Intelligent Solution Managing Automation

iSMA-B-FCU

User Manual

FCU programming

Global Control 5 Sp. z o.o.
Warsaw, Poland
www.gc5.pl

Table of contents

1 Revision history

Rev	Date	Description
1.0	20.12.2016	First edition
1.1	20.08.2017	The reason for creation of new version of the document: • Rebuilt main algorithm of FCU default application (several components of iSMA_FCU kit rebuilt). • New functionality added (new components to iSMA_FCU kit). Changes in Document: • New way of switching Heating and Cooling Modes by FCU_HeatingCoolingSwitch component. • Added: description of FCU_PI component • Added: description of FCU_PID component • Added: description of FCU_TemperatureAnalog component • Added: description of FCU_TemperatureBinary component • Added: description of FCU_ValueHolder component • Added: description of FCU_WindowStatusSwitch component • Added: description of PWM component • Added: description of Room Device kit • Added: description of Advance Control kit

Table 1 Revision history

2 Programming iSMA-B-FCU device

Each new iSMA-B-FCU device is equipped with the default application, firmware and kits. There is a possibility to modify default application for individual purposes, use or create new, custom application.

Modification of default application or creating a new one can be done only online (in real time), using SOX protocol and WorkPlace. Size of application cannot exceed 64 kB. Available memory can be checked in Mem Available slot (under Plat component).

Using FCU Updater, modified or created application can be downloaded from one iSMA-B-FCU device and uploaded to another iSMA-B-FCU device(s). iSMA-B-FCU device has two built-in RS485 ports:

- COM1 – port with screw connector, port can be used for communication using Modbus RTU/ASCII protocol or BACnet protocol (including BACnet Master Slave communication)
- COM2 – port with two RJ12 connectors, port can be used for communication using Modbus Async protocol.
- Each iSMA-B-FCU device has set of kits, which are installed with firmware. These kits are required for proper operation of default application and can be also used to develop custom application. The kits cannot be changed or delayed. iSMA-B-FCU device is equipped with the following kits:
- sys – Sedona core system module
- control – Basic function block library
- inet – IP and UDP/TCP Socket APIs
- sox – Sox service for remote management
- iSMA_BACnetMasterSlave – Kit for Master - Slave communication
    - iSMA-B-FCU device can work in defined groups where one device is master and the remaining devices (slaves) follow Master parameters.
    - It is possible for sharing of up to 150 points in this way
    - This function is available only in BACnet MSTP protocol, using RS485 port (COM1)
    - Each master device can have up to 5 slave devices.
- iSMA_FCU – Kit includes components used to develop FCU application. It consists of components controlling temperature outputs, fan etc.
- iSMA_ModbusAsyncNetwork – Kit includes components for Modbus Async communication
- Modbus Async can be used to communicate with other devices connected to built-in RS485 port (COM2).
- Possibility to read/write up to 200 points in this way.
- There is no restriction about number of connected devices.
- iSMA_platFCU – Kit includes components for all types of inputs and outputs servicing, components for communication with higher level system (using Slave Network) and NV components.
    - Slave Network component is used to manage BACnet MSTP or Modbus RTU/ASCII protocol, using RS485 port (COM1).

- Slave Network allows for sharing of up to 200 numeric points and up to 200 boolean points.
- Total number of memory cells for NV Numeric (and Integer) components cannot exceed 200. Total number of memory cells for NV Boolean components cannot exceed 200.

Note: Method of calculating memory cells for NV components is described in Plat service chapter.

The iSMA-B-FCU device has 18 built-in physical inputs and outputs:

- 4x Special Inputs
- 4x Digital Inputs
- 3x Analog Outputs
- 5x Digital Outputs
- 2x Triac Outputs
    - CFG Dip Switch – it allows for reading of eight binary signals from CFG DipSwitch mounted in iSMA-B-FCU device. Using Dip Switch is recommended to manage configuration of device.
    - Alarm LED – allows for signalizing states of working iSMA-B-FCU device predefined in application. For example, it can be used for signalizing alarms.

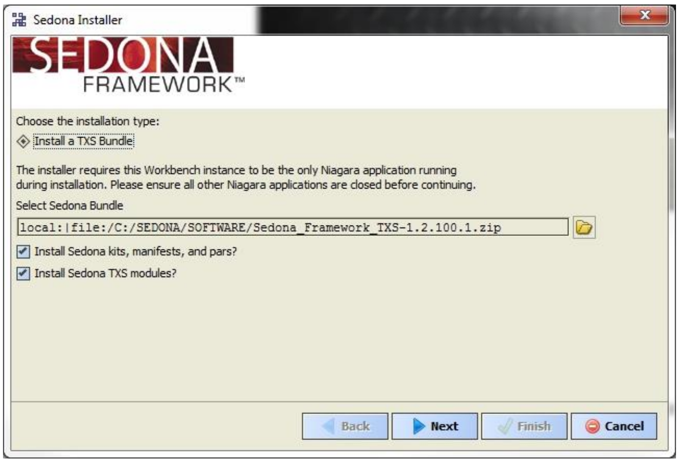
The components for servicing of all inputs, outputs shown below are placed under iSMA_platFCU palette.

3 Sedona in WorkPlace

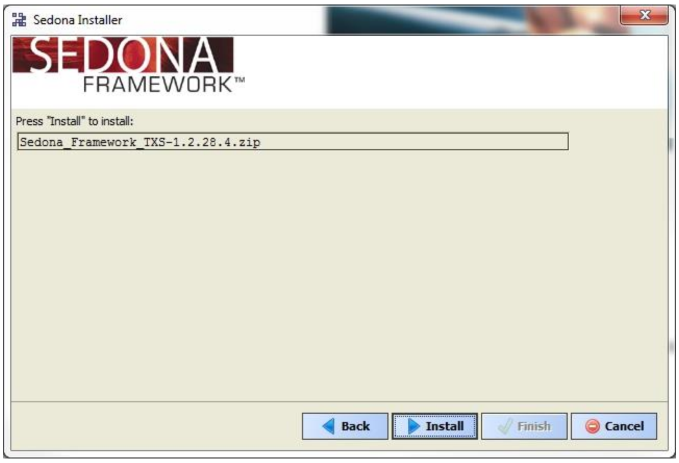
Sedona device configuration and programming is based on the WorkPlace software. Out of the box, WorkPlace has no plugin to handle Sedona devices and thus it should be installed. Installation files can be downloaded from the "Niagara Central" website or obtained from your distribution partner. WorkPlace version 3.7 and older require bundle Sedona_Framework_TXS-1.2.28. WorkPlace version 3.8 and later use bundle Sedona_Framework_TXS-1.2.100. To get Sedona Framework installation file visit Niagara Central or contact your distributor.

3.1 Environment installation step by step

Step 1 – In the menu bar, select the Tools tab and application Sedona Installer. When the first installation window pops up, select the Sedona bundle file.



Step 2 – Click “Next” and move on to the terms and conditions of the license. To continue, accept and confirm your installation by clicking “Install”.

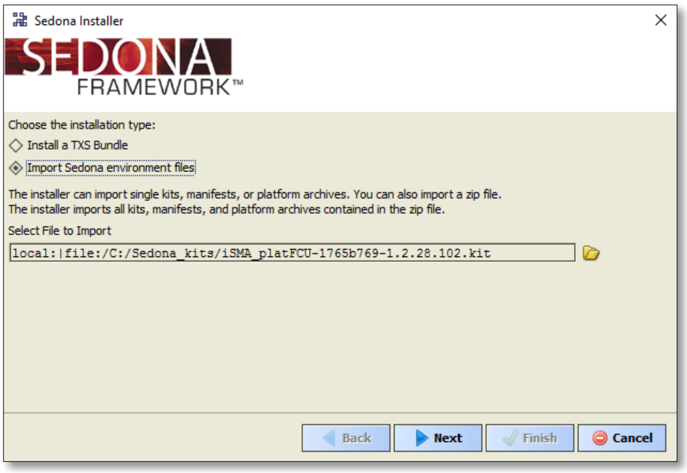


Step 3 – Upon successful installation of Sedona environment, restart WorkPlace by clicking “Restart Workbench” option followed by the “Finish” button.

3.2 Importing kits to Workplace

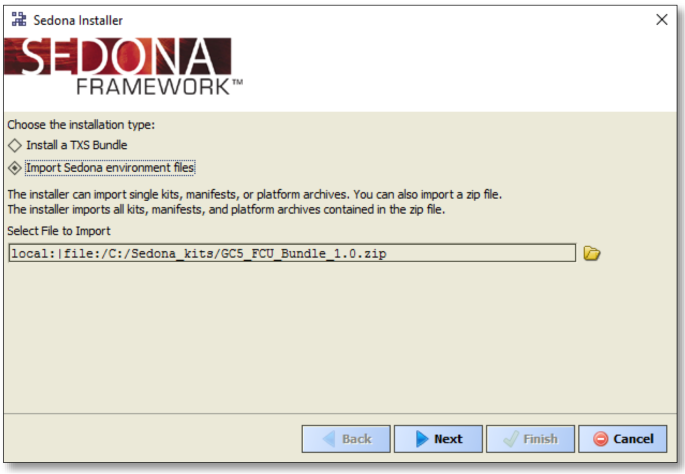
To import files to WorkPlace it is recommended to use “Sedona Installer” from Tools menu and choose “Import Sedona environment files”. The latest kits and kits’ bundles can be downloaded from support.gc5.pl website. There are two options to import kits to Workplace:

- Singly – by choosing single kit file with kit extension



Open Sedona Installer from Tools menu in Workplace, mark option “Install Sedona environment files” and select kit file. Press next button to finish importing process. This process will copy this file to Sedona kits folder.

- In groups – by choosing prepared bundle file with zip extension



Open Sedona Installer from Tools menu in Workplace, mark option "Install Sedona environment files" and select zip file. Press next button to finish importing process. This process will copy all kit's files from zip archives to Sedona kits folder.

3.3 Installing ismaUI module

ismaUI module is a special module which includes WorkPlace views for Sedona components.

To install ismaUI module, download jar file from support.gc5.pl website and copy ismaUi.jar file to Niagara/module folder on local PC. To make changes, restart WorkPlace.

3.4 Connecting to iSMA-B-FCU device

WARNING: Before starting programming iSMA-B-FCU device using SOX protocol, it is recommended to connect device to 230 V AC power supply.

- Before connecting iSMA-B-FCU device to WorkPlace, run FCU Updater.
- Connect iSMA-B-FCU device to USB port in your computer, SOX and Console buttons should be active which means that FCU Updater has got communication with iSMA-B-FCU device.

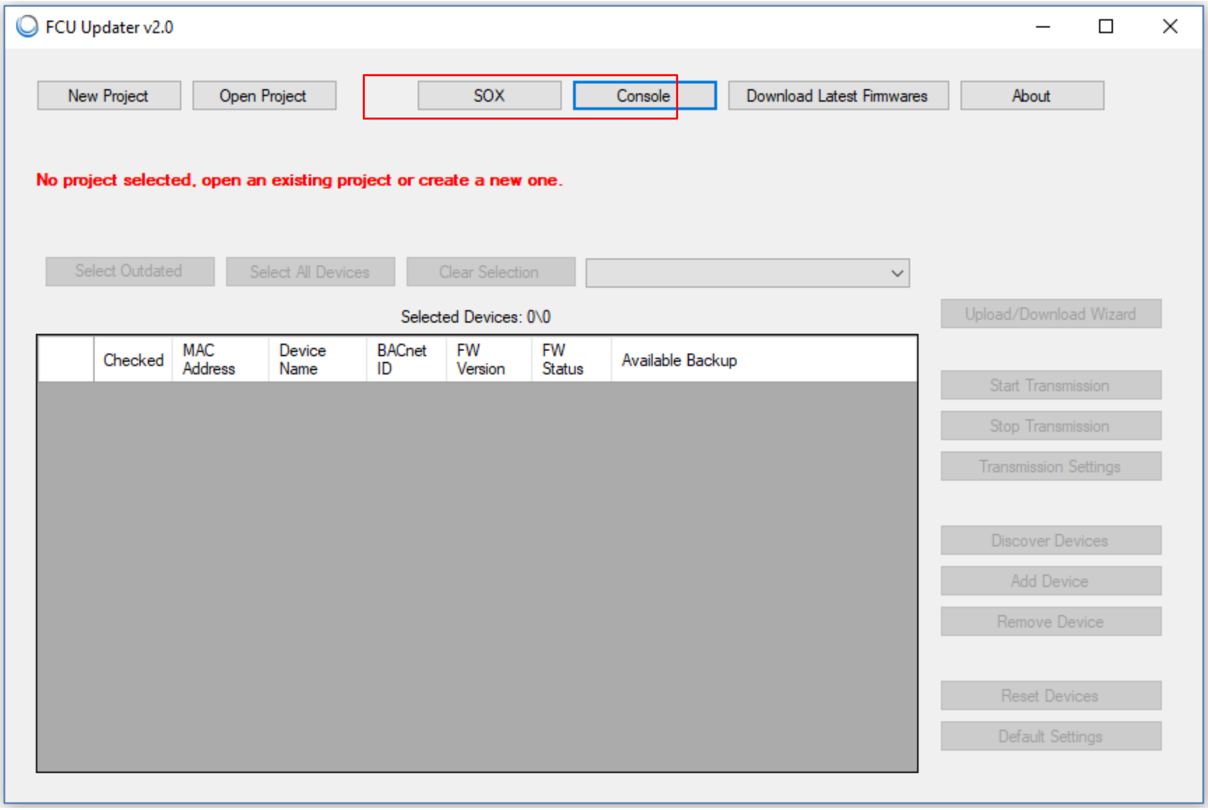


Figure 1 FCU Updater - SOX tab

Click SOX button to open popup window and then click "Begin Communication" button.

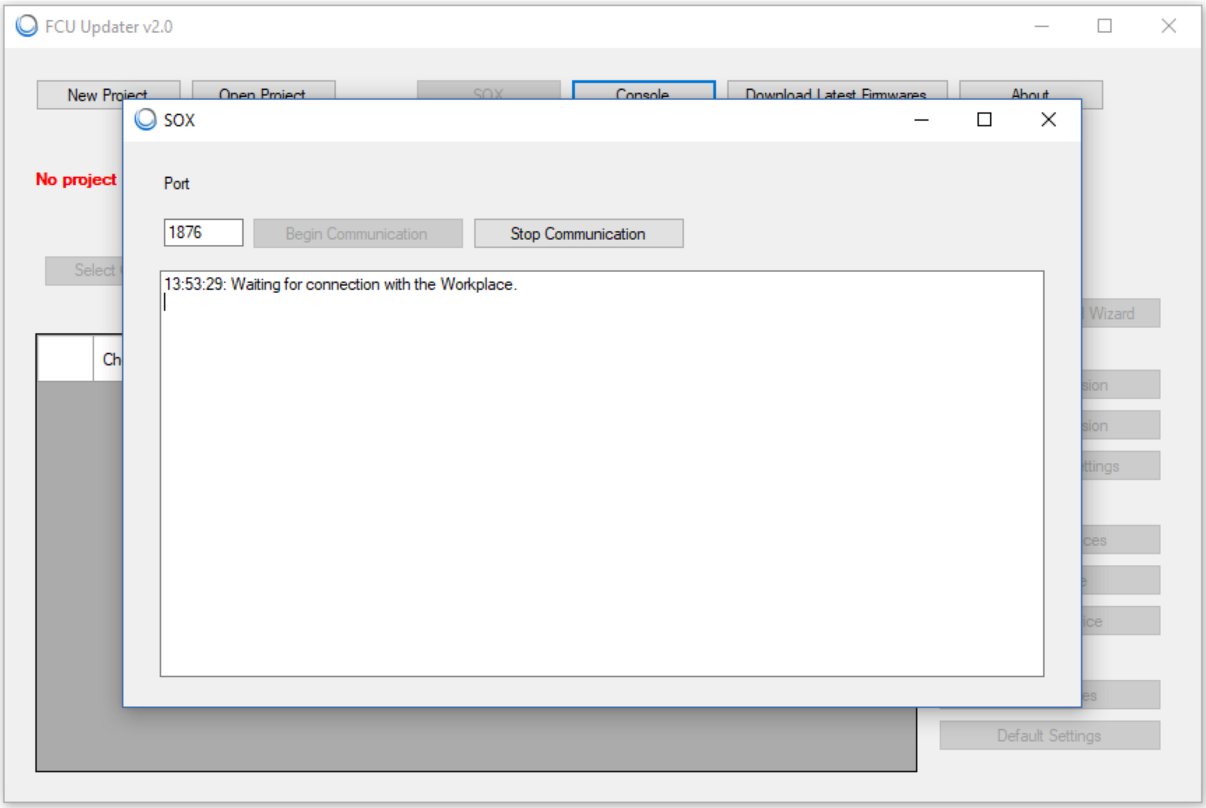


Figure 2 FCU Updater - Beginning comunication

- Now, go to WorkPlace.
- In order to connect to the device, select the following in the menu:

File -> Open -> Open Device.

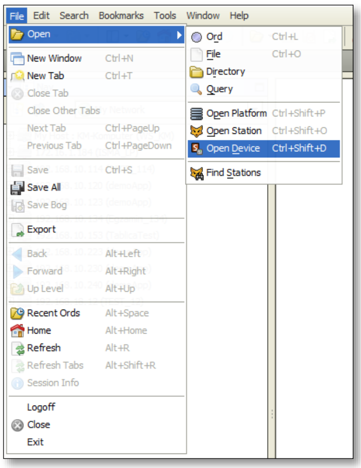


Figure 3. Device opening

Note! Should no option “Open Device” be available, no Sedona plugin has been installed in WorkPlace. In such a case, proceed to install the Sedona environment as described in the previous section.

- After selecting the above option, a device connect window pops up.

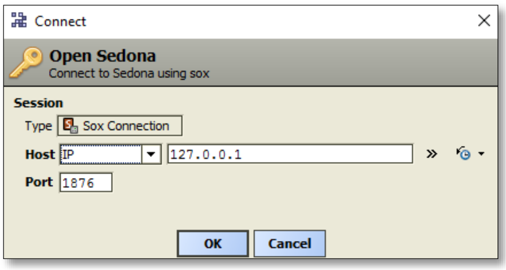


Figure 4 Connect window

Please, enter the following values:

- IP address – 127.0.0.1
- Port – 1876

- Login to driver using following Username and Password:
  - Username – admin
  - Password – empty box (no characters)

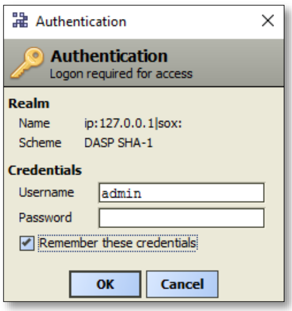


Figure 5 Authentication window

4 Sedona Tools

Sedona Tools do not support iSMA-B-FCU devices.

5 Sedona App component

Sedona Application consists of services and components available from the Palettes. Components are processed in every working cycle of the device. Services cover certain components enabling system functions, such as user management. All items should be placed under the App main component. When the application is modified, the App icon is displayed with reminder that application should be saved.

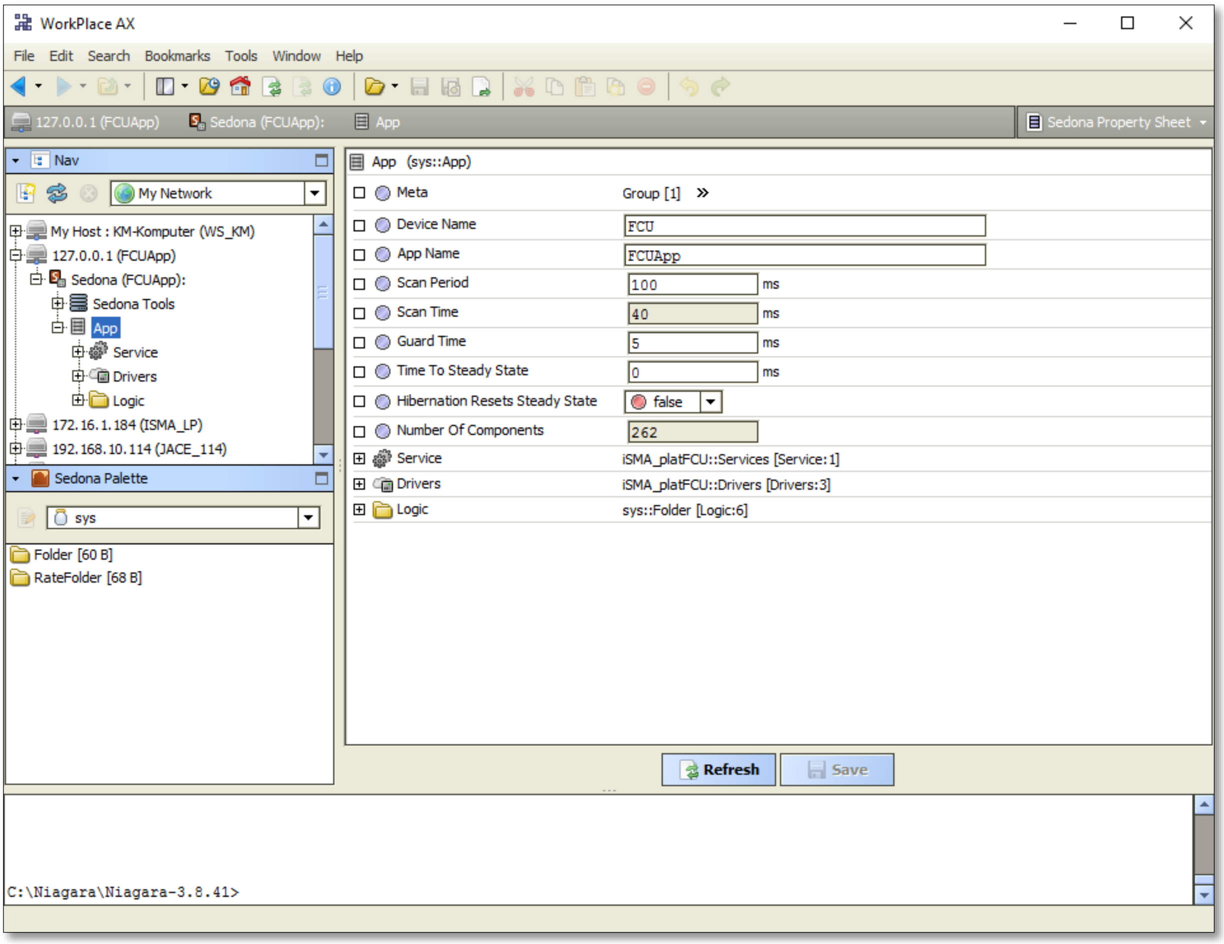


Figure 6 App component

The component has the following slots:

- Device Name – Device Name text,

- App Name – Application Name text,
- Scan Period – One cycle execution time,
- Scan Time – Real time of one cycle execution,
- Guard Time – Reserve time to finish system tasks,
- Time To Steady State - Time from app start to steady state,
- Hibernation Resets Steady State – not active in iSMA-B-FCU,
- Number Of Components – Number of components used in application.

The component offers the following actions, available under the right mouse button:

- Save – saves Sedona application in the device’s flash memory,
- Restart – restarts application (Sedona Virtual Machine),
- Reboot – reboots device.
- Quit – Closes sox connection
- Hibernate – Not active

Under App component the following components should be placed:

- Service
- Drivers
- Logic

All these components are described in sections bellow.

5.1 Service component

Service component has been created to manage services. Under this component one can find components which do not take direct part in the operation of the application. The component has to be placed under App component.

5.1.1 Plat service

The Plat service is a component which shows device's mains parameters. This component is placed under the Service folder and is associated with device hardware. The component has to be placed under Service component.

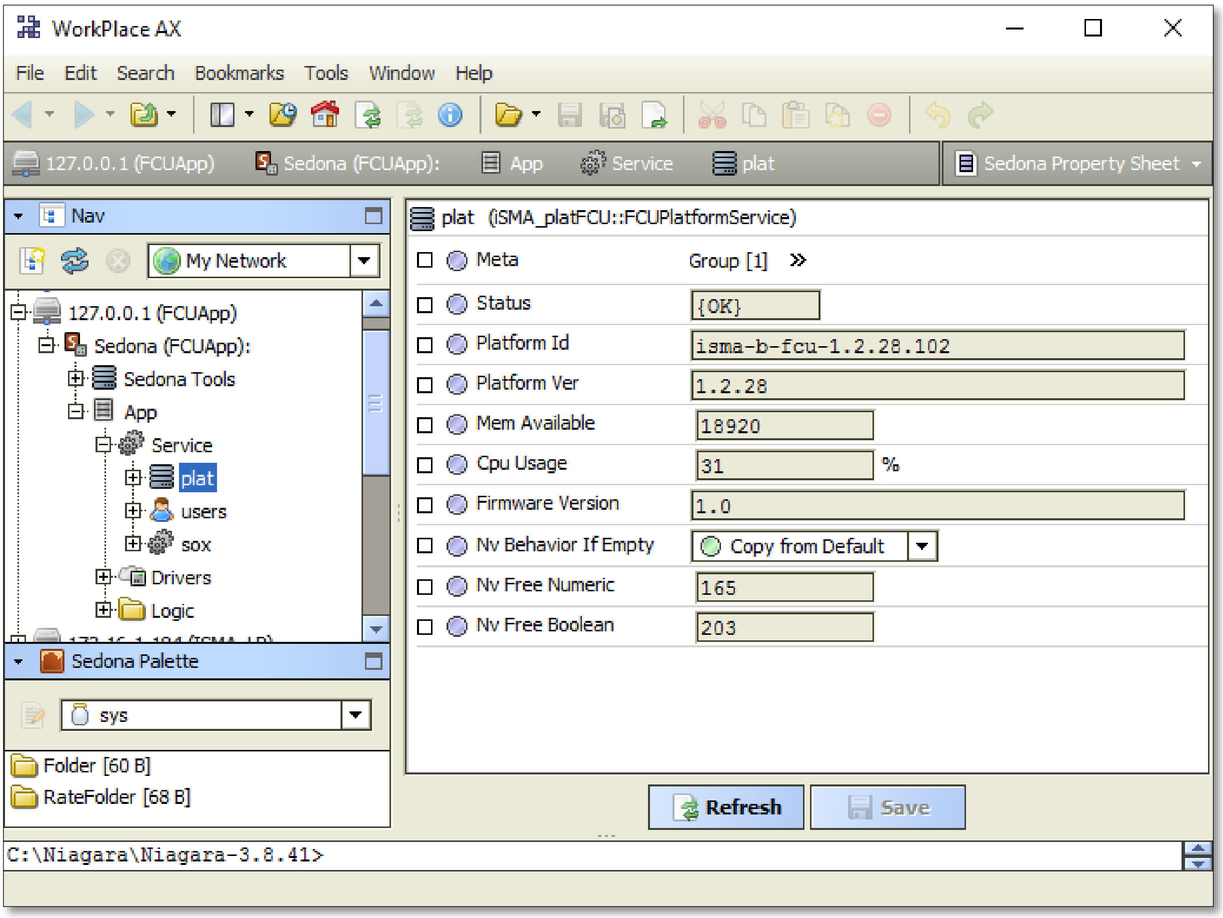


Figure 7 Plat component

The component has the following slots:

- Status – Platform status,
- Platform ID – Platform ID,
- Platform Ver – Platform Version,
- Mem Available – Controller RAM memory available. Whole application for iSMA-B-FCU device cannot exceed 64 kB,
- Cpu Usage – CPU usage from last 5 seconds,
- Firmware Version – Controller firmware version,

- Nv Behavior If Empty – If Non-Volatile memory is empty after copy NV component output value will be 0 (Options Leave 0) or copy value from Default slot (Option Copy From Default), the default settings is to copy value from Default slot,
- Nv Free Numeric – number of available Numeric Non-Volatile components,
- NV Free Boolean – number of available Boolean Non-Volatile components.

Note: Number of components physically added to application does not have to be equal to memory cells. It can be calculated in way shown below:

- Each added NV Numeric (or Integer) component (not NV Net component) uses two memory cells – one numeric cell for value and one Boolean cell for Hand/Auto mode,
- Each added NV Boolean component (not NV Net component) uses two Boolean memory cells – one for value and one cell for Hand/Auto mode,
- Each NV Net Numeric component uses one numeric cell – these components work only in Auto mode, so memory cell is required only for value,
- Each NV Net Boolean component uses one Boolean cell – these components work only in Auto mode, so memory cell is required only for value,
- For Example: When we add 2 NV Numeric components, 3 NV Boolean components, 4 NV Net Numeric components and 5 NV Net components to the application, usage of memory cells can be calculated as presented in the table below:

Type of component	Number of components	Used Boolean memory cells	Used Numeric memory cells
NV Numeric	2	2 (for Auto/Hand modes)	2 (for values)
NV Boolean	3	6 (3 for Auto/Hand modes and 3 for values)	• 0
NV Net Numeric	4	0	4 (for values)
NV Net Boolean	5	5 (for values)	0
Total		13	6

Table 2 Memory cells calculation

As presented in the table, this application uses 13 Boolean memory cells and 6 Numeric memory cells. In this case, slots “Free NV Boolean” and “Free NV Numeric” (under plat component) will display:

- Free NV Boolean: 227
  227 = 240 (available Boolean memory cells) – 13 (Boolean memory cell used in application)

- Free NV Numeric: 224
  224 = 230 (available Numeric memory cells) – 6 (Numeric memory cell used in application)

The component offers the following actions, available under right mouse button:

- Restart – restarts the application (Sedona Virtual Machine),
- Reboot – reboots the device,
- Copy From Nv To Default – copies values from Out slot to Default slot in all NV components (see NV component chapter),
- Copy From Nv To User – copies values from Out slot to User slot in all NV components (see NV component chapter),
- Copy From Default To NV – copies values from Default slot to Out slot in all NV components (see NV component chapter),
- Copy From User To NV – copies values from User slot to Out slot in all NV components (see NV component chapter),
- Set All Nv In Auto – sets all NV components in auto mode.

5.2 Users3service

Users’ Service is a service responsible for user support. Here you add and remove users as well as define their access rights to individual components. Each Sedona application component has a Meta slot, used to assign it to one or more groups. Sedona has 4 predefined groups. The component has to be placed under Service component.

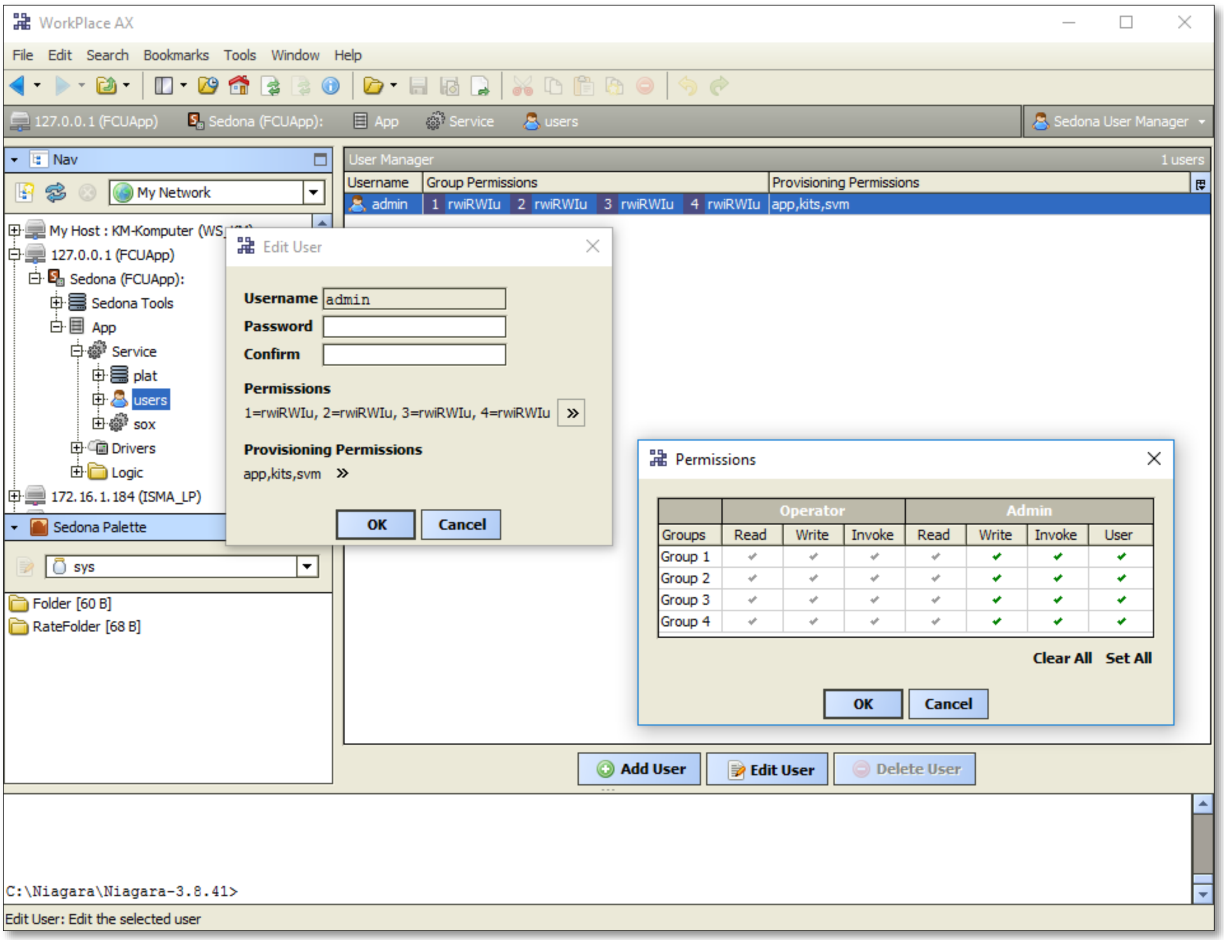


Figure 8 Users' component

Users can have the following types of rights:

- Operator Read – Reads components, read values of operator properties,
- Operator Write – Changes values of operator properties,
- Operator Invoke – Invokes operator actions,
- Admin Read – Reads values of properties, reads links, generates components links,
- Admin Write – Changes values' properties, adds components, sorts dub components, renames components, generates links to the components, deletes links to the components,
- Admin Invoke – Invokes admin actions of the components,
- Admin User – Users management (read, write, edit, delete).

Provisioning Permissions:

- Can provision app – can read/write app.sab file,
- Can provision kits – can read/write kits.scode file,
- Can provision svm – can read/write SVM files.

6 SOX protocol

SOX is the standard protocol used to communicate with Sedona devices. In this case, SOX is used to communicate iSMA-B-FCU device with WorkPlace, using USB connection.

Note: that SOX is a service type protocol, executed after application components. So, if there is little time difference between Scan Period and Scan Time (see App component), the services do not have sufficient time to execute and the programing interface can slow down. SOX is designed to be run over UDP/IP protocol (default port 1876), but in case of iSMA-B-FCU device, SOX protocol runs over USB interface.

The component has to be placed under Service component.

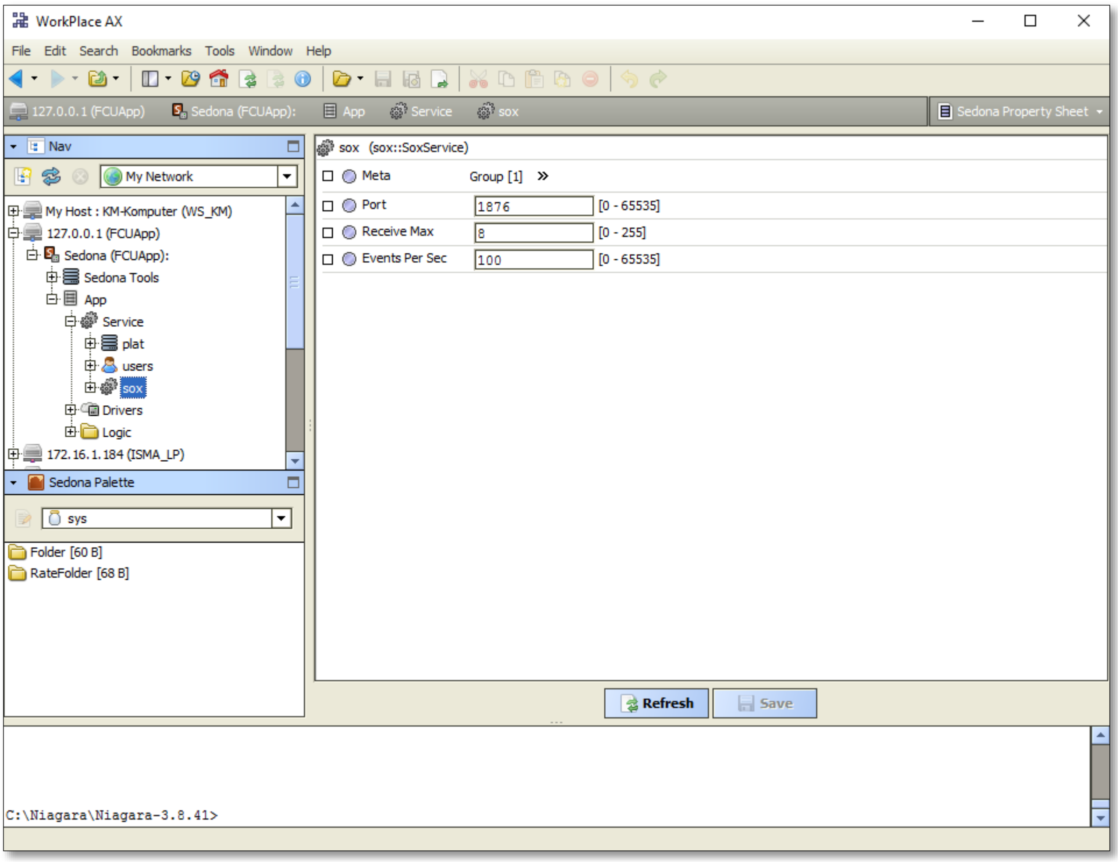


Figure 9 Sox component

The component has the following slots:

- Port – Sox UDP port (default value 1876),
- Receive Max – Maximal number of messages in receiving window,
- Events Per Sec – Maximal number of async events (telegrams) sent per second.

7 Drivers component

Drivers component has been created to manage networks used by application.

Under Drivers component all components responsible for networks and components associated with them, used by iSMA-B-FCU device are to be placed. It could be Local IO network (which manages physical inputs and outputs built-in iSMA-B-FCU device) or networks responsible for protocols, which allows to communicate with other devices (in case of iSMA-B-FCU device: Modbus Async, Slave Network and BACnet Master Slave).

The component has to be placed under App component.

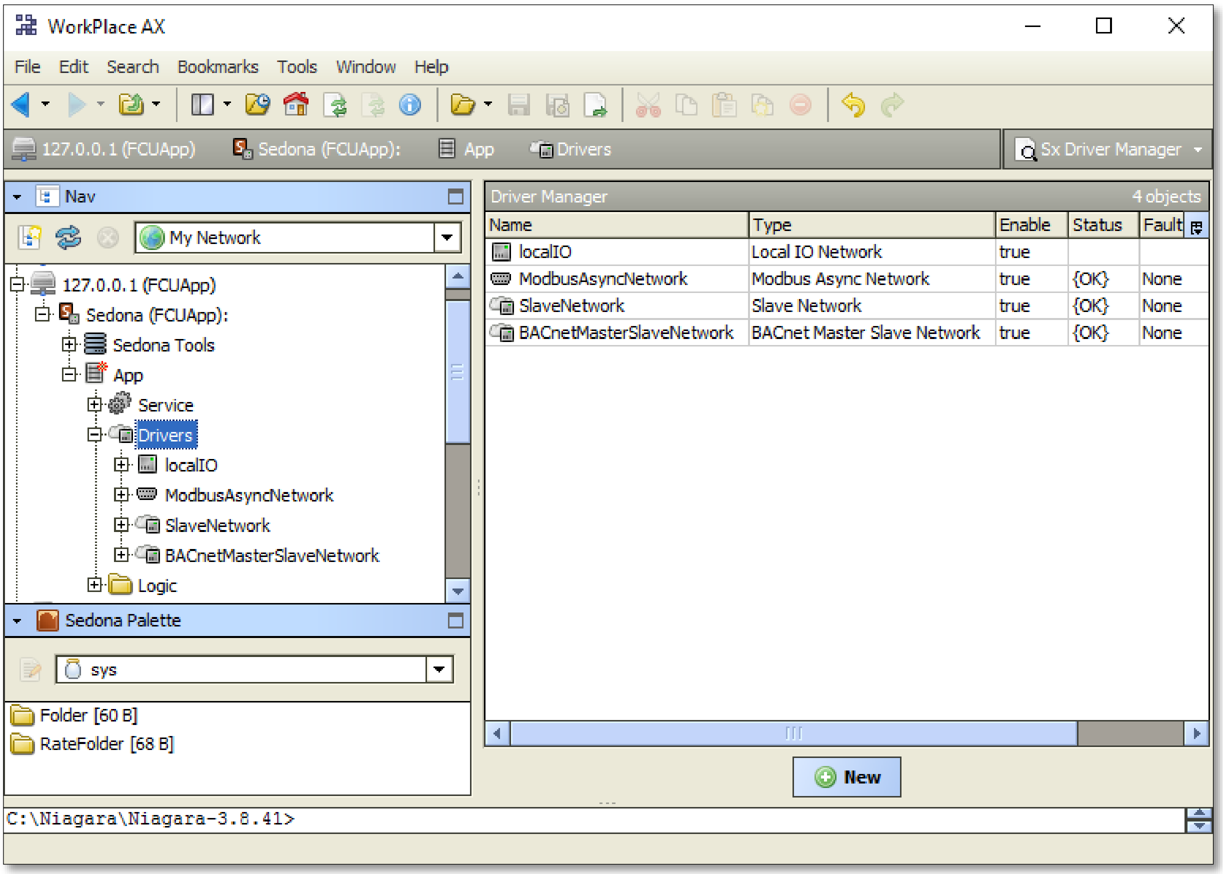


Figure 10 Drivers component

8 Logic folder

Logic Folder is a folder for grouping components used in development of logic of application, such as function blocks, NV components etc. These components can be placed everywhere under App component, but it is recommended to place them under Logic Folder – it allows for easier interpretation of all applications.

Logic folder is a component available from the sys kit palette; Folder component is component of no properties and is used exclusively to arrange applications visually.

Grouping of logic components can be also done in Rate Folder component, also available from the sys kit palette. Rate Folder component is a folder used to reduce the rate of application performance. While developing major applications, it allows to reduce a demand for processing power, consuming it for major or quick functions of the device algorithm. The folder's parameter “App Cycles To Skip” defines how frequently, in no. of scanning cycles, components under the folder are processed.

9 platFCU kit

iSMA platFCU kit contains components for servicing physical inputs and outputs of iSMA-B-FCU devices. It also contains Non-volatile components and Non-volatile Net components for servicing Slave Network. All these components are described in sections below.

9.1 NV components

NV Components (Non-Volatile) are the components the value of which can be recorded in an EEPROM device's non-volatile memory. Whenever a device is restarted or the power is down, the values of NV components remain saved.

The device has three types of NV components, which support different types of variables:

- Boolean variables – component NVBooleanWritable,
- Integer variables - component NVIntegerWritable,
- Numeric (float) variables - component NVNumericWritable.

NV components can operate in the Auto mode (the “In” slot values are transferred to the “Out” slot) or in the Hand mode (the “Out” value is entered manually by the user and cannot be changed by the application).

Since the values of the components are not stored in the Sedona application but in the non-volatile memory of the device, when an application is copied between two devices, output values are not saved and will assume the values stored in the local EEPROM memory. To copy NV components to another device along with their values (e.g. setpoint), use global actions of the plat component:

Step 1 - Use global action "Copy From NV The Default / Copy From NV To User"

Step 2 - Save the application and copy it to another device

Step 3 - Use global action on the target device "Copy From Default To NV / Copy From User To Nv"

9.1.1 MUAnnkd` mV ds` ald

NVBooleanWritable is a component that stores the output value in non-volatile memory of an EEPROM device. After rebooting the device or power failure, the component value is restored from this particular memory. The occupied space meter for EEPROM is embedded in the plat component.

Note: Way of calculating memory cells for NV components is described in Plat service section.

The NVBooleanWritable component is also used to integrate Boolean variables from various sources. It is done using the "reverse following the link" function. The Out slot is connected to the In slots of various protocols, for example BACnet or Modbus variable. Upon changing a value in one of the components, the device will perform the set action on the NV component to synchronize the values in all the connected components.

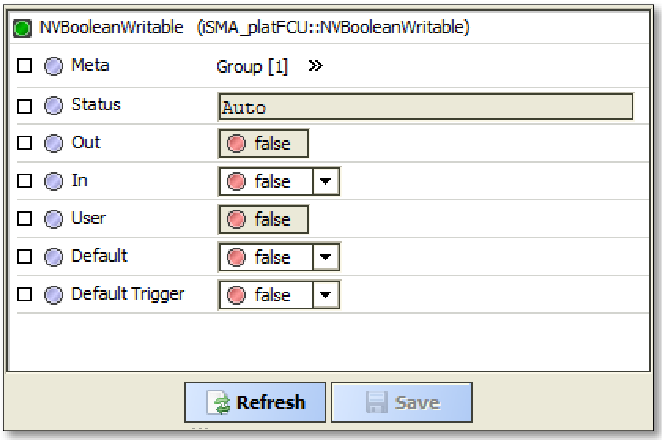


Figure 11 NVBoleanWritable component

The component has the following slots:

- Status – Point actual status Auto/Hand,
- Out – Output slot,
- In – Input Slot,
- User – User value slot (setting by Set action),
- Default – Default value slot (set by global command from plat action),
- Default Trigger – Copying trigger from Default to Out.

The component offers the following actions, available under the right mouse button:

- Set - this option sets User slot and In slot if there is no link on In slot,
- Set In Hand - this option sets the value on Out slot and blocks changing from any other slots,
- Set In Auto - this option switches off the Hand mode and sets Out slot according to the In slot's value.

9.1.2 MUHmsdfdqVqs`akd

NVIntegerWritable is a component that stores the output value in non-volatile memory of an EEPROM device. After rebooting the device or power failure, the component value is restored from this particular memory. Space meter of the occupied EEPROM's memory is located in the plat component.

Note: Way of calculating memory cells for NV components is described in Plat service section.

The NVIntegerWritable component is also used to integrate integer variables from various sources. It is done with the use of the "reverse following the link" function. The Out slot is connected to the In slots of various protocols, for example BACnet or Modbus variable. Upon changing a value in one of the components, the device will perform the set action on the NV component to synchronize the values in all the connected components.

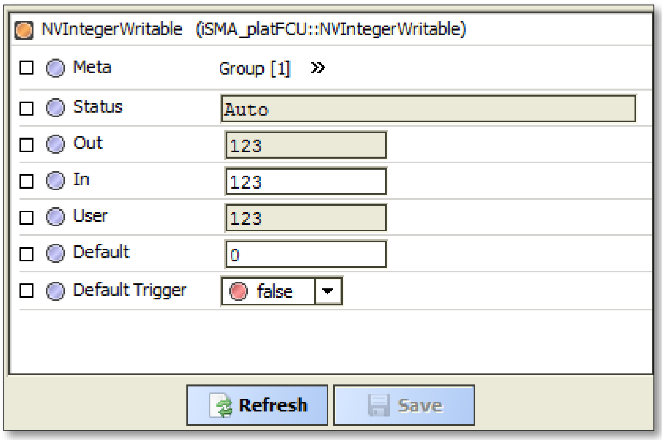


Figure 12 NVIntegerWritable component

The component has the following slots:

- Status – Point actual status Auto / Hand,
- Out – Output slot,
- In – Input Slot,
- User – User value slot (setting by Set action),
- Default – Default value slot (set by global command from plat action),
- Default Trigger – Copying trigger from Default to Out.

The component offers following actions, available under the right mouse button:

- Set – this option sets User slot and In slot if there is no link on In slot ,
- Set In Hand – this option sets value on Out slot and block changing from any other slots,
- Set In Auto – this option switches off Hand mode and sets Out slot according to In slot value.

9.1.3 MUMt l ddbV ds` ald

NVNumericWritable is a component that stores the output value in non-volatile memory of an EEPROM device. After rebooting the device or power failure, the component value is restored from this particular memory. Space meter of the occupied EEPROM's memory is located in the plat component.

Note: Way of calculating memory cells for NV components is described in Plat service section.

The NVNumericWritable component is also used to integrate Numeric (float) variables from various sources. It is done with the use of "reverse following the link" function. The Out slot is connected to the In slots of various protocols, for example BACnet or Modbus variable. Upon changing a value in one of the components, the device will perform the set action on the NV component to synchronize the values in all the connected components.

