

Unimod PRO 2

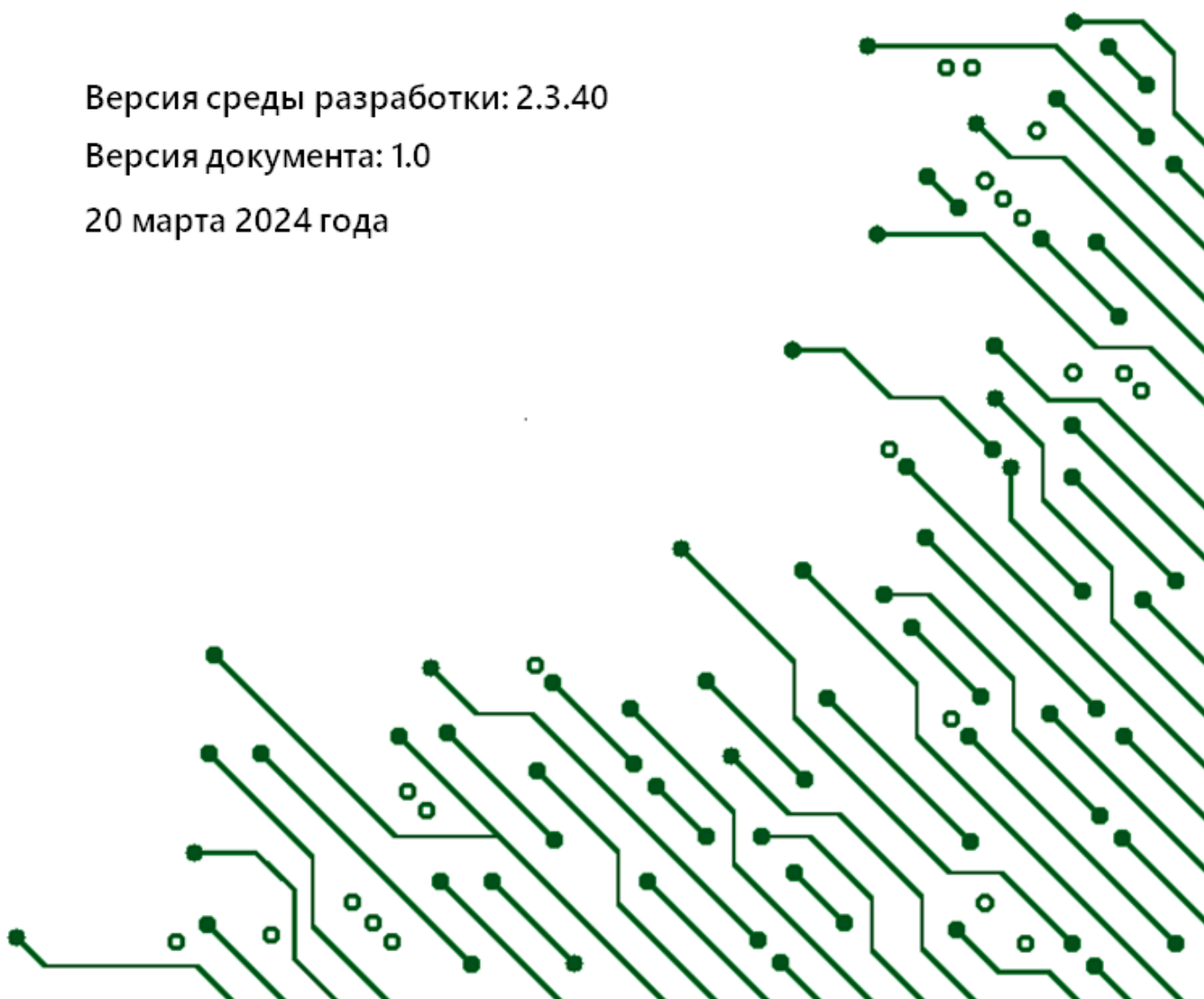
Среда разработки технологических приложений на языках стандарта МЭК 61131-3 для ПЛК TREI-5B

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Версия среды разработки: 2.3.40

Версия документа: 1.0

20 марта 2024 года



Оглавление

Введение	5
1 Общие сведения.....	7
1.1 Функциональные возможности	8
1.2 Системные требования.....	10
1.3 Совместимость проектов.....	11
1.4 Доступные для программирования контроллеры	13
2 Подготовка к работе	14
2.1 Установка среды разработки.....	15
2.1.1 Установка в Windows	16
2.1.2 Установка в Astra Linux.....	19
2.2 Запуск среды разработки.....	20
2.3 Настройка среды разработки.....	21
2.3.1 Настройка рабочего пространства	22
2.3.2 Настройка среды разработки	26
2.4 Закрытие среды разработки	28
2.5 Обновление среды разработки.....	29
2.6 Удаление среды разработки	32
3 Краткий обзор.....	33
4 Пользовательский интерфейс	47
4.1 Инструментальные панели	53
4.2 Редакторы	59
4.2.1 Редактор словаря	61
4.2.1.1 Управление переменными	62
4.2.1.2 Поиск и фильтрация	66
4.2.1.3 Экспорт/Импорт переменных.....	68
4.2.2 Редактор ST	71
4.2.3 Редактор FBD/LD.....	80
4.2.4 Управление графическими элементами FBD/LD	86
5 Состав проекта	88
5.1 Переменные	89
5.1.1 Атрибуты переменных.....	91
5.1.2 Экспорт тегов	95
5.2 Задачи.....	97
5.3 Программы.....	98
5.4 Дочерние программы.....	101
5.5 Структуры	104
5.6 Функциональные блоки	106
5.7 Функции.....	110
5.8 Массивы	113

6 Управление проектами	115
6.1 Создание проекта.....	116
6.2 Открытие проекта	118
6.3 Чтение с контроллера.....	120
6.4 Загрузка в контроллер.....	122
6.5 Работа с архивом проекта.....	128
6.6 Экспорт/импорт структуры проекта.....	131
6.7 Переименование проекта.....	133
6.8 Сохранение и копирование.....	134
6.9 Закрытие и удаление.....	136
7 Управление конфигурацией.....	137
7.1 Параметры подключения к контроллеру	139
7.2 Контроллер	141
7.2.1 Параметры резервирования.....	145
7.2.2 Сетевые настройки	148
7.2.2.1 Параметры удаленного доступа	149
7.2.2.2 Параметры сетевых интерфейсов.....	151
7.2.2.3 Параметры IP-портов.....	153
7.2.3 Параметры режима выполнения.....	154
7.2.4 Аппаратная конфигурация.....	155
7.2.5 Модули ввода-вывода.....	157
7.3 Внешние коммуникации.....	162
7.3.1 Межконтроллерный обмен	163
7.3.2 Задачи связи Modbus	167
7.3.2.1 Назначение modbus-адресов.....	169
7.3.2.2 Экспорт modbus-адресов модулей ввода-вывода	172
7.3.2.3 Настройка Modbus через дерево конфигурации.....	173
7.3.2.3.1 Настройка задач связи Modbus	174
7.3.2.3.2 Настройка Modbus-запросов	179
7.3.2.4 Настройка Modbus через функциональные блоки	183
7.3.3 Задачи связи МЭК-104.....	188
7.3.3.1 Управление группами переменных в МЭК-104	190
7.3.3.2 Управление переменными в МЭК-104.....	192
7.3.4 Задачи связи NTP	194
7.4 Конфигурирование OPC-серверов	196
7.4.1 Управление OPC-серверами	198
7.4.2 Чтение конфигурации OPC-сервера	200
7.4.3 Редактирование базы абонентов	201
7.4.4 Загрузка конфигурации OPC-сервера	205
8 Отладка.....	206
8.1 Сборка проекта	207
8.2 Отладка на контроллере.....	209

8.3	Режим эмуляции.....	212
8.4	Возможности отладки.....	214
8.4.1	Отладка в редакторе словаря.....	215
8.4.2	Отладка в редакторах программных объектов.....	216
8.4.3	Управление циклом и приложением.....	217
9	Режим диагностики.....	219
9.1	Диагностика мастер-модуля.....	221
9.1.1	Системные события и статистика.....	222
9.1.2	Световые индикаторы.....	224
9.1.2.1	Индикаторы режима работы.....	225
9.1.2.2	Индикаторы ошибок мастер-модуля.....	227
9.1.2.3	Индикаторы динамических ошибок.....	229
9.1.2.4	Индикаторы MODE переключателей.....	231
9.1.2.5	Индикаторы дискретных входов/выходов.....	233
9.1.2.6	Аппаратные индикаторы.....	234
9.1.2.7	Индикаторы Ethernet-портов.....	235
9.1.2.8	Индикаторы питания и температуры.....	236
9.1.2.9	Коды запуска и сброса.....	237
9.1.3	Управление временем и приложением.....	238
9.1.4	Экспорт данных.....	239
9.1.5	Обновление ПО.....	240
9.2	Диагностика модулей ввода-вывода.....	243
9.2.1	Проверка каналов ввода-вывода.....	244
9.2.2	Обновление прошивки.....	246
10	Библиотеки.....	248
10.1	Системная библиотека.....	249
10.2	Модули ввода-вывода.....	250
10.2.1	Состав модульных структур.....	251
11	Настройка прав доступа к проекту.....	256
11.1	Изменение режима безопасности.....	258
11.2	Настройка требований к паролям.....	260
11.3	Настройка учетных записей.....	262
11.4	Регистрация событий безопасности.....	265
12	История изменений.....	266

Введение

Среда разработки Unimod PRO 2 предназначена для конфигурирования программируемых промышленных контроллеров серии TREI-5B и разработки приложений управления технологическим процессом. Вторая версия системы разработки является кроссплатформенной и включает в себя набор инструментов для обеспечения процесса подготовки программного кода технологического приложения, описания связей переменных с их физическими представлениями (каналами ввода-вывода), конфигурации и диагностики контроллеров, а также компиляции, загрузки и отладки приложения. Разработанное приложение содержит данные о конфигурации аппаратной части контроллера, распределении памяти, программные инструкции.

Технологическое приложение компилируется в системно-независимый код, который загружается через локальную сеть Ethernet на целевую платформу контроллера TREI-5B, где сохраняется в энергонезависимой памяти и автоматически запускается на исполнение после подачи питания. Запуск и выполнение технологического приложения на контроллере осуществляется под управлением исполнительной системы Unimod PRO 2.

Настоящее руководство содержит полные сведения, необходимые для работы со средой разработки Unimod PRO 2, подключения и конфигурирования контроллеров серии TREI-5B, разработки и отладки технологических приложений на языках стандарта МЭК 61131-3. Предназначено для разработчиков программного обеспечения, а также для специалистов по проектированию, внедрению и эксплуатации систем контроля и управления технологическими процессами, построенных на базе контроллеров TREI-5B.

АО «ТРЭИ» является владельцем авторских прав на Unimod PRO 2 в целом, а также на оригинальные технические решения, с которыми работает данный программный продукт, и на встроенное системное программное обеспечение. Продукция компании АО «ТРЭИ» постоянно совершенствуется и развивается. Изменения в руководство вносятся без дополнительного уведомления пользователей.

Руководство доступно для ознакомления в двух форматах: в виде электронной веб-справки и в формате PDF. При наличии доступа в сеть Интернет рекомендуем воспользоваться [веб-справкой](#). Она всегда соответствует последней версии системы разработки и избавит вас от необходимости отслеживать актуальность версии скачанного документа, что характерно для руководств в PDF формате. Для скачивания актуальной версии руководства в формате PDF перейдите по [ссылке](#).

Содержание документа многократно проверено на наличие ошибок и соответствие последней версии среды разработки Unimod PRO 2, но на практике даже после тщательной проверки некоторые ошибки остаются незамеченными. Если вы обнаружили ошибки в тексте или несоответствие актуальной версии среды

разработки, будем рады получить от вас обратную связь. Просто напишите все выявленные недостатки нам на электронную почту с пометкой «для технического писателя», это поможет в кратчайшие сроки актуализировать информацию.

Авторские права на данный документ принадлежат АО «ТРЭИ». Копирование, воспроизведение и публикация любой части или всего документа не допускается без предварительного письменного разрешения правообладателя.

© АО «ТРЭИ», 1990 – 2024

Россия, 440028, Пенза, ул. Титова, 1Г

Телефон (fax): +7 (8412) 49-95-39, 8 (800) 201-85-39

fax: +7 (8412) 49-85-13

e-mail: tr-penza@trei.biz

сайт: <http://www.trei.biz>

1 Общие сведения

В комплект программного обеспечения Unimod PRO 2 входят:

- среда разработки Unimod PRO 2 - предназначена для конфигурирования программируемых промышленных контроллеров серии TREI-5B, создания приложения для управления технологическими процессами и его загрузки в контроллер, подробно описана в настоящем руководстве;

- исполнительная система Unimod PRO 2 - предназначена для выполнения на контроллере разработанного в среде разработки технологического приложения, осуществляет непосредственное управление технологическими процессами на основании результатов обработки поступающих данных, подробное описание в документе Исполнительная система Unimod PRO 2.

В настоящем разделе сгруппированы основные функциональные возможности среды разработки Unimod PRO 2, ограничения на ее использование и правила совместимости проектов, разработанных в разных версиях среды.

1.1 Функциональные возможности

Среда разработки Unimod PRO 2 включает в себя компоненты для разработки и отладки технологических приложений, настройки конфигурации исполнительной системы и диагностики работы контроллера.

Возможности разработки:

- поддержка языков программирования стандарта МЭК 61131-3: ST - структурированный текст, FBD - функциональные блочные диаграммы, LD - релейно-контактная логика;
- наличие текстового и графического редактора для разработки программ на языках ST и FBD/LD;
- полный набор стандартных операторов, широкий набор алгебраических, тригонометрических и сдвиговых функций;
- наличие интерактивных редакторов для описания переменных, возможность их переноса между проектами;
- структурирование данных технологического приложения;
- наличие встроенной системной библиотеки алгоритмов управления и регулирования;
- встроенная библиотека со структурным описанием параметров модулей ввода-вывода;
- проверка синтаксиса, компиляция, загрузка и выгрузка технологического приложения.

Возможности отладки:

- просмотр состояния переменных во время выполнения контроллером прикладной задачи;
- выполнение программы на контроллере по шагам, один шаг равен одному циклу контроллера;
- изменение значений переменных в процессе отладки;
- изменение продолжительности цикла приложения в процессе работы отладчика;
- эмуляция выполнения технологического приложения (при отсутствии возможности подключения к реальному контроллеру).

Конфигурация исполнительной системы позволяет настраивать:

- параметры подключения и отладки;
- сетевые параметры и резервирование;
- параметры режима выполнения и аппаратной конфигурации;
- привязки модулей ввода-вывода;
- параметры обмена данными со сторонними устройствами;
- обмен данными с OPC-серверами;
- защищенные каналы связи, аутентификацию и разграничение доступа.

Диагностические возможности:

- поиск доступных в сети контроллеров;
- мониторинг работы контроллера;
- просмотр калибровочных параметров каналов ввода-вывода;
- обновление программного обеспечения;
- управление временем и циклом выполнения;
- сбор статистики о работе системы;
- экспорт статистических данных.

1.2 Системные требования

Минимальные требования к аппаратному и программному обеспечению для корректной работы среды разработки Unimod PRO 2:

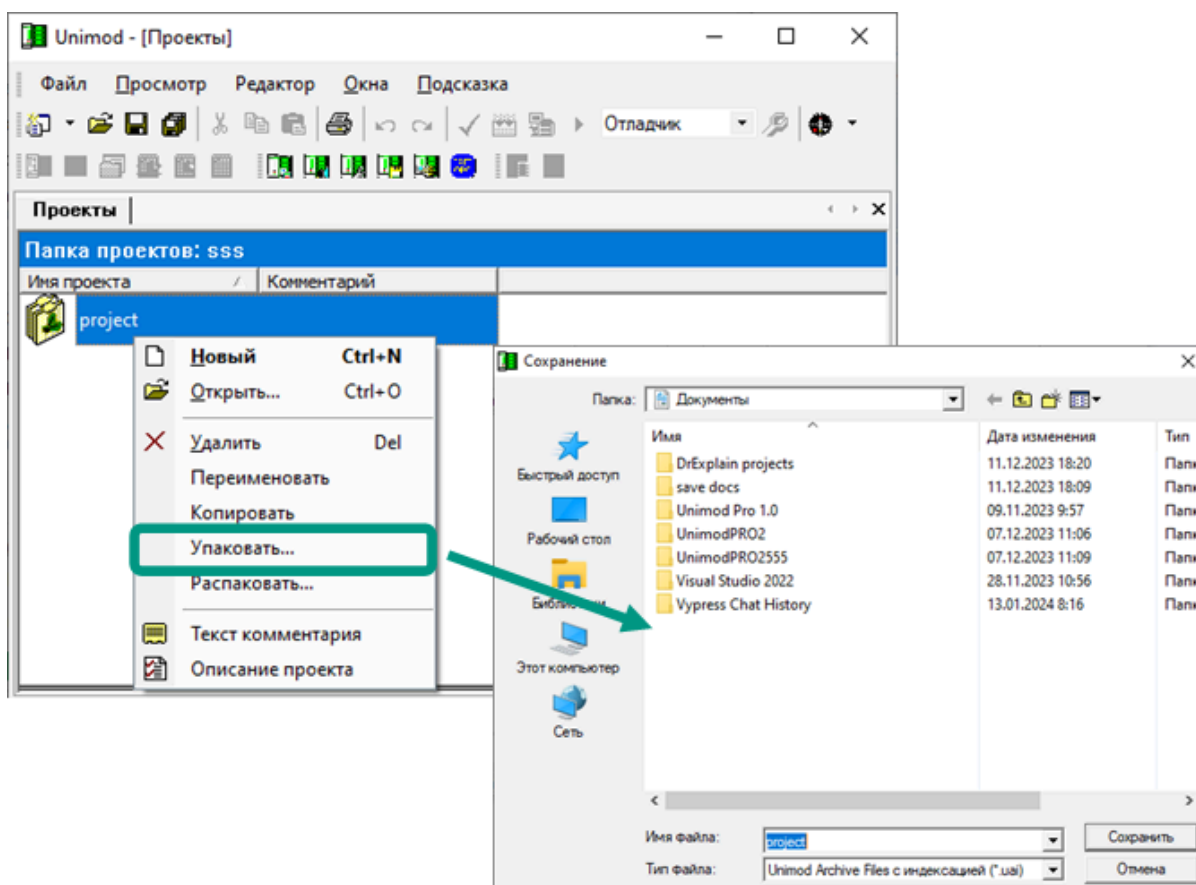
Процессор	Intel Atom Dual Core 1,6 ГГц или аналогичный
Оперативная память (ОЗУ)	не менее 2 Гбайт
Жесткий диск	не менее 1 Гбайта свободного пространства
Видеосистема	с поддержкой разрешения не ниже 1024x768
Сетевой адаптер	со скоростью обмена 10/100 Мбит/сек
Периферийные устройства	монитор, компьютерная мышь
Операционная система	Microsoft Windows 10, 11 Astra Linux в редакции Special Edition версии 1.7

1.3 Совместимость проектов

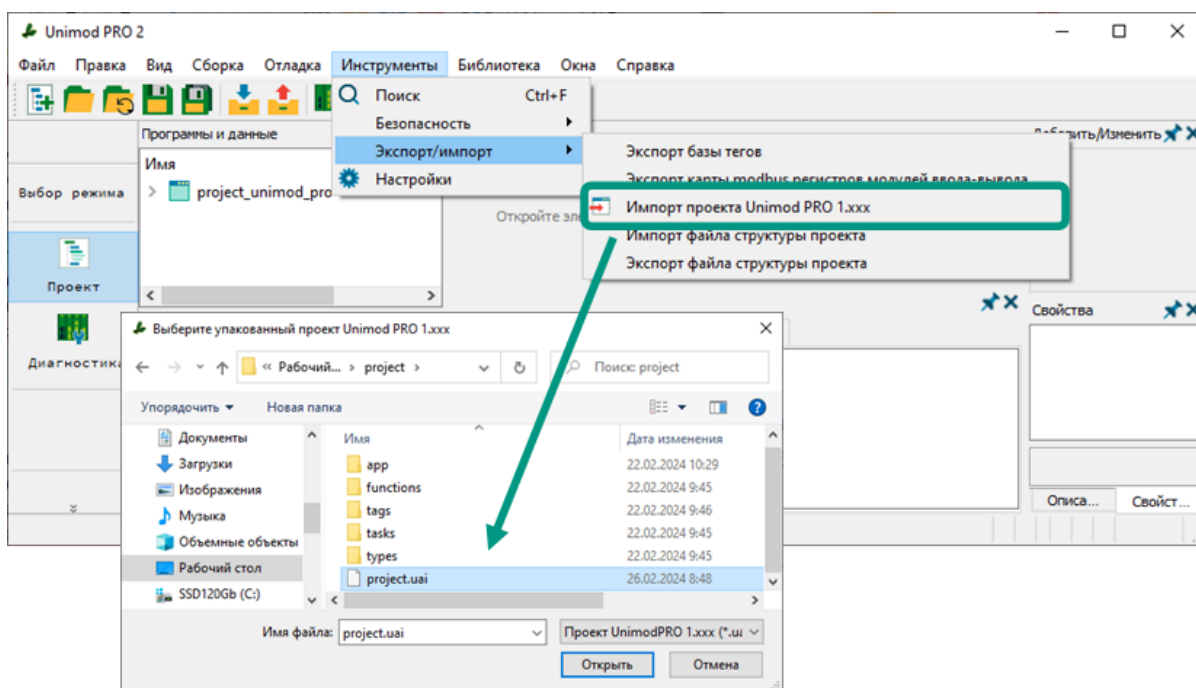
Среда разработки Unimod PRO 2 поддерживает обратную совместимость разработанных в ней проектов. Любой проект, разработанный в устаревшей версии среды разработки, может быть открыт в новой версии.

Проекты, разработанные в первой версии среды разработки Unimod PRO, могут быть конвертированы и открыты во второй версии. Для этого необходимо в первой версии среды разработки упаковать проект в формате «*.uai», после чего распаковать его во второй версии. Перенос элементов будет осуществлен в автоматическом режиме встроенным в среду разработки Unimod PRO 2 конвертором.

Для переноса проекта откройте первую версию среды разработки Unimod PRO → при закрытом проекте кликните правой кнопкой мыши по его названию и из контекстного меню выберите пункт «Упаковать...» → в открывшемся окне выберите каталог для сохранения проекта в формате «*.uai» и введите его название → нажмите кнопку «Сохранить». При этом в выбранном каталоге будет создан подкаталог с указанным именем, содержащий все файлы проекта.



На следующем шаге в среде разработки Unimod PRO 2 создайте новый проект → выберите пункт меню «Инструменты → Экспорт/импорт → Импорт проекта Unimod PRO 1.xxx» → в открывшемся окне выберите из сохраненного ранее каталога файл с расширением «*.uai» → нажмите кнопку «Открыть».



При конвертации проекта могут возникнуть ошибки. Это связано с полной или частичной переработкой некоторых элементов проекта во второй версии среды разработки. Исправление возникших при конвертации ошибок осуществляется вручную. Даже при успешной конвертации рекомендуется полностью проверить проект.

1.4 Доступные для программирования контроллеры

Разработанные в среде Unimod PRO 2 технологические приложения могут быть запущены только на контроллерах фирмы TREI, благодаря наличию в каждом выпускаемом контроллере предустановленной исполнительной системы Unimod PRO 2. Без исполнительной системы приложение не может быть выполнено.

В настоящее время для программирования доступны модули серии TREI-5B:

Название	Тип модуля	Краткое описание
M101E	Мастер-модуль	Используется в составе контроллера TREI-5B-05 серии ECO-2, без резервирования
M102ER	Мастер-модуль	Используется в составе контроллера TREI-5B-05 серии ECO-2, с возможностью резервирования
M501E	Мастер-модуль	Используется в составе контроллеров TREI-5B-05 серий STANDARD и SMART-TP
M1201E	Мастер-модуль	Используется в составе контроллера TREI-5B-04 серии STANDARD
F1001E	Мастер-модуль	Входит в серию модулей ППКУП TREI (прибор приемно-контрольный и управления пожарный TREI).

2 Подготовка к работе

Во время подготовки к работе с системой разработки Unimod PRO 2 необходимо сначала убедиться, что соблюдены минимальные системные требования, скачать установочный файл с сайта компании АО "ТРЭИ" и по возможности полностью ознакомиться с настоящим руководством. Установочный файл должен соответствовать операционной системе, в которой будет осуществляться разработка. Требования к операционной системе указаны в разделе [Системные требования](#).

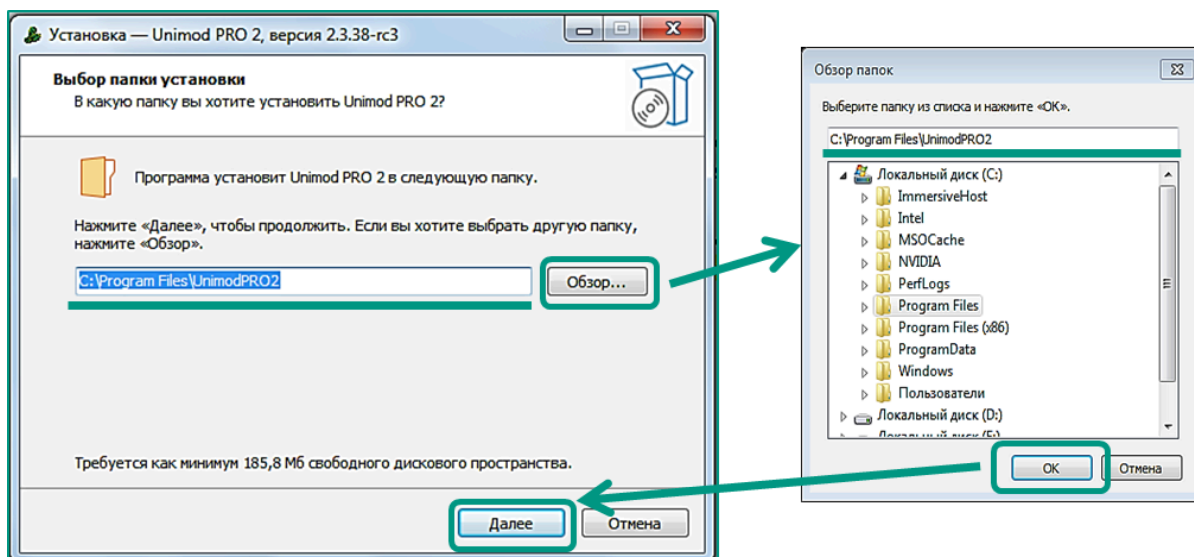
После этого можно приступать к установке и настройке системы разработки. Указанные процессы подробно описаны в настоящем разделе.

2.1 Установка среды разработки

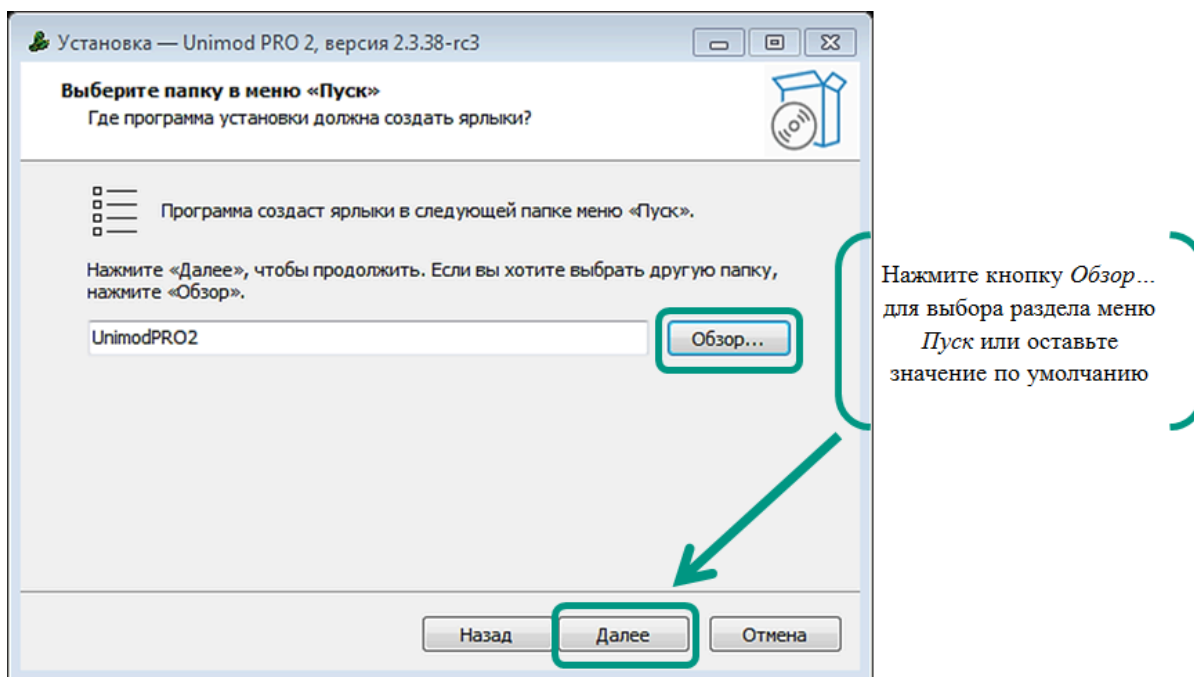
Установочный файл среды разработки Unimod PRO 2 можно скачать на сайте www.trei.biz в разделе «Программное обеспечение». Предварительно требуется пройти [регистрацию](#) или [авторизацию](#). Для скачивания доступны два варианта установочного файла в зависимости от вашей операционной системы: для Windows с расширением «*.exe» и для Astra Linux с расширением «*.deb».

2.1.1 Установка в Windows

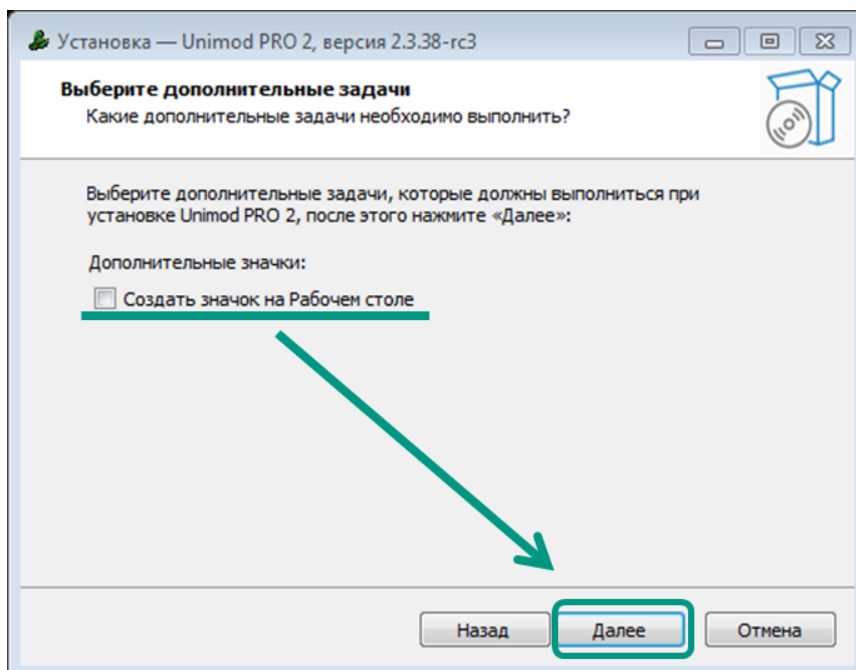
Для установки среды разработки Unimod PRO 2 в ОС Windows запустите установочный файл с расширением «*.exe» → в открывшемся диалоговом окне нажмите кнопку «Обзор...», выберите каталог для установки и нажмите кнопку «Ок» → для продолжения установки нажмите кнопку «Далее». По умолчанию среда разработки устанавливается в каталог «C:\Program Files\UnimodPRO2».



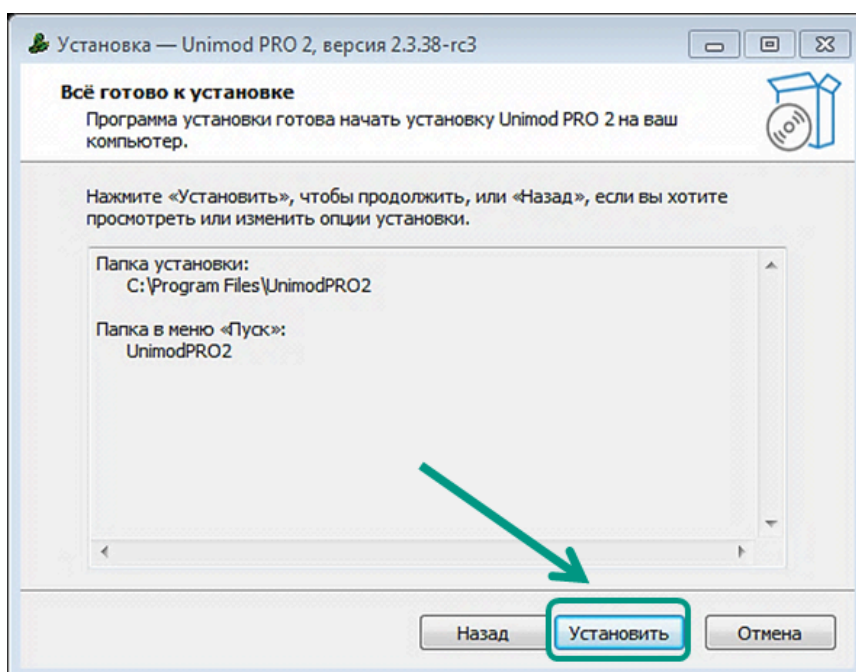
На следующем этапе нажмите на кнопку «Обзор» → в открывшемся диалоговом окне выберите раздел меню «Пуск», где будет создан ярлык программы, либо оставьте значение по умолчанию → нажмите кнопку «Далее».



На следующем этапе при необходимости выберите значение «Создать значок на Рабочем столе» и нажмите кнопку «Далее».

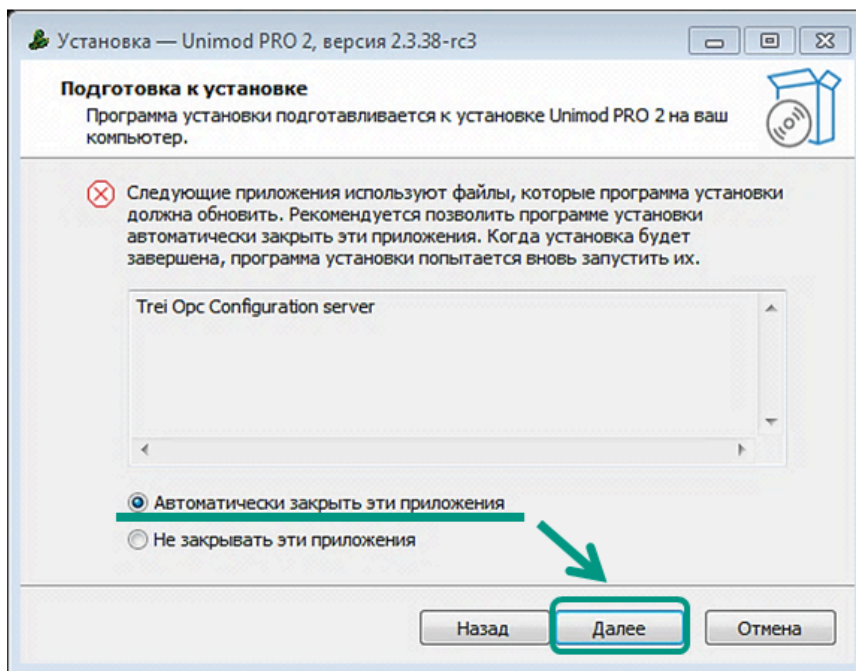


После этого убедитесь, что параметры установки выбраны верно. Для изменения параметров нажмите кнопку «Назад» и внесите необходимые изменения в соответствии с вышеуказанной инструкцией. Для запуска процесса установки нажмите кнопку «Установить».

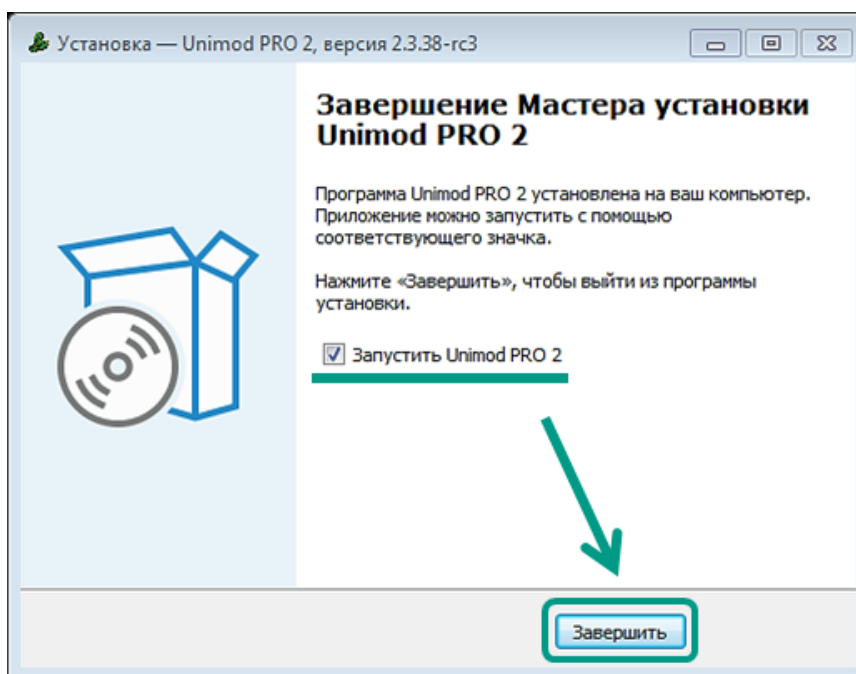


Если система разработки Unimod PRO 2 ранее была удалена с ПК не полностью, при ее повторной установке на ПК может быть выведено следующее окно. Для

корректной установки системы оставьте выбранным пункт «Автоматически закрыть эти приложения» и нажмите на кнопку «Далее».



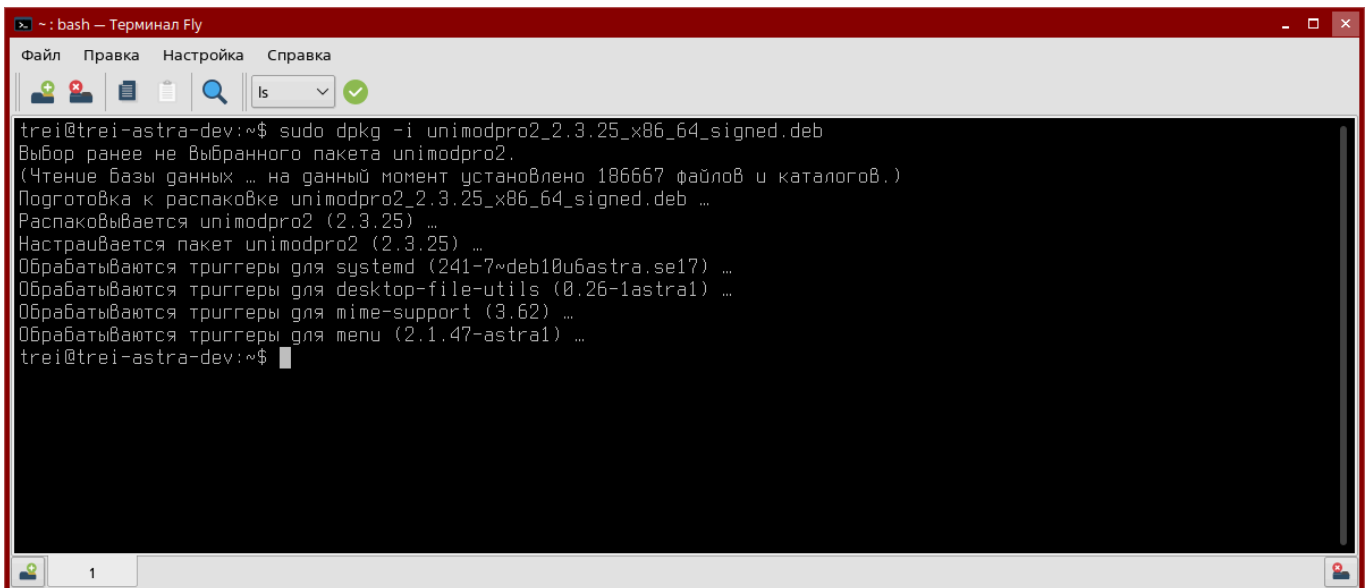
На завершающем этапе уберите галочку с раздела «Запустить Unimod PRO 2», если после установки не требуется запускать систему разработки. Для завершения установки нажмите «Завершить» и перезагрузите операционную систему.



После этого, система разработки Unimod PRO 2 установлена на ПК в указанный при установке каталог, а в меню «Пуск» станет доступен для запуска ярлык программы.

2.1.2 Установка в Astra Linux

Для установки среды разработки Unimod PRO 2 в ОС Astra Linux запустите приложение «Терминал Fly». Для этого выберите пункт графического меню «Пуск → Системные → Терминал Fly» или нажмите сочетание клавиш «Alt + T». В окне терминала через пробелы введите команды «`sudo dpkg -i`» и имя установочного файла с расширением «`.deb`», например, «`sudo dpkg -i unimodpro2_2.3.25_x86_64_signed.deb`». Для начала установки нажмите клавишу «Enter».



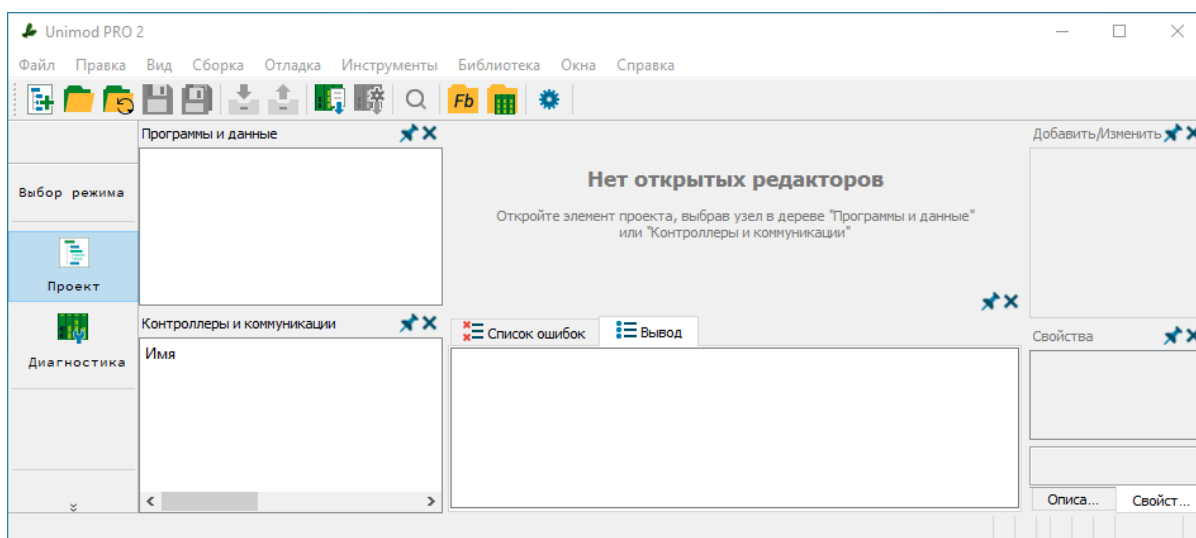
```
trei@trei-astra-dev:~$ sudo dpkg -i unimodpro2_2.3.25_x86_64_signed.deb
Выбор ранее не выбранного пакета unimodpro2.
(Чтение базы данных ... на данный момент установлено 186667 файлов и каталогов.)
Подготовка к распаковке unimodpro2_2.3.25_x86_64_signed.deb ...
Распаковывается unimodpro2 (2.3.25) ...
Настраивается пакет unimodpro2 (2.3.25) ...
Обрабатываются триггеры для systemd (241-7~deb10u6astra.se17) ...
Обрабатываются триггеры для desktop-file-utils (0.26-1astra1) ...
Обрабатываются триггеры для mime-support (3.62) ...
Обрабатываются триггеры для menu (2.1.47-astra1) ...
trei@trei-astra-dev:~$
```

После чего среда разработки Unimod PRO 2 будет установлена в каталог по умолчанию «`/opt/TREI/UnimodPRO2`». При этом проекты могут быть созданы в любом каталоге файловой системы.

2.2 Запуск среды разработки

Варианты запуска среды разработки Unimod PRO 2:

- с помощью ярлыка программы на Рабочем столе (если ярлык был создан при установке);
- с помощью ярлыка программы в меню «Пуск»;
- с помощью исполняемого файла «Unimod.exe» из каталога программы (только для ОС Windows).
- в терминале с помощью команды `«/opt/TREI/UnimodPRO2/run_unimod»` (только для ОС Astra Linux).



Общий вид среды разработки Unimod PRO 2 после запуска.

2.3 Настройка среды разработки

В среде разработки Unimod PRO 2 расположение панелей в рабочем пространстве и их размеры, а также параметры конфигуратора, компилятора и отладчика настраиваются индивидуально для каждого пользователя в соответствии с его предпочтениями.

Все изменения в настройках, размерах панелей и их расположении в рабочем пространстве сохраняются средой разработки и воспроизводятся при каждом последующем запуске.

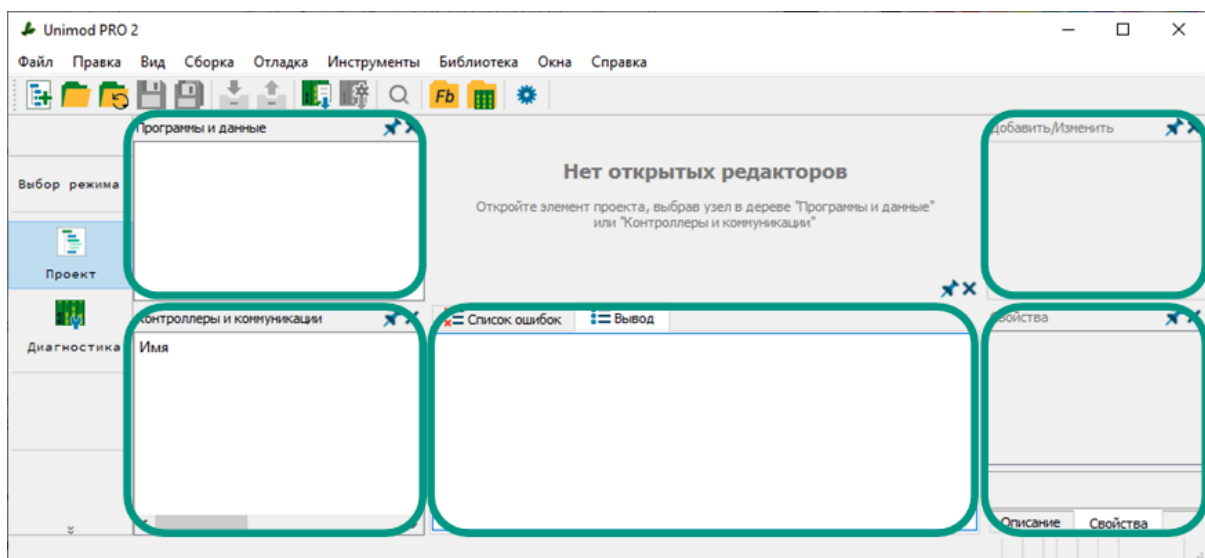
2.3.1 Настройка рабочего пространства

В рабочем пространстве Unimod PRO 2 могут располагаться панели:

- «Программы и данные»;
- «Контроллеры и коммуникации»;
- «Вывод и список ошибок»;
- «Свойства»;
- «Описание»;
- «Добавить/Изменить».

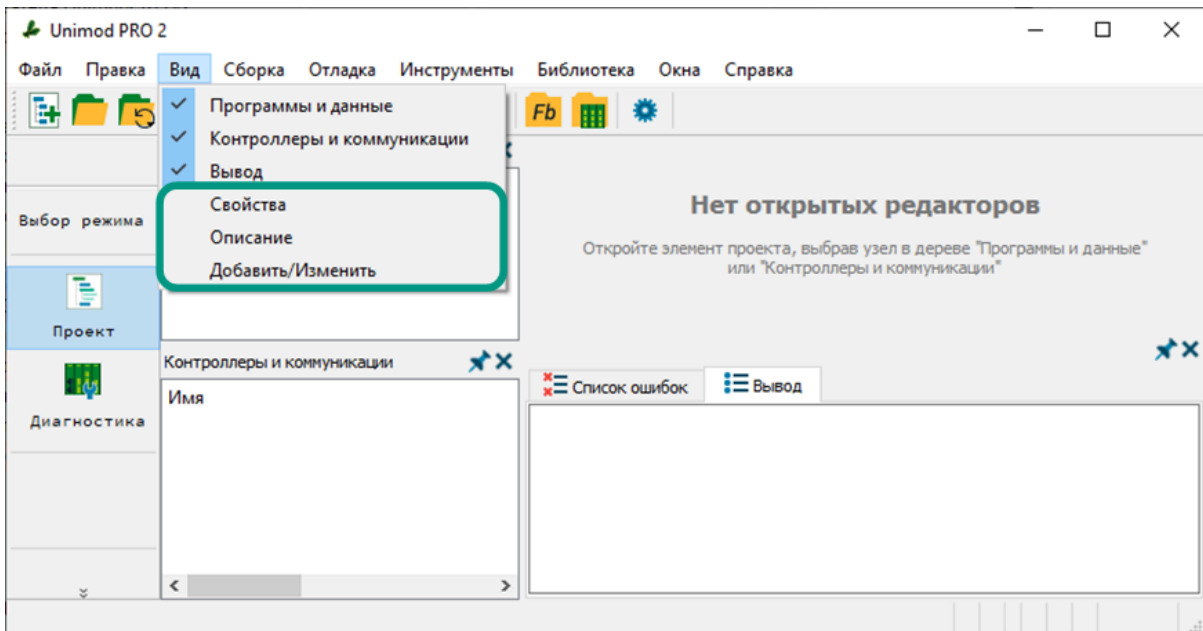
Подробнее с назначением каждой панели можно ознакомиться в разделе [Пользовательский интерфейс](#).

По умолчанию все панели находятся в активном состоянии и отображаются в рабочем пространстве среды разработки (на следующем рисунке панели выделены контуром).



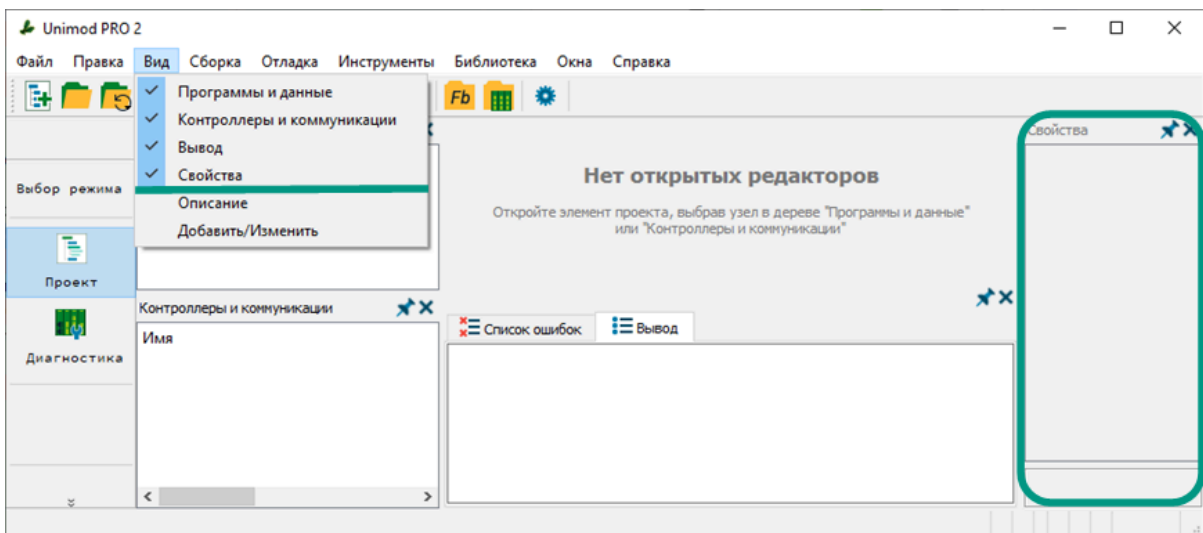
Для удаления панели из рабочего пространства нажмите кнопку **X** в правом верхнем углу панели или снимите галочку с соответствующей панели в разделе меню «Вид». После чего панель будет удалена из рабочего пространства.

На следующем изображении приведен пример с удалением панелей «Свойства», «Описание» и «Добавить/Изменить» из рабочего пространства, при этом в разделе меню «Вид» слева от неактивных панелей отсутствуют галочки.



Для добавления неактивных панелей в рабочее пространство выберите в разделе меню «Вид» требуемую панель, после чего она станет активной и будет добавлена в рабочее пространство.

На следующем изображении приведен пример с выбором в разделе меню «Вид» неактивной панели «Свойства», при этом выбранная панель добавлена в рабочее пространство среды разработки, а в разделе меню «Вид» слева от названия панели появляется галочка.

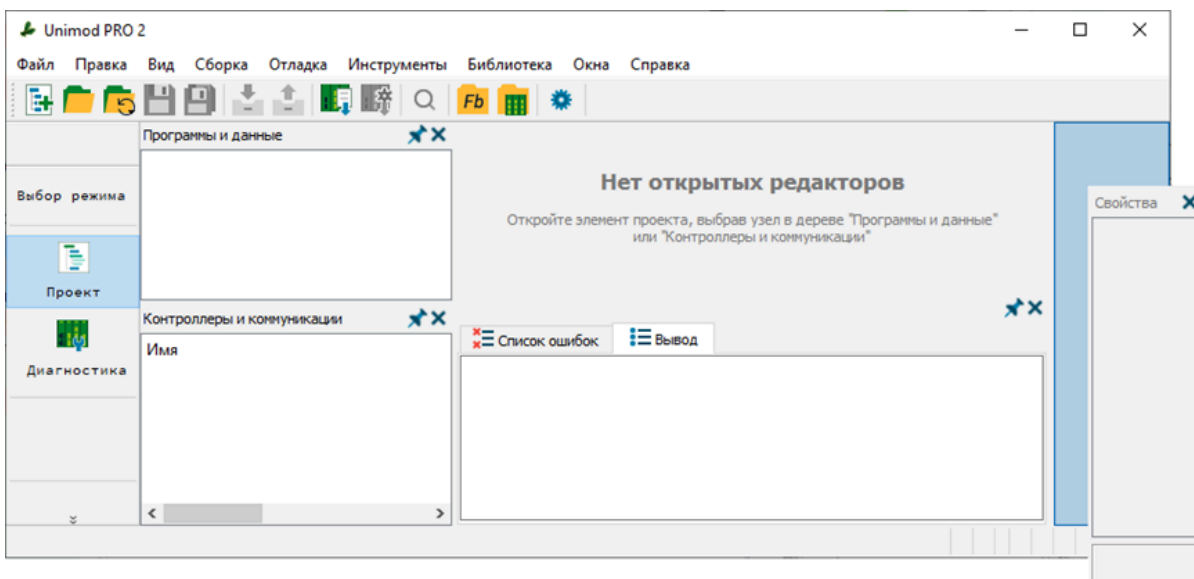


В рабочем пространстве кроме добавления и удаления панелей можно настраивать их расположение.

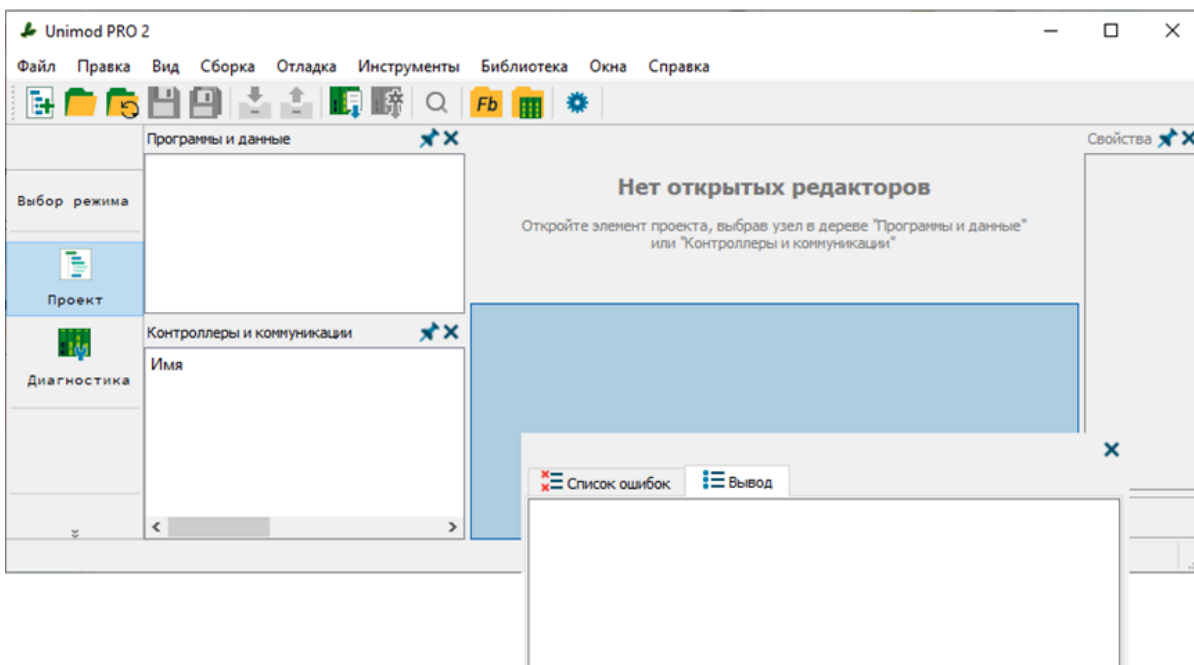
Для перемещения панели по рабочему пространству наведите курсор на верхнюю часть перемещаемой панели (где выведено название) → нажмите левую кнопку мыши и переместите панель в требуемое место рабочего пространства. При перемещении панели к одной из границ рабочего пространства среда разработки

автоматически будет подсвечивать цветом предлагаемую для закрепления область. Вдоль верхней границы панели не закрепляются.

На следующем изображении приведен пример с перемещением панели к правой границе рабочего пространства, цветом выделяется предлагаемая для закрепления область.



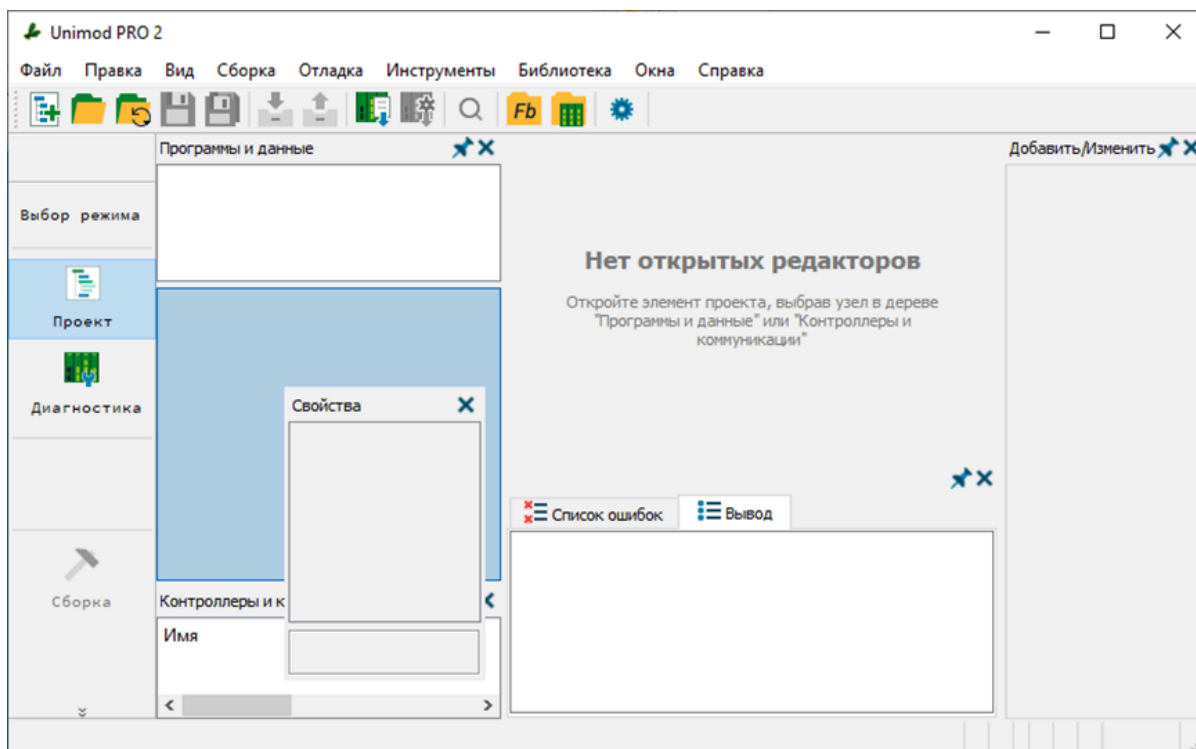
На следующем изображении приведен пример с перемещением панели к нижней границе рабочего пространства, цветом выделяется предлагаемая для закрепления область.





Вдоль одной из границ рабочего пространства можно закрепить несколько панелей, для этого перетащите панель к требуемой границе между уже


зафиксированными панелями → автоматически будет выделена область для закрепления.

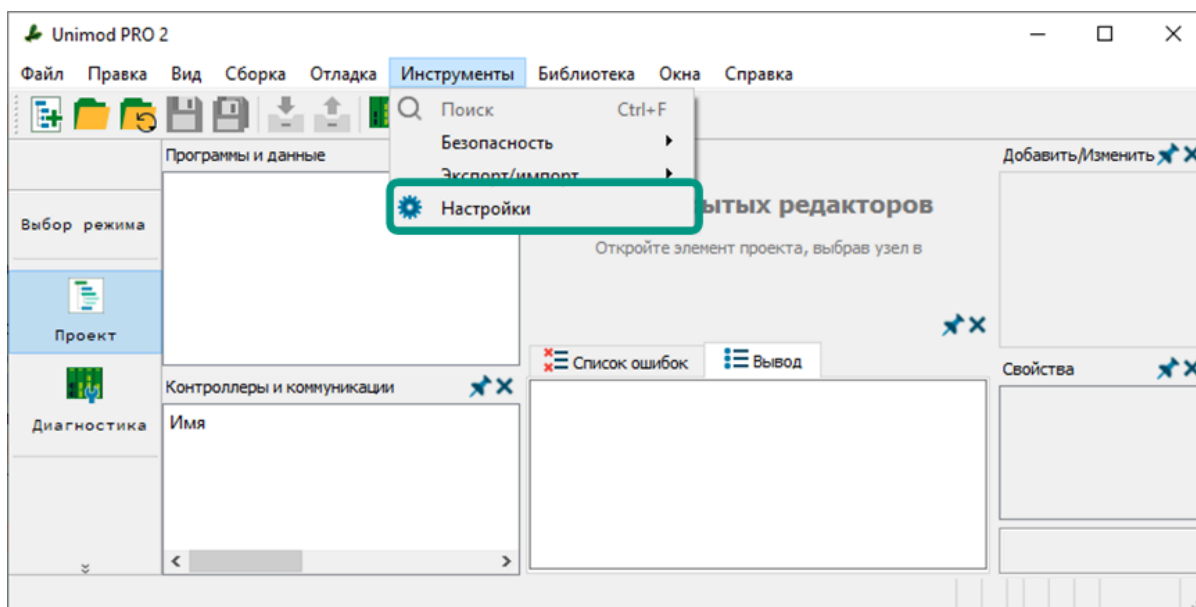
На следующем изображении приведен пример с перемещением панели к левой границе рабочего пространства между панелями «Программы и данные» и «Контроллеры и коммуникации», цветом выделяется предлагаемая для закрепления область.



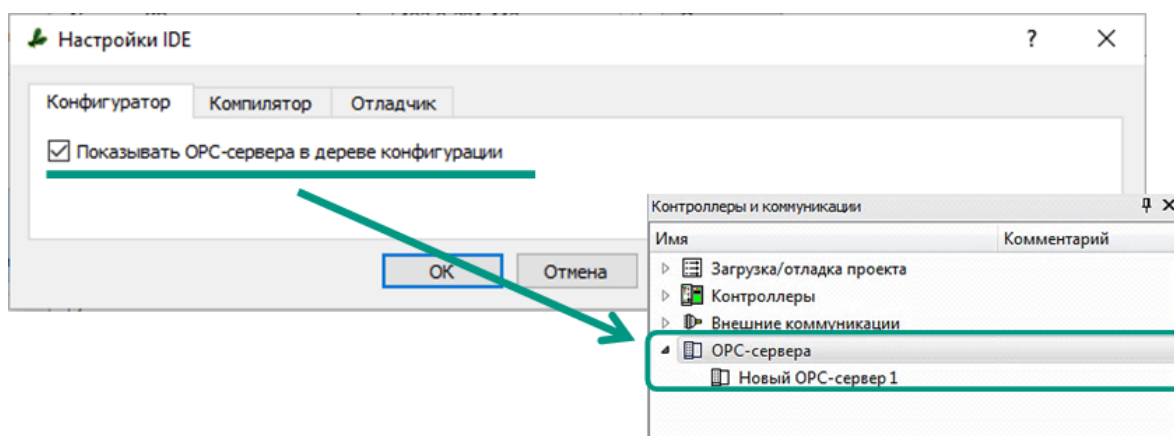
Для изменения размеров панелей наведите курсор на границу панели → вместо курсора должен появиться символ  или  → зажмите левую кнопку и перемещением мыши измените размер панели по горизонтали или вертикали.

2.3.2 Настройка среды разработки

Для настройки среды разработки Unimod PRO 2 выберите пункт меню «Инструменты → Настройки» или нажмите кнопку  на панели быстрого доступа. После чего в открывшемся окне будут доступны вкладки с настраиваемыми параметрами.



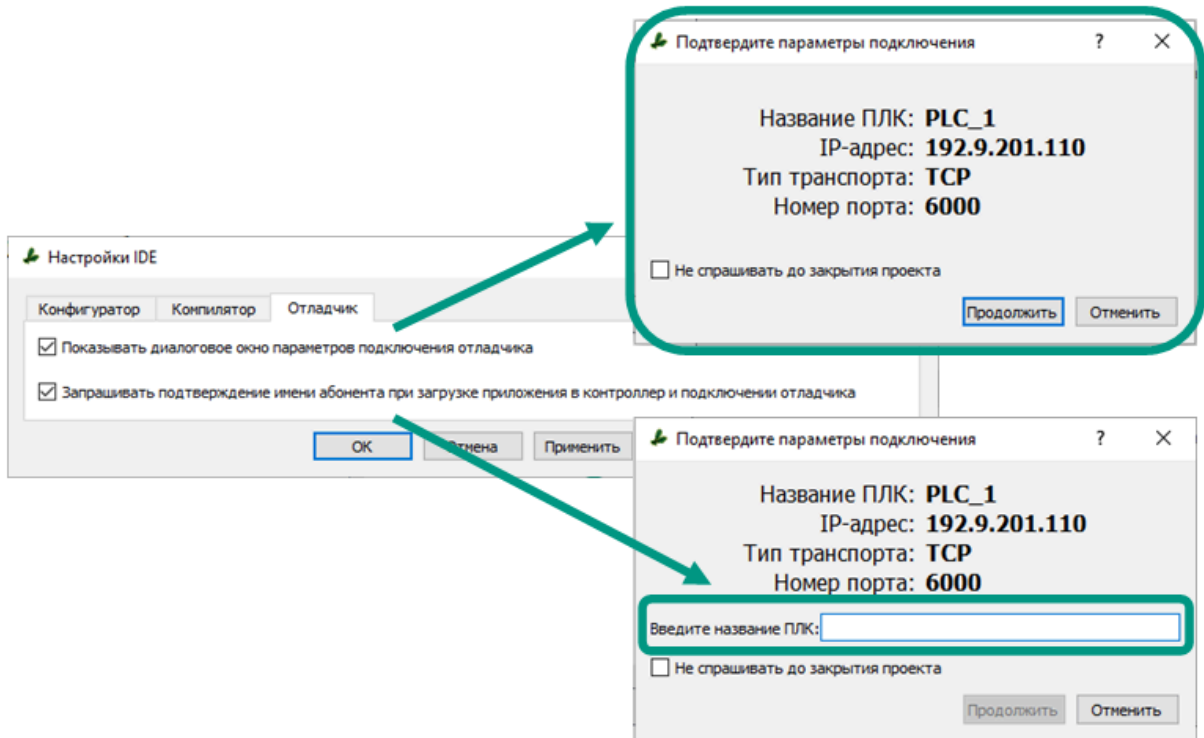
На вкладке «Конфигуратор» выбор пункта «Показывать OPC-сервера в дереве конфигурации» позволяет отображать в дереве конфигурации элемент для конфигурирования OPC-серверов.




На вкладке «Отладчик» доступны для выбора параметры:

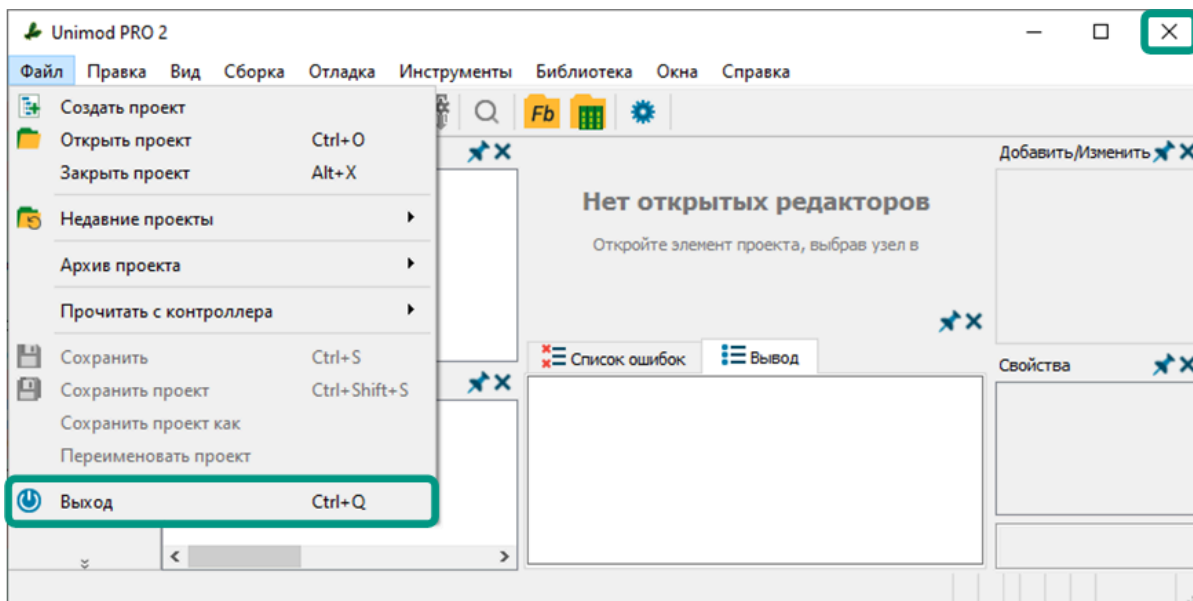
- **показывать диалоговое окно параметров подключения отладчика** - при выборе данного пункта перед подключением к контроллеру в режимах «Отладка» и «Загрузка» дополнительно будет выводиться окно для подтверждения параметров подключения;

- *запрашивать подтверждение имени абонента при загрузке приложения в контроллер и подключении отладчика* - при выборе данного пункта перед подключением к контроллеру в режимах «Отладка» и «Загрузка» в окне с параметрами подключения потребуется ввод названия целевого контроллера. Ввод названия контроллера выступает дополнительным этапом подтверждения. Акцент внимания на имени контроллера позволяет пользователю убедиться, что по указанному IP-адресу действительно расположен требуемый ему контроллер.

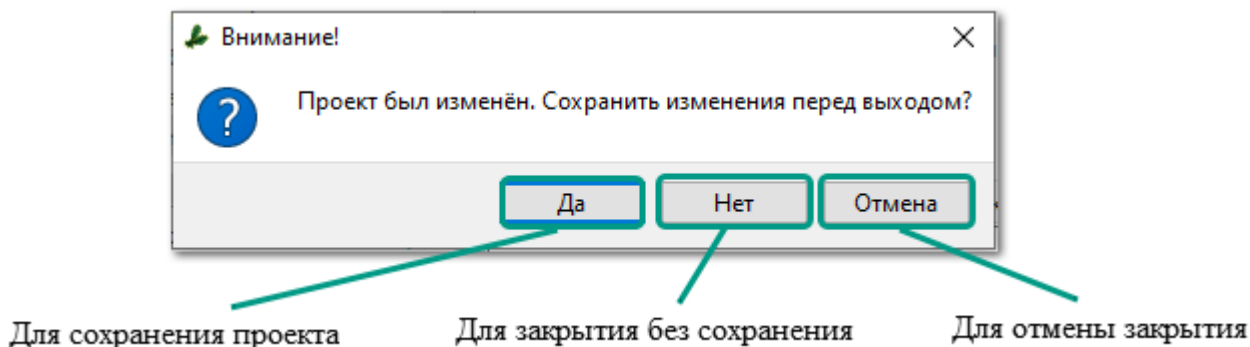


2.4 Закрытие среды разработки

Для закрытия Unimod PRO 2 нажмите кнопку  в правом верхнем углу программы либо выберите пункт меню «Файл → Выход».



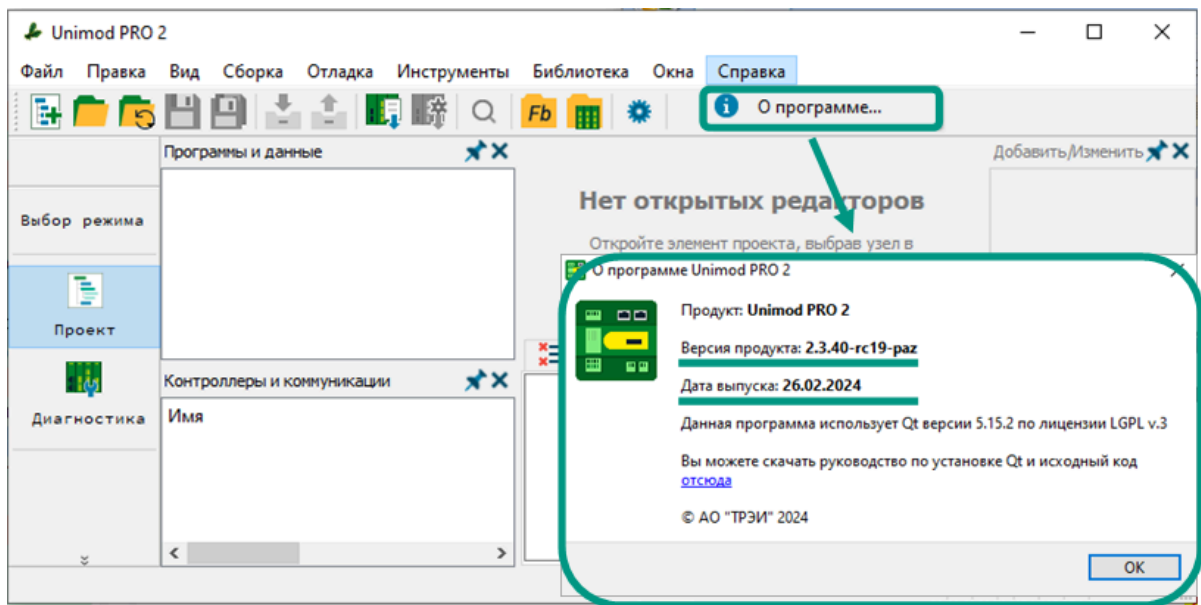
При наличии не сохраненных изменений в проекте перед закрытием среда разработки Unimod PRO 2 выведет уведомление и предложит варианты дальнейших действий.



2.5 Обновление среды разработки

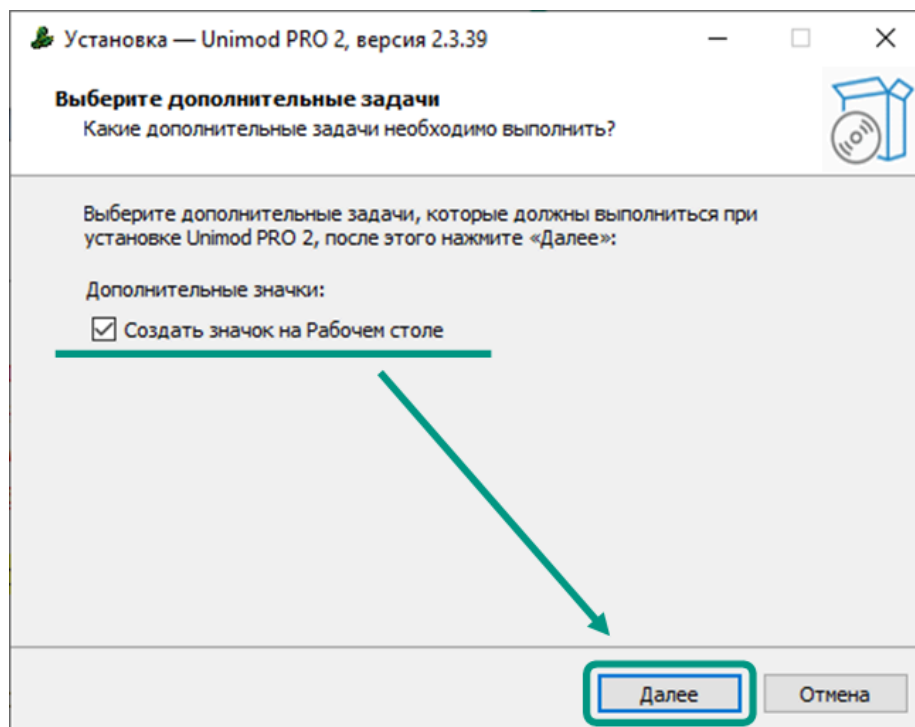
Среда разработки Unimod PRO 2 постоянно совершенствуется и развивается. В связи с этим совершенствование среды разработки может осуществляться без дополнительного уведомления пользователей. Все изменения фиксируются в настоящем руководстве в разделе [История изменений](#).

Для вывода сведений об используемой версии выберите раздел меню «Справка → О программе...».

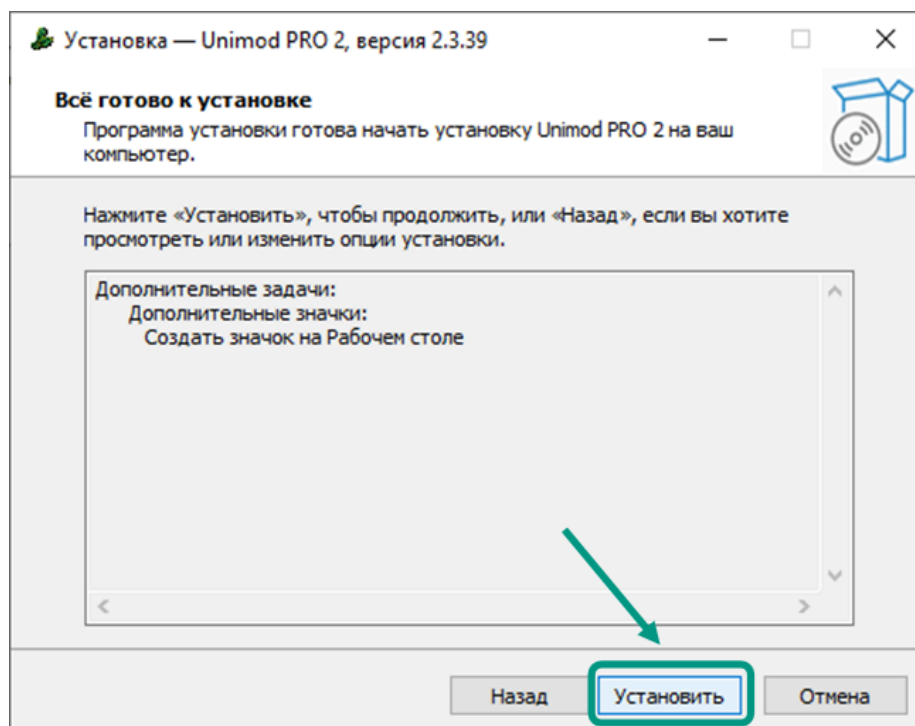


Узнать последнюю версию и скачать установочный файл для обновления среды разработки Unimod PRO 2 можно на сайте www.trei.biz в разделе «Программное обеспечение». Для скачивания доступны два варианта установочного файла в зависимости от вашей операционной системы: для Windows и для Astra Linux. Предварительно требуется пройти [регистрацию](#) или [авторизацию](#).

Для обновления среды разработки Unimod PRO 2 запустите скачанный установочный файл (в Windows с расширением «*.exe», в Astra Linux с расширением «*.deb») → в открывшемся окне при необходимости выберите значение «Создать значок на Рабочем столе» → нажмите кнопку «Далее».

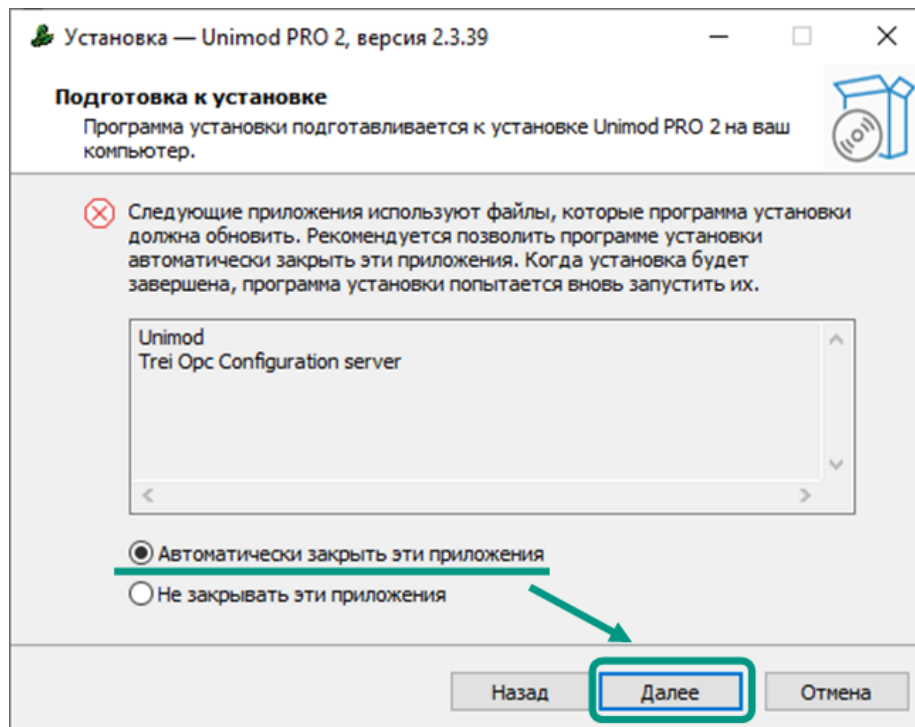


На следующем этапе для запуска процесса обновления нажмите кнопку «Установить».

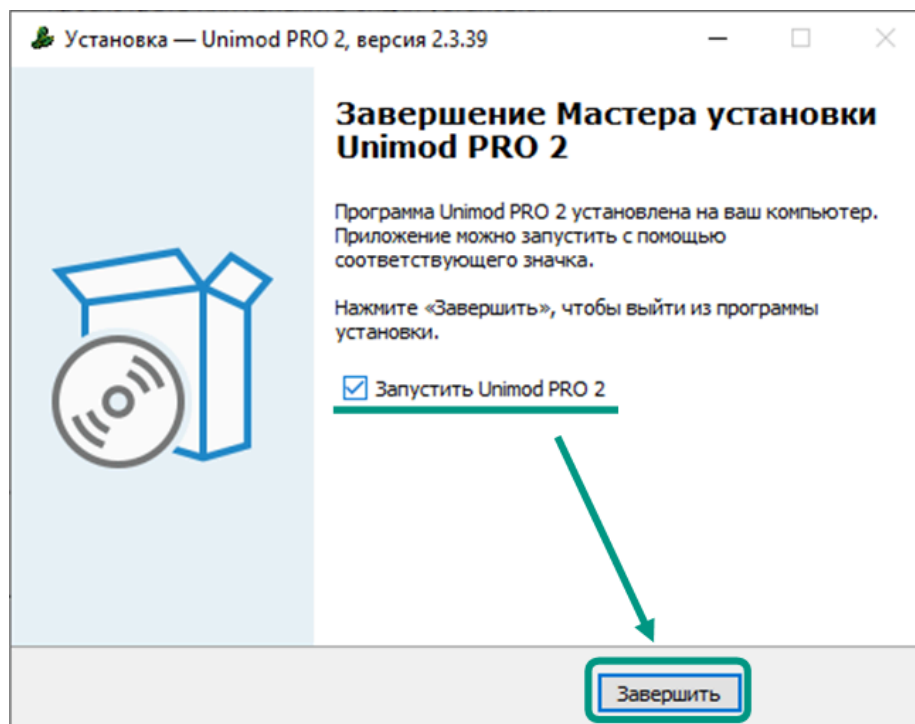


В процессе обновления программа сама определяет каталог, в котором установлена среда разработки Unimod PRO 2. Если файлы из каталога среды разработки используются каким-либо приложением ПК, во время обновления может быть выведено нижеуказанное окно. Для корректного обновления оставьте

выбранным пункт «Автоматически закрыть эти приложения» и нажмите на кнопку «Далее».



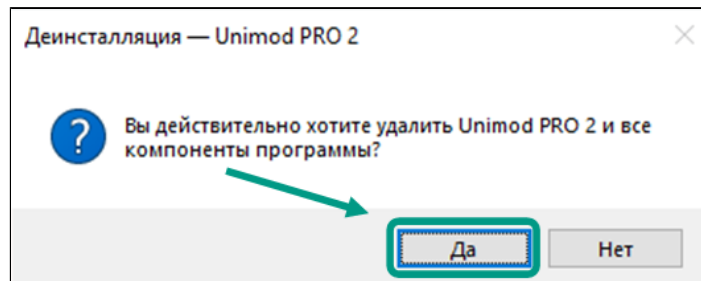
На завершающем этапе уберите галочку с раздела «Запустить Unimod PRO 2», если после установки не требуется запускать среду разработки. Для завершения установки нажмите кнопку «Завершить» и перезагрузите операционную систему.



После этого, среда разработки Unimod PRO 2 будет обновлена и готова к работе.

2.6 Удаление среды разработки

Для удаления среды разработки Unimod PRO 2 в ОС Windows запустите исполняемый файл «*unins000.exe*» из каталога программы или воспользуйтесь встроенными средствами операционной системы, после чего в открывшемся диалоговом окне подтвердите удаление нажатием кнопки «*Да*».




Для удаления среды разработки в ОС Astra Linux запустите приложение «*Терминал Fly*». Для этого выберите пункт графического меню «*Пуск → Системные → Терминал Fly*» или нажмите сочетание клавиш «*Alt + T*». В окне терминала через пробелы введите команды «*sudo dpkg -r unimodpro2*» и нажмите клавишу «*Enter*».

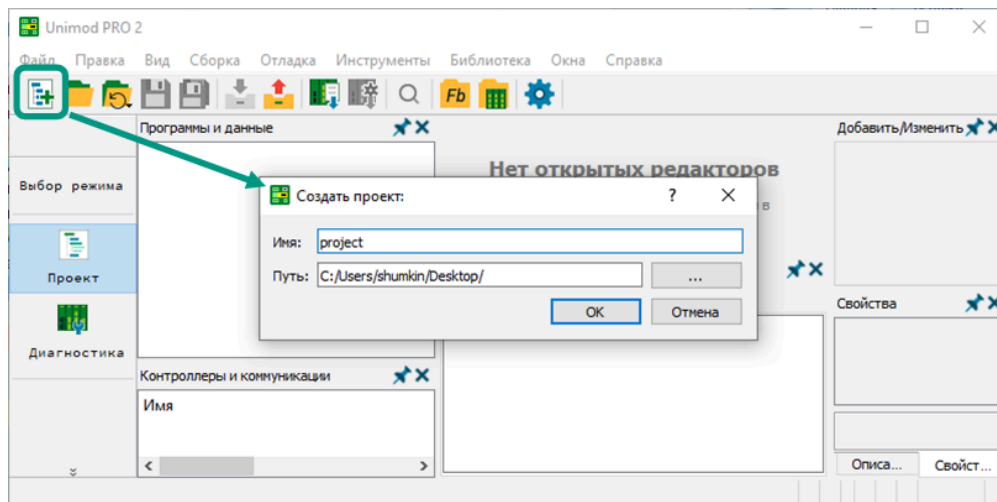
```
trei@trei-astra-dev:~$ sudo dpkg -r unimodpro2
(Чтение базы данных ... на данный момент установлено 186955 файлов и каталогов.)
Удалается unimodpro2 (2.3.25) ...
dpkg: предупреждение: при удалении unimodpro2 каталог «/opt/TREI/OPCUA» не пуст, поэтому не удалён
dpkg: предупреждение: при удалении unimodpro2 каталог «/etc/TREI/OPCUA/uaroot/CertificateStore» не пуст, поэтому не
удалён
Обработываются триггеры для desktop-file-utils (0.26-1astra1) ...
Обработываются триггеры для mime-support (3.62) ...
Обработываются триггеры для systemd (241-7~deb10u6astra.se17) ...
trei@trei-astra-dev:~$
```



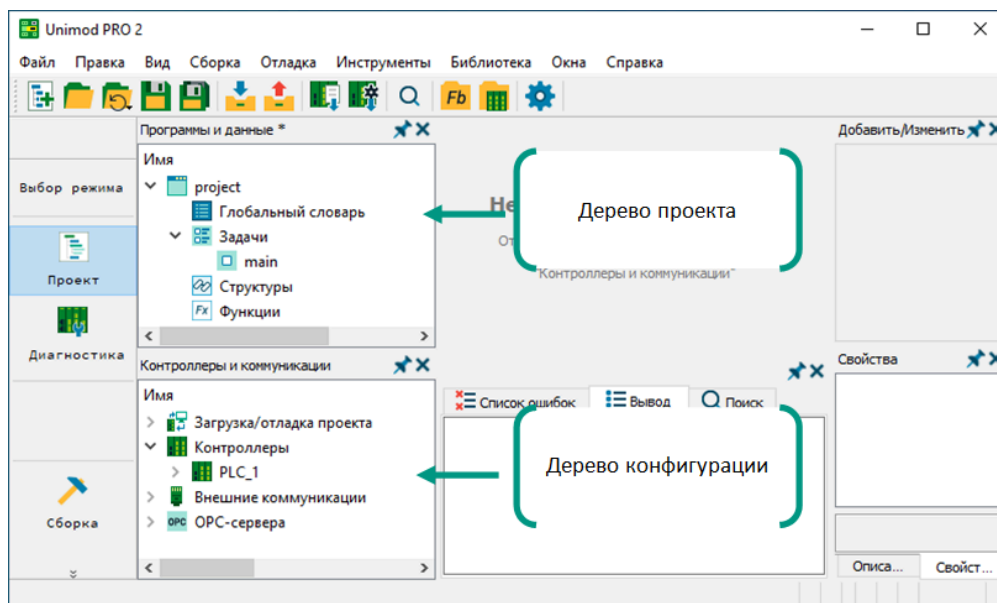
При удалении среды разработки Unimod PRO 2, разработанные в ней проекты не удаляются из файловой системы компьютера. Для продолжения работы с проектами нужно перенести их на персональный компьютер с установленной средой разработки Unimod PRO 2 или осуществить ее повторную установку на текущем устройстве.

