

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

Руководство пользователя

DPA-302.2

Версия 1.5.5

Март 2017

АННОТАЦИЯ

Настоящий документ содержит сведения о настройке обмена данными с применением стандарта IEC-104 на промышленных логических контроллерах серии Regul RX00. Настройка осуществляется с помощью программного обеспечения Epsilon LD.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
Общие сведения.....	4
Начало работы	4
НАСТРОЙКА КОНТРОЛЛЕРА В КАЧЕСТВЕ SLAVE	6
Добавление устройства Slave 104 Driver	6
Настройка списка IP-адресов Slave 104 Driver.....	7
Настройка общих параметров устройства Slave 104 Driver	8
Создание списка элементов данных, передаваемых по IEC-104	10
Создание списка команд, передаваемых по IEC-104	12
Создание переменных для работы с данными/командами, передаваемыми по IEC-104	14
Привязка переменных программы к элементам данных и командам IEC-104....	17
Автоматическая генерация переменных и привязка к каналам ввода-вывода....	19
НАСТРОЙКА КОНТРОЛЛЕРА В КАЧЕСТВЕ MASTER	21
Добавление устройства Master 104 Driver	21
Настройка общих параметров устройства IEC 104 Outer Slave	22
Создание списка элементов данных, передаваемых по IEC-104	23
Создание списка команд, передаваемых по IEC-104	26
Создание переменных для работы с данными/командами, передаваемыми по IEC-104	28
Привязка переменных программы к элементам данных и командам.....	28
Автоматическая генерация переменных и привязка к каналам ввода-вывода....	31
ЭКСПОРТ И ИМПОРТ КОНФИГУРАЦИИ IEC-104	33
Экспорт/импорт списка элементов данных.....	33
Экспорт/импорт списка команд.....	34
ТЕСТИРОВАНИЕ СИСТЕМЫ, ВЫЯВЛЕНИЕ НЕИСПРАВНОСТЕЙ И ИХ УСТРАНЕНИЕ	36
Общие действия.....	36
Использование программы OPC-104 для тестирования	38

ВВЕДЕНИЕ

Общие сведения

Контроллеры серии Regul RX00 могут участвовать в обмене данными по протоколу TCP/IP в соответствии со стандартом IEC 60870-5. Стандарт подробно описан в документе «ГОСТ Р МЭК 60870-5-104-2004 Устройства и системы телемеханики. Часть 5. Протоколы передачи. Раздел 104. Доступ к сети для ГОСТ Р МЭК 870-5-101 с использованием стандартных транспортных профилей».

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

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 www.prosoftsystems.ru.

Запустите программу **Epsilon LD**. Откройте проект, в котором требуется настроить контроллер для обмена данными по стандарту IEC-104. Если такого проекта нет, создайте его 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:

- 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.

В случае, когда контроллер Regul RX00 выступает в качестве Master, для настройки обмена данными потребуется:

- 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 Slave 104 Driver, в том числе номер порта TCP, который открыт данным slave для приема входящих соединений;
- 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.

НАСТРОЙКА КОНТРОЛЛЕРА В КАЧЕСТВЕ SLAVE

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).

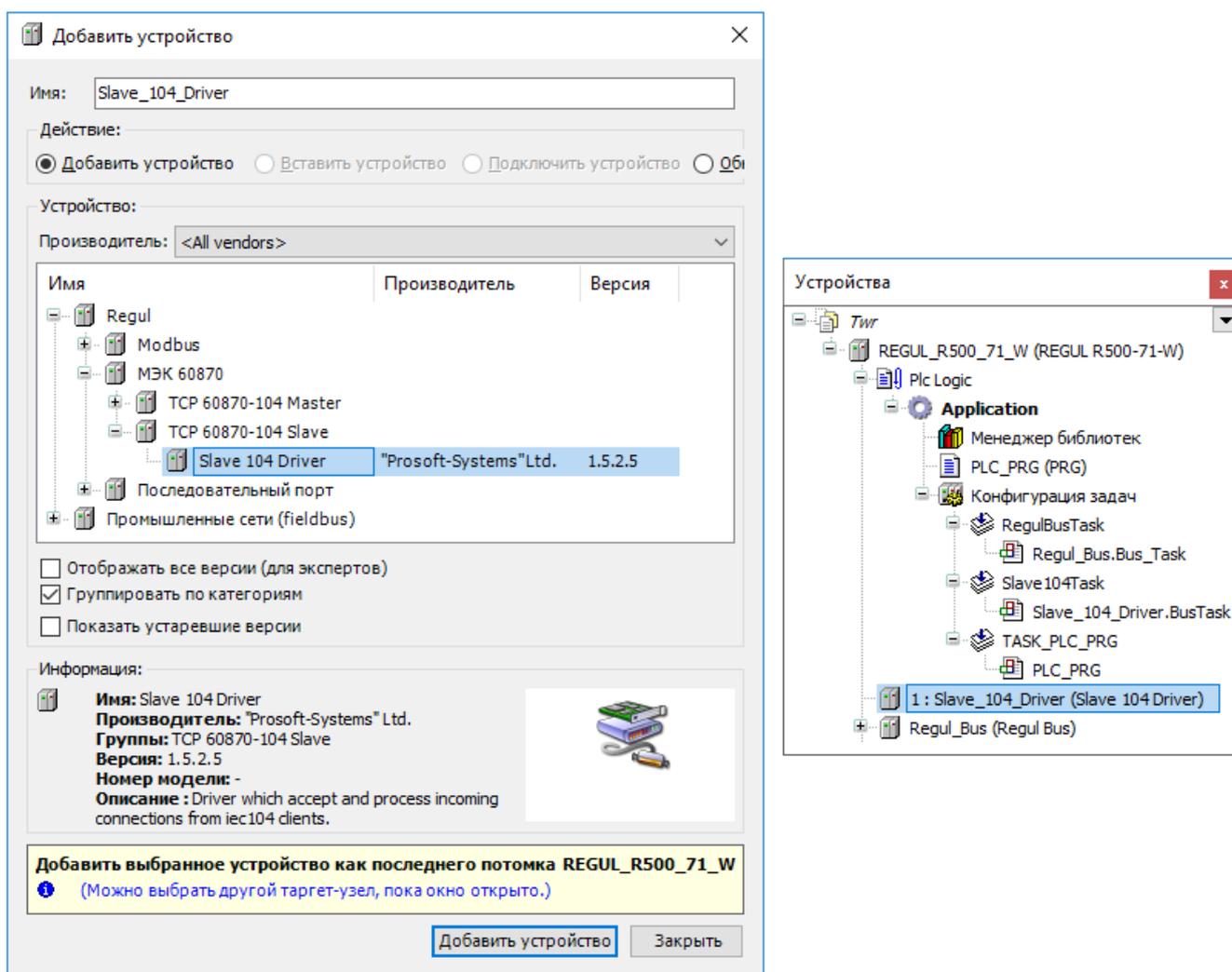


Рисунок 1. Добавление в конфигурацию контроллера устройства Slave 104 Driver

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

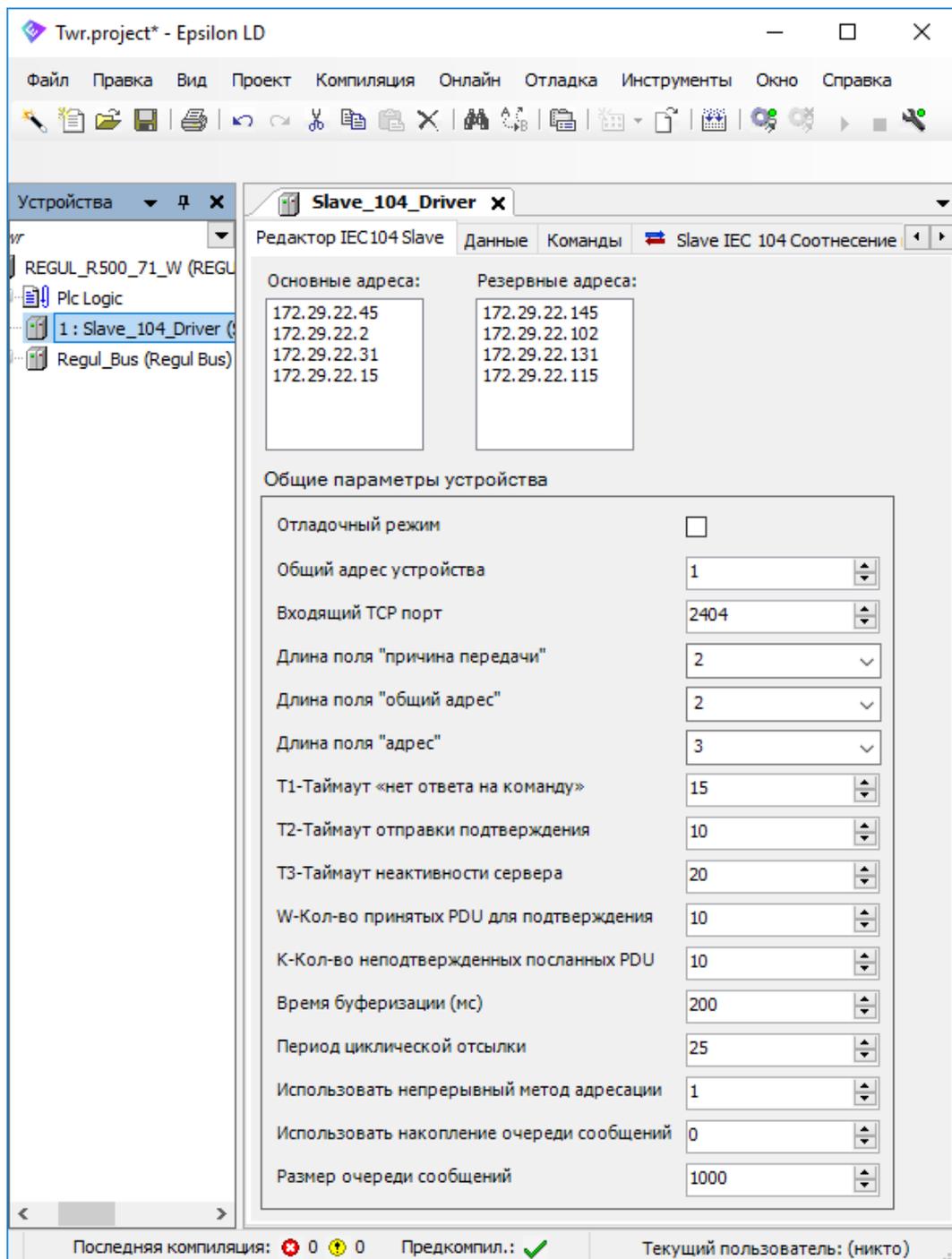


Рисунок 2. Редактор устройства IEC-104 Slave

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. Каждый мастер может вести опрос по основному или резервному каналу.

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).

