



ApolloSDK Integration Module Settings Guide (obsolete)

ACFA PSIM 1.1

Last update 05/06/2024

Table of Contents

1	List of Terms used in the ApolloSDK Integration Module Settings Guide (obsolete)	3
2	ApolloSDK Integration Module Settings Guide (obsolete). Introduction	4
2.1	Purpose of the document	4
2.2	General information about the ApolloSDK integration module.....	4
3	Configuration of the ApolloSDK integration module (obsolete)	6
3.1	How to configure the ApolloSDK integration module (obsolete).....	6
3.2	Configuring access cards in the ApolloSDK integration module (obsolete)	6
3.3	Configuration of an AAN controller (obsolete)	7
3.3.1	Configuring ports of the AAN controller (obsolete).....	10
3.3.2	Saving configuration of the AAN controller (obsolete)	11
3.4	Configuration of the connection with ACS modules and alarm panels (obsolete)	12
3.4.1	Configuration of the connection with an AIM interface module (obsolete).....	12
3.4.2	Configuration of the connection with an AIO alarm panel (obsolete)	13
3.4.3	Configuration of an ASA status panel (obsolete).....	14
3.5	Configuration of readers and sensors (obsolete)	16
3.5.1	Configuration of the AIM interface module's readers (obsolete)	16
3.5.2	Configuration of the AIO alarm panel's inputs (obsolete)	18
3.5.3	Configuration of the AIO alarm panel's relay (obsolete).....	19
3.5.4	Configuration of an ASA status panel's sensors (obsolete)	21
4	Working with the ApolloSDK integration module (obsolete)	23
4.1	General information about working with the ApolloSDK Module (obsolete)	23
4.2	Managing an AAN controller (obsolete)	23
4.3	Managing the AIM interface module's readers (obsolete)	23
4.4	Managing the AIO alarm panel's inputs (obsolete)	24
4.5	Managing the AIO alarm panel's relay (obsolete).....	24

1 List of Terms used in the ApolloSDK Integration Module Settings Guide (obsolete)

Server - a computer configured as an *Axxon PSIM* Software System **Server**.

Access control system (ACS) – a system of hardware and software designed to monitor and control access.

Controller - an electronic device designed to monitor and control access points.

Access point – a place where access is controlled.

AIM Interface Module – an interface module for connecting readers or keypads to an AAN controller.

AIO alarm panel – a microprocessor-based alarm panel that monitors the state of alarm inputs and manages relay outputs.

ASA status panel – a panel used to display the state of the security alarm.

Alarm input – an input to switch alarm sensors (magnetic-contact, impact-contact) or the output circuits of IR sensors and alarm panels.

Readers – electronic devices designed for entering a memorized code using a keypad or reading encoded data from system keys (identifiers).

Access key – a physical or digital key used to access objects in facilities, buildings, zones, and areas.

Access card – a physical access key accepted by a reader.

Access time – the time allotted to pass through an access point. When the allotted time has passed the access point locks automatically.

Impulse – a signal used to close a relay.

Time schedule - a set of any number of time intervals during a day (24 hours) defined for several days (1 to 366), and the time intervals during specific dates. Time schedule defines a schedule of access to the secured object.

Doubled readers are readers installed on both sides of the same door and controlling the same reed, lock, etc.

2 ApolloSDK Integration Module Settings Guide (obsolete). Introduction

On the page:

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

2.1 Purpose of the document

This *ApolloSDK Module Settings Guide* is a reference manual designed for *ApolloSDK* Module configuration technicians and operators. This module functions as part of security- and fire alarm systems and access control systems built on the basis of the *ACFA PSIM* Software System.

This Guide presents the following materials:

1. General information about the *ApolloSDK* integration module;
2. Configuration of the *ApolloSDK* integration module;
3. Working with the *ApolloSDK* integration module.

2.2 General information about the *ApolloSDK* integration module

Attention!

The *ApolloSDK* integration is not supported.

The *ApolloSDK* integration module is part of *SFA/ACS* systems built based on the *ACFA PSIM* Software System. It is designed to configure and control *ApolloSDK* hardware.

The following hardware is integrated with the *ACFA PSIM* Software System:

1. *AAN* controllers (an *ACS* component);
2. *AIM* interface modules (an *ACS* component);
3. *AIO* alarm panels (an *SFA* component);
4. *ASA* status panels (an *SFA* component).

The network connection for these devices is either Ethernet or RS-485.

Note:

Detailed information about the *ApolloSDK* system can be found in the official documentation (manufacturer AAM Systems).

 **Attention!**

The *ApolloSDK* software must be installed on the Server for the *ApolloSDK* integration module to operate.

Before configuring the *ApolloSDK* integration module, the following actions must be performed:

1. Install the required hardware on the site.
2. Install the *ApolloSDK* software on the Server.

3 Configuration of the ApolloSDK integration module (obsolete)

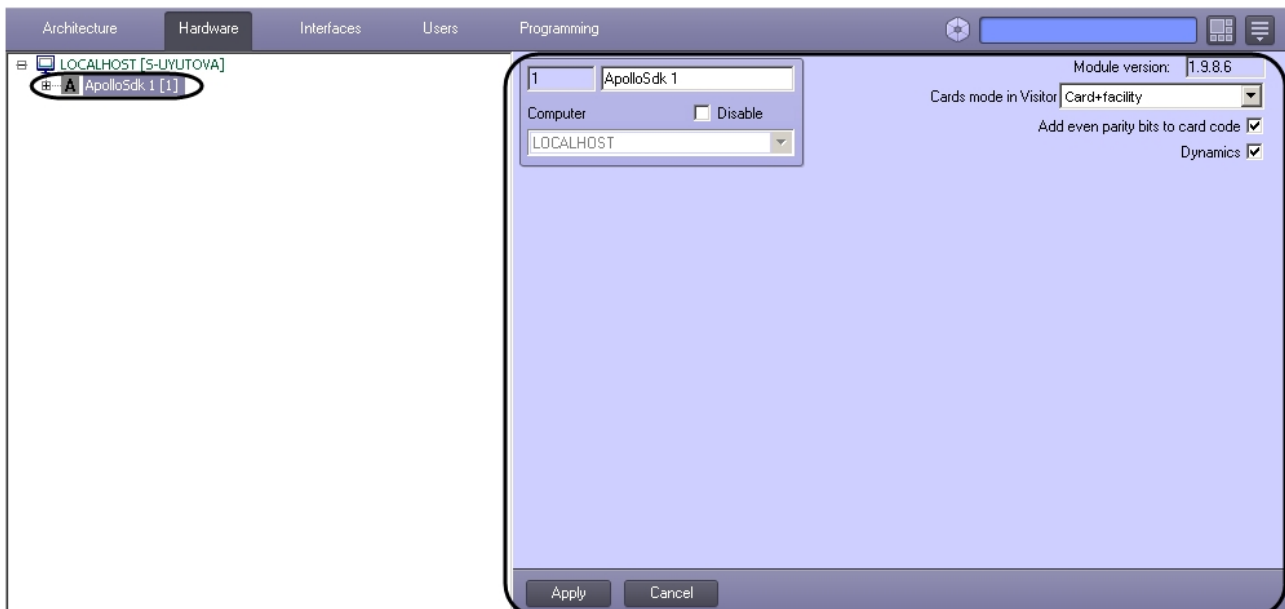
3.1 How to configure the ApolloSDK integration module (obsolete)

The *ApolloSDK* integration module in *ACFA PSIM* is configured as follows:

1. Configuration of the access cards.
2. Configuration of an *AAN* controller.
3. Configuration of the connection with ACS modules and alarm panels (obsolete).
4. Configuration of readers and sensors (obsolete).