3 Краткий обзор

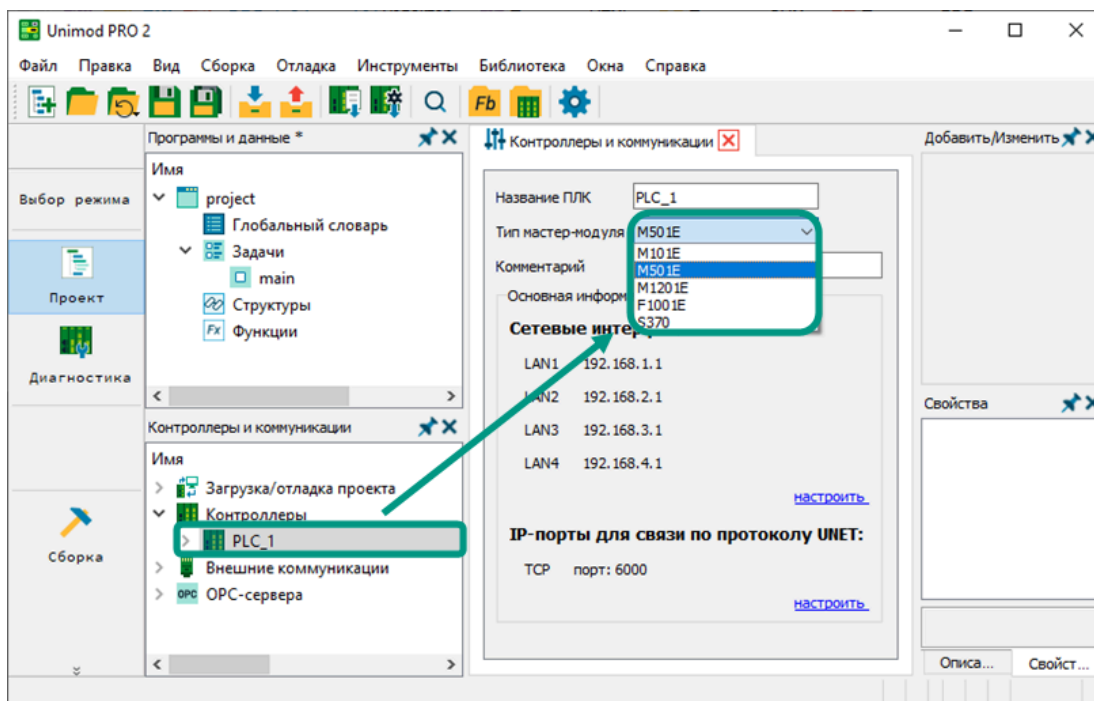
1. **Создание проекта.** Для создания проекта на панели быстрого доступа нажмите кнопку  или выберите пункт меню «Файл → Создать проект». В открывшемся окне выберите каталог для сохранения проекта и введите его имя. Имя проекта может содержать латинские буквы, цифры и символы подчеркивания, начинаться с буквы и не превышать 128 символов.



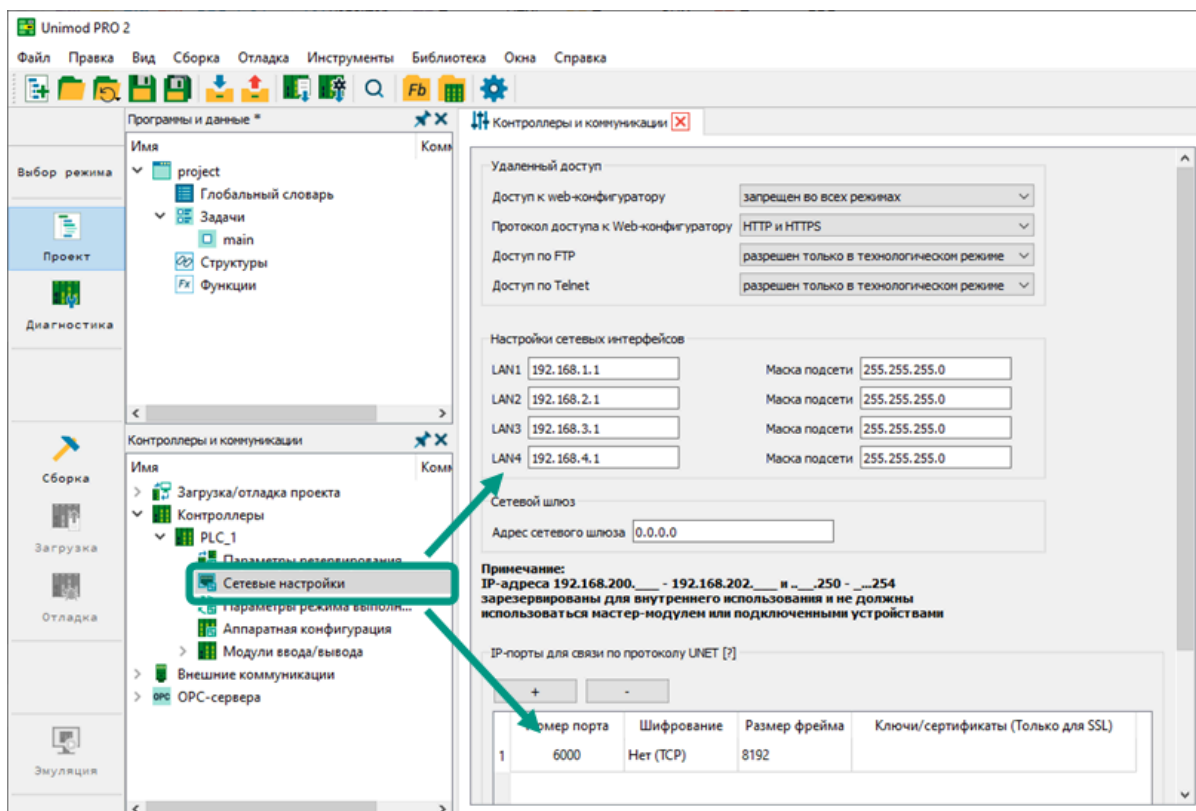
После создания проекта будут доступны для редактирования дерево проекта на панели «Программы и данные» и дерево конфигурации на панели «Контроллеры и коммуникации». В дереве проекта по умолчанию добавлена главная задача «main», а также доступны для добавления и редактирования программы, дочерние программы, функции, структуры с функциональными блоками и словари переменных. В дереве конфигурации по умолчанию добавлен контроллер с именем «PLC_1».



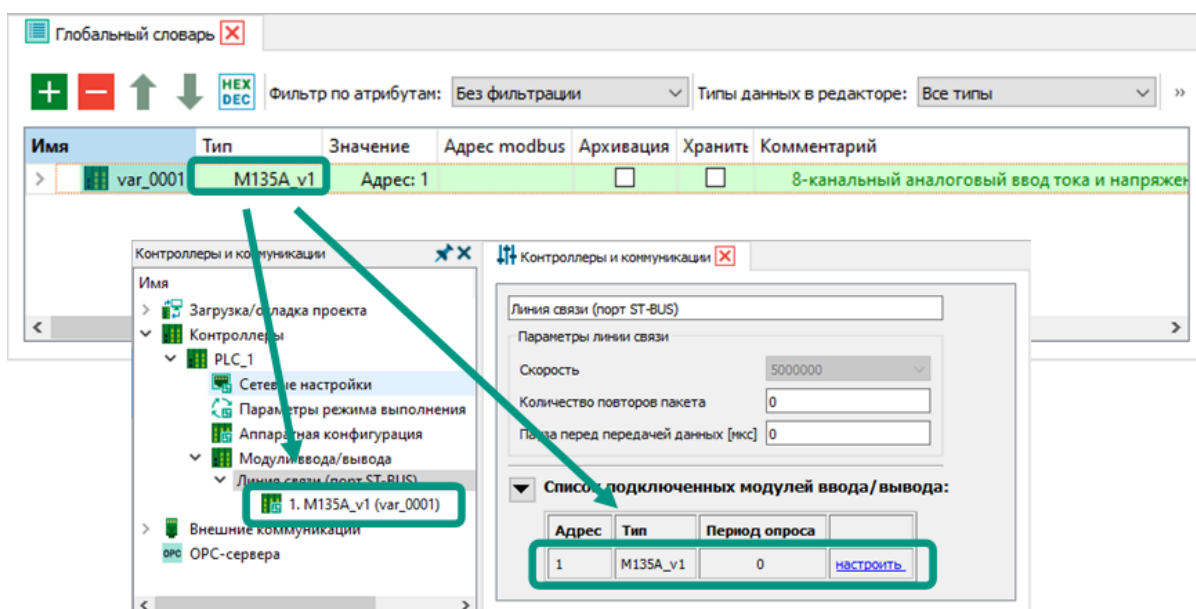
2. **Выбор типа мастер-модуля.** По умолчанию в проекте выбран мастер-модуль M501E. Для изменения типа мастер-модуля в дереве конфигурации выберите раздел «Контроллеры → PLC_1», на открывшейся форме настройки параметров выберите тип мастер-модуля и подтвердите свой выбор нажатием кнопки «Продолжить» в диалоговом окне.



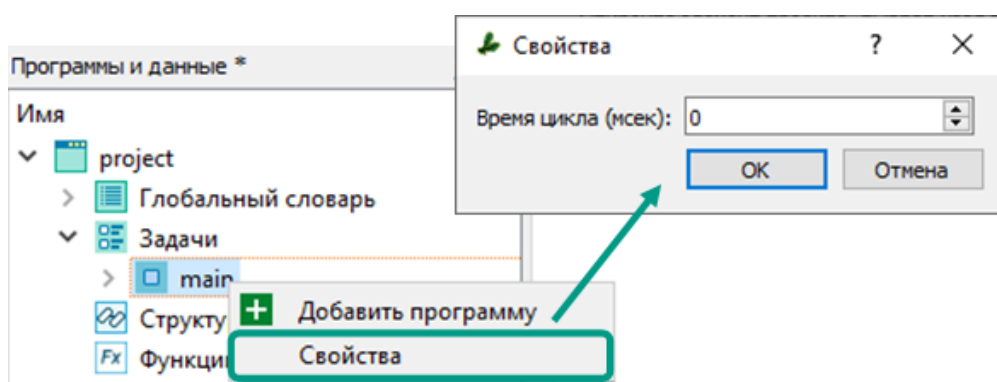
3. Настройка сетевых интерфейсов. Для сетевого доступа к контроллеру требуется настроить параметры сетевых интерфейсов. Для этого в дереве конфигурации перейдите в раздел «Контроллеры → PLC_1 → Сетевые настройки», на открывшейся форме настройки параметров введите минимум один IP-адрес (по умолчанию - 192.168.1.1), маску подсети (по умолчанию - 255.255.255.0) и при необходимости отредактируйте номер порта (по умолчанию - 6000). Указанные параметры будут использоваться при подключении к контроллеру после загрузки в него технологического приложения.



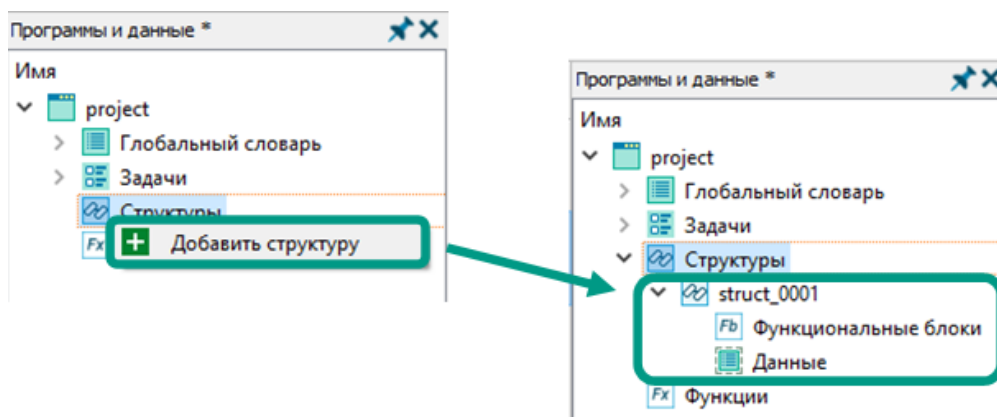
4. **Добавление модулей ввода-вывода.** Для добавления модуля ввода-вывода в глобальный словарь добавьте переменную, в качестве типа данных выберите структуру, соответствующую модулю ввода-вывода (названия структур и модулей совпадают). После чего модуль ввода-вывода будет добавлен в дерево конфигурации в раздел «Контроллеры → PLC_1 → Модули ввода/вывода → Линия связи/порт (ST-BUS)».





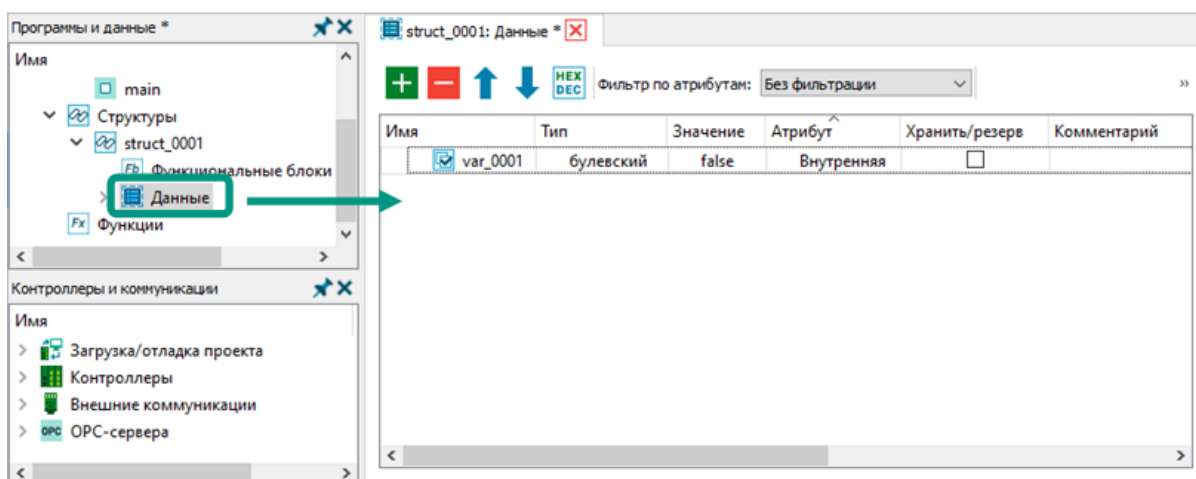
5. **Установка времени цикла.** Для установки времени цикла нажмите правой кнопкой мыши по задаче «main», из контекстного меню выберите пункт «Свойства» и в открывшемся окне введите требуемое время цикла в миллисекундах. При времени цикла равном 0 задача будет выполняться за минимально возможный цикл. При значении времени цикла >0 задача будет выполняться за указанный цикл, при условии, что он не меньше максимально возможного цикла.



6. Добавление структуры. Для добавления структуры нажмите правой кнопкой мыши по разделу «Структуры» и из контекстного меню выберите пункт «Добавить структуру». После чего в дерево проекта будет добавлена структура с двумя вложенными элементами: разделом функциональных блоков и словарем данных структуры. Для изменения имени структуры используется горячая клавиша «F2» или контекстное меню «Переименовать».



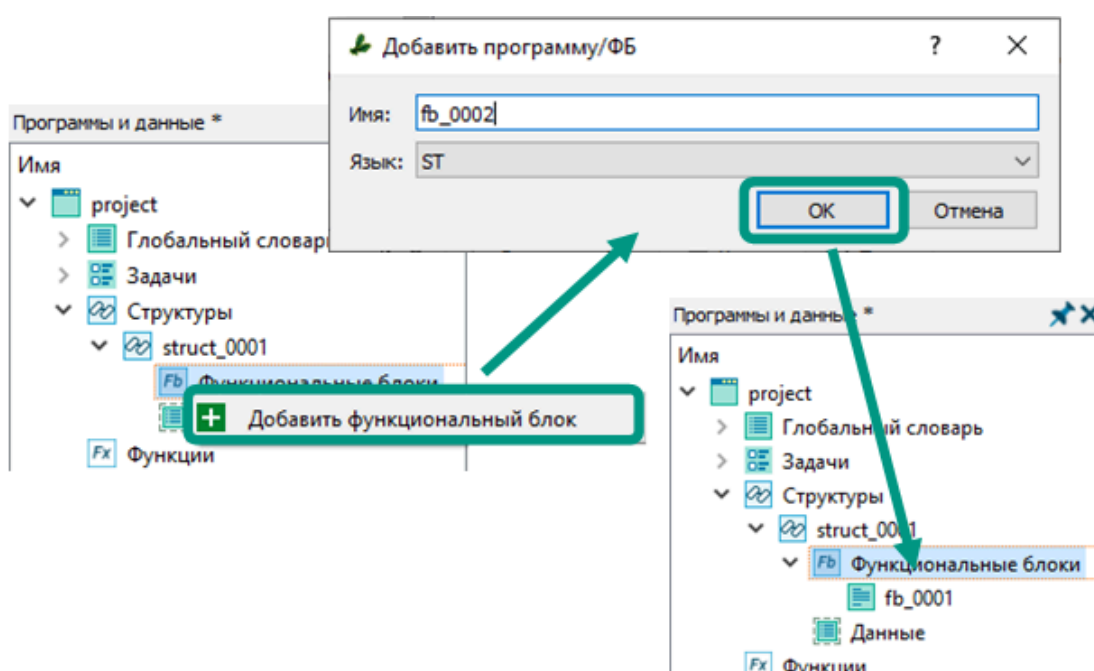
В словаре данных структуры определяется состав входящих в нее переменных. В состав структуры могут входить переменные простых типов данных, массивы, другие структуры, в том числе из системной библиотеки, без ограничений на уровень вложенности. Для открытия словаря структуры дважды кликните по нему левой кнопкой мыши или из контекстного меню выберите пункт «Открыть». Для добавления или удаления переменных используйте кнопки  и .



7. Добавление функционального блока структуры. К каждой структуре могут быть добавлены функциональные блоки – программы на языках ST и FBD/LD, для которых переменные из словаря данных структуры выступают в качестве локальных, но также доступны глобальные переменные. В случае совпадения имен приоритет имеют локальные переменные. Для добавления функционального блока кликните правой кнопкой мыши по соответствующему разделу, из контекстного меню выберите пункт «Добавить функциональный блок», в открывшемся окне введите имя блока и выберите язык программирования: ST или FBD/LD. В рамках одной структуры имена функциональных блоков должны быть уникальными.

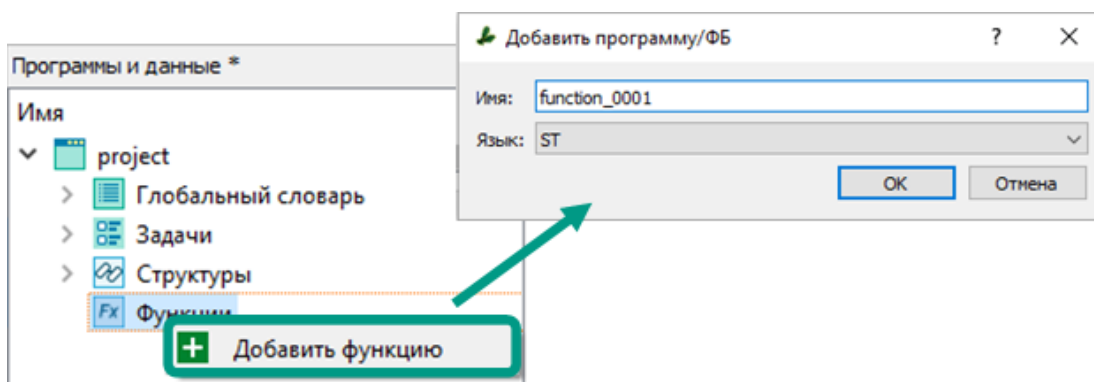
Функциональные блоки могут быть вызваны любыми программами и дочерними программами, а также функциональными блоками других структур. Для вызова функционального блока предварительно требуется в локальном словаре или словаре данных структуры объявить переменную, а в качестве типа данных указать структуру, функциональный блок которой будет использоваться. Блоки могут возвращать несколько выходных значений по нескольким входным значениям, но не могут вызывать другой функциональный блок в рамках одной структуры.

Для открытия редактора функционального блока в дереве проекта дважды щелкните по нему левой кнопкой мыши или из контекстного меню выберите пункт «Открыть».



8. Добавление функции. Для добавления функции в дереве проекта кликните правой кнопкой мыши по разделу «*Функции*», из контекстного меню выберите пункт «*Добавить функцию*» → в открывшемся окне введите имя функции, выберите язык программирования и нажмите кнопку «*Ок*». После чего в дерево проекта будет добавлена функция с вложенным элементом - локальным словарем. Для изменения имени структуры используется горячая клавиша «*F2*» или контекстное меню «*Переименовать*».

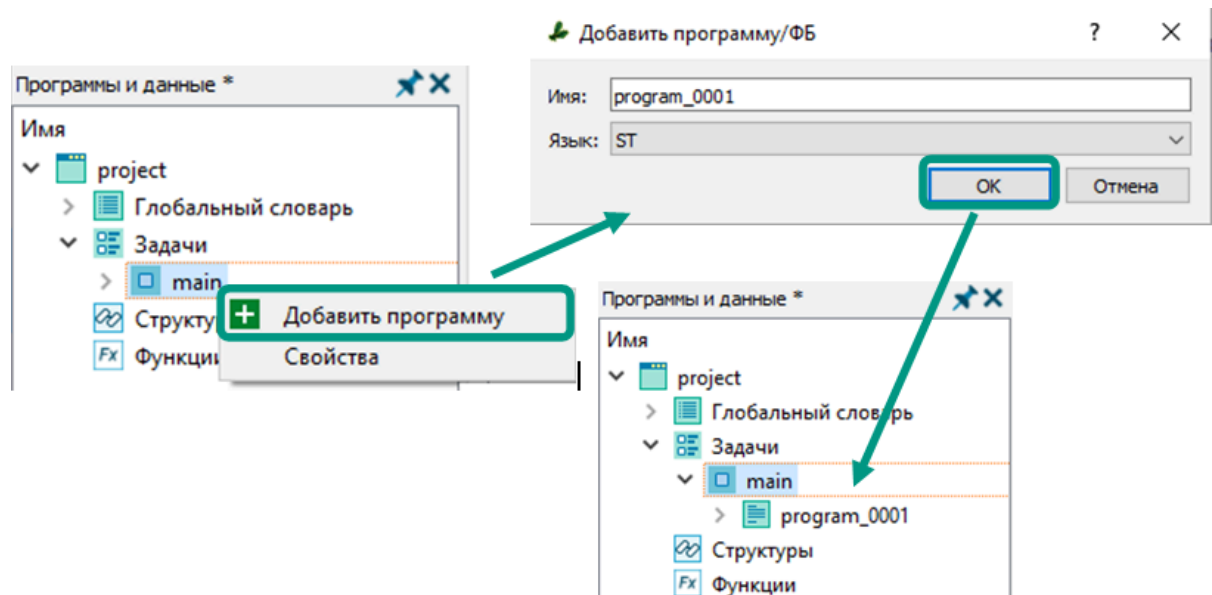
Функции могут быть напрямую вызваны в любых программах, дочерних программах и функциональных блоках, а также в других функциях, при этом функции всегда возвращают одно значение по нескольким входным. В локальном словаре функции могут быть объявлены только переменные простых типов данных.



9. Добавление к задаче программы. Создание структур, функциональных блоков и функций само по себе не приводит к изменению технологической программы. Для вызова функциональных блоков и функций необходимо к задаче добавить программу. В программе вызов функции осуществляется напрямую через указание ее имени, для вызова доступны пользовательские и системные функции. Для вызова функционального блока предварительно требуется в локальном словаре программы объявить переменную, а в качестве типа данных указать структуру, блок которой будет вызываться в программе.

Для добавления новой программы кликните правой кнопкой мыши по задаче, из контекстного меню выберите пункт «*Добавить программу*», в открывшемся окне введите имя программы и выберите язык программирования: ST или FBD/LD. В рамках одной задачи имена программ должны быть уникальными. Выполнение программ происходит в порядке их следования в дереве проекта. Порядок выполнения программ может быть изменен выполнением команд контекстного меню «*Переместить вверх*» или «*Переместить вниз*». Для изменения имени структуры используется горячая клавиша «*F2*» или контекстное меню «*Переименовать*».

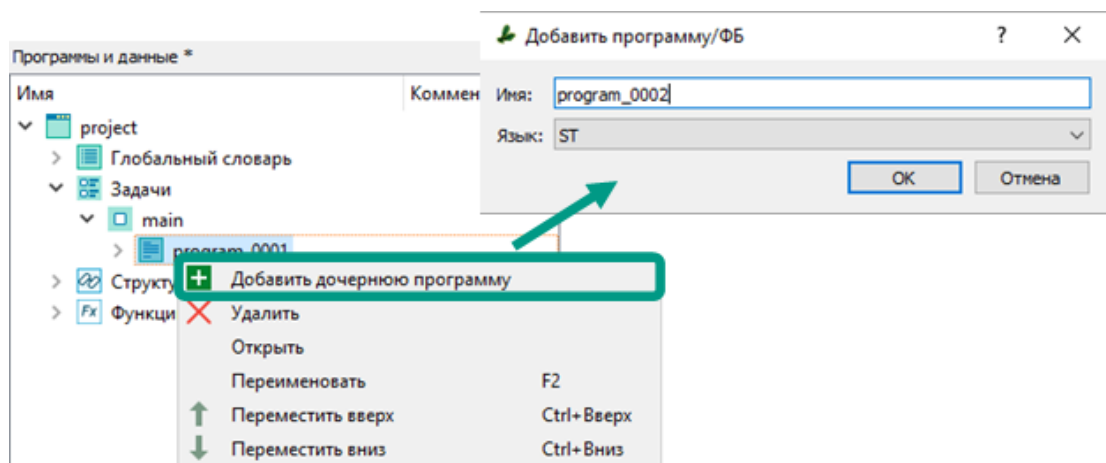
Открытие редактора программы осуществляется двойным кликом левой кнопки мыши по имени программы в дереве проекта, либо через контекстное меню «*Открыть*».



10. Добавление дочерней программы.

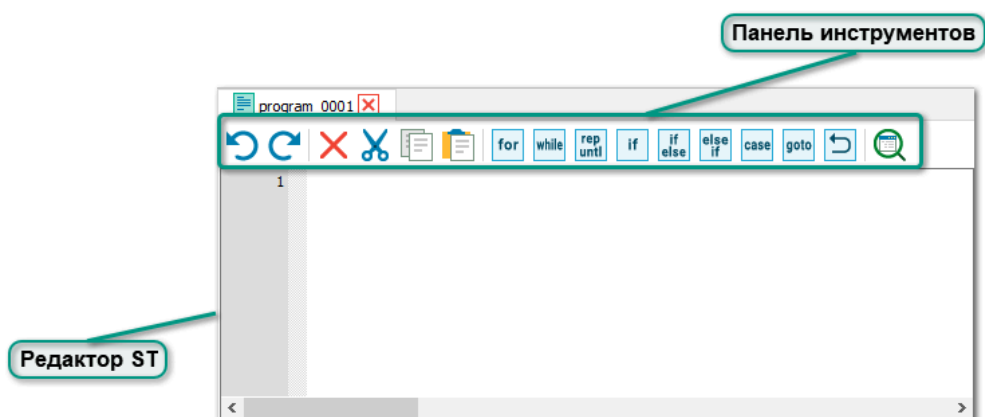
К каждой программе могут быть добавлены дочерние программы. В дочерней программе могут быть вызваны любые функциональные блоки структур и функции проекта, а сама дочерняя программа может быть вызвана только из программы, к которой добавлена. После выполнения дочерняя программа возвращает результат в виде одного выходного значения. В дочернюю программу значения передаются через входные параметры при ее вызове, а при совпадении имен глобальных и локальных переменных, приоритетом будет обладать локальная переменная. Использование типов данных в словаре дочерней программы не ограничено.

Для добавления дочерней программы в дереве проекта кликните правой кнопкой мыши по программе, из контекстного меню выберите пункт «Добавить дочернюю программу», в открывшемся окне введите имя и выберите язык программирования. Для изменения имени структуры используется горячая клавиша «F2» или контекстное меню «Переименовать».

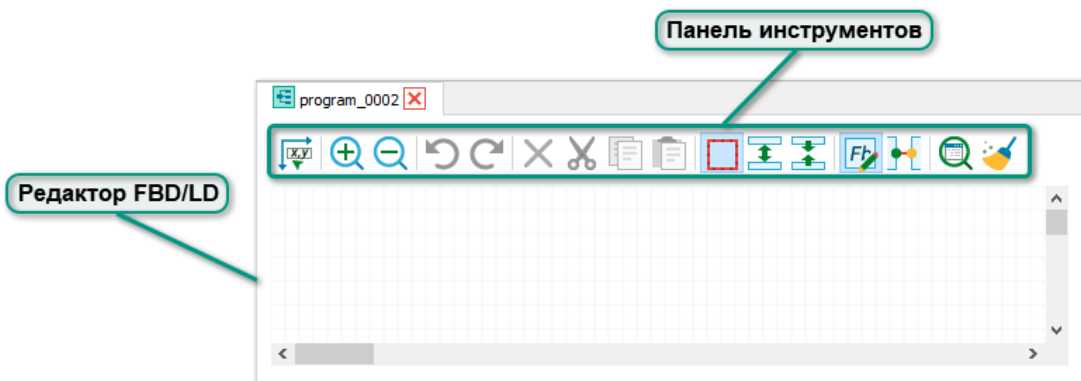


11. Использование редакторов. В зависимости от выбранного языка программирования редактирование программных объектов осуществляется в редакторе ST или FBD/LD.

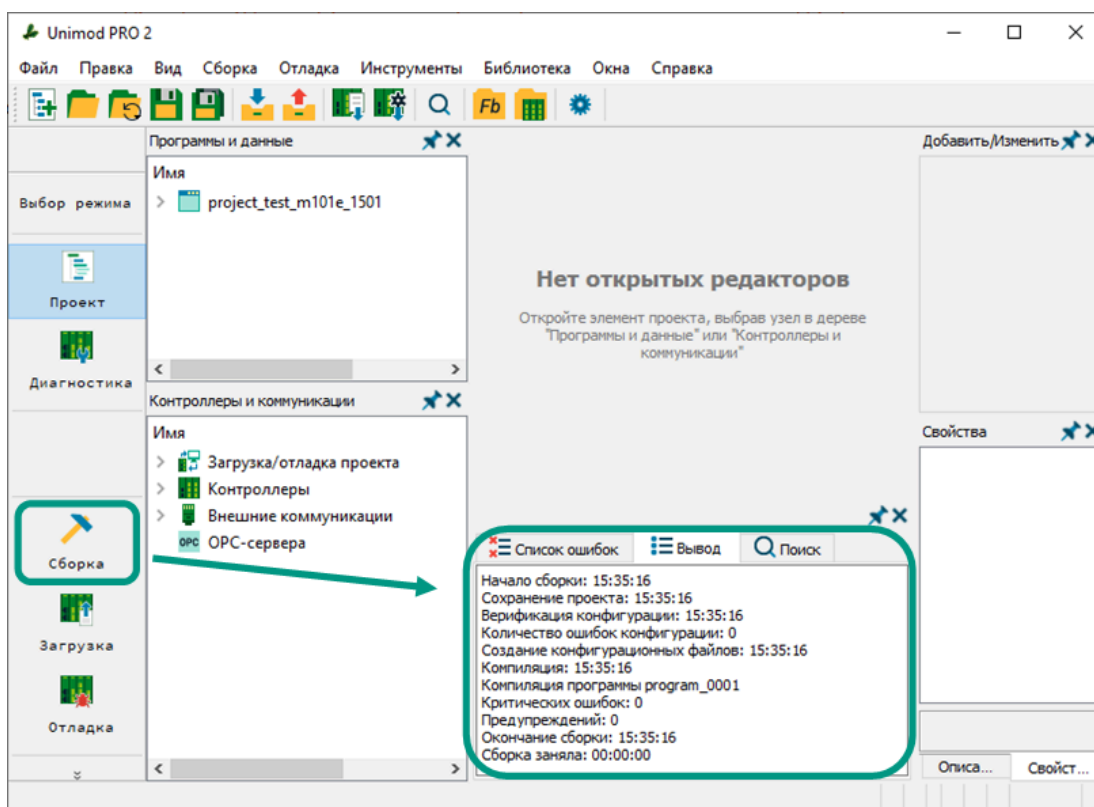
Редактор языка ST имеет функции автодополнения и подсветки кода, на панели инструментов редактора находятся иконки для быстрой вставки основных управляющих конструкции языка. Доступ к полям структур происходит через символ точки. После ввода данного символа появляется окно автодополнения со всеми полями и функциональными блоками структуры. Вызов блока осуществляется через конструкцию вида: *<имя_экземпляра_структуры>. <имя_процедуры>(<параметр1>, <параметр2>..., <параметрN>)*. Для вызова функции достаточно указать ее имя и список параметров.



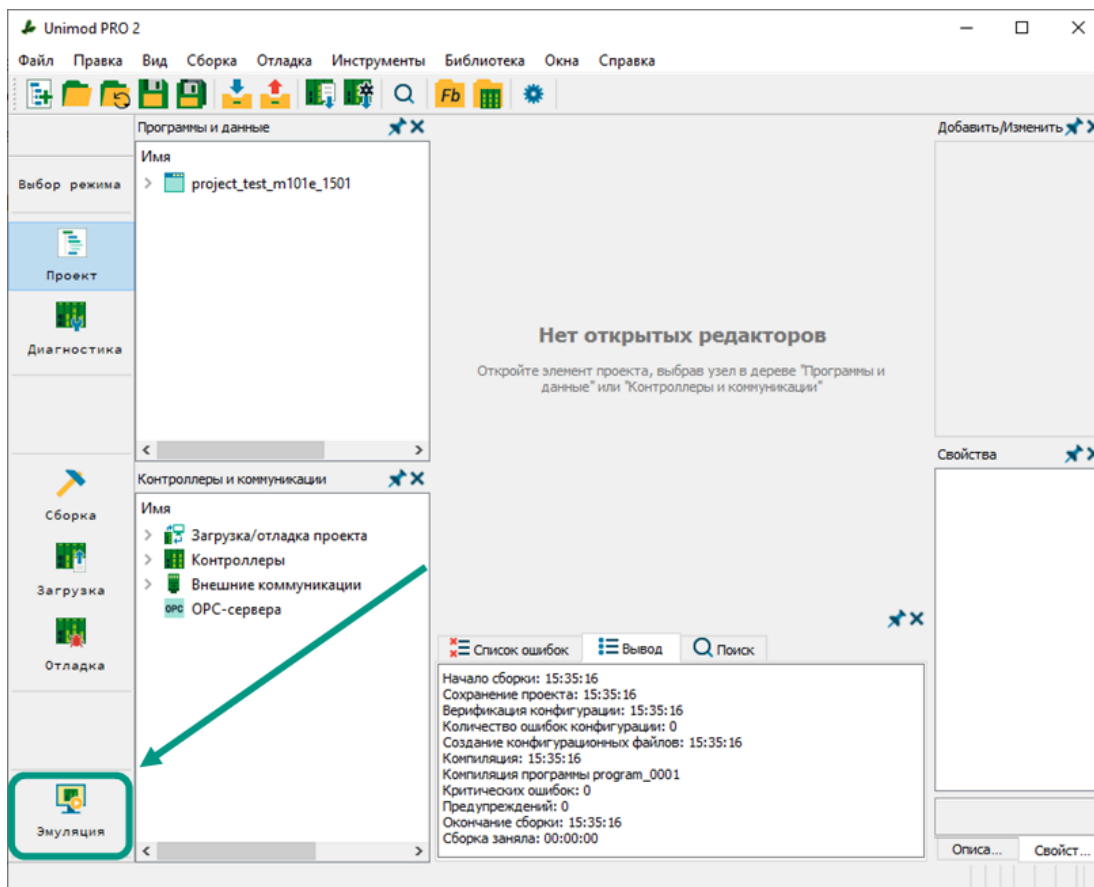
Редактор языков FBD/LD дает возможность произвольной расстановки графических элементов по выбору пользователя в едином графическом поле. Соединительные линии при этом проводятся автоматически после выбора элемента-источника и элемента-приемника. Выходные и входные параметры на соединяемых блоках могут быть выбраны отдельными кликами или путем перемещения при зажатой левой кнопке мыши. Выбранные в процессе соединения параметры выделяются цветом. Выбор элементов для вставки на схему осуществляется на панели «Добавить/Изменить».



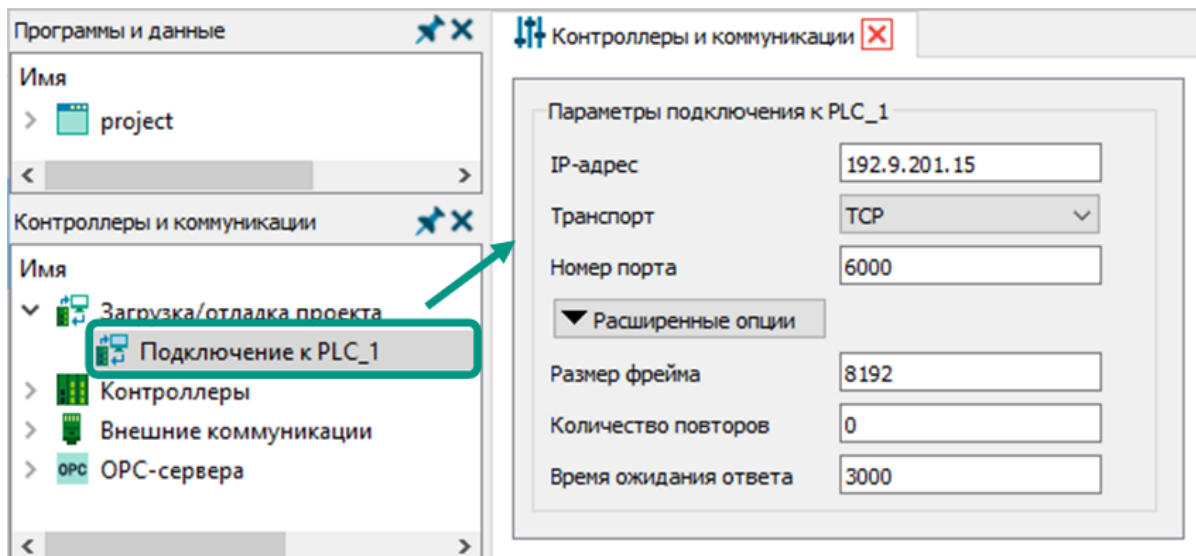
12. Сборка проекта. Сборка применяется для проверки приложения на наличие ошибок и подготовки файлов приложения для последующей загрузки в контроллер. Для сборки приложения на панели выбора режима работы нажмите кнопку «Сборка» или выберите пункт меню «Сборка → Сборка». Сборка проекта доступна только в режиме работы «Проект». Информация о процессе сборки и обнаруженных ошибках будет выведена на вкладках «Вывод» и «Список ошибок». Двойной щелчок по ошибке на вкладке «Список ошибок» позволяет открыть программный объект и установить курсор в позицию с допущенной ошибкой.



13. **Отладка в режиме эмуляции.** Если приложение собрано без ошибок, проверить правильность работы алгоритмов можно в режиме эмуляции. Для этого на панели выбора режима работы нажмите кнопку «Эмуляция» или выберите пункт меню «Отладка → Эмуляция». В режиме эмуляции доступны просмотр и редактирование значений переменных в словарях. Остановить процесс эмуляции можно повторным нажатием на кнопку «Эмуляция» на панели выбора режима работы.

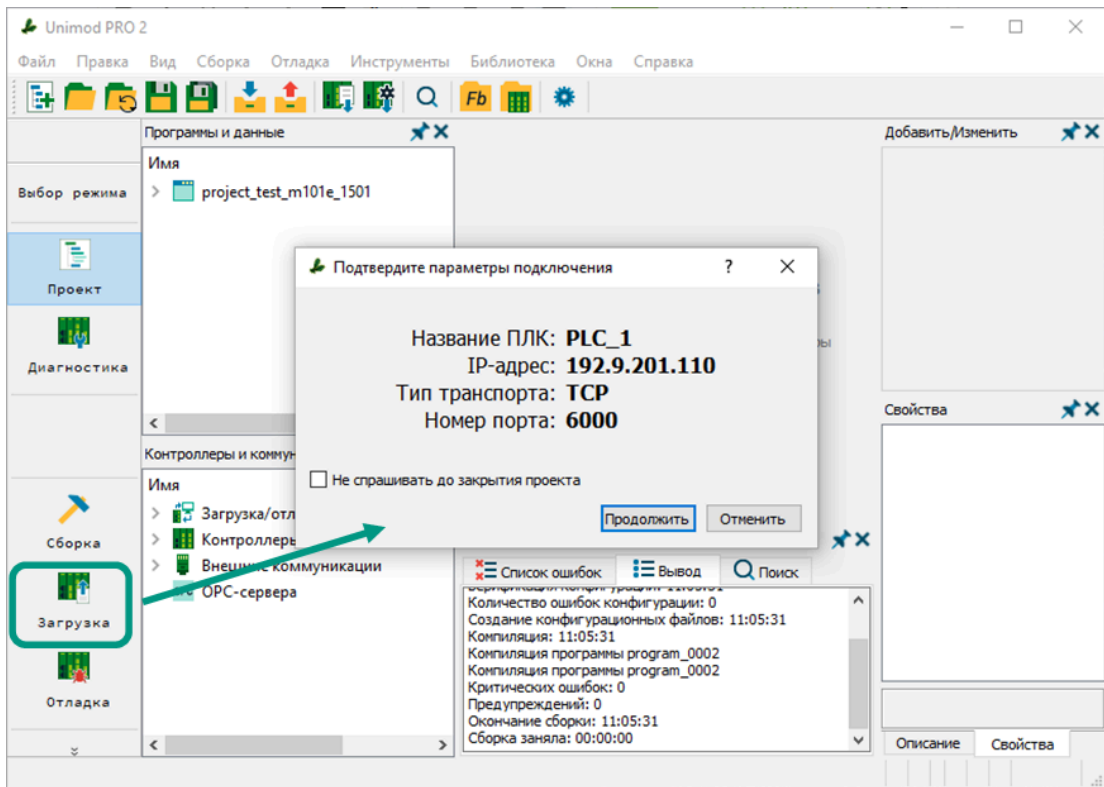


14. Загрузка проекта в контроллер. Для загрузки приложения в контроллер необходимо в дереве конфигурации в разделе «*Загрузка/отладка проекта → Подключение к PLC_1*» указать его IP-адрес, протокол передачи данных и номер порта. Эти параметры должны совпадать с параметрами сетевых настроек приложения, уже загруженного в контроллере.

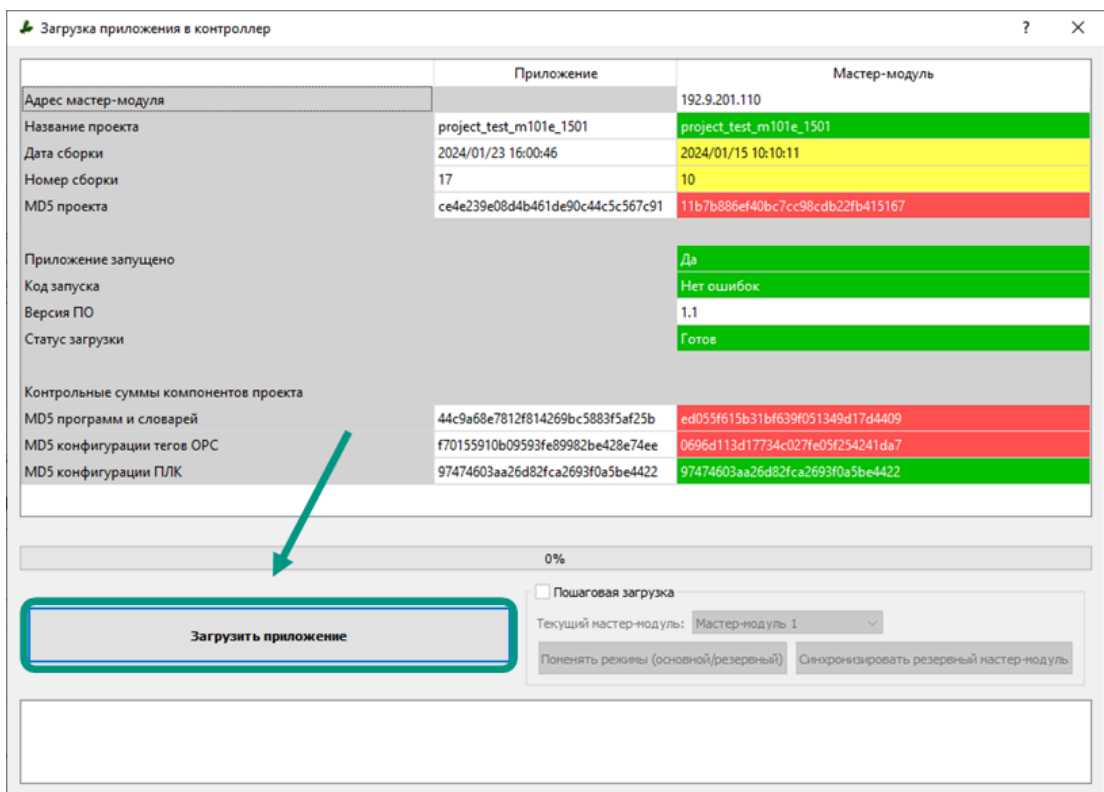


Значения IP-адресов в параметрах подключения и сетевых настройках могут различаться. В параметрах подключения указывается значение одного из текущих IP-адресов контроллера, а в сетевых настройках - значения IP-адресов, которые будут установлены на сетевые интерфейсы контроллера только после загрузки приложения. После загрузки подключение к контроллеру для загрузки и отладки будет осуществляться по одному из IP-адресов, указанных в сетевых настройках загруженного приложения.

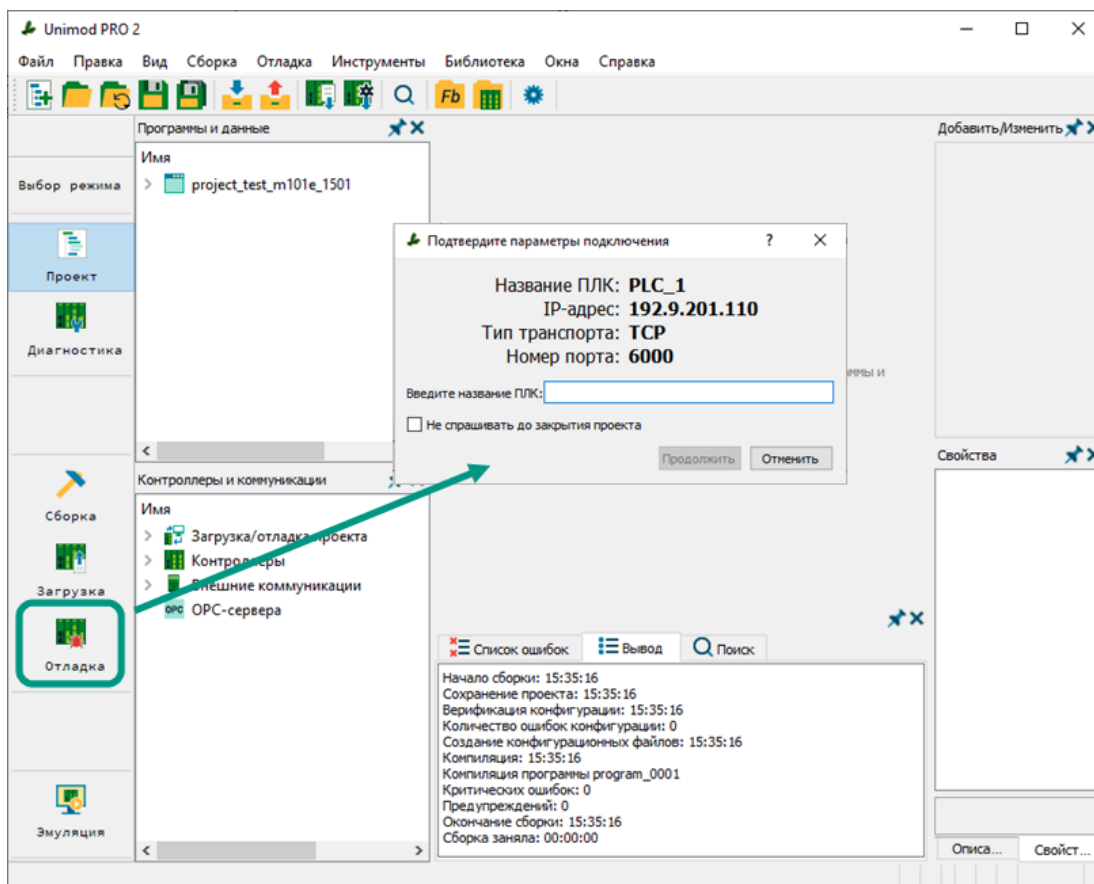
Для загрузки проекта в контроллер на панели выбора режима работы нажмите кнопку «*Загрузка*» или выберите пункт меню «*Отладка → Загрузка*».



При успешном подключении к контроллеру будет открыто окно загрузки, где доступны сведения о состоянии контроллера, параметрах текущего и загруженного в контроллере приложений, а также команды управления загрузкой. Для продолжения загрузки нажмите кнопку «Загрузить приложение».



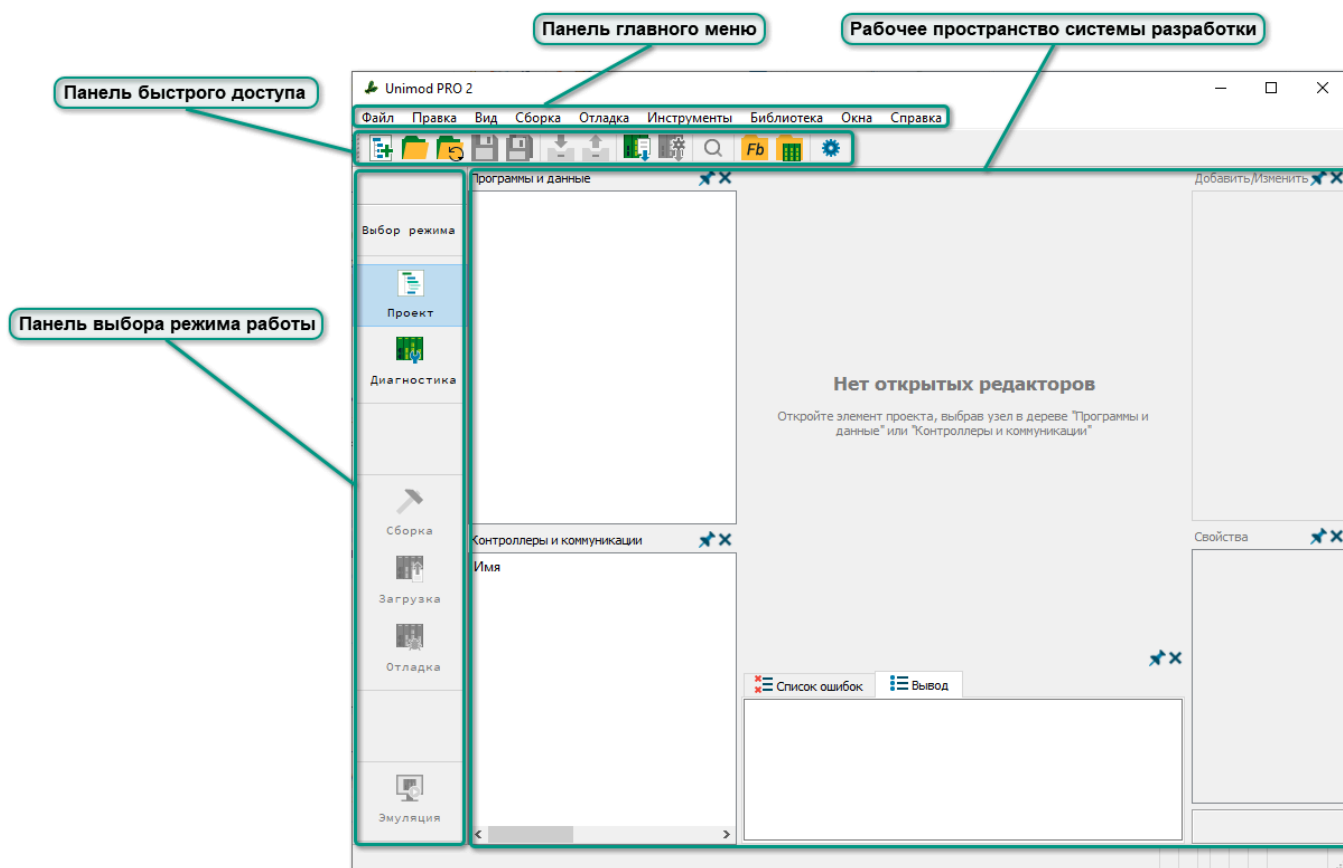
15. **Отладка проекта на контроллере.** После успешной загрузки проекта в контроллер, проверить правильность работы алгоритмов можно в режиме отладки. Для этого на панели выбора режима работы нажмите кнопку «Отладка» или выберите пункт меню «Отладка → Отладка». В режиме отладки доступен просмотр и редактирование переменных словарей. Остановить процесс отладки можно повторным нажатием на кнопку «Отладка» на панели выбора режима работы.



4 Пользовательский интерфейс

Пользовательский интерфейс среды разработки Unimod PRO 2 соответствует международному стандарту GUI (графический интерфейс пользователя), включающему многооконный режим работы, графические и текстовые редакторы, работу с мышью и т.п.

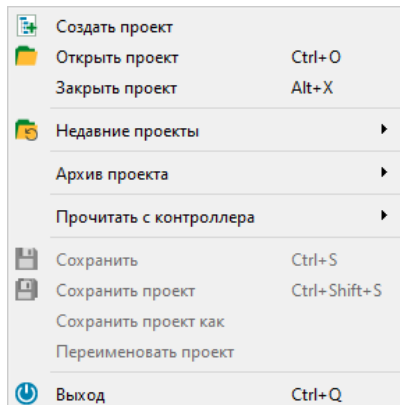
Главное окно среды разработки Unimod PRO 2 содержит следующие элементы:



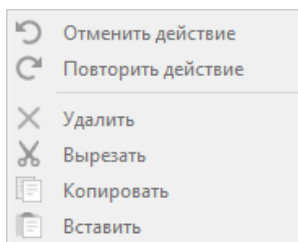
Панель главного меню

Панель главного меню содержит разделы:

Раздел меню **Файл** содержит список команд для управления проектами, а также команду закрытия программы

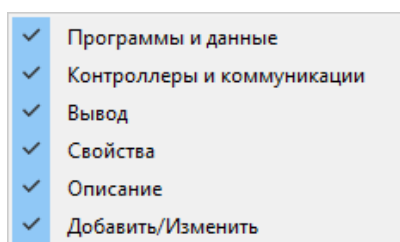


Раздел меню **Правка** содержит список действий с элементами проекта

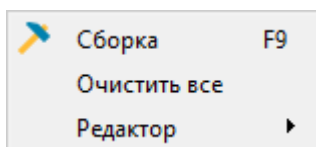


Раздел меню **Вид** содержит список панелей, доступных для открытия в рабочей области системы разработки

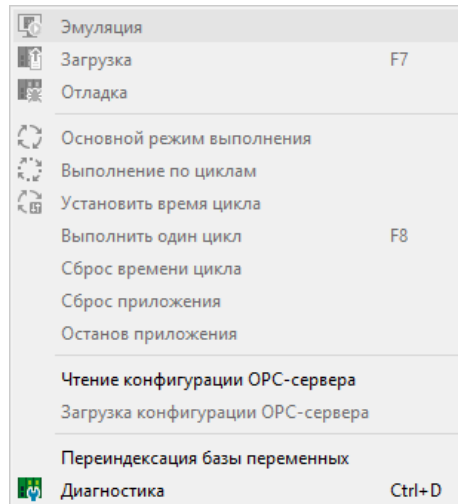
Открытые панели помечены галочкой. Подробнее про настройку рабочего пространства **системы разработки Unimod PRO 2** описано в разделе [Настройка рабочего пространства](#).



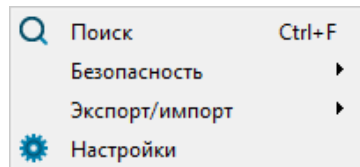
Раздел меню **Сборка** содержит список команд для сборки и проверки приложения перед его последующей загрузкой в контроллер



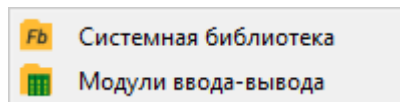
Раздел меню **Отладка** содержит список команд для отладки, диагностики и загрузки проекта в контроллер, а также работы с конфигурацией OPC-сервера



Раздел меню **Инструменты** содержит команды для настроек системы разработки и безопасности проекта, а также команды экспорта и импорта



Раздел меню **Библиотека** позволяет открыть вкладки с содержимым системной библиотеки и библиотеки модулей ввода-вывода в рабочей области системы



Раздел меню **Окна** позволяет закрыть все открытые вкладки, расположенные в рабочей области системы

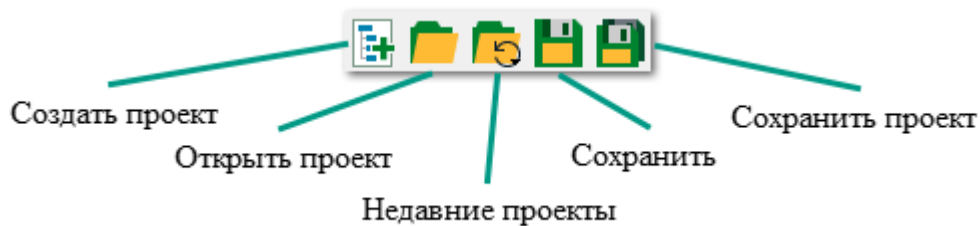
Закрыть все вкладки

Раздел меню **Справка** позволяет в отдельном окне вывести сведения о текущей версии среды разработки Unimod PRO 2

О программе...

Панель быстрого доступа

Команды управления проектом:



Команды работы с архивом проекта и чтения с контроллера:



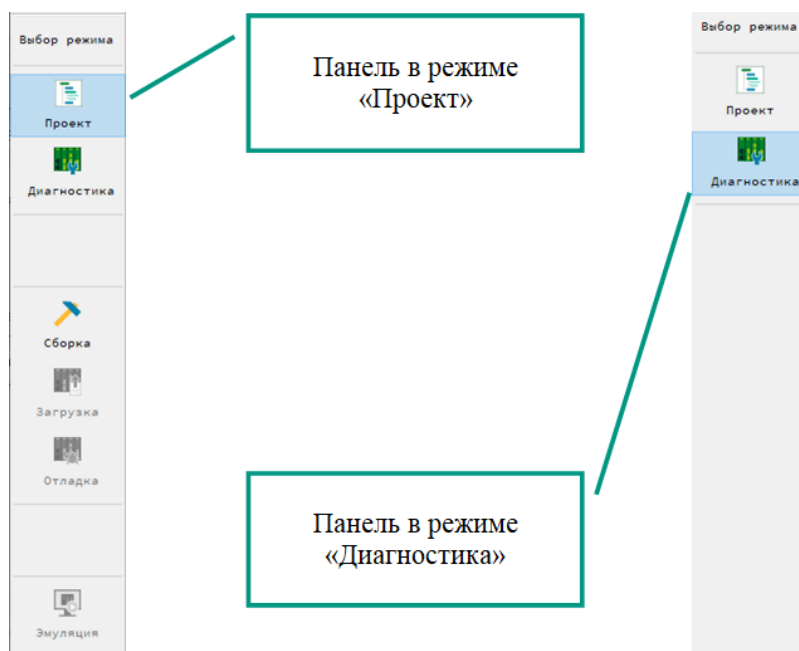
Вспомогательные команды:



Панель выбора режима работы

Панель выбора режима работы позволяет переключаться в режимы «Проект» или «Диагностика». В режиме «Проект» осуществляется разработка и отладка технологического приложения, а также настройка конфигурации исполнительной системы. В режиме «Диагностика» осуществляется мониторинг работы контроллеров, сбор статистики и обновление программного обеспечения. Подробнее про функциональные возможности разработки и диагностики в разделе [Функциональные возможности](#).

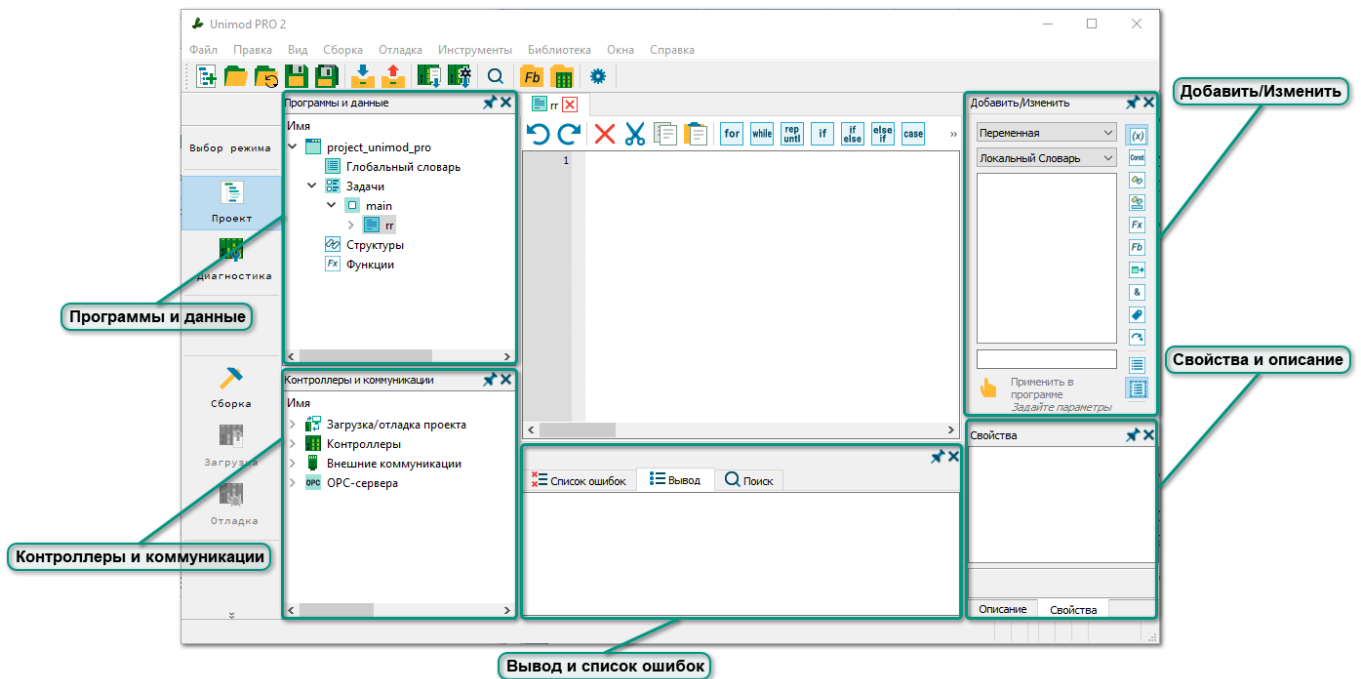
Команды «Сборка», «Загрузка», «Отладка» и «Эмуляция» отображаются на панели только в режиме «Проект».



Рабочее пространство системы разработки

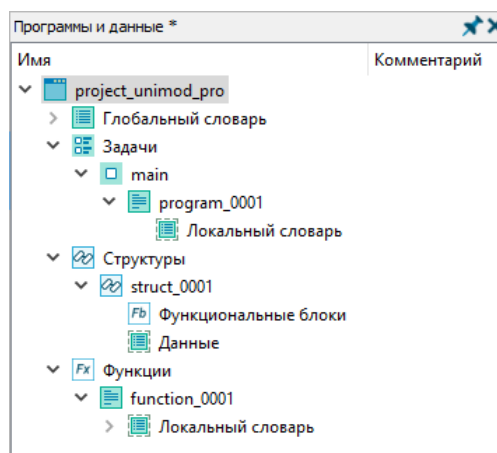
В рабочем пространстве среды разработки Unimod PRO 2 могут располагаться инструментальные панели и вкладки с открытыми редакторами, словарями и оконными формами настроек конфигурации.

4.1 Инструментальные панели





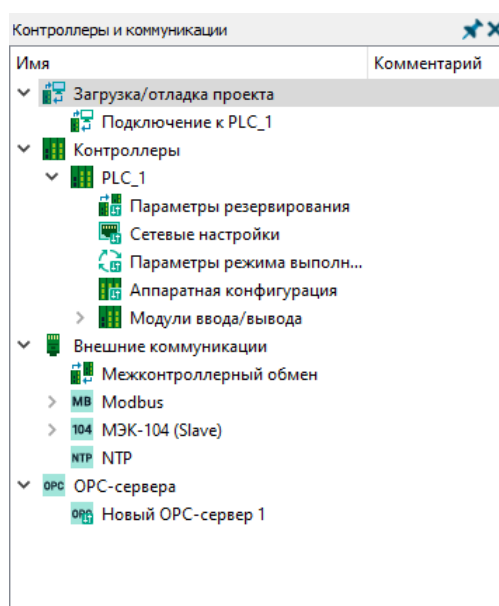
Программы и данные

Панель «Программы и данные» предназначена для управления структурами и данными проекта, задачами и пользовательскими программами, глобальными и локальными словарями. Структура проекта отображается в виде дерева проекта с несколькими уровнями вложенности элементов. Для раскрытия структуры вложенных элементов нажмите на символ **>**, для скрытия на символ **∨**.



Контроллеры и коммуникации

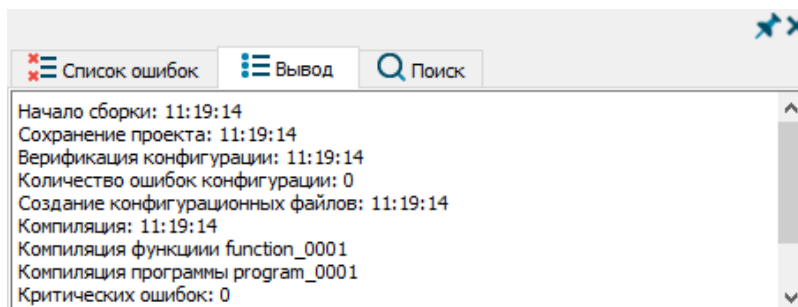
Панель «Контроллеры и коммуникации» предназначена для настройки подключения к контроллерам и их конфигурирования, управления взаимодействием с внешними устройствами, конфигурирования OPC-сервера. Структура элементов, доступных для конфигурирования в рамках технологического проекта, отображается в виде дерева конфигурации с несколькими уровнями вложенности. Для раскрытия структуры вложенных элементов нажмите на символ , для скрытия на символ .



Вывод и список ошибок

На панели располагаются вкладки «Вывод», «Список ошибок», «Поиск» и «Журнал».

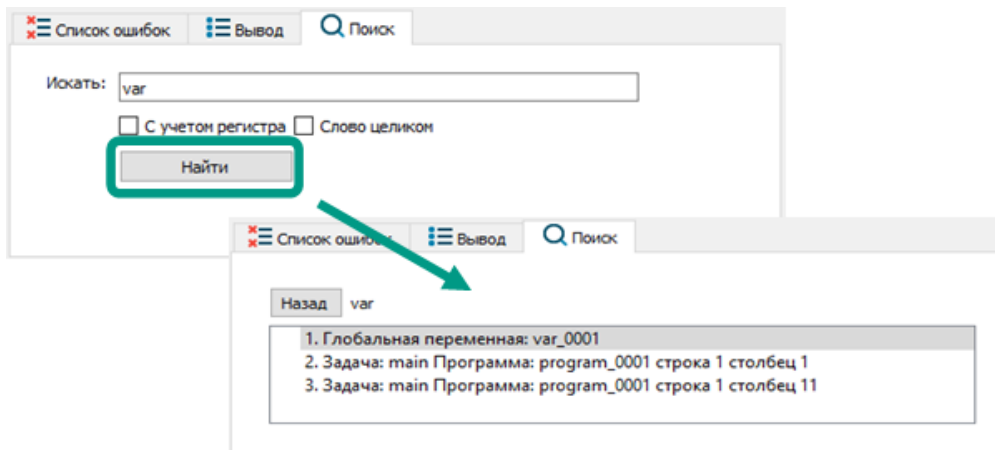
Вкладка «Вывод» предназначена для вывода информации о результатах сборки, компиляции и отладки проекта.



Список ошибок проекта выводится на одноименной вкладке и содержит порядковые номера ошибок, их краткое описание и категория, а также сведения о месте возникновения ошибки с указанием названия программы, номера строки и столбца. Для быстрого открытия редактора программы и перехода к месту возникновения ошибки дважды щелкните левой кнопкой мыши по выбранной ошибке.

№	Описание	Источник	Строка	Столбец	Категория
1	Ошибка 1: Ошибка синтаксиса	program_0001	1	1	ошибка
2	Ошибка 1: Ошибка синтаксиса	program_0001	1	2	ошибка
3	Ошибка 1: Ошибка синтаксиса	program_0001	2	1	ошибка

Вкладка «Поиск» позволяет находить указанный фрагмент текста в проекте. Поиск осуществляется по именам элементов проекта и среди текста программных объектов. Каждое совпадение имени выводится в отдельном списке, а при нажатии на строку совпадения открывается соответствующий редактор или словарь с найденным фрагментом. Для поиска с учетом регистра или слова целиком выберите соответствующие параметры.

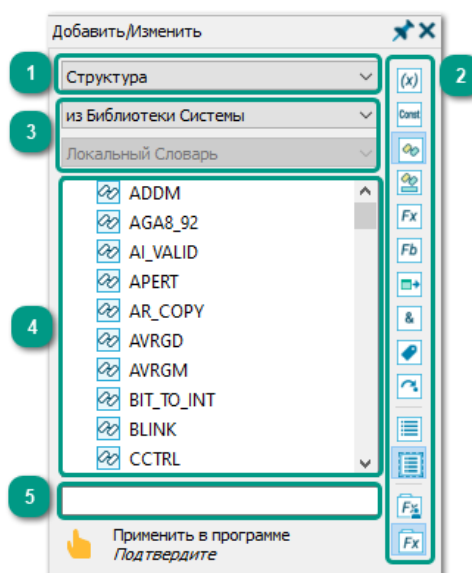


Вкладка «Журнал» доступна только при отладке проекта на реальном контроллере. На вкладке выводятся системные события контроллера.

№	Событие	Время появления	Группа события
5	Шлюз (ТСР). Подключился абонент 192:9:201:15	07:59:44 26.02.2024	Задачи связи
6	Шлюз (ТСР). Отключился абонент 192:9:201:15	15:54:26 22.02.2024	Задачи связи
7	Остановка внешним запросом	14:52:52 22.02.2024	Сброс
8	Шлюз (ТСР). Отключился абонент 192:9:201:15	14:20:58 22.02.2024	Задачи связи
9	Шлюз (ТСР). Подключился абонент 192:9:201:15	14:20:49 22.02.2024	Задачи связи
10	LAN 2. Кабель не подключен	14:20:45 22.02.2024	Аппаратная ошибка

Добавить/Изменить

Панель «Добавить/Изменить» предназначена для поиска, выбора и добавления элементов в редактор программного объекта. Полный перечень доступных для добавления элементов приведен в документе «Руководство по программированию».

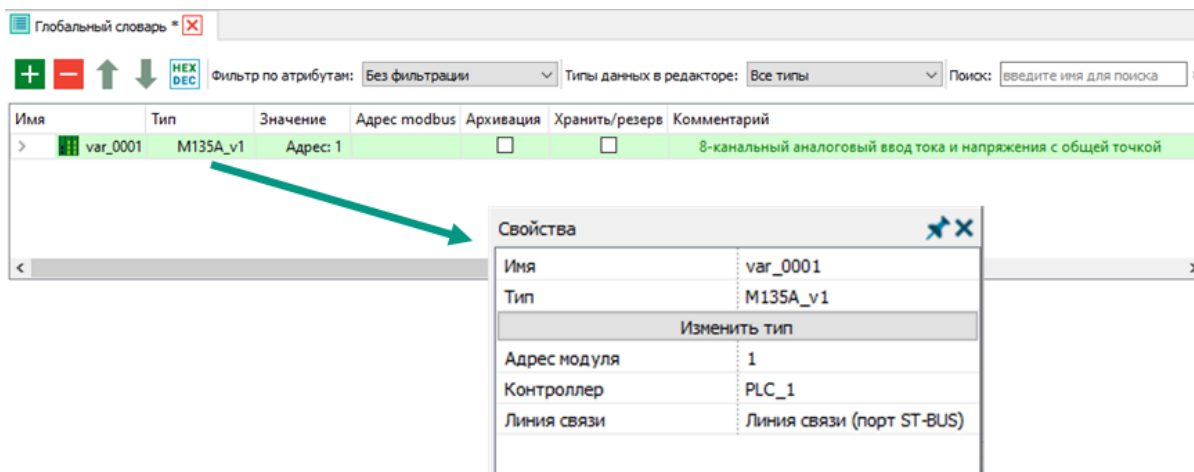


Выбрать тип требуемого элемента можно в раскрывающемся списке ¹ или на панели ², где каждый тип представлен в виде символов, при наведении на символ выводится название. Списки в области ³ и символы на панели ² зависят от выбранного типа элемента. Доступен выбор из глобального словаря (📖) или локальных словарей (📖), из библиотеки проекта (Fx) или библиотеки системы (Fx). Для некоторых типов доступны списки модификаций и поле ввода значения. Подробнее про словари и библиотеки в разделах [Переменные](#) и [Библиотеки](#) настоящего руководства.

Найденные элементы будут отображаться в окне ⁴. Для быстрого поиска элемента воспользуйтесь полем ⁵. Добавление в редактор осуществляется перетаскиванием элемента или символа 🖱️.

Свойства и описание

На вкладке «Свойства» выводятся свойства переменной модульной структуры при ее выделении в одном из словарей, а вкладка «Описание» в настоящее время находится в разработке, на ней планируется выводить краткое описание элементов системной библиотеки.



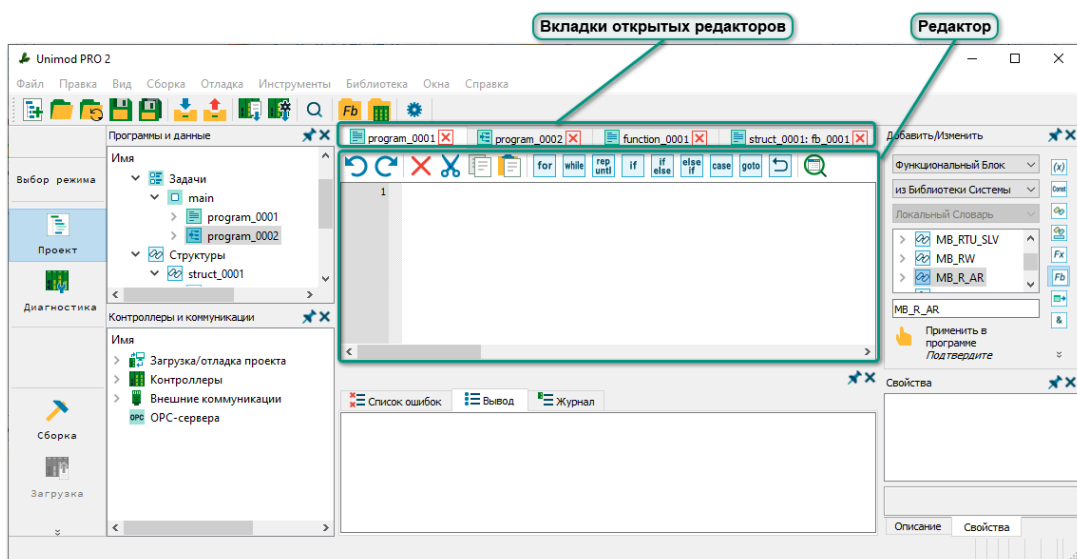
Процесс настройки размеров и расположения панелей в рабочем пространстве среды разработки описан в разделе [Настройка рабочего пространства](#).


4.2 Редакторы

Редакторы в среде разработки Unimod PRO 2 предоставляют необходимый набор инструментов для написания и редактирования созданных пользователем программных объектов: программ и функциональных блоков. Написание программных объектов производится на выбранном при их создании языке программирования, который не может быть изменен в процессе работы. Для написания программных объектов на языке *ST* используется *редактор ST*, а для языков *FBD/LD* - *редактор FBD/LD*. Выбор редактора осуществляется средой разработки автоматически. При необходимости изменить язык программирования, потребуется создать новый программный объект.

Для открытия редактора дважды кликните по программному объекту в дереве проекта. Одновременно могут быть открыты несколько редакторов, их список отображается в рабочем пространстве в виде вкладок.

Выбор требуемого редактора из нескольких вкладок осуществляется нажатием левой кнопки мыши. Порядок вкладок изменяется перемещением мыши при зажатой левой клавише.



Для закрытия вкладки нажмите по вкладке правой кнопкой мыши и из контекстного меню выберите пункт «*Закреть вкладку*» или нажмите на кнопку . Для закрытия нескольких вкладок в контекстном меню также доступны пункты «*Закреть все, кроме текущей вкладки*» и «*Закреть все вкладки*». В качестве альтернативного варианта все вкладки можно закрыть, выбрав пункт главного меню «*Окна → Закреть все вкладки*».

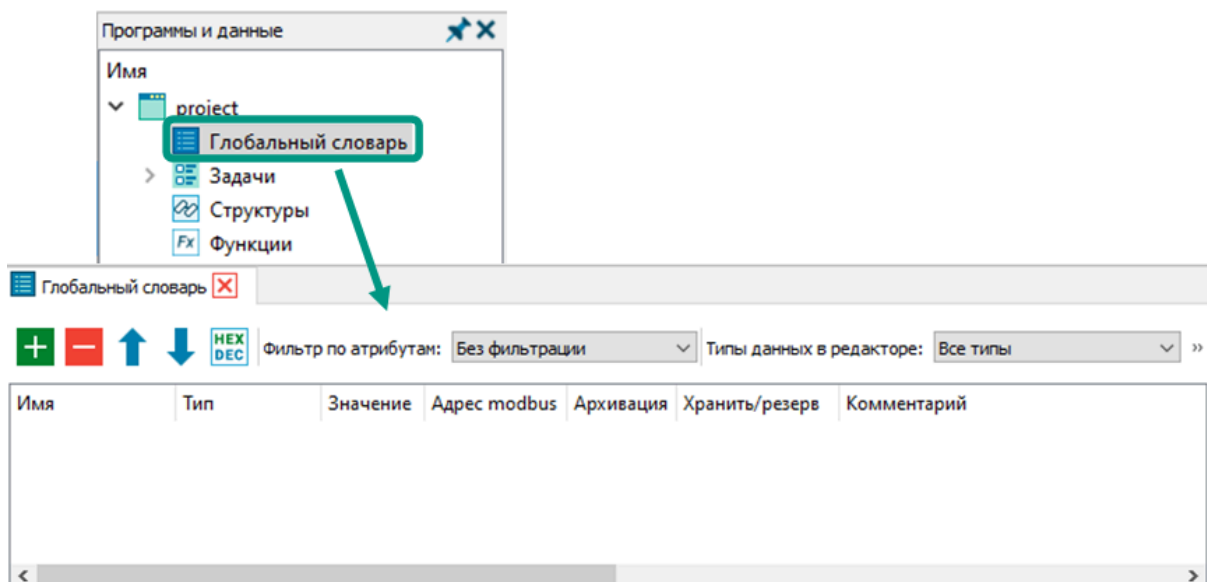
Для поиска редактируемого элемента в дереве проекта нажмите правой кнопкой мыши по вкладке и из контекстного меню выберите пункт «*Поиск в*

обозревателе решений», после чего в дереве проекта редактируемый элемент будет выделен цветом.


4.2.1 Редактор словаря

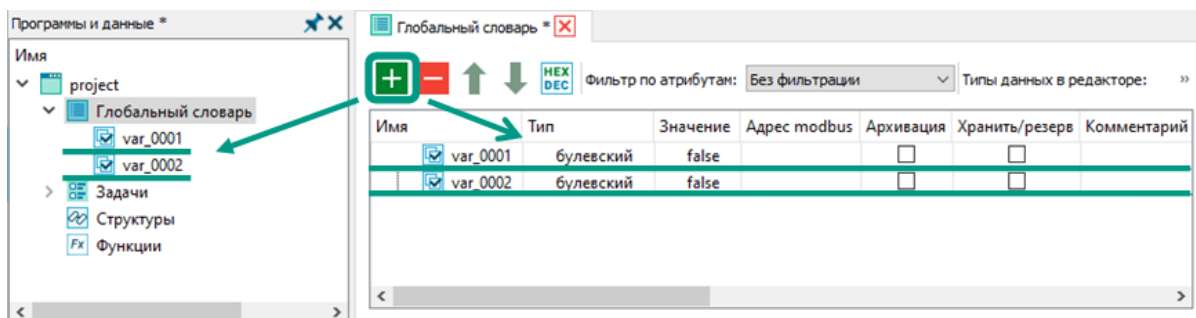
В среде разработки Unimod PRO 2 переменные хранятся в одном из трех словарей: глобальном, локальном или словаре данных структуры. Добавление, хранение, редактирование и удаление переменных, их поиск, фильтрация, экспорт и импорт осуществляются с использованием редактора словаря. При этом интерфейс и возможности редактора при работе с глобальным, локальным или словарем данных структуры идентичен, отличаются только списки фильтров по атрибутам и типам данных. Поэтому дальнейшее описание редактора будет основано на примере работы с глобальным словарем. Процессы работы с локальным словарем и словарем данных структуры построены аналогичным образом.

Для открытия редактора словаря раскройте список элементов проекта нажатием на символ \triangleright → дважды кликните левой кнопкой мыши по выбранному словарю в дереве проекта или выберите из его контекстного меню пункт «Открыть» → редактор будет открыт в рабочем пространстве среды разработки в виде вкладки.



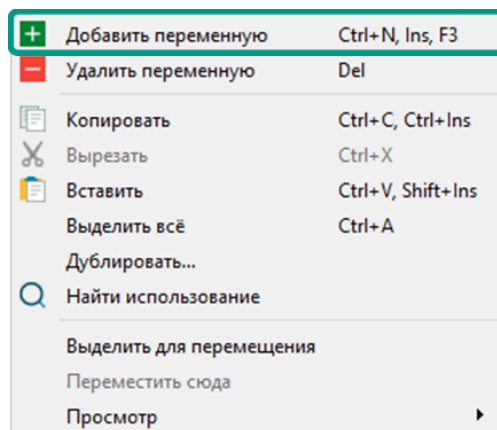
4.2.1.1 Управление переменными

Для создания переменной нажмите кнопку  → в конце списка появится запись с созданной переменной → укажите атрибуты переменной. При этом переменные словаря будут отображаться в виде списка в словаре и в дереве конфигурации. Количество переменных в словарях не ограничено и зависит от свободной памяти контроллера.

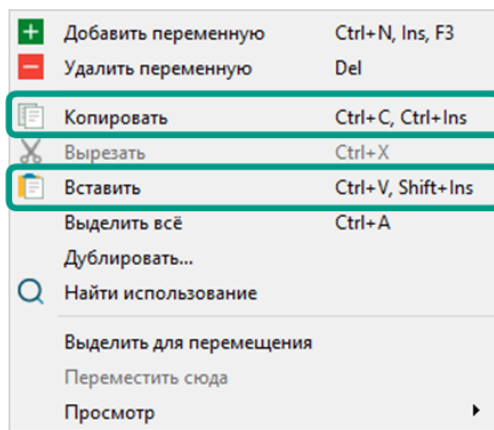


Альтернативные способы создания переменной:

- нажмите клавиши «*Ctrl + N*», «*Ins*» или «*F3*»;
- кликните правой кнопкой мыши по одной из записей в словаре → из контекстного меню выберите пункт «*Добавить переменную*».

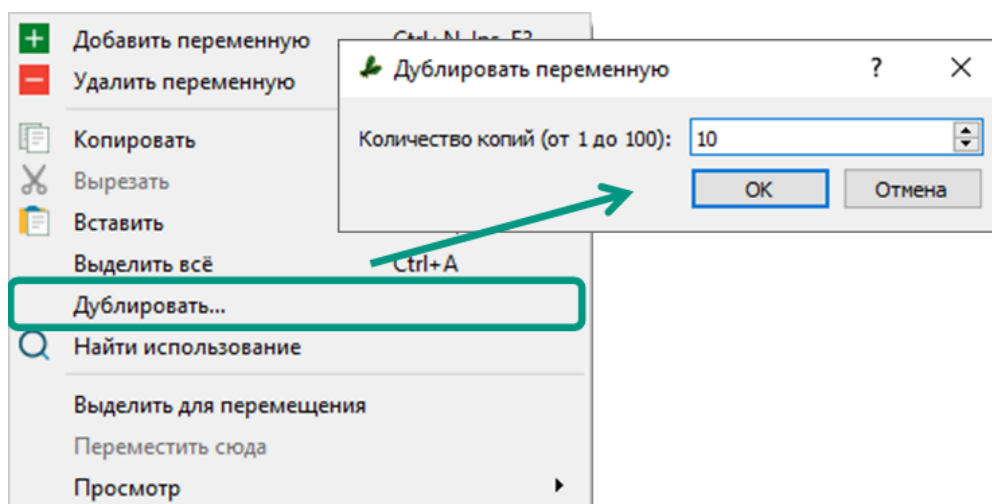


Для копирования переменной кликните правой кнопкой мыши по переменной → из контекстного меню выберите пункт «*Копировать*» → кликните правой кнопкой мыши по позиции для добавления → из контекстного меню выберите пункт «*Вставить*». При копировании создается полная копия переменной с сохранением значений атрибутов. Скопированная переменная отличается от исходной только именем, так как в пределах словаря имена переменных должны быть уникальными.




Альтернативные способы копирования переменной:

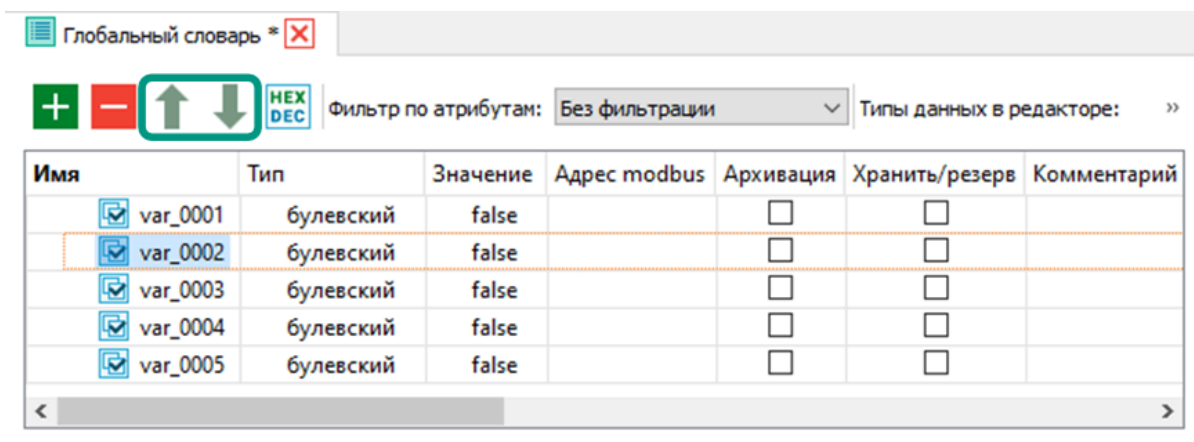
- нажмите сочетание клавиш «*Ctrl + C*» или «*Ctrl + Ins*» → с помощью мыши выберите позицию для добавления → нажмите сочетание клавиш «*Ctrl + V*» или «*Shift + Ins*»;
- кликните правой кнопкой мыши по переменной → из контекстного меню выберите пункт «*Дублировать...*» → в открывшемся диалоговом окне введите количество копий переменной от 1 до 100 → нажмите кнопку «*Ок*».



