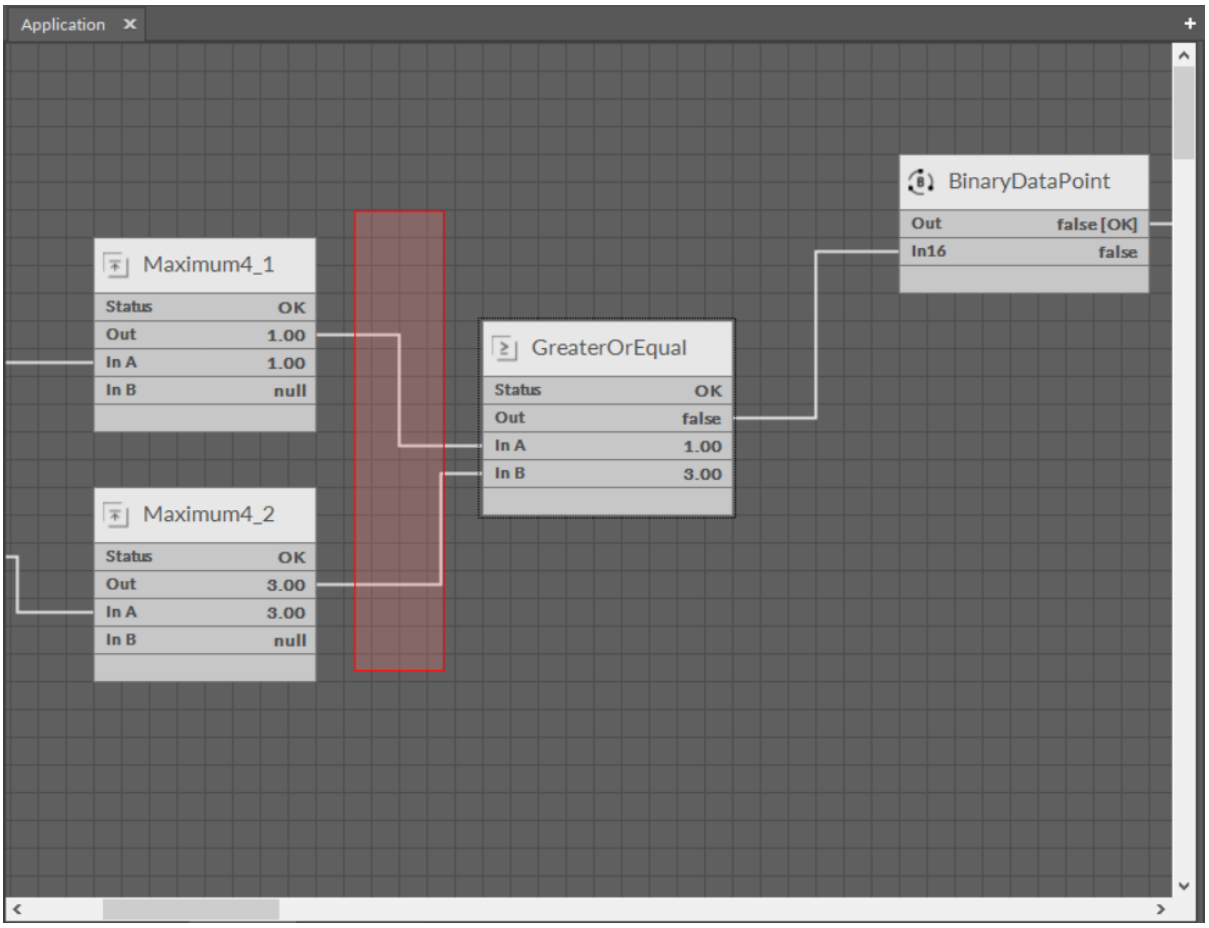



*Figure 106. Crossing selection of links*

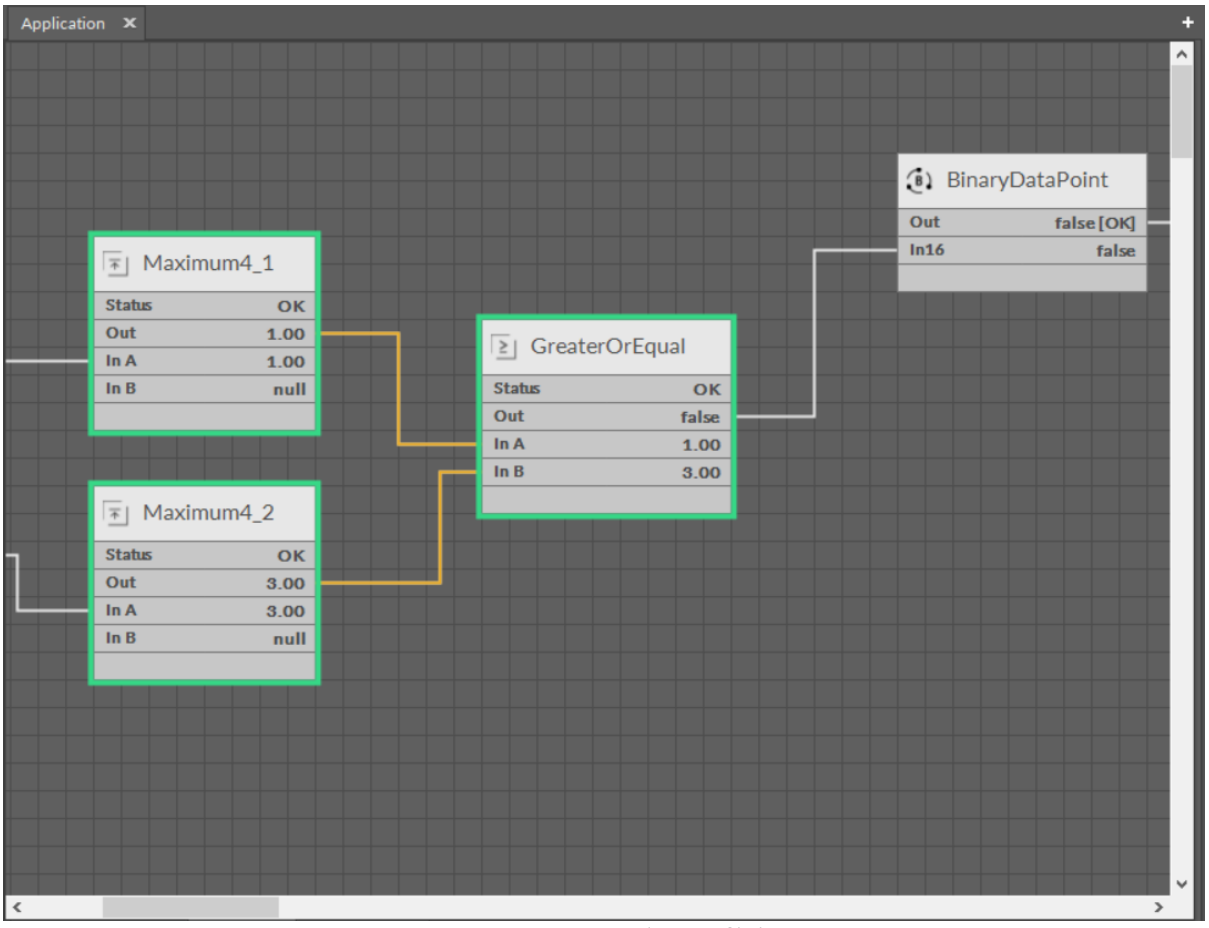



*Figure 107. Crossing selection of links*

## 7.4.4 Adding and Removing Components in the Wire Sheet

### Adding Components

There are three methods of adding a new component to the Wire Sheet view:

- dragging it from the Device Libraries/Kits window;
- copying it in the Device Libraries/Kits window and pasting it in the Wire Sheet tab;
- duplicating a component already added to the Wire Sheet (it is an option in the context menu).

If the component is dragged or pasted to the Wire Sheet, it is released where the mouse points. However, if the component has not been added directly to the Wire Sheet but, for example, placed in the Workspace Tree window, its position in the Wire Sheet will be the same as original component (including cascade offset) in case of pasting or duplicating. In case the component has been dragged from the Device Libraries/Kits window to the Workspace Tree, its position in the Wire Sheet is (0, 0), it will appear in the upper left corner.

In all cases added components are slightly relocated relative to original components, so they do not overlap. The iC Tool uses a cascade mechanism here.

#### Special Adding of Data Points, Schedules, and Folders in nano EDGE ENGINE Devices

The Workspace Tree window, the Wire Sheet and Property Sheet views offer a special way of adding Data Points, schedules, and folders (the Folder and Equipment components) to the Application, simply from the context menu. In order to add components this way, right-click in an empty space of the Wire Sheet grid or Property Sheet and choose the Add option from the context menu. In the Workspace Tree special adding of Data Points, schedules, and folders is available in the context menu of the Application component.

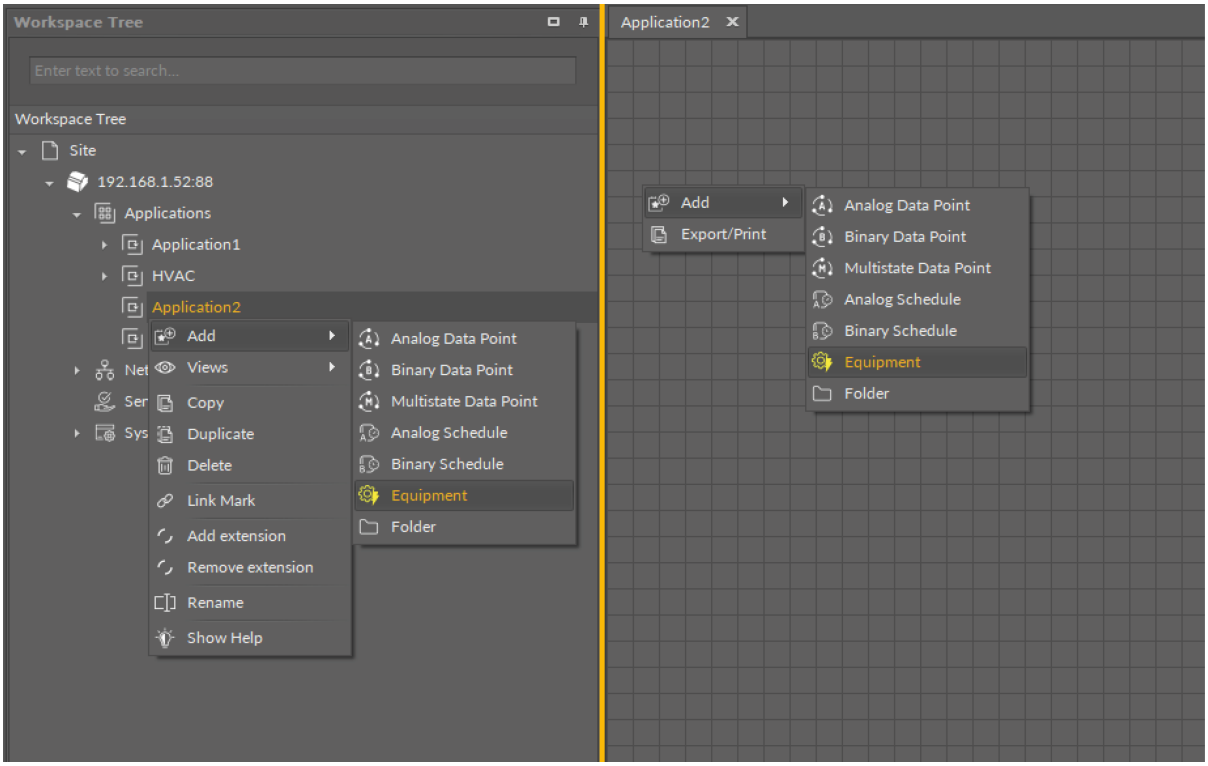



*Figure 108. Adding Data Points/folders from the context menu*

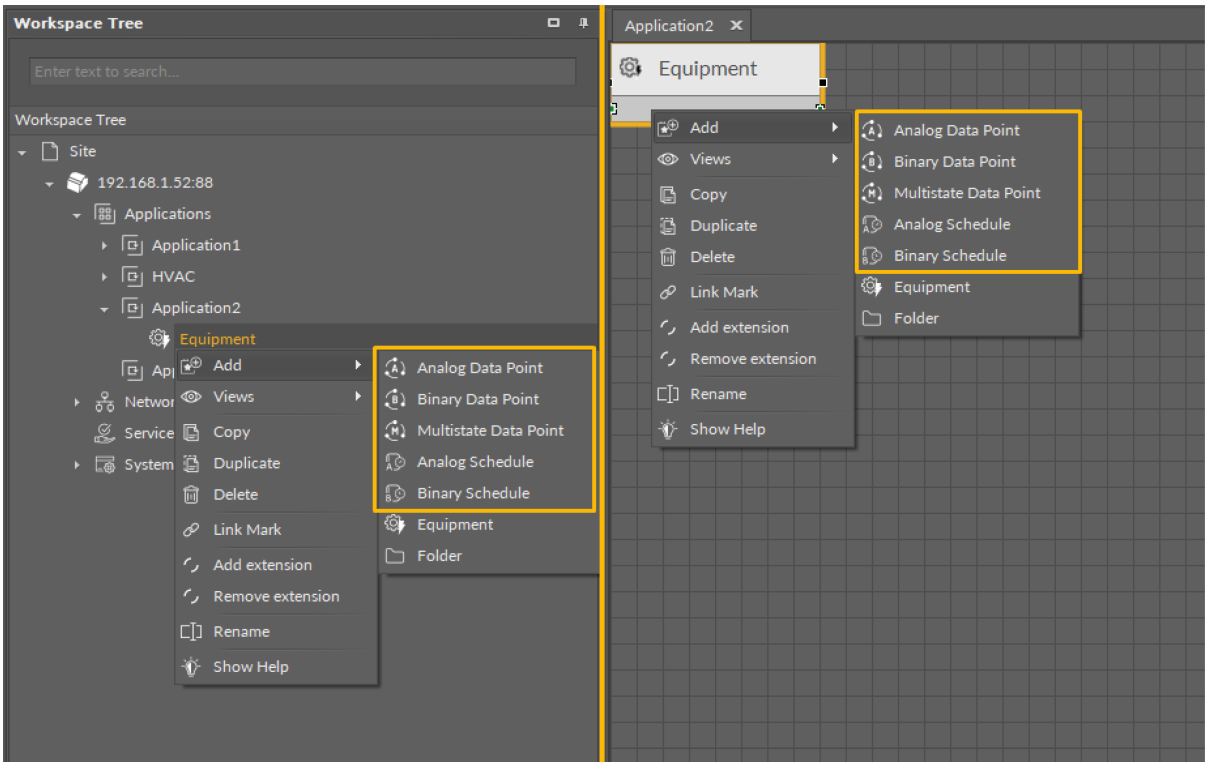



*Figure 109. Adding Data Points to folders from the context menu*

## Removing Components

A component in the Wire Sheet view can be removed after selecting it and pressing the Delete key or choosing the Delete option from the context menu. After removing the component, the iC Tool will automatically remove all links connected to the removed component. If necessary, the remaining links will be automatically redrawn to use the space remaining after the removed component.

# 7.4.5 Adding and Removing Links

## Adding Links

A link is a way of exchanging data between components. It connects components between which the data are transferred. The link identifies, which parameter is taken into account on both sides of the connection. This allows to show the data flow direction, i.e., where a specific parameter is read and where it is saved.

The Wire Sheet view shows the link as a line connecting two components. The Workspace Tree window and the Property Sheet view do not show information about existing links at all.

Views designed to check links for a specific component are the Wire Sheet view and the Links tab in the Object Properties window.

### Reference Links in nano EDGE ENGINE Devices

Linking within applications, built with the nano EDGE ENGINE, may be performed twofold: using a standard linking method or using a special Reference link designed specifically to connect Data Point class components (in the Applications container) with network point class components (in the Networks container):

- The Reference link is a special compound link designed to connect Data Points with network points. The Reference link is created between special Reference slots and **transfers values along with the component's status**. Alternatively, it may transfer values between Data Points and network points at the same time returning status from network points to Data Points, or it may return values from network points to Data Points.
- The standard linking method involves simple creating links between the input and output slots; a standard link transfers a value between the connected slots. Standard linking may be applied between all four containers of the nano EDGE ENGINE device structure.

The **Reference link is unique for the nano EDGE ENGINE solution**, and it is a recommended method to be used in application building as **it offers an advantage of transferring the status information between components.**

**Worth to Notice:**

By default, not all slots of a particular component are always visible in the Wire Sheet view. The hidden slots are not displayed in the component default view. A hidden slot becomes visible once it is an end slot to a link. The link to hidden slot may be created using the linking dialog window.

Links in the iC Tool are created in two ways:

- using the **Link Mark** and **Link From** options from the context menu: the Link Mark option selects the source component, and the Link From option selects the target component. The context menu with the Link Mark and Link From options may be opened in the Workspace Tree, Wire Sheet, and Property Sheet;
- **graphically**: drawing a line from the source component to the target component. The link may be created graphically in the Wire Sheet. In order to create link graphically, click on the source slot, and drag the line to the target slot. The iC Tool automatically highlights the fitting target slots. As the Wire Sheet displays limited number of slots in the component, drawing a line to the empty row opens a dialog window with the list of all slots to select the target slot.

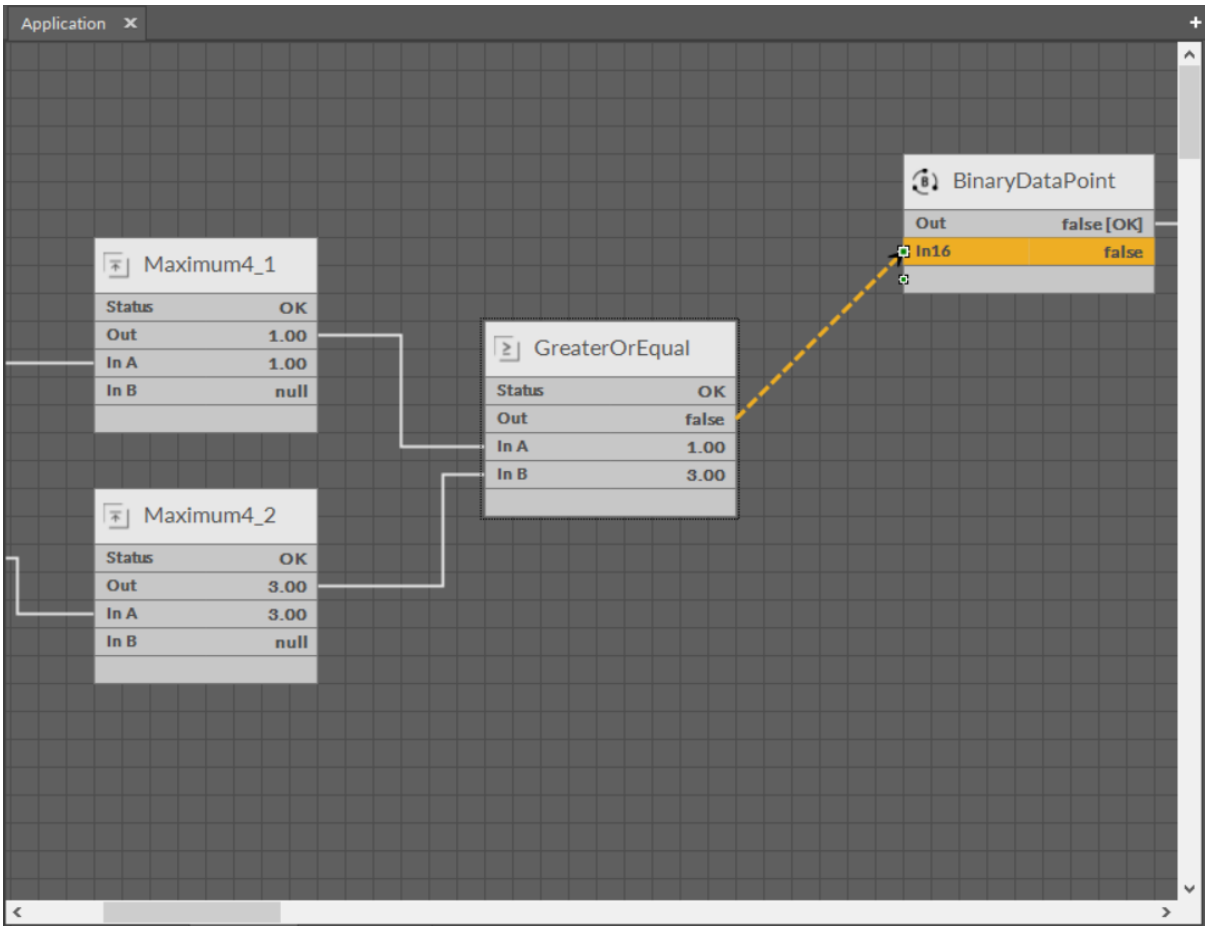



*Figure 110. Creating links graphically*

A dialog window shows the selected output slot (in the left column) and input slots available to select (in the right column). The available input slots are highlighted in the right column. Selecting the input slot has to be confirmed by pressing Ok.

Creating a link, regardless of a method used, is directly saved to the controller.

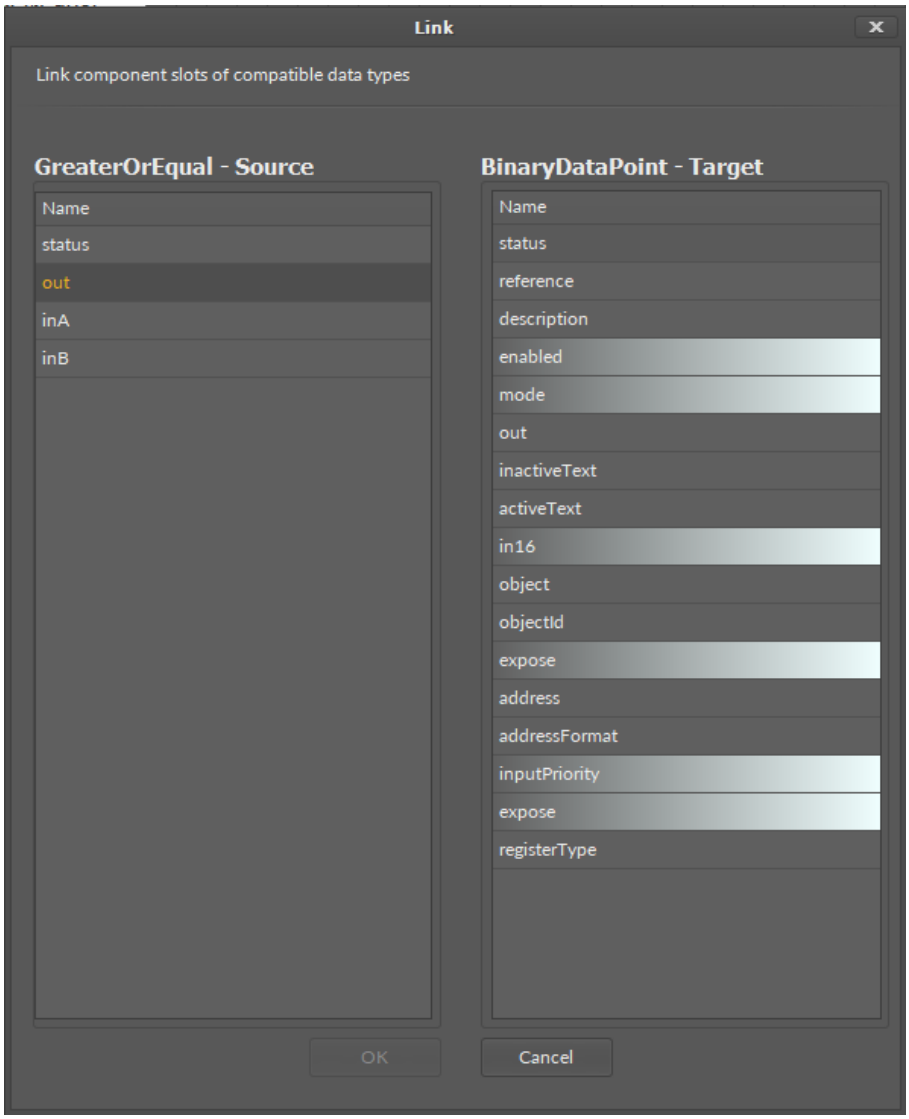



*Figure 111. A linking dialog window*

## Removing Links

A link in the Wire Sheet view can be removed after selecting it and pressing Delete key or choosing a Delete link(s) option from the context menu. If necessary, the remaining links will be automatically redrawn to use the space remaining after the removed one.

## 7.4.6 Links Between Wire Sheet Tabs

Each Wire Sheet tab displays components belonging to a common superior component. More complex applications require to create a logical connection between components belonging to different superiors, therefore, displayed in different Wire Sheet tabs. The simplest example of such logical connection would be linking network points with Data Points: network points are nested in the LocalIO component, which is placed in the Networks container; Data Points, on the other hand, are located in the Application component in the Applications container; therefore, linking them requires creating links between two separate views.

Links between components belonging to different superiors are displayed as full circles coming out of linked slots. It means that the further part of such link is continued in another Wire Sheet tab.

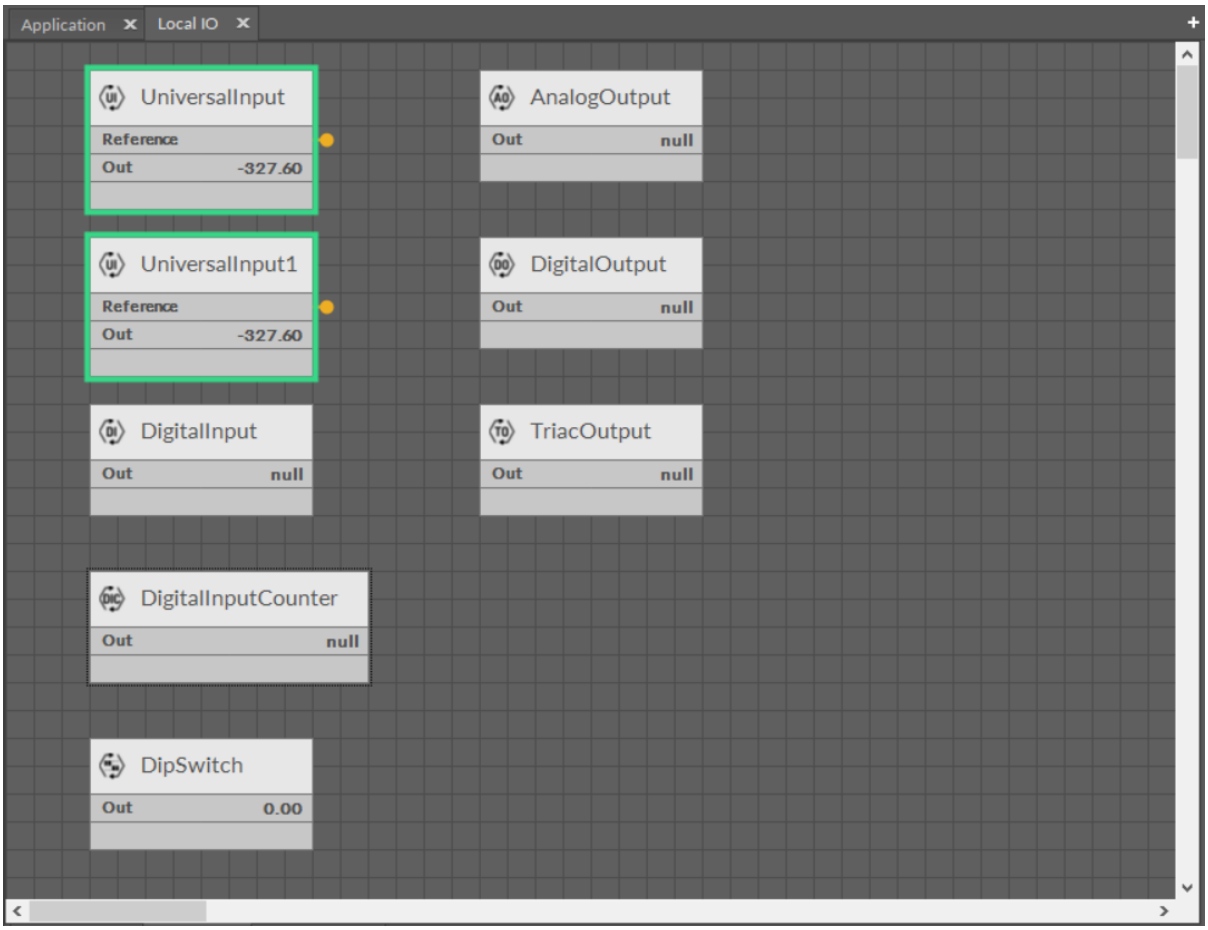



*Figure 112. Two UniversalInputs linked with Analog Data Points*

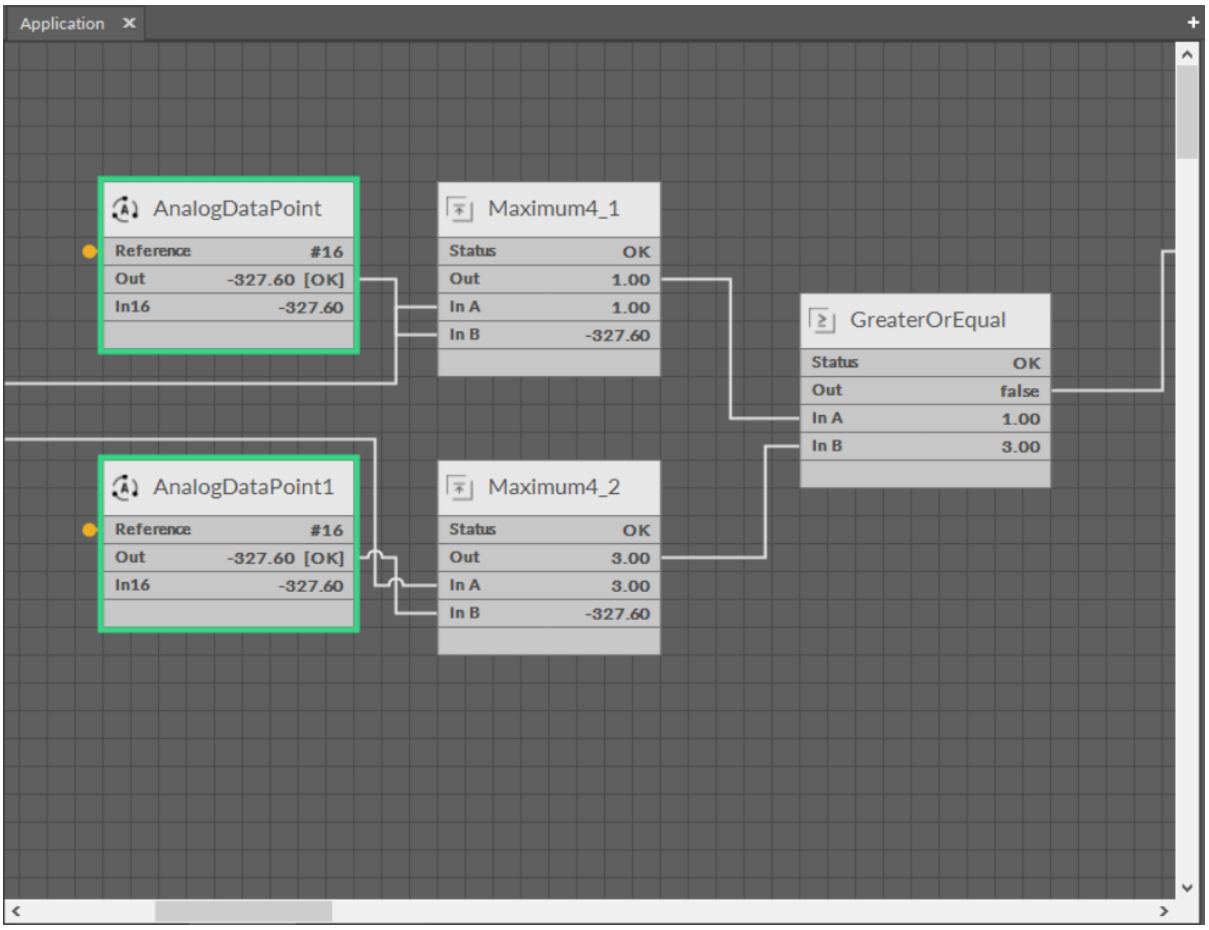



*Figure 113. Two UniversalInputs linked with Analog Data Points*

Additionally, in case there is more than one link coming out from one output slot, a number of links is displayed next to the link circle. Detailed information about the links' directions, slots, and components can be obtained in the Object Properties window after clicking the links circle.

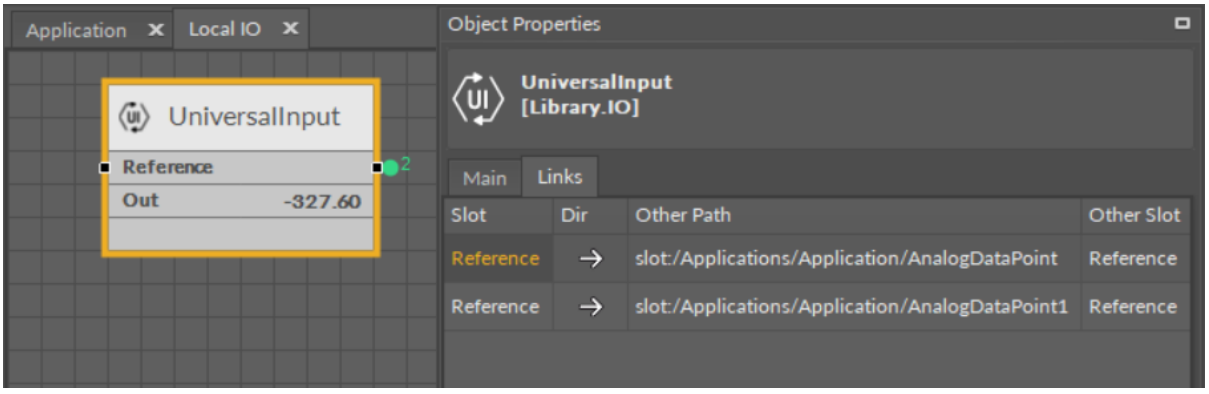



*Figure 114. Two external links coming out of the Reference slot*

## 7.5 Property Sheet

A Property Sheet is a tabular view listing all of the selected component's slots, its inferior components, and their slots. The Property Sheet provides information about slots' names, their current values, units, and ranges .

The Property Sheet view allows to select only one component at a time to perform basic functions such as copying, duplicating, and removing.

Each component in the Property Sheet view is displayed in an expandable tree structure. Slots, which values can be edited, are highlighted with a lighter background.

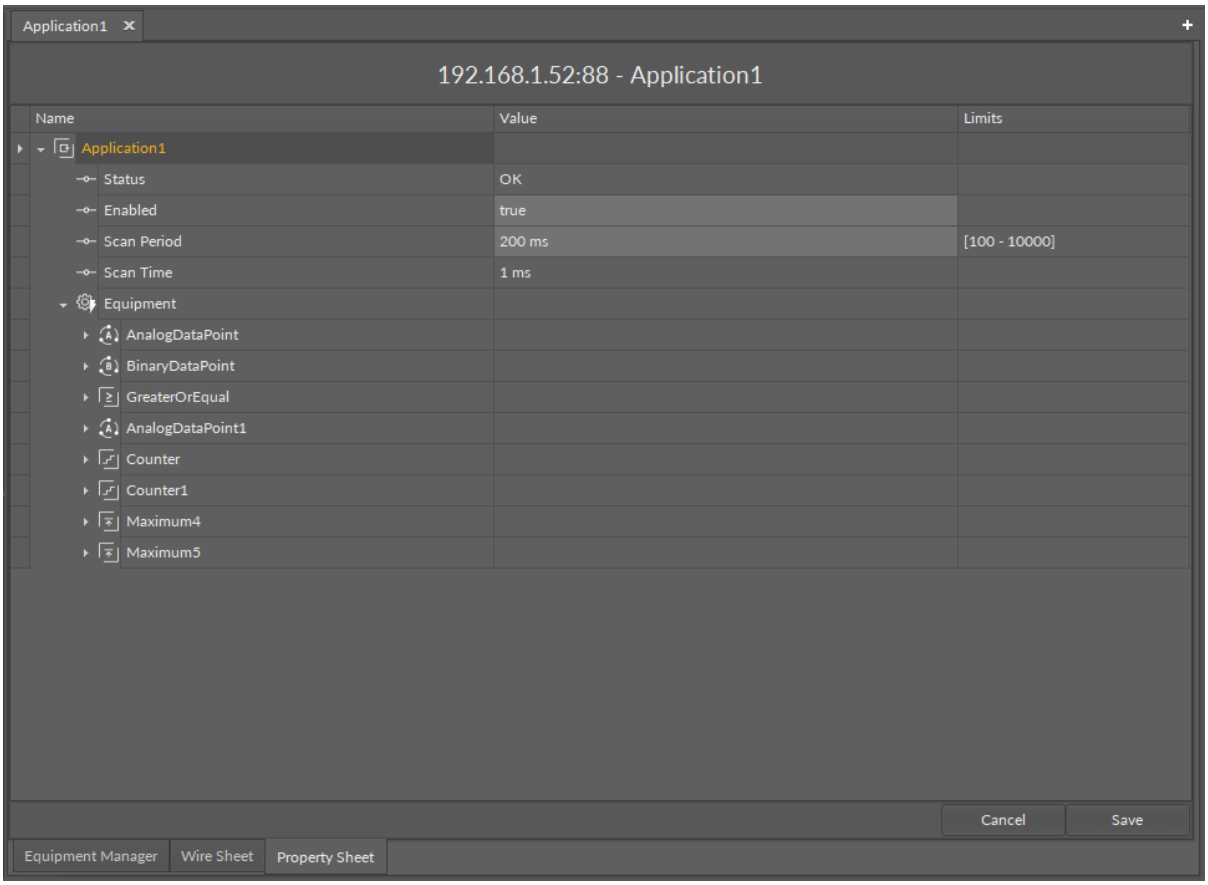



*Figure 115. The Property Sheet view*

The Property Sheet view shows three columns:

- name of component;
- value;
- limits.

**Worth to Notice**

The Limits column shows the range of values available for component. If the range of values is unlimited, the field is empty.

## 7.5.1 Adding and Removing Components in the Property Sheet

### Adding Components

In order to add a new component to the Property Sheet view, drag it from the Device Libraries/Kits window straight to the Property Sheet tab. As the Property Sheet tab can only be opened for a selected component (displaying a plain Property Sheet tab is not possible), the dragged component will always be added as an inferior of the selected component. If the dragged component is dropped in an unoccupied field in the Property Sheet view, it will be added to the main component of this view.

## Special Adding of Data Points and Folders in nano EDGE ENGINE Devices

The Workspace Tree window, the Wire Sheet and Property Sheet views offer a special way of adding Data Points and folders (the Folder and Equipment components) to the Application, simply from the context menu. In order to add components this way, right-click in an empty space of the Wire Sheet grid or Property Sheet and choose the Add option from the context menu. In the Workspace Tree special adding of Data Points and folders is available in the context menu of the Application component.

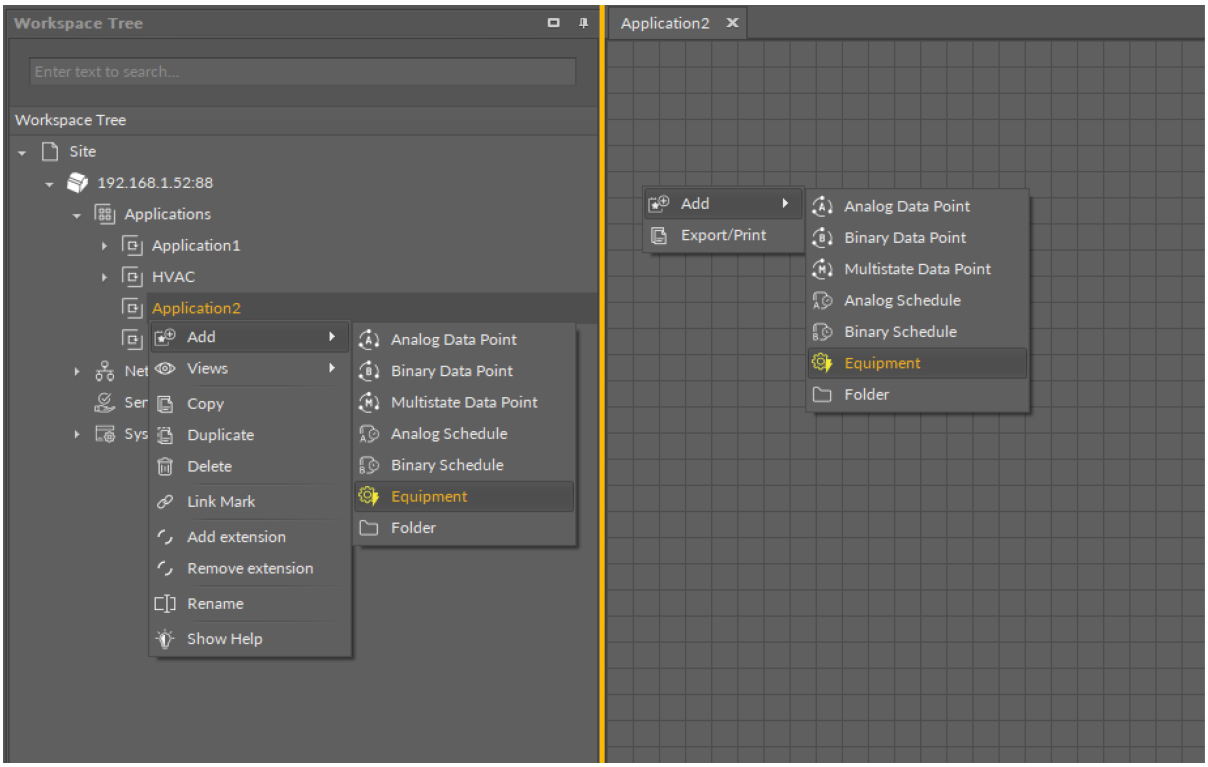



*Figure 116. Adding Data Points/folders from the context menu*

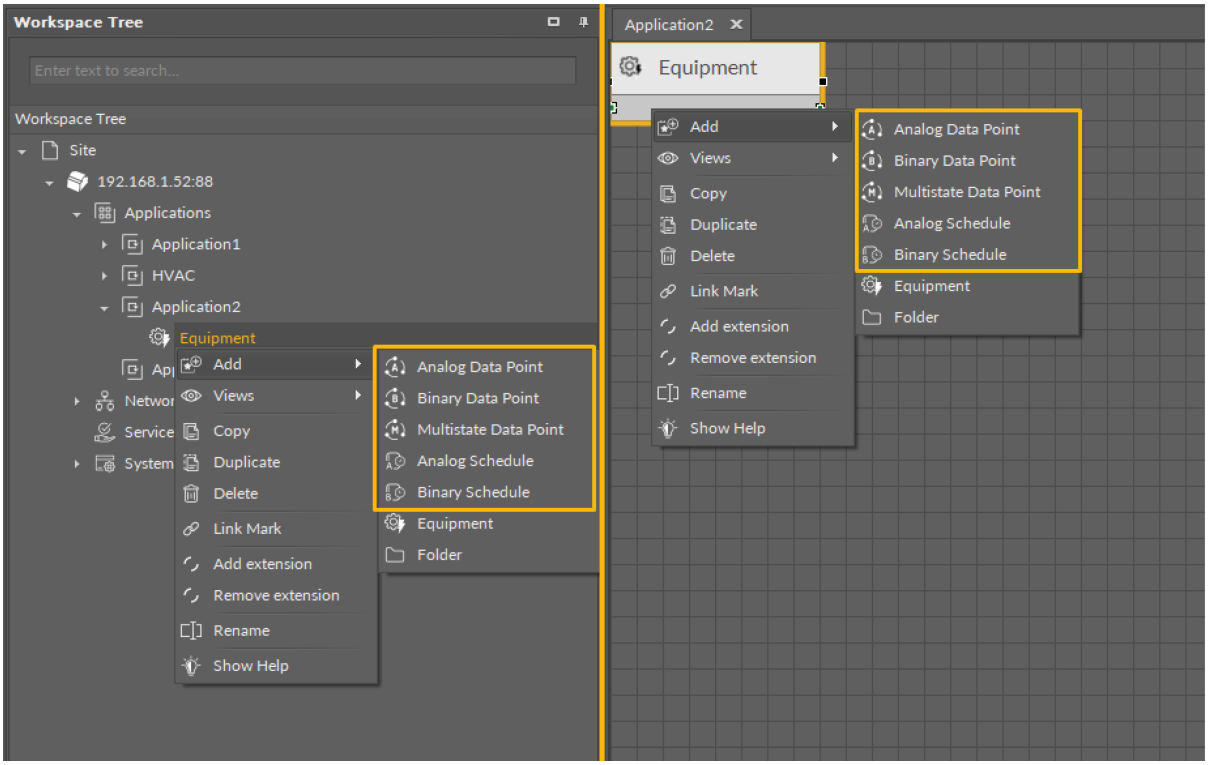



*Figure 117. Adding Data Points to folders from the context menu*

### Removing Components

A component selected in the Property Sheet tab can be deleted with a Delete key or a Delete option in the context menu.

## 7.5.2 Creating Links

The only way of creating a link in the Property Sheet view is using the context menu with the Link Mark and Link From options.

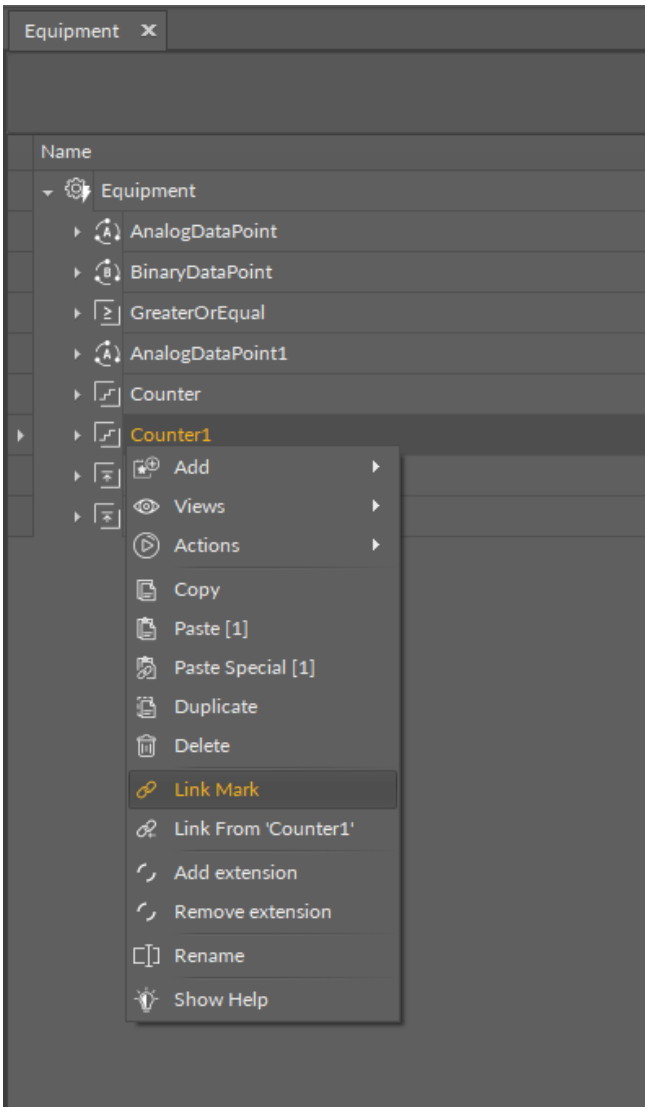



*Figure 118. The Link Mark option*

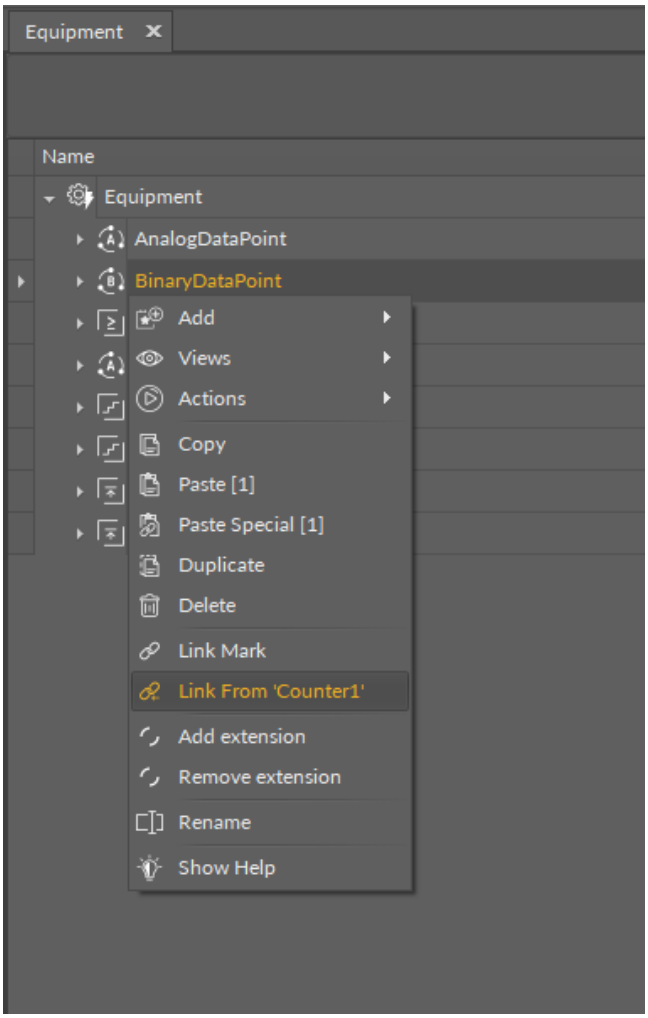



*Figure 119. The Link From option*

There is no possibility of removing a link in the Property Sheet view. To remove a link use the Wire Sheet or Object Properties view.

# 8 iC Tool with nano EDGE ENGINE Devices

For nano EDGE ENGINE devices, the iC Tool offers special views that are tailored for the nano EDGE ENGINE functionalities.

## 8.1 Applications Manager

The Applications Manager is a special view that allows to manage the Application components added to the Applications container.

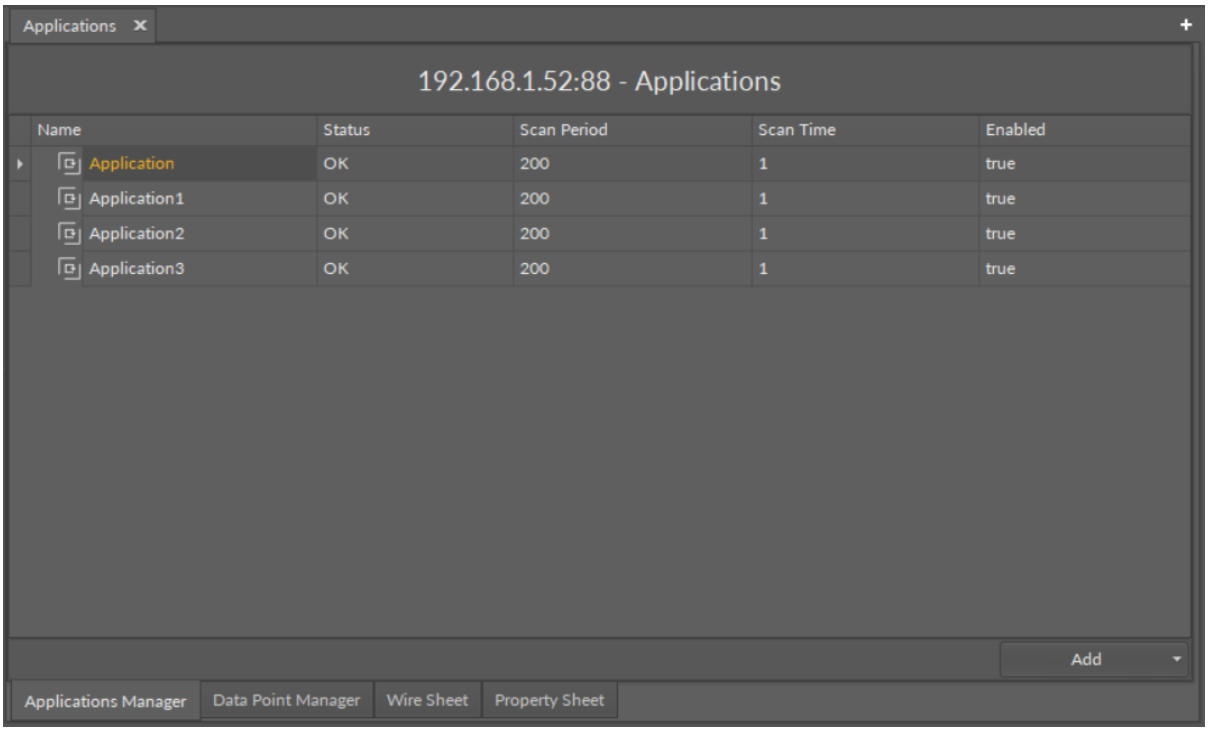



*Figure 120. The Applications Manager*

The Applications Manager lists all the Application components used on the device. The view shows the following fields:

- name of the application;
- status;
- scan period;
- scan time;
- enabled or disabled status.

In the Applications Manager, it is possible to add, remove, copy, or duplicate Application components.