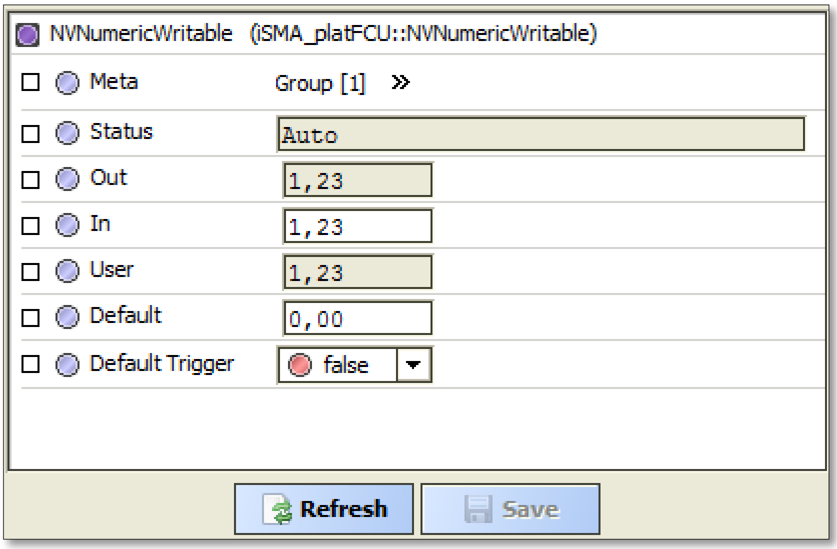


Figure 13 NVNumericWritable component

The component has the following slots:

- Status – Point actual status Auto/Hand,
- Out – Output slot,
- In – Input Slot,
- User – User value slot (setting by Set action),

- Default – Default value slot (set by global command from plat action),
- Default Trigger – Copying trigger from Default to Out.
- The component offers the following actions, available under the right mouse button:
- Set – this option sets User slot and In slot if there is no link on In slot,
- Set In Hand – this option sets value on Out slot and block changing from any other slots,
- Set In Auto – this option switches off Hand mode and sets Out slot according to In slot valueLocal IO components

The iSMA-B-FCU device is equipped 18 built-in physical inputs and outputs. The figures below show overview of all inputs and outputs of iSMA-B-FCU-HH and iSMA-B-FCU-HL devices:

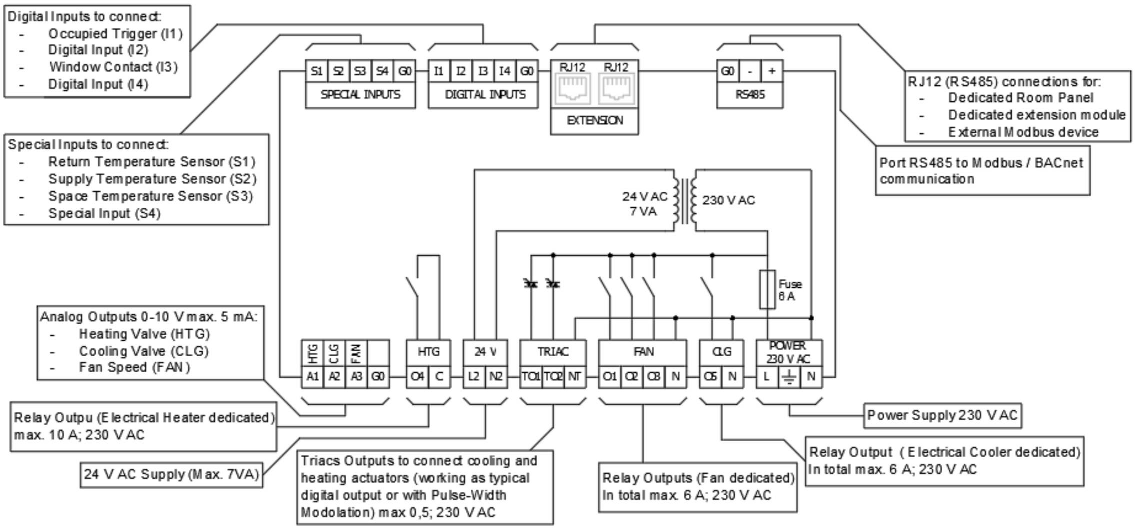


Figure 14 iSMA-B-FCU-HH terminals and internal connection view

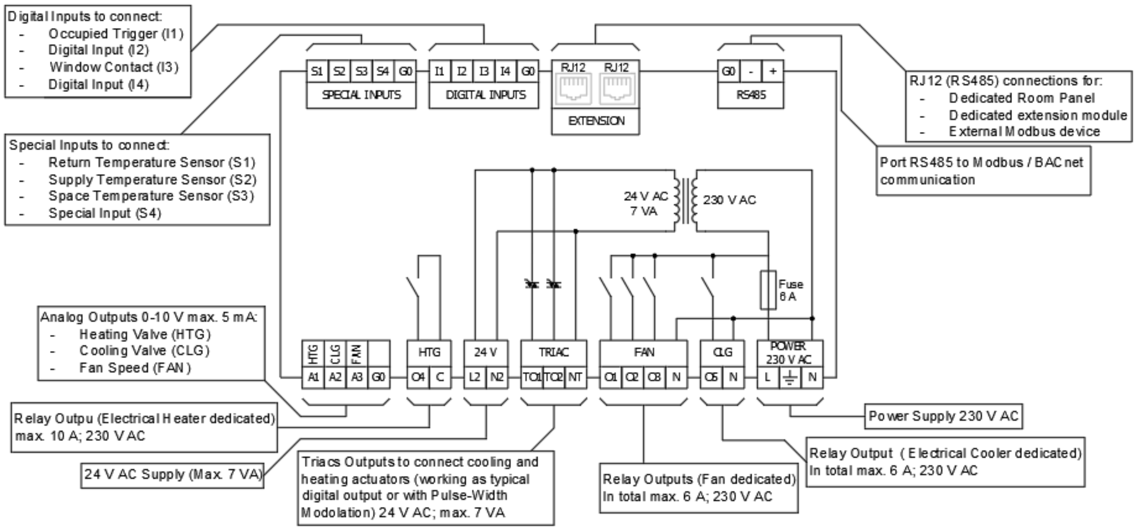


Figure 15 iSMA-B-FCU-HL terminals and internal connection view

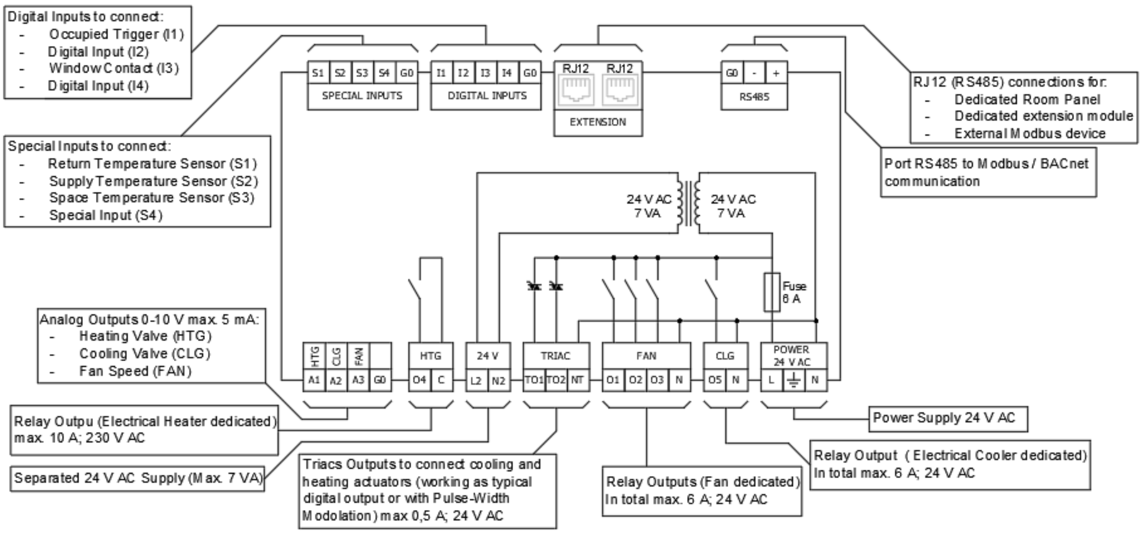


Figure 16 iSMA-B-FCU-LL terminals and internal connection view

4x Special Inputs

All Special inputs are characterised by 12-bit resolution which supports the following types of inputs:

- Digital Input – Special Input works as a digital input dry contact and reactive Boolean value “false” for open circuit and “true” for close circuit. Circuit status is measured by 1 mA current,
- Analog Input - Special Input measures voltage in range from 0 to 10 V DC (10 000 mV) with the resolution of 6 mV,
- Resistance Input - Special Input measures resistance value by voltage driver. The input works in range from 0 to 1000 kΩ (1 MΩ), with resolution ±20 Ω for 20 kΩ load,
- Temperature Input - Special Input measures NTC sensor resistance by voltage driver and converts to temperature value. Special Input has built-in conversation table for the following NTC sensors:

- 10K3A1 NTC B=3975K temperature sensor
- 10K4A1 NTC B=3695K temperature sensor
- 10K NTC B=3435K Carel temperature sensor
- 20K6A1 NTC B=4262K temperature sensor
- 2,2K3A1 NTC B=3975K temperature sensor
- 3K3A1 NTC B=3975K temperature sensor
- 30K6A1 NTC B=4262K temperature sensor
- SIE1 temperature sensor
- TAC1 temperature sensor
- SAT1 temperature sensor

4x Digital Inputs

Digital Inputs support the following types of inputs:

- Dry contact inputs
- Digital Input can work as a counter which counts dry contact impulses up to 100 Hz. Counter value is saved in non-volatile EEPROM memory.
- Note: During Restore to default process the counters' value is set to 0.
- Input working as Digital Output (in open collector mode).

3x Analog Outputs

All Analog Outputs are equipped with 12-bit ADC provides 10 mV resolution and accuracy less than ±0,5%. Analog Outputs support the following output types:

- Output 0-10 V DC
- Digital Output with maximum load up to 20 mA

5x Digital Outputs

All Digital Outputs are based on relays, working with 230 V AC voltage. The iSMA-B-FCU device has 2 types of Digital Outputs:

- O1-03, 05 – relay outputs connected directly to power supply terminal,
- O1, O2 and O3 are three-relay outputs, dedicated to connected speed Fans (max. 3). The common terminal of these outputs is connected directly to Power Supply "L" terminal.

Note: Outputs O1-O3 and output O5 are protected by integrated 6 A fuse. Total current for digital relay outputs O1-03 and O5 cannot exceed 6 A. It is forbidden to use fuse with higher current than 6 A! Higher current may permanently damage the device and cause danger to the user and equipment!

- O5 is a relay output, to which an electrical heater can be connected, for example. This relay is separated from the rest of the control circuit. Current consumption cannot exceed 10 A for 230 V AC power supply.

Note: This Digital Output has separate circuit with a 10 A relay. Such circuit requires external fuse protection of up to 10 A. Current higher than 10 A may permanently damage the device and cause danger to the user and equipment!

- O4 – relay separated from the FCU controller circuits. As an example, electrical heaters can be connected to O4. This relay is separated from the rest of the control circuit. Current consumption cannot exceed 10 A for 230 V AC power supply.

Note: This Digital Output has separate circuit with a 10 A relay. This circuit requires external fuse protection of up to 10 A. Current higher than 10 A may permanently damage the device and cause danger to the user and equipment!

2x Triac Outputs

The iSMA-B-FCU device has two Triac Outputs which are dedicated to heating- and cooling-thermal valve actuators. Triac Outputs can be connected to 230 V AC actuators (for iSMA-

B-FCU-HH) or 24 V AC actuators (for iSMA-B-FCU-HL), depending on the controller's model.

Triac Output can work as typical binary output (for Binary Temperature Control) or using PWM modulation. PWM mode has two parameters:

- Duration time in seconds (this value depends on valve parameters)
- Fill out, percentage value of signal fill out.

Note: In case of iSMA-B-FCU-HH controller, the actuators connected to each Triac Output can consume up to 0,5 A.

In case of iSMA-B-FCU-HL controller, total power consumption of both Triac Outputs and 24 V AC output cannot exceed 0,3 A (7 VA):

$$I_{max} = 0{,}3\ A = I_{TO1} + I_{TO2} + I_{24VOut}.$$

9.1.4 LocalIO

LocalIO is a main component servicing the physical I/O. Under this component all components used to reading or setting inputs or outputs have to be placed. LocalIO component has to be placed under Drivers component.

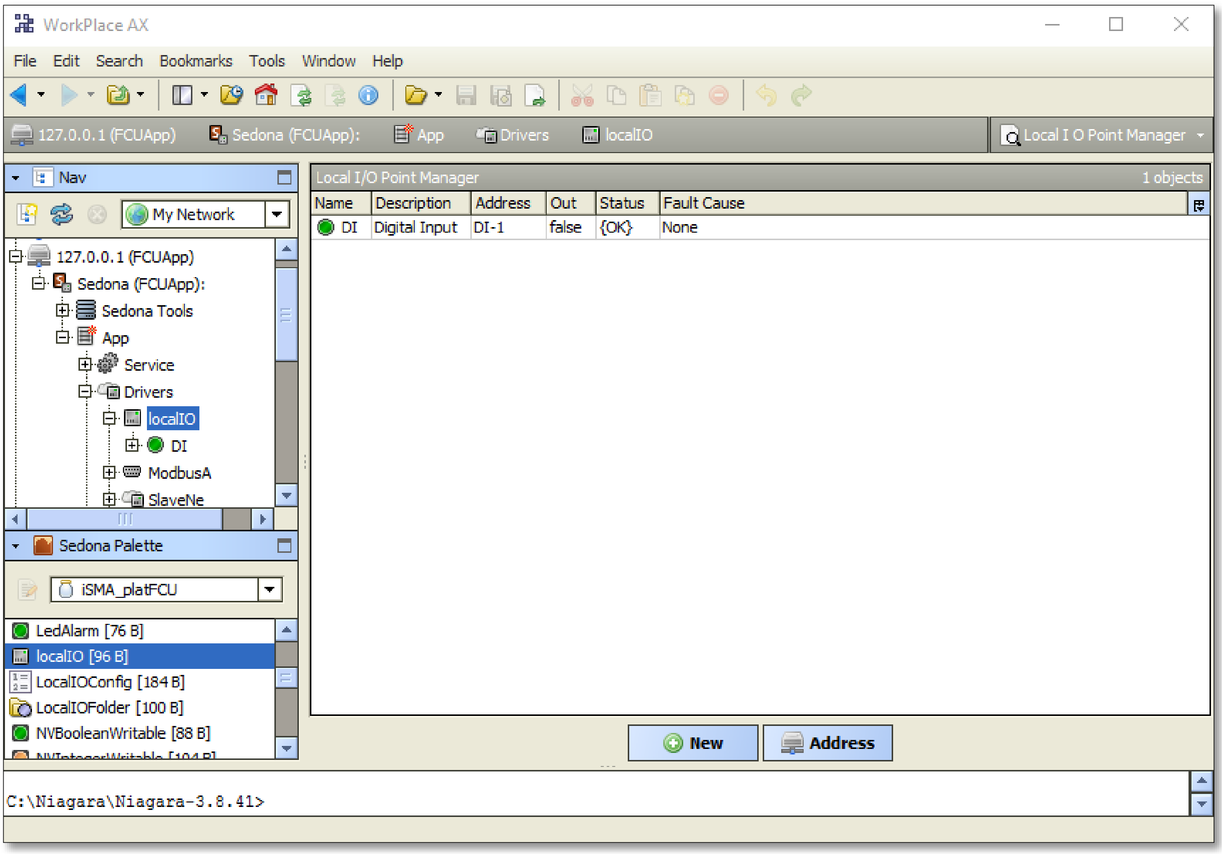


Figure 17 localIO component

9.1.5 LocalIOConfig

LocalCOnfig is a component designed for configuration of the physical inputs/outputs of the device. This component has to be placed under LocalIO component.

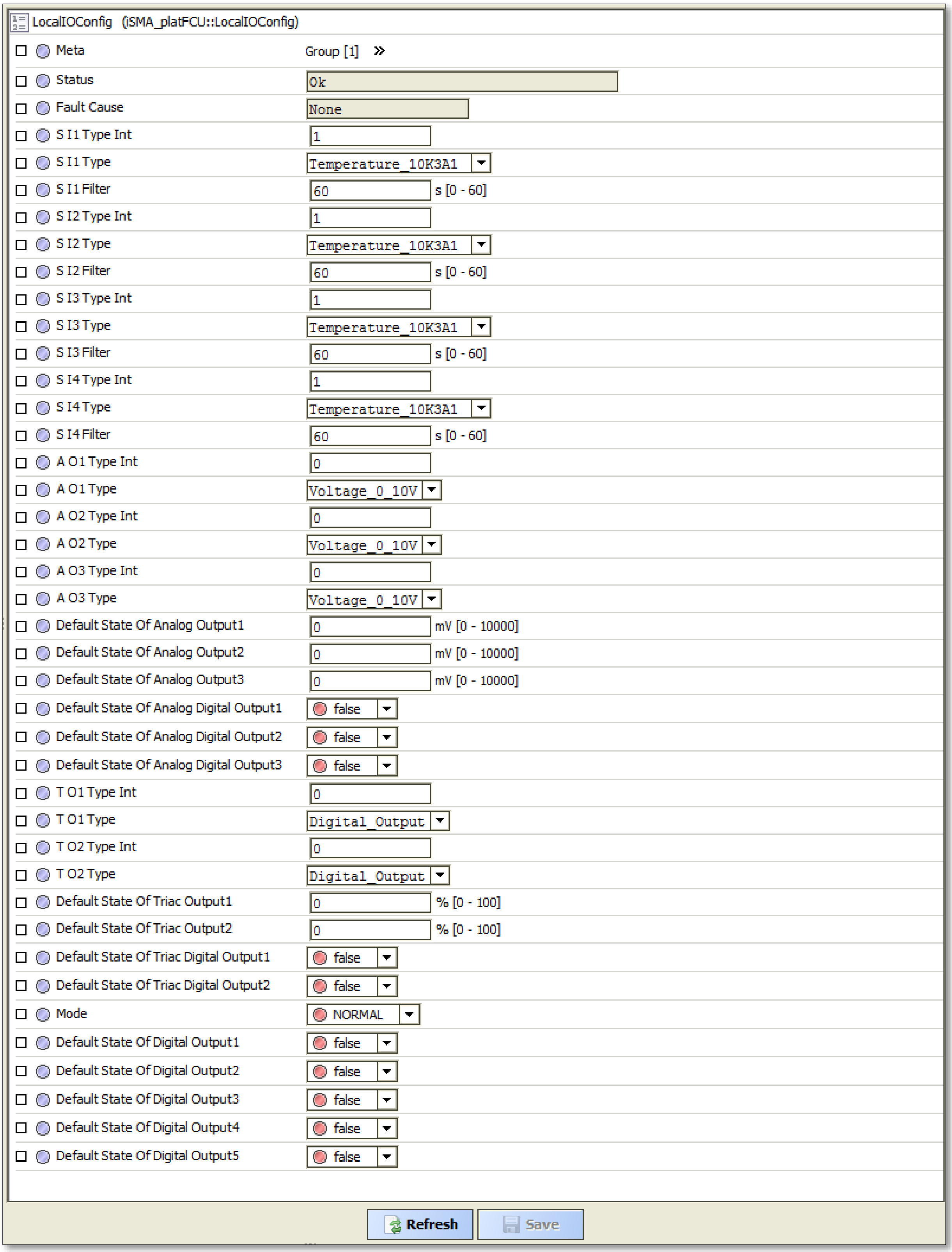


Figure 18 LocalIOConfig component

The component has the following slots:

- SIx Type Integer – Integer value corresponding to type of temperature sensor.
- SIx Type - Type of temperature sensor connected to special input; table stored in the device allows you to convert the value of sensor resistance into temperature.
    - Available options: Voltage_Measurement, Temperature_10K3A1, Temperature_10K4A1, Temperature_10K, Temperature_20K6A1, Temperature_2_2K3A1, Temperature_3K3A1, Temperature_30K6A1, Temperature_SEI1, Temperature_TAC1, Temperature_SAT1.
- SIx Filter - Time constant of the low pass filter (to eliminate signal noise)
- AOx Type - Integer value corresponding to analog output mode
- AOx Type - Analog Output Mode: Voltage 0-10 V or PWM
- Default State Of Analog Output x – The analog output default value after reboot of the controller expressed in mV.
- Default Digital State Of Analog Output x - The default value of analog output, operating in digital mode, after reboot of the device.
- TOx Type Int - Integer value corresponding to TOx Type.
- TOx Type - Triac Output Mode: Digital or PWM.
- Default State Of Triac Output x - The default value of the Triac Output, operating in PWM mode, after reboot of the device. Expressed in %.
- Default State Of Triac Digital Output x - The default value of the Triac Output, operating in digital mode, after reboot of the device.
- Mode - Slot can be set to NORMAL or FAN. When in FAN value, Digital Outputs O1, O2, and O3 cannot be set to high state at the same time. When in NORMAL value, this protection is disabled.
- Default State Of Digital Output x - Digital Output default after reboot of the device.
- Any settings of the LocalIOConfig component are stored in the component and can be transferred to other devices (quick setup of multiple devices).

9.1.6 LocalIOFolder

LocalIOFolder is a folder dedicated for grouping IO components.

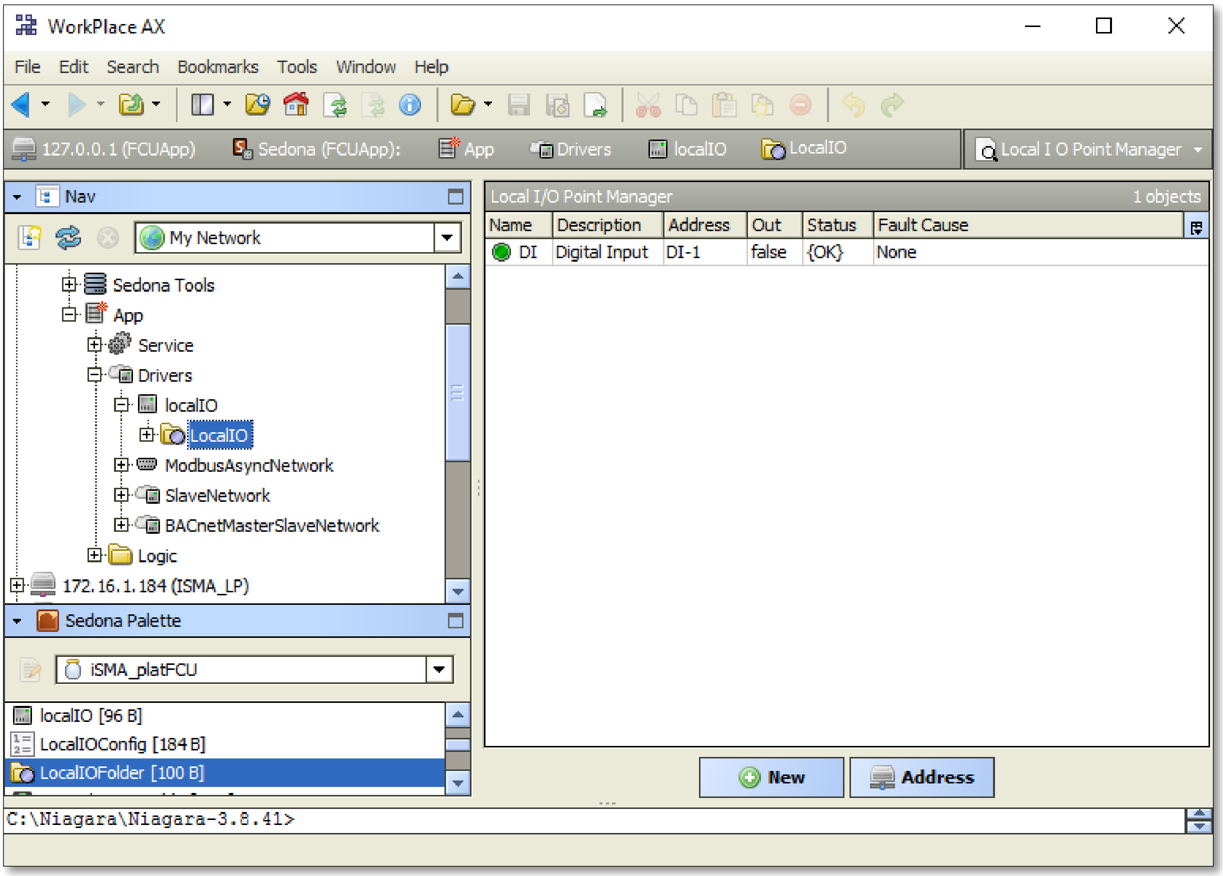


Figure 19 LocalIOFolder component

9.1.7 AODigital

AODigital is a component designed for servicing analog output in the digital mode (false - 0V, true - 10V). AODigital has to be placed under LocalIO component.

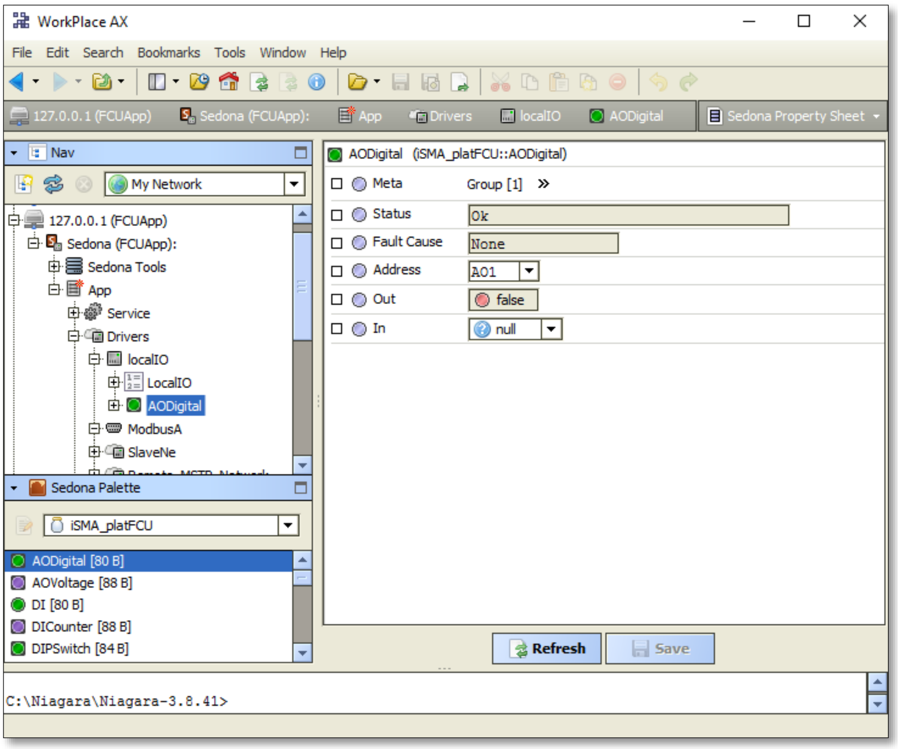


Figure 20 AODigital component

The component has the following slots:

- Status – Status of AODigital component,
- Fault Cause – Fault cause description,
- Address – Number of physical output of iSMA-B-FCU device (AO1, AO2, AO3),
- Out – Actual state of Output (true / false),
- In – Input slot.
- The component offers the following actions, available under the right mouse button:
- Set– Writes value to Out slot and sends it to the device.

9.1.8 AOVoltage

AOVoltage is a component designed for servicing Analog Output working as voltage output (range 0mV – 10 000m V). AOVoltage has to be placed under LocalIO component.

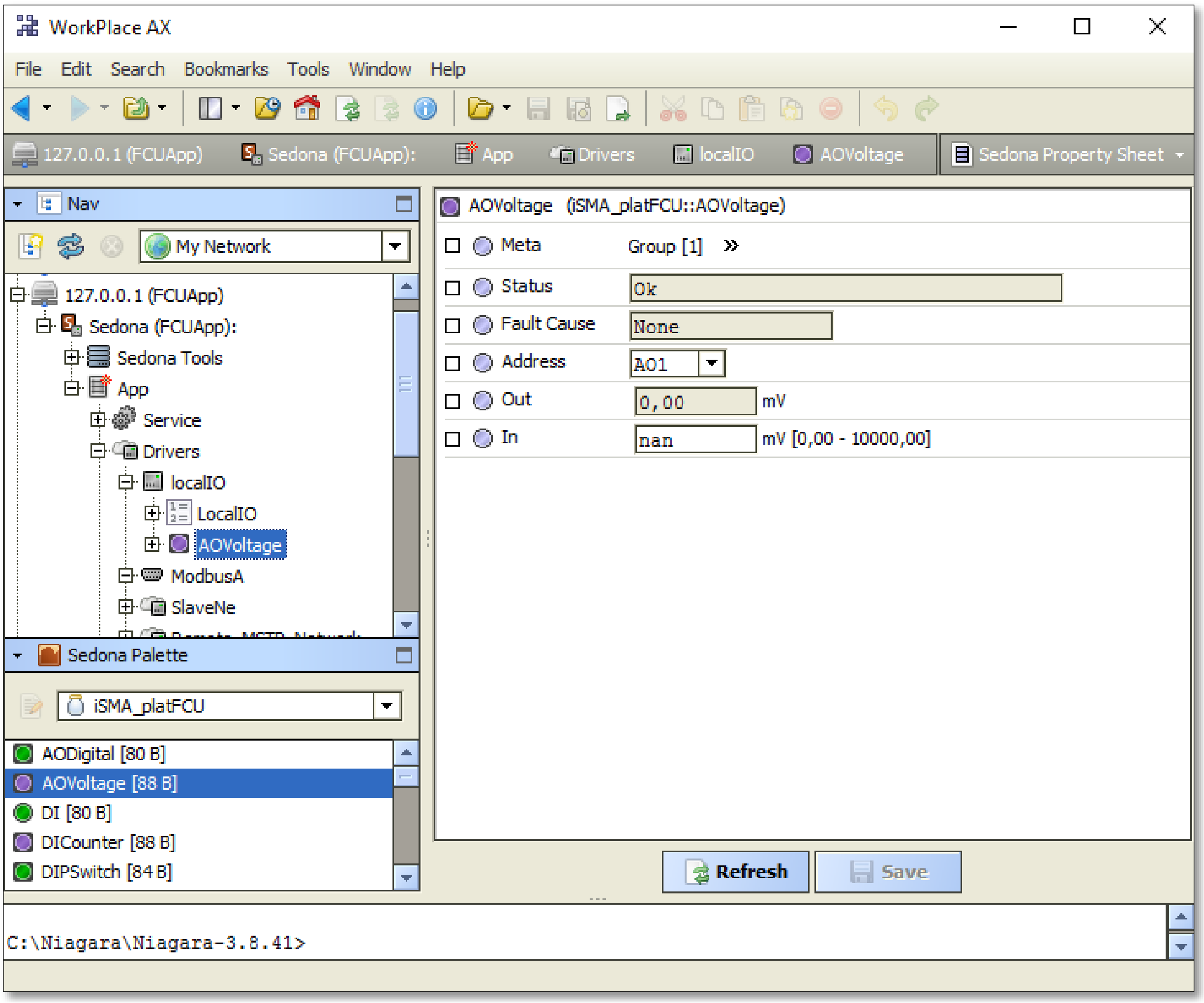


Figure 21 AOVoltage component

The component has the following slots:

- Status – Status of AOVoltage component,
- Fault Cause – Fault cause description,
- Address – Number of physical output of iSMA-B-FCU device (AO1, AO2, AO3),
- Out – Actual value of Output (0mV – 10 000 mV),
- In – Input slot.

The component offers the following actions, available under the right mouse button:

- Set– Writes value to Out slot and sends it to the device.

9.1.9 DI

DI is a component designed for reading Digital Input (true or false). DI has to be placed under LocalIO component.

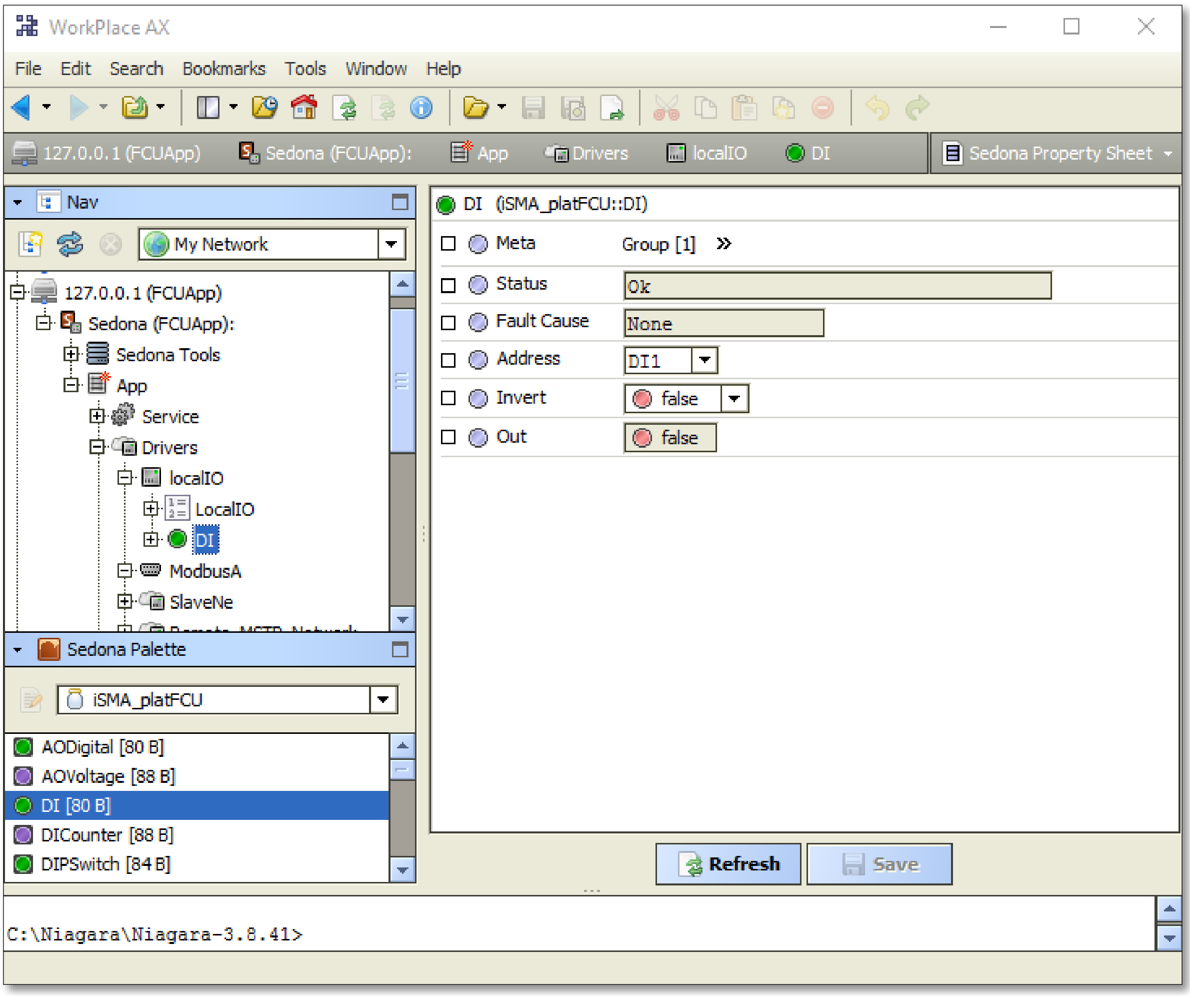


Figure 22 Di component

The component has the following slots:

- Status – Status of DI component,
- Fault Cause – Fault cause description,
- Address – Number of physical Input of iSMA-B-FCU device (DI1, DI2, DI3, DI4),
- Invert – Negation of Digital Output reading from iSMA-B-FCU device,
- Out – Actual value of Input (true or false).

9.1.10 DICounter

DICounter is a component dedicated for reading high-speed counter of Digital Inputs. DICounter has to be placed under LocalIO component.

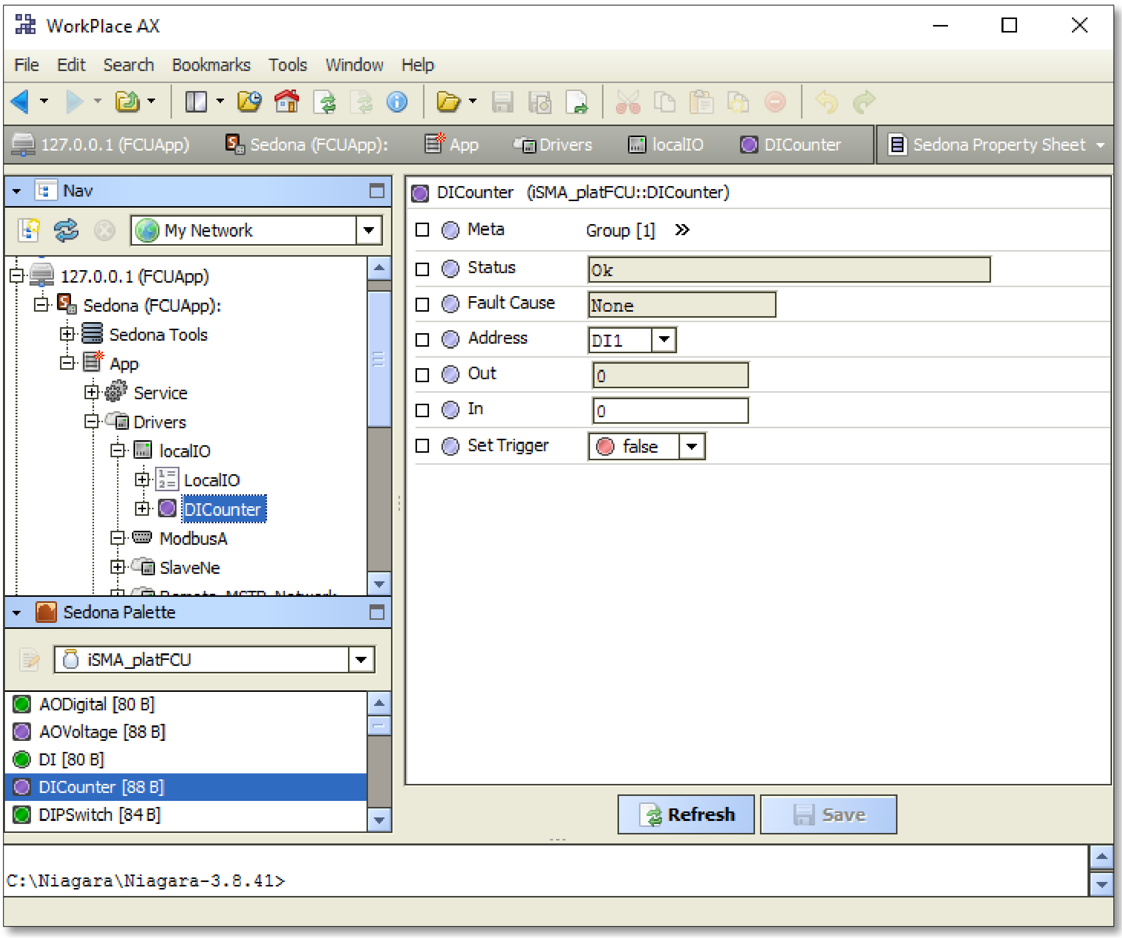


Figure 23 DICounter component

The component has the following slots:

- Status – Status of DICounter component,
- Fault Cause – Fault cause description,
- Address – Number of physical Input of iSMA-B-FCU device (DI1, DI2, DI3, DI4),
- Out – Actual value of counter servicing selected Input.
- In – Value which can be set to Out slot, when Set Trigger slot has changed from “false” to “true”,
- Set Trigger – when state of slot has changed from “false” to “true”, value of In slot is setting to Out slot.
- The component offers the following actions, available under the right mouse button:
- Set – Action allows to set the value of counter. Set value overrides Out and In slots,
- Reset Counter– Action allows to set value to 0. Set value overrides Out and In slots.

9.1.11 DO

DO is a component designed for servicing digital output. DO has to be placed under LocalIO component.

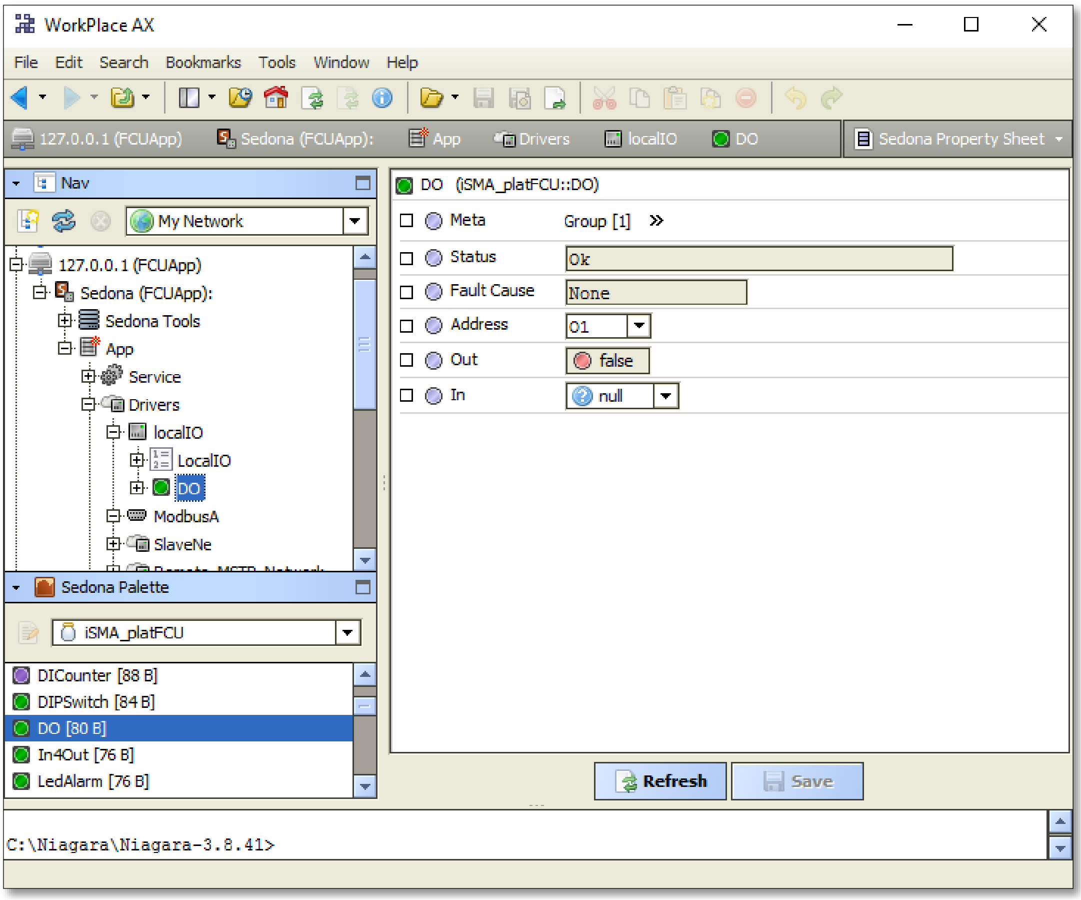


Figure 24 DO component

The component has the following slots:

- Status – Status of DO component,
- Fault Cause – Fault cause description,
- Address – Number of physical Output of iSMA-B-FCU device (O1, O2, O3, O4, O5),
- Out – Actual state of Output (true / false),
- In – Input slot.

The component offers the following action, available under the right mouse button:

- Set– Writes value to Out slot and sends it to the device.

9.1.12 SIDigital

SIDigital is a component designed for servicing Special Input in the Dry Contact read mode. SIDigital component has to be placed under LocalIO component.

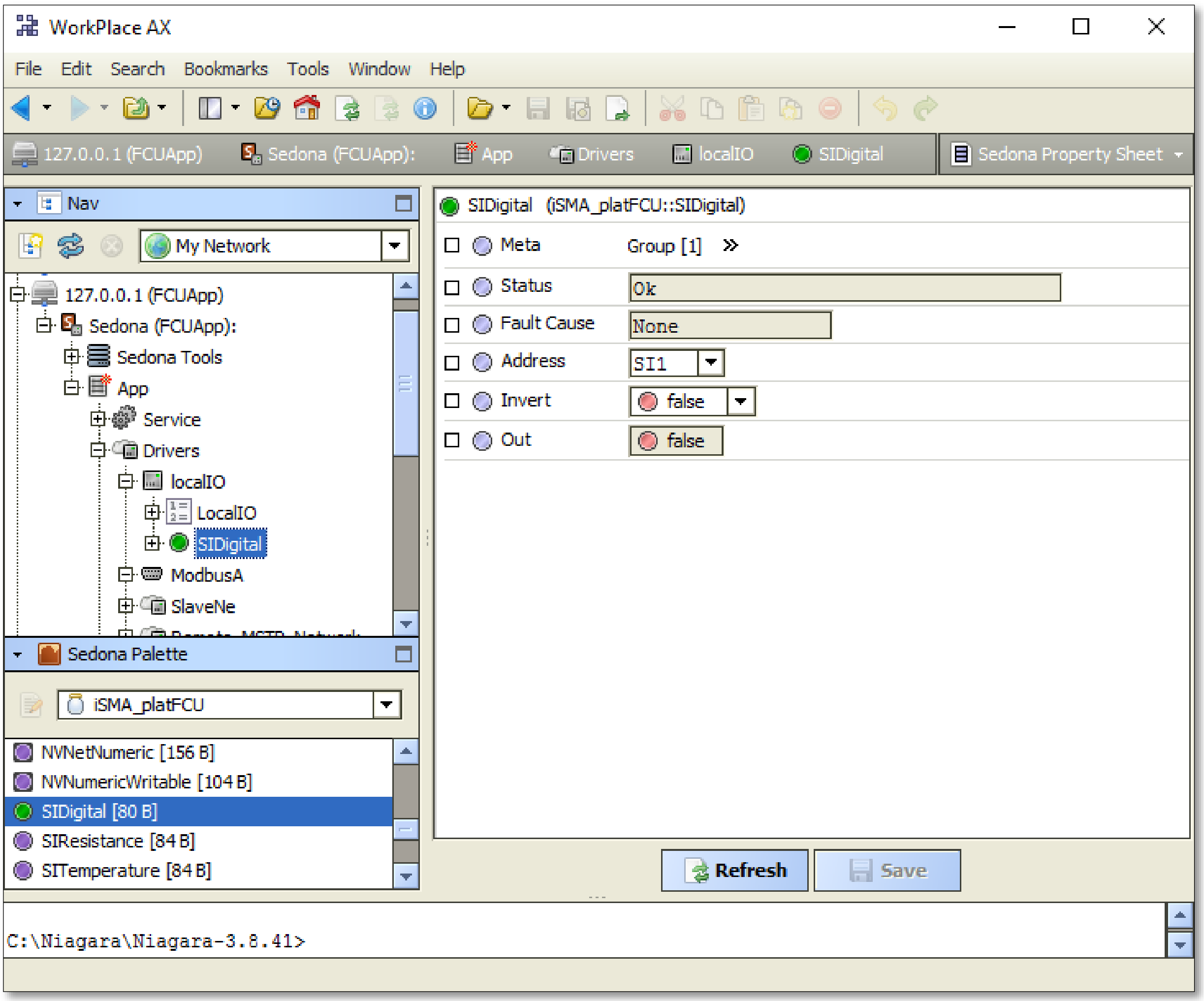


Figure 25 SIDigital component

The component has the following slots:

- Status – Status of SIDigital component,
- Fault Cause – Fault cause description,
- Address – Number of physical Input of iSMA-B-FCU device (SI1, SI2, SI3, SI4),
- Invert – Negation of special output reading from iSMA-B-FCU device,
- Out – Actual value of input (true or false).

9.1.13 SIResistance

SIResistance is a component designed for servicing special input in the resistance read mode. SIResistance component has to be placed under LocalIO component.

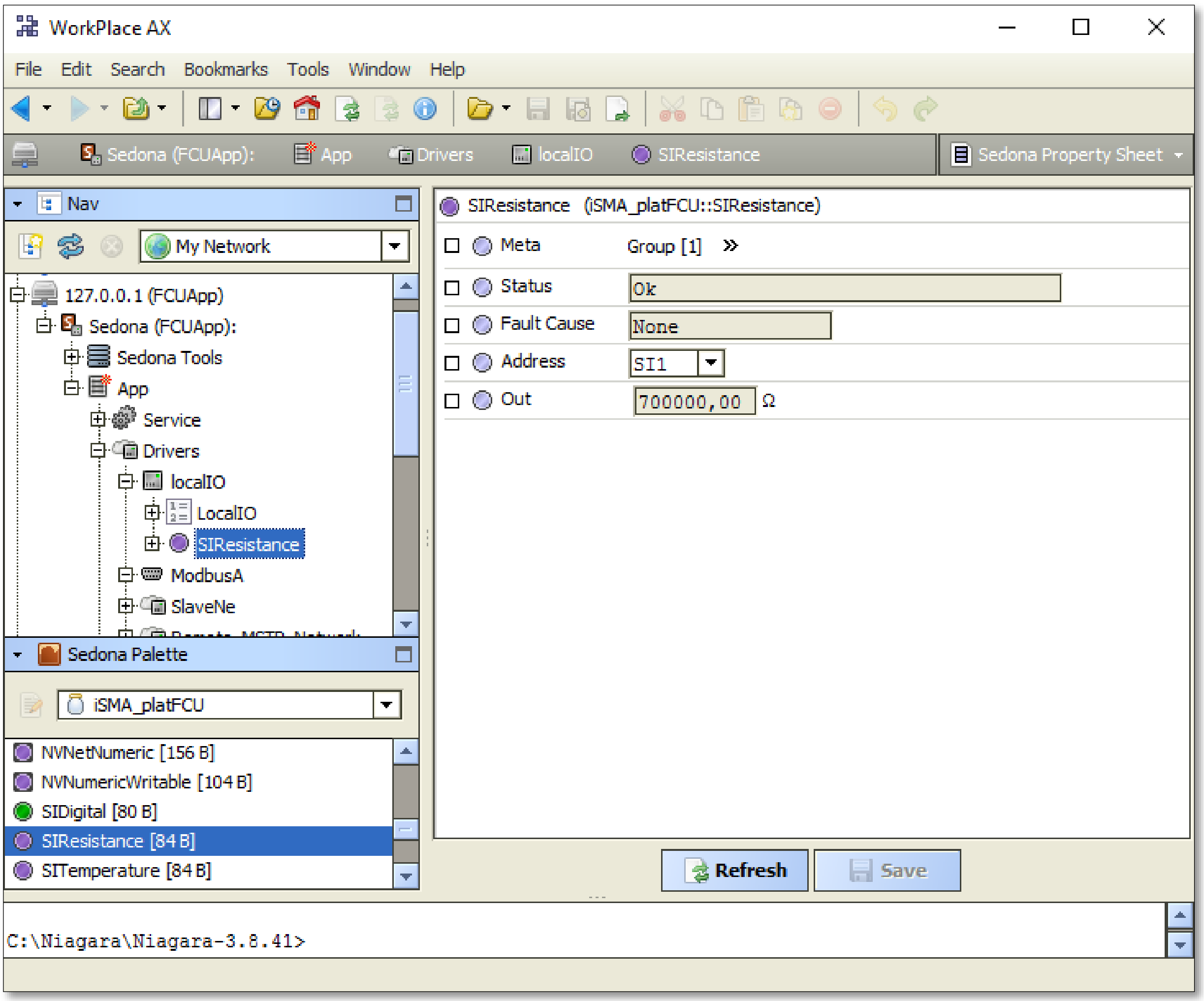


Figure 26 SIResistance component

The component has the following slots:

- Status – Status of SIResistance component,
- Fault Cause – Fault cause description,
- Address – Number of physical Input of iSMA-B-FCU device (SI1, SI2, SI3, SI4),
- Out – Actual value of Input. If voltage measurement is enabled, measuring of resistance is disabled and Out slot displays last value of resistance.

9.1.14 SITemperature

SITemperature is a component designed for servicing special Input in the temperature read mode (in order to get reliable reading, user must select the appropriate sensor type in the component LocalIOConfig). SITemperature component has to be placed under LocalIO component.

