



Контроллер программируемый логический Элсима

РУКОВОДСТВО ПО ЭКСПЛУАТАЦИИ

Страниц 233

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

март 2019

Литера

СОДЕРЖАНИЕ

СПИСОК ТЕРМИНОВ И СОКРАЩЕНИЙ.....	6
ИНФОРМАЦИЯ О ДОКУМЕНТЕ	8
УКАЗАНИЕ МЕР БЕЗОПАСНОСТИ	9
1 ХАРАКТЕРИСТИКИ И УСТРОЙСТВО КОНТРОЛЛЕРА.....	10
1.1 Назначение.....	10
1.2 Параметры электромагнитной совместимости.....	11
1.3 Технические характеристики контроллера.....	12
1.4 Аппаратный состав.....	16
1.5 Условное наименование и маркировка.....	18
1.6 Установочные размеры и монтаж контроллера	19
1.7 Общая конструкция контроллера	20
1.8 Монтаж внешних подключений	22
1.8.1 Общие требования к монтажным проводникам и их подключение.....	22
1.8.2 Подключение питания.....	22
1.8.3 Подключение соединителей дискретных входов и дискретного выхода	23
1.8.4 Установка SIM-карты.....	23
1.8.5 Использование GSM/GPRS-модема.....	23
1.8.6 Установка microSD-карты	23
1.8.7 Подключение к USB2-порту	23
1.8.8 Подключение к релейным выходам	23
1.8.9 Подключение интерфейсов RS-485.....	24
1.8.10 Подключение к портам LAN1 и LAN2.....	24
1.8.11 Подключение к порту USB2.....	25
1.8.12 Подключение к соединителям аналогового выхода.....	25
1.8.13 Подключение к соединителям аналоговых входов.....	25
1.9 Выбор режима работы.....	26
1.9.1 Выбор режима работы WatchDog-таймера.....	26
1.9.2 Выбор режима работы (исполнения/программирования).....	26
1.9.3 Перевод в режим настройки сетевых параметров.....	27
1.9.4 Выбор режима старта проекта.....	27
1.10 Индикация.....	27
2 ИСПОЛЬЗОВАНИЕ КОНТРОЛЛЕРА	29
2.1 Условия эксплуатации	29
2.2 Инструменты и ПО для работы.....	29
2.3 Общий порядок работы с контроллером	30
2.4 Распаковывание	30
2.5 Установка и подключение.....	30
2.6 Обновление программного обеспечения контроллера	31
2.7 Установка программного обеспечения.....	31
2.7.1 Общие сведения о системе программирования	31
2.7.2 Установка CoDeSys и пакета поддержки контроллера	32
2.7.3 Обновление пакета поддержки контроллера (TSP)	34
2.7.4 Приведение ранее созданных проектов к текущей версии TSP	35
2.7.5 Установка двух и более версий CoDeSys.....	36
2.8 Создание проекта.....	38
2.8.1 Создание конфигурации	38
2.8.2 Написание кода управляющей программы	51
2.9 Настройка соединения с контроллером	52
2.9.1 Настройка соединения через Ethernet	52
2.9.2 Настройка соединения через USB	53
2.9.3 Настройка шлюза связи.....	53
2.9.4 Проверка настройки соединения	54
2.10 Загрузка проекта в контроллер и отладка	54
2.10.1 Компиляция проекта	54
2.10.2 Выбор активного контроллера	55
2.10.3 Подключение и загрузка проекта.....	56
2.10.4 Запуск и отладка проекта.....	56

2.11 ЗАГРУЗКА РАНЕЕ СОЗДАННОГО ПРОЕКТА.....	57
3 КОНФИГУРИРОВАНИЕ КОНТРОЛЛЕРА.....	58
3.1 ДОБАВЛЕНИЕ МОДУЛЯ В ДЕРЕВО КОНФИГУРАЦИИ.....	58
3.2 НАСТРОЙКА ПАРАМЕТРОВ КОНТРОЛЛЕРА	59
3.2.2 Библиотека для доступа к системной информации контроллера.....	61
3.3 ПРОГРАММНЫЙ МОДУЛЬ MODBUS TCP MASTER.....	63
3.3.1 Общий принцип конфигурирования Modbus TCP Master	65
3.3.2 Модуль <i>Server</i> для MBTCPM.....	66
3.3.3 Рекомендации по работе с модулем MBTCPM.....	75
3.4 ПРОГРАММНЫЙ МОДУЛЬ MODBUS TCP SLAVE.....	76
3.4.1 Настройка конфигурационных параметров модуля <i>Slave</i>	78
3.4.2 Конфигурирование передачи данных по Modbus TCP Slave	79
3.4.3 Настройка статистических и диагностических параметров и соотнесение сигналов	85
3.4.4 Рекомендации по работе с модулем MBTCP S.....	85
3.5 ПРОГРАММНЫЙ МОДУЛЬ MODBUS RTU MASTER.....	86
3.5.1 Общий принцип конфигурирования Modbus RTU Master	86
3.5.2 Настройка модуля MBMRTU	86
3.5.3 Модуль MBMRTU <i>Server (Slave)</i> (для Modbus RTU Master).....	88
3.5.4 Настройка конфигурационных параметров модуля MBMRTU <i>Server (Slave)</i>	89
3.5.5 Конфигурирование базы сигналов протокола Modbus RTU для ведомого устройства	89
3.5.6 Настройка статистических и диагностических параметров и соотнесение сигналов	94
3.5.7 Рекомендации по работе с модулем MBMRTU.....	94
3.6 ПРОГРАММНЫЙ МОДУЛЬ MODBUS RTU SLAVE.....	95
3.6.1 Общий принцип конфигурирования Modbus RTU Slave.....	96
3.6.2 Настройка модуля MBRTUS	96
3.6.3 Модуль <i>Server</i>	97
3.6.4 Настройка конфигурационных параметров модуля <i>Server</i>	98
3.6.5 Конфигурирование карты опроса по протоколу Modbus RTU	98
3.6.6 Настройка статистических и диагностических параметров и соотнесение сигналов	103
3.6.7 Рекомендации по работе с модулем MBRTUS.....	103
3.7 ПРОГРАММНЫЙ МОДУЛЬ ОПРОСА СЧЕТЧИКОВ ЭЛЕКТРОЭНЕРГИИ СЭТ4ТМ03М и ПСЧ-4ТМ.05 МК (МД)	104
3.7.1 Общий принцип конфигурирования модуля опроса счетчиков	104
3.7.2 Настройка модуля <i>ELMicronMst</i>	104
3.7.3 Модули <i>SET4TM</i> и <i>PSH4TM</i>	106
3.7.4 Настройка конфигурационных параметров модулей <i>SET4TM</i> и <i>PSH4TM</i>	106
3.7.5 Конфигурирование базы сигналов модулей <i>SET4TM</i> и <i>PSH4TM</i>	107
3.7.6 Настройка статистических и диагностических параметров и соотнесение сигналов	116
3.8 РАБОТА СО СЧЕТЧИКАМИ ЭЛЕКТРОЭНЕРГИИ ЭНЕРГОМЕРА СЕ301/302/303/304.....	117
3.8.1 Конфигурирование ФБ <i>CE303</i>	117
3.8.2 Инициализация ФБ <i>CE30X</i> и <i>CE30X_v2</i>	118
3.8.3 Выполнение транзакции ФБ <i>CE30X</i>	119
3.8.4 Описание работы ФБ	124
3.9 РАБОТА СО СЧЕТЧИКАМИ ЭЛЕКТРОЭНЕРГИИ МЕРКУРИЙ 230/233/234	125
3.9.1 Конфигурирование ФБ <i>M23X</i>	125
3.9.2 Инициализация ФБ <i>M23X</i>	126
3.9.3 Выполнение транзакции ФБ <i>M23X</i>	127
3.9.4 Описание работы ФБ	131
3.10 ПРОГРАММНЫЙ МОДУЛЬ СИНХРОНИЗАЦИИ ВРЕМЕНИ	132
3.10.1 Настройка конфигурационных параметров модуля <i>tsync</i>	133
3.10.2 Выбор типа синхронизации	135
3.10.3 Настройка диагностических параметров и соотнесение сигналов	137
3.11 РАБОТА С МОДУЛЕМ GSM ДЛЯ ПРИЁМА И ПЕРЕДАЧИ SMS СООБЩЕНИЙ	139
3.11.1 Конфигурирование модуля <i>GSM</i>	140
3.12 ИНТЕРФЕЙС LAN1	149
3.12.1 Настройка интерфейса <i>Ethernet</i>	150
3.12.2 Настройка параметров <i>Lan1</i>	150
3.12.3 Добавление IP-слота и настройка его параметров.....	152
3.12.4 Добавление коммуникационного слота <i>CommSlot</i>	153
3.13 УПРАВЛЕНИЕ СЕТЕВОЙ ПОДСИСТЕМОЙ КОНТРОЛЛЕРА (КОННЕКТОР NETCONTROL)	154
3.13.1 Задание статической маршрутизации сетевых интерфейсов (<i>StaticRoute</i>).....	154
3.14 НАСТРОЙКА ИНТЕРФЕЙСА RS-485	158

3.14.1	Настройка параметров модуля RS485	158
3.14.2	Связывание интерфейса RS-485 с программным модулем	160
3.15	ПЕРИФЕРИЙНЫЕ УСТРОЙСТВА	160
3.16	РАБОТА С SD-КАРТОЙ И USB-FLASH	161
3.16.1	Доступ к SD-карте через CoDeSys	161
3.16.2	Работа с SD-картой из задачи пользователя	164
3.16.3	Запись логов на SD-карту	165
3.16.4	Доступ к USB-Flash через CoDeSys	165
3.16.5	Работа с USB-Flash из задачи пользователя	168
3.16.6	Запись логов на USB-Flash	169
3.17	РАБОТА С СИГНАЛАМИ ВХОДОВ/ВЫХОДОВ КОНТРОЛЛЕРА	169
3.17.1	Сигналы дискретного входа	171
3.17.2	Сигналы дискретного выхода	173
3.17.3	Сигналы аналогового входа	174
3.17.4	Сигналы аналогового выхода	178
3.18	РАБОТА СО СКАНЕРАМИ QR И ШТРИХ-КОДА	180
3.18.1	Основные функции ПО	180
3.18.2	Подключение сканера к контроллеру	181
3.18.3	Работа с ФБ библиотеки USBScanLibrary	181
3.19	ПОДДЕРЖКА РЕЖИМА GPRS GSM МОДУЛЯ	184
3.19.1	Назначение режима GPRS	184
3.19.2	Варианты использования модуля GSM в GPRS режиме	184
3.19.3	Конфигурирование работы модуля GSM в GPRS режиме	185
3.19.4	Управление работой модуля GSM в GPRS режиме	187
3.19.5	Пример программного кода	189
3.20	ОТОБРАЖЕНИЕ СОСТОЯНИЙ ПРОГРАММНЫХ И АППАРАТНЫХ МОДУЛЕЙ	189
3.20.1	Отображение состояний программных и аппаратных модулей в журнале CoDeSys	189
4	КОМПЛЕКТ ПОСТАВКИ	190
5	ТРАНСПОРТИРОВАНИЕ И ОБСЛУЖИВАНИЕ	191
5.1	ТАРА И УПАКОВКА	191
5.2	ТРАНСПОРТИРОВАНИЕ И ХРАНЕНИЕ	191
5.3	КАЛИБРОВКА	191
5.4	ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ	192
5.5	ТЕКУЩИЙ РЕМОНТ	192
6	РЕШЕНИЕ ПРОБЛЕМ	192
	ПРИЛОЖЕНИЕ А (СПРАВОЧНОЕ) СХЕМЫ ПОДКЛЮЧЕНИЯ СИГНАЛОВ КОНТРОЛЛЕРА	193
	ПРИЛОЖЕНИЕ Б (СПРАВОЧНОЕ) ИЗМЕНЕНИЕ СЕТЕВЫХ ПАРАМЕТРОВ КОНТРОЛЛЕРА	196
	ПРИЛОЖЕНИЕ В (СПРАВОЧНОЕ) ПОДДЕРЖИВАЕМЫЕ ТИПЫ ДАННЫХ	198
	ПРИЛОЖЕНИЕ Г (СПРАВОЧНОЕ) УСТАНОВКА ДРАЙВЕРА MICROSOFT RNDIS	199
	ПРИЛОЖЕНИЕ Д (СПРАВОЧНОЕ) ОПИСАНИЕ ПРИМЕНЕНИЯ ФУНКЦИОНАЛЬНЫХ БЛОКОВ MARIN, MAROUT	204
	ПРИЛОЖЕНИЕ Е (СПРАВОЧНОЕ) ПРИМЕР ПРИМЕНЕНИЯ ФУНКЦИЙ ФБ SE30X	207
	ПРИЛОЖЕНИЕ Ж (СПРАВОЧНОЕ) ПРИМЕР ПРИМЕНЕНИЯ ФУНКЦИЙ ФБ M23X И ФБ M23X_V2	211
	ПРИЛОЖЕНИЕ И (СПРАВОЧНОЕ) ПРИМЕР ПРИМЕНЕНИЯ ФУНКЦИЙ ФБ GSM	215
	ПРИЛОЖЕНИЕ К (СПРАВОЧНОЕ) ПРИМЕР ПРОГРАММНОГО КОДА ДЛЯ РАБОТЫ СО СКАНЕРАМИ	218
	ПРИЛОЖЕНИЕ Л (СПРАВОЧНОЕ) ОБНОВЛЕНИЕ СИСТЕМНОГО ПО КОНТРОЛЛЕРА	220
	ПРИЛОЖЕНИЕ М (СПРАВОЧНОЕ) ОБНОВЛЕНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ПО КОНТРОЛЛЕРА	222
	ПРИЛОЖЕНИЕ Н (СПРАВОЧНОЕ) ПЕРЕЧЕНЬ СООБЩЕНИЙ О СОСТОЯНИЯХ ПРОГРАММНЫХ И АППАРАТНЫХ МОДУЛЕЙ	224
	ПРИЛОЖЕНИЕ П (СПРАВОЧНОЕ) ПЕРЕЧЕНЬ ИЗМЕНЕНИЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И РЭ НА КОНТРОЛЛЕР ЭЛСИМА	228

Список терминов и сокращений

AC	– Alternating Current – Переменный ток;
CoDeSys	– Система программирования <i>CoDeSys</i> ;
CRC	– Cyclic redundancy check (контроль с помощью циклического избыточного кода) – Алгоритм вычисления контрольной суммы, предназначенный для проверки целостности данных;
DMA	– Direct Memory Access – Прямой доступ к памяти;
DC	– Direct Current – Постоянный ток;
FAT	– File Allocation Table – Таблица распределения файлов;
FBD	– Function Block Diagram – Функциональная блочная диаграмма;
GMT	– Greenwich Mean Time – Время по Гринвичу;
GVL	– Global Variable List – Список глобальных переменных;
IEC	– International Electrotechnical Commission, См. также МЭК;
LD	– Ladder Diagram – Релейно-контактная схема;
POU	– Program Organization Unit – Компонент организации программ, программный компонент;
RAM	– Random Access Memory – Память (запоминающее устройство) с произвольной выборкой; оперативное запоминающее устройство. См. также ОЗУ;
ROM	– Read-Only Memory – Энергозависимая память, используемая при хранении неизменяемых данных;
SMS	– Short Message Service – Сервис отправки коротких текстовых сообщений (3GPP TS 27.005);
TCP	– Transmission Control Protocol – Протокол управления передачей данных;
TCP/IP	– Transmission Control Protocol/Internet Protocol – Межсетевой протокол управления передачей данных;
TSP	– Target Support Package – Пакет поддержки контроллера;
WDT	– Watchdog timer – Программируемый сторожевой таймер;
AT-команды	– Modem Hayes command prefix ("for Attention") – Набор команд, разработанных компанией Hayes для модема, состоит из серий коротких текстовых строк, которые объединяют вместе, чтобы сформировать полные команды операций, таких как набор номера, начала соединения или изменения параметров подключения;
АЦП	– Аналого-цифровой преобразователь;
ЗИП	– Запасные части, инструменты и принадлежности;
ИС	– Исполняющая система;
Кадр	– Количество информации, состоящей из переменного числа байт передаваемой/получаемой протоколом за один раз;
"Клиент" ("Master")	– Устройство, расположенное в пункте управления и являющееся потребителем данных и осуществляющее сбор данных с КП всей системы телемеханики;
Контроллер	– Контроллер программируемый логический Элсима;
Маппинг	– Mapping – Процесс назначения переменных сигналам конфигурации для дальнейшего осуществления доступа к сигналам из управляющей программы <i>CoDeSys</i> ;
Модуль УВВ	– Модуль удаленного ввода-вывода;
МЭК	– Международная электротехническая комиссия. См. также IEC;
ОЗУ	– Оперативное запоминающее устройство; оперативная память. См. также RAM;

Оператор сотовой связи	–	Оператор сотовой связи – Организация, обеспечивающая предоставление доступа к услугам сотовой связи через абонентский терминал (GSM-модем, сотовый телефон). Оплата услуг осуществляется путём покупки SIM-карты и пополнения баланса за используемые услуги;
ОС	–	Операционная система;
ПК	–	Персональный компьютер;
ПЛК	–	Контроллер программируемый логический;
ПО	–	Программное обеспечение;
Поллинг	–	(англ. polling) Опрос – Метод предоставления доступа к среде для обеспечения информационного обмена между ведущим и подчинённым устройством путем выдачи периодических запросов к устройствам согласно таблице поллинга; используется одна таблица (очередь), адресованное устройство получает право использования канала передачи данных в течение заданного тайм-аутом времени. Таблица поллинга – таблица (список), определяющая порядок опроса подчинённых устройств и необходимые параметры запроса;
ПСЧ	–	Счетчик электрической энергии многофункциональный ПСЧ;
РЭ	–	Руководство по эксплуатации;
"Сервер" ("Server", "Slave")	–	Устройство, расположенное на контролируемом пункте системы телемеханики, осуществляющее сбор данных с технологического оборудования и являющееся поставщиком данных в информационную сеть;
СЭТ	–	Счетчик электроэнергии трехфазный;
Термопара	–	Термопреобразователь напряжения;
Управляющая программа	–	Один или несколько взаимосвязанных программных компонентов, реализованных на языках программирования ИЕС 61131-3 и определяющих логику работы контроллера;
ФБ	–	Функциональный Блок – Основной элемент для построения программ для контроллера программируемого логического;
ЦАП	–	Цифро-аналоговый преобразователь;
ЦП	–	Центральный процессор;
ЭНП	–	Энергонезависимая память – память, предназначенная для долговременного хранения переменных.

Информация о документе

В настоящем руководстве по эксплуатации (РЭ) содержится информация, необходимая пользователю для правильной и безопасной эксплуатации программируемого логического контроллера Элсима (далее – контроллер).

В данном документе представлено описание контроллера Элсима в металлическом корпусе, который относится к ревизии 2.0!

Персонал, проводящий работы с контроллером, должен иметь необходимую квалификацию для работы с электронным оборудованием и программным обеспечением, а также с объектами, которыми управляет контроллер.

Алгоритмы работы контроллера с объектом управления обеспечиваются программой, разработанной пользователем. Изготовитель не несет ответственности за ущерб, принесенный вследствие ошибочно составленной пользовательской программы.

Данные, предоставленные в документе, проверены на соответствие аппаратному и программному обеспечению на момент поставки контроллера. В связи с текущим совершенствованием продукции и документации, пользователю целесообразно следить за проводимыми обновлениями через сайт производителя.

Авторские права на настоящий документ принадлежат компании АО "ЭлеСи". Копирование и распространение настоящего документа без письменного разрешения владельца авторских прав запрещено.

Контактная информация:

- почтовый адрес: **АО "ЭлеСи"**, 634021, г. Томск, ул. Алтайская, 161а;
- тел. (3822) 601-000, факс (3822) 601-001;
- официальный сайт компании: www.elesy.ru.

Указание мер безопасности

- Сохранность технических характеристик при эксплуатации и хранении, постоянная готовность контроллера к работе обеспечиваются при строгом соблюдении требований настоящего руководства по эксплуатации и знании принципа работы контроллера. Для исключения выхода контроллера из строя из-за неправильных действий или нарушения условий безопасной работы перед началом работы необходимо внимательно изучить настоящее руководство по эксплуатации.

- Эксплуатация контроллера должна производиться в соответствии с «Правилами технической эксплуатации электроустановок потребителей», "Правил по охране труда при эксплуатации электроустановок" и главой 7.3 ПУЭ.

- Контроллер соответствует требованиям безопасности ГОСТ ИЕС 60950-1-2014, ГОСТ 12.2.003-91, ГОСТ 12.2.007.0-75, ТР ТС 004/2011.

- По способу защиты от поражения электрическим током контроллер соответствует классу II по ГОСТ ИЕС 60950-1-2014.

- Запрещается эксплуатировать контроллер со снятыми или имеющими повреждения корпусными деталями.

- Контроллер не предназначен для использования во взрывоопасной зоне.

- Контроллер удовлетворяет нормам промышленных радиопомех, установленным для оборудования класса А по ГОСТ 30428-96 и ГОСТ 30805.22-2013, и не должен применяться в жилых, коммерческих и производственных зонах с малым энергопотреблением и подключаться к низковольтным распределительным электрическим сетям.

- Все работы в процессе эксплуатации необходимо проводить с применением мер защиты от статического электричества, не допуская ударов и приложения больших усилий при стыковке разъемов.

- Запрещается эксплуатировать контроллер в помещениях с химически агрессивной средой.

1 Характеристики и устройство контроллера

1.1 Назначение

Контроллер предназначен для работы в малых системах автоматизации с количеством входных-выходных сигналов не более 100.

Алгоритм работы контроллера определяется управляющей программой, разрабатываемой пользователем в соответствии с требованиями к системе управления, создаваемой с использованием контроллера.

Контроллер представляет собой функциональное законченное изделие, имеющее необходимое количество входных и выходных каналов, конфигурируемых на различные виды входных и выходных сигналов, включая:

- дискретный вход сигналов различной полярности относительно общего проводника;
- дискретный выход сигналов типа "Общий коллектор" и "Сухой контакт";
- вход непрерывных сигналов, представленных напряжением постоянного тока, постоянным током, термопреобразователями и импульсными сигналами;
- выход непрерывных сигналов, представленных напряжением постоянного тока и постоянным током.

При необходимости, для увеличения количества сигналов, контроллер позволяет подключать модули удаленного ввода-вывода серии Элсима и аналогичные.

Основная область применения – малые системы автоматического и автоматизированного управления технологическими процессами в областях, таких как, управление климатическим оборудованием, управление малыми станками и механизмами, автоматизация котельных, работа в системах "Умный дом" и других отраслях.

Контроллер обеспечивает непрерывный необслуживаемый режим работы в условиях естественной вентиляции.

Сведения о сертификации приводятся на электронном носителе, входящем в комплект поставки изделия.

Метрологические характеристики контроллера устанавливаются в соответствии с ГОСТ 22261-94.

По эксплуатационной законченности контроллер относится к изделиям второго порядка по ГОСТ Р 52931-2008.

1.2 Параметры электромагнитной совместимости

Контроллер удовлетворяет критерию качества функционирования А по требованиям устойчивости к воздействию электромагнитных помех в соответствии с ГОСТ CISPR 24-2013, ГОСТ 30804.6.2-2013 по следующим типам воздействий:

- уровень электростатического разряда в соответствии с ГОСТ 30804.4.2-2013, степень жесткости 1;
- радиочастотное электромагнитное поле в соответствии с ГОСТ 30804.4.3-2013, степень жесткости 2;
- наносекундные импульсные помехи по цепи электропитания в соответствии с ГОСТ 30804.4.4-2013, степень жесткости 3;
- микросекундные импульсные помехи большой энергии по цепям электропитания в соответствии с ГОСТ Р 51317.4.5-99, степень жесткости 2;
- динамические изменения напряжения сети электропитания в соответствии с ГОСТ 30804.4.11-2013, класс электромагнитной обстановки 3;
- колебания напряжения питания ГОСТ Р 51317.4.14-2000, класс электромагнитной обстановки 3.

Контроллер удовлетворяет нормам промышленных радиопомех класса А по ГОСТ 30428-96 и ГОСТ 30805.22-2013.

1.3 Технические характеристики контроллера

Технические характеристики контроллера указаны в таблице 1.1.

Таблица 1.1 – Контроллер Элсима. Технические характеристики

Наименование параметра	Значение	
	Элсима-M01-ZZZU	Элсима-M01-ZZZU-GSM
Тип процессора	Cortex ARM8	
Частота процессора	300 МГц	
Максимальный размер RAM	128 Мбайт	
Максимальный размер ROM	128 Мбайт	
Максимальный объем памяти для хранения задачи пользователя	32 Мбайт	
Максимальный объем энергонезависимой памяти (ЭНП), доступной к задаче пользователя	27 Кбайт	
Скорость выполнения инструкций:		
• Логическая инструкция	0,01 мкс	
• Арифметическая операция с целыми числами	0,02 мкс	
• Арифметическая операция с числами формата Real	0,03 мкс	
Наличие часов реального времени со съемной батареей	есть	
Время автономной работы часов реального времени	5 лет	
Точность хода часов реального времени (с отключенным питанием)	3 с в сутки	
Наличие четырехпозиционного DIP-переключателя, состояние которого считывается программно	есть	
Габаритные размеры контроллера, мм, не более:	170×116×57	170×116×57 (без учета GSM-антенны)*
Масса, не более	0,4 кг	
Аппаратный WatchDog-таймер		
Первый период сброса WatchDog-таймера	70 с	
Второй и последующий периоды сброса WatchDog-таймера	от 0,9 до 2,5 с	
Возможность аппаратного отключения WatchDog-таймера	есть	
Интерфейсы контроллера		
Количество разъемов для подключения Ethernet 10/100 Mbi	2 шт.	
Напряжение гальванического разделения от цепей модулей, не менее	1000 В AC	
Наличие встроенного GSM-модема	нет	есть
Количество разъемов подключения по интерфейсу RS-485, цепи А, В, подключение экрана	1 шт.	
• Максимальная скорость обмена	115200 бит/с	
• Гальваническая развязка, не менее	750 В AC	
Работа с модулями удаленного ввода-вывода (УВВ)		
Количество одновременно подключенных модулей УВВ, не более	4 шт.	
Скорость обновления данных с модулей УВВ	от 20 мс	
Возможность работы в общих сетях Ethernet	есть, по предустановленному IP-адресу	
Дополнительное оборудование		
Разъем USB для подключения внешних устройств в режиме host	1 шт.	
Разъем USB для подключения внешних устройств в режиме device	1 шт.	
Тип SD-карты	microSD	
Объем microSD-карты	от 2 до 32 Гбайт	
Дискретные входы контроллера		
Количество гальванически развязанных групп	2 группы	
Количество дискретных входов	20 шт.	

Таблица 1.1 – Контроллер Элсима. Технические характеристики

Наименование параметра	Значение	
	Элсима-M01-ZZZU	Элсима-M01-ZZZU-GSM
Напряжение логического нуля	от минус 3 до плюс 5 В	
Напряжение логической единицы	от 15 до 30 В	
Максимальный ток логической единицы	10 мА	
Минимальная детектируемая длительность импульса, не менее	1,5 мс	
Минимальный период следования импульсов	3 мс	
Напряжение гальванического разделения между дискретными входами и внутренней шиной контроллера (эффективное значение), не менее	1500 В	
Дискретные выходы контроллера		
Количество дискретных выходов тип "Открытый коллектор" (одна группа)	4 шт.	
Общая гальваническая изоляция от внутренней шины контроллера (эффективное значение) выходов типа "Открытый коллектор", не менее	1500 В	
Максимальное коммутируемое напряжение для выходов "Открытый коллектор"	30 В	
Остаточное напряжение в состоянии "включено" для выходов "Открытый коллектор", не более	1 В	
Максимальный коммутируемый ток для выходов "Открытый коллектор"	0,3 А	
Количество гальванически разделенных групп дискретных выходов, тип "Реле"	2 группы	
Количество дискретных выходов, тип "Реле", в одной группе	2 шт.	
Максимальное коммутируемое напряжение для релейных выходов	250 В АС	
Максимальный коммутируемый ток для релейных выходов	2 А	
Гальваническая развязка от внутренней шины контроллера (эффективное значение) групп релейных выходов, не менее	2000 В	
Примечание – Предусмотрена защита от подачи напряжения обратной полярности и выбросов напряжения при коммутации индуктивной нагрузки для выходов типа "Открытый коллектор"		
Аналоговые входы		
Количество универсальных аналоговых входов	4 шт.	
Гальваническая развязка от внутренней шины контроллера (эффективное значение) каждого аналогового входа, не менее	750 В	
Возможность подключать датчики с сигналами следующих типов: <ul style="list-style-type: none"> • Ток • Напряжение • Термопары типа: <ul style="list-style-type: none"> <input type="checkbox"/> ТХА (К) <input type="checkbox"/> ТХК (L) <input type="checkbox"/> ТХКн (E) <input type="checkbox"/> ТПП10 (S) <input type="checkbox"/> ТНН (N) <input type="checkbox"/> ТПР (B) <input type="checkbox"/> ТЖК (J) <input type="checkbox"/> ТВР (A-1) <input type="checkbox"/> ТПП13 (R) • Термосопротивления в режиме трехпроводного подключения типа: <ul style="list-style-type: none"> <input type="checkbox"/> ТСМ (50М, 100М, 500М) <input type="checkbox"/> ТСП (50П, 100П, 500П, 1000П, Pt50, Pt100) <input type="checkbox"/> ТСН (100Н, 500Н, 1000Н) 	0-20 мА 0-10 В от минус 250 до плюс 900 °С от 0 до плюс 800 °С от минус 250 до плюс 1000 °С от 0 до плюс 1700 °С от минус 250 до плюс 1000 °С от плюс 250 до плюс 1800 °С от минус 200 до плюс 600 °С от 0 до плюс 2500 °С от 0 до плюс 1600 °С от минус 50 до плюс 150 °С от минус 50 до плюс 500 °С от минус 50 до плюс 150 °С	

Таблица 1.1 – Контроллер Элсима. Технические характеристики

Наименование параметра	Значение	
	Элсима-M01-ZZZU	Элсима-M01-ZZZU-GSM
Пределы допускаемой абсолютной погрешности измерений термопары в рабочих условиях (при задании параметра <i>ModeFrec="Disable"</i>)	см. таблицу 1.2	
Пределы допускаемой приведенной погрешности измерений термосопротивления, в рабочих условиях, не более	±0,5 %	
Пределы допускаемой приведенной погрешности измерений постоянного тока, в рабочих условиях, не более	±0,2 %	
Пределы допускаемой приведенной погрешности измерений напряжения постоянного тока, в рабочих условиях, не более	±0,25 %	
Минимальное время измерения одного канала для входных сигналов постоянного тока или напряжения постоянного тока, температуры в режиме измерения сигнала термопары	25 мс	
Минимальное время выполнения одного канала в режиме измерения температуры термопреобразователем сопротивления в трехпроводном режиме	200 мс	
Выходное напряжение встроенного источника питания для подключения датчиков с контролем целостности цепи для исполнения по напряжению питания 24 В DC	соответствует значению входного напряжения	
Максимальный ток нагрузки встроенного источника питания	0,3 А	
Аналоговые выходы		
Количество аналоговых выходов	2 шт.	
Количество групп аналоговых выходов	1 группа	
Гальваническая развязка группы от внутренней шины контроллера (эффективное значение), не менее	750 В	
Каждый аналоговый выход может быть программно сконфигурирован для работы в следующих режимах: <ul style="list-style-type: none"> • Ток (с внешним шунтом) • Напряжение 	0-20 мА от 0 до 10 В	
Пределы допускаемой приведенной погрешности формирования выходного сигнала, не более	±0,5 %	
Максимальное нагрузочное сопротивление аналогового выхода (R) при выходном токовом сигнале	400 Ом	
Минимальное нагрузочное сопротивление аналогового выхода (R) при выходном сигнале "Напряжение 0-10 В"	2000 Ом	
Цепи питания		
Питание контроллера	20...28 В DC	
Потребляемая мощность (без учета потребления датчиков, подключенных к встроенному источнику питания), не более	7 Вт	
<p>П р и м е ч а н и я</p> <p>1 ZZZU – исполнение контроллера по напряжению питания и по типу внешних соединителей.</p> <p>2 * Максимальные габаритные размеры контроллера исполнения Элсима-M01-ZZZU-GSM зависят от положения GSM-антенны</p>		

Пределы допускаемой абсолютной погрешности измерений термопары в рабочих условиях приведены в таблице 1.2.

Таблица 1.2 – Пределы допускаемой абсолютной погрешности измерений термопары в рабочих условиях

Характеристика термопары	Диапазон преобразования	Δ , °C
ТХА (К) -250...+900	от -250 до -100 °C	$\pm 6,9$
	от -100 до 0 °C	$\pm 5,175$
	от 0 до +600 °C	$\pm 3,45$
	от 600 до 900 °C	$\pm 4,6$
ТХК (L) 0...+800	от 0 до +200 °C	$\pm 2,4$
	от +200 до +400 °C	$\pm 2,0$
	от +400 до +800 °C	$\pm 1,6$
ТХК _n (E) -250...+1000	от -250 до -100 °C	$\pm 6,25$
	от -100 до 0 °C	$\pm 5,0$
	от 0 до +250 °C	$\pm 3,75$
	от +250 до +1000 °C	$\pm 2,5$
ТПП10 (S) 0...+1700	от 0 до +400 °C	$\pm 5,1$
	от +400 до +800 °C	$\pm 6,8$
	от +800 до +1300 °C	$\pm 8,5$
	от +1300 до +1700 °C	$\pm 10,2$
ТНН (N) -250...+1000	от -250 до 0 °C	$\pm 8,75$
	от 0 до +250 °C	$\pm 6,25$
	от +250 до +500 °C	$\pm 5,0$
	от +500 до +1000 °C	$\pm 3,75$
ТПР (B) +250...+1800	от +250 до +450 °C	$\pm 7,75$
	от +450 до +1350 °C	$\pm 5,425$
	от +1350 до +1800 °C	$\pm 6,2$
ТЖК (J) -200...+600	от -200 до 0 °C	$\pm 2,4$
	от 0 до +600 °C	$\pm 1,6$
ТВР (A-1) 0...+2500	от 0 до +2500 °C	$\pm 7,5$
ТПП13 (R) 0...+1600	от 0 до +400 °C	$\pm 6,4$
	от +400 до +1200 °C	$\pm 4,8$
	от +1200 до +1600 °C	$\pm 6,4$

Руководство по эксплуатации

Контроллер предназначен для работы в климатических условиях, указанных в таблице 1.3.

Таблица 1.3 – Контроллер Элсима. Климатические условия эксплуатации контроллера

Наименование параметра	Значение
Диапазон рабочих температур	от 0 до плюс 60 °С
Относительная влажность воздуха	от 50 до 95 % (при температуре плюс 40 °С)
Атмосферное давление	от 84,0 до 106,7 кПа (от 630 до 800 мм рт. ст.)

Контроллер устойчив к синусоидальной вибрации согласно ГОСТ ИЕС 61131-2-2012 (с частотой перехода 8,4 Гц) с параметрами, представленными в таблице 1.4.

Таблица 1.4 – Контроллер Элсима. Параметры синусоидальной вибрации

Наименование параметра	Значение
Частота	от 5 до 150 Гц
Максимальное ускорение	1,0 g
Максимальное смещение	3,5 мм
Удары с параметрами: • амплитуда • длительность • форма ударной волны	до 15 g 11 мс полусинусоида

1.4 Аппаратный состав

Контроллер разработан в металлическом корпусе, устанавливаемом на DIN-рейку. Набор доступных исполнений контроллера представлен в таблице 1.5.

Таблица 1.5 – Состав и исполнение контроллера

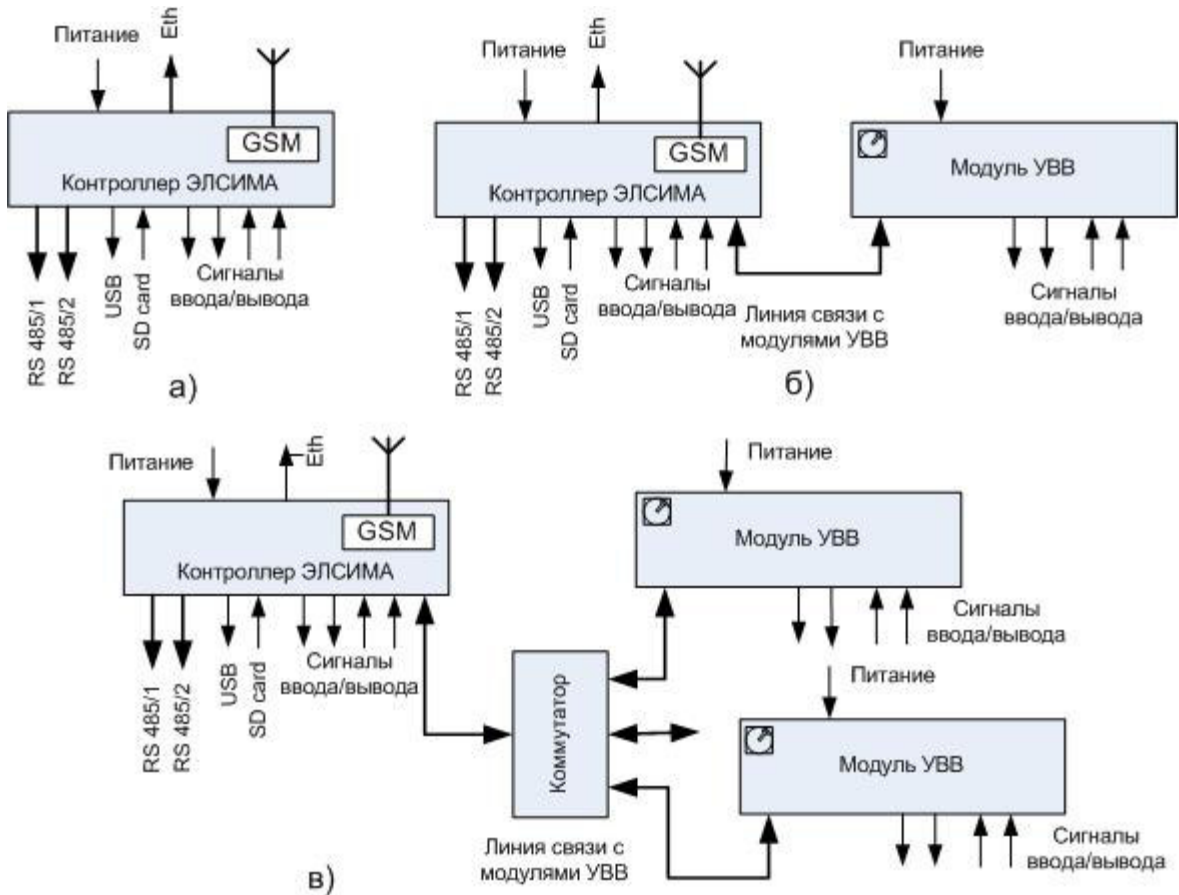
Наименование	Вариант исполнения	Корпус	Назначение
Элсима-М01	Элсима-М01-24P Элсима-М01-24P-GSM	Металлический	Контроллер программируемый логический

Примечание – В документе представлено описание контроллера Элсима в металлическом корпусе, который относится к ревизии 2.0!

На рисунке 1.1 представлены варианты построения системы с расширением входных-выходных каналов контроллера Элсима с помощью модулей УВВ. Существует возможность создания трех схем распределенных систем:

- схема подключения без модулей УВВ – рисунок 1.1, а);
- схема подключения одного модуля УВВ непосредственно к контроллеру – рисунок 1.1, б);
- схема подключения более одного модуля УВВ с использованием коммутатора – рисунок 1.1, в).

Существует возможность подключения не более четырех модулей УВВ. Модули УВВ возможно подключать через общие сети *Ethernet*, при этом не гарантируются временные показатели работы. При работе в выделенной сети адрес модуля УВВ задается переключателем на лицевой панели. При работе в общей сети адрес модуля определяется заданным заранее IP-адресом. Назначение IP-адреса производится при настройке модуля УВВ.



а) без модулей УВВ; б) с одним модулем УВВ; в) несколько модулей УВВ

Рисунок 1.1 – Структурные схемы систем на базе контроллера Элсима и модулей УВВ

1.5 Условное наименование и маркировка

Условное наименование контроллера приведено на рисунке 1.2.

Контроллер программируемый логический Элсима	M	YY	ZZZ	U	XXXX
Основное функциональное назначение: M – модификация контроллера					
Порядковый номер разработки					
Напряжение цепей питания: – 24 – 24 В DC					
Тип внешних соединителей P – разъёмы					
Наличие встроенного GSM/GPRS-модема: – нет символа – нет; – GSM – есть					

Рисунок 1.2 – Условное наименование контроллера

Доступные исполнения контроллера приведены в таблице 1.5. Примеры наименований:

- Элсима M01-24P – контроллер, порядковый номер разработки 01, исполнение для работы от 24 В постоянного тока, с разъемными соединителями.

Маркировка контроллера соответствует ГОСТ 26828-86 и содержит:

- условное наименование контроллера;
- наименование предприятия-изготовителя и (или) логотип компании;
- знак утверждения типа (для сертифицированных исполнений, см. рисунок 1.4);
- единый знак обращения продукции на рынке;
- наименование страны-изготовителя;
- матричный код, содержащий заводской номер и дату выпуска изделия, расшифровка матричного кода;
 - сведения о напряжении питания и выходной мощности;
 - маркировку переключателей, индикаторов (кроме индикаторов интерфейса *Ethernet*), разъемов;
- условное обозначение наличия встроенного GSM/GPRS-модема.

1.6 Установочные размеры и монтаж контроллера

Контроллер изготавливается в металлическом корпусе для крепления на DIN-рейку. Габаритно-установочные размеры контроллера представлены на рисунках 1.3, 1.4.

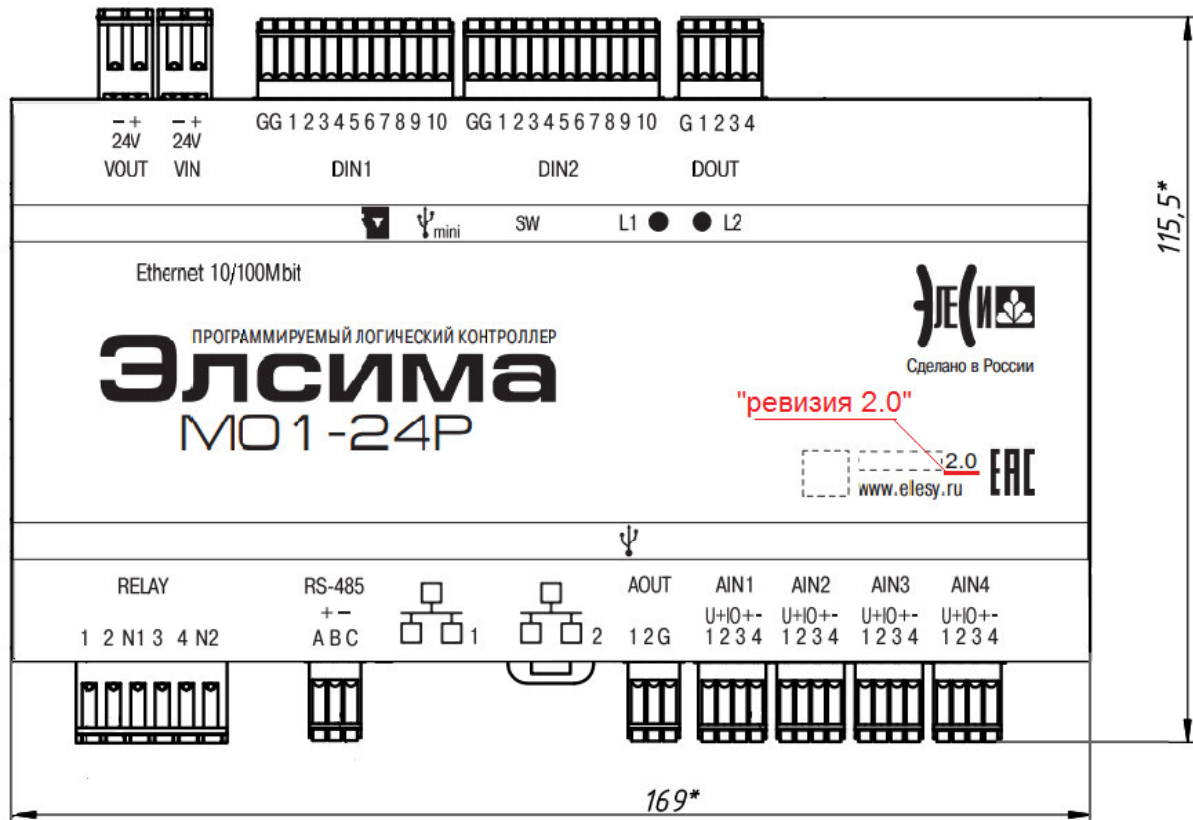


Рисунок 1.3 – Контроллер Элсима. Габаритно-установочный чертеж. Вид спереди

Контроллер устанавливается на DIN-рейку в соответствии с рисунком 1.4 с помощью фиксирующей защелки.

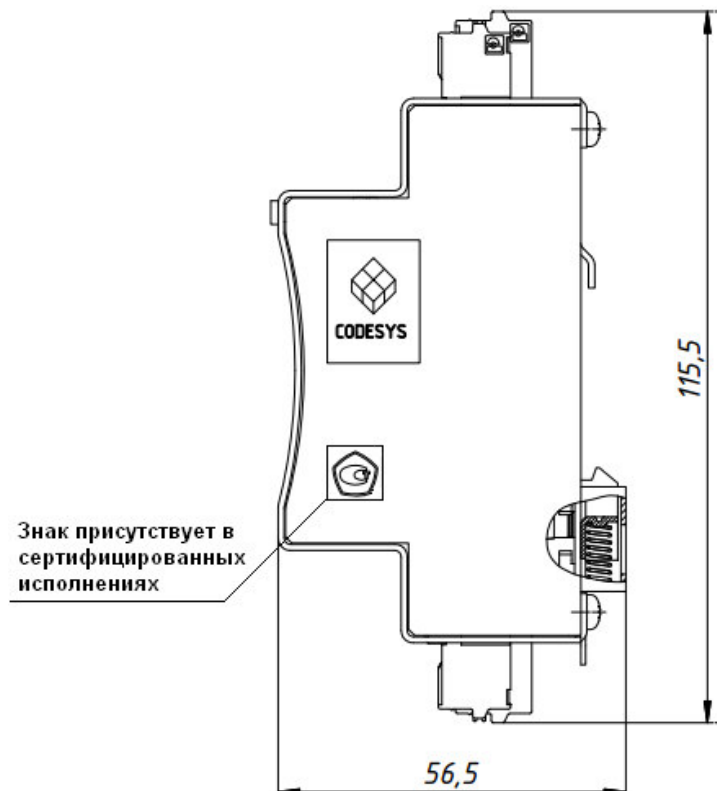


Рисунок 1.4 – Контроллер. Габаритно-установочный чертеж. Вид сбоку

1.7 Общая конструкция контроллера

На лицевой панели контроллера (рисунок 1.5) расположены:

- Индикаторы состояний контроллера:
 - "L1" – двухцветный индикатор работы контроллера (красного и зеленого цвета свечения);
 - "L2" – индикатор состояния контроллера (желтый цвет свечения);
 - "L3" – индикатор работы GSM/GPRS модема (желтого цвета), присутствует только в исполнении Элсима-M01-GSM;

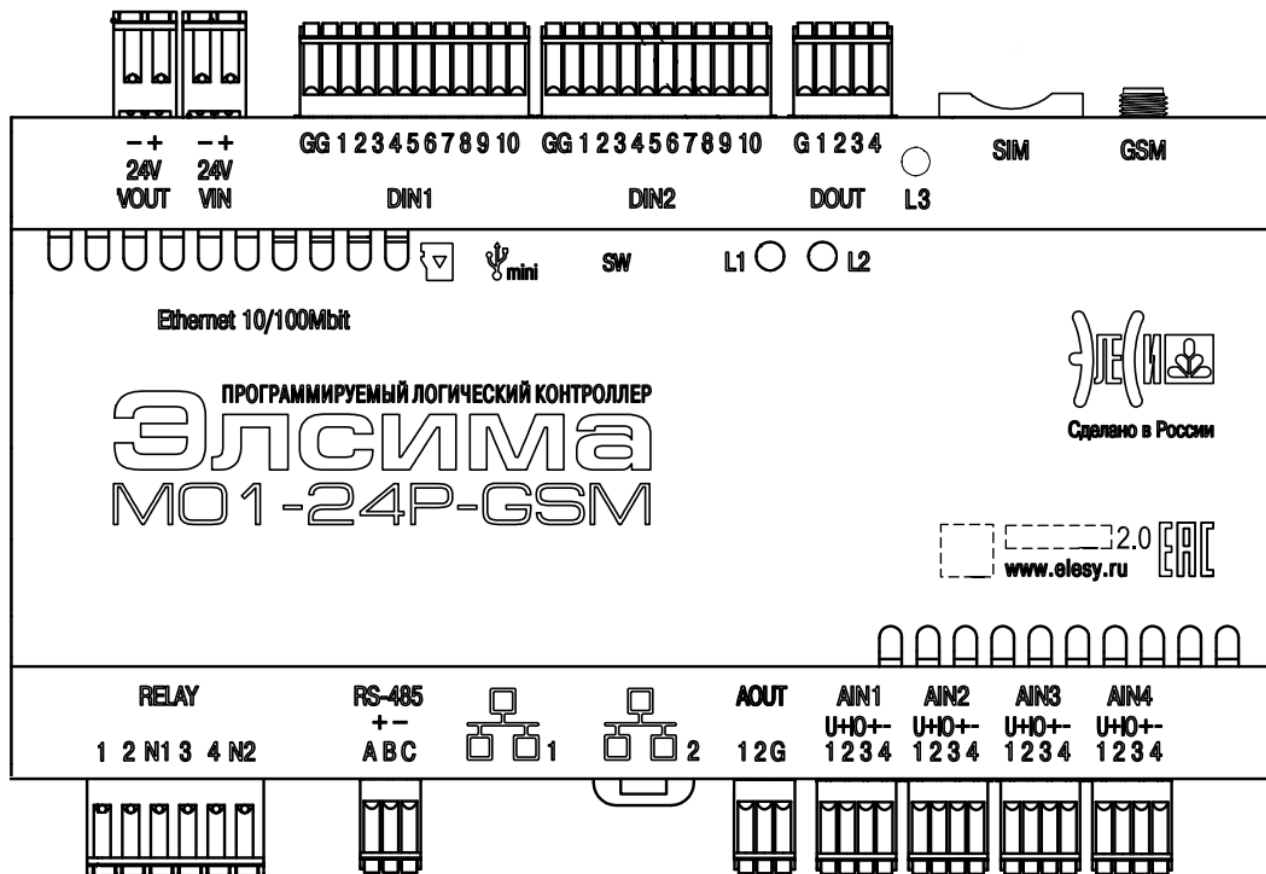


Рисунок 1.5 – Вид лицевой панели контроллера (антенна GSM/GPRS модема не показана)

На боковой стороне контроллера (далее – "верхняя сторона") расположены следующие элементы:

- "microSD" – слот для подключения карт памяти типа microSD;
- "USB mini" – порт USB для подключения внешних устройств по протоколу USB в режиме *Slave*;
- "SW" – четырехпозиционный DIP-переключатель "SW". Описание положений переключателя представлено на рисунке 1.16;
- "VOUT" – разъемный соединитель выходного питания;
- "VIN" – разъемный соединитель входного питания;
- "DIN1" и "DIN2" – разъемные соединители дискретных входов 1 и 2;
- "DOUT" – разъемный соединитель дискретного выхода;
- "SIM" – слот для подключения SIM-карты;
- "GSM" – разъем SMA-F для подключения внешней антенны GSM/GPRS-модема.

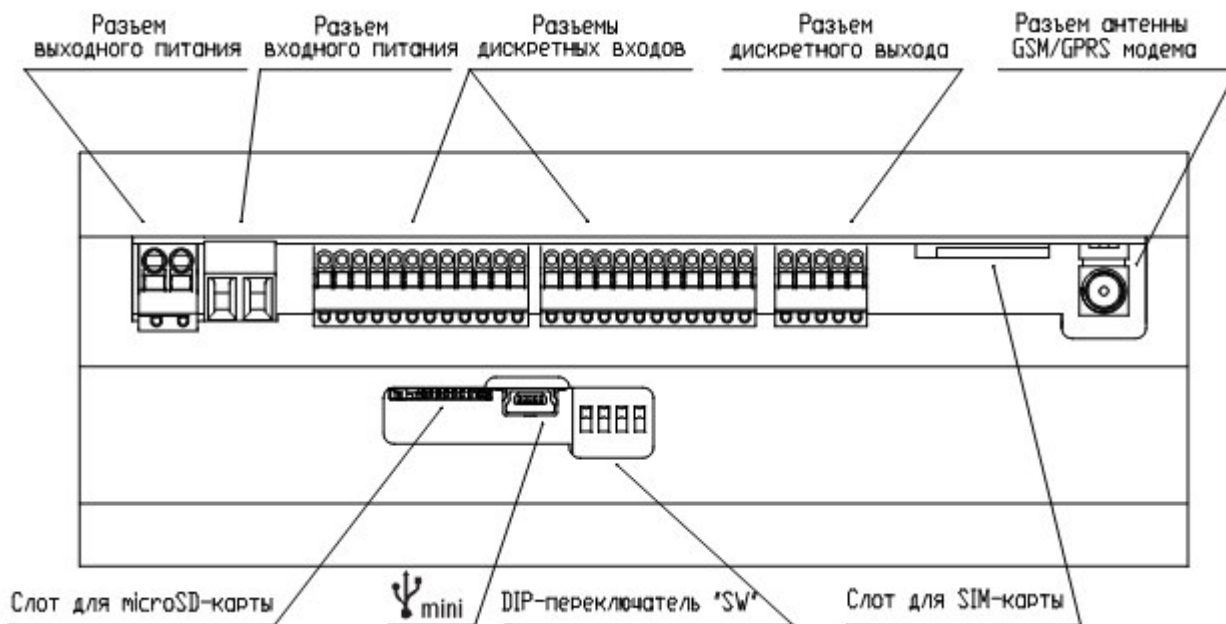


Рисунок 1.6 – Контроллер. Вид сбоку

На противоположной боковой стороне контроллера (далее – "нижняя сторона") расположены следующие элементы:

- "RELAY" – разъемный соединитель релейных дискретных выходов;
- "RS-485" – разъем соединителей для подключения внешних приборов по интерфейсу RS-485;

- "□ □ 1", "□ □ 2" – соединитель порта LAN (LAN1 и LAN2);
- "USB" – порт USB для подключения внешних устройств по протоколу *USB 2.0* в режиме **host**;

- "AOUT" – разъемный соединитель аналогового выхода;
- "AIN1", "AIN2", "AIN3", "AIN4" – разъемные соединители аналоговых входов.

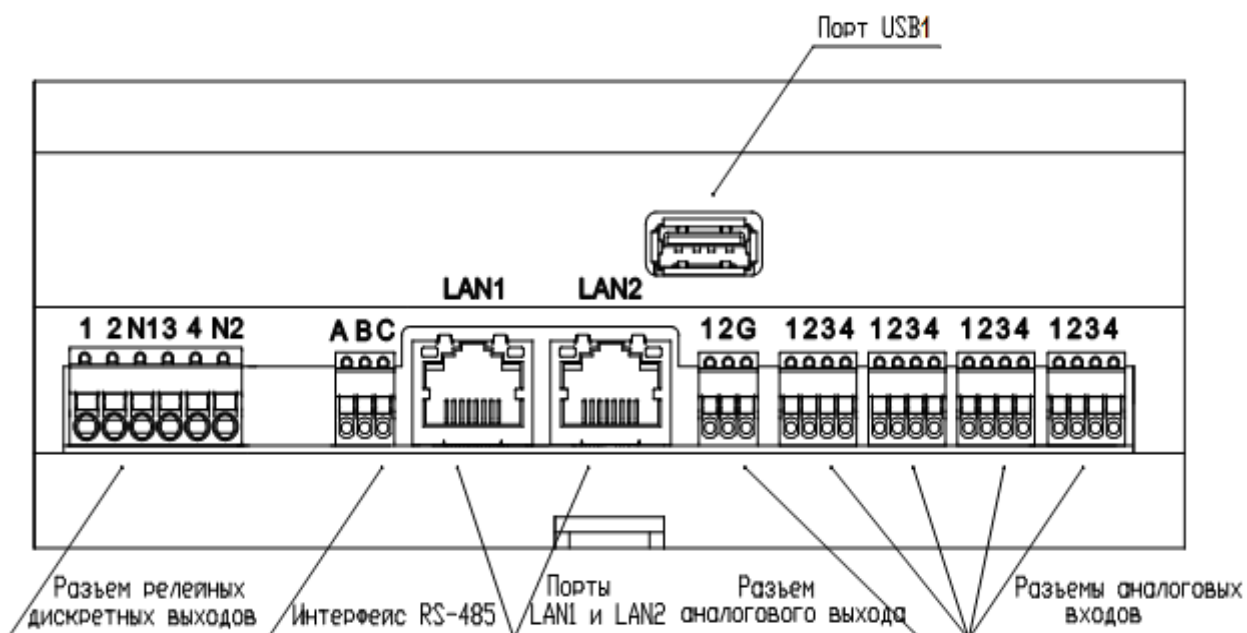


Рисунок 1.7 – Контроллер. Вид с другой (противоположной) боковой стороны (маркировка контактов показана условно)

1.8 Монтаж внешних подключений

1.8.1 Общие требования к монтажным проводникам и их подключение

Для подключения сигнальных цепей допускается использование гибких изолированных проводников сечением от 0,2 до 0,5 мм² (для разъемов RELAY, VIN, VOUT – 0,5 до 1,5 мм²).

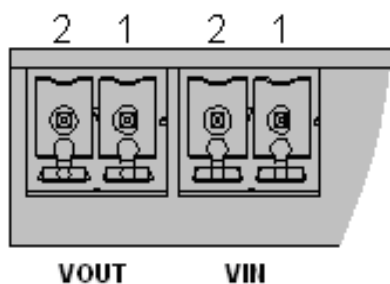
Для подключения проводников к ответной части разъема следует:

- 1 Проверить, что все подключаемые к контроллеру цепи обесточены.
- 2 Подсоединить проводник к ответной части разъема. Для этого:
 - 1) Зачистить проводник от изоляции на длину 5–6 мм. Для надежного подключения проводник рекомендуется обжать наконечником.
 - 2) Нажать отверткой на оранжевый пружинный контакт.
 - 3) Вставить проводник в круглое отверстие колодки. Отпустить отверткой пружину и убрать отвертку. Проверить надежность закрепления провода.
- 3 Подсоединить ответную часть к вилке.

ОСТОРОЖНО! Не допускается выход оголенных участков проводников над изолятором колодки.

1.8.2 Подключение питания

Разъемы "VOUT" и "VIN" являются соединителями выходного и входного питания контроллера. Назначение контактов, в зависимости от исполнения по напряжению питания, приведено на рисунке 1.8.



Исполнение по напряжению питания +24 В DC			
	Контакт	Обозначение на корп.	Цепь
VOUT	1	+	+24 В
	2	--	GND
VIN	1	+	+24 В
	2	--	GND

Примечание – Напряжение питания выводится на разъем VOUT напрямую с разъема VIN, ограничение тока в данном исполнении модуля не предусматривается.

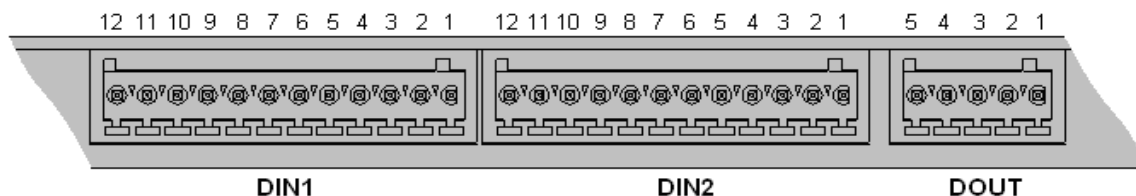
ВНИМАНИЕ! Для исполнения по напряжению питания +24 В DC используйте внешнюю защиту от короткого замыкания! Ток короткого замыкания не должен превышать 4 А!

Рисунок 1.8 – Назначение контактов разъемов "VOUT" и "VIN"

ОСТОРОЖНО! Неверное подключение питающего напряжения приводит к выходу контроллера из строя!

1.8.3 Подключение соединителей дискретных входов и дискретного выхода

Назначение контактов разъемов "DIN1", "DIN2", "DOUT" контроллера представлено на рисунке 1.9.



Контакт	Обозначение на корпусе	Разъем	Контакт	Обозначение на корпусе	Разъем	Контакт	Обозначение на корпусе	Разъем
		DIN1			DIN2			DOUT
1	10	Вход 10	1	10	Вход 10	1	4	Выход 4
2	9	Вход 9	2	9	Вход 9	2	3	Выход 3
3	8	Вход 8	3	8	Вход 8	3	2	Выход 2
4	7	Вход 7	4	7	Вход 7	4	1	Выход 1
5	6	Вход 6	5	6	Вход 6	5	G	Общий
6	5	Вход 5	6	5	Вход 5			
7	4	Вход 4	7	4	Вход 4			
8	3	Вход 3	8	3	Вход 3			
9	2	Вход 2	9	2	Вход 2			
10	1	Вход 1	10	1	Вход 1			
11	G	Общий	11	G	Общий			
12	G	Общий	12	G	Общий			

Рисунок 1.9 – Назначение контактов разъемов "DIN1", "DIN2", "DOUT"

1.8.4 Установка SIM-карты

SIM-карта устанавливается в соответствующий слот "SIM" (рисунок 1.6) контактами в сторону DIN-рейки и скошенным углом вниз.

1.8.5 Использование GSM/GPRS-модема

При использовании GSM/GPRS-модема необходима антенна, устанавливаемая в винтовое гнездо контроллера "GSM" (рисунок 1.6) соответствующего исполнения.

1.8.6 Установка microSD-карты

Карта microSD устанавливается контактами в направлении расположения DIN-рейки в соответствующий слот "SD" (рисунок 1.6) до характерного щелчка. Для извлечения необходимо сверху нажать на карту памяти. Описание работы с картой microSD представлено в подразделе 3.16.

1.8.7 Подключение к USB2-порту

"USB2" – порт USB предназначен для подключения контроллера к ПК для работы с системой программирования *CoDeSys*.

1.8.8 Подключение к релейным выходам

Назначение контактов разъемного соединителя релейных дискретных выходов "RELAY" представлено на рисунке 1.10.



Рисунок 1.10 – Назначение контактов разъема "RELAY"

1.8.9 Подключение интерфейсов RS-485

Подключение приборов по интерфейсу *RS-485* выполняется коммуникационным кабелем – только экранированной витой парой.

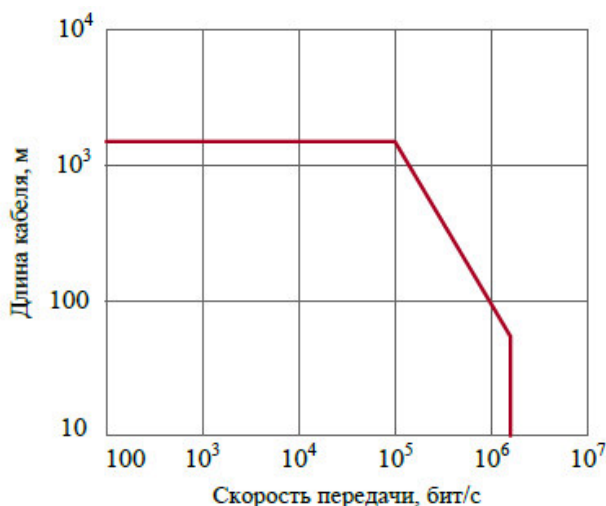
Назначение контактов разъема "RS-485" представлено на рисунке 1.11.



Рисунок 1.11 – Назначение контактов разъема "RS-485"

1.8.9.1 Определение длины кабеля

На рисунке 1.12 представлена функциональная зависимость максимальной скорости передачи от длины кабеля и ограничения на длину кабеля в зависимости от используемой скорости передачи.



Скорость, бит/с	Максимальная длина кабеля, м
от 300 до 57600	1219
115200	1058

Рисунок 1.12 – Зависимость скорости передачи от длины кабеля

Рекомендуемые марки коммуникационного кабеля: МКЭКШВ, КИПЭВ или другие с аналогичными характеристиками.

1.8.10 Подключение к портам LAN1 и LAN2

Назначение и порядок нумерации контактов соединителей портов LAN1 и LAN2 представлено на рисунке 1.13.

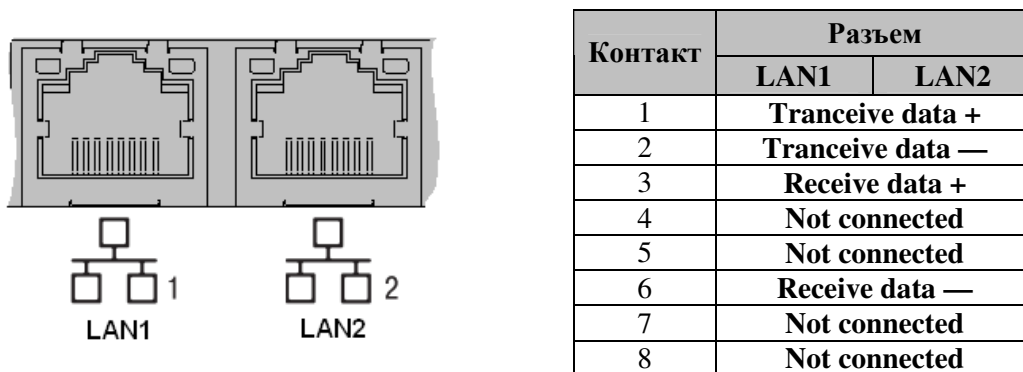


Рисунок 1.13 – Назначение контактов портов LAN1 и LAN2

1.8.11 Подключение к порту USB2

Порт USB2 предназначен для подключения внешних устройств по протоколу *USB 2.0* в режиме **host** (в некоторых версиях ПО может не поддерживаться).

1.8.12 Подключение к соединителям аналогового выхода

Назначение контактов разъема "AOUT" контроллера представлено на рисунке 1.14.

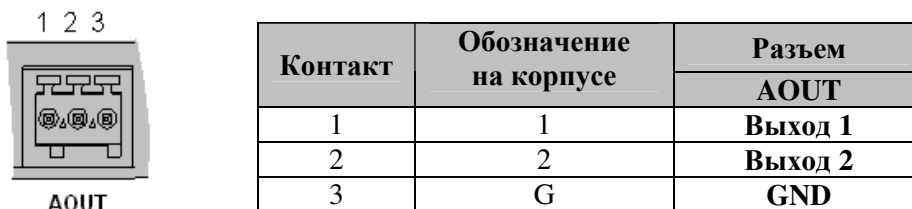


Рисунок 1.14 – Назначение контактов разъема AOUT

1.8.13 Подключение к соединителям аналоговых входов

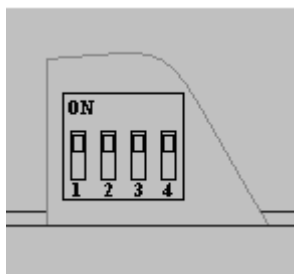
Назначение контактов разъемов "AIN1", "AIN2", "AIN3", "AIN4" контроллера представлено на рисунке 1.15. Схемы подключения приведены в приложении А.



Рисунок 1.15 – Назначение контактов разъемов AIN1-AIN4

1.9 Выбор режима работы

Режим работы контроллера задается с помощью DIP-переключателя SW, расположенного на верхней стороне контроллера (рисунок 1.16).



Переключатель SW	Состояние переключателя	Режим
"1"	"ON"	WatchDog-таймер отключен
	"OFF"	WatchDog-таймер включен
"2"	"ON"	Режим обслуживания (MAINTENANCE MODE) – "сервисный режим работы"
	"OFF"	Производственный режим (PRODUCTION MODE) – "рабочий режим"
"3"	"ON"	Режим настройки сетевых параметров
	"OFF"	Старт в штатном режиме
"4"	"ON"	Включен режим старта контроллера в безопасном режиме
	"OFF"	Выключен режим старта контроллера в безопасном режиме

Рисунок 1.16 – Выбор режима работы

1.9.1 Выбор режима работы WatchDog-таймера

WatchDog-таймер предназначен для автоматического формирования сигнала сброса процессора при подаче питания, а также в случае некорректной работы программного обеспечения. Выбор режима работы WatchDog-таймера контроллера производится установкой переключателя SW "1" в положение "ON" – "OFF" (рисунок 1.16).

1.9.2 Выбор режима работы (исполнения/программирования)

Программное обеспечение контроллера может работать в двух режимах:

- производственный режим (**PRODUCTION MODE**);
- режим обслуживания (**MAINTENANCE MODE**).

Выбор режима работы производится переводом переключателя SW "2" (рисунок 1.16). Штатным (рабочим) режимом работы считается производственный режим. В данном режиме запрещены следующие действия:

- загрузка приложения;
- обновление приложения;
- остановка приложения;
- тёплый сброс приложения;
- холодный сброс приложения;
- запись переменных.

В производственном режиме разрешены следующие действия:

- подключение к контроллеру;
- старт приложения;
- мониторинг переменных.

При установке переключателя SW "2" в положение "ON" (рисунок 1.16), контроллер переходит в режим обслуживания (MAINTENANCE MODE), в котором разрешены функции, запрещенные в производственном режиме (PRODUCTION MODE).

1.9.3 Перевод в режим настройки сетевых параметров

При установке переключателя SW "3" в положение "ON" (рисунок 1.16), контроллер переходит в режим старта с заводскими сетевыми параметрами (значения заводских сетевых параметров см. в приложении Б). В данном режиме не запускается система исполнения пользовательских задач, режим служит только для настройки контроллера. Процесс изменения сетевых параметров приведен в приложении Б.

1.9.4 Выбор режима старта проекта

При отладке пользовательского программного обеспечения может понадобиться необходимость сброса (удаления) созданного проекта (например, при некорректно созданной программе). Для этого существует возможность старта контроллера в "Безопасном" режиме. В данном режиме перед запуском исполняющей системы удаляется ранее созданный проект и производится старт без проекта. Для выбора режима следует установить переключатель SW "4" в положение "ON" (рисунок 1.16).

1.10 Индикация

Расположение и обозначение индикаторов контроллера приведено в 1.7. Описание состояния индикаторов работы контроллера представлено в таблице 1.6.

Таблица 1.6 – Контроллер Элсима. Индикация

Индикатор	Состояние индикатора	Режим работы
"L1"	Красный цвет свечения (непрерывно)	Авария ЦП, проверяется в начальной фазе инициализации системы <i>CoDeSys</i>
"L2"	Не светится	
"L1"	Не светится	Инициализация контроллера
"L2"	Желтый цвет свечения (непрерывно)	
"L1"	Мигание зеленым цветом с периодом 1 с	Система <i>CoDeSys</i> в контроллере запущена, удаляется ранее созданный проект и производится старт без проекта
"L2"	Не светится	
"L1"	Мигание красным цветом с периодом 1 с	Система <i>CoDeSys</i> запущена, не загружен проект в контроллер, произошла исключительная ситуация
"L2"	Не светится	
"L1"	Зеленый цвет свечения (непрерывно)	Система <i>CoDeSys</i> запущена, проект загружен в контроллер и прошла стадия обновления конфигурации (Update configuration), проект не запущен (в состоянии "Стоп")
"L2"	Не светится	
"L1"	Зеленый цвет свечения (непрерывно)	Проект в состоянии исполнения
"L2"	Мигание желтым цветом с периодом 1 с	

Таблица 1.6 – Контроллер Элсима. Индикация

Индикатор	Состояние индикатора	Режим работы
"L1"	Мигание красным и зеленым цветом поочередно с периодом 1 с	Произошла исключительная ситуация после загрузки проекта. В пользовательской задаче возможна фатальная ошибка
"L2"	Не светится	
"L1"	Мигание зеленым цветом с периодом 0,5 с десять раз (одновременно с миганием индикатора "L2" желтым цветом)	Функция идентификации контроллера из системы CoDeSys (функция Wink)
"L2"	Мигание желтым цветом с периодом 0,5 с десять раз (одновременно с миганием индикатора "L1" зеленым цветом)	
"L1"	Зеленый цвет свечения (непрерывно)	Работа в режиме настройки сетевых параметров
"L2"	Желтый цвет свечения (непрерывно)	
"L1"	Мигание зеленым цветом с периодом 3 с	Завершение процесса обновления ПО контроллера
"L2"	Мигание желтым цветом с периодом 3 с (одновременно с "L1")	
"L1"	Зеленый цвет свечения, мигание с периодом примерно 200 мс	Автоматическое обновление ПО контроллера
"L2"	Желтый цвет свечения, постоянно	
"L1"	Красный и зеленый цвета свечения одновременно (непрерывно)	Выход из системы CoDeSys. Данный режим работы возникает только в случае ошибки работы операционной системы, при включенном WDT контроллер будет перезапущен
"L2"	Желтый цвет свечения (непрерывно)	
"L3"	Не светится	GSM-модем отключен
"L3"	Мигание желтым цветом с периодом примерно 1 с	Работа в режиме GSM: нет связи. Возможные причины: отсутствие SIM-карты, отсутствие антенны, испорченная антенна, низкий уровень связи и др. (см. раздел 3.11)
"L3"	Мигание желтым цветом с периодом примерно 3 с	Работа в режиме GSM: установлена связь со станцией (см. раздел 3.11)
"L3"	Мигание желтым цветом с периодом 0,5 с	Работа в GPRS режиме (см. раздел 3.19)

2 ИСПОЛЬЗОВАНИЕ КОНТРОЛЛЕРА

2.1 Условия эксплуатации

Надежная и безопасная работа контроллера обеспечивается при соблюдении следующих ограничений:

- контроллер удовлетворяет нормам промышленных радиопомех, установленным для оборудования класса А по ГОСТ 30428-96 и ГОСТ 30805.22-2013, и не должен применяться в жилых, коммерческих и производственных зонах с малым энергопотреблением и подключаться к низковольтным распределительным электрическим цепям;
- контроллер не предназначен для работы во взрывоопасной зоне;
- не допускается эксплуатация контроллера со снятыми или имеющими повреждения корпусными деталями;
- контроллер должен устанавливаться на вертикальную поверхность с ориентацией в соответствии с рисунками 1.3, 1.4;
- при работе контроллера должна быть обеспечена свободная циркуляция воздуха через отверстия в корпусе;
- напряжение питания контроллера должно соответствовать варианту исполнения источника питания;
- все подключения и отключения цепей к контроллеру допускается производить только после снятия питающих напряжений;
- не допускается попадание на корпус и внутренние части контроллера агрессивных химических веществ и их паров;
- не допускается превышать нормы механических воздействий на контроллер, указанных в таблице 1.4.

2.2 Инструменты и ПО для работы

Для работы с контроллером требуется следующее программное обеспечение:

- система программирования *CoDeSys V3.x Development System* ("3S-Smart Software Solutions");
- пакет поддержки контроллера Элсима "*EleSy ELSYMA TSP_vXX.XX.XXXX*" (АО "ЭлеСи");

Для работы с системой программирования требуется ПК (или ноутбук) с характеристиками, перечисленными в таблице 2.1.

Таблица 2.1 – Аппаратные и системные требования

Требование	Значение
Процессор	Pentium V, Centrino > 3 ГГц Pentium M > 1,5 ГГц (рекомендуется Pentium V, Centrino > 3,5 ГГц, Pentium M > 2,0 ГГц)
ОЗУ (RAM)	2 Гбайт (рекомендуется 4 Гбайт)
Объем свободного места на системном диске	500 Мбайт (рекомендуется 1 Гбайт)
Операционная система	MS Windows XP/7/8

2.3 Общий порядок работы с контроллером

Работа с контроллером осуществляется в следующем порядке:

- 1 Извлечь устройство из упаковки в соответствии с требованиями, указанными в 2.4.
- 2 Собрать контроллер, установить на рабочую поверхность, подключить к сети в соответствии с приведёнными в 2.5 указаниями, подключить необходимые для работы кабели.
- 3 Установить ПО (система программирования *CoDeSys V3.x Development System* и пакет поддержки контроллера Элсима), необходимое для работы с контроллером (см. 2.6).
- 4 Создать проект и управляющую программу для контроллера (см. 2.8).
- 5 Настроить соединения с контроллером (см. 2.9).
- 6 Загрузить созданную программу в контроллер и провести отладку (см. 2.10).

2.4 Распаковывание

Распаковывание контроллера должно производиться в следующем порядке:

1 После получения, длительного хранения или транспортирования контроллеров в групповой транспортной таре произвести внешний осмотр транспортного ящика и проверить целостность упаковки.

2 Перед распаковыванием контроллера после транспортирования при температуре окружающей среды ниже 0 °С необходимо выдержать его в упаковке не менее 6 часов в помещении, в котором он будет эксплуатироваться.

3 Вскрыть транспортный ящик, извлечь из него упаковочную ведомость. Проверить соответствие комплектности упаковочной ведомости.

4 Произвести первичный осмотр контроллера на отсутствие повреждений корпуса, целостности маркировки. Для этого извлечь контроллер из упаковочного ящика и проверить:

- отсутствие видимых механических повреждений, вмятин и следов коррозии составных частей контроллера;
- отсутствие повреждений и загрязнения разъемов;
- состояние и четкость маркировки;

Повторное упаковывание контроллера должно проводиться в соответствии с указаниями, представленными в 5.1.

2.5 Установка и подключение

Порядок установки и подключения контроллера:

1 Установить контроллер на DIN-рейке в соответствии с 1.6. Отклонение от вертикальной оси не должно превышать 15°.

2 Установить требуемый режим работы контроллера с помощью переключателя "SW" согласно указаниям, представленным в 1.9.1–1.9.4.

3 Подключить внешнее питание к контроллеру согласно описанию, представленному в 1.8.2.

4 Подключить в соответствии с маркировкой кабели соединения контроллера с объектами контроля и управления и питающими напряжениями. Схемы подключения приведены в соответствующих разделах данного документа.

5 Подать питание на контроллер.

6 Через несколько секунд устанавливается индикация, соответствующая режиму инициализации, далее индикация, соответствующая рабочему режиму (описание индикации контроллера приведено в таблице 1.6).

2.6 Обновление программного обеспечения контроллера

Для контроллера предусмотрена возможность обновления системного программного обеспечения, пользовательских проектов и некоторых настроечных параметров работы контроллера (например, IP адреса). Обновление производится через заранее подготовленный USB_FLASH диск.

В случае необходимости обновления программного обеспечения контроллера (ЭЛСИМА-М01) следует выполнить действия, описанные в приложении Л.

В случае необходимости обновления пользовательских проектов и некоторых настроечных параметров контроллера (ЭЛСИМА-М01), следует выполнить действия, описанные в приложении М.

ВНИМАНИЕ! Обновление должно производиться квалифицированными и подготовленными специалистами. В случае необходимости следует обратиться в техническую поддержку для предоставления пользователю актуальной версии программного обеспечения (см. контактную информацию)!

2.7 Установка программного обеспечения

2.7.1 Общие сведения о системе программирования

Программное обеспечение контроллера основано на системе разработки *CoDeSys* компании "3S-Smart Software Solutions" (Германия) и предназначено для программирования контроллеров на языках в соответствии со стандартом *IEC 61131-3*.

В базовый состав комплекса *CoDeSys* входят две системы: система разработки и система исполнения. Система разработки функционирует на компьютере и представляет собой инструмент для проектирования, конфигурирования системы и создания кода управляющей программы для ПЛК. Система исполнения (ИС) функционирует в контроллере и обеспечивает загрузку кода прикладной программы в контроллер, исполнение управляющей программы и выполнение отладочных функций.

Базовая версия *CoDeSys* специально адаптирована для функционирования в контроллере Элсима. В дополнение к имеющимся инструментам комплекса разработаны встраиваемые компоненты поддержки контроллера для максимально эффективной разработки прикладных программ.

Разработка прикладных программ в среде *CoDeSys* обеспечивает:

- единую среду конфигурирования, разработки программ и отладки;
- возможность разработки программ на пяти языках программирования IEC 61131-3:
 - IL (Instruction List) – список инструкций;
 - ST (Structured Text) – структурированный текст;
 - FBD (Function Block Diagram) – функциональные блочные диаграммы;
 - LD (Ladder Diagram) – релейно-контактные схемы;
 - SFC (Sequential Function Chart) – последовательные функциональные схемы.

Дополнительно поддержан язык непрерывных функциональных схем CFC;

- прямую генерацию машинного кода, что обеспечивает высокое быстродействие управляющих программ;
- возможность производить отладку программы без привлечения аппаратных устройств благодаря наличию встроенного эмулятора.

2.7.2 Установка *CoDeSys* и пакета поддержки контроллера

Для программирования контроллера Элсима используется система *CoDeSys*. Систему программирования *CoDeSys* можно условно разделить на две части: базовую часть и пакет поддержки контроллера (*TSP*, предоставляется компанией “ЭлеСи”). Следовательно, перед началом работы с контроллером необходимо установить сначала непосредственно систему программирования, а затем добавить в нее пакет поддержки контроллера (*TSP*). *TSP* отличаются функционалом в зависимости от версии.

Примечание – Перед установкой *CoDeSys* 3.5.11.X в системе должен быть установлен *.NET Framework 4.6*.

Порядок установки ПО для программирования контроллера:

1 Установить систему программирования *CoDeSys*.

Для установки системы программирования *CoDeSys* следует запустить файл *Setup_CoDeSysV<Version>.exe* и далее следовать указаниям “Мастера установки” (**InstallShield Wizard**).

2 Установить пакет поддержки контроллера “*EleSy ELSYMA TSP_vXX.XX.XX.XXXXX*” (АО “ЭлеСи”). Для установки пакета следует:

1) Запустить систему программирования *CoDeSys*. Вид стартовой страницы представлен на рисунке 2.1.

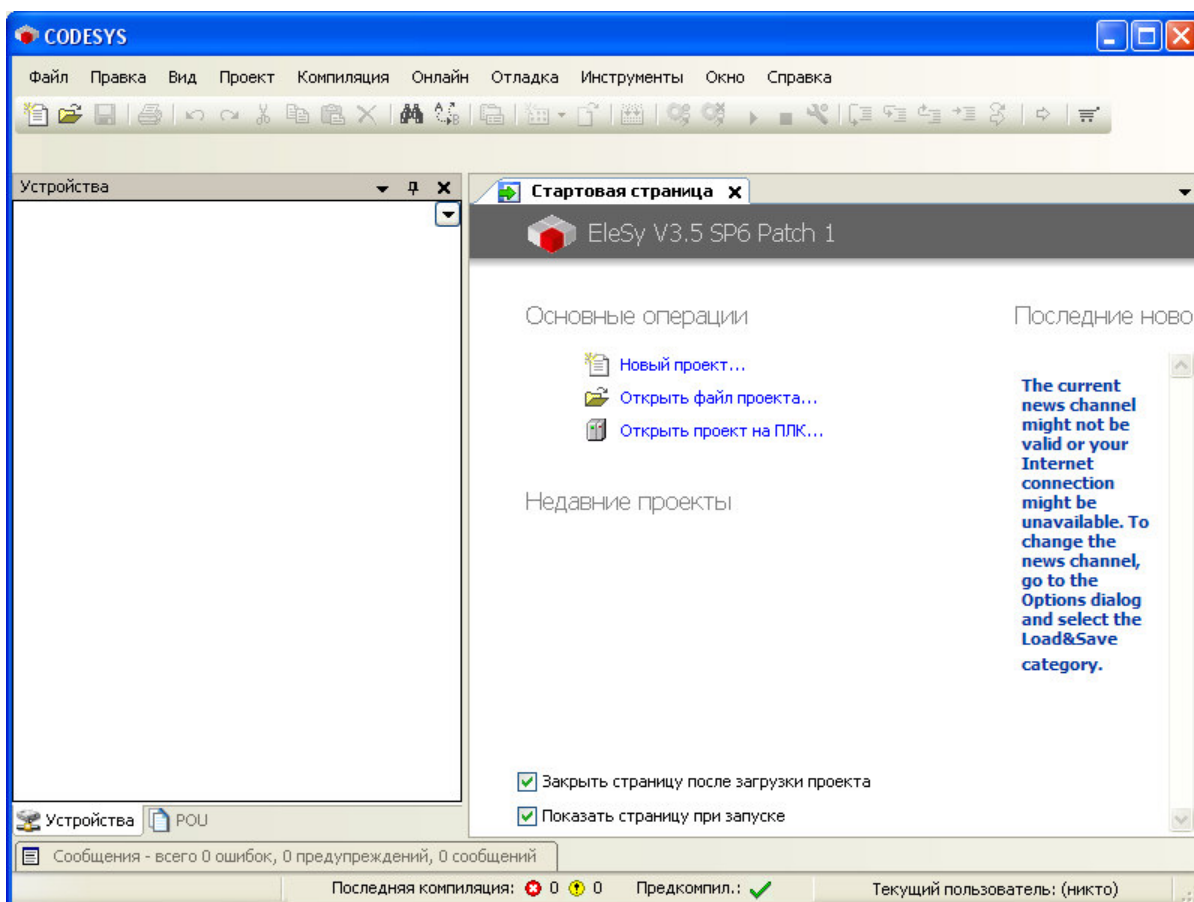


Рисунок 2.1 – Система разработки *CoDeSys*. Вид стартовой страницы

2) В меню **Инструменты** выбрать команду **Менеджер пакетов....** При этом появится окно 2.2.

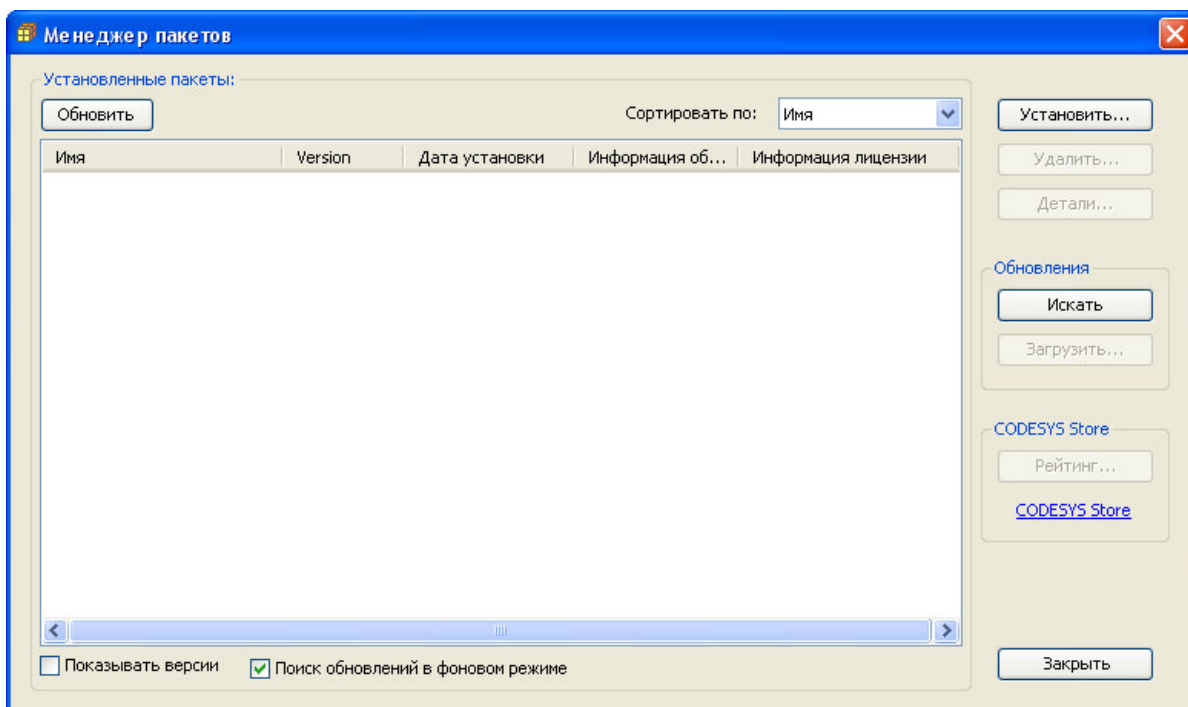


Рисунок 2.2 – Система разработки *CoDeSys*. Окно "Менеджер пакетов"

- 3) Нажать кнопку "Установить ..." и в окне выбора файла (рисунок 2.3) выбрать файл *ELSYMA TSP (<version>).package*.

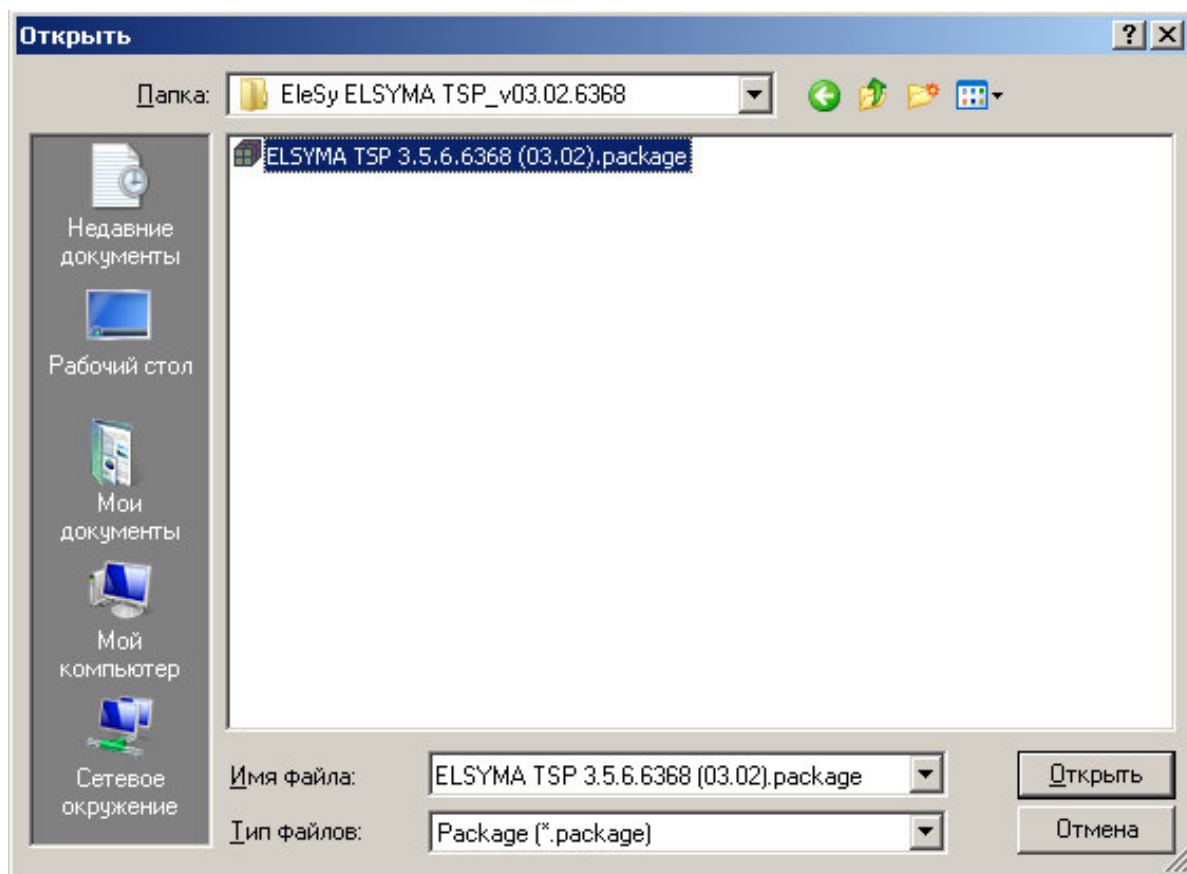


Рисунок 2.3 – Система разработки *CoDeSys*. Окно выбора файла

- 4) Далее следовать указаниям "Мастера установки".
- 5) По завершению установки следует перезапустить систему *CoDeSys* для вступления в силу всех изменений.

