Guide for configuring and working with the Emulator ACFA integration module

ACFA PSIM 1.1

Last update 10/09/2024

Table of Contents

1 Introduction into the Guide for configuring and working with the Emulator ACFA integration module 3
1.1 Purpose of the document 3
1.2 General information about the Emulator integration module 3
2 Supported hardware and licensing of the Emulator ACFA module 4
3 Configuring the Emulator ACFA module 5
3.1 Configuring the parent object of the Emulator ACFA module 5
3.2 Configuring a zone group of the Emulator ACFA module 7
3.3 Configuring fence, lock, sensor, zone, zone detector, loop, input, and output of the Emulator ACFA module 8
3.4 Configuring the reader of the Emulator ACFA module 9
4 Working with the Emulator ACFA module 11
4.1 General information about working with the Emulator ACFA module 11
4.2 Working with the parent object of the Emulator ACFA module 11
4.3 Working with the fence of the Emulator ACFA module 11
4.4 Working with the lock of the Emulator ACFA module 12
4.5 Working with the reader of the Emulator ACFA module 13
4.6 Working with the sensor of the Emulator ACFA module 14
4.7 Working with the zone group of the Emulator ACFA module 15
4.8 Working with the zone of the Emulator ACFA module 16
4.9 Working with the zone detector of the Emulator ACFA module 16
4.10 Working with the loop of the Emulator ACFA module 17
4.11 Working with the input of the Emulator ACFA module 18
4.12 Working with the output of the Emulator ACFA module 19
4.13 Working with the Swagger software 20

1 Introduction into the Guide for configuring and working with the Emulator ACFA integration module

On the page:

- Purpose of the document
- General information about the Emulator integration module

1.1 Purpose of the document

The *Guide for configuring and working with the Emulator ACFA integration module* is a reference and information manual and is intended for configuration specialists and operators of the FSA/ACS and PID. This integration module is a part of *ACFA PSIM*.

The Guide has the following information:

1. General information about the *Emulator ACFA* integration module.
2. Configuring the *Emulator ACFA* integration module.
3. Working with the *Emulator ACFA* integration module.

1.2 General information about the Emulator integration module

The *Emulator ACFA* module is a part of *ACFA PSIM* and allows you to perform the following actions:

- simulate the modes and states of the FSA/ACS and PID hardware;
- simulate the actions of the FSA/ACS and PID operator;
- simulate triggering of hardware and/or change of hardware state to which the operator must respond.

2 Supported hardware and licensing of the Emulator ACFA module

Module licensing

Per one sensor, or per one reader, or per one fence.

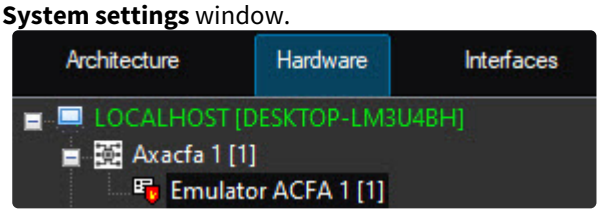
3 Configuring the Emulator ACFA module

3.1 Configuring the parent object of the Emulator ACFA module

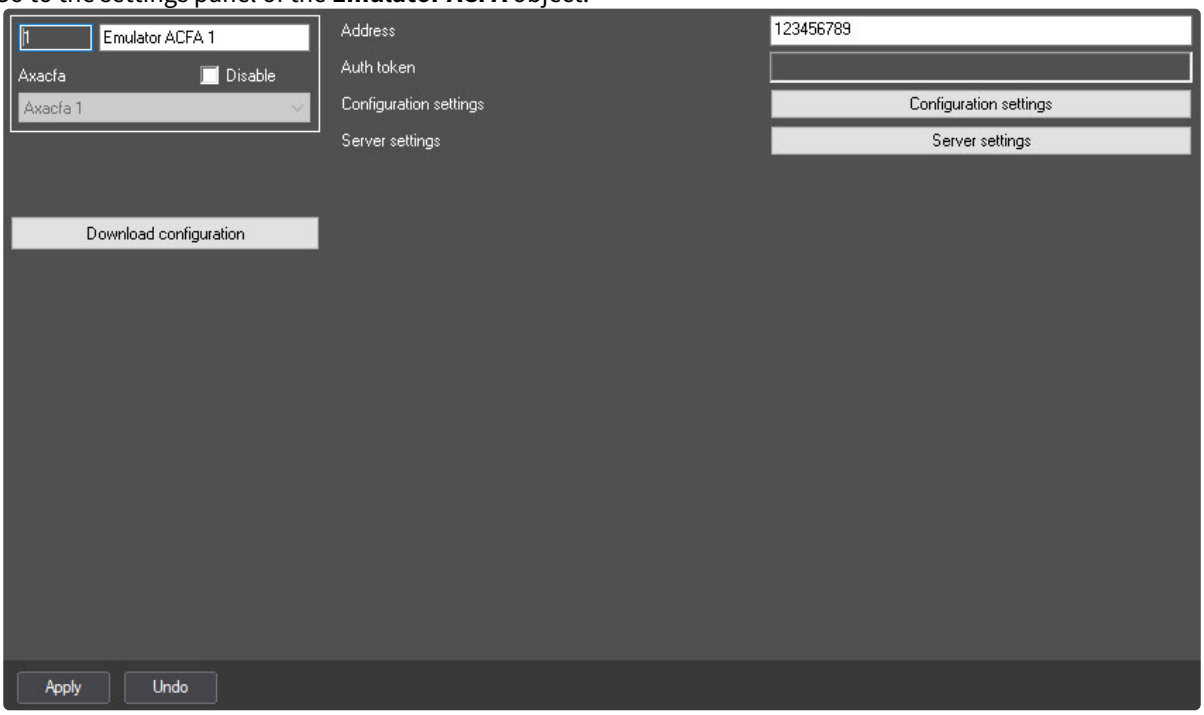
To work with the *Emulator ACFA* integration module, you must install and configure the *AxACFA* feature. For more details, see Connecting and configuring the AxACFA feature.