3.2 Configuring access cards in the ApolloSDK integration module (obsolete)

To configure access cards, go to the **ApolloSDK** object settings panel. Create this object based on the **Computer** object on the **Hardware** tab of the **System settings** dialog box.

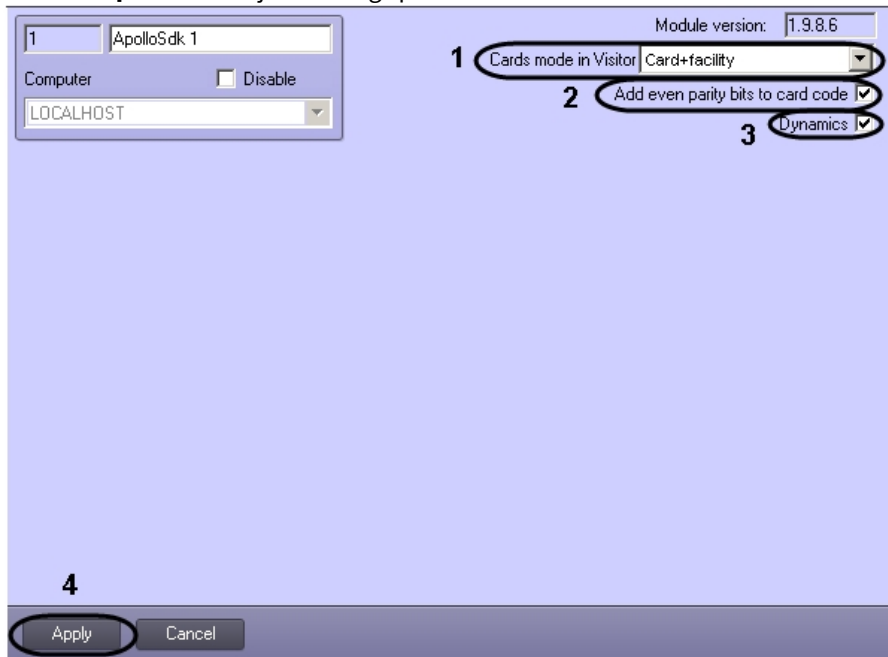


Note.

In the **Module version** field, the current version of the ApolloSDK integration module is displayed.

To configure access cards do the following:

1. Go to the **ApolloSDK** object settings panel.



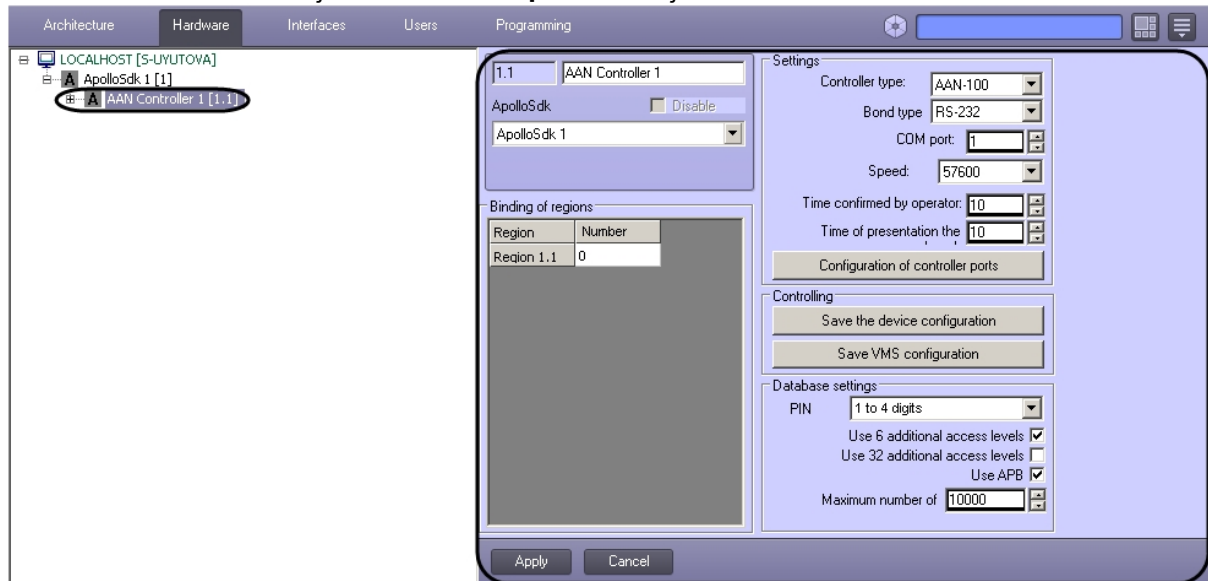
2. In the **Cards mode in Visitor** drop-down list select the way cards are set in the *Access Manager* module (1).
3. **Card+facility**: card number and facility code are in use.
4. **Card only**: only card number is in use.
5. If the even parity bit is to be added to the card code, set the corresponding checkbox (2).
6. If changes in access parameters are to be dynamically send to the ApolloSDK system, set the **Dynamics** checkbox (3).
7. Click **Apply** (4).

Access cards are now configured.

3.3 Configuration of an AAN controller (obsolete)

The *AAN* controller is configured as follows:

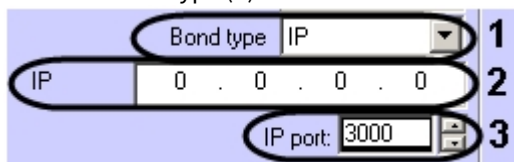
1. Create an **AAN Controller** object based on the **ApolloSDK** object.



2. Go to the **AAN Controller** object's settings page (1).
3. If the **AAN** controller is connected to the Server through a COM port, then do the following:
 - a. Set the **RS-232** bond type (1)
 - b. In the **COM port** field, enter the number of the COM port used to connect the **AAN** controller to the Server (2).



- c. From the **Speed** dropdown list, select the speed of the **AAN** controller's COM-port connection (3).
4. If the **AAN** controller is connected to the Server over Ethernet, then do the following:
 - a. Set the **IP** bond type (1).



- b. In the **IP address** field, enter the **AAN** controller's IP address (2).
- c. Enter the **AAN** controller's port number in the **IP Port** field (3).

5. Select the type of AAN controller (AAN-32 or AAN-100) from the **Type** dropdown list, depending on the type of hardware being used (1).

6. In the **Time confirmed by operator** field, enter the time in seconds allotted to the operator to make a decision to grant or deny access (2).
7. In the **Time of presentation the card** field, enter the period of time in seconds between the presentation of the first and second access cards which, if exceeded, will result in access not being granted (3).
8. In the **PIN** drop-down list select the type of PIN code (4):
- Disabled
 - 1 to 4 digits
 - 1 to 6 digits
9. By default, each card has only one access level. It is possible to forcibly enhance the number of access levels per card (up to 6, 32 or 38), but this will reduce the total number of cards that can be recorded to the controller. To enable additional access levels, do the following:
- Set the **Use 6 additional access levels** checkbox to enable 6 additional access levels (5).
 - Set the **Use 32 additional access levels** checkbox to enable 32 additional access levels (6).
10. If anti-passback is to be enabled, set the **Use APB** checkbox (7).
11. In the **Max number of cards** field, enter the maximum number of access cards that will be stored in the controller's memory (8).

