

Руководство по программированию K15.LX1

Версия 1.0

Оглавление

1. Общие сведения
2. Создание проекта
3. Подключение к ПЛК
 - 3.1. Изменение IP адресов портов ETH1 и ETH2
4. Добавление модулей K15
5. Создание соединения Modbus RTU/ASCII
 - 5.1. Slave RTU/ASCII
 - 5.2. Master RTU/ASCII
6. Создание соединения Modbus TCP
 - 6.1. Server TCP
 - 6.2. Client TCP
7. Подключение библиотек
 - 7.1 Библиотека K15_LX1_gpio
 - 7.2 Библиотека K15_LX1_retain

1. Общие сведения

K15.LX1 – программируемый логический контроллер (ПЛК) общего применения для решения различных задач автоматизации. Для создания программной логики его работы используется среда разработки (IDE) Codesys 3.5 SP16 Patch4, которая позволяет программировать прибор, используя стандартные языки МЭК 61131-3. Это облегчает реализацию проекта и существенно сокращает сроки разработки. Подробнее о среде разработки, а также об основных приемах программирования в этой среде можно узнать в справке в составе среды либо на сайте производителя www.codesys.com.

2. Создание проекта

Установите среду разработки Codesys 3.5 SP16 Patch4 с установками по умолчанию. После этого установите таргет-файл ПЛК. Для этого выберите «Инструменты > Репозиторий устройств» и нажмите «Установить...». В открывшемся окне выберите на локальном диске таргет-файл K15.LX1.devdesc. Устройство должно появиться в репозитории в разделе «ПЛК».

Далее необходимо добавить EDS-файлы модулей, которые будут установлены в корзине. Это делается аналогичным способом через репозиторий устройств. Но при указании файлов на локальном диске необходимо выбрать тип файлов «EDS и DCF файлы» в нижнем правом углу окна Проводника. Устройства должны появиться в репозитории в разделе «Промышленные сети (fieldbus) / CANopen / Удаленное устройство».

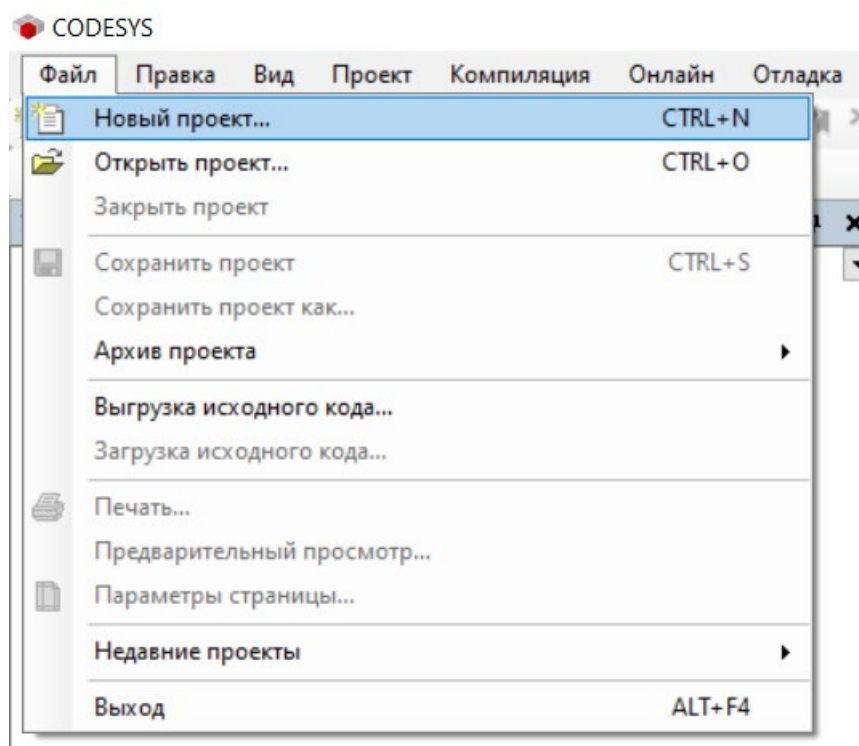


Рис.1. Создание проекта.

Теперь можно создавать новый проект. Для этого выберите «Файл > Новый проект».

В открывшемся окне «Новый проект» выберите «Стандартный проект», имя проекта и путь, по которому будут созданы файлы проекта. В следующем окне «Стандартный проект» выберите устройство K15.LX1 (A and T), а также язык программирования базового объектного блока (POU) PLC_PRG.

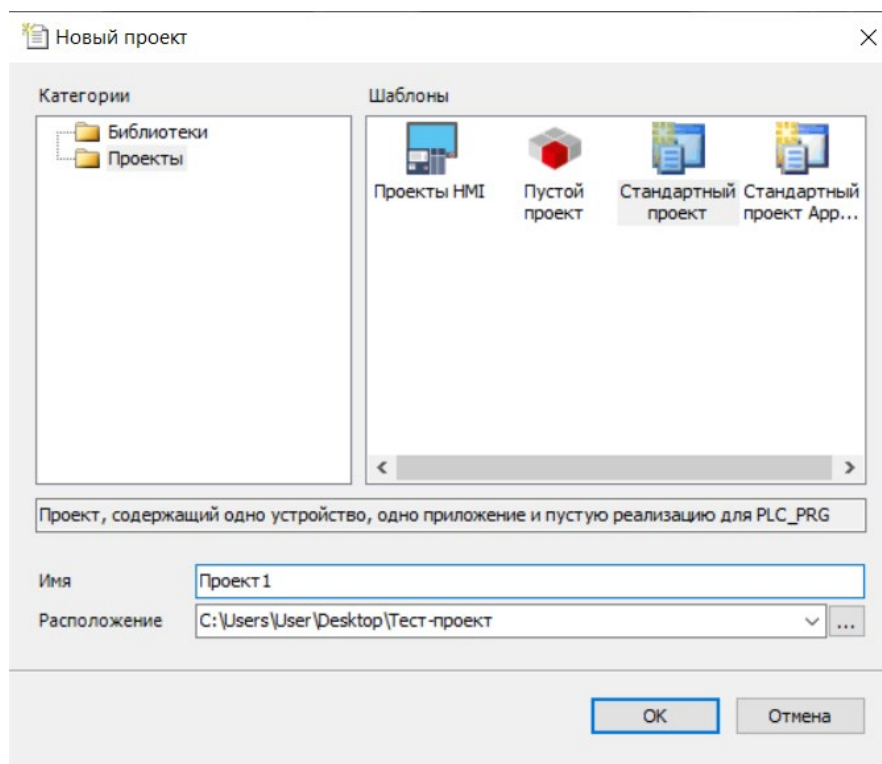


Рис.2. Стандартный проект.

3. Подключение к ПЛК

Соберите корзину ПЛК с модулями ввода-вывода, подключите порт ETН1 либо ETН2 ПЛК к ПК либо локальной сети и подайте питание согласно Руководству по эксплуатации. Убедитесь в установлении соединения с портом ETН1 либо ETН2 по индикации на портах, а также в том, что запустилось встроенное ПО ПЛК. Об этом свидетельствует короткий сигнал (1-2 сек) через несколько секунд после подачи питания.

Во вновь созданном проекте Codesys в дереве проекта слева дважды щелкните мышью на Device (K15.LX1). В открывшемся окне в правом поле ввода, соответствующем адресу ПЛК, введите IP адрес порта, к которому подключен ПК.

Внимание! ПЛК имеет статические базовые адреса портов 192.168.1.251 (ETН1) и 192.168.1.252 (ETН2). Для их смены см. Изменение IP адресов портов ETН1 и ETН2.