2.7.3 Обновление пакета поддержки контроллера (TSP)

Пакеты поддержки контроллера (*TSP*) постоянно совершенствуются и дополняются. Следовательно, *TSP* отличаются функциональными возможностями в зависимости от версии.

Для того чтобы обновить версию пакета поддержки необходимо:

1. Удалить установленную ранее версию *TSP*. Для этого необходимо

1) в системе *CoDeSys* на верхней панели выбрать меню **Инструменты**→**Менеджер проектов...**

2) В появившемся окне **Менеджер проектов** выбрать нажатием правой кнопки мыши необходимый для удаления *TSP* и нажать кнопку «Удалить...» (рисунок 2.4).

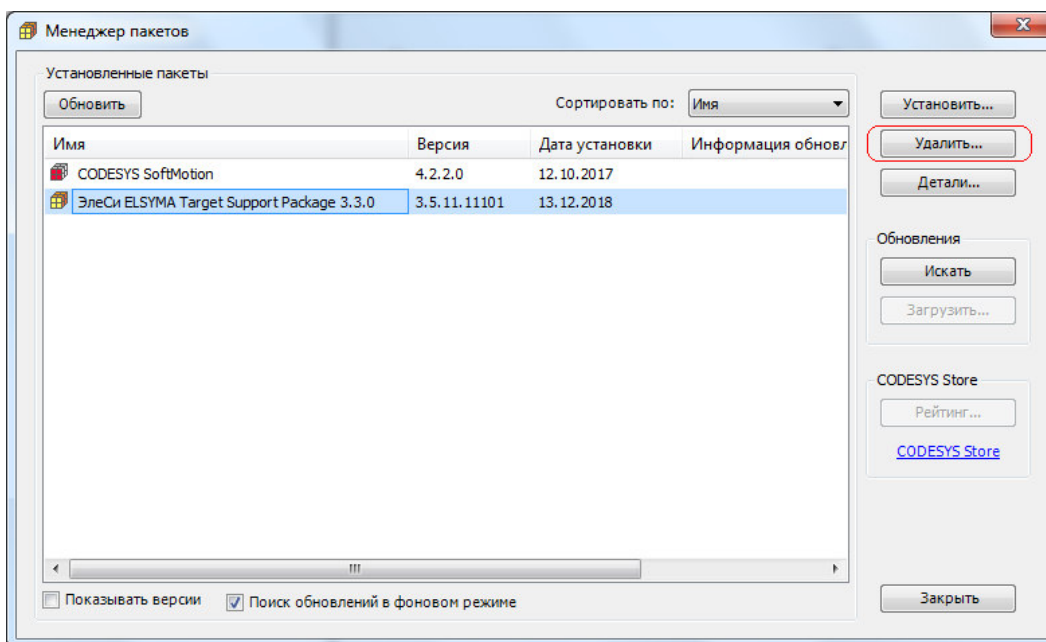


Рисунок 2.4 – Система разработки *CoDeSys*. Окно **Менеджер пакетов**.
Выбор файла для удаления

3) В появившемся окне **Удаление** нажать кнопку «Next>» (рисунок 2.5), затем следовать указаниям системы (далее рекомендуется выбрать вариант типичной установки).

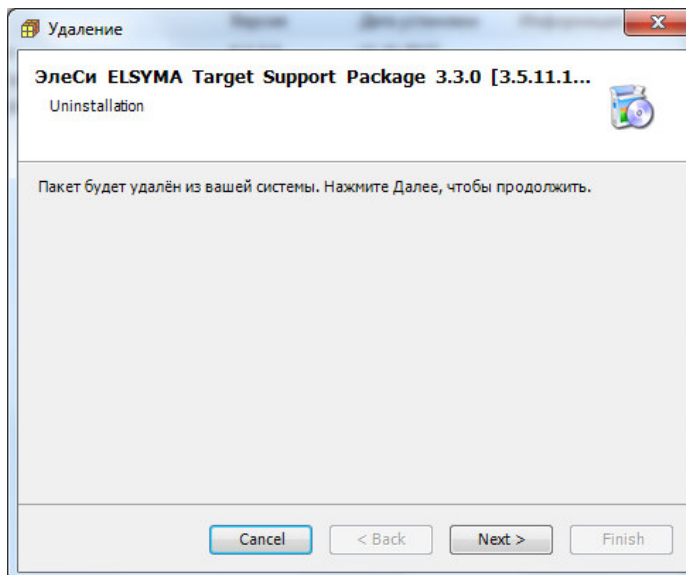


Рисунок 2.5 – Система разработки *CoDeSys*. Окно **Удаление**

- 4) Перезагрузить систему *CoDeSys* после завершения процесса удаления.
2. После перезагрузки выполнить действия 2.2) – 2.5) раздела 2.7.2.

2.7.4 Приведение ранее созданных проектов к текущей версии *TSP*

Ранее созданные проекты могут иметь отличия от текущей версии *CoDeSys* или *TSP* (т.к. могут быть добавлены новые переменные, функции и др.).

В случае открытия проекта в обновленной версии системы программирования *CoDeSys*, пользователю предлагается актуализировать окружение проекта. Для этого необходимо в появившемся окне **Среда проекта** нажать кнопку «Сделать все новейшим» (рисунок 2.6) для полной актуализации проекта (рекомендуется). Для частичной актуализации проекта (например, только для актуализации версии компилятора) необходимо в окне **Среда проекта** перейти на необходимую вкладку и в графе *Действия* выбрать «Обновить до версии X.X.X.X» (для опытных пользователей). По завершению работы с окном **Среда проекта** нажать кнопку «ОК».

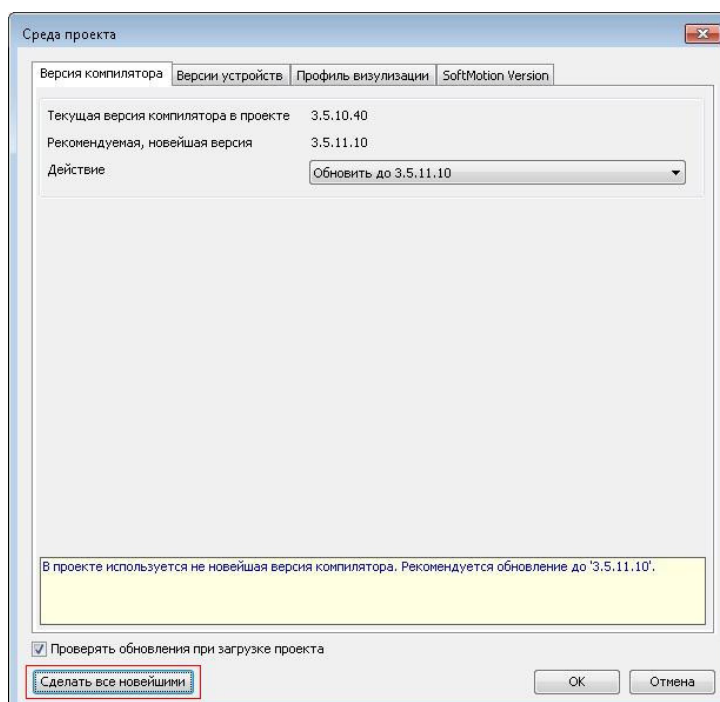


Рисунок 2.6 – Система *CoDeSys*. Окно **Среда проекта**

После актуализации *CoDeSys* отображает информационное окно, в котором указано, какие компоненты были обновлены (рисунок 2.7).

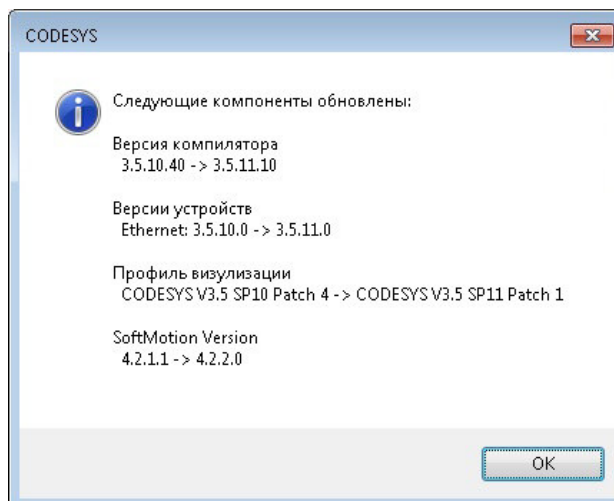


Рисунок 2.7 – Система *CoDeSys*. Информационное окно обновления компонентов

В случае открытия проекта в системе с обновленной версией *TSP*, система программирования предлагает обновить устройства в соответствии с текущей версией пакета поддержки контроллера (зеленым цветом выделены устройства, которые следует обновить). Для обновления следует поставить галочку в столбце *Обновить* напротив необходимых для обновления устройств (или выбрать все устройства, нажав на кнопку «Выбрать все») и нажать кнопку «ОК» (рисунок 2.8).

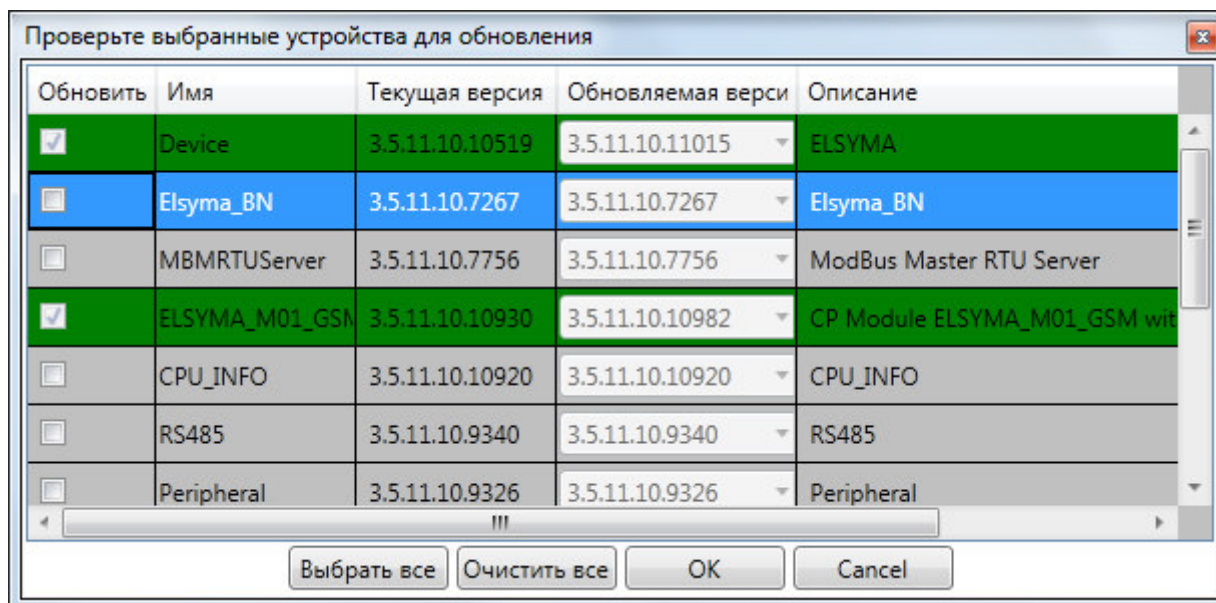


Рисунок 2.8 – Система *CoDeSys*. Окно выбора устройств для обновления

После обновления устройств *CoDeSys* отображает информационное окно (рисунок 2.9), в котором указано, какие устройства были обновлены.

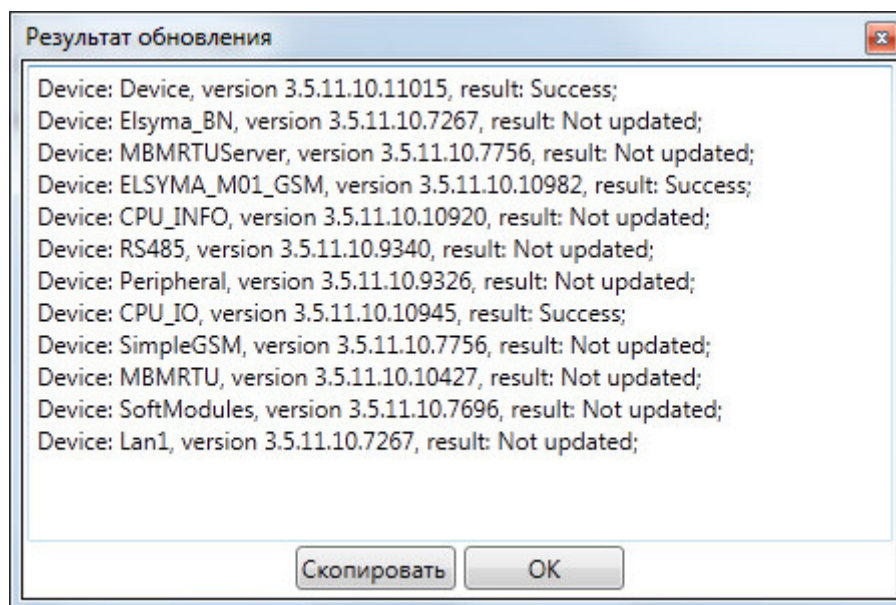


Рисунок 2.9 – Система *CoDeSys*. Информационное окно обновления устройств

Примечание – В случаях, когда *CoDeSys* не была обновлена, окно *Среда проекта* не появляется. После обновления пакета поддержки (*TSP*) в случае, если версия ни одного из устройств не была обновлена, окно обновления устройств не появляется.

2.7.5 Установка двух и более версий *CoDeSys*

При необходимости возможна установка двух и более версий *CoDeSys* на компьютер (например, версии 6 и 11).

При установке нескольких версий *CoDeSys* необходимо:

1 Для каждой версии выбрать уникальную папку расположения. В противном случае последняя установленная версия стирает предыдущую (рисунок 2.10).

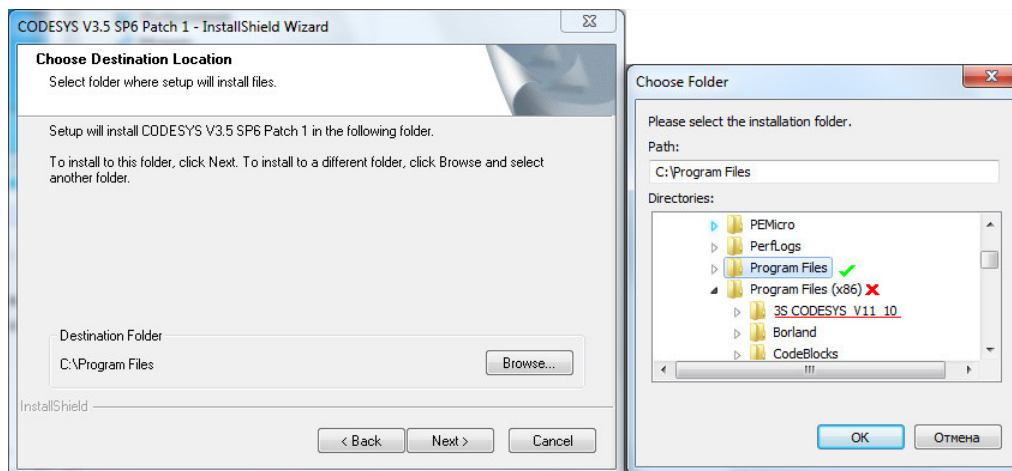


Рисунок 2.10 – Установка *CoDeSys*. Выбор папки расположения

2 При установке необходимо в списке имен папок указать версию *CoDeSys* (рисунок 2.11).

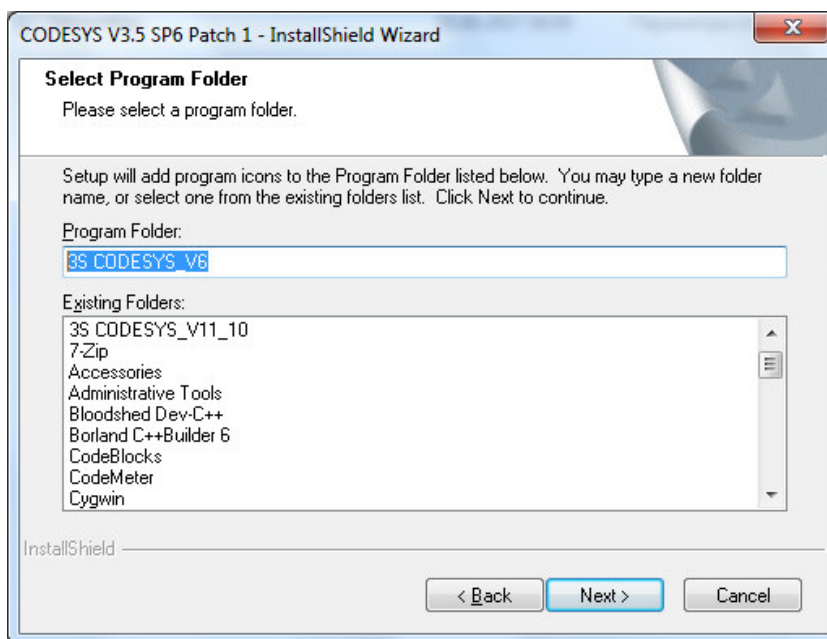


Рисунок 2.11 – Установка *CoDeSys*. Указание версии в имени папки

После установки нескольких версий *CoDeSys* в дереве конфигурации отображаются все версии устройств, но добавлять/обновлять в *CoDeSys* можно только те устройства, версии которых соответствуют запущенной версии *CoDeSys*. При попытке использования иных версий устройств система выдает сообщение о несоответствии версии (рисунок 2.12).

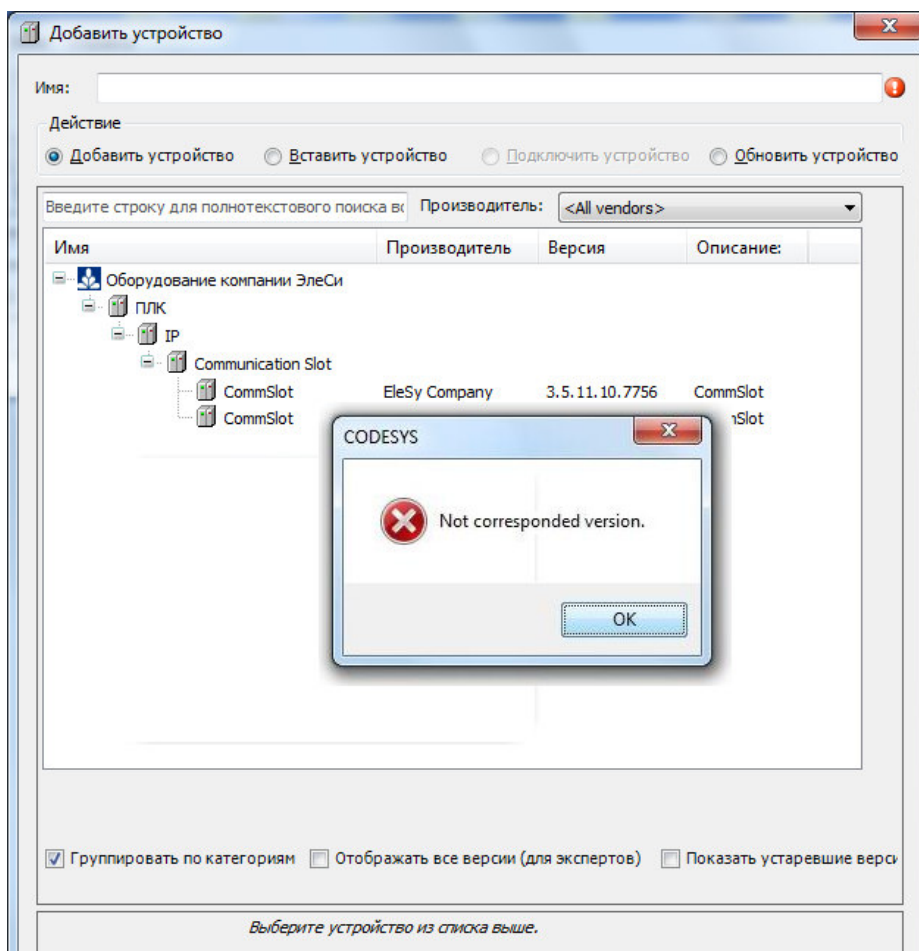


Рисунок 2.12 – Окно выбора устройств. Сообщение о выборе некорректной версии

В результате будут установлены все профили, библиотеки, компоненты и описания устройств, необходимые для обеспечения поддержки контроллера Элсима в системе *CoDeSys*.

2.8 Создание проекта

Разработка проекта контроллера Элсима включает следующие действия:

- 1 Создание конфигурации контроллера (см. 2.8.1).
- 2 Создание главной программы и других программных компонентов (см. 2.8.2).
- 3 Загрузка проекта в контроллер (см. 2.9).
- 4 Запуск задачи, мониторинг и отладка (см. 2.10.2).

2.8.1 Создание конфигурации

Порядок создания конфигурации:

- 1 Запустить систему разработки *CoDeSys* (см. 2.8.1.1).
- 2 Создать проект (см. 2.8.1.2).
- 3 Добавить устройства в конфигурацию (см. 2.8.1.3).
- 4 Редактировать конфигурацию при необходимости (см. 2.8.1.4).
- 5 Задать параметры работы отдельным модулям в дереве конфигурации (см. 2.8.1.4.1).
- 6 Назначить переменные сигналам (см. 2.8.1.4.2).

2.8.1.1 Запуск системы программирования

Запуск системы разработки *CoDeSys* следует осуществлять одним из следующих способов:



- С помощью ярлыка на рабочем столе;
- С помощью команды системного меню *Windows*:

Пуск → *Программы* → *3S CODESYS* → *CODESYS* → *CODESYS without profile*.

В появившемся окне выбора профиля (рисунок 1.5) необходимо выбрать *EleSy ELSYMA V<версия>SPx Patch<версия>* (x может отличаться от представленного далее примера).



Рисунок 2.13 – Система разработки *CoDeSys*. Выбор профиля

ВАЖНО! Во время первого запуска системы разработки *CoDeSys* появляется окно, в котором предлагается *выбрать параметры среды по умолчанию (Choose your default environment settings)*. Необходимо выбрать **профессиональный** профиль (*Professional*) (рисунок 2.14), так как он является наиболее универсальным и обладает возможностями, которые недоступны стандартному профилю. После выбора нажать кнопку «*Start*».

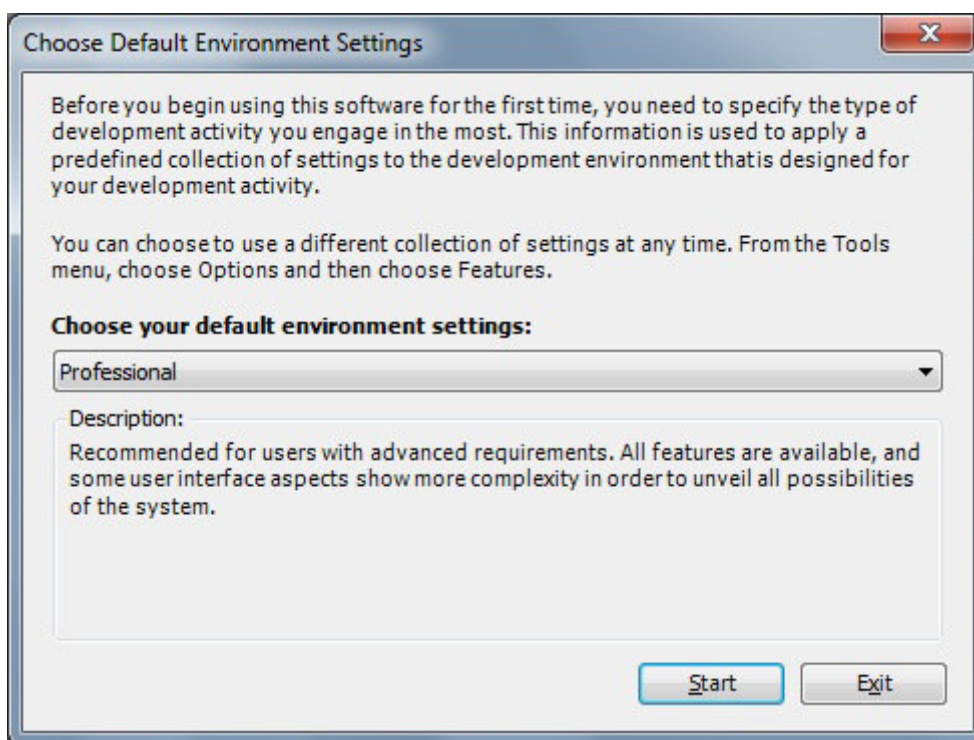


Рисунок 2.14 – Выбор параметров среды по умолчанию

2.8.1.2 Создание проекта

Конфигурация контроллера, программные компоненты (POUs), составляющие код управляющей программы, и другие объекты содержатся в проекте. Для создания проекта следует:

1 В меню **Файл** выбрать команду **Новый проект...** или нажать клавиши *[Ctrl]+[N]* (рисунок 2.1).

2 В окне "**Новый проект**" (рисунок 2.15) в списке **Шаблоны:** выбрать шаблон **Стандартный проект**.

Выбранным шаблоном проекта определяются базовые настройки проекта (структура меню, predeterminedные объекты и др.).

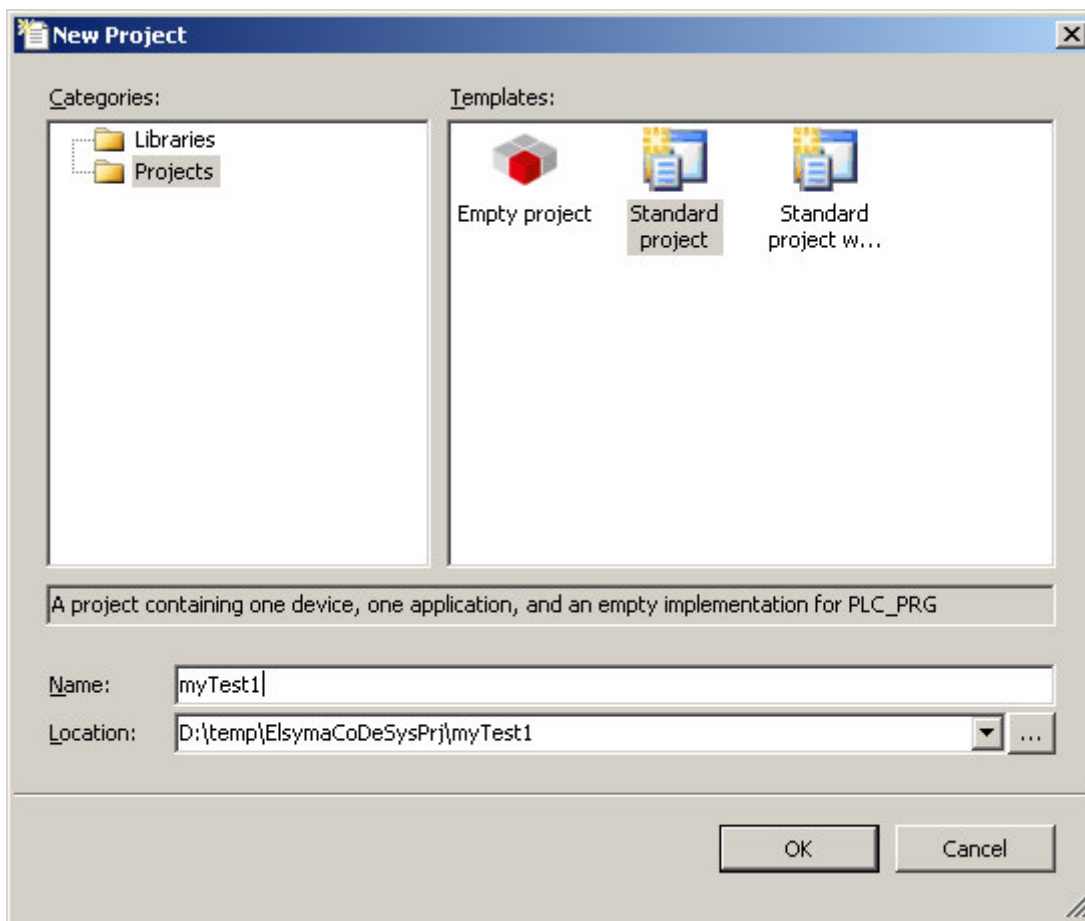


Рисунок 2.15 – Система разработки CoDeSys. Выбор шаблона проекта

3 В поле **Имя:** задать имя проекта, а в поле **Расположение:** указать место для сохранения файлов проекта.

4 Нажать кнопку "**ОК**". Проект сохраняется в указанном месте в файле `<project_name>.project`.

5 В окне "**Стандартный проект**" в списке **Устройство:** выбрать контроллер *ELSYMA (EleSy Company)*, в списке **PLC_PRG на:** – язык реализации основного программного компонента *Структурированный текст (ST)* (рисунок 2.16).

6 Нажать кнопку "**ОК**".

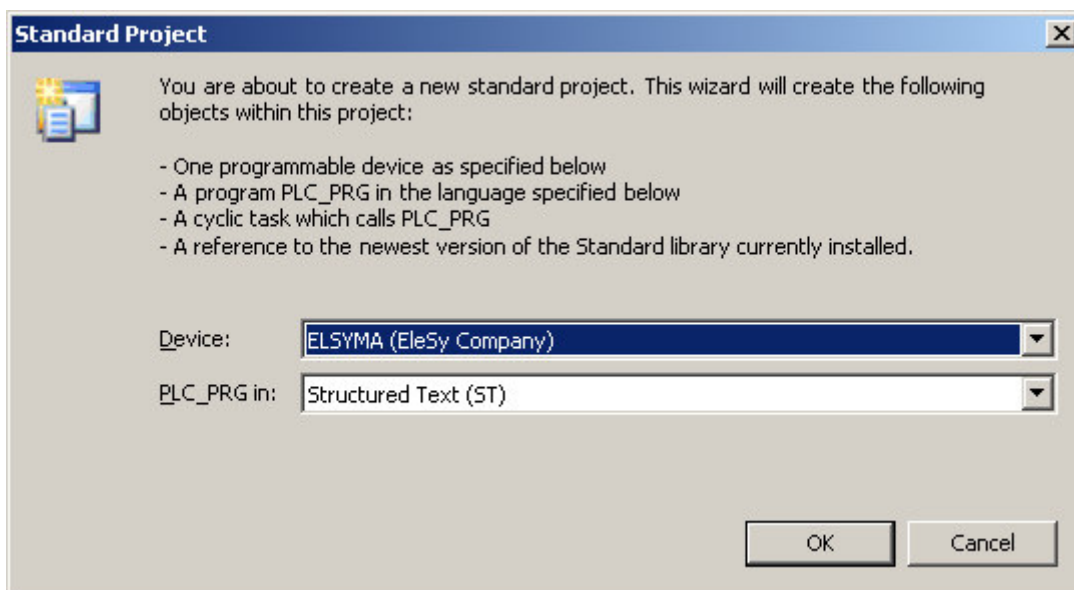


Рисунок 2.16 – Система разработки *CoDeSys*. Настройка стандартного шаблона проекта

Созданный проект отображается в области *Устройства* в виде дерева объектов (рисунок 2.17).

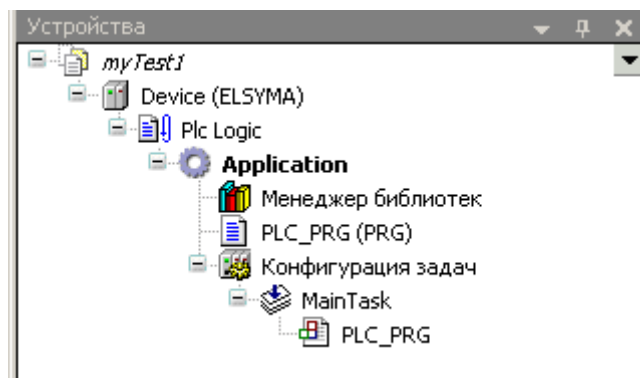


Рисунок 2.17 – Система разработки *CoDeSys*. Дерево объектов проекта

Проект включает одно или несколько устройств – элементов первого уровня (узел *Device (ELSYMA)* на рисунке 2.17). Каждое устройство включает два основных объекта: *Plc Logic* (контейнер для программных компонентов) и аппаратную конфигурацию.

Контейнер *Plc Logic*, в свою очередь, содержит:

- **Application** – представляет собой набор объектов для запуска экземпляра программы в конкретном аппаратном устройстве и включает в себя следующие компоненты:

- **Менеджер библиотек** – компонент, предоставляющий доступ к библиотекам *CoDeSys*, которые представляют собой специализированный набор функций и функциональных блоков (ФБ), таких как, например, ФБ **M23X**, **CE30X**;

- программные компоненты:

- **PLC_PRG (PRG)** – основная программа, самый верхний уровень проекта. При запуске проекта в контроллере программа *PLC_PRG* первая получает управление;

- любое количество других программных компонентов (POU);

- **Конфигурация задач** – конфигуратор для управления задачами с главной задачей *Main Task*;

- другие компоненты, поддерживаемые системой *CoDeSys* (список глобальных переменных, Interface и др.).

2.8.1.3 Добавление устройств в конфигурацию

Графически конфигурация представлена как дерево устройств. Основным узлом (самый верхний уровень) является проект контроллера Элсима. Именно основной узел определяет, какие устройства могут быть добавлены и в каком порядке.

В состав дерева устройств проекта входят следующие узлы (рисунок 2.18):

- Базовый узел подключения **Elsyma_BN**. Добавляется в первую очередь к узлу **Device (ELSYMA)** и служит для логической организации взаимодействия контроллера Элсима с модулями УВВ;

- Контроллер – добавляется к базовому узлу (через устройство **Module_CP**), при добавлении указывается исполнение контроллера, и включает в себя следующие узлы:

- настройка и работа с контроллера (**CPU_INFO**);
- программные модули (**SoftModules**);
- настройка и работа с интерфейсом *Ethernet* (**LAN**);
- настройка и работа с интерфейсом *RS-485* (**RS485**);
- настройка и работа с периферийными устройствами (**Peripheral**);
- настройка и работа с вводом-выводом контроллера (**CPU_IO**);
- настройка и работа с коммуникационным интерфейсом GSM (**SimpleGSM**);

- Модули УВВ (**ExtModules**).

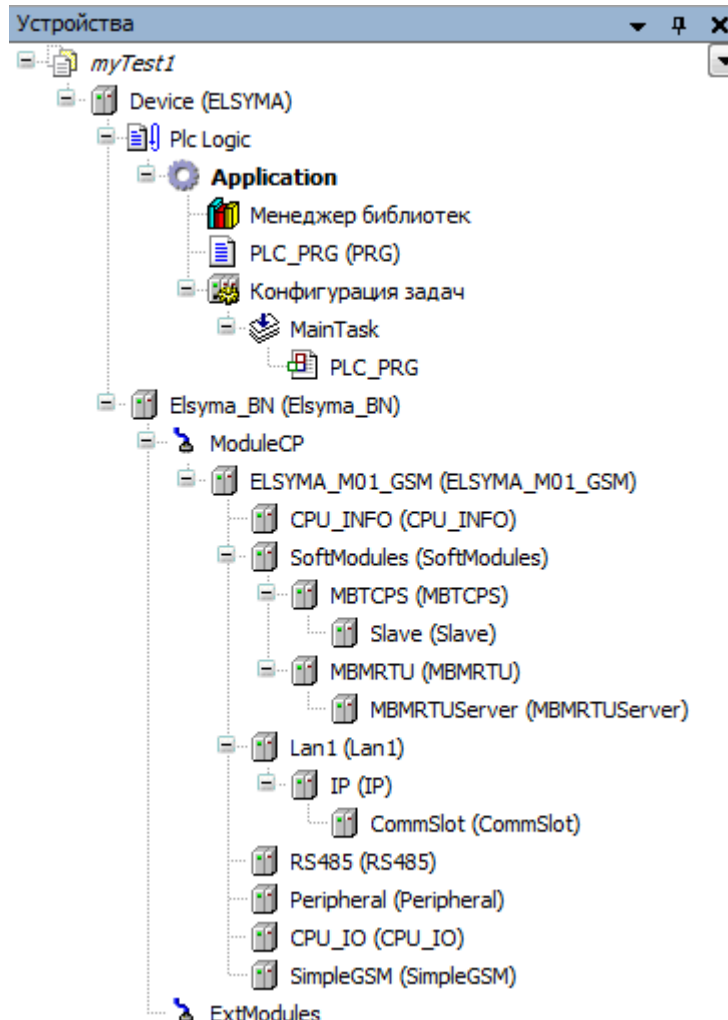


Рисунок 2.18 – Система разработки CoDeSys. Структура дерева устройств

Список поддерживаемых аппаратных и программных модулей и узел для добавления представлен в таблице 2.2. Каждый модуль в сервисной программе идентифицируется коротким символьным обозначением, эти обозначения приведены в столбце **Обозначение**.

Таблица 2.2 – Список устройств, доступных для добавления узла **ModuleCP**

Узел	Обозначение	Назначение	Номер раздела
ModuleCP	ELSYMA_M01	Программный модуль, обеспечивающий функциональность исполнения контроллера Элсима-M01-ZZZU	3
	ELSYMA_M01_GSM	Программный модуль, обеспечивающий функциональность исполнения контроллера Элсима-M01-ZZZU-GSM	3
SoftModules	MBTCPM	Программный модуль организации доступа к сигналам контроллера по протоколу <i>Modbus TCP</i> с функциональностью сервера с поддержкой 16-ти соединений	3.3
	MBTCPS	Программный модуль организации доступа к сигналам контроллера по протоколу <i>Modbus TCP</i> в режиме Slave , обеспечивающий подключение до четырех опрашиваемых устройств	3.4
	MBMRTU	Программный модуль организации доступа к сигналам контроллера по протоколу <i>Modbus RTU</i> с функциональностью Master -устройства (обеспечивает опрос до 16 подчиненных устройств по одному каналу связи)	3.5
	MBRTUS	Программный модуль организации доступа к сигналам контроллера по протоколу <i>Modbus RTU</i> в режиме Slave (Server)	3.6
	IEC104M	Программный модуль организации доступа к сигналам контроллера по протоколу <i>МЭК</i> в режиме потребителя данных (MasterM)	См. раздел *
	IEC104S	Программный модуль организации доступа к сигналам контроллера по протоколу <i>МЭК</i> в режиме поставщика данных (Slave)	См. раздел *
	ELMicronMst	Программный модуль, предназначенный для обеспечения опроса до 16-ти счетчиков электроэнергии СЭТ4ТМ03М и ПСЧ-4ТМ.05МК через интерфейс <i>RS-485</i>	3.7
	tsync	Программный модуль, предназначенный для работы в составе программного обеспечения контроллера Элсима-M01 и обеспечивающий синхронизацию временем с модулями УВВ	3.10

* См. документ "Контроллер программируемый логический Элсима. Поддержка протоколов передачи данных ГОСТ Р МЭК 60870-5-104-2004. Руководство по применению"

Для добавления устройств в конфигурацию следует:

1 Выделить нужный узел в дереве устройств и выбрать команду **Добавить устройство...** контекстного меню (рисунок 2.19).

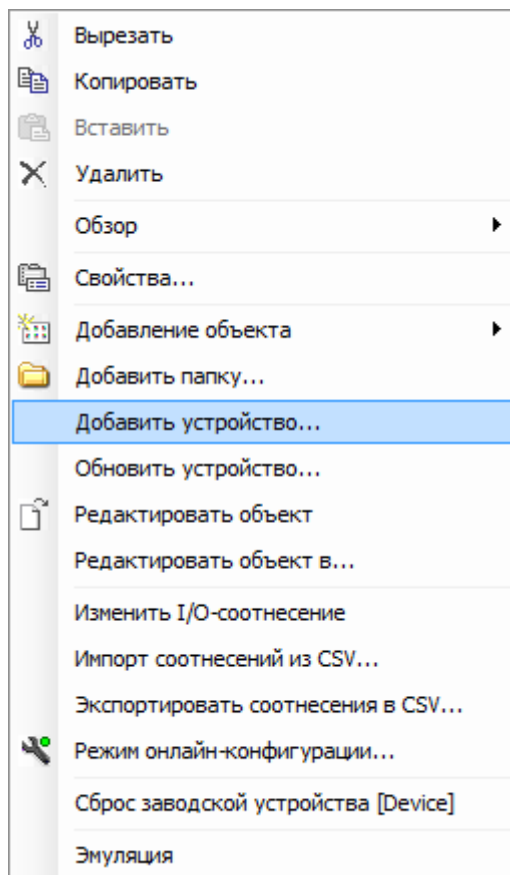



Рисунок 2.19 – Система разработки *CoDeSys*. Вид контекстного меню элементов дерева устройств

2 В окне "Добавить устройство" в группе *Устройство*: отображается список доступных для добавления к данному узлу устройств. Устройства в списке сгруппированы по функциональному назначению. В поле *Производитель*: следует выбрать *EleSy Company* – устройства компании "ЭлеСи" и нажать кнопку "Добавить устройство" (рисунок 2.20).

После этого устройство появится в дереве устройств, а диалоговое окно "Добавить устройство" остается открытым (т.к. не является модальным).

3 Для добавления остальных устройств повторить пункты 1–2.

Примечание – Если обязательное поле не было заполнено, рядом с этим полем появляется значок .

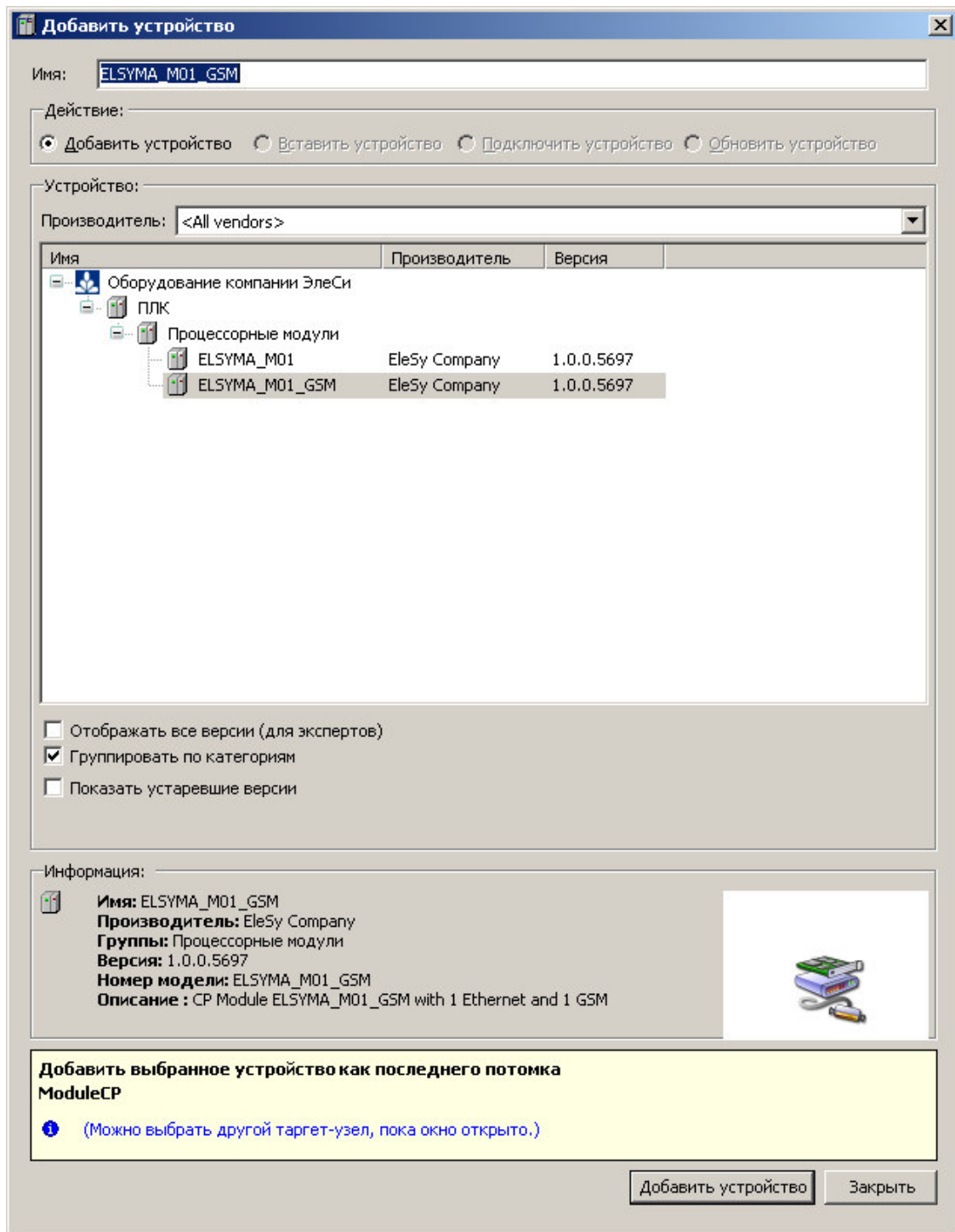


Рисунок 2.20 – Система разработки *CoDeSys*. Окно добавления устройств

2.8.1.4 Просмотр и редактирование данных модуля

Каждый функциональный модуль работает с данными нескольких категорий:

- конфигурационные параметры;
- данные модуля. По функциональному назначению данные подразделяются на три типа:
 - входные/выходные данные (измерения, сигналы управления, данные обмена по протоколам и др.);
 - диагностические данные: статус работы модуля, наличие связи с ЦП и др.;
 - статистические данные: версии ПО модулей и компонентов, количество пересбросов, ошибок передачи и т.п.

Доступ к данным модуля осуществляется в области просмотра и конфигурирования устройства.

Для просмотра данных модуля следует выделить имя модуля в дереве устройств и дважды нажать левую кнопку "мыши", при этом в области просмотра и конфигурирования появляется закладка с именем устройства.

Закладка данных модуля включает несколько встроенных закладок:

- **Редактор параметров** (см. 2.8.1.4.1);
- **Соотнесение входов/выходов** (см. 2.8.1.4.2);
- **Состояние** (см. 2.8.1.4.3);
- **Информация** (см. 2.8.1.4.4).

2.8.1.4.1 Закладка *Редактор параметров*

На закладке **Редактор параметров** (рисунок 2.21), в зависимости от реализации конкретного модуля, отображаются следующие параметры:

- **Информация модуля;**
- **Системные параметры модуля;**
- **Конфигурационные параметры модуля.**

Параметры определяют индивидуальные настройки модуля и задаются на начальном этапе конфигурирования. Параметры каждого модуля описываются в подразделах "Настройка параметров модуля".

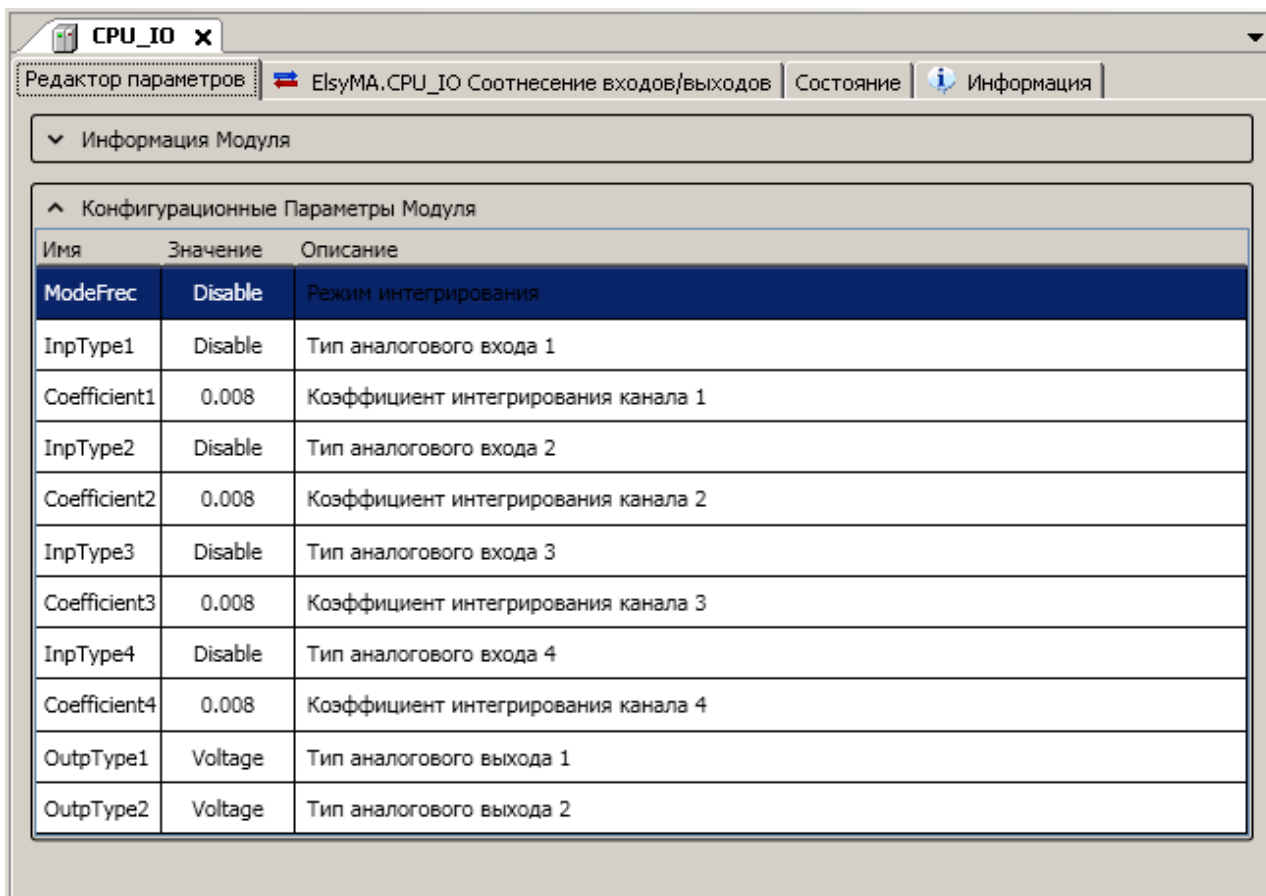


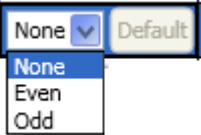



Рисунок 2.21 – Система разработки *CoDeSys*. Пример закладки *Редактор параметров*

Параметр редактируется следующим образом:

- 1 Выбрать параметр для редактирования.
- 2 Установить курсор "мыши" в область ячейки значения параметра.
- 3 Дважды нажать левую кнопку "мыши".
- 4 Ввести или выбрать необходимое значение параметра.

Редактирование параметров выполняется в ячейках столбца *Значение* с помощью одного из следующих элементов управления:

-  – счетчика значений;
-  – поля ввода значений;
-  – списка выбора значения.

Кнопка  – устанавливает значение "по умолчанию". Данная кнопка находится в активном состоянии, если значение в ячейке задано пользователем.

2.8.1.4.2 Закладка *Соотнесение входов/выходов*

На закладке *Соотнесение входов/выходов* отображаются входные/выходные и диагностические данные – см. рисунок 2.22.

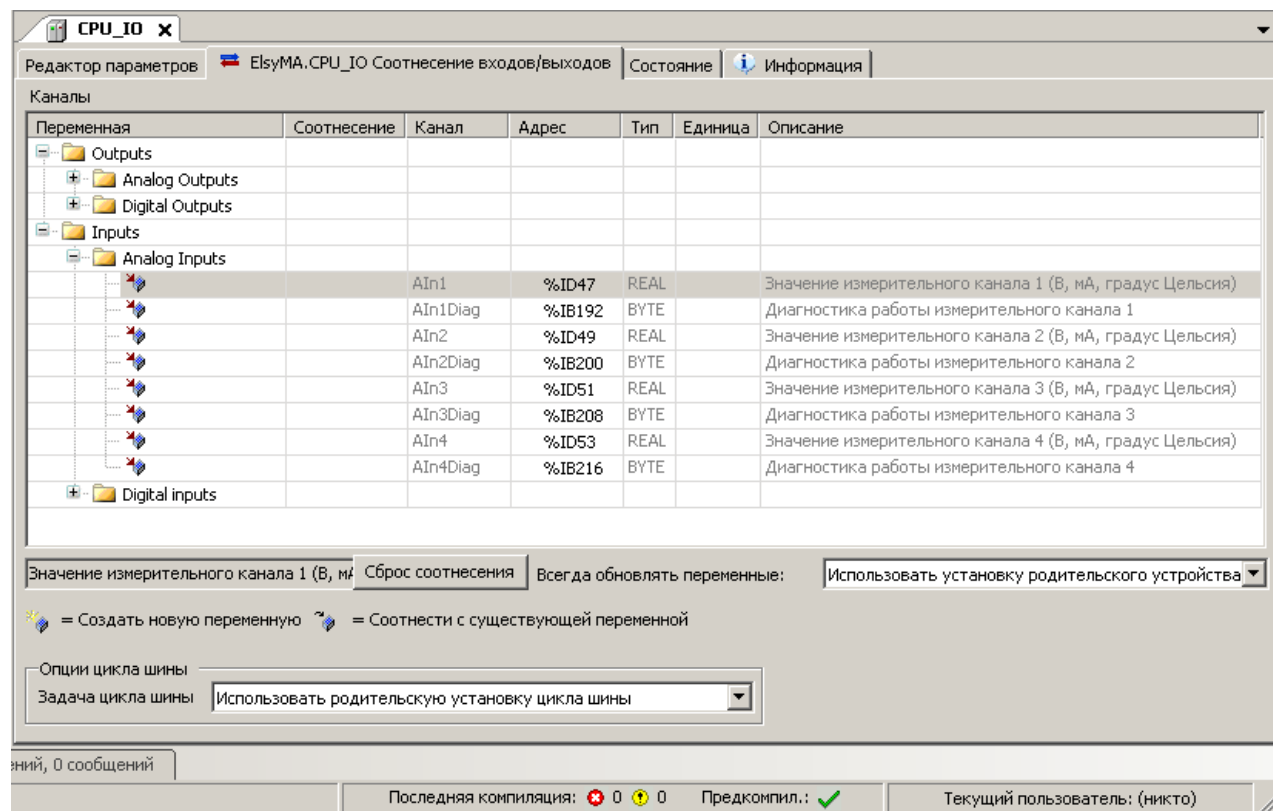


Рисунок 2.22 – Система разработки CoDeSys. Представление входных/выходных и диагностических данных

Данные представлены в форме таблицы сигналов *Каналы*. Сигнал – это элемент данных с набором атрибутов и с определенным адресом в памяти контроллера. Строками таблицы *Каналы* являются сигналы, столбцами – атрибуты сигналов.

Каждый сигнал модуля ввода-вывода может быть представлен в виде структуры или элементарного типа данных. Доступные типы структур данных для описания сигналов представлены в таблице 3.9.

Сигналы в таблице по назначению объединены в следующие группы:

- **Входные сигналы (Input Signals);**
- **Выходные сигналы (Output Signals);**
- **Диагностические сигналы (Diagnostics):**
 - **Системные сигналы (System)** – в зависимости от реализации модуль имеет **стандартный набор системных выходных сигналов.**

Состав групп и сигналов в группах индивидуален для каждого модуля и описывается в разделах на каждый модуль (см. "Настройка параметров модулей").

В столбцах представлены следующие атрибуты сигналов:

- **Переменная:**
 - для группы – имя группы;
 - для сигнала – имя назначенной сигналу переменной и обозначение области хранения переменной в соответствии с таблицей 2.3;

Таблица 2.3 – Области хранения переменных



Обозначение	Индекс	Описание
	<i>I</i>	Область входов
	<i>Q</i>	Область выходов
	<i>M</i>	Прямонадресуемая память

- **Соотнесение** – тип маппинга;
- **Канал** – имя сигнала;
- **Адрес** – адрес сигнала в памяти контроллера;
- **Тип** – тип данных значения сигнала в соответствии с таблицей В.1;
- **Текущее значение** – текущее значение сигнала (отображается в режиме **online**);
- **Единица** – единицы измерения значения сигнала;
- **Описание** – описание сигнала.

При выборе имени сигнала в столбце **Переменная**, описание сигнала отображается в поле, находящемся под таблицей сигналов.

Из управляющей программы доступ к сигналам осуществляется через переменные, назначенные сигналам в конфигурации. Такие сигналы называются смappированными, а сам процесс – **маппингом (mapping)**.

Предусмотрено два типа маппинга:

- на новую переменную –  = Создать новую переменную. Такая переменная будет автоматически объявлена во внутреннем списке глобальных переменных *CoDeSys* (с указанным именем указанного типа);
- на существующую переменную –  = Соотнести с существующей переменной. Такая переменная должна быть объявлена пользователем.

Существует возможность выполнения маппинга на следующих закладках:

1 **Карта сигналов** – данная закладка используется для создания каналов или групп сигналов и маппирования созданных сигналов в области **Привязка и автонаименование**. Процесс выполнения маппирования сигналов к переменным на закладке **Карта сигналов** описан в подразделах на отдельные модули (см., для примера, 3.4.2.1);

2 **Соотнесение входов/выходов** – данная закладка используется для маппирования ранее созданных сигналов или заданных в конфигурации диагностических/статистических параметров.

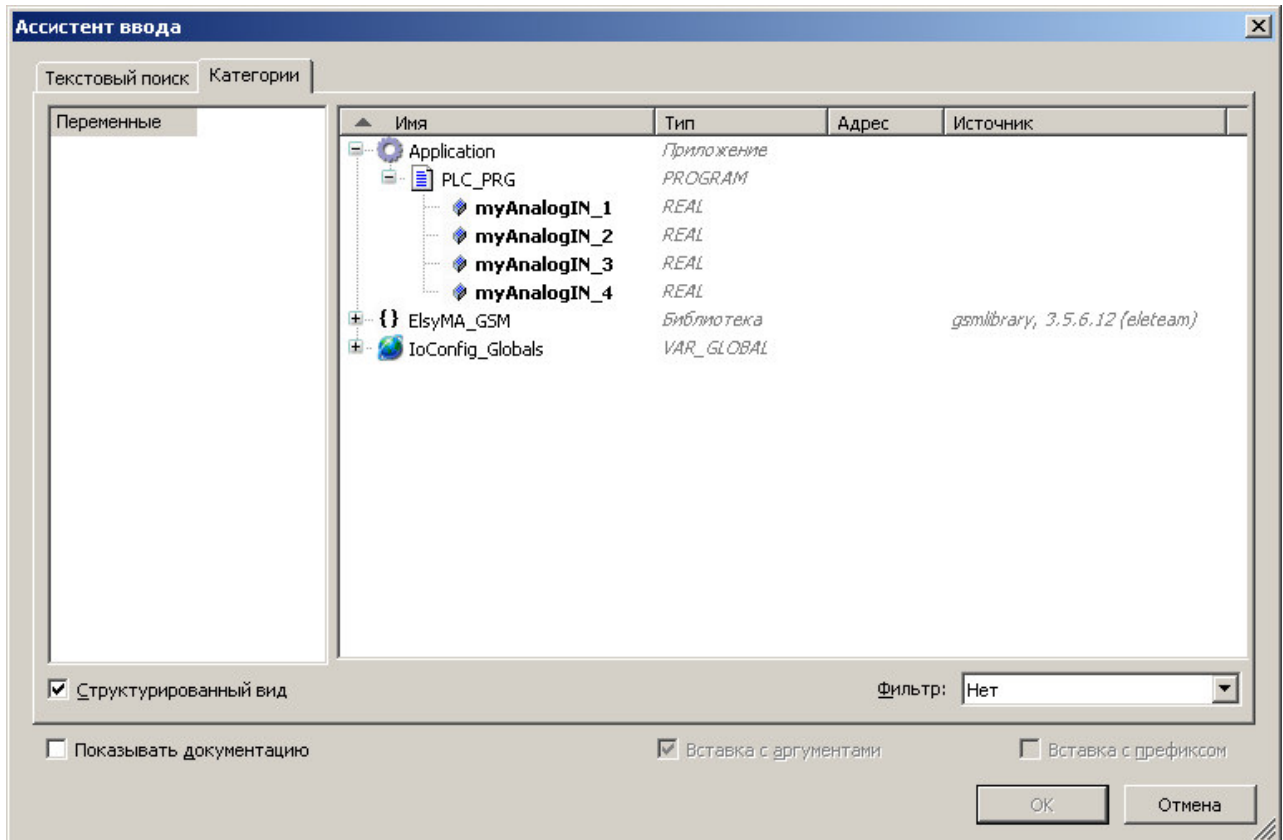


Рисунок 2.23 – Система разработки CoDeSys. Окно "Ассистент ввода"

Для того чтобы смэппировать переменную, необходимо на закладке **Соотнесение входов/выходов**:

1 В столбце **Переменная** выделить сигнал и дважды нажать левую кнопку "мыши".

2 Ввести имя для назначаемой переменной:

1) для маппинга на новую переменную ввести ее имя в поле ввода.

2) для маппинга на существующую переменную:

в ячейке нажать кнопку .

в окне "Ассистент ввода" (рисунок 2.23) на закладке **Категории**: в списке **Переменные** отображается список объектов, доступных в проекте. Следует выбрать нужную категорию из списка.

3) в области отображения переменных выбрать нужный элемент и имя переменной и нажать кнопку "ОК".

На рисунке 2.24 представлен пример отображения маппированных переменных.

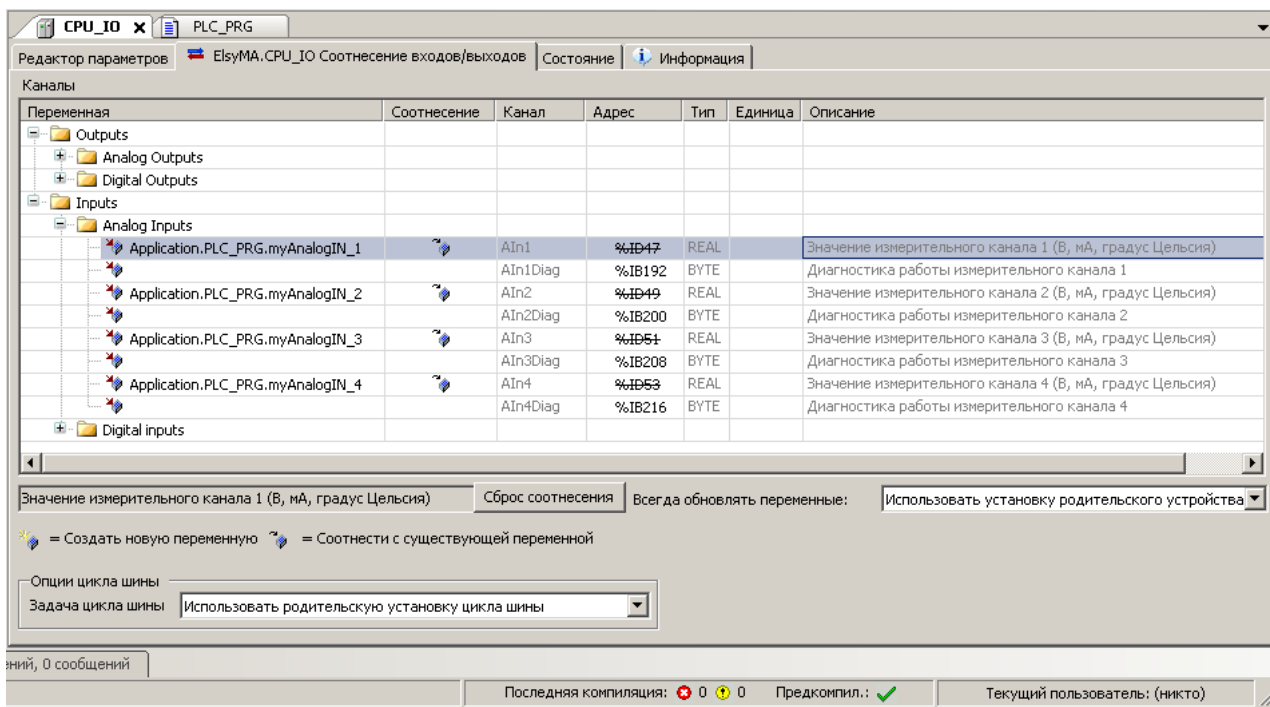


Рисунок 2.24 – Система разработки CoDeSys. Маппинг переменной

В группе *Опции цикла шины* в поле *Задача цикла шины* существует возможность задать цикл шины:

- *Main Task* – "главная" задача *PLC_PRG*, созданная в проекте "по умолчанию" и выполняемая циклически;
- *Использовать родительскую установку цикла шины* – используются настройки цикла шины вышестоящего (родительского) сигнала.

Перед запуском для корректной работы программы в контроллере необходимо задать параметры настройки обновления значений переменных, выбрав одно из следующих элементов выпадающего списка *Всегда обновлять переменные*:

- *Использовать установку родительского устройства* – использовать настройки старшего (родительского, по уровню вложенности) устройства;
- *Вкл. 1 (в задаче цикла шины, если не используется)* – использовать задачу цикла шины, если ни одна пользовательская задача не задана;
- *Включено 2 (всегда в задаче цикла шины)* – всегда обновлять переменные при выполнении задачи цикла шины.

ВНИМАНИЕ! В текущей версии сервисной программы не реализована возможность конфигурирования параметров настройки для обновления значений переменных и цикла шины. Описание данных параметров носит информационный характер и пользователю не рекомендуется задавать данные параметры.

ВНИМАНИЕ! Если сигналы были смаппированы, но не используются в управляющей программе, в *online*-режиме их значения обновляться не будут. В процессе отладки можно установить параметр *Всегда обновлять переменные* в значение "*Включено 2*" для обновления сигналов в *online*-режиме. В штатном режиме работы контроллера не рекомендуется использовать данный режим.

В случае необходимости удаления соотнесения переменных нажмите кнопку "**Сброс соотнесения**".

2.8.1.4.3 Закладка *Состояние*

На закладке *Состояние* (рисунок 2.25) отображается статус подключенных модулей УВВ. Существуют следующие состояния модулей:

- *n/a* (not available) – модуль не доступен;
- *запуск* – модуль в работе.

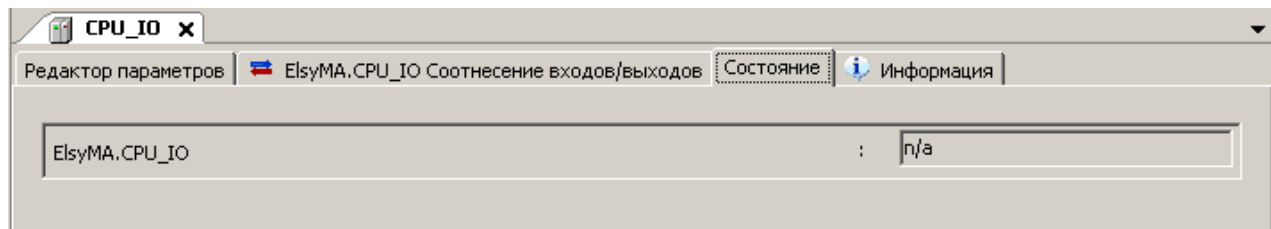


Рисунок 2.25 – Система разработки *CoDeSys*. Закладка *Состояние*

2.8.1.4.4 Закладка *Информация*

На закладке *Информация* (рисунок 2.26) в группе *Общее*: отображается информация об устройстве: символическое обозначение, производитель, тип, версия модуля, номер модели для заказа, описание назначения модуля.

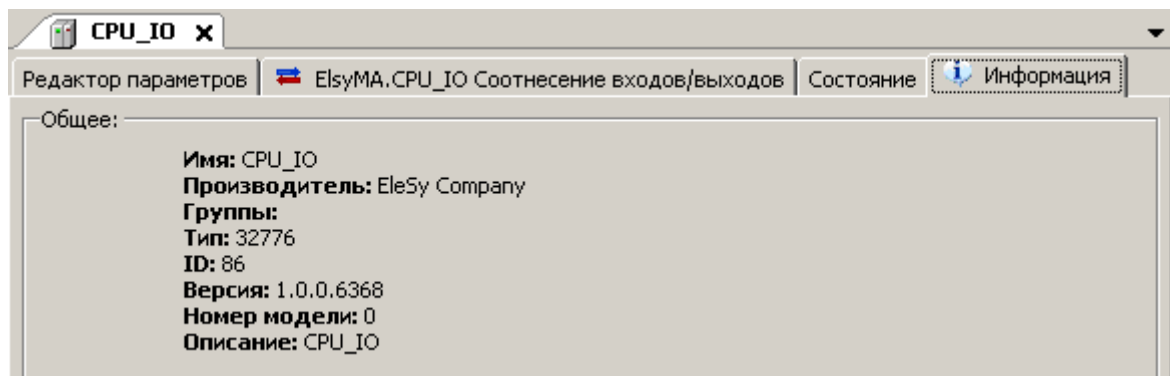


Рисунок 2.26 – Система разработки *CoDeSys*. Закладка *Информация*

2.8.2 Написание кода управляющей программы

Управляющая программа представляет собой набор программных объектов, таких как, компонент доступа к библиотекам, программные компоненты (**POUs**), **GVL** и др., и предназначена для запуска в конкретном устройстве (контроллере). Помимо стандартных типов данных, в состав пакета поддержки контроллера включены дополнительные типы данных, используемых для работы с контроллером (см. приложение В).

Последовательный процесс создания программных компонентов включает в себя добавление и объявление программного компонента (**POU**):

- объявление переменных и маппинг;
- написание тела компонента.

Правила создания **POU** представлены в стандарте IEC 61131-3, порядок выполнения действий – в документации на систему программирования *CoDeSys*.

При использовании ЭНП (переменные типа **RETAIN**) для контроллера накладывается ограничение – **максимальное количество задаваемых в ЭНП переменных не должно превышать 27 Кбайт**.

2.9 Настройка соединения с контроллером

Для работы с контроллером имеется возможность применения двух интерфейсов связи:

- *Ethernet*;
- *USB*.

2.9.1 Настройка соединения через *Ethernet*

Для работы через *Ethernet* необходимо подключить контроллер к сети через разъем **LAN1** («□□1»). Для работы с сетью *Ethernet* контроллеру задается сервисный IP-адрес (адрес, по которому контроллер будет доступен для системы *CoDeSys*). Сервисный IP-адрес может задаваться в двух режимах: *static* и *DHCP* (Dynamic Host Configuration Protocol – протокол динамической настройки узла).

В режиме *static* IP-адрес задается статически. В этом случае необходимо предварительно задать корректные сетевые настройки контроллера в соответствии с заданными настройками сети. Процесс изменения сетевых параметров контроллера приведен в приложении Б.

Начиная с версии системы 3.3.0, по умолчанию включен режим установки IP-адреса по *DHCP*. В режиме *DHCP* сервисный IP-адрес и необходимые настройки сети назначаются сервером DHCP (в этом случае в системе должен присутствовать сервер DHCP). Типовой пример подключения контроллера через сетевой коммутатор приведен на рисунке 2.27 (в данном примере сервером DHCP является сетевой коммутатор).

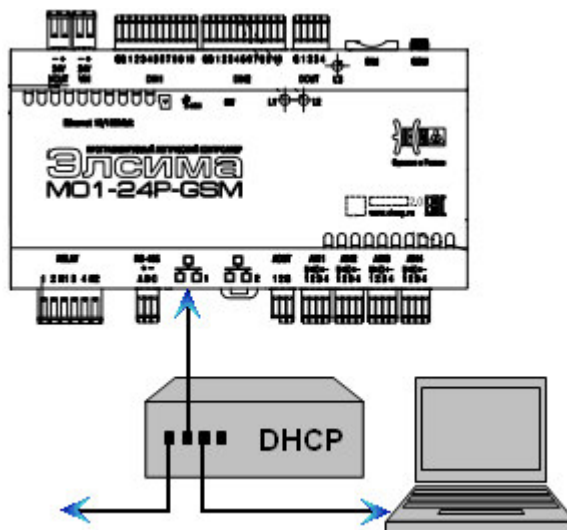


Рисунок 2.27 – Схема установки IP-адреса через коммутатор по *DHCP*

Примечания:

1. Не рекомендуется использовать режим *DHCP* в реальном проекте.
2. Не рекомендуется задавать IP-адрес рабочей системы и сервисный IP-адрес по *DHCP* в одной подсети.

Для отключения режима *DHCP* необходимо в системе *CoDeSys* установить значение параметра **LanModSet** равным *static* (см. раздел 3.2).

2.9.2 Настройка соединения через USB

Для работы через *USB* необходимо предварительно установить драйвер поддержки для работы через *USB* и подключить контроллер к ПК через разъем USB2 («*mini*»). Процесс установки драйвера приведен в приложении Г.

2.9.3 Настройка шлюза связи

Взаимодействие среды разработки с системой исполнения контроллера осуществляется через специальное приложение – шлюз связи (*Gateway*).

Система шлюзов позволяет создавать сложные разветвленные системы с использованием различных протоколов связи.

Для настройки соединения с контроллером и загрузки проекта следует задать шлюз связи по необходимости. По умолчанию шлюз связи настроен на локальный компьютер.

В случае необходимости изменения шлюза связи следует:

1 Открыть закладку данных контроллера в области отображения и конфигурирования настроек контроллера, дважды нажав левую кнопку "мыши" на имени контроллера (*Device*) в дереве устройств.

2 Выбрать закладку *Установки соединения* (рисунок 2.28) и выбрать в меню *Gateway* команду *Add new gateway...*

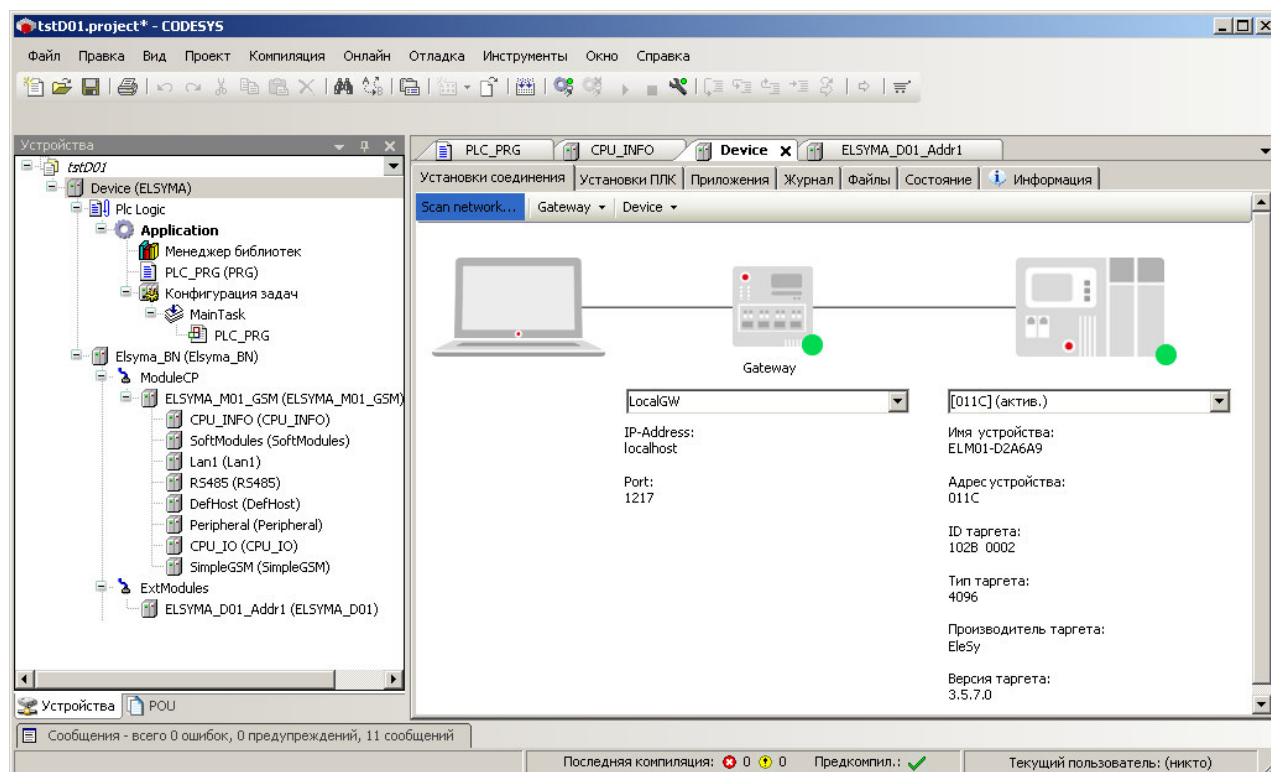


Рисунок 2.28 – Система разработки *CoDeSys*. Добавление шлюза связи

3 В окне "*Gateway*" в поле *Имя*: задать имя шлюза (для примера задано имя *ElsymaGateway*); в поле *Драйвер*: выбрать *TCP/IP* (рисунок 2.29), задать IP-адрес и порт шлюза, нажать кнопку "ОК".

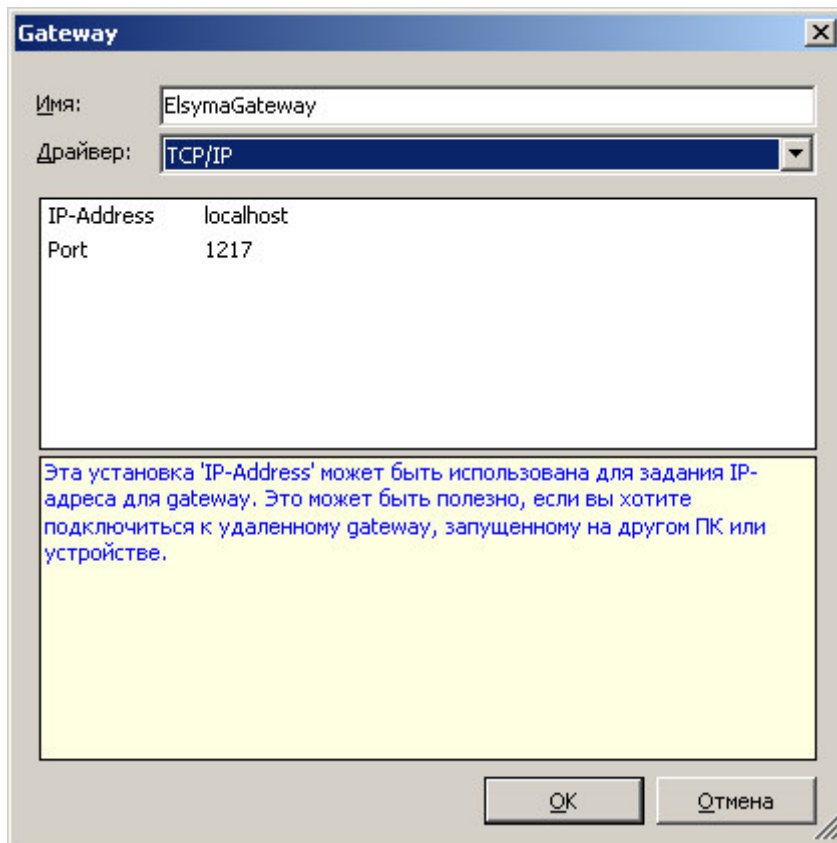


Рисунок 2.29 – Система разработки *CoDeSys*. Добавление шлюза связи

2.9.4 Проверка настройки соединения

Для проверки правильности подключения и настройки параметров необходимо выполнить поиск доступных контроллеров в сети (в общем случае, в сети может быть доступно несколько контроллеров) и выбрать из них необходимый (см. раздел 2.10.2).

2.10 Загрузка проекта в контроллер и отладка

Для загрузки проекта в контроллер и отладки следует:

- 1 Скомпилировать проект (см. 2.10.1).
- 2 Выбрать активный контроллер (контроллер, в который будет загружаться проект) (см. 2.10.2).
- 3 Подключиться к контроллеру. Загрузка проекта производится автоматически при подключении к контроллеру.
- 4 Запустить программу и отладить возможные ошибки.

2.10.1 Компиляция проекта

Для того чтобы скомпилировать проект, следует выбрать команду меню **Компиляция** или нажать кнопку **[F11]**.

Сообщения об ошибках выводятся в область отображения сообщений "**Сообщения**" (рисунок 2.30).

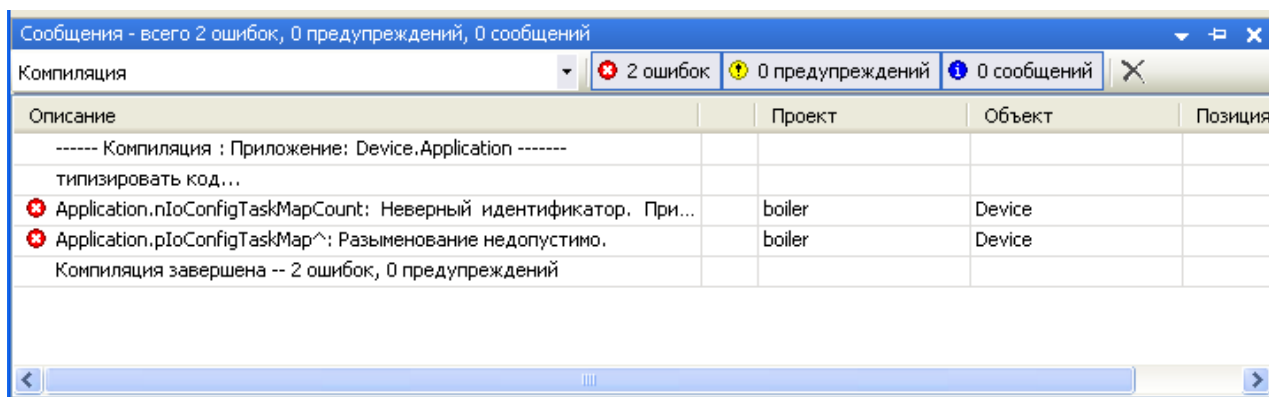


Рисунок 2.30 – Система разработки CoDeSys. Окно сообщений

2.10.2 Выбор активного контроллера

Для работы с контроллером необходимо выбрать из доступных в сети контроллеров необходимый и назначить его активным. Все последующие операции будут выполняться именно с этим контроллером. Чтобы назначить контроллер активным, необходимо:

1 На закладке **Установки соединения** (рисунок 2.28) нажать кнопку "**Scan network...**" для получения списка доступных устройств.

2 В появившемся окне "**Выбор устройства**" (рисунок 2.31) в списке устройств шлюза выделить контроллер с нужным идентификатором (ID приведен в квадратных скобках), при этом выбранный контроллер приобретает статус "**актив.**" (см. рисунок 2.28). Подключенный через **USB** контроллер будет иметь идентификатор, равный **0029.B000.0A18**. Пример выбора контроллера, подключенного через USB соединение, приведен на рисунке 2.32.

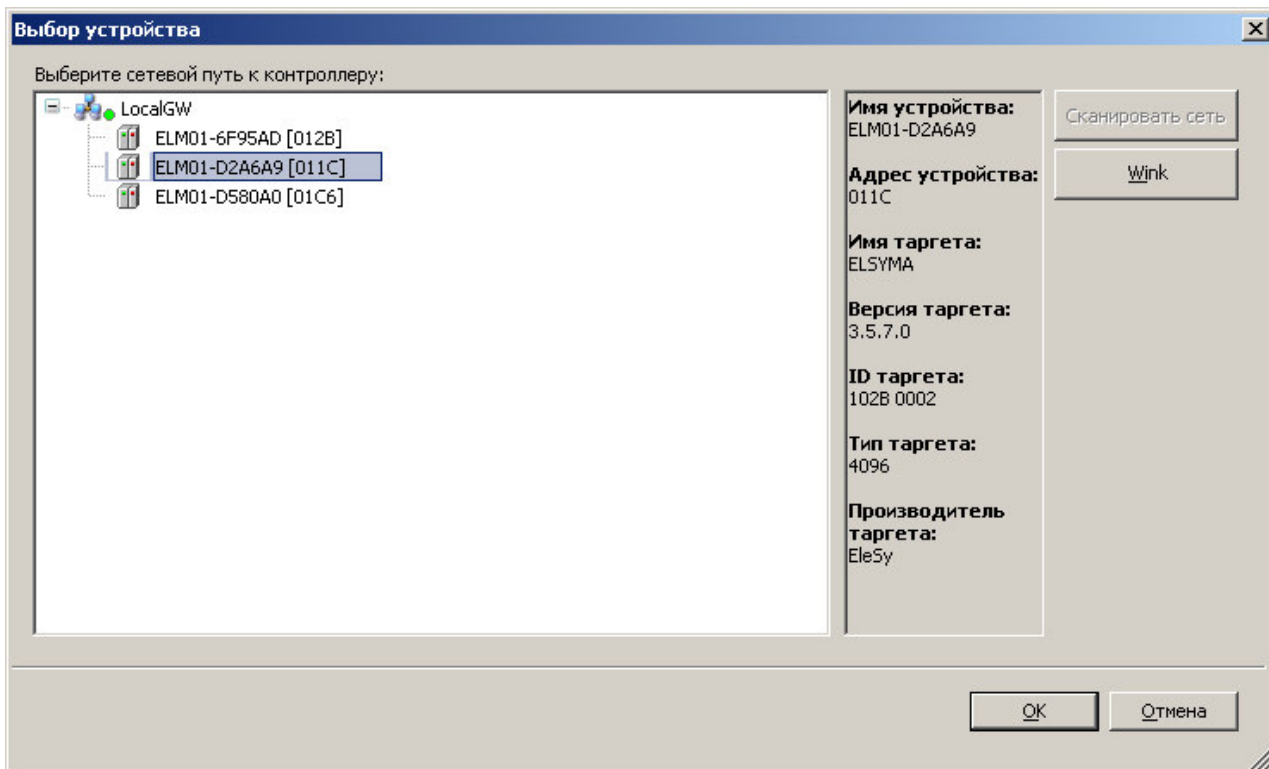


Рисунок 2.31 – Система разработки CoDeSys. Выбор активного контроллера, подключенного к общей сети

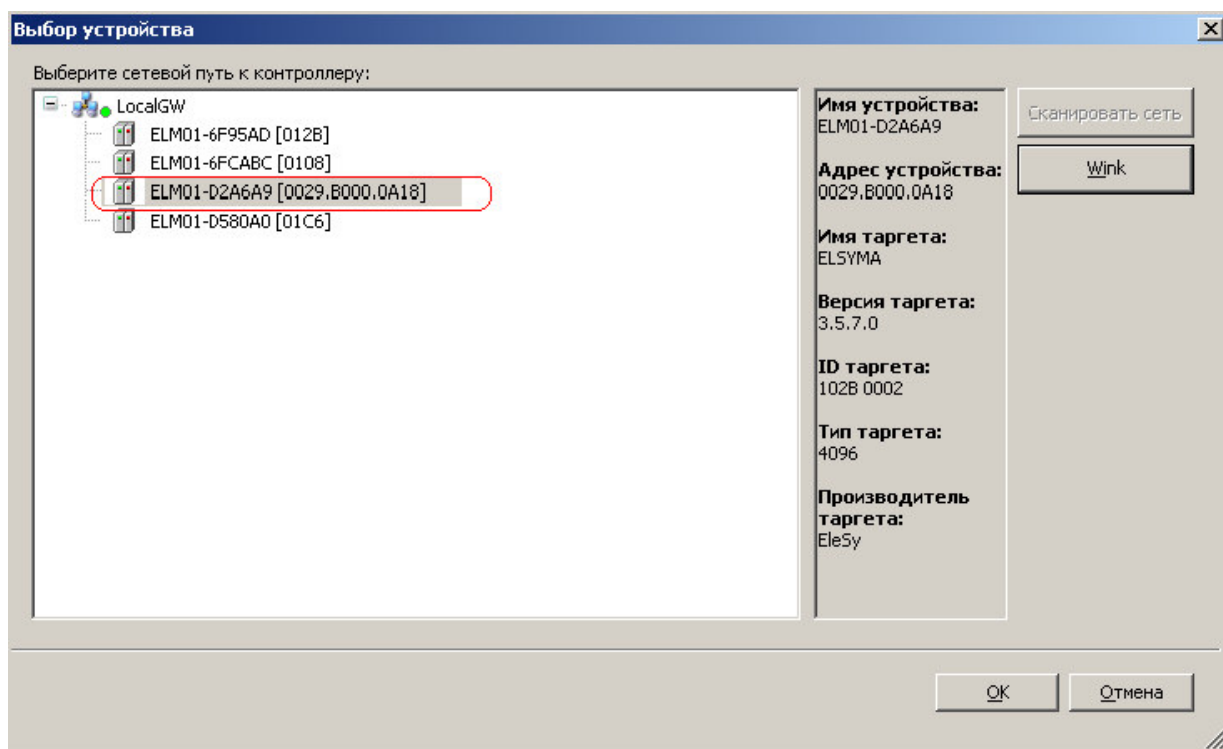


Рисунок 2.32 – Система разработки *CoDeSys*. Выбор активного контроллера, подключенного через USB соединение

3 Для идентификации выбранного контроллера можно нажать кнопку "**Wink**", при этом на выбранном контроллере происходит 10-кратное мигание индикаторами "**L1**" и "**L2**" (рисунок 1.5) с периодом 0,5 с. В случае, если в сети одновременно доступно несколько контроллеров, рекомендуется для проверки правильности выбора использовать команду **Wink**.

Все последующие операции будут выполняться именно с этим контроллером.