Note:

The maximum number of access cards that can be stored in the controller's memory depends on the number of memory cards installed in it.

12. Proceed with binding AAN controller regions with ACFA PSIM regions — in the **Binding regions** table there is a list of **Region** objects created in the ACFA PSIM software. Specify the numbers of AAN controller regions corresponding to them (9).

Note.

The **Region** objects are created based on the **Area** objects on the **Programming** tab of the **System Settings** dialog box.

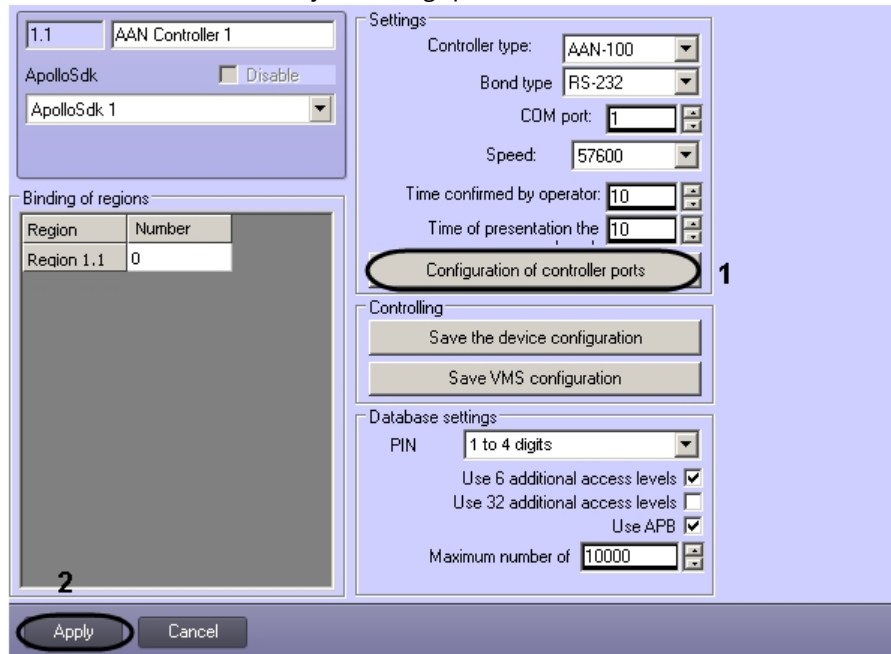
13. To save any changes made, click the **Apply** button (10).
14. To save changes into the controller send the configuration (see [Saving configuration of the AAN controller \(obsolete\)](#)).

This completes the configuration of the AAN controller.

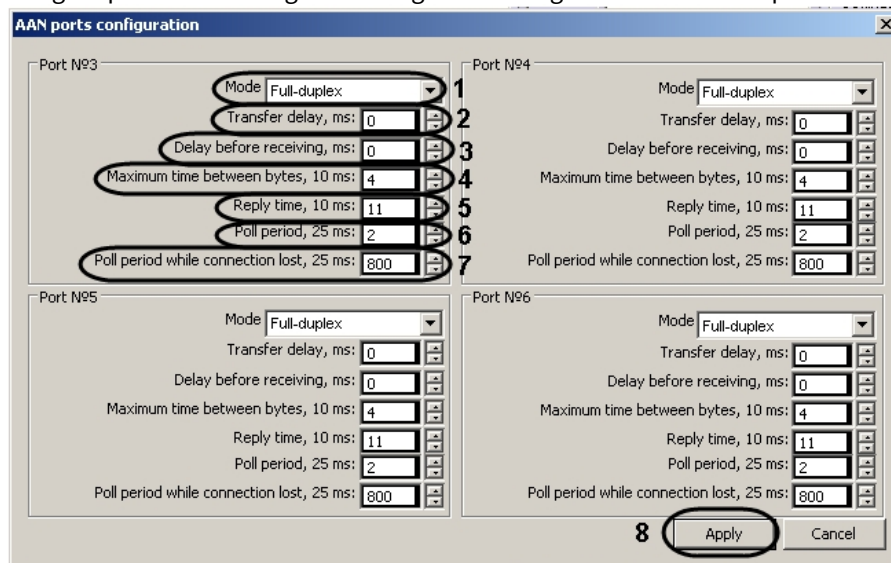
3.3.1 Configuring ports of the AAN controller (obsolete)

To configure AAN controller ports used to connect minor controllers (AIM) proceed as follows:

1. Go to the **AAN Controller** object settings panel.



2. Click **Configuration of controller ports**(1). The **AAN ports configuration** dialog box opens. One can configure ports 3 to 6 using this dialog box. Settings are similar for all ports.



3. In the **Mode** drop-down list select the mode of data transmission through port: full-duplex or half-duplex (1)
4. Specify the transfer delay in milliseconds (2).

5. Specify the delay before receiving data in milliseconds (3).
6. Specify the maximum time for waiting the next byte in milliseconds (4).

Note.
 In the name of this and subsequent settings the multiplicity of the specified value is indicated after comma. Thus, if you set the **Maximum time between bytes, 10 ms** value equal to 4, the maximum time for waiting the next byte will be equal to 40 ms.

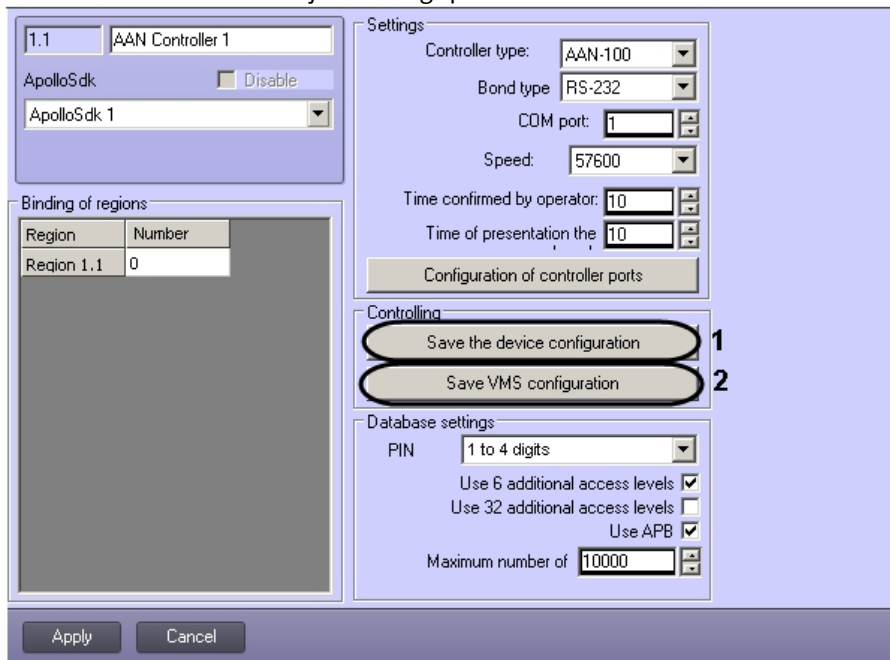
7. Specify the reply time (5) taking into account given multiplicity.
8. Specify the poll period when the link is present in milliseconds (6) taking into account given multiplicity.
9. Specify the poll period when the link is lost in milliseconds (7) taking into account given multiplicity.
10. Click **Apply** (8).
11. Click **Apply** on the settings panel of the **AAN Controller** object (2).

Ports of the AAN controller are now configured.