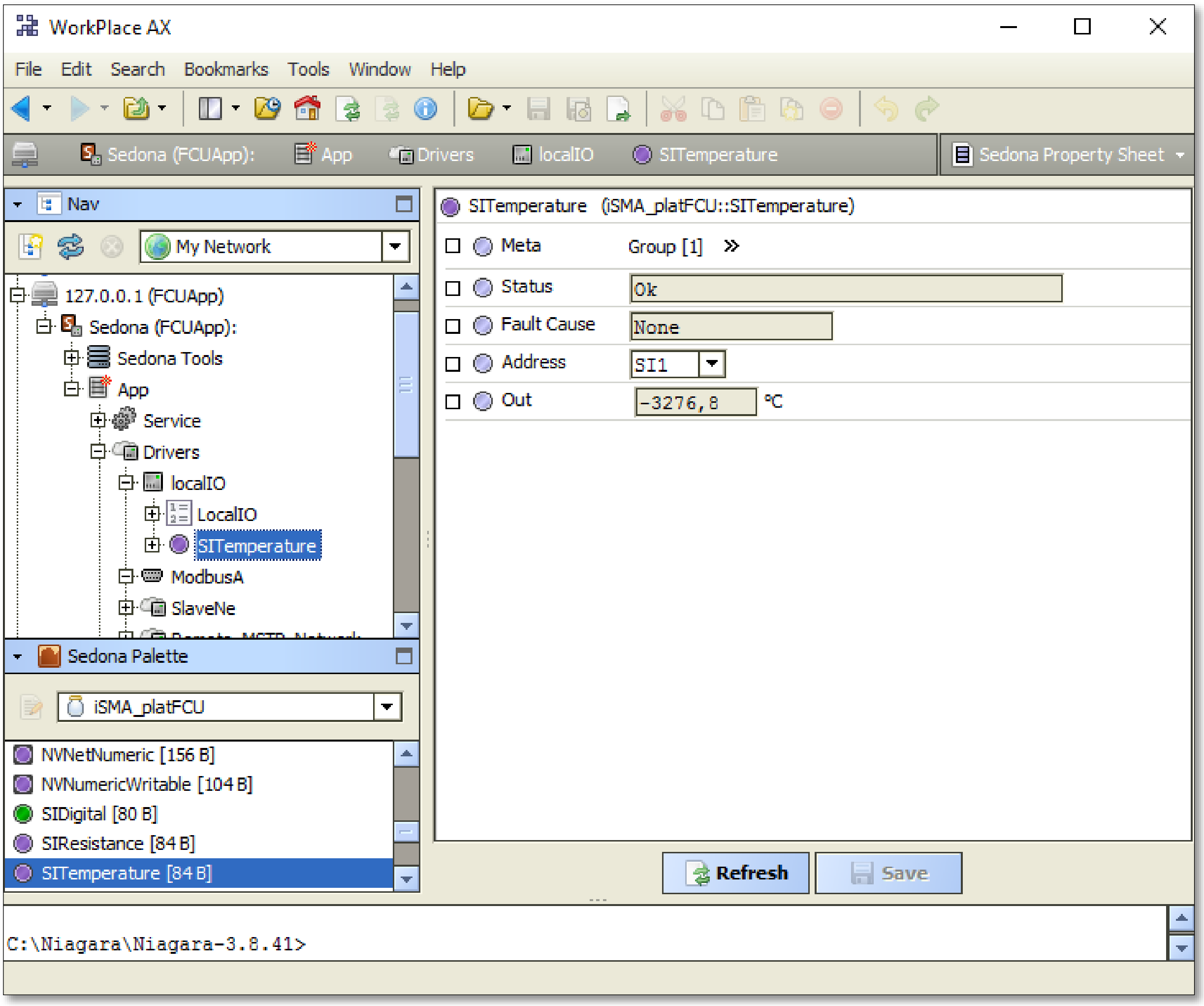


Figure 27 SITemperature component

The component has the following slots:

- Status – Status of SITemperature component,
- Fault Cause – Fault cause description,
- Address – Number of physical Input of iSMA-B-FCU device (SI1, SI2, SI3, SI4),
- Out – Actual value of Input. If voltage measurement is enabled, measuring of temperature is disabled and Out slot displays last value of temperature, measured before switching to voltage measurement.

9.1.15 SIVoltage

SIVoltage is a component designed for servicing Special Input in the voltage read mode. For proper operation of the component, type of servicing Input has to be set to Voltage_Measurement. SIVoltage component has to be placed under LocalIO component.

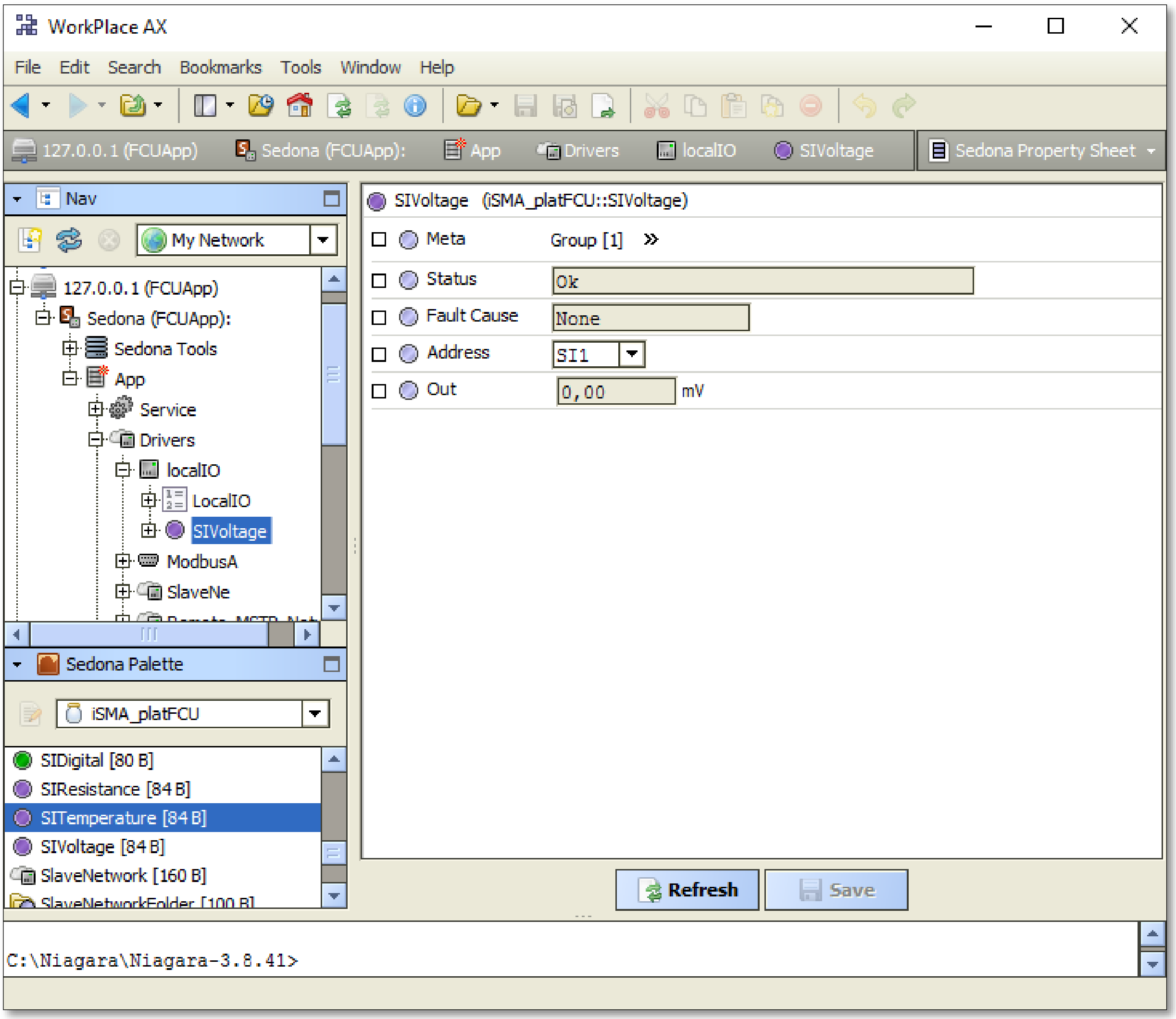


Figure 28 SIVoltage component

The component has the following slots:

- Status – Status of SIVoltage component,
- Fault Cause – Fault cause description,
- Address – Number of physical Input of iSMA-B-FCU device (SI1, SI2, SI3, SI4),
- Out – Actual value of Input. If temperature measurement is enabled, measuring of voltage is disabled and Out slot displays the last value of voltage, measured before switching to temperature measurement.

9.1.16 TODigital

TODigital is a component designed for servicing Triac Output operating in digital mode. TODigital component has to be placed under LocalIO component.

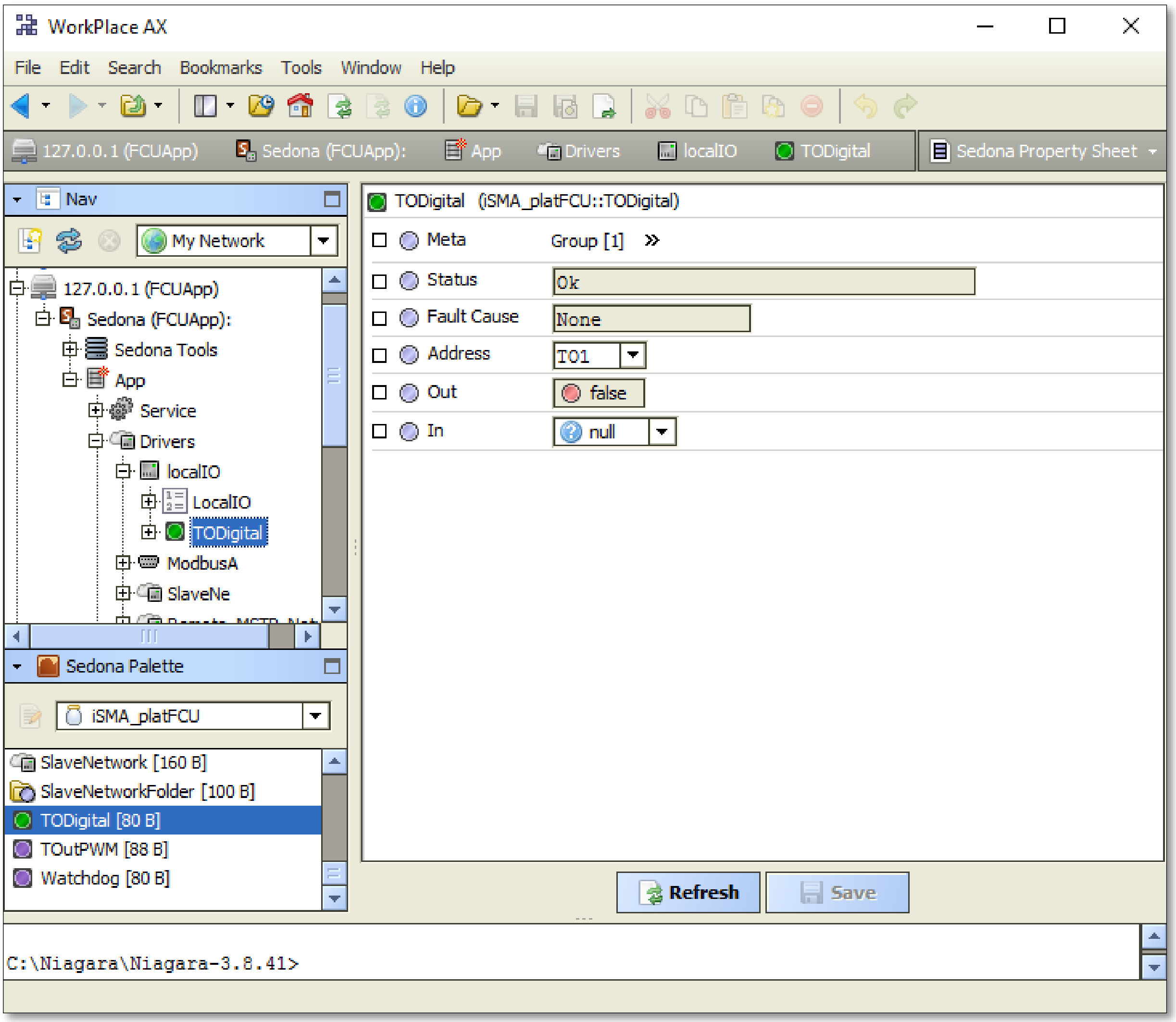


Figure 29 TODigital component

The component has the following slots:

- Status – Status of TODigital component,
- Fault Cause – Fault cause description,
- Address – Number of physical Triac Output of iSMA-B-FCU device (TO1, TO2),
- Out – Actual value of output,
- In – Input slot.
- The component offers the following actions, available under the right mouse button:
- Set– Writes value to Out slot and sends it to the device.

9.1.17 TOutPWM

TOutPWM is a component designed for servicing Triac Output working in PWM mode. TOutPWM component has to be placed under LocalIO component.

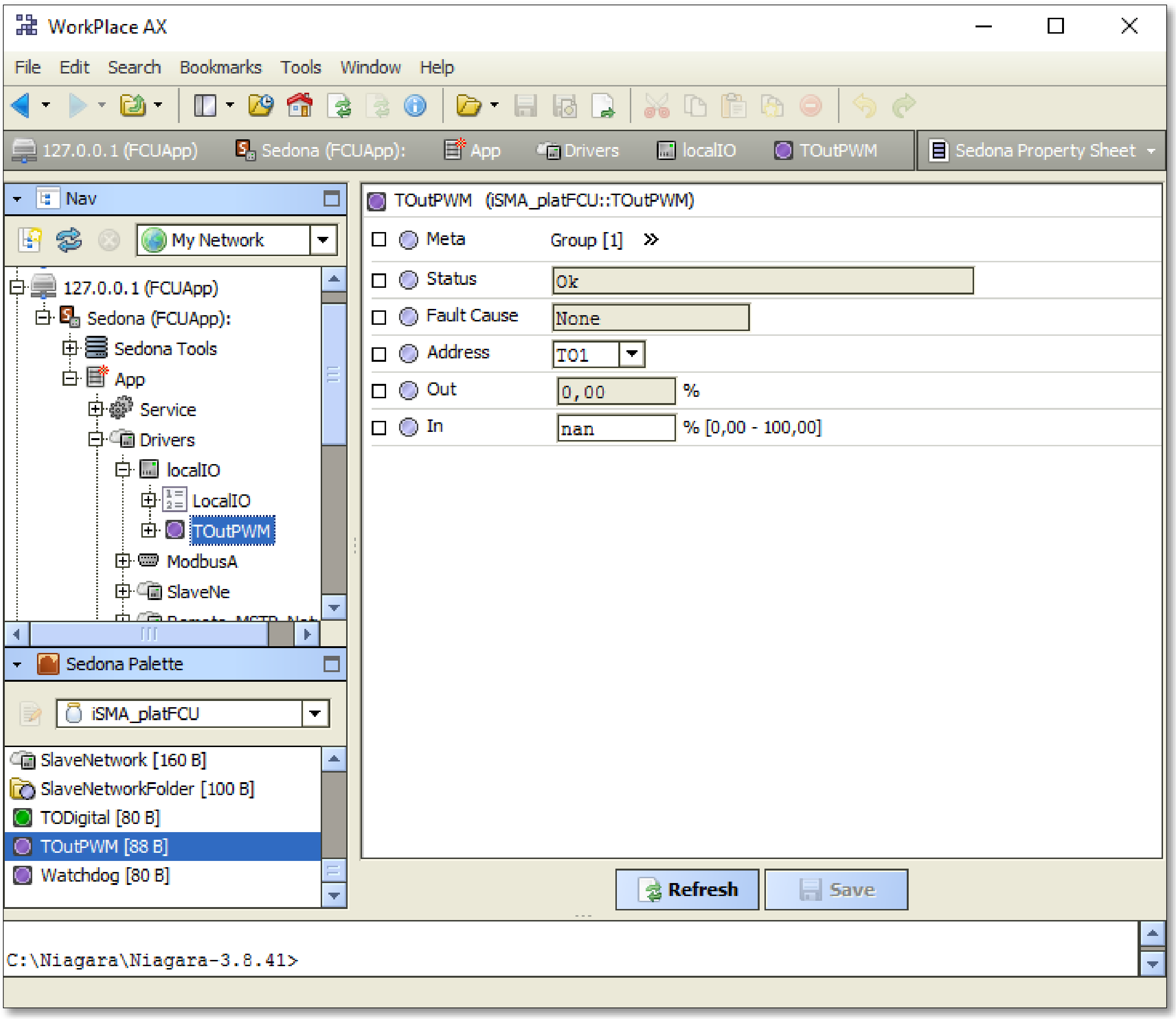


Figure 30 TOutPWM component

The component has the following slots:

- Status – Status of TOutPWM component,
- Fault Cause – Fault cause description,
- Address – Number of physical Triac Output of iSMA-B-FCU device (TO1, TO2),
- Out – Actual value of Output,
- In – Input slot.

The component offers the following action, available under the right mouse button:

- Set– Writes value to Out slot and sends it to the device.

9.1.18 DIPSwitch

DIPSwitch is a component designed for reading states of eight binary signals set by CFG DIP Switch, mounted in iSMA-B-FCU device. Using CFG DIP Switch is recommended to manage configuration of application. DIPSwitch component has to be placed under LocalIO component.

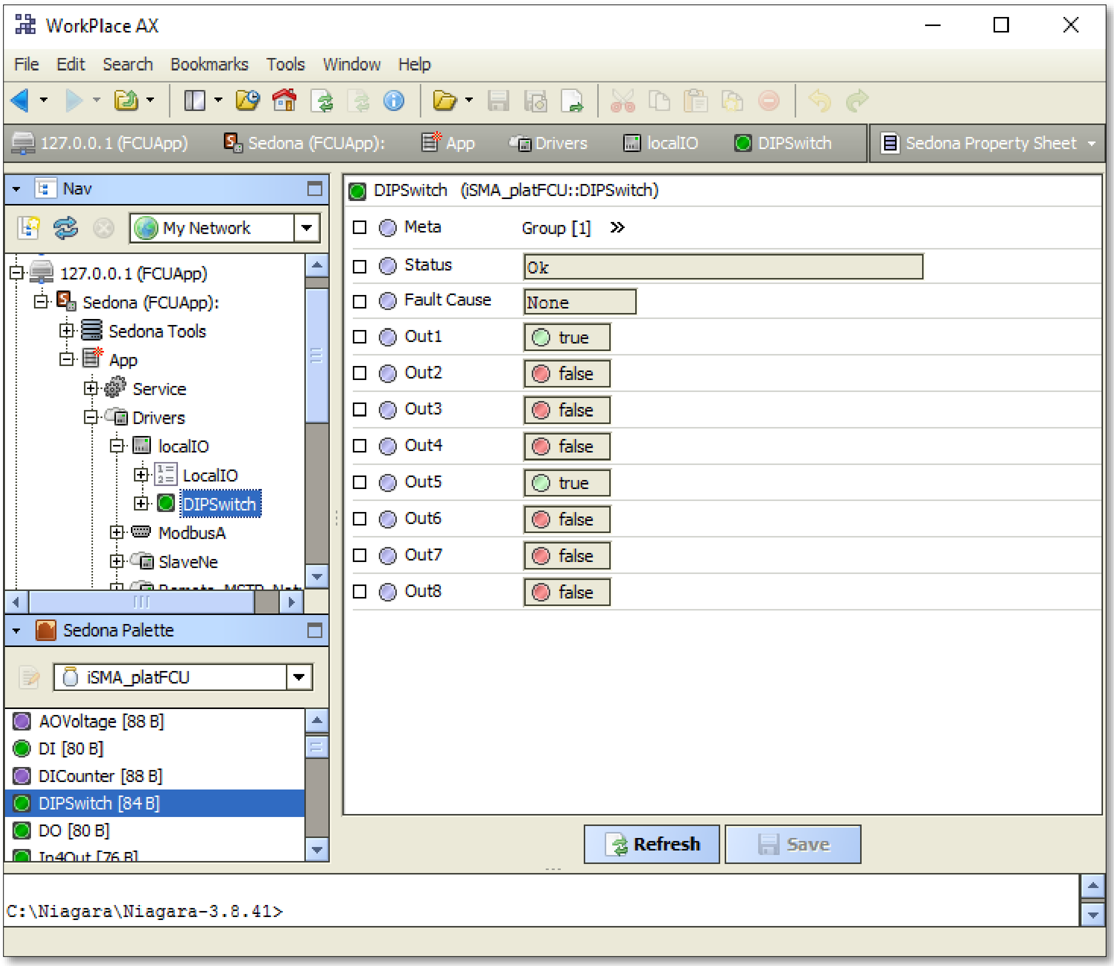


Figure 31 DIPSwitch component

The component has the following slots:

- Status – Status of DIPSwitch component,
- Fault Cause – Fault cause description,
- Out1 – Out8 – Actual states of each binary signal from DIP Switch CFG, according to the figure bellow:

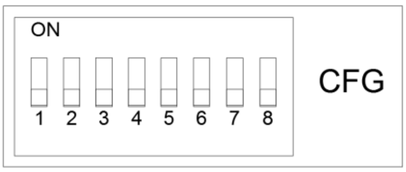


Figure 32 DIPSwitch signals

9.1.19 In4Out

In4Out is a component servicing Digital Input 4 operating as Digital Output. In4Out component has to be placed under LocalIO component.

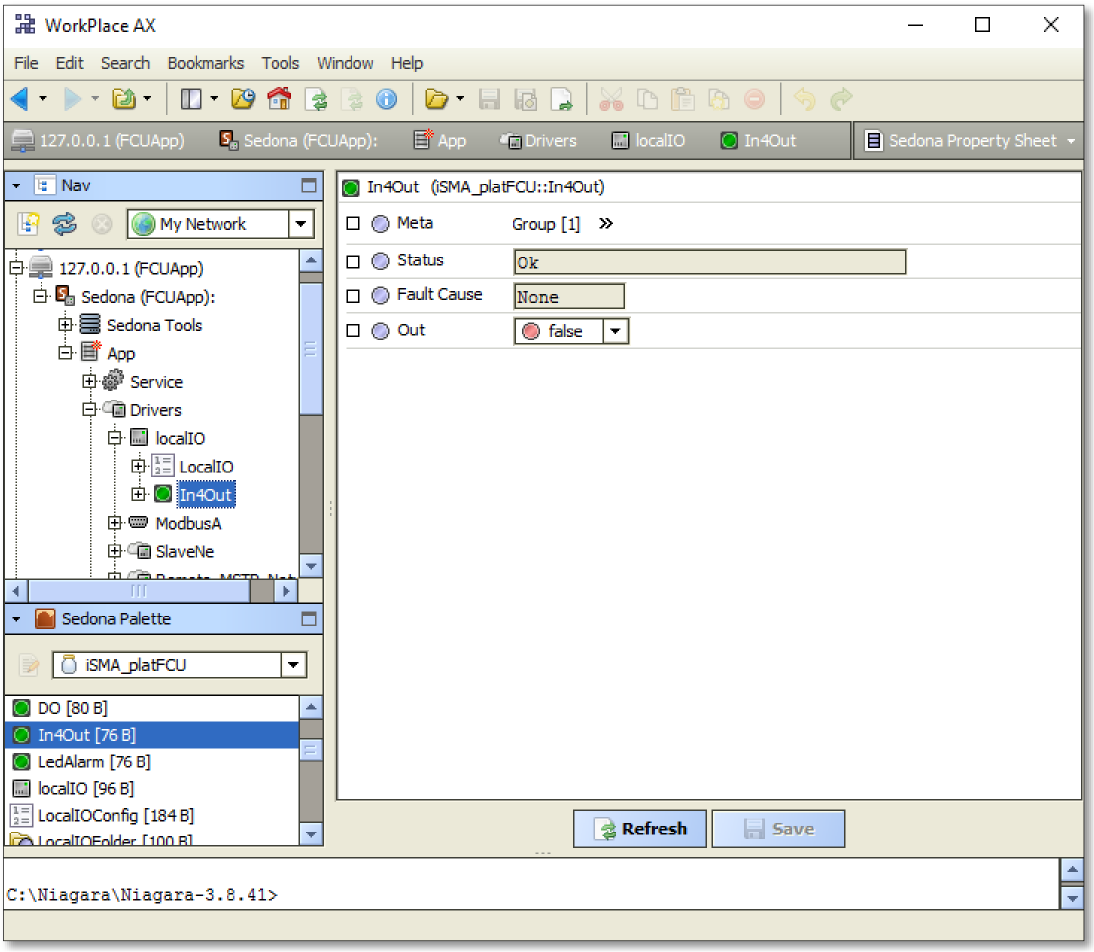


Figure 33 In4Out component

The component has the following slots:

- Status – Status of In4Out component,
- Fault Cause – Fault cause description,
- Out – Actual state of Output which is sent to iSMA-B-FCU device. State “true” overrides low state (I4 terminal is connected to G0 ground – voltage between them is equal to 0 V DC). State “false” does not affect the operation of the component – voltage between I4 and G0 equals 5V DC.

The component offers the following action, available under the right mouse button:

- Set– Writes value to Out slot and sends it to the device.

9.1.20 LedAlarm

LedAlarm is a component for servicing Alarm LED, mounted in iSMA-B-FCU device. It allows for signalizing states of operating iSMA-B-FCU devices defined in the application. For example, it can be used for signalizing alarms.

LedAlarm component has to be placed under LocalIO component.

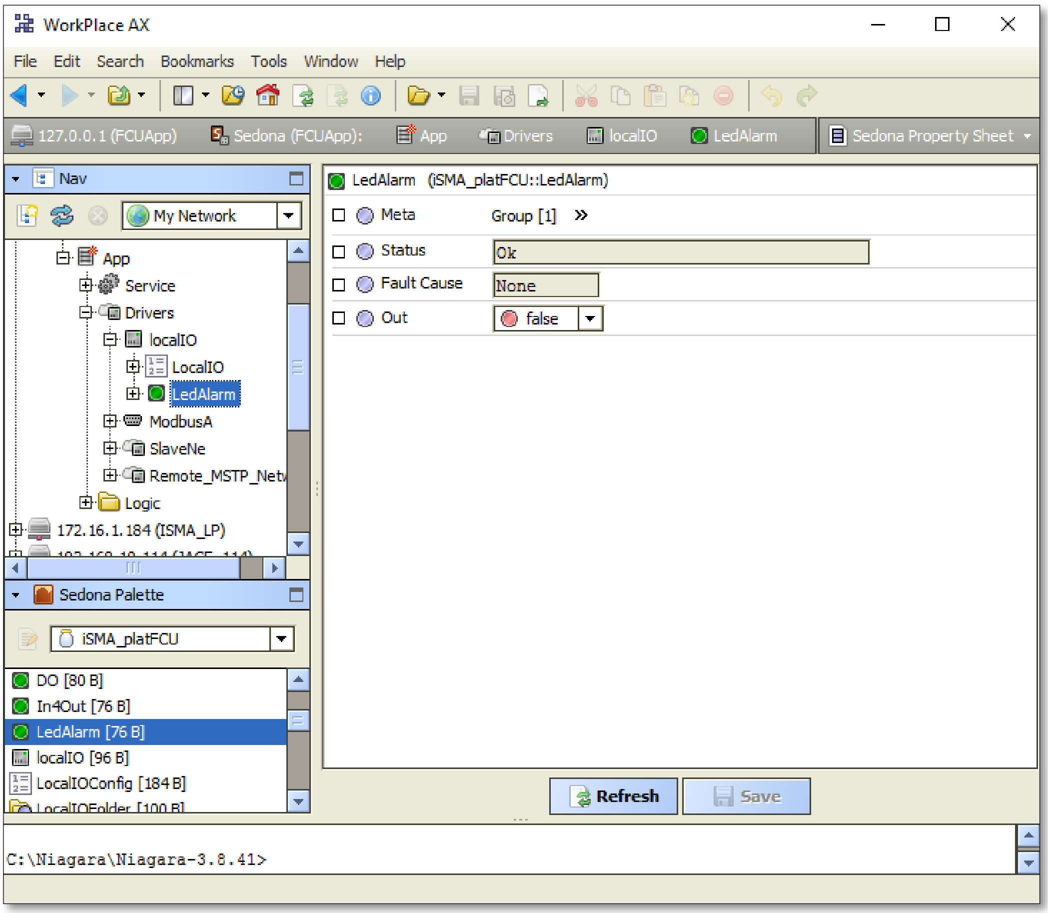


Figure 34 LedAlarm component

The component has the following slots:

- Status – Status of LedAlarm component,
- Fault Cause – Fault cause description,
- Out – Actual state of Alarm LED, which is sent to iSMA-B-FCU device. State “true” – LED lights; state “false” – LED is off.

The component offers the following action, available under the right mouse button:

- Set – Writes value to Out slot and sends it to the device.

9.1.21 Watchdog

Watchdog is a component designed for controlling communication by RS485 port (COM1) and USB (only for communication with WorkPlace using SOX protocol or in case of read/write Modbus Registers using USB connection). The component has to be placed under LocalIO component.

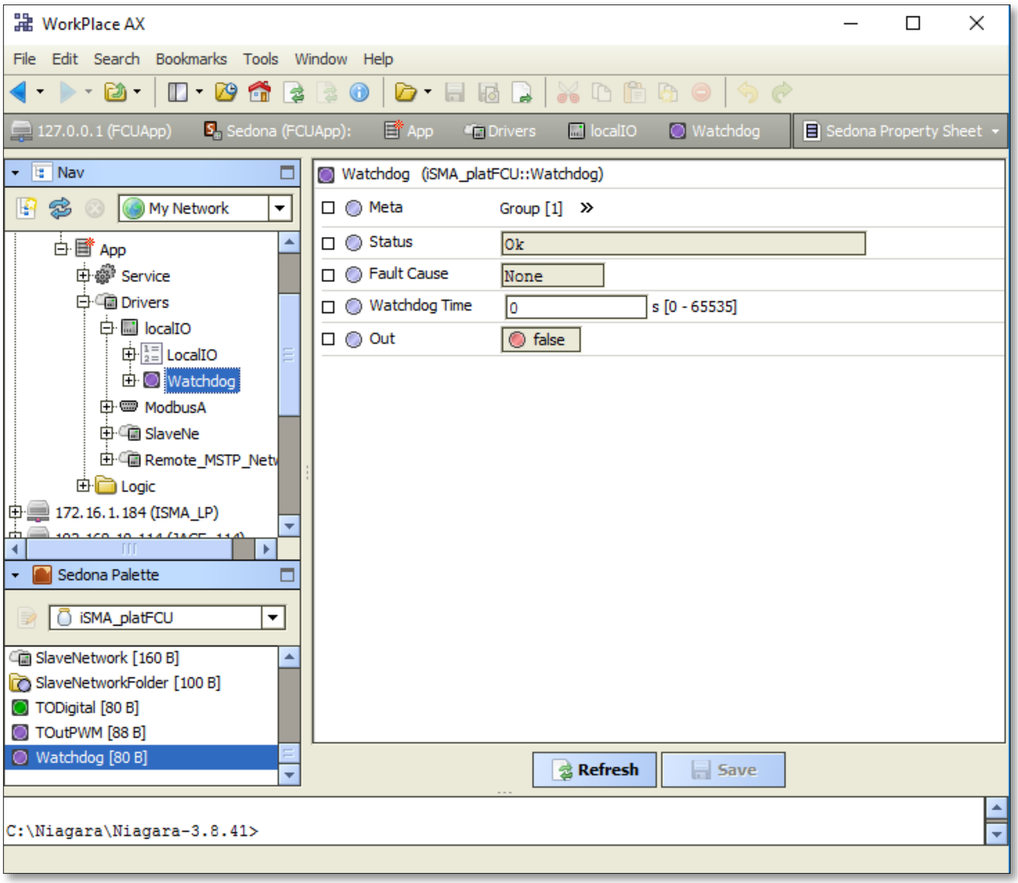


Figure 35 Watchdog component

The component has the following slots:

- Status – Status of LedAlarm component,
- Fault Cause – Fault cause description,
- Watchdog Time - Time between reception of valid messages, value 0 disables this function,
- Out – Output of Watchdog. When Watchdog is disabled (Watchdog Time slot is set to 0) or time between reception of valid messages does not exceed Watchdog Time – Out is set to “false”. When time between reception of valid messages does exceed Watchdog Time – Out is set to “true”.

The component offers the following action, available under the right mouse button:

- Set – Writes value to slot Watchdog Time.

9.2 Slave Network

Slave Network is used for communication of iSMA-B-FCU with upper level systems (for example BMS). Communication can be realized by Modbus RTU / ASCII or BACnet MSTP protocol (depends of Dip Switch PROTOCOL configuration).

9.2.1 Slave Network Component

Slave Network component is used to manage BACnet MSTP or Modbus RTU/ASCII protocols, using RS485 port. Slave Network component has to be placed under Drivers component.

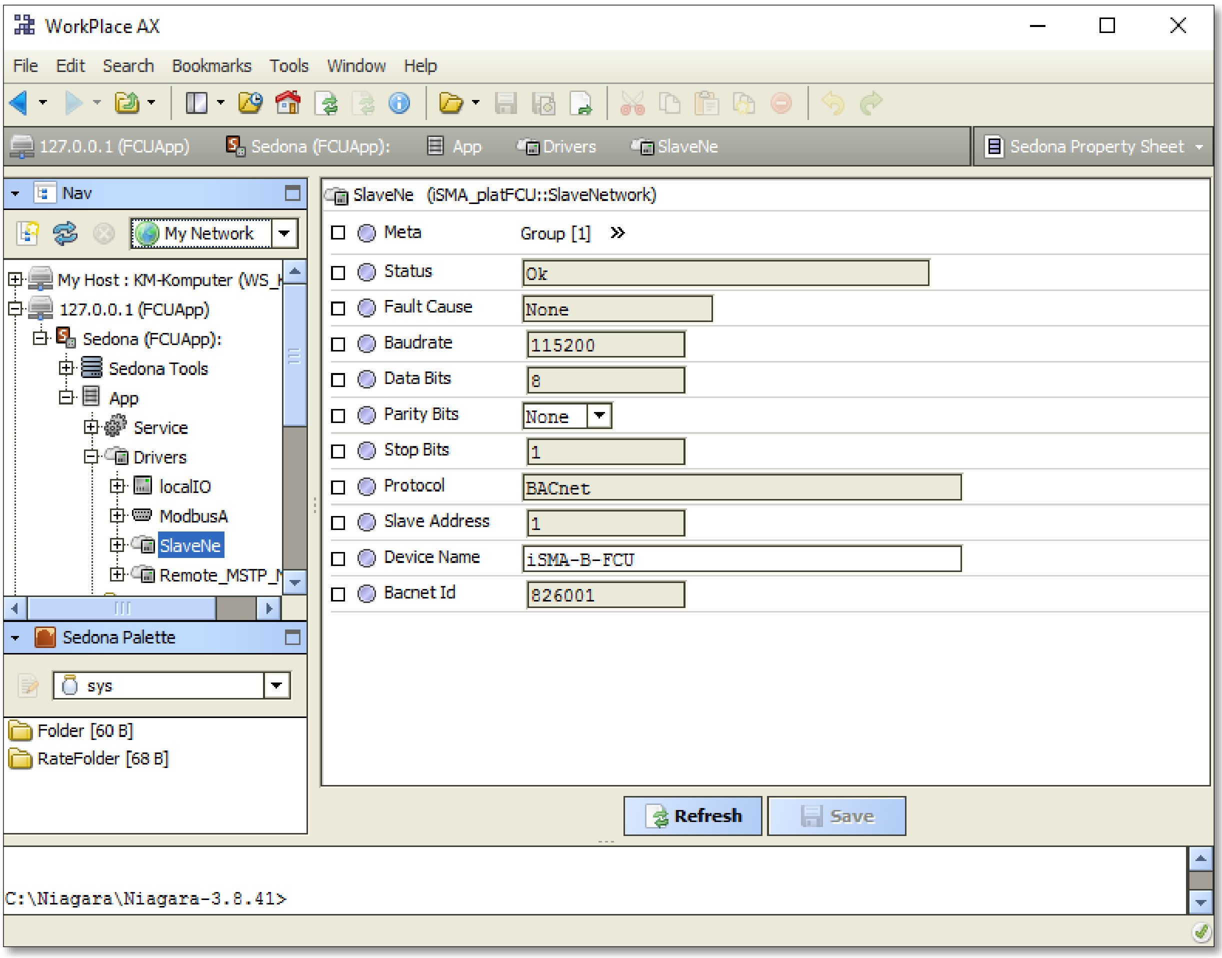


Figure 36 Slave Network component

The component has the following slots:

- Status – Network status,
- Fault Cause – Fault cause description,
- Baud rate – Baud rate of communication (set by PROTOCOL Dip Switch),
- Data Bits – Displays number of data bits. Default set of value is 8,
- Parity Bits – Configuration of Parity Bits: Parity_Disabled, Odd, Even,
- Stop Bits – Displays number of stop bits. Default set of value is 1,
- Protocol – Displays protocol used for communication: Modbus or BACnet (set by PROTOCOL Dip Switch),
- Slave Address – Displays actual address of a device (set by MAC Dip Switch),
- Device Name – BACnet device name,

  BACnet Id – Actual BACnet Id. Default value equals to 826000 + Slave Address. BACnet Id can be set using “Set BACnet ID” action.

The component offers the following action, available under the right mouse button:

- Set BACnet ID – sets BACnet ID of device.

9.2.2 NV Net components

NV Net Components (Non-Volatile Net): the value of these components can be recorded in an EEPROM device’s non-volatile memory. Whenever a device is restarted or the power is down, the values of NV Net components remain saved. These components can also be used to sending values using BACnet MSTP or Modbus RTU protocols (depending on Dip Switch PROTOCOL configuration). The device has two types of NV components, broken down by the type of variables they support.

The components include:

- Boolean variables - NVNetBoolean component
- Numeric (float) variables - NVNetNumeric component

All NV Net components have to be placed under Slave Network component.

Note: iSMA-B-FCU device supports up to 200 NVNetNumeric or up to 200 NVNetBoolean components. Number of free NV memory cells can be checked under platform properties.

Note: Method of calculating memory cells for NV components is described in Plat service section. Since the values of the components are not stored in the Sedona application but in the non-volatile memory of the device, in situation when an application is copied between two devices, Output values are not saved and will assume the values stored in the local EEPROM memory. To copy NV Net components to another device along with their values (e.g. setpoint), use global actions of the plat component:

Step 1 - Use global action "Copy From NV The Default / Copy From NV To User"

Step 2 - Save the application and copy it to another device

Step 3 - Use global action on the target device "Copy From Default To NV / Copy From User To Nv".

9.2.3 MU MdsMt l ddqb Bnl onmdms

NVNetNumeric is a component that stores the Output value in non-volatile memory of an EEPROM device. After rebooting the device or power failure, the component value is restored from this particular memory. The NVNetNumeric component occupies a single memory cell of the Numeric type.

The component has to be placed under Slave Network component.

Note: Method of calculating memory cells for NV components is described in Plat service section.

The NVNetNumeric component is used also to integrate Numeric (float) variables from various sources. This is done using the "reverse following the link" function. The Out slot is connected to the In slots of various protocols, for example BACnet or Modbus variable. After changing a value in one of the components, the device will perform the Set action on the NV Net component to synchronize the values in all the connected components. This option is enabled only when Link Set slot is set to “true”.

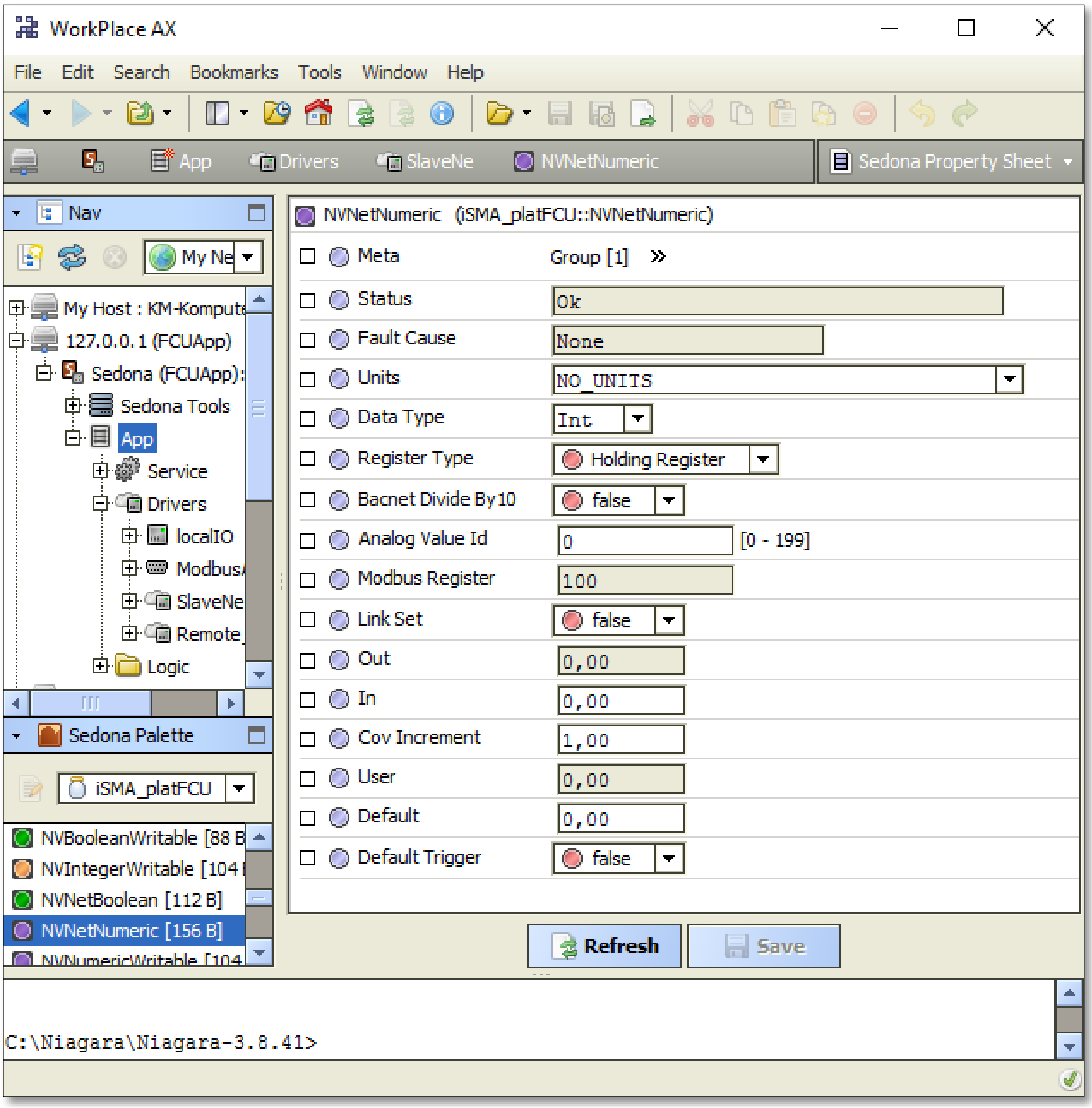


Figure 37 NVNetNumeric component

The component has the following slots:

- Status – Actual status of the component,
- Fault Cause – Fault case description,
- Units – BACnet unit of point,
- Data Type – Type of data: Int, Long, Float, SInt, SLong,
  WARNING: Data type Long, Slong and Float consume 2 registers, so next network variable address must leave one address gap.
- Register Type - Register type for Master Device: Holding Register – read/write, Input Register – read only,
- BACnet Divide By 10 – If slot is set to “true” and device communicates by BACnet protocol, value of read/write point is equal to value received from Out slot divided by 10,
- Analog Value Id – Id of BACnet object,
- Modbus Register – Displays number of Modbus register,
  - Modbus Register = 100 + Analog Value Id
- Link Set – "reverse following the link" function is activated, recalls action Set in component

linked to “In” slot
- Out – Output slot,
- In – Input Slot,
- Cov Increment – minimal change of value,
- User – User value slot (set by Set action),
- Default – Default value slot (set by global command from plat action),
- Default Trigger – On rising edge copies values from Default slot to Out slot.

The component offers the following action, available under the right mouse button:
- Set – The action sets User slot and In slot if there is no link on In slot.

9.2.4 MU MdsAnnkd` mBnl onmdms

NVNetBoolean is a component that stores the Output value in non-volatile memory of an EEPROM device. After rebooting the device or power failure, from this particular memory the component’s value is restored. The NVNetBoolean component occupies a single memory cell of the Boolean type.

The component has to be placed under Slave Network component.

Note: Method of calculating memory cells for NV components is described in Plat service section.

The NVNetBoolean component is used also to integrate Boolean variables from various sources. This is done using the "reverse following the link" function. The Out slot is connected to the In slots of various protocols, for example BACnet or Modbus variable. After changing a value in one of the components, the device will perform the Set action on the NV Net component to synchronize the values in all the connected components. This option is enabled only when Link Set slot is set to “true”.

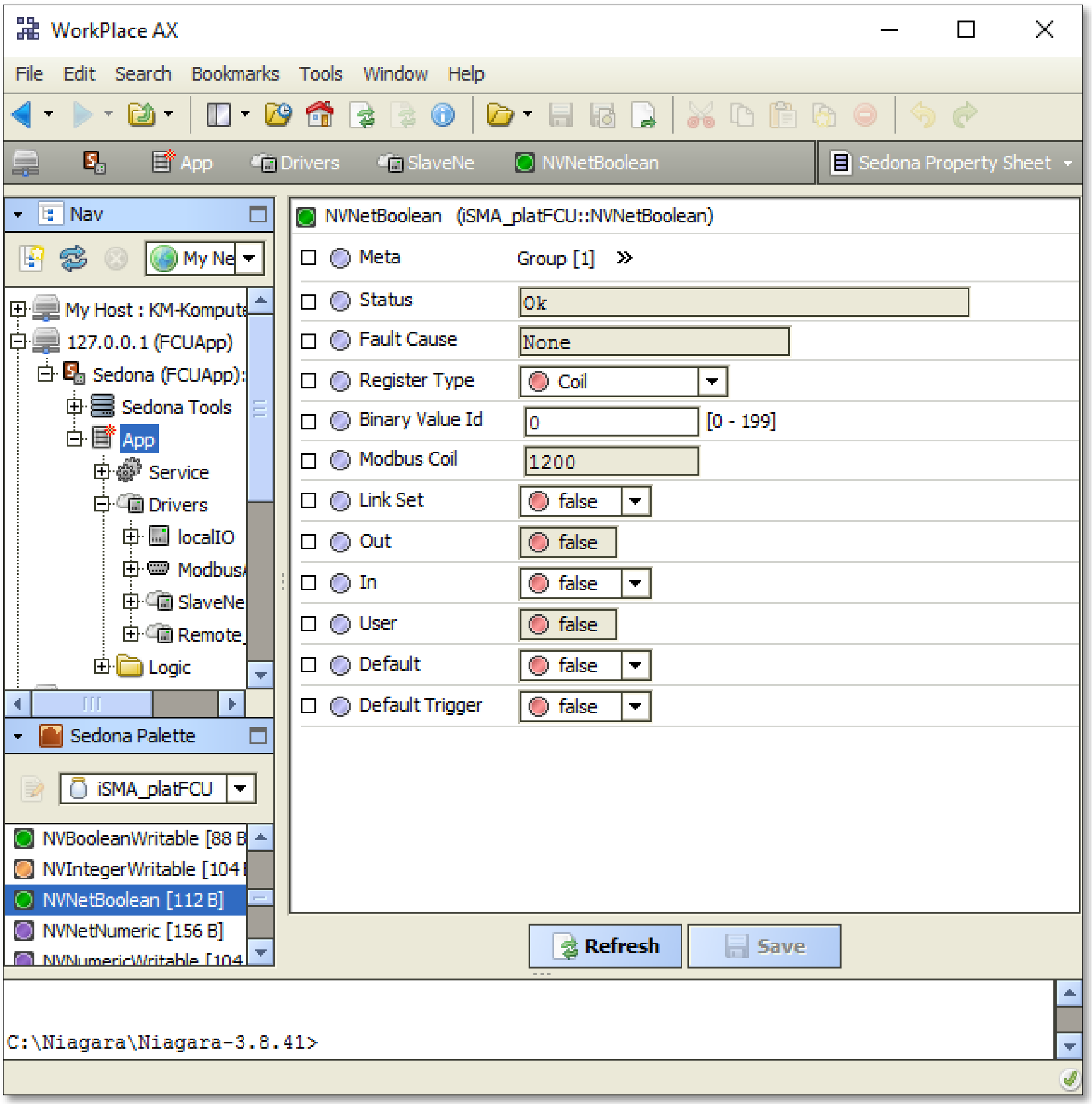


Figure 38 NVNetBoolean component

The component has the following slots:

- Status – Actual status of the component,
- Fault Cause – Fault case description,
- Register Type - Register type for Master Device: Coil – read/write, Discrete Input – read only,
- Binary Value Id – Id of BACnet object,
- Modbus Coil – Displays number of Modbus Coil
- Modbus Coil = 1200 + Binary Value Id,
- Link Set – "reverse following the link" function is activated, recalls action Set in component linked to “In” slot
- Out – Output slot,
- In – Input Slot,
- User – User value slot (set by Set action),
- Default – Default value slot (set by global command from plat action),

- Default Trigger – On rising edge copies values from Default slot to Out slot.

The component offers the following action, available under the right mouse button:
- Set – This action sets User slot and In slot if there is no link on In slot.

10 Modbus Async Network kit

This section provides a collection of procedures to be used for the iSMA-B-FCU Modbus drivers to build networks of devices with Modbus points. The iSMA-B-FCU device has two RS485 ports. Port COM2 (with RJ12 connector) can be used as a Modbus RTU/ASCII master. There is no software limit of devices connected to Modbus Async bus, but total number of all Modbus points cannot exceed 200.

10.1 Modbus Async Network component

Modbus network is a main component responsible for servicing RS485 port (COM2). The component has to be placed under Drivers folder. ModbusNetwork sets parameters such as communication baud rate and data format, testing etc. and maintains statistical data.

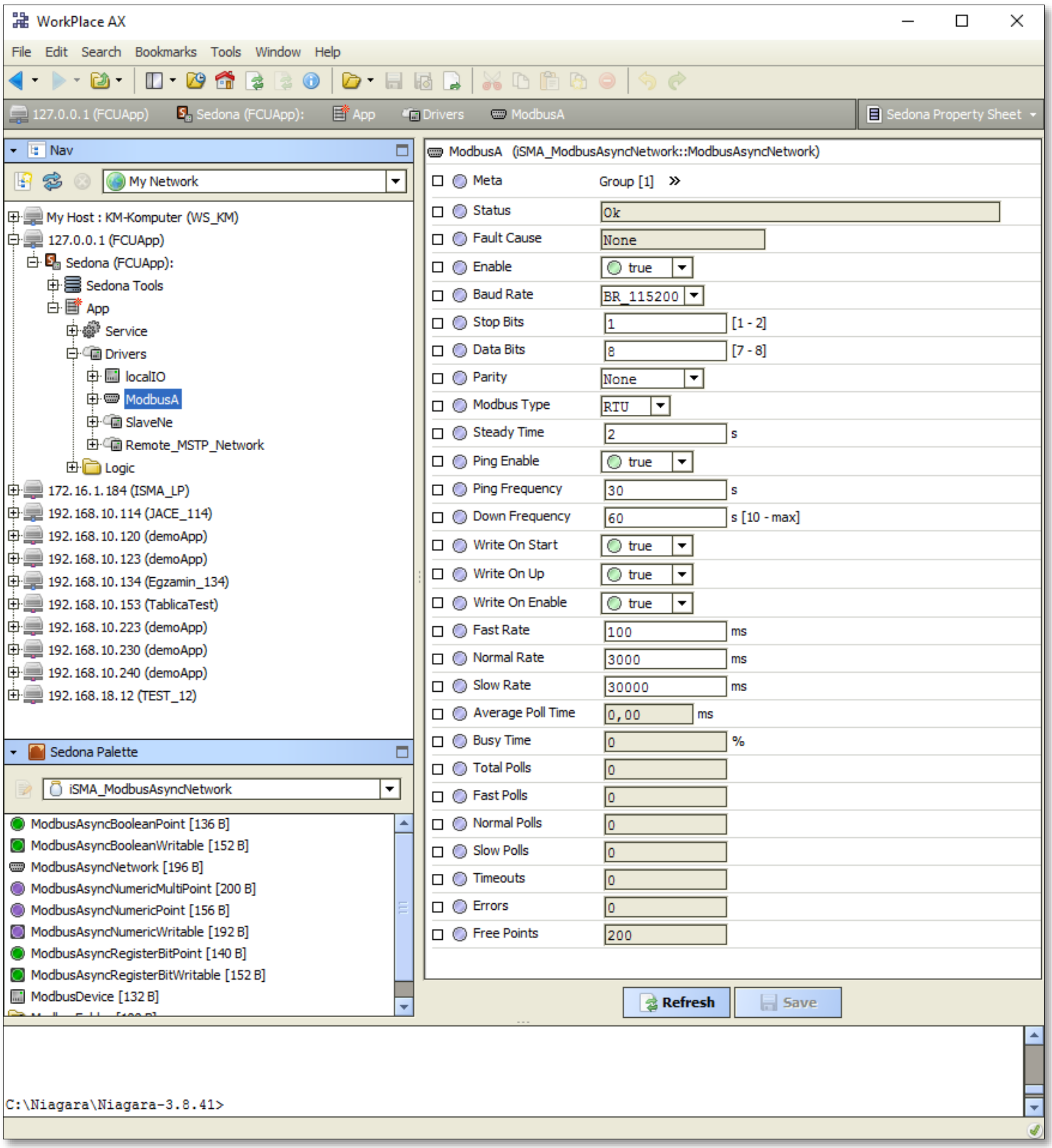


Figure 39 ModbusAsyncNetwork component

The component has the following slots:

- Status – Network status,
- Fault Cause – Fault cause description,
- Enable – this option switches on or off Modbus network ("true"- Network enabled, "false"- Network disabled),
- BaudRate – Modbus RS485 port baud rate, available options: 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps,
- Stop Bits – Stop bit definition, available options: 1 bit, 2 bits
- Data Bits – Data bits definition, available options: 7 bits or 8 bits,
- Parity – Parity bit definition, available options: None, Odd, Even, Always1, Always0,
- Modbus Type – Modbus type definition, available options: RTU or ASCII
- Steady Time – Network start up delay time after power up or reset,

- Ping Enable – Enables devices connection testing function,
- Ping Frequency – Time lapse between testing message and check device connection,
- Down Frequency – Time lapse between testing message for devices or points which have got status down, min. value 10s
- Write On Start – Writes action in device “writable” components in Modbus network after reset or power up,
- Write On Up - Writes action in device “writable” components in Modbus network after the connection with Modbus device is restored,
- Write On Enable - Writes action in device “writable” components in Modbus network after enabling the device,
- Fast Rate – Time between messages in “Fast” mode poll frequency,
- Normal Rate – Time between messages in “Normal” mode poll frequency,
- Slow Rate – Time between messages in “Slow” mode poll frequency,
- Average Poll Time – Average time for sending/receipt of one message,
- Busy Time – Percentage of Modbus network usage,
- Total Polls – Total number of messages,
- Fast Polls – Number of messages sent in “Fast” mode,
- Normal Polls – Number of messages sent in “Normal” mode,
- Slow Polls – Number of messages sent in “Slow” mode,
- Timeouts – Number of lost messages, difference between messages sent and received,
- Errors – Number of error messages (for example with wrong CRC)
- Free points – Number of available physical points in Modbus Network.

