

Инструкция использования iRidi SCADA-BMS

С базовыми возможностями iRidi Studio можно ознакомиться по адресу [iRidi studio](#).

Содержание

Оглавление

1. iRidi SCADA-BMS	7
1.1. Общая характеристика iRidi SCADA-BMS	7
1.2. ПО iRidi SCADA Server	9
1.3. Позиции для заказа iRidi SCADA.....	11
1.4. ПО iRidi SCADA App (Client/ APM).....	14
1.5. Типы клиентов	15
1.6. Позиции для заказа iRidi SCADA, рабочие места (клиенты).....	16
1.7. ПО iRidi OPC DA Connector	18
1.8. iRidi SCADA Drivers Pack	19
1.9. ПО iRidi SCADA Proxy	20
1.10. Техническая поддержка	21
1.11. Обучение	22
2. SCADA-проект.....	23
2.1. Создание SCADA-проекта	23
2.2. Настройки SCADA проекта	25
2.2.1. Настройки сервера	26
2.2.2. Настройки подключения к внешней базе данных.....	29
2.2.3. Настройки Telegram-чатов	31
2.2.4. Настройки журнала	33
2.2.5. Настройки резервирования	35
2.3. Загрузка проекта SCADA сервера	37
3. Добавление драйвера.....	38
3.1. Добавление драйвера в SCADA-проект	38
3.2. Настройка параметров подключения	42
3.2.1. Bus77 Network	43
3.2.2. iRidi Server	44
3.2.3. iRidi SCADA Server	48
3.2.4. Modbus RTU Network	50
3.2.5. Modbus TCP Network.....	52
3.2.6. MQTT.....	54
3.2.7. OPC DA Connector	56

3.2.8. OPC UA Server	57
3.2.9. Ping Network	60
3.2.10. SNMP	61
3.2.11. BACnet Network	62
4. Работа с тегами	66
4.1. Описание	66
4.2. Создание SCADA-тегов	67
4.3. Свойства SCADA-тегов	68
4.4. Modbus TCP Slave	71
4.5. Панель инструментов SCADA-тегов	74
4.5.1. Создание папок тегов	74
4.5.2. Удаление тегов и папок	75
4.5.3. Перемещение тегов в дереве	76
4.5.4. Поиск тегов	77
4.5.5. Работа с каналами управления, каналами обратной связи и токенами драйвера	78
5. Создание SCADA-клиентов	81
5.1. Создание проекта SCADA-клиента	81
5.2. Связь существующего проекта SCADA-клиента с SCADA-проектом	85
5.3. Загрузка проекта SCADA клиента	87
5.4. Добавление и сохранение адреса сервера в SCADA клиенте	88
5.5. Диспетчеризация	89
5.5.1 Для запуска Клиента - Команда SCADA Client. Открыть доп. SCADA Client (только ПК).	89
5.5.2. Для закрытия Клиента - Команда SCADA Client. Закрыть доп. SCADA Client (только ПК).	90
6. Аварийные сообщения (Алармы)	91
6.1. Описание	91
6.2. Группы аварийных сообщений (группы алармов)	92
6.2.1. Создание группы аварийных сообщений (алармов)	93
6.2.2. Удаление группы аварийных сообщений (алармов)	94
6.2.3. Клонирование группы аварийных сообщений (алармов)	95
6.2.4. Поиск по группам аварийных сообщений (алармов)	96
6.2.5. Импорт и экспорт групп аварийных сообщений	97

6.3. Настройки группы аварийных сообщений (алармов)	98
6.4. Действия	100
6.4.1. Описание основных групп действий	100
6.4.2. Создание группы действий алармов	105
6.4.3. Настройка группы действий алармов	106
6.4.4. Клонирование группы действий алармов	107
6.4.5. Удаление группы действий алармов	108
6.4.6. Как сделать снимок с камеры и отправить его в Telegram	109
6.5. Настройка системы аварийных сообщений (алармов).....	114
6.6. SCADA Alarm Label	118
6.6.1. Описание	118
6.6.2. Принцип работы	119
6.6.3. Создание SCADA Alarm Label	120
6.6.4. Параметры элемента.....	121
6.6.5. Привязка к оборудованию	123
6.7. SCADA Alarm View.....	124
6.7.1. Описание	124
6.7.2. Создание SCADA Alarm View	125
6.7.3. Параметры элемента.....	126
6.7.4. Настройка SCADA Alarm View.....	126
6.8. SCADA Alarm Indicator.....	133
6.8.1. Описание	133
6.8.2. Принцип работы	134
6.8.3. Создание Alarm Indicator	135
6.8.4. Параметры элемента.....	136
6.8.5. Настройка работы элемента SCADA Alarm Indicator.....	138
6.8.6. Программирование элемента SCADA Alarm Indicator	141
7. Авторизация.....	142
7.1. Роли и доступ	142
7.2. Создание ролей пользователей.....	143
7.2.1. Роли пользователей	143
7.2.2. Пользователи	145
7.3. Редактирование ролей и пользователей	146

7.3.1. Редактирование роли	146
7.3.2. Редактирование пользователей	147
7.4. Настройка доступа к SCADA-тегам	148
7.5. Настройка доступа к графическим элементам в проекте SCADA-клиента	149
8. Архивация тегов	151
8.1. Описание	151
8.2. Настройка базы данных и тегов	152
8.3. Создание Line Chart (Beta)	154
8.4. Добавление тегов	155
8.5. Настройка внешнего вида Line Chart (Beta)	159
8.5.1. Основные параметры элемента	159
8.5.2. Настройки элемента	160
8.5.3. Расширенные настройки	161
9. Использование Java Script в SCADA-сервере	167
9.1. Общие положения	167
9.2. Использование API	168
9.2.1. Создать JavaScript файл в iRidi Studio:	168
9.2.2. Подключить готовый JavaScript файл в iRidi Studio	169
9.2.3. Выполнение скрипта	170
9.2.4. Правила выполнения скрипта	171
9.2.5. Пространство имен	172
9.3. Методы js	173
9.3.1. Глобальный слушатель	173
9.3.2. Логирование	174
9.3.3. Телеграм	175
9.3.4. Установка и получение переменных	176
9.3.5. Таймеры	177
9.3.6. Журнал событий	178
9.3.7. Получение истории тегов	180
9.3.8. Подключение к DataBase	182
9.3.9. Алармы	183
9.3.10. Диспетчеризация склада клиента	187
10. Использование Java Script в SCADA-клиенте	189

10.1. Общие положения	189
10.2. Использование API	190
10.2.1. Создать JavaScript файл в iRidi Studio:	190
10.2.2. Подключить готовый JavaScript файл в iRidi Studio	191
10.2.3. Выполнение скрипта	192
10.2.4. Правила выполнения скрипта	193
10.2.5. Пространство имен	194
10.3. Команда ScriptCall().....	195
11. Расписание	196
11.1. Описание	196
11.2. Типы расписаний.....	197
11.3. Настройки событий расписания	197
12. Правила.....	198
12.1. Описание	198
12.2. События срабатывания, действия при срабатывании, исключаяющие условия	200
12.3. Настройка редактируемых правил.....	203
12.4. Использование редактируемых правил в проекте.....	204
13.1. Описание	206
13.2. Меню iRidi SCADA Coordinator	208
13.3. Создание проекта iRidi SCADA Coordinator	209
13.4. Работа в iRidi SCADA Coordinator	210
14.1. Описание	211
14.2. Меню iRidi OPC DA Connector	212
14.3. Создание проекта в iRidi OPC DA Connector.....	214
14.3. Добавление OPC DA сервера в проект	214
14.4. Выбор тегов для работы в iRidi SCADA.....	219
14.5. Добавление тегов OPC DA в проект SCADA.....	222

1. iRidi SCADA-BMS

1.1. Общая характеристика iRidi SCADA-BMS

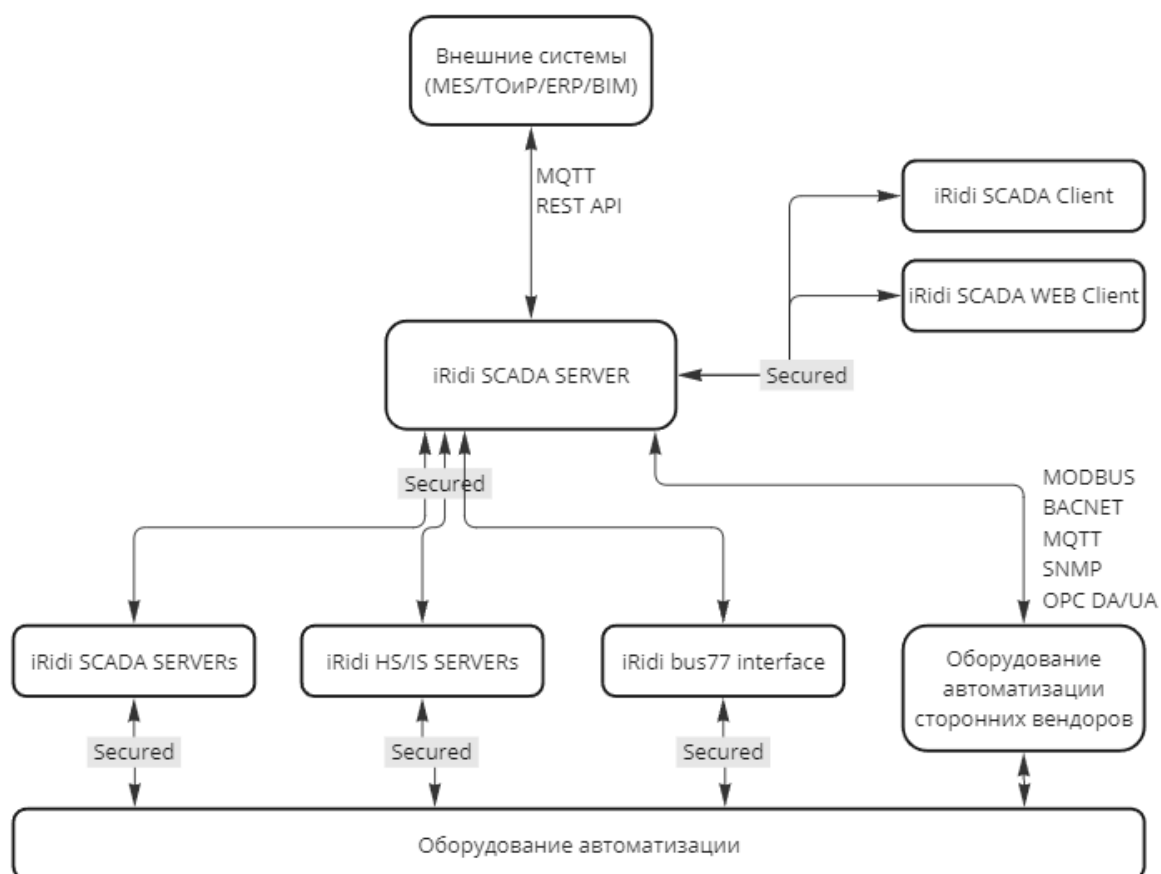
SCADA-система - ключевой компонент экосистемы автоматизации зданий iRidi. Он позволяет свести все данные здания в единое информационное пространство, обрабатывать эти данные, сохранять в базу данных и представлять пользователям системы в различном виде.

Ключевые преимущества системы:

- защищенная система связи между внешними и внутренними компонентами системы - TCP стек, протокол iRidi UA, криптозащита на базе ГОСТ 34.12-2018 Кузнечик (256 бит);
- платформенное/ аппаратное ПО:
 - сервер SCADA: windows x64,
 - клиент SCADA: windows x64;
- бесшовная интеграция с глобальным(облако), средним(контроллеры) уровнем автоматизации и диспетчеризации;
- единая среда разработки для всех уровней;
- разработано и зарегистрировано в РФ (реестр РФ ПО, ФСТЭК);
- неограниченные возможности интеграции (через интеграционные сервера iRidi - 70+ драйверов);
- неограниченные графические возможности (наследие i3Pro)

SCADA-система состоит из нескольких взаимосвязанных программных компонентов.

Схема связи компонентов представлена на рисунке.



1.2 ПО iRidi SCADA Server

SCADA-сервер осуществляет взаимодействие с различными уровнями системы автоматизации - внешними системами, контроллерами среднего уровня, низовым оборудованием. Полученные данные обрабатываются с помощью встроенных средств логико-математической обработки (скрипты, блочная логика, выражения-формулы). В процессе обработки данные нормализуются, агрегируются, а затем записываются в базу данных для дальнейшей анализа и обработки. Одновременно система контролирует поступающие данные, проверяя их на валидность и соответствие заданным уставкам (система алармов). Также система обеспечивает управление оборудованием и системами здания по расписанию.

Характеристики системы:

Емкость системы:

- до 4 000 000 тегов;
- до 500 сетевых клиентов на сервер.

Поддерживаемые драйвера:

- Bus77 IP/CAN;
- Modbus TCP/RTU/RTU_over_TCP;
- MQTT (req/pub);
- SNMP (v.1);
- PING;
- BACNET IP (base);
- VirtualTag.

Логическая обработка:

- JavaScript ECMAScript E5/E5.1 с обновленной семантикой ES2015+.

Интеграция с внешними системами:

- REST API (https);
- MQTT;
- СУБД PostgreSQL.

Система хранения данных:

- СУБД PostgreSQL (PostgreSQL PRO);
- СУБД iRidi TS.

Базовый функционал:

- нет привязки к облакам;
- ввод-вывод (драйвера) - 8 базовых и 70+ расширенных;
- продвинутая система алармов;
- расписания.

Кибербезопасность:

- ролевая модель доступа;
- зашифрованный канал обновления и загрузки проектов;
- лог действий пользователя;
- лог обновления проектов;
- интеграция в PSIM системы;

- совместимость с Kaspersky/ УЦСБ/Positive Technologies.

Резервирование:

- горячее и холодное резервирование по схеме 1+1, 1+N.

Лицензирование:

- USB ключ Guardant;
- без привязки к облакам.

Операционные системы:

- Windows 10;
- Debian 12;
- Astra Linux Common Edition (с предварительной настройкой контейнеризации с помощью Docker).

Работа в виртуальных средах:

- Virtual Box.

1.3. Позиции для заказа iRidi SCADA

iRidi SCADA Local	
Применяется для простых объектов, где клиент и сервер iRidi SCADA работают на одном ПК. Макс. кол-во полевых тегов: 500 (дополнительно 500 виртуальных). Визуализация: WEB- dashboards/PRO visu. Драйвера: bus77, modbus TCP/RTU master, iRidi Protocol СУБД: sqlite (глубина 3 месяца) Интеграция с верх.уровнем: iRidi Protocol Резервирование: нет Докачка архивных данных с полевых серверов iRidi: нет.	
Код заказа	Описание
ir-scada-lc-50	iRidi SCADA Local 50 ext tags, x1 virtual tags
ir-scada-lc-100	iRidi SCADA Local 100 ext tags, x1 virtual tags
ir-scada-lc-200	iRidi SCADA Local 200 ext tags, x1 virtual tags
ir-scada-lc-500	iRidi SCADA Local 500 ext tags, x1 virtual tags
iRidi SCADA Base	
Клиент-серверная система без резервирования сервера. Макс. кол-во полевых тегов: 50к (дополнительно 50к виртуальных). Визуализация: только WEB- dashboards. Драйвера: bus77, modbus TCP/RTU master, iRidi Protocol, OPC UA/DA, Ping СУБД: sqlite (глубина 6 месяцев), PostgreSQL Интеграция с верх.уровнем: iRidi Protocol, Modbus TCP Slave, REST API, MQTT Publisher Резервирование: нет Докачка архивных данных с полевых серверов iRidi: нет	
ir-scada-bs-150	iRidi SCADA Base 150 ext tags, x1 virtual tags
ir-scada-bs-300	iRidi SCADA Base 300 ext tags, x1 virtual tags
ir-scada-bs-500	iRidi SCADA Base 500 ext tags, x1 virtual tags
ir-scada-bs-1k	iRidi SCADA Base 1k ext tags, x1 virtual tags

ir-scada-bs-1,5k	iRidi SCADA Base 1,5k ext tags, x1 virtual tags
ir-scada-bs-3k	iRidi SCADA Base 3k ext tags, x1 virtual tags
ir-scada-bs-5k	iRidi SCADA Base 5k ext tags, x1 virtual tags
ir-scada-bs-15k	iRidi SCADA Base 15k ext tags, x1 virtual tags
ir-scada-bs-50k	iRidi SCADA Base 50k ext tags, x1 virtual tags
iRidi SCADA Standart	
Клиент-серверная система с функцией резервирования сервера. Макс. кол-во полевых тегов: 150к (дополнительно 150к виртуальных) Визуализация: WEB- dashboards/PRO visu. Драйвера: bus77, modbus TCP/RTU master, iRidi Protocol, OPC UA/DA, Ping, bacnet IP, SNMP, MQTT, RTSP СУБД: PostgreSQL Интеграция с верх.уровнем: iRidi Protocol, Modbus TCP Slave, REST API, MQTT Publisher Резервирование: есть на 1 сервер Докачка архивных данных с полевых серверов iRidi: да	
ir-scada-st-3k	iRidi SCADA Standard 3k ext tags, x2 virtual tag
ir-scada-st-15k	iRidi SCADA Standard 15k ext tags, x2 virtual tag
ir-scada-st-50k	iRidi SCADA Standard 50k ext tags, x2 virtual tag
ir-scada-st-150k	iRidi SCADA Standard 150k ext tags, x2 virtual tag
Ridi SCADA Enterprise	
Распределенная, многоуровневая система для построения распределенной диспетчерской системы с функцией расширенного резервирования. Максимальное количество полевых тегов: 1000к (дополнительно 2000к виртуальных). Визуализация: WEB-dashboards/PRO visu. Драйвера: bus77, modbus TCP/RTU master, iRidi Protocol, OPC UA/DA, Ping, bacnet IP, SNMP, MQTT, RTSP СУБД: PostgreSQL, TimeSeries RTDB Интеграция с верх.уровнем: iRidi Protocol, Modbus	

TCP Slave, REST API, MQTT Publisher Резервирование: есть N серверов Докачка архивных данных с полевых серверов iRidi: д	
ir-scada-en-50k	iRidi SCADA Enterprise 50k ext tags, x2 virtual tags
ir-scada-en-150k	iRidi SCADA Enterprise 150k ext tags, x2 virtual tags
ir-scada-en-500k	iRidi SCADA Enterprise 500k ext tags, x2 virtual tags
ir-scada-en-1000k	iRidi SCADA Enterprise 1000k ext tags, x2 virtual tags

1.4. ПО iRidi SCADA App (Client/ APM)

SCADA-клиент предназначен для создания автоматизированных рабочих мест (APM) для обслуживающего персонала (эксплуатационные службы, субподрядчики, инженеры обслуживания). Он отображает состояние инженерных систем здания/объекта в графическом виде, позволяя интерактивно взаимодействовать с системами. С его помощью можно изменять уставки, настраивать расписания и просматривать историю изменений.

Работает на различных операционных системах и аппаратных платформах:

- ПК настольные (x64/ARM с ОС Linux/Windows/MAC):
 - "толстый" клиент;
 - WEB приложение в браузере (WEB- dashboards);
- Планшеты:
 - "толстый" клиент для iOS/Android/Linux;
 - WEB приложение в браузере (WEB- dashboards);
- Смартфоны:
 - "толстый" клиент для iOS/Android/Linux;
 - WEB приложение в браузере (WEB- dashboards);
- Панели настенные/ настольные:
 - "толстый" клиент для iOS/Android/Linux;
 - WEB приложение в браузере (WEB- dashboards).

Графический функционал:

- унаследовал все графические возможности i3Pro;
- добавлены специфичные для SCADA-компоненты: AlarmView, AlarmLabel, SCADA Trend;
- работа на разрешении 8к;
- мультимониторность.

Кибербезопасность:

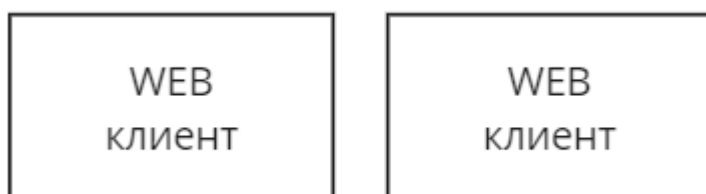
- нет привязки к облакам;
- ролевая модель доступа;
- усиленная парольная защита;
- автовыход по таймауту;
- контроль целостности проектов;
- контроль целостности исполняемых модулей;
- зашифрованный канал обновления и загрузки проектов;
- шифрование проектов;
- лог действий пользователя;
- лог обновления проектов и компонентов системы;
- совместимость с Kaspersky/ УЦСБ/Positive Technologies.

1.5. Типы клиентов



Нативное приложение устанавливаемое на целевую платформу Особенности:

- необходимо устанавливать приложение;
- высокое быстродействие;
- расширенные графические возможности.

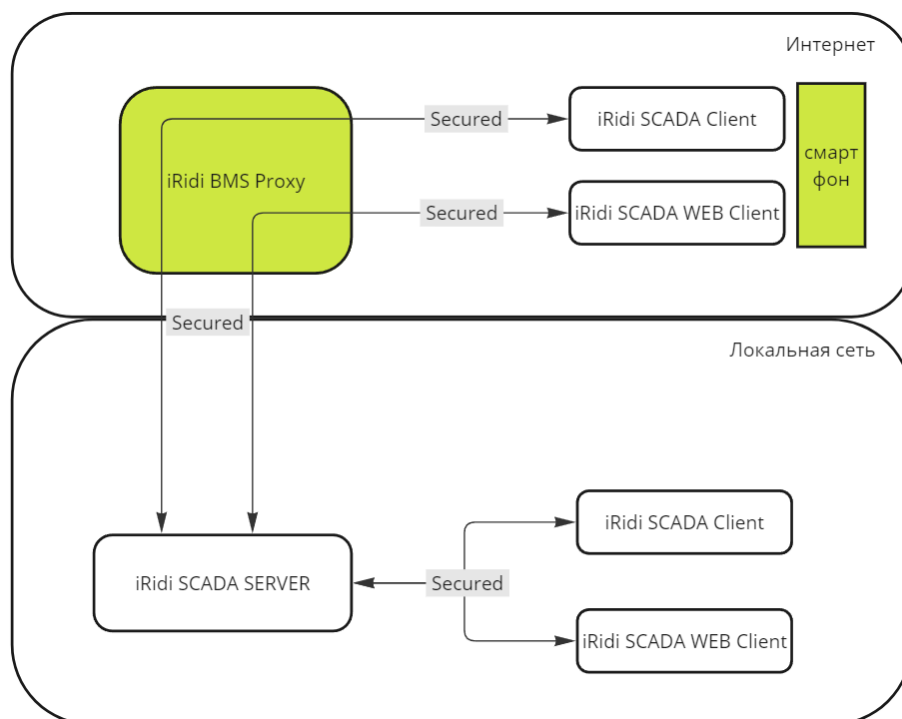


WEB в браузере

Особенности:

- работа из браузера;
- базовые графические возможности (только виджеты).

Схема связи компонентов представлена на рисунке.



1.6. Позиции для заказа iRidi SCADA, рабочие места (клиенты)

Рабочие места (АРМы оператора) поставляются в 2-х вариантах: 1. View only - доступны мнемосхемы только на просмотр, не позволяет изменять значения тегов или режимы с графического интерфейса. 2. Full Control - позволяет изменять значения тегов или режимы с графического интерфейса. При этом в обоих вариантах учитываются настройки системы пользовательского доступа. В лицензии учитывается кол-во одновременно подключенных клиентов (технология конкурентных подключений).	
“Толстые”клиенты	Клиенты с применением нативных приложений под Windows, Linux, iOS, Android (кастомная графика)
Код заказа	Описание
ir-scada-netpro-1-vo	1 соединение i3 pro scada View Only
ir-scada-netpro-5-vo	5 соединение i3 pro scada View Only
ir-scada-netpro-10-vo	10 соединение i3 pro scada View Only
ir-scada-netpro-1-fc	1 соединение i3 pro scada Full Control
ir-scada-netpro-5-fc	5 соединение i3 pro scada Full Control
ir-scada-netpro-10-fc	10 соединение i3 pro scada Full Control
WEB-dashboards	Клиенты с подключением через WEB браузер (простая графика - виджеты)
ir-scada-netweb-1-vo	1 соединение web dashboards scada View Only
ir-scada-netweb-5-vo	5 соединение web dashboards scada View Only
ir-scada-netweb-10-vo	10 соединение web dashboards scada View Only
ir-scada-netweb-1-fc	1 соединение web dashboards scada Full Control
ir-scada-netweb-5-fc	5 соединение web dashboards scada Full Control

ir-scada-netweb-10-fc	10 соединение web dashboards scada Full Control
-----------------------	---

1.7. ПО iRidi OPC DA Connector

Компонент позволяет заводить в систему данные с OPC DA серверов. Устанавливается на ПК с Windows и взаимодействует с локальными или удалёнными (сетевыми) OPC DA серверами.

Базовые параметры:

- до 100к OPC DA тегов;
- до 100 OPC DA серверов;
- до 10 одновременных подключений с iRidi SCADA серверов.

Позиции для заказа:

Код заказа	Описание
ir-scada-opcda-1k	Ridi OPC DA Connector 1k
ir-scada-opcda-5k	Ridi OPC DA Connector 5k
ir-scada-opcda-10k	Ridi OPC DA Connector 10k
ir-scada-opcda-100k	Ridi OPC DA Connector 100k

1.8. iRidi SCADA Drivers Pack

Расширяет коммуникационные возможности младших версий SCADA-системы.

Варианты:

iRidi SCADA Driver pack 1 :

- OPC UA Client
- OPC DA
- Ping

Ridi SCADA Driver pack 2 :

- bacnet IP
- SNMP
- MQTT
- RTSP

Позиции для заказа:

Код заказа	Описание
ir-scada-dp-0	iRidi SCADA Driver pack 0 (bus77, modbus TCP/RTU master)
ir-scada-dp-1	iRidi SCADA Driver pack 1 (OPC UA Client, OPC DA, Ping, iRidi Protocol)
ir-scada-dp-2	iRidi SCADA Driver pack 2 (bacnet IP, SNMP, MQTT, RTSP)

1.9. ПО iRidi SCADA Proxy

Позволяет организовать подключение сетевых клиентов через интернет. Данные между клиентом и сервером шифруются по ГОСТ. Возможна поставка в виде SaaS решения или в виде OnPremise.

Позиции для заказа:

Код заказа	Описание
ir-scada-proxy-1	1 прокси соединение i3 pro scada (интернет доступ), доп. к ir-scada-netpro-1-fc
ir-scada-proxy-5	5 прокси соединений i3 pro scada (интернет доступ), доп. к ir-scada-netpro-5-fc
ir-scada-proxy-10	10 прокси соединение i3 pro scada (интернет доступ), доп. к ir-scada-netpro-10-fc

1.10. Техническая поддержка

Компания предоставляет техническую поддержку по системе в различном объеме, в зависимости от целей и задач заказчика. Уровни технической поддержки по SCADA-системе:

Бесплатная техническая поддержка (free);

Пакет базовой технической поддержки;

Пакет расширенной технической поддержки;

Пакет эксклюзивной технической поддержки, доступна для версий Standart и Enterprise.

Позиции для заказа:

Код заказа	Описание
ir-scada-tp-free	Бесплатная техническая поддержка (free)
ir-scada-tp-base	Пакет базовой технической поддержки на 1 год
ir-scada-tp-exp	Пакет расширенной технической поддержки на 1 год
ir-scada-tp-super	Пакет эксклюзивной технической поддержки на 1 год, доступна для версий Standart и Enterprise

1.11. Обучение

Компания iRidi предлагает широкий спектр обучающих материалов и курсов по SCADA-системе и смежным областям. С ними можно ознакомиться на нашем [сайте](#).

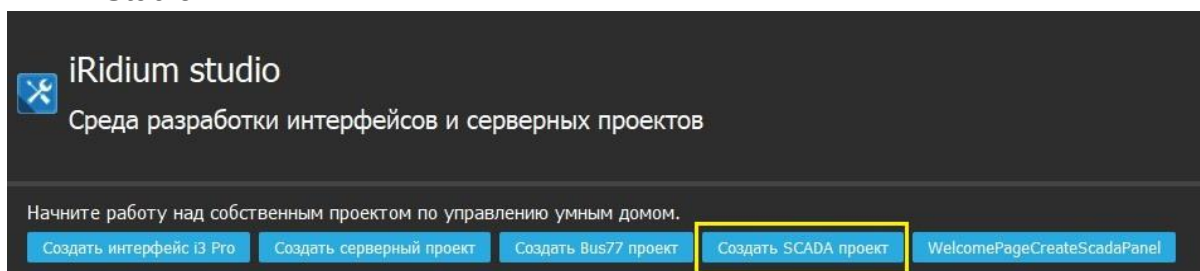
2. SCADA-проект

Проект SCADA (*.irscada) - это набор инструментов для настройки логики взаимодействия SCADA Server с оборудованием автоматизации.

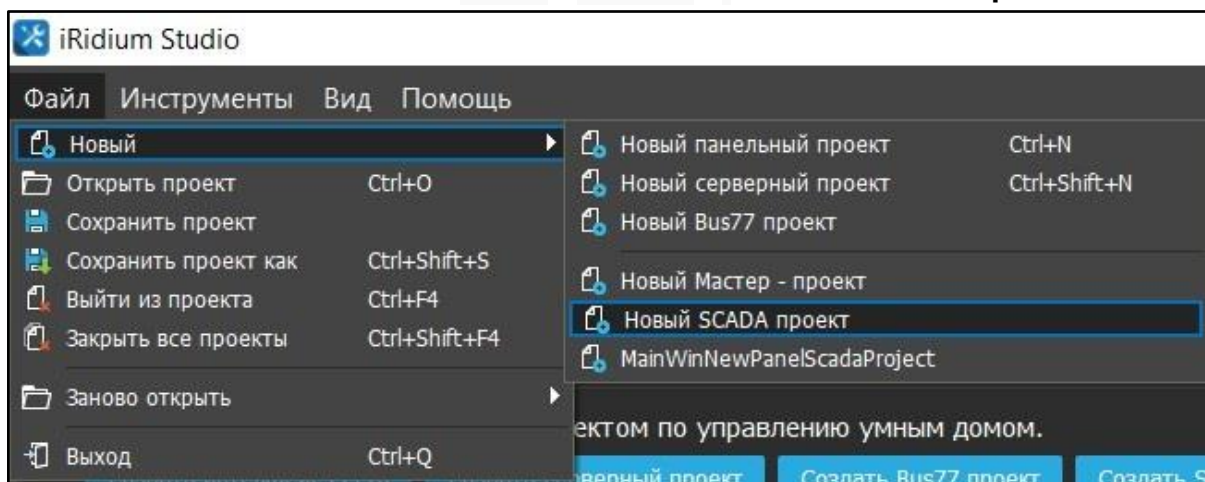
2.1. Создание SCADA-проекта

Для создания нового проекта при запуске iRidi Studio можно применить один из следующих способов:

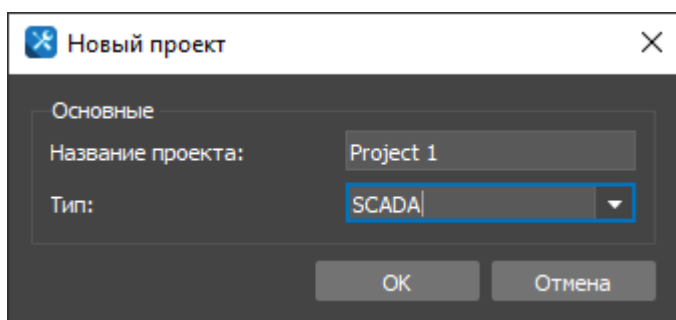
- Нажать кнопку **Создать SCADA проект** на стартовой странице iRidi Studio.



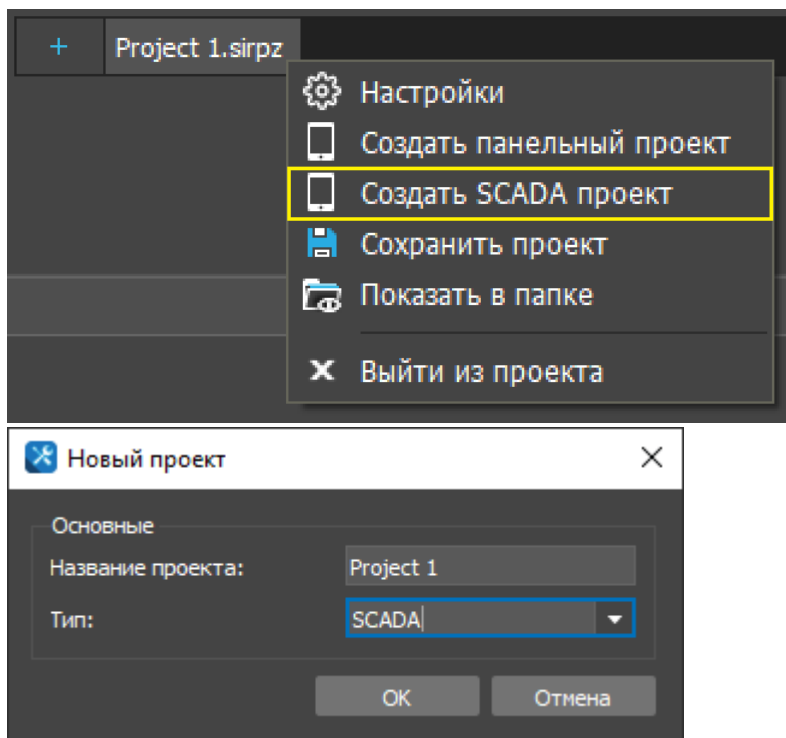
- Воспользоваться меню Файл → Новый → **Новый SCADA проект**.



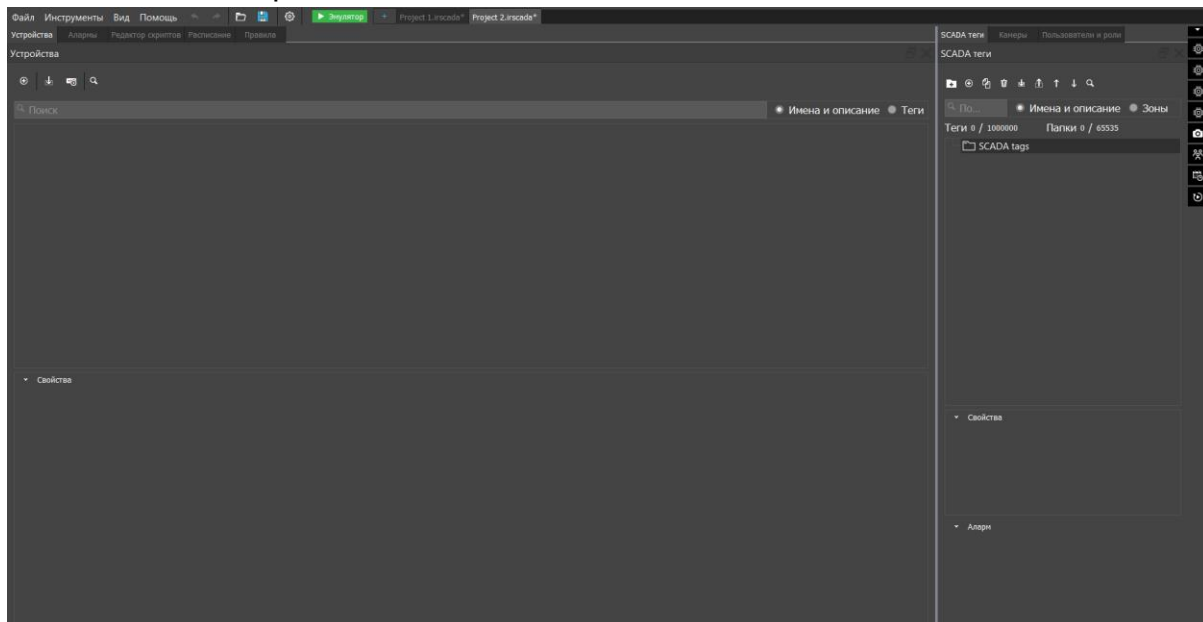
- Нажать кнопку добавления проекта на панели управления проектами. Выбрать в меню типа проекта - SCADA.



- Воспользоваться контекстным меню серверного проекта (*.sirpz).



После ввода имени и подтверждения создания в рабочей области будут открыты вкладки SCADA-проекта:

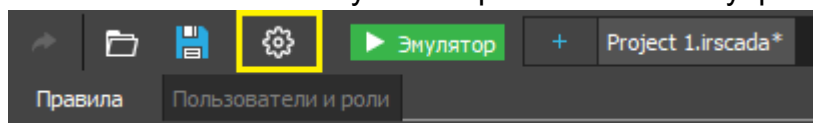


2.2. Настройки SCADA проекта

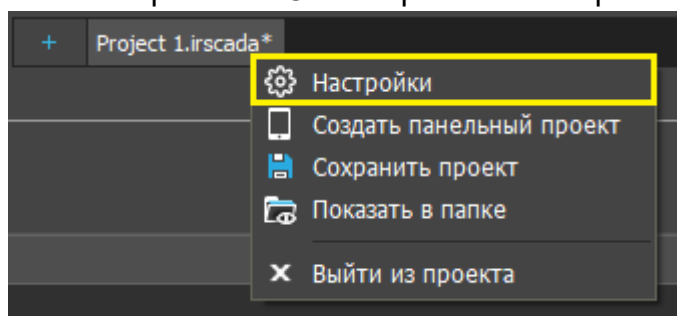
Настройки проекта включают в себя подключения к внешней базе данных, список telegram-чатов и свойства журналов.

Окно настроек SCADA-проекта можно открыть одним из следующих способов:

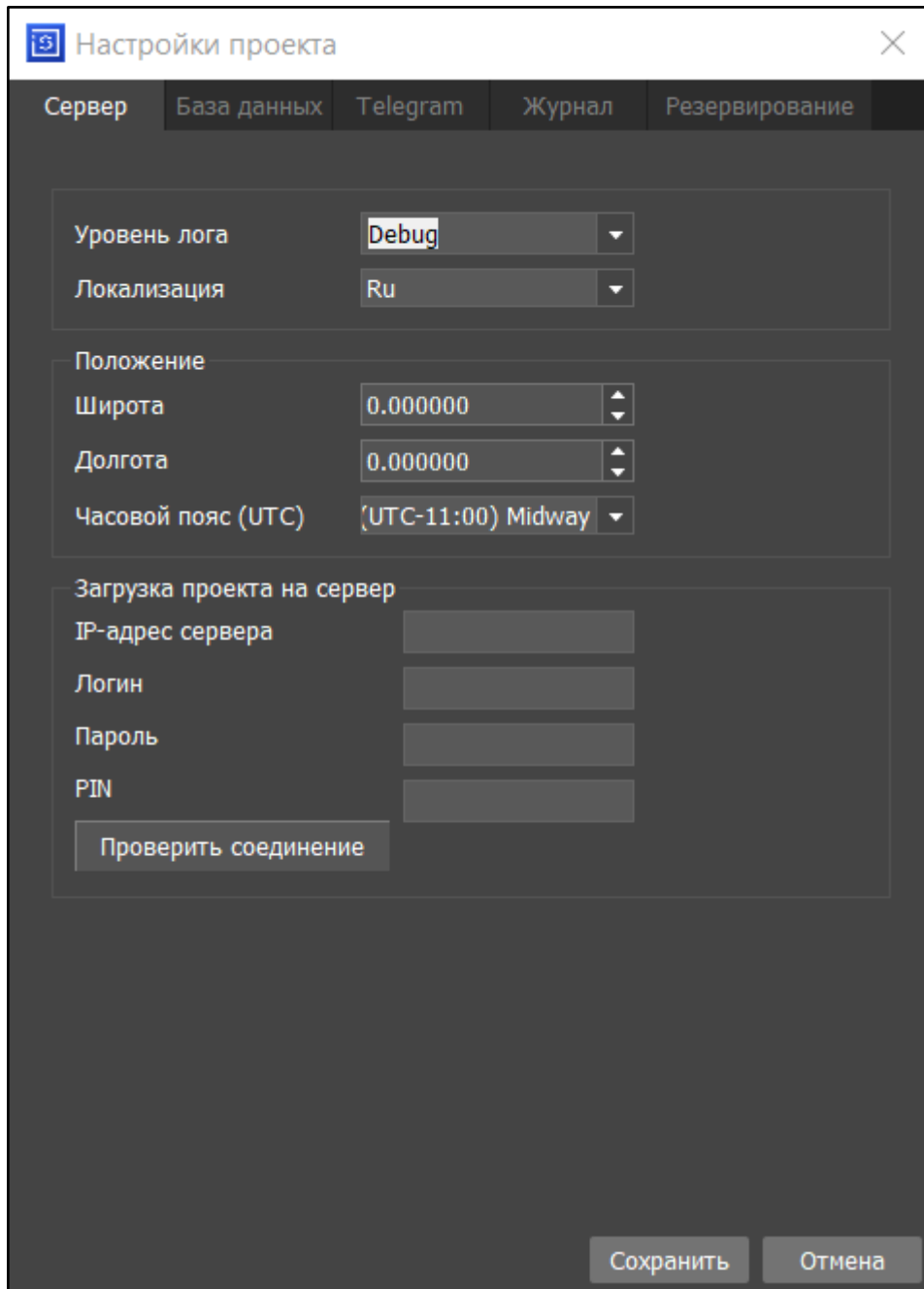
- Нажать на иконку "шестеренка" в панели управления iRidi Studio.



- Вызвать контекстное меню кликом правой клавиши мыши по вкладке открытого SCADA-проекта и выбрать пункт "Настройки".



2.2.1. Настройки сервера



The screenshot shows a software window titled "Настройки проекта" (Project Settings) with a close button in the top right corner. Below the title bar is a tabbed interface with four tabs: "Сервер" (Server), "База данных" (Database), "Telegram", and "Журнал" (Log), followed by a "Резервирование" (Reservation) button. The "Сервер" tab is currently active. The settings are organized into three sections:

- Общие (General):** Contains two dropdown menus: "Уровень лога" (Log level) set to "Debug" and "Локализация" (Localization) set to "Ru".
- Положение (Location):** Contains three input fields: "Широта" (Latitude) and "Долгота" (Longitude) both set to "0.000000", and a "Часовой пояс (UTC)" (Time zone) dropdown set to "(UTC-11:00) Midway".
- Загрузка проекта на сервер (Upload project to server):** Contains four text input fields for "IP-адрес сервера" (Server IP address), "Логин" (Login), "Пароль" (Password), and "PIN". Below these fields is a button labeled "Проверить соединение" (Check connection).

At the bottom right of the window are two buttons: "Сохранить" (Save) and "Отмена" (Cancel).

Общие:

1. **Уровень лога** - функция в разработке;
2. **Локализация** - локализация сообщений от сервера о системных событиях;

Положение:

1. **Широта, Долгота** - широта, долгота физического расположения сервера для работы астротаймеров расписания и назначения серверного времени.
2. **Часовой пояс** - часовой пояс физического расположения сервера для работы астротаймеров расписания и назначения серверного времени.

Загрузка проекта на сервер:

1. **IP-адрес на север** - IP-адрес SCADA сервера в формате IPv4;
2. **Логин** - логин пользователя, под аккаунтом которого будет производиться загрузка проекта на сервер. При отключенной авторизации не требуется.
3. **Пароль** - пароль пользователя, под аккаунтом которого будет производиться загрузка проекта на сервер. При отключенной авторизации не требуется.
4. **PIN** - защита от случайного перетирания проекта. PIN для доступа к загрузке проекта на сервер. Не требуется, если на сервере не настроен PIN-код.

Заполнение всех полей необязательно, т.к. на сервере может быть отключена авторизация или / и не назначен PIN.

При успешной проверке подключения рядом с кнопкой появится сообщение об установлении соединения с сервером.

В случае неудачи при проверке соединения рядом с кнопкой возникнет соответствующее сообщение.

Настройки проекта

СерверБаза данныхTelegramЖурналРезервирование

Уровень логаDebug

ЛокализацияRu

Положение

Широта0.000000

Долгота0.000000

Часовой пояс (UTC)05:00) Yekaterinburg

Загрузка проекта на сервер

IP-адрес сервера127.0.0.1

Логинadmin

Парольadmin

PIN

Проверить соединение

Соединение установлено / Нет соединения

Сохранить

Отмена

2.2.2. Настройки подключения к внешней базе данных

Настройки проекта

Сервер База данных Telegram Журнал Резервирование

Настройки подключения к внешней БД

База данных PostgreSQL

Хост

Порт 5432

Пользователь

Пароль

Имя БД

Проверить соединение

Сохранить Отмена

1. **База данных** - тип подключаемой внешней базы данных;
2. **Хост** - адрес сервера, на котором размещена база данных;
3. **Порт** - порт подключения к серверу PostgreSQL. По умолчанию: 5432;
4. **Пользователь** - имя пользователя или название роли, в качестве которого будет производиться подключение к базе данных;
5. **Пароль** - пароль подключаемого пользователя;
6. **Имя БД** - название базы данных, к которой вы хотите подключиться;
7. **Проверить соединение** - кнопка для проверки соединения с базой данных.

При успешной проверке подключения рядом с кнопкой появится сообщение об установлении соединения с базой данных.

Настройки подключения к внешней БД

База данных	PostgreSQL
Хост	127.0.0.1
Порт	5432
Пользователь	postgres
Пароль	root
Имя БД	postgres
Проверить соединение	Соединение установлено

В случае неудачи при проверке соединения рядом с кнопкой возникнет соответствующее сообщение.

Проверить соединение	Нет соединения
----------------------	----------------

2.2.3. Настройки Telegram-чатов

Настройки проекта

Сервер База данных **Telegram** Журнал Резервирование

Токен Частота опроса 1 Секунд

⊕ ⊖ ↑ ↓

№	Имя чата	ID чата
1	Test	chat id

Сохранить Отмена

Настройки чатов Telegram в проекте SCADA состоят из следующих элементов:

1. **Токен** - поле для ввода токена Telegram-бота, добавленного в диалоги для отправки сообщений;
2. **Частота опроса (сек.)** - значение частоты опроса изменений Telegram-чатов для получения обратной связи. Принимает значения в диапазоне от 1 до 3600 секунд;
3. **Панель управления списком Telegram-чатов проекта** - позволяет управлять чатами в таблице. Содержит кнопки добавления нового чата, удаления выделенного чата;
4. **Список чатов Telegram** - список диалогов Telegram, в которые могут быть отправлены сообщения:

- **Имя чата** - наименование чата, которое будет отображаться в iRidi Studio;
- **ID чата** - уникальный идентификатор для обращения к чату Telegram.

Чтобы отредактировать значения свойств существующей записи чата, необходимо совершить двойной клик по соответствующему полю и ввести новые данные.

2.2.4. Настройки журнала

Вкладка журнала содержит настройки стратегии очистки журнала аварий, а также настройки приоритетов системных событий для журнала истории.

The screenshot shows a window titled 'Настройки проекта' (Project Settings) with a close button. It has five tabs: 'Сервер', 'База данных', 'Telegram', 'Журнал' (selected), and 'Резервирование'. The 'Журнал' tab contains two sections:

Стратегия очистки записей в журнале аварий

- ☒ Очистка при нормализации значения тега
- ☐ Очистка при квитировании

Приоритет системных событий

Авторизация пользователя	0	▲▼
Деавторизация пользователя	0	▲▼
Запуск проекта	0	▲▼
Очистка журнала событий	0	▲▼
Доступ к тегу	0	▲▼
Изменение значения тега	0	▲▼

At the bottom right are two buttons: 'Сохранить' (Save) and 'Отмена' (Cancel).

- Стратегия очистки записей журнала аварий

Настройка позволяет выбрать режим автоматической очистки всех сообщений об аварийных состояниях, отображаемых в журнале аварий. В данный момент доступны две стратегии очистки сообщений:

- **Очистка при нормализации значения тега** - все сообщения об аварии будут исчезать из журнала аварий при возвращении значения тега, спровоцировавшего аларм, в диапазон нормы;

- **Очистка при квитировании** - сообщения об аварии уходят из журнала аварий при подтверждении пользователем.
- **Приоритеты системных событий**

Блок настроек отвечает за приоритеты не аварийных событий, происходящих в системе SCADA и отражающихся в журнале истории событий. Приоритет - важность данного сообщения. Приоритеты всех событий задаются целыми числами в диапазоне значений от 0 до 999 (наиболее серьезные алармы имеют приоритет 1). К системным событиям относятся:

- **Авторизация пользователя** - событие установления сетевого подключения к SCADA Server;
- **Деавторизация пользователя** - событие отключение сетевого соединения со SCADA Server;
- **Запуск проекта** - событие старта проекта на iRidi SCADA Server;
- **Очистка журнала событий** - событие массового подтверждения и принудительной очистки всех активных записей журнала аварий пользователем системы;
- **Доступ к тегу** - изменение уровня доступа к тегу для определенной группы пользователей;
- **Изменение значения тега** - событие изменения SCADA-системой значения любого из тегов проекта.

2.2.5. Настройки резервирования

Вкладка резервирования содержит настройки для реализации горячего резервирования системы.

Настройки проекта

Сервер База данных Telegram Журнал Резервирование

☐ Резервирование серверов

- Cluster 1
 - Server 1
 - Server 2

Свойства

Имя	Cluster 1
Таймаут пинга серверов (...)	10000
Кол-во попыток подключения	3

Доступ к управлению серверами

☐ Только после ввода пароля

Создать пароль

Сформировать файл резервирования Сохранить Отмена

- **Включения резервирования** - настройка позволяет включить резервирование серверов в системе. По умолчанию чекбокс выключен, все настройки заблокированы.
- **Древо резервирования** - содержит структуру кластеров и серверов, составляющих систему резервирования.
- **Свойства.** В разделе отображаются свойства, выделенного элемента дерева.

Для кластера:

- **Имя** - имя кластера.
- **Таймаут пинга серверов (мс)** - период опроса состояния резервируемых серверов в миллисекундах.
- **Количество попыток подключения** - количество допустимых неудачных попыток подключения к серверу, прежде чем он будет признан потерянным.

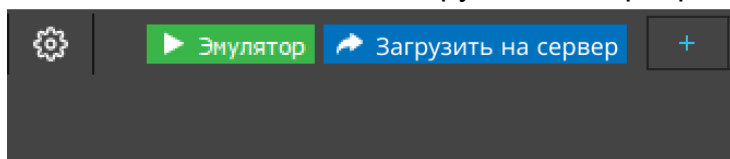
Для сервера:

- **Имя** - имя сервера.
- **Адрес** - IP-сервера.
- **HWID** - уникальный идентификатор сервера.
- **Приоритет** - приоритет сервера (1 - основной, 2 - резервный).
- **Определить HWID по адресу** - кнопка позволяет определить HWID по введенному IP-адресу сервера.
- **Проверить соединение** - кнопка для проверки соединения с сервером.
- **Доступ к управлению серверами** - настройка доступа к серверу по паролю.
 - **Только после ввода пароля** - настройка позволяет включить доступ к серверу по паролю.
 - **Создать/Изменить пароль** - настройка позволяет открыть окно для создания/изменения пароля.
- **Сформировать файл резервирования** - кнопка для создания config-файла, загружаемого на координатор резервирования iRidi SCADA Coordinator, для автоматического или ручного управления состоянием серверов.

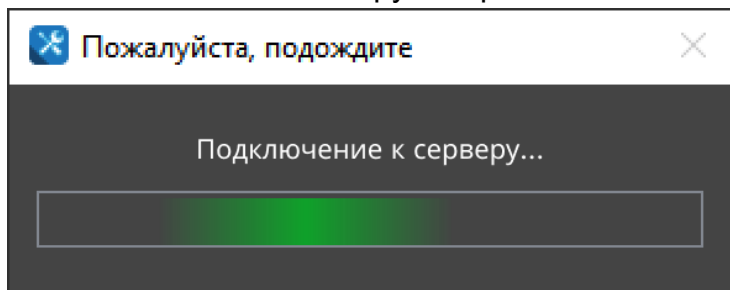
2.3. Загрузка проекта SCADA сервера

Для передачи серверного проекта SCADA на удаленный сервер:

1. Нажатием кнопки “Загрузить на сервер”

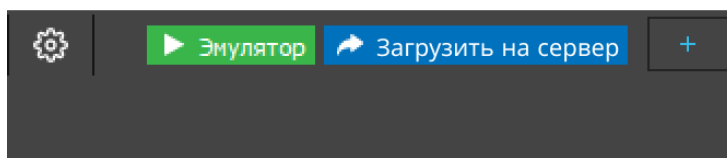


2. Появится окно загрузки проекта:



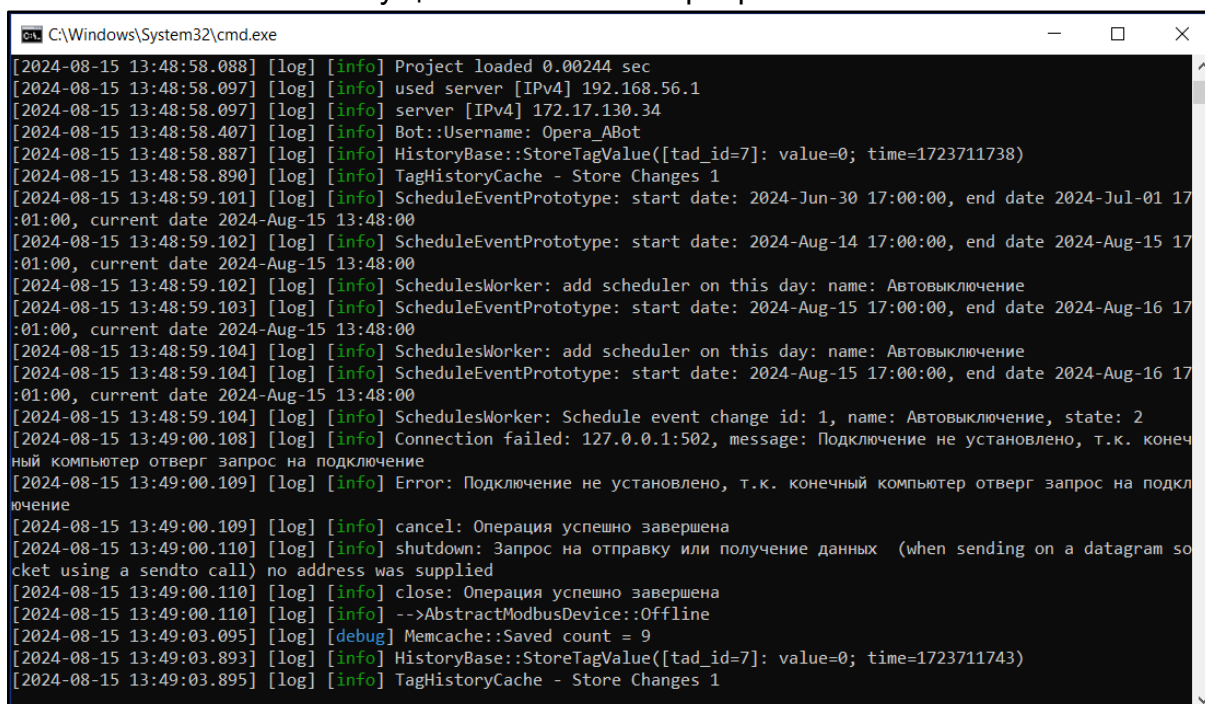
3. После завершения загрузки проекты будут загружены на сервер.

Для запуска Эмулятора:



1. Нажатием кнопки “Эмулятор”

2. Появится окно запущенного SCADA-сервера:

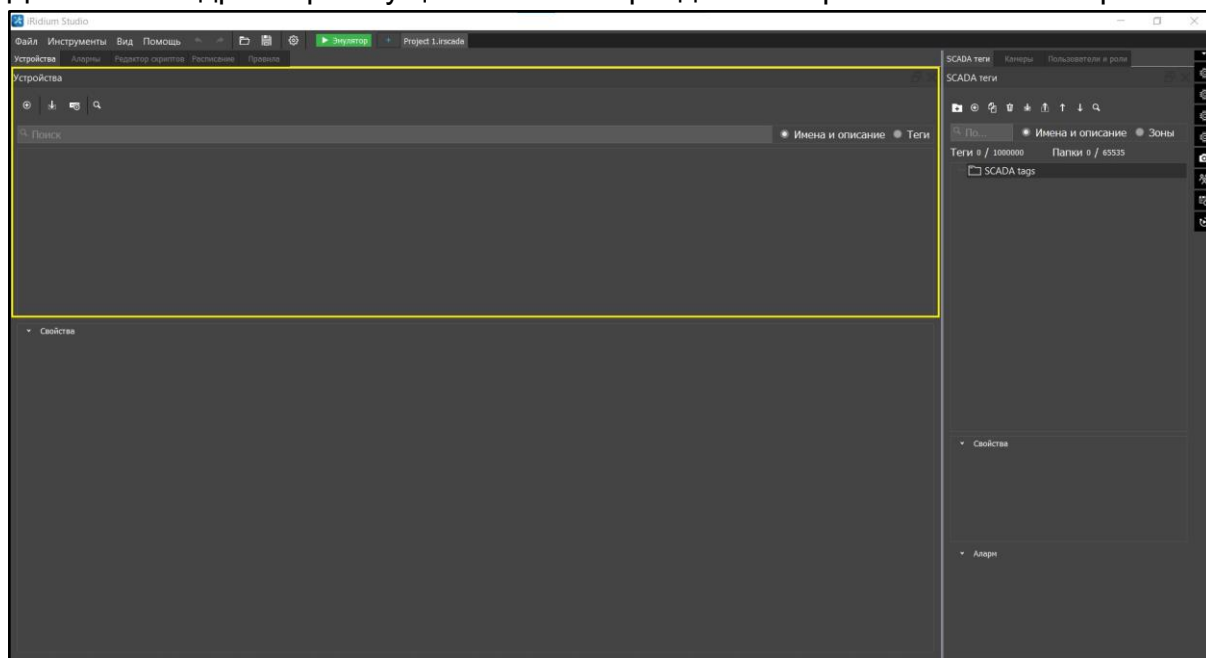


3. Добавление драйвера

Драйвер - это интерфейс для управления и получения состояния устройства системы автоматизации.

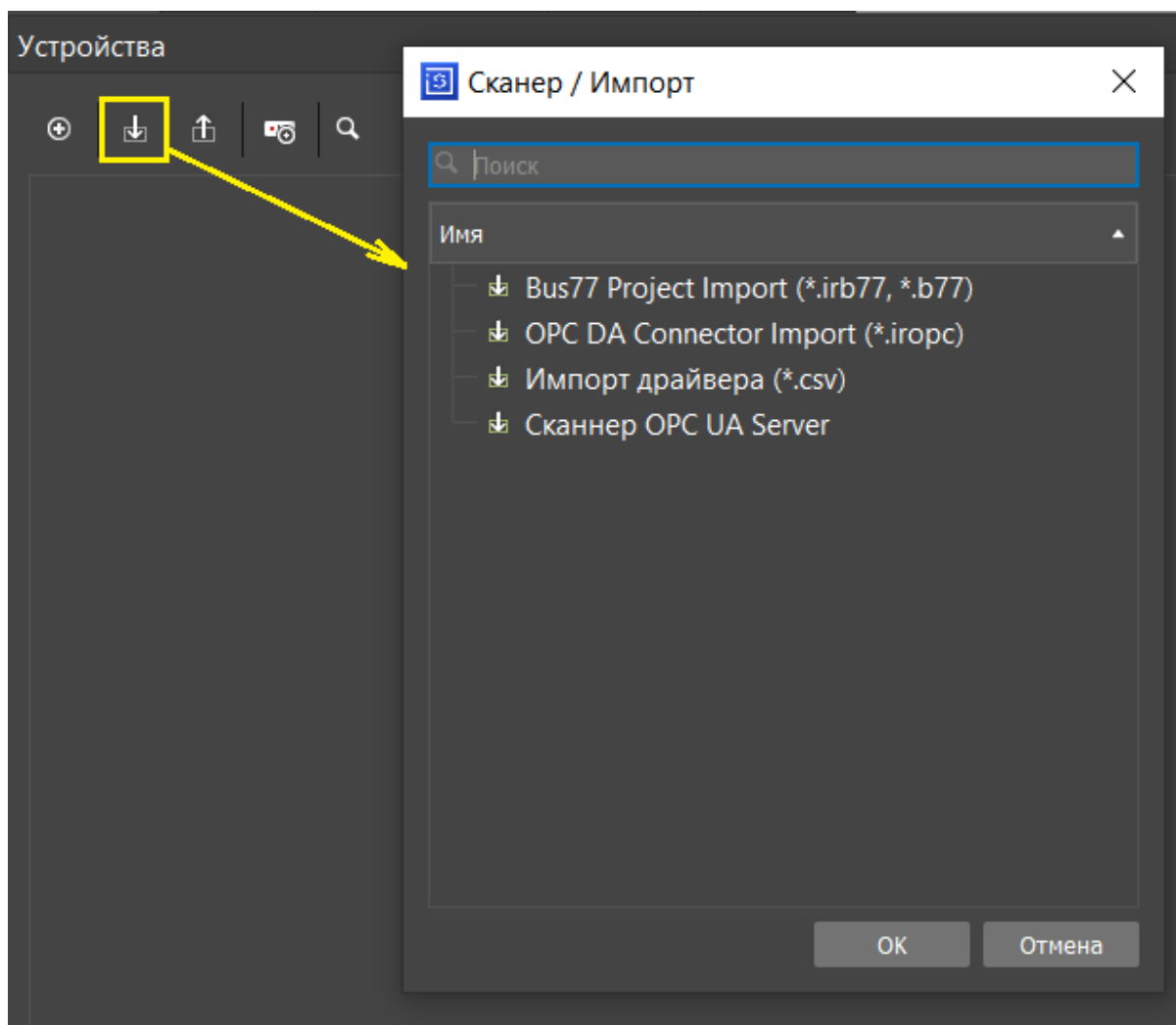
3.1. Добавление драйвера в SCADA-проект

Добавление драйвера осуществляется в разделе “Устройства” SCADA-проекта.



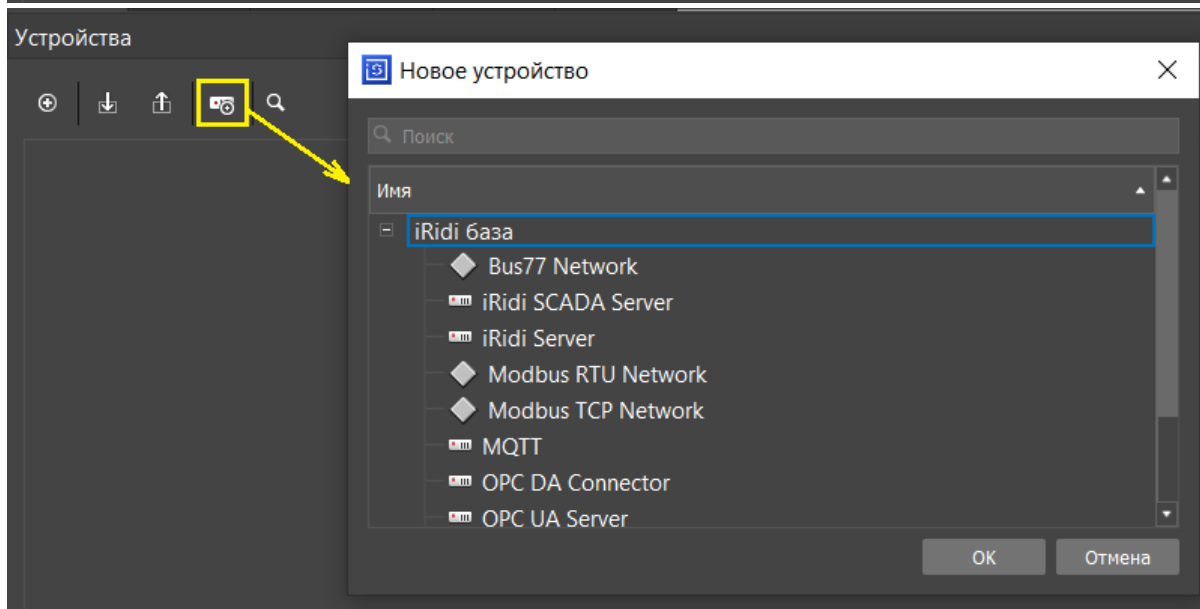
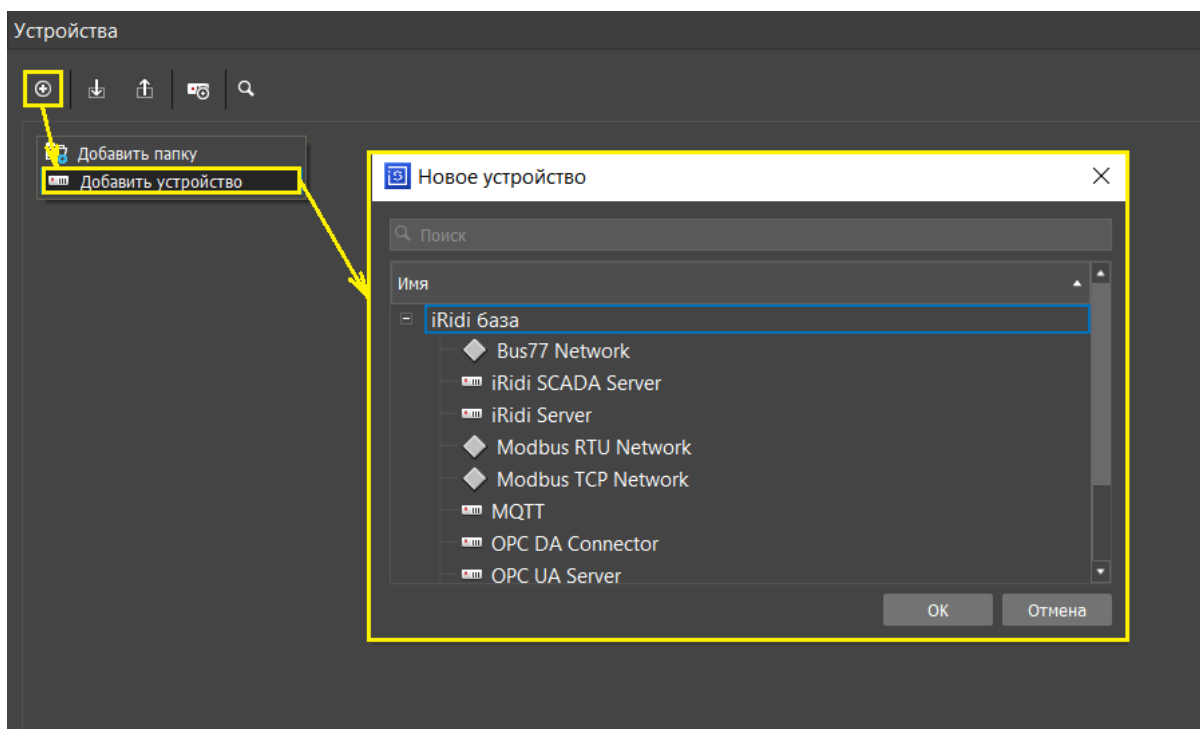
Драйвер в SCADA-проект можно добавить:

- **импортом файла** (формата *.irb77, *.b77, *.csv, *.iropc) конфигурации вашего оборудования:



В проект будет добавлен драйвер с настроенными устройствами, командами и фидбэками. Изменить настройки можно в параметрах устройств;

- из **библиотеки драйверов**, где хранится список драйверов, которые можно использовать в проекте.