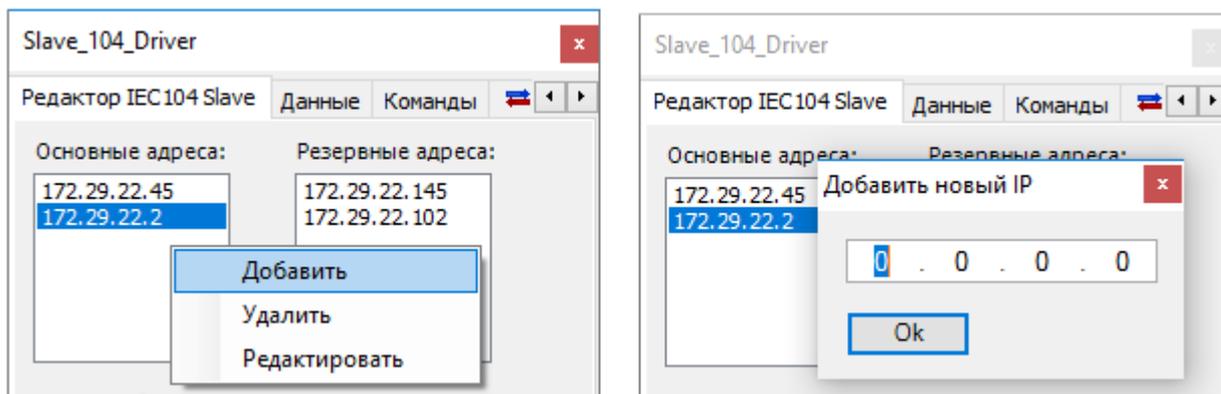


Рисунок 3. Добавление IP-адреса

Число резервных адресов не может быть больше, чем основных.

Если опрос ведется только по одному каналу, то таких мастеров следует поместить в конец списка **Основные адреса**, резервные адреса для них 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**. Будьте внимательны, программа не запрашивает подтверждение на удаление.

Настройка общих параметров устройства Slave 104 Driver

В редакторе устройства **Slave 104 Driver** в блоке **Общие параметры устройства** доступны для настройки следующие параметры:

- **Отладочный режим** – при установке флажка 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. спецификацию IEC-104);
- **Входящий TCP порт** – номер порта TCP, который открыт данным slave для приема входящих соединений. По умолчанию – 2404;
- **Длина поля «общий адрес»**, **Длина поля «адрес»** – характеристики PDU (протокольная единица, пакет), передаваемых по IEC-104, должны быть одинаковы для пары master-slave. По умолчанию для «общего адреса» – 2, допустимые значения: 1 или 2. По умолчанию для «адреса» – 3, допустимые значения: 2 или 3;

- **T1-Таймаут «нет ответа на команду»** – время, в течение которого ожидается ответ на команду или подтверждение доставки PDU. По умолчанию – 15 с. Превышение таймаута приводит к разрыву соединения;
- **T2-Таймаут отправки подтверждения** – по умолчанию – 10 с. 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;
- **T3-Таймаут неактивности сервера** – по умолчанию – 20 с. 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;
- **W-Кол-во принятых PDU для подтверждения** – количество принятых PDU, которое необходимо подтверждать. Подробнее см. описание T2. По умолчанию – 10;
- **K-Кол-во неподтвержденных посланных PDU** – если было отослано количество PDU, указанное в параметре K, и не пришло подтверждение, то отсылка приостанавливается (ожидается подтверждение). По умолчанию – 10;
- **Время буферизации (мс)** – период краткосрочного ожидания пополнения очереди на отправку, с тем, чтобы отослать несколько значений в одном пакете, для оптимизации;
- **Период циклической отсылки** – 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 сек.);
- **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 – (по умолчанию) использовать этот метод;
- **Использовать накопление очереди сообщений** – накапливать данные для мастера в оффлайн-режиме. 0 – не используется. Любое число, отличное от 0, активирует использование накопления данных;
- **Размер очереди сообщений** – размер очереди неотправленных данных, которая заполняется при отсутствии связи с мастером по основному и резервному каналам. Размер очереди должен быть не меньше планируемого количества элементов данных. Это связано с тем, что при внешнем подключении IEC104 Slave может получить команду общего опроса, что приведет к единовременному заполнению очереди последними значениями всех элементов данных. Если размер очереди меньше рекомендуемого, то часть данных не будет отправлена.

Создание списка элементов данных, передаваемых по IEC-104

В редакторе устройства **Slave 104 Driver** перейдите на внутреннюю вкладку **Данные** (Рисунок 4).

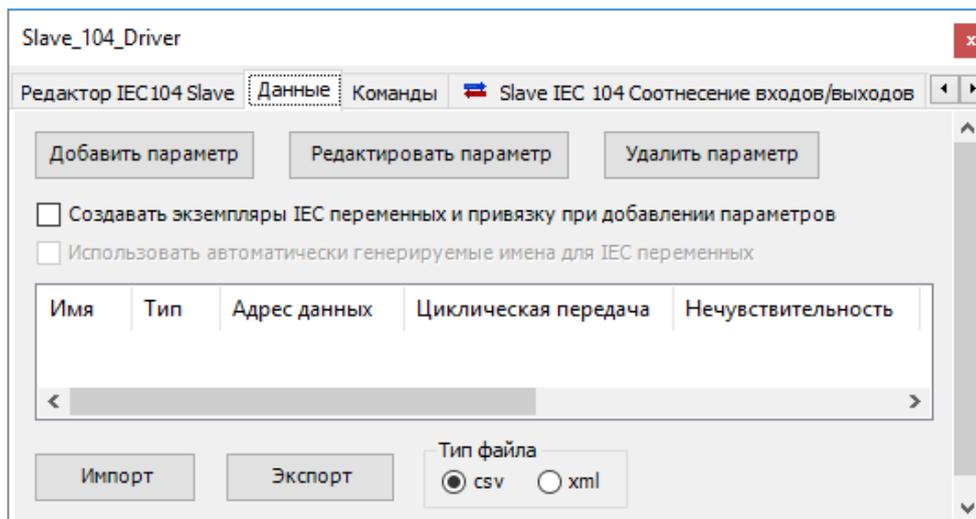


Рисунок 4. Редактор Slave 104 Driver. Вкладка «Данные»

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. Окно ввода/редактирования элемента данных выглядит следующим образом (Рисунок 5).

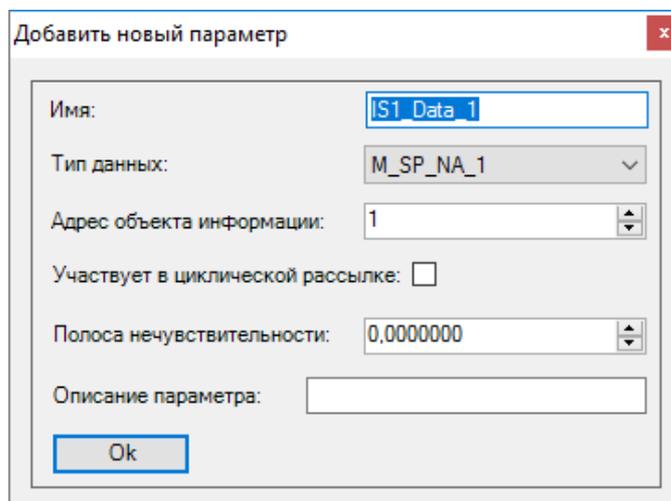


Рисунок 5. Окно добавления нового параметра (элемента данных)

Для элемента данных укажите значения в следующих полях:

- **Имя** – наименование элемента данных;
- **Тип данных** – ассоциированный с этим элементом данных тип IEC-104. Справочно типы данных IEC-104 приведены в таблице 1. Часто используемые типы данных:
 - M_SP_TV_1 – для дискретов с меткой времени,

- M_ME_TF_1 – для float-значений с меткой времени,
- M_IT_TB_1 – для целочисленных значений;
- **Адрес объекта информации** – уникальный адрес элемента данных;
- **Участвует 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. В качестве временной метки подставляется текущее время контроллера;
- **Масштаб** (для масштабируемых параметров, при выборе типа M_ME_TE_1);
- **Описание параметра** – опционально, текстовое описание элемента данных.

Таблица 1. 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	инф. о положении отпаяк 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

Заполненная вкладка **Данные** показана на рисунке 6.

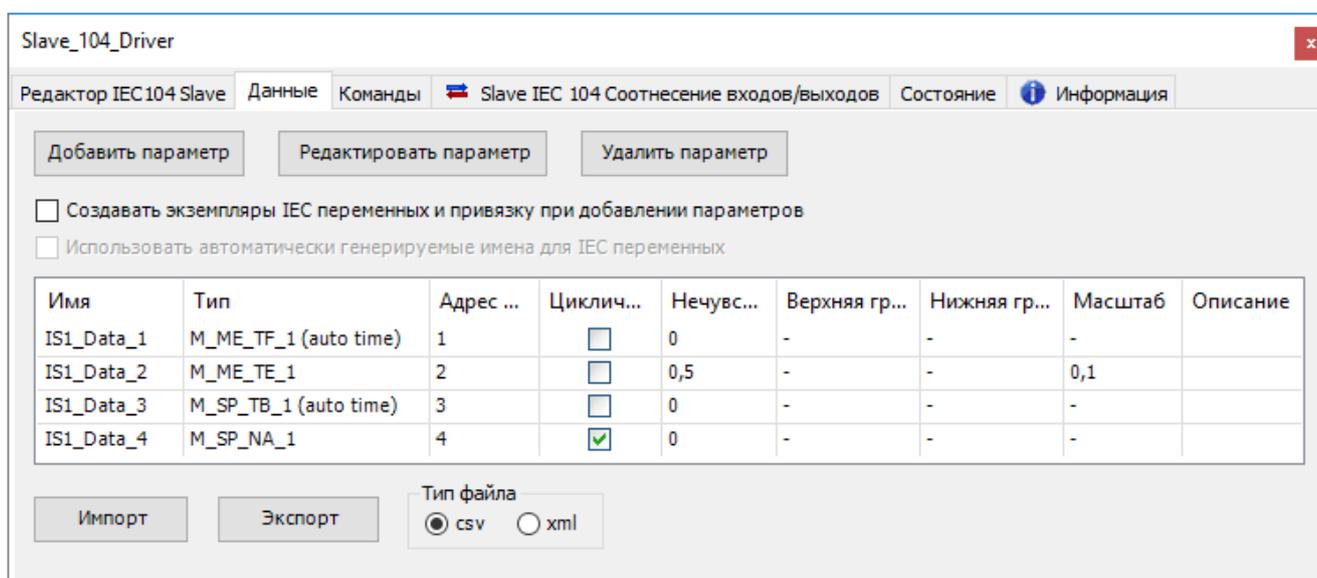


Рисунок 6. Пример списка элементов данных

Создание списка команд, передаваемых 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 7).

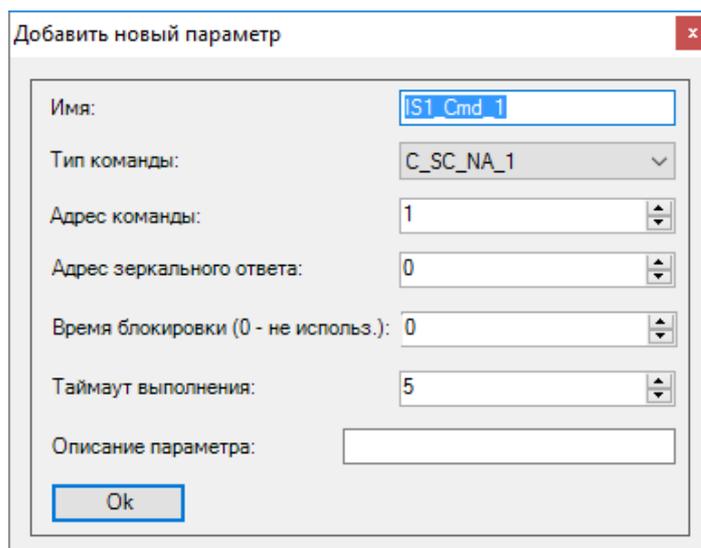


Рисунок 7. Окно добавления новой команды

Для команды укажите значения в следующих полях:

- **Имя** – наименование команды;
- **Тип данных** – ассоциированный с этой командой тип IEC-104. Справочно типы данных IEC-104 приведены в таблице 2;
- **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;
- **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.