2.10.3 Подключение и загрузка проекта

Для подключения к контроллеру необходимо нажать клавиши **[Alt]+[F8]**. При появлении вопроса о загрузке задачи в контроллер следует нажать кнопку "**Yes**".

Примечание – Для загрузки проекта контроллер должен находиться в режиме обслуживания (MAINTENANCE MODE) (см. положение DIP-переключателя "**SW**" на рисунке 1.16).

При этом, если загружаемый проект полностью соответствует загруженному в контроллер, то происходит подключение к контроллеру без загрузки проекта. После загрузки проект находится в состоянии "Стоп". Признаком того, что программа находится в состоянии "Стоп", служит обозначение **СТОП** в строке статуса *CoDeSys*.

2.10.4 Запуск и отладка проекта

2.10.4.1 Запуск программы и мониторинг значений

Для запуска программы в контроллере следует нажать клавишу **[F5]**. Признаком того, что программа запущена, служит обозначение **ЗАПУСК** в строке статуса *CoDeSys*.

При успешном запуске программы контроллер переходит в **online**-режим и запускается процесс мониторинга, при котором:

- индикация соответствует рабочему режиму;
- данные обновляются с заданным периодом;

• в окне редактора *POU* и на закладках *Редактор параметров*, *Соотнесение входов/выходов* отображаются текущие значения переменных – см. рисунок 2.33.

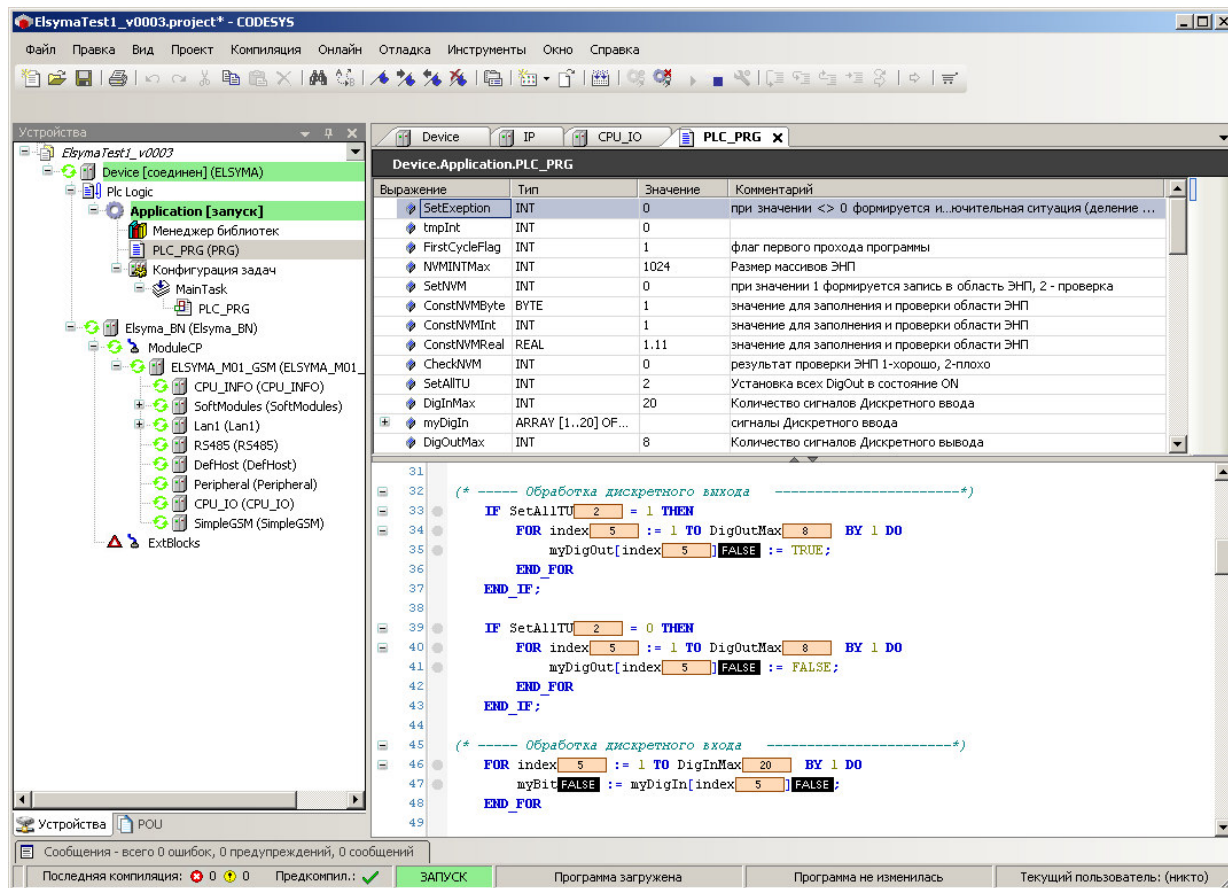


Рисунок 2.33 – Система разработки *CoDeSys*. Работа в *online*-режиме

2.10.4.2 Отладка проекта

Для отладки управляющей программы предназначен широкий спектр средств:

- мониторинг переменных с возможностью:
 - просмотра списка переменных;
 - записи и фиксации значений переменных;
 - контроля процесса выполнения **POU** в реальном времени (подсветка работающих элементов, отслеживание последовательных состояний и т.д.);
 - выполнение по циклам;
 - точки останова, пошаговое выполнение;
- возможность редактирования кода **POU** во время выполнения операции;
- развитые средства визуализации.

Подробное описание использования отладочных функций представлено в документации на систему программирования *CoDeSys*.

2.11 Загрузка ранее созданного проекта

Для загрузки ранее созданного проекта необходимо выполнить следующие действия:

- 1 Запустить систему разработки *CoDeSys* (см. 2.8.1.1).
- 2 Загрузить готовый проект с помощью меню **Файл – Открыть проект** выбрать проект для загрузки.
- 3 Подключиться к контроллеру (см. 2.10.3).
- 4 Запустить проект на исполнение (см. 2.10.4).

3 Конфигурирование контроллера

3.1 Добавление модуля в дерево конфигурации

Как уже описывалось в 2.8.1.2 и 2.8.1.3, для работы с контроллером необходимо создать конфигурацию системы, которая представлена в виде дерева устройств. Основным узлом (самый верхний уровень) является контроллер Элсима. При создании конфигурации пользователь должен выбрать тип контроллера, добавить в конфигурацию необходимый набор модулей УВВ (при необходимости) и задать набор необходимых программных модулей. В зависимости от выбранного типа контроллера может быть различный набор интерфейсов и поддерживаемых функций. На рисунке 3.1 приведен пример конфигурации (на рисунке отмечены модули, доступные для выбора пользователем).

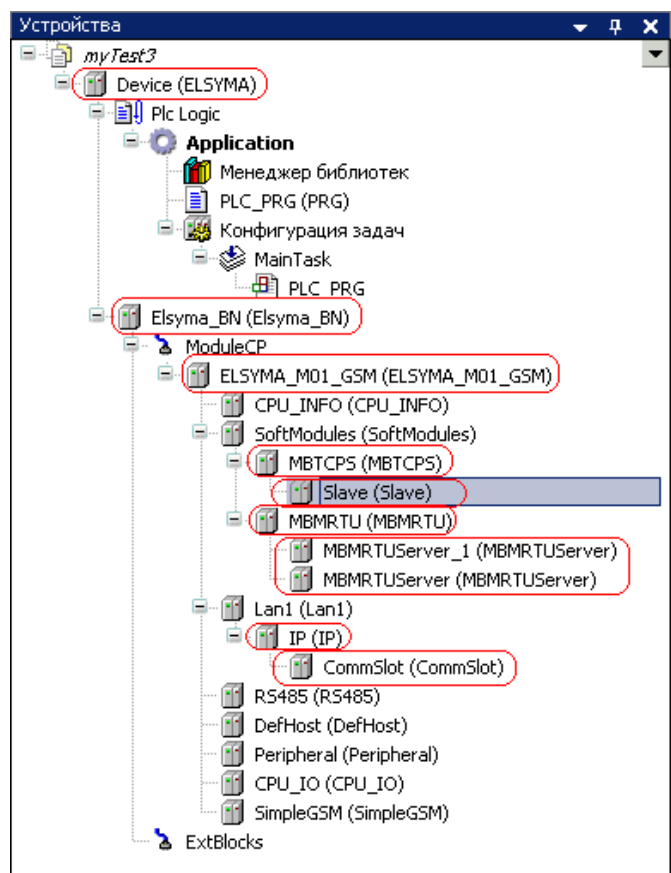


Рисунок 3.1 – Дерево устройств. Доступные для выбора модули конфигурации

3.2 Настройка параметров контроллера

На вкладке редактор параметров устройства **ELSYMA_M01** представлена информация модуля (на рисунке 3.2). Описание информационных параметров устройства **ELSYMA_M01** представлено в таблице 3.1.

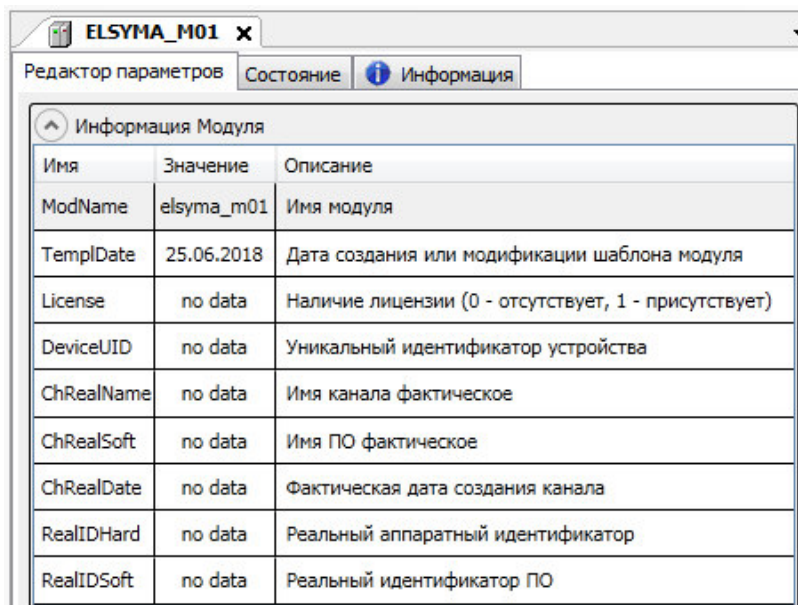


Рисунок 3.2 – ELSYMA_M01. Вкладка Редактор параметров

Таблица 3.1 – Устройство ELSYMA-M01. Информация Модуля

Имя	Значение "по умолчанию"	Описание
ModName	<i>Elsyma_m01</i>	Имя модуля
TemplDate	<i>25.06.2018</i>	Дата создания или модификации шаблона модуля
License	<i>no data</i>	Наличие лицензии (0 – отсутствует, 1 - присутствует)
DeviceUID	<i>no data</i>	Уникальный идентификатор устройства
ChRealName	<i>no data</i>	Имя канала фактическое
ChRealSoft	<i>no data</i>	Имя ПО фактическое
ChRealDate	<i>no data</i>	Фактическая дата создания канала
RealIDHard	<i>no data</i>	Реальный аппаратный идентификатор
RealIDSoft	<i>no data</i>	Реальный идентификатор ПО

Отображение версий ПО, текущих параметров работы и их изменение выполняются в системе *CoDeSys*, на закладке отображения информации и настройки контроллера (коннектор **ELSYMA_M01_XXX** – **CPU_INFO**). Для выполнения операции следует:

1 Подключиться к контроллеру (см. варианты подключения к контроллеру в разделах 2.9 и 2.10. Если неизвестны установленные сетевые параметры, то рекомендуется подключиться к контроллеру с помощью интерфейса USB и задать необходимые сетевые параметры).

2 Открыть закладку просмотра и настройки модуля **ELSYMA_M01_XXX**, выделив коннектор **CPU_INFO** в дереве устройств и дважды нажав левую кнопку "мыши".

3 Перейти на закладку **Редактор параметров** (рисунок 3.3).

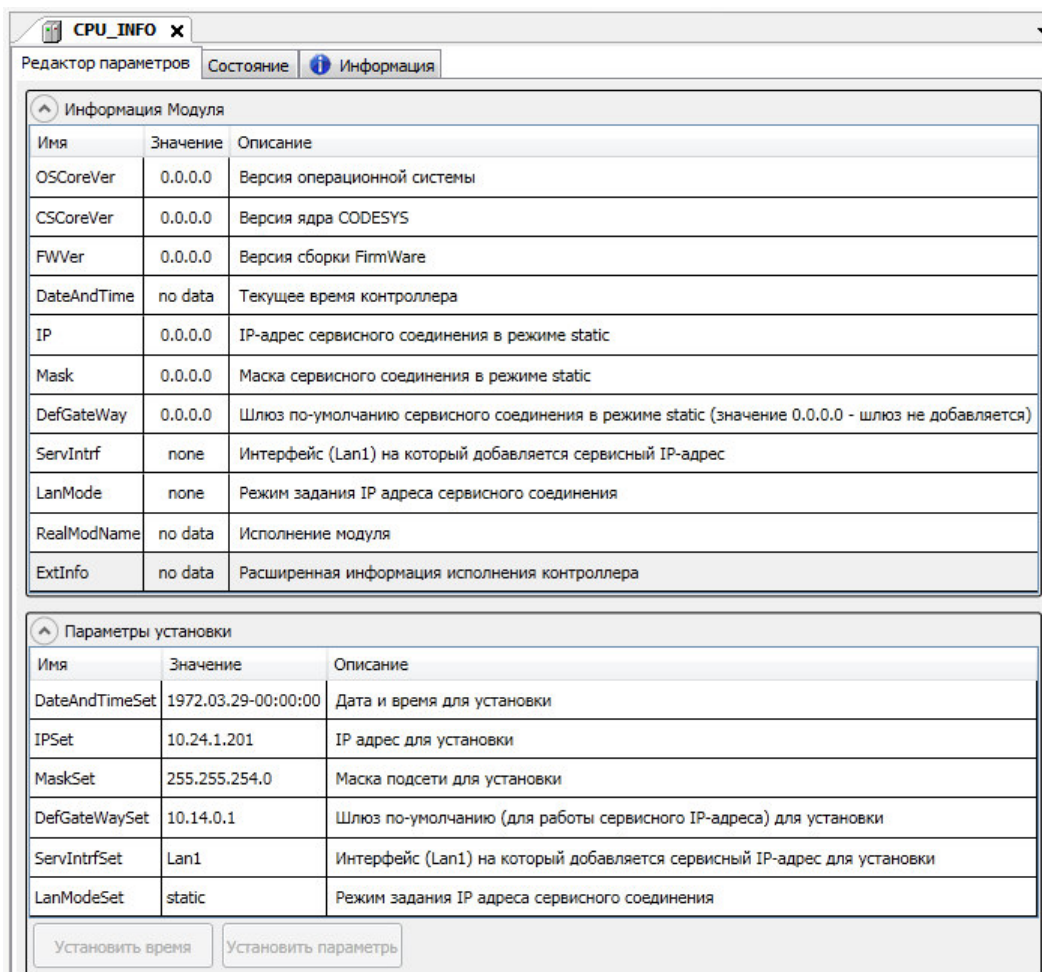


Рисунок 3.3 – Дерево устройств. Отображение информации и настройка контроллера

4 Проверить версии ПО и установленные сетевые параметры в области **Информация модуля**.

5 Изменить при необходимости соответствующие параметры контроллера в области **Параметры установки**.

Информационные параметры контроллера **ELSYMA_M01_XXX** перечислены в таблице 3.2. Данные параметры не доступны для редактирования пользователем.

Таблица 3.2 – Контроллер **ELSYMA-M01**. CPU_INFO. Информационные параметры

Имя	Значение "по умолчанию"	Описание
<i>OSCoreVer</i>	<i>0.0.0.0</i>	Версия операционной системы
<i>CSCoreVer</i>	<i>0.0.0.0</i>	Версия ядра исполнительной системы <i>CODESYS</i>
<i>FWVer</i>	<i>0.0.0.0</i>	Версия сборки FirmWare
<i>DateAndTime</i>	<i>no data</i>	Текущее время
<i>IP</i>	<i>0.0.0.0</i>	Установленный IP-адрес
<i>Mask</i>	<i>0.0.0.0</i>	Установленная сетевая маска
<i>DefGateWay</i>	<i>0.0.0.0</i>	Шлюз по-умолчанию сервисного соединения в режиме static (значение 0.0.0.0 – шлюз не добавляется)
<i>ServIntrf</i>	<i>none</i>	Интерфейс (Lan1), на который добавляется сервисный IP-адрес
<i>LanMode</i>	<i>none</i>	Режим задания IP адреса сервисного соединения
<i>RealModName</i>	<i>no data</i>	Исполнение модуля
<i>ExtInfo</i>	<i>no data</i>	Расширенная информация исполнения контроллера

Параметры установки представлены в таблице 3.3.

Таблица 3.3 – Контроллер ELSYMA-M01. CPU_INFO. Параметры установки

Имя	Значение "по умолчанию"	Описание
<i>DateAndTimeSet</i>	<i>1972.03.29-00:00:00</i>	Дата и время для установки
<i>IPSet</i>	<i>10.24.1.201</i>	IP адрес для установки
<i>MaskSet</i>	<i>255.255.254.0</i>	Маска подсети для установки
<i>DefGateWaySet</i>	<i>10.14.0.1</i>	Шлюз по-умолчанию (для работы сервисного IP-адреса) для установки
<i>ServIntrfSet</i>	<i>Lan1</i>	Интерфейс (Lan1), на который добавляется сервисный IP-адрес
<i>LanModeSet</i>	<i>static</i>	Режим задания IP адреса сервисного соединения

Для изменения параметров контроллера **ELSYMA_M01_XXX** необходимо в области **Параметры установки** задать нужные значения и нажать на кнопку "**Установить время**" или "**Установить параметры**" (соответственно).

ВНИМАНИЕ! При изменении сетевых параметров *IPSet*, *MaskSet*, *DefGateWaySet*, *LanModeSet*, *ServIntrfSet* новые параметры вступят в силу только после перезагрузки контроллера.

3.2.1.1 Установка сервисного IP-адреса

Для установки новых сетевых параметров ЦП необходимо в закладке **Редактор параметров** устройства **CPU_INFO** в области **Параметры установки** установить необходимые значения параметров *IPSet*, *MaskSet*, *DefGateWaySet* (описание представлено в таблице 3.3) и нажать кнопку "**Установить параметры**". Установленные будут отображены в области **Информационные параметры**.

ВНИМАНИЕ! Описанный выше процесс установки сервисного IP-адреса актуален только для задания статического IP-адреса (параметр *LanModeSet* равен *static*). При изменении сетевых параметров *IPSet*, *MaskSet*, *DefGateWaySet*, *LanModeSet*, *ServIntrfSet* новые параметры вступят в силу только после перезагрузки контроллера.

3.2.1.2 Установка RTC

Для установки реального времени (**RTC**) необходимо в закладке **Редактор параметров** устройства **CPU_INFO** в области **Параметры установки** задать необходимое значение параметра *DateAndTimeSet* (описание представлено в таблице 3.3) и нажать кнопку "**Установить время**". Установленное значение будет отображено в области **Информационные параметры**.

Примечание: нельзя установить параметр *DateAndTimeSet* меньше, чем дата и время сборки системы.

3.2.2 Библиотека для доступа к системной информации контроллера

Библиотека **SysInfoLibrary** предоставляет возможность получения в задаче пользователя *CoDeSys* различной системной информации. Данная библиотека функционирует во всех исполнениях контроллера Элсима.

Примечание – Данная библиотека добавляется в систему *CoDeSys* вручную.

В библиотеке реализованы следующие функциональные блоки:

- *GetSysInfo* – функциональный блок для получения системной информации контроллера Элсима-M01.
- *GetSysInfoRU* – функциональный блок для получения системной информации модулей удаленного ввода-вывода (Элсима-D01, Элсима-A01, Элсима-DA01).

Входные параметры ФБ *GetSysInfo* – структура **elsyinfotype**. Структура включает в себя параметры, представленные в таблице 3.4.

Таблица 3.4 – Параметры структуры **elsyinfotype**

Имя параметра	Описание
<i>OSCoreVer</i>	Версия операционной системы
<i>CSCoreVer</i>	Версия ядра CoDeSys
<i>FWVer</i>	Версия сборки Firm Ware
<i>DateAndTime</i>	Текущее время контроллера
<i>IP</i>	IP –адрес сервисного соединения в режиме static
<i>Mask</i>	Маска сервисного соединения в режиме static
<i>DefGateWay</i>	Шлюз по умолчанию сервисного соединения в режиме static
<i>ServIntrf</i>	Интерфейс (Lan1), на который добавляется сервисный IP-адрес
<i>LanMode</i>	Режим задания IP-адреса сервисного соединения
<i>RealModName</i>	Исполнение модуля (elsyma-m01-24p-gsm/elsyma-m01-24p-nogsm)
<i>ExtInfo</i>	Расширенная информация исполнения контроллера
<i>RealSoft</i>	Имя ПО submodule ввода/вывода
<i>SoftVer</i>	Версия ПО submodule ввода/вывода
<i>RealDate</i>	Дата создания ПО submodule ввода/вывода

Выходные значения функции *GetSysInfo*:

- Строка запрошенной информации;
- Строка "Info is not found" в случае невозможности получения запрошенной информации (некорректно задан идентификатор запрашиваемой информации, отсутствие запрашиваемой информации в системе и т.п.);
- Строка "sys error" в случае возникновения системной ошибки при выполнении функции.

Графическое представление ФБ *GetSysInfo* представлено на рисунке 3.4.

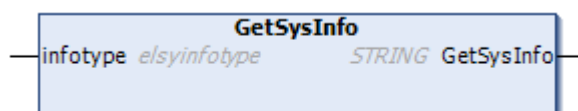


Рисунок 3.4 – Графическое представление ФБ *GetSysInfo*

Входные параметры функции *GetSysInfoRU* – структура **elsyinfotyperu** и номер позиции модуля УВВ, для которого запрашивается информация. Структура **elsyinfotyperu** включает в себя параметры, представленные в таблице 3.5.

Таблица 3.5 – Параметры структуры **elsyinfotyperu**

Имя параметра	Описание
<i>RealName</i>	Имя модуля фактическое
<i>RealSoft</i>	Имя ПО фактическое
<i>RealVer</i>	Фактическая версия ПО модуля
<i>RealDate</i>	Фактическая дата создания модуля

Выходные значения функции *GetSysInfoRU*:

- Строка запрошенной информации;
- Строка "Info is not found" в случае невозможности получения запрошенной информации (некорректно задан идентификатор запрашиваемой информации, отсутствие запрашиваемой информации в системе и т.п.);
- Строка "Module is not found" в случае отсутствия заданного номера модуля в конфигурации;

- Строка "sys error" в случае возникновения системной ошибки при выполнении функции.

Графическое представление ФБ *GetSysInfo* представлено на рисунке 3.5.

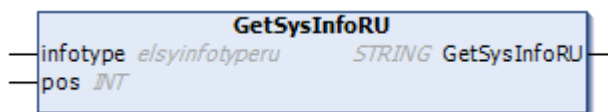


Рисунок 3.5 – Графическое представление ФБ *GetSysInfoRU*

На рисунке 3.6 представлен пример программного кода, в котором с помощью функции *GetSysInfo* в задачу пользователя получены данные о версии сборки Firm Ware контроллера Элсима, а с помощью функции *GetSysInfoRU* получены данные о фактическом имени ПО модуля УВВ, находящегося в позиции 1.

```

PLC_PRG x
1  PROGRAM PLC_PRG
2  VAR
3      myFWVer    : STRING; //переменная, отображающая версию сборки FirmWare контроллера Элсима
4      myNamePos1 : STRING; //переменная, отображающая фактическое имя ПО модуля УВВ в позиции 1
5  END_VAR

1  (* С помощью функции GetSysInfo получаем информацию о версии сборки Firm Ware контроллера *)
2  myFWVer := SysInfoLibrary.GetSysInfo( SysInfoLibrary.elsyinfotype.FWVer );
3
4  (* С помощью функции GetSysInfoRU получаем информацию о фактическом имени ПО модуля УВВ в позиции 1 *)
5  myNamePos1 := SysInfoLibrary.GetSysInfoRU( SysInfoLibrary.elsyinfotype.RealSoft, 1 );

```

Рисунок 3.6 – Пример применения функций *GetSysInfo* и *GetSysInfoRU*

3.3 Программный модуль Modbus TCP Master

В данном подразделе представлено описание программного модуля с поддержкой протокола *Modbus TCP* в режиме *Master (Client)*. Символьное обозначение модуля – **МВТСРМ**.

Применение этого программного модуля в составе ПО ПЛК позволяет использовать интерфейс *Ethernet* контроллера для взаимодействия с устройствами в сетях с применением протокола *Modbus TCP* [согласно "MODBUS MESSAGING ON TCP/IP IMPLEMENTATION GUIDE" и "MODBUS APPLICATION PROTOCOL SPECIFICATION"].

На рисунке 3.7 представлен вид закладки *Редактор параметров* программного модуля **МВТСРМ**. Параметры, приведенные в поле *Информация Модуля*, используются для идентификации и недоступны для редактирования пользователем.

ВНИМАНИЕ! Параметры поля *Конфигурационные Параметры Модуля* используются для настройки данного модуля, и не рекомендуется их изменять.

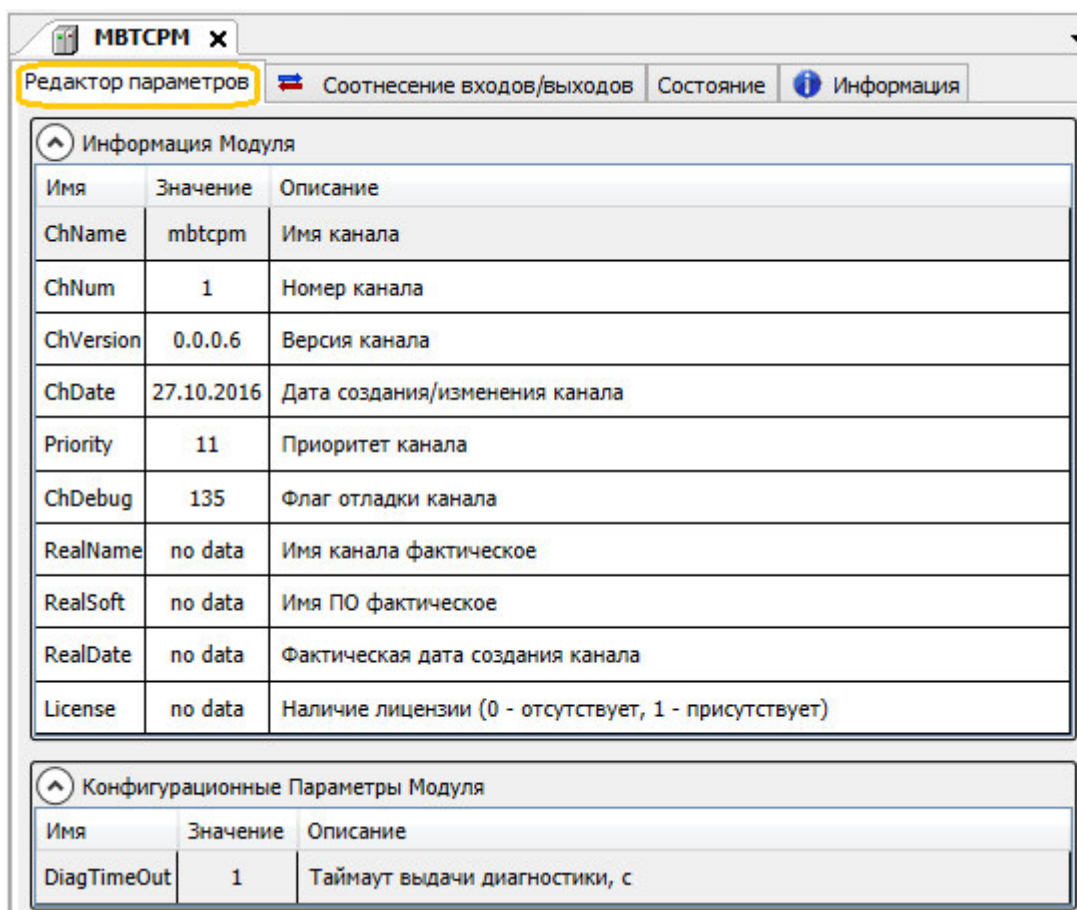


Рисунок 3.7 – Модуль Modbus TCP Master. Информация о модуле на закладке "Редактор параметров"

На рисунке 3.8 представлен вид закладки "Соотнесение входов/выходов" для программного модуля **Modbus TCP Master** с диагностическими сигналами. Набор сигналов соответствует единому шаблону программных модулей **ELSYMA_M01_GSM**.

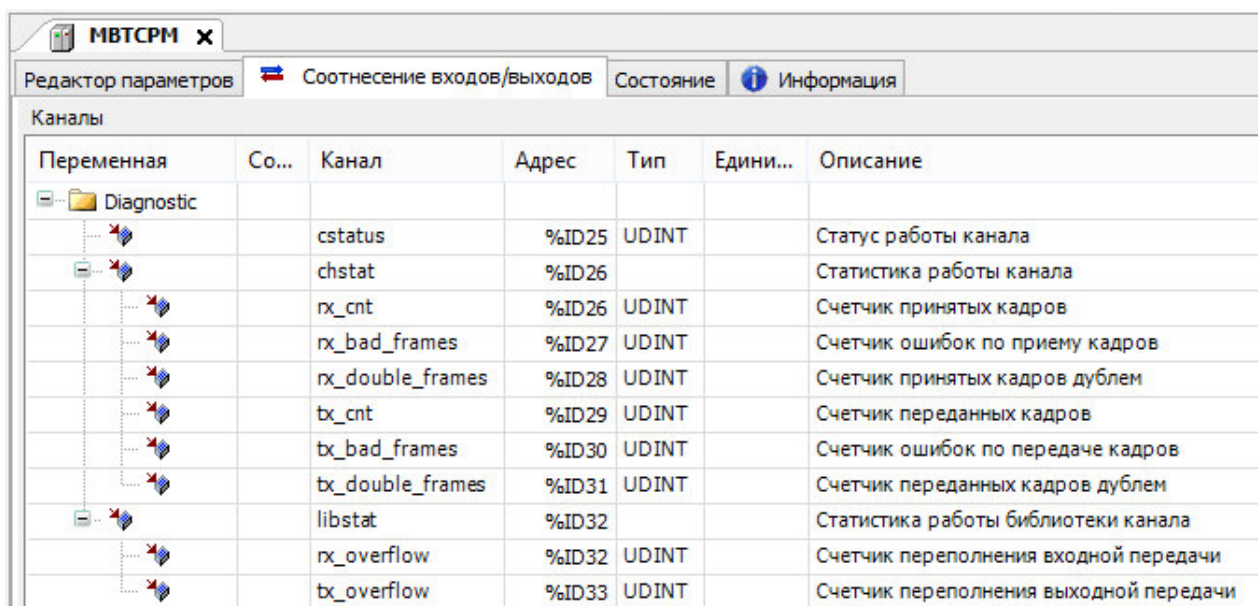


Рисунок 3.8 – Модуль Modbus TCP Master. Диагностические сигналы на закладке "Соотнесение входов/выходов"

В данной версии сервисной программы конфигурирования программного модуля **Modbus TCP Master** для настройки информационного взаимодействия необходимо:

1 Добавить и сконфигурировать устройство **Server** (в "дереве устройств" по иерархии располагается ниже **МВТСРМ**) (см. 3.3.1).

2 Для каждого из подчинённых устройств создать и сконфигурировать коммуникационный слот (**CommSlot**) в "дереве устройств" (**ELSYMA_m01_GSM -> Lan -> IP**) (см. 3.12).

3.3.1 Общий принцип конфигурирования Modbus TCP Master

Конфигурирование протокола *Modbus TCP* в режиме *Master* разбивается на следующие шаги:

1 Добавление в дерево конфигурации модуля – **МВТСРМ**. Для примера на рисунке 3.9 приведена структурная схема конфигурируемой системы. В данном примере **Master TCP** опрашивает три подчиненных устройства **Modbus TCP Slave** с установленными IP-адресами **IP1**, **IP2**, **IP3**.

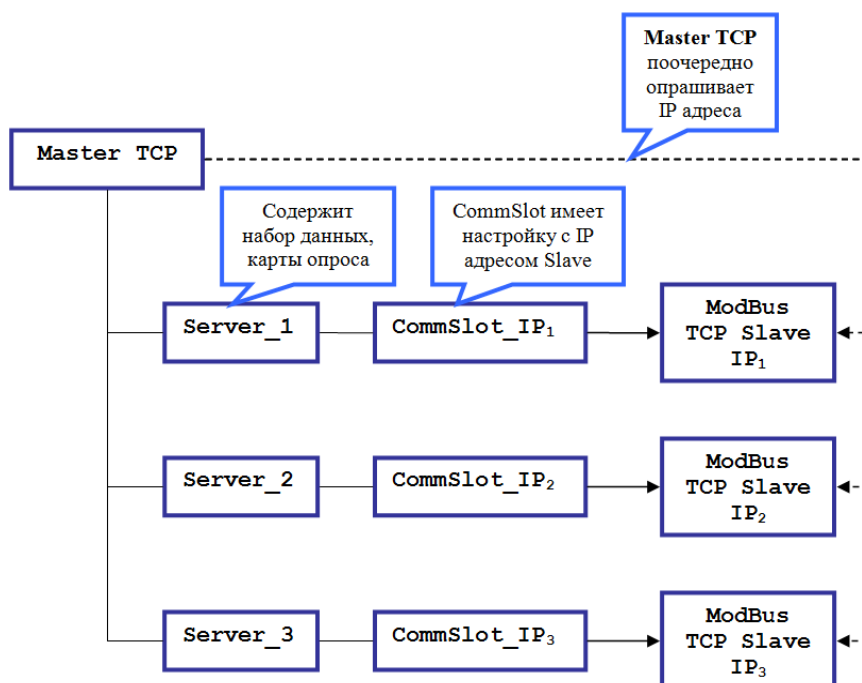


Рисунок 3.9 – Структура опроса модуля Modbus TCP Master

1 Добавление в дерево конфигурации для модуля **МВТСРМ** опрашиваемых подчиненных устройств – **МВТСРМServer (Slave)**.

2 Формирование для каждого подчиненного устройства **МВТСРМServer (Slave)** карты опроса.

3 Настройка параметров **CommSlot** и связывание его с необходимым модулем **МВТСРМServer (Slave)** (см. рисунок 3.10).

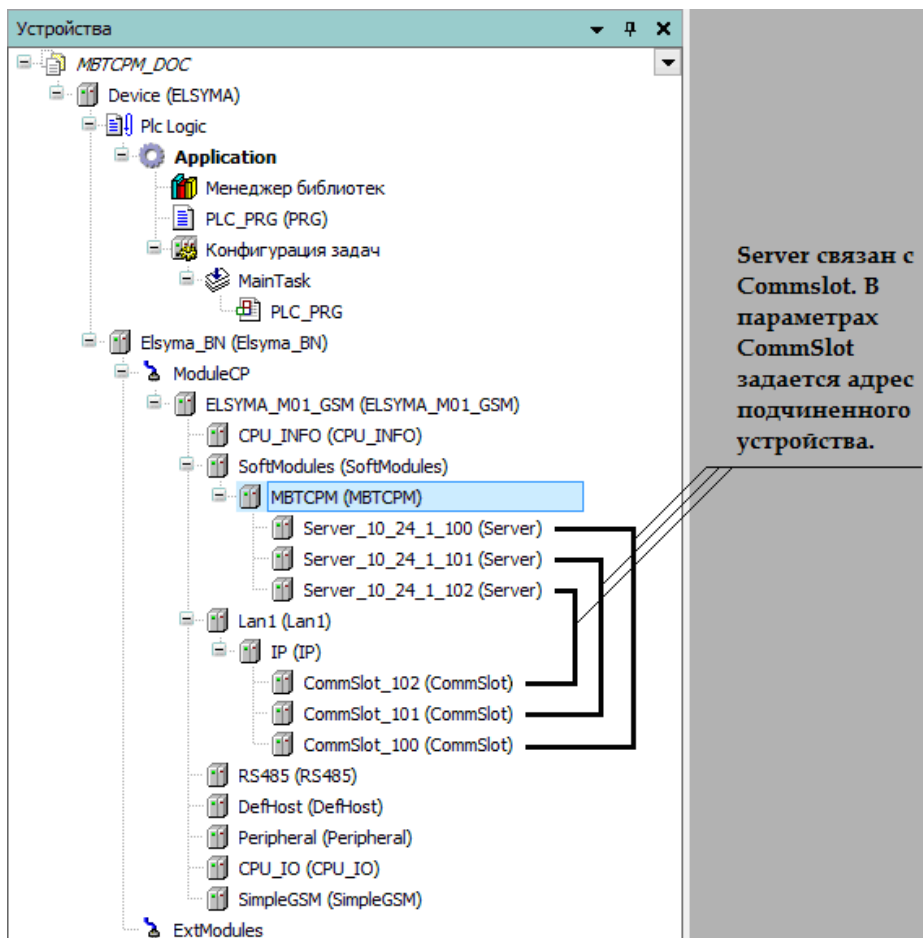


Рисунок 3.10 – Дерево устройств. Связь сигналов MBTCPMServer (Slave) и CommSlot

При конфигурировании *CommSlot* необходимо задать режим работы *TCP Client* в параметре *ModeTransport* и задать адрес подчиненного устройства в параметре *SlaveIPAddress* (см. рисунок 3.11). Подробное описание параметров *CommSlot* представлено в 3.12.4.

Редактор параметров		
Редактор соединения		
Состояние		
Информация		
Конфигурационные Параметры Модуля		
Имя	Значение	Описание
Port	502	Номер порта
MaxConnections	4	Количество соединений
ModeTransport	TCP Client	Режим работы транспортного уровня: 0 - в качестве транспортного уровня использовать TCP Server;
SendBuff	4096	Размер буфера хранения перед отправкой
TCP_NoDelay	4	Отключение алгоритма Нагля (Nagle algorithm) для обеспечения передачи данных без ожидания запятой
SlaveIPAddress	10.24.1.100	

Рисунок 3.11 – Параметры *CommSlot*

3.3.2 Модуль Server для MBTCPM

Модуль **Server** предназначен для подготовки конфигурации сигналов удалённого контроллера, опрашиваемого по протоколу *Modbus TCP* (**Slave**-устройство). Символьное обозначение модуля, используемое в сервисной программе – **Server**.

Для каждого опрашиваемого устройства создаётся выделенное сетевое соединение и должна быть подготовлена конфигурация сигналов. Программное обеспечение модуля обеспечивает до 16-ти одновременных подключений к **Server**-устройствам, при этом для каждого **Server**-устройства допускается только одно сетевое соединение. В каждом

Server-устройстве возможно создавать до 16-ти коммуникационных каналов. Каждый канал представляет собой один запрос к подчинённому устройству. Общее количество сигналов в конфигурации ведущего устройства (Master) не должно превышать 1000 сигналов.

Протокол *Modbus* оперирует битовыми значениями (ячейка – **Coil** и вход – **Input**) и 16-ти битными регистрами (регистр – **Holding Register** и входной регистр – **Input Register**).

Поддерживаемые контроллером функции *Modbus* представлены в таблице 3.6.

Таблица 3.6 – Поддерживаемые функции Modbus

Код функции	Описание функции
01 (0x01)	Read Coils (Чтение битовой ячейки)
02 (0x02)	Read Discrete Inputs (Чтение битового входа)
03 (0x03)	Read Holding Registers (Чтение содержимого регистров)
04 (0x04)	Read Input Registers (Чтение содержимого входных регистров)
05 (0x05)	Write Single Coil (Запись битовой ячейки)
06 (0x06)	Write Single Register (Запись в единичный регистр)
15 (0x0F)	Write Multiple Coils (Запись группы битовых ячеек)
16 (0x10)	Write Multiple registers (Запись группы регистров)

В зависимости от результата выполнения операции, подчиненное устройство возвращает нормальный ответ на запрос либо ответ исключения. В нормальном ответе подчиненное устройство возвращает код функции первоначального запроса и запрошенные данные. Ответ исключения формируется, когда при обработке запроса подчинённым устройством возникает та или иная ошибка, и значение в поле данных содержит код исключения, поясняющий причину возникновения ошибки. Код исключения может передаваться в программу пользователя через специальную, определённую в конфигурации, переменную. Основные коды исключений описаны в таблице 3.7.

Таблица 3.7 – Поддерживаемые коды исключения по протоколу Modbus

Код	Исключение	Описание
00	OK	Нет ошибки
01	ILLEGAL FUNCTION	Генерируется Modbus Server -устройством в ответ на запрос с кодом функции, которая не поддерживается (неисправимая ошибка, требуется коррекция конфигурации)
02	ILLEGAL DATA ADDRESS	Генерируется Modbus Server -устройством в ответ на запрос с адресом регистров, не входящих в допустимый диапазон адресов (неисправимая ошибка, требуется коррекция конфигурации)
03	ILLEGAL DATA VALUE	Генерируется при попытке записи в регистр значения, несоответствующего предусмотренному диапазону значений, или при запросе больше максимального объема данных
04	SERVER DEVICE FAILURE	Генерируется Modbus Server -устройством при возникновении неисправности устройства в процессе обработки запроса (неисправимая ошибка, требуется проверка Modbus Slave)
05	ACKNOWLEDGE	Генерируется Modbus Server -устройством при невозможности быстро ответить на запрос, для предотвращения тайм-аута ответа (исправимая ошибка, требует повтора запроса)
06	SLAVE DEVICE BUSY	Генерируется при невозможности немедленно обработать запрос. Например, при обработке команды записи, если предыдущая команда не завершилась (команда не передана в ЦП)
0A	GATEWAY PATH UNAVAILABLE	Ошибка шлюза. Сеть недоступна из-за неконфигурированного или перегруженного внешнего оборудования
0B	GATEWAY TARGET DEVICE FAILED TO RESPOND	Подчиненное устройство, подключенное через шлюз, не отвечает
81	NO CONNECTION	Отсутствует соединение с устройством Modbus Server

Настройка работы модуля **Server** в системе *CoDeSys* состоит из следующих этапов:

- 1 Настройка конфигурационных параметров (см. 3.3.2.1).
- 2 Конфигурирование запросов (карты сигналов) протокола *Modbus TCP* (см. 3.3.2.2).
- 3 Соотнесение сигналов (см. 3.3.2.3).

3.3.2.1 Настройка конфигурационных параметров модуля **Server**

Настройка конфигурационных параметров осуществляется на закладке **Редактор параметров** модуля **Server**. Для выполнения операции следует:

- 1 Открыть закладку просмотра и настройки модуля **Server**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Перейти на закладку **Редактор параметров** (рисунок 3.12).
- 3 Настроить конфигурационные параметры модуля. Описание параметров представлено в таблице 3.8.

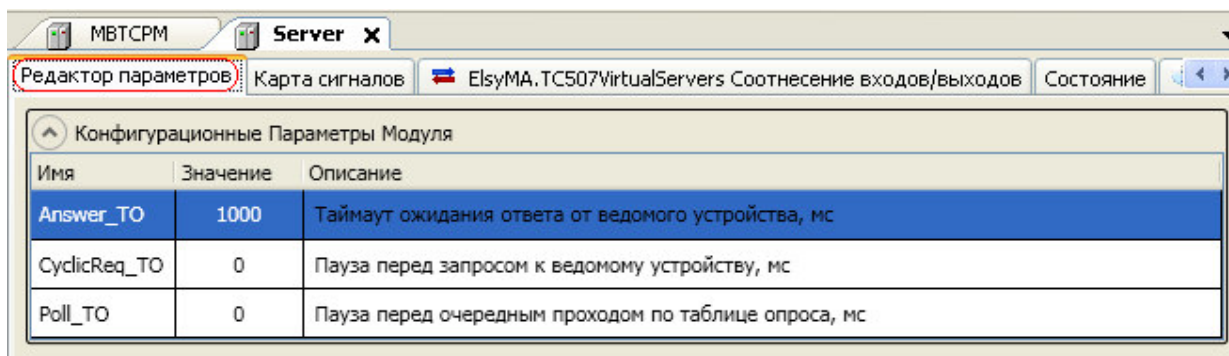


Рисунок 3.12 – Модуль **Modbus TCP Server**. Закладка "Редактор параметров"

Таблица 3.8 – Модуль **Modbus TCP Server**. Конфигурационные параметры

Имя	Значение по умолчанию	Описание
<i>Answer_TO</i>	<i>1000</i>	Адрес подчиненного устройства. Если подчиненное устройство получило запрос с адресом, отличным от адреса, заданного в конфигурационных параметрах, то данный запрос игнорируется. Тайм-аут ожидания ответа от ведомого устройства (мс). Может потребоваться увеличение, если устройство малопроизводительное или большая задержка в канале передачи данных
<i>CiclicReq_TO</i>	<i>0</i>	Пауза перед запросом к ведомому устройству (мс). Может быть необходима для снижения нагрузки на канал и малопроизводительное устройство
<i>Poll_TO</i>	<i>0</i>	Пауза перед очередным проходом по таблице опроса (мс) Может быть необходима для уменьшения нагрузки на канал

3.3.2.2 Конфигурирование передачи данных для **Modbus TCP Server**

3.3.2.2.1 Понятие коммуникационного канала

Формирование сигналов для чтения/записи данных по протоколу *Modbus TCP* осуществляется через создание коммуникационного канала, описывающего один непрерывный блок данных. Каналы могут быть логически сгруппированы в секции.

Для работы с каналами и секциями используются следующие операции:

- 1 Создание канала (см. 3.4.2.2).
- 2 Редактирование (имени секции или атрибутов канала в зависимости от контекста).

3 Копирование (для вставки в такой же модуль в конфигурации).

4 Удаление.

Для выполнения операции следует:

1 Вызвать закладку просмотра и настройки модуля, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".

2 Выбрать закладку *Карта сигналов*.

3 Вызвать контекстное меню и выбрать соответствующую команду (рисунок 3.13).

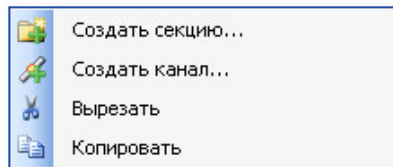


Рисунок 3.13 – Модуль Modbus TCP Server. Общий вид контекстного меню на карте сигналов при создании канала или секции

Канал имеет следующие атрибуты (рисунок 3.14):

- **Имя** – имя секции/канала. Задаёт условное наименование блока данных;

Имя	Адрес подчиненн...	Стартовый ад...	Количество дан...	Функция	Тип данных	Изменять порядок байт	Изменять порядок слов	Тип канала	Описание
Channel1	255	1	100	ReadDiscreteInputs	BIT	False	False	Входной	
Channel2	255	1	10	ReadCoils	BIT	False	False	Входной	
Channel3	255	1	100	ReadInputRegisters	WORD	False	False	Входной	
Channel4	255	1	1	ReadHoldingRegisters	WORD	False	False	Входной	

Рисунок 3.14 – Модуль Modbus TCP Server. Вид закладки "Карта сигналов"

- **Адрес подчинённой станции** – адрес опрашиваемой станции (**Server**);
- **Стартовый адрес** – начальный адрес блока данных, расположенных в непрерывной области адресов. Для удобства просмотра адреса можно выбрать формат представления адреса – *hex* или *dec*. Необходимо установить в соответствии с адресами сигналов на **Server**-устройстве;
- **Количество данных** – количество данных в блоке;
- **Функция** – функция определяет один из четырех видов данных Modbus (определяется кодом запроса к **Server**-устройству);
- **Тип данных** – тип данных выбирается для заданного сегмента (таблица 3.9);

Таблица 3.9 – Допустимые типы данных в зависимости от типов сегментов

Тип данных	Сегмент					
	Discrete Inputs	Coils		Input Registers	Holding Registers	
		Single	Multiple		Single	Multiple
BOOL	+	+	+	+	+	+
BYTE	+	–	+	+	+	+
SINT	+	–	+	+	+	+
USINT	+	–	+	+	+	+
WORD	+	–	+	+	+	+
INT	+	–	+	+	+	+
UINT	+	–	+	+	+	+
DWORD	+	–	+	+	–	+
DINT	+	–	+	+	–	+
UDINT	+	–	+	+	–	+
LWORD	–	–	–	+	–	+
LINT	–	–	–	+	–	+
REAL	–	–	–	+	–	+
LREAL	–	–	–	+	–	+

- **Тип канала** – определяет доступ к данным со стороны опросчика:
 - ◊ входной канал – используется для записи данных в подчиненное устройство (сегменты **Coils** и **HoldingRegisters**);
 - ◊ выходной канал – используется для чтения данных из подчиненного устройства (сегменты **DiscretesInputs**, **InputRegisters**, **Coils** и **HoldingRegisters**).

Типы сегментов и принцип размещения данных представлены на рисунке 3.15.

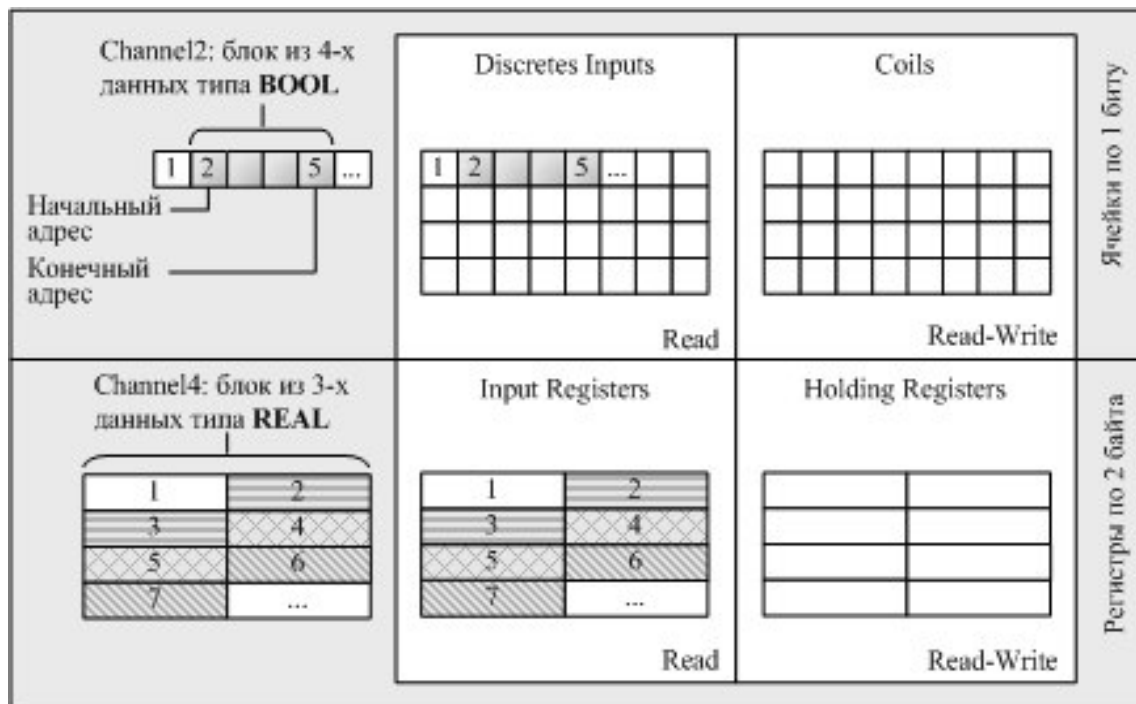


Рисунок 3.15 – Модуль Modbus TCP Server.
Типы сегментов и принцип размещения данных

Кроме этих полей, в окне установки параметров канала (рисунок 3.16) можно выбрать значения для следующих параметров:

- **Изменять порядок байтов** – устанавливается для изменения порядка последовательности байтов в значении сигнала с размером от двух байтов. Данное значение условно делится на слова (размер слова 2 байта) и в каждом слове выполняется изменение последовательности байтов (применимо к **HoldingRegisters** и **InputRegisters**);

- **Изменять порядок слов** – устанавливается для изменения порядка слов в обратной последовательности в значении сигнала с размером от двух слов (применимо к **HoldingRegisters** и **InputRegisters**);

- **Описание.**

При двойном нажатии левой кнопки "мыши" в области отображения значений любого атрибута, открывается окно редактирования "**Редактор канала**" (рисунок 3.16), описание которого представлено в 3.3.2.2.3.

3.3.2.2.2 Назначение переменных и имен сигналам канала

При создании канала предоставляется возможность назначить сигналам новую или существующую переменную, а также возможность задать параметры для автоматического формирования имен структуры (автонаименования). Формирование имен осуществляется на основе шаблонов имен, представляющих собой произвольные комбинации служебных последовательностей (таблица 3.10) и обычных символов в соответствии с разрешенными в IEC 61131-3 символами для имен переменных.

Таблица 3.10 – Символьные последовательности для шаблонов имен

Название	Обозначение	Пример	Описание
Имя канала	%CHANNEL%	Channel3	Заданное имя канала
Код сегмента (hex)	%PTYPE_X%	x03	Код сегмента данных в разных форматах
Код сегмента (dec)	%STYPE_D%	03	
Код сегмента (строка)	%STYPE_S%	HoldingRegisters	
Тип данных (IEC 61131)	%TYPE%	SIGNAL_REAL_T	Тип данных сигнала
Текущий адрес сигнала (hex)	%CURADDR_X%	x01	Адрес сигнала в различных форматах
Текущий адрес сигнала (dec)	%CURADDR_D%	1	
Номер сигнала в канале (hex)	%NUM_X%	x0000	Номер сигнала в канале в разных форматах
Номер сигнала в канале (dec)	%NUM_D%	0	
Индекс элемента в массиве	%ARRAYINDEX%	[1]	Индекс сигнала в заданном диапазоне

Редактор канала

Имя: Channel1

Описание:

Параметры канала

Адрес подчиненной станции: 255

Код функции: ReadDiscreteInputs

Стартовый адрес: 1

Количество данных: 100

Режим работы записи: Cyclic

Сигнал управления записью:

Изменить порядок байтов:

Изменить порядок слов:

Передавать сигнал всегда:

BOOL

Конечный адрес: 100

hex dec

Привязка и автоименование

Параметры привязки и автоименования

Общие

Шаблон имени: ... Предпросмотр

OK Отмена

Рисунок 3.16 – Модуль Modbus TCP Server. Создание канала

3.3.2.2.3 Порядок создания канала

Для создания канала следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля **Server** в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Выбрать закладку *Карта сигналов*.
- 3 Вызвать контекстное меню и выбрать команду *Создать канал...*

4 В окне "Редактор канала" (рисунок 3.16) в поле **Имя:** задать имя канала, в поле **Описание:** текстовое описание канала.

5 Задать атрибуты канала с помощью элементов группы **Параметры канала:**

- В поле "**Адрес подчинённой станции**" – ввести адрес опрашиваемой станции (**Server**);

- В выпадающем списке "**Код функции**": выбрать тип запроса:

- ◇ Read Coils (01 (0x01));
- ◇ Read Discrete Inputs 02 (0x02);
- ◇ Read Holding Registers 03 (0x03);
- ◇ Read Input Registers (04 (0x04));
- ◇ Write Single Coil (05 (0x05));
- ◇ Write Single Register (06 (0x06));
- ◇ Write Multiple Coils (15 (0x0F));
- ◇ Write Multiple registers 16 (0x10).

В выпадающем списке справа выбрать тип данных. Набор допустимых типов данных определяется типом сегмента (таблица 3.9).

- С помощью счетчика **Стартовый адрес:** и **Количество данных:** установить начальный адрес блока данных и количество данных в блоке. При этом в поле **Конечный адрес:** отображается конечный адрес блока.

- В выпадающем списке "**Режим работы записи**": выбрать тип запроса по старту модуля:

- ◇ циклический опрос (по умолчанию): **Cyclic** – выдаётся в каждом цикле поллинга при обращении к станции;
- ◇ запрет выдачи запроса: **DenyRequest** – запрос не выдаётся, пока режим не будет изменён из задачи пользователя;
- ◇ выдача запроса только один раз: **SingleRequest** – запрос выдаётся только в первом цикле поллинга (может быть изменён из задачи пользователя);



- Для данных с типом в сегменте **Input Registers** или **Holding Registers**, при необходимости, можно установить флаги, изменяющие порядок байт в получаемых от подчинённого устройства данных:

- ◇ "**Изменить порядок байтов**" – устанавливается для изменения порядка последовательности байтов в значении сигнала с размером от двух байтов. Данное значение условно делится на слова (размер слова – два байта) и в каждом слове выполняется изменение последовательности байтов;
- ◇ "**Изменить порядок слов**" – устанавливается для изменения порядка слов в обратной последовательности в значении сигнала с размером от двух слов;
- ◇ Радиокнопка "**Передавать сигнал всегда**" позволяет запретить или разрешить выдачу сигнала в задачу пользователя при неизменном его значении после выполнения данного запроса.

6 Настроить параметры привязки и автонаименования:

- Нажать кнопку

▼ Привязка и автоименование

- Задать шаблоны имен сигналов. Для выбора predefined последовательности используется кнопка , а для предварительного просмотра результата – кнопка .

Существующие форматы predefined последовательности для формирования имени сигнала представлены на рисунке 3.17 и описаны в таблице 3.10.

Если шаблон имени не задан, то глобальные переменные не формируются. При этом необходимо вручную смэппировать переменные (см. 2.8.1.4.2).

7 Нажать кнопку "OK".

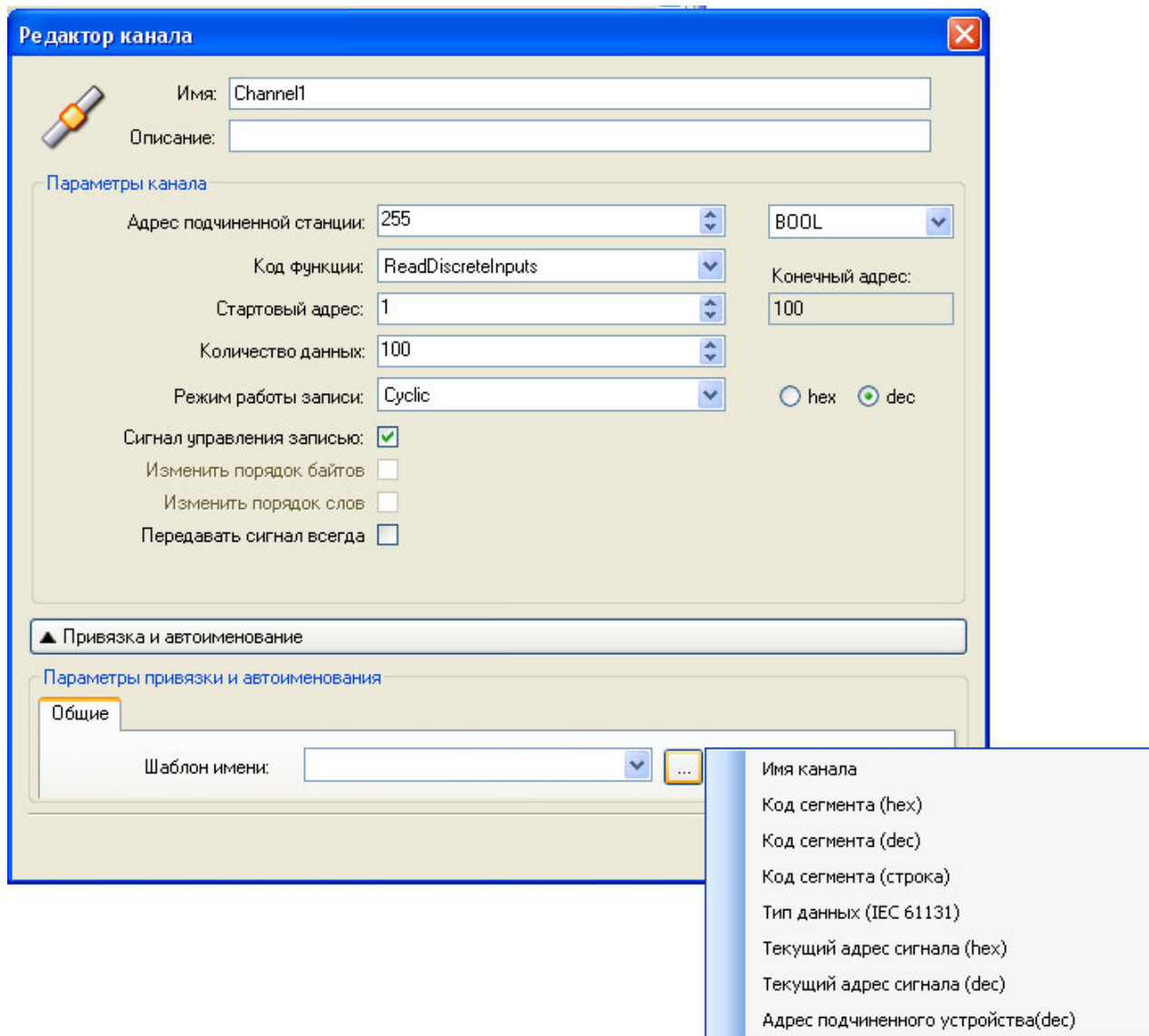


Рисунок 3.17 – Модуль **Server** в **МВТСРМ**. Задание шаблона формирования имени сигнала

Примечание – Минимальное значение *стартового адреса* равно 1.

В результате будет создан канал с указанными атрибутами. Для просмотра сигналов канала следует перейти на закладку *Соотнесение входов/выходов* модуля **Server** – см. рисунок 3.18.

Переменная	Со...	Канал	Адрес	Тип	Ед...	Описание
		Channel1_Control	%QB0	BYTE		
		Channel2_Control	%QB1	BYTE		
		Channel3_Control	%QB2	BYTE		
		Channel4_Control	%QB3	BYTE		
		Channel1	%IB148	ARRAY [1..100] OF BOOL		
		Channel1_Response	%IB248	BYTE		
		Channel2	%IB249	ARRAY [1..10] OF BOOL		
		Channel2_Response	%IB259	BYTE		
		Channel3	%IW130	ARRAY [1..100] OF WORD		
		Channel3_Response	%IB460	BYTE		
		Channel4	%IW231	ARRAY [1..1] OF WORD		
		Channel4_Response	%IB464	BYTE		
		Connect	%IB136	BYTE		Наличие связи мастера с модулем по соединению 1 (NumberConnection = 1...
		CntConn	%ID35	UDINT		Счетчик установки соединения.
		CntWErr	%ID36	UDINT		Счетчик неудачных попыток записи в подчиненное устройство

Рисунок 3.18 – Модуль **Server**. Отображение созданных сигналов канала

3.3.2.3 Настройка статистических и диагностических параметров и соотнесение сигналов

Модуль **Server** имеет набор диагностических и статистических сигналов, представленных в таблице 3.11.

Таблица 3.11 – Модуль **Server**. Диагностические и статистические сигналы

Имя	Тип	Описание
Диагностические сигналы		
<i>Connect</i>	BYTE	Наличие связи опрашиваемого устройства с подчиненным устройством по соединению <i>I</i> : – 0 – соединение не установлено; – 1 – соединение установлено
Статистические сигналы		
<i>CntConn</i>	UDINT	Счетчик установки соединения. Отображает количество удачных попыток установления связи с подчинённым устройством (Server)
<i>CntWErr</i>	UDINT	Счетчик неудачных попыток записи в подчинённое устройство (увеличивается в случае неполучения подтверждения от подчинённого устройства или при получении команды записи от прикладного уровня, но отсутствии соединения с подчинённым устройством)

Настройка сигналов (см. 2.8.1.4.2) выполняется на закладке **Соотнесение входов/выходов** модуля **Server** (рисунок 3.18).

После создания определенных каналов во вкладке **Соотнесение входов/выходов** отображаются сигналы **_Control**, **_Responce**, **_Confirmation**.

_Control – специальный сигнал для управления строками в таблице опроса. С помощью данного сигнала можно включить/исключить строку из таблицы опроса. Возможные значения сигнала:

- 0 – включить строку в таблицу опроса;
- 1 – исключить строку из таблицы опроса;
- 2 – спросить один раз и исключить строку из таблицы опроса.

Сигнал **_Responce** передает код завершения запроса на чтение.

Сигнал **_Confirmation** передает код завершения запроса на запись.

Коды исключения для сигналов *_Response* и *_Confirmation* представлены в таблице 3.7.

3.3.3 Рекомендации по работе с модулем МВТСРМ

Бывают случаи, когда есть необходимость зафиксировать факт получения входного сигнала с тем же значением, что и предшествующее. В обычной ситуации сигналы фиксируются только по изменению их значения, т.е. если протокол передает сигнал два раза с одним значением, то в задачу пользователя данный сигнал не проходит. К примеру, если **МВТСРМ** считывает со **Slave** устройства регистр с одним и тем же значением, то входной сигнал с этим значением будет передан в задачу пользователя один раз, так как значение не изменяется. В таких случаях используется функциональный блок **MapIn**, который позволяет фиксировать факт прохождения сигнала независимо от идентичности значений (дополнительно при использовании ФБ **MapIn** для **МВТСРМ** необходимо открыть «*Редактор каналов*» и поставить галочку напротив строки «*Передавать сигнал всегда*»).

В случаях, если выходному сигналу присваиваются идентичные значения, модуль **МВТСРМ** передаст данные значения (HR, Coils) **Slave** устройству один раз. Для того, чтобы передавать выходные сигналы независимо от идентичности значений, применяется функциональный блок **MapOut**.

Ниже представлен код программы, иллюстрирующий работу функциональных блоков **MapIn** и **MapOut**:

```
PROGRAM PLC_PRG
VAR
(*MapIn*)
HrCmd_100 : INT; // Holding Register по адресу 100
hr100mapin : Elesy.MapIn(ADR(HrCmd_100)); // Объявление ФБ MapIn
myflaghr100: BOOL; (* Флаг, с помощью которого можно зафиксировать принятие сигнала *)
mycnthr100: INT; // Счетчик принятых сигналов (ПС)
hr43_100_mapout : elesy.MapOut( ADR(SICmd_43_100) ); // Объявление ФБ MapOut
cmdSend : BOOL; // Переменная, отвечающая за передачу сигнала
Hr_100 : INT;
END_VAR

(*MapIn*)
hr100mapin(); // Вызов ФБ MapIn
myflaghr100 := hr100mapin.IsUpdate(); // Проверяем, принят ли сигнал
IF myflaghr100 = TRUE THEN // TRUE – сигнал зафиксирован
    mycnthr100 := mycnthr100 + 1; (* Счетчик ПС увеличивается, если сигналы с одинаковыми значениями зафиксированы *)
END_IF
Hr_100 := HrCmd_100; // Считывается принятое значение

(*MapOut*)
hr43_100_mapout(); // Вызов ФБ MapOut
SICmd_43_100[1] := 78; // Присваивание значения элементу массива
```

```

IF cmdSend = TRUE THEN (* Инициация передачи сигнала *)
  hr43_100_mapout.control := 1; (* Установить поведение для однократной передачи данных выхода; 1 –
  сигнал будет передан *)
cmdSend := FALSE;
END_IF

```

Более подробное описание ФБ **MapIn** и **MapOut** представлено в приложении Д.

3.4 Программный модуль Modbus TCP Slave

В данном подразделе представлено описание данных программного модуля с поддержкой протокола *Modbus TCP* в режиме **Slave**, обеспечивающего подключение до четырех опрашиваемых устройств. Символьное обозначение модуля, используемое в сервисной программе – **MBTCPS**.

ВНИМАНИЕ! Параметры данного модуля доступны для просмотра, но пользователю не рекомендуется выполнять настройку данных параметров.

На рисунке 3.19 представлен вид закладки *Редактор параметров* программного модуля **MBTCPS** с информационными параметрами. Данные параметры не доступны для редактирования пользователем.

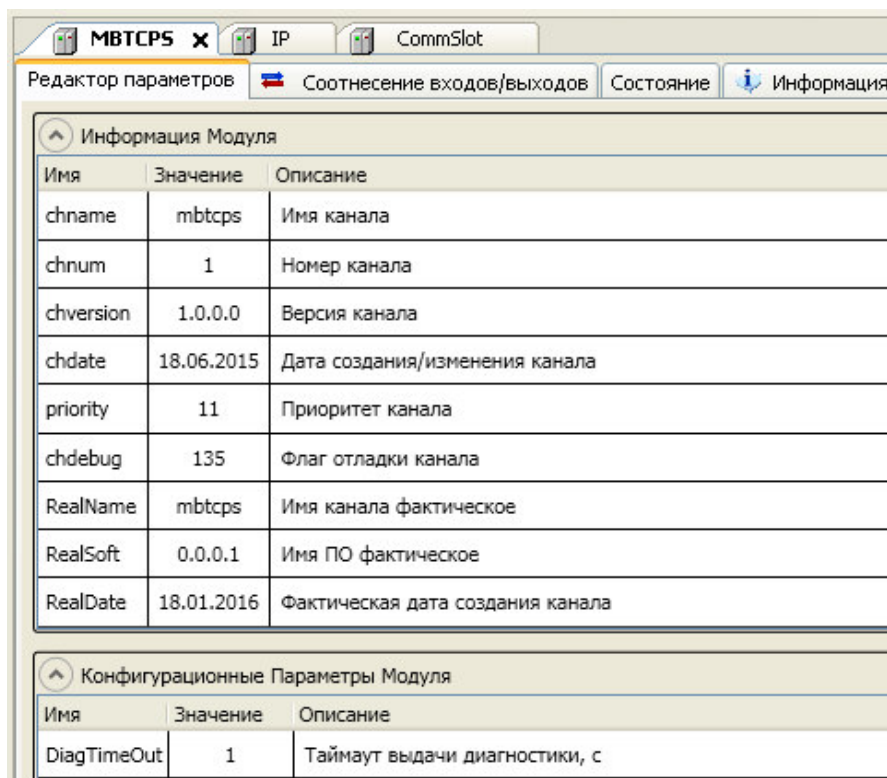


Рисунок 3.19 – Модуль **MBTCPS**. Закладка *Редактор параметров*

На рисунке 3.20 представлен вид закладки *Соотнесение входов/выходов* программного модуля **MBTCPS** с диагностическими сигналами (см. описание сигналов в таблице 3.12).

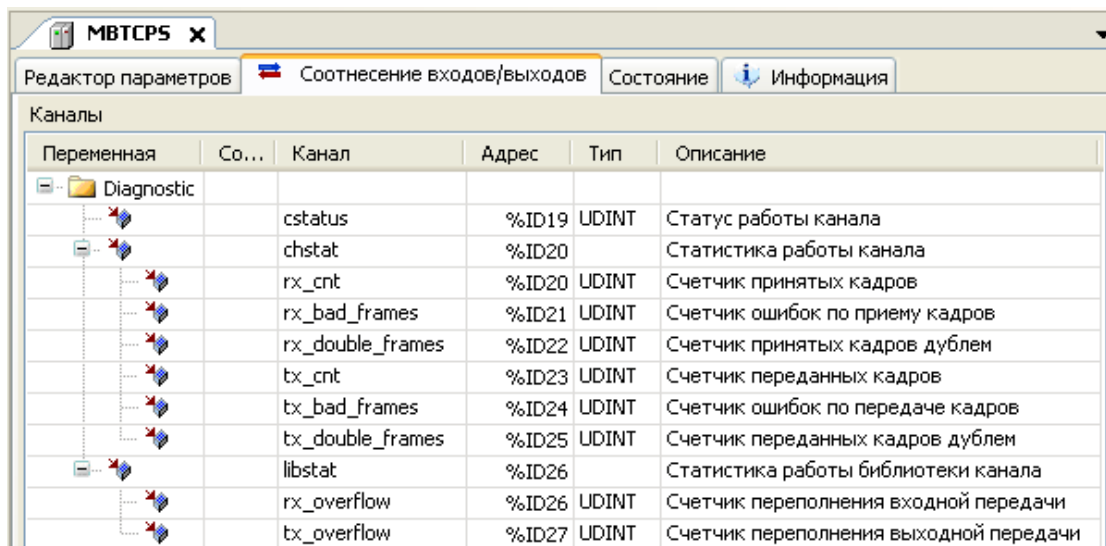


Рисунок 3.20 – Модуль MBTCP S. Закладка Соотнесение входов/выходов

Таблица 3.12 – Модуль Modbus TCP в режиме Slave. Набор диагностических сигналов

Имя	Тип	Описание
<i>cstatus</i>	UDINT	Текущее состояние работы канала. Допустимые значения сигнала: 0 – нормальный режим работы; 1 – ошибка работы канала
<i>chstat</i> – статистика работы модуля (тип EleSyTypes.chstat):		
<i>rx_cnt</i>	UDINT	Счетчик принятых по сетевому соединению (каналу) кадров
<i>rx_bad_frames</i>	UDINT	Счетчик ошибок кадров (нарушение формата, предусмотренного спецификацией) при выполнении приёма
<i>rx_double_frames</i>	UDINT	Счетчик принятых кадров дублем
<i>tx_cnt</i>	UDINT	Счетчик переданных кадров
<i>tx_bad_frames</i>	UDINT	Счетчик ошибок по передачи кадров
<i>tx_double_frames</i>	UDINT	Счетчик переданных кадров дублем
<i>libstat</i> – статистика работы библиотеки канала (тип EleSyTypes.libstat):		
<i>rx_overflow</i>	UDINT	Счетчик переполнения входной передачи
<i>tx_overflow</i>	UDINT	Счетчик переполнения выходной передачи

В данной версии сервисной программы конфигурирование базы сигналов и параметров работы осуществляется в программном модуле **Slave**, находящемся в "дереве" **MBTCP S** (описание программного модуля **Slave** см. в 3.4.1–3.4.3).

Модуль **Slave** предназначен для организации доступа к сигналам контроллера по протоколу *Modbus TCP/IP* с функциональностью сервера (Slave-устройства), позволяющего создать до 16-ти коммуникационных каналов. Символьное обозначение модуля, используемое в сервисной программе – **Slave**.

Программное обеспечение контроллера обеспечивает до четырех подключений опрашивающих устройств одновременно с возможностью принимать/передавать до 1000 сигналов.

Протокол *Modbus* оперирует битовыми значениями (ячейка – **Coil** и вход – **Input**) и 16-ти битными регистрами (регистр – **Holding Register** и входной регистр – **Input Register**).

Поддерживаемые контроллером функции *Modbus* представлены в таблице 3.13.

Таблица 3.13 – Поддерживаемые функции Modbus

Код функции	Описание функции
01 (0x01)	Read Coils (Чтение битовой ячейки)
02 (0x02)	Read Discrete Inputs (Чтение битового входа)
03 (0x03)	Read Holding Registers (Чтение содержимого регистров)
04 (0x04)	Read Input Registers (Чтение содержимого входных регистров)
05 (0x05)	Write Single Coil (Запись битовой ячейки)
06 (0x06)	Write Single Register (Запись в единичный регистр)
15 (0x0F)	Write Multiple Coils (Запись группы битовых ячеек)
16 (0x10)	Write Multiple registers (Запись группы регистров)

В зависимости от результата выполнения операции, подчиненное устройство возвращает нормальный ответ на запрос либо ответ исключения. В нормальном ответе подчиненное устройство возвращает код функции первоначального запроса и запрошенные данные. Ответ исключения формируется, когда подчиненное устройство не может обработать запрос, и содержит в поле данных код исключения с объяснением причины возникшей ошибки. Поддерживаемые в контроллере коды исключений описаны в таблице 3.14.

Таблица 3.14 – Поддерживаемые коды исключения по протоколу Modbus

Код	Исключение	Описание
01	ILLEGAL FUNCTION	Генерация при запросе клиентом функции, которая не поддерживается модулем
02	ILLEGAL DATA ADDRESS	Генерация при обращении к несуществующим Modbus-регистрам
03	ILLEGAL DATA VALUE	Генерация при попытке записи в регистр значения, несоответствующего предусмотренному диапазону значений, или при запросе больше максимального объема данных
06	SLAVE DEVICE BUSY	Генерация при невозможности немедленно обработать запрос. Например, при обработке команды записи, если предыдущая команда не завершилась (команда не передана в ЦП)

Настройка работы серверного модуля **Slave** в системе *CoDeSys* состоит из следующих этапов:

- 1 Настройка конфигурационных параметров (см. 3.4.1).
- 2 Конфигурирование передачи данных по протоколу *Modbus TCP Slave* (см 3.4.2).
- 3 Соотнесение сигналов (см. 3.4.3).

3.4.1 Настройка конфигурационных параметров модуля slave

Настройка конфигурационных параметров осуществляется на закладке просмотра и настройки модуля **Slave**. Для выполнения операции следует:

- 1 Открыть закладку просмотра и настройки модуля **Slave**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Перейти на закладку *Редактор параметров* (рисунок 3.21).
- 3 Настроить конфигурационные параметры модуля. Описание параметров представлено в таблице 3.15.

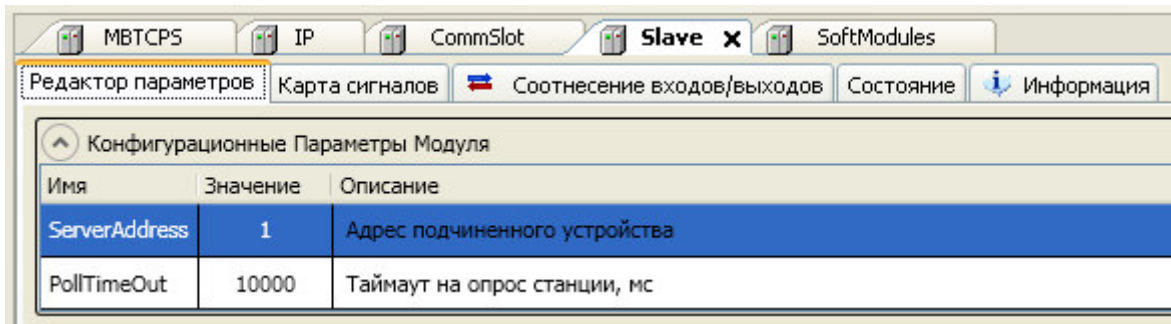


Рисунок 3.21 – Модуль Slave. Закладка Редактор параметров

Таблица 3.15 – Модуль Slave. Конфигурационные параметры

Имя	Значение "по умолчанию"	Описание
<i>ServerAddress</i>	<i>1</i>	Адрес подчиненного устройства. Если подчиненное устройство получило запрос с адресом, отличным от адреса, заданного в конфигурационных параметрах, то данный запрос игнорируется. Адрес "255" является "широковещательным", и при его установке все запросы, полученные через TCP/IP соединение с любым адресом, считаются корректными
<i>PollTimeOut</i>	<i>10000</i>	Тайм-аут на опрос станции (мс). Время, в течение которого ожидается хотя бы один запрос от Master станции. Если запрос не будет получен, то считается, что опросчик недоступен. Для быстрого перехода на резервный канал без ожидания истечения времени, предусмотренного <i>RFC793</i> , осуществляется принудительное закрытие TCP/IP соединения

3.4.2 Конфигурирование передачи данных по *Modbus TCP Slave*

3.4.2.1 Понятие коммуникационного канала

Формирование сигналов для чтения/записи данных по протоколу *Modbus TCP Slave* осуществляется через создание коммуникационного канала, описывающего один непрерывный блок данных. Каналы могут быть логически сгруппированы в секции.

Для работы с каналами и секциями используются следующие операции:

- 1 Создание канала (см. 3.4.2.2).
- 2 Редактирование (имени секции или атрибутов канала, в зависимости от контекста).
- 3 Копирование (для вставки в такой же модуль в конфигурации).
- 4 Удаление.

Для выполнения операции следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Выбрать закладку *Карта сигналов*.
- 3 Вызвать контекстное меню и выбрать соответствующую команду (рисунок 3.22).

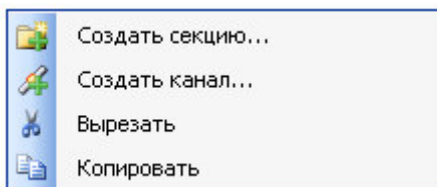


Рисунок 3.22 – Модуль Slave. Общий вид контекстного меню

Канал имеет следующие атрибуты (рисунок 3.23):

- **Имя** – имя секции/канала. Задаёт условное наименование блока данных;
- **Стартовый адрес** – начальный адрес блока данных, расположенных в непрерывной области адресов. Для заданного адреса выбирается формат представления адреса – *hex* или *dec*;
- **Количество данных** – количество данных в блоке;

Имя	Стартовый адрес	Количество дан...	Сегмент	Тип данных	Тип канала	Изменять порядок байт	Изменять порядок слов	Тип исключения при нецелостности	Описание
Section1									
Channel1	4	1	Discretes Input	BIT	Выходной	False	False	None	
Channel2	1	5	Coils	BIT	Выходной	False	False	None	

Рисунок 3.23 – Модуль Slave. Атрибуты коммуникационного канала

- **Сегмент** – типом сегмента определяется область хранения данных. Для заданного сегмента выбирается тип данных (таблица 3.16);

Таблица 3.16 – Допустимые типы данных в зависимости от типов сегментов

Тип данных	Discrete Inputs, Coils	Input Registers, Holding Registers
BOOL	+	+
BYTE	+	+
SINT	+	+
USINT	+	+
WORD	+	+
UINT	+	+
INT	+	+
DWORD	+	+
DINT	+	+
UDINT	+	+
LWORD	-	+
LINT	-	+
REAL	-	+
LREAL	-	+

- **Тип данных;**
- **Тип канала** – определяет доступ к данным со стороны опросчика:
 - входной канал – используется для записи данных в подчиненное устройство. (Сегменты **Coils** и **HoldingRegisters**);
 - выходной канал – используется для чтения данных из подчиненного устройства. (Сегменты **DiscretesInputs**, **InputRegisters**, **Coils** и **HoldingRegisters**).

Типы сегментов и принцип размещения данных представлены на рисунке 3.24.

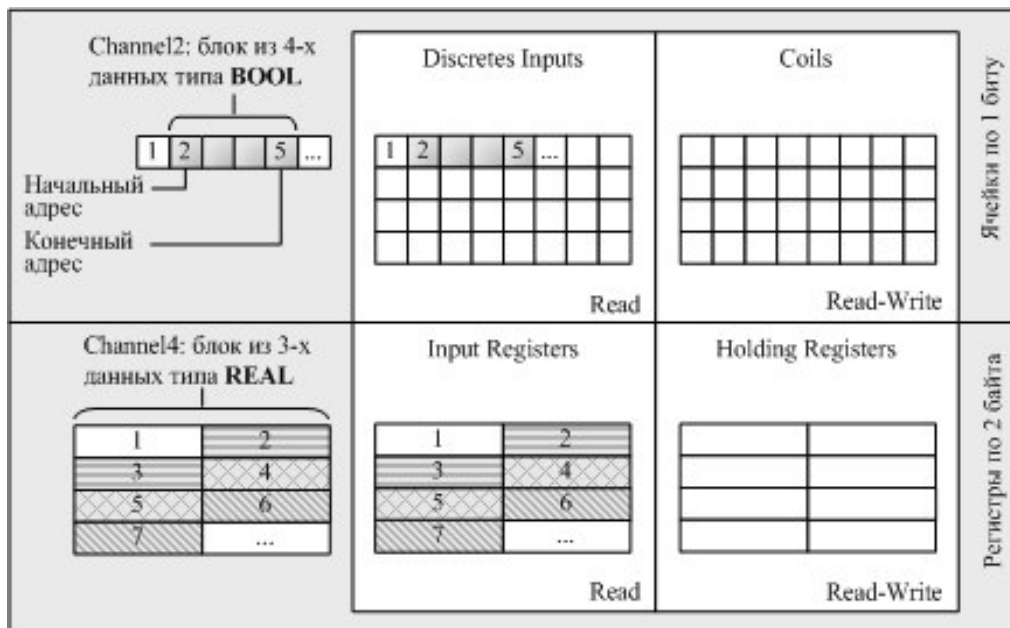


Рисунок 3.24 – Модуль Slave.

Типы сегментов и принцип размещения данных

- **Изменять порядок байтов** – устанавливается для изменения порядка последовательности байтов в значении сигнала с размером от 2 байтов. Данное значение условно делится на слова (размер слова – 2 байта), и в каждом слове выполняется изменение последовательности байтов;
- **Изменять порядок слов** – устанавливается для изменения порядка слов в обратной последовательности в значении сигнала с размером от 2 слов;
- **Тип исключения при нецелостности** – определяет целостное чтение/запись значения объекта, состоящего более чем из одного регистра *Modbus*;
- **Описание.**

При двойном нажатии левой кнопки "мыши" в области отображения значений любого атрибута открывается окно редактирования "Редактор канала", описание которого представлено в 3.4.2.2.

3.4.2.1.1 Назначение переменных и имен сигналам канала

При создании канала предоставляется возможность назначить сигналам новую или существующую переменную, а также возможность задать параметры для автоматического формирования имен структуры (автонаименования). Формирование имен осуществляется на основе шаблонов имен, представляющих собой произвольные комбинации служебных последовательностей (таблица 3.17) и обычных символов в соответствии с разрешенными в IEC 61131-3 символами для имен переменных.

Таблица 3.17 – Символьные последовательности для шаблонов имен

Название	Обозначение	Пример	Описание
Имя канала	%CHANNEL%	Channel3	Заданное имя канала Код сегмента данных в разных форматах
Код сегмента (hex)	%PTYPE_X%	x03	
Код сегмента (dec)	%STYPE_D%	03	
Код сегмента (строка)	%STYPE_S%	HoldingRegisters	
Тип данных (IEC 61131-3)	%TYPE%	SIGNAL_REAL_T	Тип данных сигнала
Текущий адрес сигнала (hex)	%CURADDR_X%	x01	Адрес сигнала в различных форматах
Текущий адрес сигнала (dec)	%CURADDR_D%	1	
Номер сигнала в канале (hex)	%NUM_X%	x0000	Номер сигнала в канале в разных форматах
Номер сигнала в канале (dec)	%NUM_D%	0	

3.4.2.2 Порядок создания канала

Для создания канала следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля **ModBusServer** в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Выбрать закладку *Карта сигналов*.
- 3 Вызвать контекстное меню и выбрать команду *Создать канал...*
- 4 В окне "Редактор канала" (рисунок 3.25) в поле *Имя*: задать имя канала, в поле *Описание*: текстовое описание канала.

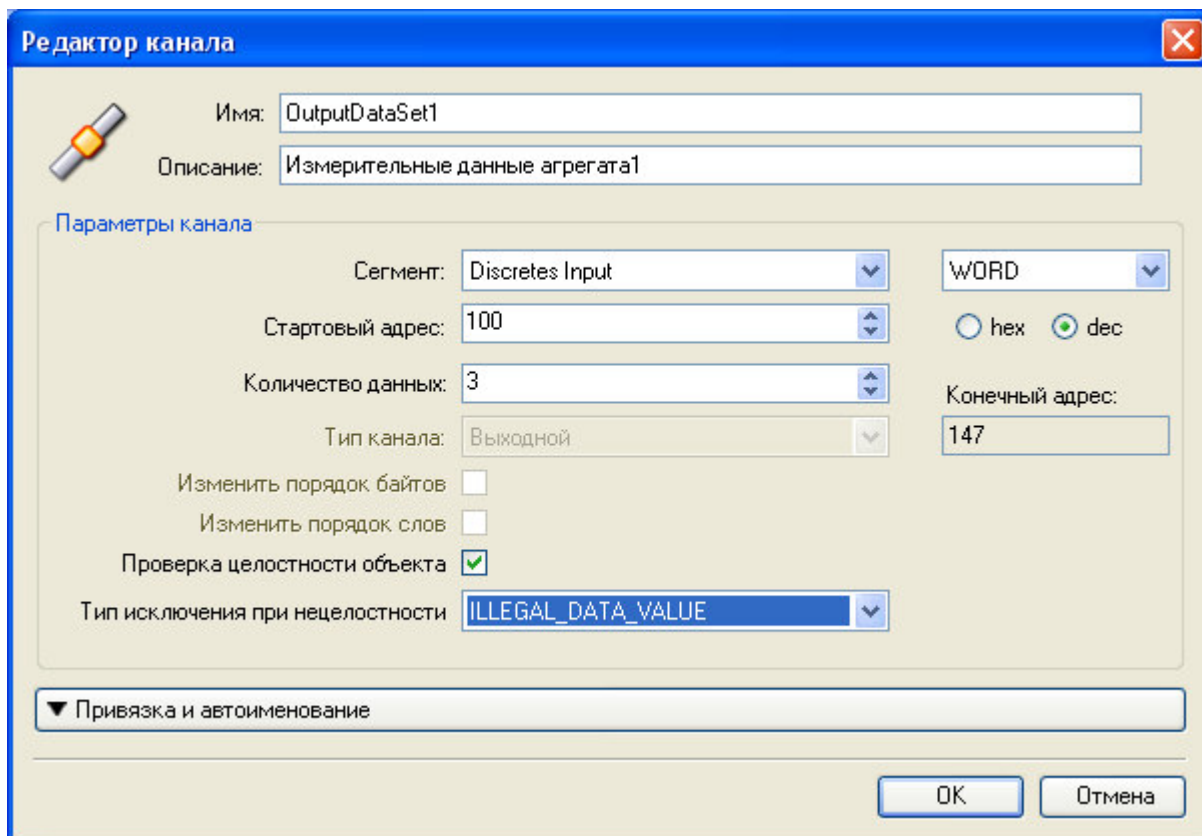


Рисунок 3.25 – Модуль Slave. Создание канала

5 Задать атрибуты канала с помощью элементов группы *Параметры канала*:

- 1) В списке *Сегмент*: выбрать тип сегмента данных:
 - **DiscretesInputs**;
 - **Coils**;
 - **Holding Registers**;
 - **InputRegisters**.

В выпадающем списке справа выбрать тип данных и формат значения с помощью переключателей *hex* и *dec*. Набор допустимых типов данных определяется типом сегмента (таблица 3.16).

- 2) С помощью счетчика *Стартовый адрес*: и *Количество данных*: установить начальный адрес блока данных и количество данных в блоке. При этом в поле *Конечный адрес*: отображается конечный адрес блока.
- 3) Для сегментов **Coils** и **Holding Registers** в списке *Тип канала*: выбрать тип канала.

- 4) Для данных в сегменте **Input Registers** или **Holding Registers** при необходимости установить флаги:
- **Изменить порядок байтов** – устанавливается для изменения порядка последовательности байтов в значении сигнала с размером от 2 байтов. Данное значение условно делится на слова (размер слова – 2 байта), и в каждом слове выполняется изменение последовательности байтов;
 - **Изменить порядок слов** – устанавливается для изменения порядка слов в обратной последовательности в значении сигнала с размером от двух слов.
- 5) Установить флаг **Проверка целостности объекта** для обеспечения целостного чтения/записи значения объекта, состоящего более чем из одного регистра *Modbus*.