Список драйверов:

- **Bus77 Network** - драйвер, позволяющий управлять домом или офисом на базе оборудования, работающего по протоколу Bus77;
- **iRidi SCADA Server** - драйвер для связи нескольких SCADA-серверов;
- **iRidi Server** - драйвер для интеграции в SCADA-систему сегментов автоматизации, управляемых iRidi-серверами;
- **Modbus RTU Network** - драйвер предназначен для управления оборудованием, работающим по протоколу Modbus RTU;
- **Modbus TCP Network** - драйвер предназначен для управления оборудованием, работающим по протоколу Modbus TCP;
- **MQTT** - драйвер предназначен для управления оборудованием, работающим по протоколу MQTT;

- **OPC DA Connector** - серверный драйвер, позволяющий обмениваться данными с переменными OPC-серверов по стандарту OPC DA.
- **OPC UA Server** - серверный драйвер, позволяющий обмениваться данными с переменными OPC-серверов по стандарту OPC UA.
- **Ping Network** - драйвер производит периодический пинг устройств по IP;
- **SNMP** - драйвер предназначен для управления оборудованием, работающим по протоколу SNMP v1.

При добавлении в проект будет добавлен нативный драйвер, который нуждается в настройке.

3.2. Настройка параметров подключения

Для нативных драйверов, добавленных из базы данных, необходимо настроить параметры подключения. Есть общие параметры, характерные для любых драйверов независимо от типа:

- **Name** - имя управляемого устройства, произвольно;
- **Driver Type** - тип драйвера, неизменен;
- **Protocol** - тип протокола;
- **Tags** - метка, которая позволит быстрее находить драйвер;
- **Описание** - общее описание драйвера.

▼ Свойства	
Name	iRidium SCADA Server
Driver Type	iRidium SCADA Server
Protocol	TCP
Source Project	
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	30464
Login	
Password	
Wait for Connec...	0
Wait for Data (...)	0
Keep Alive	60000
Mode	Direct Connection
Tags	
Описание	

3.2.1. Bus77 Network

Bus77 Network	
Свойства	
Name	Bus77 Network
Driver Type	Bus77 Network
Protocol	TCP
Source Project	
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	30464
Keep Alive	10000
LID	0
Line polling frequency (ms)	0
Tags	
Описание	

Свойства драйвера:

- **Source Project** - исходный проект;
- **Log Level** - уровень логирования драйвера;
- **Host** - IP-адрес оборудования;
- **Port** - порт подключения;
- **Keep Alive** - интервал проверки TCP соединения. В случае отсутствия ответа на запрос Keep Alive, драйвер будет принудительно переведён в состояние отключен;
- **LID** - адрес на линии.
- **Line polling frequency** - частота опроса линии. Параметр отвечает за частоту опроса устройств Bus77.

После добавления в проект драйвера Bus77 Network и его настройки нужно синхронизировать драйвер с исходным проектом для получения КУ и КОС драйвера. Для синхронизации нужно нажать кнопку “Синхронизировать”.



Настройка подключения устройств:

- **Name** - имя;
- **Driver Type** - тип драйвера;
- **LID** - адрес на линии;
- **Model** - модель устройства;
- **PIN** - пин-код устройства;
- **Tags** - метка, которая позволит быстрее находить устройство;

- **Описание** - общее описание драйвера.

▼ Свойства	
Name	DM-306PS 5
Driver Type	Bus77 Network
LID	20
Model	DM-306PS
PIN	0
Tags	
Описание	

3.2.2. iRidi Server

+ iRidi Server	
▼ Свойства	
Name	iRidi Server
Driver Type	iRidi Server
Protocol	TCP
Source Project	
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	30464
Login	
Password	
Wait for Connect (ms)	0
Wait for Data (ms)	0
Keep Alive	60000
Подключение	Remote cloud
Cloud Token	
Tags	
Описание	

Свойства драйвера:

- **Source Project** - исходный проект;
- **Log Level** - уровень логирования драйвера;
- **Host** - IP-адрес оборудования;
- **Port** - порт подключения;
- **Login** - логин пользователя (если задан на контроллере);
- **Password** - пароль пользователя (если задан на контроллере);
- **Wait for Connection (ms)** - ожидание соединения;
- **Wait for Data (мс)** - время ожидания ответа от драйвера. После заданного времени делает переподключение драйвера;

- **Keep Alive** - интервал проверки TCP соединения. В случае отсутствия ответа на запрос Keep Alive, драйвер будет принудительно переведён в состояние отключен;
- **Подключение** - тип подключения к серверу (прямое или облачное);
- **Cloud Token** (только для типа подключения Remote cloud) - ключ/токен для подключения к облачному объекту.

Для получения ключа/токена для подключения к облачному объекту необходимо:

1. Добавить в SCADA-проект драйвер iRidi Server.
2. Настроить драйвер. Задать исходный проект iRidi Pro Server, тип подключения - Remote cloud.

iRidi Server	
Свойства	
Name	iRidi Server
Driver Type	iRidi Server
Protocol	TCP
Source Project	Практика_Серверный.srpz
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	30464
Login	
Password	
Wait for Connect (ms)	0
Wait for Data (ms)	0
Keep Alive	60000
Подключение	Remote cloud
Cloud Token	
Tags	
Описание	

3. Зайти в свой личный кабинет на сайте iridi.com.
4. Открыть настройки объекта, к которому необходимо подключиться.
5. В разделе SCADA нажать на кнопку "Добавить токен".

27 марта

Pro курс

Кабинет TRIAL

Добавить объект

System

Объект 1

Добавить папку

Режим редактирования

Профиль Проекты Лицензии Пользователи и панели Группы (push) SCADA

Получить список токенов

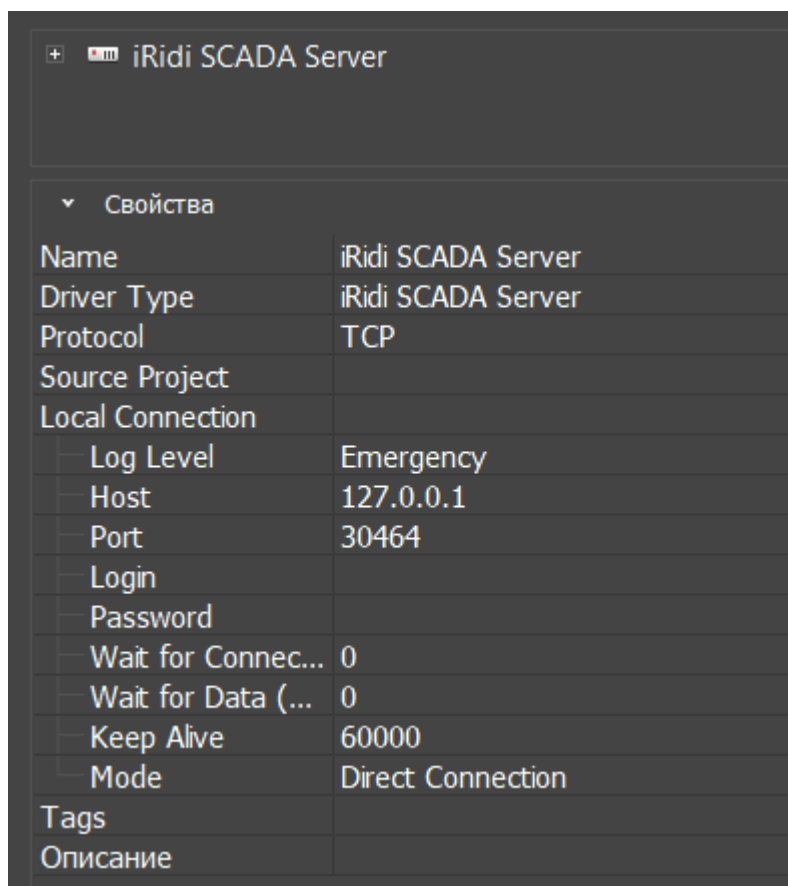
Добавить токен

Подключение	Токен	Дата создания	Действия
Серверы iRidi Scada 1			
SCADA server	eyJhbGdOUlUzITNiIsinR5cCI6IkpXVCJ9.eyJkZKZpY2VfdG9rZW4iOiZNDY2NDUxMTk2OTg5Nzk4NTkxIiwidG9rZW5fcmRZSi6iINDQURBIHNlcnZlcisiInRva2VUX3R5cGUiOiZYZFkySisimRldmijZV90eXBjIjoic2VydmVyX2FzX2NsaWVudCisinVzZXfaWQiOiZMT0NlCisim9amVjdF9pZC16jYxMDU4Iiwib2JqZW50X2tleSI6ImM4NDUxOWY5OGNINWMyYjIjLCJ0b3N0b2ljOiOUUUMTkyLjM1LjB1IiwicG9yZCI6IjkwODg1LCJpYXQiOiE3NDU0OTA1Nzh9.ajm4P2hQRSQmTfhToG_VIGJTcIfQjaWgetnhZ4g4	24.04.2025	Переименовать Отозвать

После добавления в проект драйвера iRidi Server, драйвер и его настройки нужно синхронизировать с исходным проектом. Для синхронизации нужно нажать кнопку “Синхронизировать” для получения КУ и КОС драйвера. Для обновления КУ и КОС драйвера после внесения изменений в исходный проект также нужно синхронизировать драйвер.



3.2.3. iRidi SCADA Server



Свойства драйвера:

- **Source Project** - исходный проект;
- **Log Level** - уровень логирования драйвера;
- **Host** - IP-адрес оборудования;
- **Port** - порт подключения;
- **Login** - логин пользователя (если задан на контроллере);
- **Password** - пароль пользователя (если задан на контроллере);
- **Wait for Connection (ms)** - ожидание соединения;
- **Wait for Data (ms)** - время ожидания ответа от драйвера. После заданного времени делает переподключение драйвера;
- **Keep Alive** - интервал проверки TCP соединения. В случае отсутствия ответа на запрос Keep Alive, драйвер будет принудительно переведён в состояние отключен;
- **Mode** - тип подключения к серверу (прямое или облачное).

После добавления в проект драйвера iRidi SCADA Server, драйвер и его настройки нужно синхронизировать с исходным проектом. Для синхронизации нужно нажать кнопку "Синхронизировать" для получения КУ и КОС драйвера. Для обновления КУ и КОС драйвера после внесения изменений в исходный проект также нужно синхронизировать драйвер.



3.2.4. Modbus RTU Network

+ Modbus RTU Network	
▼ Свойства	
Name	Modbus RTU Network
Driver Type	Modbus RTU Network
Protocol	TCP Gate
Local Connection	
Log Level	Emergency
Host	192.168.0.100
Port	502
Connect wait time	7000
Receive wait time	25000
Update Time	1000
Coil Write Function	Multiple
Holding Write Function	Multiple
Send wait time	12000
Tags	
Описание	

Свойства драйвера:

- **Log Level** - уровень логирования драйвера;
- **Host** - IP-адрес оборудования;
- **Port** - порт подключения;
- **Connect wait time** (мс) - время ожидания подключения к оборудованию, как часто драйвер будет пытаться переподключиться (если он offline);
- **Receive wait time** (мс) - время ожидания ответа от оборудования (пакета), как долго драйвер будет ждать ответ на свои запросы;
- **Update Time** (мс) - периодичность опроса ПЛК для обновления данных о его состоянии;
- **Coil White Function** - режим отправки команды дискретных команд(Single - запись одного регистра флагов, Multiple - запись нескольких регистров флагов);
- **Holding White Function** - режим отправки команды для выходных регистров(Single - запись одного регистра, Multiple - запись нескольких регистров);
- **Send wait time** (мс) - пауза между попытками опроса сабдевайса. Количество попыток указывается в сабдевайсе в поле "Number of attempts".

Для настройки команд и фитбеков необходимо установить значение следующих полей:

- **Name** - имя тега;

- **Полное имя** - путь к тегу, именно этот путь будет указан в скриптах, если вы захотите воздействовать на него;
- **Description** - краткое описание тега;
- **Type** - регистр хранения, доступный на чтение и запись. iRidi поддерживает следующие регистры и функции:
 - **Coil Register** (0x05, 0x0F, 0x01) - регистры флагов, один бит, чтение и запись;
 - **Holding Register** (0x06, 0x10, 0x03) - регистры хранения, 16-битное слово, чтение и запись;
 - **Discret Inputs** (0x02) - дискретные входы, один бит, только чтение;
 - **Input Register** (0x04) - регистры ввода, 16-битное слово, только чтение.
- **Address** - адрес регистра в десятичном формате;
- **Tags** - название, предназначенное для поиска тега или группы тегов в проекте через поисковую строку;
- **Описание** - общее описание команды или фитбека.

▼ Свойства	
Name	Feedback
Полное имя	Custom:Feedback
Description	
Тип	Coil
Address	1
Tags	
Описание	

▼ Свойства	
Name	Channel
Полное имя	Custom:Channel
Description	
Тип	Coil
Address	0
Tags	
Описание	

3.2.5. Modbus TCP Network

+ Modbus TCP Network	
▼ Свойства	
Name	Modbus TCP Network
Driver Type	Modbus TCP Network
Protocol	TCP
Local Connection	
Log Level	Emergency
Host	192.168.0.100
Port	502
Wait for Data (ms)	25000
Keep Alive	60000
Update Time	1000
Coil Write Function	Multiple
Holding Write Function	Multiple
Single Poll	False
Send wait time	5000
Tags	
Описание	

Свойства драйвера:

- **Log Level** - уровень логирования драйвера;
- **Host** - IP-адрес оборудования;
- **Port** - порт подключения;
- **Wait for Data (мс)** - время ожидания ответа от драйвера. После заданного времени делает переподключение драйвера;
- **Keep Alive (мс)** - интервал проверки TCP соединения. В случае отсутствия ответа на запрос Keep Alive, драйвер будет принудительно переведён в состояние отключен. Параметр сделан для корректного отображения онлайн-статуса при экстренном\неправильном отключении клиента. Значение 0 - отключение настройки;
- **Update time (мс)** - периодичность опроса контроллера для обновления данных о его состоянии;
- **Coil Write Function**- отправка отдельного пакета на запись каждого регистра. Single - запись одного регистра флагов, Multiple - запись нескольких регистров флагов);
- **Holding Write Function** - режим отправки команды для выходных регистров (Single - запись одного регистра, Multiple - запись нескольких регистров);
- **Single Poll** - при включении параметра происходит формирование отдельного пакета для опроса каждого канала сабдевайса;

- **Send Wait Time** (мс) - пауза между попытками опроса сабдевайса. Количество попыток указывается в сабдевайсе в поле "Number of attempts".

Для настройки команд и фитбеков необходимо установить значение следующих полей:

- **Name** - имя тега;
- **Полное имя** - путь к тегу, именно этот путь будет указан в скриптах, если вы захотите воздействовать на него;
- **Description** - краткое описание тега;
- **Type** - регистр хранения, доступный на чтение и запись. iRidi поддерживает следующие регистры и функции:
- **Coil Register** (0x05, 0x0F, 0x01) - регистры флагов, один бит, чтение и запись;
- **Holding Register** (0x06, 0x10, 0x03) - регистры хранения, 16-битное слово, чтение и запись;
- **Discret Inputs** (0x02) - дискретные входы, один бит, только чтение;
- **Input Register** (0x04) - регистры ввода, 16-битное слово, только чтение.
- **Address** - адрес регистра в десятичном формате;
- **Tags** - название, предназначенное для поиска тега или группы тегов в проекте через поисковую строку;
- **Описание** - общее описание команды или фитбека.

▼ Свойства	
Name	Feedback
Полное имя	Custom:Feedback
Description	
Тип	Coil
Address	1
Tags	
Описание	

▼ Свойства	
Name	Channel
Полное имя	Custom:Channel
Description	
Тип	Coil
Address	0
Tags	
Описание	

3.2.6. MQTT

MQTT	
Свойства	
Name	MQTT
Driver Type	MQTT
Protocol	TCP
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	1883
Login	
Password	
SSL	Disable
Keep Alive	60000
MQTT Version	4
Client ID	
Clean	True
Tags	
Описание	

Свойства драйвера:

- **Host** - IP-адрес контроллера;
- **Port** - порт подключения к контроллеру;
- **Login** - логин пользователя(Если задан на контроллере);
- **Password** - пароль пользователя(Если задан на контроллере);
- **MQTT version** - версия протокола на контроллере: 3 - устаревший протокол MQ Integrator SCADA Device Protocol (MQisdp), 4 - поддержка MQTT версии 3.1.1;
- **Client ID** - идентификатор панели;
- **Keep Alive** - время удержание соединения в секундах;
- **Clean** - флаг "чистый старт", удаление подписок при разрыве соединения: true - чистая сессия, false - устойчивые соединения;
- **SSL** - включить/выключить шифрование.

При использовании SSL/TLS используется только *CA signed server certificate*, сертификаты *Self signed certificates* и *CA certificate only* не поддерживаются.

Для настройки команды необходимо установить значение следующих полей:

- **Name** - имя команды;
- **Topic** - путь MQTT (максимальный объем данных, передаваемых в топик MQTT, до 268435456 байт);
- **Value Type** - тип значения;

- **QoS** - качество обслуживания, гарантия доставки сообщения от издателя брокеру:
 - *At most once* - не более одного раза, без повторений, доставка не гарантируется;
 - *At least once* - один раз и более, могут быть повторения, доставка гарантируется;
 - *Exactly once* - ровно один раз, без повторений, доставка гарантируется.
- **Retain** - сохранение публикации: брокер хранит retain-топик и при новых подписках рассылает его с флагом retain;
- **Tags** - название, предназначенное для поиска тега или группы тегов в проекте через поисковую строку;
- **Описание** - общее описание команды.

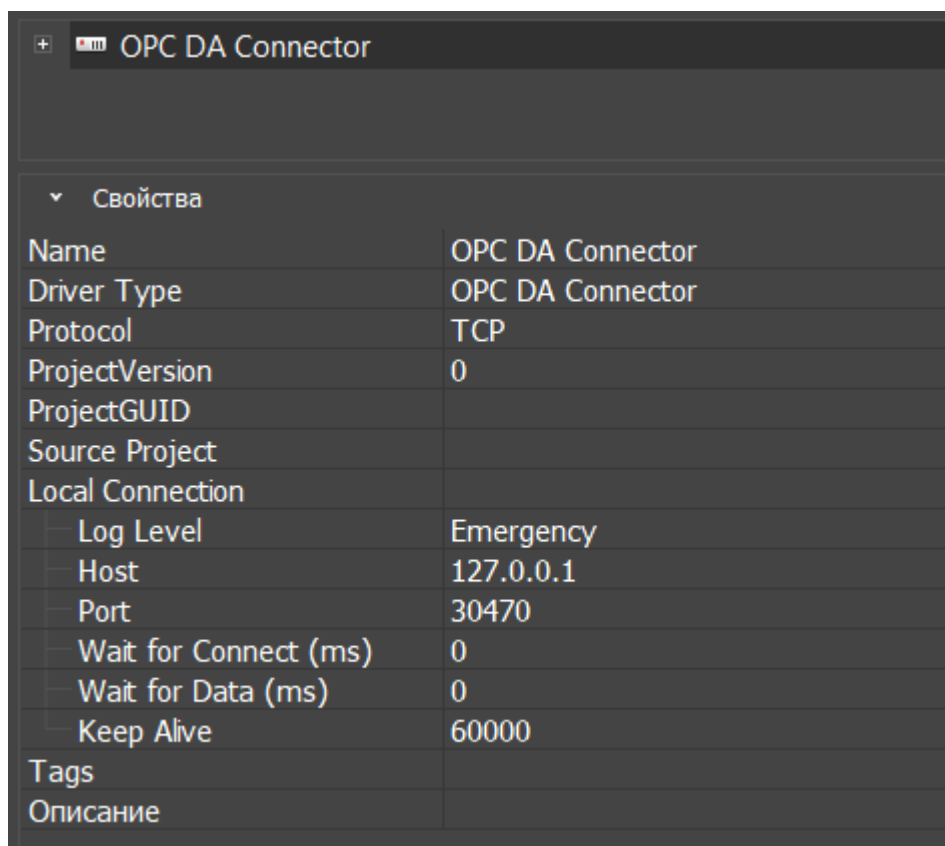
▼ Свойства	
Name	Command 1
Topic	
Value Type	Boolean
QoS	At most once
Retain	False
Tags	
Описание	

Для настройки фитбека необходимо установить значение следующих полей:

- **Name** - имя команды;
- **Topic** - путь MQTT (максимальный объем данных, передаваемых в топик MQTT, до 268435456 байт);
- **Value Type** - тип значения;
- **Tags** - название, предназначенное для поиска тега или группы тегов в проекте через поисковую строку;
- **Описание** - общее описание команды.

▼ Свойства	
Name	Feedback 1
Topic	
Value Type	Boolean
Tags	
Описание	

3.2.7. OPC DA Connector



The screenshot shows a configuration window titled "OPC DA Connector". It contains a section labeled "Свойства" (Properties) with a list of configuration parameters and their values.

Name	OPC DA Connector
Driver Type	OPC DA Connector
Protocol	TCP
ProjectVersion	0
ProjectGUID	
Source Project	
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	30470
Wait for Connect (ms)	0
Wait for Data (ms)	0
Keep Alive	60000
Tags	
Описание	

Свойства драйвера:

- **ProjectVersion** - версия проекта;
- **ProjectGUID** - уникальный идентификатор исходного проекта;
- **Source Project** - исходный проект конфигурации OPC DA сервера;
- **Log Level** - уровень логирования;
- **Host** - IP-адрес контроллера;
- **Port** - порт подключения к контроллеру;
- **Wait of Connect (ms)** - время ожидания соединения;
- **Wait for Data (мс)** - время ожидания ответа от драйвера. После заданного времени делает переподключение драйвера;
- **Keep Alive** - время удержание соединения в секундах.

3.2.8. OPC UA Server

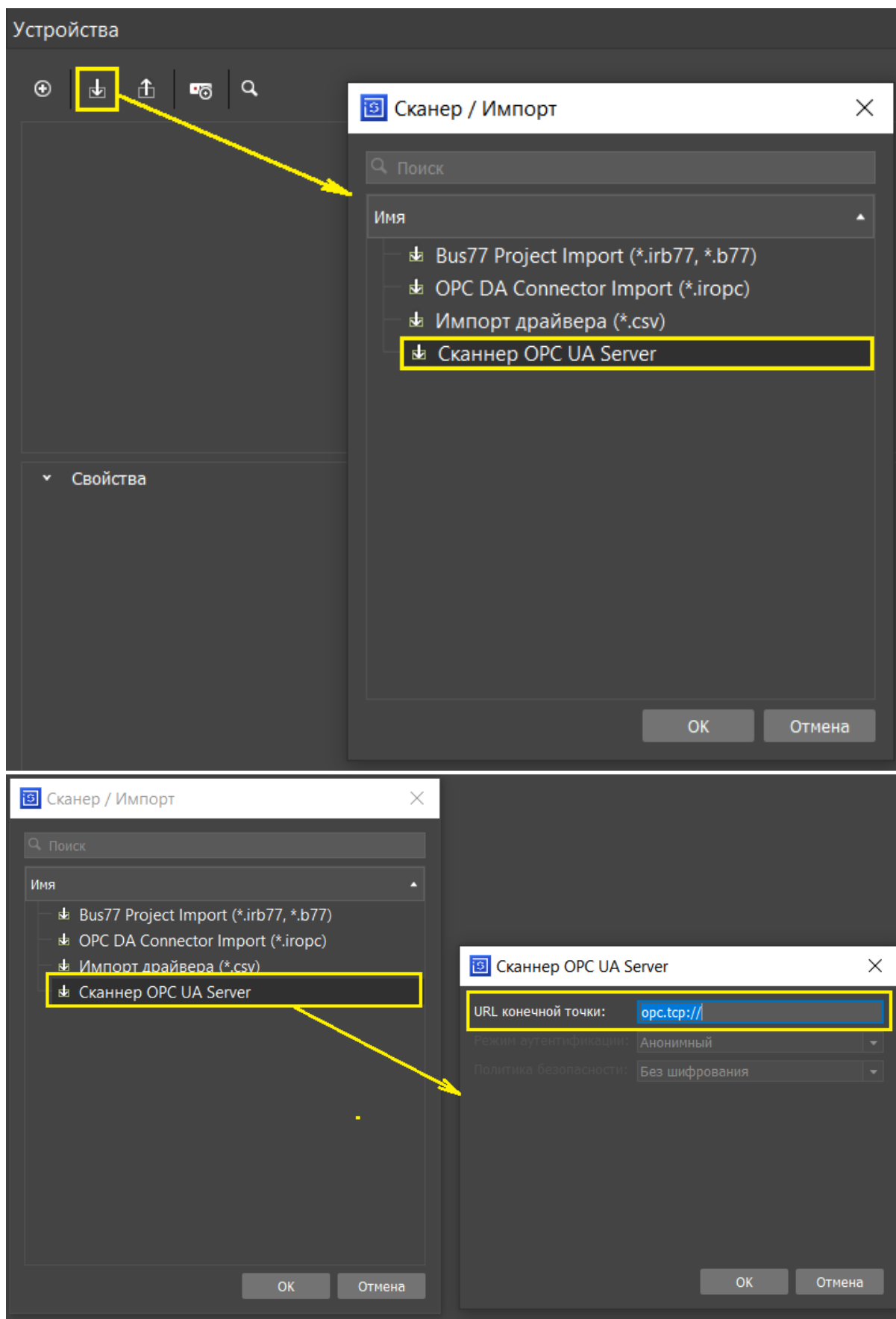
+ OPC UA Server	
▼ Свойства	
Name	OPC UA Server
Driver Type	OPC UA Server
Protocol	TCP
Authentication	Anonymous
Security Mode	No encryption
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	53530
Endpoint URL	opc.tcp://127.0.0.1:53530/Server
Tags	
Описание	

Свойства драйвера:

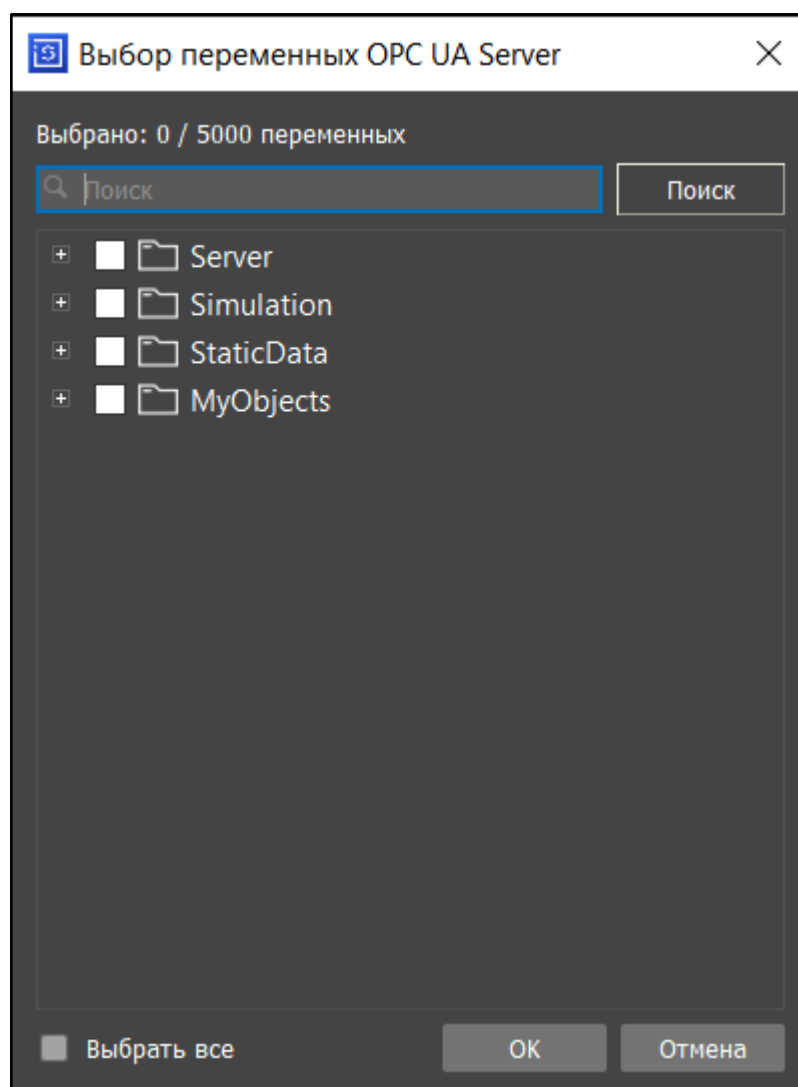
- **Authentication** - режим авторизации;
- **Security Mode** - режим безопасности соединения;
- **Log Level** - уровень логирования;
- **Host** - IP-адрес контроллера;
- **Port** - порт подключения к контроллеру;
- **Endpoint URL** - URL OPC UA сервера.

Добавление драйвера осуществляется через сканнер OPC UA Server:

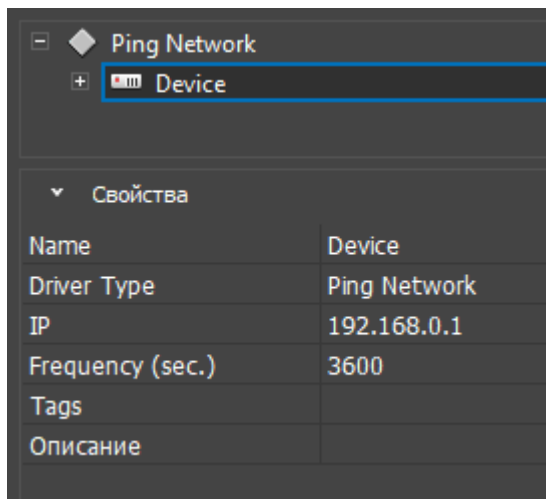
1. Открываете сканер OPC UA Server.
2. В сканере указывается URL конечной точки.



3. Выбираете переменные для добавления в проект.



3.2.9. Ping Network



Свойства драйвера:

- **IP** - IP адрес устройства;
- **Frequency (sec.)** - частота опроса устройства.

3.2.10. SNMP

SNMP	
Свойства	
Name	SNMP
Driver Type	SNMP
Protocol	UDP
Local Connection	
Log Level	Emergency
Host	192.168.0.100
Port	161
Local Port	162
Request tags at start	True
Interval between tag polls	400
Maximum number of tag polls	3
Tags	
Описание	

Свойства драйвера:

- **Log Level** - уровень логирования драйвера;
- **Host** - IP-адрес устройства;
- **Port** - порт подключения к устройству для передачи данных (по умолчанию 161, драйвер посылает запросы агенту);
- **Local Port** - порт подключения к устройству для получения данных (по умолчанию 162, агент посылает ответ драйверу);
- **Request tags at start** - опрашивать при старте(true/false);
- **Interval between tag polls** - интервал между опросами тегов. после безуспешного опроса драйвер переключается на опрос другого тега;
- **Maximum number of tag polls** - количество попыток опроса каждого тега.

Для настройки команды необходимо установить значение следующих полей:

- **Name** - имя тега;
- **OID** - числовой идентификатор объекта;
- **Тип** - тип команды:
 - Get - получить значение команды с устройства;
 - Set - установить значение команды в устройстве;
- **Community** - последовательность символов, описывающая принадлежность к группе;
- **Tags** - название, предназначенное для поиска тега или группы тегов в проекте через поисковую строку;
- **Описание** - общее описание команды.

▼ Свойства	
Name	Command 1
OID	
Тип	Get
Comunity	
Tags	
Описание	

Для настройки фидбека необходимо установить значение следующих полей:

- **Name** - имя тега;
- **OID** - числовой идентификатор объекта.
- **Тип** - команда для запроса данных с устройства:
 - *Value* - фидбек для возврата запрошенного значения;
 - *Trap* - событийный фидбек, срабатывает когда что-либо произошло.
- **Comunity** - последовательность символов, описывающая принадлежность к группе, своего рода пароль;
- **Tags** - название, предназначенное для поиска тега или группы тегов в проекте через поисковую строку;
- **Описание** - общее описание команды.

3.2.11. BACnet Network

+ BACnet Network	
▼ Свойства	
Name	BACnet Network
Driver Type	BACnet Network
Protocol	UDP
Local Connection	
Log Level	Emergency
Host	192.168.0.100
Port	47808
Send Timeout	20
Update Timeout	1000
Re-subscribe Timeout(min)	60
Local Port	47808
Tags	
Описание	

Свойства драйвера:

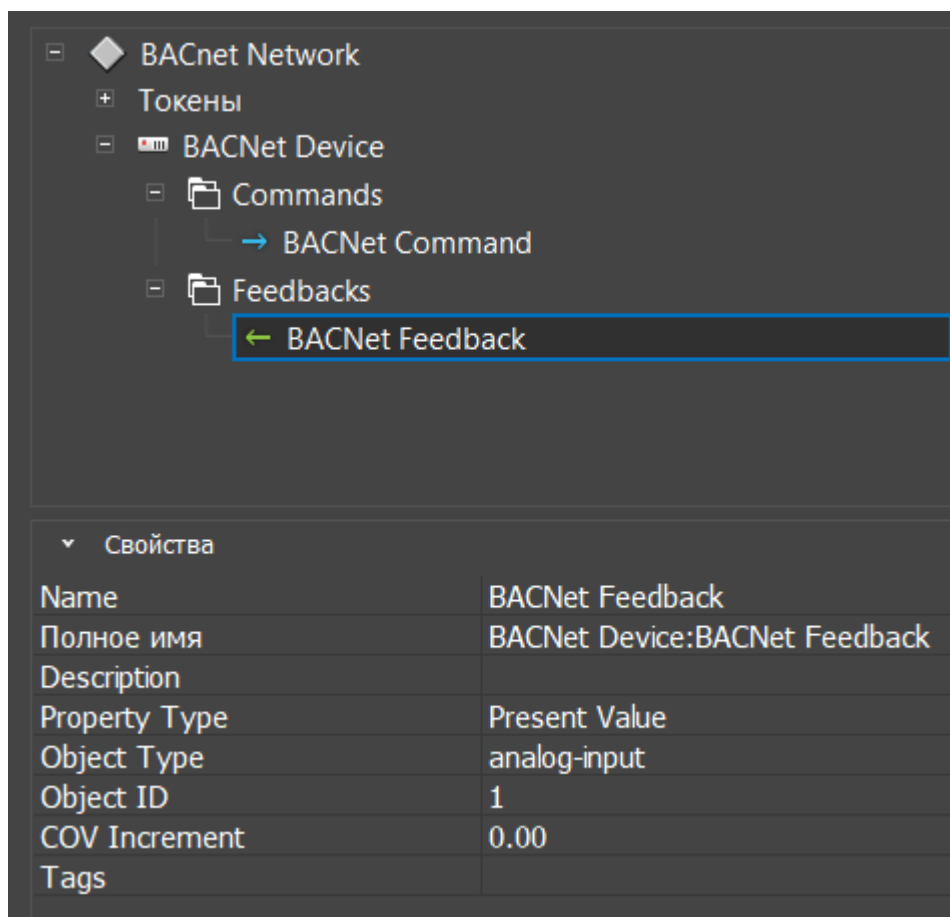
- **Log Level** - уровень логирования драйвера;
- **Host** - IP-адрес контроллера BACnet;
- **Port** - порт подключения к контроллеру BACnet (по умолчанию 47808, драйвер посылает запросы агенту);

- **Send Timeout** - время таймаута отправки пакетов в миллисекундах;
- **Update Timeout** - время подготовки данных в буфер для отправки в миллисекундах. Update Timeout должен быть в несколько раз больше чем Send Timeout,
- **Re-subscribe Timeout (min)** - время обновления подписки;
- **Local Port** - порт для обратной связи (не обязательно) (по умолчанию 47808, агент посылает ответ драйверу).

▼ Свойства	
Name	BACNet Device
Driver Type	BACnet Network
Device Instance	0
Priority	16
Network ID	65535
Mac Address (dec)	
Update type	Auto (subscribe cov or read prop...
Tags	
Описание	

Устройства в сети BACnet имеют несколько важных настроек.

- **Device Instance** - идентификатор устройства Bacnet. Его следует посмотреть в web интерфейсе контроллера
- **Priority** - приоритет операции записи, этот параметр может иметь значение в диапазоне 1-16 и обозначает.
- **Network ID** - номер сети уникально идентифицирует сеть в системе BACnet. Приложение i-Vu® автоматически генерирует все номера сети. Номера сетей должны быть уникальными в системе BACnet независимо от протокола. BACnet сетевые номера MS / TP и BACnet / IP не могут быть дублированы. Номер сети может составлять от 1 до 65534 (с 65 535 зарезервированными для трансляций).
- **Mac address** - являются идентификатором нижнего уровня в BACnet. В i-Vu® Open, MAC варьируется от 1 до 99 и должны быть уникальными для каждого устройства в пределах сети.
- **Update type** - тип обновления фидбэков в протоколе (подписка или опрос):
 - Auto (subscribe cov or read property) - автоматическое определение типа.
 - **Read property** - принудительный опрос переменных.



Для настройки команды или фидбека необходимо установить значение следующих полей:

- **Property Type** - тип значения канала/тега:
 - **Present Value** - режим чтения значения с устройства,
 - **Out of Service** - режим записи значения с панели,
- **Object Type** - тип переменной,

В данный момент поддерживаются следующие типы переменных:

Тип	Описание
analog-input	Аналоговый вход
analog-output	Аналоговый выход
analog-value	Аналоговое значение

binary-input	Бинарный вход
binary-output	Бинарный выход
binary-value	Бинарное значение
multi-state-input	Вход со многими состояниями
multi-state-output	Выход со многими состояниями
multi-state-value	Значение со многими состояниями

- **ObjectID** - идентификатор переменной (входа, выхода). Его можно найти в Web интерфейсе контроллера (BaCnet Interface - Application - Points - Переменная - Вкладка Advanced),
- **COV increment** - минимальное изменение величины, при котором происходит обновление данных.

4. Работа с тегами

4.1. Описание

Тег - это канал ввода/вывода (реальный физический или виртуальный), содержащий какое-то значение. Например:

- значение температуры, полученная с датчика температуры;
- режим работы установки;
- состояние реле;
- мощность, потребляемая нагревателем;
- управляющий сигнал (для изменения состояния коммутатора)
- и т.д.

В теге хранится значение, которое получено от прибора или сформировано скриптом. Тег может быть сохранен в базе данных или его значение может быть отправлено в Telegram-чат. Значения тега может использоваться для контроля аварийных ситуаций.

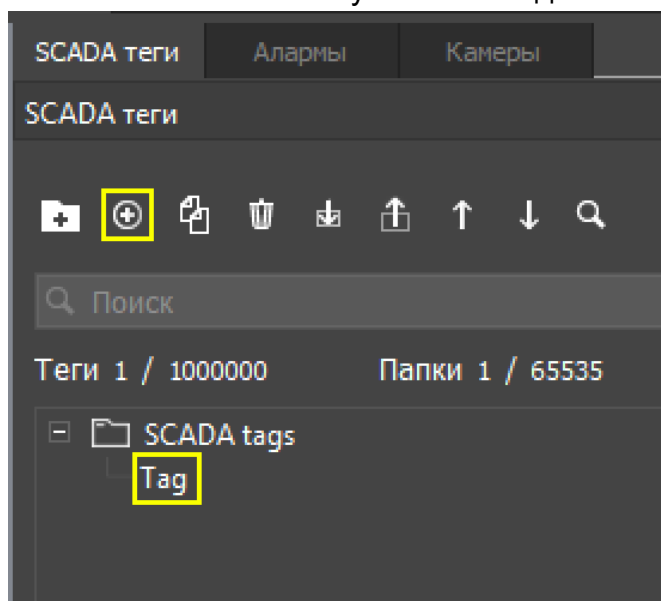
Количество тегов, которое может быть использовано в проекте, зависит от типа лицензии.

При создании SCADA-проекта настоятельно рекомендуется делать 20% запас по количеству тегов. Например, если подсчеты показывают, что для реализации необходимого функционала SCADA-системы требуется 100 тегов, то рекомендуется использовать 120 тегов.

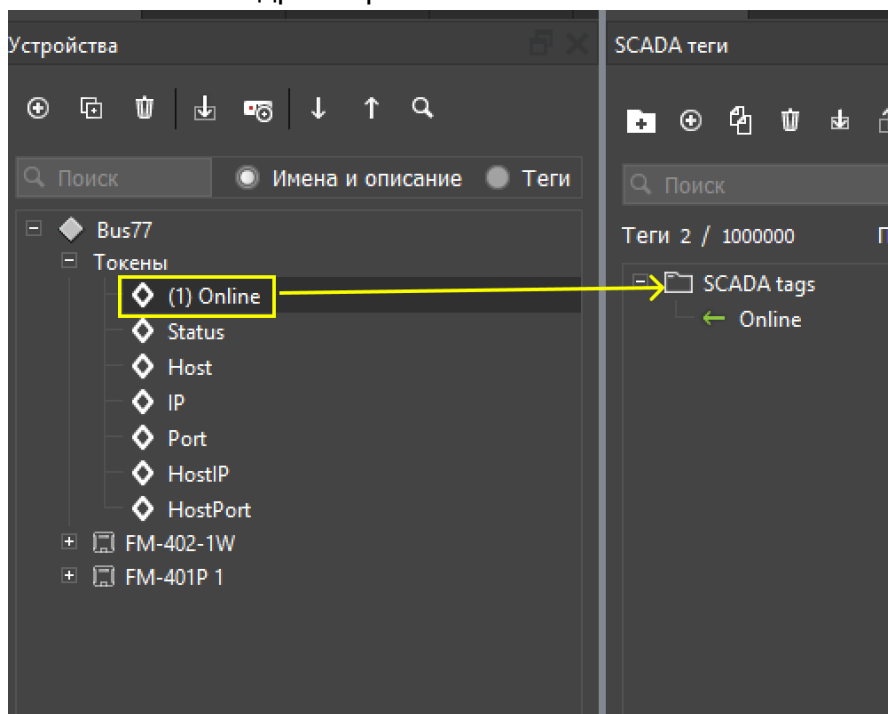
4.2. Создание SCADA-тегов

SCADA-теги можно создать двумя способами:

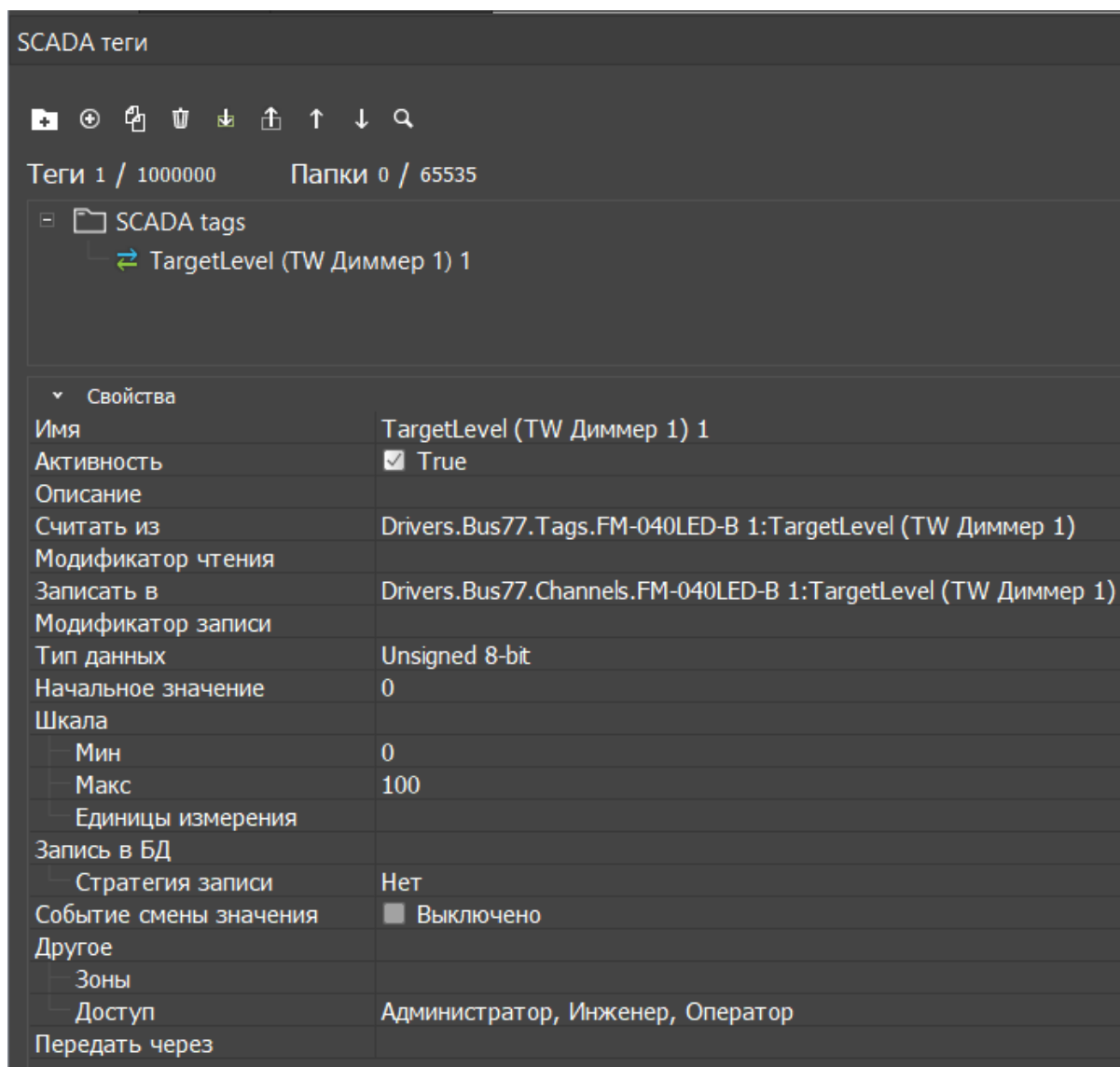
1. Нажать на иконку “+” на вкладке SCADA-теги;



2. При помощи drag-and-drop перенести в дерево тегов команды, фидбеки или токен драйвера.



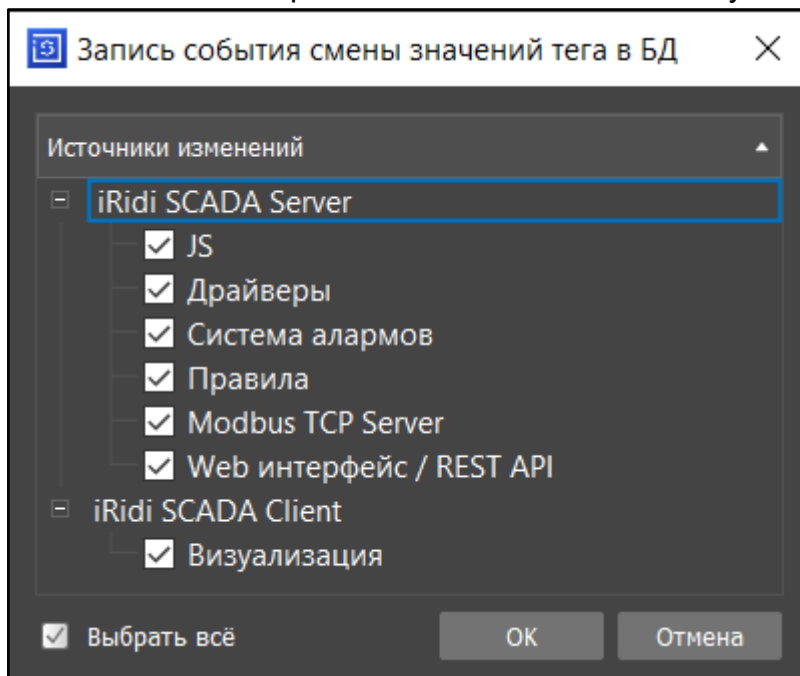
4.3. Свойства SCADA-тегов



У тега можно настроить его свойства:

- **Имя** - отображаемое имя тега;
- **Активность**:
 - True - тег опрашивается;
 - False - тег не опрашивается;
- **Описание** - описание тега;
- **Считать из** - КОС драйвера (чтение из драйвера);
- **Записать в** - КУ драйвера (запись в драйвер);
- **Тип данных** - тип данных тега;
 - Bool
 - Unsigned char
 - Signed char
 - Unsigned short
 - Signed short
 - Unsigned int

- Unsigned long
- Signed int
- Signed long
- Float 32 bit
- Float 64 bit
- String
- **Начальное значение** - значение тега при запуске сервера;
- **Запись в БД** - стратегия записи значения тега в базу данных:
 - Нет - запись в базу данных не ведётся;
 - По времени - запись в базу данных через фиксированный интервал;
 - Интервал - временной интервал записи в базу данных;
 - Зона чувствительности - запись в базу данных по изменению значения тега;
 - Значение - значение, на которое должен измениться тег для записи в БД;
- **Событие смены значения** - запись изменения значения тега в базу данных:
 - Источники изменений - выбор источников, воздействие которых на значение тега будет вызывать событие записи изменений в базу данных (со стороны iRidi SCADA Server: JS, Драйверы, Система алармов, Правила, Modbus TCP Server, Web интерфейс/REST API; со стороны iRidi SCADA Client: Визуализация);



- Зоны чувствительности - значение, на которое должен измениться тег для записи в БД;
- Другое
 - **Зоны** - зона, к которой относится тег. Зона может быть физической/реальной (например: “Подвал”, “Бойлер”, “Система вентиляции”), так и абстрактной (например: “Зона 1”). Для того

чтобы добавить новую зону, нажимаем на кнопку справа в строке “Зона”, в появившемся окне переносим нужные зоны вправо. Для добавления новой зоны нажимаем на “+” на панели инструментов.

Можно выбрать несколько зон.

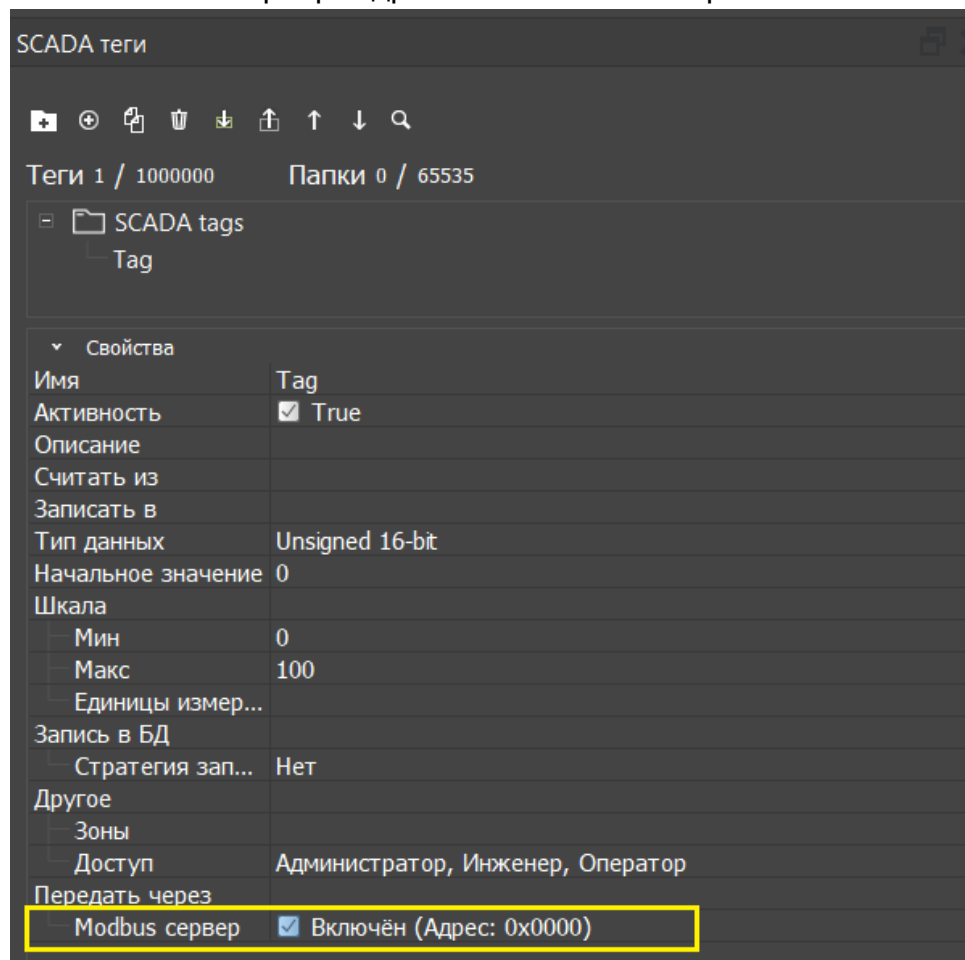
- **Доступ** - разрешение/запрет чтения и записи тега для определённых ролей пользователей (Администратор может читать и записывать все теги).
- Передать через
 - **Modbus сервер** - настройки SCADA-тега в качестве эмулятора устройства Modbus.

4.4. Modbus TCP Slave

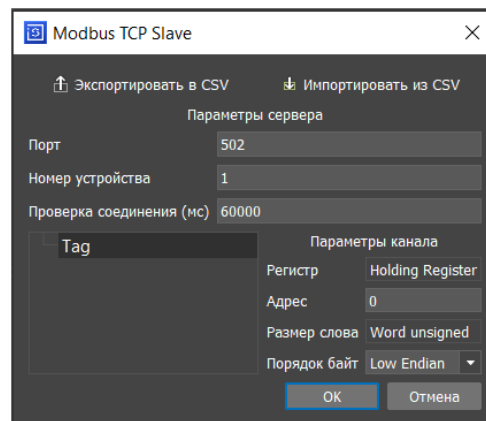
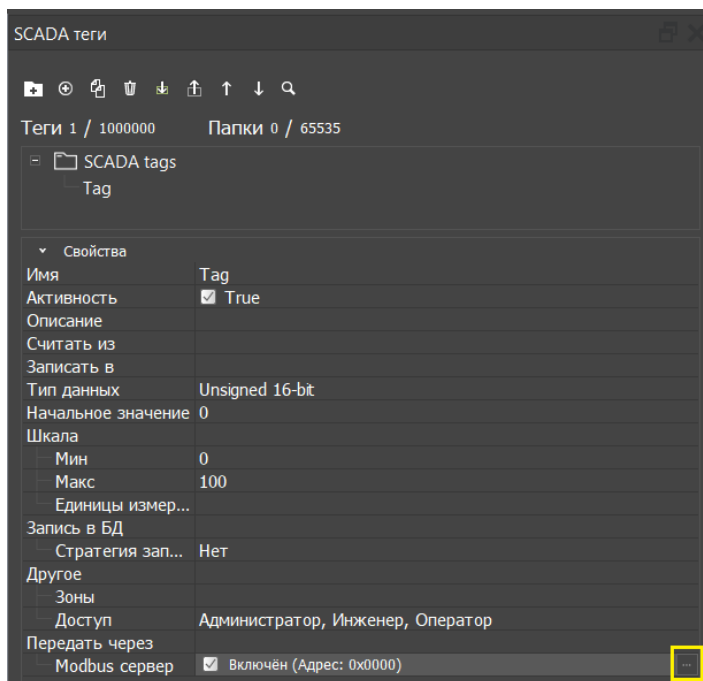
Режим “Modbus Slave” позволит iRidi SCADA Server выступать в роли устройства Modbus TCP. Это обеспечит возможность обмена данными между iRidi SCADA и внешними системами, поддерживающими протокол Modbus TCP. Также режим Modbus Slave может быть использован для эмуляции оборудования Modbus до получения реального оборудования от поставщика.

Подключение Modbus TCP Slave в свойствах тега:

- В настройках SCADA тегов активировать свойство Передать через Modbus сервер. Адрес Modbus-канала присваивается автоматически.



- Перейти в настройки устройства и модуля Modbus TCP Slave.



- Свойства модуля Modbus TCP Slave:
 - **Экспортировать в CSV** - сохранить модуль Modbus TCP Slave в *.csv формате.
 - **Импортировать из CSV** - загрузка модуля Modbus TCP Slave из документа в формате *.csv.
 - **Порт** - порт подключения к эмулируемому Modbus серверу;
 - **Номер устройства** - номер устройства эмулируемого Modbus сервером;
 - **Проверка соединения (мс)** - интервал проверки TCP соединения в миллисекундах.
- Свойства тегов модуля Modbus TCP Slave:
 - **Регистр** - регистр хранения, доступный на чтение и запись. iRidi поддерживает следующие регистры и функции:
 - **Coil Register** (0x05, 0x0F, 0x01) - регистры флагов, один бит, чтение и запись;
 - **Holding Register** (0x06, 0x10, 0x03) - регистры хранения, 16-битное слово, чтение и запись;
 - **Адрес** - адрес регистра в десятичном формате;
 - **Размер слова** - размер переменной (Word(16 bit), DWord(32 bit), Float(32 bit), Word unsigned(16 bit), DWord unsigned(32 bit));
 - **Порядок байт** - последовательность, в которой информация сохраняется и размещается в памяти оборудования (Low Endian, Big Endian, Swapped Low Endian, Swapped Big Endian).

Modbus TCP Slave

✕

↑ Экспортировать в CSV

↓ Импортировать из CSV

Параметры сервера

Порт

502

Номер устройства

1

Проверка соединения (мс)

60000

Tag

Параметры канала

Регистр

Holding Register

Адрес

0

Размер слова

Word unsigned

Порядок байт

Low Endian ▾

OK

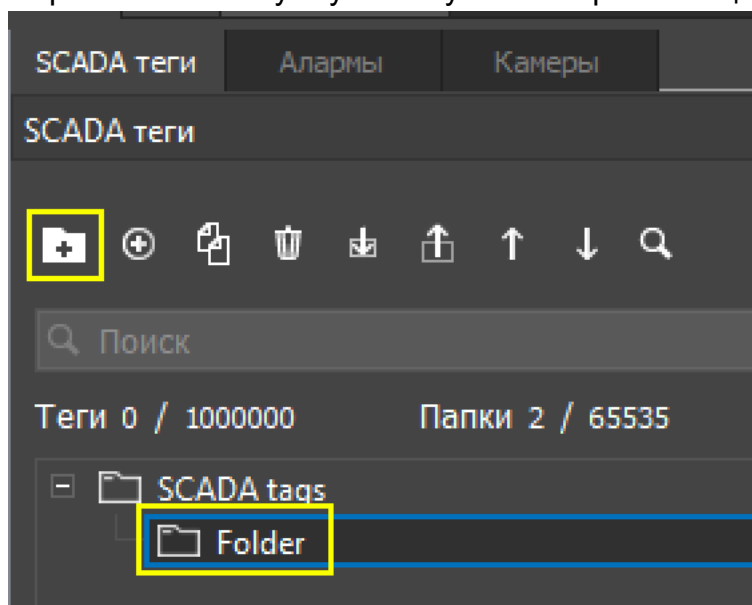
Отмена

4.5. Панель инструментов SCADA-тегов

4.5.1. Создание папок тегов

Для создания папки нужно нажать на иконку “Папка” на вкладке SCADA-теги. Папки позволяют сортировать теги по разделам и категориям, чтобы было удобно с ними работать.

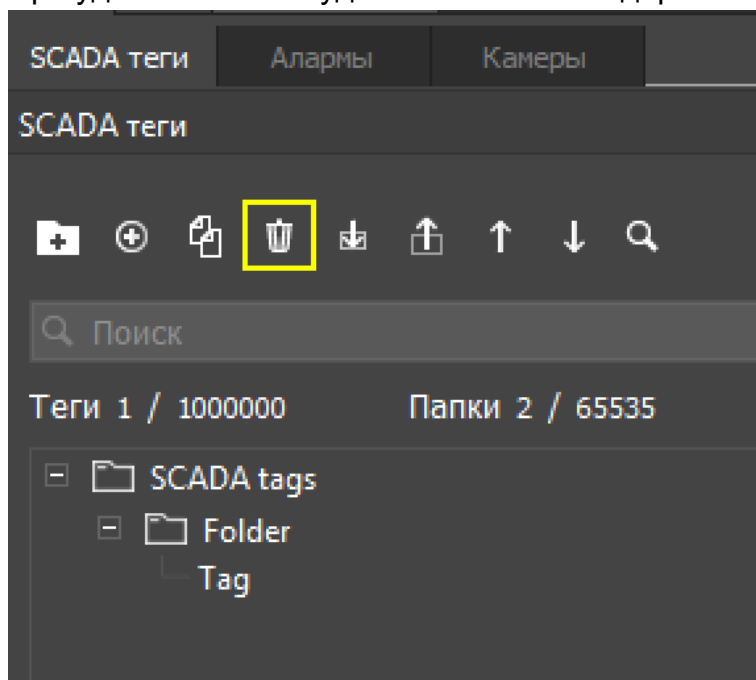
Перенести тег в нужную папку можно при помощи drag-and-drop.



4.5.2. Удаление тегов и папок

Для удаления тега или папки нужно выбрать тег или папку, которую необходимо удалить, и нажать на иконку “Корзина” на вкладке SCADA-теги.

При удалении папки удаляется всё её содержимое.

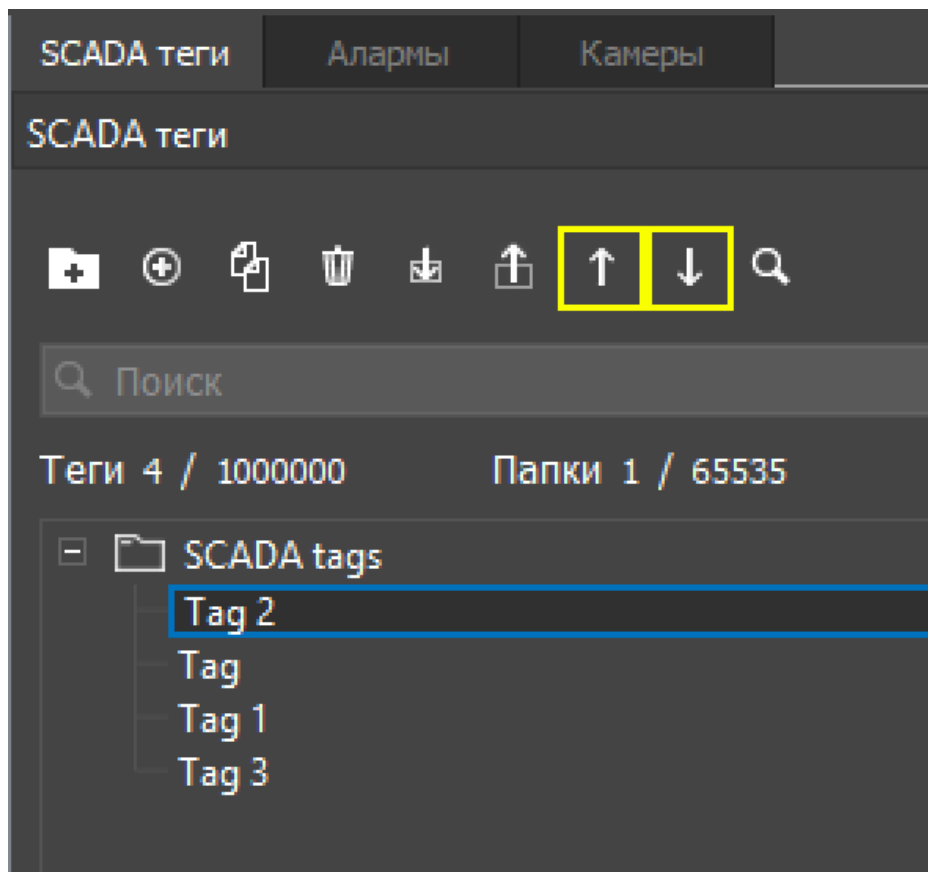


Корневую папку “SCADA tags” удалить нельзя.

4.5.3. Перемещение тегов в дереве

Перемещение тегов в дереве происходит двумя способами:

1. Перетянуть тег в нужное место;
2. Выбрать нужный тег и при помощи стрелок “вверх” и “вниз” на вкладке SCADA теги изменить положение тега.