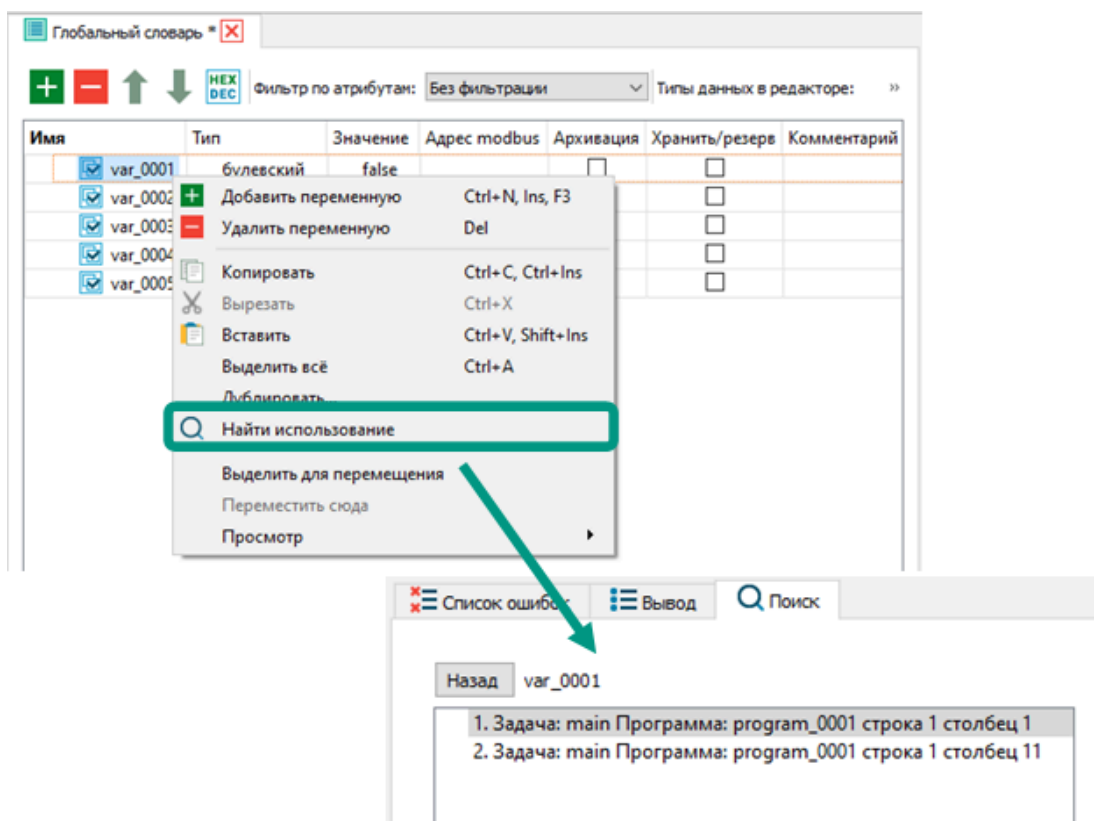
Для переименования переменной дважды кликните левой кнопкой мыши по имени переменной или выберите переменную и нажмите клавишу «*F2*» → введите новое имя → нажмите клавишу «*Enter*». При введении имени следуйте правилам, описанным для атрибута «*Имя*» в разделе [Словари](#).


Для вывода значений переменных в шестнадцатеричном формате на панели инструментов нажмите кнопку . В шестнадцатеричном формате могут быть представлены значения переменных байтового, целочисленного и вещественных типов данных.

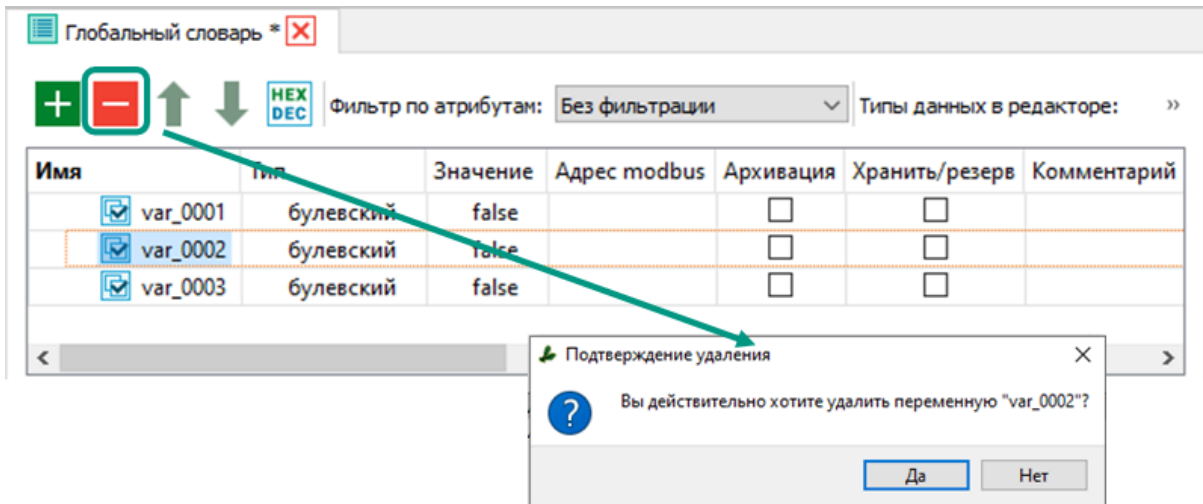
Для изменения позиции выделите переменную → нажатием кнопок ↑ или ↓ переместите переменную вверх или вниз по списку, одно нажатие на указанные кнопки перемещает переменную на одну позицию. Также доступно перемещение переменной при зажатой левой кнопке мыши.



Для поиска программных объектов, в которых используется переменная, нажмите правой кнопкой мыши по переменной → из контекстного меню выберите пункт «Найти использование». Результаты поиска в виде списка всех программных объектов, в которых используется переменная, будут выведены на вкладке «Поиск». Для перехода к месту использования переменной в редакторе программного объекта дважды щелкните левой кнопкой мыши по элементу списка.

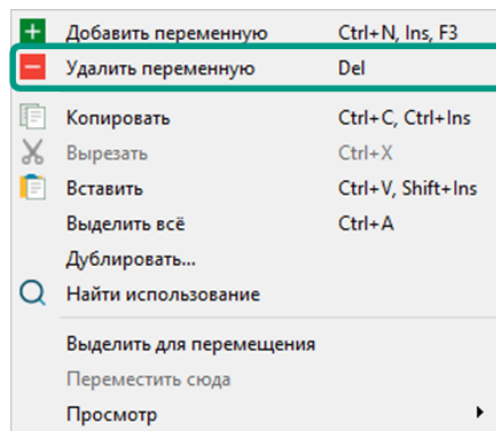


Для удаления переменной выберите удаляемую переменную → нажмите кнопку  → в открывшемся окне подтверждения нажмите кнопку «Да».



Альтернативные способы удаления переменной:

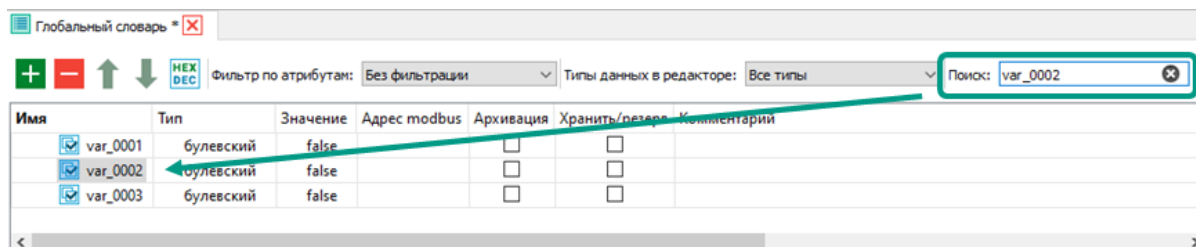
- нажмите клавишу «Delete»;
- кликните правой кнопкой мыши по переменной → из контекстного меню выберите пункт «Удалить переменную».



4.2.1.2 Поиск и фильтрация

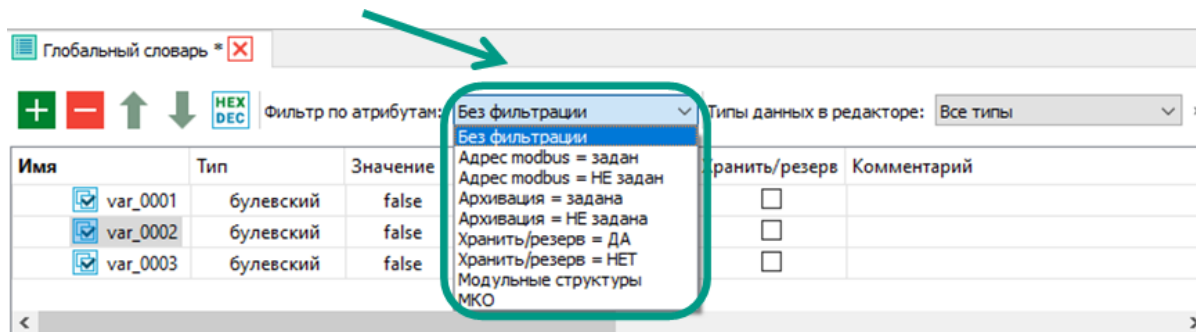
Поиск переменной в словаре.

На панели инструментов в поле «Поиск» введите имя переменной или часть имени → найденная в словаре переменная будет выделена цветом. При этом вводимый в поле «Поиск» текст будет выделяться красным цветом, если переменная с заданным именем не найдена в словаре.



Фильтрация переменных по значению атрибутов.

В словарях среды разработки предусмотрена фильтрация переменных по значениям их атрибутов. Это позволяет упростить процесс поиска требуемых переменных при большом объеме словаря. По умолчанию выводится список переменных без фильтрации. Для выбора фильтра левой кнопкой мыши нажмите на свернутый список «Фильтры по атрибутам» → выберите требуемое значение → в словаре будут выведены только переменные с выбранным значением атрибута.



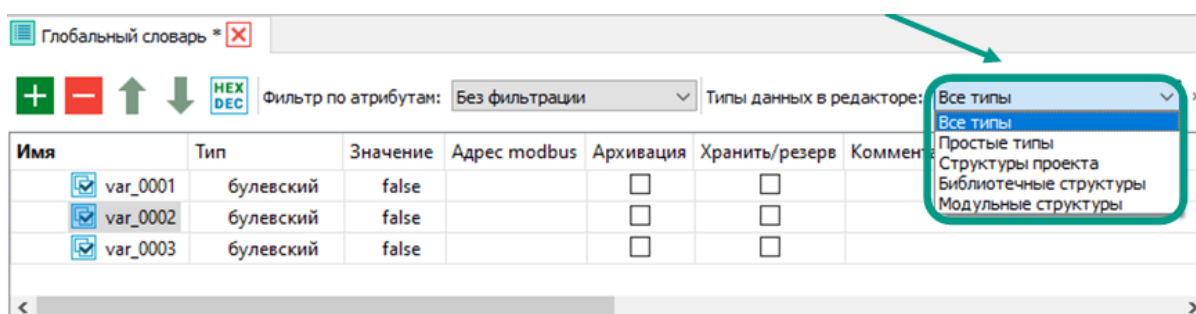
Доступные для выбора фильтры по атрибутам:

- **без фильтрации** – все переменные;
- **адрес modbus = задан** – переменные с заданным modbus-адресом;
- **адрес modbus = НЕ задан** – переменные без modbus-адреса;
- **архивация = задана** – архивируемые переменные;
- **архивация = НЕ задана** – не архивируемые переменные;
- **хранить/резерв = ДА** – сохраняемые в энергонезависимой памяти переменные;
- **хранить/резерв = НЕТ** – не сохраняемые в энергонезависимой памяти переменные;
- **модульные структуры** – переменные с типом модульных структур;
- **МКО** – переменные, задействованные в межконтроллерном обмене.

Фильтрация типов данных.

Процесс поиска типа данных переменной может быть упрощен благодаря фильтрации доступных типов. Это позволяет выбирать тип данных из ограниченного списка, а не искать среди всех доступных вариантов. Выбор типа данных для переменной осуществляется в ячейках столбца «Тип», при нажатии на которые раскрываются списки доступных типов. По умолчанию выводятся все доступные в проекте типы данных.

Для фильтрации типов данных левой кнопкой мыши нажмите на свернутый список «Типы данных в редакторе» → выберите требуемое значение → у переменных для выбора будут доступны только типы данных, указанные в фильтре.





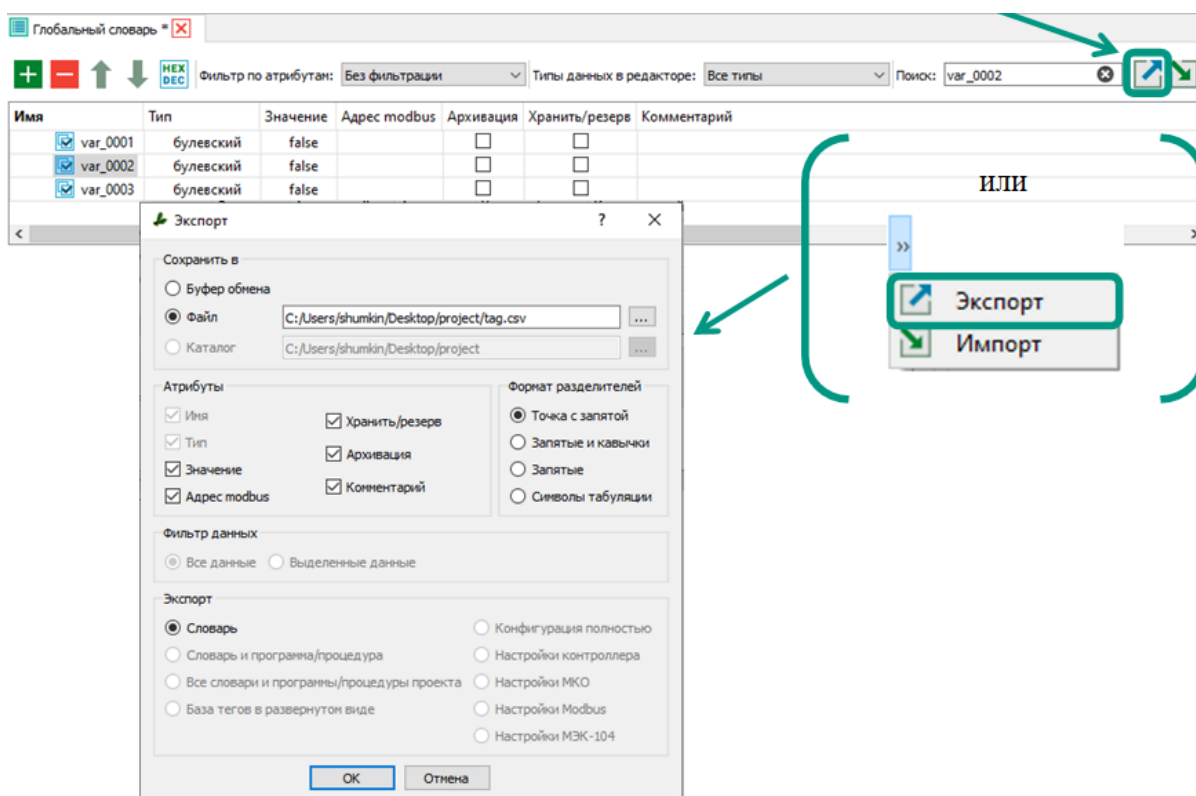
Доступные для выбора фильтры по типам данных:

- **все типы** – все доступные типы данных;
- **простые типы** – простые типы данных (выведены на вышеуказанном изображении);
- **структуры проекта** – типы пользовательских структур;
- **библиотечные структуры** – типы структур из системной библиотеки;
- **модульные структуры** – типы структур из библиотеки модулей ввода-вывода.


4.2.1.3 Экспорт/Импорт переменных

В среде разработки Unimod PRO 2 предусмотрена возможность экспортировать/импортировать переменные для их переноса между отдельными проектами.

Для экспорта переменных откройте словарь → на панели инструментов нажмите кнопку  (при недостаточной ширине окна нажмите на кнопку  в правом верхнем углу словаря и выберите пункт «Экспорт») → в открывшемся диалоговом окне настройте параметры экспорта с учетом нижеуказанных рекомендаций → нажмите на кнопку «Ок».





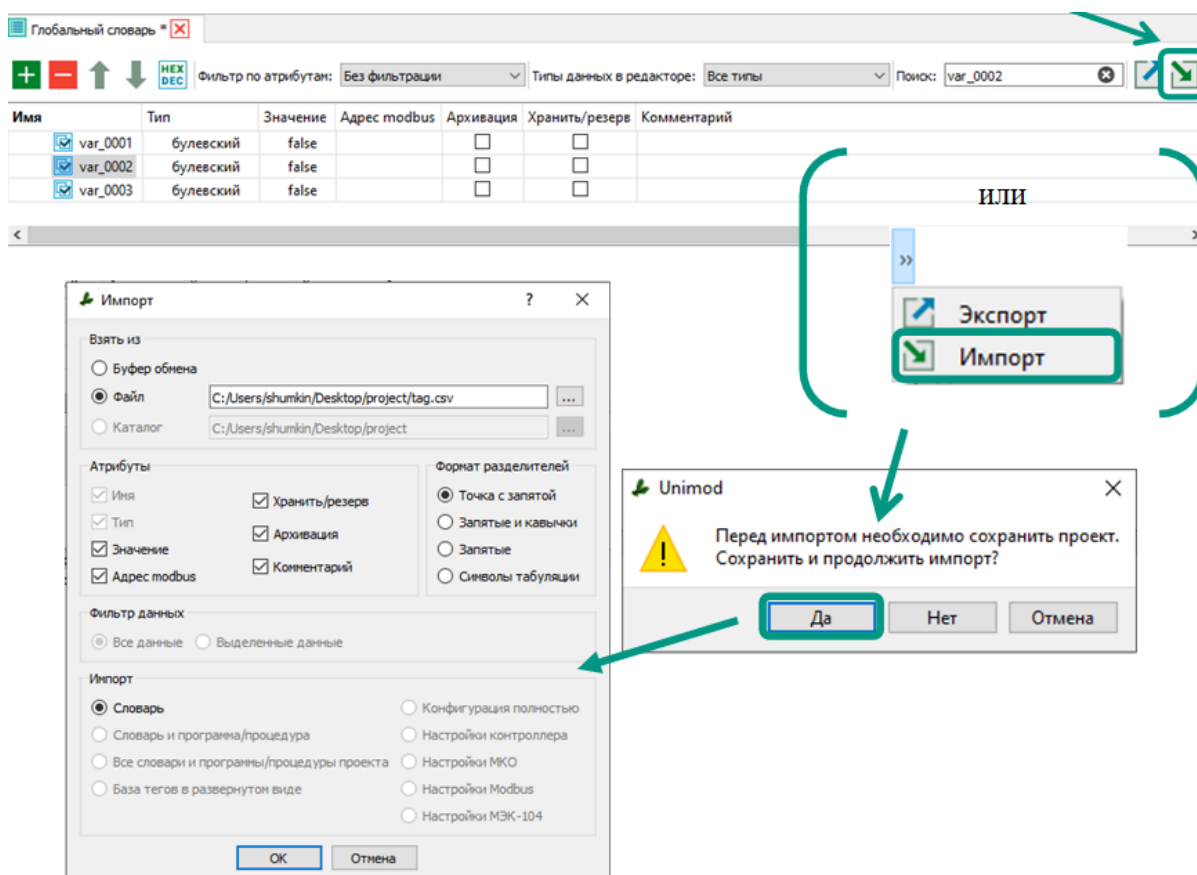
Рекомендации по настройке параметров экспорта переменных:

- для длительного хранения или переноса на сторонний ПК экспортируемых переменных выберите пункт «Сохранить в файл», а для выбора каталога сохранения и указания имени файла нажмите на кнопку .
- при указании имени файла с расширением «*.csv» экспортируемые переменные будут доступны для просмотра и редактирования в табличном виде при открытии с использованием табличных редакторов.
- выбирайте пункт «Сохранить в Буфер обмена» только при необходимости переноса переменных между проектами в пределах одного ПК. Сразу после экспорта рекомендуется импортировать переменные в требуемый проект,


чтобы исключить вероятность потери данных из-за перезаписи буфера обмена.

- не рекомендуется изменять формат разделителей, так как выбранный формат нужно будет запоминать и указывать в настройках импорта. При несовпадении формата разделителей, выводится ошибка, а импорт не производится. По умолчанию формат разделителя в настройках экспорта и импорта «Точка с запятой».

Для импорта переменных откройте словарь → на панели инструментов нажмите кнопку  (при недостаточной ширине окна нажмите на кнопку  в правом верхнем углу словаря и выберите пункт «Импорт») → в открывшемся окне подтвердите сохранение проекта нажатием кнопки «Да» → настройте параметры импорта с учетом нижеуказанных рекомендаций → нажмите на кнопку «Ок».

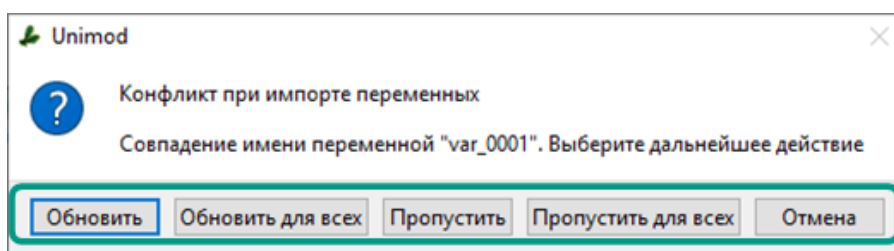


Рекомендации по настройке параметров Импорта переменных:

- для импорта из файла выберите пункт «Файл», после чего нажмите на кнопку  и в открывшемся окне выберите файл с импортируемыми переменными.
- для импорта из буфера выберите пункт «Буфер обмена». При выборе данного варианта может возникнуть ошибка, связанная с перезаписью буфера обмена и утерей экспортированных переменных.

- убедитесь, что выбранный формат разделителей соответствует настройкам экспорта. При несоответствии формата выводится ошибка, а импорт не производится.

При импорте переменных может возникнуть конфликт, если имя импортируемой переменной совпадает с именем одной из текущих переменных словаря. Для предотвращения неконтролируемой перезаписи текущих переменных словаря, при обнаружении конфликта открывается диалоговое окно с предупреждением и предложением пользователю самостоятельно выбрать один из предложенных вариантов дальнейших действий.



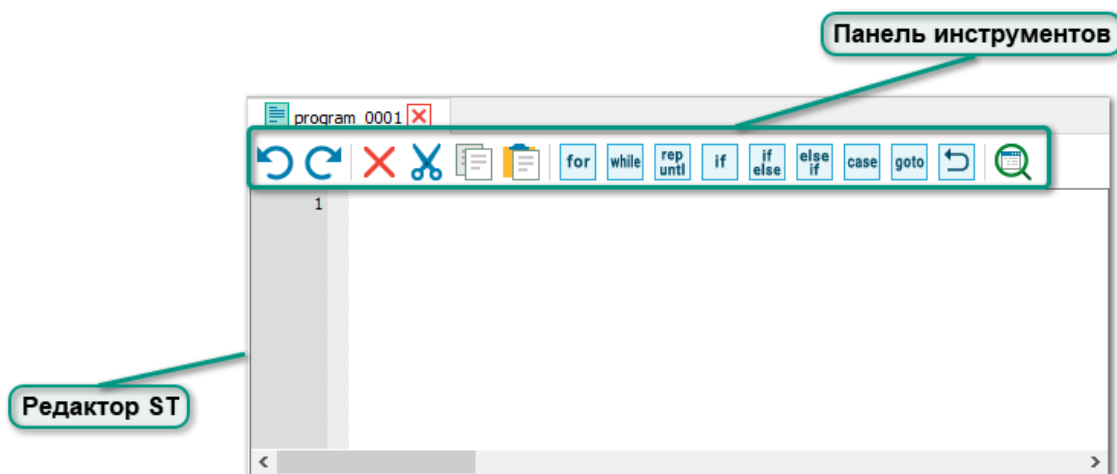
Варианты действий при конфликте импорта переменных:

- **Обновить** – текущая переменная словаря будет заменена на импортируемую. Выбирайте данный вариант с осторожностью! Могут возникнуть ошибки при несовпадении значений атрибутов у конфликтующих переменных (например, замена переменной целочисленного типа данных на импортируемую логическую переменную, приведет к ошибке при выполнении математических операций и т.д.).
- **Обновить для всех** – все текущие переменные словаря будут заменены на импортируемые. Выбирайте данный вариант с осторожностью! Могут возникнуть ошибки при несовпадении значений атрибутов у конфликтующих переменных.
- **Пропустить** – замена текущей конфликтующей переменной на импортируемую не производится.
- **Пропустить для всех** – замена всех текущих конфликтующих переменных на импортируемые не производится.
- **Отмена** – отмена импорта, изменения в проект не вносятся.

4.2.2 Редактор ST

Редактор ST включает в себя необходимый набор инструментов для создания и редактирования программных объектов (программ и функциональных блоков) на языке программирования ST. Редактор имеет функции автодополнения и подсветки кода, на панели инструментов редактора находятся кнопки для быстрой вставки основных управляющих конструкции языка. Доступ к полям структур осуществляется через символ точки, после ввода которого появляется окно автодополнения со всеми полями и функциональными блоками структуры. Вызов функционального блока осуществляется через конструкцию вида: `<имя_экземпляра_структуры>.<имя_процедуры>(<параметр1>, <параметр2>..., <параметрN>)`.

Язык ST, в основном, используется для создания сложных процедур, которые не могут быть легко выражены при помощи графических языков. Программы на языке ST выглядят в виде структурированного текста. Подробнее с синтаксисом и возможностями языка программирования ST можно ознакомиться в [Руководстве по программированию](#).

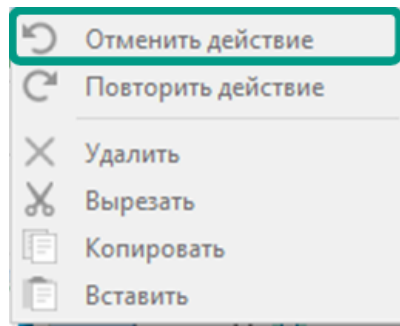



Команды редактирования расположены на панели инструментов редактора и дублируются в разделе «Правка» главного меню:

 - отменить выполненное действие

Альтернативные варианты:

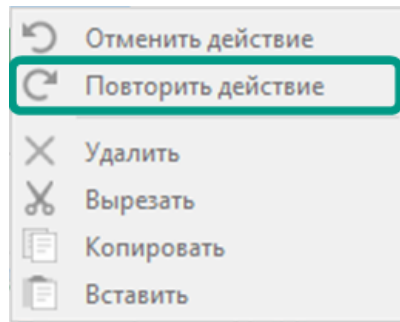
- выбрать пункт меню «Правка → Отменить действие»
- нажать комбинацию клавиш «Ctrl + Z»
- нажать правой кнопкой мыши в редакторе и из контекстного меню выбрать пункт «Отменить действие»




 - повторить отмененное действие

Альтернативные варианты:

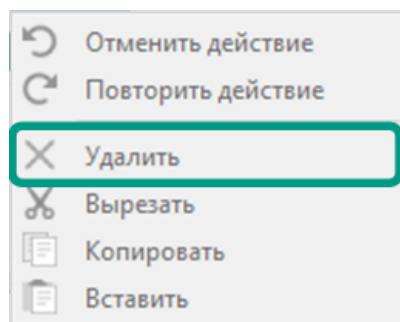
- выбрать пункт меню «Правка → Повторить действие»
- нажать комбинацию клавиш «Ctrl + Y»
- нажать правой кнопкой мыши в редакторе и из контекстного меню выбрать пункт «Повторить действие»




 - удалить выделенный в редакторе фрагмент

Альтернативные варианты:

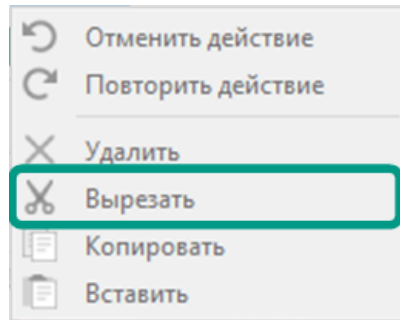
- выбрать пункт меню «Правка → Удалить»
- нажать клавишу «Delete»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «Удалить»



 - вырезать (переместить) выделенный в редакторе фрагмент в буфер обмена

Альтернативные варианты:

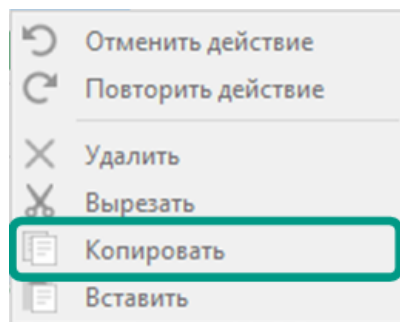
- выбрать пункт меню «*Правка → Вырезать*»
- нажать комбинацию клавиш «*Ctrl + X*»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «*Вырезать*»



- копировать выделенный в редакторе фрагмент в буфер обмена

Альтернативные варианты:

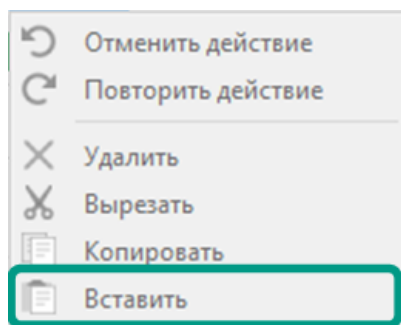
- выбрать пункт меню «*Правка → Копировать*»
- нажать комбинацию клавиш «*Ctrl + C*»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «*Копировать*»



- вставить в редактор фрагмент из буфера обмена

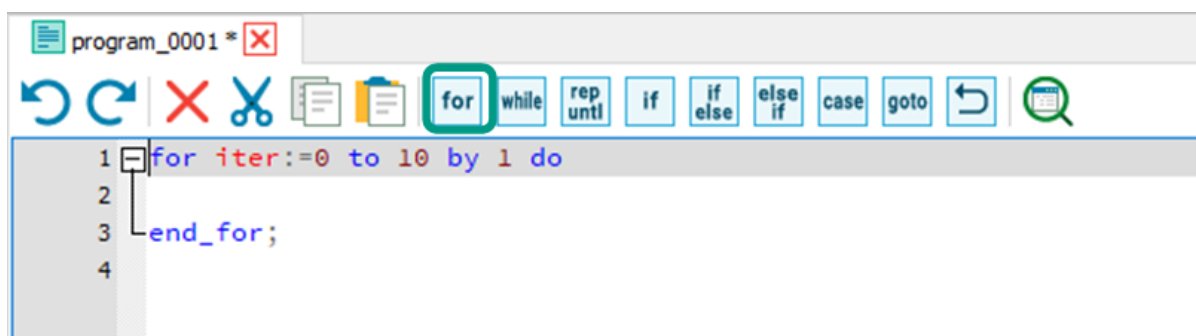
Альтернативные варианты:

- выбрать пункт меню «*Правка → Вставить*»
- нажать комбинацию клавиш «*Ctrl + V*»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «*Вставить*»



Команды для вставки операторов расположены только на панели инструментов редактора ST. Конструкция выбранного оператора будет вставлена в редактор на позицию курсора.

for - оператор цикла *FOR* обеспечивает заданное количество повторений группы выражений, расположенных в теле цикла.



Оператор FOR выполняется согласно следующим правилам:

- перед выполнением цикла счетчику присваивается начальное значение. В качестве счетчика можно использовать только целочисленную переменную.
- начальное, конечное значения и шаг цикла могут быть как константами, так и выражениями.
- после каждого выполнения тела цикла счетчик изменяется на значение (шаг), указанное после ключевого слова BY. По умолчанию, если ключевое слово BY и шаг цикла не заданы, счетчик увеличивается на 1. При этом значение счетчика может как увеличиваться, так и уменьшаться, если после ключевого слова BY задано отрицательное целочисленное значение.
- тело цикла выполняется циклически до тех пор, пока счетчик не превысит конечное значение.

Рекомендации по использованию цикла FOR:

- не изменяйте счетчик цикла и конечное значение в теле цикла, при необходимости используйте их только для чтения;
- не задавайте в качестве конечного условия максимальное для типа переменной счетчика значение, в противном случае цикл станет бесконечным.

while - оператор цикла *While* обеспечивает выполнение группы выражений в теле цикла, пока верно условное логическое выражение.

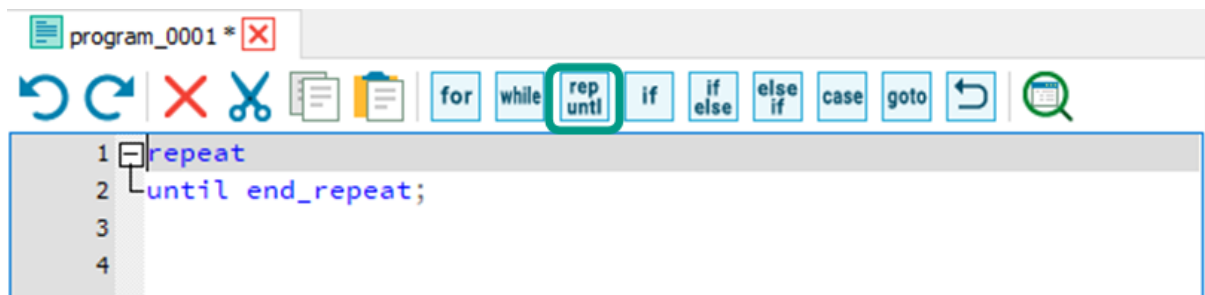


```
program_0001 * [X]
while
for while rep until if if else if case goto
1 while iter<10 do
2
3 end_while;
4
```

Оператор While выполняется согласно следующим правилам:

- условное выражение проверяется до начала цикла.
- тело цикла выполняется циклически до тех пор, пока верно (равно true) условное выражение.
- при ложном условном выражении происходит выход из цикла, а тело цикла не выполняется.
- если условное выражение всегда верно, то цикл будет выполняться бесконечно, а при изначально ложном выражении не выполниться ни разу.

rep until - оператор цикла *Repeat* обеспечивает выполнение группы выражений в теле цикла до тех пор, пока верно условное логическое выражение.

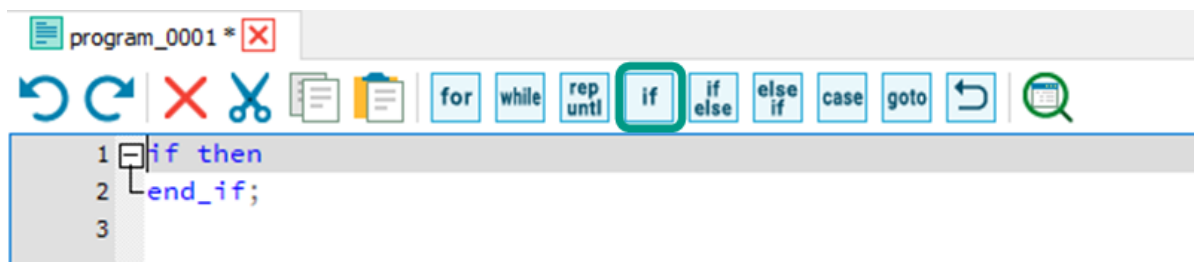


```
program_0001 * [X]
rep until
for while rep until if if else if case goto
1 repeat
2 until end_repeat;
3
4
```

Оператор Repeat выполняется согласно следующим правилам:

- условное выражение указывается после ключевого слова *until*.
- условное выражение проверяется после каждого выполнения тела цикла, поэтому выражения в теле цикла будут выполнены минимум 1 раз. При этом истинное условное выражение служит для выхода из цикла, а не для его продолжения.
- если условное выражение всегда ложно, цикл будет выполняться бесконечно.

if - условный оператор *IF* позволяет выполнить одну группу выражений, если верно условное логическое выражение.



```
program_0001 * [X]
[undo] [redo] [cancel] [cut] [copy] [for] [while] [rep until] [if] [if else] [else if] [case] [goto] [refresh] [search]
1 if then
2 end_if;
3
```

Оператор IF выполняется согласно следующим правилам:

- условное выражение задается после ключевого слова *if*.
- если условное выражение верно, то выполняются выражения, описанные после ключевого слова *then*.
- если условное выражение ложно, то управление переходит к следующему оператору.

if else - условный оператор *IF - ELSE* позволяет выполнить одну из двух групп операторов в зависимости от условного логического выражения.



```
program_0001 * [X]
[undo] [redo] [cancel] [cut] [copy] [for] [while] [rep until] [if] [if else] [else if] [case] [goto] [refresh] [search]
1 if then
2 else
3 end_if;
4
```

Оператор IF - ELSE выполняется согласно следующим правилам:

- условное выражение задается после ключевого слова *if*.
- если условное выражение истинно, то выполняются выражения, описанные после ключевого слова *then*.
- если условное выражение ложно, то выполняются выражения, описанные после ключевого слова *else*.

else if - условный оператор *IF - ELSIF - ELSE* позволяет выполнить различные группы выражений в зависимости от выполнения нескольких условных логических выражений.

```
1 if then
2   elsif
3   else
4 end_if;
5
```

Оператор IF - ELSIF - ELSE выполняется согласно следующим правилам:

- первое условное выражение задается после ключевого слова *if*
- если первое условное выражение истинно, то будут выполнены выражения, описанные после первого ключевого слова *then*
 - если первое условное выражение ложно, то по очереди будут проверяться условные выражения, заданные после ключевых слов *elsif*
 - если одно из условных выражений, следующих после *elsif* верно, то будут выполнены выражения, описанные после соответствующего ключевого слова *then*
 - если все заданные условия ложны, то будут выполнены выражения, описанные после ключевого слова *else*
- выполнение оператора завершается после выполнения любого блока *then* или блока *else*.

case - оператор выбора **CASE** позволяет выполнить различные инструкции, в зависимости от значения одной целочисленной переменной.


```
1 case of
2   :
3 end_case;
4
5
```

Оператор CASE выполняется согласно следующим правилам:

- целочисленная переменная указывается после ключевого слова **CASE**
- после ключевого слова *of* в столбик указывается список возможных значений и соответствующих им выражений, разделенные друг от друга двоеточием
 - если значение переменной совпало с одним из значений в списке, то выполняется соответствующее выражение

- если значение переменной не совпало ни с одним значением из списка, то выполняются выражения, описанные после ключевого слова *else*
- чтобы одно и то же выражение выполнялось при различных значениях переменной, необходимо перечислить эти значения в списке через запятую
- чтобы одно и то же выражение выполнялось для диапазона целых значений, необходимо указать начальное и конечное значения, разделенные двумя точками.

Подробное описание приведено в [Руководстве по программированию](#).

 - оператор безусловного перехода **GOTO** позволяет перевести выполнение программы к оператору, помеченному меткой.



Действие оператора перехода состоит в том, что после его исполнения следующими будут исполняться операторы программы, идущие в тексте непосредственно после метки, а операторы в промежутке между ключевым словом *goto* и меткой будут пропущены.

Правила использования меток:

- метка является уникальным идентификатором в пределах одной программы;
- имя метки может состоять только из латинских букв и совпадать с именем переменной;
- метки не инициализируются в словарях и могут быть расположены на любой строке программы;
- после имени метки ставится символ двоеточия, затем следует оператор, который будет выполнен при переходе к метке;
- для перехода к метке используется ключевое слово *goto*, после которого идет имя метки;
- количество меток в программе не ограничено.



Будьте внимательны при использовании меток. При переходе к метке, расположенной до оператора *goto* может произойти зацикливание программы.



- оператор **Return** позволяет закончить выполнение текущей программы.



При выполнении оператора *Return* осуществляется выход из текущей программы проекта и переход к выполнению следующей, при этом все следующие после данного оператора выражения в текущей программе останутся не выполненными.

Служебные команды



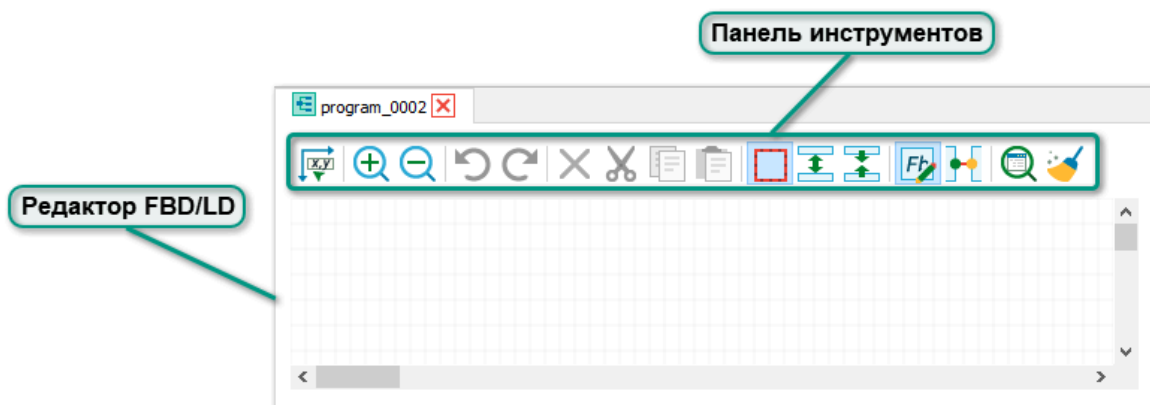
-Проверить

Запускает проверку на наличие ошибок в написанном коде. Результаты проверки будут выведены во вкладке *Вывод*, а ошибки, при их наличии, будут выведены во вкладке «Список ошибок».

4.2.3 Редактор FBD/LD

Редактор FBD/LD включает в себя необходимый набор инструментов для создания и редактирования программных объектов (программ и функциональных блоков) на языках программирования FBD/LD. Редактор поддерживает произвольную расстановку графических элементов по выбору пользователя в едином графическом поле, а траектория соединительных линий выстраивается автоматически после выбора элемента-источника и элемента приемника. Соединение элементов может осуществляться как отдельными кликами мыши, так и в режиме "нажал-переместил". В режиме соединения у графических блоков цветом выделяются выходы и входы, которые могут быть соединены. Выбор элементов для вставки на схему производится на панели «Добавить/Изменить».

При программировании в среде разработки Unimod PRO 2 могут использоваться блоки из системной библиотеки с заранее запрограммированной логикой выполнения и пользовательские блоки. Подробнее с системными блоками можно ознакомиться в [Руководстве по программированию](#). На следующем рисунке изображен общий вид редактора FBD/LD.



Команды редактирования расположены на панели инструментов редактора и дублируются в разделе «Правка главного меню»:

 - отменить выполненное действие

Альтернативные варианты:

- выбрать пункт меню «Правка → Отменить действие»
- нажать комбинацию клавиш «Ctrl + Z»

 - повторить отмененное действие

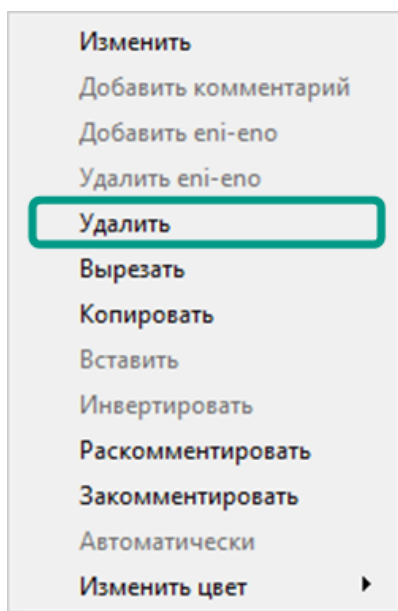
Альтернативные варианты:

- выбрать пункт меню «Правка → Повторить действие»
- нажать комбинацию клавиш «Ctrl + Y»

X - удалить выделенный в редакторе элемент или фрагмент

Альтернативные варианты:

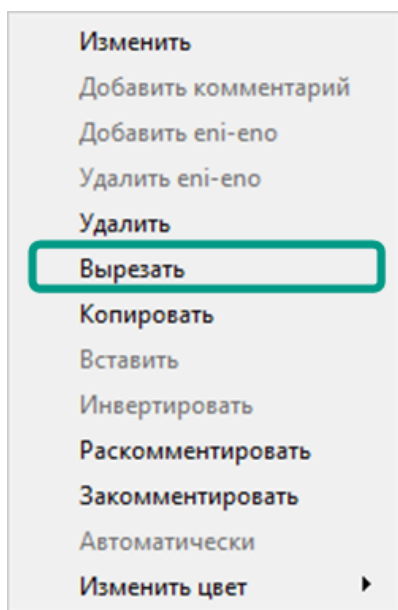
- выбрать пункт меню «*Правка → Удалить*»
- нажать клавишу «*Delete*»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «*Удалить*»



✂ - вырезать (переместить) выделенный в редакторе элемент или фрагмент в буфер обмена

Альтернативные варианты:

- выбрать пункт меню «*Правка → Вырезать*»
- нажать комбинацию клавиш «*Ctrl + X*»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «*Вырезать*»

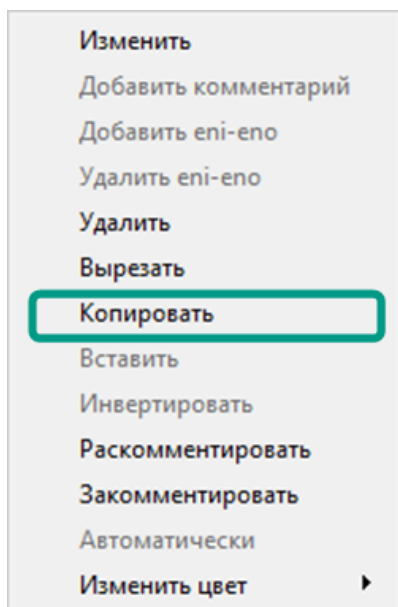




- копировать выделенный в редакторе элемент или фрагмент в буфер обмена

Альтернативные варианты:

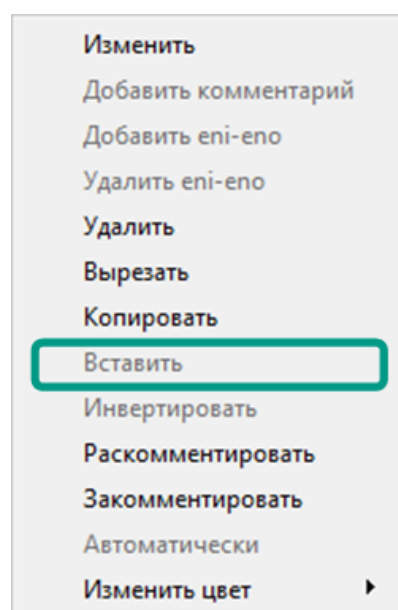
- выбрать пункт меню «*Правка → Копировать*»
- нажать комбинацию клавиш «*Ctrl + C*»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «*Копировать*»



- вставить в редактор элемент или фрагмент из буфера обмена

Альтернативные варианты:

- выбрать пункт меню «*Правка → Вставить*»
- нажать комбинацию клавиш «*Ctrl + V*»
- нажать правой кнопкой мыши по выделенному фрагменту и из контекстного меню выбрать пункт «*Вставить*»



Команды выбора режима расположены на панели инструментов редактора FBD/LD:



- Режим редактирования

Режим редактирования позволяет управлять элементами в редакторе (добавление, перемещение, удаление). Редактирование связей между элементами в данном режиме недоступно.



- Режим соединения

Режим соединения позволяет добавлять связи между элементами в редакторе. Добавление, перемещение и удаление элементов в данном режиме недоступно.



- Режим выделения

Режим выделения позволяет нажатием левой кнопки мыши выделять элементы (блоки и связи) в редакторе. Контуров всех выделенных элементов в редакторе FBD/LD будут выделены красным цветом.

Для выделения нескольких элементов при зажатой левой кнопке мыши выделите произвольную область в редакторе либо кликните мышью по требуемым элементам при зажатой клавише *Ctrl*. Для выделения всех элементов в редакторе нажмите сочетание клавиш *Ctrl + A*.

Перемещение выделенных элементов осуществляется с использованием мыши при зажатии ее левой кнопки.



- Режим Раздвинуть

Данный режим позволяет раздвинуть элементы в редакторе. Размер добавляемого между элементами пространства задается в редакторе путем выделения добавляемого пространства между элементами при зажатой левой кнопке мыши.




- Режим Сдвинуть

Данный режим позволяет сдвинуть элементы в редакторе. Размер удаляемого между элементами пространства задается в редакторе путем выделения удаляемого пространства при зажатой левой кнопке мыши.

Служебные команды расположены только на панели инструментов редактора FBD/LD:



- Показать координаты сетки

Позволяет в редакторе около курсора отображать координаты сетки. Для прекращения отображения координат нажмите на команду .



- Увеличить масштаб

Позволяет увеличить масштаб отображения сетки редактора и расположенных на ней элементов.



- Уменьшить масштаб

Позволяет уменьшить масштаб отображения сетки редактора и расположенных на ней элементов.



- Проверить

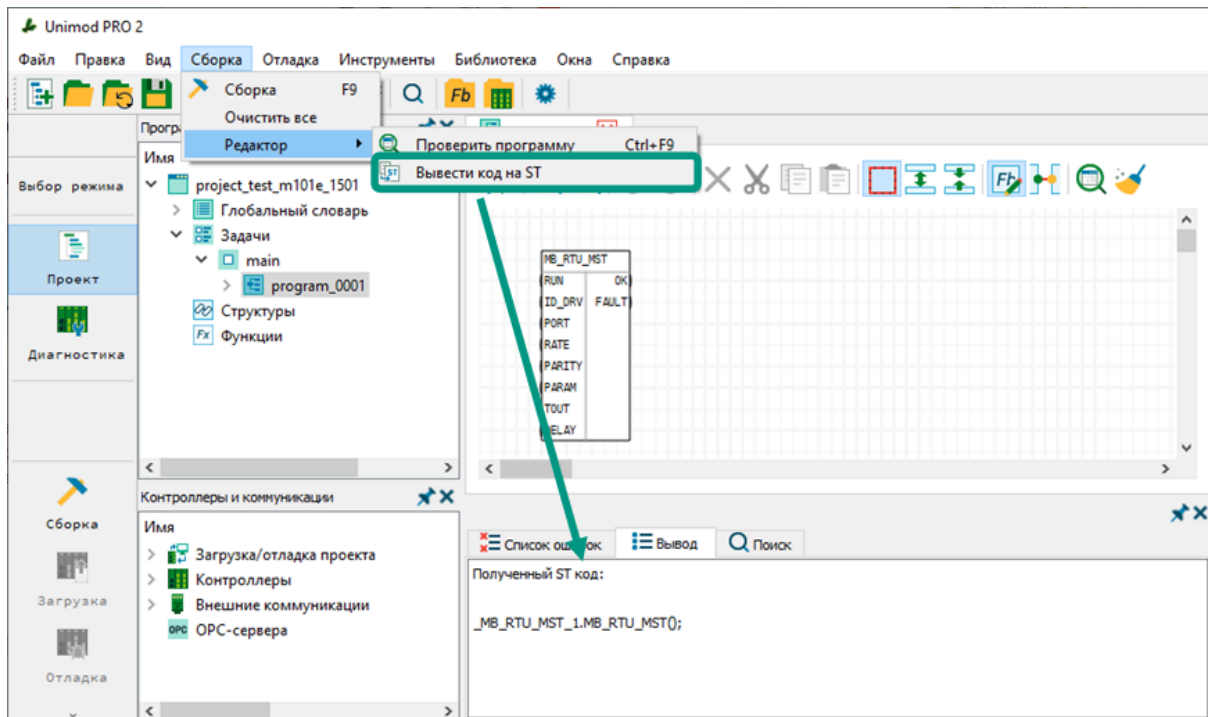
Запускает проверку на наличие ошибок в созданной схеме. Результаты проверки будут выведены во вкладке *Вывод*, а ошибки, при их наличии, будут выведены во вкладке *Ошибки*.




- Очистить неиспользуемое из словаря

Позволяет удалить все неиспользуемые в редакторе FBD/LD элементы из Локального словаря программы или Словаря данных структуры.

Код программы FBD/LD может быть выведен в виде кода на языке ST. Для этого при открытом редакторе FBD/LD выберите пункт меню «Сборка → Редактор → Вывести код на ST». После чего на вкладке «Вывод» будет выведен соответствующий код на языке ST.





4.2.4 Управление графическими элементами FBD/LD


Для выделения блока или линии связи на панели инструментов выберите команду  (режим «Редактирования»), так как в других режимах выделение невозможно. Выделение одного элемента осуществляется нажатием левой кнопки мыши.

Для открытия контекстного меню блока или линии связи необходимо выделить требуемый элемент, после чего кликнуть правой кнопкой мыши по выделенному элементу.

Добавление элементов:

- **добавление блока:** откройте панель «Добавить/Изменить» → найдите и добавьте элемент в редактор путем перетаскивания выделенного элемента или символа ). Подробное описание панели «Добавить/Изменить» приведено в разделе [Инструментальные панели](#). Добавить блок в редактор можно в любом режиме работы.
- **добавление комментария:** из контекстного меню редактора выберите пункт «Добавить комментарий», после чего введите его текст. Изменение текста комментария осуществляется после двойного клика по нему левой кнопкой мыши. При этом в редакторе FBD/LD любой элемент можно *Закомментировать/Раскомментировать*, выбрав из контекстного меню соответствующий пункт. Закомментированный элемент отображается серым цветом.
- **добавление линии связи:** на панели инструментов выберите команду  для переключения в режим «Соединение» → нажатием левой кнопки мыши выберите выходной и входной каналы соединяемых блоков → между соединенными блоками будет создана линия связи, при перемещении блоков линия связи будет перерисовываться автоматически.


Изменение элементов:

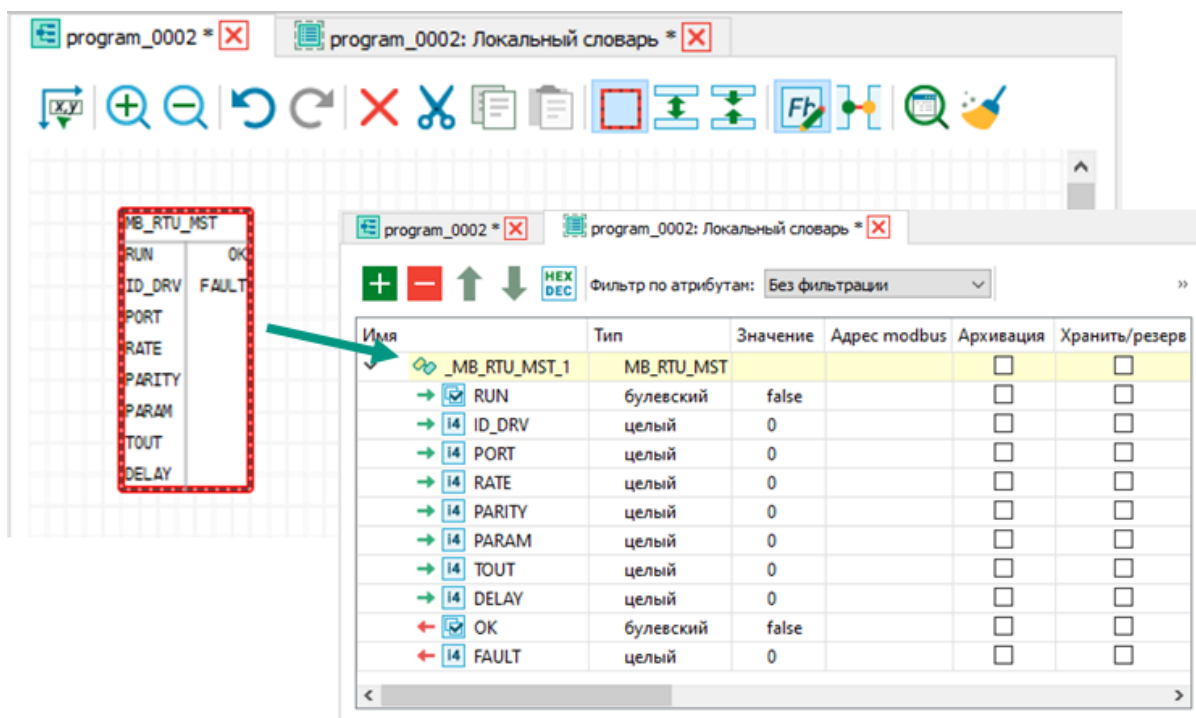
- **изменение названия и содержимого блока:** откройте контекстное меню и выберите пункт «Изменить» → на панели «Добавить/Изменить» внесите изменения → подтвердите изменения нажатием на символ .
- **добавление или удаление дополнительных логических входов (eni) и выходов (eno):** откройте контекстное меню и выберите пункт «Добавить eni-eno» или «Удалить eni-eno». Используются при программировании на языке LD.

- **изменение линии связи:** удалите существующую линию связи и добавьте новую, перерисовка линии связи происходит автоматически при перемещении блоков.
- **инвертирование линии связи:** откройте контекстное меню и выберите пункт «Инвертировать».
- **изменение цвета элемента:** откройте контекстное меню элемента → нажмите на пункт меню «Изменить цвет» → выберите один из предложенных вариантов.

Удаление элементов:

- выделите и удалите элементы одним из вышеуказанных способов. При удалении блока все соединенные с ним линии связи будут автоматически удалены.

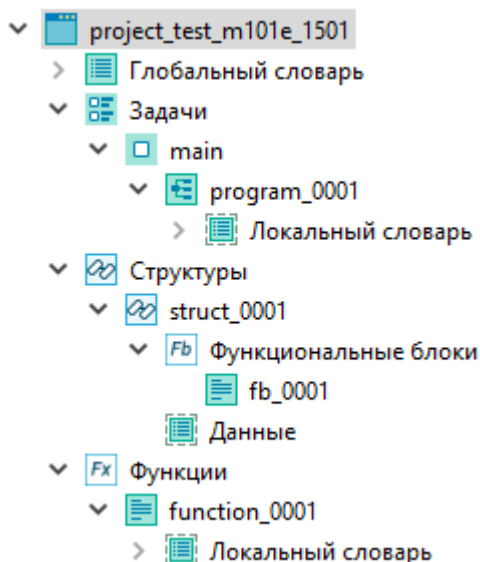
При добавлении графического блока в редактор в связанном с ним словаре автоматически инициализируется переменная. Но при удалении блока, связанная с ним переменная из словаря не удаляется, для этого нужно в редакторе FBD/LD нажать команду  (очистить неиспользуемое из словаря). Для добавления доступны как блоки системной библиотеки, так и пользовательские блоки, соответствующие переменным, структурам или программным объектам.



5 Состав проекта

Проект технологического приложения включает в себя набор программных объектов, словарей, структур для хранения данных и организации взаимосвязи переменных с каналами ввода-вывода. После загрузки в контроллер разработанный проект будет выполняться под управлением исполнительной системы Unimod PRO 2.

Структура проекта отображается в виде дерева проекта во вкладке «Программы и данные» рабочего пространства среды разработки.



Правила построения дерева проекта:

- минимально допустимая структура проекта состоит из пустого глобального словаря, списка задач с единственной задачей «*main*», пустых списков структур и функций;
- элементы минимальной структуры проекта создаются по умолчанию при создании проекта и не могут быть удалены;
- в пределах проекта создается только один глобальный словарь;
- в списке задач присутствует одна целевая задача «*main*», добавление не доступно;
- количество программ, дочерних программ, структур и функций ограничено памятью контроллера;
- для каждой программы и структуры может создаваться произвольное количество дочерних программ и функциональных блоков соответственно;
- локальные словари добавляются к каждой программе, структуре и функции по умолчанию.

Работа с элементами дерева проекта описана в следующих подразделах настоящего руководства.

5.1 Переменные

Разработанное в Unimod PRO 2 приложение представляет собой набор операций над данными, поступившими в контроллер и размещенными в его памяти. Для хранения значений параметров технологического процесса (данные об уровне жидкости, температуре, давлении и т.д.) в контроллере выделяется определенный объем памяти, а для идентификации сохраненного значения используется физический адрес – уникальный идентификатор, который указывает на конкретное место в памяти контроллера, где храниться значение параметра.

Технологическое приложение обращается к сохраненным в памяти контроллера значениям по их физическим адресам и в зависимости от значения выполняет требуемые операции (открытие/закрытие клапанов, включение / выключение приборов или предупреждающих сигналов и т.д.).

Для упрощения работы с памятью сохраненные в контроллере данные представляются в среде разработки Unimod PRO 2 в виде переменных – именованных объектов с определенными характеристиками (атрибутами). Во время объявления переменной в памяти контроллера выделяется требуемый объем памяти в зависимости от типа данных, а обращение к переменной осуществляется по заданному разработчиком имени.

Для хранения переменных в среде разработки Unimod PRO 2 предусмотрены три вида словарей: глобальный, локальный и словарь данных структуры. В каждый программный объект включен один локальный словарь (для структуры - словарь данных структуры), который создается при добавлении программного объекта в проект. А глобальный словарь в проекте один, он создается по умолчанию в отдельной ветке дерева проекта. Основные отличия словарей заключаются в области видимости и составе атрибутов их переменных, а также в доступных для использования фильтрах.

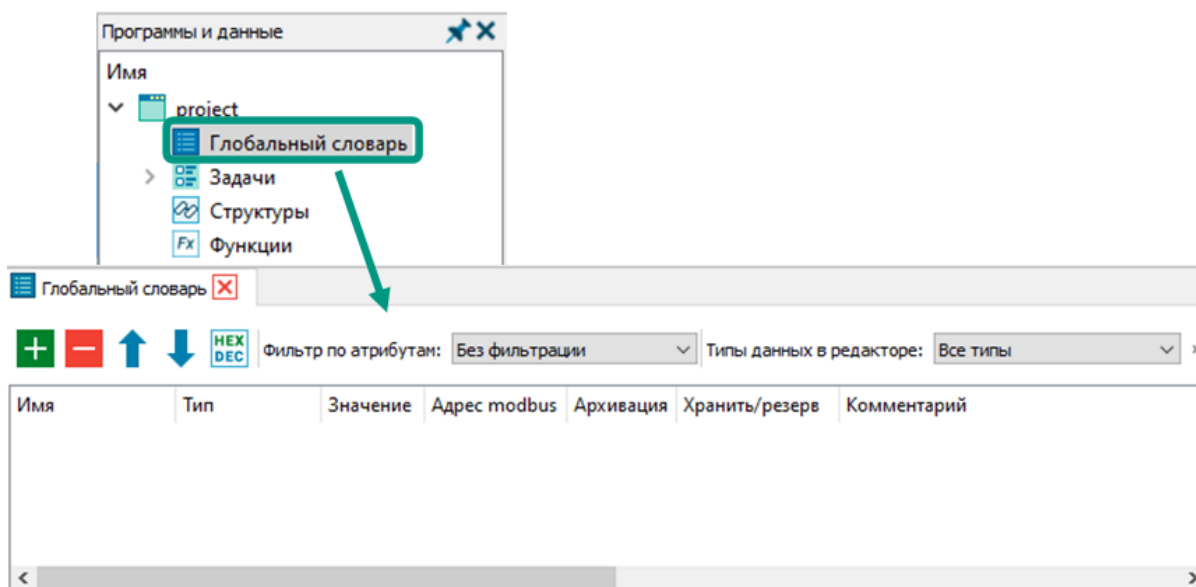
Краткое описание назначения и особенностей словарей:

- **глобальный словарь** - создаются переменные, которые будут доступны для чтения и записи из любого программного объекта текущего проекта. Переменные глобального словаря в основном используются для хранения системных параметров, которые могут потребоваться в разных элементах проекта. Для хранения параметров технологического процесса использовать глобальные переменные не рекомендуется, так как при этом затрудняется понимание логики передачи данных между программными объектами, что может усложнить отладку и поддержку приложения.
- **локальный словарь** - создаются переменные с ограниченной областью видимости. Локальные переменные, в отличие от глобальных, доступны для использования только в рамках программного объекта, в словаре которого они созданы. При этом локальные переменные дочерних программ,

функций и функциональных блоков могут использоваться в качестве входных и выходных параметров, либо в качестве вспомогательных внутренних переменных.

- **словарь данных структуры** - предназначен для управления списком содержащихся в структуре переменных. Созданные переменные напрямую доступны из функциональных блоков той же структуры, а для внешнего доступа требуется предварительно объявить переменную структурного типа в глобальном или локальном словаре. В основном используются для привязки к каналам ввода-вывода и обмена данными в задачах внешних коммуникаций.

Добавление, хранение, редактирование и удаление переменных, их экспорт и импорт осуществляются с использованием редактора словаря. Для открытия редактора раскройте список элементов проекта нажатием на символ \triangleright → дважды кликните левой кнопкой мыши по выбранному словарию в дереве проекта или из его контекстного меню выберите пункт «Открыть» → редактор будет открыт в рабочем пространстве среды разработки в виде вкладки.



Подробно работа с переменными в редакторе словаря описана в разделе [Редактор словаря](#) настоящего руководства.

5.1.1 Атрибуты переменных

Атрибуты переменных в среде разработки Unimod PRO 2:

Атрибут	Описание
Имя	<p>Используется для идентификации переменной и обращения к ней из текста программных объектов.</p> <p>Имя переменной должно начинаться с буквы и содержать только латинские буквы, цифры и символы подчеркивания, регистр символов в имени не имеет значения (например, имена param и PaRam идентичны и будут указывать на одну и ту же переменную). Длина имени не должна превышать 22-х символов. Имена переменных в разных словарях могут совпадать, в таком случае приоритет будут иметь локальные переменные или переменные структуры.</p> <p>Для создания массива после имени переменной в квадратных скобках нужно указать размер массива - целое положительное число. Поддерживаются одномерные, двумерные и трехмерные массивы, размер каждого измерения указывается в квадратных скобках через запятую. Например:</p> <p><i>var_1[10]</i> - одномерный массив; <i>var_2[10,10]</i> - двумерный массив; <i>var_3[10,10,10]</i> - трехмерный массив.</p> <p>Нумерация элементов массива начинается с нуля.</p>
Тип данных	<p>Определяет размер выделенной памяти, диапазон возможных значений и набор допустимых операций над переменной.</p> <p>Поддерживаются типы данных, предусмотренные стандартом IEC 61131-3, а также их список дополнен 64-разрядным типом данных DOUBLE - вещественное число с двойной точностью. Подробная информация о типах данных переменных приведена в документе <u>Руководство по программированию</u>.</p>

<p>Значение</p>	<p>Содержит начальное значение переменной.</p> <p>Если значение установлено пользователем, то оно присваивается переменной при каждом перезапуске приложения, а восстановление переменной даже с выбранным атрибутом «<i>Хранить/Резерв</i>» не выполняется. Для использования возможности восстановления значений, установите атрибут «<i>Хранить/Резерв</i>», а начальное значение оставьте по умолчанию.</p>
<p>Адрес Modbus</p>	<p>Используется для идентификации переменной при передаче по протоколу Modbus. Для ввода адреса дважды щелкните левой кнопкой мыши по атрибуту «<i>Адрес modbus</i>».</p> <p><i>Discrete input</i> - входное дискретное значение; <i>Coil</i> - выходное дискретное значение. <i>Input register</i> - входное аналоговое значение (байтовый, целочисленный и вещественные типы данных); <i>Holding register</i> - выходное аналоговое значение; <i>Input/Input registers</i> - группа входных дискретных или аналоговых значений; <i>Coils/Holding registers</i> - группа выходных дискретных или аналоговых значений;</p> <p>Modbus-адреса назначаются отдельным переменным и переменным структурного типа данных. Для структурных переменных указывается начальный Modbus-адрес, а всем дочерним переменным Modbus-адреса назначаются по порядку в зависимости от их типа данных.</p> <p>Для логических переменных адреса задаются последовательно, для целых и вещественных – через один. Если предполагается чтение или запись 32-х разрядных чисел, то Modbus-адреса переменных должны идти через один (два регистра на одну переменную), а для 64-х разрядных чисел - через 4 (четыре регистра на одну переменную). Рекомендуется для каждого типа данных выделять отдельный блок Modbus-адресов.</p>

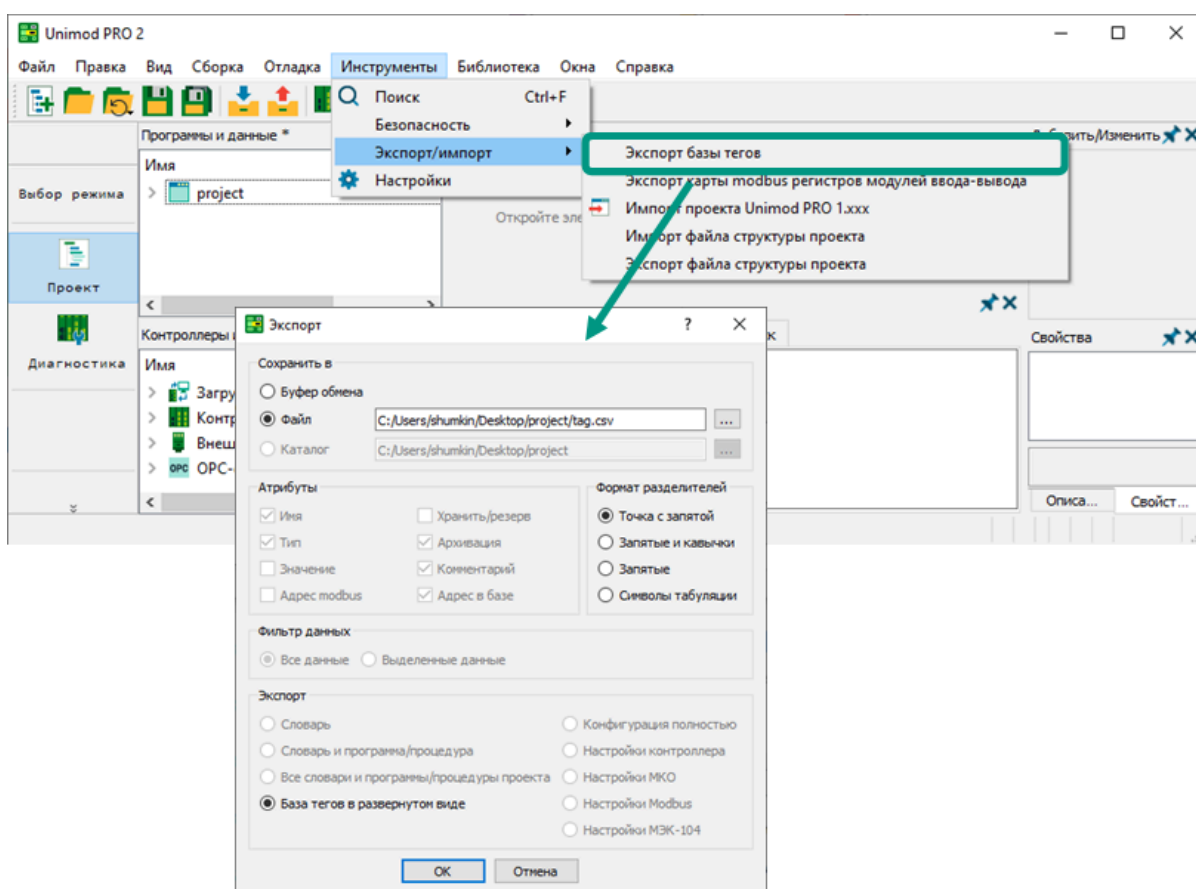
Архивация	<p>Позволяет мастер-модулю хранить значения переменной в буфере за определенный промежуток времени. В дальнейшем эти значения могут быть считаны из мастер-модуля с помощью запросов OPC HDA или по протоколу МЭК-104. Подробное описание приведено в документе Архивация переменных.</p>
Атрибут	<p>Определяет правила доступа к значениям переменной, доступен для выбора только в словаре данных структуры.</p> <p>Возможные значения:</p> <ul style="list-style-type: none"> • <i>внутренняя</i> - значения переменных доступны для чтения и записи только в функциональных блоках той структуры, в которой объявлены, и не могут быть перезаписаны или прочитаны за пределами структуры. Используются в основном для временного хранения значений вычислительных операций. Такие переменные не могут использоваться контроллером для обмена входными и выходными данными. • <i>входная</i> - доступны только для записи в них значений, которые поступили от входных каналов либо в процессе внешних коммуникаций. • <i>выходная</i> - доступны только для чтения, их значения могут быть связаны с выходными каналами и с задачами внешних коммуникаций. • <i>вход/выход</i> - значения переменной доступны для чтения и записи одновременно.
Хранить/резерв	<p>Позволяет сохранять значения переменной в энергонезависимой памяти контроллера и восстанавливать их при перезапуске приложения.</p> <p>При установленном значении атрибута текущее значение переменной сохраняется в энергонезависимой памяти контроллера и не теряется даже при потере питания контроллером. После перезапуска приложения, вызванного сбоем в работе контроллера или потерей питания, переменной будет присвоено ее последнее значение, сохраненное до перезапуска. Если у переменной установлено начальное значение, то восстановление значений не выполняется.</p>

Комментарий	Предназначен для краткого текстового описания назначения переменной.
Offset	<p>Уникальный идентификатор (индекс) переменной в пределах всего проекта.</p> <p>Индекс используется исполнительной системой для идентификации переменных, а также для соотнесения переменных приложения с базой тегов SCADA-систем (при использовании в SCADA-системе идентификации не по именам переменных, а по их индексам). Значение данного атрибута устанавливается автоматически в порядке возрастания при добавлении переменной в словарь. При удалении переменных, их индексы также удаляются, образуя диапазон незанятых значений. При большом количестве переменных наличие незанятых диапазонов приводит к нерациональному расходованию памяти контроллера. Для оптимизации занятой индексами памяти используется переиндексация, в результате которой всем переменным приложения будут назначены новые значения индекса сплошным списком. Для этого выберите пункт меню <i>«Отладка → Переиндексация базы переменных»</i>. Используйте переиндексацию переменных с осторожностью, так как это может вызвать ошибки в работе взаимосвязанных с контроллером систем, использующих индексы в качестве идентификаторов переменных.</p>

5.1.2 Экспорт тегов

Под тегами в среде разработки Unimod PRO 2 понимаем переменные с определенным набором атрибутов. Выгрузка тегов применяется для последующей привязки переменных технологического приложения к элементам SCADA-систем. Выгрузка всех переменных из глобального и локальных словарей осуществляется через экспорт базы тегов. Исключение составляют переменные словаря данных структуры, они не выгружаются.

Для экспорта базы тегов выберите пункт меню «Инструменты → Экспорт/импорт → Экспорт базы тегов», после чего в открывшемся окне настройте параметры экспорта с учетом указанных в конце настоящего раздела рекомендаций → нажмите на кнопку «Ок».




В результате в созданном файле будет содержаться список всех переменных приложения. Для каждой переменной из списка указаны ее имя, тип данных и индекс, значение атрибута архивации, количество сохраненных архивных значений, ограничения архивации по времени хранения и количеству, а также комментарий с кратким описанием назначения.

На следующем изображении в качестве примера приведен фрагмент экспортированных в файл тегов (переменных).

Name	Type	Offset	Archive	Archive count	Archive time filter	Archive value filter	Comment
var_0001.control.not_valid	bool	48	false	0	0	0	Признак ошибки/блокировки обмена
var_0001.control.err_cfg	int	49	false	0	0	0	Логическая ошибка обмена
var_0001.control.err_line1	int	53	false	0	0	0	Транспортная ошибка обмена по линии 1
var_0001.control.err_line2	int	57	false	0	0	0	Транспортная ошибка обмена по линии 2
var_0001.control.block_poll	bool	61	false	0	0	0	Блокировка обмена с модулем
var_0001.control.command	int	62	false	0	0	0	Резерв
var_0001.control.read_statistic	bool	66	false	0	0	0	Чтение блока statistic
var_0001.control.err_rw_statistic	int	67	false	0	0	0	Код завершения последнего асинхронного
var_0001.control.read_param	bool	71	false	0	0	0	Чтение блока param
var_0001.control.write_param	bool	72	false	0	0	0	Запись блока param
var_0001.control.err_rw_param	int	73	false	0	0	0	Код завершения последнего асинхронного
var_0001.statistic.Work_Time	int	177	false	0	0	0	Время наработки, сек
var_0001.statistic.Stbus_cnt_Err	int	181	false	0	0	0	Счетчик ошибок STBUS
var_0001.statistic.Reset_code	int	185	false	0	0	0	Код сброса
var_0001.statistic.Mod_temp	int	189	false	0	0	0	Температура модуля
var_0001.statistic.Stbus_err	bool	193	false	0	0	0	Ошибки связи с мастером
var_0001.statistic.Err_sequence	bool	194	false	0	0	0	Нарушение последовательности STBUS

Рекомендации по настройке параметров экспорта:

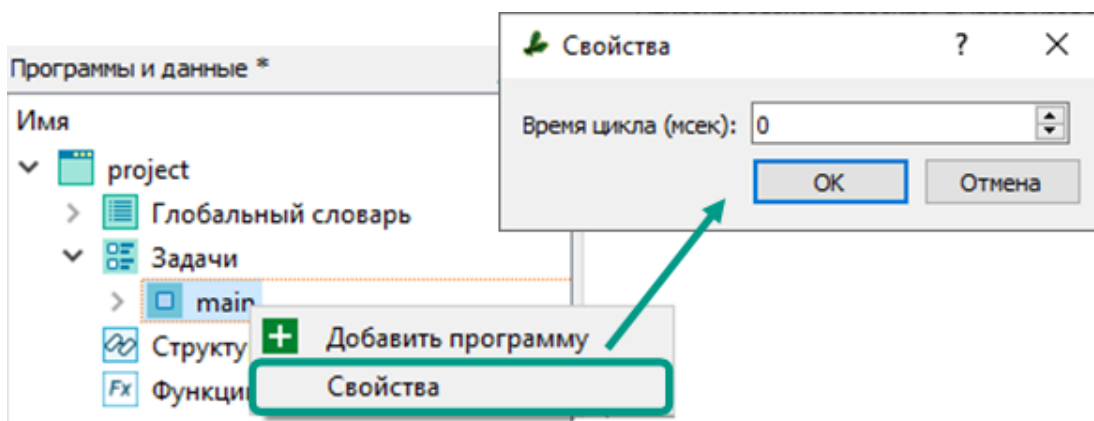
- для длительного хранения или переноса на сторонний ПК выберите пункт «Сохранить в файл», а для выбора каталога сохранения и указания имени файла нажмите кнопку .
- при указании имени файла с расширением «*.csv» экспортируемые данные будут доступны для просмотра и редактирования в табличном виде при открытии с использованием табличных редакторов.
- выбирайте пункт «Сохранить в Буфер обмена» только при необходимости переноса данных в пределах одного ПК. Сразу после экспорта рекомендуется осуществить импорт, чтобы исключить вероятность потери данных из-за перезаписи буфера обмена.
- не рекомендуется изменять формат разделителей, так как выбранный формат нужно будет запоминать и указывать в настройках импорта. При несовпадении формата разделителей в процессе импорта будет выведена ошибка. По умолчанию формат разделителя в настройках экспорта и импорта «Точка с запятой».

5.2 Задачи

В среде разработки Unimod Pro 2 может присутствовать только одна целевая задача. Она создается по умолчанию с именем «*main*» в разделе «*Задачи*» дерева проекта. Для целевой задачи могут создаваться одна или несколько независимых друг от друга программ, которые будут выполняться по очереди следования в списке задачи сверху вниз.

Целевая задача циклически выполняет следующие операции: сканирование входных каналов → выполнение программ → изменение выходных каналов. Циклом выполнения задачи считается период времени, по истечении которого будет повторно запущено выполнение задачи. Минимальный цикл выполнения задачи равен времени сканирования входных каналов, выполнения добавленных к задаче программ и изменения выходных каналов. По истечении времени одного цикла, задача выполняется сначала.

Для установки времени цикла задачи в дереве проекта нажмите правой кнопкой мыши по задаче → из контекстного меню выберите пункт «*Свойства*» → в открывшемся окне введите время в миллисекундах → нажмите кнопку «*Ок*».



При времени цикла равном 0 – задача будет выполняться за минимально возможный цикл, равный времени выполнения всех добавленных к задаче программ, после чего будет запущена повторно.

Если время цикла отлично от 0, но не превышает времени выполнения добавленных к задаче программ – задача все равно будет выполняться за минимально возможный цикл, равный времени выполнения всех добавленных программ, после чего будет запущена повторно.

Если время цикла отлично от 0 и превышает время выполнения добавленных к задаче программ – только тогда задача будет выполняться за указанное время цикла, а после окончания выполнения добавленных программ, задача перейдет в режим ожидания завершения цикла, затем запустится повторно.

5.3 Программы

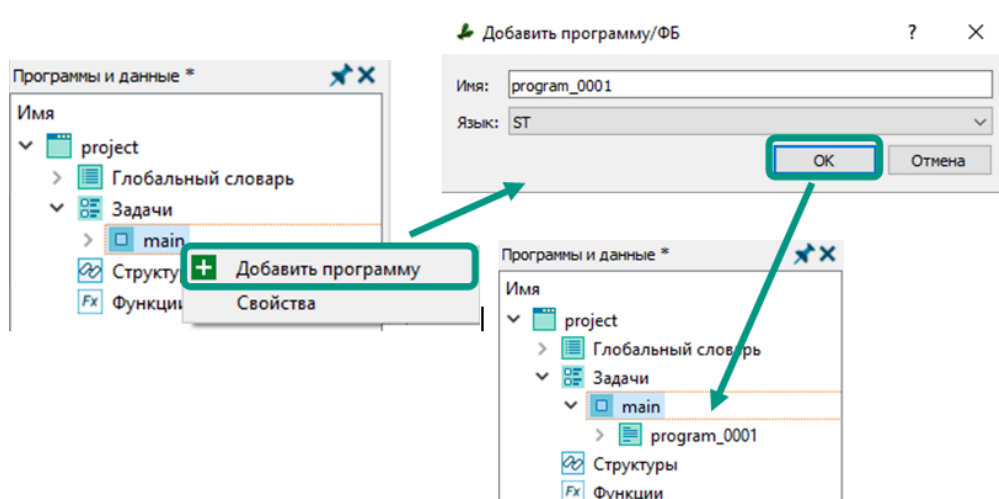
Программа - это программный объект, в котором описываются операции над переменными, содержащими значения параметров технологического процесса. Программы проекта не связаны между собой и не могут вызывать друг друга. Каждая программа может вызывать любую свою дочернюю подпрограмму, а также любую функцию или функциональный блок проекта. Перед вызовом функциональных блоков предварительно требуется в локальном словаре программы объявить переменную с соответствующим типом структуры. Дочерние программы и функции могут быть вызваны напрямую.

Каждая программа закрепляется за конкретной задачей. Все программы в каждом цикле задачи выполняются последовательно, в порядке их следования в списке. Порядок выполнения программ определяется пользователем. Подробнее про цикличность в разделе [Задачи](#).

Из программы можно обратиться к глобальным и локальным переменным текущей программы. Локальные переменные дочерних программ, функций и функциональных блоков из программ недоступны. Обмен данными с дочерними программами, функциями и функциональными блоками осуществляется через их входные и выходные параметры.

Выбор языка программирования ST или FBD/LD для разработки программы осуществляется при добавлении программы к задаче и не может быть изменен. Для создания программы на другом языке программирования нужно добавить новую программу с указанием требуемого языка разработки. По умолчанию для каждой программы создается единственный локальный словарь. Подробнее с назначением и возможностями локального словаря можно ознакомиться в разделе [Переменные](#).

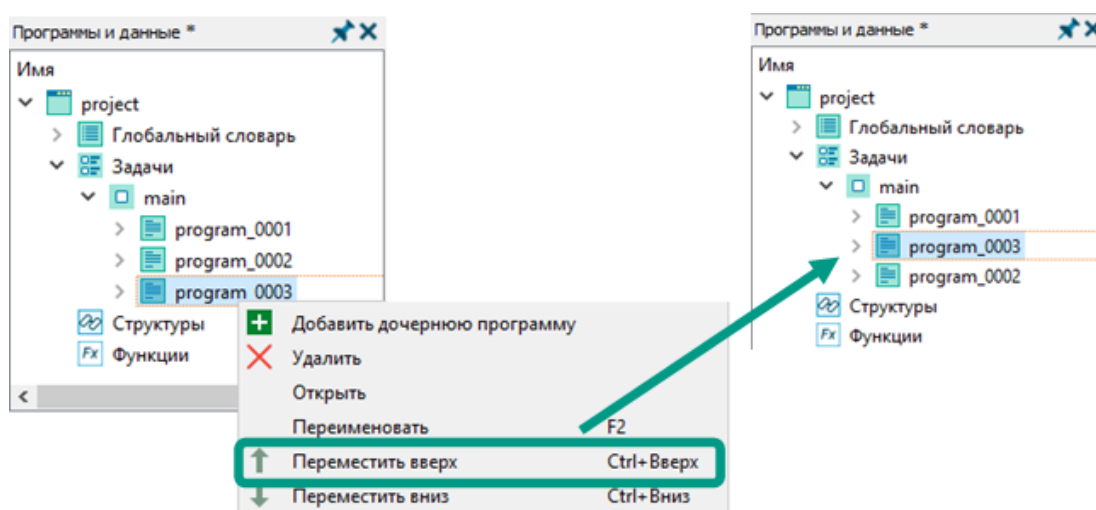
Для добавления программы к задаче щелкните по задаче правой кнопкой мыши → из контекстного меню выберите пункт «Добавить программу» → в открывшемся диалоговом окне введите имя программы и выберите язык программирования → нажмите «Ок» → новая программа будет добавлена к задаче в конец списка.



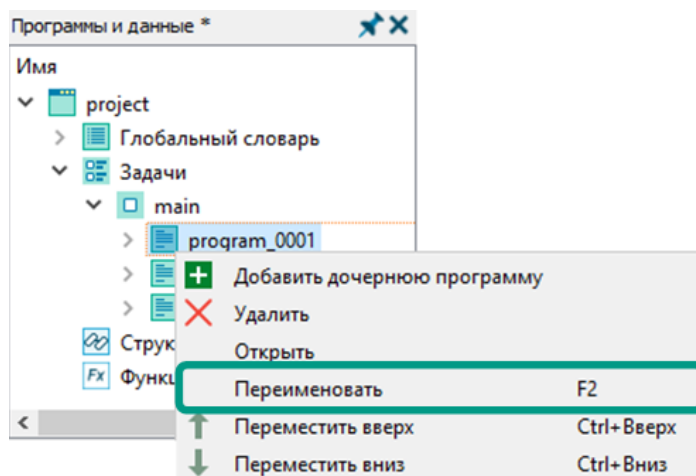
Имя программы должно соответствовать правилам:

- содержать только латинские буквы, цифры и символы подчеркивания;
- начинаться с латинской буквы;
- не превышать 128 символов.

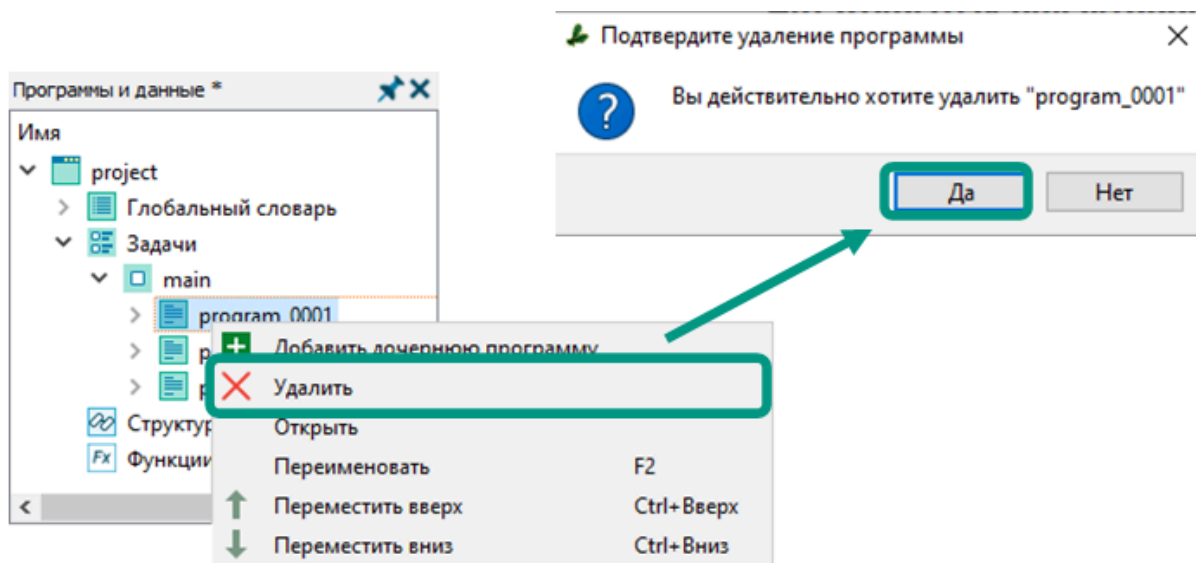
Для изменения порядка выполнения программ нажмите правой кнопкой мыши по программе в дереве проекта → из контекстного меню выберите пункт с направлением перемещения (переместить вверх или переместить вниз) или нажмите сочетание клавиш «*Ctrl+Вверх*»/«*Ctrl+Вниз*». Например, на следующем изображении для программы «*program_0002*» в контекстном меню выбран пункт «*Переместить вверх*», после чего порядок программ был изменен.



Для переименования программы нажмите правой кнопкой мыши по программе в дереве проекта и из контекстного меню выберите пункт «*Переименовать*» или выделите программу и нажмите клавишу «*F2*» → введите новое название программы → нажмите клавишу «*Enter*».



Для удаления программы нажмите правой кнопкой мыши по программе в дереве проектов → из контекстного меню выберите пункт «Удалить» → в открывшемся окне подтвердите удаление нажатием кнопки «Да».



