УТВЕРЖДАЮ

Директор ООО «Континент ЭТС»

_____ А.А. Алексеев

«30» ____06____ 2025 г.

КОМПЛЕКС ТЕХНИЧЕСКИХ СРЕДСТВ ИЗМЕРИТЕЛЬНЫЙ UZOLA PRO100

ИНСТРУМЕНТАЛЬНАЯ СИСТЕМА MASTERSCADA 4D

Руководство программиста

Лист утверждения

МПВР.00046-01 33 01-ЛУ

Представители предприятия-разработчика

Начальник отдела АСУТП

_____ А.Н. Вовк

«30» ____06____ 2025 г.

Зам. начальника отдела АСУТП

_____ А.С. Морозов

«30» ____06____ 2025 г.

Нормоконтролер

_____ Ю.А. Чанова

«30» ____06____ 2025 г.

2025

Литера

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл.	

УТВЕРЖДЕН

МПВР.00046-01 33 01-ЛУ

КОМПЛЕКС ТЕХНИЧЕСКИХ СРЕДСТВ ИЗМЕРИТЕЛЬНЫЙ

UZOLA PRO100

ИНСТРУМЕНТАЛЬНАЯ СИСТЕМА MASTERSCADA 4D

Руководство программиста

МПВР.00046-01 33 01

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл.	

Содержание

Обозначения и сокращения ... 4
Введение ... 5
1 Назначение и условия применения ... 6
1.1 Назначение ... 6
1.2 Меры безопасности при эксплуатации ... 6
1.3 Подготовка изделия к использованию ... 6
1.4 Подключение модулей КТСИ ... 6
1.5 Указания по включению и работе ... 7
1.6 Порядок выключения и демонтажа после окончания работ ... 8
2 Установка и настройка среды разработки и исполнения «MasterSCADA 4D» ... 9
2.1 Состав MasterSCADA 4D ... 9
2.2 Системные требования среды разработки MasterSCADA 4D ... 10
2.3 Установка и настройка среды разработки «MasterSCADA 4D» ... 13
2.3.1 Установка среды разработки «MasterSCADA 4D» в операционной среде Windows ... 13
2.3.2 Установка среды разработки «MasterSCADA 4D» в операционной среде Linux ... 17
2.4 Настройка среды разработки «MasterSCADA 4D» ... 24
3 Программирование в инструментальной системе «MasterSCADA 4D» ... 26
3.1 Создание проекта и настройка процессорного модуля ... 26
3.2 Добавление и конфигурирование модулей ввода-вывода ... 30
3.3 Настройка протоколов обмена с полевыми устройствами и внешними системами ... 50
3.3.1 Протокол Modbus RTU ... 50
3.3.2 Протокол Modbus TCP ... 64
3.3.3 Протокол OPC UA ... 71
3.3.4 Протокол CANOpen ... 81
3.4 Разработка пользовательского проекта ... 82
3.4.1 Анализ ... 82
3.4.2 Библиотечные элементы ... 85
3.4.3 Программы ... 91
3.4.4 Параметры (переменные проекта) ... 92
3.4.5 Окна ... 93
3.4.6 Редакторы ... 94
3.4.7 Конфигурирование связей ... 96
3.4.8 Формирование задач узлов ... 97
3.4.9 Автономная отладка в реальном времени ... 98
3.4.10 Правила разработки проектов ... 99
3.5 Настройка резервирования контроллеров и резервирования протоколов ... 101
3.6 Загрузка конфигурации в контроллер ... 112
3.7 Диагностика ... 117
3.8 Настройка прав доступа ... 125
3.9 Связь с процессорным модулем ... 129
3.10 Работа модулей в системе КТСИ ... 130
3.11 Методика поиска отказов ... 131
3.12 Устранение отказов ... 131
3.13 Техническая поддержка ... 131

3.14 Информация об Изготовителе 132
Лист регистрации изменений.....133

Обозначения и сокращения

МП – модуль процессорный;

МВВ – модуль ввода-вывода;

КТСИ – комплекс технических средств измерительный;

Введение

Настоящее руководство программиста содержит информацию, необходимую для начала работы в инструментальной системе MasterSCADA 4D. Процесс ознакомления с системой MasterSCADA 4D описан применительно к комплексу технических средств измерительному UZOLA PRO100 (далее КТСИ).

1 Назначение и условия применения

1.1 Назначение

Инструментальная система «MasterSCADA 4D» предназначена для создания пользовательской программы на комплекс технических средств измерительный UZOLA PRO100.

КТСИ имеет блочно-модульную структуру, включающую в себя модули различного типа. Модули являются основным элементом КТСИ и, в зависимости от типа, выполняют ту или иную функцию.

1.2 Меры безопасности при эксплуатации

1.2.1 К эксплуатации КТСИ допускаются лица, изучившие руководство по эксплуатации изделия и прошедшие инструктаж о соблюдении правил безопасности при работе с электроустановками.

1.2.2 Персонал, выполняющий работы по эксплуатации КТСИ, должен иметь квалификацию не ниже квалификационной группы III по ПТЭЭП.

1.3 Подготовка изделия к использованию

1.3.1 Перед началом проведения работ по эксплуатации КТСИ следует ознакомиться с эксплуатационной документацией на изделие, требованиями безопасности, а также другими нормативными и эксплуатационными документами и строго ими руководствоваться.

1.3.2 Перед началом проведения работ следует проверить комплектность КТСИ и провести его внешний осмотр в следующей последовательности:

- проверить целостность соединительных кабелей и проводников;
- проверить состояние соединителей;
- осмотреть поверхности и разъемы корпусов модулей.

При этом, наружные поверхности не должны иметь вмятин, трещин, царапин, дефектов покрытия и загрязнений, влияющих на работу модулей.

1.4 Подключение модулей КТСИ

1.4.1 Подключение модулей КТСИ производить, руководствуясь положениями п.2.4

документа «Руководство по эксплуатации» на КТСИ.

1.4.2 Подать напряжение питания 24 В постоянного тока на общую электрическую шину. Питание подаётся через процессорный модуль (рисунки 1).

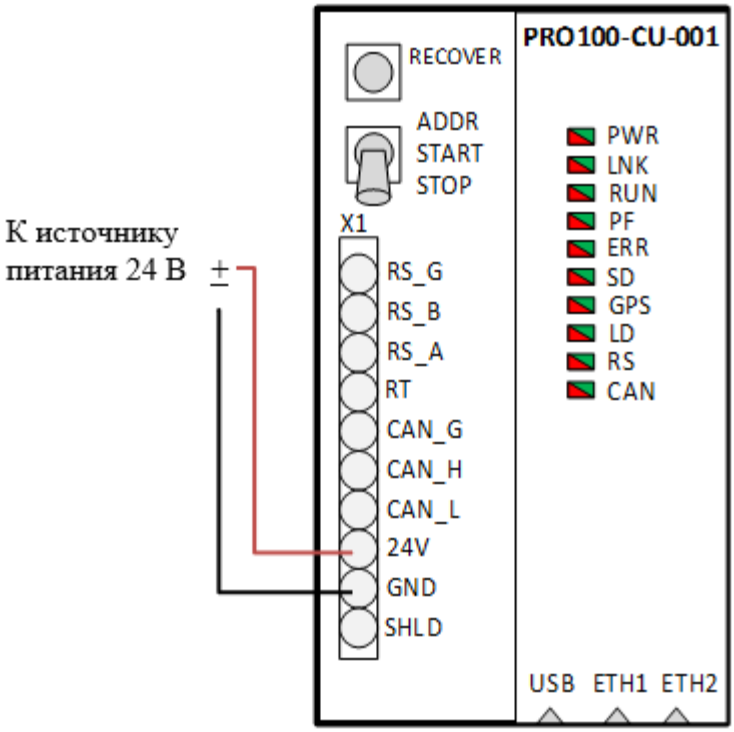


Рисунок 1 – Подача напряжения питания на общую электрическую шину через процессорный модуль UZOLA PRO100

1.4.3 Подключить МП к компьютеру по интерфейсу Ethernet (см. рисунок 1).

1.4.4 Произвести сопряжение МВВ с МП.

1.5 Указания по включению и работе

1.5.1 Перед включением КТСИ следует убедиться в правильности и полноте выполнения подключений в соответствии с пунктом 1.4.

1.5.2 Включение КТСИ производить, руководствуясь положениями п.2.5 документа «Руководство по эксплуатации» на КТСИ.

1.5.3 Управление КТСИ осуществляется при помощи инструментальной системы MasterSCADA 4D через компьютер, подключенный к процессорному модулю через канал связи Ethernet.

1.6 Порядок выключения и демонтажа после окончания работ

1.6.1 Выключение и демонтаж КТСИ производить, руководствуясь положениями п.2.6 документа «Руководство по эксплуатации» на КТСИ.

2 Установка и настройка среды разработки и исполнения «MasterSCADA 4D»

2.1 Состав MasterSCADA 4D

Среда разработки и исполнения MasterSCADA 4D (здесь и далее MasterSCADA 4D) включает в себя следующие части:

- инструментальная система/среда разработки (DT);
- исполнительная система, состоящая из набора модулей (RT);
- клиент визуализации (HMI-клиент).

Исполнительная система поставляется уже установленной комплектно с процессорным модулем PRO-100-CU и сконфигурированной лицензией.

В инструментальной системе создается проект, который впоследствии запускается в реальном времени под управлением исполнительных модулей, доступ к графической части осуществляется при помощи клиента визуализации.

2.1.1 Инструментальная среда является приложением для Windows.

Общий вид инструментальной системы показан на рисунке 2.

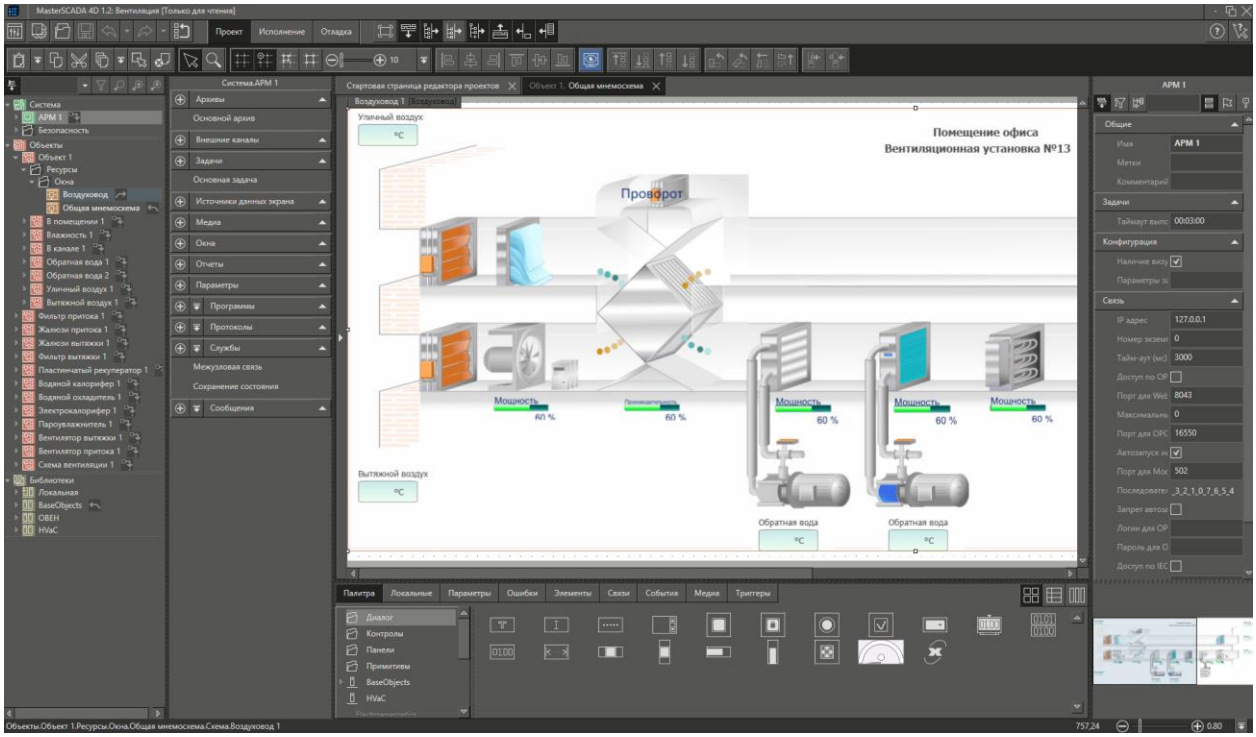


Рисунок 2 – Общий вид инструментальной системы

Проект создается в рамках единой инструментальной системы. Никаких других инструментов или редакторов не требуется. Данные, введенные в системе один раз, становятся доступны для любого элемента проекта.

Основным инструментом для создания элементов проекта является контекстное меню, а для настройки связей между элементами - механизм drag-and-drop. Среда имеет

большой набор инструментов для тиражирования готовых решений, автоматизации типовых операций, а также для отладки проектов в режимах эмуляции (на локальном ПК) и реального исполнения (с загрузкой в контроллер).

2.1.2 В исполнительной системе происходит исполнение созданного проекта.

- программируемые логические контроллеры (ПЛК);
- локальные панели управления (HMI-панели);
- рабочие места операторов (АРМ);
- архивные и технологические серверы;
- облачные сервисы.

Для этих элементов систем управления в MasterSCADA 4D имеются специализированные исполнительные системы (runtime). Они инсталлируются на данные устройства (в контроллер PRO-100-CU предустановлен за заводе-изготовителе), и по команде инструментальной системы загружают и запускают на исполнение предназначенные для них части проекта.

Важно! Запуск проекта под управлением исполнительной системы может быть совершен автоматически и без использования инструментальной системы при старте данного устройства.

В MasterSCADA в зависимости от количества получаемых и передаваемых данных, от функций, заложенных в проект, выбирается коммерческая версия среды разработки.

2.1.3 Для представления графической информации в MasterSCADA 4D используется стандарт HTML5. Для его реализации в состав серверной части большинства исполнительных модулей входит WEB-сервер. Он формирует страницы формата HTML5, которые отображаются в клиенте визуализации MasterSCADA 4D. В качестве клиента визуализации можно использовать любое устройство, имеющее в своем составе современный браузер. Это могут быть не только компьютеры, но и операторские панели, смартфоны, планшеты. С любого такого устройства можно подключиться к контроллеру и получать доступ к той же информации, которая предоставлена оператору на локальном АРМ. Естественно, это возможно только, если был открыт доступ к данной информации или управлению.

2.2 Системные требования среды разработки MasterSCADA 4D

2.2.1 Минимальные системные требования:

- операционная система (ОС) – Windows 10 SP1 x64 (операционная система должна поддерживать русский язык);
- процессор – Intel® Core™ i3, 2.3 ГГц;

- ОЗУ – 8 Гб;

- дисплей – 1280x1024

- жесткий диск - 10 Гб свободного пространства

- клавиатура, мышь.

Важно! При соблюдении минимальных системных требований среда разработки MasterSCADA 4D может быть установлена, но при разработке проектов может наблюдаться торможение при переключении между вкладками, при работе с деревьями и панелями редактора. Время выполнения длительных операций, например, конвертации проектов, может быть значительным.

2.2.2 Рекомендуемые системные требования:

- ОС – Windows 10 x64 или старше;

- процессор – современный многоядерный, не ниже Intel® Core™ i5, 3.4 ГГц

- ОЗУ – не менее 16 Гб;

- дискретная видеокарта, с актуальными драйверами (не старше 5 лет)

- дисплей – 1920x1080;

- жесткий диск - SSD

- свободное место на диске – 100 Гб

- клавиатура, мышь.

Важно! Поддерживаются только 64-битные ОС.

Системные требования среды разработки и среды исполнения MasterSCADA 4D связаны с количеством тегов, которое планируется использовать в проекте.

Рекомендуемые системные требования для среды разработки приведены в таблице 1.

Таблица 1

Количество тегов	**CPU**	**RAM**	**SSD/HDD**	**Video**	**OS**
100 - 1000	Intel Core I5 (2600 MHz+) не ниже 10го поколения и выше (либо аналоги)	16 ГБ	128 ГБ SSD	Не интегрированная видеокарта (уровня NVIDIA GeForce GT 1030)	Windows 10, Windows 11, Windows Server 2016 и выше
2500 - 10000	Intel Core I7 (3100 MHz+) не ниже 10го поколения и выше (либо аналоги)	32 ГБ	256 ГБ SSD		
60000 и более	Intel core i9-9900K не ниже 12го / (также рекомендуются более новые модификации Core™ i9-12900K и новее).	64-256 ГБ	512 ГБ SSD		

Рекомендуемые системные требования для среды исполнения приведены в таблице 2.

Таблица 2

Количество тегов	CPU	RAM	SSD/HDD	Video	OS
00 - 1000	Intel Core I5 (2600 MHz+) не ниже 10го поколения и выше (либо аналоги)	16 ГБ	128 ГБ SSD	Не требуется	Windows 10, Windows 11, Windows Server 2016 и выше. Linux (поддерживается всеми версиями RT (Standard,Pro,Enterprise) кроме Lite)): AstraLinux (не ниже 1.7 (Орел, Воронеж, Смоленск)), ALT Linux (не ниже 8 СП), РЕД ОС (не ниже 7.3), Ubunta (не ниже 20.04 LTS)
2500 - 10000	Intel Core I7 (3100 MHz+) не ниже 10го поколения и выше (либо аналоги)	32 ГБ	256 ГБ SSD		
60000 - 150000	Intel core i9-9900K не ниже 12го / (также рекомендуются более новые модификации Core™ i9-12900K и новее).	64-256 ГБ	512 ГБ SSD		
150 000 - 300 000	Процессор уровня: Intel Xeon Gold 6256 или Xeon W	256 ГБ	1 ТБ SSD		
300 000 - 500 000	Процессор уровня: Intel Xeon Gold 6256 или Xeon W	256-512 ГБ			
500 000 - 1 000 000	Процессор уровня: 2x Intel Xeon Gold 6256 и выше	512 ГБ			
1 000 000 - 2 000 000	Процессор уровня: 2x Intel Xeon Gold 6256 и выше	1 ТБ			

Рекомендуемые системные требования для сервера БД приведены в таблице 3.

Таблица 3

Количество тегов	CPU	RAM	SSD/HDD	Video	OS
PostgreSQL	8 ядер (16 логических потоков), частота – 3,5	16 ГБ и более в зависимости от кол-ва	30 ГБ + объем для данных	Стабильный канал связи от 10 Мб/сек	Windows 10, Windows 11, Windows Server 2012 и выше, Linux

Количество тегов	CPU	RAM	SSD/HDD	Video	OS
	ГГц и больше	данных			(Astra/Alt/RED OC)
Microsoft SQL Server 2019 Express и новее	x64 с тактовой частотой 2,0 ГГц или выше (AMD Opteron, AMD Athlon 64, Intel Xeon с поддержкой Intel EM64T, Intel Pentium IV с поддержкой EM64T. и выше)		20 ГБ + объем для данных	Интернет, Стабильный канал связи от 10 Мб/сек	Интернет, Стабильный канал связи от 10 Мб/сек
АВАДС	Intel Core I5 (2600 MHz+) не ниже 10ого поколения и выше (либо аналоги)		1 ГБ + объем для данных	Стабильный канал связи от 10 Мб/сек	Windows 7, Windows 10, Windows 11, Windows Server 2012 и выше, Linux (Astra/Alt/RED OC/Ubuntu/Debian)

2.3 Установка и настройка среды разработки «MasterSCADA 4D»

2.3.1 Установка среды разработки «MasterSCADA 4D» в операционной системе Windows

2.3.1.1 Дистрибутив среды разработки доступен для скачивания на сайте uzola.ru**.** При запуске MasterSCADA4D.exe открывается начальный диалог мастера установки (см. рисунок 3). В диалоге выбирается локализация продукта (поддерживаются русский и английский языки).

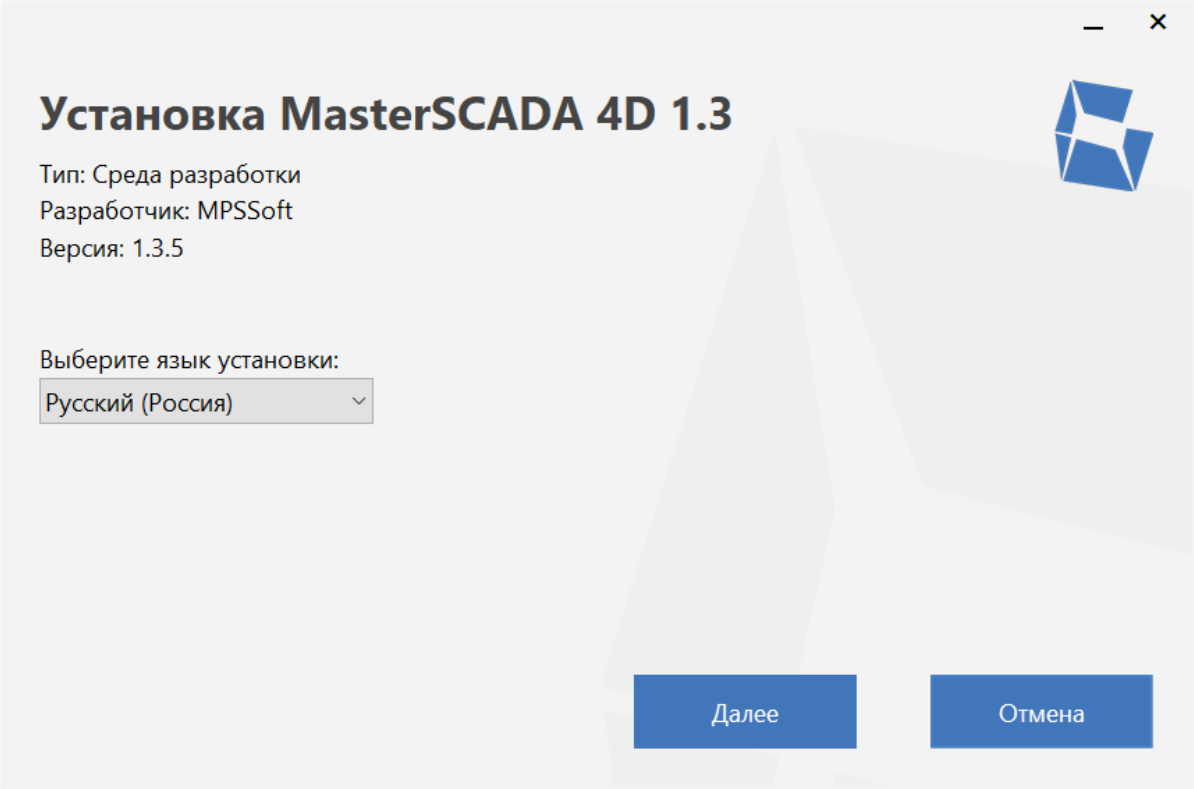


Рисунок 3 – Окно мастера установки

2.3.1.2 По команде «Далее» мастера открывается диалог лицензионного соглашения (см. рисунок 4). Прочитав текст, отметьте пункт «Я принимаю условия лицензионного соглашения» и нажмите кнопку «Далее».

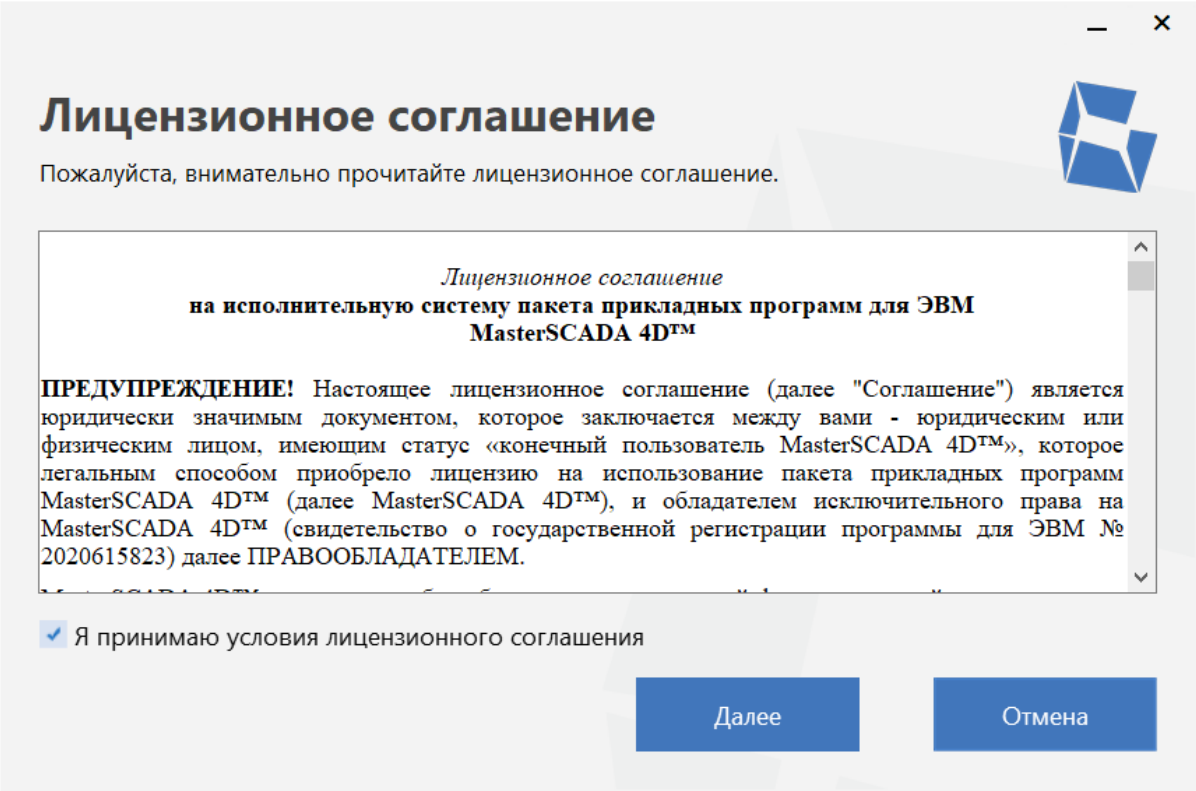


Рисунок 4 – Окно лицензионного соглашения

2.3.1.3 По команде «Далее» мастера открывается окно выбора папки для установки программы (см. рисунок 5). В этом диалоге можно изменить папку установки продукта, воспользовавшись кнопкой «Обзор», а также задать создание ярлыка программы на рабочем столе, отметив пункт «Создать ярлык на рабочем столе».

Важно! При выборе пути установки следует избегать папок, названия которых состоят из кириллических символов.

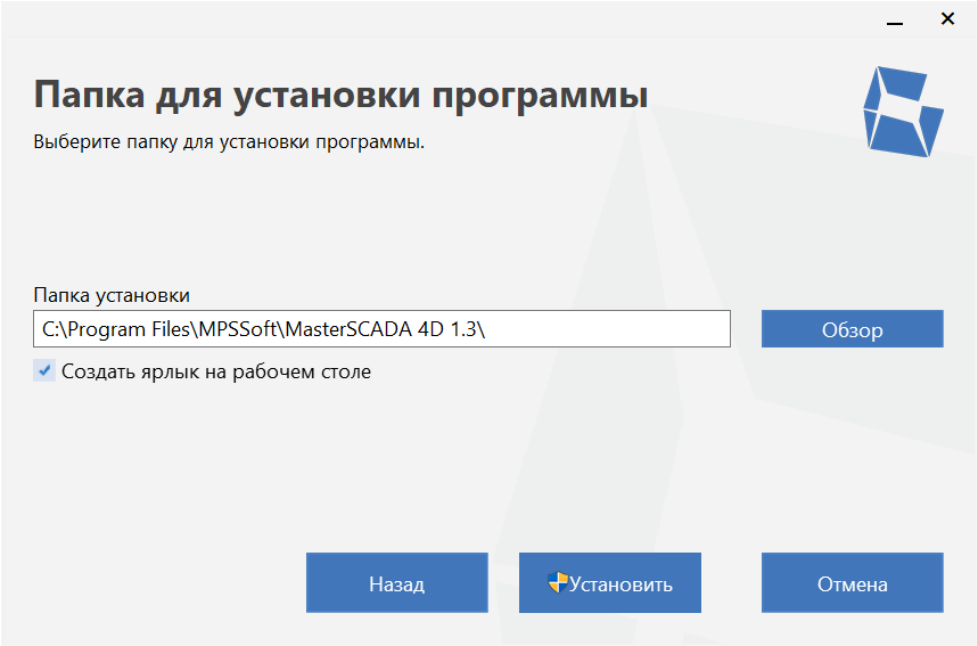


Рисунок 5 – Окно выбора папки для установки программы

2.3.1.4 Нажмите кнопку «Установить», и откроется диалог, отображающий процесс установки (см. рисунок 6).

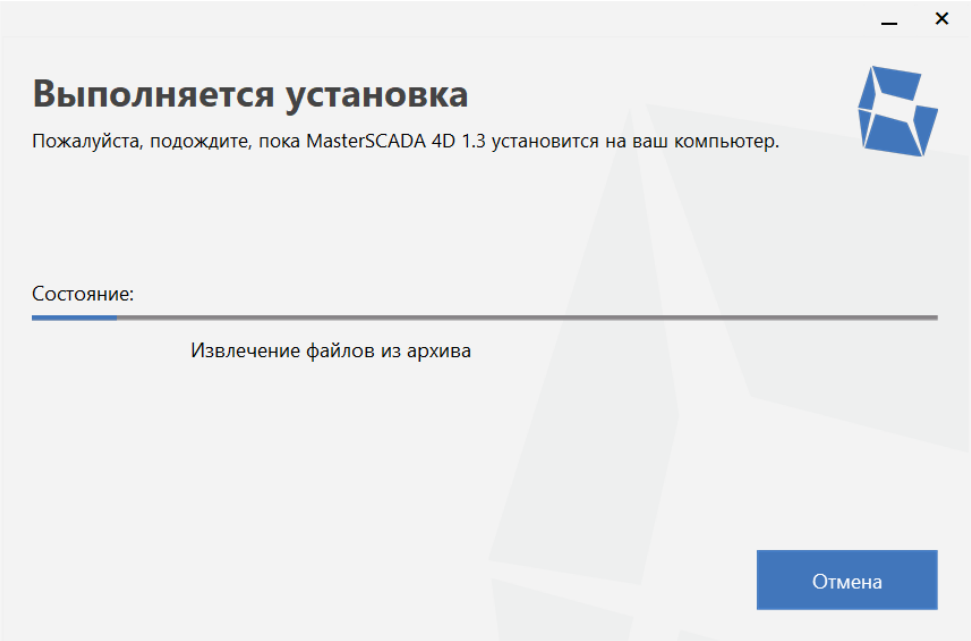


Рисунок 6 – Окно отображения процесса установки

2.3.1.5 Возможно, потребуется дополнительное разрешение на установку программы (см. рисунок 7). Для продолжения установки нужно нажать на кнопку «Да».

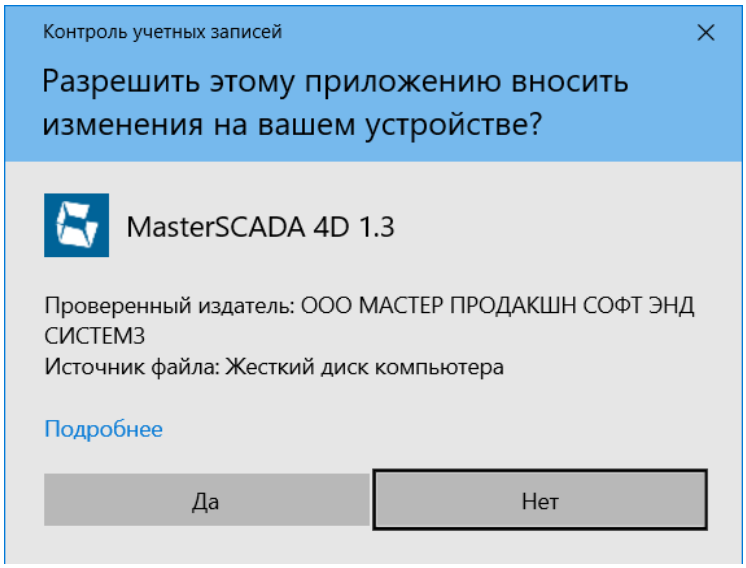


Рисунок 7 – Окно дополнительного разрешения на установку

2.3.1.6 Дождитесь завершения установки и появления последнего диалога мастера (см. рисунок 8). Для завершения работы мастера нажмите кнопку «Завершить».

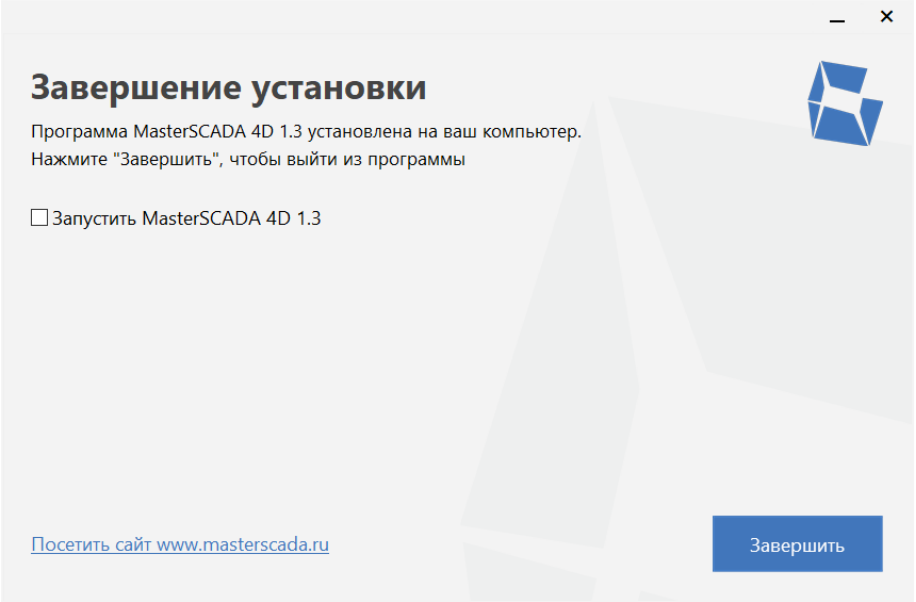


Рисунок 8 – Окно завершения установки

2.3.1.7 Для запуска среды разработки (редактора проекта) следует выбрать в меню Пуск пункт, соответствующий версии MasterSCADA 4D, либо дважды нажать левой кнопкой мыши на ярлык на рабочем столе MasterSCADA 4D [номер версии], либо открыть файл <папка установки продукта>\bin\ms4d.exe в проводнике Windows. При запуске открывается стартовое меню редактора проекта.

2.3.1.8 При старте редактора проекта в директории %профиль пользователя%\AppData\Roaming\MPSSoft\MasterSCADA4D\ создается папка sys_log, содержащая протоколы работы (файлы с именами yy_mm_dd HH_MM_SS.log).

2.3.2 Установка среды разработки «MasterSCADA 4D» в операционной среде Linux

Для установки среды разработки в Linux необходима версия операционной системы AstraLinux 1.7/1.8.

Установка производится в несколько этапов:

- Установка Postgre (не ниже 14 версии);
- Установка MS4D 2.0.x;
- Установка Firebird сервера.

2.3.2.1 Установка Postgre

Чтобы установить СУБД PostgreSQL в Astra Linux (мы используем в основном версию 1.7 и 1.8), нужно выполнить одно из следующих действий:

1 Ввести в "Менеджере пакетов Synaptics" postgresql и выбрать одно из приложений (в 1.8 предлагается версия 15). Хотя этот вариант и максимально прост, для дальнейшей настройки все равно нужно обращаться к командной строке.

2 Установить postgresql вручную, вводя команды в терминале. Далее показаны все шаги такой установки и настройки.

До загрузки PostgreSQL обновляем списки пакетов:

$ sudo apt update

Загрузим PostgreSQL с утилитой postgresql-contrib:

$ sudo apt install postgresql postgresql-contrib

Загрузятся драйверы PostgreSQL последней версии (обратите внимание: не самой последней версии, доступной у разработчика СУБД, а последней версии, поддерживаемой Вашей версией операционной системы Linux и хранящейся в её репозиторях. Чтобы СУБД начала работать, необходимо запустить её как сервис:

$ sudo systemctl start postgresql.service

По умолчанию автоматический запуск сервера отключён, чтобы можно было вручную управлять восстановлением базы данных после перезагрузки системы. Для реального сервера, естественно, автозапуск должен быть включен.

Проверить статус сервиса:

$ sudo systemctl status postgresql.service

В ответ на эту команду система запросит пароль суперпользователя, а получив его - выведет сообщение:

● *postgresql.service - PostgreSQL RDBMS Loaded: loaded*

(/lib/systemd/system/postgresql.service; enabled; vendor pr> Active: active (exited)

since Wed 2024-11-06 08:12:35 MSK; 11h ago Main PID: 2047 (code=exited, status=0/SUCCESS) CPU: 1ms

ноя 06 08:12:35 m40-A320M-H systemd[1]: Starting PostgreSQL RDBMS...

ноя 06 08:12:35 m40-A320M-H systemd[1]: Finished PostgreSQL RDBMS.

Чтобы вернуться в терминал, необходимо нажать Ctrl+Z, это переведёт команду в фоновый режим, но не остановит её выполнение.

PostgreSQL для обозначения аккаунта применяет термин «Роль». В Linux-системе роли сервиса привязываются к одноименным аккаунтам. При установке PostgreSQL в системе создаётся аккаунт postgres, получающий в СУБД максимальные права (в СУБД, но не в операционной системе!). Этот аккаунт нужен для того, чтобы создавать пользователей и новые базы данных и связывать их. Самый важный принцип работы в PostgreSQL: один пользователь - одна база данных (и его можно нарушить - но только в том случае, если вы осознаёте, чем это грозит, и принимаете на себя все риски).

Прежде чем приступить к созданию и настройке пользователя, который будет работать с базой данных, нужно обратить внимание на то, что все вводимые в консоли команды будут выполняться от имени текущего пользователя (его имя вы увидите в самом начале приглашения к вводу терминала). Предположим, что вы работаете за компьютером *kab1805* от имени пользователя *kab18-05*, тогда приглашение к вводу будет выглядеть следующим образом:

kab18-05@kab1805:~$

Примечание - Необходимо обращать внимание на то, от чьего имени запускается та или иная команда, это имеет большое значение.

Прежде всего нужно переключиться на пользователя postgres (созданного при установке СУБД):

kab18-05@kab1805:~$ sudo -i -u postgres

После этого нужно запустить PostgreSQL и создадать пользователя, попутно задав ему пароль "1234" (обращаем внимание на одинарные кавычки в команде в терминале):

postgres@kab1805:~$psql

postgres=#CREATE USER test WITH PASSWORD '1234';

Выходим из СУБД:

postgres-#\q

Примечание - в качестве примера мы создаем пользователя test, вам нужно создавать собственного пользователя, при этом пароль должен быть достаточно сильным).

Далее, не выходя из учётной записи администратора PostgreSQL (postgres), необходимо создать базу данных (название базы данных, в нашем примере "test", должно

совпадать с именем пользователя, созданного в предыдущем шаге, иначе вы не сможете назначить права на эту базу данных своему пользователю):

postgres@kab1805:~$createdb test

Снова войти в СУБД и дать пользователю test все права на базу данных test:

postgres@kab1805:~$psql

postgres=#GRANT ALL PRIVILEGES ON DATABASE test TO test;

postgres=#\q

В 14 версии Postgre этого было достаточно, а вот в 15 версии попытка создать таблицу в базе данных test пользователем test, который имеет все права на неё, выдаст ошибку:

CREATE TABLE json_data_table (data jsonb);

ERROR: permission denied for schema public

LINE 1: CREATE TABLE json_data_table

(^

Поиск причины такого поведения и способа его исправления показал, что если первый пользователь создается для новой базы данных, то его нужно назначить также и владельцем этой базы данных (выполняется от суперпользователя postgres):

postgres=# ALTER DATABASE dg_name OWNER TO user_name;

Добавление второго пользователя с полными правами не приветствуется, однако возможно сделать и это, перебрав все типы разрешённых операций для всех типов объектов базы данных.

Выйти из учётной записи администратора СУБД:

[postgres@kab1805]:~$exit

Войти под учётной записью пользователя test в базу данных test:

kab18-05@kab1805:~$psql -U test -h 127.0.0.1 -p 5432

В следующий раз, для входа в свою базу данных, необходимо использовать эту строчку, подставив в неё имя своего пользователя, соответствующая база данных будет выбрана автоматически.

Узнать статус подключения к базе данных можно с помощью команды:

postgres=# \conninfo

Выведется следующее сообщение (от версии к версии эта надпись может меняться):

You are connected to database " tester " as user " tester " via socket in "/var/run/postgresql" at port "5432"

Впоследствии можно подключиться к базе данных разными способами. Самым простым является запуск терминальной программы psql:

kab18-05@kab1805:~$ psql test

Эта команда подключит базу данных test, если это позволено пользователю СУБД, одноименному текущему пользователю операционной системы. То есть, если в СУБД создан пользователь kab18-05 и ему предоставлены все разрешения на базу данных test, эта команда будет выполнена. В ином случае будет выдано сообщение:

psql: error: connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed:

FATAL: role "kab18-05" does not exist

Работая с СУБД от имени текущего пользователя операционной системы, мы можем захотеть подключиться к одноименной базе данных (удобно, если пользователь работает только с одной базой данных). Для этого достаточно ввести команду psql без аргументов:

kab18-05@kab1805~:$ psql

Символ # в конце приглашения к вводу psql обозначает, что вы являетесь суперпользователем, на которого не распространяются никакие ограничения доступа. Такое бывает, когда вы устанавливали в систему экземпляр PosqgreSQL от имени своего пользователя. Символ > говорит о том, что СУБД была запущена от непривилегированного пользователя.

Теперь, необходимо убедиться, что всё настроено и работает правильно. Для этого после приглашения (# или >) необходимо ввести такие команды:

test=> SELECT version();

test=> SELECT current_date;

test=> SELECT 2+2;

Для выхода из режима просмотра информации о системе (или подобных режимов) нужно нажать клавишу q.

2.3.2.2 Установка MS4D 2.0.x

Дистрибутив deb пакета должен быть скопирован на устройство. Далее в консоли ввести команду:

sudo apt-get install <path_to_deb>

Или запустить двойным кликом мыши (см. рисунок 9):

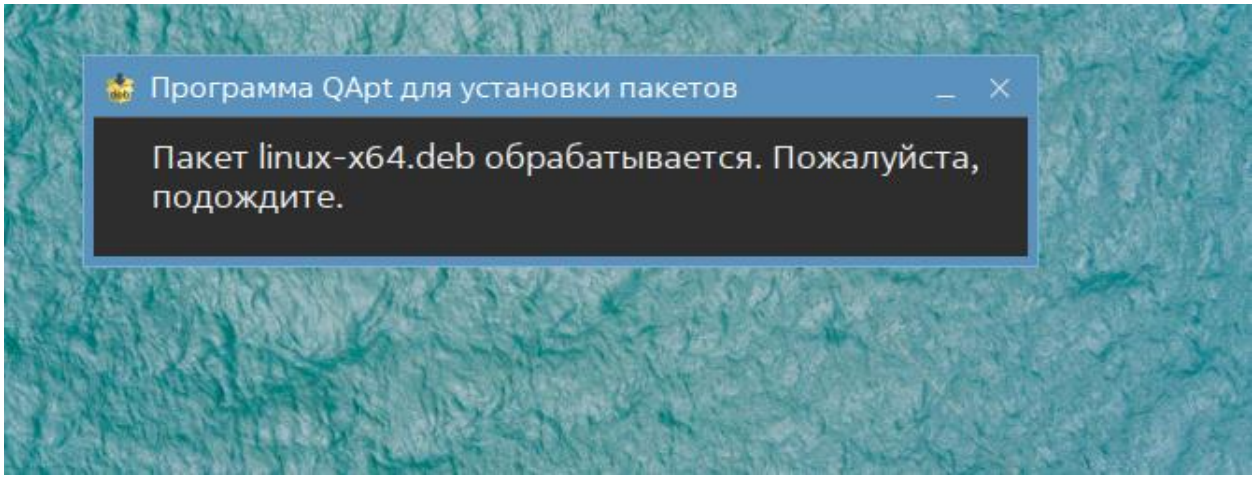


Рисунок 9

Затем ввести пароль суперпользователя и дождаться установки (см. рисунок 10).

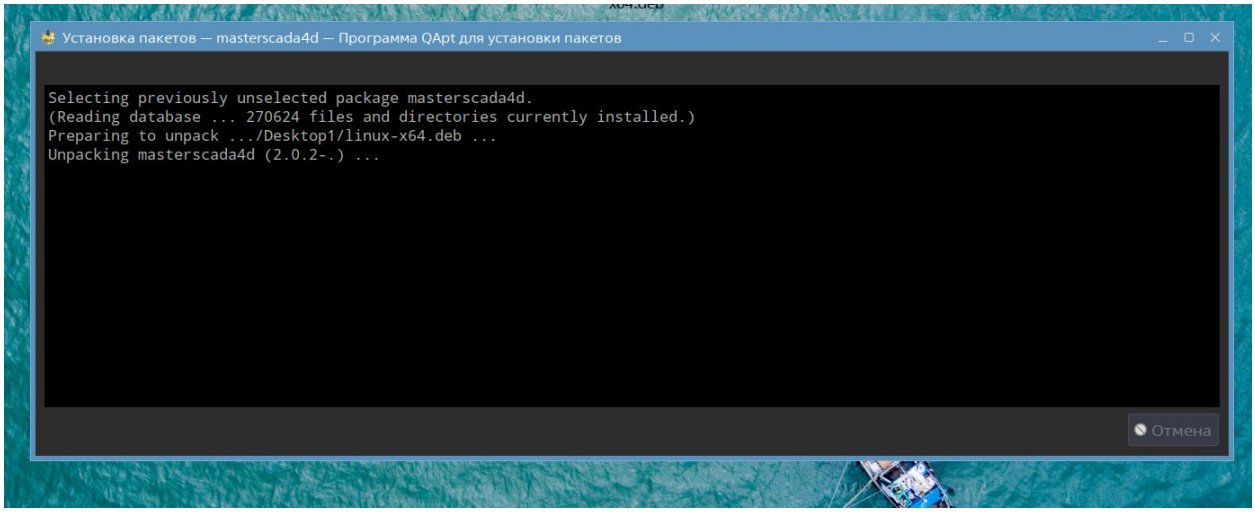


Рисунок 10

После установки в меню пуск «Разработка» появится пункт «MasterScada 4D» (см. рисунок 11).

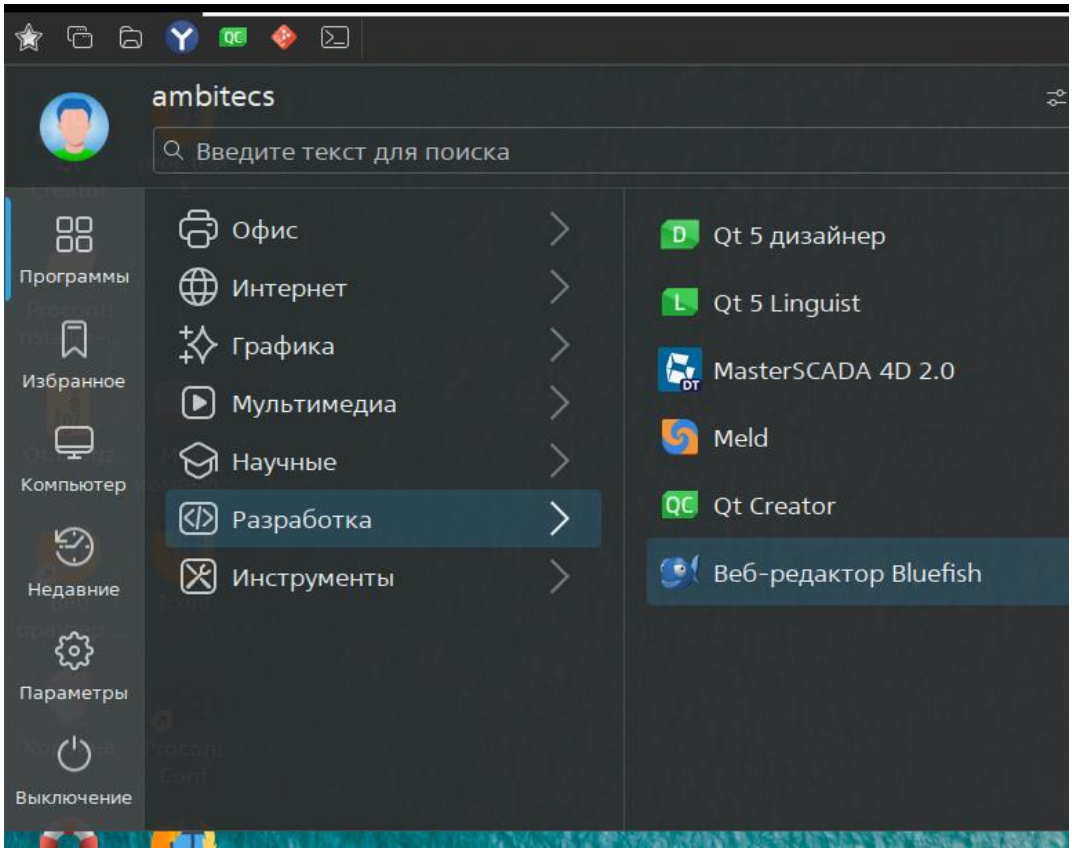


Рисунок 11

2.3.2.3 Установка Firebird сервера

Запустить терминал и запустить скрипт:

sudo /opt/MasterSCADA\\ 4D\\ <версия>/resources/Firebird/firebird-install.sh

Вместо <версия> указать конкретную версию установленной системы.