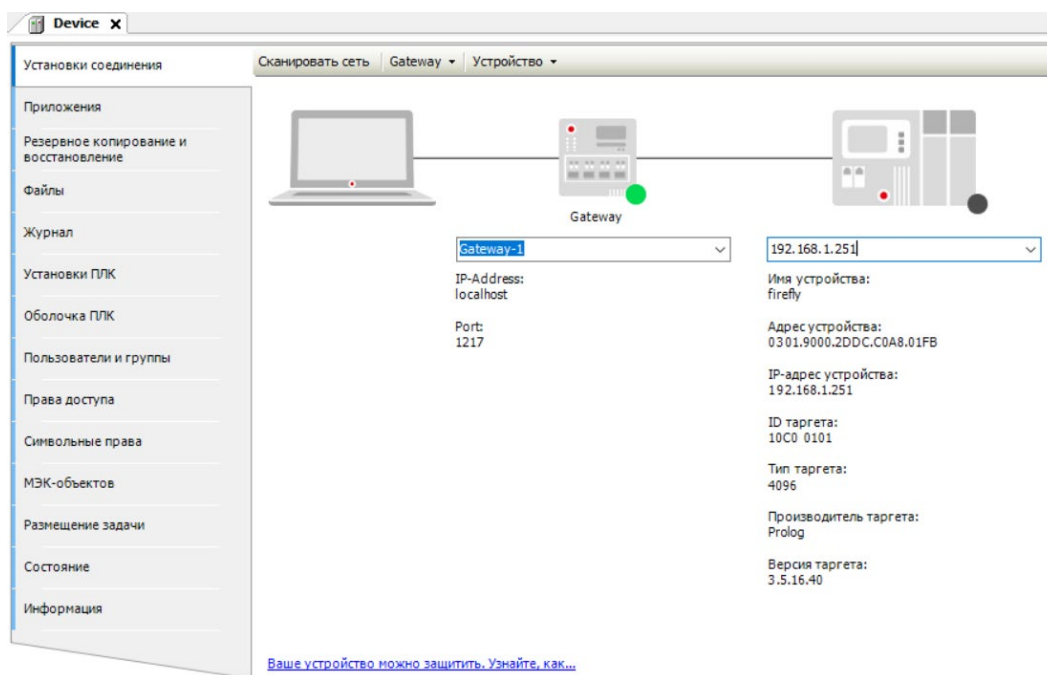


Рис.3. Подключение к ПЛК.

После ввода IP адреса нажмите Enter. Если подключение установлено, в появившемся окне «Вход в систему» введите имя пользователя «admin» и пароль «admin». После этого можно подключиться в режим Онлайн, нажав «Онлайн > Логин». В этом режиме можно загрузить проект в ПЛК, создать загрузочное приложение, сбросить ПЛК, запустить/остановить исполнение проекта и тд. Для корректировки проекта необходимо снова перейти в режим Оффлайн, нажав «Онлайн > Отключение».

3.1. Изменение IP адресов портов ETН1 и ETН2.

Для смены базовых IP адресов ПЛК предусмотрено подключение через порт Debug (Type-C) на лицевой стороне прибора. Для этого понадобится:

- кабель USB A – USB Type-C
- терминальная утилита MobaXterm.

Подключите кабель USB к ПК, установите при необходимости драйвер CH либо FTDI виртуального COM-порта.

Запустите утилиту MobaXterm, выберите Session > Serial. В открывшейся вкладке укажите виртуальный COM-порт ПЛК и укажите скорость обмена.

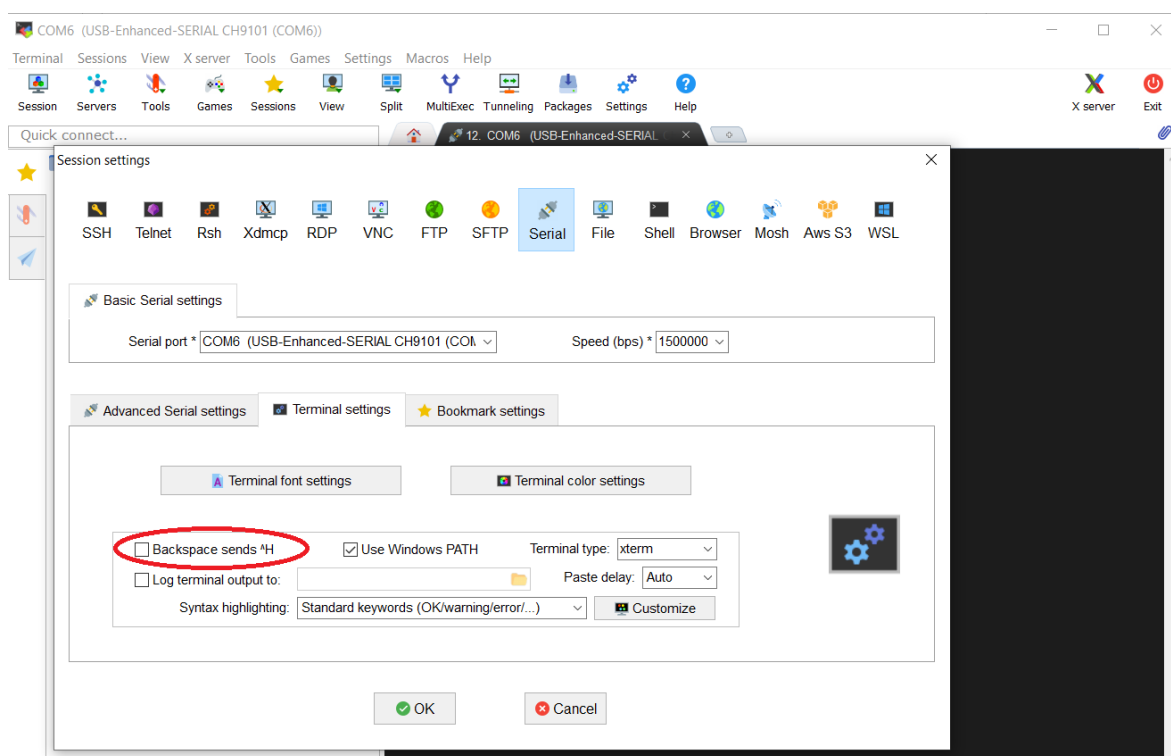


Рис.4. Утилита MobaXterm.

Внимание! В списке доступных скоростей отсутствует нужная скорость. Поэтому введите вручную в поле Speed (bps) скорость, равную 150000 (1,5 Mbps).

После этого перейдите во вкладку Terminal settings и снимите флажок «Backspace sends ^H». Затем нажмите OK. Если соединение установлено, появляется черное окно терминала справа. Нажмите Enter, после чего появится приглашение командной строки в каталог /root.

В этом каталоге на ПЛК находятся два скрипта изменения IP портов ETH1 и ETH2. Чтобы изменить параметры порта ETH1, введите в поле командной строки ./eth1.sh и нажмите Enter. При наличии соединения порта с сетью будет выведен текущий IP адрес, затем будет предложено ввести новый. Далее необходимо будет ввести gateway IP.

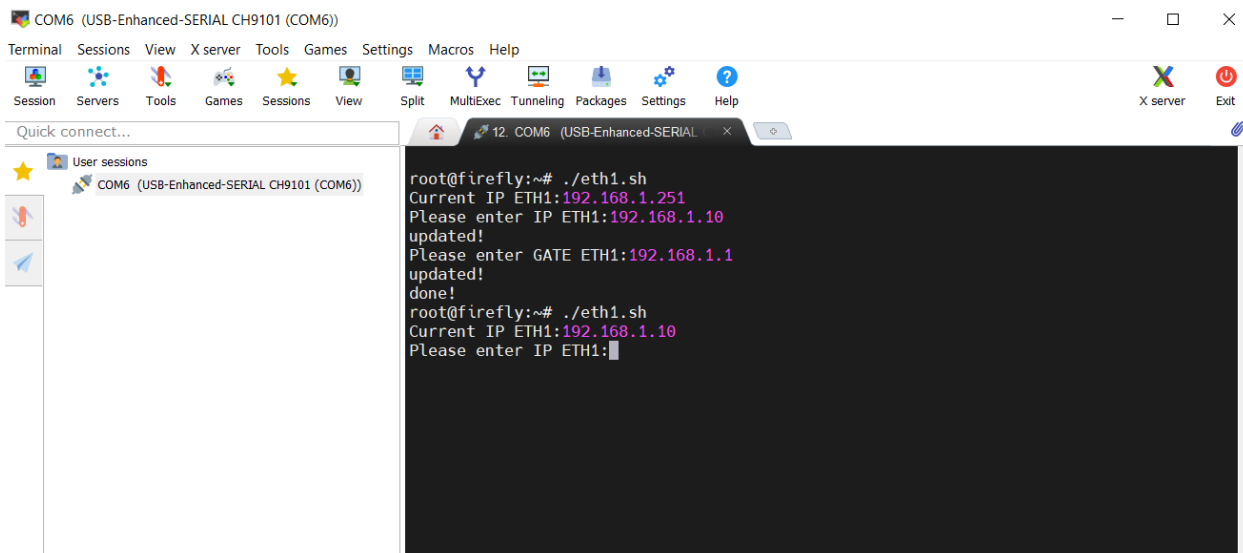


Рис.5. Изменение параметров порта.

