

А.С. Гирник, А.Л. Федянин

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ НЕФТЯНОЙ ЗАДВИЖКОЙ



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

А.С. Гирник, А.Л. Федянин

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ НЕФТЯНОЙ ЗАДВИЖКОЙ

Методические указания к выполнению лабораторной работы

Издательство
Томского политехнического университета
2024

Гирник А.С.

Г51 Автоматизированная система управления нефтяной задвижкой : методические указания к лабораторным работам / А.С. Гирник, А.Л. Федянин ; Томский политехнический университет. – Томск : Изд-во Томского политехнического университета, 2024. – 99 с.

В методических указаниях изложена последовательность проектирования автоматизированной системы управления нефтяной задвижкой в составе модульного контроллера, человеко-машинного интерфейса на борту сенсорной панели и SCADA системы.

Предназначено для студентов и других сторонних слушателей, обучающихся по направлению 13.03.02 «Электроэнергетика и электротехника».

© ФГАОУ ВО НИ ТПУ, 2024

© Гирник А.С., 2024

© Оформление. Издательство Томского политехнического университета, 2024

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	3
ВВЕДЕНИЕ.....	4
1. ОПИСАНИЕ УЧЕБНОГО ОСНАЩЕНИЯ.....	5
2. ЛАБОРАТОРНЫЙ ПРАКТИКУМ	21
2.1. Изучение лабораторной установки «Нефтяная задвижка»	21
2.2. Общая концепция системы управления в составе OPC сервера.....	24
2.3. Создание программы на языках программирования CFC и FBD в среде CODESYS для модульного контроллера XC-CPU201	27
2.4. Создание программы на языке программирования ST в среде CODESYS для контроллера HMI панели визуализации XV102	44
2.5. Создание графического интерфейса управления в среде GALILEO для HMI панели визуализации XV102	49
2.6. Создание графического SCADA интерфейса управления в среде TRACE MODE	57
2.7. Настройка OPC сервера.....	82
2.8. Аprobация созданной системы управления.....	83
3. УСТАНОВКА ПОГРАММНОГО ОБЕСПЕЧЕНИЯ	89
3.1. Установка и настройка CODESYS 2.3.9 SP8	89
3.2. Установка Trace Mode 6.09.....	91
3.3. Установка и лицензирование Galileo 8.1	91
3.4. Обновление прошивки HMI панели XV102.....	93
СПИСОК ЛИТЕРАТУРЫ.....	99

ВВЕДЕНИЕ

В настоящее время современные технологии позволяют создавать системы управления на базе таких микропроцессорных устройств, с помощью которых можно автоматизировать практически любой технологический процесс [1]. Ранее алгоритм управления реализовывался внутри больших релейно-контактных шкафов. Сегодня она может размещаться внутри компактной электронной памяти программируемой аппаратуры. Такие средства автоматизации не только повышают эффективность производства, но также освобождают человека от выполнения работы по контролю за состоянием технологического процесса и формированию управляющих воздействий на исполнительные органы рабочих механизмов. Применение программируемых логических контроллеров и реле в системах управления производственных объектов способствует автоматизации технологических процессов. В настоящее время данные электронные аппараты относятся к числу обладателей искусственного интеллекта.

Современные автоматизированные системы управления разделяются на три уровня:

1. Верхний уровень – управление технологическим процессом с помощью виртуальной панели оператора, которая представляет собой SCADA-программу, установленную на персональном компьютере.
2. Средний уровень – программно-электрическая часть, реализуемая с помощью программируемых логических контроллеров или реле, имеющих информационные цифровые или аналоговые входы и выходы.
3. Нижний (полевой) уровень – физические модели, исполнительные органы, управляемые элементами среднего уровня.

В совокупности все три уровня автоматизации представляют собой интеллектуальную систему управления технологическими процессами. В данном учебном пособии рассматривается реализация нижнего и среднего уровней на базе программного обеспечения и оборудования производства Eaton (Moeller). Также уделено внимание языкам программирования стандарта МЭК, на которых создаются программные алгоритмы управления такими объектами, как ленточный конвейер, автоматическая дверь, сообщающиеся сосуды и нефтяная задвижка. Эти объекты могут рассматриваться как отдельные элементы автоматизации.

В учебном пособии рассматриваются программные продукты, имеющиеся в распоряжении Инженерной школы энергетики НИ ТПУ, EasySoft и CodeSys.

1. ОПИСАНИЕ УЧЕБНОГО ОСНАЩЕНИЯ

Все лабораторные работы, приведённые в данном учебном пособии, проводятся на специализированном оборудовании, которым оснащён учебный центр ТПУ «Технические средства автоматизации» (рис. 1):

- пять универсальных учебных стендов с контроллерами;
- лабораторная установка ленточного конвейера;
- лабораторная установка автоматической двери;
- лабораторная установка тепло-станции (сообщающихся сосудов);
- лабораторная установка нефтяной задвижки.



*Рис. 1. Учебный центр
«Технические средства автоматизации»*

К каждому универсальному стенду (рис. 2) специальными кабелями подключена своя отдельная лабораторная установка (физическая модель).

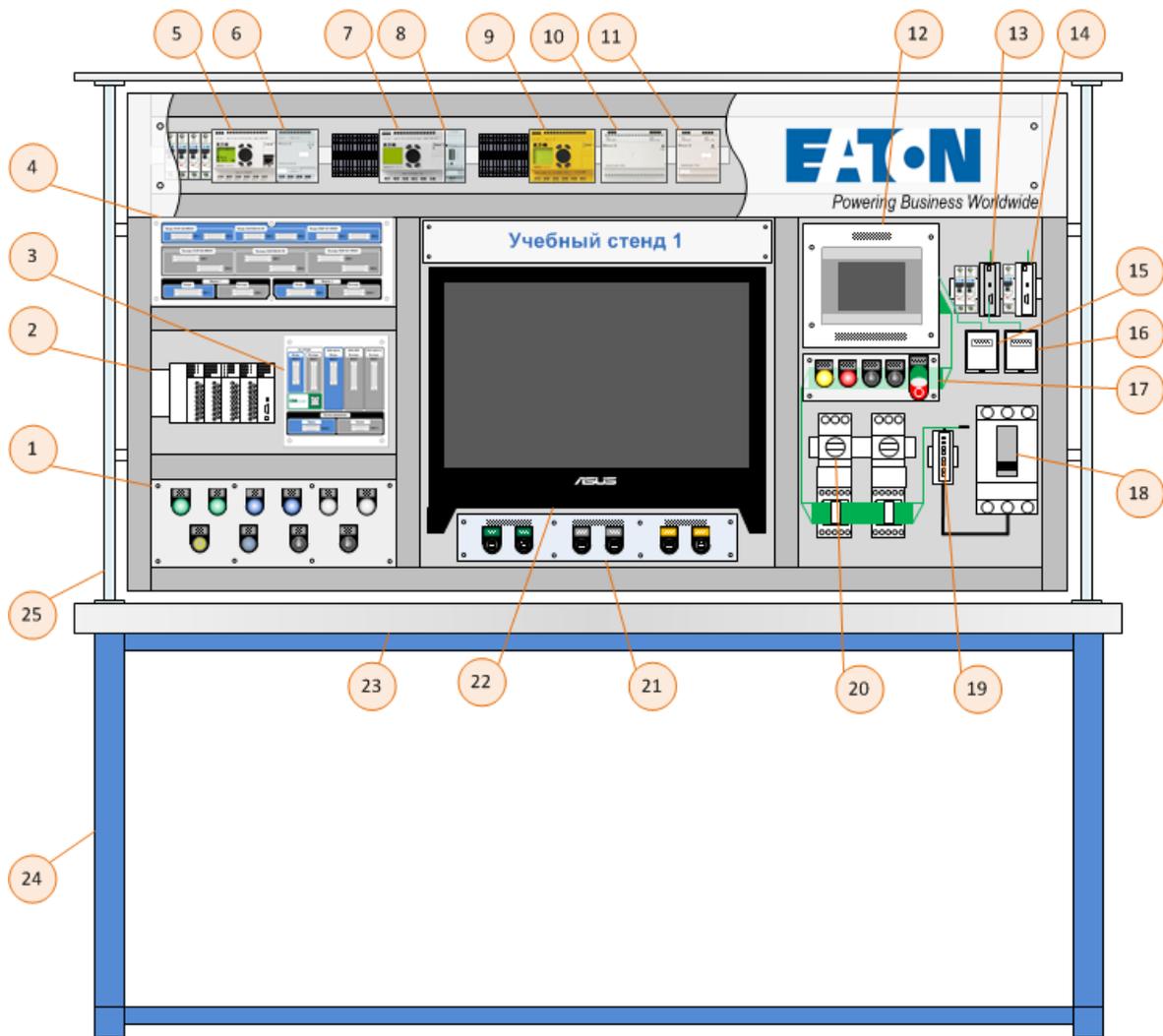


Рис. 2. Структура универсального учебного стенда:

- 1 – кнопочный пост; 2 – модульный ПЛК XC-201 с набором модулей;
 3, 4 – интерфейсные панели разъёмов; 5 – ПЛК EC4P-222-MRAD1; 6 – модуль расширения EASY411-DC-ME; 7 – программируемое реле EASY820-DC-RC;
 8 – модуль Ethernet EASY209-SE; 9 – реле безопасности Easy-Safety ES4P-221-DRXD1;
 10 – блок питания EASY600 POW; 11 – блок питания EASY400 POW; 12 – HMI-панель XV-102-E6-57TVRC-10; 13 – концентратор EU5C-SWD-CAN; 14 – концентратор EASY800 with SmartWire-DT EASY806-DC-SWD; 15 – розетка подключения к CANopen HMI-панели; 16 – розетка подключения к CANopen концентратора EU5C-SWD-CAN; 17 – кнопочный пост; 18 – автомат NZMN2-ME90; 19 – SWD-модуль NZM-XSWD-704 соединения для NZM; 20 – пусковая сборка MSC-DEA-12-M7;
 21 – панель сетевых коммуникаций; 22 – персональный компьютер;
 23 – стол; 24 – стойки; 25 – система подвески оборудования

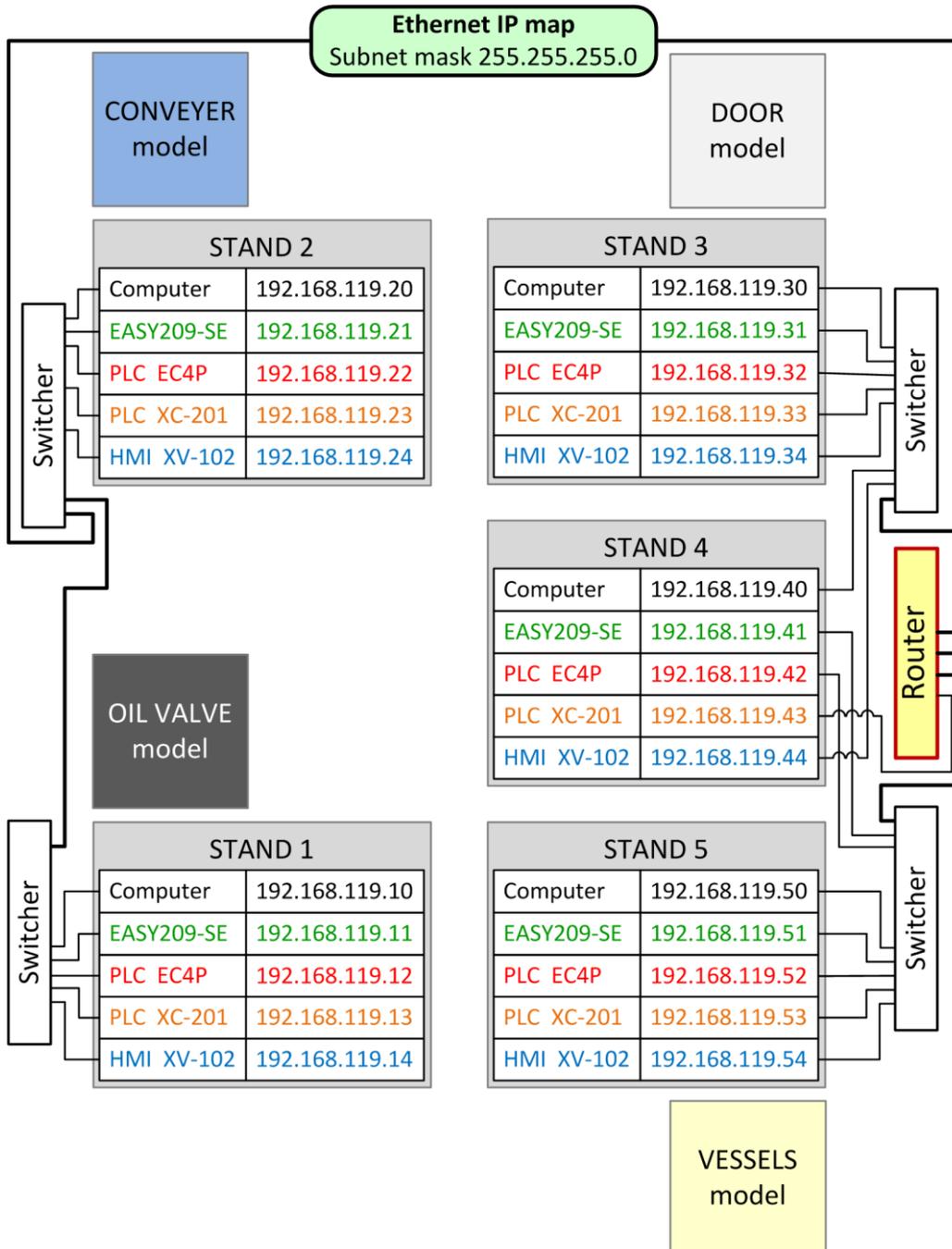


Рис. 3. Подключение всего учебного оборудования в общую сеть Ethernet

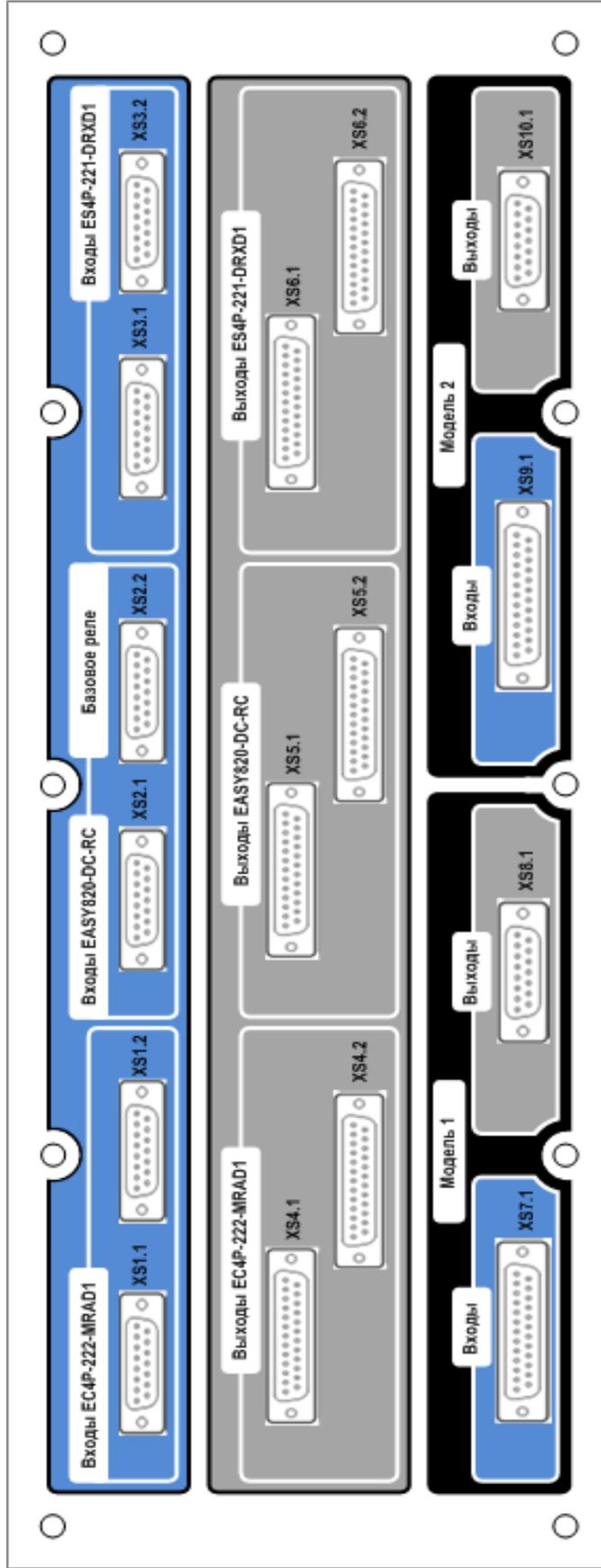


Рис. 4. Первая интерфейсная панель разъемов

На рис. 4, 5 показаны интерфейсные панели разъёмов для электрического соединения:

- 1) кнопок управления с входами аппаратов (реле, контроллеры);
- 2) сигнальных ламп с выходами аппаратов;
- 3) датчиков физических моделей с входами аппаратов;
- 4) исполнительных элементов физических моделей (контакторы пуска двигателя, соленоиды и т. д.) с выходами аппаратов.

Первая панель разъёмов (рис. 4) разделена на 3 основных секции:

1. *Синяя секция.* В ней расположены электрические разъёмы, которые соединены со входами программируемых реле и контроллеров. Причём для каждого аппарата предусмотрена пара дублированных разъёмов. То есть на каждом из двух разъёмов контакты с одинаковыми номерами соединены параллельно друг другу. Это необходимо для того, чтобы можно было подключать ко входам контроллера как кабель для передачи сигналов от датчиков лабораторной установки, так и кабель для передачи сигналов управления от кнопок на универсальном стенде.

2. *Серая секция.* В ней расположены электрические разъёмы, которые соединены с выходами программируемых аппаратов. Здесь точно так же предусмотрено парное дублирование разъёмов, для того чтобы можно было подавать управляющие сигналы от контроллеров как на лабораторную установку, так и на сигнальные лампы универсального стенда.

3. *Чёрная секция.* В ней расположены электрические разъёмы, через которые можно подключаться к датчикам лабораторной установки, а также к её исполнительным элементам.

Вторая панель разъёмов (рис. 5) предназначена для подключения к модульному контроллеру ХС-201 и его отдельным модулям. Эта панель также условно разделена на секции: синие (для входных сигналов), серые (для выходных сигналов) и черные (для ручного управления от кнопок на универсальном стенде).

На рис. 6–8 показаны электрические схемы соединения первой панели разъёмов со входами и выходами таких программируемых электронных аппаратов, как:

- программируемый логический контроллер ЕС4Р-222-MRAD1;
- программируемое реле EASY820-DC-RC;
- реле безопасности ES4P-221-DRXD1.