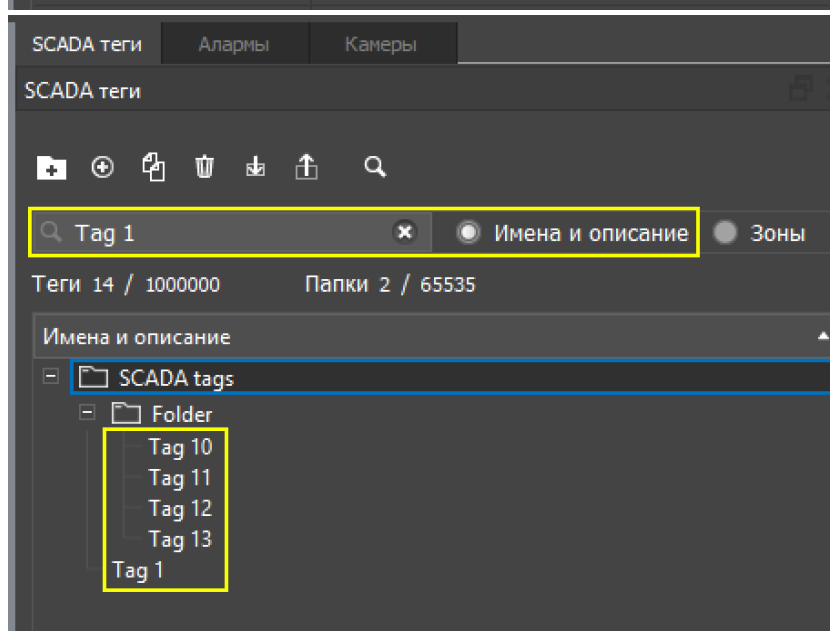
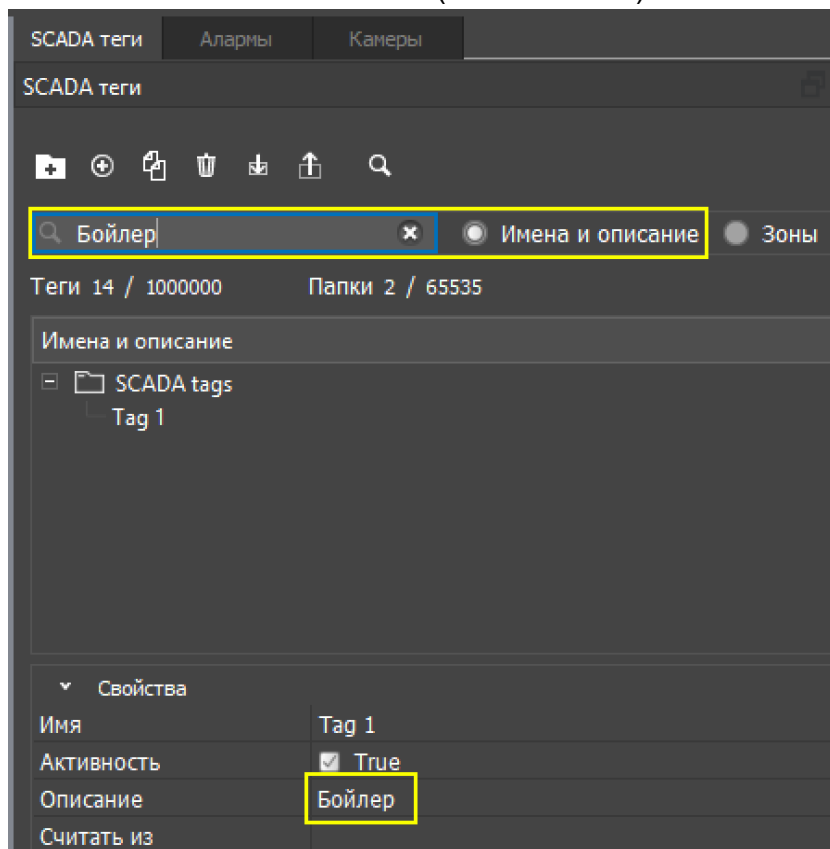


4.5.4. Поиск тегов

Поиск тегов можно осуществить по “Имени и описанию”, а также по “Зонам”.

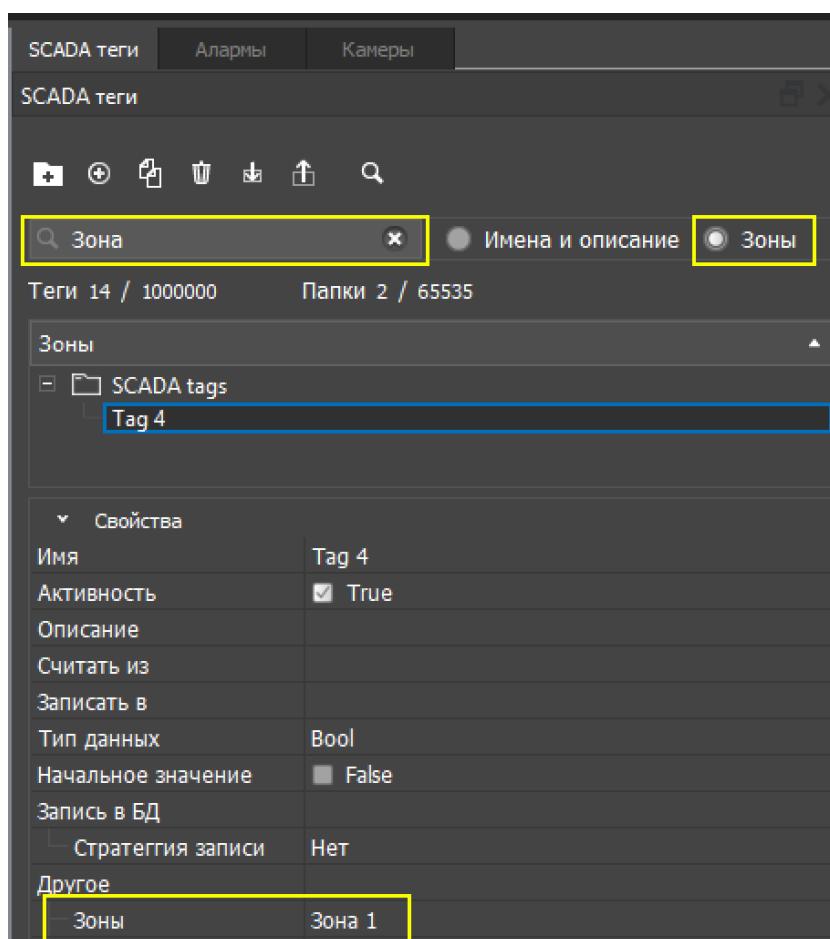
Поиск по “Имени и описанию”:

1. Выбрать правее поля поиска “Имена и описание”;
2. Ввести имя или описание (или их части) тега в поле поиска.



Поиск по “Зонам”:

1. Выбрать правее поля поиска “Зоны”;
2. Ввести имя зоны (или часть имени зоны) в поле поиска.

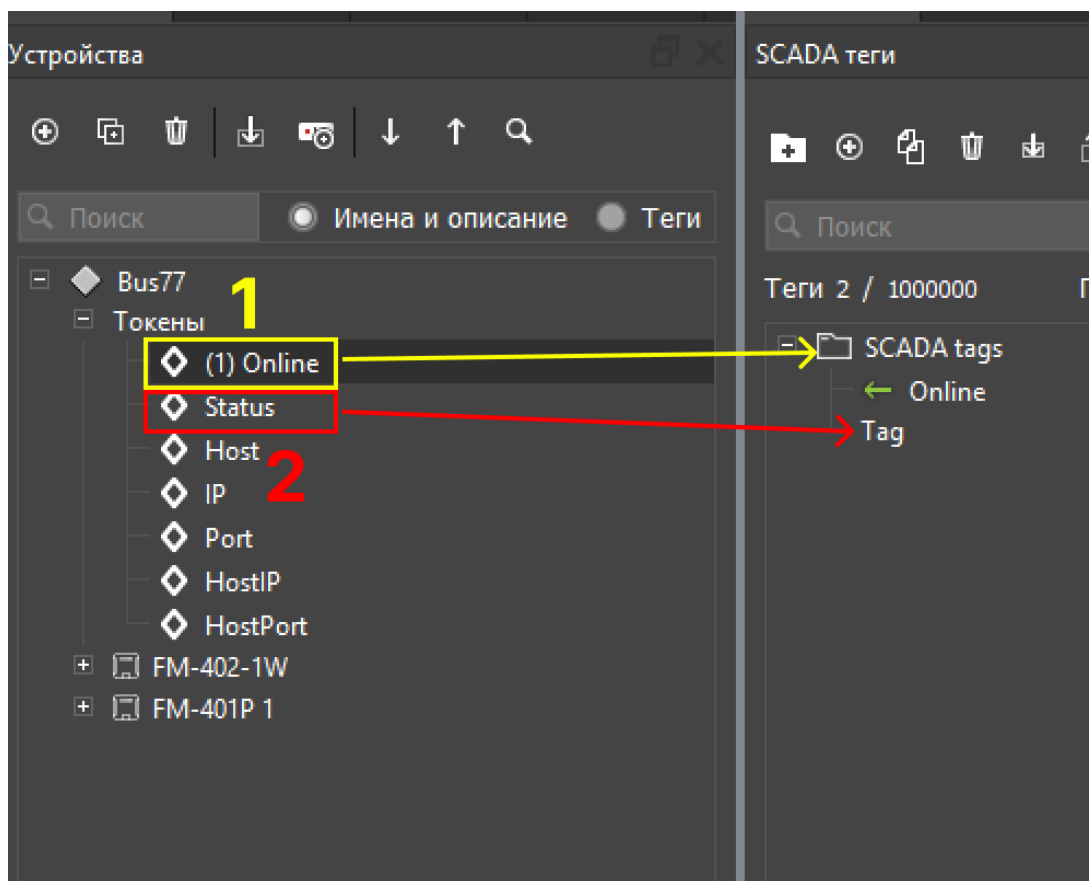


4.5.5. Работа с каналами управления, каналами обратной связи и токенами драйвера

Получить значения из КОС или записать значения в КУ драйвера можно только через SCADA-теги.

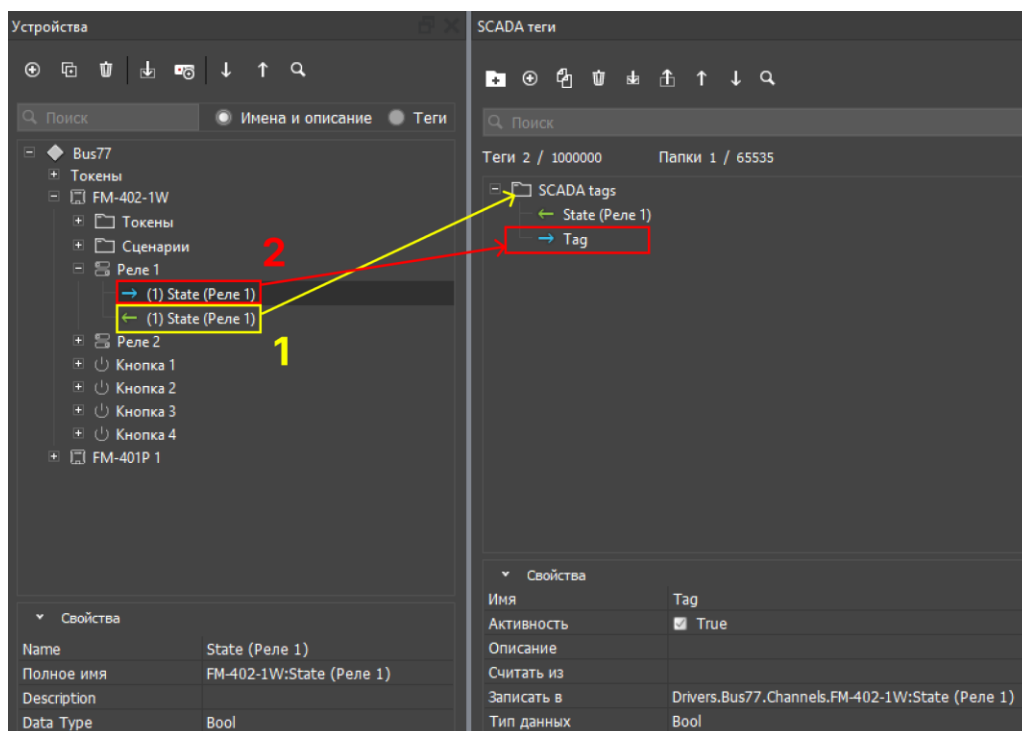
Связать токен драйвера со SCADA-тегами можно двумя способами:

1. Перенести при помощи drag-and-drop токен драйвера на папку в дереве SCADA-тегов (в дереве тегов будет создан тег с таким же именем, что и у переносимого токена);
2. Создать SCADA-тег в дереве тегов и на него при помощи drag-and-drop перенести токен драйвера.

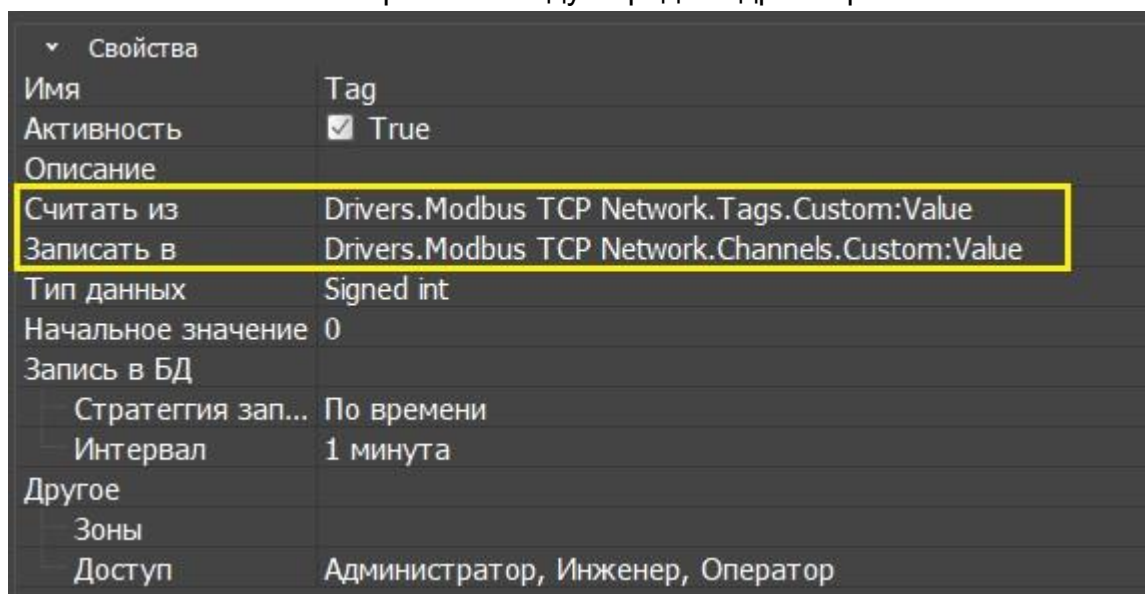


Связать фидбеки и команды драйвера со SCADA-тегами можно несколькими способами:

1. Перенести при помощи drag-and-drop фидбек или команду драйвера на папку в дереве SCADA-тегов (в дереве тегов будет создан тег с таким же именем, что и у переносимого команды и фидбека);
2. Создать SCADA-тег в дереве тегов и на него при помощи drag-and-drop перенести фидбек или команду драйвера;



3. Создать SCADA-тег в дереве тегов и в свойствах тега в полях “Считать из” и “Записать в” выбрать команду и фидбек драйвера.

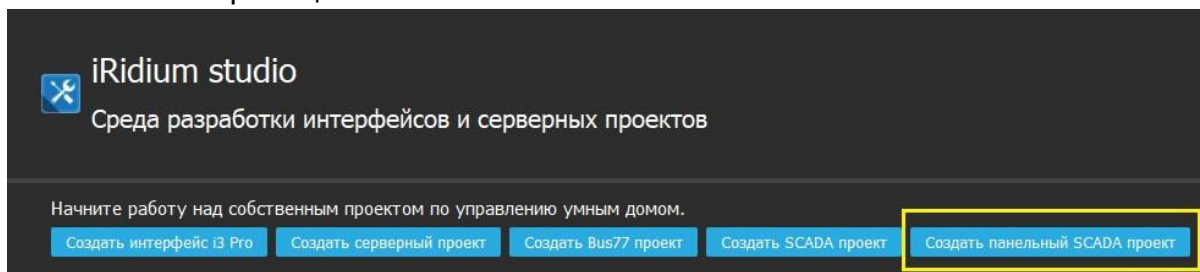


5. Создание SCADA-клиентов

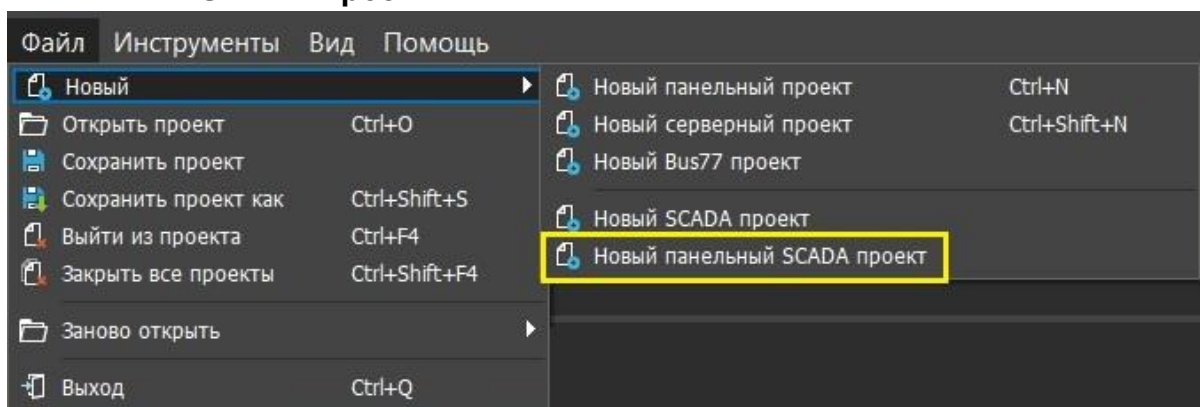
5.1. Создание проекта SCADA-клиента

Существует 2 способа создать SCADA-клиента:

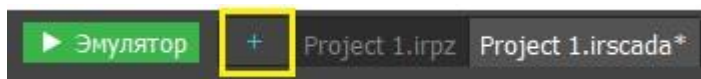
- Создать проект SCADA-клиента и затем связать его с основным SCADA-проектом. Создать проект SCADA-клиент можно:
 - Нажать кнопку **Создать панельный SCADA проект** на стартовой странице iRidi Studio.



- Воспользоваться меню **Файл** → **Новый** → **Новый панельный SCADA проект**.



- Нажать кнопку добавления проекта на панели управления проектами. Выбрать тип в меню типа проекта - Панель SCADA.



Новый проект

Основные

Название проекта: Project 1

Тип: Панель SCADA

Панель

Тип устройства: Custom

Ширина: 1024

Высота: 768

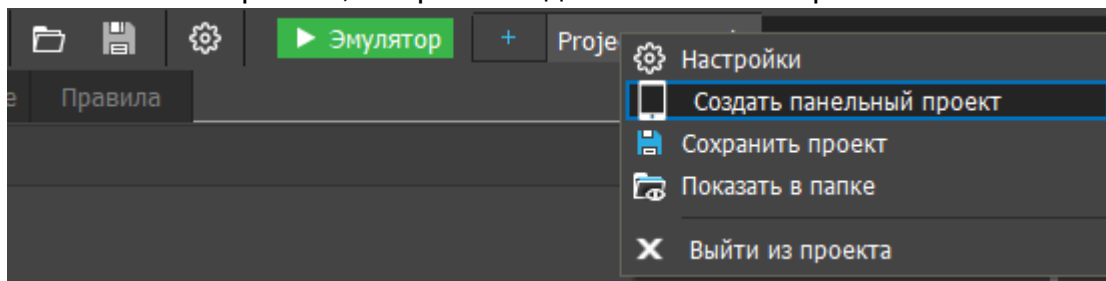
Главная страница

Имя страницы: Page 1

Ориентация: Альбомная

OK Отмена

- Создать проект SCADA-клиента, который сразу будет связан с SCADA-проектом. Для этого необходимо воспользоваться контекстным меню SCADA-проекта, выбрав “Создать панельный проект”.



В появившемся окне ввести: название проекта, тип - Panel SCADA (встает автоматическое); тип устройства и его габаритные размеры; имя страница и ее ориентацию.

Новый проект

Основные

Название проекта:

Тип: Панель SCADA

Панель

Тип устройства: Custom

Ширина:

Высота:

Главная страница

Имя страницы:

Ориентация: Альбомная

OK Отмена

Во вновь созданном проекте в устройствах появится iRidi SCADA Server со всеми тегами и токенами.

Устройства

+ 📁 🗑️ 📥 📤 ↺ ↓ ↑ 🔍

+ 📺 iRidium SCADA Server

▼ Об устройстве

Name	iRidium SCADA Server
Driver Type	iRidium SCADA Server
Protocol	TCP
Source Project	Project 1.irscada
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	30464
Login	
Password	
Wait for Connect (ms)	0
Wait for Data (ms)	0
Keep Alive	60000
Mode	Direct Connection
Tags	

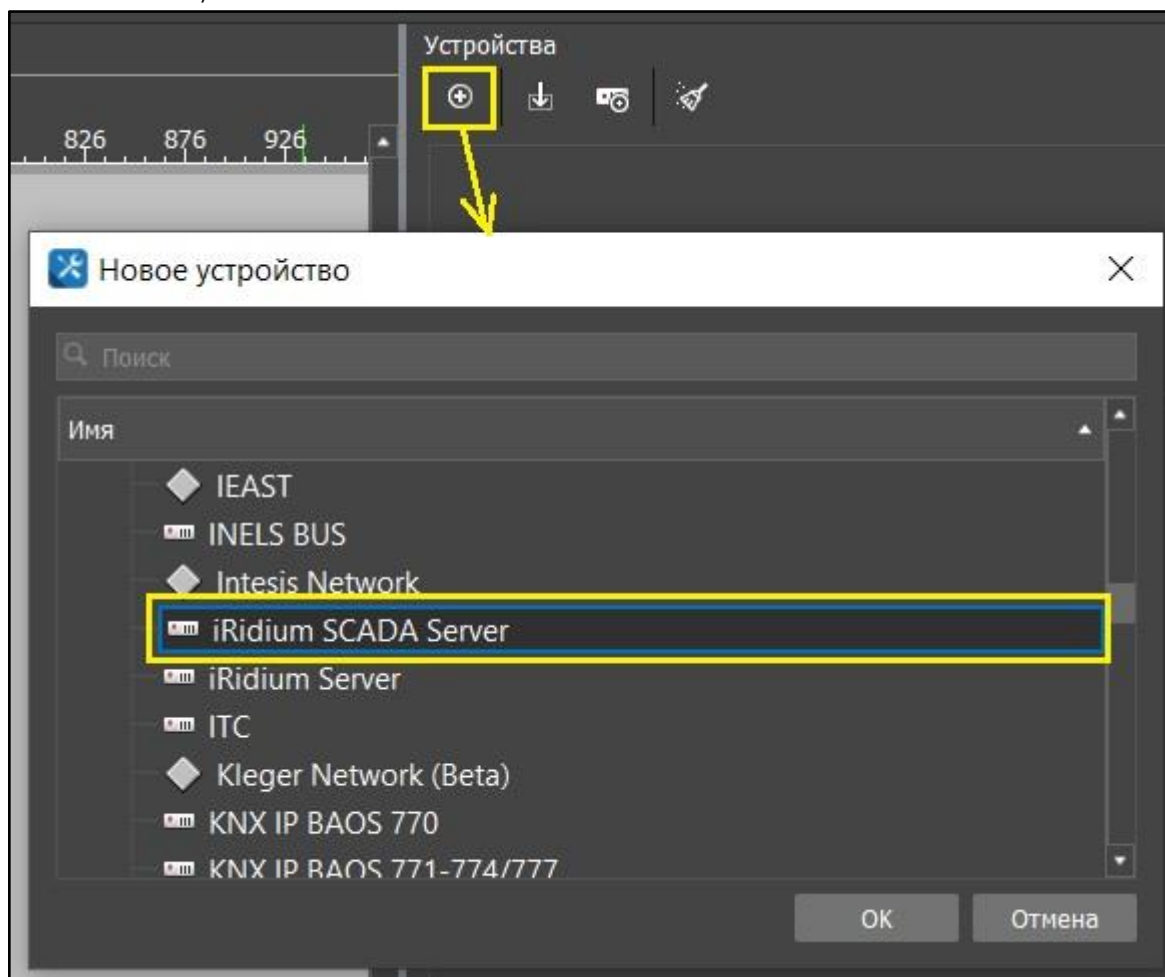
Для обновления КУ и КОС драйвера после внесения изменений в исходный проект нужно синхронизировать драйвер.



5.2. Связь существующего проекта SCADA-клиента с SCADA-проектом

Для связи уже существующего SCADA-клиента со SCADA-сервером необходимо:

1. В проекте SCADA-клиента добавить драйвер iRidi Server/iRidi SCADA Server;



2. В свойствах драйвера указать исходный SCADA-проект;

▼ Об устройстве

Name	iRidium SCADA Server
Driver Type	iRidium SCADA Server
Protocol	TCP
Source Project	
Local Connection	
Log Level	Project 1.irscada
Host	127.0.0.1
Port	30464
Login	
Password	
Wait for Connec...	0
Wait for Data (...)	0
Keep Alive	60000
Mode	Direct Connection
Tags	

3. Настроить свойства драйвера;

▼ Об устройстве

Name	iRidium SCADA Server
Driver Type	iRidium SCADA Server
Protocol	TCP
Source Project	Project 1.irscada
Local Connection	
Log Level	Emergency
Host	127.0.0.1
Port	30464
Login	
Password	
Wait for Connec...	0
Wait for Data (...)	0
Keep Alive	60000
Mode	Direct Connection
Tags	

4. После добавления в проект драйвера iRidi Server/iRidi SCADA Server и его настройки нужно синхронизировать драйвер с исходным проектом. Для синхронизации нужно нажать кнопку “Синхронизировать” для получения КУ и КОС драйвера.

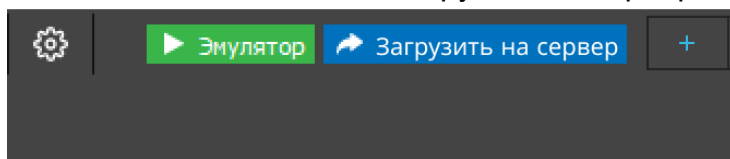
Для обновления КУ и КОС драйвера после внесения изменений в исходный проект также нужно синхронизировать драйвер.



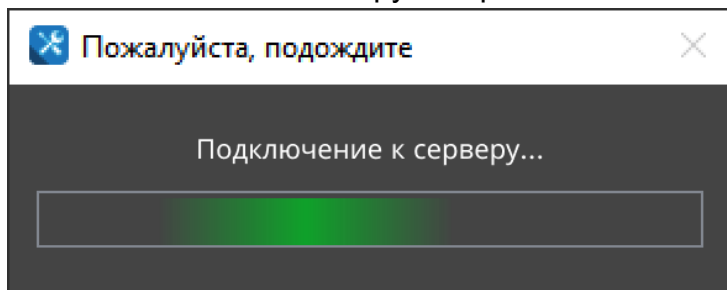
5.3. Загрузка проекта SCADA клиента

Для загрузки проекта SCADA-клиента на сервер:

1. Нажатием кнопки “Загрузить на сервер”

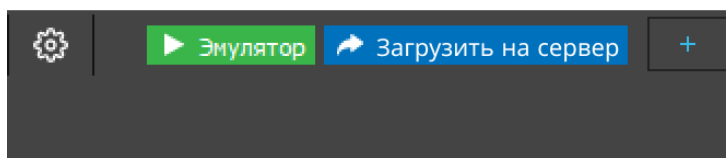


2. Появится окно загрузки проекта:



3. После завершения загрузки проект будет загружен на сервер.

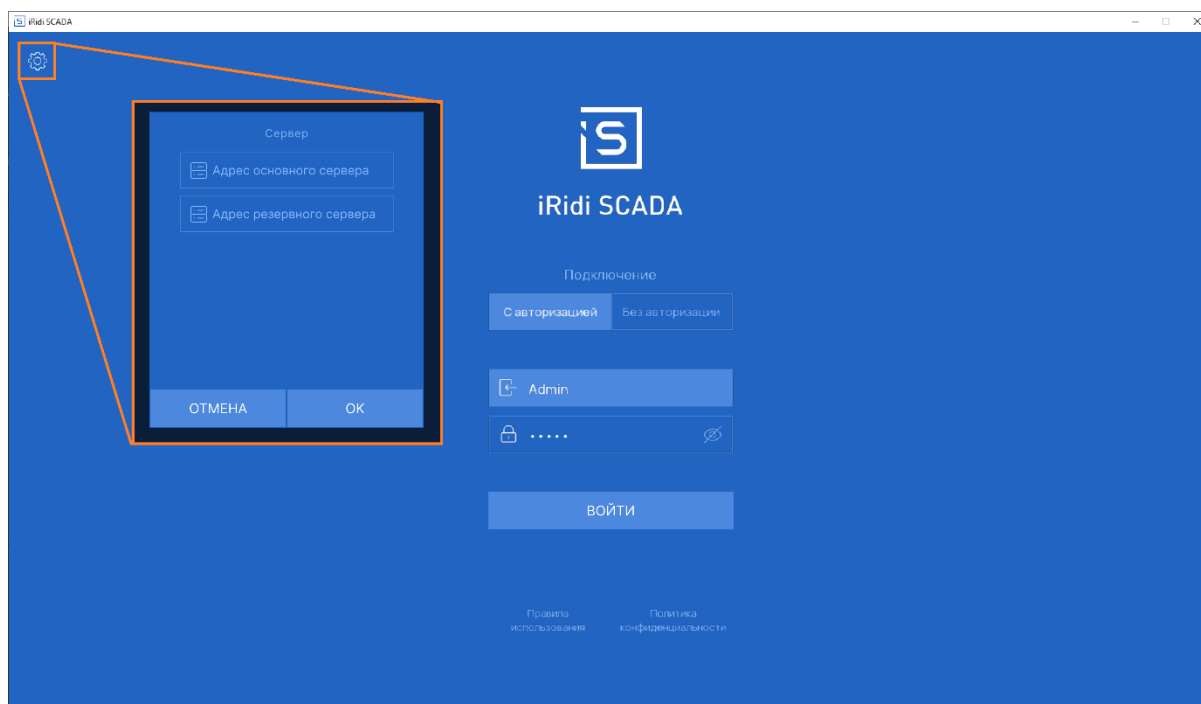
Для запуска Эмулятора:



1. Нажатием кнопки “Эмулятор”
2. Проект будет запущен.

5.4. Добавление и сохранение адреса сервера в SCADA клиенте

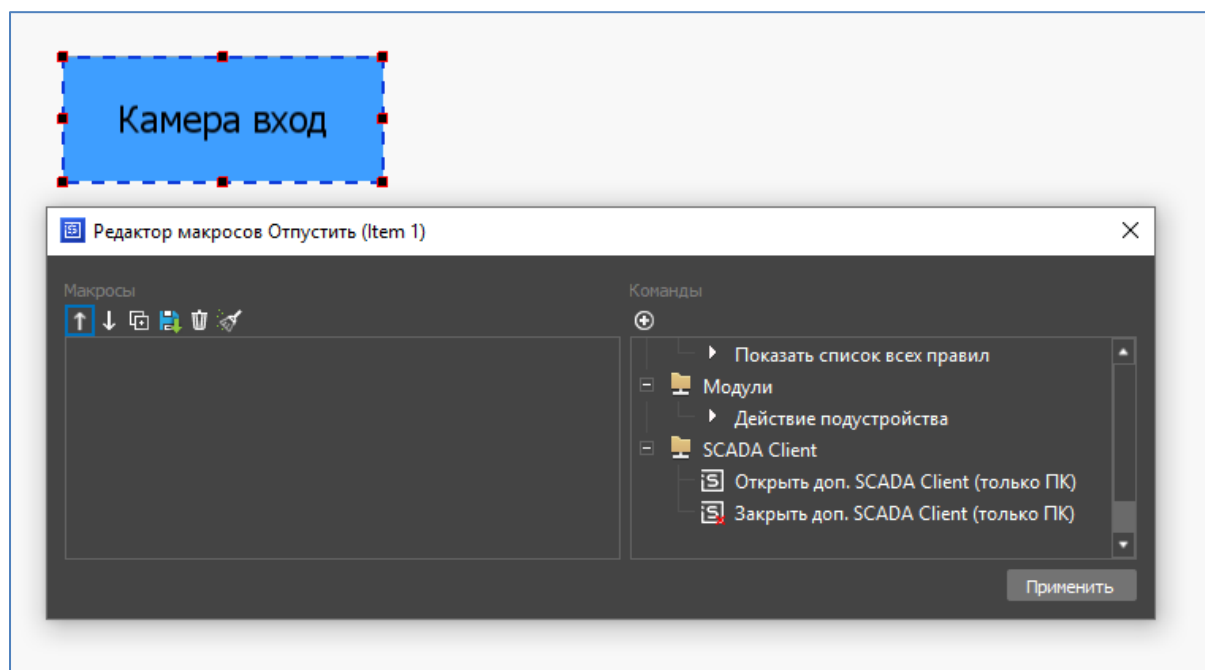
Для добавления, изменения и сохранения адреса сервера в клиенте необходимо нажать соответствующую кнопку в левом верхнем углу формы авторизации Клиента.



Далее необходимо заполнить поля «Адрес основного сервера» и «Адрес резервного сервера».

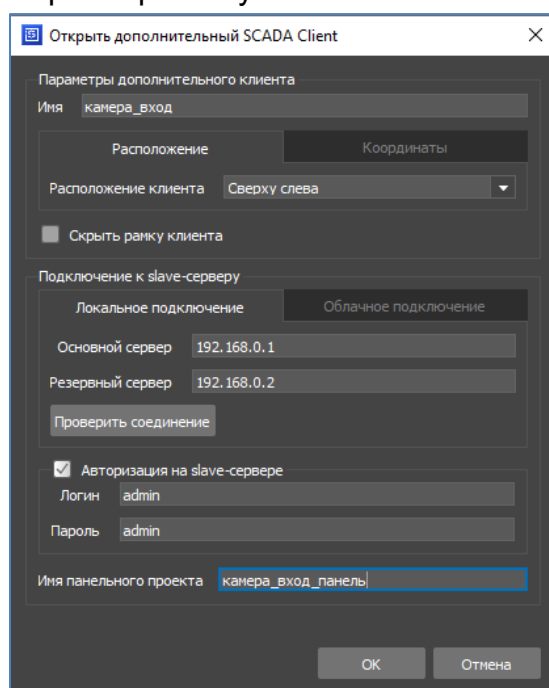
5.5. Диспетчеризация

Настройки вызова дополнительного клиента осуществляются через меню программирования элемента.



5.5.1 Для запуска Клиента - Команда SCADA Client. Открыть доп. SCADA Client (только ПК).

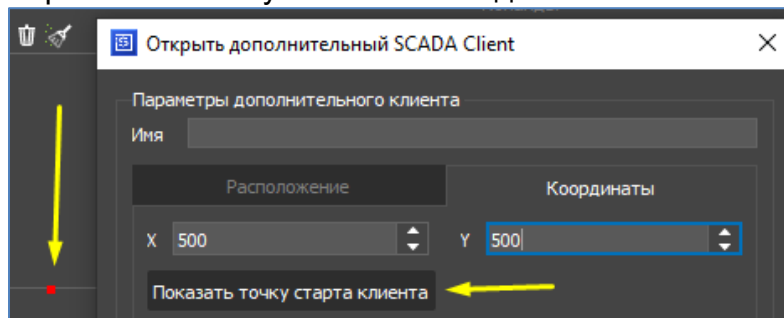
Параметры запуска SCADA Client



«Имя» – Наименование запущенной копии Клиента;

Вкладка «Расположение клиента» – Место расположения Клиента после запуска (Работает Только на главном/основном дисплее);

Вкладка «Координаты» - Задаёт координаты левой верхней точки относительно верхнего левого угла основного дисплея положения Клиента при запуске.

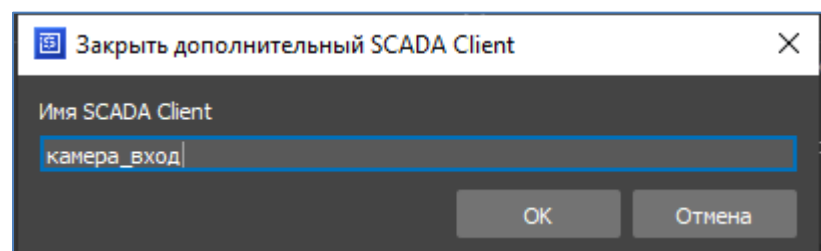


«Скрыть рамку клиента» - скрывание рамки Клиента;

«Подключение к slave-серверу» - параметры подключения к Серверу. IP адреса «Основной сервер» и «Резервный сервер»;

«Авторизация на slave-сервере» - Логин и Пароль от учётной записи на slave-сервере.

5.5.2. Для закрытия Клиента - Команда SCADA Client. Закрывать доп. SCADA Client (только ПК).



«Имя SCADA Client» - Имя Клиента, использованное для вызова Клиента.

6. Аварийные сообщения (Алармы)

6.1. Описание

Аварийные сообщения (алармы) - это сообщения, предупреждающее оператора о возникновении определенной ситуации, которая может привести к серьезным последствиям, и потому требующее его внимания и вмешательства.

6.2. Группы аварийных сообщений (группы алармов)

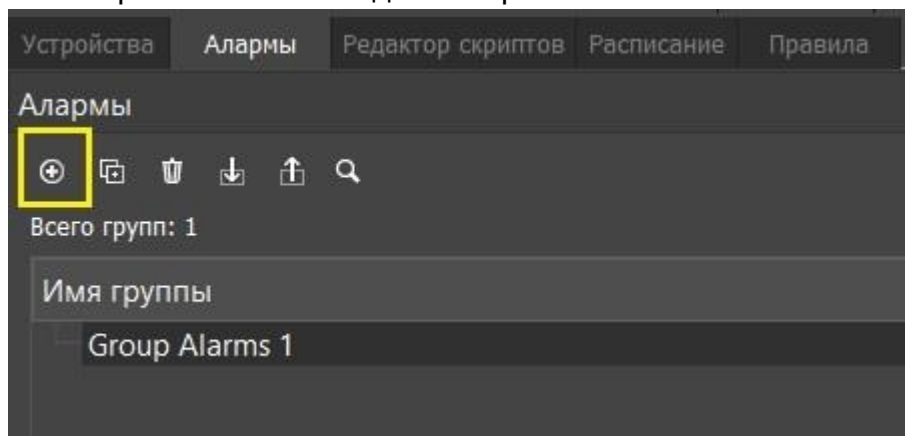
Каждое аварийное сообщение (аларм) связано с определенной логической группой аварийных сообщений (алармов). Эти группы определяются пользователем. Создание групп аварийных сообщений (алармов) позволяет сгруппировать аварийные сообщения (алармы) в зависимости от их организации, схемы размещения оборудования, приоритетов и любых других признаков. Группы аварийных сообщений (алармов) являются полезным средством фильтрации вывода информации об аварийных сообщениях (алармах) на экран дисплея или принтер.

Группы аварийных сообщений (алармов) делятся на 4 типа:

- Аналоговый 4 уставки (HiHi, Hi, Lo, LoLo);
- Аналоговый 2 уставки (Hi, Lo);
- Дискретный прямой;
- Дискретный инверсный;

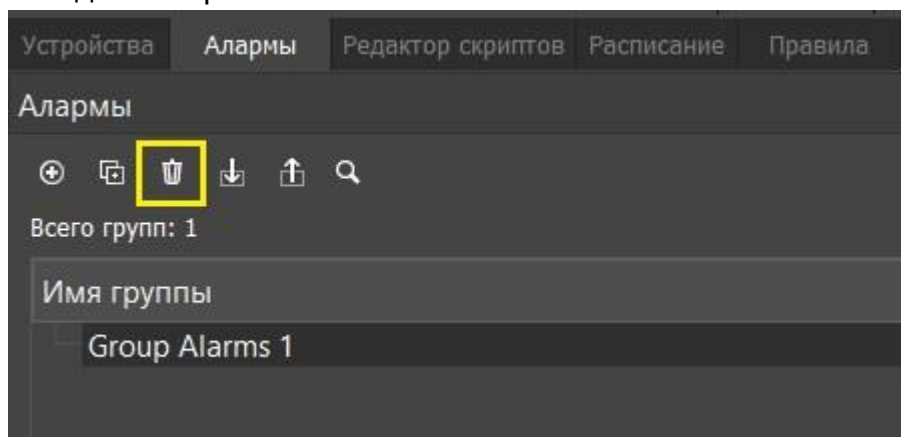
6.2.1. Создание группы аварийных сообщений (алармов)

Для создания группы аварийных сообщений (алармов) нужно нажать на иконку “+” в верхней части вкладки “Алармы”.



6.2.2. Удаление группы аварийных сообщений (алармов)

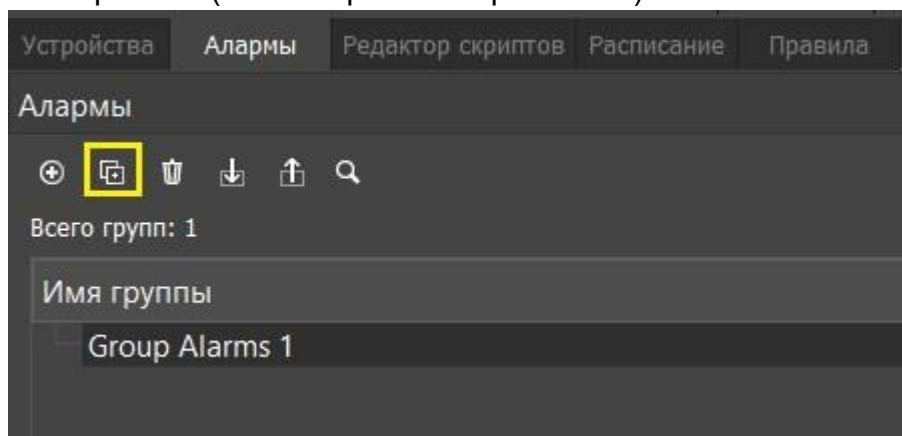
Для удаления группы аварийных сообщений (алармов) нужно выделить группу аварийных сообщений (алармов) и нажать на иконку “Корзина” в верхней части вкладки “Алармы”.



6.2.3. Клонирование группы аварийных сообщений (алармов)

Для клонирования группы аварийных сообщений (алармов) нужно выделить группу аварийных сообщений (алармов) и нажать на иконку “Клонировать” в верхней части вкладки “Алармы”.

Создаётся группа аварийных сообщений (алармов), аналогичная той, которую клонировали (все настройки сохраняются).

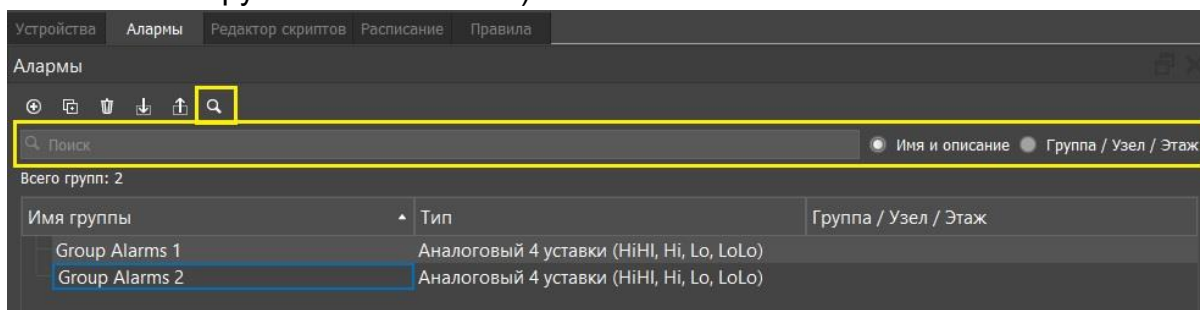


6.2.4. Поиск по группам аварийных сообщений (алармов)

Для поиска групп аварийных сообщений (алармов) нужно:

1. Нажать на иконку “увеличительное стекло” в верхней части вкладки “Алармы”.

После нажатия на иконку появится строка поиска и 2 радиокнопки (“Имя и описание” и “Группа / Узел / Этаж”).



2. Ввести имя, описание или группу / узел / этаж в строку поиска;
3. Выбрать критерий поиска:
 - по имени или описанию нужно нажать радиокнопку “Имя и описание”
 - по группе / узлу / этажу нужно нажать радиокнопку “Группа / Узел / Этаж”.

6.2.5. Импорт и экспорт групп аварийных сообщений

Для импорта групп аварийных сообщений:



1. Нажать на кнопку  в верхней части вкладки “Алармы”.
2. В открывшемся окне выбрать файл для импорта в формате *.csv.
3. Группа аварийных сообщений появится в проекте.

Для экспорта групп аварийных сообщений:



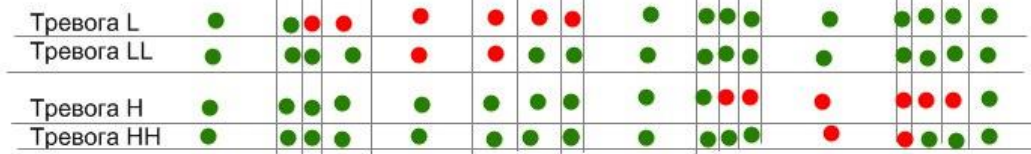
1. Нажать на кнопку  в верхней части вкладки “Алармы”.
2. В открывшемся выбрать место сохранения файла и задать имя.
3. Группы аварийных сообщений будут сохранены в формате *.csv.

6.3. Настройки группы аварийных сообщений (алармов)

▼ Настройки группы алармов	
Имя	Group Alarms 1
Тип по кол-ву уставок	Аналоговый 4 уставки (HiHi, Hi, Lo, LoLo)
Активность	☒ Включить
Тип контроля	По значению
Описание	
Приоритет HiHi (0-999)	0
Приоритет Hi (0-999)	0
Приоритет Lo (0-999)	0
Приоритет LoLo (0-999)	0
Гистерезис	1.00
Задержка аварии (сек)	0
Группа / Узел / Этаж	
Подтверждение	☒ Включить подтверждение аварий
Комментарий при квитировании	без комментария

Параметры настройки группы аварийных сообщений:

- **Имя** - имя группы аварийных сообщений (алармов);
- **Тип по кол-ву уставок** - выбор количества уставок у группы:
 - Аналоговый 4 уставки (HiHi, Hi, Lo, LoLo);
 - Аналоговый 2 уставки (Hi, Lo);
 - Дискретный прямой;
 - Дискретный инверсный;
- **Активность** - включение/отключение активности группы алармов:
 - Включить
 - Выключить;
- **Тип контроля** - на данный момент доступен только тип контроля - По значению; в дальнейшем появиться тип контроля - По скорости изменения (пока не реализовано);
- **Описание** - описание группы;
- **Приоритет (0-999)** - важность данного аларма (наиболее серьезные алармы имеют приоритет 1). При добавлении тега в группу алармов приоритет автоматически присваивается тегу;
- **Гистерезис** - порог нечувствительности при фиксации срабатывания аларма.



- **Комментарии при квитировании** - запрашивать от пользователя комментарии при квитировании аварийного сообщения:
 - Без комментариев - квитировать сообщение без запроса комментариев;
 - Комментирование не обязательно - пользователь может оставить комментарии при квитировании группы аварийных сообщений или нет (на свое усмотрение);
 - Комментирование обязательно - пользователь обязательно должен оставить комментарии при квитировании группы аварийных сообщений.

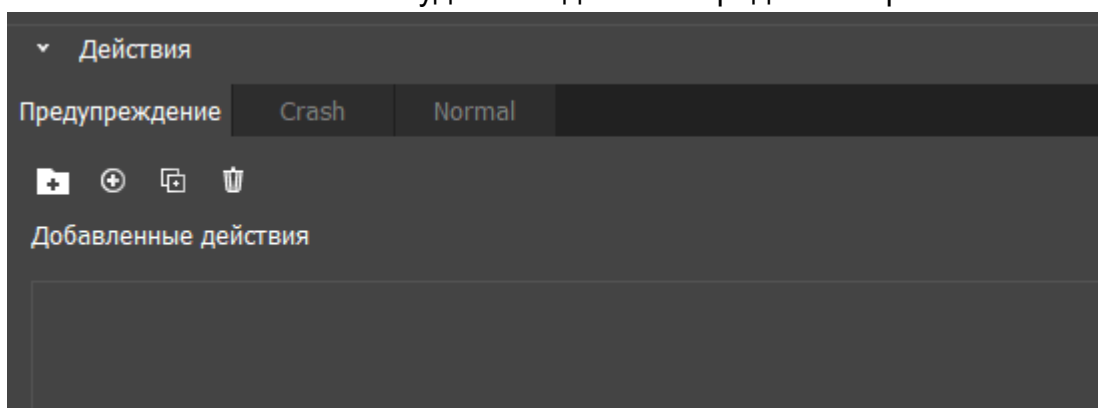
6.4. Действия

6.4.1. Описание основных групп действий

Действия - это те события, которые будут выполнены системой при возникновении аварийных ситуаций.

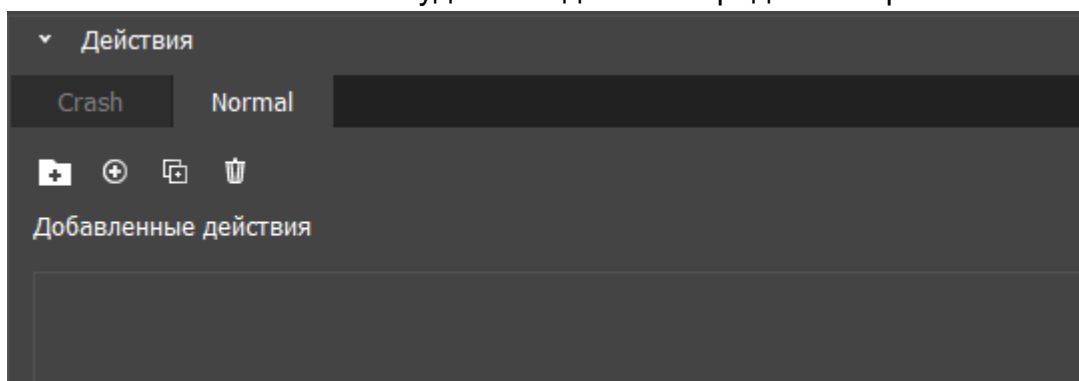
У группы аналоговых аварийных сообщений с 4 уставками можно настроить на следующие группы действий:

- **Предупреждение** - это группа действий, которые будут выполняться в том случае, если значение тега будет выше значения hi/ниже значения lo;
- **Авария** - это группа действий, которые будут выполняться в том случае, если значение тега будет выше значения hihi/ниже значения lolo;
- **Норма** - это группа действий, которые будут выполняться в том случае, если значение тега будет находиться в пределах нормы.



У группы аналоговых аварийных сообщений с 2 уставками можно настроить на следующие группы действий:

- **Авария** - это группа действий, которые будут выполняться в том случае, если значение тега будет выше значения hi/ниже значения lo;
- **Норма** - это группа действий, которые будут выполняться в том случае, если значение тега будет находиться в пределах нормы.

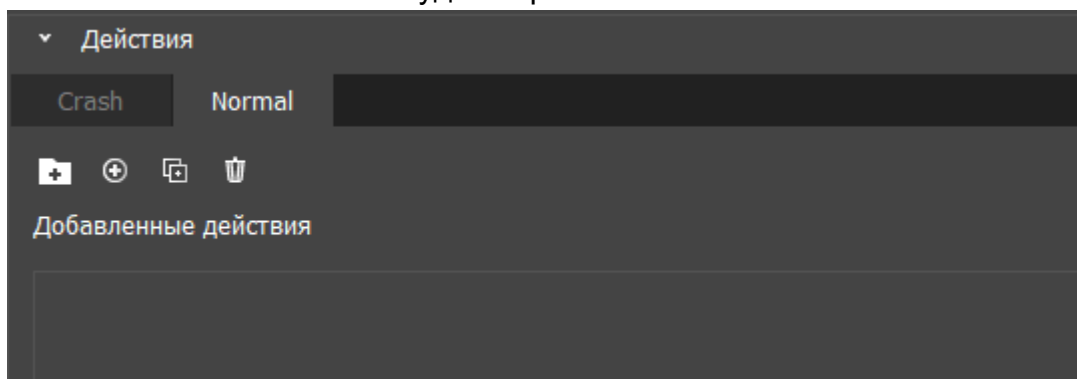


У группы дискретных прямых и дискретных инвертированных аварийных сообщений можно настроить на следующие группы действий:

- **Авария** - это группа действий, которые будут выполняться в том случае, если значение тега будет равно 1 для дискретных прямых аварийных

сообщений и равно 0 для дискретных инвертированных аварийных сообщений;

- **Норма** - это группа действий, которые будут выполняться в том случае, если значение тега будет норма

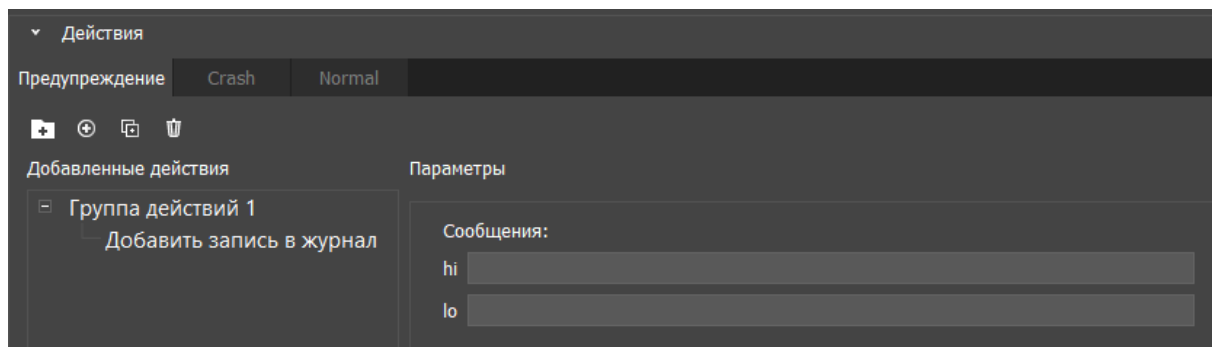


При возникновении аварии, предупреждений или нормализации события можно выполнить ряд действий:

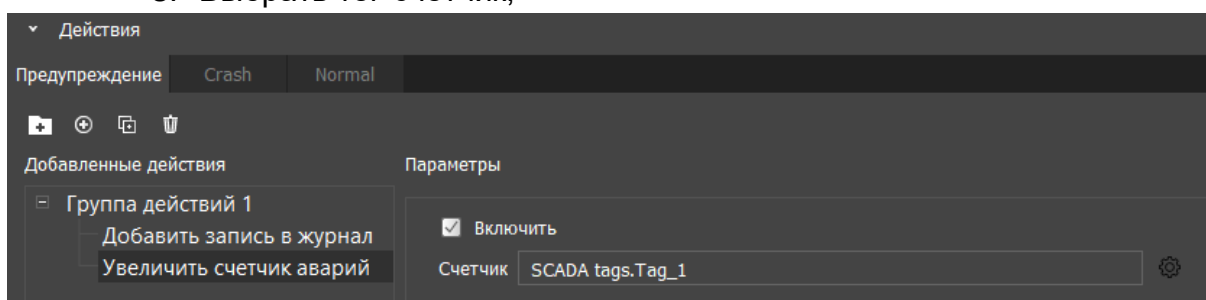
- **Добавить запись в журнал.** Если текстовое поле оставить пустым, то в журнал аварий будет записано системное сообщение:
 - Hihi: "Значение @tagName достигло @HihiAlarm @unit";
 - Hi: "Значение @tagName достигло @HiAlarm @unit";
 - Lo: "Значение @tagName достигло @LoAlarm @unit";
 - Lolo: "Значение @tagNam достигло @LoLoAlarm @unit";
 - Авария: "@tagName в состоянии аварии";
 - Норма: "Значение @tagName нормализовано";
 - Квитирование: "Аларм @tagName подтверждён".

Примечание: при создании записи в журнал можно использовать следующие параметры:

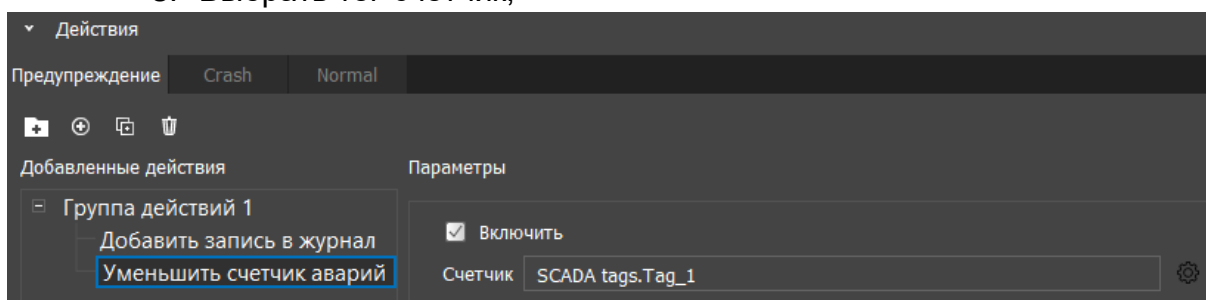
- @name - имя тега
- @V - Целая часть текущего значения тега (активировавшее аларм)
- @F1...@F5 - Текущее значение тега с плавающей точкой (1...5 -- кол-во знаков после точки)
- @unit - единицы измерения значений тега (значение одноимённого свойства тега, блок "Шкала")
- @bound - имя границы (lo, lolo, hi, hihi), для дискретных - пустое
- @e1...@e5 - ожидаемое значение границы с плавающей точкой (1...5 -- кол-во знаков после точки)
- @state - состояние аларма (Норма, Предупреждение, Авария)
- @p - приоритет (число, 0-999)
- @confirm - необходимость квитирования (да, нет)
- @time(mm:hh DD/MM/YYYY) - время срабатывания аларма с указанием формата даты и времени



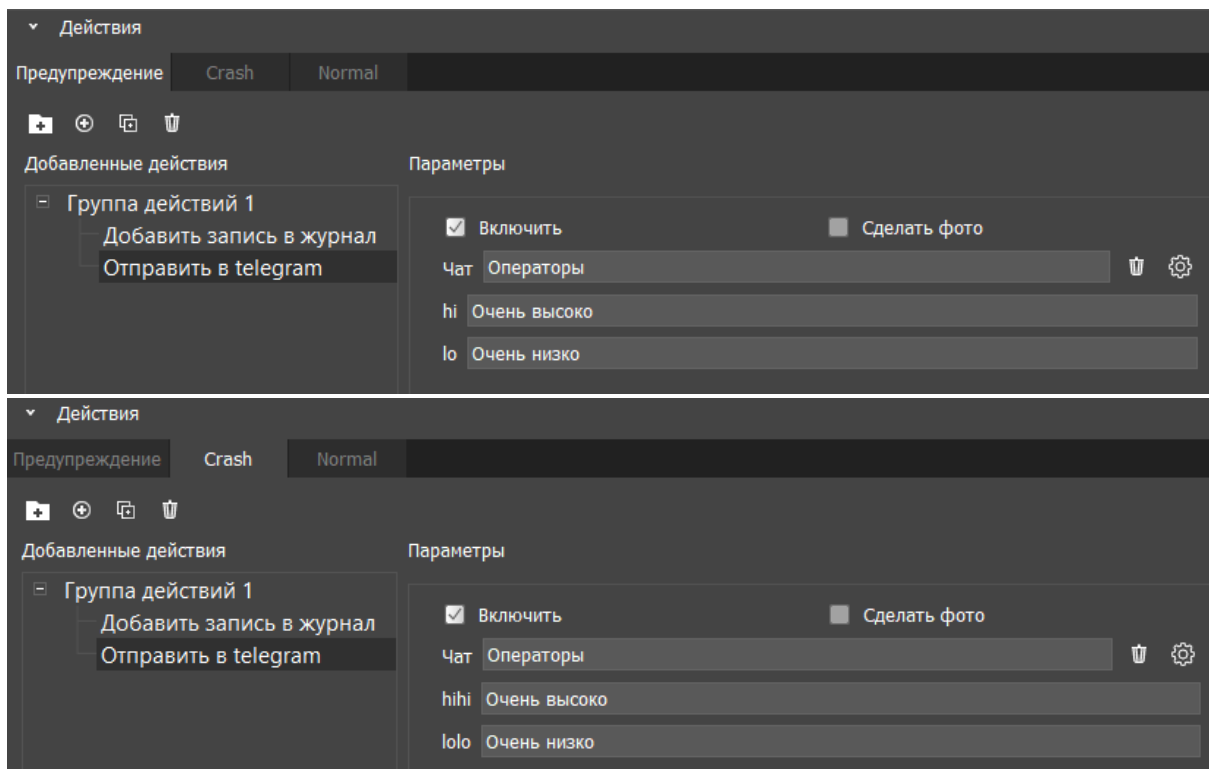
- **Увеличить счётчик аварий** - увеличивает значение тега-счётчика на 1.
 1. Добавить действие “Увеличить счётчик аварий”;
 2. Включить действие;
 3. Выбрать тег-счётчик;



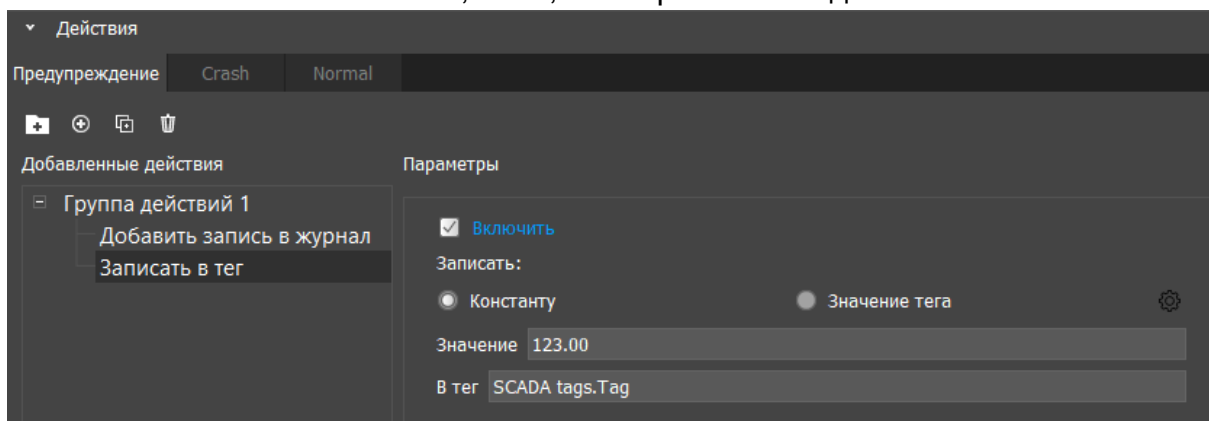
- **Уменьшить счётчик аварий** - уменьшает значение тега-счётчика на 1.
 1. Добавить действие “Уменьшить счётчик аварий”;
 2. Включить действие;
 3. Выбрать тег-счётчик;

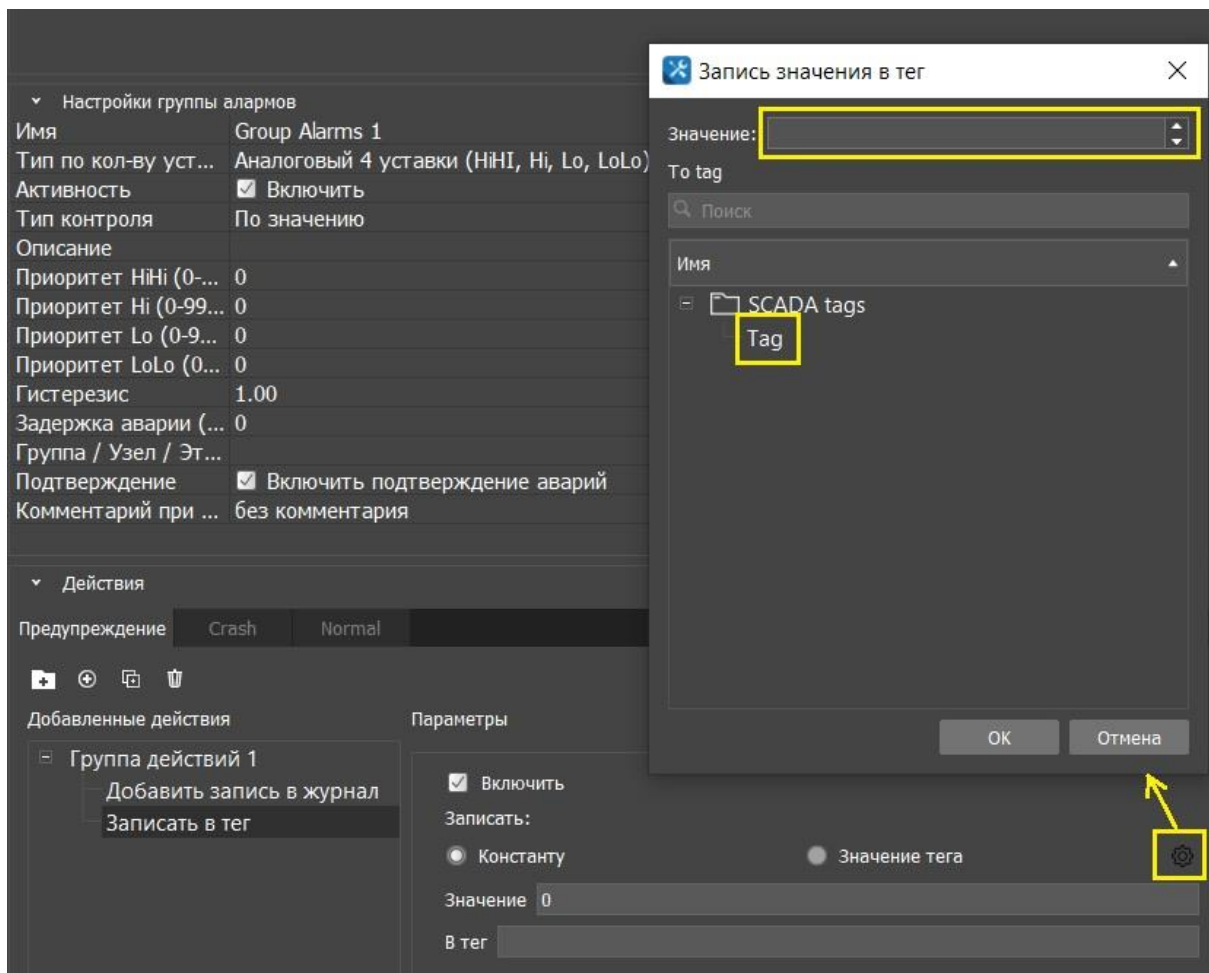


- **Отправит в telegram** - отправка сообщения об ошибке в Telegram.
 1. Добавить действие “Отправит в telegram”;
 2. Включить действие;
 3. Выбрать Telegram-чат (добавление чатов telegram);
 4. Добавить текст сообщений.



