

НАСТРОЙКА ОБМЕНА ДАНЫМИ ПО ПРОТОКОЛУ IEC-104/101 НА КОНТРОЛЛЕРАХ СЕРИИ REGUL RX00

Руководство пользователя

DPA-302.2

Версия ПО 1.6.5.9

Версия 2.4

Март 2022

История изменений руководства пользователя

Версия руководства пользователя	Описание изменения
2.1	<p>Добавлена история изменений руководства пользователя.</p> <p>Добавлены знаки с предупреждающей и поясняющей информацией.</p> <p>Добавлены новые разделы:</p> <ul style="list-style-type: none"> – «Создание списка элементов данных, получаемых по IEC-104» (Slave 104); – «Обращение в службу технической поддержки». <p><i>Раздел «Настройка списка IP-адресов Slave 104 Driver»:</i> дополнено описание возможности настраивать порты вместе с IP-адресами.</p> <p><i>Раздел «Настройка общих параметров устройства Slave 104 Driver»:</i> добавлено описание новых параметров и возможность логирования команды синхронизации.</p> <p><i>Раздел «Создание списка команд, получаемых по IEC-104»:</i> добавлено описание реализации возможности сохранять принятое время в переменной, не выполняя синхронизацию.</p> <p><i>Разделы (Master/Slave) «Привязка переменных программы к элементам данных и командам»:</i> добавлено описание реализации возможности чтение/запись привязанных переменных по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов).</p> <p><i>Раздел (Slave) «Автоматическая генерация переменных и привязка к каналам ввода-вывода»:</i> добавлено описание реализации возможности использования вызова системной функции спорадической отправки из прикладной программы.</p> <p><i>Раздел (Master) «Привязка переменных программы к элементам данных и командам»:</i> добавлено описание реализации возможности установки резервных битов в метке времени.</p> <p>Дополнительно по тексту внесены небольшие изменения с уточняющей информацией</p>
2.2	<p><i>Раздел «Настройка списка IP-адресов Slave 104 Driver»:</i> дополнено описание о возможности задания диапазонов для основных и резервных адресов.</p> <p><i>Раздел «Создание переменных для работы с данными/командами, передаваемыми по IEC 104» (для Slave/Master 104 и Master 101):</i> дополнено описание о возможности получения статуса канала (расширена диагностика канала передачи данных).</p> <p><i>Раздел «Настройка общих параметров устройства Unbalanced Secondary 101 Driver»:</i> добавлено описание нового параметра, определяющего задержку ответа на запрос при работе по RS-485.</p> <p><i>Раздел «Настройка общих параметров устройства IEC 101 Outer Slave»:</i> добавлено описание нового параметра, определяющего количество повторений запросов при возникновении ошибок.</p> <p>Дополнительно по тексту внесены небольшие изменения с уточняющей информацией</p>
2.3	<p><i>Раздел «Настройка общих параметров устройства Slave 104 Driver»:</i> дополнено описание о возможности не копировать сообщения в режиме «Стоп», даже при включенном параметре Использовать накопление очереди сообщений. Журналирование события переполнения очередей спорадических данных.</p> <p>Дополнительно по тексту внесены небольшие изменения с уточняющей информацией</p>

Версия руководства пользователя	Описание изменения
2.4	<p><i>Раздел «Настройка общих параметров устройства Slave 104 Driver»:</i> дополнено описание о возможности удаления очереди сообщений из приложения.</p> <p><i>Раздел «Настройка общих параметров устройства Unbalanced Secondary 101 Driver»:</i> добавлено описание новых параметров (<i>Игнорировать команду time sync; Считать, что время в команде time sync задано в UTC; Мс, которые надо добавлять, учитывая задержку команды TIME_SYNC</i>).</p> <p><i>Раздел «Настройка общих параметров устройства IEC 101 Outer Slave»:</i> добавлено описание нового параметра <i>Мин. интервал между PDU, мс</i></p>

АННОТАЦИЯ

Настоящий документ содержит сведения о настройке на контроллерах серии Regul RX00 передачи данных по стандартам ГОСТ Р МЭК 60870-5-104-2004 и ГОСТ Р МЭК 60870-5-101-2006. Настройка осуществляется с помощью программного обеспечения Epsilon LD.

Данное руководство предназначено для эксплуатационного персонала и инженеров-проектировщиков АСУ ТП, которые должны:

- иметь, как минимум, среднее техническое образование;
- приступить к работе только после изучения данного руководства.


Обновление информации в Руководстве

Производитель ООО «Прософт-Системы» оставляет за собой право изменять информацию в настоящем Руководстве и обязуется публиковать более новые версии с внесенными изменениями. Обновленная версия Руководства доступна для скачивания на официальном сайте Производителя: <https://www.prosoftsystems.ru/>.


Для своевременного отслеживания выхода новой версии Руководства рекомендуется оформить подписку на обновление документа. Для этого необходимо на сайте Производителя: <https://www.prosoftsystems.ru/> во вкладке «Документация» под иконками документов кликнуть на кнопку «Подписаться на обновления» и оставить свои контактные данные.

В руководстве присутствуют знаки с предупреждающей и поясняющей информацией. Каждый знак обозначает следующее:

ПРЕДУПРЕЖДАЮЩИЕ ЗНАКИ

	<p>ВНИМАНИЕ! Здесь следует обратить внимание на способы и приемы, которые необходимо в точности выполнять во избежание ошибок при эксплуатации или настройке.</p>
---	--

ИНФОРМАЦИОННЫЕ ЗНАКИ

	<p>ИНФОРМАЦИЯ Здесь следует обратить внимание на <u>важную</u> информацию</p>
---	--

СОДЕРЖАНИЕ

АННОТАЦИЯ	4
ВВЕДЕНИЕ	7
Общие сведения.....	7
Начало работы	7
НАСТРОЙКА IEC-104.....	10
Настройка контроллера в качестве Slave 104.....	10
Добавление устройства Slave 104 Driver	10
Настройка списка IP-адресов Slave 104 Driver	12
Настройка общих параметров устройства Slave 104 Driver	15
Создание списка элементов данных, передаваемых по IEC-104	19
Создание списка элементов данных, получаемых по IEC-104	22
Создание списка команд, получаемых по IEC-104	23
Создание переменных для работы с данными/командами, передаваемыми по IEC-104	26
Привязка переменных программы к элементам данных и командам IEC-104.....	29
Автоматическая генерация переменных и привязка к каналам ввода-вывода.....	32
Настройка контроллера в качестве Master 104.....	35
Добавление устройства Master 104 Driver.....	35
Настройка общих параметров устройства IEC 104 Outer Slave	36
Создание списка элементов данных, передаваемых по IEC-104	39
Создание списка команд, передаваемых по IEC-104	42
Создание переменных для работы с данными/командами, передаваемыми по IEC-104	44
Привязка переменных программы к элементам данных и командам	45
Автоматическая генерация переменных и привязка к каналам ввода-вывода.....	49
Возможность разрыва соединения со стороны IEC-программы.....	50
НАСТРОЙКА IEC-101.....	52
Добавление порта	52
Настройка параметров порта.....	54
Настройка контроллера в качества Slave 101	55
Добавление устройства Unbalanced Secondary 101 Driver.....	55
Настройка общих параметров устройства Unbalanced Secondary 101 Driver	56
Настройка контроллера в качества Master 101.....	59
Добавление устройства Unbalanced Secondary 101 Driver.....	59
Настройка общих параметров устройства IEC 101 Outer Slave	60

ЭКСПОРТ И ИМПОРТ КОНФИГУРАЦИИ ИЕС-104(101).....	64
Экспорт/импорт списка элементов данных.....	64
Экспорт/импорт списка команд.....	66
ТЕСТИРОВАНИЕ СИСТЕМЫ, ВЫЯВЛЕНИЕ НЕИСПРАВНОСТЕЙ И ИХ УСТРАНЕНИЕ.....	67
Общие действия.....	67
Использование программы ОРС-104 для тестирования	69
ОБРАЩЕНИЕ В СЛУЖБУ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ.....	74

ВВЕДЕНИЕ

Общие сведения

Контроллеры серии Regul RX00 подключаются к сетям передачи данных посредством интерфейса RS-232/485 и/или Ethernet. Каждый интерфейс поддерживает стандартный протокол обмена данными: Ethernet - ГОСТ Р МЭК 60870-5-104-2004 (далее по тексту IEC-104) и RS - 232/485 - ГОСТ Р МЭК 60870-5-101-2006 (далее по тексту IEC-101). Протокол IEC-101 поддерживает только небалансную передачу данных в локальной сети.

Стандарты подробно описаны в документе «Устройства и системы телемеханики. Часть 5. Протоколы передачи» (см. раздел 101 и 104). Протокол IEC-104 является модификацией протокола IEC-101, реализует и поддерживая набор функций протокола IEC - 101 (прикладной уровень), но с расширенными возможностями. Протокол IEC-104 ориентирован на передачу данных по TCP/IP (транспортный уровень). Оба протокола располагают схожими параметрами.

Программное обеспечение контроллера Regul позволяет сконфигурировать его как в качестве *Master*-устройства, так и в качестве *Slave*-устройства.

Перечень рекомендуемых документов

Для получения информации по настройке других параметров контроллеров серии Regul RX00 в среде разработки Epsilon LD рекомендуется ознакомиться со следующими документами:

- Программное обеспечение Epsilon LD. Руководство пользователя;
- Regul R600. Системное руководство;
- Regul R500. Системное руководство;
- Regul R400. Системное руководство;
- Regul R200. Системное руководство.

Начало работы

Установите на компьютер программное обеспечение **Epsilon LD**. Описание процесса установки программы, а также инструкции по работе с программой приведены в документе «Программное обеспечение Epsilon LD. Руководство пользователя». Программа установки и документация доступны на сайте <http://www.prosoftsystems.ru/catalog/show/programmnoe-obespechenie-epsilon-ld>.

Запустите программу **Epsilon LD**. Создайте проект в соответствии с аппаратной конфигурацией контроллера с помощью Мастера конфигурации Regul (инструкции приведены в документе «Программное обеспечение Epsilon LD. Руководство пользователя»).

В случае, когда контроллер Regul RX00 выступает в качестве Slave, для настройки обмена данными нужно выполнить следующие действия согласно таблице 1.

Таблица 1 – Настройка контроллера в качестве Slave устройства

Slave 104 Driver	Slave 101 Driver
<p>Последовательность действий:</p> <ul style="list-style-type: none"> – добавить к контроллеру устройство Slave 104 Driver – драйвер, который принимает и обрабатывает входящие соединения от IEC-104 клиентов; – указать IP-адреса, от которых контроллеру разрешается принимать входящие запросы; – настроить общие параметры устройства Slave 104 Driver, отвечающие за то, как будет происходить обмен данными; – создать список элементов данных и список команд, передаваемых по протоколу IEC-104; – создать в программном коде переменные, представляющие из себя функциональные блоки определенных типов, описанных в библиотеке PsIecCommon. После чего связать переменные программы с элементами данных и командами. Предусмотрена возможность автоматической генерации переменных и их привязка. 	<p>Последовательность действий:</p> <ul style="list-style-type: none"> – добавить к контроллеру последовательный порт – Regul Serial Port; – настроить параметры последовательного порта; – добавить устройство Unbalanced Secondary 101 Driver – драйвер, который взаимодействует с сервером IEC-101; – настроить общие параметры устройства Unbalanced Secondary 101 Driver, который открыт для приема входящих соединений; – создать список элементов данных и список команд, передаваемых по протоколу IEC-101; – создать в программном коде переменные, представляющие из себя функциональные блоки определенных типов, описанных в библиотеке PsIecCommon. После чего связать переменные программы с элементами данных и командами. Предусмотрена возможность автоматической генерации переменных и их привязка.

В случае, когда контроллер Regul RX00 выступает в качестве Master, для настройки обмена данными нужно выполнить следующие действия согласно таблице 2

Таблица 2 – Настройка контроллера в качестве Master

Master 104 Driver	Master 101 Driver
<p>Последовательность действий:</p> <ul style="list-style-type: none"> – добавить к контроллеру устройство Master 104 Driver – драйвер, который взаимодействует с сервером IEC-104. Далее к драйверу добавить IEC 104 Outer Slave – устройство, настроенное как подчиненное устройство для ведущего устройства IEC-104; – настроить общие параметры устройства IEC 104 Outer Slave, в том числе номер порта TCP, который открыт данным slave 	<p>Последовательность действий:</p> <ul style="list-style-type: none"> – добавить к контроллеру последовательный порт – Regul Serial Port; – настроить параметры последовательного порта; – добавить устройство Unbalanced Primary 101 Driver – драйвер, который принимает и обрабатывает входящие соединения от IEC-101 клиентов; – добавить и настроить общие параметры

Master 104 Driver	Master 101 Driver
<p>для приема входящих соединений;</p> <ul style="list-style-type: none">– создать список элементов данных и список команд, передаваемых по протоколу IEC-104;– создать в программном коде переменные, представляющие из себя функциональные блоки определенных типов, описанных в библиотеке PsIecCommon. После чего связать переменные программы с элементами данных и командами. Предусмотрена возможность автоматической генерации переменных и их привязка.	<p>устройства IEC 101 Outer Slave, отвечающие за то, как будет происходить обмен данными;</p> <ul style="list-style-type: none">– создать список элементов данных и список команд, передаваемых по протоколу IEC-101;– создать в программном коде переменные, представляющие из себя функциональные блоки определенных типов, описанных в библиотеке PsIecCommon. После чего связать переменные программы с элементами данных и командами. Предусмотрена возможность автоматической генерации переменных и их привязка.

НАСТРОЙКА IEC-104

Настройка контроллера в качестве Slave 104

Контроллер выступает в роли ведомого, опрашиваемого устройства.

Добавление устройства Slave 104 Driver

В окне дерева устройств поместите курсор на название контроллера, нажмите правую кнопку мыши. В появившемся контекстном меню выберите пункт **Добавить устройство...** Откроется окно **Добавить устройство**, в котором выберите *Regul* → *МЭК 60870* → *TCP 60870-104-Slave* → *Slave 104 Driver*. Нажмите кнопку **Добавить устройство** или дважды щелкните левой кнопкой мыши. Выбранное устройство появится в проекте в дереве устройств (Рисунок 1).

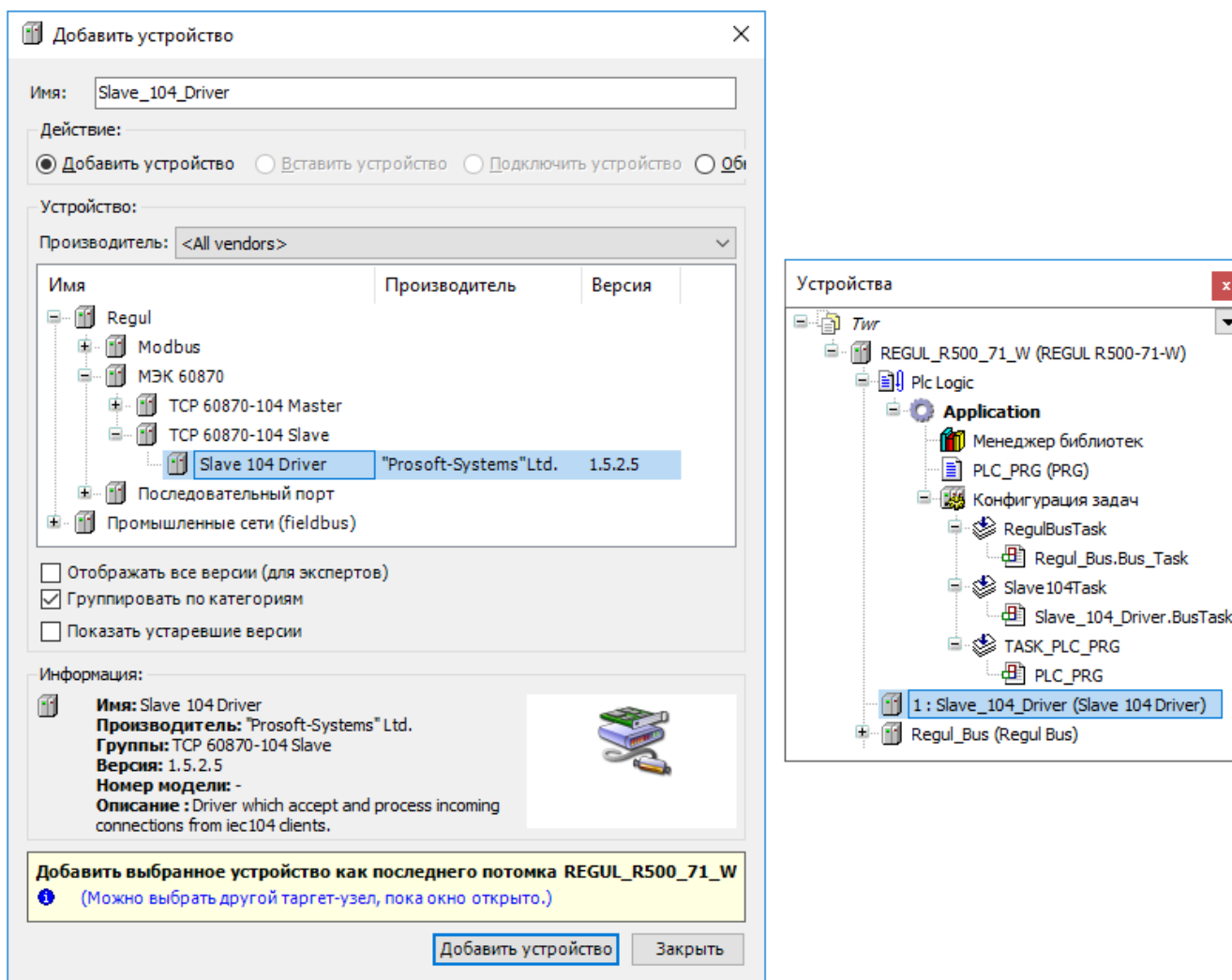


Рисунок 1 – Добавление в конфигурацию контроллера устройство Slave 104 Driver

Двойным щелчком по названию устройства **Slave 104 Driver** откройте вкладку параметров. По умолчанию открывается первая внутренняя вкладка **Редактор устройства IEC 104 Slave** (Рисунок 2).

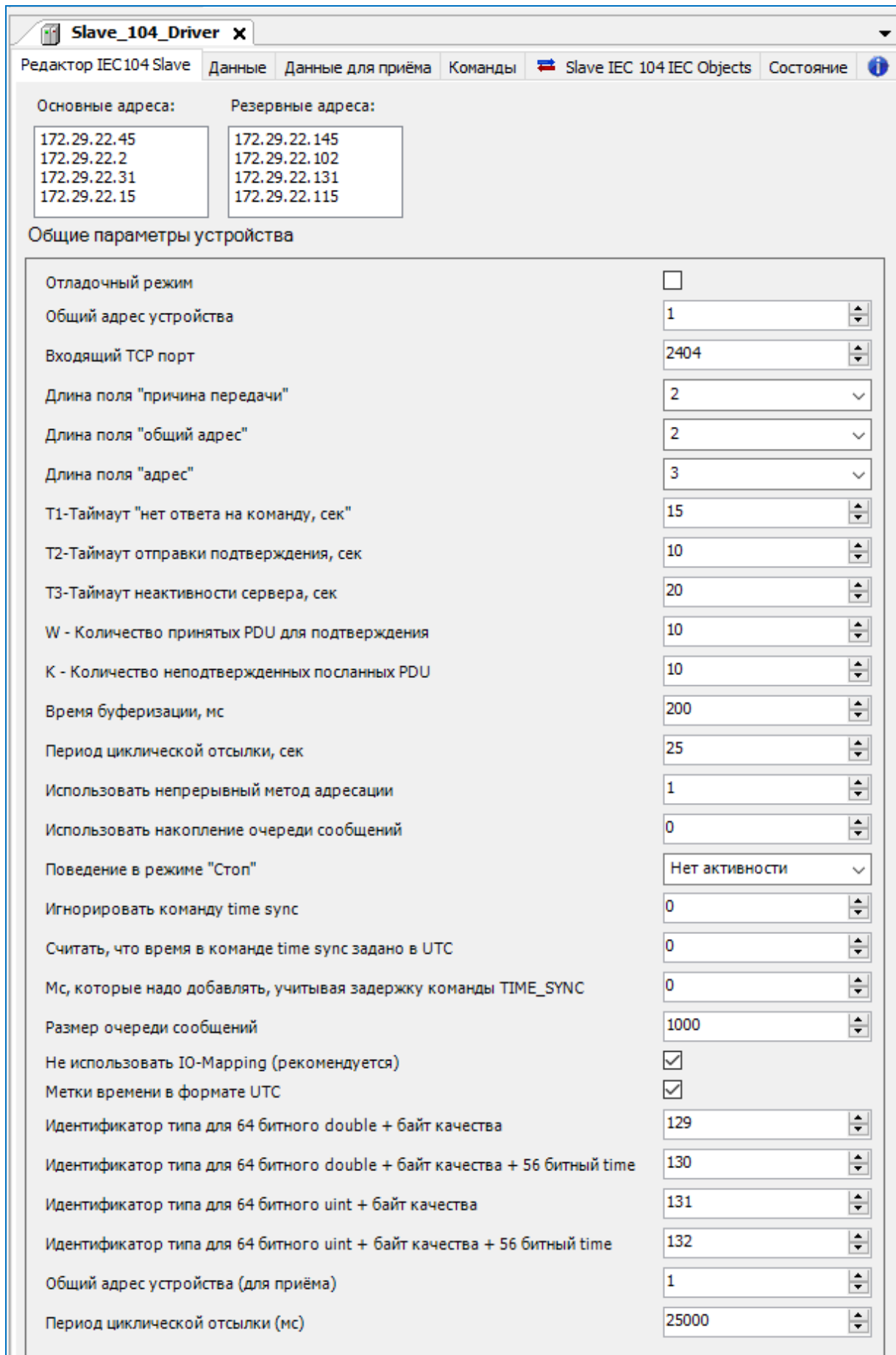



Рисунок 2 – Редактор устройства IEC104 Slave

Настройка списка IP-адресов Slave 104 Driver

В редакторе устройства **Slave 104 Driver** в верхней части вкладки находятся два блока: **Основные адреса** и **Резервные адреса**. Здесь указывается список IP-адресов, от которых разрешается принимать входящие запросы, иначе говоря, список разрешенных «мастеров» IEC-104.

	<p>ИНФОРМАЦИЯ</p> <p>Каждое соединение поддерживает отдельный буфер очереди данных. Для нормальной работы рекомендуем указывать не более 15 каналов основной/резервный для соединения</p>
---	--

Каждый мастер может вести опрос по основному или резервному каналу (в приоритете основной канал, мастер переключается на обмен с резервным, только если нет связи с основным). Добавить идентичные IP-адреса в основной и резервный список не получится.

Для добавления основного или резервного IP-адреса поместите курсор в блок адресов, нажмите правую кнопку мыши. В появившемся контекстном меню выберите пункт **Добавить**. Появится дополнительное окно, в котором нужно ввести IP-адрес (Рисунок 3).

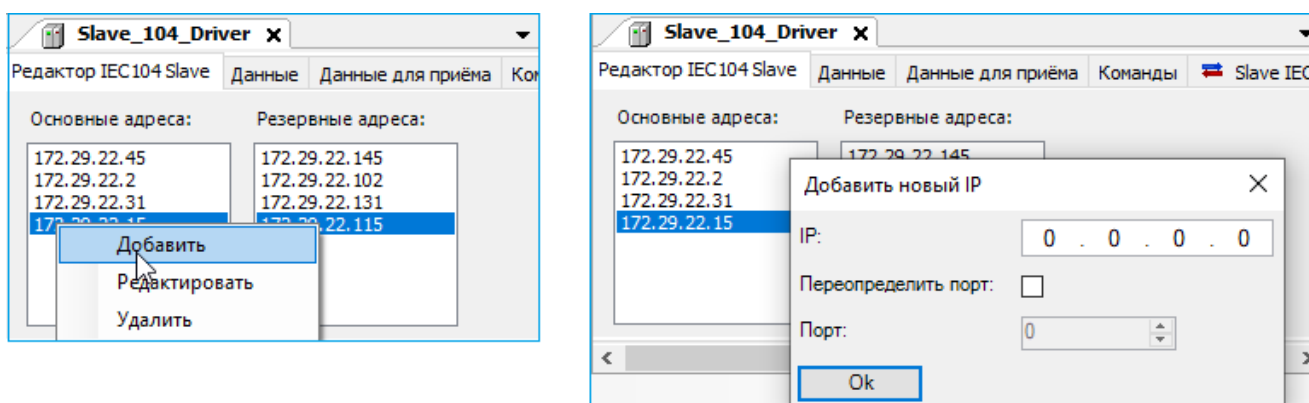


Рисунок 3 – Добавление IP-адреса

Также, кроме IP-адреса, можно задать номер порта соединения, для этого необходимо установить флажок в поле параметра **Переопределить порт** и указать необходимый номер в поле **Порт** (Рисунок 4). Можно указывать несколько комбинаций IP-адресов с номерами портов соединения. Существует ограничение на количество прослушиваемых портов, не более 10.

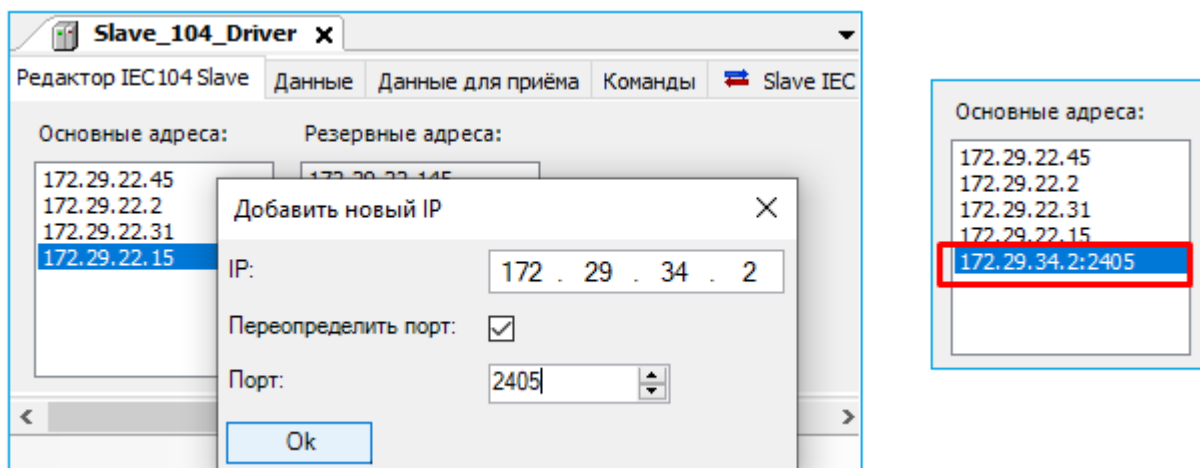


Рисунок 4 - Присвоение номера порта для IP-адреса



ИНФОРМАЦИЯ

Число резервных адресов не может быть больше, чем основных

Если опрос мастером ведется только по одному каналу, то такого мастера следует поместить в конец списка **Основные адреса**, резервные адреса для него не указывать.

Для редактирования адреса выберите его в списке, вызовите контекстное меню, пункт **Редактировать**. Появится дополнительное окно, в котором можно изменить IP-адрес. Для удаления адреса выберите его в списке, вызовите контекстное меню, пункт **Удалить**.



ВНИМАНИЕ!

Программа не запрашивает подтверждение на удаление

Начиная с 1.6.5 появилась возможность задавать диапазон адресов. Для задания диапазонов IP-адресов, необходимо в редакторе устройства **Slave 104 Driver** перейти на вкладку **Slave IEC 104 Конфигурация** (Рисунок 5).