Таблица 2. 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)

Тип данных	ID типа данных	Описание
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

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 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 3. 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

Функц. блок	Тип данных IEC-104, указываемый для канала ввода-вывода	Комментарий
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

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;
```

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 свойства следующие:

- **m_value** – значение полученной 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.g_active_connections;
iecl_real.value := 3.14; //список IP адресов активных соединений
iecl3_di.value:= 1; //задаем действительное значение
iecl3_di.timestamp := get_current_timestamp(); //задаем дискретное значение
iecl5_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

В редакторе устройства Slave 104 Driver перейдите на внутреннюю вкладку **Соотнесение входов/выходов** (Рисунок 8).

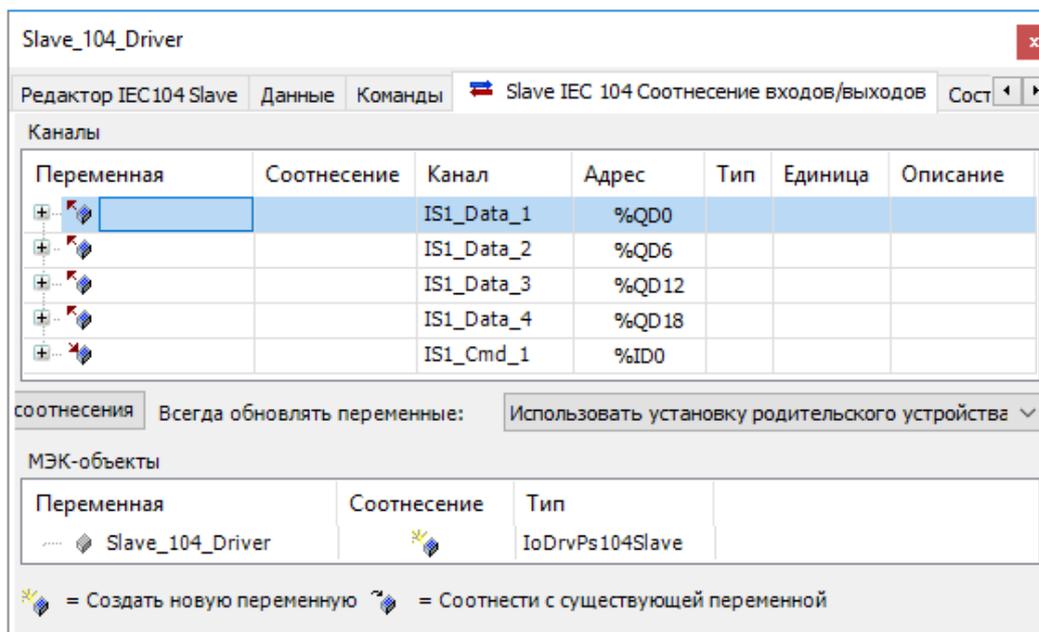


Рисунок 8. Slave 104 Driver Соотнесение входов/выходов

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 9).

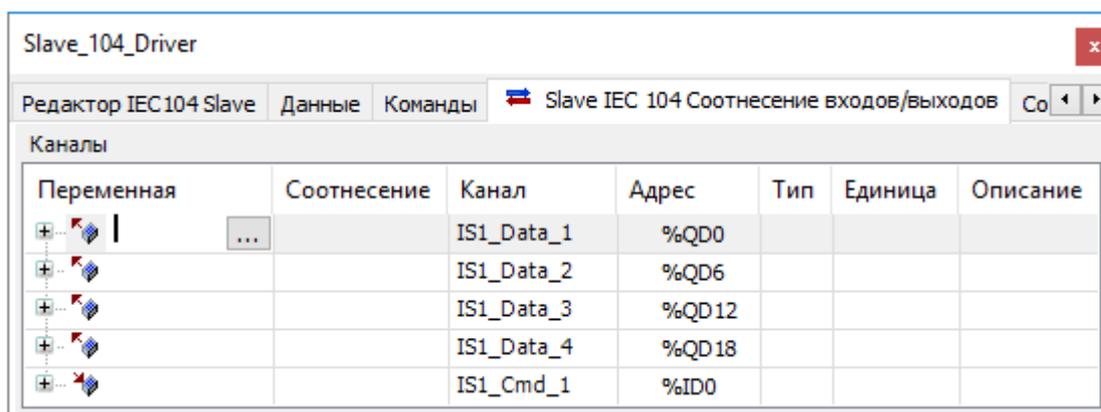


Рисунок 9. Ручной ввод переменной или вызов ассистента ввода

В окне **Ассистент ввода** (Рисунок 10) 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.

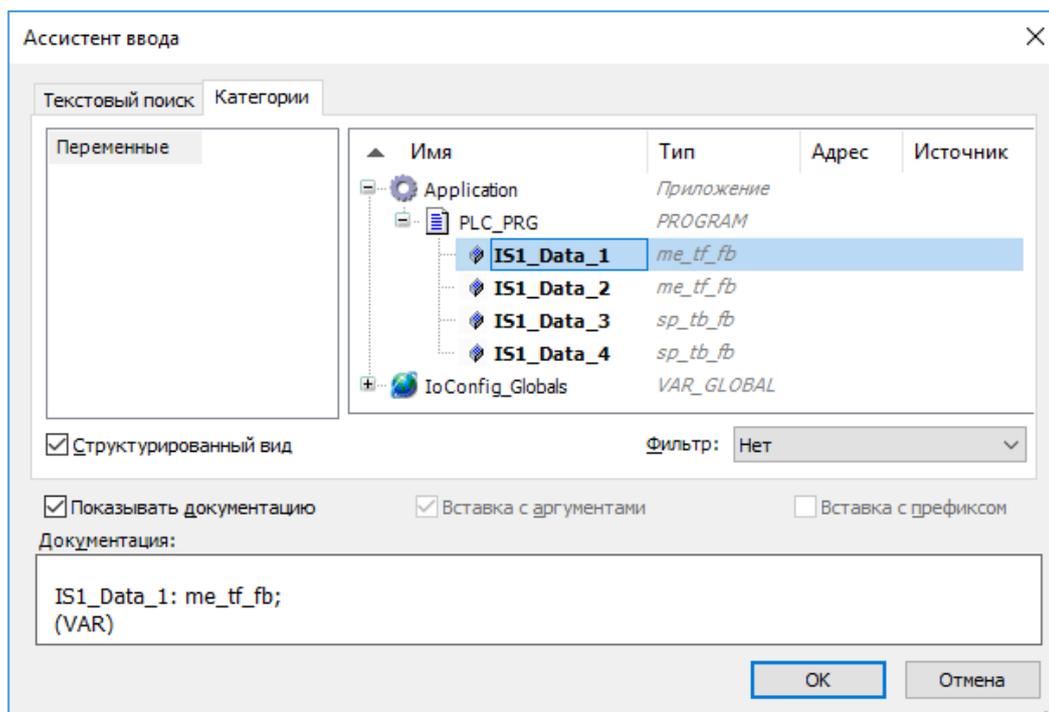


Рисунок 10. Диалоговое окно «Ассистент ввода»

После выбора переменной нажмите кнопку **OK**, закроется окно **Ассистент ввода**, а переменная появится на вкладке **Соотнесение входов/выходов** (Рисунок 11).

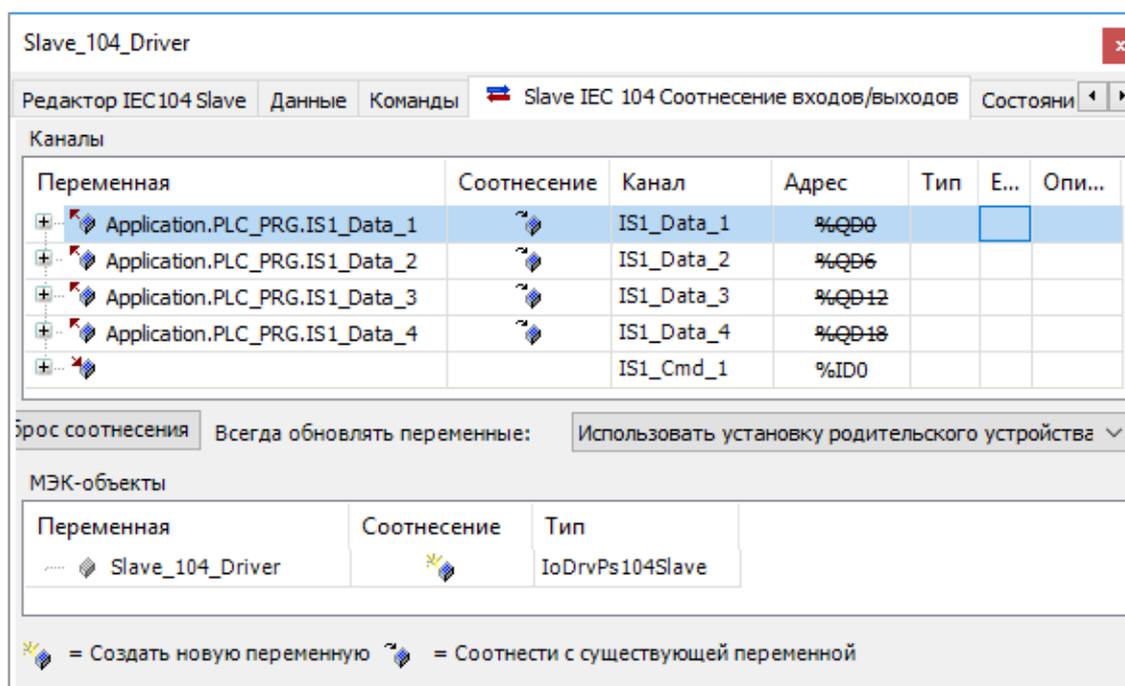


Рисунок 11. Переменные привязаны к каналам вывода

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.

Пояснение: 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 12).

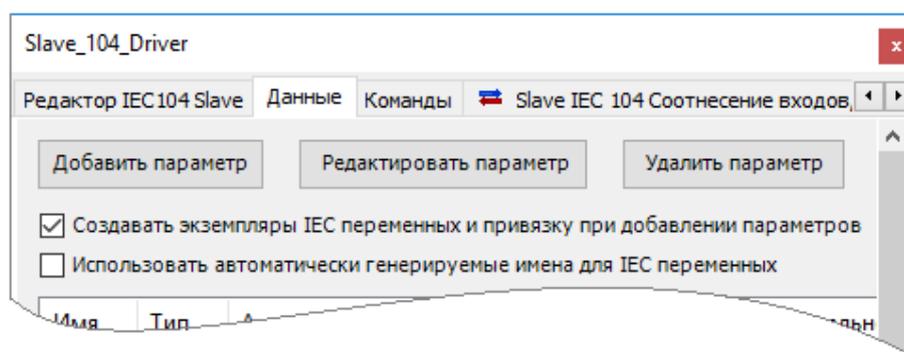


Рисунок 12. Установка флажка в поле «Создавать экземпляры IEC переменных и привязку при добавлении параметров»