The component offers the following actions, available under the right mouse button:

- Reset Stats – Resets statistics of Modbus Async Network,
- Enable – Enables Modbus Async Network,
- Disable – Disables Modbus Async Network.

10.2 Modbus Async Device component

ModbusDevice is a component responsible for servicing physical device connected to Modbus network. The component has Ping action available under the right mouse button, which sends test message to device to check the status of the device. Each ModbusDevice has a “Ping Address” container slot with 3 properties slots (Address Format, Ping Address Reg, Ping Type). These properties specify a particular data address (either Input register or holding register) to be used as the device status test (meaning “Monitor” ping requests). Ping requests are generated at the network-level by the configurable network monitor (ModbusNetwork -> Ping Enabled). When enabled, a network’s monitor periodically pings (queries) this address. Reception of any response from the device, including an exception response, is considered proof of communication, and the Modbus client device is no longer considered “down” if it had been previously marked “down”.

Component has to be placed under Modbus Async Network component.

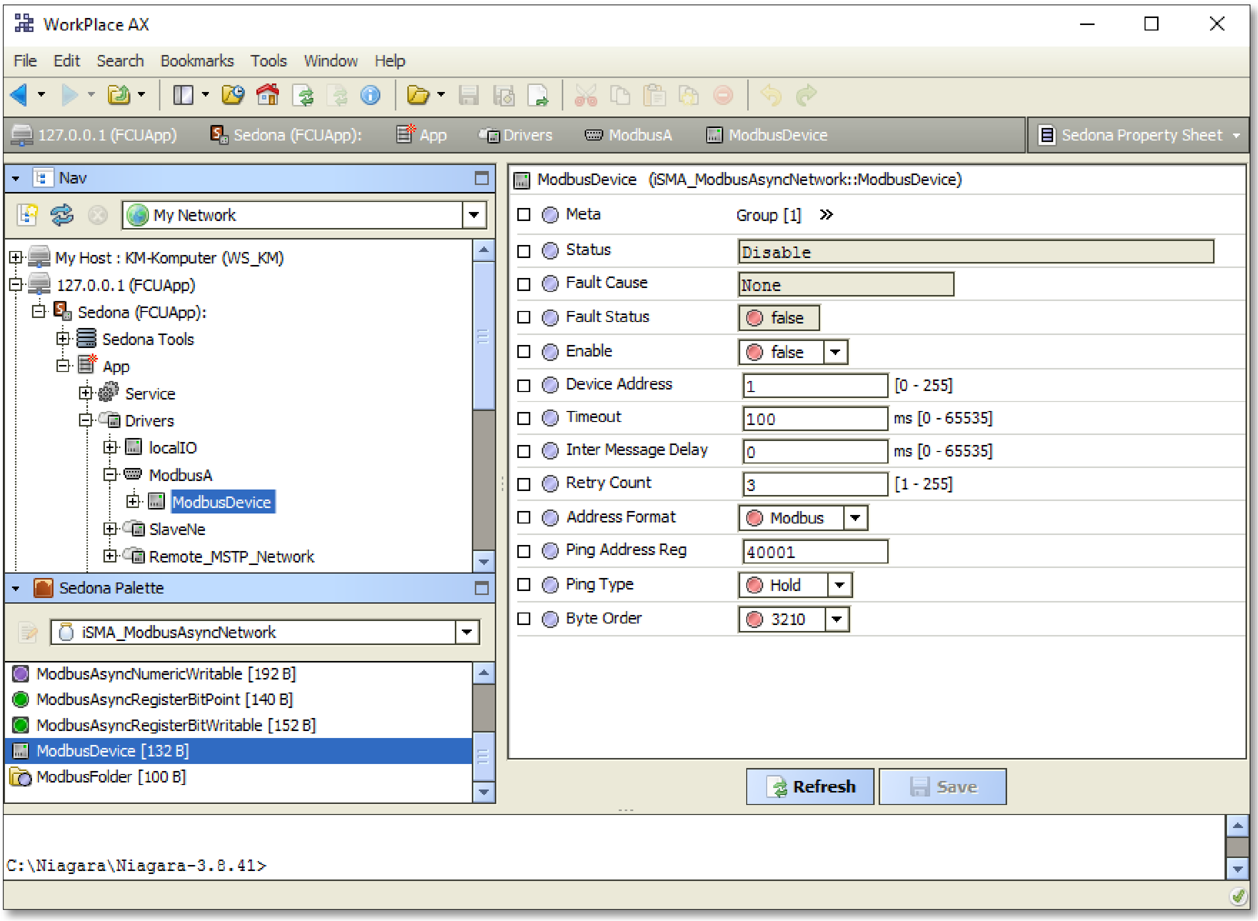


Figure 40 ModbusAsyncDevice component

The component has the following slots:

- Status – Device actual status (read Only),
- Fault Cause – Fault cause description,
- Fault Status – device error status ("true" – device communication error),
- Enable – Enable / Disable device,
- Device Address – Modbus device physical address (0 – network broadcast address, 1-248 addressing range),
- Timeout – Max. device response time calculated from device request,
- Inter Message Delay – Time between sending messages to a device,
- Retry Count – Max. error messages (CRC error, lost messages),
- Address Format – Modbus address format (Modbus, Decimal),
- Ping Address Reg – Any register number Input or Holding type, which will be read for device connection test,
- Ping Type – Tested register type: Input/Holding,
- Byte Order – Byte order reading 32 bit: 3210 (Big endian), 1032 (Little endian).

The component offers the following action, available under the right mouse button:

- Ping – Sends test message to device to check status of the device.

10.3 Modbus Async Boolean Point

Modbus Async Boolean Point is a component responsible for reading Boolean values from the device. The component has to be placed under Modbus Async Device component.

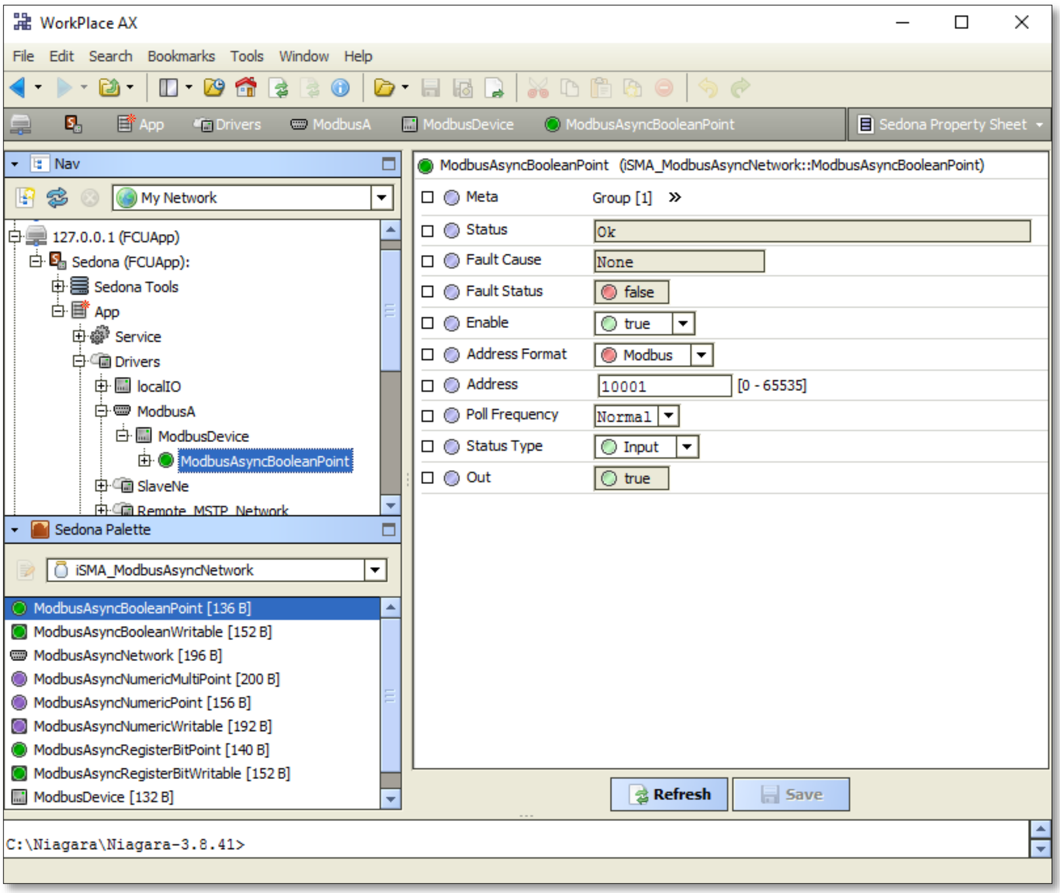


Figure 41 ModbusAsyncBooleanPoint component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Address Format – Register address format, available options: Modbus, Decimal,
- Address – Register address,
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Status Type – Type of reading register, available options: Input, Coil
- Out – Current value of read registry.

The component offers the following action, available under the right mouse button:

- Read – Action enforces reading of the point.

10.4 Modbus Async Boolean Writtable

Modbus Async Boolean Writable is a component responsible for sending and reading Boolean values from the device. Component has to be placed under Modbus Async Device component.

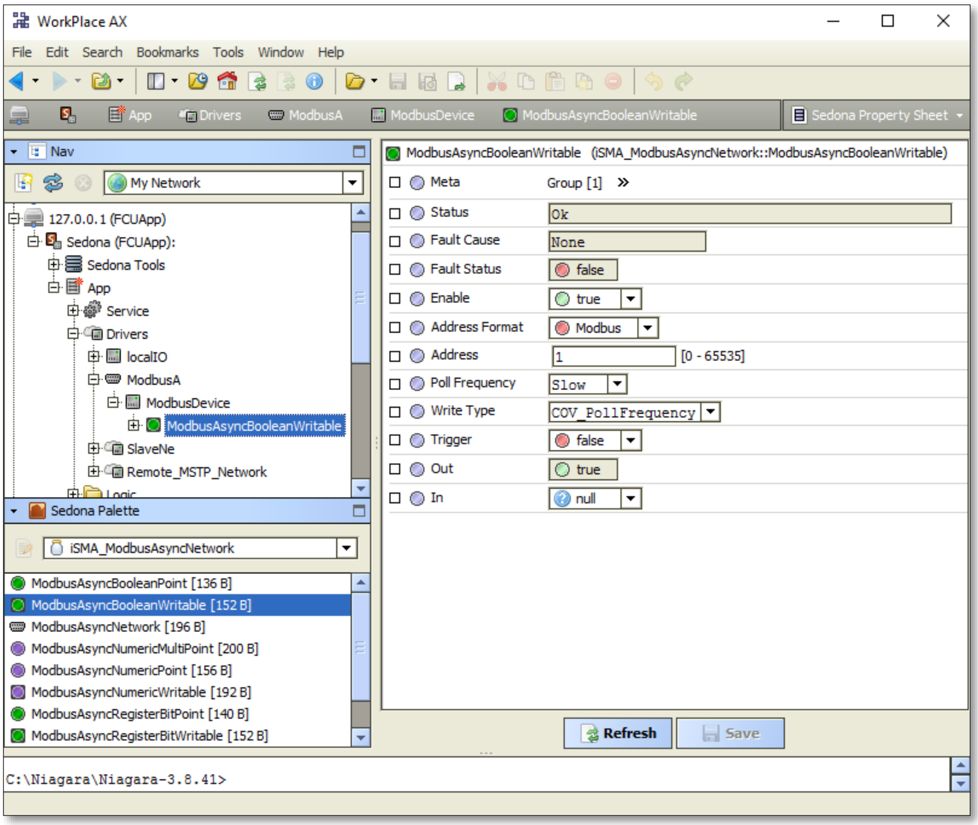


Figure 42 ModbusAsyncBooleanWritable component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status ("true" – point read/write error),
- Enable – Point enabled/disabled ("true"- Point enable, "false"- Point disable),
- Address Format – Register address format, available options: Modbus, Decimal,
- Address – Register address,
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Write Type – Writing mode, available options: COV – only on Input change, COV_PollFequrency – on Input change and periodically, PollFequrency – only periodically, COV_LinkSet – only on Input change, using the "reverse following the link" function
- Trigger – Trigger of remote enforcement of sending (on rising edge),
- Out – Output slot, current value of read/write registry,
- In – Input slot.

- The component offers the following actions, available under the right mouse button:
- Set True/Set False – Writes value to In slot and sends it to the device (not active when In slot has a link connected),
- Write – Sends value from In sloto the device,
- Send Value – Sends user value from pop-up window, without changing In slot
- Read – Reads value from device and sends to Out slot.

10.5 Modbus Async Numeric Multi Point

Modbus Async Numeric Multi Point is a component responsible for reading up to 8-16 bitsą registers from the device in one message. Component uses 0x16 Modbus command. Component has to be placed under Modbus Async Device component.

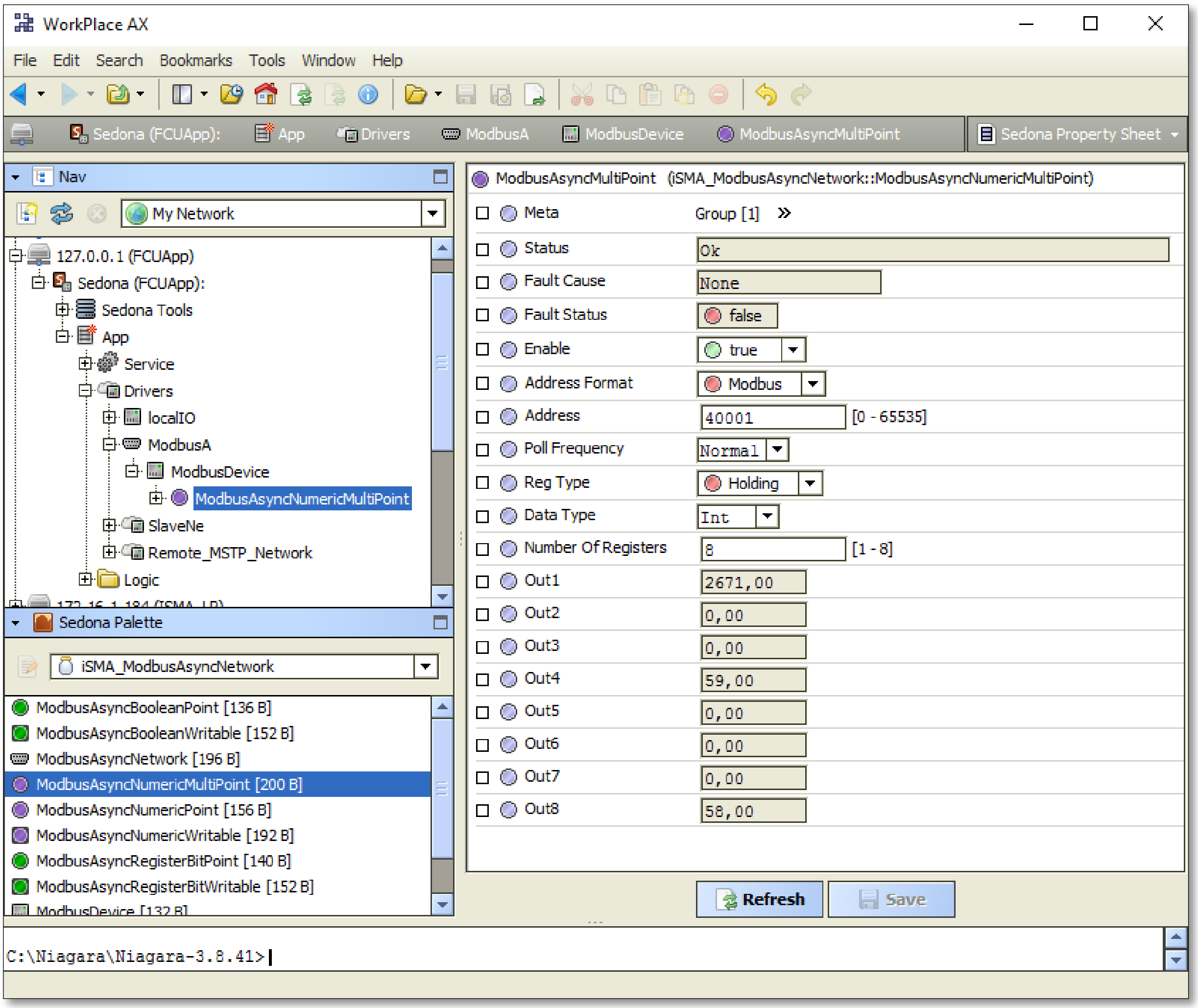


Figure 43 ModbusAsyncNumericMultiPoint

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status ("true" – point read error),
- Enable – Point enabled/disabled ("true"- Point enabled, "false"- Point disabled),
- Address Format – Register address format, available options: Modbus, Decimal,
- Address – Register address,
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Reg Type – Type of reading register, available options: Input, Holding,
- Data Type – Reading data type: Int (unsigned values), Sint (signed values),

- Number Of Registers – Number of reading register in one messages,
- Out1 – Out8 – Current values of read registry.

The component offers the following action, available under the right mouse button:

- Read – Action enforces the reading of the point.

10.6 Modbus Async Numeric Point

Modbus Async Numeric Point is a component responsible for reading numeric values from the device. The component has to be placed under Modbus Async Device component.

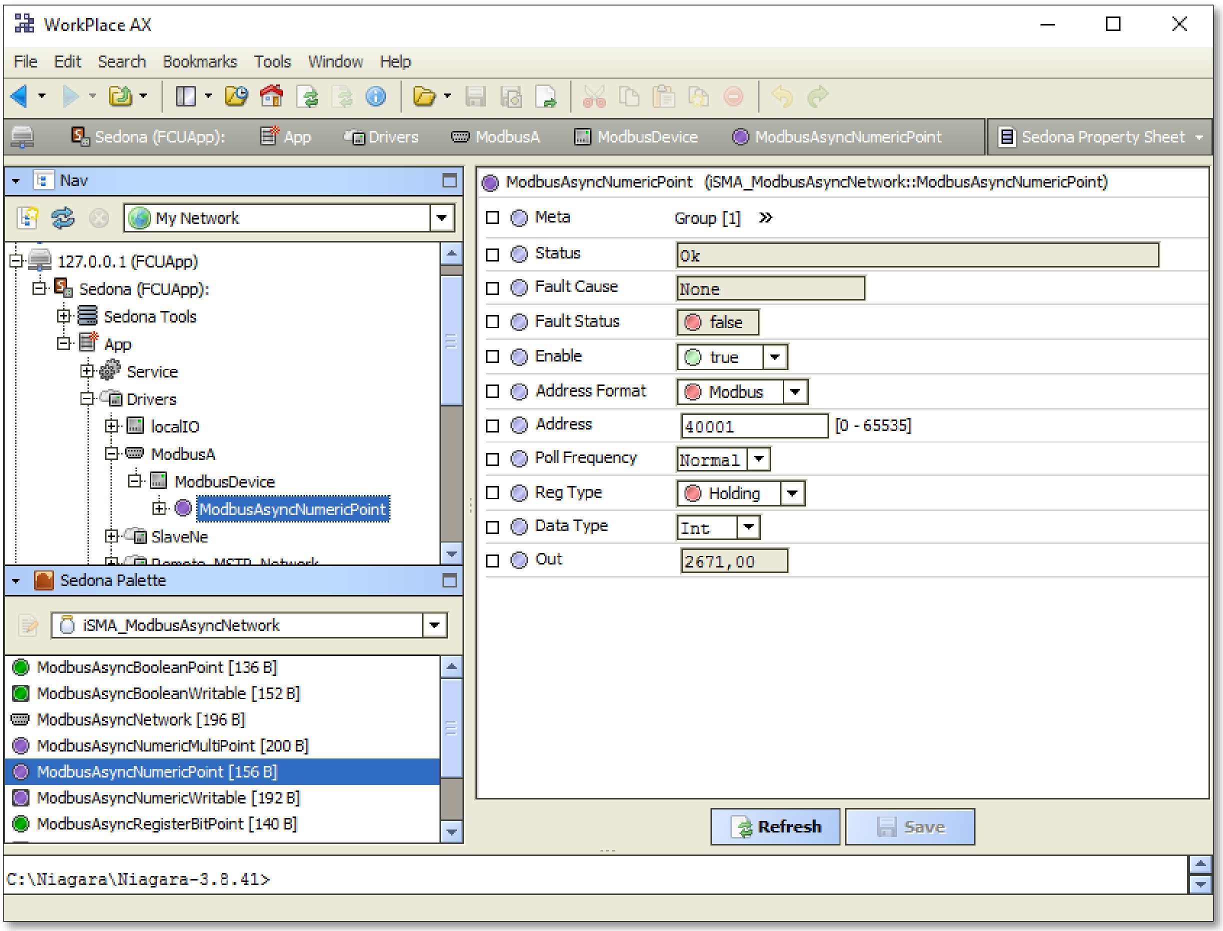


Figure 44 ModbusAsyncNumericPoint component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Address Format – Register address format, available options: Modbus, Decimal,
- Address – Register address,
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Reg Type – Type of reading register, available options: Input, Holding,
- Data Type – Reading registry data type, available options: Int – 16-bits, Long – 32- bits,
- Float – 32-bits float-point, SInt – 16-bits with sign, Slong – 32-bits with sign,
- Out – Current value of read registry.

The component offers the following action, available under the right mouse button:

- Read – Action enforces the reading of the point.

10.7 Modbus Async Numeric Writable

Modbus Async Numeric Writable is a component responsible for sending and reading numeric values from the device. Component has to be placed under Modbus Async Device component.

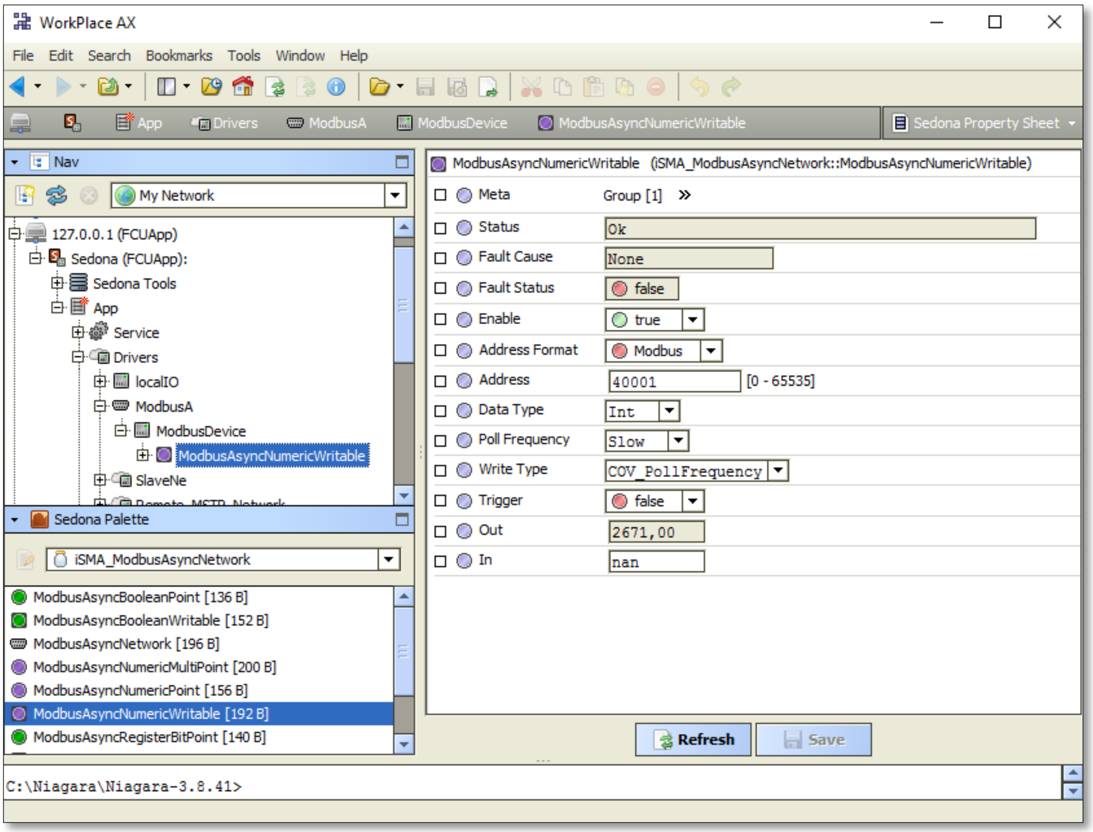


Figure 45 ModbusAsyncNumericWritable component

The component has the following slots:

- Status – Network status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read/write error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Address Format – Register address format, available options: Modbus, Decimal,
- Address – Register address,
- Data Type – Read/write registry data type, available options: Int – 16-bits, Long – 32-bits, Float – 32-bits float-point, SInt – 16-bits with sign, Slong – 32-bits with sign,
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Write Type – Writing mode, available options: COV – only on input change, COV_PollFequrency – on input change and periodically, PollFequrency – only periodically, COV_LinkSet - only on input change, using the "reverse following the link" function,
- Trigger – Trigger of remote enforcement of sending (on rising edge),
- Out – Output slot, current value of device registry,
- In – Input slot.

The component offers the following actions, available under the right mouse button:

- Set – Writes value to In slot and sends it to the device,
- Write – Sends value from In slot to the device,
- Read – Reads value from device and sends it to Out slot.
- Send Value – Sends user value from pop-up window, without changing In slot

10.8 Modbus Async Register Bit Point

Modbus Async Register Bit Point is a component responsible for reading Boolean values from device. The component has to be placed under Modbus Async Device component.

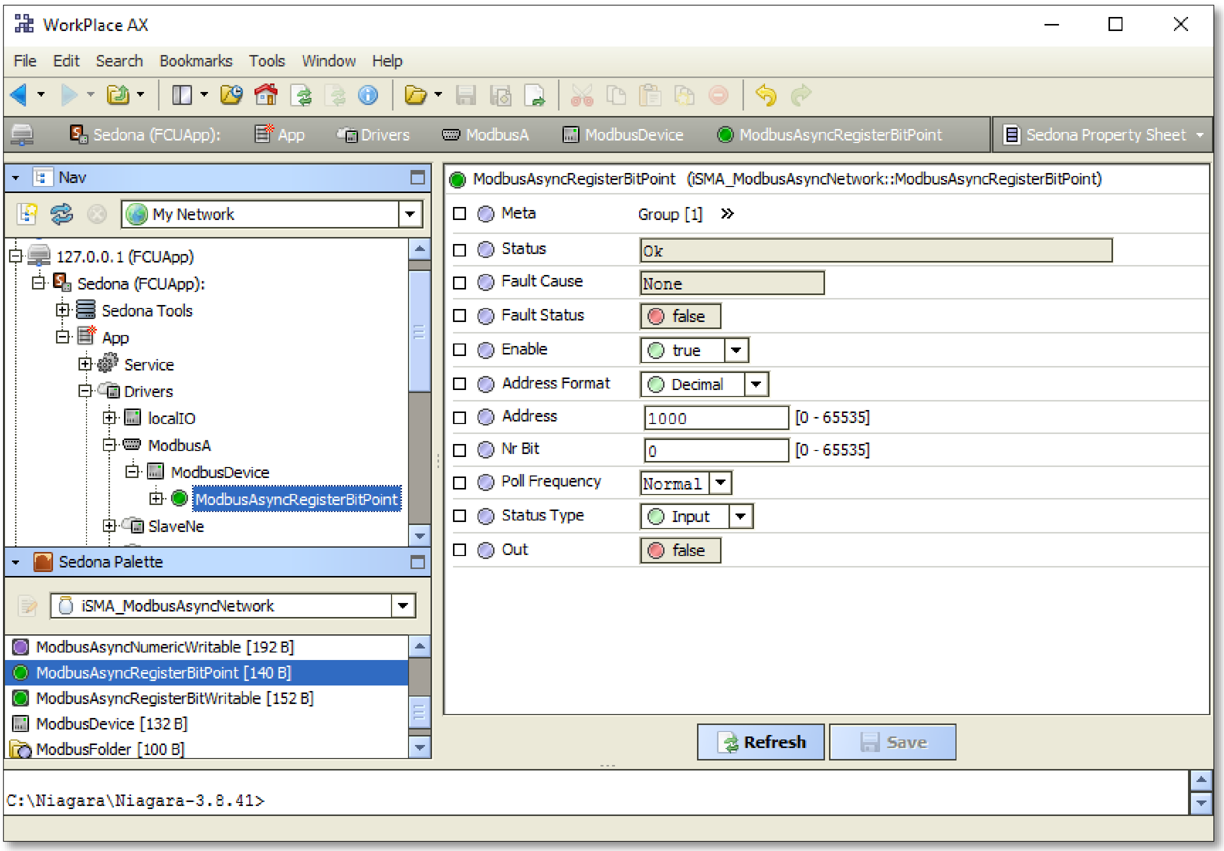


Figure 46 ModbusAsyncRegisterPoint component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Address Format – Register address format, available options: Modbus, Decimal,
- Address – Register address,
- Nr Bit – Bit number in register,
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Status Type – Type of reading register, available options: Input, Coil,
- Out – Current value of read bit.

The component offers the following action, available under the right mouse button:

- Read – Action enforces the reading of the point.

10.9 Modbus Async Register Bit Writable

Modbus Async Register Bir Writable is a component responsible for sending and reading Boolean values from the device. The component has to be placed under Modbus Async Device component.

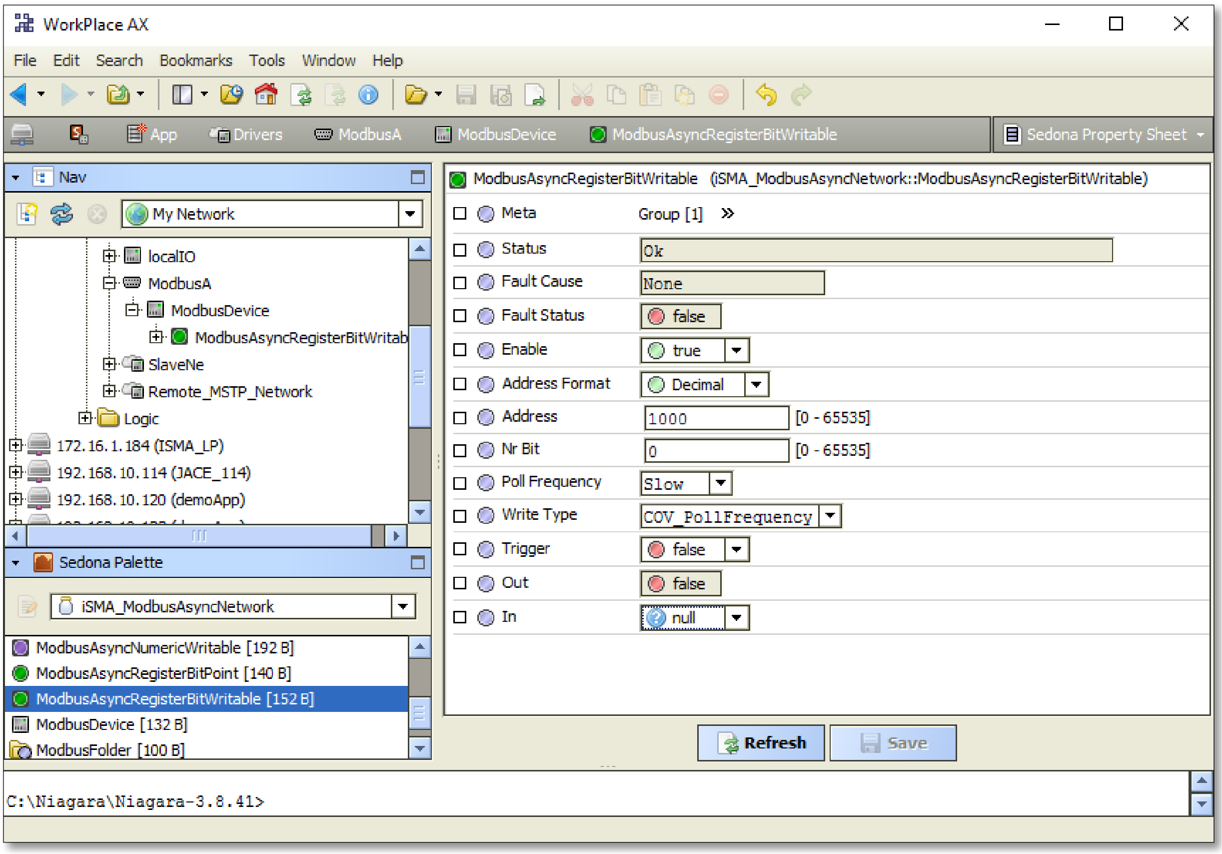


Figure 47 ModbusAsyncRegisterBitWritable component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status ("true" – point read error),
- Enable – Point enabled/disabled ("true"- Point enabled, "false"- Point disabled),
- Address Format – Register address format, available options: Modbus, Decimal,
- Address – Register address,
- Nr Bit – Bit number in register,
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Write Type – Writing mode, available options: COV – only on In slot change, COV_PollFequrency – on In slot change and periodically, PollFequrency – only periodically, COV_LinkSet – only on In slot change, using the "reverse following the link" function,
- Trigger – Trigger of remote enforcement of sending (on rising edge),
- Out – Current value of read bit,
- In – Input Slot.

The component offers the following actions, available under the right mouse button:

- Set True/Set False – Writes value to In slot and sends it to the device (not active when In slot has a link connected),

- Write – Sends value from In slot to the device,
- Read – Reads value from the device and sends it to Out slot,
- Send Value – Sends value to the device, without changing value on In slot.

10.10Modbus Folder

ModbusFolder is a component grouping and organizing Modbus Async point components.

11 iSMA Room Device Modbus kit

iSMA Room Device kit is an extension of Modbus Async Network kit which allows to easily manage iSMA-B-LP device. With kit's component, user can build application that easily communicates and configures LP panel. The Room Device Modbus kit contains following components:

- LP Temperature sensor
- LP CO2 Alarm
- LP CO2 Sensor
- LP Humidity sensor
- LP Main Menu Boolean
- LP Main Menu Numeric
- LP Submenu Boolean
- LP Submenu Numeric

11.1 Lp Temperature Sensor component

This component is responsible for Temperature sensor reading value and configuration.

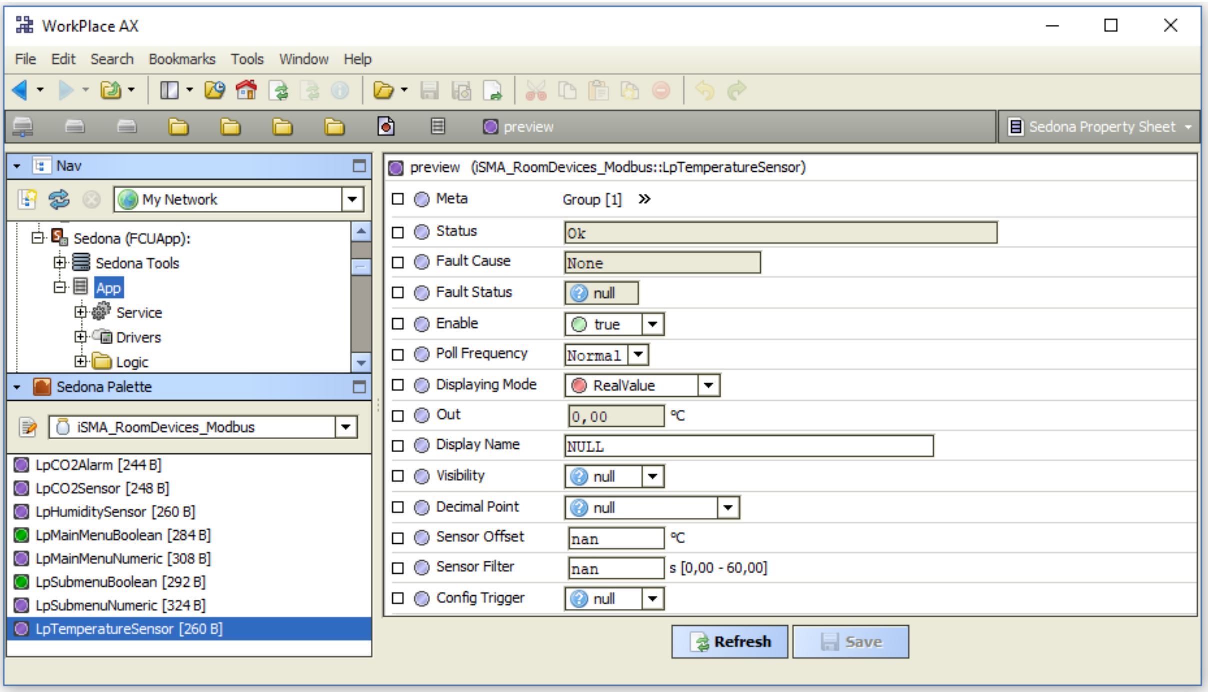


Figure 48 Temperature Sensor component view

The component has the following right-click menu actions:

- Read – Reads LP panel temperature sensor value and update Out slot,
- Read Config – Reads configuration parameters from LP panel, (Display Name, Visibility, Decimal Point, Sensor Offset, Sensor Filter)
- Write Config - Writes configuration parameters to LP panel (Display Name, Visibility, Decimal Point, Sensor Offset, Sensor Filter)

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,

- Fault Status – Point error status (“true” – point read/write error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Displaying Mode: RealValue – Out value is divided by 10 ; RegisterValue – value taken directly from register
- Out – Temperature sensor value output slot,
- Display Name – Temperature sensor display name on LCD display (up to 4 characters, only ASCII characters)
- Visibility – Active / Inactive sensor value on display,
- Decimal Point – Inactive or displays decimal point on first, second or third position.
- Sensor Offset – Temperature sensor value offset,
- Sensor Filter – Temperature sensor reading filter time in seconds,
- Config Triger – On rising edge sends to device component’s configuration parameters (Display Name, Visibility, Decimal Point, Sensor Offset, Sensor Filter)

11.2 Lp CO2 Sensor component

This component is responsible for CO2 sensor reading value and configuration.

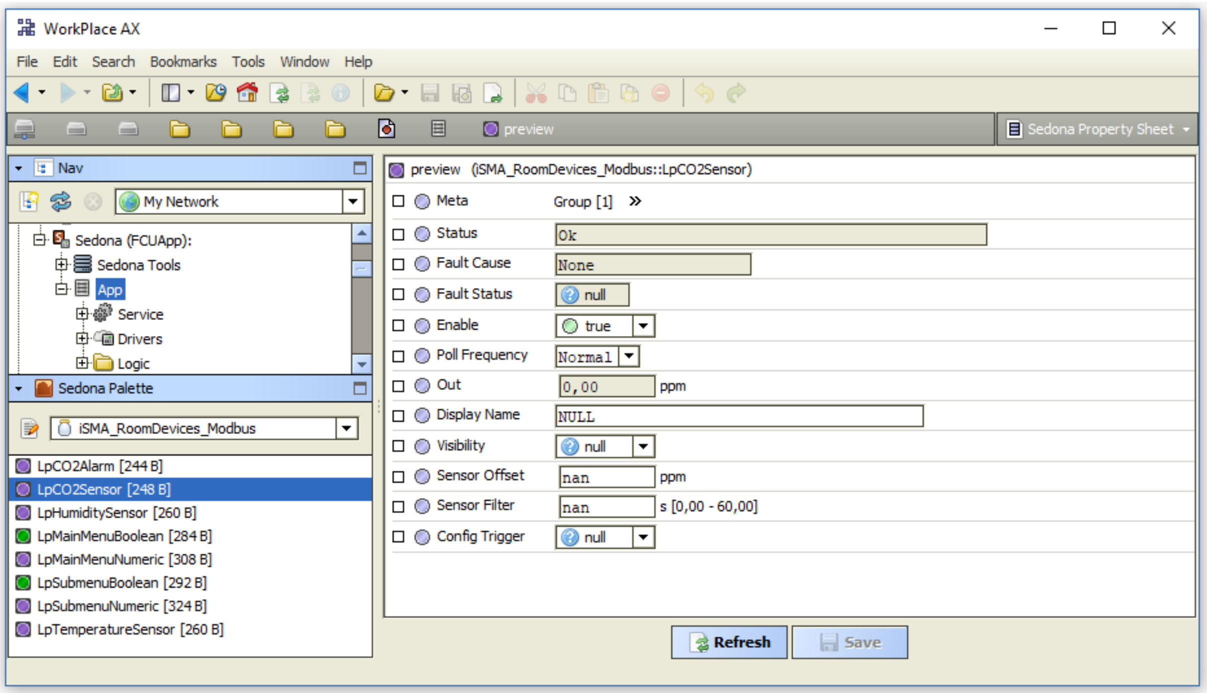


Figure 49 CO2 Sensor component view

The component has the following right-click menu actions:

- Read – Reads remote device CO2 sensor value and update Out slot,
- Read Config – Reads configuration parameters from LP panel (Display Name, Visibility, Sensor Offset, Sensor Filter),
- Write Config - Writes configuration parameters to LP panel (Display Name, Visibility, Sensor Offset, Sensor Filter).

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read/write error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Out – CO2 sensor value,
- Display Name – CO2 sensor name on LCD display (up to 4 characters, only ASCII characters)
- Visibility – Active / Inactive sensor value on display,
- Sensor Offset – CO2 sensor value offset,
- Sensor Filter – CO2 sensor reading filter time in seconds,
- Config Triger – On rising edge sends to device component’s configuration parameters (Display Name, Visibility, Sensor Offset, Sensor Filter).

11.3 Lp CO2 Alarm component

CO2 Alarm is a component dedicated to configuration of the high limit alarm function in LCD Panel.

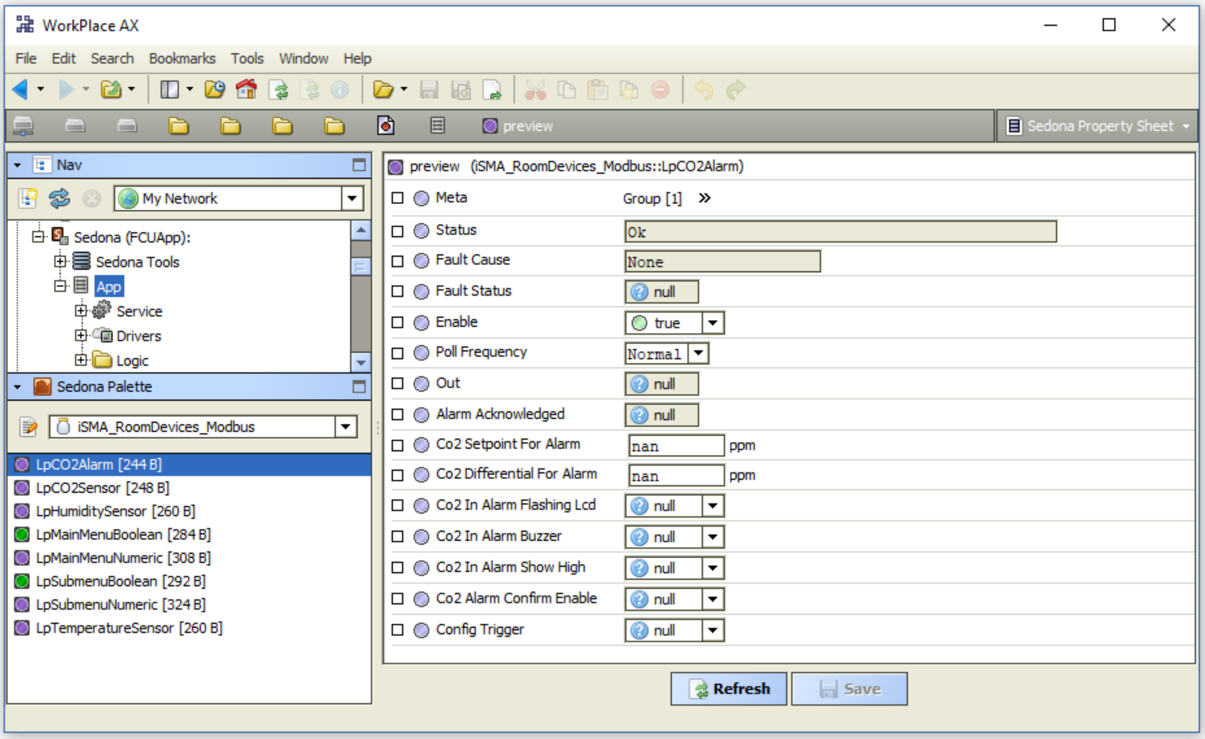


Figure 50 Alarm component view

The component has the following right-click menu actions:

- Read – Reads LP panel CO2 alarm status and updates Out slot,
- Read Config – Reads configuration parameters from LP panel, (Co2 Setpoint For Alarm, Co2 Differential For Alarm, Co2 Alarm Flashing Lcd, Co2 Alarm Buzzer, Co2 Alarm In

Alarm Show High, Co2 Alarm Confirm Enable)
- Write Config - Writes configuration parameters to LP panel (Co2 Setpoint For Alarm, Co2 Differential For Alarm, Co2 Alarm Flashing Lcd, Co2 Alarm Buzzer, Co2 Alarm In Alarm Show High, Co2 Alarm Confirm Enable).

The component has the following slots:
- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read/write error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Out – CO2 alarm status,
- Co2 Setpoint For Alarm – Co2 alarm setpoint value in ppm
- Co2 Differential For Alarm – Co2 alarm setpoint differential value in ppm
- Co2 Alarm Flashing Lcd – Active / Inactive LCD Background Illumination flashing
- Co2 Alarm Buzzer – Active / Inactive sound alarm
- Co2 Alarm In Alarm Show High – Active / Inactive “High” label display
- Co2 Alarm Confirm Enable – Active / Inactive alarm acknowledgement by any button
- Config Triger – On rising edge sends configuration parameters to LP panel

11.4 Lp Humidity Sensor component

This component is responsible for humidity sensor configuration and reading value.

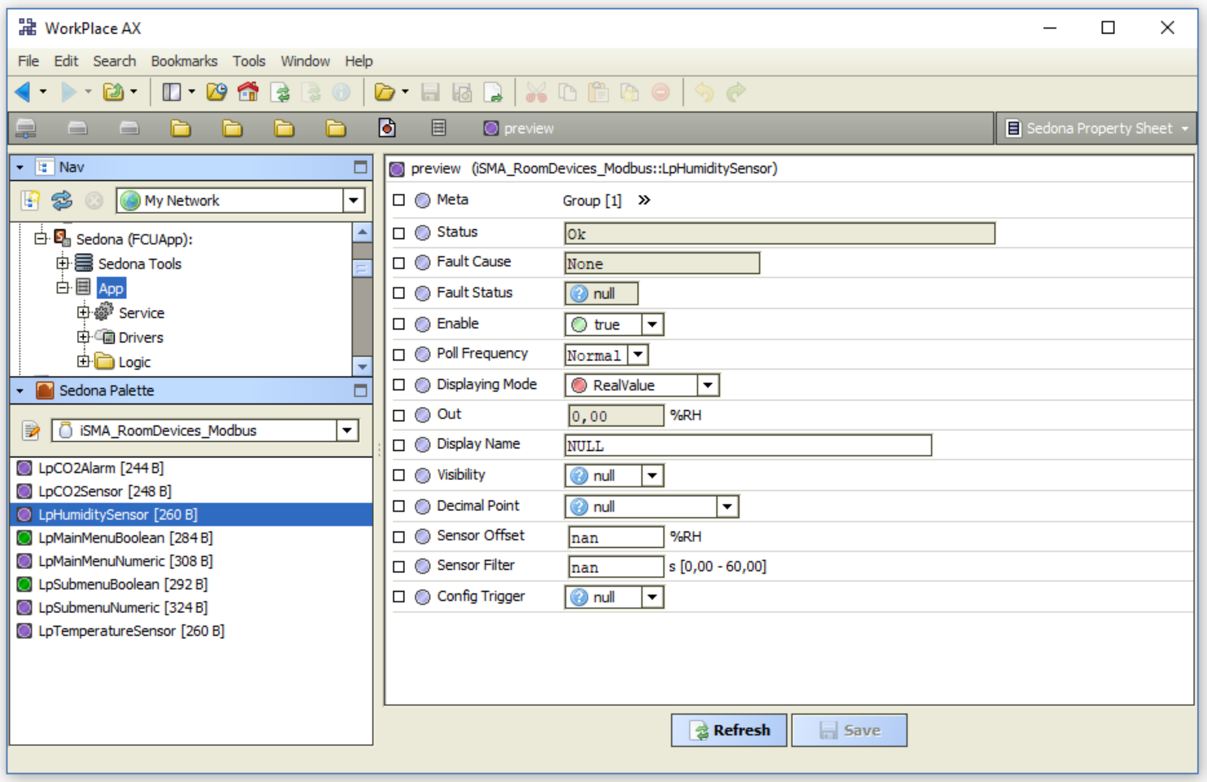


Figure 51 Humidity sensor component view

The component has the following right-click menu actions:

- Read – Reads remote device humidity sensor value and updates Out slot,
- Read Config – Reads configuration parameters from LP panel (Display Name, Visibility, Decimal Point, Sensor Offset, Sensor Filter),
- Write Config - Writes configuration parameters to LP panel (Display Name, Visibility, Decimal Point, Sensor Offset, Sensor Filter).