В процессе установки deb-пакета или выполнения скрипта *firebird-install.sh* может появиться ошибка библиотеки *libncurses5*.

В этом случае необходимо отредактировать файл */etc/apt/sources.list*, раскомментировать в нем репозитории, после чего выполнить команду "sudo apt update" (у машины должен быть доступ в Интернет в этот момент) и заново запустить скрипт *firebird-install.sh*.

На этапе установки, в котором необходимо задать пароль для пользователя SYSDBA сервера Firebird, необходимо ввести пароль "masterkey".

На одном из этапов установки Firebird понадобится через пробел перечислить всех пользователей системы, для которых планируется работа в редакторе MasterSCADA4D.

При запуске среды разработки нужно настроить подключение к базе данных нажав на кнопку «Настройка» (см. рисунок 12):

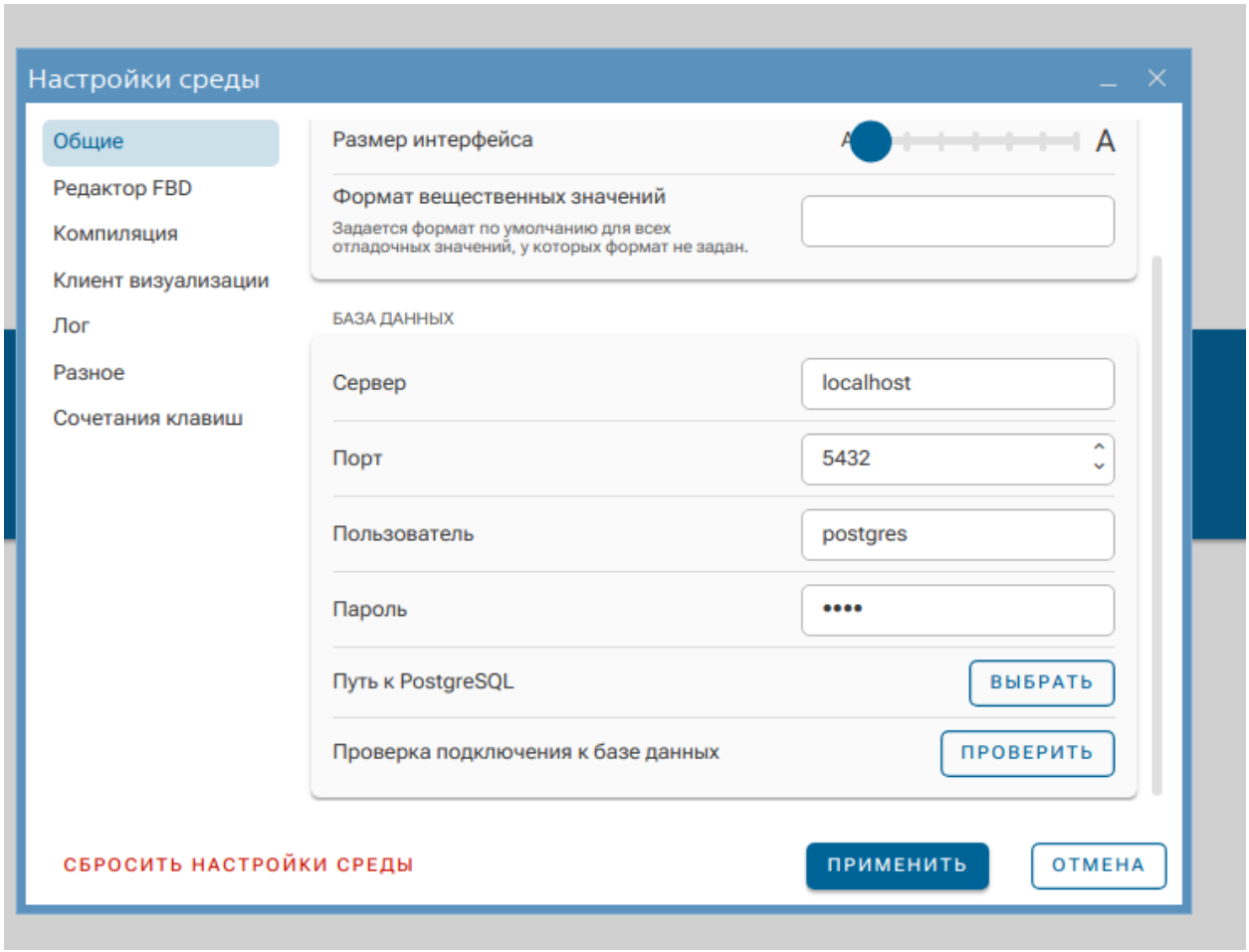


Рисунок 12

Указать пользователя, пароль, и проверить подключение к базе данных.

Открытие существующих проектов:

1 Если проекты хранятся в БД Postgre, к которой возможно подключиться из AstraLinux - достаточно указать актуальные данные подключения в настройках приложения. Проекты будут отображены в списке при выборе "Открыть" на стартовом экране.

2 В остальных случаях - открыть проект в редакторе MS4D на Windows, выполнить команду "Проект -> Экспортировать". В диалоговом окне если доступен выбор типа хранилища - рекомендуется выбрать "JSON". Сформированный zip-файл любым способом перенести в AstraLinux. На стартовом экране приложения выбрать "Импортировать", в диалоговом окне найти и выбрать zip-файл экспортированного проекта. Проект будет импортирован в БД Postgre с настройками подключения, заданными в настройках среды. Открытие проекта через кнопку "Открыть" на стартовом экране.

Возможные причины ошибки при импорте проекта:

- версия PostgreSQL для импорта ниже или выше той, в которой был сохранен проект;

- некорректные данные подключения к БД или отсутствие соедиения с БД.

Ошибка при открытии проекта "Exception while reading from stream":

- воспроизводится в конкретной версии PostgreSQL (например в 14.13), необходимо поставить другую версию PostgreSQL.

2.4 Настройка среды разработки «MasterSCADA 4D»

2.4.1 Для разработки проектов в MasterSCADA 4D для КТСИ Пролог необходимо при создании пользовательского проекта подключить библиотеку Uzola (см. рисунок 13).

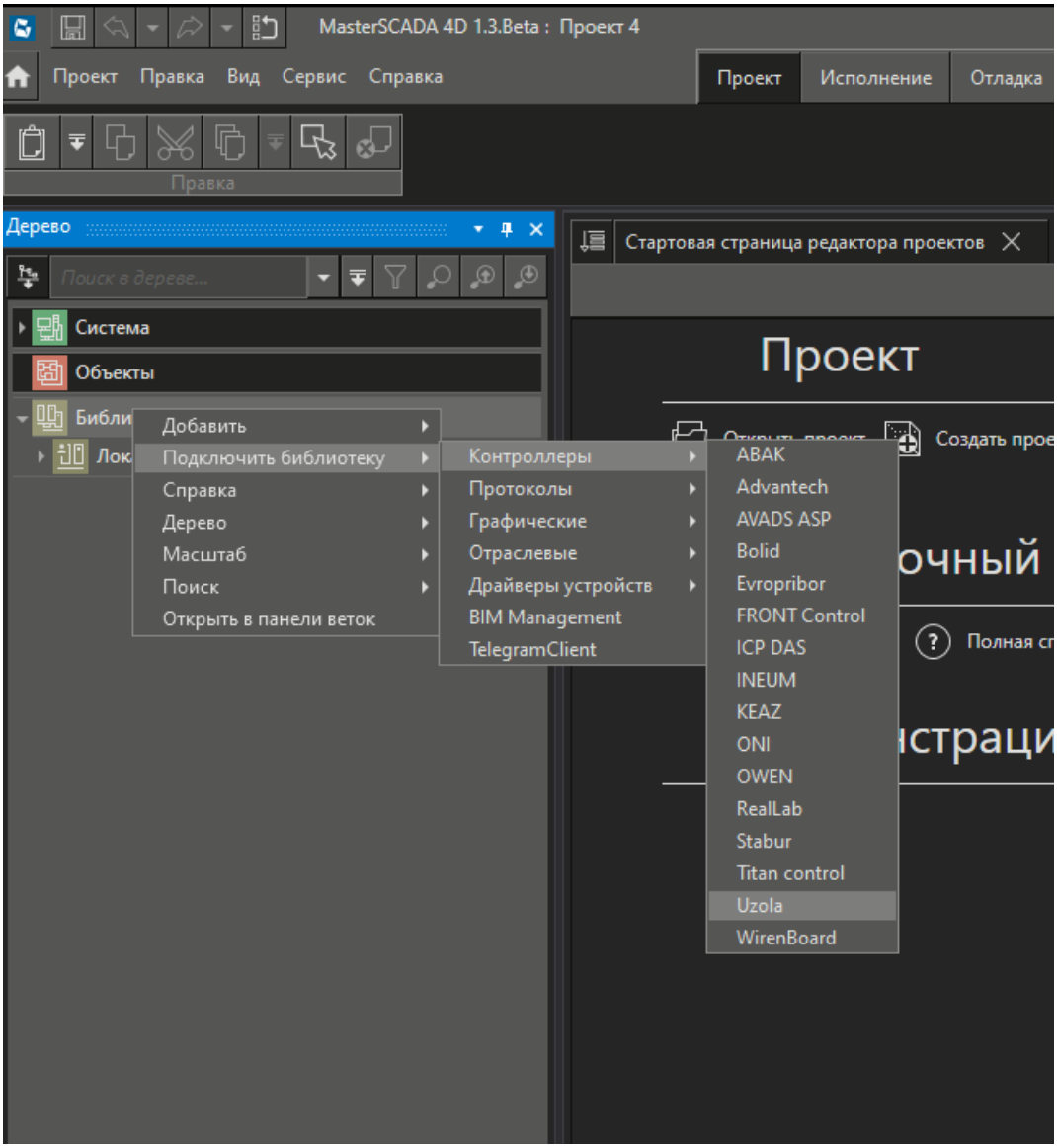


Рисунок 13

2.4.2 После добавления библиотеки возможно добавление контроллера в дерево системы (см. рисунок 14).

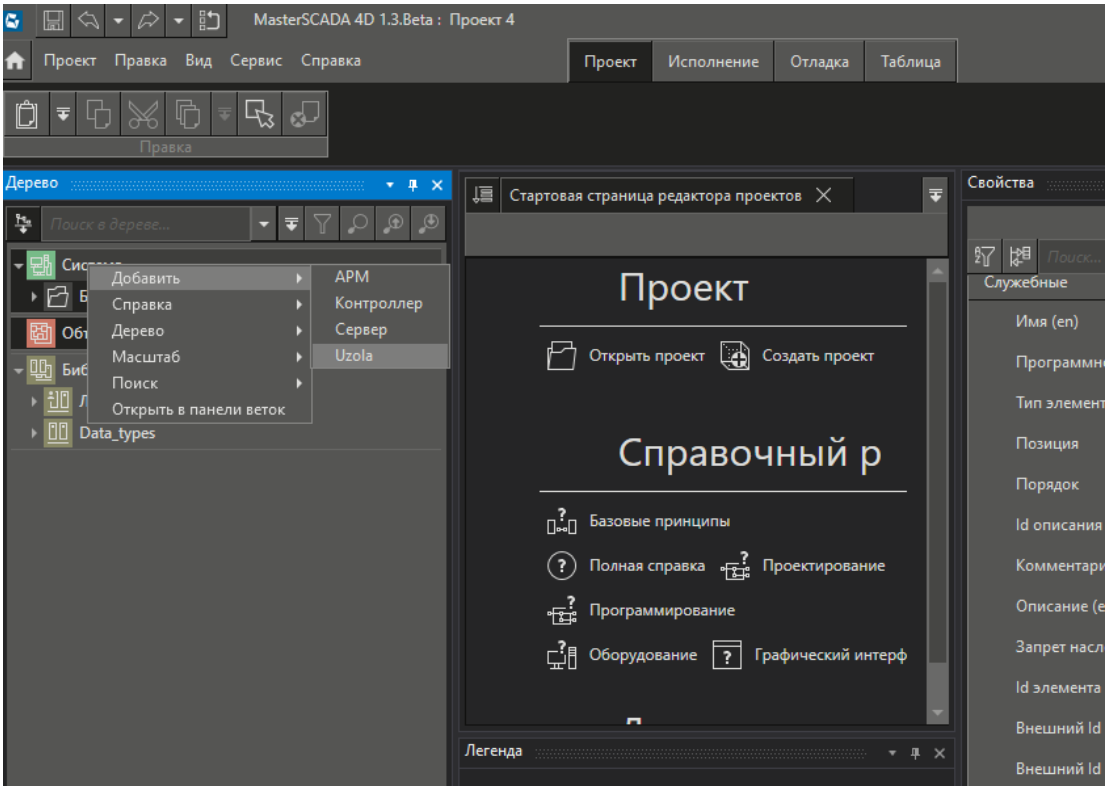

MasterSCADA 4D 1.3.Beta : Проект 4
Проект
Правка
Вид
Сервис
Справка
Проект
Исполнение
Отладка
Таблица
Правка
Дерево
Поиск в дереве...
Добавить
Справка
Дерево
Масштаб
Поиск
Открыть в панели веток
АРМ
Контроллер
Сервер
Uzola
Data_types
Стартовая страница редактора проектов
Проект
Открыть проект
Создать проект
Справочный р
Базовые принципы
Полная справка
Проектирование
Программирование
Оборудование
Графический интерф
Легенда
Свойства
Служебные
Имя (en)
Позиция
Порядок
Id описания
Id элемента
Внешний Id
Внешний Id

Рисунок 14

3 Программирование в инструментальной системе «MasterSCADA 4D»

3.1 Создание проекта и настройка процессорного модуля

Основной порядок работы с контроллером Uzola PRO100 в MasterSCADA 4D предполагает последовательное выполнение следующих шагов:

- подключение библиотеки с узлом;
- добавление узла;
- настройка узла;
- подключение к узлу.

3.1.1 Для реализации соединения с КТСИ сначала в MasterSCADA 4D нужно подключить библиотеку «Uzola».

Данная библиотека содержит элементы, необходимые для конфигурирования обмена данными с КТСИ Uzola. С техническими возможностями этого оборудования можно ознакомиться на сайте производителя.

Для подключения библиотеки нужно перейти в дерево библиотек и вызвать контекстное меню правой кнопкой мыши по элементу «Библиотеки». Затем выполнить «Подключить библиотеку» → «Контроллеры» → «Uzola» (см. рисунок 15).

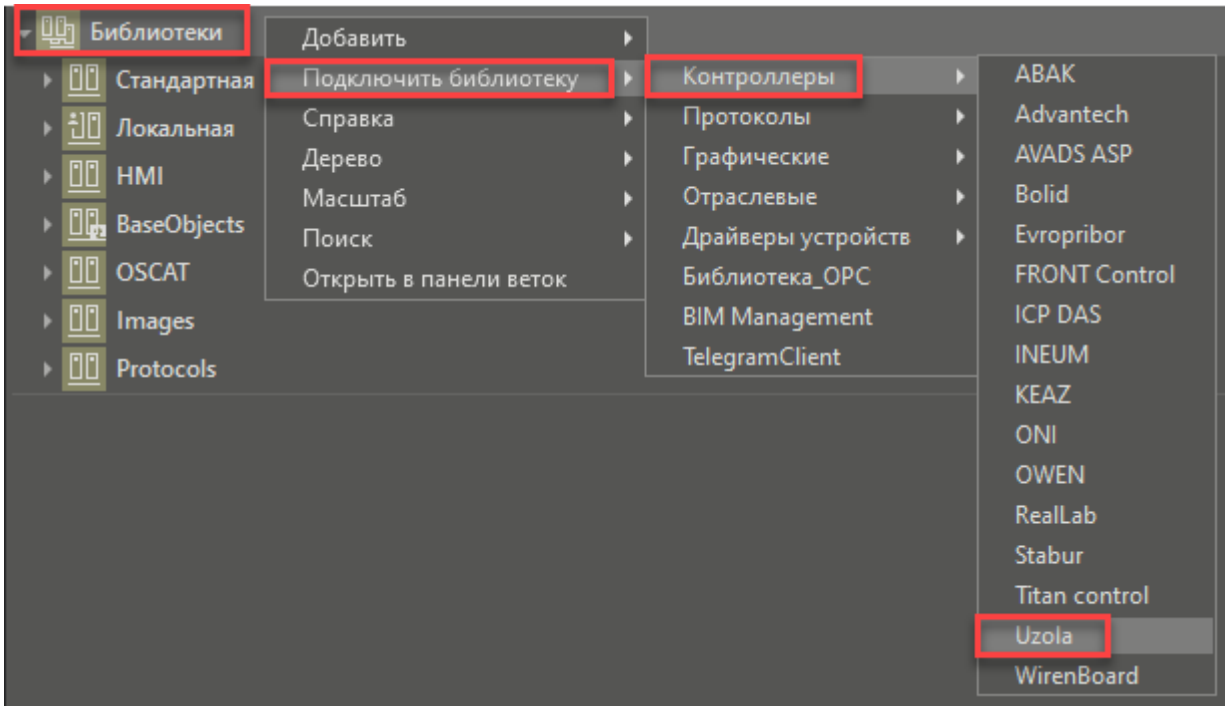


Рисунок 15

После этого библиотека Uzola отобразится в полном Дереве библиотек.

Библиотека Uzola содержит группы (см. рисунок 16):

- Узлы;
- Протоколы;

- Модули ввода-вывода.

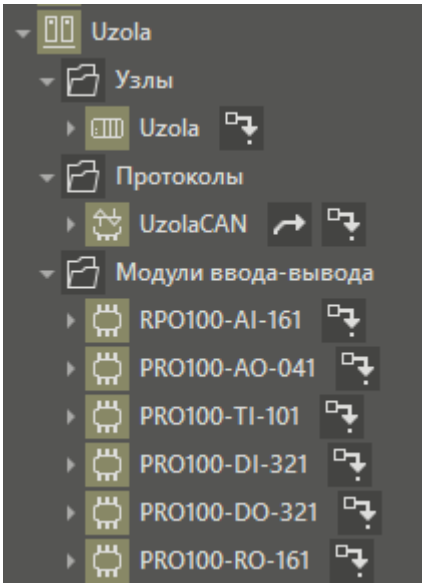


Рисунок 16

Группа «Узлы» библиотеки Uzola содержит конфигурацию контроллера Uzola PRO100.

Группа «Протоколы» содержит специализированный протокол UzolaCAN для подключения модулей ввода-вывода в КТСИ Uzola.

Группа «Модули ввода-вывода» содержит конфигурации специализированных модулей КТСИ Uzola.

После подключения библиотеки контроллер станет доступен для добавления в Дерево системы.

3.1.2 Перед тем как запрограммировать контроллер Uzola PRO100 средствами MasterSCADA 4D, следует добавить его в «Дерево системы». Для этого используется пункт «Добавить» контекстного меню «Системы» (см. рисунок 17).

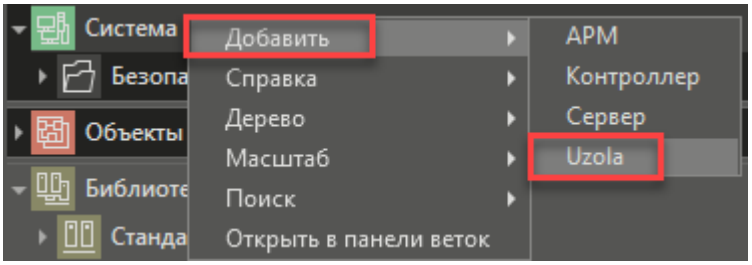


Рисунок 17

3.1.3 После добавления КТСИ Uzola в дерево системы следует его настроить. Для этого перейдите в «Панель свойств Узола».

3.1.4 Для подключения к контроллеру среды разработки достаточно в разделе «Связь» настроить два свойства (см. рисунок 18):

• IP адрес – задается IP-адрес узла, на котором установлена среда исполнения и на

которое должен загрузиться разрабатываемый проект;

• Порт для Web-сервера – если клиент визуализации запускается из среды разработки, то данный порт будет использоваться в адресной строке. Задается в случае указания нескольких адресов связи для узла или использования резервирования узлов. При резервировании по заданному порту клиент/браузер переключается на текущий MASTER.

Важно! Данная настройка не влияет на порт, по которому среда исполнения работает с клиентом визуализации. В этом случае порт определяется настройками среды исполнения при установке. По умолчанию для Linux порт – 80.

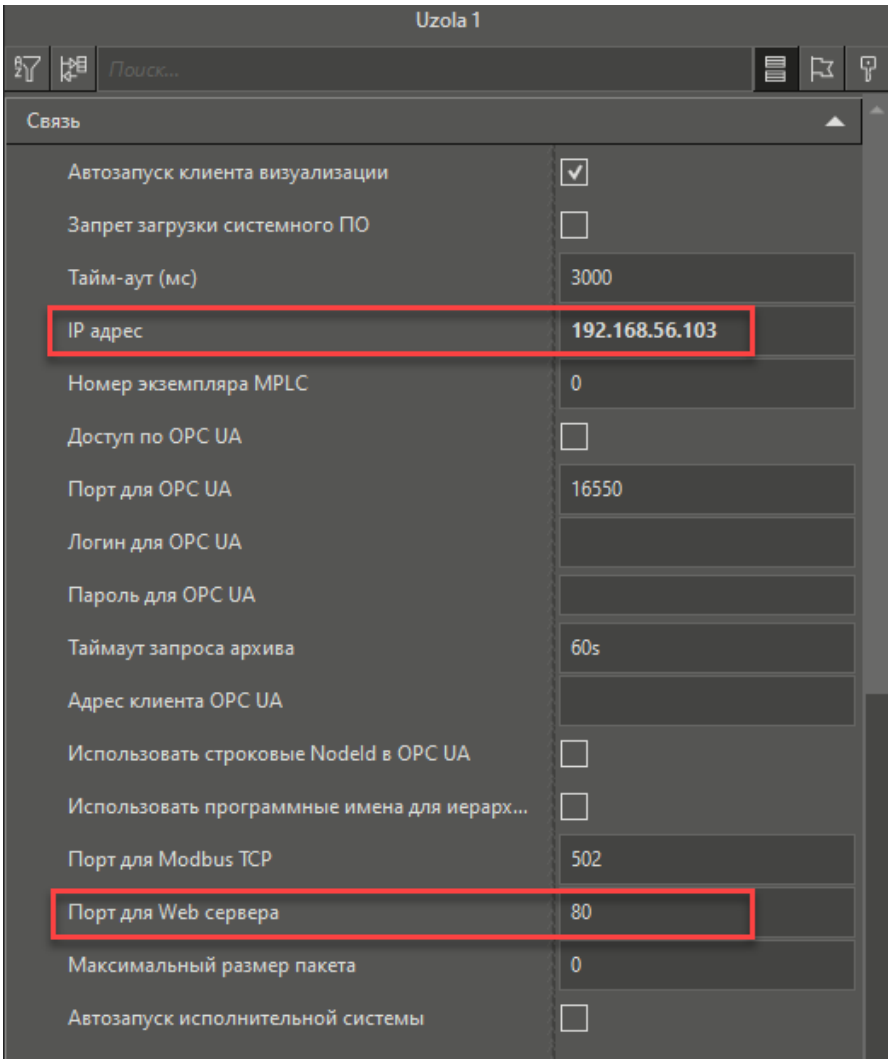


Рисунок 18

3.1.5 Для подключения среды разработки к среде исполнения и для загрузки пользовательского приложения в память контроллера необходимо выделить в дереве контроллер и либо в контекстном меню выполнить «Узел» → «Подключить узел с загрузкой конфигурации», либо нажать на кнопку «Подключить и загрузить» во вкладке инструментов «Исполнение» (см. рисунок 19).

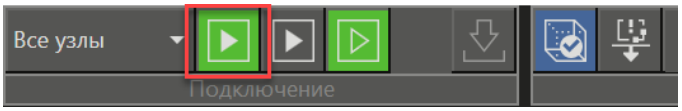


Рисунок 19

После этого проект будет загружен в контроллер и запущен на исполнение.

3.1.6 Для подключения к узлу без загрузки конфигурации проекта нужно выполнить «Узел» → «Подключить узел без загрузки конфигурации», либо нажать на кнопку «Подключить без загрузки» во вкладке инструментов «Исполнение» (см. рисунок 20).

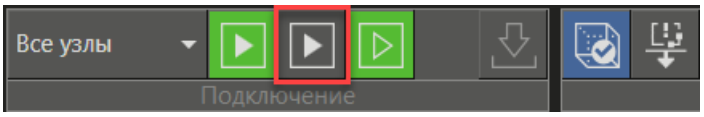


Рисунок 20

3.1.7 После выполнения команд «Подключить» произойдет подключение среды разработки к уже запущенной среде исполнения. Если запущенная среда исполнения не будет найдена, то появится сообщение об ошибке. В этом случае среда разработки не будет пытаться запустить среду исполнения.

3.1.8 Перед использованием контроллера убедитесь, что на нем установлена исполнительная система MasterSCADA 4D.

Для этого стоит обратить внимание на значок, расположенный рядом с добавленным в дерево системы КТСИ.

Если исполнительная система установлена на КТСИ, то при подключении среды разработки к контроллеру рядом с устройством в дереве системы должен появиться значок (галочка), говорящий об успешном подключении (см. рисунок 21).

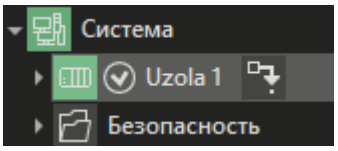


Рисунок 21

Если этого не произошло и открылось диалоговое окно с системными сообщениями об отсутствии соединения с узлом, то убедитесь, что заданный IP-адрес контроллера введен правильно.

Если данные верны, обратитесь в техподдержку производителя контроллера.

3.1.9 После подключения среды разработки к работающей среде исполнения во вкладке «Управление» «Панели Узла» появится возможность получать информацию: о состоянии контроллера, о загруженном в него программном обеспечении, об используемой версии среды исполнения (см. рисунок 22). Для открытия «Панели Узла» нужно дважды кликнуть по узлу в дереве системы.

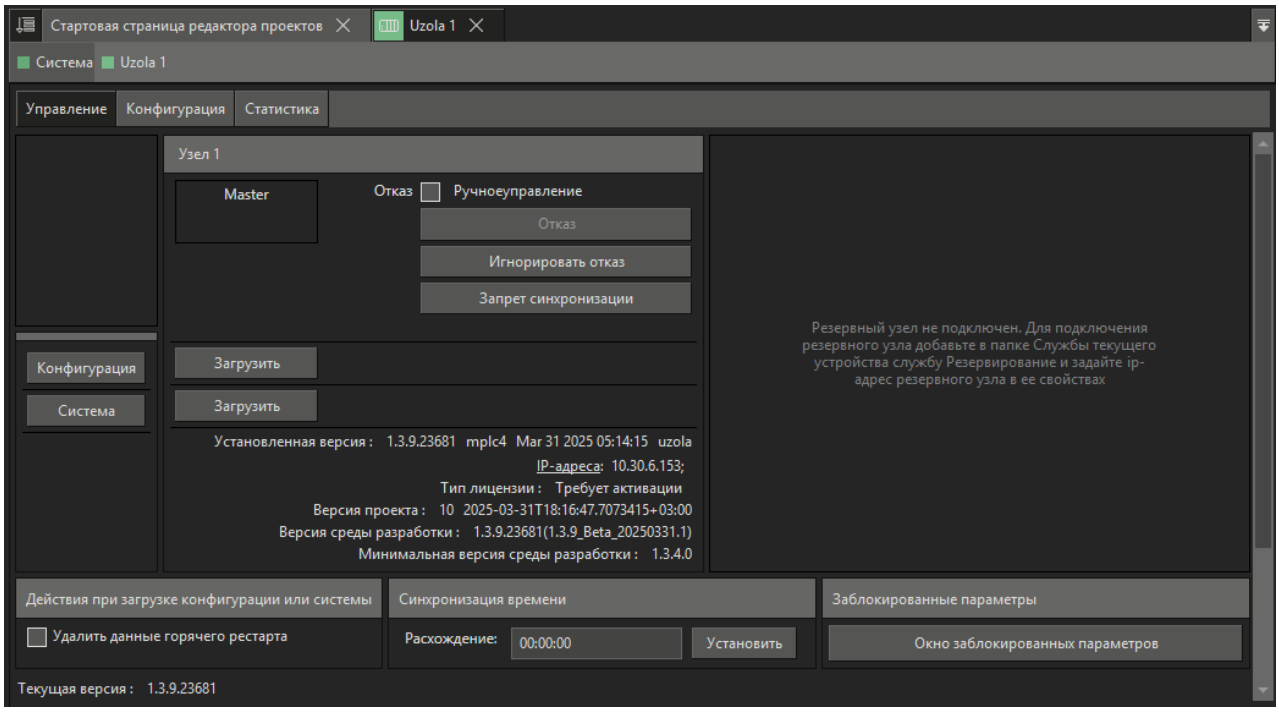


Рисунок 22

3.1.10 При первом подключении к контроллеру в среде разработки возможен вывод системного сообщения: «Исполнительная система устарела/имеет более новую версию» (см. рисунок 23).

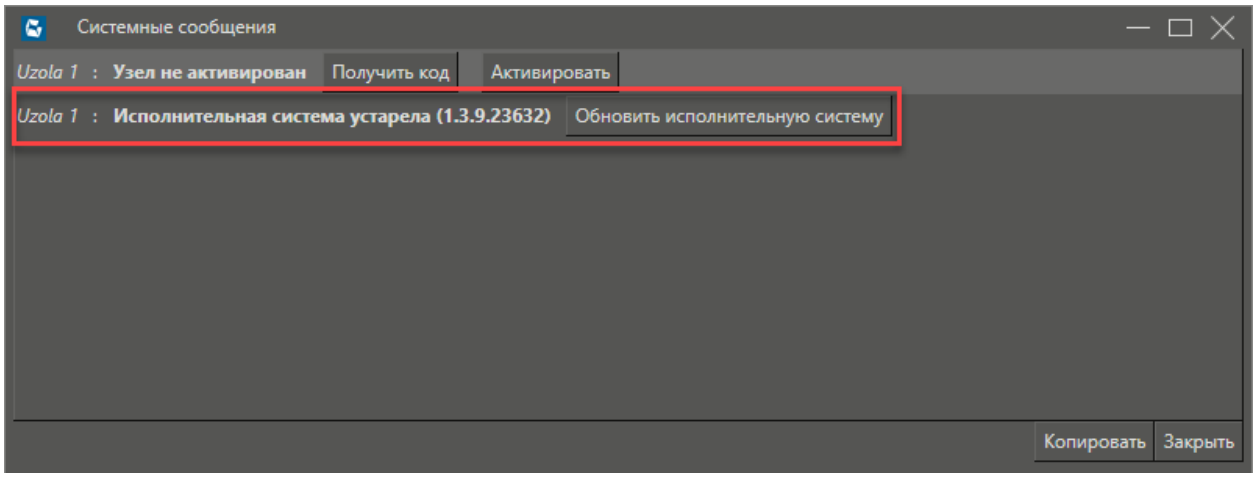


Рисунок 23

Это сообщение выводится, если версия среды разработки отличается от версии среды исполнения. В этом случае необходимо выполнить автоматическое обновление, нажав на кнопку «Обновить исполнительную систему» в диалоге «Системные сообщения».

3.2 Добавление и конфигурирование модулей ввода-вывода

Модули ввода вывода имеют свои настроечные параметры, которые могут хранится в энергонезависимой памяти микроконтроллера модуля. При добавлении в проект контроллера Uzola в дереве проекта есть ветка «Встроенные модули» и ветка

«Конфигуратор».

Ветка «Встроенные модули» предназначена для основной работы, там добавляются модули ввода-вывода в которых происходит опрос текущих состояний каналов ввода-вывода.

В ветке «Конфигуратор» из библиотеки по умолчанию добавляются все модули, которые поддерживаются контроллером Uzola, каждый модуль содержит в себе необходимые для настройки конфигурационные параметры. Для возможности изменения этих параметров пользователю необходимо сконфигурировать ветку конфигуратора в соответствии с реальным наличием и расположением модулей на шине.

Библиотека Uzola включает в себя протокол UzolaCAN и специализированные модули ввода-вывода, которые служат для опроса КТСИ (см. рисунок 24).

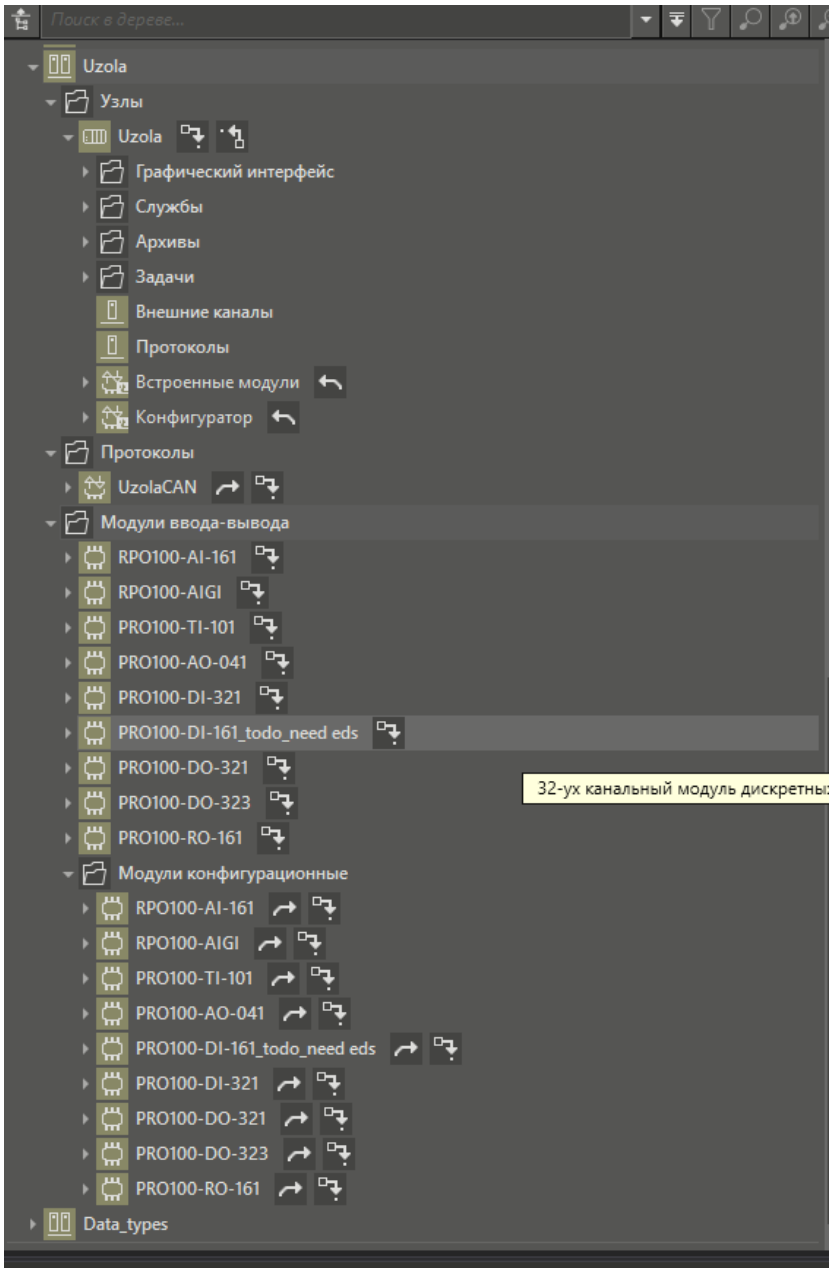


Рисунок 24

Протокол добавляется в узел по умолчанию.

Настройки протокола UzolaCAN в дереве системы задаются в свойствах элемента «Встроенные модули», который добавлен в узел по умолчанию (см. рисунок 25). Панель свойств элемента показана на рисунке 26.

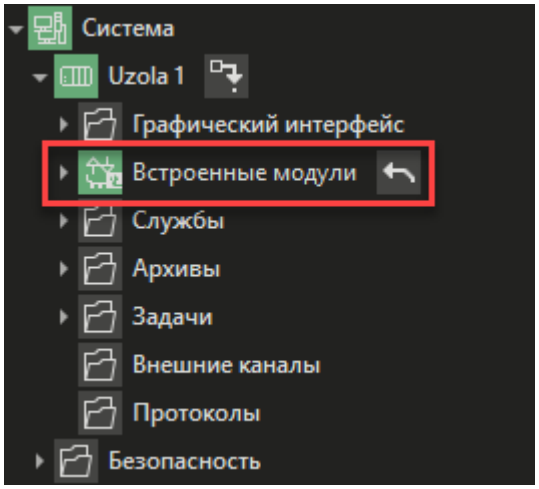


Рисунок 25

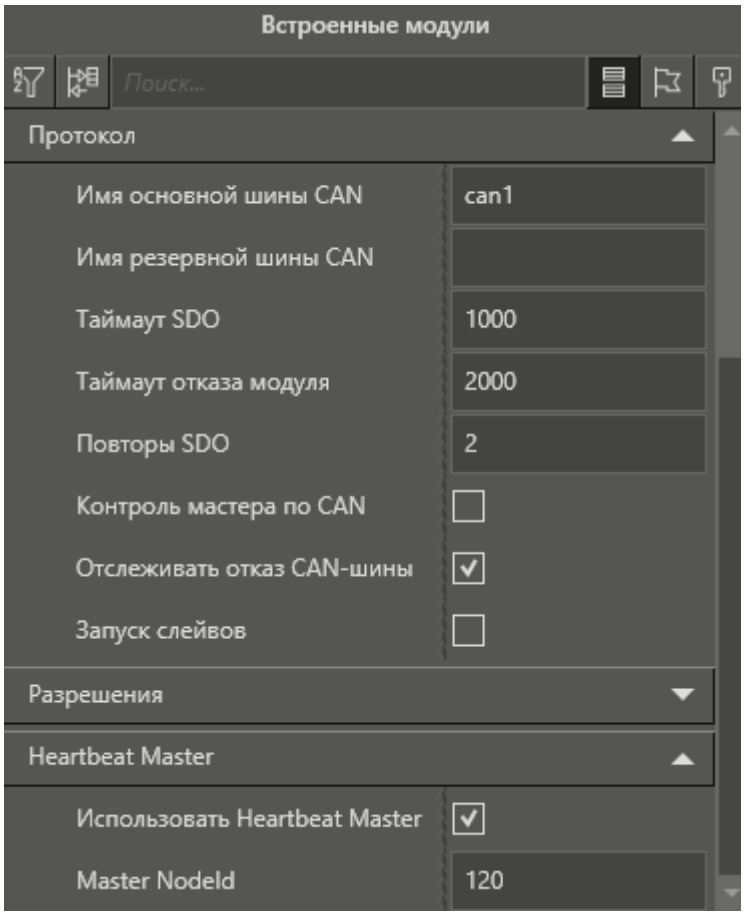


Рисунок 26

Настройки, применимые к протоколу в панели свойств «Встроенные модули», приведены в таблице 4.

Таблица 4

Свойство	Описание
Категория Протокол	
Имя основной шины CAN	Указывается имя основной шины, которое задано в контроллере. Берется из документации. Регистр вводимых символов учитывается.
Имя резервной шины CAN	Эта настройка не используется в контроллере UZOLA PRO100
Таймаут SDO	Время ожидания ответа от устройства при запросе SDO.
Таймаут отказа модуля	Устройства CANopen в процессе работы формируют специальные сообщения сердцебиения (Heartbeat), если их нет – модуль не работает. Протокол отслеживает все поступающие Heartbeat и сбрасывает таймер. Если не поступало сообщений в течение заданного времени в данном свойстве, то выдается отказ модуля. Настройки выдачи Heartbeat определяются параметрами модуля - число не должно быть меньше периода выдачи Heartbeat от модуля. По умолчанию - 2000. Рекомендуется выставлять настройку 10000.
Повторы SDO	Определяет, сколько сделать повторов, если на запрос SDO не было ответа.
Контроль мастера по CAN	Если настройка включена, то осуществляется контроль наличия мастера по CAN-шине. При отключении настройки контроль мастера осуществляется по сети Ethernet.
Отслеживать отказ CAN-шины	Настройка включена по умолчанию.
Запуск слейвов	Перевод всех модулей в режим Operational.
Категория Heartbeat Master	
Использовать Heartbeat Master	Центральный модуль при старте начинает выдавать свои Heartbeat сообщения с адресом, заданным в Master NodeID. Модули в момент включения перебирают скорости, получают пакеты и остаются на этой скорости. Если установлен флаг в данном свойстве, то будет использование этого режима. Для встроенных модулей контроллеров Uzola включено по умолчанию.
Master NodeID	Определяется ID контроллера, от которого будет слаться Heartbeat. По умолчанию 120

Модули ввода-вывода контроллеров добавляются в группу «Встроенные модули» в дереве системы (см. рисунок 27).

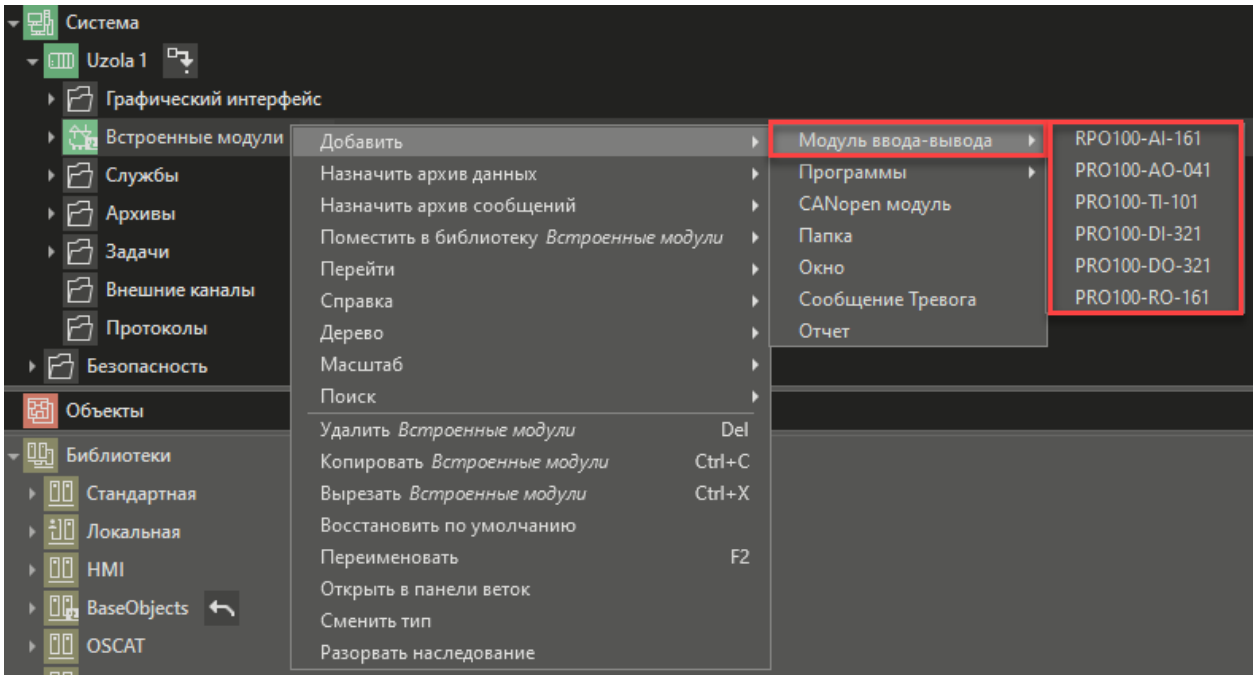


Рисунок 27

Поддерживаемые модули КТСИ Uzola приведены в таблице 5.

Таблица 5

Название	Описание
RPO100-AI-161	16-ти канальный модуль аналоговых вводов.
PRO100-AO-041	4-х канальный модуль вывода аналоговых сигналов постоянного тока и/или напряжения.
PRO100-TI-101	10-ти канальный модуль ввода термопреобразователей сопротивления.
PRO100-DI-321	32-х канальный модуль дискретных вводов постоянного напряжения 24 В.
PRO100-DO-321	32-х канальный модуль дискретных выводов.
PRO100-RO-161	16-ти канальный модуль релейных выводов.
PRO100-TC-121	12-ти канальный модуль ввода термо-ЭДС

Свойства модуля задаются в панели свойств модуля ввода/вывода (см. рисунок 28).

Основные свойства модулей ввода/вывода из раздела «Настройки» приведены в таблице 6.

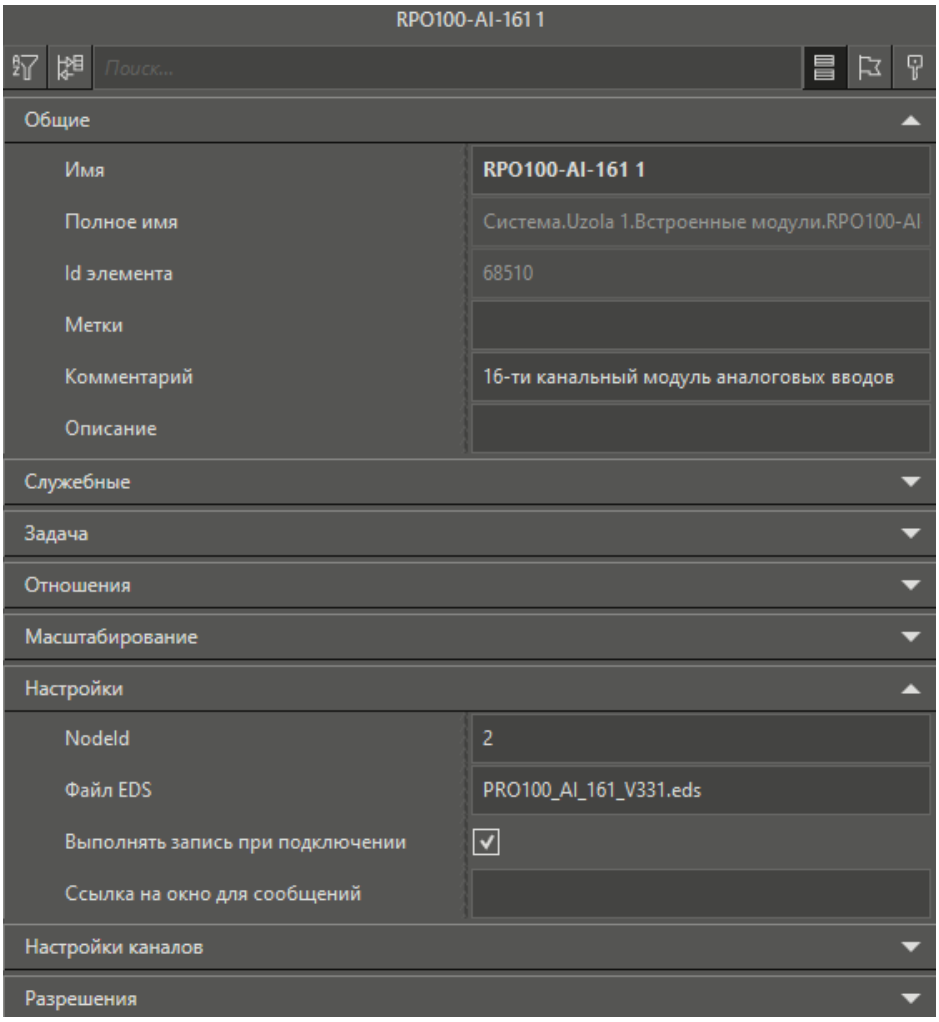
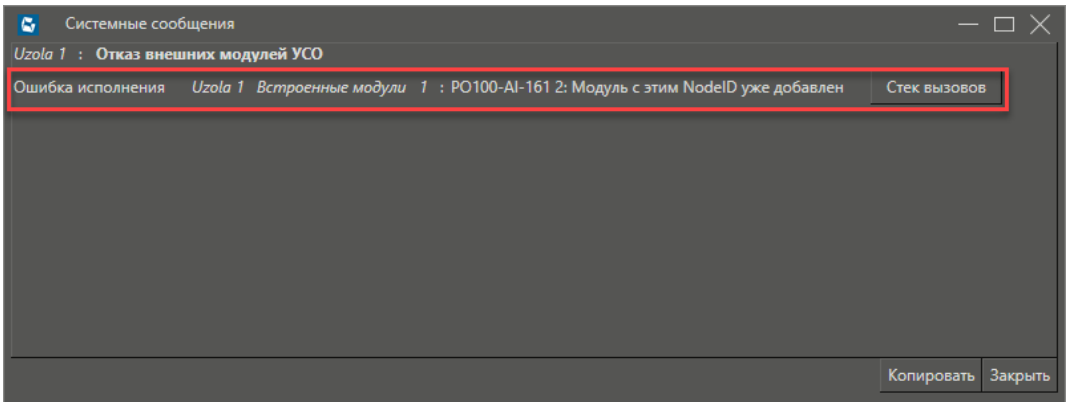
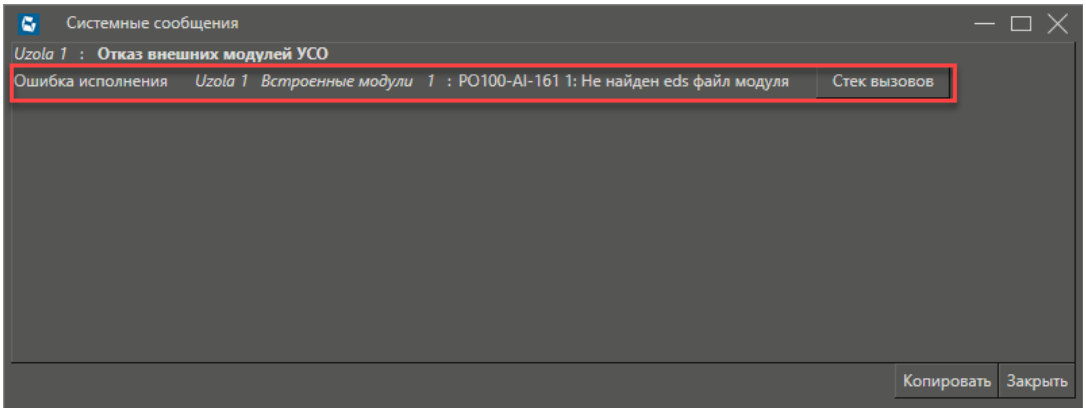


Рисунок 28

Таблица 6

Название	Описание
NodeID	Номер модуля на шине. Определяется индикацией на самом модуле при старте. Адрес модуля в шине раздается при включении питания центральным контроллером или переводом вручную переключателя на центральном контроллере в верхнее положение. Если добавить модуль с повторяющимся NodeID, то при запуске проекта на исполнение появится соответствующее сообщение:  Такой модуль в опрос не включается.