3.3.2 Saving configuration of the AAN controller (obsolete)

Sending configuration to the AAN controller is performed as follows:

1. Go to the **AAN Controller** object settings panel.



2. To send the hardware configuration click **Save the device configuration** (1).
3. To send data on users and access parameters click **Save VMS configuration** (2).

Saving configuration of the AAN controller is now completed.

3.4 Configuration of the connection with ACS modules and alarm panels (obsolete)

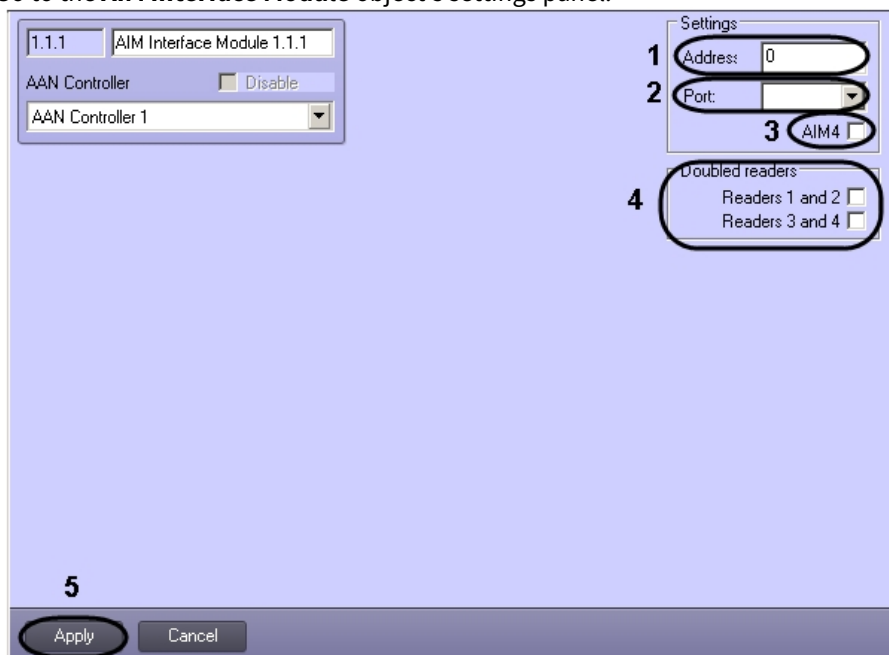
3.4.1 Configuration of the connection with an AIM interface module (obsolete)

The connection with an *AIM* Interface Module is configured on the settings panel of the corresponding object. An **AIM Interface Module** object is created based on the **AAN Controller** object.



The connection with an *AIM* Interface Module is configured as follows:

1. Go to the **AIM Interface Module** object's settings panel.



2. In the **Address** field, enter the *AIM* Interface Module's unique address (1).

3. Select the port used to connect to the *AIM* Interface Module from the **Port** dropdown list (2).
4. If the AIM4 interface module is in use, set the **AIM4** checkbox (3).
5. If doubled readers are in use in the ApolloSDK system, set the checkboxes in front of those readers which are doubled (4).
6. To save changes, click the **Apply** button (5).

This completes the configuration of the connection with the *AIM* interface module.

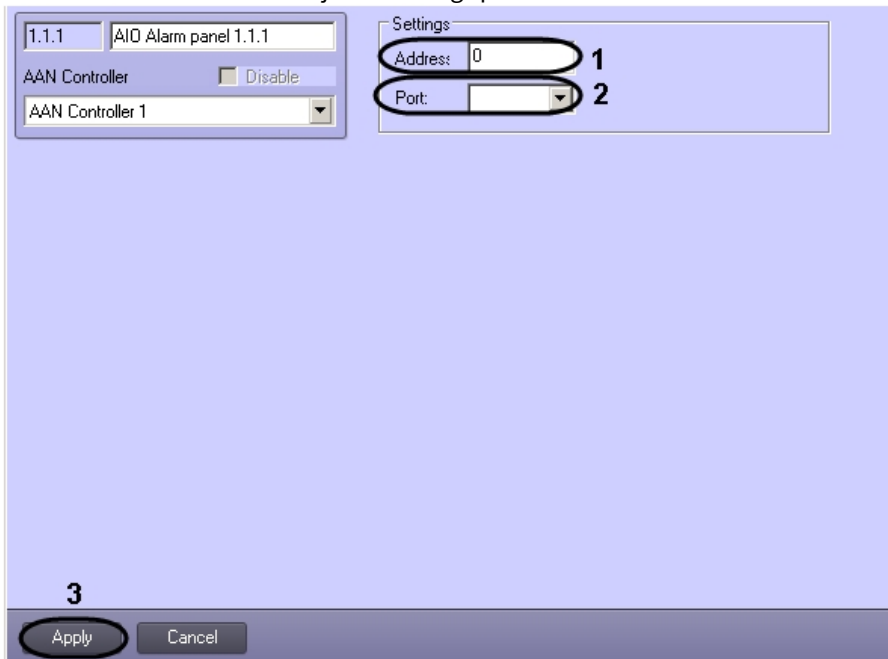
3.4.2 Configuration of the connection with an AIO alarm panel (obsolete)

The connection with an *AIO* alarm panel is configured on the settings pane of the corresponding object. An **AIO Alarm Panel** object is created based on the **AAN Controller** object.



The connection with an *AIO* alarm panel is configured as follows:

1. Go to the **AIO Alarm Panel** object's settings panel.

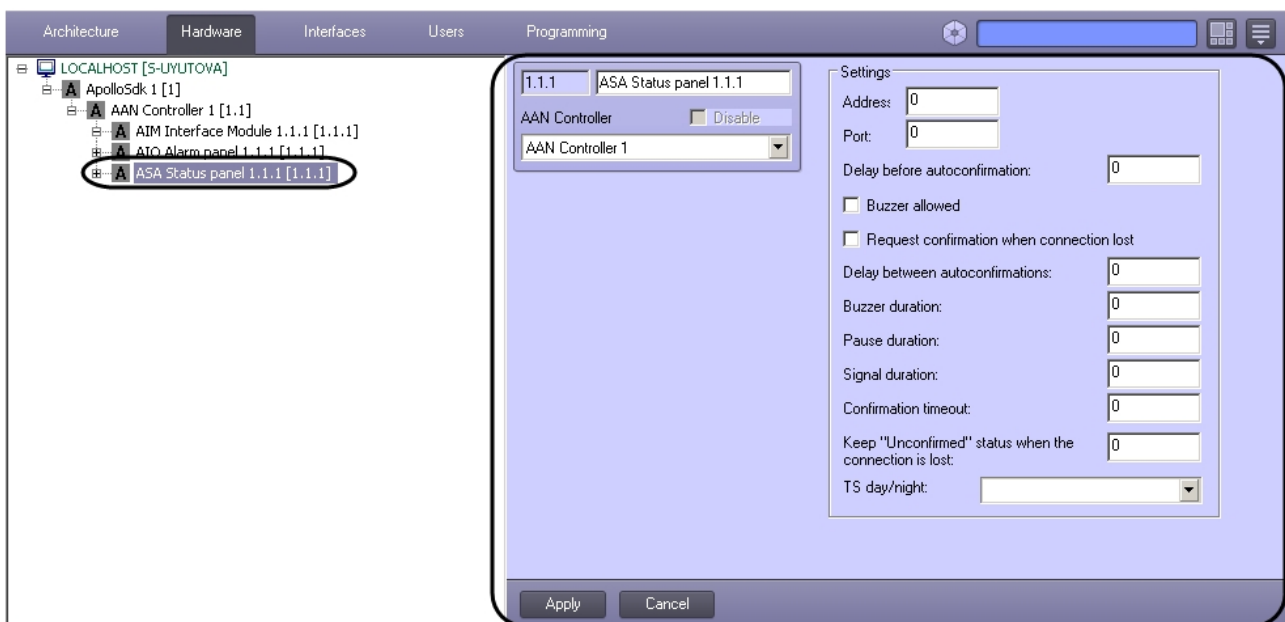


2. In the **Address** field, enter the *AIO* alarm panel's unique address (1).
3. Select the port used to connect to the *AIO* alarm panel from the **Port** dropdown list (2).
4. To save changes, click the **Apply** button (3).