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. В этом списке будет создана переменная соответствующего типа, а на вкладке **Соотнесение входов/выходов** новая переменная будет автоматически привязана к новому каналу (Рисунок 13). Наименование переменной будет совпадать с именем канала (пробелы заменяются на подчеркивания).

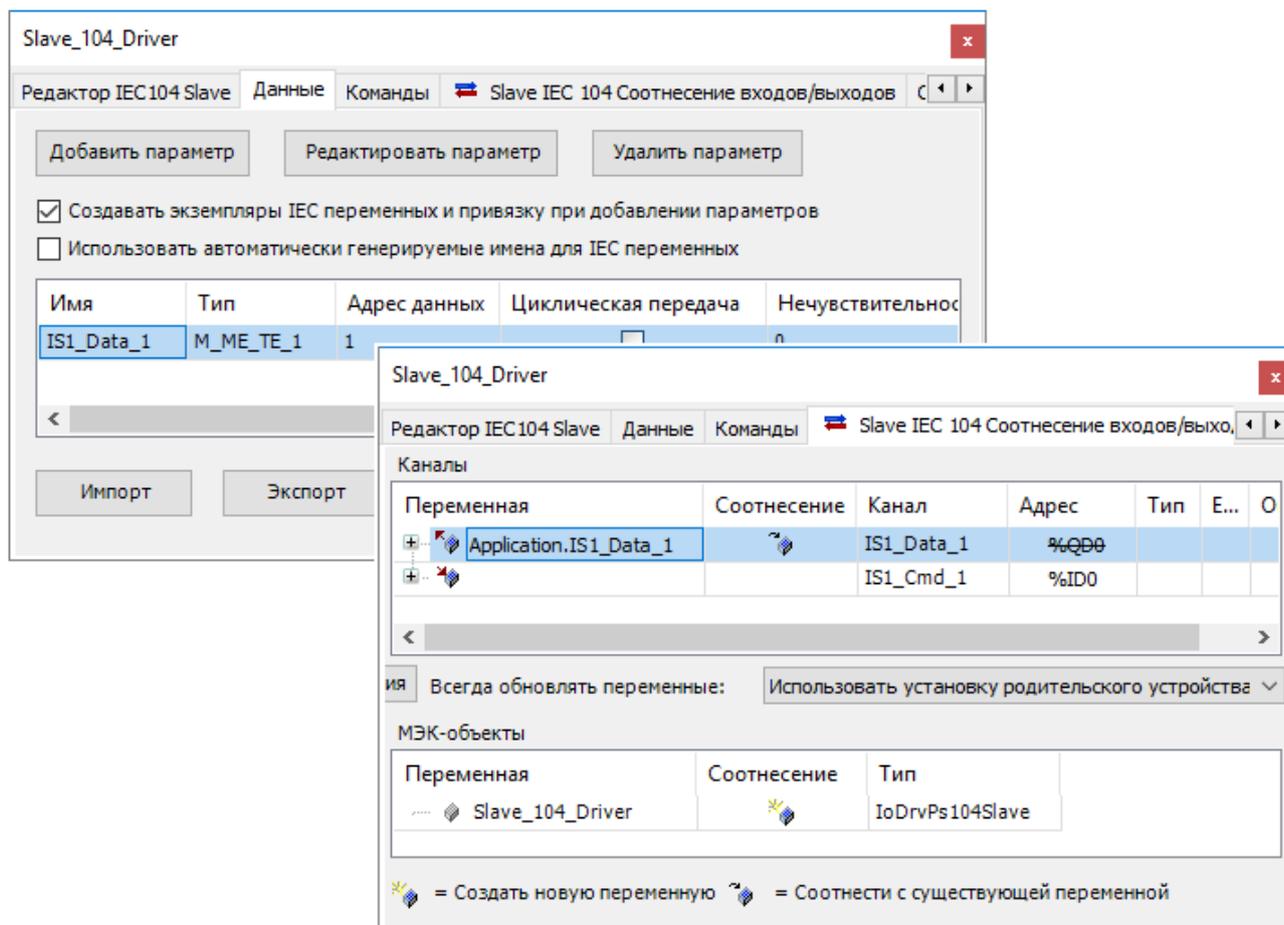


Рисунок 13. Автоматическое создание переменной и автоматическая привязка ее к каналу

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.

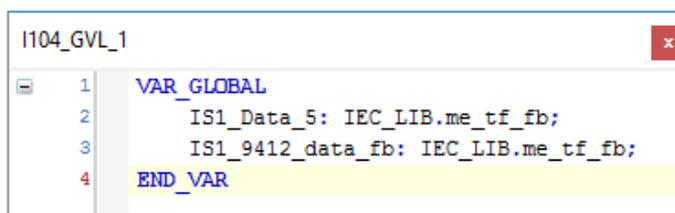


Рисунок 14. Пример описания переменных в списке I104_GVL_1

НАСТРОЙКА КОНТРОЛЛЕРА В КАЧЕСТВЕ MASTER

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 15).

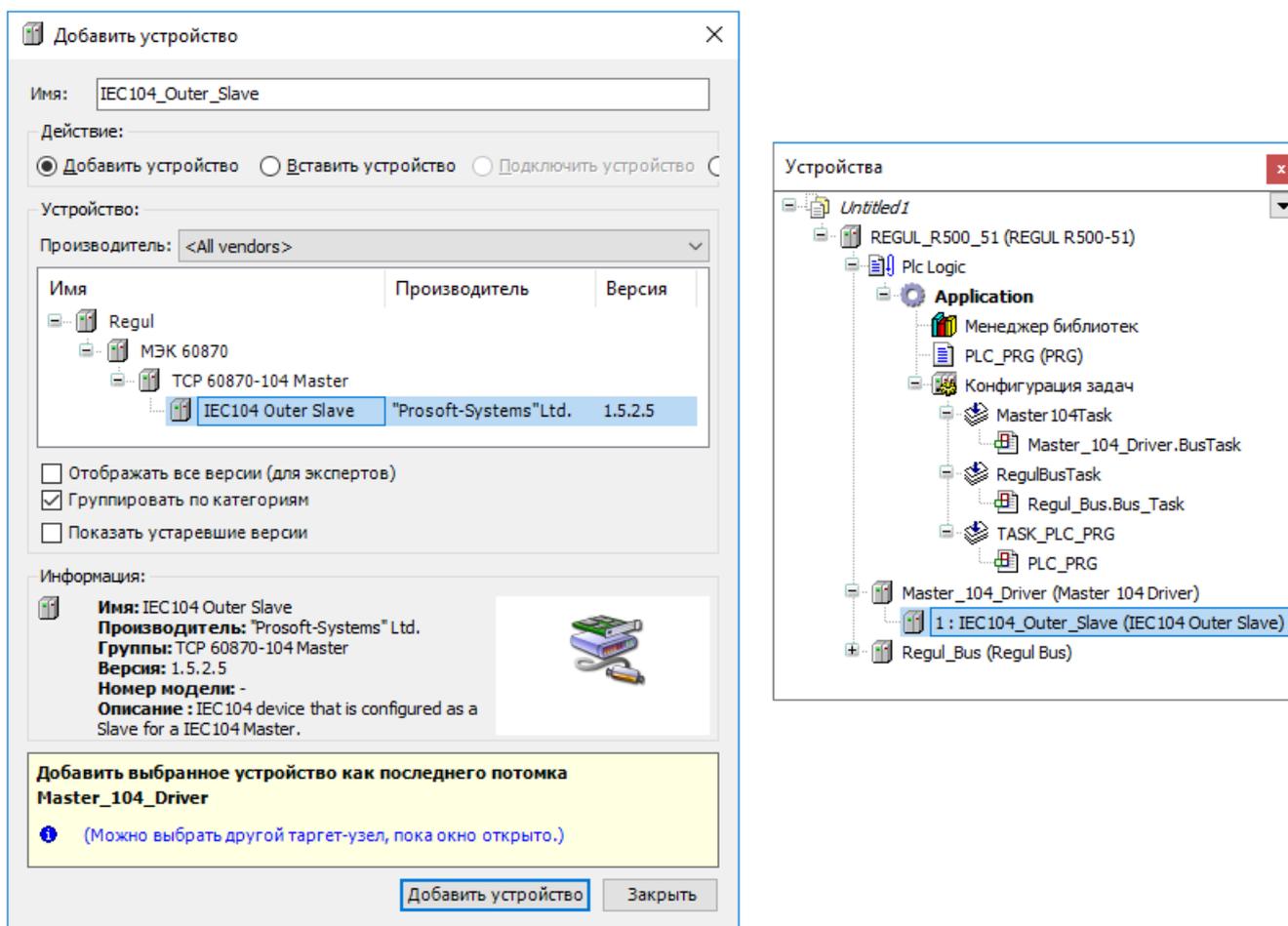


Рисунок 15. Добавление в конфигурацию контроллера устройства IEC 104 Outer Slave

Двойным щелчком по названию устройства **IEC 104 Outer Slave** откройте вкладку параметров. По умолчанию открывается первая внутренняя вкладка **Редактор IEC 104 Outer Slave** (Рисунок 16).

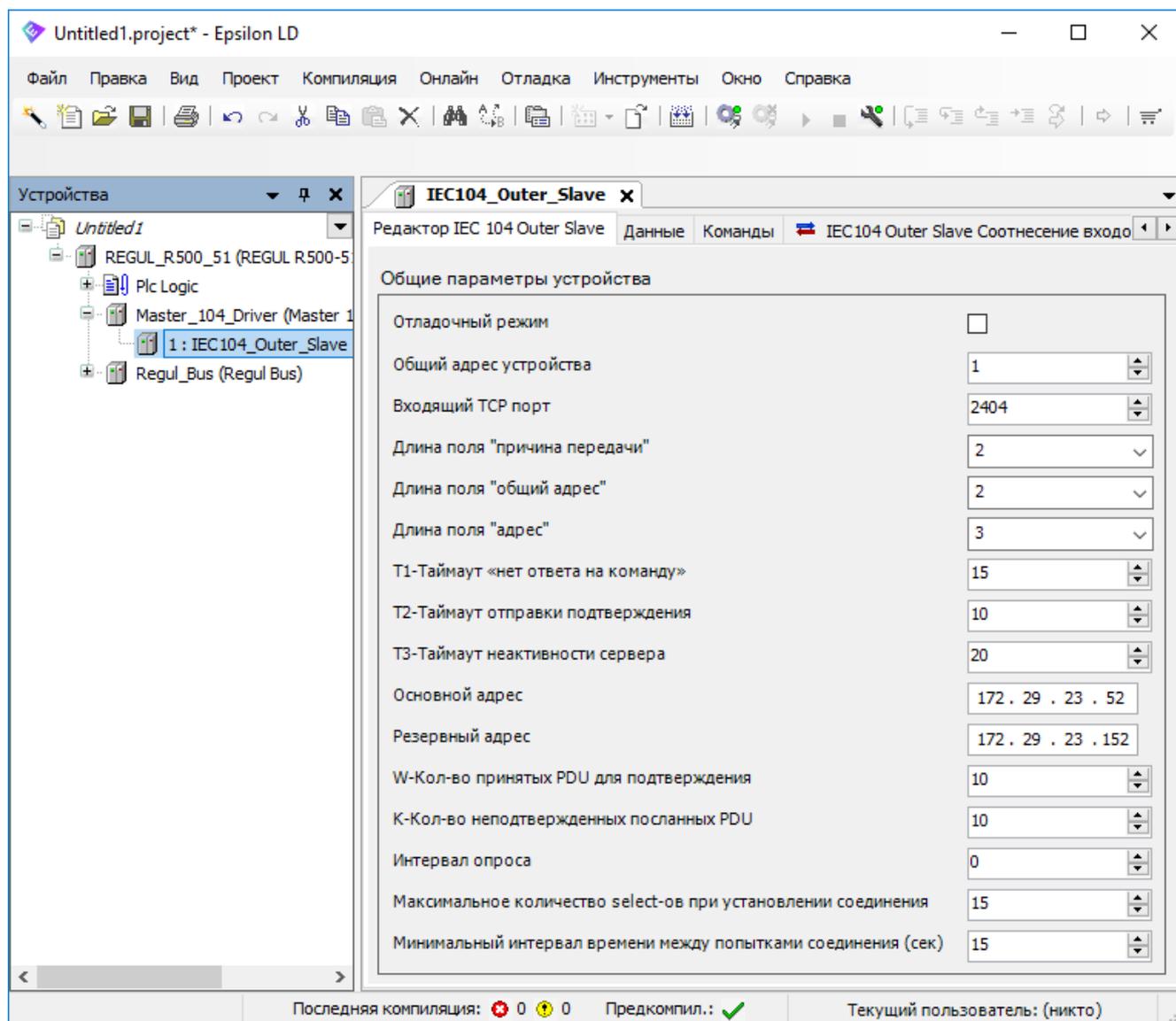


Рисунок 16. Редактор устройства IEC 104 Outer Slave

Настройка общих параметров устройства IEC 104 Outer Slave

В редакторе устройства **IEC 104 Outer Slave** в блоке **Общие параметры устройства** доступны для настройки следующие параметры:

- **Отладочный режим** – при установке флажка 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);
- **Входящий TCP порт** – номер порта TCP, который открыт данным slave для приема входящих соединений. По умолчанию – 2404;

- **Длина поля «причина передачи»** – содержит длину поля «причина передачи» (COT, Cause Of Transmission). По умолчанию этот параметр – 2. Допустимые значения: 1 или 2;
- **Длина поля «общий адрес», Длина поля «адрес»** – характеристики PDU (протокольная единица, пакет), передаваемых по IEC-104, должны быть одинаковы для пары master-slave. По умолчанию для «общего адреса» – 2, допустимые значения: 1 или 2. По умолчанию для «адреса» – 3, допустимые значения: 2 или 3;