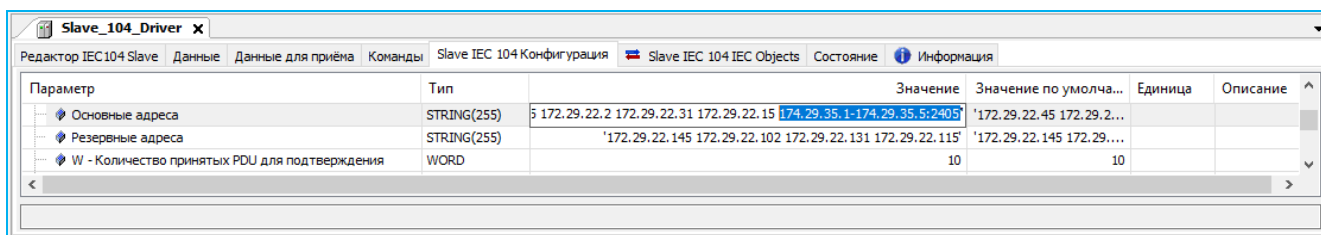


Рисунок 5 - Вкладка Slave IEC 104 Конфигурация



ИНФОРМАЦИЯ

Если вкладка **Slave IEC 104 Конфигурация** отсутствует, то необходимо перейти в меню **Инструменты** ⇒ **Опции** ⇒ найти пункт **Редактор устройств** и установить флажок напротив поля **Показывать общие окна конфигурации устройств**

На вкладке **Slave IEC 104 Конфигурация** двойной щелчок левой кнопкой мыши по ячейке в столбце **Значение** в строке **Основные / Резервные адреса** позволяет задать диапазоны соответствующих адресов.

При задании диапазона адресов необходимо учесть следующее:

- первым задается начальный адрес диапазона, затем, через дефис, конечный адрес;
- через двоеточие задается номер порта (задавать не обязательно). В случае отсутствия заданного значения порта, принимается значение по умолчанию;
- пробелы внутри спецификации диапазона не допускаются;
- первые три байта IP-адреса начала диапазона должны быть идентичны первым трем байтам конца диапазона.

Пример:

‘174.29.35.1-174.29.35.5:2405’, первые три байта IP-адреса – «174.29.35».

При некорректном вводе, в логе появится сообщение следующего вида:

«Addr. range 172.29.34.1-172.29.35.5 first 3 bytes differ».

Диапазоны могут быть заданы для основных и резервных адресов. В этом случае пары основной-резервный формируются последовательным перечислением. Например, если указаны следующие адреса:

Основные адреса	Резервные адреса
172.29.34.1	172.29.34.3
174.29.35.1-174.29.35.5	174.29.35.11-174.29.35.15
175.30.40.1-175.30.40.5*	

* - Диапазон 175.30.40.1-175.30.40.5 задает адреса основных каналов, не имеющих парных резервных. Такие диапазоны должны следовать последними в списке адресов

Пары [основной ↔ резервный] в этом случае будут следующие:

Соответствующие пары	
172.29.34.1	172.29.34.3
174.29.35.1	174.29.35.11
174.29.35.2	174.29.35.12
174.29.35.3	174.29.35.13
174.29.35.4	174.29.35.14
174.29.35.5	174.29.35.15

При формировании пар диапазонов основной ↔ резервный необходимо учесть следующее:

- количество адресов в основном и резервном диапазоне должно быть одинаковым. При некорректном вводе, в логе появится сообщение следующего вида:
«Reserve conn. 174.29.35.11-174.29.35.19 address num is different»;
- множество адресов в основном диапазоне не должно пересекаться со множеством адресов в резервном диапазоне. При некорректном вводе (например, основной:'172.29.34.1-172.29.34.9 и резервный:'172.29.34.8-172.29.34.16'), основные соединения будут приниматься, а резервные нет.

Для отображения внесенных изменений во вкладке **Редактор IEC Slave 104** закройте, а затем заново откройте вкладку редактор устройства **Slave 104 Driver**.

Настройка общих параметров устройства Slave 104 Driver

В редакторе устройства **Slave 104 Driver** в блоке **Общие параметры устройства** доступны для настройки параметры, представленные в таблице 3

Таблица 3 – Общие параметры устройства

Параметр	Описание	Значение по умолчанию
Отладочный режим	При установке флажка <input checked="" type="checkbox"/> в этом поле будут записаны в журнал работы контроллера все детали обмена по IEC-104, а именно передаваемые и принимаемые пакеты, комментарии к возникающим ошибкам и т.п.	
Общий адрес устройства	Содержит общий адрес (COMMON ADDRESS, ASDU ADDRESS) устройства, все отдаваемые данные и выполняемые команды имеют общую часть, равную значению этого параметра (см. спецификацию IEC-104)	1
Общий адрес устройства (для приема)	Содержит общий адрес (COMMON ADDRESS, ASDU ADDRESS) устройства, все получаемые данные имеют общую часть, равную значению этого параметра (см. спецификацию IEC-104)	1
Входящий TCP порт	Номер порта TCP, который открыт данным slave для приема входящих соединений	2404
Длина поля «причина передачи»	Содержит длину поля «причина передачи» (COT, Cause Of Transmission). Допустимые значения параметра: 1 или 2	2
Длина поля «общий адрес», Длина поля «адрес»	Характеристики PDU (Protocol Data Unit – протокольная единица, пакет), передаваемых по IEC-104, должны быть одинаковы для пары master slave. Допустимые значения параметра «общий адрес»: 1 или 2. Допустимые значения параметра «адрес»: 2 или 3	«общий адрес» - 2, «адрес» - 3

Параметр	Описание	Значение по умолчанию
T1-Таймаут «нет ответа на команду», сек	Время, в течение которого ожидается ответ на команду или подтверждение доставки PDU. Превышение таймаута приводит к разрыву соединения	15
T2-Таймаут отправки подтверждения, сек	При получении PDU запускается таймер на T2 секунд, подтверждение о приеме (пакет S-PDU) отсылается либо по истечению этого таймера, либо по факту приема некоторого предельного количества PDU, указанного в параметре W. T2 должен быть меньше T1	10
T3-Таймаут неактивности сервера, сек	Периодически как master, так и slave могут отсылать специальные пакеты для проверки связи. Если в течение интервала, заданного в параметре T3, нет никакого обмена, то slave может закрыть соединение	20
W-Количество принятых PDU для подтверждения	Количество принятых PDU, которое необходимо подтверждать. Рекомендуем выбирать значение W, не превышающее 2/3 от значения K (Master). Параметры должны быть согласованы на Master и Slave. Например, при K=10 на Master, W=8 на Slave	10
K-Количество неподтвержденных посланных PDU	Количество неподтвержденных отправленных PDU, после которого драйвер прекращает отправлять PDU, дожидаясь подтверждения отправленных PDU. Следует отметить, что в случае, когда требуется отправка большого количества данных, значение параметра K следует устанавливать большим. Например, если требуется отправка 20000 отсчетов в секунду, значение K нужно устанавливать от 1000 и больше. Иначе драйвер будет приостанавливать отставку данных, ожидая подтверждение клиента, размеры очередей при этом будут нарастать. Для оптимальной работы драйвера рекомендуется ограничить отставку на уровне 25000 отсчетов в секунду. Отсчет – изменение одного элемента данных. Если введено 500 элементов данных и все они изменяются раз в 10 мс, то это равносильно $500 \cdot 1000 / 10 = 50000$ отсчетов в секунду	10
Время буферизации, мс	Период краткосрочного ожидания пополнения очереди на отставку. Используется для накопления в одном PDU нескольких данных одного типа, для оптимизации	200
Период циклической отсылки (мс)	При описании элемента данных IEC-104 можно включить его в циклическую рассылку (установка флажка в поле Участствует в циклической рассылке). Для всех таких элементов их значения, даже при отсутствии изменений, будут отсылаться с указанным здесь периодом (в миллисекундах)	25000
Период циклической отсылки, сек	При описании элемента данных IEC-104 можно включить его в циклическую рассылку (установка флажка в поле Участствует в циклической рассылке). Для всех таких элементов их значения, даже при отсутствии изменений, будут отсылаться с указанным здесь периодом (в секундах).	25

Параметр	Описание	Значение по умолчанию
	Данный параметр оставлен в целях обратной совместимости. Если значение параметра Период циклической отсылки (мс) отлично от нуля, то данный параметр игнорируется. Начиная с версии СПО 1.6.5 параметр скрыт	
Использовать непрерывный метод адресации	Экономный режим IEC-104, в отсылаемых пакетах указывается адрес только первого элемента, считаем адреса остальных последовательно нарастающими. 0 – не используется, 1 – использовать этот метод	1
Использовать* накопление очереди сообщений	Накапливать данные при отсутствии связи с мастером. 0 – не используется (сбрасывается очередь данных после восстановления связи). Любое число, отличное от 0, активирует использование накопления сообщений с ограничением, которое задается параметром Размер очереди сообщений	0
Поведение в режиме «Стоп»**	Определяет поведение компонента при остановке программы. <i>Нет активности</i> - означает, что устройство Slave104 прекращает все соединения и не воспринимает новые; <i>Нормальная работа</i> - означает продолжение работы в обычном режиме	Нет активности
Игнорировать команду time sync	Если значение не равно 0, то при получении команды синхронизации времени (C_CS_NA_1), синхронизации времени не происходит. Если этот параметр равен 0, то команда синхронизации времени выполняется	0
Считать, что время в команде time sync задано в UTC	Если в этом поле задано значение 1, то, при синхронизации времени, время в составе команды будет интерпретировано, как UTC. Если задано значение 0, то, это время будет интерпретировано, как локальное время	0
Миллисекунды, которые надо добавлять, учитывая задержку команды TIME_SYNC	Корректировка команды синхронизации времени	0
Размер очереди сообщений	Определяет размер буфера сообщений для пересылки в нормальном режиме, а также при отсутствии связи. Следует избегать установки размера очереди сообщений более 50000 сообщений, поскольку в этом случае возникает риск исчерпания резерва оперативной памяти. Вместе с тем, размер очереди должен быть не меньше планируемого количества элементов данных. Это связано с тем, что при внешнем подключении IEC-104 Slave может получить команду общего опроса, что приведет к одновременному заполнению очереди последними значениями всех элементов данных. Если размер очереди	1000

Параметр	Описание	Значение по умолчанию
	меньше рекомендуемого, то часть данных не будет отправлена	
Не использовать IO-Mapping (рекомендуется)	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что чтение/запись привязанных переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов)	
Метки времени в формате UTC	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что значение метки времени соответствует UTC, иначе - локального времени	
Идентификатор типа для 64 битного double + байт качества	Задание ID для пользовательского типа LREAL(double), без метки времени	129
Идентификатор типа для 64 битного double + байт качества + 56 битный time	Задание ID для пользовательского типа LREAL(double), с меткой времени	130
Идентификатор типа для 64 битного uint + байт качества	Задание ID для пользовательского типа ULINT(uint), без метки времени	131
Идентификатор типа для 64 битного uint + байт качества + 56 битный time	Задание ID для пользовательского типа ULINT(uint), с меткой времени	132
<p>Примечания</p> <p>*- предусмотрена возможность отключения накопления очереди сообщений, когда ведомый контроллер находится в режиме «Стоп». То есть, если задан параметр Использовать накопление очереди сообщений (любое число, отличное от 0) и ведомый контроллер остановлен, то, при возобновлении работы и подключении к нему, устаревшие данные не будут поступать. Ведущий контроллер, при потерях связи, будет накапливать очередь сообщений и при подключении поступят все данные.</p> <p>1) Если требуется отключить накопление сообщений в режиме «Стоп», то пользователь может использовать следующий оператор:</p> <pre>Slave_104_Driver.m_outer.set_not_accum_in_stop(not_accum_in_stop);</pre> <p>После вызова этого оператора в Slave_104_Driver не будут копиться отчеты в режиме «Стоп».</p> <p>2) Если потребуется очистить очередь сообщений из приложения, то пользователь может использовать следующий метод:</p> <pre>Slave_104_Driver.purge_queue();</pre> <p>После вызова этого метода в Slave_104_Driver будет удалено содержимое буфера сообщений.</p> <p>** - предусмотрена возможность самостоятельно активировать «поведение в режиме «Стоп» в программном коде. Для активации режима требуется в программе присвоить значение TRUE свойству ActivateStopBehavior необходимого устройства:</p> <pre>«IEC104Slave_device_name».ActivateStopBehavior := TRUE;</pre> <p>После этого slave-устройство перейдет в Стоп-режим работы</p>		

Предусмотрено журналирование событий синхронизации времени при получении команды синхронизации (time sync). Клиент посылает команду синхронизации и на основной вкладке параметров устройства во вкладке **Журнал**, в зависимости от выбранного значения параметра **«Игнорировать команду time sync»**, должны появиться сообщения следующего вида: «Set system clock to ...», «Set system clock succeeded» или «Set system clock failed» (Рисунок 6).

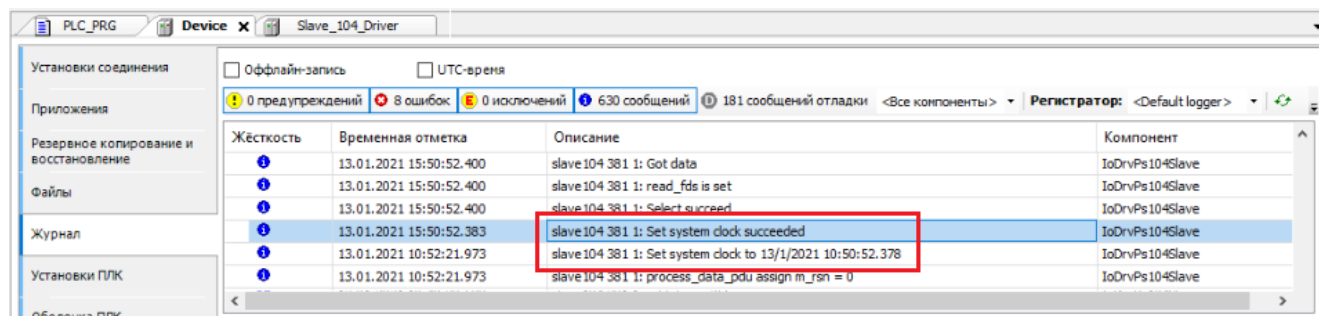


Рисунок 6 - Журналирование команды синхронизации времени

Также предусмотрено журналирование события переполнения очереди спорадических данных. Возможно два случая переполнения очереди:

- переполняется очередь изменений данных (одна очередь для всех соединений) и должно появиться сообщение следующего вида:
«Slave_104_Driver change queue overflow»
- переполняется очередь спонтанных сообщений конкретной пары соединений основное-резервное и должно появиться сообщение следующего вида:
«Slave_104_Driver 192.168.0.10:2404 queue overflow».

Создание списка элементов данных, передаваемых по IEC-104

В редакторе устройства **Slave 104 Driver** перейдите на внутреннюю вкладку **Данные** (Рисунок 7).

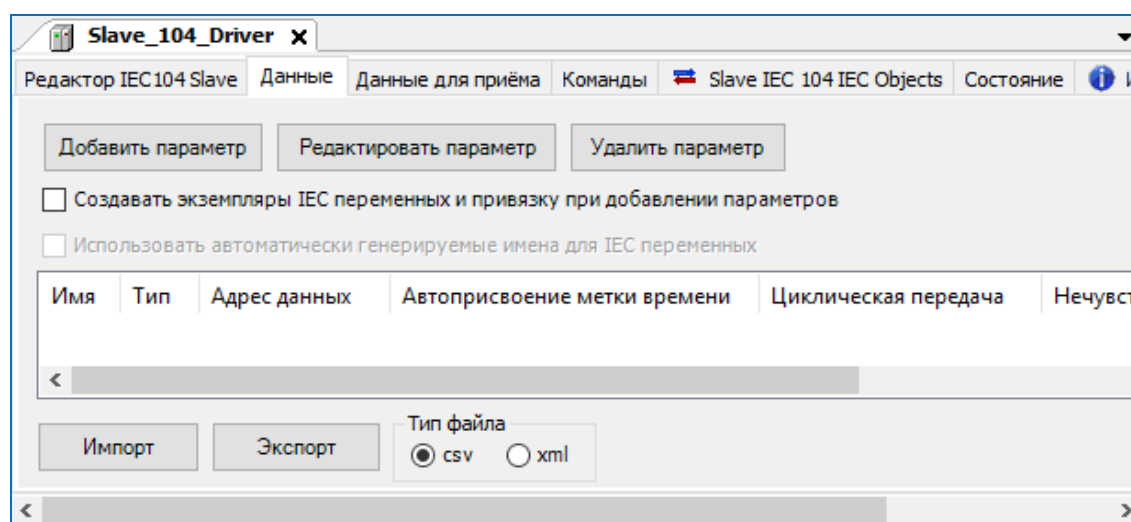


Рисунок 7 – Редактор Slave 104 Driver. Вкладка «Данные»

Элементы данных описываются как параметры. Для добавления, изменения и удаления элементов данных используются соответственно кнопки *Добавить параметр*, *Редактировать параметр*, *Удалить параметр*. Перейти к редактированию параметра также можно двойным щелчком левой кнопкой мыши по нужной строке. Окна ввода/редактирования элемента данных выглядят следующим образом (Рисунок 8).

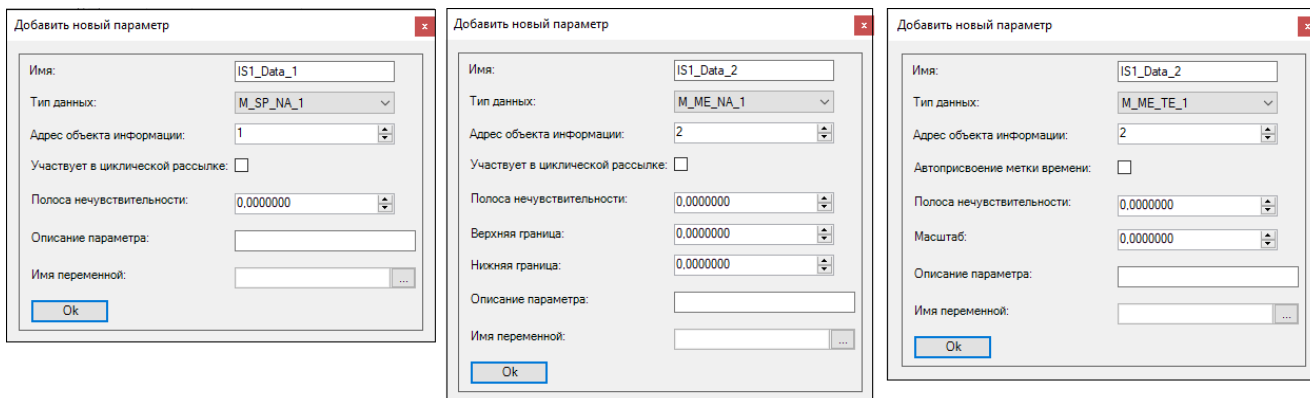




Рисунок 8 – Окна добавления нового параметра (элемента данных)

Для элемента данных укажите значения в полях. Параметры элементов представлены в таблице 4.

Таблица 4 – Параметры элементов данных IEC-104

Параметр	Описание
Имя	Наименование элемента данных
Тип данных	Ассоциированный с этим элементом данных тип IEC-104. Справочные типы данных IEC-104 приведены в таблице 5. Часто используемые типы данных: <ul style="list-style-type: none"> o M_SP_TB_1 – для дискретов с меткой времени; o M_ME_TF_1 – для float-значений с меткой времени; o M_IT_TB_1 – для целочисленных значений
Адрес объекта информации	Уникальный адрес элемента данных
Участствует в циклической рассылке	Установка флажка в этом поле активирует участие элемента данных в циклической рассылке. Для всех таких элементов их значения даже при отсутствии изменений будут отсылааться с периодом (в сек.), указанным на вкладке общих параметров
Полоса нечувствительности (deadband)	Устанавливаемая пользователем минимальная разница между последним отосланным значением и текущим измеренным (рассчитанным) в программе контроллера, превышение которой вызывает спонтанную отсылку
Верхняя граница, Нижняя граница	Параметры, задающие диапазон для нормализованных типов (M_ME_NA_1, M_ME_TA_1, M_ME_ND_1, M_ME_TD_1)
Автоприсвоение времени (для типов с меткой)	При установленном в этом поле флажке не требуется явно задавать временную метку для отсылаемого значения элемента данных.

Параметр	Описание
времени)	<p>Временная метка будет <i>автоматически</i> задаваться в локальном времени и присваиваться m_timestamp.</p> <p>При отсутствии флажка, пользователь <i>вручную</i> в программе присваивает m_timestamp (с возможностью смещения часового пояса для локального времени). Если не присвоить m_timestamp, то будет передаваться нулевая временная метка.</p> <p>Значение метки времени соответствует UTC или локальному времени в зависимости от параметра «Метки времени в формате UTC»</p>
Масштаб	Для масштабируемых параметров, при выборе типа M_ME_TE_1
Описание параметра	Опционально, текстовое описание элемента данных
Имя переменной*	<p>Указывается название переменной IEC (программы контроллера), в которой хранятся передаваемые данные.</p> <p>Для заполнения поля Имя переменной нажмите в этом поле кнопку , открывающую окно Ассистент ввода. Найдите нужную переменную. Если установлен флажок в поле Структурированный вид, то раскрывайте списки с помощью кнопки . Если флажок снят и переменные представлены одним большим списком, для удобства поиска воспользуйтесь фильтром</p>
<p>Примечание: -* - параметр присутствует в окне ввода/редактирования элемента данных, когда привязка переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов), при условии, что не снят флажок с параметра «Не использовать IO-Mapping»</p>	

Реализованы следующие идентификаторы типа, представленные в таблице 5

Таблица 5 – Перечень типов данных IEC-104

Тип данных	ID типа данных	Описание
M_SP_NA_1	1	Одноэлементная информация
M_DP_NA_1	2	Двухэлементная информация
M_ST_NA_1	5	Информация о положении отпаяк
M_BO_NA_1	7	Строка из 32 бит
M_ME_NA_1	9	Измеряемая величина, нормализованное значение
M_ME_NB_1	11	Измеряемая величина, масштабируемое значение
M_ME_NC_1	13	Измеряемая величина, формат с плавающей запятой
M_IT_NA_1	15	Интегральная сумма
M_ME_ND_1	21	Измеряемая величина, нормализованное значение без описателя качества
M_SP_TB_1	30	Одноэлементная информация с 56-битной меткой времени
M_DP_TB_1	31	Двухэлементная информация с 56-битной меткой времени
M_ST_TB_1	32	Информация о положении отпаяк с 56-битной меткой времени

Тип данных	ID типа данных	Описание
M_BO_TB_1	33	Строка из 32-х бит с 56-битной меткой времени
M_ME_TD_1	34	Измеряемая величина нормализованное значение с 56 битной меткой времени
M_ME_TE_1	35	Измеряемая величина масштабированное значение с 56 битной меткой времени
M_ME_TF_1	36	Измеряемая величина с плавающей запятой с 56 битной меткой времени
M_IT_TB_1	37	Интегральная сумма с 56-битной меткой времени
M_EP_TD_1	38	Информация о работе релейной защиты с 56-битной меткой времени
M_LR_NA	129*	Информация пользовательского типа LREAL из 64-х бит + байт качества, без метки времени
M_LR_TC	130*	Информация пользовательского типа LREAL из 64-х бит + байт качества с 56-битной меткой времени
M_ULINT_NA	131*	Информация пользовательского типа ULINT из 64-х бит + байт качества, без метки времени
M_ULINT_TC	132*	Информация пользовательского типа ULINT из 64-х бит + байт качества с 56-битной меткой времени

Примечание - * - ID задается пользователем

Заполненная вкладка **Данные** показана на рисунке 9.

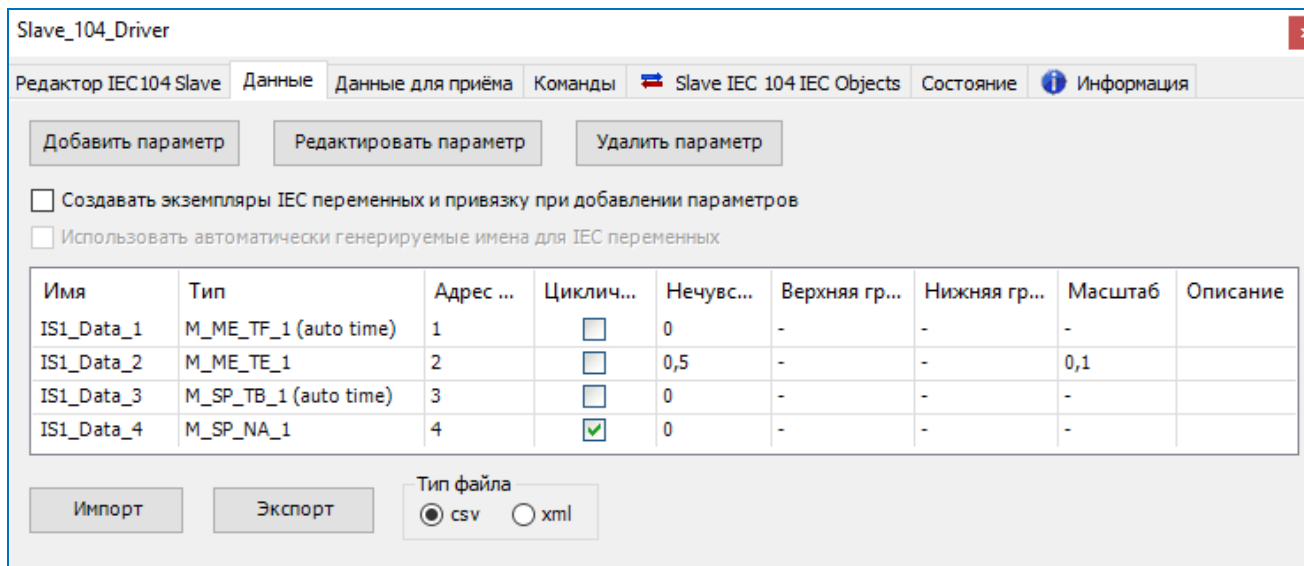


Рисунок 9 – Пример списка элементов данных

Создание списка элементов данных, получаемых по IEC-104

Наряду с отправкой изменившихся данных по протоколу IEC 60870-104, объект Slave_104_Driver также может и принимать данные от другой стороны обмена и, в соответствии с принятой протокольной информацией, изменять значения переменных приложения.

В то время как на вкладке **Данные** пользователь может конфигурировать передаваемые по протоколу данные, то на вкладке **Данные для приема** можно сконфигурировать данные, которые получаются по протоколу. Вид и назначение полей на вкладках **Данные** и **Данные для приема** идентичны (Рисунок 10).

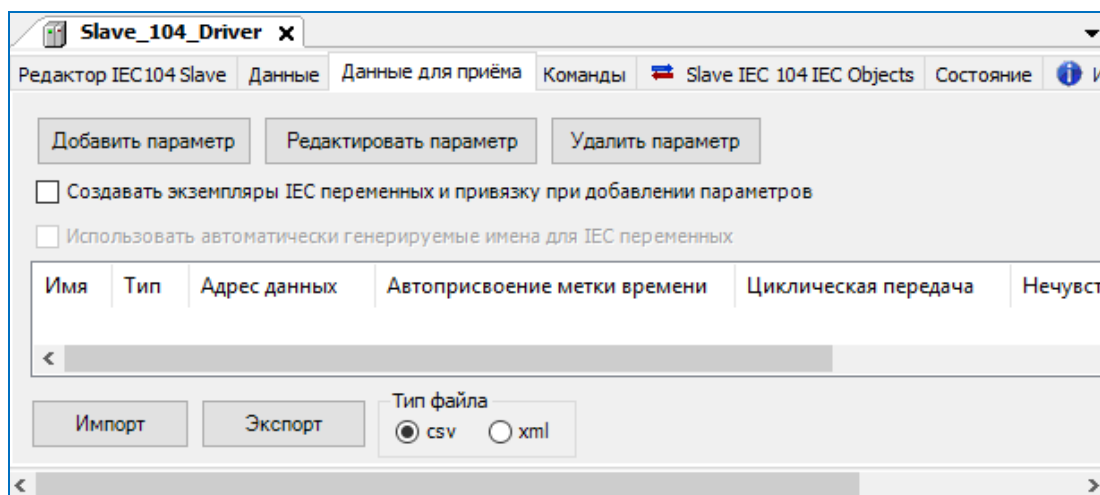


Рисунок 10 – Редактор Slave 104 Driver. Вкладка «Данные для приема»

И там и там задается соотношение между параметрами протокола (адрес, тип) и переменной приложения. Разница только в том, что переменные приложения, указанные на вкладке **Данные для приема**, будут изменять свое значение, метку времени, качество после получения входящих PDU с причиной передачи «Спонтанная», «Циклическая», «Общий опрос».