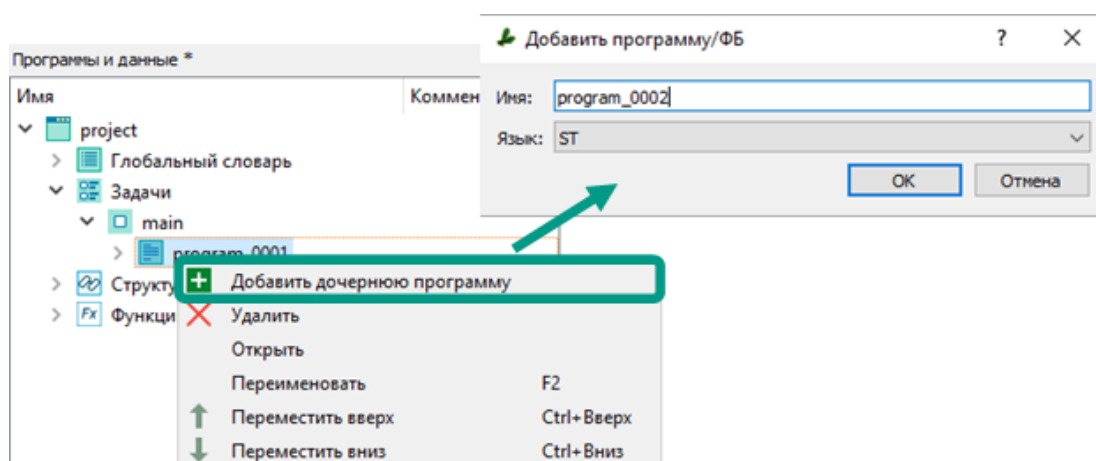
5.4 Дочерние программы

Дочерние программы являются разновидностью программных объектов и предназначены для написания алгоритмов на языках программирования ST или FBD/LD. В дочерней программе могут быть вызваны любые функциональные блоки структур и функции проекта, а сама дочерняя программа может быть вызвана только из программы, к которой добавлена. Перед вызовом функциональных блоков предварительно требуется в локальном словаре дочерней программы объявить переменную структурного типа. Функций могут быть вызваны напрямую.

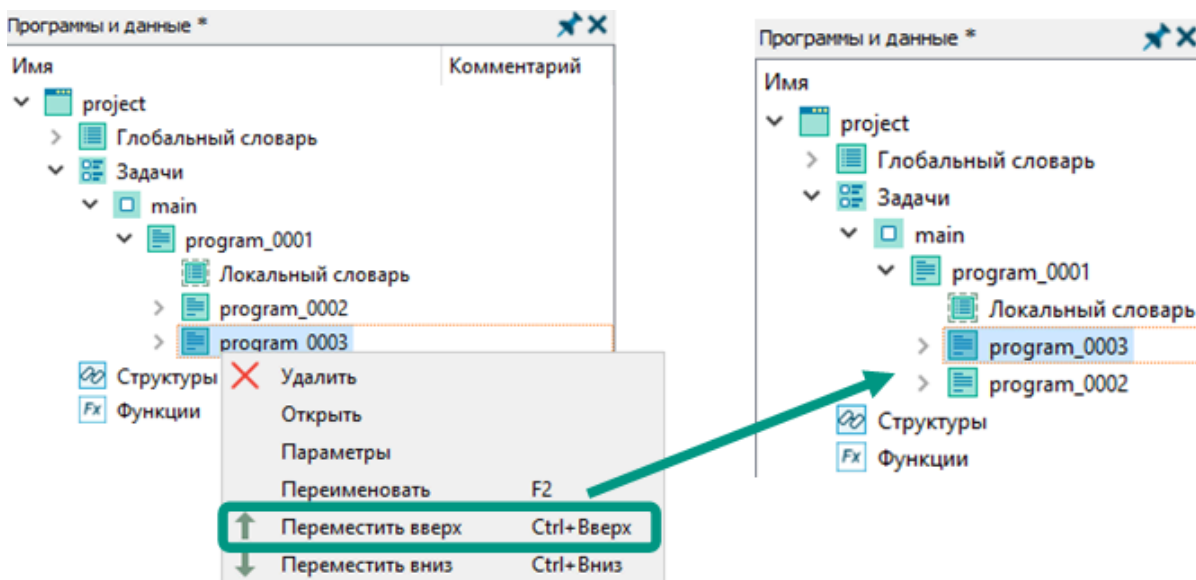
Из дочерней программы можно обратиться к глобальным и локальным переменным текущей дочерней программы, а также к локальным переменным родительской программы. При совпадении имен переменных дочерней программы с именами глобальных или локальных переменных родительской программы, приоритетом обладают переменные дочерней программы. Использование типов данных в словаре дочерней программы не ограничено.

Данные в дочернюю программу передаются через входные параметры при ее вызове в программе. Вызов дочерней программы осуществляется через конструкцию вида: *<имя_дочерней_программы>(<параметр1>, <параметр2>..., <параметрN>)*. После выполнения дочерняя программа обязательно должна вернуть результат в виде одного выходного значения. Для этого используется выходная переменная, которая по умолчанию включается в локальный словарь дочерней программы после ее добавления в проект. Внутренние локальные переменные не доступны за пределами дочерней программы и используются для временного хранения промежуточных результатов вычислений.

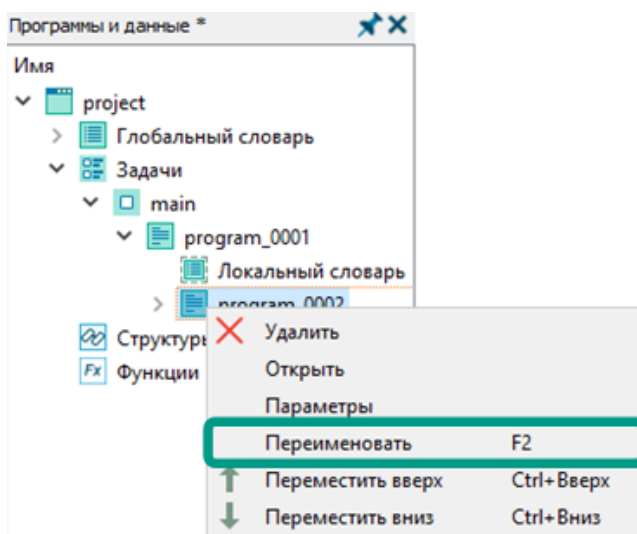
Для добавления дочерней программы в дереве проекта кликните правой кнопкой мыши по программе → из контекстного меню выберите пункт «*Добавить дочернюю программу*» → в открывшемся окне введите имя и выберите язык программирования → нажмите кнопку «*Ок*». Дочерняя программа будет добавлена к программе в конец вложенного списка. Имя дочерней программы должно соответствовать правилам, предусмотренным для имен программ.



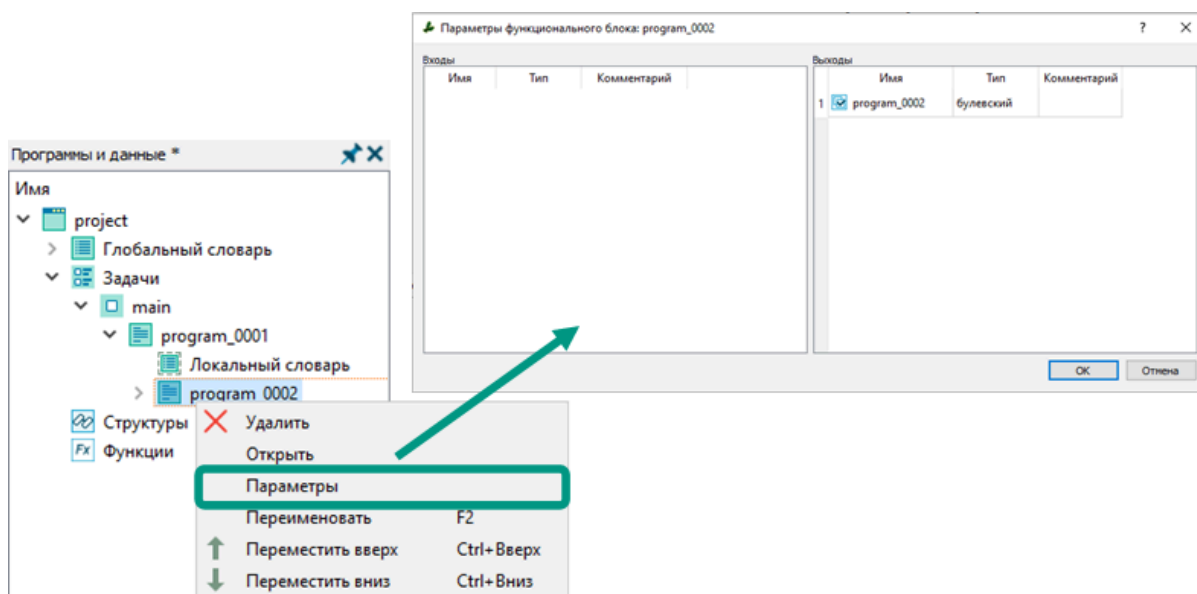
Для изменения порядка отображения в дереве проекта нажмите правой кнопкой мыши по дочерней программе → из контекстного меню выберите пункт переместить вверх ↑ (сочетание клавиш «Ctrl+Вверх») или переместить вниз ↓ (сочетание клавиш «Ctrl+Вниз»). Выбор пункта контекстного меню или сочетания клавиш перемещает подпрограмму на одну позицию.



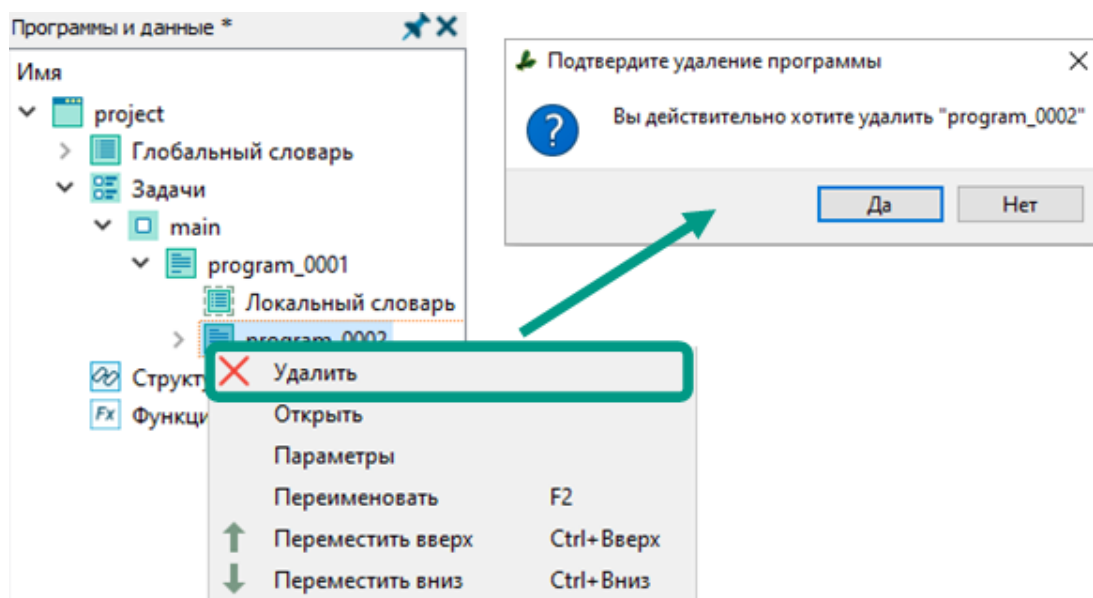
Для переименования откройте контекстное меню дочерней программы и выберите пункт «Переименовать» или выделите дочернюю программу и нажмите клавишу «F2» → введите новое имя → нажмите клавишу «Enter». Имя дочерней программы должно соответствовать правилам, предусмотренным для имен программ.



Для просмотра выходных и входных параметров откройте контекстное меню дочерней программы и выберите пункт «*Параметры*» → в открывшемся окне входные и выходные параметры с указанием имени, типа данных и комментария доступны только для просмотра.



Для удаления дочерней программы откройте контекстное меню дочерней программы и выберите пункт «*Удалить*» → в открывшемся окне подтвердите удаление нажатием кнопки «*Да*».



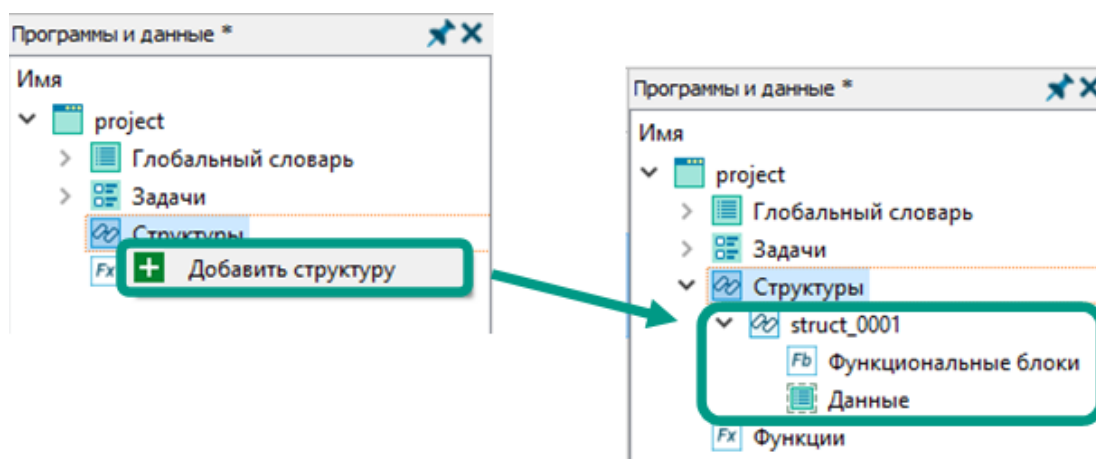
5.5 Структуры

Структура - тип данных, позволяющий в рамках одного объекта сгруппировать переменные различных типов. В составе структуры могут быть простые типы, массивы, а также другие пользовательские или системные структуры, без ограничений на уровень вложенности. Этим они отличаются от массивов, которые используются для группировки переменных одного типа.

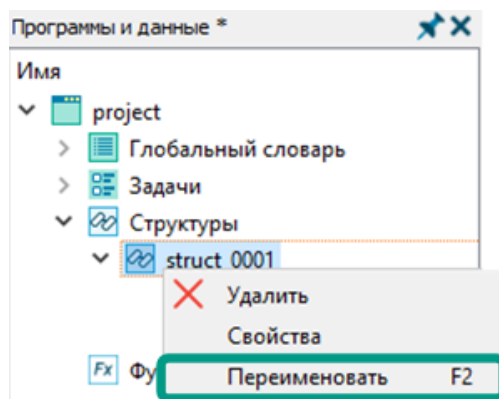
К каждой структуре в дереве проектов по умолчанию добавляются словарь данных для управления переменными структуры и раздел функциональных блоков для написания программной логики. В функциональных блоках доступны глобальные переменные и переменные из словаря структуры, которые выступают в качестве локальных переменных. Создание структур само по себе не приводит к изменению технологической программы. Для вызова функциональных блоков структуры в программах необходимо создать переменную с соответствующим структурным типом в глобальном словаре или локальном словаре одной из программ.

Доступ к переменным структуры осуществляется через конструкцию вида: `<имя_экземпляра_структуры>.<имя_переменной>`.

Для добавления структуры нажмите правой кнопкой мыши по разделу «Структуры» в дереве проектов → выберите из контекстного меню пункт «Добавить структуру» → новая структура будет добавлена в список структур с именем по умолчанию.



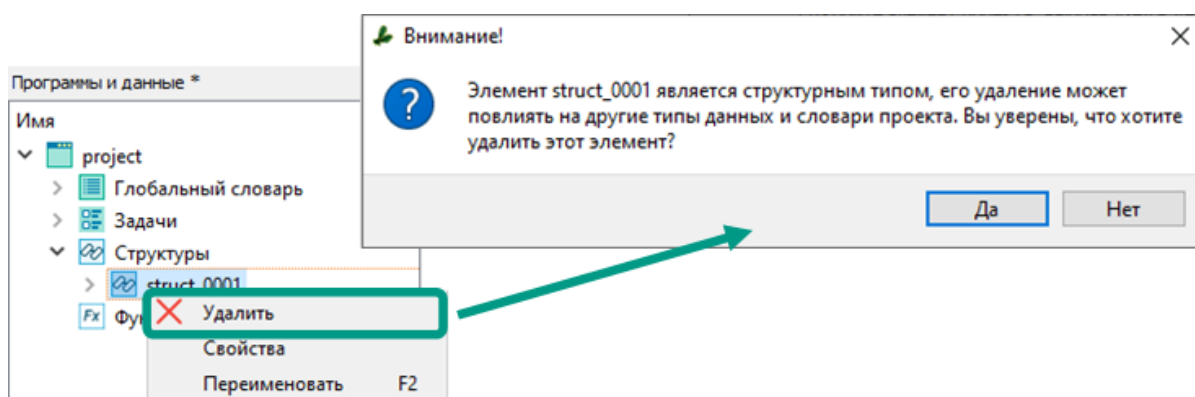
Для переименования структуры нажмите правой кнопкой мыши по структуре в дереве проектов и из контекстного меню выберите пункт «Переименовать» или выберите структуру и нажмите клавишу «F2» → введите новое имя структуры → нажмите клавишу «Enter».



Имя структуры должно соответствовать правилам:

- содержать только латинские буквы, цифры и символы подчеркивания;
- начинаться с латинской буквы;
- не превышать 128 символов.

Для удаления структуры нажмите правой кнопкой мыши по структуре в дереве проектов → из контекстного меню выберите пункт «Удалить» → в открывшемся окне подтвердите удаление нажатием кнопки «Да».



Перед удалением структуры убедитесь, что она не используется в качестве типа данных для переменных проекта.

5.6 Функциональные блоки

Функциональные блоки входят в состав структуры и содержат программный код для выполнения операций над данными. Функциональные блоки могут быть вызваны любыми программами и дочерними программами, а также функциональными блоками других структур. Функциональные блоки одной структуры не могут вызывать друг друга. Для вызова функционального блока предварительно требуется создать экземпляр структуры - объявить в локальном словаре или словаре данных структуры переменную и указать в качестве типа данных структуру, функциональный блок которой будет использоваться. Вызов функционального блока в функциях недоступен, так как в функциях могут быть объявлены переменные только простых типов данных.

Из функционального блока можно обратиться к глобальным переменным и переменным из текущего экземпляра структуры. При совпадении имен переменных структуры с глобальными переменными, приоритетом обладает переменная из словаря данных структуры. Использование типов данных в структурах не ограничено.

Данные в функциональные блоки передаются через входные параметры. Вызов функционального блока осуществляется через конструкцию вида: `<имя_экземпляра_структуры>.<имя_блока>(<параметр1>, <параметр2>..., <параметрN>)`. При этом блоки могут возвращать несколько выходных значений по нескольким входным значениям. Внутренние локальные переменные не доступны за пределами функционального блока и используются для временного хранения промежуточных результатов вычислений.

Алгоритм выполнения функционального блока:

1. Выполнение начинается только после вызова блока одной из программ проекта, при этом выполнение вызвавшей его программы приостанавливается до завершения выполнения блока.
2. Доступ к входным и выходным параметрам функционального блока организован через входные и выходные переменные, созданные в словаре данных соответствующей структуры, где также могут создаваться внутренние переменные.
3. При каждом вызове функционального блока создаются его экземпляр и экземпляры используемых в блоке переменных. Экземпляры - точные копии функциональных блоков или переменных в памяти контроллера, доступные для чтения и редактирования только в период выполнения блока. При этом внутренним переменным изначально присваивается значение по умолчанию, указанное в словаре данных структуры.
4. Затем происходит чтение входных параметров, которые могут меняться при каждом вызове блока, и их значения записываются в экземпляры входных

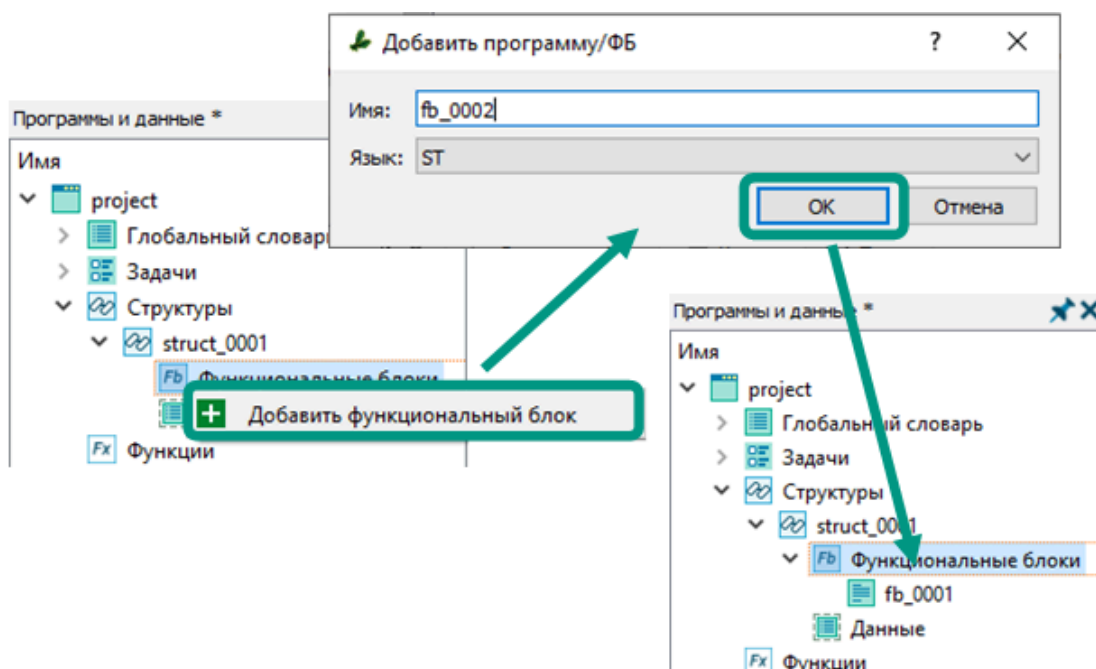
переменных. Функциональные блоки могут иметь произвольное количество входных параметров.

5. В процессе выполнения в зависимости от входных параметров формируются значения выходных параметров, которые записываются в экземпляры выходных переменных. Функциональные блоки могут иметь произвольное количество выходных параметров.
6. После окончания выполнения выходные параметры сохраняются в соответствующих переменных программы, экземпляр функционального блока вместе с экземплярами переменных удаляется из памяти контроллера, а процесс управления переходят к следующему оператору программы.

Для создания функционального блока нажмите правой кнопкой мыши по разделу «Функциональные блоки» → в контекстном меню выберите пункт «Добавить функциональный блок» → в диалоговом окне введите имя блока с учетом нижеуказанных правил и выберите язык программирования → нажмите на кнопку «Ок». Новый блок будет добавлен в конец списка функциональных блоков.

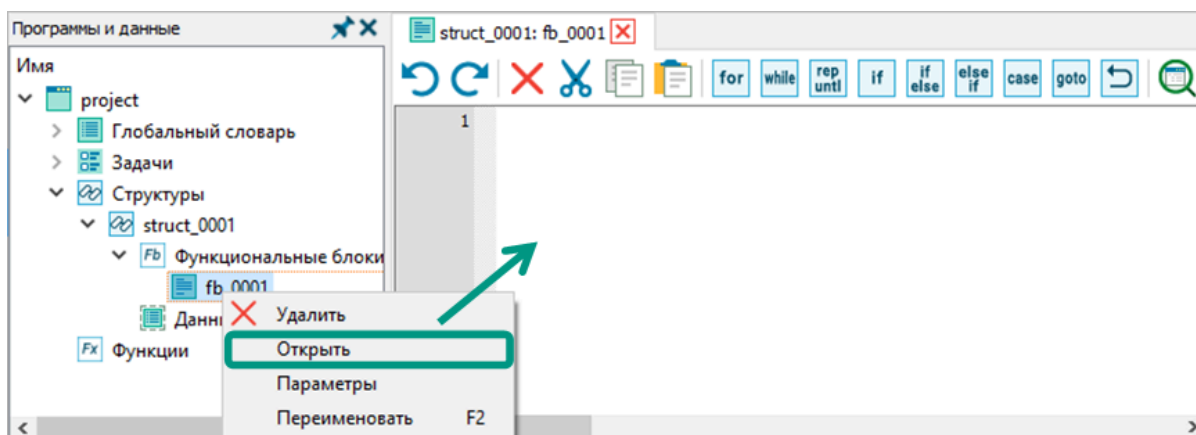
Имя функционального блока должно соответствовать правилам:

- содержать только латинские буквы, цифры и символы подчеркивания;
- начинаться с латинской буквы;
- не должно превышать 128 символов.

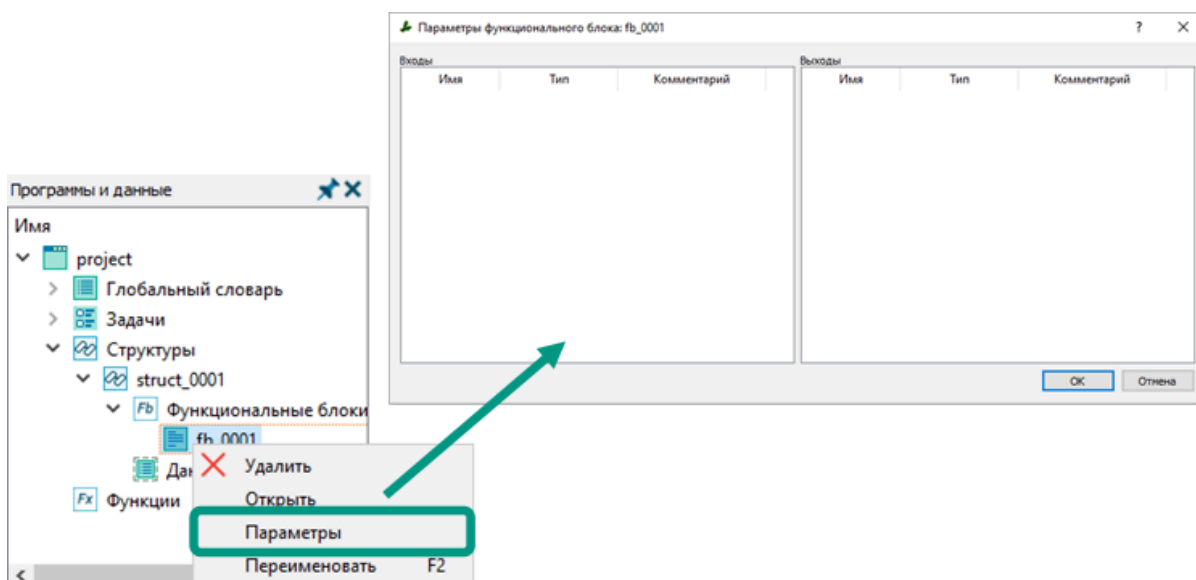


Для открытия редактора функционального блока дважды щелкните левой кнопкой мыши по функциональному блоку в дереве проекта или нажмите по блоку правой кнопкой мыши и из контекстного меню выберите пункт «Открыть» → редактор будет открыт в рабочем пространстве среды разработки.

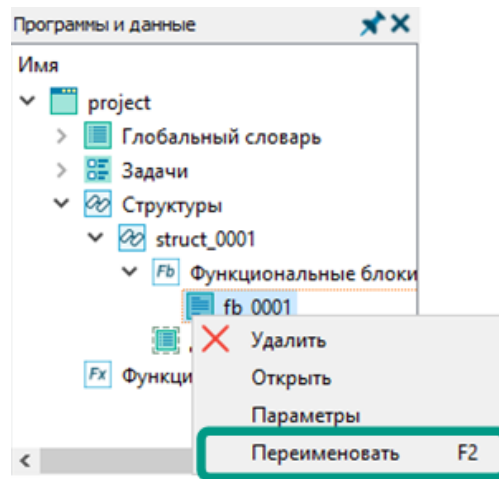
Подробное описание интерфейса и возможностей редакторов приведено в разделе [Редакторы](#).



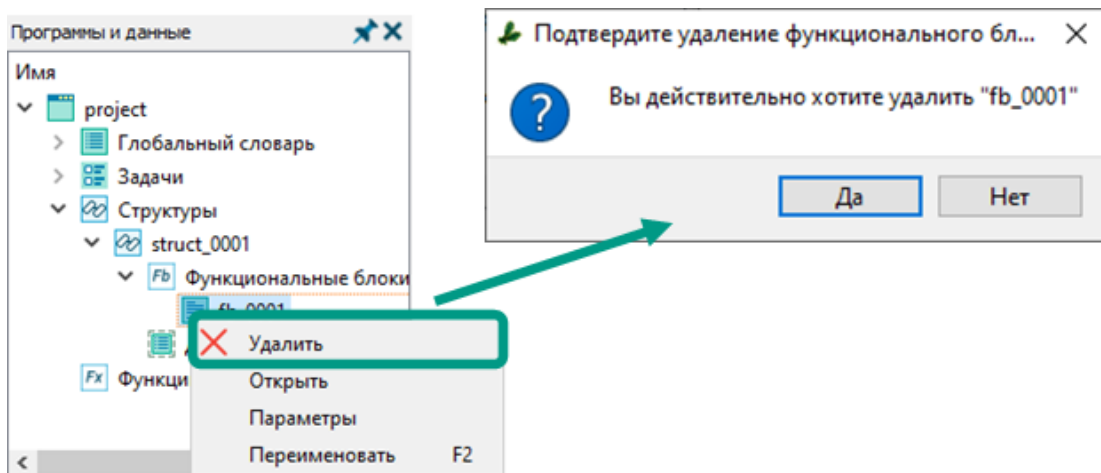
Для просмотра входных и выходных параметров нажмите правой кнопкой мыши по функциональному блоку в дереве проекта и в контекстном меню выберите пункт «Параметры» → в открывшемся окне будут выведены все входные и выходные параметры с указанием имени, типа данных и комментария.



Для переименования нажмите правой кнопкой мыши по блоку в дереве проекта и в контекстном меню выберите пункт «Переименовать» или после выделения блока нажмите клавишу «F2» → введите новое название блока с учетом правил, указанных в настоящем разделе, → нажмите клавишу «Enter».



Для удаления функционального блока нажмите правой кнопкой мыши по функциональному блоку в дереве проекта → из контекстного меню выберите пункт «Удалить» → в открывшемся диалоговом окне нажмите кнопку «Да» → блок будет удален из дерева проекта.



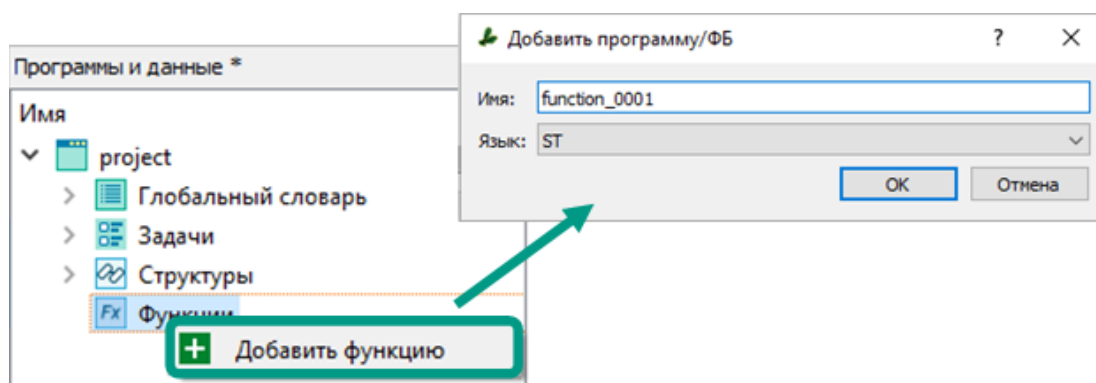
5.7 Функции

Функции - разновидность программного объекта, могут быть написаны на языках программирования ST или FBD/LD. Функции могут быть напрямую вызваны в любых программах, дочерних программах и функциональных блоках, а также в других функциях. При этом доступен рекурсивный вызов, когда функция вызывает на выполнение саму себя. При этом следует обращать внимание на глубину рекурсии и наличие условия для ее прекращения. При большой глубине рекурсии потребуется большой объем стековой памяти для хранения копий значений аргументов и адреса возврата для каждого вызова, что может привести к переполнению стека и аварийному завершению приложения. В качестве замены рекурсивного вызова можно использовать циклические конструкции.

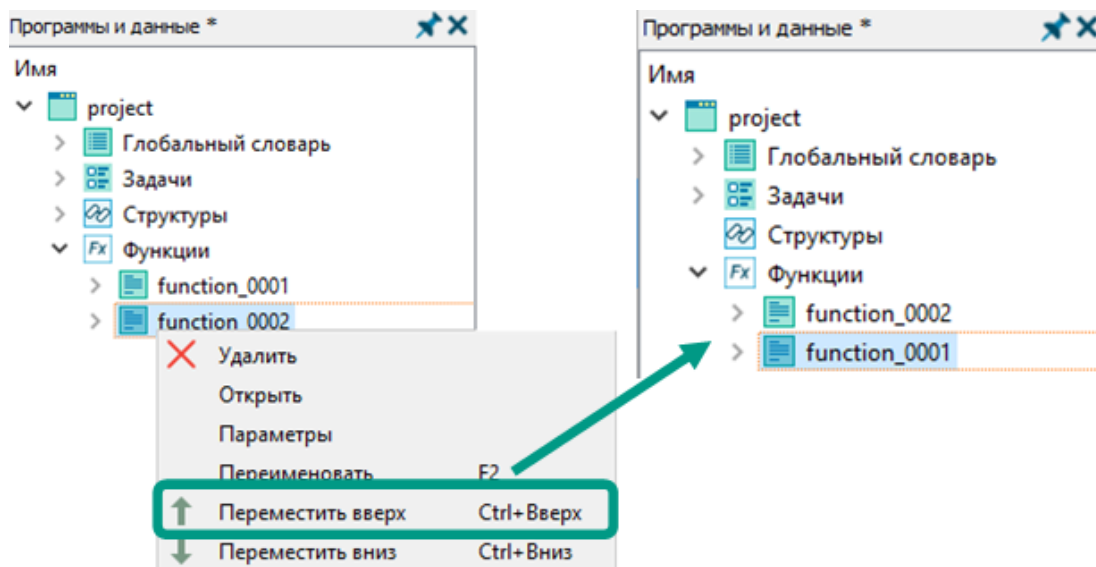
В локальном словаре функции могут быть объявлены переменные только простых типов данных. В этом их основное отличие от дочерних программ и функциональных блоков. Из функции можно обратиться к глобальным и локальным переменным текущей функции. При совпадении имен глобальных и локальных переменных, приоритетом обладает локальная переменная.

Данные в функцию передаются при ее вызове через один или несколько входных параметров. Вызов функции осуществляется через конструкцию вида: *<имя_функции>(<параметр1>, <параметр2>..., <параметрN>)*. При этом функция всегда должна возвращать одно выходное значение. Для этого используется выходная переменная, которая по умолчанию включается в локальный словарь дочерней программы после ее добавления в проект.

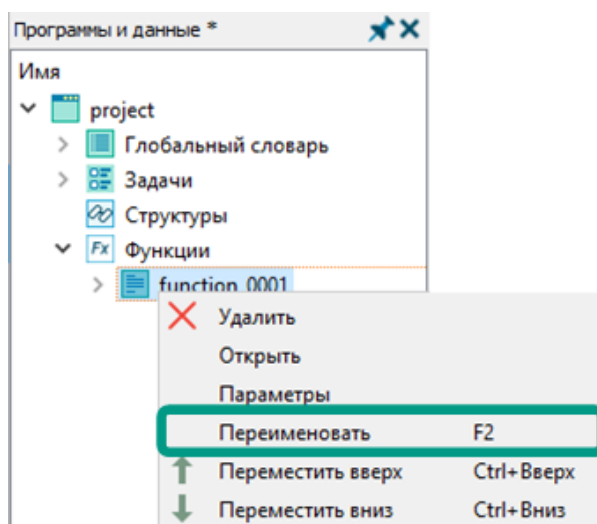
Для добавления функции в дереве проекта кликните правой кнопкой мыши по разделу «*Функции*» → из контекстного меню выберите пункт «*Добавить функцию*» → в открывшемся окне введите имя и выберите язык программирования → нажмите кнопку «*Ок*». Имя функции должно соответствовать правилам, предусмотренным для имен программ.



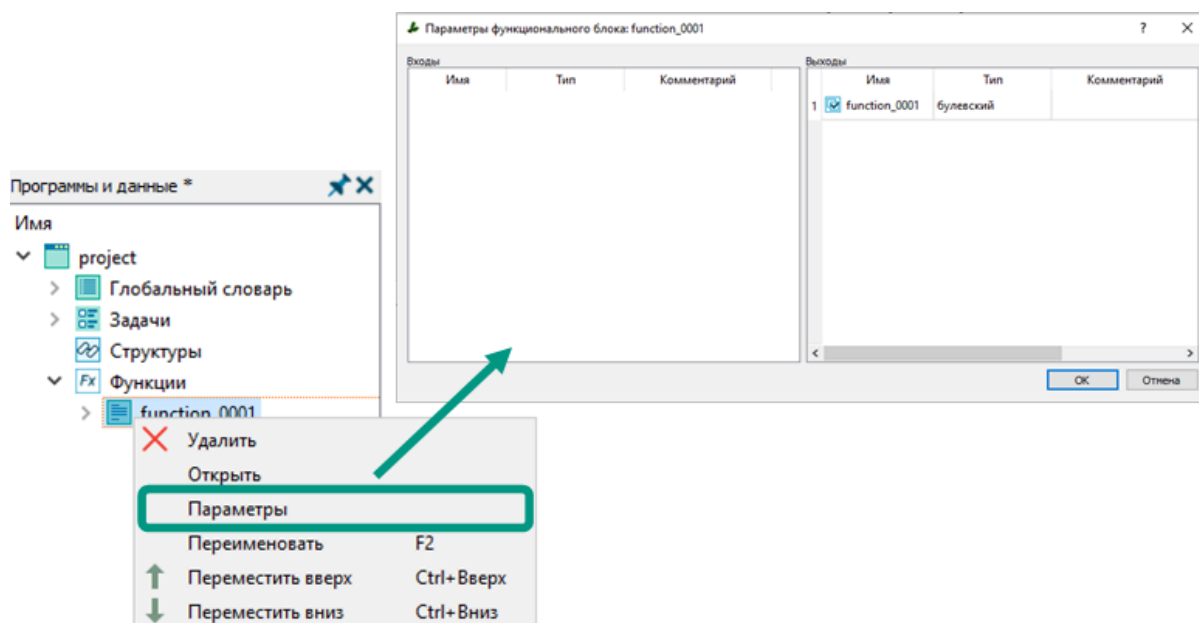
Для изменения порядка отображения функций в дереве проекта нажмите правой кнопкой мыши по функции → из контекстного меню выберите пункт переместить вверх ↑ (сочетание клавиш «Ctrl+Вверх») или переместить вниз ↓ (сочетание клавиш «Ctrl+Вниз»). Выбор пункта контекстного меню или сочетания клавиш перемещает функцию на одну позицию.



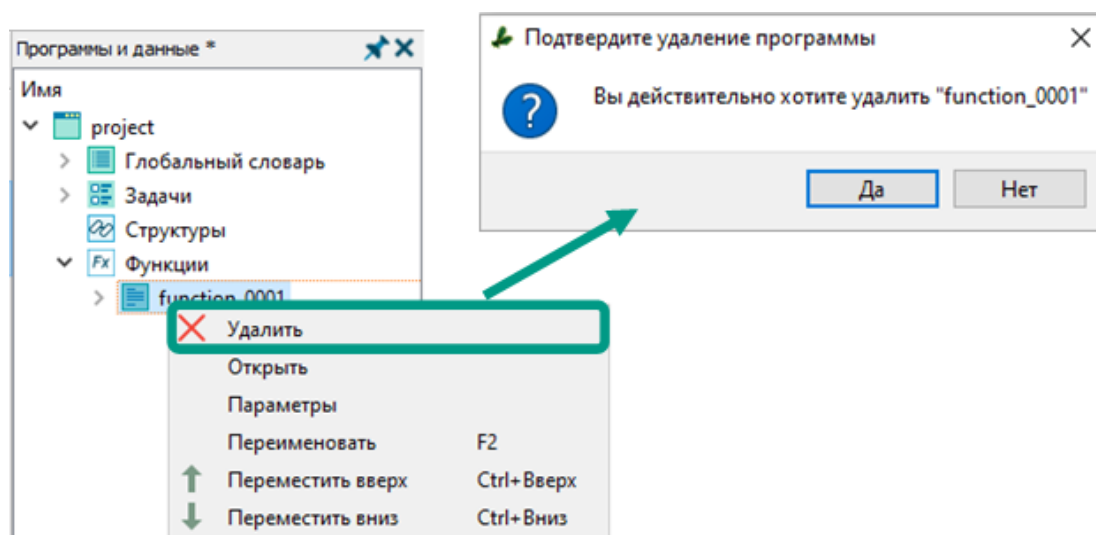
Для переименования откройте контекстное меню функции и выберите пункт «Переименовать» или выделите функцию и нажмите клавишу «F2» → введите новое имя → нажмите клавишу «Enter». Имя функции должно соответствовать правилам, предусмотренным для имен программ.



Для просмотра входных и выходных параметров откройте контекстное меню функции и выберите пункт «*Параметры*» → в открывшемся окне входные и выходные параметры с указанием имени, типа данных и комментария доступны только для просмотра.



Для удаления откройте контекстное меню функции и выберите пункт «*Удалить*» → в открывшемся окне подтвердите удаление нажатием кнопки «*Да*».



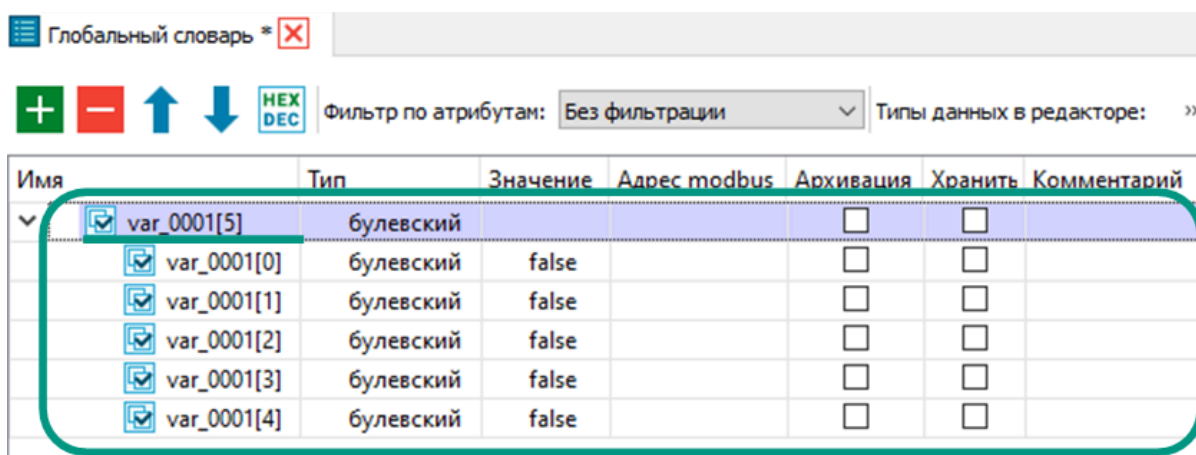
5.8 Массивы

Массив - упорядоченный набор элементов с одинаковым типом данных. В среде разработки Unimod PRO 2 массивы создаются в словарях с использованием имени переменной. Для создания массива после имени переменной в квадратных скобках нужно указать размер массива - целое положительное число. Вложенные элементы массива добавляются автоматически при его создании, их количество соответствует указанному размеру массива. Поддерживаются одномерные, двумерные и трехмерные массивы, размер каждого измерения указывается в квадратных скобках через запятую. Например:

var_1[10] - одномерный массив;

var_2[10,10] - двумерный массив;

var_3[10,10,10] - трехмерный массив.



The screenshot shows the 'Глобальный словарь' (Global Dictionary) interface. At the top, there are control buttons: a green plus sign, a red minus sign, blue up and down arrows, and a 'HEX DEC' toggle. Below these is a filter dropdown set to 'Без фильтрации' (No filtering) and a 'Типы данных в редакторе:' (Data types in editor) dropdown. The main area is a table with columns: 'Имя' (Name), 'Тип' (Type), 'Значение' (Value), 'Адрес modbus' (Modbus address), 'Архивация' (Archiving), 'Хранить' (Store), and 'Комментарий' (Commentary). The table contains one main entry, 'var_0001[5]', which is expanded to show five sub-entries: 'var_0001[0]', 'var_0001[1]', 'var_0001[2]', 'var_0001[3]', and 'var_0001[4]'. All sub-entries have the type 'булевский' (boolean) and the value 'false'. The 'Архивация' and 'Хранить' columns for all entries contain unchecked checkboxes.

Имя	Тип	Значение	Адрес modbus	Архивация	Хранить	Комментарий
var_0001[5]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[0]	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[1]	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[2]	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[3]	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[4]	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	

Для двумерных и трехмерных массивов создаются вложенные элементы с двумя и тремя уровнями вложенности соответственно. При этом в имени каждого элемента массива в квадратных скобках указывается его индекс - адрес элемента, по которому можно обратиться к его значению из программных объектов. Индексы в массивах начинаются с нуля, поэтому первому элементу массива соответствует индекс [0], второму - индекс [1] и так далее. Количество индексов зависит от размерности массива и может содержать один, два или три целых числа, разделенных запятой.

Глобальный словарь

Фильтр по атрибутам: Без фильтрации Типы данных в редакторе: »

Имя	Тип	Значение	Адрес modbus	Архивация	Хранить	Комментарий
var_000 [5,2,2]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[0]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[0,0]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[0,0,0]	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	
var_000 [0,0,1]	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[0,1]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[1]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[2]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[3]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	
var_0001[4]	булевский			<input type="checkbox"/>	<input type="checkbox"/>	

Размерность массива

Индекс элемента массива



Будьте внимательны при работе с массивами, особенно при обращении к его первому и последнему элементам. Попытка обратиться из программы к несуществующему элементу массива приведет к аварийной остановке всего приложения. Если обращение к несуществующему элементу массива произойдет в дочерней программе, функциональном блоке или функции, то их выполнение будет завершено, для выходного параметра будет сохранено последнее присвоенное значение или значение по умолчанию, а управление перейдет к следующему после них оператору.


6 Управление проектами

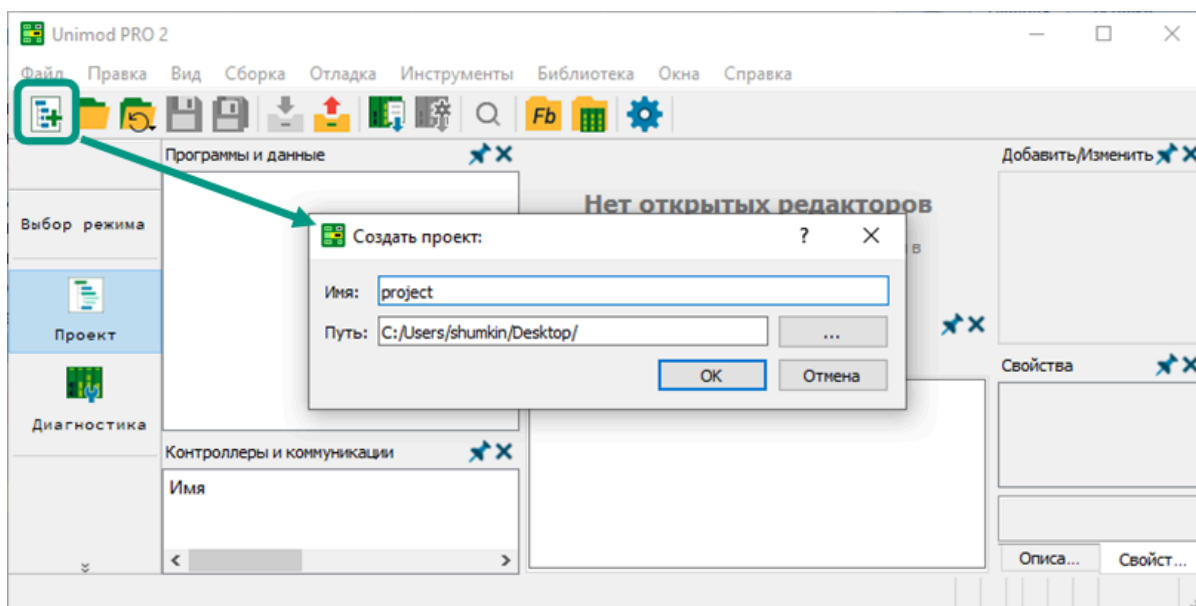
Среда разработки Unimod PRO 2 позволяет редактировать только один текущий проект. Для открытия другого проекта потребуется закрыть текущий. Работа с несколькими проектами осуществляется по очереди.

В настоящем разделе описаны основные операции по созданию, открытию, изменению, сохранению и закрытию проекта. Процессам чтения проекта с контроллера и его загрузки в контроллер, работе с архивом проекта посвящены отдельные подразделы.

В данном разделе не рассматриваются процессы работы с элементами проекта, их описание приведено в разделе [Состав проекта](#).

6.1 Создание проекта

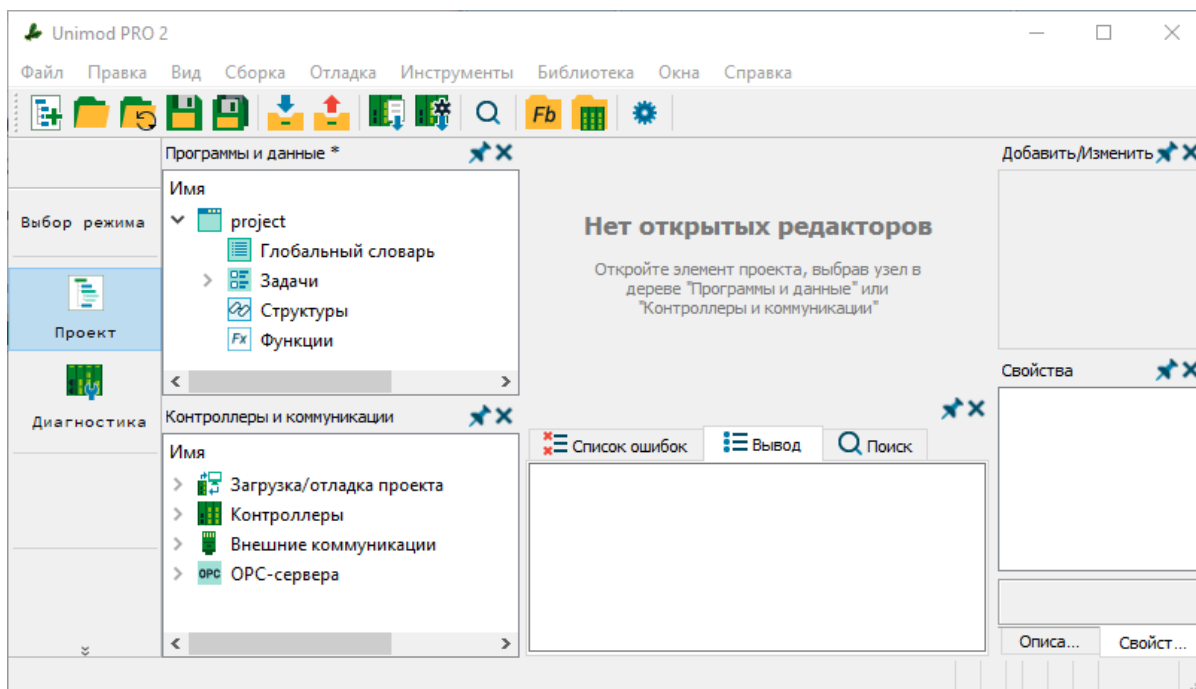
Для создания нового проекта выберите пункт меню «Файл → Создать проект» или нажмите на кнопку  на панели быстрого доступа → в открывшемся окне введите имя проекта с учетом нижеуказанных правил и укажите путь к каталогу нового проекта в операционной системе → для завершения нажмите кнопку «Ок». После чего в выбранном каталоге будет создан подкаталог с указанным именем проекта, где будут храниться все файлы проекта.



Имя проекта должно соответствовать следующим правилам:


- содержать только латинские буквы, цифры и символы подчеркивания;
- начинаться с латинской буквы;
- не должно превышать 128 символов.

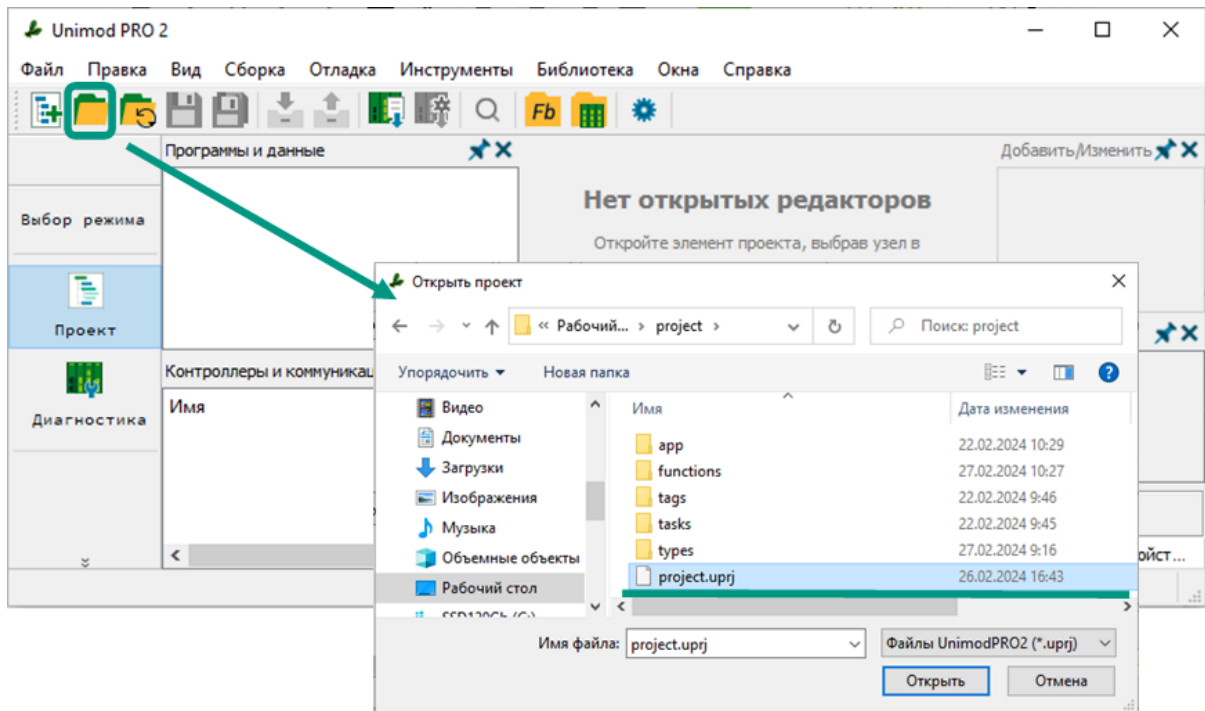
Созданный проект будет открыт в среде разработки Unimod PRO 2, а дерево проекта и дерево конфигурации будут доступны для редактирования.




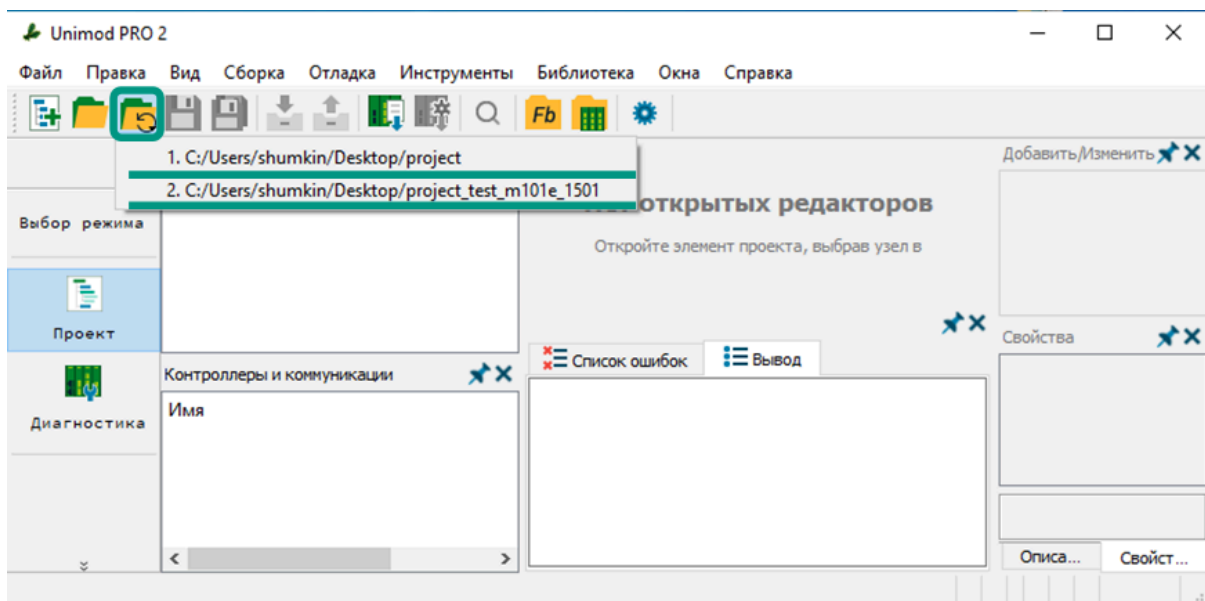
В среде разработки Unimod PRO 2 можно работать только с одним открытым проектом. Перед созданием нового проекта убедитесь, что отсутствуют открытые проекты. Для закрытия текущего проекта выберите пункт меню «Файл → Закрыть проект» или нажмите сочетание клавиш «Alt + X».

6.2 Открытие проекта

Для открытия ранее созданного проекта выберите пункт меню «Файл → Открыть проект» или нажмите кнопку  на панели быстрого доступа → в открывшемся окне перейдите в каталог проекта и выберите файл проекта с расширением «*.uprj» → нажмите кнопку «Открыть».



Для открытия недавно редактированных проектов выберите пункт меню «Файл → Недавние проекты» или нажмите кнопку  на панели быстрого доступа → из выпадающего списка выберите нужный проект.




После открытия проекта в среде разработки Unimod Pro 2, дерево проекта и дерево конфигурации будут доступны для редактирования.

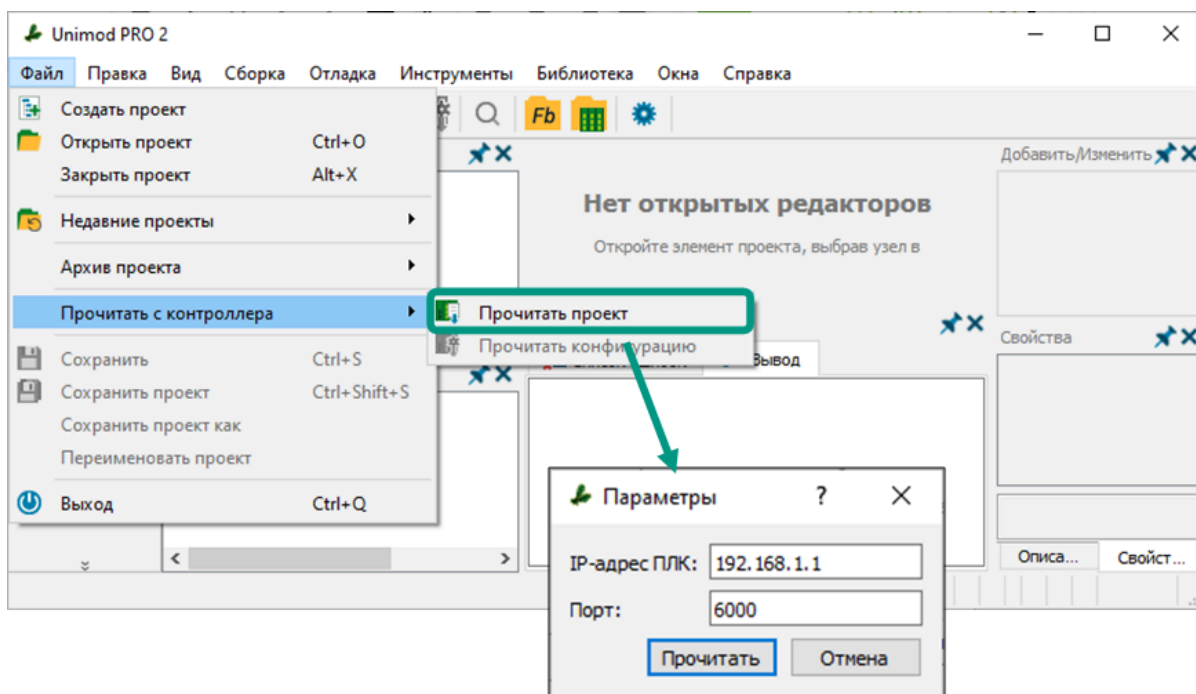


В среде разработки Unimod PRO 2 можно работать только с одним открытым проектом. Перед открытием ранее созданного проекта убедитесь, что отсутствуют открытые проекты. Для закрытия текущего проекта выберите пункт меню «Файл → Закрыть проект» или нажмите сочетание клавиш «Alt + X».

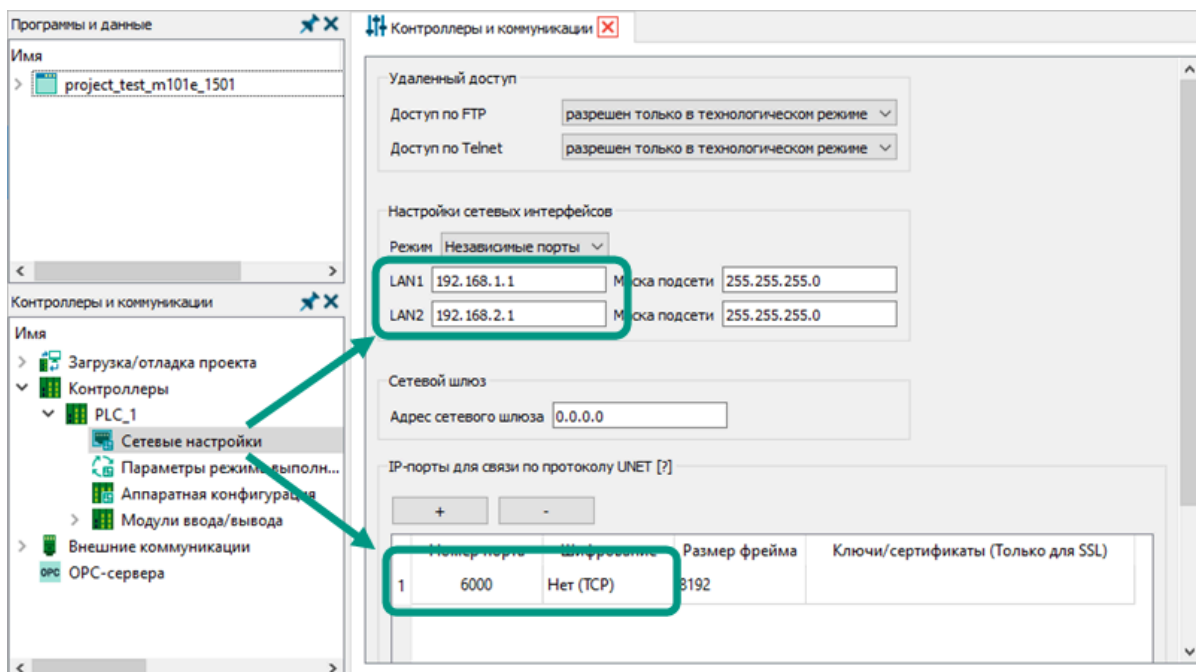
6.3 Чтение с контроллера

Среда разработки Unimod PRO 2 поддерживает возможность чтения готового проекта с контроллера для его доработки или использования в качестве основы нового проекта. Данный вариант работы актуален, когда исходный код проекта сохранен только на контроллере, и позволяет сократить временные затраты на разработку проекта с нуля.

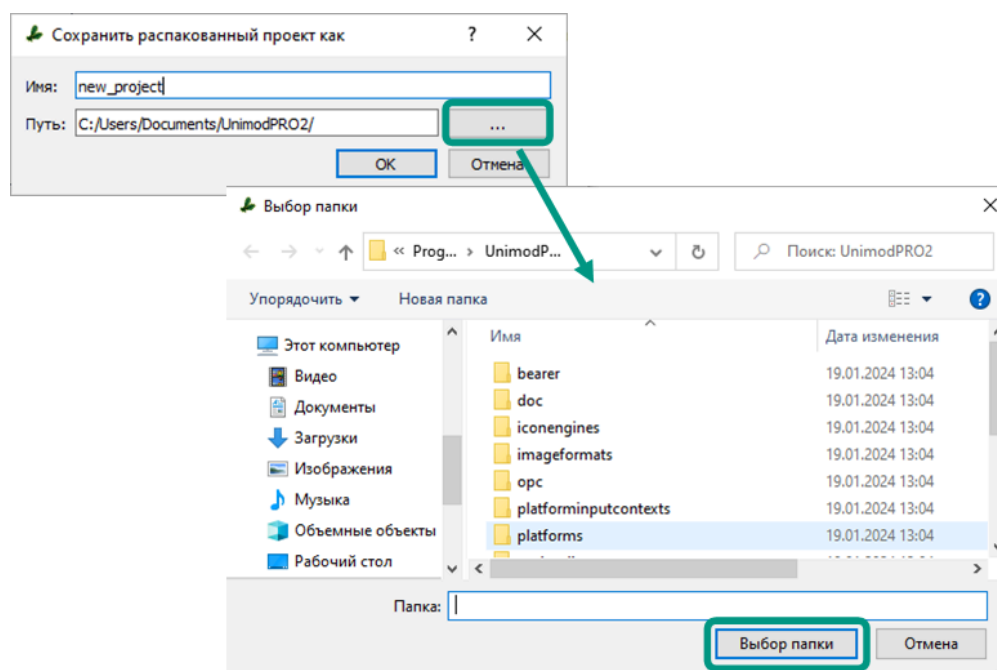
Для чтения проекта с контроллера выберите пункт меню «Файл → Прочитать с контроллера → Прочитать проект» или нажмите кнопку  на панели быстрого доступа, затем в открывшемся окне укажите IP-адрес контроллера, номер порта и нажмите кнопку «Прочитать» → произойдет попытка подключения к контроллеру. Ошибки при подключении могут возникнуть при некорректно заданных параметрах или неполадках физического подключения к контроллеру.



IP-адрес и номер порта для подключения задаются в сетевых настройках контроллера в разделе «Контроллер → PLC_1 → Сетевые настройки». При чтении проекта IP-адрес и номер порта должны совпадать с сетевыми настройками приложения, загруженного в контроллере. Подробнее в разделе [Сетевые настройки](#) настоящего руководства.



При успешном подключении к контроллеру в открывшемся окне введите имя проекта и путь к каталогу проекта в операционной системе с учетом правил, указанных в разделе [Создание проекта](#), → нажмите кнопку «Ок». При успешном завершении проект будет открыт и доступен для редактирования.



В среде разработки Unimod PRO 2 можно работать только с одним открытым проектом. Перед чтением проекта с контроллера убедитесь, что отсутствуют открытые проекты. Для закрытия текущего проекта выберите пункт меню «Файл → Закрыть проект» или нажмите сочетание клавиш «Alt + X».

6.4 Загрузка в контроллер

В процессе загрузки среда разработки Unimod PRO 2 запишет собранное приложение в энергонезависимую память контроллера и выведет результаты выполнения этой операции. На контроллере загруженное приложение будет сразу запущено на выполнение.

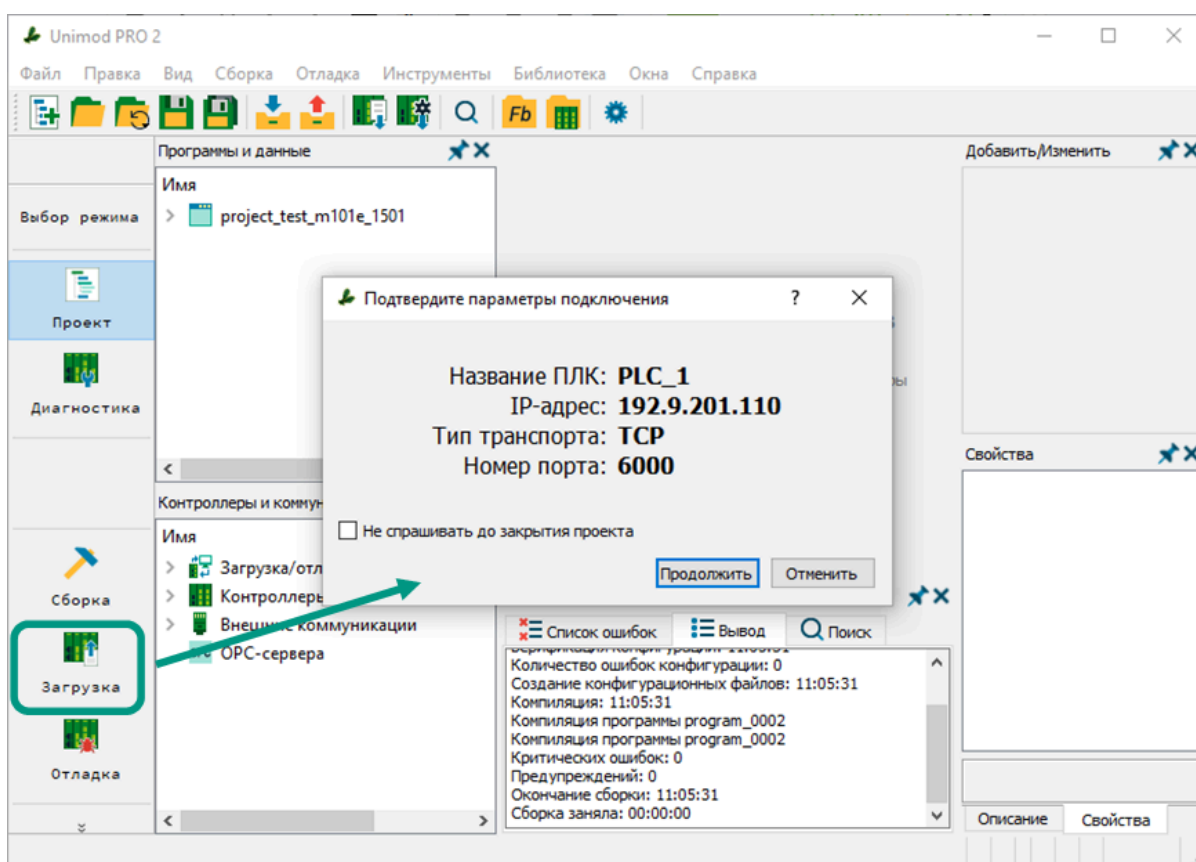
Перед загрузкой необходимо указать параметры подключения к контроллеру и собрать приложение. Параметры подключения указываются в разделе «*Загрузка/отладка проекта* → *Подключение к PLC_1*» дерева конфигурации, подробнее в разделе [Параметры подключения к контроллеру](#). Процесс сборки приложения описан в разделе [Сборка проекта](#).



Значения IP-адресов в параметрах подключения и сетевых настройках могут различаться. В параметрах подключения указывается значение одного из текущих IP-адресов контроллера, а в сетевых настройках - значения IP-адресов, которые будут установлены на сетевые интерфейсы контроллера только после загрузки приложения. После загрузки подключение к контроллеру для загрузки и отладки будет осуществляться по одному из IP-адресов, указанных в сетевых настройках загруженного приложения.

Для загрузки собранного приложения в контроллер на панели выбора режима работы выберите команду «Загрузка» → в открывшемся окне проверьте параметры подключения, введите название контроллера и нажмите на кнопку «Продолжить». В качестве альтернативного варианта для запуска процесса загрузки выберите пункт меню «Отладка → Загрузка» или нажмите клавишу «F7».

Вводимое имя должно совпадать с именем контроллера, указанным в разделе «Контроллеры» дерева конфигурации. Требования о выводе окна с параметрами подключения и дополнительном вводе названия контроллера можно задать в [настройках среды разработки](#). Эти требования позволяют пользователю убедиться, что он не ошибся при выборе контроллера для загрузки приложения.



При успешном подключении к контроллеру будет открыто окно загрузки приложения со сведениями о состоянии контроллера, параметрах текущего и загруженного в контроллере приложений. Зеленым цветом выделяются параметры успешного подключения контроллеру и безошибочной работы приложения, а также совпадающие параметры приложений. Желтым выделяются параметры, не влияющие на процесс работы контроллера, в основном это различия в дате и номере сборки. А красным выделяются ошибки и не совпадающие параметры приложений. Для продолжения загрузки нажмите кнопку «Загрузить приложение».

При включенном режиме резервирования и подключенном резервном контроллере в окне загрузки приложения также будут отображаться вышеуказанные параметры резервного контроллера, а приложение будет загружено сразу на оба контроллера.

The screenshot shows a window titled "Загрузка приложения в контроллер" with a table of application parameters and a control panel at the bottom.

	Приложение	Мастер-модуль
Адрес мастер-модуля	192.9.201.110	
Название проекта	project_test_m101e_1501	project_test_m101e_1501
Дата сборки	2024/01/23 16:00:46	2024/01/15 10:10:11
Номер сборки	17	10
MD5 проекта	ce4e239e08d4b461de90c44c5c567c91	11b7b886ef40bc7cc98cdb22fb415167
Приложение запущено		Да
Код запуска		Нет ошибок
Версия ПО		1.1
Статус загрузки		Готов
Контрольные суммы компонентов проекта		
MD5 программ и словарей	44c9a68e7812f814269bc5883f5af25b	ed055f615b31bf639f051349d17d4409
MD5 конфигурации тегов OPC	f70155910b09593fe89982be428e74ee	0696d113d17734c027fe05f254241da7
MD5 конфигурации ПЛК	97474603aa26d82fca2693f0a5be4422	97474603aa26d82fca2693f0a5be4422

Below the table, there is a progress bar showing 0% and a checkbox for "Пошаговая загрузка". A button labeled "Загрузить приложение" is highlighted with a red box and a red arrow. Other controls include a dropdown for "Текущий мастер-модуль" (Master-модуль 1) and buttons for "Поменять режимы (основной/резервный)" and "Синхронизировать резервный мастер-модуль".

После загрузки приложения в контроллер будут выведены результаты. При успешной загрузке все параметры выделяются зеленым цветом. В случае возникновения ошибки в соответствующем параметре будет выведено сообщение, выделенное красным цветом. После успешной загрузки приложения в контроллер, рекомендуется проверить правильность работы алгоритмов в режиме отладки.

	Приложение	Мастер-модуль
Адрес мастер-модуля		192.9.201.110
Название проекта	project_test_m101e_1501	project_test_m101e_1501
Дата сборки	2024/01/23 16:00:46	2024/01/23 16:00:46
Номер сборки	17	17
MD5 проекта	ce4e239e08d4b461de90c44c5c567c91	ce4e239e08d4b461de90c44c5c567c91
Приложение запущено		Да
Код запуска		Нет ошибок
Версия ПО		1.1
Статус загрузки		Загрузка приложения завершена
Контрольные суммы компонентов проекта		
MD5 программ и словарей	44c9a68e7812f814269bc5883f5af25b	44c9a68e7812f814269bc5883f5af25b
MD5 конфигурации тегов OPC	f70155910b09593fe89982be428e74ee	f70155910b09593fe89982be428e74ee
MD5 конфигурации ПЛК	97474603aa26d82fca2693f0a5be4422	97474603aa26d82fca2693f0a5be4422

100%

Пошаговая загрузка

Текущий мастер-модуль: Мастер-модуль 1

Поменять режимы (основной/резервный) | Синхронизировать резервный мастер-модуль

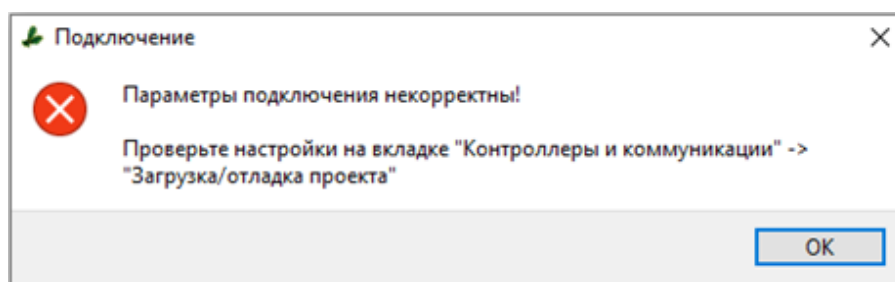
Загрузить приложение

Пошаговая загрузка приложения доступна при включенном режиме резервирования и подключенном резервном контроллере. Подробнее про настройку режима резервирования в разделе [Параметры резервирования](#). Пошаговая загрузка позволяет загружать приложение только на один из резервируемых контроллеров, менять режимы работы контроллеров и осуществлять синхронизацию резервного мастер-модуля.

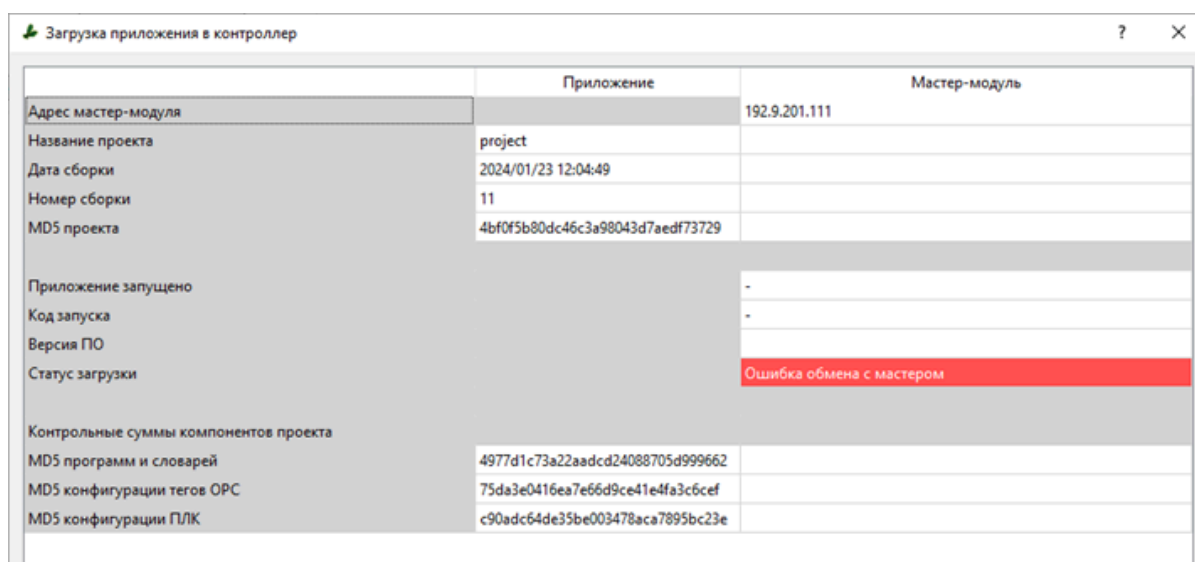
Для пошаговой загрузки в открывшемся окне отметьте галочкой пункт «Пошаговая загрузка» → выберите основной мастер-модуль (контроллер) и нажмите кнопку «Загрузить приложение». Для загрузки приложения на резервный контроллер повторите для него указанные действия или нажмите кнопку «Синхронизировать резервный мастер-модуль». При синхронизации резервный контроллер автоматически загрузит себе приложение основного.

Возможные ошибки и предупреждения при подключении к контроллеру

При отсутствии параметров подключения будет выведена ошибка. В данном случае в дереве конфигурации перейдите в раздел «*Загрузка/Отладка проекта → Подключение к PLC_1*» и укажите корректные параметры подключения: IP-адрес контроллера, номер порта и протокол передачи данных. Подробнее в разделе [Параметры подключения к контроллеру](#).

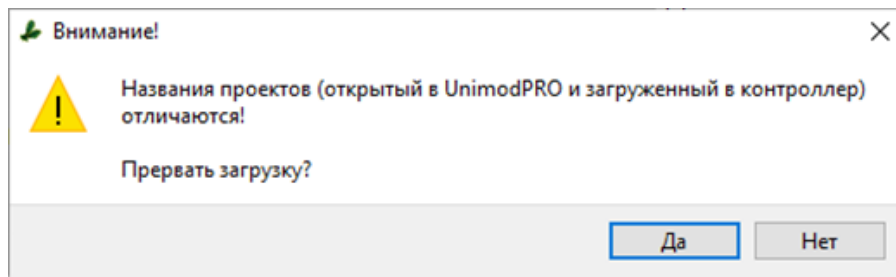


При неправильно заданных параметрах подключения или отсутствии связи с контроллером будет выведена ошибка. В данном случае в дереве конфигурации перейдите в раздел «*Загрузка/Отладка проекта → Подключение к PLC_1*» и укажите корректные параметры подключения. Подробнее в разделе [Параметры подключения к контроллеру](#). Если ошибка сохраняется при корректных параметрах, убедитесь в наличии связи с контроллером. Подробнее в разделе [Режим диагностики](#).




	Приложение	Мастер-модуль
Адрес мастер-модуля		192.9.201.111
Название проекта	project	
Дата сборки	2024/01/23 12:04:49	
Номер сборки	11	
MD5 проекта	4bf0f5b80dc46c3a98043d7aedf73729	
Приложение запущено	-	
Код запуска	-	
Версия ПО		
Статус загрузки		Ошибка обмена с мастером
Контрольные суммы компонентов проекта		
MD5 программ и словарей	4977d1c73a22aadcd24088705d999662	
MD5 конфигурации тегов OPC	75da3e0416ea7e66d9ce41e4fa3c6cef	
MD5 конфигурации ПЛК	c90adc64de35be003478aca7895bc23e	

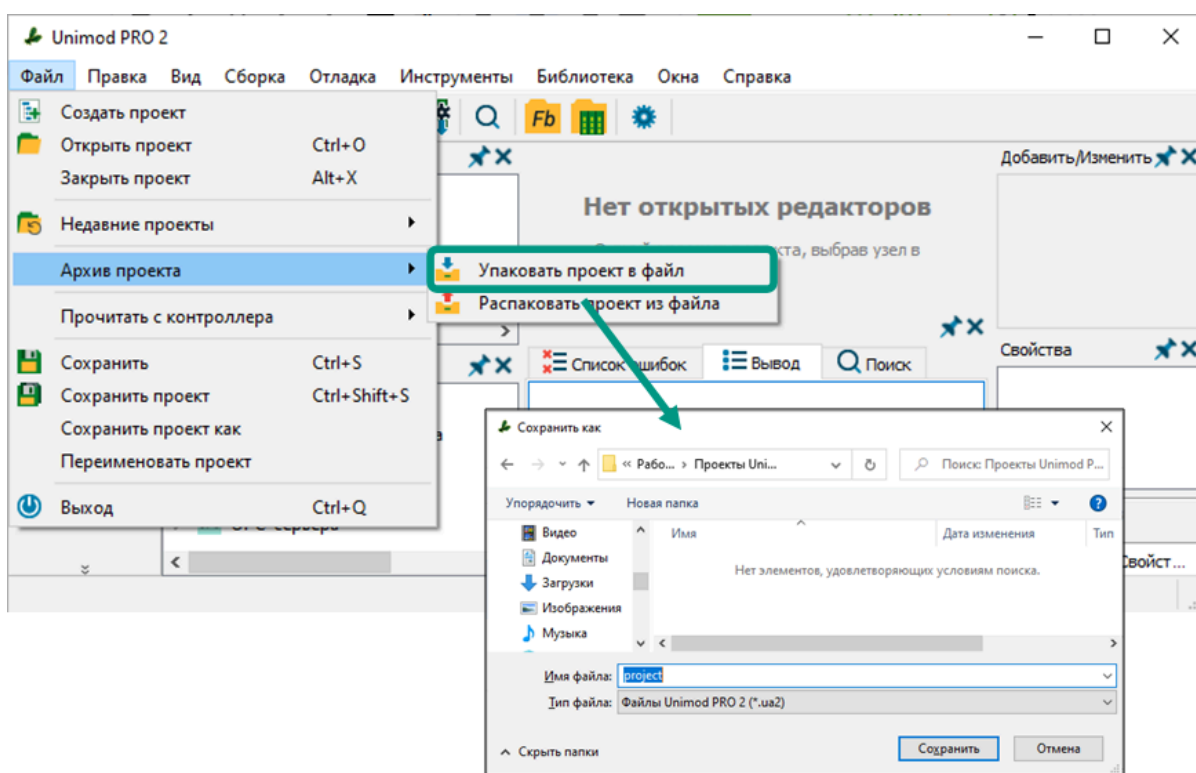
При отличии названий текущего и загруженного в контроллер проектов, в процессе подключения будет выведено предупреждение с предложением прервать загрузку. Для продолжения нажмите кнопку «*Нет*» и действуйте согласно вышеуказанной инструкции.




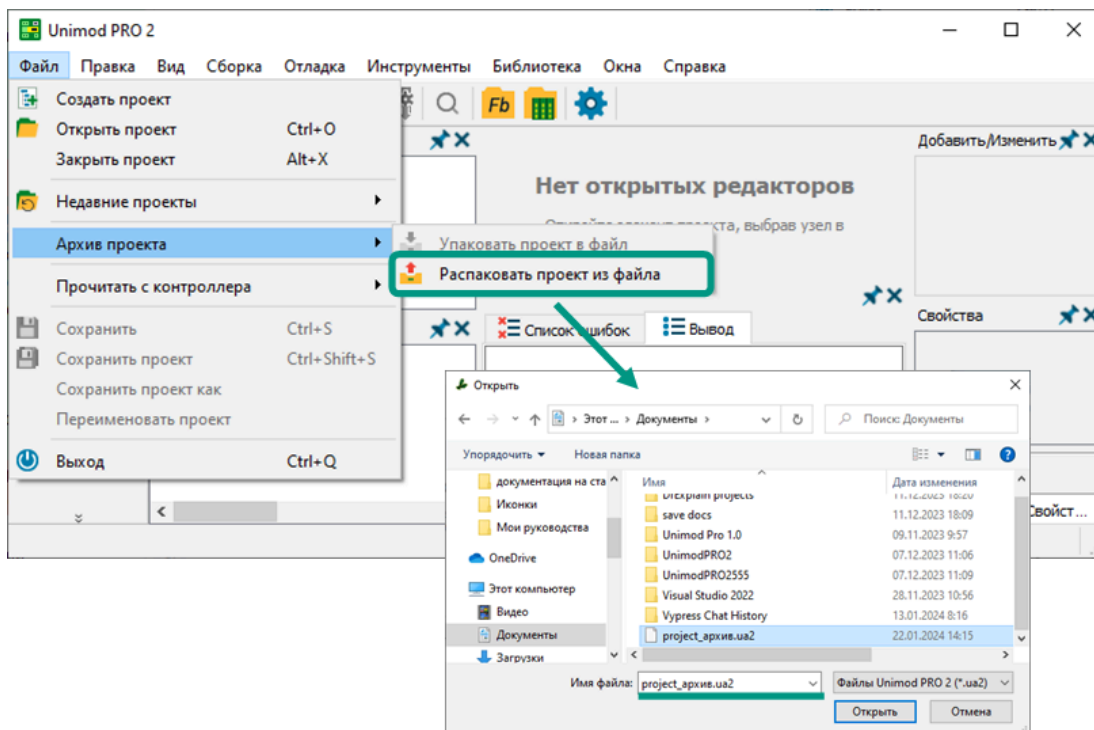
6.5 Работа с архивом проекта

В среде разработки Unimod PRO 2 предусмотрена возможность упаковки всех файлов проекта в архив для хранения или переноса на стороннее устройство, а также распаковки архива с файлами для продолжения работы с проектом.

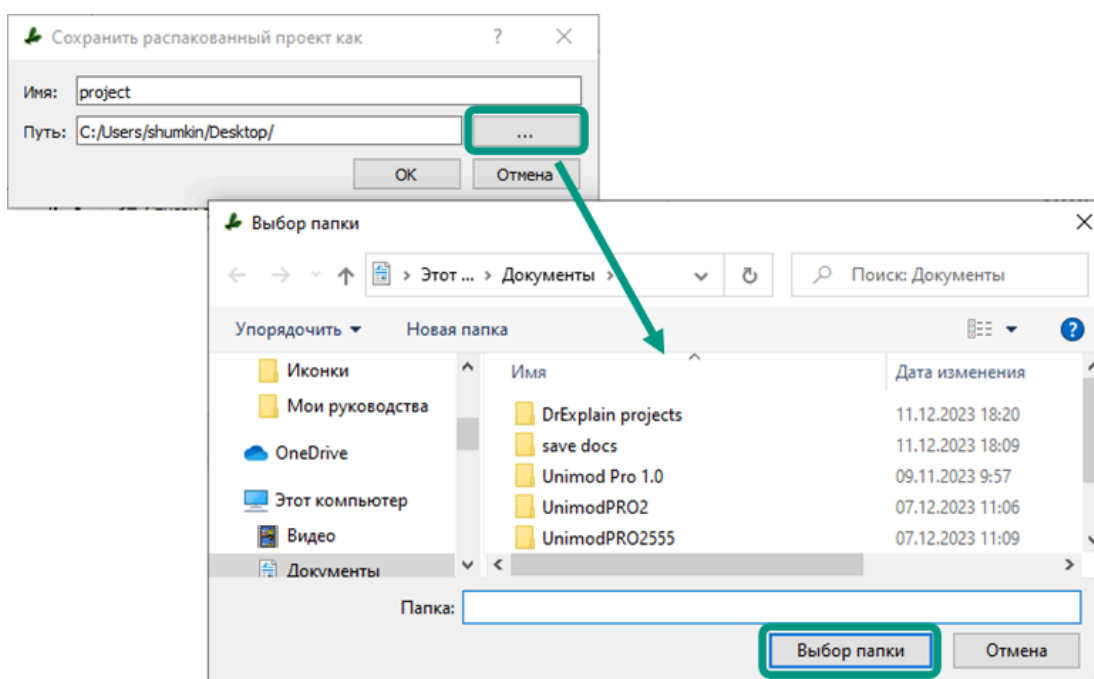
Для упаковки проекта в архив выберите пункт меню «*Файл* → *Архив проекта* → *Упаковать проект в файл*» или нажмите кнопку  на панели быстрого доступа, затем в открывшемся окне выберите каталог для сохранения, введите имя архива и нажмите кнопку «*Сохранить*». Требования к имени архива ограничиваются требованиями операционной системы, дополнительные ограничения не установлены. Архивный файл сохраняется с расширением «**.ua2*».



Для распаковки проекта из архива выберите пункт меню «Файл → Архив проекта → Распаковать проект из файла» или нажмите кнопку  на панели быстрого доступа, затем в открывшемся окне выберите файл архива с расширением «*.ua2» и нажмите кнопку «Открыть».