При корректном вводе в командной строке появится сообщение «done!» и новые параметры применятся сразу без перезагрузки ПЛК. Аналогично настраиваются параметры порта ETH2 через скрипт eth2.sh.

4. Добавление модулей K15

Для работы с модулями K15 нажмите правой кнопкой мыши на объект «Device (K15.LX1)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети > CANbus» выберите объект CANbus и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

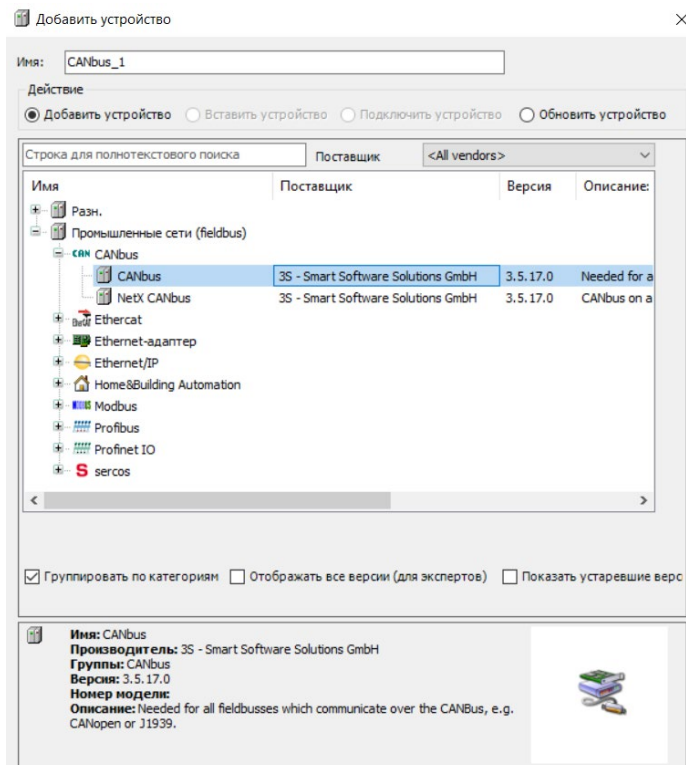


Рис.6. Добавление CANbus.

В дереве проекта двойным щелчком по объекту «CANbus (CANbus)» откройте вкладку «Общее» и установите **скорость передачи 500 кбит/с**.

Далее нажмите правой кнопкой мыши на объект «CANbus (CANbus)» и, выбрав «Добавить устройство...», перейдите в категорию «Промышленные сети (fieldbus) > CANopen > CANopenManager», выберите объект «CANopen_Manager» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

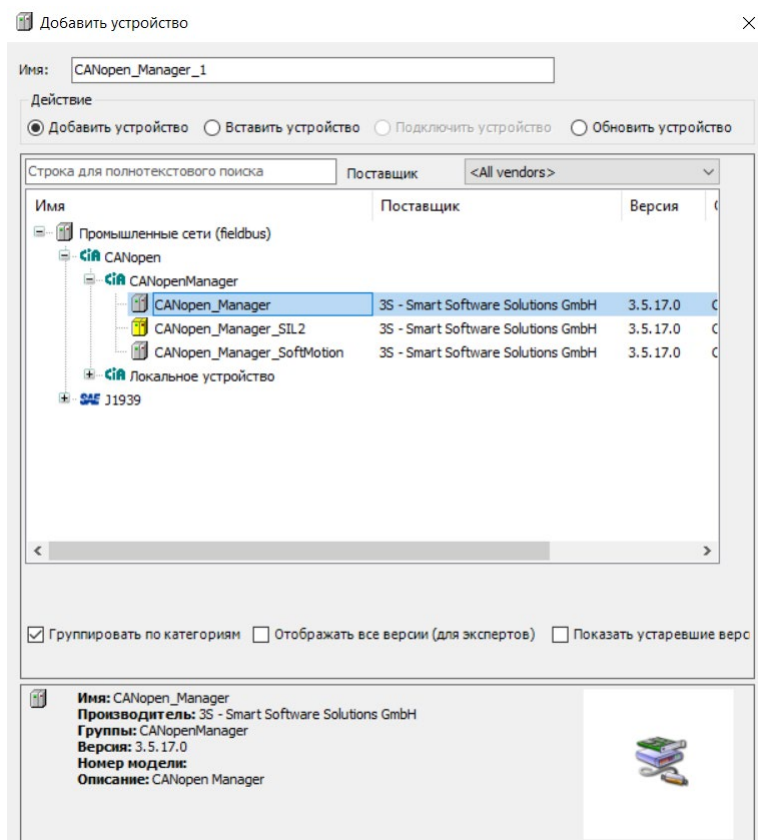


Рис.7. Добавление CANopen_Manager.

В дереве проекта двойным щелчком по объекту «CANopen_Manager (CANopen_Manager)» откройте вкладку «CANopen Соотнесение входов/выходов» и выберите в опциях цикла шины «MainTask».

Теперь можно добавлять модули K15. Для этого нажмите правой кнопкой мыши на объект «CANopen_Manager (CANopen_Manager)» и, выбрав «Добавить устройство...», перейдите в категорию «Промышленные сети (fieldbus) > CANopen > Удаленное устройство», в которой выберите те модули, которые установлены в корзину ПЛК. Модули можно выбрать последовательно по одному не закрывая окна выбора устройств, двойным щелчком добавляя каждый модуль в дерево проекта.

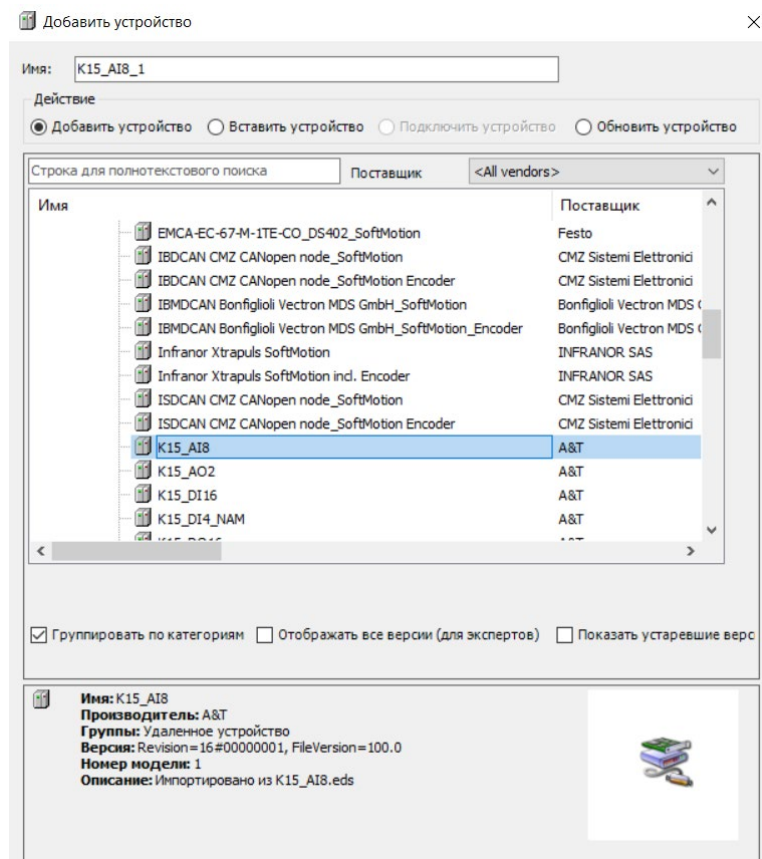


Рис.8. Добавление модулей.

После того, как модули выбраны, нужно настроить каждый из них в дереве проекта. Перейдите двойным щелчком во вкладку «Общее» модуля и убедитесь в совпадении назначенного ID узла и фактически установленного адреса модуля с помощью DIP переключателей. Затем во вкладке «CANopen Соотнесение входов/выходов» установите параметр обновления переменных в значение «Вкл.2 (всегда в задаче цикла шины)». Произведите данные действия для каждого из модулей.