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status ("true" – point read/write error),
- Enable – Point enabled/disabled ("true"- Point enabled, "false"- Point disabled),
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Displaying Mode: RealValue – Out value is divided by 10 ; RegisterValue – value taken directly from register
- Out – Humidity sensor value,
- Display Name – Humidity sensor display name on LCD display (up to 4 characters, only ASCII characters)
- Visibility – Active / Inactive humidity sensor value on display,
- Decimal Point – Inactive or displays decimal point on first, second or third position.
- Sensor Offset – Humidity sensor value offset,
- Sensor Filter – Humidity sensor reading filter time in seconds,

- Config Triger – On rising edge sends to device component's configuration parameters

11.5 Lp Main Menu Boolean component

This component is responsible for read/write and configuration of single boolean parameter which is placed in LP panel main menu.

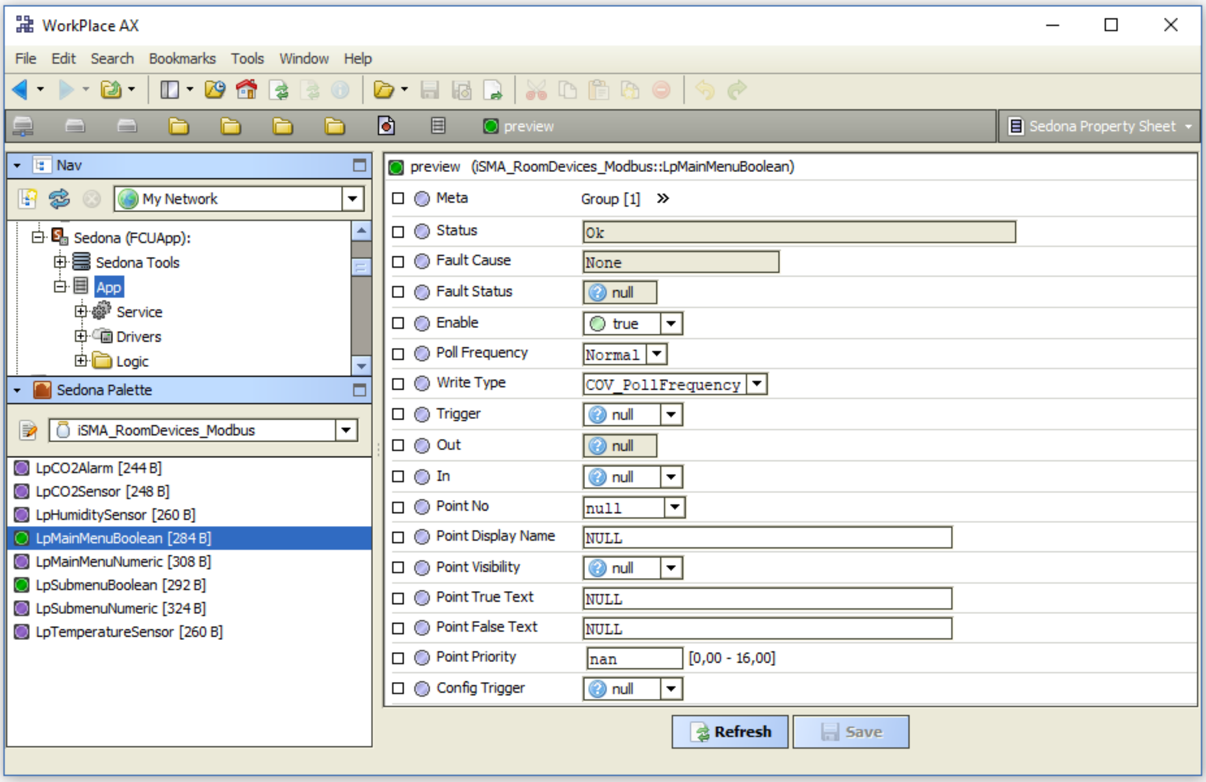


Figure 52 Menu Boolean component view

The component has the following right-click menu actions:

- Set True – Sets In on True state and sends it to main menu point,
- Set False – Sets In on False state and sends it to main menu point,
- Write - Sends In slot state to main menu point,
- Read – Reads LP panel main menu point value and sets Out slot,
- Send Value – Sends point user value to main menu, without changing input slot, from popup window,
- Read Config – Reads main menu point configuration parameters from LP panel (Point Display Name, Point Visibility, Point True Text, Point False Text, Point Priority),
- Write Config – Writes main menu point configuration parameters to LP panel (Point Display Name, Point Visibility, Point True Text, Point False Text, Point Priority).

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,

- Fault Status – Point error status (“true” – point read/write error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Write Type – Writing mode, available options: COV – only on In slot change, COV_PollFequrency – on In slot change and periodically, PollFequrency – only periodically, COV_LinkSet – only on In slot change, using the "reverse following the link" function
- Trigger - Trigger of remote enforcement of sending (on rising edge),
- Out – Main Menu point Output slot, current value,
- In – Main Menu point Input slot,
- Point No – Panel Main Menu point number,
- Point Display Name – Point display name on LCD screen (up to 4, only ASCII characters)
- Point Visibility – Active / Inactive point on display,
- Point True Text – 4 characters LCD display text in true state (only ASCII characters),
- Point False Text - 4 characters LCD display text in false state (only ASCII characters),
- Point Priority – Displaying priority on LCD screen (starting from the lowest value),
- Config Triger – On rising edge sends to device component’s configuration parameters

11.6 Lp Main Menu Numeric component

This component is responsible for read/write and configuration of a single numeric parameter which is placed in LP panel main menu.

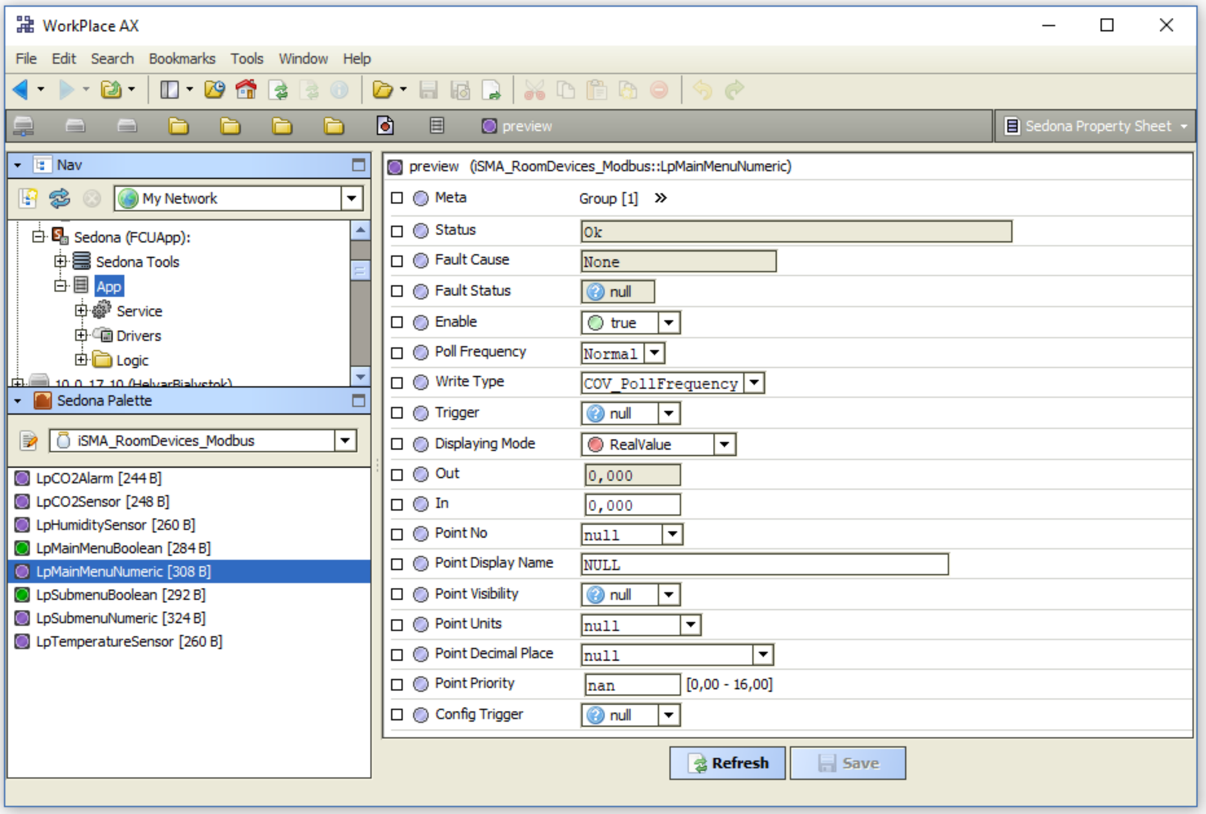


Figure 53 Main menu numeric component view

The component has the following right-click menu actions:

- Set – Sets In slot and sends it to main menu point,
- Write - Sends In slot and sends it to main menu point,
- Read – Reads LP panel main menu point value and sets Out slot,
- Send Value – Sends to main menu point user value, without changing input slot, from popup window,
- Read Config – Reads main menu point configuration parameters from LP panel (Point Display Name, Point Visibility, Point Decimal Place, Point Priority),
- Write Config – Writes main menu point configuration parameters to LP panel (Point Display Name, Point Visibility, Point Decimal Place, Point Priority).

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read/write error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Write Type – Writing mode, available options: COV – only on Input change, COV_PollFequency – on In slot change and periodically, PollFequency – only periodically, COV_LinkSet – on In slot change, using the "reverse following the link" function

- Trigger - Trigger of remote enforcement of sending (on rising edge),
- Displaying Mode: RealValue – Out value is divided by 10; RegisterValue – value taken directly from register
- Out – Point output slot, current value,
- In – Point input slot,
- Point No – Panel Main Menu number,
- Point Display Name – Point display name on LCD screen (up to 4, only ASCII characters)
- Point Visibility – Active / Inactive showing point on display,
- Point Decimal Place – Inactive or displays decimal point on first, second or third position.
- Point Priority – Displaying priority on LCD screen (starting from the lowest value),
- Config Triger – On rising edge sends to device component's configuration parameters

11.7 Lp Submenu Boolean component

This component is responsible for read/write and configuration of a single (one of 8 points) boolean parameter which is placed in LP panel submenus. There are 6 submenus in LP panel: Temperature, Fan, Light, Blind, Alarm, and Occupancy and each submenu can have up to 8 boolean points.

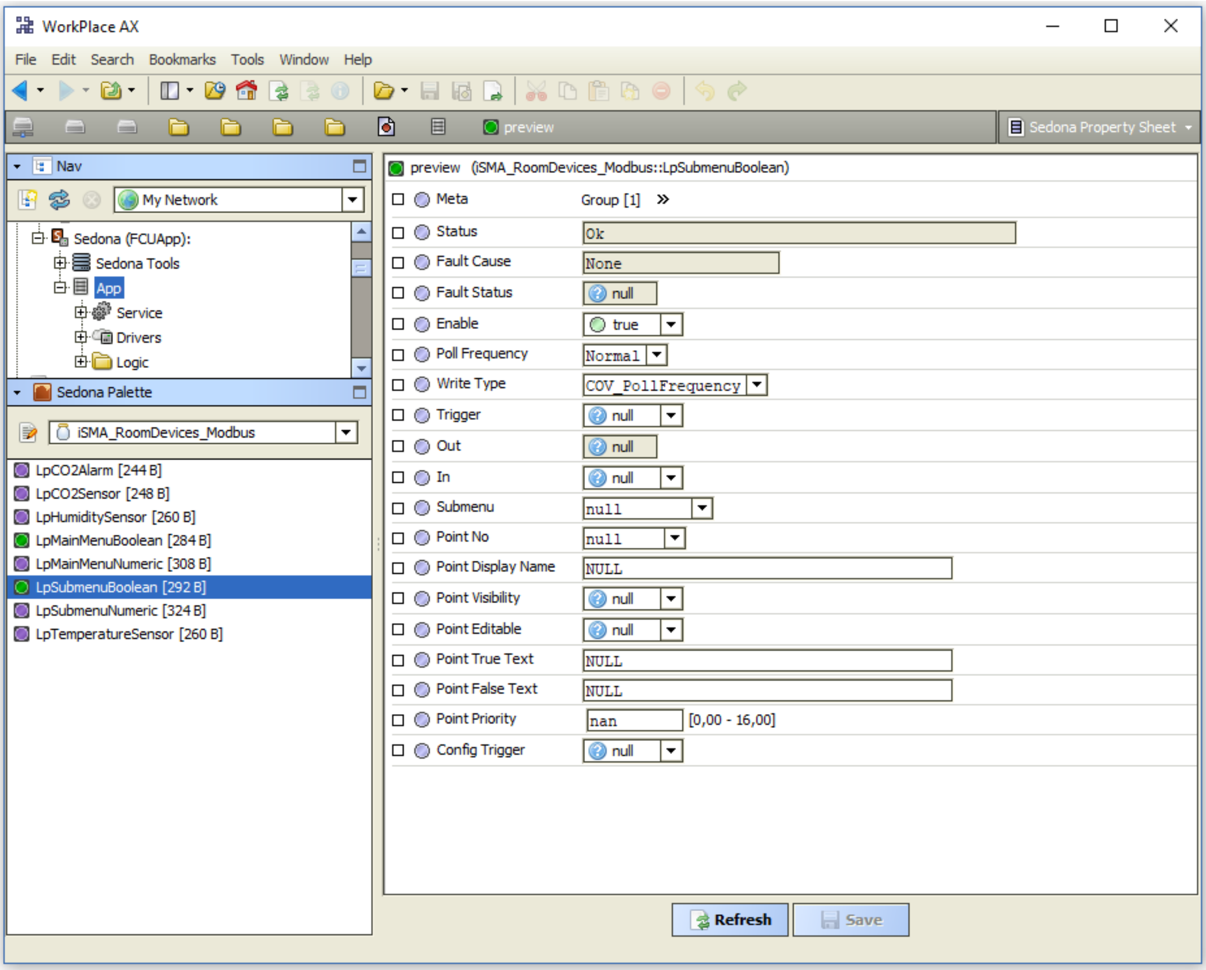


Figure 54 Submenu Boolean component view

The component has the following right-click menu actions:

- Set True – Sets In on True state and sends it to submenu point,
- Set False – Sets In on False state and sends it to submenu point,
- Write - Sends In state to submenu point,
- Read – Reads LP panel submenu point value and sets Out slot,
- Send Value – Sends user value to submenu point, without changing input slot, from popup window,
- Read Config – Reads submenu point configuration parameters from LP panel (Point Display Name, Point Visibility, Point True Text, Point False Text, Point Priority),
- Write Config – Writes submenu point configuration parameters to LP panel (Point Display Name, Point Visibility, Point True Text, Point False Text, Point Priority).

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read/write error),
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Write Type – Writing mode, available options: COV – only when In slot change, COV_PollFequrency – on In slot change and periodically, PollFequrency – only periodically, COV_LinkSet – only In slot change, using the "reverse following the link" function
- Trigger - Trigger of remote enforcement of sending (on rising edge),
- Out – Output slot, current value of read/write registry,
- In – Register input slot,
- Submenu - Panel submenu number,
- Point No – Submenu point number,
- Point Display Name – 4 characters submenu point LCD display name (only ASCII characters)
- Point Visibility – Active / Inactive point on display,
- Point Editable – Enable/disable point edit,
- Point True Text – 4 characters LCD display text in true state (only ASCII characters),
- Point False Text - 4 characters LCD display text in false state (only ASCII characters),
- Point Priority – Displaying priority on LCD screen (starting from the lowest value),
- Config Triger – On rising edge sends to device component’s configuration parameters

11.8 Lp Submenu Numeric component

This component is responsible for read/write and configuration of a single (one of 8 point) numeric user parameter which is placed in one of LP panel submenus. There are 6 submenus in LP panel: Temperature, Fan, Light, Blind, Alarm, and Occupancy, and each submenu can have up to 8 numeric points.

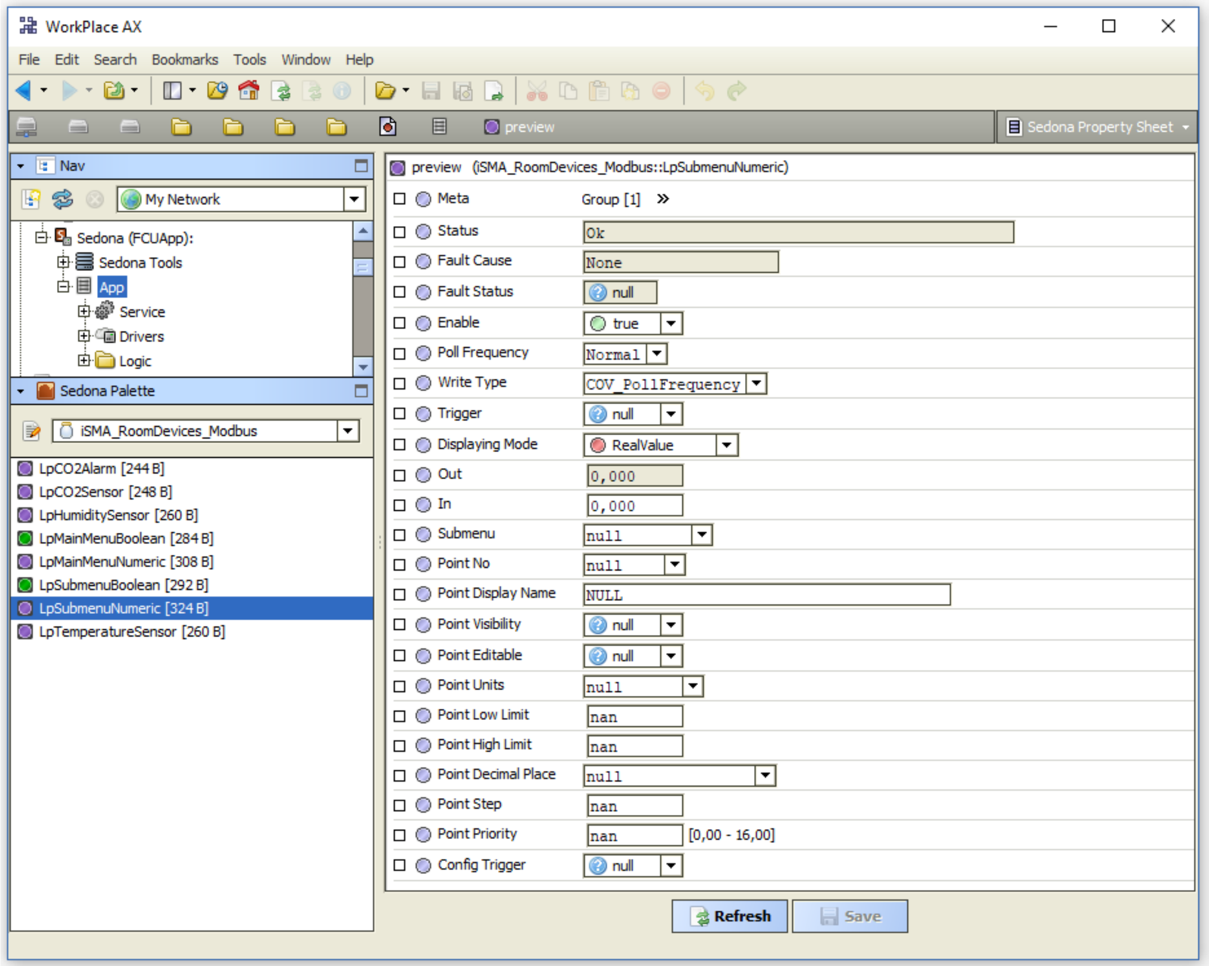


Figure 55 Submenu numeric component view

The component has the following right-click menu actions:

- Set – Sets In slot and sends it to submenu point,
- Write - Sends In slot and sends it to submenu point,
- Read – Reads LP panel submenu point value and sets Out slot,
- Send Value – Sends user value to submenu point, without changing input slot, from popup window,
- Read Config – Reads submenu point configuration parameters from LP panel (Point Display Name, Point Visibility, Point Editable, Point Units, Point Low Limit, Point High Limit, Point Decimal Place, Point Step, Point Priority),
- Write Config – Writes submenu point configuration parameters to LP panel (Point Display Name, Point Visibility, Point Editable, Point Units, Point Low Limit, Point High Limit, Point Decimal Place, Point Step, Point Priority).
-

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Fault Status – Point error status (“true” – point read/write error),
- Point Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),

- Poll Frequency – Reading poll frequency, available options: Fast, Normal, Slow,
- Write Type – Writing mode, available options: COV – only on In slot change, COV_PollFequrency – on In slot change and periodically, PollFequrency – only periodically, COV_LinkSet – only on In slot change, using the "reverse following the link" function
- Trigger - Trigger of remote enforcement of sending (on rising edge),
- Displaying Mode: RealValue – Out value is divided by 10; RegisterValue – value taken directly from register
- Out – Output slot, current value of read/write register,
- In – Register input slot,
- Submenu - Panel submenu number,
- Point No – Submenu point number,
- Point Display Name – 4 characters submenu point LCD display name (only ASCII characters)
- Point Visibility – Active / Inactive point on display,
- Point Editable – Enable/disable point edit,
- Point Units – Inactive or display value unit, available units options: °C, °F, Pa, Lx, ppm, m3/h, %RH, L/s, %, h.
- Point Low Limit – Submenu parameter low limit value,
- Point High Limit - Submenu parameter high limit value,
- Point Decimal Place – Inactive or displays decimal point on first, second or third position,
- Point Step – Out value changing step,
- Point Priority – Displaying priority on LCD screen (starting from the lowest value),
- Config Triger – On rising edge sends to device component’s configuration parameters

12 Advance Control kit

In order to facilitate the building of user applications the kit Advance Control was developed. Using this kit components, user can build an advance application in a simple way. This kit contains following components:

- Riser Lower
- Dimmer Switch
- Action Trigger

12.1 Riser Lower component

This component was created to simplify control of 3-point valve actuators. This component has following functions:

- Analog input, work with PID regulators
- 2 Digital Outputs for 3-point direct control valve actuators
- Analog Output for 3-point control valve actuators by voltage level (additional device required)
- Midnight reset function to automatically adjust physical and virtual valve position
-

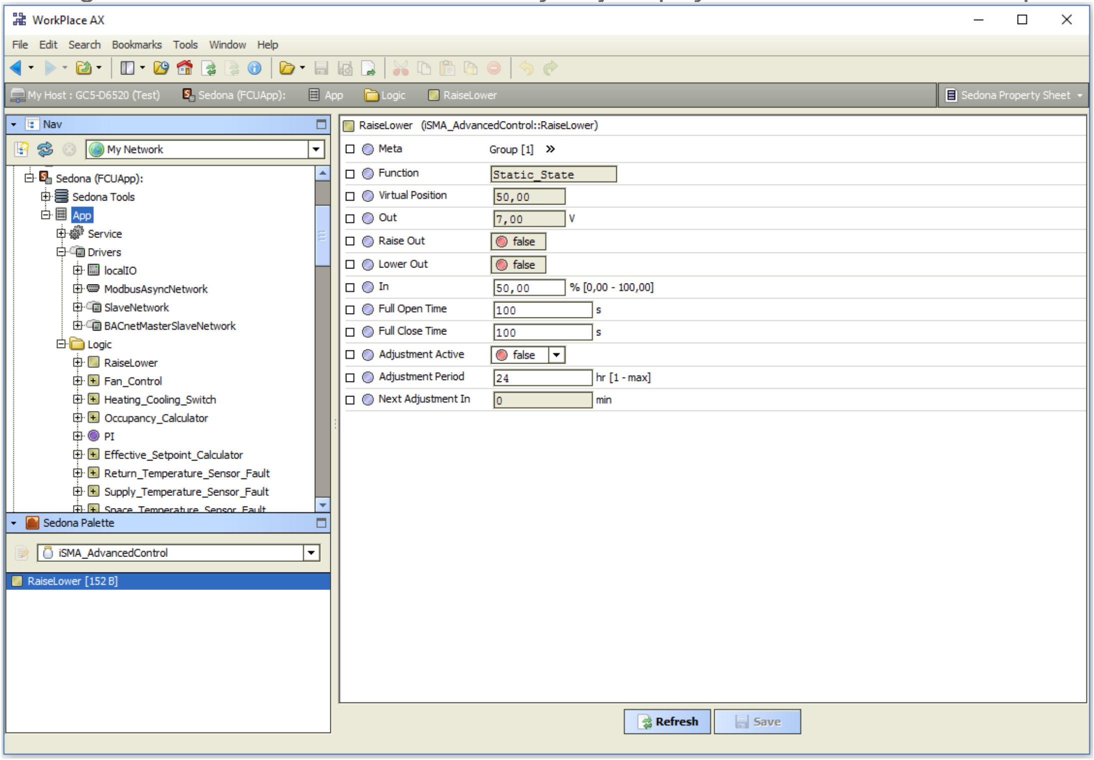


Figure 56 RiserLower component view

The component has following actions:

- Adjust – Valve adjustment procedure recall (adjusts physical and virtual valve position)

The component has following slots:

Function – Current function description (Lower_State, Static_State, Raise_State, Adjustment_Opening, Adjustment_Closing)

- Virtual Position – Valve virtual position in %
- Out – Analog output out

- Rise Out – Digital output out for Rising function
- Rise Low – Digital output out for Lowing function
- In – Valve position demand
- Full Open Time – Time for valve position open from 0% to 100%
- Full Close Time – Time for valve position close from 100% to 0%
- Adjustment Active – Remote adjustment procedure trigger
- Adjustment Period – Adjustment procedure recall period in hours
- Next Adjustment In – Time in minutes for next adjustment procedure

Out Value	Rise	Lower	Description
0	Off	Off	Off
4	Off	On	Lower
7	Off	Off	Static
10	On	Off	Rise

Table 3 Analog output voltage level function

12.2 Dimmer Switch

This component was created to control light dimmer, with the use of single button (one digital input) or two buttons (two digital input).

In Single Switch mode Switch 1 has defined function for short and long press. The short press is when button is pressed for less than the time defined in slot Short In. The long press is when button is pressed for longer than the time defined in slot Short In. The short press is dedicated for On/Off switching, the long press is dedicated for changing dimming value. Each short press toggle between on and off state. During long press the component increases or decreases dimming value.

In Double Switch mode each button has defined functions: Switch 1 is for switching on (short press) and increasing dimmer value (long press), Switch 2 is for switching off (short press) and decreasing dimmer value (long press). The short press is when high state time is less than time defined in Short In slot. The long press is when short time elapse, and button is still in high state.

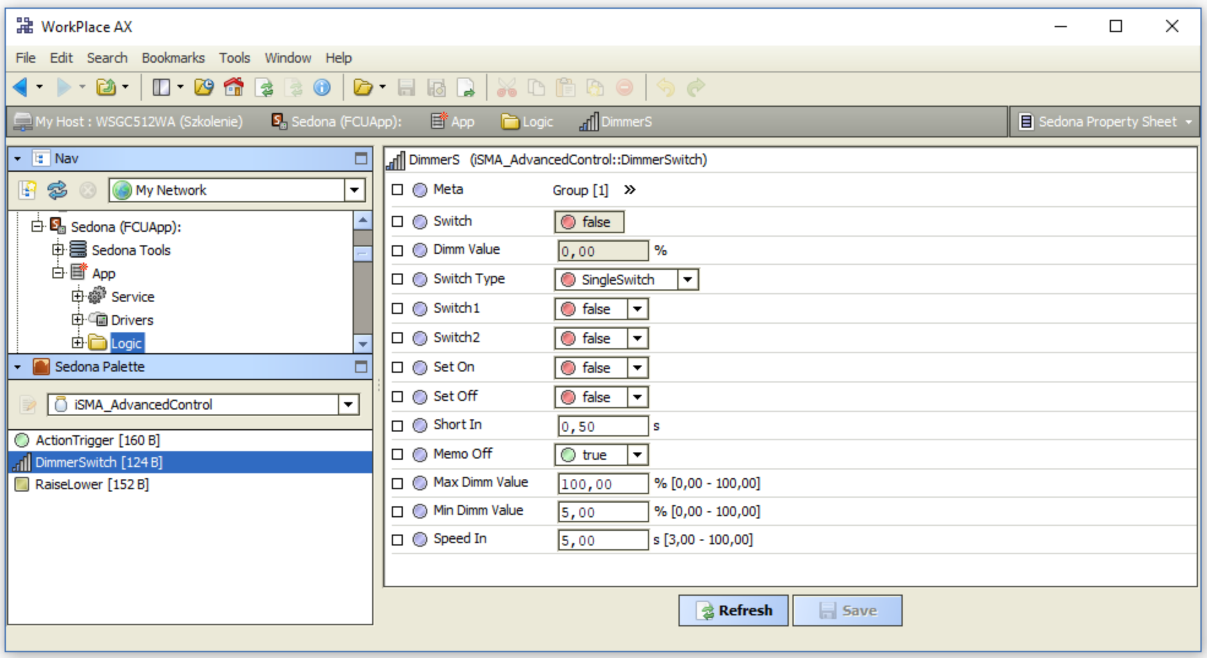


Figure 57 Dimmer Single Switch property sheet view

The component has following slots:

- Switch – Out slot for dimmer, digital value for On and Off state
- Dimm Value – Out slot for dimmer, analog 0-100% value
- Switch Type – Button input
- Switch1 – Button input, main for Single Switch mode, On or increasing output in Double Switch mode
- Switch2 – Button input, not active for Single Switch mode, Off or decreasing output in Double Switch mode
- Set On – Trigger for switch dimmer to On state (to max level)
- Set Off – Trigger for switch dimmer to Off state
- Short In – Time for short button press
- Memo Off – Enable / Disable memory function of Dimm Value during switch off
- Max Dimm Value – Max. dimmer analog value,
- Min Dimm Value – Min. dimmer analog,
- Speed In – Dimming speed time.

12.3 Action Trigger

This component was created to remotely recall action from Sedona component. The Sedona does not allow for making links to component's actions and actions can be recalled manually from programming software (for example WorkPlce) or by a dedicated component. The Action Trigger component has 3 inputs slots each dedicated for Sedona variable type (in standard we use only one of them, corresponding to component type). The programmes make link from component, from which the action will be recalled, to Action Trigger component input slot. The slot Action Name defines which action is to be recalled.

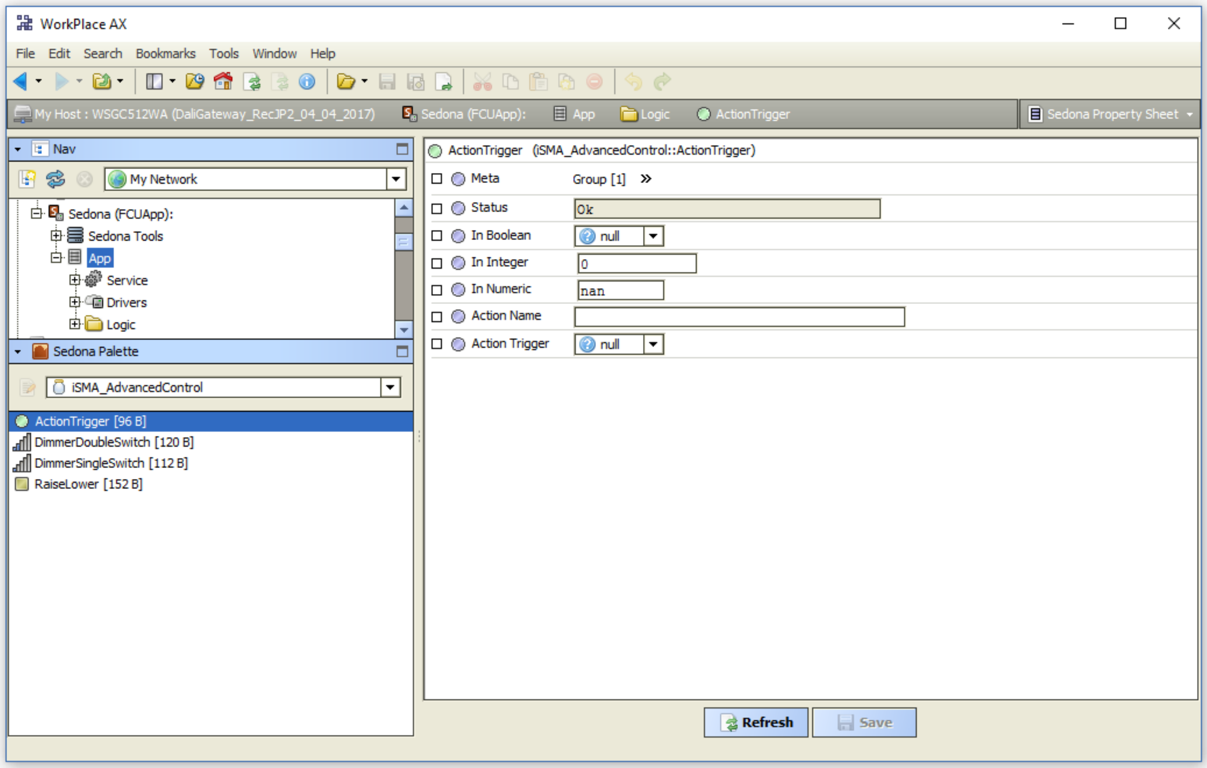


Figure 58 Action Trigger property sheet view

The component has following actions:

- Action – Manually recalls action from linked component

The component has following slots:

- Status – Component status,
- In Boolean – Input slot to make link connection between components – Boolean type,
- In Integer – Input slot to make link connection between components – Integer type,
- In Numeric – Input slot to make link connection between components – Numeric/Float type,
- Action name – Action name from linked component which will be recalled,
- Action Trigger – Recall action from component linked to one of input slot, defined in Action Name.

13 BACnet Master Slave kit

iSMA-B-FCU device can work in defined groups where one device is Master and remaining devices (slaves) follow Master parameters. Single Master device can have up to 5 Slave devices and it is possible to share up to 100 points with them.

Note: This function is available only in BACnet MSTP protocol, using RS485 port (COM1).

13.1 BacnetMasterSlaveNetwork

BACnet Master Slave Network is a main component, responsible for communication between Master device and Slave devices by BACnet MSTP protocol, using RS485 port (COM1). BACnet Master Slave Network sets parameters such as Slave devices Id, communication parameters (such as Poll Frequancy or Max Write Time) and reads status of Slave devices. The component has to be placed under Drivers folder.

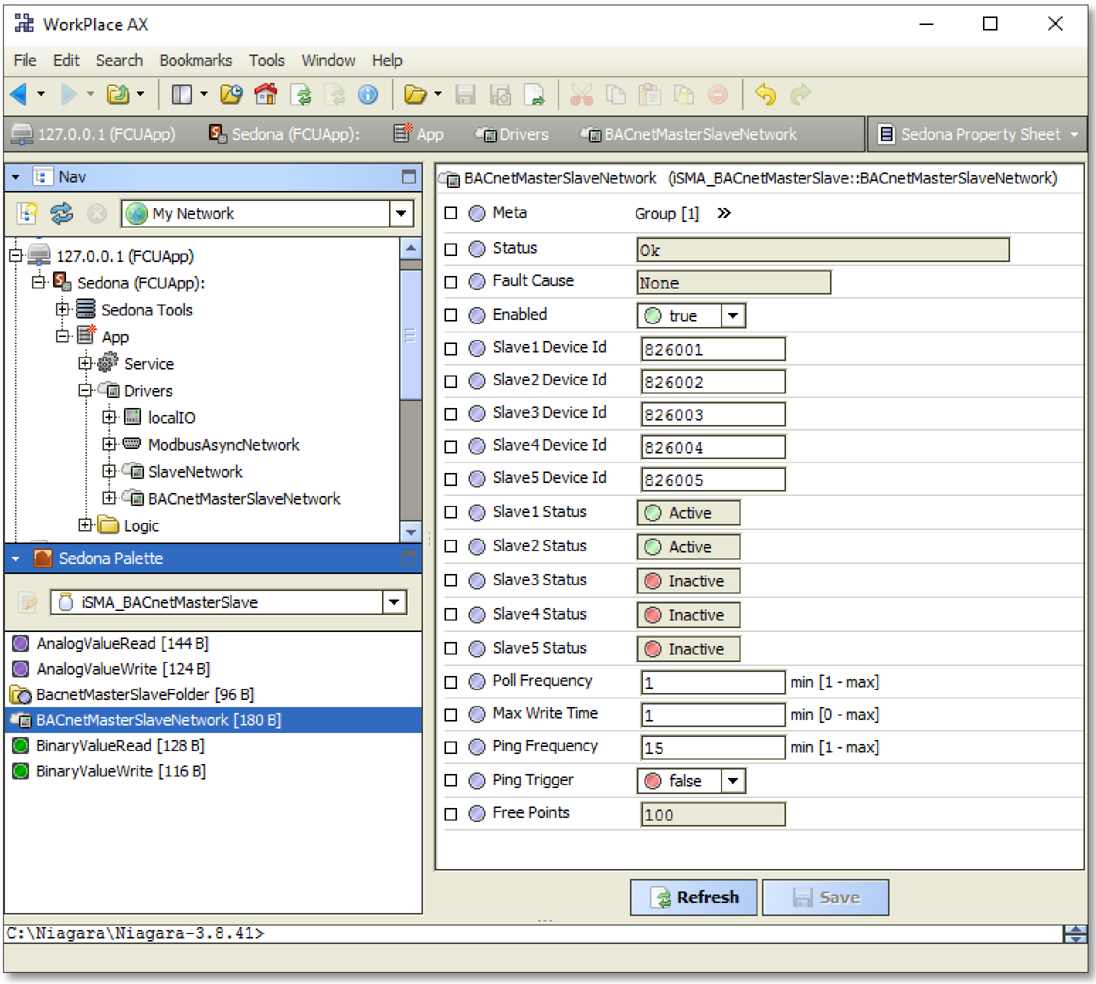


Figure 59 BACnetMasterSlaveNetwork component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Enable – Network enabled/disabled ("true"- Point enabled, "false"- Point disabled),
- Slave1DeviceId – Slave5DeviceId – BACnet Id of slave devices,
- Slave Status1 – Slave Status5 – Statuses of slave devices,
- Poll Frequency – poll frequency of all read-only points, min. 1,
- Max Write Time – Max time between sending values of all writeable points: if a value equals 0, values of writable points will be sent only "on value change",
- Ping Frequency – Time between testing message to check Slave device connection, min. 1,
- Ping Trigger – On rising edge of value, enforces Ping action,
- Free Points – Number of free BACnet Master Slave points.

The component offers the following action, available under the right mouse button:

- Ping– Sends test message to Slave devices to check devices' statuses.

13.2 AnalogValueRead

Analog Value Read is a component responsible for reading analog values from slave devices. Values are read with time period defined in Poll Frequency slot. Reading can be also forced using Read action. The component has to be placed under BACnet Master Slave component.

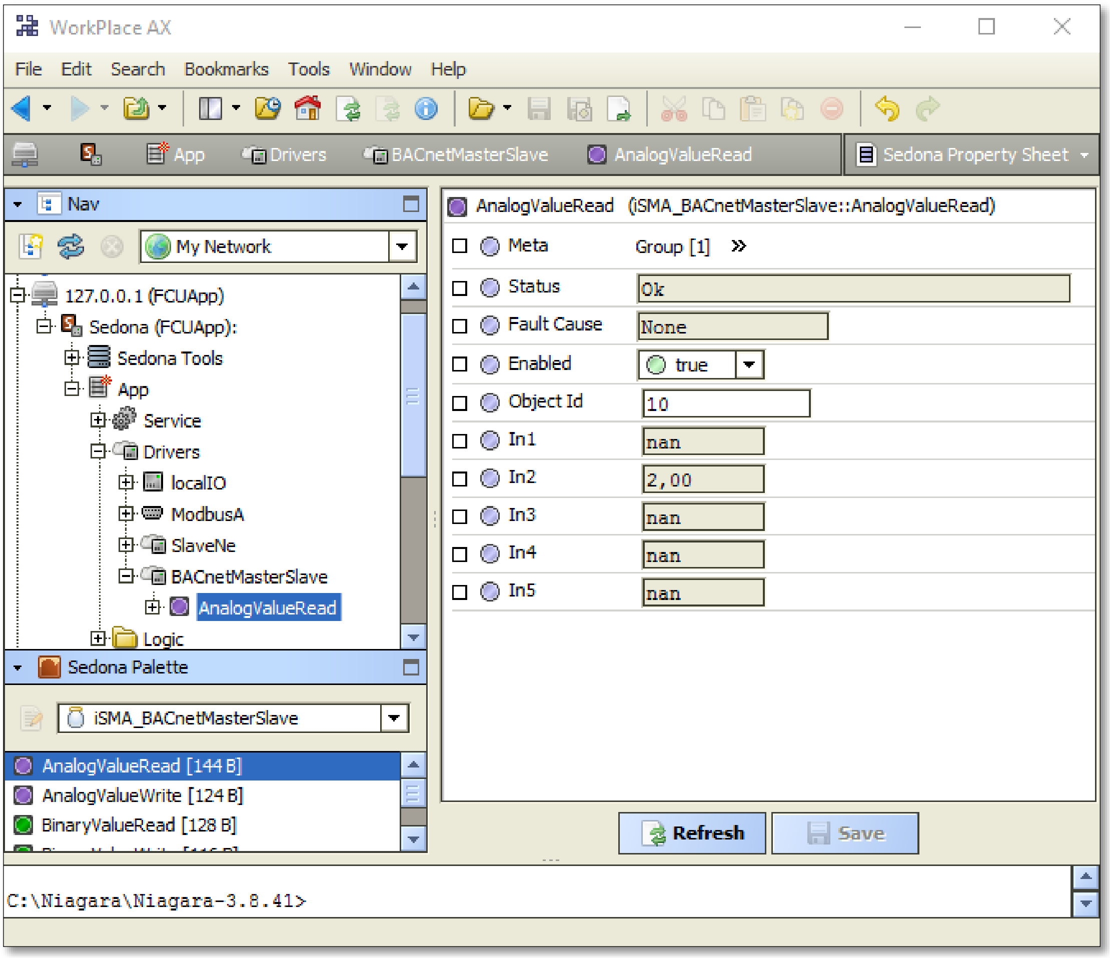


Figure 60 AnalogValueRead component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Enable – Point enabled/disabled ("true"- Point enabled, "false"- Point disabled),
- Object Id – BACnet Id of point,
- In1 – In5 – Slots storage values of points from corresponding slave devices.

Note: If communication with some Slave devices is be broken, corresponding slot stores last value, which has been read and slot Status of component displays "Some Points Down".

The component offers the following action, available under the right mouse button:

- Read – Action enforces the reading of the point.

13.3 AnalogValueWrite

Analog Value Write is a component responsible for sending analog value to slave devices. Value is written in time periods defined by Max Write Time (only when value of Max Write Time is higher than 0) and if value of In slot has changed. Writing can be also enforced by Set or Write action. The component has to be placed under BACnet Master Slave component.

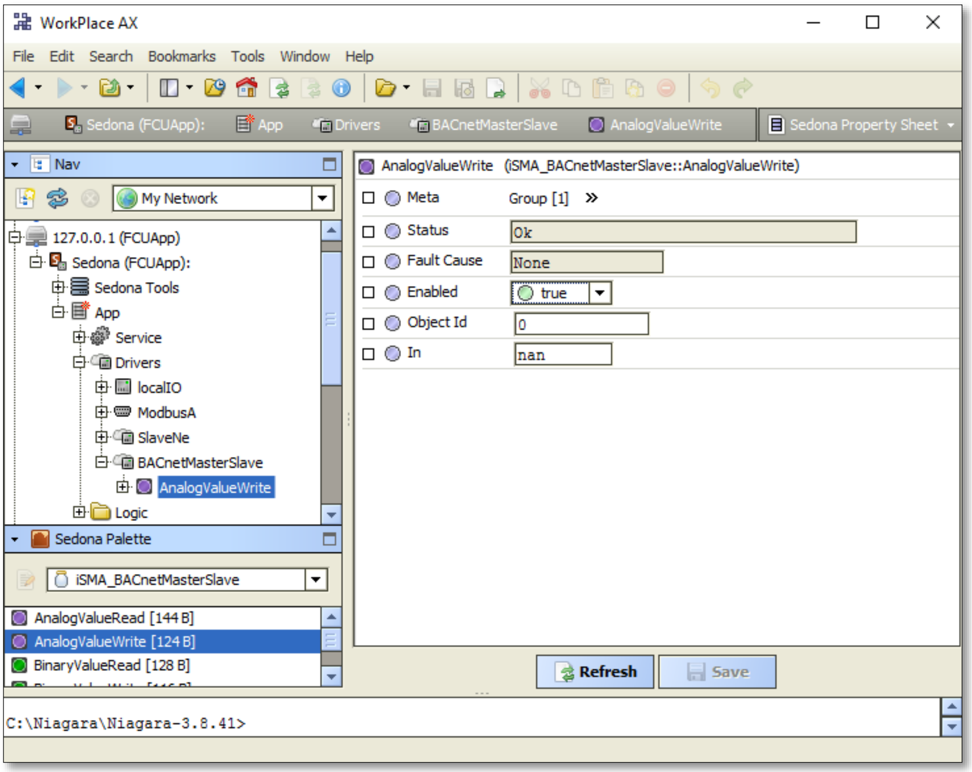


Figure 61 AnalogValueWrite component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Enable – Point enabled/disabled (“true”- Point enabled, “false”- Point disabled),
- Object Id – BACnet Id of point,
- In – Input slot; this value is sent to slave devices.

The component offers the following actions, available under the right mouse button:

- Set – Writes value to In slot and sends it to slave devices,
- Write – Sends value from In slot to slave devices.

13.4 BinaryValueRead

Binary Value Read is a component responsible for reading binary values from slave devices. Values are read in time periods defined in Poll Frequency slot. Reading can be also enforced using Read action. The component has to be placed under BACnet Master Slave component.

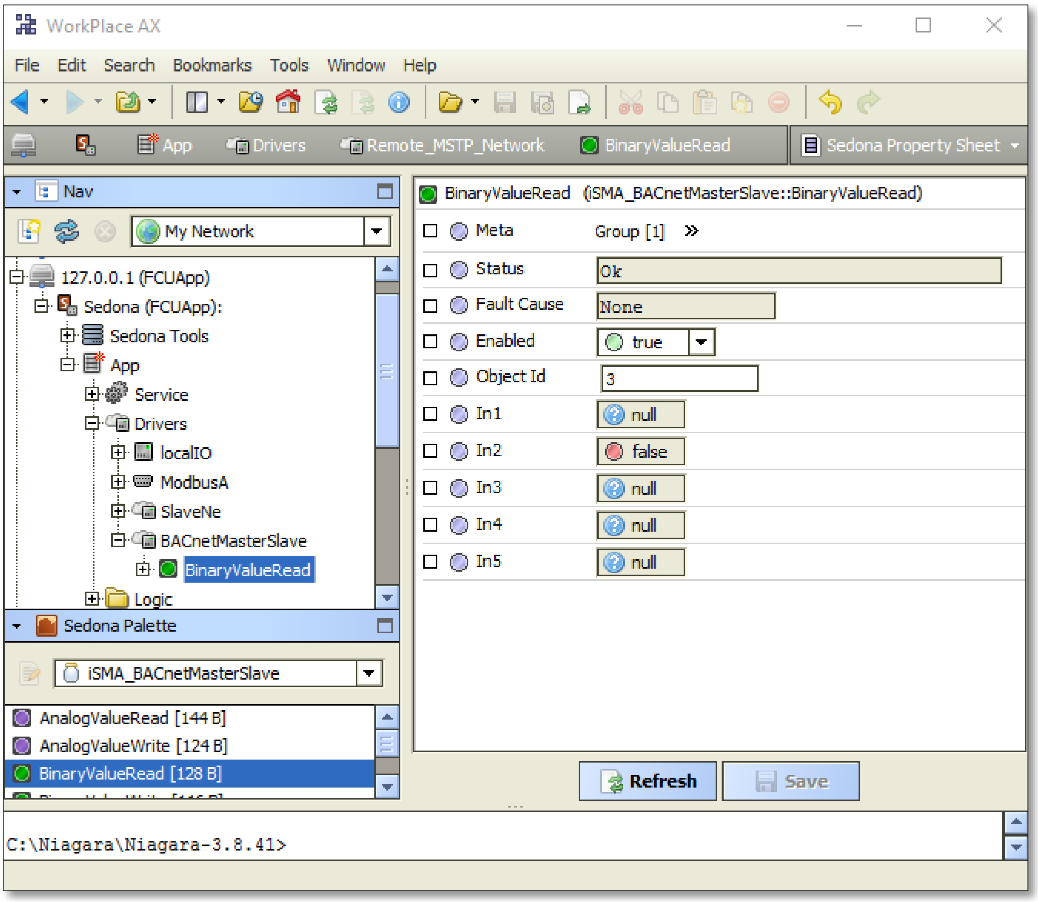


Figure 62 BinaryValueRead component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Enable – Point enabled/disabled ("true"- Point enabled, "false"- Point disabled),
- Object Id – BACnet Id of point,
- In1 – In2 – States of points from slave devices.

The component offers the following action, available under the right mouse button:

- Read – Action enforces the reading of the point.

13.5 BinaryValueWrite

Binary Value Write is a component responsible for sending binary value to slave devices. Value is written in time periods defined by Max Write Time (only when value of Max Write Time is higher than 0) and if value of In slot has changed. Writing can be also enforced by Set or Write action. The component has to be placed under BACnet Master Slave component.

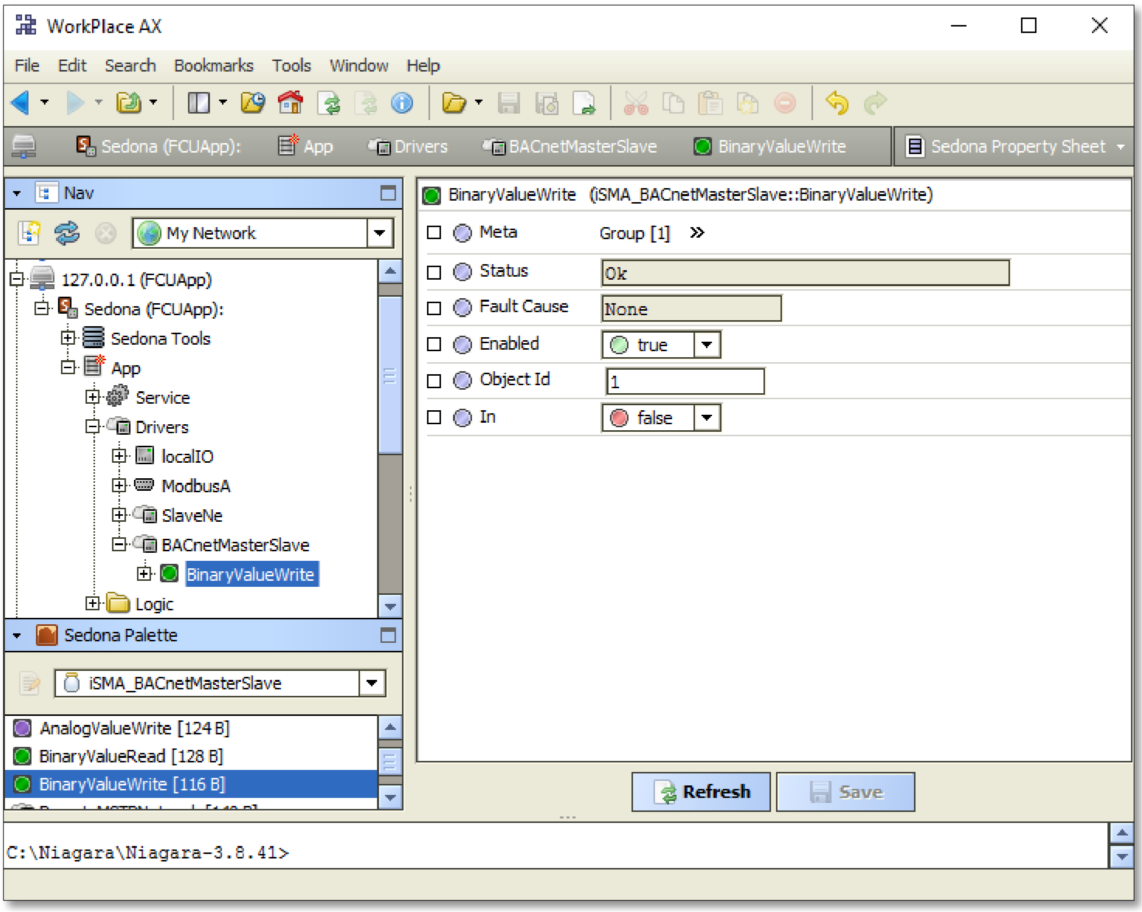


Figure 63 BinaryValueWrite component

The component has the following slots:

- Status – Point status,
- Fault Cause – Fault cause description,
- Enable – Point enabled/disabled ("true"- Point enabled, "false"- Point disabled),

- Object Id – BACnet Id of point,
- In – Input slot; this value is sent to slave devices.