На следующем этапе в открывшемся окне введите имя проекта и путь к каталогу проекта в операционной системе с учетом правил и рекомендаций, указанных в разделе [Создание проекта](#), → нажмите кнопку «Ок».



По результатам распаковки архива будет выведено уведомление, а распакованный проект будет открыт и доступен для редактирования.



В среде разработки Unimod PRO 2 можно работать только с одним открытым проектом. Перед распаковкой проекта убедитесь, что отсутствуют открытые проекты. Для закрытия текущего проекта выберите пункт меню «Файл → *Закрыть проект*» или нажмите сочетание клавиш «Alt + X».

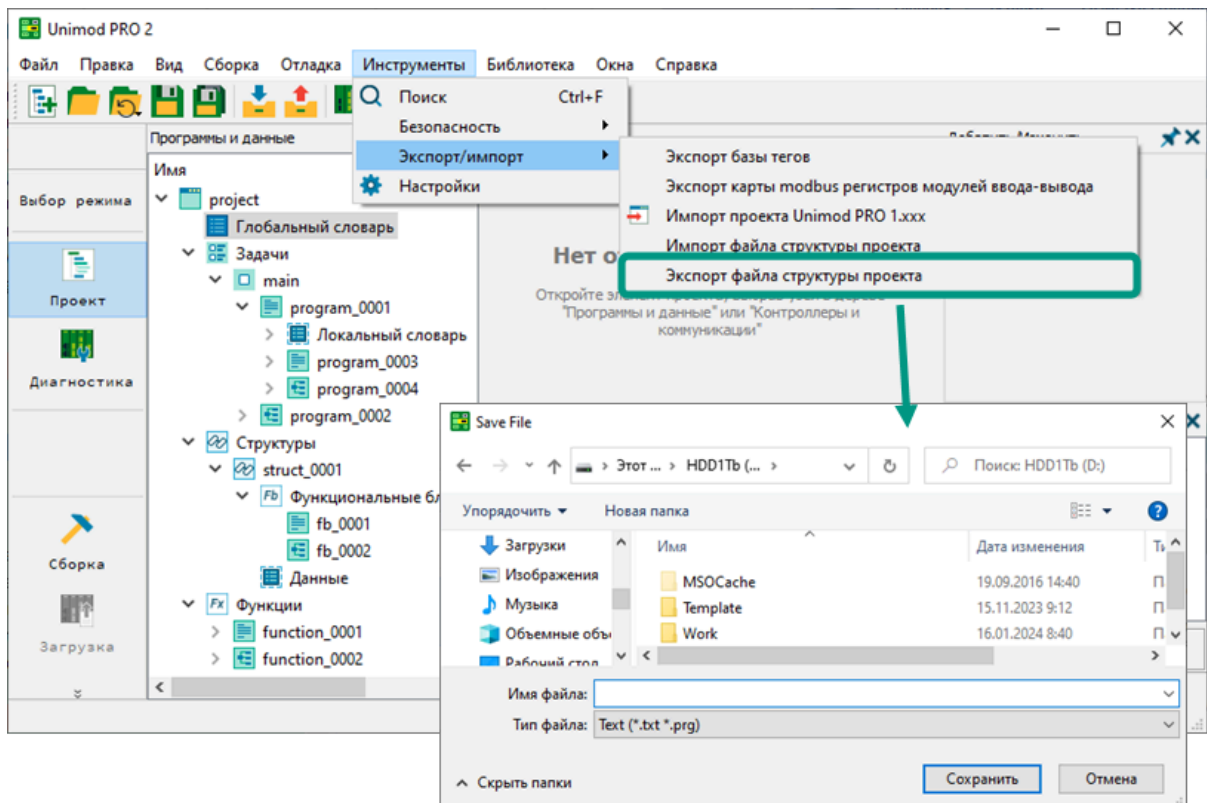
6.6 Экспорт/импорт структуры проекта

Структура проекта, разработанного в среде разработки Unimod PRO 2, может быть экспортирована в текстовый файл и перенесена в другой проект. В данном случае экспортируется только структура программных объектов без их содержимого.

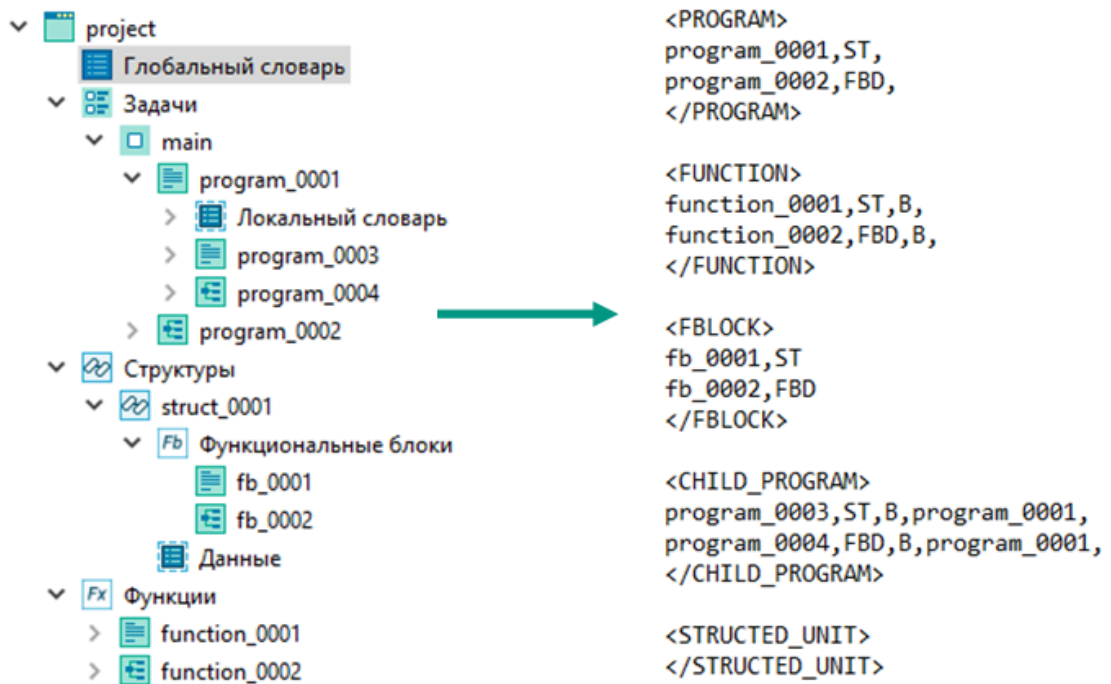


В настоящее время не рекомендуется использовать описанные в данном разделе возможности, так как они находятся на стадии тестирования и доработки, их выполнение может быть некорректным.

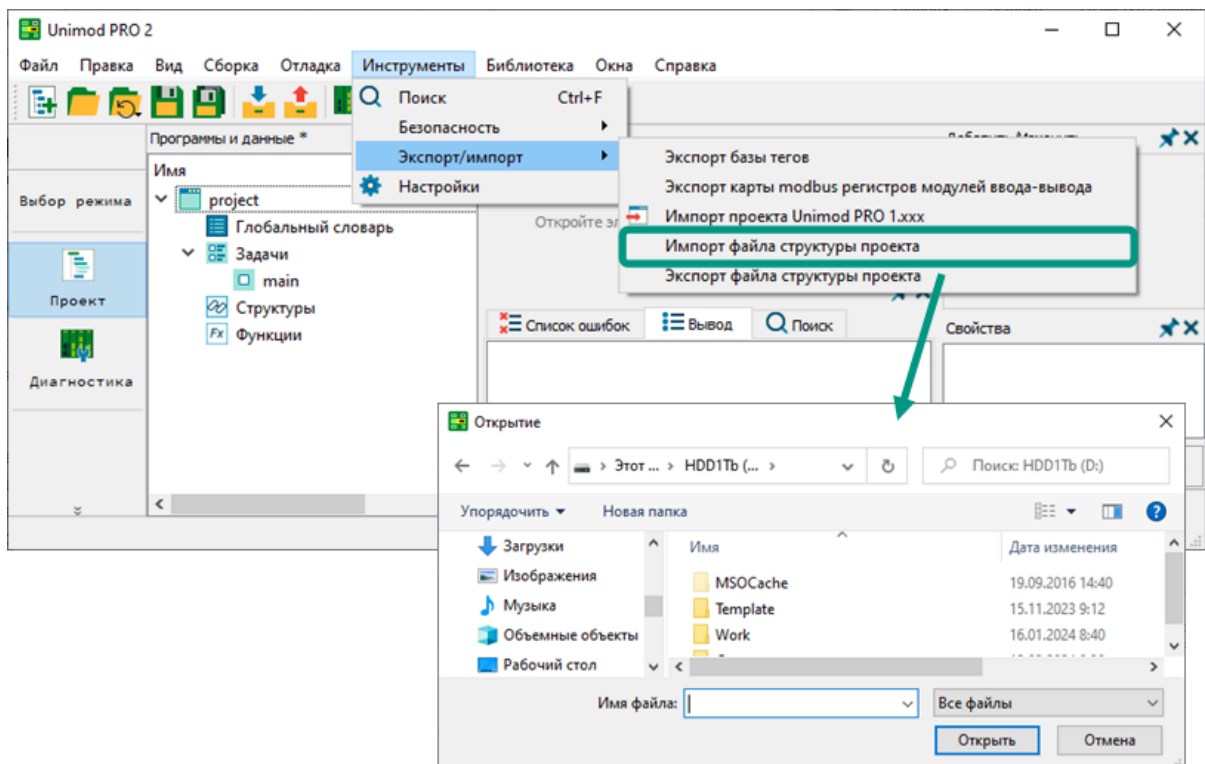
Для экспорта структуры проекта выберите пункт меню «Инструменты → Экспорт/импорт → Экспорт файла структуры проекта», в открывшемся окне выберите каталог для сохранения и введите имя файла, после чего нажмите кнопку «Сохранить».



Файл сохраняется с расширением «*.txt» и содержит список всех программных объектов технологического приложения без их содержимого.

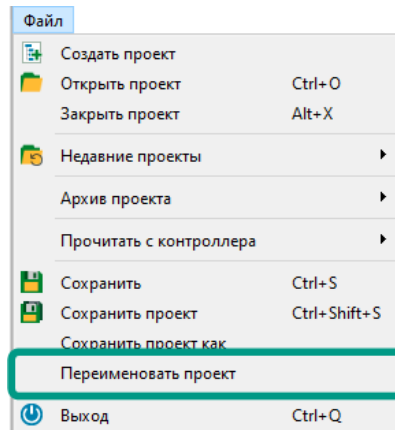


Для импорта структуры проекта выберите пункт меню «Инструменты → Экспорт/импорт → Импорт файла структуры проекта», в открывшемся окне выберите каталог для сохранения и введите имя файла, после чего нажмите кнопку «Сохранить».



Переименование проекта


Для переименования проекта выберите пункт меню «Файл → Переименовать проект» → введите новое название проекта с учетом правил, указанных в разделе [Создание проекта](#), → нажмите клавишу «Enter».

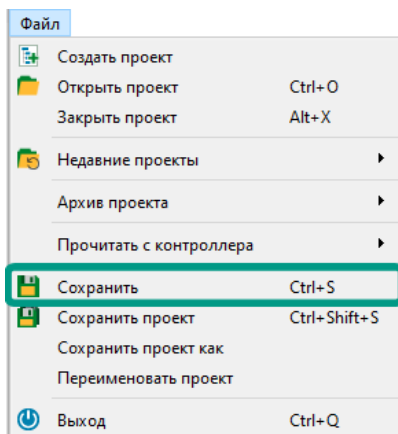



Альтернативные варианты:

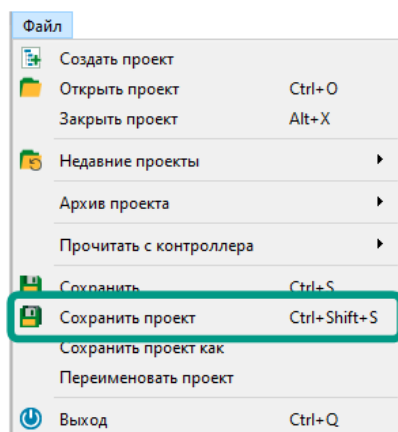
- нажмите правой кнопкой мыши по названию проекта на панели «Программы и данные» → из контекстного меню выберите пункт «Переименовать»;
- нажатием левой кнопки мыши выделите название проекта → нажмите клавишу «F2».

6.8 Сохранение и копирование

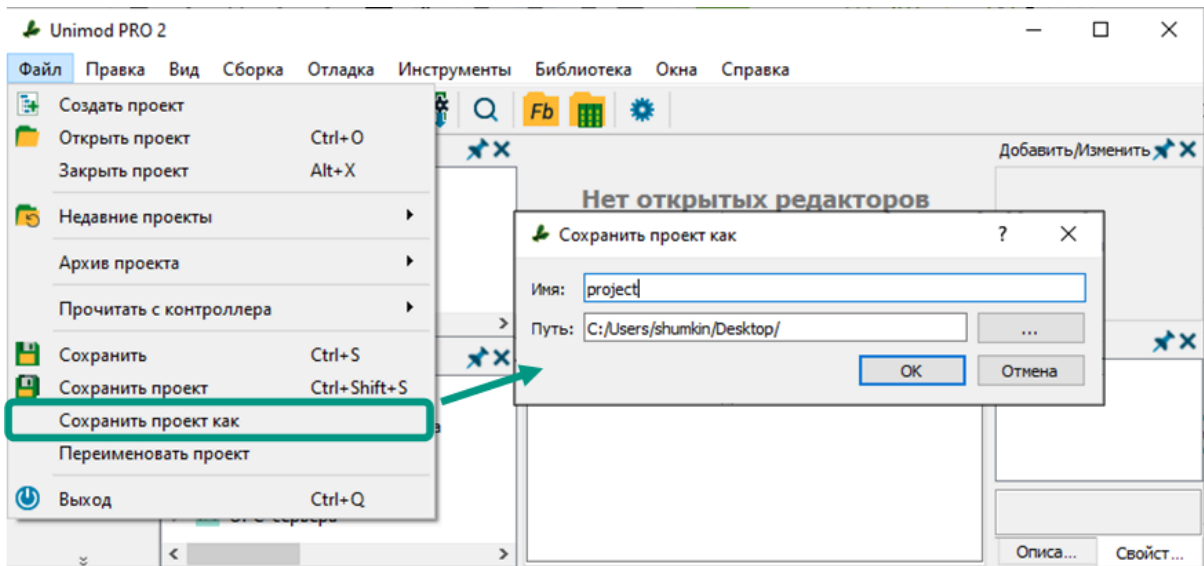
Для сохранения изменений в активном редакторе, словаре или на активной форме настройки параметров, нажмите кнопку  на панели быстрого доступа, выберите пункт меню «Файл → Сохранить» или нажмите сочетание клавиш «Ctrl + S». При этом изменения, внесенные в другие элементы проекта, сохранены не будут.



Для сохранения всех внесенных в проект изменений нажмите кнопку  на панели быстрого доступа, выберите пункт меню «Файл → Сохранить проект» или нажмите сочетание клавиш «Ctrl + Shift + S».



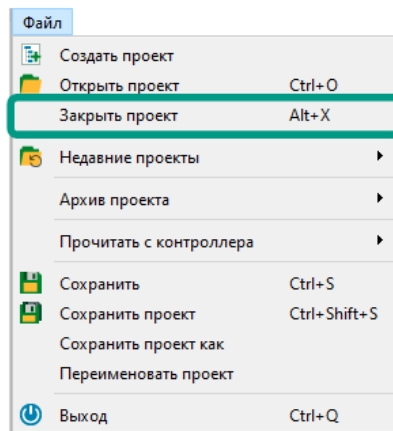
Для создания копии проекта нажмите пункт меню «Файл → Сохранить проект как...» → в открывшемся окне введите имя нового проекта и укажите путь к каталогу проекта в операционной системе с учетом правил и рекомендаций, указанных в разделе [Создание проекта](#), → нажмите кнопку «Ок».



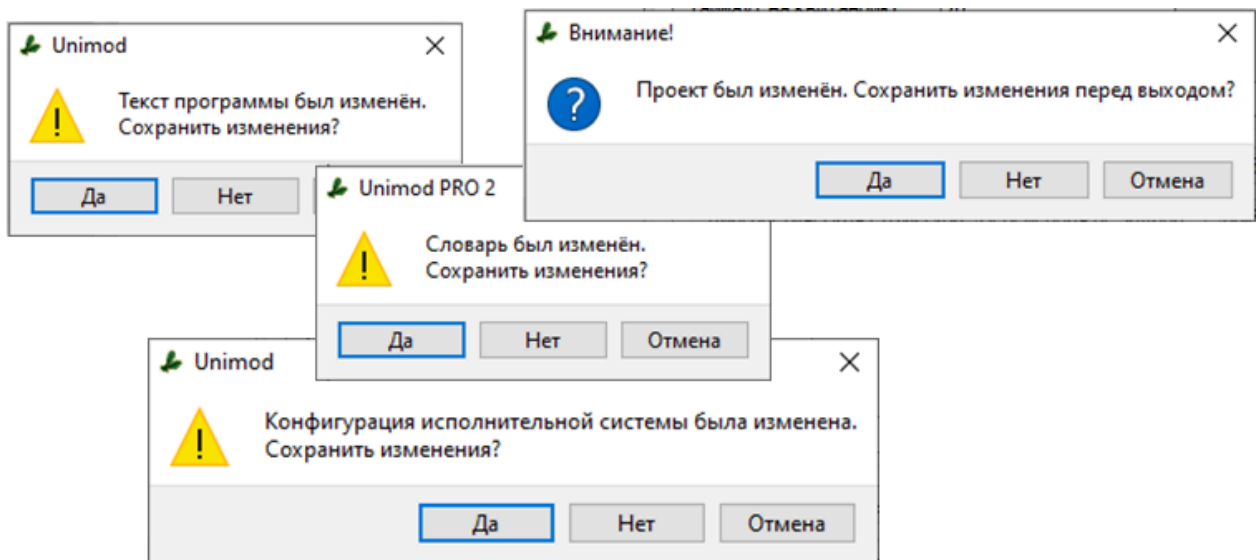
После сохранения в указанном каталоге будет создан подкаталог с полной копией проекта. При этом в среде разработки Unimod PRO 2 будет открыт и доступен для редактирования именно новый сохраненный проект.

Заккрытие и удаление

Для закрытия текущего проекта выберите пункт меню «Файл → Закрывать проект» или нажмите сочетание клавиш «Alt + X».



При наличии не сохраненных изменений в проекте среда разработки Unimod PRO 2 перед закрытием выведет одно из нижеуказанных предупреждений с предложением сохранить изменения. Для сохранения изменений в проекте перед закрытием нажмите «Да», для закрытия без сохранения нажмите «Нет», для отмены закрытия проекта нажмите «Отмена» или кнопку X.

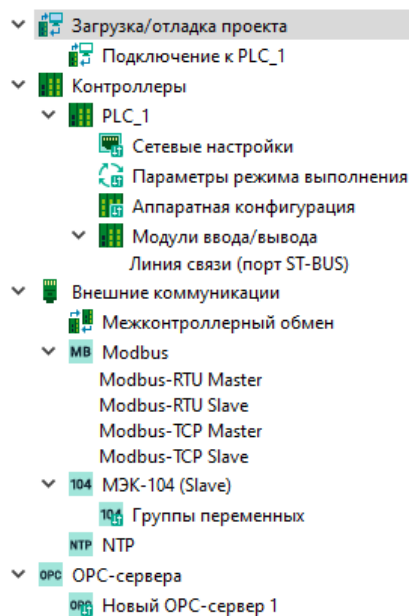


Среда разработки Unimod PRO 2 не предусматривает возможности удаления проектов. Для удаления проекта Unimod PRO 2 из файловой системы воспользуйтесь встроенными средствами операционной системы либо сторонними приложениями.

7 Управление конфигурацией

Конфигурация исполнительной системы Unimod PRO 2 - совокупность параметров для организации работы исполнительной системы. Правильно настроенная конфигурация позволит корректно выполнять разработанное приложение на контроллере и организовать взаимодействие со сторонними устройствами.

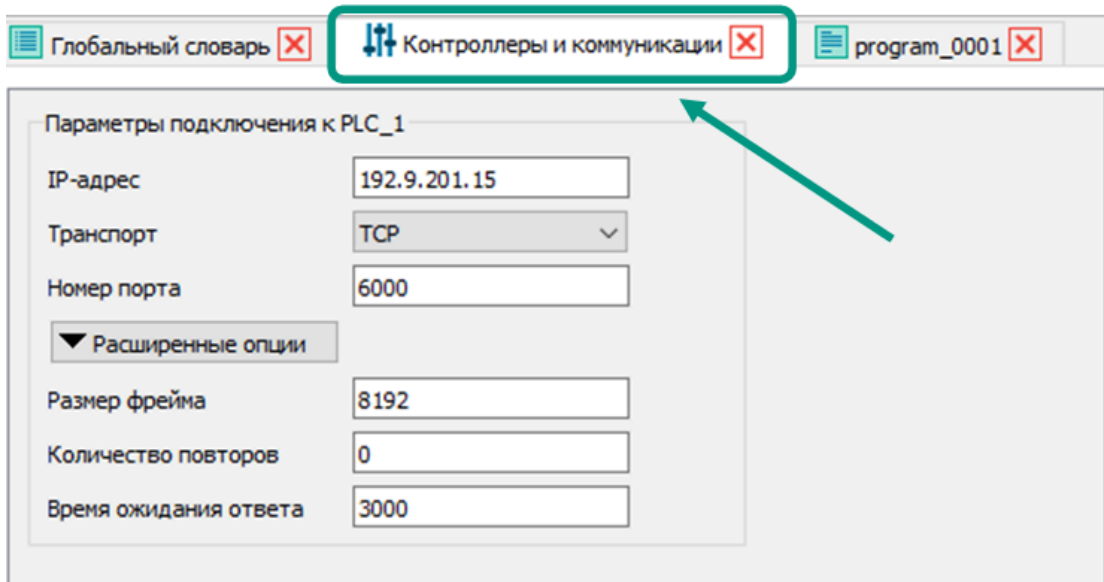
Параметры конфигурации в среде разработки Unimod PRO 2 отображаются на панели «Контроллеры и коммуникации» в виде дерева конфигурации. Минимальная структура дерева конфигурации приведена на следующем изображении.



Для настройки доступны:

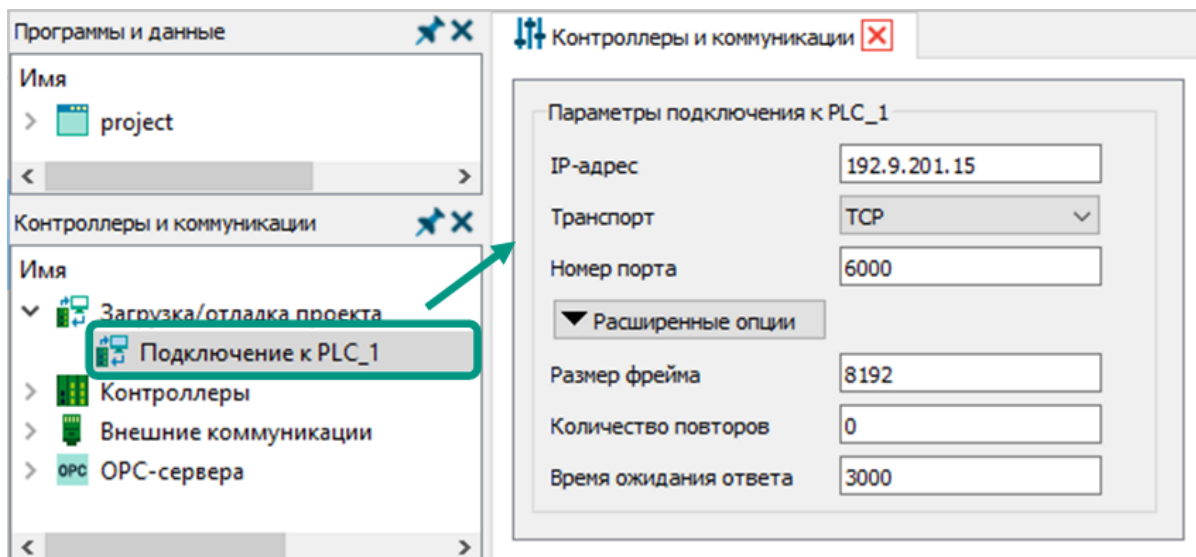
- параметры подключения к контроллеру - раздел «*Загрузка/отладка проекта*»;
- основная информация о контроллере, настройка его параметров и модулей ввода-вывода - раздел «*Контроллеры*»;
- параметры для управления задачами связи с внешними устройствами - раздел «*Внешние коммуникации*»;
- список OPC-серверов и их конфигурационные параметры - раздел «*OPC-сервера*».

Параметры настраиваются на отдельных экранных формах, которые содержат списки, поля и таблицы. Формы закреплены за разделами дерева конфигурации и открываются в рабочем пространстве в виде вкладки «Контроллеры и коммуникации». Для открытия формы выберите требуемый раздел дерева конфигурации нажатием по нему левой кнопкой мыши. Одновременно может быть открыта только одна форма настройки параметров.



7.1 Параметры подключения к контроллеру

Параметры подключения к контроллеру настраиваются в разделе «Загрузка/отладка проекта → Подключение к PLC_1» дерева конфигурации, где «PLC_1» - название контроллера. Изменить название контроллера можно в разделе «Контроллеры» дерева конфигурации.



Минимальный набор параметров для подключения к контроллеру при загрузке и отладке проекта:

- **IP-адрес** контроллера в локальной сети в формате IPv4, должен совпадать с одним из IP-адресов, установленных в сетевых настройках загруженного на контроллер приложения;
- **IP-адрес резервного контроллера** в формате IPv4, параметр отображается только при поддержке резервирования выбранным мастер-модулем;
- **номер порта**, используемый контроллером для взаимодействия со средой разработки;
- **тип протокола передачи данных:** TCP - данные передаются по сети в не зашифрованном виде без аутентификации клиента, SSL - защищенный обмен данными за счет шифрования и аутентификации. При выборе протокола SSL дополнительно нужно указать путь к файлам приватного ключа, приватного и публичного сертификатов.

Параметры подключения к PLC_1

IP-адрес

Транспорт SSL

Номер порта 6000

▼ Расширенные опции

Размер фрейма 8192

Количество повторов 0

Время ожидания ответа 3000

Приватный ключ ...

Приватный сертификат ...

Публичные сертификаты мастеров ...

Дополнительно могут быть указаны параметры:

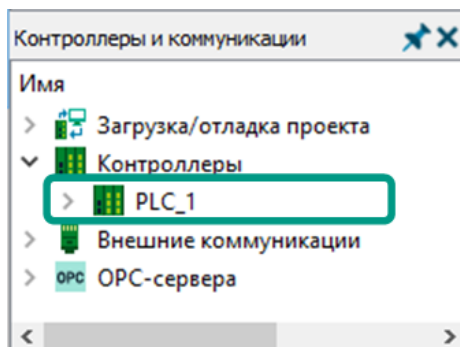
- **размер фрейма** - стандартный размер передаваемого по сети пакета данных в байтах, должен совпадать с размером фрейма, указанным для IP-порта в сетевых настройках контроллера.
- **количество повторов** - количество попыток подключения к контроллеру, при превышении которых будет выведена ошибка. Одна попытка - один период времени ожидания ответа.
- **время ожидания ответа** - период времени в миллисекундах между отправкой запроса на подключение и получением подтверждения от контроллера. Если в указанный период ответ от контроллера не поступил, одна попытка подключения считается завершенной.



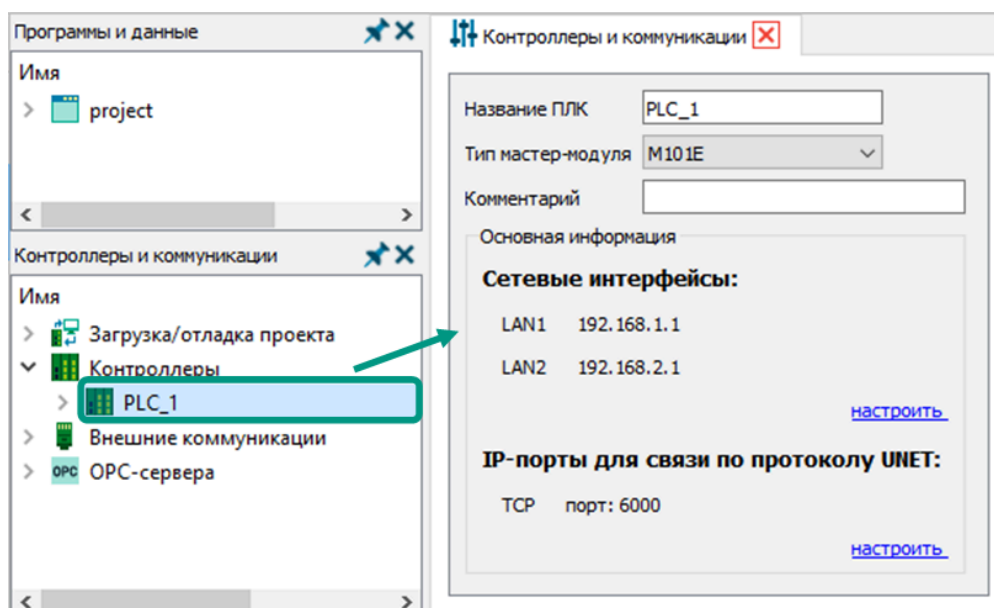
Значения IP-адресов в параметрах подключения и сетевых настройках могут различаться. В параметрах подключения указывается значение одного из текущих IP-адресов контроллера, а в сетевых настройках - значения IP-адресов, которые будут установлены на сетевые интерфейсы контроллера только после загрузки приложения. После загрузки подключение к контроллеру для загрузки и отладки будет осуществляться по одному из IP-адресов, указанных в сетевых настройках загруженного приложения.

7.2 Контроллер

В разделе «Контроллеры» дерева конфигурации содержится один контроллер, для которого создается технологическое приложение. Данный контроллер добавляется в проект по умолчанию с именем «PLC_1».



Основная информация о контроллере содержится в форме настройки параметров. Для открытия формы нажмите левой кнопкой мыши по контроллеру в дереве конфигурации.



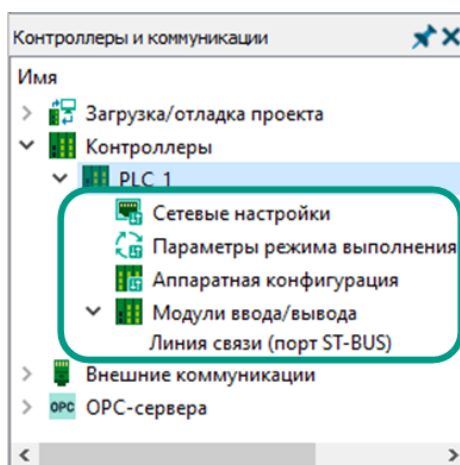
Доступные для настройки параметры:

- **название контроллера** - уникальное имя контроллера в пределах проекта. В имени допустимы латинские буквы, цифры и символы подчеркивания, при этом имя должно начинаться с латинской буквы и не превышать 128 символов.


- **тип мастер-модуля** - определяет модель контроллера, для которого создается приложение. Доступные типы выбираются из раскрывающегося списка, а описание поддерживаемых мастер-модулей приведено в разделе [Доступные для программирования контроллеры](#).
- **комментарий** - поле ввода краткого описания и назначения контроллера;

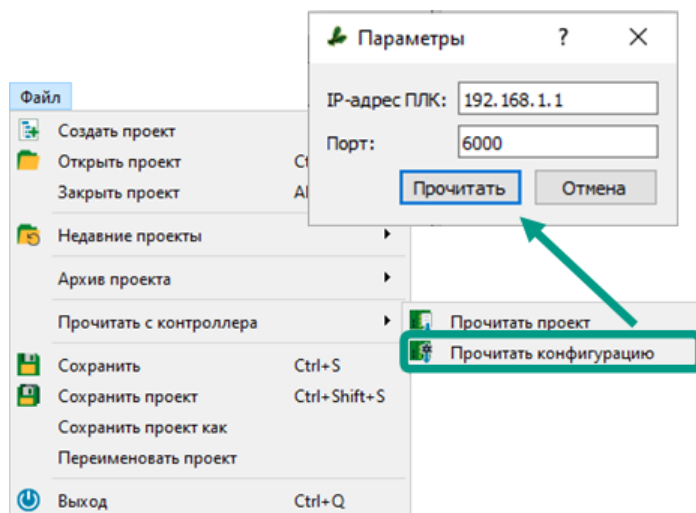
На панели «Основная информация» отображаются параметры сетевых настроек контроллера. Для перехода к разделу сетевых настроек и изменения параметров выберите соответствующий раздел в дереве конфигурации или нажмите на гиперссылку «настроить».

Параметры для настройки контроллера разделены на подразделы, которые отображаются в виде вложенного списка в дереве конфигурации.



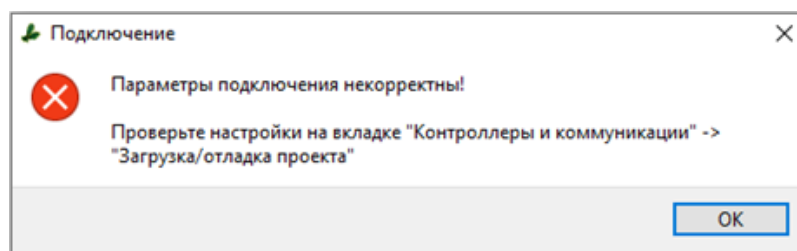
В среде разработки поддерживается возможность загрузки параметров конфигурации исполнительной системы из контроллера. Для этого тип реального контроллера должен совпадать с выбранным типом в приложении. Поддерживаемые для программирования типы контроллеров описаны в разделе [Системные требования](#).

Для загрузки параметров конфигурации выберите пункт меню «Файл → Прочитать с контроллера → Прочитать конфигурацию» или нажмите кнопку  на панели быстрого доступа → в открывшемся окне введите IP-адрес контроллера и номер порта для подключения → нажмите кнопку «Прочитать». В качестве альтернативного варианта в дереве конфигурации кликните правой кнопкой мыши по имени контроллера и из контекстного меню выберите пункт «Прочитать конфигурацию с контроллера».

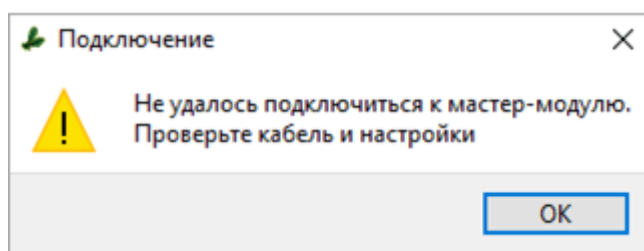


Возможные ошибки и предупреждения при чтении конфигурации с контроллера

При отсутствии параметров подключения будет выведена ошибка. Повторите операцию с указанием IP-адреса и номера порта контроллера.

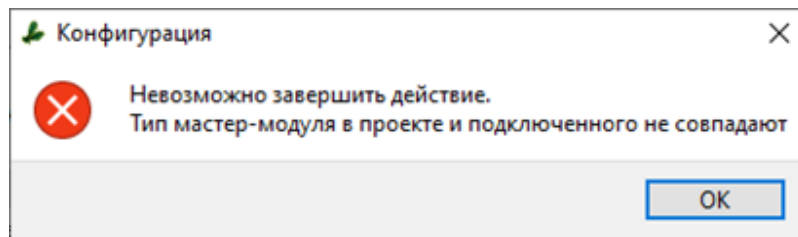


При неправильно заданных параметрах подключения или отсутствии связи с контроллером будет выведена ошибка. В данном случае повторите попытку и укажите корректные IP-адрес и номера порта контроллера. Если ошибка сохраняется при корректных параметрах, убедитесь в наличии связи с контроллером в режиме диагностики.



При несовпадении типов выбранного в проекте и подключенного контроллера (мастер-модуля), в процессе чтения конфигурации будет выведено сообщение об ошибке. Для определения типа подключенного контроллера перейдите в режим «Диагностики» → опросите доступные в сети устройства →

определите тип контроллера по значению столбца «*Тип модуля*». Подробнее в разделе [Режим диагностики](#).

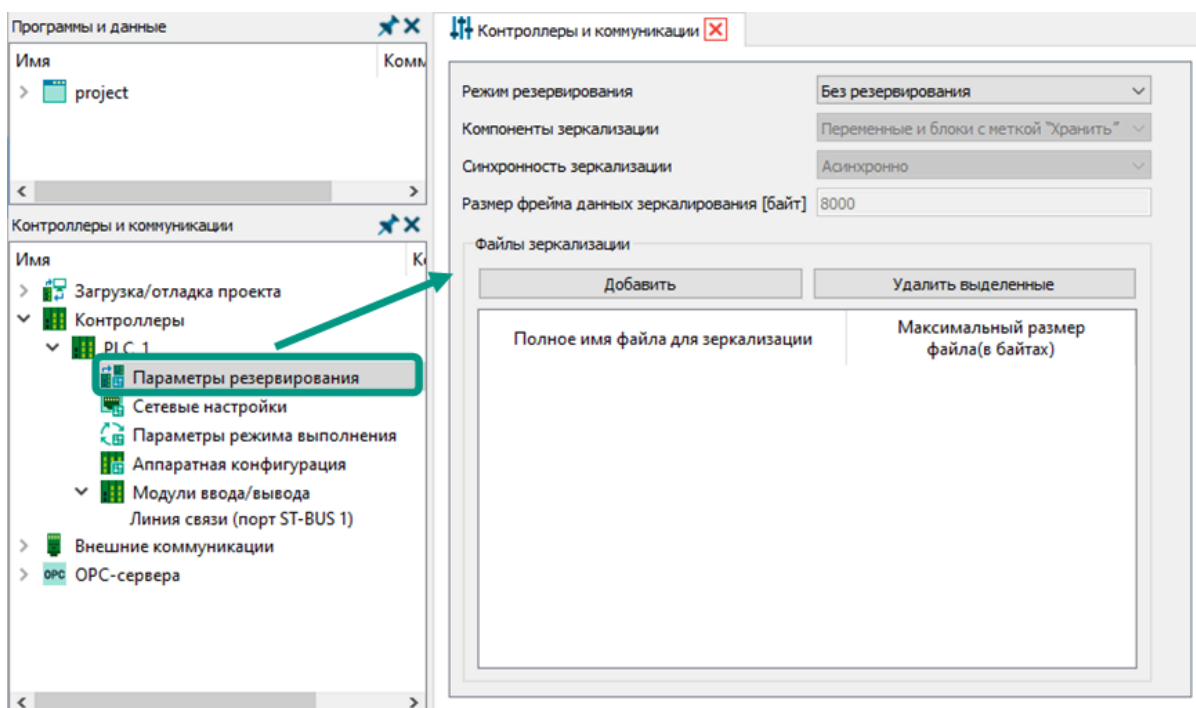


7.2.1 Параметры резервирования

Резервирование позволяет непрерывно управлять технологическими процессами при неисправности отдельных устройств или обрывах линий связи за счет их дублирования. Например, при неисправности мастер-модуля - управление автоматически будет переведено на резервный мастер-модуль, а при обрыве одной линии связи - данные будут отправляться по резервной. Это повышает надежность автоматизированных систем управления технологическими процессами.

Для бесперебойного перехода управления с основного мастер-модуля на резервный важно поддерживать на резервном устройстве идентичную версию приложения и необходимый набор данных технологического процесса. Процесс передачи данных на резервный модуль называется зеркалированием. Совпадение версий приложения достигается постоянной синхронизацией устройств, а необходимые для зеркалирования данные настраиваются в среде разработки Unimod PRO 2.

Для открытия формы настройки параметров резервирования выберите раздел «*Параметры резервирования*» в дереве конфигурации. Данный раздел не отображается в дереве конфигурации, если выбранный тип мастер-модуля не поддерживает резервирование.



Параметры резервирования:

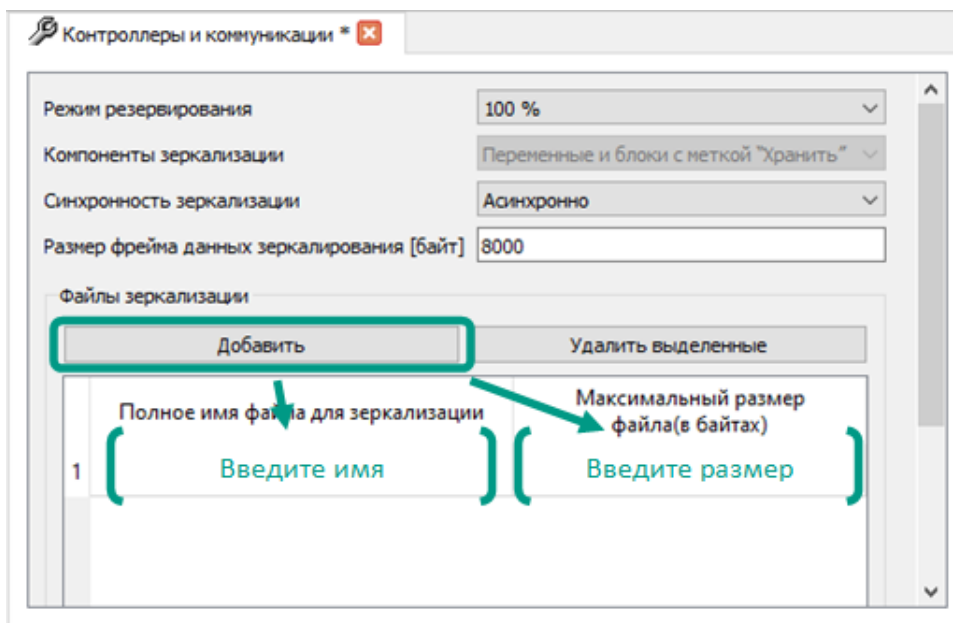
- **режим резервирования** - режим синхронизации данных между основным и резервным контроллерами:

Режим резервирования	Описание
Без резервирования	Данные не зеркалируются. При неисправности мастер-модуля процесс управления будет прерван, а данные могут быть потеряны.
Процессорной части	Зеркалируются все переменные с установленным атрибутом «Хранить/резерв». Параметр используется при конфигурации, где присутствуют основной и резервный мастер-модули, а модули ввода-вывода не резервируются. В случае неисправности основного мастер-модуля, резервный модуль продолжит работу с теми же модулями ввода-вывода, с которыми работал основной.
100%	Зеркалируются все переменные с установленным атрибутом «Хранить/резерв». Параметр используется при полном резервировании, когда резервируется мастер-модуль и модули ввода-вывода. В случае неисправности основного мастер-модуля, резервный модуль продолжит работу с подключенными к нему резервными модулями ввода-вывода.

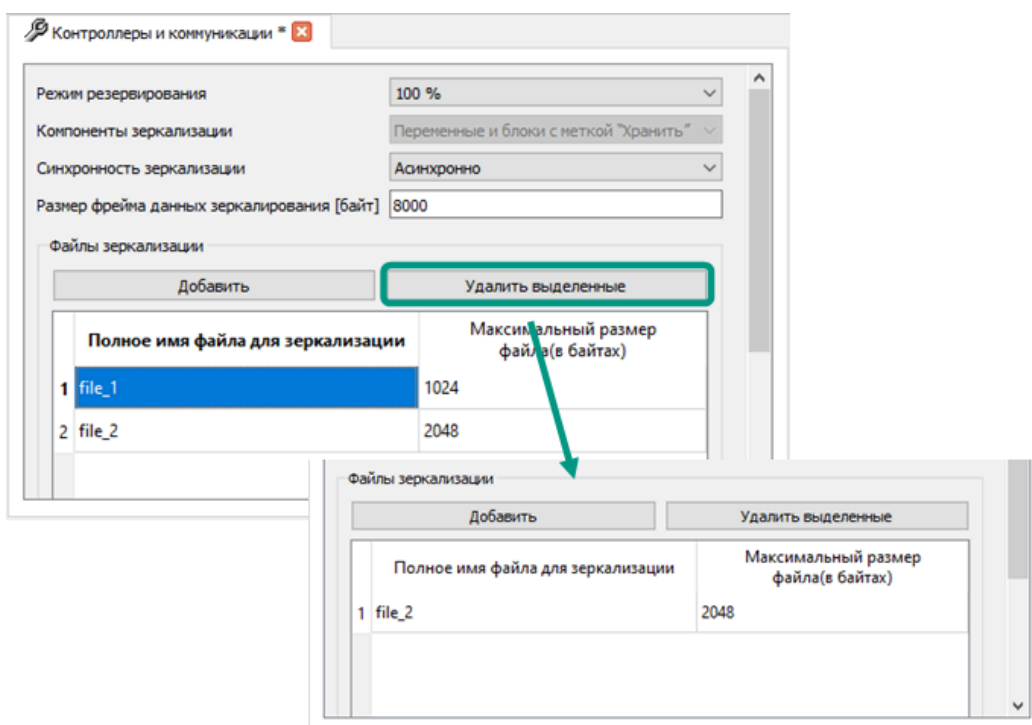
- **компоненты зеркализации** - зеркалируются все переменные приложения с установленным атрибутом «Хранить/резерв».
- **синхронность зеркализации** - определяет момент передачи зеркалируемых данных на резервный контроллер и влияет на время цикла технологической программы. При синхронизации «в каждом цикле», значения переменных с атрибутом «Хранить/резерв» будут передаваться на резервный мастер-модуль в конце каждого цикла приложения, а запуск нового цикла начнется только после полной отправки зеркалируемых данных, что увеличивает время цикла. При «асинхронном обмене» в конце каждого цикла приложения параллельно запускаются передача зеркалируемых данных на резервный мастер-модуль и новый цикл приложения. В таком случае время цикла не изменяется.
- **размер фрейма данных зеркалирования** - размер одного передаваемого по сети пакета зеркалируемых данных.

Если технологическая программа в процессе выполнения использует вспомогательные файлы, хранящиеся в файловой системе контроллера, то они также могут зеркалироваться. Для этого необходимо указать их в таблице «Файлы зеркализации».

Для добавления зеркалируемого файла нажмите кнопку «Добавить» → новая запись с пустыми ячейками будет добавлена в конец списка → введите в соответствующие ячейки имя файла в формате «/unimod/usr/<имя файла>.<расширение>» и максимальный размер файла в байтах. Если фактический размер файла превышает заданный максимальный размер, то зеркалироваться будет только начальная часть файла в пределах указанного размера.



Для удаления зеркалируемого файла выделите его в списке → нажмите кнопку «Удалить выделенные». В результате запись с именем файла будет удалена из таблицы «Файлы зеркализации», после загрузки приложения в мастер-модуль зеркализация файла прекратится, но сам файл сохранится в памяти контроллера.



7.2.2 Сетевые настройки

Раздел сетевых настроек предназначен для настройки сетевых интерфейсов контроллера и режимов удаленного доступа. Указанные в данном разделе IP-адреса и порты будут использоваться для подключения к контроллеру после загрузки в него технологического приложения, например, для чтения проекта или конфигурации, отладки приложения или загрузки нового, а также в режиме диагностики. Поэтому сетевые настройки загруженного в контроллер приложения должны совпадать с параметрами подключения к контроллеру в разделе «Загрузка/отладка проекта → Подключение в PLC_1».

The screenshot shows the 'Контроллеры и коммуникации' (Controllers and Communications) configuration window. It is divided into three main sections, each highlighted with a callout box:

- Параметры удаленного доступа (Remote Access Parameters):** This section includes settings for remote access to the web configurator, protocol (HTTP and HTTPS), and access via FTP and Telnet. All access options are set to 'разрешен только в технологическом режиме' (allowed only in technological mode).
- Параметры сетевых интерфейсов (Network Interface Parameters):** This section shows settings for four LAN interfaces (LAN1 to LAN4). Each interface has a specific IP address and a subnet mask of 255.255.255.0. The gateway address is set to 0.0.0.0.
- Параметры IP-портов (IP Port Parameters):** This section shows a table of IP ports for UNET communication. The table has columns for 'Номер порта' (Port Number), 'Шифрование' (Encryption), 'Размер фрейма' (Frame Size), and 'Ключи/сертификаты (Только для SSL)' (Keys/Certificates (Only for SSL)).

Примечание: IP-адреса 192.168.200.____ - 192.168.202.____ и __.250 - __.254 зарезервированы для внутреннего использования и не должны использоваться мастер-модулем или подключенными устройствами

Номер порта	Шифрование	Размер фрейма	Ключи/сертификаты (Только для SSL)
1	Нет (TCP)	8192	

7.2.2.1 Параметры удаленного доступа

На панели настраиваются разрешения для каждого варианта удаленного доступа к контроллеру.

Удаленный доступ	
Доступ к web-конфигуратору	запрещен во всех режимах
Протокол доступа к Web-конфигуратору	HTTP и HTTPS
Доступ по FTP	разрешен только в технологическом режиме
Доступ по Telnet	разрешен только в технологическом режиме

Параметры удаленного доступа:

- **доступ к web-конфигуратору** - режим доступа к веб-конфигуратору, который используется для настройки параметров исполнительной системы контроллера через веб-интерфейс. Не все контроллеры поддерживают доступ через веб-конфигуратор, перед использованием требуется ознакомиться с руководством по эксплуатации на конкретное устройство.
- **протокол доступа к web-конфигуратору** - выбор протокола передачи данных: *HTTP* и *HTTPS* - для соединения с контроллером может использоваться любой из указанных протоколов, *только HTTPS* - используется только протокол HTTPS, в противном случае подключение будет недоступно. По протоколу HTTP данные передаются в открытом виде, а при использовании HTTPS передаваемые данные предварительно шифруются с использованием криптографических протоколов SSL или TLS.
- **доступ по FTP** - параметр настройки режима доступа к файловой системе контроллера. Используется сетевой протокол передачи файлов FTP.
- **доступ по Telnet** - параметр настройки режима доступа к командной оболочке операционной системы, установленной в контроллере. Используется для управления контроллером посредством передачи по сети текстовых управляющих команд по протоколу Telnet.

Контроллеры фирмы TREI могут функционировать в одном из трех режимов:

- **рабочий режим** - основной режим работы контроллера, при котором запускаются исполнительная система и технологическое приложение. При этом возможны загрузка, отладка и выполнение технологического приложения в режиме реального времени, осуществляется программное управление каналами ввода-вывода, информационный обмен контроллера с внешними устройствами и SCADA-системами по различным интерфейсам, в зависимости от модели мастер-модуля может быть доступен веб-конфигуратор.

- **технологический режим** - в данном режиме на контроллере запускается только операционная система с поддержкой сетевых интерфейсов, а контроллер не участвует в управлении технологическим процессом. При этом исполнительная система и приложение не выполняются, управление каналами ввода-вывода и обмен данными с внешними устройствами не производится. Данный режим используется для выполнения сервисных работ: безопасной настройки исполнительной системы контроллера, установки и обновления программного обеспечения, копирования файловых данных и диагностики аппаратных средств.

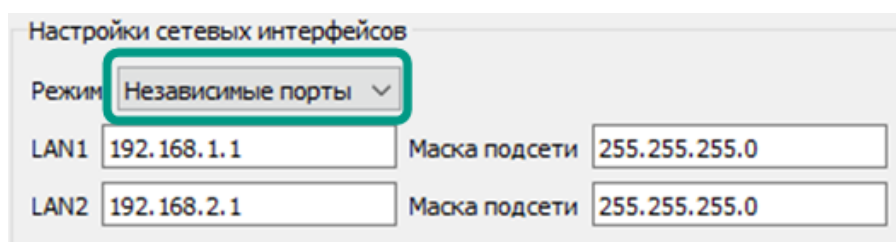
Режим работы определяется по комбинации включенных переключателей на контроллере и должен быть задан до подачи питания на контроллер. Комбинации переключателей для управления режимами описаны в руководствах по эксплуатации и могут отличаться в зависимости от модели.

7.2.2.2 Параметры сетевых интерфейсов

Настройка сетевых интерфейсов состоит из настройки режима работы портов, IP-адресов и масок подсетей. Возможность выбора режима работы, а также количество доступных IP-адресов зависят от типа выбранного мастер-модуля. IP-адрес сетевого шлюза указывается, если контроллер будет использоваться в качестве сетевого шлюза одной из сетей.

Режимы работы сетевых портов:

- **независимые порты** - в данном режиме каждый из портов на мастер-модуле будет представлять собой отдельную сеть, при этом мастер-модуль будет отображаться в сети как два независимых Ethernet-устройства с различными MAC и IP-адресами;
- **коммутатор** - мастер-модуль выполняет функции коммутатора и позволяет пропускать поступающие Ethernet-пакеты через все имеющиеся разъемы LAN-портов.
- **MRP Slave** - мастер-модуль в роли ведомого коммутатора (Slave) в топологии кольцевого резервирования сети на основе протокола MRP. В данной роли модуль отслеживает и пересылает сообщения, а также передает ведущему модулю (Master) информацию об ошибках.
- **MRP Master** - мастер-модуль в роли ведущего коммутатора (Master) в топологии кольцевого резервирования сети. В данной роли мастер-модуль будет контролировать работоспособность топологии, а при обрыве кольца активирует резервное соединение и сообщит Slave-коммутаторам о необходимости перенаправления трафика по резервному пути.



Настройка сетевых интерфейсов

Режим	Независимые порты	
LAN1	192.168.1.1	Маска подсети 255.255.255.0
LAN2	192.168.2.1	Маска подсети 255.255.255.0

Указанные в сетевых настройках IP-адреса будут использоваться для соединения с контроллером после загрузки в него технологического приложения. При включенном режиме резервирования часть IP-адресов задействована под организацию каналов резервирования и не может быть настроена пользователем.

Без режима резервирования

Настройки сетевых интерфейсов			
LAN1	<input type="text" value="192.168.1.1"/>	Маска подсети	<input type="text" value="255.255.255.0"/>
LAN2	<input type="text" value="192.168.2.1"/>	Маска подсети	<input type="text" value="255.255.255.0"/>
LAN3	<input type="text" value="192.168.3.1"/>	Маска подсети	<input type="text" value="255.255.255.0"/>
LAN4	<input type="text" value="192.168.4.1"/>	Маска подсети	<input type="text" value="255.255.255.0"/>

Настройки сетевых интерфейсов			
	В режиме "Основной"	В режиме "Резервный"	
LAN1	<input type="text" value="192.168.1.1"/>	<input type="text" value="192.168.3.1"/>	Маска подсети <input type="text" value="255.255.255.0"/>
LAN2	<input type="text" value="192.168.2.1"/>	<input type="text" value="192.168.4.1"/>	Маска подсети <input type="text" value="255.255.255.0"/>
LAN3	для резервирования, не настраивается		
LAN4	для резервирования, не настраивается		

В режиме резервирования

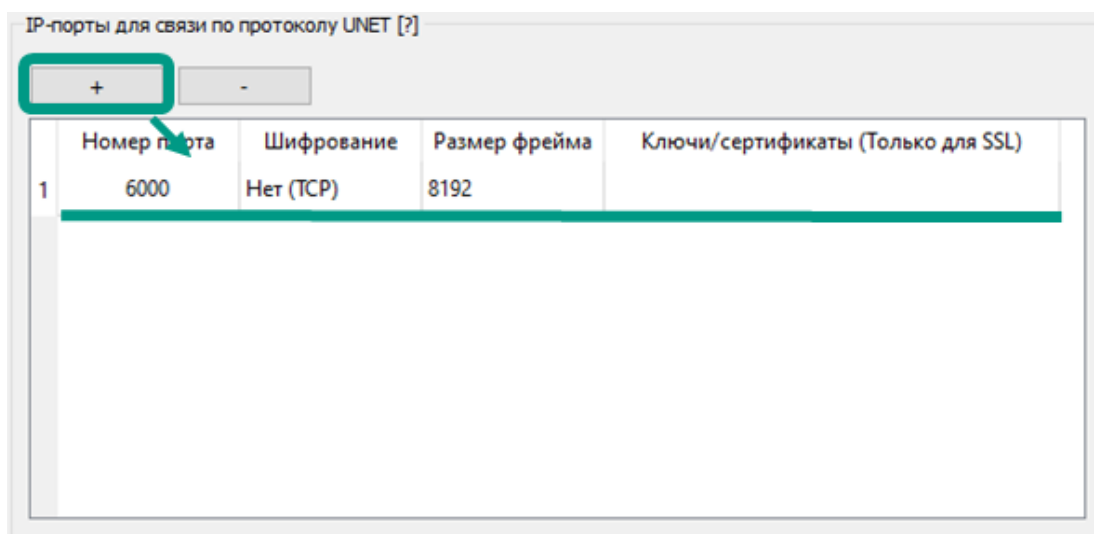


Значения IP-адресов в параметрах подключения и сетевых настройках могут различаться. В параметрах подключения указывается значение одного из текущих IP-адресов контроллера, а в сетевых настройках - значения IP-адресов, которые будут установлены на сетевые интерфейсы контроллера только после загрузки приложения. После загрузки подключение к контроллеру для загрузки и отладки будет осуществляться по одному из IP-адресов, указанных в сетевых настройках загруженного приложения.

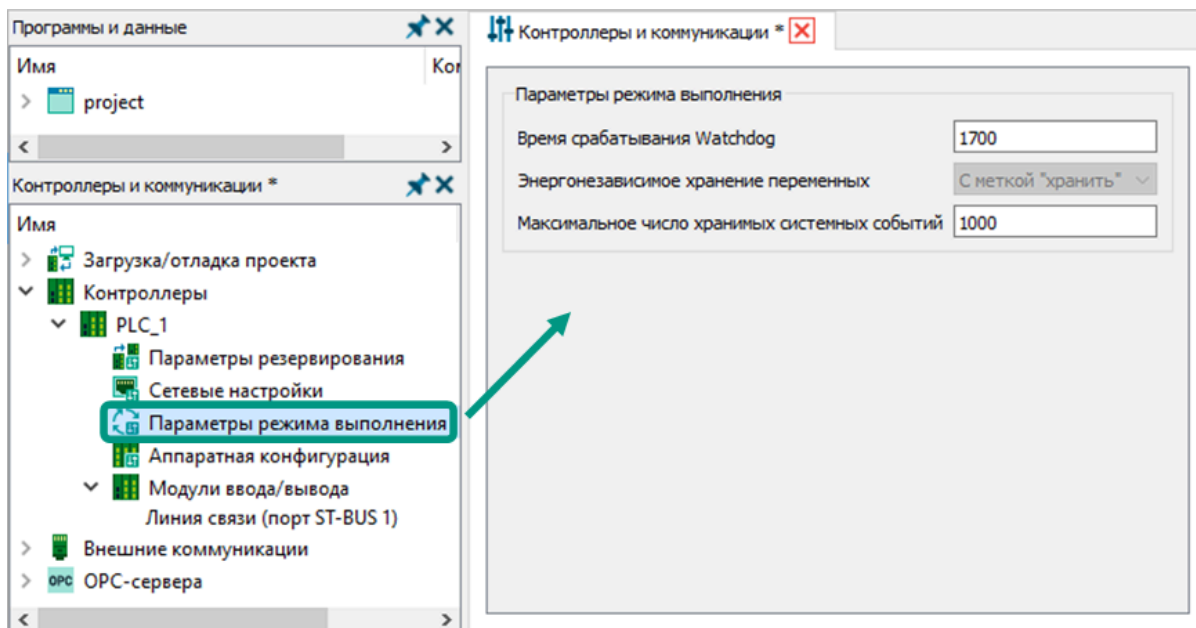
7.2.2.3 Параметры IP-портов

IP-порт и IP-адрес используются для соединения с контроллером после загрузки в него технологического приложения, поэтому в сетевых настройках должен быть указан минимум один порт. Доступные для подключения порты отображаются в виде таблицы. К параметрам порта относятся его номер, протокол передачи данных и размер фрейма. Протокол UNET используется для связи мастер-модуля со средой разработки Unimod PRO 2.

Для добавления и удаления IP-портов используйте кнопки и . При добавлении новая запись будет добавлена в список IP-портов со значениями по умолчанию, которые доступны для редактирования.



7.2.3 Параметры режима выполнения



Параметры режима выполнения:

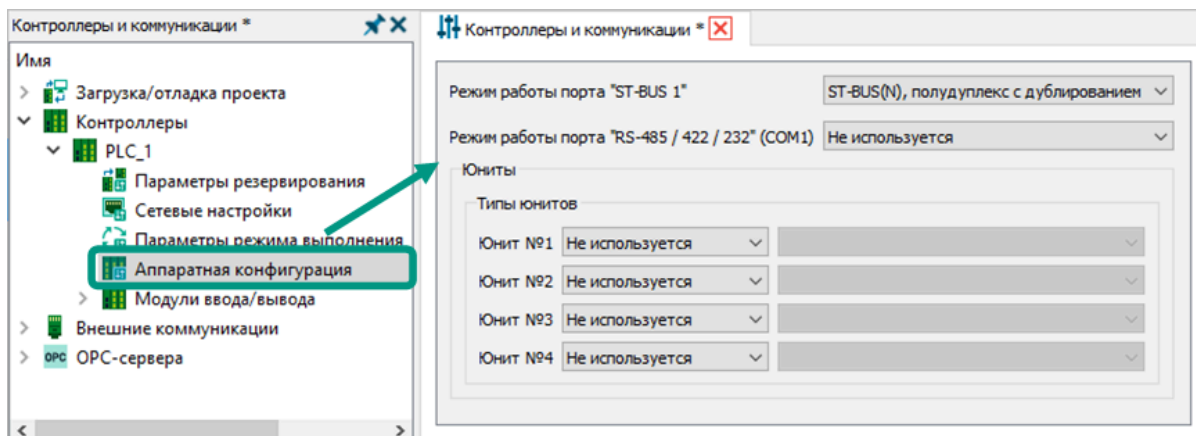
- **время срабатывания Watchdog** - период времени в миллисекундах, по истечении которого будет произведен программный сброс контроллера. Период отсчитывается от момента возникновения невосстанавливаемого сбоя в работе или зависании приложения.
- **энергонезависимое хранение переменных** - определяет переменные, значения которых будут сохраняться в энергонезависимой памяти контроллера и восстанавливаться при сбоях в работе. Параметр по умолчанию настроен на хранение переменных с установленным атрибутом «Хранить» и недоступен для изменения.
- **максимальное число хранимых системных событий** - количество системных событий, сохраняемых в памяти контроллера, по умолчанию равно 1000 записей. При превышении указанного количества новые системные события будут записываться за счет удаления самых старых записей.

7.2.4 Аппаратная конфигурация

Под аппаратной конфигурацией подразумевается количество и типы расположенных на мастер-модуле портов (интерфейсов), обеспечивающих взаимодействие с модулями ввода-вывода и другими внешними устройствами. Порты устанавливаются в модули на заводе, их количество зависит от комплектации мастер-модуля и не может быть изменено в процессе эксплуатации. Для увеличения количества портов на некоторых моделях контроллеров фирмы TREI дополнительно предусмотрены разъемы для подключения юнитов.

Юнит - компактное устройство, при необходимости подключаемое к мастер-модулю для увеличения количества портов. Конструкция юнита построена по принципу переходника: любой юнит подключается к мастер-модулю через универсальный разъем, а соединение между юнитом и физическим каналом связи зависит от типа порта, который требуется добавить. Таким образом, в один и тот же разъем контроллера могут быть подключены юниты с разными типами портов. Количество разъемов под юниты зависит от комплектации мастер-модуля и описано в руководстве по эксплуатации на конкретное устройство.

В среде разработки Unimod PRO 2 все доступные порты отображаются на форме настройки параметров, закрепленной за разделом «Аппаратная конфигурация». Для каждого порта должен быть указан один из доступных режимов работы. В случае подключения модулей ввода-вывода через юниты, необходимо в технологическом приложении предварительно настроить тип используемого юнита и режим его работы.



Режимы работы портов ST-BUS:

- **ST-BUS(N), полудуплекс** - используется одна линия связи для приема и передачи данных, что накладывает ограничение на одновременную работу, поэтому в полудуплексном режиме в один момент времени возможен либо прием, либо передача информации.

- **ST-BUS(N), полудуплекс с дублированием** - аналог полудуплексного режима, но используются две линии связи, по которым передаются одинаковые данные. Вторая линия используется в качестве дублирующей, что повышает надежность системы.

Порт ST-BUS используется для обмена данными между мастер-модулями и модулями ввода-вывода по разработанному компанией АО "ТРЭИ" интерфейсу ST-BUS. Данный интерфейс использует дублирование каналов связи благодаря наличию двух шин RS-485. При неисправности на одной из шин (короткое замыкание, обрыв на линии, неисправность в одном из модулей ввода-вывода), обмен не прекращается, а ведется по второй исправной шине.

Режимы работы COM-портов «RS-485 / 422 / 232»:

- **не используется** - порт не используется для обмена данными;
- **RS-232 без RTS/CTS** - интерфейс RS-232 без управления потоком, что подразумевает постоянную готовность устройства к приему и передаче данных. Интерфейс RS-232 работает только в дуплексном режиме, что позволяет передавать и принимать информацию одновременно, потому что используются разные физические линии для приема и передачи. Информация по интерфейсу RS-232 передается в цифровом виде логическими 0 и 1.
- **RS-232 с RTS/CTS** - интерфейс RS-232 с аппаратным методом управления потоком: для передачи данных необходимо, чтобы передатчик был готов послать данные, а приемник их принять, обмен начинается только после подтверждения готовности передатчика и приемника;
- **RS-232 с RTR/CTS** - интерфейс RS-232 с аппаратным методом управления потоком: передача может начинаться при подтверждении готовности приемника, без предварительного уведомления о готовности к передаче.
- **RS-422** - интерфейс RS-422, который похож на RS-232, так как работает в режиме дуплекс и позволяет одновременно отправлять и принимать сообщения по отдельным линиям, но использует для этого разницу потенциалов между проводниками. В сети RS-422 может быть только одно передающее устройство и до 10 принимающих устройств.
- **RS-485** - интерфейс RS-485, поддерживающий многоточечную топологию, что позволяет подключать несколько приемников и передатчиков. Интерфейс RS-485 использует дифференциальный сигнал для передачи данных и может работать в дуплексном (используются 4 канала) и полудуплексном (используются 2 канала) режимах.

Типы юнитов:

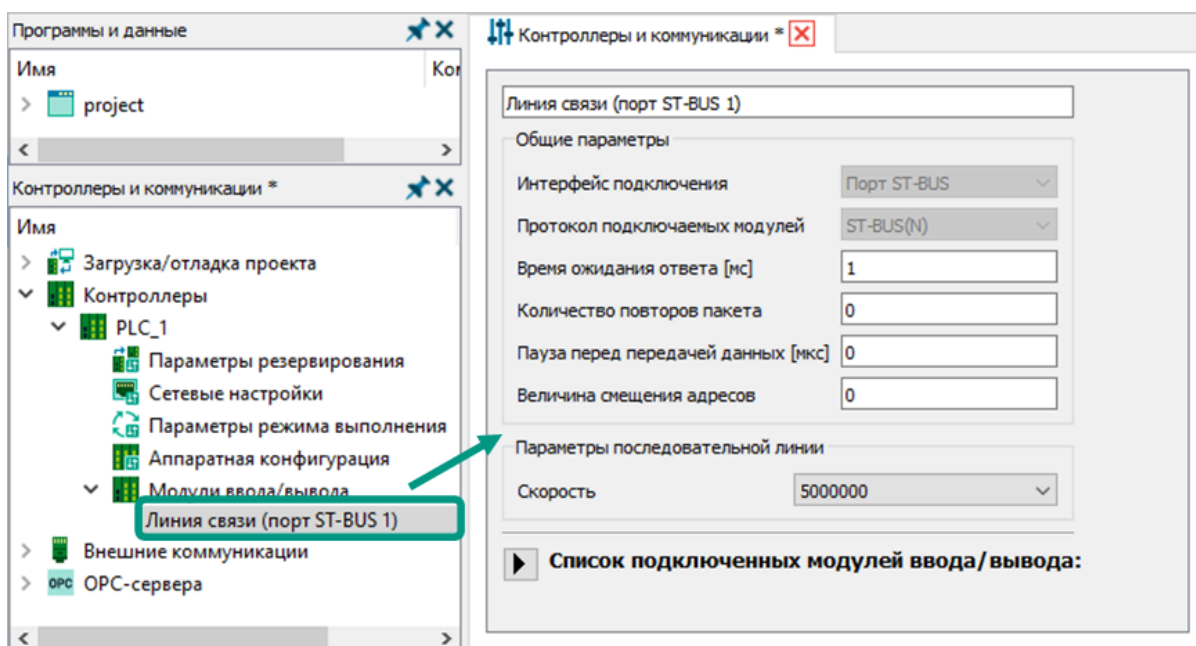
- **не используется** - юнит не используется для обмена данными;
- **UCOM2** - юнит с типом COM-порт;
- **USTB** - юнит с типом порта ST-BUS.

7.2.5 Модули ввода-вывода

Модули ввода-вывода подключаются к физическим интерфейсам датчиков и исполнительных устройств. Мастер-модуль обменивается данными с модулями ввода-вывода циклично. Чтение данных с модулей ввода осуществляется перед каждым циклом технологического приложения. Во время выполнения приложения данные обрабатываются и на их основании формируются выходные данные. Они записываются в модули вывода после завершения цикла приложения.

В контроллерах TREI обмен данными между мастер-модулями и модулями ввода-вывода осуществляется по разработанному компанией АО "ТРЭИ" протоколу ST-BUS. Данный протокол предусматривает обмен данными по специализированной шине ST-BUS, для подключения которой на модулях размещаются порты ST-BUS. Количество портов ST-BUS на модулях может отличаться в зависимости их конфигурации.

Линия связи соответствует одному порту ST-BUS на мастер-модуле, через который может осуществляться обмен данными с модулем ввода-вывода. В дереве конфигурации количество линий связи устанавливается автоматически в зависимости от выбранного типа мастер-модуля. За каждым разделом линии связи закреплена форма настройки параметров.



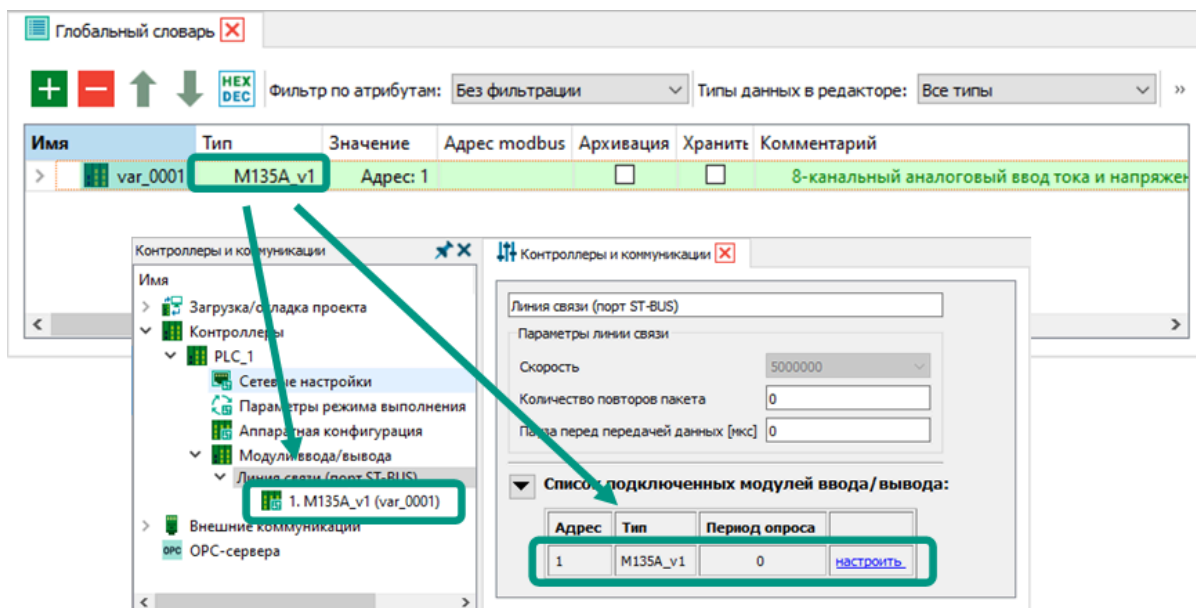
Параметры линии связи:

- **поле ввода** - используется для ввода названия линии связи и кратких пояснений;
- **интерфейс подключения** - по умолчанию выбрано значение «*Порт-ST-BUS*», оно недоступно для изменения, так как обмен данными с модулями ввода-вывода в контроллерах TREI осуществляется по протоколу ST-BUS с использованием соответствующих портов.
- **протокол подключаемых модулей** - по умолчанию выбран протокол ST-BUS(N) - последняя модификация протокола ST-BUS;
- **время ожидания ответа** - период времени, в течение которого ожидается ответ на запрос от модуля ввода-вывода. Регулируется в зависимости от скорости канала связи. При отсутствии в указанный период ответа о получении, пакет отправляется повторно. Если количество попыток отправки пакета превысило значение параметра "количество повторов пакета", в системных событиях фиксируется ошибка, а отправка пакета запускается повторно.
- **количество повторов пакета** - количество попыток отправки пакета, по истечении которых в системных событиях фиксируется ошибка, а отсчет попыток перезапускается.
- **пауза перед передачей данных** - период времени в микросекундах между отправкой данных на модуль ввода-вывода. Данный параметр актуален, если модуль ввода-вывода не успевает обрабатывать поступающие запросы, тогда их принудительно задерживают на Master-устройстве.
- **величина смещения адресов** - задает величину смещения адресов для модулей ввода-вывода. Заданное значение вычитается из номера модуля в проекте, например, если модуль в проекте имеет адрес 129, то мастер-модуль будет обращаться к нему по адресу (129-X). При этом обращение к модулям с адресами 1..X осуществляться не будет.
- **скорость** - задает скорость передачи данных в бит/с, доступные для выбора значения могут отличаться в зависимости от типа мастер-модуля.

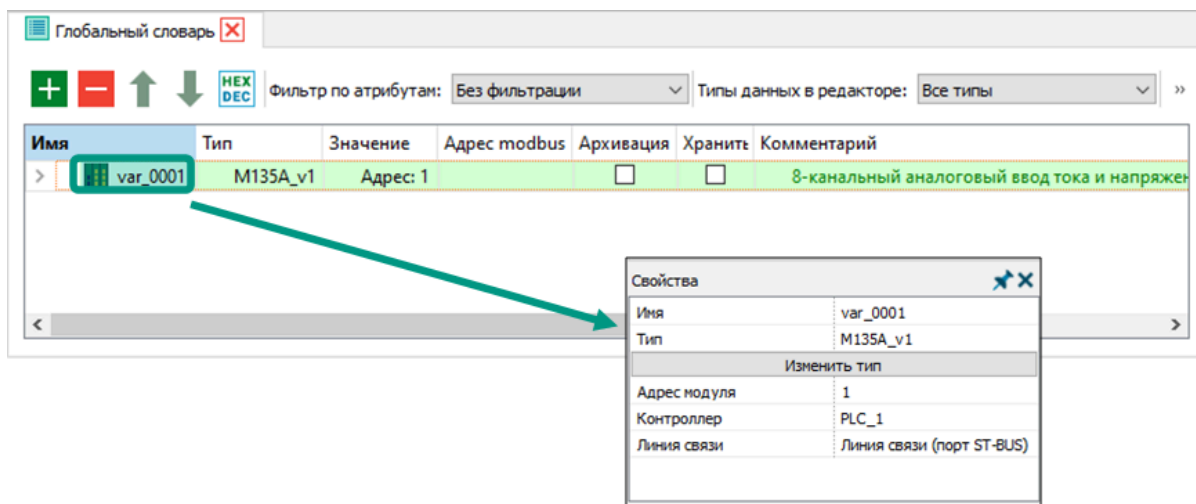
Количество доступных для настройки параметров может меняться в зависимости от выбранного типа мастер-модуля.

Для добавления модуля ввода-вывода добавьте в глобальный словарь переменную с типом структуры, соответствующей модулю ввода-вывода. После выбора структуры модуль ввода-вывода будет добавлен в дерево конфигурации в качестве вложенного элемента линии связи, а также будет отображаться на форме настройки параметров линии связи. Для просмотра всех доступных модульных структур выберите пункт меню «Библиотека – Модули ввода вывода».

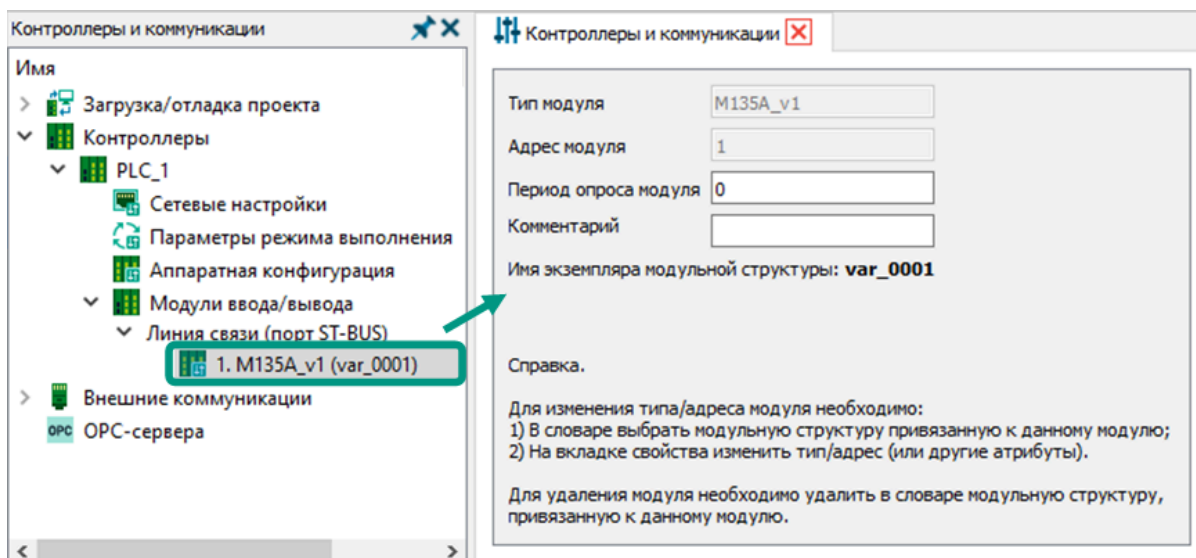
Постфикс «*_X» в названии типа модуля обозначает версию конфигурации, которая соответствует определенным версиям прошивок.



Изменение адреса модуля ввода-вывода осуществляется на панели «Свойства» после выделения переменной модульной структуры в глобальном словаре. В выпадающем списке будет доступен для выбора любой свободный адрес на линии СВЯЗИ.



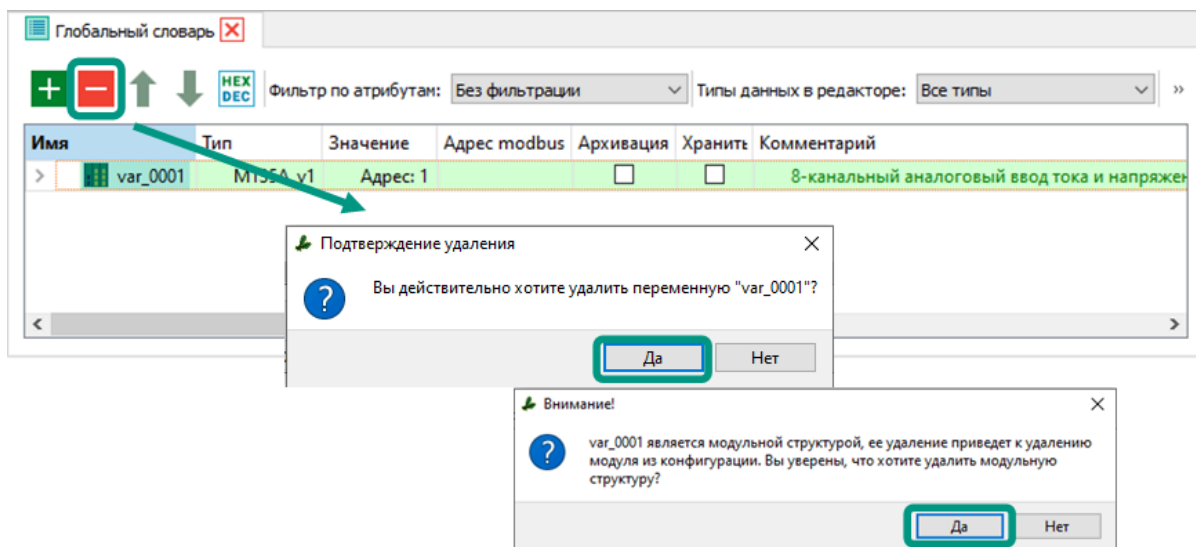
Настройка параметров модуля ввода-вывода производится в отдельной оконной форме.



Параметры модулей ввода-вывода:

- **тип модуля** - совпадает с типом модульной структуры, выбранным для глобальной переменной. Может быть изменен в глобальном словаре или на панели «Свойства». Для изменения в глобальном словаре выберите для соответствующей переменной другой тип модульной структуры. Изменение на панели «Свойства» доступно после выделения переменной в глобальном словаре.
- **адрес модуля** - уникальное целое число от 1 до 254, устанавливается автоматически при добавлении переменной модульной структуры, изменяется на панели «Свойства» после выделения соответствующей переменной в глобальном словаре.
- **период опроса модуля** - период времени в миллисекундах между запросами параметров модулей ввода-вывода. Период равный «0» соответствует опросу модуля в каждом цикле технологической программы.
- **комментарий** - поле для ввода краткого текстового описания назначения модуля ввода-вывода.

Для удаления модуля ввода-вывода в глобальном словаре выделите переменную с типом структуры, соответствующим модулю ввода-вывода, → удалите данную переменную или измените ее тип данных на отличный от модульных структур → в открывающихся окнах подтвердите удаление нажатием кнопок «Да».



Подробное описание модульных структур приведено в документе [«Руководство по программированию»](#).

7.3 Внешние коммуникации

Коммуникационные задачи позволяют исполнительной системе Unimod PRO 2 обеспечивать взаимодействие с устройствами:

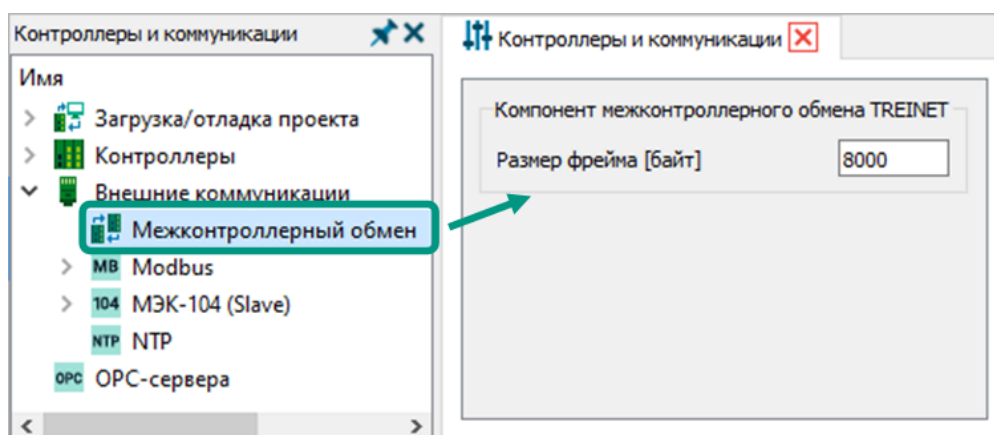
- ПК с установленной средой разработки Unimod PRO 2 для загрузки и отладки технологического приложения, чтения/записи переменных и изменения режима выполнения, метрологической поверки каналов и проверки работоспособности контроллера;
- OPC сервером «TREI OPC DA SERVER» для чтения/записи переменных;
- сервером времени для синхронизации времени;
- другими контроллерами и устройствами для обмена данными.

В цикле исполнительной системы Unimod PRO 2 коммуникационные задачи выполняются после выполнения всех программ целевой задачи. Исключение составляют задачи связи по протоколу Modbus, созданные с использованием функциональных блоков, так как они выполняются в программах целевой задачи. Очередность выполнения определена порядком следования задач в дереве конфигурации.

Для организации информационного обмена между несколькими контроллерами реализован протокол межконтроллерного обмена «*Treinet*», а для связи со сторонними устройствами – протоколы «*Modbus*» и «*МЭК-104 (Slave)*». Настройки связи по протоколам «*Treinet*», «*Modbus*» и «*МЭК-104 (Slave)*» сгруппированы в разделе дерева конфигурации «*Внешние коммуникации*».

7.3.1 Межконтроллерный обмен

Межконтроллерный обмен реализован в исполнительной системе Unimod PRO 2 для резервирования и обмена данными между мастер-модулями. Передача данных осуществляется по разработанному компанией TREI протоколу «*Treinet*». На форме настройки параметров межконтроллерного обмена доступен для редактирования размер одного передаваемого фрейма. Размер задается в байтах и по умолчанию равен 8000.



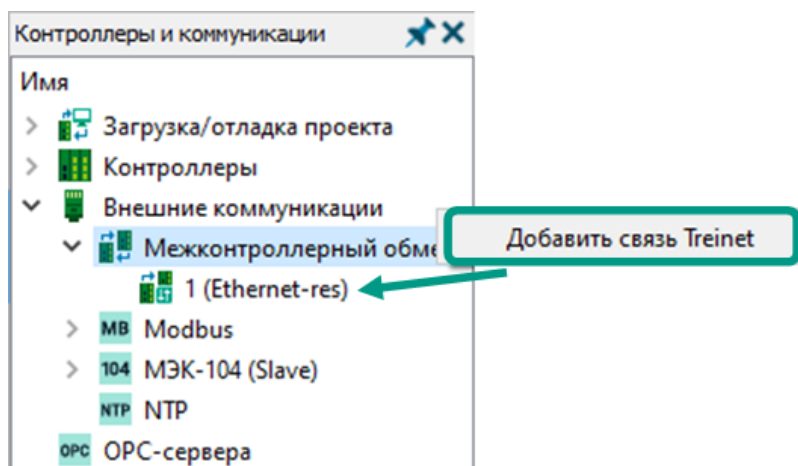
Для организации межконтроллерного обмена создаются задачи связи, в которых указываются параметры соединения и переменные, участвующие в обмене. Задачи межконтроллерного обмена однонаправленные и могут отвечать либо за передачу данных, либо за их прием. Для организации двустороннего обмена нужно создать минимум две задачи связи - одну на вход, другую на выход. Соответствие между несколькими задачами на разных контроллерах устанавливается по номеру порта TCP/IP, в связи с чем номера портов в задачах связи передающего и принимающего контроллера должны совпадать.

Передаваемые в задачах межконтроллерного обмена переменные должны быть сгруппированы в одну глобальную или локальную переменную с типом структуры или массива. Последовательность и типы данных переменных в структуре или массиве должны совпадать на передающем и принимающем устройствах. Для каждой передаваемой переменной должно быть установлено соответствующее значение атрибута: «*Вход*» - для переменных на принимающем устройстве, «*Выход*» - для переменных на отправляющем устройстве, «*Вход/Выход*» - для переменных, доступных одновременно для чтения и записи.

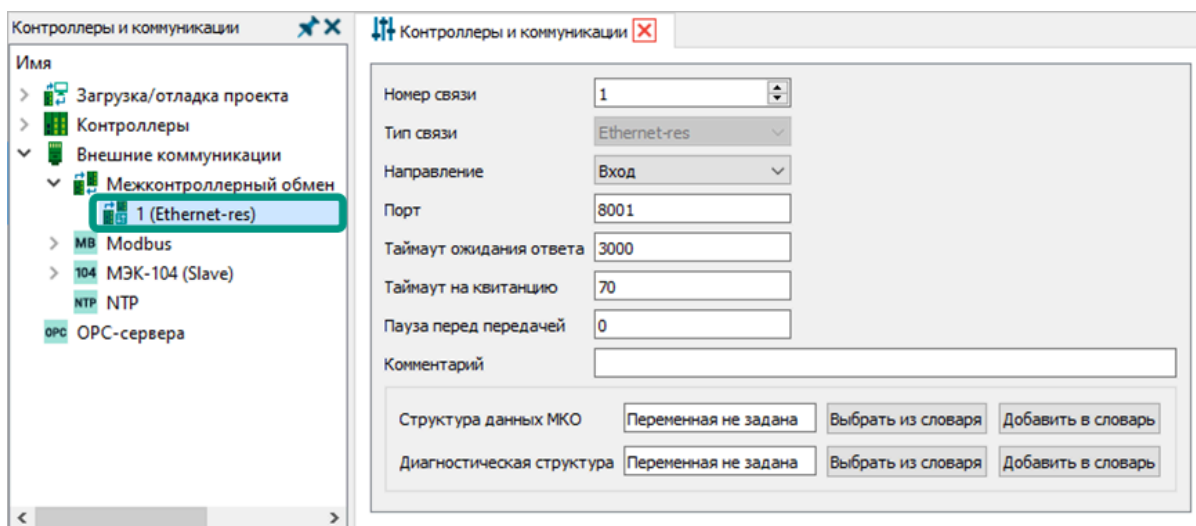
Для данного типа связи также доступна функция резервирования, для этого на отправляющем устройстве достаточно указать IP-адреса резервных устройств. Переход на резервную сеть произойдет автоматически, при исчезновении связи по основной сети. Если подключение резервной сети не планируется, то резервный IP-адрес не указывается. В противном случае контроллер будет выдавать ошибку

межконтроллерного обмена по резервной сети, так как им производится постоянный мониторинг состояния сетей по указанным адресам.

Для добавления задачи межконтроллерного обмена в дереве конфигурации нажмите правой кнопкой мыши по разделу «Межконтроллерный обмен» → из контекстного меню выберите пункт «Добавить связь Treinet» → новая задача связи будет добавлена в дерево конфигурации.



Форма настройки параметров закреплена за каждой задачей связи межконтроллерного обмена и открывается при выборе задачи в дереве конфигурации.