Адреса информационных объектов «Данных» и «Данные для приема» могут пересекаться. В рамках протокола разница определяется параметром «Адрес ASDU» (иногда его называют «Общий адрес»). Иными словами, существует 2 независимых множества адресов информационных объектов (IO address) для отдаваемых и получаемых данных. Каждое множество имеет свой уникальный общий адрес.

Например:

Общий адрес для отдаваемых данных = 1. Отдаются данные с адресами [1, 2, 3];

Общий адрес для получаемых данных = 2. Получаются данные с адресами [2, 3, 4].

В случае, когда множество получаемых данных не пустое, Slave_104_Driver формирует команду общего опроса в ответ на команду START_DATA после установки входящего соединения. Устройство типа Slave_104_Driver не посылает никаких команд за исключением команды общего опроса в вышеописанном случае.

Создание списка команд, получаемых по IEC-104

Кроме запроса данных с контроллера в IEC-104 реализованы команды, используемые для установки значения какой-либо переменной, либо выполнения по сигналу каких-либо действий.

В редакторе устройства Slave 104 Driver перейдите на внутреннюю вкладку **Команды**. Общий вид редактора команд аналогичен редактору элементов данных. Команды на данной вкладке описываются как параметры. Для добавления, изменения и удаления команд используются соответственно кнопки *Добавить параметр*, *Редактировать параметр*, *Удалить параметр*. Перейти к редактированию команды также можно двойным щелчком левой кнопкой мыши по нужной строке.

Окно ввода/редактирования команд выглядит следующим образом (Рисунок 11).

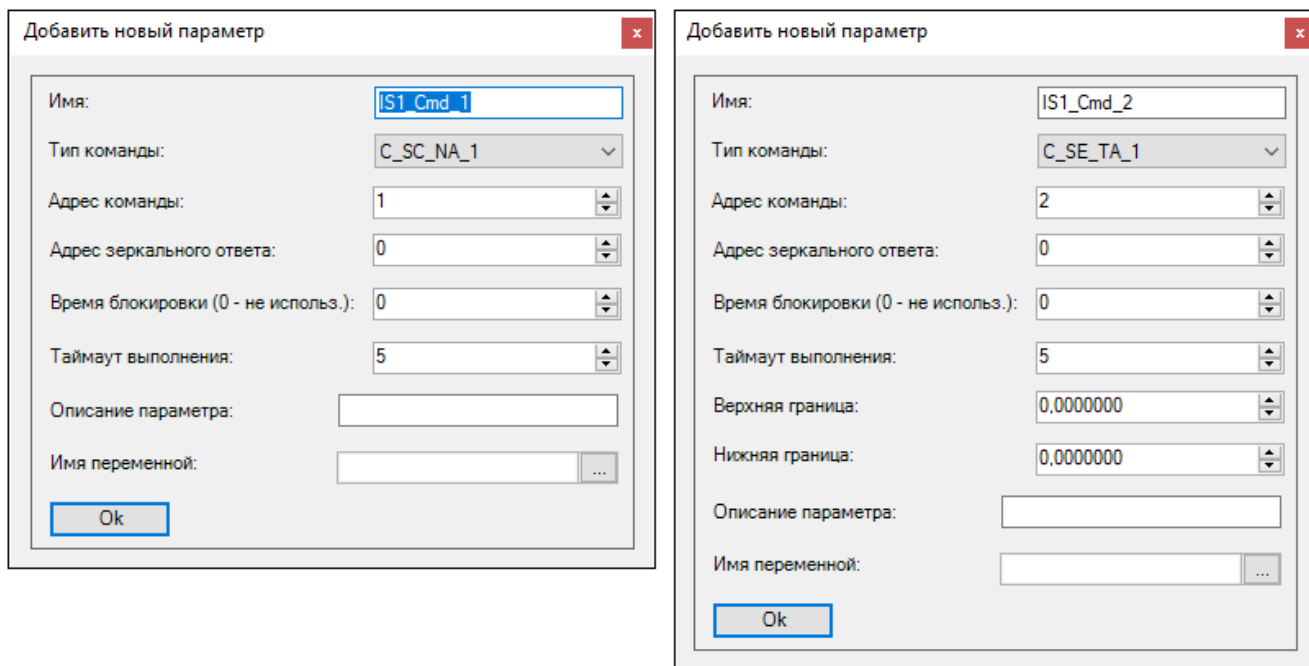


Рисунок 11 – Окно добавления новой команды

Для команд укажите значения в полях. Параметры команд представлены в таблице 6

Таблица 6 – Параметры команд IEC-104

Параметр	Описание
Имя	Наименование команды
Тип данных	Ассоциированный с этой командой тип IEC-104. Справочные типы данных IEC-104 приведены в таблице 7
Адрес объекта информации	Уникальный адрес команды
Адрес зеркального ответа	Адрес элемента данных, в который будет помещено значение, переданное командой (0 – не используется)
Время блокировки (0 – не использ.)	Команды могут быть простыми и с «выборкой перед выполнением» (Select Before Operate). При ненулевом значении это время, в течение которого команда остается в состоянии «выбрана» для выполнения (после отправки запроса select). Успешное выполнение команды (execute) возможно до истечения данного таймаута



Параметр	Описание
Масштаб	Для команд, задающих масштабируемые значения
Верхняя граница, Нижняя граница	Пара параметров, определяющих диапазон значений команды для нормализованных команд (C_SE_NA_1 и C_SE_TA_1);
Описание параметра	Опционально, текстовое описание команды
Имя переменной*	<p>Указывается название переменной IEC (программы контроллера), в которой хранятся передаваемые команды.</p> <p>Для заполнения поля Имя переменной нажмите в этом поле кнопку , открывающую окно Ассистент ввода. Найдите нужную переменную. Если установлен флажок в поле Структурированный вид, то раскрывайте списки с помощью кнопки . Если флажок снят и переменные представлены одним большим списком, для удобства поиска воспользуйтесь фильтром</p>
<p>Примечание: -* - параметр присутствует в окне ввода/редактирования команды, когда привязка переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов), при условии, что не снят флажок с параметра «Не использовать I/O-Mapping»</p>	

Таблица 7 – Перечень типов данных IEC-104

Тип данных	ID типа данных	Описание
C_SC_NA_1	45	Бинарная команда
C_DC_NA_1	46	Команда с тремя состояниями
C_RC_NA_1	47	Регулирующая пошаговая команда
C_SE_NA_1	48	Команда уставки (нормализованное значение)
C_SE_NB_1	49	Команда уставки (масштабируемое значение)
C_SE_NC_1	50	Команда уставки (значение типа float)
C_BO_NA_1	51	Команда установки побитового регистра
C_SC_TA_1	58	Бинарная команда с 56-битной меткой времени
C_DC_TA_1	59	Команда с тремя состояниями с 56-битной меткой времени
C_RC_TA_1	60	Регулирующая пошаговая команда с 56-битной меткой времени
C_SE_TA_1	61	Команда уставки (нормализованное значение) с 56-битной меткой времени
C_SE_TB_1	62	Команда уставки (масштабируемое значение) с 56-битной меткой времени
C_SE_TC_1	63	Команда уставки (значение типа float) с 56-битной меткой времени
C_BO_TA_1	64	Команда установки побитового регистра с 56-битной меткой времени
C_CS_NA_1	103	Команда синхронизации времени

Возможность сохранять принятое время в переменной, не выполняя синхронизацию

Если у объекта `Slave_104_Driver` установлен указатель на интерфейс `i_clock_handler` с помощью вызова `Slave_104_Driver.m_outer.set_clock_sync_handler(i_clock_handler)`, то при получении команды `C_CS_NA_1` сервер `iec-60870-104(101)` не будет пытаться сам синхронизировать время, вместо этого будет вызвана пользовательская функция `on_clock_sync(timestamp_type)`. В этой функции интерфейса `i_clock_sync_handler` пользователь имеет возможность выполнить действия, продиктованные логикой работы контроллера.

Пример использования:

```
FUNCTION_BLOCK clocksync_handler_fb IMPLEMENTS IEC_LIB.i_clock_sync_handler

METHOD on_clock_sync // пользовательский метод обработки команды C_CS_NA_1
VAR_INPUT
t : IEC_LIB.timestamp_type;
END_VAR
GVL_1.g_ts := t;

PROGRAM PLC_PRG
VAR
first_cycle : BOOL := TRUE;
timesync_handler : clocksync_handler_fb;
END_VAR
IF ( first_cycle )
THEN
first_cycle := FALSE;
Unbalanced_Secondary_101_Driver_1.m_outer_master.set_clock_sync_handler(timesync_h
andler); // для 101-го
Slave_104_Driver.m_outer.set_clock_sync_handler(timesync_handler); // для 104-го
END_IF
```

Создание переменных для работы с данными/командами, передаваемыми по IEC-104

Элементы данных и команды, описанные в редакторе IEC-104, в терминологии среды разработки являются **каналами ввода-вывода**. Для реализации обмена данными необходимо создать эти каналы (см. предыдущие разделы), создать в программном коде переменные специальных типов, после чего связать переменные с каналами ввода-вывода.

Требуемые типы переменных (функциональные блоки) описаны в библиотеке `PsIecCommon`. Эта библиотека, а также использующая ее библиотека `PsIoDrvIec104Slave` автоматически подключаются при добавлении устройства `Slave 104 Driver`.

Для преобразования временных меток `m_timestamp` в `sys_timedate` и обратно, в библиотеке `PsIecCommon` присутствуют – функция `sys_timedate_to_timestamp` и функциональный блок `timestamp_to_sys_timedate`.

Для данных используются следующие функциональные блоки: `bo_tb_fb`, `ep_td_fb`, `it_tb_fb`, `me_tf_fb`, `sp_tb_fb`, `me_td_fb`. В наименовании блока фактически указан тип данных IEC-104, например, блоку `bo_tb_fb` соответствует тип `M_BO_TB_1`, а `me_tf_fb` соответствует `M_ME_TF_1`. Количество функциональных блоков меньше количества типов данных IEC-104, каждый функциональный блок может использоваться сразу для нескольких типов.

Таблица 8 – Соответствие функциональных блоков и типов данных IEC-104

Функц. блок	Тип данных IEC-104, указываемый для канала ввода-вывода	Комментарий
bo_tb_fb	M_BO_NA_1, M_BO_TA_1, M_BO_TB_1	Для 4-байтных целых без знака (32-битные bitstring)
	M_DP_NA_1, M_DP_TA_1, M_DP_TB_1	Допустимые значения 0, 1, 2
	M_ST_NA_1, M_ST_TA_1, M_ST_TB_1	Допустимые значения 0 ... 255
ep_td_fb	M_EP_TD_1	Для событий защиты оборудования
it_tb_fb	M_IT_NA_1, M_IT_TA_1, M_IT_TB_1	Для 4-байтных целых со знаком
me_tf_fb	M_ME_NC_1, M_ME_TC_1, M_ME_TF_1	Для 4-байтных чисел с плавающей точкой
	M_ME_NA_1, M_ME_TA_1, M_ME_TD_1	Для нормализованных чисел с плавающей точкой, описание канала должно содержать hi/low
	M_ME_NB_1, M_ME_TB_1, M_ME_TE_1	Для масштабируемых величин (scaled values), описание канала должно содержать scale
sp_tb_fb	M_SP_NA_1, M_SP_TA_1, M_SP_TB_1	Для булевых дискретов
me_td_fb	M_ME_NA_1, M_ME_TA_1, M_ME_TD_1	Для нормализованных чисел с плавающей точкой, описание канала должно содержать hi/low

Для создания переменных откройте редактор ПЛК-программы. Например, в редакторе ST для программы PLC_PRG создание переменных выглядит следующим образом:

```
PROGRAM PLC_PRG
VAR
    IS1_Data_1:me_tf_fb;
    IS1_Data_2:me_tf_fb;
    IS1_Data_3:sp_tb_fb;
    IS1_Data_4:sp_tb_fb;
END_VAR
```

Ниже представлен пример объявления переменных:

```
active_connections : STRING;           //список соединений
linked : BOOL;                          //статус соединения

iec1_real          : PsIoDrvIec104Slave.me_tf_fb; //float с меткой времени
iec3_di            : PsIoDrvIec104Slave.sp_tb_fb; //дискрет с меткой времени
iec5_int           : PsIoDrvIec104Slave.it_tb_fb; //целочисленное с меткой времени

ts                 : PsIoDrvIec104Slave.timestamp_type;
                  //переменная типа «временная отметка для IEC-104»
cmd1               : PsIoDrvIec104Slave.common_command_type; //команда;
```

Предусмотрена возможность программно получить список подключенных к slave мастеров (соединений):

```
active_connections := Slave_104_Driver.m_active_connections;
```

Переменная *active_connections* показывает список IP-адресов в виде строки, где адреса разделены пробелами: '172.29.23.152 172.29.23.10'.

Также предусмотрена возможность получать статус соединения (true, если соединение установлено, иначе – false):

```
linked := Slave_104_Driver.Linked();
```

Свойства (properties) функциональных блоков для передачи данных следующие:

- **value** – данному свойству присваивается значение, которое будет передаваться по IEC-104. Используемый тип значения (имеется в виду простой тип – int, real, bool...) будет зависеть от того, какой тип (по IEC-104) имеет канал ввода-вывода, связанный с данной переменной. Свойство *value* принимает значение длиной не более 4 байт;
- **quality** – байт качества значения. Значение по умолчанию – 0 (качество *good*). Допустимы также 128 (80 hex) – качество *invalid*, 64 (40 hex) – качество *substituted*. Все допустимые значения указаны в описании протокола IEC-60870-5-104(101);
- **timestamp** – временная отметка, назначаемая при изменении состояния. Представлена типом данных *timestamp_type*. Может генерироваться автоматически, если указано в описании связанного канала ввода-вывода, либо задаваться вручную. В случае автоматической генерации *timestamp* назначается при изменении значения *value*.

Ниже приведен пример генерации значения *timestamp_type* из текущей даты/времени. Реализовано в виде метода для программы, требуется подключение библиотеки SysTimeRtc.

Объявление

```
METHOD PUBLIC get_current_timestamp : PsIoDrvIec104Slave.timestamp_type
VAR_INPUT
END_VAR
VAR
```

```
    stTime          : SYSTIME;
    sdtDateHiRes    : SYSTIMEDATE;
```

```
END_VAR
```

Реализация

```
IF (SysTimeRtcHighResGet(stTime)=0) AND
(systimertc.SysTimeRtcConvertHighResToDate(stTime, sdtDateHiRes)=0)
THEN
get_current_timestamp.m_year := sdtDateHiRes.wYear;
get_current_timestamp.m_month := sdtDateHiRes.wMonth;
get_current_timestamp.m_day := sdtDateHiRes.wDay;
get_current_timestamp.m_hour := sdtDateHiRes.wHour;
get_current_timestamp.m_minute := sdtDateHiRes.wMinute;
get_current_timestamp.m_second := sdtDateHiRes.wSecond;

get_current_timestamp.m_millisecond := sdtDateHiRes.wMilliseconds;
```

END_IF

Для команд предусмотрен один тип функционального блока: **common_command_type**. Его основные компоненты следующие:

- **m_value** – передается содержание всего поля Information Object PDU команды. Имеет тип `common_union` т.е. объединение размером 4 байта. Данные 4 байта могут быть интерпретированы как BYTE, WORD, DWORD, INT, DINT, REAL в зависимости от типа команды;
- **m_timestamp** – в случае, когда команда имеет тип, включающий в себя метку времени, данное поле содержит метку времени, полученную в составе команды. Для команд без метки времени данное поле не используется.

Ниже приведен пример использования переменных в программном коде.

```
active_connections := PsIoDrvIec104Slave.m_active_connections;
iecl_real.value := 3.14; //список IP адресов активных соединений
iec3_di.value:= 1; //задаем действительное значение
iec3_di.timestamp := get_current_timestamp(); //задаем дискретное значение
iecl_int.value:=515; //дополнительно генерируем timestamp
cmd_value:= cmd1.m_value; //целочисленное значение
//пришедшая команда
```

Привязка переменных программы к элементам данных и командам IEC-104

Начиная с версии СПО 1.5.7.0, чтение/запись привязанных переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов), при условии, что не снят флажок с параметра «**Не использовать IO-Mapping**» (параметр активирован).

Если IEC устройство содержит параметры старого формата, для которых применяется механизм I/O Mapping (соотнесение входов/выходов), то их можно обновить до нового формата с использованием прямой адресации переменных с помощью кнопки *Обновить параметры* (Рисунок 12). Кнопка появляется на вкладке **Данные** или **Команды** только в том случае, если активирован параметр «**Не использовать IO-Mapping**» и обнаружены параметры старого формата. Такие параметры расположены в другом диапазоне ID.

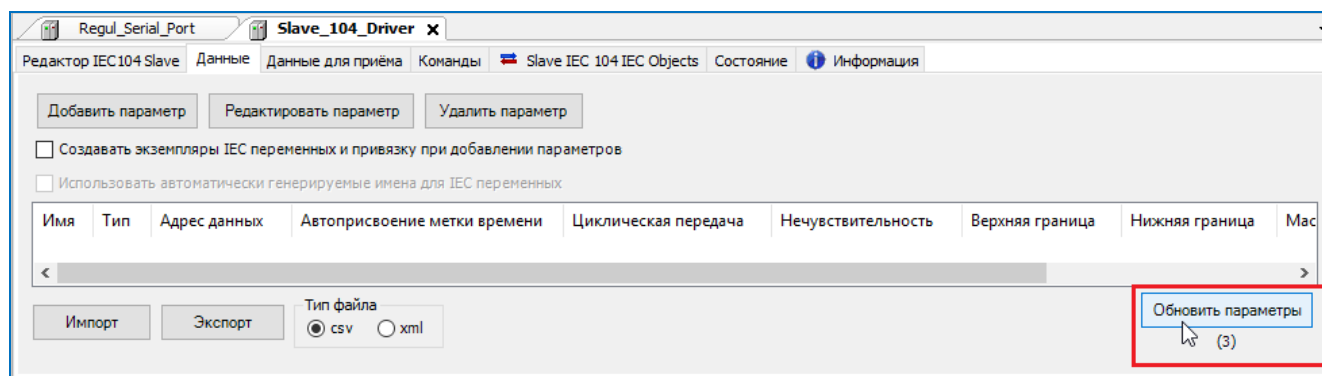


Рисунок 12 - Обновление параметров до нового формата

При активированном параметре «**Не использовать IO-Mapping**» в редакторе устройства **Slave 104 Driver** будет отсутствовать внутренняя вкладка **Соотнесение входов/выходов**.



ВНИМАНИЕ!

Рекомендуется не деактивировать параметр «**Не использовать IO-Mapping**»

Если деактивировать параметр (снять флажок), необходимо будет закрыть основную вкладку, а затем заново открыть и при добавлении данных/команд автоматически появится вкладка **Соотнесение входов/выходов** (Рисунок 13).

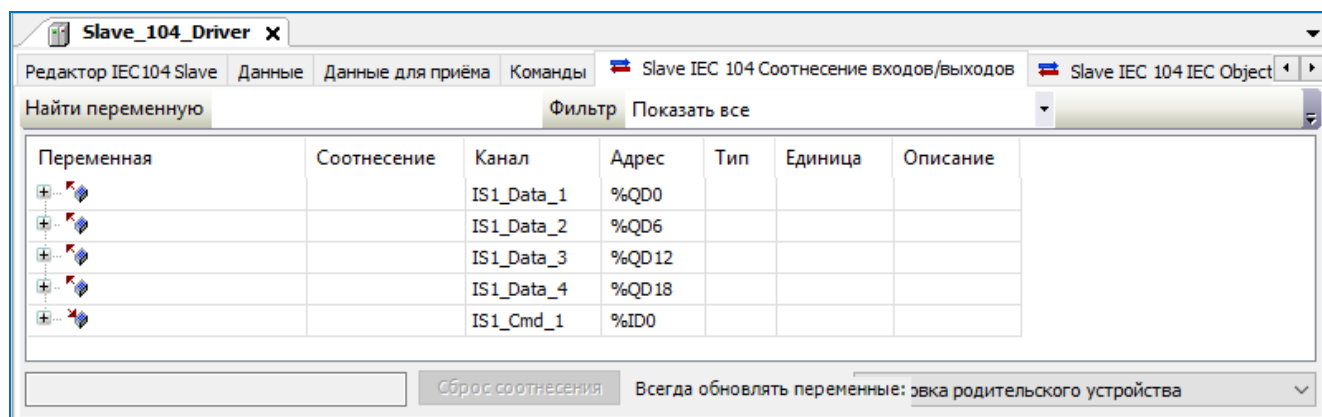



Рисунок 13 – Slave 104 Driver Соотнесение входов/выходов

На этой вкладке представлен список каналов ввода-вывода (в терминологии среды разработки), ассоциированных с устройством. Когда пользователь создает элемент данных или команду в соответствующем редакторе (вкладка **Данные**, вкладка **Команды**), этот параметр автоматически появляется здесь в виде канала вывода (для данных) или канала ввода (для команд). Наименование добавленного параметра указано в колонке **Канал**.

Для того, чтобы значения, приходящие/отправляемые по каналу, были доступны в программе контроллера, нужно привязать канал к переменной программы. Дважды щелкните левой кнопкой мыши в строке нужного канала. Появится курсор (можно вручную ввести имя переменной, семантика имен описана ниже) и кнопка , открывающая окно **Ассистент ввода** (Рисунок 14).

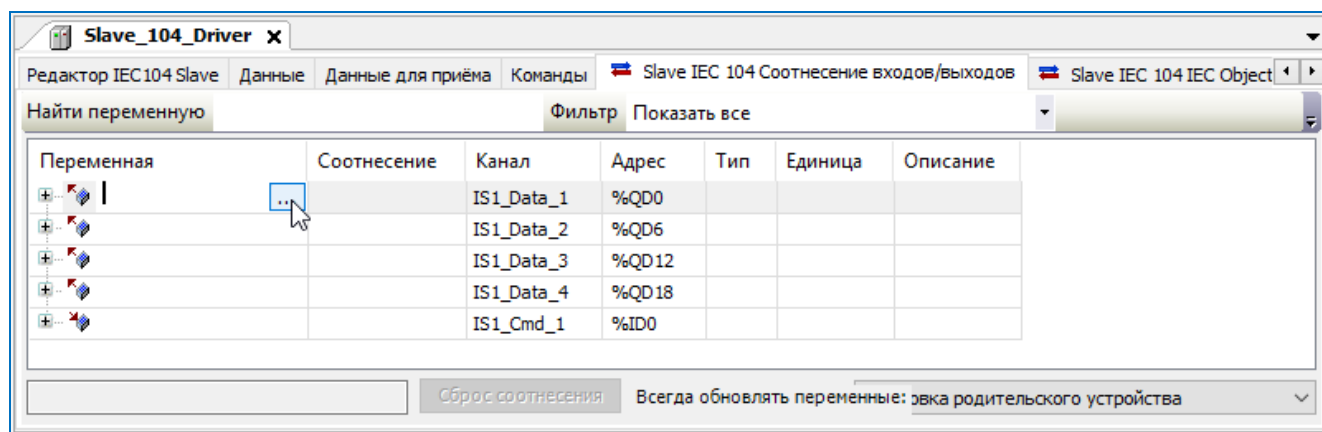


Рисунок 14 – Ручной ввод переменной или вызов ассистента ввода

В окне **Ассистент ввода** (Рисунок 15) найдите нужную переменную. Если установлен флажок в поле **Структурированный вид**, то раскрывайте списки с помощью кнопки **+**. Если флажок снят и переменные представлены одним большим списком, для удобства поиска воспользуйтесь фильтром.

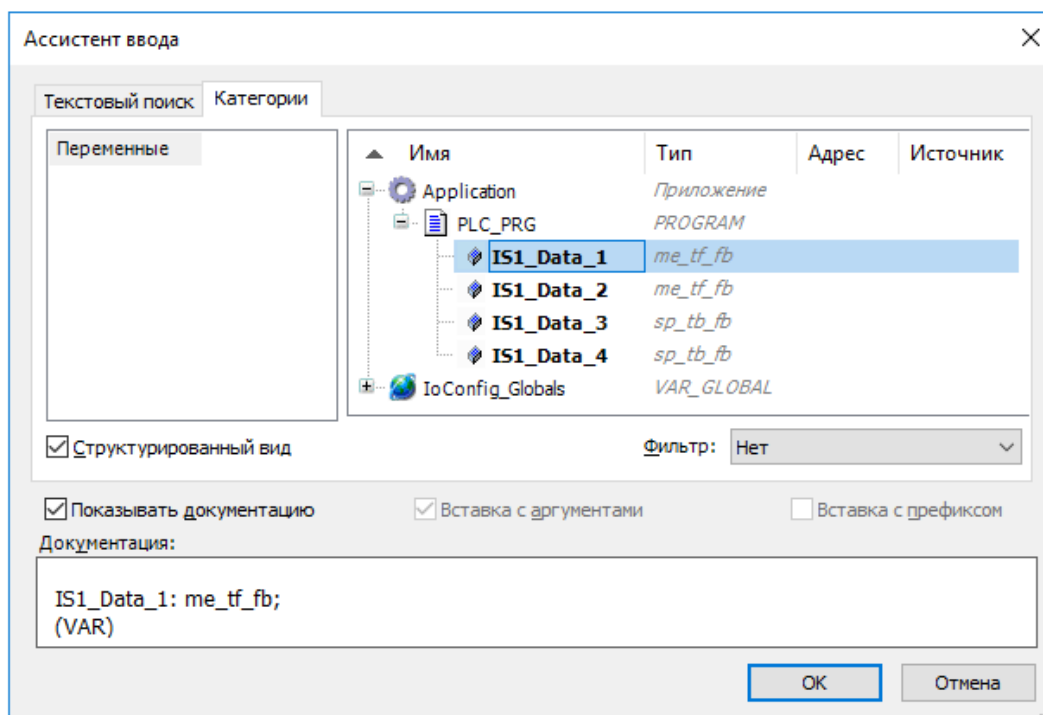


Рисунок 15 – Диалоговое окно «Ассистент ввода»

После выбора переменной нажмите кнопку **OK**, закроется окно **Ассистент ввода**, а переменная появится на вкладке **Соотнесение входов/выходов** (Рисунок 16).