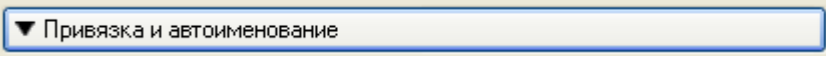


При попытке опрашивающего устройства выполнить чтение/запись части значения сигнала, например – чтение только двух байт из четырехбайтного значения, то подчиненное устройство выдает один из кодов исключения, описанных в таблице 3.18.

Таблица 3.18 – Тип исключения при нецелостности объекта

Тип	Код	Описание
ILLEGAL_FUNCTION	01	Недопустимая функция – полученный в запросе код функции не является допустимым для данного подчиненного устройства
ILLEGAL_DATA_ADDRESS	02	Недопустимый адрес данных – полученный в запросе адрес данных не является допустимым для данного подчиненного устройства
ILLEGAL_DATA_VALUE	03	Недопустимое значение – значение, содержащееся в поле данных запроса, является недопустимым для подчиненного устройства
SLAVE_DEVICE_FAILURE	04	Сбой подчиненного устройства – неисправимая ошибка, возникшая при попытке подчиненного устройства выполнить запрос
ACKNOWLEDGE	05	Квитирование – на обработку запроса требуется много времени. Данный ответ возвращается для предотвращения ошибки тайм-аута опрашивающего устройства
SLAVE_DEVICE_BUSY	06	Подчиненное устройство занято – подчиненное устройство задействовано в обработке продолжительного запроса. Следует ретранслировать запрос позже
MEMORY_PARITY_ERROR	08	Ошибка четности памяти – подчиненное устройство пыталось считать расширенную память, но выявило ошибку четности (паритета)
GATEWAY_PATH_UNAVAILABLE	0A	Путь шлюза недоступен – указывает, что шлюз не смог локализовать путь для выполнения запроса. Данный тип исключения указывает на то, что шлюз не сконфигурирован или перегружен
GATEWAY_TARGET_DEVICE_FAILED	0B	Выбранный шлюз выдал ошибку при ответе – указывает на то, что от конечного устройства не был получен ответ. Данный тип исключения указывает на то, что устройство не находится в сети

- б) В выпадающем списке **Тип исключения при нецелостности** выбрать один из кодов исключений, представленных в таблице 3.18. Данный список доступен при выбранном флаге **Проверка целостности объекта**.

6 Настроить параметры привязки и автонаименования:

- Нажать **кнопку** .
- Задать шаблоны имен сигналов. Для выбора predetermined последовательности используется кнопка , а для предварительного просмотра результата – кнопка .

Существующие форматы predetermined последовательности для формирования имени сигнала представлены на рисунке 3.26 и описаны в таблице 3.17.

Если шаблон имени не задан, то глобальные переменные не формируются. При этом необходимо вручную сопоставить переменные.

7 Нажать кнопку "ОК".

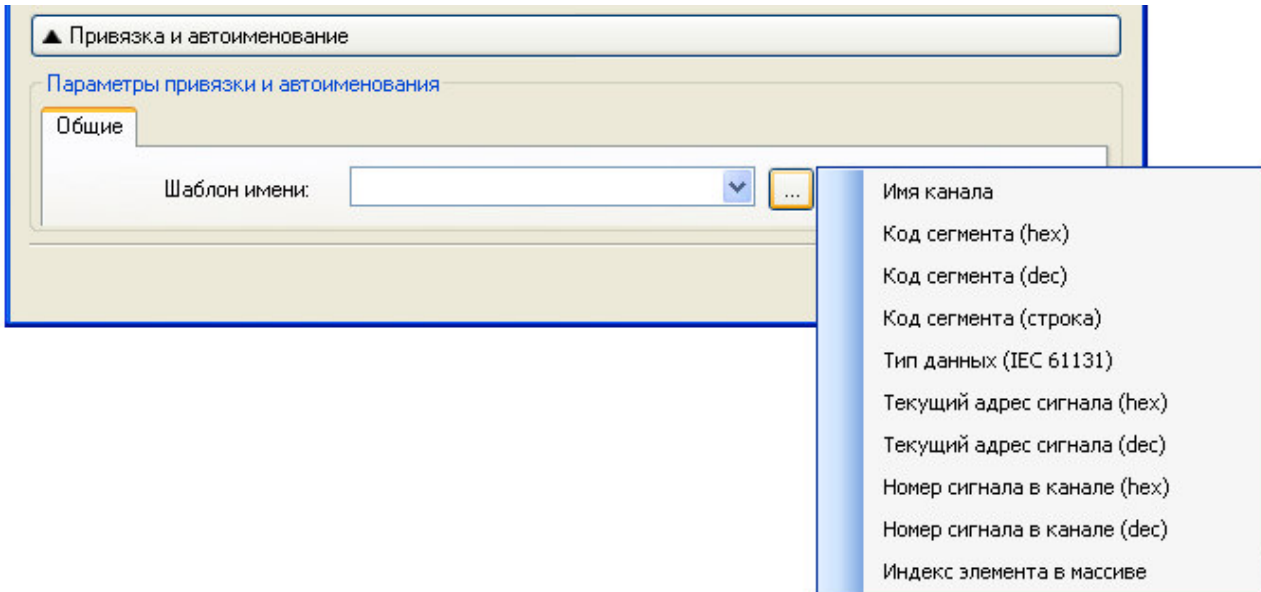


Рисунок 3.26 – Модуль Slave. Задание шаблона формирования имени сигнала

В результате будет создан канал с указанными атрибутами. Для просмотра сигналов канала следует перейти на закладку *Соотнесение входов/выходов* модуля **Slave** – см. рисунок 3.27.

The image shows the 'Slave' module configuration interface with the 'Соотнесение входов/выходов' (Signal Input/Output Mapping) tab selected. The 'Каналы' (Channels) section displays a table of created channels and their attributes.

Переменная	Соот...	Канал	Адрес	Тип	Описание
Application.PLC_P...		Channel1_1	%QB0	Enumer...	1x0005
Application.pIoCo...		Channel2_1	%QB1	Enumer...	0x0002
		Channel2_2	%QB2	Enumer...	0x0003
		Channel2_3	%QB3	Enumer...	0x0004
		Channel2_4	%QB4	Enumer...	0x0005
		Channel2_5	%QB5	Enumer...	0x0006
		Connect_1	%IB...	BYTE	Наличие связи опрашивающего устройства с подчиненным устрой...
		Connect_2	%IB...	BYTE	Наличие связи опрашивающего устройства с подчиненным устрой...
		Connect_3	%IB...	BYTE	Наличие связи опрашивающего устройства с подчиненным устрой...
		Connect_4	%IB...	BYTE	Наличие связи опрашивающего устройства с подчиненным устрой...
		CntConn_1	%ID36	UDINT	Счетчик установки соединения. 1
		CntConn_2	%ID37	UDINT	Счетчик установки соединения. 2
		CntConn_3	%ID38	UDINT	Счетчик установки соединения. 3
		CntConn_4	%ID39	UDINT	Счетчик установки соединения. 4
		CntWErrMst_1	%ID68	UDINT	Счетчик кадров, поступивших от опрашивающего устройства, на к...
		CntWErrMst_2	%ID69	UDINT	Счетчик кадров, поступивших от опрашивающего устройства, на к...
		CntWErrMst_3	%ID70	UDINT	Счетчик кадров, поступивших от опрашивающего устройства, на к...
		CntWErrMst_4	%ID71	UDINT	Счетчик кадров, поступивших от опрашивающего устройства, на к...

Рисунок 3.27 – Модуль Slave. Отображение созданных сигналов канала

3.4.3 Настройка статистических и диагностических параметров и соотнесение сигналов

Модуль **Slave** имеет набор диагностических и статистических сигналов, представленных в таблице 3.19, где $X = 1...4$ (номер соединения).

Таблица 3.19 – Модуль **Slave**. Диагностические и статистические сигналы

Имя	Тип	Описание
Диагностические сигналы		
<i>Connect_X</i>	BYTE	Наличие связи опрашивающего устройства с подчиненным устройством по соединению X: – "0" – соединение не установлено; – "1" – соединение установлено
Статистические сигналы		
<i>CntConn_X</i>	UDINT	Счетчик установки соединения X. Отображает количество удачных попыток установления связи с потребителем
<i>CntWErrMst_X</i>	UDINT	Счетчик кадров, поступивших от опрашивающего устройства, на которые подчиненное устройство ответило исключением

Настройка сигналов выполняется на закладке **Соотнесение входов/выходов** модуля **Slave** (рисунок 3.27).

3.4.4 Рекомендации по работе с модулем MBTCP S

Бывают случаи, когда есть необходимость зафиксировать сам факт получения входного сигнала с тем же значением, что и предшествующее. В обычной ситуации сигналы фиксируются только по изменению их значения, т.е. если в **Slave** записывается сигнал два раза с одним значением, то в задаче пользователя нельзя определить факт записи одного и того же сигнала. В таких случаях используется функциональный блок **MapIn**, который позволяет фиксировать факт записи сигнала независимо от идентичности значений.

В случаях, если выходному сигналу присваиваются идентичные значения, модуль **MBTCP S** передаст данные значения (HR, Coils) **Slave** устройству один раз. Для того, чтобы передавать выходные сигналы независимо от идентичности значений, применяется функциональный блок **MapOut**.

Ниже представлен код программы, иллюстрирующий работу функциональных блоков **MapIn** и **MapOut**:

```
PROGRAM PLC_PRG
VAR
(*MapIn*)
HrCmd_100 : INT; // Holding Register по адресу 100
hr100mapin : Elesy.MapIn(ADR(HrCmd_100)); // Объявление ФБ MapIn
myflaghr100: BOOL; (* Флаг, с помощью которого можно зафиксировать принятие сигнала *)
mycnthr100: INT; // Счетчик принятых сигналов (ПС)
hr43_100_mapout : elesy.MapOut( ADR(SICmd_43_100) ); // Объявление ФБ MapOut
cmdSend : BOOL; // Переменная, отвечающая за передачу сигнала
Hr_100 : INT;
END_VAR
```

```
(*MapIn*)
hr100mapin(); // Вызов ФБ MapIn
myflaghr100 := hr100mapin.IsUpdate(); // Проверяем, принят ли сигнал
IF myflaghr100 = TRUE THEN // TRUE – сигнал зафиксирован
    mycnthr100 := mycnthr100 + 1; (* Счетчик ПС увеличивается, если сигналы с одинаковыми значениями
    зафиксированы *)
END_IF
Hr_100 := HrCmd_100; // Считывается принятое значение
```

```
(*MapOut*)
hr43_100_mapout(); // Вызов ФБ MapOut
SICmd_43_100[1] := 78; // Присваивание значения элементу массива
IF cmdSend = TRUE THEN (* Инициация передачи сигнала *)
    hr43_100_mapout.control := 1; (* Установить поведение для однократной передачи данных выхода; 1 –
    сигнал будет передан *)
cmdSend := FALSE;
END_IF
```

Более подробное описание ФБ **MapIn** и **MapOut** представлено в приложении Д.

3.5 Программный модуль **Modbus RTU Master**

В данном подразделе представлено описание данных программного модуля с поддержкой протокола *Modbus RTU* в режиме **Master**.

Модуль с поддержкой протокола *Modbus* в режиме опросчика (**Master**) обеспечивает опрос до 16 подчиненных устройств по одному каналу связи. Название модуля в конфигурации – **MBMRTU**.

3.5.1 Общий принцип конфигурирования *Modbus RTU Master*

Конфигурирование протокола *Modbus RTU* в режиме *Master* разбивается на следующие шаги:

- 1 Добавление в дерево конфигурации модуля – **MBMRTU**.
- 2 Добавление в дерево конфигурации для модуля **MBMRTU** опрашиваемые подчиненные устройства – **MBMRTU Server (Slave)**.
- 3 Добавление для каждого подчиненные устройства **MBMRTU Server (Slave)** карты опроса.
- 4 Связывание сигналов модулей **MBMRTU** и **MBMRTU Server (Slave)** с пользовательскими данными.
- 5 Настройка параметров физического интерфейса *RS-485* и связывание его с модулем **MBMRTU** (см. 3.14.2).

3.5.2 Настройка модуля **MBMRTU**

Настройка модуля **MBMRTU** выполняется в системе *CoDeSys*, на закладке просмотра и настройки модуля **MBMRTU**. Для выполнения операции следует:

- 1 Открыть закладку просмотра и настройки модуля **MBMRTU**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".

2 Перейти на закладку *Редактор параметров* (рисунок 3.28).

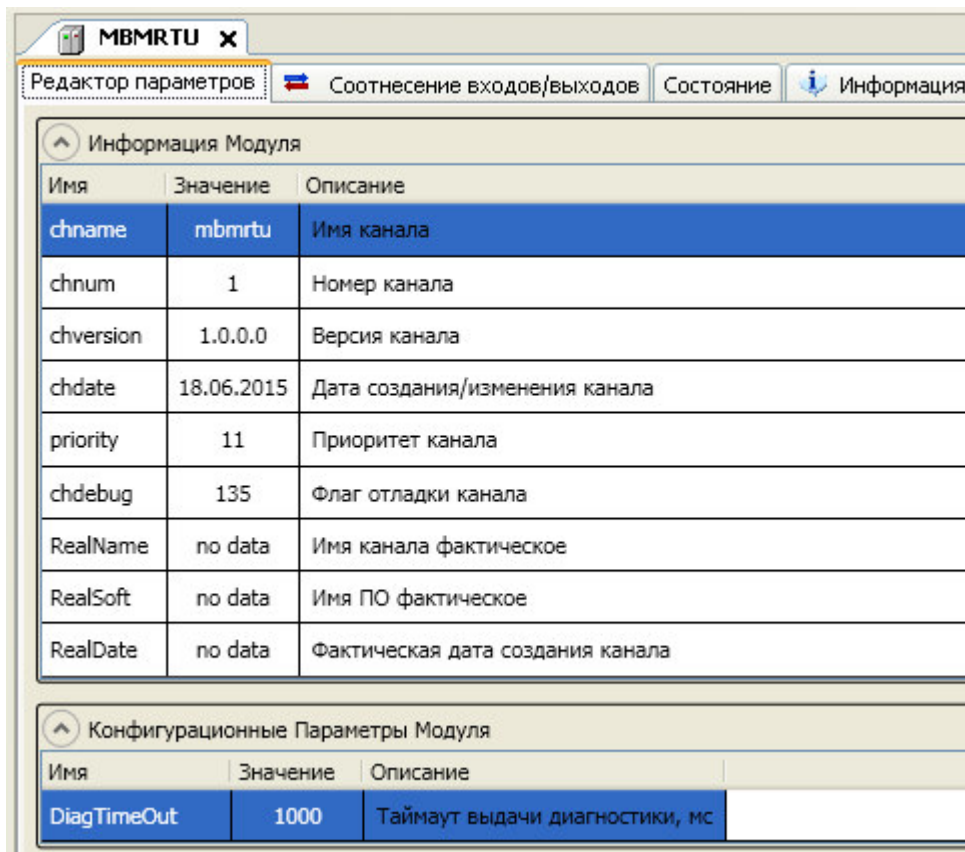


Рисунок 3.28 – Модуль MBMRTU. Закладка *Редактор параметров*

3 Выполнить настройку параметров модуля:

- информационные параметры – общая информация о программном модуле (параметры не доступны для редактирования пользователем). Описание параметров представлено в таблице 3.20.

Таблица 3.20 – Модуль MBMRTU. Информационные данные

Имя	Значение "по умолчанию"	Описание
<i>chname</i>	<i>mbmrtu</i>	Имя канала
<i>chnum</i>	<i>1</i>	Номер канала
<i>chversion</i>	<i>1.0.0.0</i>	Версия канала
<i>chdate</i>	<i>DD.MM.YY</i>	Дата создания, изменения канала в формате <i>день месяц год</i>
<i>priority</i>	<i>11</i>	Приоритет канала
<i>chdebug</i>	<i>135</i>	Флаг отладки канала
<i>RealName</i>	<i>no data</i>	Имя канала фактическое
<i>RealSoft</i>	<i>no data</i>	Имя ПО фактическое
<i>RealDate</i>	<i>no data</i>	Фактическая дата создания канала

- конфигурационные параметры модуля описаны в таблице 3.21.

Таблица 3.21 – Модуль MBMRTU. Конфигурационные параметры

Имя	Значение "по умолчанию"	Описание
<i>DiagTimeOut</i>	<i>1000</i>	Период времени для выдачи диагностической информации о работе модуля в систему

На рисунке 3.29 представлен вид закладки *Соотнесение входов/выходов* программного модуля MBMRTU с диагностическими сигналами. Сигналы диагностики

являются системными (необходимы на этапе отладки программного обеспечения) и в настоящем руководстве не описываются.

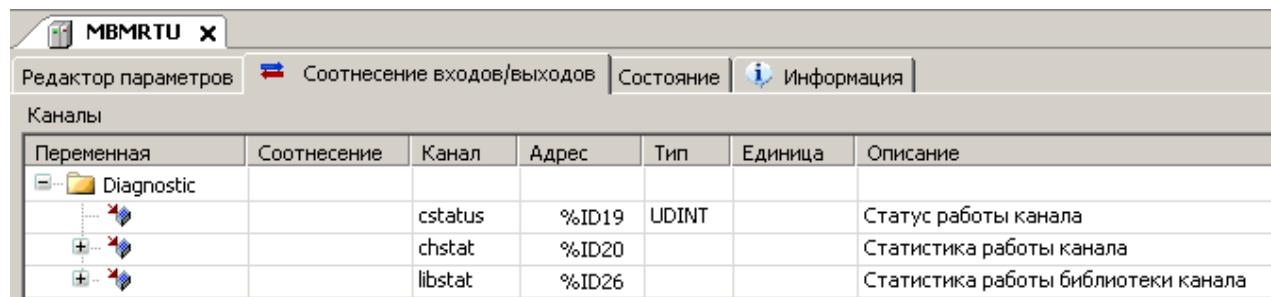


Рисунок 3.29 – Модуль MBMRTU. Закладка *Соотнесение входов/выходов*

3.5.3 Модуль MBMRTU Server (Slave) (для Modbus RTU Master)

Модуль **MBMRTU Server (Slave)** предназначен для организации доступа к сигналам контроллера по протоколу *Modbus RTU*. Символьное обозначение модуля, используемое в сервисной программе – **MBMRTU Server (Slave)**.

Для модуля **MBTCP** можно добавить до 16 устройств, опрашиваемых последовательно, с возможностью принимать/передавать до 1000 сигналов.

Протокол *Modbus* оперирует битовыми значениями (ячейка – **Coil** и вход – **Input**) и 16-ти битными регистрами (регистр – **Holding Register** и входной регистр – **Input Register**).

Поддерживаемые модуля **MBMRTU** функции *Modbus* представлены в таблице 3.22.

Таблица 3.22 – Поддерживаемые функции Modbus

Код функции	Описание функции
01 (0x01)	Read Coils (Чтение битовой ячейки)
02 (0x02)	Read Discrete Inputs (Чтение битового входа)
03 (0x03)	Read Holding Registers (Чтение содержимого регистров)
04 (0x04)	Read Input Registers (Чтение содержимого входных регистров)
05 (0x05)	Write Single Coil (Запись битовой ячейки)
06 (0x06)	Write Single Register (Запись в единичный регистр)
15 (0x0F)	Write Multiple Coils (Запись группы битовых ячеек)
16 (0x10)	Write Multiple registers (Запись группы регистров)

В зависимости от результата выполнения операции, подчиненное устройство возвращает нормальный ответ на запрос либо ответ исключения. В нормальном ответе подчиненное устройство возвращает код функции запроса и запрошенные данные. Ответ исключения формируется, когда подчиненное устройство не может обработать запрос, и содержит в поле данных код исключения с объяснением причины возникшей ошибки.

Настройка работы модуля **MBMRTU Server (Slave)** в системе *CoDeSys* состоит из следующих этапов:

- 1 Настройка конфигурационных параметров (см. 3.4.1).
- 2 Конфигурирование базы сигналов для протокола *Modbus RTU* (см 3.4.2).
- 3 Соотнесение сигналов (см. 3.4.3).

3.5.4 Настройка конфигурационных параметров модуля MBMRTU Server (Slave)

Настройка конфигурационных параметров осуществляется на закладке просмотра и настройки модуля **MBMRTU Server (Slave)**. Для выполнения операции следует:

- 1 Открыть закладку просмотра и настройки модуля **MBMRTU Server (Slave)**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Перейти на закладку **Редактор параметров** (рисунок 3.30).
- 3 Настроить конфигурационные параметры модуля. Описание параметров представлено в таблице 3.23.

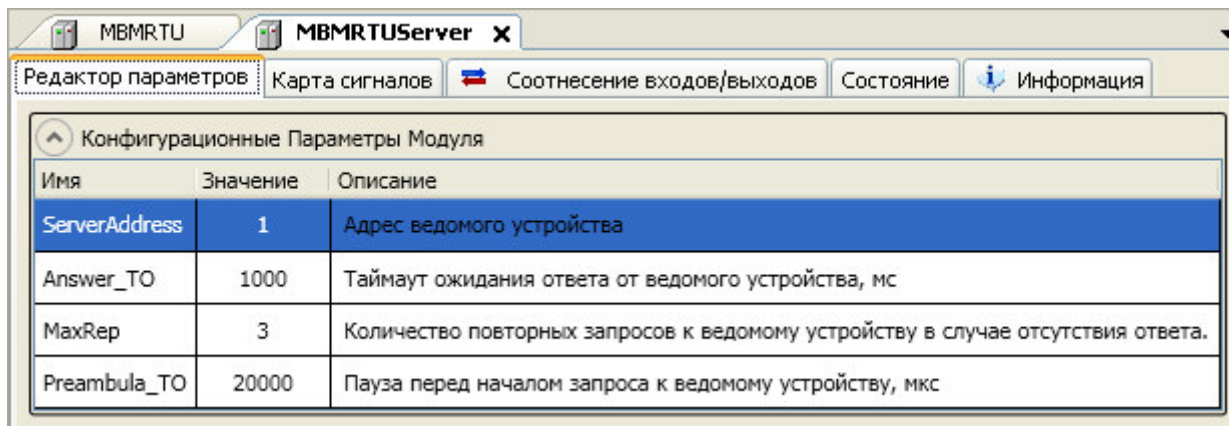


Рисунок 3.30 – Модуль MBMRTU Server (Slave). Закладка Редактор параметров

Таблица 3.23 – Модуль MBMRTU Server (Slave). Конфигурационные параметры

Имя	Значение "по умолчанию"	Описание
<i>ServerAddress</i>	<i>1</i>	Адрес ведомого устройства, для которого создаётся конфигурация
<i>Answer_TO</i>	<i>100</i>	Тайм-аут ответа (задаётся в мс). Время, в течение которого ожидается ответ ведомого устройства. Отсчёт времени начинается после выдачи запроса
<i>MaxRep</i>	<i>3</i>	Количество повторов. Число повторений запроса к устройству в случае истечения тайм-аута на ответ
<i>Preamble_TO</i>	<i>20000</i>	Пауза перед началом запроса к ведомому устройству, мкс

3.5.5 Конфигурирование базы сигналов протокола Modbus RTU для ведомого устройства

3.5.5.1 Понятие коммуникационного канала

Формирование сигналов для чтения/записи данных по протоколу *Modbus RTU* осуществляется через создание коммуникационного канала, описывающего один непрерывный блок данных. Каналы могут быть логически сгруппированы в секции.

Для работы с каналами и секциями используются следующие операции:

- 1 Создание канала/секции.
- 2 Редактирование (имени секции или атрибутов канала в зависимости от контекста).
- 3 Копирование (для вставки в такой же модуль в конфигурации).
- 4 Удаление.

Для выполнения операции следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".

2 Выбрать закладку *Карта сигналов*.

3 Вызвать контекстное меню и выбрать соответствующую команду (рисунок 3.31).

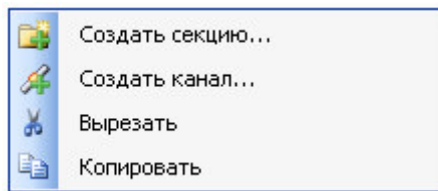


Рисунок 3.31 – Модуль MBMRTU Server (Slave). Общий вид контекстного меню

Канал имеет следующие атрибуты (рисунок 3.32):

- **Имя** – имя секции/канала. Задаёт условное название блока данных;
- **Стартовый адрес** – начальный адрес блока данных, расположенных в непрерывной области адресов. Для заданного адреса выбирается формат представления адреса – *hex* или *dec*;
- **Количество данных** – количество данных в блоке;

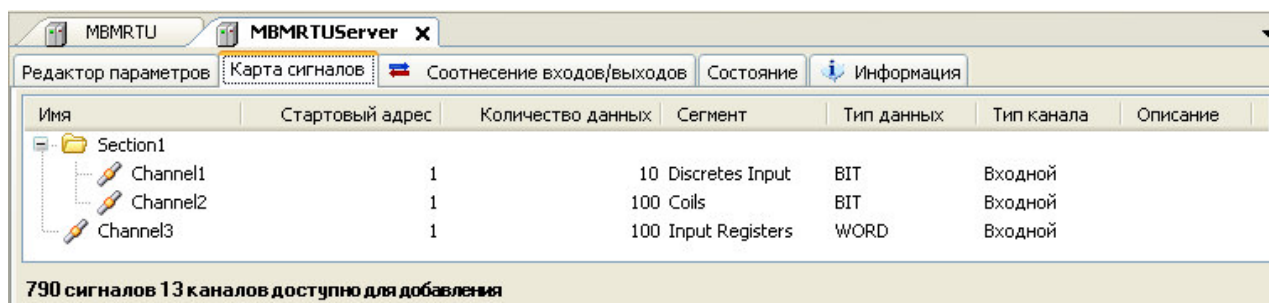


Рисунок 3.32 – Модуль MBMRTU Server (Slave). Атрибуты коммуникационного канала

- **Сегмент** – типом сегмента определяется область хранения данных. Для заданного сегмента выбирается тип данных (таблица 3.24);
- **Тип данных**;

Таблица 3.24 – Допустимые типы данных в зависимости от типов сегментов

Тип данных	Discrete Inputs, Coils	Input Registers, Holding Registers
BOOL	+	-
WORD	-	+
UINT	-	+
INT	-	+
DWORD	-	+
DINT	-	+
UDINT	-	+
REAL	-	+

- **Тип канала** – определяет доступ к данным со стороны опросчика:
 - входной канал – используется для записи данных в подчиненное устройство. (Сегменты **Coils** и **HoldingRegisters**);
 - выходной канал – используется для чтения данных из подчиненного устройства. (Сегменты **DiscretesInputs**, **InputRegisters**, **Coils** и **HoldingRegisters**).

Типы сегментов и принцип размещения, в зависимости от типа, приведены на рисунке 3.24.

При двойном нажатии левой кнопки "мыши" в области отображения значений любого атрибута открывается окно редактирования "**Редактор канала**", описание которого представлено в 3.5.5.3.

3.5.5.2 Назначение переменных и имен сигналам канала

При создании канала предоставляется возможность назначить сигналам новую или существующую переменную, а также возможность задать параметры для автоматического формирования имен структуры (автонаименования). Формирование имен осуществляется на основе шаблонов имен, представляющих собой произвольные комбинации служебных последовательностей (таблица 3.25) и обычных символов в соответствии с разрешенными в IEC 61131-3 символами для имен переменных.

Таблица 3.25 – Символьные последовательности для шаблонов имен

Название	Обозначение	Пример	Описание
Имя канала	%CHANNEL%	Channel3	Заданное имя канала
Код сегмента (hex)	%PTYPE_X%	x03	Код сегмента данных в разных форматах
Код сегмента (dec)	%STYPE_D%	03	
Код сегмента (строка)	%STYPE_S%	HoldingRegisters	
Тип данных (IEC 61131-3)	%TYPE%	SIGNAL_REAL_T	Тип данных сигнала
Текущий адрес сигнала (hex)	%CURADDR_X%	x01	Адрес сигнала в различных форматах
Текущий адрес сигнала (dec)	%CURADDR_D%	1	
Номер сигнала в канале (hex)	%NUM_X%	x0000	Номер сигнала в канале в разных форматах
Номер сигнала в канале (dec)	%NUM_D%	0	

3.5.5.3 Порядок создания канала

Для создания канала следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля **ModBusServer** в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Выбрать закладку *Карта сигналов*.
- 3 Вызвать контекстное меню и выбрать команду *Создать канал...*
- 4 В окне "Редактор канала" (рисунок 3.33) в поле **Имя:** задать имя канала, в поле **Описание:** текстовое описание канала.

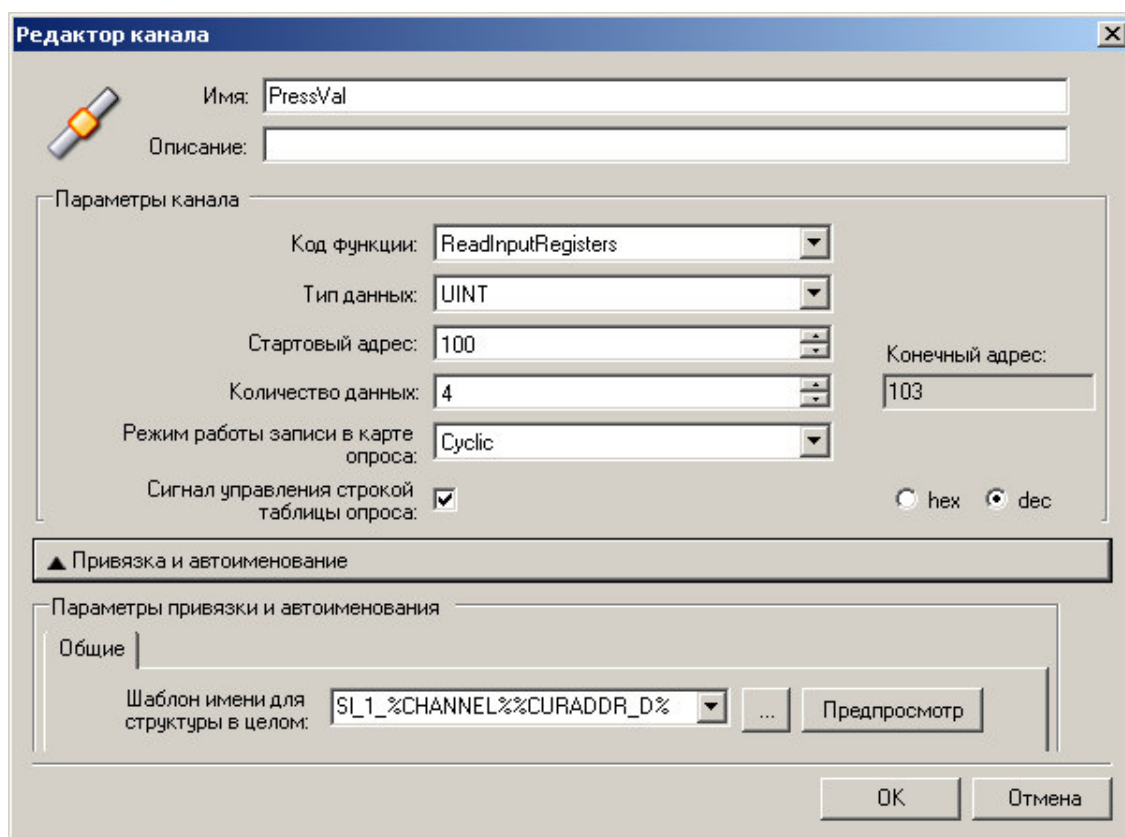


Рисунок 3.33 – Модуль MBMRTU Server (Slave). Создание канала

5 Задать атрибуты канала с помощью элементов группы **Параметры канала**.

- 1) В списке **Код функции**: выбрать соответствующую функцию для выполнения транзакции:
 - ReadDiscretInputs**;
 - ReadCoils**;
 - WriteSingleCoils**;
 - WriteMultipleCoils**;
 - ReadInputRegisters**;
 - ReadHolding Registers**;
 - WriteSingleHolding Registers**;
 - WriteMultipleHolding Registers**.
- 2) В выпадающем списке **Тип данных**: выбрать необходимый тип данных. Набор допустимых типов данных определяется типом сегмента (таблица 3.24).
- 3) С помощью счетчика **Стартовый адрес**: и **Количество данных**: установить начальный адрес блока данных и количество данных в блоке. При этом в поле **Конечный адрес**: отображается конечный адрес блока. Переключателем **hex** и **dec** можно выбрать формат значения для отображения адресов.
- 4) **Тип канала**: выбирается автоматически.

Для каждого канала создаётся отдельная запись в таблице опроса подчинённого устройства. Она определяет параметры транзакции (запроса на чтение или запись) *Modbus*.

Строка таблицы опроса, содержащая запись, может находиться в одном из трёх режимов работы:

- **Выдача каждый цикл поллинга** ("Cyclic");
- **Запрет выдачи** ("DenyRequest");
- **Выдача один раз** ("SingleRequest").

С помощью параметра **Режим работы записи в карте опроса** необходимо установить режим, требуемый для этой строки опроса по старту программы. По умолчанию используется режим *Cyclic* и запрос будет выдаваться в каждом цикле поллинга.

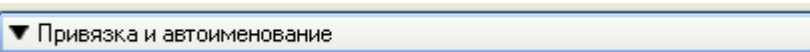
Модуль поддерживает функцию динамического изменения таблицы опроса в процессе работы программы ПЛК. Это может потребоваться, например, для уменьшения времени доставки данных путём исключения из опроса редко изменяющихся сигналов или сигналов, получаемых по другим каналам в схемах с резервированием. Особенно это будет важно и рекомендуется использовать для больших таблиц опроса. Если функция изменения режима выдачи транзакций *Modbus* востребована в программе, то необходимо создать **Сигнал управления строкой таблицы опроса**, установкой соответствующего флага. При этом будет автоматически создан сигнал управления, связанный с этим каналом по имени (**NAME_PollCtrl** во вкладке соотнесения входов/выходов).



В процессе работы программы для изменения режима необходимо присвоить соответствующий код для этого управляющего сигнала:

- "0" – перевод в режим **Выдача каждый цикл поллинга**;
- "1" – перевод в режим **Запрет выдачи**;
- "2" – перевод в режим **Выдача один раз**.

6 Настроить параметры привязки и автонаименования:

- 1) Нажать кнопку



- 2) Задать шаблоны имен сигналов. Для выбора predefined последовательности используется кнопка , а для предварительного просмотра результата – кнопка .

Существующие форматы predefined последовательности для формирования имени сигнала представлены на рисунке 3.34 и описаны в таблице 3.25.

Если шаблон имени не задан, то глобальные переменные не формируются. При этом необходимо вручную смэппировать переменные (см. 2.8.1.4.2).

7 Нажать кнопку "ОК".

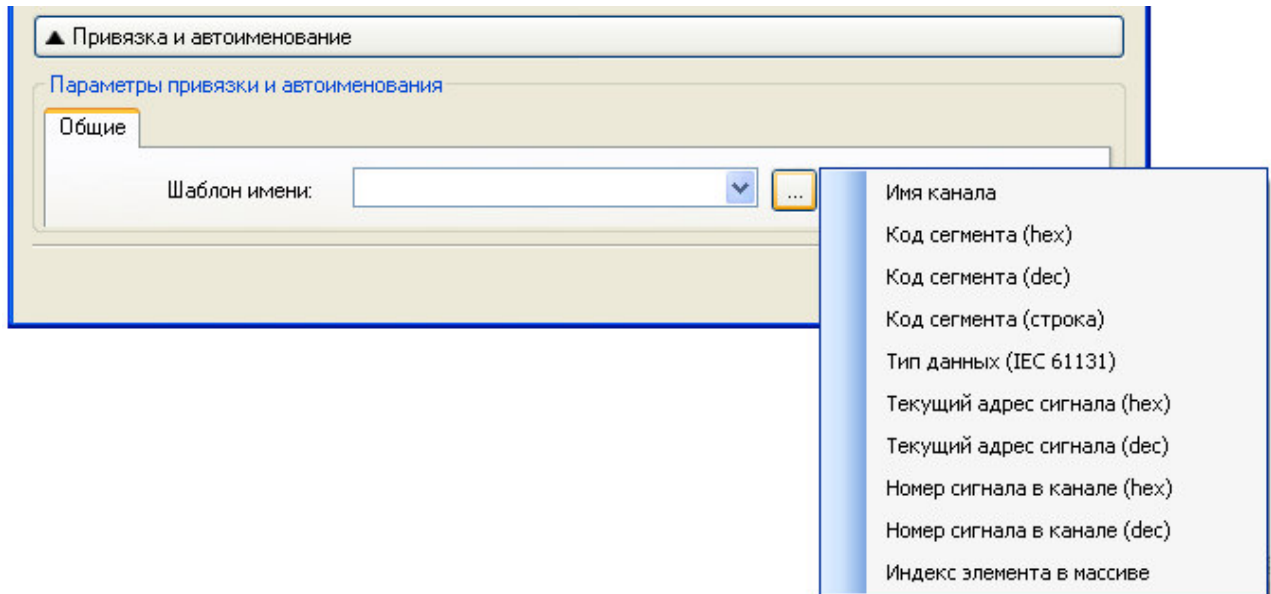


Рисунок 3.34 – Модуль MBMRTU Server (Slave). Задание шаблона формирования имени сигнала

В результате будет создан канал с указанными атрибутами. Для просмотра сигналов канала следует перейти на закладку *Соотнесение входов/выходов* модуля MBMRTU Server (Slave) (см. рисунок 3.35).

Переменная	Со...	Канал	Адрес	Тип	Ед...	Описание
Channels		Channel3_PollCtrl	%QB15	BYTE		
Section1		Channel1_PollCtrl	%QB16	BYTE		
		Channel2_PollCtrl	%QB17	BYTE		
		Channel1	%IB1301	ARRAY [1..10] OF BOOL		
		Channel1_PollResp	%IB1311	BYTE		
		Channel2	%IB1312	ARRAY [1..100] OF BOOL		
		Channel2_PollResp	%IB1412	BYTE		
		Channel3	%IW550	ARRAY [1..100] OF WORD		
		Channel3_PollResp	%IB1300	BYTE		
Diagnostics		Connect	%IB1064	BYTE		Состояние связи (0 - нет опроса; 1 - идет опрос).
Statistics		ConnectCnt	%ID267	UDINT		Счетчик установленных соединения.
		CntTxRead	%ID268	UDINT		Количество выданных запросов на чтение данных.
		CntTxWrite	%ID269	UDINT		Количество выданных запросов на запись данных.
		CntRx	%ID270	UDINT		Количество полученных ответов на запрос
		CntRxExcept	%ID271	UDINT		Количество полученных ответов Exception
		CntCRCErr	%ID272	UDINT		Количество сброшенных кадров по несовпадению CRC
		CntTimeOutErr	%ID273	UDINT		Количество ошибок по таймауту считая с последней удачной установки...
		CntPollCmpl	%ID274	UDINT		Количество завершенных проходов по таблице опроса

Рисунок 3.35 – Модуль MBMRTU Server (Slave). Отображение созданных сигналов канала

3.5.6 Настройка статистических и диагностических параметров и соотнесение сигналов

Модуль **MBMRTU**Server (**Slave**) имеет набор статистических сигналов, представленных на рисунке 3.35. Сигналы диагностики и статистики описаны в таблице 3.26.

Таблица 3.26 – Модуль **MBMRTU**Server (**Slave**). Диагностические и статистические сигналы

Имя	Тип	Описание
Диагностические сигналы		
<i>Connect</i>	BYTE	Наличие связи с опрашиваемым устройством: 0 – соединение не установлено; 1 – соединение установлено
Статистические сигналы		
<i>ConnectCnt</i>	UDINT	Счетчик установлений соединения
<i>CntTxRead</i>	UDINT	Количество выданных запросов на чтение данных
<i>CntTxWrite</i>	UDINT	Количество выданных запросов на запись данных
<i>CntRx</i>	UDINT	Количество полученных ответов на запрос
<i>CntRxExcep</i>	UDINT	Количество полученных ответов Exception
<i>CntCRCErr</i>	UDINT	Количество сброшенных кадров по несовпадению CRC
<i>CntTimeOutErr</i>	UDINT	Количество ошибок по тайм-ауту, считая с последней удачной установки связи с подчиненным устройством
<i>CntPollCmplt</i>	UDINT	Количество завершенных проходов по таблице опроса

После создания определенных каналов во вкладке **Соотнесение входов/выходов** отображаются сигналы *_PollCtrl*, *_PollResp*, *_CmdResp*.

_PollCtrl – специальный сигнал для управления строками в таблице опроса. С помощью данного сигнала можно включить/исключить строку из таблицы опроса. Возможные значения сигнала:

- 0 – включить строку в таблицу опроса;
- 1 – исключить строку из таблицы опроса;
- 2 – спросить один раз и исключить строку из таблицы опроса.

Сигнал *_PollResp* передает код завершения запроса на чтение.

Сигнал *_CmdResp* передает код завершения запроса на запись.

Коды исключения для сигналов *_PollResp* и *_CmdResp* представлены в таблице 3.7.

3.5.7 Рекомендации по работе с модулем MBMRTU

В случаях, если выходному сигналу присваиваются идентичные значения, модуль **MBTSPM** передаст данные значения (HR, Coils) **Slave** устройству один раз. Для того, чтобы передавать выходные сигналы независимо от идентичности значений, применяется функциональный блок **MapOut**.

Ниже представлен код программы, иллюстрирующий работу функционального блока **MapOut**:

```
PROGRAM PLC_PRG
VAR
hr43_100_mapout : elesy.MapOut( ADR(SICmd_43_100) ); // Объявление ФБ MapOut
cmdSend : BOOL; // Переменная, отвечающая за передачу сигнала
END_VAR
```

```

(*MapOut*)
hr43_100_mapout(); // Вызов ФБ MapOut
SICmd_43_100[1] := 78; // Присваивание значения элементу массива
IF cmdSend = TRUE THEN (* Инициация передачи сигнала *)
    hr43_100_mapout.control := 1; (* Установить поведение для однократной передачи данных выхода; 1 –
    сигнал будет передан *)
cmdSend := FALSE;
END_IF

```

Более подробное описание ФБ **MapOut** представлено в приложении Д.

В реальных условиях взаимодействие модуля **MBMRTU** осуществляется с подчиненными устройствами различных производителей, которые имеют свои особенности работы. При работе рекомендуется обращать внимание на сигналы диагностики и статистики, приведенные в таблице 3.26. В случае изменения счетчиков ошибок приема данных, следует увеличить преамбулу перед передачей ответа на запрос со стороны подчиненного устройства до 20 мс.

3.6 Программный модуль Modbus RTU Slave

В настоящем подразделе представлено описание данных программного модуля с поддержкой протокола *Modbus RTU* в режиме *Slave (Server)*.

Наименование модуля в конфигурации – **MBRTUS**.

Поддержка протокола *Modbus RTU Slave* в модуле **MBRTUS** имеет следующие особенности:

- модуль **MBRTUS** поддерживает только адресные транзакции (не поддерживает **Broadcast**-запросы).
- модуль поддерживает следующие типы исключений *Modbus*:
 - ◇ (0x1) ILLEGAL FUNCTION;
 - ◇ (0x2) ILLEGAL DATA ADDRESS;
 - ◇ (0x6) SLAVE DEVICE BUSY;
- поддерживаемые модулем **MBRTUS** функции *Modbus* представлены в таблице 3.27.

Таблица 3.27 – Поддерживаемые функции *Modbus*

Код функции	Описание функции
01 (0x01)	Read Coils (Чтение битовой ячейки)
02 (0x02)	Read Discrete Inputs (Чтение битового входа)
03 (0x03)	Read Holding Registers (Чтение содержимого регистров)
04 (0x04)	Read Input Registers (Чтение содержимого входных регистров)
05 (0x05)	Write Single Coil (Запись битовой ячейки)
06 (0x06)	Write Single Register (Запись в единичный регистр)
15 (0x0F)	Write Multiple Coils (Запись группы битовых ячеек)
16 (0x10)	Write Multiple registers (Запись группы регистров)

3.6.1 Общий принцип конфигурирования *Modbus RTU Slave*

Конфигурирование протокола *Modbus RTU* в режиме *Slave* разбивается на следующие шаги:

- 1 Добавление в дерево конфигурации модуля – **MBRTUS**.
- 2 Добавление в дерево конфигурации для модуля **MBRTUS** – опрашиваемое подчиненное устройство **Server**.
- 3 Настройка параметров модулей **MBRTUS** и **Server**.
- 4 Добавление для подчиненного устройства **Server** карты опроса.
- 5 Связывание сигналов модулей **MBRTUS** и **Server** с пользовательскими данными.
- 6 Настройка параметров физического интерфейса *RS-485* и связывание его с модулем **MBRTUS** (см. 3.14.2).

3.6.2 Настройка модуля **MBRTUS**

Настройка модуля **MBRTUS** выполняется в системе *CoDeSys* на закладке просмотра и настройки модуля **MBRTUS**. Для выполнения операции следует:

- 1 Открыть закладку просмотра и настройки модуля **MBRTUS**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Перейти на закладку *Редактор параметров* (см. рисунок 3.36).

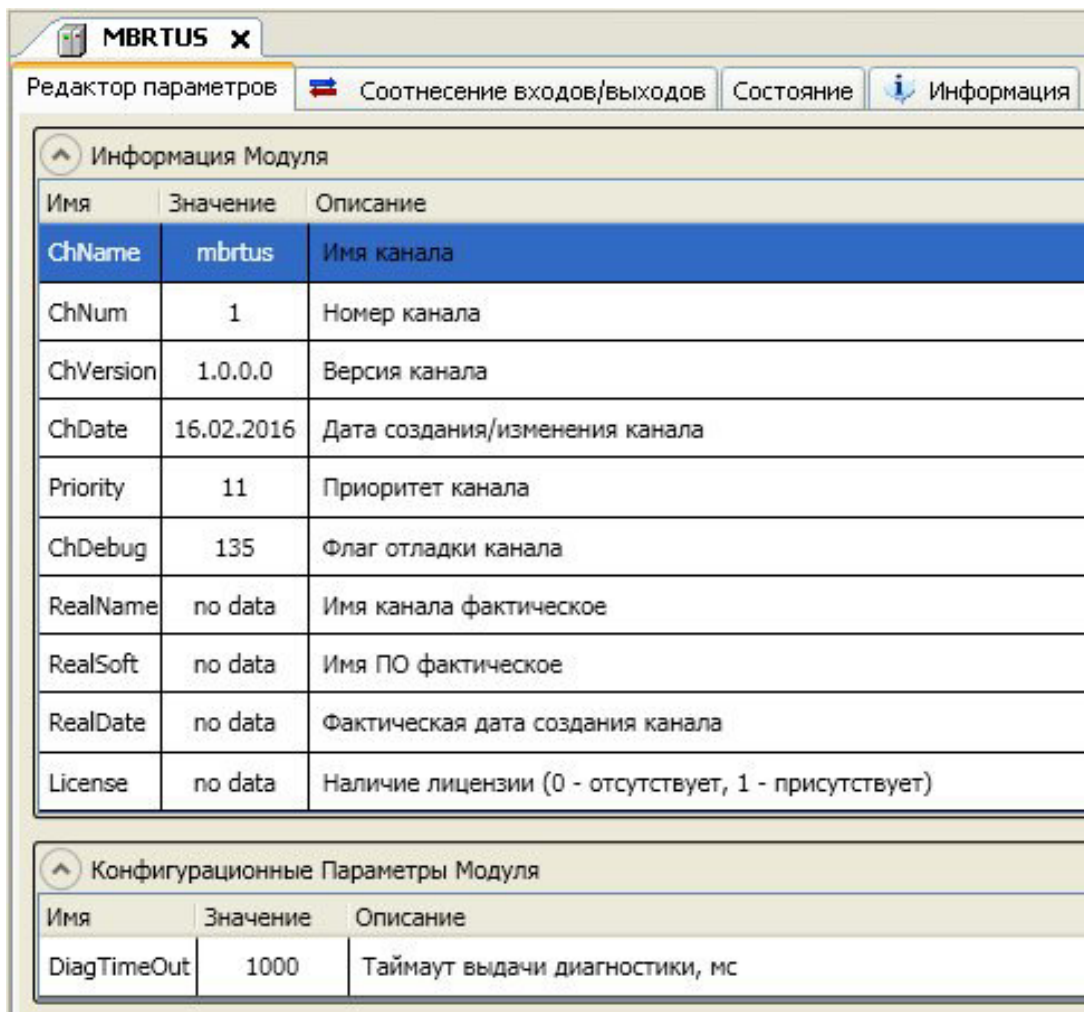


Рисунок 3.36 – Модуль **MBRTUS**. Закладка *Редактор параметров*

3 Выполнить настройку параметров модуля:

- информационные параметры – общая информация о программном модуле (параметры не доступны для редактирования пользователем). Описание параметров представлено в таблице 3.28.

Таблица 3.28 – Модуль MBRTUS . Информационные данные

Имя	Значение "по умолчанию"	Описание
<i>ChName</i>	<i>mbrtus</i>	Имя канала
<i>ChNum</i>	<i>1</i>	Номер канала
<i>ChVersion</i>	<i>1.0.0.0</i>	Версия канала
<i>ChDate</i>	<i>DD.MM.YY</i>	Дата создания/изменения канала в формате <i>день.месяц.год</i>
<i>Priority</i>	<i>11</i>	Приоритет канала
<i>ChDebug</i>	<i>135</i>	Флаг отладки канала
<i>RealName</i>	<i>no data</i>	Имя канала фактическое
<i>RealSoft</i>	<i>no data</i>	Имя ПО фактическое
<i>RealDate</i>	<i>no data</i>	Фактическая дата создания канала
<i>License</i>	<i>no data</i>	Наличие лицензии (0 – отсутствует, 1 – присутствует)

- конфигурационные параметры модуля описаны в таблице 3.29.

Таблица 3.29 – Модуль MBRTUS . Конфигурационные параметры

Имя	Значение "по умолчанию"	Описание
<i>DiagTimeOut</i>	<i>1000</i>	Период времени для выдачи диагностической информации о работе модуля в систему, мс

На рисунке 3.37 представлен вид закладки *Соотнесение входов/выходов* программного модуля **MBRTUS** с диагностическими сигналами. Сигналы диагностики являются системными (необходимы на этапе отладки программного обеспечения) и в настоящем руководстве не описываются.

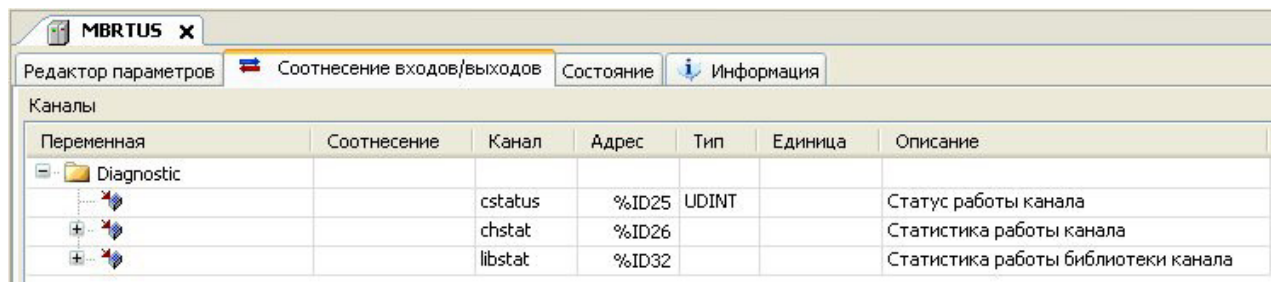


Рисунок 3.37 – Модуль MBRTUS. Закладка *Соотнесение входов/выходов*

3.6.3 Модуль Server

Модуль **Server** предназначен для организации доступа к сигналам контроллера по протоколу *Modbus RTU*. Символьное обозначение модуля, используемое в сервисной программе – **Server**.

Максимальное количество сигналов в карте опроса модуля **Server** – 1000.

Настройка работы модуля **Server** в системе *CoDeSys* состоит из следующих этапов:

- 1 Настройка конфигурационных параметров (см. 3.6.4).

2 Конфигурирование карты опроса по протоколу *Modbus RTU* (см. 3.6.5).

3 Соотнесение сигналов (см. 3.6.5.2).

3.6.4 Настройка конфигурационных параметров модуля **Server**

Настройка конфигурационных параметров осуществляется на закладке просмотра и настройки модуля **Server**. Для выполнения операции следует:

1 Открыть закладку просмотра и настройки модуля **Server**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".

2 Перейти на закладку **Редактор параметров** (рисунок 3.38).

3 Настроить конфигурационные параметры модуля. Описание параметров представлено в таблице 3.30.

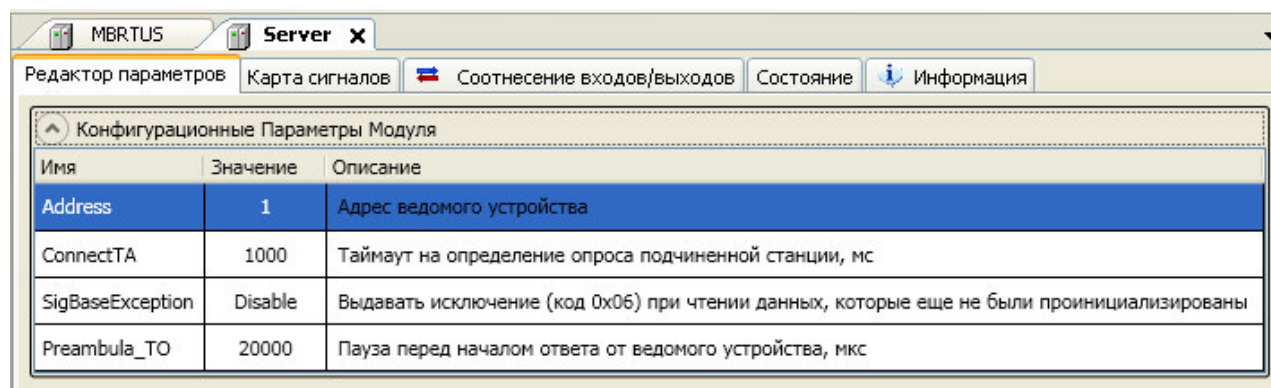


Рисунок 3.38 – Модуль **Server**. Закладка **Редактор параметров**

Таблица 3.30 – Модуль **Server**. Конфигурационные параметры

Имя	Значение "по умолчанию"	Описание
<i>Address</i>	<i>1</i>	Адрес ведомого устройства, для которого создаётся конфигурация. Диапазон изменения параметра – от <i>1</i> до <i>247</i>
<i>ConnectTA</i>	<i>1000</i>	Тайм-аут на определение опроса подчиненной станции, мс. Диапазон изменения параметра – от <i>100</i> до <i>10000</i> мс
<i>SigBaseException</i>	<i>Disable</i>	Выдавать исключение (код 0x06) при запросе данных, которые еще не были проинициализированы (<i>Enable/Disable</i>)
<i>Preamble_TO</i>	<i>20000</i>	Пауза перед началом ответа от ведомого устройства (измеряется в мкс). Диапазон изменения параметра – от <i>0</i> до <i>65535</i> мкс

3.6.5 Конфигурирование карты опроса по протоколу *Modbus RTU*

3.6.5.1 Понятие коммуникационного канала

Формирование сигналов для чтения/записи данных по протоколу *Modbus RTU* осуществляется через создание коммуникационного канала, описывающего один непрерывный блок данных. Каналы могут быть логически сгруппированы в секции.

Для работы с каналами и секциями используются следующие операции:

1 Создание канала/секции.

- 2 Редактирование (имени секции или атрибутов канала в зависимости от контекста).
- 3 Копирование (для вставки в такой же модуль в конфигурации).
- 4 Удаление.

Для выполнения операции следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Выбрать закладку **Карта сигналов**.
- 3 Вызвать контекстное меню и выбрать соответствующую команду (рисунок 3.39).

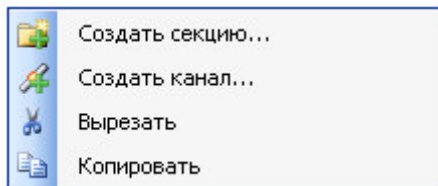


Рисунок 3.39 – Модуль **Server**. Общий вид контекстного меню

Канал имеет следующие атрибуты (см. рисунок 3.40):

- **Имя** – имя секции/канала. Задаёт условное название блока данных;
- **Стартовый адрес** – начальный адрес блока данных, расположенных в непрерывной области адресов;
- **Количество данных** – количество данных в блоке;

Имя	Стартовый ад...	Количество данных	Сегмент	Тип данных	Тип канала	Описание
RHR_1		1	1 Holding Registers	WORD	Выходной	
WHR_1		1	1 Holding Registers	WORD	Входной	

Рисунок 3.40 – Модуль **Server**. Закладка **Карта сигналов**

- **Сегмент** – типом сегмента определяется область хранения данных. Для заданного сегмента выбирается тип данных (см. таблицу 3.31);
- **Тип данных** (см. таблицу 3.31);

Таблица 3.31 – Допустимые типы данных в зависимости от типов сегментов

Тип данных	Discrete Inputs, Coils	Input Registers, Holding Registers
BOOL	+	-
WORD	-	+
UINT	-	+
INT	-	+
DWORD	-	+
DINT	-	+
UDINT	-	+
REAL	-	+

- **Тип канала** – определяет доступ к данным со стороны опросчика:
 - входной канал – используется для записи данных в подчиненное устройство (сегменты **Coils** и **Holding Registers**);
 - выходной канал – используется для чтения данных из подчиненного устройства (сегменты **Discreted Inputs**, **Input Registers**, **Coils** и **Holding Registers**).

Существует ограничение на **Количество данных** в зависимости от **Сегмента** и **Типа канала** (см. таблицу 3.32).

Таблица 3.32 – Ограничение на количество данных в зависимости от сегмента типа канала

Сегмент	Тип канала	Мин. значение	Макс. значение
Discrete Inputs	Выходной	1	500
Coils	Входной	1	500
	Выходной	1	500
Input Registers	Выходной	1	125
Holding Registers	Входной	1	123
	Выходной	1	125

При двойном нажатии левой кнопки "мыши" в области отображения значений любого атрибута открывается окно редактирования "**Редактор канала**", описание которого представлено в 3.6.5.3.

3.6.5.2 Назначение переменных и имен сигналам канала

При создании канала предоставляется возможность назначить сигналам новую или существующую переменную, а также возможность задать параметры для автоматического формирования имен структуры (автонаименования). Формирование имен осуществляется на основе шаблонов имен, представляющих собой произвольные комбинации служебных последовательностей (таблица 3.33) и обычных символов в соответствии с разрешенными в IEC 61131-3 символами для имен переменных.

Таблица 3.33 – Символьные последовательности для шаблонов имен

Название	Обозначение	Пример	Описание
Имя канала	%CHANNEL%	Channel3	Заданное имя канала
Код сегмента (hex)	%PTYPE_X%	x03	Код сегмента данных в разных форматах
Код сегмента (dec)	%STYPE_D%	03	
Код сегмента (строка)	%STYPE_S%	Holding Registers	
Тип данных (IEC 61131)	%TYPE%	REAL	Тип данных сигнала
Текущий адрес сигнала (hex)	%CURADDR_X%	x01	Адрес сигнала в различных форматах
Текущий адрес сигнала (dec)	%CURADDR_D%	1	


3.6.5.3 Порядок создания канала


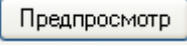
Для создания канала следует:

1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля **Server** в дереве устройств и дважды нажав левую кнопку "мыши".

- 2 Выбрать закладку *Карта сигналов*.
- 3 Вызвать контекстное меню и выбрать команду *Создать канал...*
- 4 В окне "**Редактор канала**" (рисунок 3.41) в поле **Имя:** задать имя канала, в поле **Описание:** текстовое описание канала.

Рисунок 3.41 – Модуль **Server**. Создание канала

- 5 Задать атрибуты канала с помощью элементов группы *Параметры канала*:
 - 1) В списке **Сегмент:** выбрать тип сегмента данных:
 - **DiscretesInputs;**
 - **Coils;**
 - **Holding Registers;**
 - **InputRegisters.**
 - 2) В списке **Тип данных** выбрать тип данных. Набор допустимых типов данных определяется типом сегмента (таблица 3.31).
 - 3) С помощью счетчика **Стартовый адрес:** и **Количество данных:** установить начальный адрес блока данных и количество данных в блоке. При этом в поле **Конечный адрес:** отображается конечный адрес блока.
 - 4) Для сегментов **Coils** и **Holding Registers** в списке **Тип канала:** выбрать тип канала.
- 6 Настроить параметры привязки и автонаименования:
 - 1) Нажать кнопку  **Привязка и автоименование**

- 2) Задать шаблоны имен сигналов. Для выбора predefined последовательности используется кнопка , а для предварительного просмотра результата – кнопка .

Существующие форматы predefined последовательности для формирования имени сигнала представлены на рисунке 3.42 и описаны в таблице 3.33.

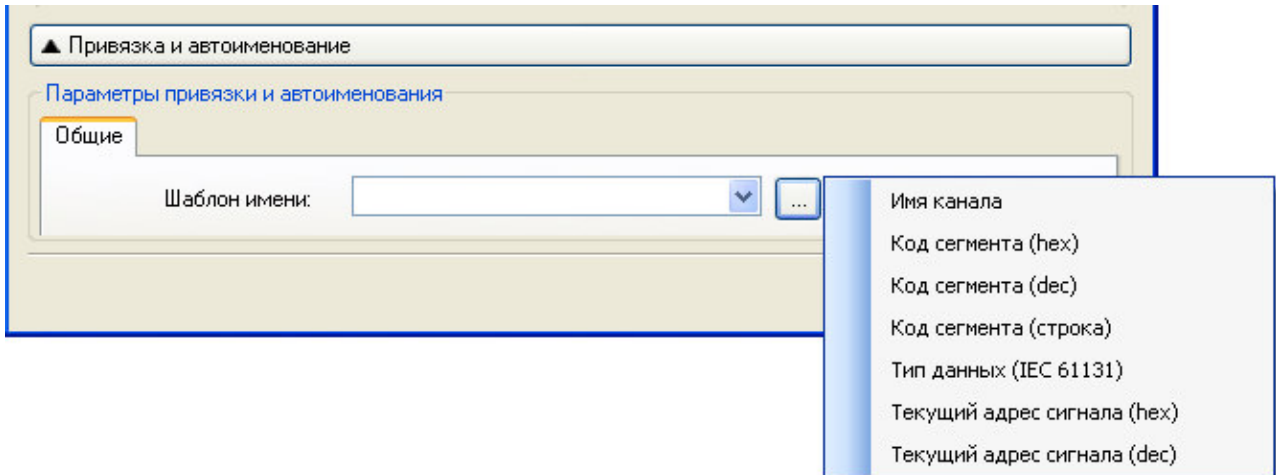


Рисунок 3.42 – Модуль **Server**. Задание шаблона формирования имени сигнала

Если шаблон имени не задан, то глобальные переменные не формируются. При этом необходимо вручную смэппировать переменные (см. 2.8.1.4.2).

- 7 Нажать кнопку "ОК".

В результате будет создан канал с указанными атрибутами. Для просмотра сигналов канала следует перейти на закладку *Соотнесение входов/выходов* модуля **Server** (см. рисунок 3.43).

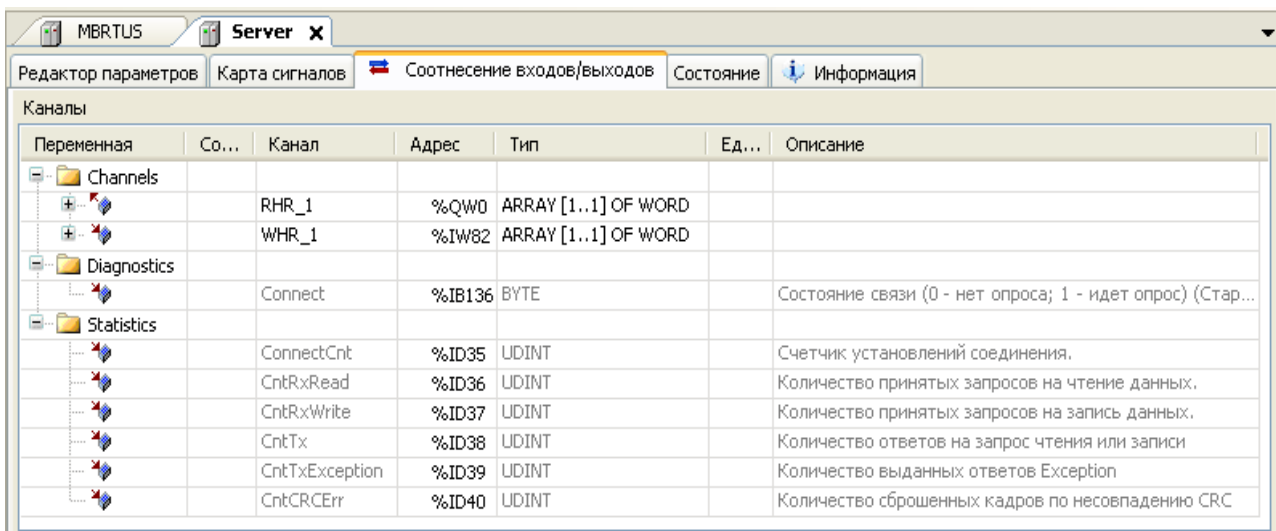


Рисунок 3.43 – Модуль **Server**. Закладка *Соотнесение входов/выходов*

3.6.6 Настройка статистических и диагностических параметров и соотнесение сигналов

Модуль **Server** имеет набор диагностических и статистических сигналов, представленных на рисунке 3.43. Сигналы диагностики и статистики описаны в таблице 3.34.

Таблица 3.34 – Модуль **Server**. Диагностические и статистические сигналы

Имя	Тип	Описание
Диагностические сигналы		
<i>Connect</i>	BYTE	Наличие связи с ведущим устройством: 0 – нет опроса; 1 – идет опрос
Статистические сигналы		
<i>ConnectCnt</i>	UDINT	Счетчик установлений соединения
<i>CntRxRead</i>	UDINT	Количество принятых запросов на чтение данных
<i>CntRxWrite</i>	UDINT	Количество принятых запросов на запись данных
<i>CntTx</i>	UDINT	Количество ответов на запрос чтения или записи
<i>CntTxException</i>	UDINT	Количество выданных ответов Exception
<i>CntCRCErr</i>	UDINT	Количество сброшенных кадров по несовпадению CRC

3.6.7 Рекомендации по работе с модулем **MBRTUS**

Бывают случаи, когда есть необходимость зафиксировать факт получения входного сигнала с тем же значением, что и предшествующее. В обычной ситуации сигналы фиксируются только по изменению их значения, т.е. если протокол передает сигнал два раза с одним значением, то в задачу данный сигнал не проходит. В таких случаях используется функциональный блок **MapIn**, который позволяет фиксировать факт прохождения сигнала независимо от идентичности значений.

В случаях, если выходному сигналу присваиваются идентичные значения, модуль **MBRTUS** передаст данные значения (HR, Coils) **Slave** устройству один раз. Для того, чтобы передавать выходные сигналы независимо от идентичности значений, применяется функциональный блок **MapOut**.

Ниже представлен код программы, иллюстрирующий работу функциональных блоков **MapIn** и **MapOut**:

```

PROGRAM PLC_PRG
VAR
(*MapIn*)
HrCmd_100 : INT; // Holding Register по адресу 100
hr100mapin : Elesy.MapIn(ADR(HrCmd_100)); // Объявление ФБ MapIn
myflaghr100: BOOL; (* Флаг, с помощью которого можно зафиксировать принятие сигнала *)
mycnthr100: INT; // Счетчик принятых сигналов (ПС)
hr43_100_mapout : elesy.MapOut( ADR(SICmd_43_100) ); // Объявление ФБ MapOut
cmdSend : BOOL; // Переменная, отвечающая за передачу сигнала
Hr_100 : INT;
END_VAR

(*MapIn*)
hr100mapin(); // Вызов ФБ MapIn

```

```

myflaghr100 := hr100mapin.IsUpdate(); // Проверяем, принят ли сигнал
IF myflaghr100 = TRUE THEN // TRUE – сигнал зафиксирован
    mycnthr100 := mycnthr100 + 1; (* Счетчик ПС увеличивается, если сигналы с одинаковыми значениями
    зафиксированы *)
END_IF
Hr_100 := HrCmd_100; // Считывается принятое значение

(*MapOut*)
hr43_100_mapout(); // Вызоб ФБ MapOut
SlCmd_43_100[1] := 78; // Присваивание значения элементу массива
IF cmdSend = TRUE THEN (* Инициация передачи сигнала *)
    hr43_100_mapout.control := 1; (* Установить поведение для однократной передачи данных выхода; 1 –
    сигнал будет передан *)
cmdSend := FALSE;
END_IF

```

Более подробное описание ФБ **MapIn** и **MapOut** представлено в приложении Д.

3.7 Программный модуль опроса счетчиков электроэнергии СЭТ4ТМ03М и ПСЧ-4ТМ.05 МК (МД)

В настоящем подразделе представлено описание данных программного модуля **ElMicronMst**. Модуль предназначен для обеспечения опроса счетчиков электроэнергии СЭТ4ТМ03М и ПСЧ-4ТМ.05МК (МД) через интерфейс RS-485.

Модуль **ElMicronMst** обеспечивает опрос до 16 счетчиков указанных моделей в любой комбинации. Наименование модуля в конфигурации – **ElMicronMst**.

3.7.1 Общий принцип конфигурирования модуля опроса счетчиков

Конфигурирование модуля **ElMicronMst** разбивается на следующие шаги:

- 1 Добавление в дерево конфигурации модуля – **ElMicronMst**.
- 2 Добавление в дерево конфигурации для модуля **ElMicronMst** модулей опрашиваемых счетчиков – **SET4ТМ** и **PSH4ТМ**.
- 3 Добавление для каждого модуля (**SET4ТМ** и **PSH4ТМ**) карты опроса.
- 4 Связывание сигналов модулей **ElMicronMst** и **SET4ТМ** (или **PSH4ТМ**) с пользовательскими переменными.
- 5 Настройка параметров физического интерфейса RS-485 и связывание его с модулем **ElMicronMst** (см. 3.14.2).

3.7.2 Настройка модуля ElMicronMst

Настройка модуля **ElMicronMst** выполняется в системе *CoDeSys* на закладке просмотра и настройки модуля **ElMicronMst**. Для выполнения операции следует:

- 1 Открыть закладку просмотра и настройки модуля **ElMicronMst**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".

2 Перейти на закладку **Редактор параметров** (см. рисунок 3.44).

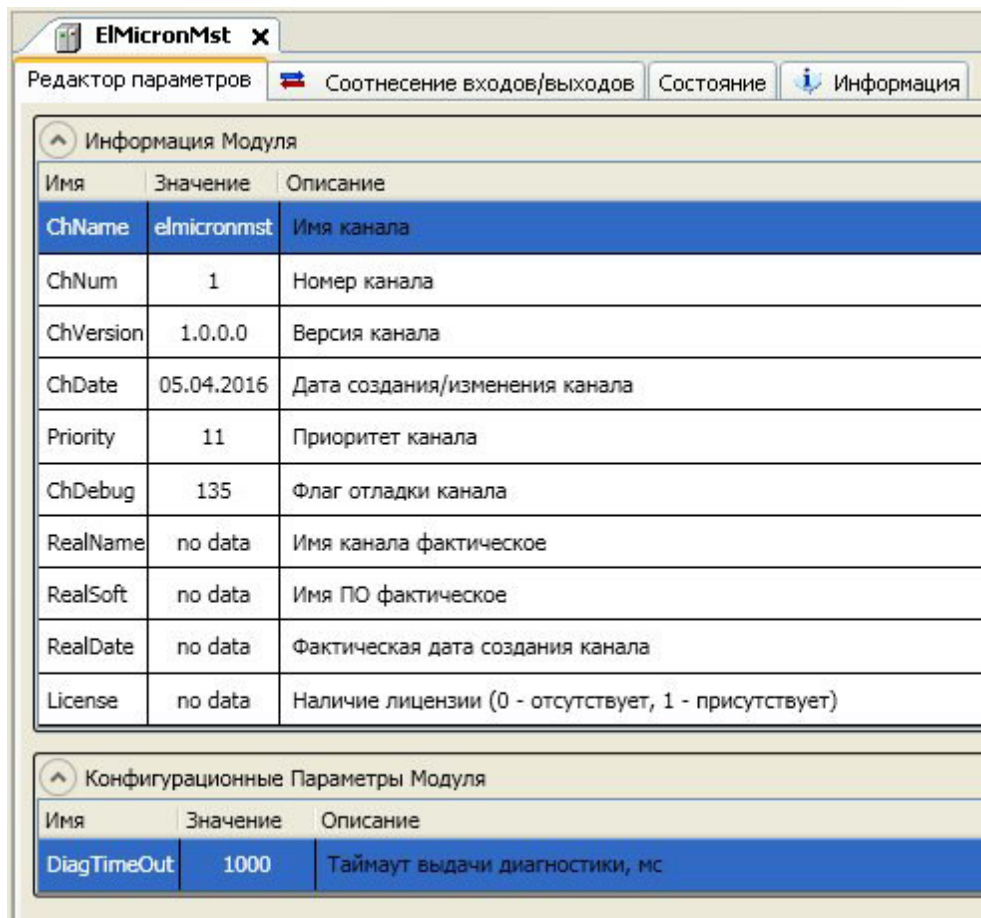


Рисунок 3.44 – Модуль ElMicronMst. Закладка **Редактор параметров**

3 Выполнить настройку параметров модуля:

- информационные параметры – общая информация о программном модуле (параметры не доступны для редактирования пользователем). Описание параметров приведено в таблице 3.35;

Таблица 3.35 – Модуль ElMicronMst . Информационные данные

Имя	Значение "по умолчанию"	Описание
<i>ChName</i>	<i>elmicronmst</i>	Имя канала
<i>ChNum</i>	<i>1</i>	Номер канала
<i>ChVersion</i>	<i>1.0.0.0</i>	Версия канала
<i>ChDate</i>	<i>DD.MM.YY</i>	Дата создания/изменения канала в формате <i>день.месяц.год</i>
<i>Priority</i>	<i>11</i>	Приоритет канала
<i>ChDebug</i>	<i>135</i>	Флаг отладки канала
<i>RealName</i>	<i>no data</i>	Имя канала фактическое
<i>RealSoft</i>	<i>no data</i>	Имя ПО фактическое
<i>RealDate</i>	<i>no data</i>	Фактическая дата создания канала
<i>License</i>	<i>no data</i>	Наличие лицензии (0 – отсутствует, 1 – присутствует)

- конфигурационные параметры модуля описаны в таблице 3.36.

Таблица 3.36 – Модуль ElMicronMst . Конфигурационные параметры

Имя	Значение "по умолчанию"	Описание
<i>DiagTimeOut</i>	<i>1000</i>	Период времени для выдачи диагностической информации о работе модуля в систему, мс

На рисунке 3.45 представлен вид закладки *Соотнесение входов/выходов* программного модуля **ElMicronMst** с диагностическими сигналами. Сигналы диагностики являются системными (необходимы на этапе отладки программного обеспечения) и в настоящем руководстве не описываются.

Переменная	Соотнесение	Канал	Адрес	Тип	Единица	Описание
Diagnostic						
		cstatus	%ID25	UDINT		Статус работы канала
		chstat	%ID26			Статистика работы канала
		libstat	%ID32			Статистика работы библиотеки канала

Рисунок 3.45 – Модуль **ElMicronMst**. Закладка *Соотнесение входов/выходов*

3.7.3 Модули **SET4TM** и **PSH4TM**

Модули **SET4TM** и **PSH4TM** предназначены для организации опроса конкретной модели счетчика – СЭТ или ПСЧ. Символьное обозначение модулей, используемое в сервисной программе – **SET4TM** и **PSH4TM**, соответственно.

Настройка работы модулей **SET4TM** и **PSH4TM** в системе *CoDeSys* состоит из следующих этапов:

- 1 Настройка конфигурационных параметров (см. 3.7.4).
- 2 Конфигурирование базы сигналов модулей **SET4TM** и **PSH4TM** (см. 3.7.5).
- 3 Соотнесение сигналов (см. 3.7.6).

3.7.4 Настройка конфигурационных параметров модулей **SET4TM** и **PSH4TM**

Настройка конфигурационных параметров осуществляется на закладке просмотра и настройки модуля **SET4TM** (**PSH4TM**). Для выполнения операции следует:

1 Открыть закладку просмотра и настройки модуля **SET4TM** (**PSH4TM**), выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".

2 Перейти на закладку *Редактор параметров* (на рисунке 3.46 показана закладка *Редактор параметров* для модуля **SET4TM**).

3 Настроить конфигурационные параметры модуля. Описание параметров модуля **SET4TM** представлено в таблице 3.37 (перечень параметров модуля **PSH4TM** аналогичен перечню параметров модуля **SET4TM**).

Имя	Значение	Описание
ServerAddress	1	Адрес подчиненного устройства
Answer_TO	1000	Таймаут ожидания ответа от ведомого устройства, мс
MaxRep	3	Количество повторных запросов к ведомому устройству в случае отсутствия ответа.
Password	000000	Пароль для работы с данным устройством.

Рисунок 3.46 – Модуль **SET4TM**. Закладка *Редактор параметров*

Таблица 3.37 – Модуль SET4TM. Конфигурационные параметры

Имя	Значение "по умолчанию"	Описание
<i>ServerAddress</i>	1	Адрес ведомого устройства, для которого создаётся конфигурация
<i>Answer_TO</i>	1000	Тайм-аут ответа (задаётся в мс). Время, в течение которого ожидается ответ ведомого устройства. Отсчёт времени начинается после выдачи запроса
<i>MaxRep</i>	3	Количество повторов. Число повторений запроса к устройству в случае истечения тайм-аута на ответ
<i>Password</i>	000000	Пароль для начала работы со счетчиком. Пароль "по умолчанию" – 6 символьных нулей

3.7.5 Конфигурирование базы сигналов модулей SET4TM и PSH4TM

3.7.5.1 Понятие коммуникационного канала

Формирование сигналов для чтения данных счетчика электроэнергии осуществляется через создание коммуникационного канала, описывающего один непрерывный блок данных. Каналы могут быть логически сгруппированы в секции.

Для работы с каналами и секциями используются следующие операции:

- 1 Создание канала/секции.
- 2 Редактирование (имени секции или атрибутов канала в зависимости от контекста).
- 3 Копирование (для вставки в такой же модуль в конфигурации).
- 4 Удаление.

Для выполнения операции следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Выбрать закладку *Карта сигналов*.
- 3 Вызвать контекстное меню и выбрать соответствующую команду (см. рисунок 3.47).

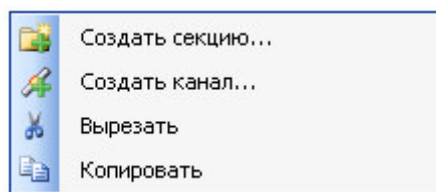


Рисунок 3.47 – Модуль SET4TM (PSH4TM). Общий вид контекстного меню

Атрибуты канала представлены на рисунке 3.48.

Имя	Элемент данных	Получать код подтверждения	Тип данных	Описание
MyCoefPowerCos	CoefPowerCos	False	stCoefPower	
MyCoefPowerSin	CoefPowerSin	False	stCoefPower	
MyCoefPowerTg	CoefPowerTg	False	stCoefPower	

Рисунок 3.48 – Модуль SET4TM (PSH4TM). Закладка *Карта сигналов*

Канал имеет следующие атрибуты:

- *Имя* – имя секции/канала. Задаёт условное название блока данных;

Руководство по эксплуатации

• **Элемент данных** – выбор элемента данных измерений, которые необходимо получить от счетчика электроэнергии. Набор поддерживаемых элементов данных представлен в таблице 3.38.

Таблица 3.38 – Элементы данных

Элемент данных	Направление Входной (I) / Выходной (Q)	Описание элемента данных
<i>Energy</i>	I	Энергия от сброса (нарастающий итог). Тариф 1
<i>PowerP</i>	I	Активная мощность по фазам
<i>PowerQ</i>	I	Реактивная мощность по фазам
<i>PowerS</i>	I	Полная мощность по фазам
<i>CoefPowerSin</i>	I	Коэффициент активной мощности $\sin \varphi$
<i>CoefPowerCos</i>	I	Коэффициент реактивной мощности $\cos \varphi$
<i>CoefPowerTg</i>	I	Коэффициент реактивной мощности $\operatorname{tg} \varphi$
<i>PowerLossP</i>	I	Мощность потерь в линии и силовом трансформаторе активная
<i>PowerLossQ</i>	I	Мощность потерь в линии и силовом трансформаторе реактивная
<i>VoltageMomentary</i>	I	Мгновенное значение напряжения
<i>VoltageAveraged</i>	I	Усредненное значение напряжения
<i>Current</i>	I	Ток
<i>Frequency</i>	I	Частота сети
<i>Temperature</i>	I	Температура внутри счетчика
<i>Diag</i>	I	Флаги состояния измерителя. Битовое поле
<i>State</i>	I	Слово состояния. Битовое поле
<i>HWVersion</i>	I	Вариант исполнения счетчика
<i>CurrentRating</i>	I	Коэффициенты трансформации

Примечание – Направление определяет доступ к данным со стороны модуля: входной канал используется для чтения данных счетчика, выходной – для записи. В данном модуле используются только входные каналы.

• **Тип данных** – типом данных определяется структура элемента данных. Типы данных представлены в таблице 3.39.

Таблица 3.39 – Типы данных

Элемент данных	Имя поля в структуре	Размер, байт	Тип	Описание элемента данных
<i>Energy</i> *			<i>sEnergy</i>	Энергия от сброса (нарастающий итог). Тариф 1 **:
	APlus	4	<i>uint</i>	– A+ – активная энергия прямого направления
	AMinus	4	<i>uint</i>	– A- – активная энергия обратного направления
	RPlus	4	<i>uint</i>	– R+ – реактивная энергия прямого направления
	RMinus	4	<i>uint</i>	– R- – реактивная энергия обратного направления
	R1	4	<i>uint</i>	– R1 – реактивная энергия 1-го квадрант
	R2	4	<i>uint</i>	– R2 – реактивная энергия 2-го

Таблица 3.39 – Типы данных

Элемент данных	Имя поля в структуре	Размер, байт	Тип	Описание элемента данных
				квадрант
	R3	4	<i>uint</i>	– R3 – реактивная энергия 3-го квадрант
	R4	4	<i>uint</i>	– R4 – реактивная энергия 4-го квадранта
PowerP			<i>stPowerP</i>	Активная мощность по фазам **::
	Sum	4	<i>uint</i>	– по сумме фаз
	Phase1	4	<i>uint</i>	– по фазе 1
	Phase2	4	<i>uint</i>	– по фазе 2
	Phase3	4	<i>uint</i>	– по фазе 3
PowerQ			<i>stPowerQ</i>	Реактивная мощность по фазам **::
	Sum	4	<i>uint</i>	– по сумме фаз
	Phase1	4	<i>uint</i>	– по фазе 1
	Phase2	4	<i>uint</i>	– по фазе 2
	Phase3	4	<i>uint</i>	– по фазе 3
PowerS			<i>stPowerS</i>	Полная мощность по фазам **::
	Sum	4	<i>uint</i>	– по сумме фаз
	Phase1	4	<i>uint</i>	– по фазе 1
	Phase2	4	<i>uint</i>	– по фазе 2
	Phase3	4	<i>uint</i>	– по фазе 3
CoefPowerSin			<i>stCoefPowerSin</i>	Коэффициент активной мощности $\sin\varphi$ **::
	Sum	4	<i>real</i>	– по сумме фаз
	Phase1	4	<i>real</i>	– по фазе 1
	Phase2	4	<i>real</i>	– по фазе 2
	Phase3	4	<i>real</i>	– по фазе 3
CoefPowerCos			<i>stCoefPowerCos</i>	Коэффициент реактивной мощности $\cos\varphi$ **::
	Sum	4	<i>real</i>	– по сумме фаз
	Phase1	4	<i>real</i>	– по фазе 1
	Phase2	4	<i>real</i>	– по фазе 2
	Phase3	4	<i>real</i>	– по фазе 3
CoefPowerTg			<i>stCoefPowerTg</i>	Коэффициент реактивной мощности $\operatorname{tg}\varphi$ **::
	Sum	4	<i>real</i>	– по сумме фаз
	Phase1	4	<i>real</i>	– по фазе 1
	Phase2	4	<i>real</i>	– по фазе 2
	Phase3	4	<i>real</i>	– по фазе 3
PowerLossP			<i>stPowerLossP</i>	Мощность потерь в линии электропередачи и силовом трансформаторе активная **::
	Sum	4	<i>real</i>	– по сумме фаз
	Phase1	4	<i>real</i>	– по фазе 1

Таблица 3.39 – Типы данных

Элемент данных	Имя поля в структуре	Размер, байт	Тип	Описание элемента данных
	Phase2	4	real	– по фазе 2
	Phase3	4	real	– по фазе 3
<i>PowerLossQ</i>			<i>stPowerLossQ</i>	Мощность потерь в линии электропередачи и силовом трансформаторе реактивная **:
	Sum	4	real	– по сумме фаз
	Phase1	4	real	– по фазе 1
	Phase2	4	real	– по фазе 2
	Phase3	4	real	– по фазе 3
<i>VoltageMomentary</i>			<i>stVoltageM</i>	Мгновенное значение напряжения, В:
	Phase1	4	real	– по фазе 1
	Phase2	4	real	– по фазе 2
	Phase3	4	real	– по фазе 3
	Interphase1	4	real	– между фазами 1 и 2
	Interphase2	4	real	– между фазами 2 и 3
	Interphase3	4	real	– между фазами 3 и 1
	Line	4	real	– прямой последовательности U1(1)
<i>VoltageAveraged</i>			<i>stVoltageM</i>	Усредненное значение напряжения, В:
	Phase1	4	real	– по фазе 1
	Phase2	4	real	– по фазе 2
	Phase3	4	real	– по фазе 3
	Interphase1	4	real	– между фазами 1 и 2
	Interphase2	4	real	– между фазами 2 и 3
	Interphase3	4	real	– между фазами 3 и 1
	Line	4	real	– прямой последовательности U1(1)
<i>Current</i>			<i>stCurrent</i>	Ток, А:
	Phase1	4	real	– по фазе 1
	Phase2	4	real	– по фазе 2
	Phase3	4	real	– по фазе 3
<i>Frequency</i>			<i>stFrequency</i>	Частота сети, Гц:
	Momentary	4	real	– мгновенное значение частоты
	Averaged	4	real	– усредненное значение частоты
<i>Temperature</i>			<i>stTemperature</i>	Температура внутри счетчика, °С:
	Temperature	4	real	– значение температуры
<i>Diag</i>			<i>stDiag</i>	Флаги состояния измерителя:
	Diag	4	dword	– битовое поле ***
<i>State</i>			<i>stState</i>	Слово состояния:
	StateWord	8	lword	– битовое поле ***
<i>HWVersion</i>			<i>stHWVersion</i>	Вариант исполнения счетчика:
	HWVersion	4	dword	– битовое поле ***
<i>CurrentRating</i>			<i>stCurRating</i>	Коэффициенты трансформации:
	Voltage	2	uint	– по напряжению, Кн
	Current	2	uint	– по току, Кт

Таблица 3.39 – Типы данных

Элемент данных	Имя поля в структуре	Размер, байт	Тип	Описание элемента данных
<p>Примечания</p> <p>1 * Счетчик ПСЧ-4ТМ.05М поддерживает только APlus, AMinus, RPlus, RMinus.</p> <p>2 ** Данные измерений, кроме энергии, выдаются в базовых единицах системы СИ, с учетом введенных в счетчик коэффициентов трансформации по напряжению и току:</p> <ul style="list-style-type: none"> • мощность – Вт, вар, В·А; • напряжение – В; • ток – А; • коэффициент активной мощности (cos φ) – без размерности; • частота сети – Гц; • коэффициенты искажения и несимметрии – %; • температура – °С. <p>Значения активной мощности и коэффициента активной мощности (cos φ) передаются со знаком «+» (прямое направление), если вектор полной мощности находится в 1-м и 4-м квадрантах, и со знаком «-» (обратное направление), если вектор полной мощности находится во 2-м и 3-м квадрантах.</p> <p>Значение реактивной мощности передается со знаком «+» (прямое направление), если вектор полной мощности находится в 1-м и 2-м квадрантах, и со знаком «-» (обратное направление), если вектор полной мощности находится в 3-м и 4-м квадрантах.</p> <p>Остальные параметры передаются со знаком «+».</p> <p>Энергия в регистрах учтенной энергии счетчиков выдается без учета коэффициентов трансформации по напряжению и току в формате внутреннего представления, а именно в числах полупериодов телеметрии и определяется постоянной счетчика.</p> <p>3 *** Расшифровку значения необходимо уточнять в документации на конкретную модель счетчика электроэнергии</p>				

• **Получать код подтверждения** – признак того, содержит ли структура элемента данных поле с кодом подтверждения запроса. На каждый корректный запрос счетчик возвращает либо данные измерений, либо код подтверждения. Код может принимать следующие значения:

- "0" – все нормально;
- "1" – недопустимая команда или параметр;
- "7" – не готов результат измерения по запрашиваемому параметру.

Если пользователь выбрал данную опцию, то структура элемента данных дополняется полями с кодом подтверждения для каждого параметра. Пример для элемента данных **Energy** приведен в таблице 3.40 и на рисунке 3.49, аналогично – для всех остальных элементов данных.