This completes the configuration of the connection with the *AIO* alarm panel.

3.4.3 Configuration of an ASA status panel (obsolete)

An **ASA** status panel is configured on the settings pane of the corresponding object. An **ASA Status Panel** object is created based on the **AAN Controller** object.



The connection with an **ASA** status panel is configured as follows:

1. Go to the **ASA Status Panel** object's settings panel.

2. In the **Address** field, enter the ASA panel's unique address (**1**).
3. Select the port used to connect to the ASA panel from the **Port** dropdown list (**2**).
4. In the **Delay before autoconfirmation** field, enter the time in seconds before an alarm is automatically confirmed (**3**).
5. Check the **Buzzer enabled** checkbox if the ASA panel's buzzer is to be enabled (**4**).
6. If confirmation is required when the connection between the ASA panel and the controller is lost, then check the appropriate checkbox (**5**).
7. In the **Delay between autoconfirmations** field, enter the time delay (in seconds) between automatically confirmed alarms (**6**).
8. In the **Buzzer duration** field, enter the duration (in seconds) of the buzzer's tones (**7**).
9. In the **Pause duration** field, enter the duration (in seconds) of the pause between the buzzer's tones (**8**).
10. In the **Signal duration** field, enter the duration (in seconds) of the buzzer's light signal (**8**).

Note:

The signal consists of buzzing and pauses. In order for the ASA panel's buzzer to operate correctly, the value of **Signal duration** must be greater than or equal to the sum of **Buzzer duration** and **Pause duration**.

11. In the **Confirmation timeout** field, enter the time period (in seconds) before confirmation of an alarm (**10**).
12. In the **Keep "Unconfirmed" status when the connection is lost** field, enter the time period (in seconds) during which the **Unconfirmed** status will be preserved during connection loss (**11**).
13. From the **TS day/night** dropdown list, select the time schedule that will define day for the ASA panel (**12**).

Note:

Any time outside the selected time schedule will be considered night.

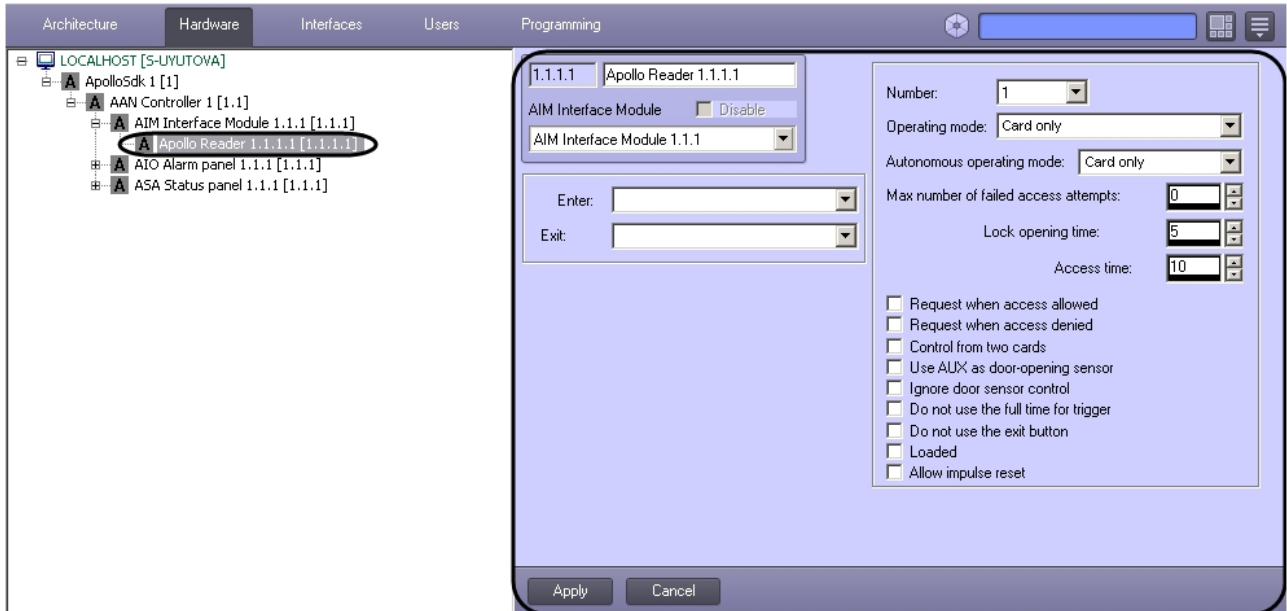
14. To save changes, click the **Apply** button (**13**).

This completes the configuration of the ASA panel.

3.5 Configuration of readers and sensors (obsolete)

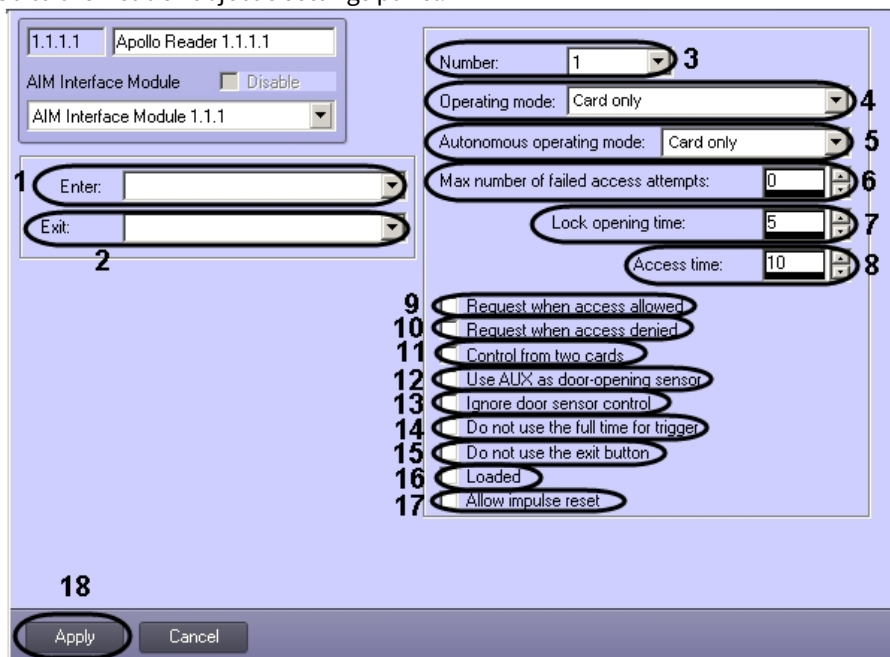
3.5.1 Configuration of the AIM interface module's readers (obsolete)

The AIM Interface Module's readers are configured on the settings panel of the corresponding object. A **Reader** object is created based on the **AIM Interface Module** object.