После этого можно привязывать переменные проекта к сигналам модулей. Для этого перейдите во вкладку «CANopen Соотнесение входов/выходов» модуля и в поле нужного канала модуля произведите соотнесение, дважды кликнув на пустую область первого столбца. Затем, нажав кнопку «...», укажите переменную из нужного объекта, к которой будет привязан сигнал модуля.

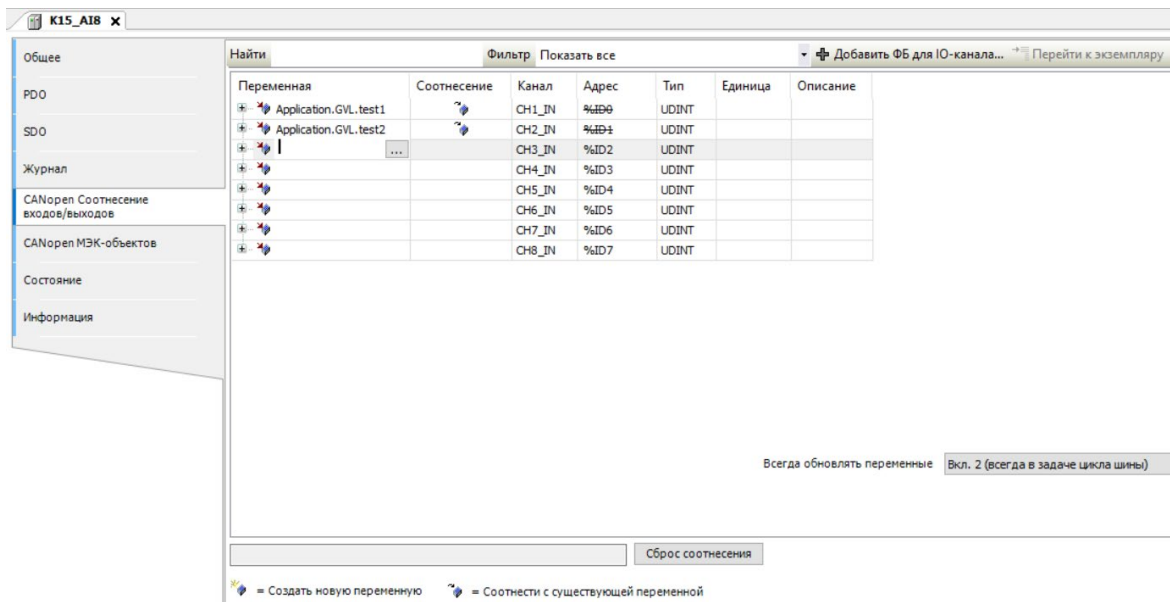


Рис.9. Привязка переменных к модулю.

Внимание! При первом подключении к ПЛК и загрузке проекта исполнение остановлено (Стоп), следовательно, шина CANbus также не запущена. Для запуска опроса модулей запустите исполнение (Отладка > Старт).

5. Создание соединения Modbus RTU/ASCII

Для создания соединения необходимо в дереве проекта добавить устройство «Modbus_COM», которое обеспечивает обмен по выбранному порту RS485/RS232. Нажмите правой кнопкой мыши на объект «Device (K15.LX1)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Порт Modbus Serial» выберите объект «Modbus COM» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

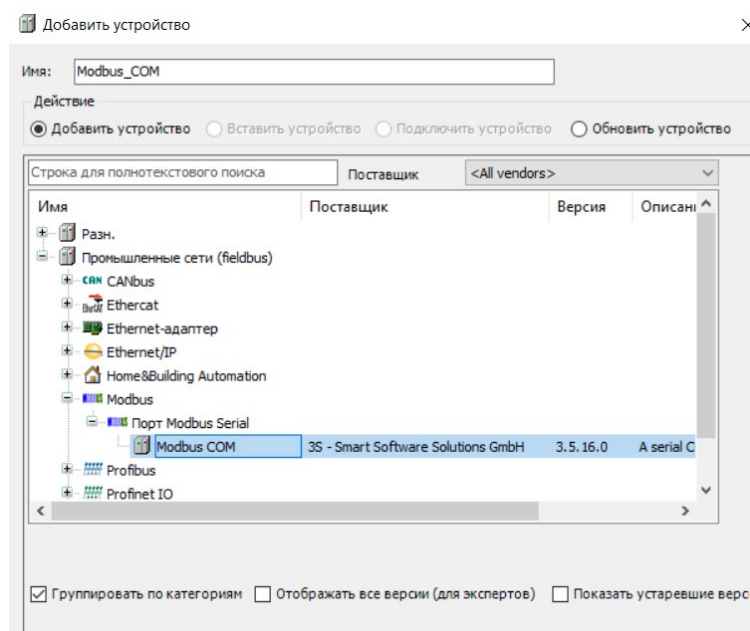


Рис.10. Добавление Modbus COM.

В дереве проекта двойным щелчком по объекту «Modbus_COM (Modbus COM)» откройте вкладку «Общее» и установите номер выбранного порта ПЛК согласно таблице:

Порт на ПЛК	COM-порт в проекте
RS485-1	1
RS485-2	2
RS485-3	3
RS232	4

Далее выберите требуемую скорость передачи, четность, и количество бит.

Внимание! Максимальное число объектов «Modbus_COM» не должно быть больше числа физических портов на ПЛК, то есть больше четырех. Каждый из четырех портов может выступать Master- либо Slave- устройством в любом сочетании.

5.1. Slave RTU/ASCII

Для создания Slave соединения (когда ПЛК принимает запросы мастер-узла) необходимо в дереве проекта добавить устройство «Modbus Serial Device». Нажмите правой кнопкой мыши на объект «Modbus_COM (Modbus COM)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Устройство Modbus Serial» выберите объект «Modbus Serial Device» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

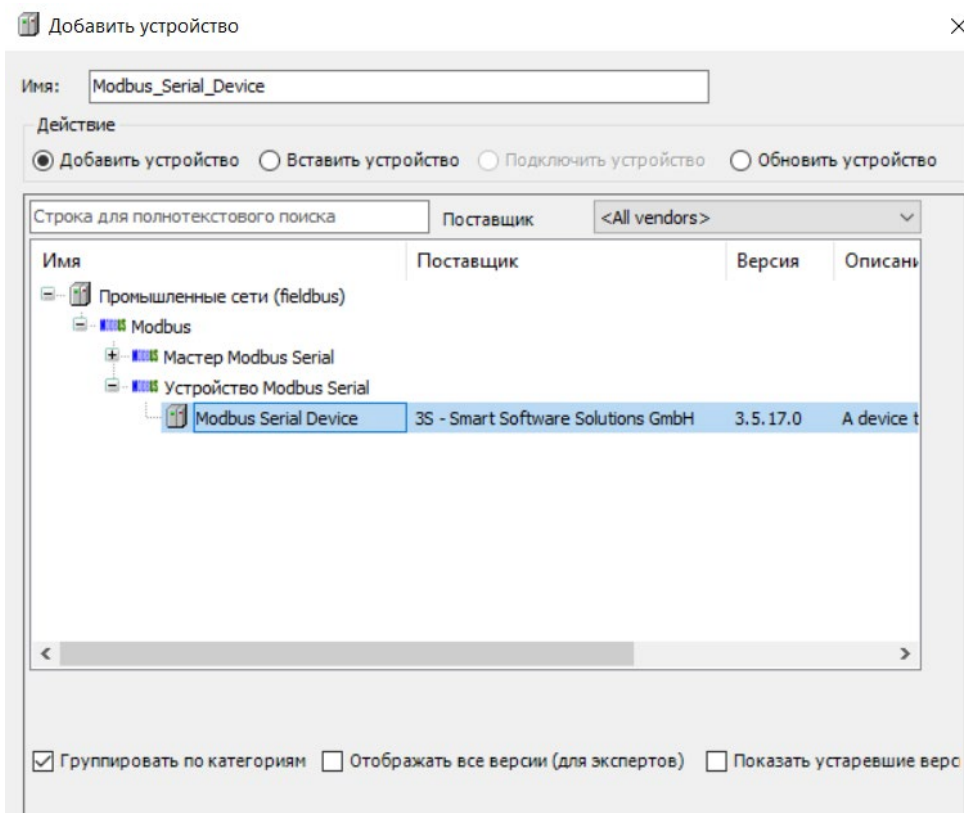


Рис.11. Добавление Modbus Serial Device.