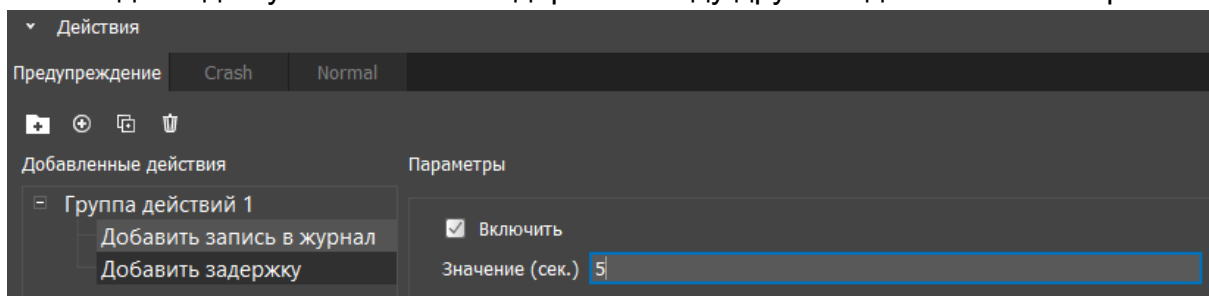
- **Записать в тег** - записывает значение (константное или из другого тега) в тег.
 1. Добавить действие “Записать в тег”;
 2. Включить действие;
 3. Выбрать, что записать (константа или значение тега);
 4. Нажать на “Шестерёнку” и указать константу или тег, из которого считать значение, и тег, в который необходимо записать значение.





- Добавить задержку (сек.)

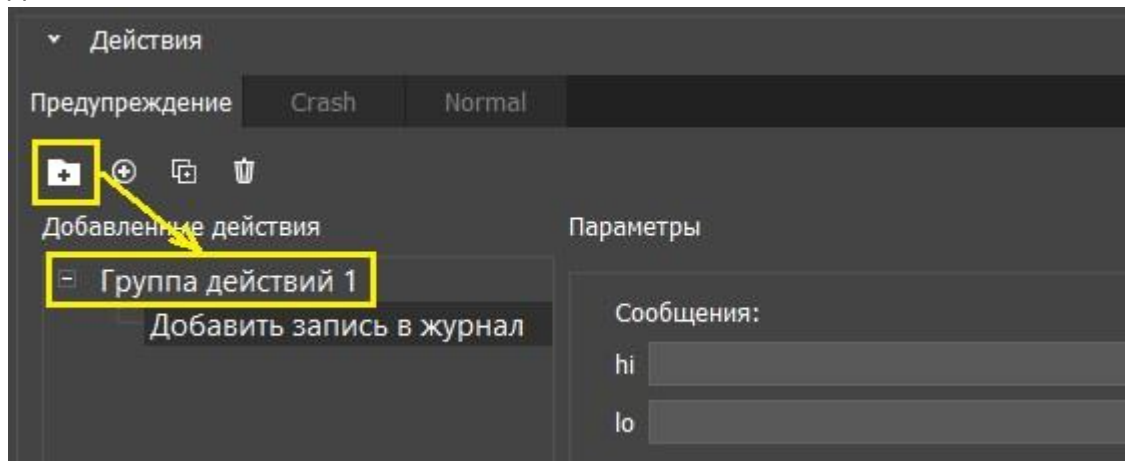
Необходима для установления задержки между другими действиями аларма.



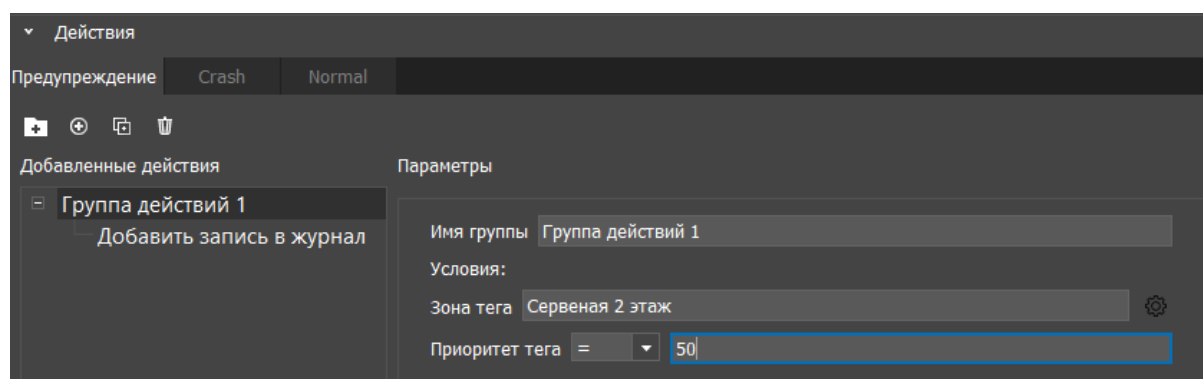
Для добавления действий алармов нужно создать и настроить группу действий.

6.4.2. Создание группы действий алармов

Для создания группы действий нужно нажать на иконку “Добавить группу действий”.



6.4.3. Настройка группы действий алармов



- **Имя группы** - имя группы действий алармов;
- **Условия** - условия срабатывания действий в группе:
 - **Зона тега** - зона, к которой относится тег. Зону можно выбрать, нажав на “Шестеренку” и выбрав нужные зоны, либо создав новые.

Если хотя бы одна из зон тега совпадает с зоной из списка - условие считается выполненным.

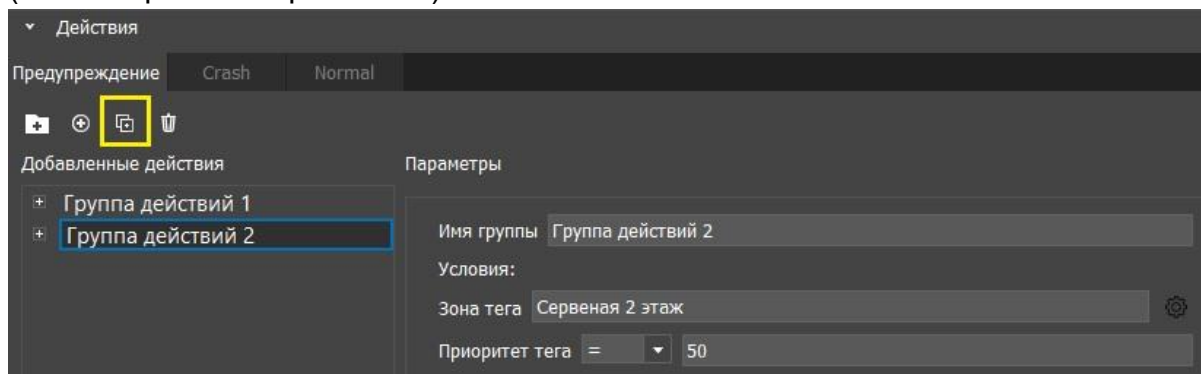
Если зоны не выбраны - условие будет соблюдаться независимо от зон тега.

- **Приоритет тега:**
 - = - приоритет тега равен приоритету группы действий;
 - < - приоритет тега меньше приоритета группы действий;
 - > - приоритет тега больше приоритета группы действий;
 - <= - приоритет тега меньше либо равен приоритету группы действий;
 - >= - приоритет тега больше либо равен приоритету группы действий;
 - != - приоритет тега не равен приоритету группы действий;
 - Любой - любой приоритет тега (учитывается только зона тега).

6.4.4. Клонирование группы действий алармов

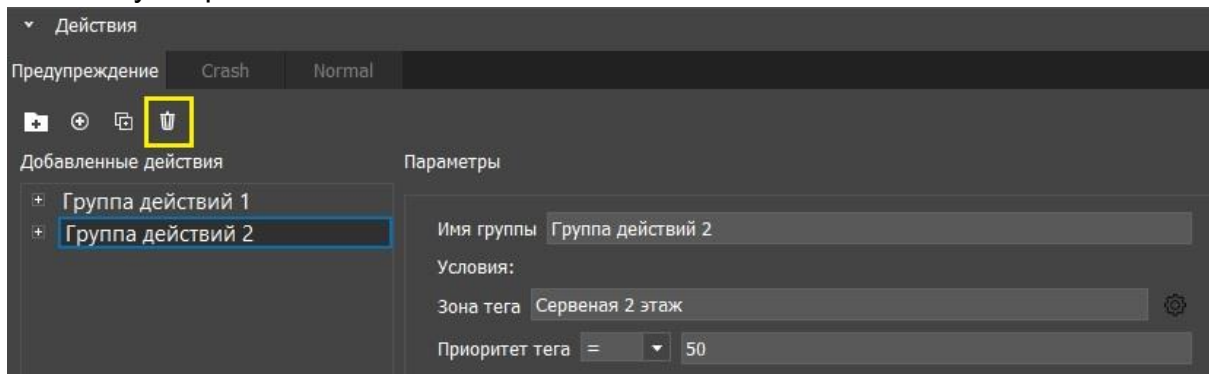
Для клонирования группы действий алармов нужно выделить группу действий и нажать на иконку “Клонировать”.

Создаётся группа действий алармов, аналогичная той, которую клонировали (все настройки сохраняются).



6.4.5. Удаление группы действий алармов

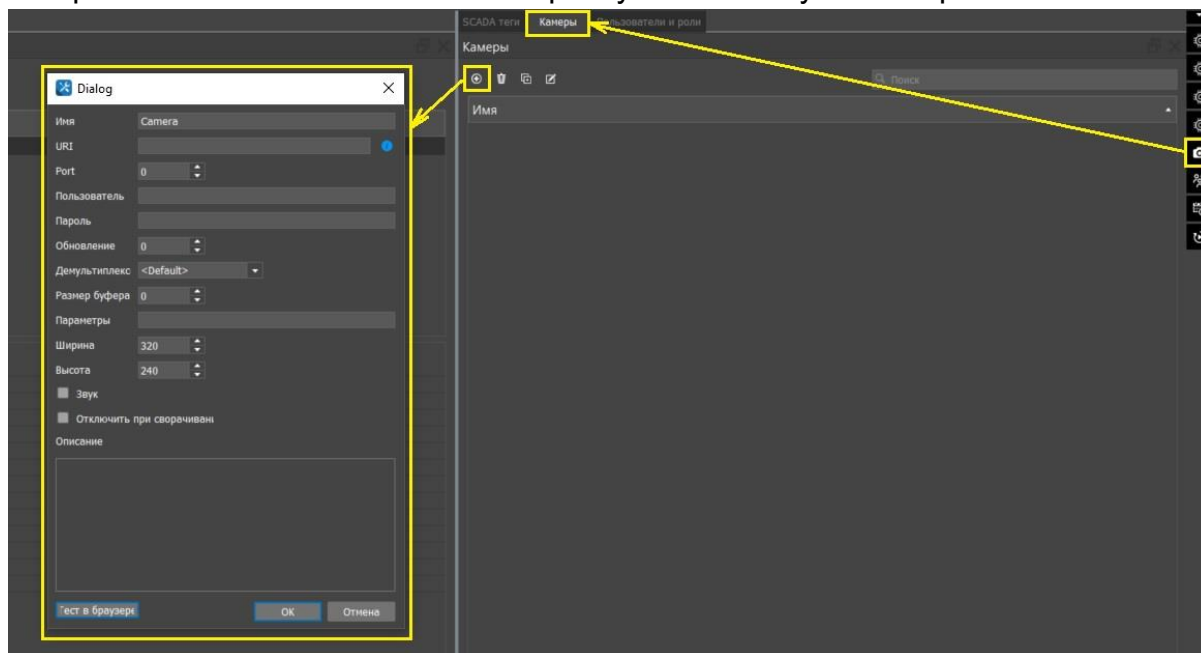
Для удаления группы действий алармов нужно выделить нужную группу и нажать на иконку “Корзина”.



6.4.6. Как сделать снимок с камеры и отправить его в Telegram

6.4.6.1. Добавление камеры

Откройте панель Камеры - хранилище ссылок на видео-поток с IP-камер, которые можно использовать и которые уже используются в проекте.



Для того чтобы открыть вкладку Камеры, если она закрыта, необходимо выбрать на панели справа иконку с камерой.

Для того чтобы добавить новую камеру, необходимо нажать на иконку с "+" на панели инструментов раздела "Камеры".

6.4.6.2. Настройка камеры

Динамические изображения (Dynamic Images) – потоковое видео или снимки, загружаемые из ресурса в Интернете или локальной сети. В качестве ресурса может выступать DVR, IP-камера, видео-регистратор, сервер.

Поддерживаются видео-кодеки:

- **JPEG / MJPEG** - потоковое видео и по кадровое получение данных от IP-камеры или регистратора кадры (снимки) - изображения *.jpg или *.png;
- **H.264 / MPEG-4** - видео высокой степени сжатия.

Для работы нужна прямая ссылка на изображение или видео-поток (без элементов интерфейса, компонентов ActiveX и т.п.).

Пример ссылки на динамическое изображение:

`http://root:root@192.168.0.100:80/source/video.mjpg`

`rtsp://root:root@mydomain.com:554/mpeg4/media.amp`

▶ протокол передачи данных (HTTP или RTSP)

▶ логин:пароль доступа к камере

▶ адрес ресурса - источника видео

▶ порт подключения к источнику

▶ ссылка на поток данных

Настроим ссылку в проекте:

The image shows a 'Dialog' window with the following fields and controls:

- Имя**: Text field containing 'Camera'.
- URI**: Text field with a blue information icon on the right.
- Port**: Spin box set to '0'.
- Пользователь**: Text field.
- Пароль**: Text field.
- Обновление**: Spin box set to '0'.
- Демультимплекс**: Dropdown menu set to '<Default>'.
- Размер буфера**: Spin box set to '0'.
- Параметры**: Text field.
- Ширина**: Spin box set to '320'.
- Высота**: Spin box set to '240'.
- Описание**: Large empty text area.
- Buttons**: 'Тест в браузере', 'ОК', and 'Отмена' at the bottom.

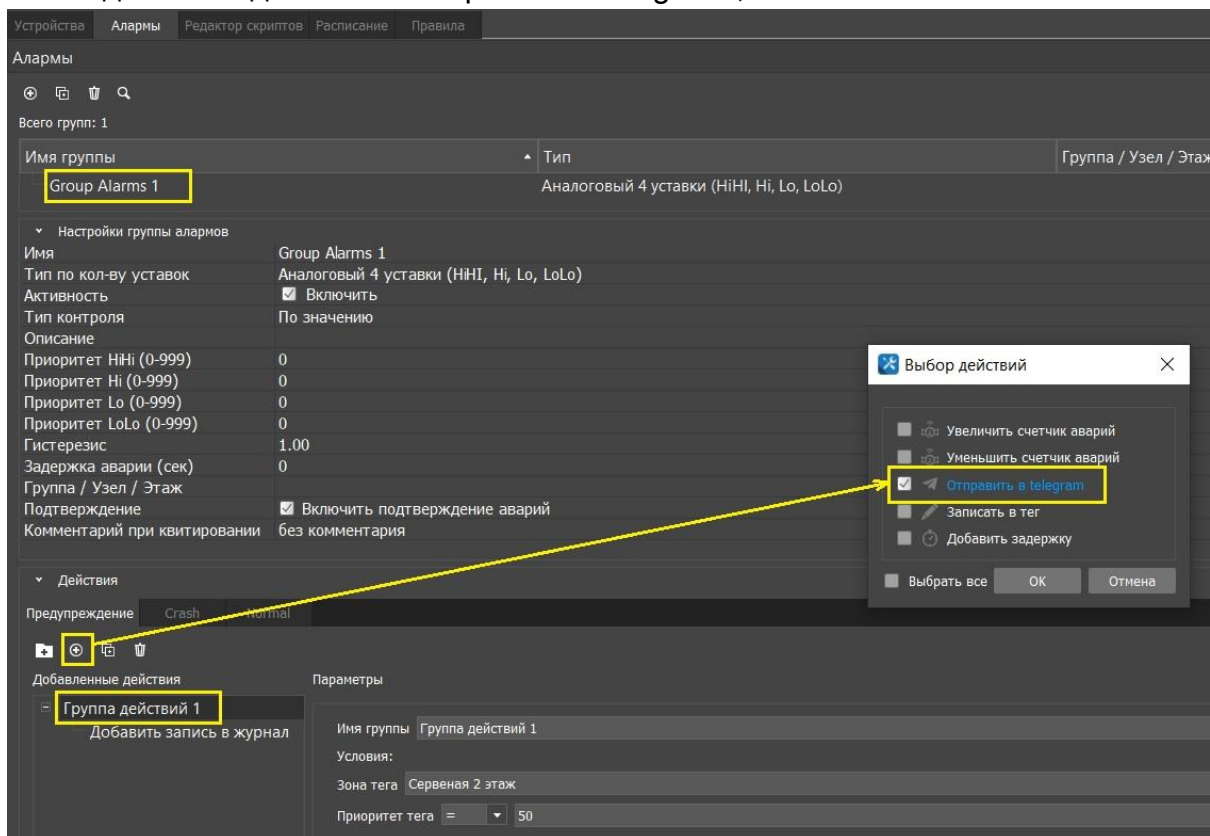
- **URI** - ссылка на видео-поток (или поток кадров). Логин, пароль и порт доступа к камере можно указать в строках ниже;

- **Port**- для JPEG/MJPEG, по умолчанию, порт: 80, для MPEG-4/H.264 порт: 554;
- **Пользователь/ Пароль** - логин и пароль доступа к видео-потoku с камеры;
- **Обновление** (мс) - переподключение к источнику видео;
 - Обновление: 0 (мс) - для потокового видео MJPEG/H.264/MPEG-4;
 - Обновление: 1000...n (мс) - для по-кадровой трансляции JPEG;
- **Демультимплекс** - автоматический или ручной выбор типа потока FFMPEG (для случаев, когда поток не определяется автоматически). Иногда используется при работе с Mobotix;
- **Размер буфера**: 50000 (Кб) - размер буфера данных в Кб, который приложение загрузит перед началом воспроизведения видео. По умолчанию, если оставить 0, составляет 5 Мб. Можно уменьшить для камер с малым разрешением или для медленного соединения, но, не рекомендуются значения менее 20000 Кб;
- **Параметры** - строка для указания типа транспорта RTSP. Укажите:
 - `rtsp_transport=tcp`; если RTSP поток с IP камеры загружается через Интернет или его качество неудовлетворительное;
 - `rtsp_transport=udp`; или пустое поле подключает загрузку RTSP потока в локальной сети по UDP;
- **Ширина / Высота** - ширина / высота кадра для масштабирования. Если оставить пустые поля, то видео отобразится с оригинальным разрешением;
- **Описание** - описание камеры.

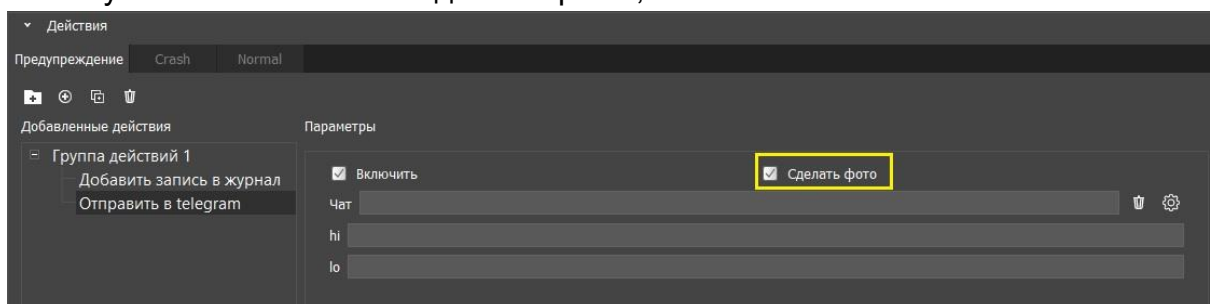
6.4.6.3. Настройка действия

Для настройки отправки фото в Telegram:

1. выбрать группу аварийных сообщений (алармов);
2. выбрать группу действий;
3. добавить действие “Отправить в telegram”;



4. установить чекбокс “Сделать фото”;



5. в свойствах тега, в разделе “Аларм”, установить группу аварийных сообщений, включить чекбокс “Активность” и в свойстве “Видеоконтроль” добавить камеру из списка.

Теги 1 / 1000000 Папки 1 / 65535

SCADA tags

Tag

Свойства

Аларм

Группа	Group Alarms 1
Активность	☐ Выключить
HiHi	0
Hi	0
Lo	0
LoLo	0
Приоритет HiHi (0-999)	0
Приоритет Hi (0-999)	0
Приоритет Lo (0-999)	0
Приоритет LoLo (0-999)	0
Видеоконтроль	

Свойства

Имя	Tag
Активность	☒ True
Описание	
Считать из	Drivers.Modbus TCP Network.Tags.Custom:Value
Записать в	Drivers.Modbus TCP Network.Channels.Custom:Value
Тип данных	Signed int
Начальное значение	0
Шкала	
Мин	0
Макс	100
Единицы измерения	Units
Запись в БД	
Стратегия записи	Нет
Другое	
Зоны	
Доступ	Администратор, Инженер, Оператор

Аларм

Группа	Group Alarms 1
Активность	☒ Включить
HiHi	0
Hi	0
Lo	0
LoLo	0
Приоритет HiHi (0-999)	0
Приоритет Hi (0-999)	0
Приоритет Lo (0-999)	0
Приоритет LoLo (0-999)	0
Видеоконтроль	

Группы алармов

Поиск

Имя

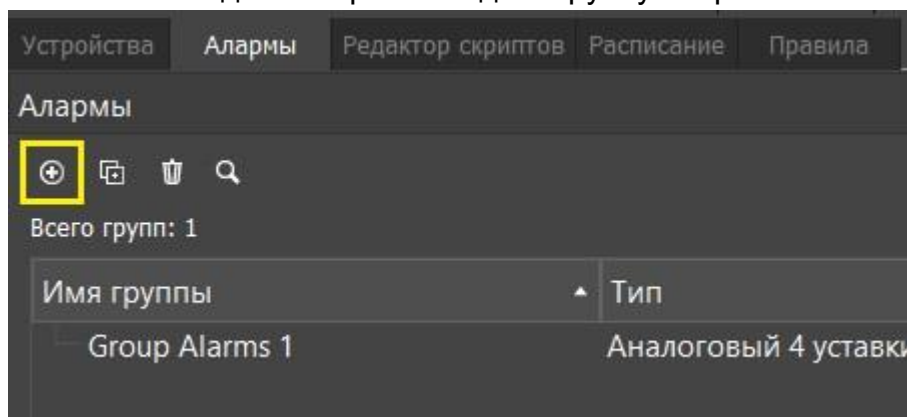
Camera

OK Отмена

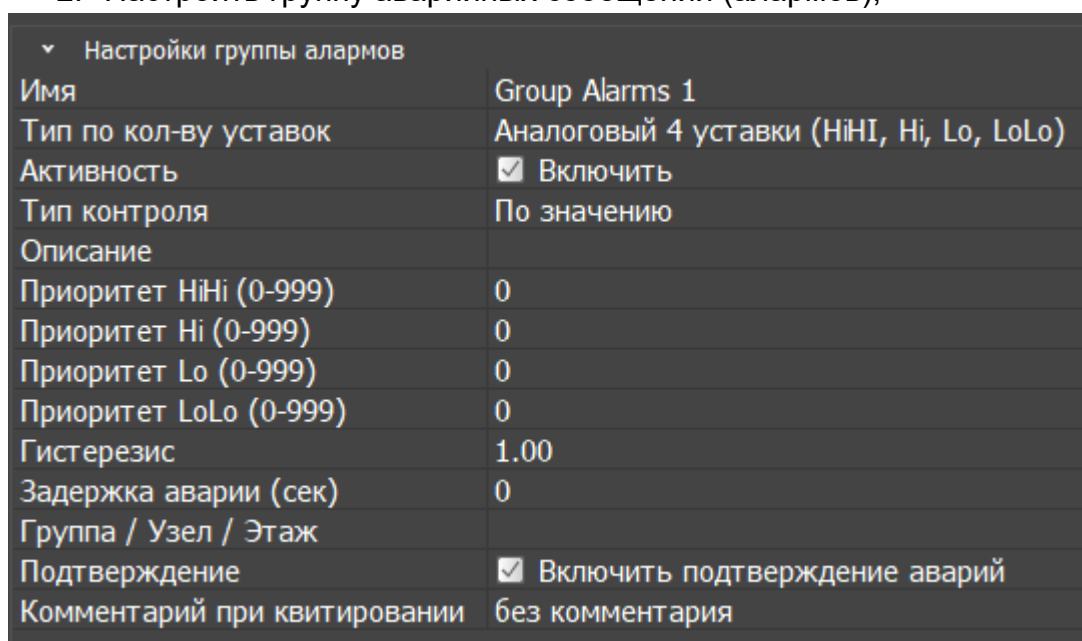
...

6.5. Настройка системы аварийных сообщений (алармов)

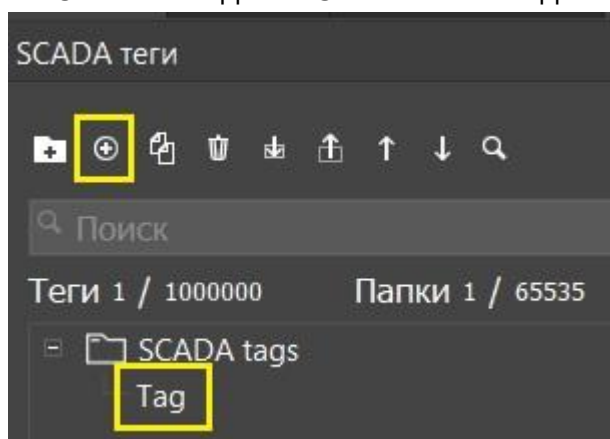
1. На вкладке “Алармы” создать группу аварийных сообщений (алармов);



2. Настроить группу аварийных сообщений (алармов);



3. На вкладке “SCADA теги” создать тег;



4. Настроить свойства тега;

Теги 1 / 1000000 Папки 0 / 65535

SCADA tags

Tag

Свойства

Имя	Tag
Активность	☒ True
Описание	
Считать из	
Записать в	
Тип данных	Signed int
Начальное значение	0
Запись в БД	
Стратегия записи	Нет
Другое	
Зоны	
Доступ	Администратор, Инженер, Оператор

Аларм

Группа	(None)
Активность	☐ Выключить
Видеоконтроль	

5. Привязать тег к группе аварийных сообщений (алармов).

Теги 1 / 1000000 Папки 1 / 65535

SCADA tags

Tag

Свойства

Имя	Tag
Активность	☒ True
Описание	
Считать из	
Записать в	
Тип данных	Signed int
Начальное значение	0
Шкала	
Мин	0
Макс	1000
Единицы измерения	мм
Запись в БД	
Стратегия записи	Зона чувствительности
Значение	10.00
Другое	
Зоны	Сервентная 2 этаж
Доступ	Администратор, Инженер, Оператор

Аларм

Группа	(None)
Активность	☐ Выключить
Видеоконтроль	

Группы алармов

Поиск

Имя

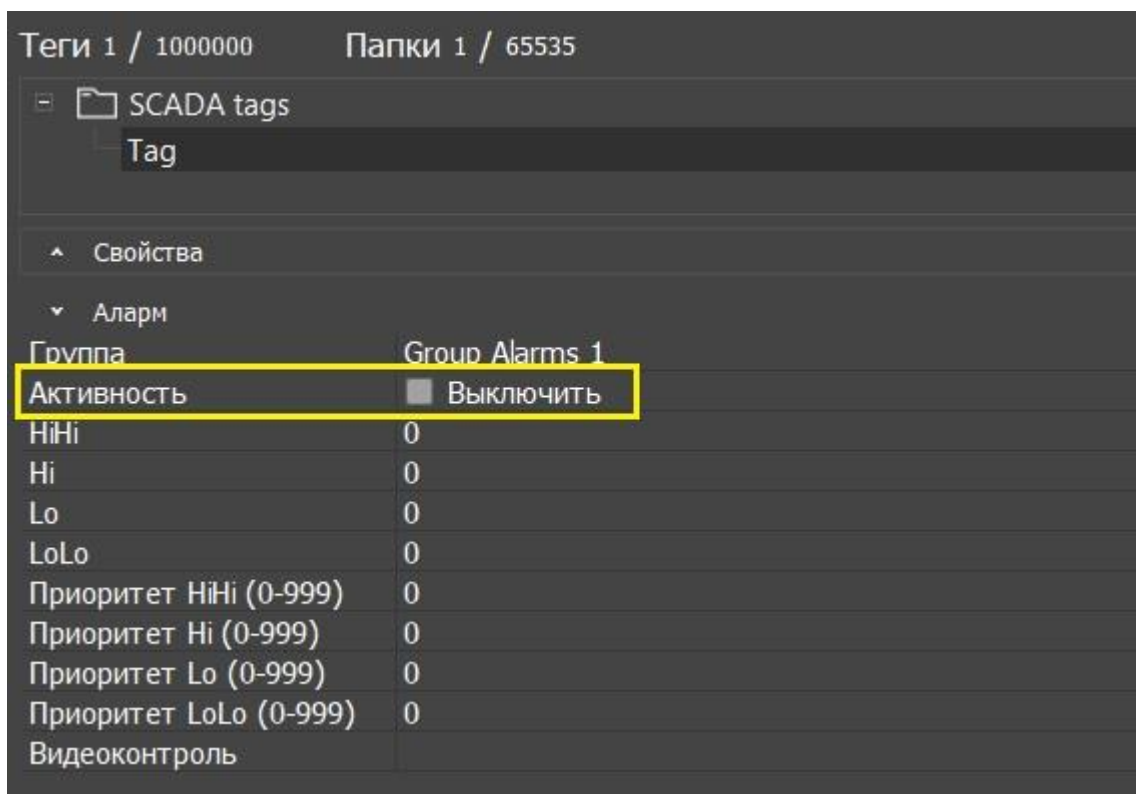
(None)

Group Alarms 1

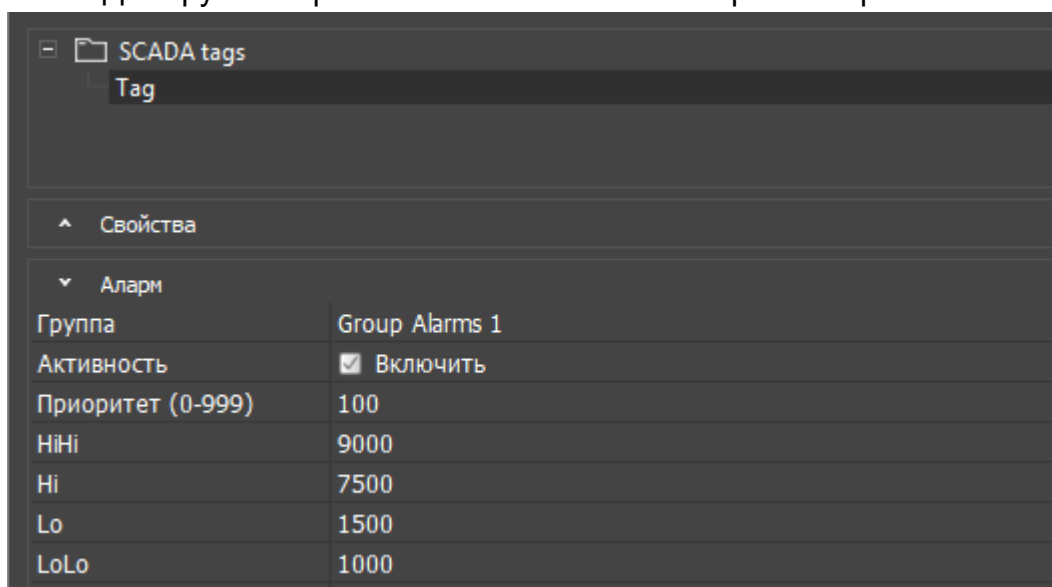
OK Отмена

...

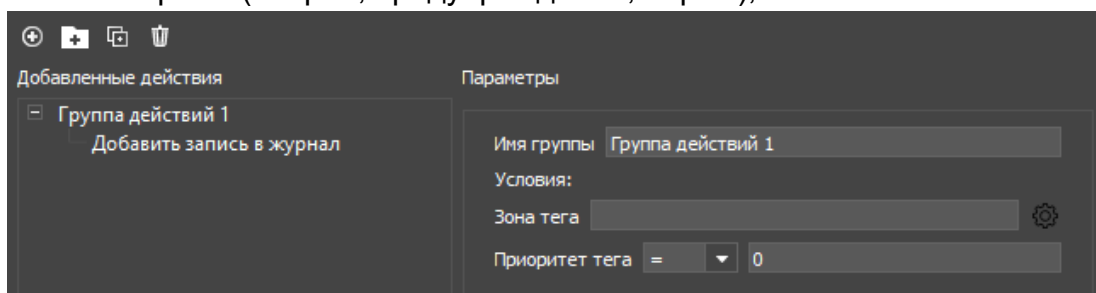
6. Включить активность аларма, при необходимости сменить приоритет;



7. Для групп алармов аналогового типа настроить пороговые значения;



8. На вкладке "Алармы" добавить группу действий для состояния группы алармов (авария, предупреждение, норма);



9. Настроить группу действий;

Параметры

Имя группы:

Условия:

Зона тега: 

Приоритет тега:

Зона должна соответствовать зоне тега.

Приоритет тега должен входить в диапазон приоритетов, выбранных для группы действий;

Приоритет тега:

- ☒ =
- ☐ <
- ☐ >
- ☐ <=
- ☐ >=
- ☐ !=
- ☐ Любой

10. Задать сообщения для записи в журнал аварий. Если оставить поля пустыми, то в журнал аварий будут записываться системные сообщения;

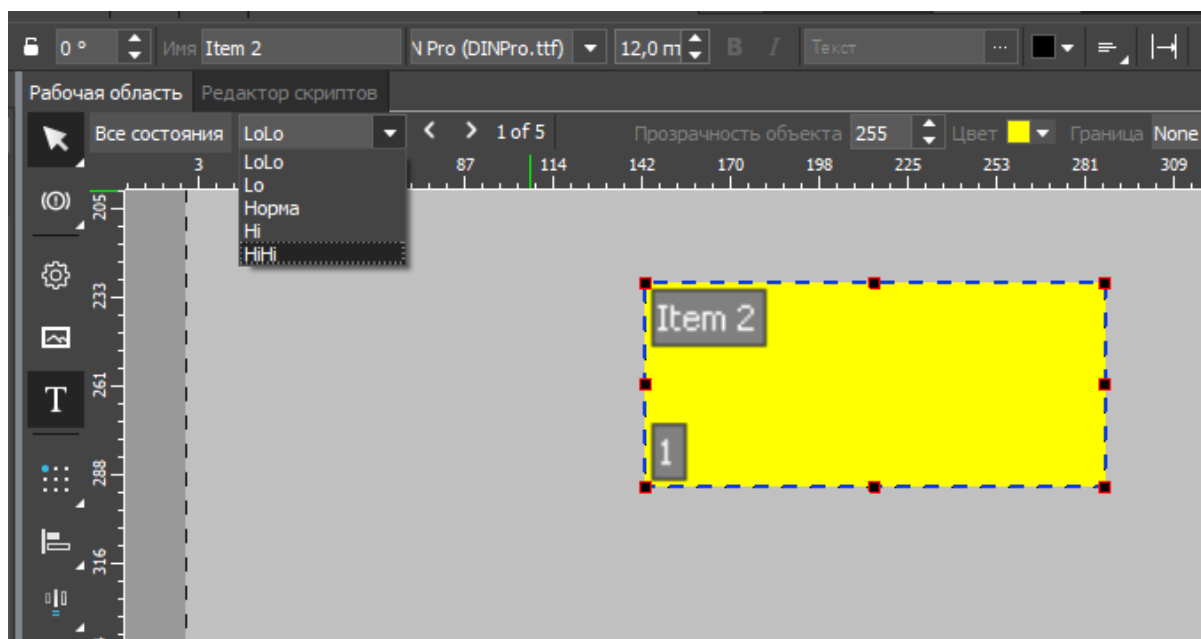
Добавленные действия	Параметры
Температура Добавить запись в журнал	Сообщения: hihi: Очень высокая температура lolo: Очень низкая температура

11. При необходимости добавить дополнительные действия групп алармов;

12. Повторить для оставшихся состояний группы алармов.

6.6. SCADA Alarm Label

6.6.1. Описание



SCADA Alarm Label - это графический элемент, позволяющий информировать пользователей об актуальном состоянии выбранного SCADA-тега. Настройка элемента производится в панельном проекте SCADA. Он имеет до 5 вариантов внешнего вида, для каждого состояния тревоги и состояния "Норма".

SCADA Alarm Label меняет свое состояние при получении значения текущей тревоги от SCADA тега.

6.6.2. Принцип работы

Когда срабатывает тревога SCADA Alarm Label начинает мигать переключаясь с состояния Норма на состояние, соответствующие уровню тревоги. Количество состояний элемента определяется типом группы тревоги.

При привязке тега у SCADA Alarm Label обновляется количество состояний элемента в зависимости от типа группы аварий.

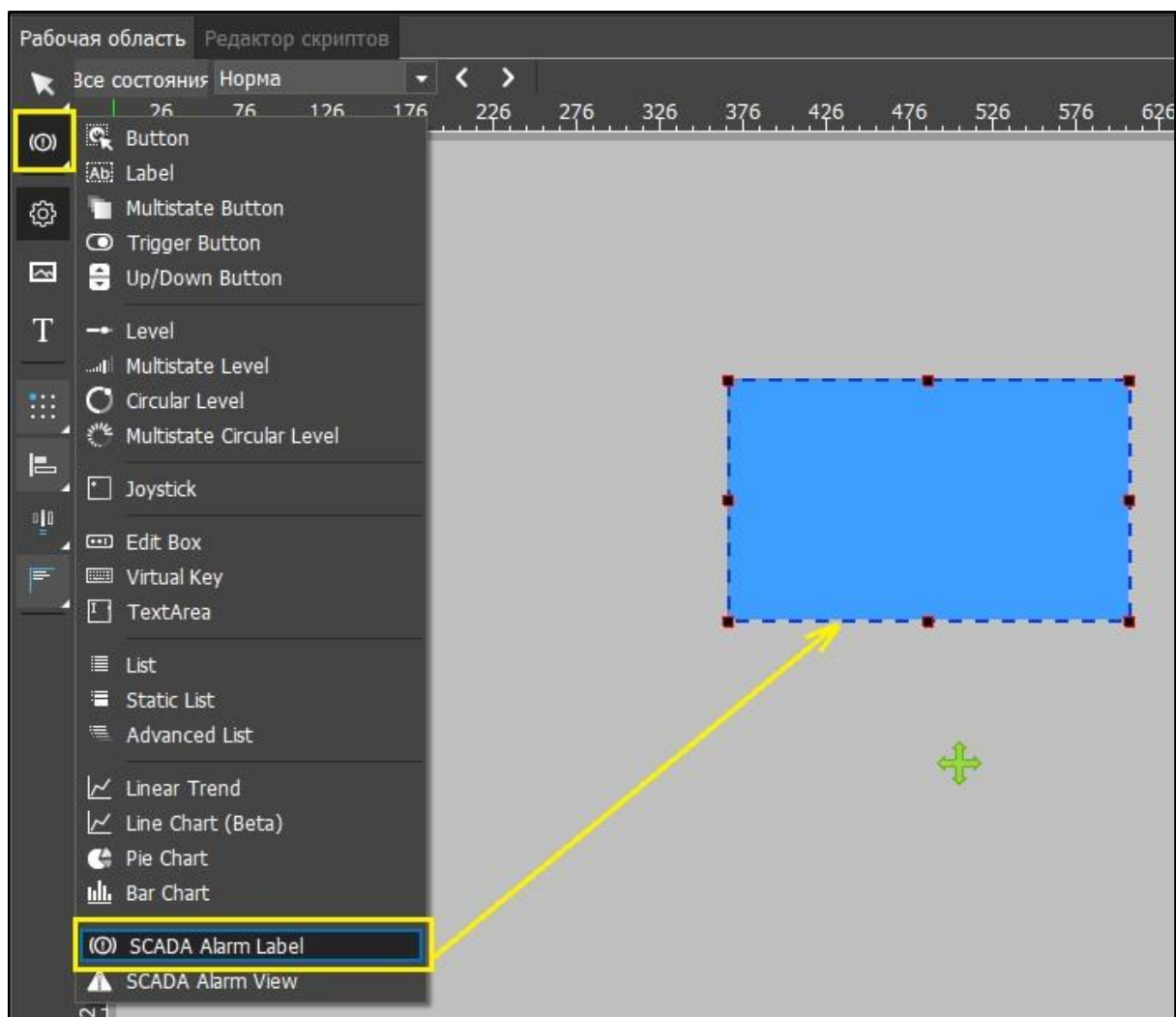
Если у группы аварий стоит уставка “Аналоговый 4 уставки (HiHi, Hi, Lo, LoLo)”, то у Alarm Label будет пять состояний: HiHi, Hi, Норма, Lo, LoLo.

Если у группы аварий стоит уставка “Аналоговый 2 уставки (Hi, Lo)”, то у Alarm Label будет три состояния: Hi, Норма, Lo.

Если у группы аварий будет уставка “Дискретный прямой”, то у Alarm Label будет два состояния: Норма, Авария.

Если у SCADA Alarm Label нет привязанного тега, то у элемента есть состояние Норма.

6.6.3. Создание SCADA Alarm Label



Выберите элемент SCADA Alarm Label из списка графических элементов, перенесите его на страницу или попап и растяните до нужных размеров.

6.6.4. Параметры элемента

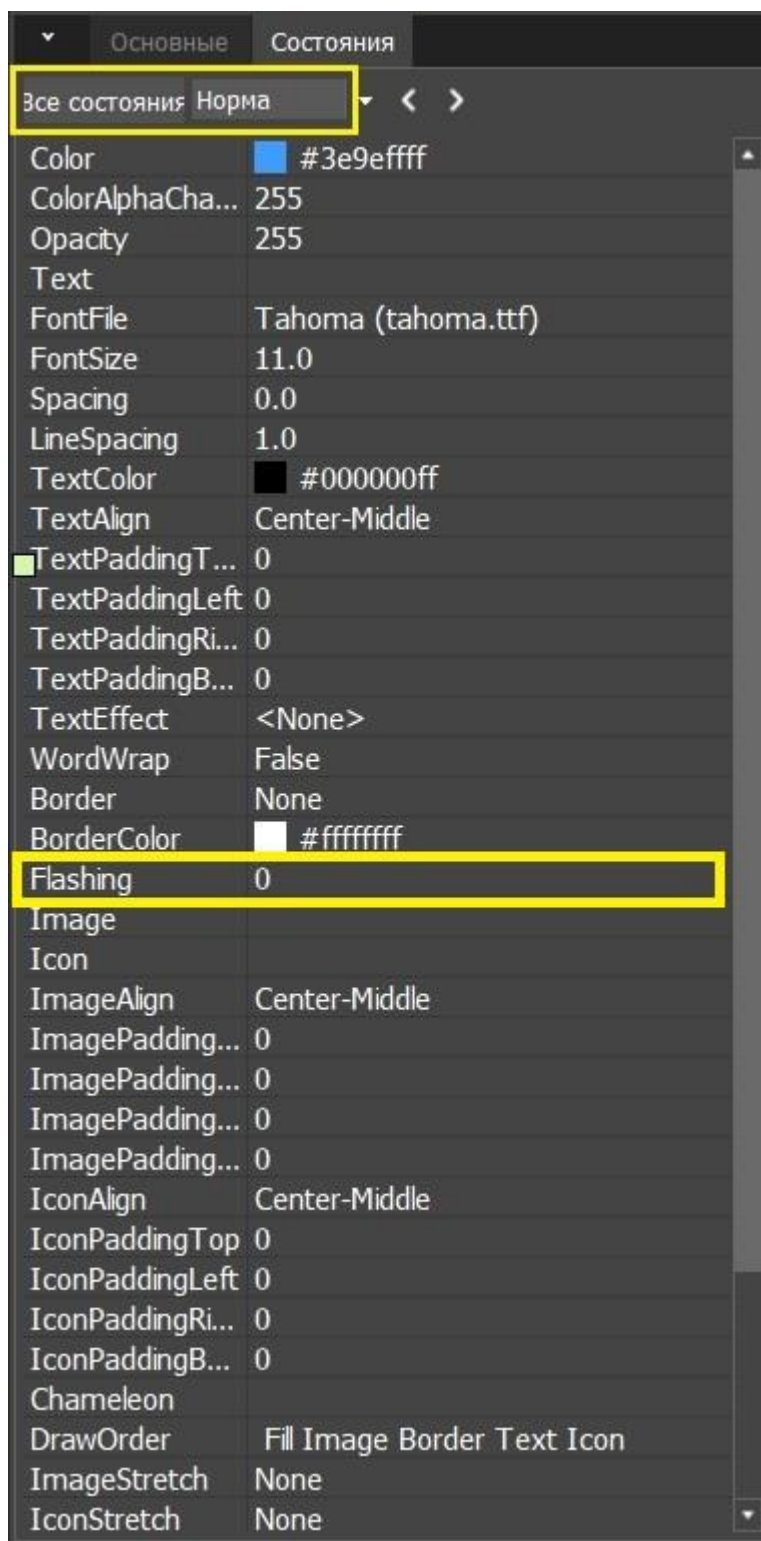
Основные параметры элемента:

▼	Основные	Состояния
Name	Item 1	
X	368	
Y	144	
Width	241	
Height	145	
Visible	True	
Type	SCADA Alarm Label	
Tags		
Доступ	Администратор, Инженер, Оператор	

- **Name** - имя элемента;
- **X, Y, Width, Height** - положение и габаритные размеры;
- **Visible** - отображение элемента по умолчанию включено. Если выключить отображение, при запуске i3 pro элемент будет полностью скрыт. Отображением элемента можно управлять в процессе работы i3 pro через свойство Visible;
- **Type** - SCADA Alarm Label;
- **Tags** - название, предназначенное для поиска графического элемента или группы элементов в проекте через поисковую строку;
- **Доступ** - выбор пользователей, которым доступен данный элемент. Для пользователей, которые не выбраны, элемент будет не доступен для отображения и использования.

Параметры состояния соответствуют параметрам остальных графических элементов и настраиваются для каждого из состояний отдельно.

Дополнительно в свойствах Alarm Label есть свойство для настройки частота мерцания - Flashing. Частота мерцания настраивается в герцах.

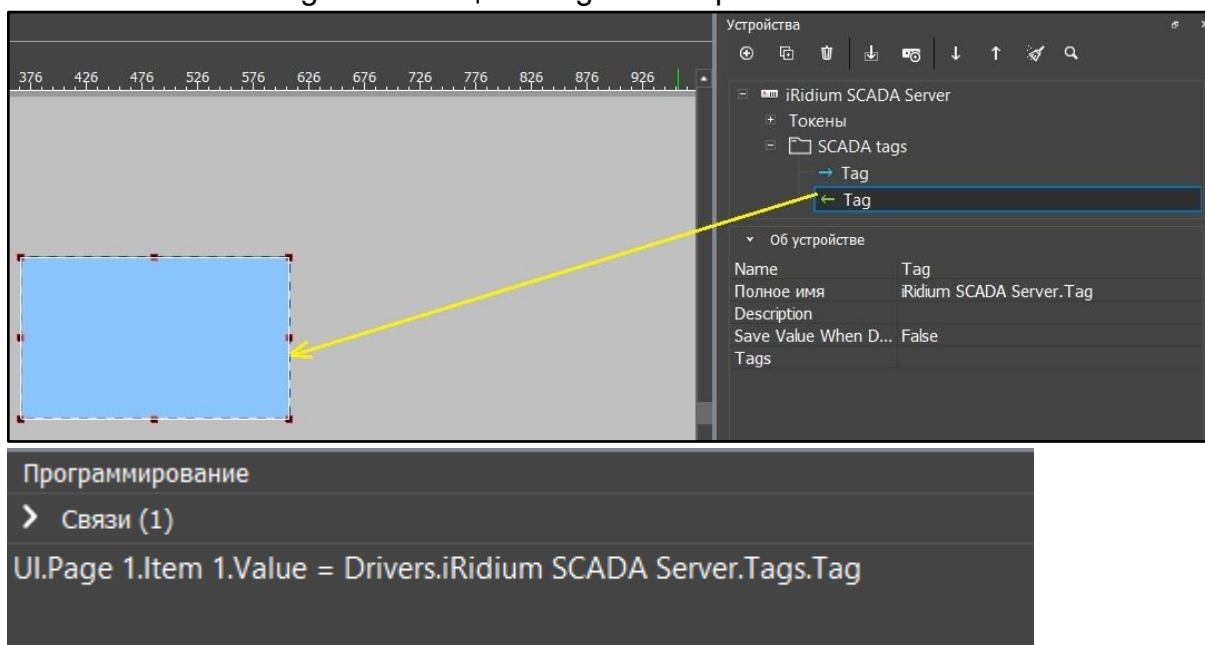


6.6.5. Привязка к оборудованию

У SCADA Alarm Label нет доступных действий в программировании, элемент взаимодействует только с тегами iRidi SCADA Server.

Используйте тег, чтобы передавать элементу текущие состояние тревоги.

Для привязки тега перетащите его из раздела “Устройства”→”iRidi SCADA Server”→”SCADA tags” с помощью drag-and-drop на элемент Alarm Label.



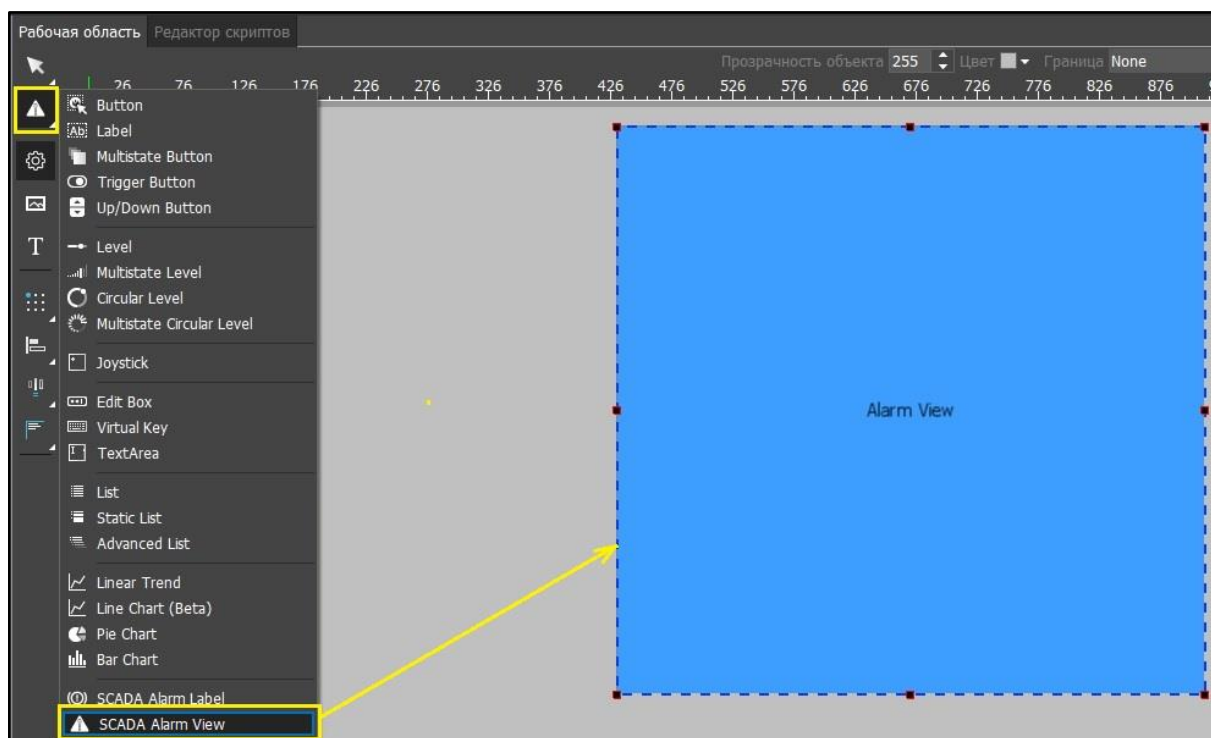
6.7. SCADA Alarm View

6.7.1. Описание

SCADA Alarm View - это графический элемент, предназначенный для отображения аварийных сообщений (алармов) в виде таблицы.

Журнал аварий								
	Сообщение	Тип	П	ГА	Зона	Группа/Узел/Этаж	D&T	Квитирование
●	R007-CO2 = 585.000000 ppm (Normal)	Normal	500	CO2	R007	CO2	15:24:37 05/04/2024	
⚠	KITCHEN2-CO2 = 611.000000 ppm (Hi)	Warning	51	CO2	2 floor	CO2	15:22:20 05/04/2024	
▲	R214-Temperature-1 = 26.875340 (Hi)	Accident	51	Temperature	R214	Temperature	15:21:11 05/04/2024	
⚠	R007-CO2 = 604.000000 ppm (Hi)	Warning	51	CO2	R007	CO2	15:08:56 05/04/2024	
▲	R002-Temperature-1 = 26.005341 (Hi)	Accident	51	Temperature		Temperature	14:48:27 05/04/2024	
▲	R201-CO2 = 1011.000000 ppm (HiHi)	Accident	50	CO2	R201	CO2	14:30:45 05/04/2024	
⚠	R201-CO2 = 984.000000 ppm (Hi)	Warning	51	CO2	R201	CO2	14:27:05 05/04/2024	
▲	R201-CO2 = 1001.000000 ppm (HiHi)	Accident	50	CO2	R201	CO2	14:25:35 05/04/2024	
▲	R210-Temperature = 26.035339 (Hi)	Accident	51	Temperature	R210	Temperature	13:47:29 05/04/2024	
⚠	R214-CO2-1 = 608.000000 ppm (Hi)	Warning	51	CO2	R214	CO2	13:38:17 05/04/2024	
⚠	R214-CO2-2 = 606.000000 ppm (Hi)	Warning	51	CO2	R214	CO2	13:37:19 05/04/2024	
▲	R205-Temperature = 26.005341 (Hi)	Accident	51	Temperature	R205	Temperature	13:09:42 05/04/2024	
⚠	R215-CO2 = 979.000000 ppm (Hi)	Warning	51	CO2	R215	CO2	13:08:52 05/04/2024	
▲	R215-CO2 = 1091.000000 ppm (HiHi)	Accident	50	CO2	R215	CO2	13:10:03 05/04/2024	
▲	KITCHEN2-Temperature = 28.555340 (Hi)	Accident	51	Temperature	2 floor	Temperature	12:14:08 05/04/2024	
▲	R213-Temperature-2 = 28.155342 (Hi)	Accident	51	Temperature	R213	Temperature	12:14:04 05/04/2024	
⚠	R213-CO2-2 = 755.000000 ppm (Hi)	Warning	51	CO2	R213	CO2	12:14:04 05/04/2024	
⚠	R213-CO2-1 = 901.000000 ppm (Hi)	Warning	51	CO2	R213	CO2	12:14:04 05/04/2024	
⚠	R212-CO2 = 723.000000 ppm (Hi)	Warning	51	CO2	R212	CO2	12:14:05 05/04/2024	
⚠	R211-CO2 = 692.000000 ppm (Hi)	Warning	51	CO2	R211	CO2	12:14:05 05/04/2024	

6.7.2. Создание SCADA Alarm View



В SCADA-клиенте выберите элемент SCADA Alarm View из списка графических элементов, перенесите его на страницу или попап и растяните до нужных размеров.

6.7.3. Параметры элемента

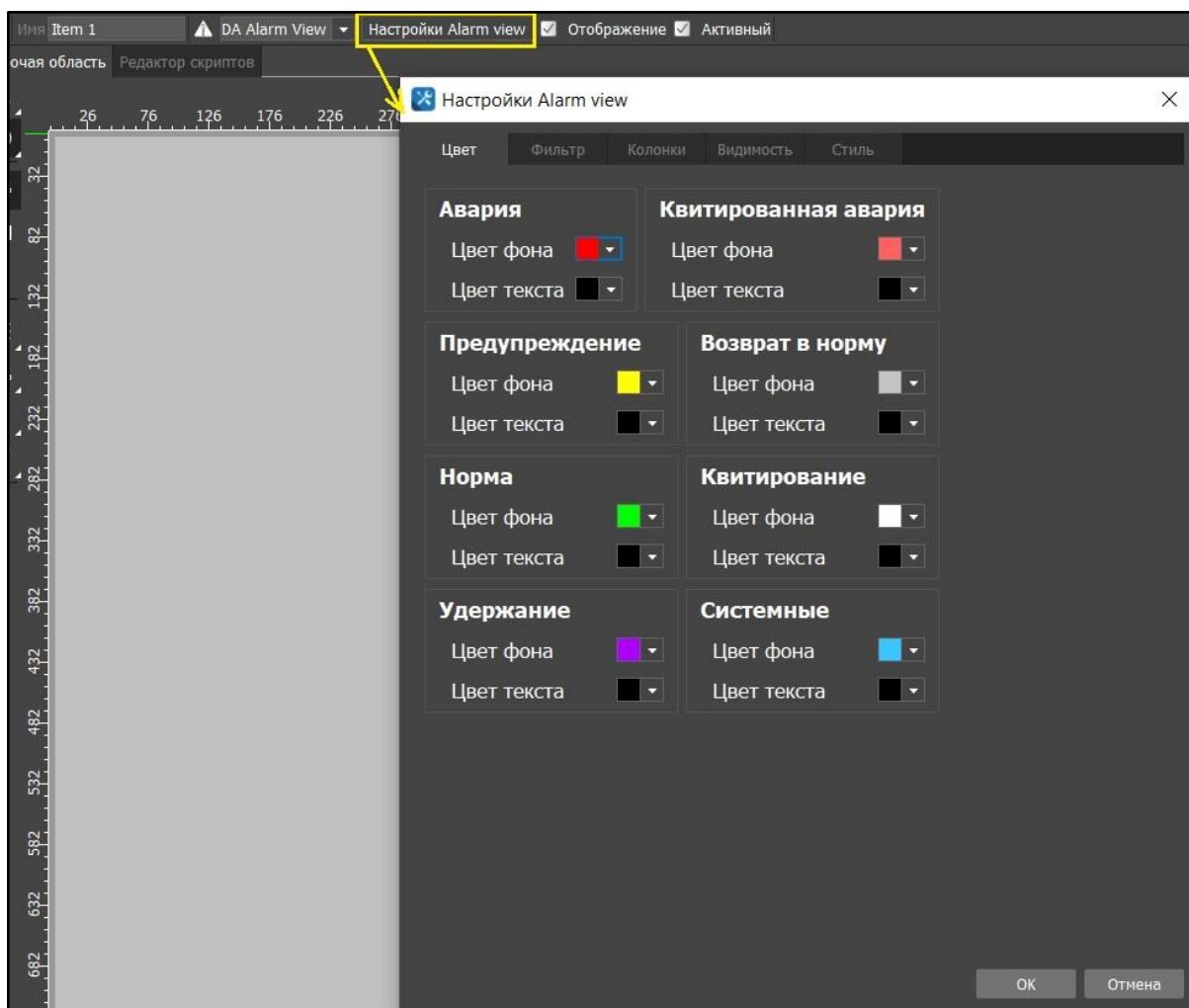
Основные параметры элемента:

▼	Основные	Состояния
Name	Item 1	
X	432	
Y	16	
Width	481	
Height	465	
Enable	True	
Visible	True	
Type	SCADA Alarm View	
Tags		
Доступ	Администратор, Инженер, Оператор	

- **Name** - имя элемента;
- **X, Y, Width, Height** - положение и габаритные размеры;
- **Visible** - видимость элемента при запуске проекта на управляющей панели. Отображение элемента по умолчанию включено. Если выключить отображение, при запуске i3 pro элемент будет полностью скрыт. Отображением элемента можно управлять в процессе работы i3Pro через свойство Visible.
- **Enable** - доступность управления элементом при запуске проекта на управляющей панели. Доступность по умолчанию включена. Если выключить доступность, при запуске i3 pro элемент будет показан, но не активен. Доступностью элемента можно управлять в процессе работы i3Pro через свойство Enable.
- **Type** - SCADA Alarm View;
- **Tags** - название, предназначенное для поиска графического элемента или группы элементов в проекте через поисковую строку.
- **Доступ** - выбор пользователей, которым доступен данный элемент. Для пользователей, которые не выбраны, элемент будет не доступен для отображения и использования.

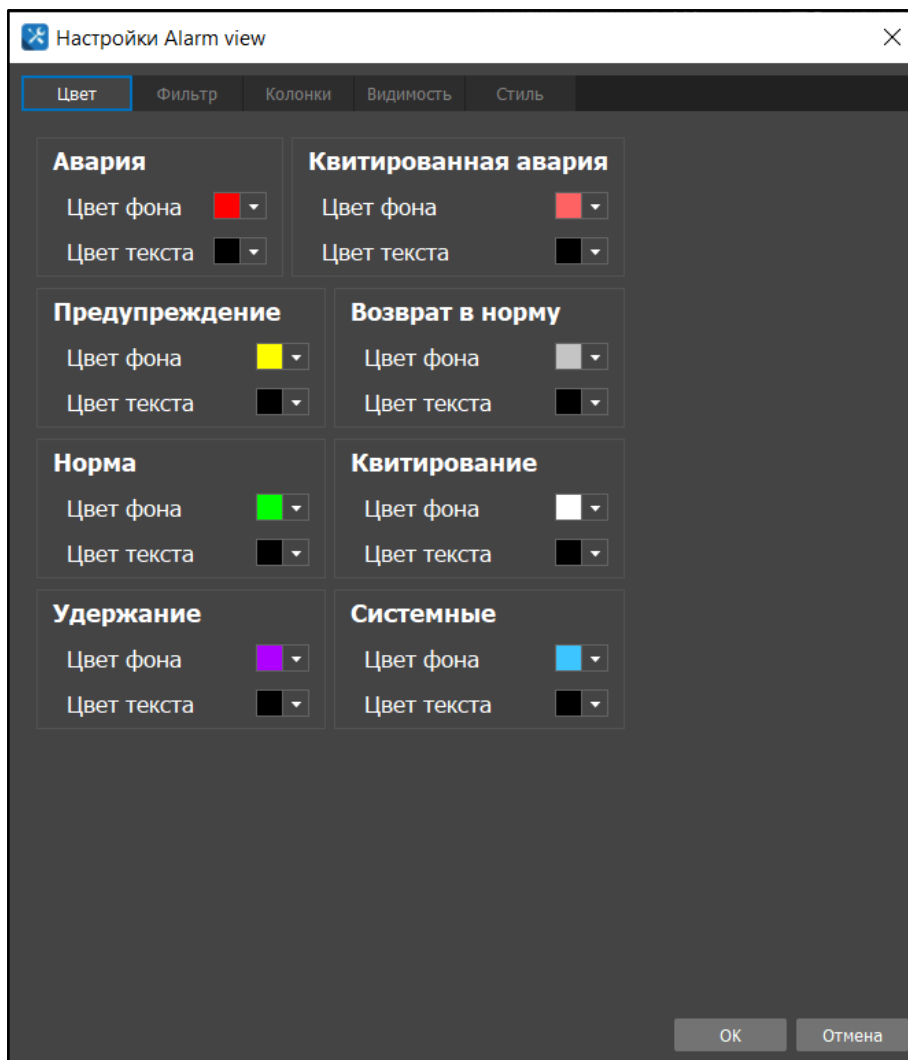
6.7.4. Настройка SCADA Alarm View

Внешний вид элемента и отображаемых сообщений настраивается в отдельном окне “Настройка Alarm View”. Окно можно вызвать двойным щелчком по элементу или нажатием кнопки “Настройка Alarm View” на панели свойств элемента.