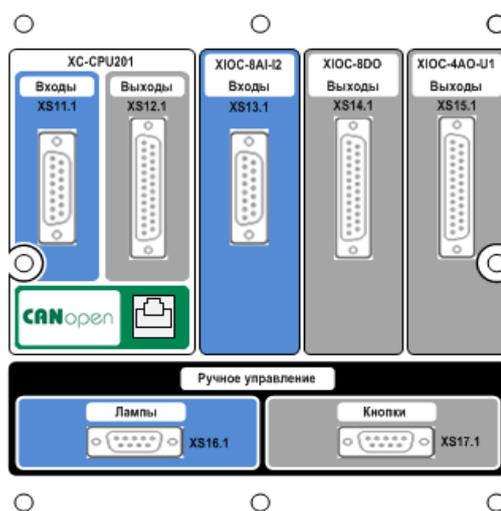


Рис. 5. Вторая интерфейсная панель разъёмов

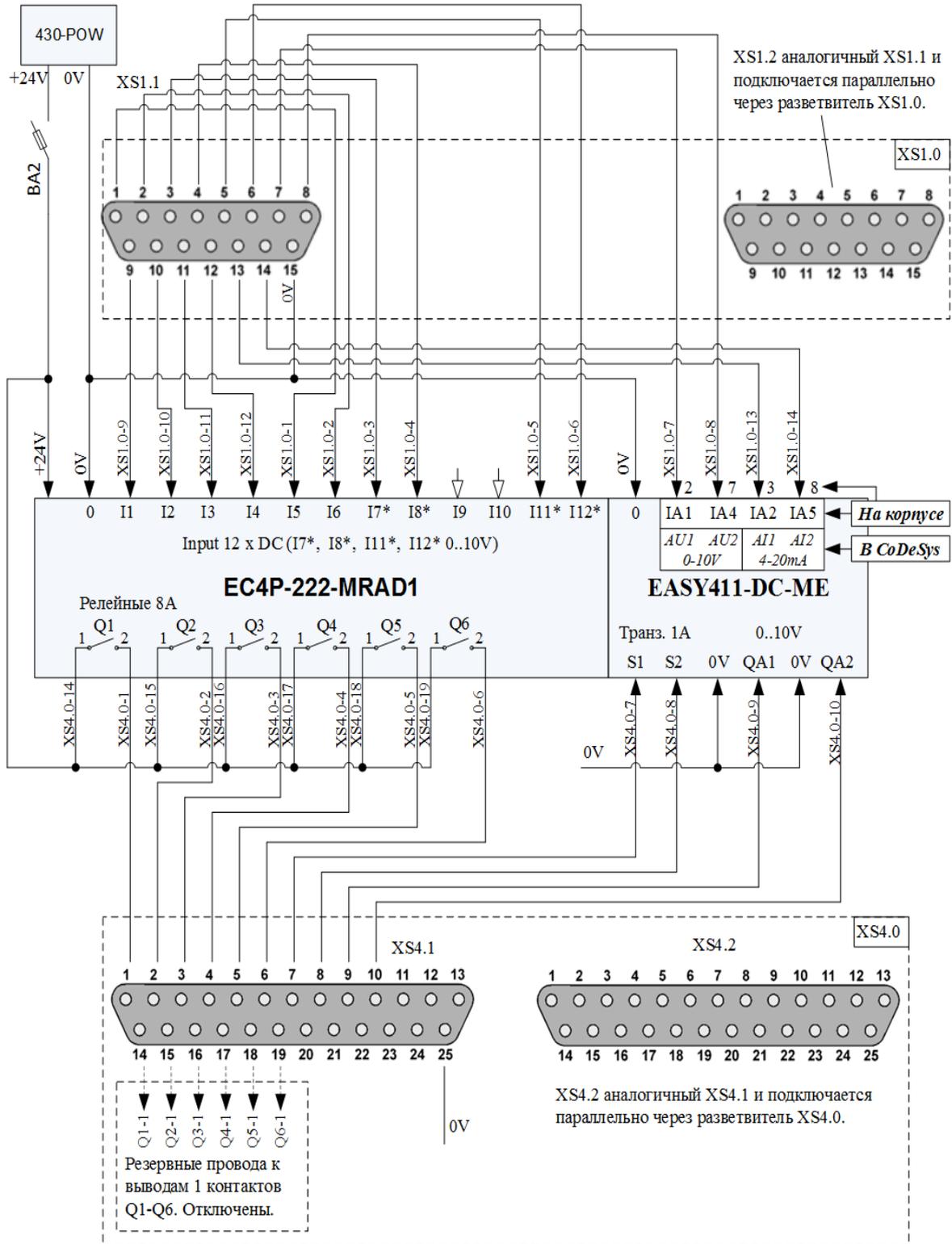


Рис. 6. Подключение контроллера EC4P-222-MRAD1 к первой панели разъемов

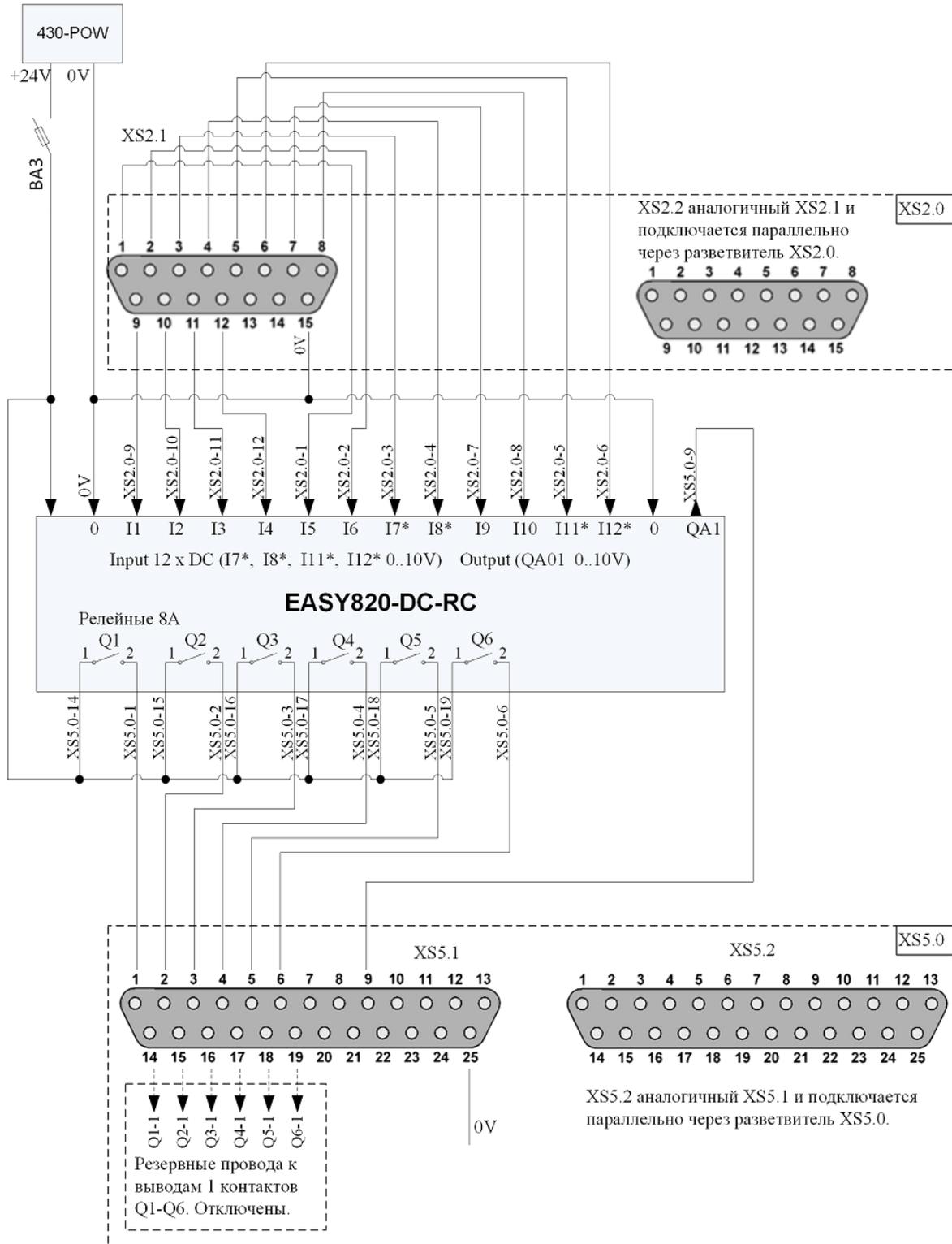


Рис. 7. Подключение реле EASY820-DC-RC к первой панели разъемов

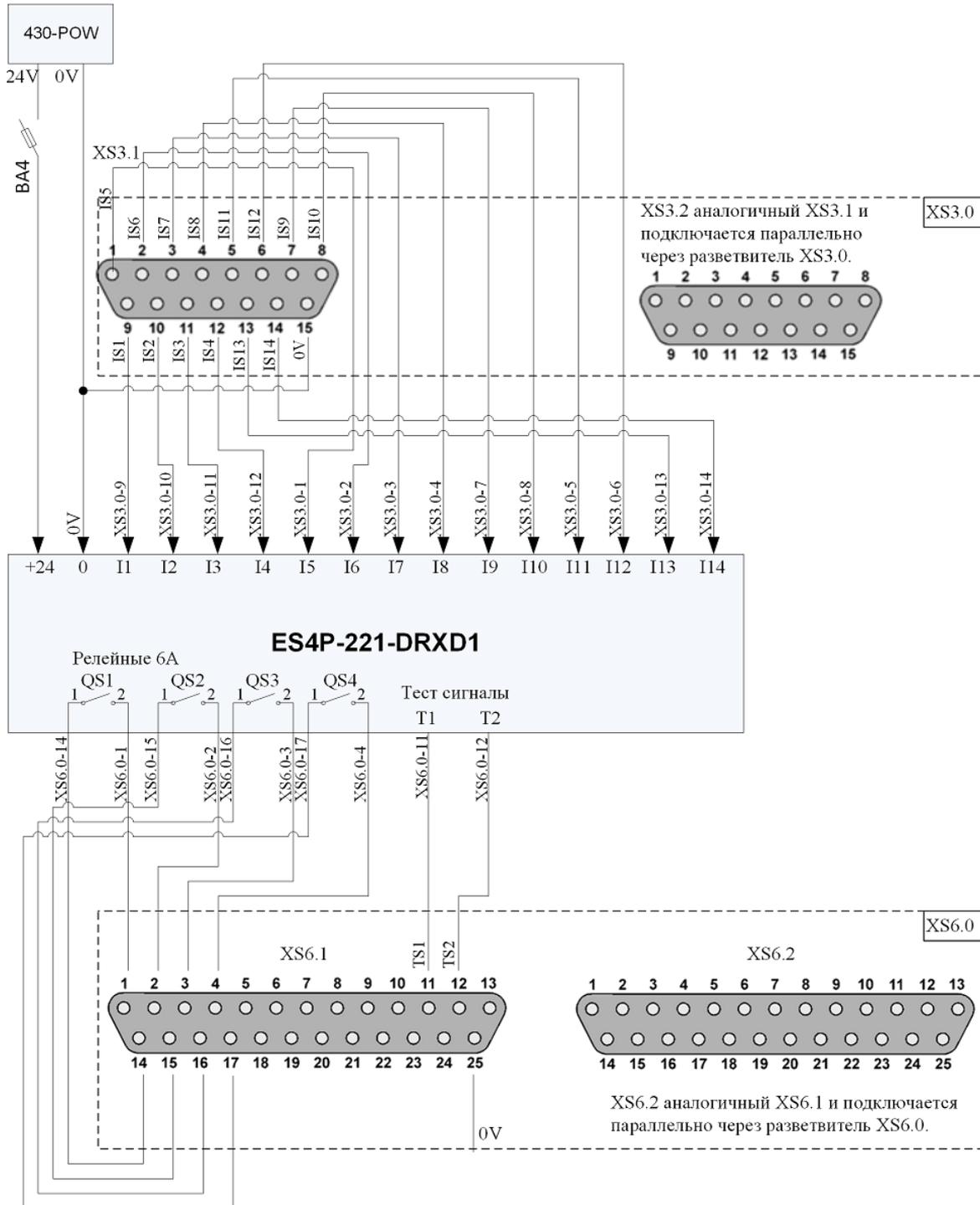


Рис.8. Подключение реле ES4P-221-DRXD1 к первой панели разъёмов

На рис. 9, 10 показаны электрические схемы соединения первой панели разъёмов с согласующими отдельными разъёмами (расположенными на задней стороне универсального стенда) для подключения к лабораторной установке.

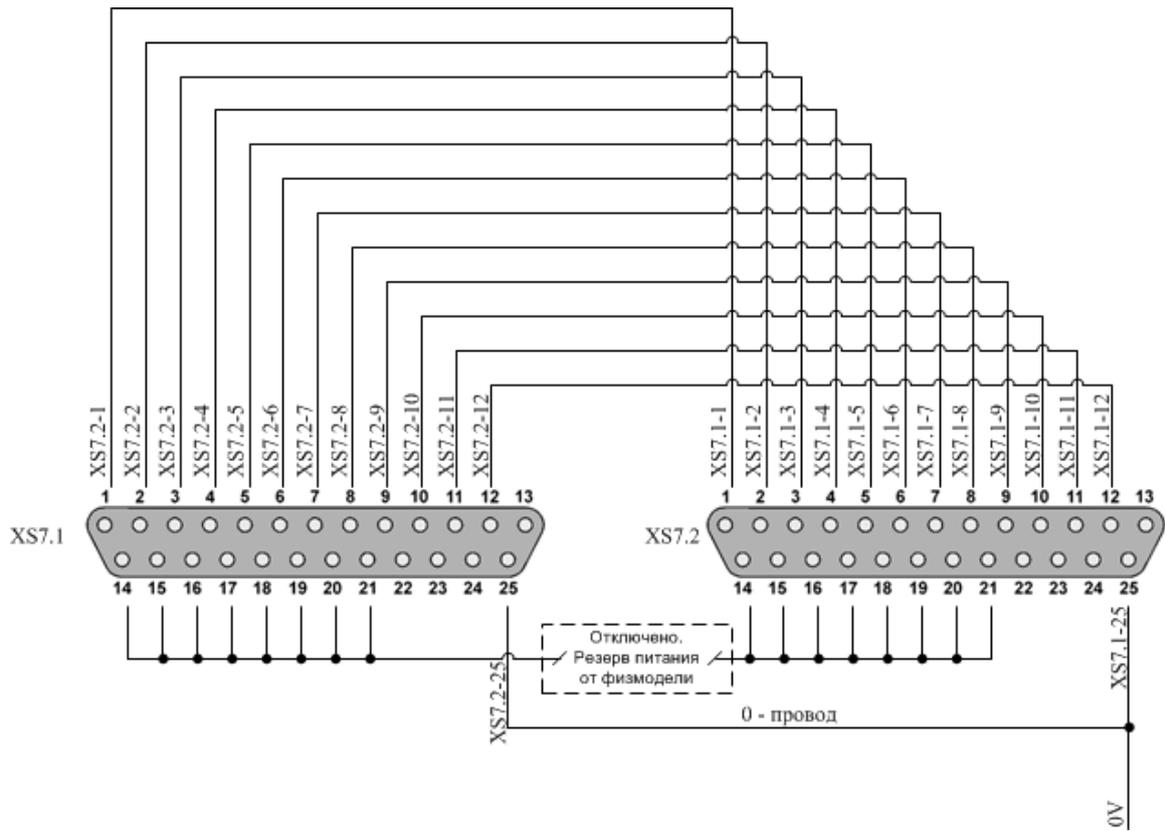


Рис. 9. Входы лабораторной установки

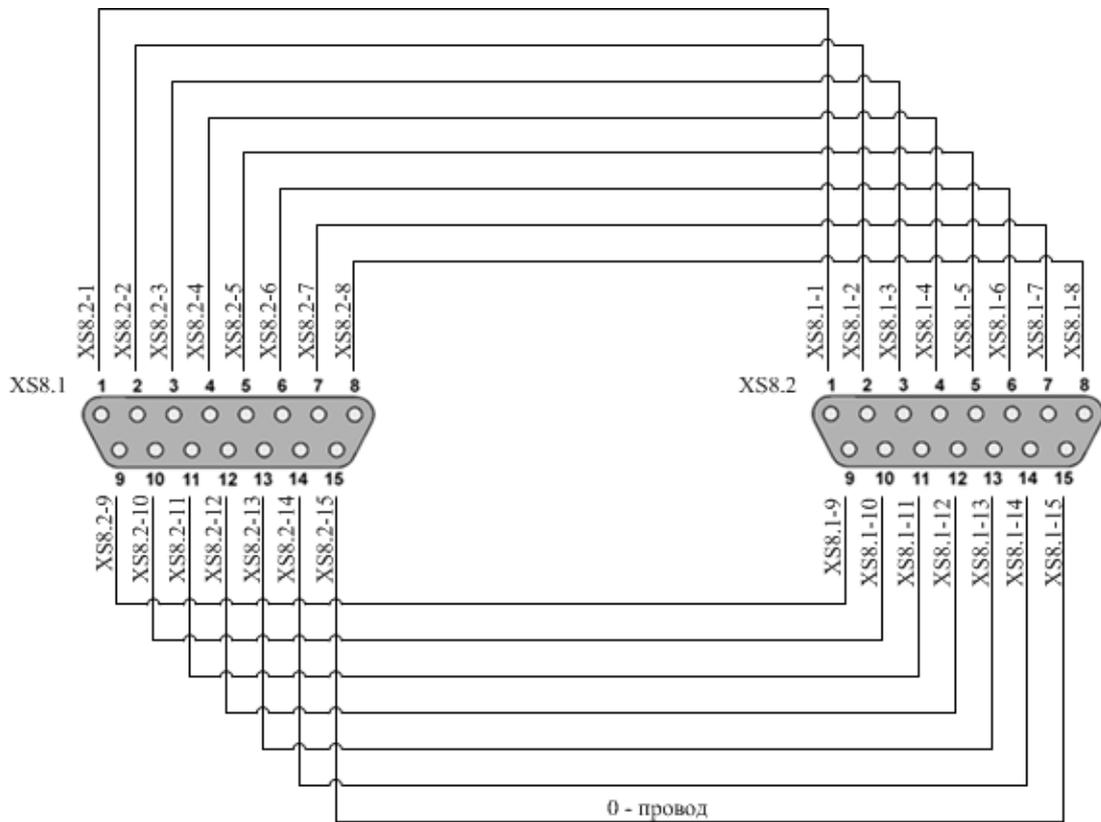


Рис. 10. Выходы лабораторной установки

На рис. 11, 12 показаны электрические схемы соединения второй панели разъёмов со входами и выходами модулей контроллера ХС-СРU201.

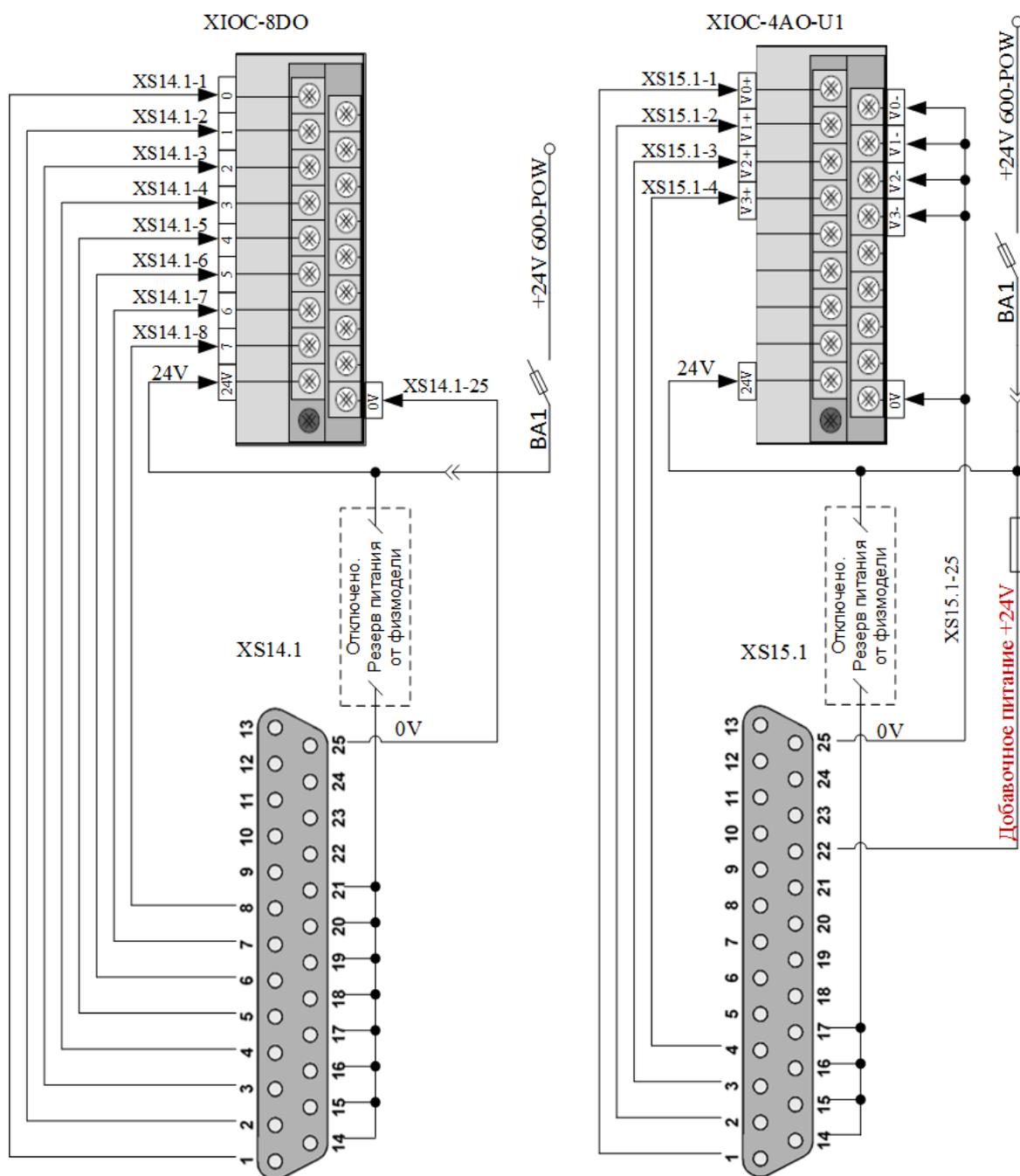


Рис. 11. Подключение выходов модулей контроллера ХС-СРU201 ко второй панели разъёмов

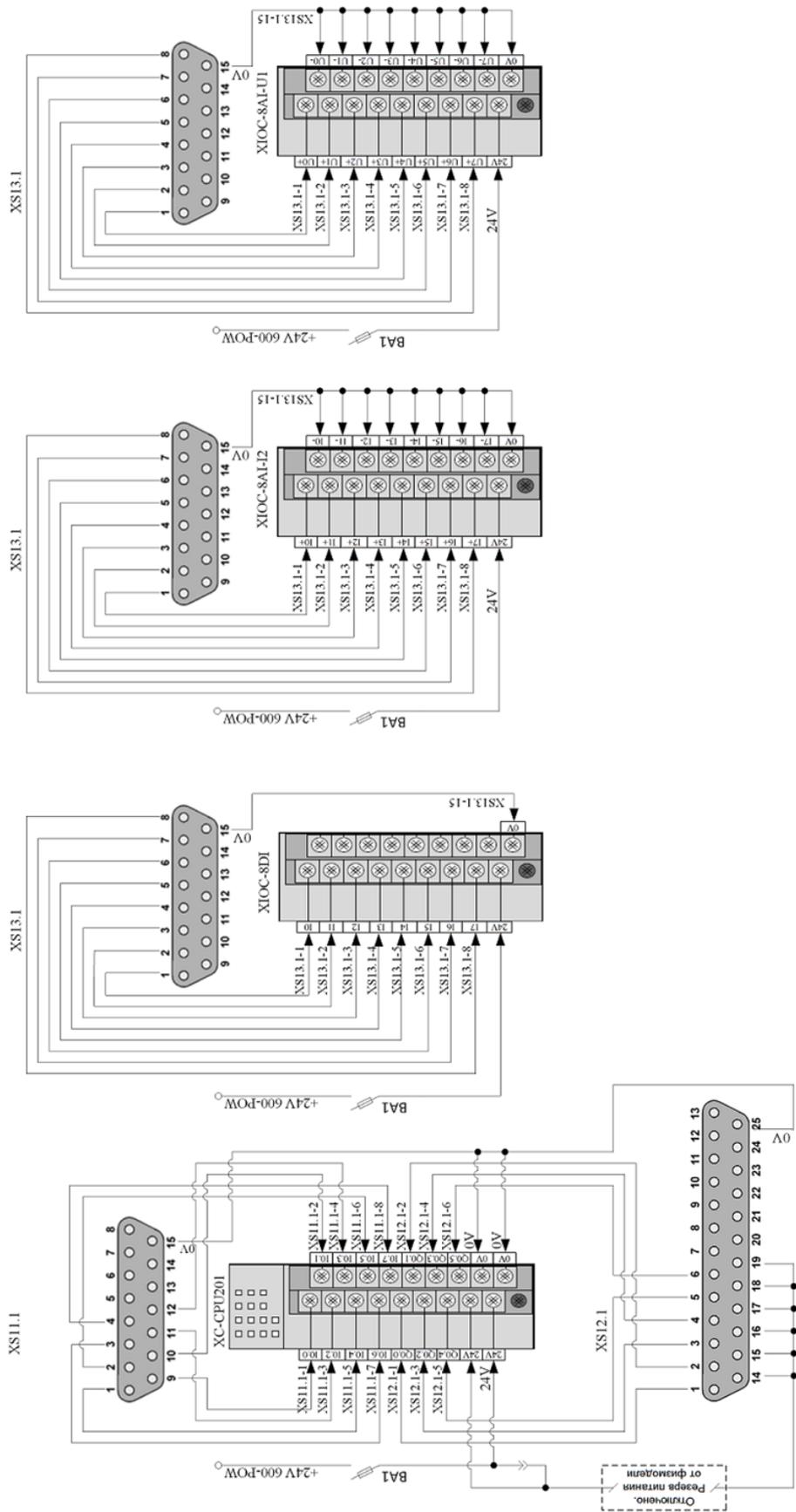


Рис. 12. Подключение модулей контроллера XC-CPU201 ко второй панели разъёмов

На рис. 13 показаны электрические схемы подключения кнопок управления и сигнальных ламп ко второй панели разъёмов.

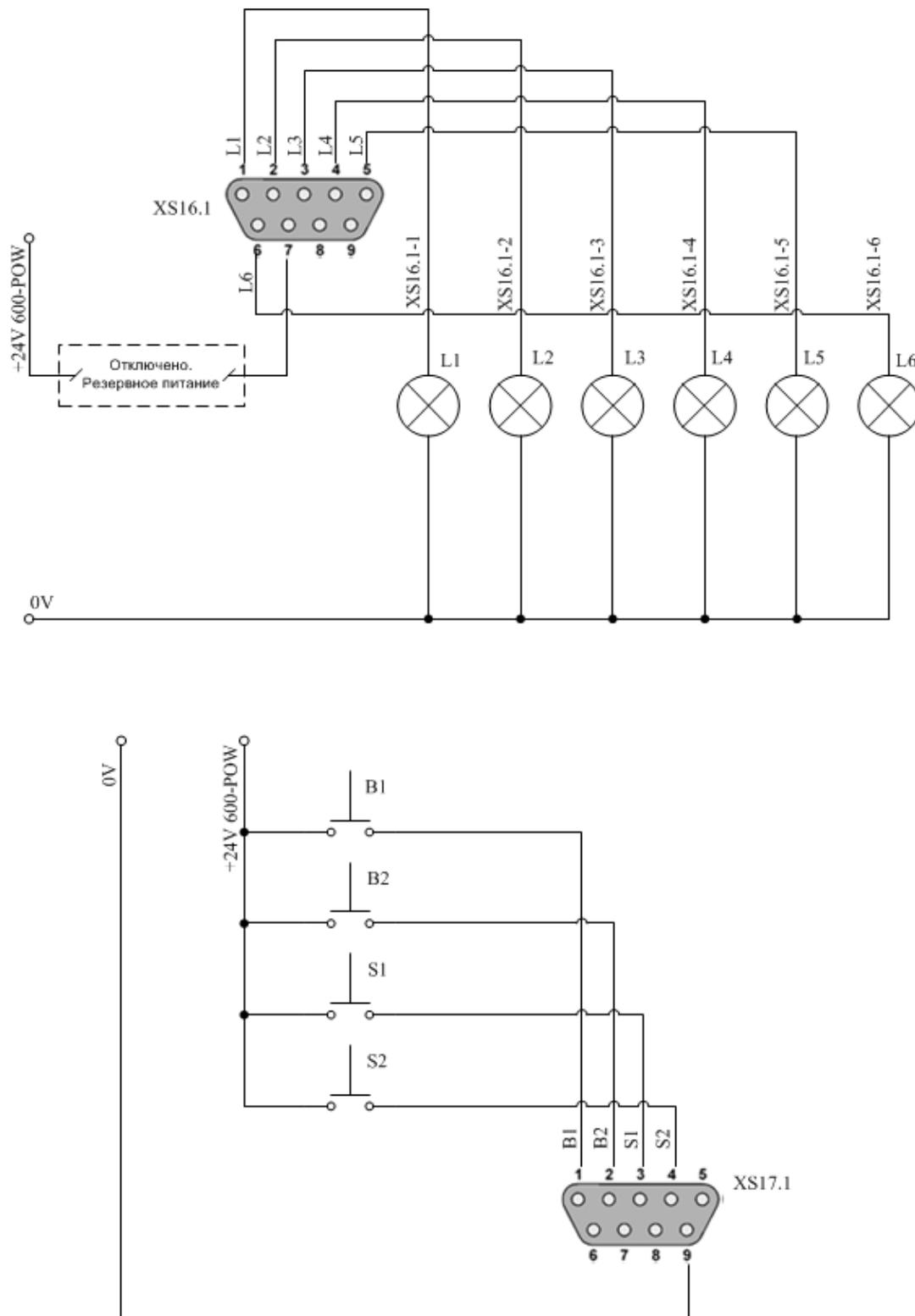
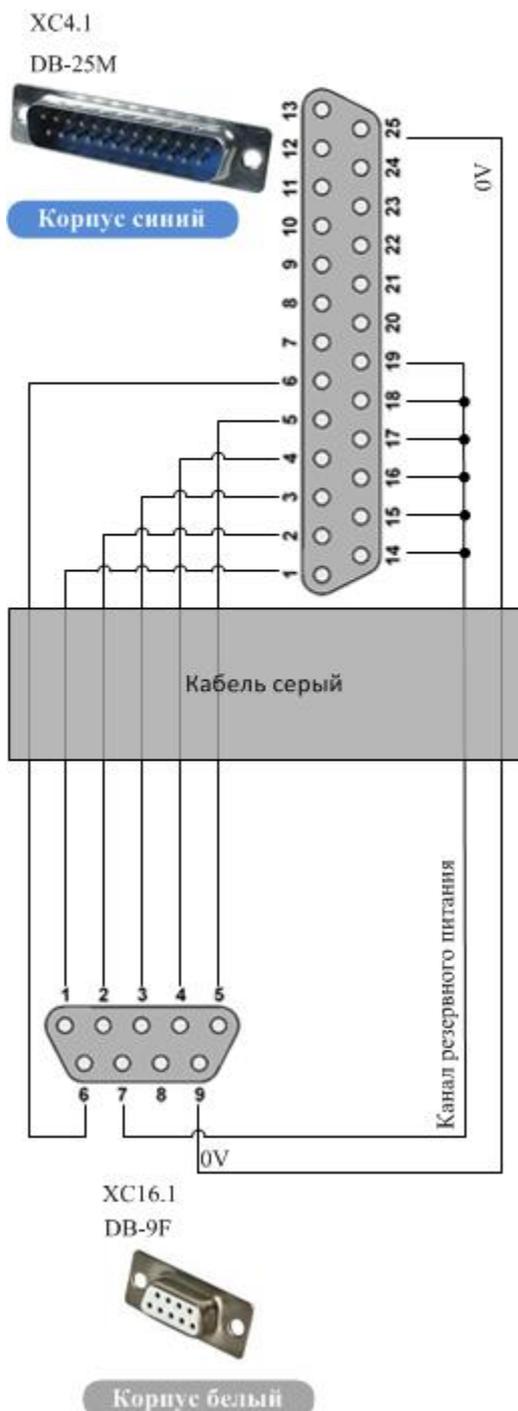


Рис. 13. Подключение элементов ручного управления ко второй панели разъёмов

На рис. 14–16 показаны электрические соединения внутри универсальных кабелей, с помощью которых можно подключать входы и выходы контроллеров к лабораторной установке через первую и вторую панели разъёмов.