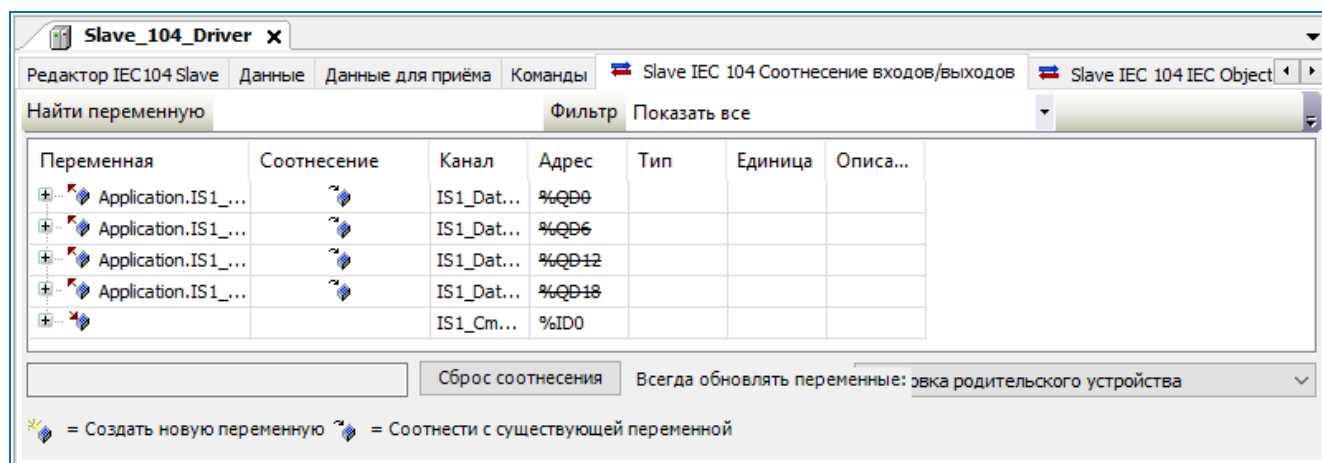


Рисунок 16 – Переменные привязаны к каналам вывода

При ручном вводе семантика имен следующая (угловые скобки при вводе НЕ используются): *<Имя приложения>.<Имя программного юнита>.<Имя переменной>*

Как ранее указано, для привязки к каналу IEC-104 используются только переменные, представляющие из себя функциональные блоки определенных типов, описанных в библиотеке PsIecCommon.



ИНФОРМАЦИЯ

Тип данных IEC-104 (например, M_BO_NA_1) задается при описании элемента данных или команды. После «привязки» переменной приложения к соответствующему каналу, переменная (типа bo_tb_fb) будет передаваться по протоколу IEC-60870-5-104(101) с использованием типа M_BO_NA_1

Все функциональные блоки, используемые для описания переменных IEC-104, имеют поле метки времени. Но в случаях, когда тип данных IEC-104, указанный в описании элемента данных или команды, не имеет метки времени (например, M_BO_NA_1), поле **Метка времени** функционального блока просто не используется. Также в зависимости от типа данных IEC-104 поле **Метка времени** может сериализоваться в 56-битный или в 24-битный timestamp.

Автоматическая генерация переменных и привязка к каналам ввода-вывода

При добавлении параметров на вкладках **Данные** и **Команды** можно активировать режим, при котором для каждого канала автоматически создаются переменные ПЛК-программы и формируется привязка переменной к каналу. Для этого при создании нового элемента данных или команды установите флажок в поле **Создавать экземпляры IEC переменных и привязку при добавлении параметров** (Рисунок 17).

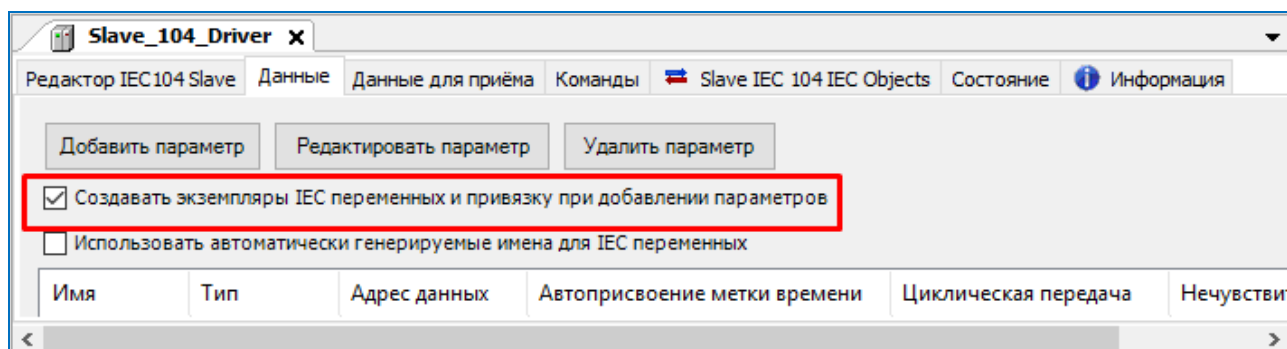


Рисунок 17 – Установка флажка в поле «Создавать экземпляры IEC переменных и привязку при добавлении параметров»

В этом режиме при добавлении канала для устройства Slave 104 Driver создается собственный список глобальных переменных с именем вида I104_GVL_X, где X – условный порядковый номер устройства, начиная с единицы (Рисунок 18). В этом списке будет создана переменная соответствующего типа.

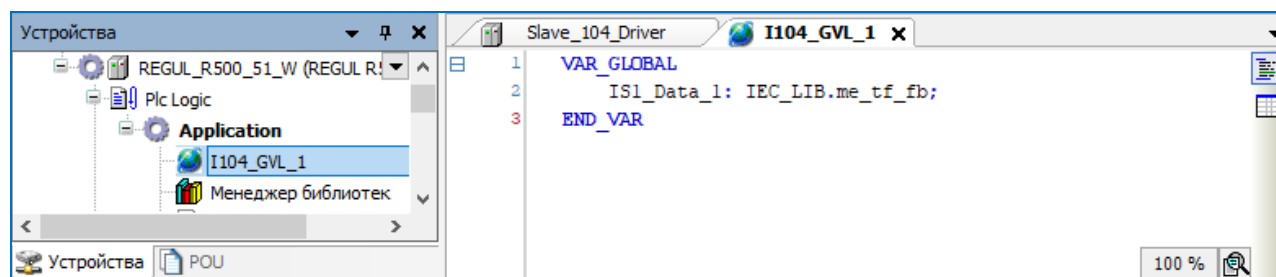


Рисунок 18 – Пример описания переменных в списке I104_GVL_1

При наличии вкладки **Соотнесение входов/выходов** новая переменная будет автоматически привязана к новому каналу (Рисунок 19). Наименование переменной будет совпадать с именем канала (пробелы заменяются на подчеркивания).

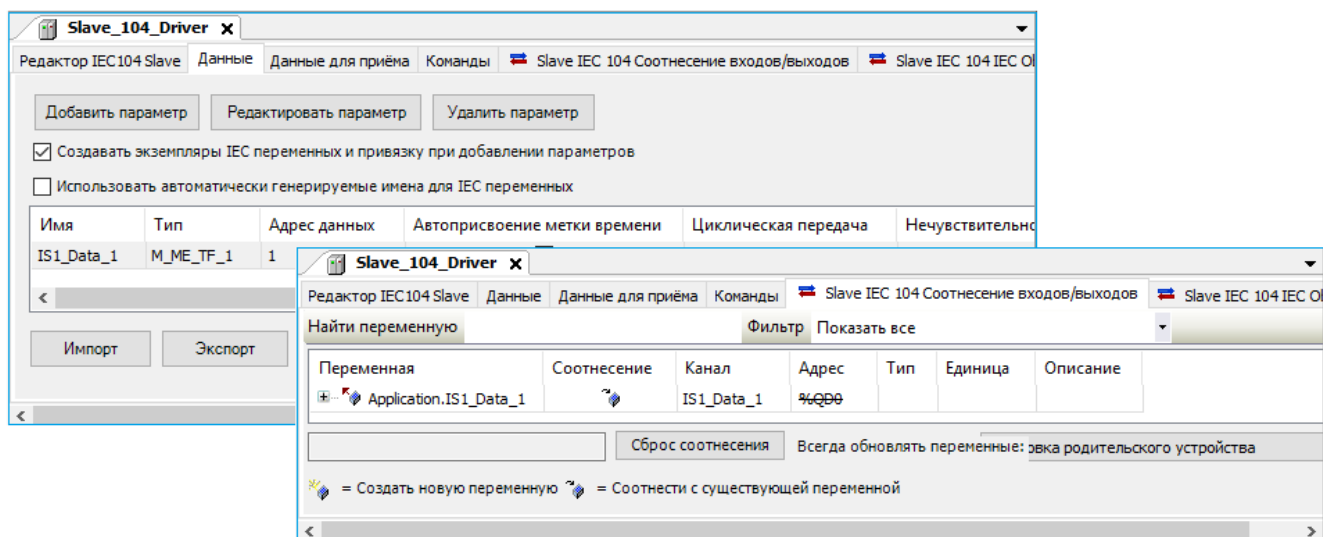


Рисунок 19 – Автоматическое создание переменной и автоматическая привязка ее к каналу

При установленном флажке в поле **Использовать автоматически генерируемые имена для IEC переменных** имена переменных будут создаваться на основе шаблона: *iec<случайное число>_data_fb*. При отсутствии флажка в этом поле имена генерируемых переменных имеют

вид: *IS<X>_Data_<N>* (для элементов данных), *IS<X>_Cmd_<N>* (для команд), где X – номер Slave-устройства, N – нумератор переменных. Генерируемые по умолчанию имена каналов данных и команд совпадают с именами связываемых переменных.



ВНИМАНИЕ!

Использование вызова системной функции спорадической отправки из прикладной программы не является стандартизированной опцией протокола. Настоятельно рекомендуем применять только в тех системах, где это является прямым требованием проекта

Спонтанные уведомления инициируются в момент, когда объекту данных присваивается значение, отличное от предыдущего.

Например:

```
sp_na : IEC_LIB.sp_tb_fb;  
sp_na.value := NOT sp_na.value;
```

Так же можно инициировать отсылку спонтанного уведомления, если значение объекта данных не изменилось. Осуществить это можно двумя способами:

- 1) `Slave_104_Driver.send_spontaneous(sp_na)` – только один драйвер отсылает эти данные;
- 2) `sp_na.send_spontaneous ()` – все драйверы, которые в конфигурации имеют `sp_na`, отправят `sp_na`.

Функция спорадической передачи `send_spontaneous()` срабатывает всегда, независимо от установленного значения параметра полосы нечувствительности (`deadband`). Данные будут отправлены с причиной передачи «спонтанная», даже если значение не вышло за границы полосы нечувствительности. В остальных случаях (когда переменная меняет свое значение), если установлен `deadband` - он применяется всегда.

Настройка контроллера в качестве Master 104

Контроллер выступает в роли ведущего, опрашивающего устройства. При настройке режима необходимо задать параметры slave-устройства, которое будет опрашиваться контроллером. Кроме того, требуется описать набор данных, который будет запрашиваться.

Добавление устройства Master 104 Driver

В окне дерева устройств поместите курсор на название контроллера, нажмите правую кнопку мыши. В появившемся контекстном меню выберите пункт **Добавить устройство...** Откроется окно **Добавить устройство**, в котором выберите: *Regul* → *МЭК 60870* → *TCP 60870-104-Master* → *Master 104 Driver*. Нажмите кнопку **Добавить устройство** или дважды щелкните левой кнопкой мыши. Выбранное устройство появится в проекте в дереве устройств.

Далее к устройству **Master 104 Driver** нужно подключить одно или несколько slave-устройств (outer slaves), которые будут опрашиваться контроллером: *Regul* → *МЭК 60870* → *TCP 60870-104-Master* → *IEC 104 Outer Slave* (Рисунок 20).

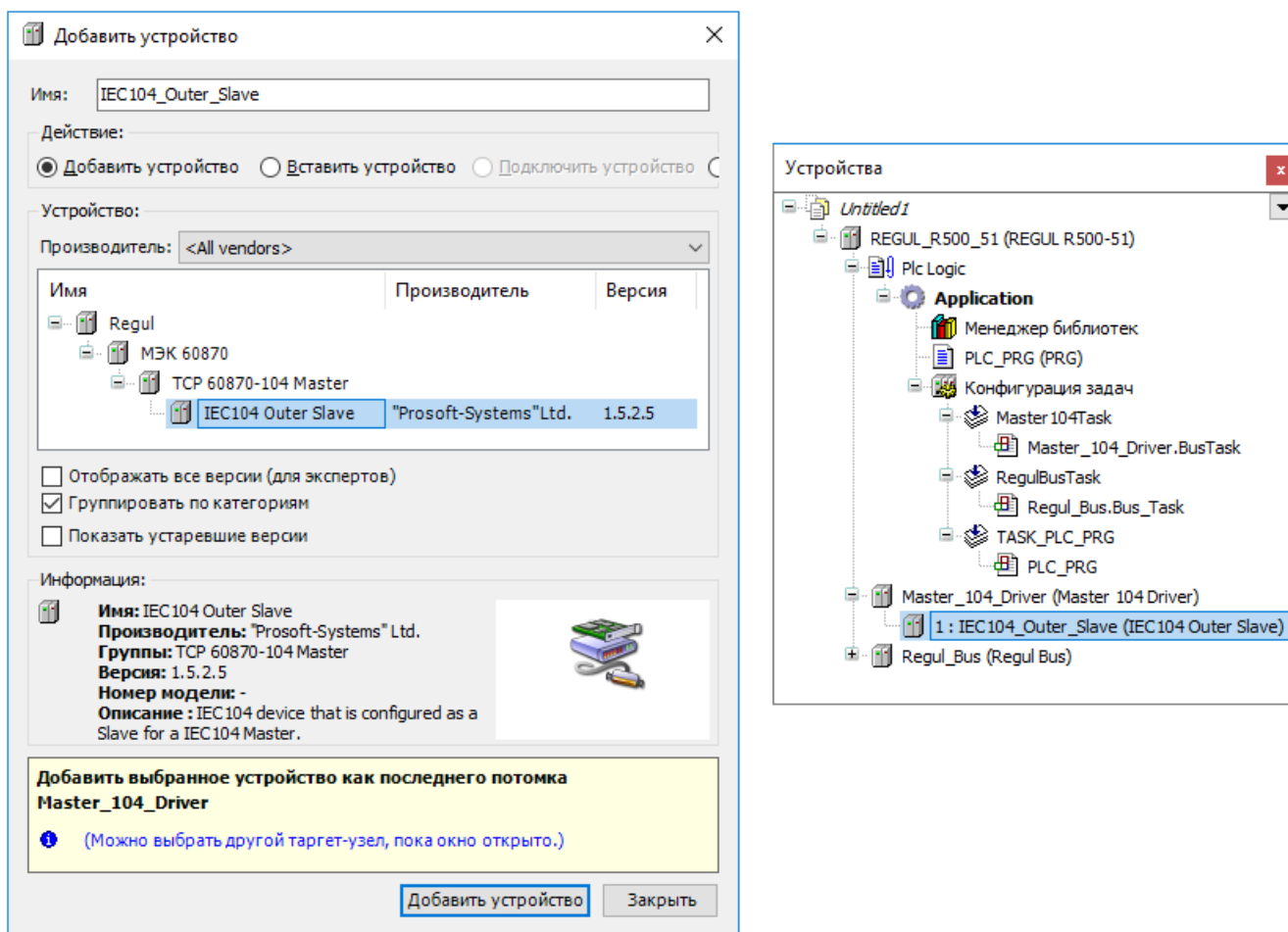


Рисунок 20 – Добавление в конфигурацию контроллера устройства IEC 104 Outer Slave

Двойным щелчком по названию устройства **IEC 104 Outer Slave** откройте вкладку параметров. По умолчанию открывается первая внутренняя вкладка **Редактор IEC 104 Outer Slave** (Рисунок 21).

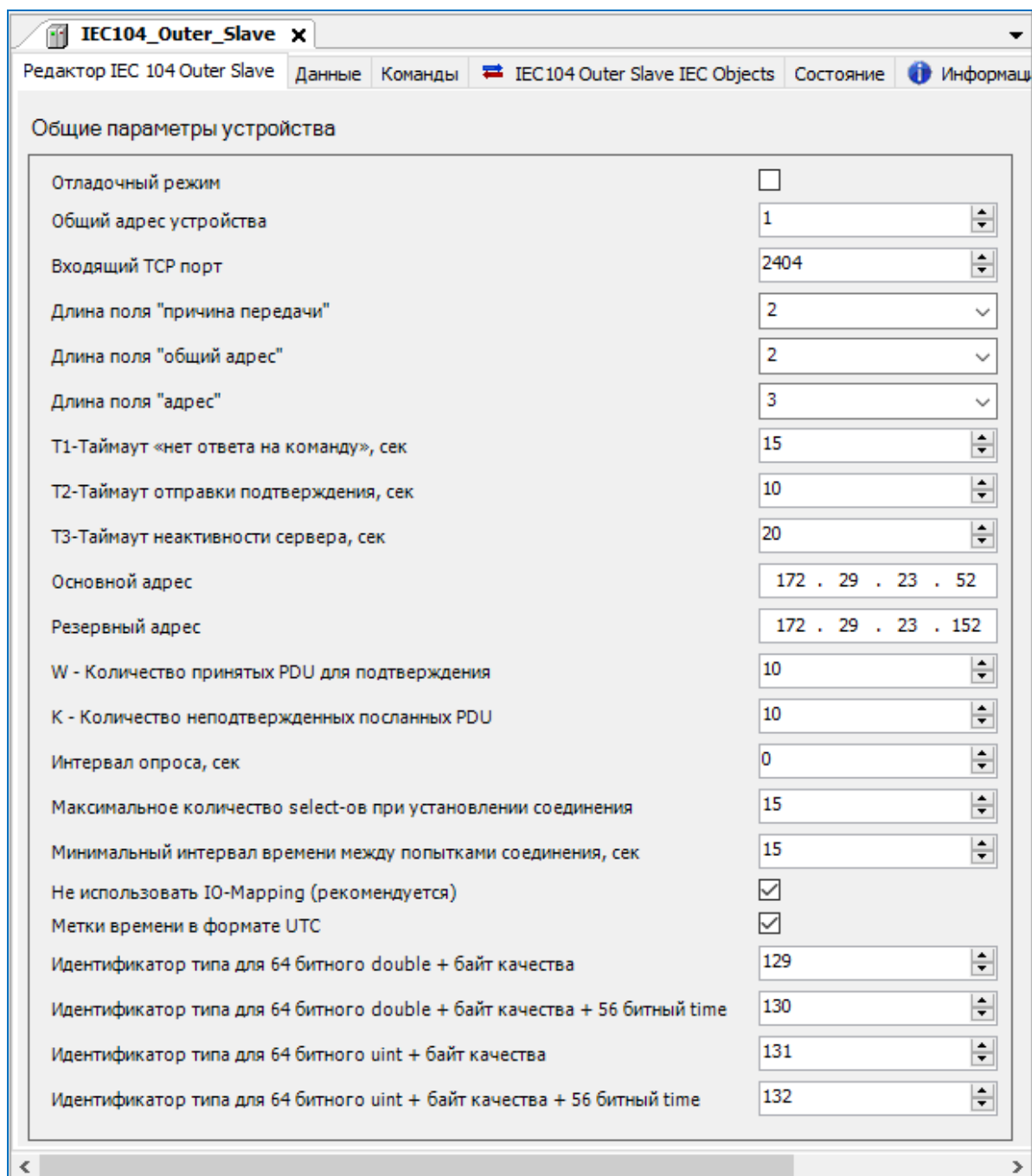


Рисунок 21 – Редактор устройства IEC 104 Outer Slave

Настройка общих параметров устройства IEC 104 Outer Slave

В редакторе устройства **IEC 104 Outer Slave** в блоке **Общие параметры устройства** доступны для настройки параметры, представленные в таблице 9.

Таблица 9 – Общие параметры устройства

Параметр	Описание	Значение по умолчанию
Отладочный режим	При установке флажка <input checked="" type="checkbox"/> в этом поле будут записаны в журнал работы контроллера все детали обмена по IEC-104, а именно передаваемые и принимаемые пакеты, комментарии к возникающим ошибкам и т.п.	
Общий адрес устройства	Содержит общий адрес (COMMON ADDRESS, ASDU ADDRESS) устройства, все данные и команды имеют общую часть, равную значению этого параметра (см. протокол IEC-104)	1
Входящий TCP порт	Номер порта TCP, который открыт данным slave для приема входящих соединений	2404
Длина поля «причина передачи»	Содержит длину поля «причина передачи» (COT, Cause Of Transmission). Допустимые значения параметра: 1 или 2	2
Длина поля «общий адрес», Длина поля «адрес»	Характеристики PDU (Protocol Data Unit – протокольная единица, пакет), передаваемых по IEC-104, должны быть одинаковы для пары master slave. Допустимые значения параметра «общий адрес»: 1 или 2. Допустимые значения параметра «адрес»: 2 или 3	«общий адрес» – 2, «адрес» – 3
T1-Таймаут «нет ответа на команду», сек	Время, в течение которого ожидается ответ на команду или подтверждение доставки PDU. Превышение таймаута приводит к разрыву соединения	15
T2-Таймаут отправки подтверждения, сек	При получении PDU запускается таймер на T2 секунд, подтверждение о приеме (пакет S-PDU) отсылается либо по истечению этого таймера, либо по факту приема некоторого предельного количества PDU, указанного в параметре W. T2 должен быть меньше T1	10
T3-Таймаут неактивности сервера, сек	Периодически как master, так и slave могут отсылать специальные пакеты для проверки связи. Если в течение интервала, заданного в параметре T3, не было никакой активности со стороны клиента, то посылается тестовое PDU и включается таймер T1. Если в течении интервала T1 не было активности со стороны другой стороны обмена, соединение разрывается	20
Основной адрес	IP-адрес основного соединения	
Резервный адрес	IP-адрес резервного соединения	
W-Количество принятых PDU для подтверждения	Количество принятых PDU, которое необходимо подтверждать. Рекомендуем выбирать значение W, не превышающее 2/3 от значения (Slave). Параметры должны быть согласованы на Master и Slave. Например, при K=10 на Slave, W=8 на Master.	10

Параметр	Описание	Значение по умолчанию
К-Количество неподтвержденных посланных PDU	<p>Количество неподтвержденных отправленных PDU, после которого драйвер прекращает отправлять PDU, дожидаясь подтверждения отправленных PDU.</p> <p>Следует отметить, что в случае, когда требуется отправка большого количества данных, значение параметра К следует устанавливать большим. Например, если требуется отправка 20000 отсчетов в секунду, значение К нужно устанавливать от 1000 и больше. Иначе драйвер будет приостанавливать отправку данных, ожидая подтверждение клиента, размеры очередей при этом будут нарастать.</p> <p>Для оптимальной работы драйвера рекомендуется ограничить отправку на уровне 25000 отсчетов в секунду. Отсчет – изменение одного элемента данных. Если введено 500 элементов данных и все они изменяются раз в 10 мс, то это равносильно $500 \cdot 1000 / 10 = 50000$ отсчетов в секунду.</p>	10
Интервал опроса, сек	Период отсылки команды общего опроса. Значение 0 – отсылка команды общего опроса не производится	0
Максимальное количество select-ов при установлении соединения	Максимальное количество запросов типа «Выборка перед выполнением» (Select before operate)	15
Минимальный интервал времени между попытками соединения, сек (T0)	Промежуток времени в секундах между попытками установления соединения с сервером	15
Не использовать IO-Mapping (рекомендуется)	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что чтение/запись привязанных переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов)	
Метки времени в формате UTC	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что значение метки времени соответствует UTC, иначе - локального времени	
Идентификатор типа для 64 битного double + байт качества	Задание ID для пользовательского типа LREAL(double), без метки времени	129
Идентификатор типа для 64 битного double + байт качества + 56 битный time	Задание ID для пользовательского типа LREAL(double), с меткой времени	130
Идентификатор типа для 64 битного uint + байт качества	Задание ID для пользовательского типа ULINT(uint), без метки времени	131
Идентификатор типа для 64 битного uint	Задание ID для пользовательского типа ULINT(uint), с	132

Параметр	Описание	Значение по умолчанию
+ байт качества + 56 битный time	меткой времени	

Создание списка элементов данных, передаваемых по IEC-104

Список элементов данных имеет ограничение в 10000 элементов для каждого устройства Slave 104 Outer Slave.

В редакторе устройства IEC 104 Outer Slave перейдите на внутреннюю вкладку **Данные** (Рисунок 22).

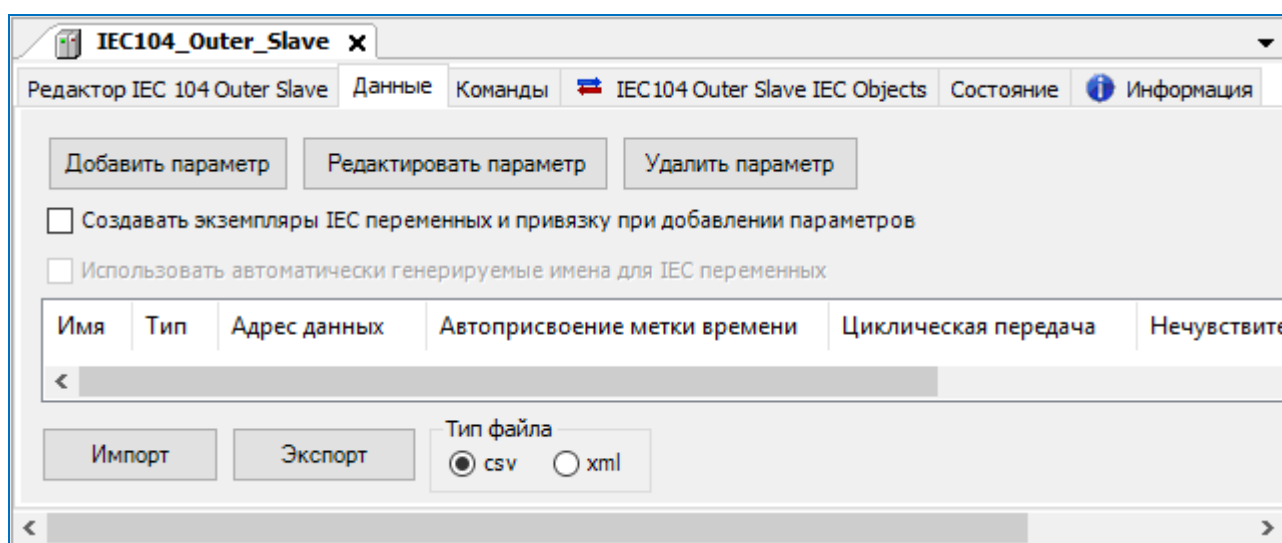


Рисунок 22 – Редактор IEC 104 Outer Slave. Вкладка «Данные»

Элементы данных описываются как параметры. Для добавления, изменения и удаления элементов данных используются соответственно кнопки **Добавить параметр**, **Редактировать параметр**, **Удалить параметр**. Перейти к редактированию параметра также можно двойным щелчком левой кнопкой мыши по нужной строке. Окно ввода/редактирования элемента данных выглядит следующим образом (Рисунок 23).

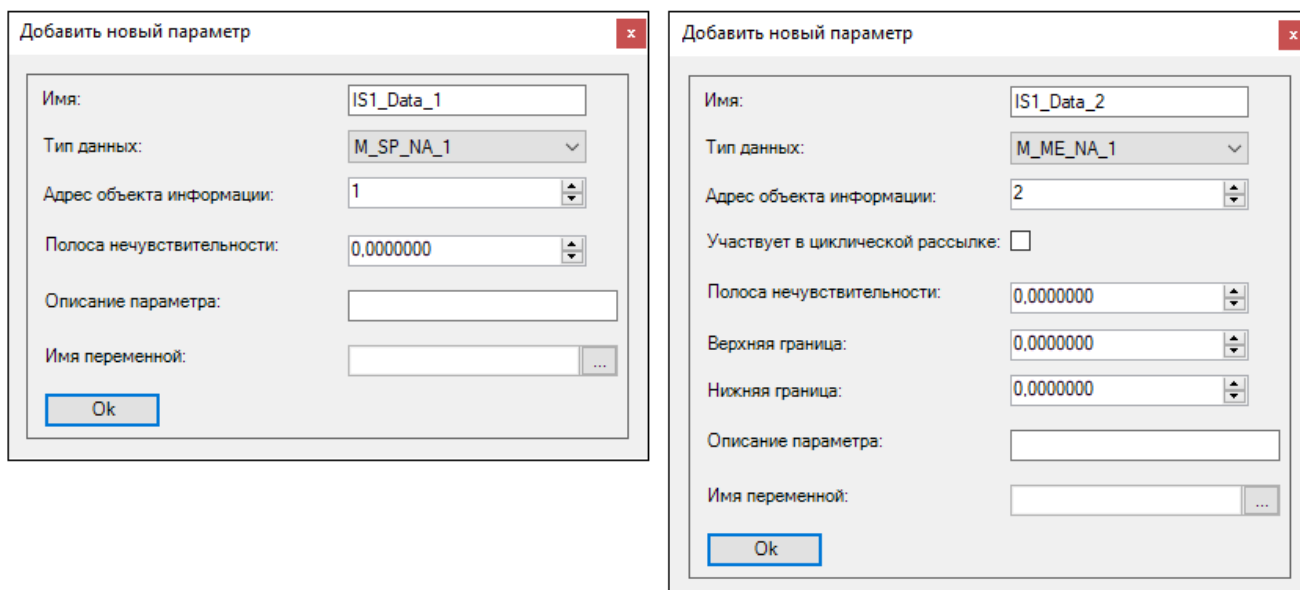


Рисунок 23 – Окно добавления нового параметра (элемента данных)

Для элемента данных укажите значения в полях. Параметры элементов представлены в таблице 10.