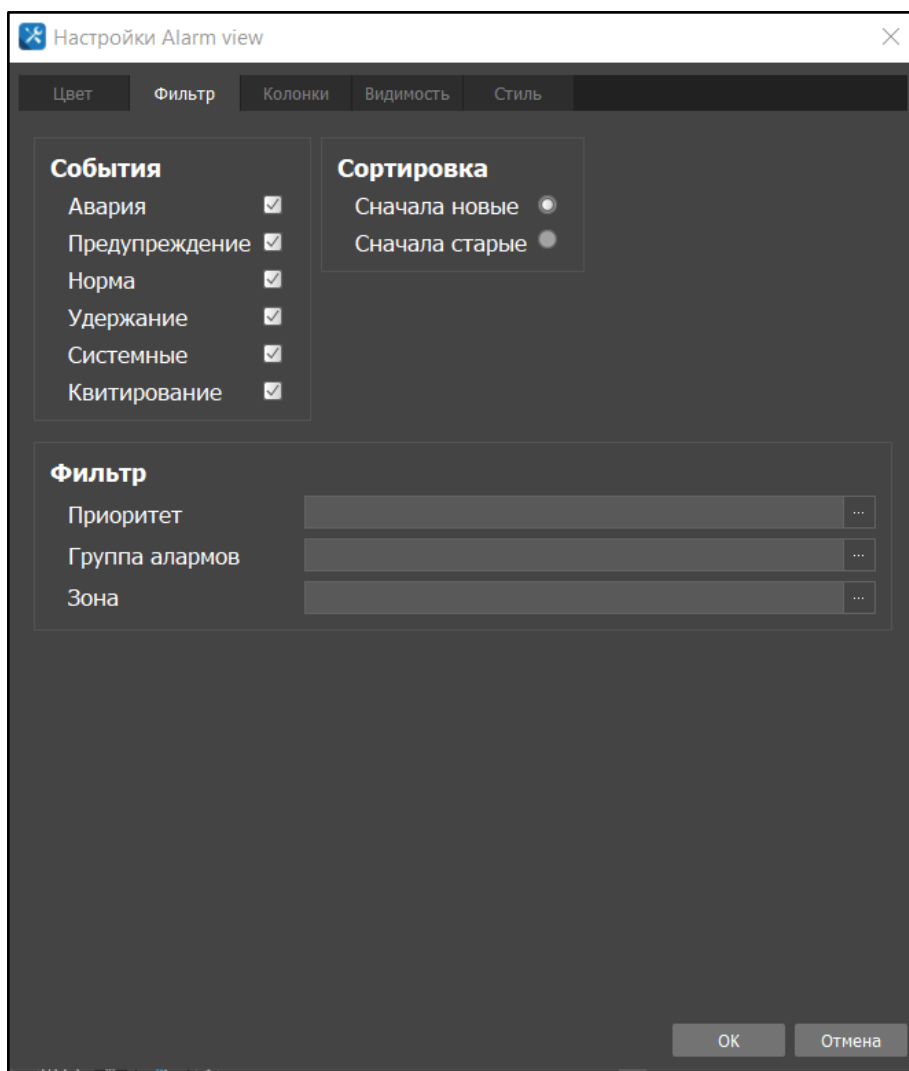


Настройка SCADA Alarm View:

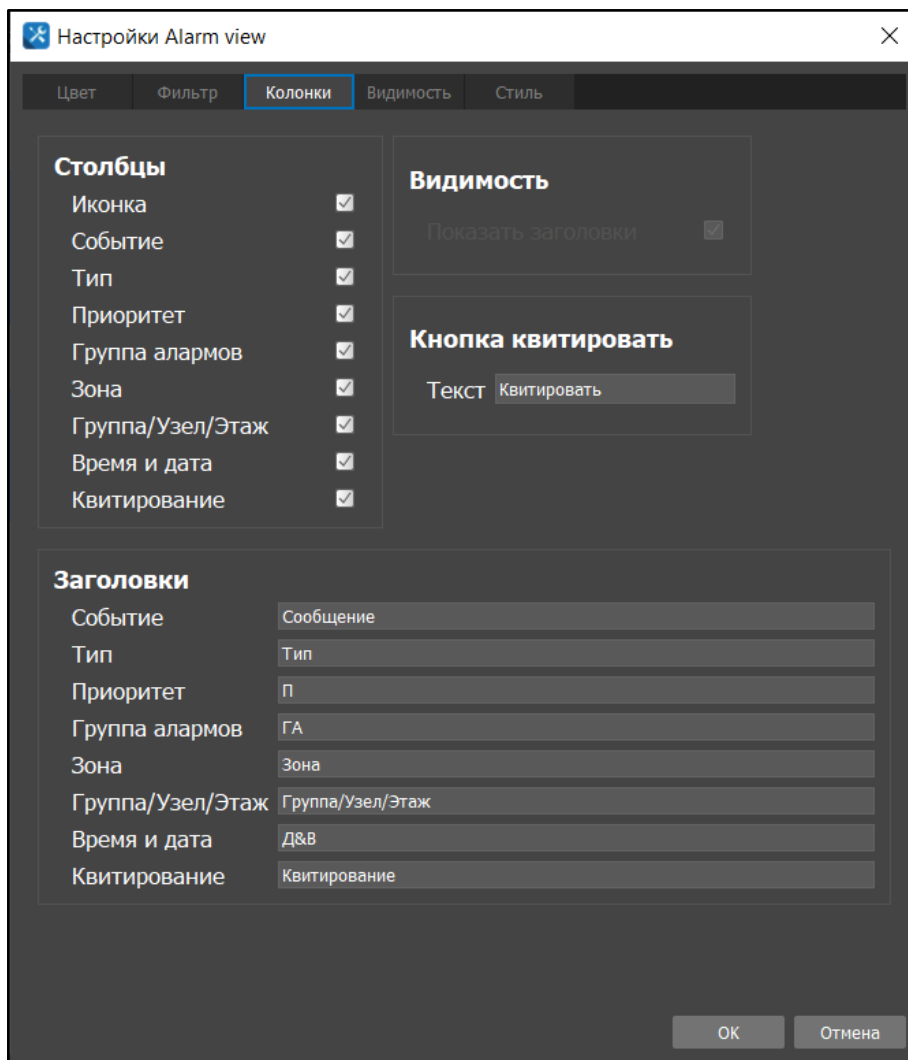
- **Цвет** - настройка цвета фона и текста разных групп и состояний аварийных сообщений/алармов (Авария, Предупреждение, Норма, Удержание, Квитирование аварий, Возврат в норму, Квитирование, Системные);



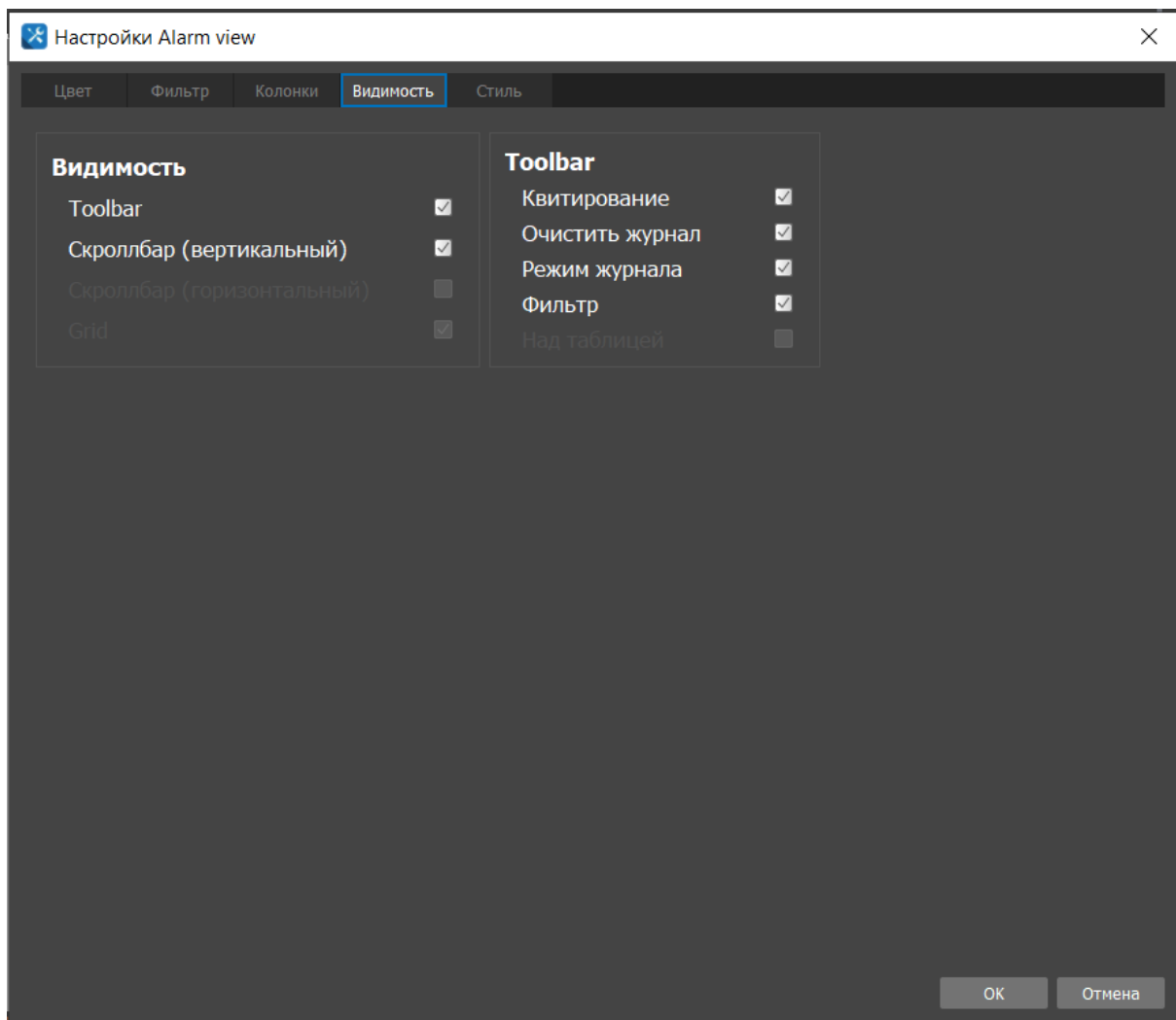
- **Фильтр** - настройка тех, сообщений, которые будут показаны в таблице:
 - **События** (Авария, Предупреждение, Норма, Удержание, Квитирование, Системные) - фильтрация по группам аварийных сообщений;
 - **Сортировка** (Сначала новые, Сначала старые) - показывать вновь пришедшие сообщения вначале или конце страницы;
 - **Фильтр** - фильтрация сообщений по приоритету, группе или зоне.



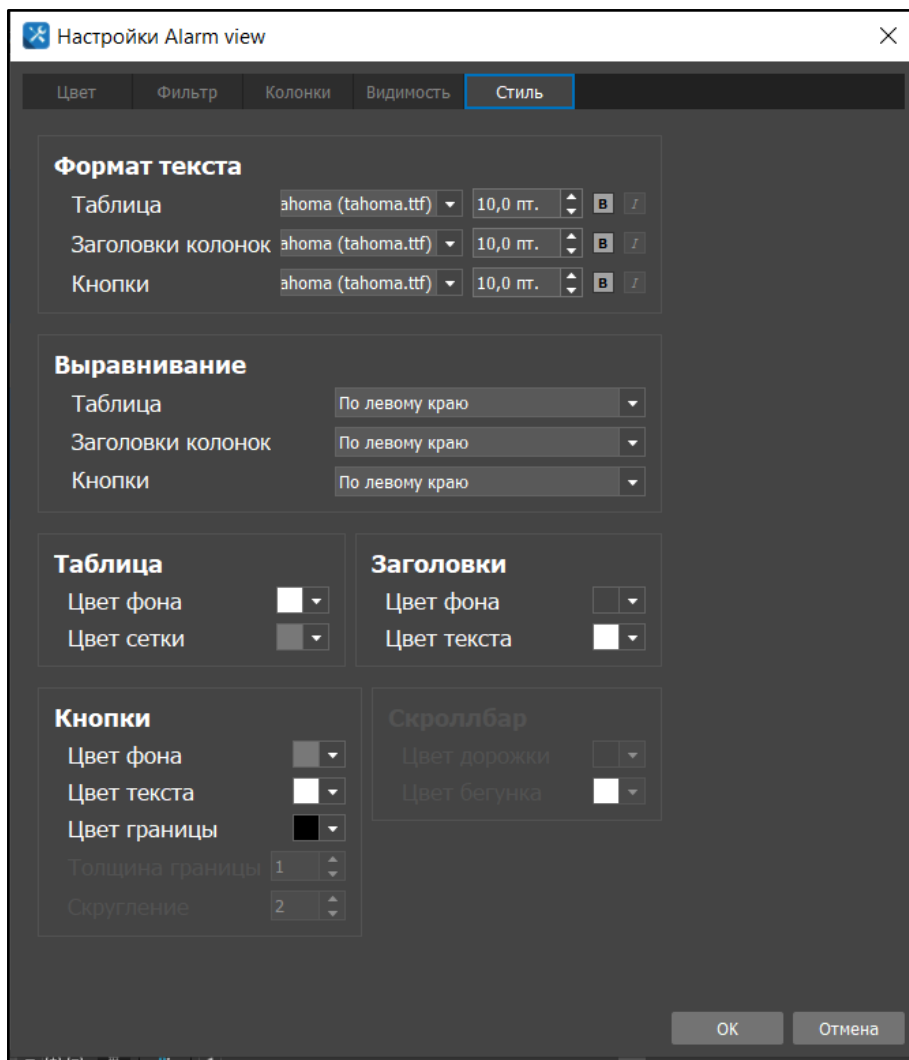
- **Колонки** - настройка той информации, которая будет отображаться в таблице:
 - **Столбцы** (Иконка, Событие, Тип, Приоритет, Группа алармов, Зона, Группа/Узел/Этаж, Время и дата, Квитирование) - настройка тех столбцов с информацией об аварийных сообщениях, которые будут отображены в таблице;
 - **Видимость** - показывать название столбцов;
 - **Кнопка квитировать** - настройка названия кнопки квитирования;
 - **Заголовки** - настройка названий столбцов.



- **Видимость** - настройка панели инструментов:
 - **Видимость:** Toolbar - показывать панель инструментов, Скроллбар (вертикальный) - показывать полосу прокрутки;
 - **Toolbar:** Квитирование - показывать кнопку квитирования, Очистить журнал - показывать кнопку очистки журнала;

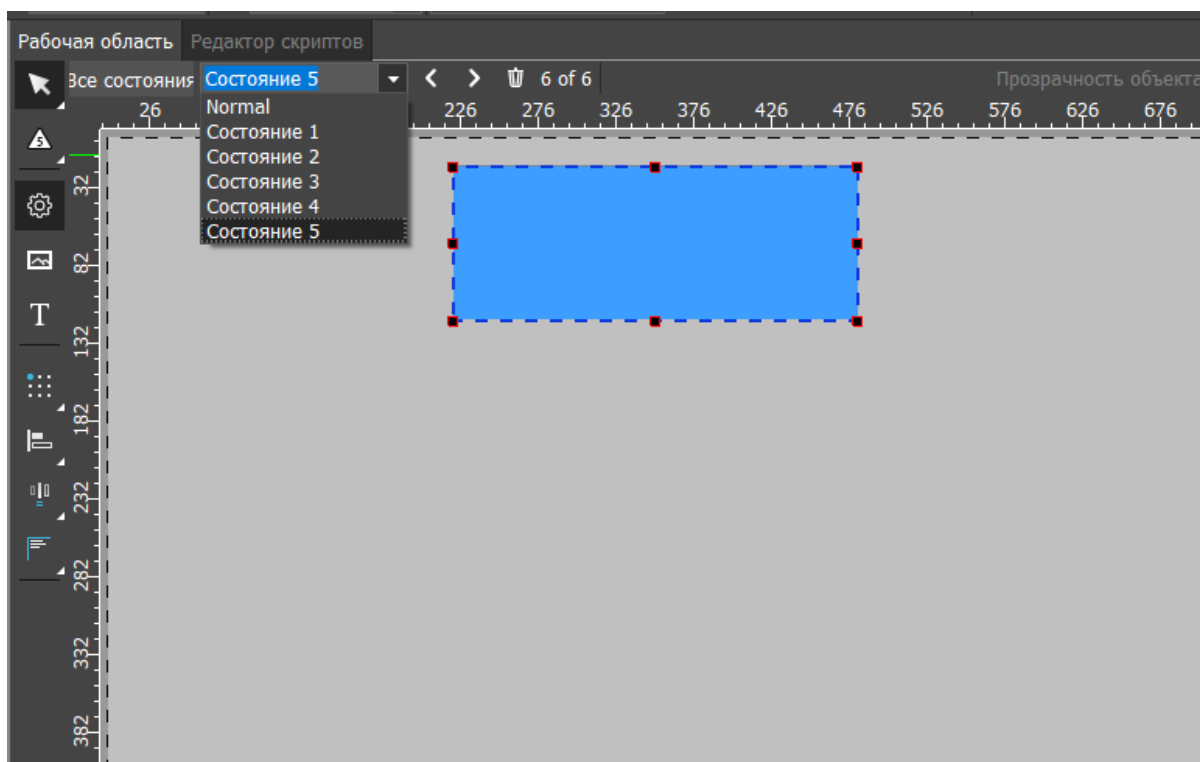


- **Стиль** - настройка стиля таблицы:
 - **Формат текста** - настройка шрифта, размера шрифта, оформления (жирный, курсив) у таблицы, заголовков колонок, кнопок на панели инструментов;
 - **Выравнивание** - настройка выравнивания текста у таблицы, заголовков колонок, кнопок на панели инструментов;
 - **Таблица** - настройка цвета фона и цвета сетки у таблицы;
 - **Заголовки** - настройка цвета фона и цвета текста у строки заголовка таблицы;
 - **Кнопки** - настройка цвета фона, цвета текста и цвета границ у кнопок на панели инструментов.



6.8. SCADA Alarm Indicator

6.8.1. Описание



SCADA Alarm Indicator - это графический элемент, предназначенный для отображения количества активных аварийных событий в реальном времени.

Настройка элемента производится в панельном проекте SCADA.

Элемент имеет **до 5 состояний** внешнего вида и состояние “Норма”.

Элемент может менять своё состояние (например, цвет или форму) в зависимости от заданных условий, таких как превышение порогового значения количества аварий.

Дополнительно поддерживается функция мигания для привлечения внимания оператора.

6.8.2. Принцип работы

SCADA Alarm Indicator получает информацию о количестве аварийных событий всего проекта.

Данные обновляются в реальном времени, учитывая только те события, которые соответствуют выбранным в фильтре критериям.

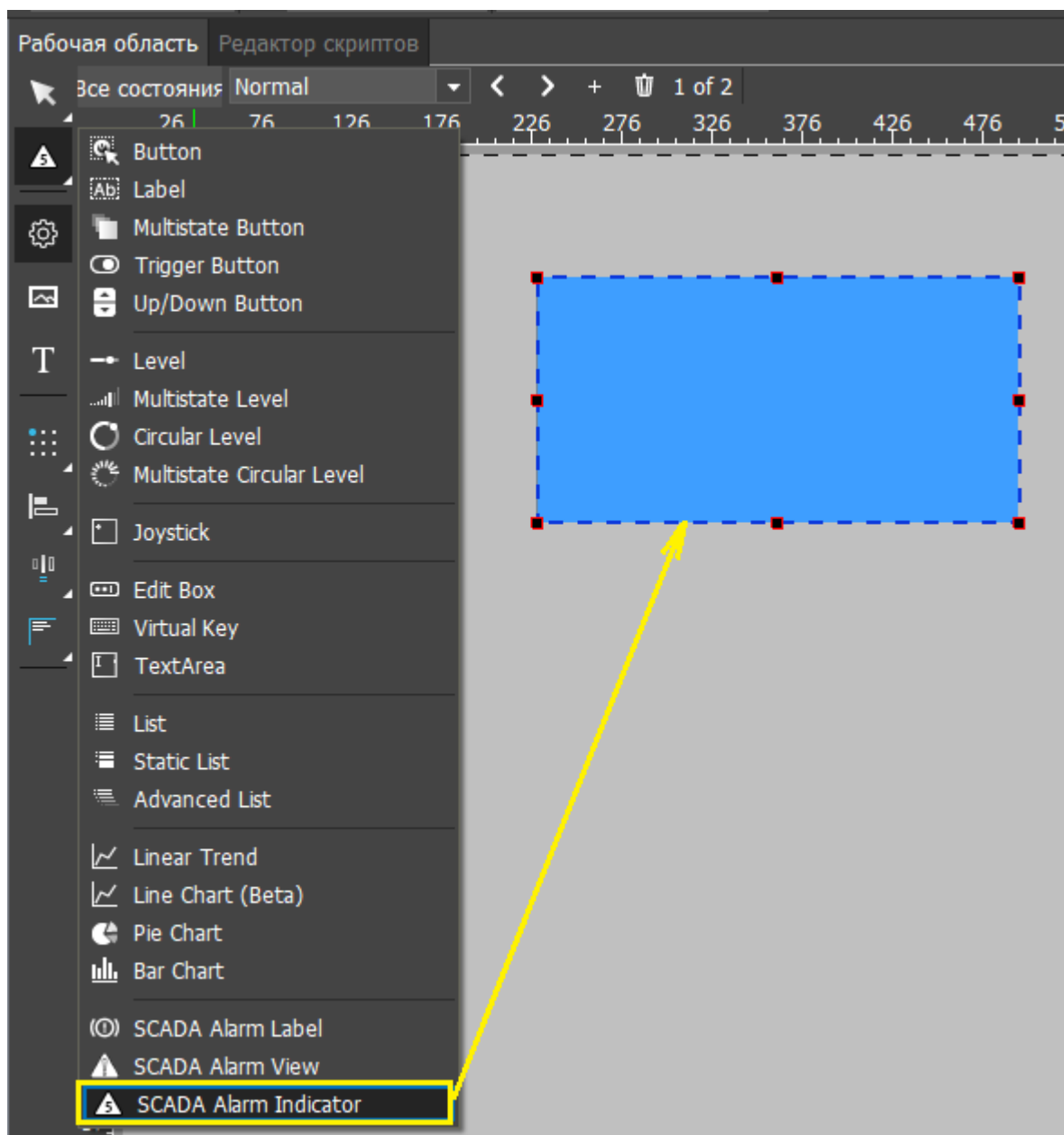
После получения количества активных событий элемент сравнивает его с заданными условиями (до 5 возможных условий):

- Условия задаются в виде неравенств (например, >2 , ≥ 5).
- При выполнении условия элемент переходит в соответствующее состояние (например, меняет цвет, форму или начинает мигать).
- Если ни одно условие не задано, элемент остаётся в состоянии Норма.

Элемент может менять внешний вид в зависимости от состояния (цвет, иконку и т.д.). Если включено мигание (Flashing), элемент периодически меняет внешний вид.

Значение элемента нельзя изменить через скрипты или вручную. Другие элементы или скрипты могут получать текущее количество аварий для логики системы.

6.8.3. Создание Alarm Indicator



Выберите элемент SCADA Alarm Indicator из списка графических элементов, перенесите его на страницу или попап и растяните до нужных размеров.

6.8.4. Параметры элемента

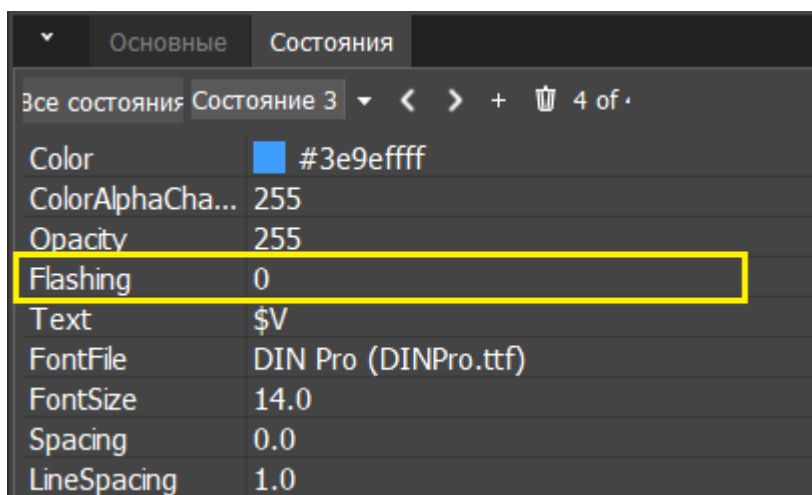
Основные параметры элемента:

▼	Основные	Состояния
Name	Item 1	
X	230	
Y	68	
Width	267	
Height	136	
Enable	True	
Visible	True	
Type	SCADA Alarm Indicator	
Tags		
Доступ	Администратор, Инженер, Оператор	

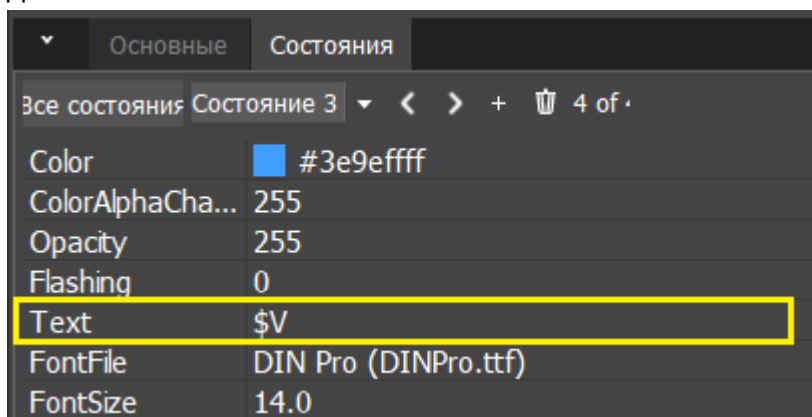
- **Name** - имя элемента;
- **X, Y, Width, Height** - положение и габаритные размеры;
- **Enable** - доступность управления элементом по умолчанию включена. Если выключить разрешение на нажатие, элемент будет заблокирован - перестанет реагировать на касание и отправлять команды, настроенные в Programming. При этом он продолжит отображать данные, полученные от оборудования. Цвет заблокированного элемента меняется - отображается в оттенках серого.
- **Visible** - отображение элемента по умолчанию включено. Если выключить отображение, при запуске i3 pro элемент будет полностью скрыт. Отображением элемента можно управлять в процессе работы i3Pro через свойство Visible;
- **Type** - SCADA Alarm Indicator;
- **Tags** - название, предназначенное для поиска графического элемента или группы элементов в проекте через поисковую строку;
- **Доступ** - выбор пользователей, которым доступен данный элемент. Для пользователей, которые не выбраны, элемент будет не доступен для отображения и использования.

Параметры состояния соответствуют параметрам остальных графических элементов и настраиваются для каждого из состояний отдельно.

Дополнительно в свойствах Alarm Indicator есть свойство для настройки частота мерцания - Flashing. Частота мерцания настраивается в герцах.

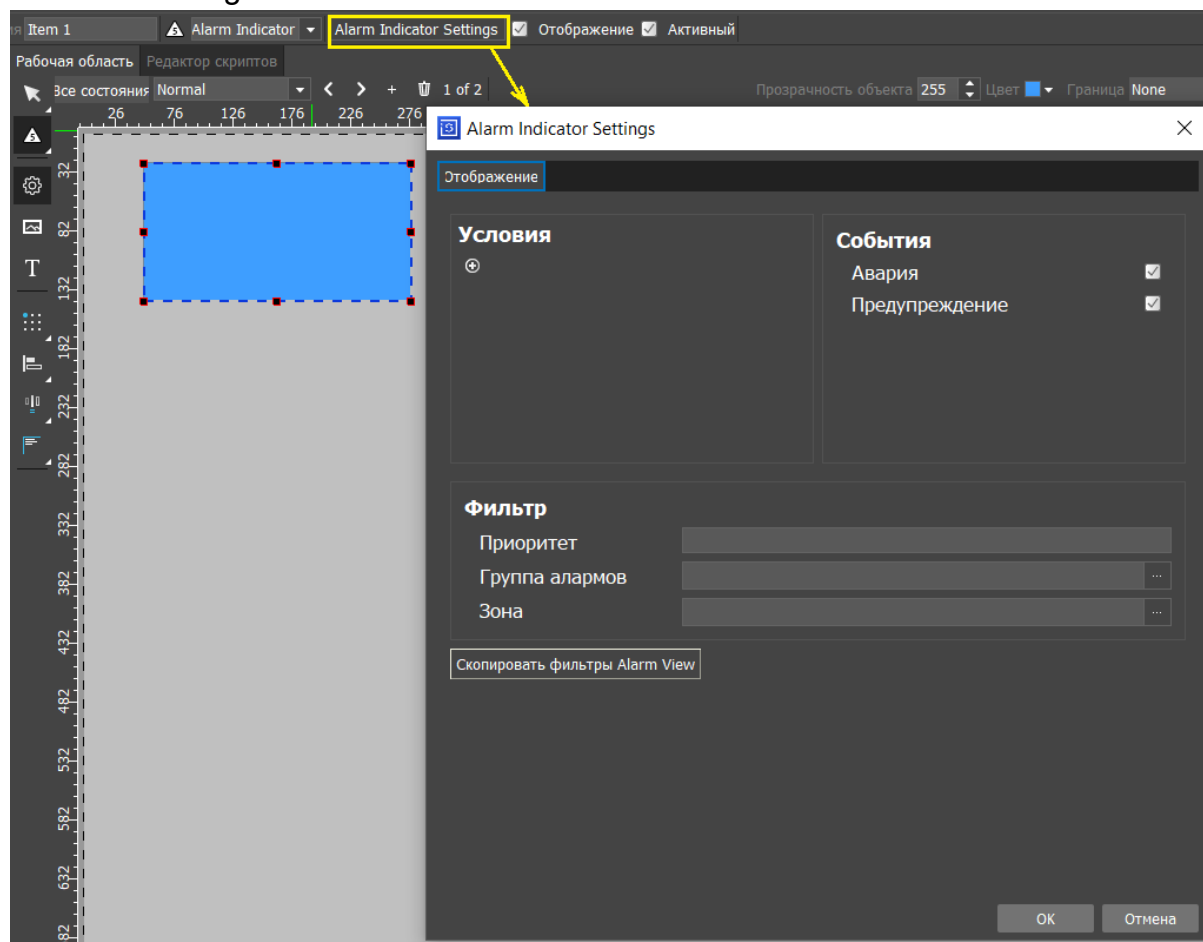


Для отображения количества аварий в проекте в свойстве текст необходимо добавить шаблон - \$V.



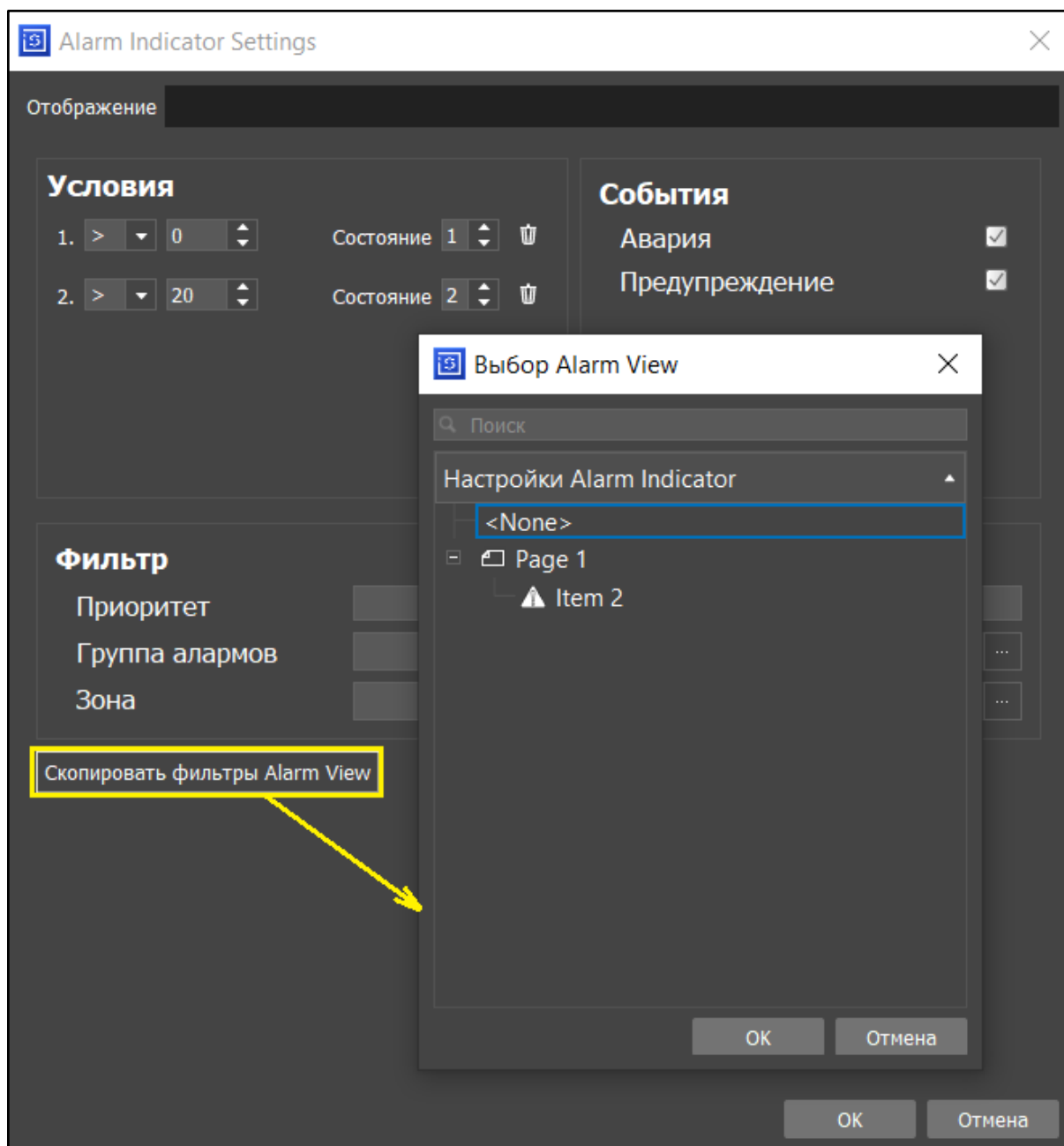
6.8.5. Настройка работы элемента SCADA Alarm Indicator

Работа элемента настраивается в отдельном окне “Alarm Indicator Setting”. Окно можно вызвать двойным щелчком по элементу или нажатием кнопки “Alarm Indicator Setting” на панели свойств элемента.



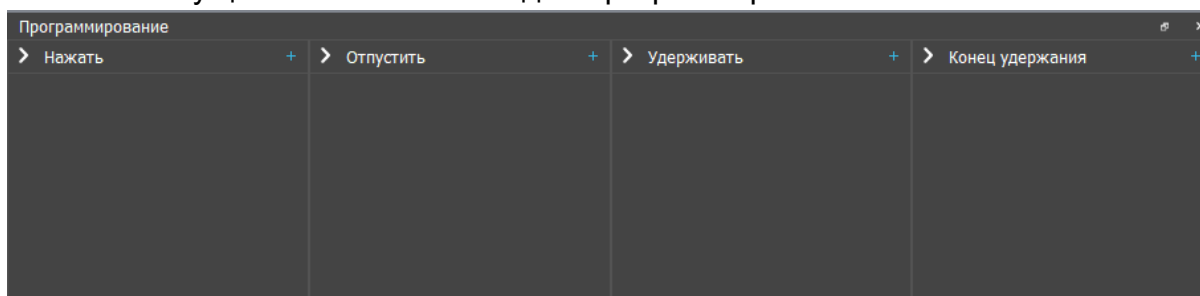
Настройка SCADA Alarm Indicator:

- **Условия** - по умолчанию условий нет, элемент всегда в состоянии Норма.
 - Можно добавить до 5 условий (кнопка ).
 - Каждое условие состоит из:
 - Оператор сравнения (> или >=).
 - Значение (например, 3).
 - Состояние, в которое переходит элемент при выполнении условия.

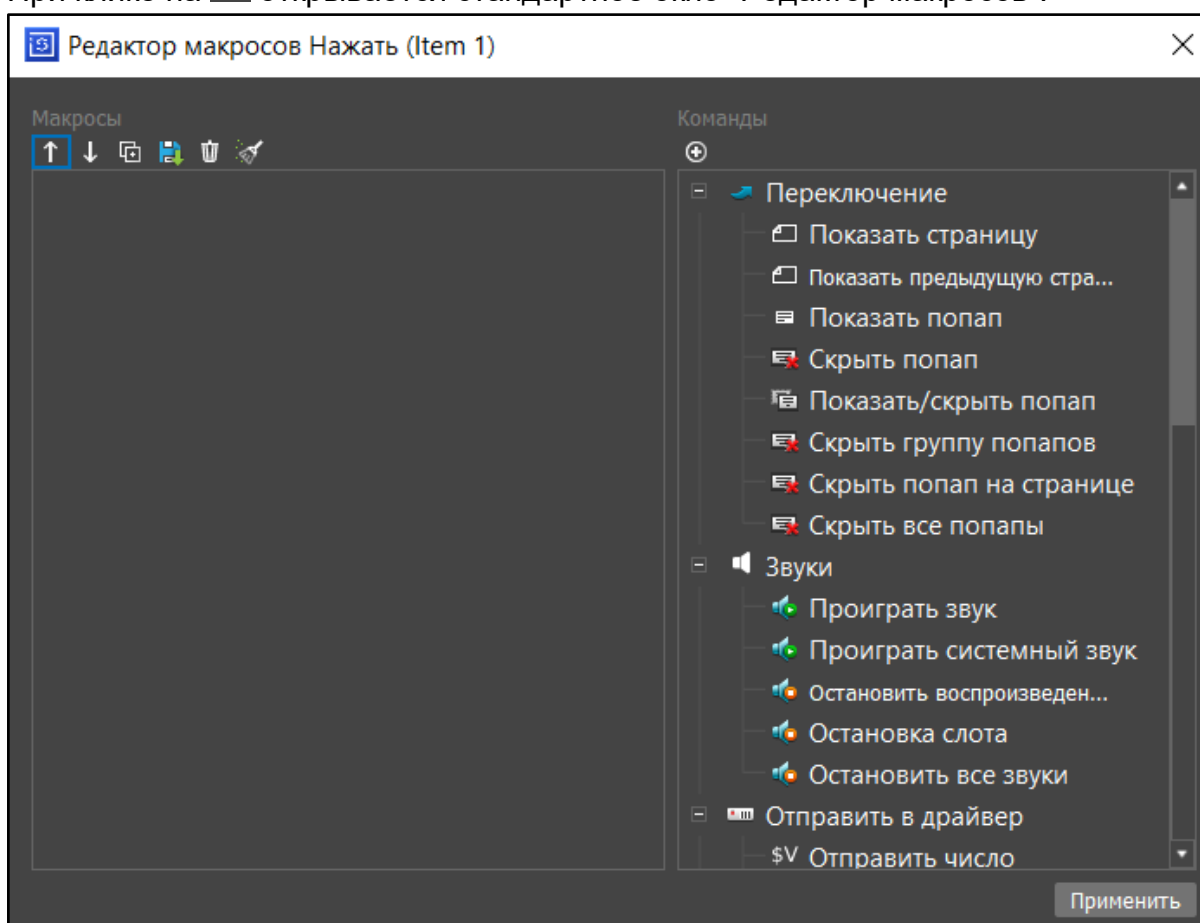


6.8.6. Программирование элемента SCADA Alarm Indicator

У элемента SCADA Alarm Indicator есть возможность выполнять действия по нажатию, отпусканию, удержанию и концу удержания. Программирование элемента осуществляется во вкладке Программирование.



При клике на  открывается стандартное окно “Редактор макросов”.



7. Авторизация

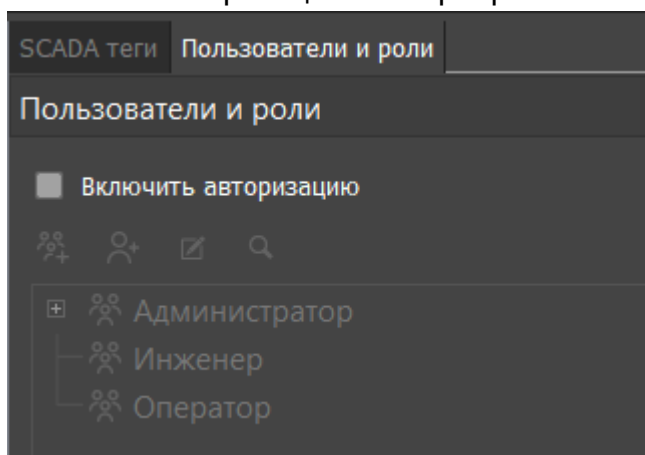
7.1. Роли и доступ

Ролевая система доступа позволяет разграничивать возможности взаимодействия с SCADA системой пользователям различных групп (должностей).

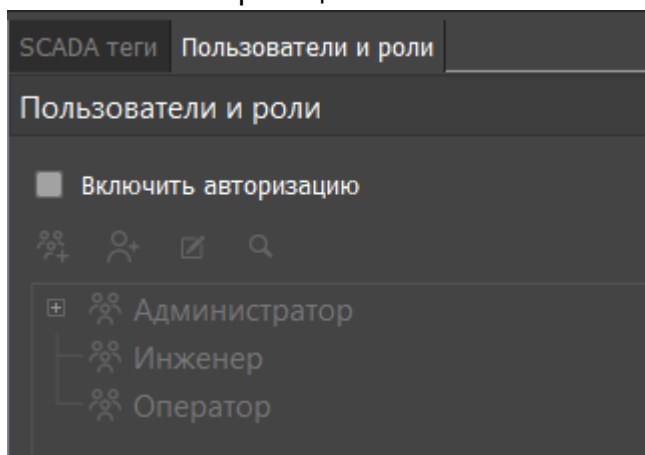
Особенности данной системы:

- Набор пользователей и ролей един для всей SCADA-системы;
- Разделяет доступ к:
 - редактированию и считыванию SCADA - тегов,
 - дашбордам в веб-интерфейсе,
 - загруженным панельным SCADA проектам,
 - доступу к графическим элементам SCADA-проекта.

Настройка ролей и пользователей проекта производится на вкладке “Роли и пользователи” SCADA-проекта iRidi Studio. Для настройки ролей необходимо включить авторизацию на сервере.



Для включения системы авторизации пользователей - установите чекбокс “Включить авторизацию”.



7.2. Создание ролей пользователей

7.2.1. Роли пользователей

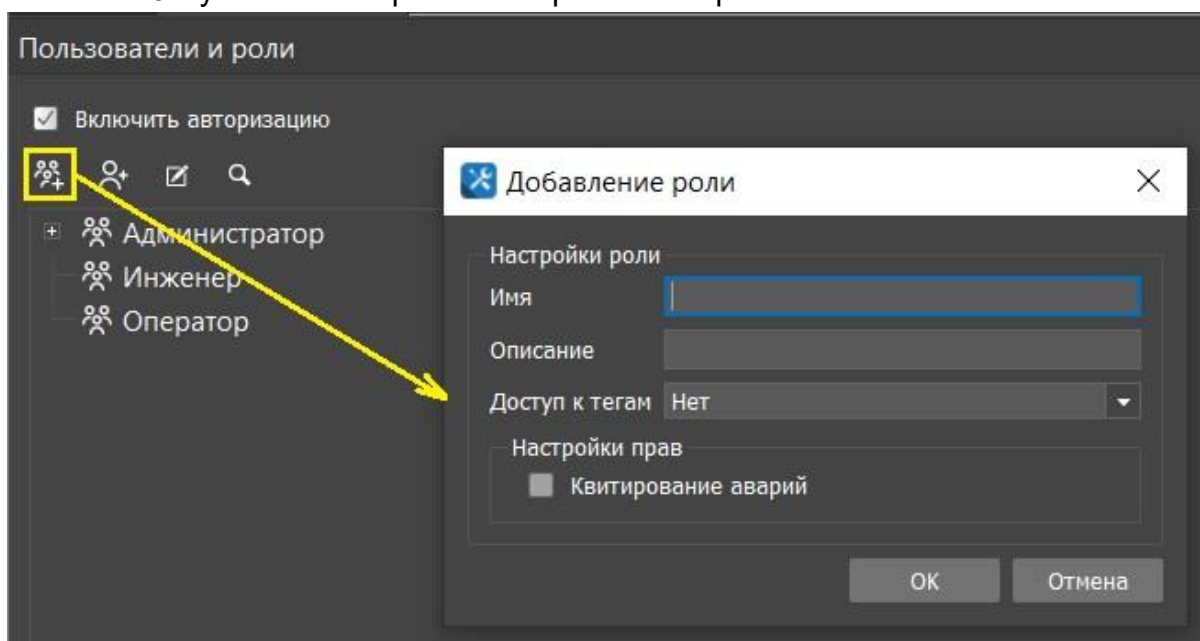
По умолчанию создаются три роли пользователей:

- Администратор
- Инженер
- Оператор

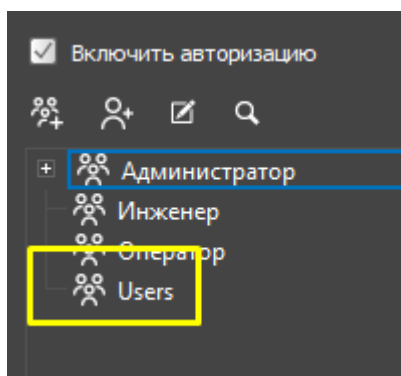
Роль “Администратор” по умолчанию имеет уже одну учетную запись, ее нельзя удалить из проекта, как и саму группу, но можно изменить пароль доступа. При необходимости в группу можно добавить дополнительные учетные записи.

Для создания новой роли необходимо:

1. Создать или открыть существующий SCADA-проект;
2. Включить авторизацию в проекте;
3. Нажать на кнопку “Добавления роли”;
4. В открывшемся окне произвести настройку роли:
 - a. ввести название - имя роли;
 - b. ввести описание роли;
 - c. установить права доступа к **уже существующим** тегам;
 - d. установить права квитировать аварии.



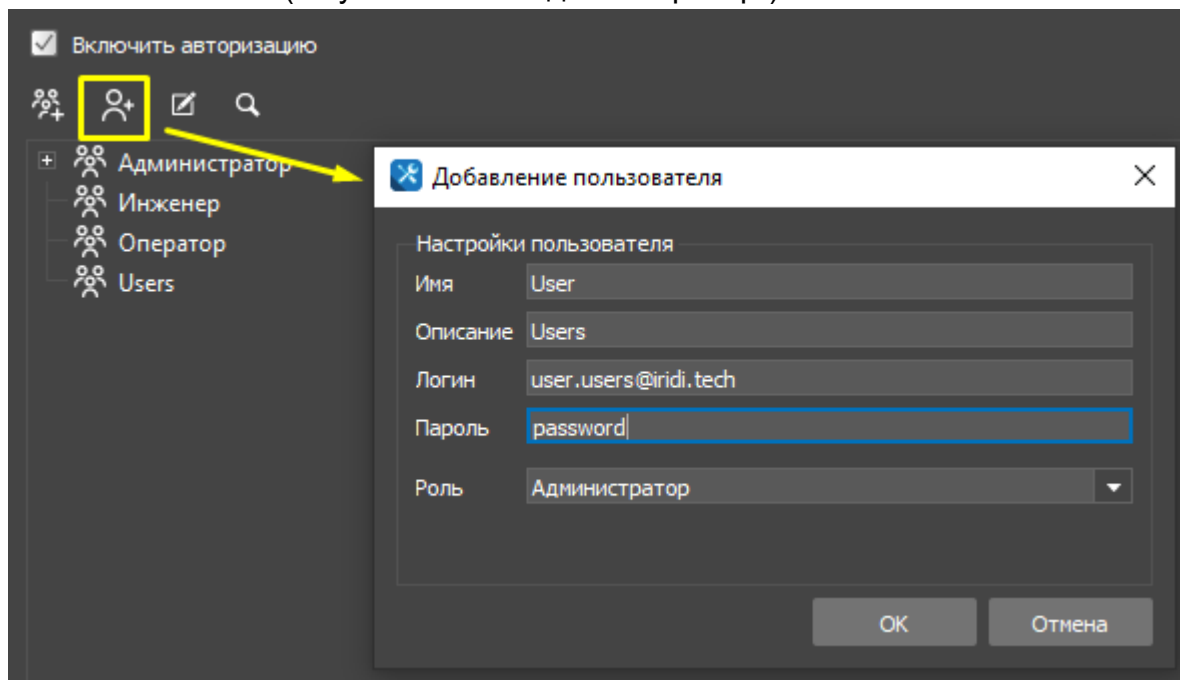
5. После сохранения настроек в списке появится новая роль:



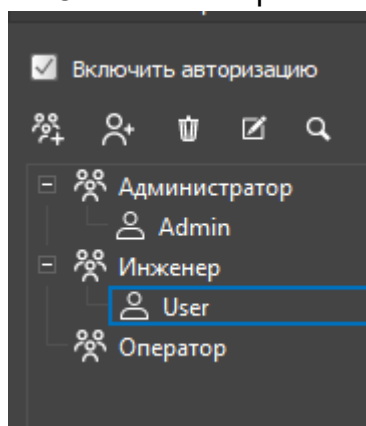
Примечание. При создании новой группы пользователей параметр “Доступ к тегам” применяется для всех существующих SCADA-тегов, для быстрой раздачи прав основной массе SCADA-тегов. При создании нового SCADA-тега он доступен всем группам на чтение и запись.

7.2.2. Пользователи

1. Для создания пользователя необходимо нажать кнопку “Добавить пользователя”;
2. В открывшемся диалоге заполнить данные пользователя:
 - а. Имя,
 - б. Описание,
 - в. Логин (минимум 6 символов),
 - г. Пароль (минимум 6 символов),
 - д. Роль (по умолчанию “Администратор”).



3. После сохранения в выбранной группе появится пользователь:

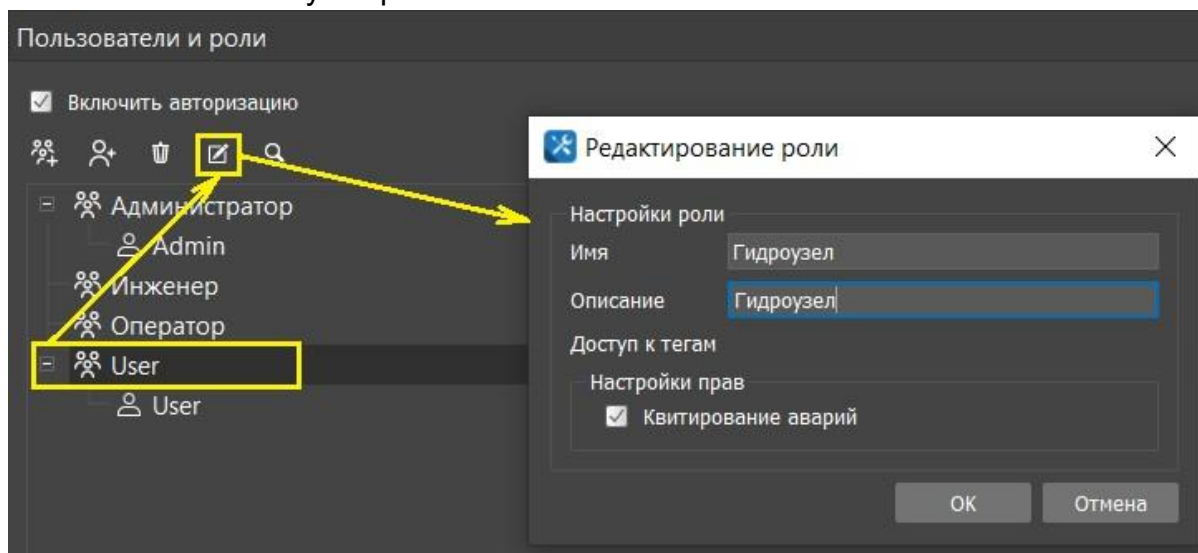


7.3. Редактирование ролей и пользователей

7.3.1. Редактирование роли

Для редактирования роли необходимо:

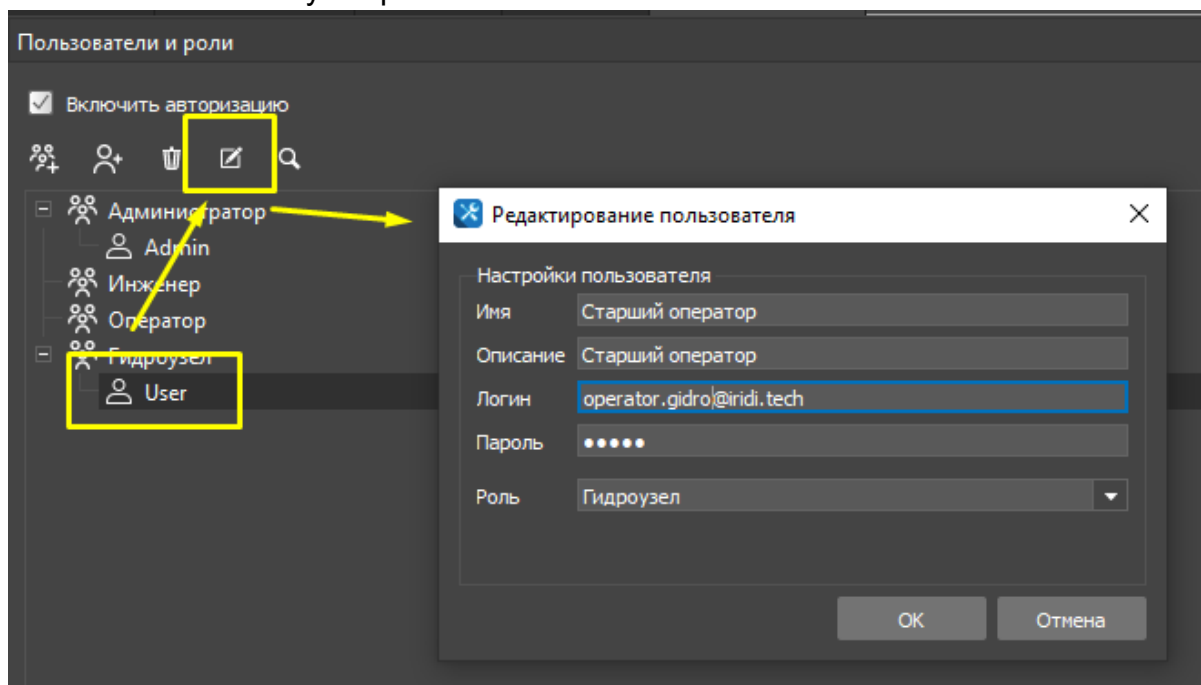
1. Выделить группу в списке;
2. Нажать кнопку “Редактирования роли”;
3. Изменить необходимые параметры;
4. Нажать кнопку сохранить.



7.3.2. Редактирование пользователей

Для редактирования пользователя необходимо:

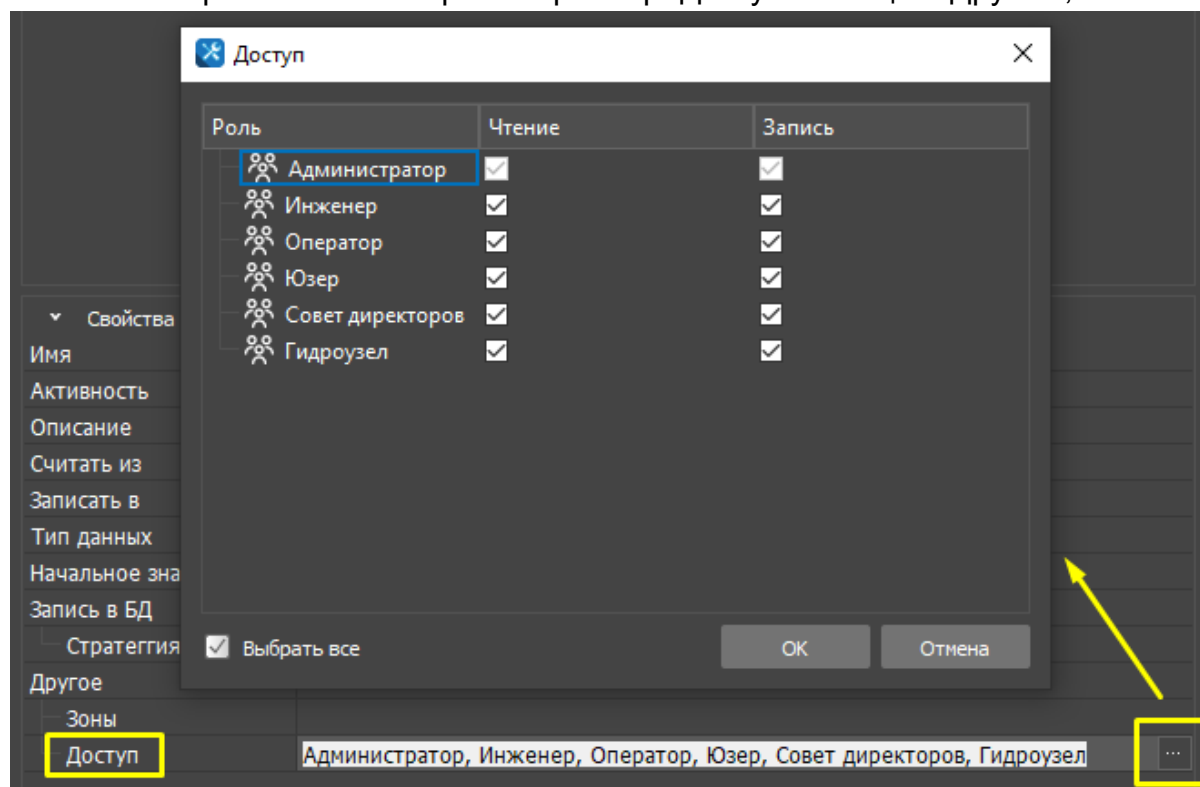
1. Выделить пользователя в списке;
2. Нажать кнопку “Редактирования пользователя”;
3. Изменить необходимые параметры;
4. Нажать кнопку сохранить.



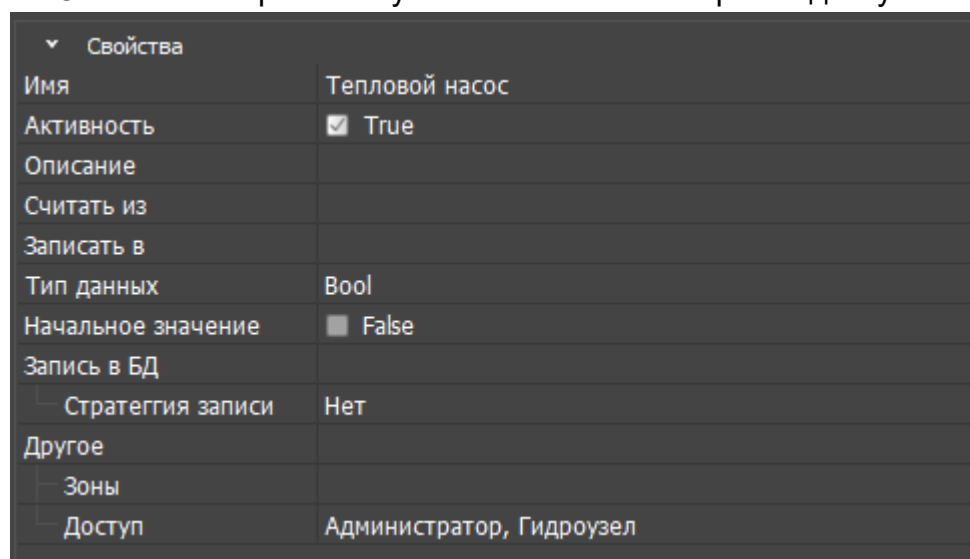
7.4. Настройка доступа к SCADA-тегам

Для настройки доступности тега пользователю необходимо:

1. В дереве SCADA-тегов выделить необходимый тег;
2. В настройках тега выбрать параметр “Доступ” в секции “Другое”;



3. В открывшемся окне выбрать роль и права доступа к тегу;
4. Сохранить изменения.
5. После сохранения у тега обновятся настройки доступа.

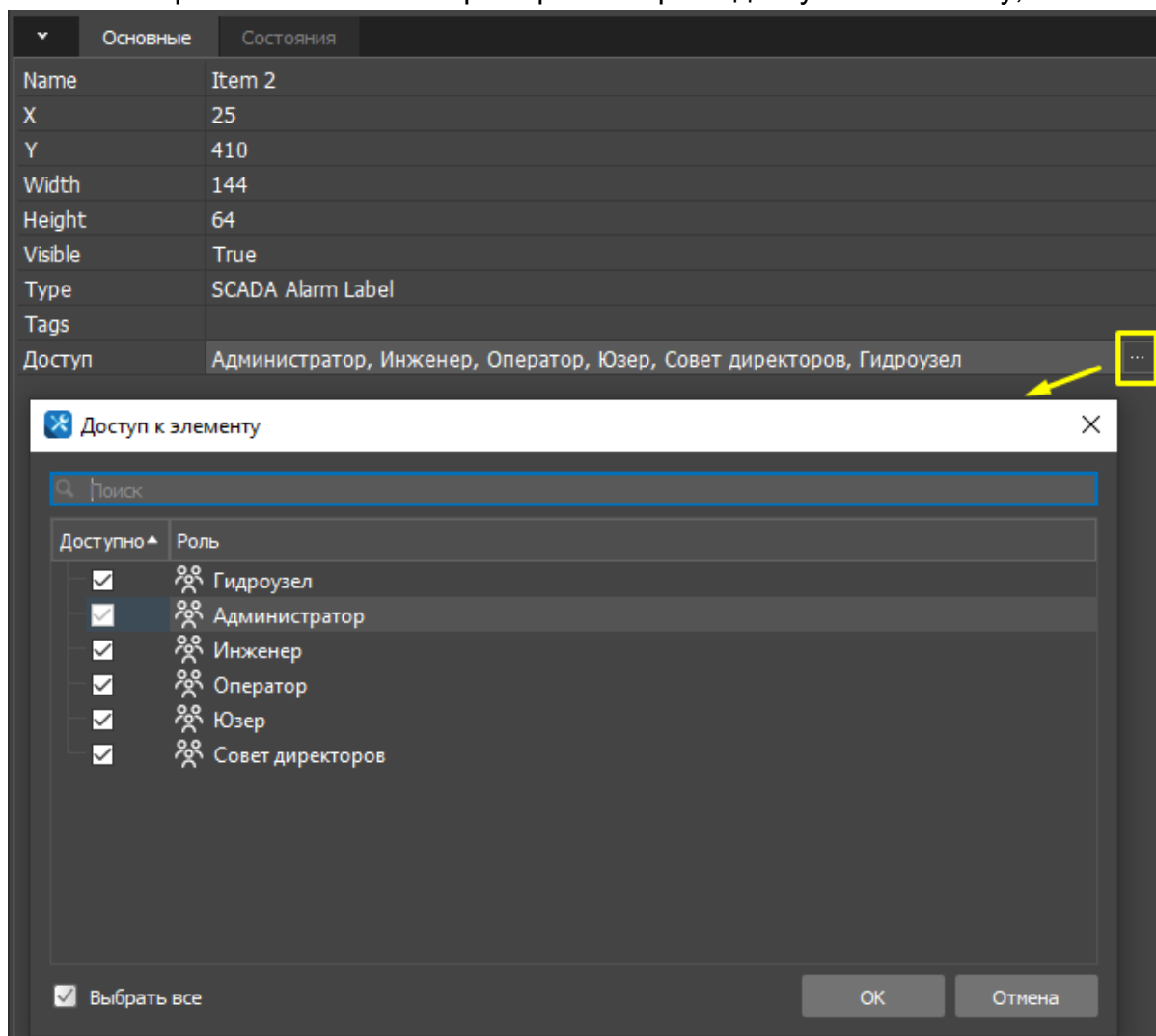


Примечание. По умолчанию новый элемент доступен всем ролям пользователей на чтение и запись. Роли “Администратор” **нельзя отключить** доступ к SCADA-тегу в настройках элемента.

7.5. Настройка доступа к графическим элементам в проекте SCADA-клиента

Для настройки доступности элемента пользователю необходимо:

1. В дереве проекта выделить необходимый элемент;
2. В основных настройках элемента выбрать параметр “Доступ”;
3. В открывшемся окне выбрать роли и права доступа к элементу;



4. Сохранить изменения.
5. После сохранения у элемента обновятся настройку доступа.

▼	Основные	Состояния
Name	Item 2	
X	25	
Y	410	
Width	144	
Height	64	
Visible	True	
Type	SCADA Alarm Label	
Tags		
Доступ	Администратор, Гидроузел	

Примечание. По умолчанию новый элемент доступен всем ролям пользователей. Роли “Администратор” **всегда доступны все** SCADA теги, роли **нельзя отключить** доступ в настройках элемента.