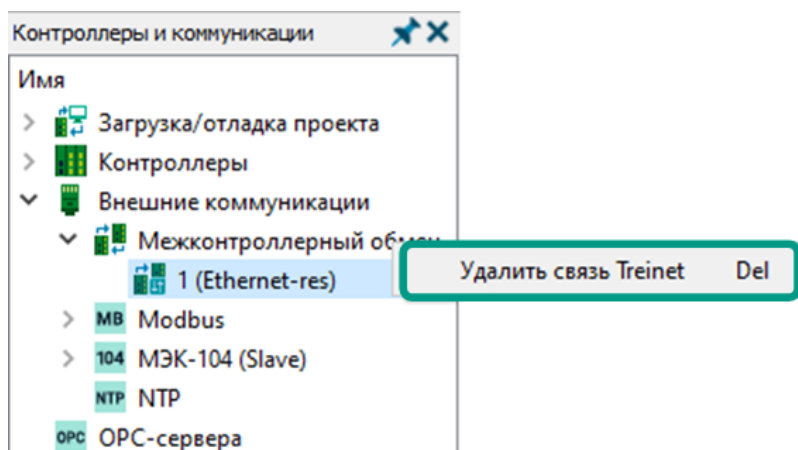
Для настройки доступны параметры:

- **номер связи** – целое число от 1 до 99, используется для нумерации задач связи в дереве конфигурации. Номер связи должен быть уникальным в пределах задач межконтроллерного обмена.

- **тип связи** – определяет интерфейс связи с удаленным контроллером, по умолчанию равен Ethernet-res (канал межмастерного обмена через сеть Ethernet с возможностью задать до 4-х удаленных устройств) и не доступен для редактирования.
- **направление передачи данных** – определяет направление потока данных: вход - для приема данных с удаленного контроллера, выход - для передачи.
- **порт** – номер порта TCP/IP, через который будет осуществляться обмен данными, должен быть уникальным в пределах задач связи. При этом номера портов в задачах связи передающего и принимающего контроллера должны совпадать.
- **адрес удаленного узла 1..4** - IP-адреса удаленных устройств в формате IPv4, доступны только при выходном направлении передачи данных, при наличии резервирования могут указываться адреса удаленных резервных контроллеров.
- **таймаут ожидания ответа** – период времени в миллисекундах, в течение которого осуществляются попытки передачи одного пакета данных. Если за указанный период не удалось отправить один пакет, в системные события выводится ошибка обмена данными, таймер перезапускается, а попытки отправки пакета продолжаются.
- **таймаут на квитанцию** – период времени в миллисекундах между отправлением одного пакета данных и получением от удаленного устройства подтверждения о его приеме. Если за указанный период ответ не поступил, пакет отправляется повторно, а таймер перезапускается. Попытки передать пакет продолжаются до истечения таймаута ожидания ответа.
- **пауза перед передачей** – период времени в миллисекундах между получением подтверждения от удаленного устройства о приеме пакета данных и отправкой нового.
- **комментарий** – поле ввода краткого описания и назначения задачи связи.
- **структура данных межконтроллерного обмена (МКО)** – выбирается переменная с типом структуры или массива, значениями которой будут обмениваться контроллеры, последовательность и типы данных сгруппированных переменных должны совпадать на передающем и принимающем устройствах. Суммарный объем данных для одной задачи межконтроллерного обмена не должен превышать 65 Кбайт. Перед добавлением переменная должна быть создана в глобальном или локальном словаре, выбор переменной осуществляется нажатием кнопки «Выбрать из словаря», создание - нажатием кнопки «Добавить в словарь». При добавлении переменная будет создана в глобальном словаре и добавлена в текущий параметр связи.

- **диагностическая структура** – выбирается переменная со структурным типом данных «*MKO_DIAG*», в которую исполнительная система Unimod PRO 2 будет записывать статистику и ошибки текущей задачи связи. Создание и выбор переменной осуществляется аналогично выбору структуры данных МКО.

Для удаления задачи связи в разделе «*Межконтроллерный обмен*» дерева конфигурации выделите задачу → нажмите по ней правой кнопкой мыши и из контекстного меню выберите пункт «*Удалить связь Treinet*» или нажмите клавишу «*Delete*» → в открывшемся окне подтвердите удаление нажатием кнопки «*Да*».



7.3.2 Задачи связи Modbus

Modbus — открытый коммуникационный протокол связи, широко применяемый в системах промышленной автоматизации и управления для организации клиент-серверного взаимодействия между промышленными устройствами. Протокол основан на архитектуре *ведущий (master) — ведомый (slave)*, где *master-устройство* может направлять запросы на получение данных, а *slave-устройства* в ответ на запрос предоставляют требуемые данные или выполняют указанные в запросе действия. Направление широковещательного запроса исполнительной системой Unimod PRO 2 не предусмотрено.

Установленная в контроллере TREI исполнительная система Unimod PRO 2 позволяет использовать два популярных варианта протокола: Modbus RTU и Modbus TCP. Modbus RTU используется для последовательной связи через интерфейсы RS-485, RS-422 или RS-232, а Modbus TCP использует сети Ethernet и стек протоколов TCP/IP для клиент-серверного взаимодействия. Для организации обмена по протоколу Modbus должны быть запущены соответствующие задачи связи.

Создание и настройка задач связи по протоколу Modbus осуществляется в среде разработки Unimod PRO 2 двумя способами: в дереве конфигурации или программно с использованием функциональных блоков.

Правила организации связи по протоколу Modbus:

- задачи связи разделены по типу используемого протокола (RTU или TCP) и режиму работы устройства в процессе обмена (Master или Slave);
- для каждой комбинации используемого протокола и режима работы созданы соответствующие разделы в дереве конфигурации и функциональные блоки в [системной библиотеке](#);
- поддерживается одновременное создание задач связи в дереве конфигурации и программно с использованием блоков;
- одна задача связи используется только для приема или передачи данных;
- номер задачи должен быть уникальным в пределах всех задач связи Modbus;
- для разных задач связи в рамках всего приложения номера портов не должны пересекаться;
- на взаимодействующих устройствах должны быть созданы задачи связи с одинаковым типом протокола, но противоположными ролями;
- в режиме *Master* контроллер направляет запросы на Slave-устройства и получает от них ответы. В режиме *Slave* контроллер только отвечает на запросы.
- для идентификации переменных по протоколу Modbus используются Modbus-адреса, которые указываются в атрибуте переменных;
- суммарный объем передаваемых/принимаемых данных в рамках одной задачи не должен превышать 65 Кбайт.

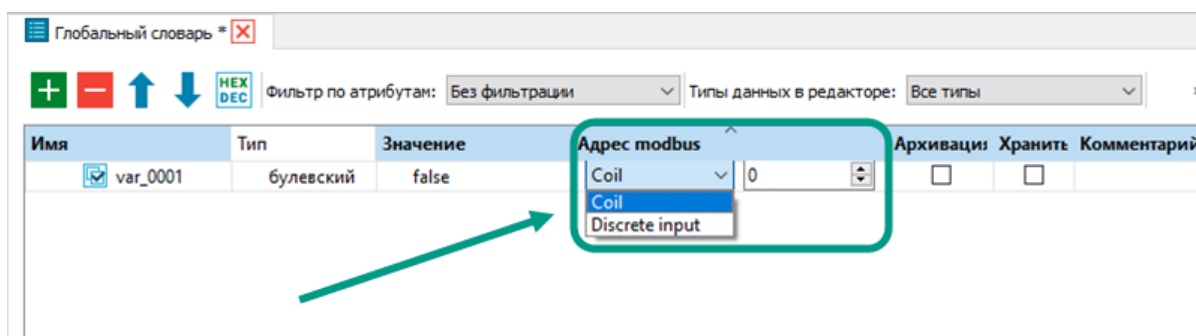
Подробное описание протокола Modbus в документе [«Unimod Pro. Протокол Modbus»](#).

7.3.2.1 Назначение modbus-адресов

При передаче данных по протоколу Modbus переменные идентифицируются по modbus-адресам. В словарях среды разработки Unimod PRO 2 modbus-адреса указываются в качестве атрибута переменной. Назначать адреса можно как одиночным, так и составным переменным, к которым относятся структуры и массивы. Для составных переменных указывается начальный modbus-адрес, а всем вложенным переменным modbus-адреса назначаются по порядку в зависимости от их типа данных.

Для логических переменных адреса задаются последовательно, для целых и вещественных – через один. Если предполагается чтение или запись 32-х разрядных чисел, то Modbus-адреса переменных должны идти через один (два регистра на одну переменную), а для 64-х разрядных чисел - через 4 (четыре регистра на одну переменную). Рекомендуется для каждого типа данных выделять отдельный блок modbus-адресов.

Для назначения modbus-адреса одиночной переменной в словаре дважды щелкните левой кнопкой мыши по атрибуту «Адрес modbus» → выберите тип modbus-переменной и введите modbus-адрес в виде целого числа → нажмите клавишу «Enter».



Тип modbus-переменной:

Discrete input - входное дискретное значение;

Coil - выходное дискретное значение.

Input register - входное аналоговое значение (байтовый, целочисленный и вещественные типы данных);

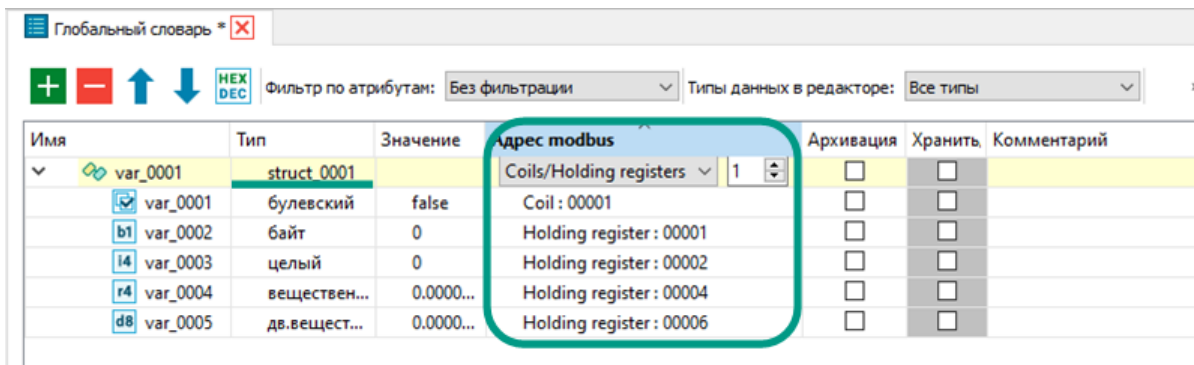
Holding register - выходное аналоговое значение;

Input/Input registers - группа входных дискретных или аналоговых значений;

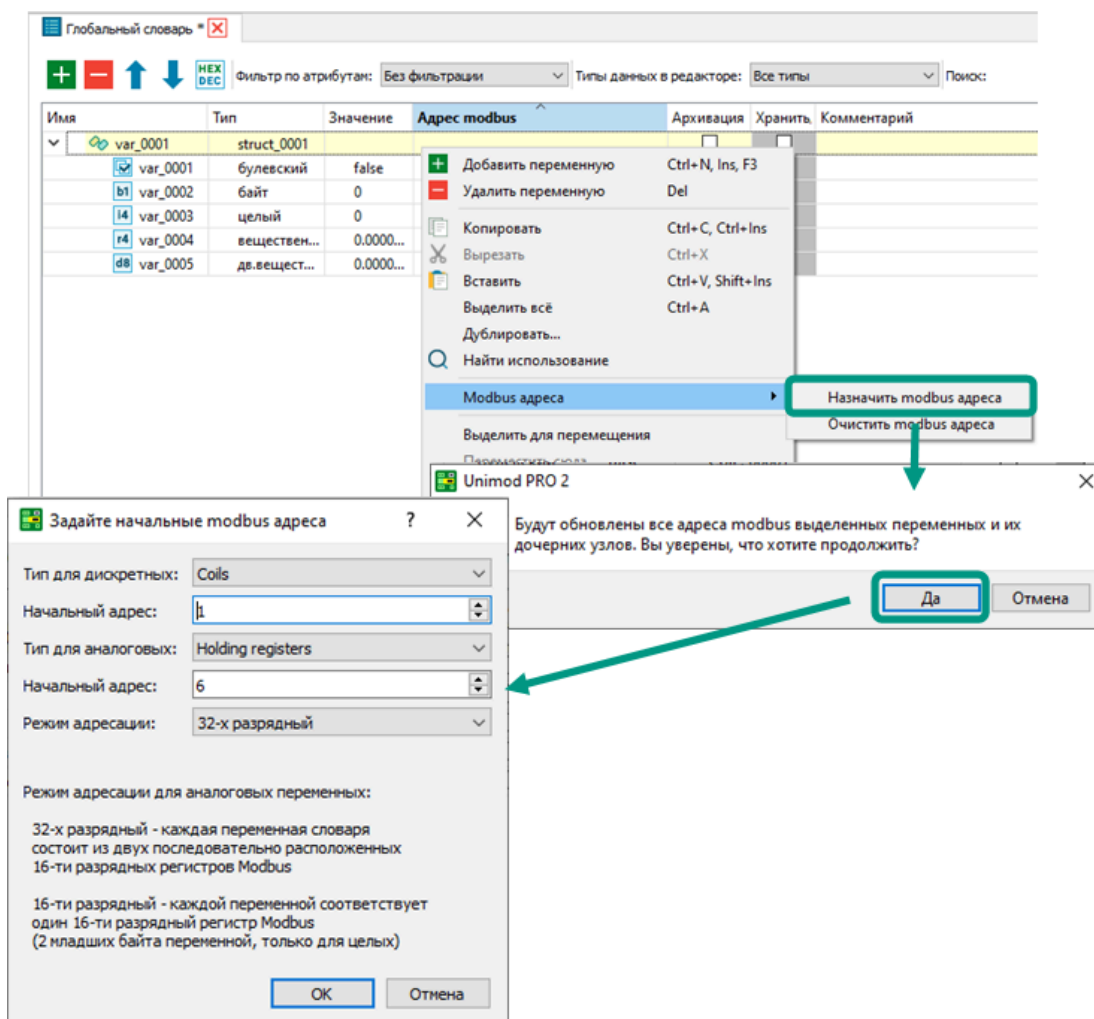
Coils/Holding registers - группа выходных дискретных или аналоговых значений.

Для назначения modbus-адреса составной переменной в словаре дважды щелкните левой кнопкой мыши по ее атрибуту «Адрес modbus» → выберите тип modbus-переменной и введите начальный modbus-адрес в виде целого

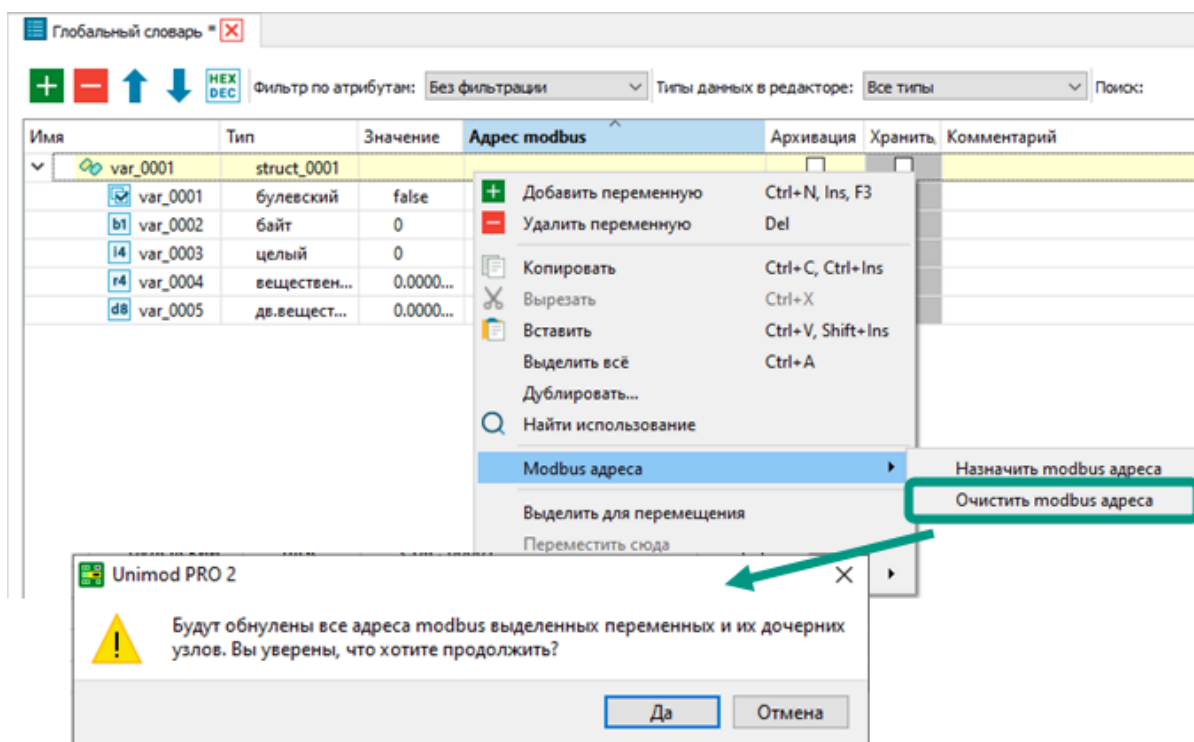
числа → нажмите клавишу «Enter». После чего для всех вложенных переменных автоматически, в зависимости от типа данных, будут добавлены modbus-адреса.



Другим способом задания modbus-адресов составным переменным является открытие окна для настройки начальных значений modbus-адресов. Для этого кликните правой кнопкой мыши по структурной переменной в словаре → из контекстного меню выберите пункт «Назначить modbus адреса» → подтвердите установку адресов нажатием в диалоговом окне кнопки «Да» → в открывшемся окне выберите типы и укажите начальные адреса для дискретных и аналоговых переменных → выберите режим адресации → нажмите кнопку «Ок». После чего modbus-адреса будут назначены всем вложенным переменным.



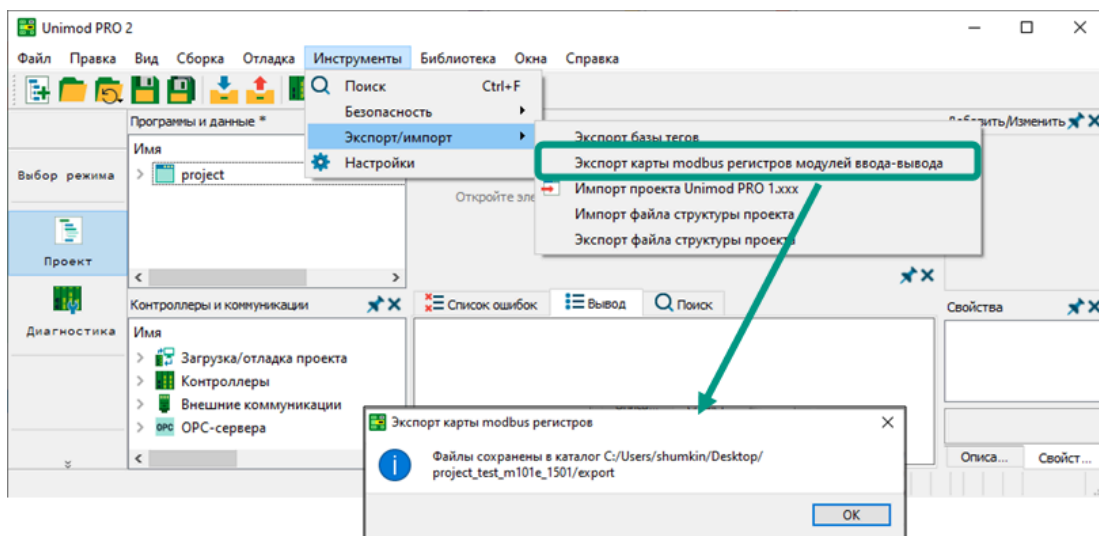
Для удаления всех modbus-адресов составной переменной кликните по ней правой кнопкой мыши → из контекстного меню выберите пункт «Очистить modbus адреса» → в открывшемся окне подтвердите удаление адресов нажатием кнопки «Да».



7.3.2.2 Экспорт modbus-адресов модулей ввода-вывода

Модули ввода-вывода представлены в среде разработки Unimod PRO 2 в виде глобальных переменных со структурным типом данных, соответствующим модели и версии прошивки выбранного модуля. Поэтому переменным модульных структур также могут быть заданы modbus-адреса.

После назначения в проекте всем модулям ввода-вывода modbus-адресов, имеется возможность сохранить их в файлах формата «*.csv». Для этого выберите пункт меню «Инструменты → Экспорт/импорт → Экспорт карты modbus регистров модулей ввода-вывода».



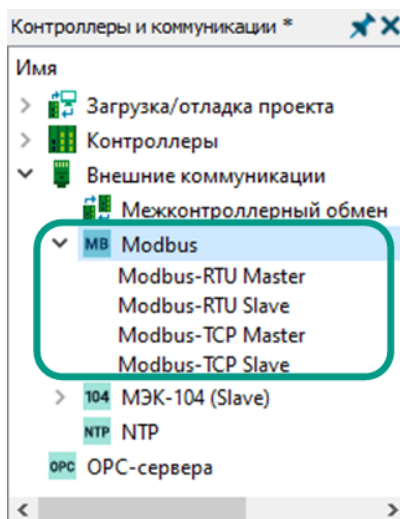
В результате в каталоге проекта будет создан вложенный каталог «*export*», в котором для каждого модуля ввода-вывода будет создан отдельный файл с именем «*moduleN_modbus.csv*», где *N* – адрес модуля ввода-вывода. В каждом файле будут содержаться переменные, входящие в структуру модуля ввода-вывода, с указанием имени, типа данных в приложении, типа modbus-регистра (числовое обозначение типов данных) и комментария с кратким описанием назначения переменной.

На следующем изображении в качестве примера приведен фрагмент сохраненных сведений о modbus-адресах модуля ввода-вывода.

Имя	Тип	Тип регистра modbus	Адрес modbus (dec)	Комментарий
var_0001.control.not_valid	Булев.	1	100	Признак ошибки/блокировки обмена
var_0001.control.err_cfg	Целый	3	100	Логическая ошибка обмена
var_0001.control.err_line1	Целый	3	102	Транспортная ошибка обмена по линии 1
var_0001.control.err_line2	Целый	3	104	Транспортная ошибка обмена по линии 2
var_0001.control.block_poll	Булев.	1	101	Блокировка обмена с модулем
var_0001.control.command	Целый	3	106	Резерв
var_0001.control.read_statistic	Булев.	1	102	Чтение блока statistic
var_0001.control.err_rw_statistic	Целый	3	108	Код завершения последнего асинхронного чтения/записи блока statistic
var_0001.control.read_param	Булев.	1	103	Чтение блока param
var_0001.control.write_param	Булев.	1	104	Запись блока param
var_0001.control.err_rw_param	Целый	3	110	Код завершения последнего асинхронного чтения/записи блока param
var_0001.statistic.Work_Time	Целый	3	112	Время наработки, сек
var_0001.statistic.Stbus_cnt_Err	Целый	3	114	Счетчик ошибок STBUS
var_0001.statistic.Reset_code	Целый	3	116	Код сброса
var_0001.statistic.Mod_temp	Целый	3	118	Температура модуля
var_0001.statistic.Stbus_err	Булев.	1	105	Ошибки связи с мастером

7.3.2.3 Настройка Modbus через дерево конфигурации

Создание и настройка параметров задач связи по протоколу Modbus осуществляется в разделе «Внешние коммуникации → Modbus» дерева конфигурации. Для каждого варианта протокола Modbus (RTU или TCP), а также режима работы устройства в процессе обмена (Master или Slave) предусмотрены соответствующие подразделы.

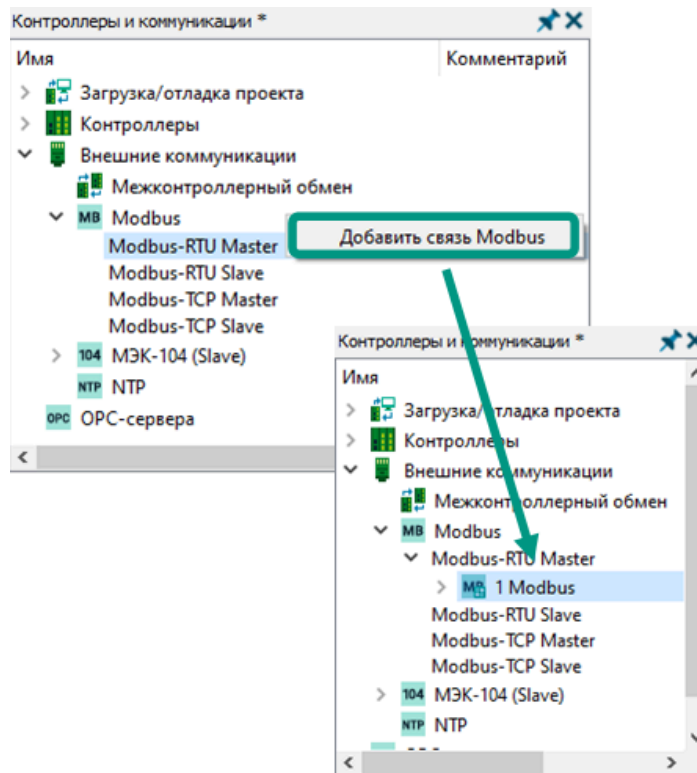


Процесс создания задачи Modbus-связи в дереве конфигурации:

1. выбрать требуемый протокол передачи данных и режим работы устройства;
2. создать в соответствующем разделе дерева конфигурации задачу связи;
3. настроить общие и специфические параметры задач, отвечающие за организацию связи по выбранному протоколу и режиму работы;
4. настроить параметры Modbus-запросов: требуемое действие (функция чтения или записи), порядок обмена данными, массив с данными для обмена и структура для мониторинга процесса связи;
5. осуществить вышеуказанный алгоритм для взаимодействующего устройства, при этом выбранные протоколы и настройки параметров Modbus-запросов должны совпадать, а режимы работы быть противоположными.

7.3.2.3.1 Настройка задач связи Modbus

Для добавления задачи связи Modbus в дереве конфигурации правой кнопкой мыши нажмите на требуемый подраздел Modbus → из контекстного меню выберите пункт «Добавить связь Modbus» → задача связи будет добавлена в соответствующий раздел дерева конфигурации. Процесс добавления задач связи Modbus-RTU Master/Slave и Modbus-TCP Master/Slave построен аналогичным образом.



Для настройки параметров связи Modbus откройте форму, выбрав задачу связи в дереве конфигурации.

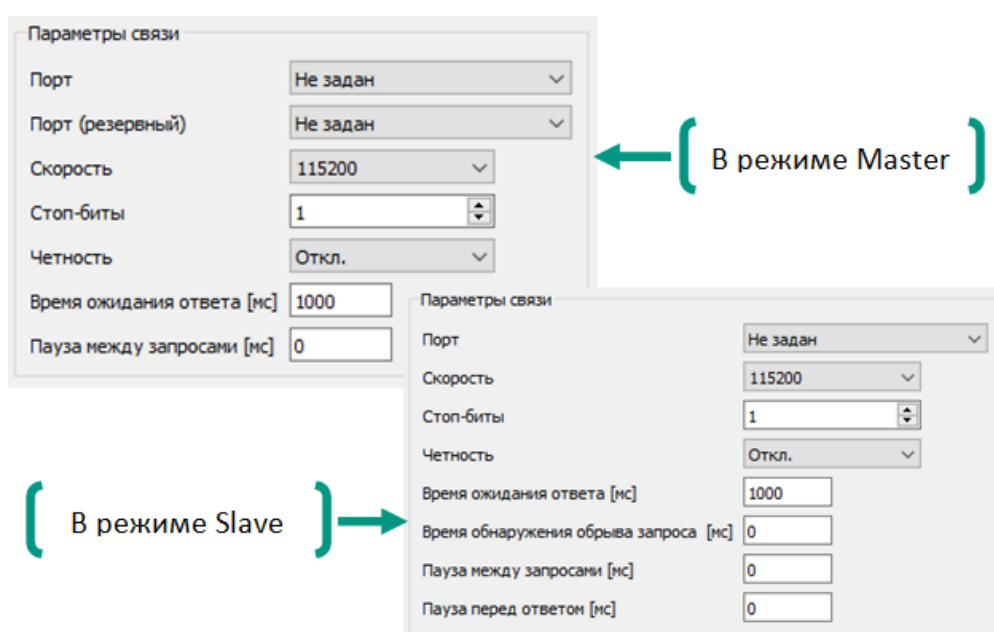
Общие параметры задач связи Modbus:

Номер связи	<input type="text" value="1"/>
Название связи	<input type="text" value="Modbus"/>
Тип связи	<input type="text" value="MODBUS-RTU"/>
Режим	<input type="text" value="Master"/>
Комментарий	<input type="text"/>

- **номер связи** - используется для нумерации задач связи и должен быть уникальным в пределах всех задач Modbus. Доступны значения от 1 до 99.

- **название связи** - имя связи, отображается в дереве конфигурации, должно содержать только латинские буквы, цифры и символы подчеркивания, начинаться с буквы и не превышать 128 символов.
- **тип связи** - определяет интерфейс связи с удаленным устройством. Modbus-RTU - для передачи данных в двоичном формате, Modbus-RTU (ASCII) - для передачи закодированных данных. Для задач связи Modbus-TCP автоматически выставляется значение Modbus-TCP.
- **режим** - Master или Slave, выставляется автоматически в зависимости от выбранного раздела, определяет режим работы устройства: на передачу или на прием.
- **комментарий** - поле ввода краткого описания и назначения задачи связи.

Параметры связи Modbus-RTU в режимах Master и Slave:



- **порт** - определяет номер используемого физического COM-порта. Одна задача связи использует один физический COM-порт, поэтому для двусторонней передачи данных используются минимум два физических COM-порта: один порт на вход, другой на выход. Количество доступных соединений по протоколу Modbus-RTU ограничивается количеством COM-портов контроллера.
- **резервный порт** - определяет номер резервного COM-порта;
- **скорость** - задает скорость передачи данных в бит/с, поддерживаются скорости: 1200, 2400, 4800, 9600, 14400, 19200, 38400, 57600 и 115200;
- **стоп-биты** - задается количество стоповых битов: 1 или 2, при отключенном режиме проверки четности всегда выставляется 2 стоповых бита.

- **четность** - задается режим контроля четности при передаче данных:
EVEN - бит четности устанавливается в единицу при четном количестве единиц в битах данных,
ODD - бит устанавливается в единицу при нечетном количестве единиц,
MARK - бит четности всегда устанавливается равным единице,
SPACE - бит четности всегда устанавливается равным нулю.

Режимы *MARK* и *SPACE* не влияют на возможность контроля ошибок. При отключенном параметре контроль четности не производится, при этом всегда выставляются два стоповых бита.

- **время ожидания ответа** - период времени, в течение которого ожидается ответ от удаленного Slave-устройства. Используется только на Master-устройстве. При отсутствии ответа в указанный период, запрос отправляется повторно.
- **пауза между запросами** - период времени, по истечении которого будет отправлен очередной запрос на Slave-устройство. Используется только на Master-устройстве. При нулевом значении, следующий запрос будет отправлен сразу после получения ответа.
- **время обнаружения обрыва запроса** - период времени между поступающими байтами одного запроса, если в указанный период очередной байт запроса не поступил, диагностируется обрыв запроса. При нулевом периоде диагностика не проводится.
- **пауза перед ответом** - период времени между моментом поступления запроса и отправкой ответа. При нулевом значении, ответ будет отправлен незамедлительно после его формирования. Используется только на Slave-устройстве.

Время в параметрах задается в миллисекундах. Скорость передачи данных, количество стоповых битов и режим контроля четности должны совпадать на передающем и принимающем устройствах.

Параметры связи Modbus-TCP Master:

Параметры связи	
IP-адрес удаленного узла	<input type="text"/>
Номер удаленного TCP порта	<input type="text" value="502"/>
IP-адрес удаленного узла (резервный)	<input type="text"/>
Номер удаленного TCP порта (резервный)	<input type="text" value="0"/>

- **IP-адрес удаленного узла** - используется при направлении запроса для идентификации Slave-устройства в сети;

- **номер удаленного TCP порта** - используется для определения входящего порта на удаленном Slave-устройстве. Должен совпадать с номером локального TCP порта в соответствующей задаче связи Slave-устройства.
- **IP-адрес удаленного узла (резервный)** - используется для идентификации резервного Slave-устройства, при отсутствии связи с основным;
- **номер удаленного TCP порта (резервный)** - используется для определения входящего порта на резервном Slave-устройстве. Должен совпадать с номером локального TCP порта в соответствующей задаче связи резервного Slave-устройства.

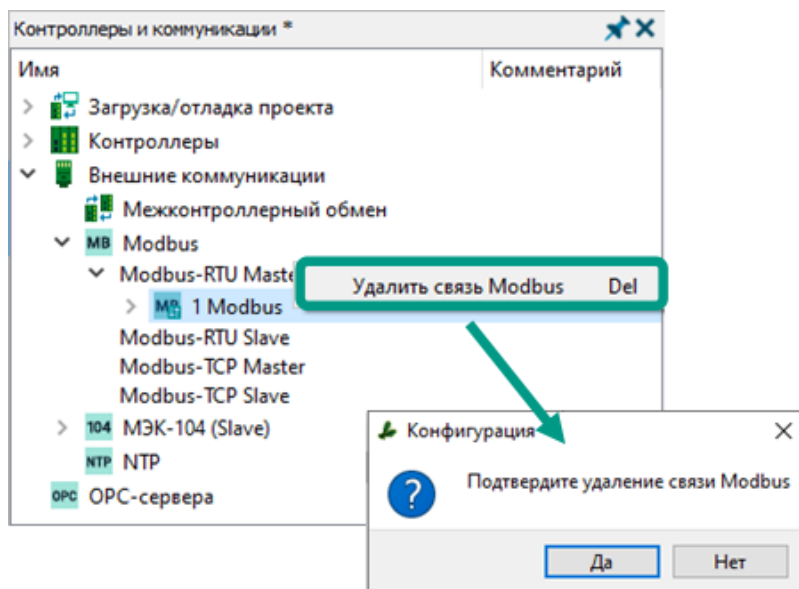
IP-адреса задаются в формате IP4. Номером порта является число в диапазоне от 502 до 65535, уникальное в пределах всех задач связи приложения.

Параметры связи Modbus-TCP Slave:

Параметры связи	
Номер локального TCP порта	502
Пауза перед ответом [мс]	0
Максимальное число соединений [1..255]	10
Время удержания соединения [мс]	15000

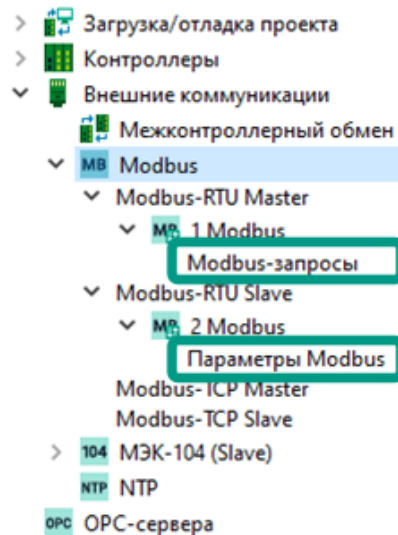
- **номер локального TCP порта** - номер TCP порта, через который Slave-устройство будет ожидать входящие запросы от Master-устройства. Должен совпадать с номером удаленного TCP порта в соответствующей задаче связи Master-устройства.
- **пауза перед ответом** - период времени в миллисекундах между моментом поступления запроса и отправкой ответа. При нулевом значении, ответ будет отправлен незамедлительно после его формирования.
- **максимальное число соединений** - целое число в диапазоне от 1 до 255, задает максимальное количество одновременных подключений через TCP-порт. Если число соединений будет равно указанному в данном параметре значению, все попытки подключения со стороны других абонентов будут игнорироваться до момента отключения абонента или по истечении времени удержания одного из соединений.
- **время удержания соединения** - период времени в миллисекундах, который используется для разрыва соединения, отсчитывается от момента последней отправки ответа. Если в указанный период от Master-устройства не поступают запросы, происходит разрыв соединения, а к TCP порту может подключиться другое устройство.

Для удаления задачи связи Modbus в дереве конфигурации кликните правой кнопкой мыши по задаче связи и из контекстного меню выберите пункт «Удалить связь Modbus» или выберите задачу связи и нажмите клавишу «Delete» → в открывшемся окне подтвердите удаление нажатием кнопки «Да». Процесс удаления задач связи Modbus-RTU Master/Slave и Modbus-TCP Master/Slave построен аналогичным образом.



7.3.2.3.2 Настройка Modbus-запросов

Для настройки параметров Modbus-запроса каждой задачи связи Modbus в дереве конфигурации создается вложенный раздел, на форме которого доступны настройки параметров Modbus: для режима Master - параметры Modbus-запроса, для режима Slave - параметры чтения Modbus-запроса.



Параметры Modbus-запросов:

Адрес устройства	Функция	Тип данных	Порядок байт	Начальный адрес на удаленном устройстве	Массив для чтения/записи	Количество переменных	Диагностическая структура
1	01 (0x01) READ COILS	Булевский	---	0	Переменная не задана	1	Переменная не задана

- **адрес устройства** - число в диапазоне от 1 до 255, определяет сетевой адрес удаленного Slave-устройства, на которое будет отправляться Modbus-запрос. Значение данного параметра должно совпадать со значением локального адреса Slave-устройства. Передача широковещательного сообщения в исполнительной системе Unimod PRO 2 не предусмотрена.
- **функция** - определяет действие, которое будет выполнено на Slave-устройстве после получения запроса. Для выбора доступны:
 - 01 (0x01) READ COILS - запрос на чтение значений нескольких дискретных выходов;
 - 02 (0x02) READ CONTACTS - запрос на чтение значений нескольких дискретных входов;

03 (0x03) READ HOLDING REGISTERS - запрос на чтение значений нескольких аналоговых выходов;

04 (0x04) READ INPUT REGISTERS - запрос на чтение значений нескольких аналоговых входов;

15 (0x0F) WRITE MULTIPLE COILS - запрос на запись значений в несколько дискретных выходов;

16 (0x10) WRITE MULTIPLE REGISTERS - запрос на запись значений нескольких аналоговых выходов.

- **тип данных** - для передачи по протоколу Modbus доступны значения следующих типов данных:

логический;

2-х байтовое целое без знака;

2-х байтовое целое со знаком;

4-х байтовое целое со знаком;

4-х байтовое вещественное;

8-и байтовое вещественное.

Логический тип данных используется для передачи дискретных данных, целочисленные и вещественные типы - для аналоговых.

- **порядок байт** - определяет порядок передачи байт данных, из которых состоит запрос:

0-1 – 2-х байтовое;

1-0 – 2-х байтовое с заменой байт;

0-1-2-3 – 4-х байтовое

1-0-3-2 – 4-х байтовое с заменой байт;

2-3-0-1 – 4-х байтовое с заменой слов;

3-2-1-0 – 4-х байтовое с заменой байт и слов;

0-1-2-3-4-5-6-7 – 8-и байтовое вещественное;

1-0-3-2-5-4-7-6 – 8-и байтовое вещественное с заменой байт;

2-3-0-1-6-7-4-5 – 8-и байтовое вещественное с заменой слов;

3-2-1-0-7-6-5-4 – 8-и байтовое вещественное с заменой байт и слов;

4-5-6-7-0-1-2-3 – 8-и байтовое вещественное с заменой двойных слов;

5-4-7-6-1-0-3-2 – 8-и байтовое вещественное с заменой двойных слов и байт;

6-7-4-5-2-3-0-1 – 8-и байтовое вещественное с заменой двойных слов и слов;

7-6-5-4-3-2-1-0 – 8-и байтовое вещественное с заменой двойных слов, слов и байт.

Используется только при обмене аналоговыми данными, при выборе логического типа данных недоступен.

- **начальный адрес на удаленном устройстве** - Modbus-адрес переменной на удаленном Slave-устройстве, с которого начнется чтение или запись. Modbus-адрес является атрибутом переменной и задается в словаре.

- **количество переменных** - число в диапазоне от 1 до 125, определяет количество переменных на Slave-устройстве, значения которых будут прочитаны или перезаписаны. Отсчет начнется с начального Modbus-адреса на удаленном устройстве, начальный адрес указывается в запросе.
- **массив для чтения/записи** - выбирается переменная с типом «Массив данных», значениями которой будут обмениваться контроллеры. При этом порядок следования переменных и их типы данных должны совпадать. Перед добавлением переменная должна быть создана в глобальном или локальном словаре. Для выбора переменной дважды щелкните мышью по ячейке таблицы.
- **диагностическая структура** - выбирается переменная со структурным типом данных «MODBUS_DIAG», в которую исполнительная система Unimod PRO 2 будет записывать статистику и ошибки текущей задачи связи. Перед добавлением переменная должна быть создана в глобальном или локальном словаре. Для выбора переменной дважды щелкните мышью по ячейке таблицы.

Параметры чтения Modbus-запроса на Slave-устройстве:

+		-	
Локальный адрес устройства		Порядок байт	
1	1	8-и байтовое 1-0-3-2-5-4-7-6 4-х байтовое 1-0-3-2 2-х байтовое 1-0	

- **локальный адрес устройства** - число в диапазоне от 1 до 255, определяет сетевой адрес Slave-устройства по протоколу Modbus, на которое поступит запрос. Значение параметра должно совпадать со значением адреса на Master-устройстве.
- **порядок байт** - определяет порядок чтения байт, из которых состоит запрос, на Slave-устройстве:
 - 0-1-2-3-4-5-6-7* – 8-и байтовое вещественное, 4-х и 2-х байтовое;
 - 1-0-3-2-5-4-7-6* – 8-и байтовое вещественное, 4-х и 2-х байтовое с заменой байт;
 - 2-3-0-1-6-7-4-5* – 8-и байтовое вещественное, 4-х и 2-х байтовое с заменой слов;
 - 3-2-1-0-7-6-5-4* – 8-и байтовое вещественное, 4-х и 2-х байтовое с заменой байт и слов;

4-5-6-7-0-1-2-3 – 8-и байтовое вещественное, 4-х и 2-х байтовое с заменой двойных слов;

5-4-7-6-1-0-3-2 – 8-и байтовое вещественное, 4-х и 2-х байтовое с заменой двойных слов и байт;

6-7-4-5-2-3-0-1 – 8-и байтовое вещественное, 4-х и 2-х байтовое с заменой двойных слов и слов;

7-6-5-4-3-2-1-0 – 8-и байтовое вещественное, 4-х байтовое и 2-х байтовое с заменой двойных слов, слов и байт.

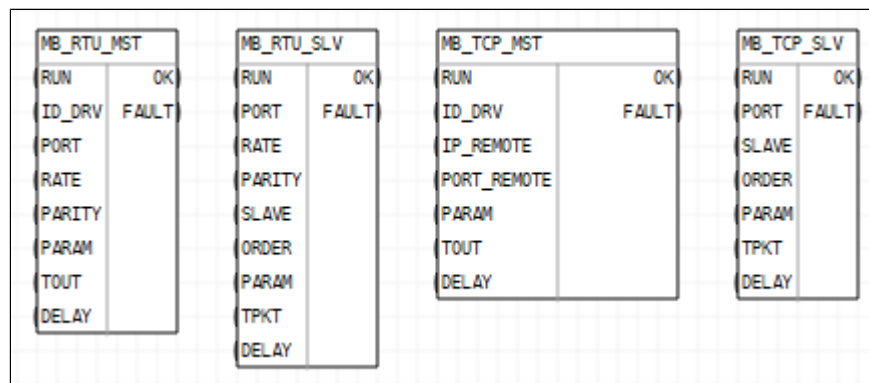
Для чтения 4-х или 2- байтового запроса выберите один из вышеуказанных вариантов с соответствующим порядком байт. Доступный для чтения порядок должен совпадать с порядком байт, указанным в параметрах Modbus-запроса на Master-устройстве.

7.3.2.4 Настройка Modbus через функциональные блоки

Для программного создания и настройки задач связи по протоколу Modbus используются функциональные блоки:

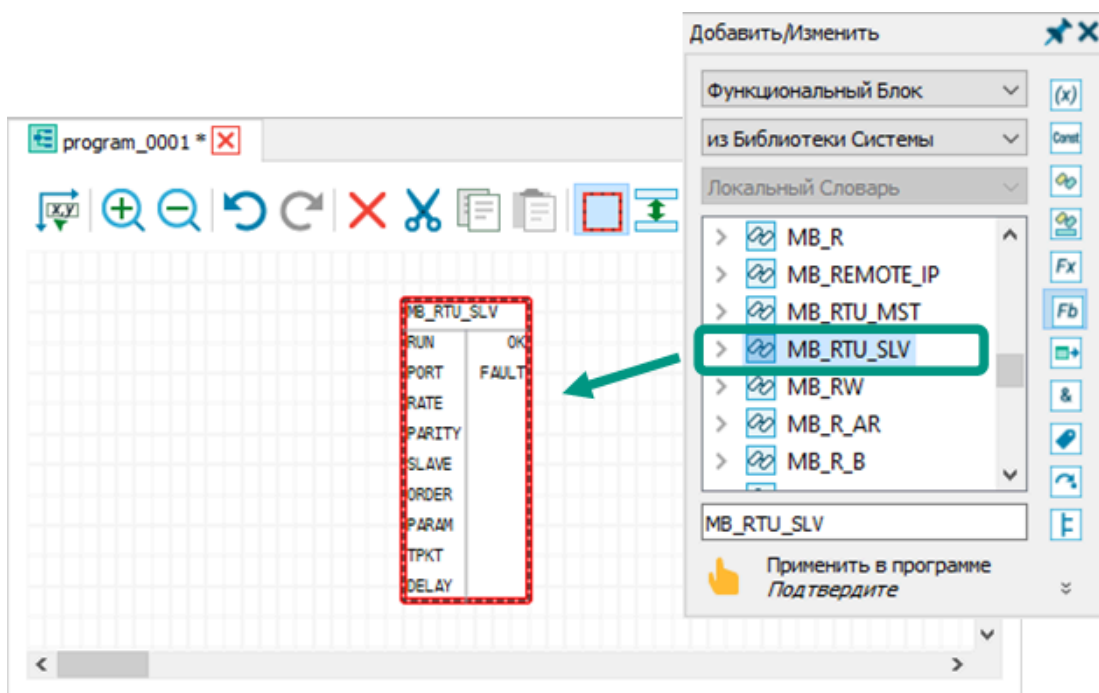
- «**MB_RTU_MST**» - задача связи по протоколу Modbus-RTU в режиме Master;
- «**MB_RTU_SLV**» - задача связи по протоколу Modbus-RTU в режиме Slave;
- «**MB_TCP_MST**» - задача связи по протоколу Modbus-TCP в режиме Master;
- «**MB_TCP_SLV**» - задача связи по протоколу Modbus-TCP в режиме Slave.

Указанные функциональные блоки содержатся в системной библиотеке, а их подробное описание приведено в документе [Руководство по программированию](#).

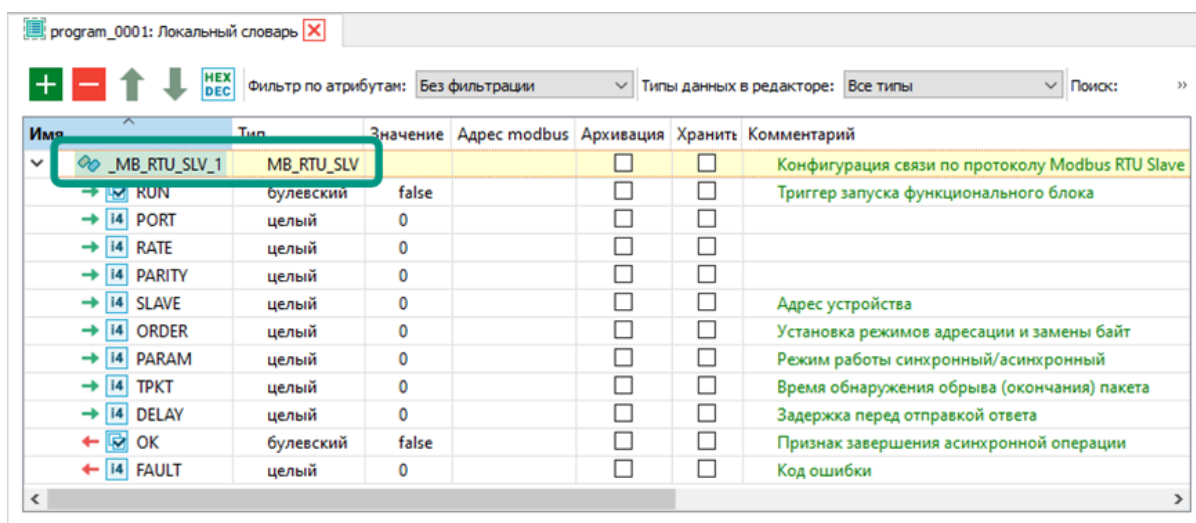


Настройка блоков для работы в режиме Slave-устройства.

Для добавления задачи связи Modbus в режиме Slave-устройства откройте редактор FBD/LD → на панели «Добавить/Изменить» в системной библиотеке найдите блок «**MB_RTU_SLV**» или «**MB_TCP_SLV**» → добавьте блок в редактор.



При добавлении блока в редактор соответствующая ему переменная структурного типа будет автоматически добавлена в словарь. Для настройки параметров задачи связи используются входные параметры добавленной структуры. Подробное описание и назначение входных и выходных параметров блоков приведено в руководстве по программированию.



Имя	Тип	Значение	Адрес modbus	Архивация	Хранить	Комментарий
MB_RTU_SLV_1	MB_RTU_SLV			<input type="checkbox"/>	<input type="checkbox"/>	Конфигурация связи по протоколу Modbus RTU Slave
RUN	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Триггер запуска функционального блока
PORT	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	
RATE	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	
PARITY	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	
SLAVE	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Адрес устройства
ORDER	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Установка режимов адресации и замены байт
PARAM	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Режим работы синхронный/асинхронный
TPKT	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Время обнаружения обрыва (окончания) пакета
DELAY	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Задержка перед отправкой ответа
OK	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Признак завершения асинхронной операции
FAULT	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Код ошибки

На этом настройка задачи связи на Slave-устройстве завершена. После запуска приложения контроллер по указанному в параметрах блока COM-порту будет отслеживать поступление от удаленного Master-устройства запроса на чтение или запись данных. В одном редакторе можно создавать произвольное количество блоков. Процесс добавления и настройки задач связи Modbus-RTU Slave и Modbus-TCP Slave построен аналогичным образом, отличаются только возможности использования портов.

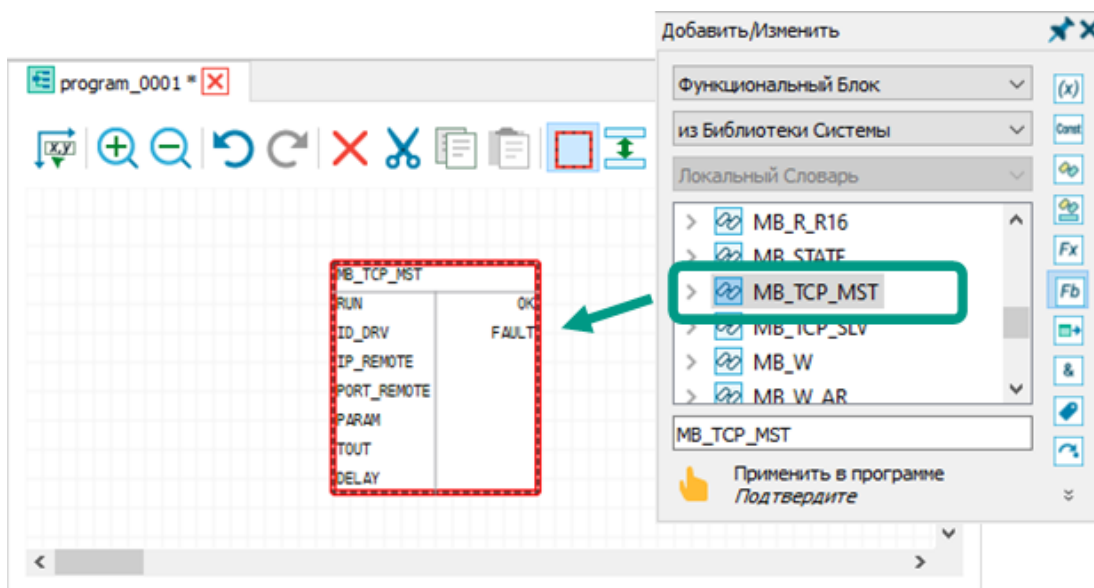
Для каждой задачи связи Modbus-RTU Slave должен быть добавлен и настроен отдельный функциональный блок «*MB_RTU_SLV*» с указанием отдельного порта, через который будет осуществляться обмен данными, номера портов не могут повторяться.

Для каждой задачи связи Modbus-TCP Slave также рекомендуется создавать отдельные блоки «*MB_TCP_SLV*» с указанием неповторяющихся портов, но возможно и подключение до 255 подключений Master-устройств к одному физическому порту. Если число текущих соединений к TCP-порту равно максимальному, то все попытки подключения со стороны других Master-устройств будут игнорироваться. Но если от удаленного абонента в течение времени удержания не приходят запросы, то соединение становится доступным для подключения другого Master-устройства.

Настройка блоков для работы в режиме Master-устройства.

Алгоритм добавления и настройки задачи связи Modbus в режиме Master-устройства аналогичен алгоритму для Slave-устройства: откройте редактор FBD/LD и добавьте в него блок «*MB_RTU_MST*» или «*MB_TCP_MST*» → настройте

входные параметры блока с использованием переменной структурного типа, созданной в словаре после добавления блока в редактор. Подробное описание и назначение входных и выходных параметров блоков приведено в руководстве по программированию.



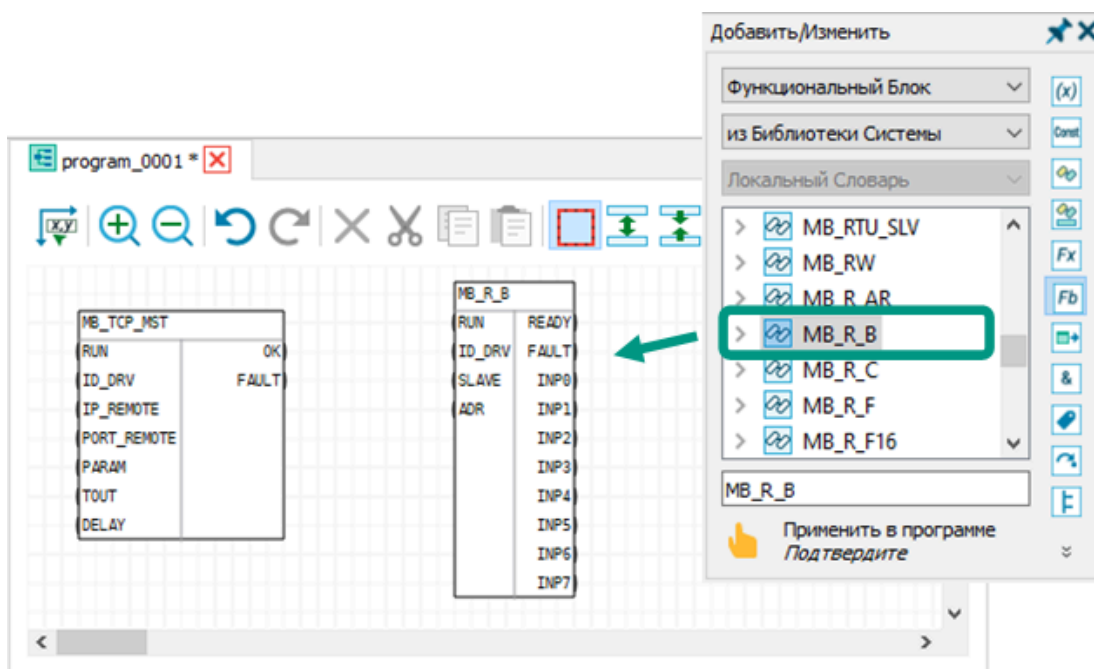
В режиме Master для чтения и записи параметров блоков «*MB_RTU_MST*» или «*MB_TCP_MST*» дополнительно требуется добавить из системной библиотеки и настроить специализированные функциональные блоки для чтения или записи. Название таких блоков начинается с префикса «*MB_**», буква R в названии блока маркирует блоки чтения, буква W – блоки записи.

Краткое описание блоков чтения и записи из системной библиотеки:

Название	Описание функционального блока
MB_R_V	Чтение состояния 8-ми дискретных входов
MB_R_C	Чтение состояния 8-ми бинарных ячеек памяти
MB_R_F	Чтение пары регистров в формате с плавающей точкой
MB_R_H	Чтение целочисленного регистра
MB_R_I	Чтение целочисленного входного регистра
MB_R_F16	Чтение 16 пар регистров в формате с плавающей точкой
MB_R_R16	Чтение 16 целочисленных регистров
MB_R	Чтение группы данных по протоколу Modbus
MB_RW	Чтение/запись регистров
MB_W_C	Запись в бинарную ячейку памяти

MB_W_F	Запись пары регистров в форме с плавающей точкой
MB_W_H	Запись целочисленного регистра
MB_W	Запись группы данных по протоколу Modbus

Для добавления блока чтения или записи откройте редактор FBD/LD → на панели «Добавить/Изменить» в системной библиотеке найдите требуемый блок → добавьте блок в редактор.



При добавлении блока чтения или записи в редактор соответствующая ему переменная структурного типа будет автоматически добавлена в словарь. Для настройки параметров задачи связи используются входные параметры добавленной структуры. Подробное описание и назначение входных и выходных параметров блоков приведено в [руководстве по программированию](#).

program_0001: Локальный словарь

HEX DEC Фильтр по атрибутам: Без фильтрации Типы данных в редакторе: Все типы Поиск: »

Имя	Тип	Значение	Адрес modbus	Архивация	Хранить	Комментарий
> MB_TCP_MST...	MB_TCP ...			<input type="checkbox"/>	<input type="checkbox"/>	Конфигурация связи по протоколу Modbus TCP Master
▼ MB_R_B_1	MB_R_B			<input type="checkbox"/>	<input type="checkbox"/>	Чтение состояния 8-ми дискретных входов
→ [✓] RUN	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Режим работы: TRUE - выполнить, FALSE не выполнять
→ [14] ID_DRV	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Идентификатор устройства Modbus
→ [14] SLAVE	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Адрес подчиненного устройства
→ [14] ADR	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Начальный адрес дискретного входа
← [✓] READY	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Операция завершена – TRUE
← [14] FAULT	целый	0		<input type="checkbox"/>	<input type="checkbox"/>	Код завершения операции
← [✓] INP0	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+0
← [✓] INP1	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+1
← [✓] INP2	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+2
← [✓] INP3	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+3
← [✓] INP4	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+4
← [✓] INP5	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+5
← [✓] INP6	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+6
← [✓] INP7	булевский	false		<input type="checkbox"/>	<input type="checkbox"/>	Состояние входа с адресом Addr+7

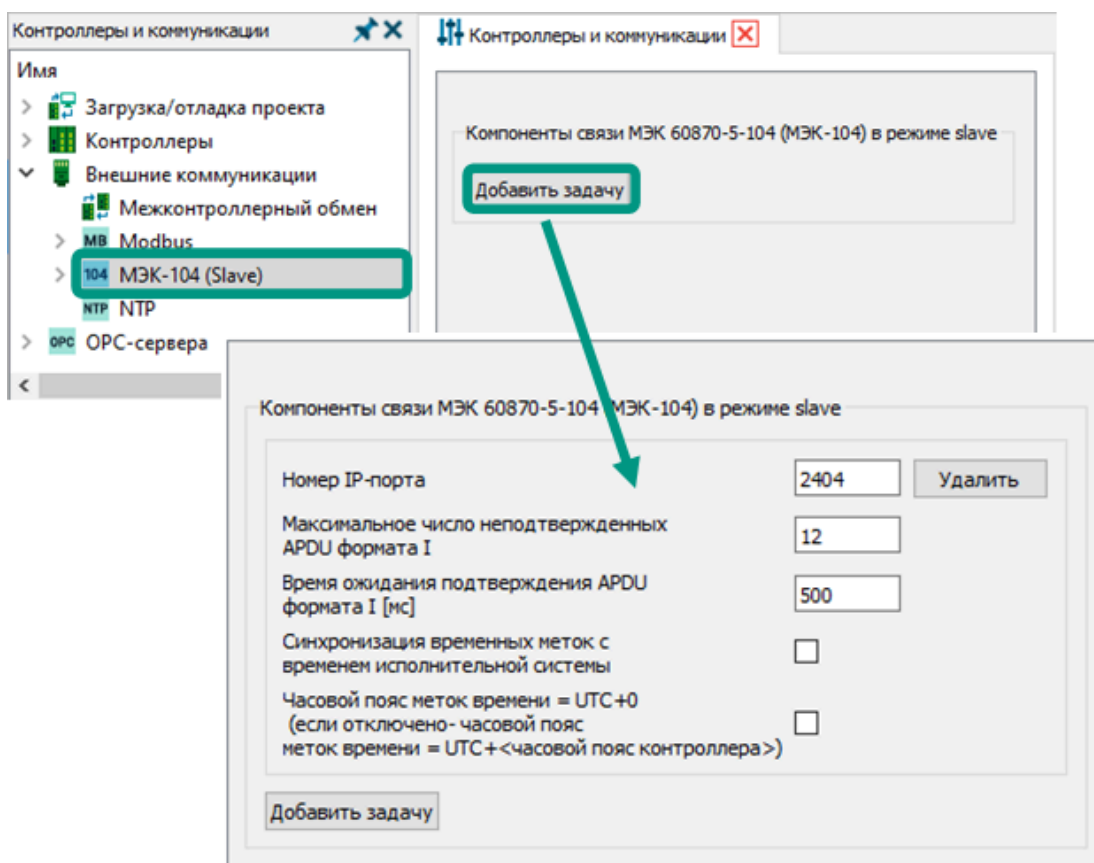
Для привязки блоков чтения и записи к блокам «MB_RTU_MST» или «MB_TCP_MST» используется входной параметр «ID_DRV». Значения данного параметра должны совпадать на блоке задачи связи Modbus и соответствующем ему блоке чтения или записи. Адрес конкретного устройства на шине, с которого нужно получить данные, указывается во входном параметре SLAVE блока чтения или записи Modbus и должен совпадать с адресом, указанным в функциональном блоке на Slave-устройстве.

7.3.3 Задачи связи МЭК-104

Протокол по стандарту ГОСТ Р МЭК 60870-5-104 (далее протокол МЭК-104) предназначен для обмена данными между устройствами сбора данных с технологического оборудования (Slave) и системами оперативно-диспетчерского управления (Master). По протоколу МЭК-104 устройства обмениваются данными по принципу клиент-серверного взаимодействия на базе стека TCP/IP.

Контроллеры TREI поддерживают протокол МЭК-104 только в режиме Slave (TCP-клиент). Для работы протокола в исполнительной системе контроллера должна быть сконфигурирована и запущена задача связи МЭК-104, которая при поступлении запроса на определенный в ней IP-порт, отправляет ответ с запрошенными значениями переменных. Подробное описание приведено в документе [«Unimod Pro. Исполнительная система»](#).

Для добавления задачи связи МЭК-104 выберите раздел «МЭК-104 (Slave)» в дереве конфигурации → на форме настройки параметров нажмите кнопку «Добавить задачу». Удаление задачи связи осуществляется нажатием кнопки «Удалить».



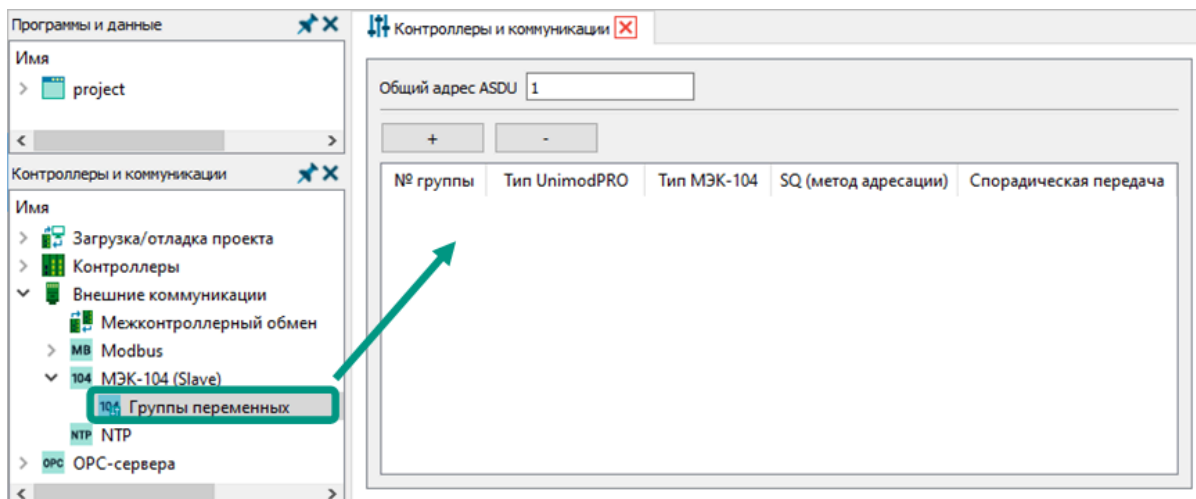
Для настройки задачи связи МЭК-104 доступны параметры:

- **номер IP-порта** - уникальный номер порта среди всех задач связи, на который внешние системы будут отправлять запрос по протоколу МЭК-104.
- **максимальное число неподтвержденных APDU формата I** - количество неподтвержденных отправленных пакетов данных формата APDU, после которого контроллер прекращает отправлять следующие пакеты до подтверждения приема ранее отправленных.
- **время ожидания подтверждения APDU формата I** - период времени в миллисекундах между отправкой пакета данных в формате APDU и получением подтверждения о его приеме. При превышении указанного значения неподтвержденный пакет отправляется повторно.
- **синхронизация временных меток со временем исполнительной системы** - при включенном параметре (помечен галочкой) для каждого значения переменной, передаваемой по протоколу МЭК-104, сохраняется время его изменения, за основу используется время исполнительной системы.
- **часовой пояс меток времени** - при включенном параметре (помечен галочкой) для меток времени применяется нулевой часовой пояс (UTC+0). При отключенном параметре применяется часовой пояс местонахождения контроллера (UTC+часовой пояс контроллера).

Для получения доступа к значениям глобальных и локальных переменных технологического приложения по протоколу МЭК-104, переменные предварительно должны быть добавлены в группы. В одной группе могут быть сгруппированы переменные с одним общим типом данных. Для каждой переменной должны быть указаны имя переменной и ее адрес по протоколу МЭК-104, который будет использоваться для обращения к значениям переменной из SCADA-систем. Подробнее в разделах [Управление группами переменных в МЭК-104](#) и [Управление переменными в МЭК-104](#) настоящего руководства.

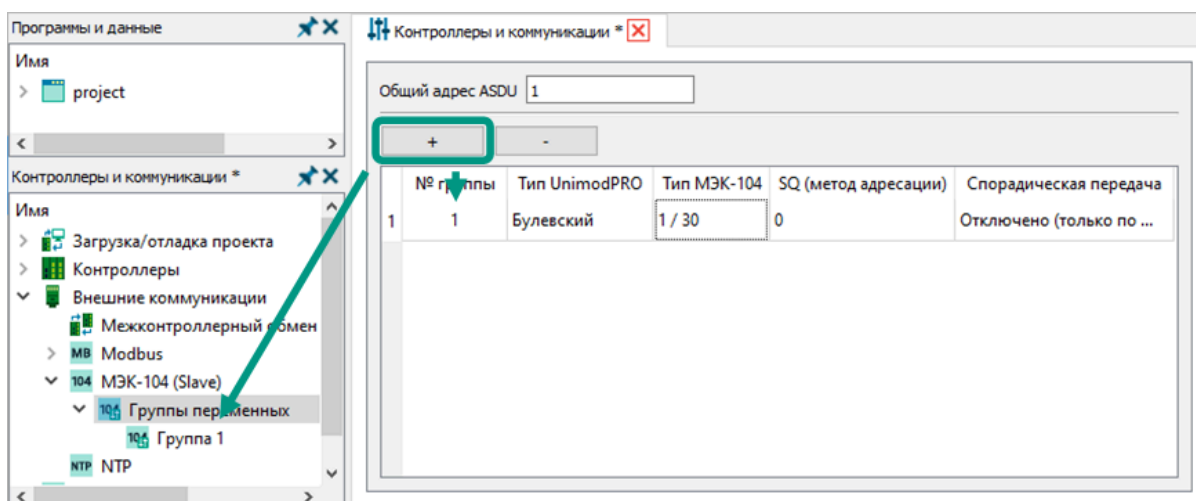
7.3.3.1 Управление группами переменных в МЭК-104

Для настройки групп переменных выберите раздел «Группы переменных» дерева конфигурации. После выбора откроется форма настройки параметров, где для редактирования доступны общий адрес блока данных ASDU и таблица редактирования групп переменных.



Общий адрес ASDU является адресом мастер-модуля по протоколу МЭК-104 и указывается в запросах из внешних систем в качестве адреса Slave-устройства.

Добавление групп осуществляется нажатием кнопки , удаление - нажатием кнопки . Добавленные группы отображаются на форме настройки параметров и в дереве конфигурации.



Для настройки группы переменных доступны параметры:

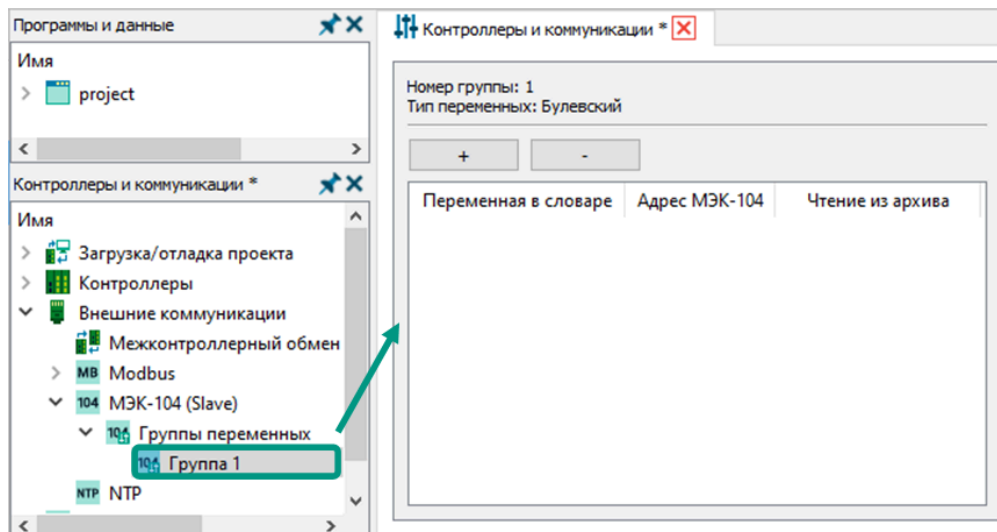
- **номер группы** - уникальное целочисленное значение в диапазоне от 1 до 255, идентифицирует группу по протоколу МЭК-104;
- **тип сгруппированных переменных** - для обмена по протоколу МЭК-104 поддерживаются четыре типа данных: логический, целочисленный 16-битный, целочисленный 32-битный и вещественный;
- **тип МЭК-104** - обозначение типа данных в протоколе МЭК-104, параметр устанавливается автоматически в зависимости от выбранного типа данных переменной:

Тип данных Unimod PRO 2	Тип данных МЭК-104
логический	1 / 30
целочисленный 32-битный	7 / 33 (чтение/запись 4 байт)
целочисленный 16-битный	11 / 35 (чтение/запись младших 2 байт)
вещественный	13 / 36

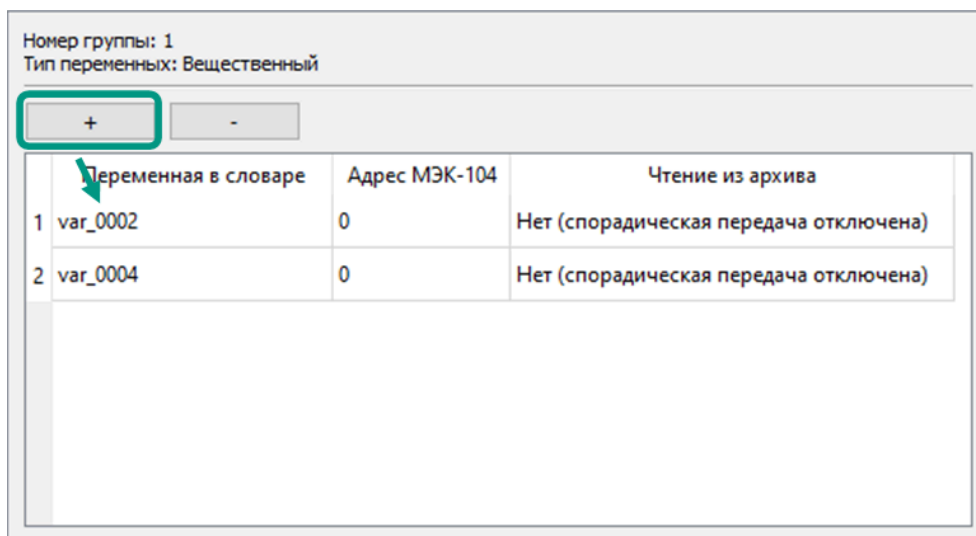
- **SQ (метод адресации)** - для выбора доступны два варианта: «0» - адресация индивидуальных одиночных элементов или комбинаций элементов в наборе объектов информации одинакового типа, параметр «1» - адресация последовательности одиночных элементов информации или одинаковых комбинаций элементов в единственном объекте ASDU;
- **спорадическая передача** - во включенном состоянии каждое измененное значение переменной из группы будет отправлено во внешнюю систему, без ожидания запроса.
- Для этого, из внешней системы требуется предварительно установить соединение с контроллером. В выключенном состоянии данные передаются только по запросу.

7.3.3.2 Управление переменными в МЭК-104

Для настройки параметров переменных, значения которых будут передаваться с использованием протокола МЭК-104, выберите раздел соответствующей группы в дереве конфигурации. После выбора откроется форма настройки параметров.



Добавление переменной осуществляется нажатием кнопки , удаление - нажатием кнопки . Добавленные переменные отображаются только на форме настройки параметров группы.



Для настройки переменной доступны параметры:

- **имя переменной в словаре** - служит для идентификации переменной в технологическом приложении. Для изменения дважды щелкните мышью по имени и в открывшемся окне выберите глобальную или локальную переменную.
- **адрес МЭК-104** - уникальный адрес переменной по протоколу МЭК-104 в пределах всех МЭК-адресов проекта, используется для идентификации переменной в SCADA-системах;
- **чтение из архива** - чтение значений из архива доступно, если у переменной установлен атрибут «Архивация», а для группы переменных включен параметр «Спорадическая передача» (*передача данных по мере их изменения*). Используется для накопления значений переменной в памяти контроллера и последующей их передачи во внешнюю систему, если на момент изменения значения переменной соединение с внешней системой отсутствовало. В отключенном состоянии все значения переменной, изменяющиеся в период отсутствия связи, будут потеряны.



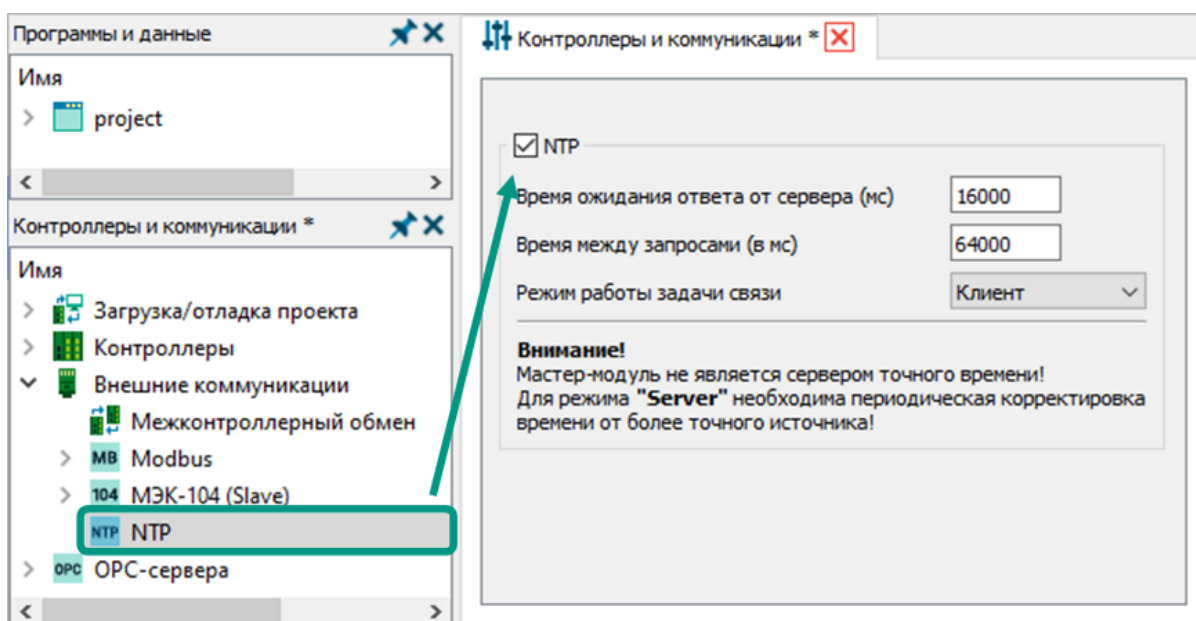
Перед добавлением переменных в группу МЭК-104 убедитесь, что в глобальном или одном из локальных словарей имеются переменные с типом данных, соответствующим типу данных группы. Если в глобальном или локальных словарях отсутствуют переменные с требуемым типом данных, при попытке их добавления в группу будет выведено предупреждение.

Для добавления переменной в глобальный или локальный словарь следуйте инструкциям из раздела [Управление переменными](#) настоящего руководства.

7.3.4 Задачи связи NTP

Протокол NTP используется для синхронизации системного времени на сетевых устройствах и узлах. В качестве источника эталонного времени может использоваться общедоступный NTP-сервер в локальной или глобальной сети. NTP-клиент с определенной периодичностью обменивается данными с NTP-сервером для корректировки своего текущего системного времени. Установленная в контроллере исполнительная система Unimod PRO 2 позволяет ему обмениваться данными с сервером времени или самому выступать в качестве сервера.

В разделе «NTP» дерева конфигурации настраиваются общие параметры работы контроллера по протоколу NTP.



Для разрешения обмена по протоколу NTP на форме настройки параметров отметьте галочкой пункт «NTP» и настройте параметры:

- **время ожидания ответа от сервера** - период времени в миллисекундах, в течение которого контроллер после отправки запроса будет ожидать ответ от сервера времени. После истечения указанного периода в системные события будет внесена соответствующая запись.
- **время между запросами** - период времени в миллисекундах между отправляемыми запросами.
- **режим работы задачи связи** - определяет, в какой роли будет выступать контроллер: «Клиент» - контроллер с указанной периодичностью направляет запросы на сервер времени для корректировки своего текущего времени, «Сервер» - контроллер выступает источником времени для других устройств, «Клиент+сервер» - контроллер, отправляя запросы на сервер времени, корректирует свое текущее время и отправляет его по запросу другим устройствам.

Для создания задачи связи NTP используйте функциональные блоки из системной библиотеки: «*TIME_FROM_SERVER*» - для получения текущего времени с удаленного устройства, «*TIME_SYNC*» - для синхронизации времени со временем удаленного устройства. Подробное описание блоков приведено в документе «Руководство по программированию».

Системная библиотека

Поиск:

Имя	Тип	Атрибут	Комментарий
TIME_FROM_SERVER			Получение текущего времени с удаленного устройства
→ RUN	булевский	Вход	Режим работы: TRUE - выполнить, FALSE - не выполнять
→ 14 ID_DRV	целый	Вход	Идентификатор устройства 0 - GPS-приемн...
→ Str PARAM	строка	Вход	Дополнительные параметры
← d8 UTC_MSEC	дв.вещественный	Выход	Время в формате UTC
← 14 FAULT	целый	Выход	Код ошибки
> TIME_GET			Получение текущего времени
> TIME_SET			Установка текущего времени
TIME_SYNC			Синхронизация времени с временем удаленного устройства
→ RUN	булевский	Вход	Режим работы: TRUE - выполнить, FALSE - не выполнять
→ 14 ID_DRV	целый	Вход	Идентификатор устройства 0 - GPS-приемн...
→ PARAM	ссылка	Вход	Дополнительные параметры
→ 14 TIMEZONE	целый	Вход	Часовой пояс
→ 14 MOD_PERIOD	целый	Вход	Период синхронизации модулей ввода-вывода (сек)
← 14 FAULT	целый	Выход	Код завершения операции
← d8 UTC_MSEC	дв.вещественный	Выход	Время в формате UTC
← 14 CUR_LINE	целый	Выход	Текущая линия
← SYNC	булевский	Выход	Признак цикла синхронизации
> TIME_TO			Преобразование времени из формата в UTC развернутый вид



Мастер-модуль не является сервером точного времени. При выборе режима «Сервер» необходима периодическая корректировка текущего времени модуля от более точного источника.

7.4 Конфигурирование OPC-серверов

Технологии OPC предоставляют единый интерфейс для управления объектами автоматизации и технологическими процессами. Обычно OPC технология применяется для обмена данными между контроллерами и SCADA-системами, но также возможна организация сложных систем на разных уровнях АСУ ТП. В основе OPC технологии лежит архитектура клиент-серверного взаимодействия. В качестве OPC-клиента выступают системы диспетчеризации, управления и сбора данных (ARM оператора, SCADA-, ERP-, MES-системы и др.), а в качестве сервера могут выступать контроллеры или отдельные компьютеры.

Среда разработки Unimod PRO 2 поддерживает три спецификации OPC:

- **OPC DA (Data Access)** — спецификация для обмена данными в режиме реального времени, при этом OPC-сервер и OPC-клиенты могут работать только под управлением операционной системы Windows;
- **OPC HDA (Historical Data Access)** - спецификация для доступа к сохраненным архивным данным, по аналогии с OPC DA поддерживает работу только на устройствах с операционной системой Windows;
- **OPC UA (Unified Architecture)** - современная аппаратно- и платформо-независимая спецификация, обеспечивает защищенный и надежный обмен данными между устройствами с разными операционными системами и объединяет возможности OPC DA и OPC HDA.

OPC-сервер TREI с поддержкой спецификаций DA, HDA и UA устанавливается вместе со средой разработки Unimod PRO 2, запускается автоматически и не требует дополнительной настройки. Предназначен для организации обмена данными между контроллерами фирмы TREI и приложениями-клиентами OPC.

Для идентификации конкретного мастер-модуля используется термин - «*абонент*». Список всех абонентов, с которыми предполагается обмен данными, составляет конфигурацию OPC-сервера. В процессе работы OPC-сервер с определенной периодичностью отправляет абонентам запросы на чтение или запись данных, а в ответ на запросы получает список требуемых переменных или подтверждение выполненной операции. OPC-серверу доступны для чтения и записи все глобальные и локальные переменные с мастер-модулей. Доступ к данным будет открыт после загрузки технологического приложения в мастер-модуль и его запуска под управлением исполнительной системы Unimod PRO 2.

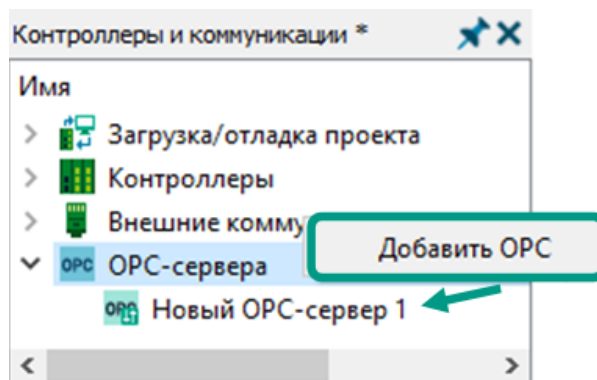
Алгоритм конфигурирования OPC-серверов TREI:

1. Установите среду разработки Unimod PRO 2 на компьютер, который будет использоваться в качестве OPC-сервера;
2. В настройках среды разработки включите отображение раздела «OPC-сервера»;
3. Добавьте в дерево конфигурации новый OPC-сервер и в настройках подключения укажите его IP-адрес и номер порта;
4. Прочитайте конфигурацию OPC-сервера;
5. Отредактируйте базу абонентов;
6. Соберите проект;
7. Загрузите отредактированную конфигурацию в OPC-сервер.

Настройка отображения OPC-серверов описана в разделе [Настройка среды разработки](#), а процесс сборки - в разделе [Сборка проекта](#).

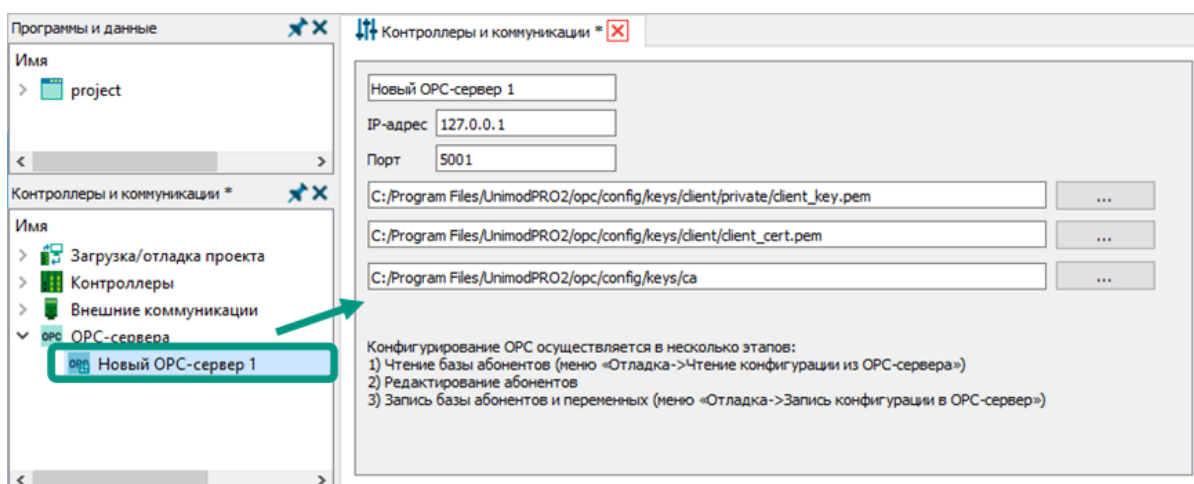
7.4.1 Управление OPC-серверами

Для добавления OPC-сервера в дереве конфигурации нажмите правой кнопкой мыши по разделу «*OPC-сервера*» → из контекстного меню выберите пункт «*Добавить OPC*» → в дереве конфигурации новый элемент будет добавлен в конец списка OPC-серверов.

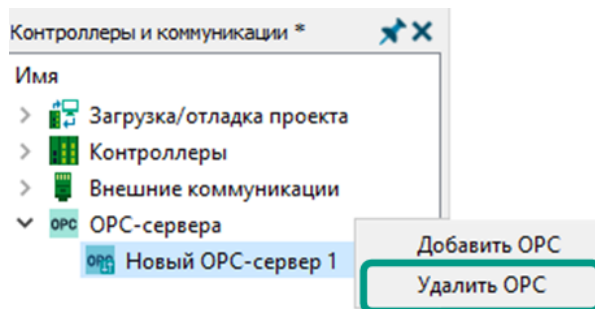


Настройка параметров подключения производится на форме, закрепленной за каждым разделом OPC-сервера. Форма настройки параметров открывается при выборе OPC-сервера в дереве конфигурации. Параметры подключения:

- **имя OPC-сервера** - отображается в дереве конфигурации, не имеет ограничений по вводимым символам и их порядку, длина до 128 символов;
- **IP-адрес и номер порта** - параметры сетевого подключения к OPC-серверу для чтения и записи его конфигурации;
- **три поля ввода** - три адреса в файловой системе, указывающие на файлы с приватным (секретным) ключом, локальным сертификатом и публичным/корневым сертификатом соответственно. Файлы с ключами и сертификатами располагаются в каталоге среды разработки и прописываются автоматически.

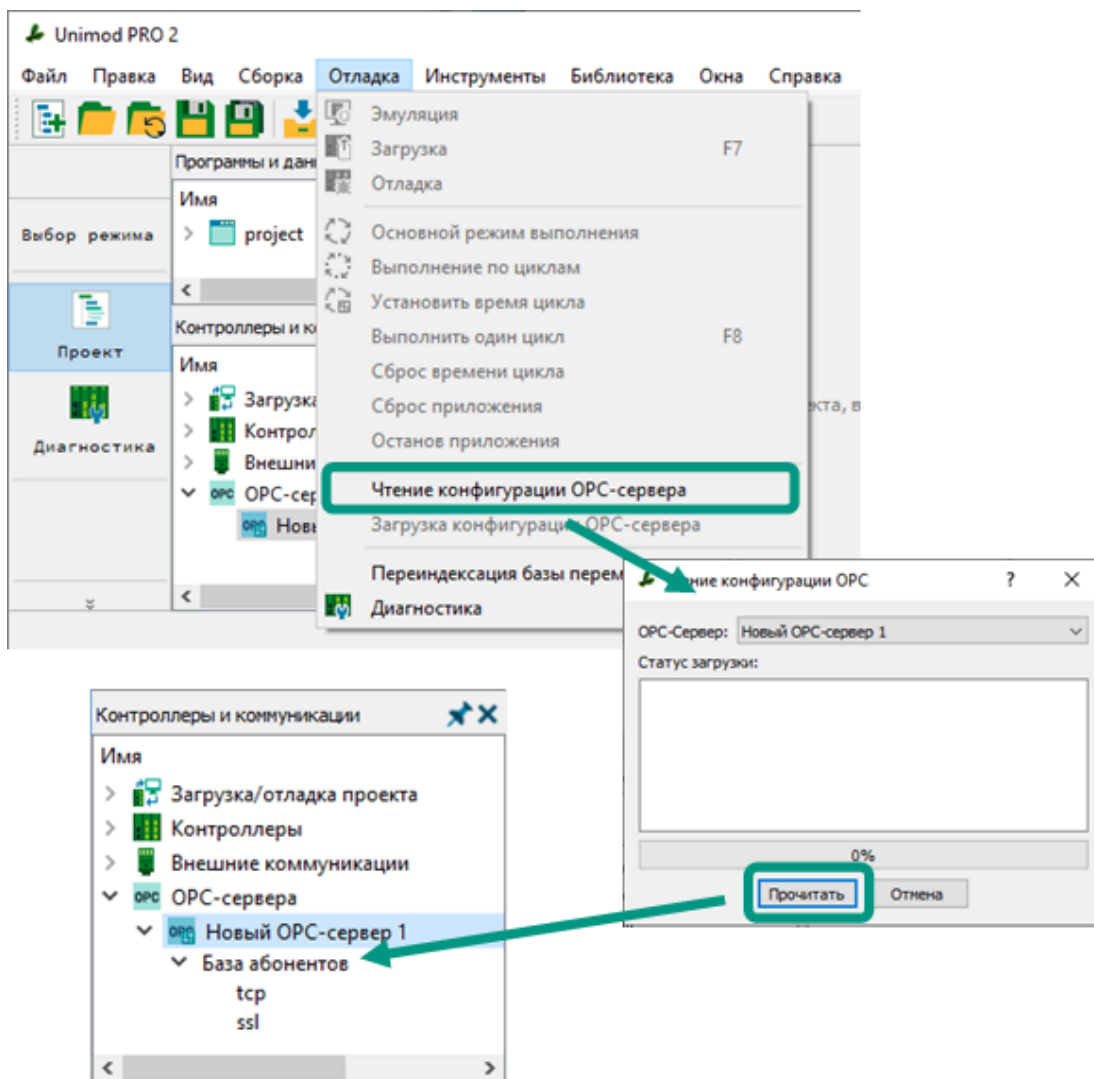


Для удаления OPC-сервера в дереве конфигурации нажмите правой кнопкой мыши по OPC-серверу → из контекстного меню выберите пункт «Удалить OPC».