В дереве проекта двойным щелчком по объекту «Modbus_Serial_device (Modbus Serial Device)» откройте вкладку «Modbus Serial Device» и установите необходимые параметры

устройства: ID (сетевой адрес Modbus), количество регистров хранения, входных регистров, дискретных битовых областей, а также смещения их адресов.

Во вкладке «Modbus Serial Device Соотнесение входов/выходов» установите параметр обновления переменных в значение «Вкл.2 (всегда в задаче цикла шины)». Затем выберите в опциях цикла шины «MainTask».

После этого можно в этой же вкладке осуществить привязку переменных проекта к регистрам Modbus аналогично привязке переменных к сигналам модулей K15.

5.2. Master RTU/ASCII

Для создания Master соединения (когда ПЛК передает запросы слейв-узлам) необходимо в дереве проекта добавить устройство «Modbus Master, COM Port». Нажмите правой кнопкой мыши на объект «Modbus_COM (Modbus COM)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Master Modbus Serial» выберите объект «Modbus Master, COM Port» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

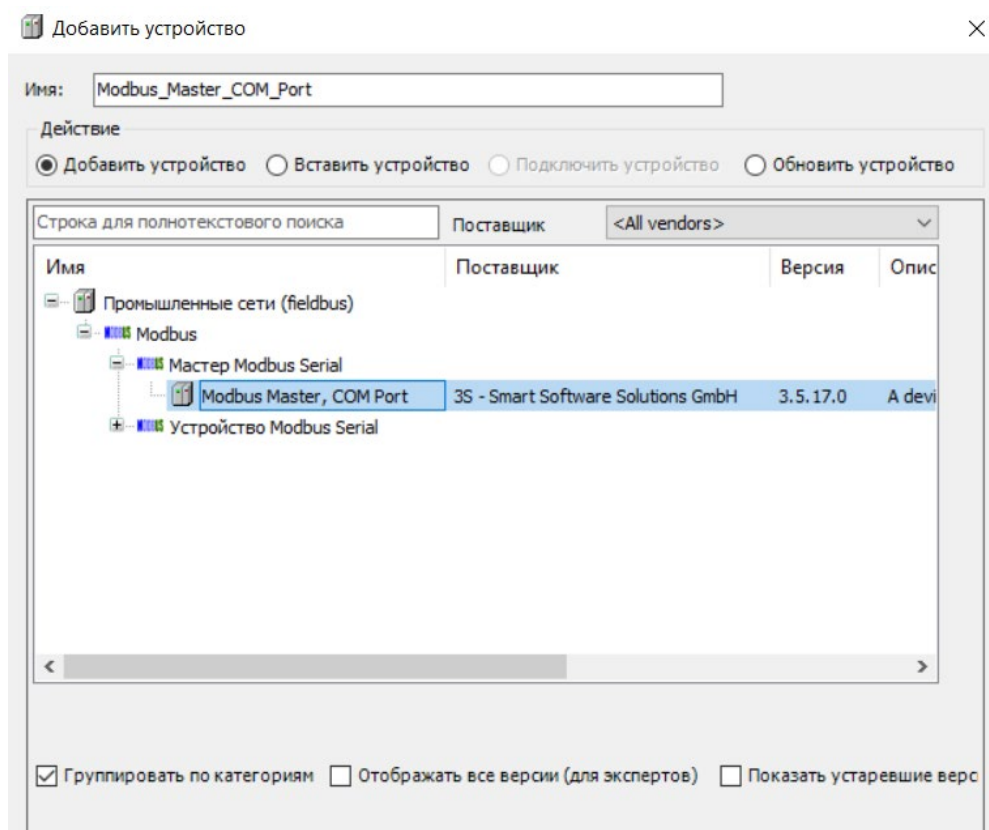


Рис.12. Добавление Modbus Master.

В дереве проекта двойным щелчком по объекту «Modbus_Master_COM_Port (Modbus Master, COM Port)» откройте вкладку «Общее» и установите параметры работы мастера. Также в этой вкладке необходимо установить флажок «Автоперезапуск соединения». Во вкладке «ModbusGenericSerialMaster Соотнесение входов/выходов» установите в опциях цикла шины «MainTask».

Затем нужно добавить слейв-узлы, которые предполагается опрашивать. Нажмите правой кнопкой мыши на объект «Modbus_Master_COM_Port (Modbus Master, COM Port)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Слейв Modbus Serial» выберите объект «Modbus Slave, COM Port» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

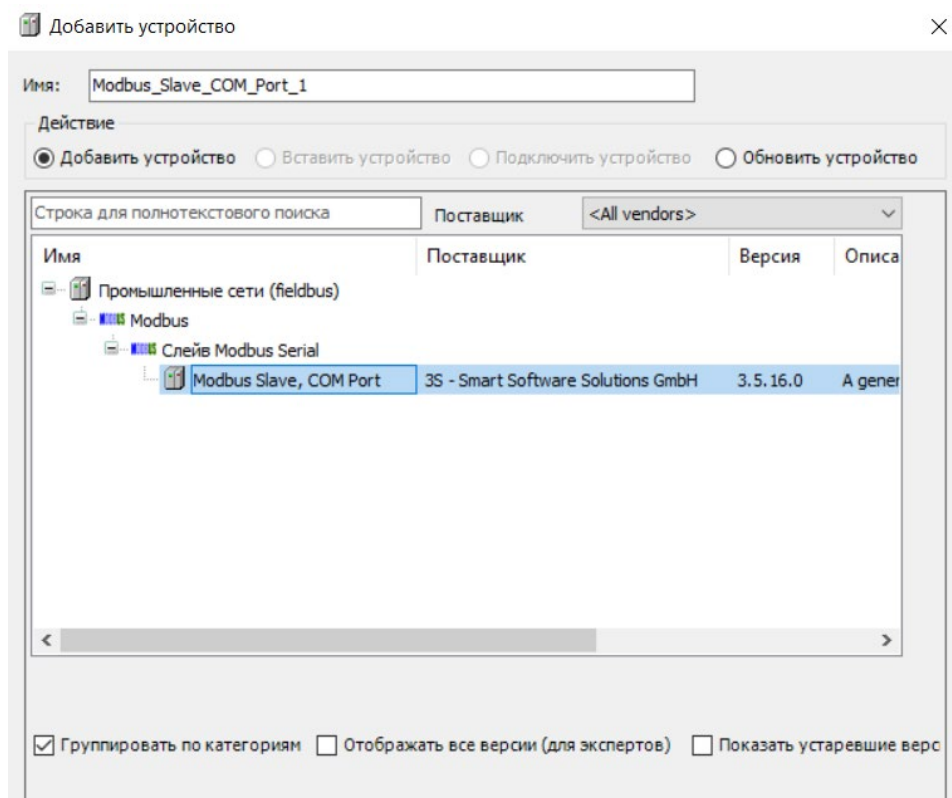


Рис.13. Добавление слейв-узлов.

После добавления узлов настройте каждый из них. В дереве проекта двойным щелчком по объекту «Modbus_Slave_COM_Port (Modbus Slave, COM Port)» откройте вкладку «Общее» и установите параметры работы узла: адрес и таймаут ответа. Затем во вкладках «Канал Modbus Slave» и «Modbus Slave Init» добавьте необходимые каналы чтения/записи регистров узла.

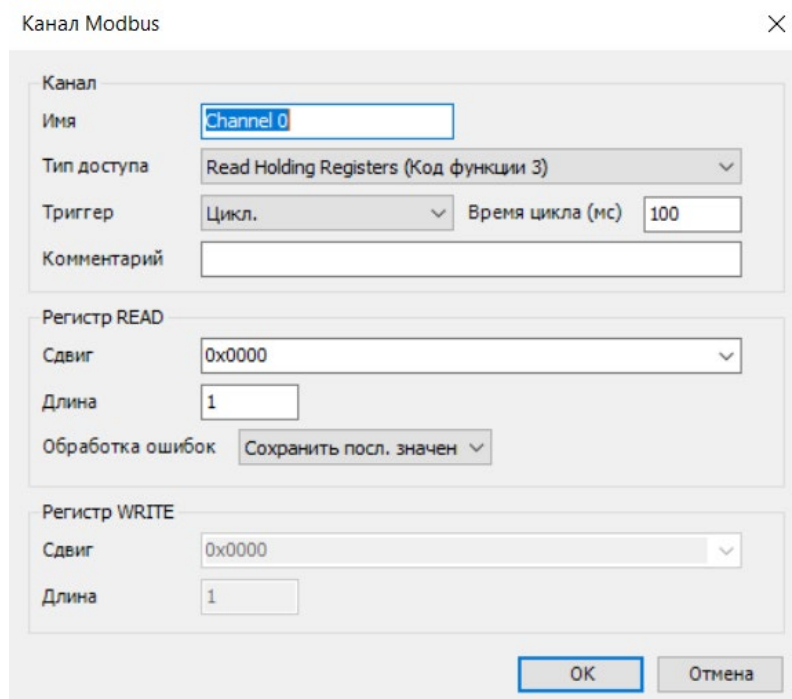


Рис.14. Добавление каналов чтения/записи регистров узла.