Название	Описание
Файл EDS	Путь к файлу eds. Полный путь к eds файлам: [папка установки mplc]/eds/[имя eds файла]. Если файлы не будут найдены по указанному пути, то модуль блокируется и при старте mplc выводится соответствующее сообщение:  В таком случае нужно обратиться в техподдержку за актуальными eds-файлами и установить их на контроллер
Выполнять запись при подключении	При значении TRUE будет производиться однократная запись значений на всех выходах при следующих ситуациях: • при первоначальном подключении; • при переключении исполнения на резервный узел в резервированной паре (запись будет выполняться даже при значении FALSE у настройки. Выполнять запись на резервном; • при переподключении после разрыва связи. При этом не имеет значения, совпадает ли текущее значение с начальным – запись будет выполнена независимо от этого. По умолчанию значение настройки TRUE.
Ссылка на окно для сообщений	Данная настройка используется для установки связи с окном, которое должно открываться при переходе из журнала к аварийным сообщениям, возникшим в этом элементе.

Для получения информации о состоянии связи с модулями ввода-вывода используются параметры «Отказ» и «Ошибка протокола» в группе «Встроенные модули» (см. рисунок 29).

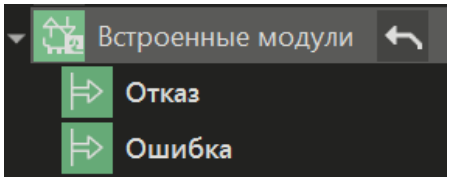


Рисунок 29

Параметр «Отказ» имеет тип значения BOOL. Если параметр принимает состояние TRUE, то это значит, что исполнительная система не может установить связь с данным модулем.

Параметр «Ошибка» имеет тип значения STRING. Если у исполнительной системы

нет связи со всеми модулями ввода/вывода, то параметр принимает значение «No connection».

Каналы модуля ввода/вывода автоматически добавляются при добавлении модуля КТСИ в проект в группу «Встроенные модули» (см. рисунок 30).

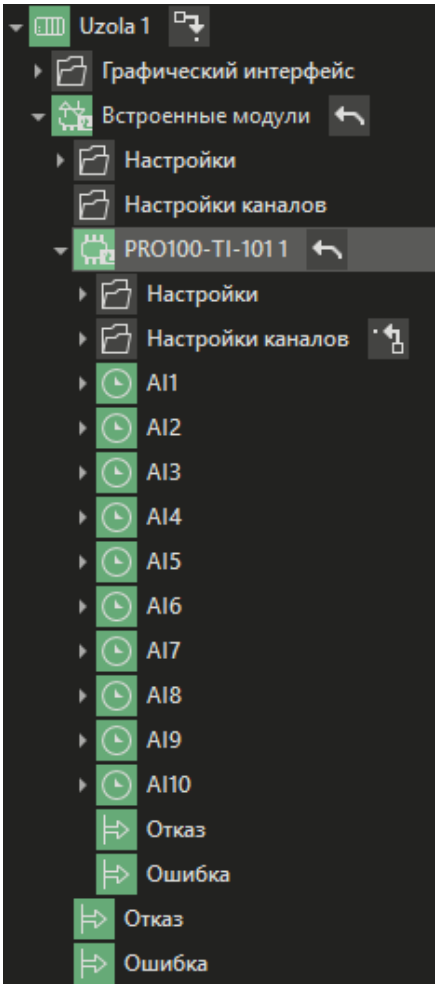


Рисунок 30

Каналы модулей ввода-вывода КТСИ Uzola имеют свою панель свойств. Для каналов модулей ввода-вывода, которые можно добавить в группу «Встроенные модули», значения свойств преднастроены (см. рисунок 31).

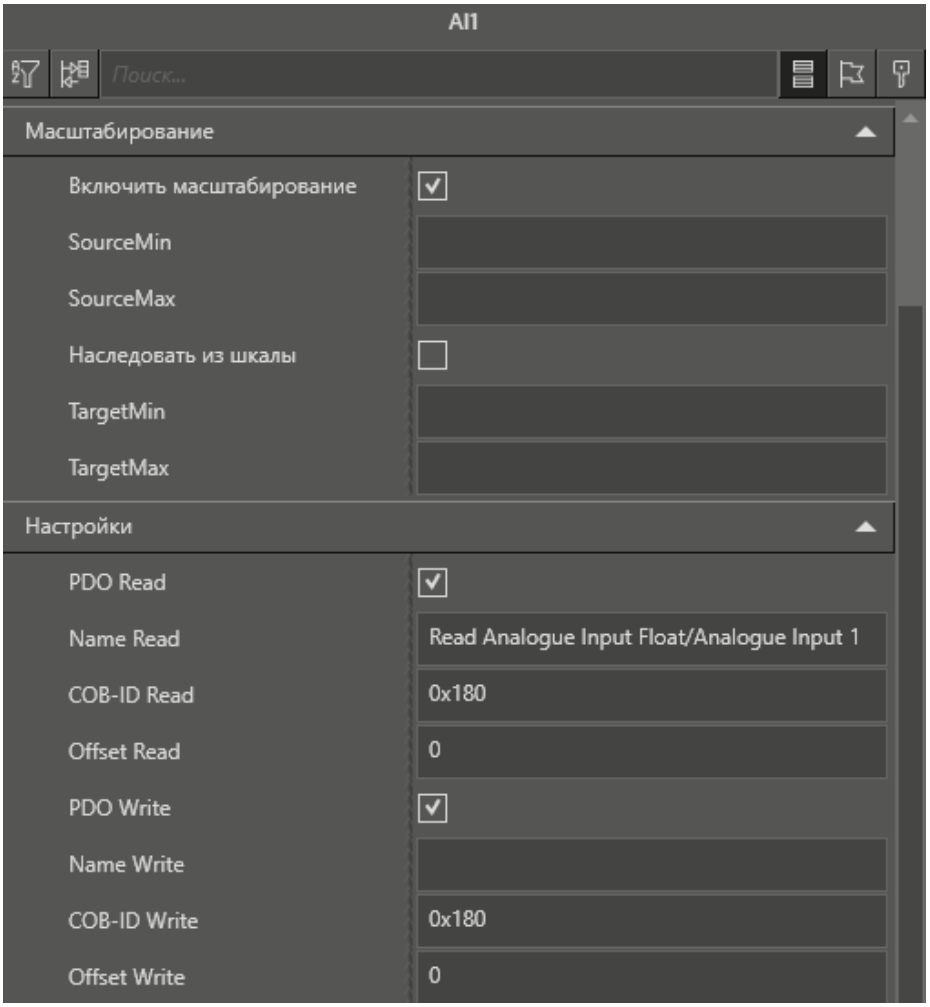


Рисунок 31

Основные свойства канала модуля приведены в таблице 7.

Таблица 7

Название	Описание
PDO Read	Если флаг установлен, то чтение модуля по PDO. Соответственно данный адрес должен быть доступен в этом режиме. Если снять флаг - чтение будет по SDO. В этом случае COD_ID Read и Offset не используется - только имя параметра.
Name Read	Указывается имя параметра для чтения. Указывается полный путь как в eds, т.е. номер секции/номер параметра. Разделитель через слэш.
COB-ID Read	COB параметра для чтения из eds файла. Прибавлять NodeId не требуется - это делает сам протокол.
Offset Read	Номер начального байта значения в принятом 8 байтовом пакете.
PDO Write	Если флаг установлен, то запись модуля по PDO. Если снят - запись по SDO.
Name Write	Имя параметра для записи.
COB-ID Write	COB для записи.
Offset Write	Смещение при записи.

Для возможности конфигурирования модулей ввода-вывода необходимо создать

отдельный проект конфигуратора. Для этого создаем новый проект и добавляем в него контроллер Uzola (см. рисунок 32).

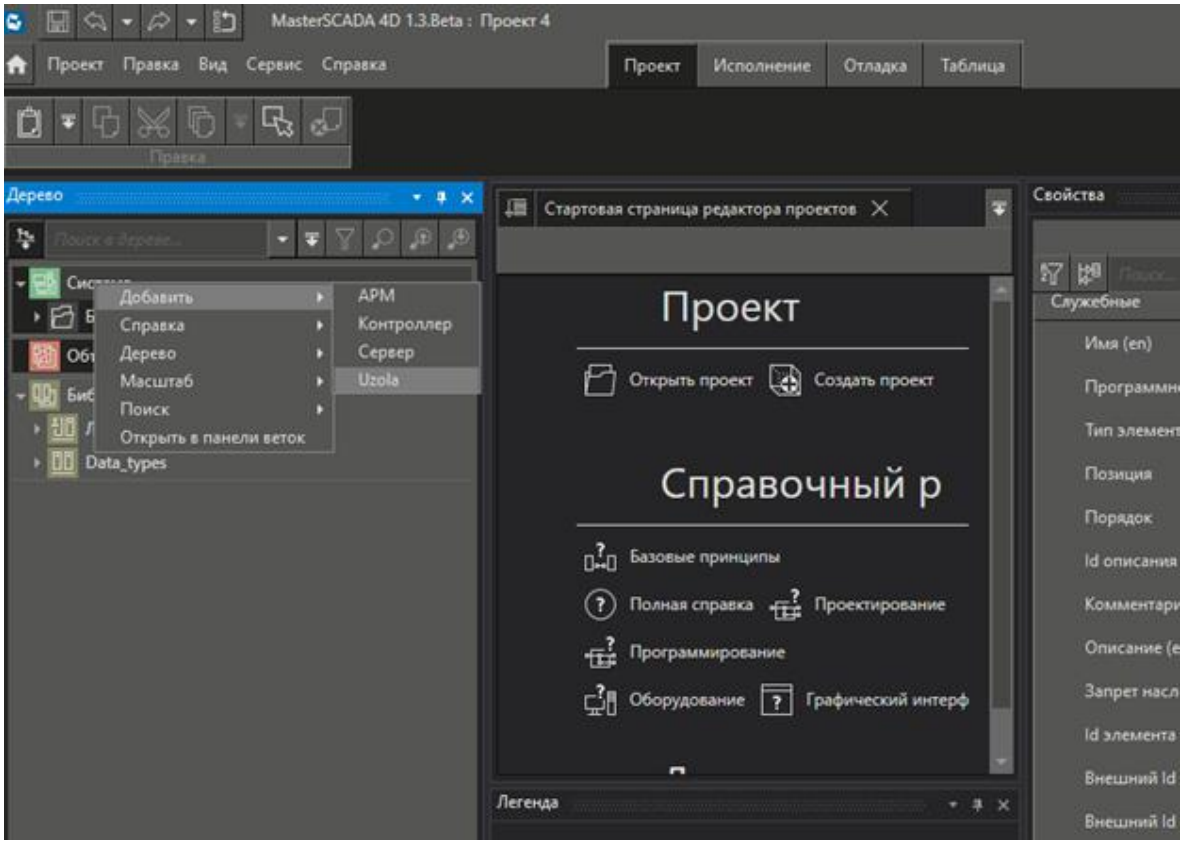


Рисунок 32

Удаляем ветку «Встроенные модули» поскольку мы создаем проект Конфигуратора, а в ветке «Конфигуратор» Удаляем/Добавляем модули ввода-вывода в соответствии с реальным наличием и расположением модулей на шине (см. рисунок 33).

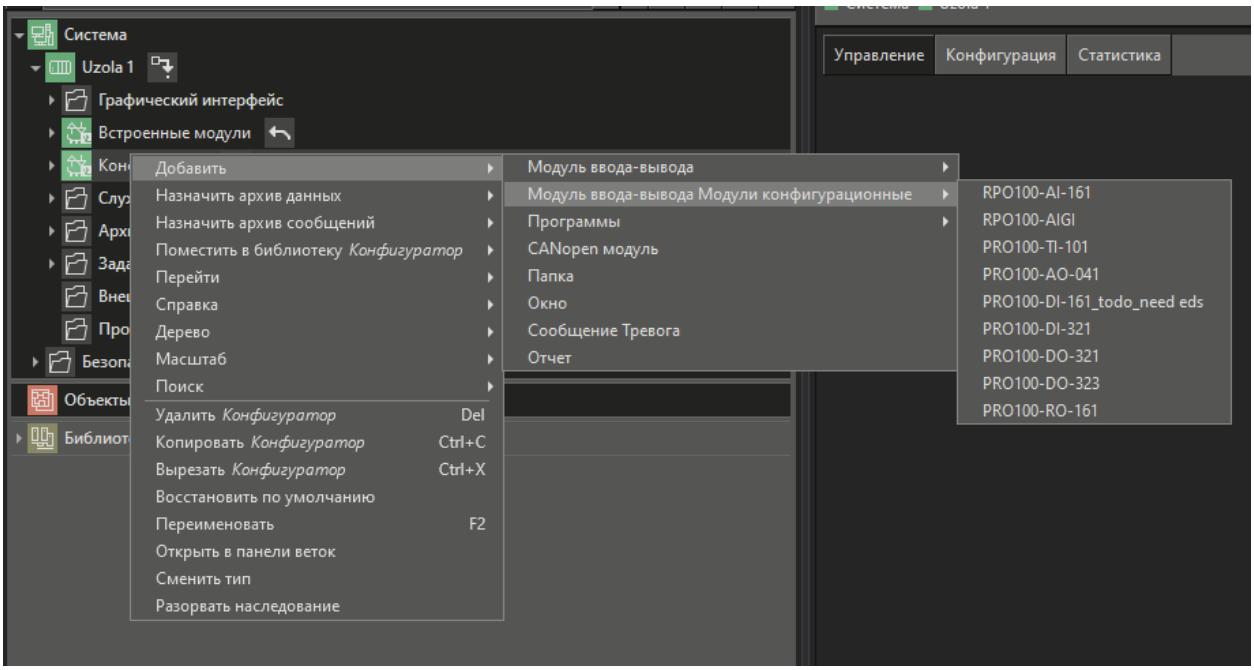


Рисунок 33

Важно! В библиотеке присутствуют модули ввода-вывода «простые» и модули «конфигурационные». В проект конфигуратора нужно добавлять модули конфигурационные.

После формирования корзины с модулями необходимо настроить параметр NodeId, в соответствии с реальным положением модуля на шине (см. рисунок 34).

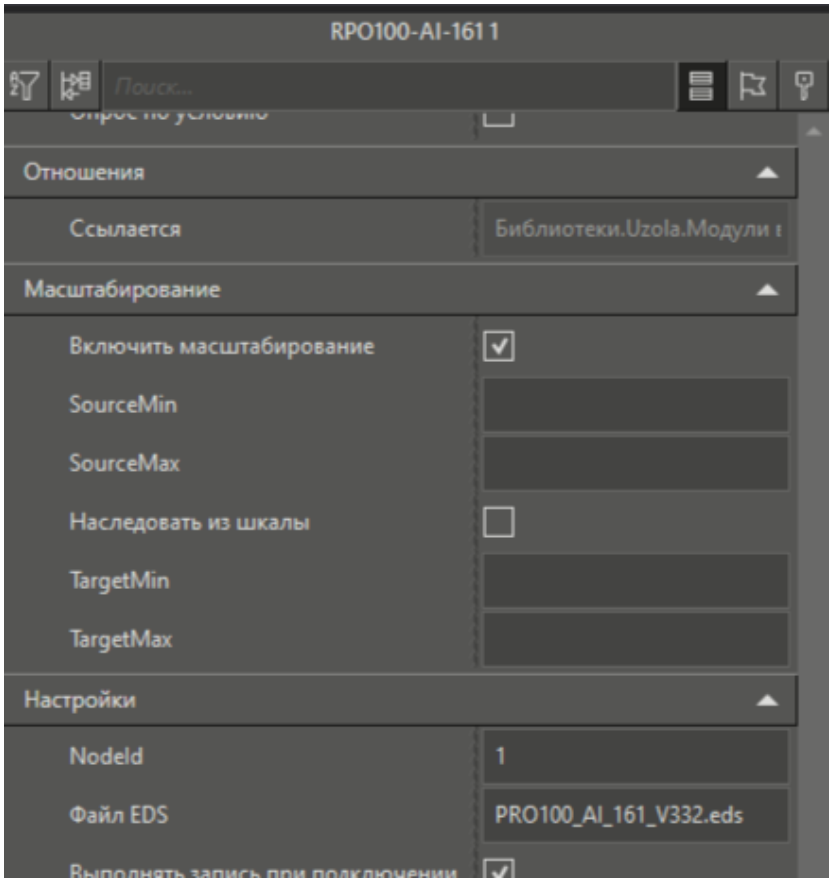


Рисунок 34

Так же обязательно должен быть отмечен параметр «Запуск слейвов» в настройках Конфигуратора (см. рисунок 35).

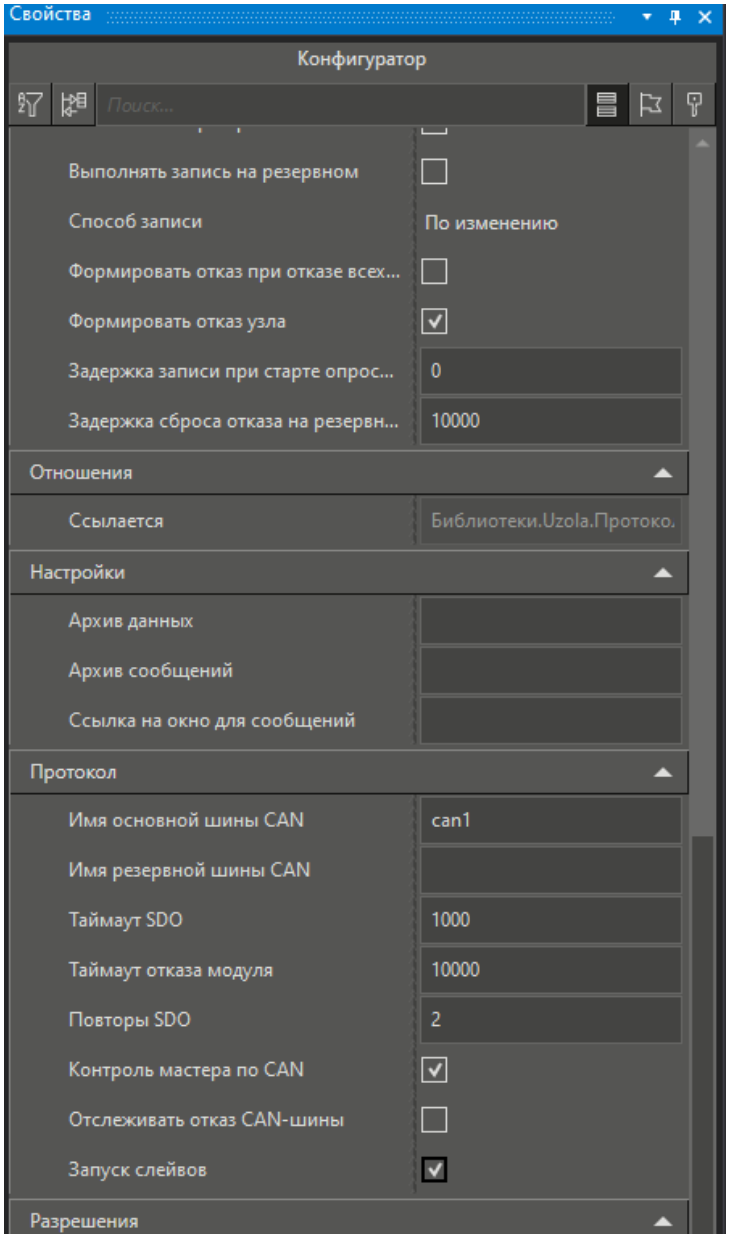


Рисунок 35

После этого можно подключаться к контроллеру и прогружать проект Конфигуратора.

Для удобства конфигурирования можно воспользоваться окнами управления, которые присутствуют в ветке ресурсы каждого конфигурационного модуля (см. рисунок 36).

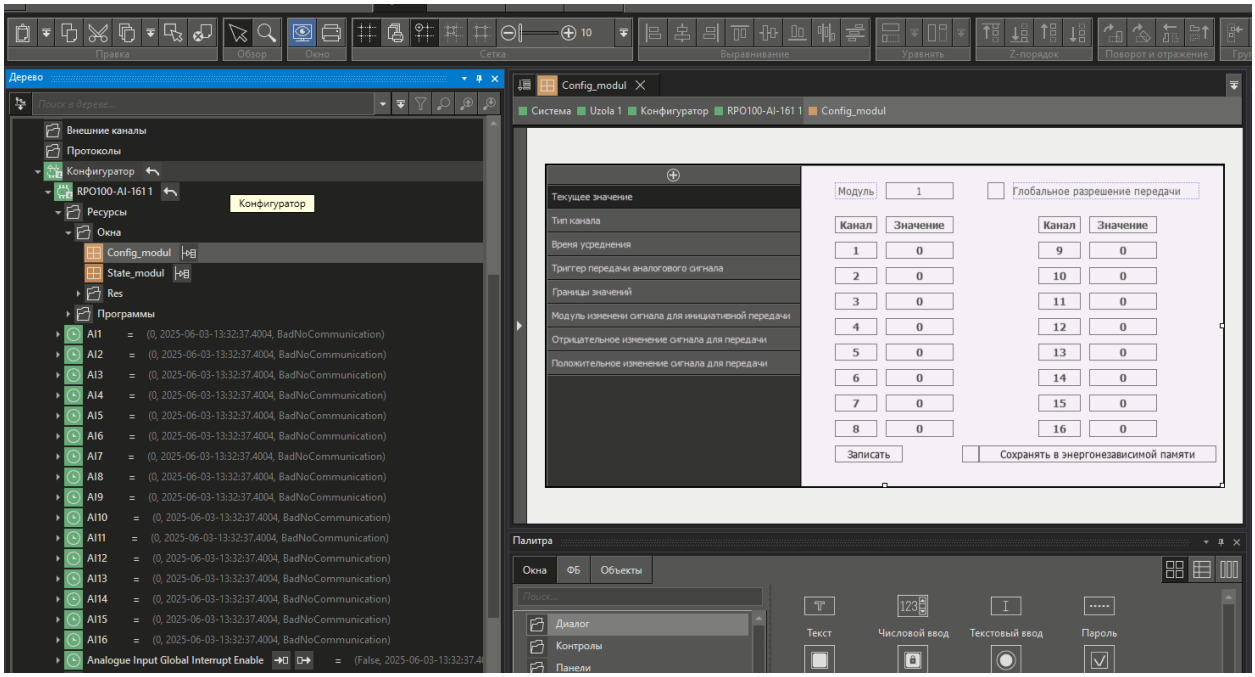


Рисунок 36

Пример перечня параметров (на примере модуля аналогового ввода PRO100-AI-161) представлен на рисунке 37.

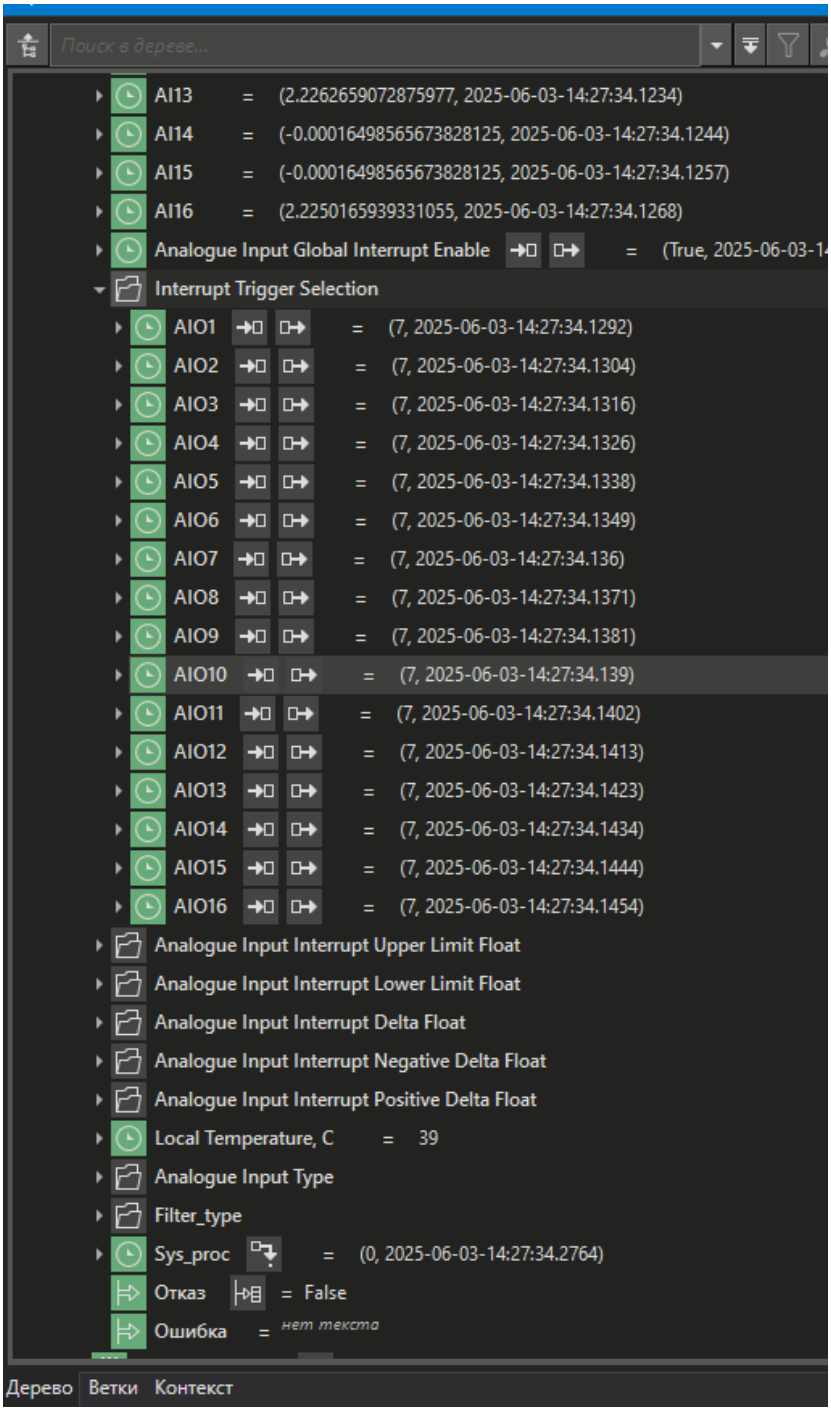


Рисунок 37

Каждый параметр имеет вход и выход (см. рисунок 38). Вход отображает текущее значение параметра, а в выход нужно записывать необходимые настроечные параметры, приведенные в таблице 8. Настройка типа подключенного ТС производится по таблице 9.

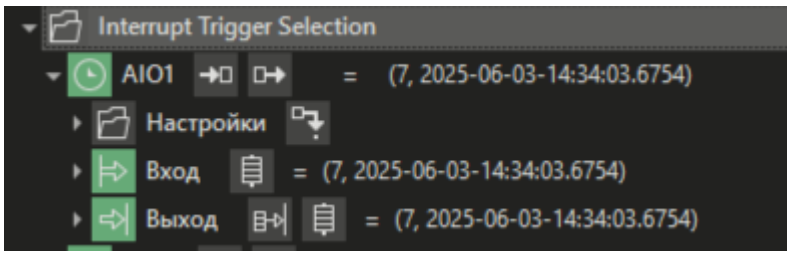


Рисунок 38

Таблица 8 – Настроечные параметры модулей ввода-вывода

Параметр		Значение по умолчанию	Комментарии
PRO100-AI-161			
Analogue Input Global Interrupt Enable	Разрешение передачи аналогового сигнала	true	Устанавливается для всех каналов одновременно. При установке значения параметра как «FALSE» никакие PDO сообщения передаваться не будут, также не будет работать индикация выходов.
Interrupt Trigger Selection	Триггер передачи аналогового сигнала	7	Устанавливается независимо для каждого канала. Представляет собой битовое поле, для которого разрешена операция «ИЛИ». Определены следующие прерывания: 1 – выход измеренного сигнала за верхнюю границу, 2 – выход измеренного сигнала за нижнюю границу, 4 – модуль изменения входного сигнала с момента последней передачи больше установленной величины, 8 – уменьшение величины входного сигнала с момента последней передачи больше установленной величины, 16 – увеличение величины входного сигнала с момента последней передачи больше установленной величины.
Analogue Input Interrupt Upper Limit Float	Верхняя граница аналогового сигнала	0	Устанавливается для каждого канала отдельно. Не обязательно должна быть равна верхней границе диапазона измерения канала. В случае, когда измеренное значение входного сигнала больше верхней границы, могут передаваться PDO сообщения и включаться индикация ошибки канала (если соответствующие прерывания разрешены).
Analogue Input Interrupt Lower Limit Float	Нижняя граница аналогового сигнала	0	Устанавливается для каждого канала отдельно. Не обязательно должна быть равна нижней границе диапазона измерения канала. В случае, когда измеренное значение входного сигнала меньше нижней границы, могут передаваться PDO сообщения и включаться индикация ошибки канала (если соответствующие прерывания разрешены)
Analogue Input Interrupt Delta Float	Модуль изменения аналогового сигнала для передачи	0	Устанавливается для каждого канала отдельно. В случае, когда модуль изменения измеренного значения входного сигнала с момента последней передачи больше данного значения, могут передаваться PDO сообщения (если соответствующие прерывания разрешены)
Analogue Input Interrupt Negative Delta Float	Отрицательное изменение аналогового сигнала для передачи	0	Устанавливается для каждого канала отдельно. В случае, когда уменьшение измеренного значение входного сигнала с момента последней передачи больше данного значения, могут передаваться PDO сообщения (если соответствующие прерывания разрешены)

Параметр		Значение по умолчанию	Комментарии
Analogue Input Interrupt Positive Delta Float	Положительное изменение аналогового сигнала для передачи	0	Устанавливается для каждого канала отдельно. В случае, когда увеличение измеренного значение входного сигнала с момента последней передачи больше данного значения, могут передаваться PDO сообщения (если соответствующие прерывания разрешены)
Local Temperature, C	Температура	-	Температура внутри модуля в градусах Цельсия
Analogue Input Type	Тип измеряемой величины и её диапазон	0	Определяется отдельно для каждого канала модуля. Допустимые значения: 0 – измерение напряжения от минус 10 до 10 вольт, 1 – измерение напряжения от минус 5 до 5 вольт, 2 – измерение напряжения от 0 до 10 вольт, 3 – измерение напряжения от 0 до 5 вольт, 4 – измерение тока от 0 до 20 миллиампер, 5 – измерение тока от 4 до 20 миллиампер.
Filter_type	Постоянная фильтра	1	Определяется отдельно для каждого канала модуля. Допустимые значения: 0 – нет усреднения входного сигнала, 1 – усреднение 20 мс, 2 – усреднение 200 мс, 3 – усреднение 1 с, 4 – усреднение 5 с.
Sys_proc	Системный регистр		Системная ячейка. Используется для записи параметров в энергонезависимую память микроконтроллера модуля
PRO100-DO-321			
Using_Output_8bit	Использование канала	0	Флаг использования канала. Представляет собой битовое поле, один бит которого соответствует одному каналу дискретного вывода (последовательно с 1 по 32 каналы). Если значение бита равно «0», тогда сообщения ключа-реле об ошибке канала не обрабатываются и отключена индикация ошибки канала. Значение регистра не влияет на выставление уровня логического выхода.
Enable_open_load_Detect	Детектирование ошибки канала	0	Флаг аппаратного детектирования ошибки канала. Представляет собой битовое поле, один бит которого соответствует одному каналу дискретного вывода (последовательно с 1 по 32 каналы). Если значение бита равно «1», ключ-реле осуществляет аппаратное детектирование ошибки канала.
Error_Overload_Outputs	Короткое замыкание нагрузки	0	Детектирование короткого замыкания или перегрева нагрузки. Представляет собой битовое поле, один бит которого соответствует одному каналу дискретного вывода (последовательно с 1 по 32 каналы). Значение бита «1» свидетельствует о детектировании короткого замыкания или перегрева нагрузки соответствующего канала.
Error_Open_load_Outputs	` Обрыв нагрузки	0	Детектирование обрыва нагрузки. Представляет собой битовое поле, один бит которого

Параметр		Значение по умолчанию	Комментарии
			соответствует одному каналу дискретного вывода (последовательно с 1 по 32 каналы). Значение бита «1» свидетельствует о детектировании обрыва нагрузки соответствующего канала
Error Value Outputs 8 Bit	Логический выход при ошибке	0	Логический выходной сигнал, устанавливаемый в случае ошибки или остановки рабочей программы модуля. Представляет собой битовое поле, один бит которого соответствует одному каналу дискретного вывода (последовательно с 1 по 32 каналы)
Sys_proc	Системный регистр	0	Системная ячейка. Используется для записи параметров в энергонезависимую память микроконтроллера модуля
PRO100-DI-321			
Counter_Overflow	Флаг переполнения счетчика		Представляет собой битовое поле, один бит которого соответствует одному каналу счета импульсов (последовательно с 1 по 8 каналы). Значение бита «1» свидетельствует о переполнении счетчика соответствующего канала.
Counter_Reset	Флаг перезагрузки счетчика		Представляет собой битовое поле, один бит которого соответствует одному каналу счета импульсов (последовательно с 1 по 8 каналы)
Counter_Enable	Разрешение счетчиков	false	Глобальное разрешение работы каналов счета импульсов и конфигурация входных каналов с 1 по 8
Global Interrupt Enable Digital	Разрешение передачи дискретного сигнала	true	Глобальное разрешение прерываний для каналов дискретного ввода
Sys_proc	Системный регистр	0	Системная ячейка. Используется для записи параметров в энергонезависимую память микроконтроллера модуля
PRO100-RO-161			
Using_Output_8bit	Флаг использования канала		Представляет собой битовое поле, один бит которого соответствует одному каналу релейного вывода (последовательно с 1 по 16 каналы). Если значение бита равно «0», тогда отключена индикация канала. Значение регистра не влияет на выдачу сигналов управления на катушку реле.
Error Value Outputs 8 Bit	Логический выход при ошибке	0	Логический выходной сигнал, устанавливаемый в случае ошибки или остановки рабочей программы модуля
Sys_proc	Системный регистр	0	Системная ячейка. Используется для записи параметров в энергонезависимую память микроконтроллера модуля
PRO100-AO-041			
Analogue Output Type	Тип канала	0	Тип задаваемой величины и её диапазон. Допустимые значения: 0 – задание тока от 4 до 20 миллиампер,

Параметр		Значение по умолчанию	Комментарии
			1 – задание тока от 0 до 20 миллиампер.
Global Interrupt Enable Digital	Разрешение прерываний	True	Глобальное разрешение прерываний для передачи признака ошибки выходных каналов. Устанавливается для всех каналов одновременно.
Analogue Output Offset Float	Нижняя граница аналогового сигнала	4	Устанавливается для каждого канала отдельно. Не обязательно должна быть равна нижней границе диапазона задания канала. В случае, когда измеренное заданное значение выходного сигнала меньше нижней границы выставляется признак ошибки канала.
Analogue Output Scaling Float	Верхняя граница аналогового сигнала	20	Устанавливается для каждого канала отдельно. Не обязательно должна быть равна верхней границе диапазона задания канала. В случае, когда заданное значение выходного сигнала больше верхней границы выставляется признак ошибки канала
Sys_proc	Системный регистр	0	Системная ячейка. Используется для записи параметров в энергонезависимую память микроконтроллера модуля
PRO100-TI-101			
Analogue Input Global Interrupt Enable	Разрешение передачи аналогового сигнала	true	Глобальное разрешение прерываний. Устанавливается для всех каналов одновременно. При установке значения параметра как «FALSE» никакие PDO сообщения передаваться не будут
Interrupt Trigger Selection	Триггер передачи аналогового сигнала	7	Выбор источника прерываний для отправки PDO сообщения о значении температуры. Устанавливается независимо для каждого канала. Представляет собой битовое поле, для которого разрешена операция «ИЛИ». Определены следующие прерывания: 1 – выход измеренной температуры за верхнюю границу, 2 – выход измеренной температуры за нижнюю границу, 4 – модуль изменения температуры с момента последней передачи больше установленной величины, 8 – уменьшение величины температуры с момента последней передачи больше установленной величины, 16 – увеличение величины температуры с момента последней передачи больше установленной величины.
Analogue Input Interrupt Upper Limit Float	Верхняя граница аналогового сигнала	400	Устанавливается для каждого канала отдельно. Не обязательно должна быть равна верхней границе диапазона измерения канала
Analogue Input Interrupt Lower Limit Float	Нижняя граница аналогового сигнала	-100	Устанавливается для каждого канала отдельно. Не обязательно должна быть равна нижней границе диапазона измерения канала

Параметр		Значение по умолчанию	Комментарии
Analogue Input Interrupt Delta Float	Модуль изменения аналогового сигнала для передачи	1	Изменение измеренного значения температуры для инициативной передачи значения. Устанавливается для каждого канала отдельно. В случае, когда модуль изменения температуры с момента последней передачи больше данного значения, могут передаваться PDO сообщения (если соответствующие прерывания разрешены)
Analogue Input Interrupt Negative Delta Float	Отрицательное изменение аналогового сигнала для передачи	0	Устанавливается для каждого канала отдельно. В случае, когда уменьшение измеренной температуры с момента последней передачи больше данного значения, могут передаваться PDO сообщения (если соответствующие прерывания разрешены)
Analogue Input Interrupt Positive Delta Float	Положительное изменение аналогового сигнала для передачи	0	Устанавливается для каждого канала отдельно. В случае, когда увеличение измеренного значение температуры с момента последней передачи больше данного значения, могут передаваться PDO сообщения (если соответствующие прерывания разрешены)
RTD Type	Тип подключенного ТС	13	Определяется отдельно для каждого канала модуля. Описание допустимых значений регистра приведено в таблице
Sys_proc	Системный регистр	0	Системная ячейка. Используется для записи параметров в энергонезависимую память микроконтроллера модуля

Таблица 9 – Настройка типа подключенного ТС

Значение регистра	Тип подключенного ТС (α)	Схема подключения	Диапазон измерения, С
0x0000	Омическое сопротивление до 150 Ом	4-проводная	
0x0001	50М(0,00428)	4-проводная	-180…+200
0x0002	50М(0,00426)	4-проводная	-50…+200
0x0003	50Р(0,00385)	4-проводная	-200…+850
0x0004	Pt50 0,00391)	4-проводная	-200…+850
0x0005	50Н(0,00617)	4-проводная	-60…+180
0x0010	Омическое сопротивление до 300 Ом	4-проводная	
0x0011	100М(0,00428)	4-проводная	-180…+200
0x0012	100М(0,00426)	4-проводная	-50…+200
0x0013	100Р(0,00385)	4-проводная	-200…+850
0x0014	Pt100(0,00391)	4-проводная	-200…+850

Значение регистра	Тип подключенного ТС (α)	Схема подключения	Диапазон измерения, С
0x0015	100Н(0,00617)	4-проводная	-60…+180
0x0020	Омическое сопротивление до 3000 Ом	4-проводная	
0x0021	1000М(0,00428)	4-проводная	-180…+200
0x0022	1000М(0,00426)	4-проводная	-50…+200
0x0023	1000Р(0,00385)	4-проводная	-200…+850
0x0024	Pt1000(0,00391)	4-проводная	-200…+850
0x0025	1000Н(0,00617)	4-проводная	-60…+180
0x0100	Омическое сопротивление до 150 Ом	3-проводная	
0x0101	50М(0,00428)	3-проводная	-180…+200
0x0102	50М(0,00426)	3-проводная	-50…+200
0x0103	50Р(0,00385)	3-проводная	-200…+850
0x0104	Pt50(0,00391)	3-проводная	-200…+850
0x0105	50Н(0,00617)	3-проводная	-60…+180
0x0110	Омическое сопротивление до 300 Ом	3-проводная	
0x0111	100М(0,00428)	3-проводная	-180…+200
0x0112	100М(0,00426)	3-проводная	-50…+200
0x0113	100Р(0,00385)	3-проводная	-200…+850
0x0114	Pt100(0,00391)	3-проводная	-200…+850
0x0115	100Н(0,00617)	3-проводная	-60…+180
0x0120	Омическое сопротивление до 3000 Ом	3-проводная	
0x0121	1000М(0,00428)	3-проводная	-180…+200
0x0122	1000М(0,00426)	3-проводная	-50…+200
0x0123	1000Р(0,00385)	3-проводная	-200…+850
0x0124	Pt1000(0,00391)	3-проводная	-200…+850
0x0125	1000Н(0,00617)	3-проводная	-60…+180

3.3 Настройка протоколов обмена с полевыми устройствами и внешними системами

3.3.1 Протокол Modbus RTU

Modbus RTU – популярный протокол для обмена данными между промышленными устройствами (ПЛК, датчики, приводы и т.д.), чтобы контролировать и мониторить оборудование на производстве.

Обычно используется интерфейс RS-485 (иногда RS-422 или RS-232). RS-485 — это двухпроводная линия (A и B), по которой данные передаются дифференциально, что позволяет подключать до 32 устройств на одной линии длиной до 1200 метров.

Данные передаются последовательно: старт-бит, 8 бит данных, (опционально бит чётности), стоп-бит.

Каждое устройство - слейв получает уникальный адрес в диапазоне 1–247. Мастер не имеет адреса. Адрес 0 - широковещательный, для команд всем сразу (ответа на такие команды не будет).

Пакет запроса, формируемый мастером, выглядит так:

[Адрес][Код функции][Данные][CRC]

- Адрес слейва (1 байт);
- Код функции (1 байт, например, 0x03 - чтение регистров);
- Данные (например, номер регистра и количество);
- CRC (2 байта, контрольная сумма).

Пример запроса:

0x01 0x03 0x00 0x10 0x00 0x02 CRC - мастер запрашивает у устройства с адресом 1 чтение двух регистров, начиная с 16-го.

Слейв слушает шину и реагирует только на пакеты, где адрес совпадает с его собственным. Если адрес не совпадает - игнорирует. Если запрос корректный и адрес совпал, слейв формирует ответ:

- Свой адрес;
- Код функции;
- Данные (например, значения регистров);
- CRC.

Если произошла ошибка, слейв возвращает код ошибки (код функции + 0x80 и код исключения).

Типы передаваемых данных:

- Coil (дискретные выходы, 1 бит)

• Discrete Input (дискретные входы, 1 бит)

• Input Register (аналоговые входы, 16 бит)

• Holding Register (аналоговые выходы, 16 бит)

Передавать можно как отдельные биты, так и 16-битные слова.

Протокол Modbus RTU входит в базовый комплект MasterSCADA 4D. Физическое подключение к процессорному модулю возможно через клеммные соединения (1 канал RS-485) или через специальный модуль расширения (3 канала RS-485).

В Дереве библиотек данные протоколы хранятся в категории «Стандартная», в группе «Протоколы». В проект добавляются, как правило, через контекстную панель узла, либо контекстное меню узла, либо контекстное меню группы «Протоколы».

Чтобы добавить возможность опроса устройств по данному протоколу, необходимо выбрать соответствующий элемент в контекстном меню узла или в контекстном меню группы «Протоколы», либо через контекстную панель узла (см. рисунок 39).

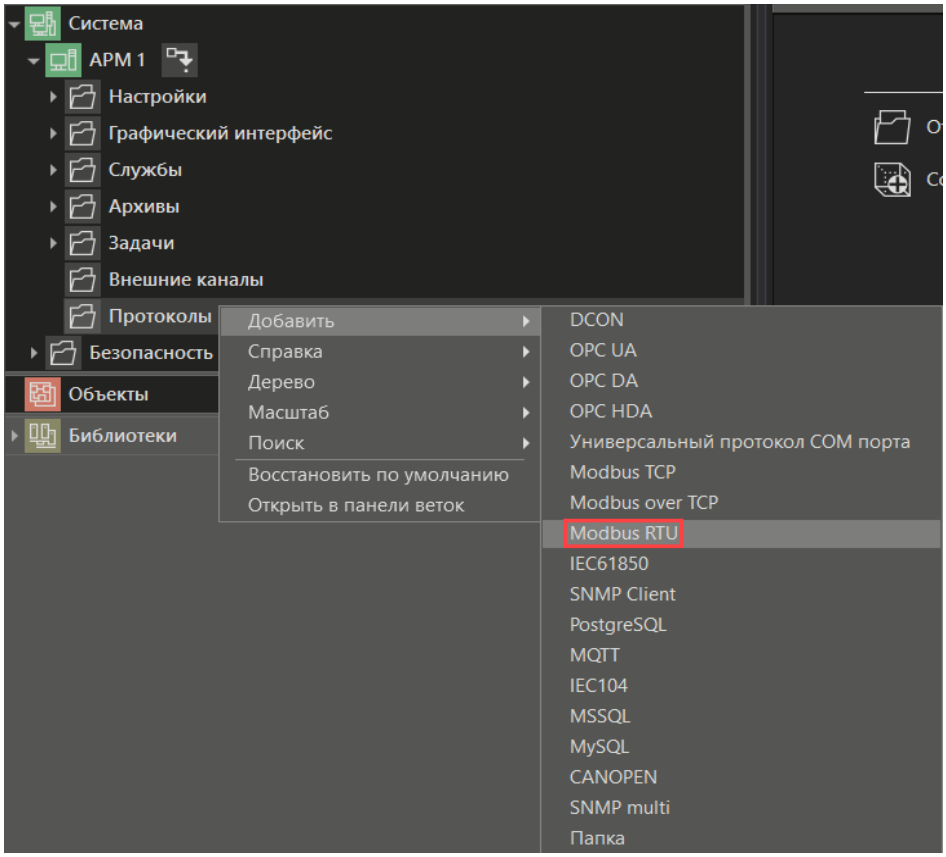


Рисунок 39

Получим результат в упрощенном дереве (см. рисунок 40).

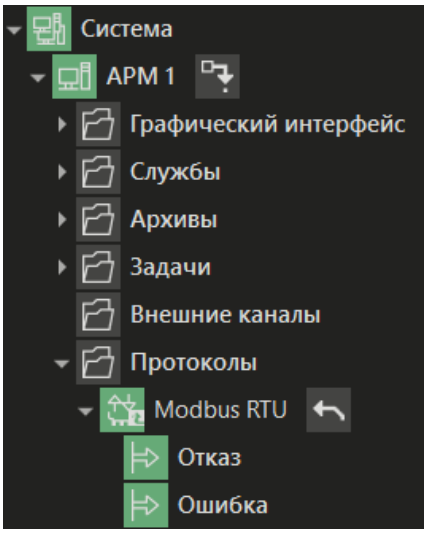


Рисунок 40

Протокол имеет параметр «Отказ». Значение данного параметра зависит от настройки «Формировать отказ при отказе всех модулей». При значении настройки TRUE параметр «Отказ» примет значение TRUE только тогда, когда откажут все модули. В случае использования резервирования переход на резервный сервер произойдет только после отказа всех модулей.