7.4.2 Чтение конфигурации OPC-сервера

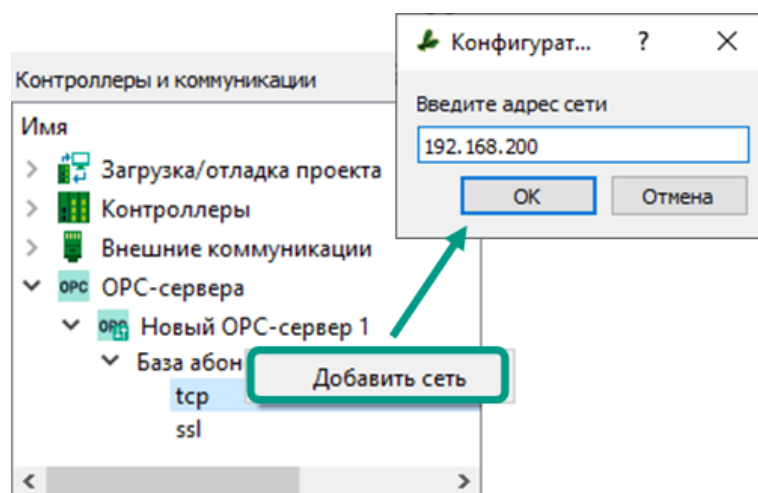
Для чтения конфигурации OPC-сервера выберите пункт меню «Отладка → Чтение конфигурации OPC-сервера» → в открывшемся окне из свернутого списка выберите требуемый OPC-сервер → нажмите кнопку «Прочитать». При успешном чтении конфигурации будет выведено окно уведомления, а в дереве конфигурации будет добавлен элемент «База абонентов». Подробнее про данный элемент описано в разделе [Редактирование базы абонентов](#).



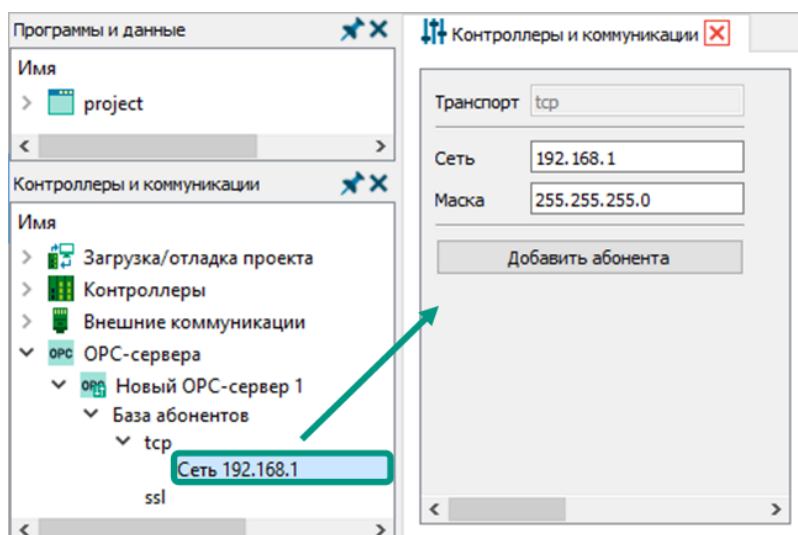
7.4.3 Редактирование базы абонентов

Раздел «База абонентов» может быть добавлен в дерево конфигурации только после чтения конфигурации OPC-сервера. Абонентами являются мастер-модули, выступающие конечным источником данных. Все абоненты в дереве конфигурации группируются в подразделах, соответствующих используемому протоколу передачи данных и номеру подсети.

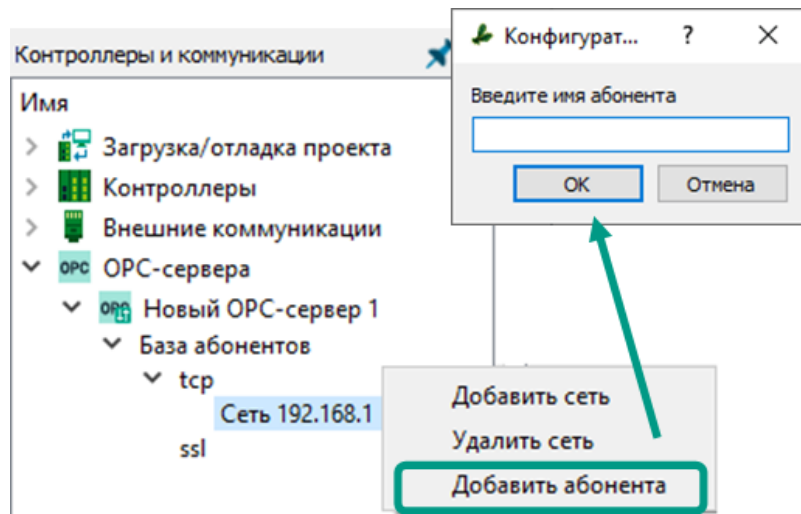
Для добавления подсети в дереве конфигурации нажмите правой кнопкой мыши по требуемому протоколу → из контекстного меню выберите пункт «Добавить сеть» → в открывшемся окне введите первые 3 цифры IP-адреса подсети → нажмите кнопку «Ок». После чего раздел новой сети будет добавлен в дерево конфигурации.



Форма настройки параметров подсети создается для каждого добавленного элемента независимо от выбранного протокола передачи данных. Для редактирования доступны IP-адрес (первые 3 цифры) и маска подсети. Для быстрого добавления абонента в подсеть нажмите кнопку «Добавить абонента».



Для добавления абонента нажмите правой кнопкой мыши по разделу подсети → из контекстного меню выберите пункт «Добавить абонента» → в открывшемся окне введите имя абонента (не более 128 символов) → нажмите кнопку «Ок». После чего новый абонент будет добавлен в дерево конфигурации.



Форма настройки параметров закреплена за каждым абонентом подсети.

The image shows a configuration form for a subscriber. It has several input fields and a dropdown menu. The fields are: "Имя" (value: id_1), "Адрес" (value: 1), "Порт" (value: 6000), "Размер фрейма" (value: 8192), "Повторы" (value: 3), "Таймаут" (value: 1000), "Таймаут для служебных команд" (value: 10000), "Таймаут если мастер занят" (value: 10), and "Доступ к данным" (value: разрешен). Below these fields are two buttons: "Добавить" and "Удалить". At the bottom, there is a section labeled "Резервные линии" with an empty text area below it.

Параметры абонентов подсети:

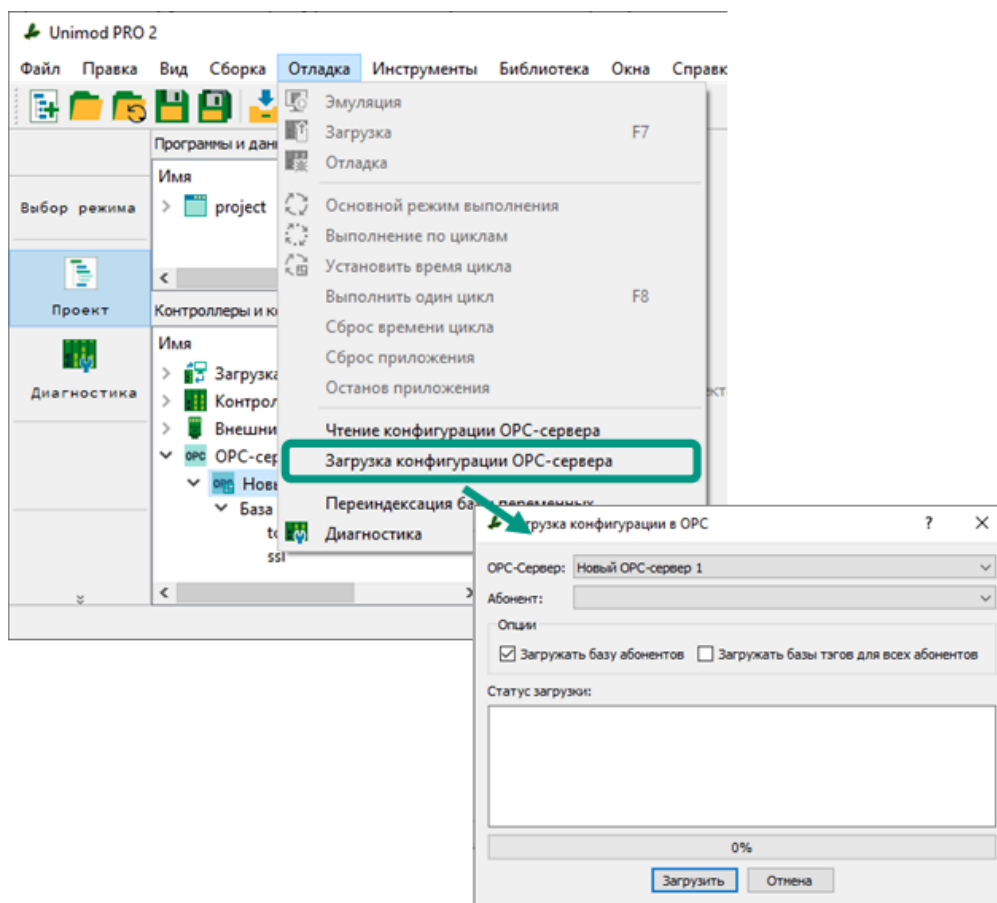
- **имя абонента** - отображается в дереве конфигурации приложения и в OPC-сервере, не должно превышать 128 символов.
- **адрес** - целое число в диапазоне от 1 до 255, задает последнюю цифру в сетевом адресе мастер-модуля, например: $192.168.200$ (адрес сети) + 1 (адрес абонента) = $192.168.200.1$ (адрес мастер-модуля в сети).
- **порт** - номер порта мастер-модуля, на который будут поступать запросы от OPC-сервера.
- **размер фрейма** - минимальный размер передаваемого по сети пакета данных.
- **повторы** - количество попыток получения ответа от мастер-модуля. Одна попытка считается израсходованной, если в течение таймаута ответ на запрос не поступил. При превышении указанного числа на сервере будет выведена ошибка обмена. После чего снова будет отправлен запрос, но отсчет повторов начнется с нуля.
- **таймаут** - период времени в миллисекундах между отправкой запроса и поступлением ответа. Если за указанное время ответ от мастер-модуля не поступил, одна попытка (повтор) подключения считается истраченной, а отсчет времени перезапускается.
- **таймаут для служебных команд** - период времени в миллисекундах между отправкой служебной команды, на выполнение которой требуется длительный промежуток времени, и поступлением подтверждения о ее выполнении.
- **таймаут, если мастер занят** - если после получения запроса мастер-модулю требуется время для формирования ответа, например, нужно опросить модули ввода-вывода, то он уведомляет OPC-сервер о своей занятости подготовкой ответа. После чего OPC-сервер с периодичностью, указанной в миллисекундах в данном параметре, будет направлять на мастер-модуль напоминания о необходимости предоставить запрошенные данные. Напоминания повторяются до возвращения от мастер-модуля ответа.
- **доступ к данным** - при разрешенном доступе данные будут доступны для чтения и записи из внешних SCADA-систем, при запрещенном - недоступны.
- **резервные линии** - указываются резервные абоненты (мастер-модули), с которыми будет взаимодействовать OPC-сервер при отсутствии связи с основным мастер-модулем. В резервную линию могут быть добавлены только абоненты с запрещенным доступом к данным, это позволяет OPC-серверу TREI взаимодействовать только с одним абонентом. При отсутствии связи с основным абонентом, доступ к данным резервного абонента открывается автоматически.

Для удаления подсети или абонента нажмите правой кнопкой мыши по удаляемому элементу → из контекстного меню выберите пункт «Удалить сеть» или «Удалить абонента».

После завершения редактирования базы абонентов, необходимо собрать проект и загрузить базу в ОРС-сервер.

7.4.4 Загрузка конфигурации OPC-сервера

Для загрузки конфигурации OPC-сервера соберите проект → в открывшемся окне выберите OPC-сервер и настройте загружаемые в него данные → нажмите кнопку «Загрузить».



В окне загрузки конфигурации настраивается объем загружаемых данных:

- **выбран один абонент (без дополнительных опций)** - в OPC-сервер будут загружены теги выбранного абонента, но в базу абонентов OPC-сервера он добавлен не будет.
- **опция «Загружать базу абонентов»** - при выборе опции в OPC-сервер кроме тегов выбранного абонента также будет загружена вся база абонентов;
- **опция «Загружать базы тегов для всех абонентов»** - при выборе опции в OPC-сервер будут загружены теги для всех абонентов сетей TCP, IP-адреса которых совпадают с адресами сетевых настроек контроллера в открытом проекте.

В OPC-технологии теги - переменные глобального и всех локальных словарей приложения. В OPC-сервер могут быть загружены только теги абонентов из сетей TCP, IP-адреса которых совпадают с адресами, указанными в сетевых настройках выбранного в проекте контроллера. При этом сконфигурированная база абонентов, не зависимо от выбранного протокола передачи данных и установленных для абонентов IP-адресов, в полном объеме может быть выгружена в OPC-сервер.

8 Отладка

Среда разработки Unimod PRO 2 в режиме отладки позволяет проверить правильность выполнения собранного технологического приложения и проанализировать изменение значений переменных для выявления возможных ошибок. Отладка доступна только после безошибочной сборки приложения и может осуществляться с использованием реального контроллера или в режиме эмуляции.

При использовании реального контроллера необходимо предварительно настроить параметры подключения, собрать и загрузить приложение на контроллер, после чего перейти в режим отладки. Процесс сборки и отладки на контроллере описан в текущем разделе, а процесс загрузки в разделе [Загрузка в контроллер](#).

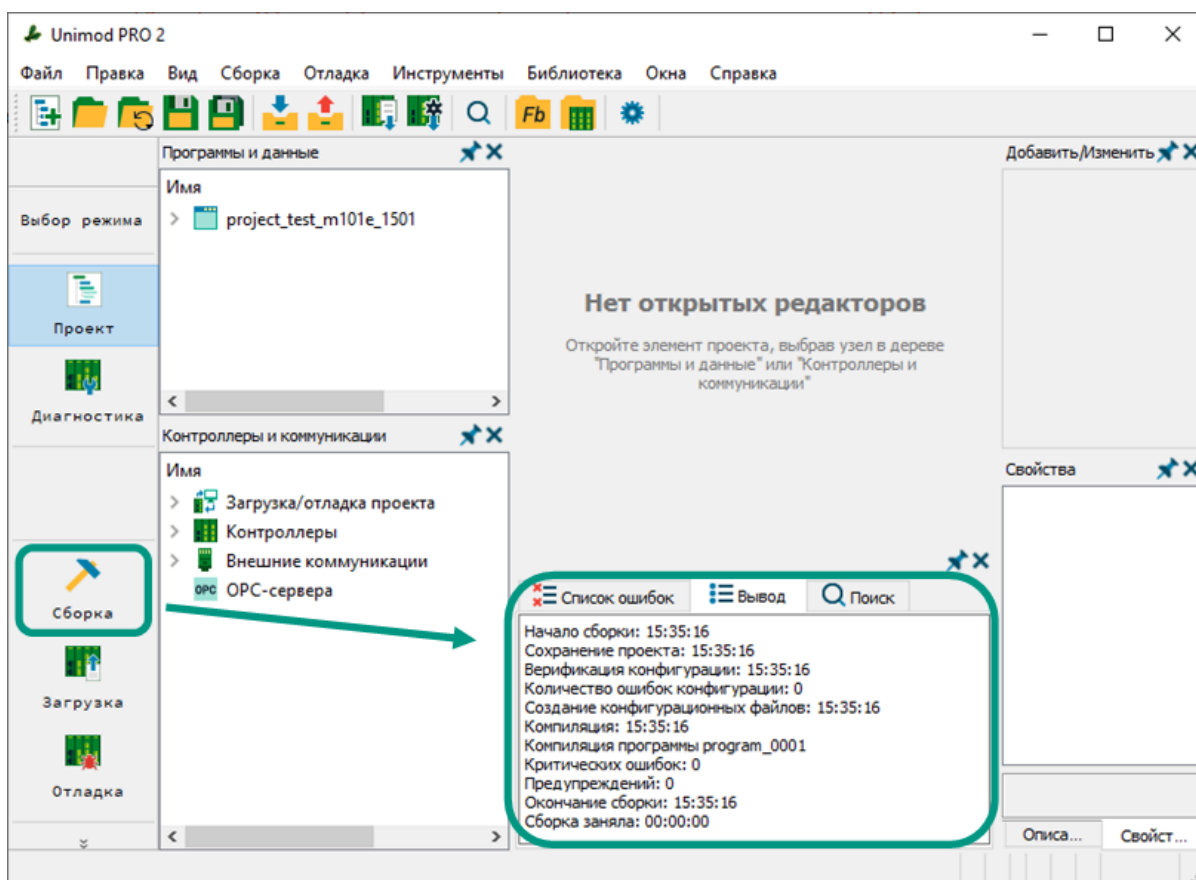
При отсутствии доступа к реальному контроллеру отладка приложения может осуществляться на виртуальном контроллере в режиме эмуляции. Для перехода в режим эмуляции достаточно безошибочной сборки приложения.

8.1 Сборка проекта

В процессе сборки среда разработки Unimod PRO 2:

1. проводит проверку конфигурационных параметров исполнительной системы для выбранного типа мастер-модуля, а также проверяет параметры задач связи;
2. при отсутствии ошибок создает конфигурационные файлы;
3. проверяет написанный на языках ST или FBD/LD программный код на наличие ошибок;
4. при отсутствии ошибок исходный программный код компилируются в бинарный файл;
5. осуществляется сборка файлов технологического приложения, которое в последующем будет загружено в контроллер и начнет выполняться под управлением исполнительной системы Unimod PRO 2.

Для сборки проекта на панели выбора режима работы выберите команду «Сборка». Альтернативными вариантами сборки проекта являются выбор пункта меню «Сборка → Сборка» или нажатие клавиши «F9». Сборка проекта доступна только в режиме работы «Проект».




Информация о сборке проекта с указанием времени сборки, обнаруженных ошибках конфигурации и компиляции будет выведена на вкладке «Вывод».

При обнаружении ошибок их список будет выведен на вкладке «Список ошибок» с указанием краткого описания ошибки, а также номера строки, номера столбца и названия программного объекта, в котором обнаружена ошибка.

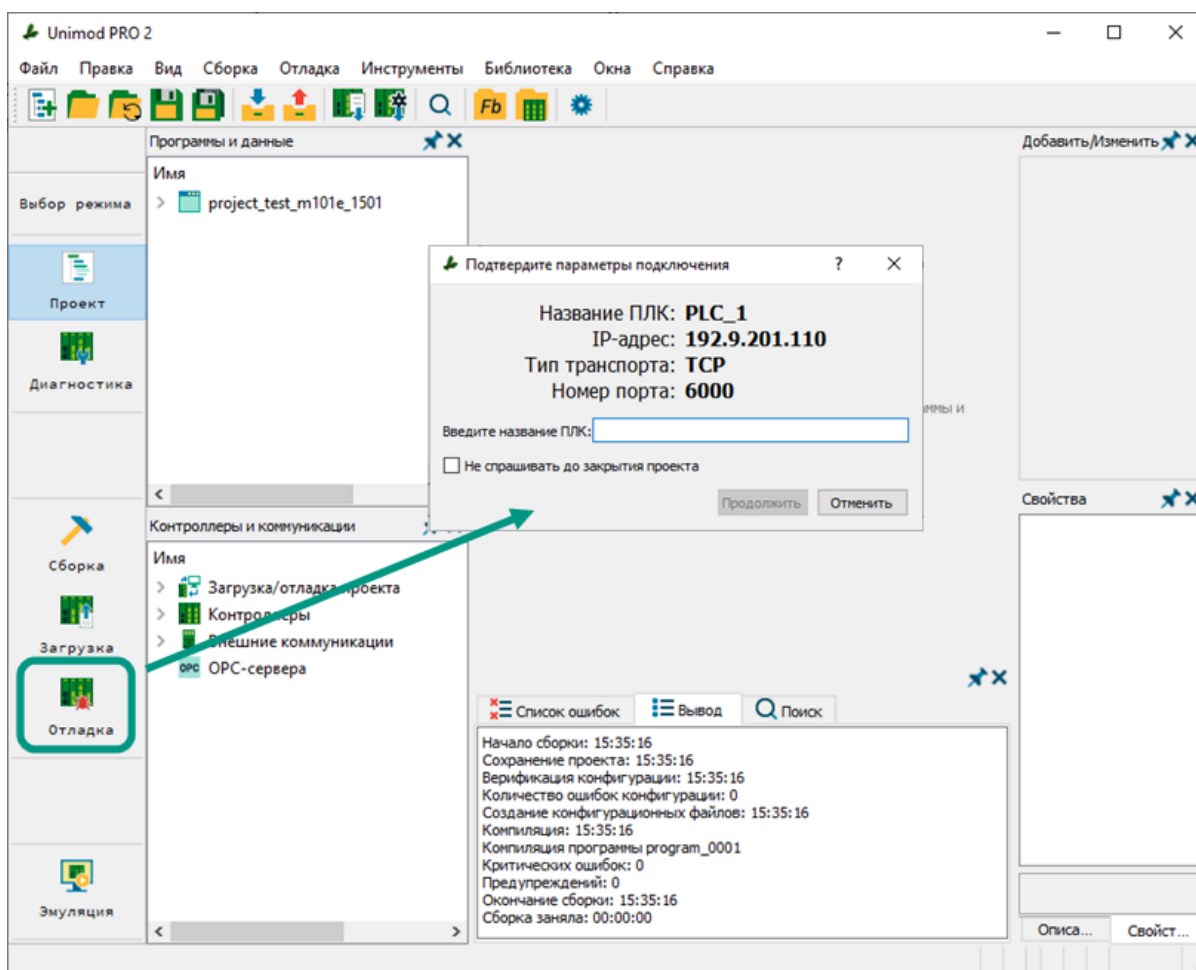
Номер каждой последующей сборки проекта увеличивается на единицу. Команды «Отладка», «Загрузка» и «Эмуляция» доступны только при безошибочной сборке проекта.


8.2 Отладка на контроллере

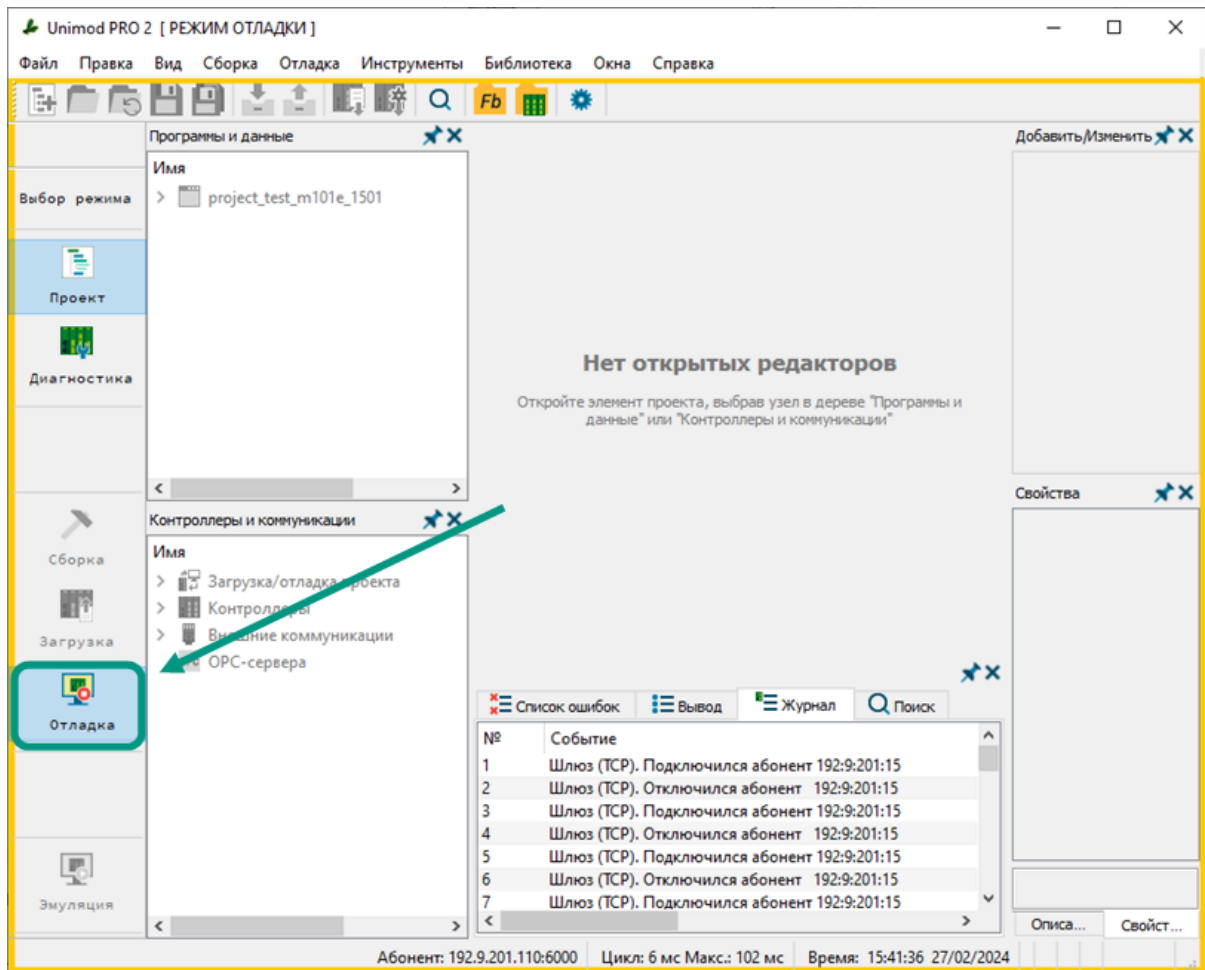
Перед началом отладки необходимо организовать соединение между ПК и реальным контроллером и загрузить собранное технологическое приложение в контроллер. Режим отладки доступен только после сборки проекта. При отсутствии доступа к реальному контроллеру воспользуйтесь отладкой приложения в [режиме эмуляции](#).

Для включения режима отладки на панели выбора режима работы нажмите команду «Отладка» (символ ) или выберите пункт меню «Отладка → Отладка» → в открывшемся окне проверьте параметры подключения, введите название целевого контроллера и нажмите на кнопку «Продолжить». При успешном подключении к контроллеру главное окно программы будет выделено желтым цветом. В режиме отладки для редактирования доступны только значения переменных.

При подключении вводимое название контроллера должно совпадать с названием, указанным в разделе «Контроллеры» дерева конфигурации. Требования о выводе окна с параметрами подключения и дополнительном вводе названия контроллера можно установить в [настройках среды разработки](#). Эти требования позволяют пользователю убедиться, что он не ошибся при выборе контроллера для отладки.

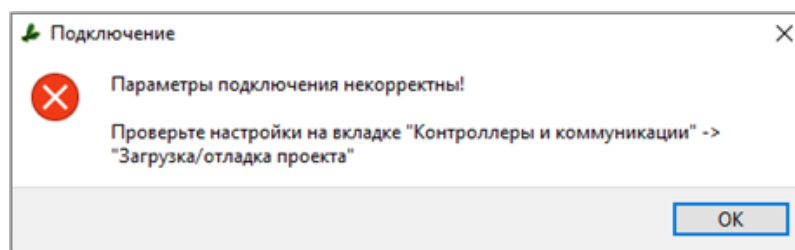


Для выхода из режима отладки на панели выбора режима работы повторно нажмите на команду «Отладка» (символ ).

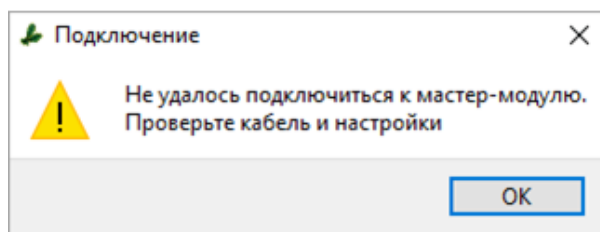


Возможные ошибки и предупреждения при подключении к контроллеру:

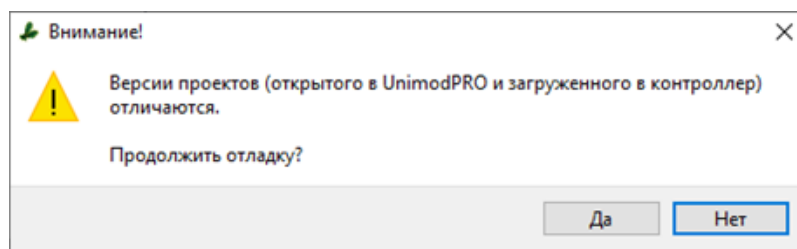
При отсутствии параметров подключения будет выведена ошибка. В данном случае в дереве конфигурации перейдите в раздел «Загрузка/Отладка проекта → Подключение к PLC_1» и укажите корректные параметры подключения: IP-адрес контроллера, номер порта и протокол передачи данных. Подробнее в разделе [Параметры подключения к контроллеру](#).



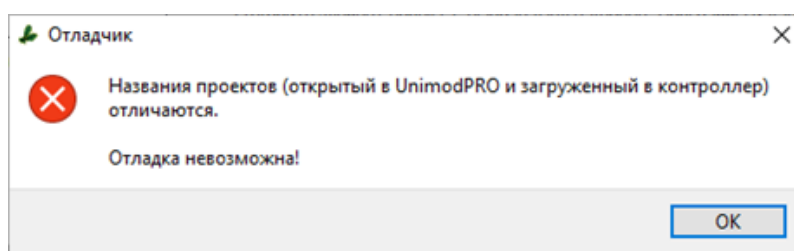
При неправильно заданных параметрах подключения или отсутствии связи с контроллером будет выведена ошибка. В данном случае в дереве конфигурации перейдите в раздел «*Загрузка/Отладка проекта → Подключение к PLC_1*» и укажите корректные параметры подключения. Подробнее в разделе [Параметры подключения к контроллеру](#). Если ошибка сохраняется при корректных параметрах, убедитесь в наличие связи с контроллером. Подробнее в разделе [Режим диагностики](#).



При несовпадении версий текущего и загруженного в контроллер проектов, в процессе подключения будет выведено предупреждение. Для продолжения отладки нажмите кнопку «*Да*», при этом будет производиться отладка уже загруженного в контроллере приложения. Для отладки текущего приложения, загрузите его в контроллер. Подробнее в разделе [Загрузка проекта в контроллер](#).




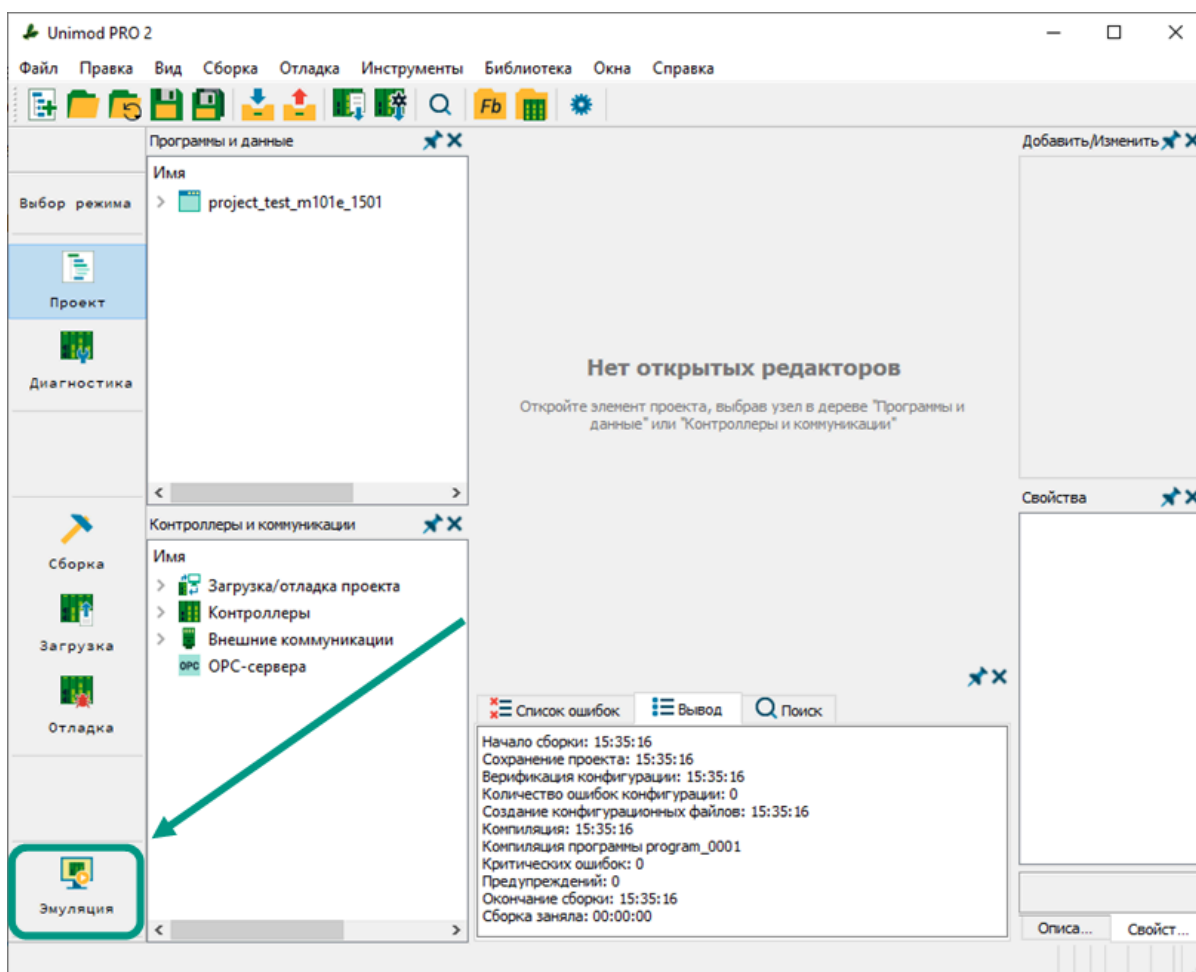
При отличии названий текущего и загруженного в контроллер проектов отладка невозможна, а при попытке подключения будет выведено предупреждение. Режим отладки доступен только при совпадении имен проектов. Для определения имени загруженного в контроллер проекта перейдите в режим загрузки и подтвердите подключение к контроллеру → в открывшемся окне загрузки будет указано имя приложения, загруженного в контроллере.




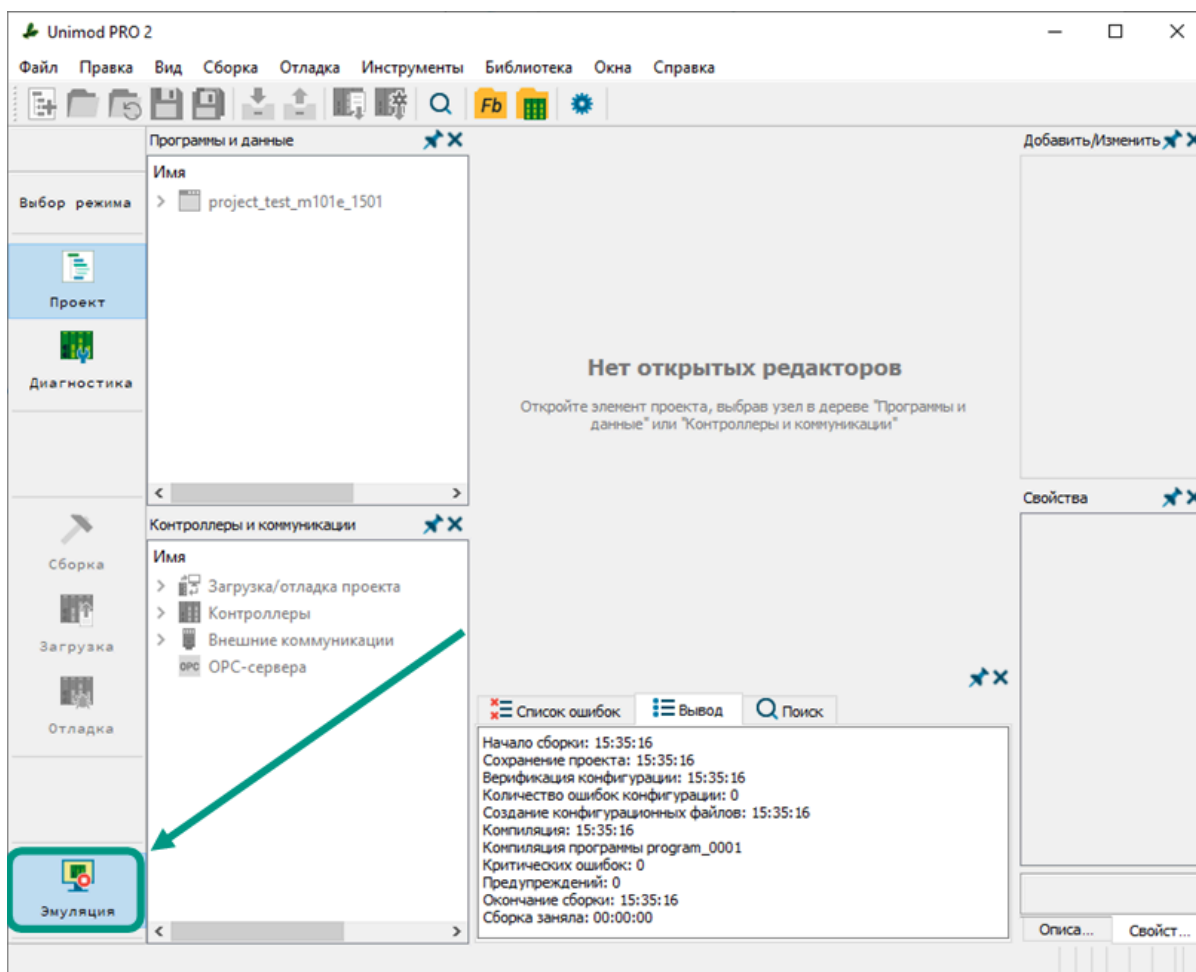
8.3 Режим эмуляции

Режим эмуляции позволяет проверить правильность выполнения технологического приложения и проанализировать изменение значений переменных для выявления возможных ошибок на виртуальном контроллере, используется для отладки технологического приложения при отсутствии доступа к реальному контроллеру.

Для переключения в режим эмуляции на панели выбора режима работы нажмите команду «Эмуляция» (символ ) или выберите пункт меню «Отладка → Эмуляция». В режиме эмуляции доступны для изменения только значения переменных.



Для выхода из режима эмуляции на панели выбора режима работы повторно нажмите на команду «Эмуляция» (символ )



8.4 Возможности отладки

Среда разработки Unimod PRO 2 в процессе отладки технологического приложения на контроллере или в режиме эмуляции позволяет просматривать и изменять значения глобальных и локальных переменных в программах, дочерних программах и функциях.

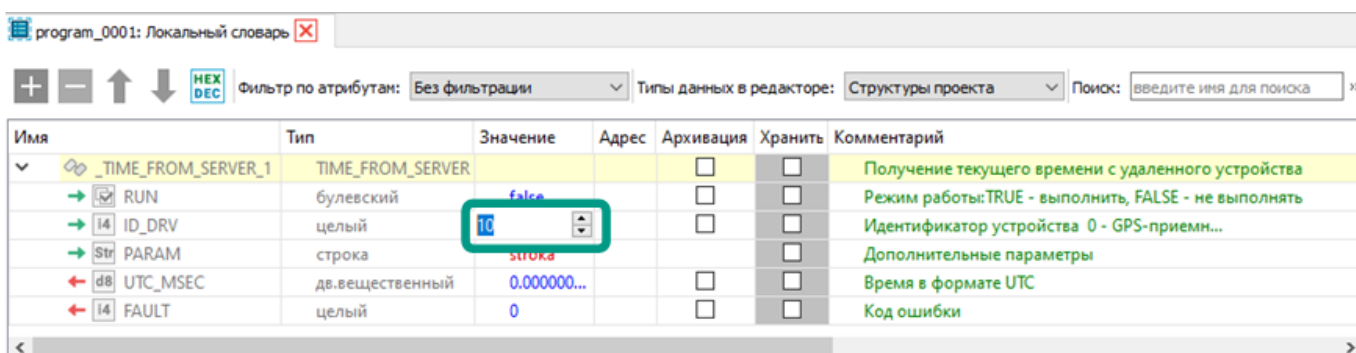
Значения переменных в словаре данных структуры недоступны для изменения, так как структура представляет собой пользовательский тип данных, который используется только для объявления переменных структурного типа в глобальном или локальном словаре, а изменение в процессе отладки значений полей структуры приведет к изменению всех переменных с данным структурным типом, что наоборот усложнит отладку. После объявления переменной с типом данных пользовательской структуры в глобальном или локальном словаре, в процессе отладки будут доступны для просмотра и редактирования все ее вложенные поля.

Просмотр и изменение значений осуществляются в редакторе словарей или в редакторах программных объектов. В редакторе ST значения переменных доступны только для просмотра, а в редакторе FBD/LD значения переменных в процессе отладки могут быть просмотрены и изменены. Но изменение в программных объектах текста на языке ST или графических схем на языках FBD/LD в процессе отладки недоступно.

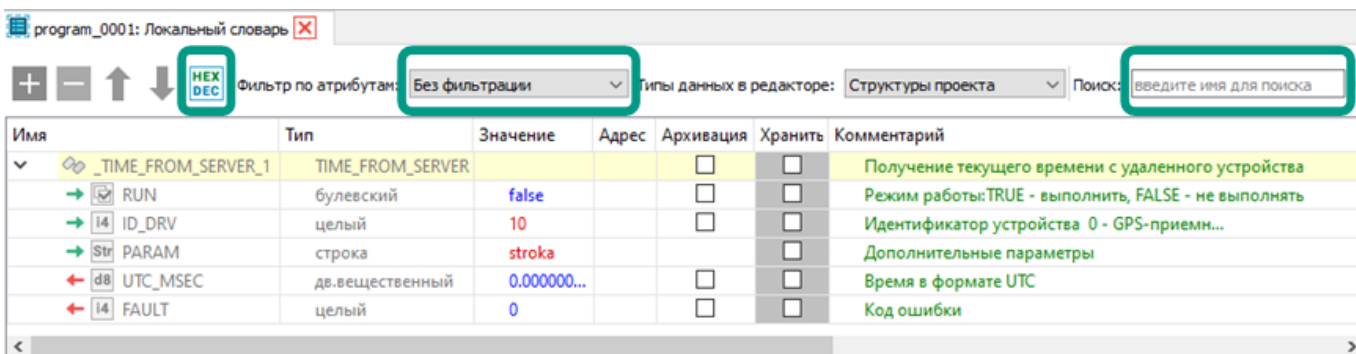
8.4.1 Отладка в редакторе словаря

При отладке технологического приложения на реальном контроллере или в режиме эмуляции в редакторе словаря доступны для просмотра и изменения значения глобальных и локальных переменных в программах, дочерних программах и функциях. Значения переменных из словаря данных структуры доступны только для просмотра.

Для изменения значения переменной в локальном или глобальном словаре дважды кликните левой кнопкой мыши по ячейке с атрибутом «Значение» и измените его.

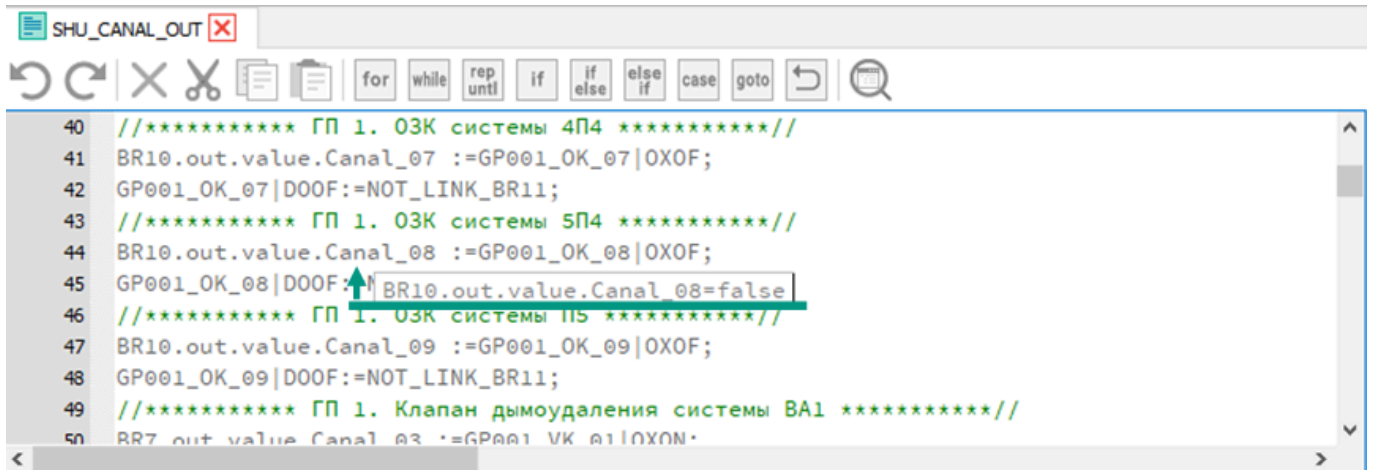


Кроме того, сохраняются возможности поиска переменной по имени, фильтрации списка переменных по значениям атрибутов и вывода значений переменных в шестнадцатеричном формате.



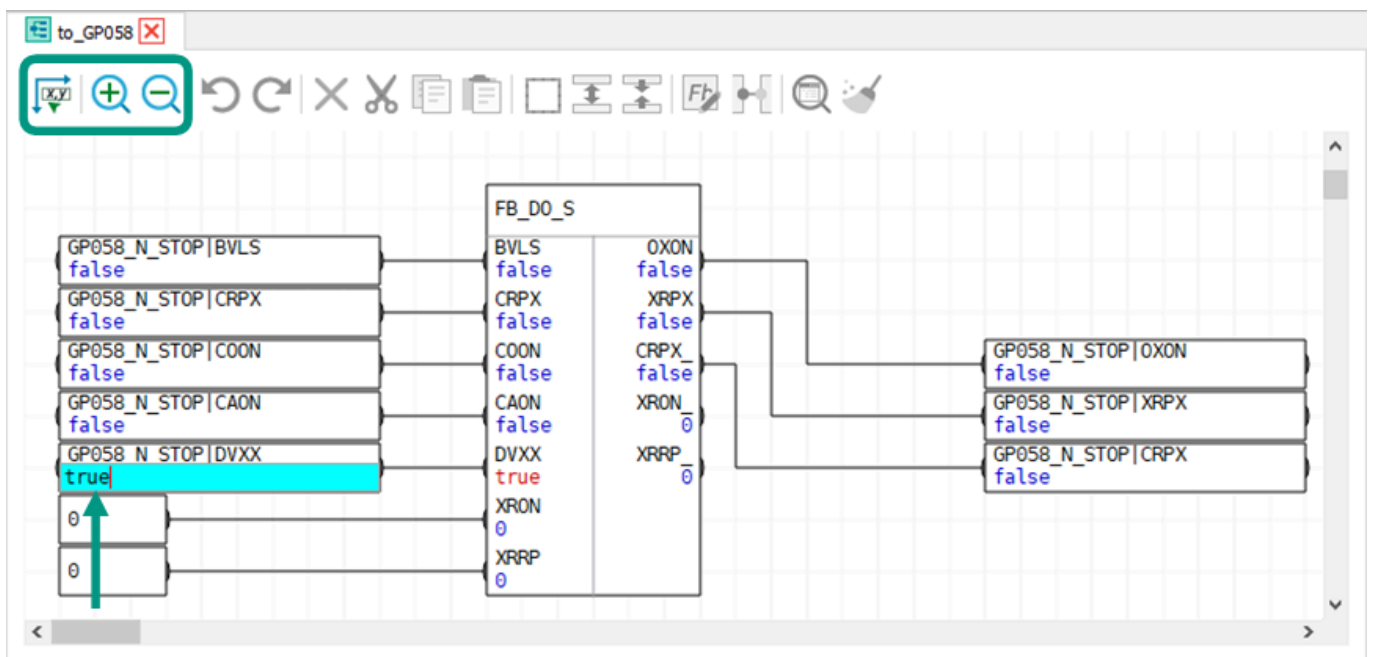
8.4.2 Отладка в редакторах программных объектов

Для просмотра значений переменных в редакторе ST наведите курсор мыши на имя переменной, ее текущее значение будет выведено во всплывающей подсказке, как на нижеуказанном изображении. В редакторе ST значения переменных недоступны для редактирования.



```
SHU_CANAL_OUT X
for while rep until if if else if case goto
40 //***** ГП 1. 03К системы 4П4 *****//
41 BR10.out.value.Canal_07 :=GP001_OK_07|OXOF;
42 GP001_OK_07|DOOF:=NOT_LINK_BR11;
43 //***** ГП 1. 03К системы 5П4 *****//
44 BR10.out.value.Canal_08 :=GP001_OK_08|OXOF;
45 GP001_OK_08|DOOF: BR10.out.value.Canal_08=false
46 //***** ГП 1. 03К системы ПБ *****//
47 BR10.out.value.Canal_09 :=GP001_OK_09|OXOF;
48 GP001_OK_09|DOOF:=NOT_LINK_BR11;
49 //***** ГП 1. Клапан дымоудаления системы ВА1 *****//
50 BR7.out.value.Canal_03 :=GP001.VK_01|OXON;
```

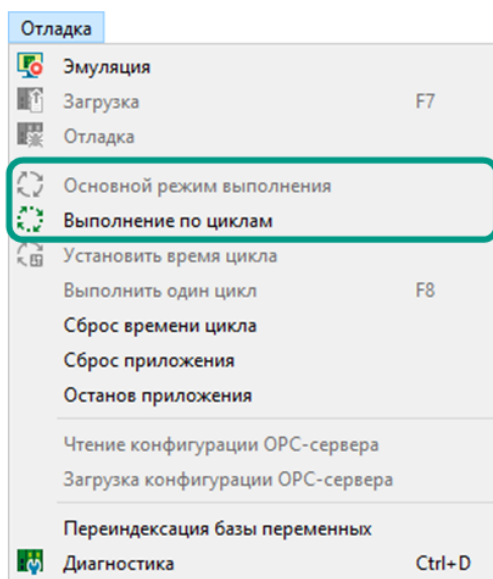
В редакторе FBD/LD значения переменных доступны для просмотра и изменения. Текущие значения отображаются внутри блоков около соответствующих входов и выходов. Для ввода нового значения дважды щелкните левой кнопкой мыши по текущему значению переменной, измените его и нажмите клавишу «Enter». Если на вход блока через линию связи поступает выходное значение с другого блока, то для изменения входного значения требуется изменить выходное значение в блоке-источнике.



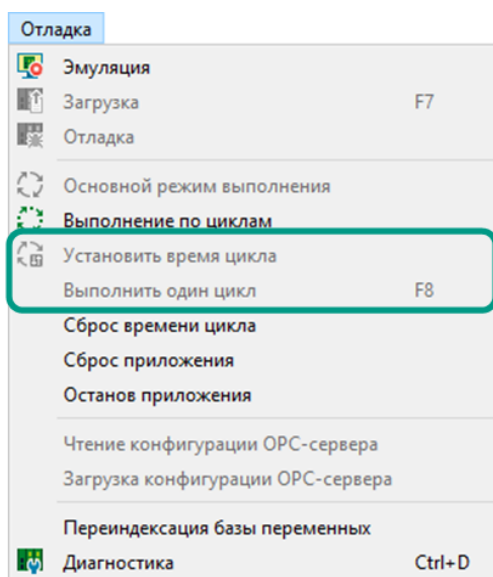
8.4.3 Управление циклом и приложением

Отладка в среде разработки Unimod PRO 2 может осуществляться в основном режиме выполнения и в режиме выполнения по циклам. По умолчанию отладка запускается в основном режиме выполнения, когда по завершении одного цикла выполнения, сразу запускается следующий. В режиме отладки по циклам приложение останавливается, а выполнение одного цикла запускается по команде пользователя. Отладка по циклам позволяет пошагово проанализировать алгоритм выполнения приложения и отследить изменения переменных.

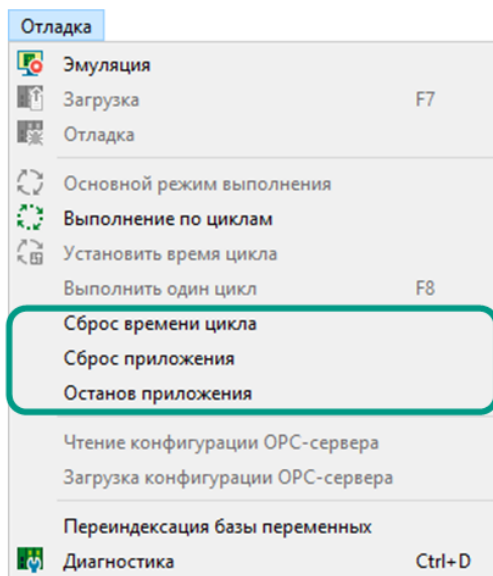
Команды выбора режима отладки размещены в разделе меню «Отладка».



Установка времени цикла и команда выполнения одного цикла доступны только в режиме выполнения по циклам.



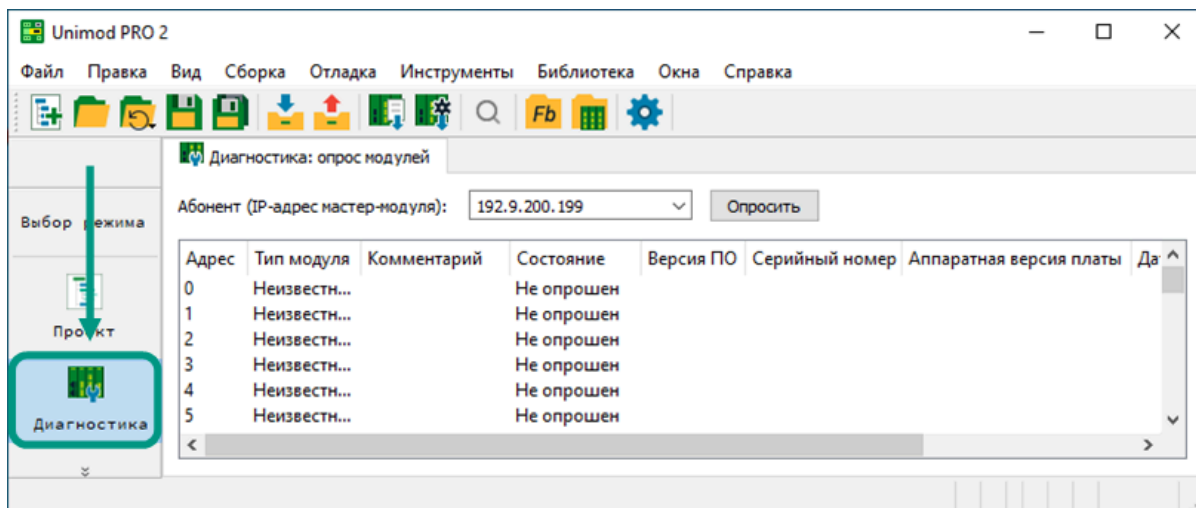
Команды сброса времени цикла, сброса приложения и остановки приложения доступны в любом режиме отладки. Команда «Сброс времени цикла» позволяет заново запустить цикл выполнения приложения, при этом само приложение не останавливается. Команда «Останов приложения» останавливает выполнение приложения, а команда «Сброс приложения» - сразу после остановки его перезапускает.



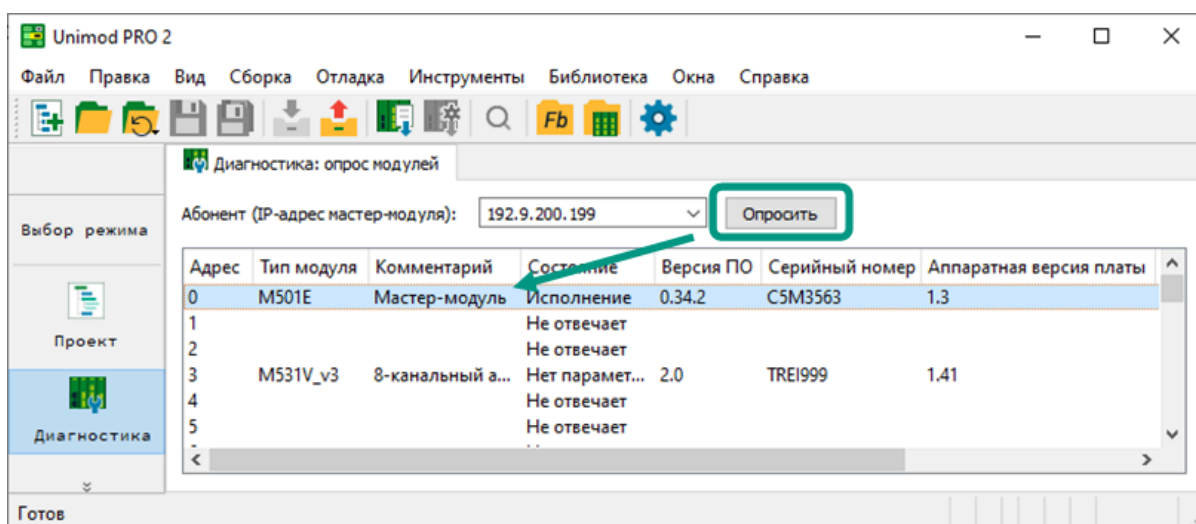
9 Режим диагностики

В режиме диагностики осуществляется анализ работы мастер-модуля и подключенных к нему модулей ввода-вывода.

Для запуска режима диагностики необходимо на панели выбора режима работы нажать кнопку «Диагностика» или выбрать пункт меню «Отладка → Диагностика».



При переходе в режим диагностики конфигурация мастер-модуля неизвестна. В данном случае под конфигурацией подразумевается совокупность мастер-модуля и подключенных к нему модулей ввода-вывода. Для определения конфигурации введите IP-адрес мастер-модуля и нажмите кнопку «Опросить».



После этого среда разработки установит соединение с мастер-модулем и опросит все подключенные к нему модули ввода-вывода. По результатам опроса будет выведен список всех доступных модулей с указанием их типа, серийного номера, аппаратной версии платы, даты изготовления и текущего состояния, а также

комментария с кратким описанием назначения модуля, версии и контрольной суммы установленного программного обеспечения.

В режиме диагностики может быть опрошено не более 255 модулей. До полного завершения опроса диагностика отдельных модулей недоступна. Для остановки процесса опроса нажмите кнопку *«Прервать»*.

9.1 Диагностика мастер-модуля

Диагностические возможности среды разработки позволяют просматривать режим работы, ошибки и системные события, редактировать текущее время, управлять циклом выполнения технологического приложения и обновлять установленное программное обеспечение, а также экспортировать сведения о контроллере и накопленных статистических данных.

Для выполнения диагностики мастер-модуля необходимо дважды щелкнуть левой кнопкой мыши по мастер-модулю в списке. После этого откроется вкладка, на которой расположена информация о режиме работы мастер-модуля и выявленных ошибках, статистическая информация и список системных событий, а также панель управления временем, технологическим приложением и обновлением.

Экспорт < 1/11 > Все события

№	Группа события	Событие
1	Задачи связи	Шлюз (ТСР). Абонент 192.9:200:208 возобновил обмен
2	Задачи связи	Шлюз (ТСР). Отключился абонент 192.9:200:208
3	Задачи связи	Шлюз (ТСР). Подключился абонент 192.9:200:208
4	Связь с модулями	Модуль 24 линия 2. Таймаут связи
5	Связь с модулями	Модуль 24 линия 1. Таймаут связи
6	Связь с модулями	Модуль 24. Нормальный режим работы
7	Связь с модулями	Модуль 23 линия 2. Таймаут связи

Статистика Версия ПО Периферия

Время цикла
Время цикла, мс: 8.2
Максимальное время цикла, мс: 60.9
Фиксированное время цикла, мс: нет
Время выполнения программ, мс: 5.1
Максимальное время выполнения приложения, мс: 8.0

Время
Время на модуле: 11:44:55 06.03.2024
Часовой пояс на устройстве: UTC+0

Режим работы
1 - Исполнение (приложение запущено)

Доп. флаги режима работы:
 RUN/STOP (MODE:6) установлен в STOP
 Технологический (сервисный) режим
 Резервированный режим работы
 Признак резервного контроллера (настройка)
 Текущий статус "В резерве"
 Установлена блокировка изменения статуса
 Установлен запрет загрузки приложения

Флаги ошибок



- Ошибки в работе модулей ввода-вывода
- Ошибки ST-BUS
- Ошибки в работе каналов
- Ошибка резервирования
- Ошибки по одной из линий резервирования
- Ошибка обмена по МК0
- Ошибки по одной из линий МК0
- Ошибки окружения (температура, питание)
- Ошибки энергонезависимого хранения/восстановления
- Ошибки мастера (системные, драйверы, аппаратные)
- Динамические ошибки выполнения приложения

Управление временем и циклом
Установить время
Сброс времени цикла

Управление приложением
Запуск приложения
Останов приложения
Сброс приложения

Обновление прошивок
Обновить ИС
Обновить КК
Обновить ОС

9.1.1 Системные события и статистика

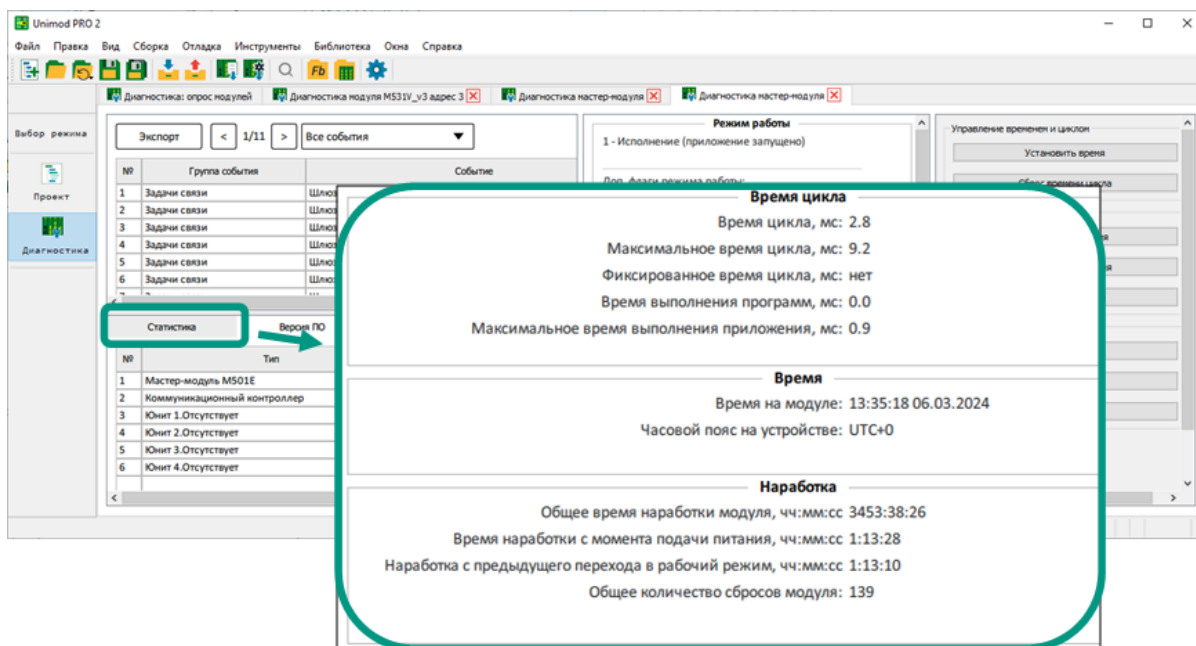
В списке системных событий содержатся записи с указанием краткого описания события, времени возникновения и группы, к которой оно относится. Список выводится постранично, на одной странице выводится не более 100 записей. Для перехода между страницами используйте кнопки  и .

№	Группа события	Событие	Время появления
1	Задачи связи	Шлюз (ТСР). Абонент 192:9:200:208 возобновил обмен	10:35:07 11.03.2024
2	Задачи связи	Шлюз (ТСР). Отключился абонент 192:9:200:208	09:59:15 11.03.2024
3	Задачи связи	Шлюз (ТСР). Абонент 192:9:200:208 возобновил обмен	09:32:30 11.03.2024
4	Задачи связи	Шлюз (ТСР). Отключился абонент 192:9:200:208	09:26:26 11.03.2024
5	Задачи связи	Шлюз (ТСР). Абонент 192:9:200:208 возобновил обмен	09:26:14 11.03.2024
6	Задачи связи	Шлюз (ТСР). Отключился абонент 192:9:200:208	09:25:27 11.03.2024
7	Задачи связи	Шлюз (ТСР). Абонент 192:9:200:208 возобновил обмен	09:25:15 11.03.2024
8	Задачи связи	Шлюз (ТСР). Отключился абонент 192:9:200:208	09:24:57 11.03.2024
9	Задачи связи	Шлюз (ТСР). Абонент 192:9:200:208 возобновил обмен	09:24:19 11.03.2024
10	Задачи связи	Шлюз (ТСР). Отключился абонент 192:9:200:208	08:49:33 11.03.2024
11	Задачи связи	Шлюз (ТСР). Абонент 192:9:200:208 возобновил обмен	08:49:20 11.03.2024

Для вывода записей отдельной группы событий используйте фильтрацию, доступные для выбора группы содержатся в свернутом списке.

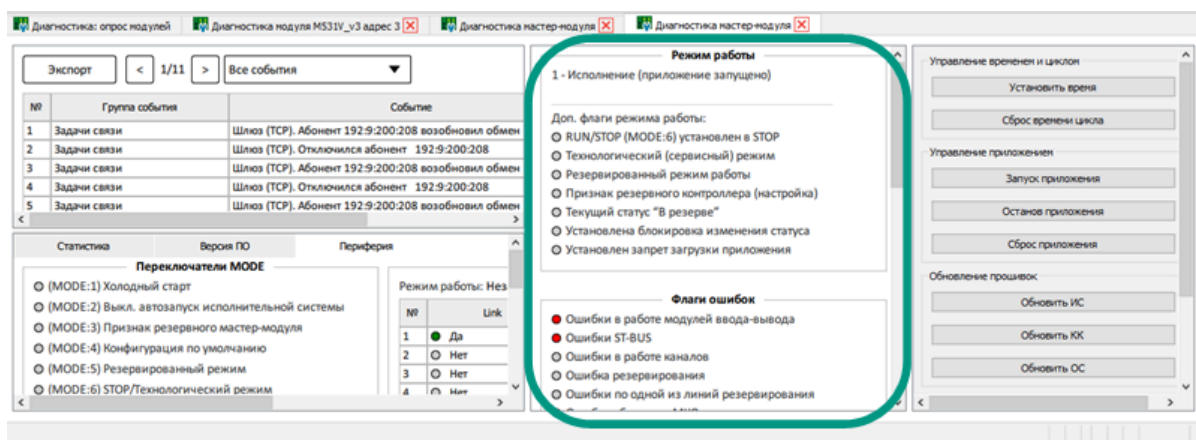
№	Группа события	Событие	Время появления
1	Задачи связи	192:9:201:15	14:20:58 22.02.2024
2	Задачи связи	192:9:201:15	14:20:49 22.02.2024
3	Аппаратная ошибка		14:20:45 22.02.2024
4	Сброс		14:20:45 22.02.2024
5	Восстановление		14:20:45 22.02.2024
6	Сброс		14:20:45 22.02.2024
7	Задачи связи	192:9:201:15	14:20:43 22.02.2024
8	Задачи связи	192:9:201:15	14:20:43 22.02.2024
9	Аппаратная ошибка	LAN 2: кабель не подключен	14:18:24 22.02.2024
10	Сброс	Нормальный режим работы модуля	14:18:24 22.02.2024

Текущие дата, время и часовой пояс мастер-модуля, параметры времени цикла и времени наработки выводятся на вкладке «Статистика».

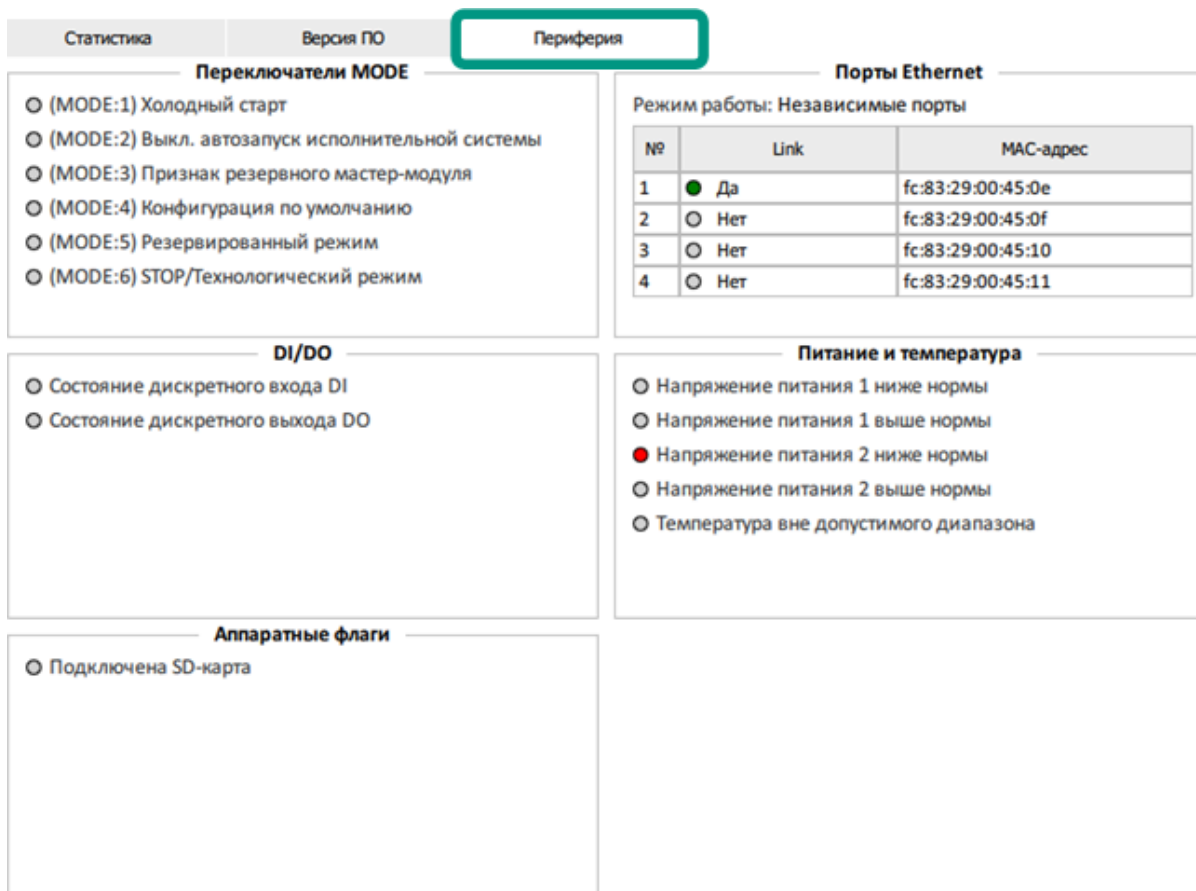


9.1.2 Световые индикаторы

На отдельной панели выводятся индикаторы режима работы, ошибок мастер-модуля и динамических ошибок выполнения приложения.



На вкладке «Периферия» выводятся индикаторы положения MODE переключателей, состояния входных и выходных дискретных каналов, Ethernet-портов, питания и температуры мастер-модуля и индикаторы аппаратуры.



9.1.2.1 Индикаторы режима работы

Режим работы	
1 - Исполнение (приложение запущено)	
Доп. флаги режима работы:	
<input type="radio"/>	RUN/STOP (MODE:6) установлен в STOP
<input type="radio"/>	Технологический (сервисный) режим
<input type="radio"/>	Резервированный режим работы
<input type="radio"/>	Признак резервного контроллера (настройка)
<input type="radio"/>	Текущий статус "В резерве"
<input type="radio"/>	Установлена блокировка изменения статуса
<input type="radio"/>	Установлен запрет загрузки приложения

Индикаторы	Описание
Код режима работы	<p>0 - инициализация;</p> <p>1 - исполнение (приложение запущено);</p> <p>2 - конфигурирование (приложение не запущено – отсутствует или содержит критические ошибки);</p> <p>3 - отладка по циклам;</p> <p>4 - обновление проекта;</p> <p>5 - пробный запуск (при резервировании, попытка работы на новом проекте в течение Watchdog);</p> <p>6 - передача проекта на резервный;</p> <p>7 - откат изменений;</p> <p>8 - запуск проекта на резервном мастере;</p> <p>9 - синхронизация рабочих данных;</p> <p>10 - режим загрузчика.</p>
RUN/STOP (MODE:6) установлен в STOP	Переключатель MODE:6 на корпусе мастер-модуля установлен в положение «On», при этом приложение остановлено, а мастер-модуль переведен в технологический режим.
Технологический (сервисный) режим	Мастер-модуль работает в технологическом режиме, когда запускается только операционная система с поддержкой сетевых интерфейсов. При этом управление технологическим процессом не производится, так как исполнительная система и приложение не выполняются. Данный режим используется для выполнения сервисных работ: безопасной настройки исполнительной системы контроллера, установки и обновления программного обеспечения, копирования файловых данных и диагностики аппаратных средств.
Резервированный режим работы	Мастер-модуль работает в режиме резервирования.

<p>Признак резервного контроллера (настройка)</p>	<p>Мастер-модуль изначально планировалось использовать в качестве резервного. Переключатель MODE:3 на корпусе мастер-модуля установлен в положении «On». Это позволяет запустить его исполнительную систему с задержкой (по умолчанию до 10 секунд), чтобы резервируемый (основной) мастер-модуль успел запуститься раньше и получил статус основного. Текущий мастер-модуль получит статус резервного только после получения уведомления от основного.</p>
<p>Текущий статус «В резерве»</p>	<p>Мастер-модуль используется в качестве резервного.</p>
<p>Установлена блокировка изменения статуса</p>	<p>Изменение статуса мастер-модуля в резервированном режиме (Основной ↔ Резервный) заблокировано. Блокировка может быть вызвана обновлением программного обеспечения или переводом модуля в технологический режим (переключатель MODE:6 на корпусе мастер-модуля установлен в положение «On»).</p>
<p>Установлен запрет загрузки приложения</p>	<p>В одном из программных объектов приложения вызвана системная функция system(52, X), где 52 - код запроса на управление возможностью загрузки приложения, а X - аргумент. Функция позволяет управлять доступом к загрузке приложения. Если аргумент равен 1, то загрузка заблокирована и индикатор загорается красным цветом. Для разблокировки потребуется присвоить аргументу значение 0 или запустить мастер-модуль в технологическом режиме, когда загрузка разрешена, а вызов функции system(52, X) игнорируется.</p>

9.1.2.2 Индикаторы ошибок мастер-модуля

Флаги ошибок

- Ошибки в работе модулей ввода-вывода
- Ошибки ST-BUS
- Ошибки в работе каналов
- Ошибка резервирования
- Ошибки по одной из линий резервирования
- Ошибка обмена по МКО
- Ошибки по одной из линий МКО
- Ошибки окружения (температура, питание)
- Ошибки энергонезависимого хранения/восстановления
- Ошибки мастера (системные, драйверов, аппаратные)
- Динамические ошибки выполнения приложения

Индикаторы	Описание
Ошибки в работе модулей ввода-вывода	Сигнал об ошибке поступает от одного или нескольких модулей ввода-вывода.
Ошибки ST-BUS	Ошибки при обмене с модулями ввода-вывода по шине ST-BUS
Ошибки в работе каналов	Ошибка в работе каналов ввода-вывода, установленных на мастер-модуле.
Ошибка резервирования	Отсутствует соединение с резервным мастер-модулем по всем линиям связи.
Ошибки по одной из линий резервирования	Отсутствует соединение с резервным мастер-модулем по одной из линий связи.
Ошибка обмена по МКО	Ошибка обмена данными в рамках одной или нескольких задачах межконтроллерного обмена.
Ошибки по одной из линий МКО	Ошибка межконтроллерного обмена по одной из 4-х доступных линий (при выходном направлении обмена можно задать до 4-х IP-адресов удаленных устройств).
Ошибки окружения (температура, питание)	Напряжение питания или температура мастер-модуля вышли за пределы заданного диапазона. Максимальные и минимальные допустимые значения устанавливаются для мастер-модуля на заводе-изготовителе, в случае длительной работы модуля за пределами заданного диапазона, возможно возникновение ошибок или сбоев в работе мастер-модуля.

Ошибки энергонезависимого хранения/восстановления	В процессе сохранения значений переменных в энергонезависимой памяти контроллера или их восстановления возникли ошибки.
Ошибки мастера (системные, драйверов, аппаратные)	Выявлены ошибки в работе аппаратной части мастер-модуля или сбои в работе его программного обеспечения.
Динамические ошибки выполнения приложения	Ошибки при выполнении кода одного из программных объектов. Индикация загорается совместно с одним или несколькими индикаторами на панели динамических ошибок.

9.1.2.3 Индикаторы динамических ошибок

Динамические ошибки выполнения приложения

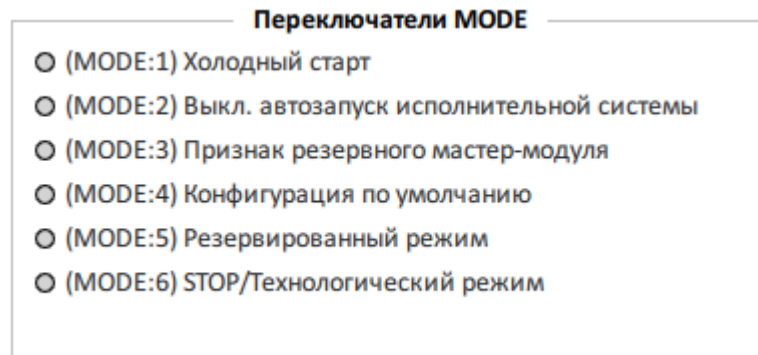
- Целочисленное деление на ноль
- Переполнение при преобразовании FPU в целое
- FPU: Деление на ноль
- FPU: Неверный формат
- FPU: Переполнение в плюс бесконечность
- FPU: Переполнение в минус бесконечность
- Выход за границы массива
- Ошибка при обращении к указателю
- Предотвращено формирование NAN
- Некорректный аргумент (от пользователя)
- Некорректный аргумент (от компилятора)

Индикаторы	Описание
Целочисленное деление на ноль	Предотвращена попытка деления целого числа на ноль. В результате операция деления на ноль и присвоение ее результата проигнорированы, а управление технологическим приложением переходит к следующему оператору.
Переполнение при преобразовании FPU в целое	При преобразовании вещественного числа в целое часть значимых бит не помещается в битовую маску целого числа, что приводит к недостоверному результату операции. После этого приложение продолжает выполняться, но может возникнуть цепочка последующих недостоверных вычислительных операций.
FPU: Деление на ноль	Предотвращена попытка деления вещественного числа на ноль. В результате операция деления на ноль и присвоение ее результата проигнорированы, а управление технологическим приложением переходит к следующему оператору.
FPU: Неверный формат	Несоответствие присвоенного значения формату вещественного числа, операция присвоения будет проигнорирована, управление перейдет к следующему оператору.
FPU: Переполнение в плюс бесконечность	Результат операции превышает максимально допустимое значение для положительных вещественных чисел, в качестве результата присваивается значение NaN. При этом выполнение приложения продолжается, но любая последующая операция с NaN приведет к ошибке и остановке приложения.

FPU: Переполнение в минус бесконечность	Результат операции превышает минимально допустимое значение для отрицательных вещественных чисел, в качестве результата присваивается значение NaN. При этом выполнение приложения продолжается, но любая последующая операция с NaN приведет к ошибке и остановке приложения.
Выход за границы массива	Попытка обращения к несуществующему элементу массива.
Ошибка при обращении к указателю	В настоящее время не используется, зарезервирована под будущие потребности.
Предотвращено формирование NAN	Предотвращена попытка выполнения вычислительной операции, в результате которой может быть присвоено значение NaN. Любая подобная операция игнорируется, а управление переходит к следующему оператору.
Некорректный аргумент (от пользователя)	Предотвращена попытка вызова функции с некорректным аргументом пользователя (несоответствие по типу данных или допустимому значению). В таком случае операция будет пропущена, а управление перейдет к следующему оператору.
Некорректный аргумент (от компилятора)	Ошибка, связанная с результатами компиляции приложения, при возникновении обратитесь к разработчикам.

9.1.2.4 Индикаторы MODE переключателей

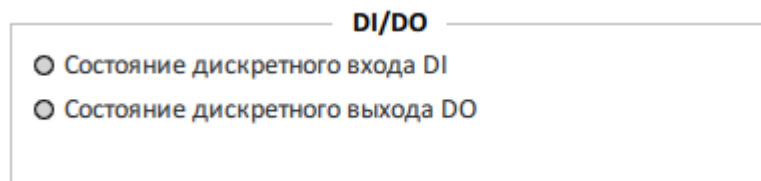
Переключатели MODE размещены на корпусе каждого мастер-модуля и позволяют выбрать требуемый режим работы. Выбор режима работы выполняется до запуска исполнительной системы.



Переключатель	Краткое описание	Описание
MODE:1	Холодный старт	● - переключатель в положении «On». При "холодном" запуске мастер-модуля технологическое приложение начинается выполняться "с нуля", без восстановления сохраненной базы приложения, а модули ввода-вывода сбрасываются. В процессе работы резервирование не производится.
MODE:2	Выкл. автозапуск исполнительной системы	● - переключатель в положении «On» отключает автоматический запуск исполнительной системы, а при включении на мастер-модуле загружается только операционная система с поддержкой сетевых интерфейсов. Данный режим может использоваться для обеспечения безопасного проведения диагностики аппаратных средств.
MODE:3	Признак резервного мастер-модуля	● - переключатель в положении «On» позволяет запустить исполнительную систему мастер-модуля с задержкой (по умолчанию до 10 секунд), чтобы исполнительная система резервируемого (основного) контроллера запустилась раньше и получила статус основного. Мастер-модуль получит статус резервного только после получения уведомления от основного.

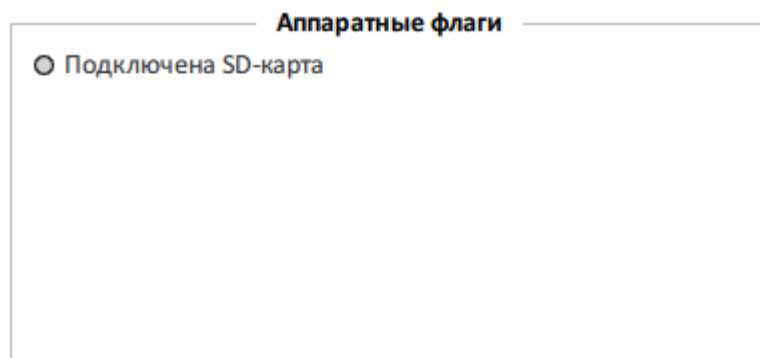
MODE:4	Конфигурация по умолчанию	по	<ul style="list-style-type: none"> ● - переключатель в положении «On» позволяет игнорировать указанные пользователем сетевые настройки, при этом IP-адресам мастер-модуля присваиваются значения по умолчанию, которые могут отличаться в зависимости от типа мастер-модуля и описаны в руководствах по эксплуатации.
MODE:5	Резервированный режим		<ul style="list-style-type: none"> ● - переключатель в положении «On» запускает мастер-модуль в режиме резервирования, когда часть IP-адресов из сетевых настроек автоматически задействуется для передачи резервируемых данных.
MODE:6	STOP/Технологический режим		<ul style="list-style-type: none"> ● - переключатель в положении «On» останавливает выполнение приложения и переводит мастер-модуль в технологический режим.

9.1.2.5 Индикаторы дискретных входов/выходов



Индикатор	Описание
Состояние дискретного входа DI	<input checked="" type="radio"/> - на дискретном входе установлена логическая единица.
Состояние дискретного выхода DO	<input checked="" type="radio"/> - на дискретном выходе установлена логическая единица.

9.1.2.6 Аппаратные индикаторы



Индикатор	Описание
Подключена SD-карта	<input checked="" type="radio"/> - SD-карта подключена к мастер-модулю; <input type="radio"/> - SD-карта отсутствует.

9.1.2.7 Индикаторы Ethernet-портов

№	Link	MAC-адрес
1	<input checked="" type="radio"/> Да	fc:83:29:00:45:0e
2	<input type="radio"/> Нет	fc:83:29:00:45:0f
3	<input type="radio"/> Нет	fc:83:29:00:45:10
4	<input type="radio"/> Нет	fc:83:29:00:45:11

На панели отображаются режим работы Ethernet-портов и их список в виде таблицы, где для каждого порта указаны MAC-адрес и индикатор: ● - кабель подключен, ○ - подключение отсутствует.

9.1.2.8 Индикаторы питания и температуры

Питание и температура	
<input type="radio"/>	Напряжение питания 1 ниже нормы
<input type="radio"/>	Напряжение питания 1 выше нормы
<input checked="" type="radio"/>	Напряжение питания 2 ниже нормы
<input type="radio"/>	Напряжение питания 2 выше нормы
<input type="radio"/>	Температура вне допустимого диапазона

Индикатор	Описание
Напряжение питания 1 ниже нормы	<input checked="" type="radio"/> - напряжение питания на разъеме 1 мастер-модуля опустилось ниже минимально допустимого значения.
Напряжение питания 1 выше нормы	<input checked="" type="radio"/> - напряжение питания на разъеме 1 мастер-модуля превысило максимально допустимое значение.
Напряжение питания 2 ниже нормы	<input checked="" type="radio"/> - напряжение питания на разъеме 2 мастер-модуля опустилось ниже минимально допустимого значения.
Напряжение питания 2 выше нормы	<input checked="" type="radio"/> - напряжение питания на разъеме 2 мастер-модуля превысило максимально допустимое значение.
Температура вне допустимого диапазона	<input checked="" type="radio"/> - температура мастер-модуля вышла за пределы диапазона допустимых значений.

Максимальные и минимальные допустимые значения устанавливаются для мастер-модуля на заводе-изготовителе, в случае длительной работы модуля за пределами заданного диапазона, возможно возникновение ошибок или сбоев в его работе.

9.1.2.9 Коды запуска и сброса

Код запуска приложения	Код сброса
0x00 - Нормальная работа. Ошибок приложения не обнаружено	0x05 - Остановка работы из-за внутренней ошибки приложения

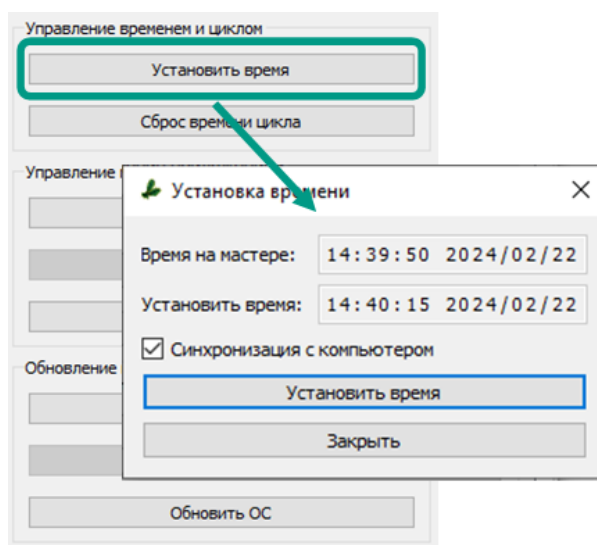
На панелях «Код запуска приложения» и «Код сброса» выводятся коды с кратким описанием состояния выполнения приложения и причин его аварийной остановки.

9.1.3 Управление временем и приложением

Управление временем и циклом.

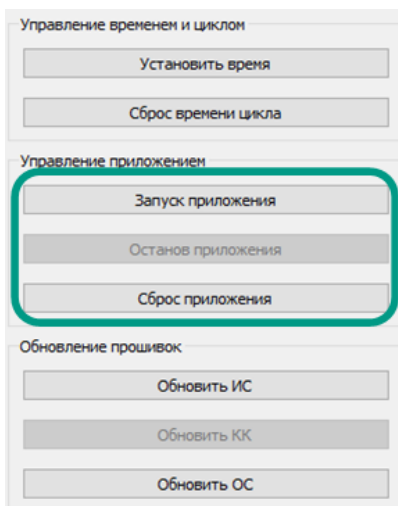
Для изменения текущего времени мастер-модуля нажмите кнопку «*Установить время*» → в открывшемся окне во второе поле введите новые дату и время → нажмите кнопку «*Установить время*». Дата и время вводятся в фиксированном формате. Для синхронизации времени мастер-модуля со временем компьютера выберите пункт «*Синхронизация времени с компьютером*».