Таблица 3.40 – Структура элемента данных **Energy** с кодом подтверждения

Элемент данных	Имя поля в структуре	Размер, байт	Тип	Описание элемента данных
Energy			<i>stEnergyConfirm</i>	Энергия от сброса (нарастающий итог). Тариф 1:
	APlus	4	<i>uint</i>	– A+ – активная энергия прямого направления
	AMinus	4	<i>uint</i>	– A- – активная энергия обратного направления
	RPlus	4	<i>uint</i>	– R+ – реактивная энергия прямого направления
	RMinus	4	<i>uint</i>	– R- – реактивная энергия обратного направления

Таблица 3.40 – Структура элемента данных *Energy* с кодом подтверждения

Элемент данных	Имя поля в структуре	Размер, байт	Тип	Описание элемента данных
				направления
	R1	4	<i>uint</i>	– R1 – реактивная энергия 1-го квадрант
	R2	4	<i>uint</i>	– R2 – реактивная энергия 2-го квадрант
	R3	4	<i>uint</i>	– R3 – реактивная энергия 3-го квадрант
	R4	4	<i>uint</i>	– R4 – реактивная энергия 4-го квадранта
	APlusConfirm	1	<i>byte</i>	– Код подтверждения A+ – активная энергия прямого направления
	AMinusConfirm	1	<i>byte</i>	– Код подтверждения A- – активная энергия обратного направления
	RPlusConfirm	1	<i>byte</i>	– Код подтверждения R+ – реактивная энергия прямого направления
	RMinusConfirm	1	<i>byte</i>	– Код подтверждения R- – реактивная энергия обратного направления
	R1Confirm	1	<i>byte</i>	– Код подтверждения R1 – реактивная энергия 1-го квадрант
	R2Confirm	1	<i>byte</i>	– Код подтверждения R2 – реактивная энергия 2-го квадрант
	R3Confirm	1	<i>byte</i>	– Код подтверждения R3 – реактивная энергия 3-го квадрант
	R4Confirm	1	<i>byte</i>	– Код подтверждения R4 – реактивная энергия 4-го квадранта

Переменная	Соотношение	Канал	Адрес	Тип	Единица	Описание
Diagnosics						
Statistics						
Channels						
St_2_Energy		MyEnergy	%ID113			
		APlus	%ID113	UDINT		A+ активная энергия прямого направления
		AMinus	%ID114	UDINT		A- активная энергия обратного направления
		RPlus	%ID115	UDINT		R+ реактивная энергия прямого направления
		RMinus	%ID116	UDINT		R- реактивная энергия прямого направления
		R1	%ID117	UDINT		R1 реактивная энергия 1-го квадранта
		R2	%ID118	UDINT		R2 реактивная энергия 2-го квадранта
		R3	%ID119	UDINT		R3 реактивная энергия 3-го квадранта
		R4	%ID120	UDINT		R4 реактивная энергия 4-го квадранта
		APlusConfirm	%IB484	BYTE		подтверждение A+ активной энергии прямого направления
		AMinusConfirm	%IB485	BYTE		подтверждение A- активной энергии обратного направления
		RPlusConfirm	%IB486	BYTE		подтверждение R+ реактивной энергии прямого направления
		RMinusConfirm	%IB487	BYTE		подтверждение R- реактивной энергии прямого направления
		R1Confirm	%IB488	BYTE		подтверждение R1 реактивной энергии 1-го квадранта
		R2Confirm	%IB489	BYTE		подтверждение R2 реактивной энергии 2-го квадранта
		R3Confirm	%IB490	BYTE		подтверждение R3 реактивной энергии 3-го квадранта
		R4Confirm	%IB491	BYTE		подтверждение R4 реактивной энергии 4-го квадранта

Рисунок 3.49 – Модуль SET4TM (PSH4TM). Структура элемента данных *Energy* с кодом подтверждения

При двойном нажатии левой кнопки "мыши" в области отображения значений любого атрибута открывается окно редактирования "Редактор канала", описание которого представлено в 3.7.5.3.

ВНИМАНИЕ! Энергия в регистрах учтенной энергии счетчиков хранится и считывается по интерфейсам связи без учета коэффициентов трансформации по напряжению и току в формате внутреннего представления, а именно в числах полупериодов телеметрии и определяется постоянной счетчика. Постоянные счетчиков в зависимости от типа счетчика и варианта исполнения необходимо брать из документации на счетчики.

Ниже приведен пример коэффициентов (актуальных на момент выпуска данного документа) для различных типов счетчиков.

Тип счетчика	Uном, В	Iном (Imax), А	Постоянная счетчика А, имп/ кВт·ч (имп/ квар·ч)	Разрешающая способность регистров энергии, Вт·ч (вар·ч)
СЭТ-4ТМ.01, СЭТ-4ТМ.02, ПСЧ-4ТМ.05	57,7	5 (7,5)	5000	0,1
	57,7	1 (1,5)	25000	0,02
	120-230	5 (7,5)	1250	0,4
	120-230	1 (1,5)	6250	0,08
ПСЧ-4ТМ.05М	57,7-115	5 (7,5)	5000	0,1
	57,7-115	1 (1,5)	25000	0,02
	120-230	5 (7,5)	1250	0,4
	120-230	1 (1,5)	6250	0,08
СЭТ-1М.01	230	5 (7,5)	5000	0,1
СЭТ-4ТМ.03	57,7	1 (10)	5000	0,1
	120-230	1 (10)	1250	0,4
СЭТ-4ТМ.02М, СЭТ-4ТМ.03М, ПСЧ-4ТМ.05МК, ПСЧ-4ТМ.05МД	57,7-115	5 (10)	5000	0,1
	57,7-115	1 (2)	25000	0,02
	120-230	5 (10)	1250	0,4
	120-230	1 (2)	6250	0,08
СЭБ-1ТМ.01	230	5 (50)	500	1
СЭБ-1ТМ.02(Д)	230	5 (75)	500	1
СЭБ-1ТМ.02М, СЭБ-1ТМ.03	230	5 (80)	500	1
ПСЧ-3ТМ.05	230	5(100)	250	2
ПСЧ-3ТМ.05М ПСЧ-4ТМ.05МК	120-230	5(100)	250	2
ПСЧ-4ТМ.05МД ПСЧ-4ТМ.05МН	120-230	5 (80)	250	2

Перевод числа из формата внутреннего представления в энергию в кВт·ч или квар·ч с учетом коэффициента трансформации производится по формуле:

$$E(\text{кВт} \cdot \text{ч}, \text{квар} \cdot \text{ч}) = \frac{N}{2 \cdot A} \cdot K_n \cdot K_t,$$

где: N – энергия формате внутреннего представления;

A – постоянная счетчика;

K_n – коэффициент трансформации напряжения;

K_t – коэффициент трансформации тока.

3.7.5.2 Назначение переменных и имен сигналам канала

При создании канала предоставляется возможность назначить сигналам новую или существующую переменную, а также возможность задать параметры для автоматического формирования имен структуры (автонаименования). Формирование имен осуществляется на основе шаблонов имен, представляющих собой произвольные комбинации служебных последовательностей (таблица 3.41) и обычных символов в соответствии с разрешенными в ИЕС 61131-3 символами для имен переменных.

Таблица 3.41 – Символьные последовательности для шаблонов имен

Название	Обозначение	Пример	Описание
Имя канала	%CHANNEL%	MyEnergy	Заданное имя канала
Адрес подчиненного устройства (dec)	%SLAVE_ADDR_DEC%	1	Сетевой адрес счетчика электроэнергии

Имя элемента данных	%ELEMENT_NAME%	Energy	Имя элемента данных
---------------------	----------------	--------	---------------------

3.7.5.3 Порядок создания канала

Для создания канала следует:

- 1 Вызвать закладку просмотра и настройки данных модуля, выделив имя модуля в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Выбрать закладку *Карта сигналов*.
- 3 Вызвать контекстное меню и выбрать команду *Создать канал...*
- 4 В окне "Редактор канала" (рисунок 3.50) в поле *Имя*: задать имя канала, в поле *Описание*: текстовое описание канала.

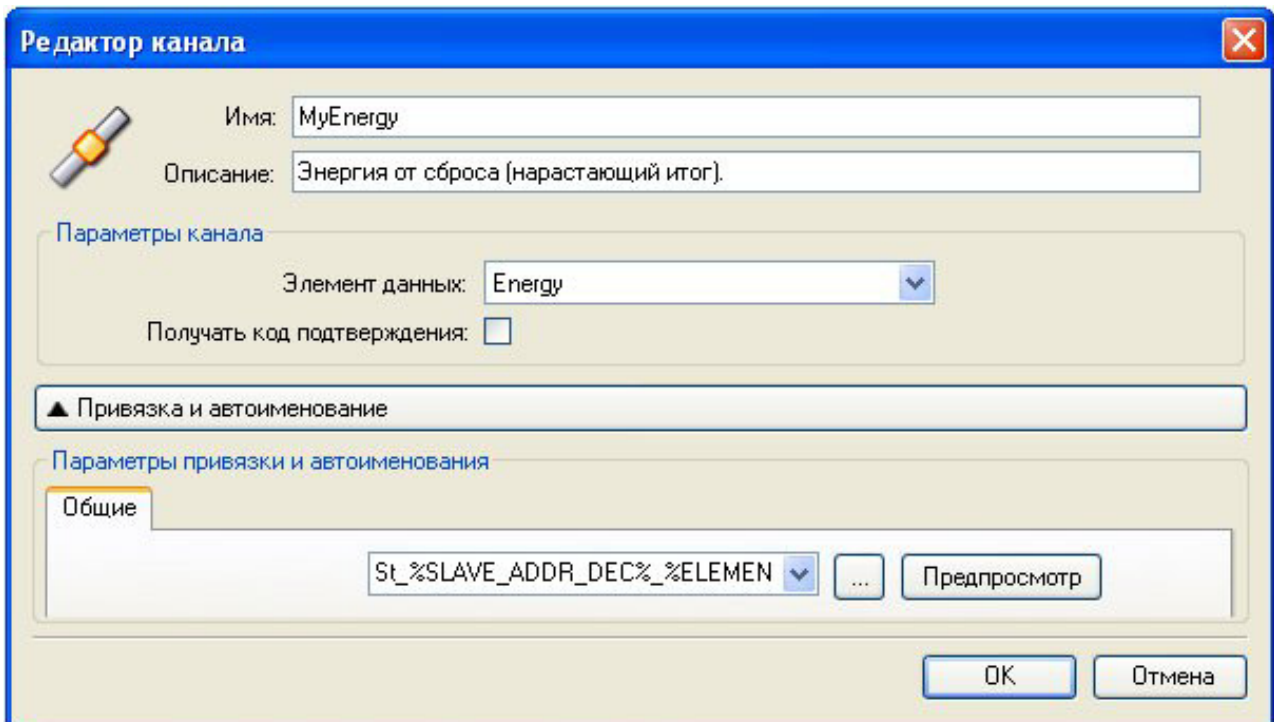


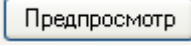


Рисунок 3.50 – Модуль SET4TM (PSH4TM). Создание канала

- 5 Задать атрибуты канала с помощью элементов группы *Параметры канала*.
 - 1) В списке *Элемент данных*: выбрать соответствующий элемент данных (таблица 3.38).
 - 2) Если необходимо получать не только данные измерений, но и код подтверждения на запрос этих данных, то нужно установить флаг *Получать код подтверждения*. Структура элемента данных, содержащего код подтверждения, приведена в таблице 3.40.
- 6 Настроить параметры привязки и автоименования:
 - 1) Нажать кнопку  *Привязка и автоименование*.
 - 2) Задать шаблоны имен сигналов. Для выбора predefined последовательности используется кнопка , а для предварительного просмотра результата – кнопка .

Существующие форматы predefined последовательности для формирования имени сигнала представлены на рисунке 3.51 и описаны в таблице 3.41.

Если шаблон имени не задан, то глобальные переменные не формируются. При этом необходимо вручную смэппировать переменные (см. 2.8.1.4.2).

7 Нажать кнопку "ОК".

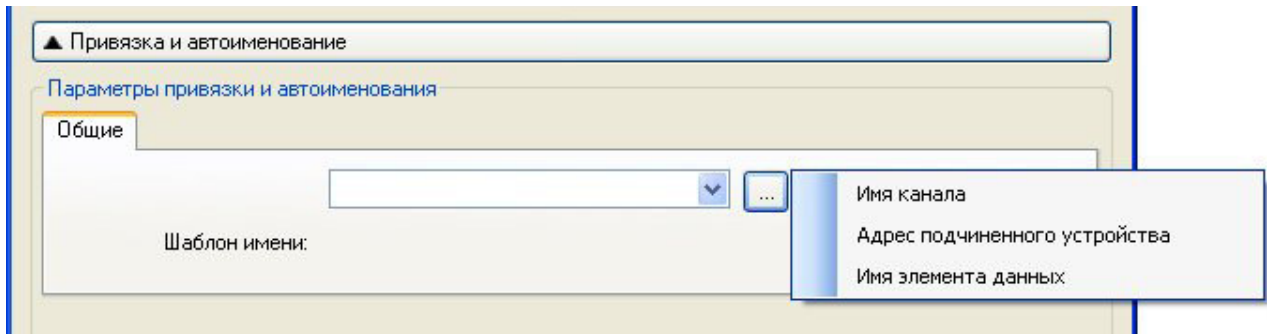


Рисунок 3.51 – Модуль SET4TM (PSH4TM). Задание шаблона формирования имени сигнала

В результате будет создан канал с указанными атрибутами. Для просмотра сигналов канала следует перейти на закладку *ElsyMA.SETPSH Соотнесение входов/выходов* (см. рисунок 3.52).

Переменная	Соотнесение	Канал	Адрес	Тип	Единица	Описание
Diagnosics						
St_1_Connect		Connect	%IB136	BYTE		Состояние связи (0 - нет связи; 1 - есть связь, идет опрос; 2 - есть связь, но задан неверный ID)
Statistics						
St_1_ConnectCnt		ConnectCnt	%ID35	UDINT		Счетчик установлений соединения.
St_1_CntTxRead		CntTxRead	%ID36	UDINT		Количество выданных запросов на чтение данных.
St_1_CntRx		CntRx	%ID37	UDINT		Количество полученных ответов на запрос
St_1_CntCRCErr		CntCRCErr	%ID38	UDINT		Количество сброшенных кадров по несовпадению CRC
St_1_CntTimeOutErr		CntTimeOutErr	%ID39	UDINT		Количество ошибок по таймауту считая с последней удачной установки связи с подчиненным устройством
Channels						
St_1_Energy		MyEnergy	%ID40			
		APlus	%ID40	UDINT		A+ активная энергия прямого направления
		AMinus	%ID41	UDINT		A- активная энергия обратного направления
		RPlus	%ID42	UDINT		R+ реактивная энергия прямого направления
		RMinus	%ID43	UDINT		R- реактивная энергия прямого направления
		R1	%ID44	UDINT		R1 реактивная энергия 1-го квадранта
		R2	%ID45	UDINT		R2 реактивная энергия 2-го квадранта
		R3	%ID46	UDINT		R3 реактивная энергия 3-го квадранта
		R4	%ID47	UDINT		R4 реактивная энергия 4-го квадранта
St_1_PowerP		MyPowerP	%ID48			
St_1_PowerQ		MyPowerQ	%ID52			

Рисунок 3.52 – Модуль SET4TM (PSH4TM). Закладка *ElsyMA.SETPSH Соотнесение входов/выходов*

3.7.6 Настройка статистических и диагностических параметров и соотнесение сигналов

Модули **SET4™** и **PSH4™** имеют набор статистических сигналов, представленных на рисунке 3.52. Сигналы диагностики и статистики описаны в таблице 3.42.

Таблица 3.42 – Модуль **SET4™** (**PSH4™**). Диагностические и статистические сигналы

Имя	Тип	Описание
Диагностические сигналы		
Connect	<i>BYTE</i>	Наличие связи с опрашиваемым устройством: 0 – нет связи; 1 – есть связь, идет опрос; 2 – есть связь, но задан неверный пароль
Статистические сигналы		
ConnectCnt	<i>UDINT</i>	Счетчик установлений связи
CntTxRead	<i>UDINT</i>	Количество выданных запросов на чтение данных
CntRx	<i>UDINT</i>	Количество полученных ответов на запрос
CntCRCErr	<i>UDINT</i>	Количество сброшенных кадров по несовпадению CRC
CntTimeOutErr	<i>UDINT</i>	Количество ошибок по тайм-ауту, считая с последней удачной установки связи с подчиненным устройством

3.8 Работа со счетчиками электроэнергии Энергомера CE301/302/303/304

Для обеспечения информационного взаимодействия ПЛК с измерительным оборудованием Энергомера CE301/302/303/304 в состав конфигурационного пакета включена библиотека **CE30XLibrary**, в которую входят функциональные блоки (ФБ) **CE30X** и **CE30X_v2** (*Device (ELSYMA) → Plc Logic → Application → Менеджер библиотек → ElsyMA_CE30X*, см. 2.8.1.2).

Вызов функций ФБ позволяет осуществить инициализацию физического интерфейса RS-485, используемого для подключения счетчика CE30x, и обеспечить получение измеренных значений, контроля состояния оборудования и чтения конфигурационных параметров.

В этом разделе руководства содержится описание работы с библиотекой ФБ **CE30X** и **CE30X_v2**, а примеры применения ФБ приведены в приложении Е.

По выполняемым функциям ФБ **CE30X_v2** полностью идентичен **CE30X**. Отличия между блоками только в типах сигналов для **CE30X_TR** и **CE30X_IN**. Для ФБ CE30X эти сигналы имеют тип STRING, для которого есть ограничения длины в 256 байт. Для ФБ CE30X_v2 эти сигналы имеют тип STR_BYTE:

```
TYPE STR_BYTE:
UNION
str:STRING;
b:ARRAY[0..1023] OF BYTE;
END_UNION
END_TYPE
```

Таким образом, при использовании ФБ CE30X_v2 снимается ограничение 256 байт для отправляемых транзакций и принимаемых ответов, при этом пользователь при формировании запросов и обработке ответов на транзакции может использовать либо тип STRING, либо массив байт ARRAY[0..1023] OF BYTE.

3.8.1 Конфигурирование ФБ CE303

Применение ФБ не требует специального конфигурирования, т.к. библиотека **CE30XLibrary** уже включена в состав пакета конфигурирования ПЛК. Применение вызовов ФБ в задаче пользователя *CoDeSys* обеспечивает выполнение следующих основных функций:

- инициализация физического интерфейса и переменных ФБ;
- выполнение транзакции (запроса на чтение или запись данных).

Для активации выполнения функции имеется свой управляющий сигнал. Среди входных сигналов ФБ сигналами активации являются:

- **CE30X_INIT** – для инициализации физического интерфейса и переменных ФБ;
- **CE30X_CONTROL** – для выполнения транзакции.

Внимание! Интерфейс RS-485 может использоваться только одним программным модулем, поддерживающим определённый протокол взаимодействия.

При использовании функции ФБ **CE30X** и **CE30X_v2** нельзя использовать интерфейс RS-485 для других коммуникаций (таких как *Modbus RTU* и т.п.). Для этого в конфигурации проекта задачи пользователя ПЛК Элсима для устройства **RS485** в закладке **Редактор соединения** необходимо установить для параметра **Server** значение "None" (рисунок 3.53).

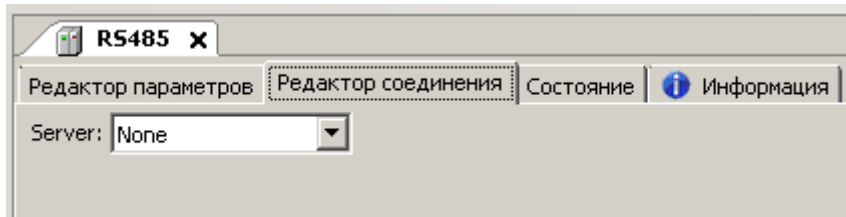


Рисунок 3.53 – Модуль RS485. Закладка *Редактор соединения*

Для использования функций библиотеки **CE30XLibrary** необходимо:

1 В разделе переменных создать экземпляр ФБ. Например:

```
FB_CE30X : ElsyMA_CE30X.CE30X;
```

2 В тексте программы задачи пользователя обеспечить вызов ФБ.

Для программы на языке **ST**, при объявлении экземпляра ФБ, как указано выше, код должен быть следующий:

```
FB_CE30X();
```

3 С помощью сервисных средств для оборудования Энергомера необходимо убедиться в уникальности адресов, подключаемых к одной линии счетчиков, проверить пароли доступа.

4 Выполнить физическое подключение линии согласно правилам подключений линий интерфейса *RS-485*.

Совпадение адресов приведёт к сбоям в работе и может вызвать поломку оборудования.

3.8.2 Инициализация ФБ **CE30X** и **CE30X_v2**

Функция инициализации вызывается один раз в начале выполнения программы с предварительно установленными значениями параметров или параметрами "по умолчанию". Функция инициализации полностью одинакова для **CE30X** и **CE30X_v2**.

Функция инициализации обеспечивает инициализацию переменных для работы ФБ и инициализацию физического интерфейса *RS-485* ПЛК Элсима системными вызовами *CoDeSys* в соответствии со значениями входных переменных и заданного в программе имени порта (имя соответствует физическому устройству *RS-485* в составе ПЛК Элсима).

Набор переменных для выполнения инициализации ФБ приведен в таблице 3.43.

Таблица 3.43 – ФБ CE30X. Входные и выходные переменные для выполнения инициализации

Имя	Тип	Значение "по умолчанию"	Описание
Входные переменные			
<i>ptTR</i>	TIME	<i>TIME#3s0ms</i>	Таймаут ожидания ответа на транзакцию
<i>CE30X_INIT</i>	BYTE	<i>0</i>	Сигнал управления началом выполнения инициализации ФБ: <i>0</i> – функция не активна (признак разрешения работы для других вызовов ФБ); <i>1</i> – активация выполнения инициализации
<i>BoudeRate</i>	BYTE	<i>5</i>	Скорость передачи данных в интерфейсе согласно ГОСТ ИЕС 61107-2011, бит/с: <i>0</i> – 300 *; <i>1</i> – 600 *; <i>2</i> – 1200 *; <i>3</i> – 2400 *; <i>4</i> – 4800; <i>5</i> – 9600
<i>Paritet</i>	BYTE	<i>2</i>	Тип контроля для символа: <i>0</i> – <i>None</i> (нет); <i>1</i> – <i>Odd</i> (нечёт.); <i>2</i> – <i>Even</i> (чёт.)
<i>BitNmb</i>	BYTE	<i>7</i>	Количество бит данных в символе. Допустимые значения – 7, 8
<i>StopBitNmb</i>	BYTE	<i>1</i>	Количество стоп-бит в символе. Допустимые значения – 1, 2
Выходные переменные			
<i>INIT_ErrCode</i>	DINT		Код ошибки. Допустимые значения: <i>0</i> – успешная инициализация порта с заданными параметрами; <> <i>0</i> – ошибка инициализации. Коды ошибок: "-1" – ошибка инициализации СОМ порта; "-2" – ошибка тестирования передачи в СОМ-порт; "-3" – ошибка тестирования приёма из СОМ-порта
Примечание – * В версии 3.5.6.1 и ниже библиотеки CE30XLibrary данное значение не доступно			

3.8.3 Выполнение транзакции ФБ CE30X

Функция выполнения транзакции с измерительным оборудованием вызывается из задачи пользователя периодически.

Функция выполнения транзакции обеспечивает выполнение одиночных или последовательности транзакций (передачи запросов и приёма ответов по интерфейсу) с предварительной обработкой результата для контроля ошибок времени ответа, искажения данных и кода ошибки в ответе.

Внимание! Вызов функции выполнения транзакции до завершения инициализации приведёт к ошибке! Для выполнения следующей транзакции необходимо дождаться окончания уже начатой транзакции! Несоблюдение этого правила приведёт к ошибке выполнения.

Набор переменных для выполнения транзакций ФБ (управление работой, задание типа получаемых данных и контроля правильности выполнения) приведен в таблице 3.44.

Таблица 3.44 – ФБ CE30X. Входные и выходные переменные для выполнения транзакций

Имя	Тип	Значение "по умолчанию"	Описание
Входные переменные			
<i>CE30X_CONTROL</i>	BYTE	0	Сигнал управления началом выполнения транзакции: 0 – функция не активна (признак разрешения работы для других вызовов ФБ); 1 – активация выполнения транзакции. После завершения транзакции функция устанавливает значение переменной, равной "0", что является сигналом разрешения выполнения следующей транзакции
<i>CE30X_ID</i>	STRING	"	Идентификатор (номер) опрашиваемого счетчика (максимальный размер – 6 символов). Необходима для обращения к выбранному устройству из множества подключенных к физическому интерфейсу. Переменная не используется при выполнении транзакции в режиме "Одиночная транзакция без установки сессии"
<i>CE30X_PASS</i>	STRING	"	Пароль для доступа к информации (максимальный размер – 255 символов) (если не используется, то пустая строка). Переменная не используется при выполнении транзакции в режиме "Одиночная транзакция без установки сессии"
<i>CE30X_TR</i>	STRING (для CE30X) или STR_BYTE (для CE30X_v2)	"	Код запроса в соответствии с форматом, соответствующим требованиям стандарта ГОСТ ИЕС 61107-2011 (см. руководство по эксплуатации на счетчики CE30X) (максимальный размер для ФБ CE30X – 256 символов). Символами транзакции могут быть непечатные символы, задаваемые с помощью знака \$. Транзакция должна завершиться кодом \$00. <i>Например, запрос модели и версии ПО «!/?!..»:</i> '\$2F\$3F\$21\$0D\$0A\$00'; <i>Запрос состояния батарейки «R1.V_BAT().e»:</i> \$01\$52\$31\$02\$56\$5F\$42\$41\$54\$28 \$29\$03\$65\$00
<i>CE30X_TRLN</i>	BYTE	0	Длина сообщения в строке <i>CE30X_TR</i> (количество символов, передаваемых в линию)
<i>CE30X_TA</i>	TIME	1000	Тайм-аут ожидания ответа на транзакцию. Диапазон изменения параметра – от 300 до 10000 мс, но не менее длительности цикла программы

Таблица 3.44 – ФБ CE30X. Входные и выходные переменные для выполнения транзакций

Имя	Тип	Значение "по умолчанию"	Описание
CE30X_MODE	BYTE	0	Режим работы транзакции: 0 – одиночная транзакция без установки сессии; 1 – транзакция с установкой сессии без закрытия; 2 – транзакция с установкой сессии с закрытием после завершения транзакции. В режиме "Одиночная транзакция без установки сессии" содержимое переменной CE30X_TR передается в линию и ожидается один ответ в течение времени, заданного переменной CE30X_TA . В режиме "Транзакция с установкой сессии без закрытия" устанавливается сессия и выдается одна транзакция из переменной CE30X_TR . После этого можно выдать несколько транзакций в режиме "Одиночная транзакция без установки сессии". Режим "Транзакция с установкой сессии с закрытием после завершения транзакции" аналогичен предыдущему, только после завершения транзакции выдается сигнал закрытия сессии: \$01\$42\$30\$03\$75
Выходные переменные			
CE30X_BUSY	BOOL	0	Сигнал занятого ФБ: 0 – ФБ свободен для выполнения транзакций; 1 – ФБ занят
CE30X_ERR	UINT	0	Код ошибки выполнения последней транзакции: 0 – нет ошибки; <>0 – код ошибки (часть кодов определена производителем счетчика (1–38), часть необходима для индикации ошибки работы ФБ (100–103), см. таблицу 3.45)
CE30X_IN	STRING (для CE30X) или STR_BYTE (для CE30X_v2)	"	Ответ на транзакцию (максимальный размер – 255 символов). В случае ошибки "03" строка будет иметь вид: ERR03
CE30X_INLN	BYTE	0	Длина принятого сообщения в строке CE30X_OUT (количество символов, принятых из линии)
CE30X_TRCURR	UINT	0	Номер текущей выполняемой транзакции (по модулю – 65536). До выполнения первой транзакции CE30X_TRCURR должен быть равен "0". После переполнения – переход в "1"
CE30X_TRLAST	UINT	0	Номер последней завершенной транзакции (по модулю – 65536). До выполнения первой должен быть равен 0. После завершения транзакции увеличивается на "1" вне зависимости от ошибки выполнения
error	DINT	0	Код ошибки выполнения последней команды (=0 - нет ошибки, <>0 - ошибка)
recvbuf	STRING(255)	0	Временный буфер принимаемых данных по COM порту (без преобразования ФБ)
isdebug	BOOL	FALSE	Флаг отладки

Руководство по эксплуатации

При выполнении вызова ФБ для транзакции с измерительным оборудованием могут возникнуть ошибки, список кодов, возвращаемых через переменную **CE30X_ERR**, приведён в таблице 3.45.

Таблица 3.45 – Коды ошибок при выполнении запросов

Код ошибки	Описание сигнала
0	Нет ошибки
1	"Пониженное напряжение питания" Проверьте правильность подключения счетчика и его соответствие напряжению сети. Если все верно, но ошибка не исчезает, счетчик необходимо направить в ремонт. Ошибка индицируется постоянно до устранения причины ее появления
3	"Неверный пароль" Означает, что при программировании был введен пароль, несовпадающий с внутренним паролем счетчика. Необходимо повторить транзакцию с верным паролем (для второй или третьей попыток)
4	"Сбой обмена по интерфейсу" Означает, что при обмене через порт связи была ошибка паритета или ошибка контрольной суммы, произошел сбой из-за неправильного соединения, неисправности интерфейсной части счетчика или подключенного к нему устройства. Если при повторных попытках сообщение повторяется, необходимо убедиться в работоспособности счетчика и подключаемого к нему устройства, правильности соединения этих устройств и работоспособности применяемой программы связи
5	"Ошибка протокола" Появляется, если сообщение, полученное счетчиком через порт связи, синтаксически неправильное. Если при повторных попытках сообщение повторяется, необходимо убедиться в работоспособности счетчика и подключаемого к нему устройства, правильности соединения этих устройств и работоспособности применяемой программы связи
7	"Тайм-аут при приеме сообщения" Означает, что в отведенное протоколом время не было получено необходимое сообщение. Если при повторных попытках сообщение повторяется, необходимо убедиться в работоспособности счетчика и подключаемого к нему устройства, правильности соединения этих устройств и работоспособности применяемой программы связи
8	"Тайм-аут при передаче сообщения" Означает, что в отведенное протоколом время не установился режим готовности канала связи. Если при повторных попытках сообщение повторяется, необходимо убедиться в работоспособности счетчика и наличии необходимых условий для связи
9	"Исчерпан лимит ошибок ввода неверного пароля" Означает, что при программировании было более трех попыток ввода неверного пароля в течение текущих суток. Дождитесь следующих суток и введите правильный пароль
10	"Недопустимое число параметров в массиве" Означает, что количество одноименных параметров превышает допустимое значение и параметр, в ответ на которого было выведено это сообщение, игнорируется
12	"Неподдерживаемый параметр" Означает, что параметр отсутствует в списке параметров счетчика. Использовать параметры, допустимые для данного счетчика
14	"Запрет программирования" Означает, что не нажата кнопка "ДСТП", не введен пароль или нет параметра в списке программирования пользователя. Необходимо снять пломбу с кнопки "ДСТП", перевести счетчик в режим программирования и/или ввести пароль. Для пользователя, при необходимости, ввести параметр в список программирования
15	"Недопустимое чтение" Означает, что не введен пароль или нет параметра в списке для чтения при парольном чтении пользователем. Необходимо ввести пароль или, при необходимости, ввести параметр в список пользователя или отменить парольное чтение

Таблица 3.45 – Коды ошибок при выполнении запросов

Код ошибки	Описание сигнала
16	"Калибровка запрещена" Означает, что произведена попытка записи технологического (метрологического) параметра без права доступа. Необходимо вскрыть счетчик (при наличии соответствующих прав) и установить технологическую перемычку
17	"Недопустимое значение параметра" Уточнить диапазон допустимых значений параметра и ввести правильное значение
18	"Отсутствует запрошенное значение параметра" Уточнить аргументы выбора запрашиваемых значений параметра и ввести правильные значения
19	"Калибровка запрещена"
20	"Ошибка измерителя" Снять со счетчика питающее напряжение. Если после подключения ошибка останется, счетчик необходимо направить в ремонт. Ошибка индицируется постоянно до устранения причины ее появления
21	"Неполадки в работе часов реального времени" Проверить правильность индикации счетчиком текущих даты и времени. Для сброса индикации ошибки произвести программирование даты или времени. Если ошибка появляется снова, включить и выключить счетчик и запрограммировать дату или время. Если ошибка не исчезает, отправить счетчик в ремонт. Ошибка индицируется циклически после каждого просматриваемого параметра
22	"Ответ на запрос превышает размер выходного буфера или размер буфера установлен равным нулю" Проверить заданный размер выходного буфера или откорректировать запрос
23	"Ошибка модуля электронной пломбы" Сбросить ошибку чтением через интерфейс или оптопорт параметра <i>STAT_</i> . Если через некоторое время ошибка появится повторно, счетчик необходимо направить в ремонт. Ошибка индицируется циклически после каждого просматриваемого параметра
30	"Признак сбоя при записи в энергонезависимую память данных" Счетчик использует резервную копию данных при записи. Сбрасывается чтением параметра состояния счетчика. При возникновении данного сообщения необходимо проверить корректность хранимых энергетических данных
36	" <i>Ошибка контрольной суммы метрологических параметров</i> " Требуется поверка счетчика и ввод технологических метрологических коэффициентов со вскрытием счетчика. Ошибка индицируется циклически после каждого просматриваемого параметра
37	" <i>Ошибка контрольной суммы накапливаемых параметров</i> " Проверить по возможности накопленную информацию на достоверность. Сбросить ошибку перепрограммированием любого параметра. Ошибка индицируется циклически после каждого просматриваемого параметра
38	"Ошибка контрольной суммы кода в памяти программ" Сбросить ошибку чтением через интерфейс или оптопорт параметра <i>STAT_</i> . Если через некоторое время ошибка появится повторно, счетчик необходимо направить в ремонт. Ошибка индицируется циклически после каждого просматриваемого параметра
100	"Не выполнена инициализация ФБ"
101	В буфере <STX><ETX><VСС> Пустой массив выдается при чтении параметра, запрещенного пользователю, или параметра, ненакопленного и незафиксированного на данный момент времени
102	За время тайм-аута, задаваемого значением сигнала <i>CE30X_TA</i> , не принято ни одного байта
103	Обнаружено искажение входящего сообщения (нарушение формата, обнаружено искажение сообщение по недопустимым кодам символов или ошибки контрольной суммы сообщения для транзакций, в которых она предусмотрена)

3.8.4 Описание работы ФБ

К одной интерфейсной линии может быть подключен один ПЛК, выполняющий функции ведущего устройства (опросчика), и до 32-х подчинённых устройств, использующих протокол ГОСТ Р МЭК 61107-2001 (IEC 61107-97) «Обмен данными при считывании показаний счетчиков, тарификации и управлении нагрузкой. Прямой локальный обмен данными».

Для обеспечения корректной совместной работы оборудования необходимо выполнить следующие действия:

1 С помощью сервисных программ производителя измерительного оборудования произвести конфигурирование параметров работы по интерфейсу *RS-485*, задав для каждого устройства уникальный адрес, одинаковую скорость передачи (рекомендуется 9600 бит/с) и единый формат данных (7.1.E).

2 Произвести подключение устройств к линии, соблюдая полярность сигналов физического интерфейса и правила подключения устройств к интерфейсу *RS-485* (см. руководство по эксплуатации на измерительное оборудование).

Внимание! Для работы оборудования Энергомера серии *CE30x* по интерфейсу *RS-485* требуется наличие внешнего источника питания.

3 Подготовить и сохранить в ПЛК проект с задачей пользователя с установкой выбранных выше параметров (см. шаг 1) для опрашиваемого оборудования.

4 Подключить питание устройств, запустить проект *CoDeSys* ПЛК в режиме отладки, убедиться в корректности опроса по отсутствию ошибок выполнения транзакций (переменные *INIT_ErrCode* и *CE30X_ERR ФБ*) и совпадению значений показаний на панели индикации и в переменных программы.

Примечание – В случае возникновения ошибок в процессе обмена произвести контроль питания интерфейса, физического подключения, конфигурационных параметров интерфейса, проверить уникальность адресов и тайм-аутов выполнения транзакций.

Для создания проекта с поддержкой функций получения данных по протоколу ГОСТ Р МЭК 61107-2001 в ПЛК Элсима последовательность шагов должна быть следующей:

1 Создайте проект ПЛК Элсима согласно 2.8.

2 Создать экземпляр ФБ и объявить необходимые переменные согласно приложению Е.

3 Обеспечить вызов экземпляра ФБ с активацией необходимой функции для получения данных от измерительного устройства Энергомера:

- 1) Инициализация ФБ (должна выполняться один раз в начале работы программы) (см. Е.1 приложения Е);
- 2) Выполнение транзакции для получения данных:
 - выполнение одиночной транзакции (см. Е.2 приложения Е);
 - выполнение группы транзакций с открытием сессии (см. Е.3 приложения Е).

Группа транзакций с открытием сессии должна состоять из следующих действий:

1) Выполнение транзакции с открытием сессии – инициализация переменных с установленными параметрами опрашиваемого устройства и выдача транзакции в режиме "*Транзакция с установкой сессии без закрытия*" (значение "*1*");

2) Продолжение работы в сессии (может быть несколько транзакций) – подготовка кода транзакции и выдача транзакции в режиме "*Одиночная транзакция без установки сессии*" (значение "*0*");

3) Закрытие сессии – выдача последней транзакции ".*B0.u*" (\$01 \$42 \$30 \$03 \$75).

Примечания:

1 Если выдача очередной транзакции на шаге 2 или 3 будет задержана более чем на 1,2 с (зависит от конфигурации и версии ПО оборудования), то сессия закрывается автоматически, на очередную транзакцию данные не поступят и для продолжения получения данных необходимо вновь выполнить шаг 1.

2 Для выполнения активированных функций в конце текста программы задачи пользователя обязательно вызвать созданный экземпляр ФБ (см. Е.4 приложения Е).

3.9 Работа со счетчиками электроэнергии Меркурий 230/233/234

Для обеспечения информационного взаимодействия ПЛК с многофункциональными приборами учета электроэнергии Меркурий 230/233/234 при использовании физического интерфейса RS-485 в состав конфигурационного пакета включена библиотека **M23XLibrary**, в которую входит ФБ **M23X** и **M23x_v2** (*Device (ELSYMA) → Plc Logic → Application → Менеджер библиотек → ElsyMA_M23X*, см. 2.8.1.2).

Вызов функций ФБ позволяет осуществить инициализацию физического интерфейса RS-485, используемого для подключения приборов учета электроэнергии Меркурий, обеспечивает возможность получения измеренных значений, массивов данных накопленной энергии, параметров времени и журналов событий, осуществляет контроль состояния оборудования и чтения основных параметров счётчика.

В этом разделе руководства содержится описание работы с библиотекой ФБ **M23X** и **M23X_v2**, а примеры применения ФБ приведены в приложении Ж.

По выполняемым функциям ФБ **M23X_v2** полностью идентичен **M23X**. Отличия между блоками только в типах сигналов для **M23X_TR** и **M23X_DATA**. Для ФБ M23X эти сигналы имеют тип STRING, для которого есть ограничения длины в 256 байт. Для ФБ M23X_v2 эти сигналы имеют тип STR_BYTE:

```
TYPE STR_BYTE:
UNION
str:STRING;
b:ARRAY[0..1023] OF BYTE;
END_UNION
END_TYPE
```

Таким образом, при использовании ФБ M23X_v2 снимается ограничение 256 байт для отправляемых транзакций и принимаемых ответов, при этом пользователь при формировании запросов и обработке ответов на транзакции может использовать либо тип STRING, либо массив байт ARRAY[0..1023] OF BYTE.

3.9.1 Конфигурирование ФБ M23X

Применение ФБ не требует специального конфигурирования, т.к. библиотека **M23XLibrary** уже включена в состав пакета конфигурирования ПЛК. Применение вызовов ФБ в задаче пользователя *CoDeSys* обеспечивает выполнение следующих основных функций:

- инициализация физического интерфейса и переменных ФБ;
- выполнение транзакции (запроса на чтение или запись данных).

Для активации выполнения функции имеется свой управляющий сигнал. Среди входных сигналов ФБ сигналами активации являются:

- **M23X_INIT** – для инициализации физического интерфейса и переменных ФБ;

- **M23X_CONTROL** – для выполнения транзакции.

Внимание! Интерфейс *RS-485* может использоваться только одним программным модулем, поддерживающим определённый протокол взаимодействия.

При использовании функции ФБ **M23X** и **M23X_v2** нельзя использовать интерфейс *RS-485* для других коммуникаций (таких как *Modbus RTU* и т.п.). Для этого в конфигурации проекта задачи пользователя ПЛК Элсима для устройства **RS485** в закладке **Редактор соединения** необходимо установить для параметра **Server** значение "None" (рисунок 3.53).

Для использования функций библиотеки **M23XLibrary** необходимо:

1 В разделе переменных создать экземпляр ФБ. Например:

```
FB_M23X : ElsyMA_M23X.M23X; //для ФБ M23X
```

или

```
FB_M23X : ElsyMA_M23X.M23X_v2; // для ФБ M23X_v2
```

2 В тексте программы задачи пользователя обеспечить вызов ФБ.

Для программы на языке **ST**, при объявлении экземпляра ФБ, как указано выше, код должен быть следующий:

```
FB_M23X();
```

3 С помощью сервисных средств для оборудования Меркурий необходимо убедиться в уникальности адресов, подключаемых к одной линии счетчиков, проверить пароли доступа.

4 Выполнить физическое подключение линии согласно правилам подключений линий интерфейса *RS-485*.

Совпадение адресов приведёт к сбоям в работе и может вызвать поломку оборудования.

3.9.2 Инициализация ФБ M23X

Функция инициализации вызывается один раз в начале выполнения программы с предварительно установленными значениями параметров или параметрами "по умолчанию". Функция инициализации для ФБ **M23X_v2** и **M23X** одинакова.

Функция инициализации обеспечивает инициализацию переменных для работы ФБ и инициализацию физического интерфейса *RS-485* ПЛК Элсима системными вызовами *CoDeSys* в соответствии со значениями входных переменных и заданного в программе имени порта (имя соответствует физическому устройству *RS-485* в составе ПЛК Элсима).

Набор переменных для выполнения инициализации ФБ приведен в таблице 3.46.

Таблица 3.46 – ФБ **M23X**. Входные и выходные переменные для выполнения инициализации

Имя	Тип	Значение "по умолчанию"	Описание
Входные переменные			
<i>ptTR</i>	TIME	<i>TIME#3s0ms</i>	Таймаут ожидания ответа на транзакцию
<i>M23X_INIT</i>	BYTE	0	Сигнал управления началом выполнения инициализации ФБ: 0 – функция не активна (признак разрешения работы для других вызовов ФБ); 1 – активация выполнения инициализации
<i>BoudeRate</i>	BYTE	5	Скорость передачи данных в интерфейсе, бит/с: 1 – 4800; 2 – 9600
<i>Parity</i>	BYTE	0	Тип контроля для символа: 0 – None (нет);

Таблица 3.46 – ФБ М23Х. Входные и выходные переменные для выполнения инициализации

Имя	Тип	Значение "по умолчанию"	Описание
			1 – <i>Odd</i> (нечёт.); 2 – <i>Even</i> (чёт.)
<i>StopBitNmb</i>	BYTE	1	Количество стоп-бит в символе. Допустимые значения – 1, 2
Выходные переменные			
<i>INIT_ErrCode</i>	DINT		Код ошибки. Допустимые значения: 0 – успешная инициализация порта с заданными параметрами; <>0 – ошибка инициализации. Коды ошибок: "-1" – ошибка инициализации СОМ-порта; "-2" – ошибка тестирования передачи в СОМ-порт; "-3" – ошибка тестирования приёма из СОМ-порта
Примечание – * В версии 3.5.6.1 и ниже библиотеки M23XLibrary данное значение не доступно			

3.9.3 Выполнение транзакции ФБ М23Х

Функция выполнения транзакции с измерительным оборудованием вызывается из задачи пользователя периодически.

Функция выполнения транзакции обеспечивает выполнение одиночных или последовательности транзакций (передачи запросов и приёма ответов по интерфейсу) с предварительной обработкой результата для контроля ошибок времени ответа, искажения данных и кода ошибки в ответе.

Внимание! Вызов функции выполнения транзакции до завершения инициализации приведёт к ошибке! Для выполнения следующей транзакции необходимо дождаться окончания уже начатой транзакции! Несоблюдение этого правила приведёт к ошибке выполнения.

Набор переменных для выполнения транзакций ФБ (управление работой, задание типа получаемых данных и контроля правильности выполнения) приведен в таблице 3.47.

Таблица 3.47 – ФБ М23Х. Входные и выходные переменные для выполнения транзакций

Имя	Тип	Значение "по умолчанию"	Описание
Входные переменные			
<i>M23X_CONTROL</i>	BYTE	0	Сигнал управления началом выполнения транзакции: 0 – функция не активна (признак разрешения работы для других вызовов ФБ); 1 – активация выполнения транзакции. После завершения транзакции функция устанавливает значение переменной, равной "0", что является сигналом разрешения выполнения следующей транзакции
<i>M23X_ADR</i>	STRING	"	Идентификатор (номер) опрашиваемого счетчика (максимальный размер – 6 символов). Необходима для обращения к выбранному устройству из множества подключенных к физическому интерфейсу. Переменная не используется при выполнении транзакции в режиме " <i>Одиночная транзакция без установки сессии</i> "

Таблица 3.47 – ФБ М23Х. Входные и выходные переменные для выполнения транзакций

Имя	Тип	Значение "по умолчанию"	Описание
<i>M23X_PASS</i>	STRING	"	Пароль для доступа к информации (максимальный размер – 255 символов) (если не используется, то пустая строка). Переменная не используется при выполнении транзакции в режиме " <i>Одиночная транзакция без установки сессии</i> "
<i>M23X_FUNC</i>	BYTE		Код функции запрашиваемых данных; 1 – инициализация соединения и открытие парольной сессии; 2 – завершение сессии; 4 – запросы на чтение параметров времени и журналов событий; 5 – запросы на чтение массивов данных накопленной энергии; 8 – запросы основных параметров счётчика
<i>M23X_TR</i>	STRING (для М23Х) или STR_BYTE (для М23Х_v2)	"	Код запроса в соответствии с форматом, соответствующим требованиям протокола Меркурий (максимальный размер для М23Х– 256 символов). Символами транзакции могут быть непечатные символы, задаваемые с помощью знака \$. Транзакция должна завершиться кодом \$00. <i>Например, запрос накопленной энергии за текущие сутки:</i> '\$00\$05\$40\$00\$21\$E5\$00';
<i>M23X_TRLN</i>	BYTE	0	Длина сообщения в строке <i>M23X_TR</i> (количество символов, передаваемых в линию)
<i>M23X_TA</i>	TIME	1000	Тайм-аут ожидания ответа на транзакцию. Диапазон изменения параметра – от 300 до 10000 мс, но не менее длительности цикла программы
<i>M23X_MODE</i>	BYTE	0	Режим работы транзакции: 0 – одиночная транзакция без установки сессии; 1 – транзакция с установкой сессии без закрытия; 2 – транзакция с установкой сессии с закрытием после завершения транзакции. В режиме " <i>Одиночная транзакция без установки сессии</i> " содержимое переменной <i>M23X_TR</i> передаётся в линию и ожидается один ответ в течение времени, заданного переменной <i>M23X_TA</i> . В режиме " <i>Транзакция с установкой сессии без закрытия</i> " устанавливается сессия и выдаётся одна транзакция из переменной <i>M23X_TR</i> . После этого можно выдать несколько транзакций в режиме " <i>Одиночная транзакция без установки сессии</i> ". Режим " <i>Транзакция с установкой сессии с закрытием после завершения транзакции</i> " аналогичен предыдущему, только после завершения транзакции выдаётся сигнал закрытия сессии: \$00\$02\$80\$71
Выходные переменные			

Таблица 3.47 – ФБ M23X. Входные и выходные переменные для выполнения транзакций

Имя	Тип	Значение "по умолчанию"	Описание
<i>M23X_BUSY</i>	BOOL	<i>0</i>	Сигнал занятого ФБ: <i>0</i> – ФБ свободен для выполнения транзакций; <i>1</i> – ФБ занят
<i>M23X_ERR</i>	INT	<i>0</i>	Код ошибки выполнения последней транзакции: <i>0</i> – нет ошибки; <> <i>0</i> – код ошибки (часть кодов определена производителем счетчика (<i>1–48</i>), часть необходима для индикации ошибки работы ФБ (<i>100–103</i>), см. таблицу 3.48)
<i>M23X_DATA</i>	STRING (для M23X) или STR_BYTE (для M23X_v2)	"	Ответ на транзакцию (максимальный размер – для M23X 256 байт). В случае ошибки значение поля неопределённое
<i>M23X_INLN</i>	BYTE	<i>0</i>	Длина принятого сообщения в строке <i>M23X_OUT</i> (количество символов, принятых из линии)
<i>error</i>	DINT	<i>0</i>	Код ошибки выполнения последней команды (=0 - нет ошибки, <>0 - ошибка)
<i>recvbuf</i>	STRING(255)	<i>0</i>	Временный буфер принимаемых данных по COM порту (без преобразования ФБ)
<i>isdebug</i>	BOOL	<i>FALSE</i>	Флаг отладки

При выполнении вызова ФБ для транзакции с измерительным оборудованием могут возникнуть ошибки, список кодов, возвращаемых через переменную *M23X_ERR*, приведён в таблице 3.48.

Таблица 3.48 – Коды ошибок при выполнении запросов

Код ошибки	Описание сигнала
0	<i>Нет ошибки</i>
	E01 Неисправность батареи питания E02 Нарушено функционирование памяти № 2 E03 Нарушено функционирование UART1 E04 Нарушено функционирование ADS E05 Ошибка обмена с памятью E06 Неисправность часов E07 Нарушено функционирование памяти № 3 E09 Ошибка целостности ПО E10 Ошибка калибровочных коэффициентов E11 Ошибка регистров энергии от сброса E12 Ошибка сетевого адреса E13 Ошибка серийного номера E14 Поврежден пароль E15 Ошибка массива вариантов исполнения E16 Поврежден флаг тарификатора E17 Поврежден флаг отключения нагрузки E18 Ошибка лимита мощности E19 Ошибка лимита энергии E20 Ошибка параметров интерфейса E21 Ошибка параметров индикации по тарифам E22 Ошибка параметров индикации по периодам E23 Ошибка множителя тайм-аута интерфейса E24 Поврежден байт программируемых флагов E25 Повреждено расписание праздничных дней E26 Повреждено тарифное расписание E27 Поврежден массив таймера E28 Ошибка перехода зима/лето E29 Ошибка местоположения прибора E30 Повреждены коэффициенты трансформации E31 Повреждены регистры энергии E32 Ошибка параметров среза E33 Повреждены регистры среза E34 Ошибка указателей журнала событий E35 Ошибка записи журнала событий E36 Поврежден регистр учета технических потерь E37 Ошибка мощности технических потерь E38 Поврежден регистр накопленной энергии потерь E39 Повреждены регистры энергии пофазного учета E40 Флаг поступления широковещ. сообщения E41 Ошибка указателей журнала ПКЭ E42 Ошибка записи журнала ПКЭ E47 Процедура коррекции времени E48 Напряжение батареи менее 2,65 В (" <i>Пониженное напряжение питания</i> ")
100	"Не выполнена инициализация ФБ"
101	В буфере пустой массив. Выдается при чтении параметра, запрещенного пользователю, или параметра, ненакопленного и незафиксированного на данный момент времени
102	За время тайм-аута, задаваемого значением сигнала <i>M23X_TA</i> , не принято ни одного байта
103	Обнаружено искажение входящего сообщения (нарушение формата, обнаружено искажение сообщение по недопустимым кодам символов или ошибки контрольной суммы сообщения для транзакций, в которых она предусмотрена)

3.9.4 Описание работы ФБ

К одной интерфейсной линии может быть подключен один ПЛК, выполняющий функции ведущего устройства (опросчика), и до 32-х подчинённых устройств, использующих единый протокол Меркурий.

Для обеспечения корректной совместной работы оборудования необходимо выполнить следующие действия:

1 С помощью сервисных программ производителя измерительного оборудования произвести конфигурирование параметров работы по интерфейсу RS-485, задав для каждого устройства уникальный адрес, одинаковую скорость передачи (рекомендуется 9600 бит/с) и единый формат данных (8.1.N).

2 Произвести подключение устройств к линии, соблюдая полярность сигналов физического интерфейса и правила подключения устройств к интерфейсу RS-485 (см. руководство по эксплуатации на измерительное оборудование).

Внимание! Для работы оборудования Меркурий серии **M23X** и **M23X_v2** по интерфейсу *RS-485* требуется наличие внешнего источника питания.

3 Подготовить и сохранить в ПЛК проект с задачей пользователя с установкой выбранных выше параметров (см. шаг 1) для опрашиваемого оборудования.

4 Подключить питание устройств, запустить проект *CoDeSys* ПЛК в режиме отладки, убедиться в корректности опроса по отсутствию ошибок выполнения транзакций (переменные *INIT_ErrCode* и *M23X_ERR ФБ*) и совпадению значений показаний на панели индикации и в переменных программы.

Примечание – В случае возникновения ошибок в процессе обмена произвести контроль питания интерфейса, физического подключения, конфигурационных параметров интерфейса, проверить уникальность адресов и тайм-аутов выполнения транзакций.

Для создания проекта с поддержкой функций получения данных по протоколу ГОСТ Р МЭК 61107-2001 в ПЛК Элсима последовательность шагов должна быть следующей:

1 Создайте проект ПЛК Элсима согласно 2.8.

2 Создать экземпляр ФБ и объявить необходимые переменные согласно приложению Ж.

3 Обеспечить вызов экземпляра ФБ с активацией необходимой функции для получения данных от измерительного устройства Меркурий:

1) Инициализация ФБ (должна выполняться один раз в начале работы программы) (см. Ж.1 приложения Ж);

2) Выполнение транзакции для получения данных:

– выполнение одиночной транзакции (см. Ж.2 приложения Ж);

– выполнение группы транзакций с открытием сессии (см. Ж.3 приложения Ж).

Группа транзакций с открытием сессии должна состоять из следующих действий:

1) Выполнение транзакции с открытием сессии – инициализация переменных с установленными параметрами опрашиваемого устройства и выдача транзакции в режиме "*Транзакция с установкой сессии без закрытия*" (значение "*1*");

2) Продолжение работы в сессии (может быть несколько транзакций) – подготовка кода транзакции и выдача транзакции в режиме "*Одиночная транзакция без установки сессии*" (значение "*0*");

3) Закрытие сессии – выдача последней транзакции "*.B0.u*" (\$01 \$42 \$30 \$03 \$75).

Примечания

1 Если выдача очередной транзакции на шаге 2 или 3 будет задержана более чем на 1,2 с (зависит от конфигурации и версии ПО оборудования), то сессия закрывается автоматически, на очередную транзакцию данные не поступят и для продолжения получения данных необходимо вновь выполнить шаг 1.

2 Для выполнения активированных функций в конце текста программы задачи пользователя обязательно вызвать созданный экземпляр ФБ (см. Ж.4 приложения Ж).

3.10 Программный модуль синхронизации времени

Программный модуль **tsync**, представляющий собой канал синхронизации, предназначен для работы в составе программного обеспечения контроллера Элсима-М01 и для обеспечения синхронизации временем с модулями УВВ.

Программный модуль **tsync** обеспечивает выполнение следующих функций:

- конфигурирование и запуск службы NTP для получения времени от источников по протоколу NTP;
- для сторонних источников времени (ANYTIME) получение времени от системы *CoDeSys* и передачу его в службу NTP.

Служба NTP выполняет выбор наиболее достоверных источников времени из предоставленного пользователем списка по следующим параметрам и в следующем порядке:

1 *Stratum* (стратум, уровень) – это число от "1" до "16", указывающее на точность сервера. Единица означает максимальную точность, "16" – сервер недоступен.

2 *Offset* (смещение) – разница между системными часами контроллера и удаленного сервера. В ходе синхронизации это значение должно понижаться (приближаться к нулю), указывая на то, что системные часы контроллера идут все точнее.

3 *Jitter* (дисперсия) – мера статистических отклонений от значения смещения (поле *offset*) по нескольким успешным парам запрос-ответ. Меньшее значение дисперсии предпочтительнее, поскольку позволяет точнее синхронизировать время.

4 *Delay* (задержка) – определяет время (в секундах), необходимое для получения ответа на запрос о времени.

Список эталонных NTP-серверов регулярно пересматривается службой. Более подробная информация о работе службы NTP представлена в стандартах NTP (RFC 778, RFC 891, RFC 956, RFC 958, RFC 1305, RFC2030, RFC 4330, RFC 5905).

На рисунке 3.54 представлена схема работы модуля **tsync** в составе программного обеспечения контроллера ЭЛСИМА, а также схема передачи данных от различных источников времени (NTP-серверов, сторонних источников времени (ANYTIME)) в сервис NTP.

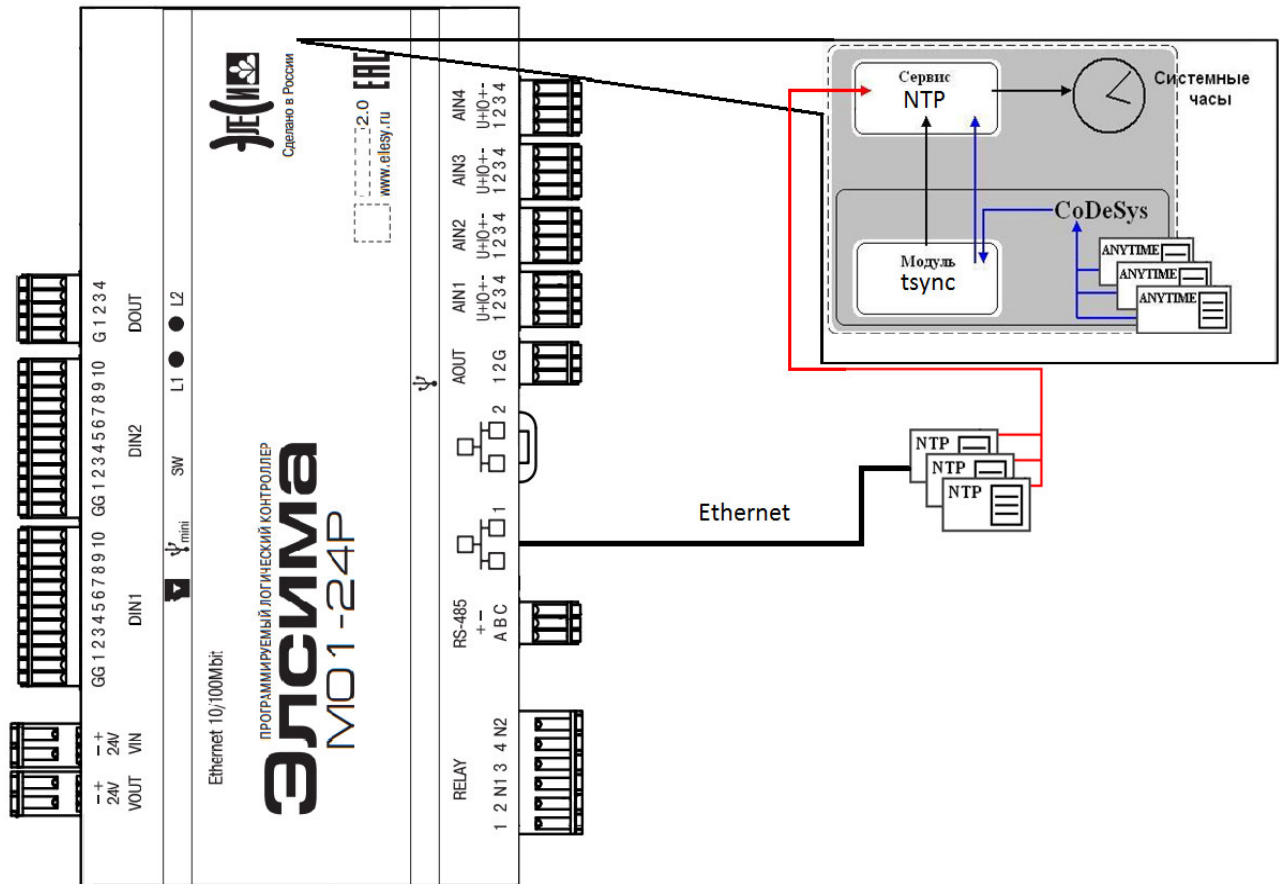


Рисунок 3.54 – Модель **tsync**. Схема передачи данных в сервис NTP

Символьное обозначение модуля, используемое в сервисной программе – **tsync** (таблица 3.49).

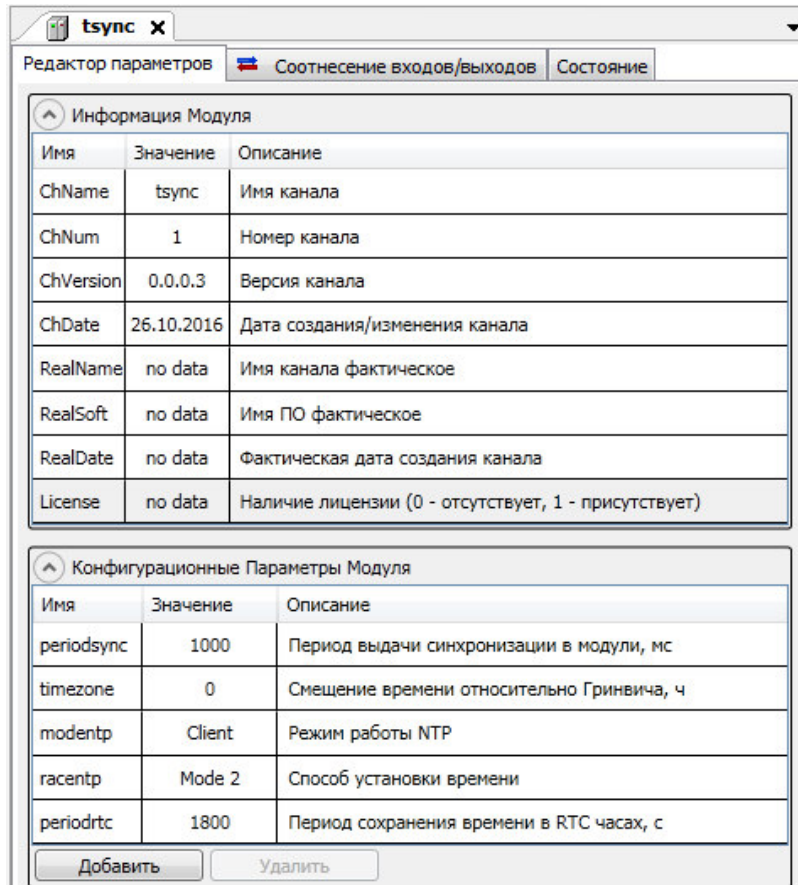
3.10.1 Настройка конфигурационных параметров модуля **tsync**

Настройка конфигурационных параметров осуществляется на закладке просмотра и настройки модуля **tsync**. Для выполнения операции следует

1 Добавить модуль **tsync** в конфигурацию. Для этого в дереве устройств найти **SoftModules**, нажать правой кнопкой «мыши» и в контекстном меню выбрать команду «Добавить устройство...». В появившемся окне выбрать **Оборудование компании ЭлеСи->ПЛК->Tsync->tsync** и нажать кнопку «Добавить устройство».

2 Открыть закладку просмотра и настройки модуля **tsync**, выделив имя модуля в дереве устройств и дважды нажав левую кнопку «мыши».

3 Перейти на закладку «Редактор параметров» (рисунок 3.55).

Рисунок 3.55 – Модуль **tsync**. Закладка «Редактор параметров»

4 Настроить соответствующие параметры модуля:

- Информационные параметры модуля **tsync** представлены в таблице 3.49. Данные параметры недоступны для редактирования пользователем;

Таблица 3.49 – Модуль **tsync**. Информационные данные

Имя	Значение по умолчанию	Описание
<i>ChName</i>	<i>tsync</i>	Имя канала
<i>ChNum</i>	<i>1</i>	Номер канала
<i>ChVersion</i>	<i>0.0.0.3</i>	Версия канала
<i>ChDate</i>	<i>DD.MM.YY</i>	Дата создания/изменения канала в формате <i>день месяц год</i>
<i>RealName</i>	<i>no data</i>	Имя канала фактическое
<i>RealSoft</i>	<i>no data</i>	Имя ПО фактическое
<i>RealDate</i>	<i>no data</i>	Фактическая дата создания канала
<i>License</i>	<i>no data</i>	Наличие лицензии (0 – отсутствует, 1 - рисутствует)

ВНИМАНИЕ! Начиная с версии ПО 02.08, параметр **License** показывает наличие лицензии на данный программный модуль. Значение параметра **License**, равное "0", свидетельствует об отсутствии лицензии. В данном случае программный модуль не будет запускаться.

- Конфигурационные параметры модуля **tsync** представлены в таблице 3.50.

Таблица 3.50 – Модуль **tsync**. Конфигурационные параметры

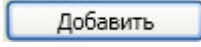
Имя	Значение по умолчанию	Описание
<i>periodsync</i>	<i>1000</i>	Период выдачи синхронизации в модули, мс. Диапазон допустимых значений: от 0 до 4294967295 мс.
<i>timezone</i>	<i>0</i>	Смещение времени относительно Гринвича, ч. Диапазон допустимых

Таблица 3.50 – Модуль *tsync*. Конфигурационные параметры

Имя	Значение по умолчанию	Описание
		значений: от -128 до 127 ч.
<i>modentp</i>	<i>Client</i>	Режим работы NTP. Допустимые значения: <ul style="list-style-type: none"> • 0 – <i>Client</i>; • 1 – <i>Client_Server</i>; • 2 – <i>Server</i>
<i>racentp</i>	<i>Mode 2</i>	Способ установки времени: <ul style="list-style-type: none"> • 0 – <i>Mode1</i> – время устанавливается всегда плавно; • 1 – <i>Mode2</i> – время устанавливается скачком по старту, при работе плавно; • 2 – <i>Mode3</i> – время устанавливается скачком по старту и во время работы; • 3 – <i>Mode4</i> – время устанавливается скачком по старту, при работе отключение
<i>periodrtc</i>	<i>1800</i>	Период сохранения времени в RTC часах, с. Диапазон допустимых значений: от 10 до 1800 с.

5. Выбрать тип синхронизации (см. 3.10.2).

3.10.2 Выбор типа синхронизации

Выбор типа синхронизации выполняется в закладке «*Редактор параметров*» модуля *tsync* с помощью кнопки , при нажатии которой открывается окно "Выбор типа синхронизации" – см. рисунок 3.56.

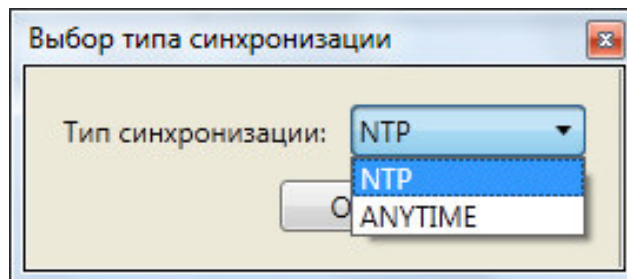


Рисунок 3.56 - Модуль *tsync*. Окно выбора типа синхронизации

Поле *Тип синхронизации:* имеет следующие типы источников времени:

- *NTP* – получение сигнала синхронизации от NTP-сервера (через порт *Ethernet*). Существует возможность выбора до пяти NTP-серверов. При выборе NTP в качестве источника времени, закладка «*Редактор параметров*» принимает вид, представленный на рисунке 3.57;

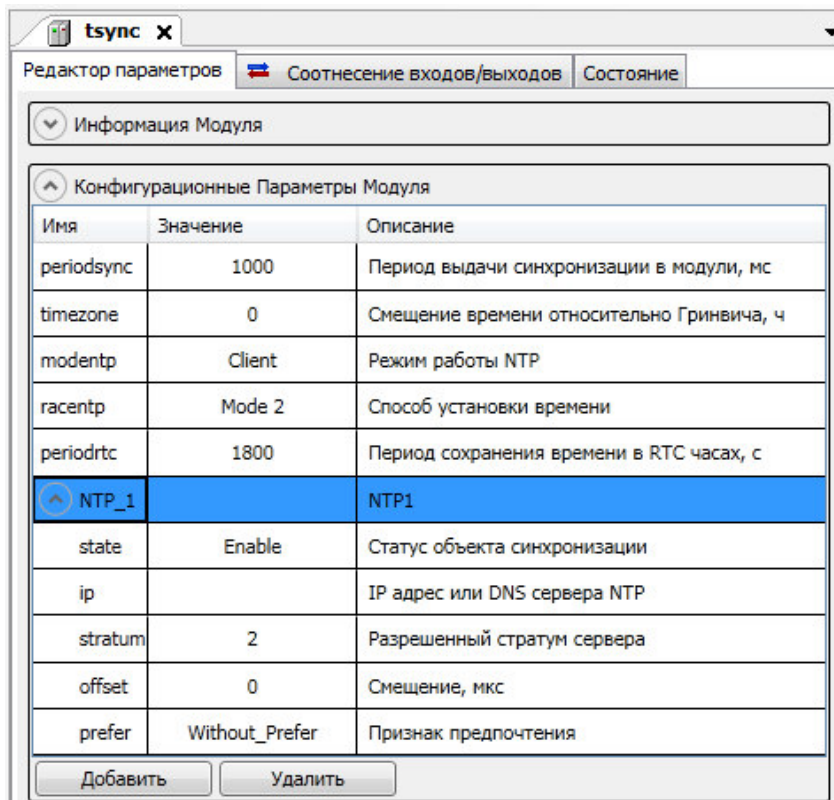


Рисунок 3.57 - Модуль `tsync`. Закладка «Редактор параметров». Конфигурационные параметры NTP-сервера

- *ANYTIME* – получение сигнала синхронизации от сторонних источников времени. Существует возможность выбора не более шести сторонних источников. При выборе синхронизации времени от стороннего источника, закладка «Редактор параметров» принимает вид, представленный на рисунке 3.58.

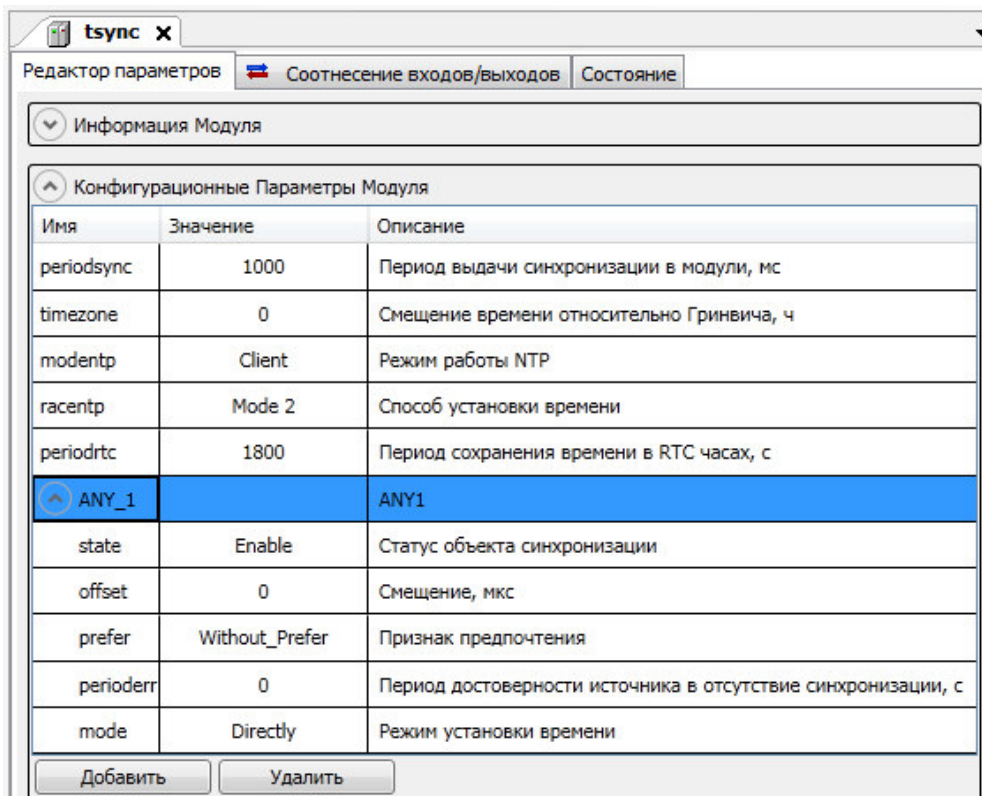


Рисунок 3.58 - Модуль `tsync`. Закладка «Редактор параметров». Конфигурационные параметры стороннего источника времени

Описание параметров типов источников времени представлено в таблице 3.51.

Таблица 3.51 – Модуль **tsync**. Конфигурационные параметры источников времени

Имя	Значение по умолчанию	Описание
NTP-сервер		
<i>state</i>	<i>Enable</i>	Состояние объекта синхронизации: <ul style="list-style-type: none"> • 1 – <i>Enable</i> – включено; • 0 – <i>Disable</i> – выключено
<i>ip</i>		IP-адрес или DNS NTP-сервера
<i>stratum</i>	2	Разрешенный стратум сервера. Допустимый диапазон значений: от 0 до 16. При установке пользователем значения меньше, чем реальный стратум источника времени, служба NTP признает данный источник невалидным
<i>offset</i>	0	Смещение, мкс. Допустимый диапазон значений: от - 2147483648 до 2147483647
<i>prefer</i>	<i>Without_Prefer</i>	Признак предпочтения: <ul style="list-style-type: none"> • 1 - <i>With_Prefer</i> – есть предпочтение; • 0 - <i>Without_Prefer</i> – нет предпочтения
Сторонний источник времени ANYTIME		
<i>state</i>	<i>Enable</i>	Состояние объекта синхронизации: <ul style="list-style-type: none"> • 1 – <i>Enable</i> – включено; • 0 – <i>Disable</i> – выключено
<i>offset</i>	0	Смещение, мкс. Допустимый диапазон значений: от - 2147483648 до 2147483647
<i>prefer</i>	<i>Without_Prefer</i>	Признак предпочтения: <ul style="list-style-type: none"> • 1 – <i>With_Prefer</i> – есть предпочтение; • 0 – <i>Without_Prefer</i> – нет предпочтения
<i>perioderr</i>	0	Период достоверности источника в отсутствие синхронизации, с: <ul style="list-style-type: none"> • 0 – бесконечная достоверность, • 1 и далее – установленный период. Допустимый диапазон значений: от 0 до 4294967295
<i>mode</i>	<i>Directly</i>	Режим установки времени: <ul style="list-style-type: none"> • 0 – <i>Directly</i> – непосредственно. Установка времени выполняется непосредственно в системные часы. Однако, в данном режиме возможно отключение службы NTP, если параметр <i>racentp</i> = 2 (см. таблицу 3.50) не будет установлен; • 1 – <i>NTP</i> – через NTP-службу. ВНИМАНИЕ! Использование режима <i>Directly</i> для источника ANYTIME может нарушить работу службы NTP, поэтому рекомендуется отключать другие источники времени с помощью параметра <i>state</i>
<p>Примечание – Признак предпочтения <i>prefer</i> устанавливается пользователем определенному источнику времени в том случае, если данный источник считается наиболее достоверным. Ответ от предпочтительного источника времени будет отброшен, если он значительно отличается от ответов других источников, в противном случае он будет использоваться безотносительно к другим ответам</p>		

3.10.3 Настройка диагностических параметров и соотнесение сигналов

Модуль **tsync** имеет следующий набор диагностических сигналов, отображающих работу модуля синхронизации времени. Настройка сигналов выполняется на закладке «Соотнесение входов/выходов» модуля **tsync** (см. 2.8.1.4.2) Описание диагностических сигналов представлено в таблице 3.52.

Таблица 3.52 – Модуль **tsync**. Набор диагностических сигналов

Имя	Тип	Описание
<i>cstatus</i>	UDINT	Текущее состояние работы канала. Допустимые значения сигнала:

Таблица 3.52 – Модуль **tsync**. Набор диагностических сигналов

Имя	Тип	Описание
		0 – нормальный режим работы; 1 – ошибка работы канала
<i>chstat</i> – статистика работы модуля (тип EleSyTypes.chstat):		
<i>rx_cnt</i>	UDINT	Счетчик принятых по сетевому соединению (каналу) кадров
<i>rx_bad_frames</i>	UDINT	Счетчик ошибок кадров (нарушение формата, предусмотренного спецификацией) при выполнении приёма
<i>rx_double_frames</i>	UDINT	Счетчик принятых кадров дублем
<i>tx_cnt</i>	UDINT	Счетчик переданных кадров
<i>tx_bad_frames</i>	UDINT	Счетчик ошибок по передачи кадров
<i>tx_double_frames</i>	UDINT	Счетчик переданных кадров дублем
<i>libstat</i> – статистика работы библиотеки канала (тип EleSyTypes.libstat):		
<i>rx_overflow</i>	UDINT	Счетчик переполнения входной передачи
<i>tx_overflow</i>	UDINT	Счетчик переполнения выходной передачи

При выборе типа синхронизации (см. 3.10.2) на закладке «Соотнесение входов/выходов» модуля **tsync** появляется дополнительная группа выходных диагностических сигналов в зависимости от выбранного источника времени:

- *NTP* – выходной сигнал *DiagNTP_X*, где *X* – номер NTP-клиента. Структура сигнала *DiagNTP_X* представлена на рисунке 3.59.

Переменная	Соотнесение	Канал	Адрес	Тип	Описание
DynamicChannels					
		DiagNTP_1	%ID25		DiagNTP_1
		status	%IB100	Enumeration of USINT	Статус сервера NTP
		delay	%ID26	REAL	Задержка ответа сервера, мс
		offset	%ID27	REAL	Разница времени, мс
		jitter	%ID28	REAL	Дисперсия фазы, мс
		stratum	%IB116	USINT(1..16)	Стратум сервера
Diagnostic					

Рисунок 3.59 - Модуль **tsync**. Закладка «Соотнесение входов/выходов». Структура сигнала *DiagNTP_X*

Описание выходного сигнала *DiagNTP_X* модуля **tsync** представлено в таблице 3.53.

Таблица 3.53 – Модуль **tsync**. Выходные сигналы NTP-сервера

Имя	Тип	Описание
<i>status</i>	USINT	Статус NTP-сервера: <ul style="list-style-type: none"> • 0 – работает, • 1 – не обновлялся; • 2 – плохой (непригодный для обновления) сервер; • 3 – является кандидатом
<i>delay</i>	REAL	Задержка ответа сервера, мс. Время между отправкой и получением ответа от сервера
<i>offset</i>	REAL	Разница времени, мс
<i>jitter</i>	REAL	Дисперсия фазы, мс
<i>stratum</i>	USINT	Стратум сервера. Допустимые значения сигнала: от 1 до 16

- *ANYTIME* – входной сигнал *anytime_X* и выходной сигнал *DiagANY_X*, где *X* – номер стороннего источника времени. Структура сигналов представлена на рисунке 3.60.

Переменная	Соот...	Канал	Адрес	Тип	Описание
DynamicChannels					
		anytime_1	%QD0		anytime_1
		seconds	%QD0	DATE_AND_TIME	Количество секунд с 1 января 00:00 1970 года (тип DATE_AND_TIME)
		microseconds	%QD1	UDINT	Количество микросекунд от начала секунды
		DiagANY_1	%ID25		DiagANY_1
		status	%IB100	Enumeration of...	Статус ANY
		delay	%ID26	REAL	Задержка ответа сервера, мс
		offset	%ID27	REAL	Разница времени, мс
		jitter	%ID28	REAL	Дисперсия фазы, мс
		stratum	%IB116	USINT(1..16)	Стратум сервера
Diagnostic					

Рисунок 3.60 - Модуль **tsync**. Закладка «Соотнесение входов/выходов». Структура сигналов стороннего источника времени ANYTIME

Описание входного и выходного сигналов *anytime_X* и *DiagANY_X* модуля **tsync** представлено в таблице 3.54.

Таблица 3.54 – Модуль **tsync**. Входной и выходной сигналы стороннего источника времени

Имя	Тип	Описание
Сигнал <i>anytime_X</i>		
<i>seconds</i>	DATE_AND_TIME	Количество секунд с 1 января 00 : 00 1970 года
<i>microseconds</i>	UDINT	Количество секунд от начала секунды
Сигнал <i>DiagANY_X</i>		
<i>status</i>	USINT	Статус ANYTIME: <ul style="list-style-type: none"> • 0 – работает, • 1 – не обновлялся; • 2 – плохой (непригодный для обновления) сервер; • 3 – является кандидатом
<i>delay</i>	REAL	Задержка ответа сервера, мс. Время между отправкой и получением ответа от сервера
<i>offset</i>	REAL	Разница времени, мс. Смещение времени между контроллером и сервером времени
<i>jitter</i>	REAL	Дисперсия фазы, мс
<i>stratum</i>	USINT	Стратум сервера. Допустимые значения сигнала: от 1 до 16

3.11 Работа с модулем **GSM** для приёма и передачи SMS сообщений

ПЛК в исполнении Элсима-M01-ZZZU-GSM имеет встроенный GSM-модем, который позволяет осуществлять приём и отправку SMS сообщений из задачи пользователя.

Встроенный модем обеспечивает возможность работы в сетях GSM с диапазоном частот 850/900/1800/1900 МГц, что обеспечивает возможность работы с операторами сетей второго поколения (2G, 2.5G) по всему миру. Приём и отправка коротких сообщений возможна из любых сетей, поддерживающих этот сервис (т.е. из сетей 3G, 4G и др.).

Для удобства работы с модулем **GSM** в состав конфигурационного пакета ПЛК входит специально разработанный для него функциональный блок **GSM** (*Device (ELSYMA) → Plc Logic → Application → Менеджер библиотек → ElsyMA_GSM*, см. 2.8.1.2). Вызовы этого ФБ **GSM** позволяют осуществить выдачу необходимых последовательностей управляющих воздействий и АТ-команд с контролем правильности выполнения для безошибочного выполнения всех разрешённых действий.

В этом разделе руководства содержится описание работы с библиотекой ФБ **GSM** и примеры их применения.

ВНИМАНИЕ! Перед использованием выполнить подключения.

Для использования функций модуля **GSM** необходимо выполнить подключения антенны согласно 1.8.5 (см. примечание 1) и установку SIM-карты согласно 1.8.4 (см. примечание 2).

Примечания:

1 При подключении антенны необходимо убедиться, что её частотный диапазон соответствует диапазону используемой сети сотового оператора. Особенно это важно в условиях "плохой видимости станций", т.е. низкого уровня сигнала GSM, связанного с большим удалением от базовой станции или нахождения антенны в области радиотени (работе могут мешать здания, возвышенности, деревья, линии электропередач и т.п.). При необходимости, можно использовать антенну с большим коэффициентом усиления или повторитель соответствующего частотного диапазона. Для диапазона 850/900 МГц модуль соответствует классу мощности "4" (2 Вт), для диапазона 1800/1900 МГц – класс мощности "1" (1 Вт).

2 В связи с относительно высокой стоимостью передачи информации текстовыми сообщениями, рекомендуется использовать SIM-карту со специальным тарифом для передачи SMS или подключенной услугой пакета SMS.

Для успешной работы с модулем **GSM**, без выполнения лишних команд в теле программы ПЛК, перед использованием SIM-карты в модуле **GSM** рекомендуется подготовить её к работе, выполнив следующие действия:

1 Убедиться в наличии разрешённых функций приёма и отправки SMS.

Обычно эти функции в современных сетях для большинства тарифов разрешены "по умолчанию", но для определённых тарифов и операторов связи требуется отдельная активация услуг.

2 Выполнить запрет приёма информационных SMS сотового оператора.

Например, для МТС:

- Подключить услугу "*Запрет приема информационных SMS и SMS/MMS с сайта МТС*";

- Отключить услугу "*Вам звонили*";

- Отключить услугу "*Автоинформирование о балансе*";

- Произвести запрет входящих вызовов (услуга может быть платной).

3 Отключить запрос PIN кода.

Если на SIM-карте установлен PIN код, перед использованием модуля будет необходимо выдать команду "AT+CPIN=pin".

4 Очистить историю SMS-сообщений, находящихся на SIM-карте.

5 Установить адреса центра обработки SMS сотового оператора.

6 Убедиться в наличии средств на SIM-карте, достаточных для работы, и своевременно их пополнять (в процессе работы контроль доступен вызовом AT-команды).

3.11.1 Конфигурирование модуля GSM

Для работы с модулем **GSM** в состав пакета конфигулятора ПЛК включена библиотека **GSMLibrary** (дополнительное имя: **ElsyMA_GSM**). Для применения в задаче пользователя *CoDeSys* из этой библиотеки доступны функциональные блоки **ELSYGSM** и **ELSYGSMgprs** (ФБ **ELSYGSMgprs** доступен в версии библиотеки 3.5.11.15 и выше). Функциональный блок **ELSYGSMgprs** должен использоваться только при конфигурировании работы с **GPRS**. В реализации ФБ отключены возможности управления питанием модуля GSM (управление модулем **GSM** в режиме **GPRS** см. в разделе 3.19.4). Данные функциональные блоки обеспечивают выполнение следующих основных функций:

- инициализация модуля **GSM**;
- отправка AT-команды;
- отправка текстового сообщения (SMS);
- приём текстового сообщения (SMS);
- аппаратный рестарт модуля **GSM**.

Для включения в проект экземпляра функционального блока необходимо в разделе переменных указать переменную с типом **ElsyMA_GSM.ELSYGSM**.

Интерфейс ФБ представлен на рисунке 3.61.

Для каждой функции имеется сигнал активации. Среди входных сигналов сигналами активации являются:

- **controlreset** – для функции "Аппаратный рестарт модуля GSM";
- **controlinit** – для функции "Инициализация модуля GSM";
- **controlsendsms** – для функции "Отправка текстового сообщения (SMS)";
- **controlreadsms** – для функции "Приём текстового сообщения (SMS)";
- **controlat** – для функции "Передача AT-команды".

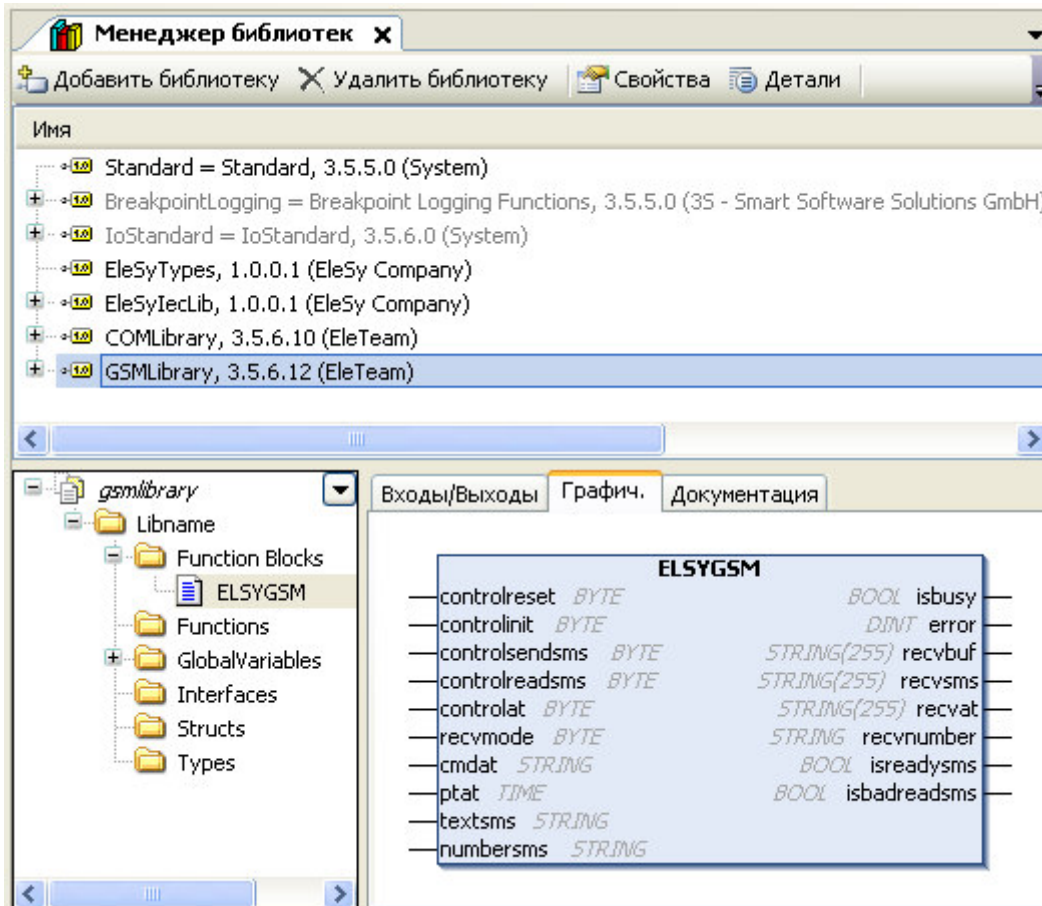


Рисунок 3.61 – Функциональный блок **ELSYGSM**

Для активации функции необходимо присвоить значение "1" для соответствующей управляющей переменной. Для инициализации **GSM** необходимо присвоить значение "1" для переменной **controlinit**.

Внимание! После выполнения функции "Аппаратный рестарт модуля GSM" для корректной работы требуется выполнить функцию "Инициализация модуля GSM".

Кроме управляющих сигналов некоторые функции имеют дополнительные сигналы. Значения дополнительных входных сигналов необходимо обязательно инициализировать до

активации соответствующей функции. Каждая функция имеет свой набор дополнительных сигналов, перечисленных ниже, и на выполнение других функций они влияния не оказывают.

Для функции "**Передача AT-команды**" дополнительными сигналами являются:

- Входные сигналы:
 - *cmdat* – строка длиной до 70 символов, которая должна содержать код выполняемой AT-команды, завершающаяся символами *\$R\$R\$n*. Например, 'AT+*CUSD=1*, "#100#" *\$R\$R\$n*';
 - *ptat* – тайм-аут выполнения команды. Время, по истечении которого должен быть получен ответ. Тип переменной – *TIME*. Например, для присвоения значения 5 с: *T#5S*;
- Выходные сигналы:
 - *recvat* – ответ на выполненную AT-команду.

Примечание – В стандарте ответ предусмотрен не для всех AT-команд, а некоторые команды могут выполняться десятки секунд (см. руководство версии **1.08** "**SIM800 Series AT Commands Manual**", файл *SIM800_Series_AT_Command_Manual_V1.08.pdf*), поэтому необходимо устанавливать соответствующее значение для сигнала *ptat*. Завершение выполнения команды в любом случае будет по истечении времени тайм-аута.

Для функции "**Отправка текстового сообщения (SMS)**" дополнительными являются входные сигналы:

- *textsms* – текст передаваемого SMS-сообщения длиной до 160 символов, содержащий код печатных ASCII символов из диапазона (32–126), например, 'test message';
- *numbersms* – телефонный номер абонента, для которого предназначены SMS-сообщения, например, '8961XXXXXXX'.

Для функции "**Приём текстового сообщения (SMS)**" дополнительными сигналами являются:

- Входные сигналы:
 - *recvmode* – (BYTE) режим удаления СМС при выполнении приема (0-автоматический, 1-пользователем);
- Выходные сигналы:
 - *isreadysms* – (BOOL) Признак нового сообщения – сигнал, принимающий следующие значения:
 - "*TRUE*" – успешный прием SMS в буфер *recvsms*;
 - "*FALSE*" – нет готовых SMS.
 - *isbadreadsms* – (BOOL) ошибка получения SMS-сообщений – сигнал, принимающий следующие значения:
 - "*TRUE*" – возникла ошибка при чтении;
 - "*FALSE*" – успешный приём сообщения.
 - *recvnumber* – номер абонента в принятом сообщении (если удалось его распознать, т.к. составные SMS и SMS оператора не имеют номера);

□ *recvsms* – текст SMS-сообщения, включая служебную информацию о номере абонента и времени приёма. Длина сообщения до 160 печатных ASCII символов из диапазона (32–126).