Readers are configured as follows:

1. Go to the **Reader** object's settings panel.



2. From the **Enter** dropdown list select the **Section** object that corresponds to the area to which this reader enters (1).
3. From the **Exit** dropdown list select the **Section** object that corresponds to the area to which this reader exits (2).
4. From the **Number** dropdown list select the reader's index number (3).
5. From the **Operating mode** dropdown list select the reader's operating mode (4).

Operating mode	Operating mode description
Closed	Access is denied to all
Card only	Access is granted based on access cards
PIN or Card	Access is granted based on access cards or PIN codes
PIN and Card	Access is granted based on access cards and PIN codes
Opened	Access is granted to all
Company code	Access is granted based on company codes

6. From the **Autonomous operating mode** dropdown list select the operating mode to be employed by the reader if the connection with the controller is lost (5).
7. In the **Max number of failed access attempts** field, enter the number of failed access attempts that can be made before a **Break-in Attempt** message is sent (6).
8. In the **Lock opening time** field, enter the time period (in seconds) in which the lock will be unlocked (7).
9. In the **Access time** field, enter the time allotted for access (8).
10. Check the **Request when access allowed** checkbox if an access request must be sent to the operator when an access attempt has been made successfully (9).
11. Check the **Request when access denied** checkbox if an access request must be sent to the operator when an access attempt has failed (10).
12. Check the **Control from two cards** checkbox if access requires two access cards being presented (11).
13. If the *AIM* module's auxiliary sensor *AUX* is to be used as the door-opening sensor, then check the corresponding option (12).
14. If the door sensor should be ignored, then check the corresponding option (13).
15. If it is not required to wait for the full access time after the card is put to the reader for the lock of the "trigger" type, i.e. the lock is to be closed immediately after the door is closed, set the **Do not use the full time for trigger** checkbox (14).
16. If the exit button is to be forbidden, set the corresponding checkbox (15).
17. By default, the cards are saved in the main AAN controller only. If the cards are to be saved not only in the main controller but in minor controllers too, so that when connection between them is lost the readers could operate off-line, set the corresponding checkbox (16).

 **Note.**

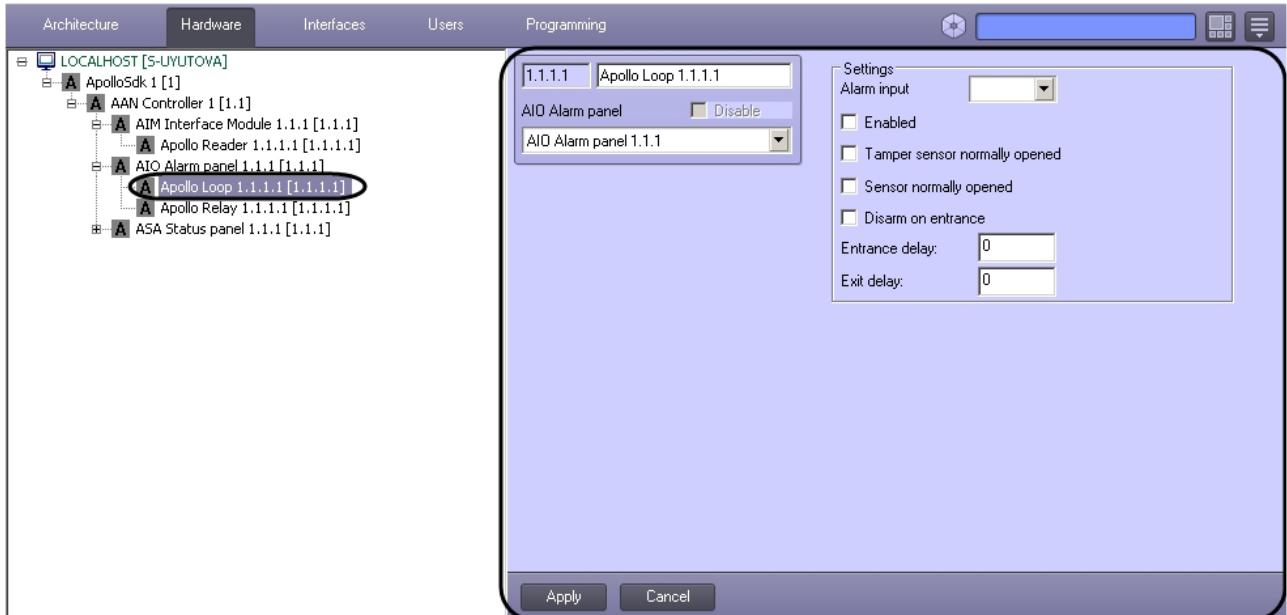
Enabling this option increases the recording configuration time.

18. If impulse reset is to be permitted, set the corresponding checkbox (17).
19. To save any changes made, click the **Apply** button (18).

This completes the configuration of the *AIM* interface module's readers.

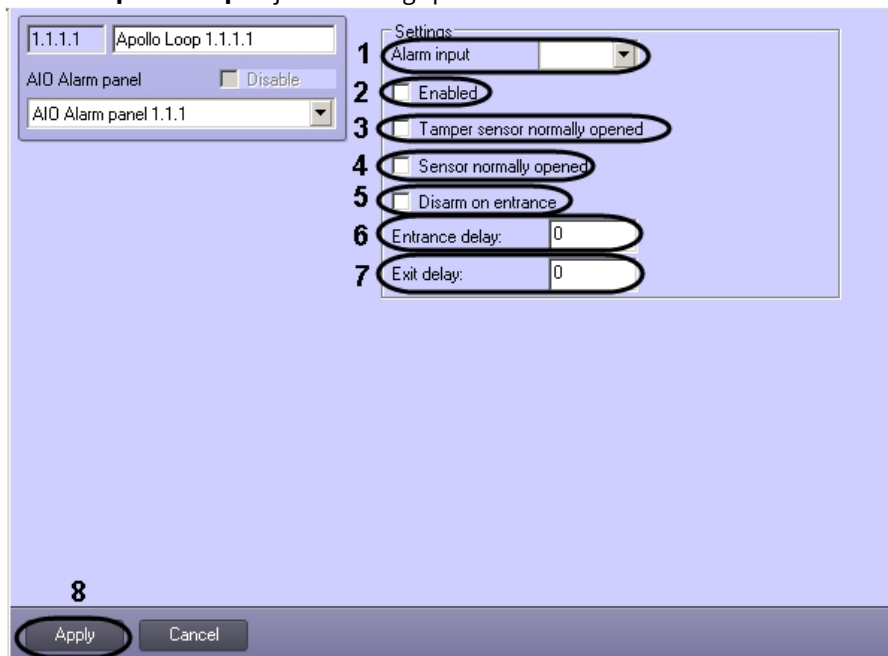
3.5.2 Configuration of the AIO alarm panel's inputs (obsolete)

The *AIO* alarm panel's inputs are configured on the settings panel of the corresponding object. An **Apollo Loop** object is created based on an **AIO Alarm Panel** object.