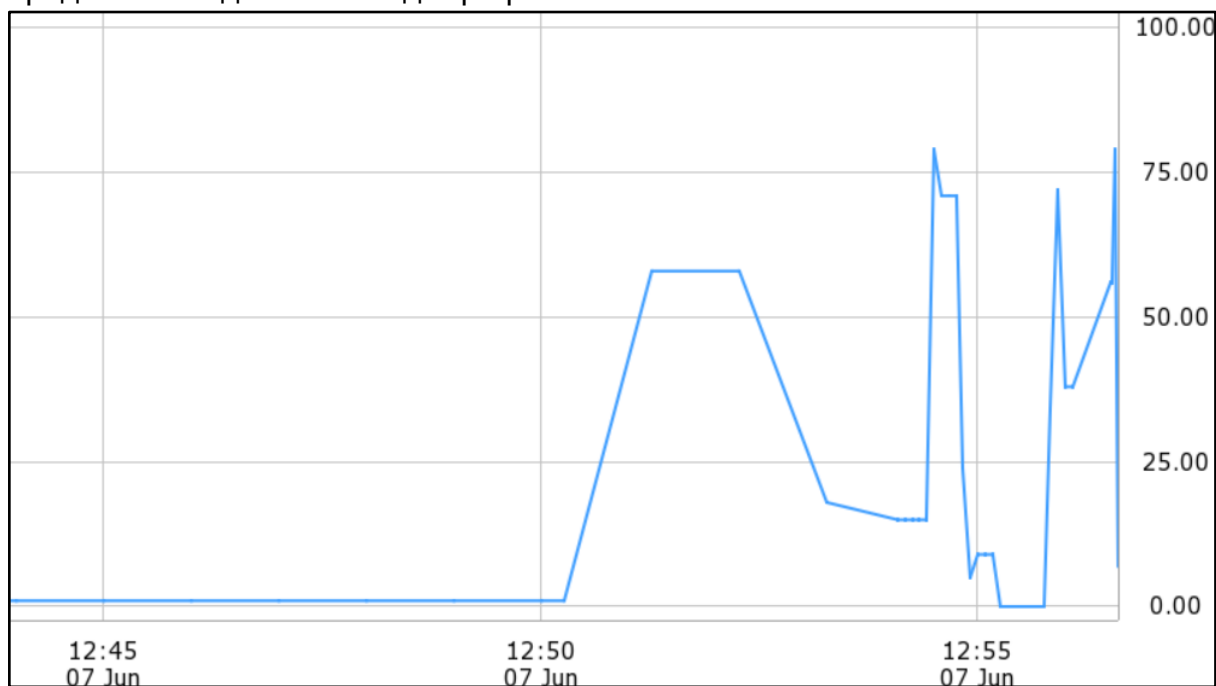
После запуска проекта в SCADA-клиенте, в интерфейсе отображаются только доступные для авторизованного пользователя элементы.

Если к элементу управления привязан SCADA-тег, к которому у пользователя нет доступа, то элемент не сможет отправлять и получать значения, в том числе и через **скрипты**.

8. Архивация тегов

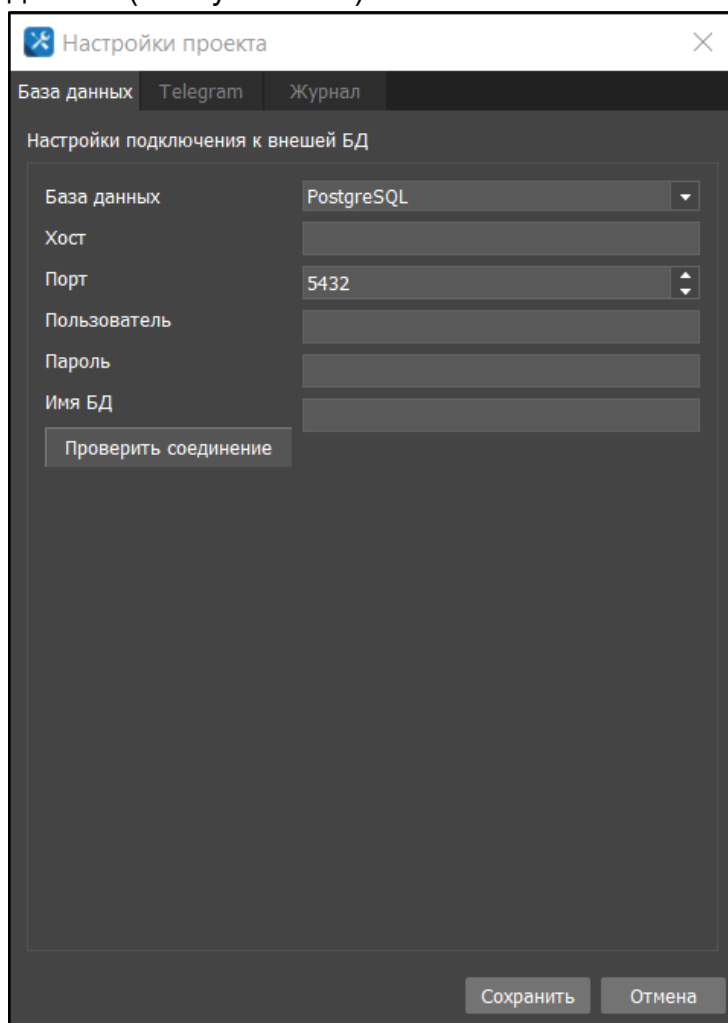
8.1. Описание

Архивация тегов - позволяет записать значение тегов в базу данных и представлять данные в виде графиков.



8.2. Настройка базы данных и тегов

Для архивации значений тегов, необходимо в настройках проекта добавить базу данных (см. пункт 1.2.1).



The image shows a screenshot of a software window titled 'Настройки проекта' (Project Settings). It has three tabs: 'База данных' (Database), 'Telegram', and 'Журнал' (Log). The 'База данных' tab is active. Below the tabs, the title 'Настройки подключения к внешней БД' (External database connection settings) is displayed. The settings are organized into a table-like structure with labels on the left and input fields on the right:

База данных	PostgreSQL
Хост	
Порт	5432
Пользователь	
Пароль	
Имя БД	

Below the input fields is a button labeled 'Проверить соединение' (Check connection). At the bottom right of the dialog are two buttons: 'Сохранить' (Save) and 'Отмена' (Cancel).

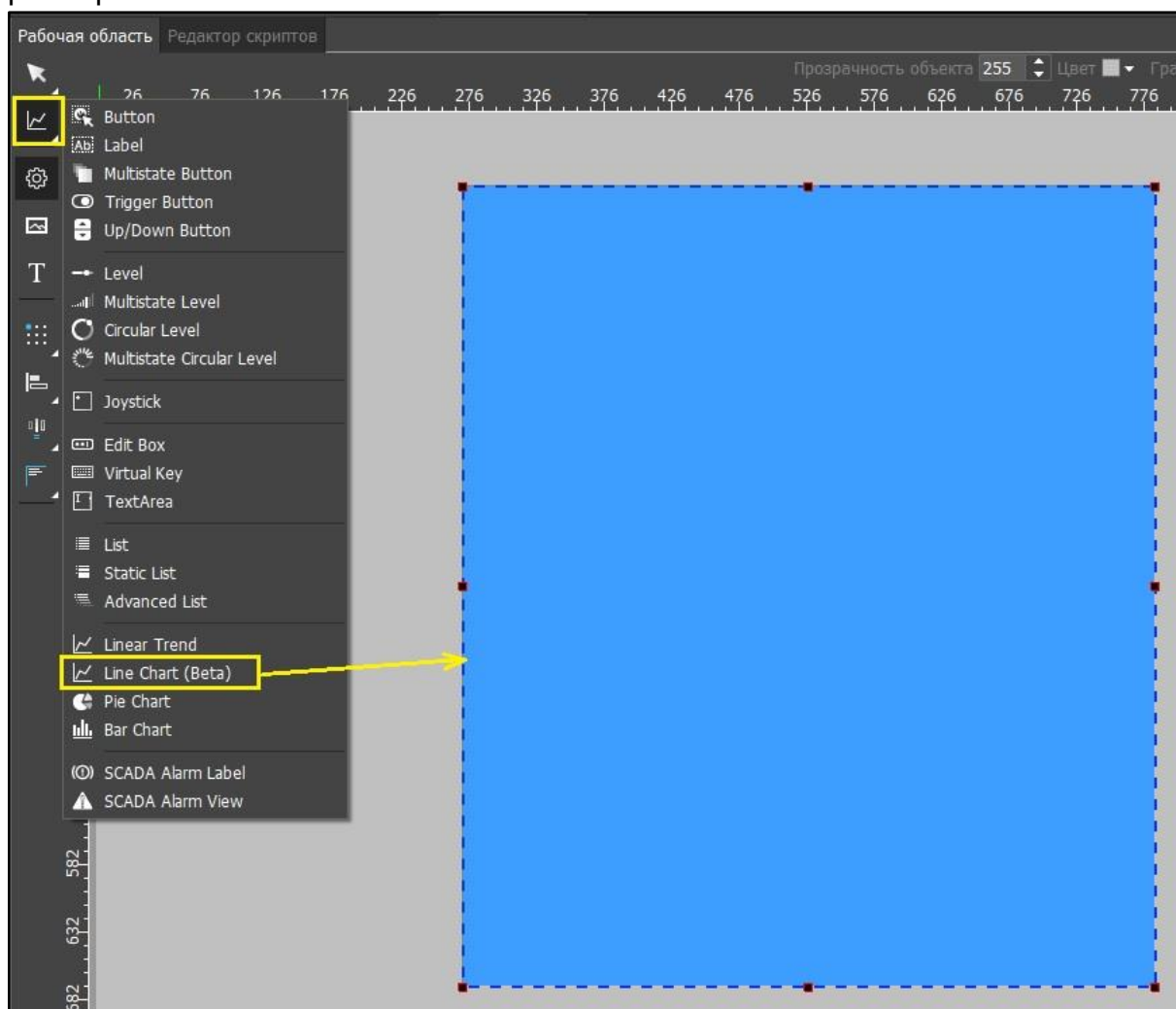
В настройках SCADA-тегов, которые необходимо задать следующие параметры:

- **Запись в БД** - Стратегия записи в базу данных:
 - Нет - запись в базу данных не ведётся;
 - По времени - запись в базу данных через фиксированный интервал;
 - Интервал - временной интервал записи в базу данных;
 - Зона чувствительности - запись в базу данных по изменению значения тега;
 - Значение - значение, на которое должен измениться тег для записи в БД;

▼ Свойства	
Имя	Tag
Активность	☒ True
Описание	
Считать из	Drivers.Bus77.Tags.FM-402-1W:State (Реле 1)
Записать в	Drivers.Bus77.Channels.FM-402-1W:State (Реле 1)
Тип данных	Bool
Начальное значение	☐ False
Запись в БД	
Стратегия записи	По времени
Интервал	1 минута
Другое	
Зоны	
Доступ	Администратор, Инженер, Оператор

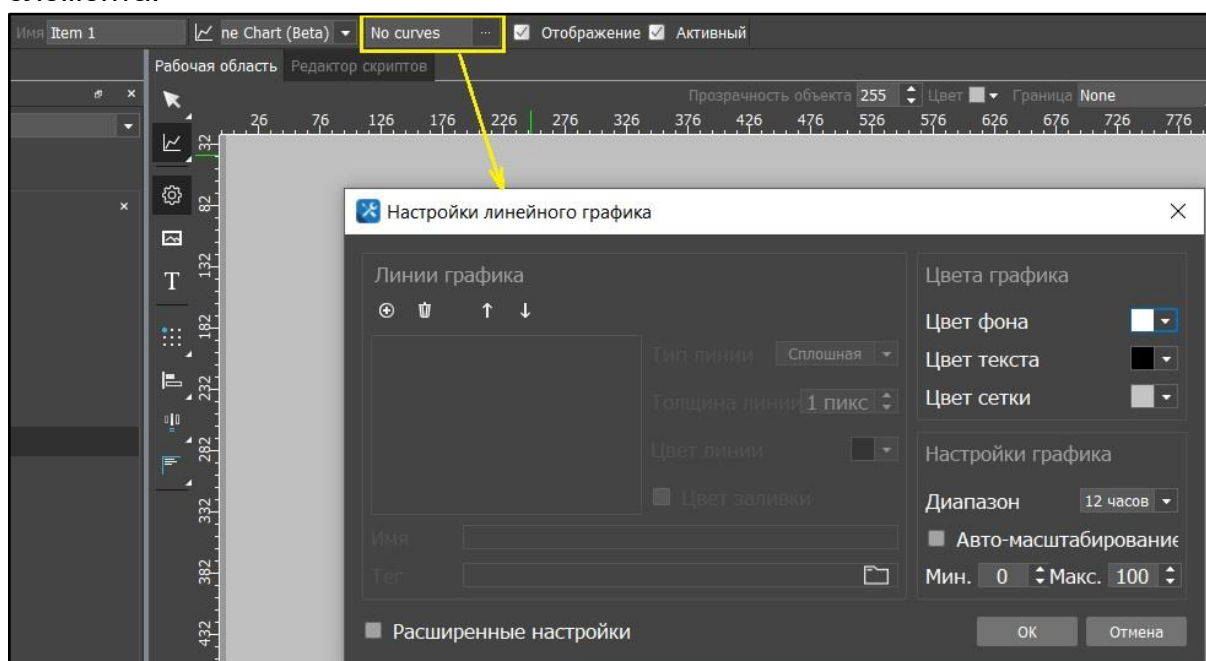
8.3. Создание Line Chart (Beta)

В SCADA-клиенте выберите элемент Line Chart (Beta) из списка графических элементов, перенесите его на страницу или попап и растяните до нужных размеров.



8.4. Добавление тегов

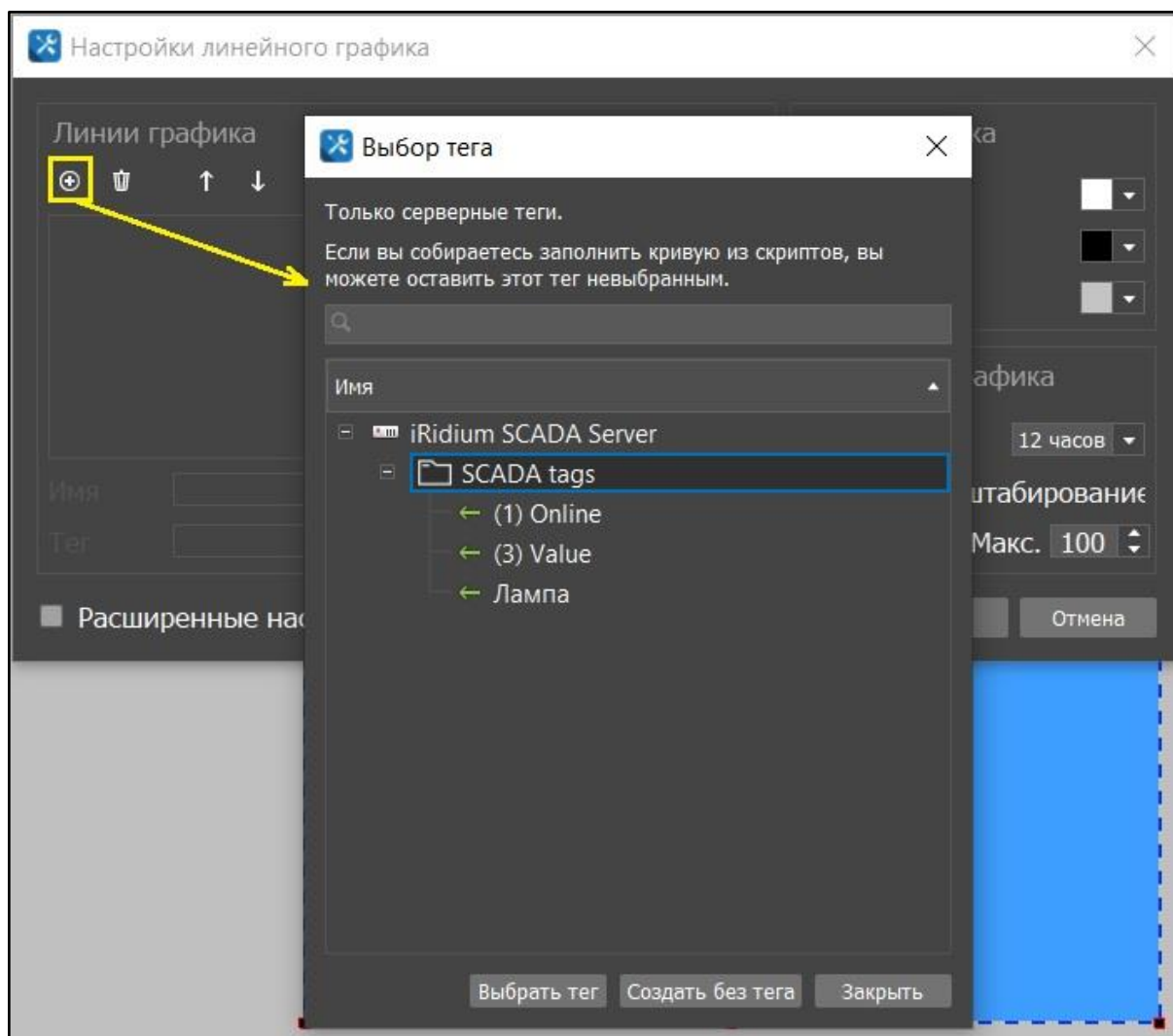
Внешний вид элемента и добавление тегов настраивается в отдельном окне “Настройка Line Chart (Beta)”. Окно можно вызвать двойным щелчком по элементу или нажатием кнопки “Настройка Line Chart (Beta)” на панели свойств элемента.



8.4.1. Добавление тегов в Line Chart (Beta)

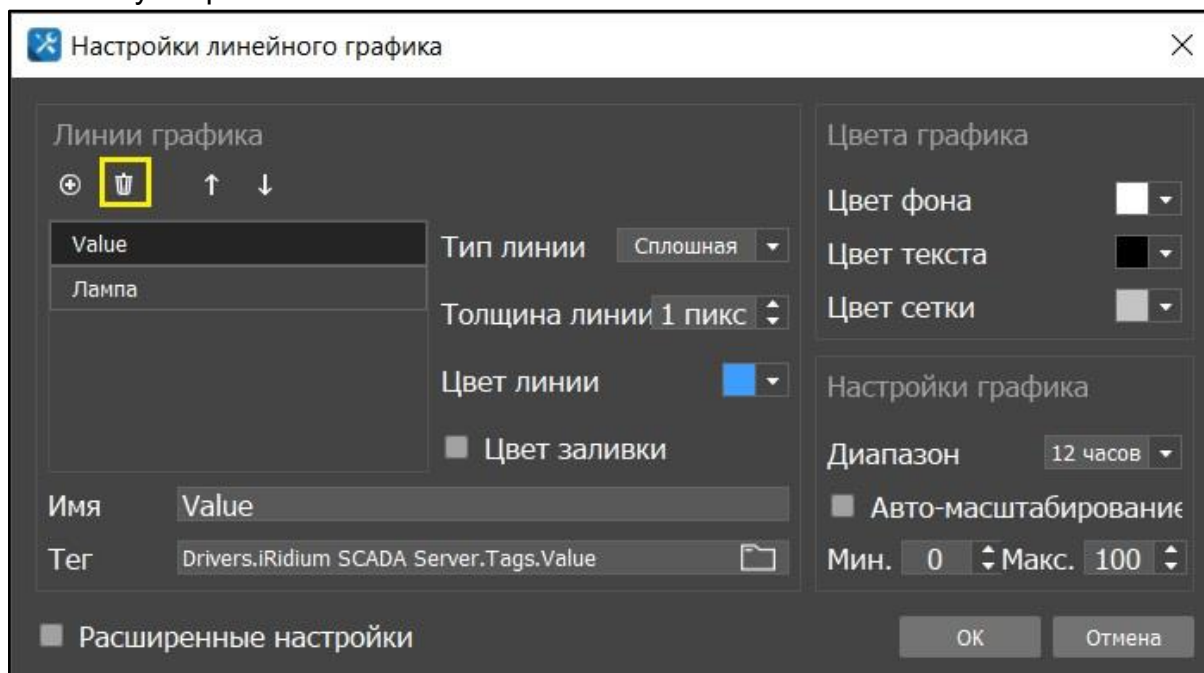
Для того чтобы добавить тег, необходимо:

1. нажать на иконку “+” в разделе Линии графика;
2. выбрать тег из списка тегов и нажать на кнопку “Выбрать тег”. Или выбрать вариант - “Создать без тега”.



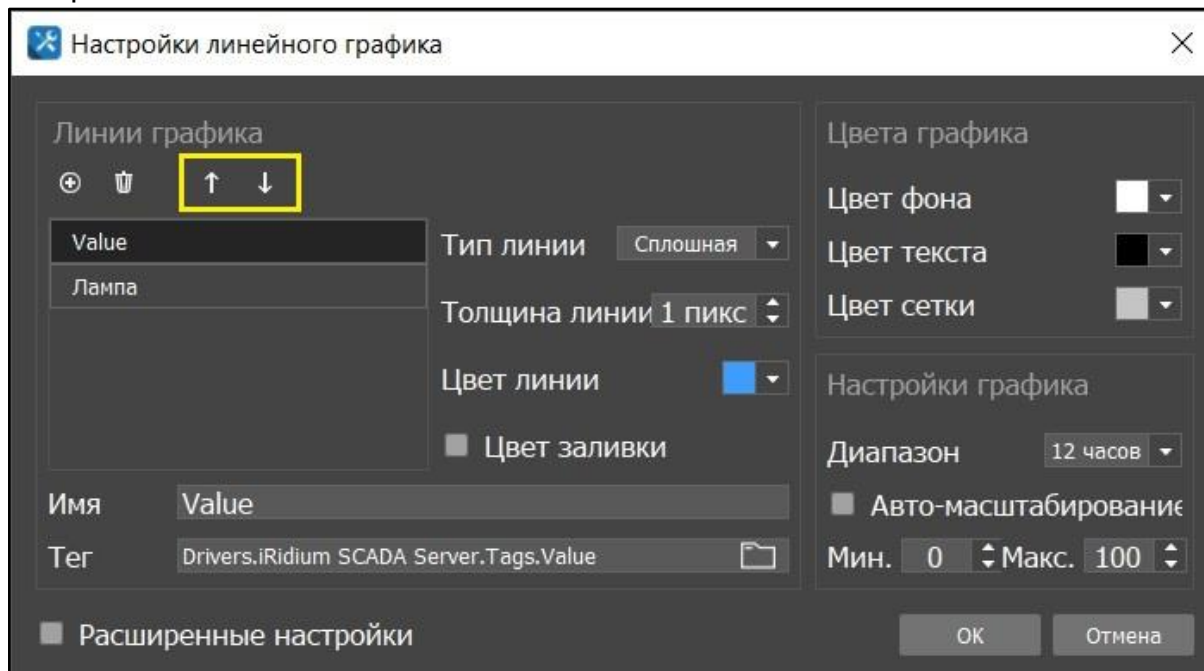
8.4.2. Удаление тегов в Line Chart (Beta)

Для удаления тега нужно выбрать тег, который необходимо удалить, и нажать на иконку “Корзина”.



8.4.3. Перемещение тегов

Для перемещения тегов нужно выбрать нужный тег и при помощи стрелок “вверх” и “вниз” изменить положение тега.



8.5. Настройка внешнего вида Line Chart (Beta)

8.5.1. Основные параметры элемента

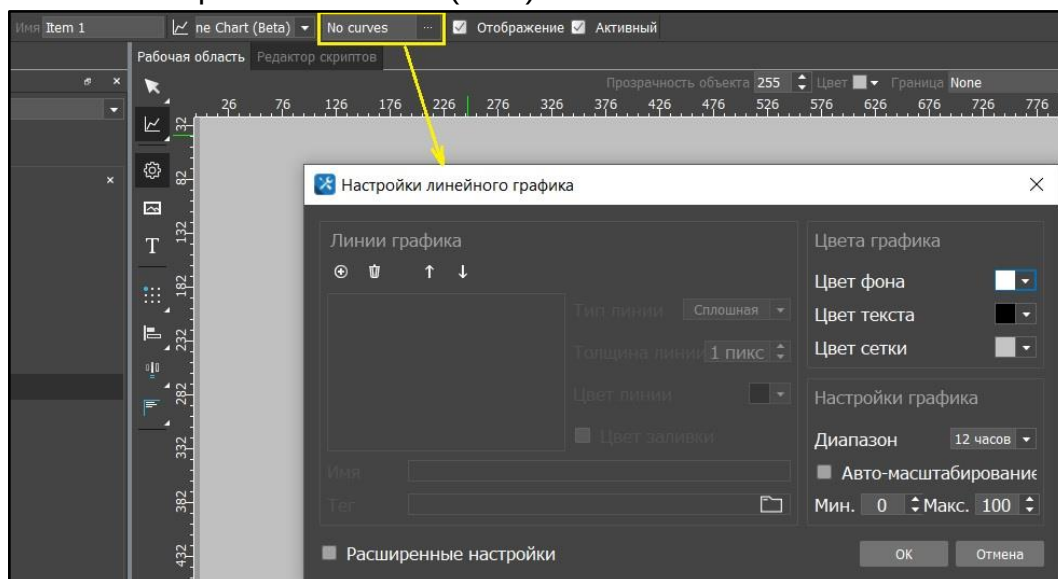
▼	Основные	Состояния
Name	Item 1	
X	272	
Y	80	
Width	513	
Height	593	
Enable	True	
Visible	True	
Type	Line Chart (Beta)	
Tags		
Доступ	Администратор, Инженер, Оператор	

В основных параметрах элемента настраиваются:

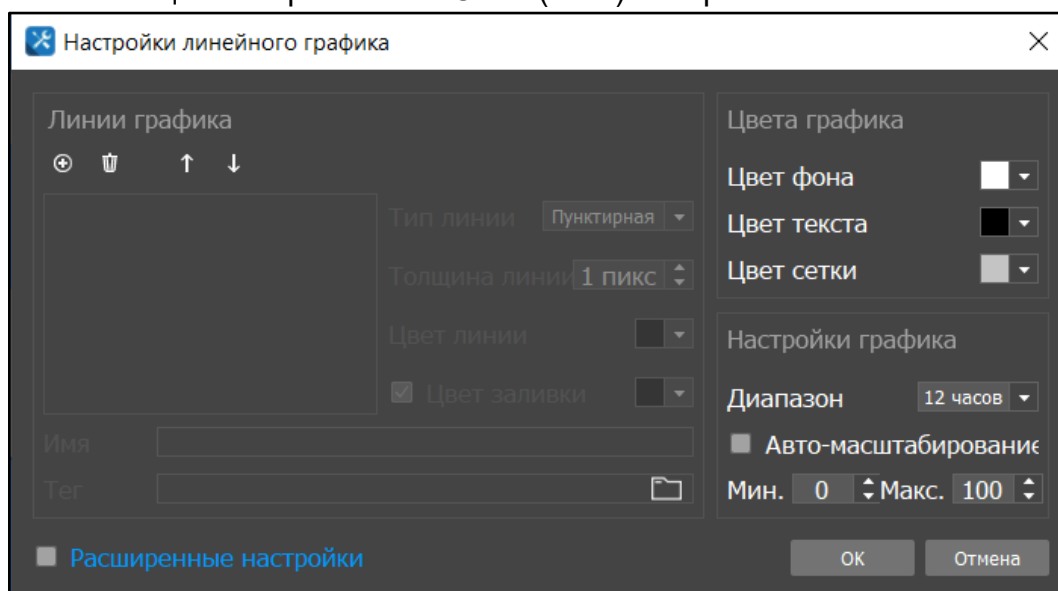
- **Name** - имя элемента;
- **X, Y, Width, Height** - положение и габаритные размеры;
- **Visible** - видимость элемента при запуске проекта на управляющей панели. Отображение элемента по умолчанию включено. Если выключить отображение, при запуске i3 pro элемент будет полностью скрыт. Отображением элемента можно управлять в процессе работы i3 pro через свойство Visible.
- **Enable** - доступность управления элементом при запуске проекта на управляющей панели. Доступность по умолчанию включена. Если выключить доступность, при запуске i3 pro элемент будет показан, но не активен. Доступностью элемента можно управлять в процессе работы i3 pro через свойство Enable.
- **Type** - Line Chart (Beta);
- **Tags** - название, предназначенное для поиска графического элемента или группы элементов в проекте через поисковую строку.
- **Доступ** - выбор пользователей, которым доступен данный элемент. Для пользователей, которые не выбраны, элемент будет не доступен для отображения и использования.

8.5.2. Настройки элемента

Внешний вид элемента настраивается в отдельном окне “Настройка Line Chart (Beta)”. Окно можно вызвать двойным щелчком по элементу или нажатием кнопки “Настройка Line Chart (Beta)” на панели свойств элемента.



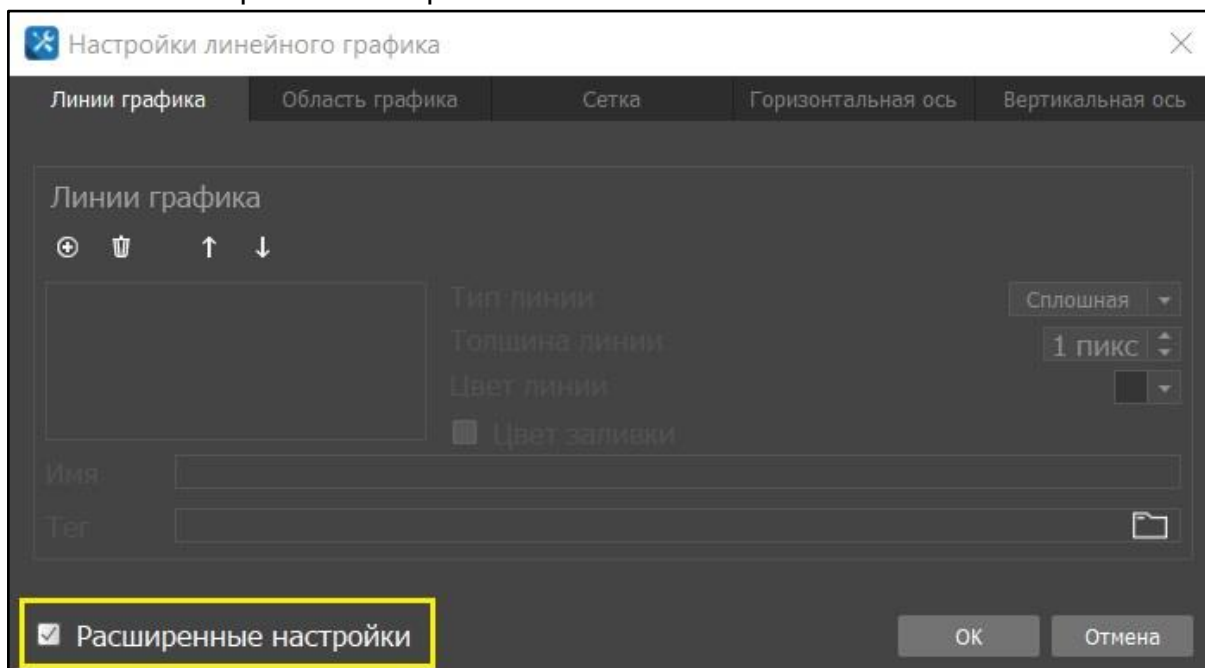
В окне общей настройки Line Chart (Beta) отображается:



- **Линии графика** - настройка тегов, которые необходимо отображать на графике; настройка типа, толщины и цвета линий (более подробно читай в п.8.4);
- **Цвет графика** - настройка цвета фона, цвета текста и цвета сетки;
- **Настройки графика**:
 - **Диапазон** - временной диапазон, в котором отображается график;
 - **Автомасштабирование** - настройка диапазонов отображения происходит автоматически, в зависимости от значений тега;
 - **Мин. и Макс.** - настройка диапазона значений тега, в пределах которого будет отображаться график.

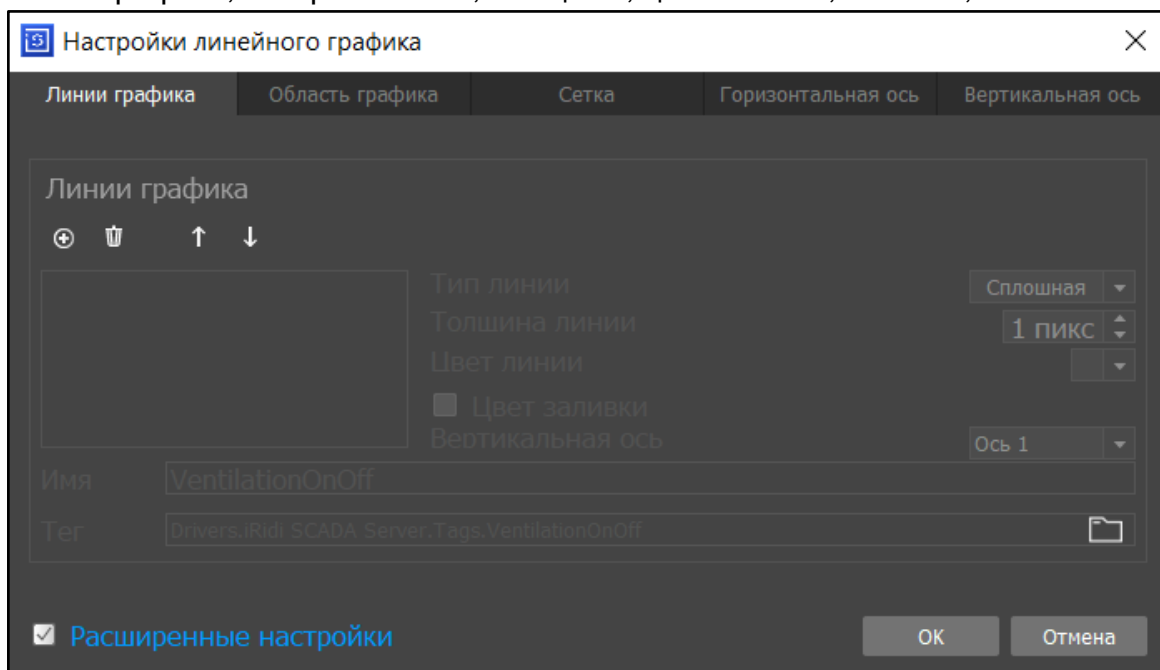
8.5.3. Расширенные настройки

Для того чтобы увидеть расширенные настройки, необходимо установить чекбокс “Расширенные настройки”.



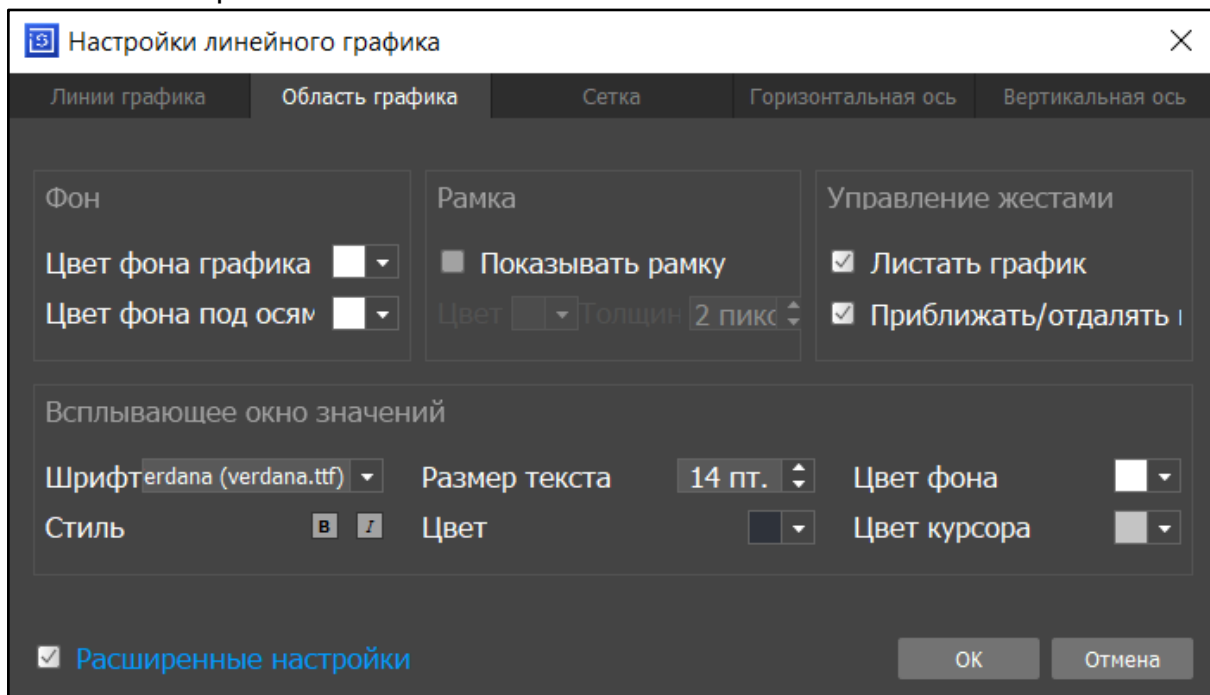
Расширенные настройки линейного графика Line Chart (Beta):

- **Линии графика** - настройка тегов, которые необходимо отображать на графике; настройка типа, толщины, цвета линий, заливки;

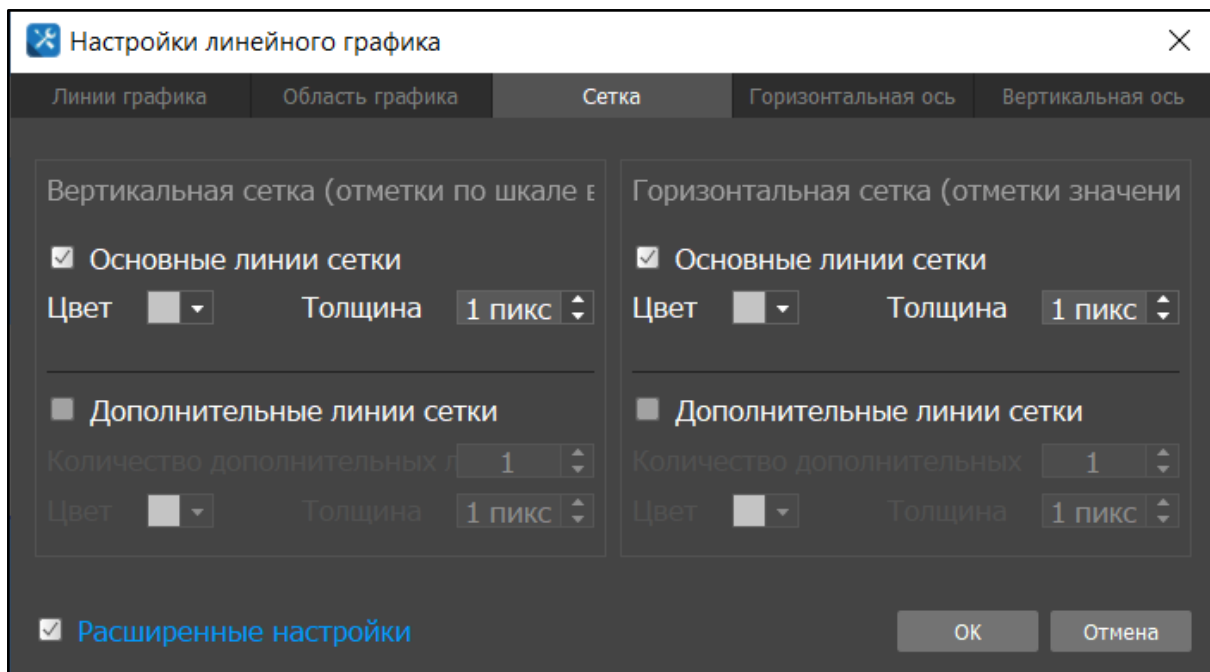


- **Область графика:**
 - **Фон** - настройка фона (цвета фона, на котором отображается график, настройка цвета под осями);
 - **Рамка** - настройка рамки, вокруг графика (показывать рамку или нет, ее цвет, толщина);

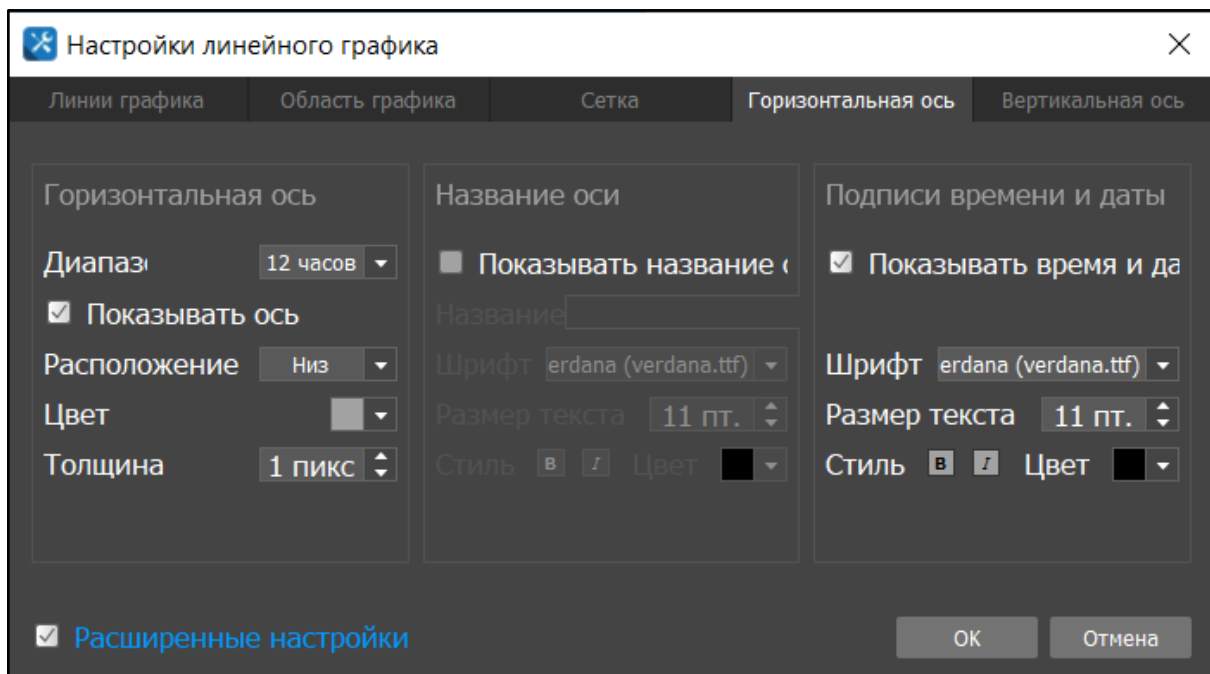
- **Управление жестами** - настройка возможности пролистывать график, приближать/отдалять график;
- **Всплывающее окно значений** - настройка всплывающего окна значений. Всплывающее окно значений и вертикальная ось появляются при длительном удержании левой кнопки мыши, курсор должен быть на графике. Данный функционал позволяет определить значение тега на графике в определенный момент времени.



- **Настройка сетки** - вертикальной и горизонтальной сетки на графике (показывать сетку или нет, ее цвет и толщина, показывать или нет дополнительные линии сетки, сколько дополнительных линий показывать между основными, их толщина и цвет);



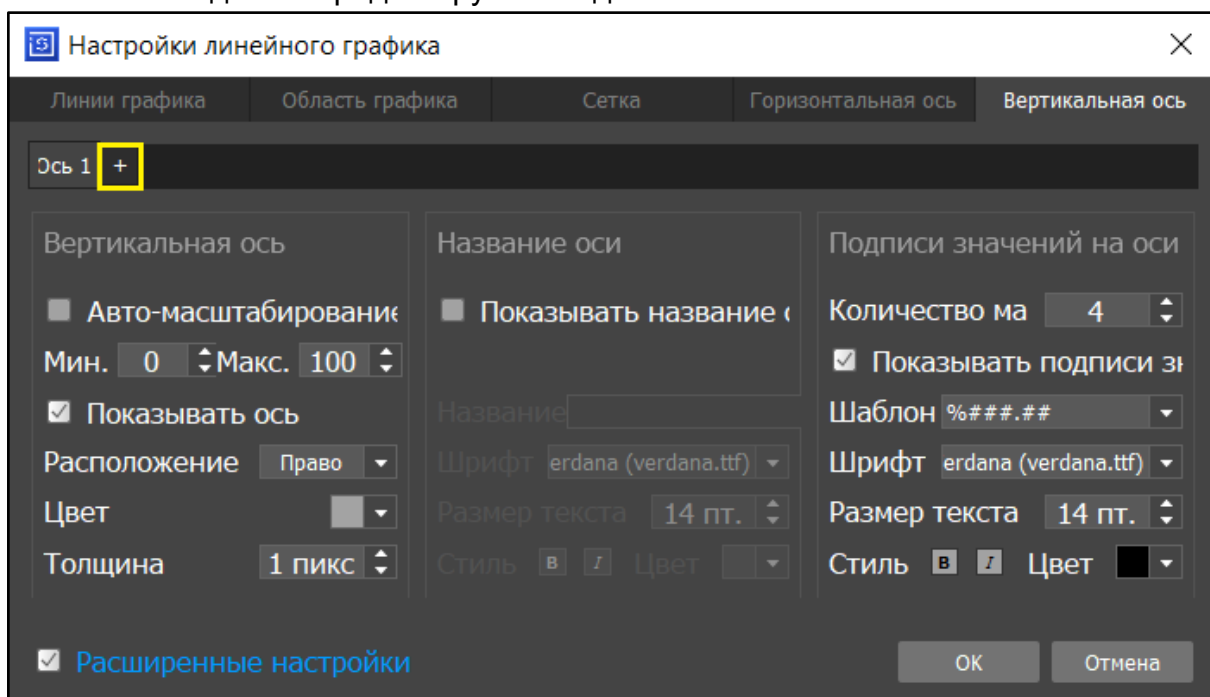
- **Горизонтальная ось** - настройка горизонтальной оси:
 - Горизонтальная ось:
 - **Диапазон** - временной диапазон, отображаемый на графике;
 - **Показывать ось** - показывать горизонтальную ось или нет;
 - **Расположение** - расположение горизонтальной оси - сверху или снизу;
 - **Цвет** - цвет линии горизонтальной оси;
 - **Толщина** - толщина линии горизонтальной оси.
 - Название оси:
 - **Показывать название оси** - показывать или нет название оси;
 - **Название** - название горизонтальной оси;
 - **Шрифт** - шрифт названия горизонтальной оси;
 - **Размер текста** - размер текста названия горизонтальной оси;
 - **Стиль** - стиль текста горизонтальной оси (утолщенный, курсив);
 - **Цвет** - цвет текста горизонтальной оси;
 - Подписи времени и даты:
 - **Показывать времени и даты** - показывать время и дату на горизонтальной оси или нет;
 - **Шрифт** - шрифт текста отображаемых даты и времени;
 - **Размер текста** - размер текста отображаемых даты и времени;
 - **Стиль** - стиль текста отображаемых даты и времени (утолщенный, курсив);
 - **Цвет** - цвет текста отображаемых даты и времени.



- **Вертикальная ось** - настройка горизонтальной оси:
 - В проекте может быть использовано несколько вертикальных осей. Использование нескольких осей позволяет увеличить наглядность графиков для тегов, значение которых находится в разных диапазонах.

Для того, чтобы добавить дополнительную ось необходимо нажать на **+** в верхней части окна.

Свойство каждой оси редактируется отдельно.



- Вертикальная ось:

- **Автомасштабирование** - настройка диапазонов отображения происходит автоматически, в зависимости от значений тега;
- **Мин. и Макс.** - настройка диапазона значений тега, в пределах которого будет отображаться график;
- **Показать ось** - показывать вертикальную ось или нет;
- **Расположение** - расположение вертикальной оси - сверху или снизу;
- **Цвет** - цвет линии вертикальной оси;
- **Толщина** - толщина линии вертикальной оси.
- Название оси:
 - **Показывать название оси** - показывать или нет название оси;
 - **Название** - название вертикальной оси;
 - **Шрифт** - шрифт названия вертикальной оси;
 - **Размер текста** - размер текста названия вертикальной оси;
 - **Стиль** - стиль текста вертикальной оси (утолщенный, курсив);
 - **Цвет** - цвет текста вертикальной оси;
- Подписи значений на оси:
 - **Количество меток** - количество интервалов значений тега, на которые будет разделен диапазон значений тега;
 - **Показывать подписи значений** - показывать подписи значений тега на вертикальной оси или нет;
 - **Шрифт** - шрифт текста отображаемых значений;
 - **Размер текста** - размер текста отображаемых значений;
 - **Стиль** - стиль текста отображаемых значений (утолщенный, курсив);
 - **Цвет** - цвет текста отображаемых значений.

Настройки линейного графика

Линии графика

Область графика

Сетка

Горизонтальная ось

Вертикальная ось

Ось 1

Ось 2

+

Вертикальная ось

☐ Авто-масштабирование

Мин. 0 Макс. 100

☒ Показывать ось

Расположение

Право

Цвет

Толщина 1 пикс

Название оси

☐ Показывать название

Название

Шрифт

erdana (verdana.ttf)

Размер текста 14 пт.

Стиль

В

И

 Цвет

Подписи значений на оси

Количество ма 4

☒ Показывать подписи значений

Шаблон %###.##

Шрифт

erdana (verdana.ttf)

Размер текста 14 пт.

Стиль

В

И

 Цвет

☒ Расширенные настройки

OK

Отмена

166

9. Использование Java Script в SCADA-сервере

9.1. Общие положения

JavaScript API iRidi - это набор функций и событий API iRidi и JavaScript. API iRidi взаимодействует с приложением SCADA, которое интерпретирует и выполняет JavaScript файлы. Функции API iRidi ссылаются на объект IR, они доступны только в оригинальном приложении, версия которого соответствует версии API.

Язык предок: **JavaScript**

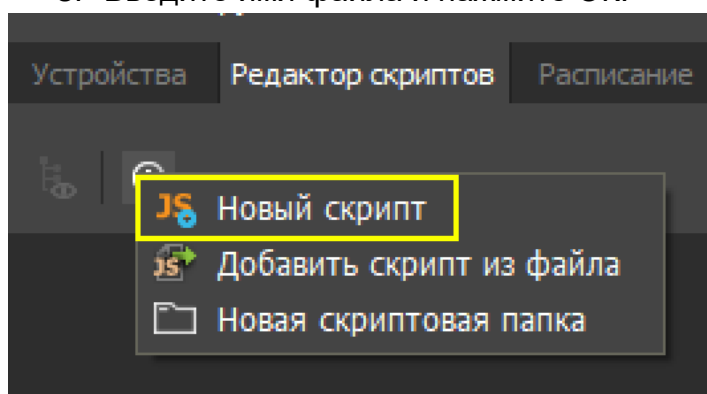
Спецификация: ECMAScript E5/E5.1

9.2. Использование API

Чтобы использовать JavaScript, нужно добавить JavaScript файл (*.js) в SCADA-проект, созданный с помощью приложения iRidi Studio. Файл хранится внутри SCADA проекта и загружается вместе с проектом. Выполнение скрипта начинается при запуске SCADA проекта.

9.2.1. Создать JavaScript файл в iRidi Studio:

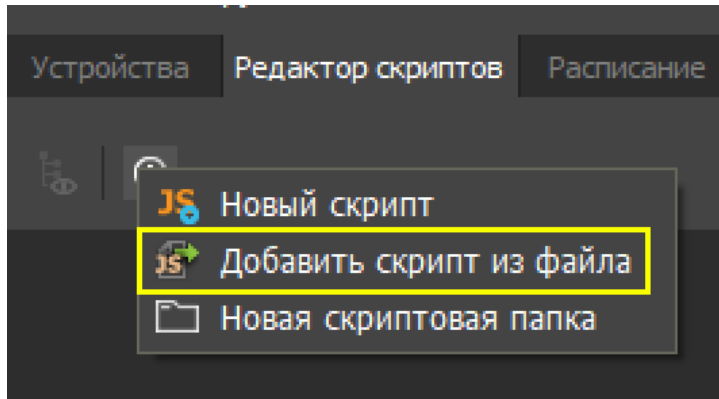
1. Откройте “Редактор скриптов”;
2. Нажмите “Новый скрипт”;
3. Введите имя файла и нажмите ОК.



Файл можно экспортировать из проекта iRidi в формате *.js и использовать повторно.

9.2.2. Подключить готовый JavaScript файл в iRidi Studio

1. Откройте iRidi Script Editor;
2. Нажмите "Добавить скрипт из файла";
3. Выберите файл с расширением *.js, он добавится в список JavaScript файлов.



9.2.3. Выполнение скрипта

JavaScript файл может содержать необходимый набор функций и событий API iRidi. Функция выполняется в случае срабатывания одного из событий API iRidi.

- JavaScript файл - это файл с расширением *.js, который прикреплен к проекту
- Событие - процесс, сброшенный в скрипты и имеющий идентификатор для отслеживания
- Слушатель - это функция, которая выполняется в случае срабатывания указанного ей события
- IR - глобальный объект, содержащий в себе все функции и константы API iRidi

Код размещается в файлах. В коде размещаются слушатели. Слушателям передаются события, объекты и функции.

9.2.4. Правила выполнения скрипта

Код выполняется асинхронно. Когда iRidi обращается к какому-то устройству, он не прерывает выполнение работы сервера, а принимает данные в отдельном потоке, как если бы вы обращались к удаленному веб-сайту из вашего браузера. Однако при выполнении большого количества операций, к примеру циклов, работа сервера прерывается и ожидает завершения выполнения операции, и только после этого продолжает выполняться.

Обычные слушатели можно объявлять несколько раз. Глобальные - только один раз.

Файлы скриптов при запуске проекта работают в одном пространстве имен - не используйте переменные и функции с одинаковыми именами в разных файлах скрипта.

9.2.5. Пространство имен

JavaScript файлы в проекте iRidi выполняются в одном пространстве имен, инициализация файлов производится последовательно, сверху вниз. Учитывайте это при выборе имен функций и определении областей видимости. Если в разных JS файлах определена переменная или функция с одинаковым идентификатором (именем), то во время выполнения будет использоваться функция, которая была размещена в памяти последняя, т.е. в файле, который находился ниже.

Это не относится к слушателям. Если в проекте определены слушатели на одинаковые события, выполняются оба.

9.3. Методы js

9.3.1. Глобальный слушатель

9.3.1.1. Метод установки глобального слушателя для подписанных тегов

`IR.SetGlobalListener(function(name, value){});`

`IR.SetGlobalListener(listener_func);`

Параметры:

- `function(name, value)()` - функция, принимающая `name` (string) и `value` (string, number, bool в зависимости от тега) изменившегося тега;

Пример:

```
var listener_func = function(name, value){  
    IR.Log("SCRIPT: name: " + name + " _ value: " + value + " _ value type: " + typeof value);  
};  
  
IR.SetGlobalListener(listener_func);
```

9.3.1.2. Метод подписки на изменение тега

`IR.SubscribeTagChange(name);`

Параметры:

- `name` - полное имя тега (string). Для SCADA тегов это "Server.Tags.name".

Пример:

```
IR.SubscribeTagChange("Server.Tags.tag 2");
```

9.3.1.3. Метод отписки на изменение тега

`IR.UnsubscribeTagChange(name);`

Параметры:

- `name` - полное имя тега (string). Для SCADA тегов это "Server.Tags.name";

Пример:

```
IR.UnsubscribeTagChange("Server.Tags.tag 2");
```

9.3.2. Логирование

Методы вывода текста в консоль.

```
IR.Log(text);
```

```
console.log(text);
```

Параметры:

- *text*- текст (string).

9.3.3. Телеграм

9.3.3.1. Метод установки слушателя на прием сообщений

```
var result = IR.SetTelegramListener(function(chat_id, message){});
```

Параметры:

- *function(chat_id, message)* - функция, принимающая chat_id (string) и message (string) собеседника.

Возвращаемое значение result (bool) - успешность установки.

Пример:

```
IR.SetGlobalListener(listener_func);

var telegram_func = function(chat_id, text){
    IR.Log("SCRIPT: chat_id: " + chat_id + " _ text: " + text);
}
var telegram_result = IR.SetTelegramListener(telegram_func);
```

9.3.3.2. Метод отписки приема сообщений

```
var result = IR.UnsetTelegramListener();
```

Параметры:

Без параметров.

Возвращаемое значение result (bool) - успешность отписки.

9.3.3.3. Метод отправки сообщения

```
IR.SendTelegramMessage(chat_id, message);
```

Параметры:

- *chat_id (Number)* - идентификатор чата из базы;
- *message (String)* - сообщение

Пример:

```
IR.SendTelegramMessage(1, "Сообщение принято!");
```

9.3.4. Установка и получение переменных

Применение: Пока действует только для SCADA-тегов!!!

9.3.4.1. Метод установки значения переменной

```
var result = IR.SetVariable(name, value);
```

Параметры:

- *name* (*string*) - полное имя переменной. Для скада тегов это "Server.Tags.name";
- *value* (*string*, *number*, *bool* в зависимости от тега) - значение.

Возвращаемое значение *result* (*bool*) - успешность установки.

Пример:

```
var result = IR.SetVariable("Server.Tags.tag 2", 1000);
```

9.3.4.2. Метод получения значения переменной

```
var value = IR.GetVariable(name);
```

Параметры:

- *name* (*string*) - полное имя переменной. Для скада тегов это "Server.Tags.name"

Возвращаемое значение *value* (*string*, *number*, *bool* в зависимости от тега) - значение

Пример:

```
var value = IR.GetVariable("Server.Tags.tag 2");  
IR.Log("SCRIPT: value: " + value + " _ value type: " + typeof value);
```

9.3.5. Таймеры

9.3.5.1. Циклический таймер

```
var id = IR.SetInterval(1000, function() {  
    IR.Log("Привет");  
});
```

Параметры:

- *id (string)* - идентификатор таймера;
- *1000 (number)* - время, через которое повторять функцию
- *function (function)* - обработчик функция для срабатывания таймера

9.3.5.2. Одноразовый таймер

```
var id = IR.SetTimeout(1000, function() {  
    IR.Log("Привет");  
});
```

Параметры:

- *id (number)* - идентификатор таймера;
- *1000 (number)* - время, через которое повторять функцию;
- *function (function)* - обработчик функция для срабатывания таймера

9.3.5.3. Удалить таймер

```
var result = IR.ClearInterval(id);
```

Параметры:

- *id (number)* - идентификатор таймера;
- *result (boolean)* - успешность удаления.

9.3.6. Журнал событий

9.3.6.1. Отправка события в журнал

IR.SendCustomEvent(*priority*, *text*);

Параметры:

- *priority (number)* - приоритет аларма;
- *text (string)* - текст сообщения.

Строка сообщения не должна быть пустой.

9.3.6.2. Получение истории событий

```
function Run()
{
    var offset = 0;
    var limit = -1;

    var events_history = IR.GetEventsHistory(IR.HISTORY_EVENT_TYPE_ALL, (start.getTime() / 1000), (end.getTime() / 1000), offset, limit);
    if(events_history instanceof Array)
        IR.Log(JSON.stringify(events_history));
}

if(IR.IsDbConnected())
    Run();

IR.SetDbConnectedListener(Run);
```

где,

IR.HISTORY_EVENT_TYPE_ALL (*number*) тип события

(start.getTime() / 1000) - время начала (*number*), timestamp в секундах, поэтому делим на 1000

(end.getTime() / 1000) - время конца (*number*), timestamp в секундах, поэтому делим на 1000

offset - количество (*number*) элементов, которое нужно пропустить (необязательный параметр)

limit - количество (*number*) элементов, которое нужно получить (необязательный параметр)

events_history - массив объектов

```
[
{
    "id": 1,
    "type": 3,
    "message": "Scada server started",
    "date_created": 1709029828,
    "event_priority": 0,
    "user_id": 0,
    "alarm_id": 0,
    "alarm_group_id": 0,
    "alarm_state": 0,
    "action_group_id": 0,
    "alarm_image_path": "",
    "tag_id": 0,
```

```

    "bound_name": "",
    "bound_index": 0,
    "zones": []
  },
  {
    "id": 3,
    "type": 5,
    "message": "[analog_2_setpoints] ALARM HI! hi='40.000000'; value='40.000000'",
    "date_created": 1709029828,
    "event_priority": 0,
    "user_id": 0,
    "alarm_id": 1,
    "alarm_group_id": 1,
    "alarm_state": 1,
    "action_group_id": 1,
    "alarm_image_path": "",
    "tag_id": 1,
    "bound_name": "hi",
    "bound_index": 2,
    "zones": []
  },
]

```

9.3.6.3. Получение истории событий

```

IR.HISTORY_EVENT_TYPE_ALL
IR.HISTORY_EVENT_TYPE_USER_LOGIN
IR.HISTORY_EVENT_TYPE_USER_LOGOUT
IR.HISTORY_EVENT_TYPE_PROJECT_STARTED
IR.HISTORY_EVENT_TYPE_CUSTOM_EVENT
IR.HISTORY_EVENT_TYPE_ALARM
IR.HISTORY_EVENT_TYPE_ALARM_CONFIRM
IR.HISTORY_EVENT_TYPE_CLEAR_ALARMS
IR.HISTORY_EVENT_TYPE_TAG_ACCESS

```

9.3.7. Получение истории тегов

9.3.7.1. Метод получения

```
if (IR.IsDbConnected())
    Run();

IR.GetDbConnectedListener(Run);

var start = new Date(2024, 1, 21, 12, 0, 0);
var end = new Date(2024, 1, 22, 14, 12, 0);

var offset = 0;
var limit = -1;

var array = IR.GetTagHistory("Tag", (start.getTime() / 1000), (end.getTime() / 1000), IR.DUTY_CYCLE_SECONDS_1, offset, limit);
if (array instanceof Array)
    IR.Log(JSON.stringify(array));
```

где,

Tag (string) - имя тега или id тега (*number*)

(start.getTime() / 1000) - время начала (*number*), timestamp в секундах, поэтому делим на 1000

(end.getTime() / 1000) - время конца (*number*), timestamp в секундах, поэтому делим на 1000

IR.DUTY_CYCLE_SECONDS_1 - скважность (*number*), константа

offset - количество (*number*) элементов, которое нужно пропустить (необязательный параметр)

limit - количество (*number*) элементов, которое нужно получить (необязательный параметр)

array - массив объектов, состоящий из `date` ((string) Timestamp) и `value` ((*number*) или (string))

```
[
  {
    "date": 1708592377,
    "value": 410
  },
  {
    "date": 1708592378,
    "value": 70
  }
]
```

9.3.7.2. Типы скважности (константы)

IR.DUTY_CYCLE_DAY

IR.DUTY_CYCLE_HOURS_12

IR.DUTY_CYCLE_HOURS_6

IR.DUTY_CYCLE_HOURS_3

IR.DUTY_CYCLE_HOURS_1

IR.DUTY_CYCLE_MINUTES_30

IR.DUTY_CYCLE_MINUTES_15

IR.DUTY_CYCLE_MINUTES_5

IR.DUTY_CYCLE_MINUTES_2
IR.DUTY_CYCLE_MINUTES_1
IR.DUTY_CYCLE_SECONDS_30
IR.DUTY_CYCLE_SECONDS_15
IR.DUTY_CYCLE_SECONDS_5
IR.DUTY_CYCLE_SECONDS_2
IR.DUTY_CYCLE_SECONDS_1

9.3.8. Подключение к DataBase

9.3.8.1. Слушатель подключения к базе данных

```
IR.SetDbConnectedListener(function(){  
    var start = new Date(2024, 1, 21, 12, 0, 0);  
    var end = new Date(2024, 1, 22, 14, 12, 0);  
  
    var result = IR.GetTagHistory("Tag", (start.getTime() / 1000), (end.getTime() / 1000), IR.DUTY_CYCLE_SECONDS_1);  
    if(result instanceof Array)  
        IR.Log(JSON.stringify(result));  
});
```

где,

function (function) - функция-обработчик при подключении к базе данных.

9.3.8.2. Метод проверки подключения к базе

```
var result = IR.IsDbConnected();
```

Параметры:

- result (Bool) - результат проверки подключения к базе данных (true/false).

9.3.9. Алармы

9.3.9.1. Получение истории алармов

```
function Run()
{
    var offset = 0;
    var limit = -1;

    var alarms = IR.GetAlarms((start.getTime() / 1000), (end.getTime() / 1000), offset, limit);
    if(alarms instanceof Array)
        IR.Log(JSON.stringify(alarms));
}

if(IR.IsDbConnected())
    Run();

IR.SetDbConnectedListener(Run);
```

где,

2. ```(start.getTime() / 1000)``` - время начала {Number}, timestamp в секундах, поэтому делим на 1000
3. ```(end.getTime() / 1000)``` - время конца {Number}, timestamp в секундах, поэтому делим на 1000
5. ```offset``` - количество {Number} элементов, которое нужно пропустить (необязательный параметр)
6. ```limit``` - количество {Number} элементов, которое нужно получить (необязательный параметр)
7. ```alarms``` - массив объектов

...

```
[
{
    "id": 1,
    "alarm_id": 1,
    "alarm_group_id": 1,
    "message": "[analog_2_setpoints] ALARM HI! hi='40.000000'; value='40.000000'",
    "creation_date": 1709026423,
    "is_confirmation_required": true,
    "confirm_message": "",
    "confirmation_user_id": 0,
    "date_confirm": 0,
    "state": 1,
    "alarm_priority": 0,
    "action_group_id": 1,
    "confirmation_type": 0,
    "alarm_image_path": "",
    "tag_id": 1,
    "bound_name": "hi",
    "bound_index": 2,
```

```
    "zones": []  
  }  
]
```

9.3.9.2. Типы состояний аларма (константы)

Поле "state": 1

```
IR.ALARM_STATE_NORMAL  
IR.ALARM_STATE_ACCIDENT  
IR.ALARM_STATE_WARNING
```

9.3.9.3. Типы подтверждения аларма (константы)

Поле "confirmation_type": 0

```
IR.ALARM_CONFIRMATION_NO_COMMENT  
IR.ALARM_CONFIRMATION_COMMENT_NOT_REQUIRED  
IR.ALARM_CONFIRMATION_COMMENT_REQUIRED
```

9.3.9.4. Создание кастомного аларма с записью в журнал

Метод позволит пользователям генерировать в системе алармы с записью сообщений в журналы аварий и событий, условия срабатывания которых не привязаны к тега, зонам, границам и группам алармов.