To configure the parent object of the *Emulator ACFA* module, do the following:

1. Create the **Emulator ACFA** parent object on the basis of the **Axacfa** object on the **Hardware** tab of the **System settings** window.



2. Go to the settings panel of the **Emulator ACFA** object.



3. The **Address** field automatically contains the local IP address of the computer that you can change if necessary. In future versions, it will be possible to specify the IP address of the computer for remote connection and control of the emulator.

4. In the **Auth token** field, specify a token for logging into Swagger software—a framework that allows you to automatically describe APIs based on its code.
5. Click the **Configuration settings** 

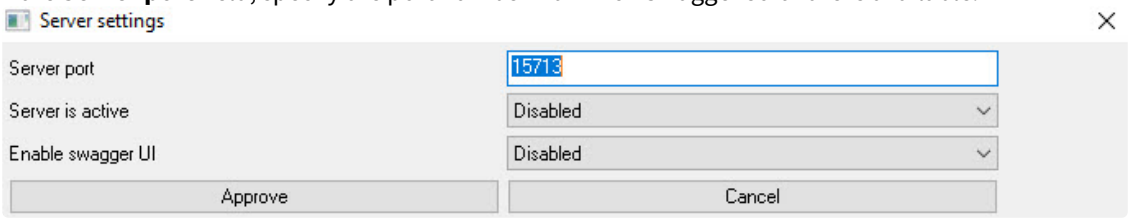
button. As a result, a window with the configuration settings of the *Emulator ACFA* module opens.
    a. In the **ACS units** field, specify the number of ACS (Access Control System) objects you want to create. The default value is **20**. One object consists of one lock and two readers (child objects). You cannot

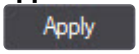
change the number of child readers of one lock.

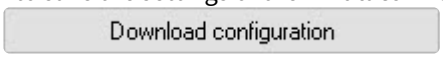
b. In the **FSA units** field, specify the number of FSA (Fire and Security Alarm) of the **Sensor** object. The default value is **20**.
c. In the **Fence units** field, specify the number of PID (Perimeter Intrusion Detection System) of the **Fence** object. The default value is **20**.
d. In the **Loop units** field, specify the number of **Loop** objects. The default value is **20**.
e. In the **Zone group units** field, specify the number of **Zone group** objects. The default value is **20**.
f. In the **Download delay, s** field, specify the interval in seconds between the creation of two objects. The default value is **0**. We don't recommend changing this value.
g. From the **Generate exception** drop-down list, select when a download error is generated: **None** (default), **At beginning**, **In the middle**, **At the end**.
h. Click the **Approve** button to save the configuration settings. Click the **Cancel** button to cancel the changes.

6. Click the **Server settings** 

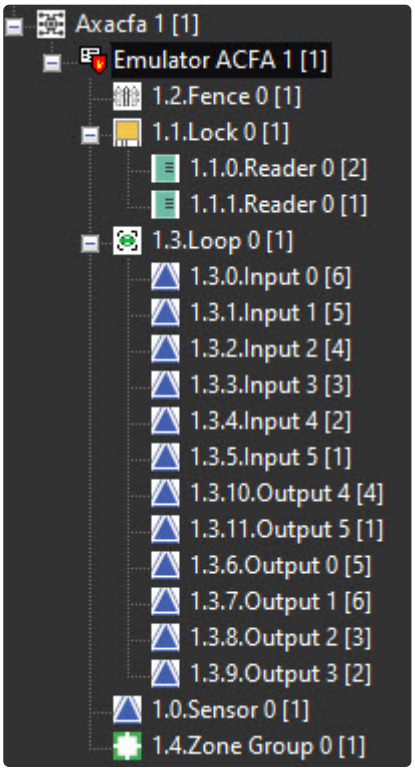
button. The **Server settings** window opens.
   a. In the **Server port** field, specify the port number via which Swagger software is available.



   b. From the **Server is active** drop-down list, select **Enabled** to activate Swagger software. By default, Swagger is disabled (the **Disabled** value is selected).
   c. From the **Enable swagger UI** drop-down list, select **Enabled** to activate Swagger software in the browser. By default, Swagger is disabled in the browser (the **Disabled** value is selected).
   d. Click the **Approve** button to save the server settings. Click the **Cancel** button to cancel the changes.
7. Click the **Apply** 

button to save the settings of the **Emulator ACFA** parent object.
8. Click the **Download configuration** 

button to automatically build the hardware tree according to the configuration specified in the previous steps:



Note

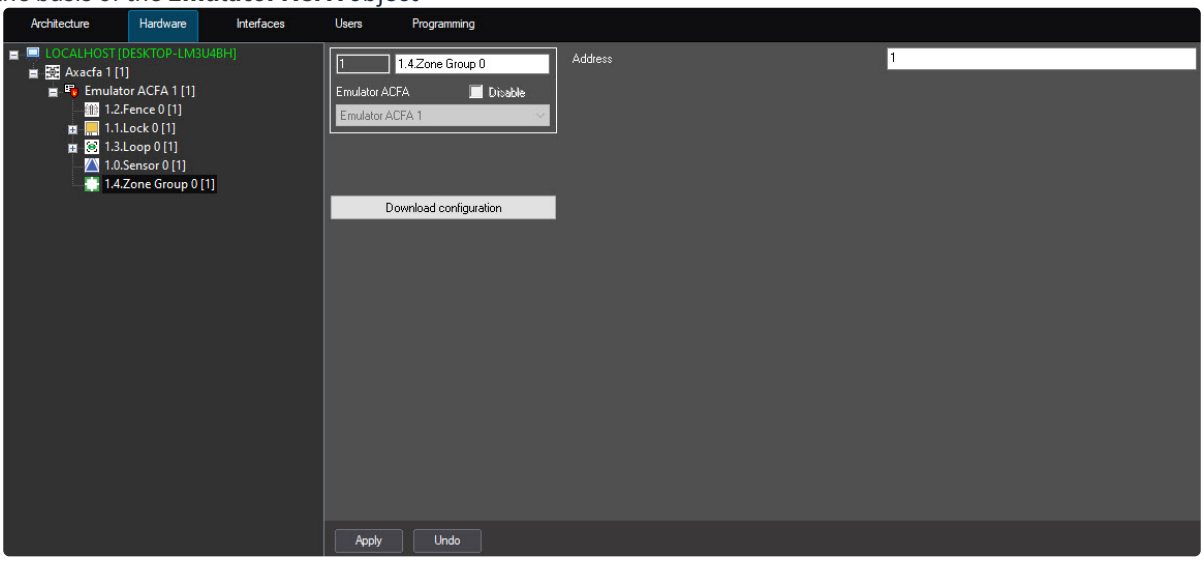
After you download the configuration, six inputs and six outputs are created on the basis of the **Loop** object.

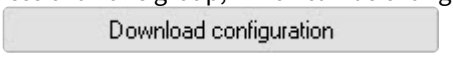
Configuration of the **Emulator ACFA** parent object is complete.

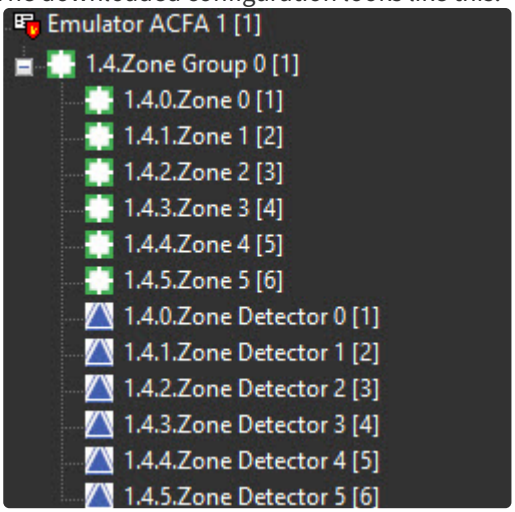
3.2 Configuring a zone group of the Emulator ACFA module

To configure a zone group of the *Emulator ACFA* module, do the following:

1. Go to the settings panel of the **Zone Group** object that is created after you download the configuration on the basis of the **Emulator ACFA** object



2. In the **Address** field, specify the address of a zone group, which can be changed if necessary.
3. Click the **Download configuration**  button to download the configuration.
   As a result, six zones and six zone detectors are created on the basis of the **Zone Group** object.
   The downloaded configuration looks like this:



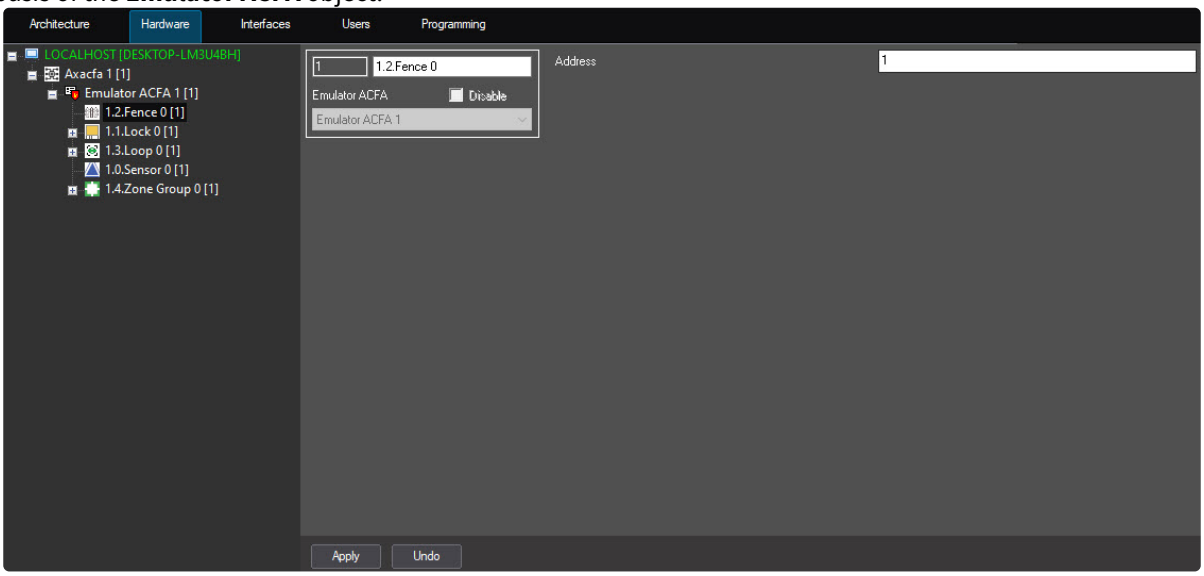
4. Click the **Apply** button to save the settings.

Configuration of a zone group of the *Emulator ACFA* module is complete.

3.3 Configuring fence, lock, sensor, zone, zone detector, loop, input, and output of the Emulator ACFA module

Configuration of the fence, lock, sensor, zone, zone detector, loop, input, and output is shown by the example of the **Fence** object. You can configure the **Lock**, **Sensor**, **Zone**, **Zone Detector**, **Loop**, **Input**, and **Output** objects of the *Emulator ACFA* module in the same way.

1. Go to the settings panel of the **Fence** object that is created when you download the configuration on the basis of the **Emulator ACFA** object.



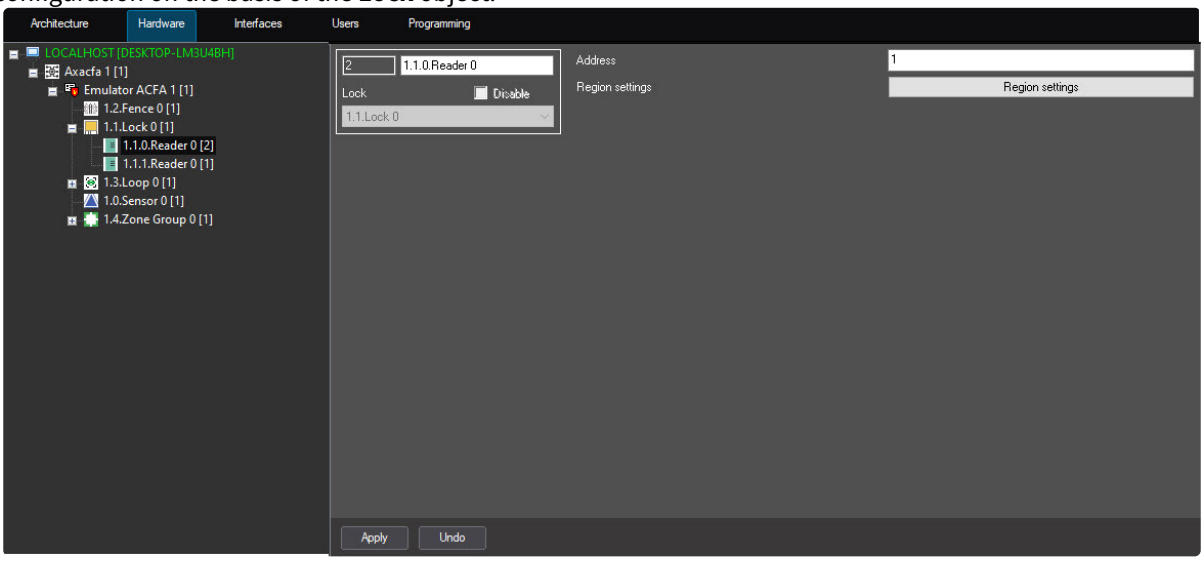
2. The **Address** field, specify the address of the device that you can change if necessary.
3. Click the **Apply** button to save the settings.

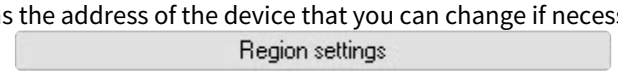
Configuration of the fence of the *Emulator ACFA* module is complete.