П р и м е ч а н и я

1 Возможен приём SMS длиной не более 160 символов. Сообщения, превышающие эту длину, будут переданы оператором связи отдельными SMS без указания номера источника сообщения во второй и последующих частях. В некоторых случаях, при сбоях сети оператора, встречались пропадание частей сообщения или перестановка их местами.

2 Следует обратить внимание, что строковая переменная в *CoDeSys* имеет длину, не превышающую 70 символов, в случае необходимости можно увеличить размер строки для переменной программы пользователя (см. руководство пользователя на систему программирования *CoDeSys*).

3 Формат принятого сообщения в буфере *recvsms* имеет вид: '4,"REC UNREAD","+79XXXXXXXXXX","","16/02/26,15:34:26+24 "\$R\$NTEST SMS 1\$R\$N\$R\$N', в котором:

- 4 – номер буфера в памяти GSM модуля, в который было принято сообщение;
- "REC UNREAD" – признак того, что сообщение ни разу не прочитано из буфера;
- "+79XXXXXXXXXX" – номер отправителя;
- "" – служебное поле (может быть ненулевой длины);
- "16/02/24,12:34:26+24" – дата и время приема сообщения;
- TEST SMS 1 – текст сообщения.

Для функций "*Аппаратный рестарт модуля GSM*" и "*Инициализация модуля GSM*" дополнительных сигналов не предусмотрено.

Существует два выходных сигнала, являющихся общими для ФБ, и их значения устанавливаются при выполнении любой функции ФБ:

□ *isbusy* – сигнал занятого ФБ. Устанавливается в "1" (*TRUE*) при активации любой команды. Сбрасывается в "0" (*FALSE*) автоматически, после завершения выполнения команды (успешной или по ошибке). При значении "1" этого сигнала новые команды активировать запрещается;

□ *error* – код ошибки выполнения команды ФБ модуля **GSM**.

Возможны следующие коды ошибок:

- "0" – нет ошибки выполнения команды;
- "-1" – при выполнении команды возникла ошибка системного интерфейса;
- "-2" – превышен размер буфера при приеме ответа;
- "-3" – модуль **GSM** не готов к работе. Причиной возникновения ошибки может быть:
 - сбой при получении ответа;
 - отсутствует SIM-карта;
 - не удалось зарегистрироваться в сети из-за слабого сигнала.

В случае возникновения такой ошибки рекомендуется повторная инициализация, после трех сбоев необходимо проверить внешние подключения и SIM-карту;

• "-4" – ошибка выполнения функции "*Передача AT-команды*", за время тайм-аута не получено ни одного ответного символа на команду (для команд без ответа – это нормальная ситуация). Данный код ошибки является предупреждением, сохраняется возможность

Руководство по эксплуатации

работы с ФБ, для детализации причины возникновения данного предупреждения смотри значение выходного сигнала *recvat*;

- "-5" – ошибка выполнения функции "Отправка текстового сообщения (SMS)";
- "-6" – ошибка выполнения функции "Приём текстового сообщения (SMS)".

Если значение сигнала ошибки от выполнения команды (*error*) необходимо для работы программы, то перед активацией ФБ необходимо сохранить это значение в локальной переменной.

Для применения функций в задаче пользователя *CoDeSys* необходимо соблюдать следующий порядок вызова функций:

1 Вызов функции "Инициализация модуля **GSM**".

2 Опционально: "Периодический вызов AT-команды" (например, для контроля баланса денежных средств на счете SIM-карты).

3 Опционально: Вызов функции "Отправка текстового сообщения (SMS)", если необходимо обеспечить отправку данных на сотовый телефон по событиям в программе.

4 Опционально: Если необходимо осуществлять управление логикой выполнения программы по командам через сотовую связь, то выполнять периодический вызов функции "Приём текстового сообщения (SMS)".

В случае сбоя модуля **GSM** (например, из-за помехи или выключения модуля AT-командой) рекомендуется выполнить вызов функции "Аппаратный рестарт модуля **GSM**".

Модуль **GSM** поддерживает большой набор команд в соответствии со спецификациями:

- ITU-T recommendation V.25ter "Serial asynchronous automatic dialling and control";
- GSM 07.07 / 3GPP TS27.007 (ETS 300 916): "Digital cellular telecommunications system (Phase 2+); AT command set for GSM Mobile Equipment (ME)".

Основные команды для работы с GSM-модемом представлены в таблице 3.55.

Таблица 3.55 – Основные AT-команды модуля GSM

Команда	Ответ	Комментарии
1 Проверка доступности интерфейса с модулем GSM		
AT	OK	В случае получения "OK", можно продолжать работу. В случае отсутствия ответа (модуль GSM недоступен), необходимо выполнить "Аппаратный рестарт модуля GSM ".
2 Проверка готовности SIM-карты		
AT+CPIN?	+CPIN: <CODE> OK Пример ответа: +CPIN: READY OK Пример ответа:	Если поле <CODE> в ответе содержит значение "READY" и получен "OK" отдельной строкой, можно продолжать работу. Возможные ошибки: <CODE> = "NOT INSERTED": - отсутствует SIM-карта. <CODE> = "SIM PIN": SIM-карта защищена PIN кодом. <CODE> = "+CPIN: NOT READY" Модуль GSM не готов к работе В случае получения ошибки для

Таблица 3.55 – Основные AT-команды модуля GSM

Команда	Ответ	Комментарии
	ERROR	продолжения работы необходимо её устранить.
3 Проверка состояния модуля		
AT+CPAS	+CPAS: 0	Информация о состоянии модуля: 0 – готов к работе; 2 – неизвестно; 3 – входящий звонок; 4 – голосовое соединение
4 Установить время модуля GSM		
AT+CCLK="YY/MM/DD,HH:MM:SS±zz"	OK	Формат "yy/MM/dd,hh:mm:ss±zz", где yy -год, MM-месяц, dd - день, hh - час, mm - минута, ss - секунда, zz - смещение в четверть часовых интервалов относительно "времени по Гринвичу" (GMT). Допустимый диапазон для zz: -47...+48. Для Томска (GMT + 06:00), команда для установки времени будет иметь вид: 'AT+CCLK="15/10/09,09:00:00+24"'
5 Запрос баланса		
AT+CUSD=1,"#100#" (для Мегафон и МТС) AT+CUSD=1,"#102#" (для Билайн ("Вымпелком"))	OK +CUSD: 0, "Balance:15r ", 15	Вид сообщения о сумме баланса зависит от подключенного тарифа и оператора связи
6 Проверка доступности сети		
AT+CREG?	+CREG: A,B Пример ответа: +CREG: 0,1 OK	Тип регистрации сети A: 0 – нет кода регистрации сети; 1 – есть код регистрации сети; 2 – есть код регистрации сети +доп. параметры. Тип регистрации сети B: 0 – не зарегистрирован, поиска сети нет; 1 – зарегистрирован, домашняя сеть; 2 – не зарегистрирован, идёт поиск новой сети; 3 – регистрация отклонена; 4 – неизвестно; 5 – роуминг
7 Запрос уровня сигнала		
AT+CSQ	+CSQ: AA,BB OK	Запрос уровня сигнала. AA – Уровень сигнала (больше значение – лучше): 0 -115 дБ и меньше; 1 -112 дБ; 2...30 -110..-54 дБ; 31 -52 дБ и выше;

Таблица 3.55 – Основные AT-команды модуля GSM

Команда	Ответ	Комментарии
		<p>99 – нет сигнала.</p> <p>ВВ – качество принимаемого сигнала (меньшее значение – лучше)</p> <p>0 – BER < 0.1 %</p> <p>1 – 0,26 % < BER < 0,30 %</p> <p>2 – 0,51 % < BER < 0,64 %</p> <p>3 – 1,00 % < BER < 1,30 %</p> <p>4 – 1,90 % < BER < 2,70 %</p> <p>5 – 3,80 % < BER < 5,40 %</p> <p>6 – 7,60 % < BER < 11,0 %</p> <p>7 – BER > 15,0 %</p>
8 Запрос имени сети оператора GSM		
AT+COPS?	<p>+COPS: 0,0,"Operator Name"</p> <p>Пример ответа: +COPS: 0,0,"MTS RUS"</p>	<p>Запрос имени зарегистрированной сети.</p> <p>Operator Name – Имя сети оператора, через которую производится работа модуля GSM.</p> <p>В примере это сеть "MTS RUS" (Российский МТС)</p>

Последовательность шагов для создания проекта с поддержкой функций передачи данных через сеть GSM в ПЛК должна быть следующей:

1 В дерево устройств *Device* (**ELSYMA**) добавить коммутационную панель **Elsyma_BN** согласно 2.8.1.3.

В составе ПЛК для устройства **Elsyma_BN** автоматически будет создано два модуля:

- **ModuleCP** – основной управляющий модуль ПЛК;
- **ExtModules** – модуль УВВ ПЛК Элсима.

2 В ветку **Elsyma_BN** → **ModuleCP** добавить устройство **ELSYMA_M01_GSM**. Для этого переместить указатель в дереве устройств на **ModuleCP** и в меню добавления устройств выбрать **ELSYMA_M01_GSM** с наибольшей версией (или конкретный номер версии, если этого требует проект) (см. рисунок 3.62).

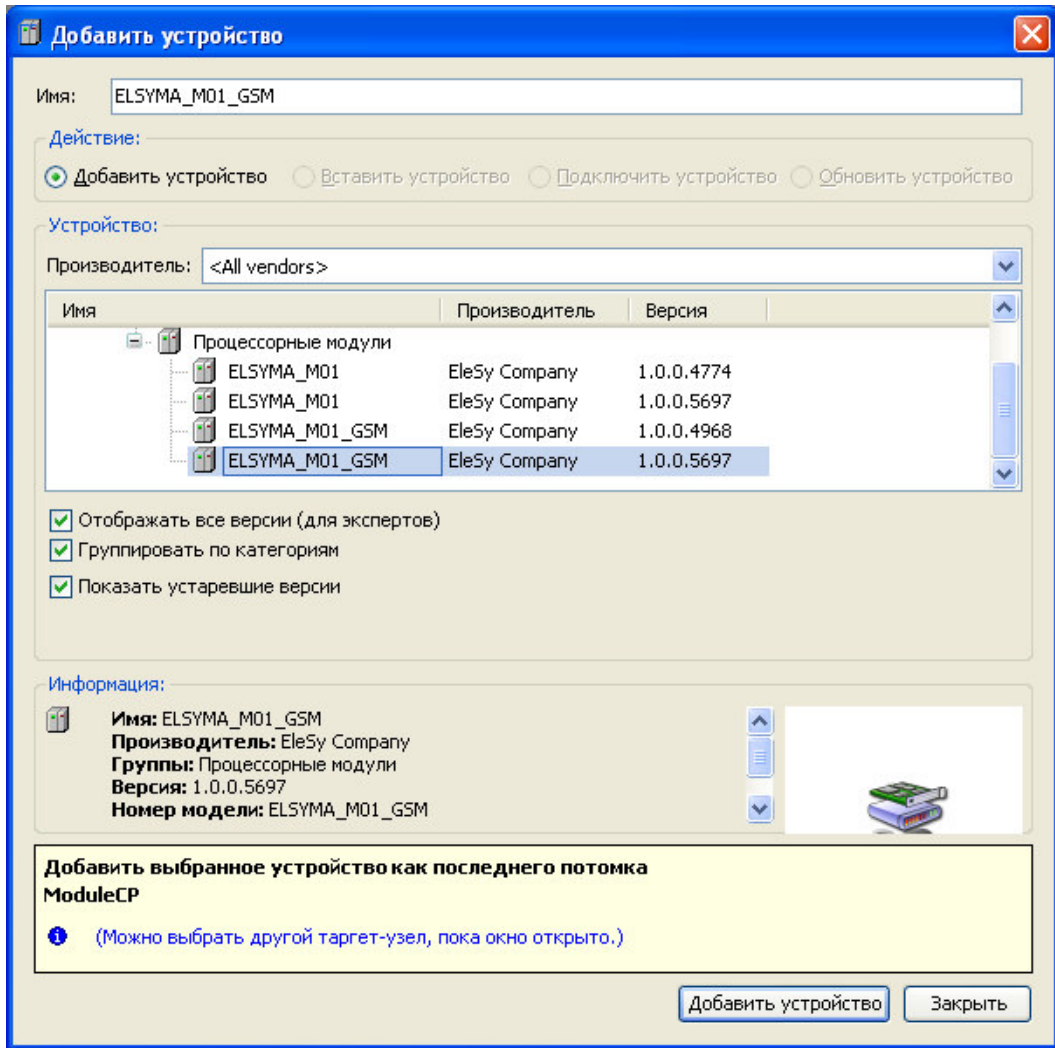


Рисунок 3.62 – Окно выбора типа ПЛК (устройство **ELSYMA_M01_GSM**)

3 В окне проекта для устройства **ELSYMA_M01_GSM** → **GSM** → **SimpleGSM** в первой вкладке "*ElsyMA.GSM Соотнесение входов/выходов*" должно быть три сигнала, представленных на рисунке ниже:

- выходной сигнал **Power** (%QB9);
- выходной сигнал **PowerKey** (%QB10);
- входной сигнал **State** (%IB224).

Необходимо осуществить "соотнесение" этих сигналов с соответствующими сигналами функционального блока, указав соответствующее имя в столбец "Переменная" для каждого канала, как показано на рисунке 3.63:

- "*Application.ElsyMA_GSM.power*" для канала **Power**;
- "*Application.ElsyMA_GSM.powerkey*" для канала **PowerKey**;
- "*Application.ElsyMA_GSM.state*" для канала **State**.

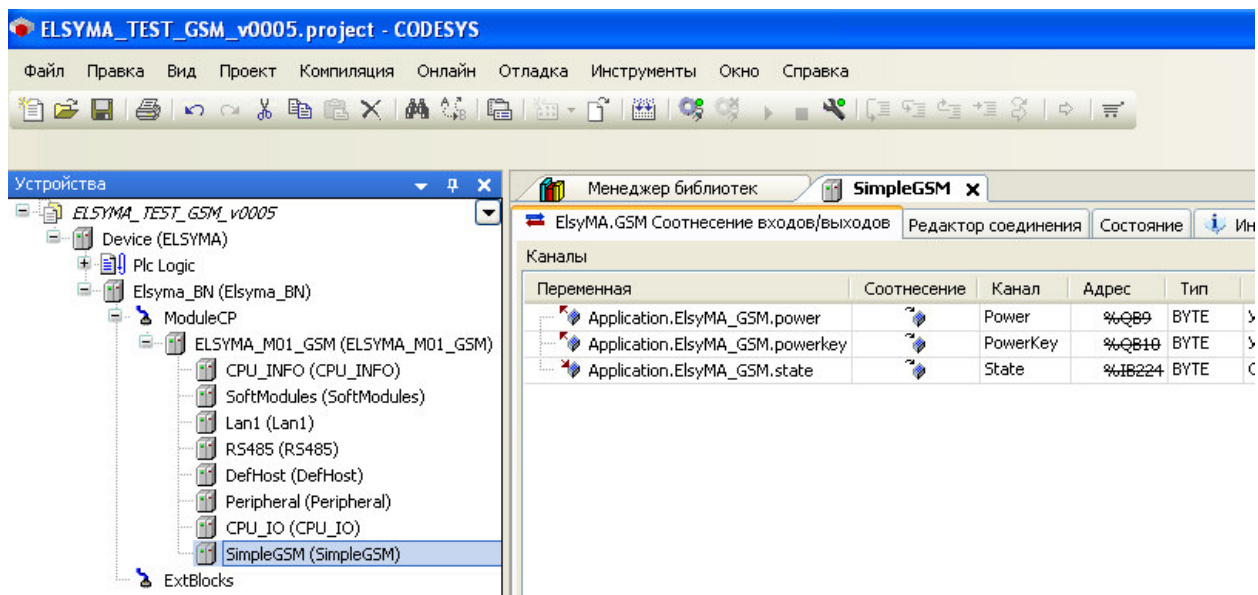


Рисунок 3.63 – Окно назначения сигналов для модуля SimpleGSM

Также возможно управление выходными сигналами и контроль состояния модуля **GSM** через обращение к соответствующим каналам и сигналам из программы пользователя, но для упрощения работы рекомендуется использовать ФБ и указанные сигналы. Ниже приведено описание работы с каналами, если не использовать функциональный блок **GSM**.

Выходной сигнал **Power** (%QB9) служит для управления питанием модуля **GSM**. Установка значения "1" для этого сигнала обеспечивает подачу питания на модуль **GSM** и перевод в состояние "включен". Значение "0" приводит к аппаратному выключению.

По старту приложения *CoDeSys*, значение сигнала соответствует состоянию "выключен" ("0").

После разрешения подачи питания сигналом **Power**, модуль **GSM** включается в "спящем" режиме. В этом режиме сеть GSM недоступна и потребление модуля минимально.

Перевод в рабочее состояние ("активный" режим) осуществляется сигналом **PowerKey** (%QB10). Отрицательный импульс длительностью *I* с переводит модуль из "спящего" режима в "активный".

Входной сигнал **State** позволяет проконтролировать состояние модуля **GSM**.

Значение "1" для этого сигнала означает готовность модуля к работе, т.е. на модуле присутствует стабильное питание и модуль переведён в "активный" режим.

При значении "0" для этого сигнала работа с модулем **GSM** невозможна.

Если в проекте каналы управляются сигналами ФБ (выполнено соотнесение, как на рисунке 3.63), то функция "Инициализации модуля GSM" осуществляет все необходимые действия с сигналами **power** и **powerkey**, соответственно, для пользователя они становятся недоступны.

4 В программе пользователя для ПЛК в ветке "Plc Logic" – "Application" – "PLC_PRG (PRG)" или "Конфигурация задач" – "MainTask" – "PLC_PRG" в секции переменных добавить переменные для работы с ФБ **GSM**.

Обязательно включить в проект экземпляр функционального блока, указав переменную с типом **ElsyMA_GSM.ELSYGSM**.

Кроме этого могут потребоваться дополнительные переменные.

Пример применения функций ФБ **GSM** приведён в приложении И.

Функция "*Инициализации модуля GSM*" обеспечивает подачу питания на модуль **GSM**, переводит его в активный режим, инициализирует системный интерфейс для передачи команд модулю **GSM**, обеспечивает проверку доступности и исправности модуля **GSM**, наличие SIM-карты и доступность сети оператора, выполняя команды:

- "*Проверка готовности SIM-карты*" (команда AT+CPIN?);
- "*Проверка состояния модуля*" (AT+CPAS);
- "*Проверка уровня сигнала*" (AT+CSQ).

Функция "*Передача AT-команды*" требует её подготовки в буфере ".cmdat" и установки длительности тайм-аута получения ответа в переменной ".ptat" (длительность выполнения команд зависит от её типа и может составлять от 0,1 до 30,0 с).

Функция "*Отправка текстового сообщения (SMS)*" требует подготовки в буфере ".textsms" текста сообщения (цифры и латинские буквы) и указания номера телефона в переменной ".numbersms".

Для приёма SMS, в теле программы, необходимо периодически активизировать сигнал ".controlreadsms", разрешающий выполнение функции "*Приём текстового сообщения (SMS)*", и проверять сигналы готовности, кода ошибки и буфер принятого сообщения.

3.12 Интерфейс LAN1

В данном подразделе представлено описание аппаратных интерфейсов, доступных для работы с сетью *Ethernet* в модуле центрального процессора. Доступные аппаратные интерфейсы для работы с *Ethernet* отражаются в дереве устройств в ветке **Lan1** (рисунок 3.64).

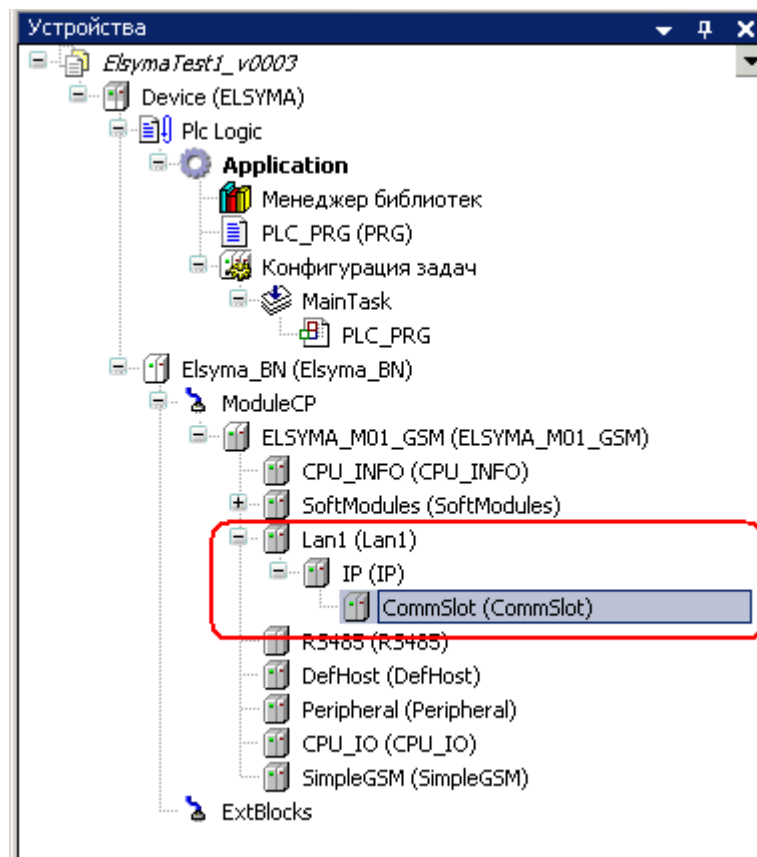


Рисунок 3.64 – Дерево устройств. Аппаратные интерфейсы модуля центрального процессора

Настройка аппаратного интерфейса *Lan1* выполняется с помощью следующих операций:

- 1 Настройка конфигурационных параметров *Lan1* (см. 3.12.1).
- 2 Добавление IP-слота и настройка его параметров (см. 3.12.3).
- 3 Добавление коммуникационного слота *CommSlot* и настройка его параметров (см. 3.12.4).

3.12.1 Настройка интерфейса Ethernet

Для работы с сетью *Ethernet* используется модуль *Lan1* (в других модификациях контроллера возможно наличие нескольких физических интерфейсов *Ethernet*). Настройка аппаратного интерфейса *Lan1* выполняется в системе *CoDeSys*, на закладке просмотра и настройки модуля *Lan1*.

Для настройки аппаратного интерфейса необходимо:

- 1 Задать конфигурационные параметры модуля *Lan1* (см. 3.12.2).
- 2 Добавить IP-слот (см. 3.12.3).
- 3 Добавить коммуникационный слот (см. 3.12.4).

3.12.2 Настройка параметров *Lan1*

Для выполнения операции следует:

- 1 Открыть закладку просмотра и настройки модуля *Lan1*, выделив имя интерфейса в дереве устройств и дважды нажав левую кнопку "мыши".
- 2 Перейти на закладку *Редактор параметров* (рисунок 3.65) и настроить соответствующие параметры модуля.

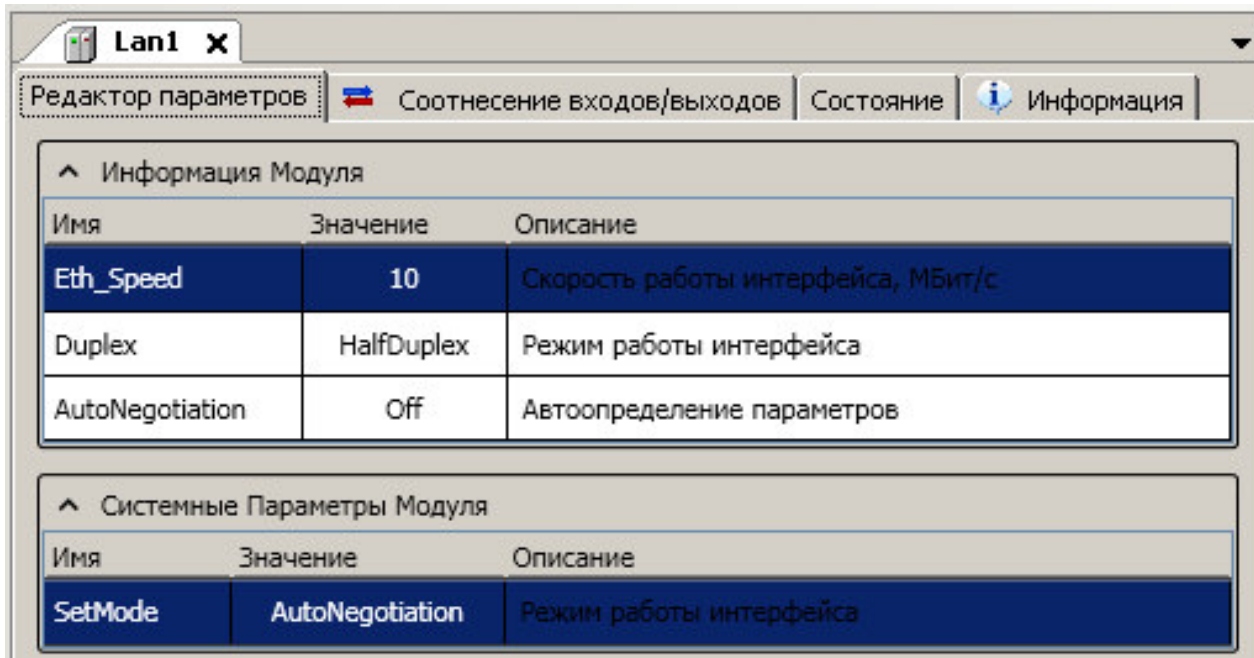


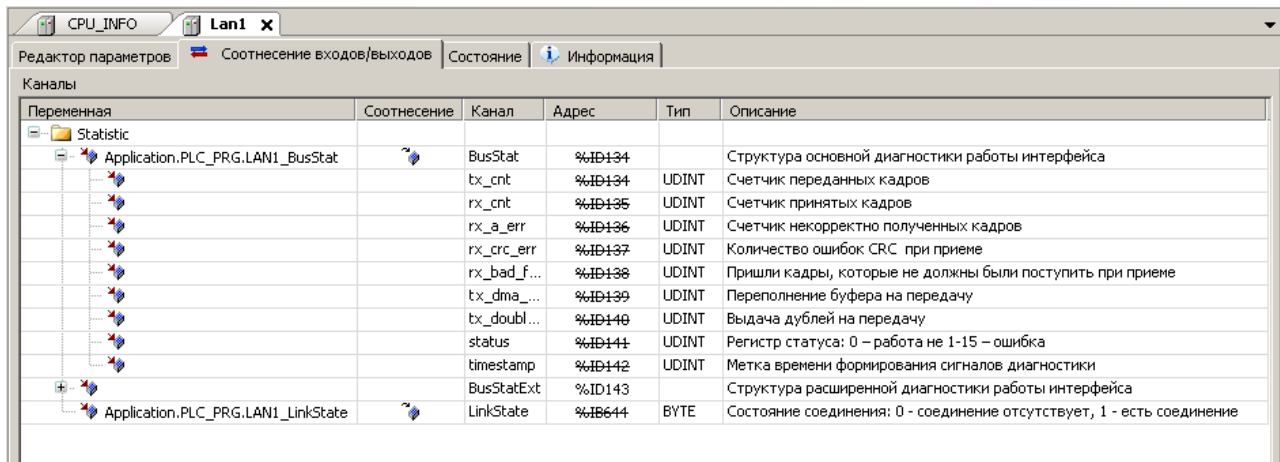
Рисунок 3.65 – Модуль *Lan1*. Закладка *Редактор параметров*

Конфигурационные параметры модуля *Lan1* представлены в таблице 3.56.

Таблица 3.56 – Конфигурационные параметры аппаратного интерфейса Lan_1

Имя параметра	Значение "по умолчанию"	Описание параметра
Информационные параметры		
<i>Eth_Speed</i>	10	Скорость работы интерфейса, Мбит/с
<i>Duplex</i>	HalfDuplex	Режим работы интерфейса
<i>AutoNegotiation</i>	Off	Автоопределение параметров
Системные параметры		
<i>SetMode</i>	AutoNegotiation	Установка режима работы интерфейса. Допустимые значения: <ul style="list-style-type: none"> • <i>Speed10_Half</i>; • <i>Speed10_Full</i>; • <i>Speed100_Half</i>; • <i>Speed100_Full</i>; • <i>AutoNegotiation</i>

На рисунке 3.66 представлен вид закладки *Соотнесение входов/выходов* модуля Lan1 со статистическими параметрами.

Рисунок 3.66 – Модуль Lan1. Закладка *Соотнесение входов/выходов*

В таблице 3.57 представлено описание статистических параметров.

Таблица 3.57 – Статистические параметры программного модуля Lan1

Имя параметра	Тип	Описание параметра
<i>BusStat</i>		Структура основной диагностики работы интерфейса
<i>tx_cnt</i>	UDINT	Счетчик переданных кадров
<i>rx_cnt</i>	UDINT	Счетчик принятых кадров
<i>rx_a_err</i>	UDINT	Счетчик некорректно полученных кадров
<i>rx_crc_err</i>	UDINT	Количество ошибок CRC при приеме
<i>rx_bad_frames</i>	UDINT	Пришли кадры, которые не должны были поступить при приеме
<i>tx_dma_overflow</i>	UDINT	Переполнение буфера на передачу
<i>tx_double_frames</i>	UDINT	Выдача дублей на передачу
<i>status</i>	UDINT	Регистр статуса: 0 – работа, 1-15 – ошибка
<i>timestamp</i>	UDINT	Метка времени формирования сигналов диагностики
<i>BusStatExt</i>		Структура расширенной диагностики работы интерфейса
<i>runt_err</i>	UDINT	Фрагментированный (сбойный) кадр (меньше 64 байтов)
<i>len_err</i>	UDINT	Ошибка длины кадра
<i>out_of_err</i>	UDINT	Счетчик кадров с размером больше допустимого (длина кадра больше максимального размера)
<i>mii_rxe</i>	UDINT	Счетчик некорректно полученных кадров по интерфейсу МП
<i>jabber_tout</i>	UDINT	Тайм-аут передачи сообщений о возникновении ошибки

Таблица 3.57 – Статистические параметры программного модуля *Lan1*

Имя параметра	Тип	Описание параметра
<i>loss_of_carrier</i>	UDINT	Потеря несущей
<i>late_collision</i>	UDINT	Поздняя коллизия (за пределами временного окна коллизии)
<i>frame_underflow</i>	UDINT	Обнуление кадра данных
<i>excessive_deferral</i>	UDINT	Избыточная отсрочка
<i>excessive_collision</i>	UDINT	Избыточная коллизия
<i>dribble_bit_err</i>	UDINT	Битовая ошибка
<i>mii_err</i>	UDINT	Ошибка интерфейса МП
<i>broadcast_cnt</i>	UDINT	Счетчик широковещательных пакетов данных
<i>min_time</i>	UDINT	Минимальное зафиксированное время в период от постановки кадра в DMA до получения подтверждения на него
<i>avg_time</i>	UDINT	Среднее зафиксированное время в период от постановки кадра в DMA до получения подтверждения на него
<i>max_time</i>	UDINT	Максимальное зафиксированное время в период от постановки кадра в DMA до получения подтверждения на него
<i>status</i>	UDINT	Регистр статуса: 0 – работа, 1-15 – ошибка
<i>timestamp</i>	UDINT	Метка времени формирования сигналов диагностики
<i>LinkState</i>	BYTE	Состояние физического соединения: 0 – соединение отсутствует, 1 – соединение установлено

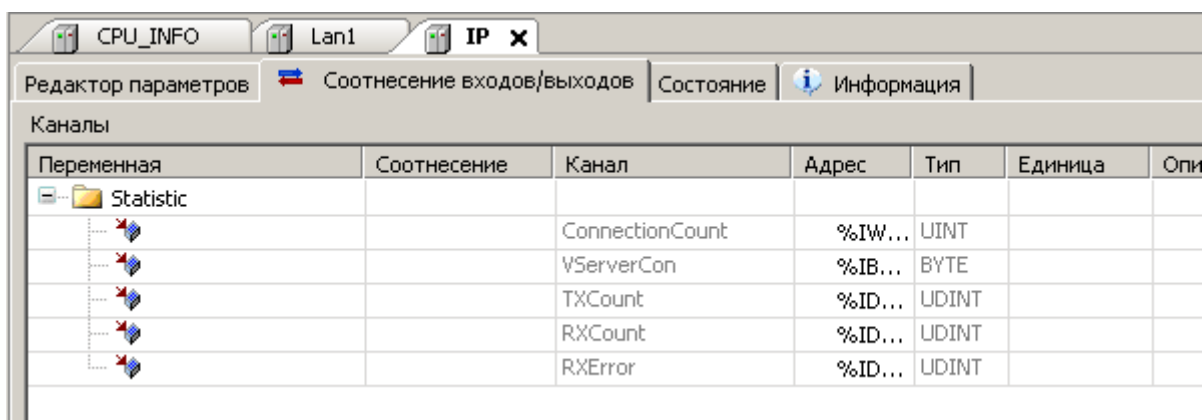
3.12.3 Добавление IP-слота и настройка его параметров

В дереве устройств в ветке *LAN1* существует возможность добавить *IP* для слота с помощью команды *Добавить устройство...* контекстного меню. Системные параметры IP-слота приведены в таблице 3.58.

Таблица 3.58 – Системные параметры IP-слота

Имя параметра	Тип	Описание параметра
<i>IP-Address</i>	STRING	IP-адрес порта
<i>Mask</i>	STRING	Сетевая маска
<i>Gateway</i>	STRING	Адрес шлюза

На рисунке 3.67 представлен вид закладки *Соотнесение входов/выходов* IP-слота со статистическими сигналами, отображающими состояние IP-слота.

Рисунок 3.67 – Модуль IP. Закладка *Соотнесение входов/выходов*

3.12.4 Добавление коммуникационного слота *CommSlot*

В дереве устройств в ветке *IP* возможно добавить коммуникационный слот *CommSlot* с помощью команды *Добавить устройство...* контекстного меню. Системные параметры слота *CommSlot* представлены в таблице 3.59.

Таблица 3.59 – Системные параметры *CommSlot*

Имя параметра	Значение "по умолчанию"	Описание параметра
<i>NonBlock</i>	0	Режим работы Socket
<i>KeepAlive</i>	1	Признак включения службы KEEPALIVE
<i>KeepCnt</i>	3	Количество попыток передачи пакетов KEEPALIVE
<i>KeepIdle</i>	10	Интервал времени, по истечении которого начинается передача пакетов KEEPALIVE
<i>KeepIntvl</i>	10	Интервал времени между выдачей пакетов KEEPALIVE, с
<i>SynCnt</i>	3	Число попыток передачи пакета SYN перед разрывом соединения

Конфигурационные параметры слота *CommSlot* представлены в таблице 3.60.

Таблица 3.60 – Конфигурационные параметры *CommSlot*

Имя параметра	Значение "по умолчанию"	Описание параметра
<i>Port</i>	502	Порт заданного коммуникационного слота
<i>MaxConnections</i>	4	Количество разрешённых соединений для серверных каналов. Диапазон значений – от 1 до 4
<i>ModeTransport</i>	TCP Server	Режим работы транспортного уровня. Возможные значения: <ul style="list-style-type: none"> • 0 – TCP Server; • 1 – TCP Client; • 2 – UDP Server; • 3 – UDP Client
<i>SendBuff</i>	4096	Размер буфера хранения перед отправкой. Диапазон допустимых значений – от 0 до 4294967295
<i>TCP_NoDelay</i>	4	Отключение алгоритма Нагеля для обеспечения передачи данных без ожидания заполнения пакета данными

На рисунке 3.68 представлен вид закладки *Редактор параметров* коммуникационного слота *CommSlot* с конфигурационными и системными параметрами модуля.

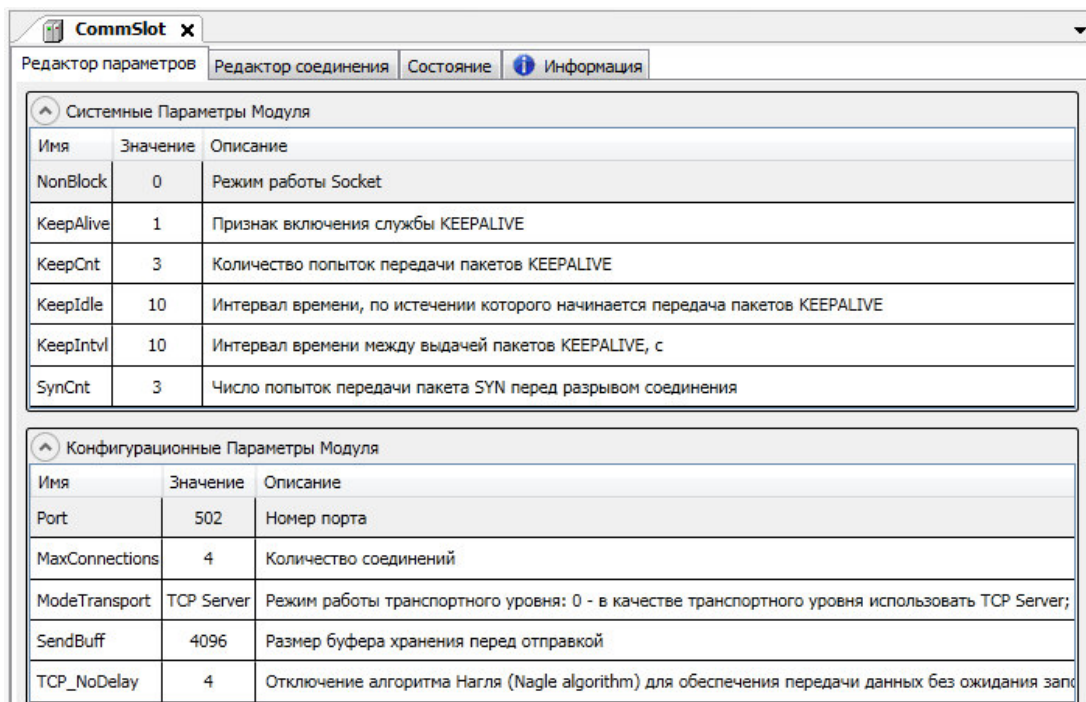


Рисунок 3.68 – Слот *CommSlot*. Закладка *Редактор параметров*

Для присвоения IP-коммуникационного слота программным модулям, через которые должен быть осуществлен доступ к сигналам контроллера, на закладке **Редактор соединения** (рисунок 3.69) в области настройки и просмотра параметров слота **CommSlot** выбрать в выпадающем списке **Server**: необходимый программный модуль.

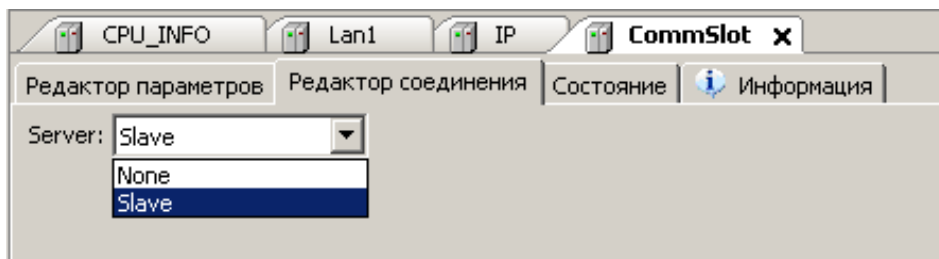


Рисунок 3.69 – Конфигурационные параметры **CommSlot**. Закладка **Редактор соединения**

ВНИМАНИЕ! В данной версии сервисной программы реализована работа с модулем **Modbus-Server TCP/IP – ModBusServer (Slave)**. Для присвоения IP-коммуникационного слота программному модулю **Modbus-Server TCP/IP** необходимо выполнить следующие действия:

1 На закладке **Редактор соединения** в выпадающем списке **Server**: выбрать значение – **Slave**.

2 На закладке **Редактор параметров** для модуля **CommSlot** задать номер порта – 502 и количество соединений – 4.

3 На закладке **Редактор параметров** для модуля **IP** заполнить IP-адрес, сетевую маску и адрес шлюза (см. 3.12.3).

3.13 Управление сетевой подсистемой контроллера (коннектор NetControl)

В контроллере реализована возможность конфигурирования и управления сетевой подсистемой, в частности настройки правил преобразования сетевых адресов. Данная возможность осуществляется с помощью коннектора **NetControl**, который позволяет расширить функциональность контроллера при работе с различными протоколами.

Для работы с необходимыми функциями к коннектору **NetControl** необходимо добавить устройство **StaticRoute** и настроить соответствующие параметры.

3.13.1 Задание статической маршрутизации сетевых интерфейсов (StaticRoute)

При добавлении в дерево конфигурации устройства **StaticRoute** в контроллере активируется функция конфигурирования и управления сетевой подсистемой, в частности настройки статической маршрутизации сетевых данных. Маршруты указываются в явном виде при конфигурации маршрутизатора. Вся маршрутизация при этом происходит без участия каких-либо протоколов маршрутизации.

К основным функциям статической маршрутизации относятся:

- обеспечение возможности более гибкого управления сетевой подсистемой;
- обеспечение возможности задания в явном виде маршрутов передачи данных в сети Ethernet.

Допустим, что в контроллере необходимо реализовать задачу, схема которой приведена на рисунке 3.70: необходимо обеспечить связь **Modbus Master** и подчиненных устройств, при условии нахождения в разных подсетях, посредством интерфейса **GPRS** и интерфейса **LANI**; если устройство **Master** и **Slave** находятся в одной подсети, то связь устанавливается напрямую (без шлюза связи).

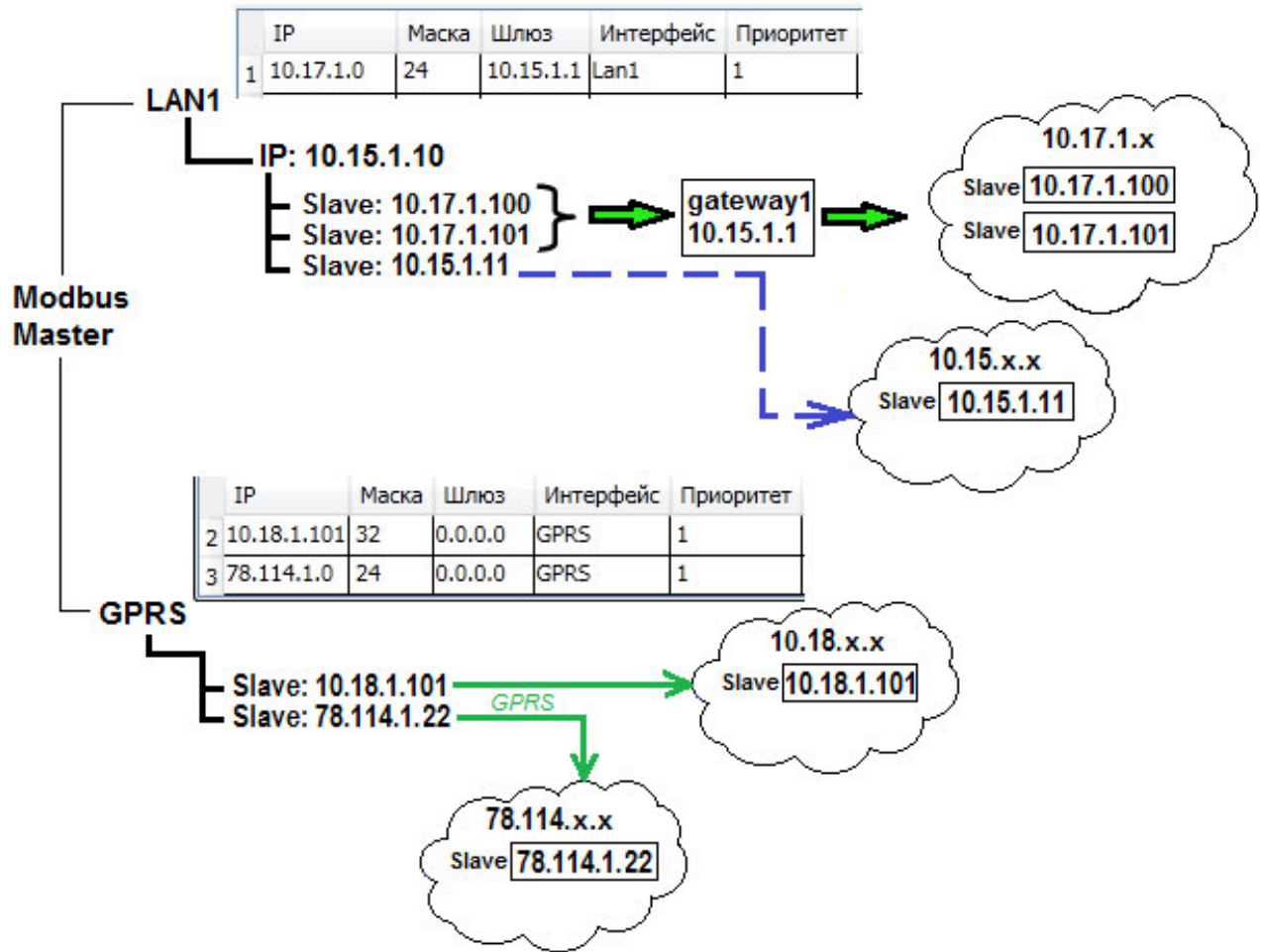


Рисунок 3.70 – Схематичное представление статической маршрутизации

На рисунке 3.71 представлен пример итогового дерева конфигурации для реализации приведенной выше задачи. В дерево добавлены устройство для конфигурирования таблицы маршрутов **StaticRoute**, устройство **GPRS_Init** и его коммуникационные слоты **CommSlot**, устройство **IP** и его коммуникационные слоты **CommSlot**, устройство поддержки протокола **MBTCPM** и соответствующие **Slave**-устройства, каждое из которых связано с коммуникационным слотом **CommSlot**. При конфигурировании во вкладках **Редактор соединения** устройств **CommSlot** необходимо задать **Сервер** в соответствии с IP адресами **Slave**-устройств (значение IP адресов устройств прописаны в именах).

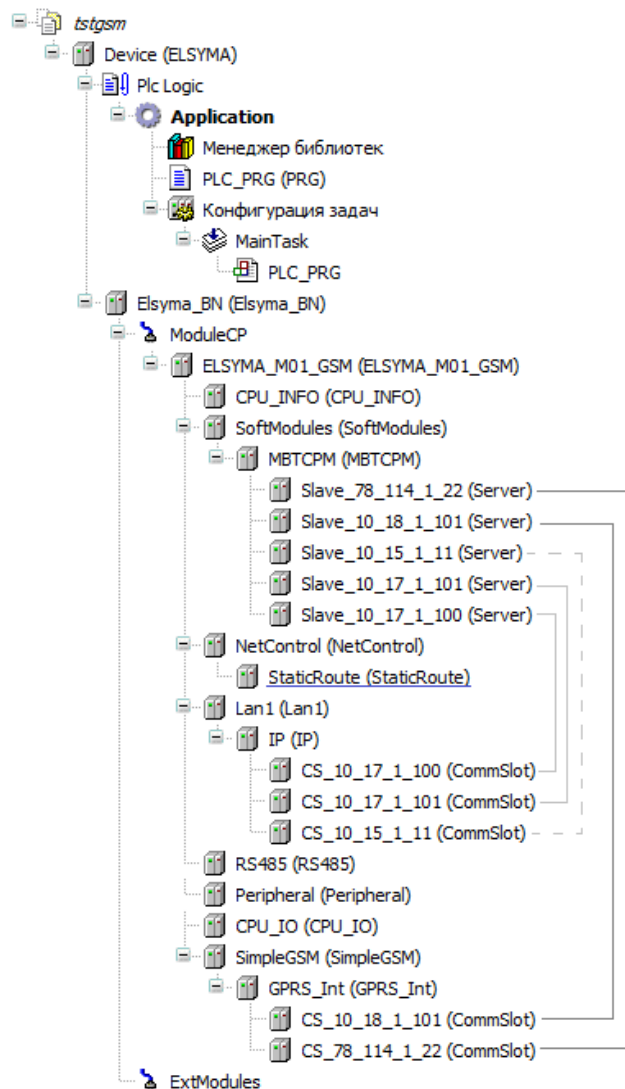
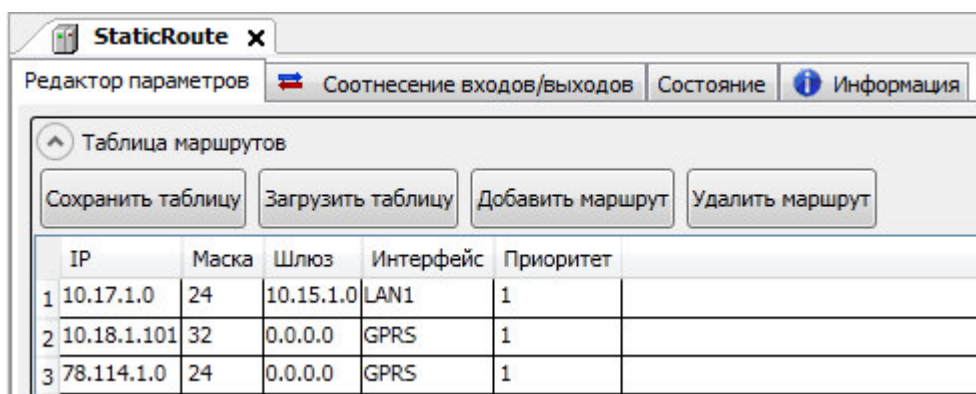


Рисунок 3.71 – Дерево конфигурации проекта

Примечание: в дерево устройств можно добавить только одно устройство *StaticRoute*.

Для реализации схемы, приведенной на рисунке 3.70, необходимо во вкладке *Редактор параметров* устройства **StaticRoute** задать в *таблице маршрутов* соответствующие параметры (рисунок 3.72).

Рисунок 3.72 – Устройство StaticRoute. Вкладка *Редактор параметров*. *Таблица маршрутов*

Пользователь имеет возможность задать до **32-х** статических маршрутов. Каждая запись таблицы маршрутов имеет следующие параметры:

- **IP** – адрес сети, на которую маршрутизируется трафик.

- **Маска** – маска сети, на которую маршрутизируется трафик.
- **Шлюз** – адрес шлюза, который способствует дальнейшей маршрутизации (или подключен к маршрутизируемой сети напрямую).
- **Интерфейс** – интерфейс, на который следует направить трафик сети и указать дополнительные условия, согласно которым выбирается маршрут (при выборе в таблице **StaticRoute** интерфейса **GPRS** всегда следует устанавливать шлюз 0.0.0.0).
- **Приоритет** – число переходов до места назначения. При наличии нескольких маршрутов на одну и ту же сеть некоторые маршрутизаторы выбирают маршрут с минимальным приоритетом.

IP можно задать двумя способами: в явном виде и в виде подсети. В таблице маршрутов (рисунок 3.72) в соответствии с задачей, приведенной на рисунке 3.70, для интерфейса **LAN1** адрес был задан в виде подсети (маска 24), для интерфейса **GPRS** – в виде подсети и в явном виде (конкретно заданный адрес, маска 32).

Примечание – При задании IP-адреса необходимо проверить соответствие заданного IP-адреса и заданной маски подсети.

Параметр **IP** может принимать значение «*default*» (рисунок 3.73). При задании данного значения пакет отправляется в том случае, если маршрут к сети назначения пакета неизвестен (не задан явным образом в таблице маршрутизации).

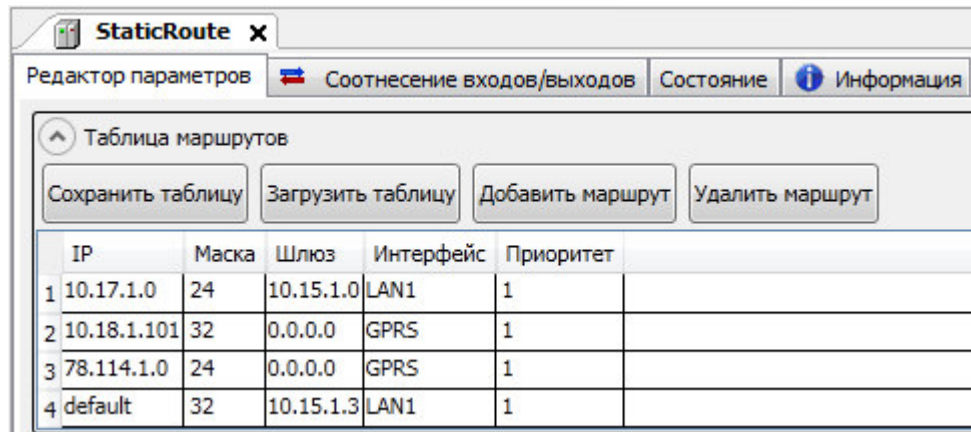


Рисунок 3.73 – Устройство StaticRoute. Вкладка *Редактор параметров*. *Таблица маршрутов*

Над таблицей маршрутов расположены четыре кнопки.

Кнопка «**Добавить маршрут**» позволяет пользователю добавить в таблицу статический маршрут.

Удалить выделенный статический маршрут можно с помощью кнопки «**Удалить маршрут**».

При нажатии на кнопку «**Сохранить таблицу**» пользователь сохраняет заданные статические маршруты. Файл с таблицей маршрутов сохраняется в формате **.txt**; формат записи строки маршрута в файле – '*IP;Маска;Шлюз;Интерфейс;Приоритет*' (например, '*10.14.0.1;32;10.14.0.1;Lan1;0*').

С помощью кнопки «**Загрузить таблицу**» пользователь имеет возможность загрузить статические маршруты из текстового файла сгенерированного ранее программно (при помощи функции «**Сохранить таблицу**») или созданного пользователем с помощью текстового редактора в соответствии с форматом строки маршрута '*IP;Маска;Шлюз;Интерфейс;Приоритет*'.

Сигналы устройства **StaticRoute** формируются во вкладке **Соотнесение входов\выходов** (рисунок 3.74) в соответствии с таблицей 3.61.

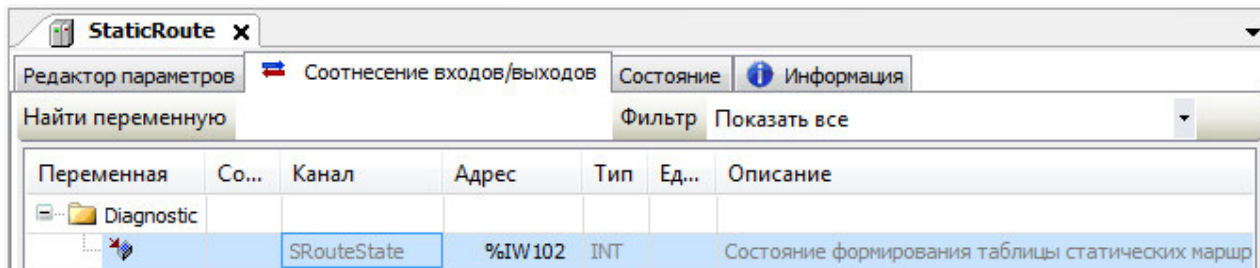
Рисунок 3.74 – Устройство StaticRoute. Вкладка *Соотнесение входов\выходов*

Таблица 3.61 – Диагностические сигналы устройства StaticRoute

Имя параметра	Тип	Значение по умолчанию	Описание параметра
Диагностические сигналы модуля			
<i>SRouteState</i>	<i>Int</i>	0	Состояние формирования таблицы статических маршрутов. Возвращает следующие значения: <i>1</i> – успешная установка таблицы статической маршрутизации; <i>-1</i> – ошибка.

В случае возникновения ошибки *-1* в log системы (журнал *CoDeSys*) выводится сообщение «**StaticRoute: Parameter RoutesCount do not EQ of the project**».

В случае возникновения ошибки *-2* в log системы выводится сообщение «**StaticRoute: ip gw prio is not valid**» (на каждую запись).

При успешном добавлении записи в статической маршрутизации в log системы выводится сообщение: «**StaticRoute: ip gw prio was added**» (на каждую запись).

3.14 Настройка интерфейса RS-485

Для работы с сетью *RS-485* используется модуль **RS485** (в других модификациях контроллера возможно наличие нескольких физических интерфейсов *RS-485* или *RS-232*). Настройка аппаратного интерфейса *RS-485* выполняется в системе *CoDeSys*, на закладке просмотра и настройки модуля **RS485**.

Для настройки аппаратного интерфейса необходимо:

- 1 Задать конфигурационные параметры модуля **RS485** (см. 3.14.1).
- 2 Выбрать устройство для работы через данное соединение (см. 3.14.2).

3.14.1 Настройка параметров модуля RS485

Для выполнения операции следует:

1 Открыть закладку просмотра и настройки модуля **RS485**, выделив имя интерфейса в дереве устройств и дважды нажав левую кнопку "мышь".

2 Перейти на закладку *Редактор параметров* (рисунок 3.75) и настроить соответствующие параметры модуля.

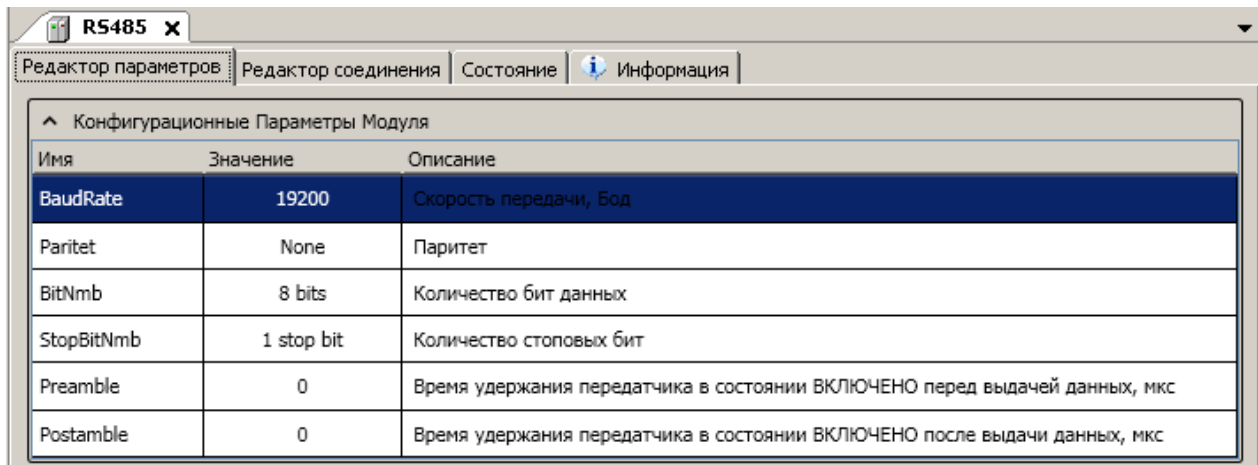


Рисунок 3.75 – Модуль RS485. Закладка Редактор параметров

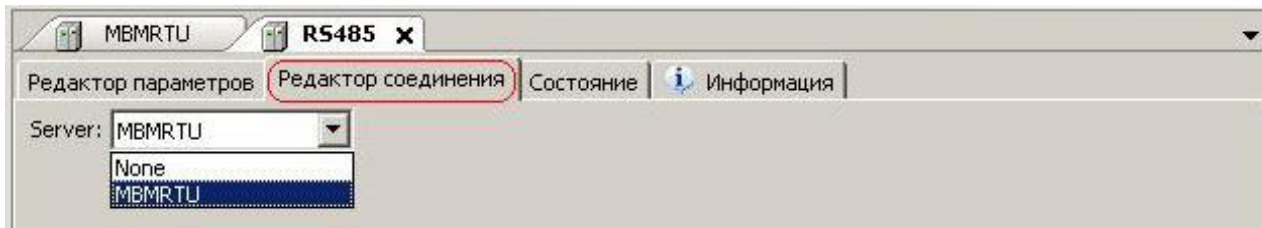
Конфигурационные параметры модуля RS485 представлены в таблице 3.62.

Таблица 3.62 – Конфигурационные параметры модуля RS485

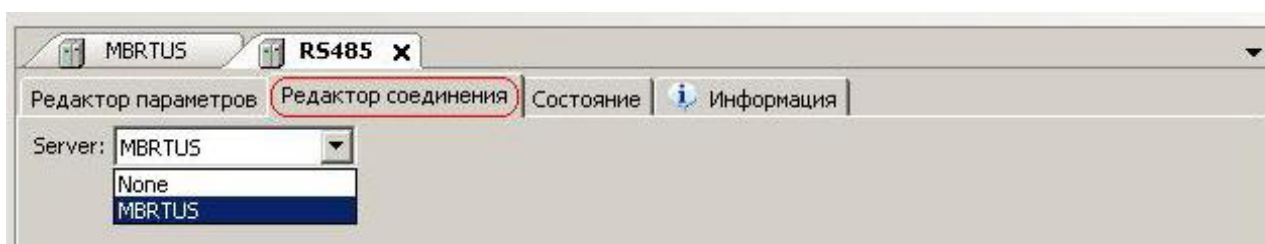
Имя параметра	Значение "по умолчанию"	Описание параметра
BaudRate	19200	Скорость передачи данных, бит/с <ul style="list-style-type: none"> • 300; • 600; • 1200; • 2400; • 4800; • 9600; • 19200; • 28800; • 38400; • 57600; • 115200
Paritet	None	Паритет: <ul style="list-style-type: none"> • None; • Odd; • Even
BitNmb	8 bits	Количество бит данных: <ul style="list-style-type: none"> • 5 bits; • 6 bits; • 7 bits; • 8 bits
StopBitNmb	1 stop bit	Количество стоповых бит данных: <ul style="list-style-type: none"> • 1 stop bis; • 2 stops bis
Preamble	0	Время удержания передатчика в состоянии "Включено" перед выдачей данных, мкс. Диапазон допустимых значений – от 0 до 65535
Postamble	0	Время удержания передатчика в состоянии "Включено" после выдачи данных, мкс. Диапазон допустимых значений – от 0 до 65535

3.14.2 Связывание интерфейса RS-485 с программным модулем

Для связывания коммуникационного слота с программным модулем, через который должен осуществляться доступ к сигналам контроллера, на закладке *Редактор соединения* в выпадающем списке *Server:* выбрать необходимый программный модуль (например, **MBMRTU**, **MBRTUS**, **ELMicronMst** или т.п.) (см. рисунок 3.76).



а) Закладка *Редактор соединения* для программного модуля **MBMRTU**



б) Закладка *Редактор соединения* для программного модуля **MBRTUS**

Рисунок 3.76 – Конфигурационные параметры **RS485**. Закладка *Редактор соединения*

3.15 Периферийные устройства

В данном подразделе представлено описание периферийных устройств. Доступные периферийные устройства отражаются в дереве устройств в ветке *Peripheral*.

На рисунке 3.77 представлен вид закладки *Соотнесение входов/выходов* периферийных устройств.

Переменная	Соотнесение	Канал	Адрес	Тип	Описание
Application.PLC_PRG.myDipSwitch		DipSwitch	%IB...	BYTE	Состояние DipSwitch (первые 4 бита)
		SDOn	%IB...	BYTE	Признак наличия SD карты в слоте (0 - не установлена; 1 - установлена)

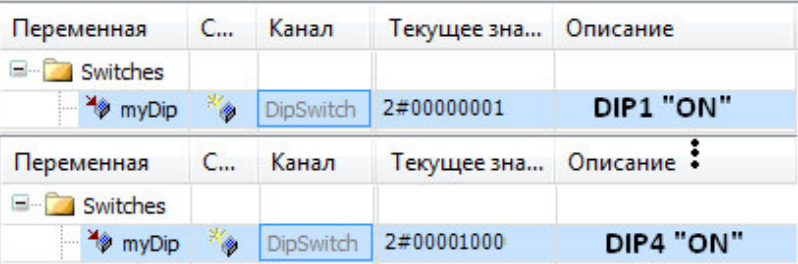
Рисунок 3.77 – Периферийные устройства. Закладка *Соотнесение входов/выходов*

Модуль поддерживает набор сигналов для отображения состояния переключателей контроллера и сигнал наличия установленной SD-карты в слот (таблица 3.63).

Таблица 3.63 – Модуль **Peripheral**. Данные модуля

Имя	Тип	Описание
Switches – состояние переключателей контроллера		
<i>DipSwitch</i>	BYTE	<p>Отображает состояние переключателя <i>DipSwitch</i> (первые 4 бита переключателя SW). Положение "ON" переключателя соответствует установленному биту. Соответствие бит и номера переключателя:</p> <ul style="list-style-type: none"> • <i>bit 0</i> – соответствует переключателю DIP1; • <i>bit 1</i> – соответствует переключателю DIP2; • <i>bit 2</i> – соответствует переключателю DIP3; • <i>bit 3</i> – соответствует переключателю DIP4 • <i>bit 4</i> – соответствует кнопке «R»: <p>1 – кнопка нажата, 0 – кнопка не нажата</p>

Таблица 3.63 – Модуль Peripheral. Данные модуля

Имя	Тип	Описание
		
<i>SDOn</i>	BYTE	<p>Отображает наличие SD-карты в слоте:</p> <ul style="list-style-type: none"> • 1 – карта не установлена; • 2 – карта установлена

3.16 Работа с SD-картой и USB-Flash

SD-карта памяти может использоваться для записи и хранения необходимых пользователю данных (п. 3.16.1), для чтения информации (п. 3.16.2), а также для записи отладочной информации в процессе работы контроллера (п. 3.16.3). Аналогичные функции выполняет USB-Flash (п. 3.16.4 - 3.16.6). Для работы с контроллером существует возможность использования карты памяти *MicroSD* и *USB-Flash* с объемом от 2 до 32 Гбайт. Перед началом работы с SD-картой или USB-Flash необходимо выполнить форматирование устройств в файловую систему FAT.

3.16.1 Доступ к SD-карте через CoDeSys

Доступ к SD-карте контроллера может осуществляться через *CoDeSys*. С помощью среды программирования можно записывать файлы с компьютера на SD-карту, а также удалять лишнюю информацию.

Для работы с SD-картой через *CoDeSys* необходимо:

1 Установить SD-карту. Для этого необходимо отключить питание контроллера и установить SD-карту в слот контроллера так, чтобы раздался характерный щелчок.

ВНИМАНИЕ! Перед извлечением и перед установкой SD-карты в слот контроллера, предварительно необходимо отключить питание контроллера. В ходе работы контроллера запрещается извлекать SD-карту ввиду потери данных незавершенных операций обмена.

2 В дереве устройств выбрать устройство *Device* и дважды нажать на него левой кнопкой мыши. В появившейся закладке найти вкладку **Файлы** (рисунок 3.78). Левое окно вкладки отображает данные на компьютере, правое окно – содержание памяти контроллера.

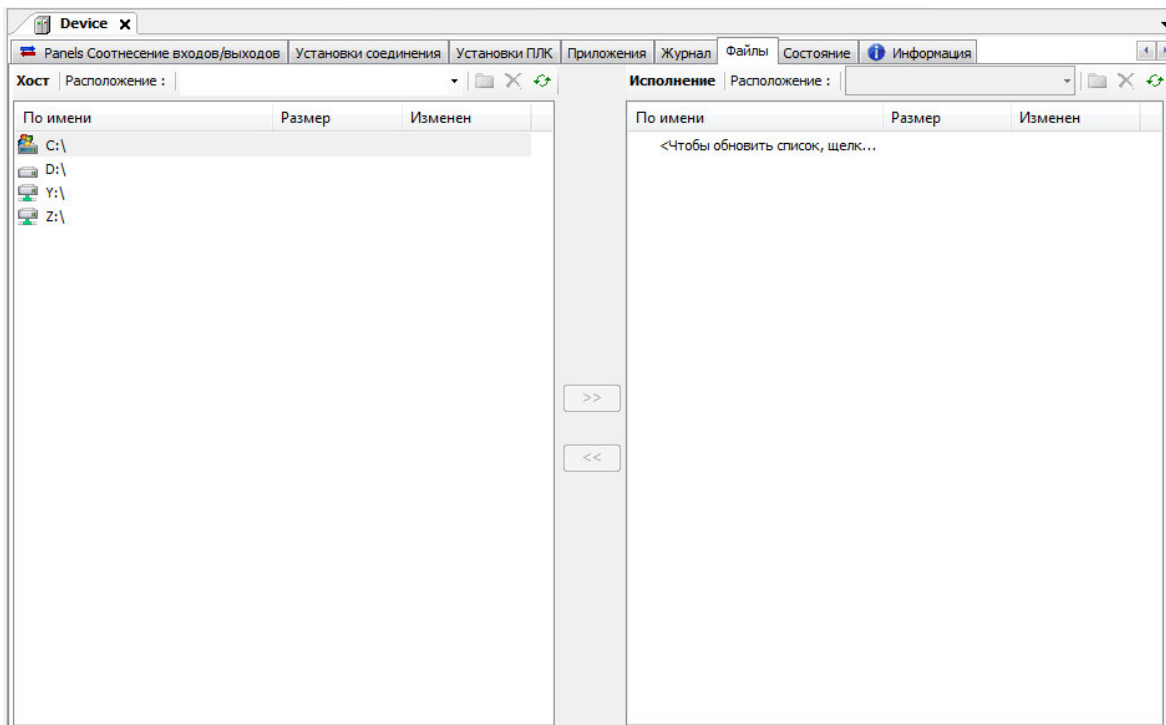





Рисунок 3.78 – Вкладка **Файлы**. Начало работы с SD-картой в среде CoDeSys

3 Обновить информацию памяти контроллера, нажав на кнопку , расположенную в правом верхнем углу. После обновления в правом окне вкладки **Файлы** отображаются данные из памяти контроллера (рисунок 3.79), кнопки  – «Новая директория» и  – «Удалить элемент» становятся активными.

4 Откройте папку «**\$SDDDev\$**» (рисунок 3.79). Данная папка отвечает за содержание SD-карты.

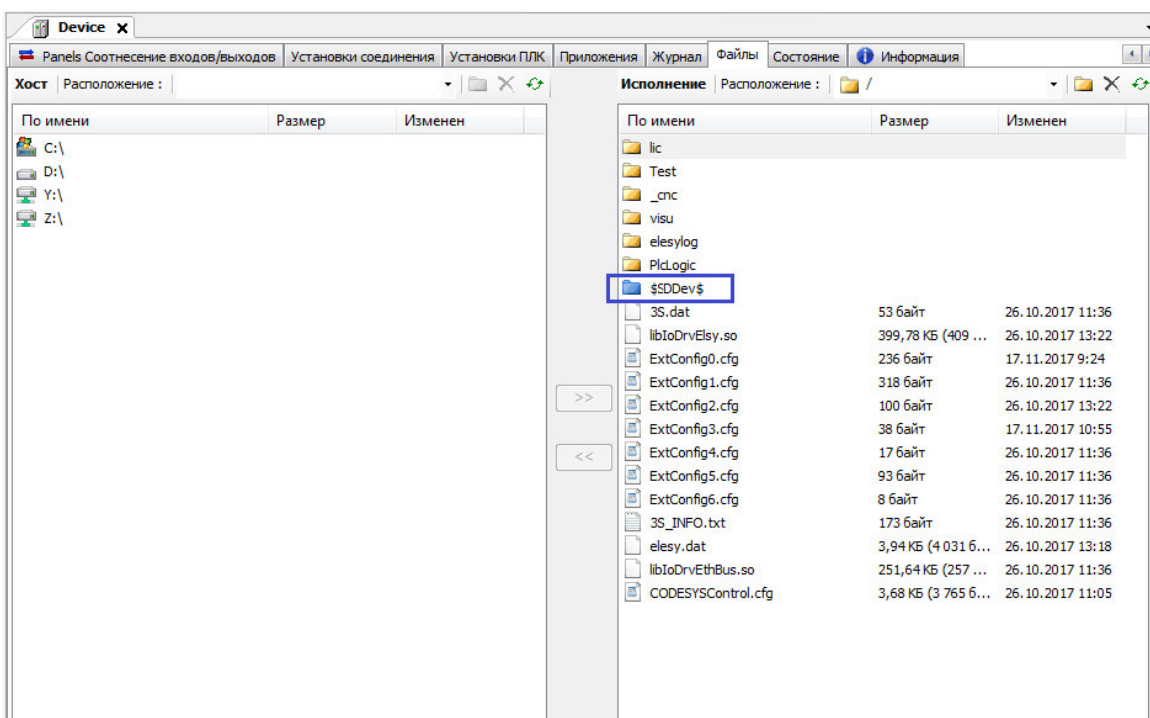



Рисунок 3.79 – Вкладка **Файлы**. Отображение содержания памяти контроллера после обновления

5 Чтобы создать папку на карте памяти необходимо нажать на кнопку . В появившемся окне «Новая директория» (рисунок 3.80) ввести имя директории.

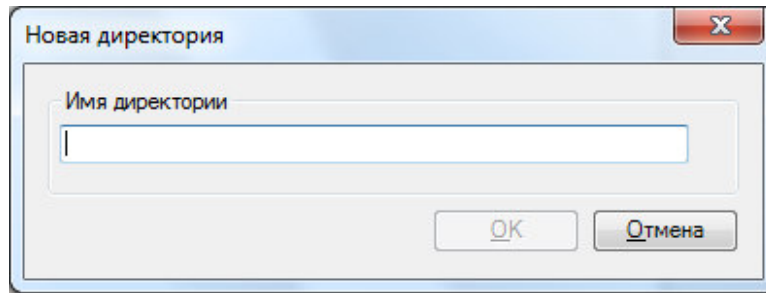



Рисунок 3.80 – Окно «Новая директория»

6 Чтобы копировать элемент из компьютера в SD-карту необходимо в правом окне вкладки **Файлы** открыть директорию, в которую будет скопирован файл, в левом окне вкладки найти внутри «\$SDDev\$» файл для копирования и выделить его, нажав левой кнопкой мыши, затем копировать файл на SD-карту, нажав кнопку  (рисунок 3.81).

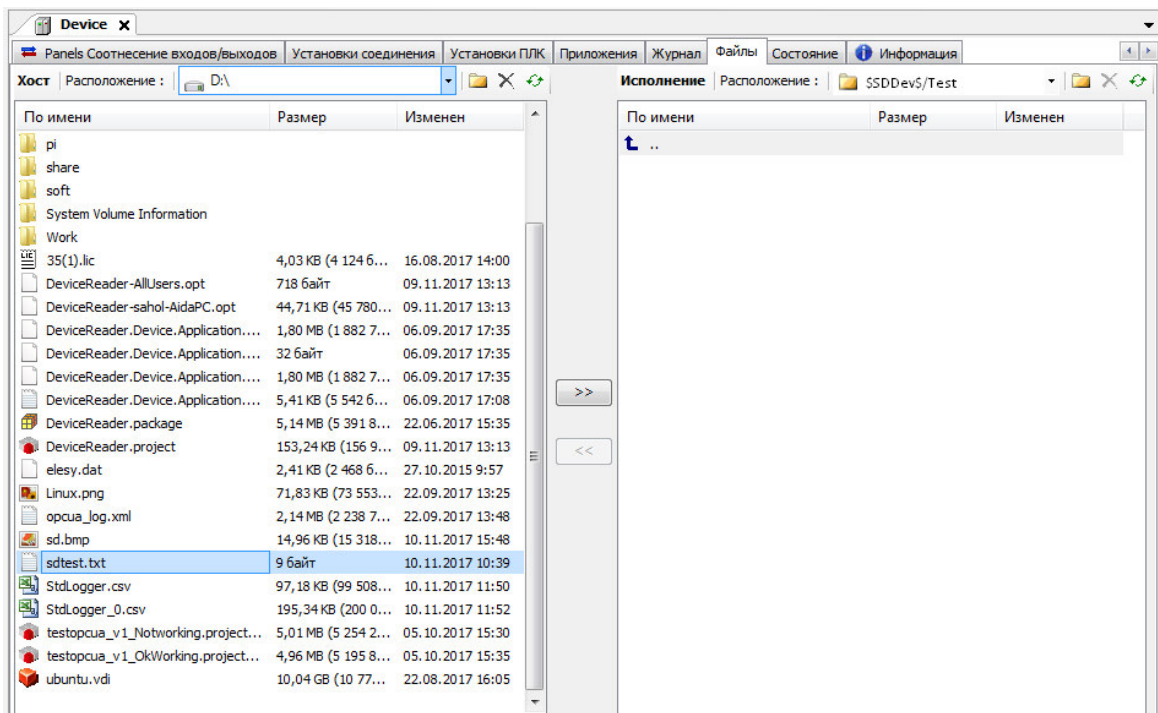



Рисунок 3.81 – Копирование файла с компьютера на карту памяти с помощью CoDeSys

7 Чтобы копировать элемент из SD-карты на компьютер необходимо в левом окне вкладки **Файлы** открыть директорию, в которую будет скопирован файл, в правом окне вкладки найти внутри «\$SDDev\$» файл для копирования и выделить его, нажав левой кнопкой мыши, затем копировать файл в папку компьютера, нажав кнопку  (рисунок 3.82).

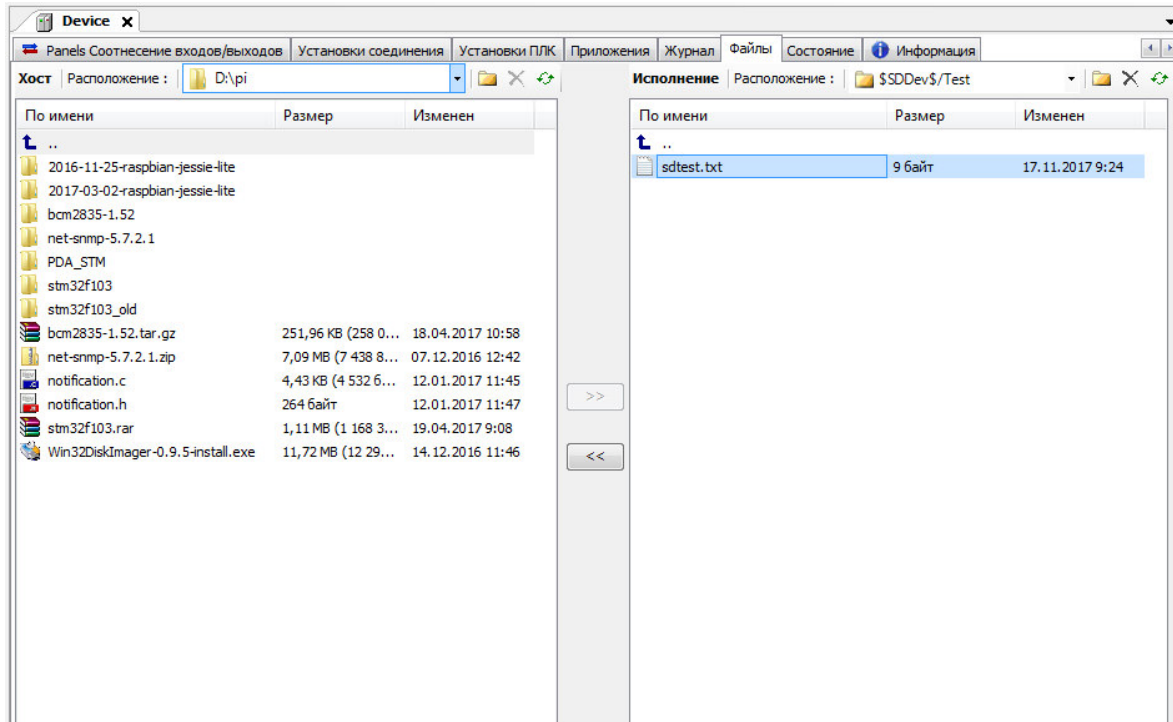


Рисунок 3.82 – Копирование файла из карты памяти на компьютер с помощью CoDeSys

Примечание: при возникновении неполадок во время работы с контроллером пользователю необходимо отключить питание контроллера, извлечь и форматировать SD-карту.

3.16.2 Работа с SD-картой из задачи пользователя.

Пользователь имеет возможность записывать необходимую информацию на карту, а также считывать ее с помощью из задачи пользователя *CoDeSys*.

Ниже представлен пример задачи пользователя по считыванию информации из текстового файла, находящегося на SD-карте контроллера. Для реализации задачи необходимо:

1 Создать папку **"testsd"** в корневой папке устройства.

2 В папке **"testsd"** создать файл **"sdtest.txt"** и записать в него фрагмент текстовой информации.

3 Установить SD-карту. Для этого необходимо отключить питание контроллера и установить SD-карту в слот контроллера так, чтобы раздался характерный щелчок.

ВНИМАНИЕ! Перед извлечением и перед установкой SD-карты в слот контроллера, предварительно необходимо отключить питание контроллера. В ходе работы контроллера запрещается извлекать SD-карту ввиду потери данных незавершенных операций обмена.

4 Загрузить на контроллер программный код, представленный ниже:

```
PROGRAM PLC_PRG
```

```
VAR
```

```
SDResult : RTS_IEC_RESULT;
```

```
hSDFile : RTS_IEC_HANDLE;
```

```
mySDStr : string;
```

```
END_VAR
```

```
hSDFile := .SysFile.SysFileOpen('$$SSDDev$$/testsd/sdtest.txt', ACCESS_MODE.AM_READ, ADR(SDResult));
```

```

IF( hSDFile <> SysFile.RTS_INVALID_HANDLE ) THEN
SysFile.SysFileRead(hSDFile, ADR(mySDStr), 8, ADR(SDResult) );
SysFileClose(hSDFile);
END_IF

```

ВАЖНО! Библиотеки **SysFile** и **ISysTypes2** необходимо добавлять в проект вручную.

После запуска программы в *CoDeSys* переменная *mySDStr* отображает текстовую информацию из файла "sdtest.txt". В зависимости заданных при форматировании SD карты параметров (наличия 'partition') задаваемое имя *SDDev* может меняться на *SDDev1*.

3.16.3 Запись логов на SD-карту

Для записи отладочной информации с картой необходимо выполнить следующие действия:

- 1 Создать папку "**elesylog**" в корневой папке устройства.
- 2 Установить SD-карту. Для этого необходимо отключить питание контроллера и установить SD-карту в слот контроллера так, чтобы раздался характерный щелчок.

ВНИМАНИЕ! Перед извлечением и перед установкой SD-карты в слот контроллера, предварительно необходимо отключить питание контроллера. В ходе работы контроллера запрещается извлекать SD-карту ввиду потери данных незавершенных операций обмена.

Информация записывается в директорию "**elesylog**" программой *CoDeSys* автоматически в зависимости от информационных сообщений.

3.16.4 Доступ к USB-Flash через CoDeSys

Доступ к USB-Flash контроллера может осуществляться через *CoDeSys*. С помощью среды программирования можно записывать файлы с компьютера на USB-Flash, а также удалять с носителя лишнюю информацию.

Для работы с USB-Flash через *CoDeSys* необходимо:

- 1 Установить USB-Flash. Для этого необходимо отключить питание контроллера и установить USB-Flash в соответствующий слот контроллера.

ВНИМАНИЕ! Перед извлечением и перед установкой USB-Flash в слот контроллера, предварительно необходимо отключить питание контроллера. В ходе работы контроллера запрещается извлекать USB-Flash ввиду потери данных незавершенных операций обмена.

- 2 В дереве устройств выбрать устройство **Device** и дважды нажать на него левой кнопкой мыши. В появившейся закладке найти вкладку **Файлы** (рисунок 3.83). Левое окно вкладки отображает данные на компьютере, правое окно – содержание памяти контроллера.

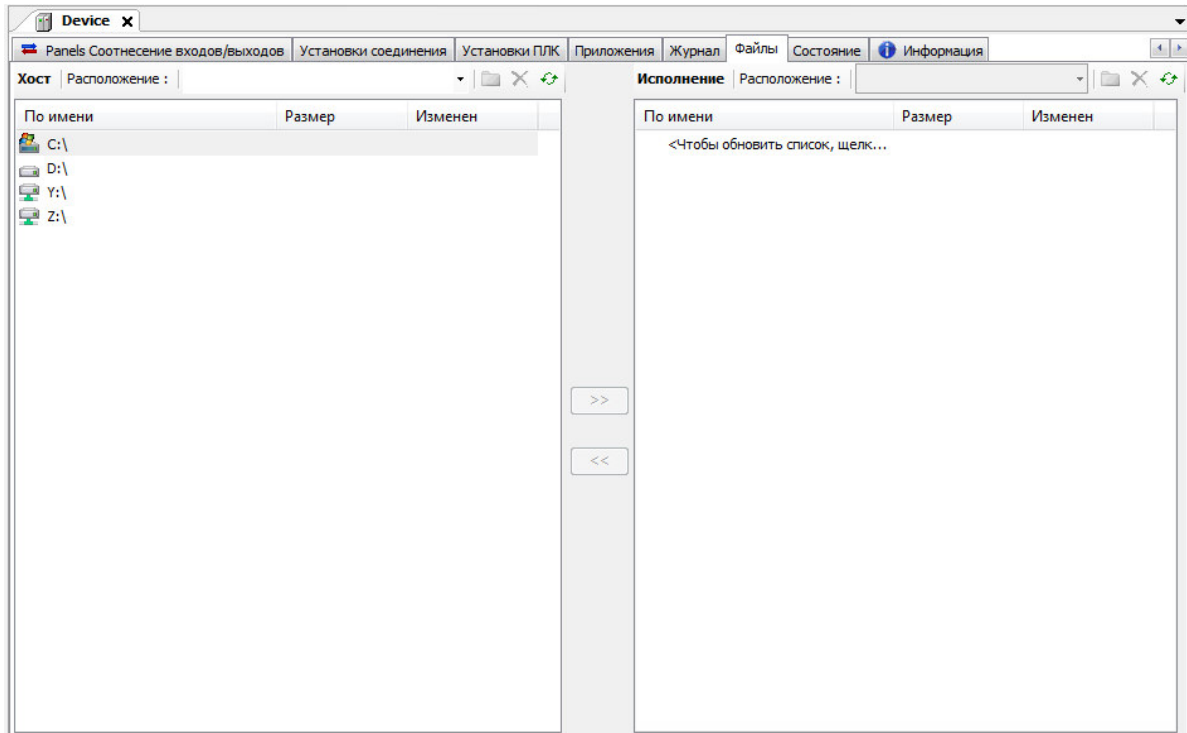





Рисунок 3.83 – Вкладка **Файлы**. Начало работы с USB-Flash в среде CoDeSys

3 Обновить информацию памяти контроллера, нажав на кнопку , расположенную в правом верхнем углу. После обновления в правом окне вкладки **Файлы** отображаются данные из памяти контроллера, кнопки  – «Новая директория» и  – «Удалить элемент» вверху правого окна становятся активными (рисунок 3.84).

4 Открыть папку «\$USBDev\$» (рисунок 3.84). Данная папка отвечает за содержание USB-Flash.

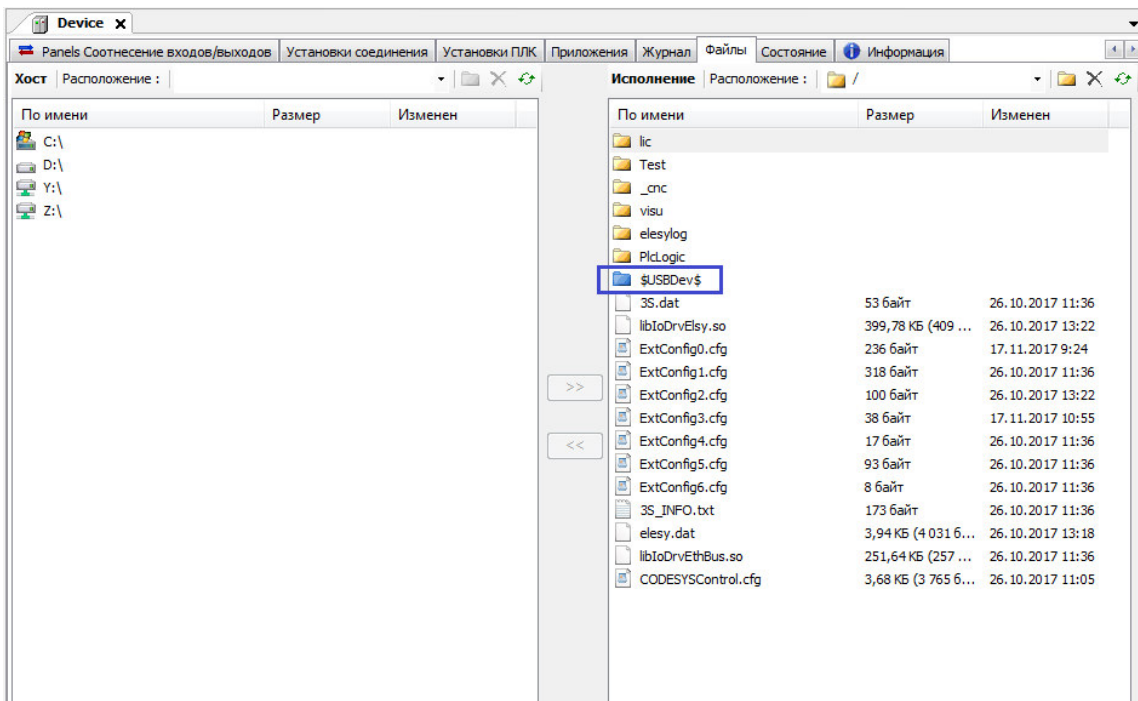


Рисунок 3.84 – Вкладка **Файлы**. Отображение содержания памяти контроллера после обновления

5 Чтобы создать папку на USB-Flash необходимо нажать на кнопку . В появившемся окне «Новая директория» (рисунок 3.85) ввести имя директории.

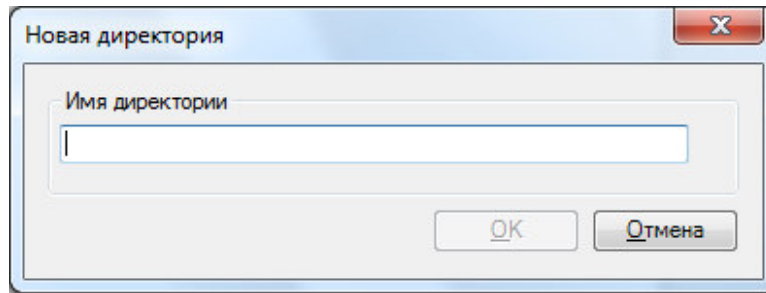



Рисунок 3.85 – Окно «Новая директория»

6 Чтобы копировать элемент из компьютера в USB-Flash необходимо в правом окне вкладки **Файлы** открыть директорию внутри «\$USBDev\$», в которую будет скопирован файл, в левом окне вкладки найти нужный файл для копирования и выделить его, нажав левой кнопкой мыши, затем копировать файл на USB-Flash, нажав кнопку  (рисунок 3.86).