Таблица 10 – Параметры элементов данных IEC-104

Параметр	Описание
Имя	Наименование элемента данных
Тип данных	Ассоциированный с этим элементом данных тип IEC-104. Справочные типы данных IEC-104 приведены в таблице 11. Часто используемые типы данных: <ul style="list-style-type: none"> o M_SP_TV_1 – для дискретов с меткой времени; o M_ME_TF_1 – для float-значений с меткой времени; o M_IT_TV_1 – для целочисленных значений
Адрес объекта информации	Уникальный адрес элемента данных
Участует в циклической рассылке	Установка флажка в этом поле активирует участие элемента данных в циклической рассылке. Для всех таких элементов их значения даже при отсутствии изменений будут отсылааться с периодом (в сек.), указанным на вкладке общих параметров
Полоса нечувствительности (deadband)	Устанавливаемая пользователем минимальная разница между последним отосланным значением и текущим измеренным (рассчитанным) в программе контроллера, превышение которой вызывает спонтанную отсылку
Верхняя граница, Нижняя граница	Параметры, задающие диапазон для нормализованных типов (M_ME_NA_1, M_ME_TA_1, M_ME_ND_1, M_ME_TD_1)
Автоприсвоение времени (для типов с меткой времени)	При установленном в этом поле флажке не требуется явно задавать временную метку для отсылаемого значения элемента данных. Временная метка будет <i>автоматически</i> задаваться в локальном времени и присваиваться m_timestamp .



Параметр	Описание
	<p>При отсутствии флажка, пользователь <i>вручную</i> в программе присваивает m_timestamp (с возможностью смещения часового пояса для локального времени). Если не присвоить m_timestamp, то будет передаваться нулевая временная метка.</p> <p>Значение метки времени соответствует UTC или локальному времени в зависимости от параметра «Метки времени в формате UTC»</p>
Масштаб	Для масштабируемых параметров, при выборе типа M_ME_TE_1
Описание параметра	Опционально, текстовое описание элемента данных
Имя переменной*	<p>Указывается название переменной IEC (программы контроллера), в которой хранятся передаваемые данные.</p> <p>Для заполнения поля Имя переменной нажмите в этом поле кнопку , открывающую окно Ассистент ввода. Найдите нужную переменную. Если установлен флажок в поле Структурированный вид, то раскрывайте списки с помощью кнопки . Если флажок снят и переменные представлены одним большим списком, для удобства поиска воспользуйтесь фильтром</p>
<p>Примечание: *- параметр присутствует в окне ввода/редактирования элемента данных, когда привязка переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов), при условии, что не снят флажок с параметра «Не использовать IO-Mapping»</p>	

Таблица 11 – Перечень типов данных IEC-104

Тип данных	ID типа данных	Описание
M_SP_NA_1	1	Одноэлементная информация
M_DP_NA_1	2	Двухэлементная информация
M_ST_NA_1	5	Информация о положении отпак
M_BO_NA_1	7	Строка из 32 бит
M_ME_NA_1	9	Измеряемая величина, нормализованное значение
M_ME_NB_1	11	Измеряемая величина, масштабируемое значение
M_ME_NC_1	13	Измеряемая величина, формат с плавающей запятой
M_IT_NA_1	15	Интегральная сумма
M_ME_ND_1	21	Измеряемая величина, нормализованное значение без описателя качества
M_SP_TB_1	30	Одноэлементная информация с 56-битной меткой времени
M_DP_TB_1	31	Двухэлементная информация с 56-битной меткой времени
M_ST_TB_1	32	Информация о положении отпак с 56-битной меткой времени
M_BO_TB_1	33	Строка из 32-х бит с 56-битной меткой времени
M_ME_TD_1	34	Измеряемая величина нормализованное значение с 56 битной меткой времени

Тип данных	ID типа данных	Описание
M_ME_TE_1	35	Измеряемая величина масштабированное значение с 56 битной меткой времени
M_ME_TF_1	36	Измеряемая величина с плавающей запятой с 56 битной меткой времени
M_IT_TB_1	37	Интегральная сумма с 56-битной меткой времени
M_EP_TD_1	38	Информация о работе релейной защиты с 56-битной меткой времени
M_LR_NA	129*	Информация пользовательского типа LREAL из 64-х бит + байт качества, без метки времени
M_LR_TC	130*	Информация пользовательского типа LREAL из 64-х бит + байт качества с 56-битной меткой времени
M_ULINT_NA	131*	Информация пользовательского типа ULINT из 64-х бит + байт качества, без метки времени
M_ULINT_TC	132*	Информация пользовательского типа ULINT из 64-х бит + байт качества с 56-битной меткой времени

Примечание - * - ID задается пользователем

Заполненная вкладка **Данные** показана на рисунке 24.

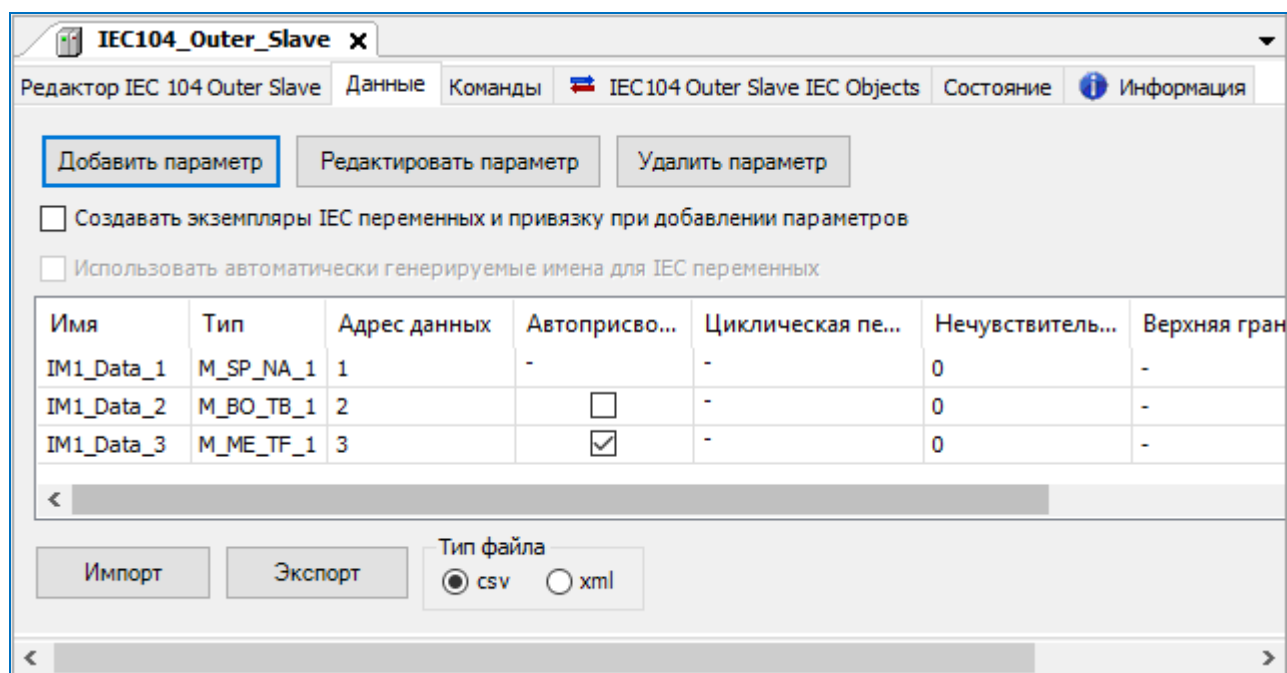


Рисунок 24 – Пример списка элементов данных

Создание списка команд, передаваемых по IEC-104

Кроме передачи данных по IEC-104 реализованы команды, используемые для установки значения какой-либо переменной, либо выполнения по сигналу каких-либо действий.

Список команд имеет ограничение в 10000 элементов для каждого устройства Slave 104 Outer Slave.

В редакторе устройства **IEC 104 Outer Slave** перейдите на внутреннюю вкладку **Команды**. Общий вид редактора команд аналогичен редактору элементов данных. Команды на данной вкладке описываются как параметры. Для добавления, изменения и удаления команд используются соответственно кнопки *Добавить параметр*, *Редактировать параметр*, *Удалить параметр*. Перейти к редактированию команды также можно двойным щелчком левой кнопкой мыши по нужной строке.

Окно ввода/редактирования команд выглядит следующим образом (Рисунок 25).

Рисунок 25 – Окно добавления новой команды

Для команды укажите значения в полях. Параметры команд представлены в таблице 12

Таблица 12 – Параметры команд IEC-104

Параметр	Описание
Имя	Наименование команды
Тип данных	Ассоциированный с этой командой тип IEC-104. Справочные типы данных IEC-104 приведены в таблице 13
Адрес объекта информации	Уникальный адрес команды
Адрес зеркального ответа	Адрес элемента данных, в который будет помещено значение, переданное командой (0 – не используется)
Время блокировки (0 – не использ.)	Команды могут быть простыми и с «выборкой перед выполнением» (Select Before Operate). При ненулевом значении это время, в течение которого команда остается в состоянии «выбрана» для выполнения (после отправки запроса select). Успешное выполнение команды (execute) возможно до истечения данного таймаута



Параметр	Описание
Таймаут выполнения	Определяет время ожидания выполнения команды со стороны управляющего узла. Задается в секундах. Если в течении данного интервала времени не был получен ответ (позитивный или негативный), то управляющий узел полагает, что произошла нештатная ситуация и помечает команду как ошибочную. Параметр <code>exes_timeout</code> актуален только для управляющего узла (Master 104 Driver)
Масштаб	Для команд, задающих масштабируемые значения
Описание параметра	Опционально, текстовое описание команды
Имя переменной*	Указывается название переменной IEC (программы контроллера), в которой хранятся передаваемые команды. Для заполнения поля Имя переменной нажмите в этом поле кнопку  , открывающую окно Ассистент ввода . Найдите нужную переменную. Если установлен флажок в поле Структурированный вид , то раскрывайте списки с помощью кнопки  . Если флажок снят и переменные представлены одним большим списком, для удобства поиска воспользуйтесь фильтром
Примечание: -* - параметр присутствует в окне ввода/редактирования команды, когда привязка переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов), при условии, что не снят флажок с параметра « Не использовать IO-Mapping »	

Таблица 13 – Перечень типов данных IEC-104

Тип данных	ID типа данных	Описание
C_SC_NA_1	45	Бинарная команда
C_SE_NC_1	50	Команда уставки (значение типа float)
C_BO_NA_1	51	Команда установки побитового регистра

Создание переменных для работы с данными/командами, передаваемыми по IEC-104

Элементы данных и команды, описанные в редакторе IEC-104, в терминологии среды разработки являются **каналами ввода-вывода**. Для реализации обмена данными необходимо создать эти каналы (см. предыдущие разделы), создать в программном коде переменные специальных типов, после чего связать переменные с каналами ввода-вывода.

Требуемые типы переменных (функциональные блоки) описаны в библиотеке PsIECCommon. Эта библиотека, а также использующая ее библиотека PsIoDrvIec104Master автоматически подключаются при добавлении устройства Master 104 Driver.

Для данных используются следующие функциональные блоки: `bo_tb_fb`, `ep_td_fb`, `it_tb_fb`, `me_tf_fb`, `sp_tb_fb`, `me_td_fb`. В наименовании функционального блока фактически указан тип данных IEC-104, например, блоку `bo_tb_fb` соответствует тип `M_BO_TB_1`, а `me_tf_fb` соответствует `M_ME_TF_1`. Количество функциональных блоков меньше количества типов

данных IEC-104, каждый функциональный блок может использоваться сразу для нескольких типов.

При создании списка элементов данных IEC-104 Outer Slave используются те же самые функциональные блоки, которые описаны в разделе «Настройка контроллера в качестве Slave. Создание переменных для работы с данными/командами, передаваемыми по IEC-104» (таблица 3). Описание этих функциональных блоков и примеры создания переменных также приведены в вышеуказанном разделе.

Предусмотрена возможность программно управлять и получать информацию об уставленном соединении основной/резервный с помощью следующих методов:

IEC104_Outer_Slave.base_connected() – true, если основное соединение установлено;

IEC104_Outer_Slave.reserv_connection() – true, если резервное соединение установлено;

IEC104_Outer_Slave.close_all_connections() – закрыть основное и резервное соединения;

IEC104_Outer_Slave.close_base_connection() – закрыть основное соединение;

IEC104_Outer_Slave.close_reserv_connection() – закрыть резервное соединение.

Также предусмотрена возможность получать статус соединения (true, если соединение установлено, иначе – false):

```
linked : BOOL
-----
linked := IEC104_Outer_Slave.Link();
-----
```

Для команд используются следующие функциональные блоки: active_sc_fb, active_se_nc_fb, active_bo_fb.

Основные свойства функциональных блоков, описывающих команды, следующие:

- **value** – данному свойству присваивается значение, которое будет передаваться командой. Используемый тип значения (имеется в виду простой тип – int, real, bool...) будет зависеть от типа функционального блока. Свойство *value* принимает значение длиной не более 4 байт;
- **m_timestamp** – в случае, когда команда имеет тип, включающий в себя метку времени, данное поле содержит метку времени, передаваемую в составе команды. Для команд без метки времени данное поле не используется.

Привязка переменных программы к элементам данных и командам

Начиная с версии СПО 1.5.7.0, чтение/запись привязанных переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов), при условии, что не снят флажок с параметра «**Не использовать Ю-Mapping**» (параметр активирован).

Если IEC устройство содержит параметры старого формата, для которых применяется механизм I/O Mapping (соотнесение входов/выходов), то их можно обновить до нового формата с использованием прямой адресации переменных с помощью кнопки *Обновить параметры* (Рисунок 26). Кнопка появляется на вкладке **Данные** или **Команды** только в том случае, если активирован параметр «**Не использовать IO-Mapping**» и обнаружены параметры старого формата. Такие параметры расположены в другом диапазоне ID.

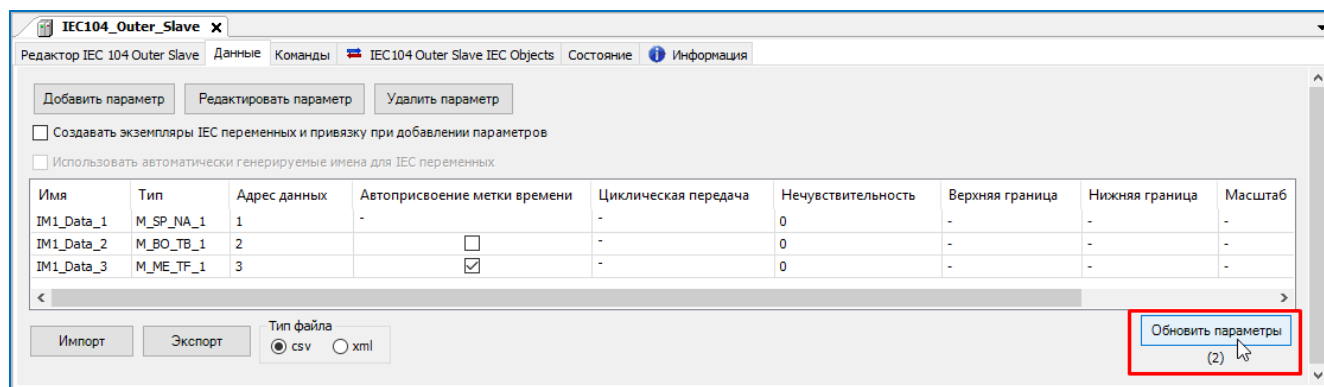


Рисунок 26 - Обновление параметров до нового формата

При активированном параметре «**Не использовать IO-Mapping**» в редакторе устройства **IEC 104 Outer Slave** будет отсутствовать внутренняя вкладка **Соотнесение входов/выходов**.



ВНИМАНИЕ!

Рекомендуется не деактивировать параметр «**Не использовать IO-Mapping**»

Если деактивировать параметр (снять флажок), необходимо будет закрыть основную вкладку устройства **IEC 104 Outer Slave**, а затем заново открыть и при добавлении данных/команд автоматически появится вкладка **Соотнесение входов/выходов** (Рисунок 27).

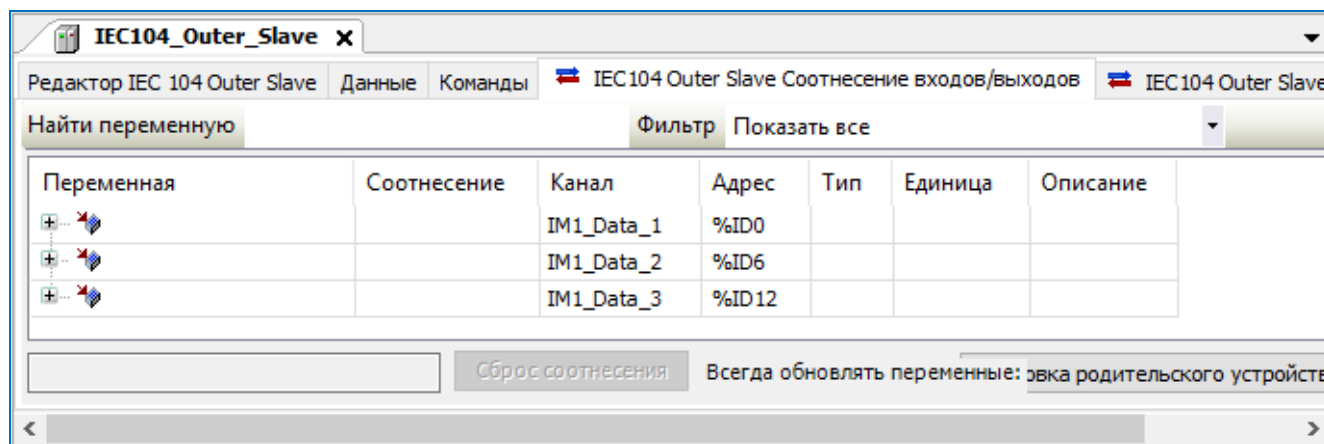


Рисунок 27 – IEC 104 Outer Slave Соотнесение входов/выходов

На этой вкладке представлен список каналов ввода-вывода (в терминологии среды разработки), ассоциированных с устройством. Когда пользователь создает элемент данных или команду в соответствующем редакторе (вкладка **Данные**, вкладка **Команды**), этот параметр

автоматически появляется здесь в виде канала вывода (для данных) или канала ввода (для команд). Наименование добавленного параметра указано в колонке **Канал**.

Для того, чтобы значения, приходящие/отправляемые по каналу, были доступны в программе контроллера, нужно привязать канал к переменной программы. Дважды щелкните левой кнопкой мыши в строке нужного канала. Появится курсор (можно вручную ввести имя переменной, семантика имен описана ниже) и кнопка **...**, открывающая окно **Ассистент ввода** (Рисунок 28).

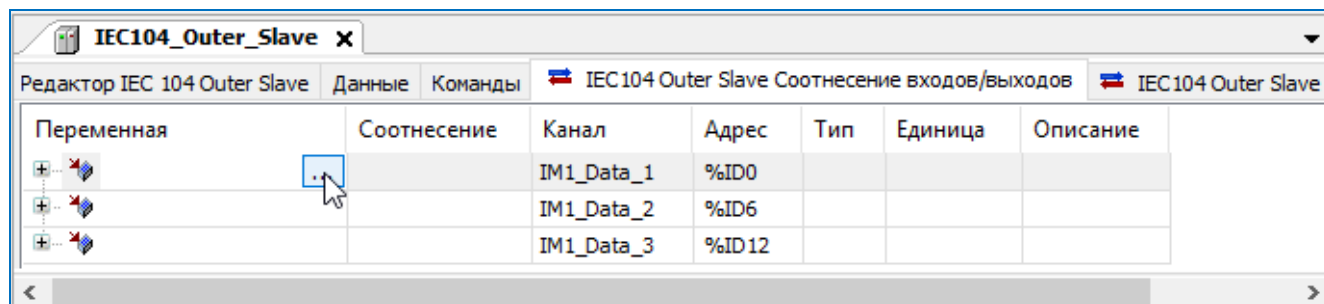


Рисунок 28 – Ручной ввод переменной или вызов ассистента ввода

В окне **Ассистент ввода** (Рисунок 29) найдите нужную переменную. Если установлен флажок в поле **Структурированный вид**, то раскрывайте списки с помощью кнопки **+**. Если флажок снят и переменные представлены одним большим списком, для удобства поиска воспользуйтесь фильтром.

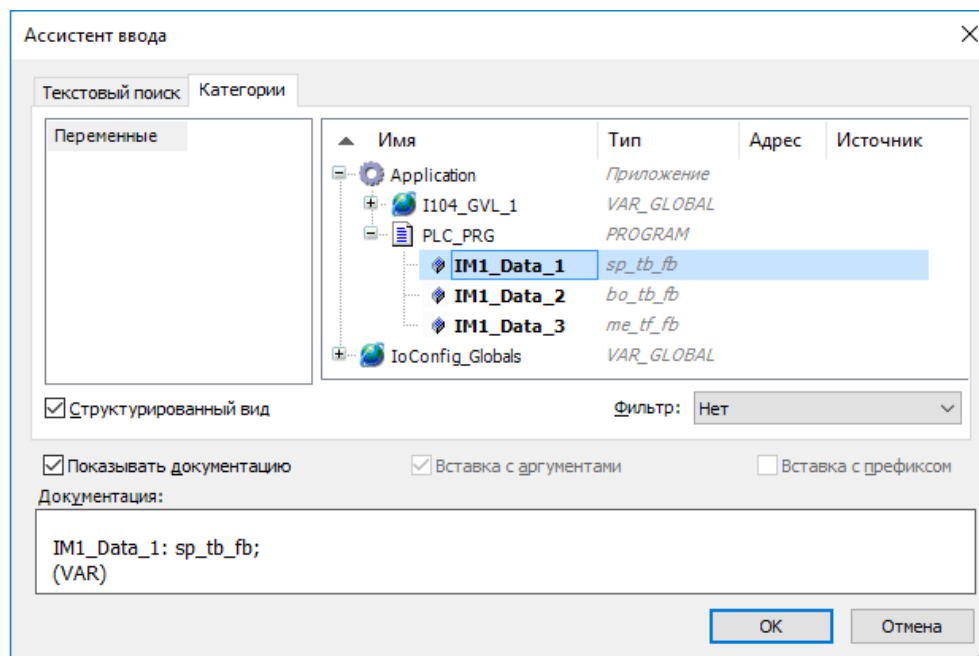


Рисунок 29 – Диалоговое окно «Ассистент ввода»

После выбора переменной нажмите кнопку **OK**, закроется окно **Ассистент ввода**, а переменная появится на вкладке **Соотнесение входов/выходов** (Рисунок 30).

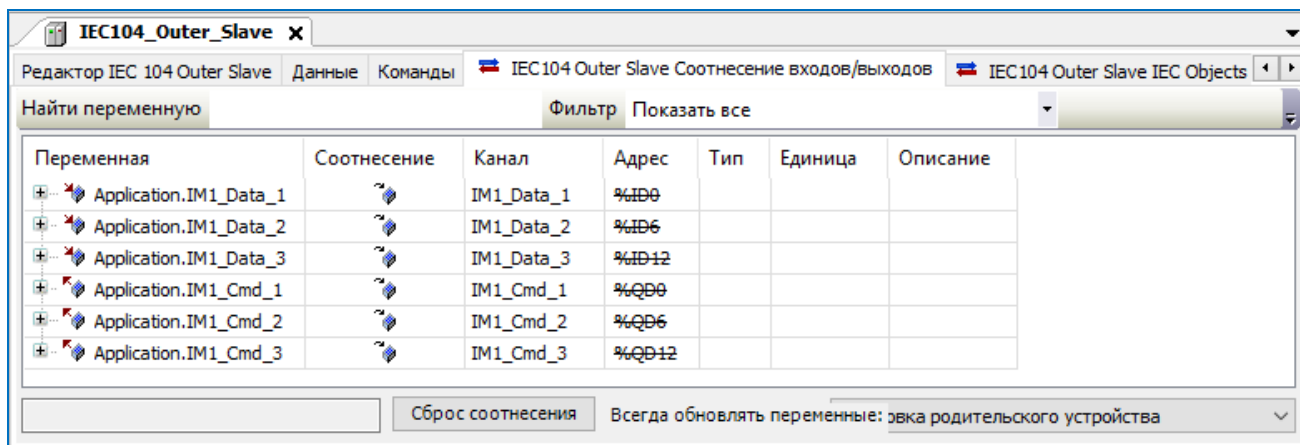


Рисунок 30 – Переменные привязаны к каналам вывода

При ручном вводе семантика имен следующая (угловые скобки при вводе НЕ используются):
 <Имя приложения>. <Имя программного юнита>. <Имя переменной>

Как ранее указано, для привязки к каналу IEC-104 используются только переменные, представляющие из себя функциональные блоки определенных типов, описанных в библиотеке PsIecCommon.



ИНФОРМАЦИЯ

Тип данных IEC-104 (например, M_VO_NA_1) задается при описании элемента данных или команды. После «привязки» переменной приложения к соответствующему каналу, переменная (типа vo_tb_fb) будет передаваться по протоколу IEC-60870-5-104(101) с использованием типа M_VO_NA_1

Все функциональные блоки, используемые для описания переменных IEC-104, имеют поле метки времени. Но в случаях, когда тип данных IEC-104, указанный в описании элемента данных или команды, не имеет метки времени (например, M_VO_NA_1), поле **Метка времени** функционального блока просто не используется. Также в зависимости от типа данных IEC-104 поле **Метка времени** может сериализоваться в 56-битный или в 24-битный timestamp.

Установка резервных бит в метке времени

Имеется возможность передачи резервных битов в timestamp (IEC 60870-5-4 п 6.8 COMPOUND INFORMATION ELEMENTS CP56Time2a). Это осуществляется путем установки старших битов в компоненты переменной типа IEC_LIB.timestamp_type.

Octet 3 биты IV, Res1 можно установить с помощью
 ts.m_minute := ts.m_minute OR 16#C000; // установлены 2 бита

Octet 4 биты SU, RES2 можно установить с помощью
 ts.m_hour := ts.m_hour OR 16#E000; // установлены 3 бита

Octet 5 биты Day of week можно установить с помощью
 ts.m_day := ts.m_day OR 16#E000; // установлены 3 бита

Octet 6 биты RES3 можно установить с помощью
 ts.m_month := ts.m_month OR 16#F000; // установлены 4 бита

Octet 7 бит RES1 можно установить с помощью


```
ts.m_year := ts.m_year OR 16#8000; // установлен 1 бит.
```

Автоматическая генерация переменных и привязка к каналам ввода-вывода

При добавлении параметров на вкладках **Данные** и **Команды** можно активировать режим, при котором для каждого канала автоматически создаются переменные ПЛК-программы и формируется привязка переменной к каналу. Для этого при создании нового элемента данных или команды установите флажок в поле **Создавать экземпляры IEC переменных и привязку при добавлении параметров** (Рисунок 31).