Далее нужно настроить «Панель свойств». Затем добавить модуль «Modbus Device», который работает по этому протоколу.

При необходимости, в протоколе могут быть созданы программы, окна и др. элементы. В этом случае у протокола появится группа «Ресурсы», такая же, как и у элементов объект, тег, канал.

Настройка элемента производится в панели свойств. Вид панели приведен на рисунке 41. Описание свойств панели приведено в таблице 10.

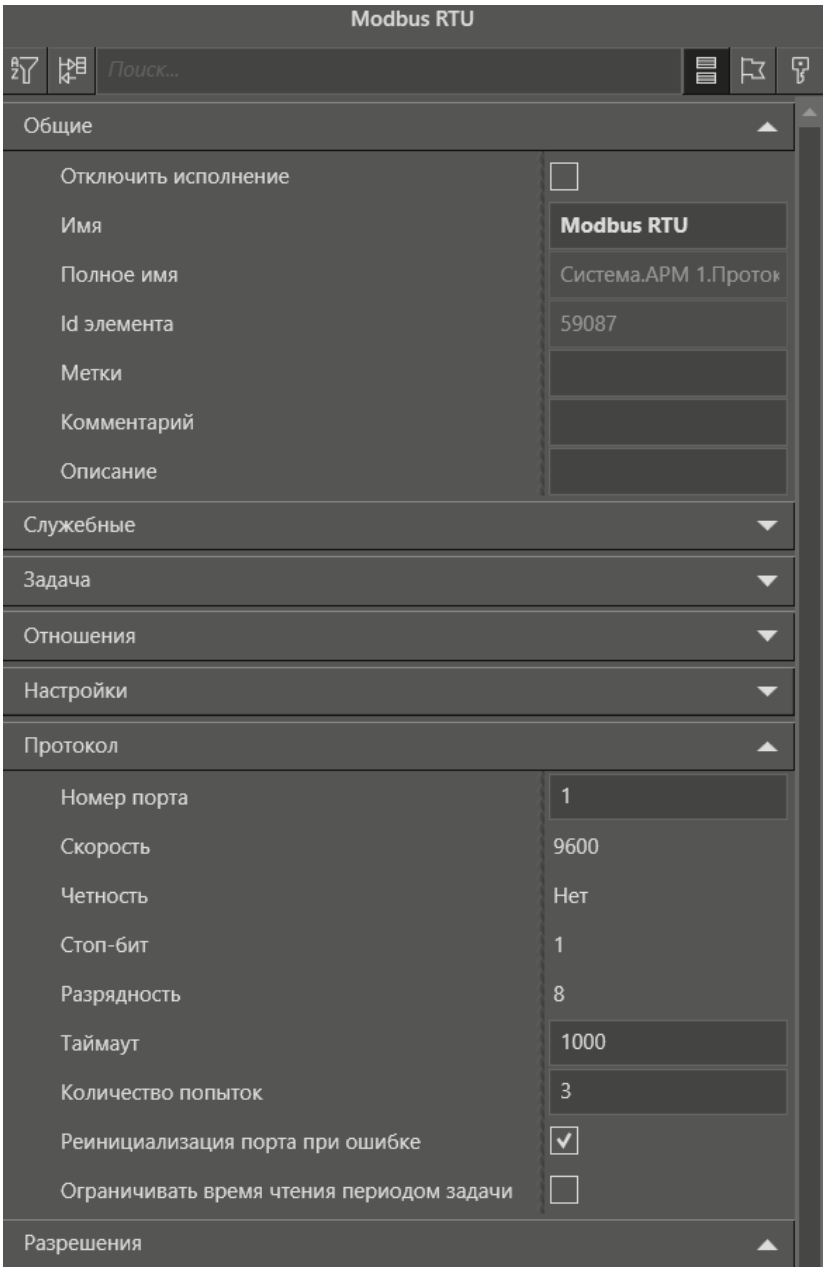


Рисунок 41

Таблица 10

Название	Рекомендации
Категория Общие	Как правило, для данного элемента не настраивается. Подробное описание данной категории приведено в соответствующем разделе
Категория Задача	Задается период, с которым будут опрашиваться устройства. Определяется способ работы протокола при резервировании узла.
Номер порта	Номер COM-порта, к которому подключаются устройства.
Скорость	Скорость, на которой будет производится опрос устройств (бит/с). Значение должно выбираться в зависимости от возможностей подключаемого устройства. Значение по умолчанию 9600. Если установить значение выше 115200, то в течение первых 3 миллисекунд после запуска цикл будет функционировать без задержек между запросами.
Четность	Определяется необходимость контроля четности. Возможные значения: Нет, Чет, Нечет.

Название	Рекомендации
Стоп-бит	Задается число стоповых битов (1, 1.5, 2), значение по умолчанию – 1;
Разрядность	Задается число информационных битов (7 или 8), значение по умолчанию – 8;
Таймаут	Задается значение таймаута по умолчанию для подключаемых устройств. Это максимальное время ожидания ответа от устройства (мс). Если ответ от устройства не будет получен в течении заданного интервала времени, то попытка опроса будет считаться неудачной, и зафиксируется ошибка данного запроса для конкретного устройства.
Количество попыток	Указывается предельное количество неудачных попыток получить ответ от устройства, произошедших подряд. Если за указанное количество попыток получить ответ от устройства не удалось, то сформируется сигнал TRUE у параметра «Отказ» у того устройства, с которым не удалось обнаружить связь.
Реинициализация порта при ошибке	Если при опросе будет получен отказ на всех устройствах, то произойдет переоткрытие порта. По умолчанию значение TRUE.
Ограничивать время чтения периодом задачи	Настройка приоритета записи над чтением. Если флаг установлен, то обработка данных происходит по следующему алгоритму: 1 Обработка всех устройств протокола, нахождение выходов (каналов, отвечающих за запись данных в устройство). Запись значений в устройство. 2 Начало опроса устройств. 3 После прочтения устройства, проверяется - не истекло ли время задачи. 4 Если истекло - запоминается номер устройства, на котором опрос остановился. В теги записываются все считанные значения за текущий цикл задачи. 5 Выход из задачи. 6 При следующей обработке протокола опять проводится запись значений (если таковые есть), и читается следующее по списку устройство.

Вид панели свойств модуля Modbus Device приведен на рисунке 42. Описание свойств панели приведено в таблице 11.

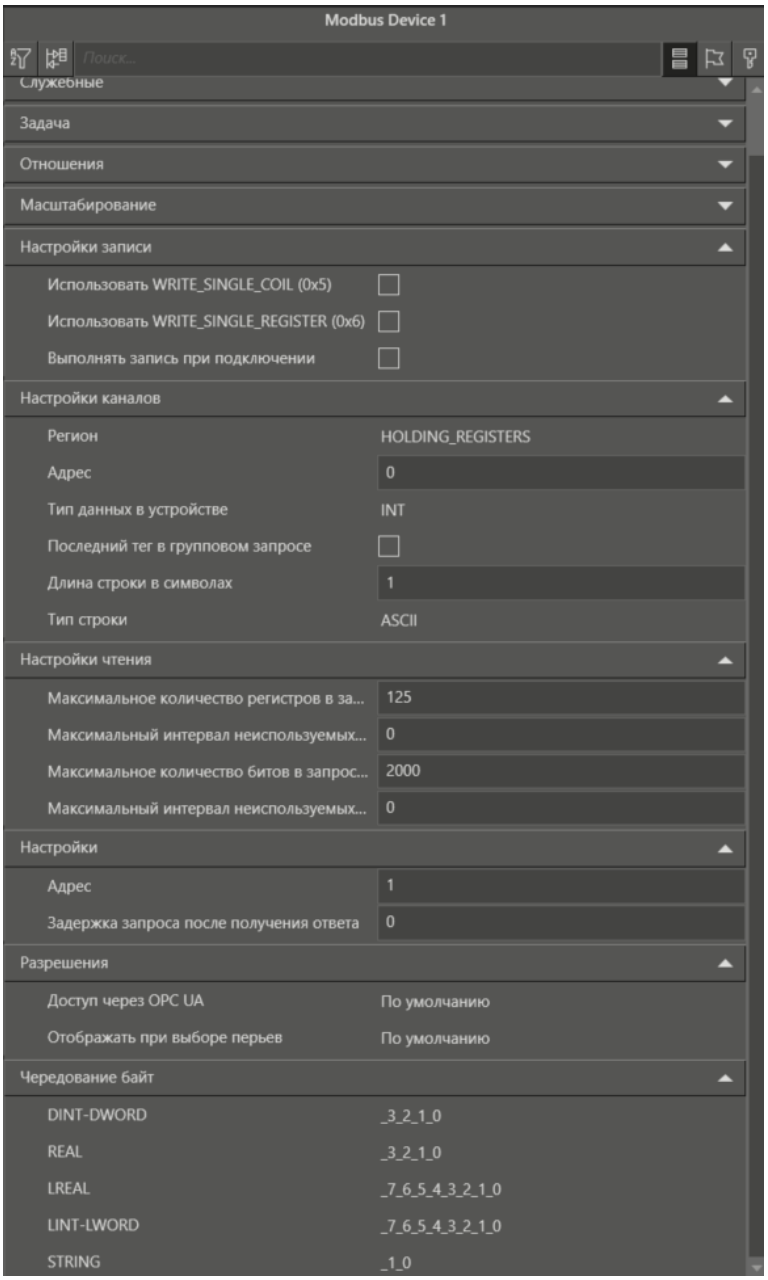


Рисунок 42

Таблица 11

Название	Рекомендации
Категория **Общие**	Как правило, для данного элемента не настраивается.
Категория **Задача**	Задается способ опроса устройства
Категория **Масштабирование**	Если Modbus-устройство не имеет встроенного масштабирования, то при помощи данной категории можно задать соответствие между значениями в устройстве и реальными значениями.
Категория **Настройки**	
Адрес	Задается адрес устройства (значение адреса определяется документацией на подключаемое устройство).

Название	Рекомендации
Задержка запроса после получения ответа	Задается интервал ожидания после получения ответа на запрос чтения или записи, по истечении которого выполняется следующий запрос.
Категория **Настройки записи**	
Использовать WRITE_SINGLE_COIL (0x5)	При значении TRUE используется функция записи в ячейку региона Coils - 0x05 (запись в одиночную ячейку). При значении FALSE используется функция записи в группу ячеек региона Coils - 0x0F (групповая запись). По умолчанию значение настройки FALSE.
Использовать WRITE_SINGLE_REGISTER (0x6)	При значении TRUE используется функция записи в ячейку региона Holding Registers - 0x06 (запись в одиночную ячейку). При значении FALSE используется функция записи в группу ячеек региона Holding Registers - 0x10 (групповая запись). По умолчанию значение настройки FALSE.
Выполнять запись при подключении	При значении TRUE будет производиться однократная запись значений на всех выходах при следующих ситуациях: • при каждом подключении к устройству, включая первоначальное; • при переключении исполнения на резервный узел в резервированной паре (запись будет выполняться даже при значении FALSE у настройки Выполнять запись на резервном); • при переподключении после разрыва связи. При этом не имеет значения, совпадает ли текущее значение с начальным, запись будет выполнена независимо от этого. По умолчанию значение настройки FALSE.
Категория **Настройки каналов**	Задаются значения по умолчания для создаваемых каналов. В дальнейшем эти настройки у каналов можно будет изменить.
Категория **Настройки чтения**	
Максимальное количество регистров в запросе чтения	Задается максимальное значение Input и Holding регистров для одного запроса чтения. По умолчанию значение равно 125 (максимально допустимое).
Максимальный интервал неиспользуемых регистров	Задается максимальное значение для разрыва между адресами регистров. Если разрыв меньше указанного значения, то происходит объединение в один запрос. Например, производится опрос регистров с номерами - 0, 1, 2, 4, 5. Если параметр будет установлен в 0, то регистры будут запрошены за два запроса (0-2, 4-5), если же установить его в 1 или выше, то за один запрос (0-5, а не использованный регистр 3 будет отброшен при разборе ответа). Увеличение данного параметра сокращает количество запросов и ускоряет опрос, но удлиняет кадр ответа. Также следует помнить, что некоторые Modbus устройства имеют ограничение на количество передаваемых за один запрос параметров.
Максимальное количество битов в запросе чтения	Задается максимальное значение Coils и Discrete Inputs битов для одного запроса чтения. По умолчанию значение равно 2000 (максимально допустимое).
Максимальный интервал неиспользуемых битов	Задается максимальное значение для разрыва между адресами битов. Если разрыв меньше указанного значения, то

Название	Рекомендации
	происходит объединение в один запрос. Также следует помнить, что некоторые Modbus устройства имеют ограничение на количество передаваемых за один запрос параметров.
Категория **Чередование байт**	
DINT-DWORD	Порядок следования байтов в целых числах. Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в целых числах. Доступны чередования: • _3_2_1_0 • _1_0_3_2 • _0_1_2_3 • _2_3_0_1
REAL	Порядок следования байтов в вещественных числах. Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в вещественных числах. Доступны чередования: • _3_2_1_0 • _1_0_3_2 • _0_1_2_3 • _2_3_0_1
LREAL	Порядок следования байтов в вещественных числах (8 байт). Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в вещественных числах. Доступны чередования: • _1_0_3_2_5_4_7_6 • _3_2_1_0_7_6_5_4 • _7_6_5_4_3_2_1_0 • _0_1_2_3_4_5_6_7
LINT-LWORD	Порядок следования байтов в целых числах (8 байт). Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в целых числах. Доступны чередования: • _1_0_3_2_5_4_7_6 • _3_2_1_0_7_6_5_4 • _7_6_5_4_3_2_1_0 • _0_1_2_3_4_5_6_7
STRING	Порядок следования байтов в строках. Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в строках. Доступны чередования: • _1_0 • _0_1

Далее необходимо настроить каналы опроса у каждого устройства (см. рисунок 43).

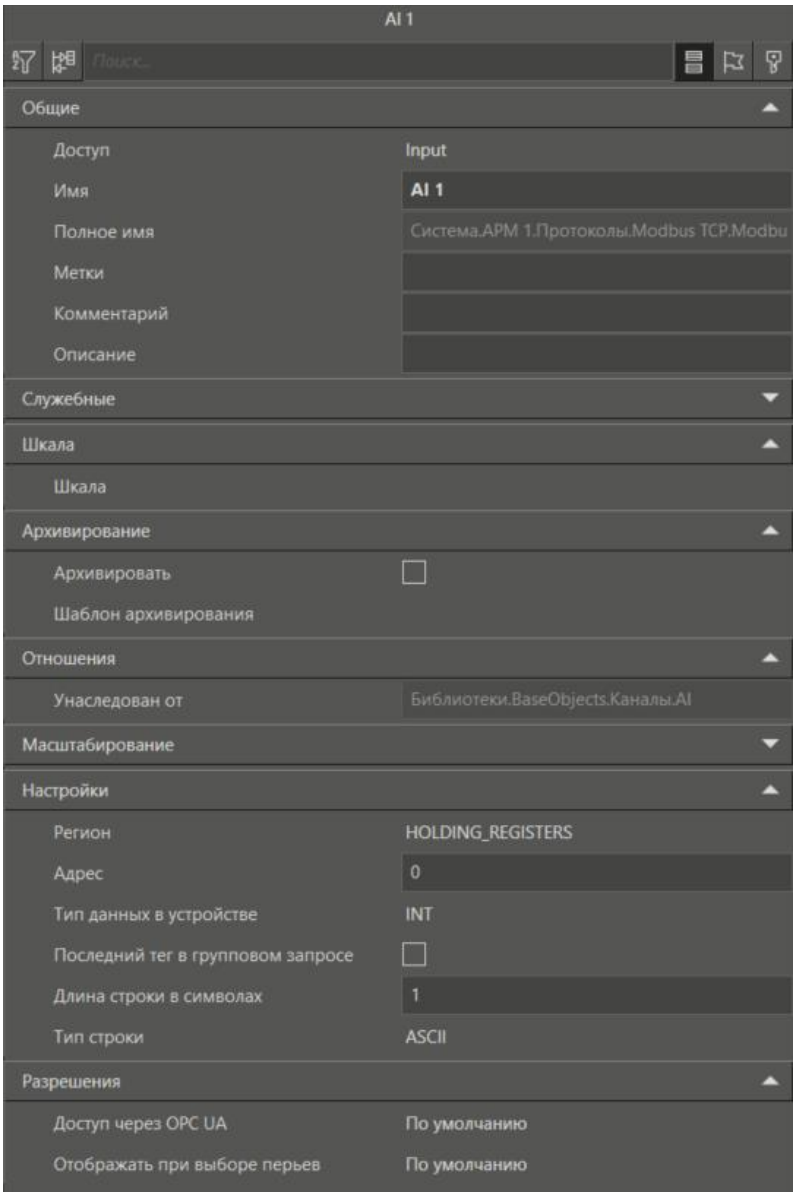


Рисунок 43

Описание свойств панели приведено в таблице 12.

Таблица 12

Название	Рекомендации
Категория **Общие**	Как правило, для данного элемента не настраивается.
Категория **Масштабирование**	Если устройство не имеет встроенного масштабирования, то при помощи данной категории можно задать соответствие между значениями в устройством и реальными значениями.
Категория **Шкала**	В данной категории можно назначить шкалу и определить ее настройки.
Категория **Архивирование**	Если значение канала требуется архивировать, то необходимо установить флаг в свойстве «Архивировать». В этом случае, значения канала будут сохраняться в базе данных. При необходимости, можно выбрать шаблон архивирования. При этом появятся настройки, соответствующие данному шаблону, и которые можно переопределить для конкретного канала.

Название	Рекомендации
Категория **Настройки**	
Регион	Задается регион пространства Modbus: •COILS; •DISCRETE_INPUTS; •INPUTS_REGISTERS; •HOLDING_REGISTERS. Если выбирается регион INPUTS_REGISTERS или DISCRETE_INPUTS, то тип данных доступа автоматически меняется на Input.
Адрес	Указывается адрес регистра, из которого требуется прочитать/записать данные (смещение адреса в выбранном адресном пространстве).
Тип данных в устройстве	Указывается тип читаемых данных, определяет количество считываемых регистров. При изменении данной настройки, у основных параметров канала автоматически меняется тип данных на соответствующий, но вида SYSTEM_*_PARAM. Например, если установить тип данных INT, то параметры канала автоматически изменят свой тип данных на SYSTEM_INT_PARAM.
Последний тег в групповом запросе	Если значение TRUE, то следующий за ним регистр будет читаться в отдельном запросе. По умолчанию значение FALSE.
Длина строки в символах	Задается длина строки в символах. •ASCII - 2 символа в регистре; •HEX - 4 символа в регистре; •Unicode - 1 символ в регистре. Если задать нечетное значение, то оно будет округленно до следующего четного. Если при записи HEX строки символ не будет соответствовать указанному типу, то он будет заполнен 0.
Тип строки	Указывается тип читаемых строк. Поддерживаются типы: •ASCII; •HEX; •Unicode.

Регион задаётся в соответствии со спецификацией Modbus, предоставляется производителем подключаемого устройства, и может быть только одним из четырёх типов, приведённых в таблице ниже. Иногда производители указывают не регион, а номер функции. Соответствие регионов и номеров функций приведено в таблице 13.

Таблица 13

Тип регион	Тип значения региона	Номер функции	
		Чтение	Запись
COILS	Bool	1 (0x01)	5 (0x05) - одиночная; 15 (0x0F) - групповая
DISCRETE_INPUTS	Bool	2 (0x02)	

Тип регион	Тип значения региона	Номер функции	
		Чтение	Запись
HOLDING_REGISTERS	Word	3 (0x03)	6 (0x06) - одиночная; 16 (0x10) - групповая
INPUT_REGISTERS	Word	4 (0x04)	

Например, если производитель указал, что значение должно читаться функцией 1, то необходимо указать регион COILS, функцией 2 - DISCRETE_INPUTS и т.д.

Запись может осуществляться функциями одиночного (5 или 6 - в зависимости от региона) или группового (15 или 16) доступа (в том случае, когда нужно изменить значения нескольких регистров, адреса которых следуют подряд).

Для использования одиночной записи необходимо в настройках устройства (модуля) на панели свойств установить флаг «Использовать WRITE_SINGLE_COIL (0x5)» или «Использовать WRITE_SINGLE_REGISTER (0x6)».

Для чтения всегда используются групповые запросы. Максимальное количество записей в запросе определяется настройкой модуля «Максимальное количество регистров в запросе чтения» или «Максимальное количество битов в запросе чтения».

Если настройки сделаны верно, то при подключении к устройству и загрузке в него конфигурации должна появится связь с Modbus-устройством.

Для того чтобы КТСИ работал в роли Slave по протоколу Modbus RTU, необходимо настроить группу узла «Внешние каналы» для передачи данных.

В качестве Master может выступать и другое устройство (КТСИ), на котором установлена исполнительная система MasterSCADA 4D.

Необходимо задать адреса Modbus и определить, какая функция будет использоваться мастером для чтения/записи данных. Для этого в панели настроек каналов необходимо задать поля «Адрес» и «Тип значения» (см. рисунок 44).

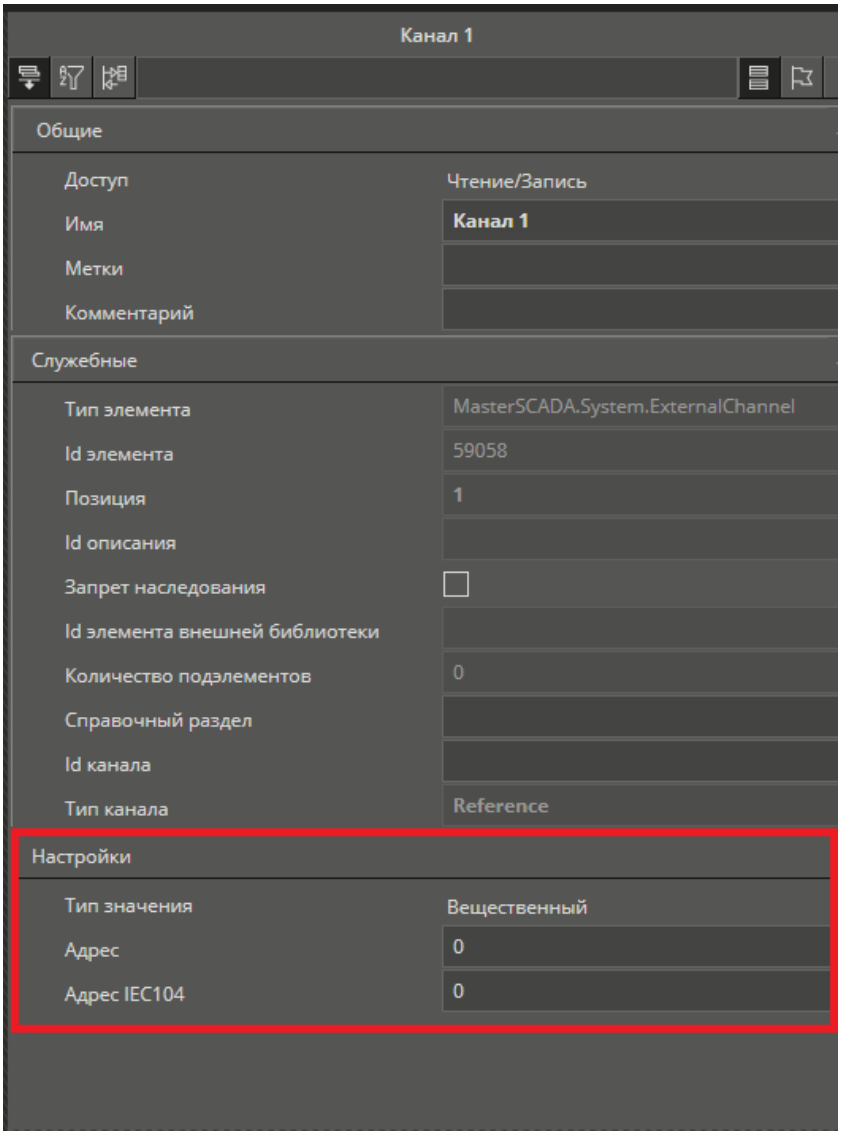


Рисунок 44

Важно! Поддерживается только простые типы данных. Массивы, структуры, строки, время не поддерживаются.

Адреса могут формироваться автоматически (рекомендуется). Для этого необходимо в контекстном меню группы «Внешние каналы» выбрать пункт меню «Создать карту Modbus» (см. рисунок 45).

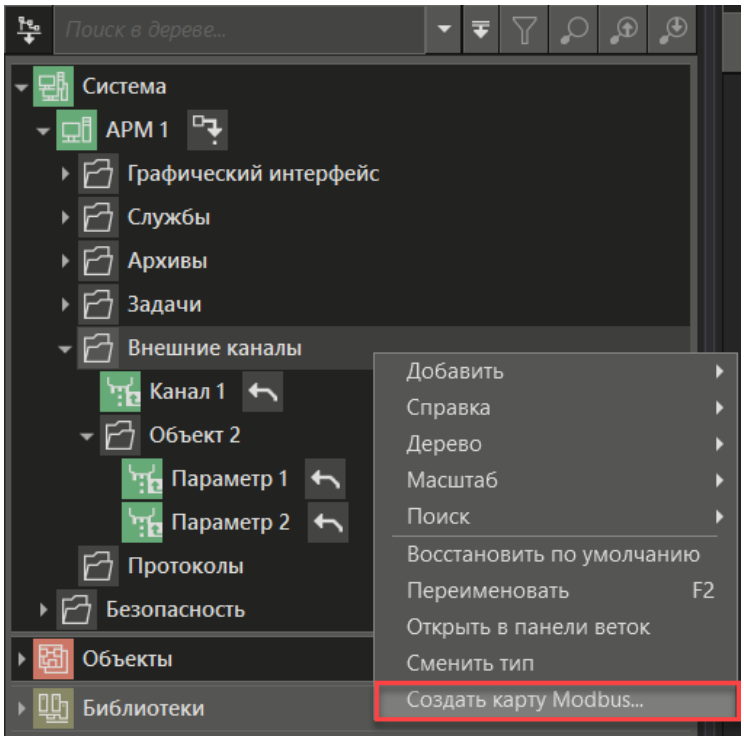


Рисунок 45

При этом откроется диалоговое окно, в котором нужно указать место хранения и имя csv-файла. Вид csv-файла приведен на рисунке 46.

	A	B	C	D	E	F	G	H
1	Name	Region	Address	DataType	Access	ByteOrder	Comment	
2	Канал 1	HOLDING_REGISTERS	0	double	ReadWrite	32107654		
3	Объект 2{g}Параметр 1	HOLDING_REGISTERS	4	double	ReadWrite	32107654		
4	Объект 2{g}Параметр 2	HOLDING_REGISTERS	8	double	ReadWrite	32107654		
5								

Рисунок 46

Адресное пространство для логических и вещественных переменных различается. Система будет выбирать тот или иной канал для получения/отправки, в зависимости от того какой функцией она будет опрошена.

Чтобы устройство, запрограммированное MasterSCADA 4D, отвечало по Modbus RTU Slave, необходимо в настройке узла «Параметры запуска RT» задать строку параметров, как показано на рисунке 47.

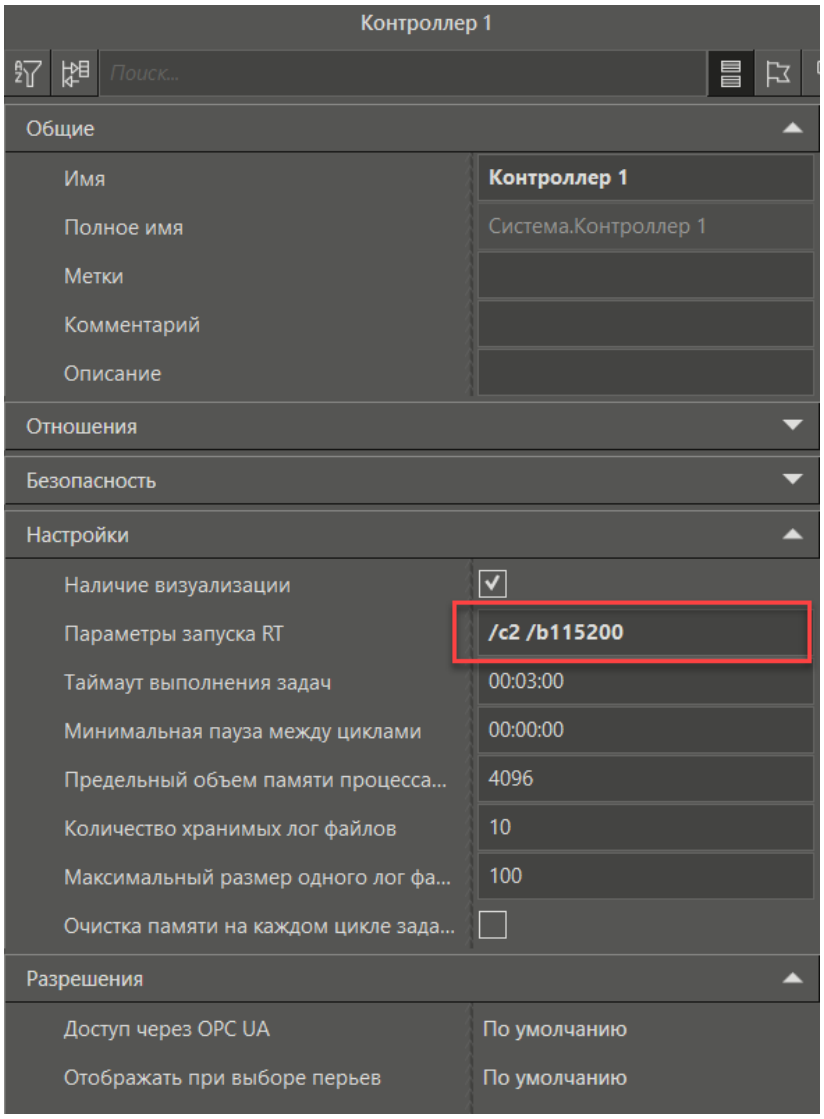


Рисунок 47

Настройка имеет вид /cN /bS, где N - номер порта, а S -скорость.

При этом адрес устройства по умолчанию будет равен 1. Если к одному com-порту подключено несколько устройств, запрограммированных средствами MasterSCADA 4D, то необходимо для каждого устройства задать свой адрес, тогда настройка «Параметры запуска RT» будет иметь вид: /c2 /b115200 /a2, где a2 - адрес устройства.

В случае, если в качестве клиента используется MasterSCADA 4D, т.е. когда создаются два автономных проекта, связанных между собой через протокол Modbus, то в проект клиента можно добавить карту переменных автоматически.

Для этого необходимо воспользоваться «Диалогом импорта каналов Modbus» в редакторе каналов (см. рисунок 48).

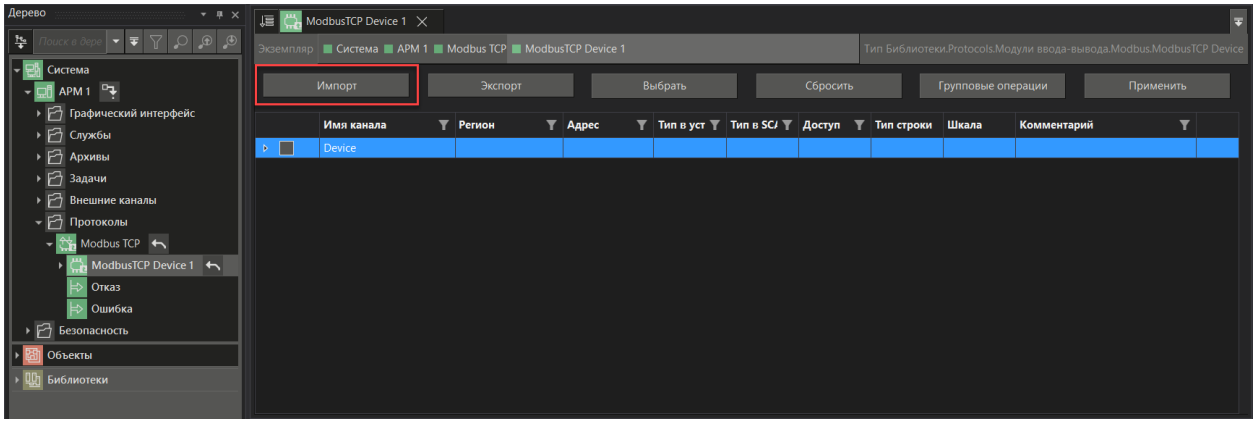


Рисунок 48

После нажатия на кнопку «Импорт» откроется диалоговое окно. Необходимо выбрать csv-файл, который получился в результате автоматического создания карты переменных в проекте, служащим поставщиком данных.

3.3.2 Протокол Modbus TCP

Modbus TCP – это протокол обмена по Modbus по сети по протоколу TCP/IP. При обмене узел играет роль ведущего (Master).

Чтобы добавить возможность опроса устройств по данному протоколу необходимо выбрать соответствующий элемент в контекстном меню узла, или в контекстном меню группы Протоколы, либо через контекстную панель узла (см. рисунок 49).

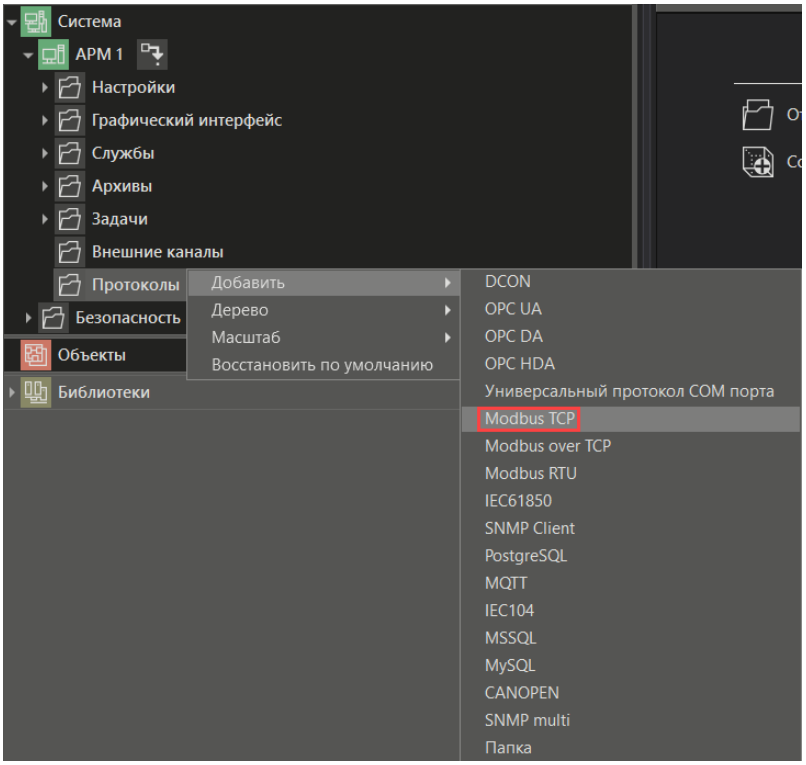


Рисунок 49

Получим результат в упрощенном дереве (см. рисунок 50).

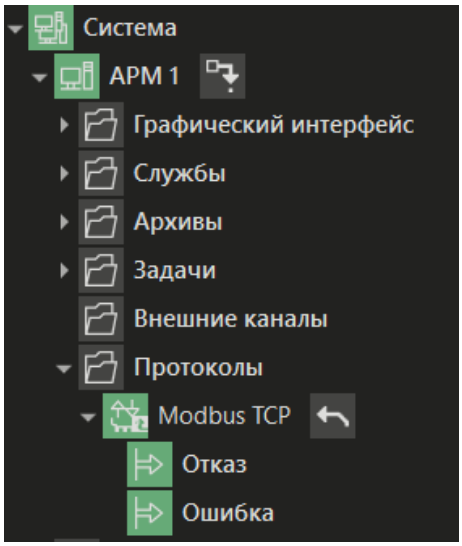


Рисунок 50

Протокол имеет параметр «Отказ». Значение данного параметра зависит от настройки «Формировать отказ при отказе всех модулей». При значении настройки TRUE параметр Отказ примет значение TRUE только тогда, когда откажут все модули. В случае использования резервирования переход на резервный сервер произойдет только после отказа всех модулей.

Далее нужно добавить и настроить модуль ModbusTCP Device, который работает по этому протоколу.

При необходимости, в протоколе могут быть созданы программы, окна и др. элементы. В этом случае у протокола появится группа «Ресурсы», такая же, как и у элементов объект, тег, канал.

Настройка элементов производится в панели свойств. Вид панели свойств модуля ModbusTCP Device показан на рисунке 51. Описание элементов панели свойств ModbusTCP Device приведено в таблице 14.

ModbusTCP Device 1

Поиск...

Общие

Служебные

Задача

Отношения

Асинхронный опрос

Использовать асинхронный опрос	☐
Количество асинхронных запросов чтен...	10
Количество асинхронных запросов запи...	10
Прерывать цикл чтения при превышени...	☐

Масштабирование

Настройки записи

Использовать WRITE_SINGLE_COIL (0x5)	☐
Использовать WRITE_SINGLE_REGISTER (0...	☐
Выполнять запись при подключении	☑

Настройки каналов

Регион	HOLDING_REGISTERS
Адрес	0
Тип данных в устройстве	INT
Последний тег в групповом запросе	☐
Длина строки в символах	1
Тип строки	ASCII

Настройки резервирования

Использовать резервирование	☐
IP адреса	
Порты	
Выставлять "Отказ" при переключении	☐

Настройки связи

IP адрес	127.0.0.1
IP port	502
Адрес	1
Таймаут	1000
Количество попыток	3
Реинициализация порта при ошибке	☐

Настройки чтения

Максимальное количество регистров в з...	125
Максимальный интервал неиспользуем...	0
Максимальное количество битов в запр...	2000
Максимальный интервал неиспользуем...	0

Разрешения

Доступ через OPC UA	По умолчанию
Отображать при выборе перьев	По умолчанию

Чередование байт

DINT-DWORD	_3_2_1_0
REAL	_3_2_1_0
LREAL	_7_6_5_4_3_2_1_0
STRING	_1_0

Рисунок 51

Таблица 14

Название	Описание
Категория **Общие**	Как правило, для данного элемента не настраивается.
Категория **Задача**	Задается способ опроса устройства.
Категория **Асинхронный опрос**	
Использовать асинхронный опрос	При значении TRUE включается Асинхронный опрос. Асинхронный опрос работает по принципу отправки запроса, не дожидаясь ответа от предыдущего, что позволяет ускорять опрос. По умолчанию значение настройки FALSE. Рекомендуется использовать следующим образом. При первом запуске использовать обычный опрос (с выключенным асинхронным опросом). После этого посмотреть статистику контроллера и проверить Общее время задачи. Далее перезапустить проект на исполнение, но уже с включенным асинхронном опросом. Если данные поступают, то повторно проверяем Общее время задачи. Если оно стало меньше, чем при первом запуске, то рекомендуется оставить асинхронный режим включенным. Для дополнительного ускорения получения данных рекомендуется включить настройку. Прерывать цикл чтения при превышении времени задачи. Важно! Функция асинхронного опроса поддерживается не всеми устройствами Modbus. Возможность использования определяется документацией на подключаемое устройство.
Количество асинхронных запросов чтения	Задается максимальное количество запросов в очереди на чтение. По умолчанию значение равно 10.
Количество асинхронных запросов записи	Задается максимальное количество запросов в очереди на запись. По умолчанию значение равно 10. Важно! Если при асинхронной записи пропадет связь с устройством, то все запросы на запись будут удалены.
Прерывать цикл чтения при превышении времени задачи	При значении TRUE результаты запросов будут обновляться каждый период выполнения задачи, не дожидаясь выполнения всех запросов из очереди. По умолчанию значение настройки FALSE. Рекомендуется использовать данную настройку, если необходимо как можно чаще получать и записывать данные.
Категория **Масштабирование**	Если Modbus- устройство не имеет встроенного масштабирования, то при помощи данной категории можно задать соответствие между значениями в модуле и реальными значениями.
Категория **Настройки записи**	
Использовать WRITE_SINGLE_COIL (0x5)	При значении TRUE используется функция записи в ячейку региона Coils - 0x05 (запись в одиночную ячейку). При значении FALSE используется функция записи в группу ячеек региона Coils - 0x0F (групповая запись). По умолчанию значение настройки FALSE.
Использовать WRITE_SINGLE_REGISTER (0x6)	При значении TRUE используется функция записи в ячейку региона Holding Registers - 0x06 (запись в одиночный регистр). При значении FALSE используется функция записи в группу регистров региона Holding Registers - 0x10 (групповая запись).

	По умолчанию значение настройки FALSE.
Выполнять запись при подключении	При значении TRUE будет производиться однократная запись значений на всех выходах при следующих ситуациях: •при каждом подключении к устройству, включая первоначальное; •при переключении исполнения на резервный узел в резервированной паре (запись будет выполняться даже при значении FALSE у настройки «Выполнять запись на резервном»); •при переподключении после разрыва связи. При этом не имеет значения, совпадает ли текущее значение с начальным, запись будет выполнена независимо от этого. По умолчанию значение настройки FALSE.
Категория **Настройки каналов**	Задаются значения по умолчания для создаваемых каналов. В дальнейшем эти настройки у каналов можно будет изменить.
Категория **Настройки резервирования**	
Использовать резервирование	При значении TRUE включается режим резервирования каналов связи. В штатном режиме опрос производится по основному каналу, заданному в поле IP адрес. Если возникает отказ устройства (отсутствие ответа) и на входе параметра «Установить канал» задано -1 (автоматический режим), то происходит переход к следующему резервному каналу. Номер канала, к которому происходит попытка подключения указывается в параметре Текущий канал. Если подключения по всем каналам были неудачными, то у модуля выставляется флаг Отказ, попытки подключения продолжаются по кругу - начиная с основного канала. По умолчанию значение настройки FALSE.
IP адреса	Данная настройка используется только при значении TRUE у настройки «Использовать резервирование». Для резервирования каналов указываются несколько IP адресов через запятую. Переключение на резервный IP адрес произойдет, когда модулю будет выставлен Отказ после достижения максимального количества попыток подключения, указанного в настройке Количество попыток.
Порты	Данная настройка используется только при значении TRUE у настройки «Использовать резервирование». Для резервирования каналов указываются несколько портов через запятую. Если количество IP адресов будет больше, чем портов, то будет использоваться порт основного канала. Если для резервированных каналов используется такой же порт как для основного, то данное поле можно оставить пустым.
Вставлять "Отказ" при переключении	При значении TRUE на время переключения канала всем тегам будет выставлен Отказ. Если флаг снят, то отказ устанавливается когда попытки подключения по всем каналам были неудачными. По умолчанию значение настройки FALSE.
Категория **Настройки связи**	
IP адрес	Задается IP-адрес устройства (значение IP-адреса определяется документацией на подключаемое устройство).
IP порт	Задается TCP-порт устройства (номер порта определяется документацией на подключаемое устройство).

Адрес	Задается адрес устройства (значение адреса определяется документацией на подключаемое устройство).
Таймаут	Максимальное время ожидания ответа от устройства (мс). Если ответ от модуля не будет получен в течении заданного интервала времени, то попытка опроса будет считаться неудачной, и зафиксируется ошибка данного запроса для конкретного модуля
Количество попыток	Задается максимальное количество попыток запроса к устройству. После достижения указанного значения, устройству будет выставлен Отказ.
Реинициализация порта при ошибке	Если значение настройки TRUE, то при возникновении Отказа у всех устройств на шине, будет происходить переоткрытие порта. По умолчанию значение настройки FALSE.
Категория **Настройки чтения**	
Максимальное количество регистров в запросе чтения	Задается максимальное значение Input и Holding регистров для одного запроса чтения. По умолчанию значение равно 125 (максимально допустимое).
Максимальный интервал неиспользуемых регистров	Задается максимальное значение для разрыва между адресами регистров. Если разрыв меньше указанного значения, то происходит объединение в один запрос. Например, производится опрос регистров с номерами - 0, 1, 2, 4, 5. Если параметр будет установлен в 0, то регистры будут запрошены за два запроса (0-2, 4-5), если же установить его в 1 или выше, то за один запрос (0-5, а не использованный регистр 3 будет отброшен при разборе ответа). Увеличение данного параметра сокращает количество запросов и ускоряет опрос, но удлиняет кадр ответа. Также следует помнить, что некоторые Modbus устройства имеют ограничение на количество передаваемых за один запрос параметров.
Максимальное количество битов в запросе чтения	Задается максимальное значение Coils и Discrete Inputs битов для одного запроса чтения. По умолчанию значение равно 2000 (максимально допустимое).
Максимальный интервал неиспользуемых битов	Задается максимальное значение для разрыва между адресами битов. Если разрыв меньше указанного значения, то происходит объединение в один запрос. Также следует помнить, что некоторые Modbus устройства имеют ограничение на количество передаваемых за один запрос параметров.
Категория **Чередование байт**	
DINT-DWORD	Порядок следования байтов в целых числах. Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в целых числах. Доступны чередования: • _3_2_1_0 • _1_0_3_2 • _0_1_2_3 • _2_3_0_1
REAL	Порядок следования байтов в вещественных числах. Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в вещественных числах. Доступны чередования: • _3_2_1_0 • _1_0_3_2 • _0_1_2_3 • _2_3_0_1

LREAL	Порядок следования байтов в вещественных числах (8 байт). Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в вещественных числах. Доступны чередования: • _1_0_3_2_5_4_7_6 • _3_2_1_0_7_6_5_4 • _7_6_5_4_3_2_1_0 • _0_1_2_3_4_5_6_7
LINT-LWORD	Порядок следования байтов в целых числах (8 байт). Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в целых числах. Доступны чередования: • _1_0_3_2_5_4_7_6 • _3_2_1_0_7_6_5_4 • _7_6_5_4_3_2_1_0 • _0_1_2_3_4_5_6_7
STRING	Порядок следования байтов в строках. Данный параметр обеспечивает обмен с устройствами с различным порядком следования байтов в строках. Доступны чередования: • _1_0 • _0_1

Далее необходимо настроить каналы для чтения/записи. Каналы настраиваются аналогично протоколу Modbus RTU (см. п. 3.3.1 настоящего руководства).

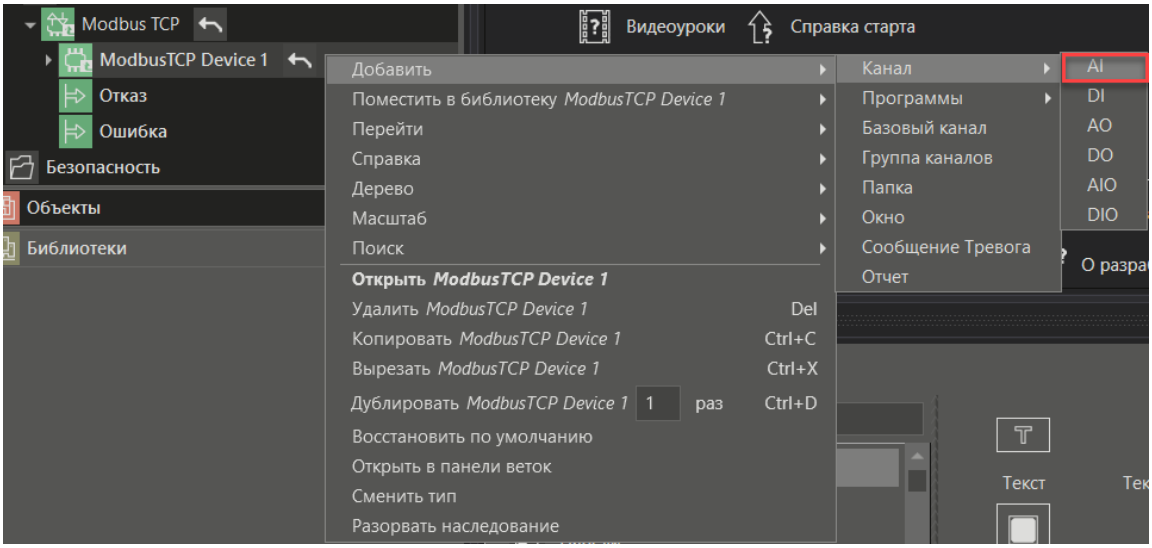


Рисунок 52

3.3.3 Протокол OPC UA

КТСИ с MasterSCADA 4D поддерживает платформонезависимый стандарт OPC UA.

Для того чтобы MasterSCADA 4D выступала в роли клиента, необходимо в группу узла «Протоколы» добавить соответствующий протокол (см. рисунок 53).