Кнопка «*Сброс времени цикла*» позволяет обнулить значение максимального времени цикла.



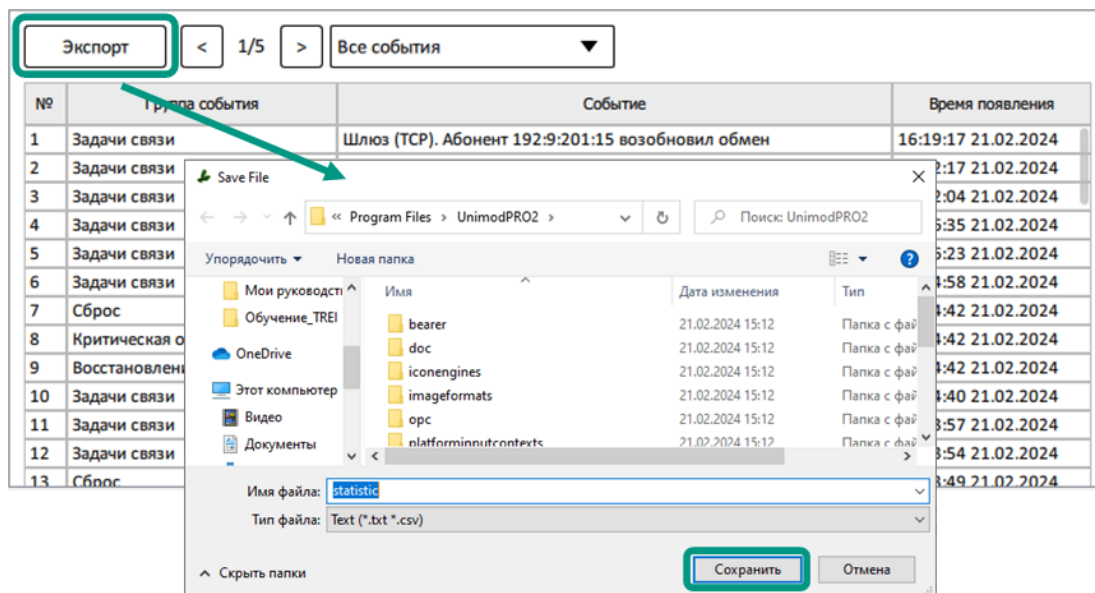
Управление приложением.

Кнопка «*Запуск*» позволяет запустить выполнение технологического приложения. Для остановки приложения используйте кнопку «*Останов приложения*», а для остановки и перезапуска используйте кнопку «*Сброс приложения*».



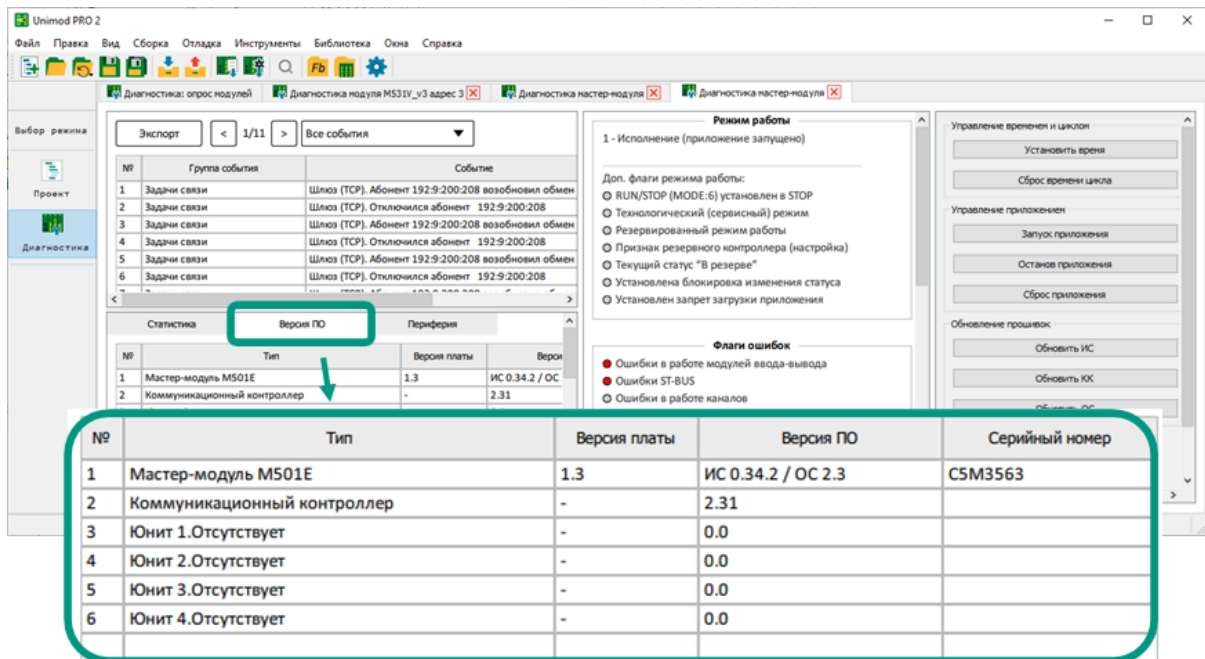
9.1.4 Экспорт данных

Все доступные в режиме диагностики сведения о мастер-модуле и установленном в нем программном обеспечении, значения индикаторов, системные события и накопленная статистика могут быть сохранены в текстовом файле. Для этого нажмите кнопку «Экспорт» → в открывшемся окне выберите каталог для сохранения и введите имя файла → нажмите кнопку «Сохранить».



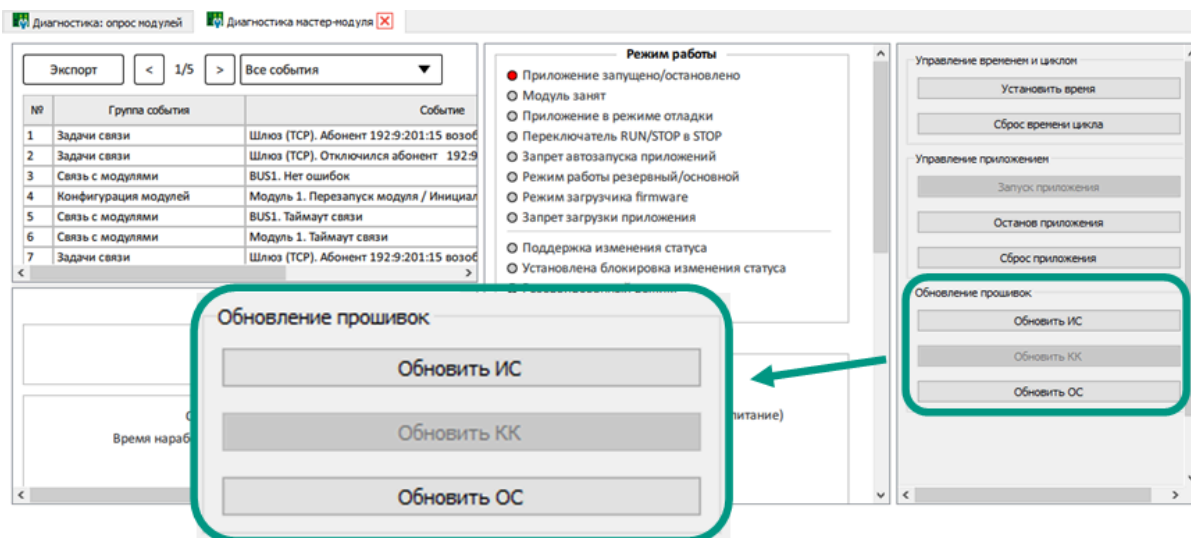
9.1.5 Обновление ПО

Версия установленного на мастер-модуле программного обеспечения (ИС - версия исполнительской системы, ОС - версия операционной системы), а также версия и серийный номер его аппаратной платы выводятся в виде таблицы на вкладке «Версия ПО». Указанные сведения также выводятся для коммуникационного контроллера и подключенных юнитов, при их наличии в составе мастер-модуля.

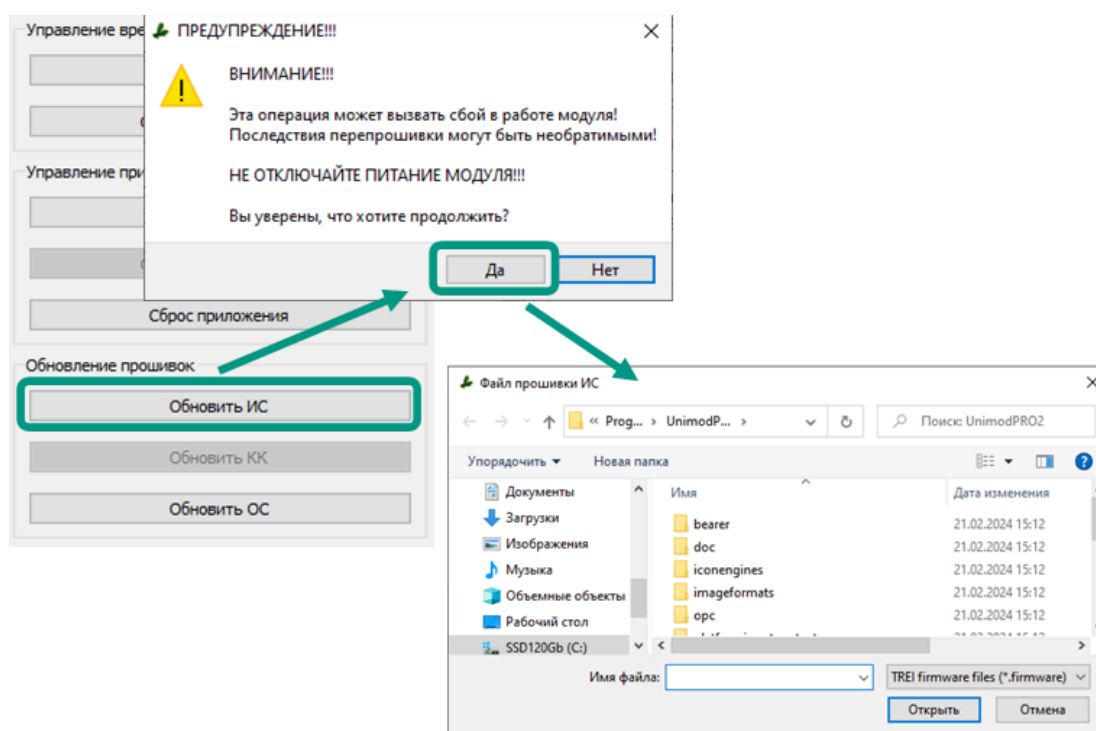


В режиме диагностики на мастер-модуле можно обновить:

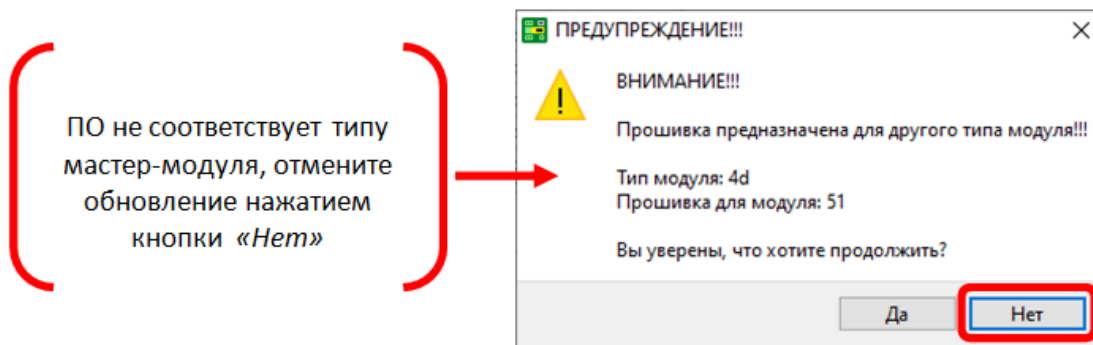
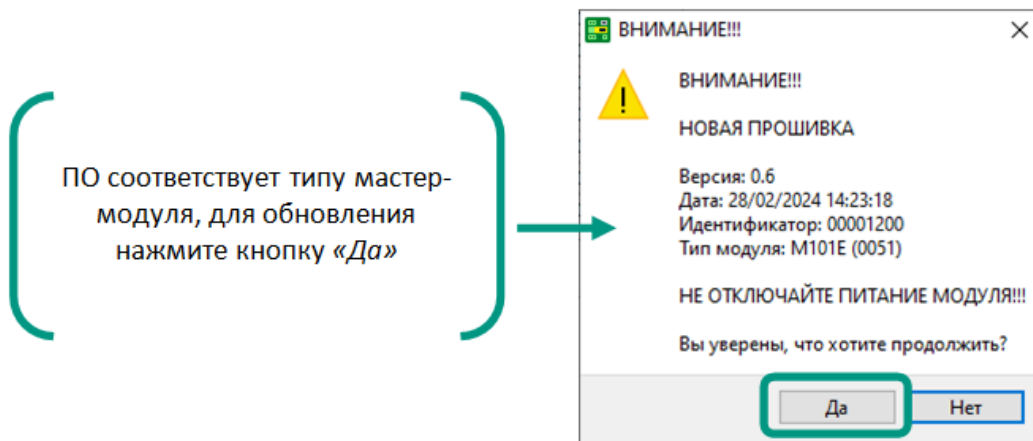
- операционную систему - кнопка «Обновить ОС» (файл обновления в формате «*.osfirmware»);
- исполнительную систему - кнопка «Обновить ИС» (файл обновления в формате «*.firmware»);
- прошивку коммуникационного контроллера - кнопка «Обновить КК» (файл обновления в формате «*.stmfirmware»). Данный контроллер может входить в состав некоторых мастер-модулей и отвечает за обмен данными по шине ST-BUS. Обновление прошивки коммуникационного контроллера доступно только при его наличии в составе мастер-модуля.



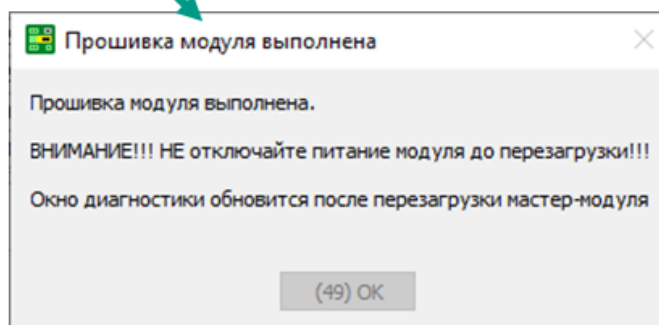
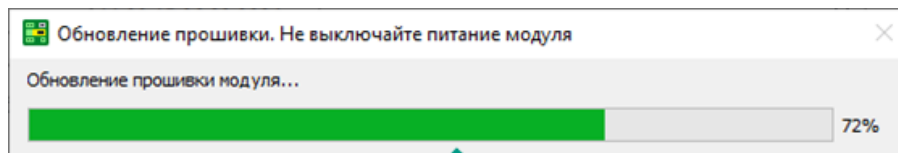
Для обновления нажмите соответствующую кнопку → в диалоговом окне подтвердите обновление нажатием кнопки «Да» → в открывшемся окне выберите файл обновления в соответствующем формате → нажмите кнопку «Открыть». Процессы обновления построены аналогичным образом, отличаются только форматы файлов.



На следующем этапе внимательно ознакомьтесь с сообщением, выведенным в диалоговом окне. Если выбранное программное обеспечение соответствует типу мастер-модуля, и вы убедились в актуальности его версии, запустите обновление нажатием кнопки «Да». При несоответствии выбранного программного обеспечения типу мастер-модуля или иных выведенных ошибках, прекратите обновление нажатием кнопки «Нет».



После этого среда разработки Unimod PRO 2 обновит выбранное программное обеспечение и перезагрузит мастер-модуль. Не выключайте питание мастер-модуля до его полной перезагрузки, время до окончания перезагрузки будет выведено в открывшемся диалоговом окне.



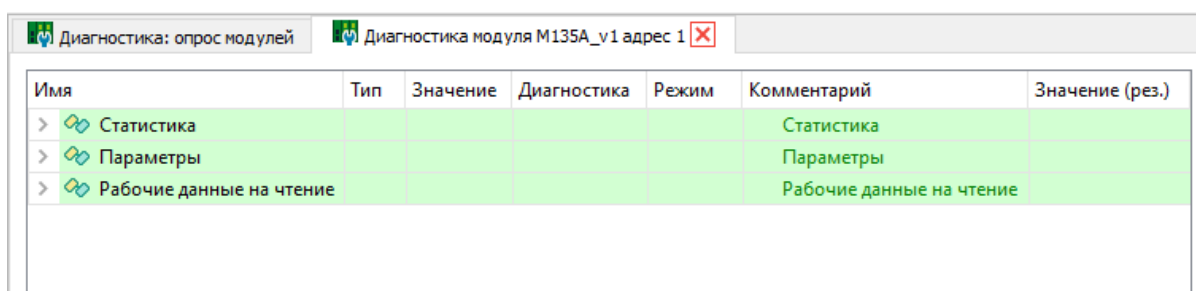
Файлы для обновления программного обеспечения доступны для скачивания с FTP-сервера АО «ТРЭИ». Для открытия доступа на скачивание файлов направьте официальный запрос на электронную почту trpenza@trei.biz.

9.2 Диагностика модулей ввода-вывода

Диагностика модулей ввода-вывода позволяет просматривать статистические данные, параметры работы и состояния каналов ввода-вывода, а также применять инструменты для просмотра калибровочных параметров каналов ввода-вывода и обновления программного обеспечения.

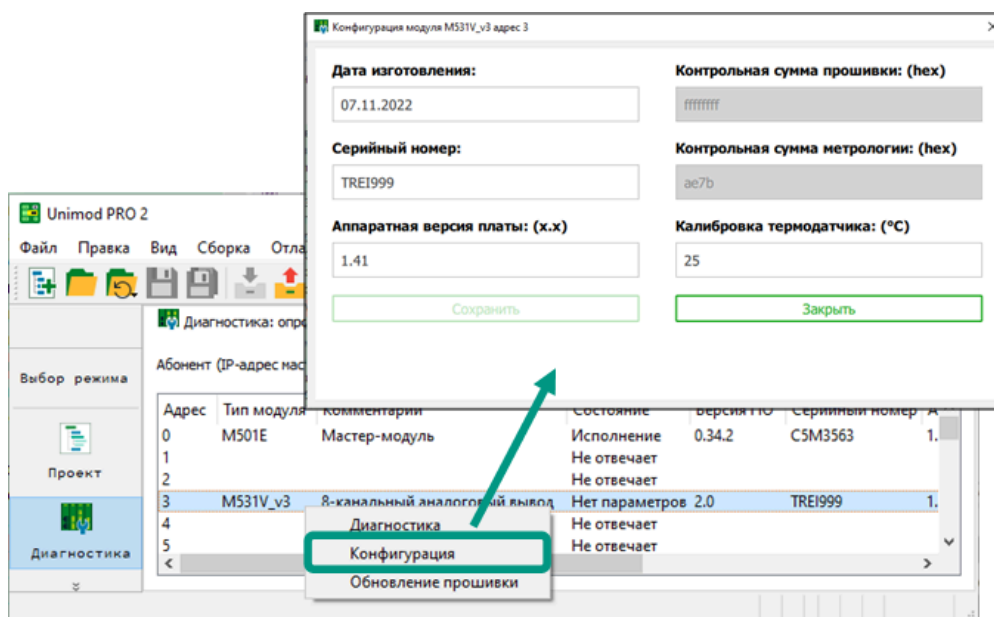
Для выполнения диагностики необходимо дважды щелкнуть левой кнопкой мыши по модулю в списке или из его контекстного меню выбрать пункт «Диагностика». После этого откроется вкладка, на которой расположена модульная структура, соответствующая выбранному модулю ввода-вывода, с информацией о каналах ввода-вывода, параметрах и состоянии устройства. Для просмотра вложенных элементов структуры нажмите символ > у соответствующего раздела.

Состав модульной структуры зависит от выбранной модели модуля ввода-вывода. Подробное описание модульных структур приведено в документе [Руководство по программированию](#) и в руководствах по эксплуатации на конкретное устройство.



Имя	Тип	Значение	Диагностика	Режим	Комментарий	Значение (рез.)
> Статистика					Статистика	
> Параметры					Параметры	
> Рабочие данные на чтение					Рабочие данные на чтение	

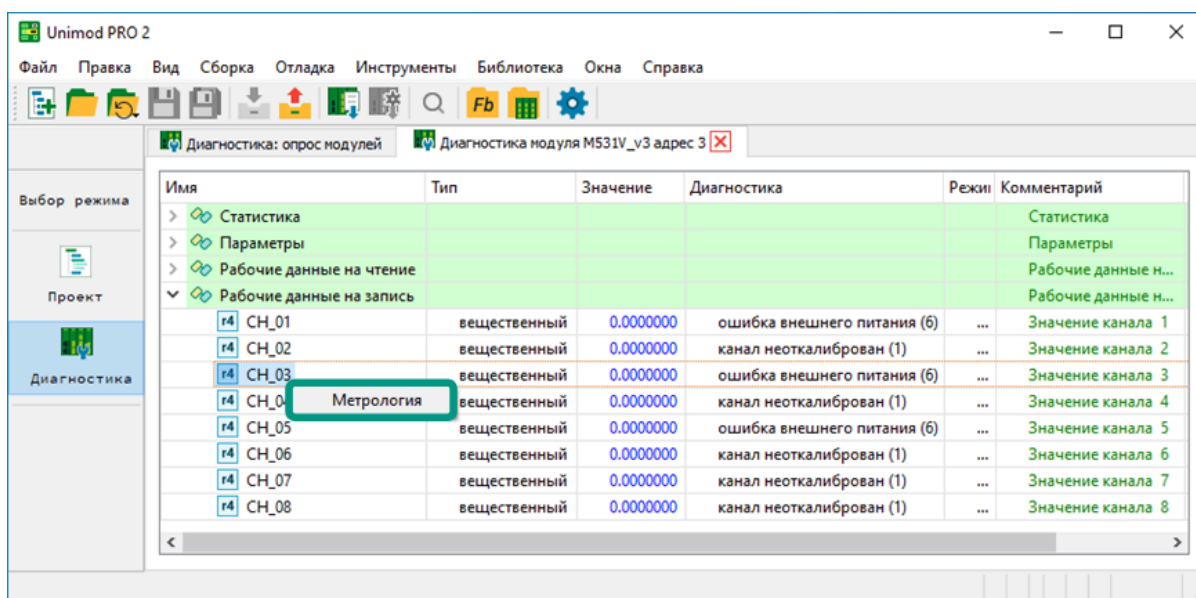
Для просмотра конфигурационных параметров модуля ввода-вывода кликните правой кнопкой мыши по модулю в списке опроса и из контекстного меню выберите пункт «Конфигурация».



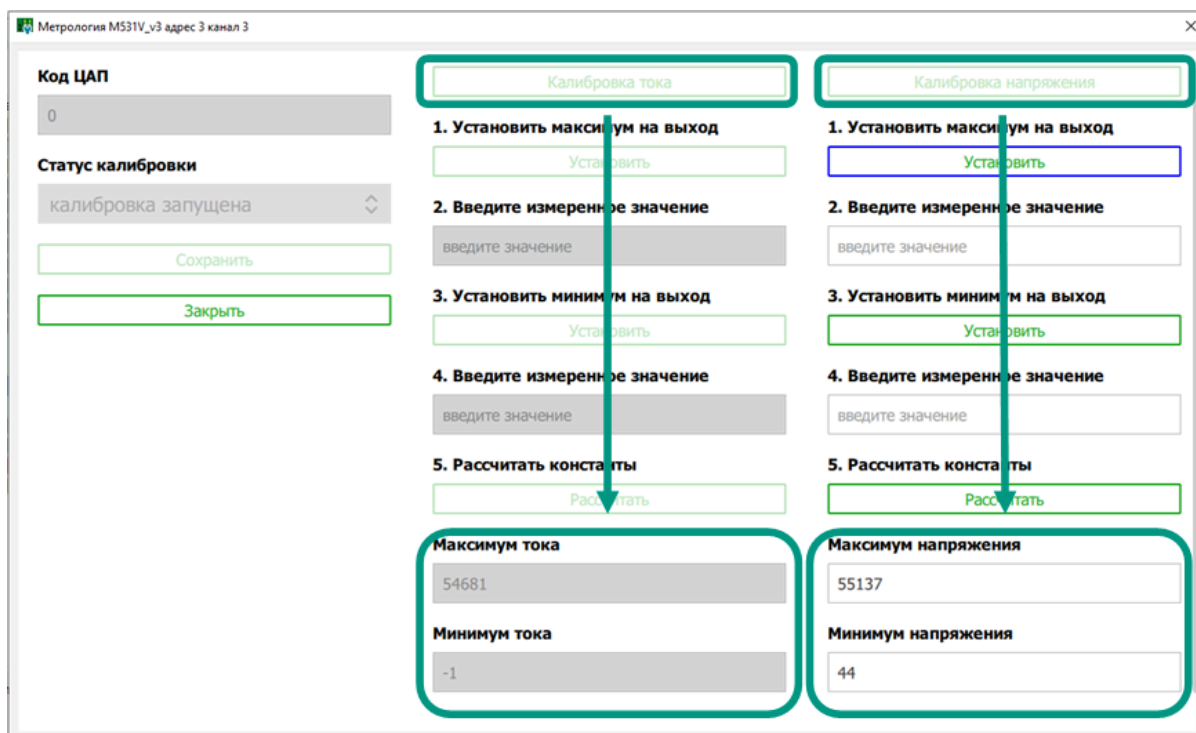
9.2.1 Поверка каналов ввода-вывода

В режиме диагностики доступны для просмотра калибровочные параметры входных и выходных каналов. К таким параметрам относятся минимальное и максимальное значение тока и напряжения на каждом из каналов модуля. Калибровочный диапазон возможных значений тока и напряжения задается на заводе изготовителе и не доступен для изменения пользователем, но может потребоваться в процессе поверки каналов на соответствие характеристикам, заявленным в паспорте на конкретный модуль.

Для просмотра калибровочных параметров на вкладке диагностики модуля ввода-вывода раскройте раздел с рабочими данными на чтение для модулей ввода или данными на запись для модулей вывода → кликните правой кнопкой мыши по одному из каналов → из контекстного меню выберите пункт «Метрология».

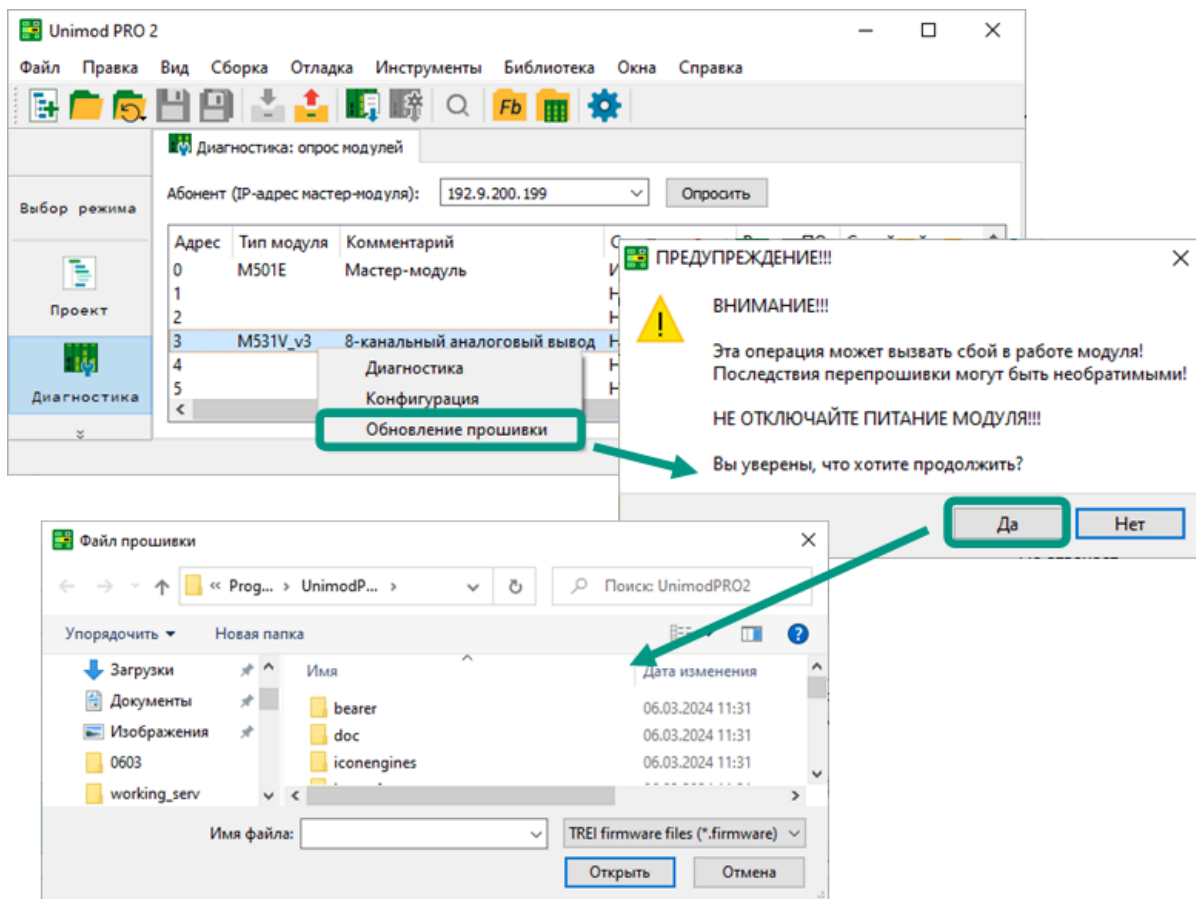


В открывшемся окне для просмотра максимальных и минимальных значений тока и напряжения нажмите на кнопку *Калибровка тока* или *Калибровка напряжения*. Указанные параметры доступны только для чтения, все попытки ввести новые значения будут проигнорированы.

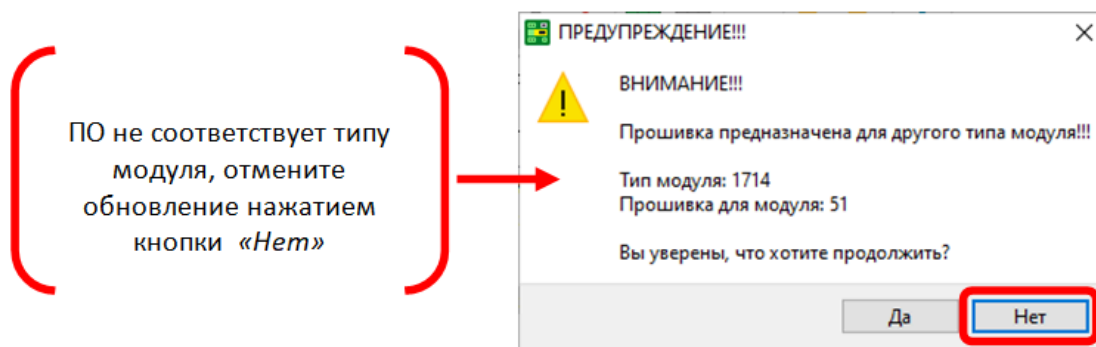
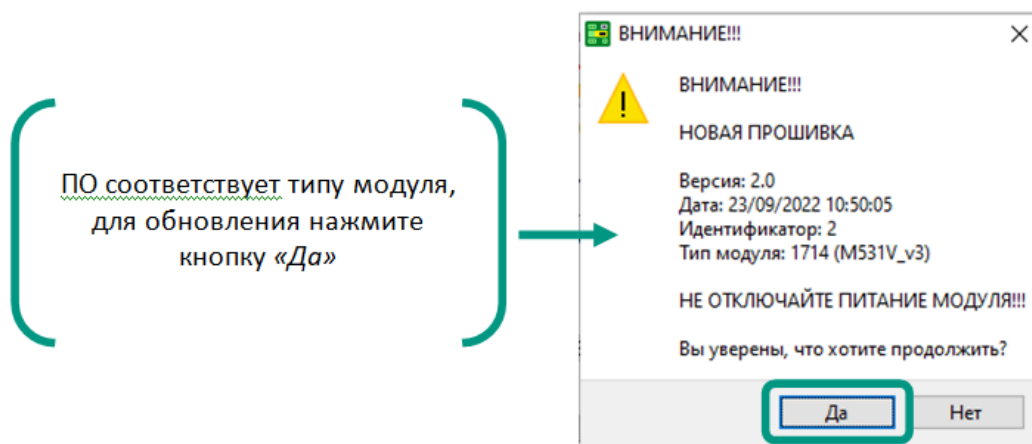


9.2.2 Обновление прошивки

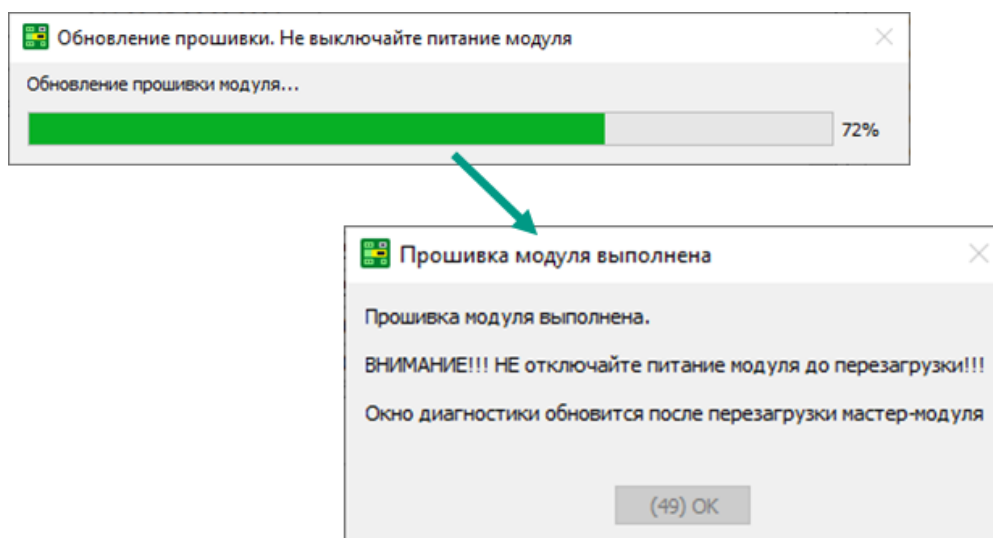
Для обновления прошивки кликните правой кнопкой мыши по модулю ввода-вывода в списке опрошенных и из контекстного меню выберите пункт «Обновление прошивки» → в диалоговом окне подтвердите обновление нажатием кнопки «Да» → в открывшемся окне выберите файл обновления в формате «*.firmware» → нажмите кнопку «Открыть».



На следующем этапе внимательно ознакомьтесь с сообщением, выведенным в диалоговом окне. Если выбранное программное обеспечение соответствует типу модуля ввода-вывода, и вы убедились в актуальности его версии, запустите обновление нажатием кнопки «Да». При несоответствии выбранного программного обеспечения типу модуля или иных выведенных ошибках, прекратите обновление нажатием кнопки «Нет».



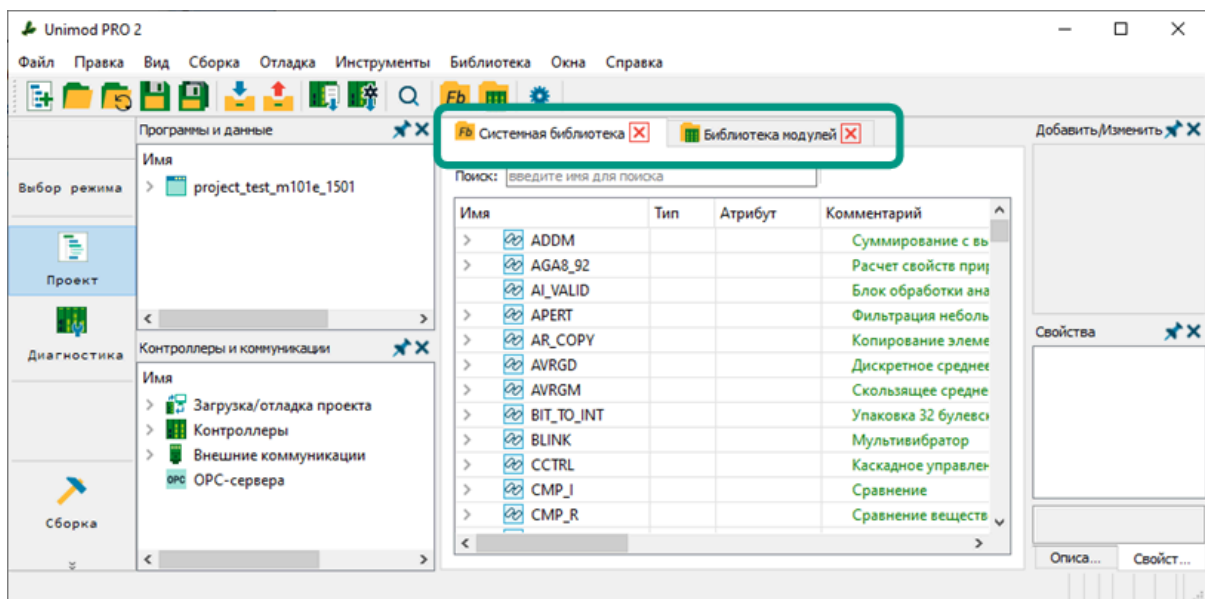
После этого среда разработки Unimod PRO 2 обновит выбранное программное обеспечение и перезагрузит модуль ввода-вывода. Не выключайте питание модуля до его полной перезагрузки, время до окончания перезагрузки будет выведено в открывшемся диалоговом окне.



Файлы с последними версиями прошивок доступны для скачивания с FTP-сервера АО «ТРЭИ». Для открытия доступа на скачивание файлов направьте официальный запрос на электронную почту tr-penza@trei.biz.

10 Библиотеки

Среда разработки Unimod PRO 2 содержит две встроенные библиотеки: системную библиотеку и библиотеку модулей ввода-вывода. Элементы встроенных библиотек периодически обновляются в связи с изменением потребностей пользователей на использование определенного функционала, выпуском новых модулей ввода-вывода или модификацией существующих. Библиотеки в рабочем пространстве среды разработки открываются в виде отдельных вкладок, а содержащиеся в библиотеках элементы доступны только для просмотра. Элементам библиотек и вложенным в них переменным добавлены комментарии с описанием их назначения. Список вложенных элементов раскрывается нажатием на символ >.

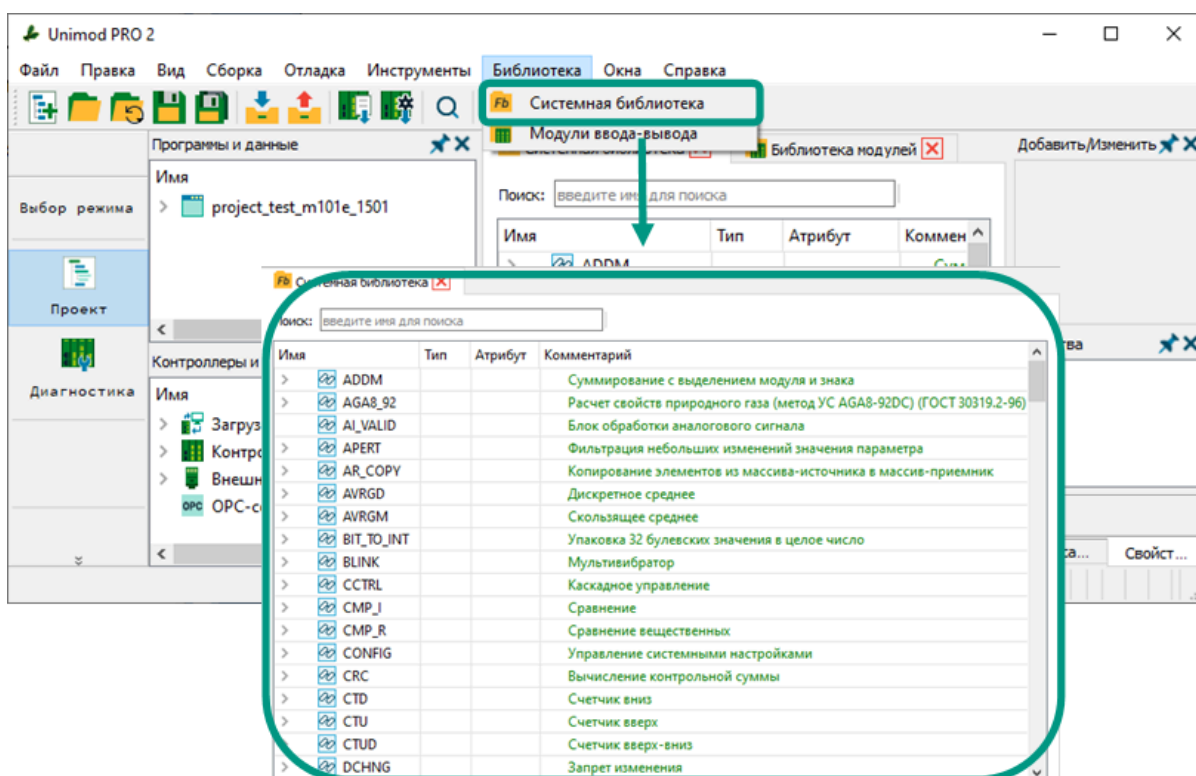


Для поиска элемента библиотеки воспользуйтесь соответствующим полем, расположенным в верхней части вкладки. Поиск осуществляется по названиям элементов посимвольно. Элемент с совпадающим именем будет выделен в списке серым цветом, а при отсутствии совпадений красным цветом будут выделены введенные символы.

10.1 Системная библиотека


В системной библиотеке содержатся готовые для использования функции и функциональные блоки с вложенными входными и выходными переменными. Количество и типы переменных зависят от назначения элемента. Состав системной библиотеки, а также функционал ее отдельных элементов периодически могут изменяться. Полный список элементов системной библиотеки с их подробным описанием приведен в документе [Руководство по программированию](#).

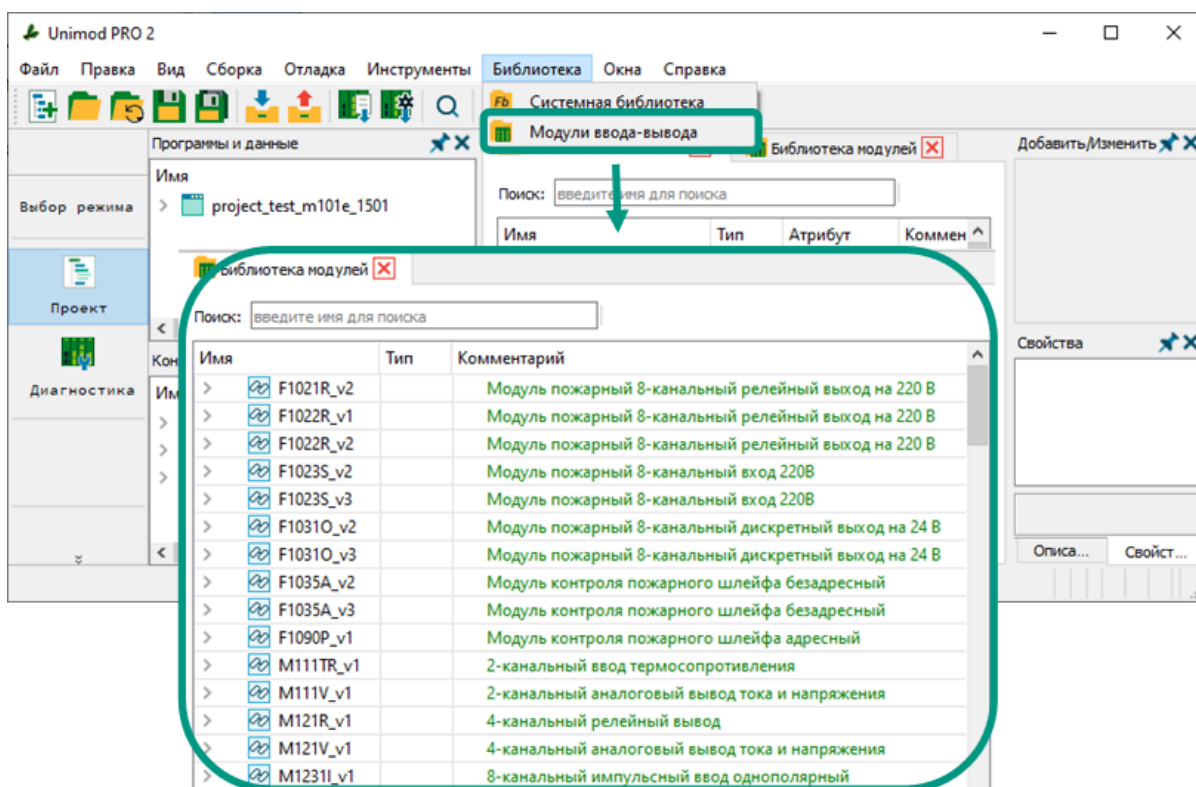
Для открытия системной библиотеки выберите пункт меню «Библиотека → Системная библиотека» или нажмите кнопку **Fb** на панели быстрого доступа.



10.2 Модули ввода-вывода

В библиотеке модулей ввода-вывода содержится список структур, соответствующих модулям ввода-вывода фирмы TREI. В данных структурах предусмотрены переменные для входных и выходных каналов, отслеживания системных параметров и состояния модулей. Названия структур соответствуют названиям физических модулей ввода-вывода. Постфикс «*_vX» в названии типа модуля обозначает версию его аппаратной конфигурации. Список модульных структур, а также их внутренний состав могут изменяться в связи с выпуском новых модулей ввода-вывода или модификацией существующих. Полный список модульных структур с их подробным описанием приведен в документе [Руководство по программированию](#).

Для открытия списка модульных структур выберите пункт меню «Библиотека → Модули ввода-вывода» или нажмите кнопку  на панели быстрого доступа.

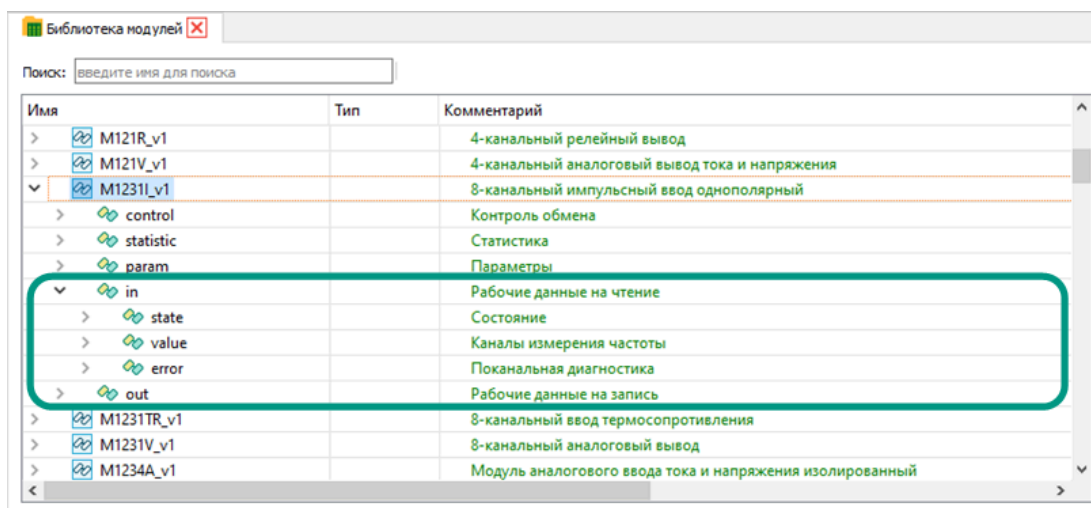


10.2.1 Состав модульных структур

Все переменные модульных структур сгруппированы в логические блоки. Данные блоки можно разделить на три группы: «Рабочие», «Служебные» и «Контроль обмена».

Рабочие данные. Рабочие данные содержатся в блоках: «*in*» - рабочие данные на чтение, «*out*» - рабочие данные на запись, и вложенных в них блоках. В блоке «*in*» содержатся текущие значения каналов ввода-вывода, флаги текущего состояния модуля и поканальная диагностика. В блок «*out*» записываются значения, которые должны быть установлены на каналах ввода-вывода. Чтение и запись рабочих данных выполняется мастер-модулем автоматически в каждом цикле технологического приложения.

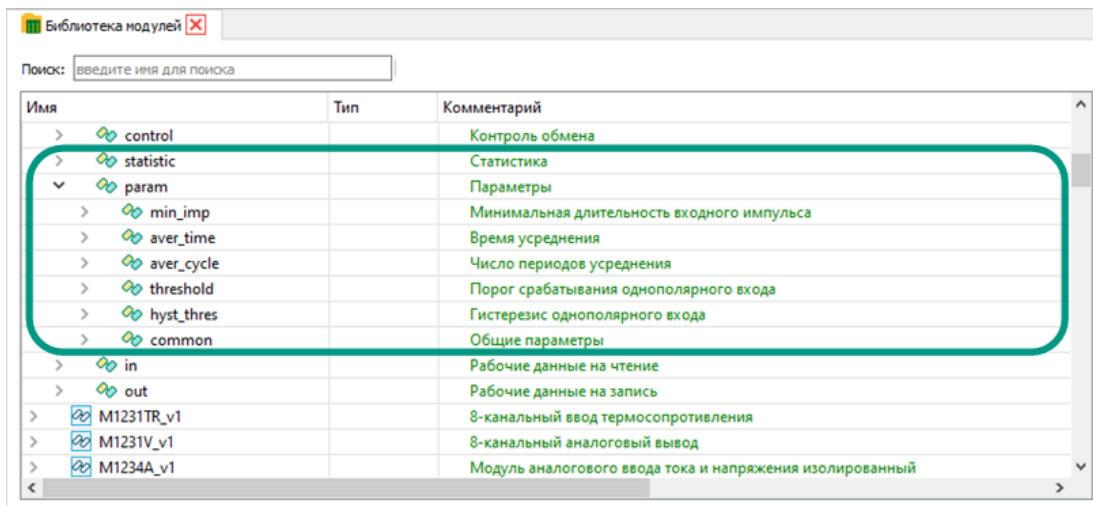
Количество блоков с рабочими данными и их состав может отличаться в зависимости от выбранной модульной структуры. Подробное описание рабочих данных и возможные значения приведены в руководствах по эксплуатации на выбранные модули ввода-вывода и в документе [«Руководство по программированию»](#).



Служебные данные. Служебные данные содержатся в блоках «*statistic*», «*param*» и вложенных в них блоках. В блоке «*statistic*» содержатся статистические данные о работе модуля ввода-вывода, а в блоке «*param*» - настройки модуля ввода-вывода и его отдельных каналов. При этом значения переменных, содержащихся в блоке «*param*» и его подблоках, однократно автоматически записываются на модуль ввода-вывода сразу после старта мастер-модуля. После успешной записи блока «*param*», мастер-модуль переходит в режим обмена рабочими данными с модулем ввода-вывода. Поэтому, для первоначального задания настроек модулю ввода-вывода и его отдельным каналам, необходимо в глобальном словаре для переменной с типом модульной структуры в блоке «*param*» указать требуемые значения. Для изменения настроек модуля ввода-вывода в процессе обмена рабочими данными, необходимо использовать управляющие переменные из группы «*control*». Для уменьшения времени цикла мастер-модуля чтение и запись служебных

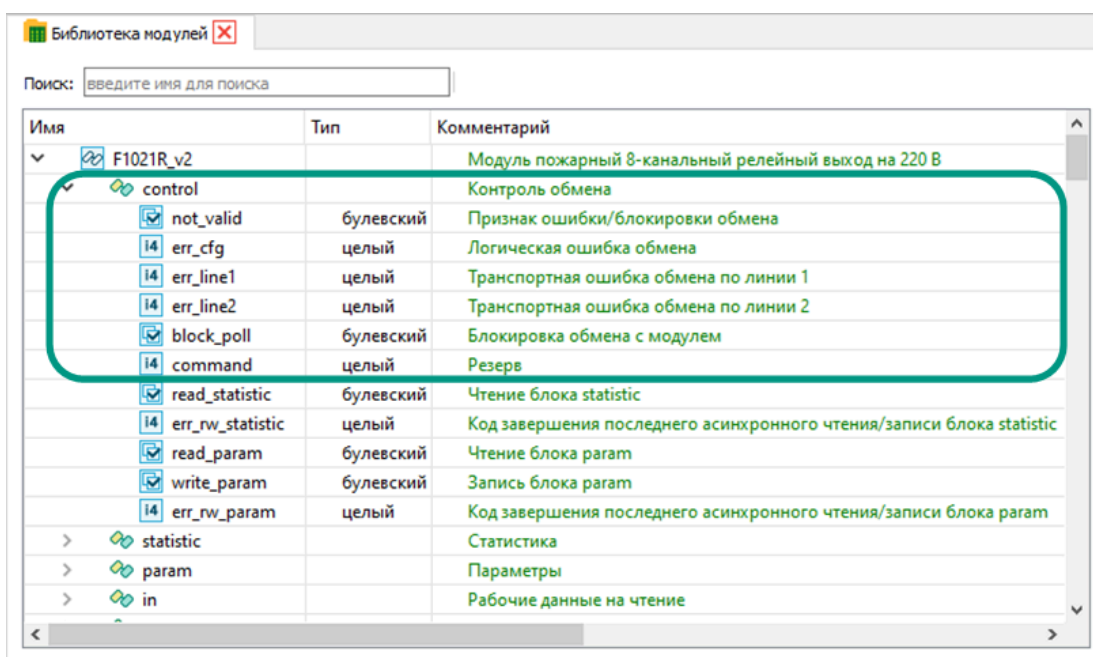
блоков выполняется не в каждом цикле, а по необходимости (с использованием управляющих переменных группы «control»).

Количество блоков со служебными данными и их состав может отличаться в зависимости от выбранной модульной структуры. Подробное описание служебных данных и возможные значения приведены в руководствах по эксплуатации на выбранные модули ввода-вывода и в документе «Руководство по программированию».

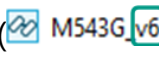


Контроль обмена. Значения переменных из группы «control» одинаковы для всех типов модулей и используются для управления чтением и записью служебных блоков «*statistic*» и «*param*», диагностики ошибок обмена с модулем ввода-вывода и управления блокировкой обмена (переменная «*block_poll*»).

Значения первой части переменных не участвуют в обмене с модулем ввода-вывода, а устанавливаются и анализируются мастер-модулем. Также эти переменные доступны для чтения и записи по внешним коммуникациям.

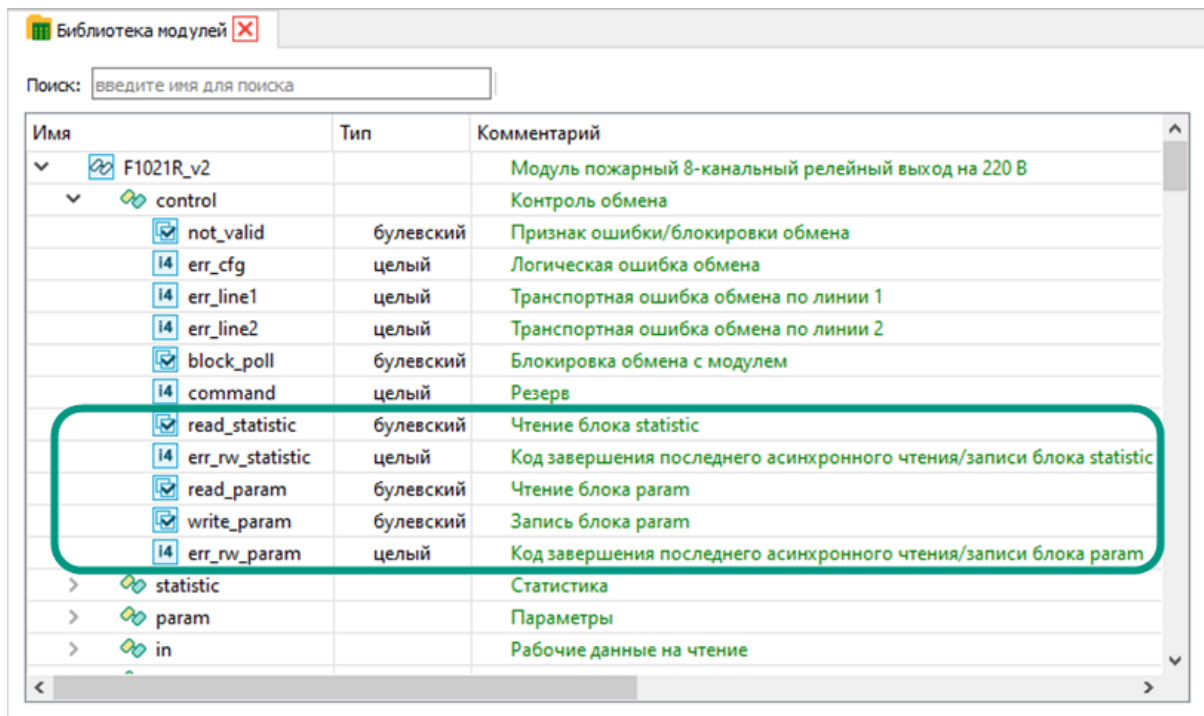


Описание параметров первой части:

Переменная	Тип данных	Описание								
not_valid	Логический	Наличие ошибки обмена с модулем ввода-вывода: <ul style="list-style-type: none"> • true - выявлены ошибки обмена; • false - нет ошибок. Устанавливается в true при наличии ошибок конфигурации модуля ввода-вывода (переменная err_cfg содержит отличное от нуля значение) или при одновременных ошибках на всех линиях связи.								
err_cfg	Целый	Код ошибки обмена с модулем ввода-вывода: <ul style="list-style-type: none"> 0 – нет ошибок; 1 – несоответствие типа подключенного модуля и добавленного в приложении (название модели физического модуля должно совпадать с именем модульной структуры в проекте); 2 – несоответствие версии аппаратной конфигурации подключенного модуля и добавленного в приложении, версия конфигурации указывается в конце имени модульной структуры (); 3 – несоответствие описания аппаратной конфигурации на модуле и в проекте, при возникновении обратитесь к разработчикам; 4 – не удалось определить тип модуля; 5 – ошибка записи блока «param» (блок №3 - Параметры) на модуль; 6 – ошибка чтения уникального идентификатора модуля; 7 – модуль находится в режиме загрузчика (в процессе обновления программного обеспечения). 								
err_line1, err_line2	Целый	Код транспортной ошибки обмена с модулем ввода-вывода по конкретной линии, представляет собой целое число, соответствующее битовой маске: <table border="1" data-bbox="774 1579 1173 1635"> <tr> <td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td> </tr> </table> , где <ul style="list-style-type: none"> 0 бит - нет ошибки, широковещательный запрос; 1 бит - ошибка переполнения UART; 2 бит - ошибка на линии; 3 бит - ошибка адреса (ответ идет с отличным от запроса адресом); 4 бит - ошибочное количество байт данных; 5 бит - повторная стартовая комбинация; 6 бит - ошибка контрольной суммы; 7 бит - срабатывание таймаута при ожидании очередного байта на приеме. 	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1			

block_poll	Логический	Блокировка обмена с модулем ввода-вывода: <ul style="list-style-type: none"> • true - обмен с модулем заблокирован; • false - осуществляется обмен.
command	Целый	Зарезервированный параметр, в настоящее время не используется.

Вторая часть переменных блока «control» позволяет управлять чтением и записью переменных служебных блоков. Состав второй части переменных зависит от типа модуля ввода-вывода и состава его служебных блоков.



Описание параметров второй части:

Переменная	Тип данных	Описание
read_statistic	Логический	Отвечает за чтение с модуля ввода-вывода статистических данных из блока «statistic»: <ul style="list-style-type: none"> • true - чтение статистики в каждом цикле мастер-модуля; • false - чтение не производится (по умолчанию).
read_param	Логический	Отвечает за чтение параметров модуля ввода-вывода из блока «param»: <ul style="list-style-type: none"> • true - чтение параметров в каждом цикле мастер-модуля; • false - чтение параметров не производится (по умолчанию).

write_param	Логический	Запись значений на модуль в служебный блок « <i>param</i> » осуществляется однократно, только при изменении данного значения с false на true. Для новой записи значений в служебные блоки требуется вернуть для данного параметра значение false и снова изменить его на true.
err_rw_statistic	Целый	Код завершения последнего асинхронного чтения/записи блока « <i>statistic</i> »: 0 - безошибочное чтение или запись; 0 - безошибочное чтение или запись; 1 - неверный код команды, команда не поддерживается модулем; 2 - неверные данные в запросе, неверный формат запроса. 3 - выполнение запроса недопустимо в текущем режиме работы модуля; 255 - модуль занят обработкой полученного запроса.
err_rw_param	Целый	Код завершения последнего асинхронного чтения/записи блока « <i>param</i> »: 0 - безошибочное чтение или запись; 1 - неверный код команды, команда не поддерживается модулем; 2 - неверные данные в запросе, неверный формат запроса. 3 - выполнение запроса недопустимо в текущем режиме работы модуля; 255 - модуль занят обработкой полученного запроса.

11 Настройка прав доступа к проекту

Среда разработки Unimod PRO 2 предоставляет возможность ограничивать права доступа к разрабатываемому проекту. Это позволяет предотвратить несанкционированный доступ и изменение проекта, а также разграничить права участников проекта в соответствии с их зонами ответственности и выполняемыми обязанностями. Для ограничения доступа к проекту и отдельным возможностям среды разработки предусмотрены три режима безопасности: открытый, однопользовательский и защищенный.

Открытый. По умолчанию технологический проект создается в открытом режиме, который не содержит никаких ограничений, а проект может быть открыт и изменен любым пользователем, у которого появился к нему доступ.

Однопользовательский. В однопользовательском режиме доступ к проекту ограничивается одним пользователем, которому будут доступны все возможности среды разработки, за исключением настроек безопасности. Этот режим подходит, если пользователь разрабатывает проект один и ему не требуется сложная настройка многопользовательского режима, а достаточно просто защитить проект паролем. В данном режиме сложность, срок действия пароля и прочие параметры учетной записи пользователя не проверяются. Из однопользовательского режима проект можно перевести только в открытый режим.

Защищенный. В защищенном режиме в качестве администратора выступает пользователь, который активировал данный режим. После активации режима потребуется указать свой логин и пароль, которые будут в последующем использоваться для открытия проекта. В проекте у администратора имеется возможность создавать роли пользователей с ограниченными правами на использование возможностей среды разработки, управлять пользователями и назначенными им ролями, настраивать требования к паролям пользователей и периодичности их обновления, а также настраивать параметры сохранения событий системы безопасности. При этом доступ к технологическому проекту у администратора ограничен, он может видеть дерево проекта и дерево конфигурации, открывать формы настройки параметров конфигурации, но не может открывать элементы проекта и вносить в них изменения. У администратора открыт доступ только к диагностическим возможностям.

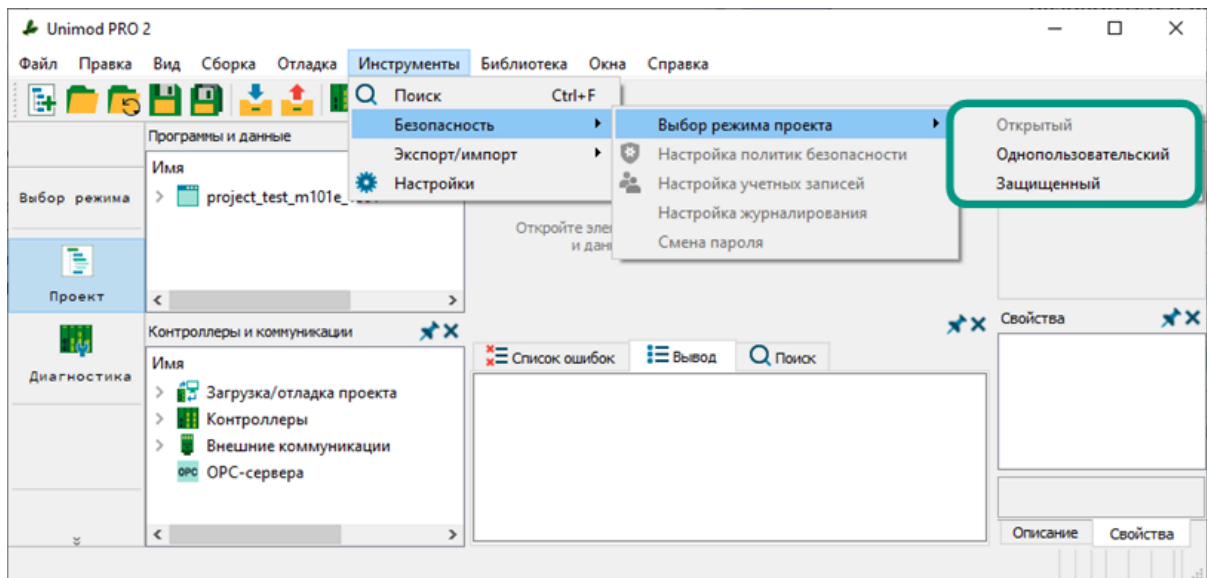
Для доступа к проекту в защищенном режиме администратор каждому новому пользователю выдает логин и пароль, заданный им в настройках безопасности. При первом открытии проекта в среде разработки новый пользователь должен изменить свой пароль, что позволяет предотвратить неправомерный доступ администратора к проекту под чужой учетной записью. Пользователь может сменить свой пароль в любое время работы над проектом либо должен будет это сделать, если администратор выставил ограниченный период для его обновления. В

среде разработки пользователю будут доступны ее возможности, в соответствии с присвоенной ему ролью.

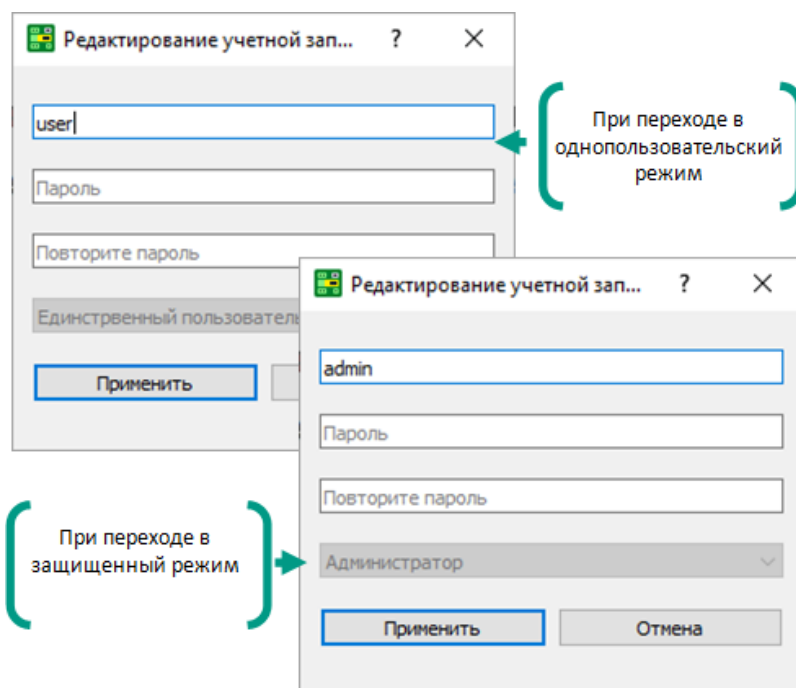
Из защищенного режима проект может быть переведен только в открытый режим и только администратором. При этом данные о пользователях будут удалены, но роли с закрепленными за ними функциями среды разработки останутся сохраненными в проекте и будут доступны при последующем переходе в защищенный режим.

11.1 Изменение режима безопасности

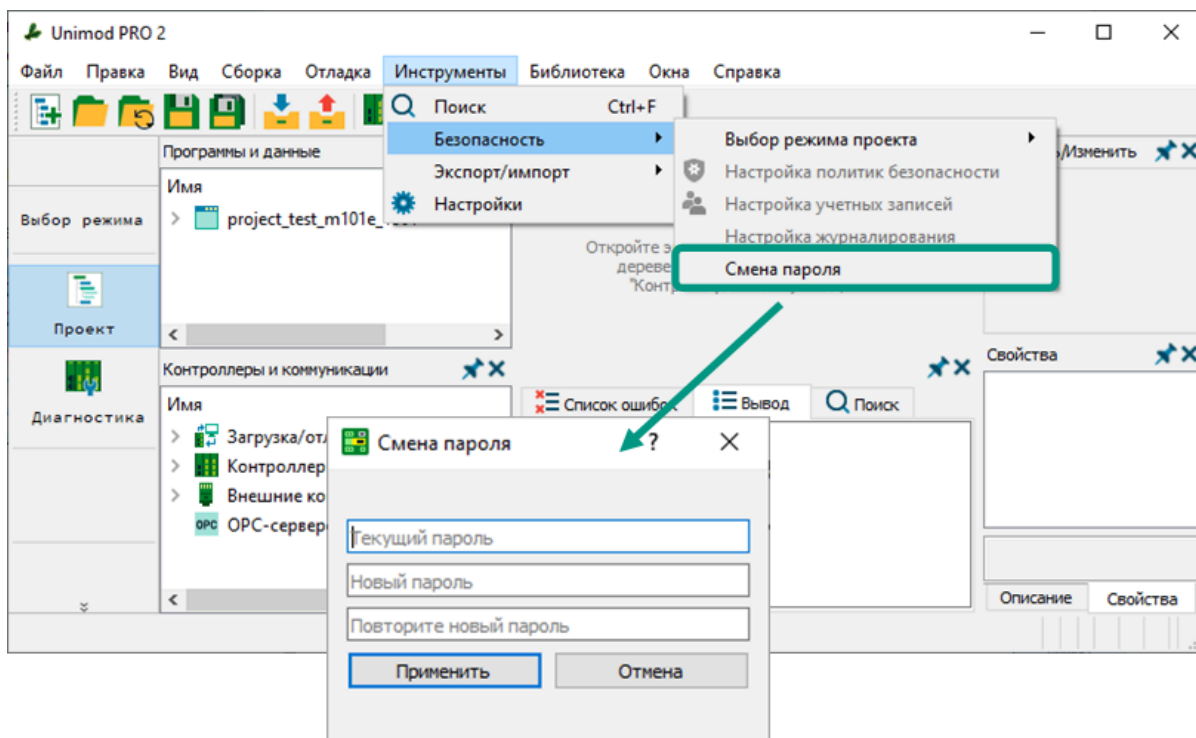
Для изменения режима безопасности перейдите в пункт меню «*Инструменты* → *Безопасность* → *Выбор режима проекта*» и выберите требуемый режим.



При выборе однопользовательского или защищенного режима безопасности в открывшемся диалоговом окне введите свой логин и дважды свой пароль, после чего нажмите кнопку «*Применить*». Режим безопасности будет изменен, а при каждом последующем открытии проекта в среде разработки потребуется вводить свои логин и пароль. По умолчанию пароль должен содержать не менее 8 символов. После изменения режима безопасности рекомендуется закрыть и снова открыть проект.



Для смены пароля выберите пункт меню «Инструменты → Безопасность → Смена пароля», в открывшемся окне введите текущий пароль и дважды новый пароль, после чего нажмите кнопку «Применить».

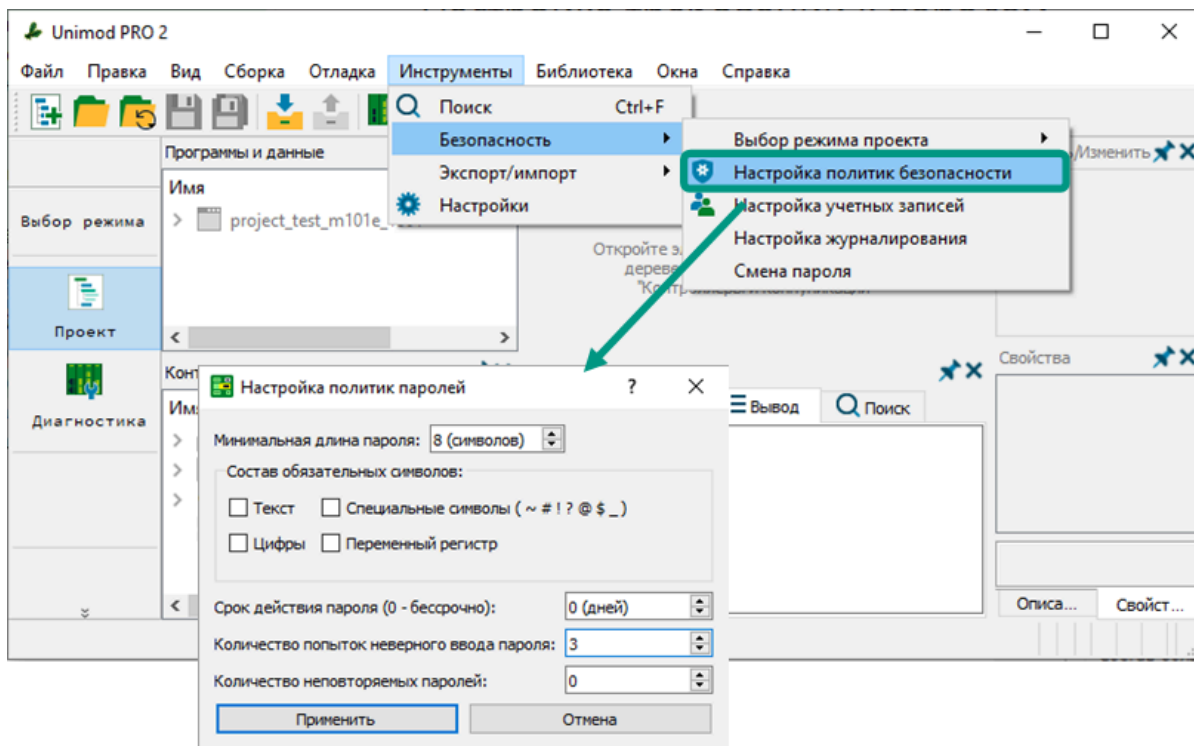


Внимание! Пароль в однопользовательском режиме и пароль администратора в защищенном режиме не восстанавливаются. В случае их утери доступ к приложению будет заблокирован, а для продолжения работы с контроллером потребуется сбросить его до заводских настроек.

11.2 Настройка требований к паролям

Требования к составу пароля, периодичности его изменения и количеству попыток его введения для доступа к проекту могут быть настроены администратором только в защищенном режиме.

Для открытия окна с настройками требований к паролям выберите пункт меню «Инструменты → Безопасность → Настройка политик безопасности».

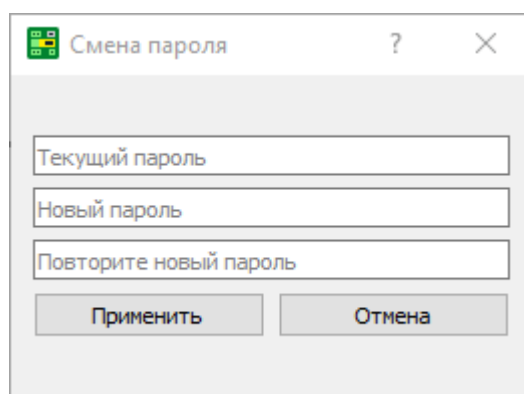


Настраиваемые требования к паролям:

- **минимальная длина пароля** – количество символов в паролях в диапазоне от 8 до 20.
- **состав обязательных символов** – устанавливает требование об обязательном наличии в составе пароля определенной группы символов, что повышает надежность пароля и усложняет его подбор. При выборе параметров «Текст», «Цифра» или «Специальный символ» в пароле должны присутствовать минимум одна буква, цифра или специальный символ соответственно, а при выборе параметра «Переменный регистр» регистр хотя бы одного символа должен отличаться от остальных.
- **срок действия пароля** – количество дней в диапазоне от 0 до 365, по истечении которых пароль должен быть изменен. При нулевом значении срок действия пароля не ограничен.

- **количество попыток неверного ввода пароля** – количество попыток в диапазоне от 3 до 99, при превышении которых доступ к проекту для любого пользователя будет заблокирован. Учетные записи пользователей в защищенном режиме может разблокировать администратор.
- **количество неповторяемых паролей** – настраивает сравнение нового пароля с предыдущими:
 - 0 - сравнение паролей не производится, новый пароль может совпадать с текущим;
 - 1 - новый пароль не должен совпадать с текущим паролем.

Требования к составу пароля распространяются на всех пользователей, включая администратора. При увеличении требований к паролям (увеличение длины, добавлении обязательных символов или переменных регистров) каждому пользователю при следующем открытии проекта потребуется изменить свой пароль в соответствии с новыми требованиями. При уменьшении требований, изменение паролей не требуется.



Смена пароля

Текущий пароль

Новый пароль

Повторите новый пароль

Применить Отмена

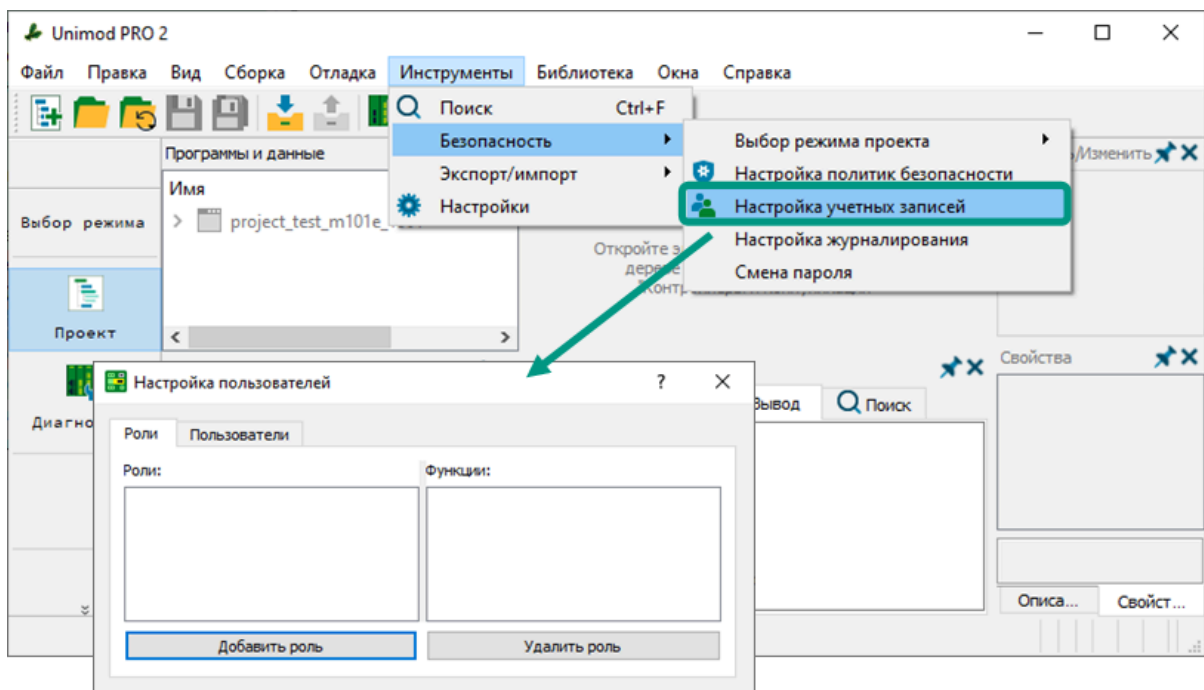


Внимание! При превышении попыток ввода пароля любой пользователь, в том числе и администратор, будет заблокирован, а доступ к приложению будет закрыт. Разблокировать пользователей может только администратор в защищенном режиме. В остальных случаях потребуется создать новое приложение, а контроллер сбросить до заводских настроек.

11.3 Настройка учетных записей

При работе над проектом группы сотрудников, для каждого из них может быть создана отдельная учетная запись с определенной ролью в проекте. Роли позволяют предоставить сотрудникам разрешение на использование отдельных функций среды разработки, которые необходимы при работе над проектом в рамках их должностных обязанностей. Работа с учетными записями пользователей доступна только в защищенном режиме.

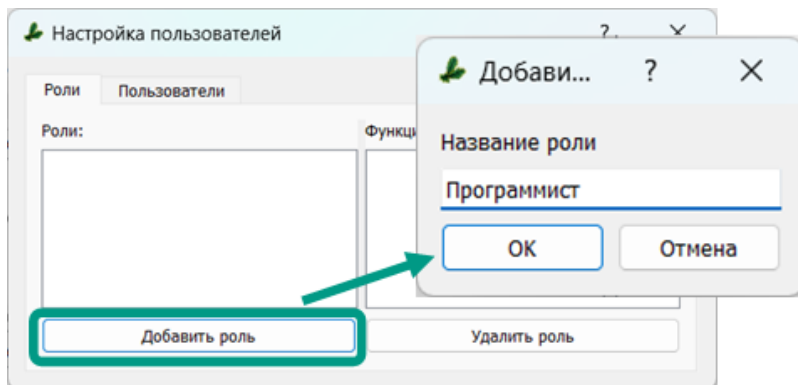
Для настройки учетных записей выберите пункт меню «Инструменты → Безопасность → Настройки учетных записей». В открывшемся окне на вкладке «Роли» в левом столбце отображаются все добавленные в проект роли, а в правом - все доступные для роли функции среды разработки.



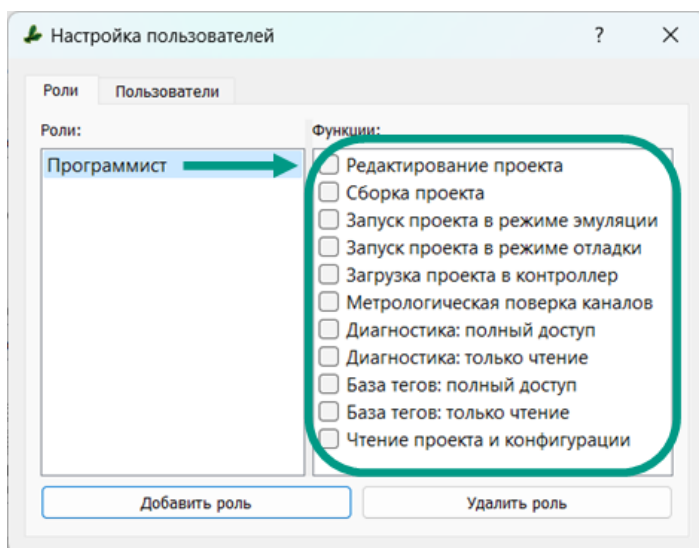
Процесс настройки учетных записей:

1. добавить роли пользователей;
2. выбрать для каждой роли доступные функции среды разработки;
3. добавить пользователей с указанием роли в проекте, логина и первоначального пароля;
4. в процессе работы над проектом при необходимости изменять количество пользователей и ролей, изменять роли пользователей и их пароли, блокировать и разблокировать доступ отдельных пользователей к проекту.

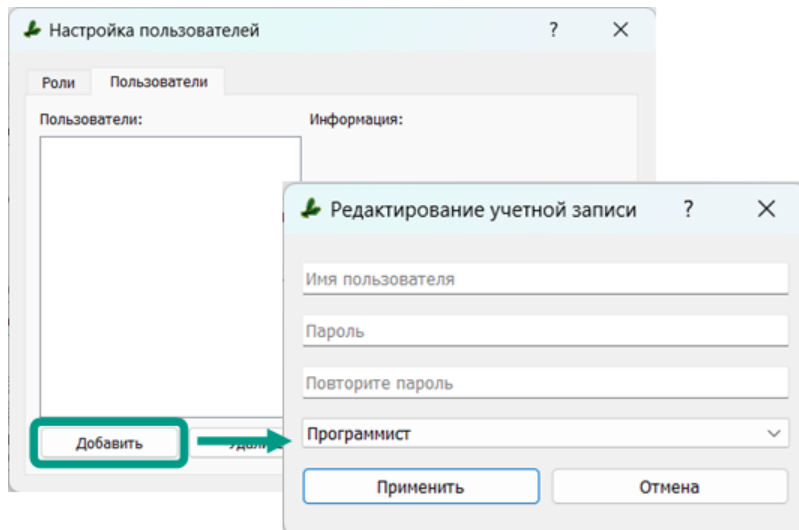
Для добавления роли нажмите кнопку «Добавить роль» → в открывшемся окне введите название роли → нажмите кнопку «Ок».



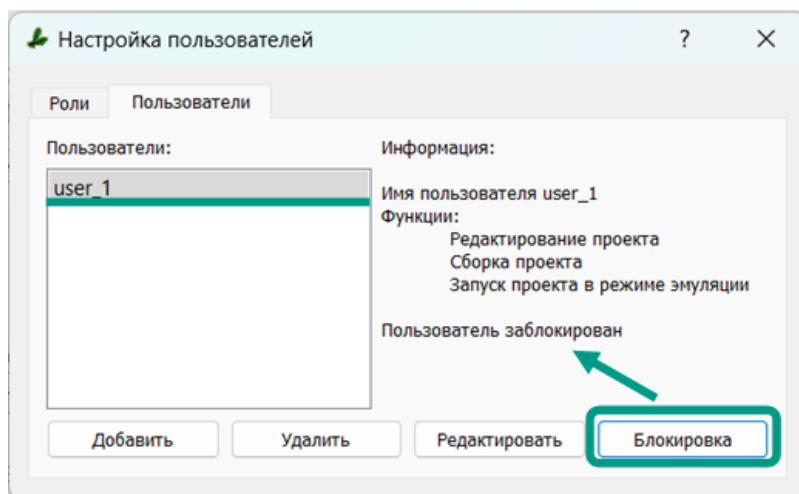
Настройка доступных для роли функций среды разработки осуществляется в правом столбце «Функции». Разрешенная для использования функция помечается галочкой.



Для добавления пользователя перейдите на вкладку «Пользователи» → введите логин и пароль, который будет использоваться пользователем для первого открытия защищенного проекта → выберите одну из ранее созданных ролей → нажмите кнопку «Применить». При этом логин пользователя вводится только при его добавлении и в последующем не может быть изменен. Краткие сведения о пользователе будут выведены в разделе «Информация» при выделении пользователя в списке.



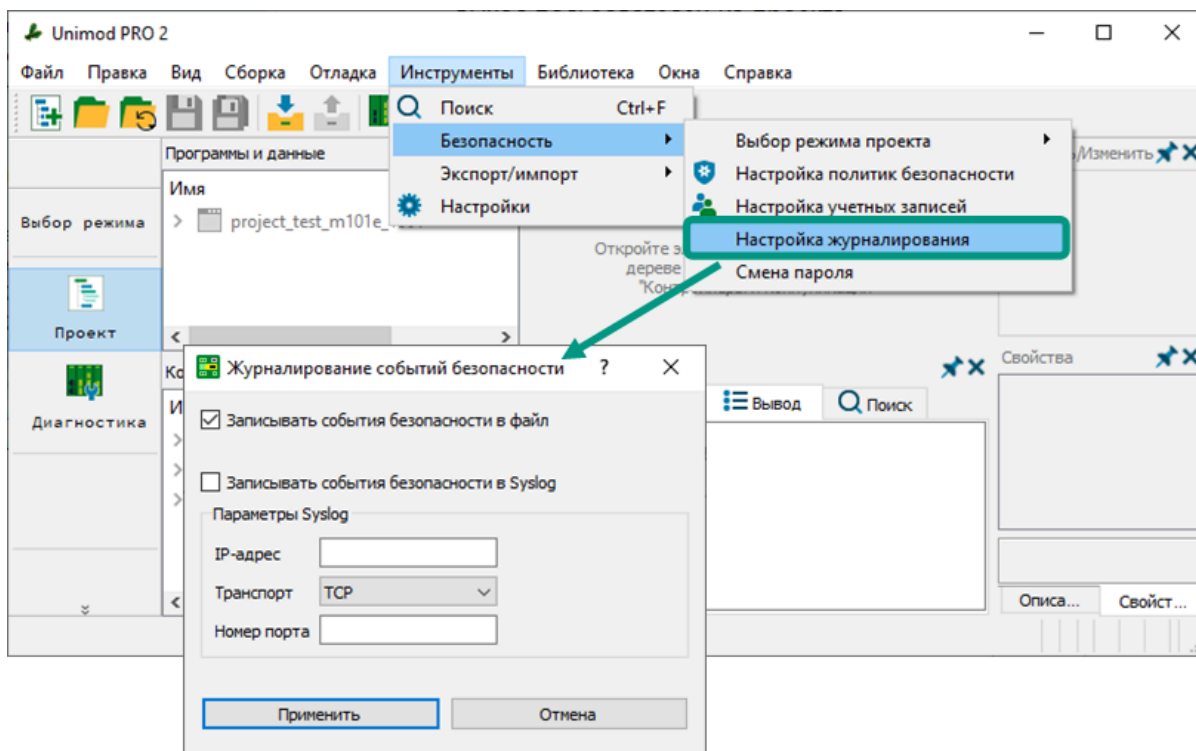
Для блокировки пользователя выделите его в списке и нажмите кнопку «Блокировка». Заблокированный пользователь не имеет доступа к проекту, а при попытке открытия будет выведено уведомление. Блокировка осуществляется администратором или средой разработки при превышении попыток введения неверного пароля. Разблокировать пользователя может только администратор, после выделения пользователя в списке и нажатия кнопки «Блокировка».



Для удаления пользователя или роли выберите их из соответствующего списка и нажмите кнопку «Удалить».

11.4 Регистрация событий безопасности

При работе над проектом среда разработки Unimod PRO 2 позволяет регистрировать добавление и удаление учетных записей, успешные и неуспешные аутентификации пользователей при открытии проекта, вход и выход пользователей. Правила регистрации событий безопасности настраиваются в отдельном окне, для открытия которого выберите пункт меню «*Инструменты* → *Безопасность* → *Настройка журналирования*».



События безопасности могут быть сохранены в локальном текстовом файле с названием «*unimod_siem*», который добавляется в каталог проекта, или на удаленном сервере. Для сохранения событий в локальном файле выберите пункт «*Записывать события безопасности в файл*», а для передачи событий на удаленный сервер выберите пункт «*Записывать события безопасности в Syslog*», введите IP-адрес в формате IPv4, номер порта и протокол передачи данных.

12 История изменений

Версия среды разработки Unimod PRO 2	Описание изменений
2.3.40	Релизная версия