Кабель «Контроллер / лампы»



Кабель «Кнопки / контроллер»

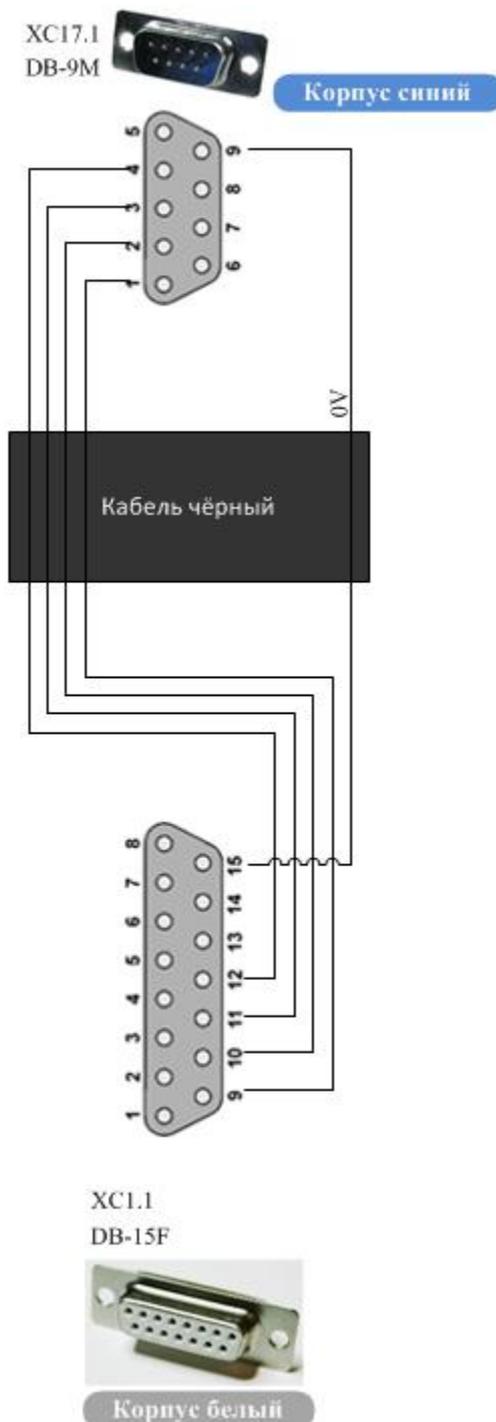


Рис. 14. Кабели ручного управления

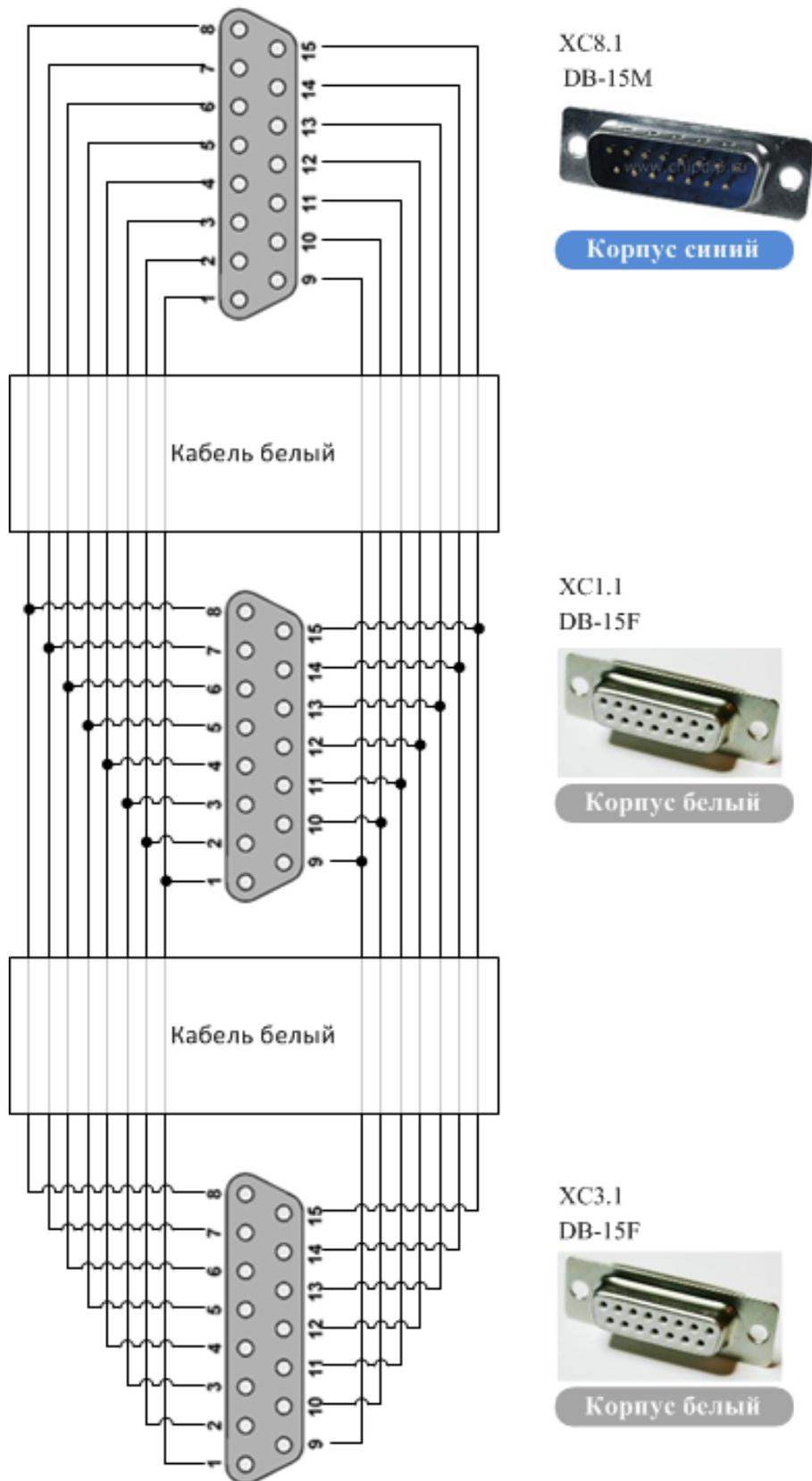


Рис. 15. Кабель подключения лабораторной установки к входам контроллера

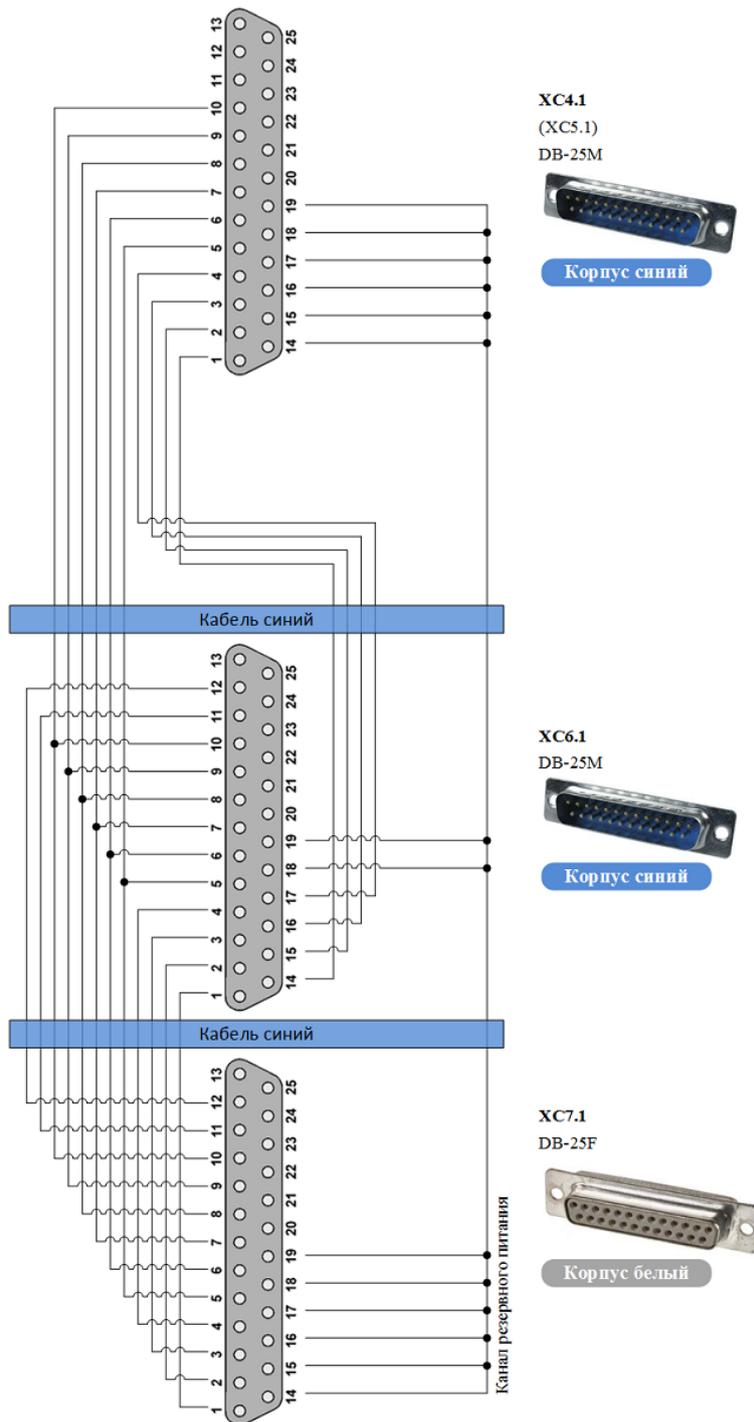


Рис. 16. Кабель подключения лабораторной установки к выходам контроллера

На рис. 17 показаны варианты подключения любого из контроллеров к лабораторной установке: напрямую (А) и через реле безопасности (В). На рис. 18 показана электрическая схема разветвителя, который предназначен для распределения сигналов датчиков от нефтяной задвижки ко входам модульного контроллера ХС-СРУ201.

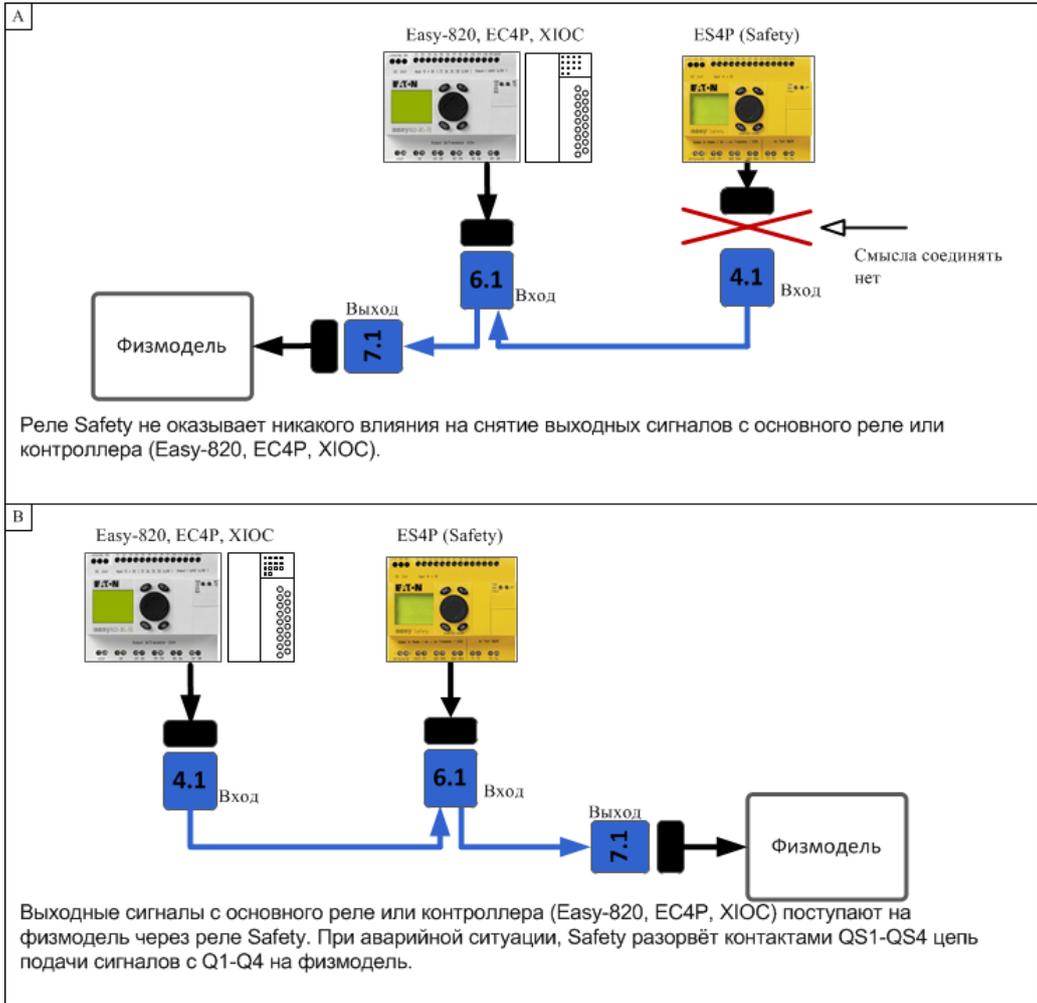


Рис. 17. Варианты подключения лабораторной установки к выходам разных контроллеров

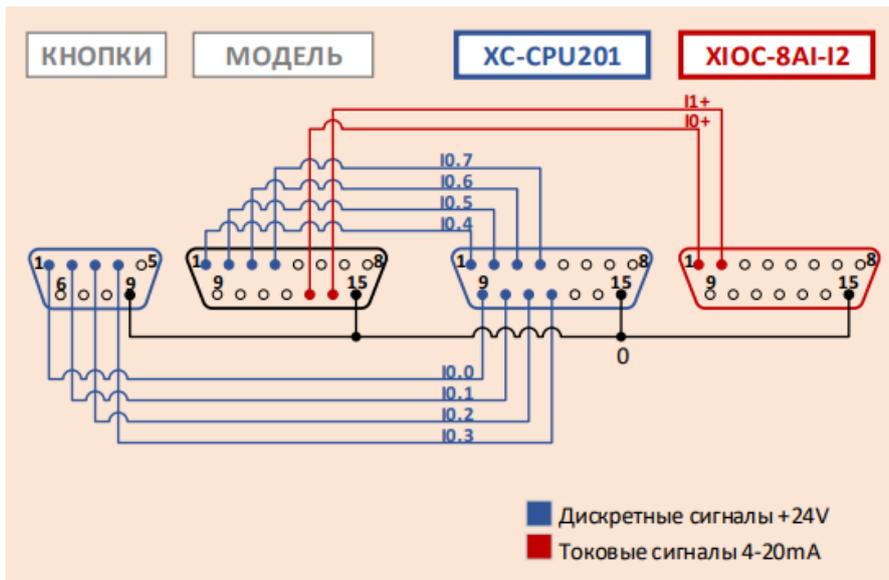


Рис. 18. Разветвитель сигналов от нефтяной задвижки к модульному контроллеру

2. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Цель: научиться создавать программы для автоматизированной системы управления нефтяной задвижкой на разных языках программирования в средах разработки CODESYS, TRACEMODE и GALILEO.

Порядок выполнения работы

2.1. Изучение лабораторной установки «Нефтяная задвижка»

На рис. 19 приведена функциональная схема лабораторной установки с нефтяной задвижкой. Установка представляет собой замкнутую систему регулирования давления в трубопроводе. В качестве активной среды для упрощения эксплуатации установки используется поток воздуха, создаваемый вентилятором. Поток воздуха направляется в трубопровод, в котором расположен датчик (измеритель) давления, преобразующий давление в электрический сигнал 4...20 мА, подающийся на программируемый логический контроллер с программой задания степени открытия заслонки задвижки. Сама заслонка (клапан) нефтяной задвижки располагается непосредственно в трубопроводе. Для создания нагрузки, т. е. для увеличения давления в трубопроводе путем внешнего воздействия, используется нагрузочный клапан, приводимый в движение вручную.

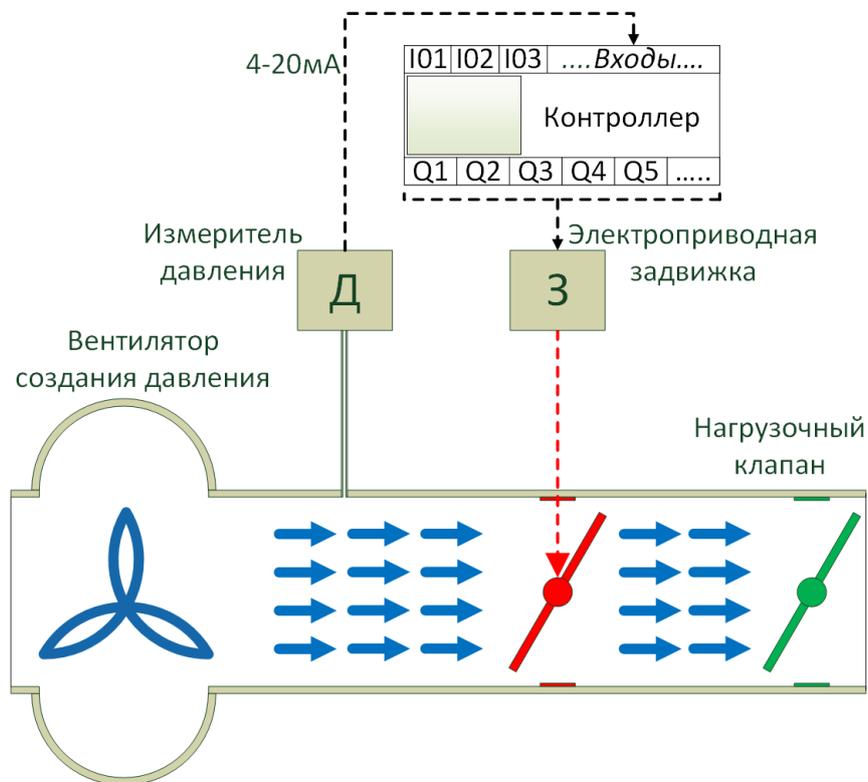


Рис. 19. Функциональная схема лабораторной установки

Таким образом, замкнутый процесс регулирования давления сводится к следующему: при закрытии нагрузочного клапана, в первые секунды давление в трубопроводе повысится, датчик давления зафиксирует это повышение давления и начнёт подавать на аналоговый вход программируемого контроллера увеличенный сигнал управления для дополнительного открытия заслонки с целью вернуть давление потока в трубопроводе к прежнему значению, т. е. скомпенсировать его повышение. В первые секунды из-за гидравлической и механической инерционности всей системы задвижка не сразу приоткроет свою заслонку, это будет происходить некоторое время (несколько секунд), за которое можно успеть зафиксировать изменение давления и проследить за самим переходным процессом. Внешний вид лабораторной установки «Нефтяная задвижка» показан на рис. 20.

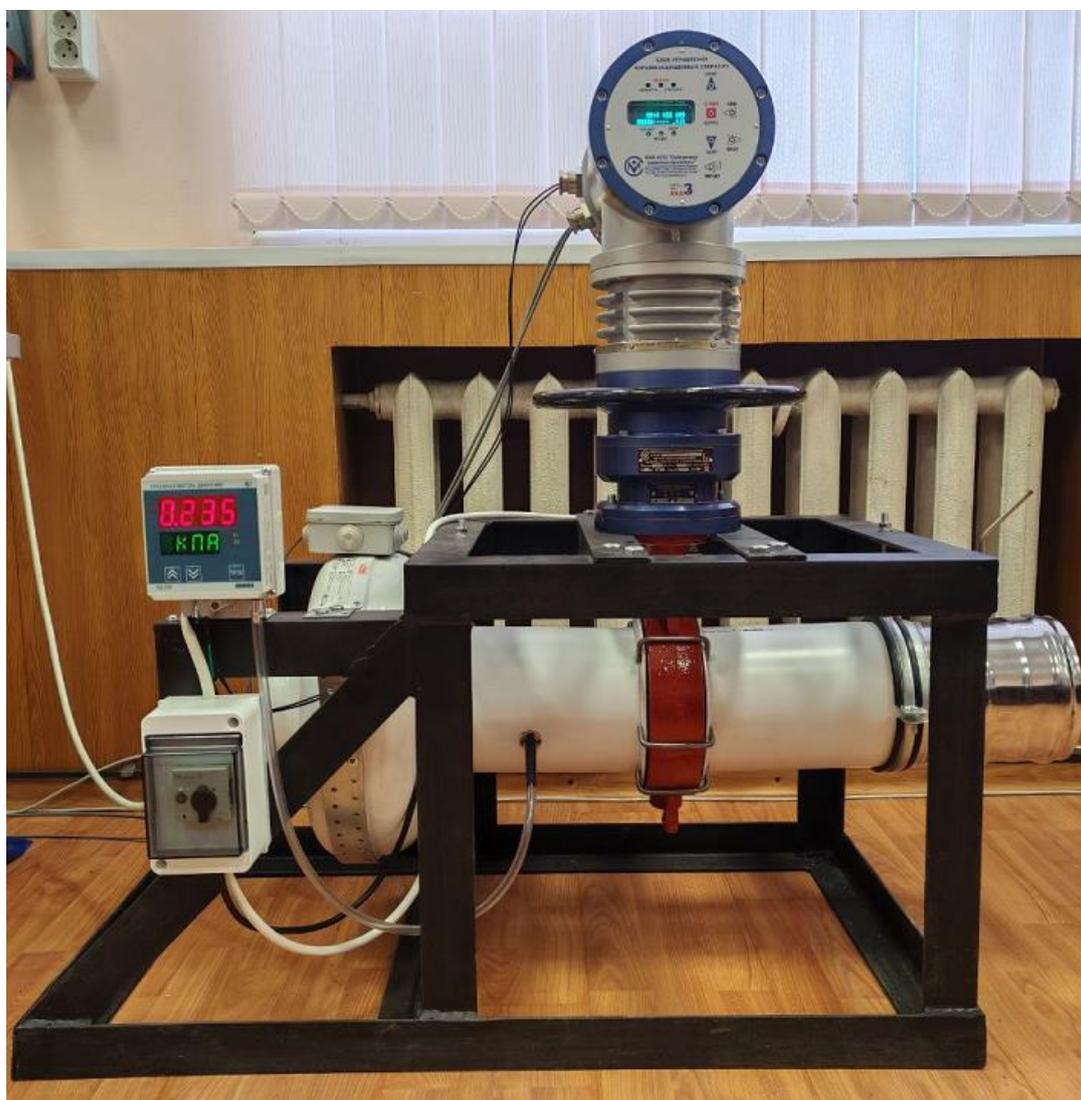


Рис. 20. Лабораторная установка «Нефтяная задвижка»

В табл. 1 и 2 приведены номера контактов подключения к лабораторной установке через разъёмы DB-15 и DB-25 входов и выходов контроллера ХС-CPU201.

Таблица 1

Подключение ко входам контроллера ХС-CPU201

Назначение	Контакт на разъёме DB-15	Поступление на входы	
		ХС-CPU201	ХИОС-8АI-I2
Дискретный сигнал 24 В об открытом состоянии заслонки от задвижки	1	I0.4	–
Дискретный сигнал 24 В о закрытом состоянии заслонки от задвижки	2	I0.5	–
Дискретный сигнал 24 В о наличии аварии от задвижки	3	I0.6	–
Дискретный сигнал 24 В о местном режиме управления от задвижки	4	I0.7	–
Сигнал на открытие заслонки с кнопки управления на стенде	9	I0.0	–
Сигнал на закрытие заслонки с кнопки управления на стенде	10	I0.1	–
Сигнал на остановку движения заслонки с кнопки управления на стенде	11	I0.2	–
Сигнал на включение автоматического режима с кнопки управления на стенде	12	I0.3	–
Токовый сигнал 4...20 мА о текущем давлении от датчика давления	13	–	I0+
Токовый сигнал 4...20 мА о текущем положении заслонки от задвижки	14	–	I1+
<i>Общий провод</i>	15	0	0

Таблица 2

Подключение к выходам контроллера ХС-CPU201

Назначение	Контакт на разъёме DB-25	Выход ХС-CPU201
Дискретный сигнал 24 В на открытие заслонки	1	Q0.0
Дискретный сигнал 24 В на закрытие заслонки	2	Q0.1
Дискретный сигнал 24 В на останов заслонки	3	Q0.2
<i>Общий провод</i>	25	0

2.2. Общая концепция системы управления в составе OPC сервера

Общий принцип работы подобной системы управления состоит в следующем:

- Передача сигналов управления (пакетов данных) осуществляются со стороны SCADA системы на персональном компьютере в сторону OPC сервера для записи значения управляющего сигнала в символьный файл. В нашем случае OPC сервер в виде специальной программы находится в составе самого компьютера. В отдельных случаях данные сигналы управления можно передавать дистанционно в OPC сервер по протоколу TCP/IP Ethernet через специальный кабель.
- Значение управляющего сигнала из символьного файла передаётся в программируемый логический контроллер для запуска в нём определённого программного алгоритма. В нашем случае контроллер подсоединён в общую информационную сеть Ethernet с компьютером.
- Электрический сигнал управления передаётся со стороны контроллера в сторону исполнительных органов лабораторной установки нефтяной задвижки.
- Передача сигналов с датчиков лабораторной установки нефтяной задвижки осуществляется в обратном порядке в SCADA систему.
- Дополнительным звеном управления в системе является НМІ панель, которая соединена с контроллером специальным кабелем для двусторонней передачи данных по сетевому протоколу CAN Open. В данном случае обмен данных производится только между контроллером и самой панелью. OPC сервер напрямую здесь не участвует. Но принятые контроллером сигналы управления от НМІ панели могут поступать далее в OPC сервер уже по протоколу TCP/IP Ethernet, попадая в символьный файл и далее в SCADA систему.