- **T1-Таймаут «нет ответа на команду»** – время, в течение которого ожидается ответ на команду или подтверждение доставки PDU. По умолчанию – 15 с. Превышение таймаута приводит к разрыву соединения;
- **T2-Таймаут отправки подтверждения** – по умолчанию – 10 с. 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;
- **T3-Таймаут неактивности сервера** – по умолчанию – 20 с. 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 посылаются тестовое PDU и включается таймер T1. Если в течении интервала T1 не было активности со стороны другой стороны обмена, соединение разрывается;
- **Основной адрес** – IP-адрес основного соединения;
- **Резервный адрес** – IP-адрес резервного соединения;
- **W-Кол-во принятых PDU для подтверждения** – количество принятых PDU, которое необходимо подтверждать. Подробнее см. описание T2. По умолчанию – 10;
- **K-Кол-во неподтвержденных посланных PDU** – если было отослано количество PDU, указанное в параметре K, и не пришло подтверждение, то отсылка приостанавливается (ожидается подтверждение). По умолчанию – 10;
- **Интервал опроса** – период отсылки команды общего опроса. Значение 0 – отсылка команды общего опроса не производится;
- **Максимальное количество select-ов при установлении соединения** – максимальное количество запросов типа «Выборка перед выполнением» (Select before operate);
- **Минимальный интервал времени между попытками соединения (сек).**

Создание списка элементов данных, передаваемых по IEC-104

В редакторе устройства IEC 104 Outer Slave перейдите на внутреннюю вкладку **Данные** (Рисунок 17).

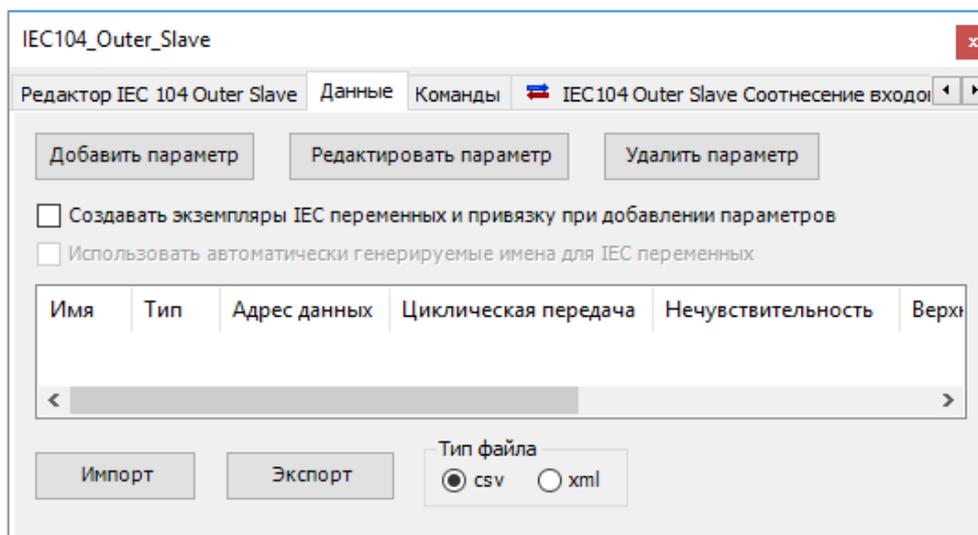


Рисунок 17. Редактор IEC 104 Outer Slave. Вкладка «Данные»

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 18).

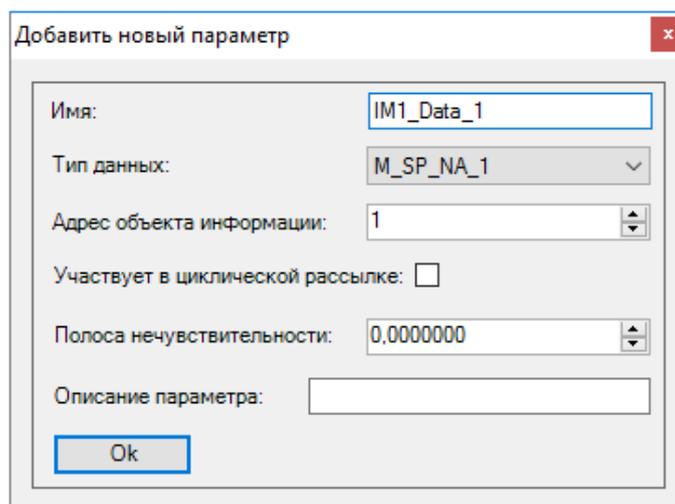


Рисунок 18. Окно добавления нового параметра (элемента данных)

Для элемента данных укажите значения в следующих полях:

- **Имя** – наименование элемента данных;
- **Тип данных** – ассоциированный с этим элементом данных тип IEC-104. Справочно типы данных IEC-104 приведены в таблице 4. Часто используемые типы данных:
 - M_SP_TV_1 – для дискретов с меткой времени,
 - M_ME_TF_1 – для float-значений с меткой времени,
 - M_IT_TV_1 – для целочисленных значений;
- **Адрес объекта информации** – уникальный адрес элемента данных;

- **Участвует 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. В качестве временной метки подставляется текущее время контроллера;
- **Масштаб** (для масштабируемых параметров, при выборе типа M_ME_TE_1);
- **Описание параметра** – опционально, текстовое описание элемента данных.

Таблица 4. 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 отпаек
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	инф. о положении отпаек 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
M_ME_TE_1	35	измеряемая величина масштабированное значение с 56 битной меткой времени

Тип данных	ID типа данных	Описание
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

Заполненная вкладка **Данные** показана на рисунке 19.

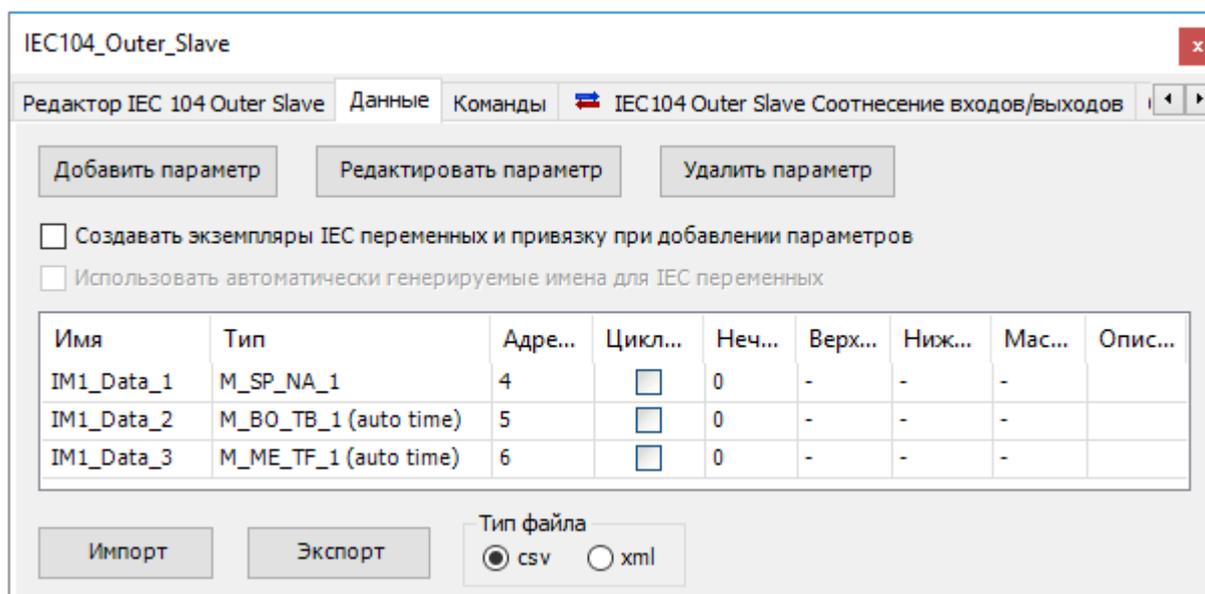


Рисунок 19. Пример списка элементов данных

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.

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 20).

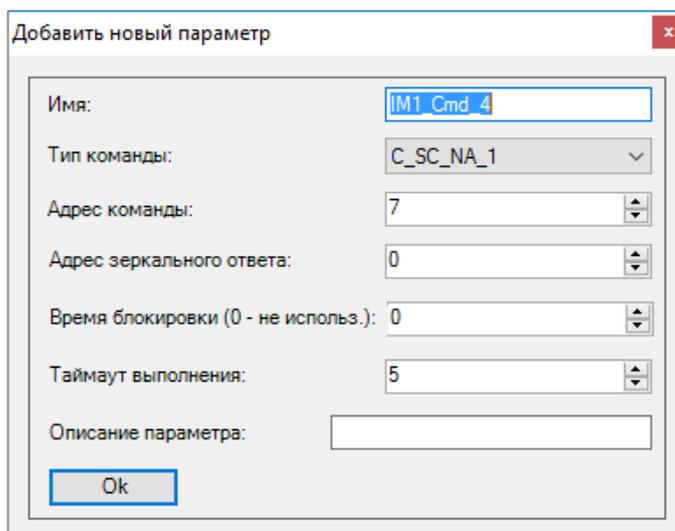


Рисунок 20. Окно добавления новой команды

Для команды укажите значения в следующих полях:

- **Имя** – наименование команды;
- **Тип данных** – ассоциированный с этой командой тип IEC-104. Справочно типы данных IEC-104 приведены в таблице 5;
- **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;
- **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 exes_timeout актуален только для управляющего узла (Master 104 Driver);
- **Масштаб** – для команд, задающих масштабируемые значения.
- **Описание параметра** – опционально, текстовое описание команды.

Таблица 5. 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.

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

В редакторе устройства IEC 104 Outer Slave перейдите на внутреннюю вкладку **Соотнесение входов/выходов** (Рисунок 21).

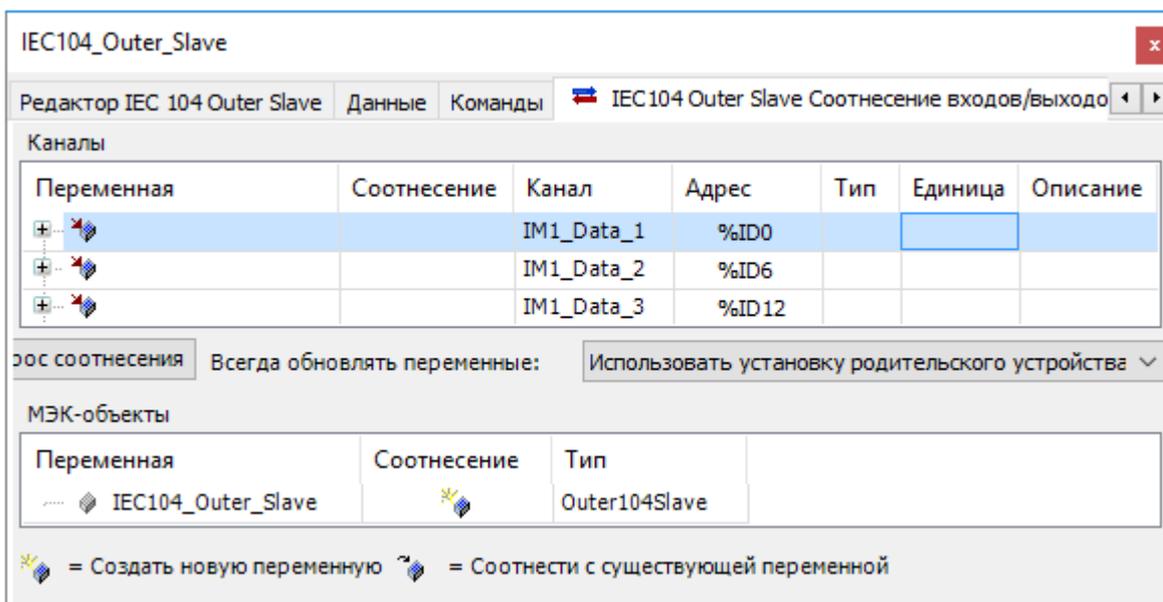


Рисунок 21. IEC 104 Outer Slave Соотнесение входов/выходов