The component offers the following actions, available under the right mouse button:

- Set – Writes value to In slot and sends it to slave devices,
- Write – Sends value from In slot to slave devices.

13.6 BACnet Master Slave Folder

BACnet Master Slave Folder is a component which groups and organizes BACnet Master Slave points components.

14 FCU kit

iSMA_FCU kit includes dedicated components, which can be used in typical FCU application.

14.1 FCU_Antifrost

FCU_Antifrost is a component used to protect against a drop in space temperature below the set threshold (with the hysteresis).

When value of space temperature drops below Threshold – Diff (6 ºC for values in figure below) and there is no sensor fault, Antifrost slot will change to “true”. This Action will be kept until room temperature increases above the value equal to Threshold + Diff (8 ºC for values in figure below). When Temperature sensor is faulty (slot Sensor Fault is set to “true”), Antifrost slot will be always equal “false” – component is inactive.

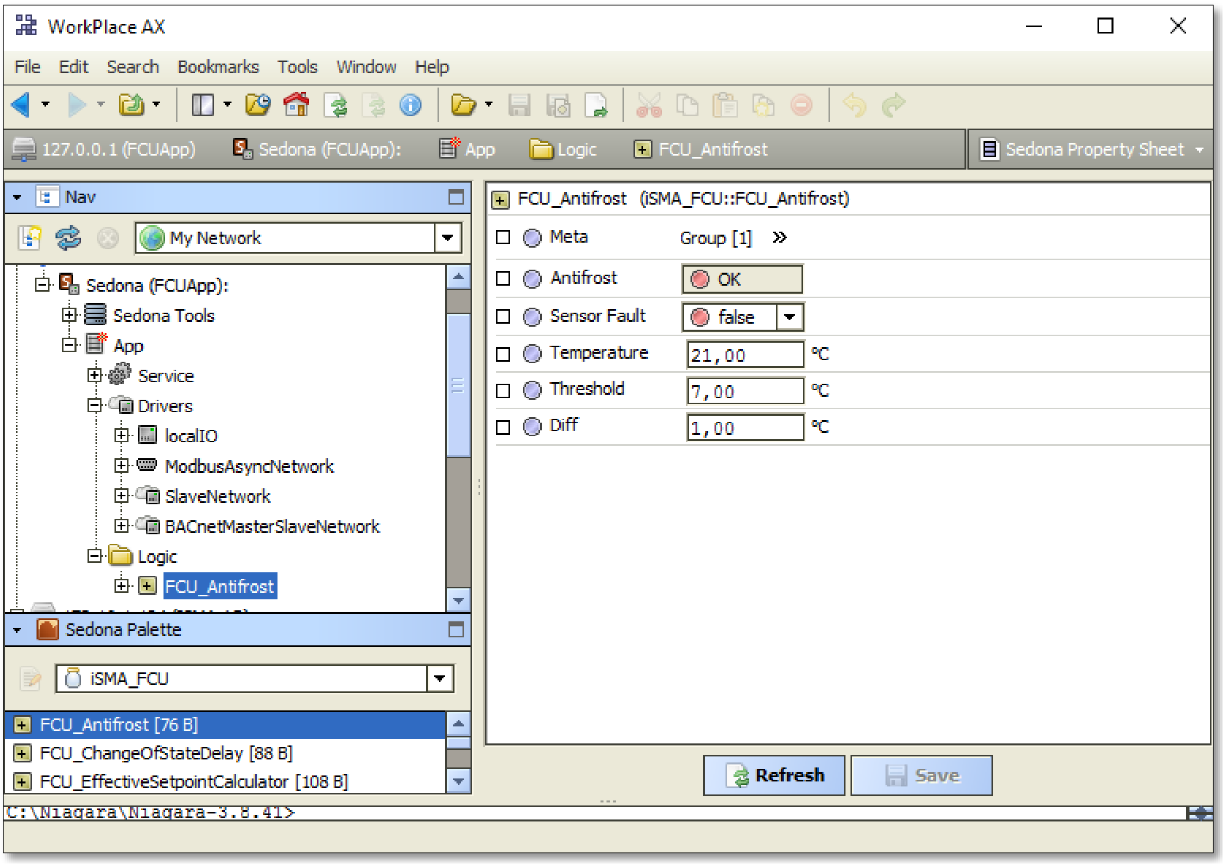


Figure 64 FCU_Antifrost component

The component has the following slots:

- Antifrost – Output Boolean slot; true – “Antifrost”, false – “OK” (no antifrost),
- Sensor Fault – Input Boolean slot, information about fault of temperature sensor; If slot has value “true” – Antifrost component is inactive (because value read from temperature sensor is faulty and will be incorrect) and value of Antifrost slot is set to “false”; if Sensor Fault slot has value “false” – Antifrost component is active,
- Temperature – Value from temperature sensor for controlling space,
- Threshold – Threshold value of temperature,

- Diff – Differential for hysteresis.

14.2 FCU_ChangeOfStateDelay

FCU_ChangeOfStateDelay component allows for delaying binary value of time defined in Delay Time Slot. The component is an extended version of On Delay and Off Delay – both functions are used in one component and there is possibility to set defined by user status to Output slot (during and after delay time).

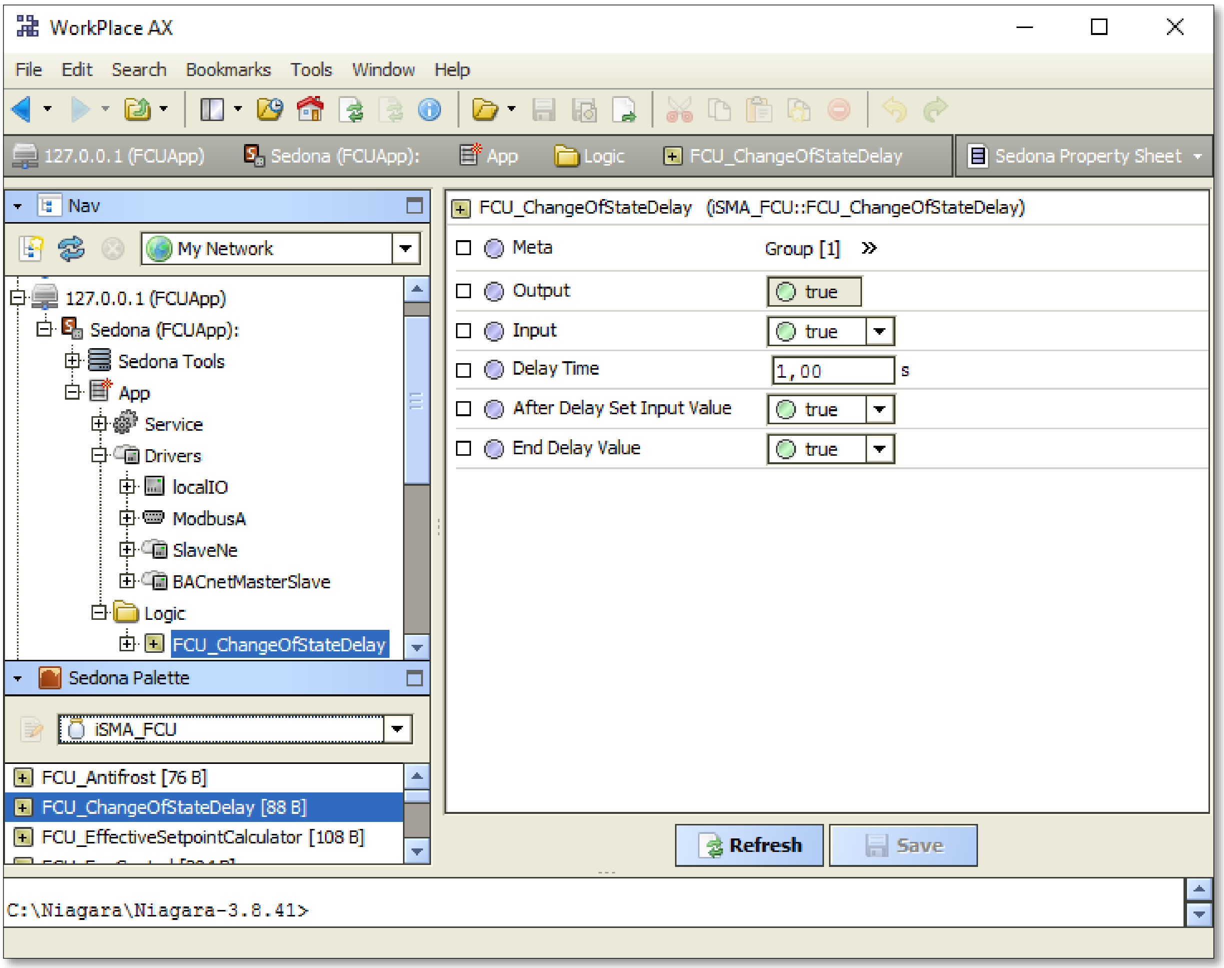


Figure 65 FCU_ChangeOfStateDelay component

The component has the following slots:

- Output – Output Binary slot,
- Input – Input Binary slot; when state of slot has changed (disregarding rising or falling edge), the component starts counting delay time,
- Delay time – Value of delay time in seconds,
- After Delay Set Input Value – Component mode, available values:
    - true – If state of Input slot has changed and counting of delay time has been initiated, Output slot has changed to state from Input slot; If counting is in progress, the previous state is set to Output slot. For example, if Input slot is in “true” state and it changes to “false”, it means that Output slot is still set to “true” (counting is in progress). When delay time ends, Output slot will be set to “false”.
    - false – If state of Input slot has changed and counting of delay time has been initiated, Output slot has changed to state from End Delay Value slot. During the counting process, Output slot is set in opposite to state of End Value Slot.
- End Delay Value – State, which is set to Output slot, after counting process has ended. During counting process, Output slot is set in opposite to state of End Value Slot. This option is available only when After Delay Set Input Value slot has value false.

14.3 FCU_EffectiveSetpointCalculator

FCU_EffectiveSetpointCalculator is a component which allows calculation of value of effective temperature setpoint, according to setpoint value from higher level system (for example BMS), offset of this value, occupancy mode and FCU mode (heating or cooling).

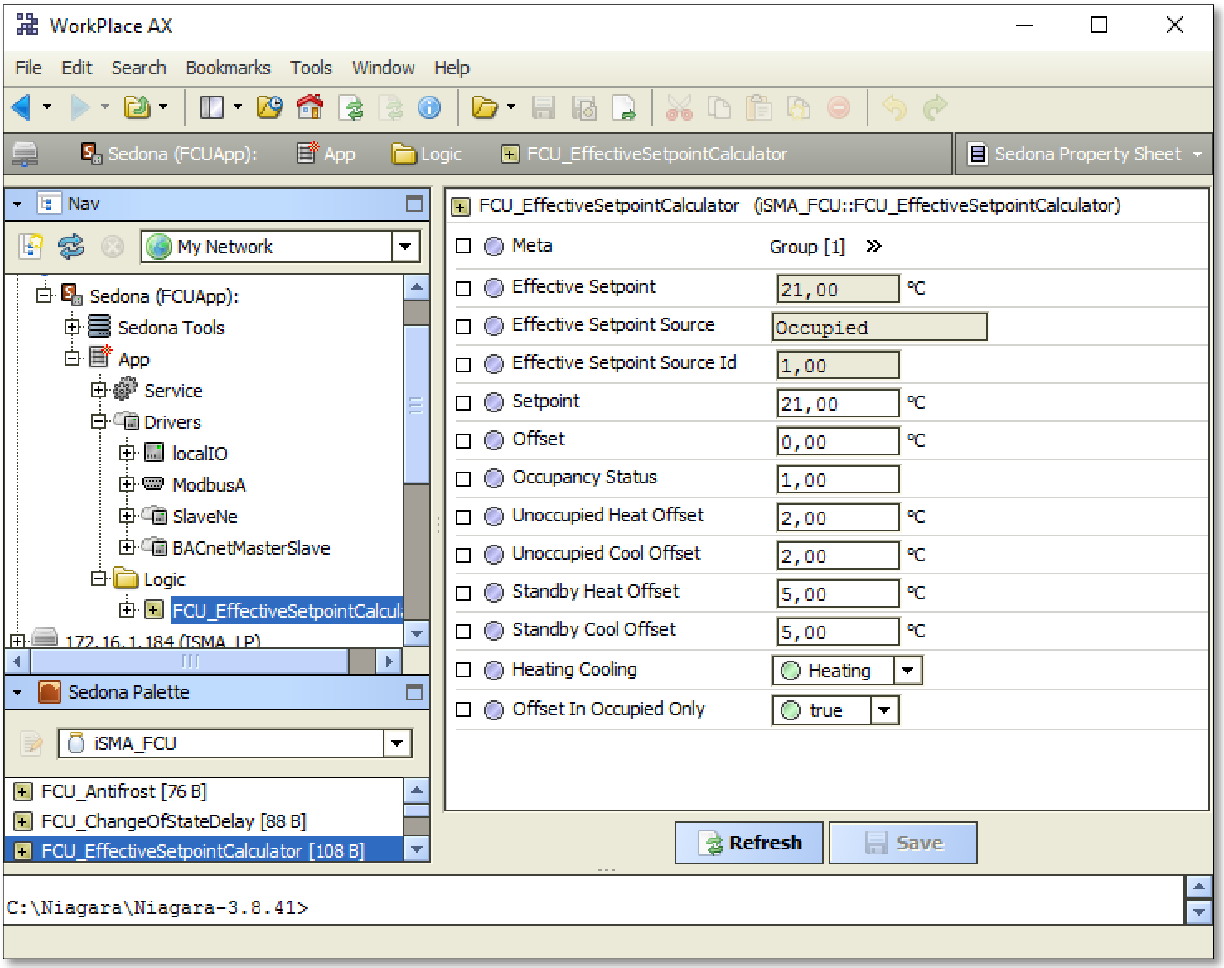


Figure 66 FCU_EffectiveSetpointCalculator component

The component has the following slots:

- Effective Setpoint – Main output slot of a component, value is equal to calculated effective temperature setpoint, according to values of other slots of the component,
- Effective Setpoint Source – Slot displays information about method of calculation, and Effective Setpoint, as well as according to occupancy status and FCU mode, the Slot has the following values:
    - 1 – Occupied,
    - 2 – Unoccupied_Heating,
    - 3 – Unoccupied_Cooling,
    - 4 – Standby_Heating,
    - 5 – Standby_Cooling.
- Effective Setpoint Source Id – slot, which displays numeric value, corresponding to Effective Setpoint Source slot,
- Setpoint – main input of component, to which value of temperature setpoint from higher

level system is connected,

- Offset – Value of setpoint offset,
- Occupancy Status – value corresponding to occupancy status:
  - 0 – Unoccupied
  - 1– Occupied
  - 2 – Standby.
- Unoccupied Heat Offset – Value of offset subtracted from setpoint in Unoccupied mode, when algorithm works in Heating mode.
- Unoccupied Cool Offset – Value of offset added to setpoint in Unoccupied mode, when algorithm works in Cooling mode.
- Standby Heat Offset – Value of offset subtracted from setpoint in Standby mode, when algorithm works in Heating mode.
- Standby Cool Offset – Value of offset added to setpoint in Standby mode, when algorithm works in Cooling mode.
- Heating Cooling – Temperature mode, “true” – component works in Heating mode, “false” – component works in Cooling mode.
- Offset In Occupied Only – Boolean value that determines whether the calculation of Effective Setpoint is to be included in process of calculating the value of the Offset slot, if a component is not in Occupied mode; “true” – for Unoccupied and Standby modes the value of Offset slot is not to be included in process of calculating the Effective Setpoint; “false” – value of Offset slot is to be included in process of calculating the Effective Setpoint in all occupancy modes.

Methods of calculating the Effective Setpoint value, according to settings of Effective Setpoint Calculator component are presented in the below table:

Occupancy Status	Heating / Cooling	Offset in Occupied Only	Effective Setpoint	Effective Setpoint Source
0 (Unoccupied)	Heating	False	Effective Setpoint = Setpoint + Offset – Unoccupied Heating Offset	2 (Unoccupied Heating)
0 (Unoccupied)	Cooling	False	Effective Setpoint = Setpoint + Offset + Unoccupied Cooling Offset	2 (Unoccupied Cooling)
0 (Unoccupied)	Heating	True	Effective Setpoint = Setpoint – Unoccupied Heating Offset	2 (Unoccupied Heating)
0 (Unoccupied)	Cooling	True	Effective Setpoint = Setpoint + Unoccupied Cooling Offset	2 (Unoccupied Cooling)
1 (Occupied)	-	-	Effective Setpoint = Setpoint + Offset	0 (Occupied)
2 (Standby)	Heating	False	Effective Setpoint = Setpoint + Offset – Standby Heating Offset	2 (Standby Heating)
2 (Standby)	Cooling	False	Effective Setpoint = Setpoint + Offset + Standby Cooling Offset	2 (Standby Cooling)
2 (Standby)	Heating	True	Effective Setpoint = Setpoint – Standby Heating Offset	2 (Standby Heating)
2 (Standby)	Cooling	True	Effective Setpoint = Setpoint + Standby Cooling Offset	2 (Standby Cooling)

Table 4 Effective Setpoint calculation

14.4 FCU_FanControl

FCU_FanControl is a component which allows to control the fan. The component has been created for 1-, 2- or 3-speed fans and for fans with analog inputs (type of fan can be selected by the user). Fan algorithm can be split to two modes:

- Standard – when demand for the fan is active and fan speed is calculated based on temperature value. The standard mode is when:
  - FanDemand is false
  - Antifrost is false
  - HeatingOccupiedActive is false
  - CoolingOccupiedActive is false
  - TestMode is 0.
- Non-Standard - when additional parameters override fan speed. The Non-Standard mode must comprise at least one of the following slot states: FanDemand, Antifrost, HeatingOccupiedActive, CoolingOccupiedActive is “true: or Test Mode is higher than 0.

In standard mode the fan will be switched on when the internal variable FanControlValue (calculated on the basis of CV and Setpoint) is higher than Fan Speed 1 Threshold and switched off when Fan Control Value is lower than Fan Off Threshold.

The Non-Standard operation is defined by the slot states combinations and are described below.

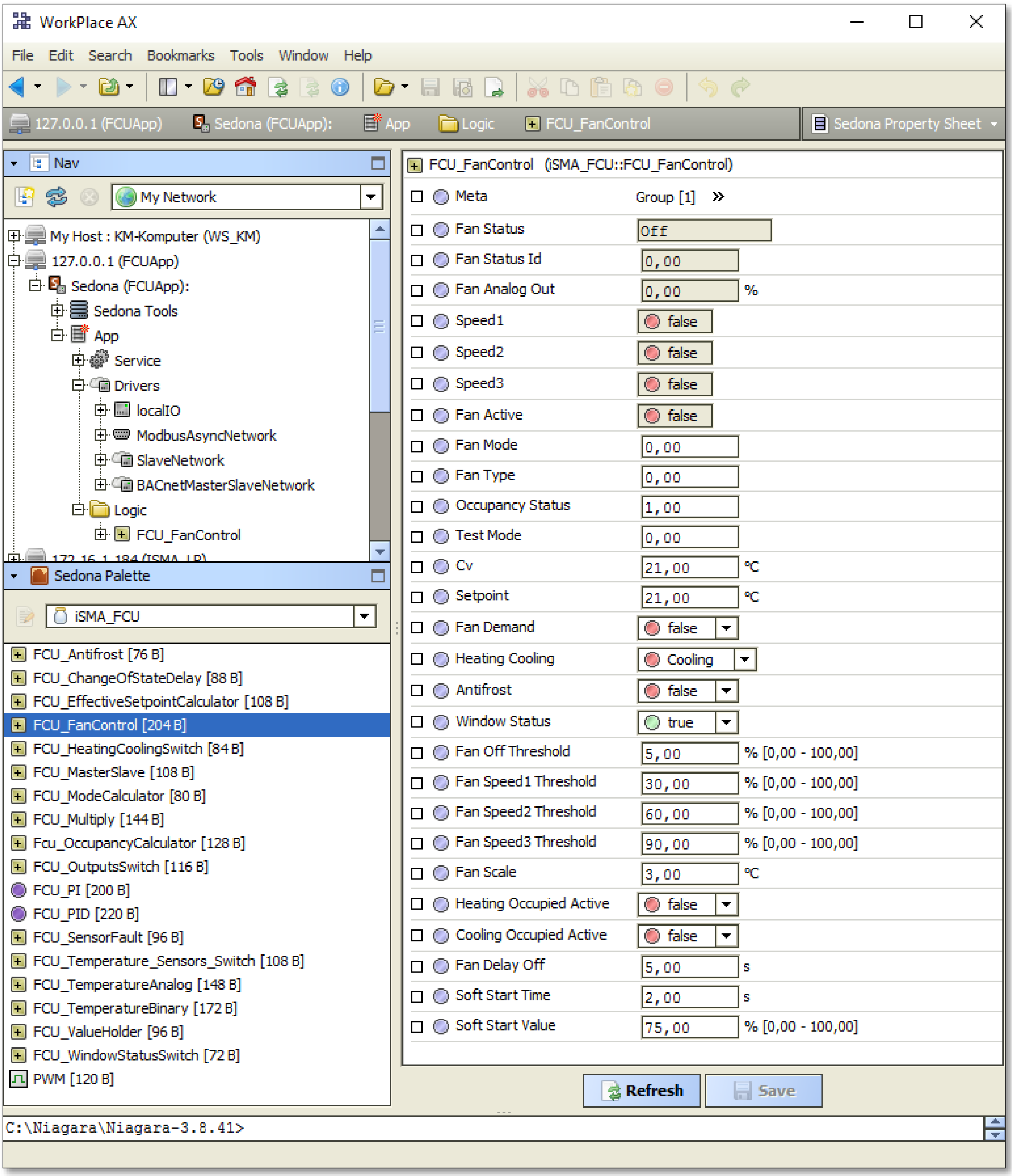


Figure 67 FCU_FanControl component

The component has the following slots:

- Fan Status – Slot displays current status of a fan. Available values:
    - 0 – Off
    - 1 – Speed 1 (Manual)
    - 2 – Speed 2 (Manual)

  - 3 – Speed 3 (Manual)
  - 4 – Speed 1 (Auto)
  - 5– Speed 2 (Auto)
  - 6 – Speed 3 (Auto)
- Fan Status Id – Numeric value corresponding to Fan Status slot.
- Fan Analog Out – Output of component for fan with analog input, expressed as percentage. For fans with discrete inputs (Fan Type slot is set to 1, 2 or 3), value of Fan Analog out is equal 0%.
- Speed 1, Speed 2, Speed 3 – Outputs for fans with binary inputs. For fans with analog input (Fan Type slot is set to 0), states of Speed 1, Speed 2 and Speed 3 slots are set to “false” and cannot be changed by algorithm of component.

  Note: The FCU_FanControl component has built-in protection against enabling several speeds at the same time, what can cause physical damage to the fan, and is connected to iSMA-B-FCU device. When current speed have to be changed to other one, all binary outputs responsible for speeds are disabled for 1 second and after that, a new speed is enabled.
- Fan Active – Binary output slot to confirm operation of a fan. If value of Fan Status ID slot is higher than 0, Fan Active slot is set to “true”. In other cases state of slot is set to “false”.
- Fan Mode – Main input of described component. Available values:
  - 0 – Off – Fan is disabled
  - 1 – Manual Speed 1 – Fan works with speed 1, regardless of temperature values. If Fan Type slot is set to 0 (fan with analog input), value of Fan Analog Out is set to value from Fan Speed 1 Threshold slot.
  - 2 – Manual Speed 2 – Fan works with speed 2, regardless of temperature value. If Fan Type slot is set to 0 (fan with analog input), value of Fan Analog Out is set to value from Fan Speed 2 Threshold slot.
  - 3 – Manual Speed 3 – Fan works with speed 3, regardless of the temperature values. If Fan Type slot is set to 0 (fan with analog input), value of Fan Analog Out is set to value from Fan Speed 3 Threshold slot.
  - 4 – Auto – Fan works in automatic mode, current speed depends on current space temperature and set Setpoint.

Note: Value of Fan Mode slot (or current speed, without change of Fan Mode slot) can be overridden by in-built algorithm of a component, disregarding the value that is set to Fan Mode slot. It can occur in the following cases:

- The component works in Unoccupied or Standby mode (value of Occupancy Status slot equals 0 or 2) – Fan Mode slot is overridden to 4 (Auto mode), always when set value is different than 0. The slot’s overriding will stop when a component will work in Occupancy mode (value of Occupancy Status slot will be equal to 1).
- Window is Open (Window Status slot is set to “false”) – Fan Mode slot is overridden to 0 (Off mode). Overriding will stop when Window Status slot will change to “true”.
- The component works in Antifrost mode (Antifrost slot is set to “true”, even slot Window Status is set to “false”) – current speed will be overridden by maximum value available for type of the fan (depending on value of Fan Type slot).

Overriding will stop when Antifrost slot will change to “false”.

- The component works in Testing mode (value of Test Mode slot is not equal to 0) – current speed will be overridden by maximum value available for type of the fan (depending on value of Fan Type slot). Overriding will stop when Test Mode slot will change to 0.

- Fan Type – numeric value corresponding to type of controlled fan. Available values:
    - 0 – Fan with analog input
    - 1 – Fan with 1 binary speed
    - 2 – Fan with 2 binary speeds
    - 3 – Fan with 3 binary speeds.

Note: The FCU_FanControl component has built-in protection against enabling speeds higher than these resulting from value of Fan Type slot. For example, if Fan Type slot is set to 1 (fan with 1 binary speed), it is not possible to enable speeds higher than 1. This protection pertains only to fans with binary outputs.

- Occupancy Status – numeric slot corresponding to occupancy status of space. Available values:
    - 0 – Unoccupied
    - 1 – Occupied
    - 2 – Standby.
- Test Mode – numeric input slot for enabling or disabling Testing mode. This mode is inactive when value of slot equals 0. In other cases, fan works in Testing Mode – current speed will be overridden by maximum value available for fan type (depending on value of Fan Type slot)
- Cv – value of space temperature, which is used for calculating fan speed in Auto mode
- Setpoint – value of space temperature setpoint, which is used for calculating fan speed in Auto mode
- Fan Demand – binary slot, used to force switching the fan ON, when it is off (Fan Active slot is equal to “false”). If fan is Off (Fan Mode is set to 0) or works in Auto mode (Fan Mode is set to 4), but speed calculated by algorithm equals 0, fan can be switched on by setting “true” to Fan Demand slot. In this case fan works with speed 1 (for fans with binary inputs) or with analog value set to Fan Speed 1 Threshold slot (for fan with analog input). When speed (or analog output) calculated by algorithm is higher than speed 1 (or value from Fan Speed 1 Threshold slot, for analog output), speed switches ON using Fan Demand, overridden by this value.
- Heating Cooling – binary input slot with information in which temperature mode fan should actually work (Heating or Cooling); “true” – component is working in Heating mode, “false” – component is working in Cooling mode.
- Antifrost – binary input slot to switch on Antifrost mode:
    - true – Antifrost mode enabled – current speed will be overridden by maximum value available for fan type (depending on value of Fan Type slot)
    - false – Antifrost mode disabled.
- Window Status - binary input slot to enable Window Is Open mode:
    - true – Window Is Open mode disabled
    - false – Window Is Open mode enabled – current value of Fan Mode slot will be

overridden by 0 (Off Mode).

Note: Window Is Open mode can be overridden only by Antifrost mode or Test mode.

- Fan Off Threshold, Fan Speed 1 Threshold, Fan Speed 2 Threshold, Fan Speed 3 Threshold – Values of thresholds used for switching fan speeds (in Auto mode, for fans with binary inputs), calculating value of Fan Analog Out slot (in Manual modes, for fan with analog output).
- Fan Scale – numeric input slot with value, which is used for calculating fan speed in Auto mode.

Note: Calculating the fan speeds is based on internal variable named Fan Control Value. The way of calculating this value is presented by the figure below:

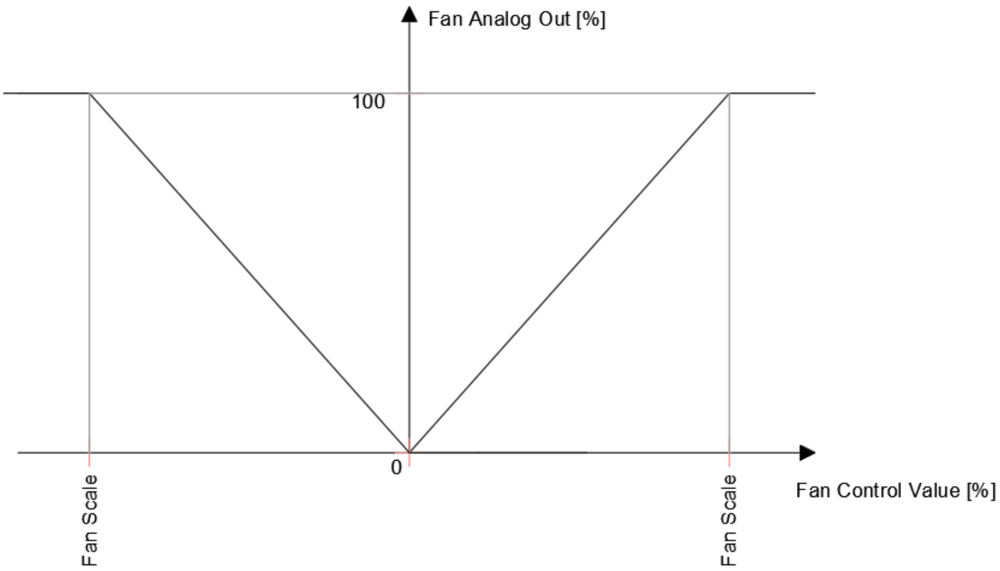


Figure 68 Way of calculating Fan Control Value

Fan Control Value is used to calculate current speed of the fan (for fans with binary inputs) or value of Fan Analog Out slot (for fans with analog input). The way of calculating both values is presented by the below figures:

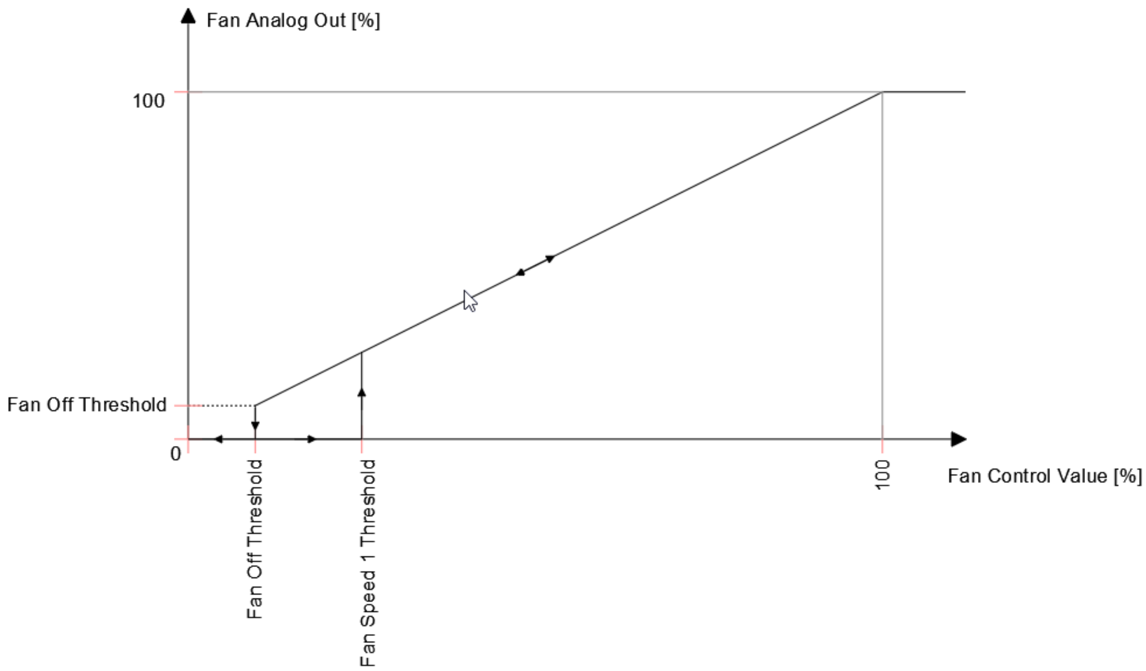


Figure 69 Control of fan with analog input in Auto mode

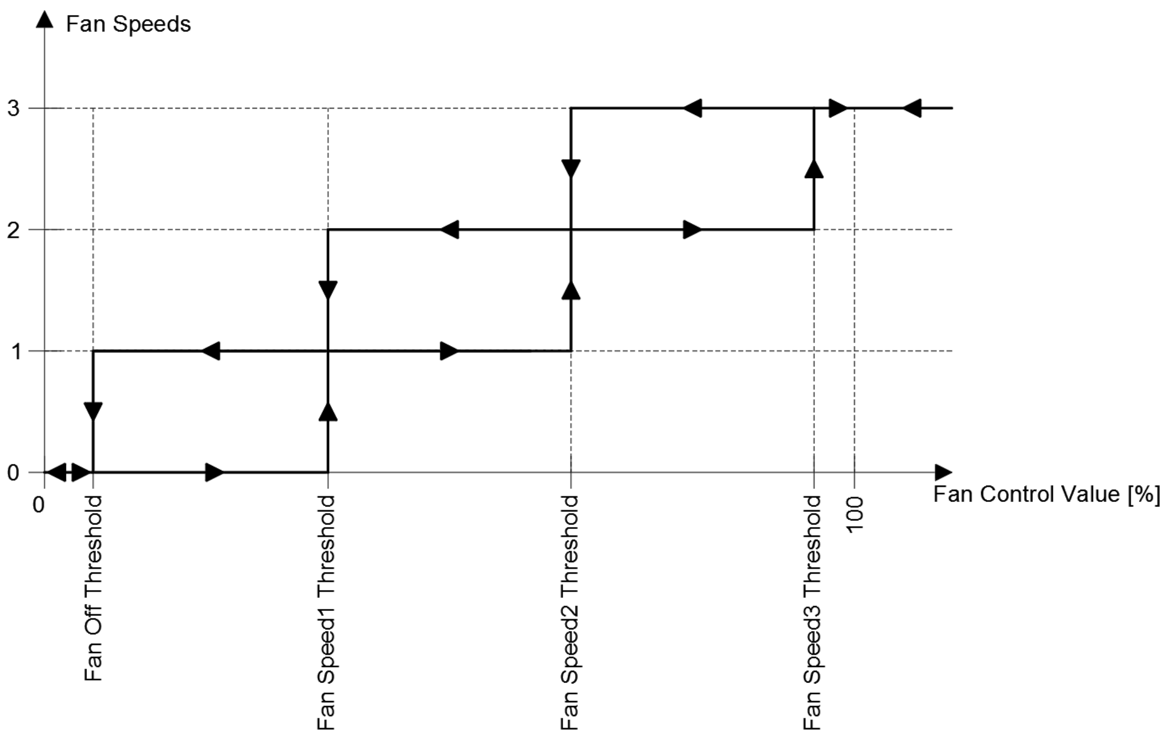


Figure 70 Control of fan with binary inputs in Auto mode

- Heating Occupied Active – binary input slot, enable/disable function, enforcing fan operation. Available values:

- True – function is enabled – if fan works in Auto mode (Fan Mode slot is set to 4), space is occupied (Occupancy Status slot is set to 1) and FCU_FanControl component works in Heating temperature mode (Heating Cooling slot is set to "true"), Fan will always be switched on, even if value of Setpoint slot is lower than value of Cv slot and (according to the control algorithm) fan should be switched off.
- False – function is disable.

• Cooling Occupied Active – binary input slot, enable/disable function, enforcing fan operation. Available values:
  - True – function is enabled – if fan works in Auto mode (Fan Mode slot is set to 4), space is occupied (Occupancy Status slot is set to 1) and FCU_FanControl component works in Cooling temperature mode (Heating Cooling slot is set to "false"), Fan will always be switched on, even if value of Setpoint slot is higher than value of Cv slot and (according to the control algorithm) fan should be switched off.
  - False – function is disabled.

Note: The way of calculating current speed of fan (for fans with binary inputs) or value of Fan Analog Out slot (for fan with analog input), when Cooling/Heating Occupied Active function is enabled, is presented in the figures below:

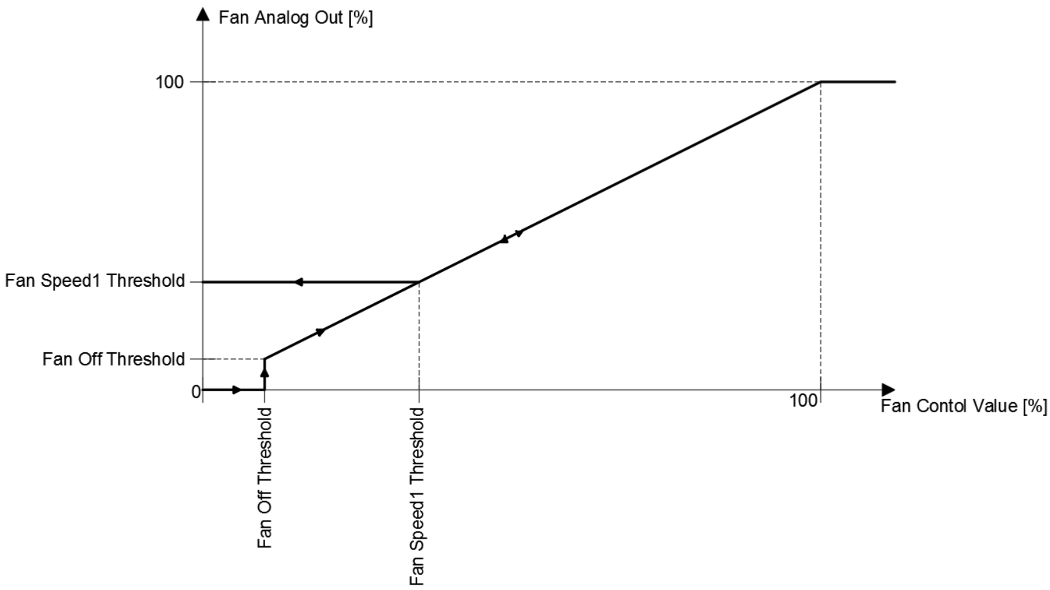


Figure 71 Heating/Cooling Occupied Active function for fan with analog input

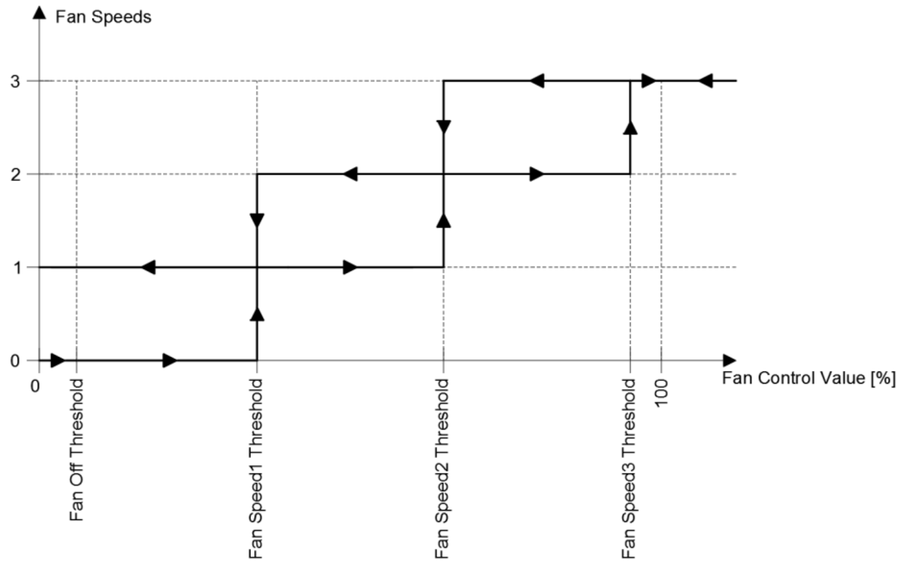


Figure 72 Heating / Cooling Occupied Active function for fans with binary inputs

- Fan Delay Off – value of delay off time, expressed in seconds. Every time, when value of slot Fan Status ID is higher than 0, and fan should be switched off, it works with current speed by time equal to value of this slot. After this time, Fan will switch off. If slot is set to 0, function is disabled.

- Soft Start Time – time, in which fan is working in Soft Start mode, in seconds. Value 0 disables this function.
- Soft Start Value – value for Soft Start mode, expressed as percentage.

Note: If Soft Start Value is lower than Fan Speed 1 Threshold, the value will be taken from Fan Speed 1 Threshold

Note: Soft Start function is dedicated to fans with Analog Input. If fan start with small control value ramp lasts too long or is impossible, overheating of driver or motor can occur. In this function, fan start output value will be increased to Fan Soft Star Value for time defined in “Fan Soft Start Time”. If time of soft start is finished, Fan Analog Out slot is set to current value calculated by algorithm of the component.

14.5 FCU_HeatingCoolingSwitch

The FCU_HeatingCoolingSwitch component allows for switching between Heating and Cooling temperature mode, according to current temperature, its setpoint, occupancy status, etc.

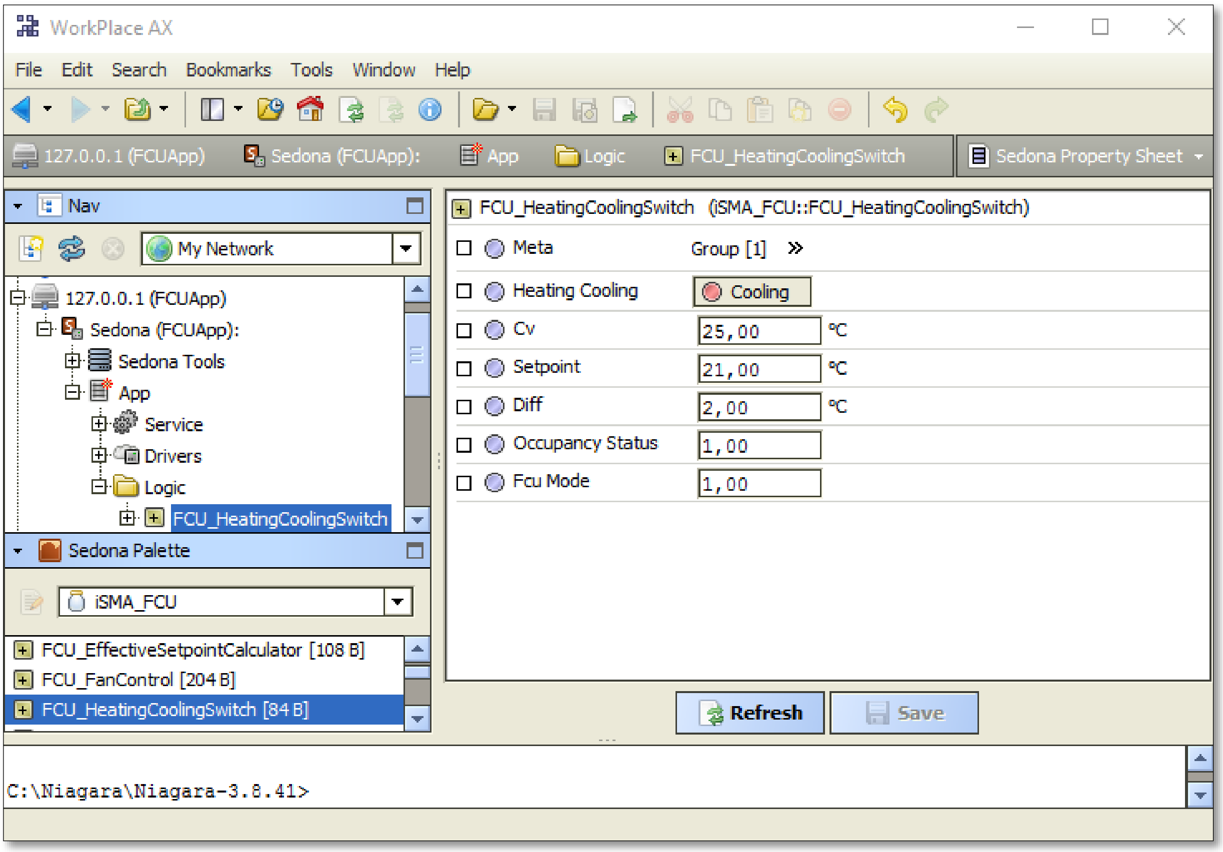


Figure 73 FCU_HeatingCoolingSwitch component

The component has the following slots:

- Heating Cooling – main output of a component. Available values:

    - True – Heating mode
    - False – Cooling mode
- Cv – controlled temperature,
- Setpoint – Setpoint of controlled temperature,
- Diff – Differential for hysteresis,
- Occupancy Status – value corresponding to occupancy status. Available values:
    - 0 – Unoccupied
    - 1 – Occupied
    - 2 – Standby
- Fcu Mode – value corresponding to FCU mode, for example from higher level system. Available values:
    - 0 – Off – FCU is disabled; Heating Cooling slot is set to “true”.
    - 1 – Auto – FCU works in Auto mode, it is possible to switch between Heating and Cooling modes.
    - 2 – Heating Only – FCU works only in Heating mode; Heating Cooling slot is set to “true”
    - 3 – Cooling Only – FCU works only in Cooling mode; Heating Cooling slot is set to “false”
    - 4 – Fan Only – Only fan can work; Heating Cooling slot is set to “false”.

The FCU_HeatingCoolingSwitch component has predefined modes for values from Fcu Mode slot:

- If Fcu Mode slot is set to 0 (Off), slot Heating Cooling will always be set to “true” (Heating Mode), regardless of temperature and setpoint values.
- If Fcu Mode slot is set to 2 (Heating Only), slot Heating Cooling will always be set to “true” (Heating Mode), regardless of temperature and setpoint values.
- If Fcu Mode slot is set to 3 (Cooling Only), slot Heating Cooling will always be set to “false” (Cooling Mode), regardless of temperature and setpoint values.
- If Fcu Mode slot is set to 4 (Fan Only), slot Heating Cooling will always be set to “false” (Cooling Mode), regardless of temperature and setpoint values.
- If Fcu Mode slot is set to 1 (Auto), slot Heating Cooling is switching between Heating and Cooling, according to the figure below:

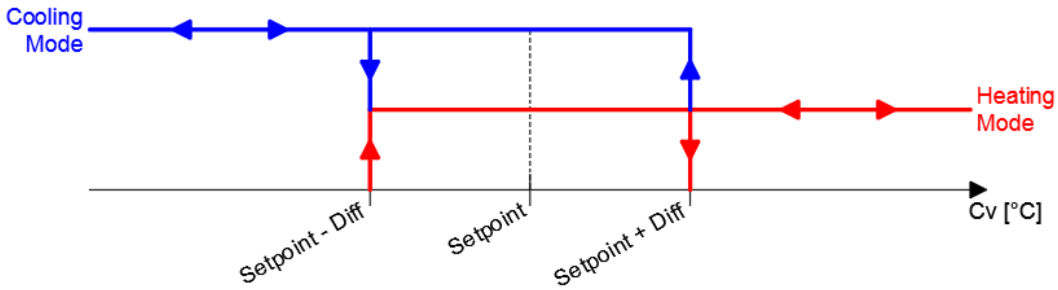


Figure 74 Switching between Heating and Cooling mode

Note: Heating and Cooling modes can be switched only when Occupancy Status slot is set to 1 (Occupied mode). If Occupancy Status slot has the other value, Heating Cooling slot is set to the last mode which has been calculated in Occupied mode.

14.6 FCU_MasterSlave

The FCU_MasterSlave component allows to automatically calculate BACnet Device Id of Slave devices in BACnet Master Slave Network, depending on the BACnet Device ID of Master devices. This function is called Auto Binding. The table below shows values of Master BACnet Device Id and corresponding to them BACnet Device Id of Slave devices for Auto Binding Function:

L ` r sdq Hc	Rk ud 0 HC	Rk ud 1 HC	Rk ud 2 HC	Rk ud 3 HC	Rk ud 4 HC
826101	826001	826002	826003	826004	826005
826102	826006	826007	826008	826009	8260010
826103	826011	826012	826013	826014	826015
826104	826016	826017	826018	826019	826020
826105	826021	826022	826023	826024	826025
826106	826026	826027	826028	826029	826030
826107	826031	826032	826033	826034	826035
826108	826036	826037	826038	826039	826040
826109	826041	826042	826043	826044	826045
826110	826046	826047	826048	826049	826050
826111	826051	826052	826053	826054	826055
826112	826056	826057	826058	826059	826060
826113	826061	826062	826063	826064	826065
826114	826066	826067	826068	826069	826070
826115	826071	826072	826073	826074	826075
826116	826076	826077	826078	826079	826080
826117	826081	826082	826083	826084	826085
826118	826086	826087	826088	826089	826090
826119	826091	826092	826093	826094	826095
826120	826096	826097	826098	826099	826100
Other	0	0	0	0	0

Table 5 Master Slave Id - Auto Binding

Auto Binding function can be disabled (by setting value true to Local Remote Auto Binding slot). In this case, Id of Slave devices has to be set by the user (in slots Remote Slave 1 Device Id – Remote Slave 5 Device Id).