Общая концепция работы данной системы управления в виде структурной схемы приведена на рис. 21.

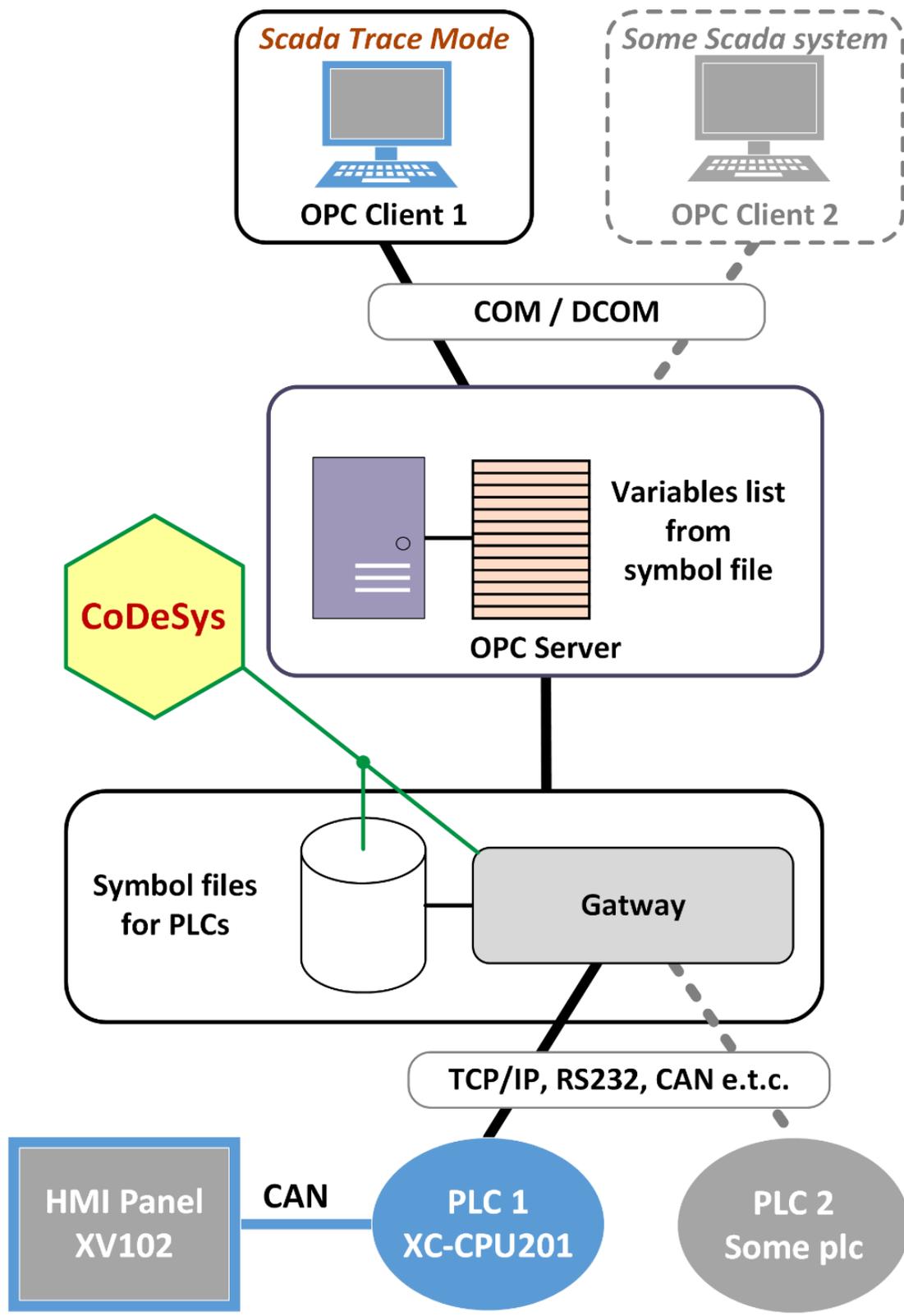


Рис. 21. Общая концепция системы управления

На рис. 22 показан внешний вид универсального учебного стенда, на борту которого находится такое необходимое для системы управления оборудование, как программируемый логический контроллер XC-CPU201, НМІ панель серии XV102, кнопки управления, персональный компьютер, панели разъёмов.



Рис. 22. Внешний вид учебного стенда

Всё программируемое оборудование данного стенда подключено в общую информационную сеть Ethernet по рис. 3. Это в первую очередь необходимо для загрузки программных алгоритмов во внутреннюю память оборудования со стороны персонального компьютера.

2.3. Создание программы на языках программирования CFC и FBD в среде CODESYS для модульного контроллера XC-CPU201

1. Запустите программную среду CODESYS.
2. Создайте новый проект: «Файл/Создать» (File/New).
3. Настройте целевую платформу (Target Settings). Откройте вкладку «Целевая платформа» (Target Settings) и на странице диалогового окна «Конфигурация» (Configuration) выберите тип контроллера **XC-CPU201-EC512K-8DI-6DO-XV V2.3.9 SP8**. Во вкладке основных настроек (General) должна стоять галочка напротив «Download symbol file». Во вкладке сетевой функциональности (Network functionality) поставьте галочку на против «Support network variables» и в открывшемся поле ввода пропишите «CAN» (рис. 23). Подтвердите ввод нажатием кнопки ОК.

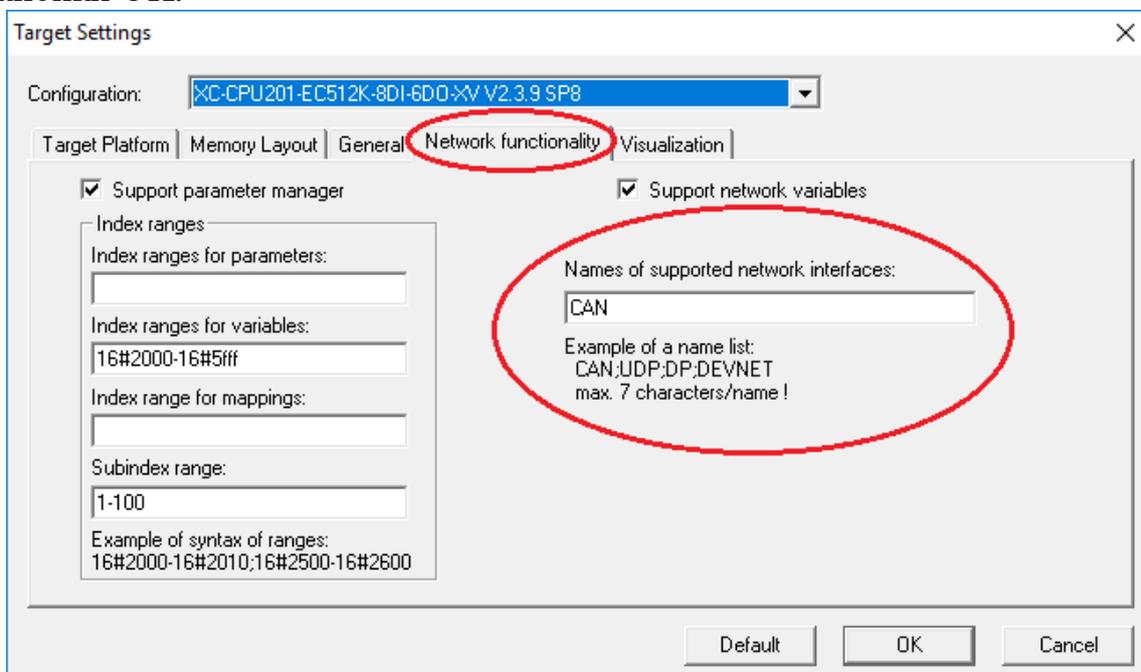


Рис. 23. Установка поддержки сетевой передачи данных через CAN Open протокол

4. Создайте главную программу. После настройки целевой платформы в автоматически появившемся диалоговом окне «Новый объект» (New POU) выберите тип компонента «Программа» (Program), язык CFC, задайте имя программы PLC_PRG.

5. Настройте конфигурацию контроллера. Зайдите во вкладку ресурсов «Resources» проекта и откройте конфигурацию контроллера «PLC Configuration» (рис. 24). Далее нажмите правой клавишей мыши по верхнему уровню меню (Configuration XC-CPU201...) и добавьте «CanMaster».

Нажмите внутри опции «Non Display[SLOT]» по «EMPTY-SLOT» правой клавишей мыши и выберите элемент «XIOC-8AI-I2» - это модуль расширения для приёма аналоговых токовых сигналов от датчика давления и датчика угла положения заслонки нефтяной задвижки.

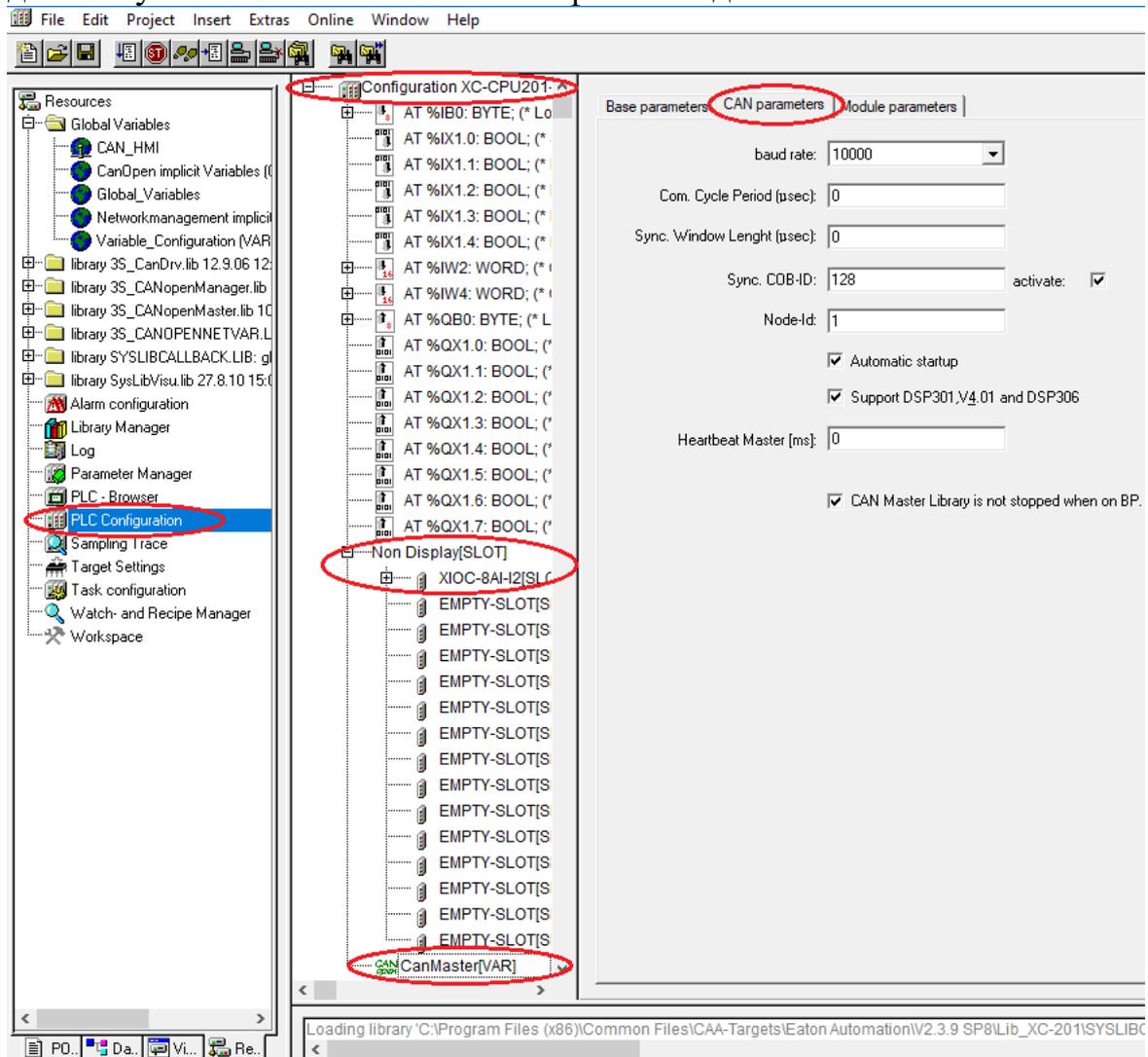


Рис. 24. Конфигурация контроллера

Общая структура системы на языке CFC показана на рис. 25 (главная программа PLC_PRG). Чтобы создать данную программу, необходимо сначала создать дополнительные функциональные блоки, а затем соединить их между собой.

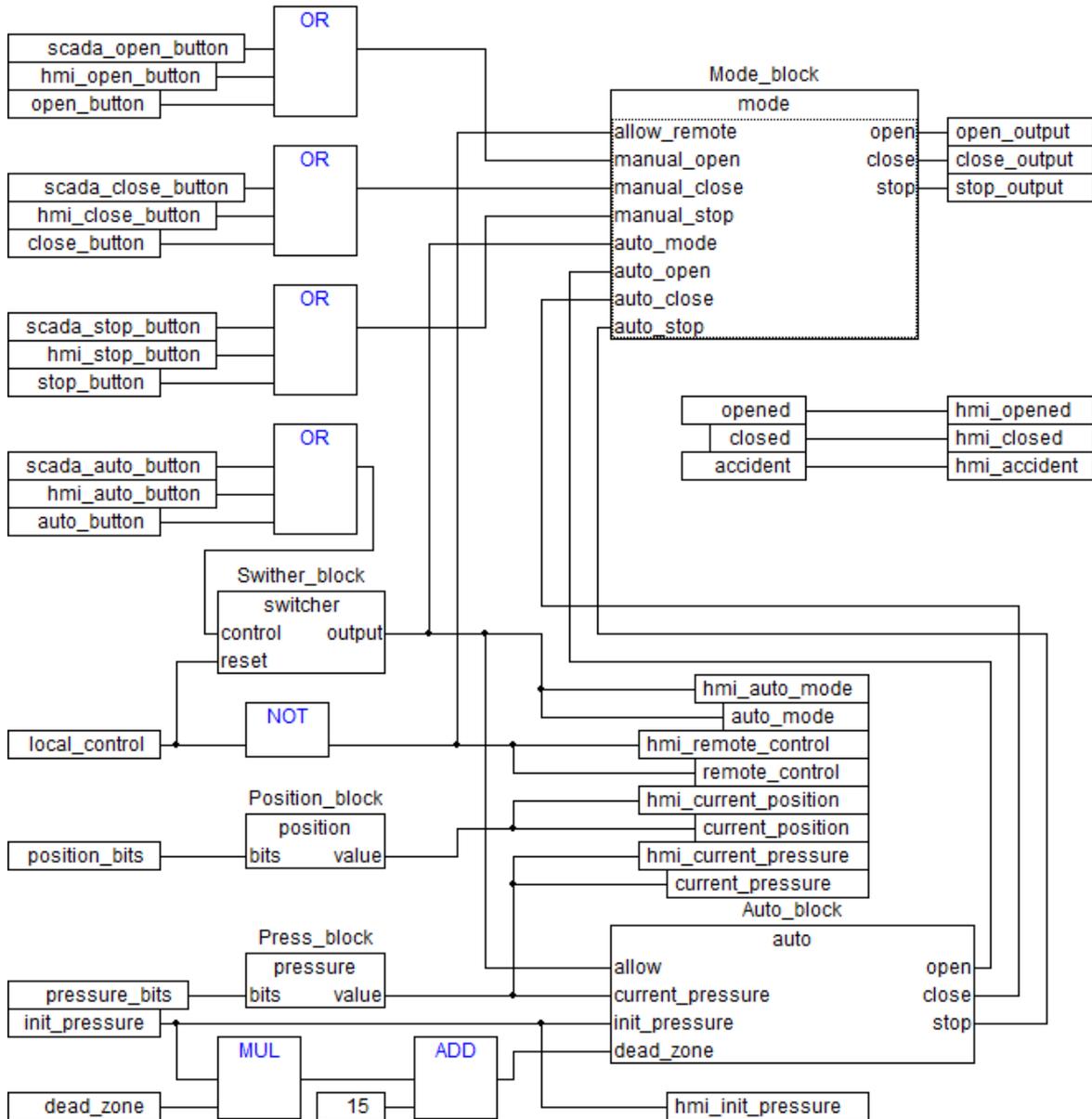


Рис. 25. Общая структура системы управления на языке CFC

Предварительно разберём эту программу. Она включает следующие функциональные блоки:

- mode – функциональный блок приёма управляющих сигналов (открыть, закрыть, останов) либо с кнопок ручного управления, либо с блока автоматического управления auto;
- swither – обычный выключатель для включения или отключения автоматического режима;
- position – преобразователь битового сигнала положения заслонки в градусы (физически сигнал передаётся от задвижки к контроллеру в виде тока 4...20 мА, а в программе контроллера представляется в виде числа в диапазоне 0...4095);

- pressure – преобразователь битового сигнала давления в паскали (физически сигнал передаётся от датчика давления к контроллеру в виде тока 4...20 мА, а в программе контроллера представляется в виде числа в диапазоне 0...4095);

- auto – блок автоматического управления, где сравнивается текущее давление в трубопроводе с уставкой, и если оно больше уставки (например когда создана нагрузка в трубе), то блок выдаёт сигнал на при-открытие заслонки, а если меньше – на закрытие. То есть всё сводится к поддержанию постоянного уровня давления в трубопроводе.

В программе PLC_PRG есть входные переменные, привязанные ко входам контроллера:

- open_button – переменная для открытия заслонки в ручном режиме с кнопки управления на стенде;

- close_button – переменная для закрытия заслонки в ручном режиме с кнопки управления на стенде;

- stop_button – переменная для остановки движения заслонки в ручном режиме с кнопки управления на стенде;

- auto_button – переменная включения автоматического режима управления с кнопки управления на стенде;

- local_control – переменная включения местного режима управления непосредственно через интерфейс нефтяной задвижки; сигнал приходит от нефтяной задвижки;

- position_bits – переменная положения заслонки; сигнал приходит от нефтяной задвижки в виде тока 4...20 мА;

- pressure_bits – переменная величины текущего давления в трубопроводе; сигнал приходит от датчика давления в виде тока 4...20 мА.

Выходные переменные PLC_PRG, привязанные к выходам контроллера:

- open_output – переменная открытия заслонки;

- close_output – переменная закрытия заслонки;

- stop_output – переменная остановки движения заслонки.

На рис. 26 приведены переменные главной программы PLC_PRG.

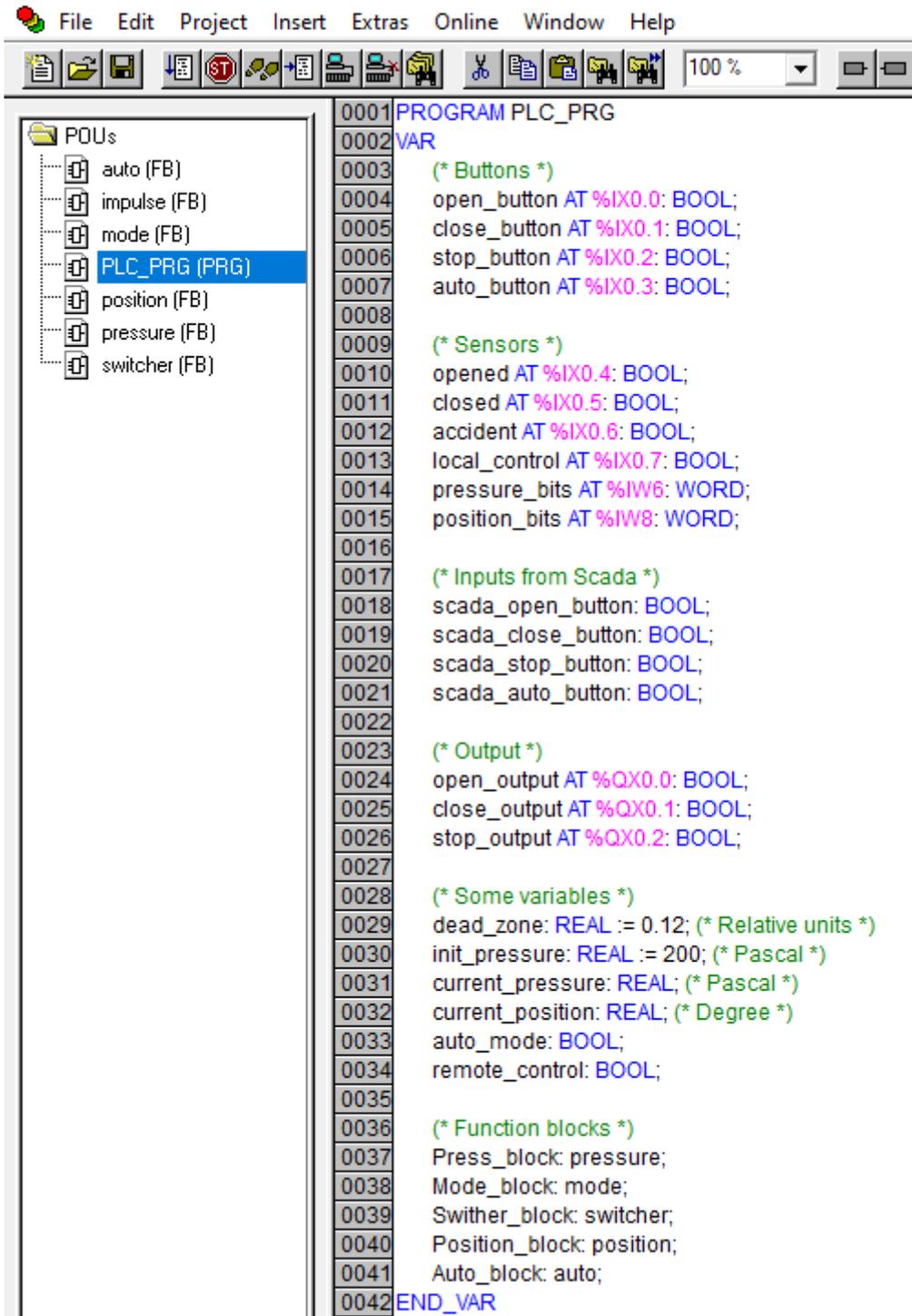


Рис. 26. Переменные главной программы PLC_PRG

6. Создайте функциональный блок режима управления. В первой системной вкладке, где список POU, щёлкните правой клавишей мыши и в появившемся контекстном меню выберите команду «Добавить объект» (Add Object). В автоматически открывшемся диалоговом окне вы-

берите тип объекта «Функциональный блок» (Function Block), задайте ему имя mode и язык CFC. Программный код блока показан на рис. 27.