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 22).

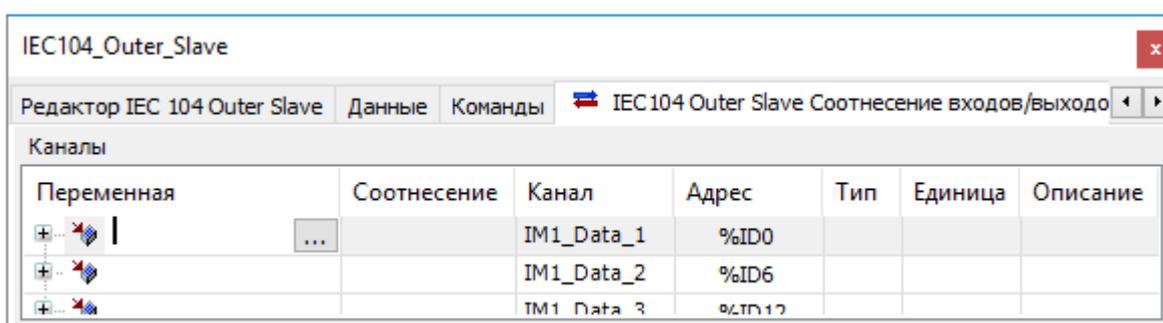


Рисунок 22. Ручной ввод переменной или вызов ассистента ввода

В окне **Ассистент ввода** (Рисунок 23) 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.

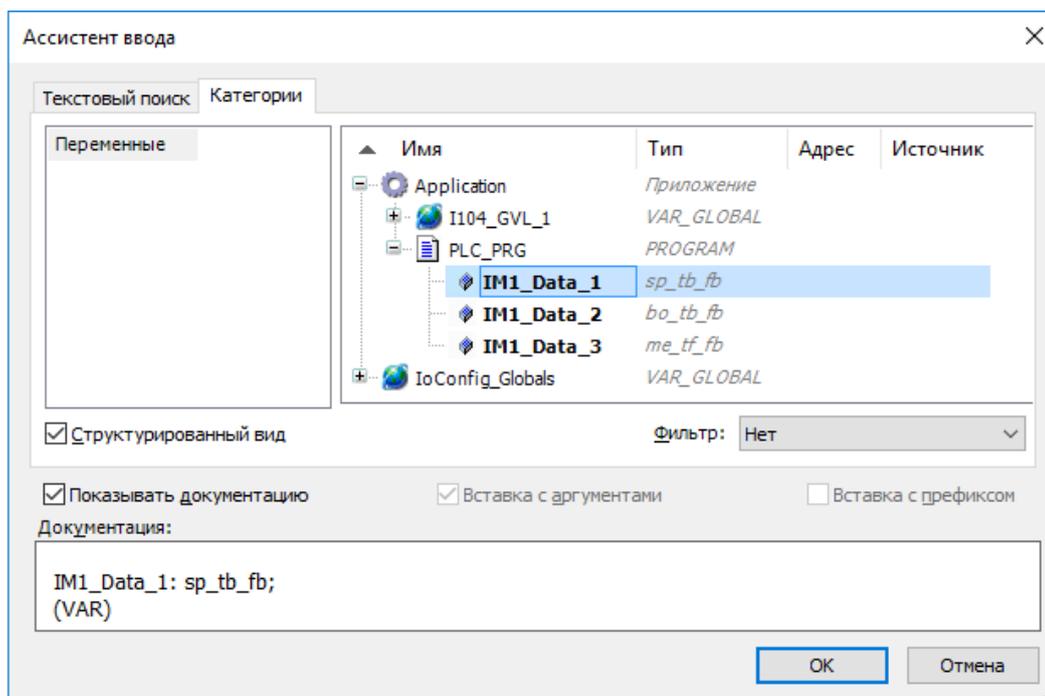


Рисунок 23. Диалоговое окно «Ассистент ввода»

После выбора переменной нажмите кнопку **OK**, закроется окно **Ассистент ввода**, а переменная появится на вкладке **Соотнесение входов/выходов** (Рисунок 24).

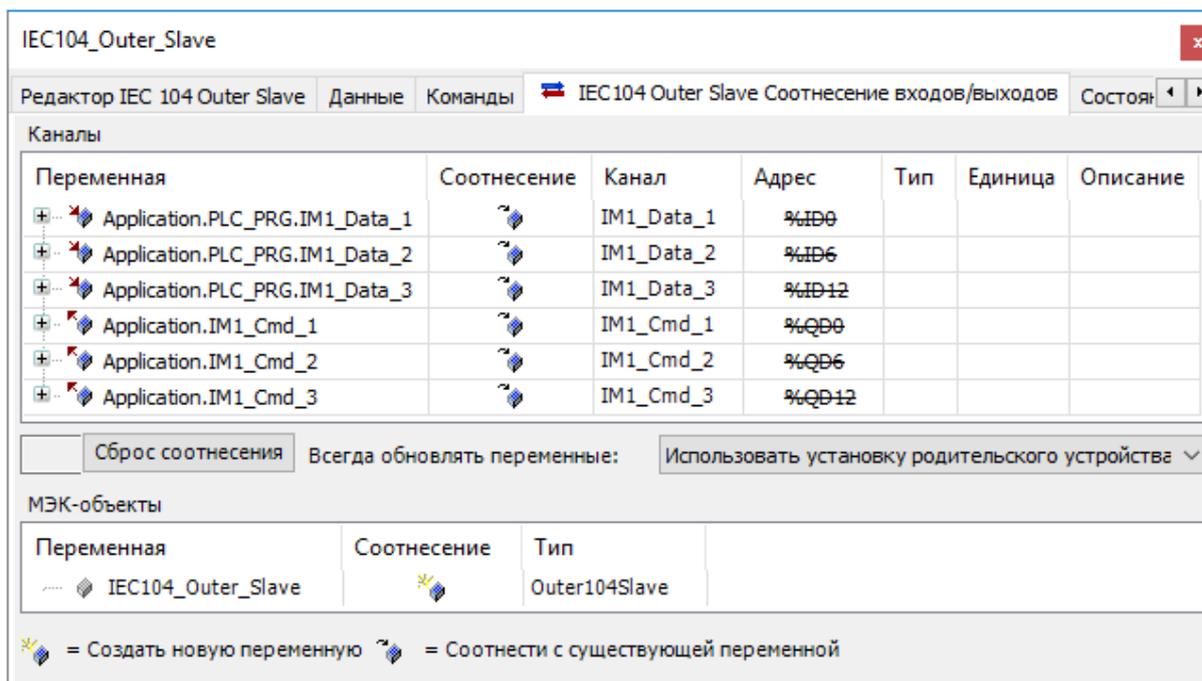


Рисунок 24. Переменные привязаны к каналам вывода

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.

Пояснение: 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 25).

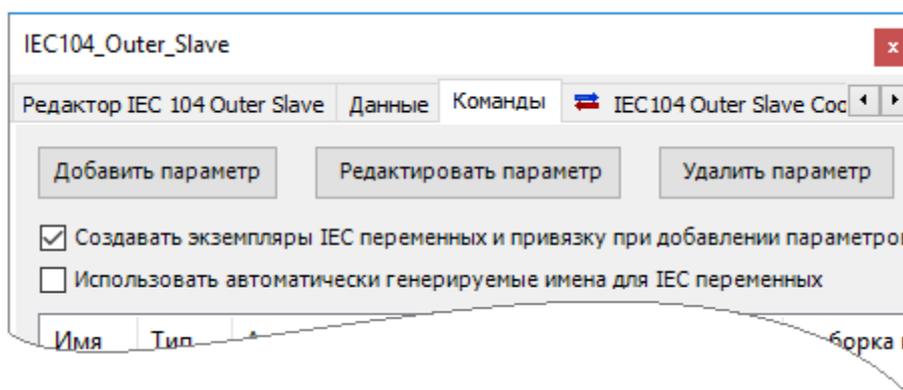


Рисунок 25. Установка флажка в поле «Создавать экземпляры IEC переменных и привязку при добавлении параметров»

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. В этом списке будет создана переменная соответствующего типа, а на вкладке **Соотнесение входов/выходов** новая переменная будет автоматически привязана к новому каналу (Рисунок 26). Наименование переменной будет совпадать с именем канала (пробелы заменяются на подчеркивания).

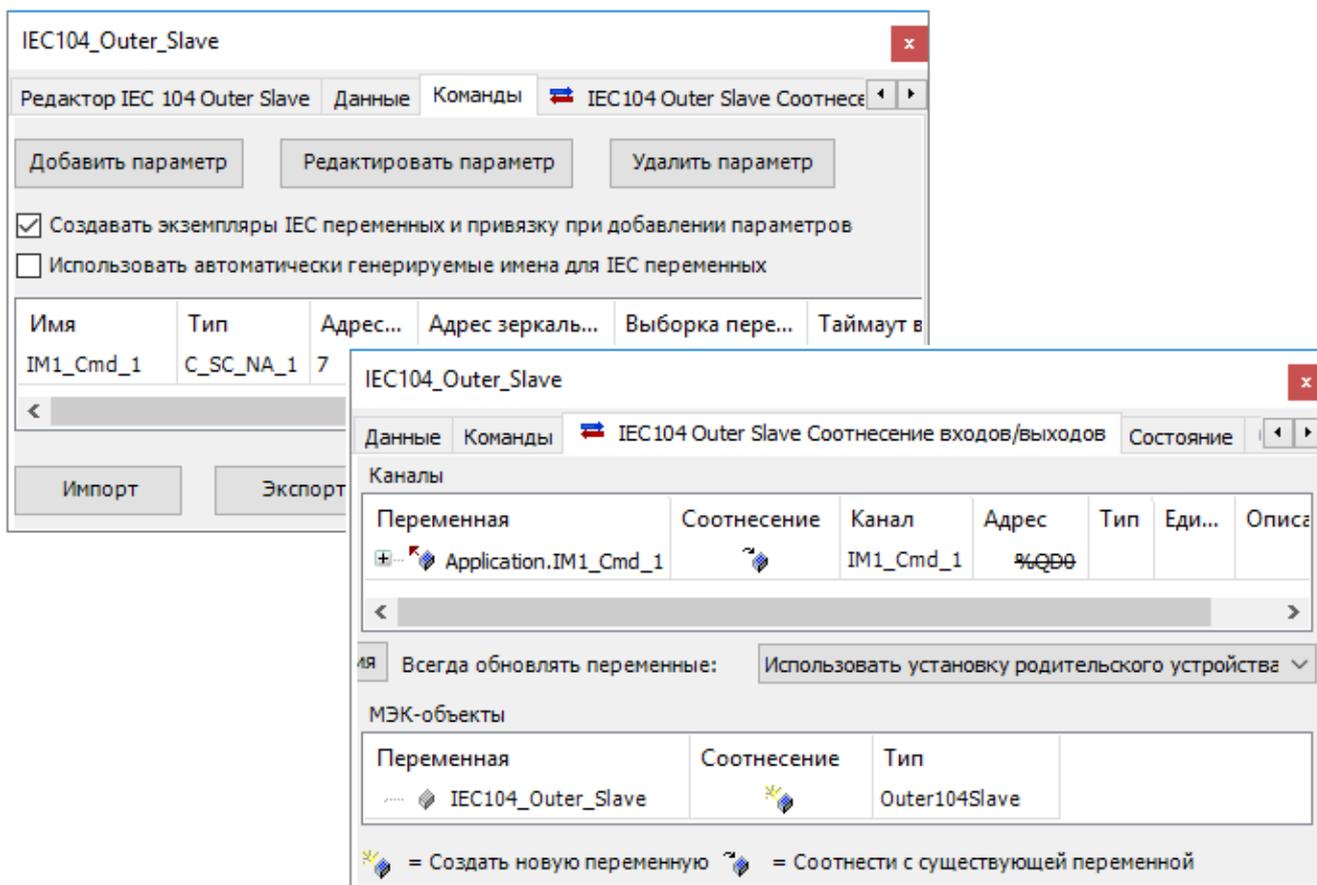


Рисунок 26. Автоматическое создание переменной и автоматическая привязка ее к каналу

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.

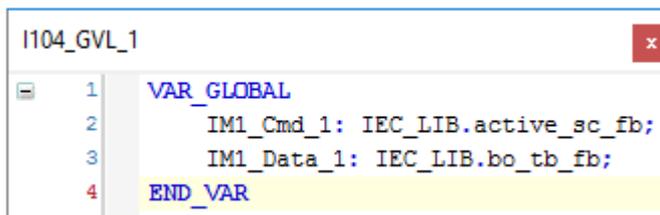


Рисунок 27. Пример описания переменных в списке I104_GVL_1

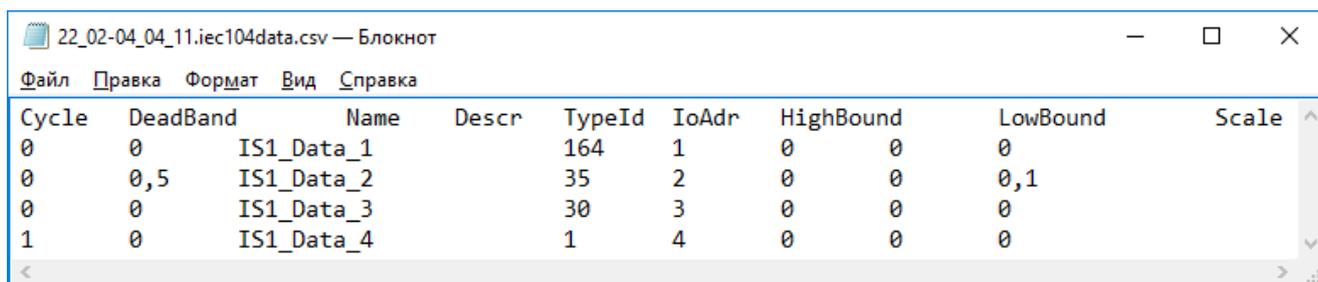
ЭКСПОРТ И ИМПОРТ КОНФИГУРАЦИИ IEC-104

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

Конфигурацию контроллера в качестве Slave, экспортированную в файл, можно импортировать и использовать при настройке контроллера в качестве Master.

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 settings**. Определите папку, в которой будет храниться файл конфигурации IEC-104, 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 28. Структура файла конфигурации для экспорта/импорта данных