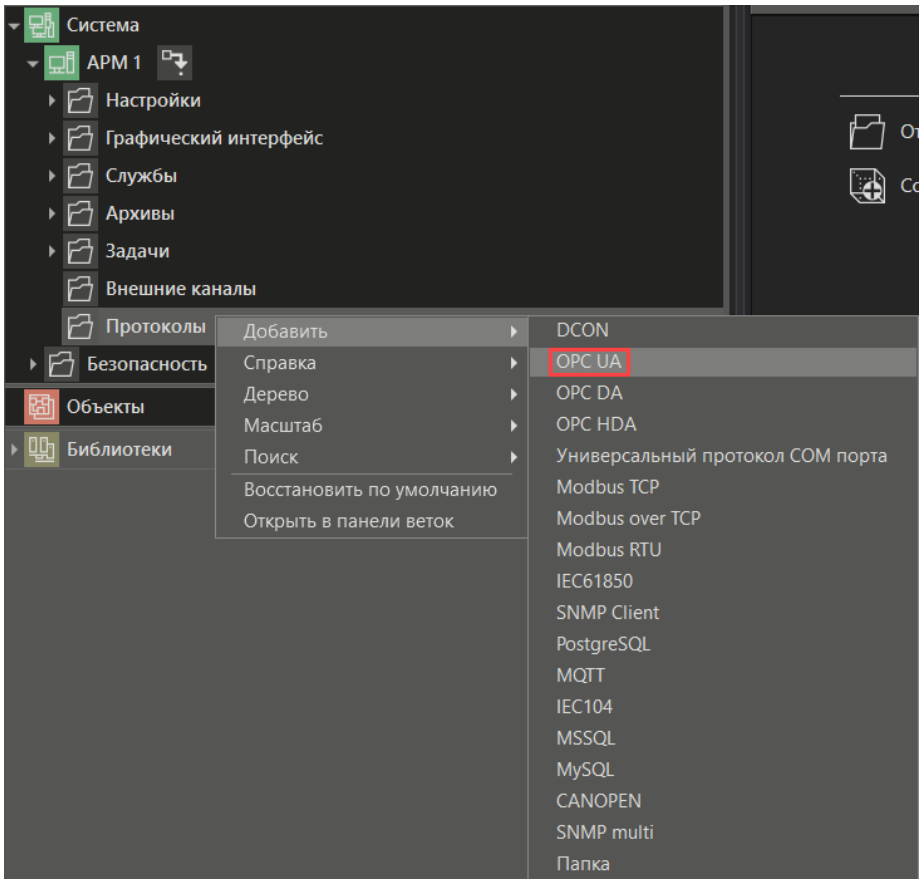


Рисунок 53

Получим результат в упрощенном дереве (см. рисунок 54).

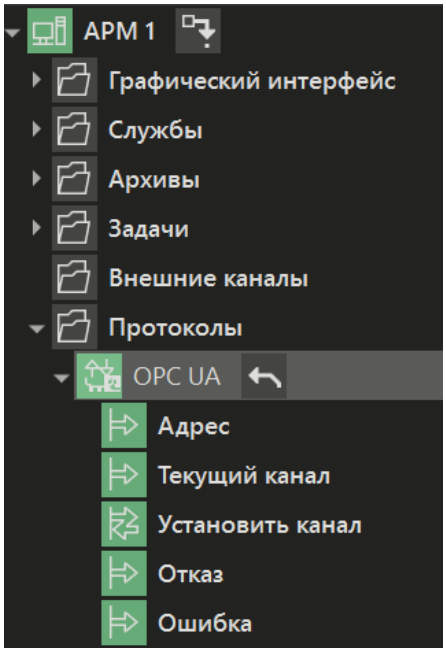


Рисунок 54

Протокол имеет параметр «Отказ». Если он принимает состояние TRUE, то это значит, что исполнительная система не может установить связь со всеми модулями данного узла. При этом если используется резервирование, то произойдет переход на следующий канал.

Также протокол содержит служебные параметры «Адрес», «Текущий канал» и «Установить канал», использующиеся при резервировании каналов.

Параметр «Адрес» выдаёт IP-адрес опрашиваемого сервера.

Параметр «Текущий канал» выдает индекс текущего опрашиваемого адреса, начиная с 0. В случае отсутствия подключения, будет выдаваться значение минус 1.

Параметр «Установить канал» по умолчанию имеет значение минус 1. В этом случае используется автоматическое переключение. Если задан индекс, отличающийся от текущего канала, выполняется принудительное переключение на заданный адрес. Пока установлено значение ≥ 0 на каждом цикле будет выполняться попытка подключения к данному адресу.

Затем необходимо выполнить подключение к OPC UA серверу для добавления каналов.

Для этого нужно дважды нажать левую кнопку мыши по элементу в дереве. При этом откроется вкладка, в которой можно выполнить подключение к OPC UA серверу (см. рисунок 55).

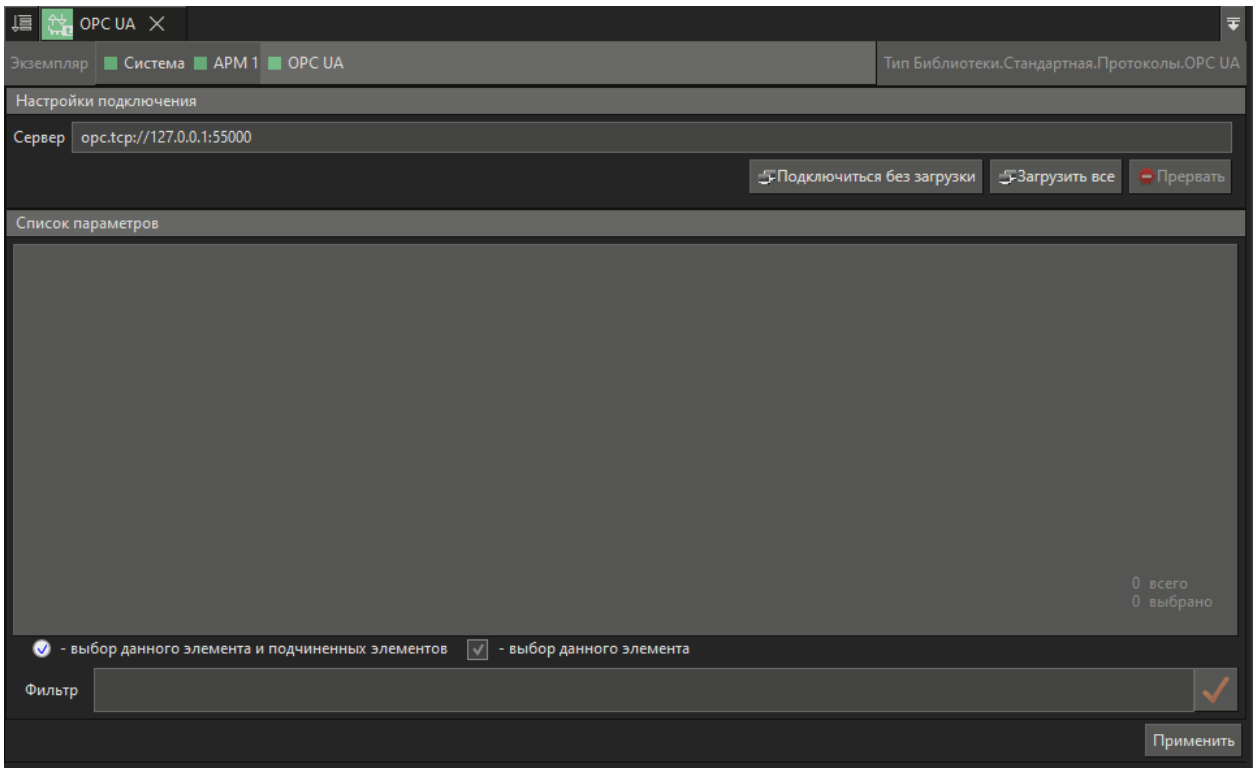


Рисунок 55

В строке «Сервер» необходимо задать IP-адрес и TCP/IP порт сервера в виде:

opc.tcp://[IP-адрес][номер порта].

Пример записи показан в значении по умолчанию: *opc.tcp://127.0.0.1:55000*.

Важно! Перед началом подключения к серверу необходимо убедиться, что сервер запущен в режим исполнения.

Затем следует нажать кнопку «Подключиться». Для отмены подключения необходимо нажать кнопку «Прервать». Если попытка подключения окажется удачной, то в группе «Список параметров» отобразятся параметры подключенного сервера в виде дерева, в котором следует выбрать те элементы, с которыми требуется работать в проекте MasterSCADA 4D.

Далее необходимо нажать кнопку «Применить». После этого в дерево системы добавятся каналы предопределенных типов**,** либо массив.

После подключения к серверу необходимо настроить панель свойств OPC UA сервера.

При работе по протоколу OPC UA создается, если его не было ранее, сертификат в рабочей папке исполнительной системы *CltCertificateStore*.

При работе в роли сервера КТСИ с MasterSCADA 4D поддерживает режимы чтения и подписки. Поддерживаются скалярные типы данных, параметры типа SYSTEM_*_PARAM. Также возможна работа со структурами, одномерными массивами, состоящих из элементов простых типов данных, и одномерными массивами структур.

Для работы КТСИ в режиме OPC UA сервера необходимо включить доступ по OPC UA в настройках узла и указать порт (см. рисунок 56).

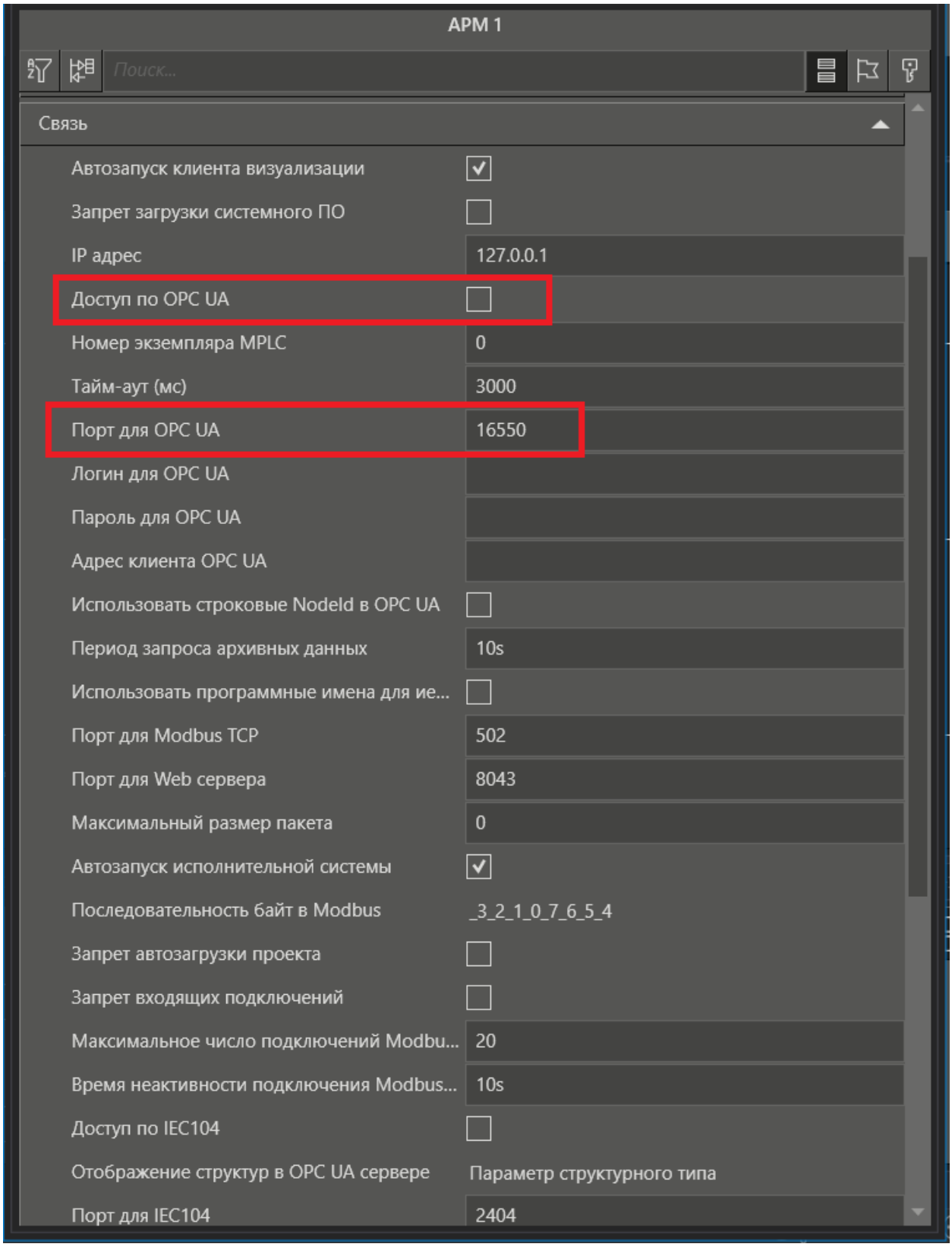


Рисунок 56

Предположим, что в проекте создан узел с параметрами связи, как показано на рисунке 57.

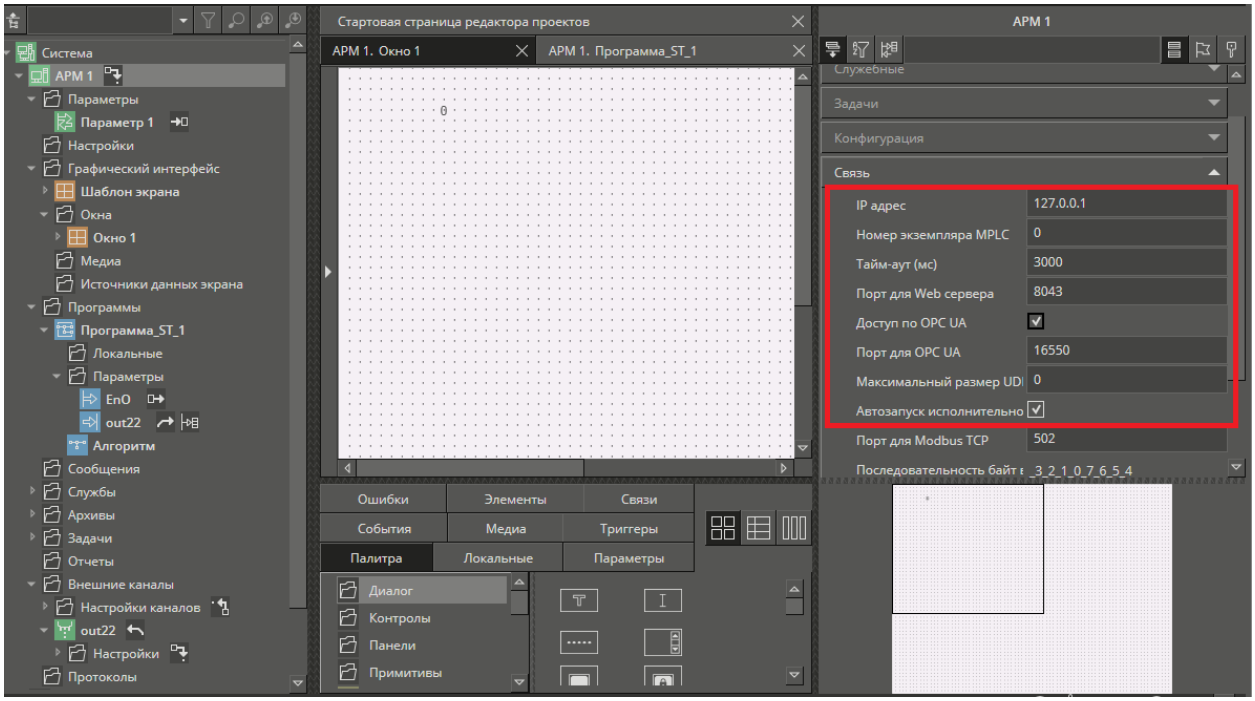


Рисунок 57

Запустим среду исполнения и загрузим в нее текущий проект. Для этого можно в контекстном меню узла выбрать пункт Узел → Подключить с загрузкой конфигурации (см. рисунок 58).

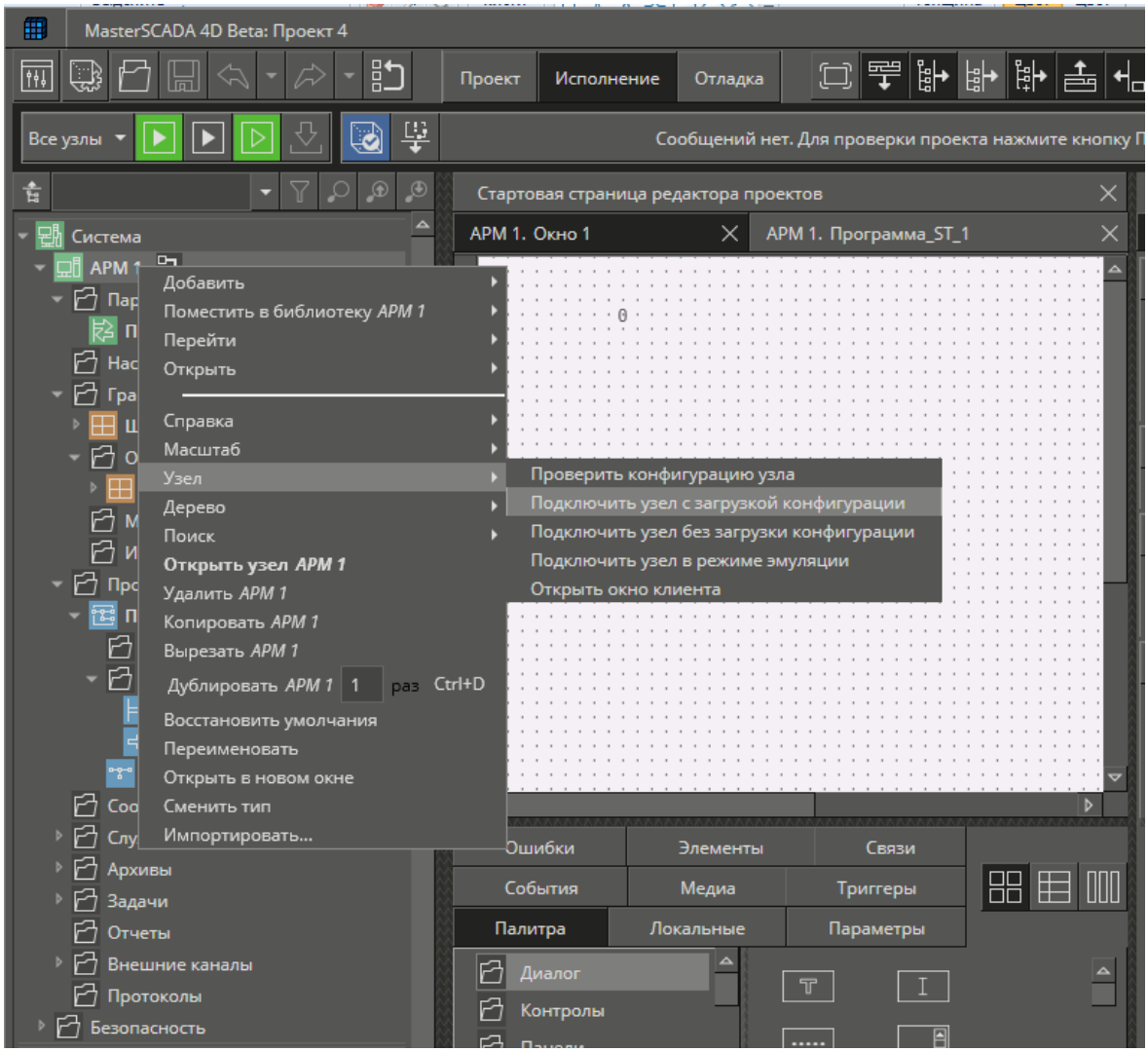


Рисунок 58

Запускаем клиент OPC UA (в примере используется UaExpert) и нажимаем кнопку добавления сервера «Add Server» (см. рисунок 59).

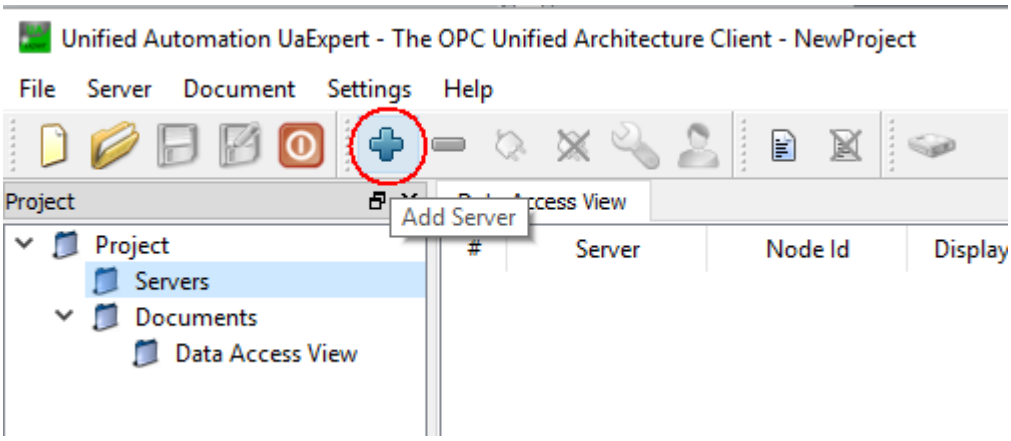


Рисунок 59

На вкладке Advanced открывшегося диалога конфигурируем подключение к исполнительной системе КТСИ с MasterSCADA 4D как к серверу OPC UA (см. рисунок 60).

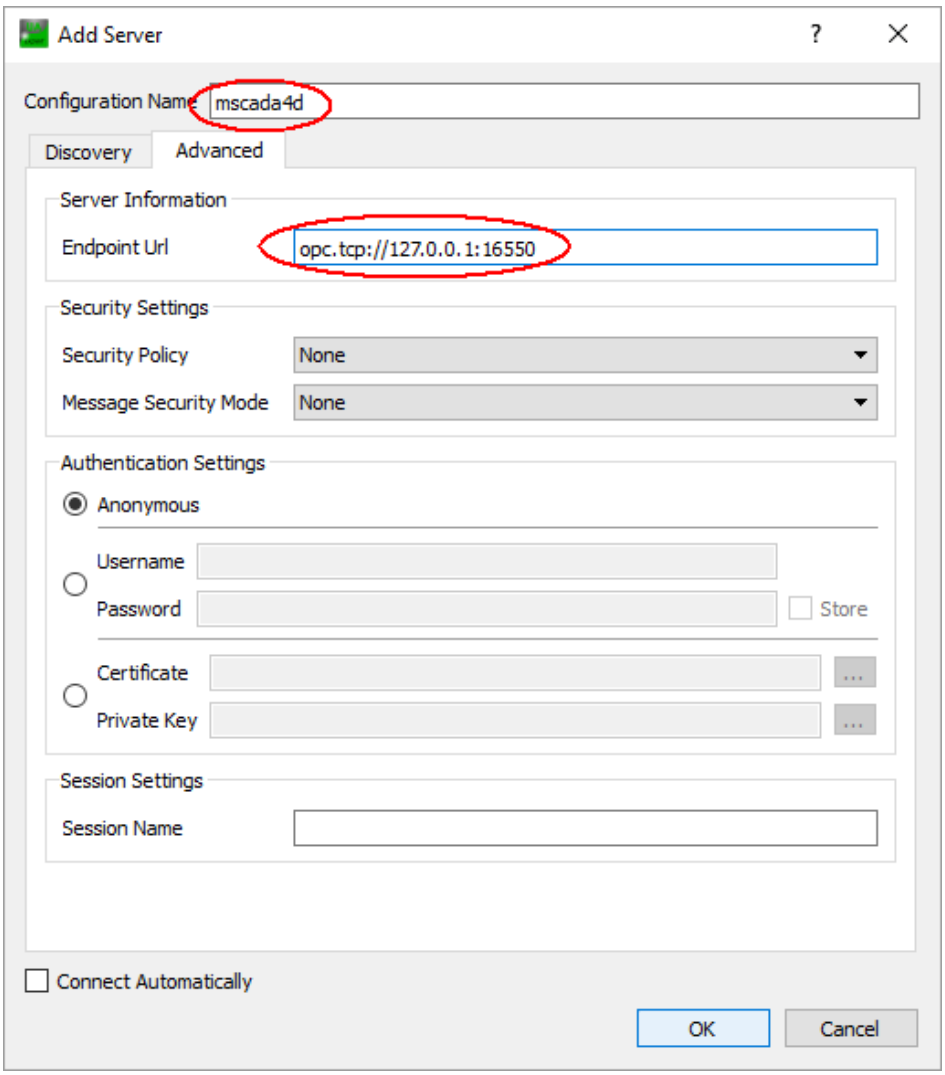


Рисунок 60

Сервер добавляется в дерево «UaExpert». Для подключения к этому серверу выполняем команду «Connect» из его контекстного меню (см. рисунок 61).

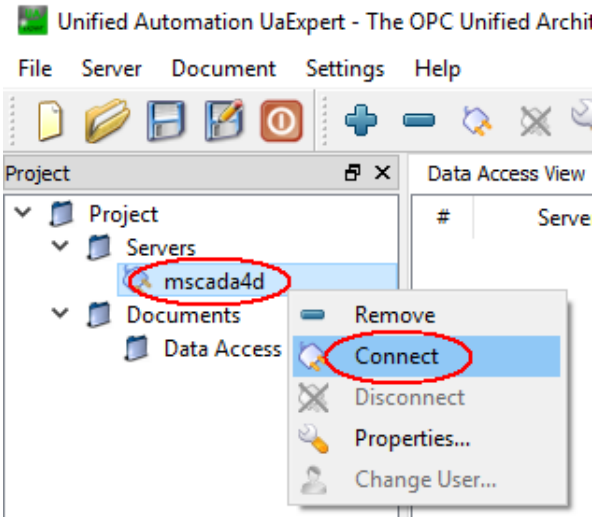


Рисунок 61

После этого открывается сообщение (см. рисунок 62).

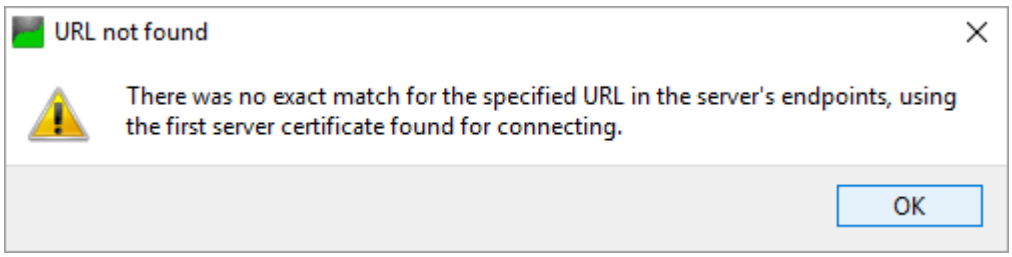


Рисунок 62

Нажимаем «OK» – открывается диалог с сообщением о том, что сертификат OPC UA сервера (т.е. сертификат MasterPLC, исполнительной системы КТСИ) не принадлежит к числу доверенных (см. рисунок 63).

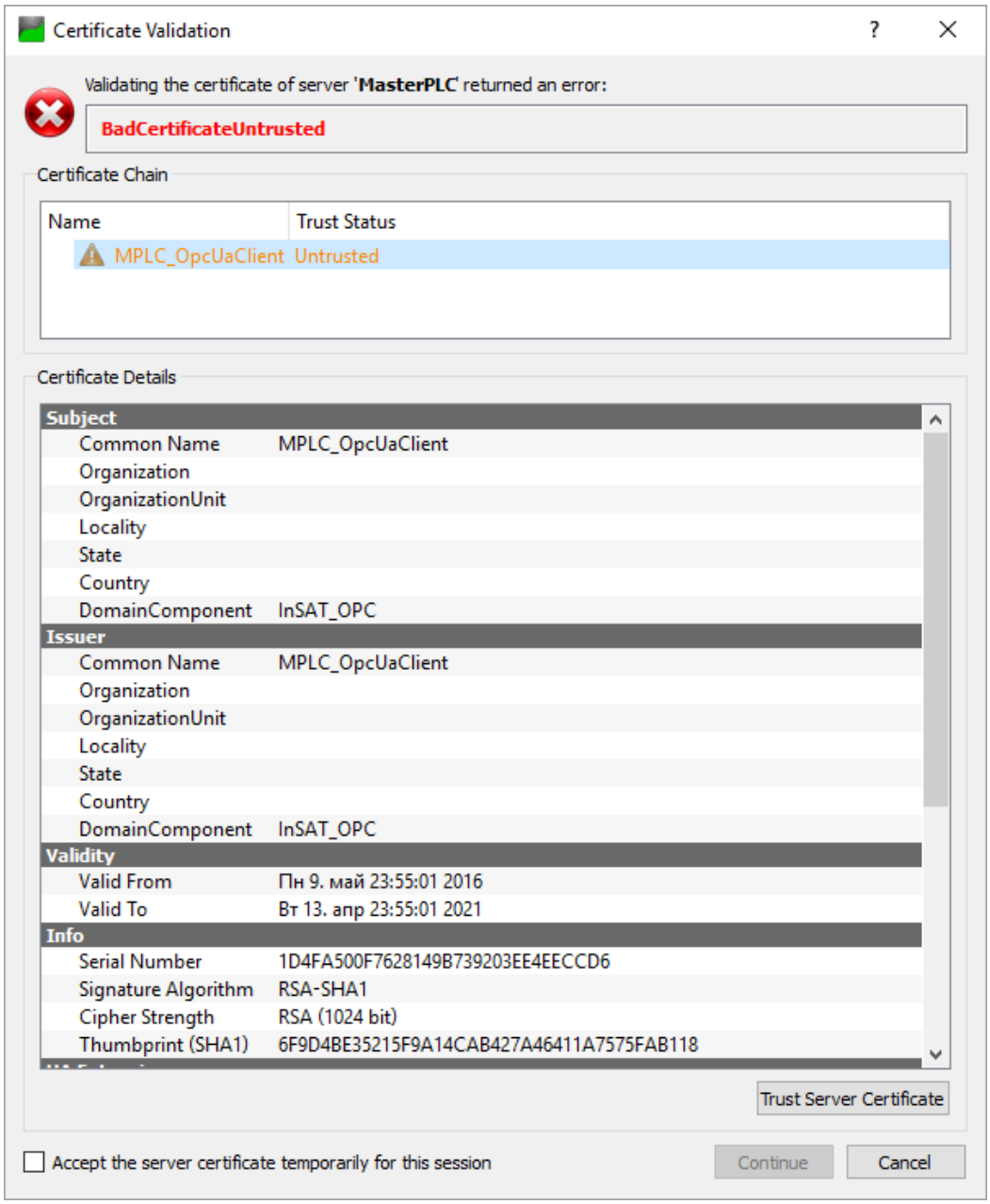


Рисунок 63

Для исправления ситуации нажимаем кнопку «Trust Server Certificate». Откроется окно, показанное на рисунке 64.

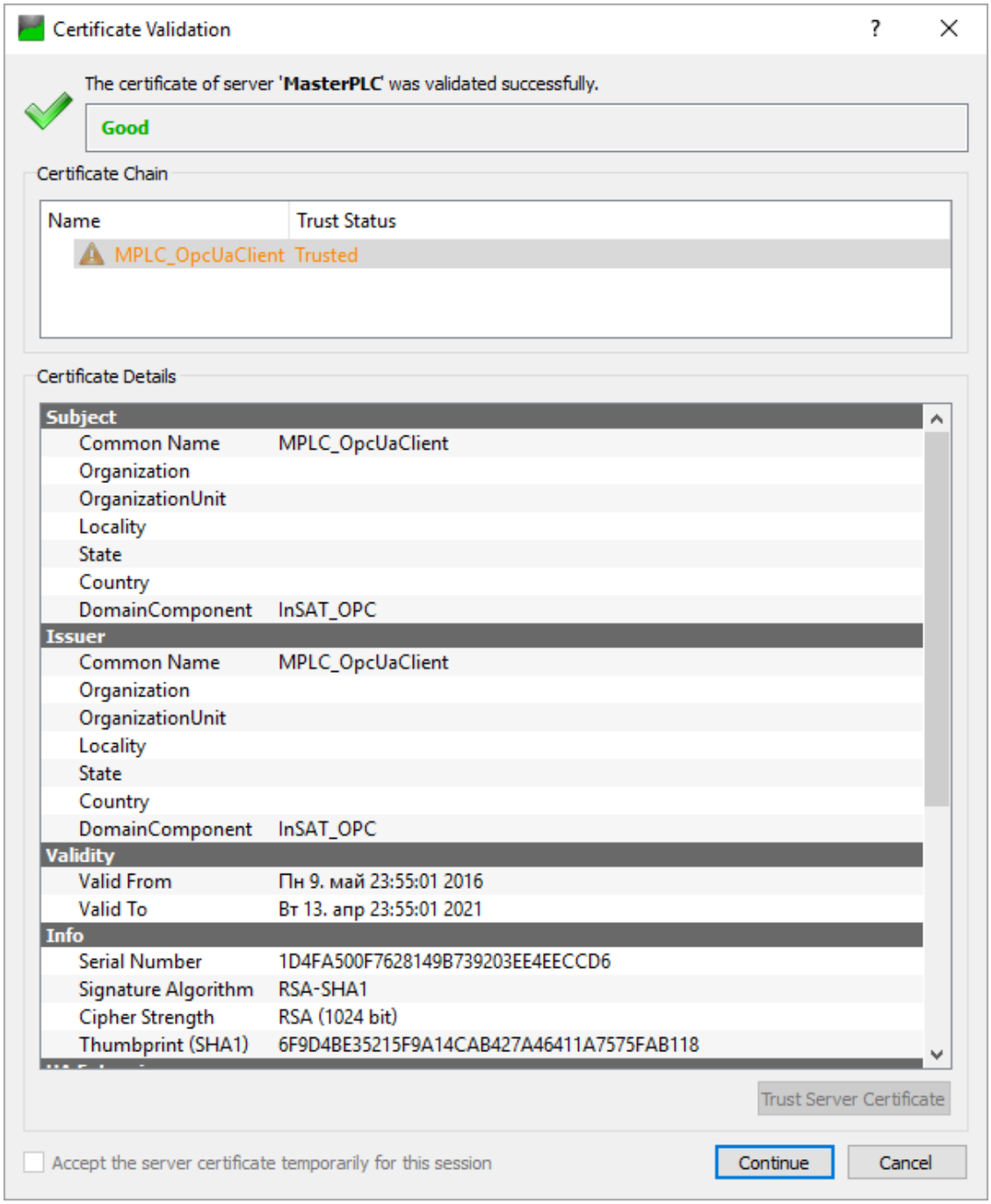


Рисунок 64

Нажимаем кнопку «Continue», после чего выполняется подключение к серверу (см. рисунок 65).

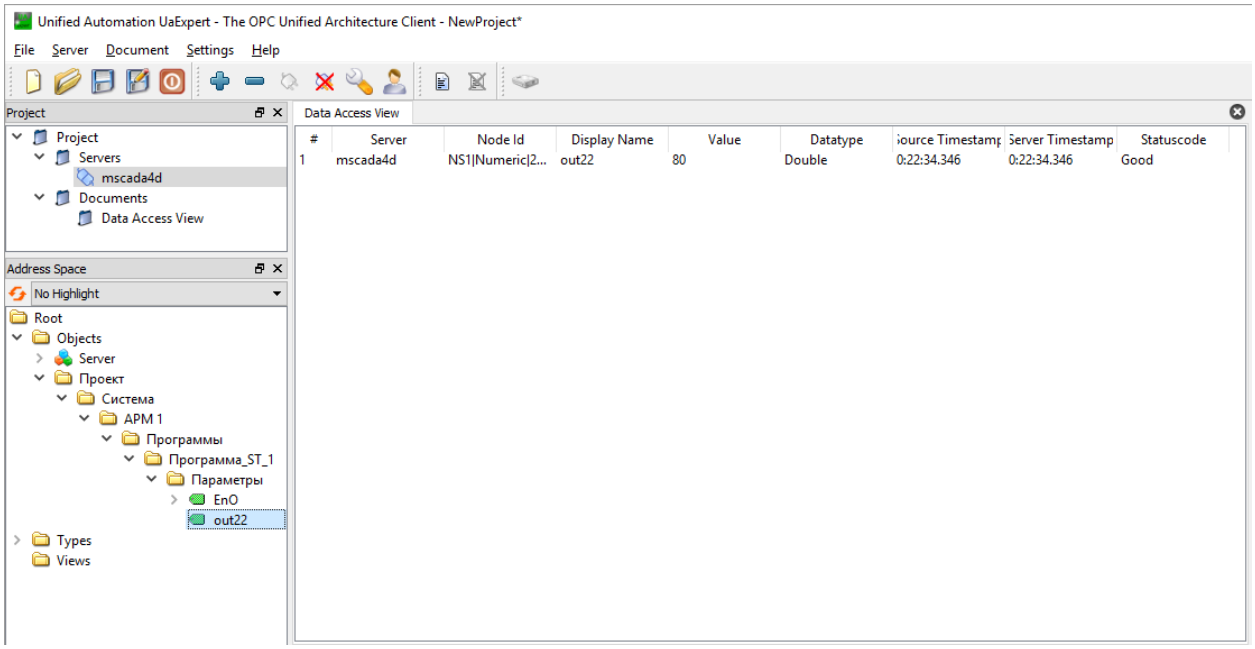


Рисунок 65

Элементы проекта будут отображаться в дереве клиента в зависимости от настройки элементов «Доступ через OPCUA». Способ отображения структур в OPC UA клиенте определяется свойством узла «Отображение структур в OPC UA сервере».

При работе по протоколу OPC UA SLAVE создается, если его не было ранее, сертификат в рабочей папке исполнительной системы CertificateStore.

В клиенте, получающим данные от исполнительной системы КТСИ, можно установить следующие настройки согласно стандарту OPC UA.

Publishing Interval - период, с которым сервер отправляет клиенту пакеты с данными

Queue Size - настройка, задаваемая у тега, указывает максимальное количество значений по этому тегу, передаваемых в одной подписке

SamplingInterval - это частота, с которой вы хотите, чтобы сервер выполнял выборку/опрос/мониторинг элемента. Элементы подписки могут поступать из разных источников, и каждый из них может отбираться по своему усмотрению. Если установлено значение -1 или 0, то клиенту отправляется каждое изменение параметра (не более QueueSize в каждом пакете). Если больше 0, то клиенту отправляются значение, если его метка отличается от предыдущего, отправленного больше на SamplingInterval.

Например, в клиенте установлено: PublishingInterval = 1сек., QueueSize = 10, SamplingInterval = -1, а значение меняется раз в 100мс, то сервер должен отправлять клиенту по 10 значений, но не чаще 1 раз в секунду.

Поддержка параметра у тега подписки MonitoringMode = Sampling. Установка данного параметра означает, что элемент только отбирается, а изменения значений помещаются в очередь, которая есть у каждого отслеживаемого элемента, но об изменениях фактически не сообщается клиенту.

3.3.4 Протокол CANOpen

Напрямую в среде разработки и исполнения нет возможности добавлять опрос устройств по шине CAN (кроме встроенных модулей ввода-вывода), поэтому при добавлении узла Uzola для опроса устройств по внешней can-шине необходимо сделать копию ветки «Встроенные модули» или добавить протокол UzolaCAN из библиотеки Uzola (см. рисунок 66).

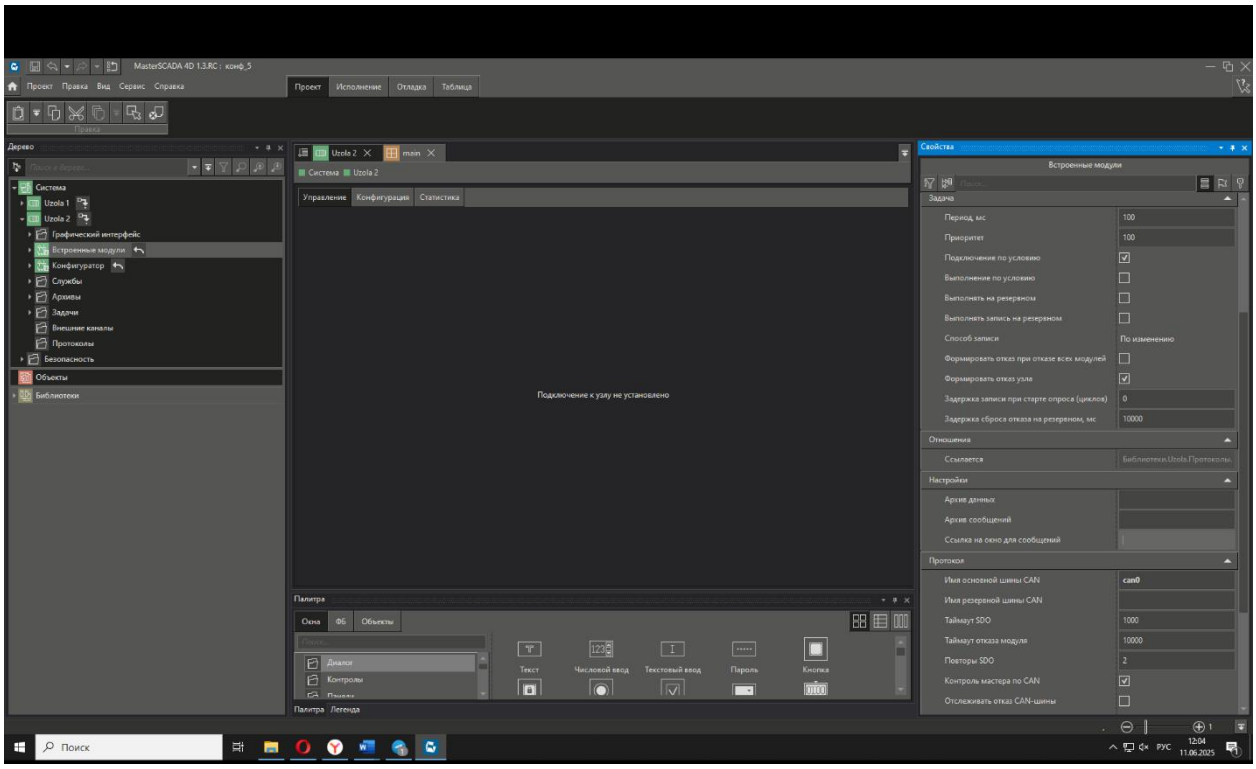


Рисунок 66

В свойствах для использования внешней can-шины необходимо указать имя основной шины – can0. Далее добавить CANOpen модуль, и в свойствах модуля указать NodeID, и название файла описания устройства *файл описания.eds* (см. рисунок 67).

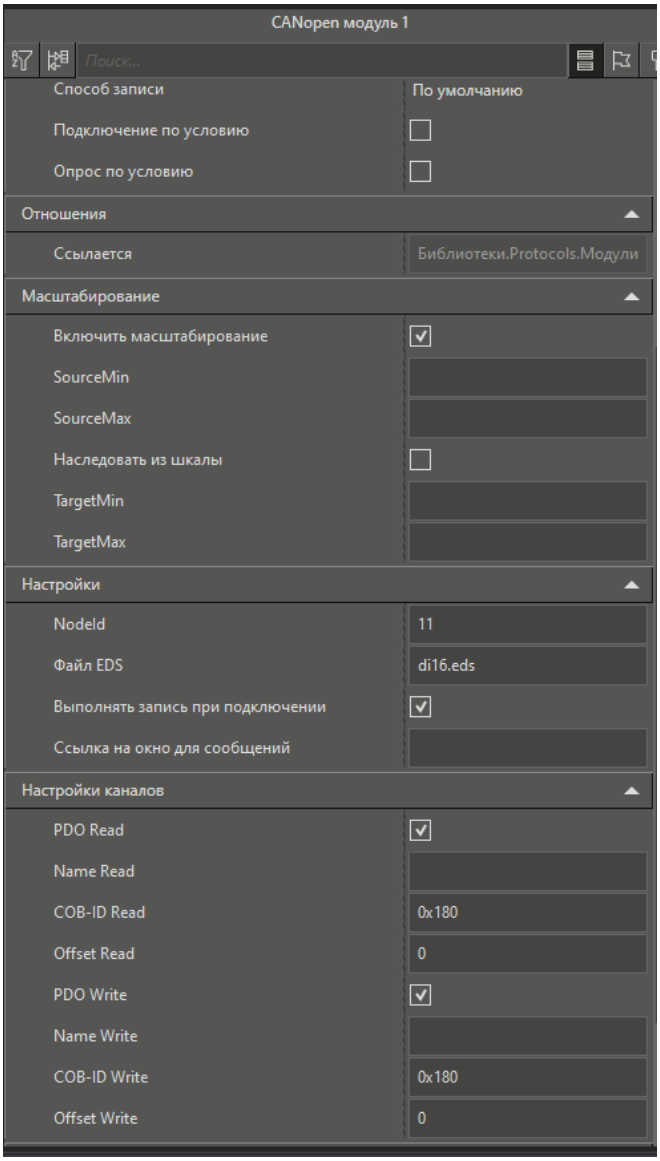


Рисунок 67

Далее необходимо добавить каналы модуля (устройства) для чтения/записи в соответствии с картой адресов объектов PDO/SDO из файла описания устройства и/или документации производителя. Протокол имеет параметр «Отказ». Значение данного параметра зависит от настройки «Формировать отказ при отказе всех модулей». При значении настройки TRUE параметр «Отказ» примет значение TRUE только тогда, когда откажут все модули.

3.4 Разработка пользовательского проекта

3.4.1 Анализ

Прежде чем приступать к созданию проекта в среде разработки MasterSCADA 4D, следует провести анализ объекта автоматизации. Самая главная цель анализа состоит в выделении типовых элементов, встречающихся в проекте неоднократно и представляющих

собой законченные части программы.

Это необходимо сделать для того, чтобы создать в библиотеке полный набор типовых элементов, которые могут быть использованы в дальнейшем при разработке. Все элементы проекта являются наследниками какого-либо библиотечного элемента. Внесение изменений в библиотечный объект повлечет изменение его экземпляров, что позволит сократить время на модернизацию и отладку проекта.

Типовые элементы проекта разрабатываются в дереве библиотек. Если элемент планируется использовать только в рамках текущего проекта, то его можно разрабатывать в библиотеке «Локальная». В этом случае созданные элементы будут видны только в рамках текущего проекта. Если предполагается использовать элементы в нескольких проектах, либо когда разработку типовых элементов начинают несколько пользователей одновременно, то следует создать пользовательскую библиотеку при помощи контекстного меню либо контекстной панели (см. рисунок 68).

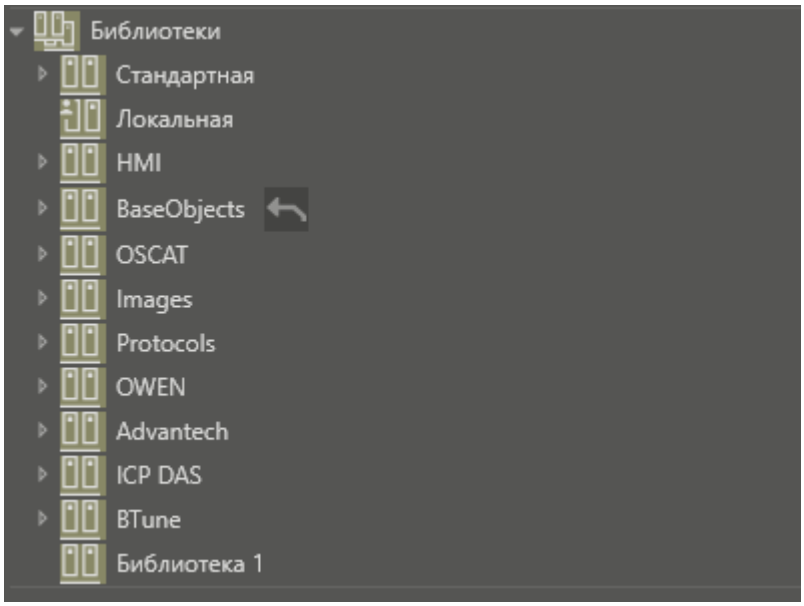


Рисунок 68

Если необходимо создать библиотечный объект, который включает в себя окна и программы, то сначала создается библиотечная программа, библиотечное окно, а затем уже библиотечный объект

По такому принципу сделана одна из нескольких библиотек, включенных в поставку MasterSCADA 4D по умолчанию, - BaseObjects.

Так, например, если проанализировать объект «Насос», то можно заметить, что сначала были созданы окна и ФБ, а затем из этих элементов собрали библиотечный объект (см. рисунок 69).