Alarm inputs are configured as follows:

1. Go to the **Apollo Loop** object's settings panel.



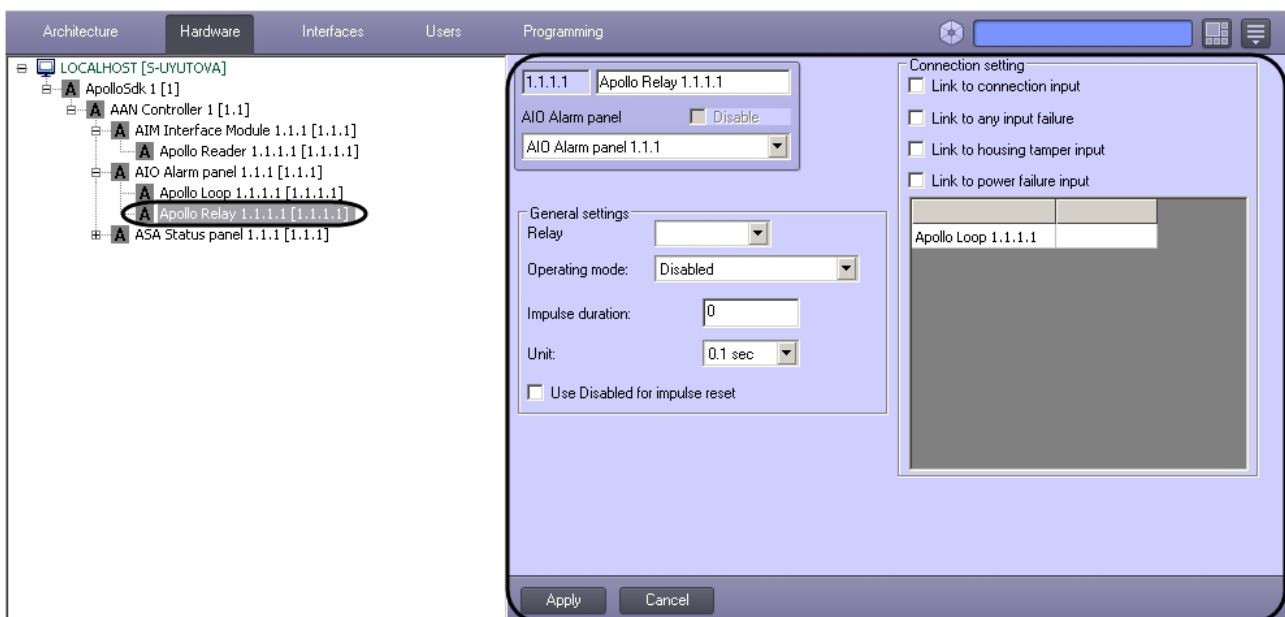
2. From the **Alarm input** dropdown list select the alarm input's index number (1).
3. Check the **Enabled** checkbox if the alarm input is operational (2).
4. Check the **Tamper sensor normally opened** checkbox if the alarm events are to be received while destroying or tampering the sensor cabinet (5).

5. Check the **Sensor normally opened** checkbox if the alarm input should be in its normal state (non-alarm state) when the contacts are open (4).
6. If the alarm input should be disarmed upon entrance, set the corresponding option (5).
7. In the **Entrance delay** field, enter the time period (in seconds) in which the alarm input will be disarmed on entrance (6).
8. In the **Exit delay** field, enter the time period (in seconds) in which somebody can exit once the alarm input has been armed (6).
9. To save any changes made, click the **Apply** button (8).

This completes the configuration of the *AIO* alarm panel's inputs.

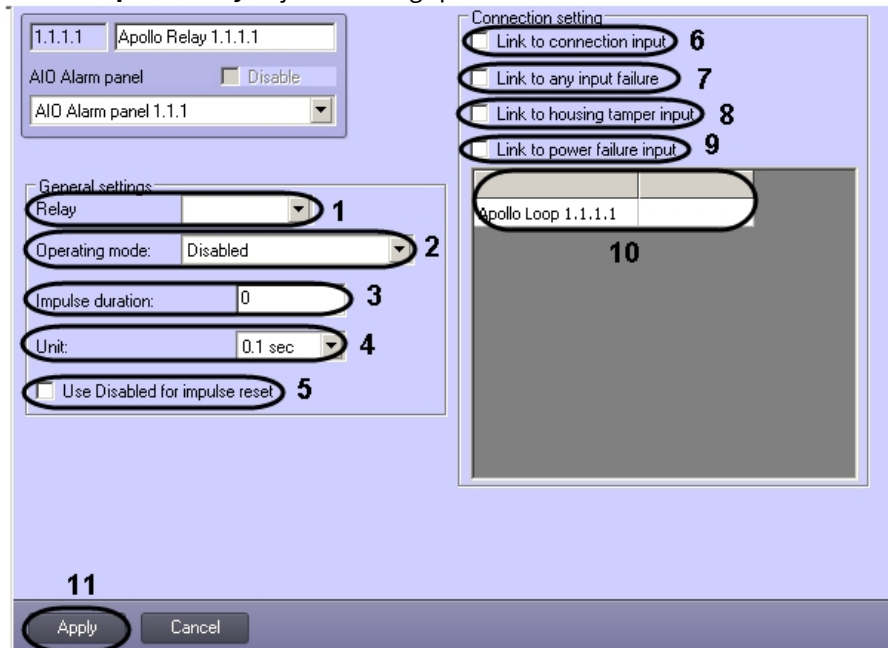
3.5.3 Configuration of the AIO alarm panel's relay (obsolete)

The *AIO* alarm panel's relay is configured on the settings panel of the corresponding object. An **Apollo Relay** object is created based on an **AIO Alarm Panel** object.



The relay is configured as follows:

1. Go to the **Apollo Relay** object's settings panel.



2. From the **Relay number** dropdown list select the relay's index number (1).
3. From the **Operating mode** dropdown list select the relay's operating mode (2).

Relay operating mode	Description
Disabled	The relay is open
Enabled	The relay is closed
Locally linked to inputs	The state of the relay depends on the state of the alarm inputs and inputs

4. Set the Impulse duration:
 - a. From the **Unit** dropdown list select the units of measurement for the impulse duration (4).
 - b. In the **Impulse duration** field, specify the duration (in terms of the selected units of measurement) of the impulse (3).
5. To make it possible to reset the impulse when changing the operating mode to **Disabled**, set the **Use Disabled for impulse reset** option (5).
6. Set the **Link to connection input** option if the relay should react to the state of the **Connection** input (6).
7. Set the **Link to any input failure** option if the relay should react to a fault on any of the inputs (7).
8. Set the **Link to housing tamper input** option if the relay should react to the state of the alarm panel's cabinet (8).
9. Set the **Link to power failure input** option if the relay should react to the state of the alarm panel's power supply (9).
10. Configure the interaction between the relay and the alarm inputs (10). In the **Action** column, select the alarm input states in which the relay will close.