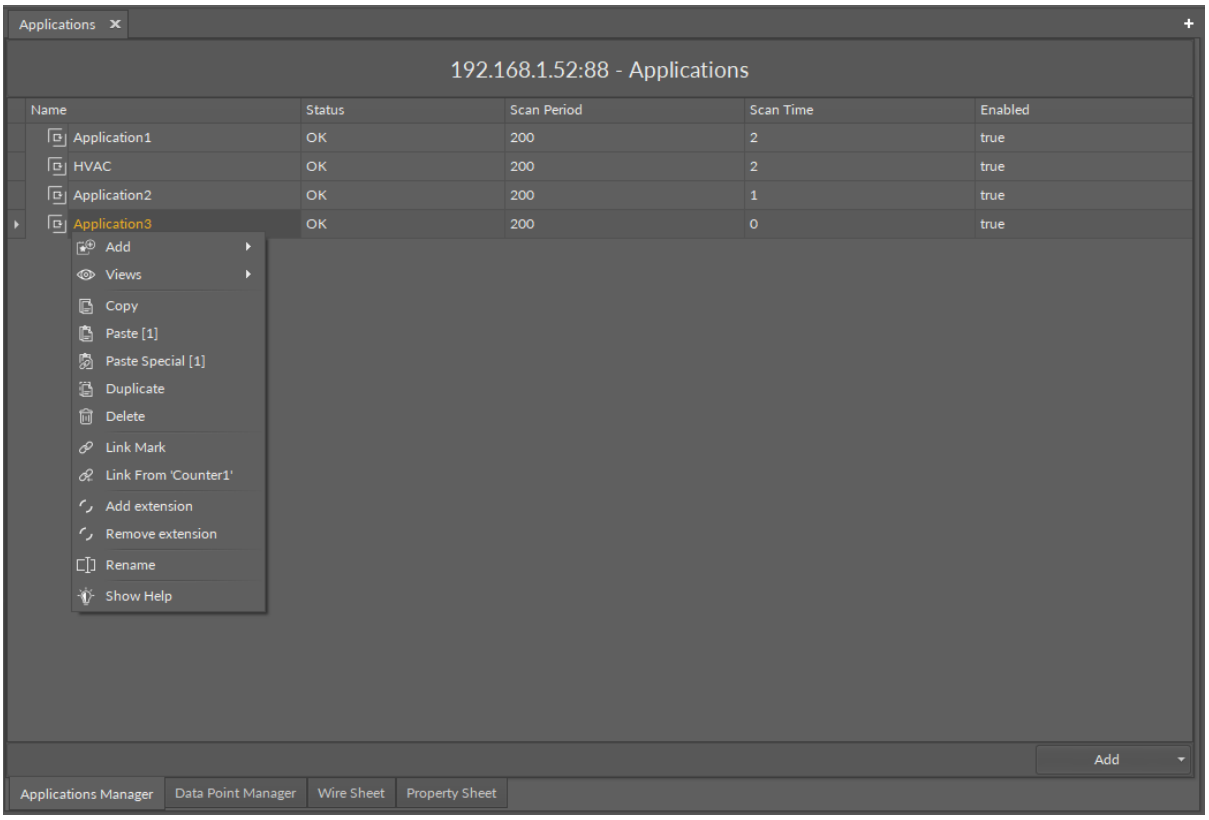



*Figure 121. Options in the Applications Manager*

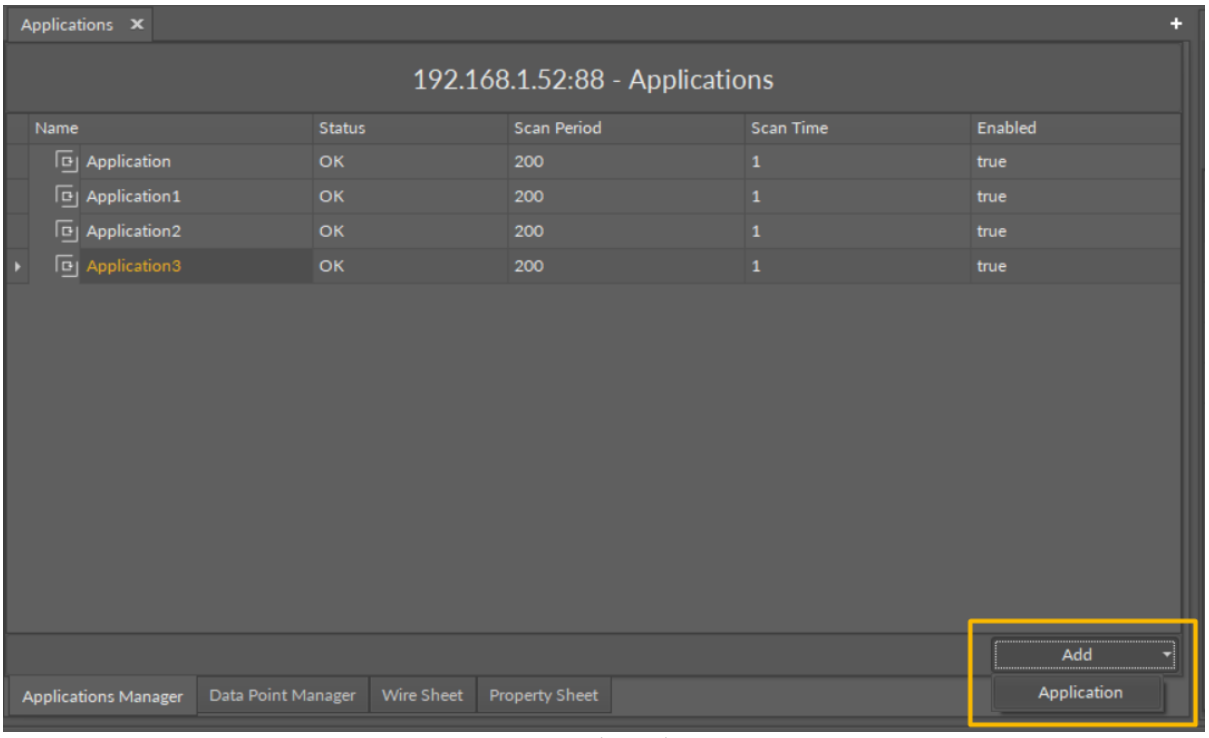



*Figure 122. Options in the Applications Manager*

## 8.1.1 Opening the Applications Manager

The Applications Manager view is accessible in the context menu of the Applications container.

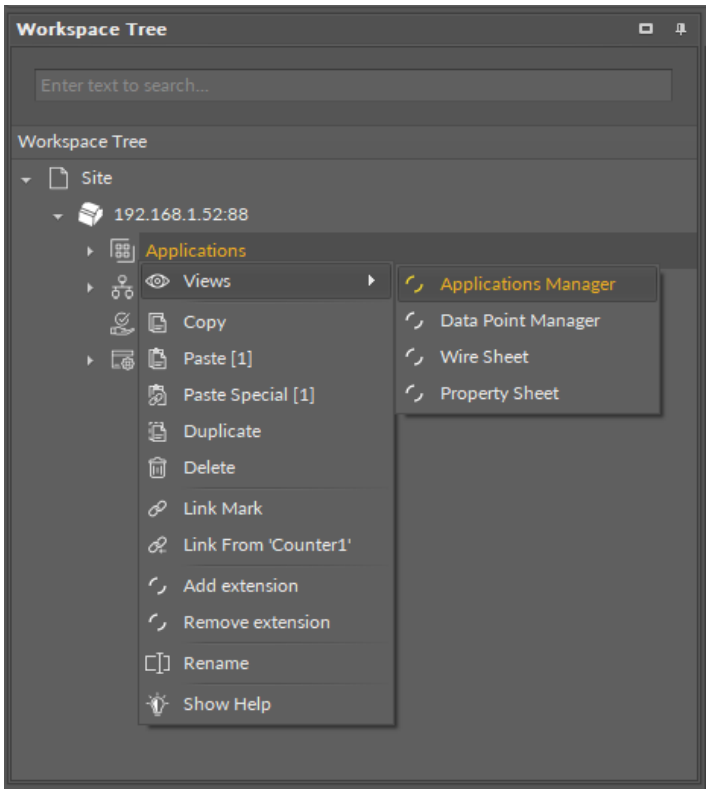



*Figure 123. Accessing the Applications Manager from the Applications container context menu*

The Applications Manager view is also automatically opened if the Applications container is double-clicked in the Workspace Tree window.

## 8.2 Equipment Manager

The Equipment Manager is a special view that allows to manage the Equipment components, Data Points, and Folder components added to a specific Application component.

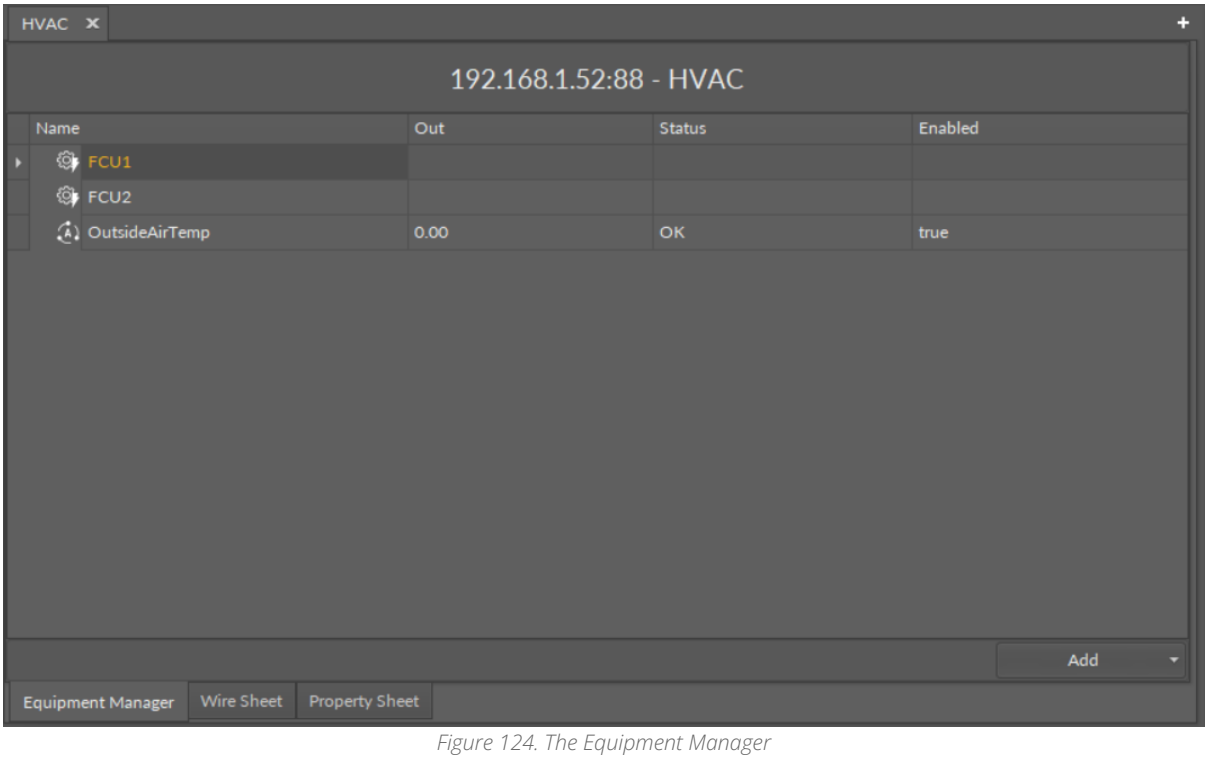



*Figure 124. The Equipment Manager*

The Equipment Manager lists all Equipment components and Data Points used in the specific Application component. The view shows the following fields:

- name of the Data Point;
- value on the Out slot;
- Data Point's status;
- enabled or disabled status.

The Equipment Manager view is not editable; however, it allows to add, remove, duplicate, and rename Equipment components, Data Points (withing the Equipment components and directly in the Application component), and Folder components, which help organize components in the application.

Once a specific Data Point is clicked in the Equipment Manager view, it is displayed in the Object Properties window, where it can be freely edited. Also, the view allows for multiediting of compatible Data Points–if compatible Data Points are selected in the Application Manager, their common slots are available for multiedition in the Object Properties window.

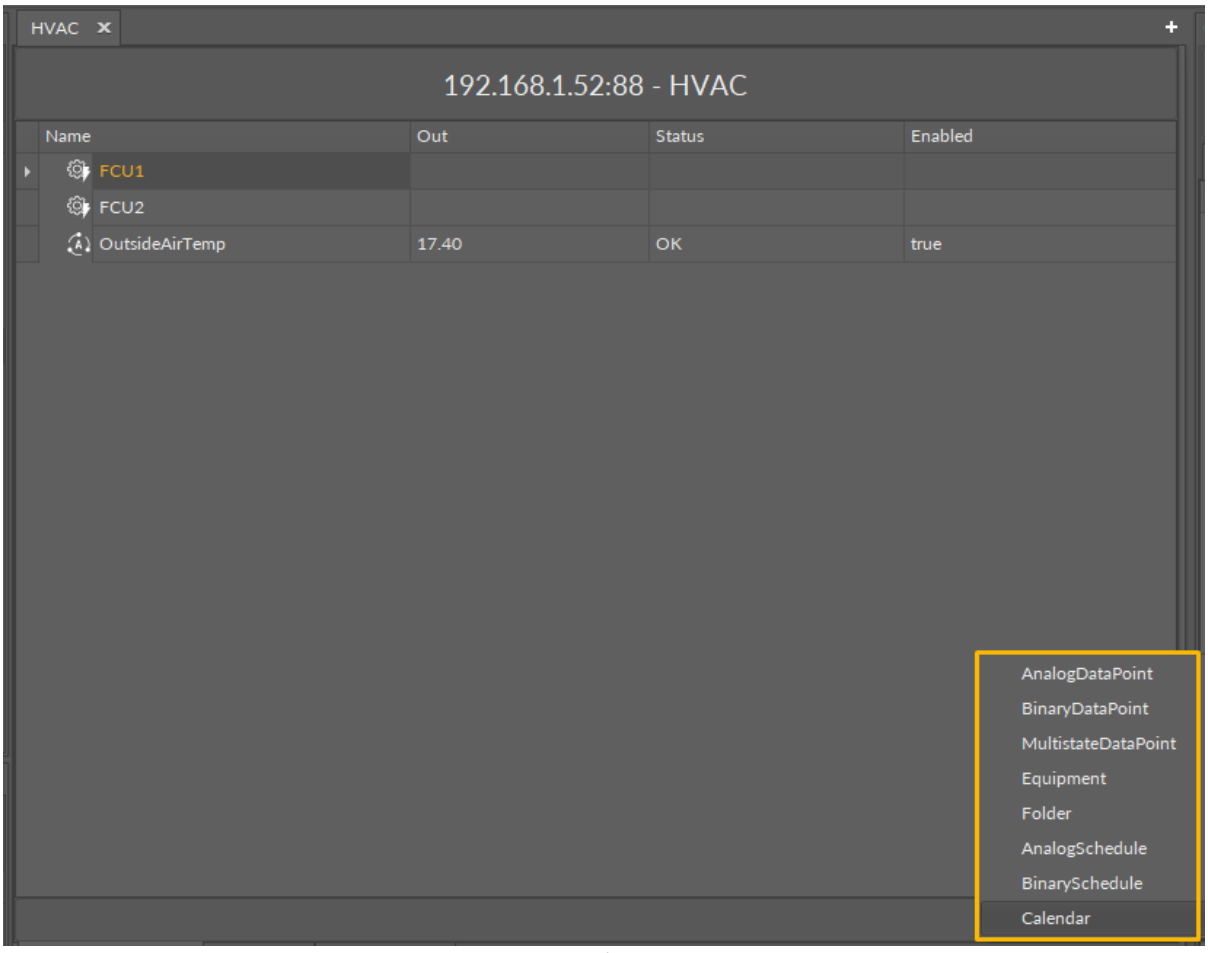



*Figure 125. Options in the Equipment Manager*

## 8.2.1 Opening the Equipment Manager

The Equipment Manager view is accessible in the context menu of the Application component.

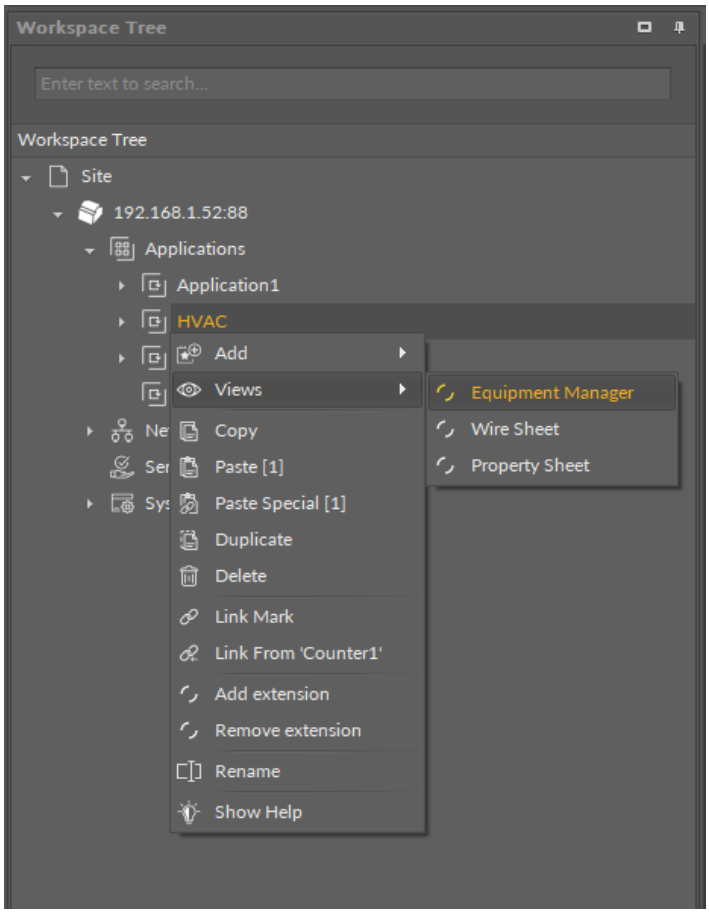



*Figure 126. Accessing the Application Manager from the Applications container context menu*

The Equipment Manager view is also automatically opened if either the Application component is double-clicked in the Workspace Tree window.

## 8.3 Data Point Manager

The Data Point Manager is a special view that allows to manage the Data Points available within the nano EDGE ENGINE license.

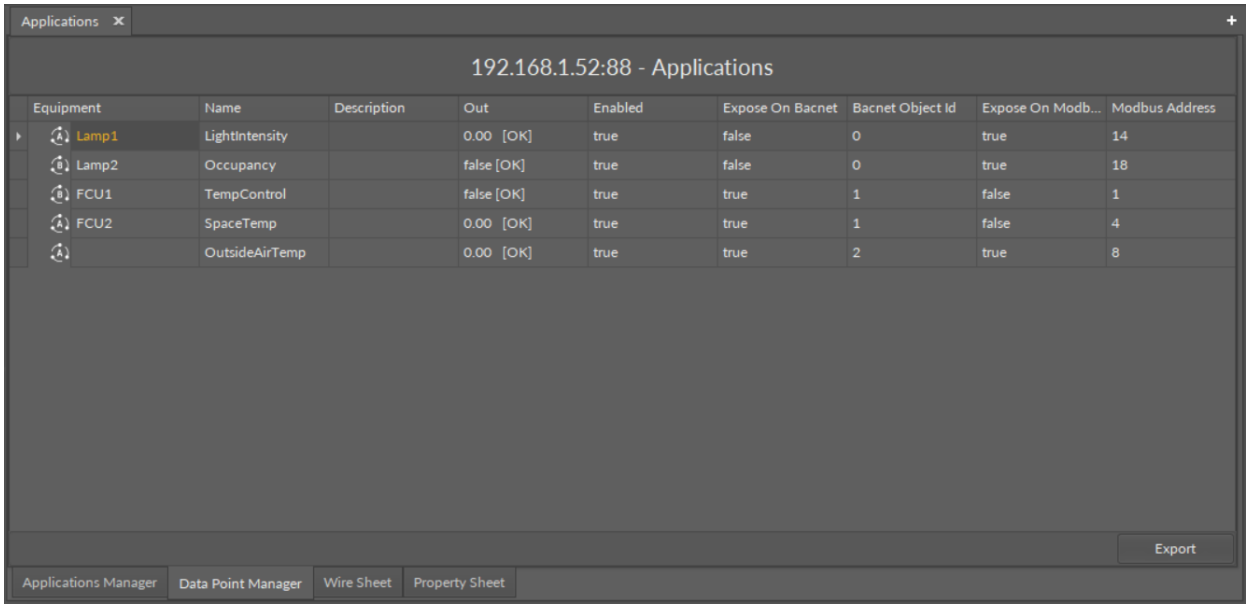



*Figure 127. The Data Point Manager*

The Data Point Manager lists all the Data Points used in applications saved on the device. The view shows the following fields:

- Equipment, which the Data Point belongs to;
- name of the Data Point;
- description;
- value on the Out slot;
- enabled or disabled status;
- exposed on BACnet status;
- BACnet object Id;
- exposed on Modbus status;
- Modbus address.

The Data Point Manager view is not editable; however, once a specific Data Point is clicked in the Manager view, it is displayed in the Object Properties window, where it can be freely edited. Also, the view allows for multiediting of compatible Data Points–if compatible Data Points are selected in the Data Point Manager, their common slots are available for multiedition in the Object Properties window.

### 8.3.1 Opening the Data Point Manager

The Data Point Manager view is accessible from two locations:

- in the context menu of the Applications container;
- in the context menu of the RAC18-IP (Device for BACnet) component;
- in the context menu of the RAC18-IP (Device for Modbus) component.

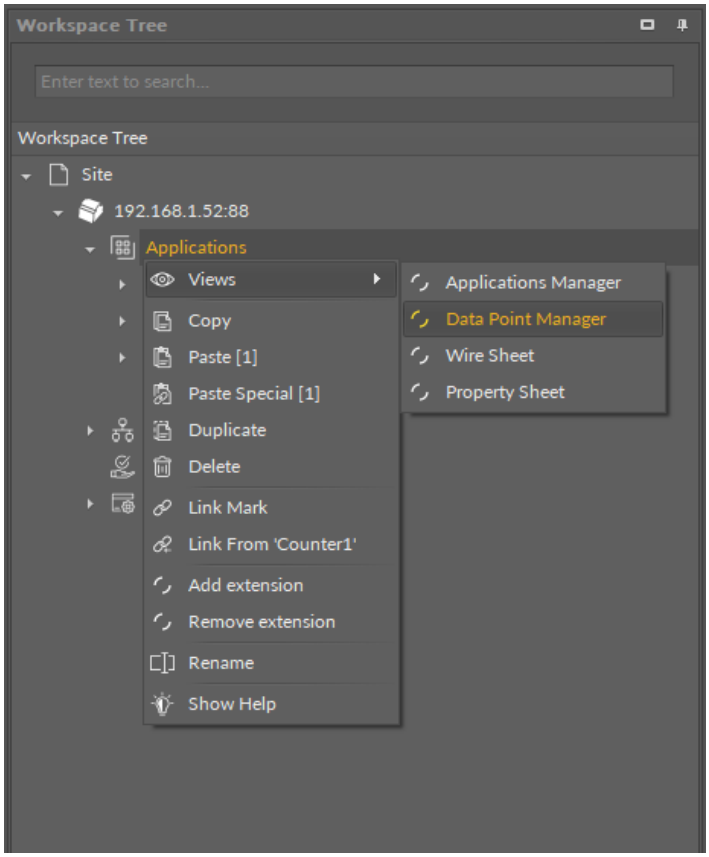



*Figure 128. Accessing the Data Point Manager from the Applications container context menu*

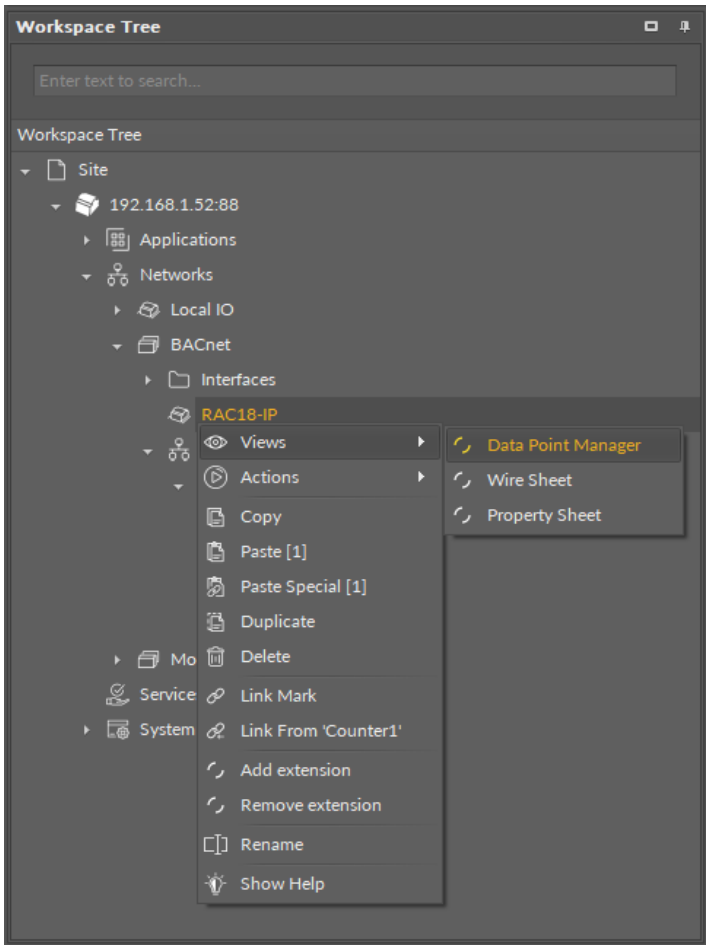
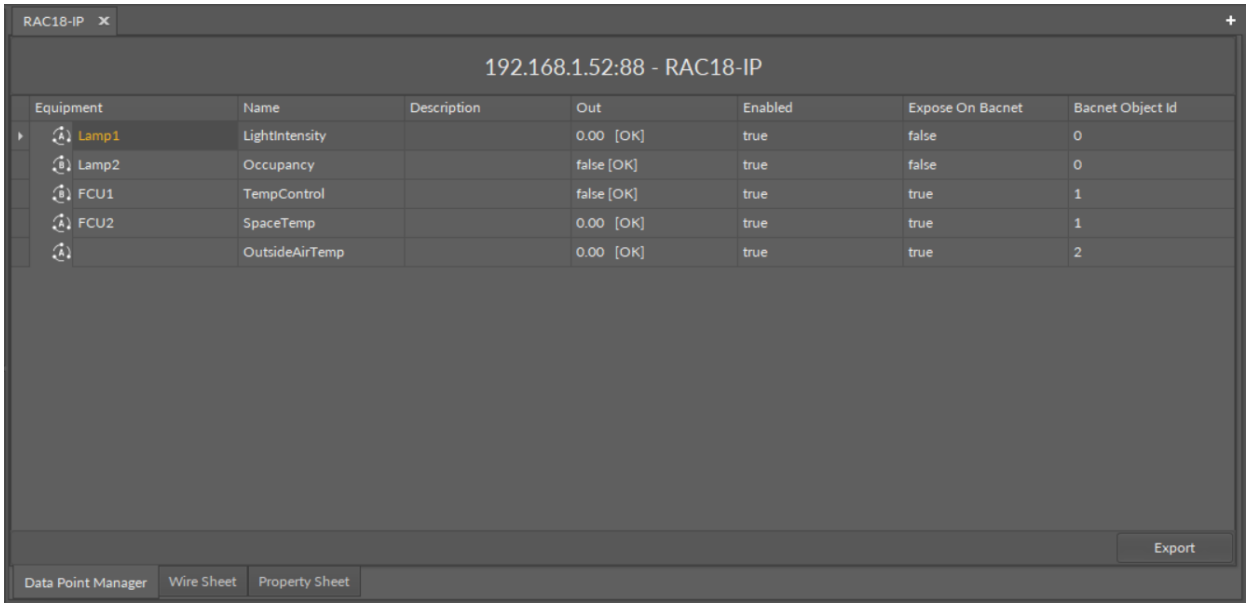



*Figure 129. Accessing the Data Point Manager from the context menu of RAC18-IP (LocalDevice) component in the Networks container*

The Data Point Manager view is also automatically opened if either the Applications container or the RAC18-IP (Device for BACnet or Modbus) component is double-clicked in the Workspace Tree window.

The view adjusts its contents if opened for the BACnet or Modbus RAC18-IP Device showing information specific to the network:




| Equipment | Name | Description | Out | Enabled | Expose On Bacnet | Bacnet Object Id |
|---|---|---|---|---|---|---|
| Lamp1 | LightIntensity | | 0.00 [OK] | true | false | 0 |
| Lamp2 | Occupancy | | false [OK] | true | false | 0 |
| FCU1 | TempControl | | false [OK] | true | true | 1 |
| FCU2 | SpaceTemp | | 0.00 [OK] | true | true | 1 |
| | OutsideAirTemp | | 0.00 [OK] | true | true | 2 |

*Figure 130. The Data Point Manager for the RAC18-IP BACnet LocalDevice*

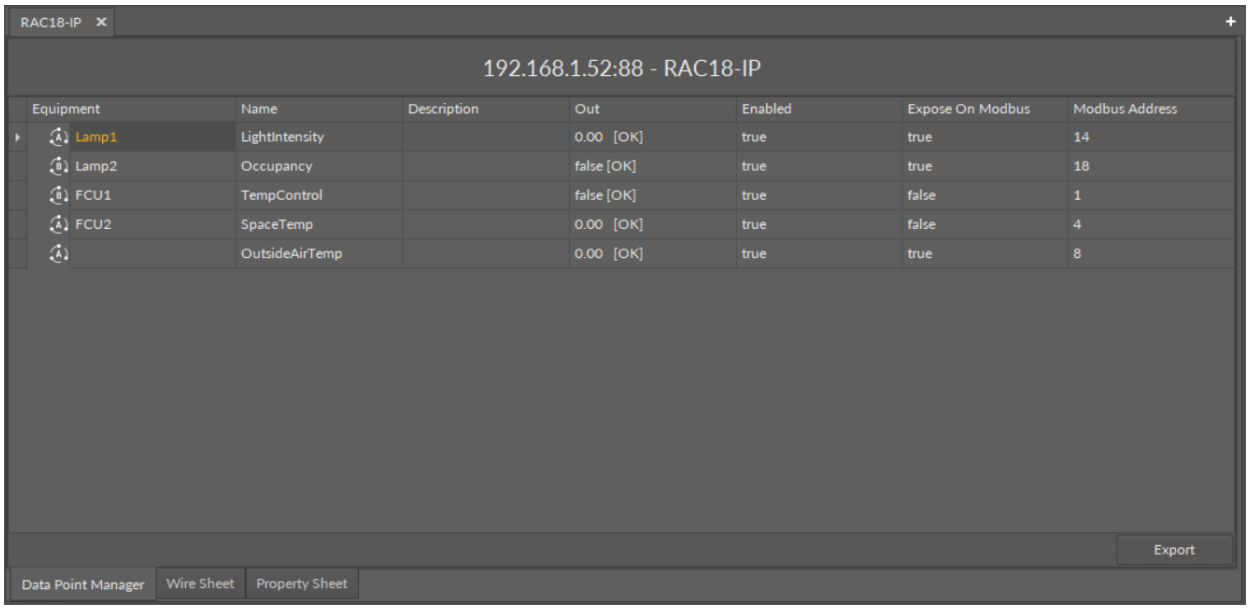



*Figure 131. The Data Point Manager for the RAC18-IP Modbus LocalDevice*

**Licensing**

The license for the new generation of iSMA CONTROLLI controllers driven by the nano EDGE ENGINE is constructed against the number of Data Points: each device based on the nano EDGE ENGINE is granted a specified number of license points (Data Points in this case), which can be used within applications. Therefore, the licensing system is only of quantitative, not functional, character–only the real number of Data Points in applications is taken into account, regardless of how many communication protocols are used to expose them, or how many network points are controlled. With the nano EDGE ENGINE-generation devices it is possible to create as big an application (or applications) as the number of licensed Data Points. No elements in the Networks, Services, or System containers are subject to license limitations, other than Data Points in the Applications container.

Note: In order to check the number of license points, please refer to the License in the device.

## 8.4 Network Manager

### 8.4.1 BACnet

The Network Manager view is available for the BACnet component. It lists all BACnet networks configured on the device's ports. The Network Manager view shows the statuses, ports (which the network is configured on), and enabled or disabled states of the the network. Once the network, listed in the Network Manager, is double-clicked, the respective Network component is opened.

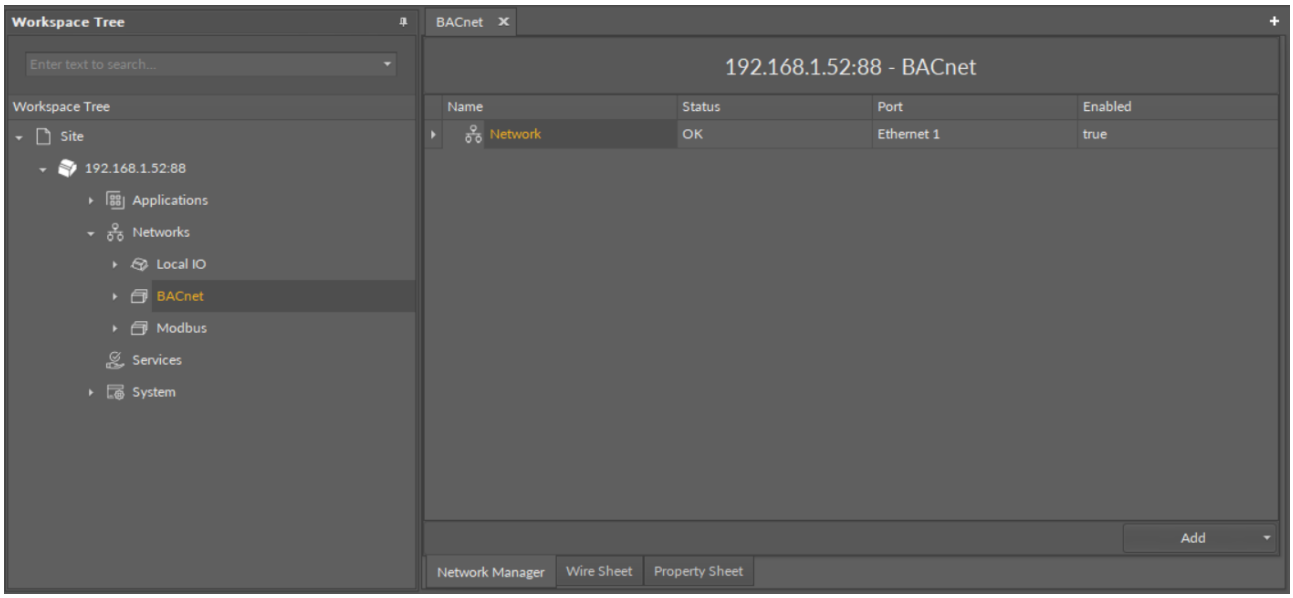



*Figure 132. The Network Manager for BACnet*

## Opening Network Manager

The Network Manager view is accessible from the context menu of the BACnet component. It is also automatically opened if the BACnet component is double-clicked in the Workspace Tree window.

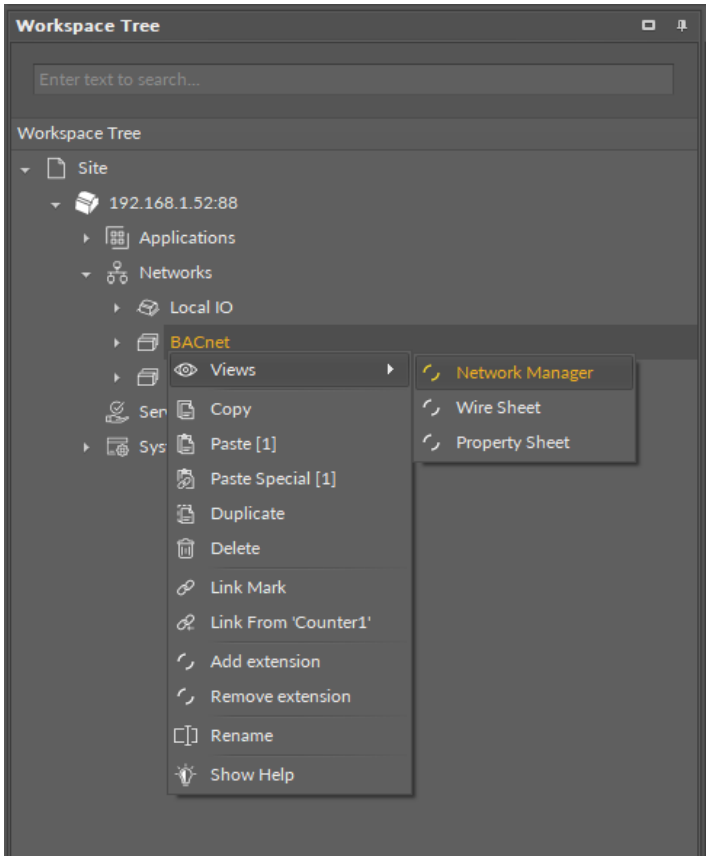



*Figure 133. Opening the Network Manager*

## Adding BACnet Networks

The networks may be added to the Network Manager twofold: dragging and dropping the Network component to the BACnet component from the Core library (in the Device Libraries window), or using a special Add function in the Network Manager view available in the bottom right corner.

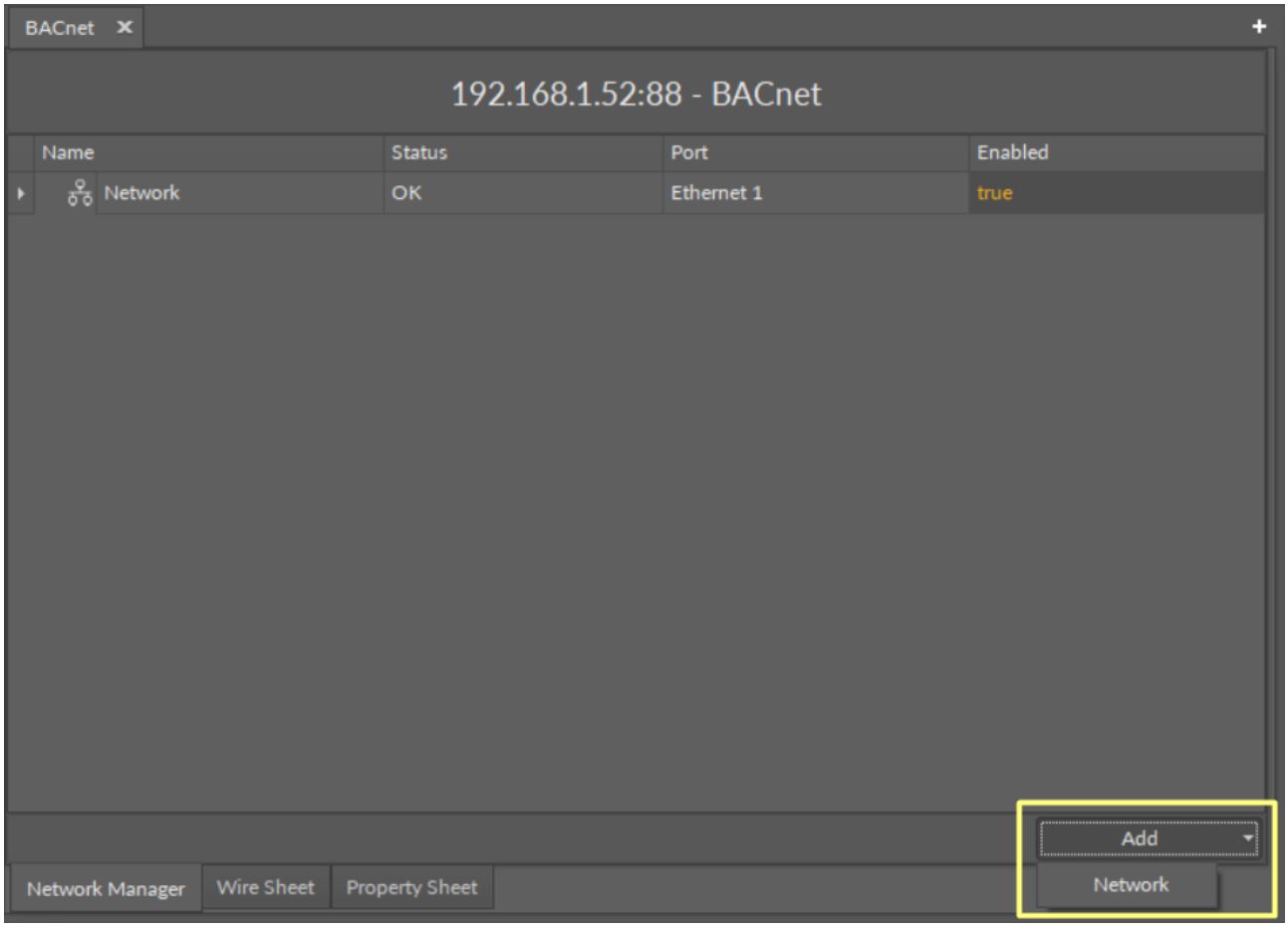



*Figure 134. The Add button*

Using this Add button opens the dialog window, which allows to adjust the quantity of networks to be added.

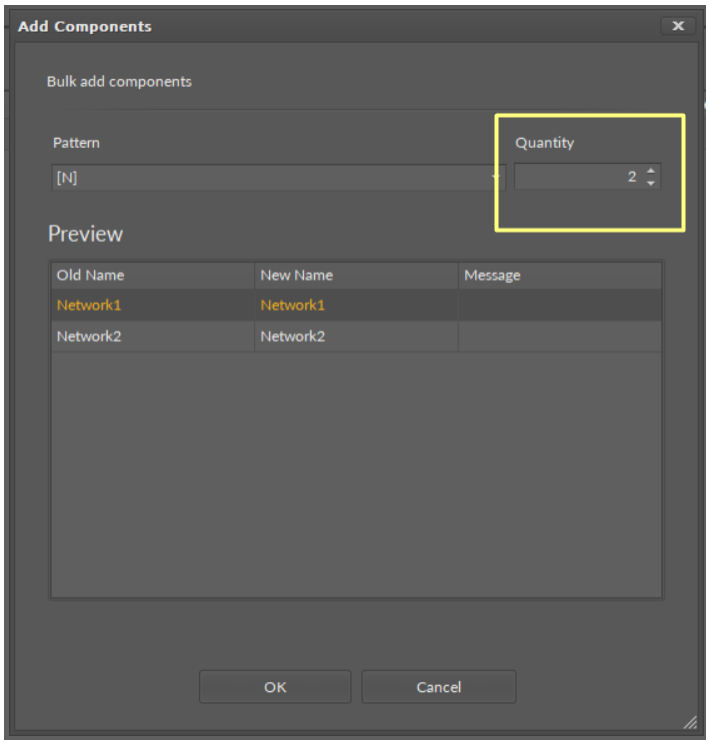



*Figure 135. The dialog window*

## Multiediting of Common Slots

The Network Manager view allows multiediting of common slots in components of the same type, for example, to enable all one-type components at once. Multiediting is available in the Object Properties window, upon selecting one-type components in the Network Manager with Ctrl or Shift keys.

## 8.4.2 Modbus

The Network Manager view is available for the Modbus component. It lists all Modbus networks configured on the device's ports. The Network Manager view shows the statuses, ports (which the network is configured on), and enabled or disabled states of the the network. Once the network, listed in the Network Manager, is double-clicked, the respective Network component is opened.

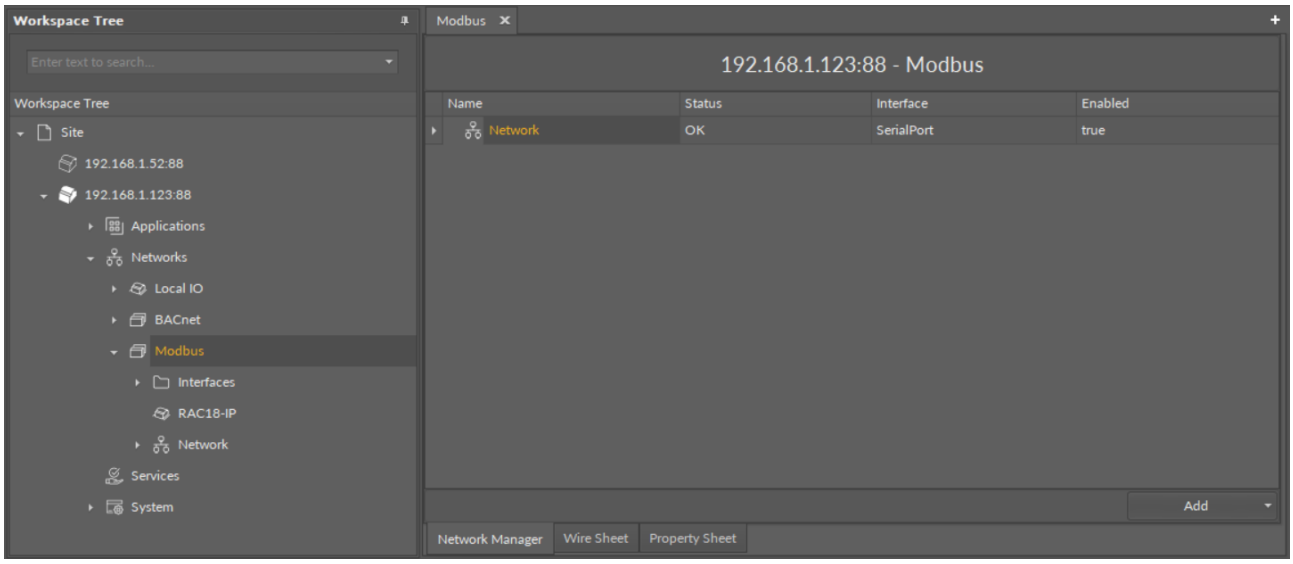



*Figure 136. The Network Manager view*

## Opening Network Manager

The Network Manager view is accessible from the context menu of the Modbus component. It is also automatically opened if the Modbus component is double-clicked in the Workspace Tree window.

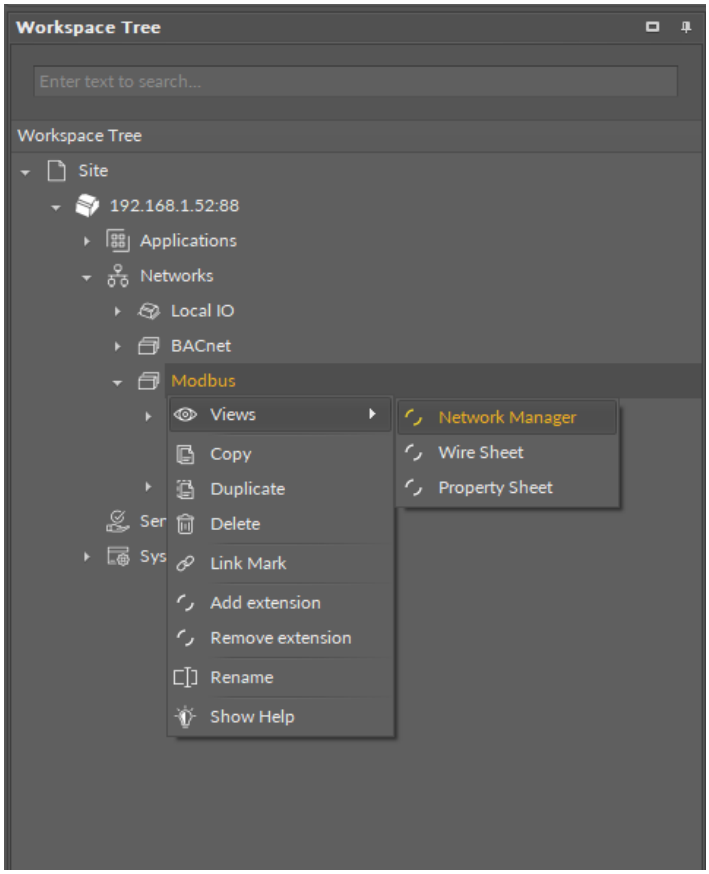



*Figure 137. Opening the Network Manager*

## Adding Modbus Networks

The networks may be added to the Network Manager twofold: dragging and dropping the Network component to the Modbus component from the Core library (in the Device

Libraries window), or using a special Add function in the Network Manager view available in the bottom right corner.

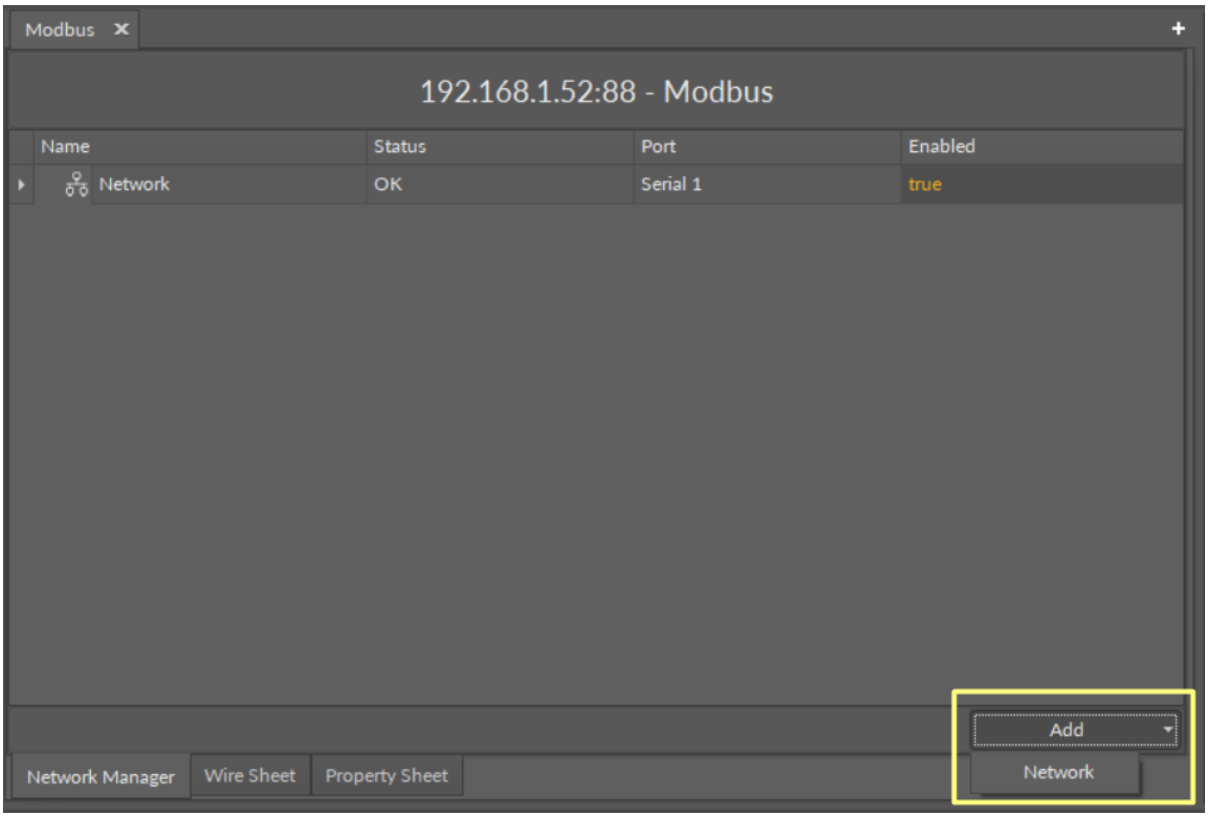



*Figure 138. Adding the Modbus network*

Using this Add button opens the dialog window, which allows to adjust the quantity of networks to be added.

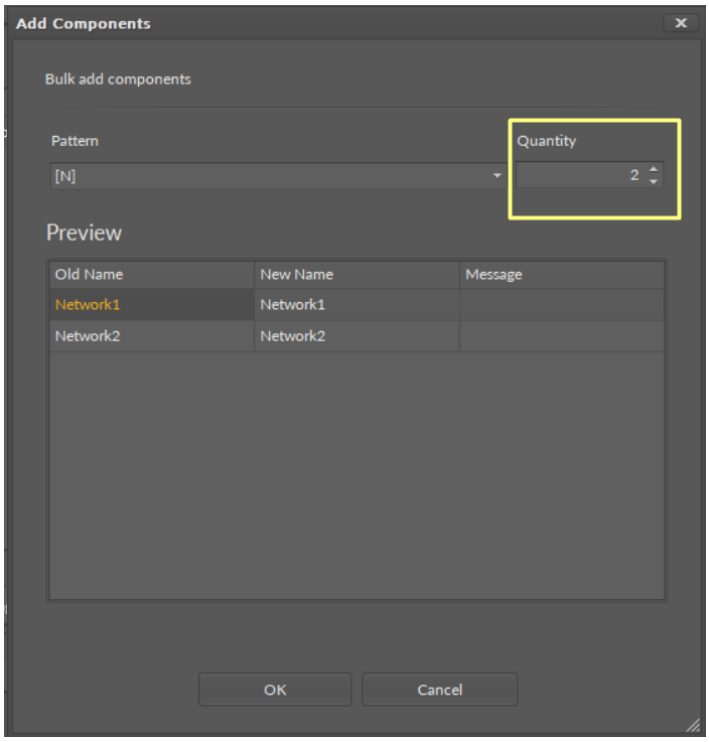



*Figure 139. The dialog window*

### Multiediting of Common Slots

The Network Manager view allows multiediting of common slots in components of the same type, for example, to enable all one-type components at once. Multiediting is available in the Object Properties window, upon selecting one-type components in the Network Manager with Ctrl or Shift keys.

## 8.5 Device Manager

### 8.5.1 BACnet

The Device Manager view is available for the BACnet Network component. It lists all devices added to the configured BACnet network. The Device Manager view shows the statuses, BACnet device names and IDs, and enabled or disabled states of the devices in the network. Once the device listed in the Device Manager is double clicked, the respective Device component is opened.