6. Создание соединения Modbus TCP

Для создания соединения необходимо в дереве проекта добавить устройство «Ethernet (Ethernet)», которое обеспечивает обмен по выбранному порту ETH1/ETH2. Нажмите правой кнопкой мыши на объект «Device (K15.LX1)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Ethernet-адаптер» выберите объект «Ethernet» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

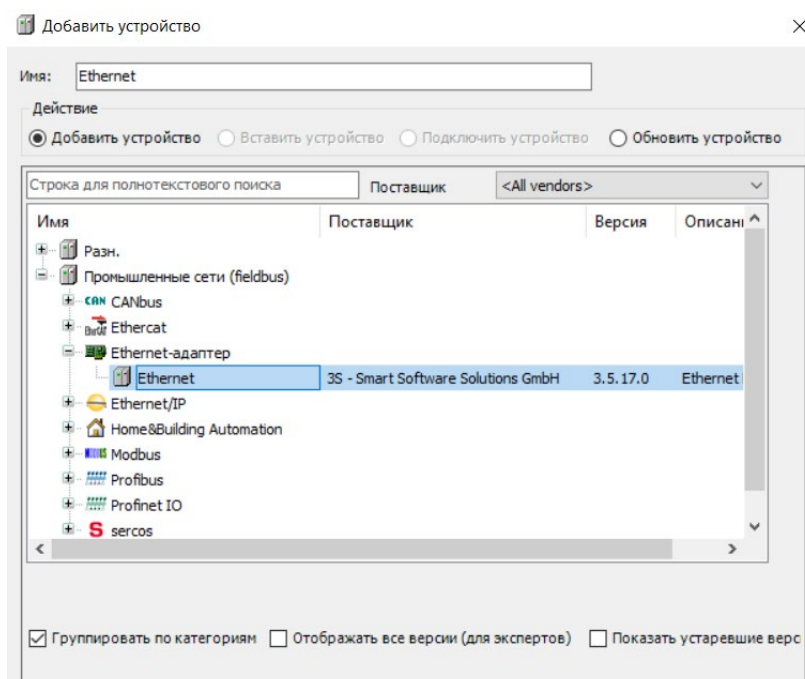


Рис.15. Добавление Ethernet.

В дереве проекта двойным щелчком по объекту «Ethernet (Ethernet)» откройте вкладку «Конфигурация ethernet» и установите интерфейс сети, соответствующий выбранному порту, согласно таблице:

Порт на ПЛК	Интерфейс в проекте
ETH1	eth0
ETH2	eth1

Для этого нажмите на кнопку «...» и выберите нужный интерфейс в открывшемся окне «Сетевые адаптеры». Поля IP адреса, маски сети и шлюза по умолчанию будут прочитаны из ПЛК и вставлены в конфигурацию Ethernet.

Внимание! Для того, чтобы данные об интерфейсах загрузились из ПЛК, необходимо предварительно убедиться в успешном соединении с ПЛК. Затем отключиться от ПЛК. При нажатии кнопки «...» подключение установится автоматически.

6.1. Server TCP

Создание сервера Modbus TCP похоже на создание Slave RTU. Для создания Server TCP (когда ПЛК принимает запросы клиентов TCP) необходимо в дереве проекта добавить устройство «ModbusTCP Slave Device». Нажмите правой кнопкой мыши на объект «Ethernet (Ethernet)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Слейв-устройство ModbusTCP» выберите объект «ModbusTCP Slave Device» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

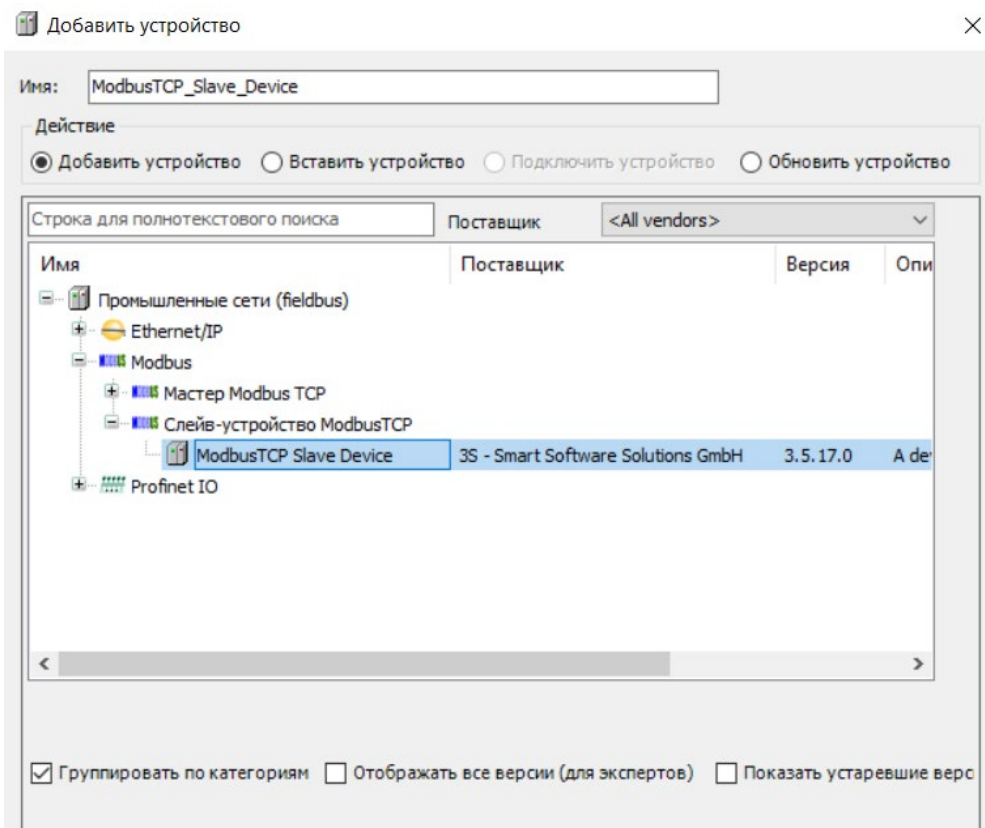


Рис.16. Добавление ModbusTCP Slave Device.

В дереве проекта двойным щелчком по объекту «ModbusTCP_Slave_Device (ModbusTCP Slave Device)» откройте вкладку «Страница конфигурации» и установите необходимые параметры устройства: порт, количество регистров хранения, входных регистров, дискретных битовых областей, а также смещения их адресов.

Во вкладке «Modbus TCP Slave Device Соотнесение входов/выходов» установите параметр обновления переменных в значение «Вкл.2 (всегда в задаче цикла шины)». Затем выберите в опциях цикла шины «MainTask».

После этого можно в этой же вкладке осуществить привязку переменных проекта к регистрам Modbus аналогично привязке переменных к сигналам модулей K15.