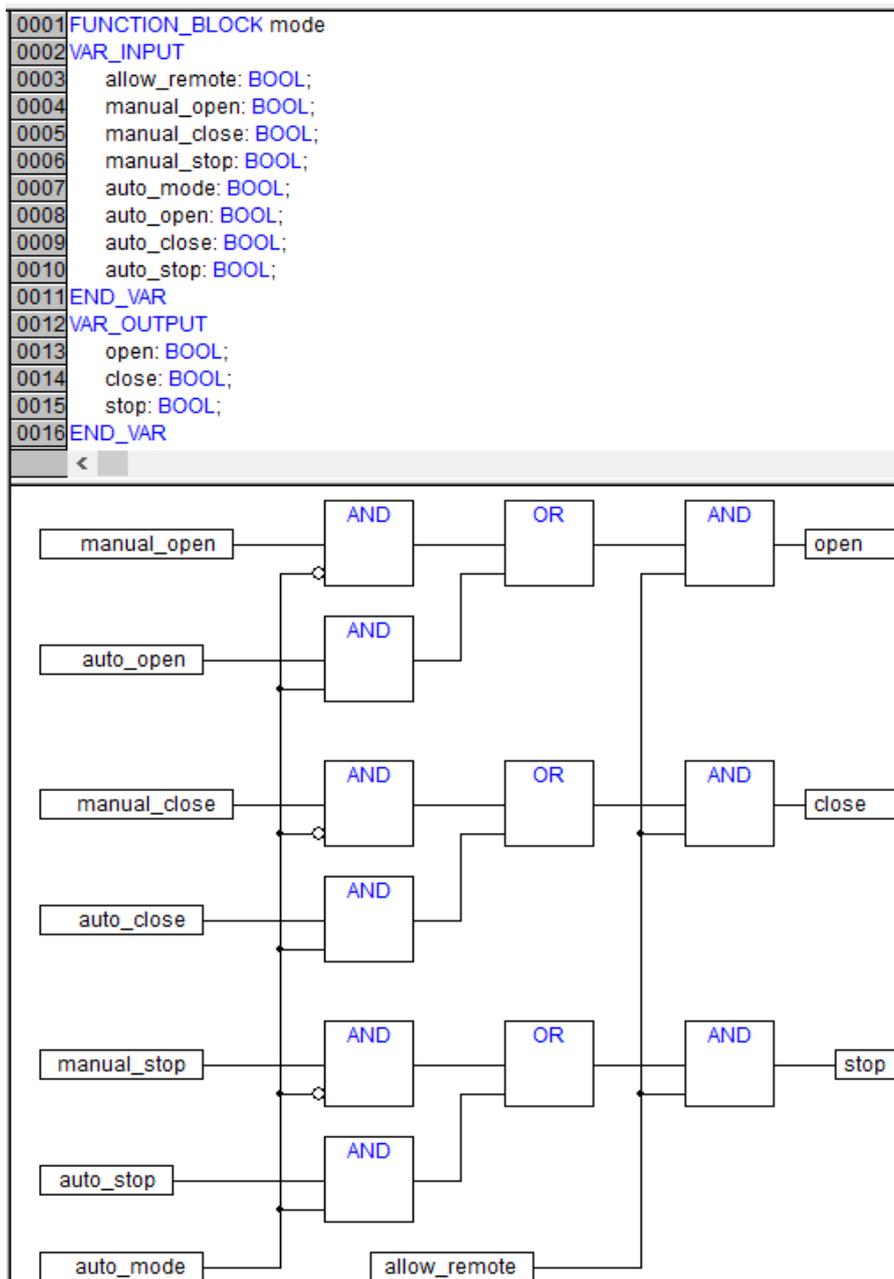


Рис. 27. Структура блока, реализованного на языке CFC, для задания режима управления

7. Создайте функциональный блок автоматического управления. В первой системной вкладке, где список ROU, щёлкните правой клавишей мыши и в появившемся контекстном меню выберите команду «Добавить объект» (Add Object). В автоматически открывшемся диалоговом окне выберите тип объекта «Функциональный блок» (Function Block), задайте ему имя auto и язык CFC. Программный код блока показан на рис. 28.

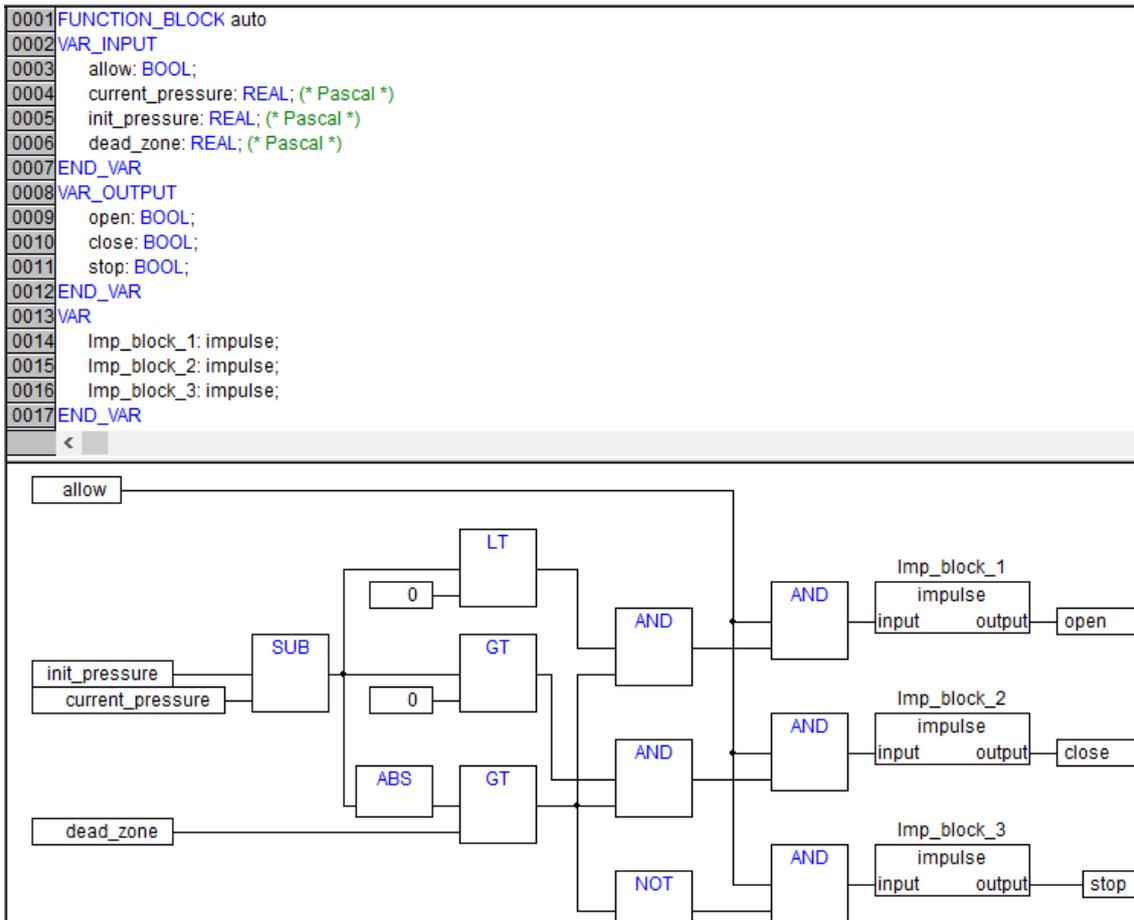


Рис. 28. Блок автоматического управления на языке CFC

Принцип работы блока auto заключается в следующем. Текущее реальное давление в трубопроводе `current_pressure` вычитается из значения заданной уставки `init_pressure` с помощью оператора `SUB`. Эта разность давлений (текущее рассогласование) преобразуется в абсолютную величину `ABS` для дальнейшего сравнения с допустимым рассогласованием, т. е. с зоной нечувствительности, с помощью оператора сравнения `GT` (зона нечувствительности формируется за пределами этого блока в Паскалях, т. е. в главной программе `PLC_PRG`, исходя из начально заданной зоны нечувствительности `dead_zone` в относительных единицах). Если текущее давление настолько сильно отличается от заданной уставки, что текущее рассогласование превышает допустимое (зону нечувствительности), то будет выработан сигнал `TRUE`, либо на открытие заслонки, либо на закрытие, в зависимости от полярности рассогласования. На отрицательную полярность реагирует оператор сравнения `LT` (если меньше 0), а на положительную – оператор сравнения `GT` (если больше 0). Выходные сигналы здесь выдаются в виде коротких импульсов. Это связано с особенностью работы самой задвижки – она управляется по заднему фронту входного управляющего сигнала.

Важный момент для понимания: зона нечувствительности здесь предусмотрена специально. Это нужно для того, чтобы система не перешла в режим колебательного процесса, когда из-за инерционности механизма заслонки она может закрыться сильнее, чем положено для полной стабилизации давления. При этом бы снова появилось рассогласование, но с противоположным знаком, что заставило бы систему, наоборот, приоткрыть заслонку, так и не поймав оптимального положения. Таким образом, когда давление стабилизируется при движении заслонки, т. е. когда текущее давление возвратится к заданной уставке, появившееся рассогласование будет находиться в зоне нечувствительности и заслонка зафиксирована в определённом положении.

8. *Создайте функциональный блок выключателя (тумблер).* В первой системной вкладке, где список ROU, щёлкните правой клавишей мыши и в появившемся контекстном меню выберите команду «Добавить объект» (Add Object). В автоматически открывшемся диалоговом окне выберите тип объекта «Функциональный блок» (Function Block), задайте ему имя switcher и язык FBD. Программный код блока показан на рис. 29.

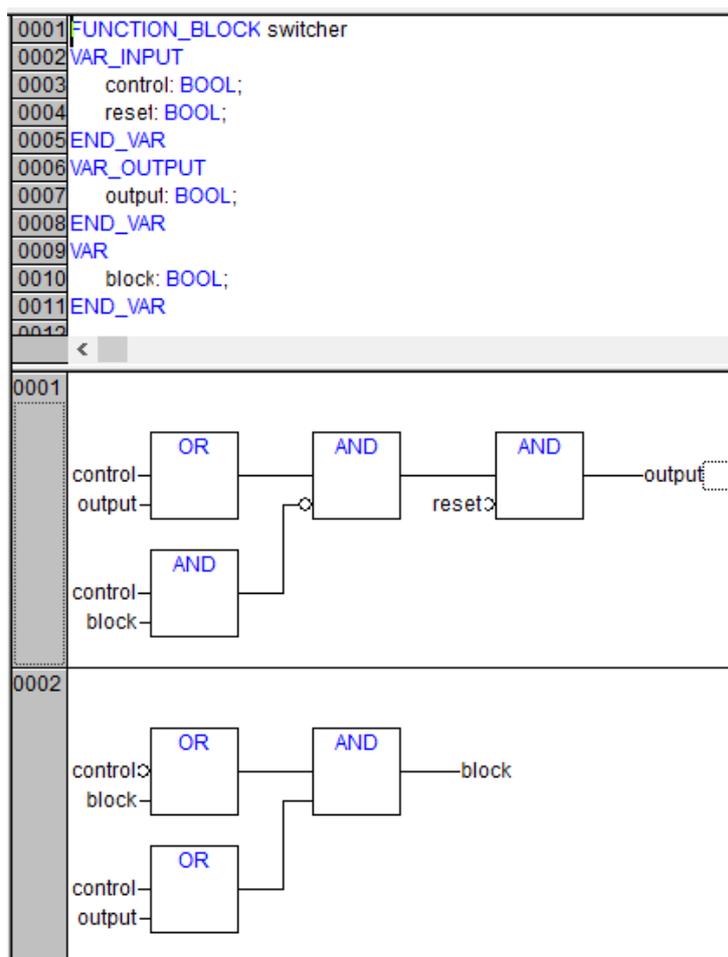


Рис. 29. Функциональный блок выключателя на языке FBD

9. *Создайте функциональный блок преобразователя давления.* В первой системной вкладке, где список ROU, щёлкните правой клавишей мыши и в появившемся контекстном меню выберите команду «Добавить объект» (Add Object). В автоматически открывшемся диалоговом окне выберите тип объекта «Функциональный блок» (Function Block), задайте ему имя pressure и язык FBD. Программный код блока показан на рис. 30.

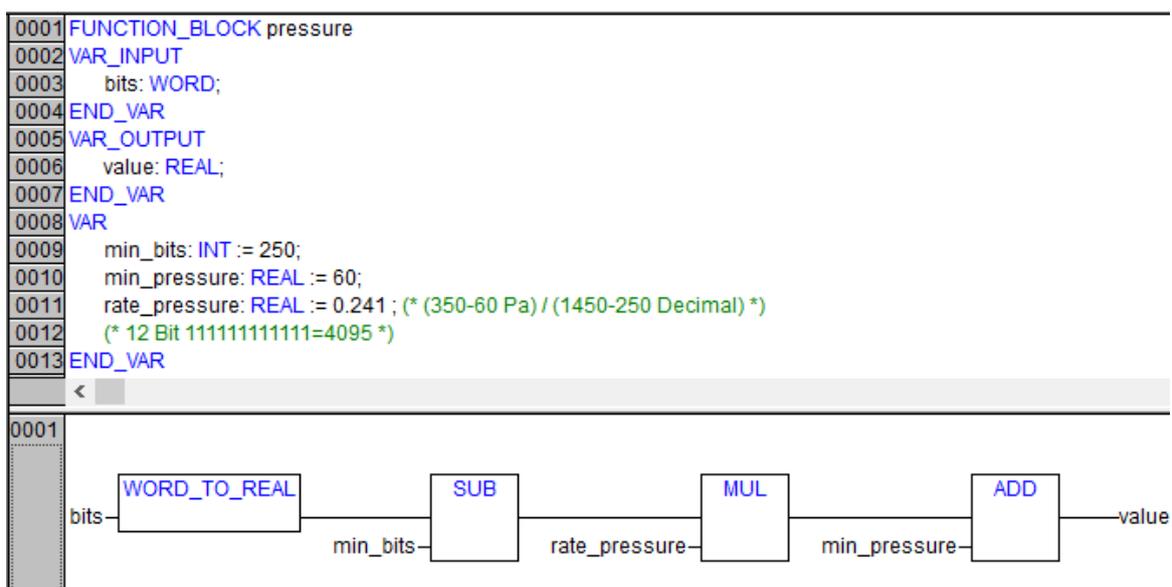


Рис. 30. Преобразователь давления на языке FBD

Здесь следует обратить внимание на коэффициент давления rate_pressure, который рассчитан как отношение разностей максимального и минимального давления к разности максимальных и минимальных битов. Эти минимальные и максимальные величины замерены опытным путём.

10. *Создайте функциональный блок преобразователя положения.* В первой системной вкладке, где список ROU, щёлкните правой клавишей мыши и в появившемся контекстном меню выберите команду «Добавить объект» (Add Object). В автоматически открывшемся диалоговом окне выберите тип объекта «Функциональный блок» (Function Block), задайте ему имя position и язык FBD. Программный код блока показан на рис. 31. Блок работает аналогично преобразователю давления.

11. *Создайте функциональный блок формирования импульса.* В первой системной вкладке, где список ROU, щёлкните правой клавишей мыши и в появившемся контекстном меню выберите команду «Добавить объект» (Add Object). В автоматически открывшемся диалоговом окне выберите тип объекта «Функциональный блок» (Function Block), задайте ему имя impulse и язык SFC. Программный код блока показан на рис. 32. Задача программного кода – сформировать кратковременный

сигнал TRUE на выходе output длительностью не менее одной секунды с момента подачи на управляемый вход input сигнала TRUE любой продолжительности, в том числе менее одной секунды.

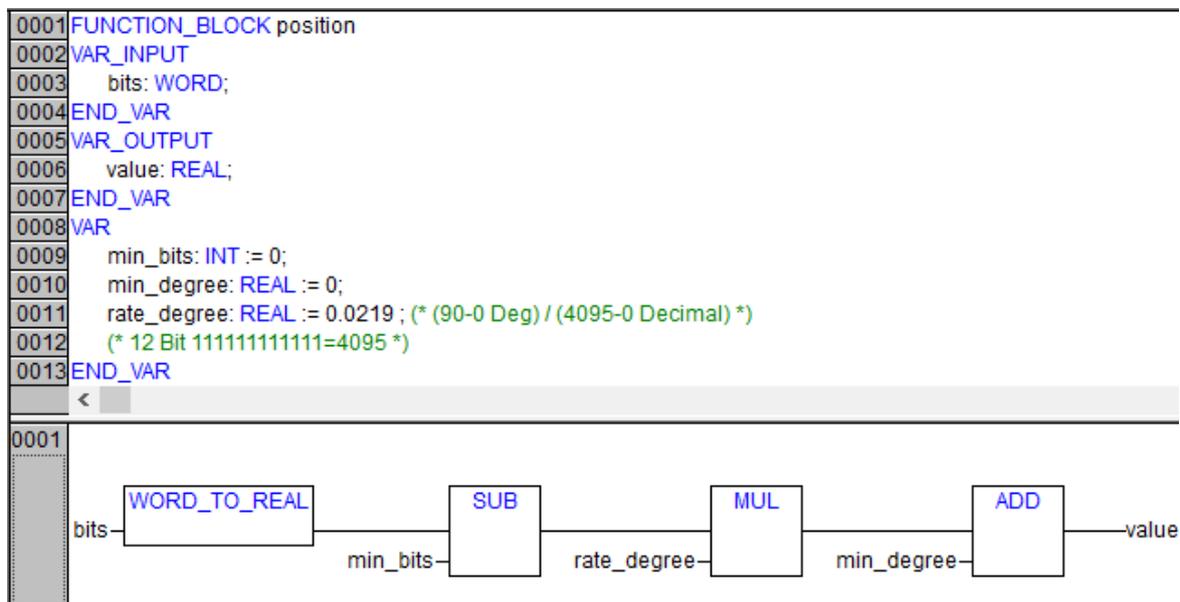


Рис. 31. Преобразователь положения заслонки на языке FBD

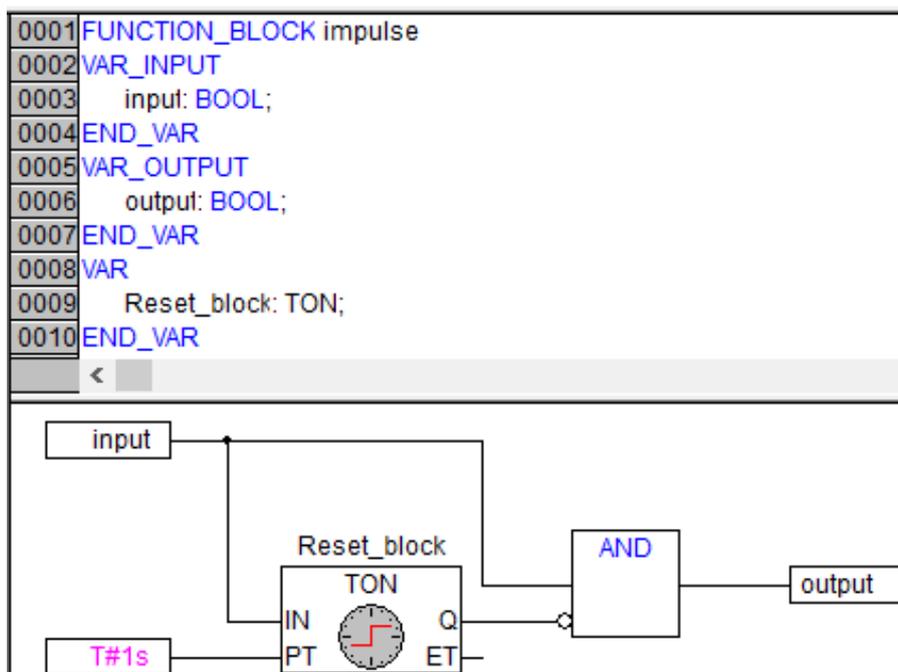


Рис. 32. Блок создания импульса на языке CFC

12. Объявите переменные передачи данных по CAN Open. Перейдите на вкладку «Ресурсы» (Resources), и далее в подраздел «Глобальные переменные» (Global_Variables). Кликните правой клавишей мыши по

разделу (папке) глобальных переменных добавьте новый объект с настройками как на рис. 33.

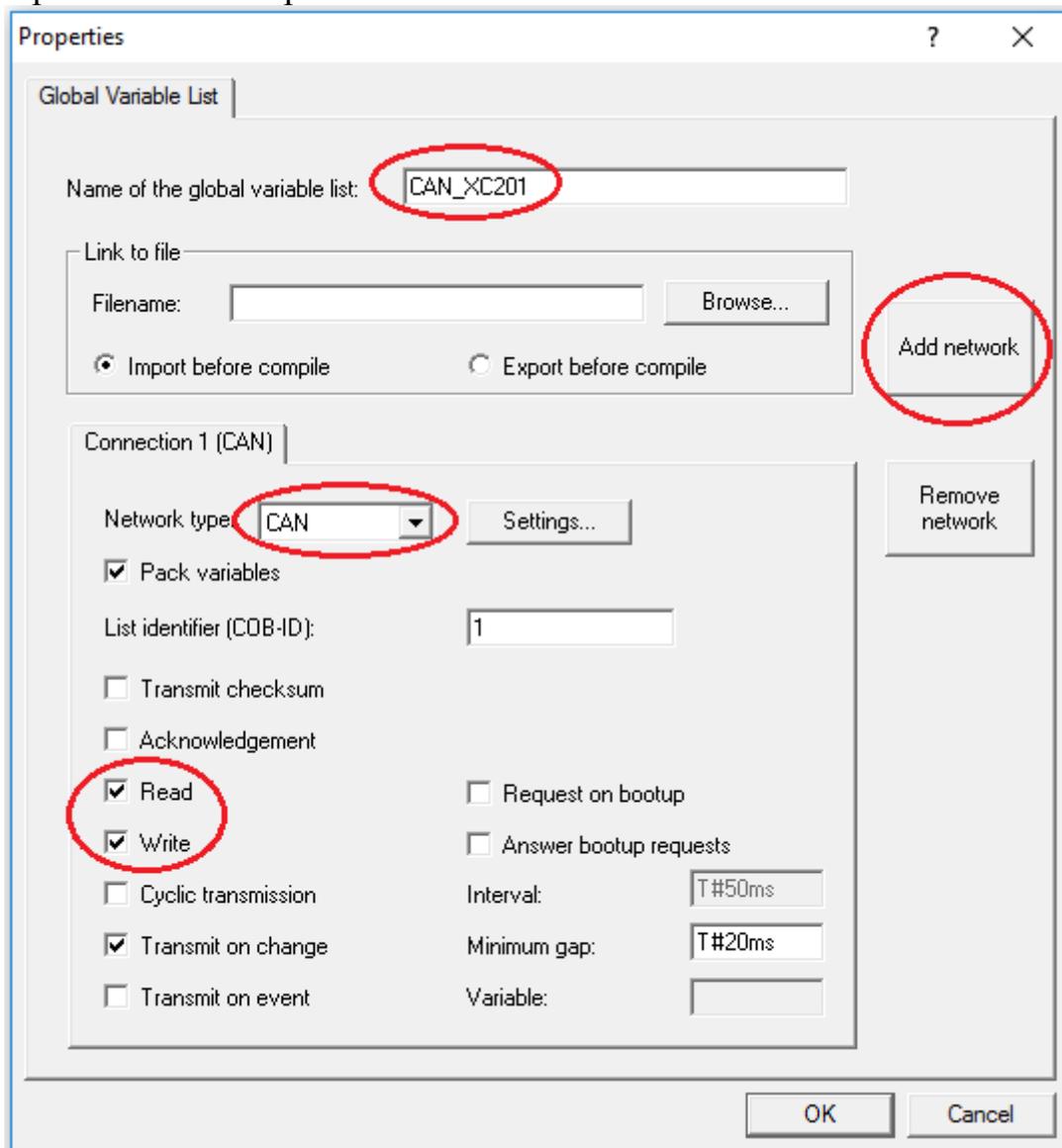


Рис. 33. Добавление нового объекта в глобальные переменные

Задайте переменные передачи данных по протоколу CAN Open от контроллера в HMI панель, так и от HMI панели в контроллер, как показано на рис. 34.

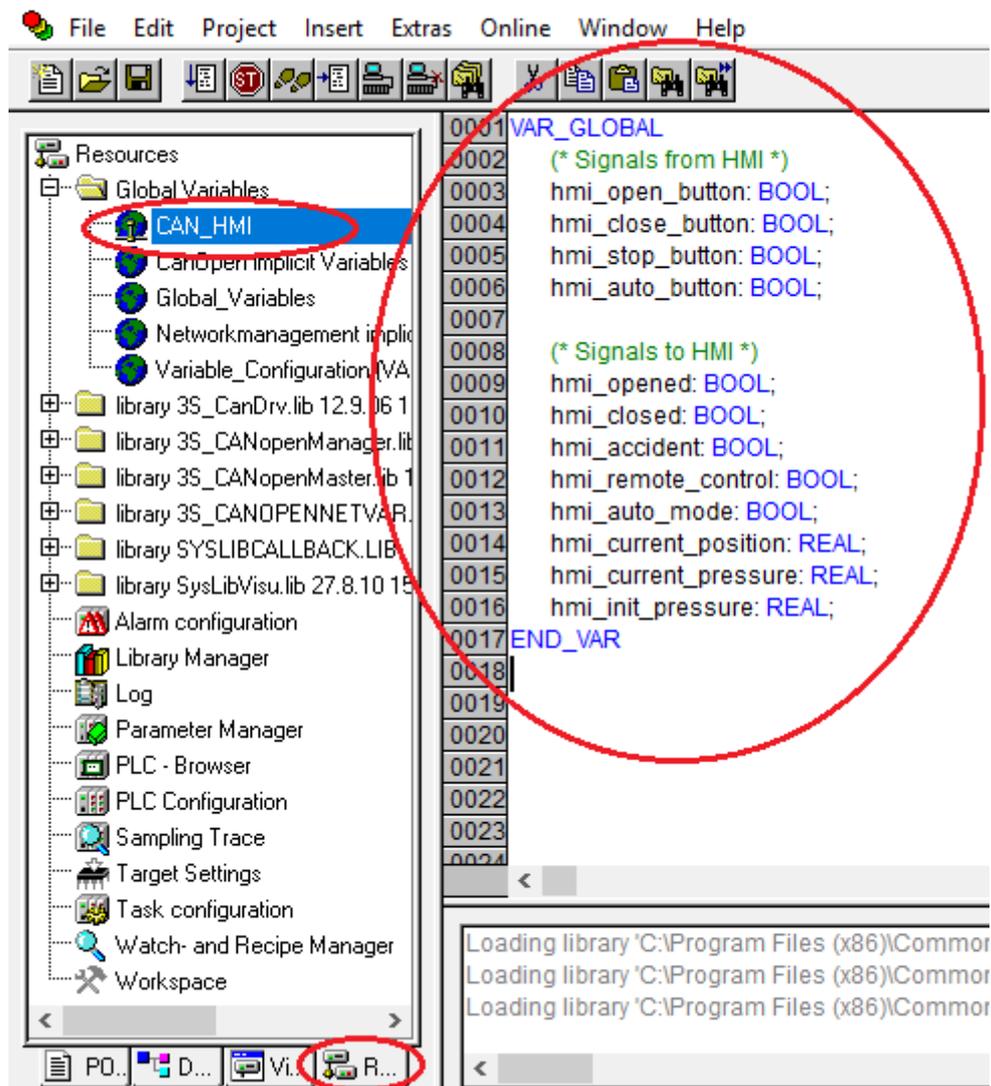


Рис. 34. Переменные передачи CAN Open данных

13. *Создайте объект визуализации.* Перейдите на третью страницу организатора объектов CODESYS с названием Visualization. Далее правой кнопкой мыши выберите команду «Добавить объект» (Add object). Присвойте новому объекту имя valve. Далее перейдите во внутреннюю рабочую область объекта valve и нарисуйте графические элементы управления (рис. 35).