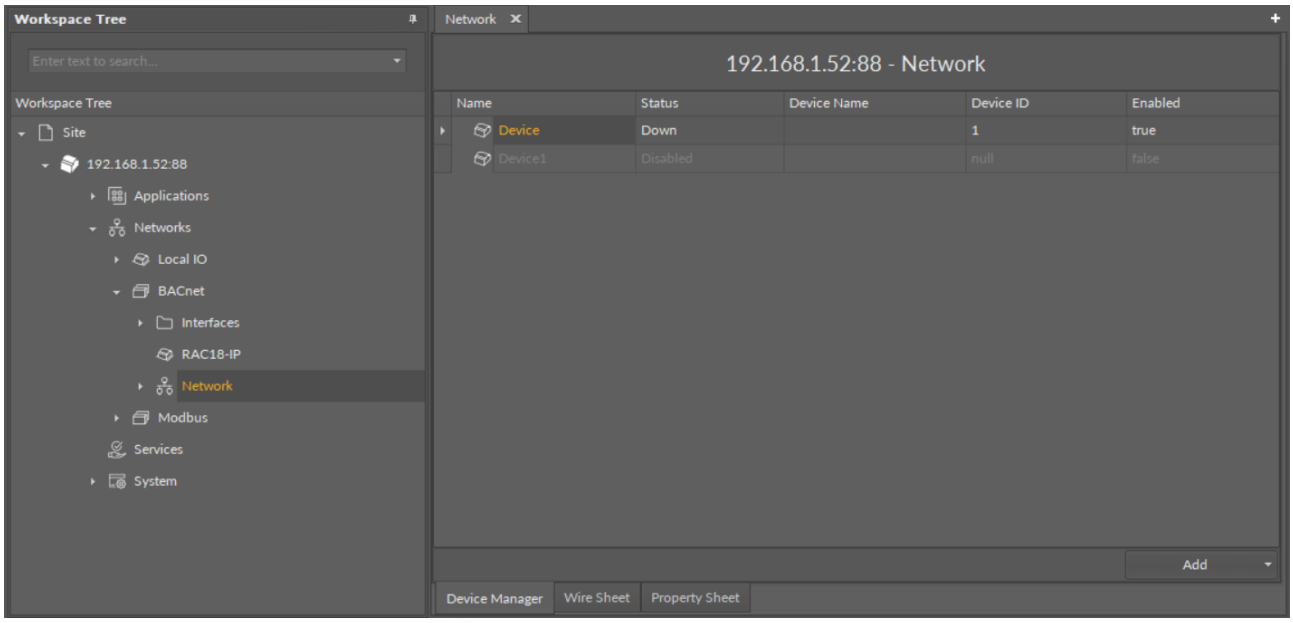



*Figure 140. The Device Manager*

### Opening Device Manager

The Device Manager view is accessible from the context menu of the Network component. It is also automatically opened if the Network component is double-clicked in the Workspace Tree window.

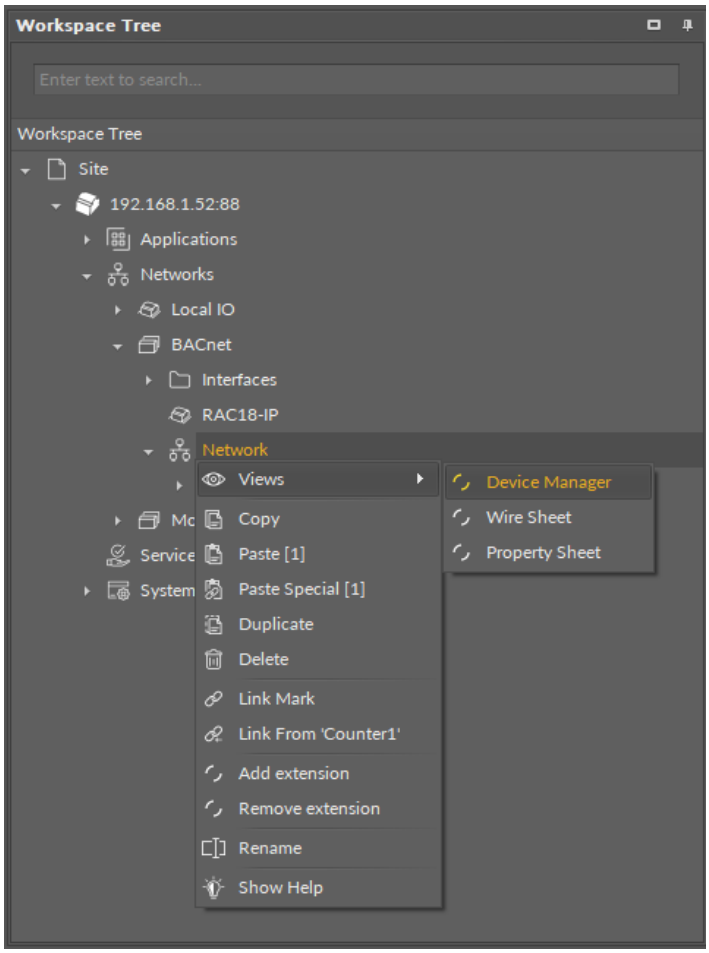



*Figure 141. Opening the Device Manager*

## Adding BACnet Devices

The devices may be added to the BACnet network twofold: dragging and dropping the Device component to the Network component from the BACnet library (in the Device Libraries window), or using a special Add function in the Device Manager view available in the bottom right corner.

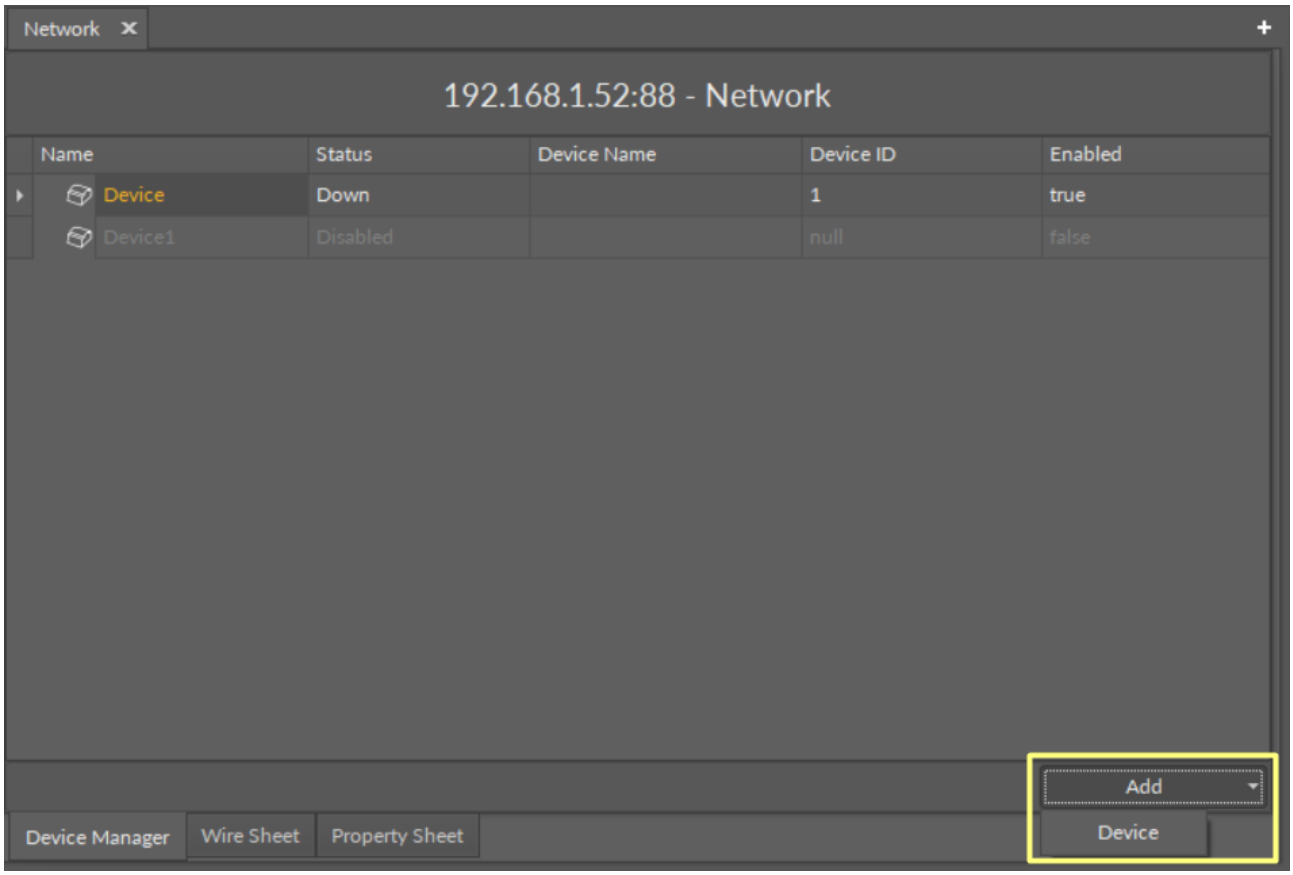



*Figure 142. The Add button in the Device Manager*

Using this Add button opens the dialog window, which allows to adjust the quantity of devices to be added.

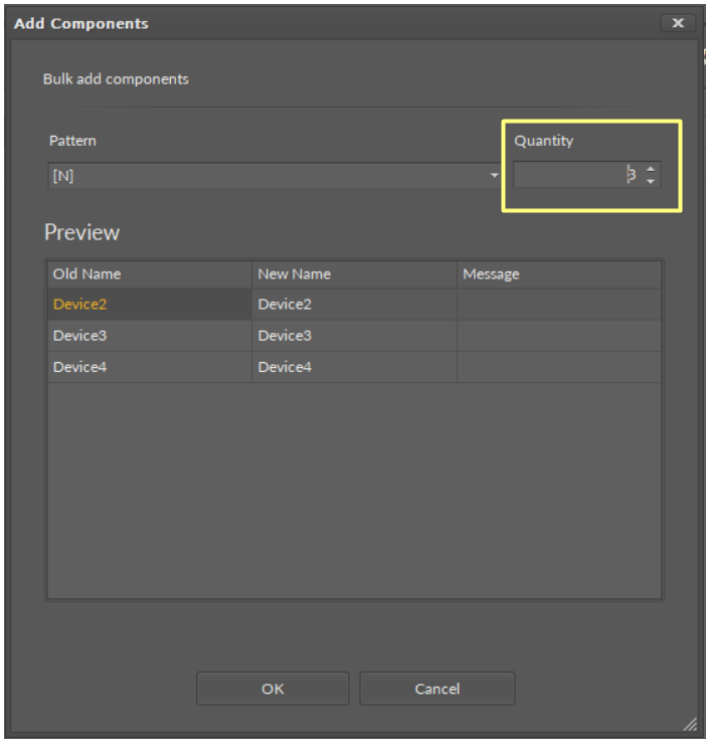



*Figure 143. The dialog window*

## Multiediting of Common Slots

The Device Manager view allows multiediting of common slots in components of the same type, for example, to enable all one-type components at once. Multiediting is available in the Object Properties window, upon selecting one-type components in the Device Manager with Ctrl or Shift keys.

## 8.5.2 Modbus

The Device Manager view is available for the Modbus Network component. It lists all devices added to the configured Modbus network. The Device Manager view shows the statuses, device addresses, and enabled or disabled states of the devices in the network. Once the device, listed in the Device Manager, is double clicked, the respective Device component is opened.

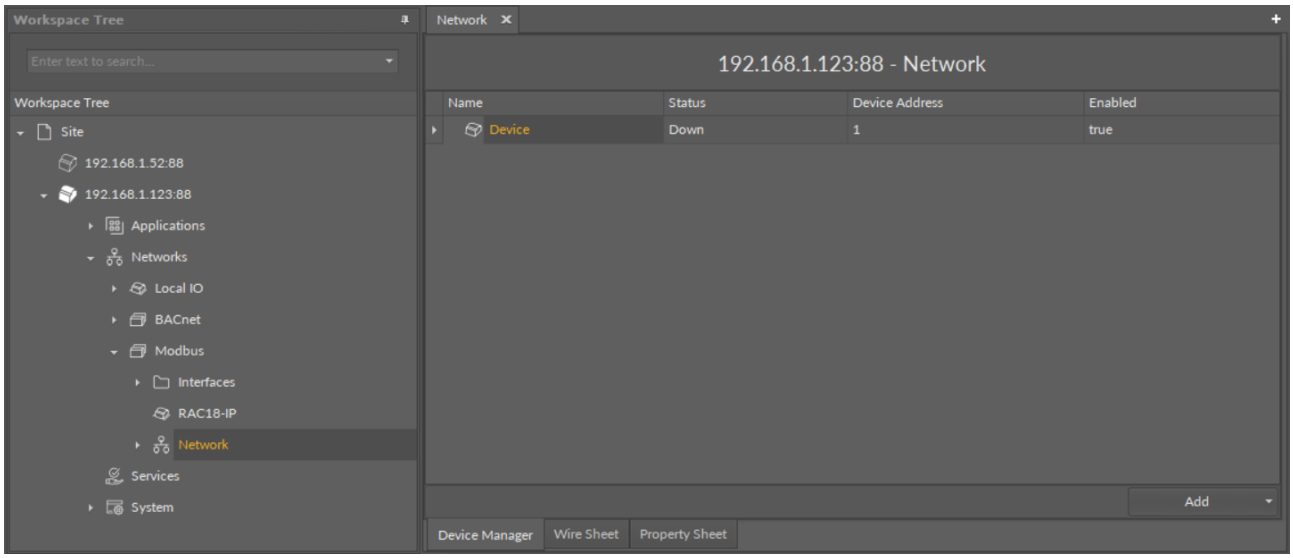



*Figure 144. The Device Manager view*

## Opening Device Manager

The Device Manager view is accessible from the context menu of the Network component. It is also automatically opened if the Network component is double-clicked in the Workspace Tree window.

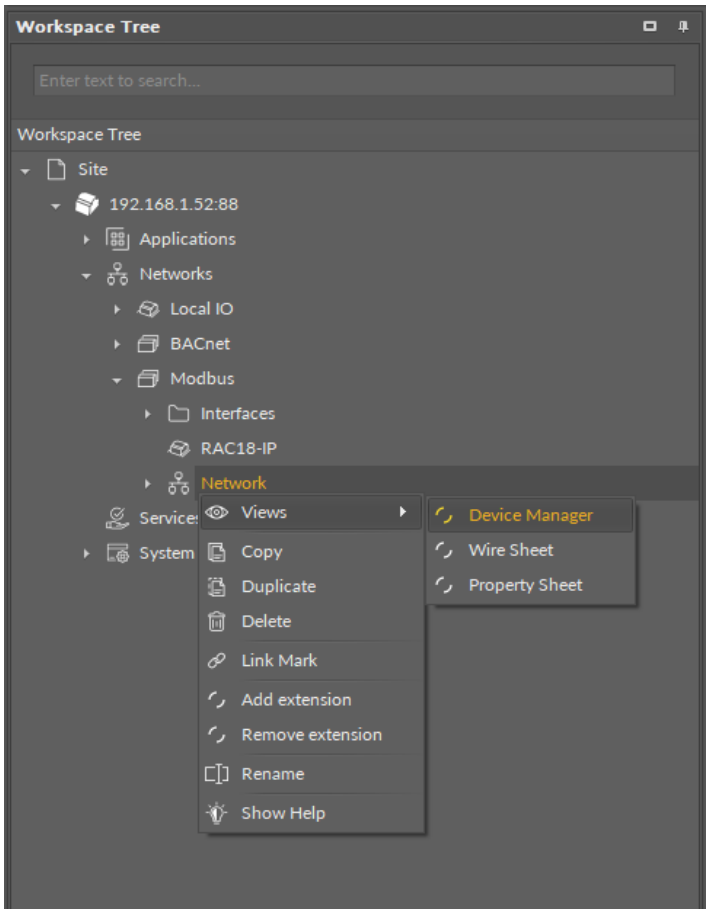



*Figure 145. Opening the Device Manager*

## Adding Modbus Devices

The devices may be added to the Modbus network twofold: dragging and dropping the Device component to the Network component from the Modbus library (in the Device Libraries window), or using a special Add function in the Device Manager view available in the bottom right corner.

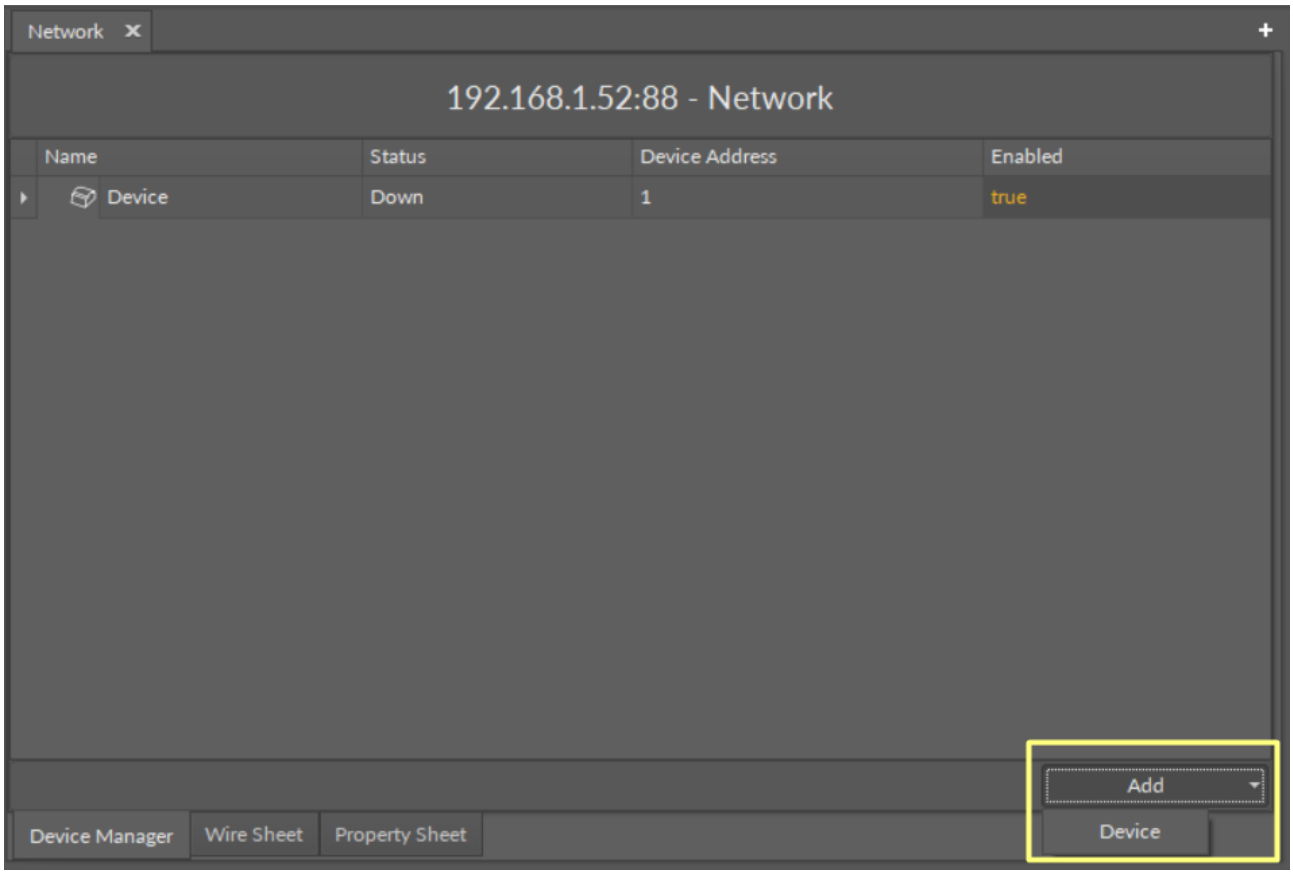



*Figure 146. Adding Modbus devices*

Using this Add button opens the dialog window, which allows to adjust the quantity of devices to be added.

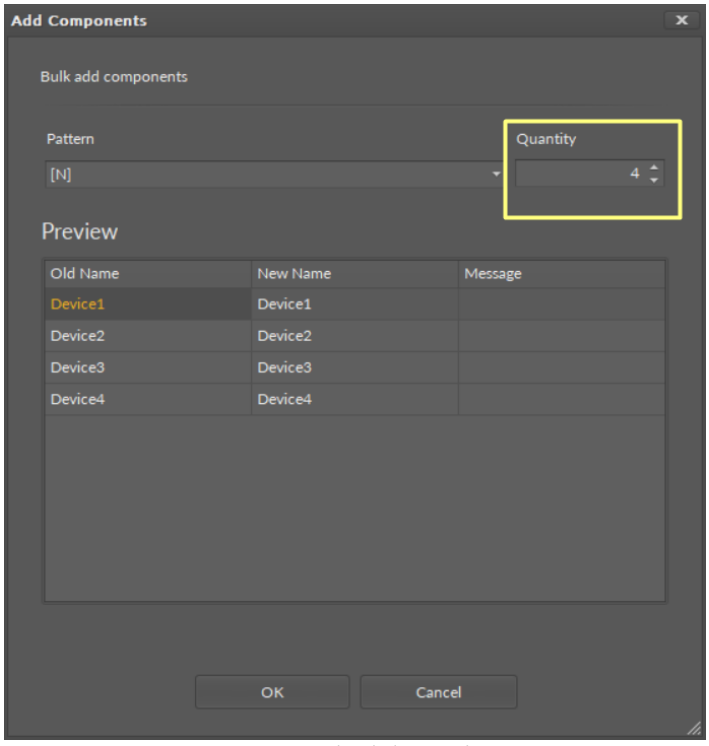



*Figure 147. The dialog window*

### Multiediting of Common Slots

The Device Manager view allows multiediting of common slots in components of the same type, for example, to enable all one-type components at once. Multiediting is available in the Object Properties window, upon selecting one-type components in the Device Manager with Ctrl or Shift keys.

## 8.6 Point Manager

### 8.6.1 LocalIO

The LocalIO Point Manager view is available for the LocalIO component. It lists all I/O points added to the LocalIO component, and shows their:

- Out slot value;
- unit (for analog values);
- status;
- number;
- enabled or disabled state.

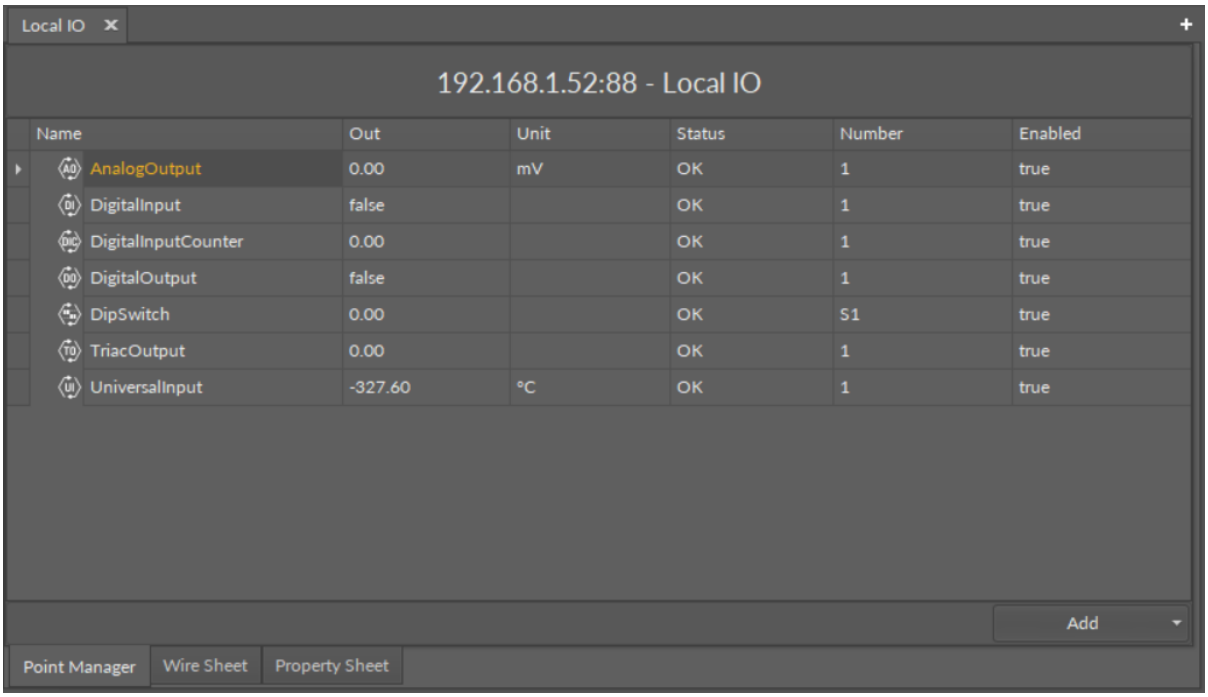



*Figure 148. The Point Manager view*

### 8.6.2 Opening Point Manager

The Point Manager view is accessible from the context menu of the LocalIO component. It is also automatically opened if the LocalIO component is double-clicked in the Workspace Tree window.

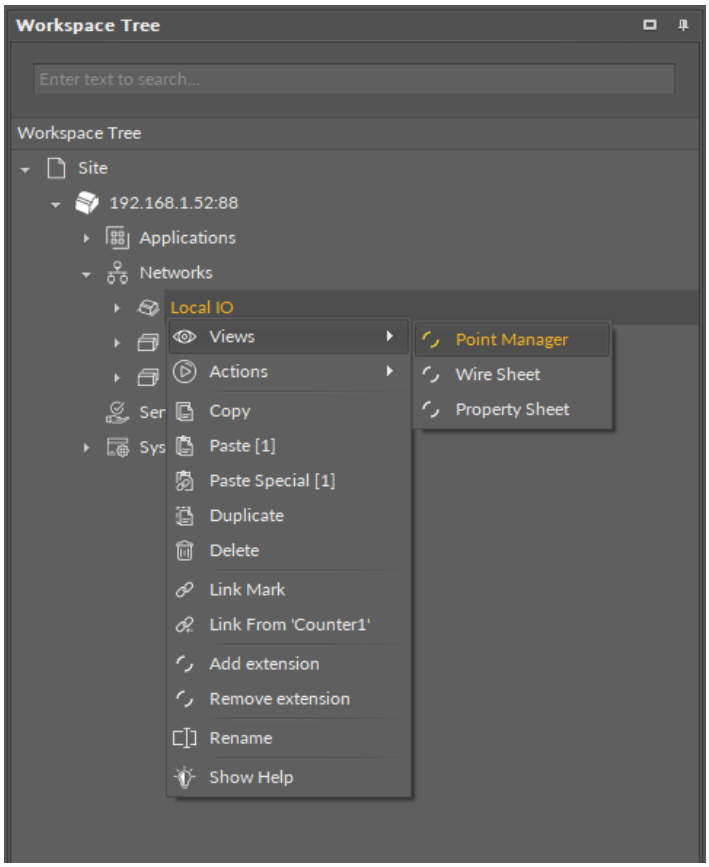



*Figure 149. Opening the Point Manager view*

### 8.6.3 Adding I/O Points

The I/O points may be added twofold: dragging and dropping the I/O points to the LocalIO component from the IO library (in the Device Libraries window), or using a special Add function in the LocalIO Point Manager view available in the bottom right corner. The Add function allows to add any of the I/O points available in the IO library.

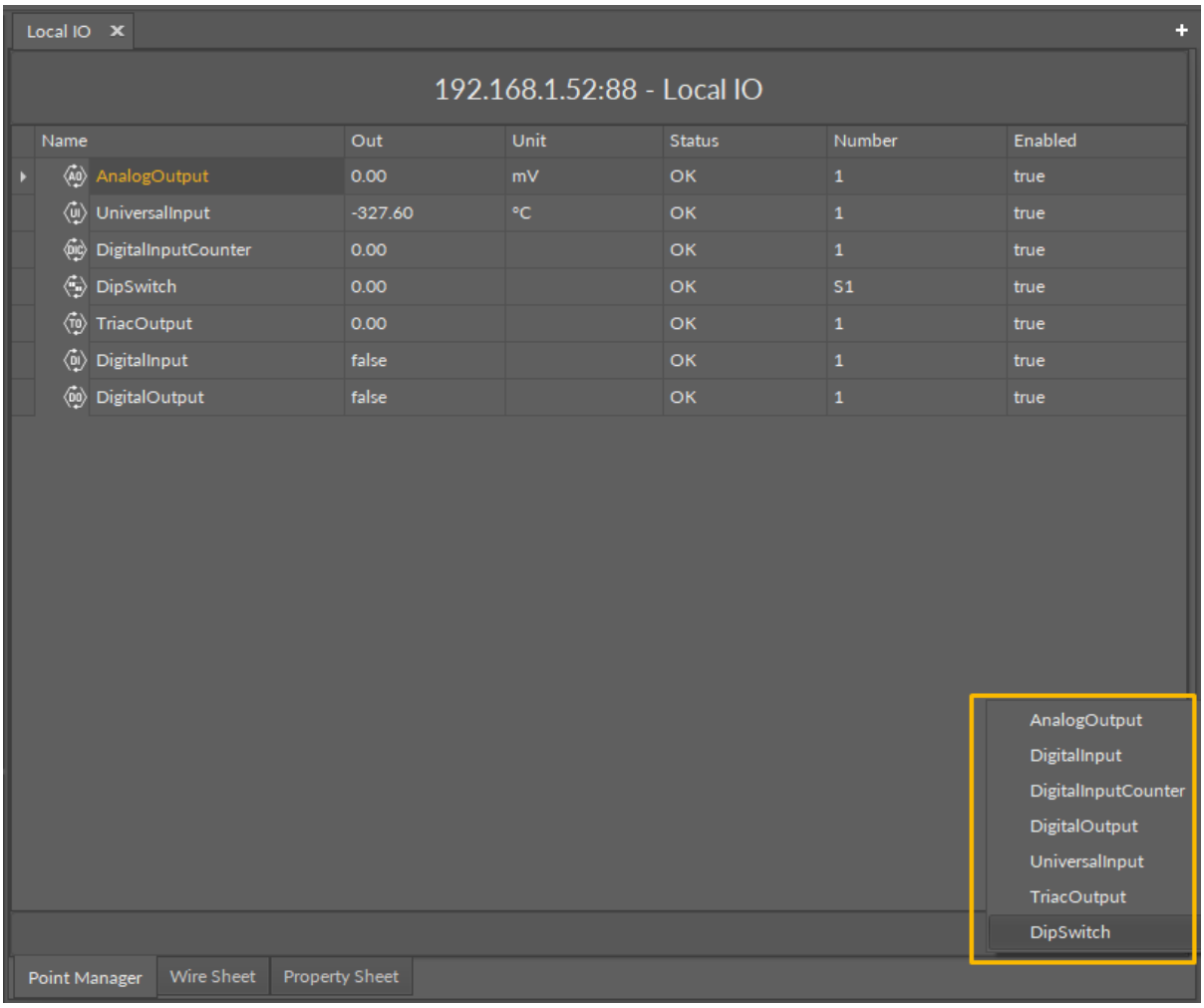



*Figure 150. The Add button in the Point Manager*

Using this Add button opens the dialog window, which allows to adjust the quantity of I/O points to be added.

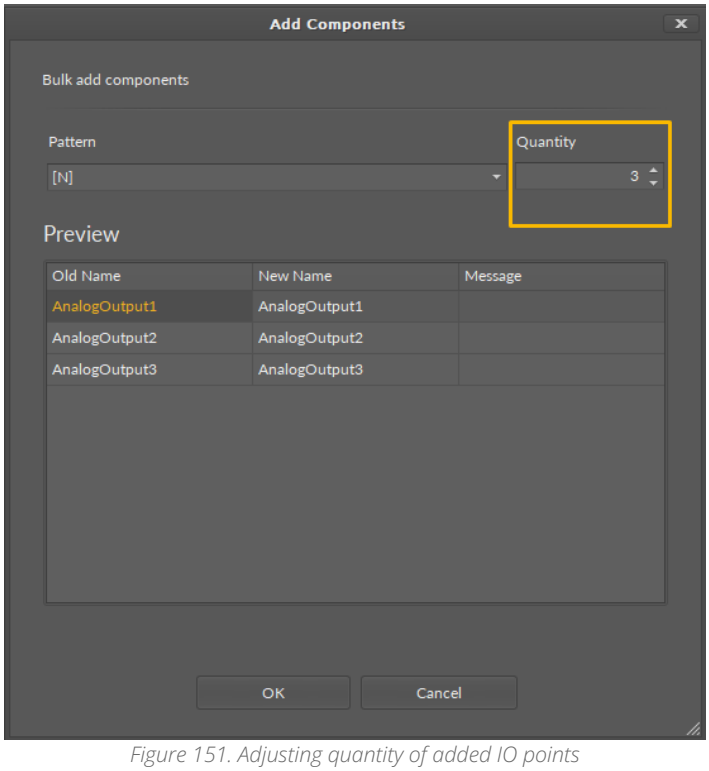



*Figure 151. Adjusting quantity of added IO points*

### 8.6.4 Multiediting of Common Slots

The Point Manager view allows multiediting of common slots in components of the same type, for example, to enable all one-type components at once. Multiediting is available in the Object Properties window, upon selecting one-type components in the Point Manager with Ctrl or Shift keys.

### 8.6.5 BACnet

The Point Manager view is available for each device added to the BACnet network. It lists all BACnet points added to the Device component, and shows their Out slot value, status, object name and ID, polling mode, and enabled or disabled state.

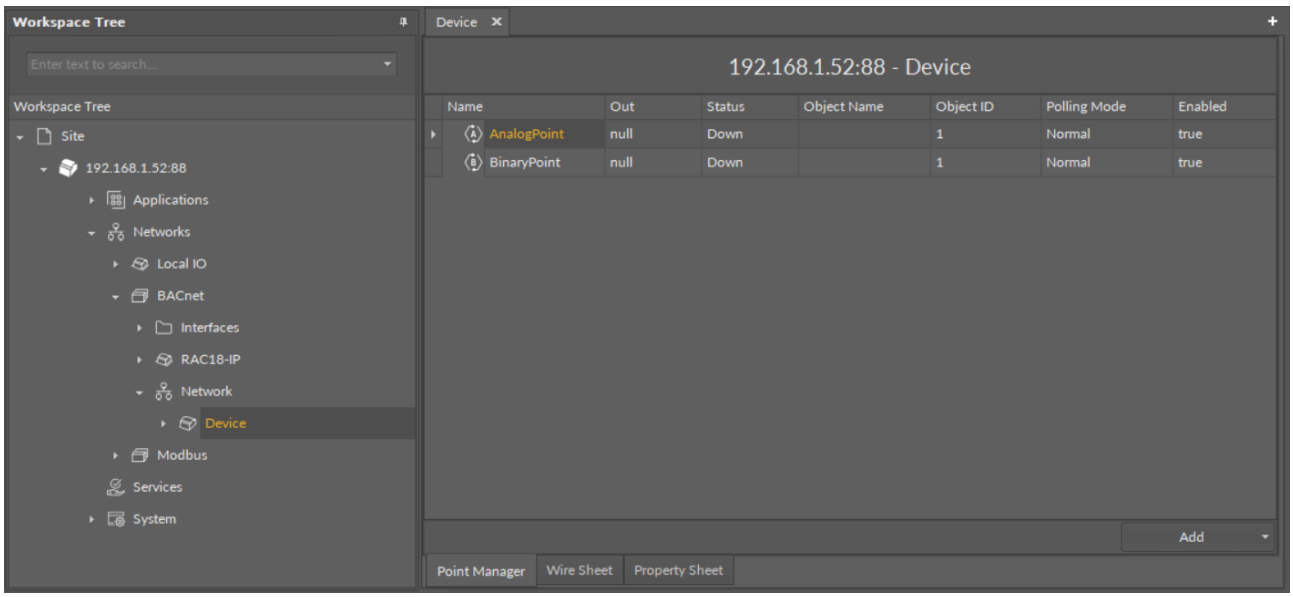



*Figure 152. The Point Manager*

### Opening the Point Manager

The Point Manager view is accessible from the context menu of the Device component. It is also automatically opened if the Device component is double-clicked in the Workspace Tree window.

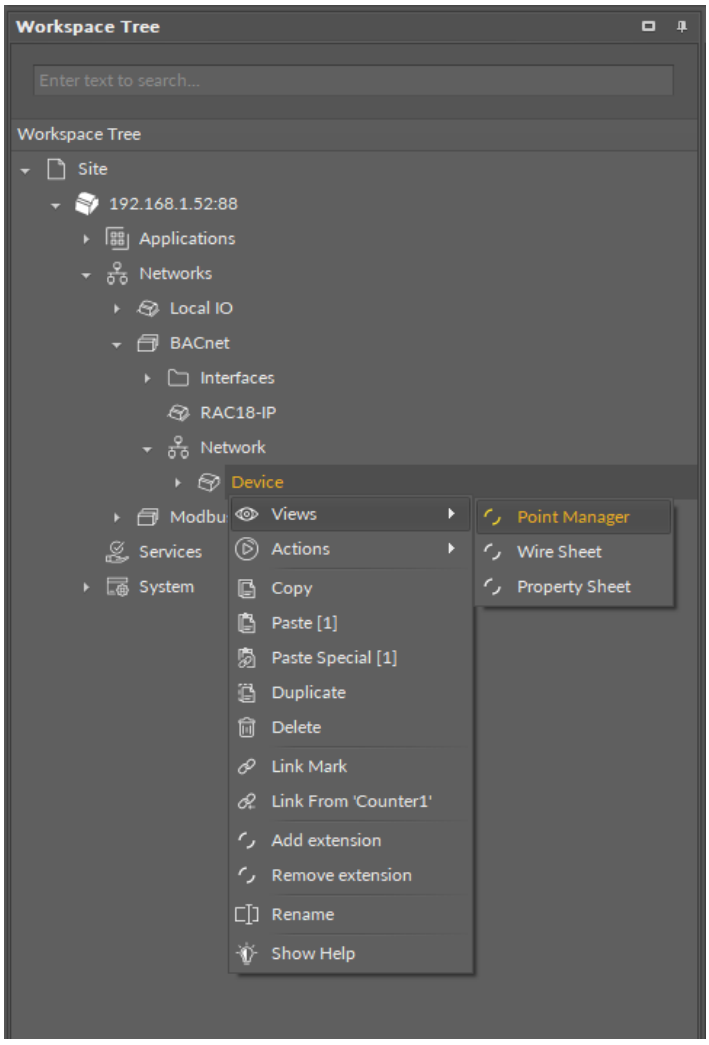



*Figure 153. Opening the Point Manager*

## Adding BACnet Points

The BACnet points may be added to the device twofold: dragging and dropping the BACnet points to the Device component from the BACnet library (in the Device Libraries window), or using a special Add function in the Point Manager view available in the bottom right corner. The Add function allows to add any of the BACnet points available in the BACnet library.

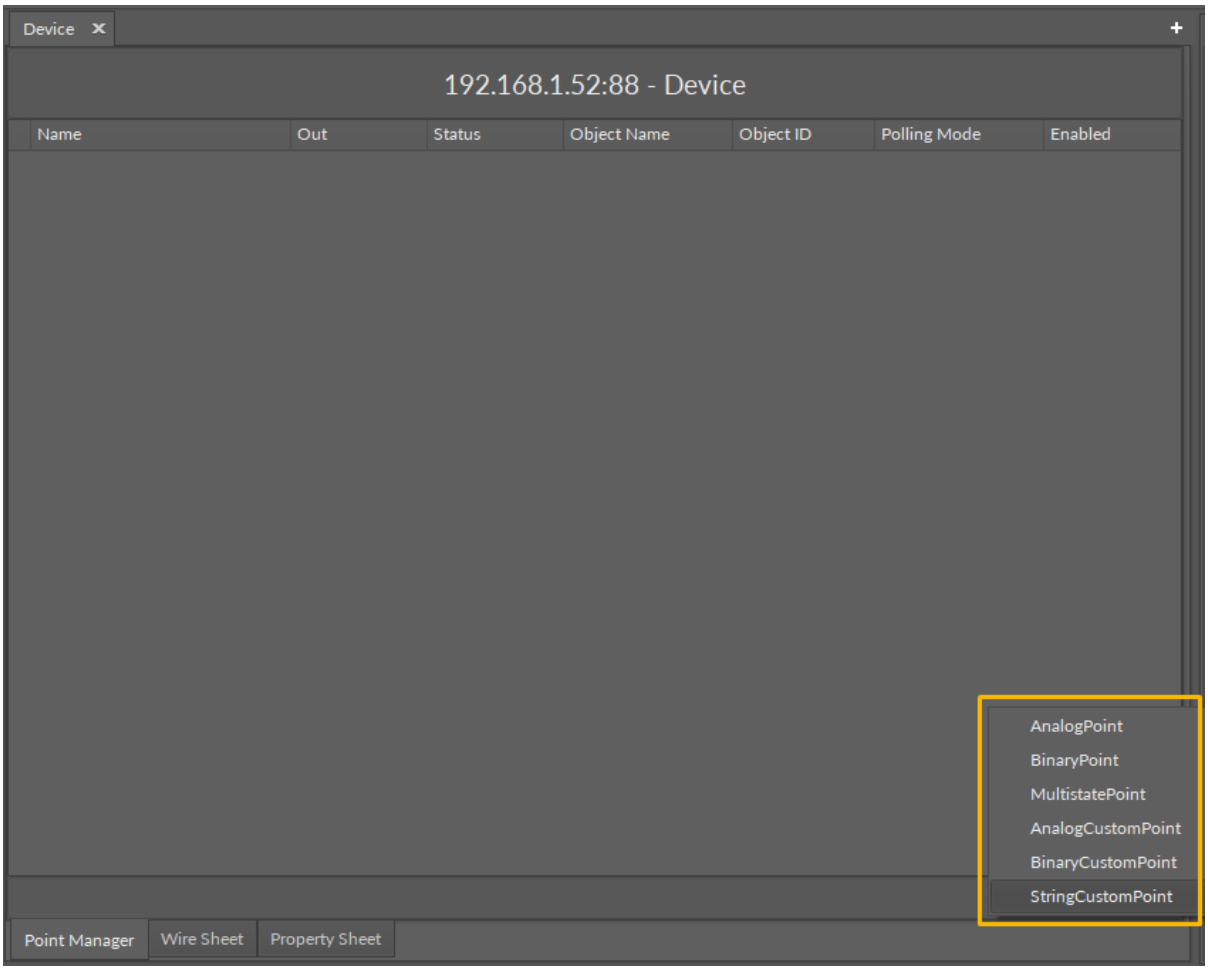



*Figure 154. The Add button*

Using this Add button opens the dialog window, which allows to adjust the quantity of BACnet Points to be added.

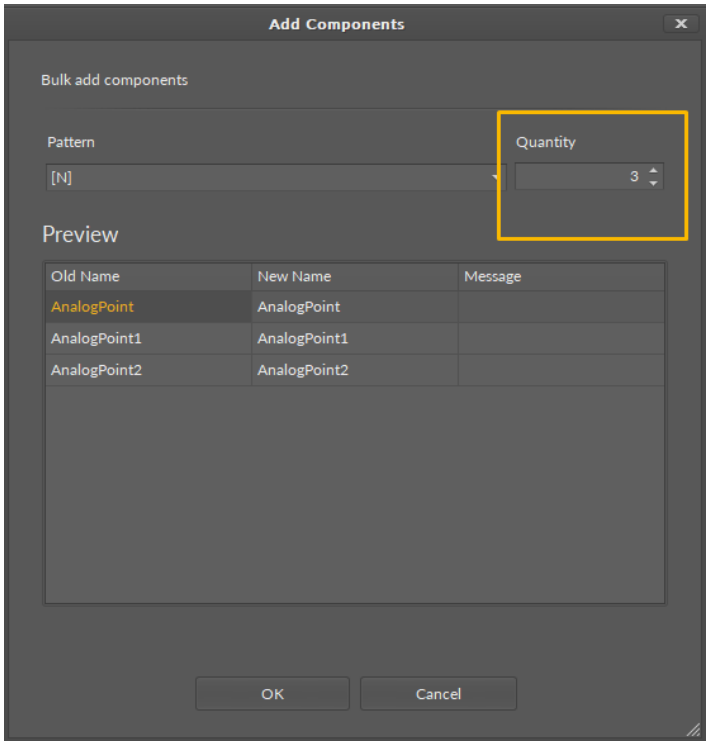



*Figure 155. The dialog window*

## Multiediting of Common Slots

The Point Manager view allows multiediting of common slots in components of the same type, for example, to enable all one-type components at once. Multiediting is available in the Object Properties window, upon selecting one-type components in the Point Manager with Ctrl or Shift keys.

## 8.6.6 Modbus

The Point Manager view is available for each device added to the Modbus network. It lists all Modbus Points added to the Device component, and shows their Out slot value, status, address, polling mode, and enabled or disabled state.

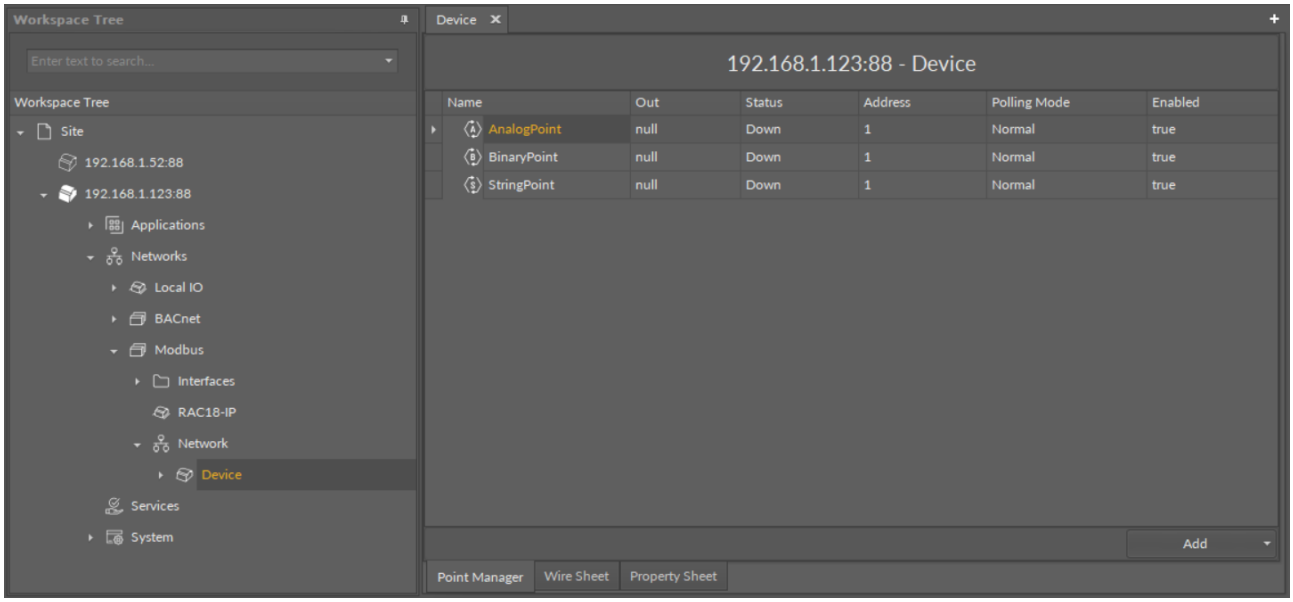



*Figure 156. The Point Manager*

## Opening Point Manager

The Point Manager view is accessible from the context menu of the Device component. It is also automatically opened if the Device component is double-clicked in the Workspace Tree window.

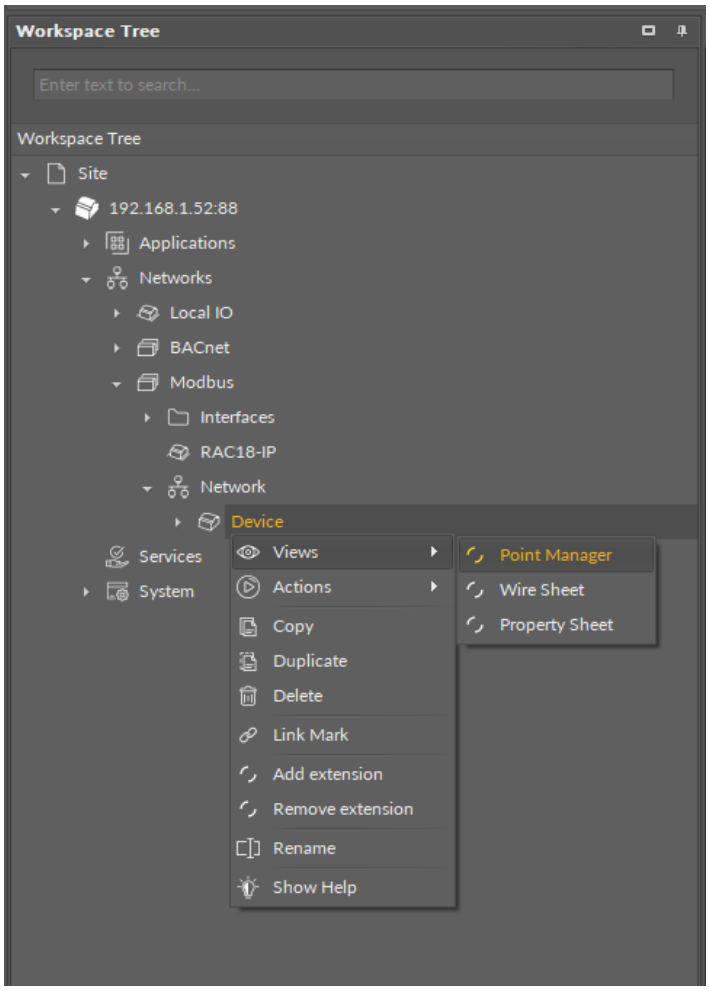



*Figure 157. Opening the Point Manager*

## Adding Modbus Points

The Modbus points may be added to the device twofold: dragging and dropping the Modbus points to the Device component from the Modbus library (in the Device Libraries window), or using a special Add function in the Point Manager view available in the bottom right corner. The Add function allows to add any of the Modbus points available in the Modbus library.

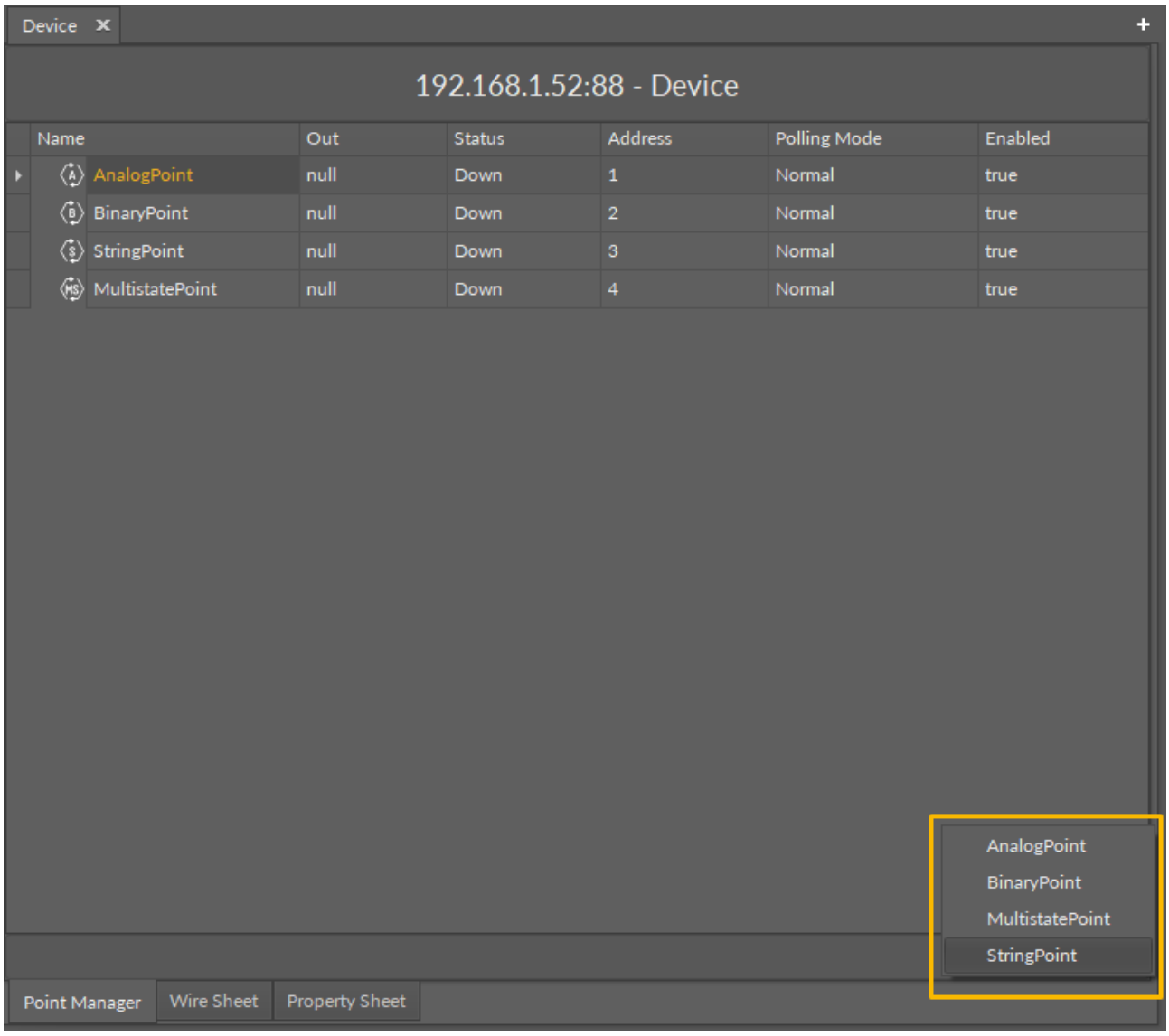



*Figure 158. The Add button*

Using this Add button opens the dialog window, which allows to adjust the quantity of Modbus Points to be added.