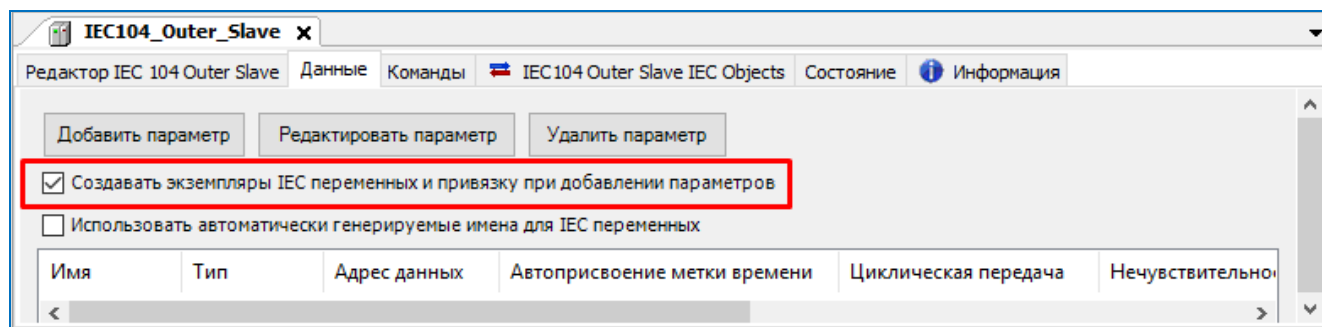


Рисунок 31 – Установка флажка в поле «Создавать экземпляры IEC переменных и привязку при добавлении параметров»

В этом режиме при добавлении канала для устройства **IEC 104 Outer Slave** создается собственный список глобальных переменных с именем вида I104_GVL_X, где X – условный порядковый номер устройства, начиная с единицы (Рисунок 32). В этом списке будет создана переменная соответствующего типа.

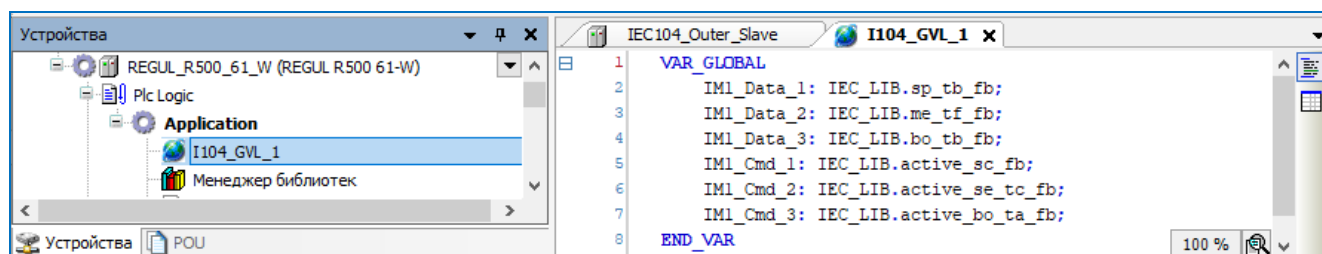


Рисунок 32 – Пример описания переменных в списке I104_GVL_1

При наличии вкладки **Соотнесение входов/выходов** новая переменная будет автоматически привязана к новому каналу (Рисунок 33). Наименование переменной будет совпадать с именем канала (пробелы заменяются на подчеркивания).

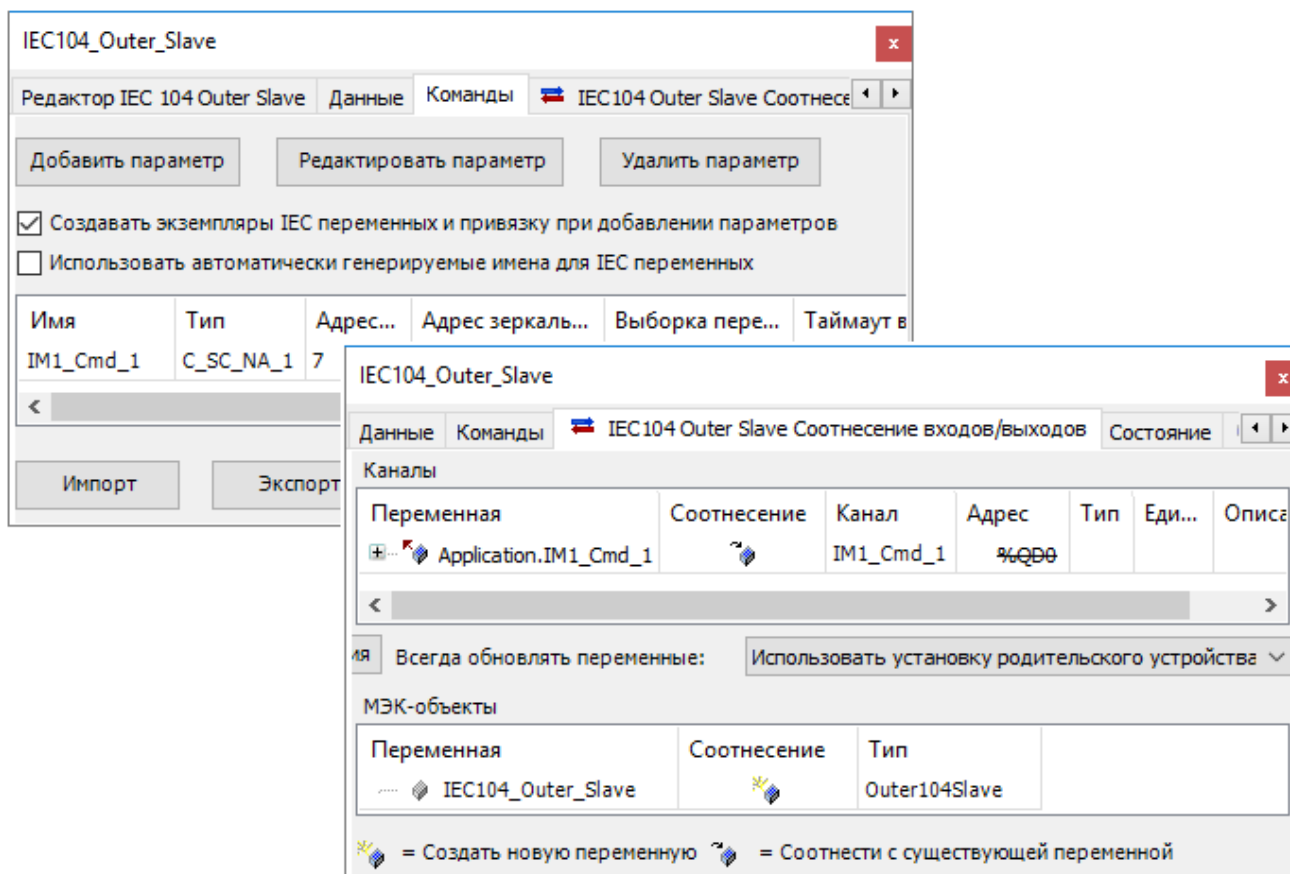


Рисунок 33 – Автоматическое создание переменной и автоматическая привязка ее к каналу

При установленном флажке в поле **Использовать автоматически генерируемые имена для IEC переменных** имена переменных будут создаваться на основе шаблона: *iec<случайное число>_data_fb*. При отсутствии флажка в этом поле имена генерируемых переменных имеют вид: *IS<X>_Data_<N>* (для элементов данных), *IS<X>_Cmd_<N>* (для команд), где X – номер Slave-устройства, N – нумератор переменных. Генерируемые по умолчанию имена каналов данных и команд совпадают с именами связываемых переменных.

Возможность разрыва соединения со стороны IEC-программы

Для контроллера в качестве Master добавлена функциональность, позволяющая IEC-приложению управлять разрывом соединения. Когда программа определяет, что обмен данными по протоколу IEC-104 осуществляется некорректно, выполняется команда, которая разрывает основное соединение и происходит переключение на резервный канал. Попытка соединения по основному каналу состоится через некоторый промежуток времени (*reconnect_interval*). Если она будет успешной, то резервное соединение закроется и обмен продолжится по основному каналу.

The screenshot displays the SIMATIC Manager interface for a project named 'use_master104'. The main window shows the ladder logic for 'PLC_PRG'. The code includes an IF statement for 'do_disconnect' which is currently FALSE. Below the code, the Watch window shows the state of two connection-related variables:

Expression	Application	Type	Value	Prepared value	Execution point
IEC104_Outer_Slave.m_base_connection.m_state	REGUL_R500_51_61.Application_use_master104	CONNECTION_104_STATE_E	ready_for_data		Cyclic Monitoring
IEC104_Outer_Slave.m_reserv_connection.m_state	REGUL_R500_51_61.Application_use_master104	CONNECTION_104_STATE_E	disconnected		Cyclic Monitoring

Рисунок 34 – Разрыв соединения со стороны программы

НАСТРОЙКА IEC-101

Добавление последовательного порта в конфигурацию контроллера

Перед настройкой IEC-101 в конфигурацию контроллера необходимо добавить последовательный порт.

Добавление порта

Для обмена данными будет использоваться один из портов: COM1 (интерфейс RS-232) или COM2 (интерфейс RS-485) на модуле центрального процессора, либо один из портов на модуле коммуникационного процессора (интерфейс RS-485).

Добавление последовательного порта модуля центрального процессора

Если для обмена данными будет использоваться порт модуля центрального процессора, то в конфигурацию необходимо добавить к контроллеру устройство **Regul Serial Port**. Для этого в окне дерева устройств поместите курсор на головное устройство (Device), нажмите правую кнопку мыши. В появившемся контекстном меню выберите пункт **Добавить устройство...** Откроется окно **Добавить устройство**, в котором выберите *Regul* → *Последовательный порт* → *Regul Serial Port*. Нажмите кнопку **Добавить устройство** или дважды щелкните левой кнопкой мыши. Выбранное устройство появится в проекте в дереве устройств (Рисунок 35).

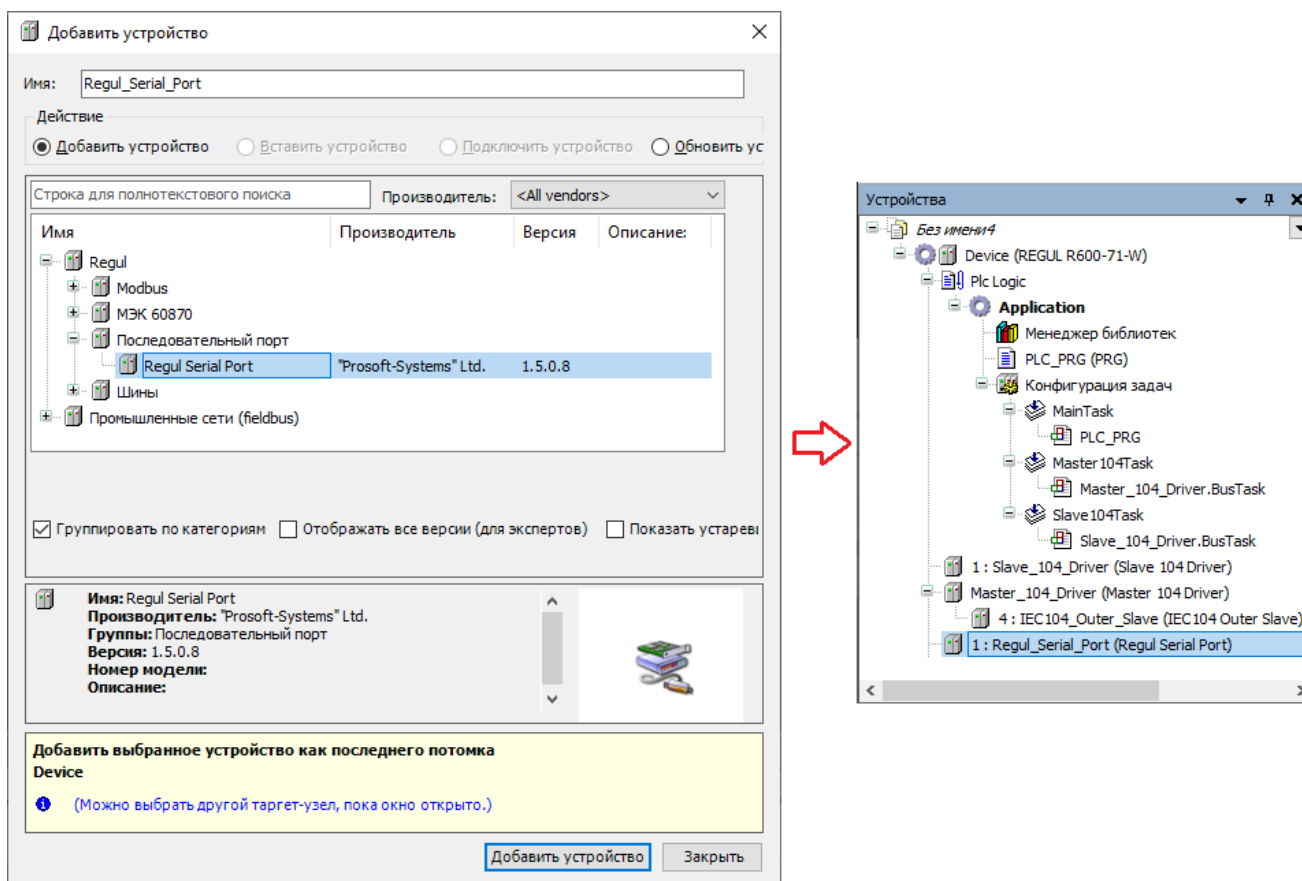


Рисунок 35 – Добавление последовательного порта Regul Serial Port

Двойным щелчком по названию **Regul Serial Port** откроется вкладка параметров (Рисунок 36).

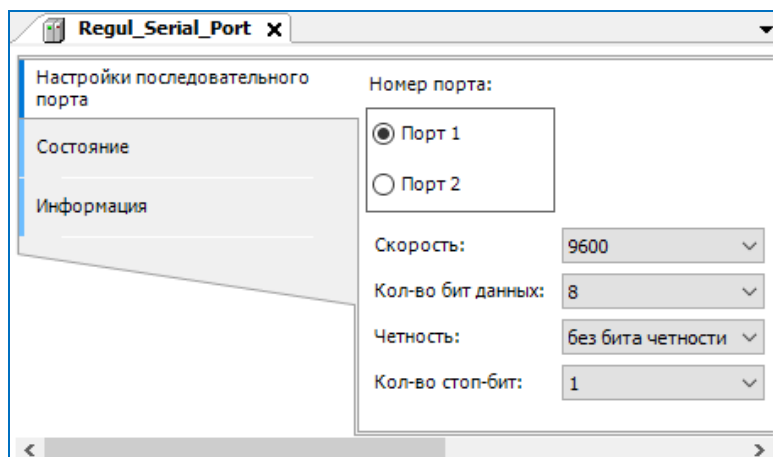


Рисунок 36 – Настройки последовательного порта Regul Serial Port

Добавление последовательного порта коммуникационного модуля

Если для обмена данными будет использоваться порт модуля коммуникационного процессора, то в конфигурацию необходимо добавить к модулю коммуникационного процессора устройство **Extended Regul Serial Port** (*Regul* → *Последовательный порт* → *Extended Regul Serial Port*). Выбранное устройство появится в проекте в дереве устройств (Рисунок 37).

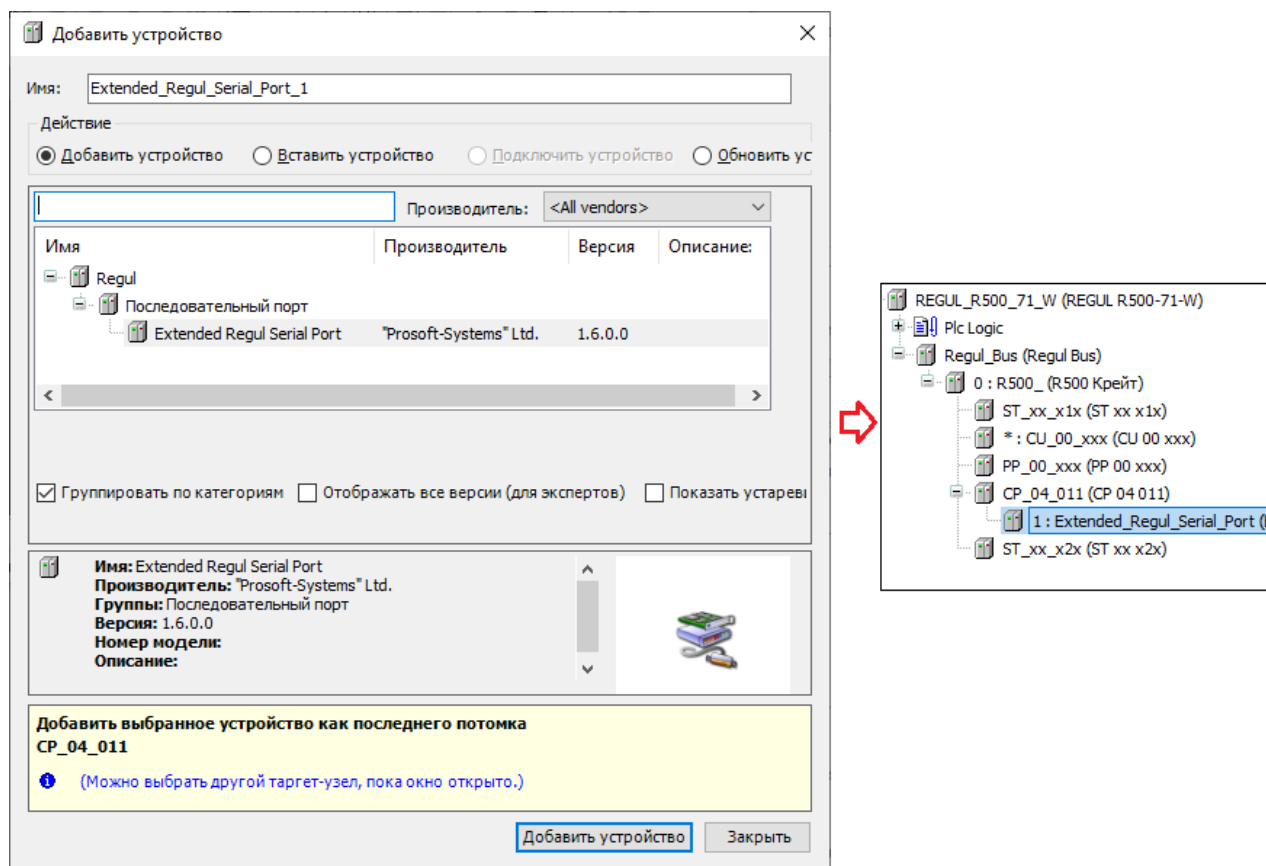


Рисунок 37 – Добавление последовательного порта Extended Regul Serial Port

Двойным щелчком по названию **Extended Regul Serial Port** откроется вкладка параметров (Рисунок 38).

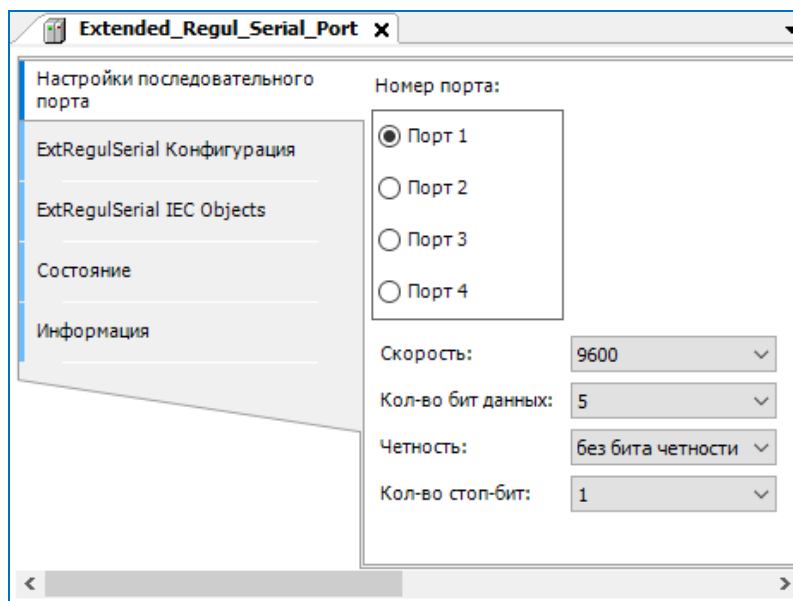


Рисунок 38 – Настройки последовательного порта Extended Serial Port

Настройка параметров порта

Откройте главную вкладку параметров порта и перейдите на внутреннюю вкладку **Настройки последовательного порта** (Рисунок 37, 38). В поле **Номер порта** выберите из списка порт модуля, к которому подключено внешнее устройство и установите переключатель в соответствующей строке. Далее, выбирая значение в раскрывающемся списке, задайте следующие параметры:

- **Скорость** – значения скорости передачи данных: 1200, 2400, 4800, 9600, 19200, 38400, 115200;
- **Кол-во бит данных** – значения: 5, 6, 7, 8;
- **Четность** – значения: без бита четности, проверка на нечетность, проверка на четность;
- **Кол-во стоп-бит** – 1 или 2.

Настройка контроллера в качестве Slave 101

Добавление устройства Unbalanced Secondary 101 Driver

Добавьте устройство **Unbalanced Secondary 101 Driver** к последовательному порту **Regul Serial Port** или **Extended Regul Serial Port** (*Regul* → *МЭК 60870* → *Serial 60870-101 Slave* → *Unbalanced Secondary 101 Driver*). Выбранное устройство появится в проекте в дереве устройств (Рисунок 39).

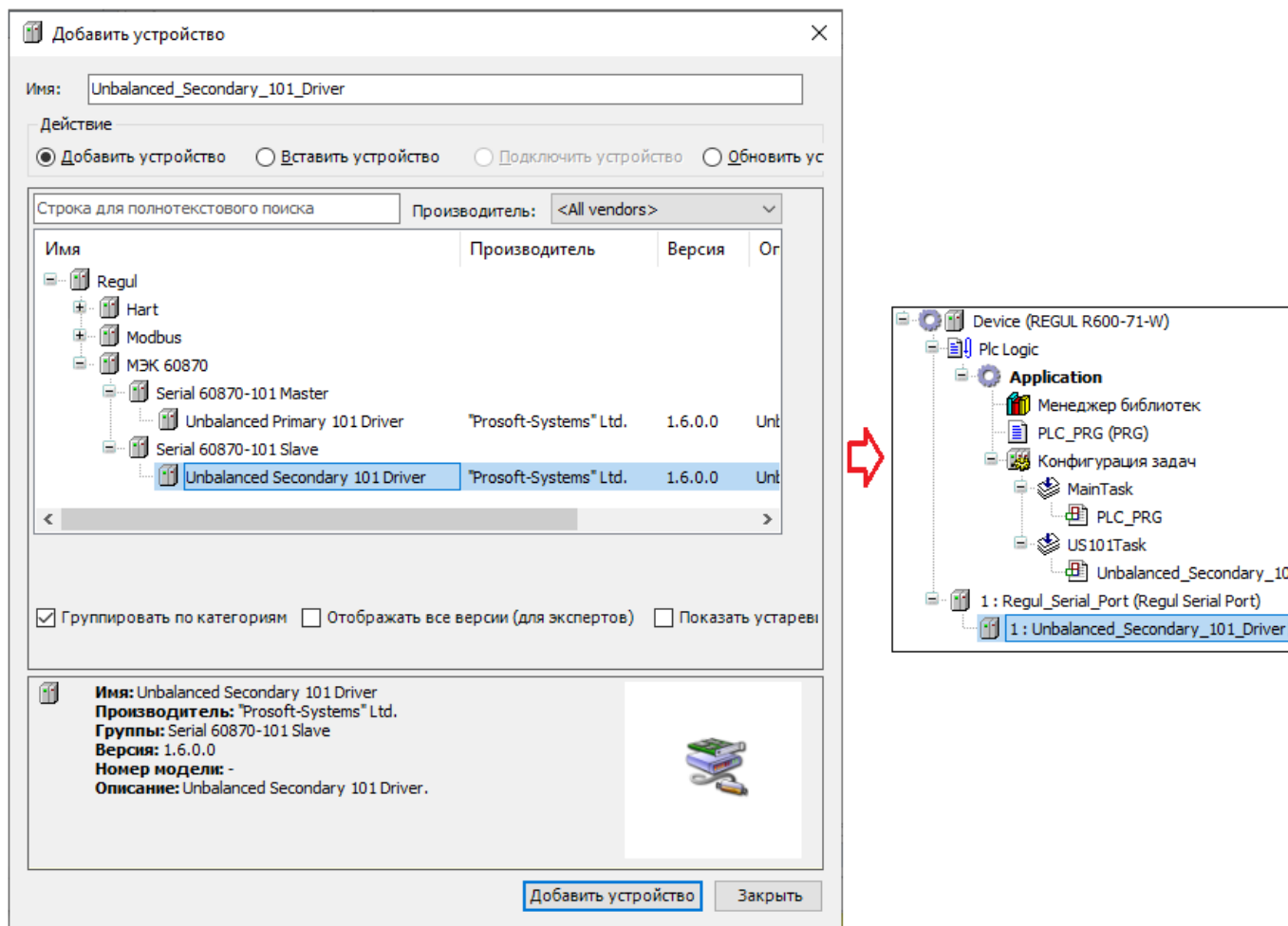


Рисунок 39 – Добавление в конфигурацию контроллера устройство Unbalanced Secondary 101 Driver

Двойным щелчком по названию устройства **Unbalanced Secondary 101 Driver** откройте вкладку параметров. По умолчанию открывается первая внутренняя вкладка **Редактор IEC 101 Unbalanced Secondary** (Рисунок 40).

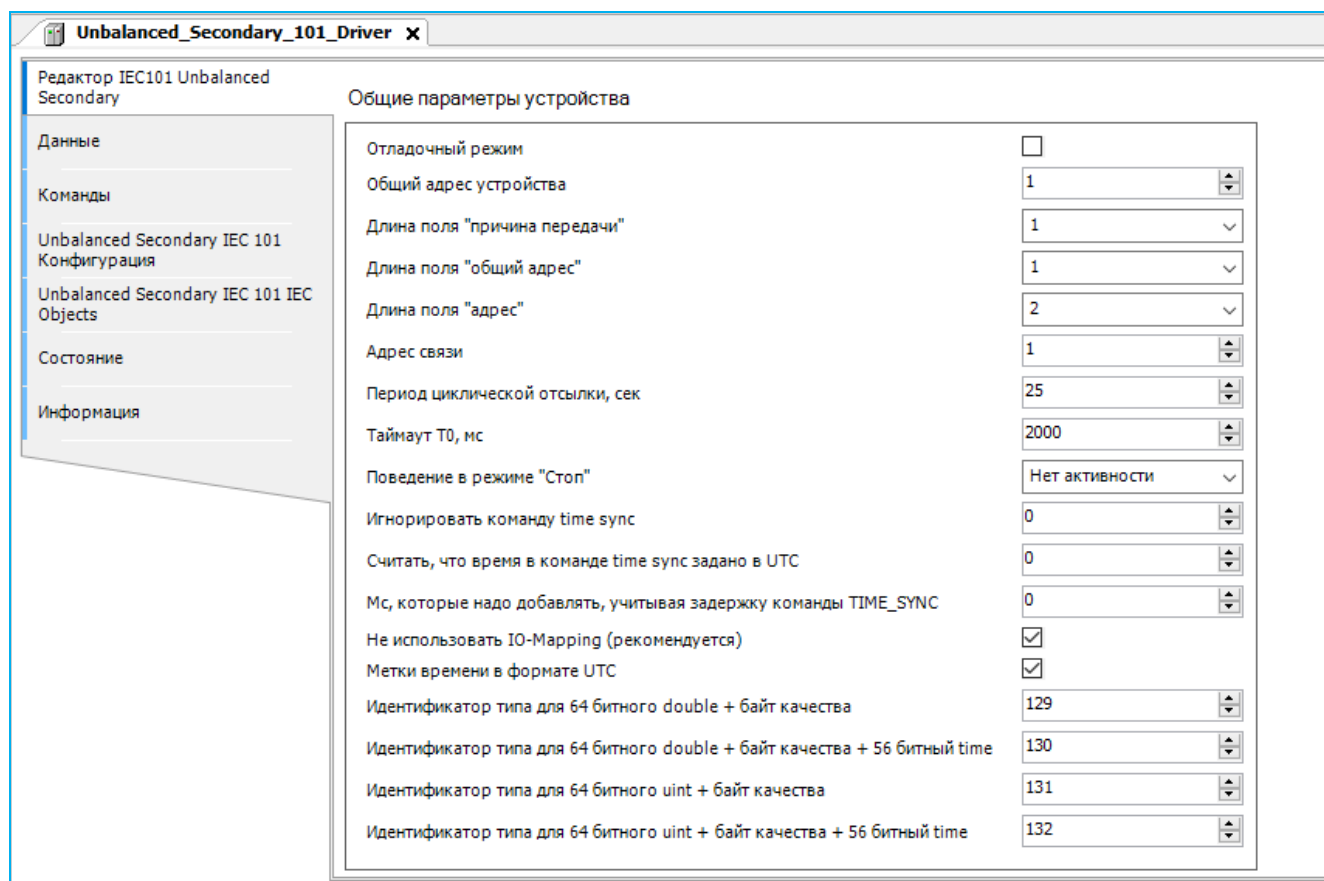


Рисунок 40 – Редактор устройства IEC101 Unbalanced Secondary

Настройка общих параметров устройства Unbalanced Secondary 101 Driver

В редакторе устройства (Рисунок 40) в блоке **Общие параметры устройства** доступны для настройки параметры, представленные в таблице 14.