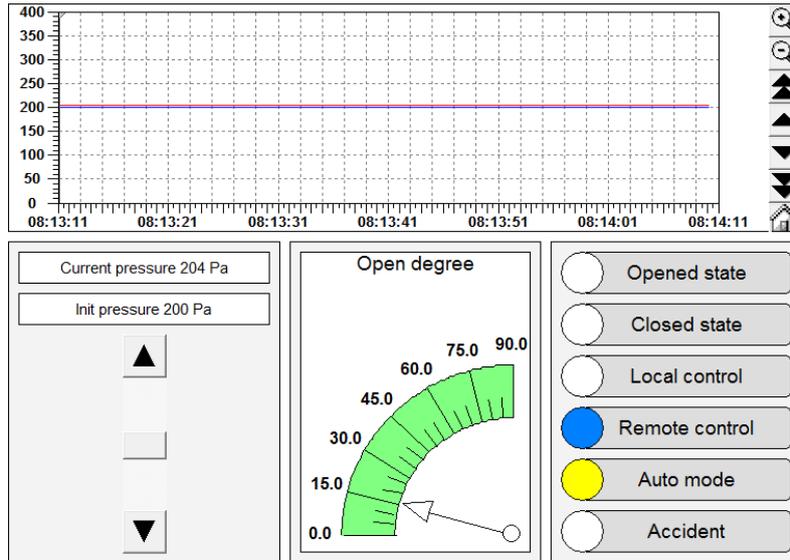


Рис. 35. Элементы визуализации управления нефтяной задвижкой

Затем сделайте необходимые привязки объектов визуализации к переменным главной корневой программы. К объектам визуализации относятся следующее: прямоугольники для отображения текущего и заданного давления, «ползунок» для задания давления (левая часть визуализации), шкала положения заслонки (средняя часть визуализации), сигнальные лампы (правая часть визуализации).

Следуйте настройкам для элемента трендов (рис. 36 и 37), а также настройкам отображения текущего давления и текущего положения заслонки (рис. 38 и 39).

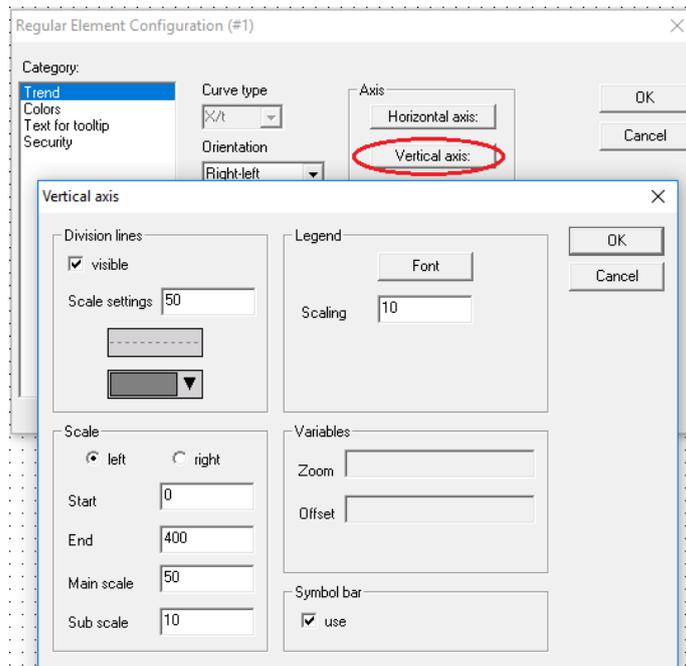


Рис. 36. Настройки вертикальной шкалы для элемента трендов

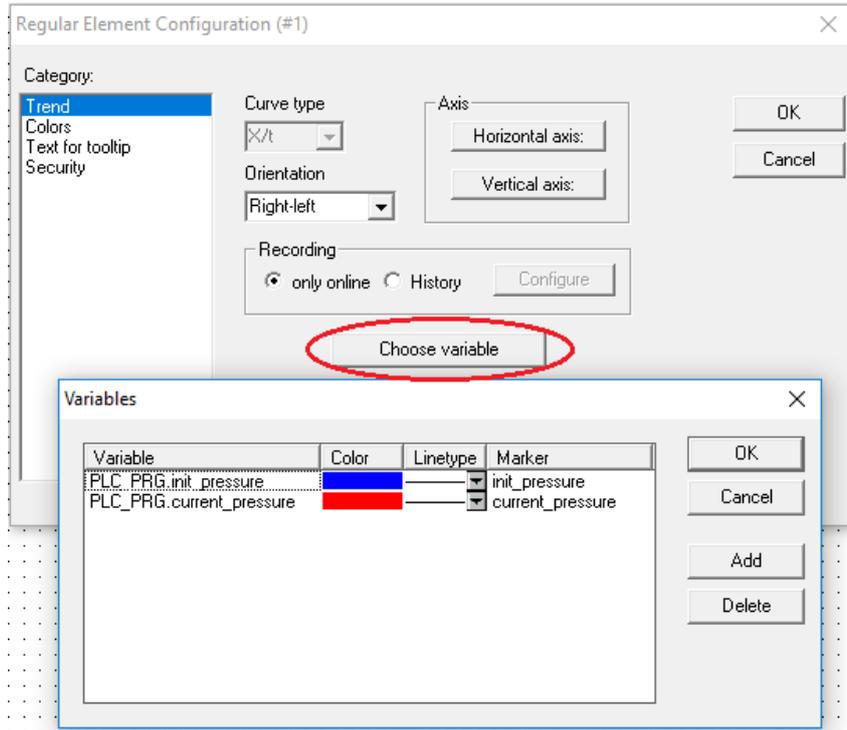


Рис. 37. Настройки переменных для элемента трендов

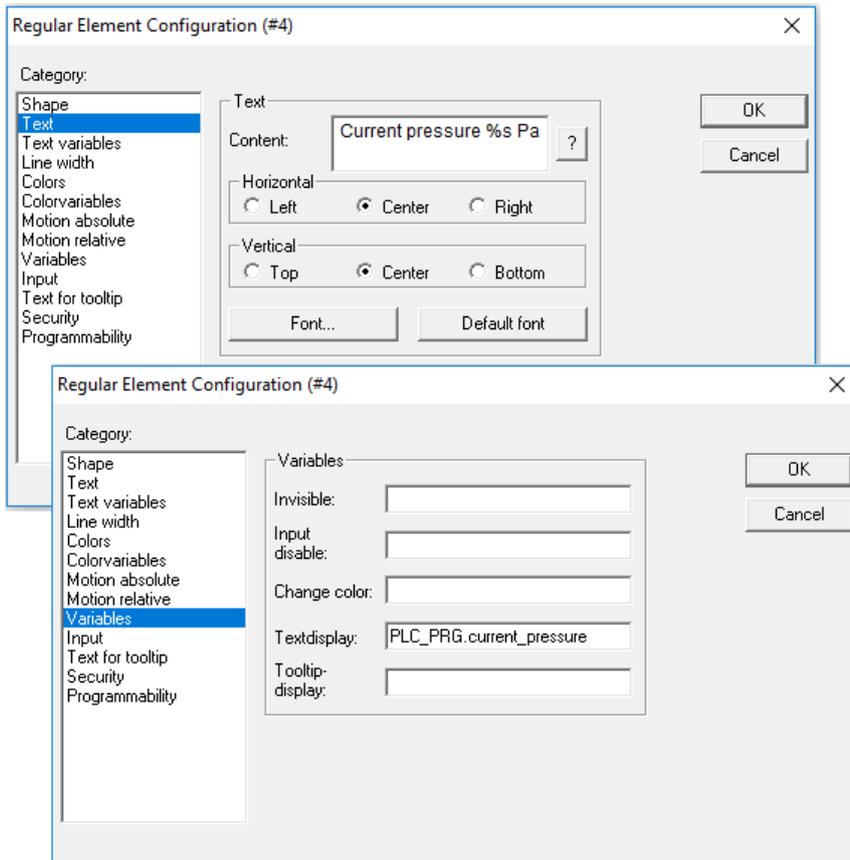


Рис. 38. Настройки отображения текущего давления

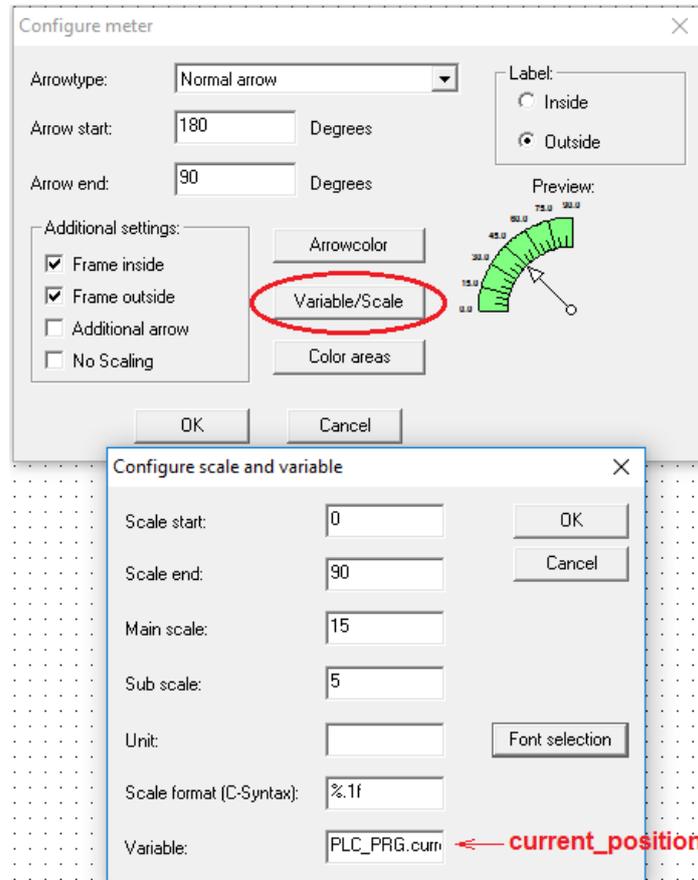


Рис. 39. Настройки отображения положения заслонки

На рис. 40 и 41 приведены настройки сигнализатора открытого положения Opened state.

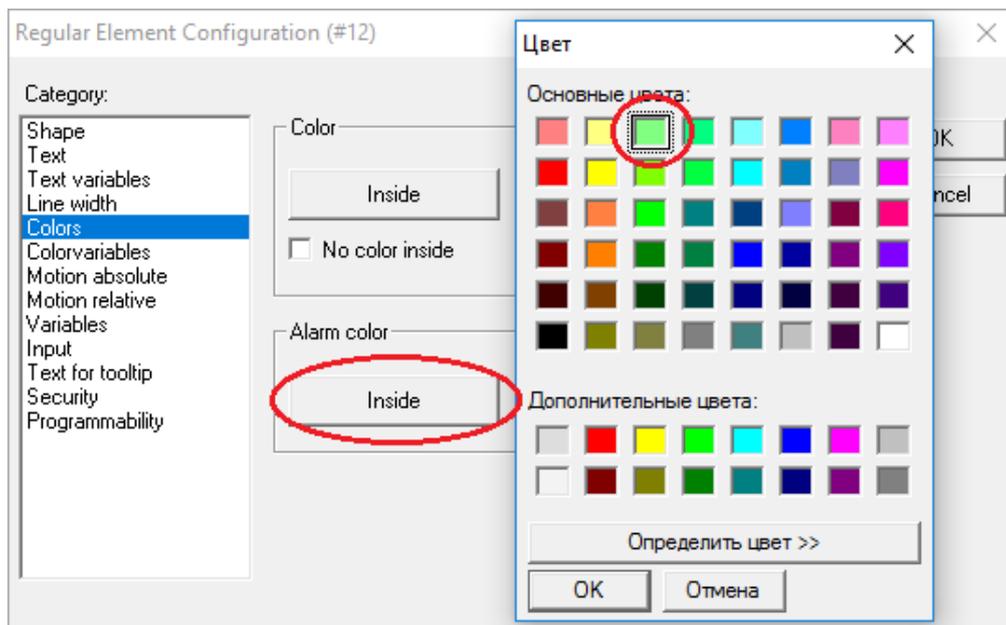


Рис. 40. Настройки цвета круглого сигнализатора Opened state

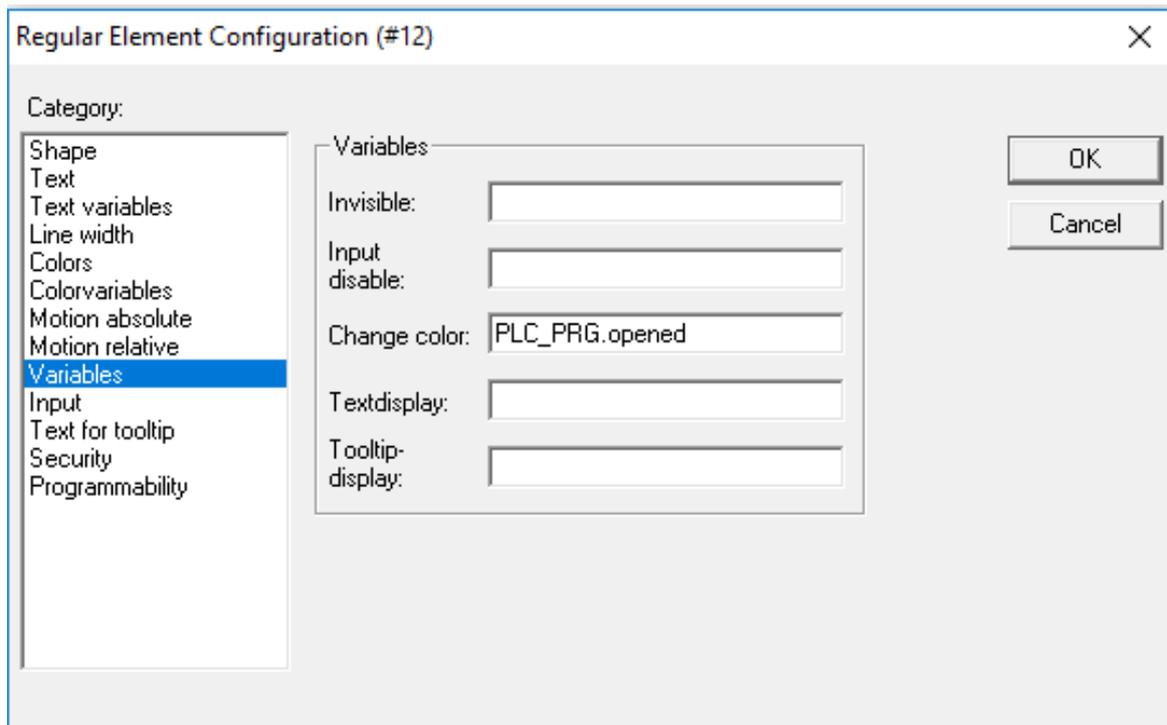


Рис. 41. Привязка переменной круглого сигнализатора *Opened state*

14. Проверьте работоспособность программы в режиме эмуляции. Для запуска проекта в режиме эмуляции установите во вкладке «Онлайн» (Online) галочку против «Эмуляция» (Simulation). Откомпилируйте проект: «Проект/Компилировать всё» (Project/Rebuild All). Установите соединение с контроллером: «Онлайн/Соединение» (Online/Login). Запустите проект: «Онлайн/Запуск» (Online/Run). Поэкспериментируйте с заданием различных переменных.

15. Сформируйте символьный файл с тегами OPC переменных. Для этого откройте опции проекта в *Project/Options*. Далее зайдите в символьную конфигурацию и в окне установки атрибутов выберите все те переменные программы PLC_PRG (п. 16). Эти переменные отвечают за следующее: приём сигналов с электрических кнопок, приём сигналов с кнопок и датчиков от SCADA системы, вывод управляющих сигналов с контактных разъёмов контроллера, посылку сигналов в SCADA систему (рис. 42).

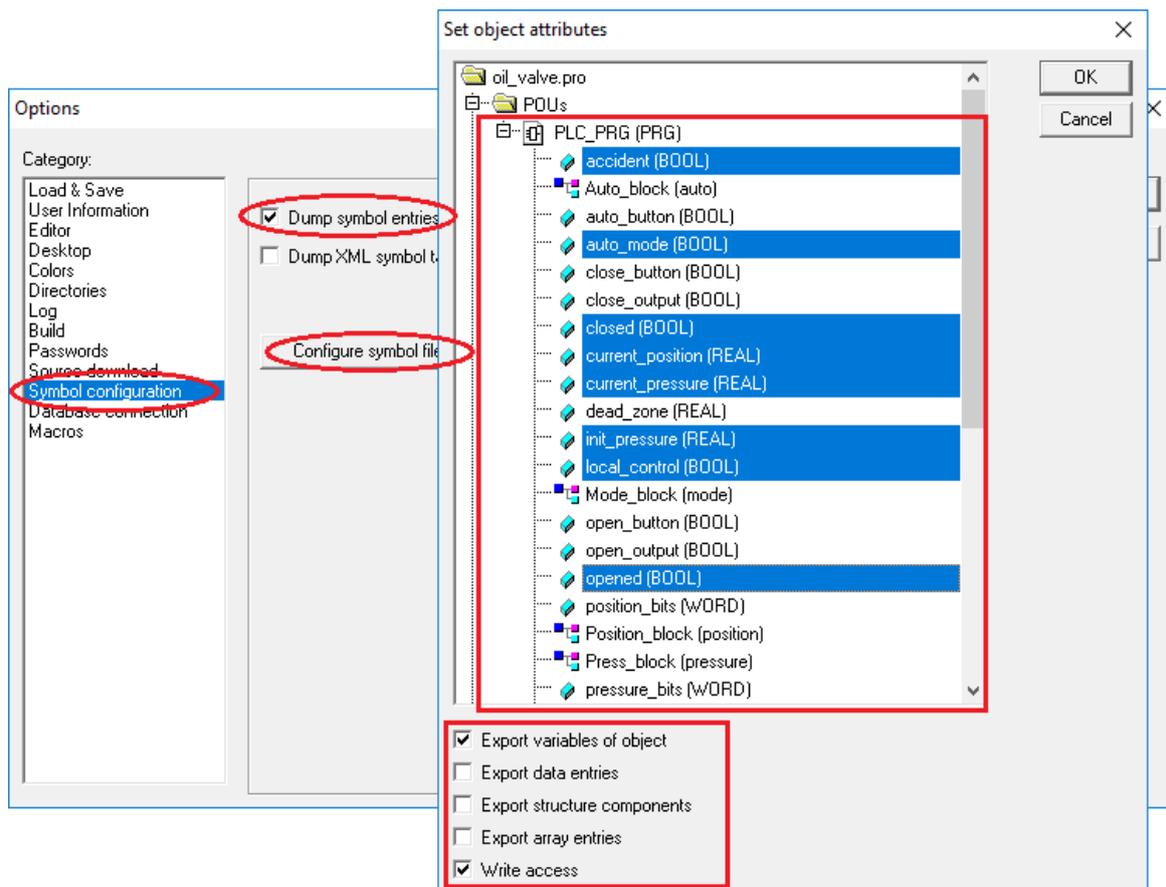


Рис. 42. Конфигурация символьного файла

16. Откомпилируйте проект «Проект/Компилировать всё» (*Project/Rebuild All*). В результате в папке проекта должен появиться символьный файл с расширением «SYM». В составе тегов символьного файла будут присутствовать следующие записи:

```

PLC_PRG.accident:BOOL:1:6:0:b:16#00000020
PLC_PRG.auto_mode:BOOL:4:246:1:b:16#00000040
PLC_PRG.closed:BOOL:1:5:0:b:16#00000020
PLC_PRG.current_position:REAL:4:260:4:b:16#00000040
PLC_PRG.current_pressure:REAL:4:256:4:b:16#00000040
PLC_PRG.init_pressure:REAL:4:252:4:b:16#00000040
PLC_PRG.local_control:BOOL:1:7:0:b:16#00000020
PLC_PRG.opened:BOOL:1:4:0:b:16#00000020
PLC_PRG.remote_control:BOOL:4:247:1:b:16#00000040
PLC_PRG.scada_auto_button:BOOL:4:245:1:b:16#00000040
PLC_PRG.scada_close_button:BOOL:4:243:1:b:16#00000040
PLC_PRG.scada_open_button:BOOL:4:242:1:b:16#00000040
PLC_PRG.scada_stop_button:BOOL:4:244:1:b:16#00000040

```

17. Загрузите программу во внутреннюю память контроллера XC-CPU201. Во вкладке «Онлайн» (Online) откройте диалог «Параметры соединения» (Communication parameters) и нажмите кнопку «Создать» (New) для настройки нового соединения с типом TCP/IP. Далее присвойте ему осмысленное имя. Задайте IP-адрес в соответствии с картой IP адресов (рис. 3), в данном случае 192.168.119.23. Установите соединение с контроллером: «Онлайн/Соединение» (Online/Login). Далее подтвердите загрузку (download) кода проекта. Также во вкладке «Онлайн» (Online) выберите опцию «Создать загрузочный проект» (Create boot project).

2.4. Создание программы на языке программирования ST в среде CODESYS для контроллера HMI панели визуализации XV102

1. Запустите программную среду CODESYS 2.3.9.

2. Создайте новый проект: «Файл/Создать» (File/New). **ВНИМАНИЕ!** путь к папке с файлом проекта и имя самого файла должны быть названы латинскими буквами.

3. Настройте целевую платформу (Target Settings). Откройте вкладку «Целевая платформа» (Target Settings) и на странице диалогового окна «Конфигурация» (Configuration) выберите тип контроллера XV-1xx-V2.3.9 SP8. Во вкладке сетевой функциональности (Network functionality) поставьте галочку на против «Support network variables» и в открывшемся поле ввода пропишите «CAN» (рис. 43). Подтвердите ввод нажатием кнопки ОК.

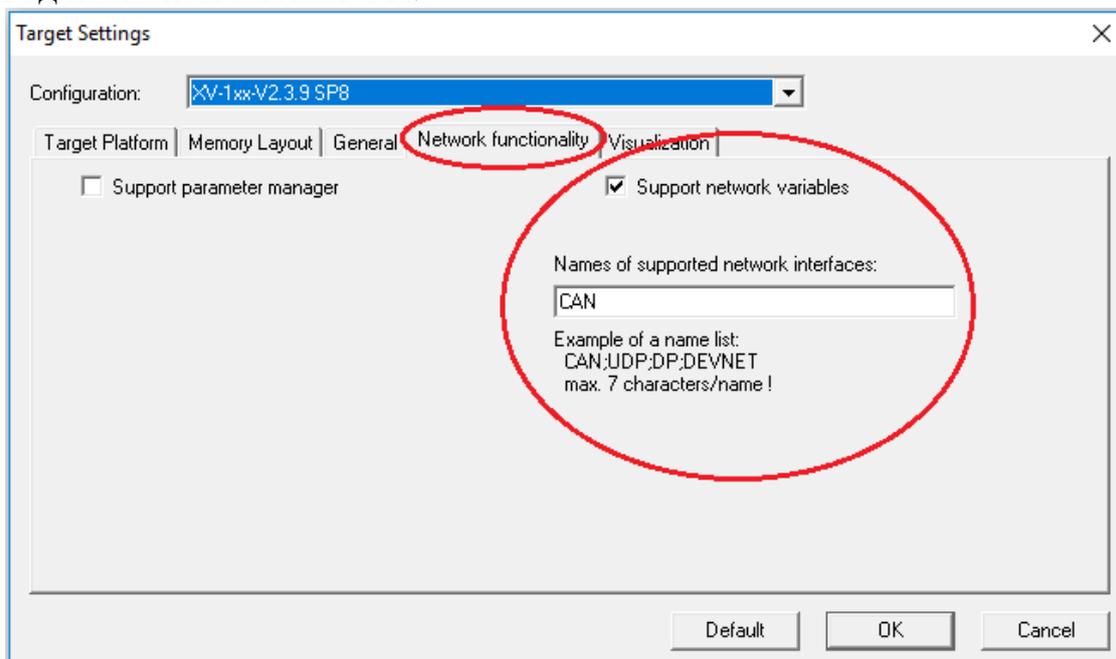


Рис. 43. Установка поддержки сетевой передачи данных через CAN Open протокол

4. *Создайте главную программу.* После настройки целевой платформы в автоматически появившемся диалоговом окне New POU выберите тип компонента «Программа» (Program), язык ST, задайте имя программы PLC_PRG.