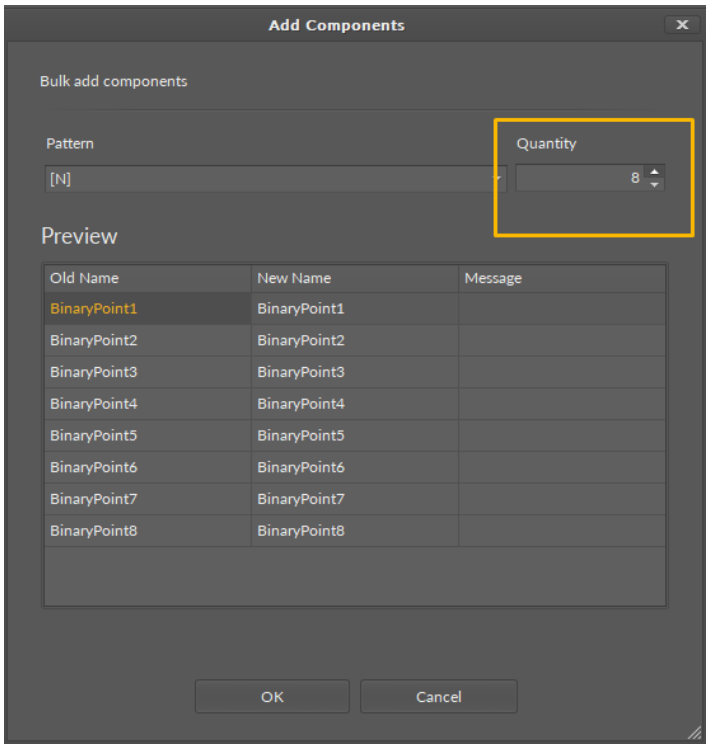



*Figure 159. The dialog window*

### Multiediting of Common Slots

The Point Manager view allows multiediting of common slots in components of the same type, for example, to enable all one-type components at once. Multiediting is available in the Object Properties window, upon selecting one-type components in the Point Manager with Ctrl or Shift keys.

## 8.7 Log Viewer

The nano Edge Engine Log Viewer provides logs for diagnosing and analyzing the work of the device driven by the nano Edge Engine.

Logs are categorized by their source of occurrence:

- OS.Firmware;
- OS.SDK;
- Library.BACnet;
- Library.Core;
- Library.IO;
- Library.Modbus;
- Library.Schedules;
- OS.Core;
- OS.iFnet;
- OS.Storage.

The logs levels are defined in the Logs component. The Log Viewer is the default view of the Logs component.

## 8.7.1 Opening the Log Viewer

The Log Viewer is accessible by double-clicking the Logs component in the System container in the Workspace Tree window. Alternatively, it can be opened from the context menu of the Logs component, in the Views option.

Upon opening the Log Viewer shows the logs from the current log file, and updates it while running.

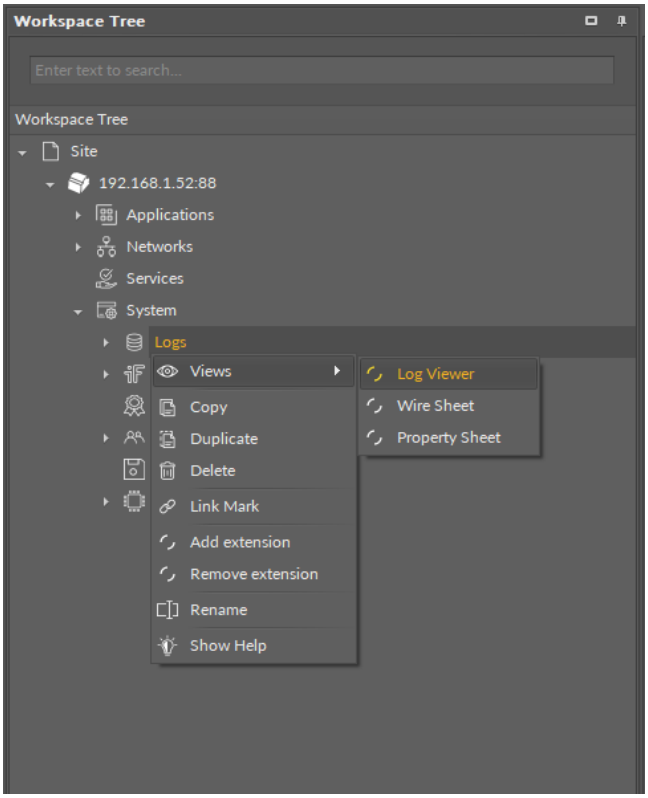



*Figure 160. Accessing Log Viewer*

## 8.7.2 Using the Log Viewer

### Filtering

Once the Log Viewer is opened, it displays logs according to the configuration of the Logs component. The logs are categorized, and can be filtered by dates. The filtering option is available in the bottom left corner of the window.

Filtering can be time consuming, depending on the selected filtering option, however, it does not have any impact on the work of the iC Tool.

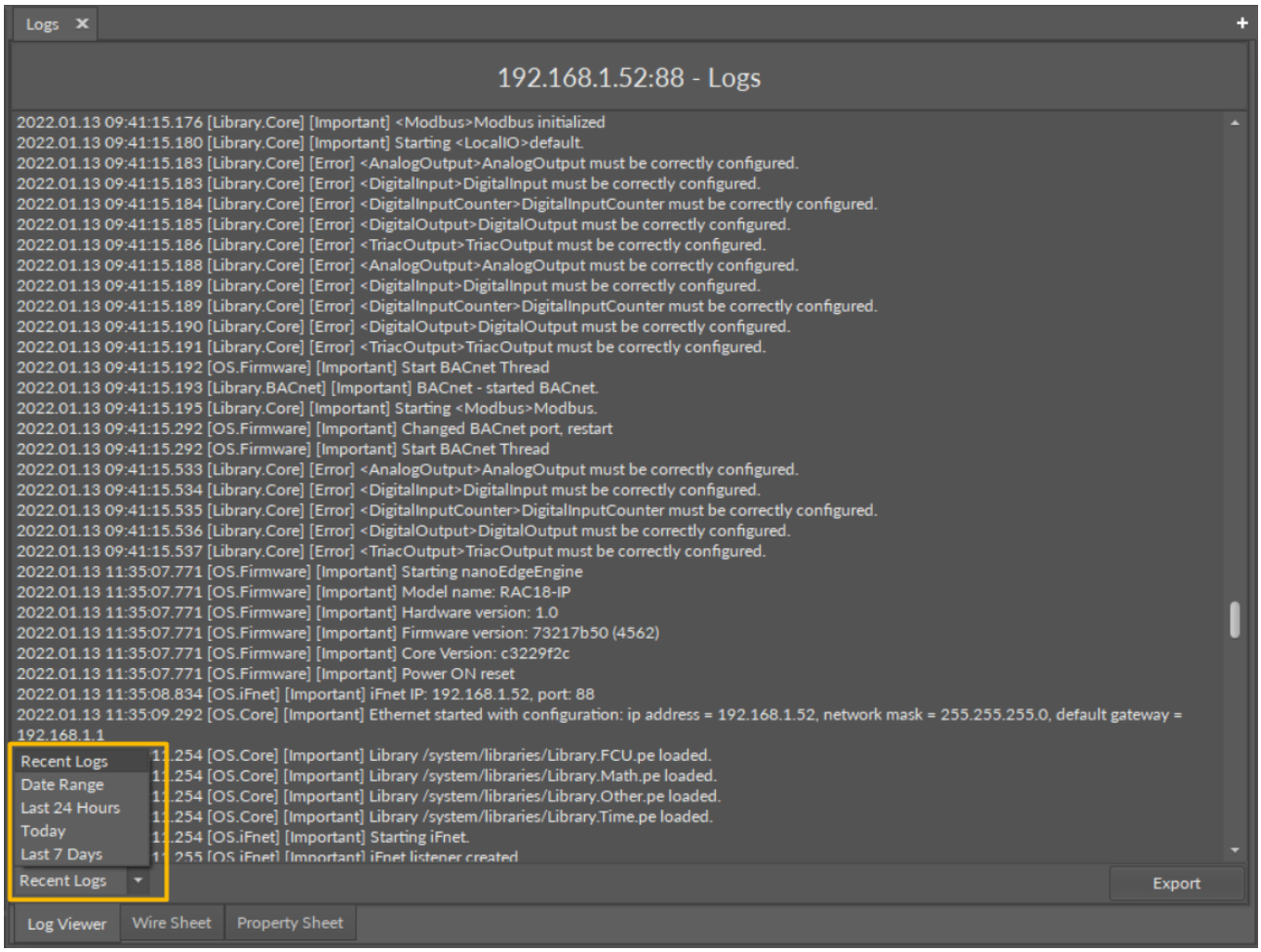



*Figure 161. Filtering*

## Exporting

The contents of the Log Viewer can be exported to a chosen location on the hard drive. The Export options is available in the bottom right corner of the window. The logs file is exported as a .txt file, which name includes the date and hour of the export.

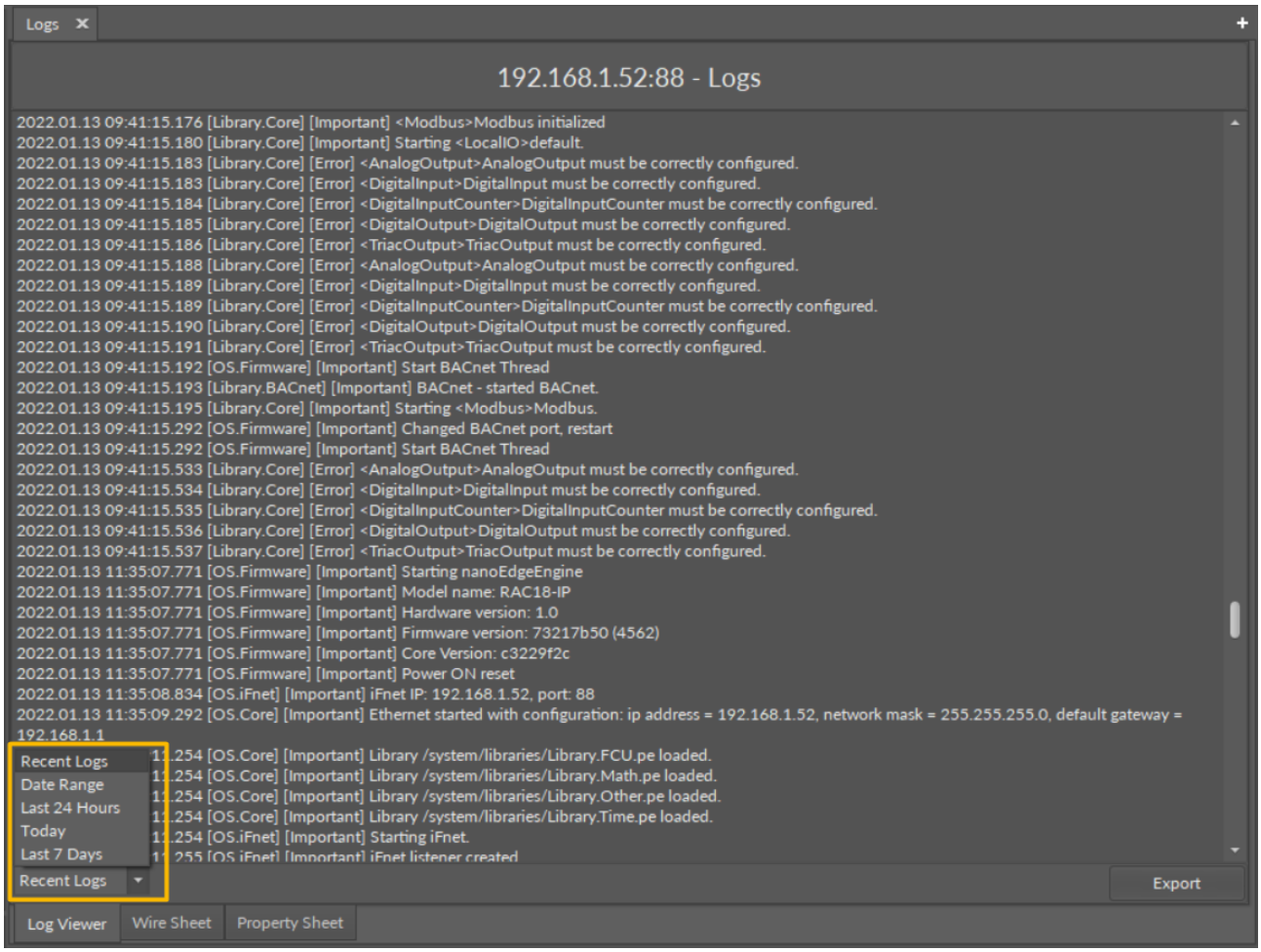
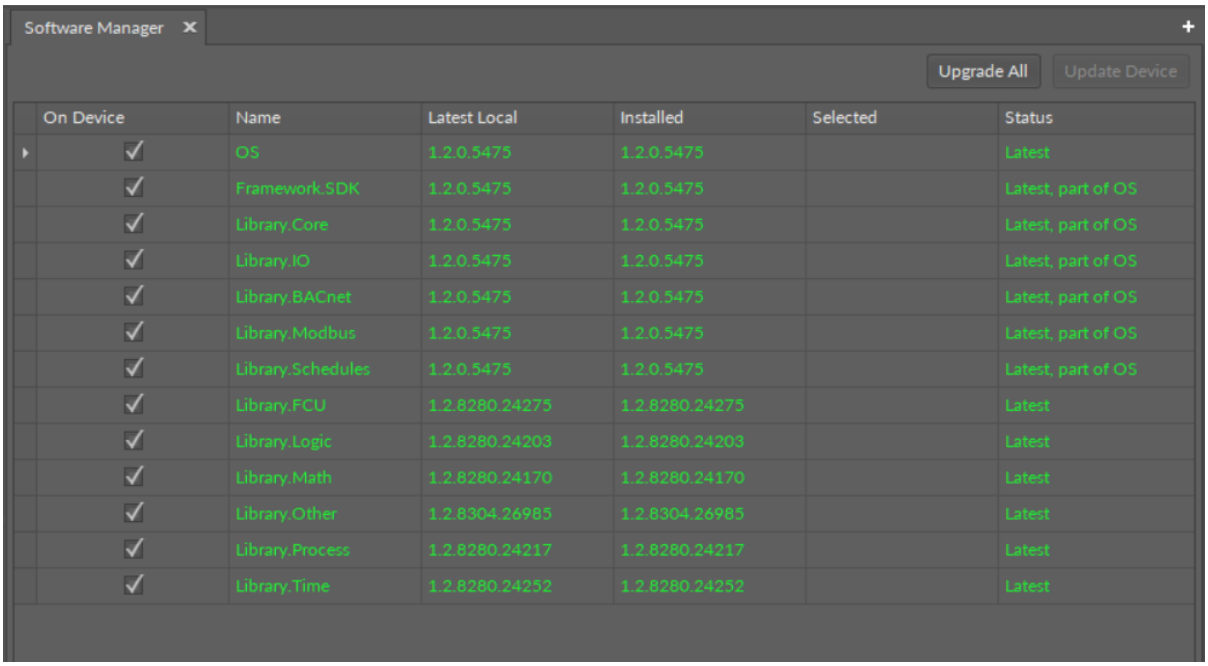



*Figure 162. Exporting*

## 8.8 Software Manager

The Software Manager is a tool allowing the management of the device operating system and its libraries. The Software Manager allows installing, uninstalling, and changing versions of the OS and libraries.

The Software Manager lists all elements (OS, libraries) that are available locally in the iC Tool to be installed on the device. The list of elements, visible in the Software Manager table, corresponds with a physical content of the nano EDGE ENGINE folder in the main iC Tool's location: \home\nanoEdgeEngine. Proper placing of the user's own libraries in this location allows to add them to the Software Manager.




| On Device | Name | Latest Local | Installed | Selected | Status |
|---|---|---|---|---|---|
| ☑ | OS | 1.2.0.5475 | 1.2.0.5475 | | Latest |
| ☑ | Framework.SDK | 1.2.0.5475 | 1.2.0.5475 | | Latest, part of OS |
| ☑ | Library.Core | 1.2.0.5475 | 1.2.0.5475 | | Latest, part of OS |
| ☑ | Library.IO | 1.2.0.5475 | 1.2.0.5475 | | Latest, part of OS |
| ☑ | Library.BACnet | 1.2.0.5475 | 1.2.0.5475 | | Latest, part of OS |
| ☑ | Library.Modbus | 1.2.0.5475 | 1.2.0.5475 | | Latest, part of OS |
| ☑ | Library.Schedules | 1.2.0.5475 | 1.2.0.5475 | | Latest, part of OS |
| ☑ | Library.FCU | 1.2.8280.24275 | 1.2.8280.24275 | | Latest |
| ☑ | Library.Logic | 1.2.8280.24203 | 1.2.8280.24203 | | Latest |
| ☑ | Library.Math | 1.2.8280.24170 | 1.2.8280.24170 | | Latest |
| ☑ | Library.Other | 1.2.8304.26985 | 1.2.8304.26985 | | Latest |
| ☑ | Library.Process | 1.2.8280.24217 | 1.2.8280.24217 | | Latest |
| ☑ | Library.Time | 1.2.8280.24252 | 1.2.8280.24252 | | Latest |

*Figure 163. The Software Manager view*

**Warning!**

Every time the OS or libraries are edited in the device, it requires a restart of the device.

### 8.8.1 Opening the Software Manager

Once the iC Tool connects to the device, the Software Manager view is available from the Objects Properties window (blue button) or from the context menu of the device.

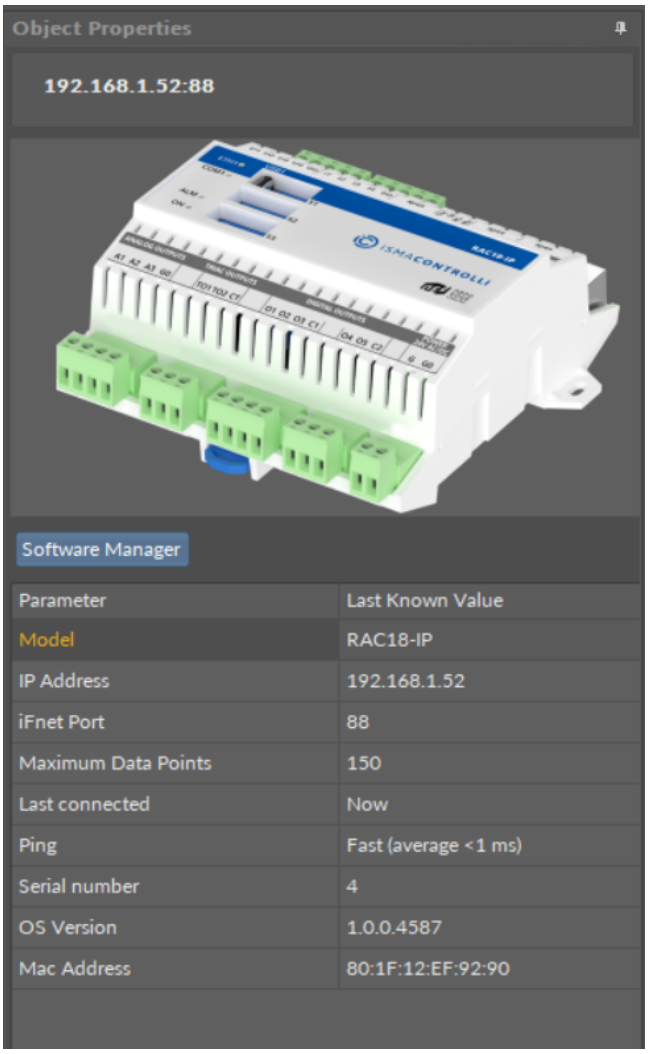



*Figure 164. The Software Manager button*

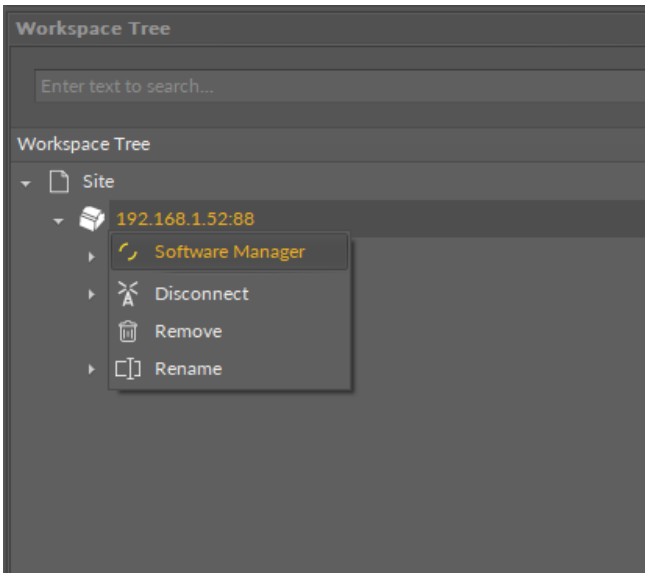



*Figure 165. The Software Manager in the device's context menu*

## 8.8.2 Using the Software Manager

The Software Manager view lists the OS and libraries available locally in the iC Tool. The view highlights each row (OS or libraries) according to its status:

- **green:** the element is up to date, and requires no action;
- **orange:** the element is out of date, and can be updated;
- **blue:** the action is about to be taken on the element.

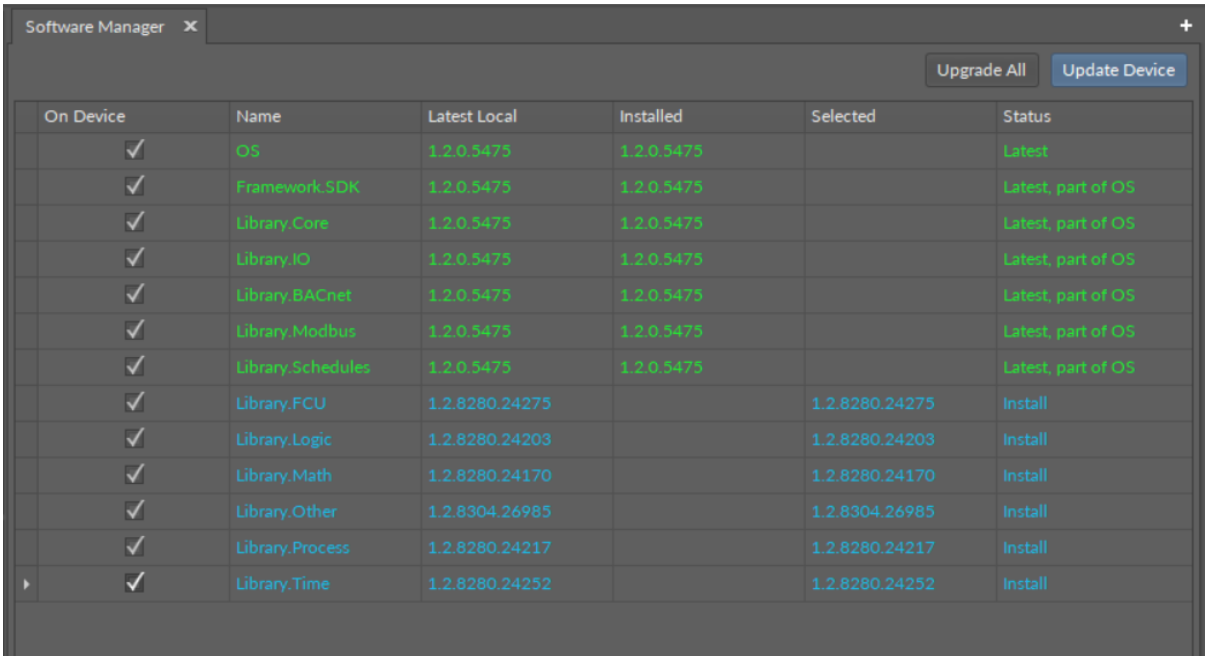



*Figure 166. Updating*

The Software Manager table contains the following columns:

- **On Device:** indicates, whether a given element is already installed on the device.
- **Name:** shows the name of the element.
- **Latest Local:** shows the latest version available locally to be installed on the device.
- **Installed:** shows the version of the element installed on the device.
- **Selected:** opens a dropdown list with all versions available locally for a selected element.
- **Status:** indicates, which action is to be performed on the element, once a specific version has been selected in the Action column.
    - Available information: Latest, Out of Date, Upgrade, Downgrade, Install, Uninstall, none (the selected version is the same as the one installed on the device).

In order to upgrade or downgrade the selected element, choose the desired version of the element in the Selected column, and press the Update Device option (highlighted in blue in the right upper corner of the Software Manager). This option executes all actions indicated in the Status column.

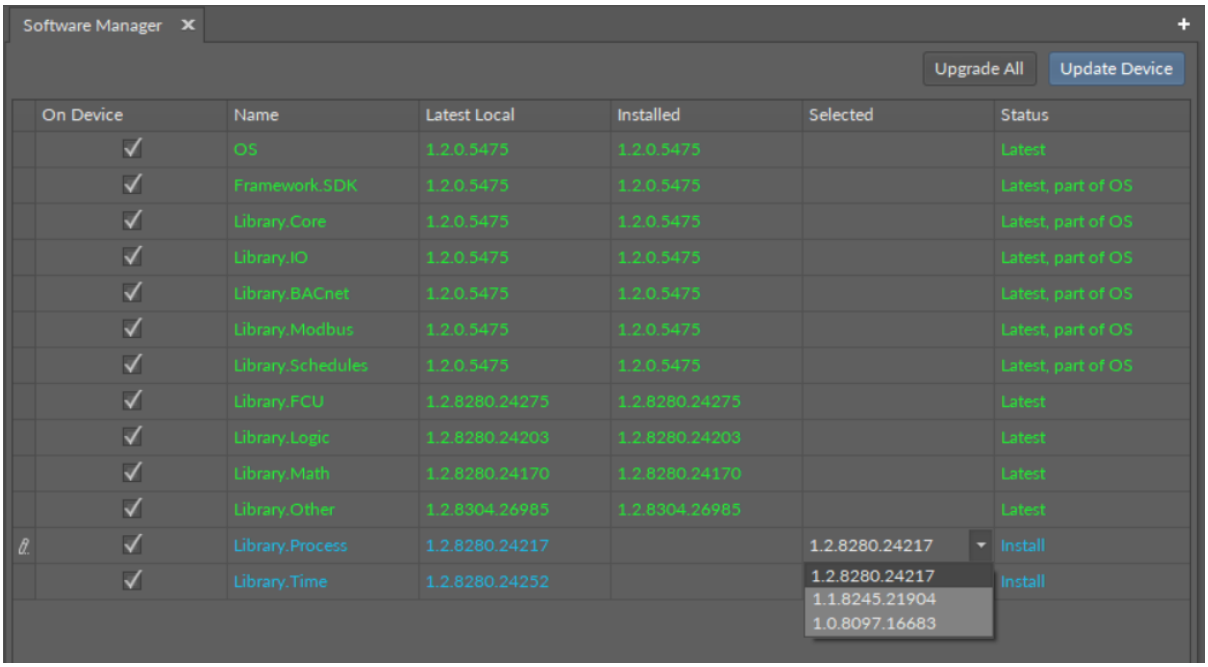



*Figure 167. Selecting a library's version*

Unless the user intends to manually select the versions to be installed, there is also the option to automatically select all newest versions for all out of date elements using the Upgrade All button.

**Worth to Notice:**

Depending if the the version indicated in the Selected column is higher or lower than the version already installed on the device, the Status column shows either the Upgrade or Downgrade option.

Regardless of the fact, whether the user intends to add or remove the library, available in the Software Manager, each operation requires performing three steps:

- check the box in the On Device column to install the library, or uncheck it to uninstall the library;
- provided the library is to be installed on the device, check its preferred version in the Selected column–by default, the newest version available locally is indicated to be installed;
- once selection of all libraries to be installed or uninstalled on the device is complete, hit the Update Device command.

**Warning!**

The OS cannot be removed from the device; it is preinstalled on the device's SD card, and the only operations, which can be performed on this element, are upgrading or downgrading it.

**Warning!**

If the OS is upgraded from the 1.0/1.1 versions to 1.2 version, the Applications libraries (FCU, Logic, Math, Process, Time, and Other) are not loaded. It is required to install them manually.

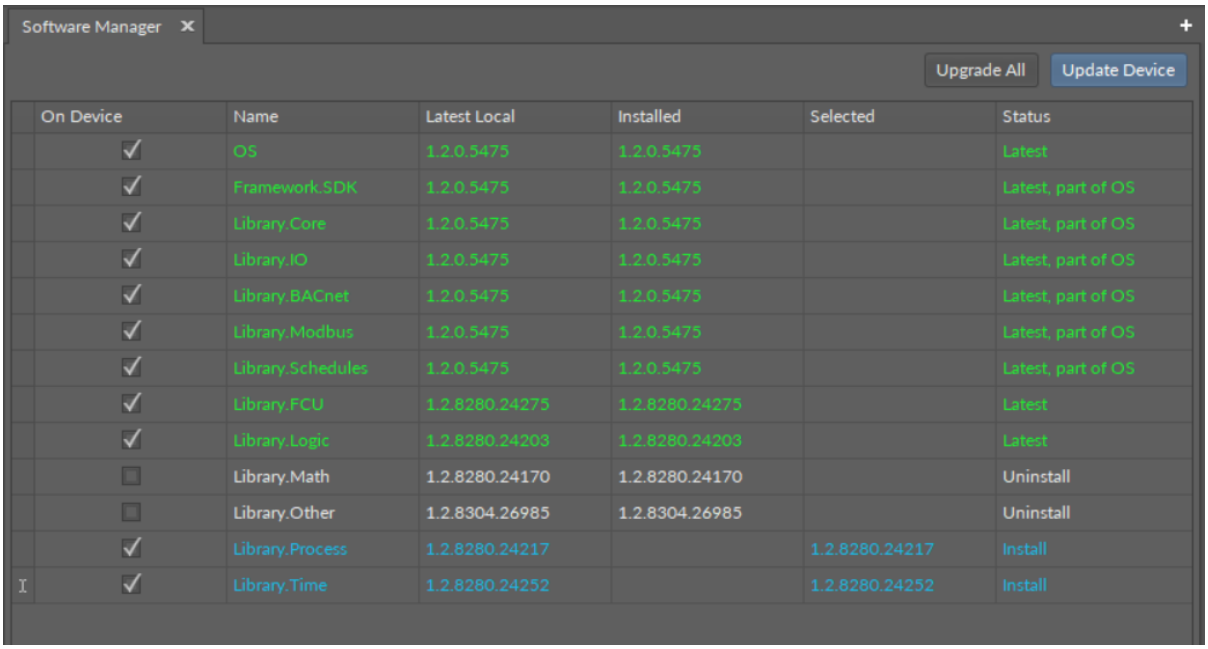



*Figure 168. Adding or removing libraries*

## 8.9 Backup Manager

The Backup Manager allows to perform and restore backups. The view is available in the Backups component in the System container.

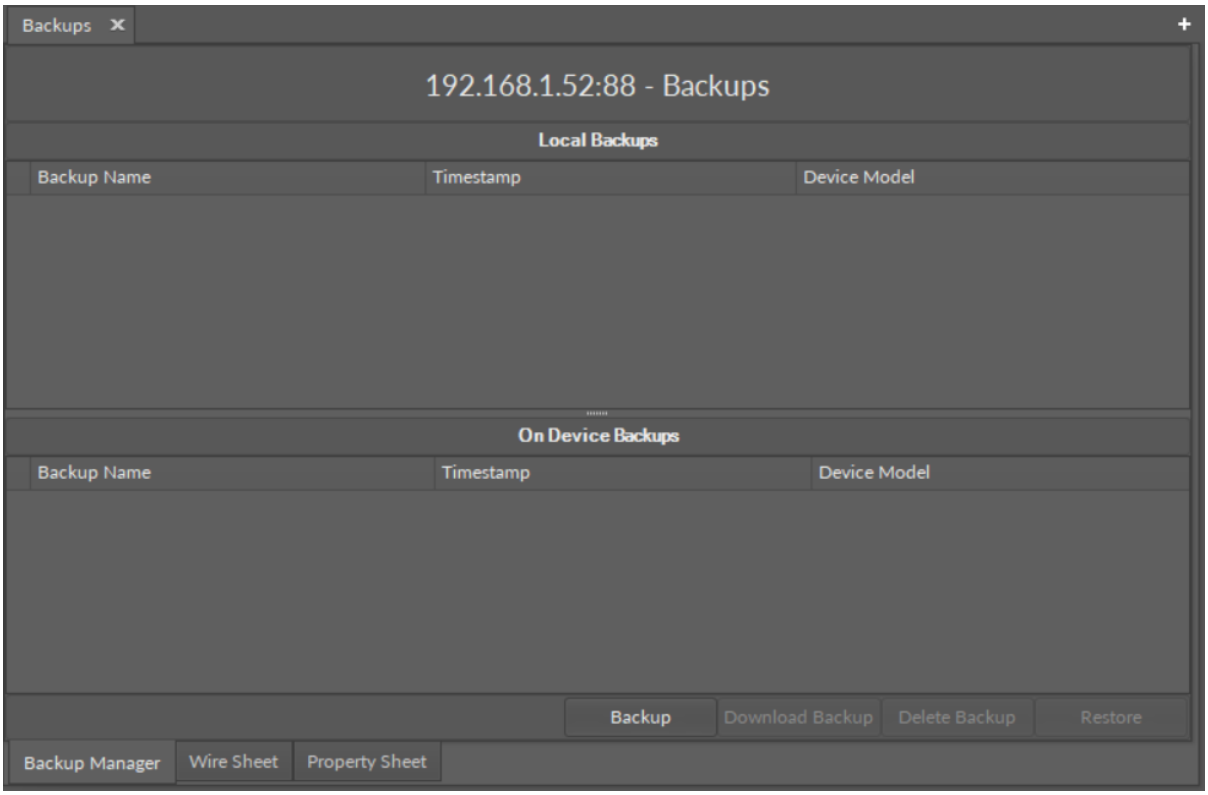



*Figure 169. The Backup Manager view*

The following options are available in the Backup Manager:

- **Backup:** creates a backup and saves it on the device; each next backup overwrites the last backup unless saved locally with the Download Backup option;

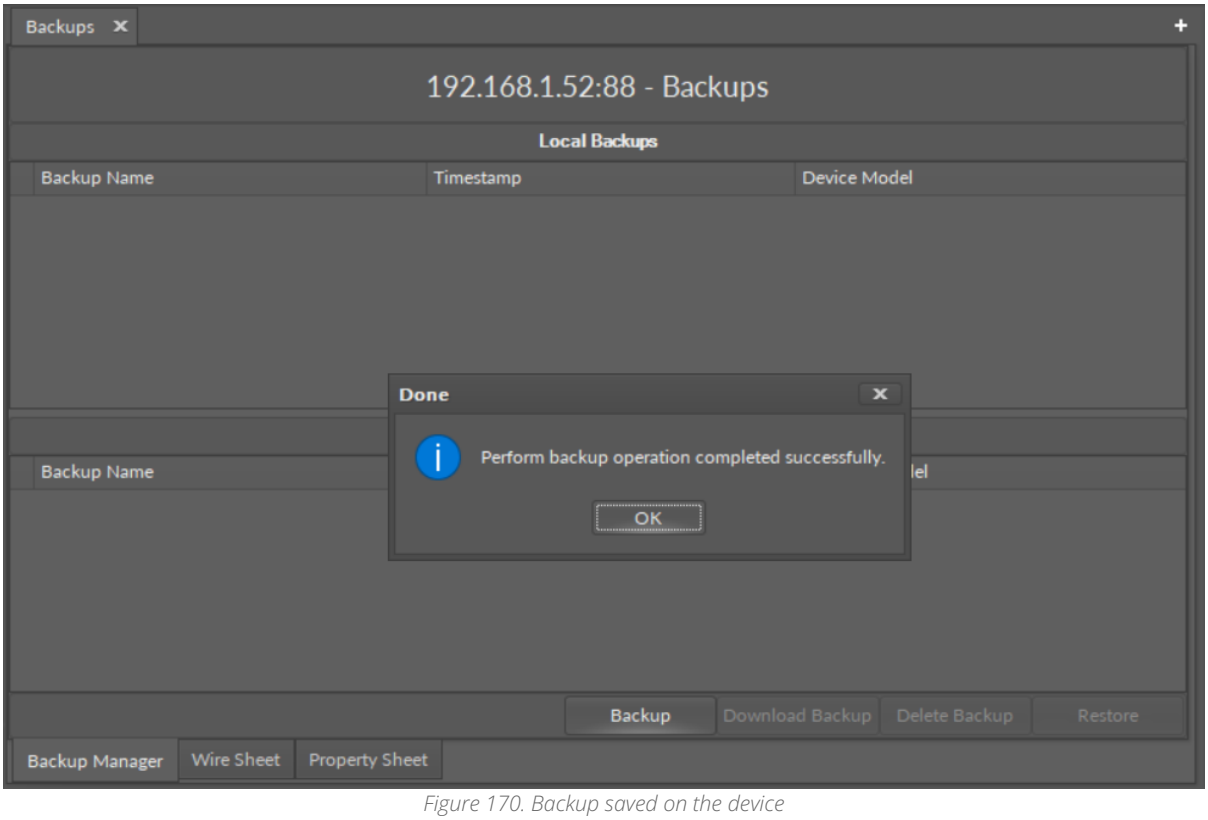



*Figure 170. Backup saved on the device*

- **Download Backup:** downloads a backup to the PC and saves it under a selected name in the iSMA Tool's default backup folder (home\backup\nano EDGE ENGINE); after downloading, the backup is displayed in the Local Backups section;

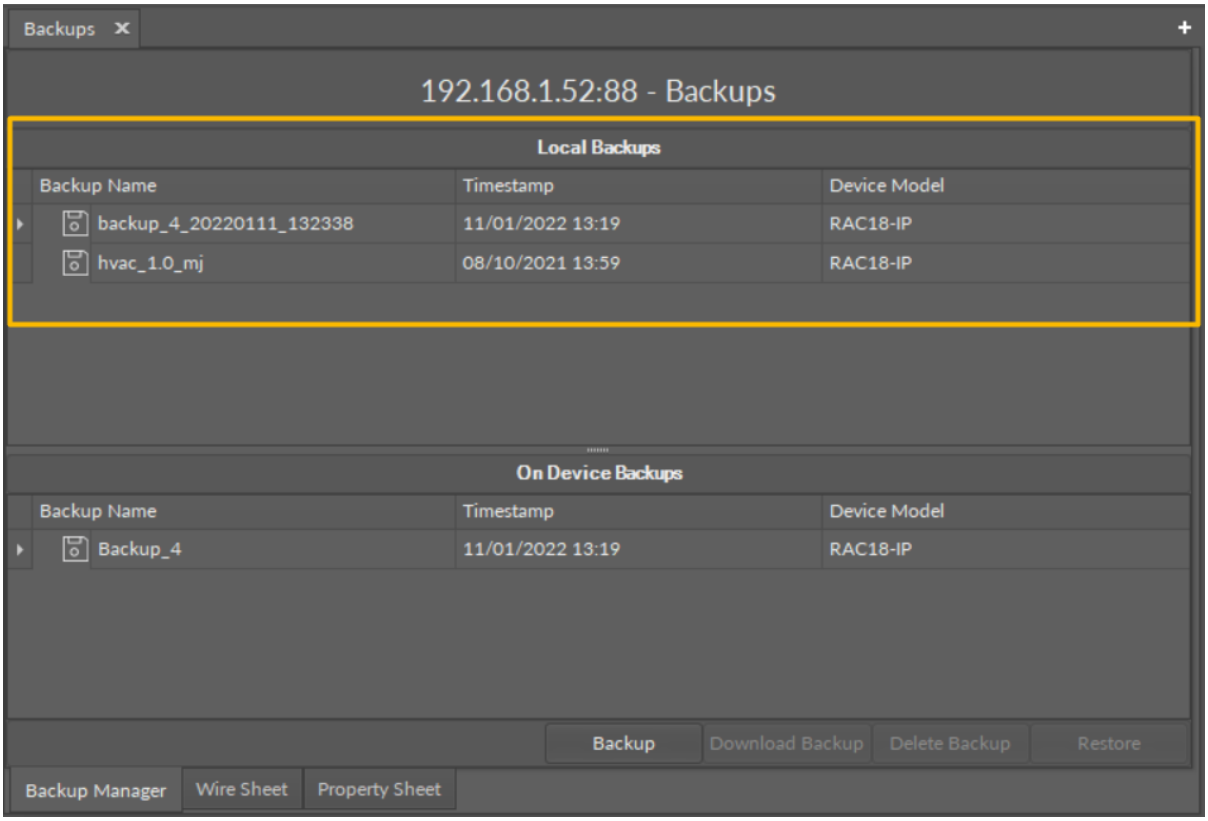



*Figure 171. Backups downloaded locally*

While downloading a backup, it is possible to set its individual name:

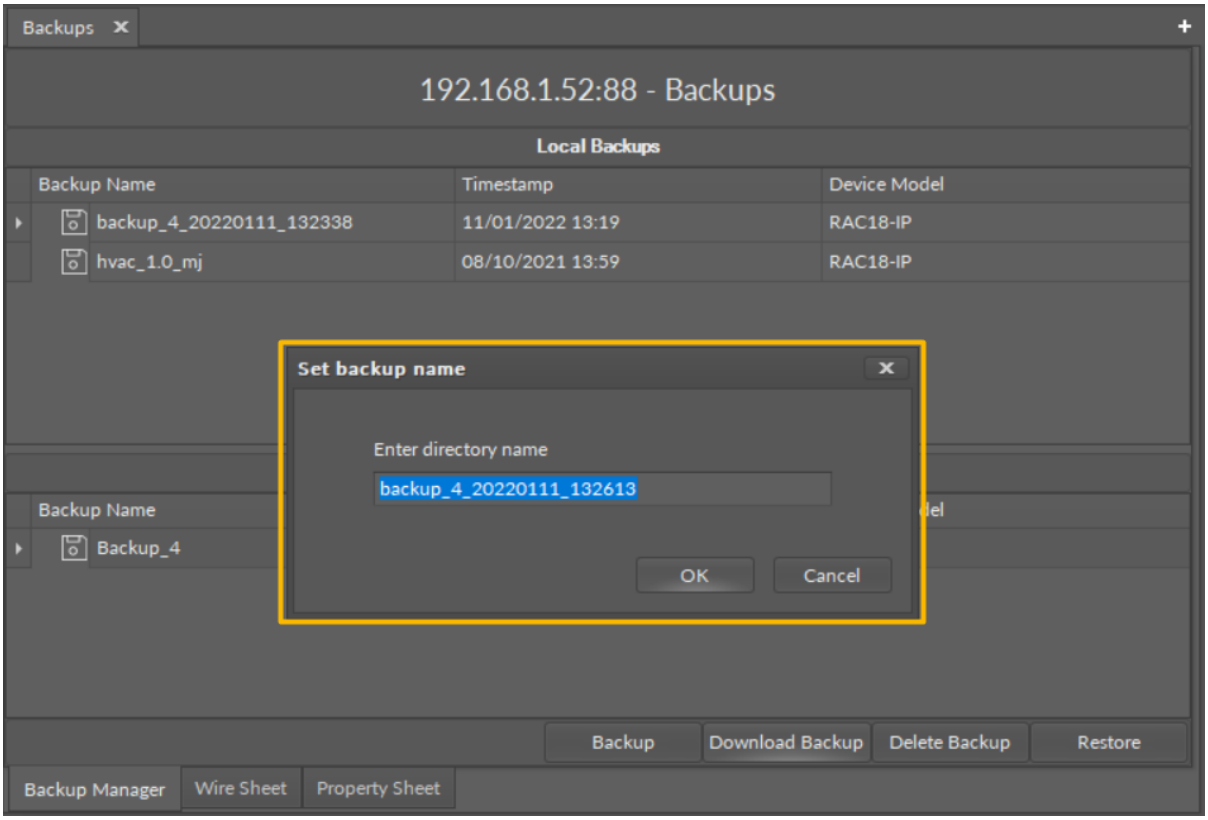



*Figure 172. Setting backup's individual name*

- **Delete Backup:** removes a backup from a selected location (PC or device);
- **Restore:** restores a backup to the device from a selected location (PC or most current backup saved on the device).

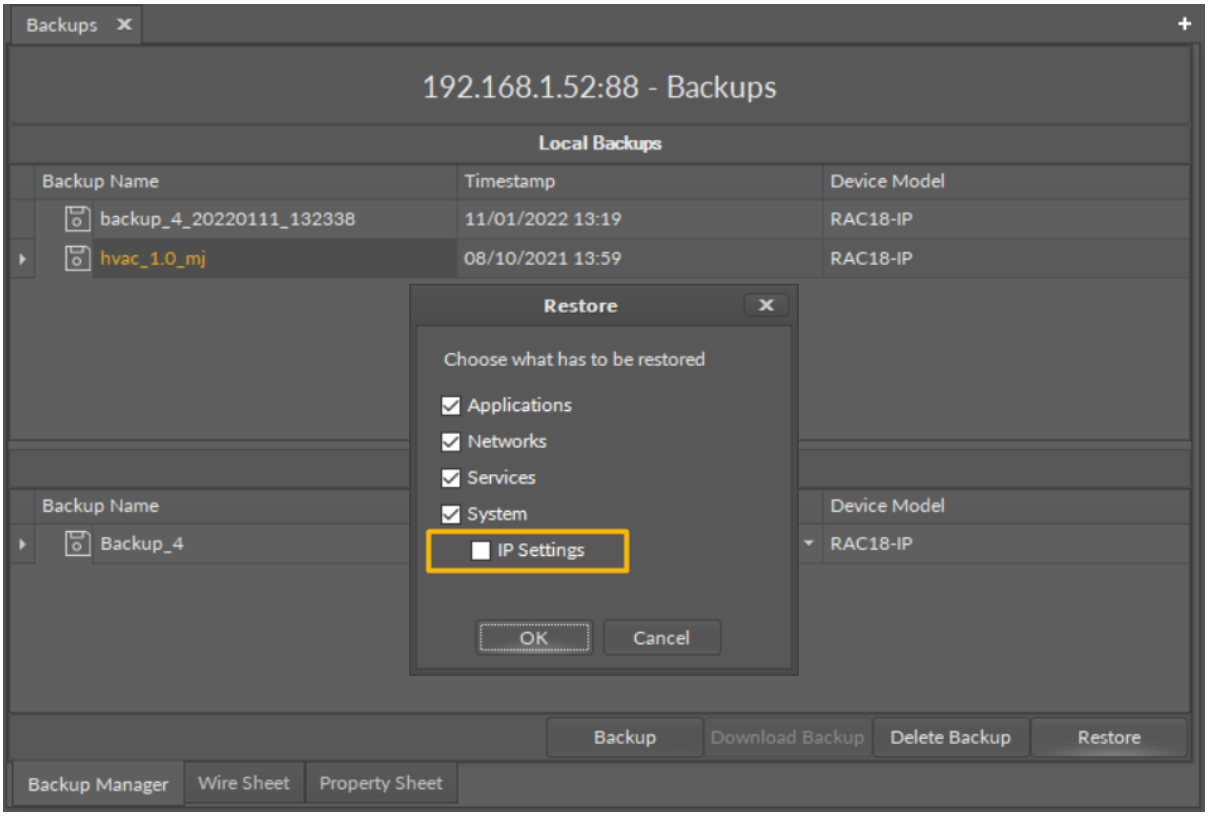



*Figure 173. Restoring a backup*

While restoring a backup, it is possible to selected whether the IP address is restored from the backup too, or lest as set on the device. By default, the option to restore IP address from a backup is unchecked.

**Note:** The only thing subject to selection in backup restoring is the IP address. The other items, including users and passwords are restored from the backup.

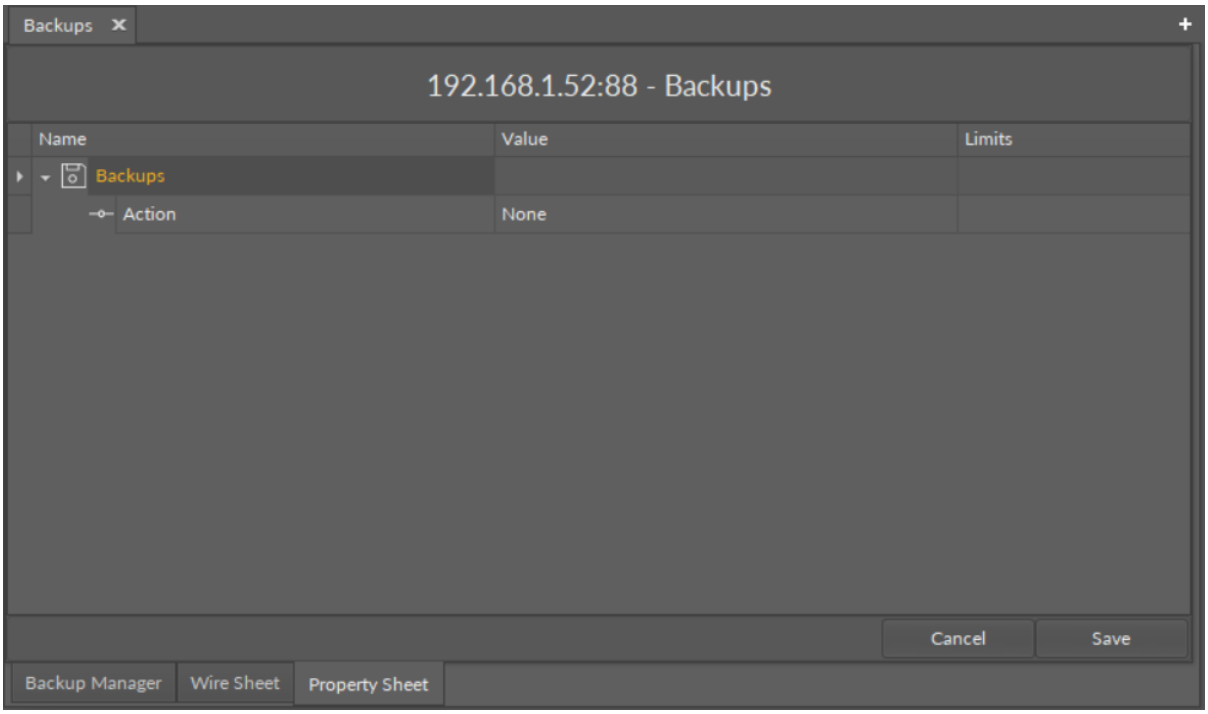



*Figure 174. The Backups component's read-only Action slot*

The Backups component has one read-only slot:

- **Action:** informs, which action is currently ongoing.

## 8.10 Weekly Schedule

The Weekly Schedule view is designed to set schedules for the AnalogSchedule and BinarySchedule components. The view is automatically opened upon double-clicking of the component.

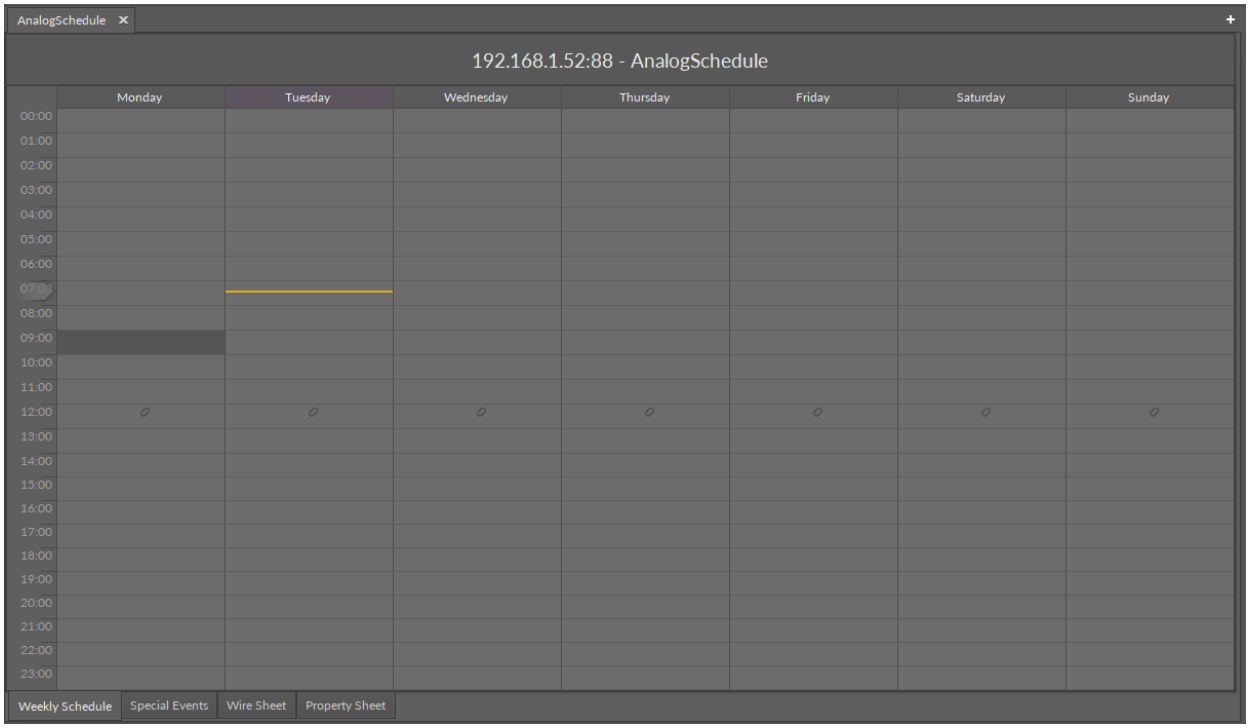



*Figure 175. The Weekly Schedule view*

To set a schedule, first, set the required time interval. If the required time interval lasts for an hour or less, double-click on the required day and time, open the context menu, choose Add Event option, and enter the required value and time limit in the displayed dialog window. If the required time is longer than an hour, click and hold the click on the beginning of the required time interval and drag until the end of it. Open the context menu, choose Add Event option, and set the required value in the displayed dialog window.