3.4 Configuring the reader of the Emulator ACFA module

To configure the reader of the *Emulator ACFA* module, do the following:

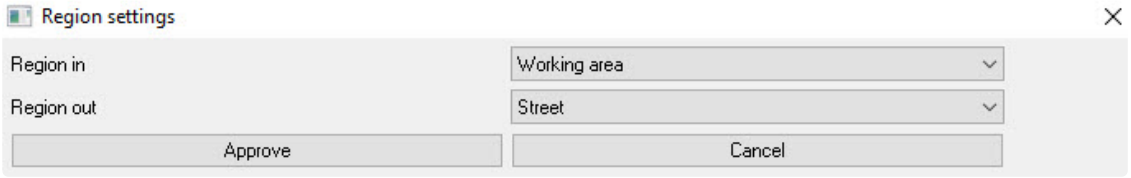
1. Go to the settings panel of the **Reader** object that is created automatically when you download the configuration on the basis of the **Lock** object.



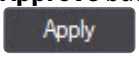
2. The **Address** field contains the address of the device that you can change if necessary.
3. Click the **Region settings** 

button to configure the region in and region out.
4. In the **Region settings** window:
    a. From the **Region in** drop-down list, select the area on the reader exit side.

b. From the **Region out** drop-down list, select the area on the reader entry side.



c. Click the **Approve** button to save the configuration of regions.

5. Click the **Apply**  button to save the changes.

Configuration of the reader of the *Emulator ACFA* module is complete.

4 Working with the Emulator ACFA module

4.1 General information about working with the Emulator ACFA module

The following interface objects are used to work with the *Emulator ACFA* module:

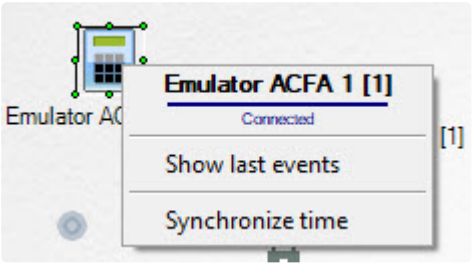
1. **Map**.
2. **Event Viewer**.

For the information on configuring these interface objects, see the *Axxon PSIM* Administrator's Guide.

For the information on working with these interface objects, see the *Axxon PSIM* Operator's Guide.

4.2 Working with the parent object of the Emulator ACFA module

You can manage the parent object of the *Emulator ACFA* module in the **Map** window using the menu of the **Emulator ACFA** object.



Command to manage the parent object of the *Emulator ACFA* module:

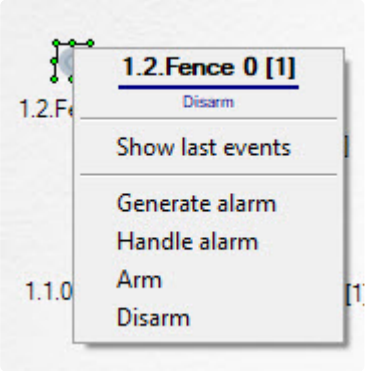
- Synchronize time—write the current time to all devices.

The parent object of the *Emulator ACFA* module can have the following states:

	Connected
	Disconnected
	Cracked

4.3 Working with the fence of the Emulator ACFA module

You can manage the fence of the *Emulator ACFA* module in the **Map** window using the menu of the **Fence** object.



Commands for managing the fence of the *Emulator ACFA* module are described in the table:

Function menu command	**Function**
Generate alarm	Generation of a fence alarm event, executed only after the **Arm** command
Handle alarm	Confirmation of a fence alarm
Arm	Arm the fence
Disarm	Disarm the fence

The fence of the *Emulator ACFA* module can have the following states:

	Disarm
	Arm
	Alarm

4.4 Working with the lock of the Emulator ACFA module

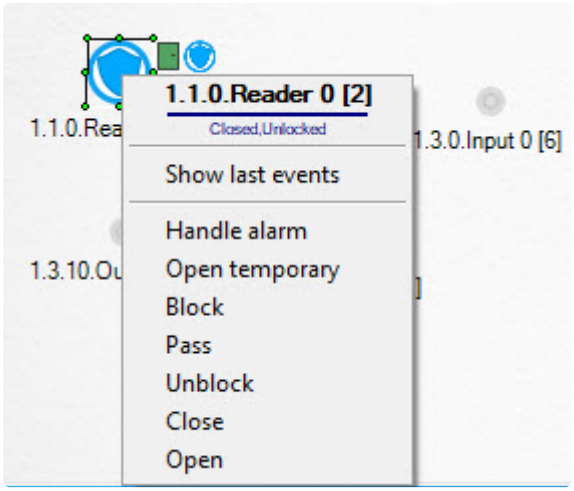
You cannot manage the **Lock** object of the *Emulator ACFA* module in the **Map** window.

The **Lock** object of the *Emulator ACFA* module can have the following states:

	Locked
	Unlocked

4.5 Working with the reader of the Emulator ACFA module

You can manage the reader of the *Emulator ACFA* module in the **Map** window using the menu of the **Reader** object.