5. *Настройте конфигурацию контроллера в HMI.* Зайдите во вкладку ресурсов «Resources» проекта и откройте конфигурацию контроллера «PLC Configuration» (рис. 44). Далее нажмите правой клавишей мыши по верхнему уровню меню (Configuration) и добавьте «CanMaster».

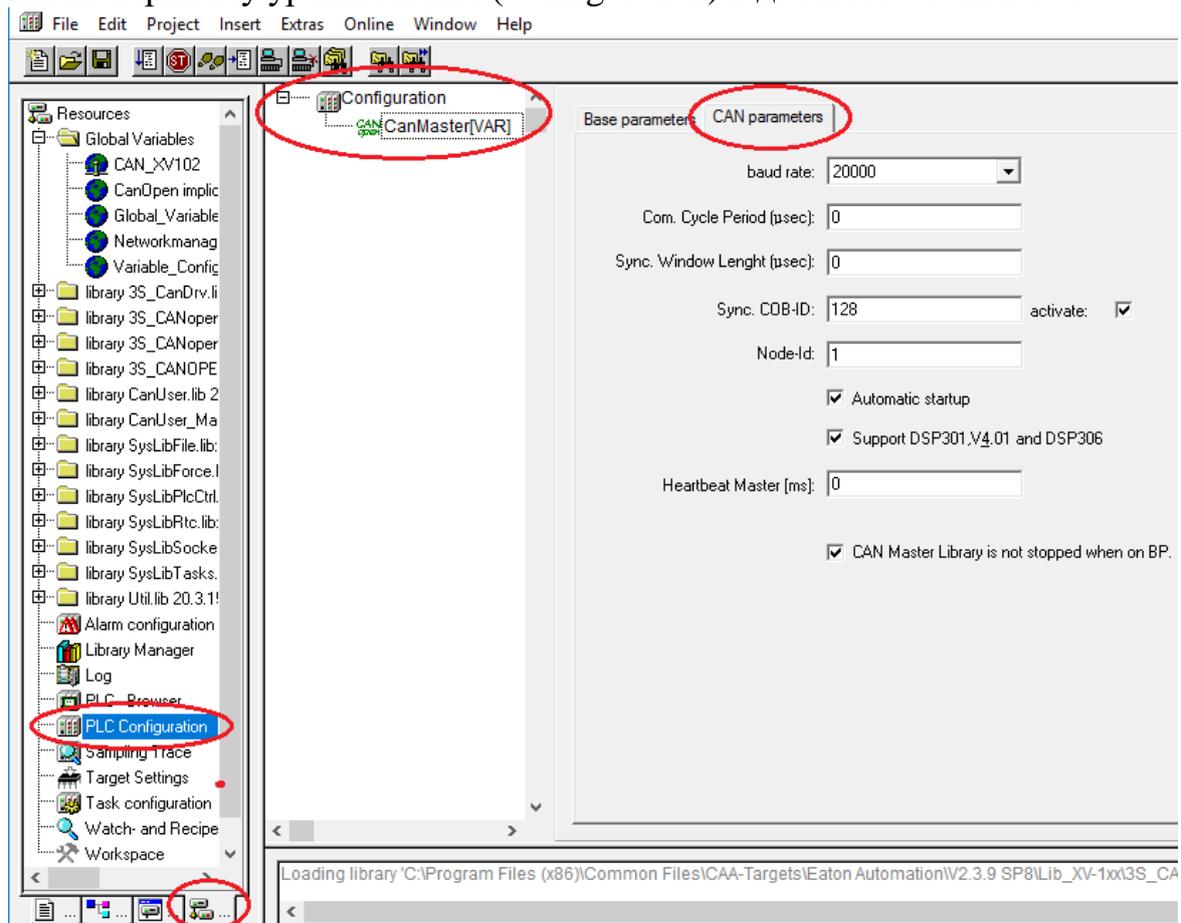


Рис. 44. Конфигурация контроллера в HMI

6. *Объявите переменные передачи данных по CAN Open.* Перейдите на вкладку «Ресурсы» (Resources), и далее в подраздел «Глобальные переменные» (Global_Variables). Кликните правой клавишей мыши по разделу (папке) глобальных переменных добавьте новый объект с настройками как на рис. 45.

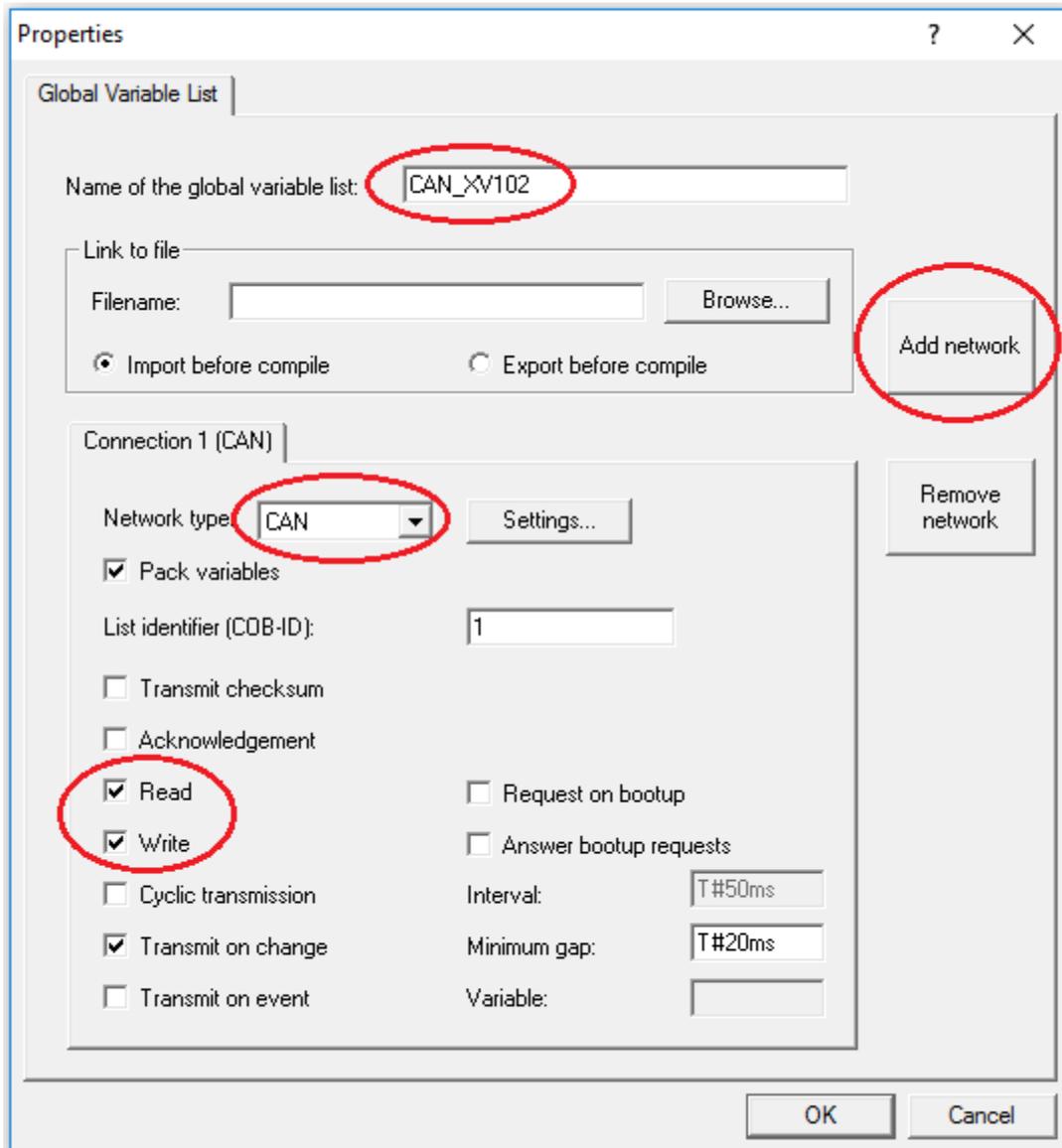


Рис. 45. Добавление нового объекта в глобальные переменные

Задайте переменные передачи данных по протоколу CAN Open от контроллера в HMI панель, так и от HMI панели в контроллер, как показано на рис. 46. **ВНИМАНИЕ! Наименование CAN переменных и тип данных должны совпадать в проекте ПЛК, ПЛК панели и проекте визуализации.**

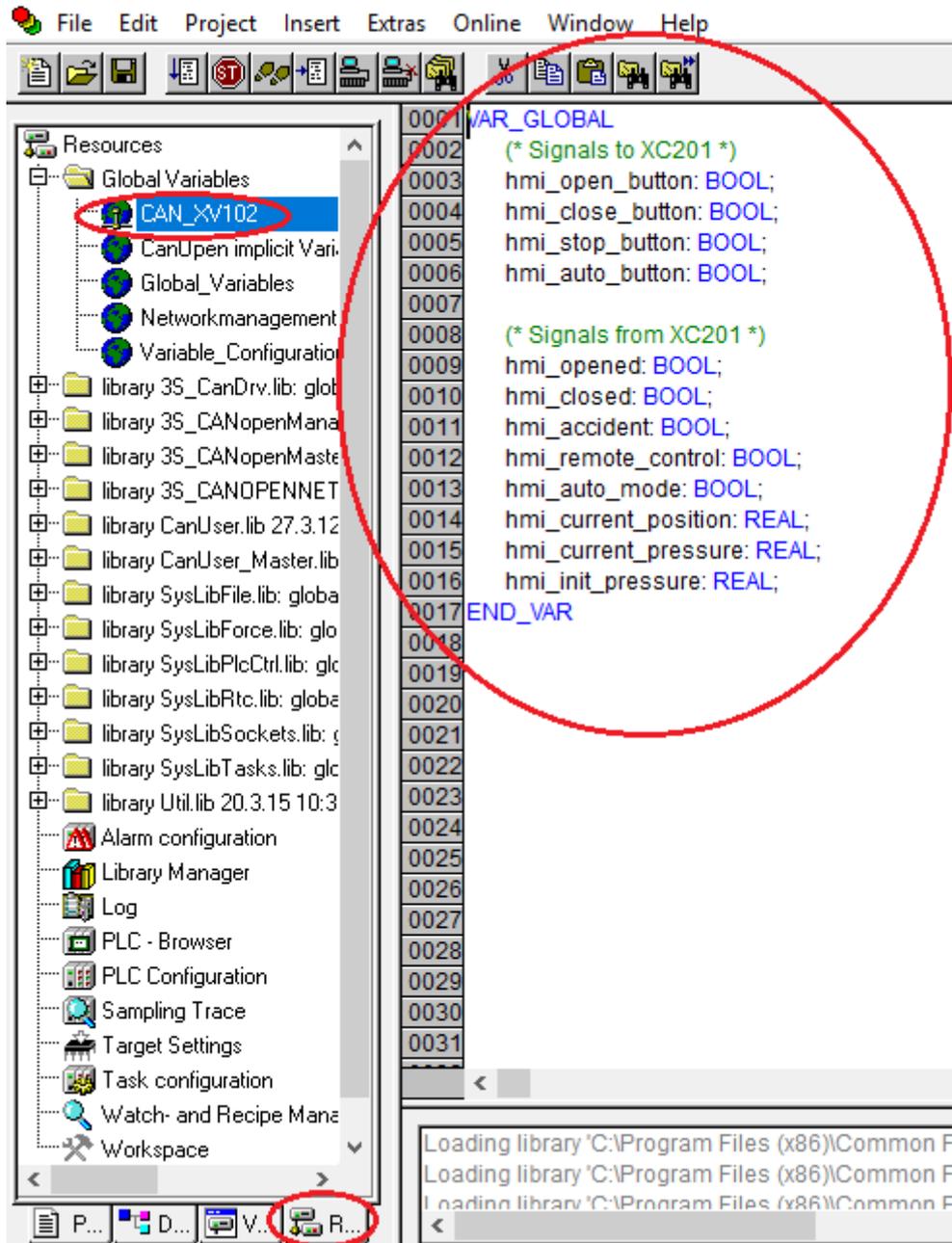


Рис. 46. Переменные передачи CAN Open данных

Главная программа во внутреннем контроллере HMI панели на языке ST показана на рис. 47.

POUs	0001	PROGRAM PLC_PRG
PLC_PRG (PRG)	0002	VAR
	0003	open_button: BOOL;
	0004	close_button: BOOL;
	0005	stop_button: BOOL;
	0006	auto_button: BOOL;
	0007	opened: BOOL;
	0008	closed: BOOL;
	0009	accident: BOOL;
	0010	remote_control: BOOL;
	0011	auto_mode: BOOL;
	0012	current_position: REAL;
	0013	current_pressure: REAL;
	0014	init_pressure: REAL;
	0015	END_VAR
	<	
	0001	IF auto_button = TRUE
	0002	THEN hmi_auto_button := TRUE; ELSE hmi_auto_button := FALSE;
	0003	END_IF
	0004	
	0005	IF hmi_auto_mode = FALSE THEN
	0006	IF open_button = TRUE
	0007	THEN hmi_open_button := TRUE; ELSE hmi_open_button := FALSE;
	0008	END_IF
	0009	IF close_button = TRUE
	0010	THEN hmi_close_button := TRUE; ELSE hmi_close_button := FALSE;
	0011	END_IF
	0012	IF stop_button = TRUE
	0013	THEN hmi_stop_button := TRUE; ELSE hmi_stop_button := FALSE;
	0014	END_IF
	0015	END_IF
	0016	
	0017	IF hmi_opened = TRUE
	0018	THEN opened := TRUE; ELSE opened := FALSE;
	0019	END_IF
	0020	
	0021	IF hmi_closed = TRUE
	0022	THEN closed := TRUE; ELSE closed := FALSE;
	0023	END_IF
	0024	
	0025	IF hmi_accident = TRUE
	0026	THEN accident := TRUE; ELSE accident := FALSE;
	0027	END_IF
	0028	
	0029	IF hmi_remote_control = TRUE
	0030	THEN remote_control := TRUE; ELSE remote_control := FALSE;
	0031	END_IF
	0032	
	0033	IF hmi_auto_mode = TRUE
	0034	THEN auto_mode := TRUE; ELSE auto_mode := FALSE;
	0035	END_IF
	0036	
	0037	current_position := hmi_current_position;
	0038	current_pressure := hmi_current_pressure;
	0039	init_pressure := hmi_init_pressure;

Рис. 47. Главная программа во внутреннем контроллере HMI панели

7. Загрузите программу во внутреннюю память контроллера HMI панели XV102. Во вкладке «Онлайн» (Online) откройте диалог «Параметры соединения» (Communication parameters) и нажмите кнопку «Создать» (New) для настройки нового соединения с типом TCP/IP. Далее присвойте ему осмысленное имя. Задайте IP-адрес в соответствии с картой IP адресов (рис. 3), в данном случае 192.168.119.24. Установите соединение с контроллером: «Онлайн/Соединение» (Online/Login). Далее подтвердите загрузку (download) кода проекта. Также во вкладке «Онлайн» (Online) выберите опцию «Создать загрузочный проект» (Create boot project).

2.5. Создание графического интерфейса управления в среде GALILEO для HMI панели визуализации XV102

1. Откройте Galileo 8 и создайте новый проект Project/New.

2. Выберите тип панели Panel Type. В автоматически открывшемся окне выбрать Panel selection и после этого выбрать наименование панели оператора (рис. 48). Также после создания проекта, тип панели можно выбрать в Config/Panel Type.

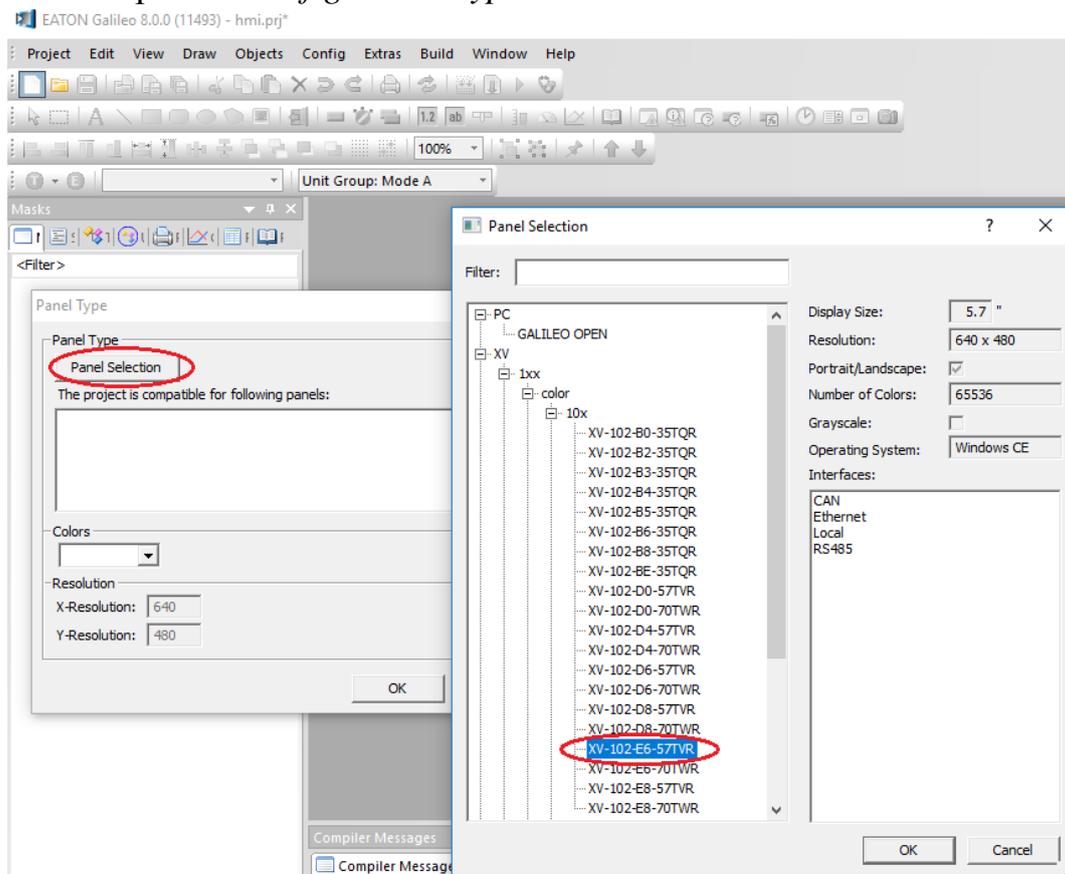


Рис. 48. Выбор панели оператора

3. Выберите коммуникацию Config/Select Communication. В открывшемся окне нажмите кнопку Add, затем выберите Codesys Xsoft-CoDesys-2/MXpro (рис. 49).

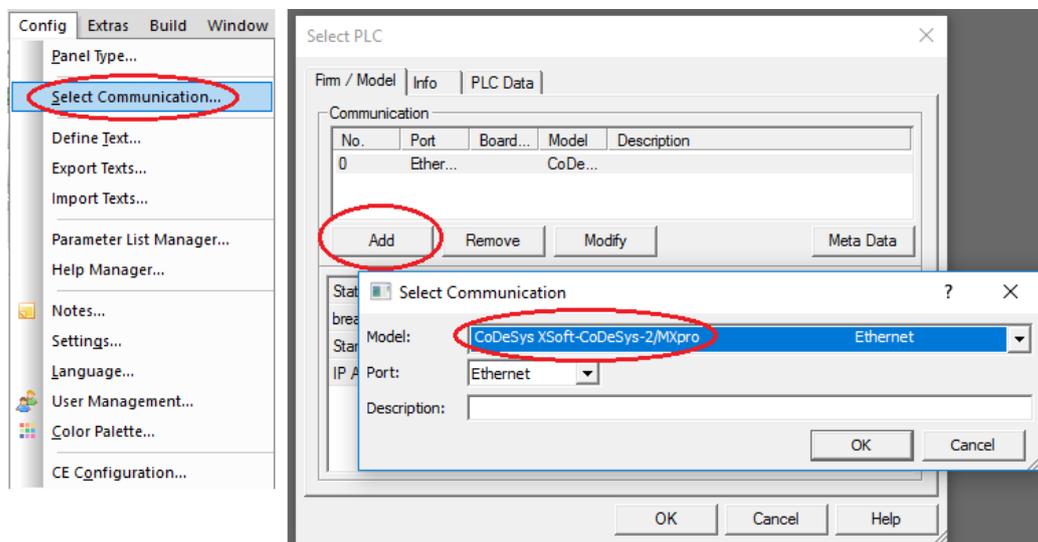


Рис. 49. Выбор коммуникации

4. Добавьте теги. В дереве проекта выбрать вкладку Tag, нажать правой клавишей мыши на соответствующем типе данных и задать имя тэгу (рис. 50). **ВНИМАНИЕ! Наименование САН переменных и тип данных должны совпадать в проекте ПЛК, ПЛК панели и проекте визуализации.**

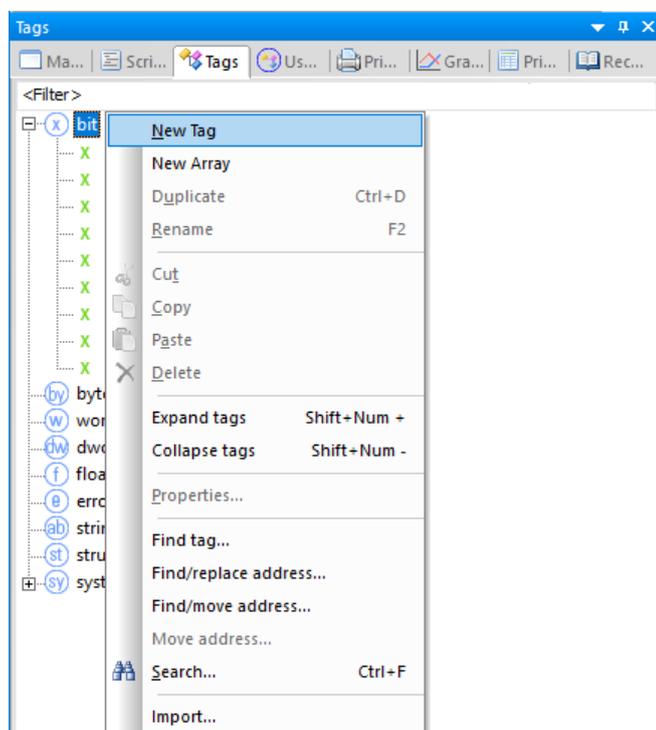


Рис. 50. Добавление тега

5. *Настройте тег.* Два раза кликните на созданном тэге. Перейдите на вкладку Address и в соответствующей строке нажмите клавишу выбора. После этого задайте параметр <prog> и <tag> (рис. 51).

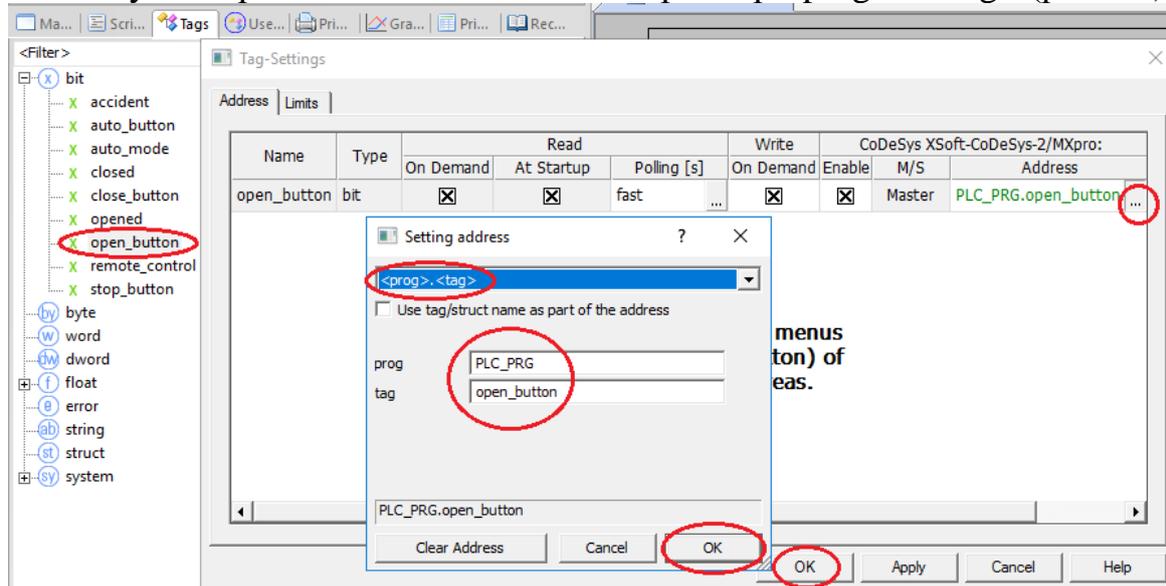


Рис. 51. Задание параметров тега

Аналогичным образом, добавьте и настройте все остальные теги в соответствии со следующим списком необходимых тегов:

- accident
- auto_button
- auto_mode
- closed
- close_button
- opened
- open_button
- remote_control
- stop_button

6. *Добавьте маску.* В дереве проекта выберите вкладку Mask, нажмите правой клавишей мыши и в диалоговом окне выберите New. Задайте имя маски (рис. 52).