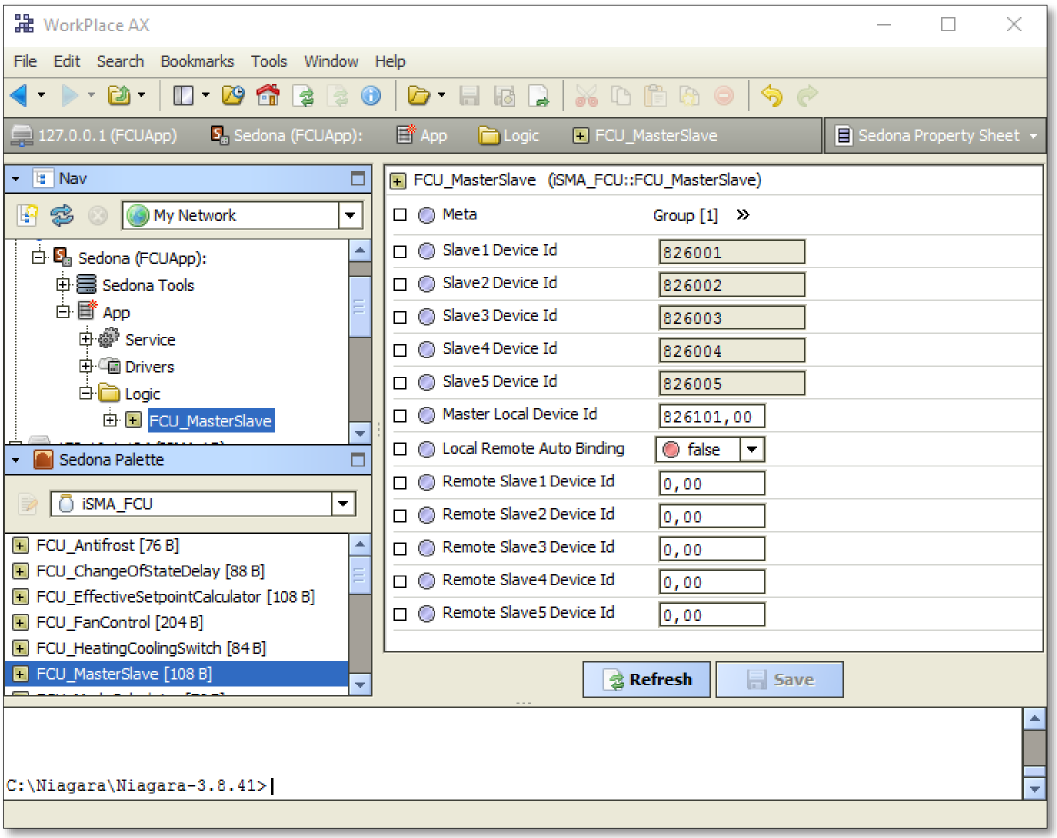


Figure 75 FCU_MasterSlave

The component has the following slots:

- Slave1 Device Id – Slave5 Device Id – outputs of component, display calculated or set Id of five Slave devices,
- Master Local Device Id – Id of Master device.

Note: If component uses Auto Binding function, value of Master Local Device Id slot has to be set to value from range 826101 – 826120. For other values, all output slots (Slave1 Device Id – Slave5 Device Id) will be set to 0.

- Local Remote Auto Binding – slot allows for switching between Auto Binding and Remote Binding, available values:
  - True – Remote Banding – IDs of each Slave Device (slots Slave1 Device Id – Slave5 Device Id) are set to corresponding to them values of Remote IDs (slots Remote Slave 1 Device Id – Remote Slave 2 Device Id)
  - False – Auto Binding – IDs of each Slave Device (slots Slave1 Device Id – Slave5 Device Id) are calculated according to the table above.
- Remote Slave 1 Device Id – Remote Slave 2 Device Id – slots with remote Id of Slave devices. If slot Local Remote Auto Binding is set to “true”, these values are set to corresponding outputs (slots Slave1 Device Id – Slave5 Device Id).

14.7 FCU_ModeCalculator

The FCU_ModeCalculator component allows for switching of main modes of application (for example override Fan Mode, according to others values or enable/disable temperature control), according to Fan Mode, Fcu Mode, and type of temperature control (analog or binary). The component can be used to protect against incorrect operation of FCU device.

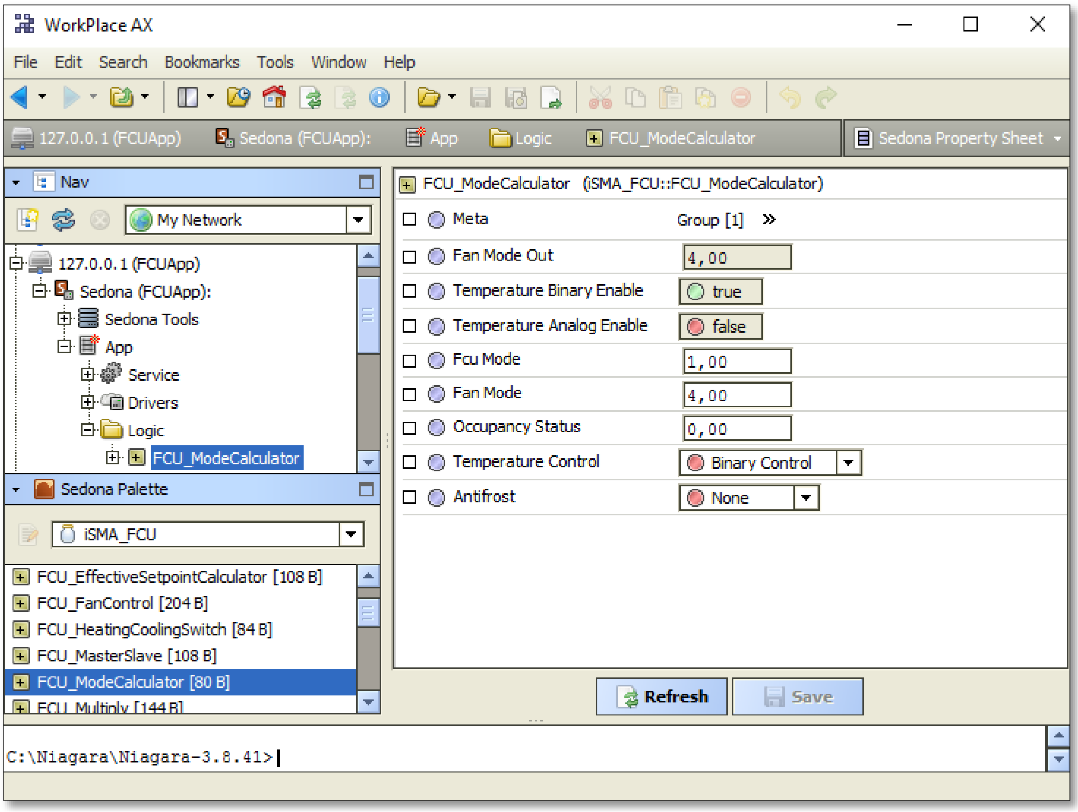


Figure 76 FCU_ModeCalculator component

The component has the following slots:

- Fan Mode Out – Output of Fan Mode value, calculated according to the algorithm of component:
    - If value of Fcu Mode slot is equal to 0 (FCU is switched Off from upper level system), slot Fan Mode Out is also set to 0 (fan is also switched off).
    - If value of Fan Mode slot is lower than 4 (Fan is Off or works in one of manual modes) and value of slot Occupancy Status is not equal to 1 (the component works in Unoccupied or Standby mode), slot Fan Mode Out is set to 4 (fan works in Auto mode).
    - In other cases, value of Fan Mode Out slot is equal to value of Fan Mode slot.

- Occupancy Status – value corresponding to Occupancy Status; available values:
    - 0 – Unoccupied,
    - 1 – Occupied,
    - 2 – Standby.
- Temperature Binary Enable – output binary slot used to enable or disable binary temperature mode, according to the algorithm of component:
    - If slot Temperature Control is set to “false” (Binary Control) and value of Fan Mode Out is higher than 0 (fan is switched on), Temperature Binary Enable slot is set to “true”.
    - If slot Temperature Control is set to “false” (Binary Control) and slot Antifrost is set to “true” (component works in Antifrost mode), Temperature Binary Enable slot is set to “true”.
    - In the other cases, Temperature Binary Enable slot is set to “false”.
- Temperature Analog Enable – output binary slot used to enable or disable analog temperature mode, according to the algorithm of the component:
    - If slot Temperature Control is set to “true” (Analog Control) and value of Fan Mode Out is higher than 0 (fan is switched on), Temperature Analog Enable slot is set to “true”.
    - If slot Temperature Control is set to “true” (Analog Control) and slot Antifrost is set to “true” (component works in Antifrost mode), Temperature Analog Enable slot is set to “true”.
    - In the other cases, Temperature Analog Enable slot is set to “false”.
- FCU Mode – numeric input with value corresponding to mode of the FCU; available values:
    - 0 – Off
    - 1 – Auto
    - 2 – Heating Only
    - 3 – Cooling Only
    - 4 – Fan Only.

Note: If FCU Mode slot is set to 4 (Fan Only) and Antifrost slot is set to “false” (no Antifrost mode), slots Temperature Binary Enable and Temperature Analog Enable are overridden to “false”.

- Fan Mode – numeric input with value corresponding to fan mode; available values:
  - 0 – Off
  - 1 – Manual Speed 1
  - 2 – Manual Speed 2
  - 3 – Manual Speed 3
  - 4 – Auto
- Temperature Control – binary input slot to select mode of temperature control; available values:
  - True – Analog Control
  - False – Binary Control
- Antifrost – binary input slot to enable/disable Antifrost mode;
  - True – Antifrost mode enabled
  - False – Antifrost mode disabled.

14.8 FCU_Multiply

The FCU_Multiply component allows to multiply up to 10 different values by the same value.

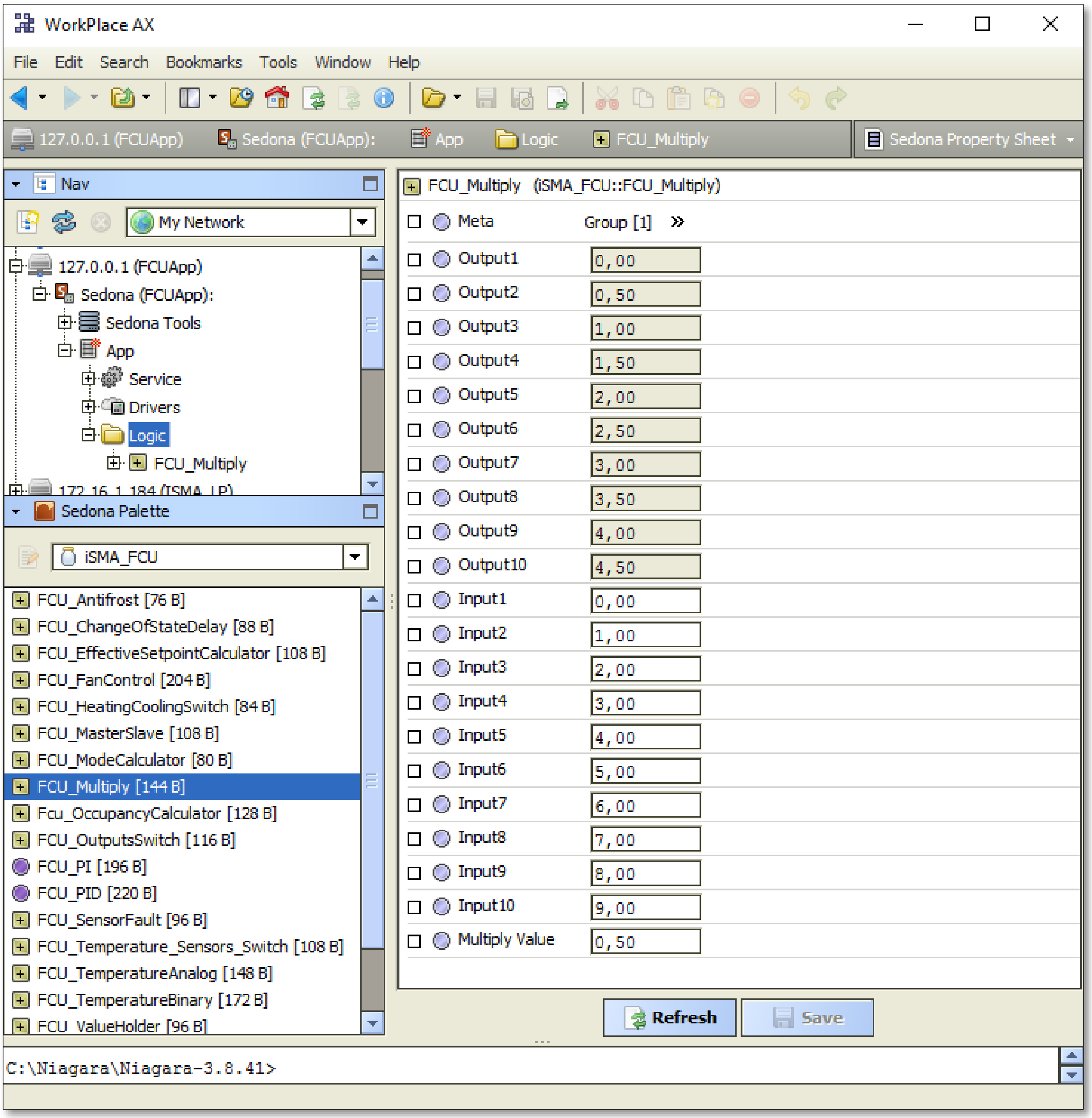


Figure 77 FCU_Multiply component

The component has the following slots:

- Output1 – Output10 – Output slots of the component,
- Input1 – Input10 – Input slots of the component,
- Multiply Value – Value by which Inputs are multiplied,

14.9 FCU_PI

The FCU_PI component is a regulator with proportional and integral actions.

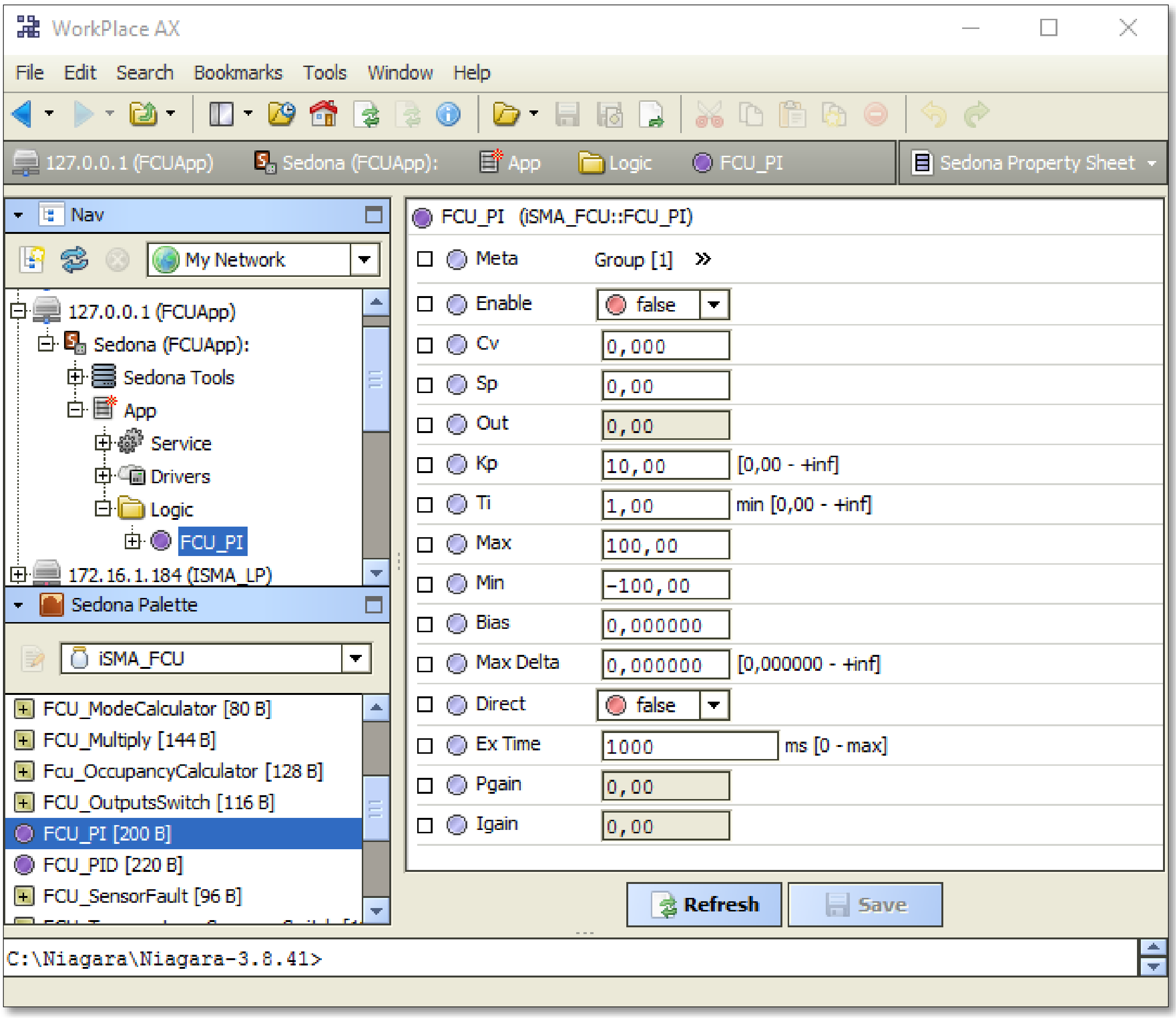


Figure 78 FCU_PI component

The component has the following slots:

- Enable – Loop enable slot. If component is disabled (Enable slot is set to “false”), Out slot is set to 0,
- Cv – Numeric input slot with controlled value,
- Sp – Numeric input with setpoint of controlled value,
- Out – Output slot of component,
- Kp – Numeric input with value of proportional gain constant,
- Ti – Numeric input with value of integral time constant. Setting to 0 disables this integral action,
- Max – Numeric Input slot with maximal value of output of component,
- Min – Numeric input slot with minimal value of output of component,

- Bias – Numeric input with bias. This value is added to Out slot, if Ti slot is set to 0,
- Max Delta – Numeric input slot with the maximum amount out can change in exTime. Setting to 0 disables this function,
- Direct – Binary input slot to set acting process; available values:
    - True – direct acting process
    - False – reverse acting process
    - Ex Time – Numeric input slot with period of loop execution,
    - Pgain – Numeric output with value of output signal, which was calculated by proportional action,
    - Igain – Numeric output with value of output signal, which was calculated by integral action.

14.10FCU_PID

The FCU_PID component is a regulator with proportional, integral and derivative actions.

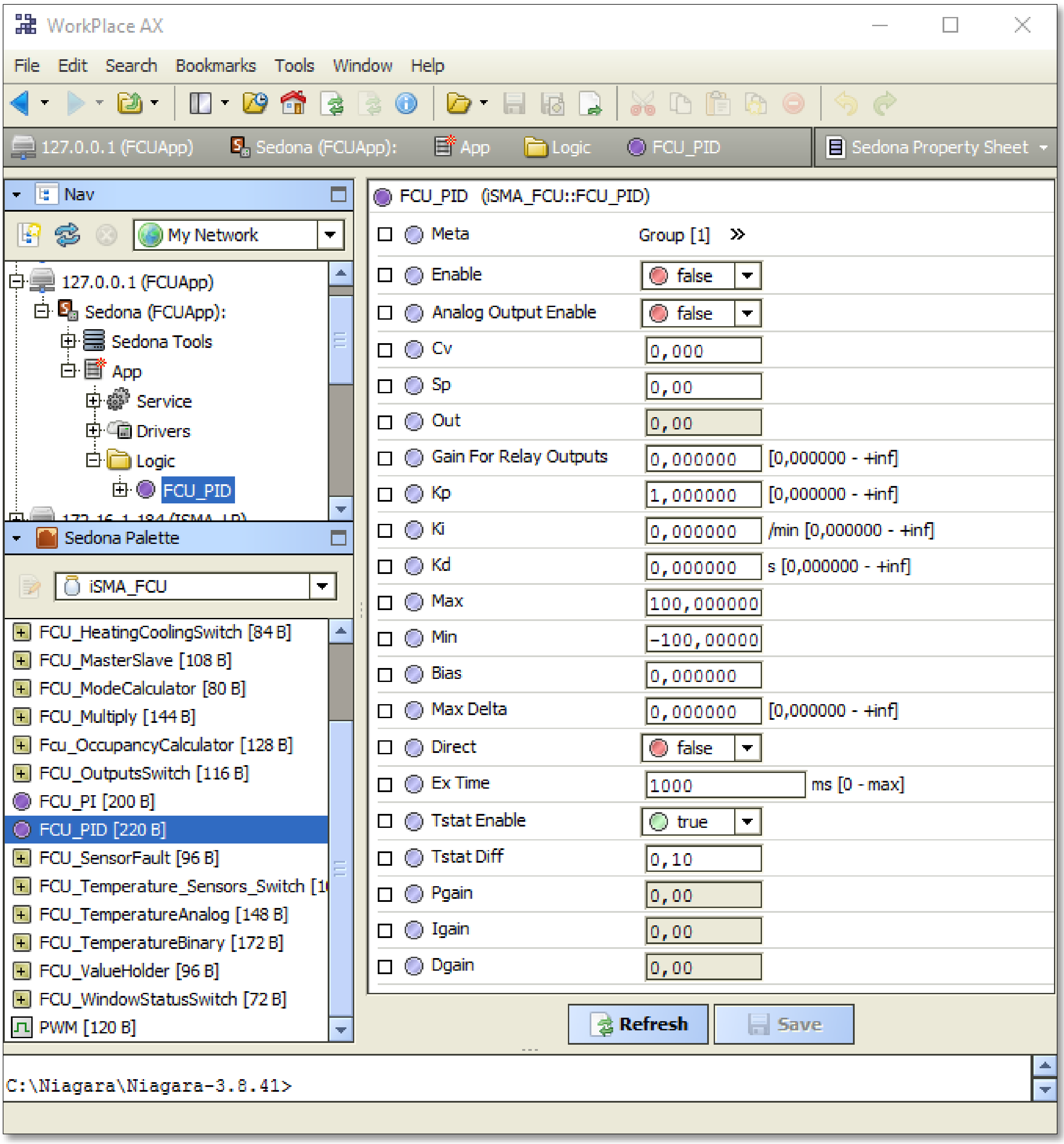


Figure 79 FCU_PID component

The component has the following slots:

- Enable – Loop enable slot. If component is disabled (Enable slot is set to "false"), Out slot is set to 0,
- Analog Output Enable – Binary input slot with information about control type; available states:
    - true – analog control – the component uses available actions:
  - proportional with gain equal to value of Kp slot,

        - integral, according to constant set in Ki slot,
        - derivative, according to constant set in Kd slot,
    - false – binary control – component works using only proportional gain. In this type of control, proportional gain is set in Gain For Relay Outputs slot,
- Cv – Numeric input slot with controlled value,
- Sp – Numeric input with setpoint of controlled value,
- Out – Output slot of component,
- Gain For Relay Outputs – Numeric input slot with gain used to calculate Out slot, when the component controls binary outputs (Analog Output Enable slot is set to "false"),
- Kp – Numeric input with value of proportional gain constant,
- Ki – Numeric input with value of integral constant. Setting to 0 disables this integral action,
- Kd – Numeric input with value of derivate constant. Setting to 0 disables this derivate action,
- Max – Numeric input slot with maximal value of output of component,
- Min – Numeric input slot with minimal value of output of component,
- Bias – Numeric input with bias. This value is added to Out slot, if Ti slot is set to 0.
- Max Delta – Numeric input slot with the max amount out can change in exTime. Setting to 0 disables this function,
- Direct – Binary input slot to set acting process; available values:
    - True – direct acting process,
    - False – reverse acting process,
- Ex Time – Numeric input slot with period of loop execution,
- Tstat Enable – Binary input slot to enable/disable Tstat function,
- Tstat Diff – Numeric input slot with differential value for Tstat function,

Note: Tstat function allows to reset integral action when absolute value of difference between Cv slot and Sp slot is lower than value from Tstat Diff slot. In this case, sum of error (used for calculating of integral gain) is set to 0.

- Pgain – Numeric output with value of output signal, which was calculated by proportional action,
- Igain – Numeric output with value of output signal, which was calculated by integral action,
- Dgain – Numeric output with value of output signal, which was calculated by derivate action.

14.11 FCU_OccupancyCalculator

The FCU_OccupancyCalculator component manages Occupancy Status, according to the occupancy modes and states of presence sensor, card holders etc.

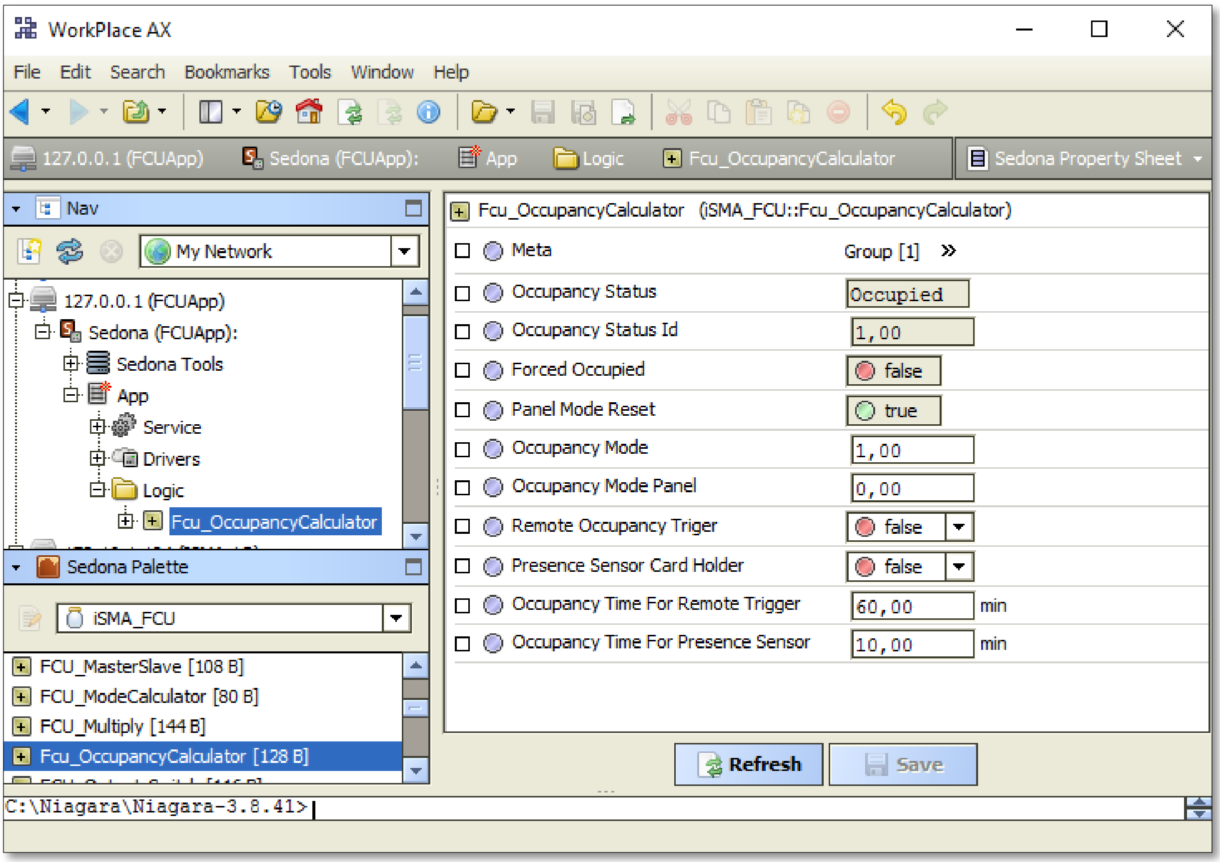


Figure 80 Fcu_OccupancyCalculator component

The component has the following slots:

- Occupancy Status – slot displays current Occupancy Status; available values:
    - 0 – Unoccupied
    - 1 – Occupied
    - 2 – Standby
- Occupancy Status Id – output numeric value corresponding to Occupancy Status slot

- Forced Occupied – binary output slot with information on occupancy status source. Available values:
    - True – Occupied mode has been forced. It means, that component works in Occupied mode (Occupancy Status Id slot is set to 1), but this mode has not been calculated according to Occupancy Mode. Occupied mode can be forced by Panel (value of Occupancy Mode Panel slot), presence sensor or card holder (detecting rising edge on Presence Sensor Card Holder slot) or by remote trigger (detecting rising edge on Remote Occupancy slot).
    - False – Occupied mode has not been forced.
- Panel Mode Reset – binary output, which can be used to reset value from component connected to Occupancy Mode Panel after switching Occupancy Mode off, which has been forced by Panel. Panel Mode Reset slot has value “false”. It is set to “true” for one application cycle, when time of Occupancy mode forced by panel has passed.
- Occupancy Mode – value corresponding to Occupancy mode set from upper level system; available values:
    - 0 – Unoccupied
    - 1 – Occupied
    - 2 – Standby
- Occupancy Mode Panel – value corresponding to Occupancy mode set from external source (for example dedicated to iSMA-B-FCU device panel); available values:
    - 0 – Unoccupied
    - 1 – Occupied

Note: Value of Occupancy Mode Panel slot allows to force Occupied Mode from the panel. If Occupancy Mode slot is set to 0 or 2 (Unoccupied or Standby), setting value of Occupancy Mode Panel slot to 1 (Occupied) will force Occupied mode for time defined in Occupancy Time For Remote Trigger slot. During this time Forced Occupied slot has value “true”. After that time, value of Panel Mode Reset slot is set to “true” for one cycle of application and component goes back to previous Occupancy Mode. Occupied mode forced in this way can be also cancelled by setting back Occupancy Mode Panel slot to 0 (Unoccupied).

- Remote Occupancy Trigger – binary input (recommended for remote occupancy trigger), allows to force Occupied mode. If Occupancy Mode slot is set to 0 or 2 (Unoccupied or Standby), rising edge detected on this slot will force Occupied mode for time defined in Occupancy Time For Remote Trigger slot. Counting of this time starts when falling edge is detected on the described slot. During this time Forced Occupied slot has value “true”. After that time, the component goes back to previous Occupancy Mode. Occupied mode forced in this way cannot be cancelled.
- Presence Sensor Card Holder – binary input (recommended for presence sensor or card holder), allows to force Occupied mode. If Occupancy Mode slot is set to 0 or 2 (Unoccupied or Standby), rising edge detected on this slot will force Occupied mode for time defined in Occupancy Time For Presence Sensor slot. Counting of this time starts, when falling edge is detected on the described slot. During this time Forced Occupied slot has value “true”. After that time, the component goes back to previous Occupancy Mode. Occupied mode forced in this way cannot be cancelled.
- Occupancy Time For Remote Trigger – Time of Occupancy mode forced by Panel or Remote Trigger.
- Occupancy Time For Presence Sensor – Time of Occupancy mode forced by presence sensor or card holder.

14.12FCU_OutputsSwitch

The FCU_OutputSwitch component allows to manage outputs for temperature and fan control, according to FCU configuration (2 or 4 pipe system, analog or binary temperature control etc.).

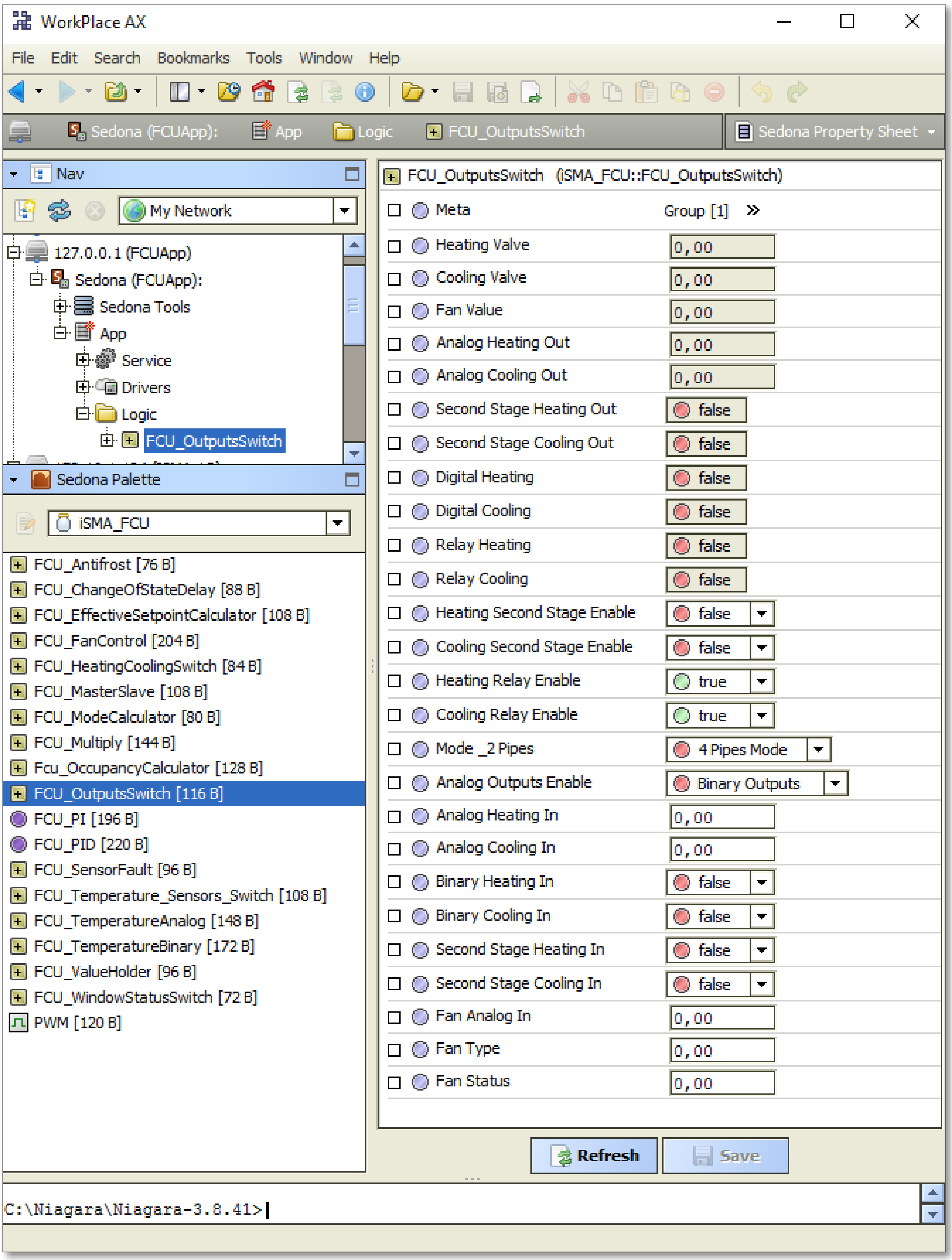


Figure 81 FCU_OutputsSwitch compomnent

The component has the following slots:

- Heating Valve – Numeric output slot, which displays status of heating valve. Value can be displayed in two ways:
    - Displaying 0 or 1 – for temperature binary output (if Analog Output Enable slot is set to “false”); 0 – valve closed, 1– valve open
    - Displaying value of Analog Heating In slot – for temperature analog output (if Analog Output Enable slot is set to “true”).
- Cooling Valve – Numeric output slot, which displays status of cooling valve. Value can be displayed in two ways:
    - Displaying 0 or 1 – for temperature binary output (if Analog Output Enable slot is set to “false”); 0 – valve closed, 1– valve open
    - Displaying value of Analog Cooling In slot – for temperature analog output (if Analog Output Enable slot is set to “true”).
    -
- Fan Value – Numeric output slot, which displays status of the fan. Value can be displayed in two ways:
    - Displaying 0, 1, 2 or 3 – for fan binary outputs (if Fan Type slot is not equal to 0), according to value of Fan Status slot; available values:
        - 0 – Off (if Fan Status slot is equal 0),
        - 1 – Speed 1 (if Fan Status slot is equal 1 or 4),
        - 2 – Speed 2 (if Fan Status slot is equal 2 or 5),
        - 3 – Speed 3 (if Fan Status slot is equal 3 or 6).
    - Displaying value of Fan Analog In slot – for fan analog output (if Fan Type slot is equal to 0).
- Analog Heating Out – numeric output slot for analog heating valve.
    - For 2-pipe system (slot Mode _2 Pipes is set to “true”), when slot Analog Outputs Enable is set to “true”, Analog Heating Out slot displays value of Analog Heating In slot or Analog Cooling In slot, depending on which slot has value greater than 0 (this value is set to Analog Heating Out slot).
    - For 4-pipe system (slot Mode _2 Pipes is set to “false”), slot Analog Heating Out can only display value of Analog Heating In.
    - If slot Analog Outputs Enable is set to “false” (the component uses only binary outputs), slot Analog Heating Out is set to 0.

- Analog Cooling Out – numeric output slot for analog cooling valve.
  - For 4-pipe system (slot Mode _2 Pipes is set to “false”), slot Analog Cooling Out displays value of slot Analog Cooling In.
  - If slot Analog Outputs Enable is set to “false” (the component uses only binary outputs) or slot Mode _2 Pipes is set to “false” (for 4-pipe system), slot Analog Cooling Out is set to 0.
- Second Stage Heating Out – binary output slot for second stage heating. Slot displays value from slot Second Stage Heating In.

Note: If Heating Second Stage Enable slot is set to “false” or Heating Relay Enable slot is set to “false”, Second Stage Heating Out slot cannot be set to “true”.

- Second Stage Cooling Out – binary output slot for second stage cooling. Slot displays value from slot Second Stage Cooling In.

Note: If Cooling Second Stage Enable slot is set to “false” or Cooling Relay Enable slot is set to “false”, Second Stage Cooling Out slot cannot be set to “true”.

- Digital Heating – binary output slot for digital heating (recommended to service heating valve switched on/off by Triac).
  - For 2-pipe system (slot Mode _2 Pipes is set to “true”), when slot Analog Outputs Enable is set to “false”, Digital Heating slot displays value of Binary Heating In slot or Binary Cooling In slot, depending on which slot has “true” value (this value is set to Digital Heating slot).
  - For 4-pipe system (slot Mode _2 Pipes is set to “false”), slot Digital Heating can only display value of Binary Heating In.
  - If slot Analog Outputs Enable is set to “true” (the component uses only analog outputs), slot Digital Heating is set to “false”.
- Digital Cooling – binary output slot for digital cooling (recommended to service cooling valve switched on/off by Triac).
  - For 4-pipe system (slot Mode _2 Pipes is set to “false”), slot Digital Cooling displays value of slot Binary Cooling In.
  - If slot Analog Outputs Enable is set to “true” (the component uses only analog outputs) or slot Mode _2 Pipes is set to “false” (for 4-pipe system), slot Digital Cooling is set to “false”.

- Relay Heating – binary output slot for digital heating in 1st or 2nd stage (recommended to service heating valve switched by relay output or electrical heaters).
  - If slot Heating Relay Enable is set to "true" and slot Heating Second Stage Enable is set to "false" (heating in 1st stage only), the value from slot Binary Heating In is set to Relay Heating slot.
  - If slot Heating Relay Enable is set to "true", and slot Heating Second Stage Enable is set to "true" (heating in 1st and 2nd stage), the value from slot second Stage Heating In is set to Relay Heating slot.
  - If slot Heating Relay Enable is set to "false" (heating relay is disabled), value "false" is set to Relay Heating slot.
- Relay Cooling – binary output slot for digital cooling in 1st or 2nd stage (recommended to service cooling valve switched by relay output or electrical coolers).
  - If slot Cooling Relay Enable is set to "true" and slot Cooling Second Stage Enable is set to "false" (cooling in 1st stage only), the value from slot Binary Cooling In is set to Relay Cooling slot.
  - If slot Cooling Relay Enable is set to "true", and slot Cooling Second Stage Enable is set to "true" (cooling in 1st and 2nd stage), the value from slot second Stage Cooling In is set to Relay Cooling slot.
  - If slot Cooling Relay Enable is set to "false" (cooling relay is disabled), value "false" is set to Relay Cooling slot.
- Heating Second Stage Enable – binary input allows to enable or disable 2nd stage heating; true – enabled, false – disabled,
- Cooling Second Stage Enable – binary input allows to enable or disable 2nd stage cooling; true – enabled, false – disabled,
- Heating Relay Enable – binary input allows to enable or disable relay for heating; true – enabled, false – disabled,
- Cooling Relay Enable – binary input allows to enable or disable relay for cooling; true – enabled, false – disabled,
- Mode _2 Pipes – binary input allows to switch between 2-pipe system and 4-pipe system; "true" – 2-pipe system, "false" – 4-pipe system,
- Analog Outputs Enable – binary input allows to switch between analog or binary control of temperature outputs; "true" – Analog Outputs, "false" – Binary Outputs,
- Analog Heating In – Numeric input of analog value for heating,
- Analog Cooling In – Numeric input of analog value for cooling,
- Binary Heating In – Input of binary value for 1st stage heating,
- Binary Cooling In – Input of binary value for 1st stage cooling,
- Second Stage Heating In – Input of binary value for 2nd stage heating,
- Second Stage Cooling In – Input of binary value for 2nd stage cooling,
- Fan Analog In – Numeric input of analog value for fan
- Fan Type – Numeric input corresponding to type of fan; available values:
  - 0 – fan with analog output
  - Other values – fan with binary outputs.
- Fan Status – numeric input slot of current fan speed; available values:
  - 0 – Off,
  - 1 or 4 – Speed 1,
  - 2 or 5 – Speed 2,

- 3 or 6 – Speed 3.

14.13FCU_SensorFault

The FCU_SensorFault component allows for detection of sensor fault according to temperature value.

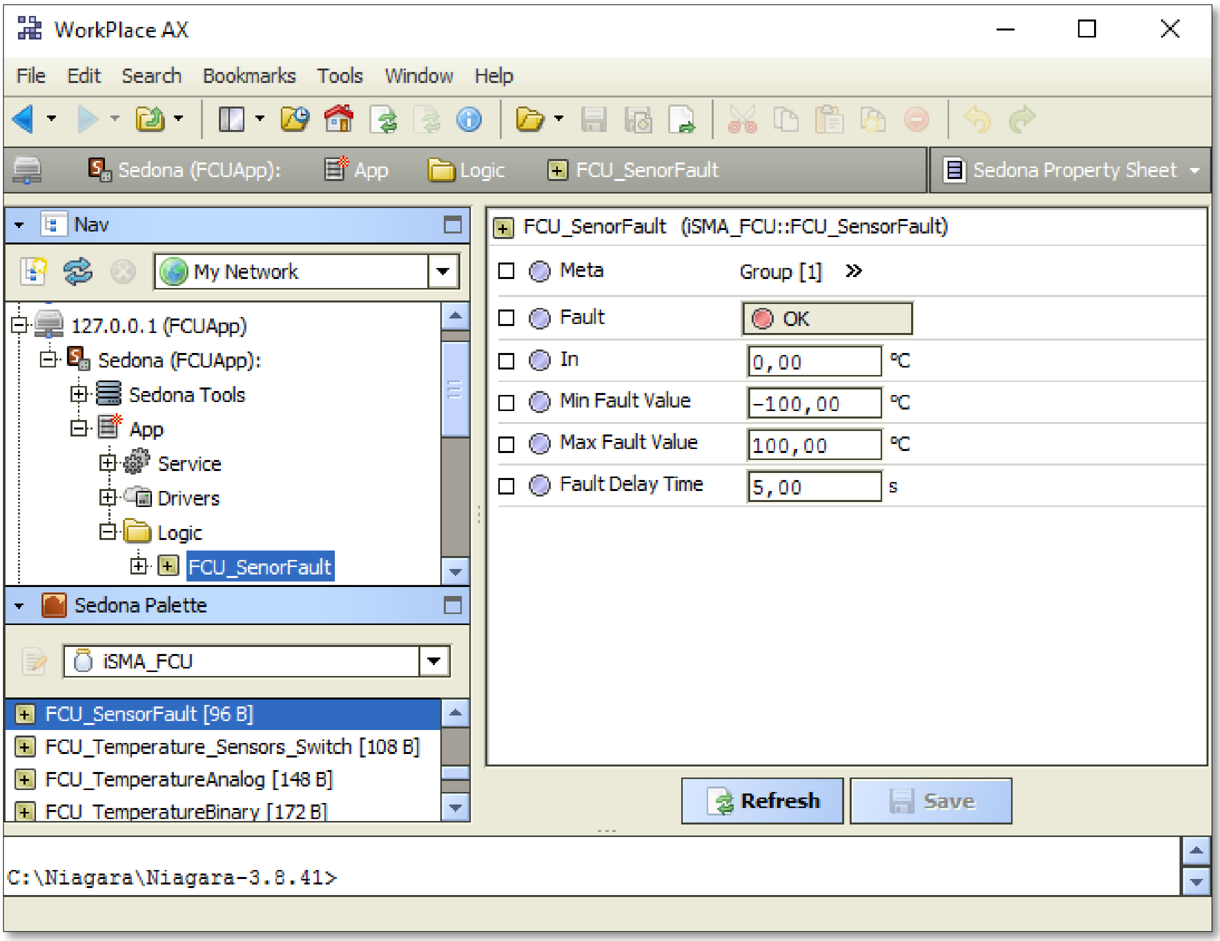


Figure 82 FCU_SensorFault component

The component has the following slots:

- Fault – Binary output switch. If temperature from sensor (set to In slot) is lower than value of Min Fault Value or higher than value of Max Fault Value slot for time longer than set to Fault Delay Time slot, Fault slot is set to “true” (Sensor Fault). In another case, Fault slot is set to “false” (OK).
- In – Input for temperature value from sensor,
- Min Fault Value – Value of minimum acceptable temperature,
- Max Fault Value – Value of maximum acceptable temperature,
- Fault Delay Time – Delay time for detecting sensor fault.

14.14 FCU_Temperature_Sensors_Switch

The FCU_Temperature_Sensors_Switch component allows for switching temperature sensors, according to selected source.

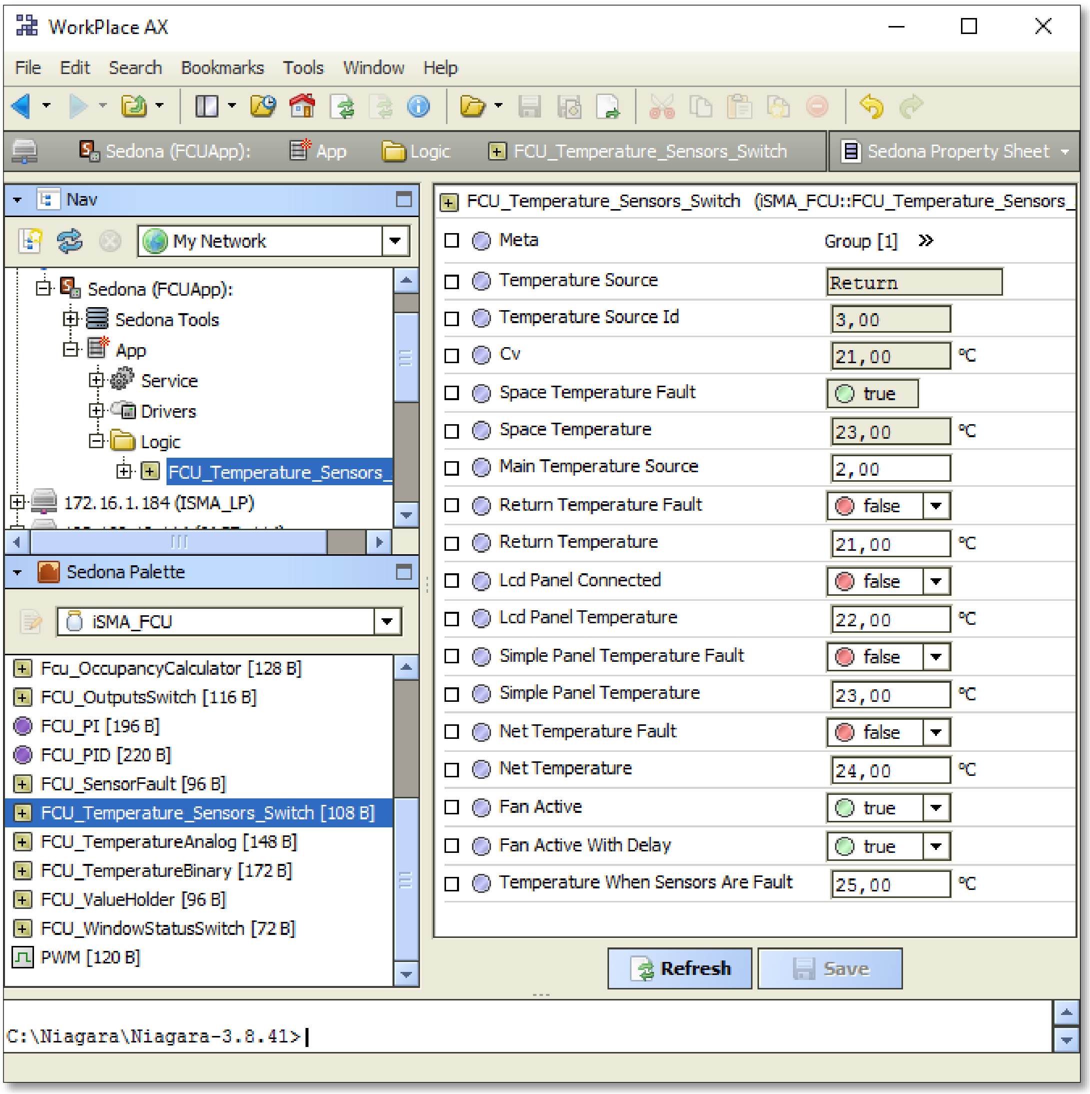


Figure 83 FCU_Temperature_Sensors_Switch component

The component has the following slots:

- Temperature Source – Slot displays information about source of temperature set to Cv slot; available values:
  - 0 – Sensor Fault (fault of sensor selected by Main Temperature Source slot)
  - 1 – LCD Panel
  - 2 – Simple Panel
  - 3 – Return
  - 4 – Net Temperature
- Temperature Source Id – Numeric output with value corresponding to Temperature Source slot,
- Cv – Output value of temperature, switching according to value of Main Temperature slot.
  - If Main Temperature Source slot is set to 0 (LCD Panel) and:

- If panel is connected (slot Lcd Panel Connected is set to “true”), value from Lcd Panel Temperature slot is set to Cv slot.
- If panel is not connected (slot Lcd Panel Connected is set to “false”), value from Temperature When Sensors Are Fault slot is set to Cv slot.
- For other sources:
  - If fault of sensor selected by value of Main Temperature Source value has not been detected (corresponding to fault slot is set to “false”), value from slot corresponding to selected temperature is set to Cv slot.
  - If fault of sensor selected by value of Main Temperature Source value has not been detected, value from slot Temperature When Sensors Are Fault slot is set to Cv slot.
- If value from Return Temperature is set to Cv slot, it is possible to switch temperature from Return to Space. The Space Temperature will be downloaded when fan is off. There could be also delay for switching to Return Temperature after fan start to blow the duct. Fan Active and Fan Active With Delay slots are used for this function. If one or both slots have false states, then value from Space Temperature slot is set to Cv slot (instead of value from Return Temperature slot). If both slots have the same “true” states, then value from Return Temperature slot is set to Cv slot.

Note: For proper operation of this function external component is required, which delays value informing about fan status. Value without delay has to be connected to Fan Active slot and value with delay has to be connected to Fan Active With Delay slot.

Note: If Return Temperature Fault slot is set to “true” (sensor fault), it downloads value from Temperature When Sensors Are Fault (instead Return Temperature). If Space Temperature Fault slot is set to “true” (sensor fault), function is inactive.