In case no specific value is set for a given time interval, the default value is applied.

An event can be added by double-clicking on a schedule or in a context menu.

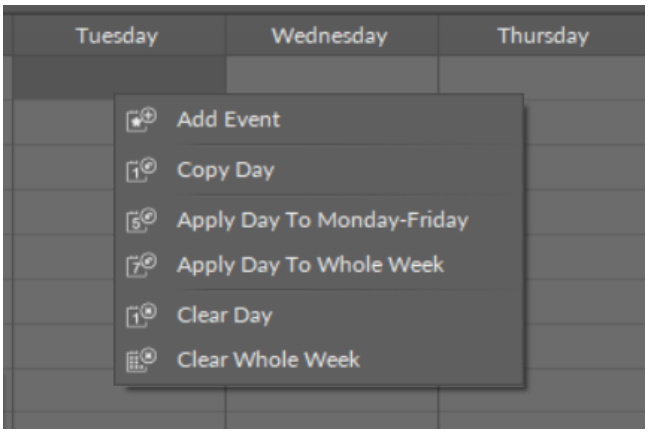



*Figure 176. The context menu in the Weekly Schedule view*

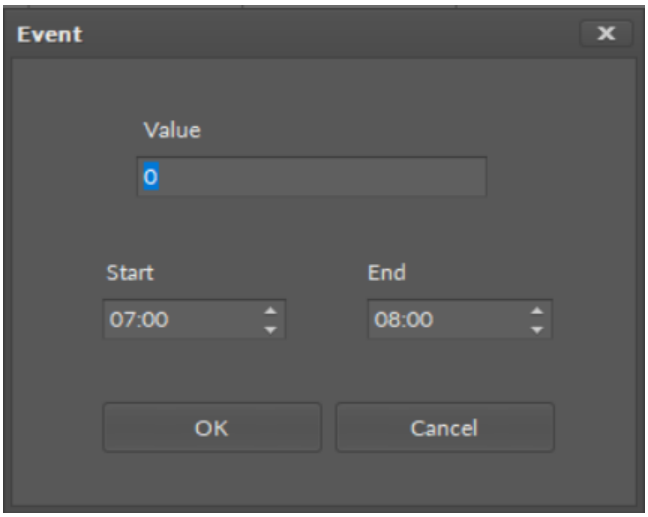



*Figure 177. Adding event in the AnalogSchedule component*

The Event dialog window allows to enter basic data of an event, value and starting and ending time. Confirm with OK to set the event.

## 8.10.1 Context Menu for the Weekly Schedule View

The context menu of the Weekly Schedule view has the following options:

- **Add Event:** opens the Event dialog window and allows to add a single event;
- **Copy Day:** copies all events in a day the context menu is invoked for;
- **Apply Day To Monday-Friday:** copies all events in a day the context menu is invoked for, and pastes these events to Monday, Tuesday, Wednesday, Thursday, and Friday;
- **Apply Day To Whole Week:** copies all events in a day the context menu is invoked for, and pastes these events to all days in a week;

Note: If the Apply Day To Monday-Friday or Apply Day To Whole Week option is invoked on a day with no events, all events on other days will be erased.

- Clear Day: removes all events scheduled for a day the context menu is invoked for;
- Clear Whole Week: removes all events in a whole week.

## 8.10.2 Context Menu for the Event

The context menu for a single event has the following options:

- Edit Event: opens the Event dialog window and allows to edit the selected event;
- Delete Event: remove the selected event from the schedule.

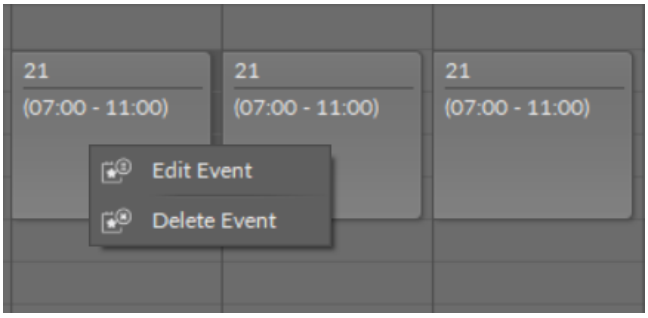



*Figure 178. The context menu for a single event*

**Worth to Notice:**

A single event can also be edited by a drag and drop method.

## 8.11 Special Events

The Special Events view is designed to set irregular events for the AnalogSchedule and BinarySchedule components. The view is automatically opened upon double-clicking of the component.

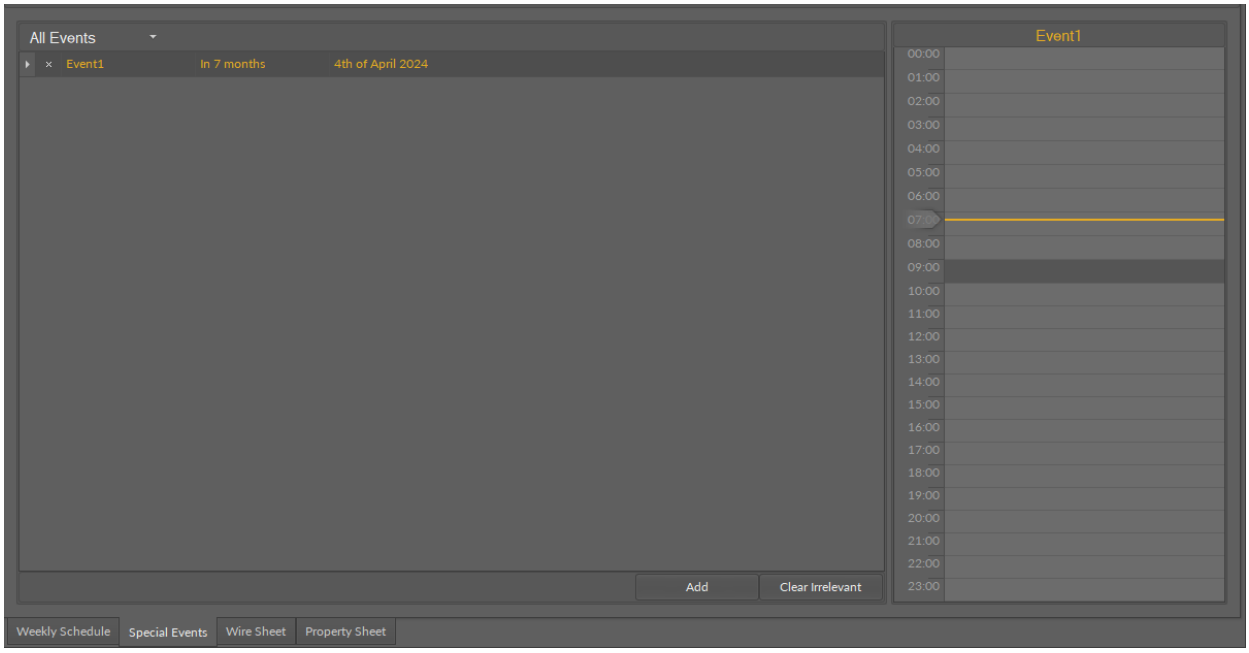



*Figure 179. The Special Events view*

To add a special event, use the Add button in the bottom right corner of the view; the pop-up window appears.

In the Add event dialog window, it is possible to set the following parameters:

- **BACnet priority:** from 1 to 16;
- **Type:** type of the event;
  - Available options: Date Pattern, Date Range, Week and Day, Calendar;
- **Name:** allows to set a name of the event, specific to the user's requirements.

The type of event defines the data entered for each type.

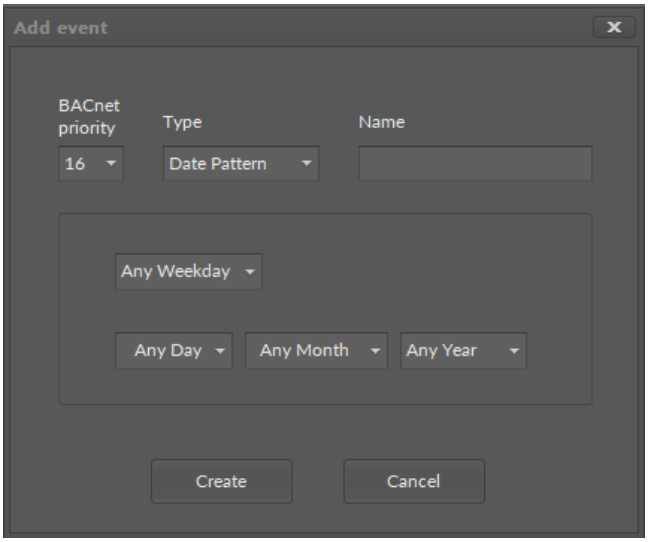



*Figure 180. The Add event dialog window - Date Pattern parameters*

For Date Pattern:

- weekday;
- specific day of a month;
- specific month of a year;
- specific year.

Each of these parameters may be left to any (Any Day, Any Month, etc.).

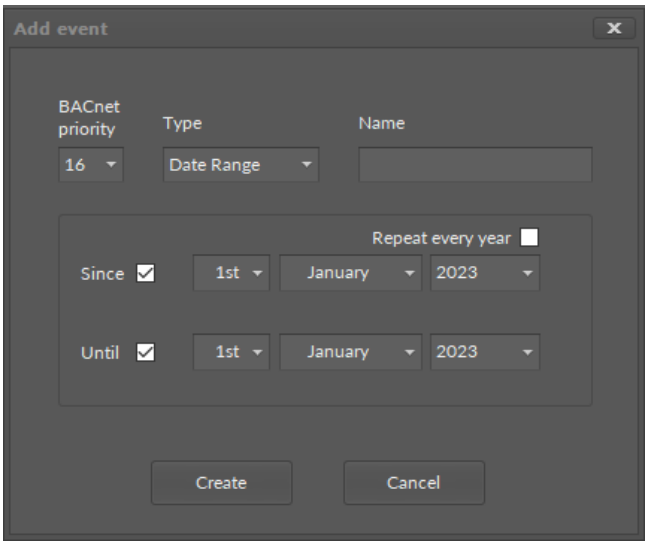



*Figure 181. The add event dialog window - Date Range parameters*

For Date Range:

- Since/Until: allows to specify a beginning or end day of the event (unchecked sets the "future" value);
- specific day, month, and year of a calendar date;
- Repeat every year: if checked, disables the year field.

For Week and Day:

- specific weekday, week, and month of the event.

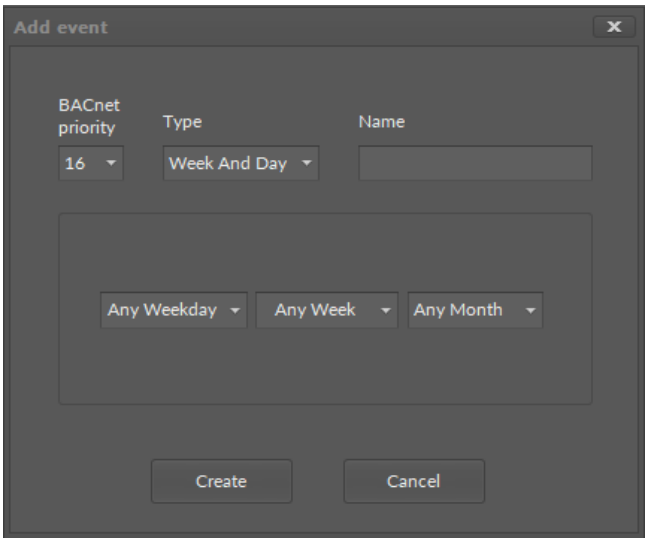



*Figure 182. The Add event dialog window - Week and Day parameters*

For Calendar:

- source Calendar component.

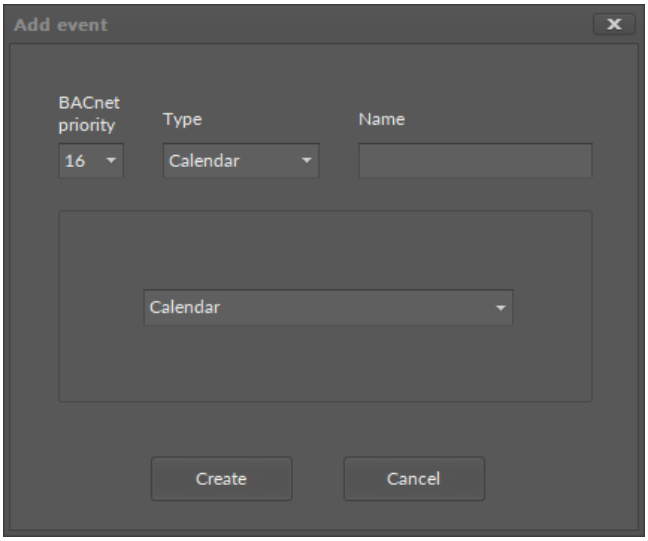



*Figure 183. The Add event dialog window - Calendar parameters*

**Note:** It is possible to set 24 exception event for 1 Calendar source component.

### 8.11.1 Setting Values for Special Events

To set a value for special events created in the schedule, double click in the right column of the view to open the Event dialog window.

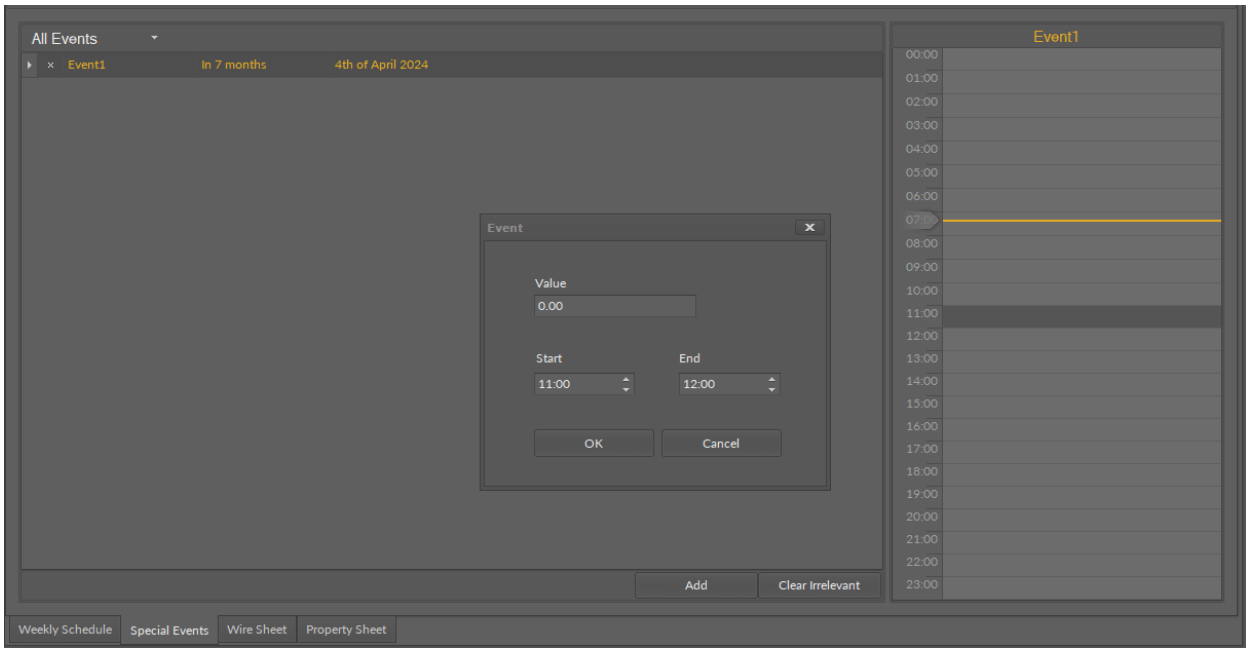



*Figure 184. The Event dialog window*

In the Event dialog window, set the value, start and end time of the event.

## 8.12 Calendar

The Calendar view is designed to set one time or repeatable special days in a year. The view is automatically opened upon double-clicking of the component.

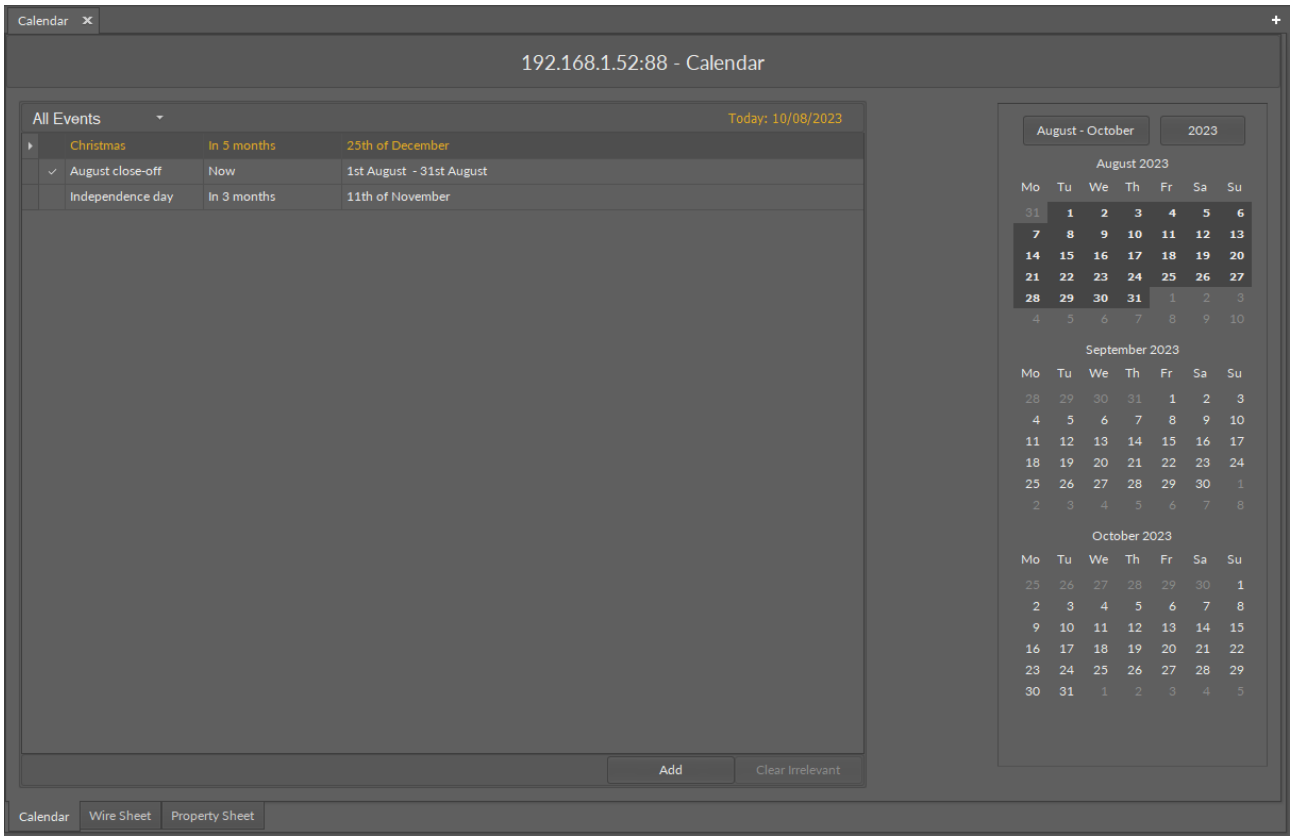



*Figure 185. The Calendar view*

To add an event in the calendar, use the Add button in the bottom right corner of the view; a pop-up window appears.

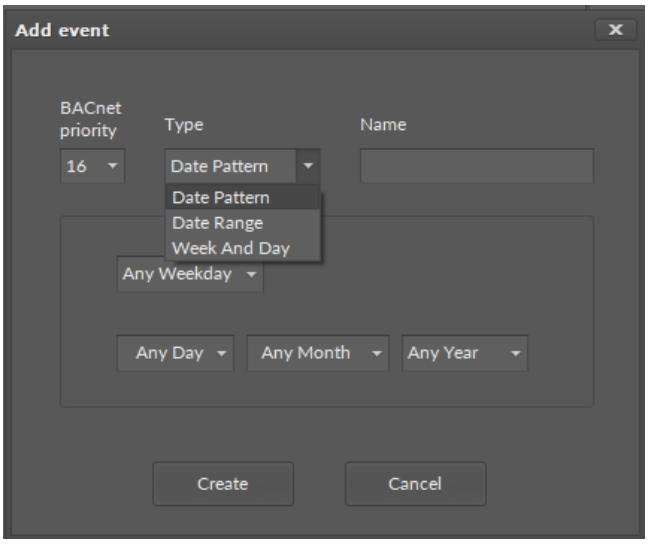



*Figure 186. The Add event dialog window - Date Pattern parameters*

In the Add event dialog window, it is possible to set the following parameters:

- **BACnet priority:** from 1 to 16;
- **Type:** type of the event;
  - Available options: Date Pattern, Date Range, Week and Day;
- **Name:** allows to set a name of the event, specific to the user's requirements.

The type of event defines the data entered for each type.

For Date Pattern:

- weekday;
- specific day of a month;
- specific month of a year;
- specific year.

Each of these parameters may be left to any (Any Day, Any Month, etc.).

For Date Range:

- Since/Until: allows to specify a beginning or end day of the event (unchecked sets the "future" value);
- specific day, month, and year of a calendar date;
- Repeat every year: if checked, disables the year field.

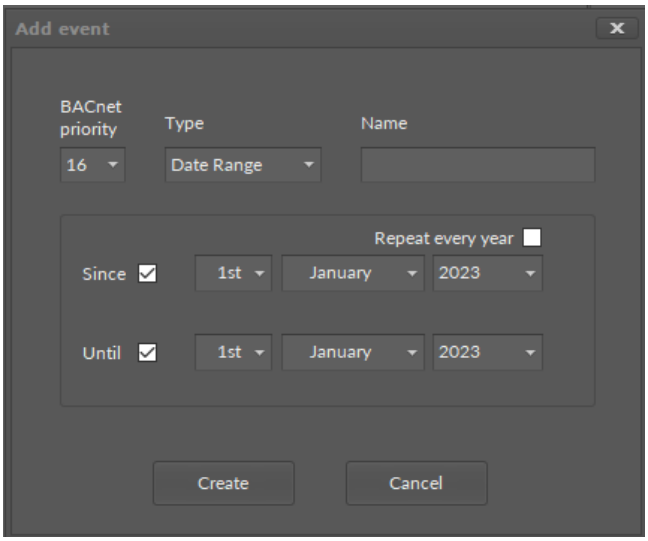



*Figure 187. The Add event dialog window - Date Range parameters*

For Week and Day:

- specific weekday, week, and month of the event.

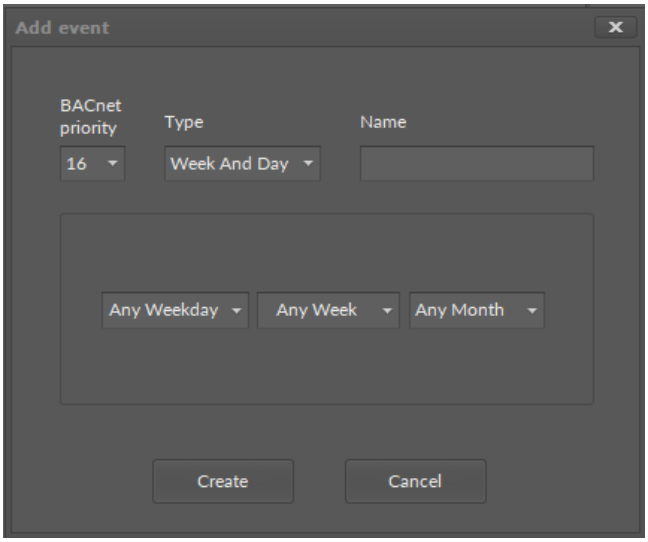



*Figure 188. The Add event dialog window - Week and Day parameters*

Please note that the events in the Calendar cannot be assigned a specific value. Specific values for these events may be set in a relevant schedule component (AnalogSchedule/BinarySchedule) by sourcing the Calendar in the Special Events view. All events in the Calendar will be then assigned with identical values if active. To assign a different type of values, create multiple Calendar components with events of corresponding types.

## 8.13 Time

The Time view is available for the Time component in the System container. The view shows a current time (date and time, time zone, and UTC offset) in the device, and allows to change time values (in the Desired window). The view also allows for enabling the use of a daylight saving time and changing its settings.

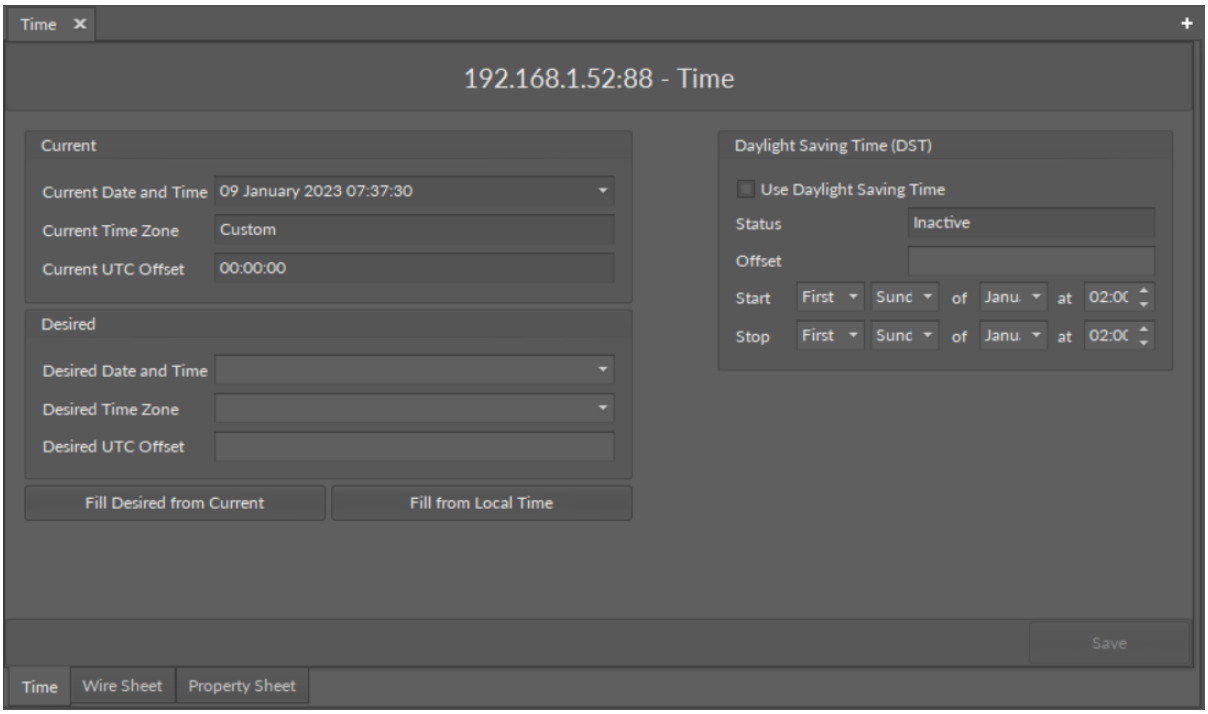



*Figure 189. The Time view*

The Time view has the following fields:

Current window

- **Current Date and Time:** shows a current date and time values in the device;
- **Current Time Zone:** shows a current time zone value in the device;
- **Current UTC Offset:** shows a current UTC offset value in the device;

Desired window

- **Desired Date and Time:** allows to set date and time values to be changed in the device;
- **Desired Time Zone:** allows to set a time zone value to be changed in the device;
- **Desired UTC Offset:** allows to set a UTC offset value to be changed in the device;
- **Fill Desired from Current:** writes current values to the desired fields;
- **Fill from Local Time:** writes PC local date and time values to the desired fields;

Daylight Saving Time (DST) window

- **Use Daylight Saving Time:** enables or disables using a daylight saving time;
- **Status:** informs if a daylight saving time is used (active/inactive);
- **Offset:** allows to set an offset value for a daylight saving time;
- **Start:** allows to set beginning date and time values for using a daylight saving time;
- **Stop:** allows to set ending date and time values for using a daylight saving time.

# 9 iC Tool with Sedona Devices

From the 1.2 version, the iC Tool supports devices based on the Sedona Framework.

## 9.1 Kit Manager

The Kit Manager is a specific tool in the device allowing the management of its kits. The Kit Manager allows installing and uninstalling of kits as well as changing versions of kits already existing in the device.

The Kit Manager allows indication of actions which are to be performed on kits (adding, removing, version change) and performing all actions at once after giving a command to update the device's kits.

**Warning!**

Every time kits are edited in the device, it requires a restart of the device.

**Warning!**

Working with the Kit Manager for the iSMA-B-FCU controller is blocked due to the construction and purpose of the controller.

### 9.1.1 Kits Installed in iC Tool

The Kit Manager may be opened in two ways–defining or without defining the device the Kit Manager will cooperate with–already at the start-up stage.

Opening the Kit Manager without defining the device it will work with, allows to check what kits are currently available (installed) in the iC Tool, which will be available to be added to the device in the next stage.

To open the Kit Manager with a list of kits available the in iC Tool it is necessary to go to the main menu -> View -> Device Managers -> Kit Manager.

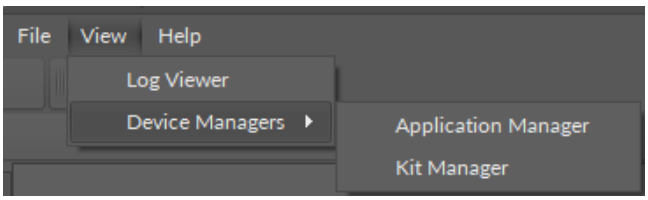



*Figure 190. Accessing the Kit Manager*

The appearing view consists of two sections: empty upper one for cooperating with devices and lower one with a kit list, see the figure below.

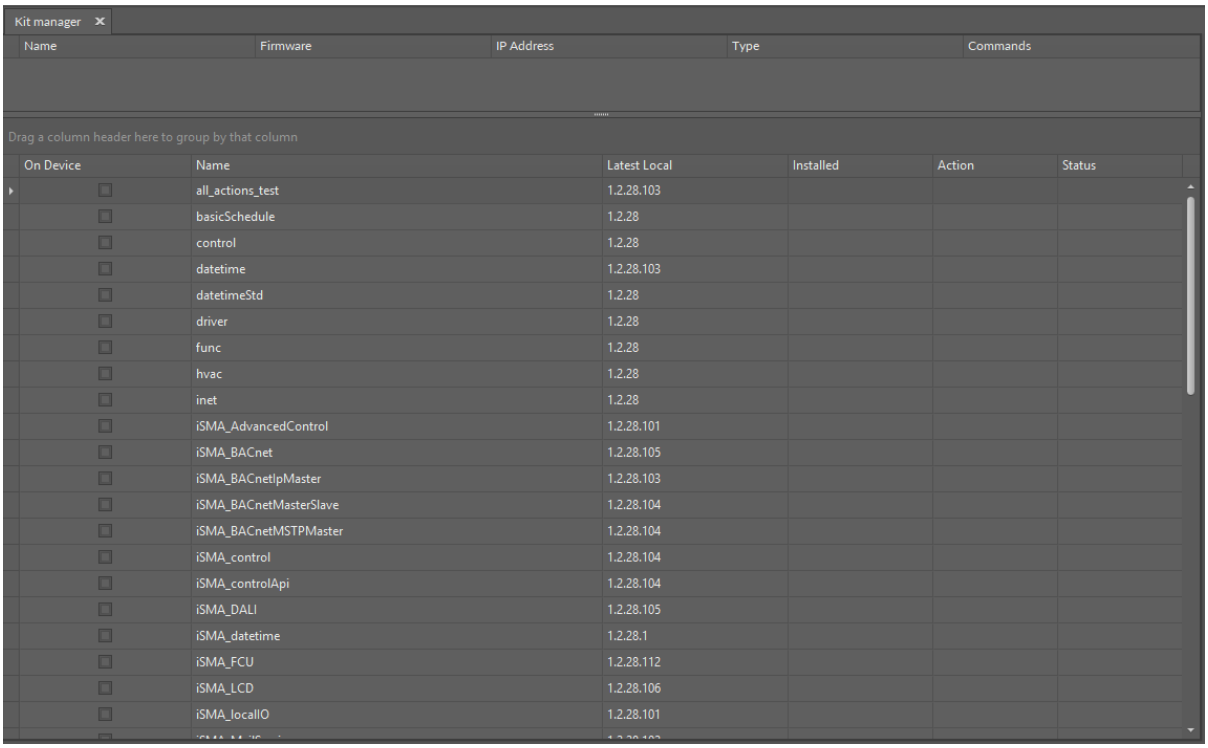



*Figure 191. Kit Manager*

The lower section shows an uneditable table where every row represents a kit installed in the iC Tool. The meaning of columns is as follows:

- **On Device:** informs whether the kit is currently installed on a chosen device. In this case the table shows only kits present in the iC Tool, so all the checkboxes in this column are empty;
- **Name:** shows a name of the kit;
- **Latest Local:** shows a number of the newest version of the kit installed locally in the iC Tool.

The list of kits visible in the table corresponds with a physical content of kits folder in the main iC Tool location: /home/kits. Proper placing of user's own kit in this location allows to add it to the Kit Manager.

To update kits in a device, the device to work with has to be exactly defined.

There are two methods to initiate the Kit Manager at the same time defining the device it will cooperate with. They are:

- initiating the Kit Manager from the context menu;
- initiating the Kit Manager from the Object Properties Window.

## 9.1.2 Adding Devices to Kit Manager

The function of adding the device to the Kit Manager, described below, may be used regardless of the method the Kit Manager has been initiated.

To add a single device or many devices at once to the Kit Manager, the required device needs to be dragged from the Workspace Tree into the upper part of the Kit Manager view, where the list of chosen devices is. There is also a possibility of dragging the whole subtree from the Workspace Tree with many devices at once. In this situation the iC Tool will filter the dragged objects and add only the devices to a view list. The figure below

shows a view after dragging (adding) all the devices placed in the Site B–Shopping Centre subtree.

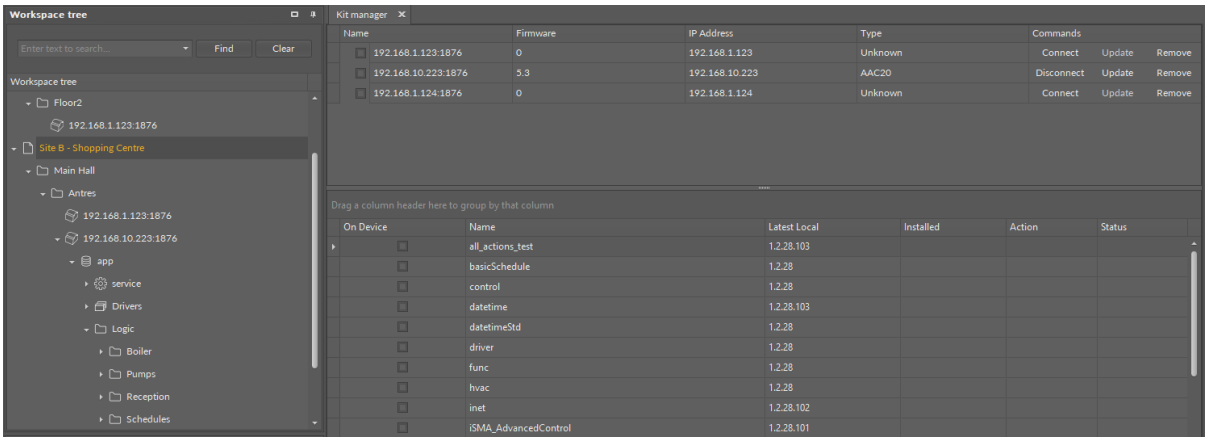



*Figure 192. Devices added to the Kit Manager*

After adding the devices to the Kit Manager, it is necessary to select one of the devices, which will be operated, by ticking a checkbox. Only when the device is defined, it will be possible to use the Kit Manager for this device.

Note: The Kit Manager allows to work with one device at a time. The device list in the upper part of the Kit Manager is used for an easy switching between devices.

## 9.1.3 Initiating Kit Manager from the Context Menu

Another method to add a single device to work with the Kit Manager is using the context menu, initiated on the device in the Workspace Tree window, see the figure below.

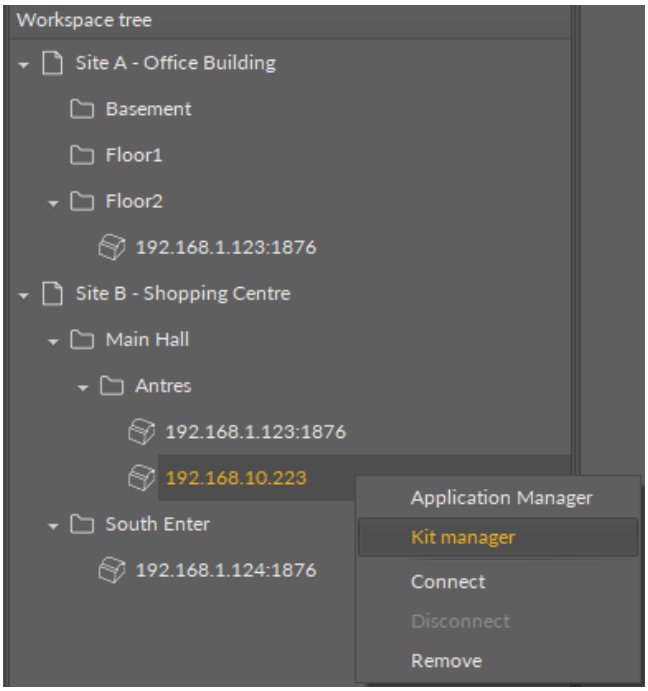



*Figure 193. Initiating the Kit Manager from a context menu*

After automatic opening of the Kit Manager, the device is added in the upper area of the view, as shown on the figure below.

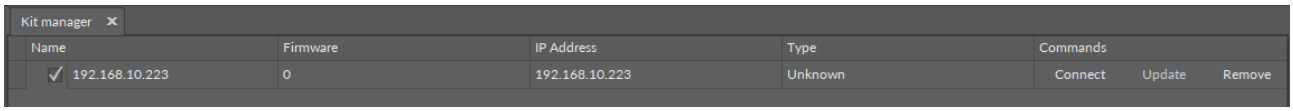



*Figure 194. Kit Manager opened for a device*

If the device parameters shown in the table (such as on the figure above) are not specified, before working further with the device it needs to be connected using a Connect button. After a correct authorization and connection with the device, all the device parameters will be shown as on the figure below.

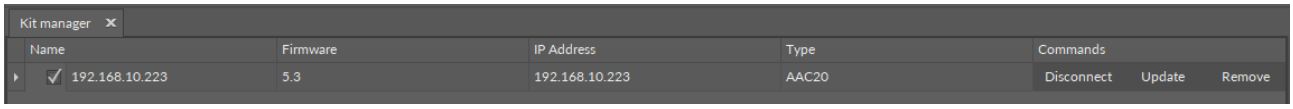



*Figure 195. Kit Manager for a connected device*

It is now possible for the Kit Manager to work with this device.

### 9.1.4 Initiating Kit Manager from the Object Properties

Another convenient way of initiating the Kit Manager for a preselected device is using the button available in the Object Properties window.


*Figure 196. Initiating the Kit Manager in the Object Properties window*

After selecting the device in the Workspace Tree, its data will be shown in the Object Properties window, as shown on the figure above. It is possible that data will partially be not available due to a lack of information about the device until connecting with this device. Regardless of whether the device is connected or offline, the Kit Manager can be started by pressing the Kit Manager button, which will open with a device chosen.

## 9.1.5 Using the Kit Manager

After choosing the device to work with, using methods described earlier, a table with a list of kits is shown in the bottom part of the Kit Manager. In the upper part of the table, the kits, which are currently installed in the device, are grouped, and in the lower part of the table there are filtered kits, fitting to a selected device, which are available to be loaded into the device. After deselecting the device, the filter will be turned off, and a whole list of locally available kits will be shown in the iC Tool, see the figure below.

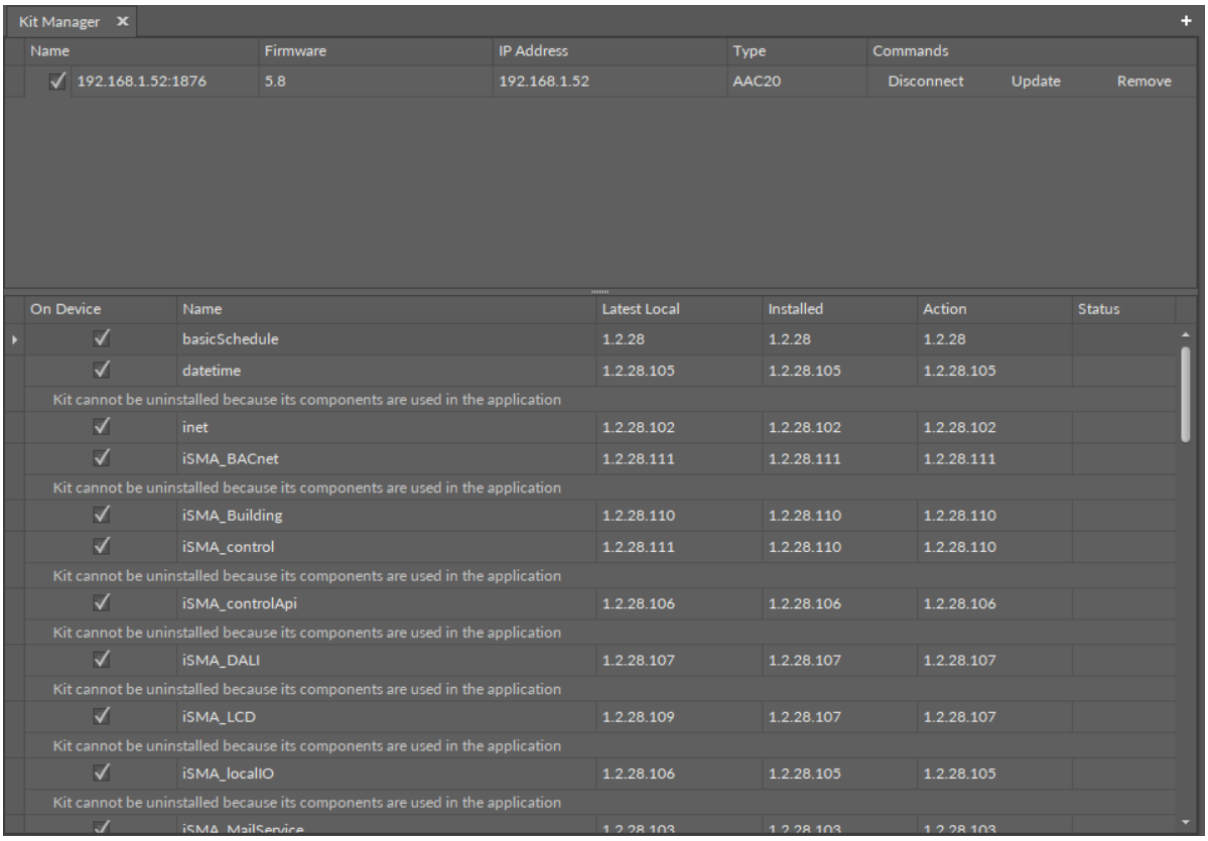



*Figure 197. The Kit Manager*

The columns in the above table are defined as follows:

- **On Device:** shows if a particular kit is already installed on the selected device;
- **Name:** the name of the kit;
- **Latest Local:** the number of the latest kit version installed locally in the iC Tool;
- **Installed:** the number of the kit version currently installed on the device;
- **Action:** the list of kits versions, which are available locally in the iC Tool and are ready to be installed on the device. A version number to be installed on the device is chosen from a drop-down menu (upgrade or downgrade process).
- **Status:** the column informs about an action that will be taken on the selected kit.

The user's role is limited to editing two columns: On Device and Action.

## Operations on Kits

Below there are descriptions how to remove, replace, and add a kit to a device.

### Adding Kits

In order to add the kit to the device, select its checkbox, and, if needed, define the version of added kit (by default, the newest version is selected). The Install command appears in

the Status column, and the kit is added once the user confirms the Update command in the upper Device panel of the Kit Manager.

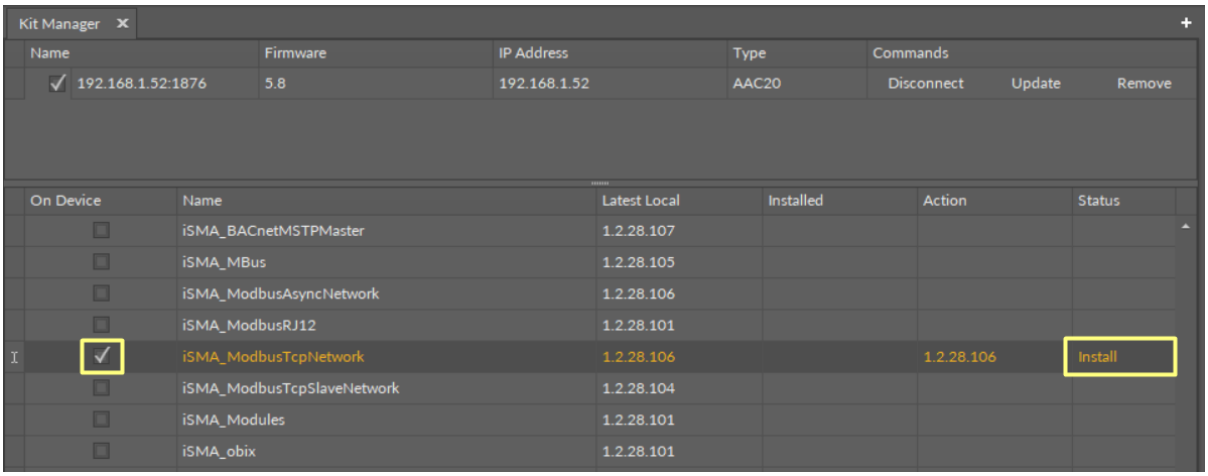



*Figure 198. Adding a kit*

## Upgrading/Downgrading Kits

The kit installed on the device may be replaced by installing its newer or older version. Both upgrade and downgrade operations are done by choosing a proper number of the kit version in the Action column.

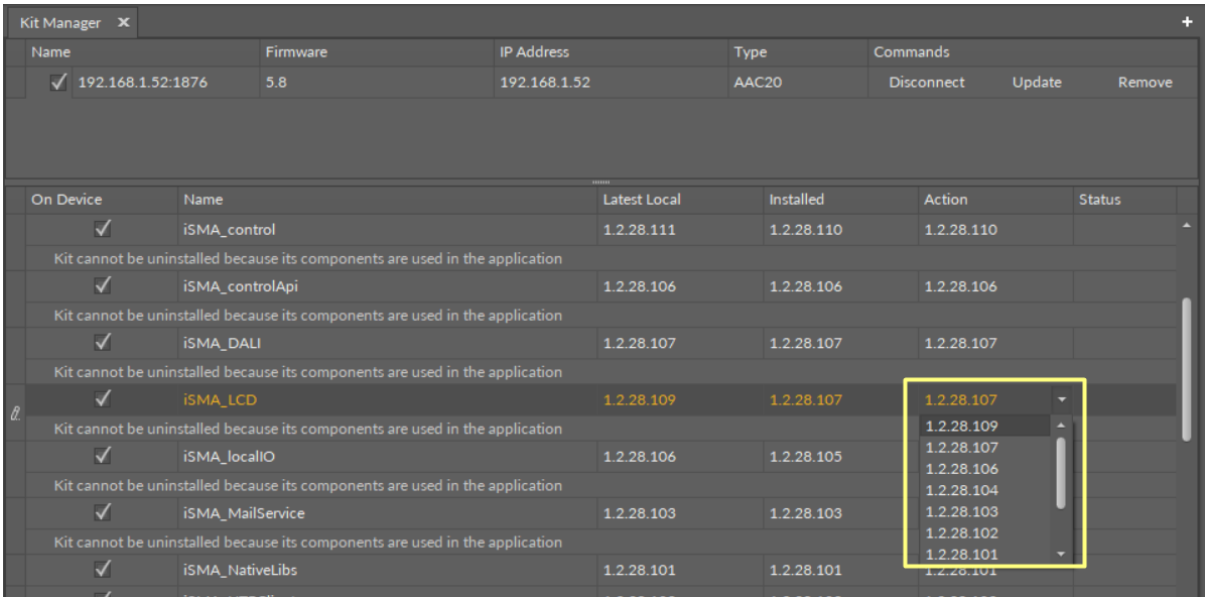



*Figure 199. Selecting a version to upgrade or downgrade in the Action column*

Depending if the current kit is replaced with a newer or older one, the iC Tool will display the planned action in the Status column, Upgrade or Downgrade.

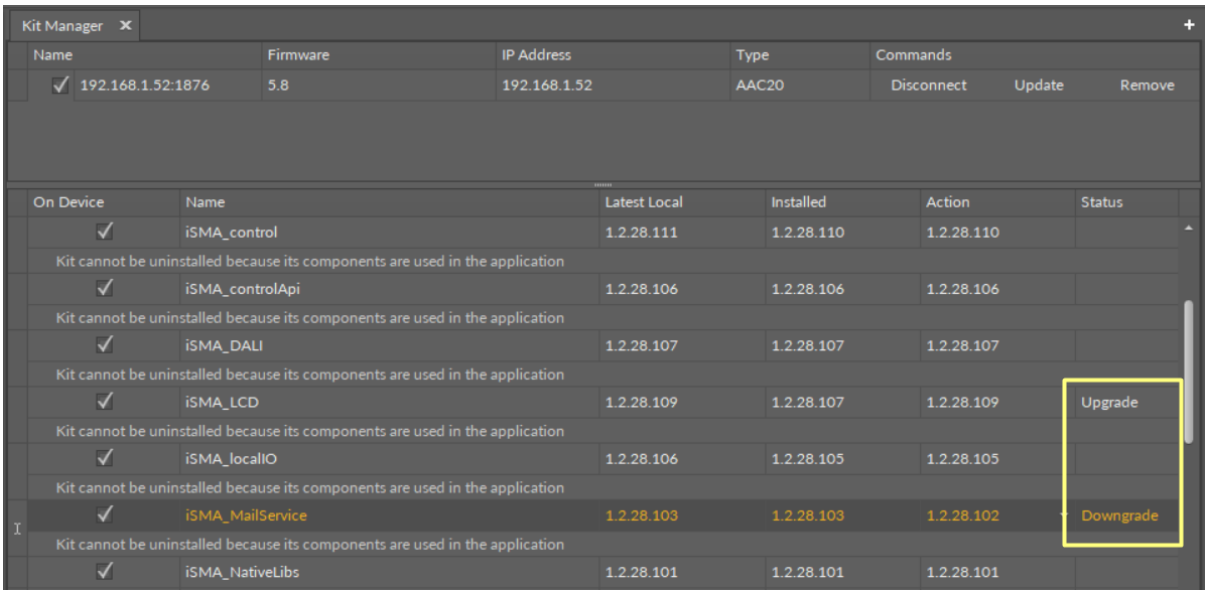



*Figure 200. Upgrade or downgrade options*

## Removing/Uninstalling Kits

If a user deselects the checkbox, the selected kit is chosen to be removed. The Uninstall command appears in the Status column, and the kit is removed once the user confirms the Update command in the upper Device panel of the Kit Manager.

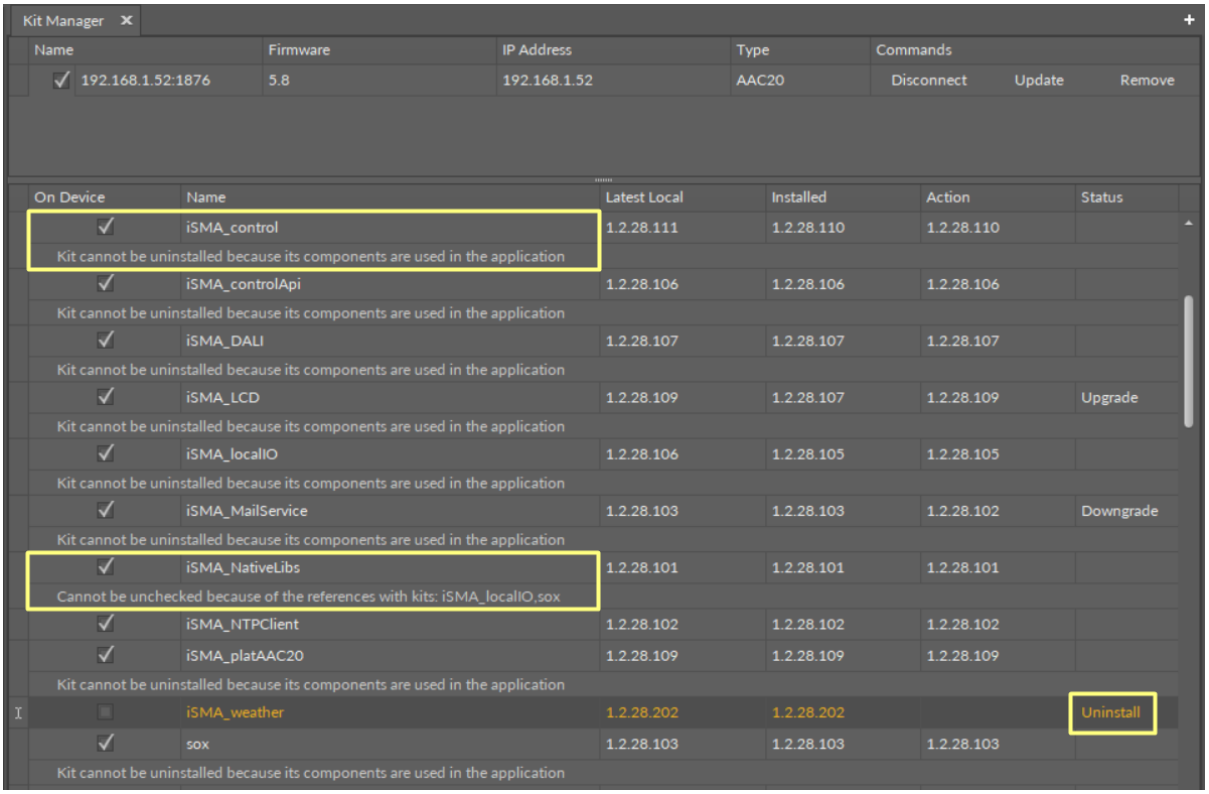



*Figure 201. Removing kits*

In order for the kit to be removed, the components used in the application shall be removed, or the kit needs to be left installed.