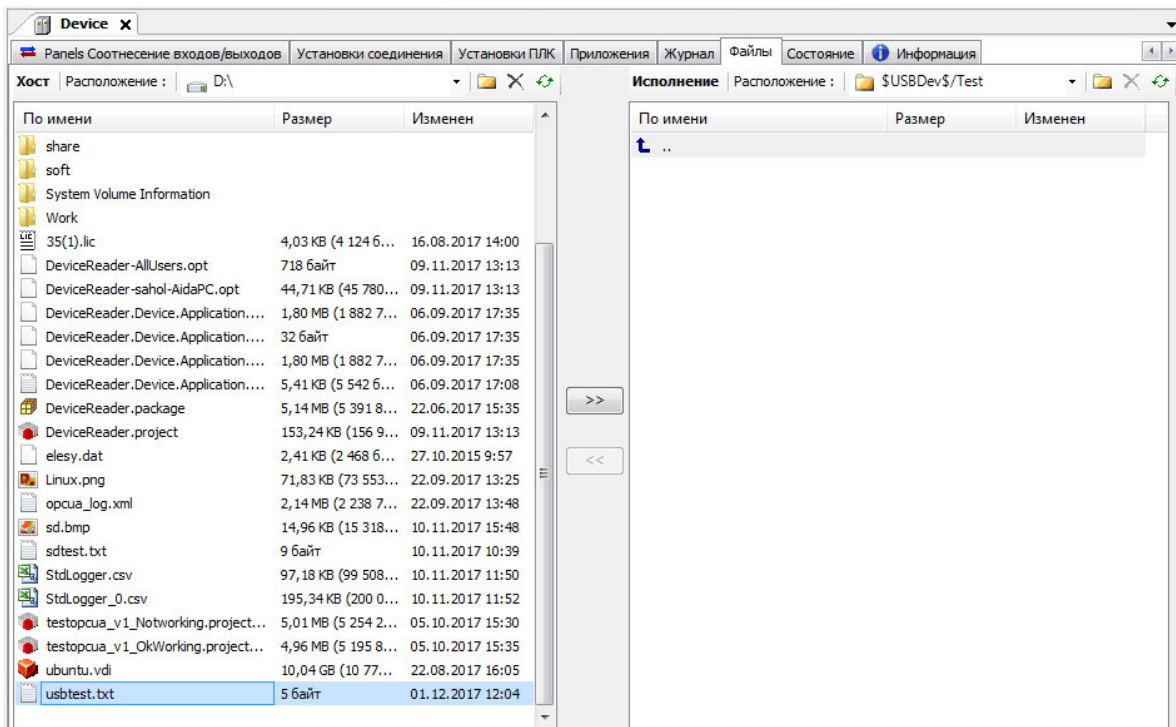



Рисунок 3.86 – Копирование файла с компьютера на USB-Flash с помощью CoDeSys

7 Чтобы копировать элемент из USB-Flash на компьютер необходимо в левом окне вкладки **Файлы** открыть директорию, в которую будет скопирован файл, в правом окне вкладки найти внутри «\$USBDev\$» файл для копирования и выделить его, нажав левой кнопкой мыши, затем копировать файл в папку компьютера, нажав кнопку  (рисунок 3.87).

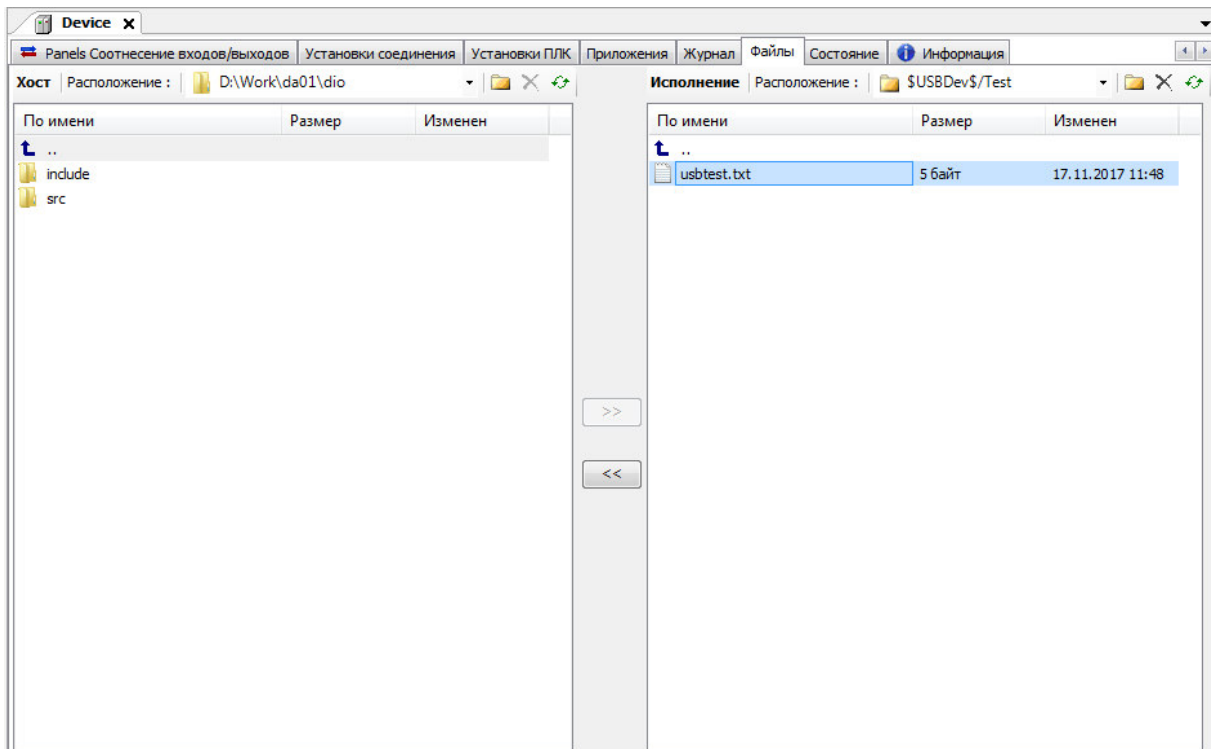


Рисунок 3.87 – Копирование файла из карты памяти на компьютер с помощью CoDeSys

Примечание: при возникновении неполадок во время работы с контроллером пользователю необходимо отключить питание контроллера, извлечь и форматировать USB-Flash.

3.16.5 Работа с USB-Flash из задачи пользователя.

Пользователь имеет возможность записывать необходимую информацию на USB-Flash, а также считывать ее с помощью из задачи пользователя *CoDeSys*.

Ниже представлен пример задачи пользователя по считыванию информации из текстового файла, находящегося на USB-Flash контроллера. Для реализации задачи необходимо:

1 Создать папку "**testusb**" в корневой папке устройства.

2 В папке "**testusb**" создать файл "**usbtest.txt**" и записать в него фрагмент текстовой информации.

3 Установить USB-Flash. Для этого необходимо отключить питание контроллера и установить USB-Flash в соответствующий слот контроллера.

ВНИМАНИЕ! Перед извлечением и перед установкой USB-Flash в слот контроллера, предварительно необходимо отключить питание контроллера. В ходе работы контроллера запрещается извлекать USB-Flash ввиду потери данных незавершенных операций обмена.

4 Загрузить на контроллер программный код, представленный ниже:

```
PROGRAM PLC_PRG
VAR
  USBResult : RTS_IEC_RESULT;
  hUSBFile : RTS_IEC_HANDLE;
  myUSBStr : string;
END_VAR

hUSBFile := .SysFile.SysFileOpen('$USBDev$/testusb/usbtest.txt', ACCESS_MODE.AM_READ,
ADR(USBResult));
IF (hUSBFile <> SysFile.RTS_INVALID_HANDLE ) THEN
  SysFile.SysFileRead(hUSBFile, ADR(myUSBStr), 15, ADR(USBResult) );
  SysFileClose(hUSBFile);
END_IF
```

ВАЖНО! Библиотеки **SysFile** и **ISysTypes2** необходимо добавлять в проект вручную.

После запуска программы в *CoDeSys* переменная `myUSBStr` отображает текстовую информацию из файла **"usbtest.txt"**. В зависимости заданных при форматировании USB-Flash карты параметров (наличия 'partition') задаваемое имя `SDDev` может меняться на `USBDev1`.

3.16.6 Запись логов на USB-Flash

Для записи отладочной информации на USB-Flash необходимо выполнить следующие действия:

1 Создать папку **"elesylog"** в корневой папке устройства.

2 Установить USB-Flash. Для этого необходимо отключить питание контроллера и установить USB-Flash в USB-порт контроллера.

ВНИМАНИЕ! Перед извлечением и перед установкой USB-Flash в USB-порт контроллера, предварительно необходимо отключить питание контроллера. В ходе работы контроллера запрещается извлекать USB-Flash ввиду потери данных незавершенных операций обмена.

Информация записывается в директорию **"elesylog"** программой *CoDeSys* автоматически в зависимости от информационных сообщений.

3.17 Работа с сигналами входов/выходов контроллера

В данном подразделе представлено описание функциональных сигналов входов/выходов контроллера. Основные функциональные сигналы входов/выходов расположены в узле **CPU_IO** (входные и выходные сигналы контроллера). Все сигналы входов/выходов разделены на четыре группы:

- Входные дискретные сигналы;
- Выходные дискретные сигналы;
- Входные аналоговые сигналы;
- Выходные аналоговые сигналы.

Настройка параметров работы и служебная информация располагаются на закладке "Редактор параметров" коннектора **CPU_IO**. Для выполнения операции следует:

1 Открыть закладку просмотра и настройки модуля **ELSYMA_M01_XXX**, выделив коннектор **CPU_IO** в дереве устройств и дважды нажав левую кнопку "мыши".

2 Перейти на закладку *Редактор параметров* (рисунок 3.88).

3 Настроить соответствующие параметры модуля.

4 На вкладке *Редактор параметров* расположены две области:

- информация модуля;
- конфигурационные параметры модуля.

Область "Информация Модуля" служит для представления служебной информации работы контроллера.

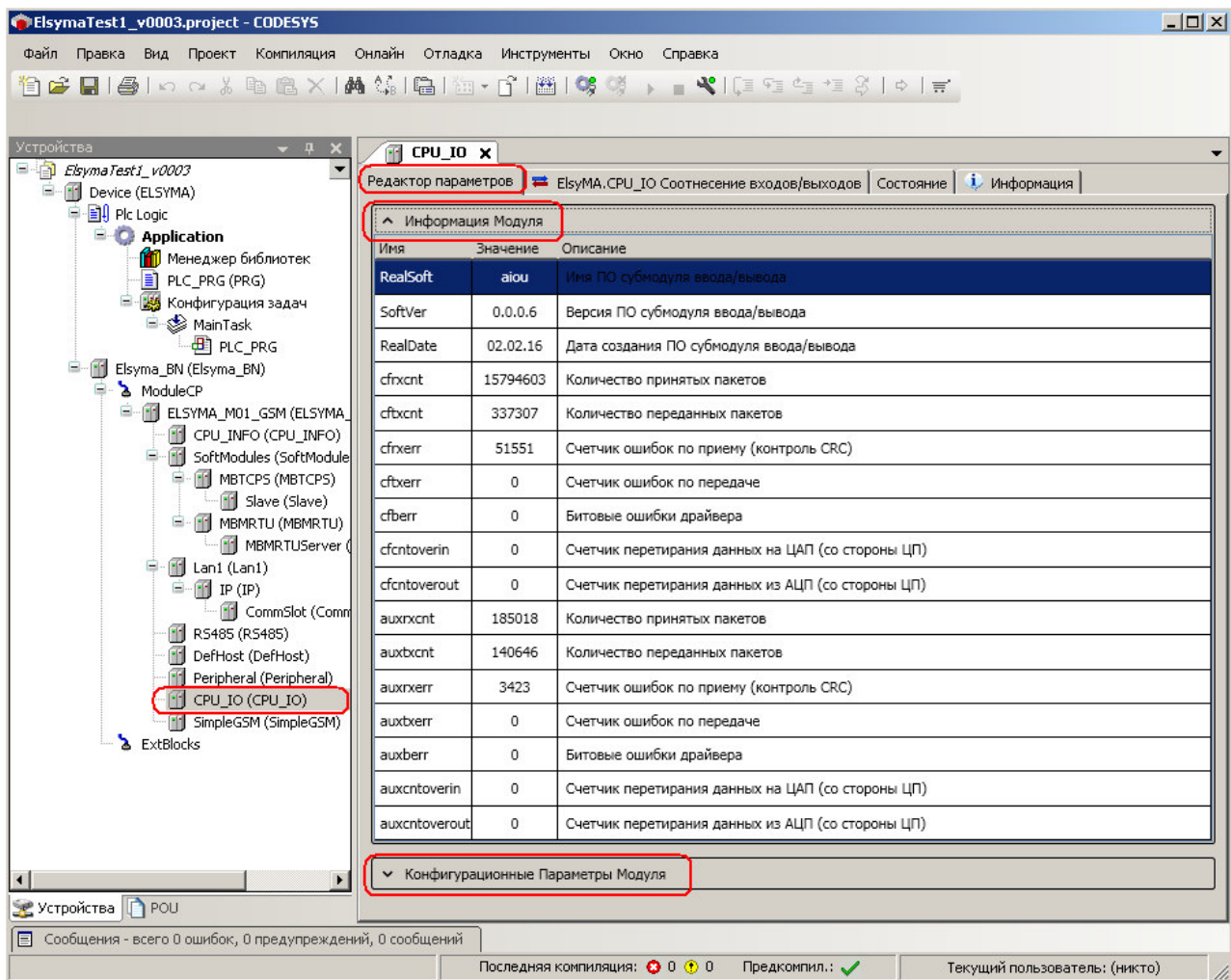


Рисунок 3.88 – Контроллер Элсима. Настройка каналов входа-выхода

Информационные параметры модуля **ELSYMA_M01_XXX-CPU_IO** перечислены в таблице 3.64. Данные параметры не доступны для редактирования пользователем.

Таблица 3.64 – Модуль CPU_IO. Информация модуля

Имя	Значение "по умолчанию"	Описание
<i>RealSoft</i>	'no data'	Имя ПО submodule ввода/вывода
<i>SoftVer</i>	'no data'	Версия ПО submodule ввода/вывода
<i>RealDate</i>	'no data'	Дата создания ПО submodule ввода/вывода
Статистика работы драйвера поддержки сопроцессора ввода/вывода со стороны ЦП		
<i>cfrcxcnt</i>		Количество принятых пакетов

Таблица 3.64 – Модуль CPU_Ю. Информация модуля

Имя	Значение "по умолчанию"	Описание
<i>cftxcnt</i>		Количество переданных пакетов
<i>cfrxerr</i>		Счетчик ошибок по приему (контроль CRC)
<i>cftxerr</i>		Счетчик ошибок по передаче
<i>cfberr</i>		Битовые ошибки драйвера
<i>cfntoverin</i>		Счетчик перетирания данных на ЦАП (со стороны ЦП)
<i>cfntoverout</i>		Счетчик перетирания данных из АЦП (со стороны ЦП)
Статистика работы драйвера поддержки сопроцессора ввода/вывода со стороны сопроцессора		
<i>aurxcnt</i>		Количество принятых пакетов
<i>autxcnt</i>		Количество переданных пакетов
<i>aurxerr</i>		Счетчик ошибок по приему (контроль CRC)
<i>autxerr</i>		Счетчик ошибок по передаче
<i>auberr</i>		Битовые ошибки драйвера
<i>aucntoverin</i>		Счетчик перетирания данных на ЦАП (со стороны ЦП)
<i>aucntoverout</i>		Счетчик перетирания данных из АЦП (со стороны ЦП)

Область "Конфигурационные Параметры Модуля" служит для задания параметров работы сопроцессора ввода/вывода. Данные параметры доступны для редактирования пользователем. Редактирование параметров доступно только в **off-line** режиме. Для изменения параметров работы необходимо отключиться от контроллера, изменить параметры и загрузить новый проект. Описание конфигурационных параметров представлено в 3.17.3.

3.17.1 Сигналы дискретного входа

Для работы с входными дискретными сигналами параметры не задаются. На рисунке 3.89 представлен вид закладки *ElsyMA.CPU_IO Соотнесение входов/выходов*. Входные дискретные сигналы сгруппированы в папке "Digital inputs".

Переменная	Соотнесение	Канал	Адрес	Тип	Описание
		DigInDiag	%IB697	BYTE	Диагностика работы измерительных каналов
Application.PLC_PRG.myDigIn[1]		DigIn1_1	%IX698-0	BIT	Состояние канала 1 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[2]		DigIn1_2	%IX698-1	BIT	Состояние канала 2 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[3]		DigIn1_3	%IX698-2	BIT	Состояние канала 3 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[4]		DigIn1_4	%IX698-3	BIT	Состояние канала 4 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[5]		DigIn1_5	%IX698-4	BIT	Состояние канала 5 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[6]		DigIn1_6	%IX698-5	BIT	Состояние канала 6 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[7]		DigIn1_7	%IX698-6	BIT	Состояние канала 7 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[8]		DigIn1_8	%IX698-7	BIT	Состояние канала 8 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[9]		DigIn1_9	%IX699-0	BIT	Состояние канала 9 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[10]		DigIn1_10	%IX699-1	BIT	Состояние канала 10 дискретного ввода 1-й группы (разъем DIN1)
Application.PLC_PRG.myDigIn[11]		DigIn2_1	%IX699-2	BIT	Состояние канала 1 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[12]		DigIn2_2	%IX699-3	BIT	Состояние канала 2 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[13]		DigIn2_3	%IX699-4	BIT	Состояние канала 3 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[14]		DigIn2_4	%IX699-5	BIT	Состояние канала 4 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[15]		DigIn2_5	%IX699-6	BIT	Состояние канала 5 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[16]		DigIn2_6	%IX699-7	BIT	Состояние канала 6 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[17]		DigIn2_7	%IX700-0	BIT	Состояние канала 7 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[18]		DigIn2_8	%IX700-1	BIT	Состояние канала 8 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[19]		DigIn2_9	%IX700-2	BIT	Состояние канала 9 дискретного ввода 2-й группы (разъем DIN2)
Application.PLC_PRG.myDigIn[20]		DigIn2_10	%IX700-3	BIT	Состояние канала 10 дискретного ввода 2-й группы (разъем DIN2)

Рисунок 3.89 – Сигналы дискретного входа.
Закладка *ElsyMA.CPU_IO Соотнесение входов/выходов*

Описание входных дискретных сигналов и соответствие с физическим входом представлено в таблице 3.65. Схема подключения сигналов приведена на рисунке А.1 приложения А. Технические характеристики дискретных входов приведены в таблице 1.1.

Таблица 3.65 – Модуль CPU_IO. Входные дискретные сигналы

Имя	Тип	Подключение		Описание
		Разъем	Вход	
DigInDiag	BYTE	-	-	Диагностика работы измерительных каналов. В данной версии ПО всегда равно "0"
DigIn1_1	Bit	DIN1	1	Состояние канала 1 дискретного входа разъема DIN1
DigIn1_2	Bit	DIN1	2	Состояние канала 2 дискретного входа разъема DIN1
DigIn1_3	Bit	DIN1	3	Состояние канала 3 дискретного входа разъема DIN1
DigIn1_4	Bit	DIN1	4	Состояние канала 4 дискретного входа разъема DIN1
DigIn1_5	Bit	DIN1	5	Состояние канала 5 дискретного входа разъема DIN1
DigIn1_6	Bit	DIN1	6	Состояние канала 6 дискретного входа разъема DIN1
DigIn1_7	Bit	DIN1	7	Состояние канала 7 дискретного входа разъема DIN1
DigIn1_8	Bit	DIN1	8	Состояние канала 8 дискретного входа разъема DIN1
DigIn1_9	Bit	DIN1	9	Состояние канала 9 дискретного входа разъема DIN1
DigIn1_10	Bit	DIN1	10	Состояние канала 10 дискретного входа разъема DIN1
DigIn2_1	Bit	DIN2	1	Состояние канала 1 дискретного входа разъема DIN2
DigIn2_2	Bit	DIN2	2	Состояние канала 2 дискретного входа разъема DIN2
DigIn2_3	Bit	DIN2	3	Состояние канала 3 дискретного входа разъема DIN2
DigIn2_4	Bit	DIN2	4	Состояние канала 4 дискретного входа разъема DIN2
DigIn2_5	Bit	DIN2	5	Состояние канала 5 дискретного входа разъема DIN2
DigIn2_6	Bit	DIN2	6	Состояние канала 6 дискретного входа разъема DIN2
DigIn2_7	Bit	DIN2	7	Состояние канала 7 дискретного входа разъема DIN2
DigIn2_8	Bit	DIN2	8	Состояние канала 8 дискретного входа разъема DIN2
DigIn2_9	Bit	DIN2	9	Состояние канала 9 дискретного входа разъема DIN2
DigIn2_10	Bit	DIN2	10	Состояние канала 10 дискретного входа разъема DIN2

ВАЖНО! В соответствии со схемой подключения, приведенной на рисунке А.1, значение сигнала, равное **TRUE (1)**, соответствует замкнутому ключу **Kx.x**. Значение сигнала, равное **FALSE (0)**, соответствует разомкнутому ключу **Kx.x**.

ВАЖНО! Обновление входных дискретных сигналов производится в каждом цикле задачи на стадии обновления входных данных. Обновление одного дискретного входа занимает примерно 17 мкс, соответственно, при обновлении всех данных цикл задачи увеличится на 340 мкс.

На рисунке 3.90 приведен пример кода для описания переменных дискретного входа.

```

16 SetAllITU      : INT := 2;      (* Установка всех DigOut в состояние ON *)
17
18 (* Дискретный ввод *)
19 DigInMax      : INT := 20;     (* Количество сигналов Дискретного ввода *)
20 myDigIn       : ARRAY[1..20] OF BOOL; (* сигналы Дискретного ввода *)
21
22 (* Дискретный вывод *)
23 DigOutMax     : INT := 8;     (* Количество сигналов Дискретного вывода *)
24 myDigOut     : ARRAY[1..8] OF BOOL; (* сигналы Дискретного вывода *)
25
26

```

Рисунок 3.90 – Входные дискретные сигналы. Пример описания переменных

Сигналы могут быть описаны в виде массива переменных или в виде отдельных переменных. Тип переменных должен быть **BOOL**.

3.17.2 Сигналы дискретного выхода

Для работы с выходными дискретными сигналами параметры не задаются. На рисунке 3.91 представлен вид закладки *ElsyMA.CPU_IO Соотнесение входов/выходов*. Сигналы дискретного выхода сгруппированы в папке "Digital Outputs".

Переменная	Соотнесение	Канал	Адрес	Тип	Описание
Application.PLC_PRG.myDigOut[1]		DigOut1	%QX48-0	BIT	Управление каналом 1 дискретного вывода типа Открытый коллектор
Application.PLC_PRG.myDigOut[2]		DigOut2	%QX48-1	BIT	Управление каналом 2 дискретного вывода типа Открытый коллектор
Application.PLC_PRG.myDigOut[3]		DigOut3	%QX48-2	BIT	Управление каналом 3 дискретного вывода типа Открытый коллектор
Application.PLC_PRG.myDigOut[4]		DigOut4	%QX48-3	BIT	Управление каналом 4 дискретного вывода типа Открытый коллектор
Application.PLC_PRG.myDigOut[5]		RelayOut1	%QX48-4	BIT	Управление каналом 1 дискретного вывода типа Реле
Application.PLC_PRG.myDigOut[6]		RelayOut2	%QX48-5	BIT	Управление каналом 2 дискретного вывода типа Реле
Application.PLC_PRG.myDigOut[7]		RelayOut3	%QX48-6	BIT	Управление каналом 3 дискретного вывода типа Реле
Application.PLC_PRG.myDigOut[8]		RelayOut4	%QX48-7	BIT	Управление каналом 4 дискретного вывода типа Реле
		DigOutDi...	%IB751	BYTE	Диагностика работы каналов управления

Рисунок 3.91 – Сигналы дискретного выхода.
Закладка *ElsyMA.CPU_IO Соотнесение входов/выходов*

Описание сигналов дискретного выхода и соответствие с физическим выходом представлено в таблице 3.66. Схема подключения сигналов приведена на рисунке А.2. Технические характеристики дискретных выходов приведены в таблице 3.66.

Таблица 3.66 – Модуль CPU_IO. Сигналы дискретного выхода

Имя	Тип	Подключение		Описание
		Разъем	Выход	
<i>DigOut_1</i>	Bit	DOUT	1	Управление каналом 1 дискретного выхода типа "Открытый коллектор"
<i>DigOut_1</i>	Bit	DOUT	2	Управление каналом 2 дискретного выхода типа "Открытый коллектор"
<i>DigOut_1</i>	Bit	DOUT	3	Управление каналом 3 дискретного выхода типа "Открытый коллектор"
<i>DigOut_4</i>	Bit	DOUT	4	Управление каналом 4 дискретного выхода типа "Открытый коллектор"
<i>RelayOut1</i>	Bit	RELAY	1	Управление каналом 1 дискретного выхода типа "Реле"
<i>RelayOut1</i>	Bit	RELAY	2	Управление каналом 2 дискретного выхода типа "Реле"
<i>RelayOut1</i>	Bit	RELAY	4	Управление каналом 3 дискретного выхода типа "Реле"
<i>RelayOut1</i>	Bit	RELAY	5	Управление каналом 4 дискретного выхода типа "Реле"
<i>DigOutDiag</i>	BYTE	-	-	Диагностика работы каналов управления. В данной версии ПО всегда равно "0"

ВАЖНО! В соответствии со схемой подключения, приведенной на рисунке А.2, значение сигнала, равное **TRUE (1)**, соответствует включенной нагрузке (выходной ключ открыт или контакты релейного выхода замкнуты). Значение сигнала, равное **FALSE (0)**, соответствует выключенной нагрузке (выходной ключ закрыт или контакты релейного выхода разомкнуты).

ВАЖНО! Обновление входных дискретных сигналов производится в каждом цикле задачи на стадии обновления выходных данных. Обновление одного дискретного выхода занимает примерно 25 мкс, соответственно, при обновлении всех данных цикл задачи увеличится на 200 мкс.

ВАЖНО! При включении питания или срабатывании **Watch-dog** таймера все выходы аппаратно переводятся в состояние "**Выключено**" (**FALSE**).

На рисунке 3.92 приведен пример кода для описания переменных дискретного выхода.

```

CPU_IO | PLC_PRG x
19 DigInMax : INT := 20; (* Количество сигналов Дискретного ввода *)
20 myDigIn : ARRAY[1..20] OF BOOL; (* Сигналы Дискретного ввода *)
21
22 (* Дискретный вывод *)
23 DigOutMax : INT := 8; (* Количество сигналов Дискретного вывода *)
24 myDigOut : ARRAY[1..8] OF BOOL; (* Сигналы Дискретного вывода *)
25
26 (* Аналоговый ввод *)
27 AnalInMax : INT := 4; (* Количество сигналов Аналогового ввода *)
28 myAnalIn : ARRAY[1..4] OF REAL; (* Сигналы Аналогового ввода *)
29 myAInDiag : ARRAY[1..4] OF BYTE; (* Диагностика каналов Аналогового ввода *)

```

Рисунок 3.92 – Сигналы дискретного выхода. Пример описания переменных

Сигналы могут быть описаны в виде массива переменных или в виде отдельных переменных. Тип переменных должен быть **BOOL**.

3.17.3 Сигналы аналогового входа

В контроллере входные аналоговые сигналы могут работать в следующих режимах:

- измерение значения напряжения;
- измерение значения тока;
- измерение значения температуры с датчика типа "Термопара";
- измерение значения температуры с датчика типа "Термосопротивление".

Выбор режима работы каждого канала и необходимые параметры задаются на вкладке *Редактор параметров*. На рисунке 3.93 представлен внешний вид вкладки и выделены параметры, отвечающие за обработку входных аналоговых сигналов.

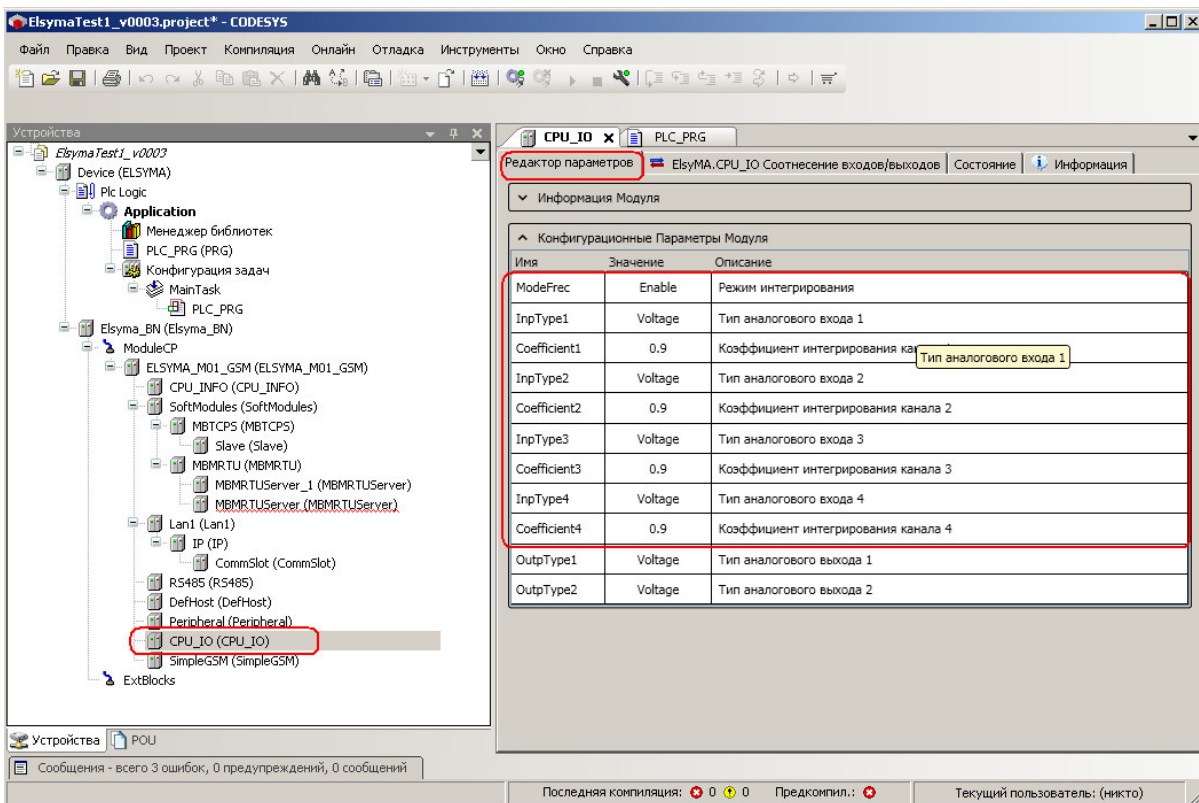


Рисунок 3.93 – Параметры аналогового входа. Закладка *Редактор параметров*

ВАЖНО! При выборе определенного режима работы аналогового входа необходимо учитывать схему подключения для выбранного режима. Схемы подключения для различных режимов приведены на рисунках А.3, А.4, А.5 и А.6.

Параметры для работы с каналами аналогового входа модуля **ELSYMA_M01_XXX-CPU_IO** перечислены в таблицах 3.67 и 3.68. Параметры доступны для редактирования пользователем только в режиме **off-line**.

Таблица 3.67 – Модуль CPU_IO. Параметры модуля (аналоговый вход)

Имя	Значение "по умолчанию"	Описание
<i>ModeFrec</i>	Disable	Режим интегрирования входных сигналов (<i>Enable/Disable</i>). При включенном режиме время измерения каждого сигнала составляет 150 мс, а при выключенном – 25 мс. Общее время измерения всех каналов можно вычислить, умножив время измерения одного канала на количество активированных каналов (значение типа входа не равно "Disable")
<i>InpType1</i>	Voltage	Тип аналогового входа 1. Задается в соответствии с таблицей 3.68
<i>Coefficient1</i>	0,008	Коэффициент интегрирования канала 1 ($0,0001 < Coefficient \leq 1,0$)
<i>InpType2</i>	Voltage	Тип аналогового входа 2. Задается в соответствии с таблицей 3.68
<i>Coefficient2</i>	0,008	Коэффициент интегрирования канала 2 ($0,0001 < Coefficient \leq 1,0$)
<i>InpType3</i>	Voltage	Тип аналогового входа 3. Задается в соответствии с таблицей 3.68
<i>Coefficient3</i>	0,008	Коэффициент интегрирования канала 3 ($0,0001 < Coefficient \leq 1,0$)
<i>InpType4</i>	Voltage	Тип аналогового входа 4. Задается в соответствии с таблицей 3.68
<i>Coefficient4</i>	0,008	Коэффициент интегрирования канала 4 ($0,0001 < Coefficient \leq 1,0$)

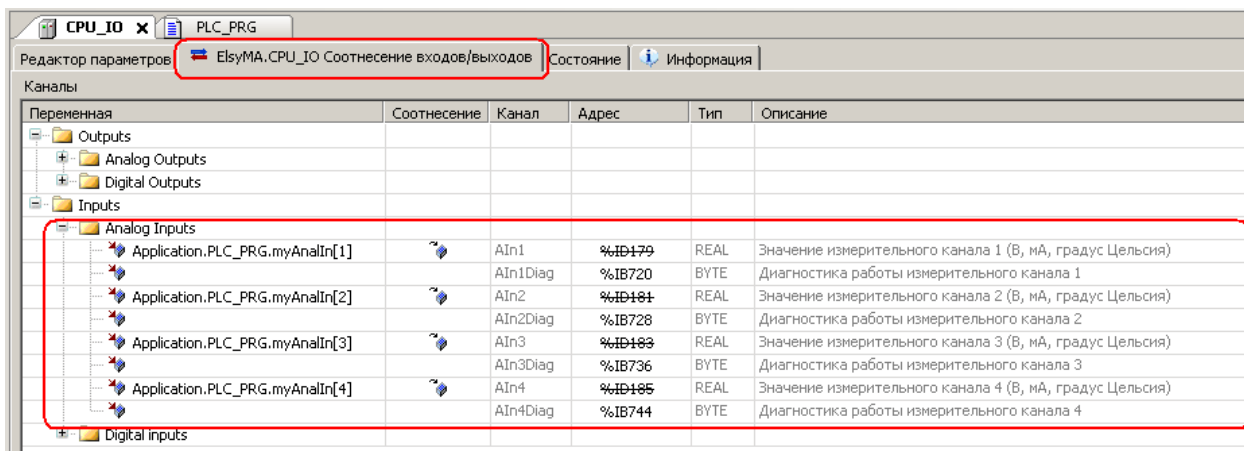
Таблица 3.68 – Модуль CPU_IO. Тип аналогового входа

Тип	Описание	Диапазон измерения, °C	
		Мин.	Макс.
<i>Dissable</i>	Канал не обрабатывается	–	–
<i>Current</i>	Ток 0–20 мА	–	–
<i>Voltage</i>	Напряжение 0–10 В	–	–
<i>TXAK</i>	Термопары типа ТХА (К)	-250	+900
<i>TXAK_тк</i>	Термопары типа ТХА (К) с компенсацией холодного спая	-250	+900
<i>TXAL</i>	Термопары типа ТХК (L)	0	+800
<i>TXAL_тк</i>	Термопары типа ТХК (L) с компенсацией холодного спая	0	+800
<i>TXAE</i>	Термопары типа ТХКн (Е)	-250	+1000
<i>TXAE_тк</i>	Термопары типа ТХКн (Е) с компенсацией холодного спая	-250	+1000
<i>ТПП10</i>	Термопары типа ТПП10 (S)	0	+1700
<i>ТПП10_тк</i>	Термопары типа ТПП10 (S) с компенсацией холодного спая	0	+1700
<i>ТНН</i>	Термопары типа ТНН (N)	-250	+1000
<i>ТНН_тк</i>	Термопары типа ТНН (N) с компенсацией холодного спая	-250	+1000
<i>ТПР</i>	Термопары типа ТПР (В)	+250	+1800
<i>ТПР_тк</i>	Термопары типа ТПР (В) с компенсацией холодного спая	+250	+1800
<i>ТЖК</i>	Термопары типа ТЖК (J)	-200	+600
<i>ТЖК_тк</i>	Термопары типа ТЖК (J) с компенсацией холодного спая	-200	+600
<i>ТВР</i>	Термопары типа ТВР (А-1)	0	+2500
<i>ТВР_тк</i>	Термопары типа ТВР (А-1) с компенсацией холодного спая	0	+2500
<i>ТПП13</i>	Термопары типа ТПП13 (R)	0	+1600
<i>ТПП13_тк</i>	Термопары типа ТПП13 (R) с компенсацией холодного спая	0	+1600
<i>TSM 50M</i>	Термосопротивления в режиме трехпроводного подключения типа TSM 50M	-50	+150
<i>TSM 100M</i>	Термосопротивления в режиме трехпроводного подключения типа TSM 100M	-50	+150
<i>TSM 500M</i>	Термосопротивления в режиме трехпроводного подключения типа TSM 500M	-50	+150

Таблица 3.68 – Модуль CPU_IO. Тип аналогового входа

Тип	Описание	Диапазон измерения, °С	
		Мин.	Макс.
<i>ТСП 50П</i>	Термосопротивления в режиме трехпроводного подключения типа ТСП 50П	-50	+500
<i>ТСП 100П</i>	Термосопротивления в режиме трехпроводного подключения типа ТСП 100П	-50	+500
<i>ТСП 500П</i>	Термосопротивления в режиме трехпроводного подключения типа ТСП 500П	-50	+500
<i>ТСП 1000П</i>	Термосопротивления в режиме трехпроводного подключения типа ТСП 1000П	-50	+500
<i>ТСП Pt50</i>	Термосопротивления в режиме трехпроводного подключения типа ТСП Pt50	-50	+500
<i>ТСП Pt100</i>	Термосопротивления в режиме трехпроводного подключения типа ТСП Pt100	-50	+500
<i>ТСН 100Н</i>	Термосопротивления в режиме трехпроводного подключения типа ТСН 100Н	-50	+150
<i>ТСН 500Н</i>	Термосопротивления в режиме трехпроводного подключения типа ТСН 500Н	-50	+150
<i>ТСН 1000Н</i>	Термосопротивления в режиме трехпроводного подключения типа ТСН 1000Н	-50	+150

На рисунке 3.94 представлен вид закладки *ElsyMA.CPU_IO Соотнесение входов/выходов*. Входные аналоговые сигналы сгруппированы в папке "Analog Inputs".

Рисунок 3.94 – Сигналы аналогового входа. Закладка *ElsyMA.CPU_IO Соотнесение входов/выходов*

Описание входных аналоговых сигналов и соответствие с физическим входом представлено в таблице 3.69. Схемы подключения сигналов приведены на рисунках А.3, А.4, А.5 и А.6. Технические характеристики аналоговых входов приведены в таблице 1.1.

Таблица 3.69 – Модуль CPU_IO. Сигналы аналогового входа

Имя	Тип	Подключение		Описание
		Разъем	Вход	
<i>AIn1</i>	Real	AIN1	1...4	Значение измерительного канала 1 (В, мА, °С)
<i>AIn1Diag</i>	Byte	-	-	Диагностика работы измерительного канала 1
<i>AIn2</i>	Real	AIN2	1...4	Значение измерительного канала 2 (В, мА, °С)
<i>AIn2Diag</i>	Byte	-	-	Диагностика работы измерительного канала 2
<i>AIn3</i>	Real	AIN3	1...4	Значение измерительного канала 3 (В, мА, °С)
<i>AIn4Diag</i>	Byte	-	-	Диагностика работы измерительного канала 3
<i>AIn4</i>	Real	AIN4	1...4	Значение измерительного канала 4 (В, мА, °С)
<i>AIn4Diag</i>	Byte	-	-	Диагностика работы измерительного канала 4

ВАЖНО! Единицы измерения входных сигналов зависят от установленного режима работы (В, мА, °С), при этом подключенные датчики и схема подключения должны соответствовать установленному режиму.

Таблица 3.70 – Модуль CPU_IO. Диагностика каналов аналогового входа (AIn1Diag...AIn4Diag)

Бит	Значение	Описание
0	0	Канал обрабатывается
	1	Канал не обрабатывается (задан параметр "Disable")
1	0	Измеренное значение находится в диапазоне измерения
	1	Измеренное значение находится вне диапазона измерения
2	0	Нормальная работа АЦП
	1	Ошибка работы с АЦП (ошибка SPI_ERR)
...
7	0	Было обновление измеренного значения
	1	Не было обновления измеренного значения. Возможно, не работает суббмодуль аналогового ввода. Бит сбрасывается в нулевое значение при первом корректном приеме данных от субмодуля

На рисунке 3.95 приведен пример кода для описания переменных аналогового входа.

```

22 (* Дискретный вывод *)
23 DigOutMax : INT := 8; (* Количество сигналов Дискретного вывода *)
24 myDigOut : ARRAY[1..8] OF BOOL; (* сигналы Дискретного вывода *)
25
26 (* Аналоговый ввод *)
27 AnalInMax : INT := 4; (* Количество сигналов Аналогового ввода *)
28 myAnalIn : ARRAY[1..4] OF REAL; (* сигналы Аналогового ввода *)
29 myAInDiag : ARRAY[1..4] OF BYTE; (* диагностика каналов Аналогового ввода *)
30
31 (* Аналоговый вывод *)
32 AnalOutMax : INT := 2; (* Количество сигналов Аналогового вывода *)
33 myAnalOut : ARRAY[1..2] OF REAL; (* сигналы Аналогового вывода *)

```

Рисунок 3.95 – Сигналы аналогового входа. Пример описания переменных

Сигналы могут быть описаны в виде массива переменных или в виде отдельных переменных. Тип переменных должен быть *REAL* для значений сигналов и *Byte* для сигналов диагностики.

Период измерения каждого канала зависит от выбранных параметров работы модуля, а именно режим измерения и режим интегрирования для каждого из каналов. Обработка каналов ведется последовательно. Время, затрачиваемое на обработку одного канала, можно рассчитать по формуле:

$$t_{\text{изм, мс}} = MF * ST,$$

где $MF = 25$, если *ModeFrec = Disable* и $MF = 150$, если *ModeFrec = Enable*;

$ST = 0$, если вход отключен;

$ST = 1$, если вход сконфигурирован для измерения тока, напряжения или термопары без термокомпенсации;

$ST = 2$, если вход сконфигурирован для измерения термопары с термокомпенсацией;

$ST = 3$, если вход сконфигурирован для измерения термосопротивления;

Период съема значений определяется как сумма времени, затраченная модулем на обработку каждого из каналов.

Пример 1.

Модуль сконфигурирован в режиме измерения тока по первому каналу и напряжения по второму каналу. Остальные каналы отключены, ModeFrec = Disable. Тогда период съема значений будет равен:

$$T = 25 \text{ мс} * 1 + 25 \text{ мс} * 1 = 50 \text{ мс.}$$

Пример 2.

Модуль сконфигурирован для подключения термосопротивления ТСМ 50М, ModeFrec = Enable для всех каналов. Тогда период съема значений будет равен:

$$T = 150 \text{ мс} * 3 * 4 = 1800 \text{ мс.}$$

Измеренные значения поступают на вход фильтра первого порядка, пересчитывающего измеренные значения по формуле:

$$U_{\text{вых}} = U_{\text{вых-1}} * (1 - \text{Coeff}) + U_{\text{изм}} * \text{Coeff},$$

где $U_{\text{вых}}$ – выходное значение фильтра;

$U_{\text{вых-1}}$ – выходное значение фильтра на время получения предыдущего входного отчета;

$U_{\text{изм}}$ – измеренное значение величины, поступающее на вход фильтра

Coeff – коэффициент фильтрации, задаваемый в конфигурации на каждый из измерительных каналов. Чем меньше коэффициент фильтрации, тем дольше будет нарастать выходное значение фильтра при скачкообразном повышении сигнала на входе. В таблице 3.71 приведено соответствие заданного коэффициента и необходимое количество отсчетов до получения выходного значения равного 0,9 от реального и 0,995 от реального.

Таблица 3.71 – Выходные значения коэффициента фильтрации

<i>Coeff</i>	Число отсчетов до уровня 0,9	Число отсчетов до уровня 0,995
1	1	1
0,1	22	52
0,01	230	528

Для того чтобы определить, за какое время статический сигнал на входе модуля будет измерен с заданной точностью, необходимо умножить период съема значений для заданной конфигурации на необходимое число отсчетов для достижения заданной точности. Так при заданном коэффициенте *Coeff* = 0,1 и конфигурации, описанной в *Примере 1*, время измерения составит 50 мс * 22 = 1100 мс, а для *Coeff* = 1 составит 50 мс.

3.17.4 Сигналы аналогового выхода

В контроллере сигналы аналогового выхода могут работать в следующих режимах:

- выход напряжения;
- выход тока.

Выбор режима работы каждого канала и необходимые параметры задаются на вкладке *Редактор параметров*. На рисунке 3.96 представлен внешний вид вкладки и выделены параметры, отвечающие за обработку сигналов аналогового выхода.

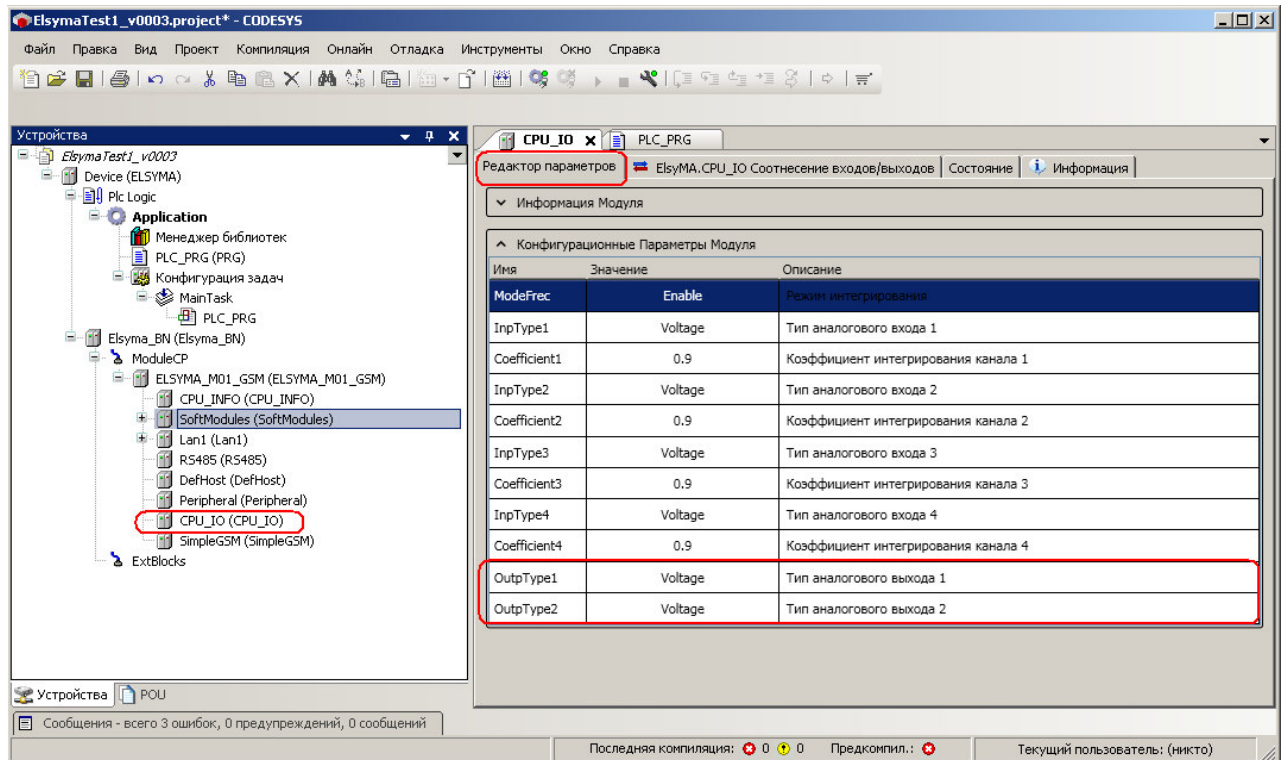


Рисунок 3.96 – Сигналы аналогового выхода.
Закладка *ElsyMA.CPU_IO Соотнесение входов/выходов*

ВАЖНО! При выборе определенного режима работы аналогового выхода необходимо учитывать схему подключения для выбранного режима. Схемы подключения для различных режимов приведены на рисунках А.7 и А.8.

Параметры для работы с каналами аналогового выхода модуля *ELSYMA_M01_XXX-CPU_IO* перечислены в таблице 3.72. Параметры доступны для редактирования пользователем только в режиме **off-line**.

Таблица 3.72 – Модуль CPU_IO. Параметры модуля (Аналоговый выход)

Имя	Значение "по умолчанию"	Описание
<i>OutType1</i>	Voltage	Тип аналогового выхода 1. Доступное значение "Voltage" – формирование напряжения, "Current" – формирование тока
<i>OutType2</i>	Voltage	Тип аналогового выхода 2. Доступное значение "Voltage" – формирование напряжения, "Current" – формирование тока.

На рисунке 3.94 представлен вид закладки *ElsyMA.CPU_IO Соотнесение входов/выходов*. Сигналы аналогового выхода сгруппированы в папке "Analog Outputs".

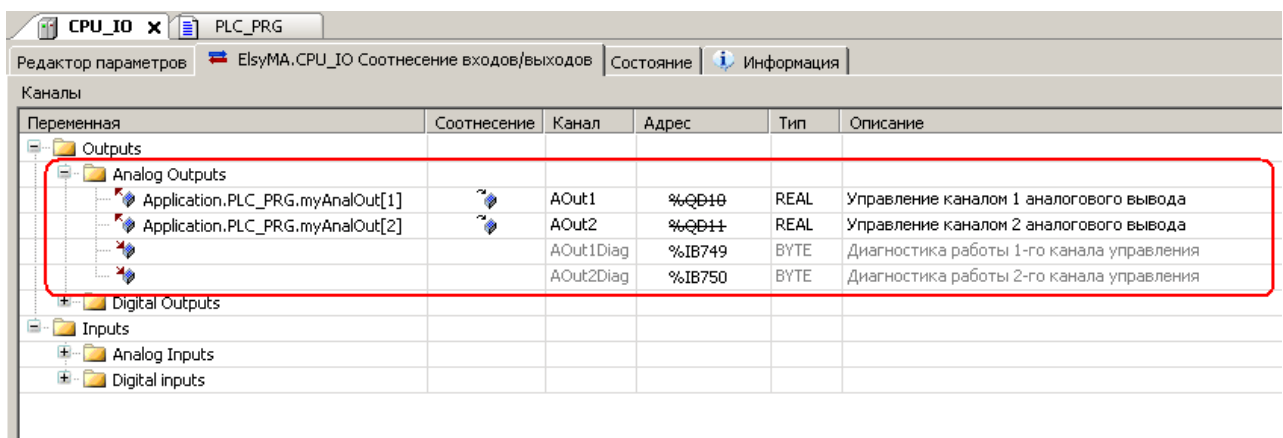


Рисунок 3.97 – Сигналы аналогового выхода.
Закладка *ElsyMA.CPU_IO Соотнесение входов/выходов*

Описание сигналов аналогового выхода и соответствие с физическим выходом представлено в таблице 3.73. Схемы подключения сигналов приведены на рисунках А.7 и А.8. Технические характеристики аналоговых выходов приведены в таблице 3.73.

Таблица 3.73 – Модуль CPU_Ю. Сигналы аналогового выхода

Имя	Тип	Подключение		Описание
		Разъем	Выход	
AOut1	Real	AOUT	1	Управление каналом 1 аналогового выхода (В, мА)
AOut2	Real	AOUT	2	Управление каналом 2 аналогового выхода (В, мА)
AOut1Diag	Byte	-	-	Диагностика работы канала управления 1
AOut2Diag	Byte	-	-	Диагностика работы канала управления 2

ВАЖНО! Единицы измерения выходных сигналов зависят от установленного режима работы (В, мА), при этом подключенные датчики и схема подключения должны соответствовать установленному режиму.

Значения бит сигналов диагностики аналогового выхода представлены в таблице 3.74.

Таблица 3.74 – Модуль CPU_Ю. Диагностика каналов аналогового выхода (AOut1Diag...AOut2Diag)

Бит	Значение	Описание
0	0	Канал обрабатывается.
	1	Канал не обрабатывается (задан параметр Disable).
1	0	Заданное значение находится в диапазоне формирования выхода.
	1	Заданное значение находится вне диапазона формирования выхода.
2	0	Нормальная работа ЦАП.
	1	Ошибка работы с ЦАП (ошибка SPI_ERR).
3	0	Норма (канал калиброван или калибровка не требуется)
	1	Ошибка калибровки (CRC разрушена)
4	0	Канал калиброван
	1	Канал не калиброван (используются коэффициенты по умолчанию). Устанавливается при установке бита 3.

На рисунке 3.98 приведен пример кода для описания переменных аналогового выхода.

```

28 myAnalIn : ARRAY[1..4] OF REAL; (* сигналы Аналогового ввода *)
29 myInDiag : ARRAY[1..4] OF BYTE; (* диагностика каналов Аналогового ввода *)
30
31 (* Аналоговый вывод *)
32 AnalOutMax : INT := 2; (* Количество сигналов Аналогового вывода *)
33 myAnalOut : ARRAY[1..2] OF REAL; (* сигналы Аналогового вывода *)
34 myAOutDiag : ARRAY[1..2] OF BYTE; (* диагностика каналов Аналогового вывода *)
35 myOut1 : REAL;
36 myOut2 : REAL;

```

Рисунок 3.98 – Сигналы аналогового входа. Пример описания переменных

Сигналы могут быть описаны в виде массива переменных или в виде отдельных переменных. Тип переменных должен быть **REAL** для значений сигналов и **Byte** для сигналов диагностики.

3.18 Работа со сканерами QR и штрих-кода

3.18.1 Основные функции ПО

Для обеспечения информационного взаимодействия ПЛК со сканерами QR или штрих-кода в ПЛК реализована поддержка устройств такого типа. Это обеспечивает выполнение следующих функций в контроллере:

- Взаимодействие с физическим интерфейсом USB для получения списка подключенного оборудования;
- Взаимодействие с выбранным из полученного списка устройством (сканером);
- Предварительная обработка полученной информации (формирование целостной строки данных);
- Передача в задачу пользователя полученной информации от сканера.

3.18.2 Подключение сканера к контроллеру

Сканеры QR или штрих-кода подключаются к контроллеру ЭЛСИМА через интерфейс **USB1**. Предусмотрена возможность подключения нескольких сканеров к контроллеру через расширитель USB интерфейса. Допускается подключение до 16 устройств на один USB порт.

ВАЖНО! Сканеры QR или штрих-кода должны быть настроены для работы в следующих режимах:

- Standart Product Defaults;
- USB Serial;
- Add CR Suffix.

3.18.3 Работа с ФБ библиотеки USBScanLibrary

Для работы со сканерами QR и штрих-кодов в состав пакета конфигуратора ПЛК включена библиотека **USBScanLibrary**. Для применения в задаче пользователя *CoDeSys* из этой библиотеки доступны функциональные блоки **GetUSBDevices** и **USBScan**.

ВАЖНО! Библиотеку **USBScanLibrary** необходимо добавлять в проект вручную.

ФБ **GetUSBDevices** предназначен для получения списка подключенных к **USB1** порту устройств контроллера ЭЛСИМА. Информация, полученная с помощью ФБ **GetUSBDevices**, может использоваться для выбора необходимых для работы устройств.

ВАЖНО! При вызове ФБ **GetUSBDevices** производится вызов системных функций операционной системы для сканирования подключенных устройств и получения от них доступной информации. Обработка данного вызова занимает примерно 25-30 мс на одно подключенное устройство. Данную особенность необходимо учитывать при создании программы.

Интерфейс ФБ **GetUSBDevices** представлен на рисунке 3.99. Параметры функционального блока представлены в таблице 3.75.

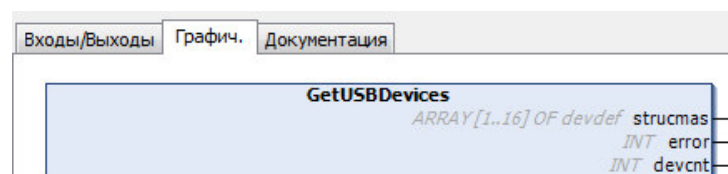


Рисунок 3.99 – Функциональный блок GetUSBDevices

Таблица 3.75 – Параметры функционального блока GetUSBDevices

Имя	Тип	Описание
<i>strucmas</i>	Array [1..16] of devdef	Массив из 16 структур типа devdef найденных USB устройств
<i>error</i>	Int	Код завершения вызова функционального блока: <i>0</i> – блок не вызывался <i>1</i> – нет ошибки <i>-1</i> – системная ошибка (ошибка создания объекта для доступа к

Таблица 3.75 – Параметры функционального блока **GetUSBDevices**

Имя	Тип	Описание
		подсистеме управления устройствами linux) -2 – системная ошибка (ошибка создания enumerate для доступа к подсистеме управления устройствами Linux)
<i>devcnt</i>	Int	Количество найденных устройств

Структура **devdef** служит для описания подключенного к USB порту устройства и содержит поля, необходимые для работы с этим устройством. Параметры структуры представлены в таблице 3.76.

Таблица 3.76 – Параметры структуры **devdef**

Имя	Тип	Описание
<i>devdef</i>	String(81)	Описатель устройства – поле зарезервировано
<i>manufacturer</i>	String(81)	Имя производителя
<i>product</i>	String(81)	Название продукта
<i>sernmb</i>	String(81)	Серийный номер (если данная информация доступна в устройстве)
<i>vid</i>	String(81)	Vendor ID (если данная информация доступна в устройстве)
<i>pid</i>	String(81)	Product ID (если данная информация доступна в устройстве)
<i>portnmb</i>	String(81)	Логический порт (например, ttyUSB0)

ФБ **USBScan** предназначен для работы со сканерами QR или штрих-кода, подключенными к **USB1** порту контроллера ЭЛСИМА. Для работы с ФБ **USBScan** необходимо задать логический порт используемого устройства (параметр *portnmb*). Например, при подключении сканера **Metrologic MS160** непосредственно к интерфейсу **USB1**, данный параметр должен принимать значение **'/dev/ttyUSB0'**. В общем случае, значение для данного параметра можно получить с помощью ФБ **GetUSBDevices** (см. пример в приложении К). Параметры *sernmb*, *vid*, *pid* передаются в ФБ для использования метода **CheckState** (см. ниже). Код завершения вызова ФБ **USBScan** и текущей ошибки:

Код завершения вызова ФБ **USBScan** и текущей ошибки:

0 – Блок не вызывается;

1 – Нет ошибки, заданный порт открыт;

-1 – Ошибка открытия порта (ошибка вызова open).

Интерфейс ФБ **USBScan** представлен на рисунке 3.100. Параметры функционального блока представлены в таблице 3.77.

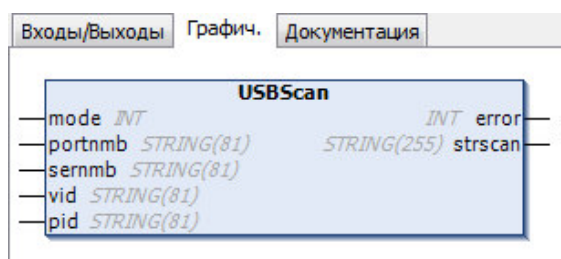
Рисунок 3.100 – Функциональный блок **USBScan**

Таблица 3.77 – Параметры функционального блока USBScan

Имя	Тип	Описание
<i>mode</i>	Int	Режим работы блока (зарезервировано для дальнейшего использования. Должен передаваться 0-м значением)
<i>portnmb</i>	String(81)	Логический порт (например, ttyUSB0, может быть получен через блок GetUSBDevices)
<i>sernmb</i>	String(81)	Серийный номер (может быть получен через блок GetUSBDevices)
<i>vid</i>	String(81)	Vendor ID (если данная информация доступна в устройстве)
<i>pid</i>	String(81)	Production ID (если данная информация доступна в устройстве)
<i>error</i>	Int	Код завершения вызова функционального блока и текущей ошибки: 0 – блок не вызывался 1 – нет ошибки, заданный порт открыт -1 – ошибка открытия заданного порта (ошибка вызова open)
<i>strscan</i>	String(255)	Полученная от сканера строка данных

В состав ФБ **USBScan** входят методы **CheckState** и **Receive**.

Метод **CheckState** (тип **Int**) должен вызываться из задачи для проверки работоспособности подключенного устройства. Проверка проводится по параметрам *sernmb*, *vid*, *pid*. Данные параметры могут быть не заданы, в этом случае метод **CheckState** проверяет только состояние порта заданным параметром *portnmb* (открыт или закрыт). Метод **CheckState** возвращает следующие коды завершения:

- 0** – Все нормально (порт открыт, готов к работе, серийный номер, *vid*, *pid* соответствуют заданным);
- 1** – Системная ошибка (ошибка создания объекта для доступа к подсистеме управления Linux);
- 2** – Системная ошибка (ошибка создания **enumerate** для доступа к подсистеме управления устройствами Linux);
- 3** – Ошибка, возникающая, если ни разу не вызван блок **USBScan**;
- 4** – Заданный параметр не доступен;
- 5** – Заданный в параметре *vid* номер не соответствует реальным данным (если параметр задан);
- 6** – Заданный в параметре *pid* номер не соответствует реальным данным (если параметр задан);
- 7** – Заданный в параметре *sernmb* серийный номер не соответствует реальным данным (если параметр задан).

ВАЖНО! При вызове метода **CheckState** производится вызов системных функций операционной системы. Обработка данного вызова занимает примерно 25-30 мс. Данную особенность необходимо учитывать при создании программы. Не рекомендуется вызывать метод **CheckState** в каждом цикле задачи.

Метод **Receive** (тип **Int**) служит для выполнения действий по получению данных из устройства. Метод должен вызываться циклически из задачи. Метод возвращает следующие значения:

- 0** – Нет принятых данных;

1..n – Количество принятых символов;

-1 – Порт закрыт (возникает при ошибке открытия порта или если не вызывался блок USBScan);

-2 – Системная ошибка (ошибка вызова poll);

-3 – Системная ошибка (ошибка вызова read).

Пример программного кода для работы со сканерами представлен в приложении К.

3.19 Поддержка режима GPRS GSM модуля

3.19.1 Назначение режима GPRS

Для обеспечения более широких коммуникационных возможностей в ПЛК ЭЛСИМА предусмотрена возможность работы модуля **GSM** в **GPRS** режиме, что позволяет подключить контроллер к сети Интернет. В данном режиме выполняются следующие функции:

- Настройка и начальная инициализация встроенного модема для работы в режиме **GPRS**;
- Предоставлять возможность коммуникации программного модуля (ModBus TCP Master/Slave, IEC104 Master/Slave, **tsync**) через **GPRS**;
- Предоставлять статистику и диагностику работы **GPRS** режима встроенного модема;
- Предоставлять возможность приема-передачи SMS сообщений (в том числе возможность формирования AT-команды) одновременно с работой в **GPRS** режиме;

3.19.2 Варианты использования модуля GSM в GPRS режиме

При включении **GPRS** режима появляется возможность подключения ПЛК к Internet через встроенный **GSM** модуль. Так же, как и для работы через LAN интерфейс, взаимодействие программных модулей производится через **CommSlot** (см. 3.12.4). На рисунке 3.101 приведен пример конфигурирования работы ModbusTCP через встроенный **GSM** модуль.

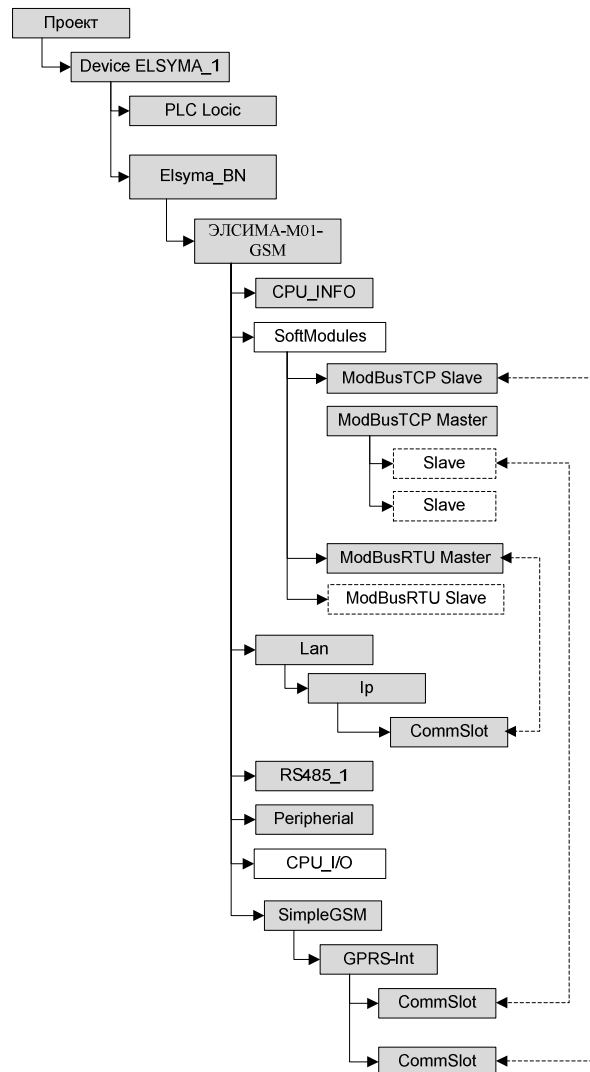


Рисунок 3.101 – Варианты применения GPRS режима

3.19.3 Конфигурирование работы модуля GSM в GPRS режиме

Для настройки конфигурации модуля GPRS необходимо:

- 1) В дереве устройств найти *SimpleGSM*, нажать правой кнопкой мыши и в контекстном меню выбрать команду *Добавить устройство....*
- 2) В появившемся окне выбрать **Оборудование компании ЭлеСи->ПЛК->IP->GPRS_Int** и нажать кнопку «Добавить устройство».
- 3) В дереве устройств найти *GPRS_Int*, нажать правой кнопкой мыши и в контекстном меню выбрать команду *Добавить устройство....*
- 4) В появившемся окне выбрать **Оборудование компании ЭлеСи->ПЛК->IP->Communication Slot->CommSlot** и нажать кнопку «Добавить устройство».
- 5) В дереве устройств найти *CommSlot* и дважды нажать левой кнопкой мыши. В появившейся закладке выбрать вкладку **Редактор соединения**. В графе сервер можно установить связь с программными модулями. На рисунке 3.102 представлен пример установки связи с программным модулем IEC104M.

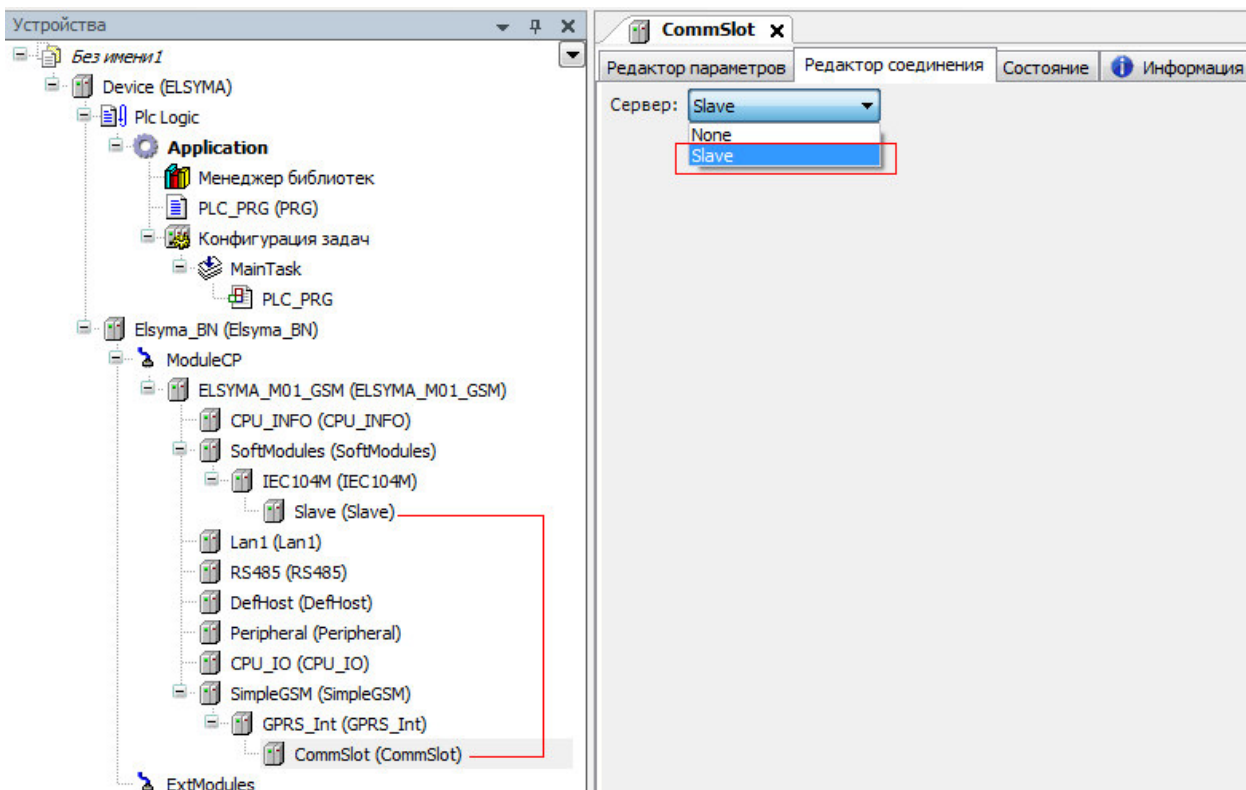


Рисунок 3.102 – Дерево устройств. Конфигурация GPRS

Параметры **GPRS** представлены на вкладке *Редактор параметров* устройства *GPRS_Int* (рисунок 3.103). Перечень параметров и их описание представлены в таблицах 3.78 и 3.79.

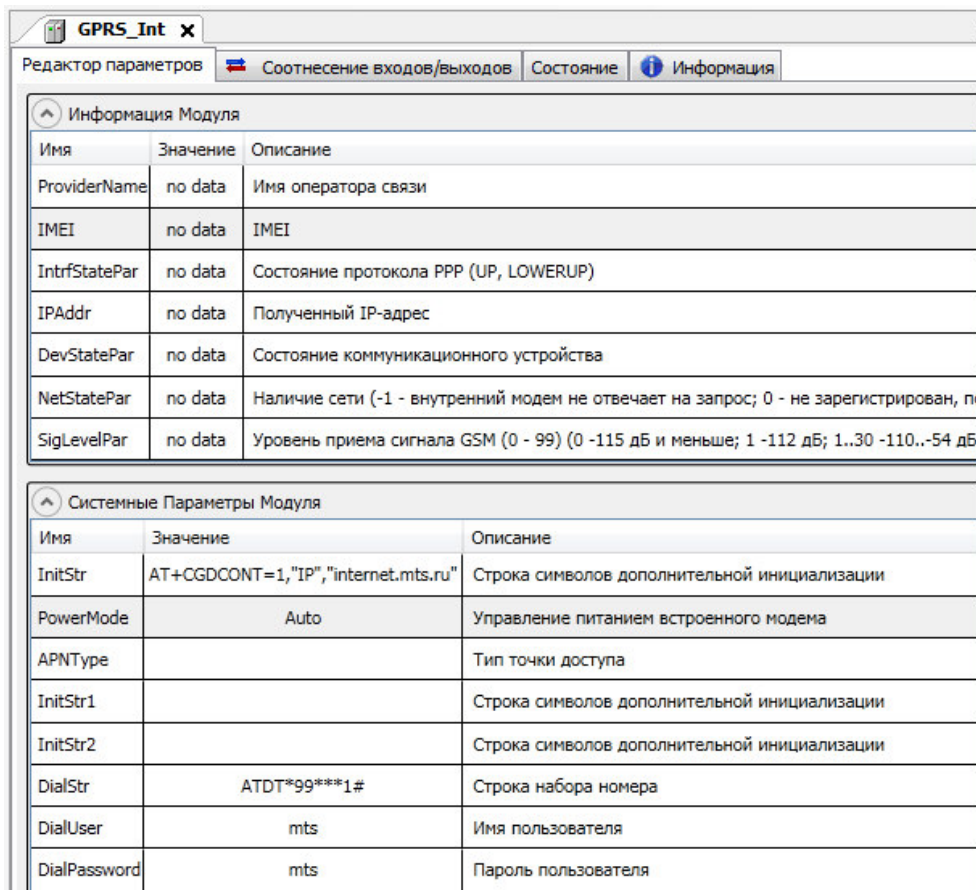


Рисунок 3.103 - Вкладка Редактор параметров устройства GPRS_Int

Таблица 3.78 – Информация Модуля

Имя	Значение	Описание
<i>ProviderName</i>	no data	Имя оператора связи
<i>IMEI</i>	no data	IMEI
<i>IntrfStatePar</i>	no data	Состояние протокола PPP (UP, LOWERUP)
<i>IPAddr</i>	no data	Полученный IP
<i>DevStatePar</i>	no data	Состояние коммуникационного устройства
<i>NetStatePar</i>	no data	Наличие сети: <i>0</i> – не зарегистрирован, поиска сети нет; <i>1</i> – зарегистрирован, домашняя сеть; <i>2</i> – не зарегистрирован, идет поиск новой сети; <i>3</i> – регистрация отключена; <i>4</i> – неизвестно; <i>5</i> – роуминг; <i>-1</i> – внутренний модем не отвечает на запрос
<i>SigLevelPar</i>	no data	Уровень приема сигнала GSM (0-99) (0 – 115 дБ и меньше; 1 - 112 дБ; 1...30 – 110.. – 54 дБ; 31 – 52 дБ и выше)

Таблица 3.79 – Системные Параметры Модуля

Имя	Значение	Описание
<i>InitStr</i>	AT+CGDCONT=1, “IP”, “internet.mts.ru”	Строка символов дополнительной инициализации
<i>PowerMode</i>	Auto	Управление питанием встроенного модема
<i>APNType</i>		Тип точки доступа
<i>InitStr1</i>		Строка символов дополнительной инициализации
<i>InitStr2</i>		Строка символов дополнительной инициализации
<i>DialStr</i>	ATDT*99***1#	Строка набора номера
<i>DialUser</i>	mts	Имя пользователя
<i>DialPassword</i>	mts	Пароль пользователя

3.19.4 Управление работой модуля GSM в GPRS режиме

Для того чтобы просмотреть сигналы управления устройства *GPRS_Int*, необходимо в дереве устройств дважды нажать левой кнопкой мыши на устройство *GPRS_Int*. В появившейся закладке открыть вкладку **Соотнесение входов/выходов** (рисунок 3.104). Набор сигналов диагностики устройства *GPRS_Int* представлен в таблице 3.80.

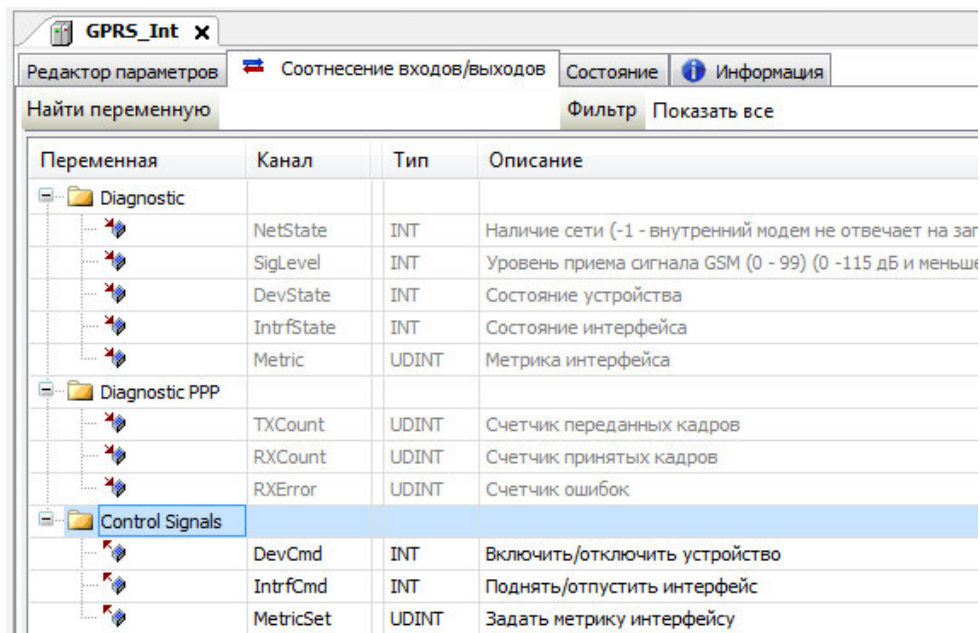


Рисунок 3.104 – GPRS_Int. Вкладка Соотнесение входов/выходов

Таблица 3.80 – Набор диагностических сигналов устройства GPRS_Int

Имя	Тип	Описание
<i>Диагностические сигналы встроенного модема</i>		
<i>NetState</i>	INT	Наличие сети: 0 – не зарегистрирован, поиска сети нет; 1 – зарегистрирован, домашняя сеть; 2 – не зарегистрирован, идет поиск новой сети; 3 – регистрация отключена; 4 – неизвестно; 5 – роуминг; -1 – внутренний модем не отвечает на запрос
<i>SigLevel</i>	INT	Уровень приема сигнала GSM (0-99) (0 – 115 дБ и меньше; 1 - 112 дБ; 1...30 – 110.. – 54 дБ; 31 – 52 дБ и выше)
<i>DevState</i>	INT	Состояние устройства
<i>IntrfState</i>	INT	Состояние интерфейса
<i>Metric</i>	UDINT	Метрика интерфейса
<i>Диагностические сигналы интерфейса PPP</i>		
<i>TXcount</i>	UDINT	Счетчик переданных кадров
<i>RXCount</i>	UDINT	Счетчик принятых кадров
<i>RXError</i>	UDINT	Счетчик ошибок
<i>Сигналы управления</i>		
<i>DevCmd</i>	INT	Включить/отключить устройство
<i>IntrfCmd</i>	INT	Поднять/опустить интерфейс
<i>MetricSet</i>	UDINT	Задать метрику интерфейсу

3.19.5 Пример программного кода

Пример программного кода по работе с **GPRS** представлен в документе «Работа с GPRS через встроенный модем. Краткое руководство пользователя». В данном примере описан проект, в котором с помощью GPRS устанавливается связь между *Modbus TCP Master* контроллера Элсима и *Modbus TCP Slave* стороннего ПК (в примере *Modbus TCP Slave* на ПК реализован с помощью программы *ModSim32*) с последующим получением данных от *Modbus TCP Slave* по GPRS.

3.20 Отображение состояний программных и аппаратных модулей

3.20.1 Отображение состояний программных и аппаратных модулей в журнале CoDeSys

В журнале CoDeSys (устройство *Device*, вкладка *Журнал*) отображается процесс работы системы, в частности, состояния аппаратных модулей (модули УВВ) и программных модулей (tsync, Modbus и др.) в различных ситуациях.

Подробный перечень сообщений о состояниях модулей приведен в приложении Н.

4 КОМПЛЕКТ ПОСТАВКИ

Комплект поставки контроллера приведен в таблице 4.1.

Таблица 4.1 – Контроллер Элсима. Комплект поставки

Наименование	Количество
1 Контроллер программируемый логический Элсима ТУ 4210-090-28829549-2016	1 шт.
2 Контроллер программируемый логический Элсима. Паспорт	1 экз.
3 Контроллер программируемый логический Элсима. Гарантийный талон	1 экз.
4 Электронный носитель, содержащий следующие документы и ПО: 4.1 Контроллер программируемый логический Элсима. Руководство по эксплуатации 4.2 Дистрибутив системы программирования <i>CoDeSys</i> 4.3 Пакет поддержки контроллера Элсима в системе <i>CoDeSys EleSy ELSYMA TSP_vXX.XX.XXXX</i> 4.4 Копия сертификата соответствия 4.5 Контроллер программируемый логический Элсима. Поддержка протоколов передачи данных ГОСТ Р МЭК 60870-5-104-2004. Руководство по применению 4.6 Примеры работы с протоколами ГОСТ Р МЭК 60870-5-104-2004 в контроллере 4.7 ПЛК ЭЛСИМА. Краткое руководство пользователя 4.8 Работа с GSM в режиме SMS. Краткое руководство пользователя 4.9 УВВ ЭЛСИМА-А01. Краткое руководство пользователя 4.10 УВВ ЭЛСИМА-D01. Краткое руководство пользователя 4.11 УВВ ЭЛСИМА-DA01. Краткое руководство пользователя 4.12 Функциональный блок СЕ30Х . Краткое руководство пользователя 4.13 Функциональный блок Меркурий 23х. Краткое руководство пользователя	1 шт.
5 Антенна ANT GSM/3G ВУ-3G-03-2 SMA-M (только для исполнения Элсима-M01-ZZZU-GSM)	1 шт.
6 Отвертка-шлиц	1 шт.
7 Резистор С2-29-0,125-100R 0,1%	4 шт.
8 Упаковка	1 компл.

5 ТРАНСПОРТИРОВАНИЕ И ОБСЛУЖИВАНИЕ

5.1 Тара и упаковка

Контроллер упакован в отдельную индивидуальную тару в соответствии с требованиями ГОСТ 23170-78.

Транспортная тара обеспечивает сохранность контроллера при выполнении погрузочно-разгрузочных работ, транспортировании в закрытых транспортных средствах, необходимую защиту от воздействия внешних факторов, а также при хранении у поставщика и потребителя в складских условиях в пределах гарантийного срока хранения.

При поставке в смонтированном виде в составе других устройств (щитов, стоек) способ упаковки контроллера определяется условиями поставки устройств (щитов, стоек).

5.2 Транспортирование и хранение

Транспортирование упакованных контроллеров может осуществляться всеми видами транспорта в закрытых транспортных средствах: крытых автомашинах, крытых вагонах, самолетом, водным транспортом при размещении в трюмах судов.

Не допускается транспортирование контроллеров в негерметизированных и не отапливаемых отсеках самолетов и морским транспортом без специальных упаковочных средств.

На контроллер в транспортной таре допускается воздействие следующих климатических и механических факторов:

- температура окружающего воздуха – от минус 55 до плюс 70 °С;
- относительная влажность воздуха – от 5 до 100 % без конденсации;
- синусоидальная вибрация по группе F3 ГОСТ Р 52931-2008;
- свободное падение с высоты согласно ГОСТ Р 52931-2008.

Упакованные контроллеры должны быть закреплены в транспортных средствах и защищены от атмосферных осадков и брызг воды.

Размещение и крепление в транспортном средстве должно обеспечить устойчивое положение контроллеров, исключать возможность ударов друг о друга, а также о стенки транспортного средства. Допускается транспортирование с использованием контейнеров.

При соблюдении условий механических воздействий, соответствующих рабочим, контроллер может транспортироваться в составе законченных систем управления (например, стоек или шкафов).

Условия хранения контроллера в упаковке предприятия-изготовителя у поставщика и потребителя должны соответствовать категории 2 (С) по ГОСТ 15150-69.

5.3 Калибровка

Порядок проведения калибровки приведен в документе "Контроллер программируемый логический Элсима. Методика калибровки". Результаты первичной и периодических калибровок заносятся в паспорт на контроллер.

5.4 Техническое обслуживание

С целью обеспечения постоянной исправности и готовности контроллера к эксплуатации необходимо не реже, чем один раз в год проводить техническое обслуживание.

Порядок технического обслуживания:

1 Отключить питание контроллера.

2 Отстыковать от контроллера все подключенные кабели.

3 Промыть контакты разъемов составных частей контроллера этиловым ректифицированным техническим спиртом по ГОСТ Р 55878-2013. При промывке контакты разъемов должны находиться в вертикальном положении. Норма расхода спирта – 0,05 л на 100 контактов.

4 Просушить на воздухе не менее 30 минут.

5 Подключить кабели, подать питание на контроллер.

5.5 Текущий ремонт

Ремонт контроллера должен осуществляться предприятием-изготовителем или специализированным предприятием, имеющим соответствующее оборудование и подготовленный персонал.

Для передачи контроллера на гарантийный ремонт потребитель должен выслать по адресу предприятия-изготовителя отказавший контроллер в заводской упаковке, с паспортом и с указанием характера отказа и обстоятельств его возникновения.

По истечении гарантийного срока ремонт проводится за счет потребителя.

6 Решение проблем

В случае возникновения проблем при работе с контроллером, обратиться к документации. Если проблему не удастся решить самостоятельно, необходимо обратиться к поставщику контроллера (см. контактную информацию на предпоследней странице настоящего руководства по эксплуатации).