IR.SendCustomAlarm(type, priority, message);

Параметры:

- *type (number)* - определяет тип записанного сообщения;
- *priority (number)* - приоритет аларма;
- *message (string)* - текст сообщения.

Строка сообщения не должна быть пустой.

Пример:

```

        this.name = name;
        this.warningPriority = warningPriority;
        this.alarmPriority = alarmPriority;
        this.warning_list = [];
        this.accident_list = [];

        // попробуем восстановить, но если в момент создания класса ЕД недоступна, методично вызывать повторно
        this.RestoreEvents();
    }

    // Метод работает только при условии уникальности warningPriority и alarmPriority.
    // иные ситуации будут вызывать ошибки
    AlarmCreator.prototype.RestoreEvents = function() {
        var self = this;
        IR.GetCustomAlarms().forEach(function(alarm) {
            if (alarm.type == 2 && alarm.priority == self.warningPriority) {
                self.warning_list.push(alarm);
            }
            else if (alarm.type == 1 && alarm.priority == self.alarmPriority) {
                self.accident_list.push(alarm);
            }
        });
    }

    AlarmCreator.prototype.SetWarning = function (mess) {
        this.warning_list.push(IR.SendCustomAlarm(IR.ALARM_STATE_WARNING, this.warningPriority, mess));
    }

    AlarmCreator.prototype.SetAccident = function (mess) {
        this.accident_list.push(IR.SendCustomAlarm(IR.ALARM_STATE_ACCIDENT, this.alarmPriority, mess));
    }

    AlarmCreator.prototype.ResetWarnings = function (mess) {
        this.warning_list.forEach(function (alarm) {IR.ResetCustomAlarm(alarm.id, mess)});
        this.warning_list = [];
    }

    AlarmCreator.prototype.ResetAccident = function (mess) {
        this.accident_list.forEach(function (alarm) {IR.ResetCustomAlarm(alarm.id, mess)});
        this.accident_list = [];
    }

    AlarmCreator.prototype.ResetAll = function (mess) {
        this.ResetWarnings(mess);
        this.ResetAccident(mess);
    }
}

```

9.3.9.5. Сброс кастомных алармов с записью в журнал

JS-метод предназначен сброса кастомных алармов (аварий и предупреждений), т.к. кастомные алармы не имеют параметров нормализации для автоматического сброса.

Для работы метода необходимо знать id сбрасываемого кастомного аларма. Поскольку id формируется динамически в момент создания аларма необходима возможность получать его значение из ответа iR.SendCustomAlarm для последующего использования в методе сброса.

IR.ResetCustomAlarm(*alarm ID*, *message*);

Параметры:

- *alarm ID (number)* - ID сбрасываемого аларма;
- *message (string)* - текст сообщения нормализации.

Пример:

```

IR.SubscribeTagChange("Server.Tags.Alarm Tag");
if (IR.GetVariable(AlarmTag) == 'D00')
{
    iR.ResetCustomAlarm(123 , "Ошибка устранена");
};

```

При получении тегом Alarm Tag значения “D00” пользовательский аларм с id 123 сбрасывается, в журнал алармов записывается сообщение “Ошибка устранена” с приоритетом события 0.

9.3.10. Диспетчеризация склада клиента

9.3.10.1. Как это работает

В приложениях **iRidi SCADA Client** и **i3 pro** добавлены параметры запуска:

- `/new_app_instance` - идентификация открытия нового экземпляра клиента
- "Ваше имя клиента" - имя клиента, которое служит идентификатором запускаемого клиента. Оно должно быть уникально для запуска нового клиента. Оно содержится в шапке имени клиента, в названии папки в документах и подмешивается с `hwid`, чтобы он был уникальный для каждого клиента.
- "Параметры" - параметры для клиента
- `i3pro.exe /new_app_instance "Client1" "ip1=192.168.0.1;login=a;pass=b;"` - пример запуска

9.3.10.2. Скриптовые методы

Добавлены скриптовые методы:

- `IR.LauchNewAppInstance("Ваше имя клиента", "Параметры");` - запуск нового склада клиента. **"Параметры"** - необязательны, но для автоматического развертывания вашего проекта необходимы. Если их не указать, клиент откроется, но придется самому авторизовываться и качать проект.
- `IR.CloseAppInstance("Ваше имя клиента");` - закрытие склада клиента
- `IR.LauchI3ProAppInstance("Ваше имя клиента", "Параметры", "Кастомный путь до i3 pro")` - запуск нового **i3 pro** клиента. **"Параметры"** - этот параметр не обязателен, но для автоматического развертывания вашего проекта необходимы. Если их не указать, клиент откроется, но придется самому авторизовываться и качать проект. **"Кастомный путь до i3 pro"** - этот параметр не обязателен. Если его не указать, то **i3 pro** будет брать из папки **iRidium/iRidium64.exe** в ваших программных файлах.
- `IR.CloseI3ProAppInstance("Ваше имя клиента");` - закрытие **i3 pro** клиента

9.3.10.3. Параметры

"Параметры" прописываются как строка `"ключ=значение;ключ=значение;"`.
Доступные параметры:

- **ip1** - главный адрес для подключения к серверу. Доступен в **iRidi SCADA Client** и **i3 pro**

- **ip2** - резервный адрес для подключения к серверу. Доступен в **iRidi SCADA Client**
- **login** - логин подключения. Доступен в **iRidi SCADA Client** и **i3 pro**
- **pass** - пароль для подключения. Доступен в **iRidi SCADA Client** и **i3 pro**
- **project_name** - имя проекта, который хотите скачать. Доступен в **iRidi SCADA Client** и **i3 pro**. Для оффлайн режима в **i3 pro** необходимо указывать имя с расширением, пример **Project1.irpz**
- **object_name** - имя объекта, в котором находится объект. Доступен в **i3 pro**
- **is_offline_auth** - флаг авторизации, необходимый для понимания, как вы хотите авторизоваться в **i3 pro**
- **no_frame** - флаг запуска клиента с рамкой или без. Доступен в **iRidi SCADA Client** и **i3 pro**
- **X** - координата X для запуска клиента в определенной позиции. Доступен в **iRidi SCADA Client** и **i3 pro**
- **Y** - координата Y для запуска клиента в определенной позиции. Доступен в **iRidi SCADA Client** и **i3 pro**
- **position** - запуск окна приложения в определенной позиции. Доступен в **iRidi SCADA Client** и **i3 pro**. Позиции:

1 - Слева вверху

2 - Вверху по центру

3 - Справа вверху

4 - Слева по центру

5 - По центру

6 - Справа по центру

7 - Слева внизу

8 - Внизу по центру

9 - Справа внизу

X и **Y** будет приоритетней чем **position**

9.3.10.4. Пример всех параметров:

```
"ip1=192.168.0.1;ip2=192.168.0.2;login=a;pass=b;project_name=project;object_name=object;is_offline_auth=0;no_frame=1;X=0;Y=0;position=5;"
```

10. Использование Java Script в SCADA-клиенте

10.1. Общие положения

JavaScript API iRidi - это набор функций и событий API iRidi и JavaScript 1.5, предназначенных для управления визуальной частью и драйверами в проекте SCADA iRidi. API iRidi взаимодействует с приложением SCADA клиент, которое интерпретирует и выполняет JavaScript файлы. Функции API iRidi ссылаются на объект IR, они доступны только в оригинальном приложении, версия которого соответствует версии API.

Язык предок: **JavaScript**

Версия языка: **1.5**

Спецификация: ECMAScript Edition 3

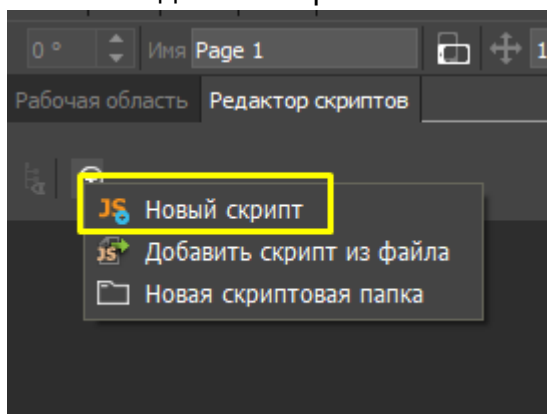
Все доступные объекты API подробно описаны в справочнике по программному интерфейсу.

10.2. Использование API

Чтобы использовать JavaScript, нужно добавить JavaScript файл (*.js) в панельный SCADA проект, созданный с помощью приложения iRidi Studio. Файл хранится внутри панельного SCADA проекта и загружается вместе с проектом в приложение SCADA клиент. Выполнение скрипта начинается при запуске проекта в приложении SCADA клиент.

10.2.1. Создать JavaScript файл в iRidi Studio:

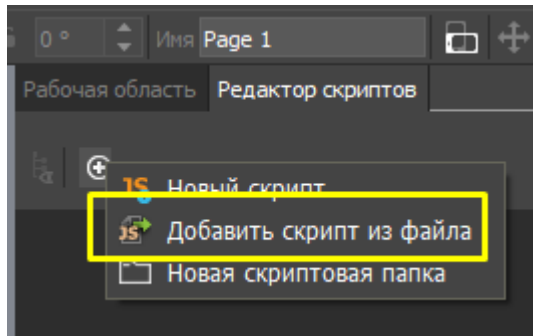
1. Откройте iRidi Script Editor;
2. Нажмите "New Script";
3. Введите имя файла и нажмите ОК.



Файл можно экспортировать из проекта iRidi в формате *.js и использовать повторно.

10.2.2. Подключить готовый JavaScript файл в iRidi Studio

1. Откройте iRidi Script Editor;
2. Нажмите "Add Script from file";
3. Выберите файл с расширением *.js - он добавится в список JavaScript файлов.



10.2.3. Выполнение скрипта

JavaScript файл может содержать необходимый набор функций и событий API iRidi. Функция выполняется в случае срабатывания одного из событий API iRidi. Таким событием может быть запуск приложения, нажатие на графический элемент, получение данных от оборудования и др.

- JavaScript файл - это файл с расширением *.js, который прикреплен к проекту
- Событие - процесс, сброшенный в скрипты и имеющий идентификатор для отслеживания
- Слушатель - это функция, которая выполняется в случае срабатывания указанного ей события
- IR - глобальный объект, содержащий в себе все функции и константы API iRidi

Код размещается в файлах. В коде размещаются слушатели. Слушателям передаются события, объекты и функции.

10.2.4. Правила выполнения скрипта

Код выполняется асинхронно. Когда iRidi обращается к какому-то устройству, он не прерывает выполнение работы интерфейса, а принимает данные в отдельном потоке, как если бы вы обращались к удаленному веб-сайту из вашего браузера.

Однако при выполнении большого количества операций, к примеру циклов, весь GUI интерфейс прерывается и ожидает завершения выполнения операции, и только после этого продолжает выполняться.

Обычные слушатели можно объявлять несколько раз. Глобальные - только один раз.

Файлы скриптов при запуске проекта работают в одном пространстве имен - не используйте переменные и функции с одинаковыми именами в разных файлах скрипта.

Не удаляйте графический элемент до того, как он завершил выполнение связанных с ним операций, это приведет к ошибке выполнения команд.

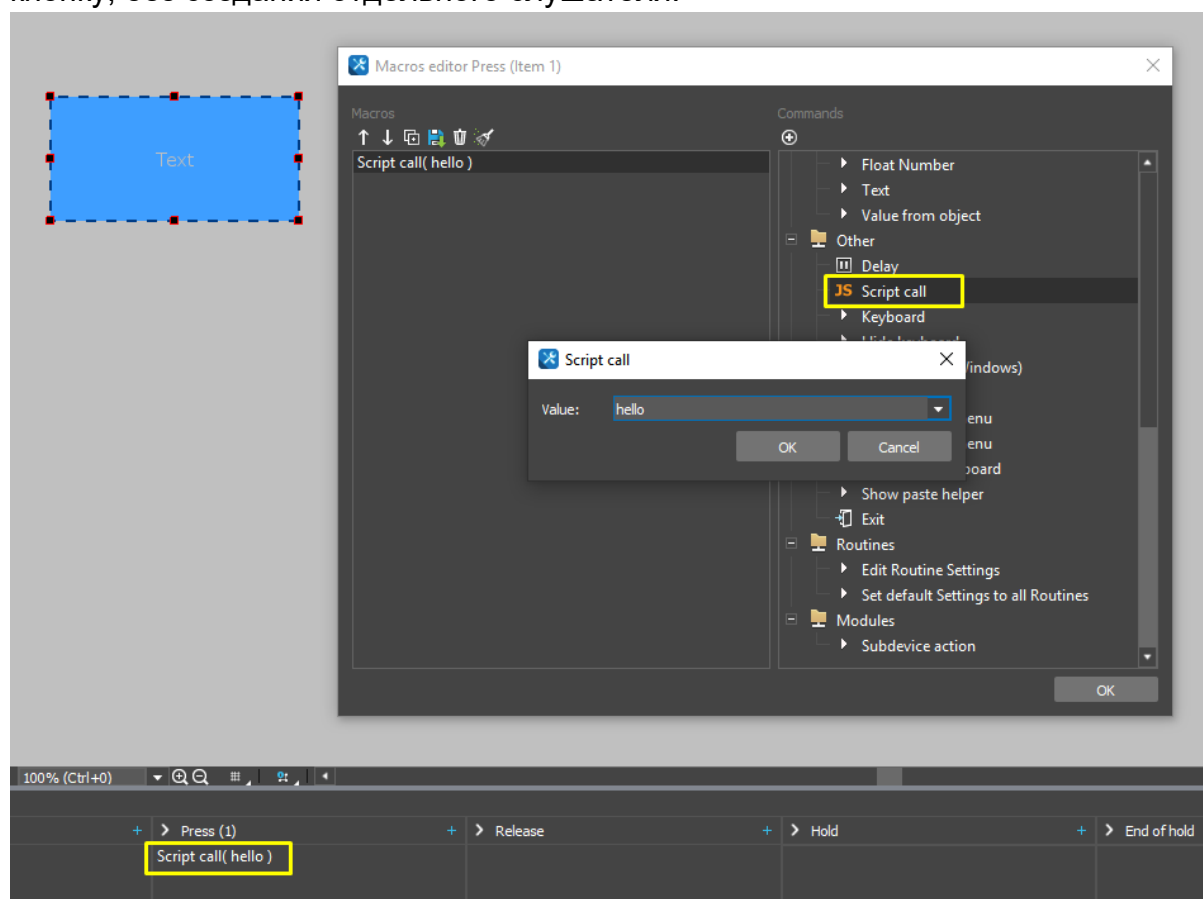
10.2.5. Пространство имен

JavaScript файлы в проекте iRidi выполняются в одном пространстве имен, инициализация файлов производится последовательно, сверху вниз. Учитывайте это при выборе имен функций и определении областей видимости. Если в разных JS файлах определена переменная или функция с одинаковым идентификатором (именем), то во время выполнения будет использоваться функция, которая была размещена в памяти последняя, т.е. в файле, который находился ниже.

Это не относится к слушателям. Если в проекте определены слушатели на одинаковые события, выполняются оба.

10.3. Команда ScriptCall()

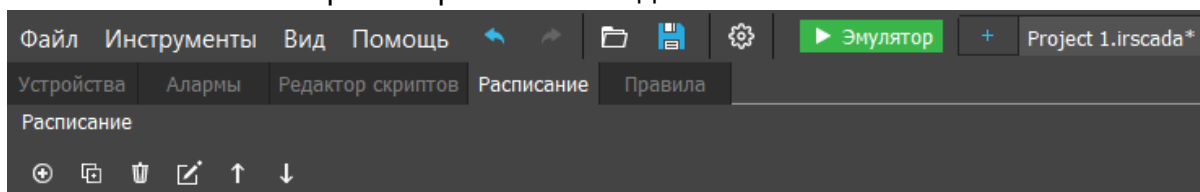
Скрипт можно вызвать командой ScriptCall(), привязанной к графическому элементу. Команде можно передать только указатель на функцию без входных параметров, но она позволит выполнить скрипт в любой момент, по нажатию на кнопку, без создания отдельного слушателя:



11. Расписание

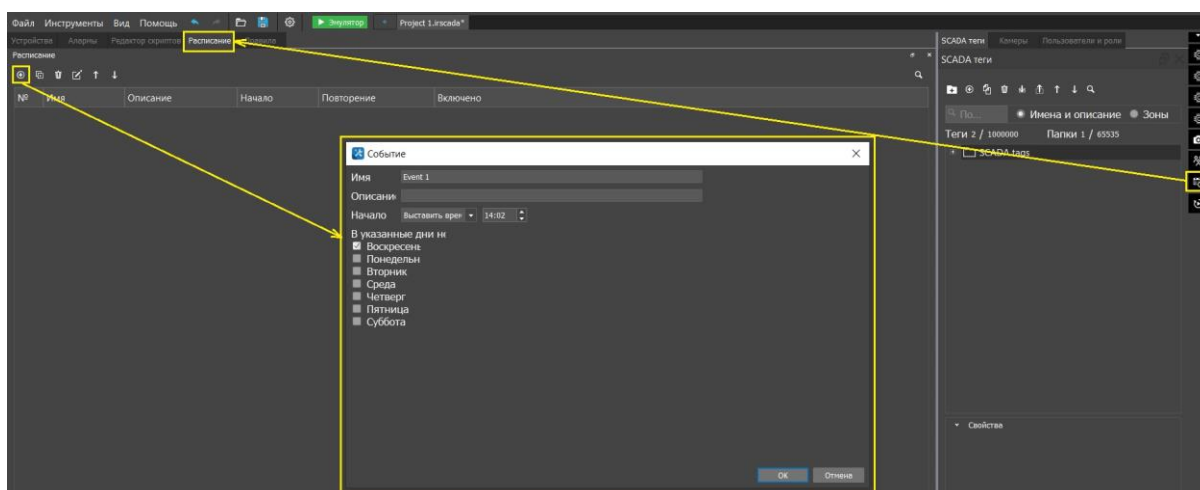
11.1. Описание

Расписание в серверной логике - это планировщик с событиями, доступен в панели Расписание при настройке логики для iRidi Server и SCADA Server.



Свойства событий:

- события могут срабатывать однократно и с заданной периодичностью;
- событие может начаться (Start) и закончиться (Finish), начало события может активировать сценарии сервера, конец - завершить;
- событие может находиться в процессе выполнения (In process) или не выполняться прямо сейчас (Not in process). Состояние события можно использовать, как исключающее условие (BUT ONLY IF) в сценариях сервера;
- время срабатывания события можно менять в процессе работы сервера;
- периодичность срабатывания события нельзя менять в процессе работы сервера;
- событие можно включить и выключить в процессе работы сервера.



Для того чтобы открыть вкладку Расписание, если она закрыта, необходимо выбрать на панели справа иконку с календарем.

Для того чтобы создать новое расписание, необходимо нажать на иконку с "+" на панели инструментов раздела "Расписание".

11.2. Типы расписаний

SCADA Server поддерживает только простой тип расписаний.

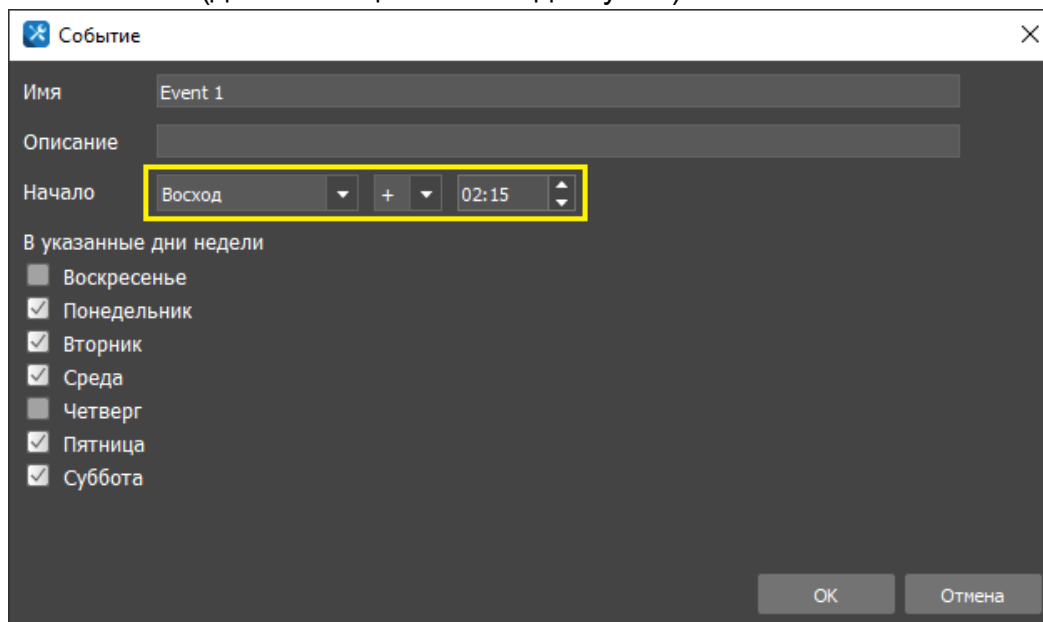
Простое расписание имеет меньше параметров настроек, но они дают возможность работать с редактируемыми сценариями из запущенного SCADA-клиента.

В простых расписаниях возможно настроить только дату и время начала события. Для того чтобы привязать действие к концу события, необходимо создать отдельное событие (Event).

Подробное описание настройки редактируемых сценариев описано здесь.

11.3. Настройки событий расписания

- **Имя** - название события.
- **Описание** - краткое описание (не обязательно).
- **Начало** - время начала события.
 - Выставить время - время активации события.
 - Восход, Закат - время восхода или заката солнца. Для использования нужно настроить астротаймер в настройках проекта (данная опция пока недоступна).

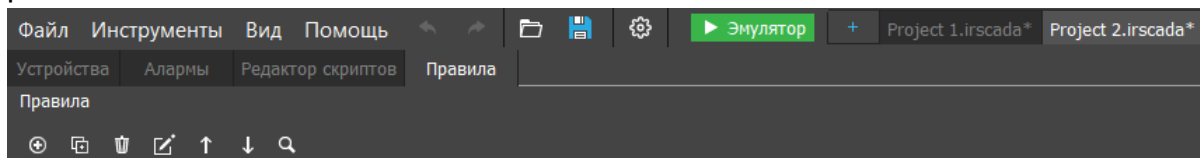


- **Повторение** - периодичность повторения события. Для упрощенных событий устанавливается повторение по дням недели, т.е. срабатывание события будет происходить каждую неделю в установленное время отмеченных дней.

12. Правила

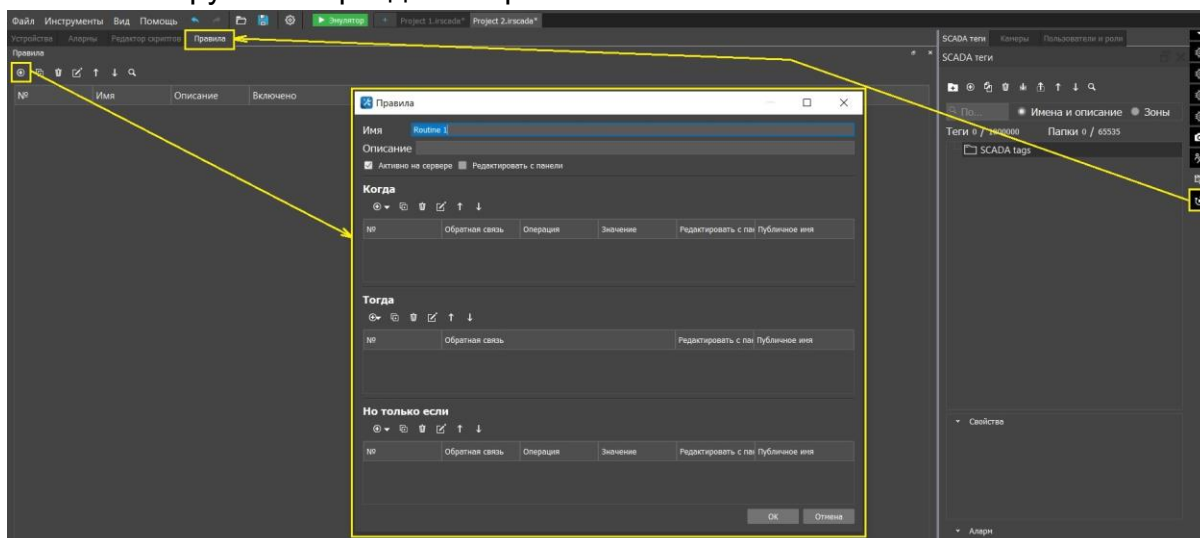
12.1. Описание

Правило автоматически запускается при возникновении определенного события: при получении данных от оборудования или срабатывании событий расписания



Для того чтобы открыть вкладку Правила, если она закрыта, необходимо выбрать на панели справа иконку с круглой стрелкой.

Для того чтобы создать новое правило, необходимо нажать на иконку с “+” на панели инструментов раздела “Правила”.

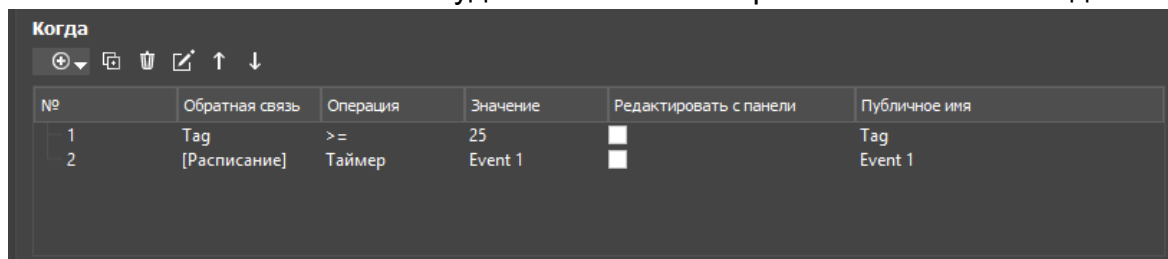


Окно правил содержит следующие параметры настройки:

12.2. События срабатывания, действия при срабатывании, исключаяющие условия

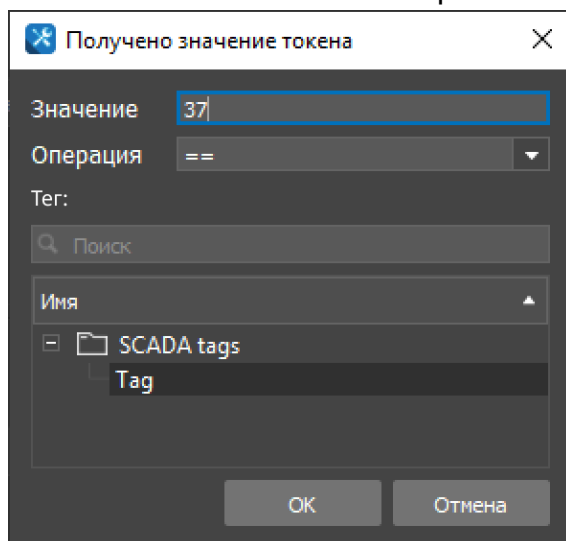
Правила работают на основе событийной логики:

- **Событие срабатывания (КОГДА)** - событие, которое вызывает выполнение команд, указанных в ТОГДА, но только если при этом выполняются условия, указанные в НО ТОЛЬКО ЕСЛИ. Для запуска правила достаточно наступления одного из событий. Правило без событий не будет активировано никогда.



№	Обратная связь	Операция	Значение	Редактировать с панели	Публичное имя
1	Tag	>=	25	☐	Tag
2	[Расписание]	Таймер	Event 1	☐	Event 1

- **Тег** - scada-тег проекта. Срабатывает однократно, в момент изменения переменной от текущего значения к указанному в условии. Следующее присвоение того же значения переменной не вызовет срабатывания события (нужно сначала сбросить значение, затем присвоить заново).



Получено значение токена

Значение: 37

Операция: ==

Тег:

Поиск

Имя

- SCADA tags
 - Tag

OK Отмена

- **Оператор** - операторы сравнения, активирующий правило при выполнении условия между текущим значением тега и константой.
- **Значение** - константа, с которой сравнивается значение тега для запуска правила.
- **Событие расписания** - активирует правило при запуске простого события.
- **Редактировать с панели** - редактирование события в панельном проекте.
- **Публичное имя** - имя события (для отображения в панельном проекте).

- **Действия при срабатывании (ТОГДА)** - все команды, настроенные в ТОГДА будут выполнены друг за другом, начиная с верхнего элемента в списке.

Тогда			
+ ↕ ✖ ✎ ↑ ↓			
№	Обратная связь	Редактировать с панели	Публичное имя
1	Значение(7 > Tag)	☒	Tag

- **Значение** - записывает в scada-тег выбранное значение
 - **Тег** - scada-тег проекта, в который при активации правила будет записано указанное значение.
 - **Значение** - константа, которая будет записана в тег при активации правила.
- **Редактировать с панели** - редактирование действия в панельном проекте.
- **Публичное имя** - имя действия (для отображения в панельном проекте).
- **Исключающие условия (НО ТОЛЬКО ЕСЛИ)** - если условия не выполняются, команды из ТОГДА не сработают. Для активации правила необходимо выполнение всех условий.

Но только если					
+ ↕ ✖ ✎ ↑ ↓					
№	Обратная связь	Операция	Значение	Редактировать с панели	Публичное имя
1	Tag	==	-6	☒	Tag

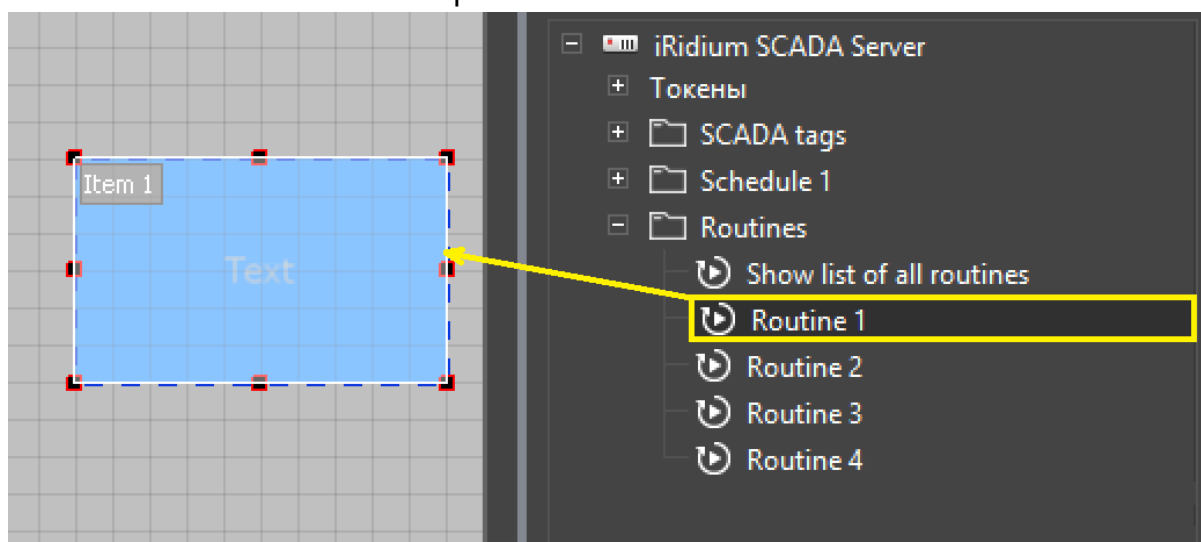
- **Тег** - scada-тег проекта. Значение которого должно удовлетворять условию, задаваемому оператором и константой.

- **Оператор** - операторы сравнения:
 - ==, !=, <, >, <=, >= - сравнение значения тега с указанным значением константы.
- **Значение** - константа для сравнения со значением тега.
- **Редактировать с панели** - редактирование действия в панельном проекте.
- **Публичное имя** - имя действия (для отображения в панельном проекте).

12.3. Настройка редактируемых правил

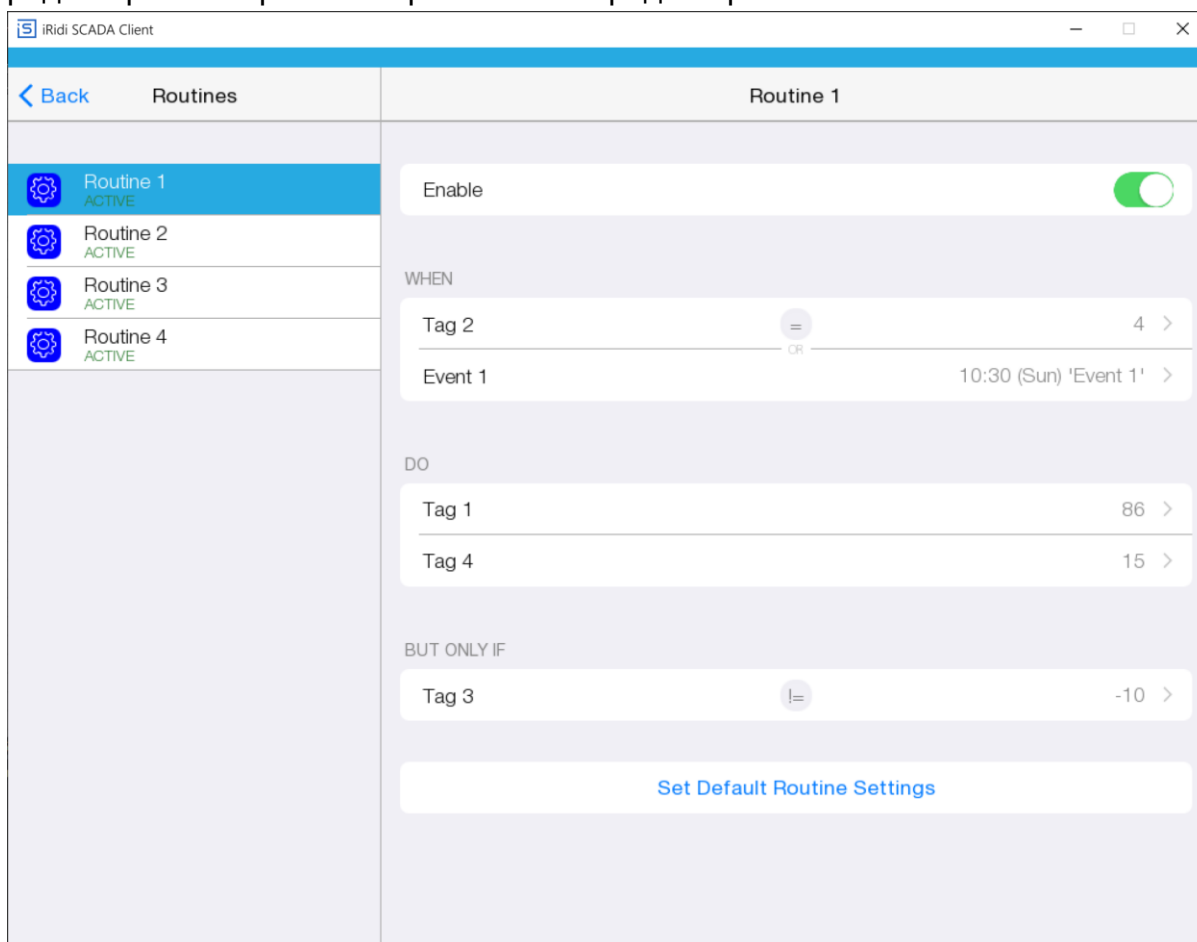
Редактируемые правила - это созданные в iRidi Studio правила, параметры которых можно редактировать конечному пользователю в запущенном панельном проекте.

1. Создайте в проекте SCADA нужное количество правил и задайте первоначальные параметры настройки (которые потом мы сможем редактировать из панельного проекта).
2. Синхронизируйте проект SCADA с панельным проектом SCADA с помощью драйвера iRidi SCADA Server.
3. Перетащите команду вызова интерфейса редактирования правила на элемент в панельном проекте.



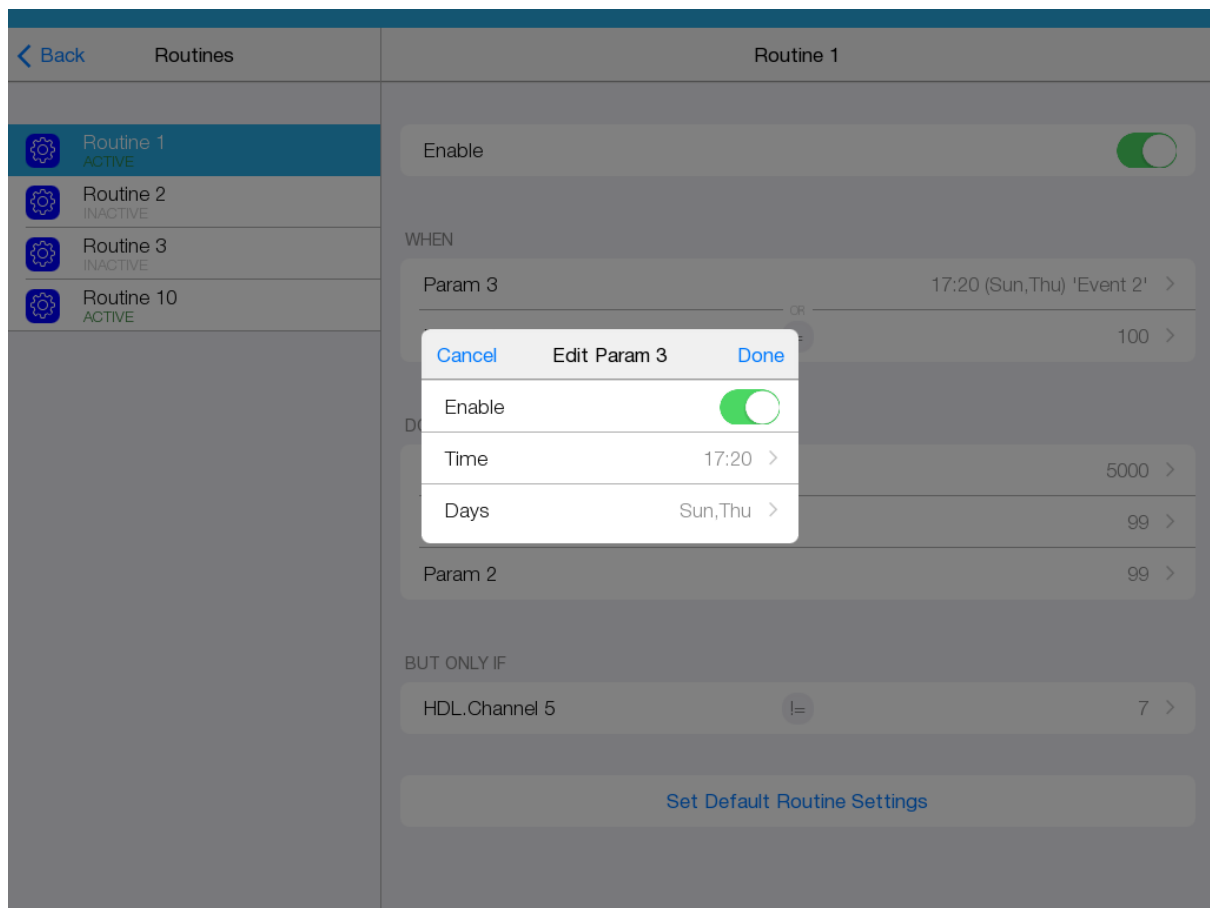
12.4. Использование редактируемых правил в проекте

Для того чтобы начать редактирование серверных правил, в SCADA-клиенте необходимо нажать кнопку, к которой привязана команда открытия окна редактора. На экране отобразится окно редактора:



В окне редактора можно:

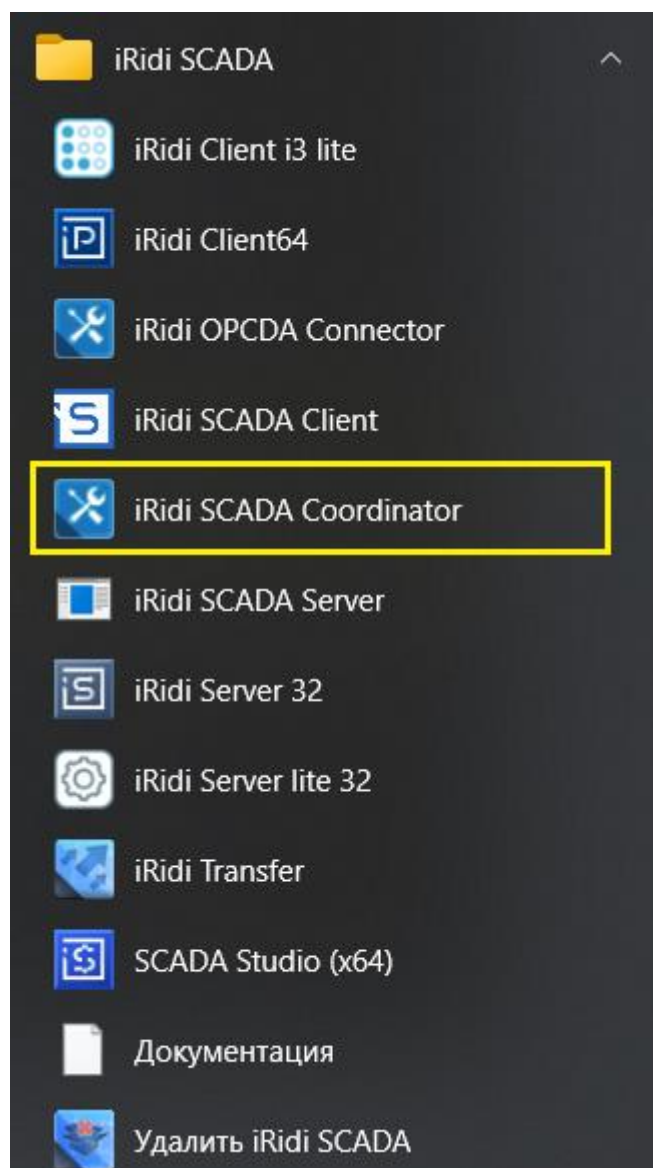
- Активировать/деактивировать параметр или всё правило (Enable),
- Изменить оператор сравнения (==, <, > и др.),
- Изменить значение параметра или время активации правила,
- Сбросить настройки правила до настроенных по умолчанию (Set Default Routine Settings).



13. iRidi SCADA Coordinator

13.1. Описание

iRidi SCADA Coordinator - это приложение, обеспечивающее мониторинг и управление состояниями серверов группы резервирования. Он позволяет получать актуальные состояния и менять статусы резервирования серверов в автоматическом или ручном режиме (пока не реализовано).

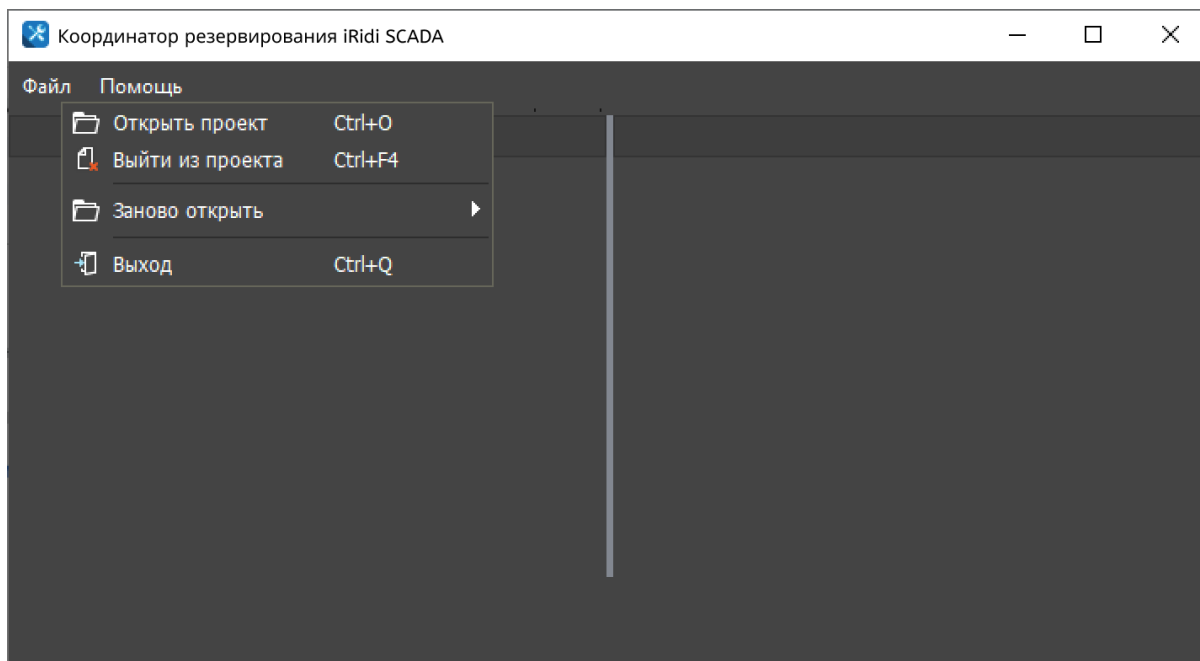


iRidi SCADA Coordinator поставляется в составе сетапа iRidi SCADA.

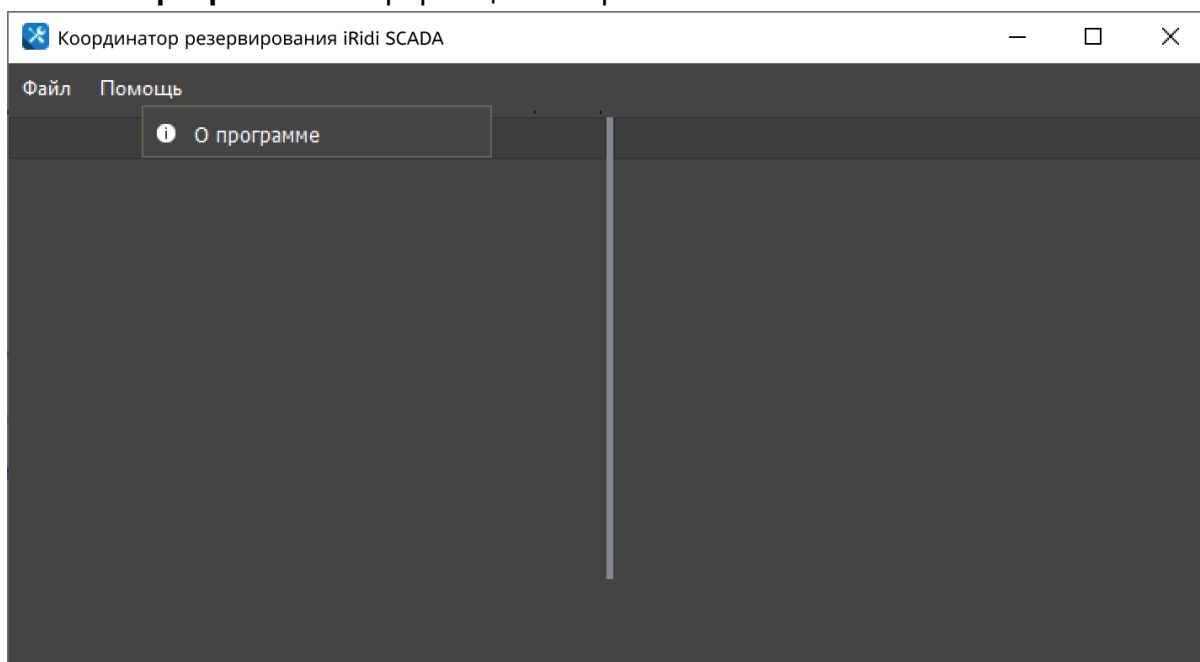
После настройки резервирования в разделе Резервирование в настройках проекта (**п.2.2.5 данной инструкции**), без открытия и запуска проекта в iRidi SCADA Coordinator, **система работать не будет.**

13.2. Меню iRidi SCADA Coordinator

1. Меню **Файл**:
 - **Открыть проект** - открыть сформированный файл резервирования (формат .irsc).
 - **Выйти из проекта** - выйти из проекта.
 - **Заново открыть** - просмотреть список последних просматриваемых проектов.
 - **Выход** - выйти из iRidi SCADA Coordinator.



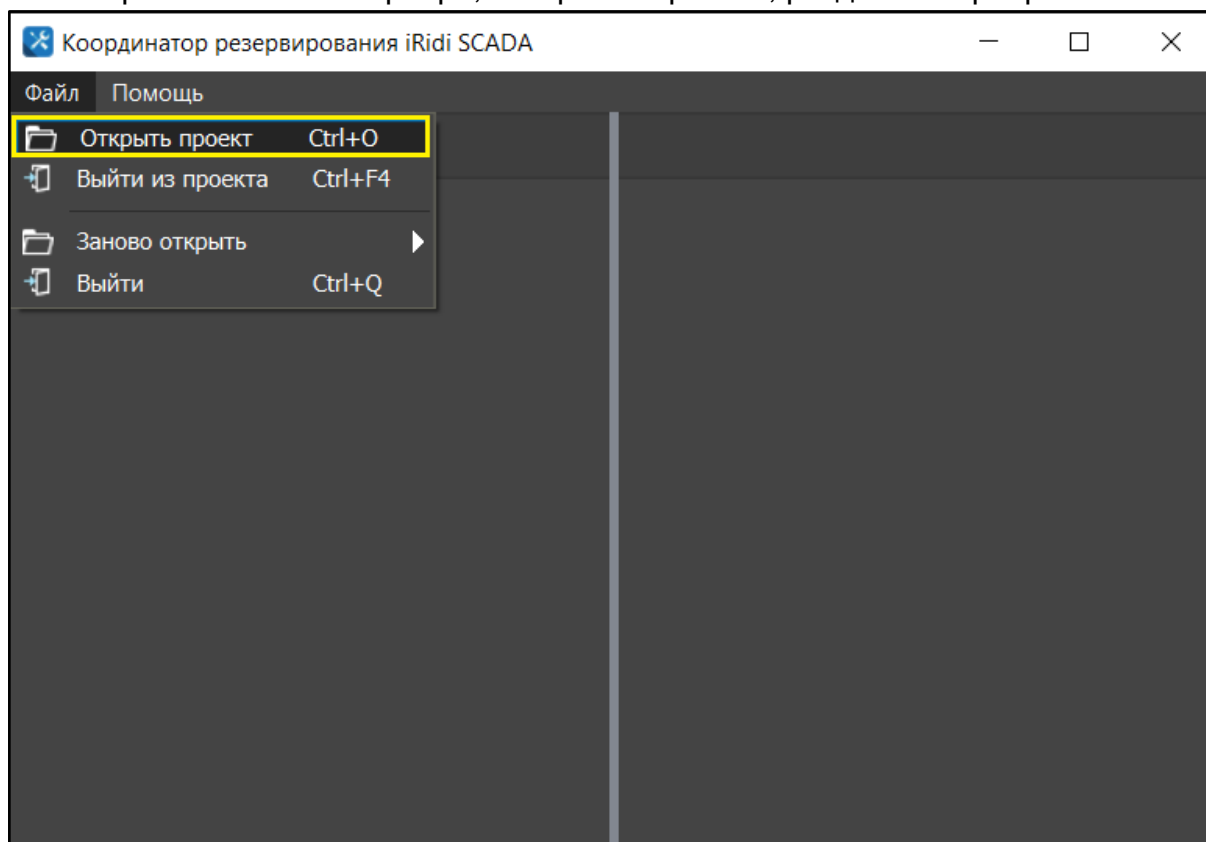
2. Меню **Помощь**:
 - **О программе** - информация о версии iRidi SCADA Coordinator.



13.3. Создание проекта iRidi SCADA Coordinator

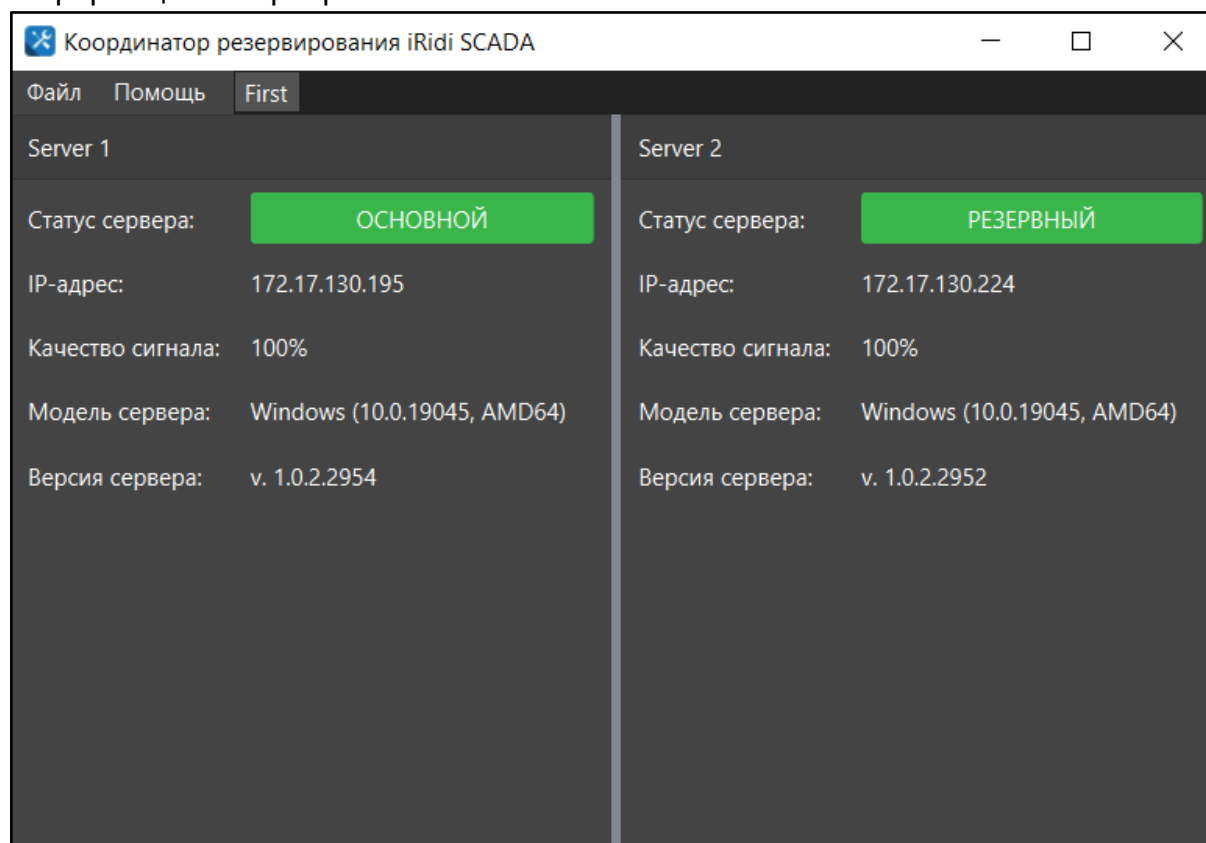
Для того чтобы создать проект в iRidi SCADA Coordinator, необходимо:

1. Открыть iRidi SCADA Coordinator.
2. В меню Файл - Открыть проект - находим сформированный файл для резервирования (формат .irsc). Файл для резервирования формируется в проекте SCADA-сервера, настройки проекта, раздел Резервирование.

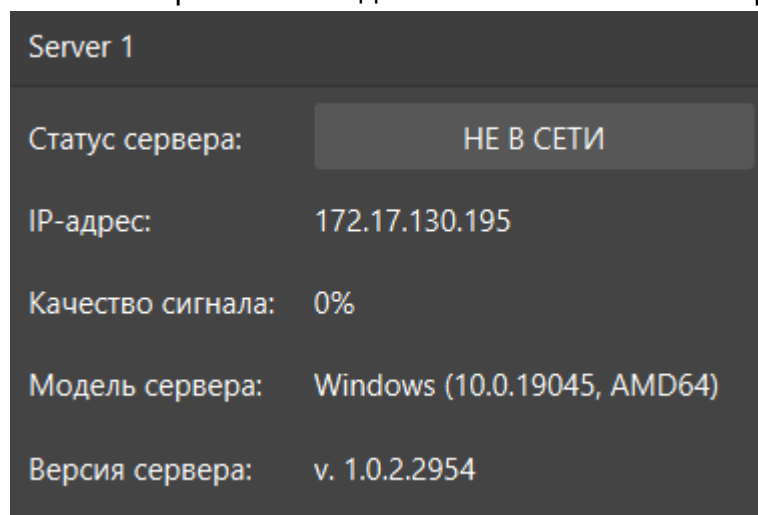


13.4. Работа в iRidi SCADA Coordinator

После открытия проекта в iRidi SCADA Coordinator отобразится следующая информация о серверах:



- **Статус сервера** - здесь отображается статус сервера и его приоритете (ОСНОВНОЙ или РЕЗЕРВНЫЙ). Если сервер в сети, то отображается приоритет сервера на зеленом фоне, если сервер не в сети, то отображается надпись - "НЕ В СЕТИ" на сером фоне.



- **IP-адрес** - IP-адрес сервера в сети;
- **Качество сигнала** в процентах;
- **Модель сервера** - модель устройства, на котором запущен сервер;
- **Версия сервера** - версия ПО на сервере.

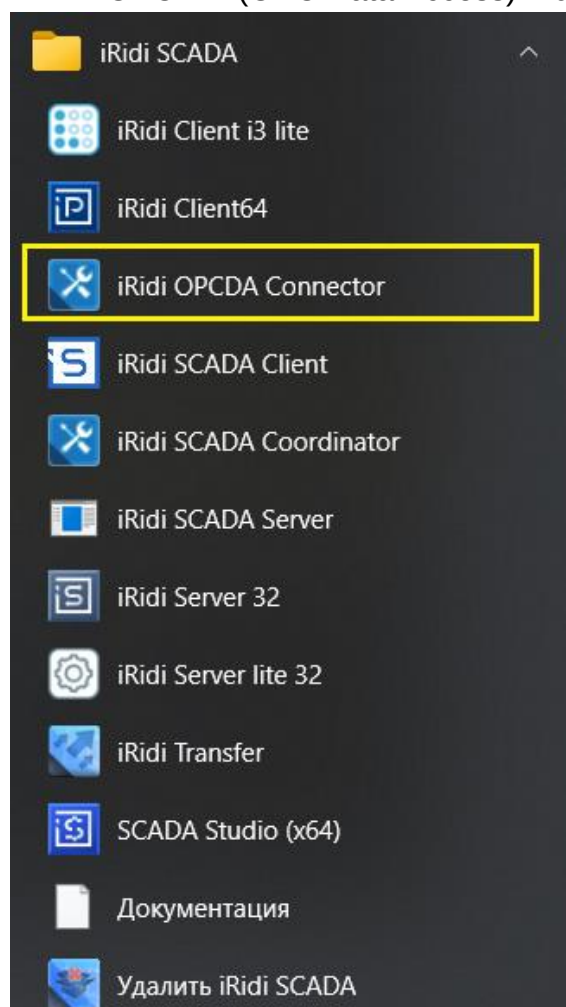
14. iRidi OPC DA Connector

14.1. Описание

iRidi OPC DA Connector - это приложение, обеспечивающее подключение к OPC DA серверам, перенаправляет полученные данные в SCADA iRidi и обратно на OPC DA сервера.

Подключается к OPC DA серверам стандартов:

- OPC DA (OPC Data Access) 1.0a;
- OPC DA (OPC Data Access) 2.0a;
- OPC DA (OPC Data Access) 2.05a.

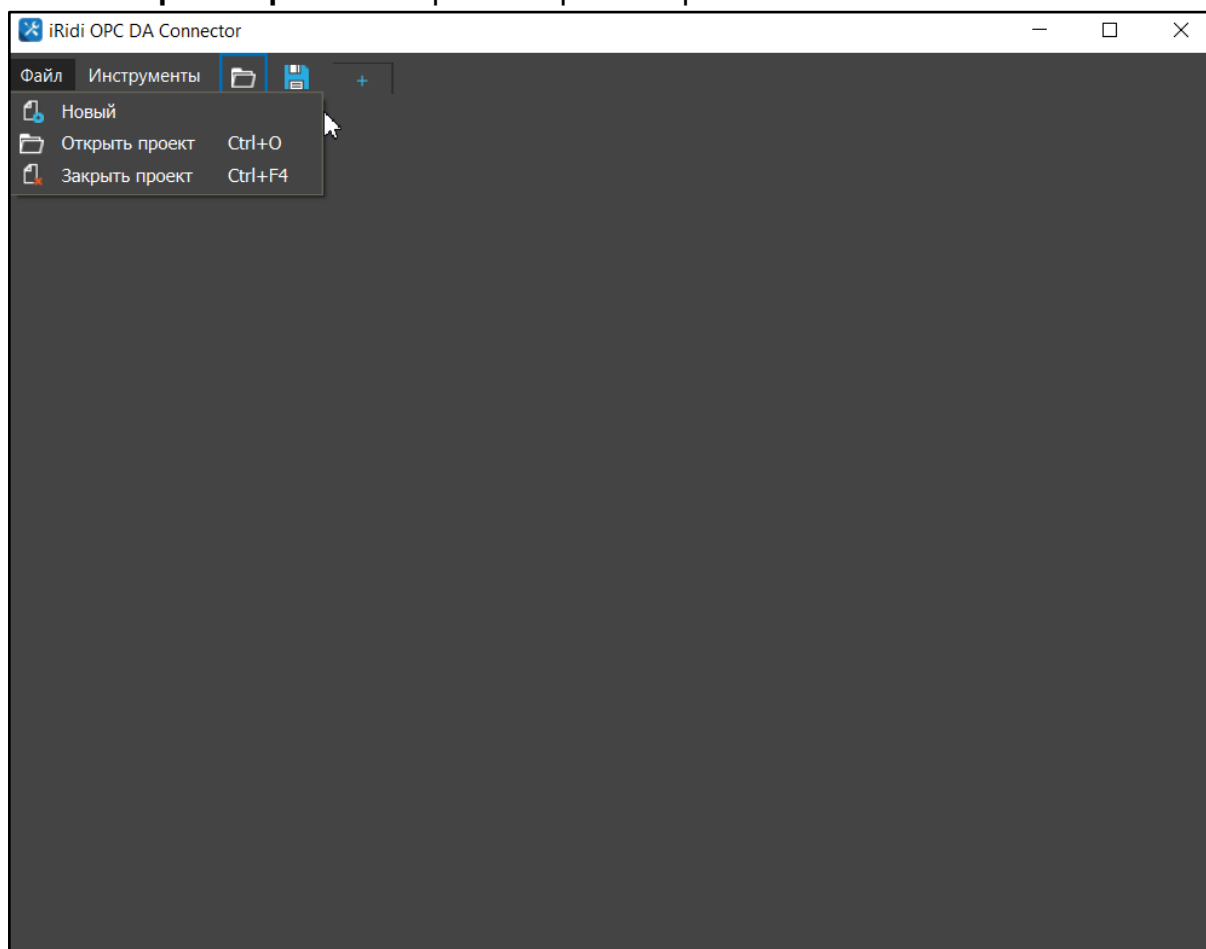


iRidi OPC DA Connector поставляется в составе сетапа iRidi SCADA.
Работает **только** с локальными OPC DA серверами.

14.2. Меню iRidi OPC DA Connector

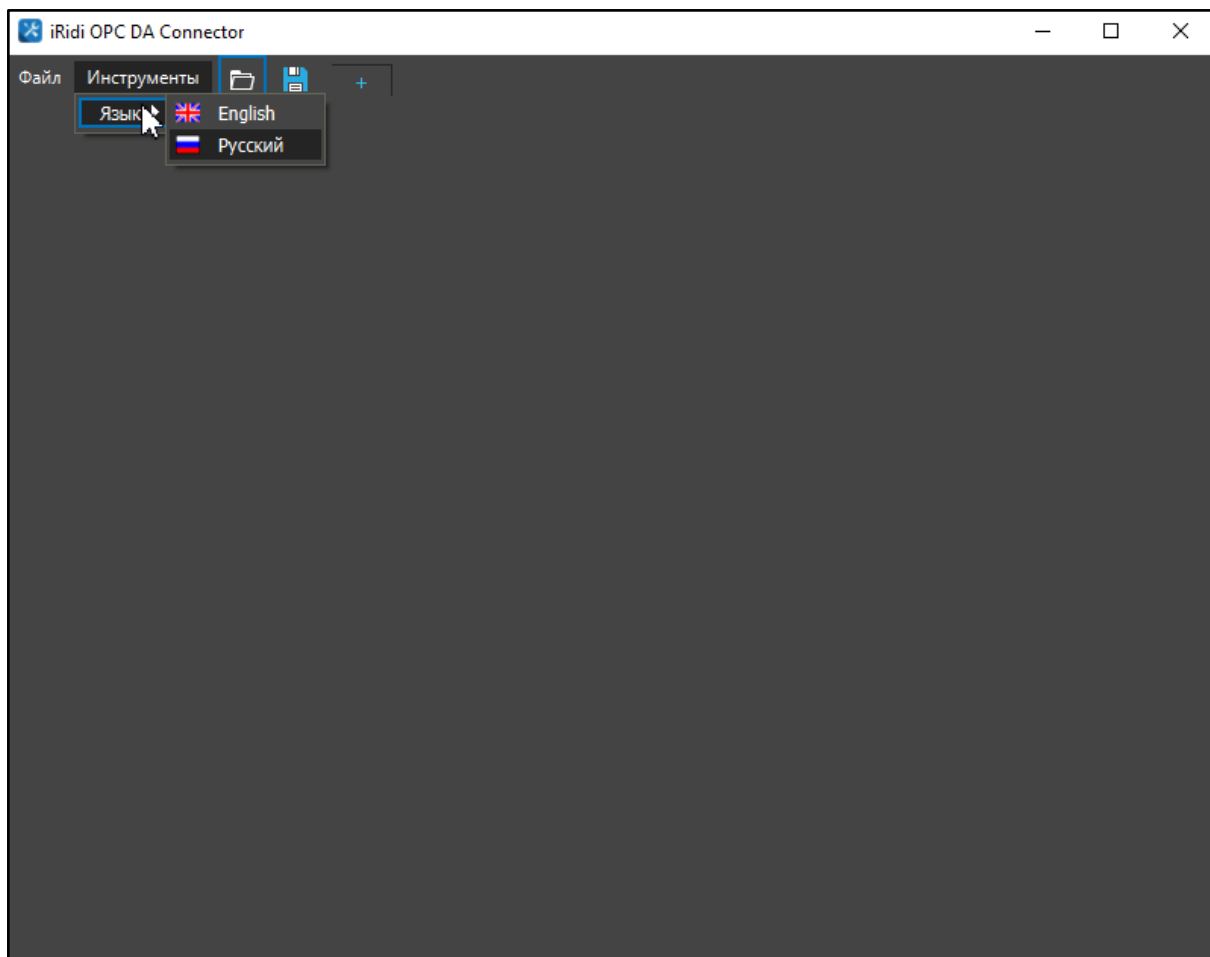
1. Меню **Файл**:

- **Новый** - создать новый проект (формат .igorc).
- **Открыть проект** - открыть ранее созданный проект.
- **Заккрыть проект** - закрыть открытый проект.