## Completing Operations - Updating Kits

After defining the kits to be added, removed, or changed the Update command needs to be confirmed in the upper Device panel of the Kit Manager. The Update command is displayed in the Commands column.

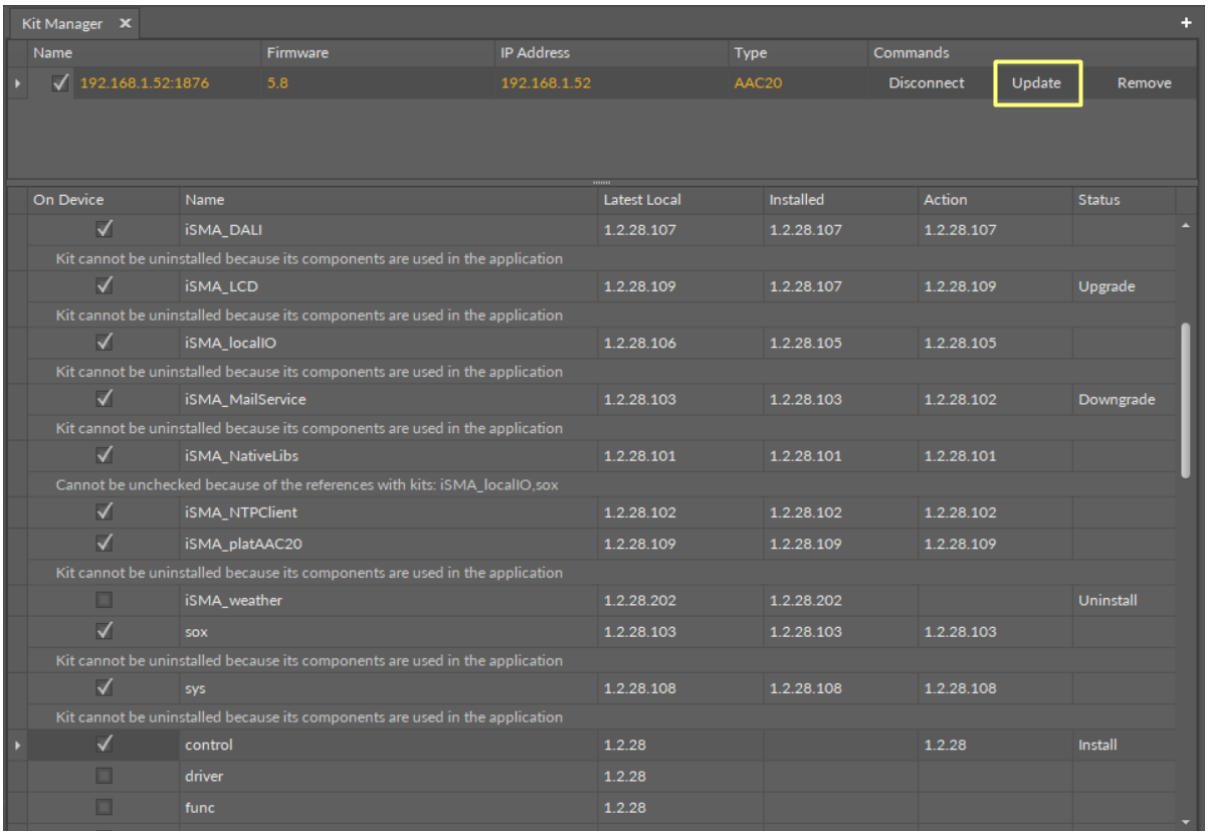



*Figure 202. Updating kits*

While updating the kits in the device, the iC Tool interface is blocked for the time.

**Warning!**

Finalizing the kits update requires restarting the device.

After restarting the device iC Tool reconnects with the device.

Note: The update process is monitored in detail in the Console window, which helps in analyzing the situation in case the update could not be completed.

## 9.1.6 Adding External Kits

The iC Tool allows to expand the default kit base with external kits, for example, user's own kits. To add them, select the Import Framework Files option, available in the main menu under Import:

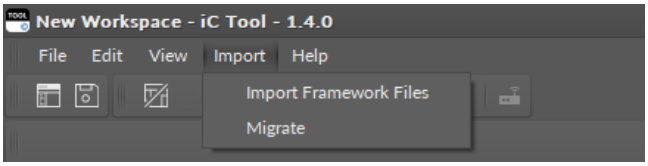



*Figure 203. Import options*

Once selected, a dialog window opens to specify the location of the imported files.

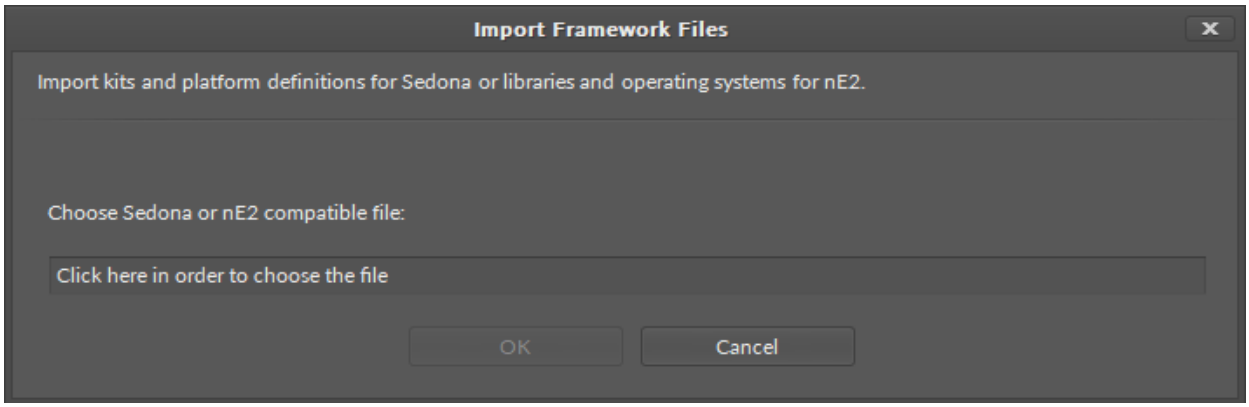



*Figure 204. Adding external kits*

The iC Tool may import single files or compressed folders with .zip extension containing a single kit or whole packages containing kits, manifests, and .par files.

Note: The iC Tool can read the .zip files including other .zip files containing kits, manifests, and .par files, and can display a summary of how many files have been imported at the end of the process, in a window and in console, as shown on the figure below. If such package contains files, which are not kits, manifests, or .par files, they are skipped.

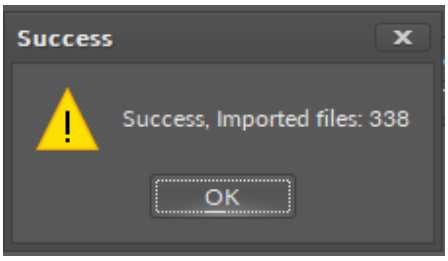



*Figure 205. Successful upload*

After a successful import the added kits are uploaded to the Kit Manager. The list of kits, which can be installed on the selected device, is displayed in the bottom part of the Kit Manager view  It includes the kits, which were added manually. If the external kits do not appear on the list of available kits, it means they are not compatible with the selected device and cannot be uploaded. In this case, after deselecting the device, external kits will be displayed on the list in the lower part of the view, because in this mode the iC Tool shows all kits installed in it.

## 9.2 Application Manager

The Application Manager is a tool that allows to manage applications of a specific device and the applications saved locally in the iC Tool.

The Application Manager allows to:

- manage applications saved locally in the iC Tool;
- load applications to the device;
- download applications from the device.

**Warning!**

Loading an application to the device overwrites the application already saved in the device.

**Warning!**

Each loading of application to the device requires restarting the device.

Application Manager may be initiated in two ways: without defining the device it will work with and with defining the device it will work with.

### 9.2.1 Applications Installed in iC Tool

Opening the Application Manager without defining the device it will work with, allows to check what applications are currently saved in the iC Tool, which will be available to load into the device at the next stage.

To initiate the Application Manager with a list of applications available in the iC Tool go to the main menu: View -> Device Managers -> Application Manager.

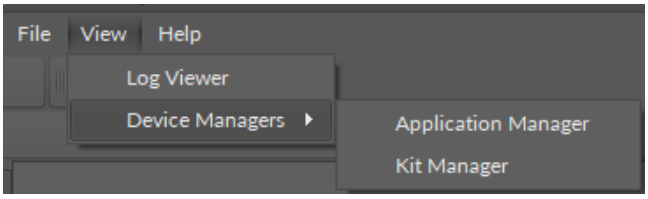



*Figure 206. Accessing the Application Manager*

The view which will appear is organized into two sections: upper, showing no connected devices, and lower with a list of available applications, see the figure below.

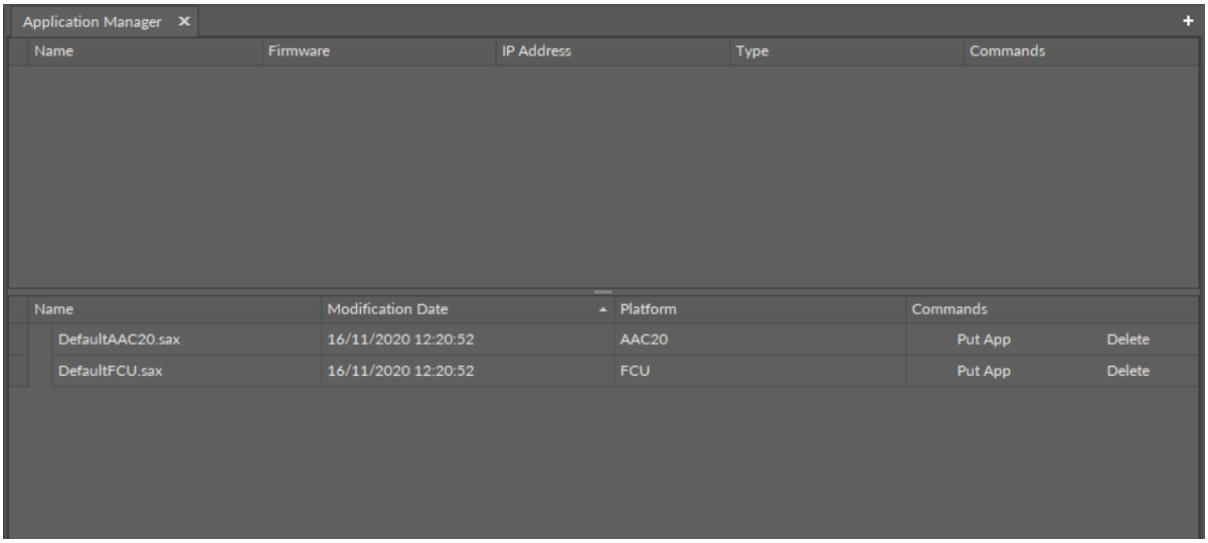



*Figure 207. Application Manager*

Lower section shows a table where every row represents an application saved in the iC Tool along with a respective .sax file. In the iC Tool structures the .sax file is placed in the /home/Applications folder. Copying a custom application (.sax file) to this location allows to add it to Application Manager.

The meaning of columns is as follows:

- **Name:** the name of application and associated .sax file. The user can edit the name by clicking the application name. Renaming the application automatically changes the name of the .sax. file.
- **Modification Date:** the date of the last modification of the. sax file.
- **Platform:** the type of the device, which the application was downloaded from.

- **Commands:** buttons allowing to send the application to a particular device (Put App button) and the Delete button removing local application along with the .sax file from the iC Tool.

> **Warning!**
>
> Using the Delete button removes not only an application from the Application Manager list, but also a physical .sax file from the hard drive.

To download or upload an application from/to the device, the device to work with needs to be specifically defined.

There are a few ways to initiate the Application Manager:

- initiating the Application Manager from the context menu;
- initiating the Application Manager from the Object Properties window.

The rules of initiating the iC Tool Managers, as well as choosing the device to work with, are discussed in detail in the Kit Manager section.

## 9.2.2 Using the Application Manager

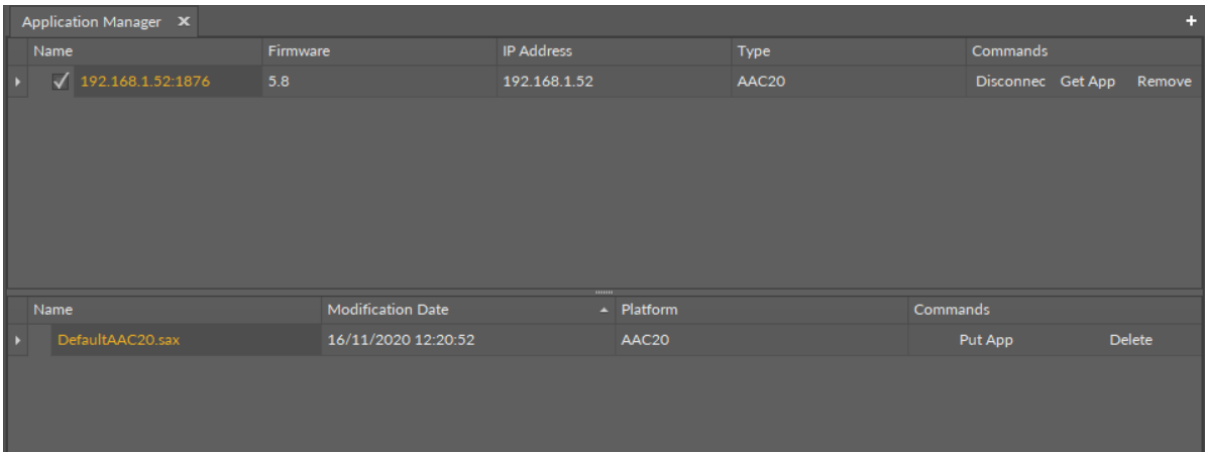



*Figure 208. Application Manager*

The columns in the above table are defined as follows:

- **On Device:** shows if a particular kit is already installed on the selected device;
- **Name:** the name of the kit;
- **Latest Local:** the number of the latest kit version installed locally in the iC Tool;
- **Installed:** the number of the kit version currently installed on the device;
- **Action:** the list of kits versions, which are available locally in the iC Tool and are ready to be installed on the device. A version number to be installed on the device is chosen from a drop-down menu (upgrade or downgrade process).
- **Status:** the column informs about an action that will be taken on the selected kit.

The user's role is limited to editing two columns: On Device and Action.

### Operations on Applications

Below there are descriptions how to remove, replace, and add a kit to a device.

### Downloading Application from Device (Get App)

After connecting with a chosen device its application can be downloaded by pressing the Get App button, see the figure below.

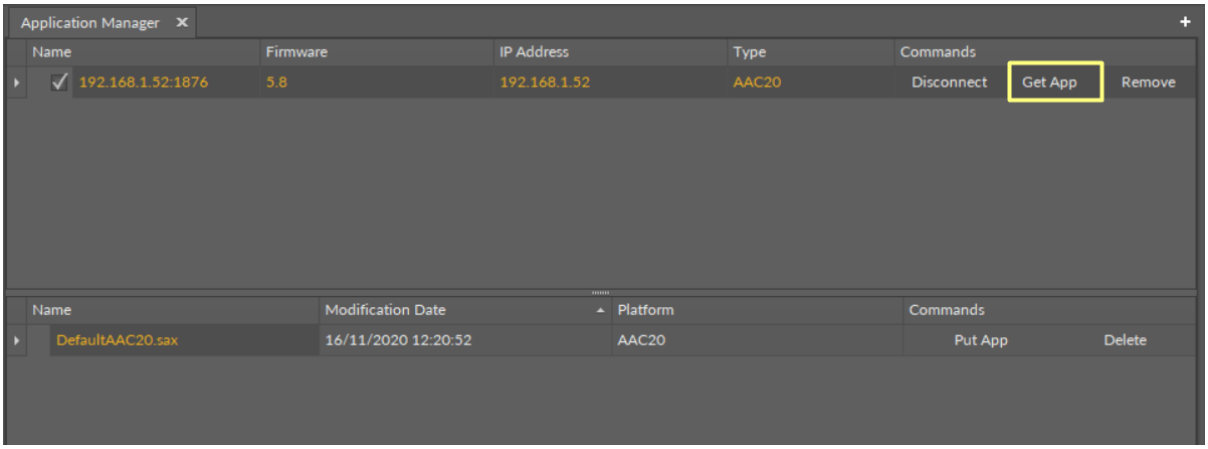



*Figure 209. Download application*

After initiating the command, a starting process reads an application from the device and creates the .sax file (with a name consisting of the application name and the device's IP address). This process blocks the iC Tool for the time of its operation. The newly created file will be added to the list of available applications saved in the iC Tool, and the physical .sax file will be located in the iC Tool main folder (/home/Applications). The default name of the application file may be changed by editing the Name column.

Note: Downloading an application from the device does not stop or restart the device. Downloading does not disrupt the device's work in any way other than pausing its interface for the time of downloading.

Note: Application downloading process is monitored in detail in the Console window, which helps to analyze the situation, in case downloading the application could not be completed.

### Uploading Application to Device (Put App)

Application uploading procedure needs to begin with defining the device to work with Application Manager by selecting a checkbox for a particular device in the upper part of the Manager view.

Next, one of the available applications in the lower part of the Manager needs to be selected by pressing the Put App button in the Commands column.

> **Warning!**
>
> Pressing the Put App button will overwrite the application already installed in the device and reset the device in the final phase of the process.

The figure below shows a situation before beginning of the application upload into the iSMA-B-AAC20 controller, where the user is asked for confirmation due to overwriting of the application currently installed in controller.

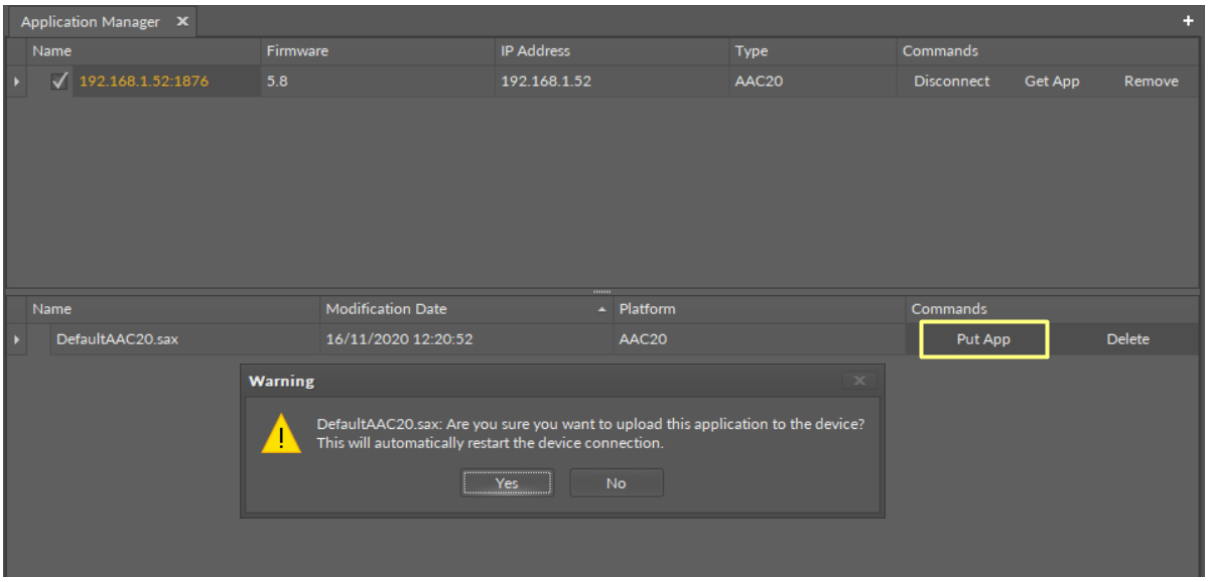



*Figure 210. Put App notice*

After approving the confirmation, a process blocking the iC Tool interface runs for a period of time, when the iC Tool uploads a chosen application and restarts and reconnects the device.

Note: Application uploading process is monitored in detail in the Console window, which helps to analyze the situation when application upload could not finish.

## 9.3 Sedona Specific Views

For Sedona based devices, the iC Tool offers views that are tailored for the Sedona functionalities.

### 9.3.1 Slot Sheet

A Slot Sheet view displays a list of slots of a selected component. It is not a dynamic view– the values of component's slots are read once, when the Slot Sheet is opened, and there is no possibility to edit them.

The Slot Sheet view provides information about the slots of the selected component such as slot type, slot variable type, and slot facet interface.

The slot list contains all component's slots, including hidden slots which are not visible in the Wire Sheet view. For the LeadLag component the number of slots presented in the Slot Sheet view is much higher than visible in the Wire Sheet view (see the figure below).

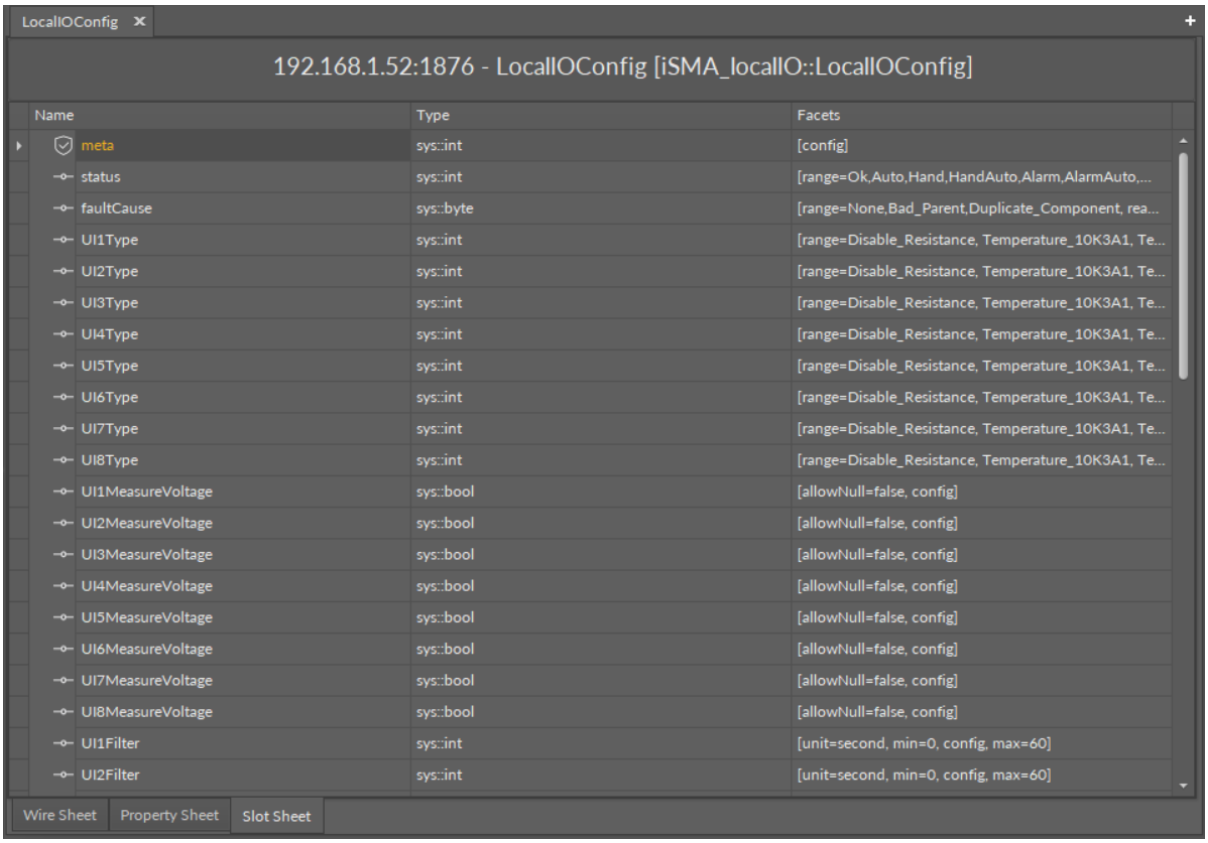



*Figure 211. The Slot Sheet view*

## 9.3.2 Schedule Weekly

In the iC Tool there are two components that allow to set schedules for specific events– the BooleanScheduleWeekly and NumericScheduleWeekly. Both offer a Schedule Weekly view, which facilitates setting a schedule, and a Special Events view, which allows to add special events to the schedule.

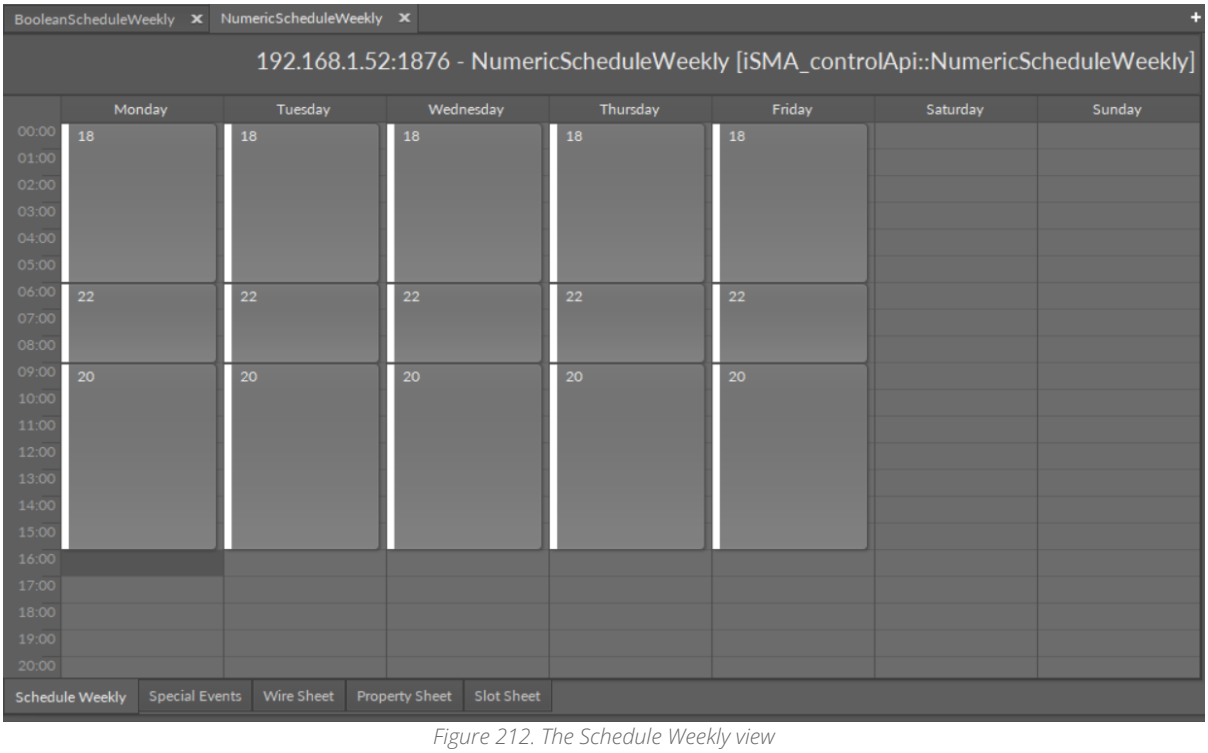



*Figure 212. The Schedule Weekly view*

In order to open the Schedule Weekly view, double-click either one of the scheduling components. The Schedule Weekly view displays a one week view, each day divided by hours.

To set a schedule, first, set the required time interval. If the required time interval lasts for an hour or less, double-click on the required day and time, open the context menu, choose Add Event option, and enter the required value and time limit in the displayed dialog window. If the required time is longer than an hour, click and hold the click on the beginning of the required time interval and drag until the end of it. Open the context menu, choose Add Event option, and set the required value in the displayed dialog window.

In case no specific value is set for a given time interval, the default value is applied.

The example from the figure above shows the schedule, which may be read as a room temperature profile. From Monday to Friday, from 00:00 AM to 06:00 AM the set value is 18, and then it changes from 06:00 AM to 09:00 AM to 22. From 09:00 AM to 04:00 PM the set value is 20. After 04:00 PM the default value is applied.

Editing of the schedule involves selecting the required time interval and using the context menu. The context menu offers the following options:

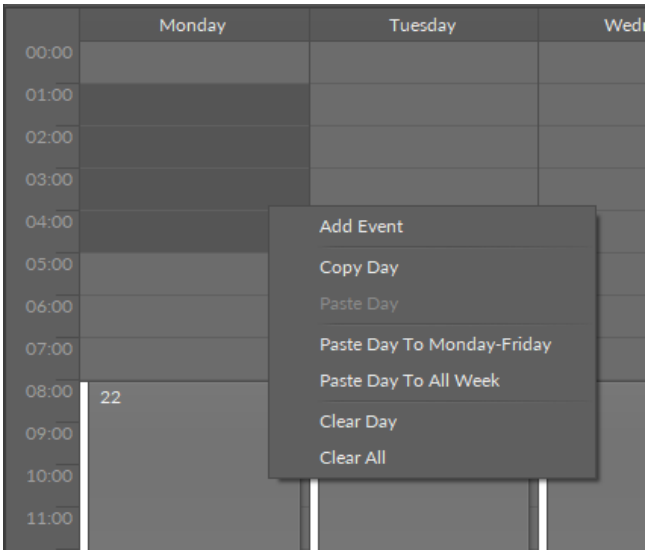



*Figure 213. A context menu in the Schedule Weekly view*

- **Add Event:** allows to create a new event, which means setting a required value to be applied in the define time. The Add Event option opens a dialog window:

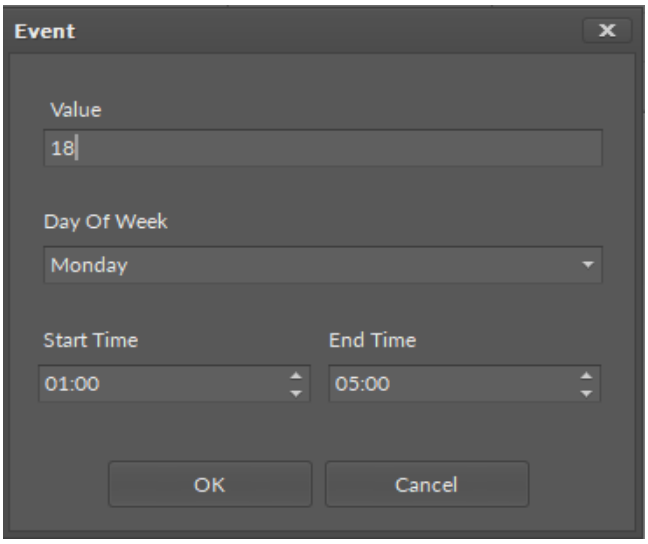



*Figure 214. The Add Event dialog window*

The Value field allows to set a required value, depending on the scheduling component type, it is a numeric or Boolean value.

The Day Of Week field allows to adjust the starting day of the selected time interval.

The Start Time and End Time fields allow to adjust the selected time interval.

The event is created after confirming with the OK button in the dialog window. Now, on Monday, from 01:00 AM to 05:00 AM the setting will be 18.

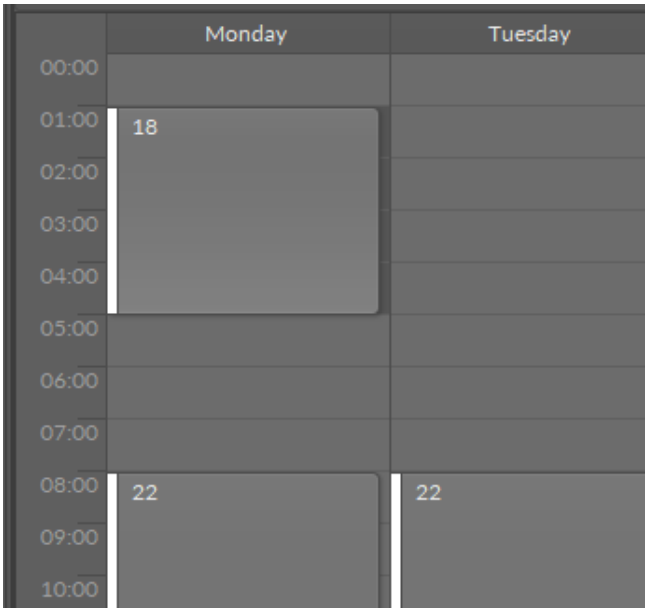



*Figure 215. New setting in a schedule*

- **Copy Day:** allows to copy all events of a day;
- **Paste Day:** allows to overwrite the selected day with the events copied from another day (Copy Day);
- **Paste Day To Monday-Friday:** allows to copy and overwrite all the events of selected day to all days from Monday to Friday;
- **Paste Day To All Week:** similar to the Paste Day To Monday-Friday option, but in this instance all days of the week will be overwritten;
- **Clear Day:** allows to remove all events of a day;
- **Clear All:** allows to remove all events from the schedule, regardless of which day the option was initiated on.

To edit of an already existing event:

- drag the event to a new time in a calendar;
- pull the upper or bottom border of the event, changing its beginning or ending hours

Note: The events cannot overlap during one day.

- double-click the event;
- open the event's context menu and choose the Edit Event option.

The selected event may be removed using Delete button on the keyboard or choosing the Delete Event option from the context menu.

Note: All changes in the schedule are sent directly to the controller and are immediately taken into account in the application's logic. It means that removing any event may result in a change of the application's behavior due to the change of the schedule output parameter value.

While using the Schedule Weekly view, additional data are displayed in the Object Properties window:

- **Current Time:** shows the current time;
- **Current Output:** shows the current output value;
- **Default Value:** allows to set a default value, which is used in case no event is planned;
- **Next Event Value:** shows the output value of the next scheduled event;
- **Next Event Time:** shows the time when the next output value will be applied.

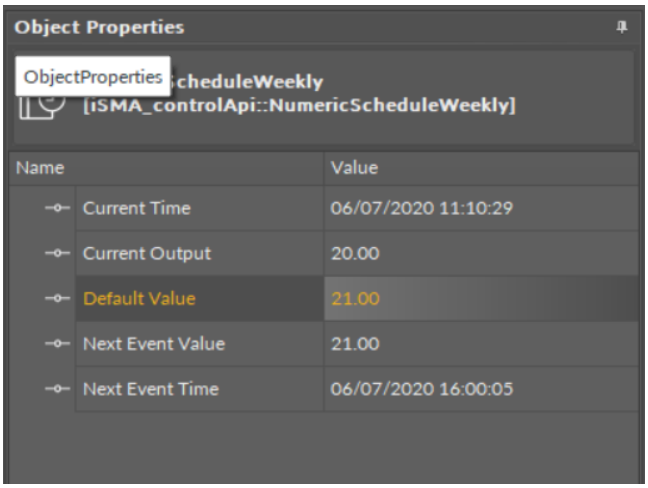



*Figure 216. The Object Properties view for scheduling components*

Note: All settings, which are available in the Schedule Weekly view and the Object Properties window, can also be edited in the Property Sheet view.

## Special Events

Apart from the Schedule Weekly view, the scheduling components have also the Special Events view available.

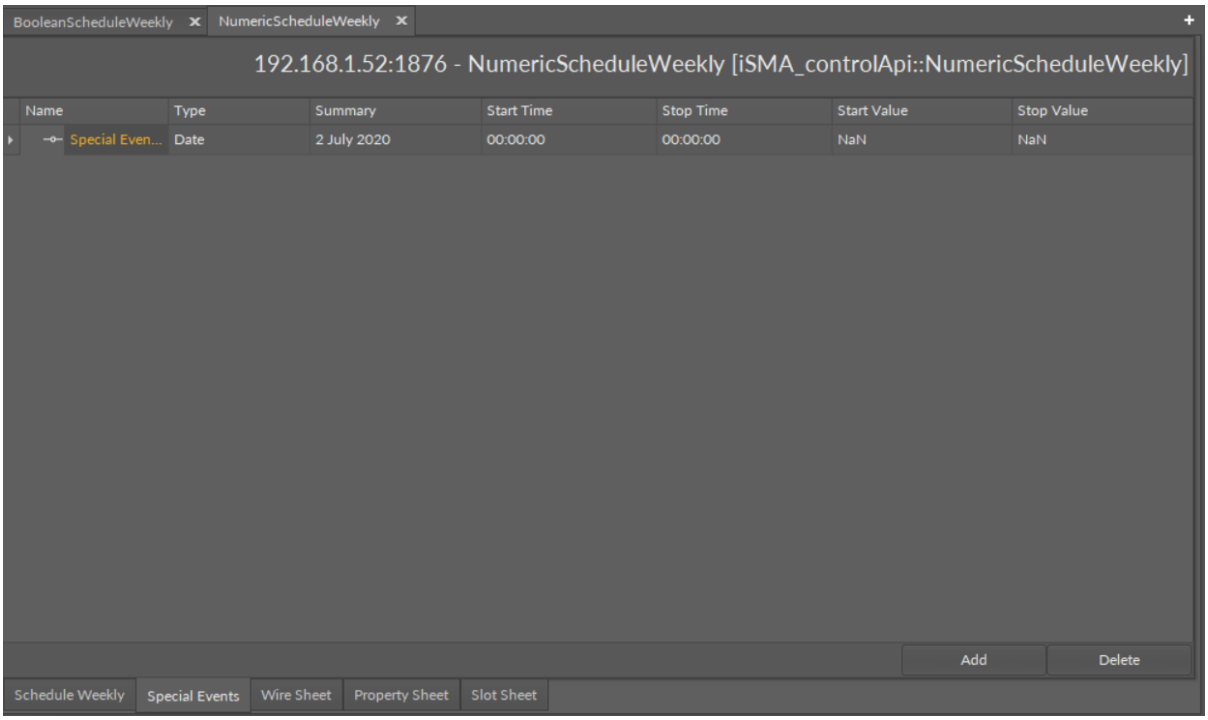



*Figure 217. The Special Events view*

The Special Events tab allows to set individual events, which are not part of a weekly schedule. These events are not displayed in the Schedule Weekly view. They are added or deleted using the Add or Delete buttons at the bottom of the tab. The settings available for a special event are the following:

- **Type:** the setting edited in the Object Properties window; the special event can be set either for a specific date, the date range, or a specific week and day (this option allows also to select month which the special event will take place in);

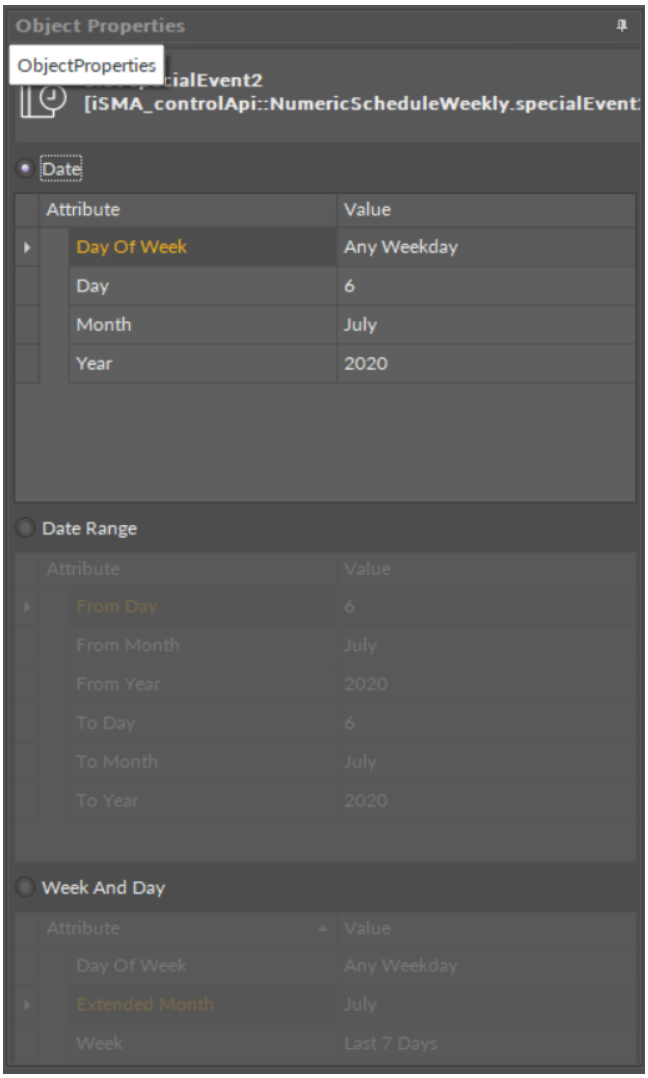



*Figure 218. Special Event in the Object Properties window*

- **Start Time:** the starting time for the special event;
- **Stop Time:** the ending time of the special event;
- **Start Value:** the starting value of the special event;
- **Stop Value:** the ending value of the special event.

### 9.3.3 Drivers Views

The iC Tool has specially designed tabular drivers views to make the work of adding and maintaining the devices working in the network and exchanging data among them as easy as possible. Predefined drivers views will be discussed based on an exemplary complex configuration of many networks and devices.

Tabular drivers views may be divided into 3 hierarchical segments:

- First segment is a table with the view of all drivers, that is networks. It is called Driver Manager. For example, the table shows the status of, among others, the following networks: localIO, ModbusAsync, OneWire, ModbusTCP, BACnet.

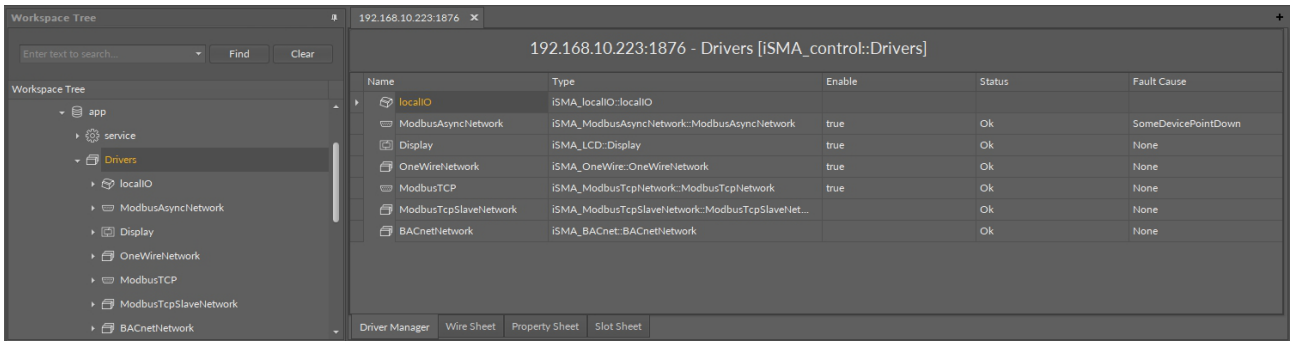



*Figure 219. The first segment of tabular drivers views*

- Second segment shows the devices included in the network selected from the first segment. In the below example, the devices are 'MIX18' and 'Mini4I4O' for the network (driver)–ModbusTCP. The name of this view depends on the type of the chosen network, here it is the Modbus TCP Device Manager view.

| Name | Enable | IP Address | Port Number | Device Address | Status | Fault Cause |
|---|---|---|---|---|---|---|
| MIX18 | true | 192.168.10.101 | 502 | 1 | Ok | None |
| Mini4I4O | true | 192.168.10.125 | 502 | 4 | Ok | None |

*Figure 220. The second segment of tabular drivers views*

- Third segment is a list of network points, which are variables allowing to read/record particular data from a particular device. In the below example, the digital points are RoomLight1-4, and they are read by the iSMA-B-4I4O device. The name of the views showing network points depends on the name of the device the data comes from, e.g., Modbus TCP Point Manager view.

| Name | Enable | Address Format | Address | Poll Frequency | Out | Status | Fault Cause |
|---|---|---|---|---|---|---|---|
| RoomLight1 | true | Decimal | 272 | Normal | false | Ok | None |
| RoomLight2 | true | Decimal | 273 | Normal | false | Ok | None |
| RoomLight3 | true | Decimal | 274 | Normal | false | Ok | None |
| RoomLight4 | true | Decimal | 275 | Normal | false | Ok | None |

*Figure 221. The third segment of tabular drivers views*

Generally, the three drivers views' segments described above may graphically be presented as follows:

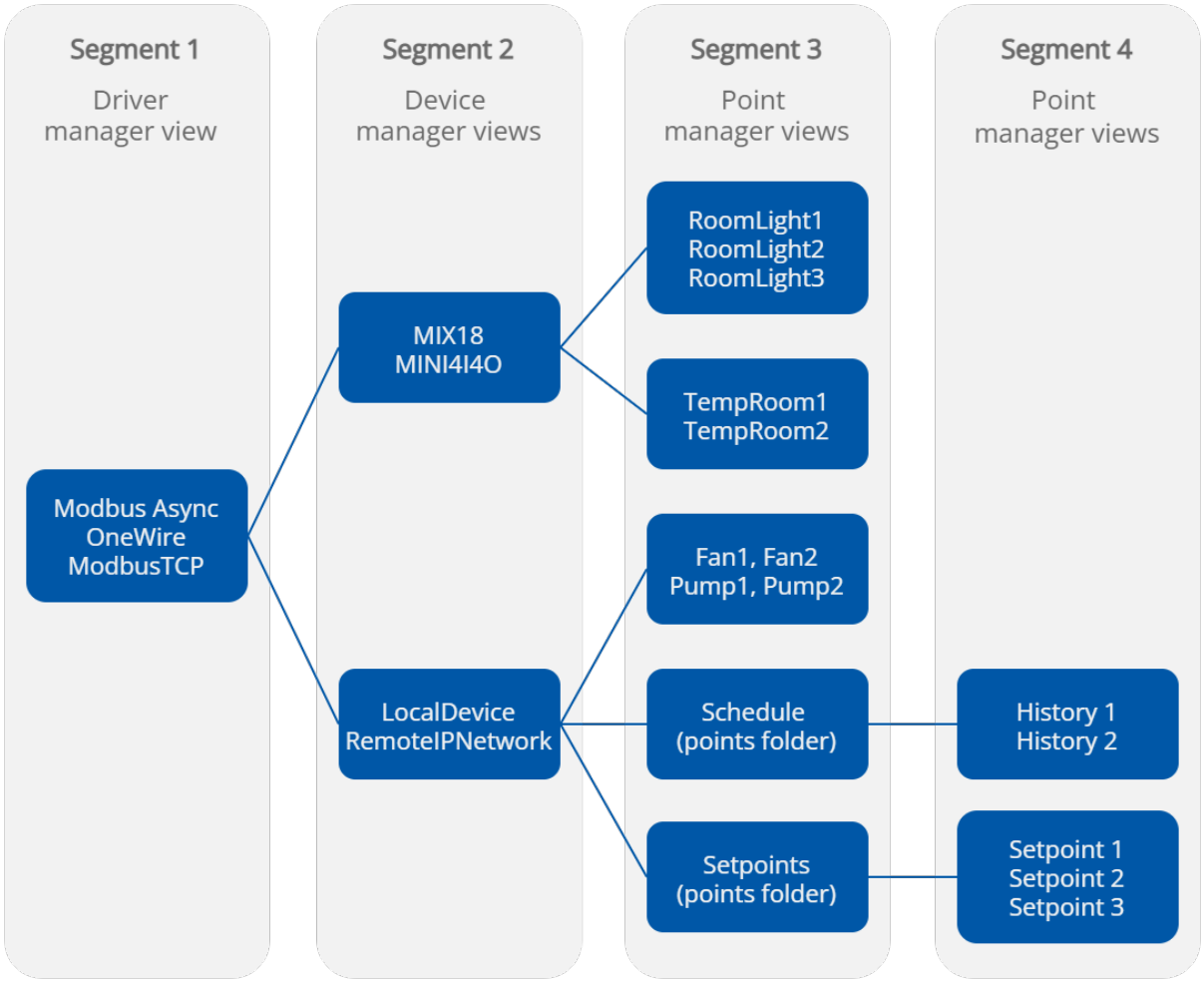



*Figure 222. Driver views segments graphically*

As depicted in the above diagram, network points may be grouped in network folders (Points Folder) and nested any number of times, which allows grouping and ordering large quantities of network points under a particular device (here, Schedule and Setpoints are folders grouping points History1,2 and Setpoint1,2,3, respectively).

The views construction of particular segments is very similar, so one type of table for each of the segments may display all the necessary data.

The next part focuses on how networks should be built based on devices distributed among Kits and points available in the iC Tool.

The further part focuses on how the XML configuration files are built and how the inbuilt tabular views for drivers views were defined. That knowledge may be used to build custom tabular views to expand drivers views with custom networks, devices, points, or to create completely new views for other application zones.

## Using Drivers Views

This section shows how to add a network, devices and points, using built-in tabular views divided into segments.

### Segment 1

To add a new network (driver) to the application:

- open a proper Driver Manager view available at app -> Drivers in the Workspace Tree;

- drag the required driver (component) representing the proper network from the Device Kits window;

Note: To make working with drivers views (but also device and point views) easier, the iC Tool filters components available to use in the Device Kits window, leaving only the ones, which can be added to the particular view. The figure below shows the open Driver Manager view with a list of available components in the Device Kits window. The list of components has been limited to these, which define networks, e.g., ModbusAsyncNetwork, ModbusTCPNetwork, ModbusTcpSlaveNetwork, OneWireNetwork, localIO, etc. The figure below shows the process of adding a driver ModbusAsyncNetwork to the Driver Manager view.

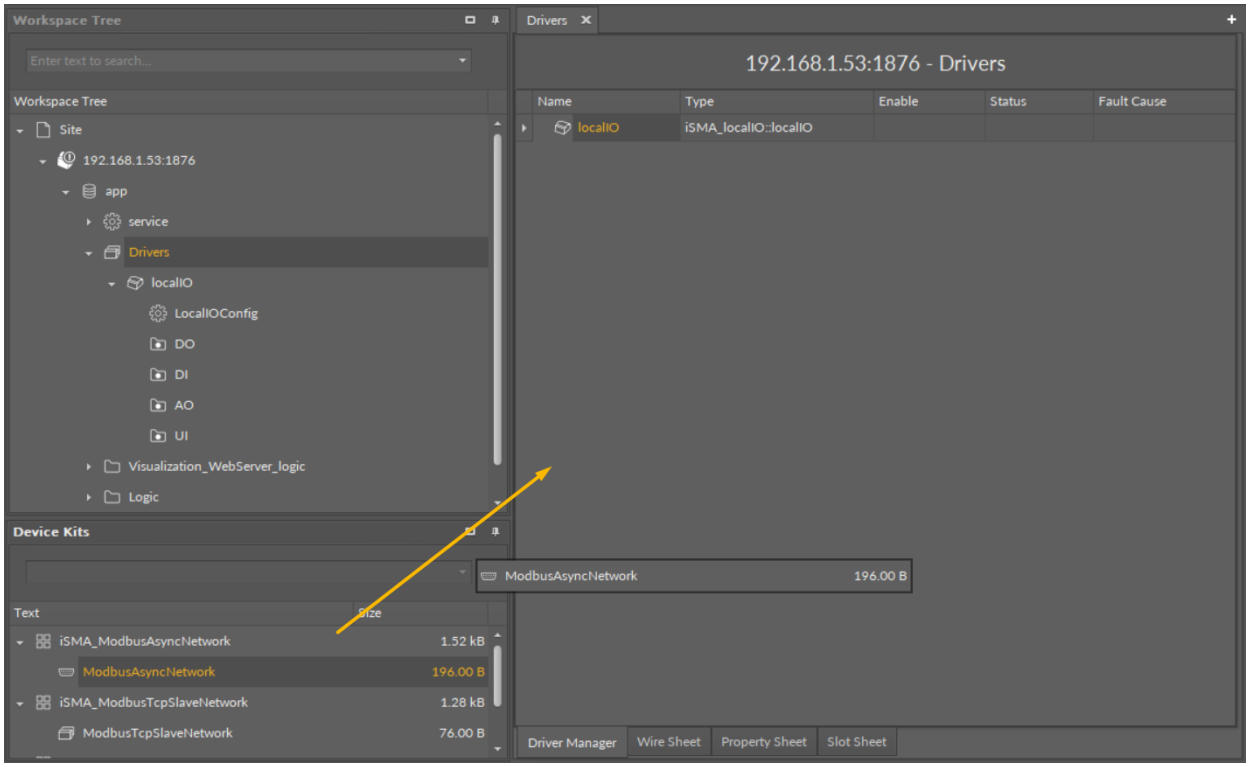



*Figure 223. Adding a component to the Drivers view*

- adjust the necessary settings for the added network–it is best to use the Object Properties window, which shows parameters of the added driver after it has been selected, see the figure below. A basic thing to set in the beginning is to enable the network using the Enable action in the Object properties window.