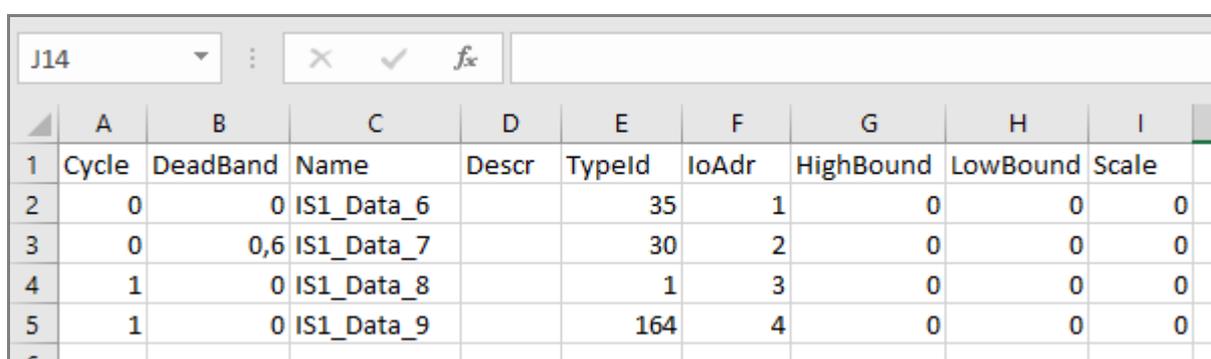
В файле конфигурации присутствуют следующие поля:

- **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 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

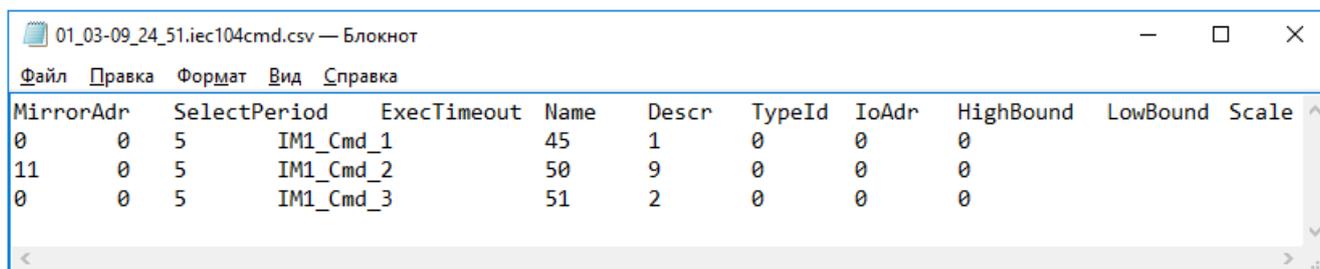
Рисунок 29. Создание в электронной таблице списка переменных IEC-104 для последующего импорта в проект

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 30 показана структура файла конфигурации для экспорта/импорта команд.



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

Рисунок 30. Структура файла конфигурации для экспорта/импорта команд

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 31).

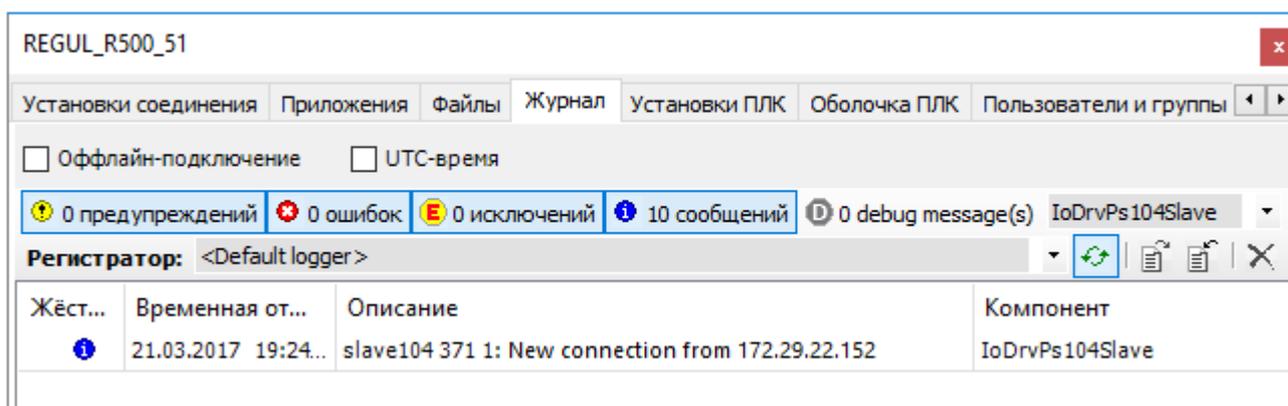


Рисунок 31. Сообщение от компонента IoDrvPs104Slave

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 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 32, Рисунок 33).

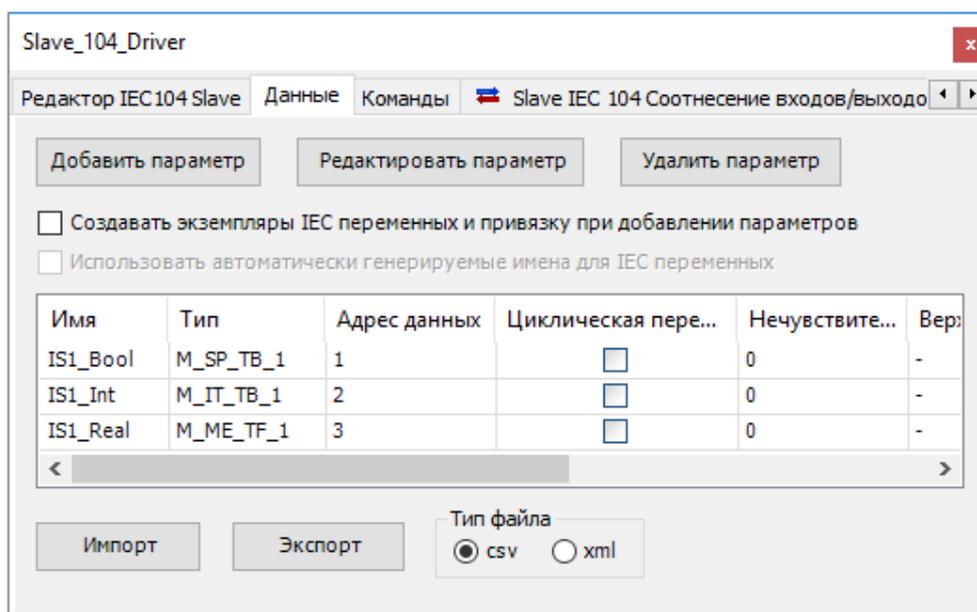


Рисунок 32. Пример списка элементов данных

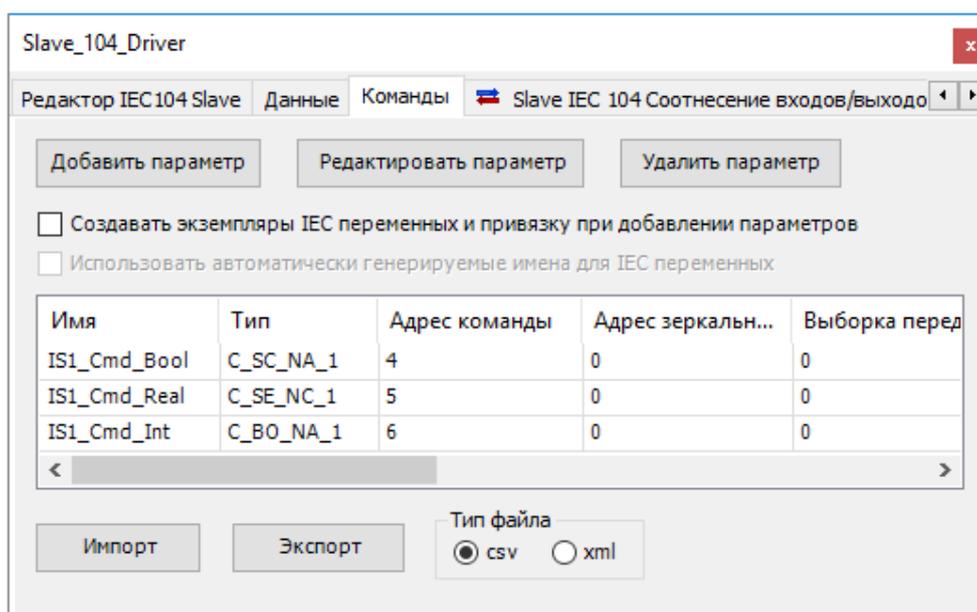


Рисунок 33. Пример списка команд

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_Boolean.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_Boolean.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 34).

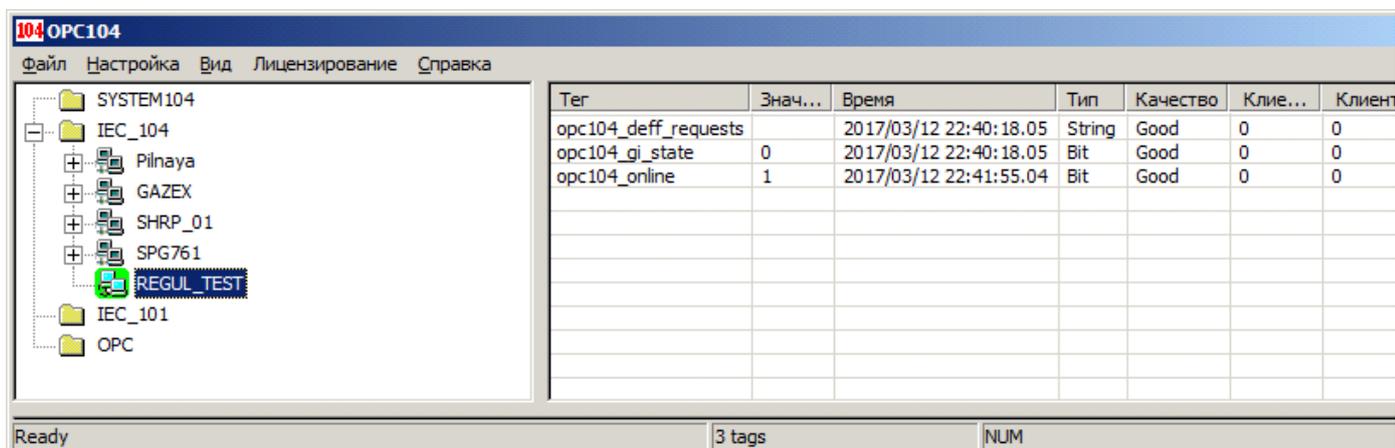


Рисунок 34. Программный шлюз-конвертор OPC IEC 60870-5-104 (МЭК-104)

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 **OK** (Рисунок 35). Предполагается, что остальные настройки узла соответствуют тем, которые заданы на контроллере.

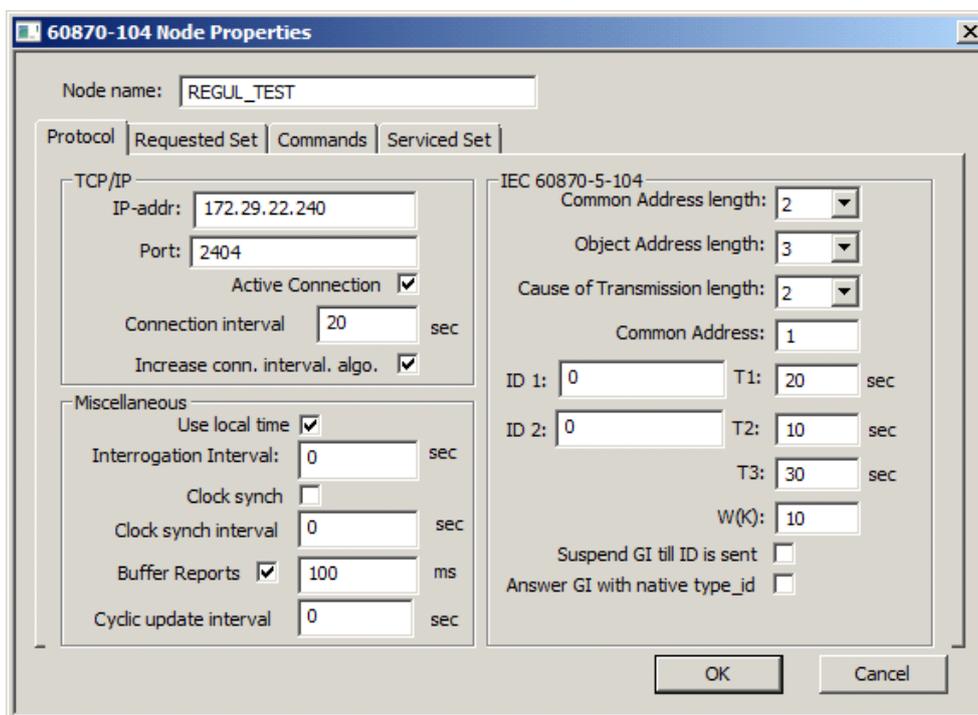


Рисунок 35. Установка параметров узла

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 36).

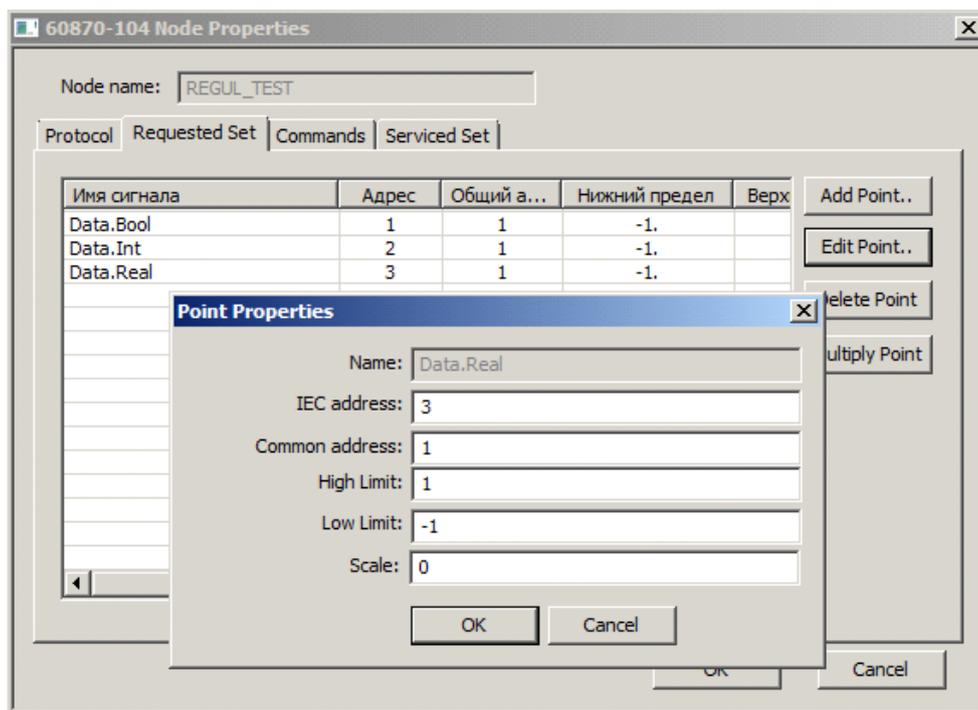


Рисунок 36. Параметры элемента данных

Перейдите на вкладку **Commands**, где опишите соответствующие команды – для каждой команды нужно указать тип и адрес (Рисунок 37). Типы и адреса соответствуют командам, описанным в контроллере.

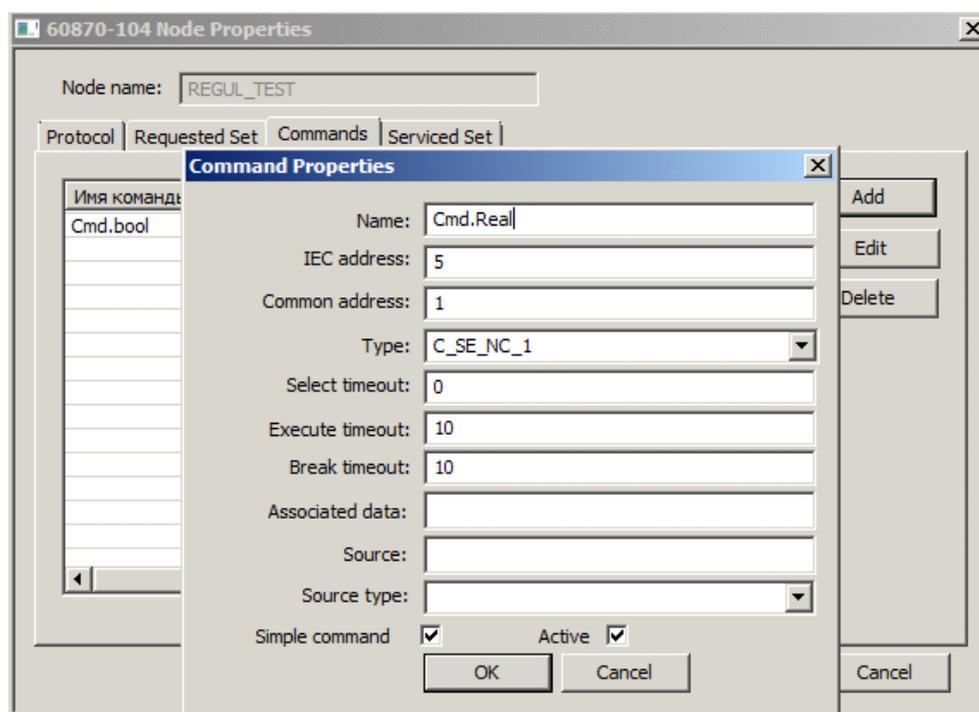


Рисунок 37. Параметры команды

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.