Commands for managing the reader of the *Emulator ACFA* module are described in the table:

Function menu command	**Function**
Handle alarm	Confirmation of a reader alarm. Generation of an alarm is possible when you send the **Open** or **Pass** commands only after the **Block** command
Open temporary	Temporary set the reader to the **Open** state
Block	Set the reader to the **Block** state
Pass	Generation of a pass through the reader event
Unblock	Set the reader to the **Unblock** state
Close	Set the reader to the **Close** state
Open	Set the reader to the **Open** state

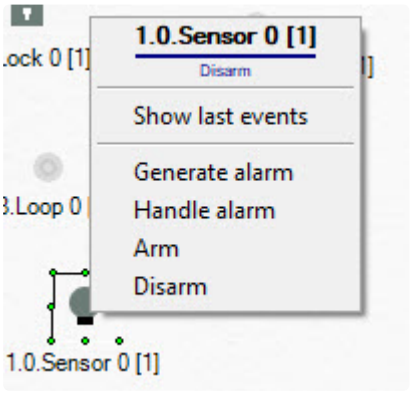
The reader of the *Emulator ACFA* module can have the following states:

	Closed
	Opened

	Locked
	Unlocked
	Alarm

4.6 Working with the sensor of the Emulator ACFA module

You can manage the sensor of the *Emulator ACFA* module in the **Map** window using the menu of the **Sensor** object.



Commands for managing the sensor of the *Emulator ACFA* module are described in the table:

Function menu command	Function
Generate alarm	Generation of a sensor alarm event, executed only after the **Arm** command
Handle alarm	Confirmation of a sensor alarm
Arm	Arm the sensor
Disarm	Disarm the sensor

The sensor of the *Emulator ACFA* module can have the following states:

	Disarm

	Arm
	Alarm

4.7 Working with the zone group of the Emulator ACFA module

You can manage the zone group of the *Emulator ACFA* module in the **Map** window using the menu of the **Zone Group** object.

Commands for managing the zone group of the *Emulator ACFA* module are described in the table:

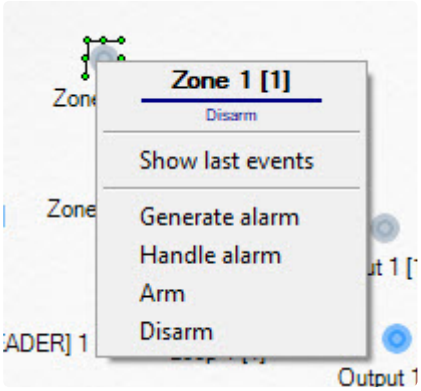
Function menu command	Function
Generate alarm	Generation of an alarm event from all zones and detection tools of a group, executed only after the **Arm** command
Handle alarm	Confirmation of an alarm from all zones and detection tools of a group
Arm	Arm all zones and detection tools of a group
Disarm	Disarm all zones and detection tools of a group

The zone group of the *Emulator ACFA* module can have the following states:

	Disarm
	Arm
	Alarm

4.8 Working with the zone of the Emulator ACFA module

You can manage the zone of the *Emulator ACFA* module in the **Map** window using the menu of the **Zone** object.



Commands for managing the zone of the *Emulator ACFA* module are described in the table:

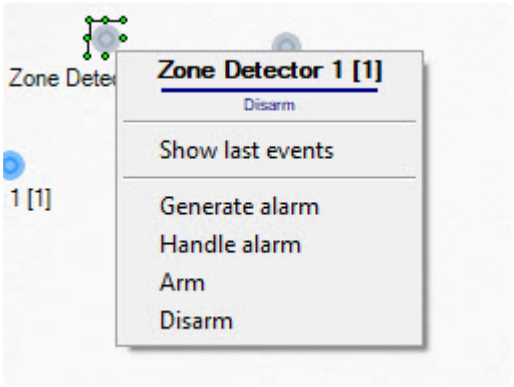
Function menu command	Function
Generate alarm	Generation of a zone alarm event, executed only after the **Arm** command
Handle alarm	Confirmation of a zone alarm
Arm	Arm the zone
Disarm	Disarm the zone

The zone of the *Emulator ACFA* module can have the following states:

	Disarm
	Arm
	Alarm

4.9 Working with the zone detector of the Emulator ACFA module

You can manage the zone detector of the *Emulator ACFA* module in the **Map** window using the menu of the **Zone Detector** object.



Commands for managing the zone detector of the *Emulator ACFA* module are described in the table:

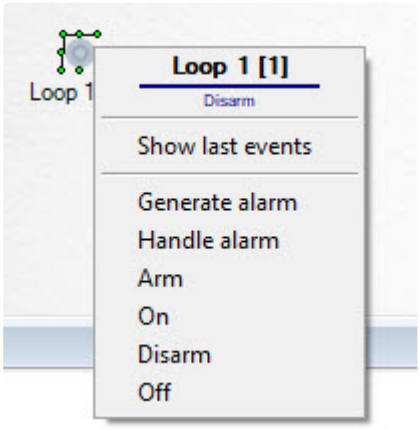
Function menu command	Function
Generate alarm	Generation of an alarm event from the zone detector, executed only after the **Arm** command
Handle alarm	Confirmation of an alarm from the zone detector
Arm	Arm the zone detector
Disarm	Disarm the zone detector

The zone detector of the *Emulator ACFA* module can have the following states:

	Disarm
	Arm
	Alarm

4.10 Working with the loop of the Emulator ACFA module

You can manage the loop of the *Emulator ACFA* module in the **Map** window using the menu of the **Loop** object.