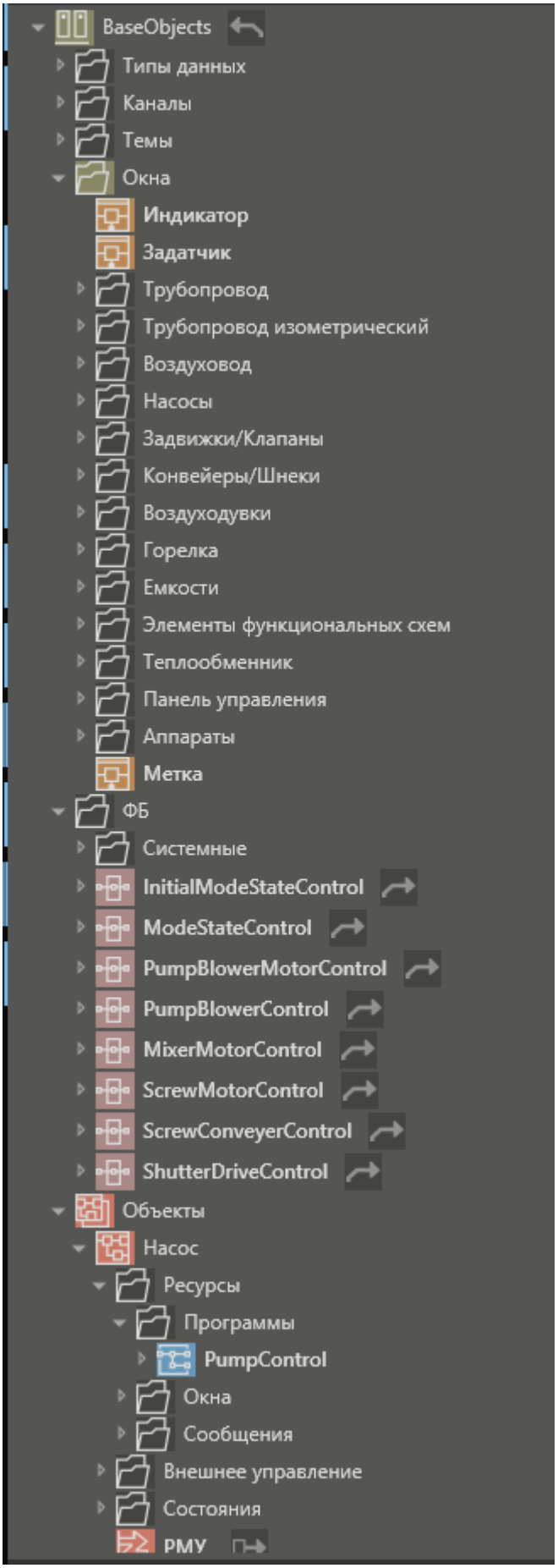

BaseObjects
Типы данных
Каналы
Темы
Окна
Индикатор
Задатчик
Трубопровод
Трубопровод изометрический
Воздуховод
Насосы
Задвижки/Клапаны
Конвейеры/Шнеки
Воздуходувки
Горелка
Емкости
Элементы функциональных схем
Теплообменник
Панель управления
Аппараты
Метка
ФБ
Системные
InitialModeStateControl
ModeStateControl
PumpBlowerMotorControl
PumpBlowerControl
MixerMotorControl
ScrewMotorControl
ScrewConveyerControl
ShutterDriveControl
Объекты
Насос
Ресурсы
Программы
PumpControl
Окна
Сообщения
Внешнее управление
Состояния
РМУ

Рисунок 69

3.4.2 Библиотечные элементы

В MasterSCADA 4D проект создается из библиотечных элементов (см. рисунок 70):

- «Система»;

- «Объекты».

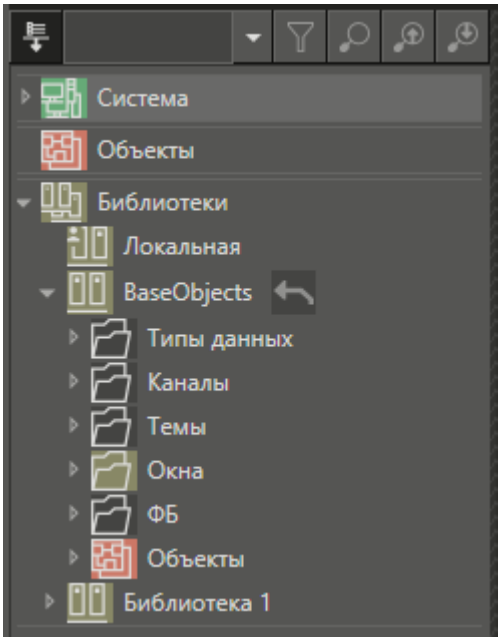


Рисунок 70

3.4.2.1 В группе «Система» создаются узлы - конфигурации, загружаемые для исполнения в реальные устройства проекта (АРМы, контроллеры и т. д.).

Для создания узла можно использовать контекстное меню (см. рисунок 66), контекстную панель или метод перетаскивания (drag-n-drop) из соответствующей библиотеки, в которой определена начальная структура поддерживаемых устройств (см. рисунок 71).

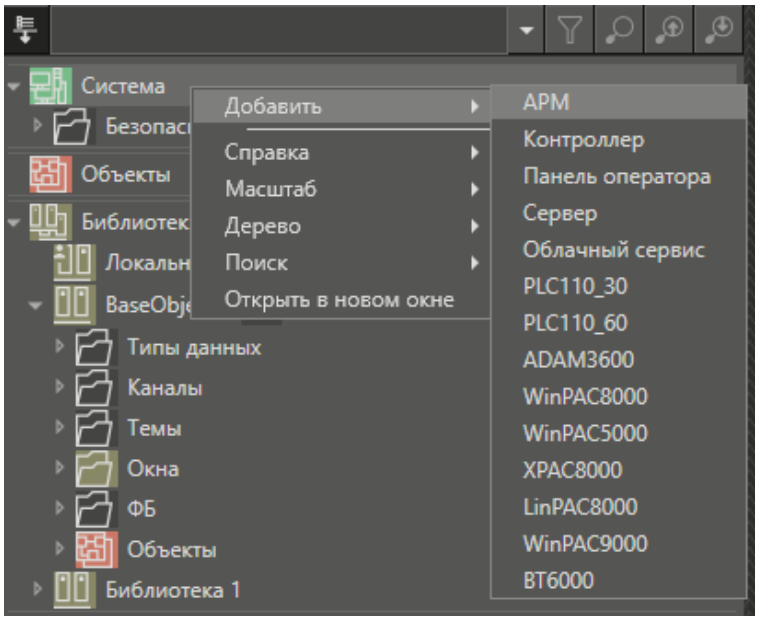


Рисунок 71

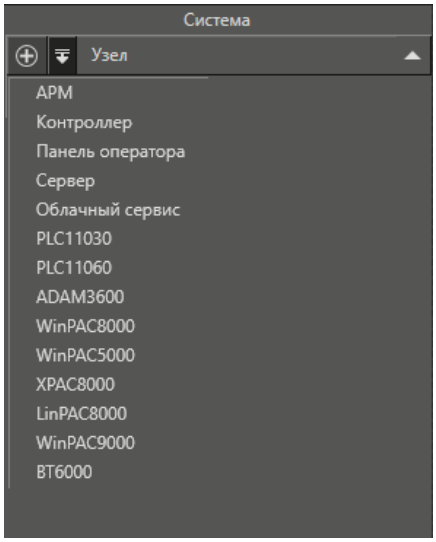


Рисунок 72

В узле конфигурируются его внешние связи с устройствами и/или OPC-серверами (указываются устройства, протоколы, каналы связи).

Для конфигурирования взаимодействия с оборудованием используется интуитивно понятная логика аппаратно-программного информационного потока**:** узел – протокол – устройство (модуль) – каналы (структуры или параметры):

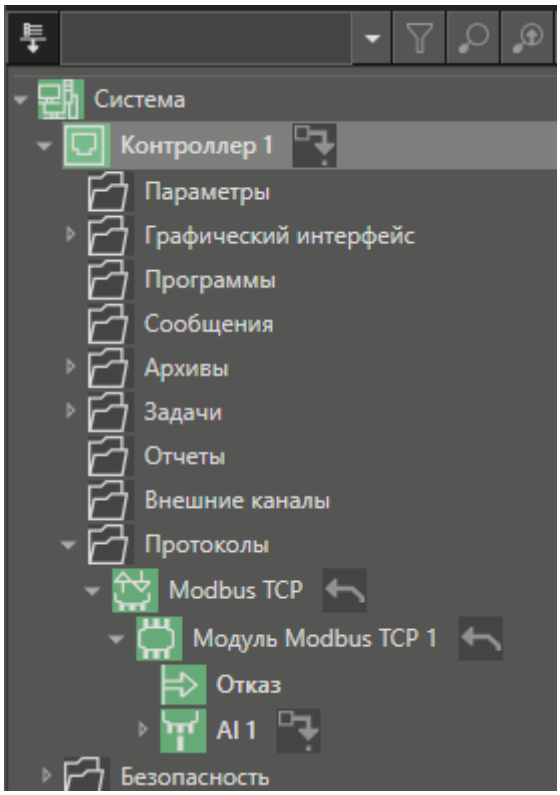


Рисунок 73

Для всех реальных аппаратных и программных элементов системы управления (АРМов, контроллеров, портов, протоколов и т.п.) в структуре проекта создаются соответствующие элементы проекта. Набор всех необходимых элементов содержится в

библиотеках, откуда элементы могут быть вставлены в соответствующую группу дерева методом перетаскивания. Для создания элементов могут быть также использованы команды контекстных меню групп дерева.

То есть в группе «Протоколы» вначале создается группа протокола (Modbus TCP в примере), при этом в окне свойств задаются параметры самого протокола (см. рисунок 74).

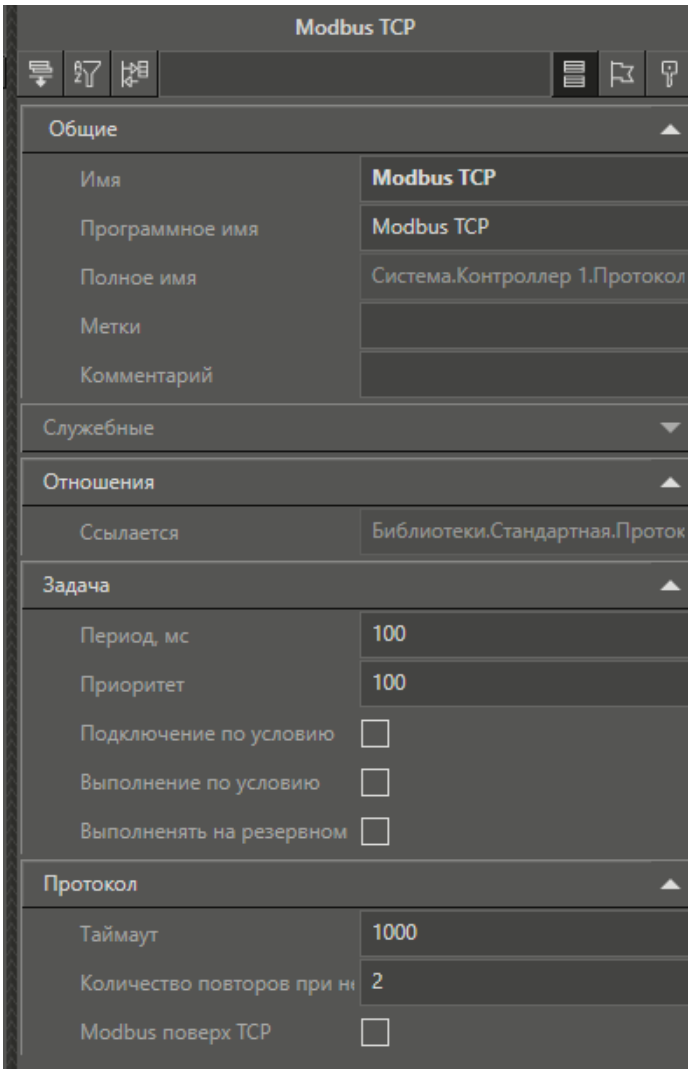


Рисунок 74

Затем в настройках модуля задаются параметры TCP/IP соединения, приведенные на рисунке 75.

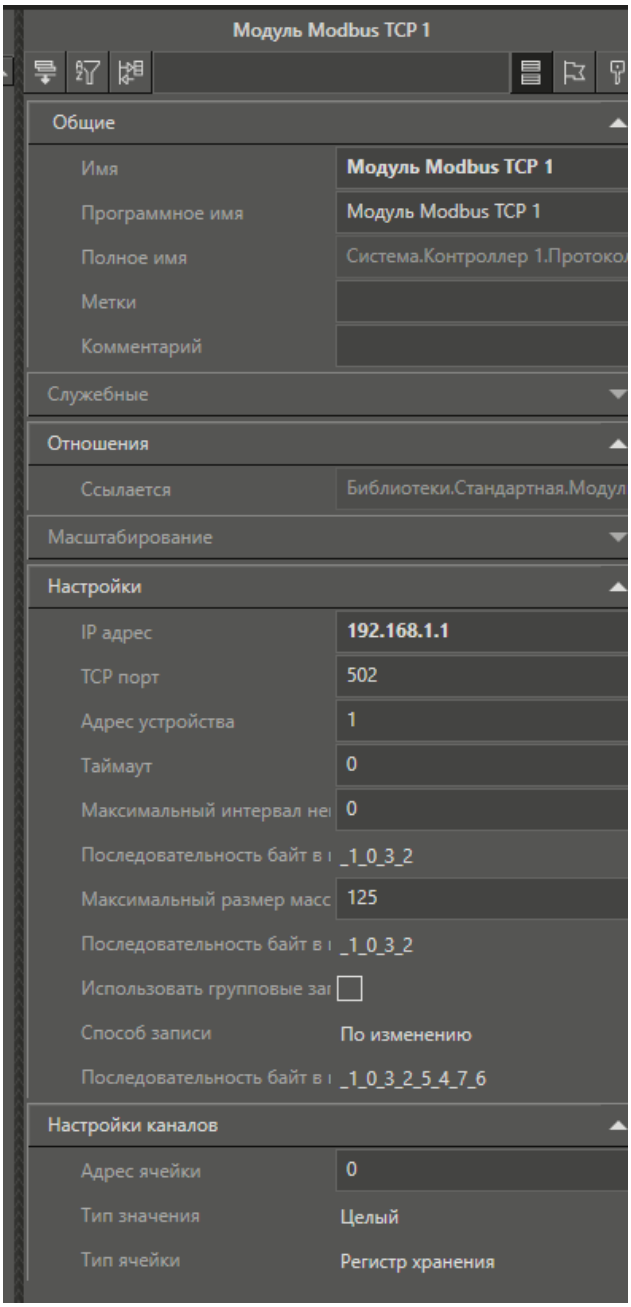


Рисунок 75

После чего следует задать адреса каналов и типы ячеек, из которых необходимо читать данные (см. рисунок 76).

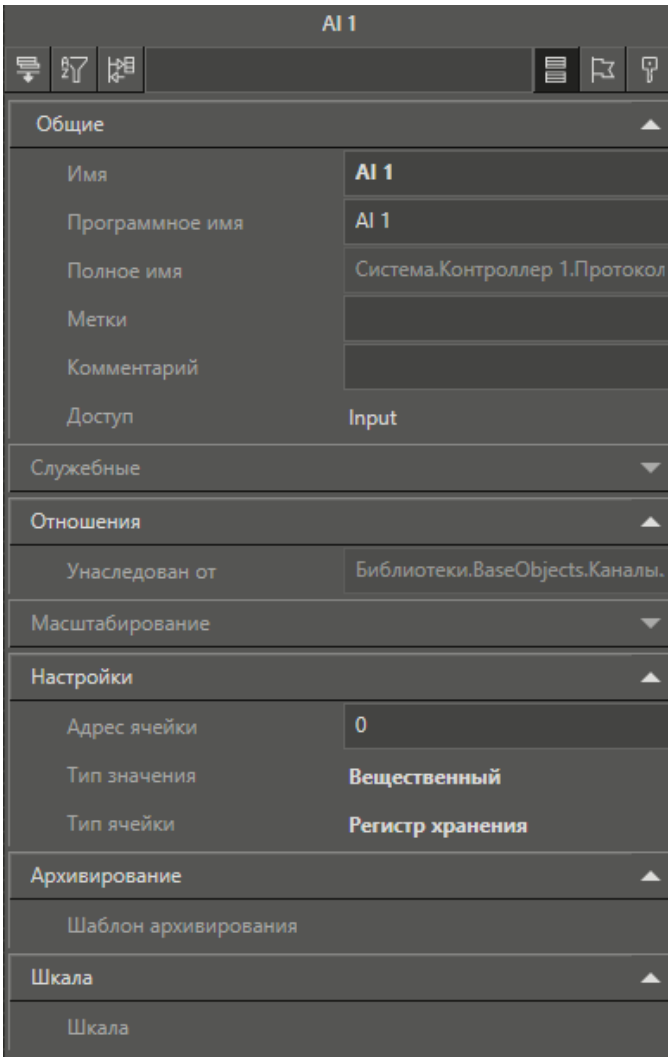


Рисунок 76

Канал – это структура, содержащая один или несколько параметров для работы с данными и ряд конфигурационных параметров. Например, канал обмена по протоколу Modbus содержит значение, прочитанное драйвером или поставленное в очередь на запись. Специфическими конфигурационными параметрами такого канала являются «Тип ячейки» и «Адрес ячейки» (соответственно адресное пространство Modbus и смещение адреса в этом пространстве), а также «Тип значения» (логический, целый и т.п.). Кроме того, для канала задаются такие общие параметры, как «Доступ» (Input – канал чтения, Output – канал записи), «Имя», «Комментарий» и др.

3.4.2.2 В группе «Объекты» задается вся логика проекта. При этом в MasterSCADA 4D основной идеологией при разработке проекта является объектно-ориентированный подход.

Объект MasterSCADA 4D представляет собой именованную совокупность графического представления технологического объекта, его параметров, алгоритмов контроля и управления, окон управления и других доступных элементов проекта (в том числе объект может содержать другие объекты). То есть объект MasterSCADA 4D может

соответствовать любому реальному технологическому объекту (датчику, исполнительному механизму, аппарату, участку, цеху) и является основной структурной единицей проекта. По сути, объект MasterSCADA 4D является мини-проектом: он содержит все параметры технологического объекта, программы и окна контроля и управления и т.п. Для связи с другими элементами проекта в объекте создаются элементы «Параметр».

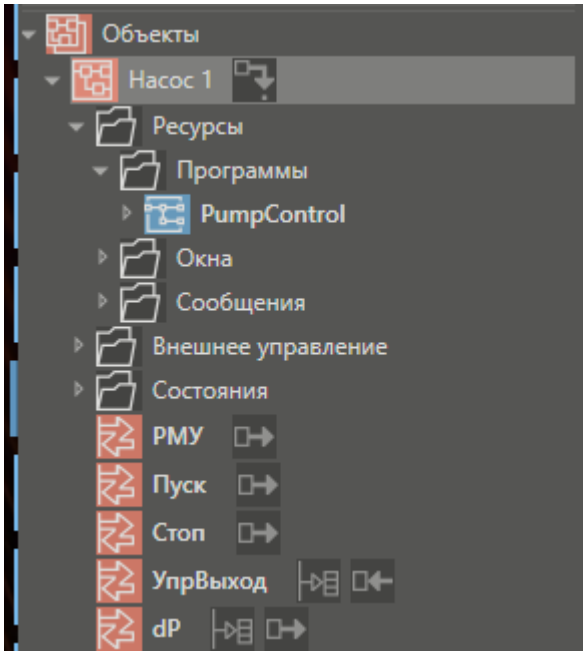


Рисунок 77

Поскольку объект может содержать другие объекты, в дереве объектов может быть создана вся иерархия технологического объекта управления – от датчика до завода в целом (см. рисунок 78).

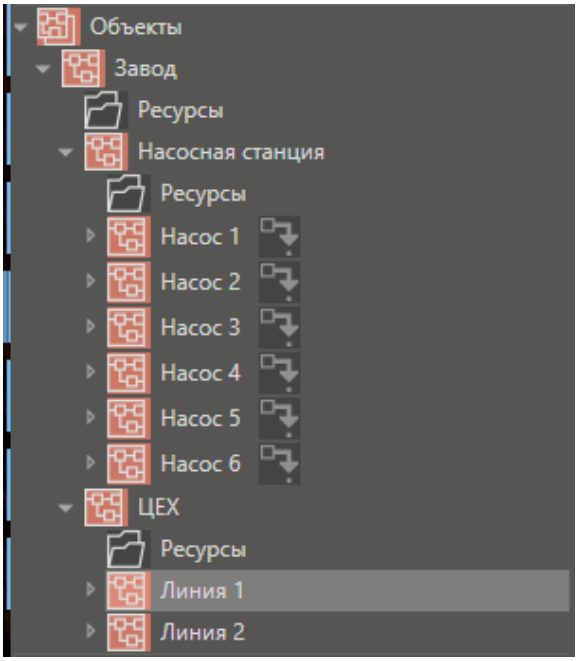


Рисунок 78

Важно! Типовые объекты разрабатываются заранее и помещаются в библиотеку, а впоследствии из них, как из кирпичиков, формируется дерево проекта.

Если узел в проекте один, то никаких действий для добавления того или иного объекта в узел не требуется, все объекты будут назначены в него автоматически (т.е. в режиме исполнения написанная программа будет исполняться в системе единственного узла). Если узлов несколько, то для того, чтобы задать исполнение объекта в том или ином узле, необходимо воспользоваться командой «Назначить в узел». Именно на этом узле и будет физически исполнятся созданный объект.

3.4.3 Программы

Программы по смыслу относятся к какому-либо объекту, однако бывают случаи, когда тот или иной элемент проекта нельзя отнести к какому-то конкретному объекту. Например, если программа относится ко всему узлу сразу и работает со всеми объектами, исполняемыми в узле. В этом случае программы создаются в подгруппе «Программы» группы узла (группы «Ресурсы» в случае объекта или канала). Пользовательские функции и типы функциональных блоков создаются соответственно в подгруппах «Функции» и «ФБ» пользовательской библиотеки (по умолчанию библиотеки «Локальная») (см. рисунок 79).

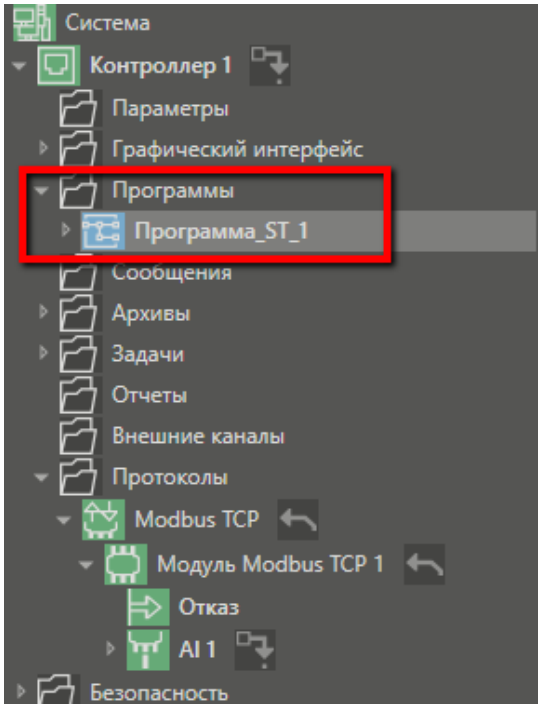


Рисунок 79

Для упрощения разработки программ в MasterSCADA 4D встроена библиотека «Стандартная», а также поставляются различные подключаемые библиотеки (например, библиотека «OSCAT»), которые содержат большое количество функций и

типов функциональных блоков. Предустановленные ФБ и функции доступны в легенде по закладкам редакторов различных языков программирования (см. рисунок 80).

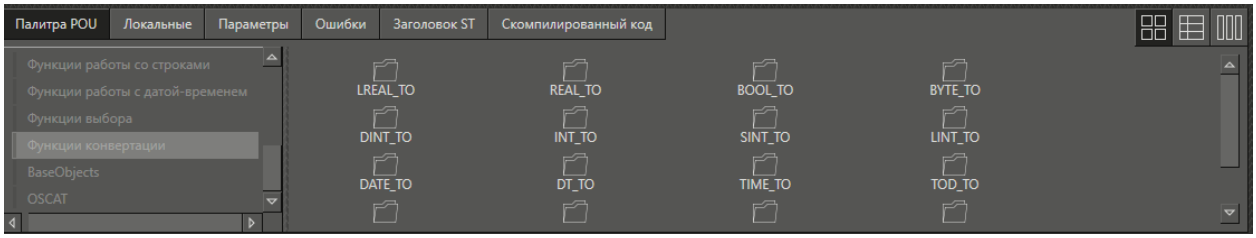


Рисунок 80

3.4.4 Параметры (переменные проекта)

Типы параметров определяются в соответствующем редакторе. Редактор открывается двойным нажатием левой кнопкой мыши на параметр в дереве (см. рисунок 76). Типы параметров можно также задать в панели свойств (см. рисунок 81).

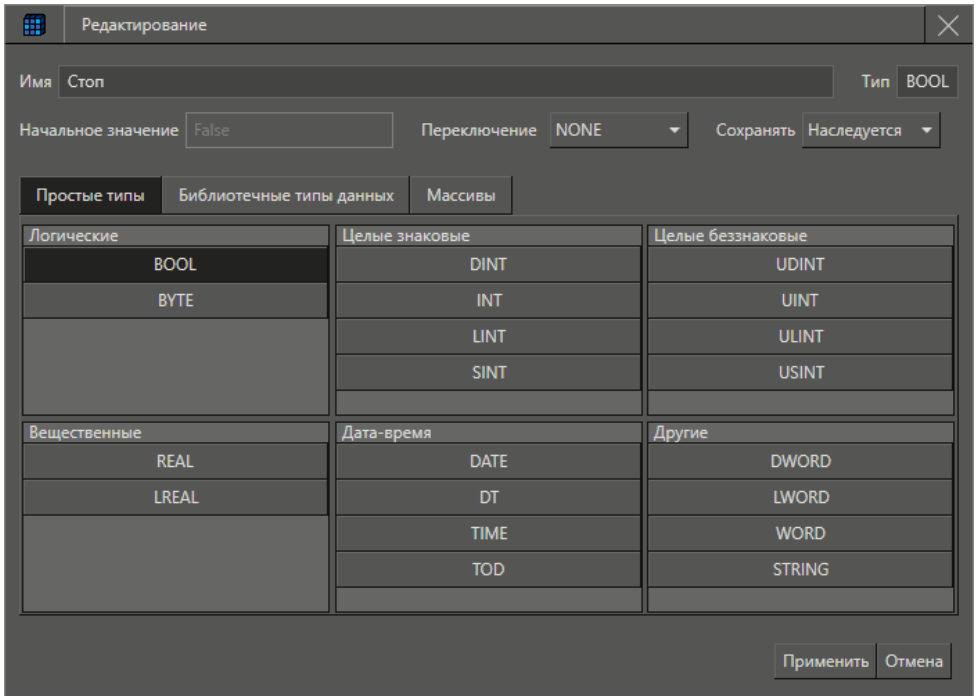


Рисунок 81

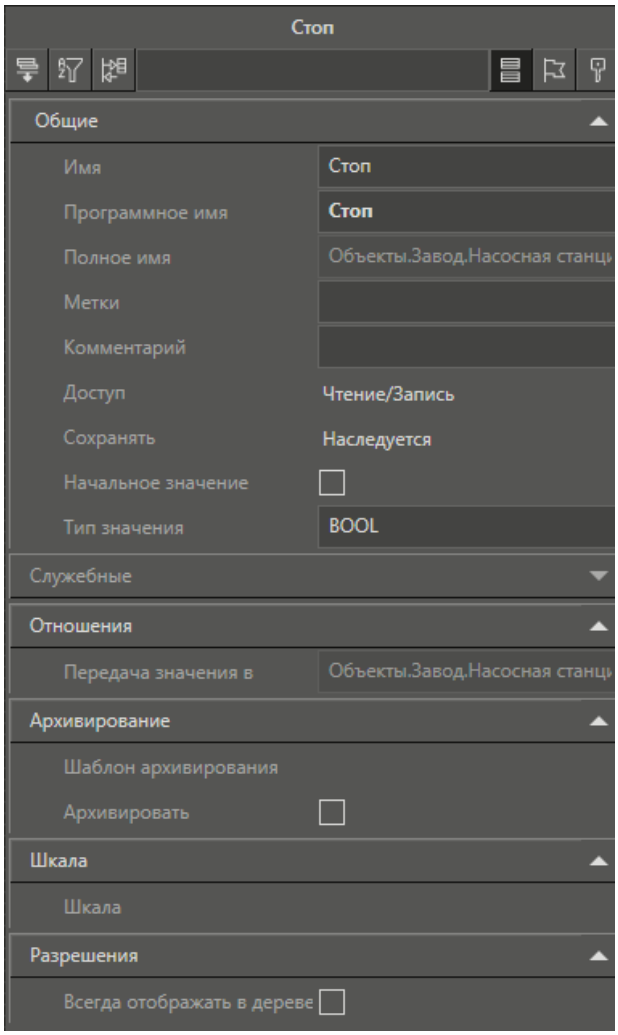


Рисунок 82

3.4.5 Окна

Окна создаются при помощи контекстного меню Объекта или Узла. При этом в центральной части интерфейса MasterSCADA 4D откроется редактор HMI (см. рисунок 83).

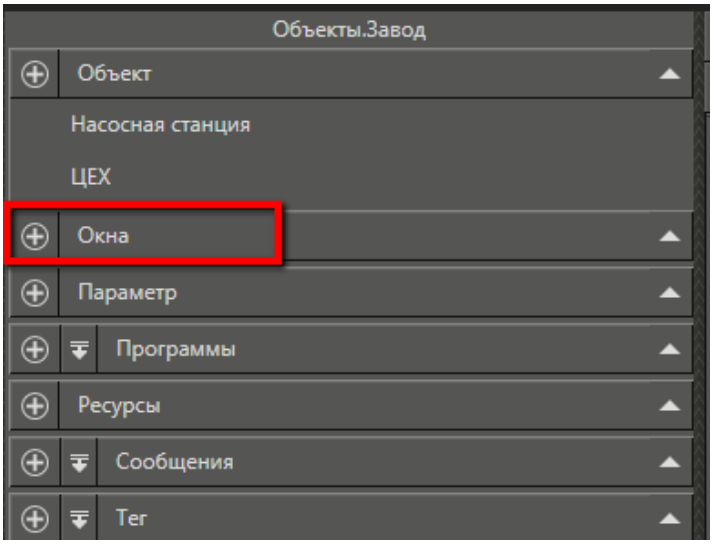


Рисунок 83

3.4.6 Редакторы

Для разработки программного элемента (программы, функции или функционального блока) в MasterSCADA 4D встроены редакторы. Для каждого языка программирования имеется свой редактор (см. рисунок 84). Создание программы производится аналогично созданию других элементов: через контекстное меню, контекстное окно или перетаскиванием из библиотеки.

Важно! Типовые программы разрабатываются в библиотеке. Для отладки в режиме исполнения программу необходимо добавить в дерево системы или дерево проекта.

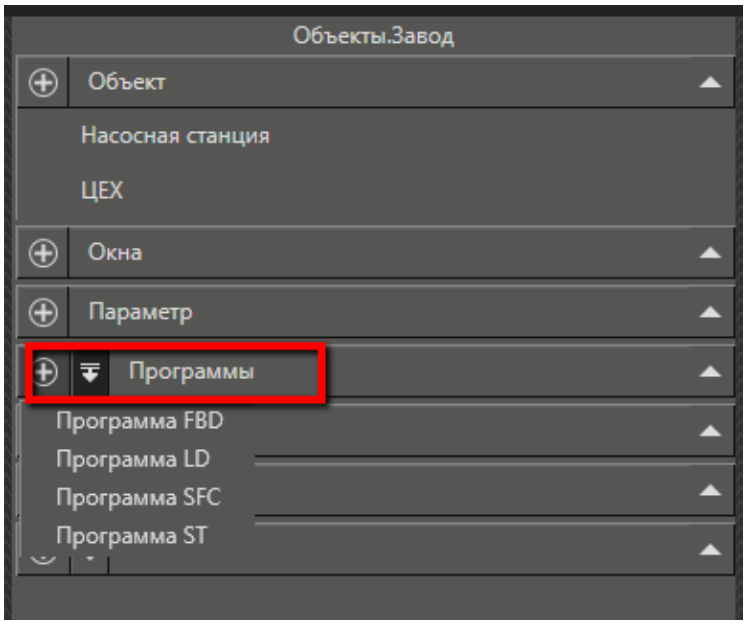


Рисунок 84

Чтобы открыть программу в необходимом редакторе, нужно дважды нажать левой кнопкой мыши на соответствующем названии программы в дереве. Редакторы открываются в виде вкладок в специальном окне. При этом легенда отображает различную информацию программного элемента, а специфические инструменты редактирования (при их наличии в редакторе) добавляются в область меню MasterSCADA 4D. На рисунке 85 показан вид MasterSCADA 4D при открытии редактора для языка ST.

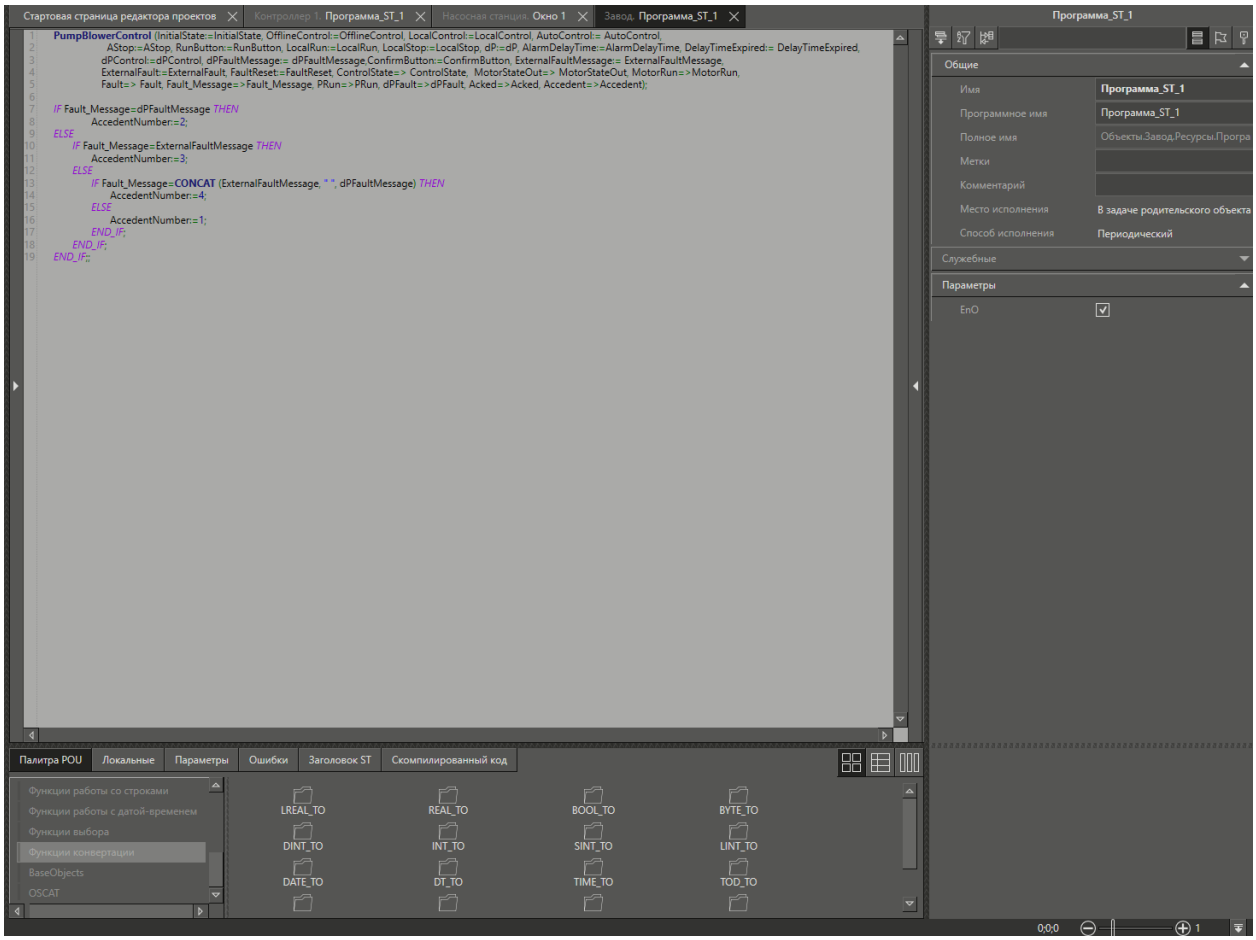


Рисунок 85

Для проверки программы на наличие ошибок следует нажать кнопку «Проверить программу» на панели инструментов → «Проверить» на вкладке «Проект» (см. рисунок 86).

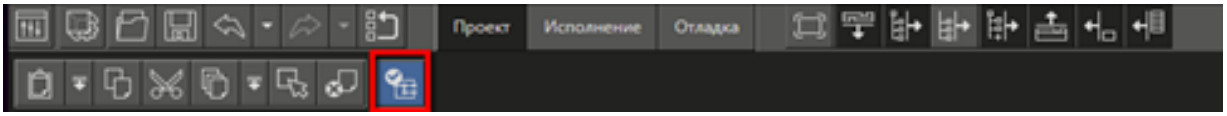


Рисунок 86

Если ошибок не обнаружено, то кнопка останется синей, а на вкладках «Заголовок ST» и «Скомпилированный код легенды» отобразятся соответственно заголовок и полный текст программного элемента на языке Master ST.

При наличии ошибок кнопка изменяет свой цвет на красный – , а информация об ошибках выводится на вкладку «Ошибки» в нижней части экрана (см. рисунок 87).

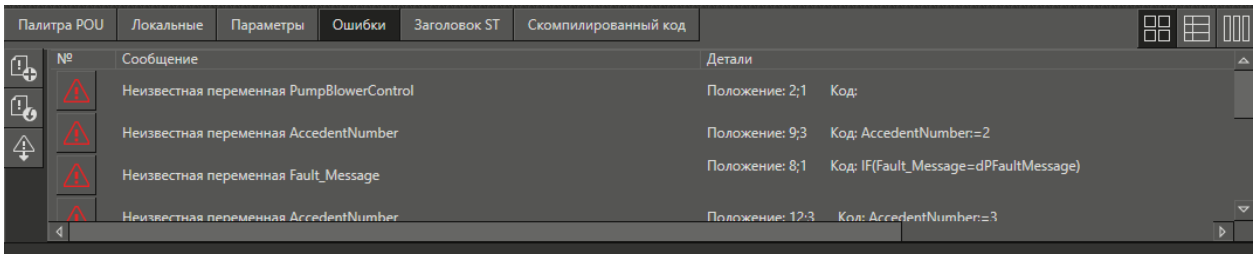


Рисунок 87

3.4.7 Конфигурирование связей

Связи в пределах одного узла (например, Параметр канала – Параметр программы) создаются с помощью перетаскивания элемента на элемент (см. рисунок 88).

Важно! Если связываемые параметры могут как получать, так и отправлять значения, то перетаскивать следует источник данных на приемник данных.

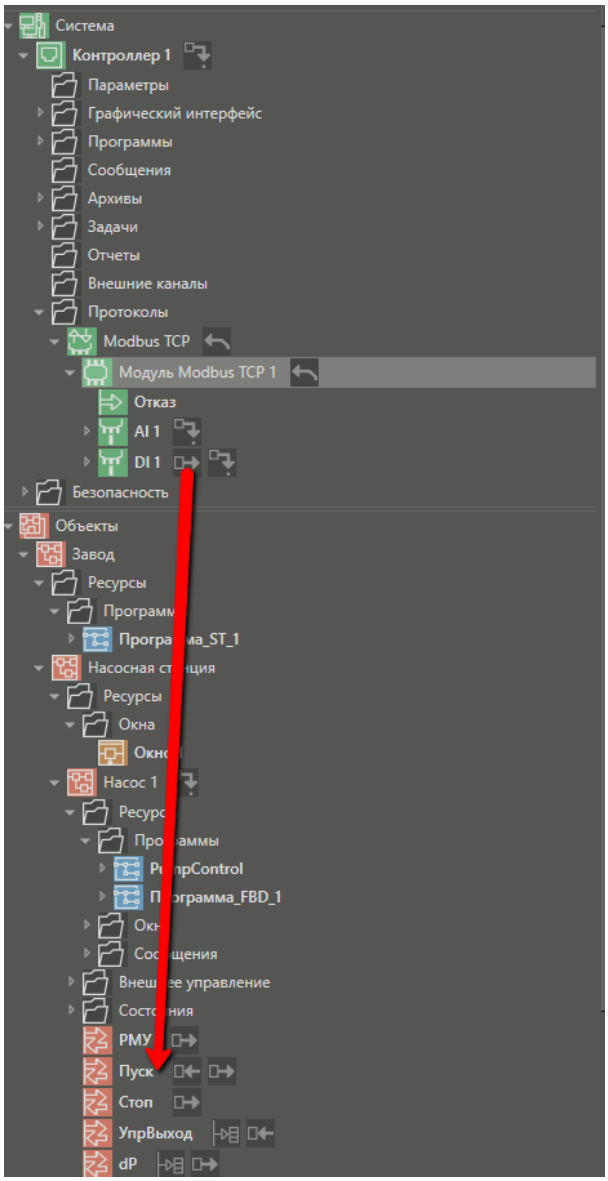


Рисунок 88

Если установка связи невозможна (например, при перетаскивании выходного сигнала на выходной), то отображается специальный индикатор:

Для удаления связи используется команда «Разорвать» контекстного меню элемента (см. рисунок 89).

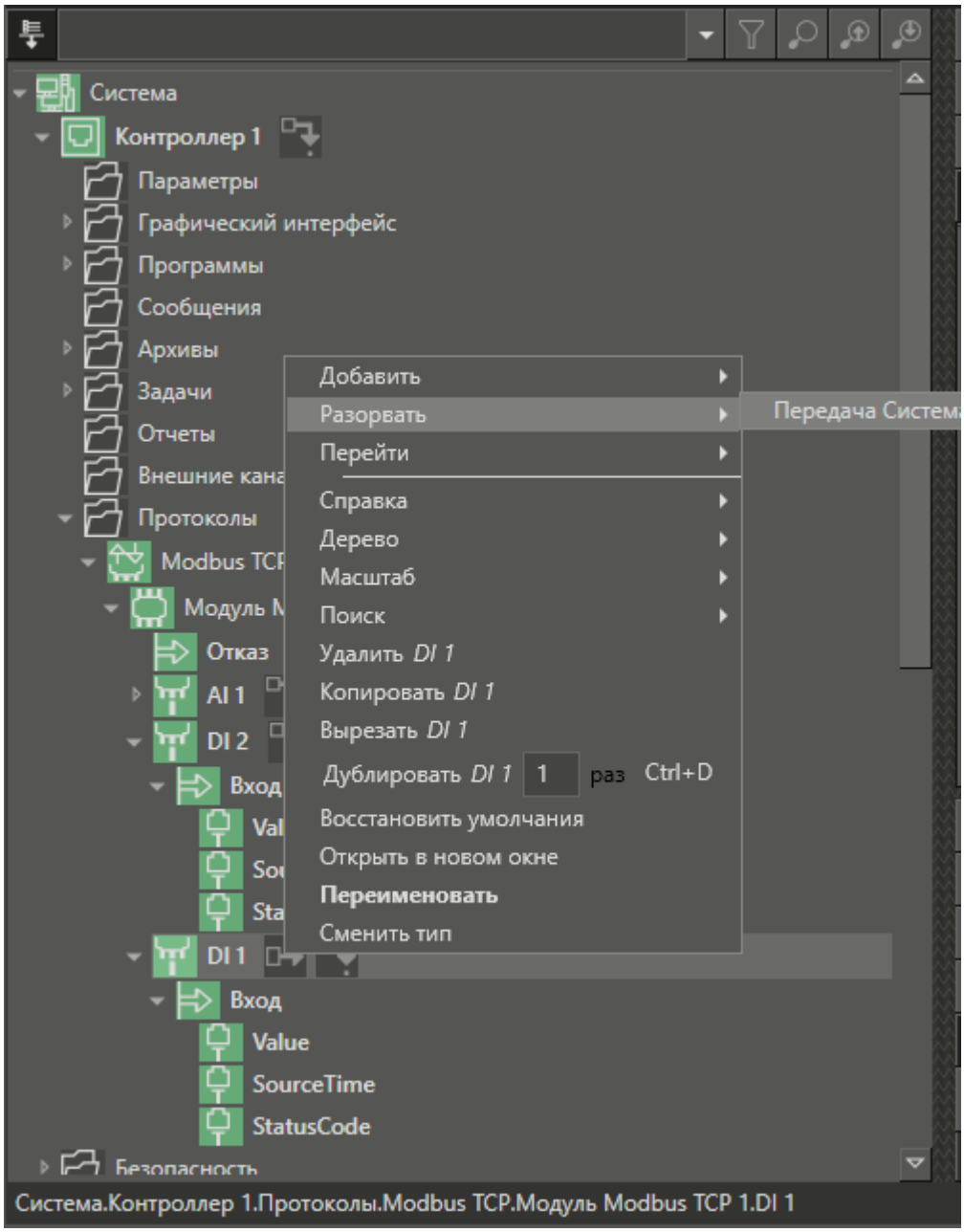


Рисунок 89

3.4.8 Формирование задач узлов

По умолчанию, все объекты исполняются в «Основной задаче» узла. Период обработки программы зависит от настроек задачи.

В случае если необходимо, чтобы часть объектов или программ исполнялись

независимо от основной задачи, то создаются дополнительные задачи в узле в соответствующей группе. Затем тот или иной элемент проекта при помощи контекстного меню назначается в ту или иную задачу (см. рисунок 90).

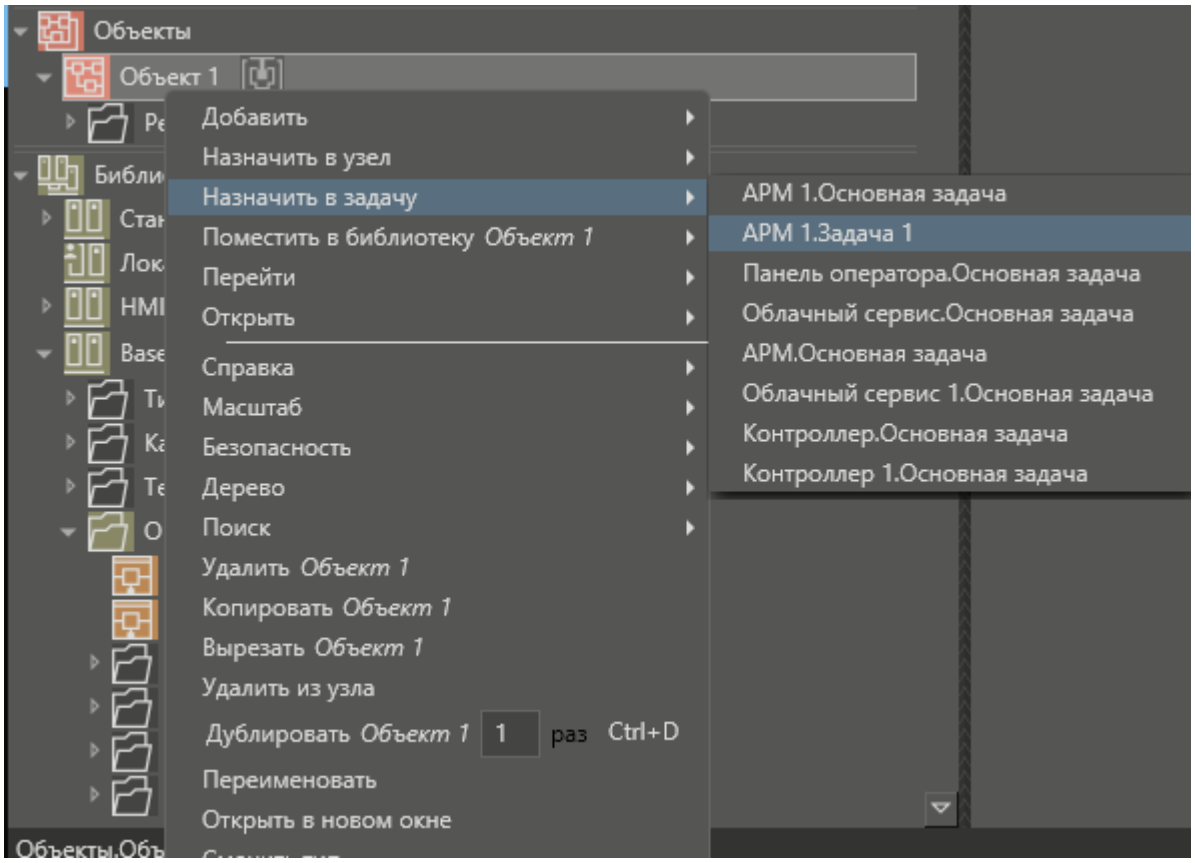


Рисунок 90

3.4.9 Автономная отладка в реальном времени

В состав MasterSCADA 4D входит эмулятор, с помощью которого можно проверить работу конфигурации узла в реальном времени автономно (без загрузки непосредственно в устройство).

Для запуска эмуляции узла необходимо либо в контекстном меню выполнить пункт Узел → Подключить узел в режиме эмуляции, либо нажать на кнопку «Эмуляция» во вкладке инструментов «Исполнение» (см. рисунок 91).