6.2. Client TCP

Создание клиента Modbus TCP похоже на создание Master RTU. Для создания клиент-соединения (когда ПЛК передает запросы серверу TCP) необходимо в дереве проекта добавить устройство «Modbus TCP Master». Нажмите правой кнопкой мыши на объект «Ethernet (Ethernet)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Master Modbus TCP» выберите объект «Modbus TCP Master» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

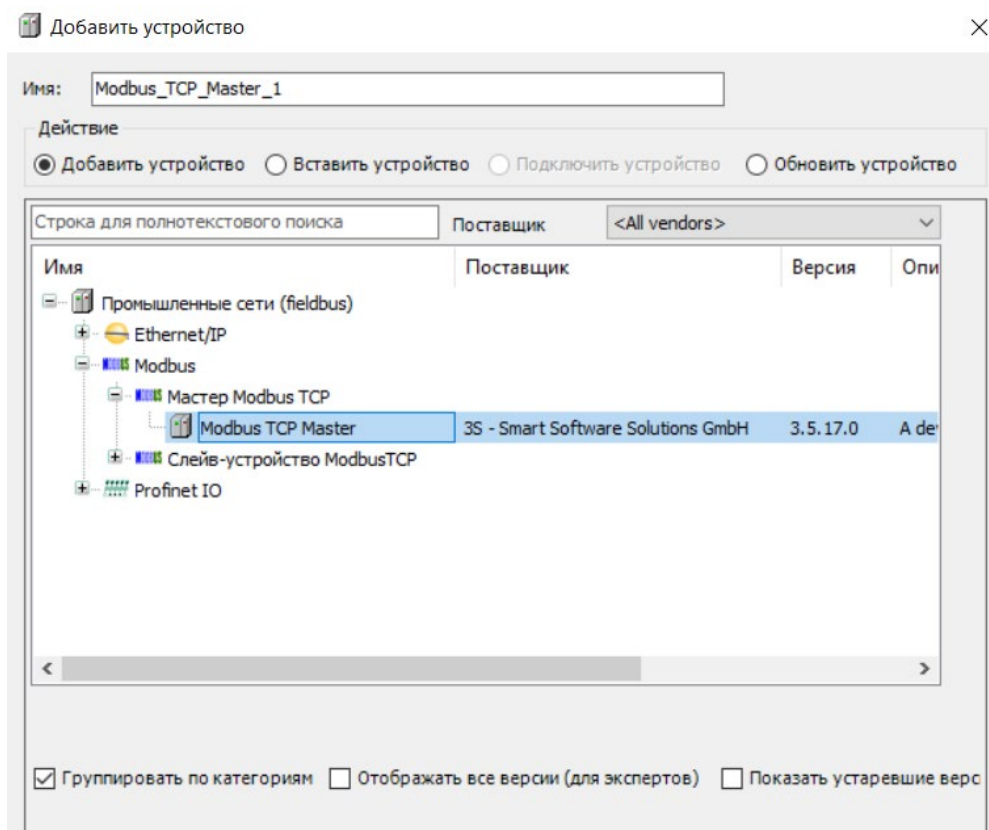


Рис.17. Добавление Modbus TCP Master.

В дереве проекта двойным щелчком по объекту «Modbus_TCP_Master (Modbus TCP Master)» откройте вкладку «Общее» и установите параметры работы мастера. Также в этой вкладке необходимо установить флажок «Автоподключение». Во вкладке

«ModbusTCPMaster Соотнесение входов/выходов» установите в опциях цикла шины «MainTask».

Затем нужно добавить слейв-узлы, которые предполагается опрашивать. Нажмите правой кнопкой мыши на объект «Modbus_TCP_Master (Modbus TCP Master)» в дереве проекта и выберите «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Слейв Modbus TCP» выберите объект «Modbus TCP Slave» и двойным щелчком добавьте его в дерево проекта. Закройте окно добавления устройств.

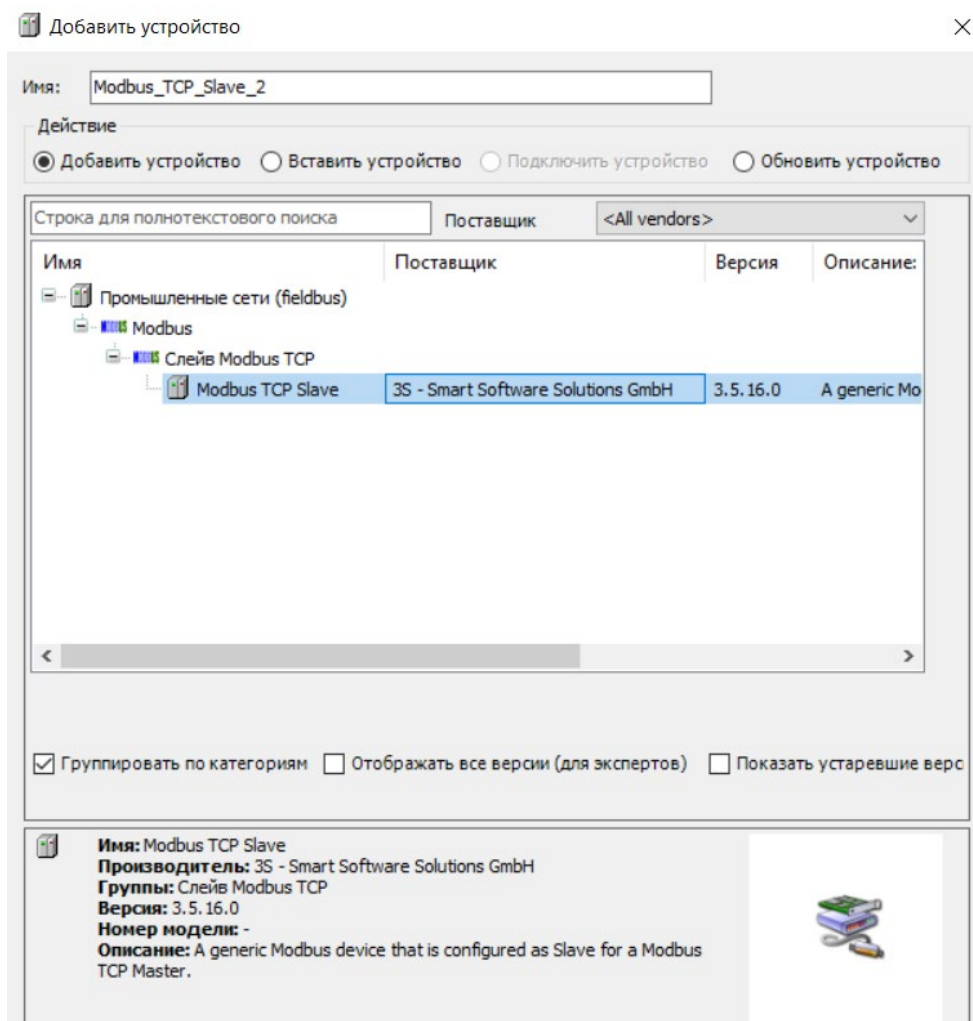


Рис.18. Добавление слейв-узлов.

После добавления узлов настройте каждый из них. В дереве проекта двойным щелчком по объекту «Modbus_TCP_Slave_(Modbus TCP Slave)» откройте вкладку «Общее» и установите параметры работы узла: IP-адрес, порт и таймаут ответа. Затем во вкладках «Канал Modbus Slave» и «Modbus Slave Init» добавьте необходимые каналы чтения/записи регистров узла.

Канал Modbus ×

Канал		
Имя	<input type="text" value="Channel 0"/>	
Тип доступа	Read Holding Registers (Код функции 3) ▾	
Триггер	Цикл. ▾	Время цикла (мс) <input type="text" value="100"/>
Комментарий	<input type="text"/>	
Регистр READ		
Сдвиг	<input type="text" value="0x0000"/>	
Длина	<input type="text" value="1"/>	
Обработка ошибок	Сохранить посл. значен ▾	
Регистр WRITE		
Сдвиг	<input type="text" value="0x0000"/>	
Длина	<input type="text" value="1"/>	
		<input type="button" value="OK"/> <input type="button" value="Отмена"/>

Рис.19. Добавление каналов чтения/записи регистров узла.

7. Подключение библиотек

Библиотеки предназначены для расширения функционала ПЛК, а также для упрощения реализации тех или иных частей проекта. Чтобы использовать библиотеку, необходимо ее подключить в свой проект. Для этого сначала библиотека добавляется в репозиторий библиотек Codesys.

Перейдите в «Инструменты > Репозиторий библиотек» и щелкните «Установить...». Укажите расположение файла компилированной библиотеки. Закройте окно «Репозиторий библиотек».

Чтобы подключить библиотеку, в дереве проекта дважды нажмите на объект «Менеджер библиотек». В открывшемся окне нажмите «Добавить библиотеку» и выберите нужную библиотеку из репозитория.