Commands for managing the loop of the *Emulator ACFA* module are described in the table:

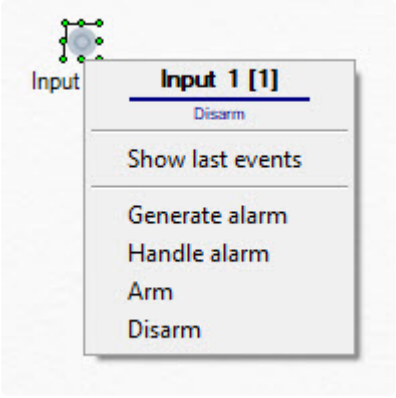
Function menu command	**Function**
Generate alarm	Generation of an alarm event from all inputs and outputs of the loop, executed only after the **Arm** command
Handle alarm	Confirmation of an alarm from all inputs and outputs of the loop
Arm	Arm all inputs and outputs of the loop
On	Turn on all inputs and outputs of the loop
Disarm	Disarm all inputs and outputs of the loop
Off	Turn off all inputs and outputs of the loop

The loop of the *Emulator ACFA* module can have the following states:

	Disarm
	Arm
	Alarm

4.11 Working with the input of the Emulator ACFA module

You can manage the input of the *Emulator ACFA* module in the **Map** window using the menu of the **Input** object.



Commands for managing the input of the *Emulator ACFA* module are described in the table:

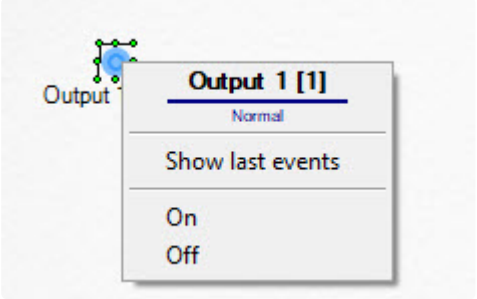
Function menu command	**Function**
Generate alarm	Generation of an input alarm event, executed only after the **Arm** command
Handle alarm	Confirmation of an input alarm
Arm	Arm the input
Disarm	Disarm the input

The input of the *Emulator ACFA* module can have the following states:

	Disarm
	Arm
	Alarm

4.12 Working with the output of the Emulator ACFA module

You can manage the output of the *Emulator ACFA* module in the **Map** window using the menu of the **Output** object.



Commands for managing the output of the *Emulator ACFA* module are described in the table:

Function menu command	Function
On	Turn on the output
Off	Turn off the output

The output of the *Emulator ACFA* module can have the following states:

	Normal
	Active

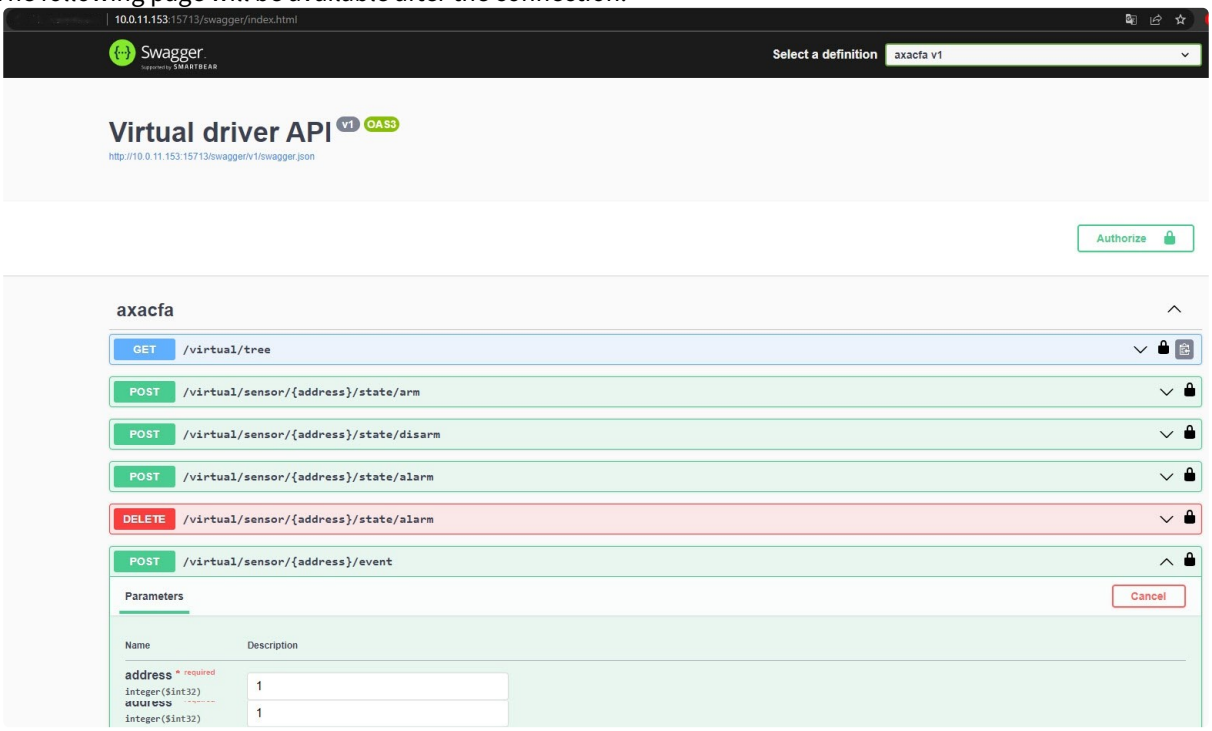
4.13 Working with the Swagger software

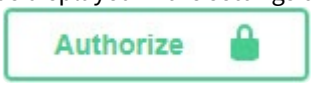
In the *Emulator ACFA* integration module, you can work with the Swagger software directly via POST requests. To do this, do the following:

1. Activate the Swagger software and Swagger UI (see Configuring the parent object of the Emulator ACFA module).
2. Connect to the Swagger software in a web browser using the string:

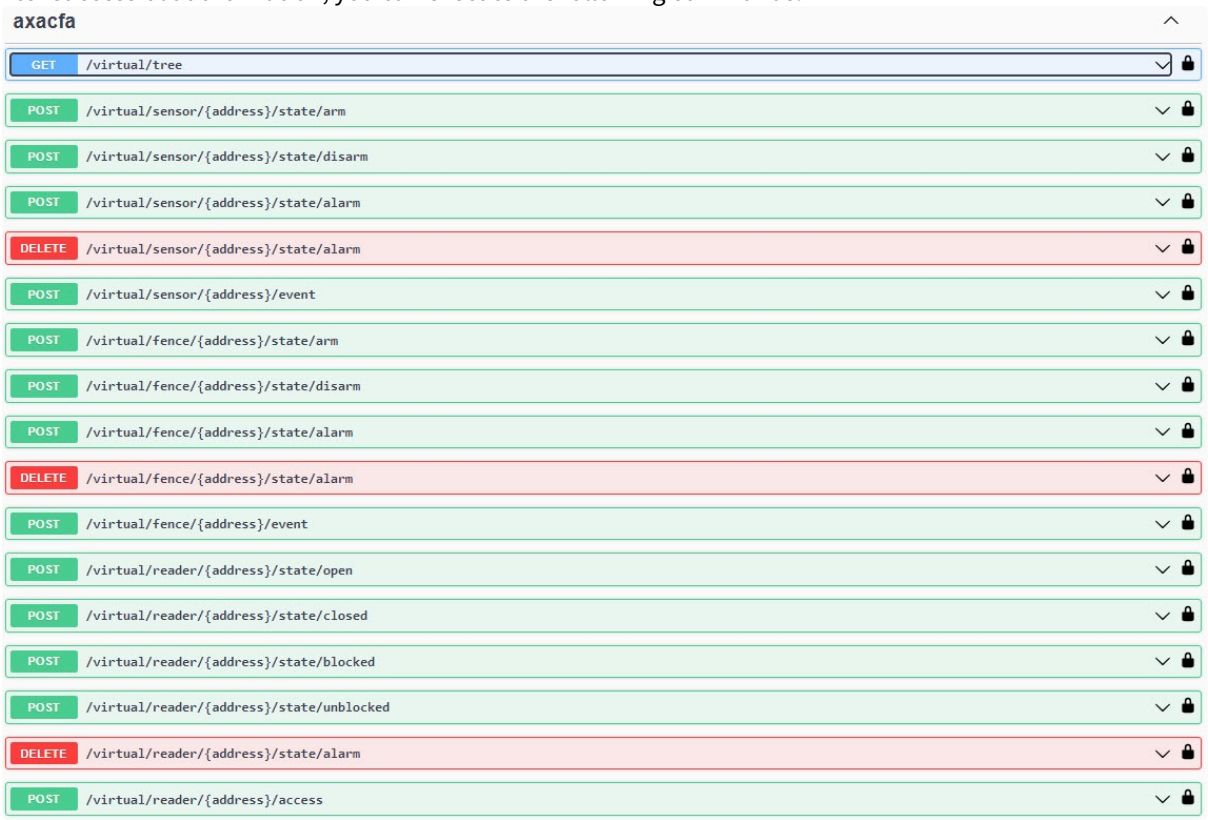
```
http://{ip}:{port}/swagger
```

The following page will be available after the connection:

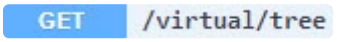


3. You can authorize by the token, which was displayed in the settings of the parent object of the *Emulator ACFA* module, by clicking the **Authorize** 

button in the upper right corner of the page.
4. After successful authorization, you can execute the following commands:



Examples of commands:

a. Read the hardware tree with the current states. To do this, select the 

request, first click the **Try it Out** button and then click the **Execute** button.
b. Switch the sensor with the specified address to the **arm** state by selecting the corresponding request and following the same steps as in step 4a.
c. Switch the sensor with the specified address to the **disarm** state by selecting the corresponding request and following the same steps as in step 4a.
d. Switch the sensor with the specified address from the **arm** state to the **alarm** state by selecting the corresponding request and following the same steps as in step 4a.
e. Process the alarm and switch the sensor to the **disarm** state by selecting the corresponding request and following the same steps as in step 4a.
f. And so on, similarly to the described commands.