Таблица 14 – Общие параметры устройства

Параметр	Описание	Значение по умолчанию
Отладочный режим	При установке флажка <input checked="" type="checkbox"/> в поле, все детали обмена по IEC-101 будут записаны в журнал работы контроллера, а именно передаваемые и принимаемые пакеты, комментарии к возникающим ошибкам и т.п.	
Общий адрес устройства	Содержит общий адрес (COMMON ADDRESS, ASDU ADDRESS) устройства, все данные и команды имеют общую часть, равную значению этого параметра (см. спецификацию IEC-101)	1
Длина поля «причина передачи»	Содержит длину поля «причина передачи» (COT, Cause Of Transmission). Допустимые значения параметра: 1 или 2	1
Длина поля «общий адрес», Длина поля «адрес»	Характеристики PDU (Protocol Data Unit – протокольная единица, пакет), передаваемых по IEC-101, должны быть одинаковы для пары master-slave. Допустимые значения параметра «общий адрес»: 1 или 2. Допустимые значения	«общий адрес» - 1, «адрес» - 2

Параметр	Описание	Значение по умолчанию
	параметра «адрес»: 2 или 3	
Адрес связи (Link Address)	Уникальный адрес для процедур обмена данными канального уровня	1
Период циклической отсылки, сек	При описании элемента данных IEC-101 можно включить его в циклическую рассылку (установка флажка в поле Участствует в циклической рассылке). Для всех таких элементов их значения даже при отсутствии изменений будут отсылаться с указанным здесь периодом (в секундах)	25
Таймаут T0, мс	Интервал времени в миллисекундах. Определяет максимальную задержку ответа на запрос. Если задержка превышает данный интервал, то соединение считается потерянным	2000
Поведение в режиме «Стоп»*	Определяет поведение компонента при остановке программы. <i>Нет активности</i> - означает, что устройство Unbalanced Secondary 101 Driver прекращает все соединения и не воспринимает новые; <i>Нормальная работа</i> - означает продолжение работы в обычном режиме	Нет активности
Игнорировать команду time sync	Если значение не равно 0, то при получении команды синхронизации времени (C_CS_NA_1), синхронизации времени не происходит. Если этот параметр равен 0, то команда синхронизации времени выполняется	0
Считать, что время в команде time sync задано в UTC	Если в этом поле задано значение 1, то, при синхронизации времени, время в составе команды будет интерпретировано, как UTC. Если задано значение 0, то, это время будет интерпретировано, как локальное время	0
мс, которые надо добавлять, учитывая задержку команды TIME_SYNC	Корректировка команды синхронизации времени	0
Не использовать IO-Mapping (рекомендуется)	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что чтение/запись привязанных переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов)	
Метки времени в формате UTC	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что значение метки времени соответствует UTC, иначе - локального времени	
Идентификатор типа для 64 битного double + байт качества	Задание ID для пользовательского типа LREAL(double), без метки времени	129

Параметр	Описание	Значение по умолчанию
Идентификатор типа для 64 битного double + байт качества + 56 битный time	Задание ID для пользовательского типа LREAL(double), с меткой времени	130
Идентификатор типа для 64 битного uint + байт качества	Задание ID для пользовательского типа ULINT(uint), без метки времени	131
Идентификатор типа для 64 битного uint + байт качества + 56 битный time	Задание ID для пользовательского типа ULINT(uint), с меткой времени	132
<p>Примечание -*- предусмотрена возможность самостоятельно активировать «поведение в режиме «Стоп» в программном коде. Для активации режима требуется в программе присвоить значение TRUE свойству ActivateStopBehavior необходимого устройства:</p> <p>«IEC101USSlave_device_name».ActivateStopBehavior := TRUE;</p> <p>После этого slave-устройство перейдет в Стоп-режим работы</p>		

На вкладке **Unbalanced Secondary IEC 101 Конфигурация**, можно задать значение параметра **Interleave between request and response. Must be >0 for RS-485** (Рисунок 41). Параметр определяет время задержки при работе по RS-485, так как мгновенный ответ на запрос вызывает искажение битов.

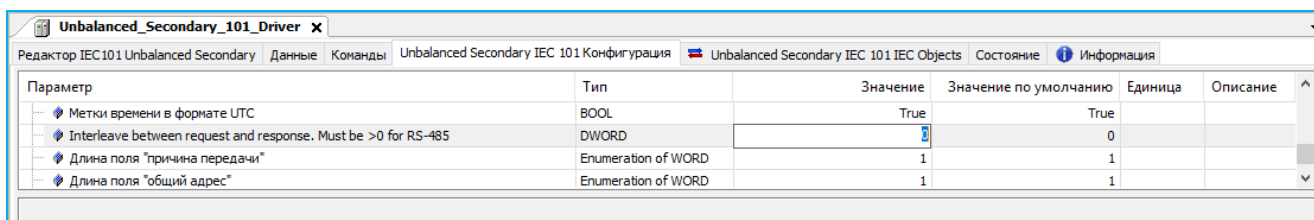


Рисунок 41 - Вкладка Unbalanced Secondary IEC 101 Конфигурация



ИНФОРМАЦИЯ

Если вкладка **Unbalanced Secondary IEC 101 Конфигурация** отсутствует, то необходимо перейти в меню **Инструменты** ⇒ **Опции** ⇒ найти пункт **Редактор устройств** и установить флажок напротив поля **Показывать общие окна конфигурации устройств**

Двойной щелчок левой кнопкой мыши по ячейке в столбце **Значение** в строке **Interleave between request and response. Must be >0 for RS-485** позволяет задать необходимое значение (по умолчанию 0). Для RS-485, если параметр = 0, возможны явления интерференции и ошибки в передаче данных.

Все последующие настройки областей данных Slave 101 и приемы работы с ними идентичны тем, что описаны в разделе **Настройка контроллера в качестве Slave 104**, начиная с подраздела [Создание списка элементов данных, передаваемых по IEC-104](#).

Настройка контроллера в качества Master 101

Добавление устройства Unbalanced Secondary 101 Driver

Добавьте устройство **Unbalanced Primary 101 Driver** к последовательному порту **Regul Serial Port** или **Extended Regul Serial Port** (*Regul → МЭК 60870 → Serial 60870-101 Master → Unbalanced Primery 101 Driver*). Нажмите кнопку **Добавить устройство** или дважды щелкните левой кнопкой мыши. Выбранное устройство появится в проекте в дереве устройств.

Далее к устройству **Unbalanced Primary 101 Driver** необходимо подключить одно или несколько slave-устройств (outer slaves), которые будут опрашиваться контроллером: *Regul → МЭК 60870 → Serial 60870-101 Master → IEC 101 UP Outer Slave* (Рисунок 42).

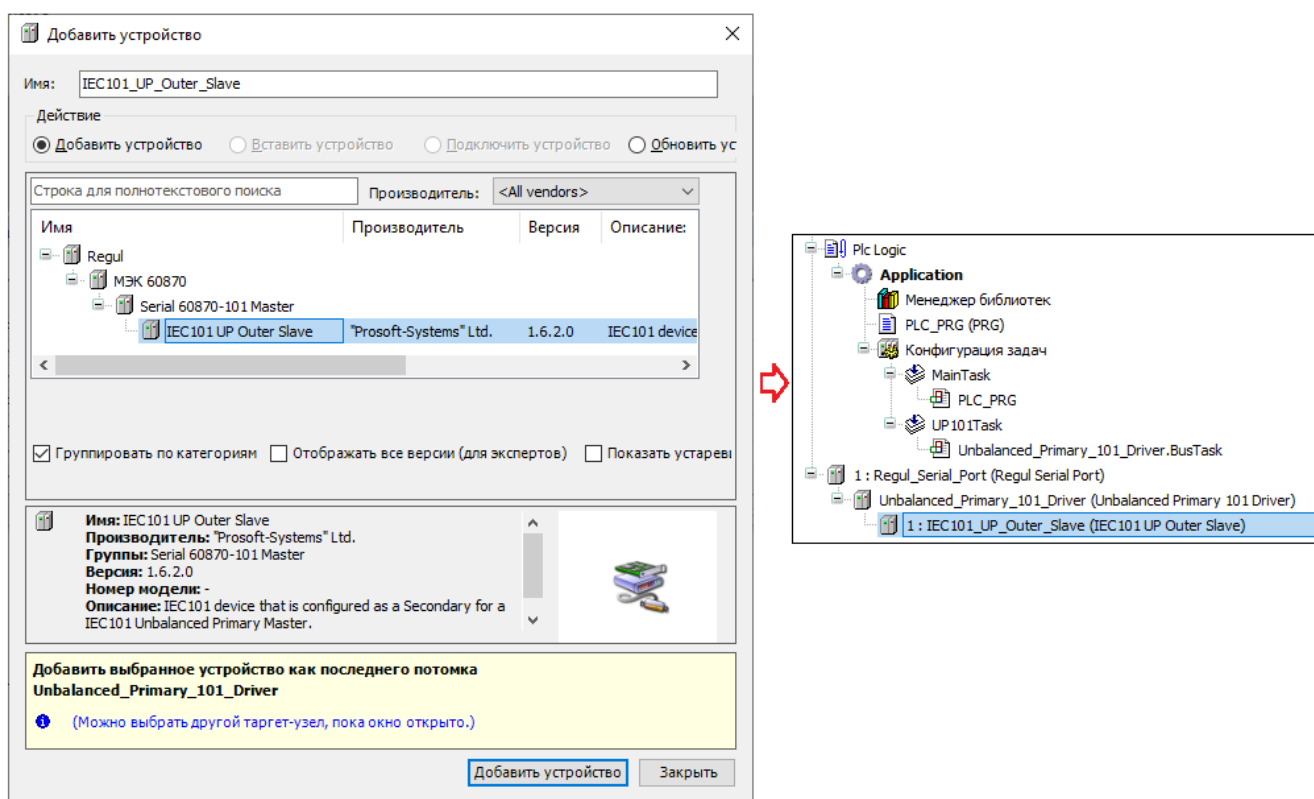


Рисунок 42 – Добавление в конфигурацию контроллера устройство IEC 101 UP Outer Slave

Двойным щелчком по названию устройства **IEC 101 UP Outer Slave** откройте вкладку параметров. По умолчанию открывается первая внутренняя вкладка **Редактор IEC 101 Outer Slave** (Рисунок 43).

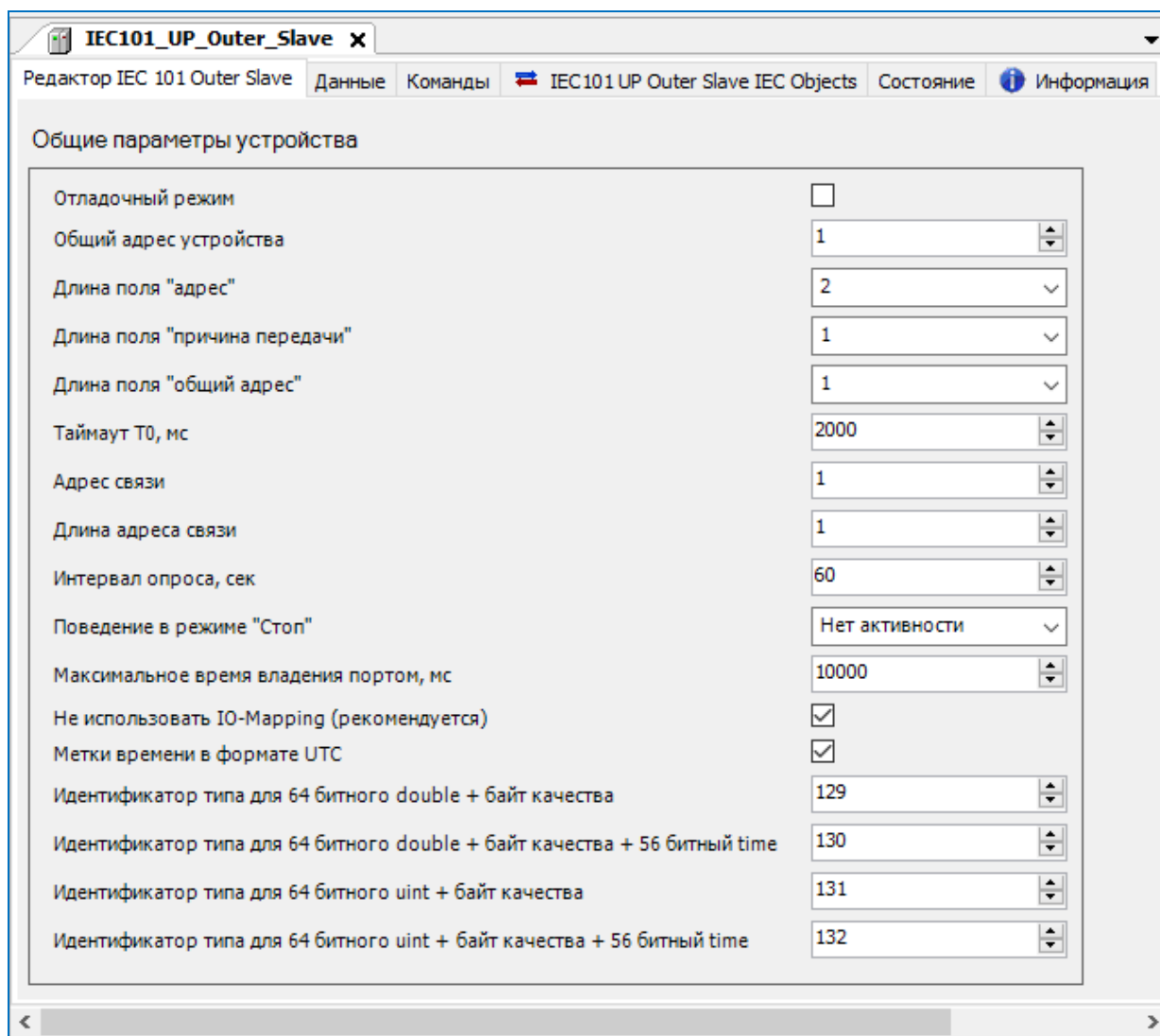


Рисунок 43 – Редактор устройства IEC 101 Outer Slave

Настройка общих параметров устройства IEC 101 Outer Slave

В редакторе устройства (Рисунок 43) в блоке **Общие параметры устройства** доступны для настройки параметры, представленные в таблице 15.

Таблица 15 – Общие параметры устройства

Параметр	Описание	Значение по умолчанию
Отладочный режим	При установке флажка <input checked="" type="checkbox"/> в поле, все детали обмена по IEC-101 будут записаны в журнал работы контроллера, а именно передаваемые и принимаемые пакеты, комментарии к возникающим ошибкам и т.п.	
Общий адрес устройства	Содержит общий адрес (COMMON ADDRESS, ASDU ADDRESS) устройства, все данные и команды имеют общую часть, равную значению этого параметра (см. спецификацию IEC-101)	1

Параметр	Описание	Значение по умолчанию
Длина поля «адрес», Длина поля «общий адрес»	Характеристики PDU (Protocol Data Unit – протокольная единица, пакет), передаваемых по IEC-101, должны быть одинаковы для пары master-slave. Допустимые значения параметра «адрес»: 2 или 3. Допустимые значения параметра «общий адрес»: 1 или 2	«адрес» – 2, «общий адрес» – 1
Длина поля «причина передачи»	Содержит длину поля «причина передачи» (COT, Cause Of Transmission). Допустимые значения параметра: 1 или 2	1
Таймаут T0, мс	Интервал времени в миллисекундах. Определяет максимальную задержку ответа на запрос. Если задержка превышает данный интервал, то соединение считается потерянным	2000
Адрес связи	Уникальный адрес одного из подчиненных вторичных устройств, которое опрашивается первичным	1
Длина адреса связи	Длина поля уникального адреса, размер: от 1 до 65535 (1 или 2 байта)	1
Интервал опроса, сек	Период отсылки команды общего опроса. Значение 0 – отсылка команды общего опроса не производится	60
Поведение в режиме «Стоп»*	Определяет поведение компонента при остановке программы. <i>Нет активности</i> - означает, что Unbalanced Primary 101 Driver прекращает взаимодействовать с соответствующим IEC 101 UP Outer Slave; <i>Нормальная работа</i> - означает продолжение работы в обычном режиме	Нет активности
Максимальное время владения портом, мс	Максимальный интервал времени обмена данными с одним устройством, в миллисекундах	10000
Не использовать IO-Mapping (рекомендуется)	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что чтение/запись привязанных переменных происходит по адресу (прямой доступ к памяти ПЛК), без использования механизма I/O Mapping (соотнесение входов/выходов)	
Метки времени в формате UTC	Флажок <input checked="" type="checkbox"/> , установленный в поле, означает, что значение метки времени соответствует UTC, иначе - локального времени	
Идентификатор типа для 64 битного double + байт качества	Задание ID для пользовательского типа LREAL(double), без метки времени	129
Идентификатор типа для 64 битного double + байт качества + 56 битный time	Задание ID для пользовательского типа LREAL(double), с меткой времени	130

Параметр	Описание	Значение по умолчанию
Идентификатор типа для 64 битного uint + байт качества	Задание ID для пользовательского типа ULINT(uint), без метки времени	131
Идентификатор типа для 64 битного uint + байт качества + 56 битный time	Задание ID для пользовательского типа ULINT(uint), с меткой времени	132

На вкладке **IEC101 UP Outer Slave Конфигурация**, можно задать значение параметров (Рисунок 44):

- **Мин. интервал между PDU, мс.** Параметр определяет минимальный интервал времени между PDU, в случае, когда используется RS485 (по умолчанию **0**, рекомендуемое значение для RS485: **5...20**);
- **Количество повторов PDU.** Параметр определяет количество повторов отправки запроса, в случае, когда ответ от вторичного устройства отсутствует, либо содержит ошибку (по умолчанию **3**).

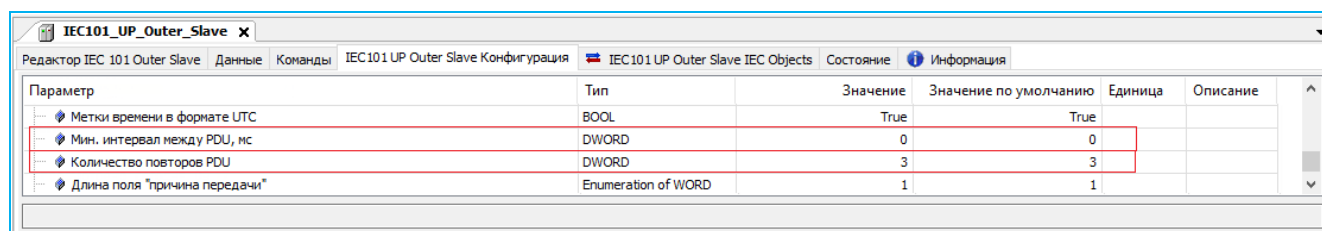


Рисунок 44 - Вкладка **IEC101 UP Outer Slave Конфигурация**



ИНФОРМАЦИЯ

Если вкладка **IEC101 UP Outer Slave Конфигурация** отсутствует, то необходимо перейти в меню **Инструменты** ⇒ **Опции** ⇒ найти пункт **Редактор устройств** и установить флажок напротив поля **Показывать общие окна конфигурации устройств**

Двойной щелчок левой кнопкой мыши по ячейке в столбце **Значение** в строке **Количество повторов PDU/ Мин. интервал между PDU, мс** позволяет задать необходимое значение.

Все последующие настройки областей данных Master 101 и приемы работы с ними идентичны тем, что описаны в разделе **Настройка контроллера в качестве Master 104**, начиная с подраздела [Создание списка элементов данных, передаваемых по IEC-104](#).

Для получения статуса соединения Master IEC 101 (true, если соединение установлено, иначе – false) необходимо прописать, например:

```
IEC101_UP_Outer_Slave
IEC101_UP_Outer_Slave_1
-----
```

```
linked : BOOL;  
linked2 : BOOL;  
-----  
linked := IEC101_UP_Outer_Slave.Link();  
linked2 := IEC101_UP_Outer_Slave_1.Link();
```

ЭКСПОРТ И ИМПОРТ КОНФИГУРАЦИИ IEC-104(101)

Для упрощения создания списка каналов и их привязки к переменным программы предусмотрена возможность импорта/экспорта конфигурации IEC-104 (101).

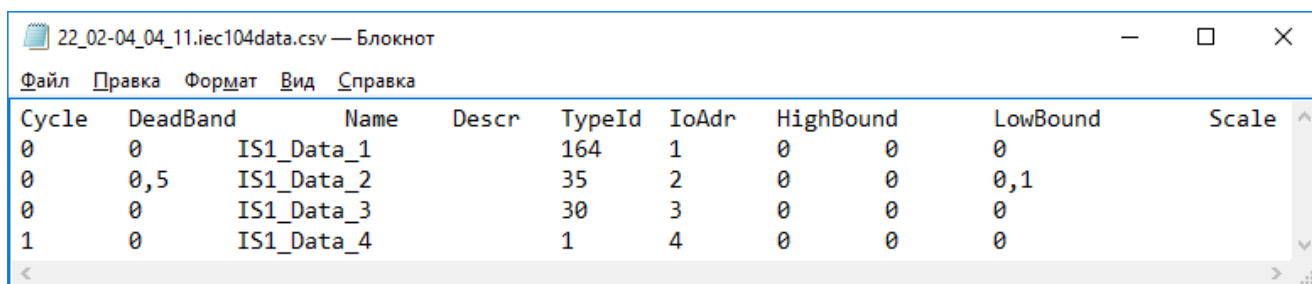
Конфигурацию контроллера в качестве Slave, экспортированную в файл, можно импортировать и использовать при настройке контроллера в качестве Master. При этом требуется следить за уникальностью имен сгенерированных переменных. В противном случае, т.е. когда имена переменных не уникальны, происходит следующее:

- при настройке контроллера в качестве Slave пользователь экспортирует список IEC-104(101) - переменных,
- затем в этом проекте пользователь настраивает контроллер в качестве Master и использует список переменных для импорта,
- в результате процедуры импорта ранее созданные переменные с теми же именами будут стерты, и заменены новыми, характерными для мастера.

Экспорт/импорт списка элементов данных

В редакторе устройства на вкладке **Данные** создайте список элементов данных.

В нижней части окна в блоке **Тип файла** поставьте переключатель на значение **CSV**. Нажмите кнопку **Экспорт**. Откроется окно **Save iec 104(101) settings**. Определите папку, в которой будет храниться файл конфигурации IEC-104(101), задайте ему имя, нажмите кнопку **Сохранить**. Информация будет сохранена в файл формата csv (текстовый формат), где все поля отделяются символом табуляции (ТАВ), первая строка обязательно представляет имена столбцов, все столбцы обязательны. Если в описании элемента данных для какого-либо столбца ничего не указано, то получается два символа ТАВ, идущие подряд.



Cycle	DeadBand	Name	Descr	TypeId	IoAdr	HighBound	LowBound	Scale
0	0	IS1_Data_1		164	1	0	0	0
0	0,5	IS1_Data_2		35	2	0	0	0,1
0	0	IS1_Data_3		30	3	0	0	0
1	0	IS1_Data_4		1	4	0	0	0

Рисунок 45 – Структура файла конфигурации для экспорта/импорта данных на примере IEC-104

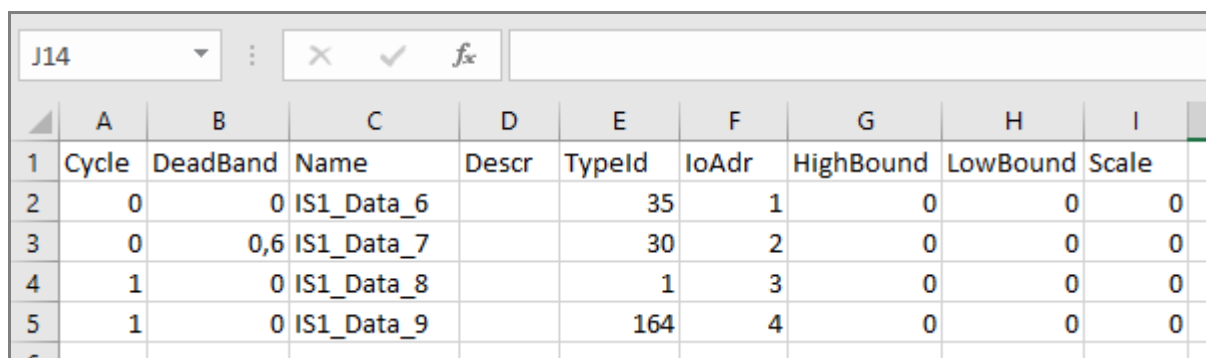
В файле конфигурации присутствуют следующие поля:

- **Cycle** – участвует ли в циклической рассылке (значение 0 – нет, 1 – да);
- **DeadBand** – полоса нечувствительности (если применима к данному типу, значение 0 – не используется);

- **Name** – наименование канала (и соответствующей переменной, в режиме автоматической генерации);
- **Descr** – описание канала (недопустим символ табуляции в описании, пустое значение допустимо);
- **TypeId** – идентификатор типа согласно протоколу IEC-60870-5-104(101). Для типов с меткой времени можно указать автоматическую генерацию временных меток, прибавив к идентификатору типа 128 (80 hex);
- **IoAdr** – уникальный адрес элемента согласно протоколу IEC-60870-5-104(101);
- **HighBound** – верхняя граница для диапазонных типов данных (0 – не используется);
- **LowBound** – нижняя граница для диапазонных типов данных (0 – не используется);
- **Scale** – масштаб, для масштабируемых величин (0 – не используется).

Для импорта конфигурации из файла в проект выберите тип файла, далее нажмите кнопку **Импорт**. Откроется окно **Import iec 104(101) settings**. Выберите на компьютере нужный файл, нажмите кнопку **Открыть**. На вкладке **Данные** появится новый список элементов данных.

Файл с описанием элементов данных не обязательно должен быть создан в проекте и экспортирован. Пользователь может самостоятельно создать файл формата Tab delimited с помощью электронных таблиц, например, Microsoft Excel. В электронной таблице необходимо создать и заполнить поля в соответствии со структурой файла (описана выше). Далее сохранить файл с расширением csv.