- Space Temperature Fault – Status of space temperature; true – fault, false – no fault.
- Space Temperature – Output value of space temperature. It is calculated in the way described below:
  - Lcd Panel Temperature has the highest priority. If Panel is connected (Lcd Panel Connected slot is set to “true”), value from Lcd Panel Temperature slot is set to Space Temperature slot.
  - If Panel is disconnected (Lcd Panel Connected slot is set to “false”) and there is no fault of simple panel temperature sensor (Simple Panel Temperature Fault is set to “false”), value from Simple Panel Temperature slot is set to Space Temperature slot.
  - If Lcd Panel is disconnected, Simple Panel sensor is faultz and there is no fault of Net Temperature, value from Net Temperature slot is set to Space Temperature slot.
  - In case when none of above mentioned conditions is meet, and there is no fault of temperature sensor which is used to calculate the Cv value (value of Temperature Source Id slot is higher than 0), value from Cv slot is set to Space Temperature slot.

Note: In all cases described above, slot Space Temperature Fault is set to “false” (there is no fault). In other cases, this slot is set to “true”.

- Main Temperature Source – Input numeric slot, which allows to select source of temperature; available values:

- 0 – LCD Panel Temperature
- 1 – Simple Panel Temperature
- 2 – Return Temperature
- 3 – Net Temperature

- Return Temperature Fault – Binary input slot with status of Return Temperature sensor; true – sensor fault, false – sensor's operation is correct,
- Return Temperature – Numeric input with temperature from Return Temperature sensor,
- Lcd Panel Connected – Binary input slot with status of LCD Panel; true – Panel connected, false – Panel disconnected,
- Lcd Panel Temperature – Numeric input with temperature from LCD Panel Temperature sensor,
- Simple Panel Temperature Fault – Binary input slot with status of Simple Panel Temperature sensor; true – sensor fault, false – sensor's operation is correct,
- Simple Panel Temperature – Numeric input with temperature from Simply Panel Temperature sensor,
- Net Temperature Fault – Binary input slot with status of Net Temperature sensor; true – sensor fault, false – sensor's operation is correct,
- Net Temperature – Numeric input with Net Temperature value,
- Fan Active – Binary input slot with status of fan; true – fan switches on, false – fan switches off,
- Fan Active With Delay – Binary input slot with status of fan (with external delay); true – fan switches on, false – fan switches off,
- Temperature When Sensors Are Fault – Value of temperature, which will be set to Cv slot, if sensor selected by value of MainTemperature Source is faulty.

14.15FCU_TemperatureAnalog

The FCU_TemperatureAnalog component allows to calculate numeric values for Heating Analog Output and Cooling Analog Output slots, according to the external analog Control Value from range <-100%, 100%> (for example, from analog PID regulator). The component is dedicated to control valve actuators with Analog Inputs (or actuators which allow can be controlled by Triac Outputs, with PWM). Negative values of Control Value are used for calculating Heating Analog Out Slot and positive values for calculating Cooling Analog Out slot.

The FCU_TemperatureAnalog component can work in two temperature control modes:

- One stage mode – Only Analogs Outputs (Heating Analog Output and Cooling Analog Output slots) are calculated, according to Control Value,
- Two stages mode – Analog Outputs are used for 1st stage, and dedicated binary outputs (Heating Second Stage Binary Output and Cooling Second Stage Binary Output) are used for 2nd stage.

Note: Operation in two stages mode can be selected by setting slots Heating Second Stage Enable and/or Cooling Second Stage Enable to "true". If one of these slots is set to "false", the component will operate in one stage mode for corresponding temperature mode (Heating or Cooling).

For proper operation, the component has to be enabled (Temperature Analog Enable slot set to “true”) and Fan Active slot has to be set to “true”. If second condition is not met, the component is enabled, but main outputs are blocked – values of Heating Analog Output and Cooling Analog Output slots are set to 0 and Heating Second Stage Binary Output and Cooling Second Stage Binary Output slots are set to “false”.

The values of Heating Analog Output and Cooling Analog Output slots for one-stage mode control are calculated as shown in the figures below:

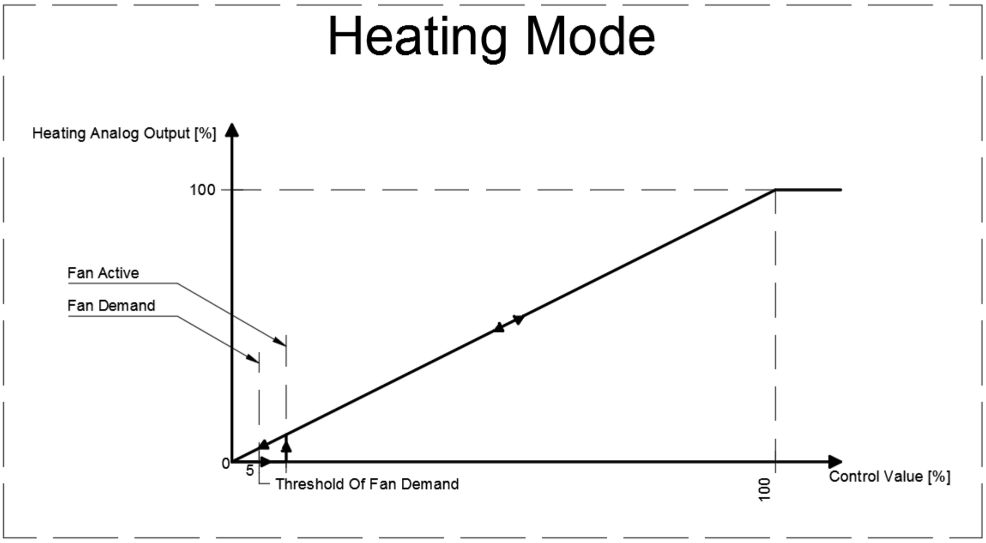


Figure 84 Analog control of temperature for 1st stage only – Heating mode

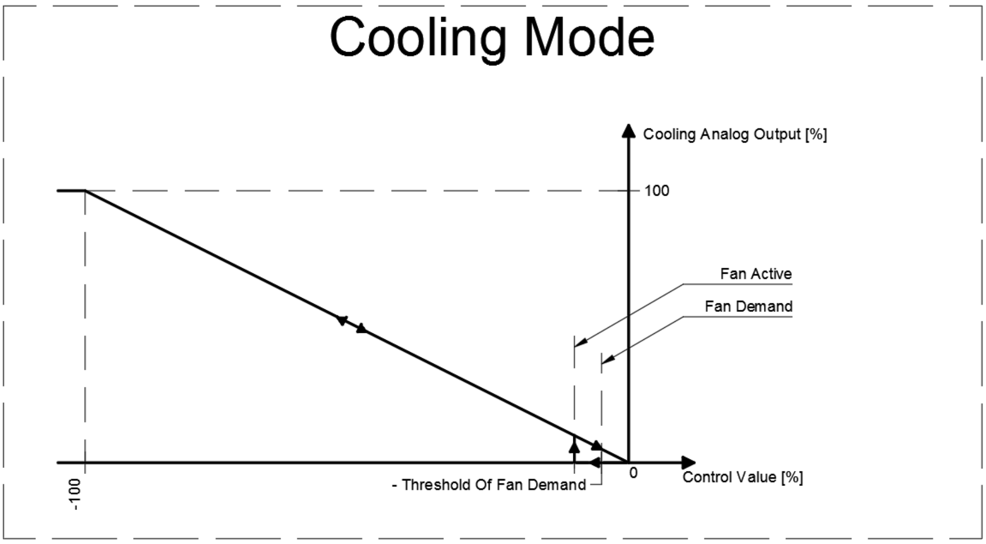


Figure 85 Analog control of temperature for 1st stage only – Cooling mode

In this mode, the slots for 2nd stage are not used (Heating Second Stage Binary Output and Cooling Second Stage Binary Output slots are still set to “false”).

The values of Heating Analog Output and Cooling Analog Output slots for 1st stage and the values of Heating Second Stage Binary Output and Cooling Second Stage Binary Output slots (for 2nd stage) for two-stage mode are calculated as shown in the figures below:

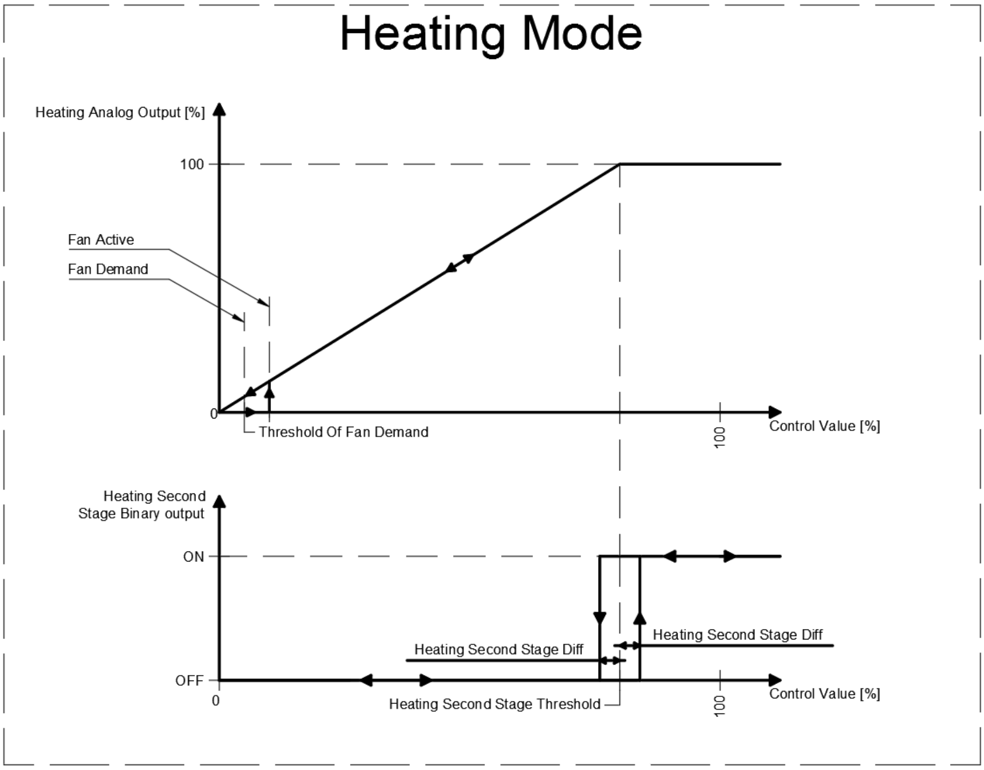


Figure 86 Analog control of temperature for 1st and 2nd stage - Heating mode

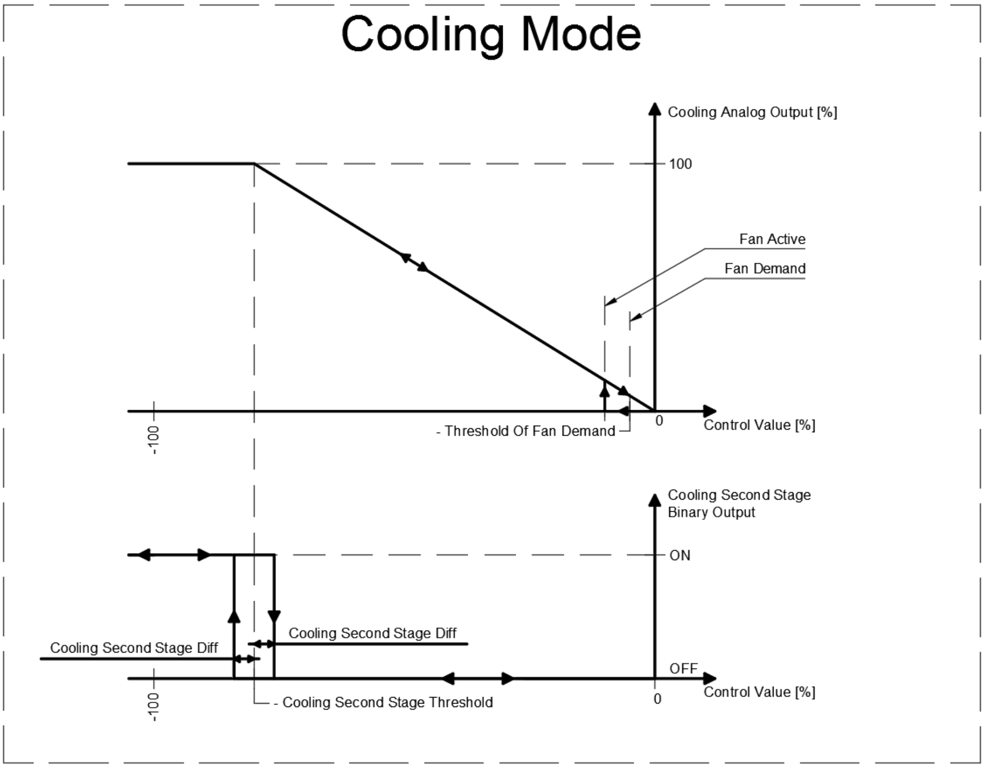


Figure 87 Analog control of temperature for 1st and 2nd stage – Cooling mode

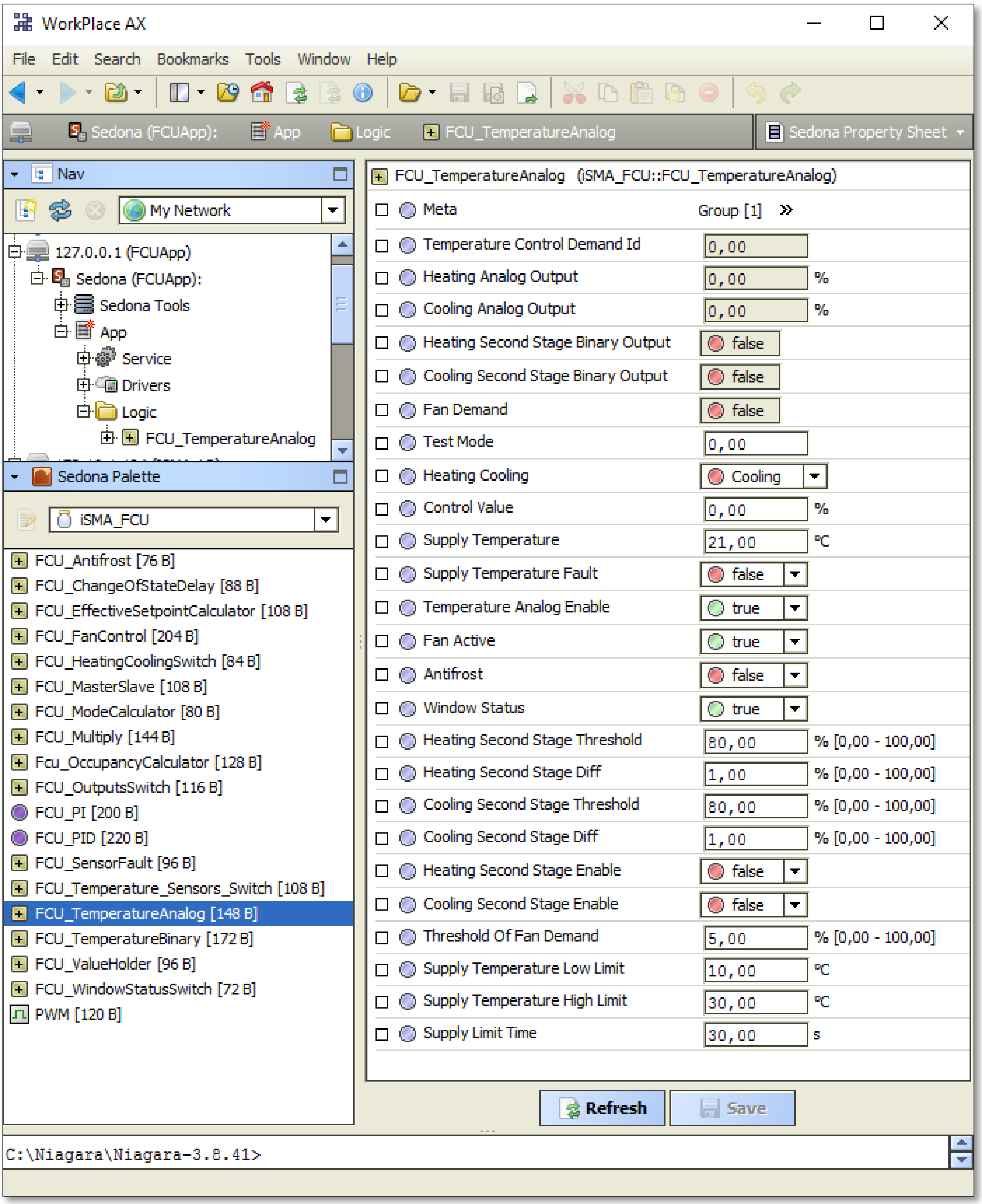


Figure 88 FCU_TemperatureAnalog component

The component has the following slots;

- Temperature Control Demand Id – Numeric output slot with value corresponding to the current component temperature demand; available values:
    - 1 – Heating demand
    - 2 – Cooling demand
- Heating Analog Output – Numeric output slot with value of heating demand level, expressed as percentage
- Cooling Analog Output – Numeric output slot with value of cooling demand level, expressed as percentage,
- Heating Second Stage Binary Output – Binary output slot with state of heating in 2nd stage,
- Cooling Second Stage Binary Output – Binary output slot with state of cooling in 2nd stage,
- Fan Demand – Binary output slot of fan demand. Fan Demand slot is set to “true”, when absolute value from Control Value slot exceeds value set in Fan Demand Threshold slot. In other cases, slot Fan Demand is set to “false”.
- Test Mode – Numeric input slot which corresponds to predefined Test modes; available values:
    - 0 – None – The component works using main algorithm,
    - 1 – Full Heating:
        - Fan Demand slot is set to “true”,:
        - Slot Heating Analog Output is set to 100, Slot Heating Second Stage Binary Output is set to “true” (only if Heating Second Stage Enable is set to “true”),
    - 2 – Full Cooling:
        - Fan Demand slot is set to “true”,
        - Slot Cooling Analog Output is set to 100;
        - Slot Cooling Second Stage Binary Output is set to “true” (only if Cooling Second Stage Enable is set to “true”),

Note: Before starting Full Heating or Full Cooling mode, value of Fan Active slot has to be set to “true”.

- Heating Cooling – Binary input slot with value of current temperature mode; available values:
    - True – Heating mode
    - False – Cooling mode

Note: If the component operates in one of the above modes, output for the other mode is blocked. For example, if the component works in Heating mode, Cooling Analog Output is still set to 0 and Heating Second Stage Binary Output slot is still set to “false”.

- Control Value – Numeric input slot with value from external component (for example from PID regulator), accordingly the values for output slots of component are calculated; range: <-100%, 100%>. Negative values of Control Value are used for calculating open level of heating valve and positive values for calculating open level of cooling valve.
- Supply Temperature – Numeric input slot with value of supply temperature,
- Supply Temperature Fault – Binary input slot with information about fault of supply

temperature,
- Temperature Analog Enable – Binary input slot, which allows to enable or disable FCU_TemperatureAnalog component. If component is disabled, all outputs are set to 0 (for numeric outputs) or to false (for binary outputs); true – component enable, false – component disable,
- Fan Active – Binary input slot, which is used to inform FCU_TemperatureAnalog component that fan is switched on. If slot is set to "false", analog outputs (Heating Analog Output and Cooling Analog Output slots) are set to 0 and binary output slots (Heating Second Stage Binary Output and Cooling Second Stage Binary Output slot) are set to "false". These slots can be set to other values (calculated by main algorithm) only when Fan Active slot is set to "true".
- Antifrost – Binary input slot to switch on Antifrost mode:
  - true – Antifrost mode enabled
    - Fan Demand slot is set to "true",
    - Heating Analog Output slot is set to 100,
    - Heating Second Stage Binary output slot is set to "true" (only if Heating Second Stage Enable slot is set to "true"),

    Note: Antifrost mode has higher priority than main algorithm, but can be overridden by Test mode.
  - false – Antifrost mode disabled, normal work of component
- Window Status – Binary input slot to switch on Window Is Open mode:
  - true – Window Is Open mode disabled
  - false – Window Is Open mode enabled – the component operates in saving energy mode, analog outputs are set to 0 and binary output for second stage is set to "false",

  Note: Window Is Open mode can be overridden only by Antifrost mode or Test mode.
- Heating Second Stage Threshold – Numeric input slot with threshold of Control Value, above which (with the hysteresis) heating in 2nd stage is switched on,
- Heating Second Stage Diff – Differential for hysteresis of switching on/off heating in 2nd stage,
- Cooling Second Stage Threshold – Numeric input slot with threshold of Control Value, above which (with the hysteresis) cooling in 2nd stage is switched on,
- Cooling Second Stage Diff – Differential for hysteresis of switching on/off cooling in 2nd stage,
- Heating Second Stage Enable – Binary input slot, which allows to enable/disable heating in 2nd stage; available values: true – enabled, false – disabled,
- Cooling Second Stage Enable – Binary input slot, which allows to enable/disable cooling in 2nd stage; available values: true – enabled, false – disabled,
- Threshold Of Fan Demand – Numeric input slot with threshold of Control Value, above which slot Fan Demand is set to "true",
- Supply Temperature Low Limit – Minimum acceptable value of supply temperature. This value is used in Supply Air Temperature Limitation function,
- Supply Temperature High Limit – Maximum acceptable value of supply temperature. This value is used in Supply Air Temperature Limitation function,
- Supply Limit Time – Delay time for activation of function Supply Air Temperature

Limitation.

Supply Air Temperature Limitation

For room user comfort, supply air can have temperature limitation. This function is available only when temperature from supply air sensor is connected and works correct. Supply air temperature can have upper limitation defined by Supply Temperature High Limit slot and lower limitation defined by Supply Temperature Low Limit slot. The range between Supply Temperature Low Limit and Supply Temperature High Limit values is called “comfort” range.

- Supply Air Temperature limitation in 1st stage analog control

  In analog control, when supply air temperature approaches by 1°C to the "comfort" range limit, the FCU_TemperatureAnalog component set in Supply Limit Time slot delay time will start countdown. After this time, if supply air temperature value is still approaching by 1°C to the "comfort" range limit, the component will start in-built algorithm which will reduce air temperature (if temperature value is close to or above Supply Temperature High Limit) or will increase air temperature (if temperature value is close to or below Supply Temperature Low Limit). When the supply air temperature value returns to “comfort” range ±1°C, the component will reset delay counter and return to normal operation.

- Supply Air Temperature limitation in 2nd stage analog control

  In analog control, when supply air temperature approaches by 1°C to the "comfort" range limit, the FCU_TemperatureAnalog component will disable 2nd stage and start counting delay time set in Supply Limit Time slot. After this time, if supply air temperature value is still approaching by 1°C to the "comfort" range limit, the component will start in-built algorithm which will reduce air temperature (if temperature value is close to or above Supply Temperature High Limit) or will increase air temperature (if temperature value is close to or below Supply Temperature Low Limit). When the supply air temperature value returns to “comfort” range ±1°C, the component will reset delay counter, enable 2nd stage and return to normal operation.

14.16FCU_TemperatureBinary

The FCU_TemperatureBinary component allows to calculate binary values for Heating Binary Output and Cooling Binary Output slots, according to difference between Setpoint and Cv.

The FCU_TemperatureBinary component can work in two temperature control modes:

- One-stage mode – Only Heating Binary Output and Cooling Binary Output slots are switched on/off,
- Two-stage mode – Heating Binary Output and Cooling Binary Output slots are used for 1st stage, and dedicated binary outputs (Heating Second Stage Binary Output and Cooling Second Stage Binary Output) are used for 2nd stage.

Note: Operate in two-stage mode can be select by setting slots Heating Second Stage Enable and/or Cooling Second Stage Enable to “true”. If one of these slots is set to “false”, the component will operate in one-stage mode for corresponding temperature mode (Heating or Cooling).

For proper operation, the component has to be enabled (Temperature Binary Enable slot set to "true") and Fan Active slot has to be set to "true". If second condition is not met, the component is enabled, but main outputs are blocked – values of Heating Binary Output, Cooling Binary Output, Heating Second Stage Binary Output and Cooling Second Stage Binary Output slots are set to "false".

The conditions for switching no/off heating binary output slots for one-stage mode (Heating Binary Output and Cooling Binary Output slots) are presented in the figures below:

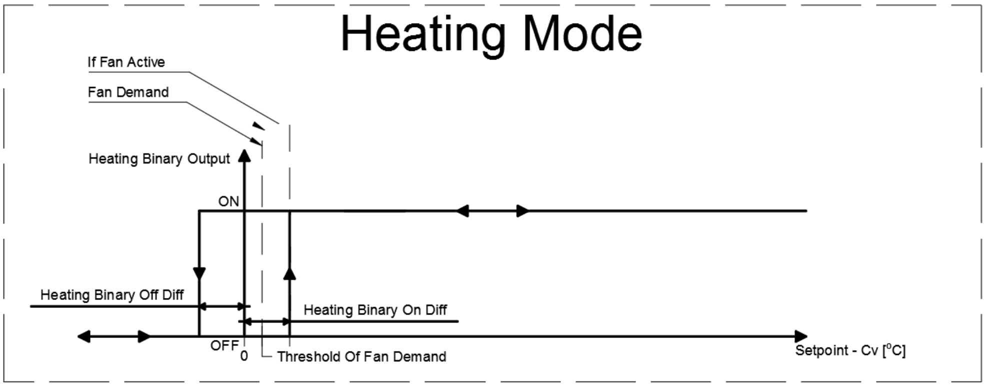


Figure 89 Binary control of temperature for 1st stage only – Heating mode

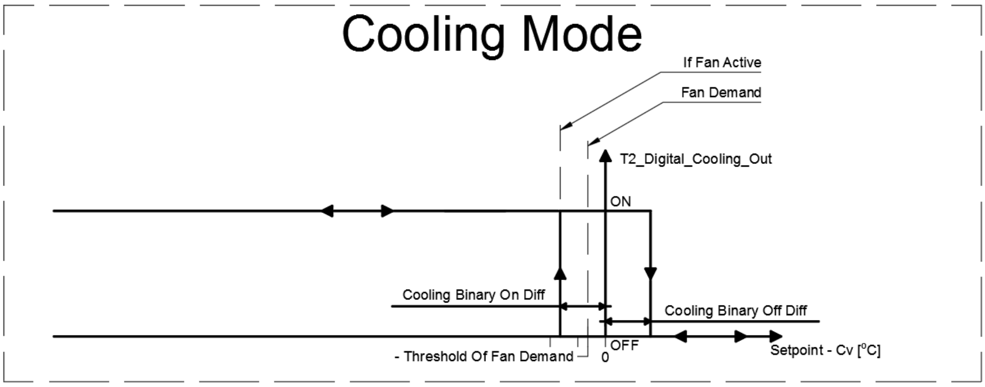


Figure 90 Binary control of temperature for 1st stage only – Cooling mode

The values of Heating Binary Output and Cooling Binary Output slots for 1st stage and the values of Heating Second Stage Binary Output and Cooling Second Stage Binary Output slots (for 2nd stage) for two-stage mode are calculated as shown in the figures below:

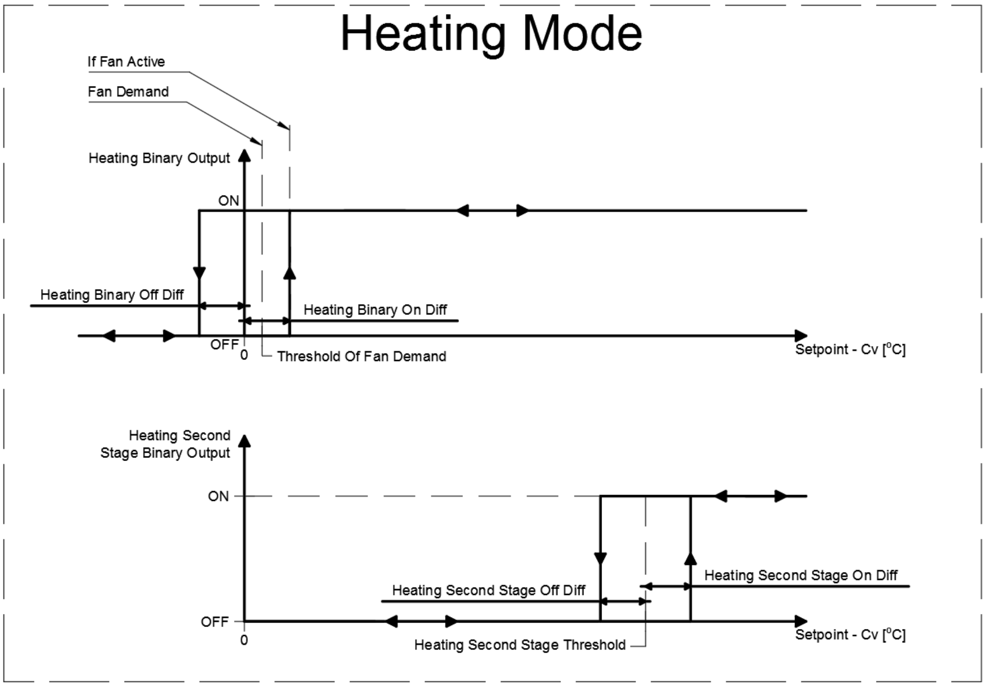


Figure 91 Binary control of temperature for 1st and 2nd stage – Heating mode

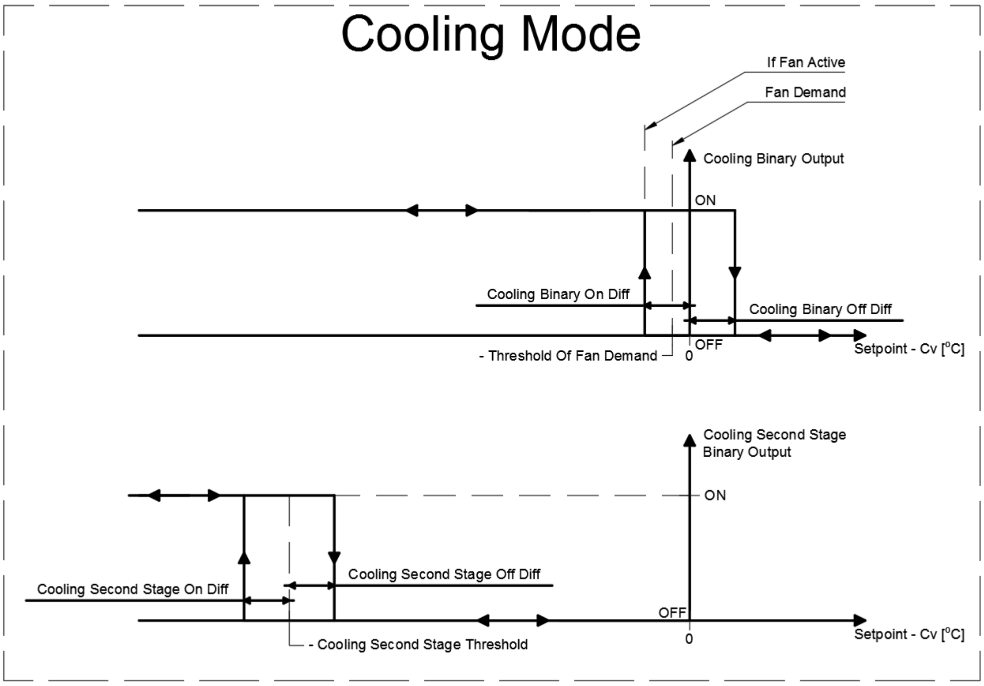


Figure 92 Binary control of temperature for 1st and 2nd stage – Cooling mode

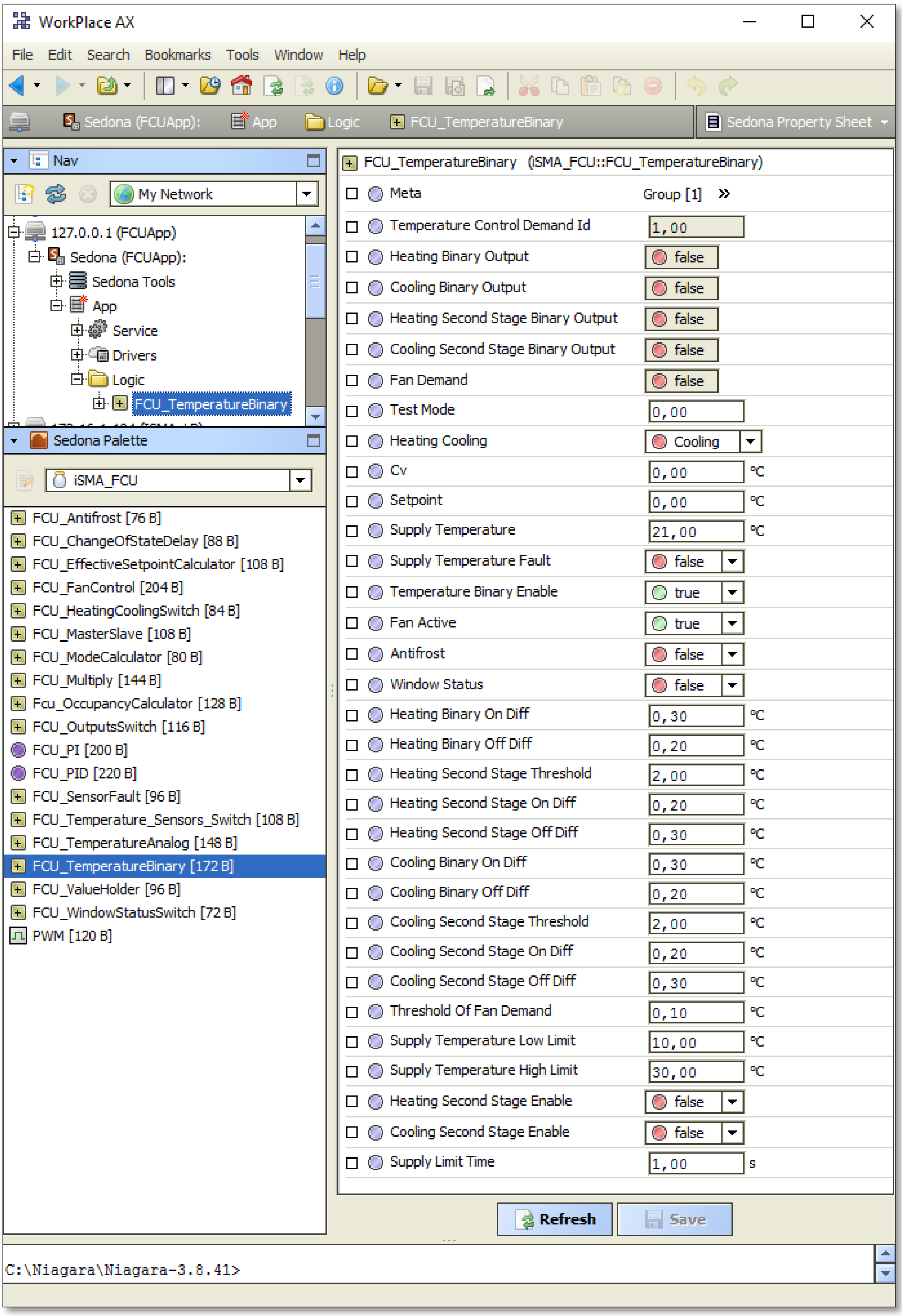


Figure 93 FCU_TemperatureBinary component

The component has the following slots:

- Temperature Control Demand Id – Numeric value corresponding to the current component temperature demand; available values:
    - 0 – Heating demand
    - 1 – Cooling demand
- Heating Binary Output – Binary output slot with state of heating demand,
- Cooling Binary Output – Binary output slot with state of cooling demand,
- Heating Second Stage Binary Output – Binary output slot with state of heating in 2nd stage,
- Cooling Second Stage Binary Output – Binary output slot with state of cooling in 2nd stage,
- Fan Demand – Binary output slot to force fan work. Fan Demand slot is set to “true”, when absolute value from Control Value slot exceeds value set in Fan Demand Threshold slot. In other cases, slot Fan Demand is set to “false”,
- Test Mode – Numeric input which value corresponding to the predefined Test modes; available values:
    - 0 – None – The component works using main algorithm,
    - 1 – Full Heating:
        - Fan Demand slot is set to “true”,
        - Slot Heating Binary Output is set to “true”,
        - Slot Heating Second Stage Binary Output is set to “true” (only if Heating Second Stage Enable is set to “true”).
    - 2 – Full Cooling:
        - Fan Demand slot is set to “true”,
        - Slot Cooling Analog Output is set to “true”,
        - Slot Cooling Second Stage Binary Output is set to “true” (only if Cooling Second Stage Enable is set to “true”).

Note: Before starting Full Heating or Full Cooling mode, value of Fan Active slot has to be set to “true”.

- Heating Cooling – Binary slot with value of current temperature mode; available values:
    - True – Heating mode
    - False – Cooling mode

Note: If the component operates in one of the above modes, output for other modes are blocked. For example, if the component works in Heating mode, Cooling Binary Output and Heating Second Stage Binary Output slots are still set to “false”.

- Cv – Numeric input slot with current value of controlled temperature,
- Setpoint – Numeric input slot with setpoint of controlled temperature,
- Supply Temperature – Numeric input slot with value of supply temperature,
- Supply Temperature Fault – Binary input slot with information about fault of supply temperature,
- Temperature Binary Enable – Binary input slot, which allows to enable or disable the

FCU_TemperatureBinary component. If component is disabled, all outputs are set to "false"; true – component enabled, false – component disabled,

- Fan Active – Binary input slot, which is used to inform FCU_TemperatureBinary component that fan is switched on. If fan is switched off, outputs for heating and cooling valves' actuators (Heating Binary Output and Cooling Binary Output slots) are set to "false" and binary output slots for Heating and Cooling in 2nd stage (Heating Second Stage Binary Output and Cooling Second Stage Binary Output slot) are set to "false". These slots can be set to "true" only when Fan Active slot is set to "true".
- Antifrost Antifrost – Binary input slot to switch on Antifrost mode:
  - true – Antifrost mode enabled
    - Fan Demand slot is set to "true",
    - Heating Binary Output slot is set to "true",
    - Heating Second Stage Binary Output slot is set to "true" (only if Heating Second Stage Enable slot is set to "true"),

  Note: Antifrost mode has higher priority than main algorithm, but can be overridden by Test mode.

  - false – Antifrost mode disabled, normal work of component
- Window Status – Binary input slot to switch on Window Is Open mode:
  - true – Window Is Open mode disabled
  - false – Window Is Open mode enabled – the component operates in saving energy mode, all outputs are set to "false".

  Note: Window Is Open mode can be overridden only by Antifrost mode or Test mode.
- Heating Binary On Diff – Numeric input slot with value of temperature (difference between Setpoint and Cv) above which Heating Binary Output is switched on,
- Heating Binary Off Diff – Numeric input slot with value of temperature (difference between Setpoint and Cv) above which Heating Binary Output is switched off,
- Heating Second Stage Threshold – Numeric input slot with threshold of temperature (difference between Setpoint and Cv) above which (with the hysteresis) heating in 2nd stage is switched on,
- Heating Second Stage On Diff – Differential for hysteresis of switching on heating in 2nd stage,
- Heating Second Stage Off Diff – Differential for hysteresis of switching off heating in 2nd stage,
- Cooling Binary On Diff – Numeric input slot with value of temperature (difference between Setpoint and Cv) above which Cooling Binary Output is switched on,
- Cooling Binary Off Diff – Numeric input slot with value of temperature (difference between Setpoint and Cv) above which Cooling Binary Output is switched off,
- Cooling Second Stage Threshold – Numeric input slot with threshold of temperature (difference between Setpoint and Cv) above which (with the hysteresis) cooling in 2nd stage is switched on,
- Cooling Second Stage On Diff – Differential for hysteresis of switching on cooling in 2nd stage,
- Cooling Second Stage Off Diff – Differential for hysteresis of switching off cooling in 2nd stage,
- Threshold Of Fan Demand – Numeric input slot with threshold of temperature (difference

between Setpoint and Cv) above which slot Fan Demand is set to “true”,

- Supply Temperature Low Limit – Minimum acceptable value of supply temperature. This value is used in Supply Air Temperature Limitation function,
- Supply Temperature High Limit – Maximum acceptable value of supply temperature. This value is used in Supply Air Temperature Limitation function,
- Heating Second Stage Enable – Binary input slot, which allows to enable/disable heating in 2nd stage; available values: true – enabled, false – disabled,
- Cooling Second Stage Enable – Binary input slot, which allows to enable/disable cooling in 2nd stage; available values: true – enabled, false – disabled,
- Supply Limit Time – Delay of Supply Air Temperature Limitation function activation.

Supply Air Temperature Limitation

For room user comfort, supply air can have temperature limitation. This function is available only when temperature from supply air sensor is connected and works correctly. Supply air temperature can have upper limitation defined by Supply Temperature High Limit slot and lower limitation defined by Supply Temperature Low Limit slot. The range between Supply Temperature Low Limit and Supply Temperature High Limit values is called “comfort” range.

- Supply Air Temperature limitation in 1st stage - binary control

  When Supply Air Temperature value is out of “comfort” range, the component will start counting delay time set in Supply Limit Time slot. After this time, if supply air temperature value is still out of “comfort” range, the component will disable heating (if temperature value is above Supply Temperature High Limit) or cooling (if temperature value is above Supply Temperature Low Limit). When the supply air temperature value returns to “comfort” range, the component will reset delay counter and return to normal operation.

- Supply Air Temperature limitation in 2nd stage - binary control

  When Supply Air Temperature value is out of “comfort” range, the component will disable 2nd stage and start counting delay time set in Supply Limit Time slot. After this time, if supply air temperature value is still out of “comfort” range, the FCU component will disable heating (if temperature value is above Supply Temperature High Limit) or cooling (if temperature value is above Supply Temperature Low Limit). When the supply air temperature value returns to “comfort” range, the component will reset delay counter, enable 2nd stage and return to normal operation

14.17FCU_ValueHolder

The FCU_ValueHolder component allows for holding the previous value of input by defined time. If value of Input slot has changed, value of Output slot is set to previous value of Input slot for time defined in Holding Time slot. After this time, the Out slot is set to current value from Input slot.

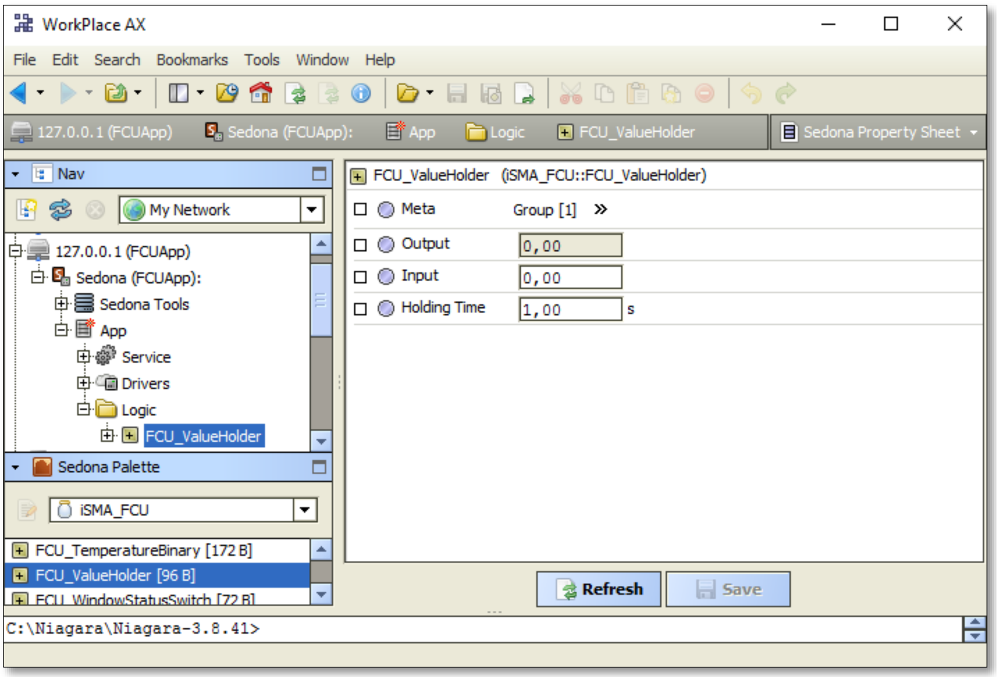


Figure 94 FCU_ValueHolder component

The component has the following slots:

- Input – Input slot,
- Output – Output slot,
- Holding Time – Time in which input value is holding after change.

14.18FCU_WindowStatusSwitch

The FCU_WindowStatusSwitch component calculates Window Status, according to up to 6 Window Status of Master and Slave devices. If Master Window Status slot is set to “true” and Window Statuses of active Slave devices are set to “true”, Window Status Out slot is set to “true”. In other cases, Window Status Out slot is set to “false”.

Note: Slave device is inactive when status slot corresponding to it (slots Slave1 Status – Slave5 Status) is set to “false”. Window Status from inactive device is not used in Window Status calculation.

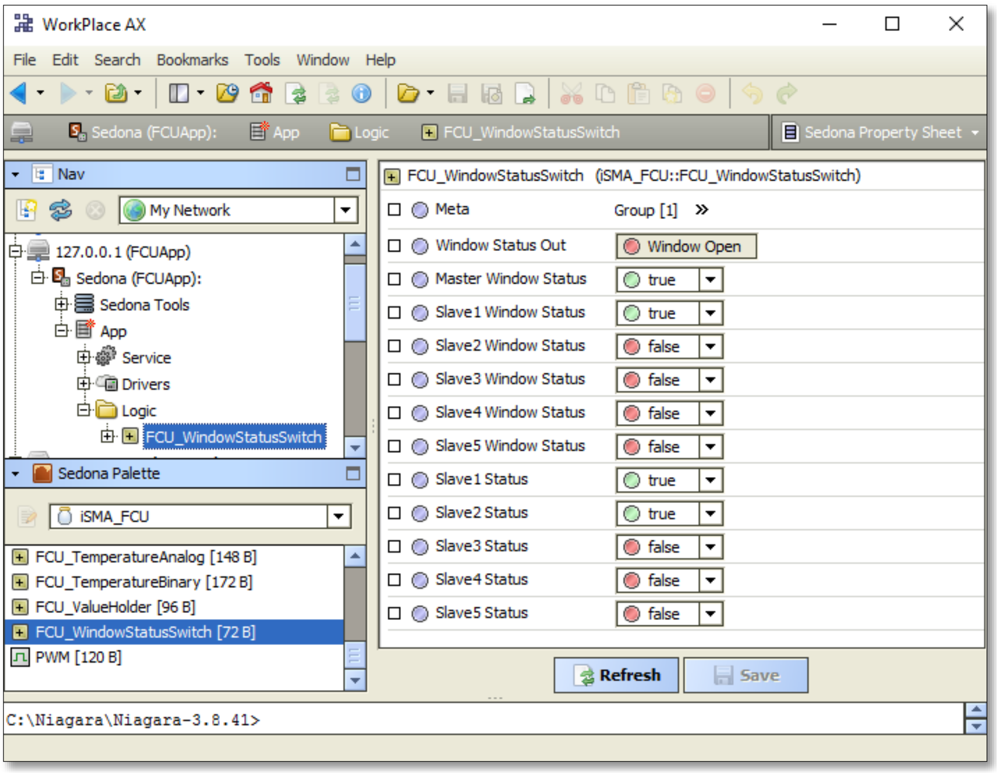


Figure 95 FCU_WindowStatusSwitch component

The component has the following slots:

- Window Status Out – Binary output slot with information of Window Status. Available values:
    - True – Window Closed
    - False – Window Open
- Master Window Status – Binary input slot with Window Status from Master device. Available values:
    - True – Window Closed
    - False – Window Open
- Slave1 Window Status – Slave5 Window Status – Binary input slots with Windows Statuses from Slave devices. Available values:
    - True – Window Closed
    - False – Window Open
- Slave1 Status – Slave5 Status – Binary input slots with statuses of Slave devices. Available values:
    - True – Slave device active
    - False – Slave device inactive

14.19PWM

The PWM component realizes Pulse-Width Modulation (PWM). The PWM bases on two slots:

- Period slot – Time period of cycle,
- Duty Cycle slot – Value of pulse width, expressed as percentage. Duty Cycle slot defines period of time (value from Period slot), during which Out slot will be set to “true”. During the remaining period of time, Out slot will be set to “false”.

- For example, according to the values in figure below:
- Out slot will be set to “true” for 10 seconds – value from Duty Cycle slot multiplied by value from Period slot (10% * 100s),
- After this, Out slot will be set to “false” for 90 seconds (90% * 100s).
- The state of Out slot will be changed in described above way periodically (with periods defined in Period slot).

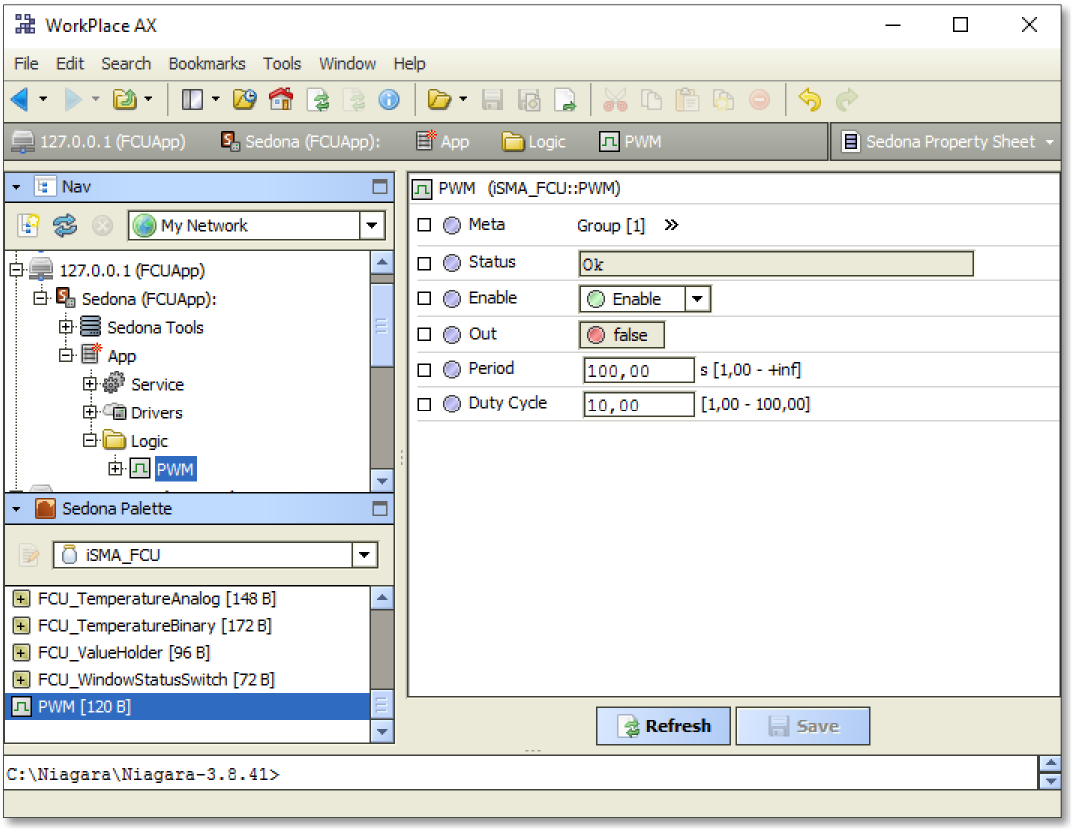


Figure 96 PWM component

The component has the following slots:

- Status – Slot displays status of PWM component,
- Enable – Enable/disable of component; true – enabled, false – disabled,
- Out – Binary output slot with current state, calculated by PWM component,
- Period – Numeric input slot with period of modulation,
- Duty Cycle – Numeric input slot corresponding to the pulse width.