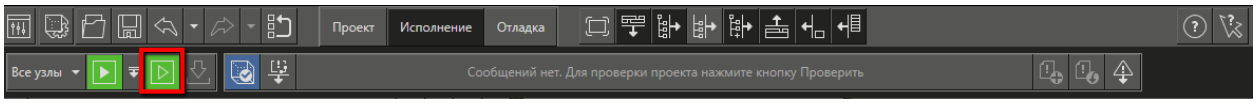


Рисунок 91

Важно! В режиме эмуляции опрос подключенных к узлу по различным протоколам устройств и встроенных модулей не происходит.

Если в проект включены несколько узлов, то запускаются несколько исполнительных систем одновременно.

В режиме эмуляции значения параметров узла отображаются и могут быть вручную изменены в различных местах, например, в дереве (см. рисунок 92).

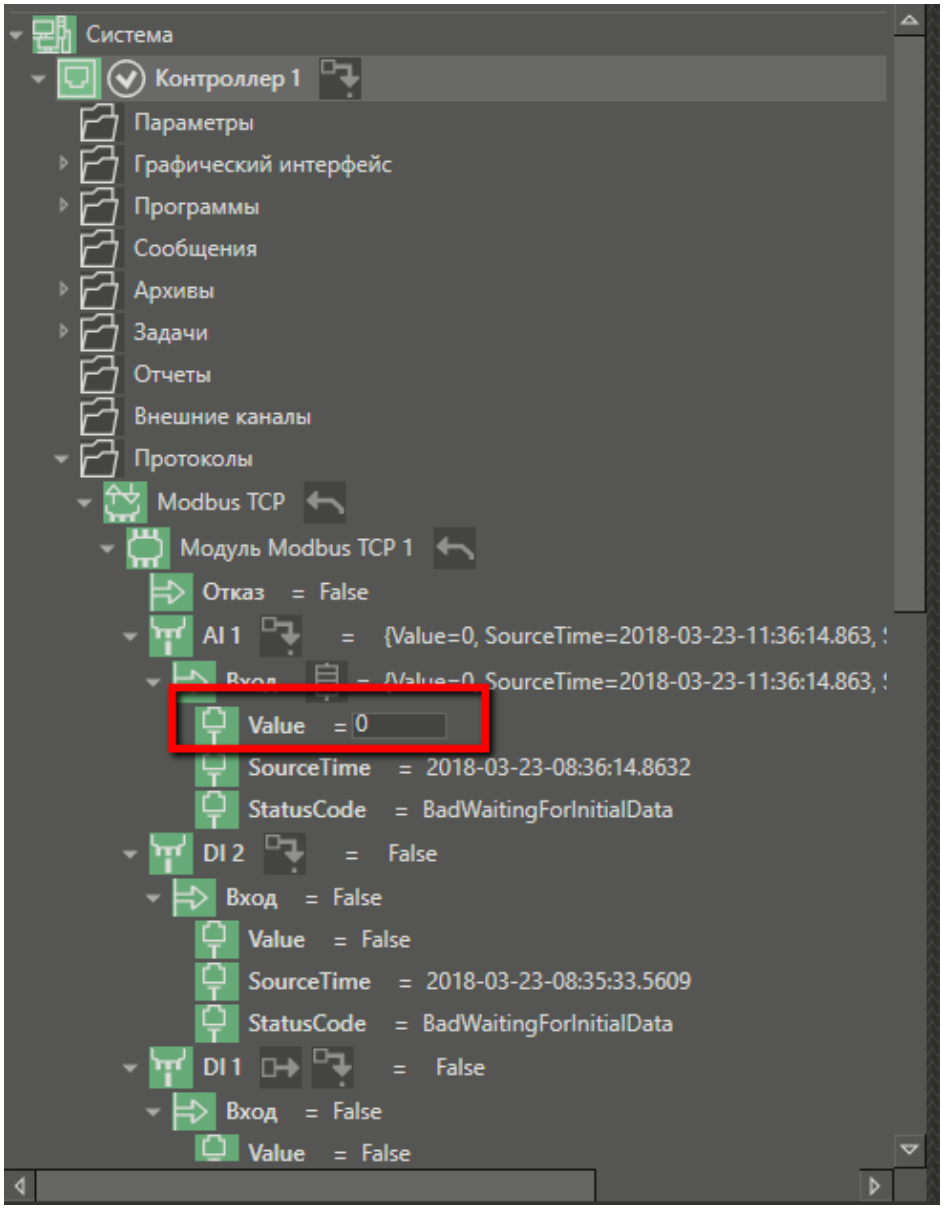


Рисунок 92

Если конфигурация узла содержит окно, то помимо эмулятора запускается веб-сервер, и окно открывается в клиенте визуализации.

Для выхода из режима эмуляции необходимо нажать кнопку «Отключить» панели инструментов вкладки «Исполнение».

Если окна содержатся в нескольких узлах, то откроется несколько клиентов визуализации одновременно.

3.4.10 Правила разработки проектов

Приведен список пунктов, которые помогут оценить качество проекта:

1 Если в проекте встречаются одинаковые программы, окна, объекты, шаблоны архивирования, шкалы, модули ввода-вывода и т.п. более одного раза, то в дерево библиотек добавлены типы, а в дереве проекта используются их экземпляры, где это возможно, или наследники, где экземпляры создать нельзя.

2 Каналы протоколов или модулей ввода-вывода имеют связи только с параметрами объектов. Если канал имеет прямую связь с графическим элементом окна объекта, то проект сделан не оптимально с точки зрения. Канал может иметь связь с графическим элементом окна, если окно расположено в протоколе или модуле ввода-вывода, которому принадлежит канал.

3 Каналы имеют минимальное количество связей с параметрами объекта: одна связь для каналов с доступом Input, Output; две связи для каналов InOut.

4 Типы параметров объектов должны соответствовать типам параметров каналов, с которыми они связаны. За исключением случаев, когда польза при конвертации выше потери данных о качестве канала и метке времени, например, канал с System_dint_param связан с параметром типа Перечисление.

5 Архивы формируются по изменению.

6 Для архивируемых параметров указана мертвая зона архивирования.

7 На тренды одновременно выводится минимально возможное для проекта количество перьев. Не более 5-7 в один момент времени в одном окне.

8 Минимизировано число архивируемых параметров и параметров, для которых свойство «Сохранять» = Да.

9 Если программа выполняется по вызову из окна, и значения параметров программы определяются в окне, то их нужно задавать, через параметры действия (параметры действия связываются с соответствующими свойствами графических элементов, а не параметры программы в дереве с графическими элементами напрямую). Если параметры программы определяются, например, в других программах, то связи могут быть заданы явно от одной программы в другую, или из параметра в дереве проекта в программу или обращаться через прямой доступ к параметрам проекта.

10 Реальный средний период выполнения задач не превышает заданный, и не увеличивается со временем.

11 Если проект с резервированием, то разработчик понимает, какие параметры синхронизируются между узлами (свойство «Сохранять» = Да). Это может быть особенно актуально, если опрос нижнего уровня ведет только узел, который является основным.

12 Период основной задачи узла (цикл работы программы в исполнительной системе) выбран оптимально для большинства объектов.

13 Объекты, для которых реально нужна высокая скорость обработки назначены в

дополнительную задачу узла.

14 Объекты, которые можно выполнять реже, чем другие назначены в дополнительную задачу узла.

15 Имена параметров, объектов и других компонентов имеют уникальные названия, которые понятны коллегам и службе эксплуатации объектов, заданы комментарии.

16 Названия элементов не начинаются с цифры и не содержат спецсимволов.

3.5 Настройка резервирования контроллеров и резервирования протоколов

3.5.1 Описание алгоритма резервирования

3.5.1.1 Общие положения

Два устройства с загруженной в них исполнительной системой MasterSCADA 4D специальной конфигурации работают одновременно: одно имеет статус Основной (MASTER), другое - Резервный (SLAVE). Клиент должен подключаться к тому узлу, который находится в режиме основного. Если связь с этим узлом будет потеряна, то страница клиента автоматически переподключится к тому узлу, который находится в состоянии основного. Программы на обоих узлах выполняются независимо. Опрос устройств могут вести два узла одновременно. Однако для протоколов, в которых параллельный опрос устройств невозможен, получение текущих данных по протоколам, добавленным в проект, производит только один из них - тот, который работает в режиме основного. В процессе работы, параметры, у которых в настройке «Сохранять» установлено значение «Да», передаются от основного узла резервному. Период передачи данных определяется в настройках службы резервирования. В общем случае, при возникновении отказа основного узла управление переходит к резервному. Переключение клиентов визуализации к тому узлу, который находится в состоянии MASTER происходит без перезагрузки страницы клиента визуализации. Раз в 10 секунд происходит синхронизация сессий пользователей между MASTER и SLAVE.

3.5.1.2 Причины возникновения отказа

3.5.1.2.1 В результате отказа протокола добавленного в проект

В зависимости от настройки протокола «Формировать отказ узла» также может сформироваться отказ узла. Если оба узла выставят отказ, то в режиме основного останется последний узел, получивший этот статус.

3.5.1.2.2 Программный отказ

Устанавливается через функциональный блок RedundancyControl (на вход SetProgramFault подается значение True) или кнопкой «Отказ» на панели управления узлом.

3.5.1.3 Правила переключения

Переключение MASTER - SLAVE происходит в следующих случаях:

1 Если текущий узел SLAVE и при этом другой узел не отвечает или тоже находится в режиме SLAVE, то осуществляется переключение текущего узла в MASTER;

2 Если текущий узел MASTER, и у него возникает отказ, а на другом узле нет отказа, то осуществляется переключение текущего узла в SLAVE, а другой узел переключается в MASTER по правилу 1;

3 Если текущий узел MASTER и другой узел тоже MASTER, то осуществляется переключение текущего узла в SLAVE.

Оба узла равноправны, однако на узле 2 имеется пауза при старте 10 сек, для того чтобы при одновременном включении узел 1 загрузился раньше и стал MASTER. Но в дальнейшем переключение идет по общим правилам.

3.5.2 Настройка резервирования

3.5.2.1 Сначала необходимо добавить службу «Резервирование». Для этого в контекстном меню группы «Службы» нужно выбрать соответствующий пункт (см. рисунок 93).

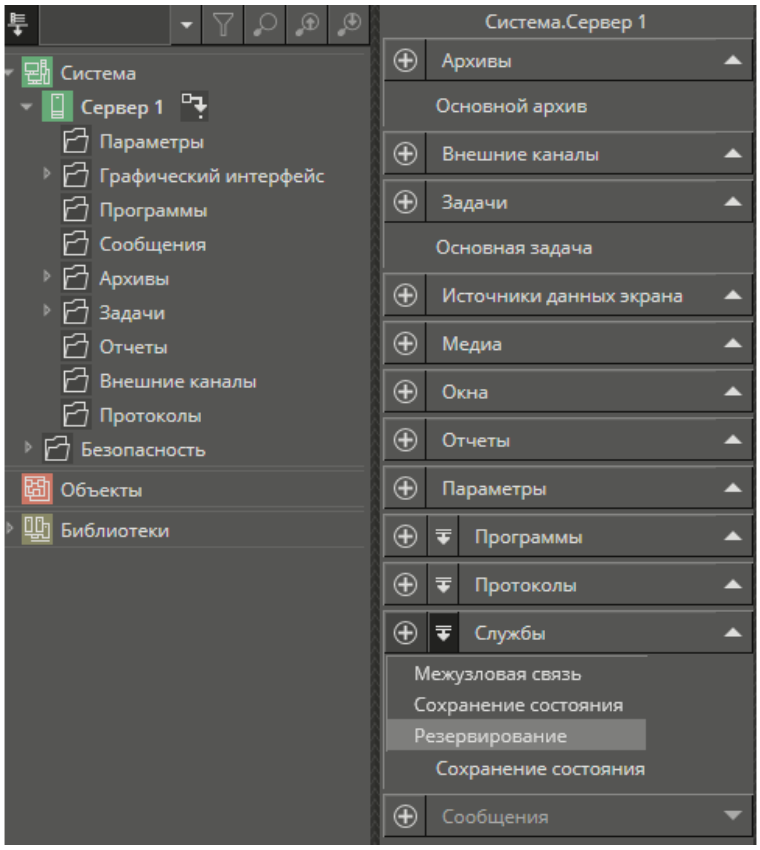


Рисунок 93

3.5.2.2 После выполнения предыдущего пункта, в панели свойств узла появятся дополнительные настройки, позволяющие определить в сети второй сервер из резервируемой пары (см. рисунок 94). Описание свойств узла группы «Резервирование»

приведены в таблице 15.

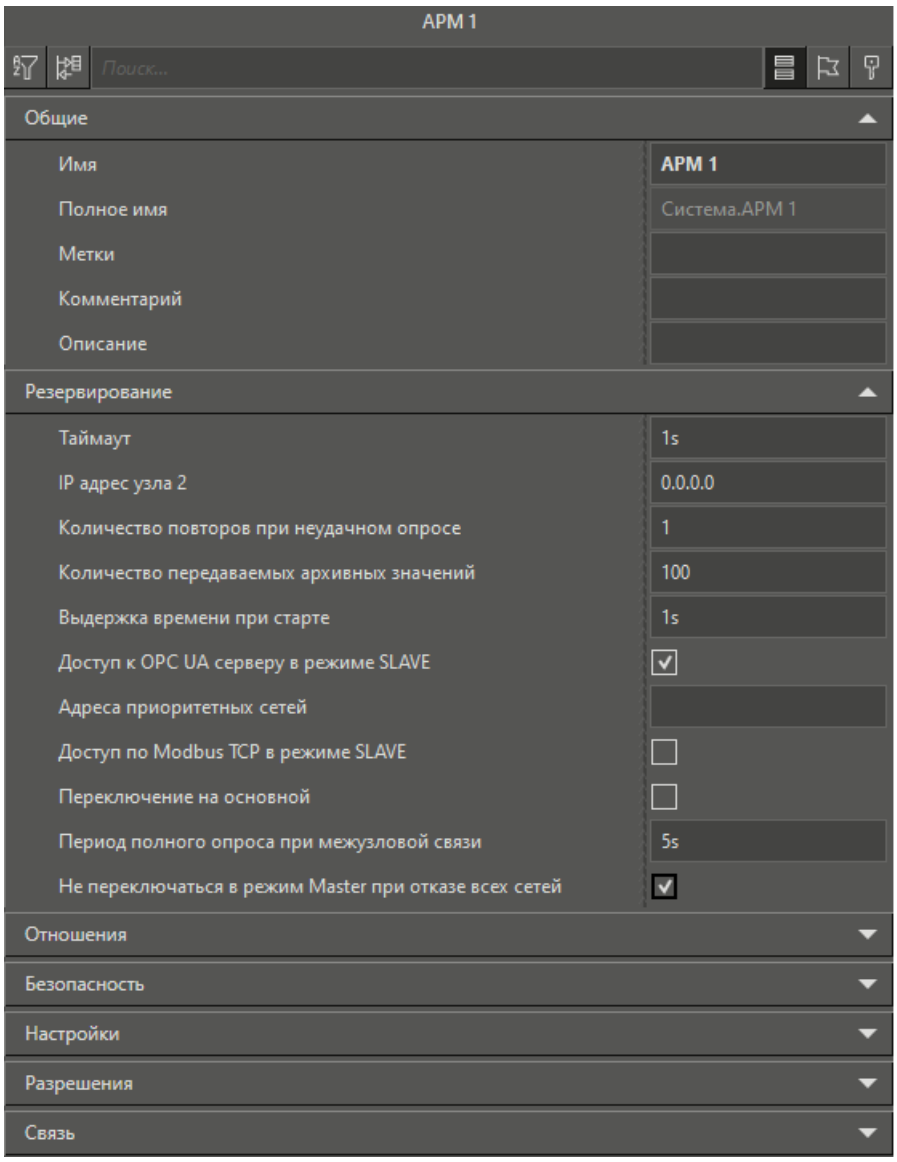


Рисунок 94

Таблица 15

Свойство	Назначение
IP-адрес узла 2	Задается IP-адрес устройства (компьютера, контроллера и т. п.), которое выполняет роль второго сервера. Если устройство имеет две сетевые карты, то задаются два IP-адреса через запятую. IP-адрес узла 1 при резервировании задается в группе настроек Связь в поле IP адрес.
Таймаут	Таймаут опроса состояния компьютеров.
Количество повторов при неудачном опросе	Определяется количество неуспешных попыток связи основного и резервного серверов, выполненных подряд, прежде чем будет сформирована ошибка связи.
Количество передаваемых архивных значений	Указывается максимальное количество значений одного параметра, передаваемых в одном запросе архива при резервировании. По умолчанию установлено значение 100

Свойство	Назначение
Выдержка времени на старте	Определяет через какое время (N) после загрузки конфигурации будет возможен переход с основного на резервный для обновления основного Рассмотрим порядок работы: 1) Обновляется конфигурация в резервном сервере (SLAVE). 2) Сервер перезагружается 3) В течении времени N после перезагрузки резервного принудительно остается в режиме SLAVE (основной видит этот статус и не пытается передать управление при своем отказе, среда разработки считает, что загрузка все еще идет в него) 4) Снимается запрет резервного на переход в MASTER. Среда получает статус успешной загрузки и начинает загрузку файлов в него 5) После загрузки файлов в основной он передает управление на резервный и на N секунд ставится признак запрета на переход в Master. 6) По истечении задержки основной перегружается. 7) основной загрузился. Аналогично п.3 в течении N секунд действует запрет на перевод основного в Master 8) По истечении этого времени снимается запрет и среда разработки выдает статус о загрузке конфигурации в оба контроллера. Если нет отказов, то остается сервер, что был в начале алгоритма основным - SLAVE, а тот, что был резервный - MASTER.
Доступ к OPC UA серверу в режиме SLAVE	Определяет возможность подключения по протоколу OPC UA к серверу, который находится в состоянии резервного. При получении данных сторонним клиентом OPC UA нужно учитывать следующую информацию: • Если флаг установлен и сервер находится в состоянии резервного, то устанавливаются параметры: Server.ServiceLevel = 1 Server.ServerStatus.State = Suspended (3) • В режиме основного (при любом значении настройки Доступ к OPC UA серверу в режиме SLAVE, также без резервирования): Server.ServiceLevel = 255 Server.ServerStatus.State = Running (0)
Адреса приоритетных сетей	Задаются адреса сети вида 192.168.56.0/24 через запятую. Среди адресов другого узла в первую очередь используются адреса, принадлежащие указанным в порядке следования. Если ни один не отвечает, используются оставшиеся адреса, но раз в минуту делается попытка обратится по приоритетным адресам.
Доступ по Modbus TCP в режиме SLAVE	Определяет возможность подключения к резервному серверу по протоколу Modbus TCP. По умолчанию флаг отключен. При установке флага, в режиме SLAVE, будет продолжать работать Modbus TCP-сервер.
Переключение на основной	По умолчанию значение настройки FALSE. При значении TRUE всегда происходит переключение на основной узел, если на нем нет Отказа. То есть если на основном узле возникает отказ и резервный узел переходит в состояние MASTER, то без установки данной настройки после снятия отказа на основном узле резервный так и останется в роли MASTER. При установленной настройке мастерство переключится на основной узел после снятия на нем отказа.

Свойство	Назначение
Период полного опроса при межузловой связи	Задается период, с которым будет происходить отправка всех данных, без учета изменений, при межузловой связи.
Не переключаться в режим Master при отказе всех сетей	Данная настройка используется для предотвращения автоматического переключение узла в режим Master в случае отказа всех сетей. При значении TRUE, если все сети для резервирования недоступны, то узел переключится в режиме Slave. По умолчанию значение этой настройки FALSE.

Важно! IP-адреса основного и резервного узла не могут совпадать. Также служба резервирования не работает на узле с локальным IP-адресом (127.0.0.1 или localhost). В обоих случаях при компиляции проекта будет выдаваться ошибка (см. рисунок 95)

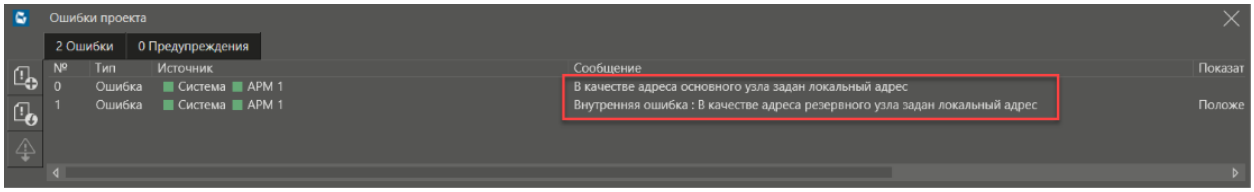


Рисунок 95

3.5.2.3 Расширенный доступ к настройкам, отвечающим за резервирование, можно получить в панели свойств. Для этого в дереве системы следует выделить службу «Резервирование» (см. рисунок 96). Описание основного свойства приведено в таблице 16.

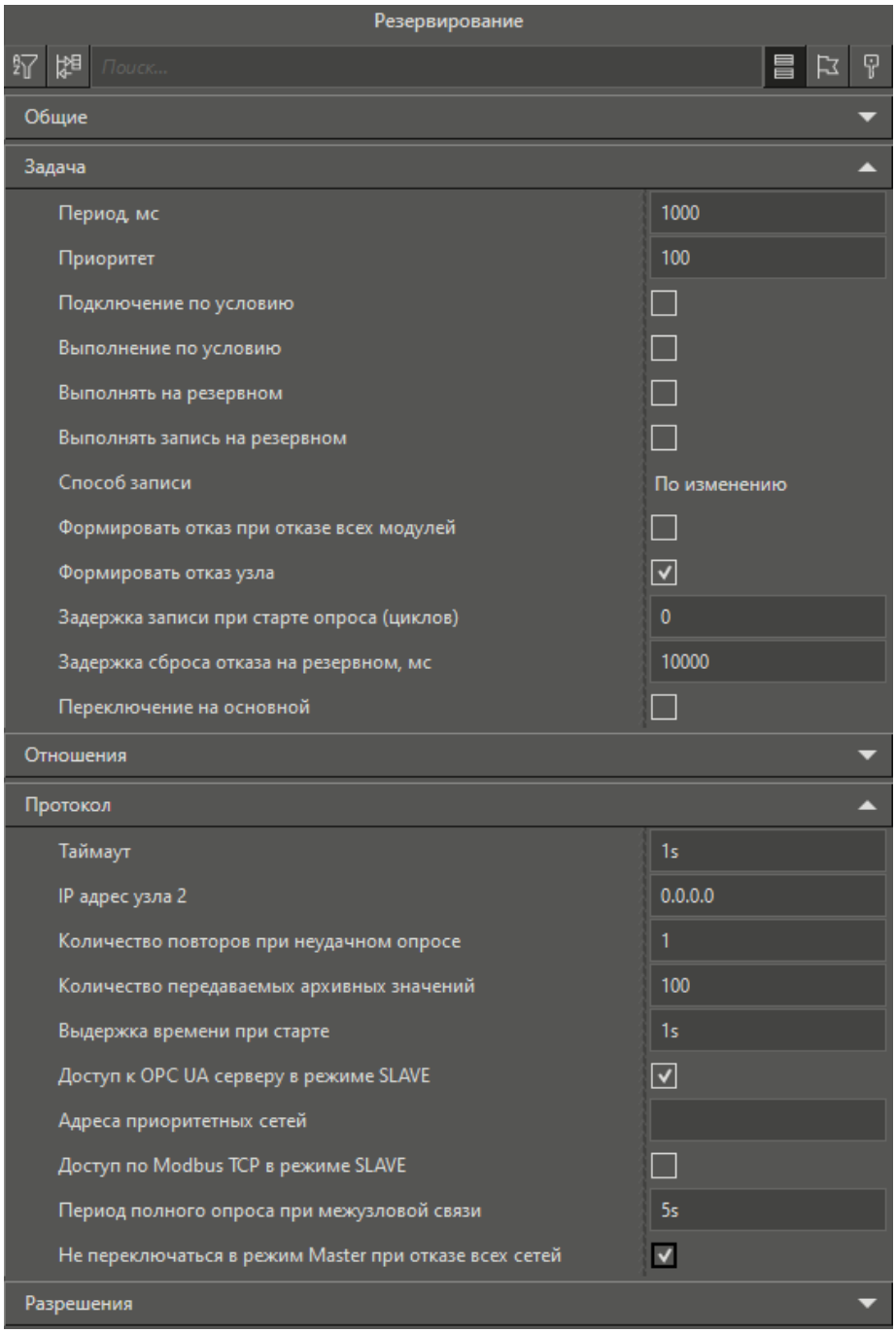


Рисунок 96

Таблица 16

Свойство	Назначение
Категория **Задача**	
Переключение на основной	При значении TRUE всегда происходит переключение на основной узел, если на нем нет Отказа. По умолчанию значение настройки FALSE

3.5.3 Порядок синхронизации данных между основным и резервным

3.5.3.1 Синхронизация текущих данных

Между основным и резервным передаются только те текущие значения, для которых свойство «Сохранить» имеет значение «Да» или «Наследуется», но во втором случае у одного из родительских элементов должно быть установлено Сохранять=Да. Поддерживается синхронизация значений параметров различных типов данных, в том числе длинных строк (более 256 символов), при условии, что параметр имеет тип STRING, поддерживается синхронизация динамических массивов (массивов переменной длины).

Кроме значения передается и признак блокировки. В независимости от настройки свойства «Сохранить», заблокированный параметр Объекта, назначенного в узел со статусом MASTER, всегда будет сохранять значение даже при изменении статуса узла на SLAVE.

Важно! На узле со статусом SLAVE невозможно изменять, блокировать или разблокировать параметры. При любой попытке изменить параметр на узле со статусом SLAVE его значение или состояние будет сбрасываться к текущему.

Период, с которым основной будет передавать информацию резервному, определяется одноименной настройкой, которая находится в категории «Задача» службы Резервирование.

Данные передаются по внутреннему протоколу. Каждый сервер имеет свой файл, необходимый для восстановления после рестарта. После получения необходимых данных от основного резервный записывает и в свой файл session.bin. Данные записываются в файл согласно периоду выполнения службы «Сохранение состояния», а в задаче они будут использованы в следующем цикле выполнения задачи.

3.5.3.2 Синхронизация архивных данных

Узел, работающий в режиме MASTER, сохраняет все данные в выбранную базу данных. Узел SLAVE делает запрос на передачу данных 1 раз секунду. Архивные данные передаются от основного сервера к резервному по средствам протокола TCP. За один раз по каждому параметру может быть передано значений не больше, чем указано в свойстве «Количество передаваемых архивных значений».

Если основной или резервный узел был отключен, то после включения произойдет синхронизация пропущенных участков архивов данных, и узел получит все сообщения, которые возникали с момента последней записи в базу данных.

3.5.3.3 Синхронизация сообщений

Узел, работающий в режиме MASTER, сохраняет все сообщения в выбранную базу данных. Узел SLAVE делает запрос на сообщения 1 раз в секунду. За 1 секунду может быть

передано не более 1000 сообщений. Если резервный сервер был отключен, то после включения он получит все сообщения, которые возникали с момента последней записи в базу данных.

При отказе основного узла он переходит в режим SLAVE, а резервный узел переходит в режим MASTER и сохраняет сообщения в свою БД.

Важно! Если переход узла из режима SLAVE в режим MASTER происходит быстрее, чем синхронизация сообщений от MASTER к SLAVE, то сообщения будут потеряны.

3.5.3.4 Синхронизация настроек безопасности при резервировании

Если вы не используете внешнюю БД для хранения настроек безопасности, то файлы JSON будут автоматически синхронизироваться между основным и резервным серверами. Это означает, что настройки безопасности будут обновляться на обоих серверах, чтобы гарантировать их актуальность.

В случае использования сетевой базы данных для хранения настроек безопасности, любые изменения, внесенные на одном сервере, будут автоматически доступны и на другом сервере. Таким образом, оба сервера всегда будут иметь актуальные настройки безопасности.

3.5.3.5 Обработка запросов при обработке резервирования

UDP запросы следующих типов:

- запрос резервного сервера на получение архивных данных;
- запрос резервного сервера на получение архивных/новых сообщений.

Обрабатываются в отдельном потоке. Данные типы запросов в статистике отображаются как N2.

3.5.3.6 Настройки резервирования протоколов

Контроллер поддерживает резервирование в протоколах. Включение и настройка происходит в свойствах протокола, например, Modbus TCP (см. рисунок 97). Описание свойств резервирования протокола Modbus TCP приведено в таблице 17.

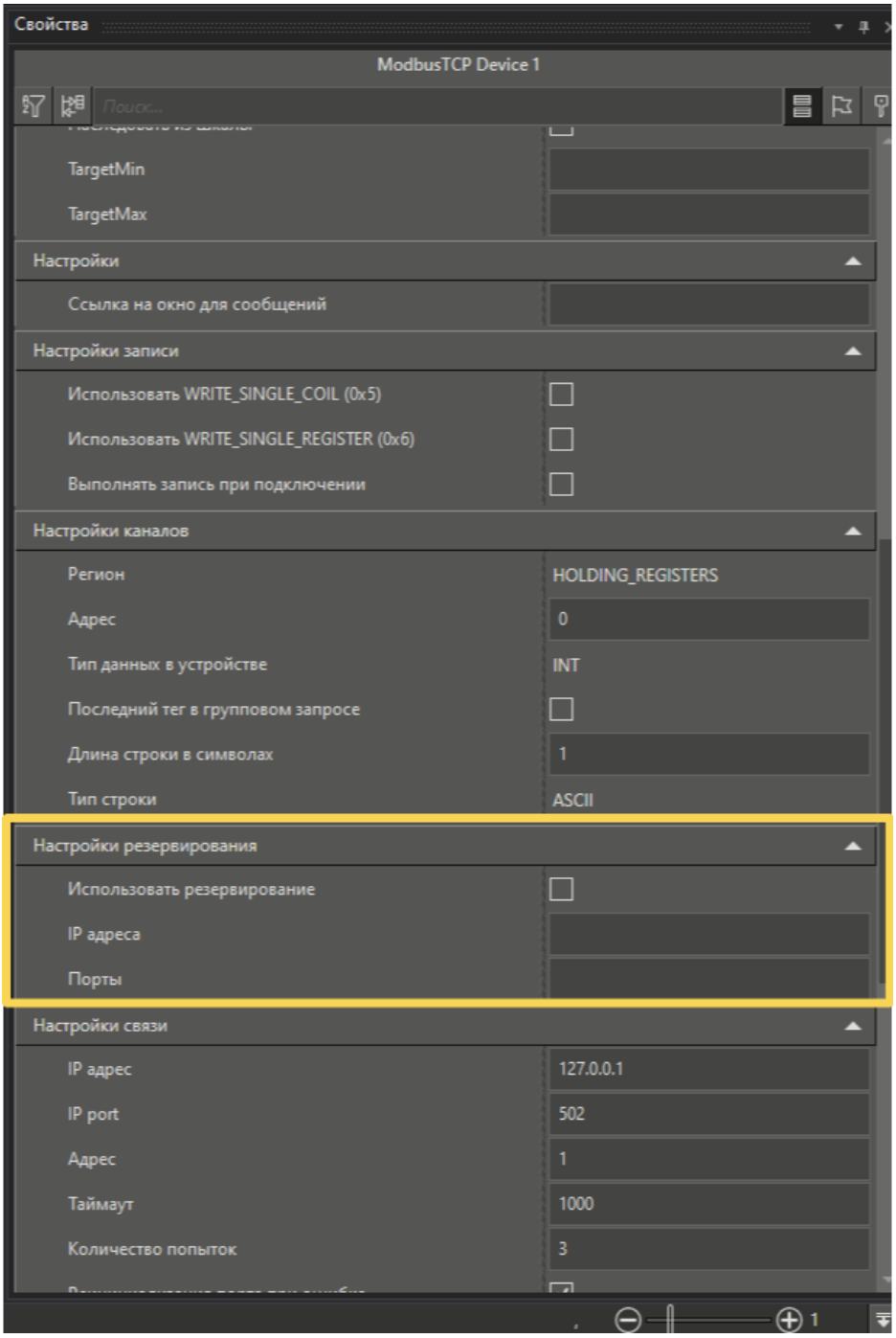


Рисунок 97

Таблица 17

Категория **Настройки резервирования**	
Использовать резервирование	При значении TRUE включается режим резервирования каналов связи. В штатном режиме опрос производится по основному каналу, заданному в поле IP адрес. Если возникает отказ устройства (отсутствие ответа) и на входе параметра «Установить канал» задано -1 (автоматический режим), то происходит переход к следующему резервному каналу. Номер канала, к которому происходит попытка подключения указывается в параметре «Текущий канал». Если подключения по всем каналам были неудачными, то

Категория **Настройки резервирования**	
	у модуля выставляется флаг Отказ, попытки подключения продолжаются по кругу - начиная с основного канала. По умолчанию значение настройки FALSE.
IP адреса	Данная настройка используется только при значении TRUE у настройки «Использовать резервирование». Для резервирования каналов указываются несколько IP адресов через запятую. Переключение на резервный IP адрес произойдет, когда модулю будет выставлен Отказ после достижения максимального количества попыток подключения, указанного в настройке Количество попыток.
Порты	Данная настройка используется только при значении TRUE у настройки «Использовать резервирование». Для резервирования каналов указываются несколько портов через запятую. Если количество IP адресов будет больше чем портов, то будет использоваться порт основного канала. Если для резервированных каналов используется такой же порт как для основного, то данное поле можно оставить пустым.
Вставлять "Отказ" при переключении	При значении TRUE на время переключения канала всем тегам будет выставлен Отказ. Если флаг снят, то отказ устанавливается, когда попытки подключения по всем каналам были неудачными. По умолчанию значение настройки FALSE.

Или, например, протоколе IEC104 (см. рисунок 98). Описание свойств резервирования протокола IEC104 приведено в таблице 18.

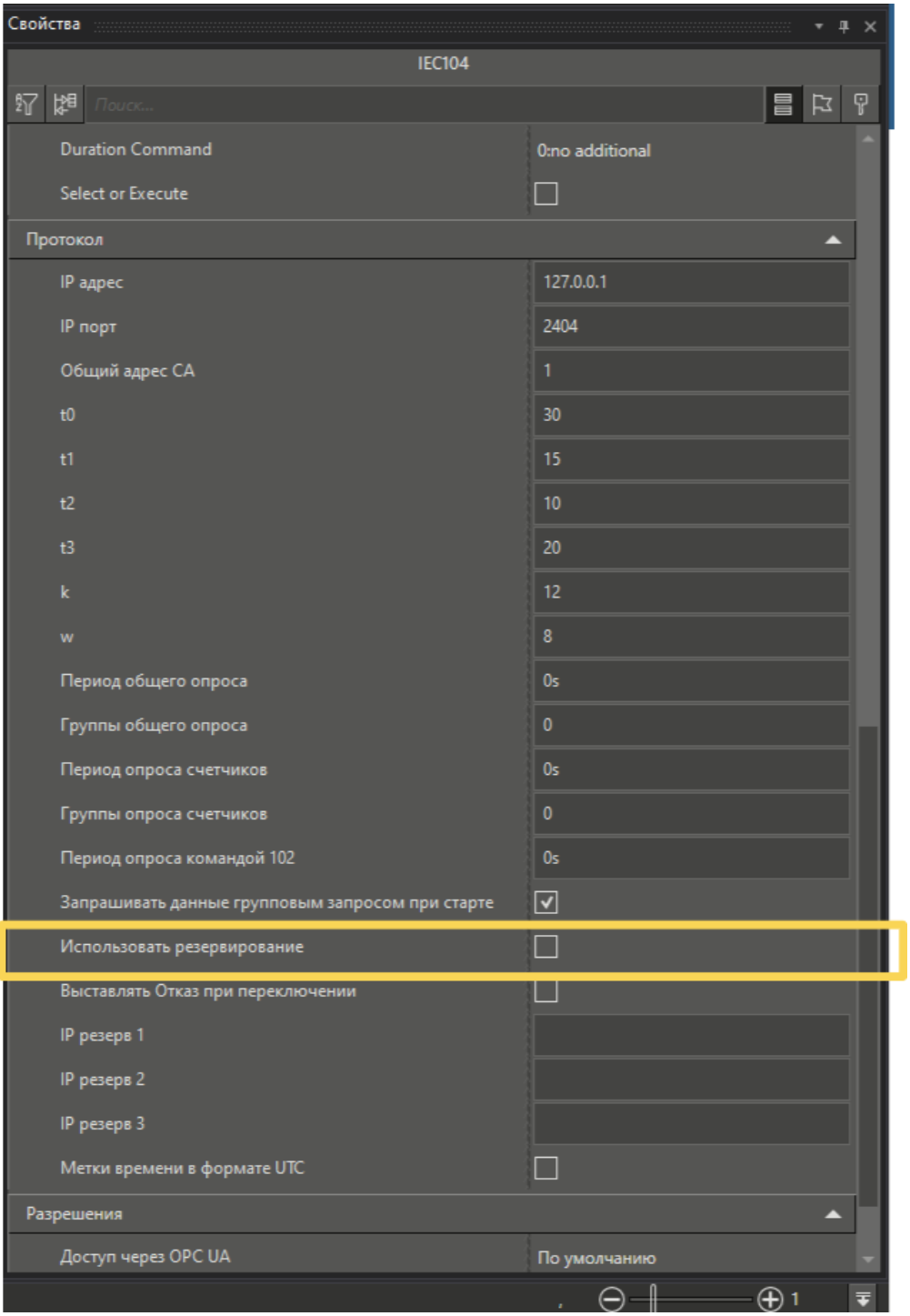


Рисунок 98

Таблица 18

Название	Рекомендации
Категория Протокол	
Использовать резервирование	Включает функцию резервирования каналов связи. В протоколе IEC104 могут использоваться три канала для опроса данных: • 0 - основной канал (IP-адрес); • 1 - первый резервный канал (IP резерв 1); • 2 - второй резервный канал (IP резерв 2).

Название	Рекомендации
	Важно! Нумерация каналов опроса в протоколе идет подряд. Если IP резерв 1 не будет указан, а IP резерв 2 будет, то тогда 1 каналу будет соответствовать адрес, указанный в IP резерв 2. Переключение на резервный канал может осуществляться двумя способами: • Вручную, для этого нужно указать номер резервного канала в параметре протокола «Установить канал». Переключение вручную на другой канал возможно только при наличии связи. • Автоматически, для этого нужно в параметре «Установить канал» указать значение -1. Исполнительная система отслеживает состояние связи с основным каналом, и если связь пропадает происходит переключение на резервный канал 1, в случае отказа резервного канала 1 - переключение на канал 2 и т.д. по кругу. Автоматического возврата на основной канал при его восстановлении не происходит - это необходимо сделать вручную. Автоматическое переключение происходит если устройство закрыло соединение или обрыве TCP-соединения с ним.
Выставлять Отказ при переключении	Определяет - нужно ли выставлять "плохой" признак качества каналам, пока происходит переключение на резервный канал.
IP резерв 1	Первый резервный IP-адрес устройства, которое отправляет и получает данные от MasterSCADA 4D. Если подключение будет по этому каналу, то на выходе «Текущий канал» будет значение 1
IP резерв 2	Второй резервный IP-адрес устройства, которое отправляет и получает данные от MasterSCADA 4D. Если подключение будет по этому каналу, то на выходе «Текущий канал» будет значение 2
Метки времени в формате UTC	По умолчанию значение флага FALSE, в этом случае, при получении метки времени происходит преобразование из локального времени в формат UTC. При значении флага TRUE метка времени будет получаться по локальному времени.

3.6 Загрузка конфигурации в контроллер

Разработанный проект должен быть скомпилирован. Для этого необходимо нажать кнопку «Проверить» панели инструментов вкладки «Исполнение» (рисунок 99).

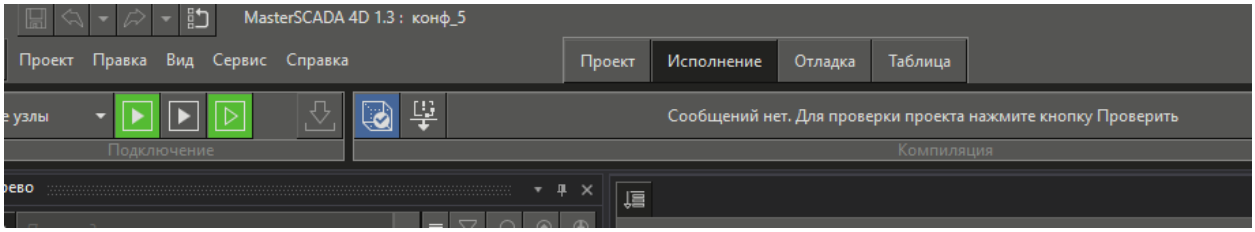


Рисунок 99

Для проверки конфигурации отдельного узла следует выполнить соответствующую команду контекстного меню в дереве системы узла.

При компиляции проекта производится также проверка и компиляция программ.

В случае успешной компиляции проекта кнопка останется зеленой, и на экране отобразится соответствующее сообщение (см. рисунок 100).

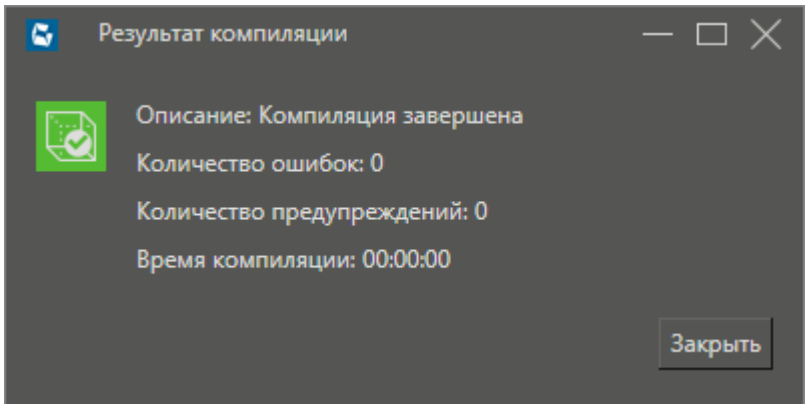


Рисунок 100

Если в результате компиляции не будет ошибок, но будут предупреждения, то кнопка изменит свой цвет на желтый, а на экране отобразится соответствующее предупреждение. Информация об ошибках выведется в окно панели «Компиляция» вкладки «Исполнение».

Для загрузки конфигурации в устройство следует выполнить соответствующую команду контекстного меню в дереве системы узла (см. рисунок 101).

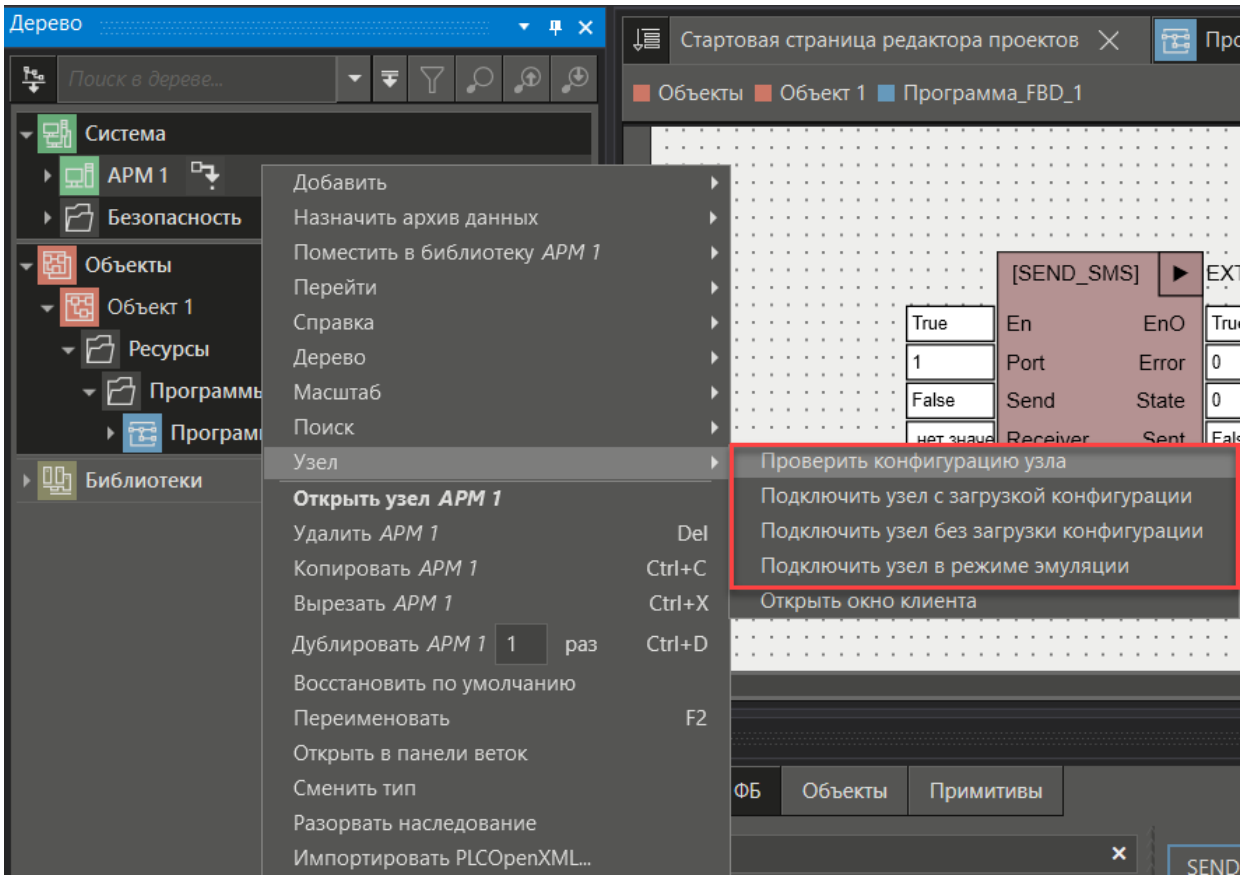


Рисунок 101

Либо нажать кнопку «Подключить и выполнить» на панели

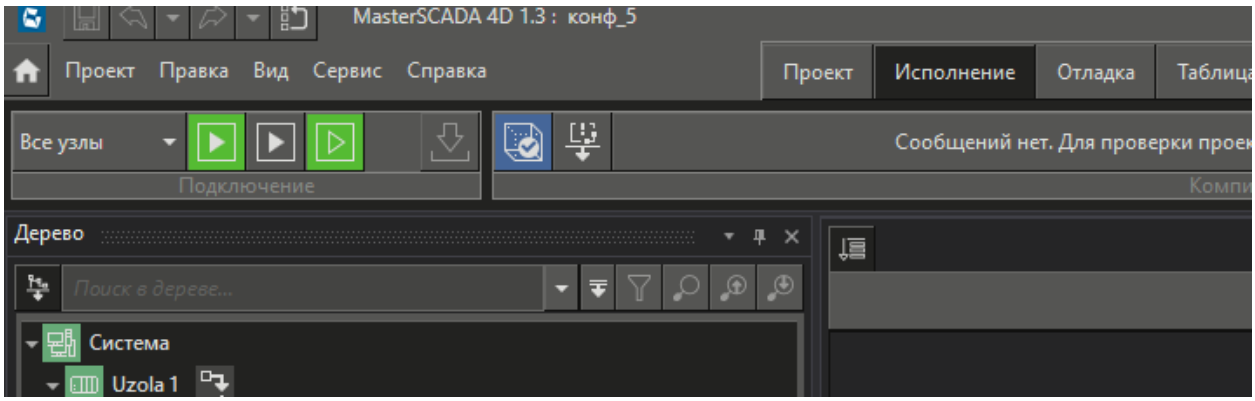


Рисунок 102

До этого должна быть настроена связь с контроллером (ip адрес) в настройках узла.

В случае успешной компиляции и подключения к контроллеру произойдет загрузка проекта в контроллер и перезапуск исполнительной системы. Если будет невозможно установить связь появится соответствующее сообщение (см. рисунок 103).

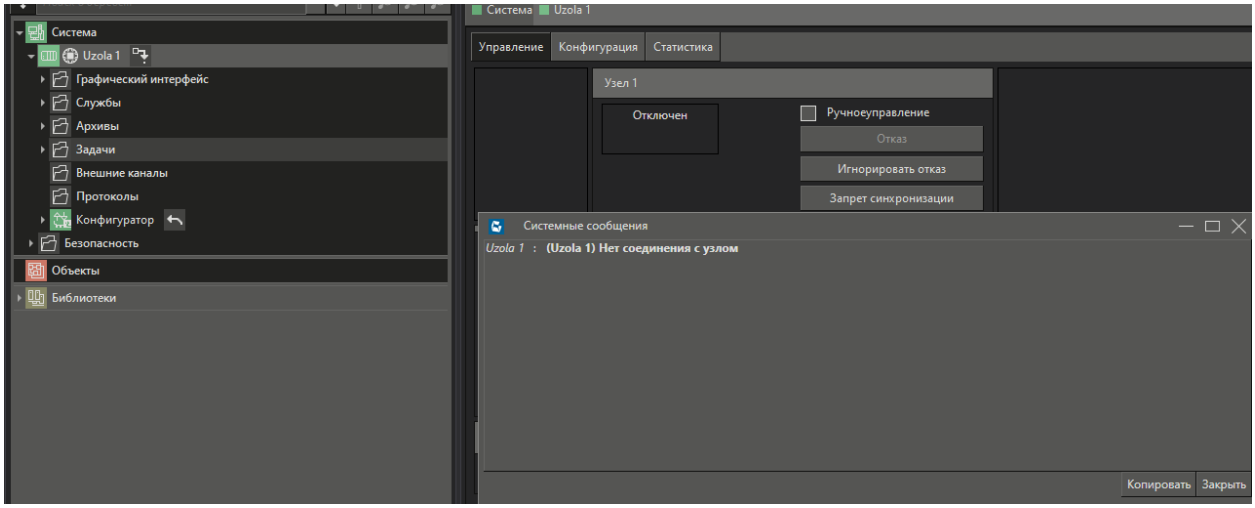


Рисунок 103

В таком случае необходимо проверить доступность контроллера по сети (ping IP контроллера).