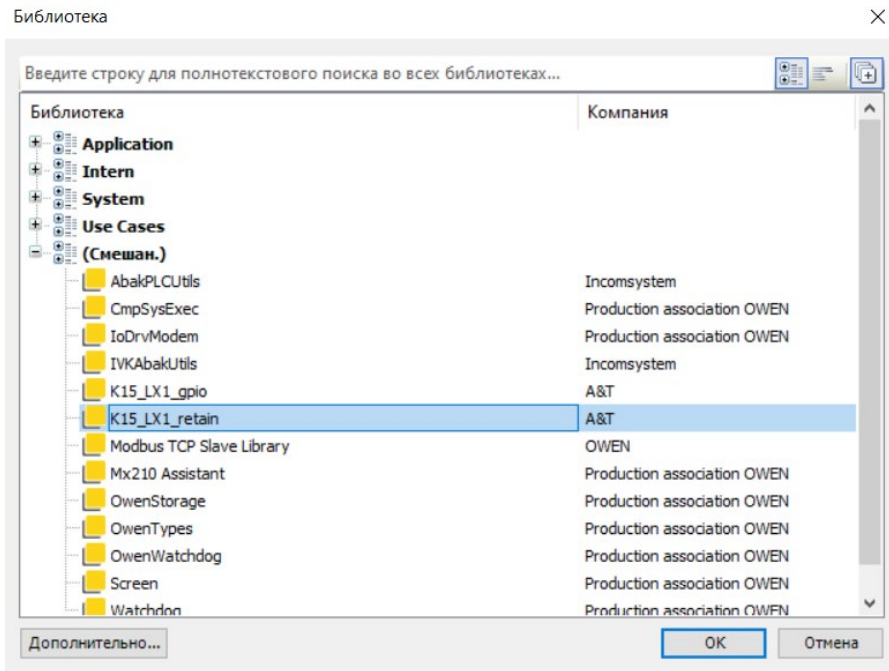


Рис.20. Подключение библиотеки в проект.

7.1. Библиотека K15_LX1_gpio

Данная библиотека реализует работу встроенных дискретных входов DI и дискретных выходов DO. Для использования в проекте доступны два функциональных блока библиотеки:

GET_DI – возвращает состояние выбранного DI в формате BOOL.

SET_DO – записывает состояние выбранного DO в формате BOOL.

Разместите нужные функциональные блоки в любом из программных модулей и привяжите к соответствующим переменным.

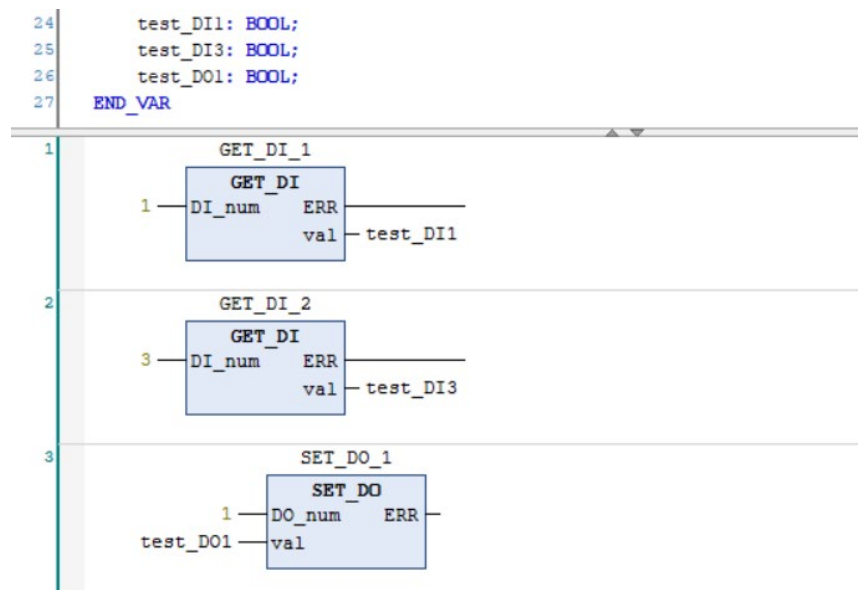


Рис.21. Пример использования библиотеки K15_LX1_gpio

Параметры функционального блока GET_DI:

DI_num – номер канала DI (1..3),

val – состояние канала (TRUE/FALSE),

ERR – ошибка получения состояния (TRUE/FALSE),

Параметры функционального блока SET_DO:

DO_num – номер канала DO (1..2),

val – состояние канала (TRUE/FALSE),

ERR – ошибка записи состояния (TRUE/FALSE).

7.2. Библиотека K15_LX1_retain

Данная библиотека позволяет создавать retain/persistent переменные в памяти ПЛК. Для использования в проекте доступны две функции библиотеки:

MAP_RETAIN – привязывает переменную к retain/persistent области.

UPDATE_RETAIN – обновляет значения переменных, привязанных к retain/persistent области.

Функцию MAP_RETAIN разместите в любом программном модуле. Функцию UPDATE_RETAIN разместите в любом циклически исполняемом программном модуле, желательно с наименьшим периодом цикла.

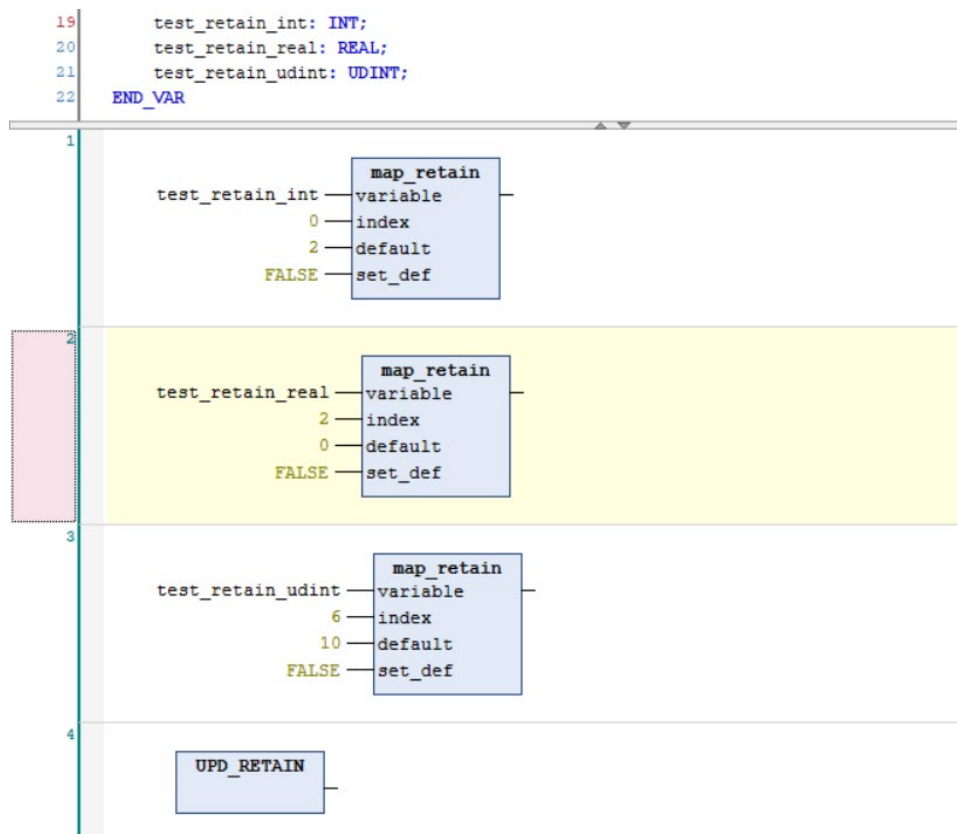


Рис.22. Пример использования библиотеки K15_LX1_retain

Параметры функции MAP_RETAIN:

variable – переменная, которой необходимо присвоить тип retain/persistent,

index – смещение в памяти ПЛК, по которому хранится значение переменной,

default – значение переменной по умолчанию,

set_def – устанавливает значение переменной равным значению по умолчанию. TRUE – установить значение.

Параметры функции UPDATE_RETAIN:

Функция не имеет параметров. Необходимо разместить ее вызов в одном из циклически исполняемых программных модулей.

Внимание! Переменные, привязанные с помощью функции MAP_RETAIN, являются PERSISTENT переменными, поэтому чтобы сбросить их на начальные значения, используйте параметры default и set_def.