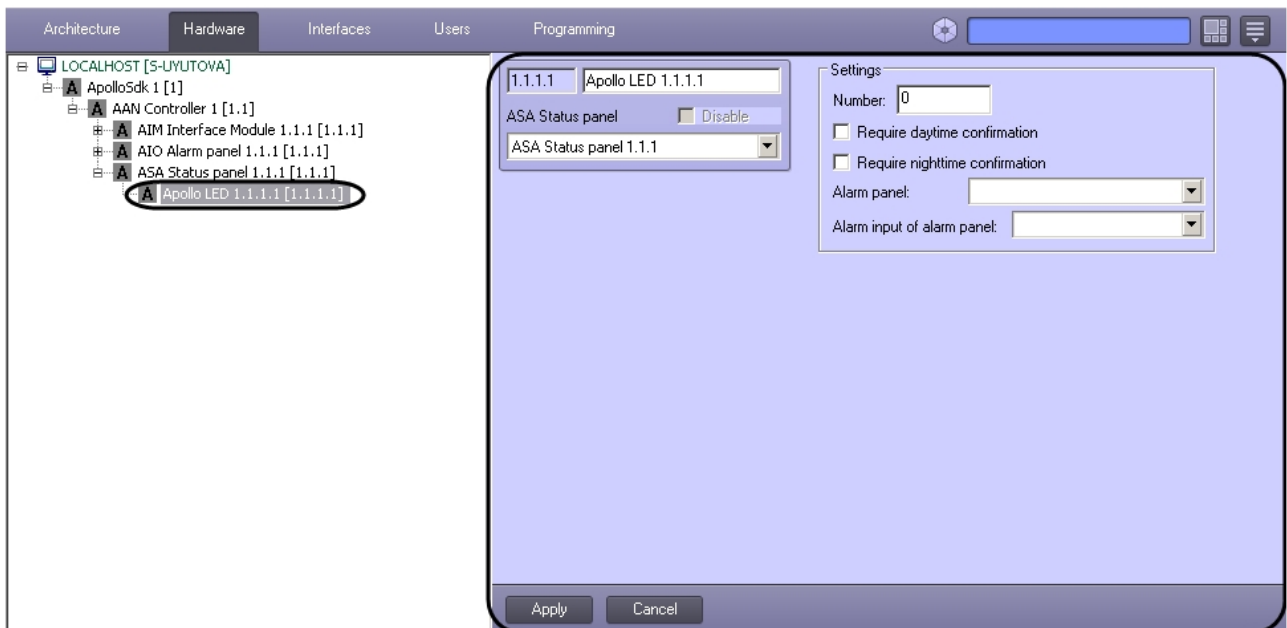
State	Description
None	The relay will be open regardless of the alarm input state
Alarm and tamper sensor	The relay will be actuated in the following cases: <ol style="list-style-type: none"> a. An alarm message is received from the alarm input. b. A message is received from the tamper sensor.
Alarm/mask and tamper sensor	The relay will be actuated in the following cases: <ol style="list-style-type: none"> a. An alarm message is received from the alarm input. b. An alarm message is received from the alarm input when it is disarmed. c. A message is received from the tamper sensor.
Alarm/mask, tamper sensor and failure	The relay will be actuated in the following cases: <ol style="list-style-type: none"> a. An alarm message is received from the alarm input. b. An alarm message is received from the alarm input when it is disarmed. c. A message is received from the input's tamper sensor. d. A fault message is received.

11. To save any changes made, click the **Apply** button (**11**).

This completes the configuration of the *AIO* alarm panel's relay.

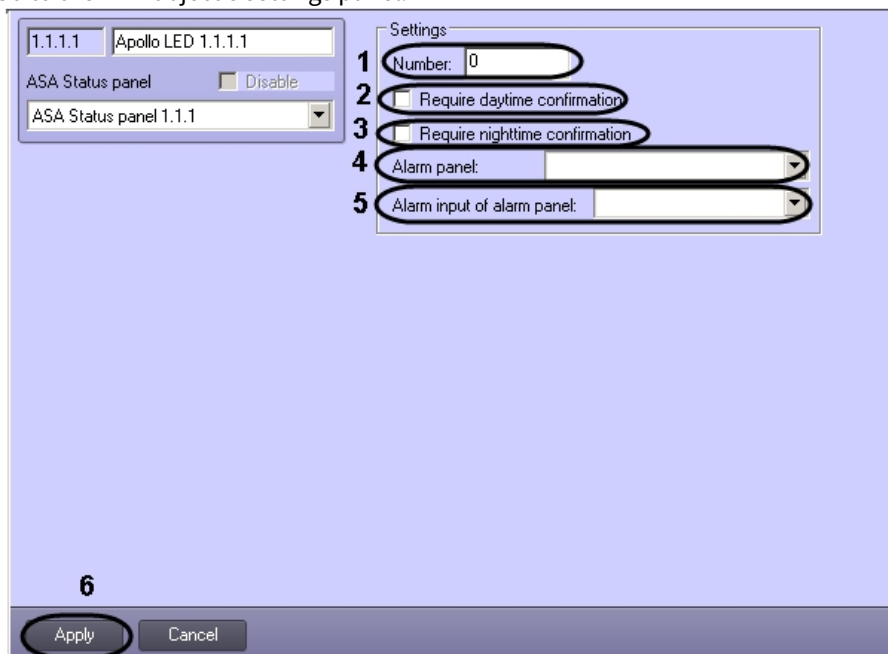
3.5.4 Configuration of an ASA status panel's sensors (obsolete)

An ASA status panel's sensors are configured on the **LED** object's settings panel. This object is created based on an **ASA Status Panel** object.



The ASA status panel's sensors are configured as follows:

1. Go to the **LED** object's settings panel.



2. In the **Number** field, enter the sensor's index number (1).
3. If alarms must be confirmed during the day, check the **Require daytime confirmation** option (2).
4. If alarms must be confirmed at night, check the **Require nighttime confirmation** option (3).
5. Select the alarm panel associated with this sensor from the corresponding dropdown list (4).
6. Select the alarm panel input associated with this sensor from the corresponding dropdown list (5).
7. To save any changes made, click the **Apply** button (11).

This completes the configuration of the AIO alarm panel's relay.

4 Working with the ApolloSDK integration module (obsolete)

4.1 General information about working with the ApolloSDK Module (obsolete)

The following interface objects are used to work with the *ApolloSDK* integration module:

1. **Map.**
2. **Event Log.**
3. **Pass and ID Office.**
4. **Event Manager.**

Information about configuring these interface objects is presented in the following [Axxon PSIM Software System documents](#). [Administrator's Guide](#), [Access Manager Module Settings and Operation Guide](#), and [Event Manager Module Settings and Operation Guide](#).

How to work with interface objects is described in detail in [Axxon PSIM Software System. Operator's Guide](#).

4.2 Managing an AAN controller (obsolete)

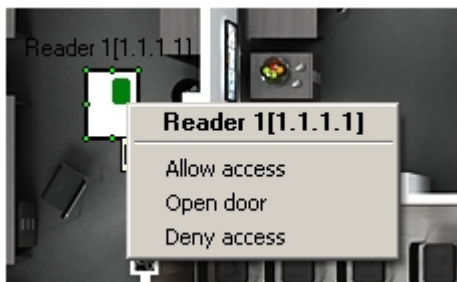
An *AAN* controller is managed in the interactive **Map** window using the corresponding object's menu:



To reset the controller's settings, select **Restart controller** in the **AAN Controller** object's menu.

4.3 Managing the AIM interface module's readers (obsolete)

The *ApolloSDK* integration module's readers are managed in the interactive **Map** window using the **Reader** object's menu.



Description of the **Reader** object's menu commands is given in the table.

Menu command	Function
Allow access	Grants access

Menu command	Function
Open door	Opens the door for the "access time" period
Deny access	Denies access

4.4 Managing the AIO alarm panel's inputs (obsolete)

The *AIO* alarm panel's inputs are managed in the interactive **Map** window using the **Alarm Input** object's menu.



Description of the **Alarm Input** object's menu commands is given in the table.

Menu command	Function
Arm	Arms the alarm input
Disarm	Disarms the alarm input

4.5 Managing the AIO alarm panel's relay (obsolete)

The *AIO* alarm panel's relay is managed in the interactive **Map** window using the **Relay** object's menu.



Select **Send impulse** in the **Relay** object's menu to send an impulse.