На тот случай, если автоматическая загрузка проекта в среду исполнения через среду разработки невозможна, предусмотрена его ручная загрузка путем копирования необходимых файлов в рабочую папку среды исполнения.

Среда исполнения автоматически запускает последний загруженный в нее проект. Под проектом понимается папка, которая создается в результате компиляции проекта. Для каждого узла создается своя папка.

Для того чтобы подготовить файлы для копирования в рабочую папку исполнительной системы, необходимо дважды нажать левой кнопкой мыши на узел в дереве системы. В средней части интерфейса откроется вкладка **управления узлом**, содержащая несколько разделов, в том числе раздел «Конфигурация». Сначала необходимо

нажать на кнопку «Обновить» для формирования или обновления конфигурации, а затем - на кнопку «Экспорт конфигурации проекта» (см. рисунок 104).

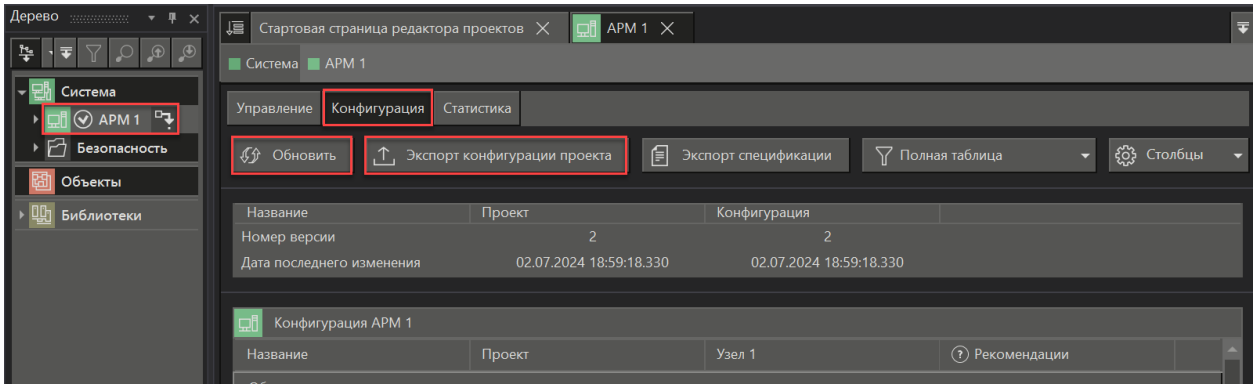


Рисунок 104

После нажатия на кнопку «Экспорт конфигурации проекта» откроется диалоговое окно Windows, предлагающее выбрать папку для сохранения архива проекта, предназначенного для работы в среде исполнения выбранного узла (см. рисунок 105).

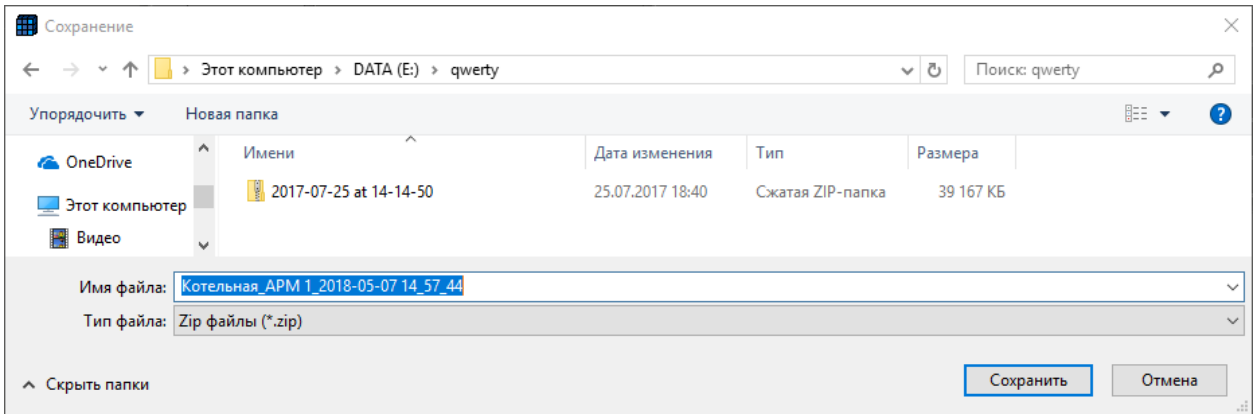


Рисунок 105

После нажатия на кнопку «Сохранить» создастся архив, имеющий, по умолчанию, имя : *[название проекта]_[имя узла]_[дата и время экспорта]*.

После этого содержимое архива необходимо разместить в рабочей папке среды исполнения. Процесс переноса содержимого архива будет зависеть от того, доступна ли файловая система контроллера настройками безопасности.

Для удобства переноса конфигурации проекта на контроллер можно использовать программу WinSCP.

Включите контроллер с установленной средой исполнения.

Создайте новое подключение, указав IP-адрес устройства и существующего пользователя

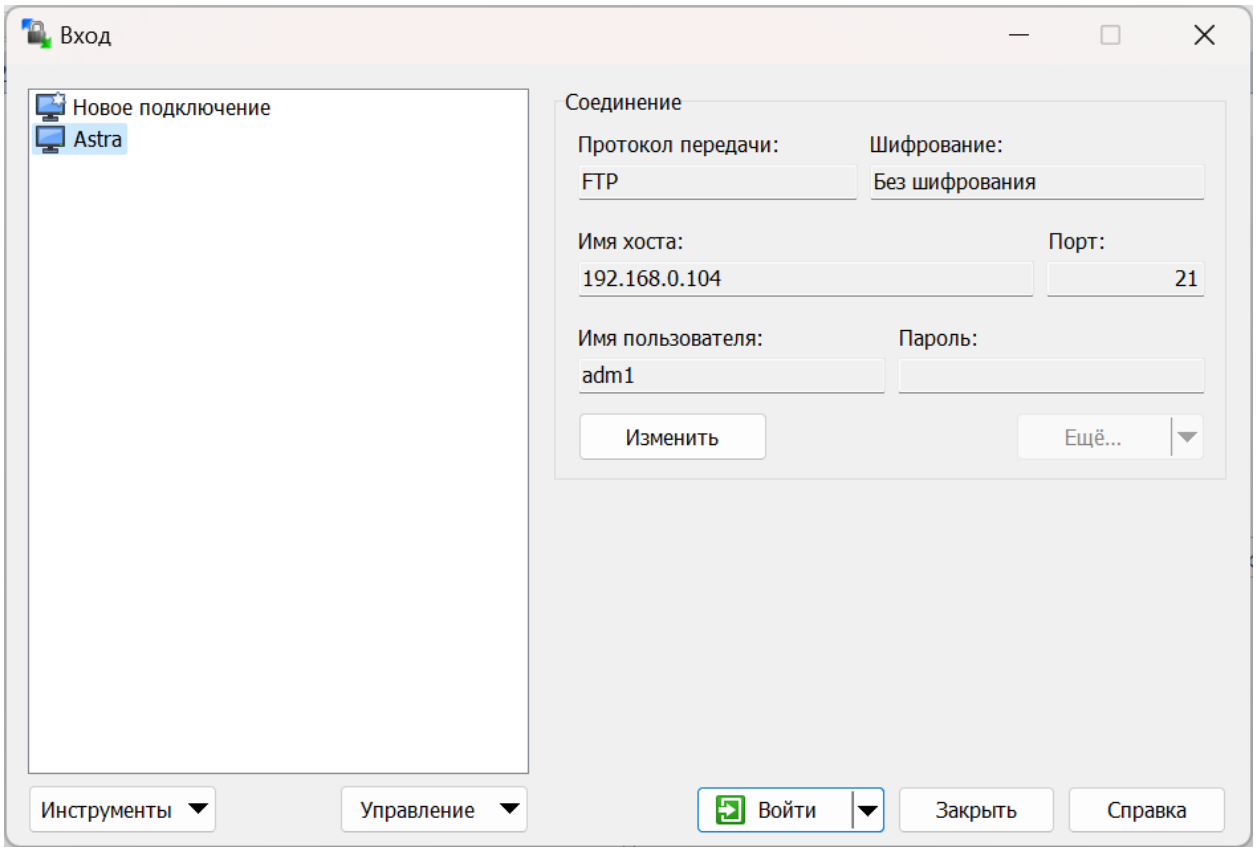


Рисунок 106

3.6.1 После того, как произойдёт подключения к устройству, перейдите в рабочую папку среды исполнения: */opt/mplc4*.

Перед копированием конфигурации проекта в рабочую папку убедитесь, что у пользователя, под которым произошло подключение к устройству, есть права доступа на изменение содержимого папки (рисунок 107).

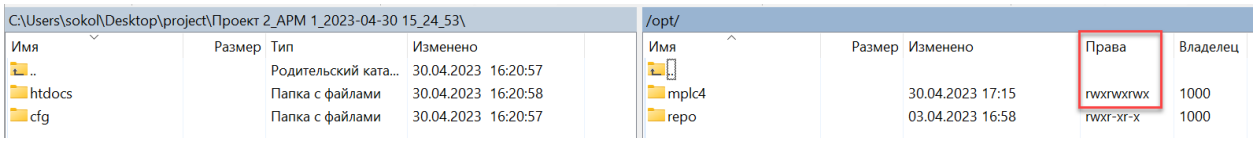


Рисунок 107

Скопируйте содержимое архива в рабочую папку *mplc4*.

Среда исполнения автоматически запускается при запуске системы. Поэтому для того, чтобы перенесённый проект был загружен в среду RT, требуется перезапустить её, используя следующую команду в терминале контроллера:

/etc/init.d/mplc4 restart

3.7 Диагностика

3.7.1 Вывод диагностической информации

Работающая среда исполнения на контроллере выдает диагностическую информацию по TCP/IP через порт 31550. Для чтения этой информации можно воспользоваться любым программным эмулятором терминала, например HyperTerminal, PuTTY и др.

Рассмотрим запись лога с отладочной информацией при помощи свободно распространяемой программы PuTTY – *https://www.putty.org*

Запустить программу можно с помощью ярлыка

.

По умолчанию, программа минимизируется при запуске в панели задач. Для выполнения настройки программы необходимо нажать правой кнопкой мыши на её ярлык в панели задач, и в контекстном меню выбрать пункт New Session (см. рисунок 108).

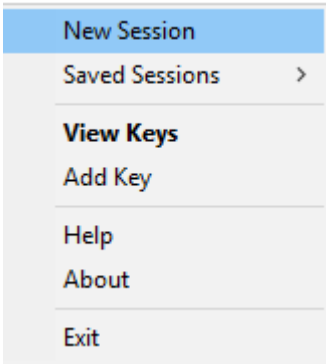


Рисунок 108

При этом откроется окно настроек, в котором следует задать IP-адрес узла с установленной средой исполнения, порт и тип соединения (см. рисунок 109).

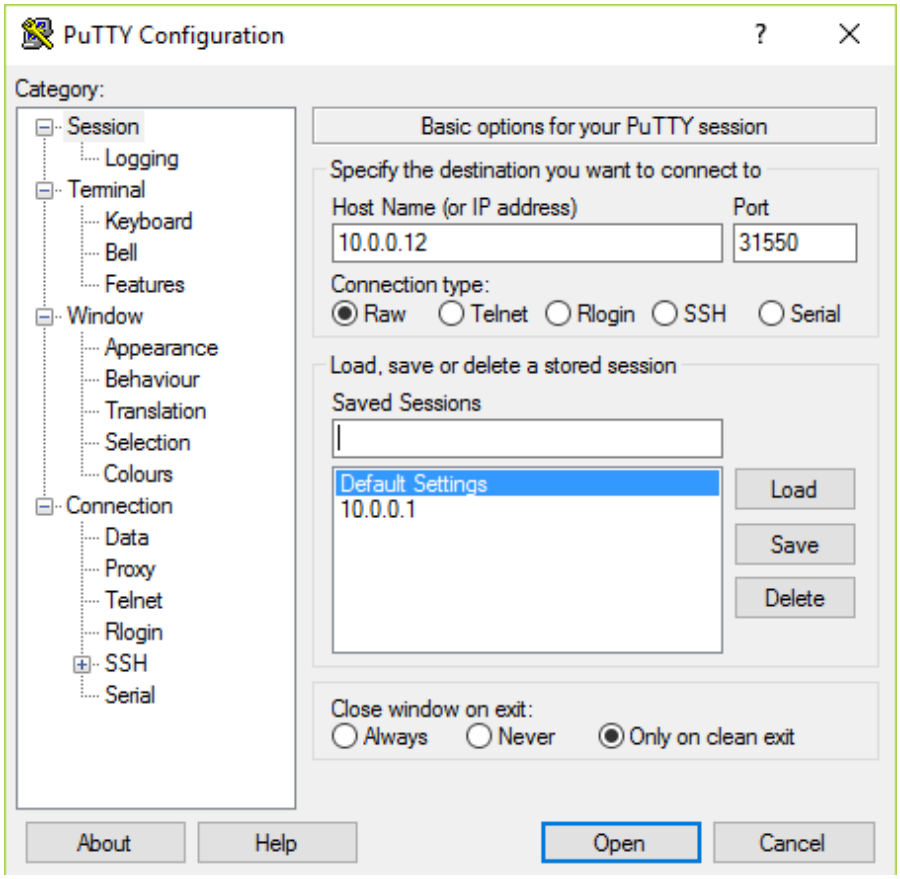


Рисунок 109

После этого требуется определить место хранения лог-файлов:

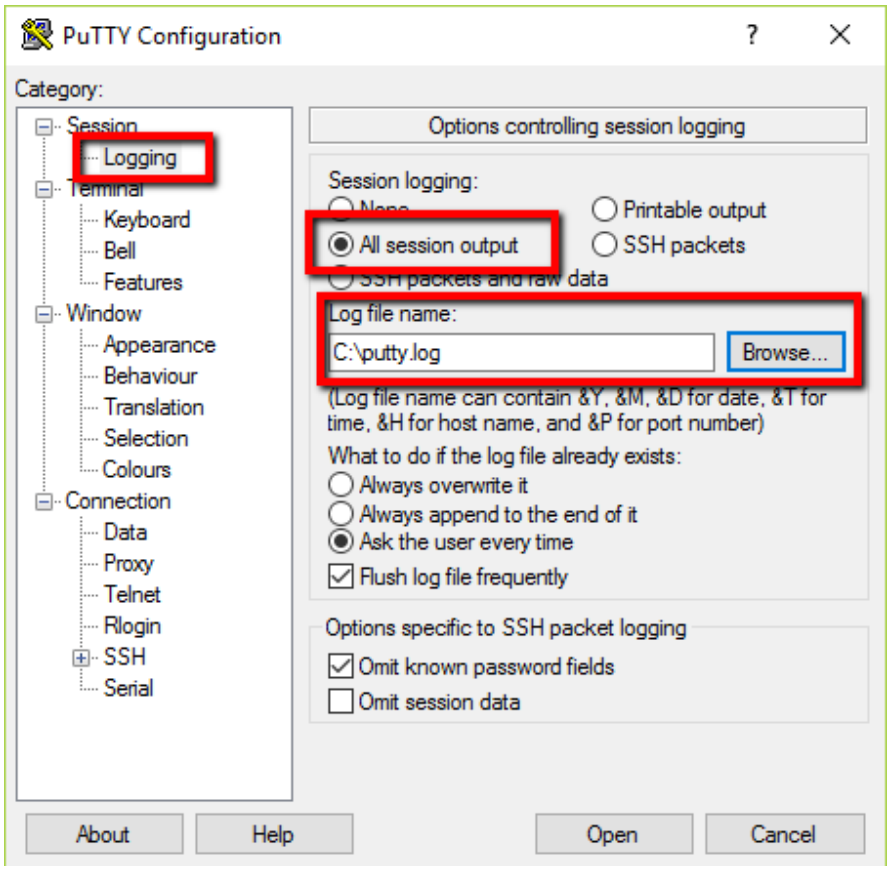


Рисунок 110

Место, в которое планируется сохранить лог-файл, должно быть доступно для записи.

Рекомендуется сохранить данную сессию, для того чтобы в дальнейшем иметь возможность оперативно подключиться к среде исполнения без ввода настроек подключения (см. рисунок 111).

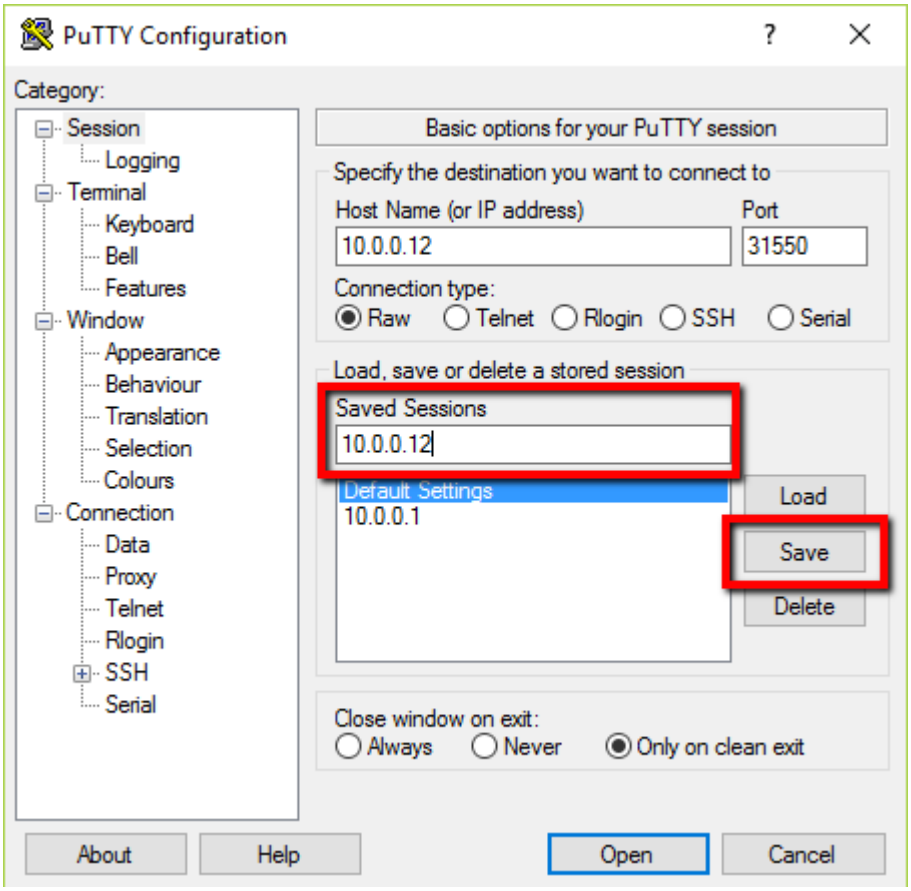


Рисунок 111

Затем необходимо нажать кнопку «Open».

Откроется окно, в которое будет выводиться отладочная информация (см. рисунок 112).

Один раз в 10 секунд будут выводиться сообщения со статистикой всех задач. Информация о работе системы архивирования будет выдаваться один раз в минуту. Файл с сохраненной отладочной информацией необходимо предоставить HelpDesk систему технической поддержки для анализа.

```
127.0.0.1 - PuTTY
05/05 05:53:31.195 N=0 A=0 M=6048Kb
05/05 05:53:41.196 N=0 A=0 M=6048Kb
05/05 05:53:51.198 N=0 A=0 M=6048Kb
05/05 05:54:01.200 N=0 A=0 M=6048Kb
05/05 05:54:11.203 N=0 A=0 M=6048Kb
05/05 05:54:21.208 N=0 A=0 M=6048Kb
05/05 05:54:31.217 N=0 A=0 M=6048Kb
05/05 05:54:41.218 N=0 A=0 M=6048Kb
05/05 05:54:51.222 N=0 A=0 M=6048Kb
05/05 05:55:01.225 N=0 A=0 M=6048Kb
05/05 05:55:11.230 N=0 A=0 M=6048Kb
05/05 05:55:21.232 N=0 A=0 M=6048Kb
05/05 05:55:31.238 N=0 A=0 M=6048Kb
05/05 05:55:41.241 N=0 A=0 M=6048Kb
05/05 05:55:51.247 N=0 A=0 M=6048Kb
05/05 05:56:01.249 N=0 A=0 M=6048Kb
05/05 05:56:11.251 N=0 A=0 M=6048Kb
05/05 05:56:21.254 N=0 A=0 M=6048Kb
05/05 05:56:31.259 N=0 A=0 M=6048Kb
05/05 05:56:41.262 N=0 A=0 M=6048Kb
05/05 05:56:51.268 N=0 A=0 M=6048Kb
05/05 05:57:01.268 N=0 A=0 M=6048Kb
05/05 05:57:11.273 N=0 A=0 M=6048Kb
```

Рисунок 112

Важно! Для получения расширенной диагностической информации требуется нажать клавишу клавиатуры *o*, а затем *Enter*.

В этом случае на экран будет выдаваться трассировка (см. рисунок 113):

```
127.0.0.1 - PuTTY
2171953: End get (count=15) dt=0 q=155 [addr=127.0.0.1 port=62340]
2171980: err=-3 Task0 ModbusRead module=0(A=1), request=0, StartAddr=0, Count=9
1 3
0ModbusTCP read 010.000.006.010
0ModbusTCP write 010.000.006.010
0ModbusTCP write 010.000.006.010
2172055: End get (count=15) dt=0 q=155 [addr=127.0.0.1 port=62340]
2172087: err=-3 Task=0 ModbusWrite module=1(A=2), request=0, StartAddr=0, Count=
1 1 0
2172155: End get (count=15) dt=0 q=155 [addr=127.0.0.1 port=62340]
2172187: err=-3 Task=0 ModbusWrite module=1(A=2), request=1, StartAddr=1, Count=
1 2 0
2172256: End get (count=15) dt=0 q=155 [addr=127.0.0.1 port=62340]
2172288: err=-3 Task=0 ModbusWrite module=1(A=2), request=2, StartAddr=2, Count=
1 3 0
Trace disabled !
2174562: Error TCP connect : 0 (port=502 state=0) 010.000.006.010
2177849: Error TCP connect : 0 (port=502 state=0) 010.000.006.010
11/05 06:06:44.822 N=492 A=0 O0=570(0,e0) U0=587(100,100,104)(0,0,1) S0=97,e97(6
04,458,2409)(599,452,2404) HR=1(4) M=7360Kb
MT 0.00 MR 0 NRq 0 NI 0 R(0 0) Rq(0 0) WI(-0.00 0.00 0.00) TWII(-0.00 0.00 0.00)
 TWIT(-0.00 0.00 0.00) RqI(-0.00 0.00 0.00) RqT(-0.00 0.00 0.00) DR(0 0)  TDII(-
0.00 0.00 0.00) TDIT(-0.00 0.00 0.00)
```

Рисунок 113

Логи состоят из 5 уровней сообщений:

• INF - информационные сообщений;

• TRA - трассировка (обмен с оборудованием);

• WAR - предупреждения. Для того, чтобы предупреждения записывались в логи исполнительной системы, необходимо, чтобы в свойстве узла «Параметры запуска RT» был установлен ключ /w;

• SYS - системные сообщения;

• ERR - ошибки.

Сообщения формируются в следующем формате:

Time: |Thread| [level] Message

Например:

8876153: |11680| [INF] *(10.30.6.96:5000) <Mitsubishi SLMP> Received 13 bytes*

8876153: |11680| [TRA] *(10.30.6.96:5000) Tx: [21] 50 00 00 FF FF 03 00 0C 00 00 00 01 04 00 00 90 01 00 A8 02 00*

8879193: |11680| [TRA] *(10.30.6.96:5000) Rx: [12] D0 00 00 FF FF 03 00 03 00 00 00 00*

8879193: |11680| [TRA] *(10.30.6.96:5000) Tx: [21] 50 00 00 FF FF 03 00 0C 00 00 00 01 04 01 00 01 00 00 C1 01 00*

8879251: |11680| [TRA] *(10.30.6.96:5000) Rx: [13] D0 00 00 FF FF 03 00 04 00 00 00 00 00*

8879251: |11680| [INF] *(10.30.6.96:5000) <Mitsubishi SLMP> Read done*

Выделенное жирным - информация, сформированная макросом, остальное - пользовательское сообщение.

В круглых скобках выводится информацию о подсистеме. Например, для драйвера - параметры подключения к устройству, для архивной системы - параметры подключения БД.

В треугольных скобках выводится статичное название подсистемы - название драйвера, подсистемы архивации и т.д.

3.7.2 Периоды выполнения задач

Каждые 10 сек отображается диагностическое сообщение со статистикой по всем задачам:

M14/04 20:28:39.771 N=880550(1,71) N2=34707 A=0 O0=41552(0,e0)
U0=167423(100,100,422)(90,83,115)(,0,42,46)13797Kb
U1=16746(1000,1000,1454)(1,1,1454)(,0,0,1)525KbS0=314891,e0/0(53,49,57)(2,1,27)368Kb
HR=16603(6) R=828487(0,0,e1) M=110060Kb

где:

• В начале строки признак M - Master, S - Slave

• дата - время

• N - количество обработанных запросов UDP (за исключением отдельных типов, которые считаются в N2). В скобках указывается - <максимальное время обработки запроса, N функции протокола, по которой была самая долгая обработка>

• N2 - количество обработанных запросов UDP, которые используются для взаимодействия со средой разработки при отладке программ при указанной опции /udp2, а также при синхронизации архивов (данные запросы обрабатываются в отдельном потоке)

• M=<кол-во обработанных запросов> - в случае Modbus RTU Slave задачи

• A=<количество обработанных запросов от HMI клиентов>

• O<индекс контроллера>=<кол-во циклов>(<время последнего цикла опроса>,<общее кол-во ошибок>) - статистика задачи межузловой связи (для связи с каждым контроллером отдельная задача).

• U<индекс задачи>=<кол-во циклов>(<среднее цикла>,<мин цикла>,<макс цикла>)(<среднее реального выполнения>,<мин>,<макс>)(,<время чтения входных данных в последнем цикле>,<время выполнения задачи в последнем цикле>,<время записи выходных данных в последнем цикле>)<объем памяти виртуальной машины>Kb - статистика задачи объектов

• S<индекс задачи>=<кол-во циклов>,e<общее кол-во ошибок>/<кол-во ошибок в последнем цикле>(<среднее цикла>,<мин цикла>,<макс цикла>)(<среднее реального выполнения>,<мин>,<макс>)<объем памяти виртуальной машины>Kb - – статистика задачи протоколов

• HR=<кол-во сохранений горячего рестарта>(<время, затраченное на последнее сохранение>)

• R=<кол-во циклов задачи резервирования>(<время последнего цикла>,<время последней синхронизации данных>,<макс время ответа от другого контроллера>,<кол-во ошибочных ответов>). В режиме SLAVE на каждом цикле выполняется проверка состояния второго контроллера, а также получение данных, если с прошлого получения прошел заданный период задачи. В конце каждого цикла дополнительно пауза 20мс

• M=<используемый объем памяти процессом>Kb

3.7.3 Статистика архивирования

Статистика архивирования выдается один раз в минуту. Строка с информацией имеет вид:

Stat: {Archive Info}

R(Tm:float, Ct:int, Av:int, Rc:int, Rq:int) W(Tm:float, Ct:int, Av:int, Wt:int, Ps: int, Ls:int)

D(Tm:float, Ct:int, Av:int)

где:

R - Чтение

W - Запись

D - Удаление

Tm - Число секунд, затраченных на обработку R/W/D

Ct - Количество записей обработанных за последнюю минуту R/W/D

Av - Средняя скорость в сек процессорного времени R/W/D

Wt - Количество записей в очереди на запись в БД

Ps: - Количество записей в сек реального времени

Ls: - Количество удаленных записей в результате прореживания (по переполнению очереди записи)

Rc: - Количество выполненных запросов чтения

Rq: - Количество запросов чтения в очереди

3.7.4 Статистика передачи архивов между узлами

RemoteData(<индекс> ip=<адрес>) cycle=<кол-во полных циклов опроса> period=<период последнего цикла в мс> last_err=<кол-во ошибок на последнем цикле>/<кол-во опрашиваемых параметров> all_err=<кол-во ошибок всего> values=<кол-во значений на последнем цикле>/<кол-во значений всего> iteration=<кол-во проходов на последнем цикле> more=<на последнем цикле не все данные прочитаны>

3.7.5 Записи о начале и завершении сессии пользователя

Информация о начале и завершении сессии пользователем фиксируется в лог файлы. В таких отладочных сообщениях выводится хешированное значение SessionID.

Начало сессии:

Session created 'хешированный SessionID' by login 'Имя пользователя'

Login source:127.0.0.1 login:'Имя пользователя' code:0 out_msg:Вход в систему

Завершение сессии:

Logout 'Имя пользователя' (session 'хешированный SessionID')

3.7.6 Вывод логов в драйверах

Для вывода отладочных сообщений о драйверах необходимо сформировать расширенный лог.

Лог имеет вид: *Время: |Task| [Level] Message*

Строки с информацией о чтении и отправки данных с устройства имеют уровень TRA (трассировки):

77581943: |12304| [TRA] (127.0.0.1:502) Tx: [0012] 04 00 00 00 00 06 01 03 00 00 00 01

77581953: |12304| [TRA] (127.0.0.1:502) Rx: [0011] 04 00 00 00 00 05 01 03 02 00 00

77581943: |12304| [TRA] (COM1) Tx: [0012] 04 00 00 00 00 06 01 03 00 00 00 01

77581953: |12304| [TRA] (COM1) Rx: [0011] 04 00 00 00 00 05 01 03 02 00 00

В круглых скобках выводятся параметры подключения к устройству (IP-адрес, порт).

3.7.7 Логи HMI клиента

Если в настройках среды на вкладке «Лог» установлен флаг «Передача на сервер лога HMI клиента», то все сообщения об ошибках клиента визуализации будут записываться в лог исполнительной системы.

Пример лога с предупреждением:

30551875: |24868| [WAR] 2024-08-09T15:31:52.381Z In file F:\azure-agents\MS4\SRV-BUILD-1_work\3\s\ms4-src\Source\MasterPLC\addins\\mplc_data

\data_request_processor.cpp at line 407 2024 -08-09 15:31:52.381 (from @127.0.0.1) : 09.08.2024, 18:31:52 convert warning FROM id="491154" TO

*id="491126":
 Значение "Это текст" не может быть преобразовано в тип "DATE AND TIME"*

30551875: |24868| [WAR] 2024-08-09T15:31:52.382Z In file F:\azure-agents\MS4\SRV-BUILD-1_work\3\s\ms4-src\Source\MasterPLC\addins\mplc_data

\data_request_processor.cpp at line 407 2024 -08-09 15:31:52.382 (from @127.0.0.1) : 09.08.2024, 18:31:52 convert warning FROM id="491154" TO

*id="491140":
 Значение "Это текст" не может быть преобразовано в тип "TIME_OF_DAY"*

Для получения диагностической информации встроенными средствами MasterSCADA 4D можно использовать специальные функциональные блоки библиотеки BaseObjects:

• GetSystemInfo - получение информации об установленной версии исполнительной системы и загруженном проекте.

• RedundancyControl - управление резервированием.

• FileIntegrityControl - управление контролем целостности ПО и проекта.

• GetOpcUaServerInfo - получения информации и статистики по работе OPC UA сервера.

• GetDataArchiveStatistic - получения информации и статистики по работе архива данных.

• GetEventsArchiveStatistic - получения информации и статистики по работе архива данных.

• GetTaskStatistics - получение статистики работы задач.

3.8 Настройка прав доступа

В группе «Безопасность» содержатся элементы для настройки прав доступа операторов (пользователей) в режиме исполнения.

Права доступа могут назначаться как индивидуально, так и для группы лиц. В связи с этим, группа «Безопасность» содержит в себе следующие основные подгруппы: «Пользователи», «Роли», «Права доступа» и «Группы прав», также вспомогательные «Архивы» и «Настройки пользователей» (см. рисунок 114).

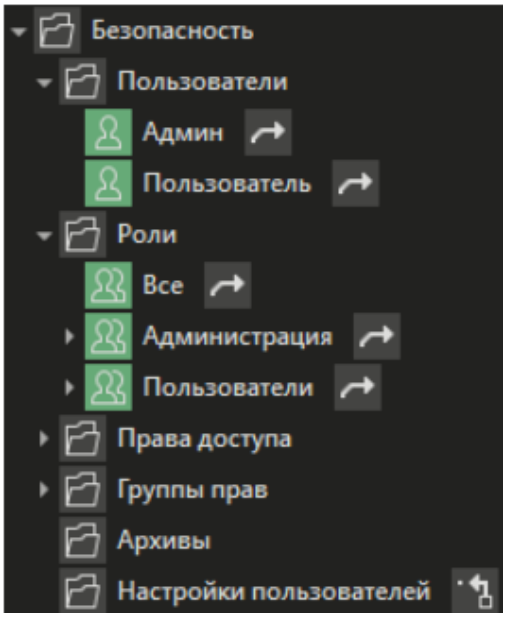


Рисунок 1114

Если выделить группу «Безопасность», перейти в панель «Свойств», и переключиться в режим просмотра прав доступа, то можно получить сводную таблицу разрешений по всем пользователям и ролям, добавленным в проект (см. рисунок 115).

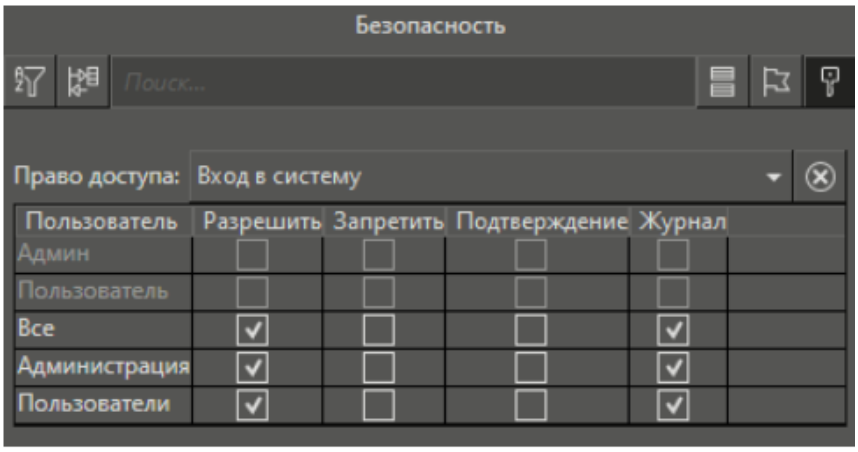


Рисунок 115

Список прав доступа показан на рисунке ниже.

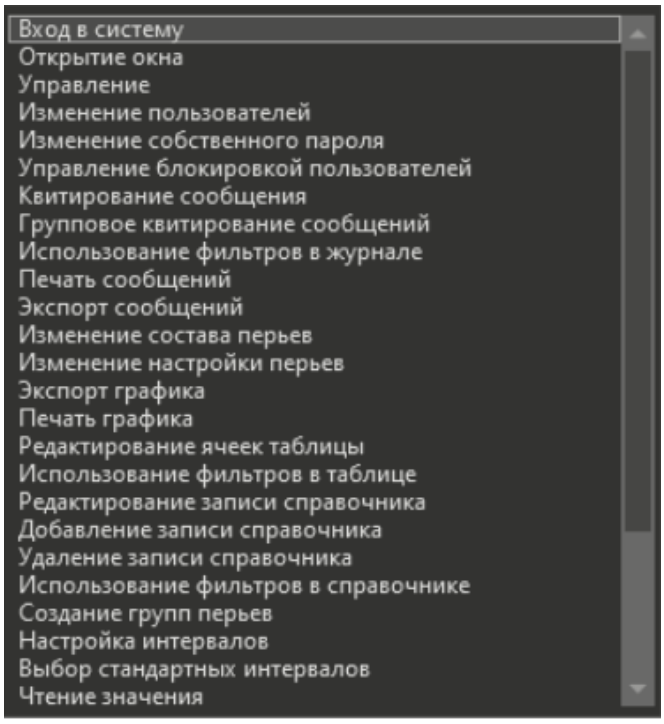


Рисунок 116

Таблица 19

Название	Программное имя	Назначение
Общие		
Вход в систему	Login	Позволяет пользователю залогиниться в клиенте визуализации, который подключен к тому или иному узлу. Если в проекте несколько узлов, и для каждого нужно назначить разные права доступа, то это право доступа находится в панели свойств узла. По умолчанию для данного права доступа установлен флаг «Журнал».
Открытие окна	OpenWindow	Позволяет пользователю открывать окна в режиме исполнения в клиенте визуализации. Если у пользователя нет прав на открытие окна, то появится

Название	Программное имя	Назначение
		соответствующее сообщение. Данное право настраивается в дереве системы по умолчанию для всех окон проекта, в дальнейшем умолчания можно изменить в объекте, для всех его дочерних окон, и непосредственно в каждом отдельном окне.
Управление	Control	Позволяет нажимать на кнопки, изменять значения, при помощи различных графических элементов, например, Числовой ввод, Текстовый ввод и др. Если право на управление имеется, то будут также исполнятся назначенные действия. Данное право настраивается в дереве системы по умолчанию для всех элементов проекта, в дальнейшем право можно изменить для конкретного элемента в его панели свойств. Действия захват, покидание и перемещение мыши не учитываются.
Управление пользователями		
Изменение пользователей	ChangeUsers	Позволяет изменять параметры пользователей в режиме исполнения. Если у права стоит флаг «Журнал», то в журнал сообщений выдается подробная информация: у какого пользователя что поменяли (если меняется пароль, то текст пароля не пишется, в остальных случаях указывается старое и новое значение настройки/имена измененных ролей). ChangeUsers - программное имя права, для использования его в программах ST. Если установлен флаг «Журнал», напротив данного права, то будет сформировано сообщение, содержащее информацию как о новом значении, так и о предыдущем, например, Переименован пользователь с именем Иванов на пользователя Петров. Тип сообщения будет «Действие пользователей».
Изменение собственного пароля	ChangeSelfPassword	Позволяет изменить собственный пароль пользователя, который авторизован в текущий момент в клиенте визуализации. Если у права стоит флаг «Журнал», то в журнал сообщений выдается подробная информация: у какого пользователя что поменяли (если меняется пароль, то текст пароля не пишется, в остальных случаях указывается старое и новое значение настройки/имена измененных ролей). ChangeSelfPassword - программное имя права, для использования его в программах ST.
Управление блокировкой пользователей	BlockUsers	Позволяет заблокировать или снять блокировку пользователя. Блокировка учетной записи пользователя, это состояние, при котором пользователь не может запустить новый сеанс работы в клиенте визуализации, но сведения об этом пользователе не удаляются из исполнительной системы.
Чтение пользователей	ReadUsers	Позволяет считывать информацию о пользователях в режиме исполнения, например, информация о всех

Название	Программное имя	Назначение
		пользователях, всех ролях пользователей или всех сессиях пользователей. По умолчанию – «Разрешено» для роли «Все».
Права доступа элементов		
Квитирование сообщений, Групповое квитирование сообщений, Использование фильтров в журнале, Печать сообщений, Экспорт сообщений	JournalAck, JournalGroupAck, JournalUseFilters, JournalPrint, JournalSave	Данные права доступа относятся к элементам «Журнал» и «Архивный журнал». В дереве системы настраиваются умолчания для всех элементов данного типа, которые будут использоваться в проекте. Для каждого отдельного элемента права доступа можно будет изменить, а также изменить умолчания у родительского элемента, например, изменение данного права у объектов, приведет к изменению умолчаний для всех элементов данного типа, которые находятся в окнах этого объекта.
Изменение состава перьев, Изменение настройки перьев, Экспорт графика, Печать графика	TrendAddPens, TrendChangePens, TrendSave, TrendPrint	Данные права доступа относятся к элементам «Тренд», «Легенда», «График XY». В дереве системы настраиваются умолчания для всех элементов данного типа, которые будут использоваться в проекте. Для каждого отдельного элемента права доступа можно будет изменить, а также изменить умолчания у родительского элемента, например, изменение данного права у объектов, приведет к изменению умолчаний для всех элементов данного типа, которые находятся в окнах этого объекта.
Редактирование ячеек таблицы, Использование фильтров в таблице	DataTableEdit, DataTableFilter,	Данные права доступа относятся к элементу «Таблица данных». В дереве системы настраиваются умолчания для всех элементов данного типа, которые будут использоваться в проекте. Для каждого отдельного элемента права доступа можно будет изменить, а также изменить умолчания у родительского элемента, например, изменение данного права у объектов, приведет к изменению умолчаний для всех элементов данного типа, которые находятся в окнах этого объекта.
Редактирование записи справочника, Добавление записи справочника, Удаление записи справочника, Использование фильтров в справочнике	DirectoryEdit, DirectoryAdd, DirectoryDelete, DirectoryFilter	Данные права доступа относятся к элементу «Справочник». В дереве системы настраиваются умолчания для всех элементов данного типа, которые будут использоваться в проекте. Для каждого отдельного элемента права доступа можно будет изменить, а также изменить умолчания у родительского элемента, например, изменение данного права у объектов, приведет к изменению умолчаний для всех элементов данного типа, которые находятся в окнах этого объекта.
Создание групп перьев, Настройка перьев,	PenGroupsCreate, TrendChangePens, IntervalsSettings, StandardIntervalsSel	Данные права доступа относятся к элементу «Столбчатая диаграмма». В дереве системы настраиваются умолчания для всех элементов данного типа, которые будут использоваться в проекте. Для

Название	Программное имя	Назначение
Настройка интервалов, Выбор стандартных интервалов, Экспорт графика, Печать графика	ect, TrendSave, TrendPrint	каждого отдельного элемента права доступа можно будет изменить, а также изменить умолчания у родительского элемента, например, изменение данного права у объектов, приведет к изменению умолчаний для всех элементов данного типа, которые находятся в окнах этого объекта.
Права на параметры		
Чтение значения, Запись значения	ReadData, WriteData	Данное право обеспечивает возможность для определенных пользователей/ролей ограничивать про смотр и изменение параметров. По умолчанию – «Разрешено» для роли «Все». Данное право настраивается в дереве системы по умолчанию для всех элементов проекта, в дальнейшем право можно изменить для конкретного элемента в его панели свойств. Право доступа может быть определено в: Параметре, Программе, ФБ , Объекте, Узле, Модуле, Протоколе, Канале, Теге.
Права на программы		
Вызов программы	CallProgram	Данное право обеспечивает возможность для определенных пользователей/ролей ограничивать вызов программ (FBD, LD, SFC, ST, C#) и ФБ. По умолчанию – «Разрешено» для роли «Все». Данное право настраивается в дереве системы по умолчанию для всех элементов проекта, в дальнейшем право можно изменить для конкретного элемента в его панели свойств. Право доступа может быть определено в: Программе, ФБ , Объекте, Узле, Модуле, Протоколе, Канале, Теге.

3.9 Связь с процессорным модулем

В КТСИ UZOLA PRO100 реализованы два независимых контроллера Ethernet (10/100 и 10/100/1000 base-TX). Коммуникационные порты Ethernet расположены снизу МП. Порт Ethernet1(в среде MasterSCADA 4D – Eth0) имеет адрес по умолчанию – 192.168.1.110, Ethernet2 (в среде MasterSCADA 4D – Eth1) имеет адрес по умолчанию – 192.168.5.150. Переопределение адресов портов осуществляется посредством ПО «Конфигуратор ПЛК Uzola» (п.4.5 «Руководство по установке и эксплуатации ПО «Конфигуратор ПЛК Uzola»).

Примечание - Одновременная работа обоих портов Ethernet в одной сети некорректна.

3.10 Работа модулей в системе КТСИ

3.10.1 Работа контроллера в целом и программная модель

В КТСИ UZOLA PRO100 обмен между процессорными модулем и модулями ввода-вывода осуществляется по CAN – шине на скорости 500 Кбод. Используется протокол обмена CANopen с элементами профилей DS-301, DS-401.

Процессорный модуль является ведущим устройством, модули ввода-вывода являются ведомыми устройствами. Каждому ведомому устройству присваивается свой адрес (NodeID). Нумерация адресов начинается с 1 и увеличивается слева направо, т.е. ближайший к процессорному модулю модуль ввода-вывода должен иметь адрес 1, следующий справа модуль ввода-вывода должен иметь адрес 2 и т.д.

Параметры, которые может принимать или передавать модуль ввода-вывода (также называемые «словарь объектов»), содержатся в файле описания устройства (*.eds). Каждому типу модуля ввода-вывода соответствует свой словарь объектов; надо также иметь ввиду что при смене версии программного обеспечения модуля ввода-вывода словарь объектов может измениться. Объекты словаря обычно делят на PDO и SDO. С точки зрения программной модели, объекты PDO – это параметры модуля ввода-вывода, которые передаются инициативно, а объекты SDO – это параметры модуля ввода-вывода, которые передаются по запросу.

При работе протокола CANopen на ведущем устройстве, каждому объявленному ведомому устройству присваивается СОСТОЯНИЕ в соответствии с машиной переходов протокола. Основные состояния: NOT_AVIAIL, UNKNOWN, RESET, INIT, PRE_OPERATIONAL, OPERATIONAL, STOPPED. Основные рабочие состояния: INIT, PRE_OPERATIONAL, OPERATIONAL; остальные состояния имеют служебный характер или возникают при обработке ошибок.

После установления первичной связи с ведомым устройством, этому устройству присваивается состояние INIT. В состоянии INIT осуществляется запрос идентификационных данных ведомого устройства. Если возвращаемые идентификационные данные совпадают с идентификационными данными словаря объектов, соответствующему данному устройству, осуществляется переход в состояние PRE_OPERATIONAL.

В состоянии PRE_OPERATIONAL запрещена передача PDO, разрешена передача SDO. В частности, в этом состоянии мастер передает ведомому устройству все SDO, которым при создании проекта был присвоен статус «передать при инициализации». Как правило, эти SDO содержат настроечные и конфигурационные параметры. После передачи

таких SDO мастер дает ведомому устройству команду на переход в состояние OPERATIONAL.

В состоянии OPERATIONAL работа модуля ввода-вывода по назначению осуществляется в полном объеме.

3.11 Методика поиска отказов

Исправное функционирование модулей КТСИ обозначается постоянным свечением зеленым цветом светодиодов «Питание», «CAN».

3.12 Устранение отказов

При возникновении отказов следует обратиться в службу технической поддержки изготовителя.

КТСИ поддерживает функцию «горячей замены» модулей ввода/вывода (кроме процессорного модуля) при возникновении отказов.

При обнаружении аппаратных неисправностей следует руководствоваться положениями, изложенными в руководстве по эксплуатации.

3.13 Техническая поддержка

Для обращения в службу технической поддержки Пользователю необходимо сформировать запрос на сайте технической поддержки: https://uzola.ru, либо отправить письмо по электронной почте: info@uzola.ru.

Номер телефона технической поддержки КТСИ: +7(920)0722008.

Обращение обязательно должно содержать следующие сведения:

- подробное описание проблемы;
- наименование объекта и его месторасположение;
- наименование системы автоматизации;
- серийный номер КТСИ;
- версия инструментальной системы MasterSCADA 4D;
- файл экспорта сетевых настроек контроллера;
- архив с лог-файлами, включающими в себя период времени, когда произошел отказ;
- дата и время возникновения отказа;

- а также периодичность и устойчивость повторения подобных отказов в случае, если такая информация имеется.

Желательно прислать проект для MasterSCADA 4D, т.к. это может значительно упростить и ускорить процесс поиска причины отказа.

Лог-файлы, скопированные на компьютер, желательно поместить в архив. Объем заархивированных текстовых файлов сокращается примерно в 10 раз.

3.14 Информация об Изготовителе

Изготовитель: ГК «Узола», Россия, г. Нижний Новгород, ул. Ларина, д.7а

Телефон: 8-800-7-759-759.

Сайт: https://uzola.ru

Email: info@uzola.ru

Лист регистрации изменений

Изм.	Номера листов (страниц)				Всего листов (страниц) в документе	Номер документа	Входящий номер сопроводительного документа и дата	Подпись	Дата
	изме-нённых	заме-нённых	новых	аннули-рованных					