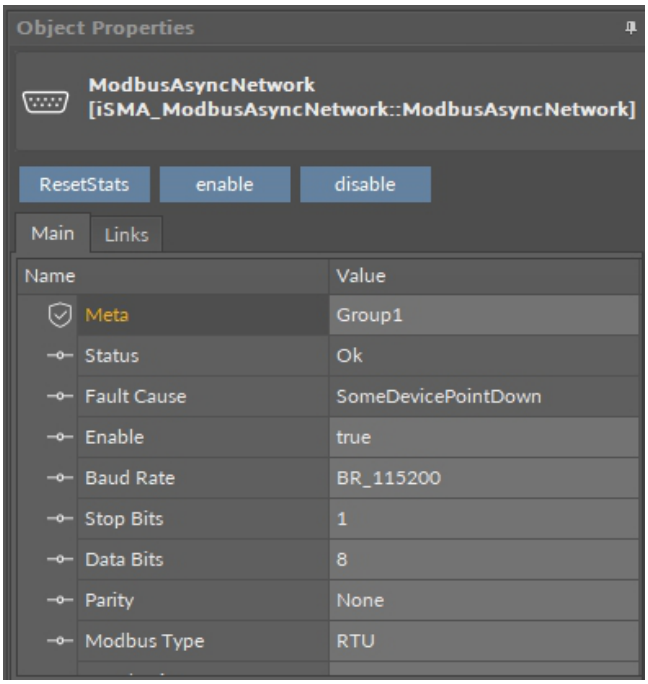



*Figure 224. The Object Properties of an added network*

**Warning!**

Inactive disabled drivers are marked with gray color, see the figure below.

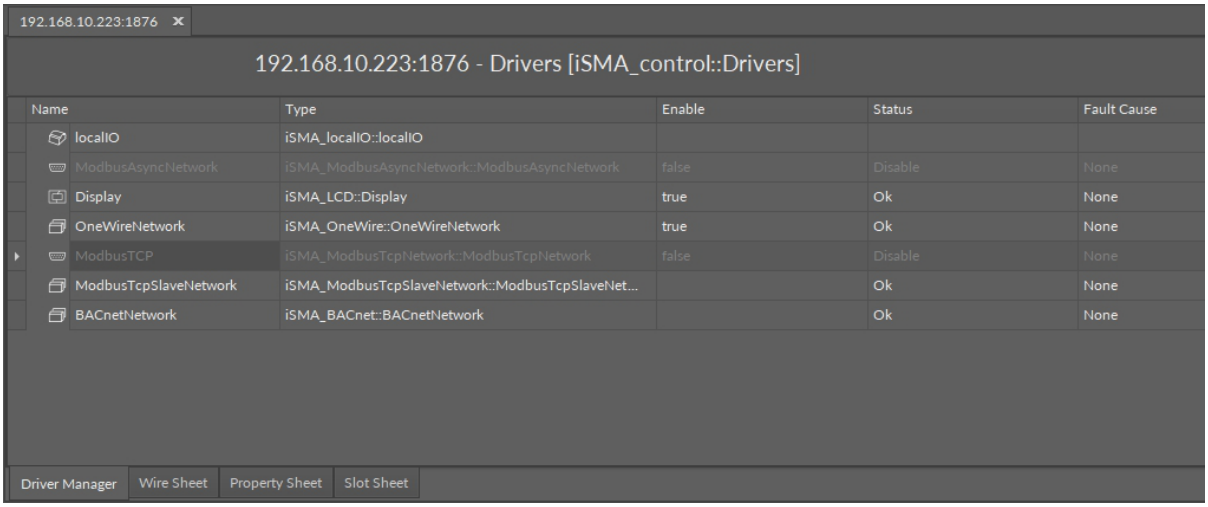



*Figure 225. Disabled network view*

## Segment 2

Next, it is necessary to add the devices to the ModbusAsyncNetwork, which are part of this network:

- double-click the added network (driver) and open the Modbus Async Device Manager view;
- drag the components representing device types installed in this network into the view. The figure below shows dragging the component representing the iSMADevice type of device.

Note: To make working with device views easier, the iC Tool filters devices, which support the selected communications network (ModbusAsync); in the example below the devices are: ModbusDevice and iSMADevice.

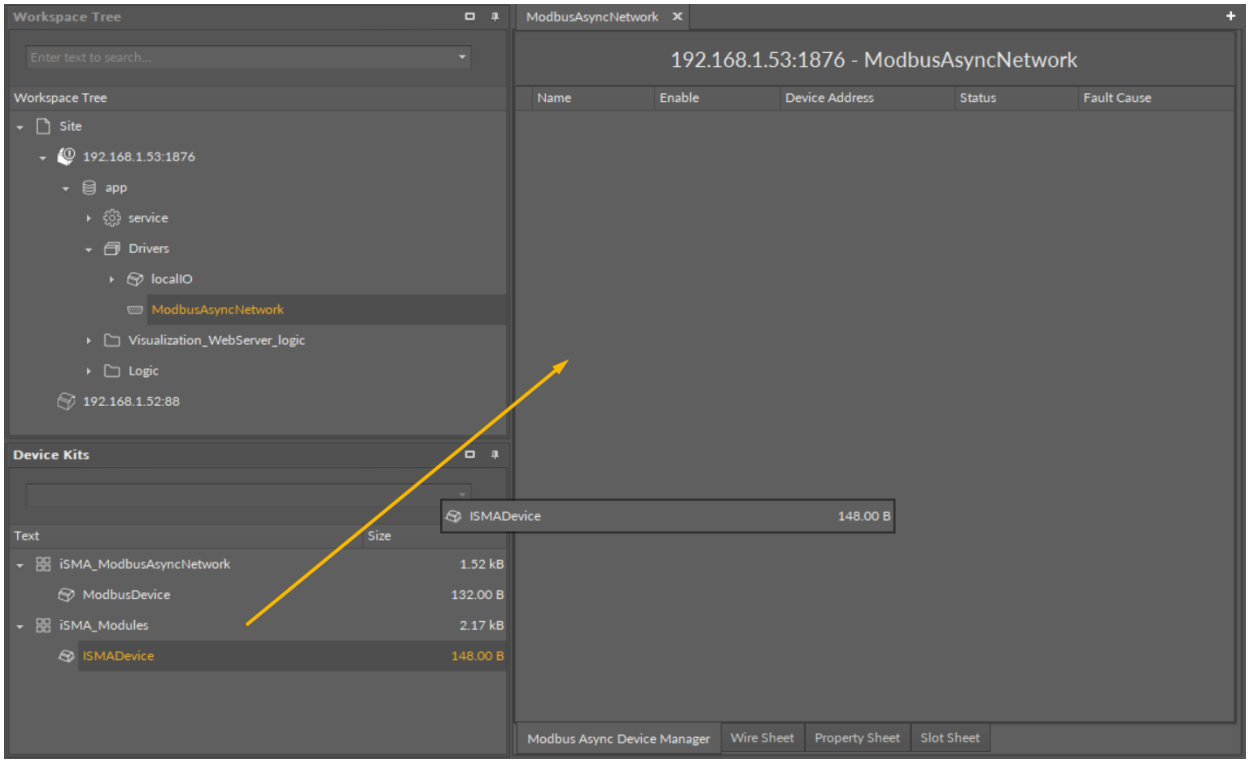



*Figure 226. Adding a device to the network*

- configure the device settings in the Object Properties window.

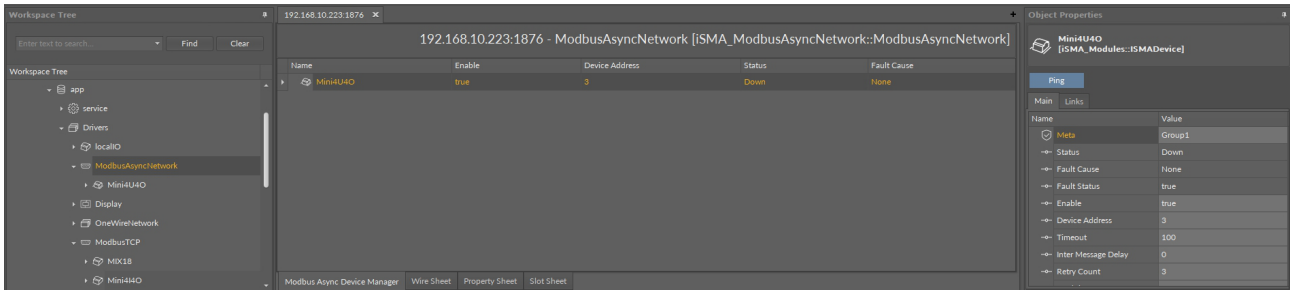



*Figure 227. COnfigure the device in the Object Properties window*

**Warning!**

Inactive devices are indicated by gray color, as shown in the figure below. To activate a device, the Enabled slot needs to be set to true using the Object Properties window.

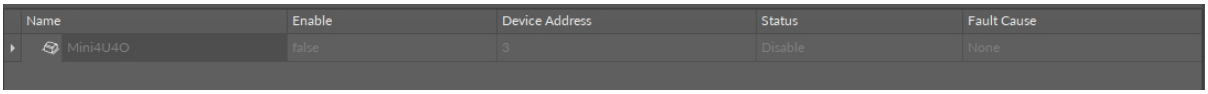



*Figure 228. Disabled device view*

Note: Red color indicates faulty devices with Fault status, see the figure below.


*Figure 229. Faulty devices view*

### Segment 3

Next, add network points to the defined device, so the application in the controller will be able to read and record data from/to an external device:

- open the Modbus Async iSMA Module Point Manager view double-clicking the freshly added device;

This time as well, the component list in Device Kits is limited to those, which cooperate with the selected device. Besides typical read and write points (here components with N and B icons), there is also a component of another type, the ModulesFolder, see the selected folder in the figure below.

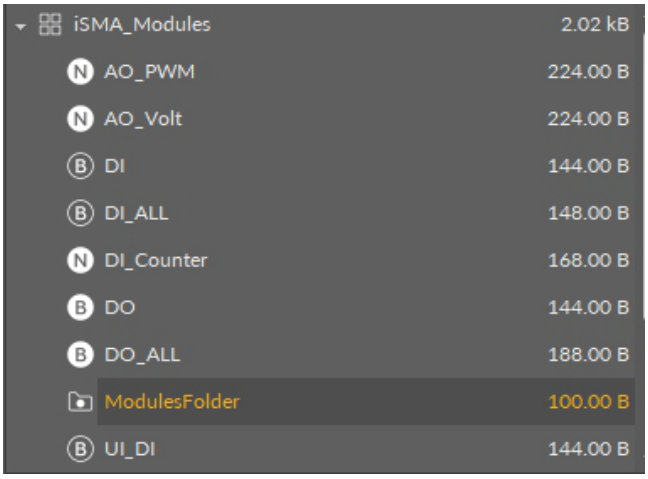



*Figure 230. The ModulesFolder container*

The ModulesFolder is a container (folder) allowing to group the network points. It may be nested in the third segment views any number of times. The figure below shows the process of adding the UI_Temp network point to the iSMA-B-4U4O device.

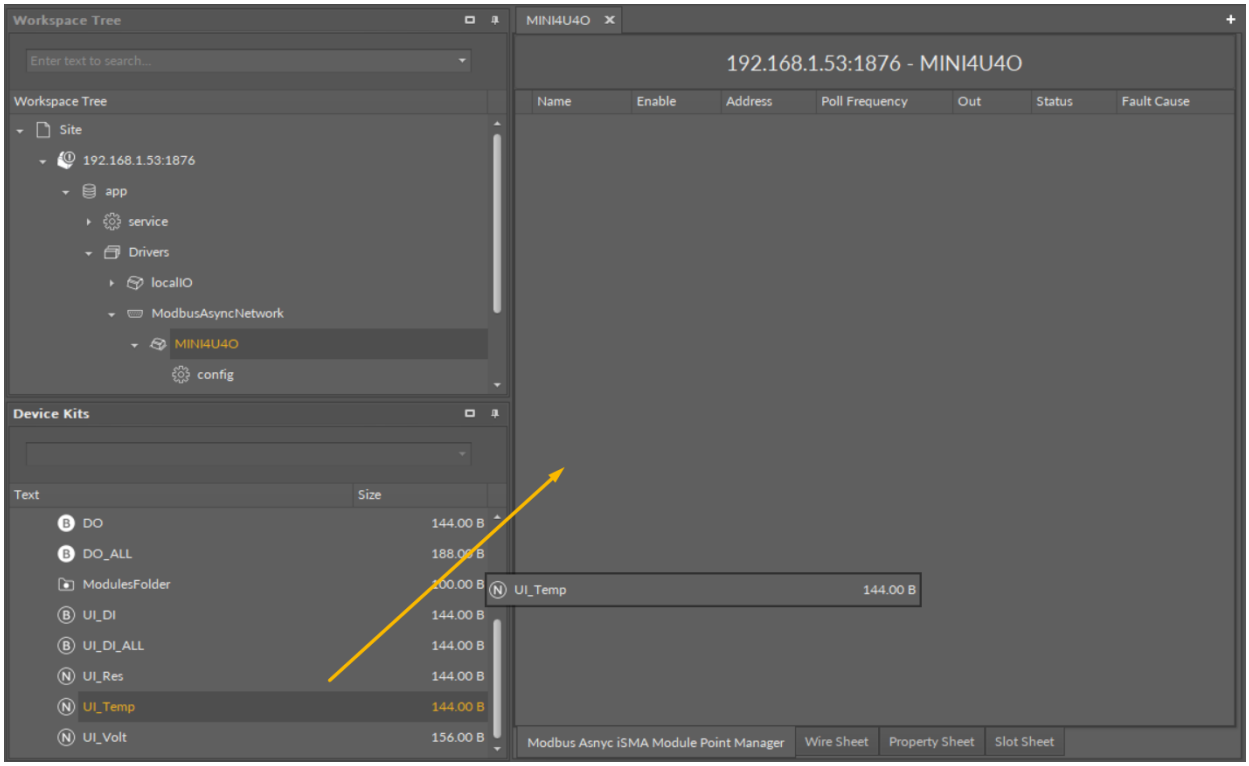



*Figure 231. Adding a network point to the device*

**Warning!**

To determine which device the particular network point belongs to, or which network it belongs to, the access path (figure below) is displayed above the Main window in the upper part of the screen. This dependency is also visible in the Workspace Tree.

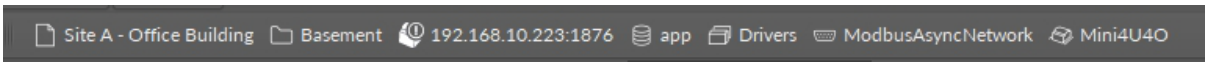



*Figure 232. Device and network points dependency*

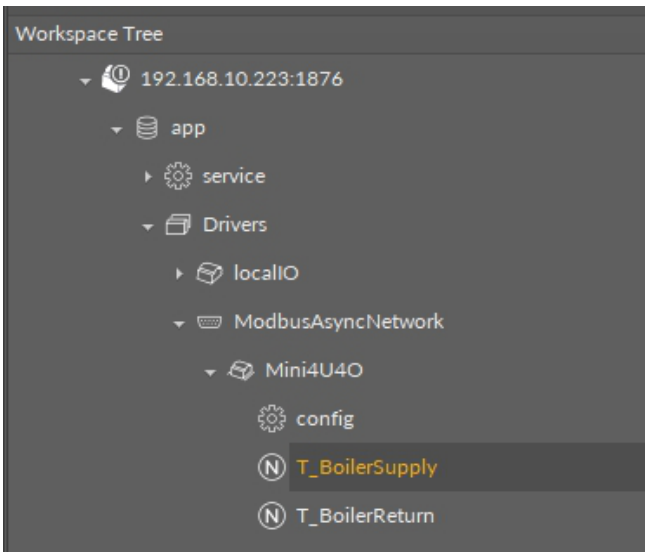
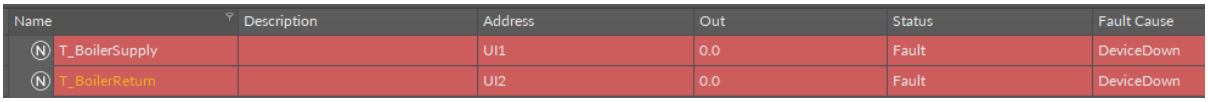



*Figure 233. Device and network points dependency*

- configure the parameters required for each of network point separately, e.g., the address, in the Object Properties window.

**Warning!**

Red color indicates faulty points with 'Fault' status, see the figure below.



| Name | Description | Address | Out | Status | Fault Cause |
|---|---|---|---|---|---|
| T_BoilerSupply | | UI1 | 0.0 | Fault | DeviceDown |
| T_BoilerReturn | | UI2 | 0.0 | Fault | DeviceDown |

*Figure 234. Faulty network points view*

After successfully configuring the network points parameters, its device, and network, which the device works in, the current value will be read by the network point.

The process described above needs to be performed repeatedly for every network, every device in network, and all required network points.

### Modbus CSV Export

For Modbus TCP and Modbus Async slave networks, an additional view is available–the Modbus Export CSV view. The view lists all points in the network with the following parameters and allows to export the list to a CSV file:

- name;
- description;
- address format;
- address;
- information if the point writable or not;
- data type;

- bit number.

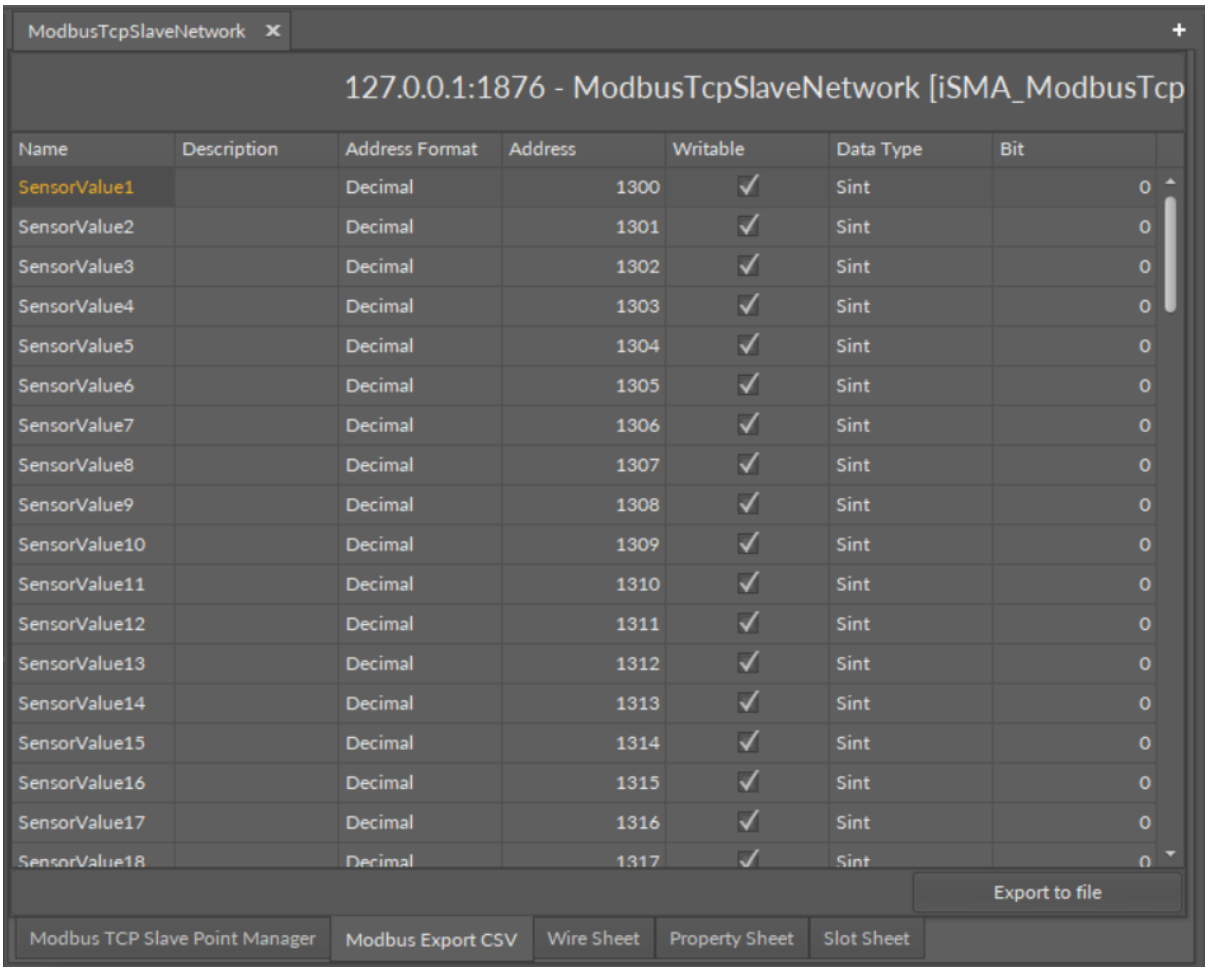



*Figure 235. The Modbus Export CSV view*

To export the list, click the Export to file button. Exported file can include all data or specific columns:

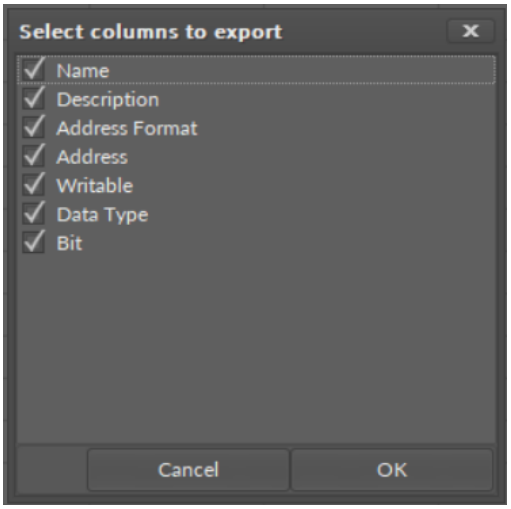



*Figure 236. Selecting columns to export*

Press OK to confirm, and the file is downloaded automatically.

## 9.4 Services

This section outlines services available for Sedona-based devices.

### 9.4.1 User Manager

The iC Tool allows to manage users of the controller with a dedicated User Manager view. It allows to define a range of changes in controller application for each user. The User Manager is opened by double-clicking the user service in the Workspace Tree window.

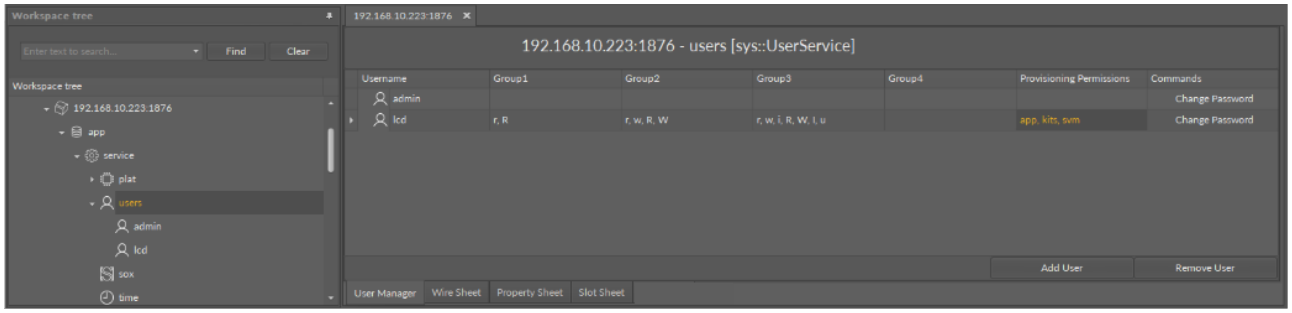



*Figure 237. The User Manager view*

The User Manager view shows the defined users, allows to edit their permissions and passwords, and makes it possible to add a new user and remove an already existing one.

The figure above shows two users defined in the controller, admin and lcd,, with different access permissions defined in table columns. The admin user has no restrictions (empty columns) and has full access to the controller, while the lcd user has defined restrictions in accordance with rules set by Sedona environment and represented by columns:

- Group1-Group4 shows permissions given to components assigned to these groups;
- Provisioning Permissions defines what services a particular user is able to use;
- Commands column has a button allowing to set a new password for a user.

Double-clicking on each column opens a pop-up window with check boxes to edit users options and permissions.

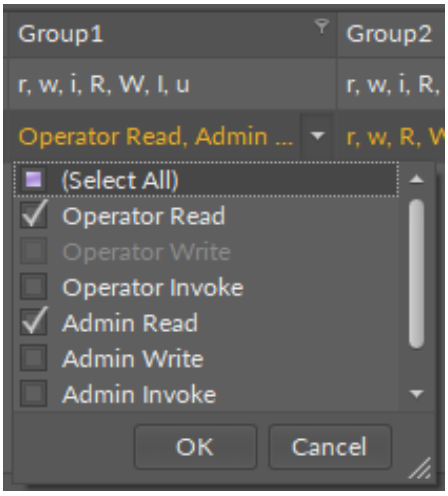



*Figure 238. Users editing options*

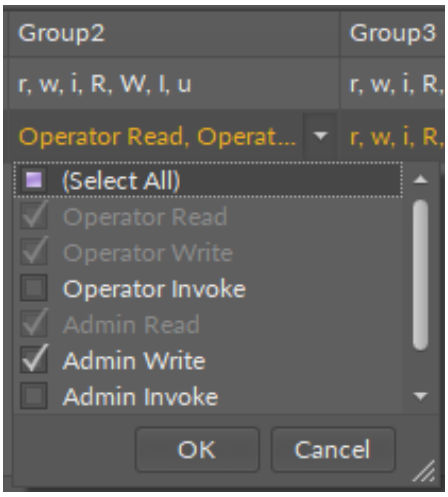



*Figure 239. Users editing options*

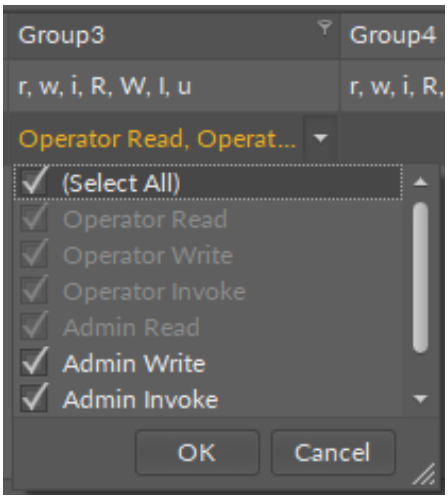



*Figure 240. Users editing options*

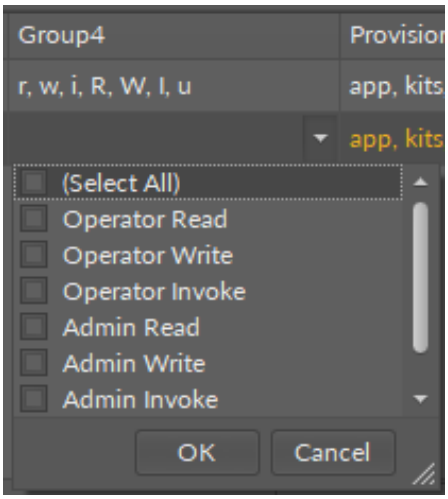



*Figure 241. Users editing options*

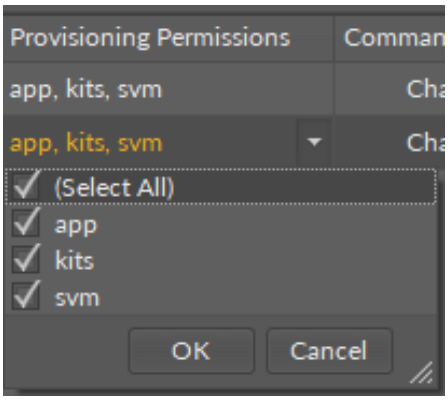



*Figure 242. Users editing options*

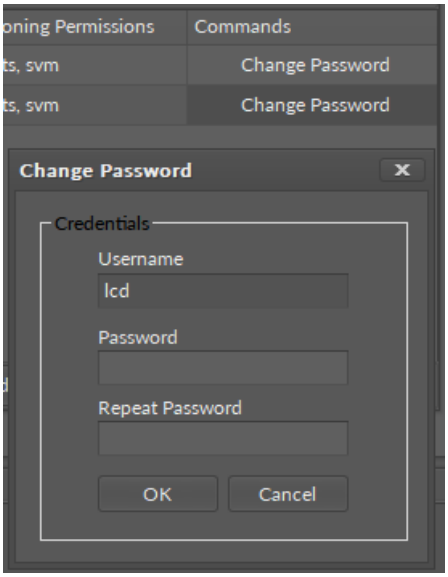



*Figure 243. Users editing options*

At the bottom of the User Manager tab are two buttons to add and remove users. In order to add a user, enter a username and a password. Adding a new user in the iC Tool will automatically create a component named the same as the username and expand the user list in the User Manager view.

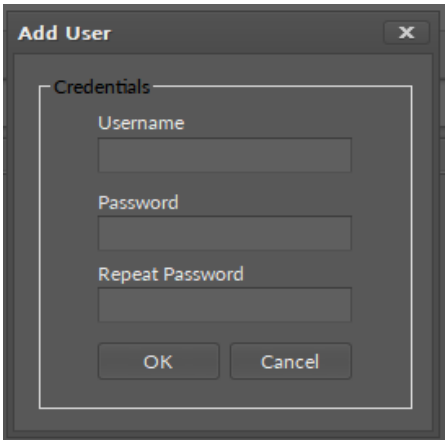



*Figure 244. Dialog window for adding a user*

Note: It is not recommended to change the username (component's name) after setting a password. If it is necessary to change the username, a new password should be set for it.

Note: In case a component representing a user is duplicated to create a new user with the same permissions, new user should also have a new password set.

Removing of a particular user may be done using two methods: either by removing the row in User Manager view, selecting it and pressing the Remove user button, or by removing the component representing that user.

### 9.4.2 Historical Data

The iC Tool allows to perform an analysis of historical trends by using two views, History Chart and History Table. The first view allows to work with a single trend or many trends on a graph versus time. The second view is a table of registered samples of one of the processes.

Accessing both views is possible from the Workspace Tree at app -> service -> history -> db1. Under the db1 component, which is a historical database, there are historical variables registered in the iC Tool shown by components (see the figure below).

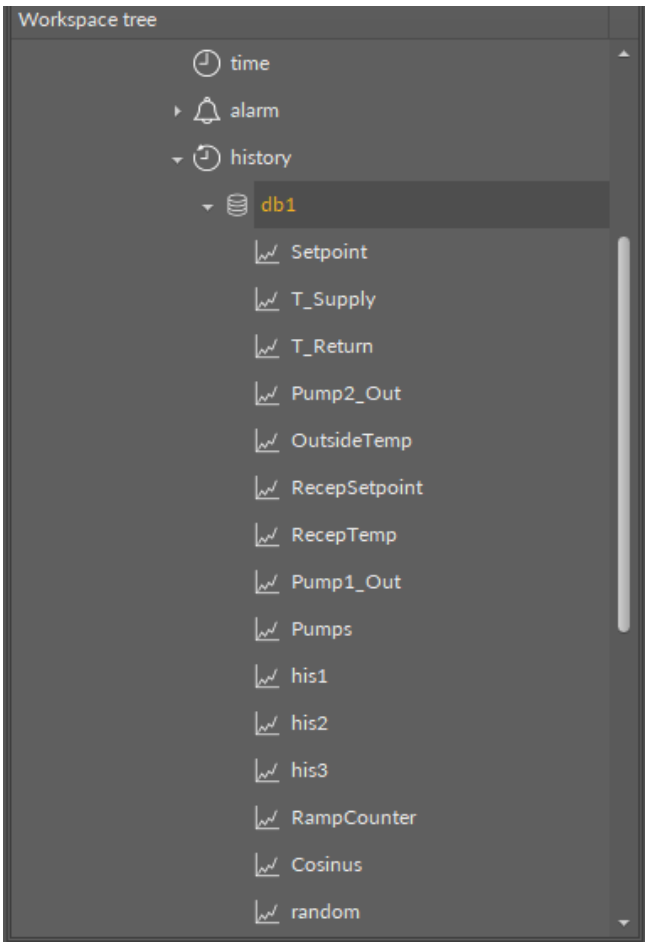



*Figure 245. The db1 component's tree*

Opening any component under the db1 component will open the selected component in the main screen along with the History Chart and History Table.

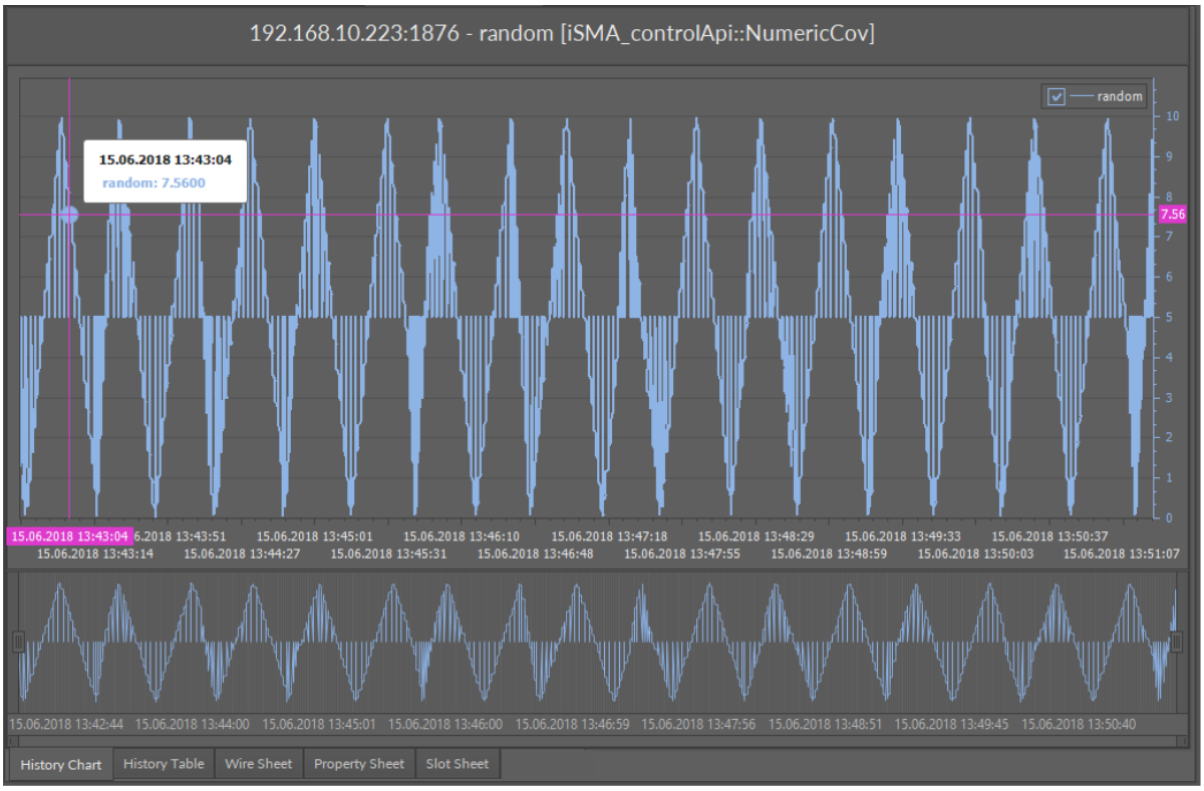



*Figure 246. The History Chart view*

The second method of working with historical data is expanding the db1 component and dragging the component we are interested in from the Workspace Tree to an empty History Chart view. The figure below shows the empty History Chart view of the db1 component before selecting the component to present its historical data.

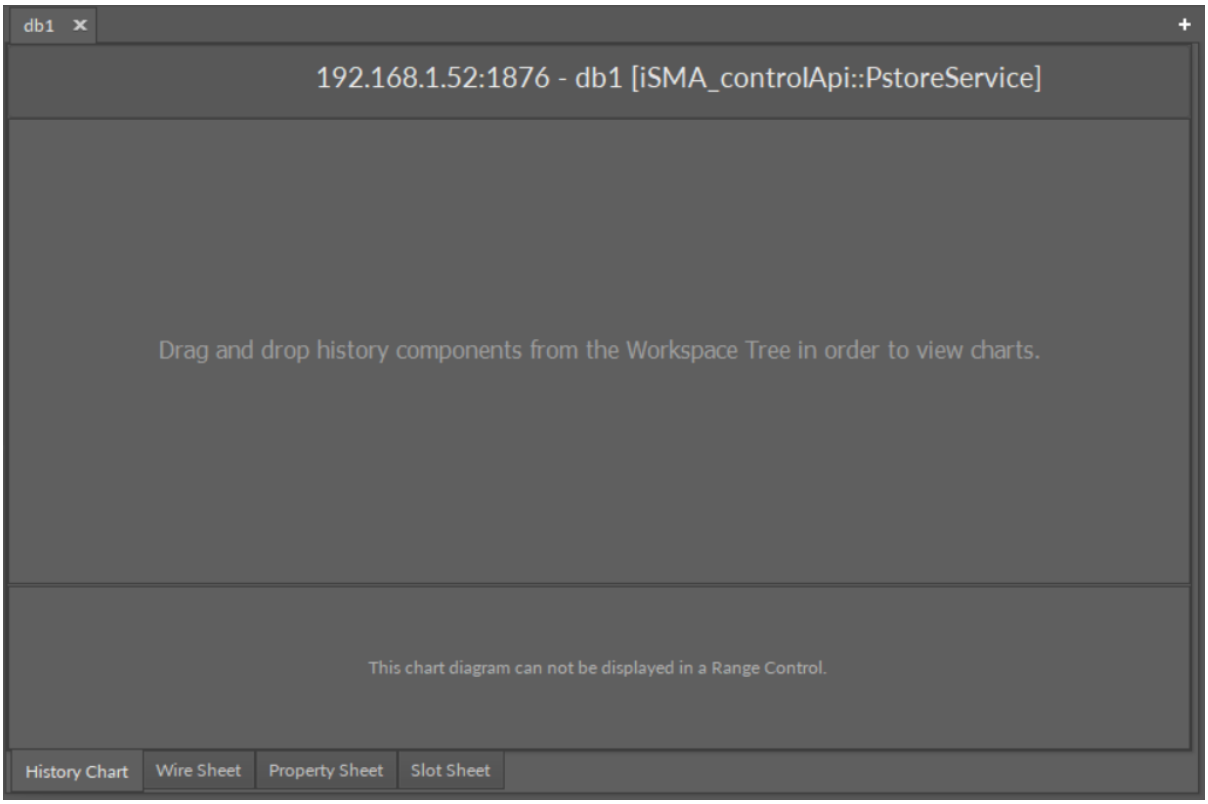



*Figure 247. An empty History Chart*

## Exporting and Printing Program (Preview)

The Preview program is launched with the Export/Print option in the graph's or history table context menu. Exporting or printing the graph or table requires defining few of the parameters described below.

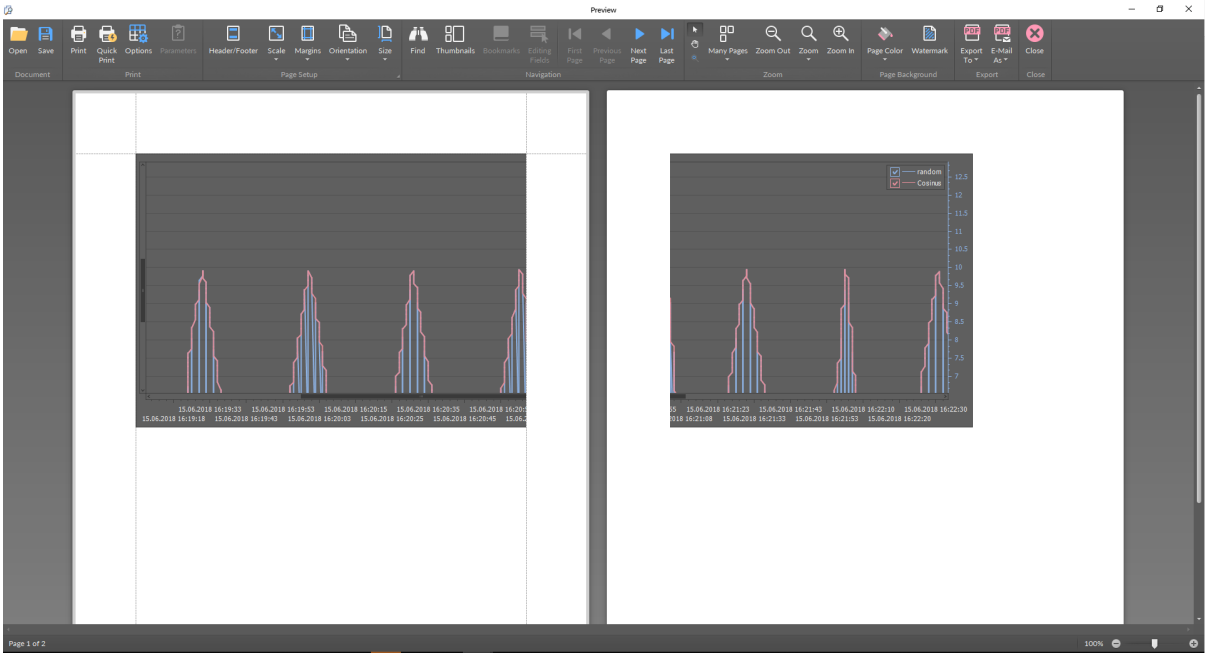


*Figure 248. The Preview program*

The program toolbar offers functions to adjust the graph to be exported or printed. The basic editing options are the following:

- **Header/Footer:** allows to define a header or footer for the document;
- **Scale:** allows to scale the contents to a page size;
- **Orientation:** changes page orientation to horizontal or vertical;

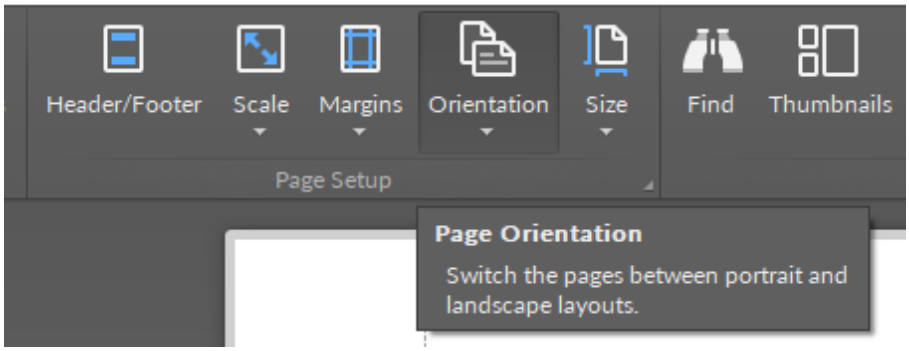



*Figure 249. The Orientation button on a toolbar*

- **Size:** adjusts margins and the page size
- **Page Color:** allows to set the background page color;
- **Watermark:** allows to add the watermark to the page background.

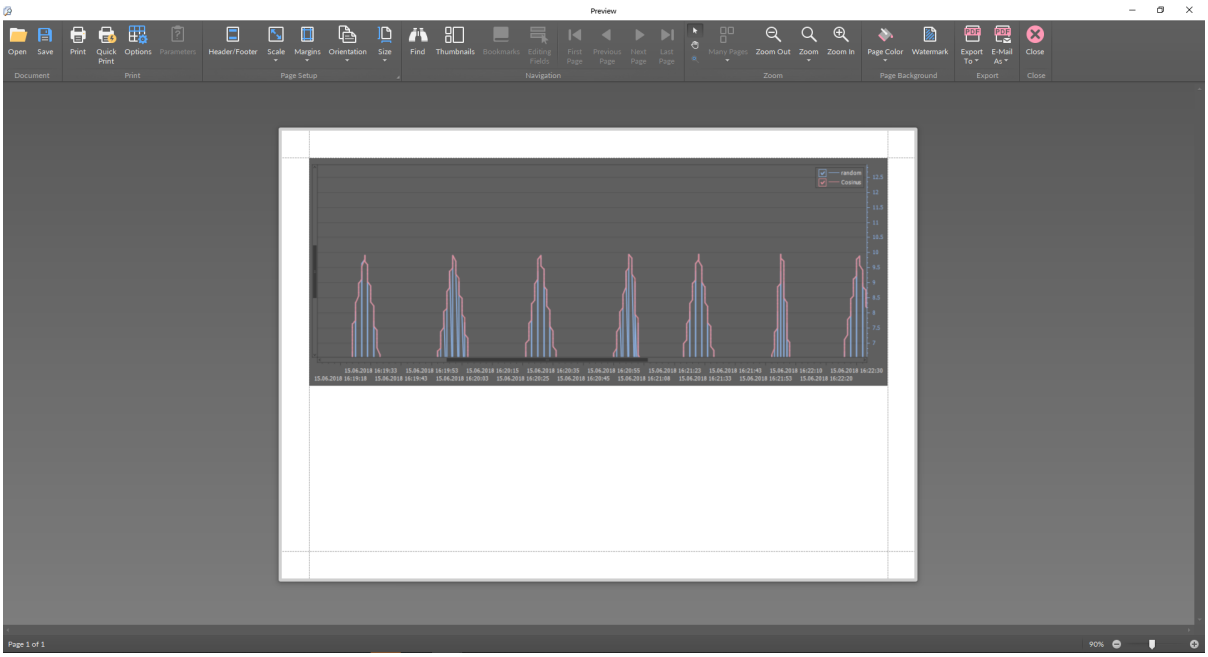



*Figure 250. The graph adjusted to exporting/printing*

Once the document is edited and adjusted to be printed or exported, the toolbar includes further actions buttons:

- **Open:** allows to open a previously saved file to be exported or printed;
- **Save:** allows to save a file for further action in available formats: .pdf, .html, .docx, .xlsx, .bmp, .gif, .jpeg, .png, .tiff, .emf, and .wmf;
- **Print:** launches printing and allows to choose the printer and its parameters;
- **Quick Print:** launches printing with default printing settings;
- **Export To:** launches exporting the file to available formats: .pdf, .html, .docx, .xlsx, .bmp, .gif, .jpeg, .png, .tiff, .emf, and .wmf;
- **Email As:** exports the file and attaches it to an e-mail message.

## History Chart View

The History Chart view is a view dedicated for a graphic presentation of historical data versus time, commonly called a graph. A graph is a line connecting subsequent samples, defined in a time interval and a range of values. The graph is located in the upper and main area of the view. Below there is a smaller area with a preview of the whole available range saved in the controller.

In the preview area there is a possibility of setting time limits against which the trend will be drawn in the main area above.

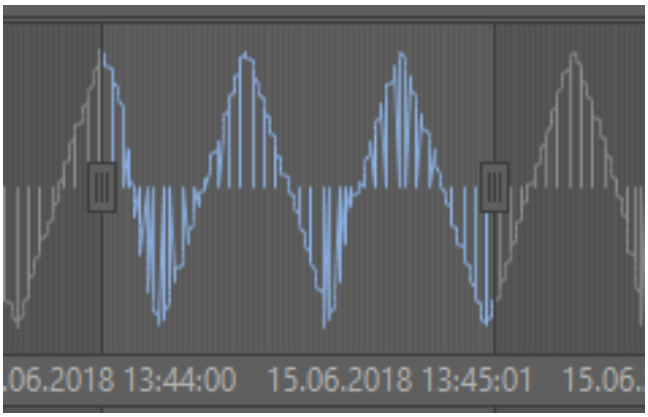



*Figure 251. Setting time limits*

After setting the time limits, the corresponding graph has been significantly narrowed and looks as shown on the figure below.

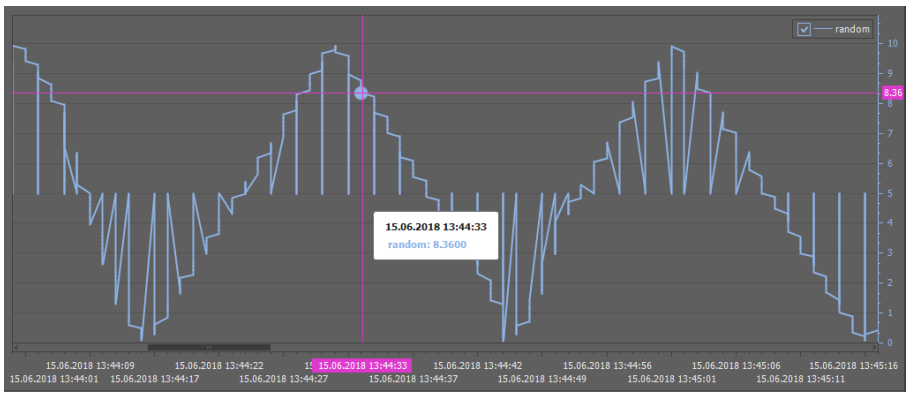



*Figure 252. A graph after setting time limits*

To analyze values on the History Chart move the mouse pointer over the registered samples–it shows their precise values. Additionally, the auxiliary lines show values on vertical and horizontal axes of the graph.

Clicking and holding the click allows to move the graph, while using the mouse wheel allows to zoom in/out the fragment of a graph. Each change of the graph's range is updated in the preview area under the graph.

It is possible to add another trend to the observed chart and display them both simultaneously for analysis. To add another trend it needs to be dragged from the Workspace Tree, from under the db1 tree to the opened chart. The figure below shows a situation after adding the RampCounter trend.

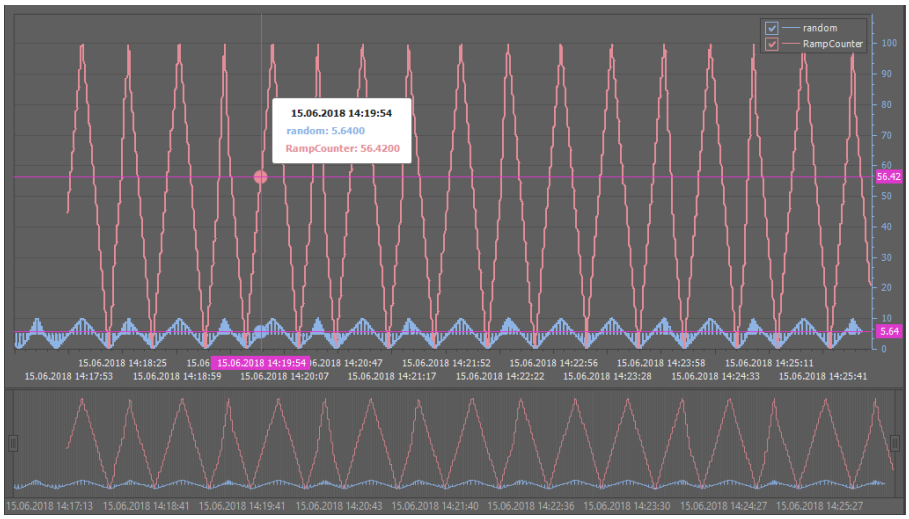



*Figure 253. Two trends displayed in the History Chart*

After adding another trend, the upper graph and the preview area have been updated accordingly. As the two trends are displayed simultaneously, the auxiliary lines track and show registered samples for the two trends independently.

By default, after adding a second trend the value scale is common for both processes, but individual scales can be displayed for each of historical processes. This is allowed by the Enable/Disable Secondary Axes option in the context menu of the graph.

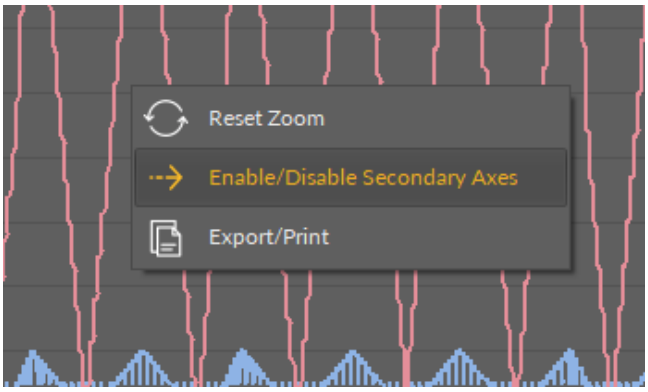



*Figure 254. A context menu of a graph*

After initiating an individual oy axis, for each of the processes, trends are automatically rescaled to cover as big part of the graph area as possible (see the figure below).

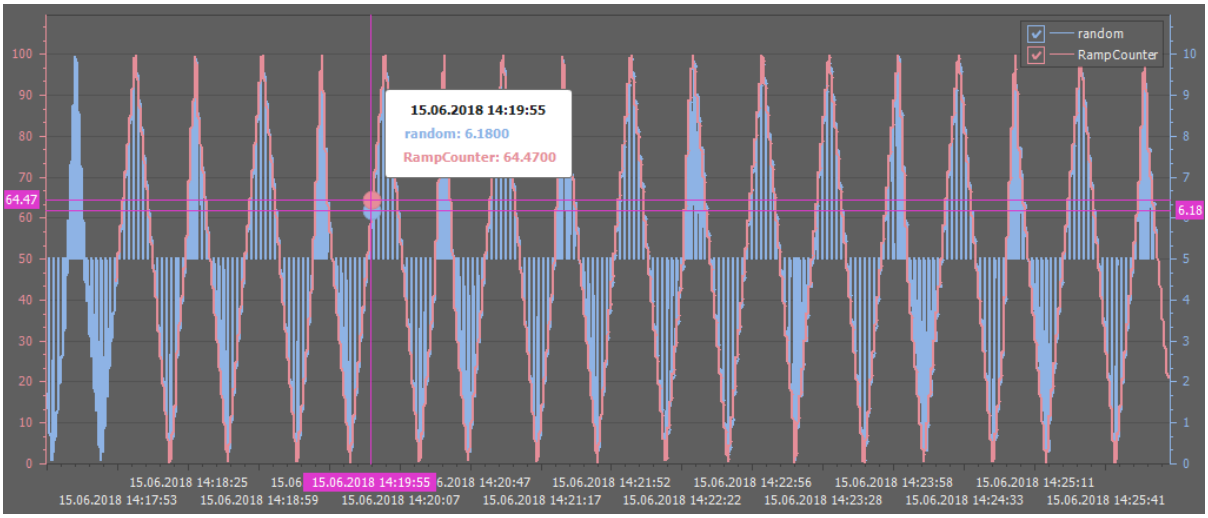



*Figure 255. A rescaled graph*

A function of temporary hiding of historical trends is available when working with many trends. In order to temporarily hide the trend, deselect it on the key placed in the upper right corner of the graph.