	A	B	C	D	E	F	G	H	I
1	Cycle	DeadBand	Name	Descr	TypeId	IoAdr	HighBound	LowBound	Scale
2	0	0	IS1_Data_6		35	1	0	0	0
3	0	0,6	IS1_Data_7		30	2	0	0	0
4	1	0	IS1_Data_8		1	3	0	0	0
5	1	0	IS1_Data_9		164	4	0	0	0

Рисунок 46 – Создание в электронной таблице списка переменных на примере IEC-104 для последующего импорта в проект

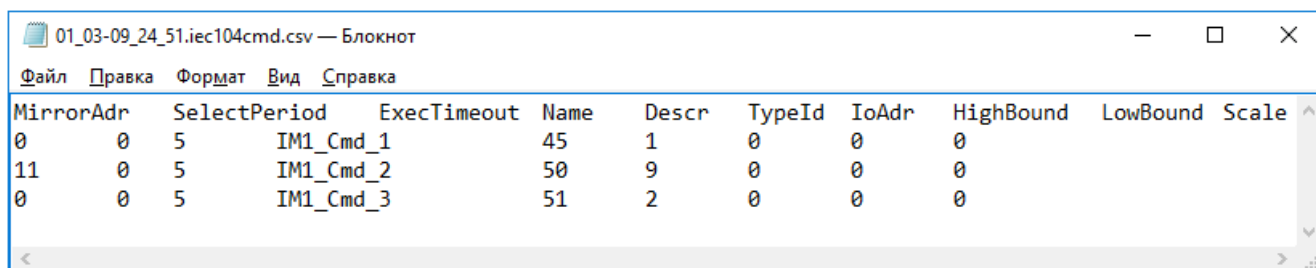
При экспорте конфигурации может быть выбран формат xml (в блоке **Тип файла** поставьте переключатель на значение xml). Файл будет представлять собой правильный XML-документ, его структура логически будет такая же, как у файла Tab delimited, с учетом специфики xml-форматирования.

Импорт файла типа xml выполняется аналогично импорту файла типа csv.

Экспорт/импорт списка команд

Сохранение списка команд в файл (как с расширением csv, так и с расширением xml), создание файлов конфигурации другими инструментами (вне проекта), импорт файлов выполняется так же, как и аналогичные действия со списком элементов данных. На рисунке 47 показана структура файла конфигурации для экспорта/импорта команд.

Максимальное количество команд для каждого устройства составляет 2000 элементов.



MirrorAdr	SelectPeriod	ExecTimeout	Name	Descr	TypeId	IoAdr	HighBound	LowBound	Scale
0	5	45	IM1_Cmd_1	1	0	0	0	0	
11	5	50	IM1_Cmd_2	9	0	0	0	0	
0	5	51	IM1_Cmd_3	2	0	0	0	0	

Рисунок 47 – Структура файла конфигурации для экспорта/импорта команд на примере IEC-104

В файле конфигурации присутствуют следующие поля:

- **MirrorAdr** – адрес элемента данных, в который будет помещено значение, переданное командой (0 – не используется);
- **SelectPeriod** – время блокировки для команд с функциональностью «Выборка перед выполнением» (0 – не используется);
- **ExecTimeout** – таймаут выполнения;
- **Name** – наименование канала (и соответствующей переменной, в режиме автоматической генерации);
- **Descr** – описание канала (недопустим символ табуляции в описании, пустое значение допустимо);
- **TypeId** – идентификатор типа согласно протоколу IEC-60870-5-104(101). Для типов с меткой времени можно указать автоматическую генерацию временных меток, прибавив к идентификатору типа 128 (80 hex);
- **IoAdr** – уникальный адрес элемента согласно протоколу IEC-60870-5-104(101);
- **HighBound** – верхняя граница для диапазонных типов данных (0 – не используется);
- **LowBound** – нижняя граница для диапазонных типов данных (0 – не используется);
- **Scale** – масштаб, для масштабируемых величин (0 – не используется).

ТЕСТИРОВАНИЕ СИСТЕМЫ, ВЫЯВЛЕНИЕ НЕИСПРАВНОСТЕЙ И ИХ УСТРАНЕНИЕ

Общие действия

После настройки IEC-104 Slave (на контроллерах серии Regul RX00) нередко возникает ситуация, когда в силу каких-либо причин IEC-104 Master, представленный SCADA-системой или иным сторонним приложением, не может подключиться к контроллеру по протоколу IEC-104. Основные вероятные причины (в порядке проверки):

- контроллер недоступен из подсети с клиентом IEC-104 Master;
- отсутствует возможность связаться с контроллером по порту, указанному в настройках IEC-104 Slave;
- неправильные настройки списка основных и резервных мастер-адресов IEC-104 Slave;
- ошибки в работе подключающегося клиента IEC-104 Master, несоответствие протоколу.

Доступность контроллера требуется проверить, если подсеть, в которой находится клиент IEC-104 Master, не совпадает с подсетью, к которой принадлежит компьютер (с установленным Epsilon LD), используемый для настройки контроллера. Для проверки используйте команду **ping** командного интерпретатора Windows, либо аналогичную команду в shell-терминале Linux-подобных операционных систем. Выполняется команда на компьютере с клиентом IEC-104 Master, укажите IP-адрес контроллера серии Regul RX00.

Пример успешного выполнения ping (командный интерпретатор Windows):

```
C:\>ping 172.29.22.240
Обмен пакетами с 172.29.22.240 по 32 байт:
Ответ от 172.29.22.151: число байт=32 время<1мс TTL=64
...
```

Пример – ping не выполнен:

```
C:\>ping 172.29.22.153
Обмен пакетами с 172.29.22.153 по 32 байт:
Превышен интервал ожидания для запроса.
...
```

Если ping до контроллера не выполняется, то следует провести диагностику сетевых подключений и настроек шлюзов (gateways) и т.п. В общем случае это выходит за рамки данного руководства. Возможно также, что на тестируемом компьютере запущен брандмауэр, в настройках которого заблокированы исходящие ICMP-пакеты. Следует изменить настройки брандмауэра, либо временно его отключить.

При наличии связи с контроллером проверьте возможность TCP-соединения с портом, указанным в настройках IEC-104 Slave, по умолчанию порт 2404. В качестве тестовой программы можно использовать Telnet-клиент, которая входит в поставку ОС семейства Windows, как опциональный компонент (можно установить через панель «Администрирование»).

Пример попытки подключения telnet (командный интерпретатор Windows) к IP 172.29.22.153, порт 2404:

```
C:\>telnet 172.29.22.153 2404
```

При сбое подключения появится следующее сообщение:

```
Подключение к 172.29.22.153...
Не удалось открыть подключение к этому узлу, на порт 2404: Сбой подключения
```

В такой ситуации следует подключиться к контроллеру (выполнить Login) и проверить состояние устройства **Slave 104 Driver** в дереве объектов. Значок «красный треугольник» показывает сбой в работе – в этом случае, скорее всего, потребуется обратиться в техподдержку, приложив журнал контроллера и файл проекта. Однако более вероятна ситуация, когда устройство **Slave 104 Driver** запущено и работает, но отсутствуют внешние подключения, и устройство помечено серым треугольником.

Если не удастся подключиться к требуемому порту контроллера, то следует проверить и изменить при необходимости настройки брандмауэра (правила для исходящих соединений) на тестируемом компьютере с клиентом IEC-104 Master. Если этот компьютер подключается к контроллеру из другой сети, то может потребоваться настройка маршрутизаторов.

При успешном подключении клиент Telnet покажет пустое окно терминала, без сообщений об ошибках. В программе Epsilon LD в журнале контроллера можно будет увидеть следующую запись от компонента IoDrvPs104Slave (Рисунок 48).

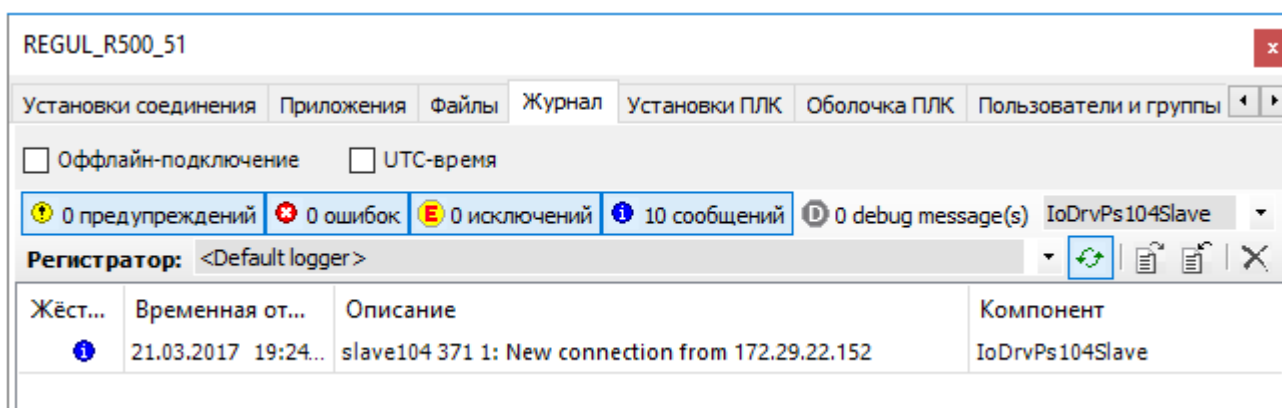


Рисунок 48 – Сообщение от компонента IoDrvPs104Slave

В окне дерева устройств состояние устройства **Slave 104 Driver** будет показано зеленым кружком, что означает, что устройство в работе и появились внешние подключения. Если, несмотря на это, клиент IEC-104 Master не получает данных, следует проверить наличие

IP-адреса клиента в списке основных или резервных IP-адресов в настройках Slave 104 Driver и, при отсутствии, добавить. Если все настройки верны, но данных нет, то в рамках «стандартной» проверки предлагается использовать ПО «Программный шлюз-конвертор OPC IEC 60870-5-104 (МЭК-104)» (далее OPC-104) производства компании «Прософт-Системы». Демо-версия OPC-104 доступна по запросу. Программа принимает данные по IEC-104/101 и отдает по OPC DA, значения запрашиваемых параметров доступны для просмотра через интерфейс программы.

Использование программы OPC-104 для тестирования

Предполагается, что в настройках Slave 104 Driver на контроллере задано несколько элементов данных и команд, как показано на рисунках (Рисунок 49, Рисунок 50).

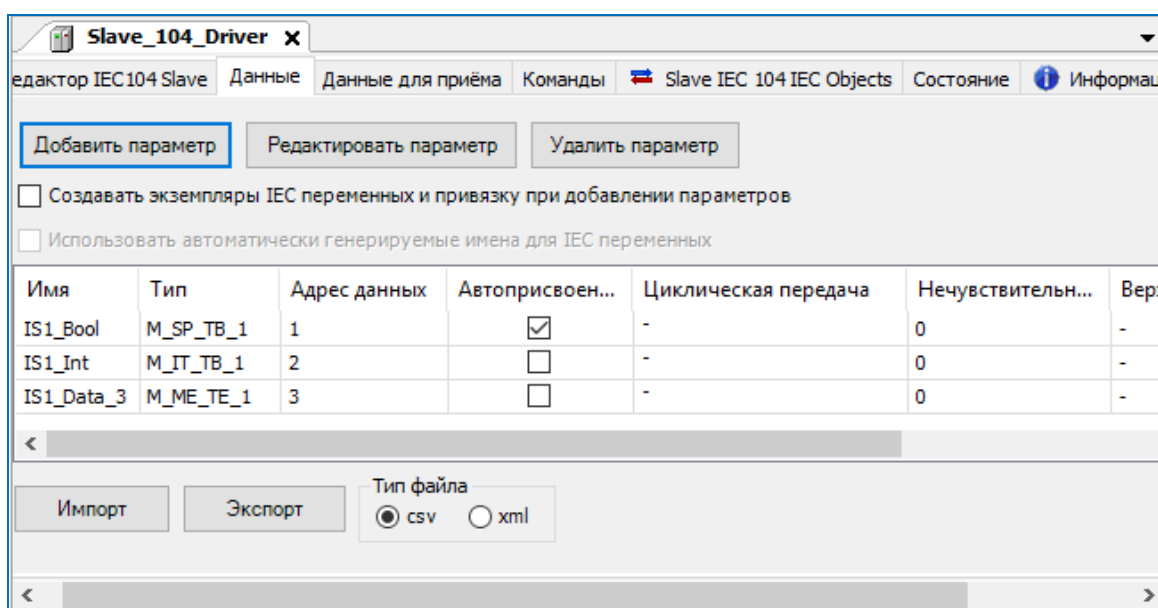


Рисунок 49 – Пример списка элементов данных

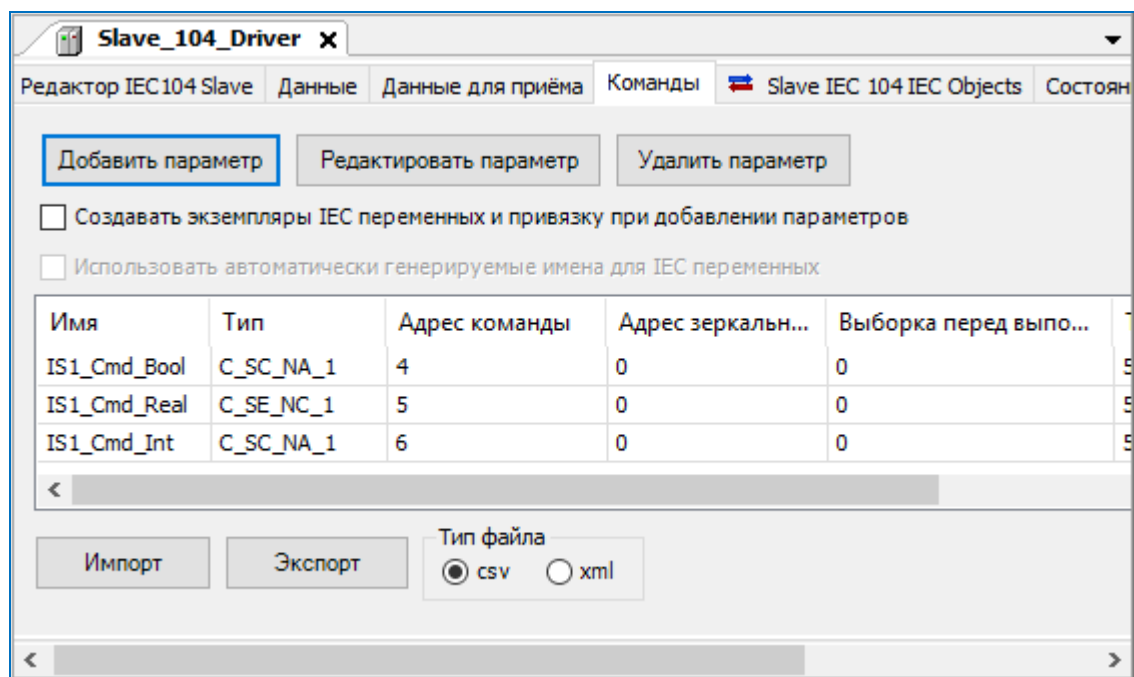


Рисунок 50 – Пример списка команд

Для обработки данных и команд в контроллере используется следующий программный код:

Объявление

```
PROGRAM Test_IEC104
VAR
    //переменные для имитации данных МЭК-104
    bool_value : BOOL;
    int_value: DINT;
    real_value: REAL;
    //переменные для отображения принятых команд
    bool_cmd: BOOL;
    int_cmd: DINT;
    real_cmd: REAL;
    //для генерации дискретного значения
    timer_of_bool_value:TON:=(PT:=T#2S);
END_VAR
```

Реализация

```
//генерация данных
timer_of_bool_value(IN:=TRUE);
IF (timer_of_bool_value.Q) THEN
    bool_value:= NOT bool_value;
    timer_of_bool_value(IN:=FALSE);
END_IF
int_value:=int_value+1;
real_value:=SIN (2*3.141592653*TIME_TO_REAL(TIME())/5000);
//добавление данных в очередь отправки IEC-104 Slave
IS1_Bool.value:= BOOL_TO_WORD(bool_value);
IS1_Real.value:= real_value;
IS1_Int.value:= int_value;
//получение значений из команд, принятых IEC-104 Slave
bool_cmd := WORD_TO_BOOL(IS1_Cmd_Bool.m_value.m_word);
int_cmd := IS1_Cmd_Int.m_value.m_dint;
real_cmd := IS1_Cmd_Real.m_value.m_float;
```

Для проверки связи с контроллером по протоколу IEC-104 установите на тестируемом компьютере ПО OPC-104, запустите программу (Рисунок 51).

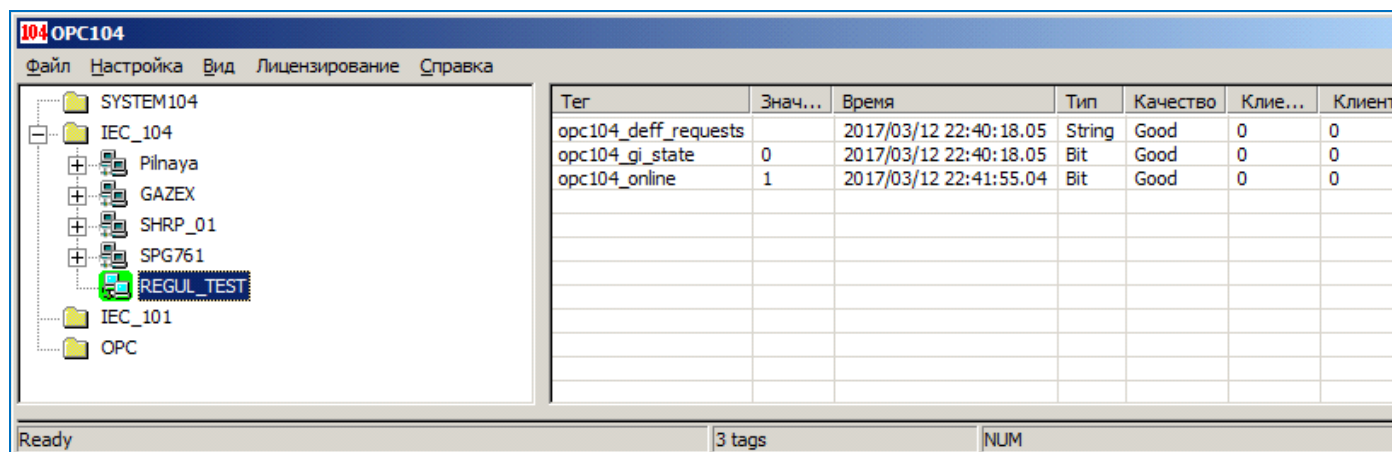


Рисунок 51. Программный шлюз-конвертор OPC IEC 60870-5-104 (МЭК-104)

В дереве объектов выберите элемент IEC_104, нажмите правую кнопку мыши и в появившемся контекстном меню выберите пункт **Добавить узел**. Укажите условное имя узла, IP-адрес контроллера, порт; установите флажок в поле **Active connection**, затем нажмите кнопку **ОК** (Рисунок 52). Предполагается, что остальные настройки узла соответствуют тем, которые заданы на контроллере.

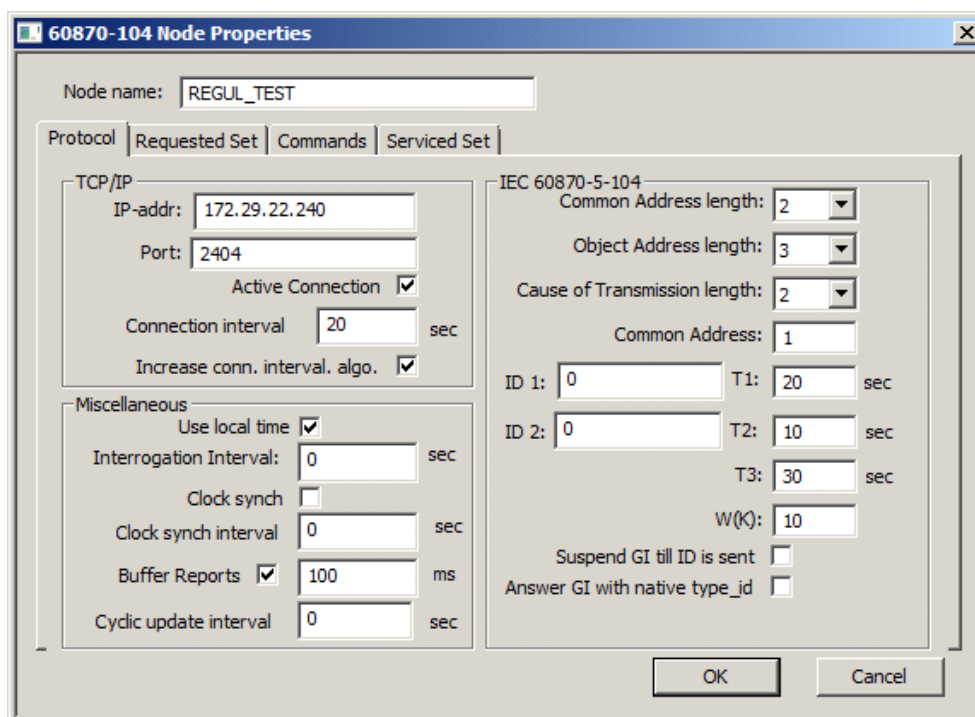


Рисунок 52. Установка параметров узла

Далее опишите элементы данных и команды, соответствующие тем, что были созданы в контроллере. С помощью контекстного меню откройте окно **Node Properties**, перейдите на вкладку **Requested Set**. Опишите 3 элемента данных с адресами с 1 по 3 (Рисунок 53).

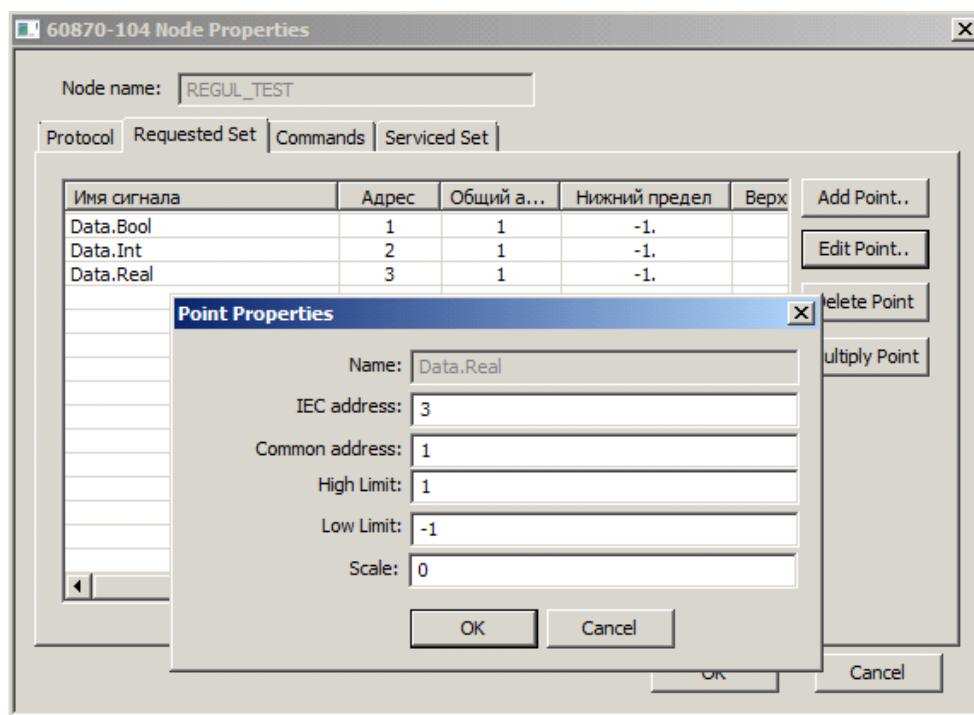


Рисунок 53. Параметры элемента данных

Перейдите на вкладку **Commands**, где опишите соответствующие команды – для каждой команды нужно указать тип и адрес (Рисунок 54). Типы и адреса соответствуют командам, описанным в контроллере.

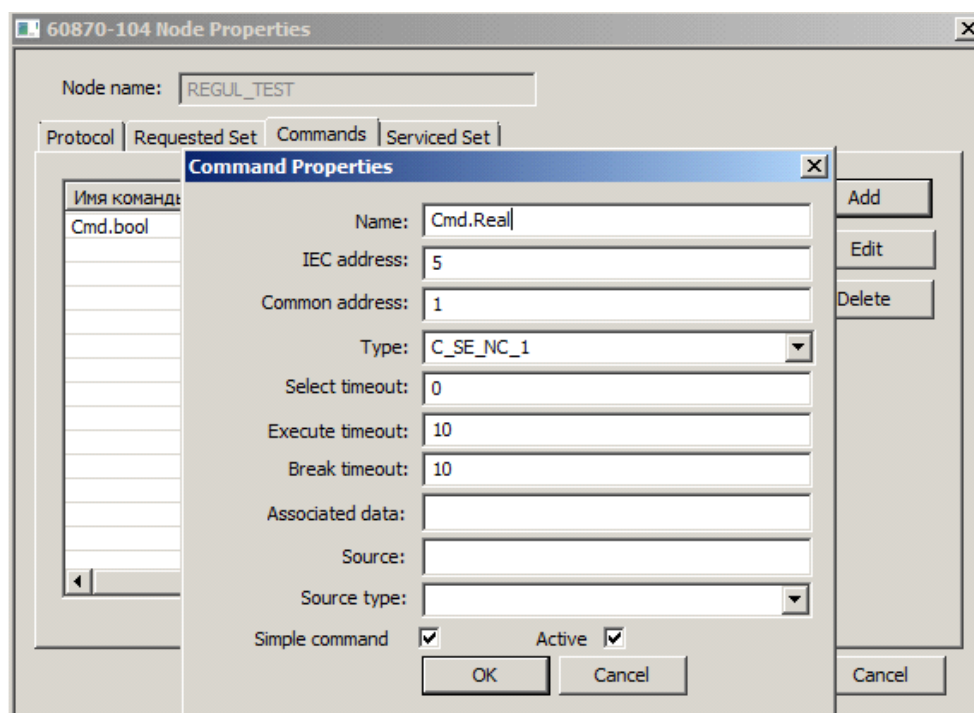


Рисунок 54. Параметры команды

После окончания настроек нажмите кнопку **OK**.

Для запуска тестирования выберите в контекстном меню узла пункт **Открыть**. Пиктограмма узла окрасится зеленым цветом. При раскрытии списка тегов узла становится видно, что с контроллера приходят данные.

Для тестирования команды укажите на нее в списке, нажмите клавишу **Enter**. Появится форма для ввода нового значения, где укажите значение, далее нажмите клавишу **Enter**. Значение должно быть записано на контроллер.

Если, несмотря на правильные настройки, вы не смогли добиться связи с контроллером, то следует обратиться в техподдержку.

ОБРАЩЕНИЕ В СЛУЖБУ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ

Для обращения в техническую поддержку Пользователю необходимо сформировать запрос на сайте технической поддержки: <https://support.prosoftsystems.ru>, либо отправить письмо по электронной почте: support@prosoftsystems.ru. В первом случае требуется предварительная регистрация.

Обращение обязательно должно содержать следующие сведения:

- подробное описание сложившейся ситуации;
- наименование объекта и его месторасположение;
- наименование системы автоматизации;
- модель ПЛК;
- серийный номер ПЛК;
- версия среды разработки Epsilon LD;
- версия СПО-контроллера;
- архив с лог-файлами (см. документ «Epsilon LD User Guide DPA 302. Раздел «Журнал событий»);
- архив с лог-файлами, включающими в себя период времени, когда произошел отказ;
- дата и время возникновения отказа. А также периодичность и устойчивость повторения подобных отказов в случае, если такая информация имеется.

Желательно прислать проект для Epsilon LD, так как это может значительно упростить и ускорить процесс поиска причины отказа.

Для того, чтобы узнать, как получить необходимую информацию (сведений о версии Epsilon LD, версии СПО и так далее), ознакомьтесь с содержанием документа «Epsilon LD User Guide DPA 302».