Приложение А (справочное)

Схемы подключения сигналов контроллера

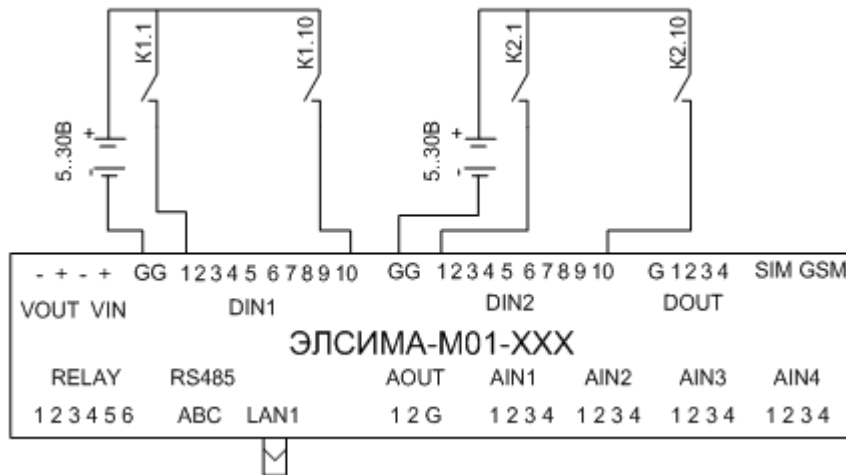


Рисунок А.1 – Подключение входных дискретных сигналов

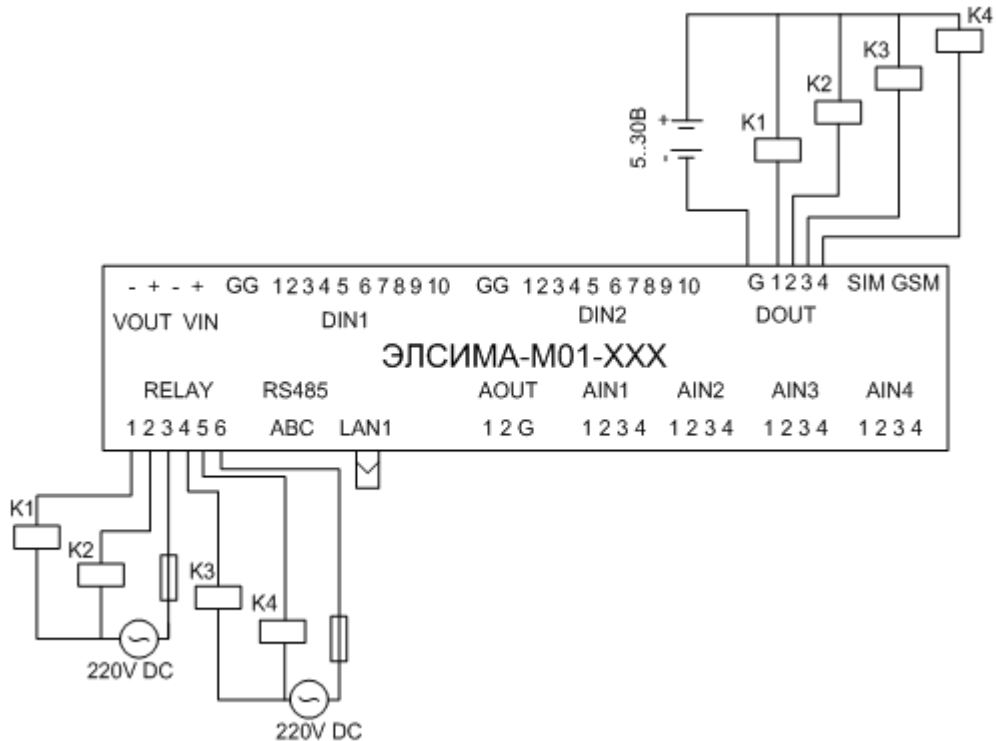


Рисунок А.2 – Подключение сигналов дискретного выхода

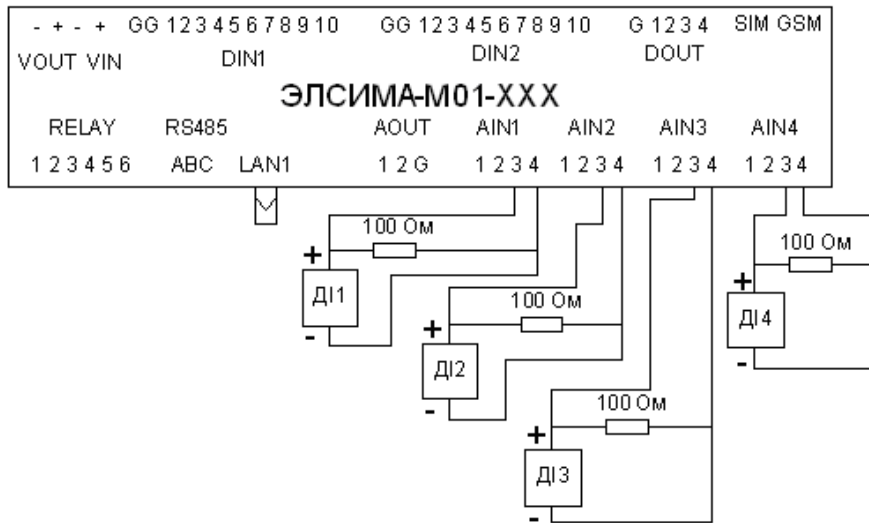


Рисунок А.3 – Подключение датчиков тока

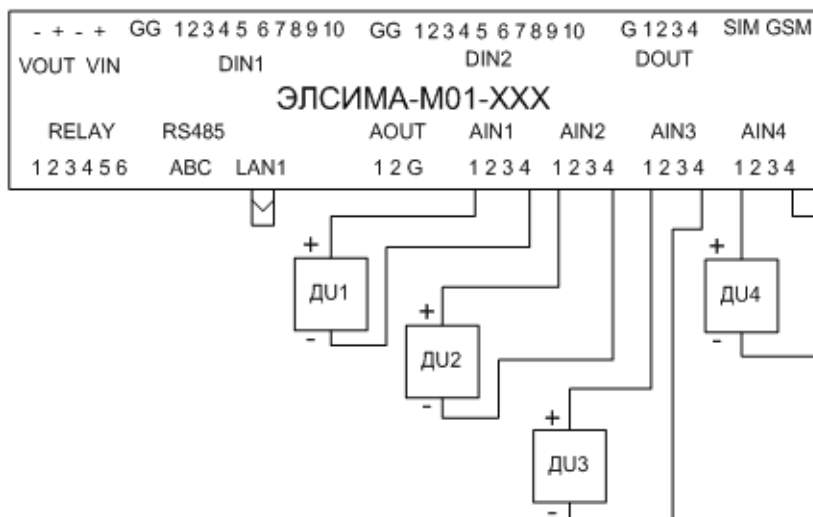


Рисунок А.4 – Подключение датчиков напряжения

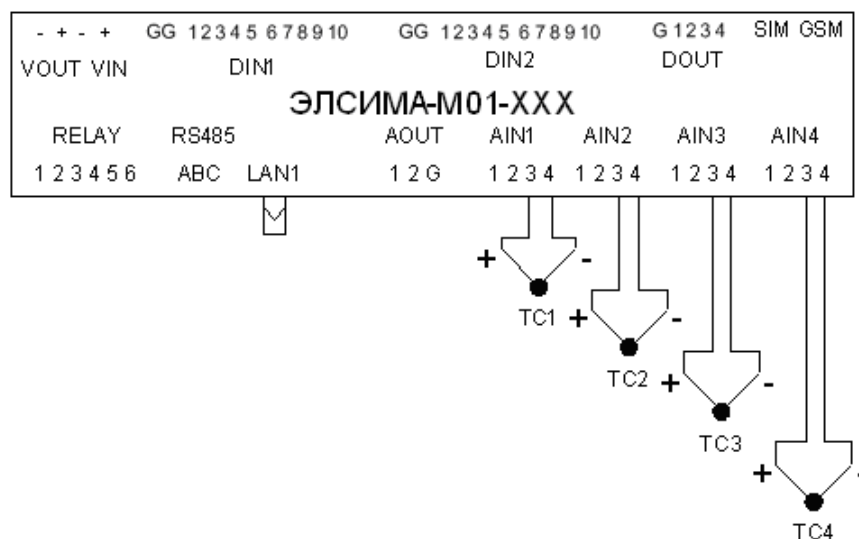


Рисунок А.5 – Подключение датчиков терморпар

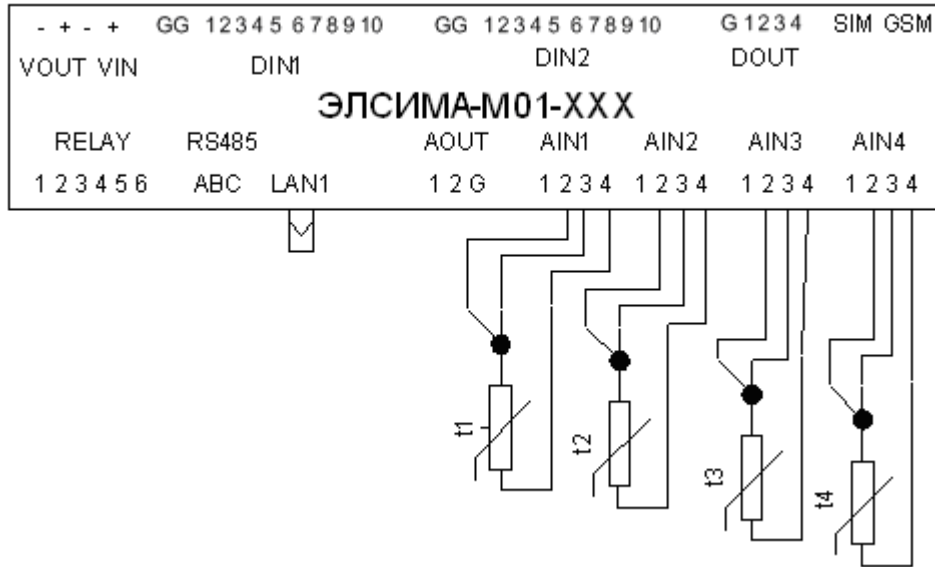


Рисунок А.6 – Подключение датчиков термосопротивлений

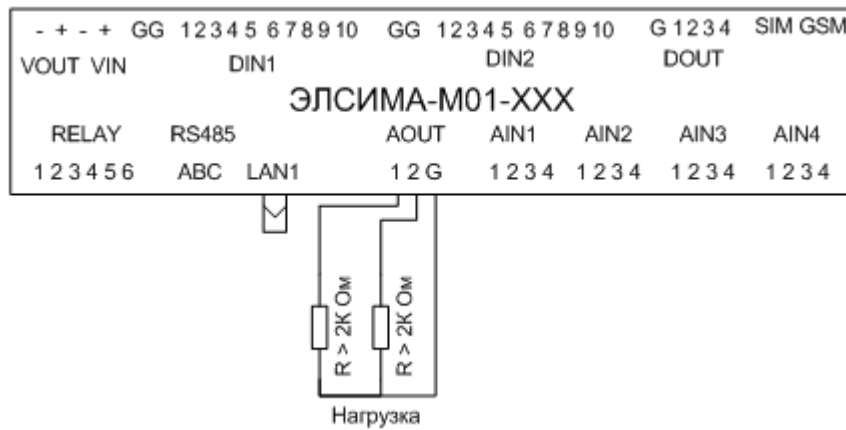


Рисунок А.7 – Подключение аналогового выхода в режиме напряжения

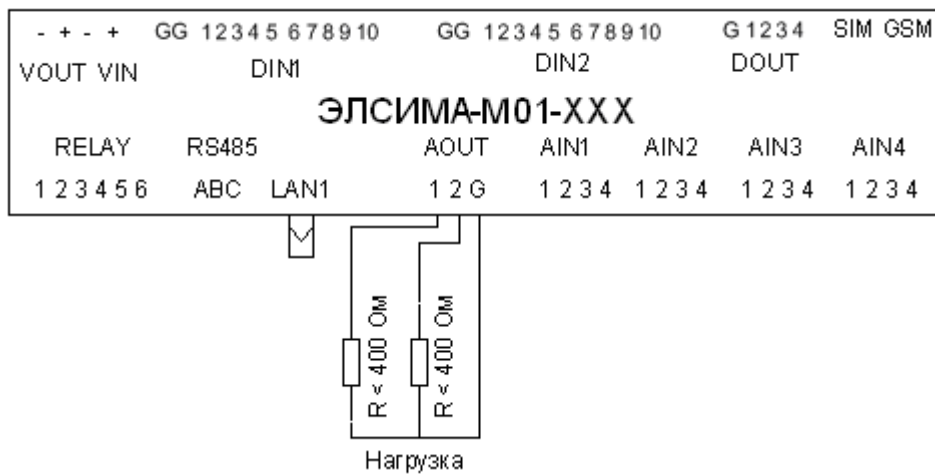


Рисунок А.8 – Подключение аналогового выхода в режиме тока

Приложение Б (справочное)

Изменение сетевых параметров контроллера

Изменение сетевых параметров контроллера реализуется следующим образом:

1 Выключить питание контроллера.

2 Заблокировать работу **WatchDog**-таймера с помощью установки переключателя **SW "1"** в положение **"ON"**.

3 Перевести контроллер в режим старта с заданными заводскими настройками с помощью установки переключателя **SW "3"** в положение **"ON"**.

4 Подать питание на контроллер и ожидать около 1 минуты включение непрерывного свечения индикатора **"L1"** зеленым цветом. При этом индикатор **"L2"** светится желтым цветом непрерывно (контроллер находится в режиме настройки сетевых параметров).

5 Подключиться к контроллеру по заводским сетевым параметрам (IP-адрес – **"10.24.1.200"**, маска подсети – **"255.255.254.0"**) с помощью программы *Telnet*. Далее приведен пример изменения статического IP-адреса контроллера с **10.24.1.200** на **10.14.1.53** с адресом шлюза **10.14.0.1**.

6 Запустить приложение *Telnet* с помощью команды системного меню **Windows: Пуск → Программы → Выполнить...**

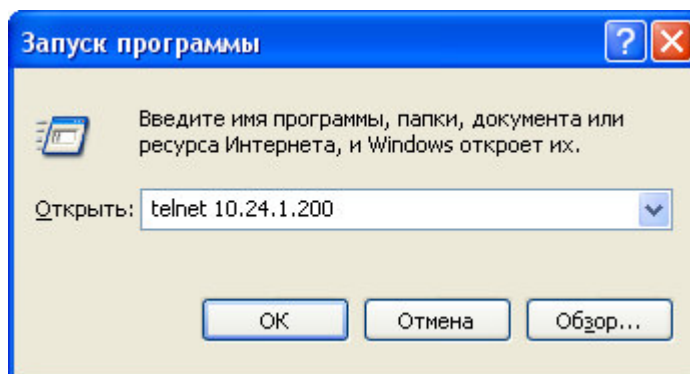


Рисунок Б.1 – Подключение к контроллеру через *Telnet*

7 В поле **Открыть** ввести строку **"telnet 10.24.1.200"** и нажать кнопку **"ОК"**. В окне приложения *Telnet* появится приглашение для регистрации в ОС.

8 В поле **login:** ввести команду **"setip"** и на запрос пароля в поле **Password:** ввести – **"user1pwd"**.

9 Установить переключатель **SW "3"** в положение **"OFF"**. После процедуры изменения сетевых настроек контроллер автоматически перезапускается.

Примечание – После ввода пароля в окне *Telnet* появляется строка «There is no possible change port 1», что говорит о невозможности установки иного порта, кроме 1.

10 В поле **mode:** ввести необходимый режим задания IP-адреса – **"static"**.

11 Подтвердить правильность введенного режима.

Примечание – Установка в поле **mode:** значения *dhcp* завершает процесс изменения сетевых параметров контроллера. Режим DHCP автоматически задает IP-адрес контроллера.

12 В поле **ipaddress:** ввести необходимый IP-адрес – **"10.14.1.53"**.

13 Подтвердить правильность введенного IP-адреса.

14 В поле *networkmask*: ввести необходимую сетевую маску – "255.255.0.0".

15 Подтвердить правильность введенной сетевой маски.

16 В поле *gateway*: ввести необходимый адрес шлюза – "10.14.0.1".

17 Подтвердить правильность введенной сетевой маски.

ВНИМАНИЕ! Замена IP-адреса выполняется только в том случае, если пользователь ввёл значения для полей *ipaddress:*, *gateway:* и *networkmask:*.

Если пользователь не вводил новые значения для полей *ipaddress:*, *gateway:* и *networkmask:*, то остается тот IP-адрес, который был задан до операции.

18 После подтверждения шлюза (*gateway:*) контроллер будет перезапущен (процесс загрузки можно наблюдать на индикаторе). Примерно через 2 минуты можно подключиться к контроллеру по новому адресу.

Пример Б.1.

```

C:\WINDOWS\system32\cmd.exe
Set up service port
Y - apply, n - return to editing, q - changing quit
There is no possible change port 1
itf.0.servlan=1
Set up DHCP mode
Y - apply, n - return to editing, q - changing quit
itf.0.mode=dhcp [static|dhcp] static
itf.0.mode=static [Ynqly]
Set up ip address
Y - apply, n - return to editing, q - changing quit
itf.0.ipaddress=10.14.1.53
itf.0.ipaddress=10.14.1.53 [Ynqly]
Set up network mask
Y - apply, n - return to editing, q - changing quit
itf.0.networkmask=255.255.0.0
itf.0.networkmask=255.255.0.0 [Ynqly]
Set up gateway
Y - apply, n - return to editing, q - changing quit
itf.0.gateway=10.14.0.1
itf.0.gateway=10.14.0.1 [Ynqly]
ln: /home/root/CoDeSysSP/elesylog: File exists
Files /home/root/CoDeSysSP/ExtConfig1.cfg and /var/run/ExtConfig1.cfg differ
ip setting have been changed

The system is going down for reboot NOW! <pts/0> <Fri Dec 14 15:11:19 2018>:

Подключение к узлу утеряно.
C:\Documents and Settings\pischuk.plant>_

```

Приложение В (справочное)

Поддерживаемые типы данных

В таблице В.1 представлен список стандартных типов данных IEC 61131-3, поддерживаемых в контроллере Элсима.

Таблица В.1 – Типы данных

Наименование	Описание
ELSYGSM	Функциональный блок для работы со встроенным модулем GSM
MapIn	Функциональный блок организации процесса приема данных программных модулей и модулей УВВ
MapOut	Функциональный блок организации процесса передачи данных программных модулей и модулей УВВ
ElsyMA_CE30X. CE30X	Функциональный блок для работы со счетчиком Энергомера CE301/302/303/304
ElsyMA_M23X. M23X	Функциональный блок для работы со счетчиком электроэнергии Меркурий 230/233/234
chstat	Статистика работы канала
libstat	Статистика работы библиотеки канала
busstat	Структура основной диагностики работы интерфейса
busstat_ext	Структура расширенной диагностики работы интерфейса
SAI_t	Значение входа измерительного канала 1–8
DAGrOut_t	Группа выходных сигналов
GrSDI_t	Входные дискретные сигналы

Приложение Г (справочное)

Установка драйвера Microsoft RNDIS

Конфигурирование и загрузка проектов в ПЛК средствами *CoDeSys* осуществляется с применением стека протоколов TCP/IP. При этом возможно два варианта подключения:

1 С применением сетевого интерфейса *Ethernet*, путем подключения разъема LAN1 (RJ45) ПЛК сетевым кабелем, через маршрутизатор/роутер или напрямую к ПК.

2 С применением интерфейса USB, путем подключения разъема "mini-USB тип B" ПЛК кабелем USB к ПК с использованием драйвера виртуальной сети RNDIS.

При подключении ПЛК к ПК через интерфейс *Ethernet* используется соответствующий сетевой адаптер, драйвер этого адаптера и стек протоколов TCP/IPv4, входящий в состав ОС.

Для подключения ПЛК через интерфейс USB необходимо применения драйвера удаленного сетевого интерфейса (**Remote Network Driver Interface, RNDIS**), который создаст виртуальный сетевой адаптер с возможностью работы сетевых соединений TCP/IP, как и при использовании *Ethernet*.

Драйвер **RNDIS** является разработкой компании **Microsoft** и входит в состав ОС. ОС **Windows** не всегда удаётся установить корректно драйвер **RNDIS** для устройств, требующих его работы, поэтому потребуется выполнение следующих шагов для установки и конфигурирования виртуальной сети:

1 Подключите ПЛК к ПК кабелем UDB 2.0 "USB A - mini-USB B" (рисунок Г.1) длиной не более 1,8 м и включите питание ПЛК. После инициализации ПЛК и обнаружения нового USB устройства (около 30 с), ОС **Windows** будет выполнять поиск подходящего драйвера **RNDIS** и его установку в автоматическом режиме.

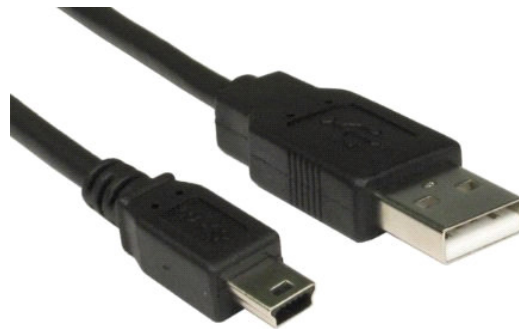


Рисунок Г.1 – Вид разъёмов на кабеле UDB 2.0 "USB A – mini-USB B"

2 Открыть "Диспетчер устройств" (выбором соответствующего пункта в меню "*Мой компьютер*" -> "*Свойства*" или после запуска в командной строке "*mmc compmgmt.msc*") и убедиться в корректной установке драйвера **RNDIS**.

Если установка драйвера завершена некорректно, то в списке "**Другие устройства**" будет находиться "**RNDIS/Ethernet Gadget**", отмеченное значком ошибки и предупреждением в окне состояния устройства "*Для устройства не установлены драйверы. (Код 28), Для этого устройства отсутствуют совместимые драйверы*" (рисунок Г.2).

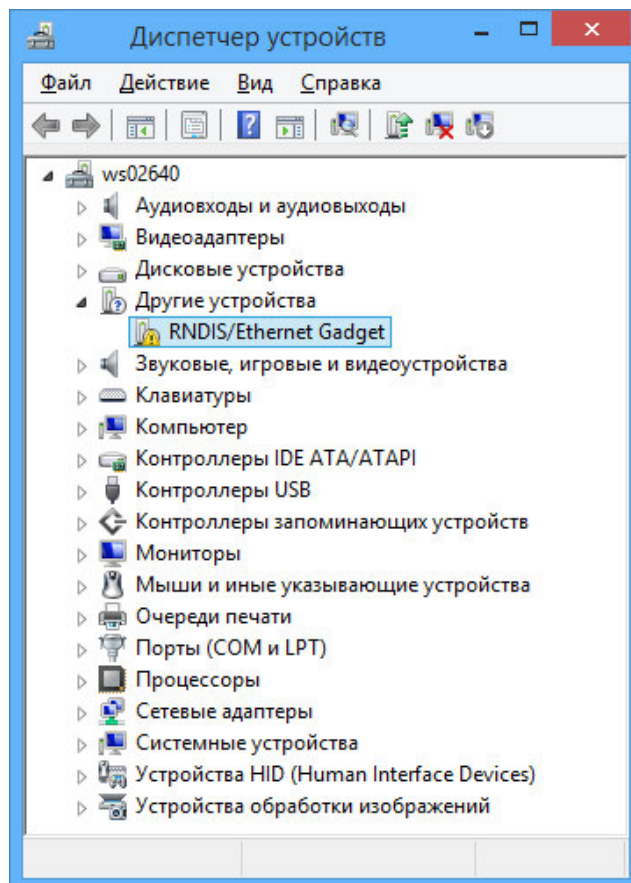


Рисунок Г.2 – Вид "Диспетчер устройств"

3 Откройте свойства "RNDIS/Ethernet Gadget" и во вкладке "Общие" или "Драйвер" нажмите кнопку "Обновить..." (рисунок Г.3).

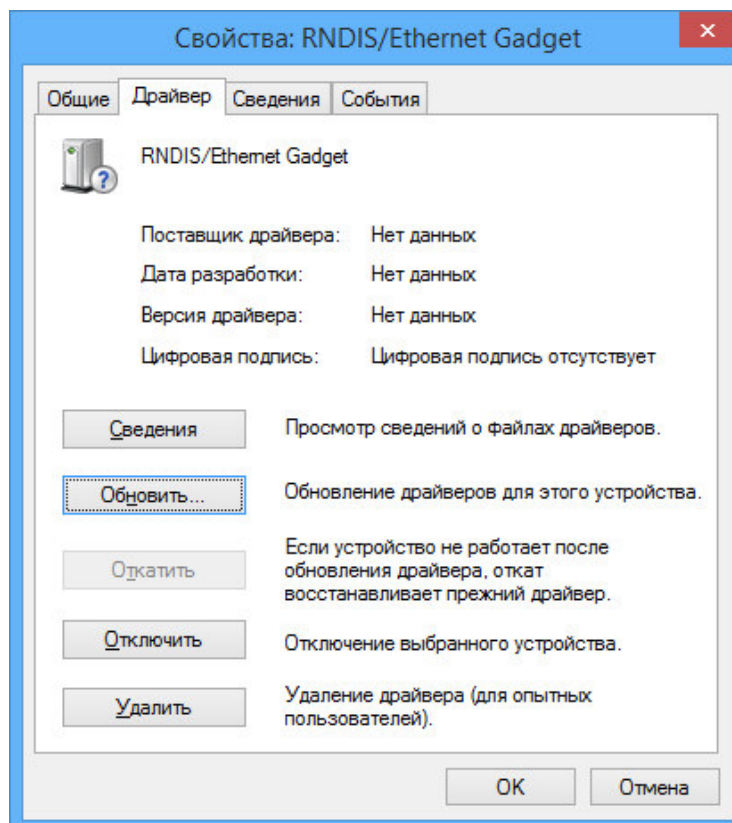


Рисунок Г.3 – Информация о драйвере "RNDIS/Ethernet Gadget"

4 В открывшемся окне обновления выбрать режим поиска "Выполнить поиск драйвера на этом компьютере" (рисунок Г.4).

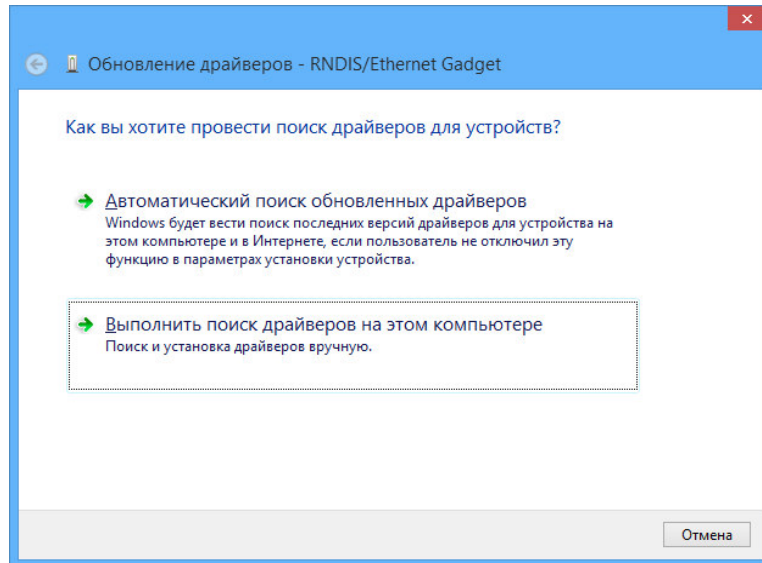


Рисунок Г.4 – Выбор режима поиска драйвера "RNDIS/Ethernet Gadget"

5 Отобразить список драйверов, нажав "Выбрать драйвер из списка уже установленных драйверов ..." в открывшемся окне (рисунок Г.5).

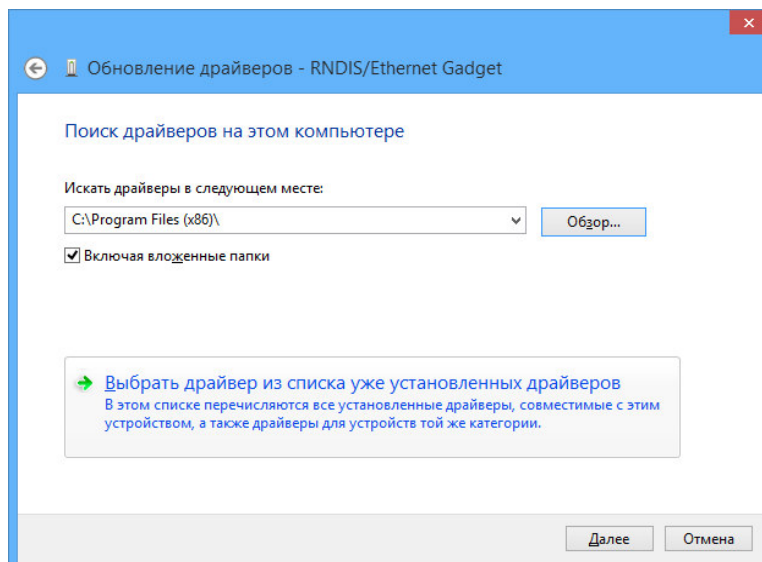


Рисунок Г.5 – Открытие списка установленных драйверов

6 Выбрать из списка раздел "Сетевые адаптеры" (рисунок Г.6).

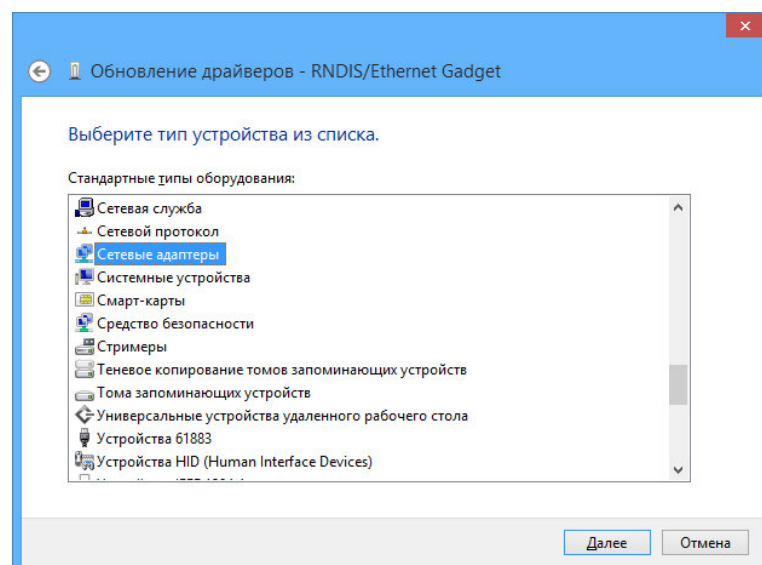


Рисунок Г.6 – Открытие списка драйверов для сетевых адаптеров

7 В разделе "Сетевые адаптеры" выбрать производителя устройств "Microsoft" в левом списке и для этого производителя "Удалённое NDIS-совместимое устройство" из правого списка уже установленных драйверов (рисунок Г.7). Нажать кнопку "Далее".

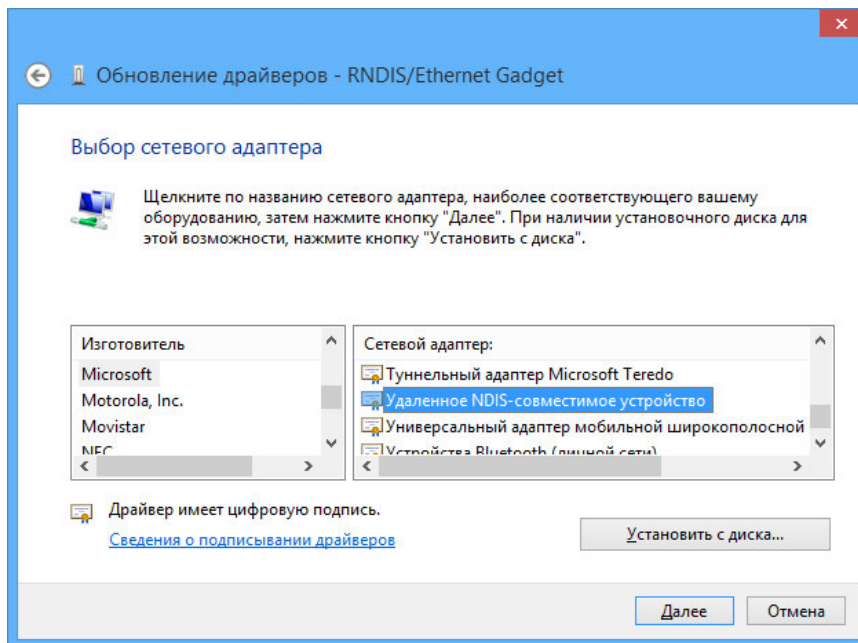


Рисунок Г.7 – Выбор драйвера для "RNDIS/Ethernet Gadget"

8 Дождаться окончания обновления драйвера для устройства "RNDIS/Ethernet Gadget" и появления окна (рисунок Г.8). Нажать кнопку "Заккрыть".

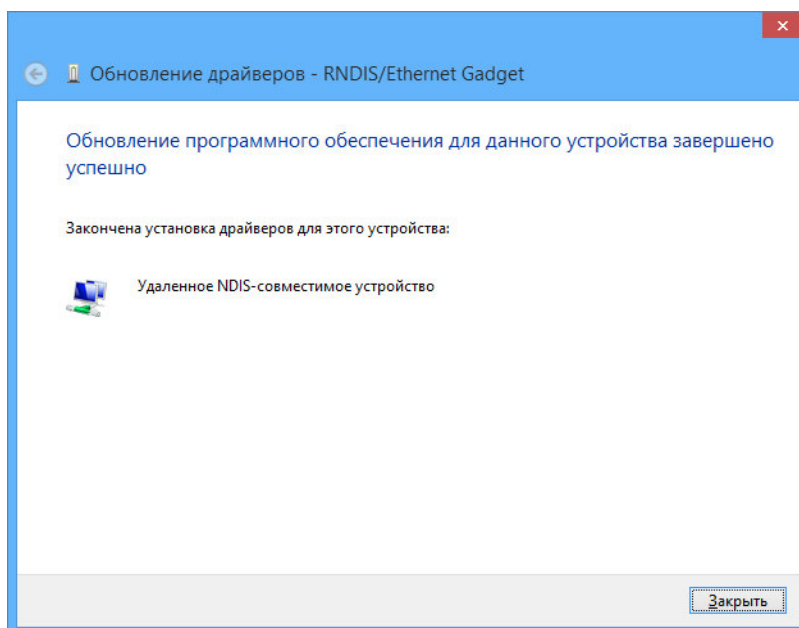


Рисунок Г.8 – Сообщение Выбор драйвера для "RNDIS/Ethernet Gadget"

9 После успешной установке драйвера необходимо проверить доступность устройства "RNDIS/Ethernet Gadget" с помощью "Диспетчер устройств" (рисунок Г.9). Значок ошибки драйвера должен исчезнуть и устройство перемещается из списка нераспознанных устройств ("Другие устройства") в список "Сетевые адаптеры". Изменения в системе могут продолжаться несколько минут, поэтому для полного их завершения и перехода к использованию сети на основе драйвера RNDI рекомендуется сделать паузу 4–5 минут.

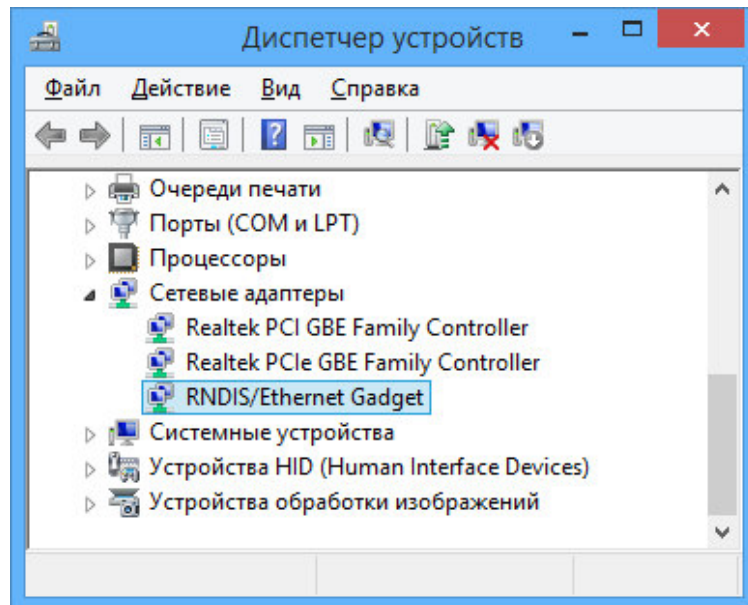


Рисунок Г.9 – Список доступных сетевых адаптеров с "RNDIS/Ethernet Gadget"

10 Созданный виртуальный адаптер *Ethernet* появится в списке сетевых подключений и будет использоваться для доступа к ПЛК (рисунок Г.10).

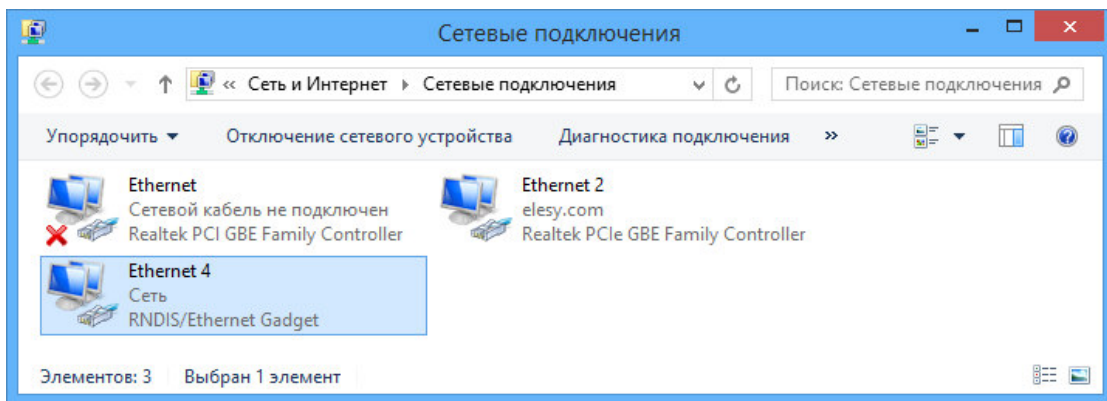


Рисунок Г.10 – Список сетевых подключений с добавленным адаптером RNDIS

11 При взаимодействии через виртуальные устройства *Ethernet* на базе драйвера *RNDIS*, оно автоматически получает от контроллера сетевые настройки (IP-адрес, маску подсети, адрес шлюза и т.д.). Адрес контроллера совпадает с адресом шлюза. Начальная конфигурация сети на основе драйвера *RNDIS* займёт не более 3 минут.

12 После выполнения указанных действий можно запустить конфигуратор *CoDeSys* для контроллера. Создать проект и выполнить поиск доступных устройств. Список доступных контроллеров, включая доступ через виртуальную сеть, показан на рисунке Г.11.

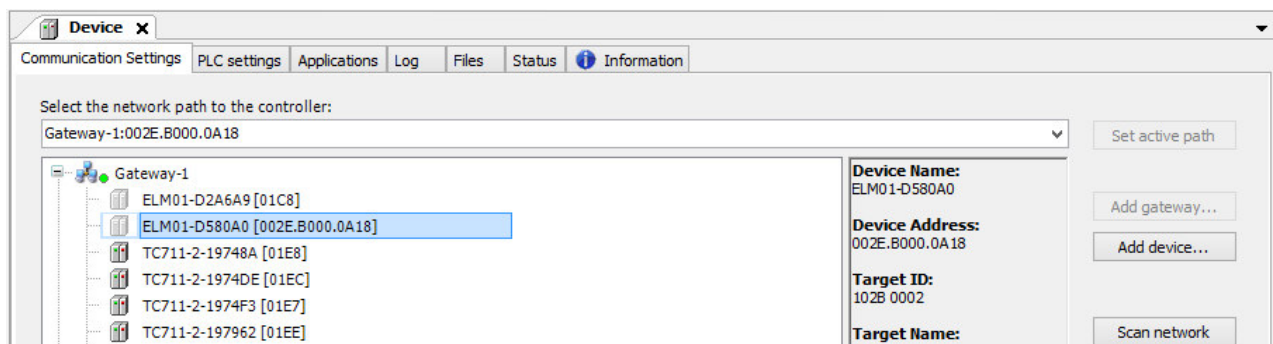


Рисунок Г.11 – Список контроллеров, включая подключения через адаптер RNDIS

Приложение Д (справочное)

Описание применения функциональных блоков MapIn, MapOut

Д.1 Использование ФБ MapIn

Функциональный блок **MapIn** позволяет фиксировать факт прохождения сигнала независимо от идентичности значений. Назначение ФБ **MapIn** – проверка обновления ответа на запись holding a/ов или coil a/ов (например, в **MBMRTU**).

Выход ФБ:

- *error*: BYTE; возвращает код ошибки ФБ. Возможные значения:
 - 0 – нет ошибок;
 - 1 – переменная пользователя не связана с выходом программного модуля;
 - 2 – системная ошибка;
 - 3 – системная ошибка;
 - 4 – системная ошибка;
 - 5 – системная ошибка;
 - 6 – системная ошибка;
 - 7 – системная ошибка;
 - 9 – не вызван экземпляр функционального блока.

Порядок работы с ФБ MapIn на примере языка ST:

Шаг 1. Инициализация экземпляра ФБ с передачей адреса выхода (например, out1 связана с выходом ModBusServer)

```
PROGRAM PLC_PRG
VAR
mapinanswcmd1 : Elesy.MapIn(ADR(inanswcmd1));
END_VAR
```

Шаг 2. Связать экземпляр ФБ с выходом

mapinanswcmd1(); (*При первом вхождении связывание экземпляра, при тысячи входов, выполняется за 300 микросекунд в худшем случае, при втором и более вхождении связывание не выполняется*)

Шаг 3. Проверить обновление ответа на запись holding a/ов или coil a/ов

mapinanswcmd1.IsUpdate(); (* Возвращает TRUE - обновление было, возвращает FALSE - обновления не было *)

Д.2 Использование ФБ MapOut

Функциональный блок **MapOut** предназначен для принудительной передачи выходных сигналов независимо от идентичности значений. Назначение ФБ **MapOut** – изменение поведения передачи данных выхода программного модуля (например, **MBSTCP** и т.п.).

Вход ФБ:

- *control*: BYTE; установка поведения передачи данных выхода программного модуля. Возможные значения:

0 – не передавать данные выхода (по умолчанию);

1 – передать данные выхода один раз (после передачи значение *control* автоматически сбрасывается в "0");

2 – передавать данные выхода по изменению.

Выход ФБ:

- *error*: BYTE; возвращает код ошибки ФБ. Возможные значения:

0 – нет ошибок;

1 – переменная пользователя не связана с выходом программного модуля;

2 – системная ошибка;

3 – системная ошибка;

4 – системная ошибка;

5 – системная ошибка;

6 – системная ошибка;

7 – системная ошибка;

8 – системная ошибка;

9 – не вызван экземпляр функционального блока.

Порядок работы с функциональным блоком MapOut на примере языка ST:

Шаг 1. Инициализация экземпляра ФБ с передачей адреса выхода (например, *out1* связана с выходом **ModBusServer)**

```
PROGRAM PLC_PRG
VAR
mapout1 : Elesy.MapOut(ADR(out1));
END_VAR
```

Шаг 2. Связать экземпляр ФБ с выходом

`mapout1();` (*При первом вхождении связывание экземпляра, при тысячи выходов, выполняется за 300 микросекунд в худшем случае, при втором и более вхождении связывание не выполняется *)

Шаг 3. Установить поведение для однократной передачи данных выхода

```
mapout1.control := 1;
```

Шаг 4. Применить поведение передачи данных выхода
mapout1.Send();

Приложение Е (справочное)

Пример применения функций ФБ CE30X

Для использования функций библиотеки **CE30XLibrary** необходимо в программе пользователя для ПЛК **PLC_PRG (PRG)** (в ветке *"Plc Logic"* – *"Application"* или *"Конфигурация задач"* – *"MainTask"*) в секции переменных обязательно **создать экземпляр функционального блока**, указав переменную с типом **ElsyMA_CE30X.CE30X** и добавить переменные для работы с этим ФБ.

```
PROGRAM PLC_PRG
VAR
FB_CE30X   : ElsyMA_CE30X.CE30X;
// или FB_CE30X   : ElsyMA_CE30X.CE30X_v2;
Timer_SendTR : TON;
set_ptTR    : TIME := T#5S;
RecvTR     : STRING;
cnt_TR     : UDINT := 1;
err_TR     : UDINT := 0;
CntErr_Init : UDINT := 0;
CntErr_SendTR : UDINT := 0;
Start_Init  : UDINT := 1;
Start_TR    : UDINT := 0;
.....
END_VAR
```

После этого, в программе можно использовать вызов ФБ.

Е.1 Инициализация ФБ

Для *"Инициализации ФБ"* в программе пользователя код может быть следующий:

```
IF Start_Init AND (FB_CE30X.CE30X_INIT = 0) AND
(FB_CE30X.CE30X_CONTROL = 0) THEN
  Start_Init := 0;
  FB_CE30X.BoudeRate :=5; // По умолчанию
  FB_CE30X.Paritet :=2; // По умолчанию
  FB_CE30X.BitNmb :=7; // По умолчанию
  FB_CE30X.StopBitNmb :=5; // По умолчанию
  CntErr_Init :=0; // Подготовка счетчиков ошибок
  FB_CE30X.CE30X_TA:= set_ptTR; // Тайм-аут
  FB_CE30X.CE30X_INIT := 1; // Активация инициализации с
установленными параметрами
  cnt_init := cnt_init + 1;

  ELSIF (Start_Init =0) AND (FB_CE30X.CE30X_INIT = 0) THEN//
Ожидание окончания инициализации
  IF FB_CE30X.INIT_ErrCode = 0 THEN
    .....// Инициализация завершена успешно
  ELSE
    CntErr_Init := CntErr_Init + 1;
    err_init := FB_CE30X.INIT_ErrCode; // Код ошибки
    Start_Init:=1;
  END_IF //IF CntErr_Init = 3 THEN
```

END_IF

Для ФБ CE30X_v2 код программы для инициализация будет иметь тот же вид.

Е.2 Пример выполнения одиночной транзакции

Если получение необходимых данных от измерительного устройства укладывается в одну транзакцию, то необходимо обеспечить вызов транзакции в режиме "Транзакция с установкой сессии с закрытием после завершения транзакции" (значение "2" в параметре *CE30X_MODE*).

Для ФБ CE30X:

```
IF Start_TR AND (FB_CE30X.CE30X_INIT = 0) AND
(FB_CE30X.CE30X_CONTROL = 0) THEN
  Start_TR := 0;
  FB_CE30X.CE30X_ID:= '1';
  FB_CE30X.CE30X_PASS:= '';

FB_CE30X.CE30X_TR:='$01$52$31$02$4D$4F$44$45$4C$28$29$03$4A$00'
;
  //- .R1.MODEL().J
  FB_CE30X.CE30X_TRLN:=
INT_TO_BYTE(LEN(FB_CE30X.CE30X_TR));
  FB_CE30X.CE30X_TA:= set_ptTR;
  FB_CE30X.CE30X_MODE:= 2;
  FB_CE30X.CE30X_CONTROL := 1;
END_IF
```

Для ФБ CE30X_v2 будет иметь тот же вид, за исключением присваивания значения сигналу FB_CE30X.CE30X_TR и вычислением длины отправляемой транзакции:

```
FB_CE30X.CE30X_TR.str:='$01$52$31$02$4D$4F$44$45$4C$28$29$03
$4A$00'; //- .R1.MODEL().J
  FB_CE30X.CE30X_TRLN:=
INT_TO_BYTE(LEN(FB_CE30X.CE30X_TR.str));
```

Е.3 Пример выполнения группы транзакций с открытием сессии

Для выполнения группы транзакций с открытием сессии в программе пользователя код может быть следующий:

Шаг 1. Выполнение транзакции с открытием сессии

```
IF Start_TR AND (FB_CE30X.CE30X_INIT = 0) AND
(FB_CE30X.CE30X_CONTROL = 0) THEN
  Start_TR := 0;
  FB_CE30X.CE30X_ID:= '1';
  FB_CE30X.CE30X_PASS:= '';

FB_CE30X.CE30X_TR:='$01$52$31$02$4D$4F$44$45$4C$28$29$03$4A$00'
;
  //- .R1.MODEL().J
```



```

    {либо
FB_CE30X.CE30X_TR.str:='$01$52$31$02$4D$4F$44$45$4C$28$29$03$4A$
00';
    //- .R1.MODEL().J}

    FB_CE30X.CE30X_TRLN:=
INT_TO_BYTE(LEN(FB_CE30X.CE30X_TR));
    {
    либо
    FB_CE30X.CE30X_TRLN:=
INT_TO_BYTE(LEN(FB_CE30X.CE30X_TR.str));
    }
    FB_CE30X.CE30X_TA:= set_ptTR;
    FB_CE30X.CE30X_MODE:= 1;
    FB_CE30X.CE30X_CONTROL := 1;
END_IF

```

В режиме "Транзакция с установкой сессии без закрытия" устанавливается сессия и выдается одна транзакция из переменной **CE30X_TR**. После этого можно выдать несколько транзакций в режиме "Одиночная транзакция без установки сессии" (значение "0" в параметре **CE30X_MODE**).

После выдачи транзакции необходимо дождаться, когда переменная **FB_CE30X.CE30X_CONTROL** станет равной "0", а значение переменной **CE30X_BUSY** ("Сигнал занятого ФБ") будет **FALSE**, что означает завершение выполнения транзакции.

Если переменная **CE30X_ERR** ("Код ошибки выполнения последней транзакции") равна "0", то результат выполнения запроса будет находиться в строковой переменной **CE30X_IN** ("Ответ на транзакцию"), а длина сообщения в переменной **CE30X_INLN** (в ответе могут содержаться непечатные символы).

Коды ошибок приведены в таблице 3.45 настоящего РЭ.

Шаг 2. Продолжение работы в сессии (выполнение одиночной транзакции без закрытия)

Достаточно установки значений для четырёх переменных:

CE30X_TR, CE30X_MODE, CONTROL, CE30X_TRLN

```

    FB_CE30X.CE30X_TR:=
'$01$52$31$02$53$54$41$54$5F$28$29$03$74$00';
    // Запрос состояния .R1.STAT_().t
    { либо FB_CE30X.CE30X_TR:=
'$01$52$31$02$53$54$41$54$5F$28$29$03$74$00'; // Запрос состояния
.R1.STAT_().t
    }
    FB_CE30X.CE30X_TRLN:= INT_TO_BYTE(LEN(FB_CE30X.CE30X_TR));
    {либо
    FB_CE30X.CE30X_TRLN:=
INT_TO_BYTE(LEN(FB_CE30X.CE30X_TR.str));
    }
    FB_CE30X.CE30X_MODE:= 0;
    FB_CE30X.CE30X_CONTROL := 1;

```

Так же как и на шаге 1, дождаться завершения выполнения транзакции.

Ответ будет:

Руководство по эксплуатации

```
.STAT_(0,0).... (02 53 54 41 54 5F 28 30 2C 30 29 0D 0A 03 12)
```

По завершению можно повторить запрос к счетчику с открытой сессией, присвоив переменной **FB_CE30X.CE30X_TR** новое значение, например, "*Запрос состояния батареи*":

```
FB_CE30X.CE30X_TR:=
'$01$52$31$02$56$5F$42$41$54$28$29$03$65$00'
//.R1.V_BAT().e
```

На этот запрос в переменной **CE30X_IN** будет находиться ответ вида:

```
.V_BAT(3.7)....
(02 56 5F 42 41 54 28 33 2E 37 29 0D 0A 03 0F)
```

Шаг 3. Закрытие сессии

```
FB_CE30X.CE30X_TR:= '$01$42$30$03$75$00'; // Завершение сессии
// «.B0.u»
FB_CE30X.CE30X_MODE:= 0;
FB_CE30X.CE30X_CONTROL := 1;
```

Так же как и на шаге 1, дождаться завершения выполнения транзакции.

После выполнения шага 3 можно открывать сессию и выполнять транзакции для другого измерительного оборудования в линии.

Е.4. Пример вызова экземпляра ФБ

Для программы на языке **ST**, при объявлении экземпляра ФБ, как указано выше, строка кода вызова должна быть следующей:

```
FB_CE30X();
```

Приложение Ж (справочное)

Пример применения функций ФБ M23X и ФБ M23X_v2

Для использования функций библиотеки **M23XLibrary** необходимо в программе пользователя для ПЛК **PLC_PRG (PRG)** (в ветке *"Plc Logic" – "Application"* или *"Конфигурация задач" – "MainTask"*) в секции переменных обязательно **создать экземпляр функционального блока**, указав переменную с типом *ElsyMA_M23X.M23X* и добавить переменные для работы с этим ФБ.

```
PROGRAM PLC_PRG
VAR
FB_M23X      : ElsyMA_M23X.M23X;
{
либо
FB_M23X      : ElsyMA_M23X.M23X_v2;
}
Timer_SendTR : TON;
set_ptTR     : TIME := T#5S;
RecvTR       : STRING (255);
cnt_TR       : UDINT := 1;
err_TR       : UDINT := 0;
CntErr_Init  : UDINT := 0;
CntErr_SendTR : UDINT := 0;
Start_Init   : UDINT := 1;
Start_TR     : UDINT := 0;
.....
END_VAR
```

После этого, в программе можно использовать вызов ФБ.

Ж.1 Инициализация ФБ

Для *"Инициализации ФБ"* в программе пользователя код может быть следующий:

```
IF Start_Init AND (FB_M23X.M23X_INIT = 0) AND
(FB_M23X.M23X_CONTROL = 0) THEN
  Start_Init := 0;
  FB_M23X.BoudeRate :=2; // Скорость 9600 (по умолчанию)
  FB_M23X.Paritet :=0; // Контроль чётности отсутствует (по
умолчанию)
  FB_M23X.BitNmb :=8; // Количество бит данных (по умолчанию)
  FB_M23X.StopBitNmb :=1; // Количество стоп-бит (по умолчанию)
  CntErr_Init :=0; // Подготовка счетчиков ошибок инициализации
  FB_M23X.M23X_TA:= set_ptTR; // Тайм-аут
  FB_M23X.M23X_INIT := 1; // Активация ФБ для инициализации
интерфейса RS- // 485 с установленными параметрами
  cnt_init := cnt_init + 1; // Счётчик попыток инициализации
увеличивается

ELSIF (Start_Init =0) AND (FB_M23X.M23X_INIT = 0) THEN//
Ожидание окончания инициализации
  IF FB_M23X.INIT_ErrCode = 0 THEN
```

```

.....// Инициализация завершена успешно
ELSE
  CntErr_Init := CntErr_Init + 1;
  err_init := FB_M23X.INIT_ErrCode; // Код ошибки
  Start_Init:=1;
  END_IF //IF CntErr_Init = 3 THEN
END_IF

```

Для ФБ **M23X_v2** код программы для инициализации будет иметь такой же вид

Ж.2 Пример выполнения одиночной транзакции

Если получение необходимых данных от измерительного устройства укладывается в одну транзакцию, то необходимо обеспечить вызов транзакции в режиме "*Транзакция с установкой сессии с закрытием после завершения транзакции*" (значение "2" в параметре **M23X_MODE**).

```

IF Start_TR AND (FB_M23X.M23X_INIT = 0) AND
(FB_M23X.M23X_CONTROL = 0) THEN
  Start_TR := 0;
  FB_M23X.M23X_ID:= '1';
  FB_M23X.M23X_PASS:= '';
  // Опрос накопленной энергии от начала сброса
  FB_M23X.M23X_TR:='$00$05$00$00$10$25$00';
  FB_M23X.M23X_TRLN:= INT_TO_BYTE(LEN(FB_M23X.M23X_TR));
  FB_M23X.M23X_TA:= set_ptTR;
  FB_M23X.M23X_MODE:= 2;
  FB_M23X.M23X_CONTROL := 1;
END_IF

```

Для ФБ **M23X_v2** будет иметь тот же вид, за исключением присваивания значения сигналу **FB_M23X.M23X_TR** и вычислением длины отправляемой транзакции:

```

FB_M23X.M23X_TR.str:='$00$05$00$00$10$25$00';
FB_M23X.M23X_TRLN:= INT_TO_BYTE(LEN(FB_M23X.M23X_TR.str));

```

Ж.3 Пример выполнения группы транзакций с открытием сессии

Для выполнения группы транзакций с открытием сессии в программе пользователя код может быть следующий:

Шаг 1. Выполнение транзакции с открытием сессии

```

IF Start_TR AND (FB_M23X.M23X_INIT = 0) AND
(FB_M23X.M23X_CONTROL = 0) THEN
  Start_TR := 0;
  FB_M23X.M23X_ID:= '1';
  FB_M23X.M23X_PASS:= '';
  // Опрос накопленной энергии за предыдущие сутки
  FB_M23X.M23X_TR:= '$00$05$50$00$2C$25$00';
  {
  либо
  FB_M23X.M23X_TR.str:= '$00$05$50$00$2C$25$00';
  }
  FB_M23X.M23X_TRLN:= INT_TO_BYTE(LEN(FB_M23X.M23X_TR));
  {

```

```

либо
  FB_M23X.M23X_TRLN:= INT_TO_BYTE(LEN(FB_M23X.M23X_TR.str));
}
  FB_M23X.M23X_TA:= set_ptTR;
  FB_M23X.M23X_MODE:= 1;
  FB_M23X.M23X_CONTROL := 1;
END_IF

```

В режиме "Транзакция с установкой сессии без закрытия" устанавливается сессия и выдается одна транзакция из переменной **M23X_TR**. После этого можно выдать несколько транзакций в режиме "Одиночная транзакция без установки сессии" (значение "0" в параметре **M23X_MODE**).

После выдачи транзакции необходимо дождаться, когда переменная **FB_M23X.M23X_CONTROL** станет равной "0", а значение переменной **M23X_BUSY** ("Сигнал занятого ФБ") будет **FALSE**, что означает завершение выполнения транзакции.

Если переменная **M23X_ERR** ("Код ошибки выполнения последней транзакции") равна "0", то результат выполнения запроса будет находиться в строковой переменной **M23X_IN** ("Ответ на транзакцию"), а длина сообщения в переменной **M23X_INLN** (в ответе могут содержаться непечатные символы).

Коды ошибок приведены в таблице 3.45 настоящего РЭ.

Шаг 2. Продолжение работы в сессии (выполнение одиночной транзакции без закрытия)

Достаточно установки значений для четырёх переменных:

```

M23X_TR, M23X_MODE, CONTROL, M23X_TRLN
// Опрос накопленной энергии за текущие сутки
FB_M23X.M23X_TR:= '$00$05$40$00$21$E5$00';
{
либо
  FB_M23X.M23X_TR.str:= '$00$05$40$00$21$E5$00';
}
FB_M23X.M23X_TRLN:= INT_TO_BYTE(LEN(FB_M23X.M23X_TR));
{
либо
  FB_M23X.M23X_TRLN:= INT_TO_BYTE(LEN(FB_M23X.M23X_TR.str));
}
FB_M23X.M23X_MODE:= 0;
FB_M23X.M23X_CONTROL := 1;

```

Так же как и на шаге 1, дождаться завершения выполнения транзакции.

Ответ будет:

```
$00$08$16$21$4F$9E$00$00[$25$04]$00[$4E$09]$00[$B8$1F]$AF$BA
```

По завершению можно повторить запрос к счетчику с открытой сессией, присвоив переменной **FB_M23X.M23X_TR** новое значение, например, "Сила тока по фазам":

```
FB_M23X.M23X_TR:= '$00$08$16$21$4F$9E$00' // Сила тока по фазам
```

На этот запрос в переменной **M23X_IN** будет находиться ответ вида:

\$00\$08\$16\$21\$4F\$9E\$00\$00[\$25\$04]\$00[\$4E\$09]\$00[\$B8\$1F]\$AF\$BA

Шаг 3. Закрытие сессии

```
FB_M23X.M23X_TR:= '$01$42$30$03$75$00'; // Завершение сессии
// «.B0.u»
FB_M23X.M23X_MODE:= 0;
FB_M23X.M23X_CONTROL := 1;
```

Так же как и на шаге 1, дождаться завершения выполнения транзакции.

После выполнения шага 3 можно открывать сессию и выполнять транзакции для другого измерительного оборудования в линии.

Ж.4 Пример вызова экземпляра ФБ

Для программы на языке **ST**, при объявлении экземпляра ФБ, как указано выше, строка кода вызова должна быть следующая:

```
FB_M23X();
```

Приложение И (справочное)

Пример применения функций ФБ GSM

Для работы контроллера с применением функций ФБ GSM необходимы следующие объявления переменных:

```
PROGRAM PLC_PRG
VAR
mygsm : ElsyMA_GSM.ELSYGSM;
cntinit: UDINT;
cntsms : UDINT;
cntinsms : UDINT;
cntat : UDINT;
recvstr : STRING(255);
sms1 : STRING(255);
sms1num : STRING(255);
temp : UDINT;
cntsend : BYTE;
cntread : BYTE;
cntreset: BYTE;
cntnoterr : BYTE;
myatrecvstr : STRING(255);
isbadreset : BOOL;
gsm_state: BYTE;
start_readsms: BOOL;
END_VAR
```

После этого, в программе можно использовать функции GSM.

Для "Инициализации модуля GSM" в программе пользователя код может быть следующий:

```
(* Инициализация GSM *)
IF mygsm.error = 0 AND mygsm.isbusy = 0 AND cntinit = 0 THEN
mygsm.controlinit := 1;
cntinit := cntinit + 1;
END_IF
(* 2 попытки переинициализации GSM - в случае недоступности GSM *)
IF mygsm.error = -3 AND mygsm.isbusy = 0 AND cntinit < 3 THEN
mygsm.controlinit:= 1;
cntinit := cntinit + 1;
END_IF
```

В программе пользователя для передачи AT-команды запроса баланса код может быть следующий:

```
(* Передача AT команды *)
IF mygsm.error = 0 AND mygsm.isbusy = 0 AND mygsm.controlreset =
0 AND cntat = 0
THEN (* Подготовка at команды *)
mygsm.cmdat := 'AT+CUSD=1,"#100#" $R$R$n';
mygsm.ptat := T#5S;
(* Разрешаем передать at команду 1 раз *)
mygsm.controlat := 1;
cntat := cntat + 1;
END_IF
```

Руководство по эксплуатации

```
(* Попытка сброса GSM - в случае неудачных попыток передачи AT команды *)
```

```
IF mygsm.error = -4 AND mygsm.isbusy = 0 AND cntat = 3 AND cntreset = 0 THEN
```

```
mygsm.controlreset:= 1;
```

```
cntat := 0;
```

```
cntreset := 1;
```

```
END_IF
```

```
(* Приём ответа на первую AT команду *)
```

```
IF mygsm.error = 0 AND cntat = 1 AND mygsm.isbusy = 0 THEN
```

```
myatrecvstr := mygsm.recvstr;
```

```
cntat := 3;
```

```
END_IF
```

В программе пользователя код для применения функции "*Отправка текстового сообщения (SMS)*" может быть следующий:

```
(* Передача SMS *)
```

```
IF mygsm.error = 0 AND mygsm.isbusy = 0 AND cntsms = 0 THEN (* Инициализация данных для передачи SMS *)
```

```
mygsm.numbersms := '8961XXXXXXX';
```

```
mygsm.textsms := 'test message';
```

```
(* Разрешение передачи SMS 1 раз *)
```

```
mygsm.controlsendsms := 1;
```

```
cntsms := cntsms + 1;
```

```
END_IF
```

```
IF mygsm.error = -5 AND mygsm.isbusy = 0 AND cntsms < 3 THEN (* Возникли ошибки по тайм-ауту GSM - 2 попытки передачи SMS *)
```

```
mygsm.numbersms := '8961XXXXXXX';
```

```
mygsm.textsms := 'test message';
```

```
(* Разрешение передачи SMS 1 раз *)
```

```
mygsm.controlsendsms := 1;
```

```
cntsms := cntsms + 1;
```

```
END_IF
```

В программе пользователя код для применения функции "*Приём текстового сообщения (SMS)*" может быть следующий:

```
(* Запуск проверки принятых сообщений, если ФБ GSM свободен от выполнения других функций *)
```

```
IF mygsm.error = 0 AND mygsm.isbusy=0 AND start_readsms THEN
```

```
mygsm.controlreadsms := 1;
```

```
cntread := 0;
```

```
END_IF
```

```
(* Проверка готового SMS в буфере приёма*)
```

```
IF mygsm.error = 0 AND mygsm.controlreadsms = 0 AND mygsm.isbusy=0 and mygsm.isreadysms THEN
```

```
IF mygsm.isbadreadsms = FALSE THEN
```

```
sms1num := mygsm.recvnumber;
```

```
sms1:= mygsm.recvsms;
```

```
END_IF
```

```
END_IF
```

```
IF mygsm.error = -6 AND mygsm.controlreadsms = 0 AND cntread < 2 and mygsm.isbusy=0 THEN
```

```
(* Возникли ошибки по тайм-ауту GSM, выполнить 2 попытки запроса входящего SMS *)
```



```
mygsm.controlreadsms = 1;  
cntread := cntread + 1;  
END_IF
```

Для выполнения функции "*Аппаратный рестарт модуля GSM*", например, в случае возникновения ошибки с кодом "=-3" при инициализации модуля, текст программы может быть следующий:

```
(* Попытка сброса GSM - в случае неудачных попыток  
инициализации *)  
IF mygsm.error = -3 AND mygsm.isbusy = 0 AND cntinit = 3 AND  
cntreset = 0 THEN  
mygsm.controlreset:= 1; // Активация аппаратного рестарта модуля  
GSM  
cntinit := 0;  
cntreset := 1;  
END_IF
```

Приложение К (справочное)

Пример программного кода для работы со сканерами

Данная программа предназначена для работы со сканерами *Metrologic* и *Honeywell Imaging & Mobility*. После загрузки программы на контроллер с помощью среды программирования *CoDeSys*, контроллер определяет количество подключенных сканеров и их модель. Подключив сканер к контроллеру и просканировав QR или штрих-код, пользователь получает строку значения кода, которая отображается в *CoDeSys* в переменной **myScan1Str** для первого сканера (в данном примере *Metrologic*) и в переменной **myScan2Str** для второго сканера (в данном примере *Honeywell Imaging & Mobility*).

(* 09 nov 17 Nesterenko P. EleTeam

Пример использования сканеров штрих кода в контроллере ЭЛСИМА.

- По старту инициализируем подключенные сканеры типа Metrologic и Honeywell.
- В процессе работы получает от сканеров данные и считает количество полученных строк.
- В процессе работы следит за состоянием подключенных устройств, при необходимости переинициализирует устройства.

*)

PROGRAM PLC_PRG

VAR

(* ФБ для получения доступных в системе устройств *)

myUSBGet : USBScanLibrary.GetUSBDevices;

usbgetflag : BOOL := TRUE; (* Флаг запуска инициализации устройств *)

flagready : BOOL; (* Флаг, что инициализация завершилась *)

(* ФБ для работы со сканером Metrologic *)

myScanner1 : USBScanLibrary.USBScan;

myScan1Str : STRING(255); (* полученная от сканера строка для дальнейшей обработки *)

mystr1len : INT; (* длина принятой строки для контроля *)

scan1Cnt : INT; (* счетчик полученных штрих кодов *)

scan1check : INT; (* код завершения проверки устройства *)

(* ФБ для работы со сканером Honeywell Imaging & Mobility *)

myScanner2 : USBScanLibrary.USBScan;

myScan2Str : STRING(255); (* полученная от сканера строка для дальнейшей обработки *)

mystr2len : INT; (* длина принятой строки для контроля *)

scan2Cnt : INT; (* счетчик полученных штрих кодов *)

scan2check : INT; (* код завершения проверки устройства *)

(* Для контроля работы подключенных устройств *)

myCheckTimer : TON; (* Таймер 30 сек на проверку устройств *)

myInt : INT;

i : INT;

END_VAR

(* Считываем доступные в системе устройства по старту контроллера,
для получения пути, по которому будем работать со сканером *)

IF usbgetflag = TRUE THEN

myUSBGet();

usbgetflag := FALSE;

flagready := TRUE;

END_IF

(* Так как мы хотим работать только со сканерами Metrologic и Honeywell Imaging & Mobility ищем их в полученном списке найденных устройств. Если такие находятся, работаем с ними. *)

```

IF flagready = TRUE THEN
  FOR i := 1 TO myUSBGet.devcnt DO
    (* Проверка сканера Metrologic и его инициализация *)
    IF myUSBGet.strucmas[i].manufacturer = 'Metrologic' THEN
      myScanner1.portnmb := myUSBGet.strucmas[i].potnmb;
      myScanner1();
    END_IF
    (* Проверка сканера Honeywell Imaging & Mobility и его инициализация *)
    IF myUSBGet.strucmas[i].manufacturer = 'Honeywell Imaging & Mobility ' THEN
      myScanner2.portnmb := myUSBGet.strucmas[i].potnmb;
      myScanner2();
    END_IF
  END_FOR;
  flagready := FALSE;
END_IF;

(* Если myScanner1 успешно открыл порт, получаем от него строку штрих кода *)
IF myScanner1.error = 1 THEN
  (* Вызываем метод для получения данных от сканера. Данных может не быть *)
  mystr1len := myScanner1.Receive();
  (* Если mystr1len равно 0, значит данных нет. Строка должна быть от 5 до 20 символов. *)
  IF mystr1len > 5 AND mystr1len < 20 THEN
    myScan1Str := myScanner1.strscan;
    scan1Cnt := scan1Cnt + 1; (* счетчик полученных штрих кодов *)
  END_IF
END_IF

(* Если myScanner2 успешно открыл порт, получаем от него строку штрих кода *)
IF myScanner2.error = 1 THEN
  (* Вызываем метод для получения данных от сканера. Данных может не быть *)
  mystr2len := myScanner2.Receive();
  (* Если mystr2len равно 0, значит данных нет. Строка должна быть от 6 до 30 символов. *)
  IF mystr2len > 6 AND mystr2len < 30 THEN
    myScan2Str := myScanner2.strscan;
    scan2Cnt := scan2Cnt + 1; (* счетчик полученных штрих кодов *)
  END_IF
END_IF

(* Проверка доступности устройств каждые 300 сек. И запуск переинициализации при необходимости *)
myCheckTimer( IN := TRUE, PT := T#30S );
IF myCheckTimer.Q = TRUE THEN
  myCheckTimer( IN := FALSE );
  (* Проверка устройства 1 *)
  scan1check := myScanner1.CheckState();
  (* Проверка устройства 2 *)
  scan2check := myScanner2.CheckState();
  (* запуск переинициализации если 1-е или 2-е устройство не работает *)
  IF (scan1check <> 0) OR (scan2check <> 0) THEN
    usbgetflag := TRUE; (* Флаг запуска инициализации устройств *)
    flagready := FALSE; (* Флаг, что инициализация завершилась *)
  END_IF;
END_IF;

myInt := myInt + 1;

```

Приложение Л (справочное)

Обновление системного ПО контроллера

Обновление ПО производится с **USBFlash** носителя и условно разбивается на 4 этапа:

1. Подготовка **USBFlash** носителя для обновления системного ПО.
2. Подготовка контроллера для обновления ПО.
3. Непосредственно процесс автоматического обновления ПО.
4. Проверка корректности обновления ПО.

ВНИМАНИЕ! Автоматическое обновление системного ПО доступно только начиная с версии 3.1.0.9311.

Л.1 Подготовка **USBFlash** носителя

Для обновления ПО должен использоваться **USBFlash** носитель объемом не менее 256Мб, файловая система FAT32. Для подготовки носителя необходимо выполнить следующие действия:

1. В корневом каталоге **USBFlash** носителя создать каталог **elsymaupdate/sysup**, если данный каталог уже существует **необходимо удалить всё содержимое в нем**;
2. Скопировать в каталог **elsymaupdate/sysup** файл обновления (для примера, имя файла обновления версии 3.1.1 имеет вид elm-m01r2-base-v3.3.0.11125.zip).

Примечание: подготовить **USBFlash** можно один раз в ходе обновления ПО для партии изделий.

ВНИМАНИЕ! После обновления программного обеспечения на **USBFlash** будут созданы файл update.log и следующие каталоги:

elsymaupdate/backup/elm01-<HWID> - резервная копия рабочего каталога ПЛК,
elsymaupdate/sysup/elm01-<HWID> - содержит журнал обновления.

где HWID имеет вид - ххууzz что соответствует 3-м последним октетам MAC-адреса ПЛК. Наличие этих каталогов на **USBFlash** предотвращает вторичное обновление ПО ПЛК. При необходимости еще раз обновить ПО, следует выполнить пункт Л.1.1!

Л.2 Подготовка контроллера

1. Выключить питание контроллера.
2. Установить переключатели SW «1» и «3» в положение «ON». Переключатели SW «2» и «4» установить в положение «OFF».
3. Установить в разъем **USB1** контроллера **SBFlash**.

Л.3 Автоматическое обновление ПО

1. Подать питание на контроллер.
2. Процесс обновления системного ПО запускается автоматически при выполнении условий описанных выше примерно через 40 сек. после подачи питания. Индикация процесса обновления ПО соответствует режиму «Автоматическое обновление ПО контроллера» (индикация режима представлена в таблице 1.6).

3. Процесс автоматического обновления ПО завершается при переходе в режим «Завершение процесса обновления ПО контроллера» примерно через 4-5 минут (индикация режима представлена в таблице 1.6).

ВНИМАНИЕ! При обновлении системного программного обеспечения будут восстановлены предыдущие сетевые настройки (IP адрес и маска подсети).

ВНИМАНИЕ! При обновлении системного программного обеспечения удаляется существующий проект в контроллере. После обновления необходимо загрузить в контроллер соответствующий проект штатным способом.

Л.4 Проверка корректности обновления ПО

1. Для продолжения работы устройства в штатном режиме достаточно перевести переключатели SW “2” в положение “ON” и SW “3” в положение “OFF” непосредственно после обновления или на выключенном устройстве.
2. После окончания инициализации подключиться к контроллеру системой CoDeSys и загрузить в контроллер любой проект. Проверить значение параметра FWVer коннектора CPU_INFO. Значение должно соответствовать версии установочного файла (для приведенного выше примера значение сигнала FWVer будет иметь значение «3.3.0.11125»).

Приложение М (справочное)

Обновление пользовательского ПО контроллера

Пользователь имеет возможность обновить следующие компоненты, относящиеся к пользовательскому программному обеспечению:

1. Исполняемый в контроллере проект (возможность обновить проект без системы программирования CoDeSys);
2. Заданный сервисный IP адрес (таким образом можно задать необходимые IP адреса контроллерам если нет возможности подключиться к контроллеру по заданному ранее IP адресу);
3. Сетевое имя контроллера (таким образом можно задать необходимые имена контроллерам для идентификации в сети без использования системы программирования CoDeSys).

ВНИМАНИЕ! Обновление пользовательского ПО доступно только начиная с версии 3.1.0.9311.

Обновление производится через заранее подготовленный **USBFlash** диск и условно разбивается на 3 этапа:

1. Подготовка USBFlash носителя для обновления ПО.
2. Подготовка контроллера для обновления ПО.
3. Непосредственно процесс обновления ПО.

М.1 Подготовка USBFlash носителя

Для обновления ПО должен использоваться **USBFlash** носитель объемом не менее 256Мб, файловая система FAT32. Для подготовки носителя необходимо выполнить следующие действия:

1. В корневом каталоге **USBFlash** носителя создать каталог **elsymaupdate/userup**, если данный каталог уже существует **необходимо удалить всё содержимое в нем**;
2. Скопировать в каталог **elsymaupdate/userup** файлы обновления *app, *crc, *cfg и необходимые подкаталоги (формат файлов *cfg приведен ниже).

ВНИМАНИЕ! После обновления программного обеспечения на **USBFlash** будут созданы файл update.log и каталог elsymaupdate/userup/elm01-<HWID> – признак обновления, где HWID имеет вид - ххууzz что соответствует 3-м последним октетам MAC-адреса ПЛК. Наличие каталога на **USBFlash** предотвращает вторичное обновление ПО ПЛК. При необходимости еще раз обновить ПО необходимо выполнить пункт М.1.1!

М.2 Подготовка контроллера

1. Выключить питание контроллера.
2. Установить переключатели SW «1» и «3» в положение «ON». Переключатели SW «2» и «4» установить в положение «OFF».
3. Установить в разъем **USB1** контроллера **SBFlash**.

М.3 Автоматическое обновление ПО

1. Подать питание на контроллер.

2. Процесс обновления пользовательского ПО запускается автоматически при выполнении условий описанных выше примерно через 40 сек. после подачи питания. Индикация процесса обновления ПО соответствует режиму «Автоматическое обновление ПО контроллера» (индикация режима представлена в таблице 1.6).
3. Процесс автоматического обновления ПО завершается при переходе в режим «Завершение процесса обновления ПО контроллера» примерно через 10-20 секунд (индикация режима представлена в таблице 1.6).

Для изменения сервисного IP адреса необходимо создать файл ExtConfig1.cfg и задать в нем следующие строки:

```
[CmpBlkDrvUdp]
itf.0.mode=static
itf.1.ipaddress=10.14.1.53
itf.0.name=main
itf.0.networkmask=255.255.0.0
itf.0.servlan=1
itf.0.gteway=10.14.0.1
```

Для изменения имени контроллера необходимо создать файл ExtConfig3.cfg и задать в нем следующие строки:

```
[SysTarget]
NodeName=ELSYMA-M01-pahanTst
```

Приложение Н (справочное)

Перечень сообщений о состояниях программных и аппаратных модулей

Аппаратные модули (модули УВВ)

1. Верификация модуля (на примере `elsyma_a01`, IP=10.9.0.1)

Проверка версий в контроллере осуществляется в 2 этапа: при загрузке конфигурации и при работе с модулем. Порядок контроля версий изложен в соответствующей главе руководства.

При загрузке конфигурации конфигуратор проверяет версию модуля УВВ шаблона с версией, установленной в конфигурации, и предлагает актуализировать до версии шаблона. Пользователь может либо отказаться, либо согласиться.

В режиме online при работе с модулем система проверяет версию конфигурации с версией, получаемой с модуля УВВ.

Если имеются нарушения в формате версий, могут выдаваться следующие сообщения:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_parser_realvers: error version data"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_parser_realvers: not support format=3 version"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_parser_realvers: error format version"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_parser_realvers: bad version data"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_parser_cfgvers: bad version data"
```

Модуль не сможет продолжить работу.

Примечание - Если не удалось распознать версию, то нераспознанные цифры заменяются на "-1".

Если версия отличается в 4 октете, то в журнал заносится предупреждение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgvers=0.1.2.3, realvers=0.1.2.4"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal sw_build version"
```

Модуль сможет продолжить работу.

Если версия отличается в 3 октете, в журнал заносится предупреждение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgvers=0.1.2.3, realvers=0.1.3.3"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal sw_patch version"
```

Модуль сможет продолжить работу.

Если версия отличается в 2 октете, то в журнал заносится ошибка:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgvers=0.1.2.3, realvers=0.2.2.3"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal sw_minor version"
```

Модуль не сможет продолжить работу.

Если версия отличается в 1 октете, то в журнал заносится ошибка:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgvers=0.1.2.3, realvers=1.1.2.3"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal sw_major version"
```

Модуль не сможет продолжить работу

Если имеются критические ошибки в конфигурации, могут выдаваться следующие сообщения:


```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys::          ceb_checkmod:
EBUS_Device_Module_ModNameID not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys::          ceb_checkmod:
EBUS_Device_Module_SoftNameID not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: EBUS_Device_Module_SoftVerID
not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys::          ceb_checkmod:
EBUS_Device_Module_TemplDateID not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys::          ceb_checkmod:
EBUS_Device_Module_RealNameID not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: EBUS_Device_Module_RealSoftID
not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: EBUS_Device_Module_RealVerID
not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys::          ceb_checkmod:
EBUS_Device_Module_RealDateID not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: EBUS_Device_Module_IdHardID
not found"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: EBUS_Device_Module_IdSoftID
not found"
```

Модуль не сможет продолжить работу.

Если имеются не критические ошибки в конфигурации, могут выдаваться следующие сообщения:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: EBUS_Device_Module_CntResID
not found"
```

Модуль сможет продолжить работу, при этом счетчик сброса не будет работать.

Если отличается аппаратный идентификатор, в журнал заносится ошибка:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgidhard=0, realidhard=1"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal idhard"
```

Модуль не сможет продолжить работу.

Если отличается программный идентификатор, в журнал заносится ошибка:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgidsoft=0, realidsoft=1"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal idsoft"
```

Модуль не сможет продолжить работу.

Если различается имя, модуля в журнал заносится ошибка:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgname=elsyma_a01,
realname=elsyma_da01"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal module name"
```

Модуль не сможет продолжить работу.

Если различается имя программного обеспечения, в журнал заносится ошибка.

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgsoft=ai8, realsoft= di20a6o8"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal soft name"
```

Модуль не сможет продолжить работу.

Если различается дата создания модуля, в журнал заносится предупреждение

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: cfgdate=10.10.2018,
realdate=11.10.2018"
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: not equal date"
```

Модуль сможет продолжить работу.

ИТОГО

Если модуль не сможет продолжить работу, после верификации выведется сообщение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: check module done error"
```

В дереве устройств в CoDeSys пиктограмма модуля УВВ окрасится в красный цвет.

Если не обнаружены критические ошибки, то выведется сообщение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_checkmod: check module done ok"
```

В дереве устройств в CoDeSys пиктограмма модуля УВВ окрасится в зеленый цвет (если контроллер в режиме старт), или примет серый цвет (если контроллер в режиме стоп).

2. Отсутствие связи с модулем по старту контроллера

При отсутствии связи с модулем по старту контроллера выведется предупреждение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_cputask: module no active"
```

Модуль не сможет продолжить работу (модуля нет).

3. Разрыв соединения с модулем при работе контроллера

При разрыве связи с модулем при работе контроллера выводится следующее предупреждение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_cputask: module disconnect"
```

Модуль не сможет продолжить работу (модуля нет).

4. Восстановление соединения с модулем при работе контроллера

При восстановлении связи с модулем при работе контроллера выводится следующее сообщение

```
"elsyma_m01_gsm[10.9.32.1]_msys..elsyma_a01[10.9.0.1]_msys:: ceb_cputask: module connect"
```

Модуль продолжает работать.

Программные модули

1. Верификация программного модуля (на примере программного модуля *tsync*)

Проверка версий в контроллере осуществляется в 2 этапа: при загрузке конфигурации и при работе с модулем. Порядок контроля версий изложен в соответствующей главе руководства.

При загрузке конфигурации конфигуратор проверяет версию программного модуля шаблона с версией, установленной в конфигурации, и предлагает актуализировать до версии шаблона. Пользователь может либо отказаться, либо согласиться.

В режиме online при работе с модулем система проверяет версию конфигурации с версией, получаемой с программного модуля.

Если имеются нарушения в формате версий, могут выдаваться следующие сообщения:

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_parser_realvers: error version data"
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_parser_realvers: not support format=3 version"
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_parser_realvers: error format version"
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_parser_realvers: bad version data"
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_parser_cfgvers: bad version data"
```

Модуль не сможет продолжить работу.

Если версия отличается в 4 октете, то в журнал заносится предупреждение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: cfgvers=0.1.2.3, realvers=0.1.2.4"
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: not equal sw_build version"
```

Модуль сможет продолжить работу/

Если версия отличается в 3 октете в журнал заносится предупреждение

```
"elsyma_m01_gsm(10.9.32.1)_msys..tsync[1]:: ceb_checkch: cfgvers=0.1.2.3, realvers=0.1.3.3"
```

```
"elsyma_m01_gsm(10.9.32.1)_msys..tsync[1]:: ceb_checkch: not equal sw_patch version"
```

Модуль сможет продолжить работу.

Если версия отличается в 2 октете, в журнал заносится ошибка:

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: cfgvers=0.1.2.3, realvers=0.2.2.3"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: not equal sw_minor version"
```

Модуль не сможет продолжить работу/

Если версия отличается в 1 октете, в журнал заносится ошибка

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: cfgvers=0.1.2.3, realvers=1.1.2.3"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: not equal sw_major version"
```

Модуль не сможет продолжить работу

Если имеются критические ошибки в конфигурации, могут выдаваться следующие сообщения:

```
"elsyma_m01_gsm(10.9.32.1)_msys:: ceb_checkch: EBUS_Device_Module_ChRealNameID not found"
```

```
"elsyma_m01_gsm(10.9.32.1)_msys:: ceb_checkch: EBUS_Device_Module_ChRealSoftID not found"
```

```
"elsyma_m01_gsm(10.9.32.1)_msys:: ceb_checkch: EBUS_Device_Module_ChRealDateID not found"
```

```
"elsyma_m01_gsm(10.9.32.1)_msys:: ceb_checkch: EBUS_Device_Module_ChNameID not found"
```

```
"elsyma_m01_gsm(10.9.32.1)_msys:: ceb_checkch: EBUS_Device_Module_ChVersionID not found"
```

```
"elsyma_m01_gsm(10.9.32.1)_msys:: ceb_checkch: EBUS_Device_Module_ChDateID not found"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: EBUS_Device_Channel_RealNameID not found"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: EBUS_Device_Channel_RealSoftID not found"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: EBUS_Device_Channel_RealDateID not found"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: EBUS_Device_Channel_ChNameID not found"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: EBUS_Device_Channel_ChVersionID not found"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: EBUS_Device_Channel_ChDateID not found"
```

Если различается имя модуля, в журнал заносится ошибка:

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: cfgname=tsync, realname=tsynk"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: not equal module name"
```

Модуль не сможет продолжить работу.

Если различается дата создания модуля, в журнал заносится предупреждение

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: cfgdate=10.10.2018, realdate=11.10.2018"
```

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: not equal date"
```

Модуль сможет продолжить работу.

ИТОГО

Если модуль не сможет продолжить работу, после верификации выведется сообщение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: check channel done error"
```

В дереве устройств в Codesys пиктограмма программного модуля окрасится в красный цвет.

Если не обнаружены критические ошибки, то выведется сообщение:

```
"elsyma_m01_gsm[10.9.32.1]_msys..tsync[1]:: ceb_checkch: check channel done ok"
```

В дереве устройств в Codesys пиктограмма программного модуля окрасится в зеленый цвет (если контроллер в режиме старт), или примет серый цвет (если контроллер в режиме стоп).

Приложение II (справочное)

Перечень изменений программного обеспечения и РЭ на контроллер Элсима

Таблица П.1 – Перечень произведенных изменений в программном обеспечении и РЭ

Дата изменения	Номера версий		Описание изменений
19.07.18	Версия системн. ПО	3.2.0 (сборка 10500)	<p>Расширение функциональности:</p> <ol style="list-style-type: none"> 1. С данного момента (18.07.18) контроллер производится в металлическом корпусе. Нумерация разъемов DIN1, DIN2, DOUT меняется на противоположное. 2. Старые версии контроллера в данной системе не поддерживаются (будет неправильно работать дискретный ввод/вывод). 3. Имя системы меняется на elm-m01r2-base-v3.2.0.xxxx.zip, elm-m01r2-web-v3.2.0.xxxx.zip 4. Добавлен вывод строки при изменении режима безопасности (DIP2) 5. Добавлена проверка версий программных модулей 6. Увеличено максимальное допустимое количество каналов для ModBus RTU <p>Корректировка документации:</p> <ol style="list-style-type: none"> 7. Добавлено исполнение контроллера Элсима-M01 «ревизия 2.0» и иллюстрирующие схемы к ней в раздел «1 Характеристики и устройство контроллера». 8. В приложении А исправлены схемы подключения в соответствии с новым исполнением контроллера Элсима «ревизия 2.0». 9. Исправлена схема «Модель tsync. Схема передачи данных в сервис NTP» в соответствии с новым исполнением контроллера Элсима «ревизия 2.0».
	Версия пакета поддержки (TSP)	3.2.0 (сборка 10495)	
	Версия CoDeSys	3.5.11.10	
	Версия сопроцессора	0.0.2.1	
	Версия РЭ M01 rev2	01	
	Версия РП на IEC104 rev2	01	
	УВВ ЭЛСИМА-A01	0.2.1.4	
	Версия РЭ A01	06	
	УВВ ЭЛСИМА-D01	0.2.1.2	
	Версия РЭ D01	10	
	УВВ ЭЛСИМА-DA01	0.2.1.4	
Версия РЭ DA01	08		
19.11.18	Версия системн. ПО	3.2.0 (сборка 10500)	<p>Корректировка документации:</p> <p>Убрано описание контроллера в пластиковом корпусе.</p>
	Версия пакета поддержки (TSP)	3.2.0 (сборка 10519)	
	Версия CoDeSys	3.5.11.10	

Контроллер программируемый логический Элсима

Таблица П.1 – Перечень произведенных изменений в программном обеспечении и РЭ

Дата изменения	Номера версий		Описание изменений
	Версия сопроцессора	0.0.2.1	
	Версия РЭ M01 rev2	02	
	Версия РП на IEC104 rev2	01	
	УВВ ЭЛСИМА-A01	0.2.2.0	
	Версия РЭ A01	03	
	УВВ ЭЛСИМА-D01	0.2.2.0	
	Версия РЭ D01	03	
	УВВ ЭЛСИМА-DA01	0.2.2.1	
	Версия РЭ DA01	03	
21.01.18	Версия системн. ПО	3.3.0 (сборка 11439)	Расширение функциональности: 1. Добавлен коннектор NetControl->StaticRoute. 2. Добавлены параметры управления каналом TCP (KeepAlive). 3. Добавлен сервисный режим IP - DHCP/Static, сервисный GateWay. 4. Добавлено автоматическое приведение ранее созданных проектов к последней версии TSP. 5. Добавлены параметры версии TSP, в которой создан проект, последняя версия TSP обновления проекта. 6. Добавлены комментарии битов калибровки модулей УВВ. 7. В модули УВВ добавлен параметр расширенной информации (SoftId, ModuleID). 8. Внесены изменения в отображение состояния модулей (программный и аппаратных) в различных ситуациях. 9. В модули УВВ добавлен параметр расширенной информации (SoftId, ModuleID) 10. Добавлено чтение и установка сервисного GateWay из .cfg. 11. Добавлено чтение и установка режима работы сервисного интерфейса из .cfg. 12. Добавлена обработка коннектора NetControl->StaticRoute 13. Добавлен биндинг IP-адреса. 14. Добавлен Bind CommSlot к определенному IP-адресу. 15. Добавлены в конфигурацию параметры KeepAlive. Передаются при инициализации CommSlot. 16. Добавлена поддержка библиотеки SysInfoLibrary. 17. Добавлен параметр для идентификации исполнения контроллера (узел CPU_INFO параметры RealModName, ExtInfo). 18. Добавлена установка IP-адреса в зависимости от заданного режима в .cfg. 19. По умолчанию в системе, начиная с 0.3.3.0, режим установки IP по DHCP. 20. В случае отсутствия режима в .cfg, задается DHCP. 21. Скорректирован скрипт /etc/udev/scripts/mount.sh, который автоматически выполняется при монтировании устройств USB,SD. 22. Проводиться проверка файловой системы (ФС) EXT3 и VFAT.
	Версия пакета поддержки (TSP)	3.3.0 (сборка 11436)	
	Версия CoDeSys	3.5.11.10	
	Версия сопроцессора	0.0.3.2	
	Версия РЭ M01 rev2	03	
	Версия РП на IEC104 rev2	01	
	УВВ ЭЛСИМА-A01	0.2.3.0	
	Версия РЭ A01	02	
	УВВ ЭЛСИМА-D01	0.2.2.0	
	Версия РЭ D01	02	
	УВВ ЭЛСИМА-DA01	0.2.3.0	

Таблица П.1 – Перечень произведенных изменений в программном обеспечении и РЭ

Дата изменения	Номера версий		Описание изменений
	Версия РЭ DA01	02	<p>23. Добавлены библиотеки e2fsprogs-e2fsck libcom-err2 libe2p2 libext2fs2 для обеспечения работ по корректировкам ФС ext3.</p> <p>24. Добавлена служба фиксации состояний компоненты (в устройстве Device новый параметр).</p> <p>25. Унифицированы сообщения по старту и завершению работы.</p> <p>26. Из конфигурации убран коннектор DefHost.</p> <p>27. Добавлена проверка непривязанных ключей и слотов, попытка загрузки проекта с некорректной запрещена.</p> <p>28. Добавлена проверка обнаружения нулевых сетевых параметров в CPU_INFO.</p> <p>29. Скорректированы атрибуты для параметров объектов определенных сигналов UBB.</p> <p>30. Внесены корректировки в карту сигналов (установлено ограничение в 1000 сигналов).</p> <p>Исправление ошибок:</p> <ol style="list-style-type: none"> Исправлена логика маппинга в ModBus TCP Master, ModBus RTU Master. Исправлены сообщения вывода в log в разных ситуациях. Изменена идентификация сообщения (ранее – «C0-> ...») Исправлена ошибка при добавлении default gateway при обновлении конфигурации. Исправлена переинициализация программных и аппаратных модулей. Исправлена работа DHCP в заводском режиме. MBTCP.Slave: в редакторе канала исправлена кнопка "привязка и автоименование". Исправлена Ошибка формирования счетчиков маршрутов в StaticRoute. Исправлена ошибка пересчета количества сигналов в MBTCPM.Server, IEC104S.ASDU. Исправлено максимальное количество каналов в MBTCP.Slave: максимальное количество каналов перенесено в параметры. <p>Корректировка документации:</p> <ol style="list-style-type: none"> Добавлено описание работы индикатора L3. Добавлены разделы и приложения: «Приведение ранее созданных проектов к текущей версии TSP»; «Установка двух и более версий CoDeSys»; «Библиотека для доступа к системной информации контроллера»; «Управление сетевой подсистемой контроллера (коннектор NetControl)»; <p>Приложение «Перечень сообщений о состояниях программных и аппаратных модулей».</p> <ol style="list-style-type: none"> Внесены изменения в приложение «Изменение сетевых параметров контроллера». Внесены изменения в раздел «Настройка соединения с контроллером»: добавлено описание режима DHCP. Изменен раздел «Настройка параметров контроллера»: добавлено описание устройства ELSYMA_M01, расширено описание CPU_INFO: добавлены подразделы «Установка сервисного IP-адреса» и «Установка RTC». В раздел «Добавление коммуникационного слота CommSlot» добавлена таблица с описанием системных параметров (KeepAlive). Убран раздел с описанием DefHost. В раздел «Сигналы аналогового выхода» добавлена таблица «Модуль CPU_IO. Диагностика каналов аналогового выхода (AOut1Diag...AOut2Diag)».
22.03.19	Версия системн. ПО	3.3.0 (сборка 11439)	<p>Корректировка документации:</p> <p>Добавлен знак утверждения типа на титульный лист и на рисунок 1.4 (знак для сертифицированных исполнений контроллера).</p>
	Версия пакета поддержки (TSP)	3.3.0 (сборка 11484)	
	Версия CoDeSys	3.5.11.10	

Таблица П.1 – Перечень произведенных изменений в программном обеспечении и РЭ

Дата изменения	Номера версий		Описание изменений
	Версия сопроцессора	0.0.3.4	
	Версия РЭ М01 rev2	04	
	Версия РП на IEC104 rev2	01	
	УВВ ЭЛСИМА-А01	0.2.4.0	
	Версия РЭ А01	04	
	УВВ ЭЛСИМА-D01	0.2.3.0	
	Версия РЭ D01	03	
	УВВ ЭЛСИМА-DA01	0.2.4.0	
	Версия РЭ DA01	04	

Контактная информация

По всем вопросам, связанным с эксплуатацией контроллера, обращаться в сервисный центр АО "ЭлеСи":

тел.: +7 (3822) 49-94-94

E-mail: service@elesy.ru

Сервисный центр располагается в г. Томске (часовой пояс +4 МСК).

При обращении просим сообщать следующие данные:

- полное наименование изделия (указано на изделии или в паспорте);
- проект *CoDeSys*, в котором возникает проблема;
- версия установленного на компьютере пакета *EleSy ELSYMA TSP (Target Support Package)*;
- подробное описание проблемы (попытайтесь наиболее полно пояснить суть проблемы и обстоятельства или условия, которые привели к ней).