2. Меню **Инструменты**:

- **Язык** - выбрать язык приложения. На данный момент доступно 2 языка - Русский и Английский.

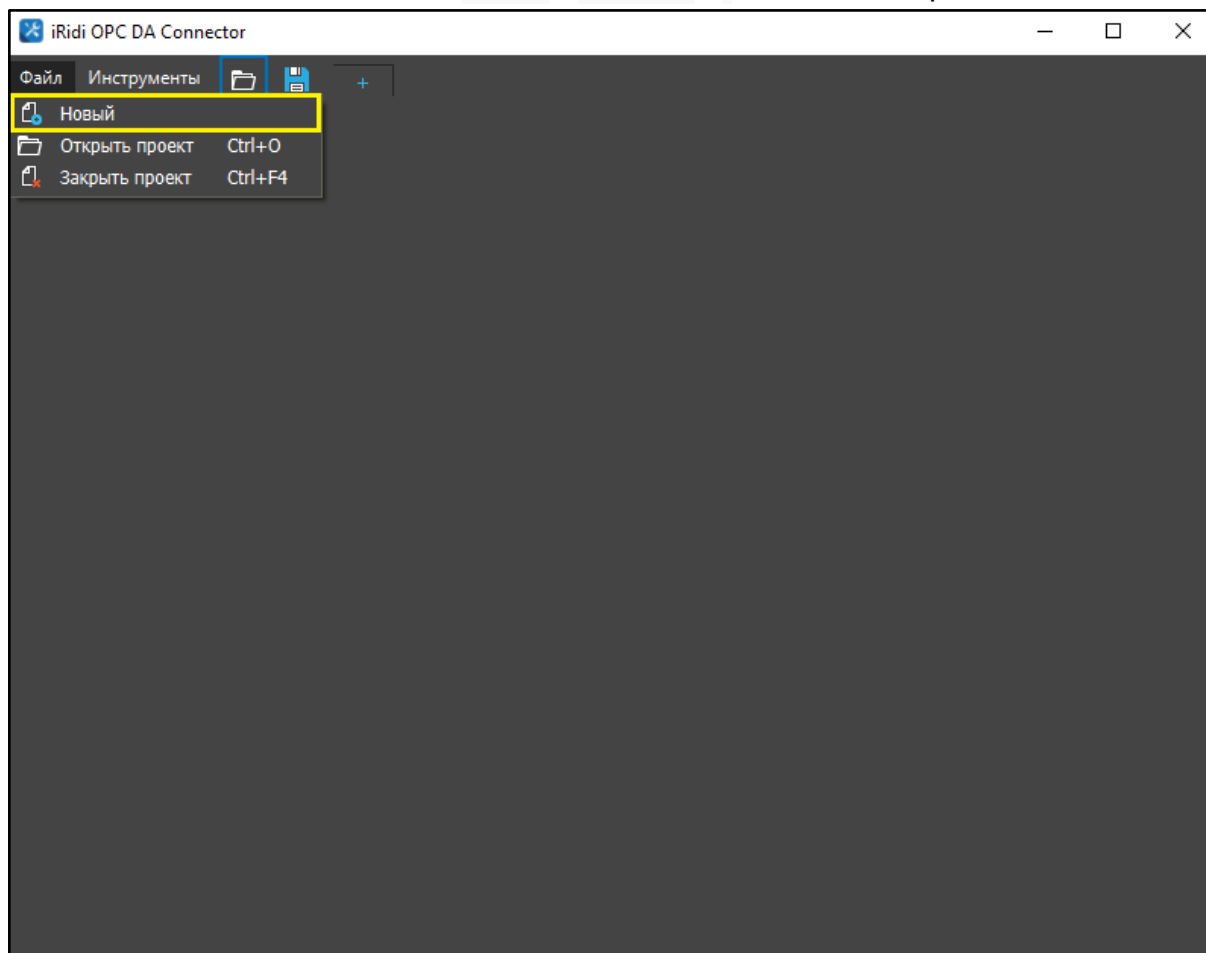


3. На панели инструментов ярлык  - открыть ранее созданный проект,  - сохранить.

14.3. Создание проекта в iRidi OPC DA Connector

Для создания нового проекта в iRidi OPC DA Connector можно применить один из следующих способов:

- Воспользоваться меню Файл → Новый → Ввести имя проекта.

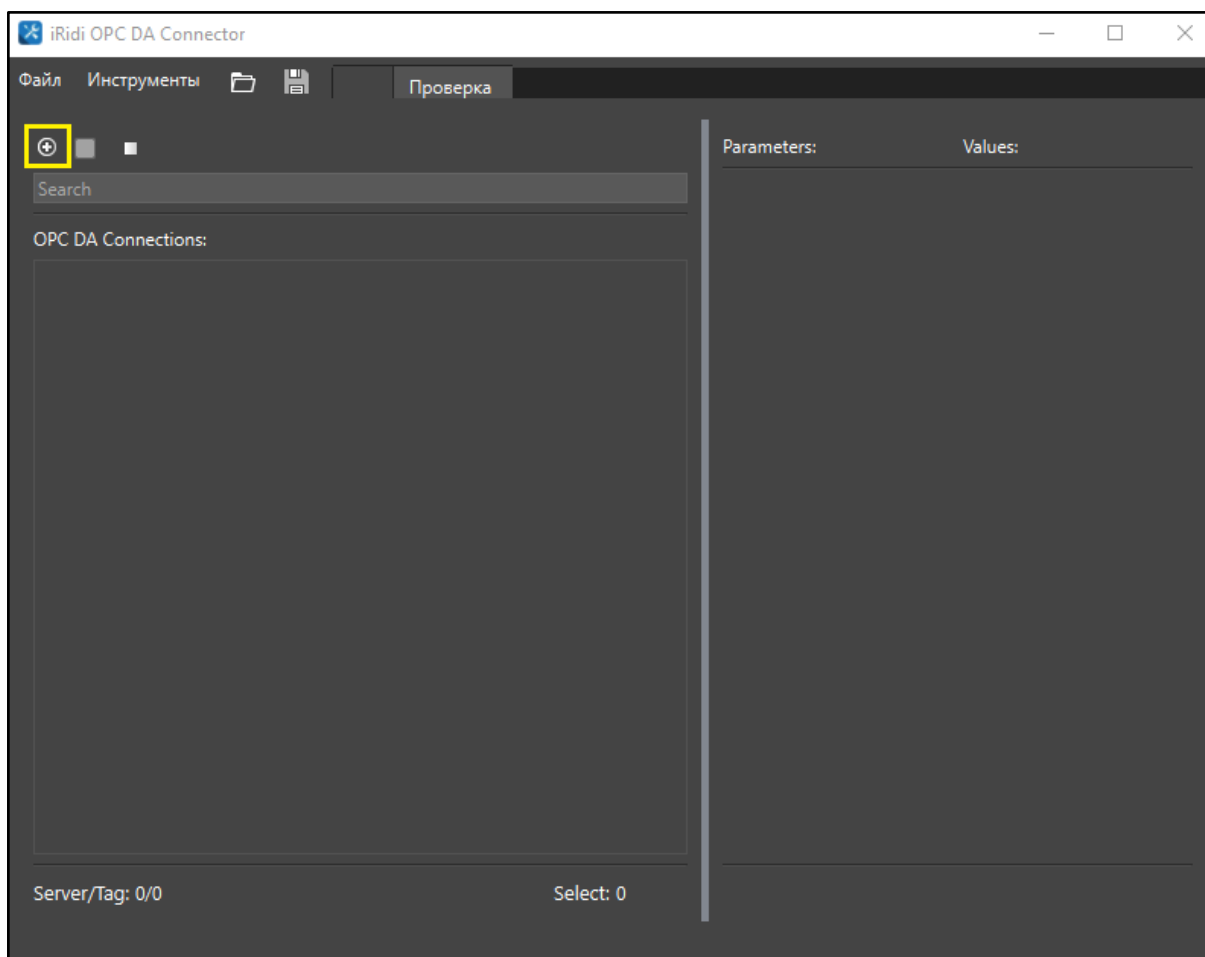


- Нажав иконку  на панели инструментов и ввести имя проекта.

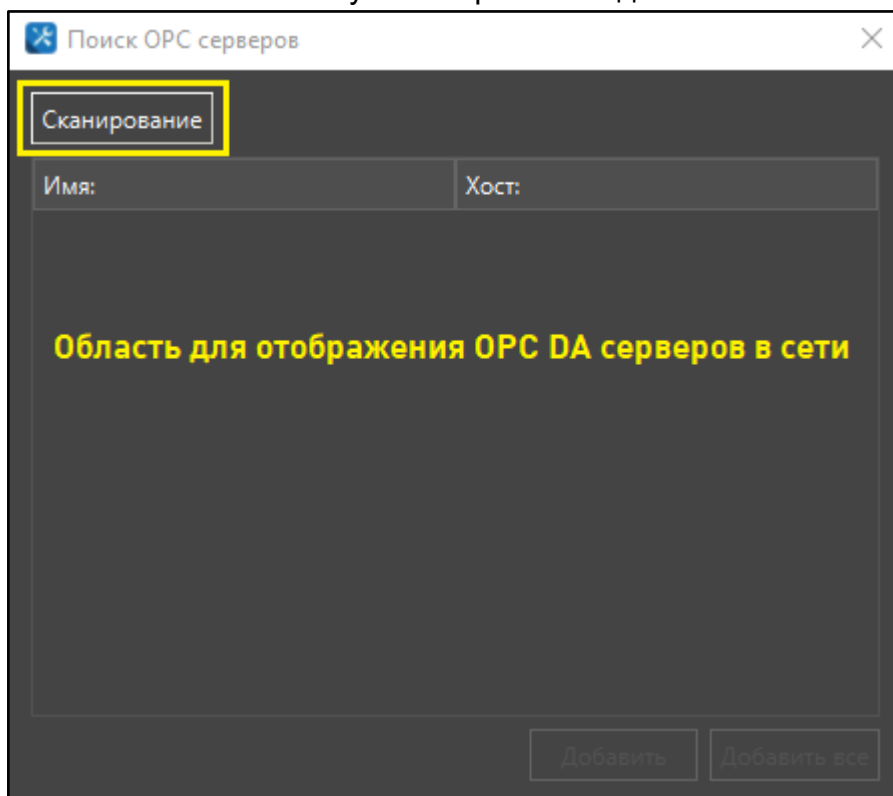
14.3. Добавление OPC DA сервера в проект

Для того чтобы добавить новый OPC DA сервер в проект необходимо:

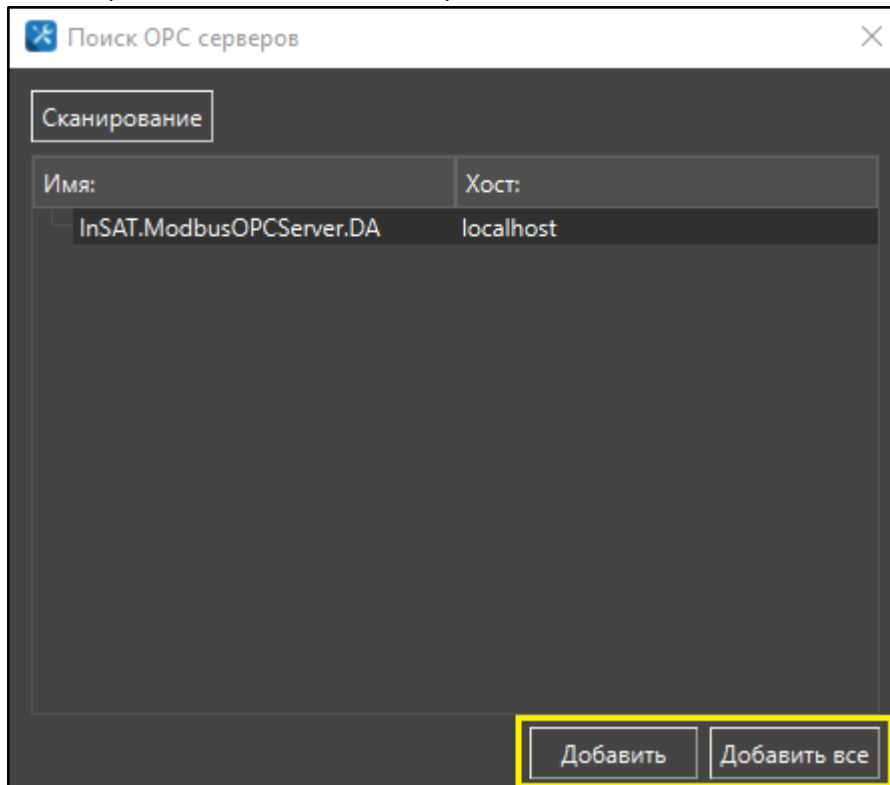
- Нажать на кнопку “+” для открытия окна поиска.



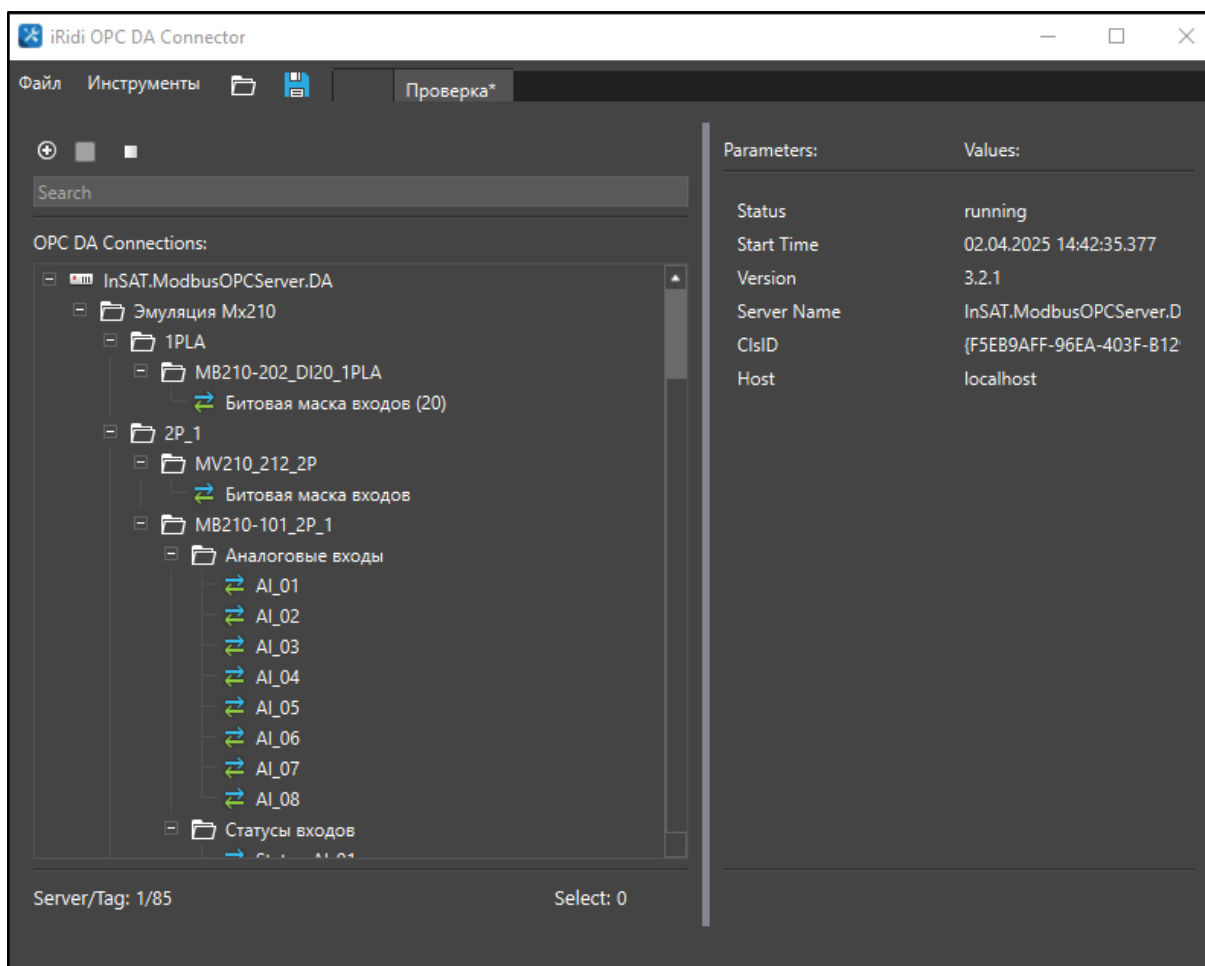
- Нажать на кнопку “Сканирование” для поиска OPC DA серверов в сети.



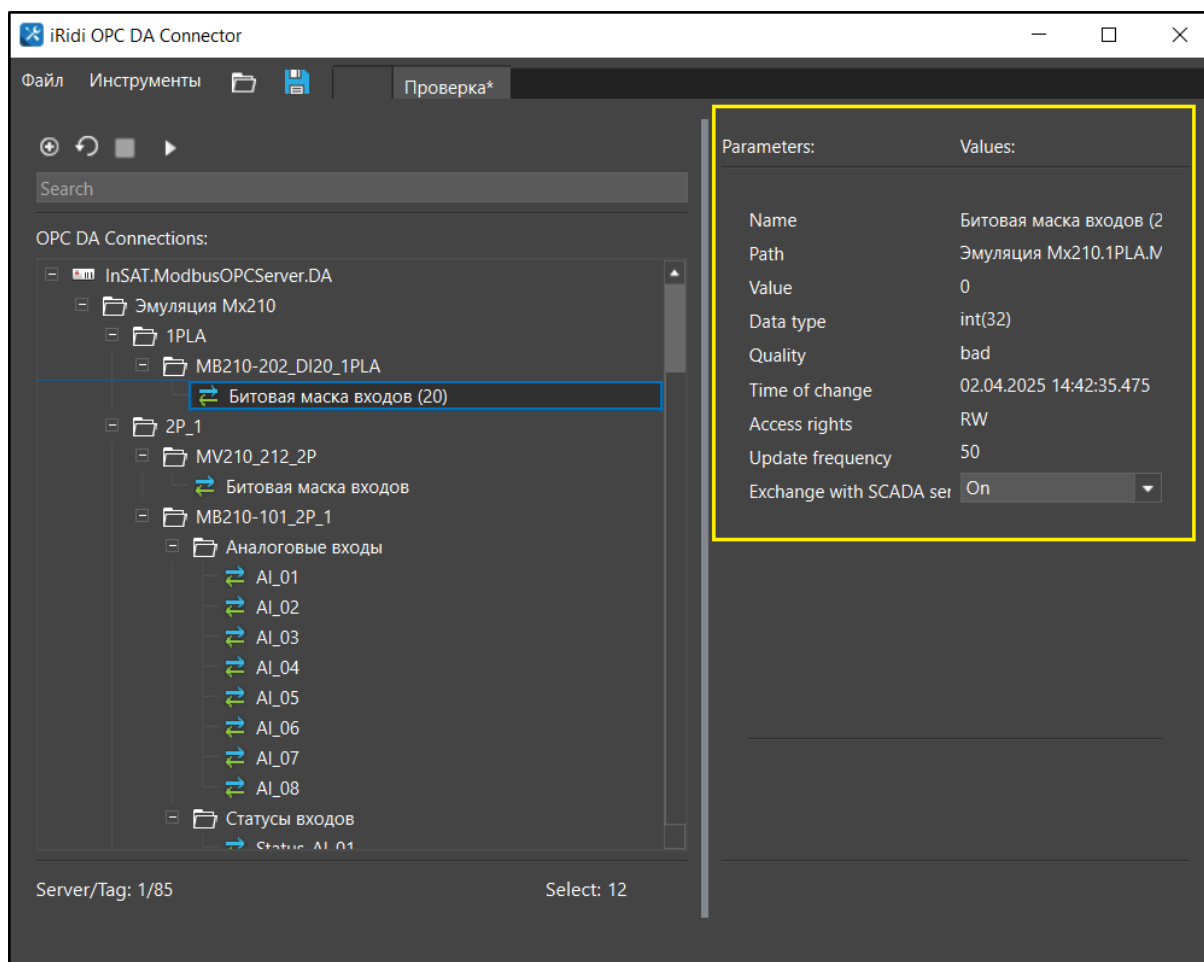
- После того, как были найдены все OPC DA сервера в сети, необходимо добавить нужный сервер (кнопка “Добавить”) или все сервера в проект (кнопка “Добавить все”).



- В проекте появится сервер со всеми папками, подустройствами и тегами.



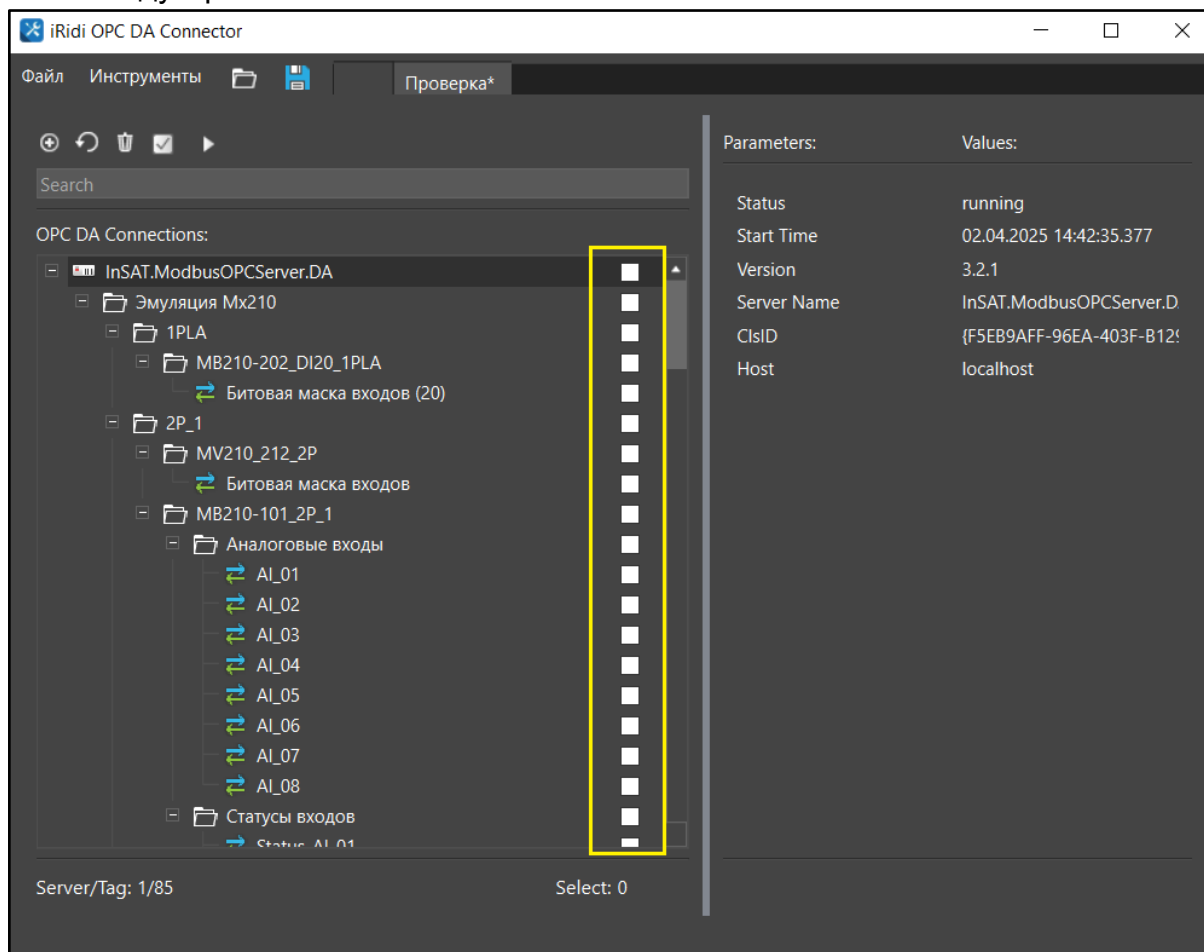
- Кнопка  - актуализирует данные по OPC серверам (удаляет теги, добавляет отсутствующие теги, актуализирует права доступа).
- Кнопка  - удаляет из дерева OPC сервер, отменяет передачу тегов удалённого OPC сервера в iRidi SCADA.
- Кнопка  - делает видимыми чек-боксы выбора тегов для передачи в SCADA.
- Кнопка  /  - запускает или останавливает опрос всех OPC серверов.
 -  - активный опрос серверов (при клике останавливает опрос);
 -  - неактивный опрос серверов (при клике запускает опрос).
- Информация о каждом из элементов в дереве проекта представлена справа.



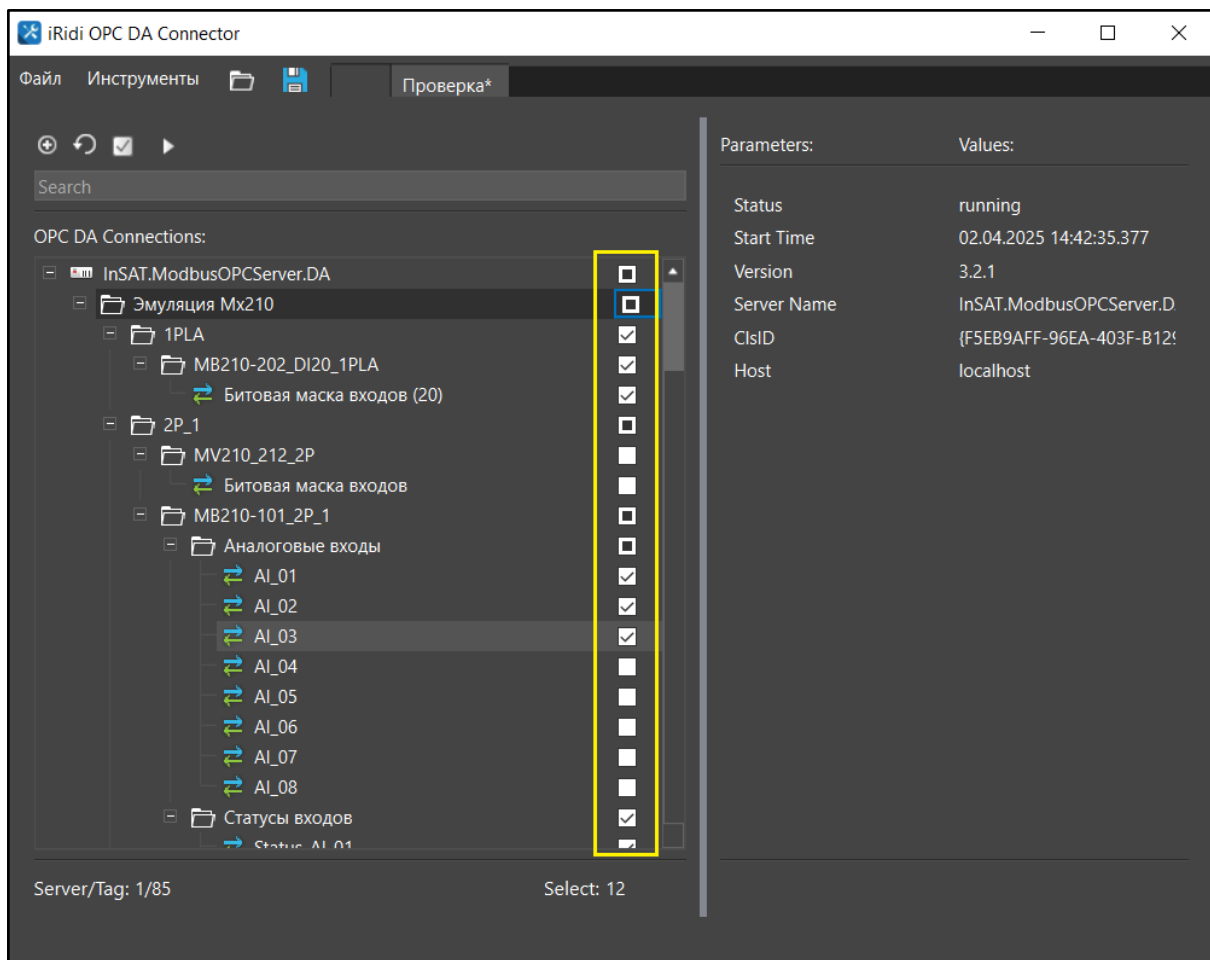
14.4. Выбор тегов для работы в iRidi SCADA

Для того чтобы добавить теги, которые будут отображаться в серверном SCADA-проекте:

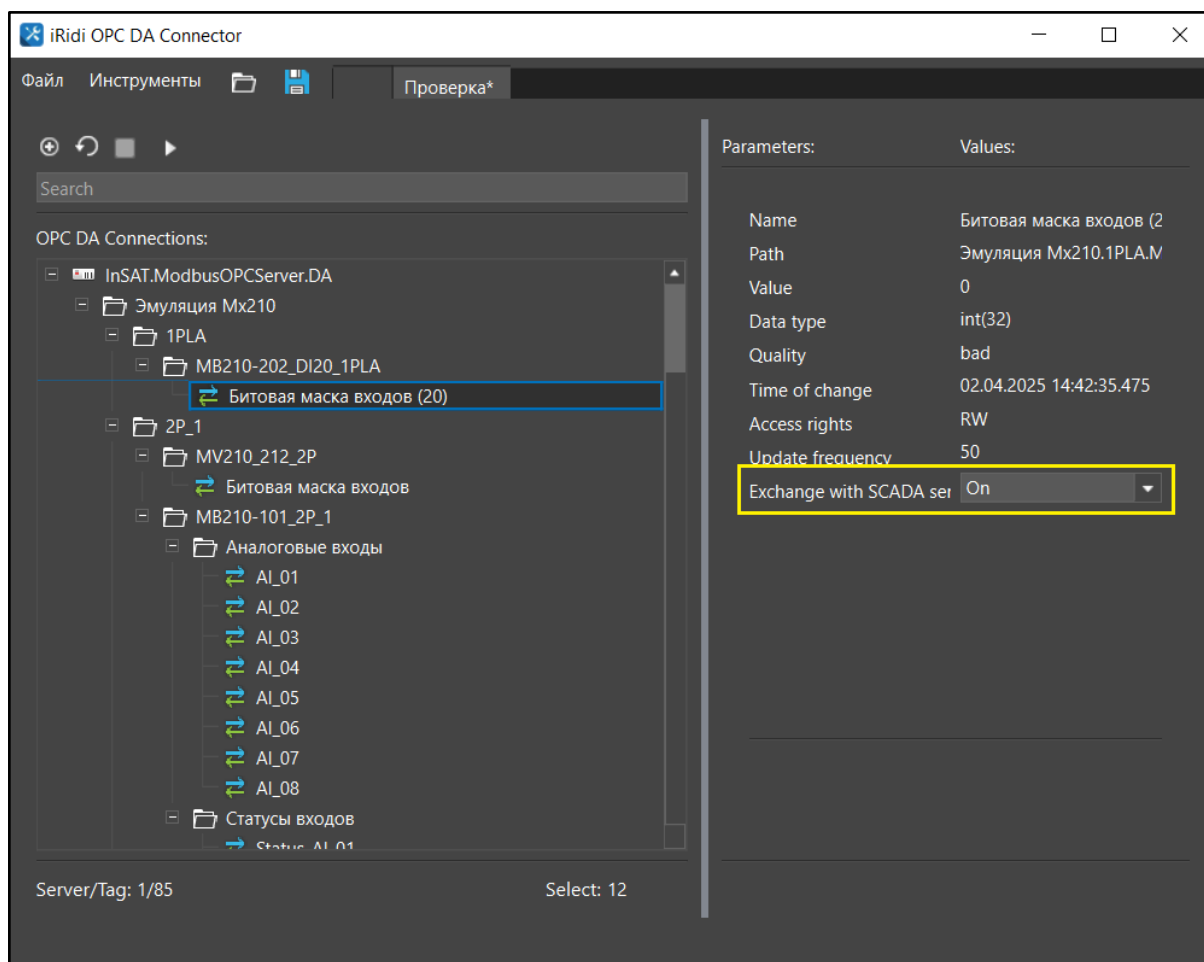
- Нажимаем значок  на панели инструментов. У всех устройств, папок, подустройств и тегов появятся чекбоксы.



- Выберите те элементы, которые хотите перенести в серверный SCADA-проект.



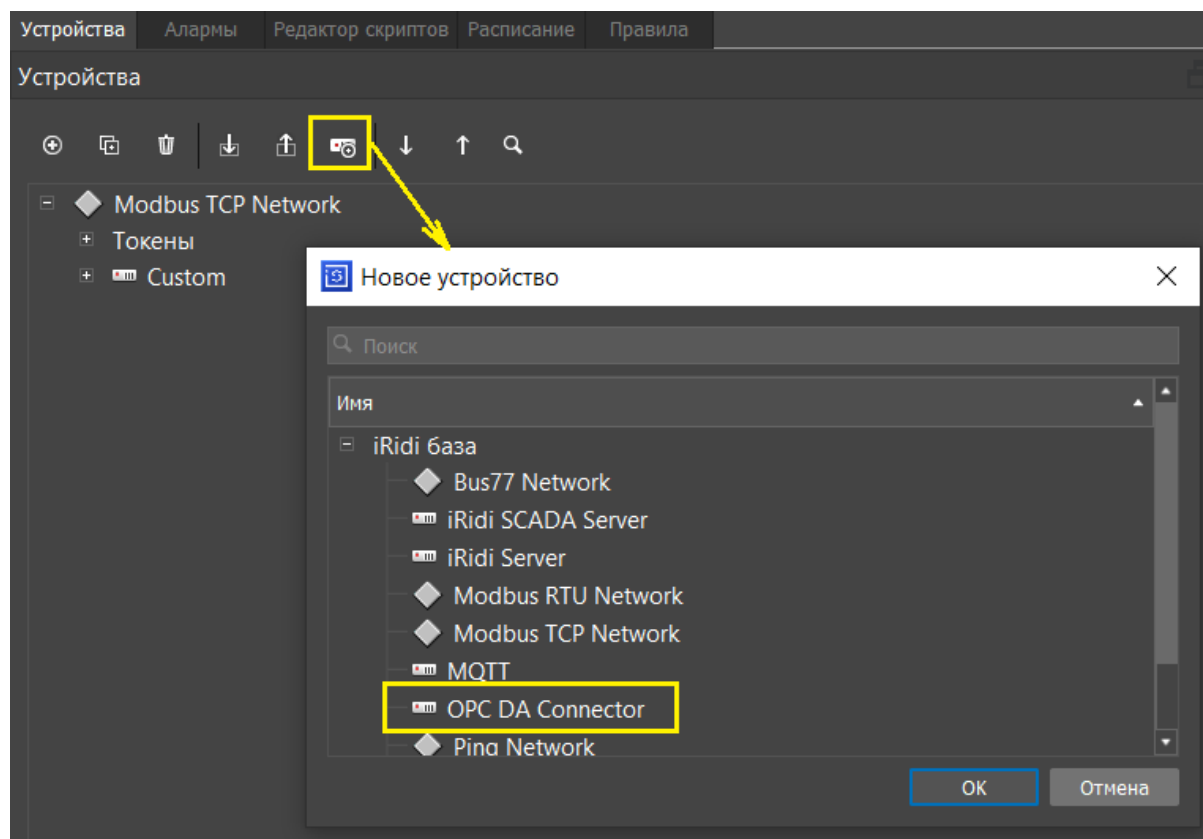
- Знак ☐ обозначает, что элемент не выбран, ☒ - в папке есть выбранные теги, но не все, ☒ - элемент выбран.
- Если выключить режим выбора тегов, то чекбоксы скрываются, но теги сохраняют состояние выбранного.
- Для выбора элементов можно использовать свойство - Exchange with SCADA server (On - передавать значение в SCADA, Off - не передавать значение в SCADA).



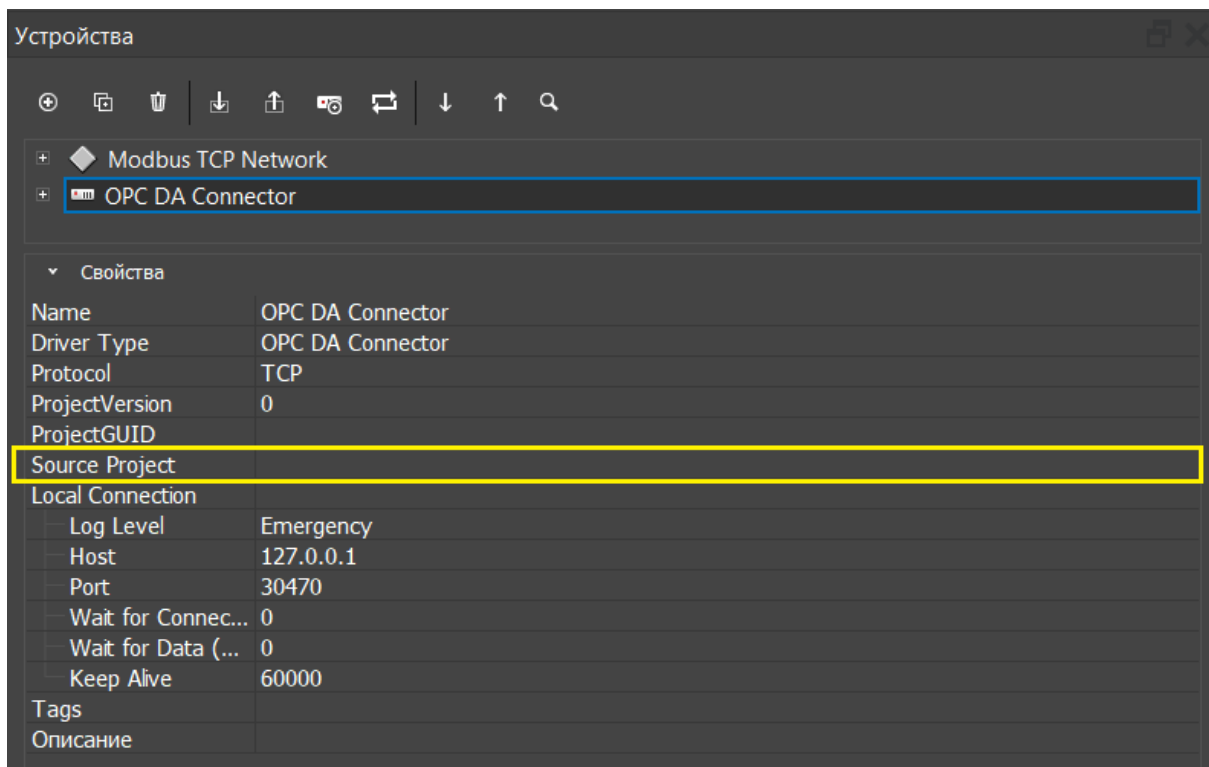
14.5. Добавление тегов OPC DA в проект SCADA

Последовательность добавления тегов OPC DA в проект SCADA-сервера:

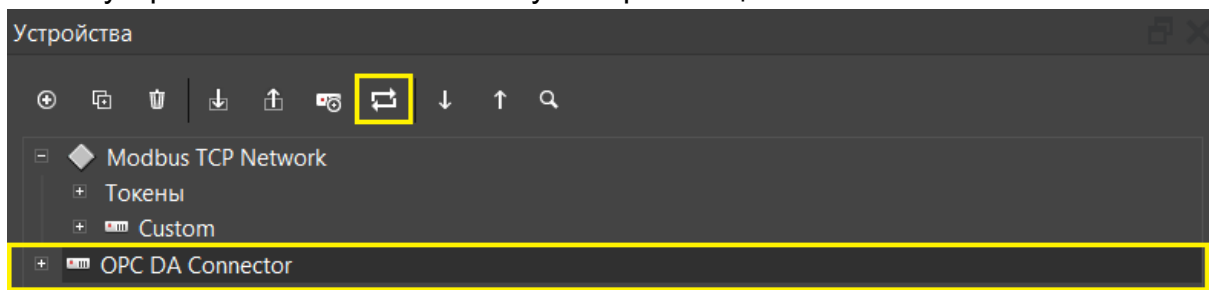
- В разделе Устройства серверного SCADA-проекта добавить драйвер OPC DA Connector.



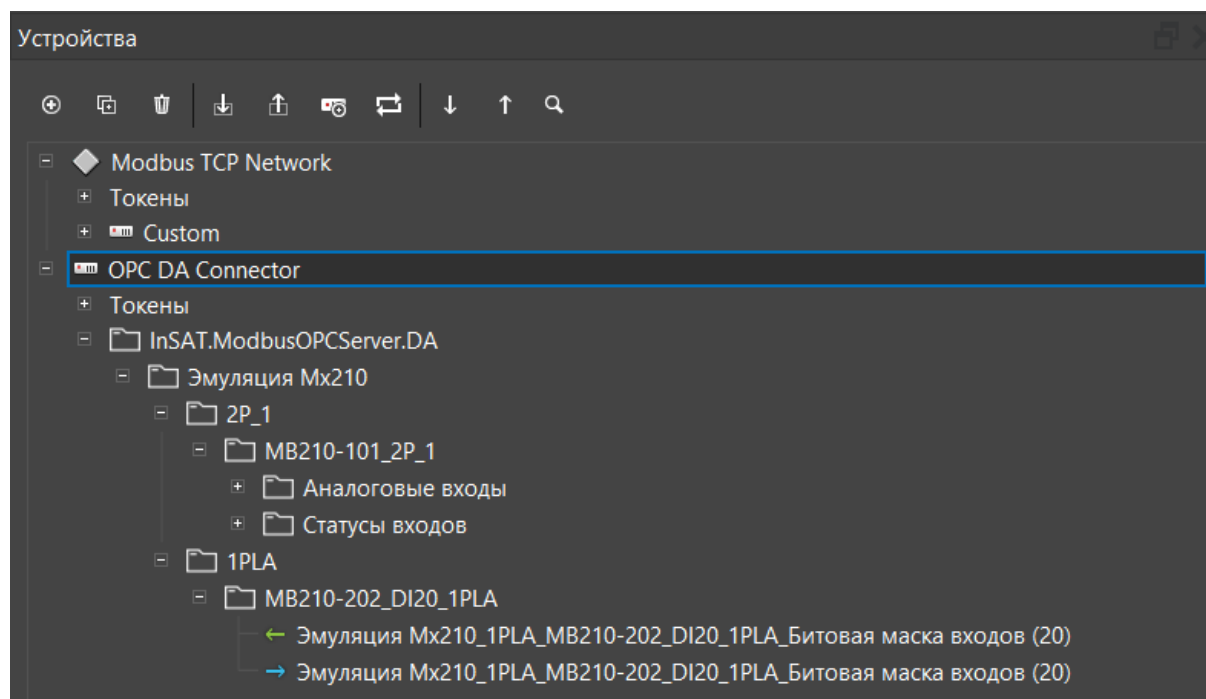
- В настройках драйвера в качестве Source Project выбираем проект созданные в приложении iRidi OPC DA Connector (файл с расширением .iropc).



- Синхронизируем проекты. Для этого выбираем драйвер в списке устройств и нажимаем кнопку синхронизации.



- В списке отобразятся все теги выбранные в Ridium OPC DA Connector.

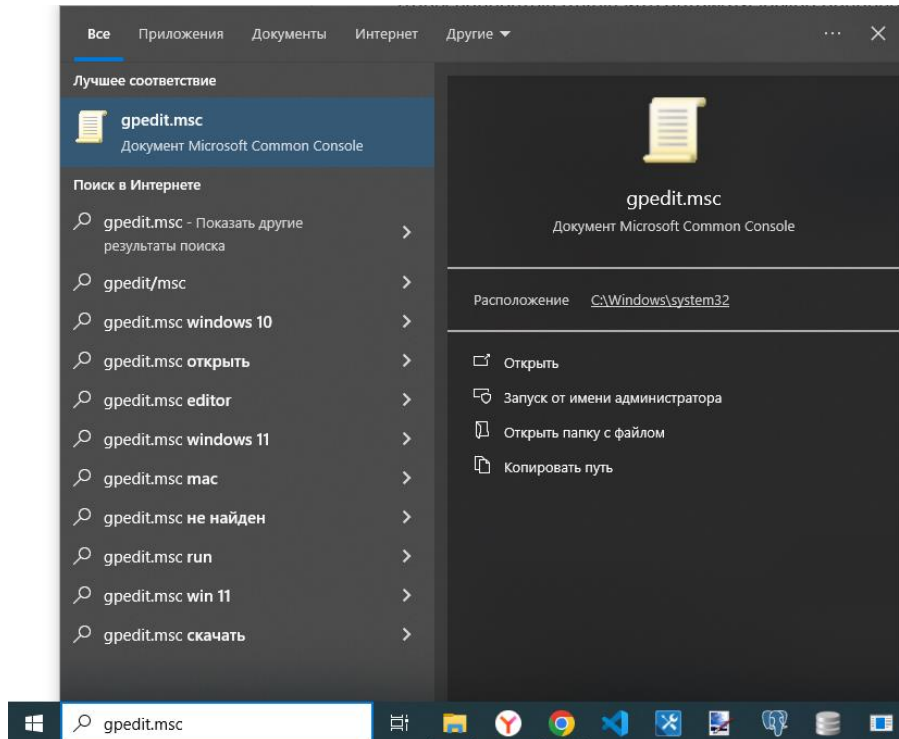


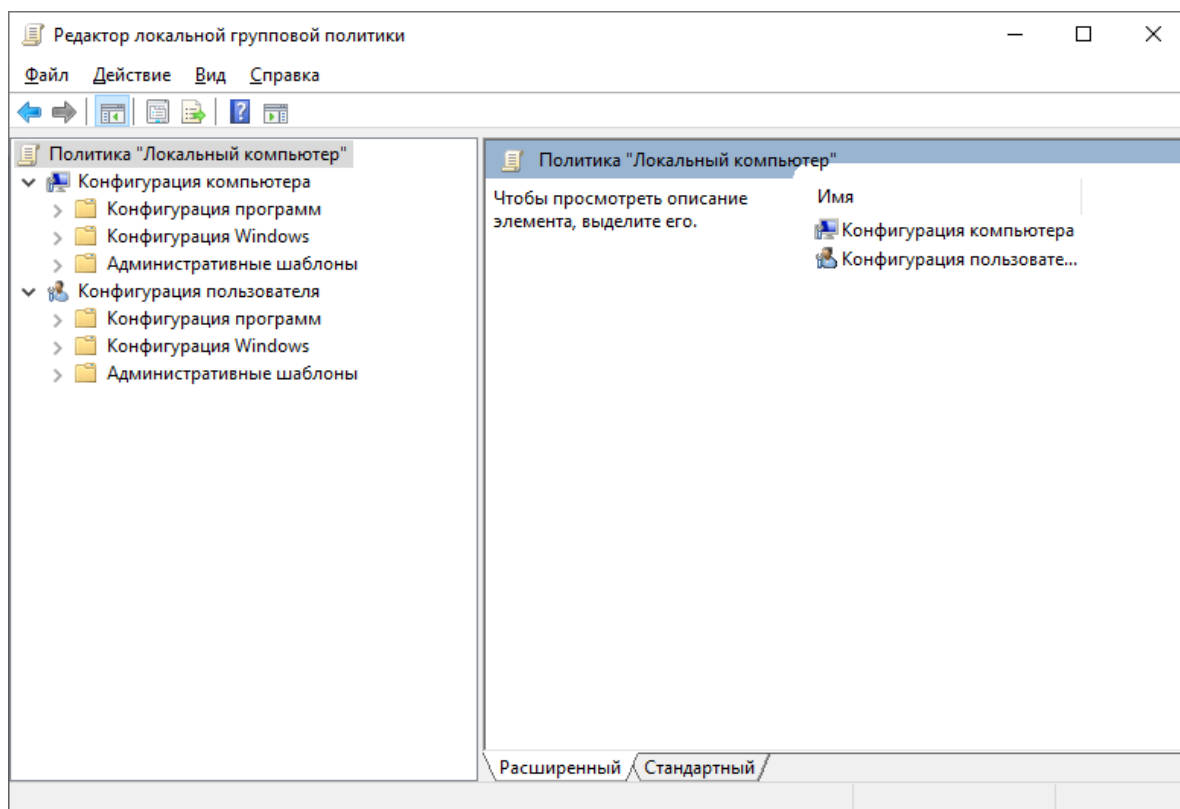
15. Отключение автоматического обновления Windows

Источник: <https://learn.microsoft.com/ru-ru/windows/iot/iot-enterprise/os-features/updates#disable-automatic-windows-updates>

Чтобы полностью отключить автоматическое обновление Windows с Обновл. Windows:

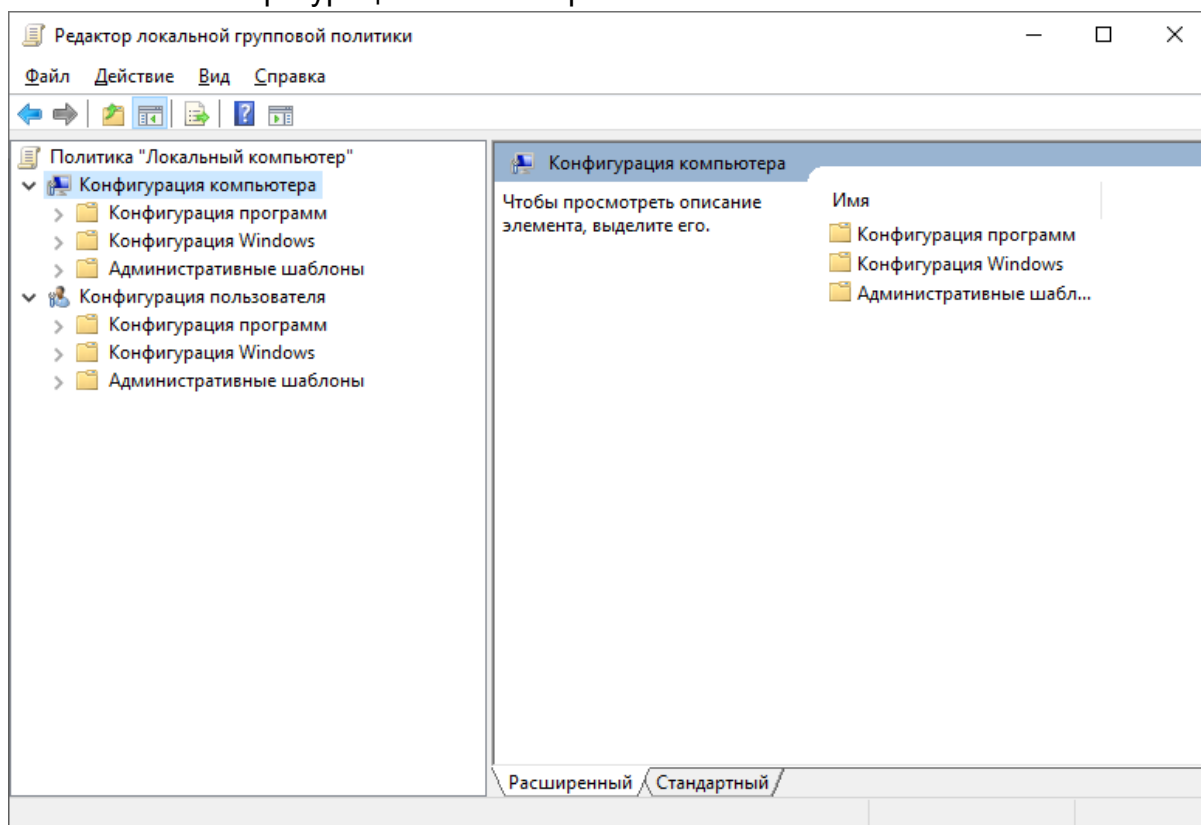
1. Откройте редактор групповой политики (gpedit.msc)



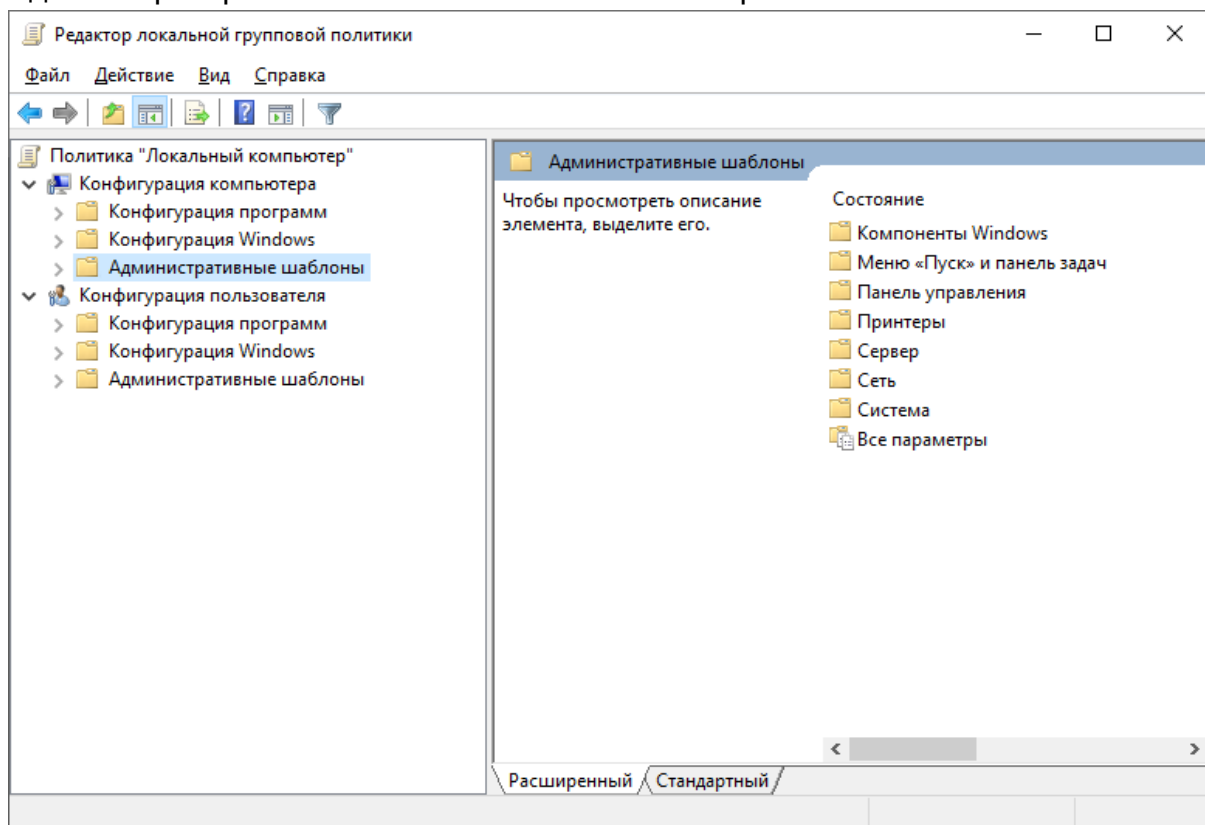


2. перейдите в раздел:

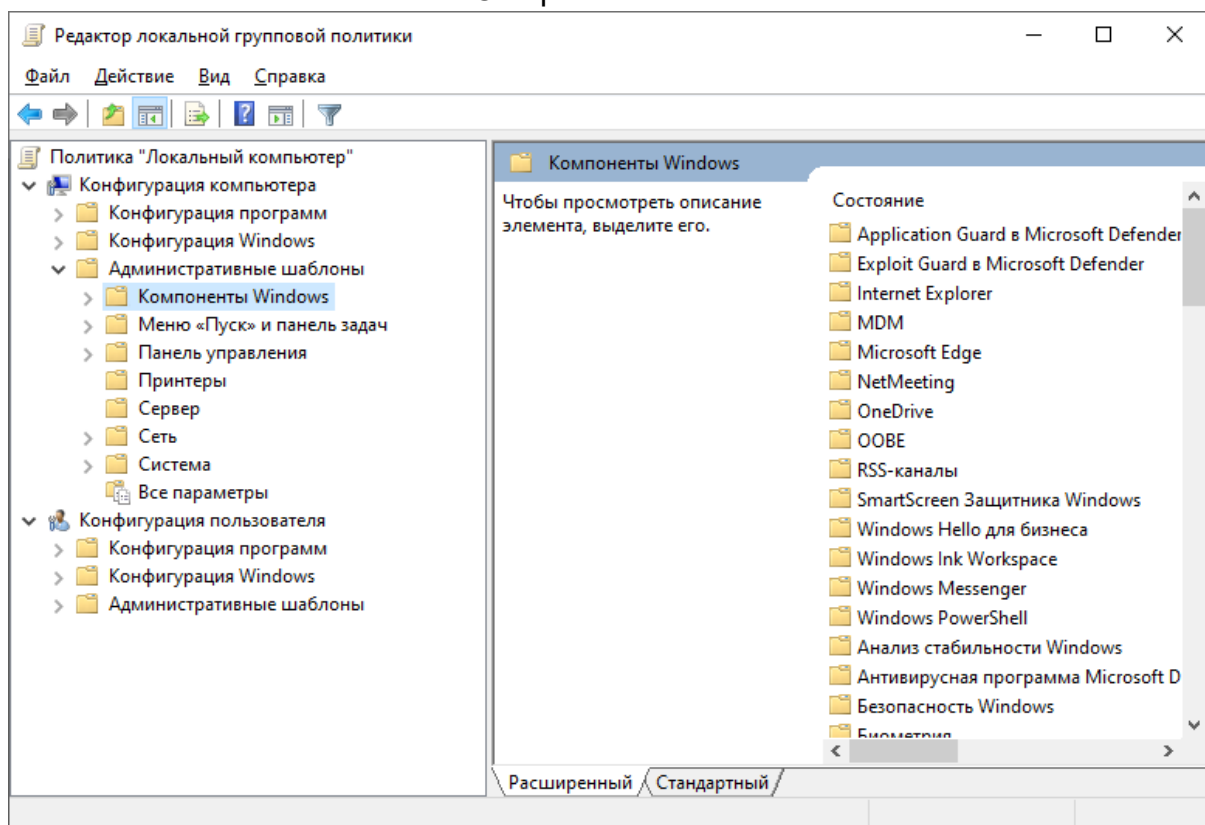
2.1. Конфигурация компьютера →



→



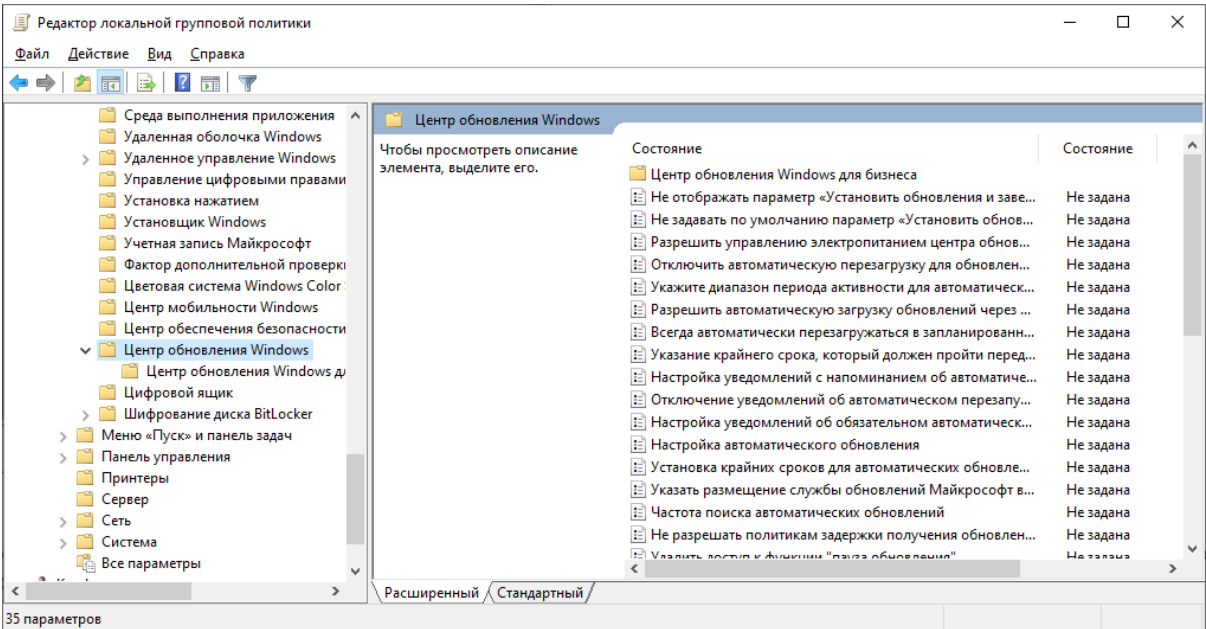
→



2.4.

Windows

update

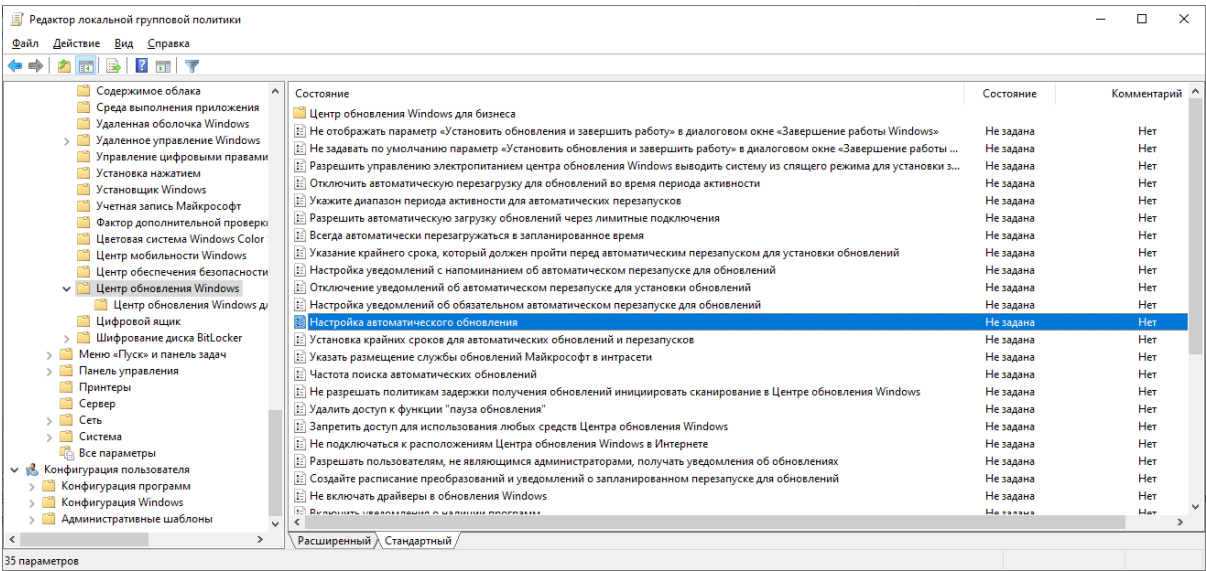


2.5.

Configure

Automatic

Обновления.



3. Явно задайте для политики значение "Отключено"

Настройка автоматического обновления

Настройка автоматического обновления

Предыдущий параметр Следующий параметр

☐ Не задано Комментарий:

☐ Включено

☒ Отключено

Требования к версии: Windows XP Professional с пакетом обновления 1 (SP1) или по меньшей мере Windows 2000 с пакетом обновления 3 (SP3)
Вариант 7 поддерживается только на серверах под управлением

Параметры: Справка:

Настройка автоматического обновления:

Следующие параметры политики являются обяза

☐ Устанавливать во время автоматического обсл

Установка по расписанию — день:

Установка по расписанию — время:

Если вы выбрали вариант "4 — автоматическое с установки и указали расписание, у вас также есть две недели или в месяц), используя варианты, оп

☐ Еженедельно

☐ Первая неделя месяца

☐ Вторая неделя месяца

Этот параметр определяет, будет ли данный компьютер получать обновления для системы безопасности и другие важные загружаемые компоненты через службу автоматического обновления Windows.

Примечание. Эта политика не применяется к Windows RT.

Этот параметр позволяет указать, разрешено ли автоматическое обновление на этом компьютере. Если служба включена, необходимо выбрать один из четырех вариантов в параметре групповой политики:

2 = Уведомлять перед загрузкой и установкой любых обновлений.

Когда Windows находит обновления, применимые к данному компьютеру, пользователи получают уведомления о готовности обновлений к загрузке. После перехода в центр обновления Windows пользователи могут загрузить и установить все доступные обновления.

OK Отмена Применить

Если для этого параметра задано значение "Отключено", все доступные обновления из Обновл. Windows необходимо скачать и установить вручную, что можно сделать в приложении Параметры в разделе "Обновление и безопасность > Обновл. Windows".