The remaining options of the context menu allow to return to the default values of zoom with the Reset Zoom function, and to export or print the graph with the Export/Print function.

Initiating the Export/Print function starts a Preview application allowing to print and export the displayed graph. The offered export formats are .pdf, .html, .docx, .xlsx, .bmp, .gif, .jpeg, .png, .tiff, .emf, and .wmf. A detailed description of the function is available in section Exporting and Printing Program–Preview.

## History Table View

The History Table view shows all saved samples for one historical component in a tabular form. As in case of the History Chart view, the table can be printed or exported using the Preview program. The Preview program is initiated from the context menu in this view and its description is available in section Exporting and Printing Program–Preview.

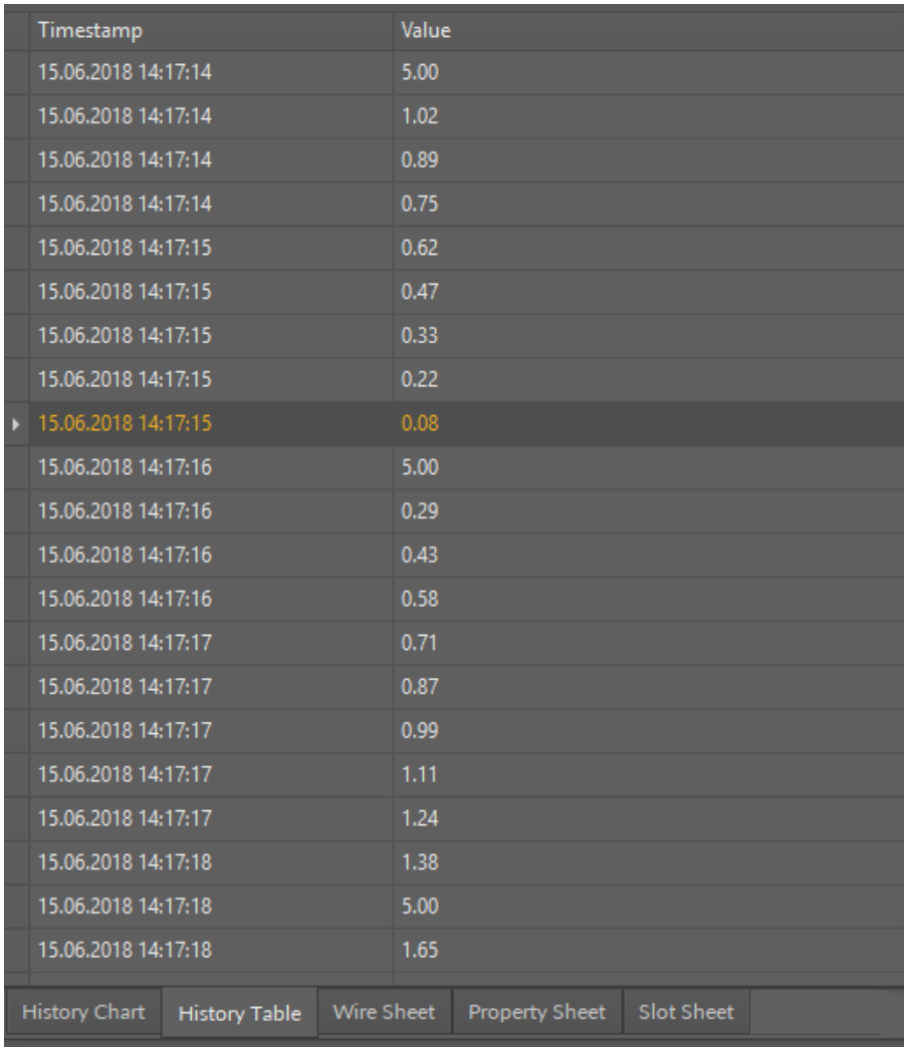



*Figure 256. The History Table view*

### 9.4.3 DateTime Service

The iC Tool allows to precisely set the time in the connected controller, including the time zone and a potential change of time connected with daylight saving. For this purpose there is a special DateTimeService view, accessible by double-clicking the time component available at app -> service -> time in the Workspace Tree window.

The DateTimeService shows the following data:

- current time of the controller;
- desired time, which is a setting applied to the controller when saved;
- daylight saving time, which sets an automatic time change on a specific day of the year.

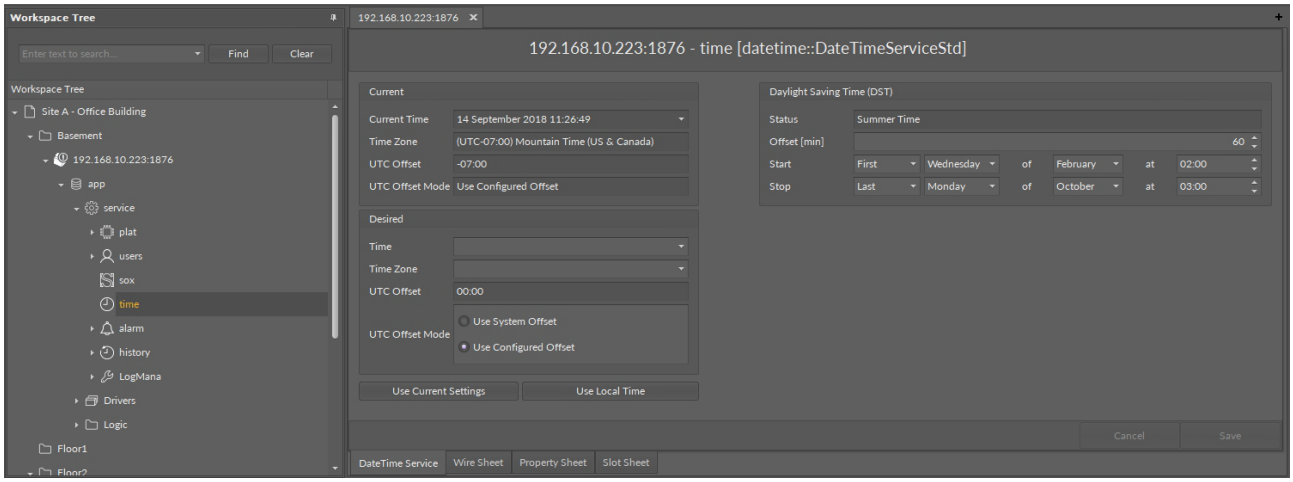



*Figure 257. The DateTime service*

## Setting Time and Time Zone

The time and time zone are set in the Desired panel. The new settings are sent to the controller after pressing the Save button in the lower right corner of the view.

To set time in the controller, the time and time zone need to be set up using one of three methods:

- manually enter the time and time zone expanding Time and Time zone fields;
- read the current time and time zone of the controller by pressing the Use Current Settings button, and then edit it;
- read the current time and time zone of the computer operating system in which the iC Tool is running by pressing the Use Local Time button, and then edit it.

Using one of the three methods allows to set the time and time zone ready to be sent to the controller, and it is indicated by a blue color of the edited settings (see the figure below).

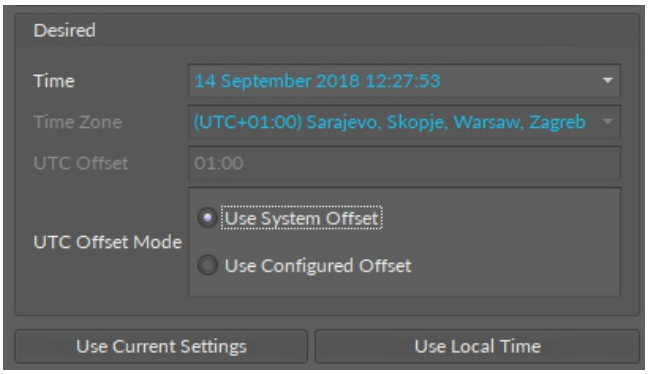



*Figure 258. Editing time settings*

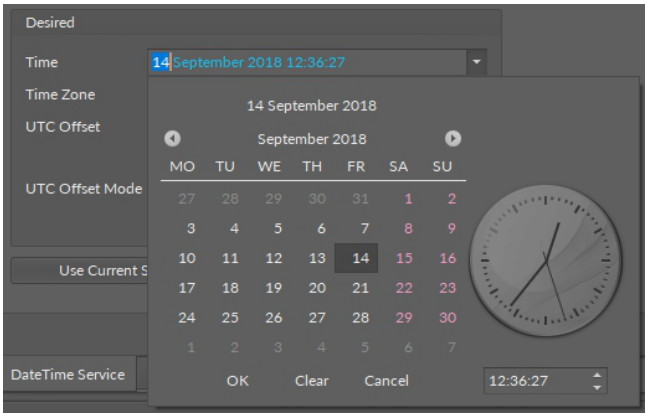



*Figure 259. Editing time settings*

After pressing the Save button, the settings sent to the controller return to their initial colors.

**Warning!**

System will not allow setting time before year 2000.

## Daylight Saving Time

The Daylight Saving Time (DST) panel on the right side of the DateTimeService view is used for setting the daylight saving time, which is an automatic time change on a specific day of the year.

**Warning!**

If changing the time due to the daylight saving is not required (e.g., the controller is used in the country that does not use the daylight saving time), 0 should be entered in the Offset field.

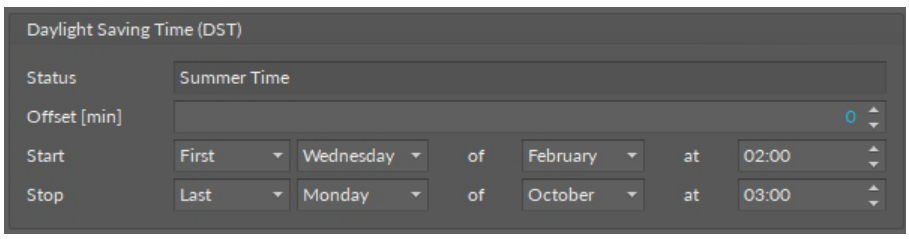



*Figure 260. No Daylight Saving Time settings*

When the settings for the beginning and end of daylight saving time are entered, the changed parameters are indicated with a blue color. New settings are sent to the controller with the Save button. When the data are successfully sent to the controller, the color returns to the default one.

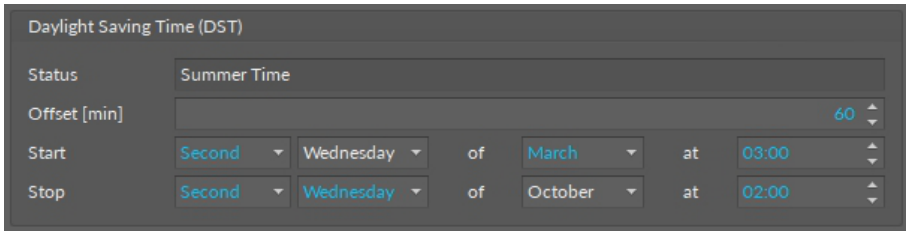



*Figure 261. New settings to DST*

**Note:** In case the DST data are not saved while leaving, the iC Tool shows a warning, which should be accepted in order not to lose the entered data.

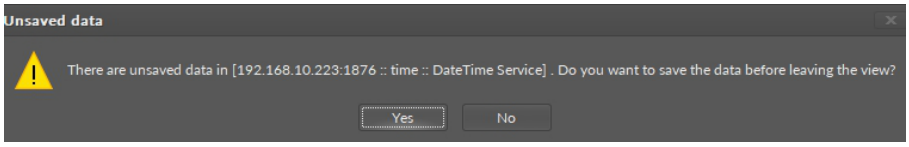



*Figure 262. The unsaved data warning*

**Warning!**

Sending the settings to the controller does not mean that the time data are permanently saved in the controller. After sending the data with the Save button, it is necessary to additionally save it in a non-volatile flash memory of the controller by clicking the floppy disk icon on a toolbar in the upper part of the screen.

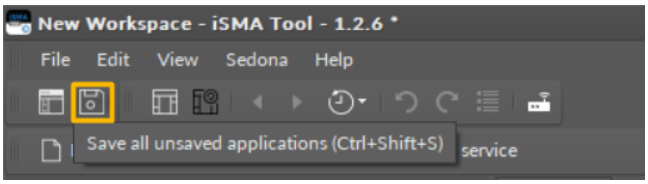



*Figure 263. Full saving process is confirmed by clicking a floppy disk on a toolbar*

## 9.4.4 Alarm Service

The iC Tool provides a summary table of all the alarms registered in the controller. This table is presented in the Alarm Table view available at app -> service -> alarm -> alarmDb.

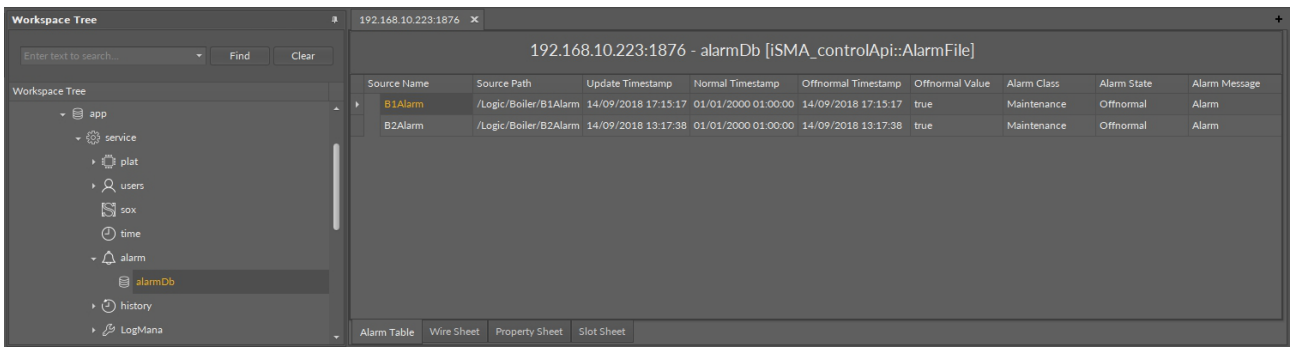



*Figure 264. The Alarm service*

Once the Alarm Table view is opened, the iC Tool downloads a file with the registered alarms from the controller and displays its contents in a table. The table columns are defined as follows:

- **Source Name:** shows the name of the component, which generated the alarm;
- **Source Path:** shows the source component's path;
- **Update Timestamp:** shows the time of the last update;
- **Normal Timestamp:** shows the time of switching to normal state after the alarm;
- **Offnormal Timestamp:** shows the time of switching to alarm state;
- **Offnormal Value:** shows the alarm value;
- **Alarm Class:** shows the alarm class;
- **Alarm State:** shows the alarm status;
- **Alarm Message:** shows the alarm message.

**Warning!**

The Alarm view is a non-editable view and its contents are read from the alarm file only while the view is loading.

## 9.5 Custom Tabular View

The iC Tool allows each user to define individual tabular views for any component by defining an individual .xml file.

In such case the Tabular view is available beside main views of each component, which are the Wire Sheet, Property Sheet, and Slot Sheet.

The Tabular view filters subordinate components (children) making it much easier to work with complex applications composed of many components (children) of different types.

In a situation when a folder-type component (parent) has many different components, under it, including, for example, NV components, the Tabular view facilitates searching for the required components. Without the Tabular view it would be necessary to identify required components among many other components, and view their values in the Object Properties window. By defining one .xml file properly it is possible to create an individual Tabular view for each type of component in the iC Tool (it will list only the required type of components). For example, such view would be filter all NV components.

iSMA CONTROLLI wants the user to be able to create tabular summaries of components according to his individual needs, without the need to delegate it to iSMA CONTROLLI.

Each view created that way will require following characteristics to be defined:

- Tabular view name;
- Types of components for which the tabular view will be displayed;
- Types of components, which will be displayed in tabular view along with their common slots.

The figure below shows an example of a tabular view filtering NV components. While working with a view defined this way (here named 'NVValuesManager'), the iC Tool limits the number of components available in the Device Kits window, making it easier to add proper components to the view. After switching to other view (choosing a tab for other views), the Device Kits window will return to normal working mode displaying all components available in the device.

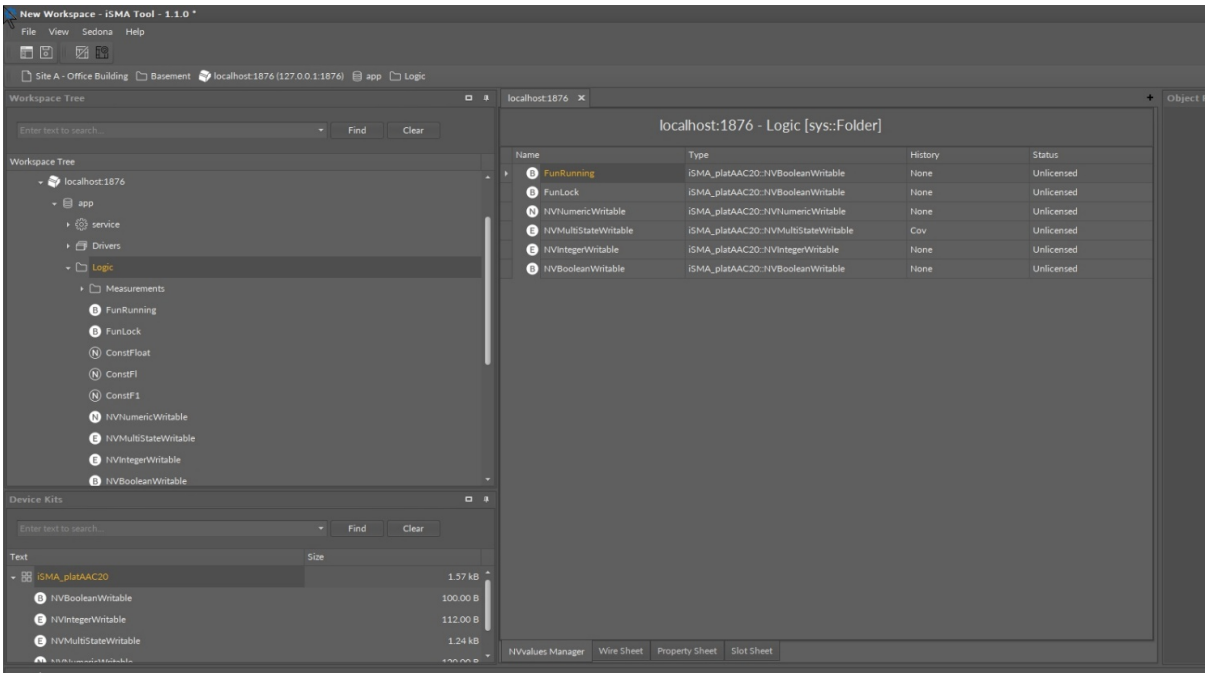



*Figure 265. Custom table view*

The iC Tool provides properly configured tabular views for all the devices from the iSMA family, which are described in detail in the next chapter.

### 9.5.1 Creating Custom Tabular View

The iC Tool allows to define the view contents as well as compatible (fitting) components through configuring a proper .xml file. This approach allows the iC Tool user to build required views for any type of component in the application.

The .xml configuration files need to be placed in the iC Tool main folder under /home/CustomTableViews folder.

By default the folder includes the following files created for tabular drivers views:

- one drivers.xml file–defining all drivers (networks) available in the iC Tool;
- [driver name]DeviceManager.xml or [driver name]NetworkManager.xml files–defining the look for a particular driver (device) view;
- [Driver name]PointManager.xml files–defining the look of network points for a particular device.

The .xml files have the following tags, which meaning is necessary for proper construction of any tabular views:

- <ViewName>Driver Manager</ViewName>–defines the name of the view connected to the settings in remaining tags;

```
<ViewName>Driver Manager</ViewName>
```

*Figure 266. Creating custom table view*

- <ForcomponentTypes>–defines the names of components, for which the particular view will be used. This means that for components defined within this tag, one of the assigned views will be a view defined by <ViewName> tag;

```
<ForComponentTypes>
  <ForComponentType>iSMA_control::Drivers</ForComponentType>
</ForComponentTypes>
```

*Figure 267. Creating custom table view*

- <Filtercomponents>–defines the names of components, which will appear in the table on the defined view and limits the number of components shown in the Device Kits window;

```
<FilterComponents>
  <FilterComponent>iSMA_BACnet::BACnetNetwork</FilterComponent>
  <FilterComponent>iSMA_DALI::DALINetwork</FilterComponent>
  <FilterComponent>iSMA_localIO::localIO</FilterComponent>
  <FilterComponent>iSMA_LCD::Display</FilterComponent>
  <FilterComponent>iSMA_ModbusAsyncNetwork::ModbusAsyncNetwork</FilterComponent>
  <FilterComponent>iSMA_ModbusTcpNetwork::ModbusTcpNetwork</FilterComponent>
  <FilterComponent>iSMA_ModbusTcpSlaveNetwork::ModbusTcpSlaveNetwork</FilterComponent>
  <FilterComponent>iSMA_obix::Obix</FilterComponent>
  <FilterComponent>iSMA_OneWire::OneWireNetwork</FilterComponent>
  <FilterComponent>iSMA_p2p::P2pNetwork</FilterComponent>
  <FilterComponent>iSMA_MBus::MbusIpNetwork</FilterComponent>
  <FilterComponent>iSMA_MBus::MbusLocalNetwork</FilterComponent>
  <FilterComponent>iSMA_platFCU::SlaveNetwork</FilterComponent>
  <FilterComponent>iSMA_platFCU::localIO</FilterComponent>
  <FilterComponent>iSMA_BACnetMasterSlave::BACnetMasterSlaveNetwork</FilterComponent>
</FilterComponents>
```

*Figure 268. Creating custom table view*

- <Columns>–a group of settings allowing to read particular slots from components assigned to particular view along with the name replacing the name of the slot;

```
<Columns>
  <Column slotName="name" slotCaption="Name"/>
  <Column slotName="type" slotCaption="Type"/>
  <Column slotName="Enable" slotCaption="Enable"/>
  <Column slotName="status" slotCaption="Status"/>
  <Column slotName="faultCause" slotCaption="Fault Cause"/>
</Columns>
```

*Figure 269. Creating custom table view*

The following actions need to be taken to extend the offered tabular drivers views with custom networks, devices; and points:

- In 'drivers.xml' file extend the list <Filtercomponents> with component type defining new network (driver) unless such network has already been entered.
- Duplicate any existing '.....DeviceManager.xml' file to define the devices view and optionally change the file name.
- Open newly created file, change the name of the view under the tag <ViewName> and properly edit the definition inside the tags: <ForcomponentTypes>, <Filtercomponents> and <Columns>.
- Duplicate any existing '.....PointManager.xml' file to define the points view for a particular device and optionally change the file name .
- Open newly created file, change the name of the view under the tag <ViewName> and properly edit the definition inside the tags: <ForcomponentTypes>, <Filtercomponents> and <Columns>.

Close and restart the iC Tool, as the iC Tool checks drivers views definitions only on start-up.

**Warning!**

A thorough description of tags is also available in the drivers.xml file. Getting to know it will allow proper configuration of this and other .xml files.

**Warning!**

The iC Tool reads all .xml configuration files for tabular views only during start-up, and if an error is found in the .xml file syntax, it will display a proper information in the Console.

## 9.6 iSMA-B-AAC20 Simulator

Thanks to the iC Tool software, the iSMA-B-AAC20 device can be programmed in the offline mode using the iSMA-B-AAC20 simulator.

The latest simulator can be downloaded at https://ismacontrolli.com.

Windows compatibility – Simulator is an application intended for Windows operating system only, and can be run on Windows 7, 8, and 10 series.

### 9.6.1 Running App on a Simulated Device

The Sedona Device Simulator tool lets to run a Sedona app on the iC Tool as a simulated device using the Sedona device vendor-supplied simulator, SVM (Sedona Virtual Machine).

This procedure describes how to run an app in the Sedona Device Simulator.

#### Start

- Run a "run" bat file from a Simulator folder to launch the simulator program:

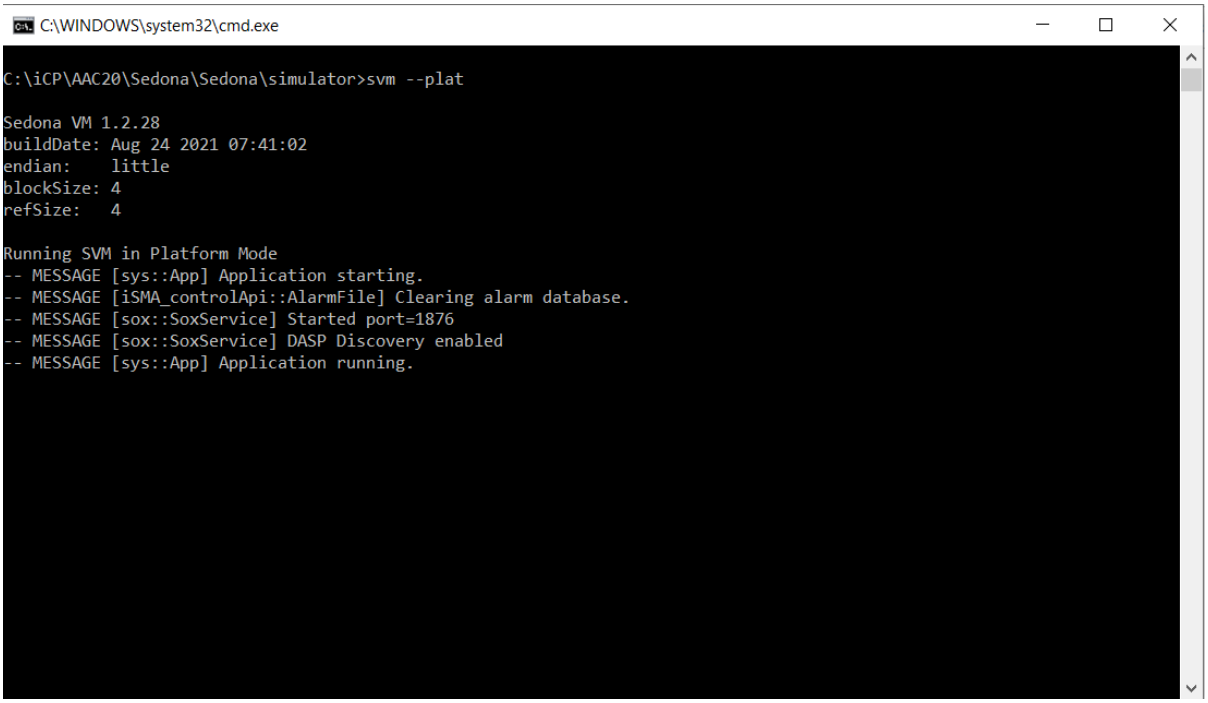



*Figure 270. Launching the simulator*

- Open a direct Sox connection using a localhost in the iC Tool. Select Add Device.

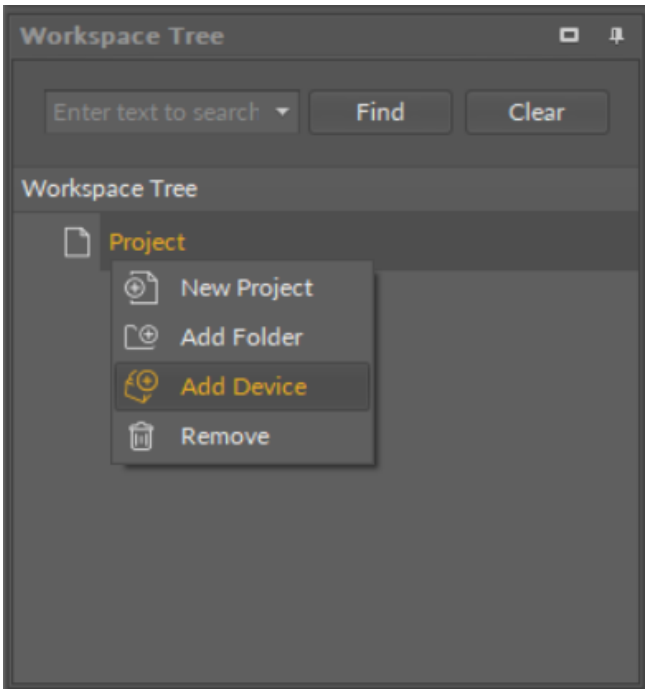



*Figure 271. Add device*

- In the Connect dialog box, for Host IP, enter: localhost. In the Authentication window, in the Username and Password fields, enter the credentials used in the new app file that has been created, and click Finish.

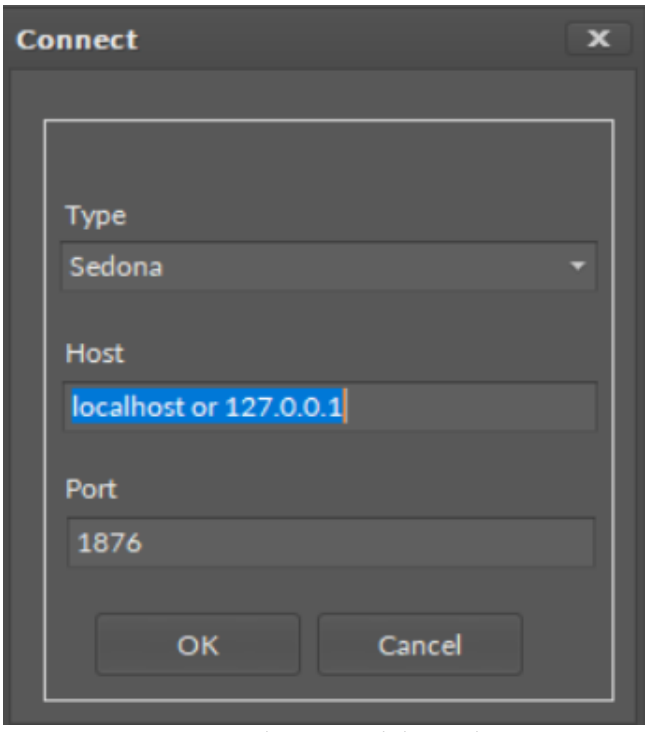



*Figure 272. The Connect dialog window*

If the sample app is used on the sample simulator, the default credentials should be entered: Username: admin, Password: <blank>.

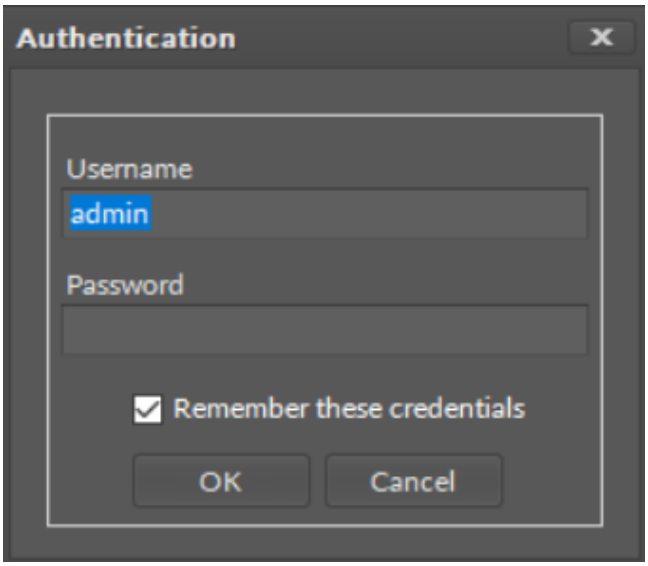



*Figure 273. The Authentication dialog window*

- Once connected, a direct Sox session is established with a connection to the app running in the simulated device:

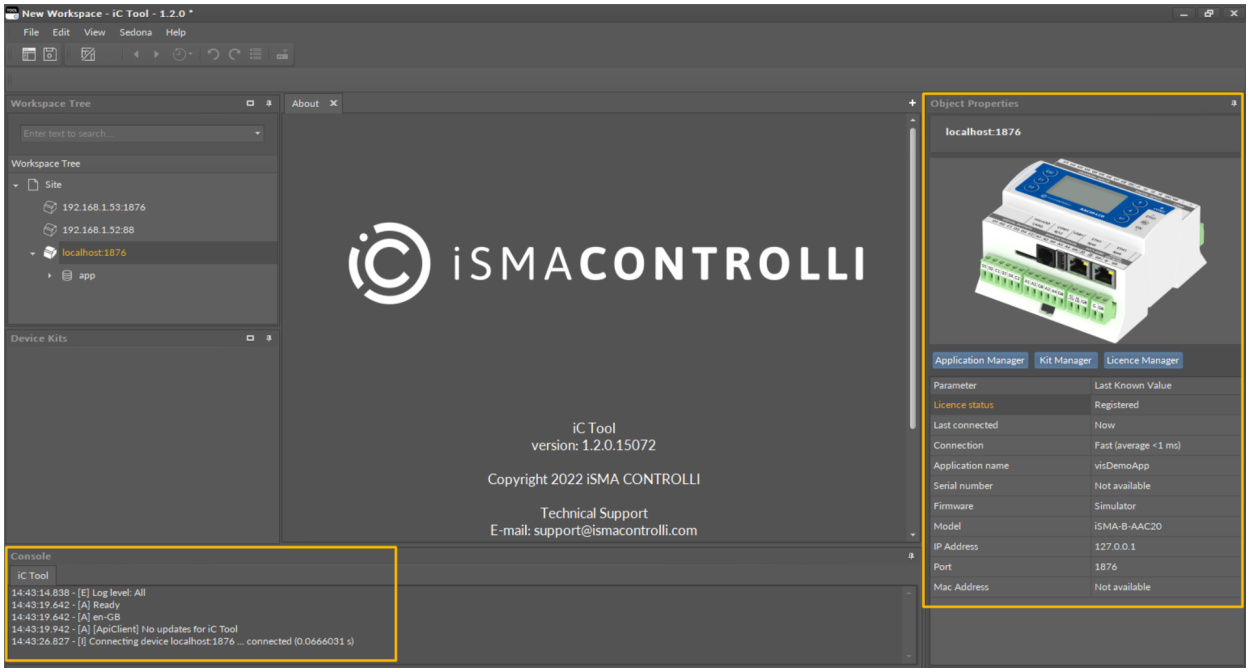



*Figure 274. Connected simulator*

## 9.6.2 Making Changes to the App Running on a Simulated Device

This procedure provides an example of making changes in the app running on a simulated device using the Kit Manager tool under the device to add a kit to the app.

### Requirements

- The app must be successfully running in the device simulator.
- There is a running Sox connection to the app currently executed on the simulated device.

## Steps

- In the Objects properties, click the Kit Manager to launch the tool.
- In the Kit Manager view, click the checkbox to select the kit, and click Update.

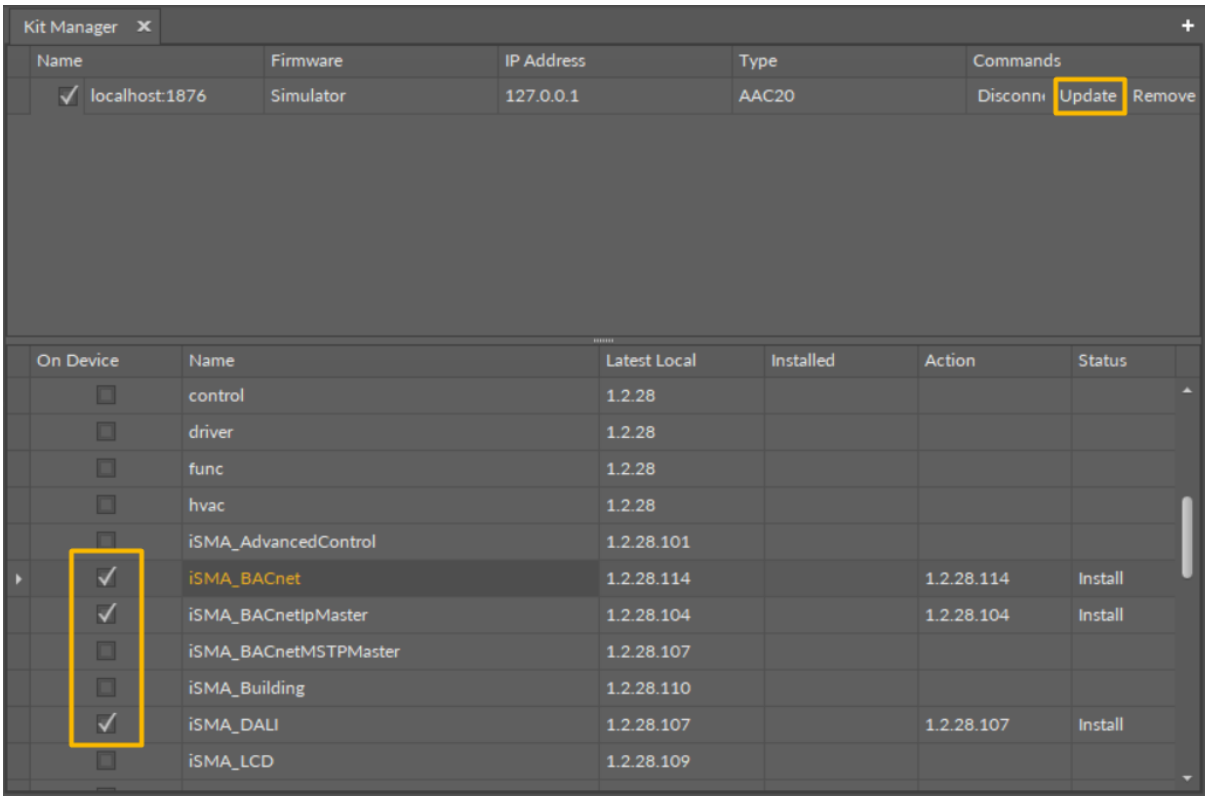



*Figure 275. Installing new kits in the Kit Manager*

- When kit changes have been successfully completed, the simulator automatically restarts and connects.

## 9.6.3 Saving the Modified Simulator App File

This procedure describes how to stop the simulation and save the modified app.

## Steps

- In the Objects Properties, click the Application Manager to launch the tool.
- In the Application Manager view, click the Get App option to save the application, and click Update.

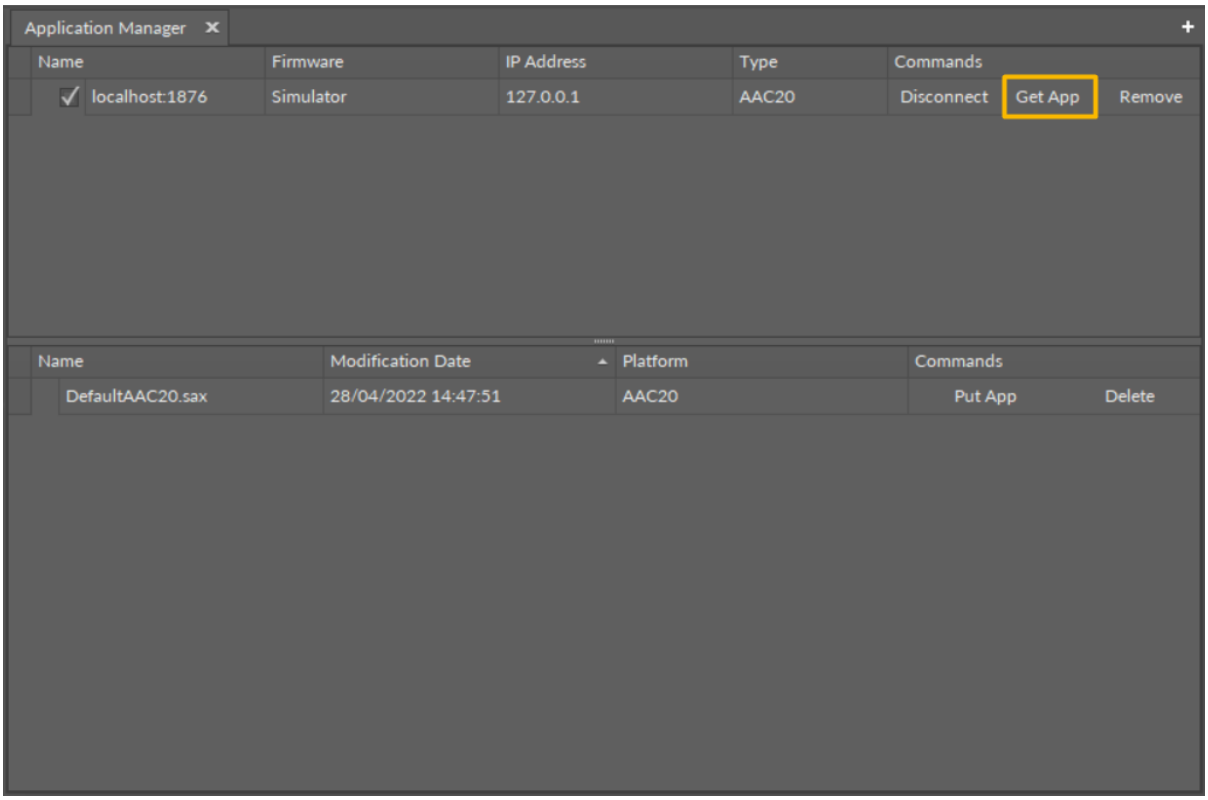



*Figure 276. Saving application*

- The modified app has been saved successfully to a selected location.

## 9.6.4 Scope of Simulator's Actions

### Scope of Actions in Managers:

- Application Manager and License Manager: full functionality;
- Kit Manager: not supported (view mode only).

### Scope of Actions in Services:

- Plat: all options functioning (working on localhost, some options are set to output values);
- Users: adding, removing, and editing of users functioning;
- Sox: all options are configurable and saved;
- Time: all options are configurable and saved;
- Alarm: all options are configurable and saved;
- History: all options are configurable and saved;
- Drivers: not available;
- Logic:
  - Sys: all options functioning;
  - iSMA_ControAPI: all options functioning;
  - iSMA_Control: scope of options is limited in regards to integer elements, which are set to a minimum value and cannot be configured;
  - iSMA_platAAC20: not implemented in the simulator;

## 9.7 iSMA-B-FCU in the iC Tool

This section describes how to control the iSMA-B-FCU device in the iC Tool.

### 9.7.1 Software Requirements for iSMA-B-FCU in the iC Tool

The iC Tool requires a connection with the iSMA-B-FCU using a dedicated software, the FCU Updater. The latest software can be downloaded here: https://ismacontrolli.com.

#### Windows Compatibility

The FCU Updater is an application intended for Windows operating system only, which can be run on Windows 7, 8, and 10 series.

> **Warning!**
>
> To use all features of FCU Updater the latest Oracle Java 32-bit is required.

Please, download the latest Java software from Oracle Download website first.

### 9.7.2 SOX Protocol for Connecting iSMA-B-FCU with iC Tool

In order to establish a connection between the iSMA-B-FCU controller and iC Tool, the SOX connection needs to be opened.

The SOX function allows online programming of the iSMA-B-FCU controller (real time). The controller must be connected directly via a USB cable. The SOX button is always enabled each time the USB cable is connected directly to the iSMA-B-FCU controller and opening of any project is not required.

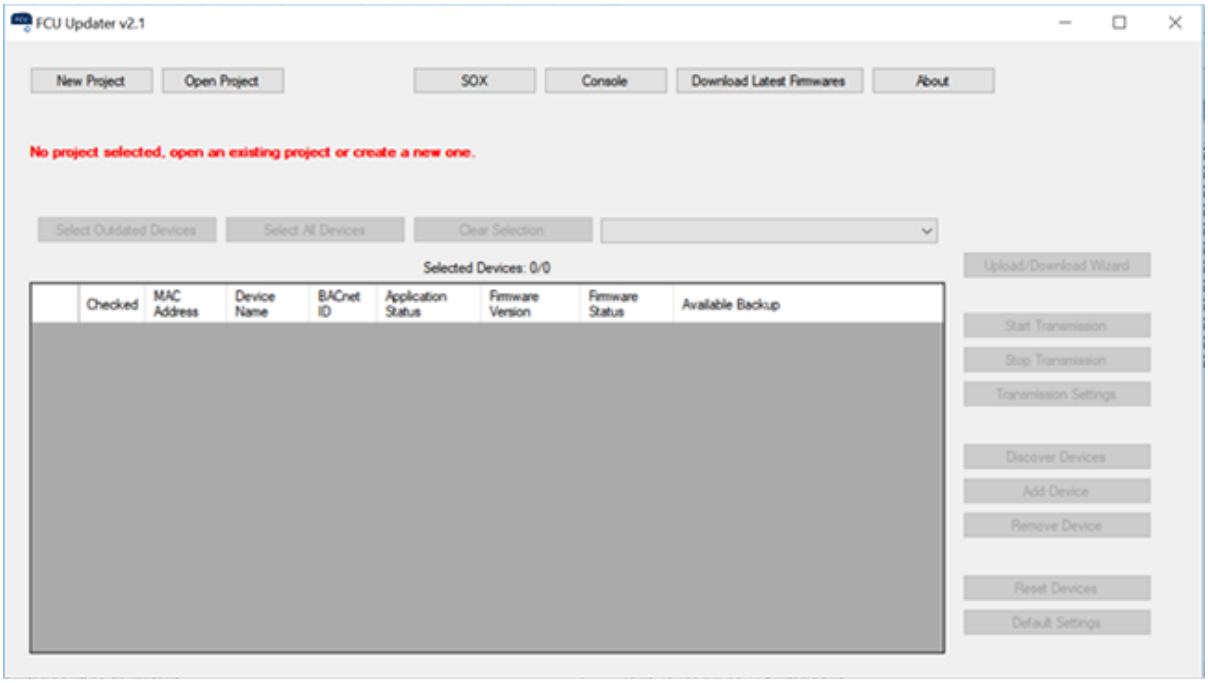



*Figure 277. Opening SOX protocol*

The SOX button guides the user to the next window, where the SOX protocol can be started by pressing the 'Begin Communication' button. After starting the communication

to the connected controller, the User can edit or build the application inside the iSMA-B-FCU controller by using the iC Tool software.

The SOX window has the following communication parameters:

- Port: port number (usually 1876) used by SOX protocol;
- Begin Communication button: start SOX communication. It is recommended to open SOX communication with the iSMA-B-FCU controller before running the iC Tool software;
- Stop Communication button: stop SOX communication.

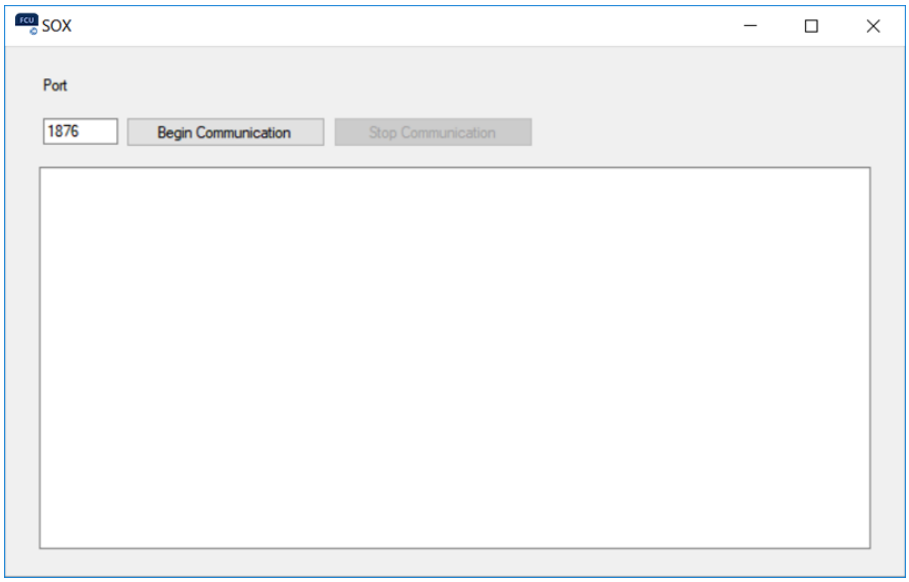



*Figure 278. SOX protocol console*

**Note:** Only one instance of the SOX protocol console can be opened at a time. If the SOX button in the main application window is disabled, the SOX protocol console has been already opened, and probably it is hidden somewhere under other windows on the user's screen.

If all the above actions have been performed, the communication via SOX protocol can be established with the iSMA-B-FCU controller using the iC Tool software.

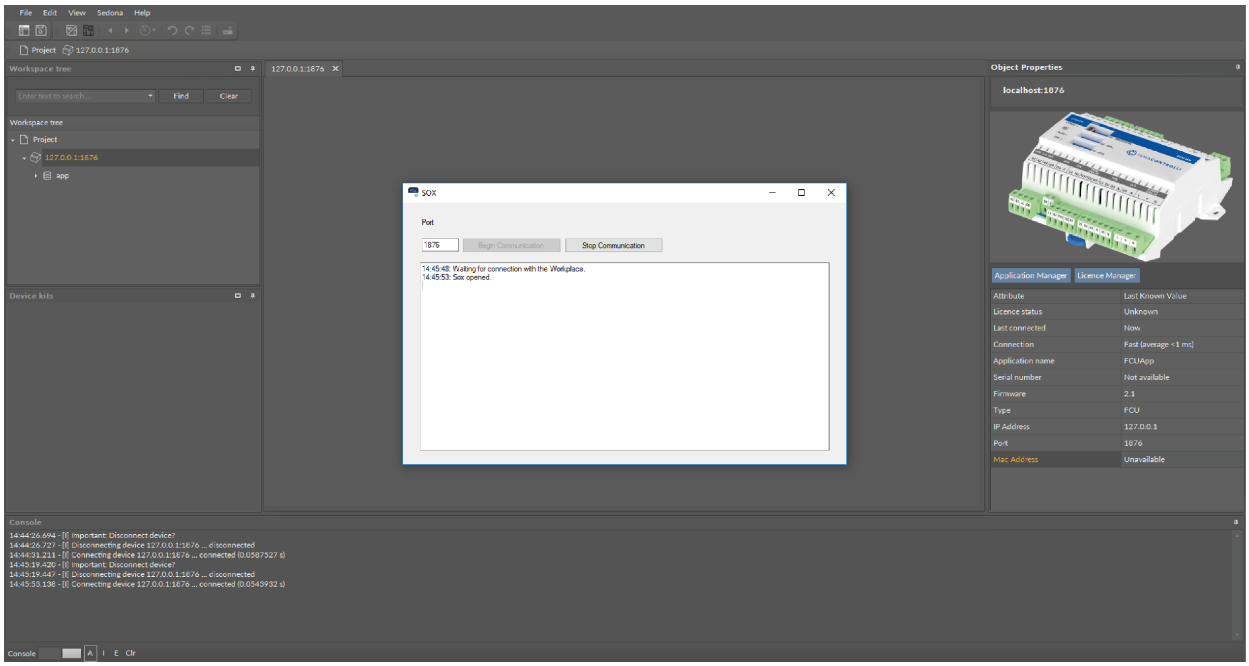



*Figure 279. Open communication beetwen iSMA-B-FCU device and iC Tool*

**Note:** A complete iSMA-B-FCU controller programming manual can be downloaded here: https://ismacontrolli.com.

# 10 Keyboard Shortcuts

| Shortcut | Description |
|---|---|
| Ctrl + N | New workspace (creates a new workspace on Workspace Tree) |
| Ctrl + O | Opens a dialog window to choose a *.itw file saved earlier (Workspace) |
| Ctrl + S | Save changes in the Workspace Tree window tree to a defined workspace in *.itw file. Does not apply to saving data in controller. |
| Ctrl + Q | Exiting iC Tool with confirmation of the intent to leave |
| Ctrl + C | Copying of selected components |
| Ctrl + V | Pasting of the components copied with (Ctrl + C) option |
| Ctrl + D | Duplication of selected components. The option does not overwrite what was remembered during (Ctrl + C) copying. |
| Delete | Removal of selected elements (component and/or links) |
| Ctrl or Shift | The keys allow to select components and links selectively. Ctrl adds the chosen element, and Shift defines a set of added elements. |
| Ctrl + N | Creates new Workspace Tree window for new projects |
| Ctrl + T | Opens a new tab |
| F | Rescales all components on the Wire Sheet view to see the all on the screen |
| Ctrl + mouse wheel | Zooms the Wire Sheet in/out |
| Ctrl + A | Selects all objects on active view |
| F11 | Hides/shows all other windows to make main screen as large as possible |
| F12 | Restores previous windows position and size before used F11 option |
| Ctrl + Tab | Changes focus to next window within opened windows in the iC Tool |
| Ctrl + Shift + Tab | Changes focus to previous window within opened windows in the iC Tool |
| Tab | Navigates to the next object in the current tree or list |
| Shift +Tab | Navigates to the previous object in the current tree or list |

| | |
|---|---|
| PageUp/ PageDown | Jumps to the first/last object in the current tree or list |
| Ctrl + F | Uses a filter on an active view |
| F2 | Renames selected object |
| F10 | Changes focus to the main menu with a possibility to navigate between menu items |
| Ctrl + P | Prints/exports current view (option not available on all views) |
| Ctrl + Left/ Right | Expands/collapses the tree |
| Esc | Cancels current settings |
| Enter | Accepts current settings |
| Alt + F4 | Exits the iC Tool with a prompt pop-up |
| Ctrl + Z | Undoes previous action on components |
| Ctrl + Y | Redoes previously undone action on components |
| Alt + Left/Right | Navigates a history view |
| Alt + Shift + Left/Right | Shows a history view flyout panel |

*Table 3. Useful keyboard shortcuts*