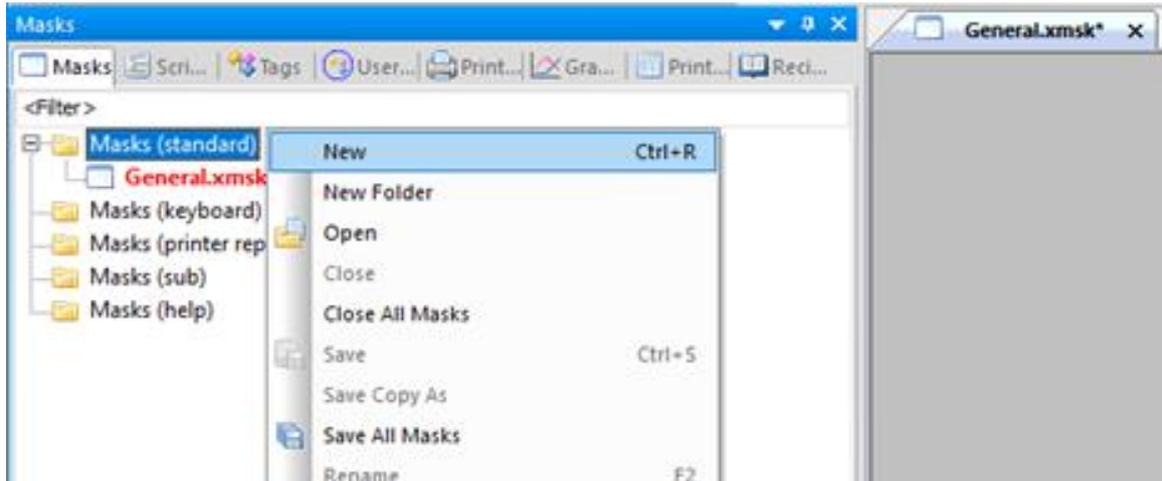


Рис. 52. Добавление маски

7. Нарисуйте необходимые графические элементы (рис. 53). Для этого используйте такие типы элементов, как «Text», «Rectangle», «Button», «Flag Display» (рис. 54).

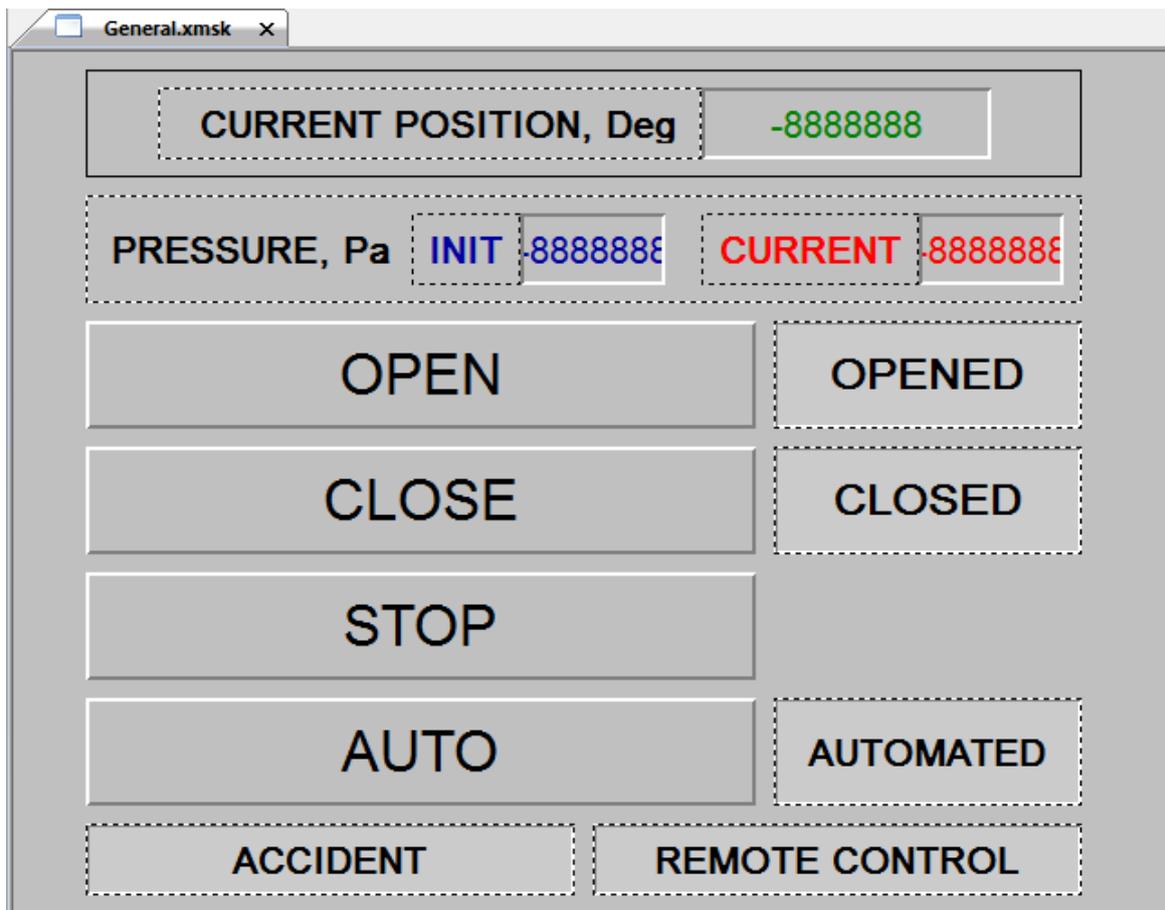


Рис. 53. Графические элементы управления на главной маске

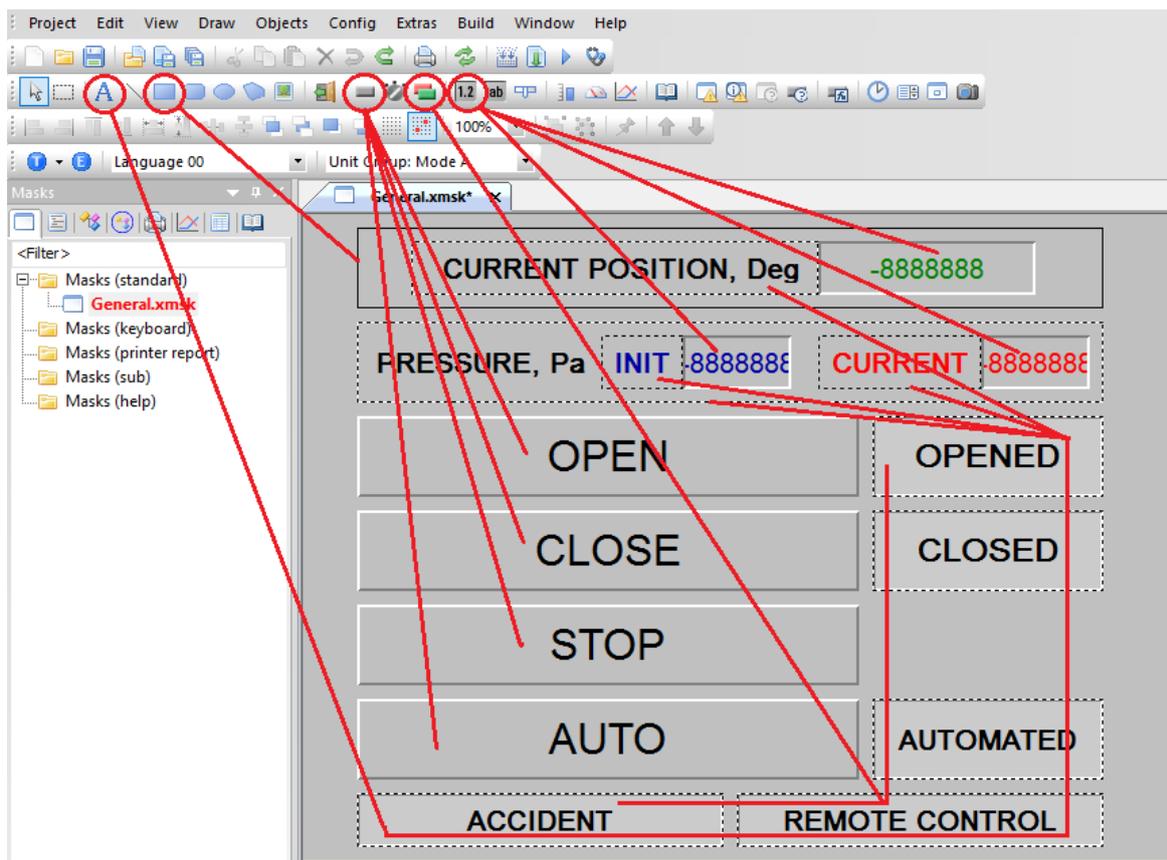


Рис. 54. Инструменты рисования графических элементов

8. *Настройте кнопки управления.* Для этого двойным кликом левой клавишей мыши перейдите в настройки данного графического элемента. Пример настроек для кнопки «open_button» (OPEN) приведён на рис. 55, 56.

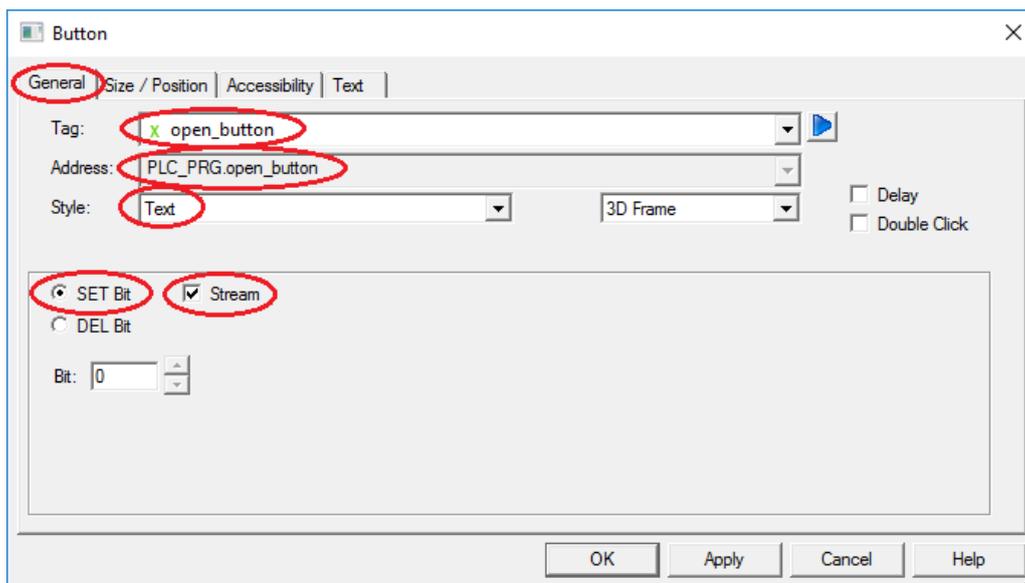


Рис. 55. Основные настройки кнопки «open_button»

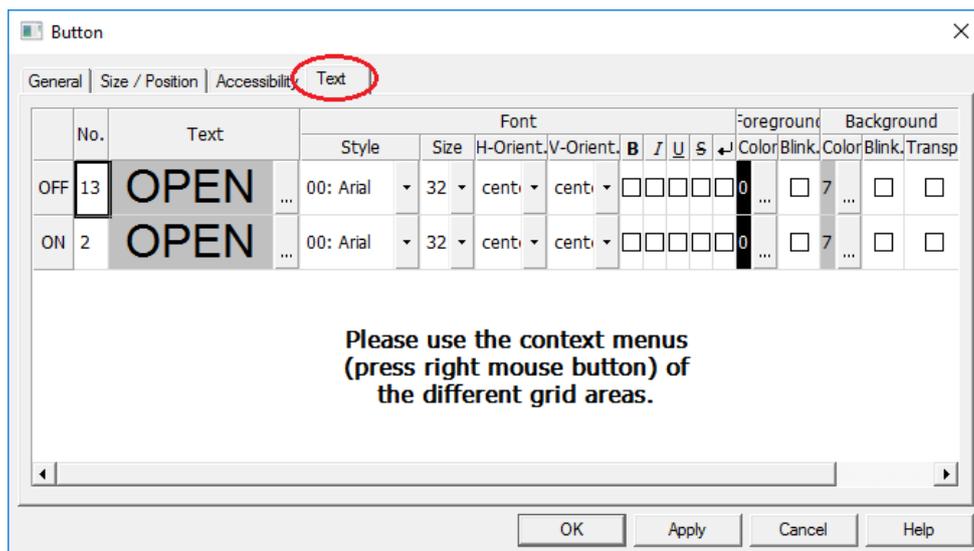


Рис. 56. Настройки текста кнопки «open_button»

Аналогичным образом настройте все остальные кнопки управления, считая, что кнопки должны привязываться к тегам, имена которых заканчиваются на «_button».

Дополнительно для кнопок «OPEN», «CLOSE» и «STOP» настройте блокировку, запрещающую управление нефтяной задвижкой от НМІ панели при включенном автоматическом режиме (рис. 57).

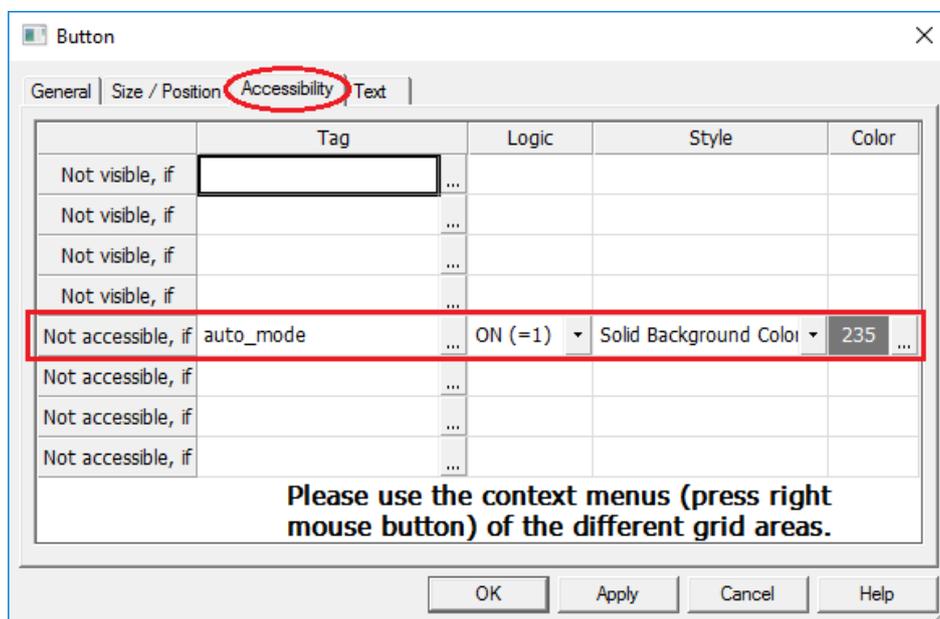


Рис. 57. Блокировка кнопки при автоматическом режиме

9. *Настройте флаги отображения.* Для этого двойным кликом левой клавишей мыши перейдите в настройки данного графического элемента. Пример настроек для флага «opened» приведён на рис. 58, 59.

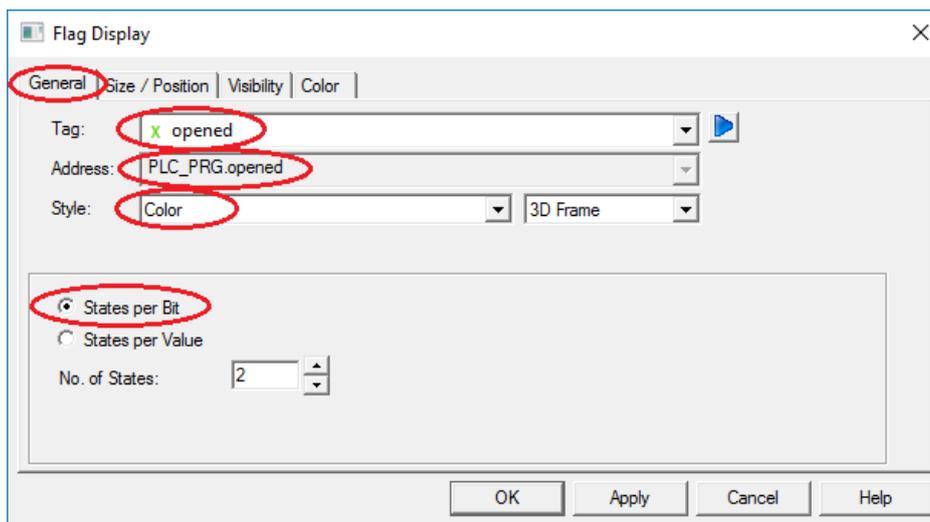


Рис. 58. Основные настройки флага «opened»

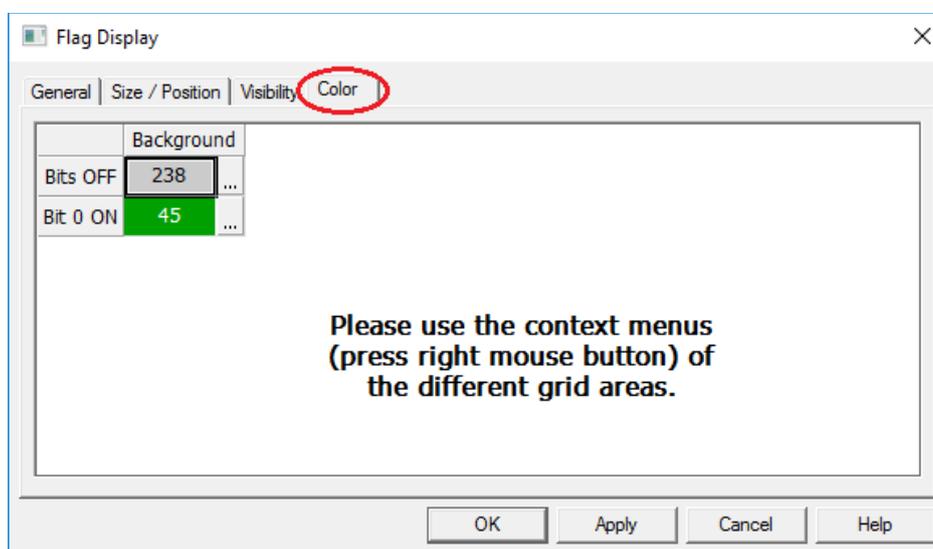


Рис. 59. Цветовые настройки флага «opened»

Аналогичным образом настройте все остальные флаги отображения, считая, что флаги должны привязываться к тегам, имена которых заканчиваются на «_work».

10. Откомпилируйте проект перед загрузкой в HMI панель Build/Compile.

11. На панели оператора включите FTP-сервер Start/Communication/FTP-Server.

12. Загрузите проект визуализации в панель оператора. Для этого выберите Build/Download и в открывшемся окне нажмите кнопку FTP Path. Затем New Connection. В открывшемся окне задайте имя, укажите

IP адрес панели (в данном случае 192.168.119.14 в соответствии со схемой по рис. 3) и нажмите ОК (рис. 60). Далее подтвердите загрузку кнопкой «Download» (рис. 61). Для успешного проведения загрузки необходимо, чтобы путь к сохраненным файлам проекта не содержал русских букв.

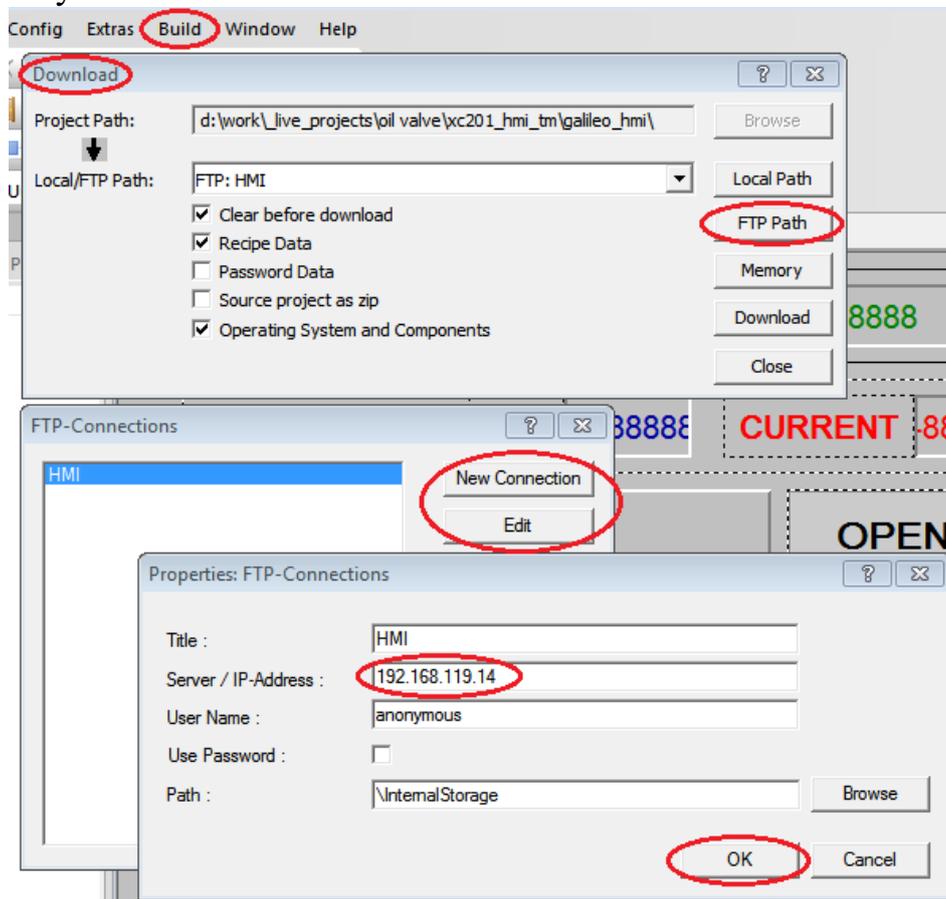


Рис. 60. Настройка FTP соединения с HMI панелью для загрузки проекта

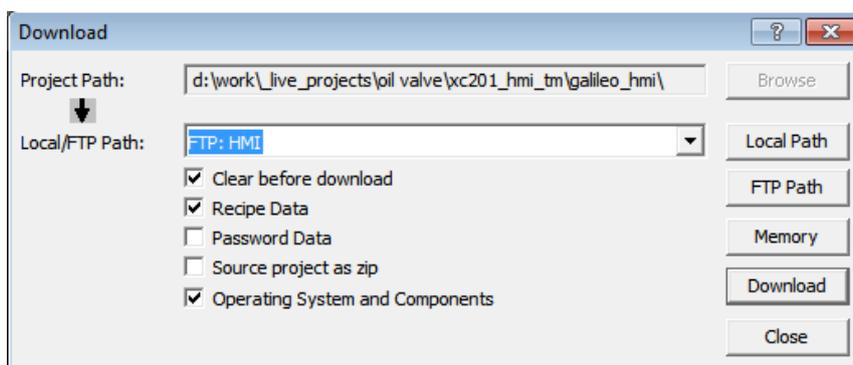


Рис. 61. Подтверждение FTP загрузки проекта в HMI панель

13. На панели оператора включите RS-сервер Start/Communication/RS-Server.

2.6. Создание графического SCADA интерфейса управления в среде TRACE MODE

В режиме исполнения проекта SCADA системы будет установлен двусторонний обмен данными с OPC сервером. При этом сигналы будут поступать по следующим направлениям:

- Нажатие какой-либо SCADA кнопки → передача сигнала («TRUE» в случае дискретного сигнала) в выходной канал → передача сигнала в выходной OPC источник → воздействие сигнала на OPC тег символического файла → передача значения сигнала через тег символического файла в контроллер → воздействие сигнала на переменную в контроллере.
- Изменение значения переменной в контроллере (например в «TRUE» в случае дискретного сигнала) → передача значения переменной через тег символического файла в OPC сервер → передача сигнала во входной OPC приёмник → передача сигнала во входной канал → воздействие сигнала на графический элемент SCADA мнемосхемы (например светящуюся лампу).

1. Откройте Trace Mode 6 и создайте новый проект Файл/Новый (рис. 62).

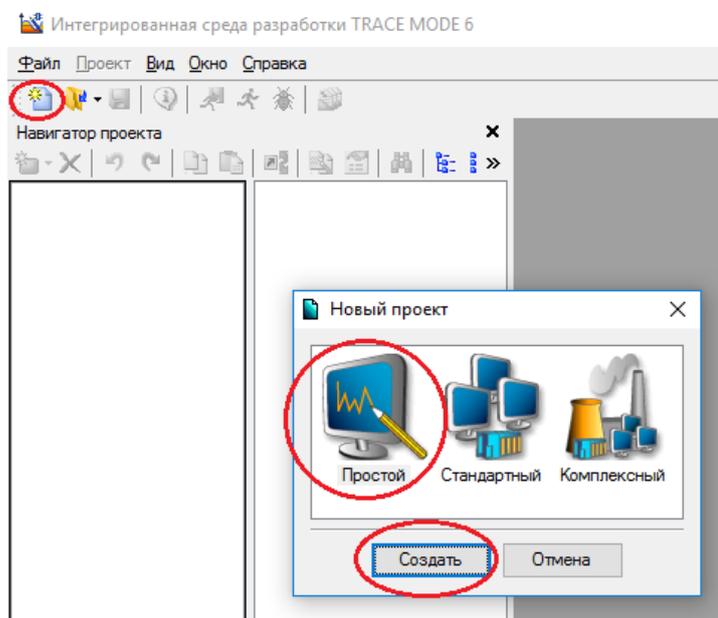


Рис. 62. Создание нового проекта Trace Mode

Для начала обратите внимание на внешний вид мнемосхемы в режиме исполнения проекта, которая в дальнейшем должна получиться путём рисования графических элементов (рис. 63).

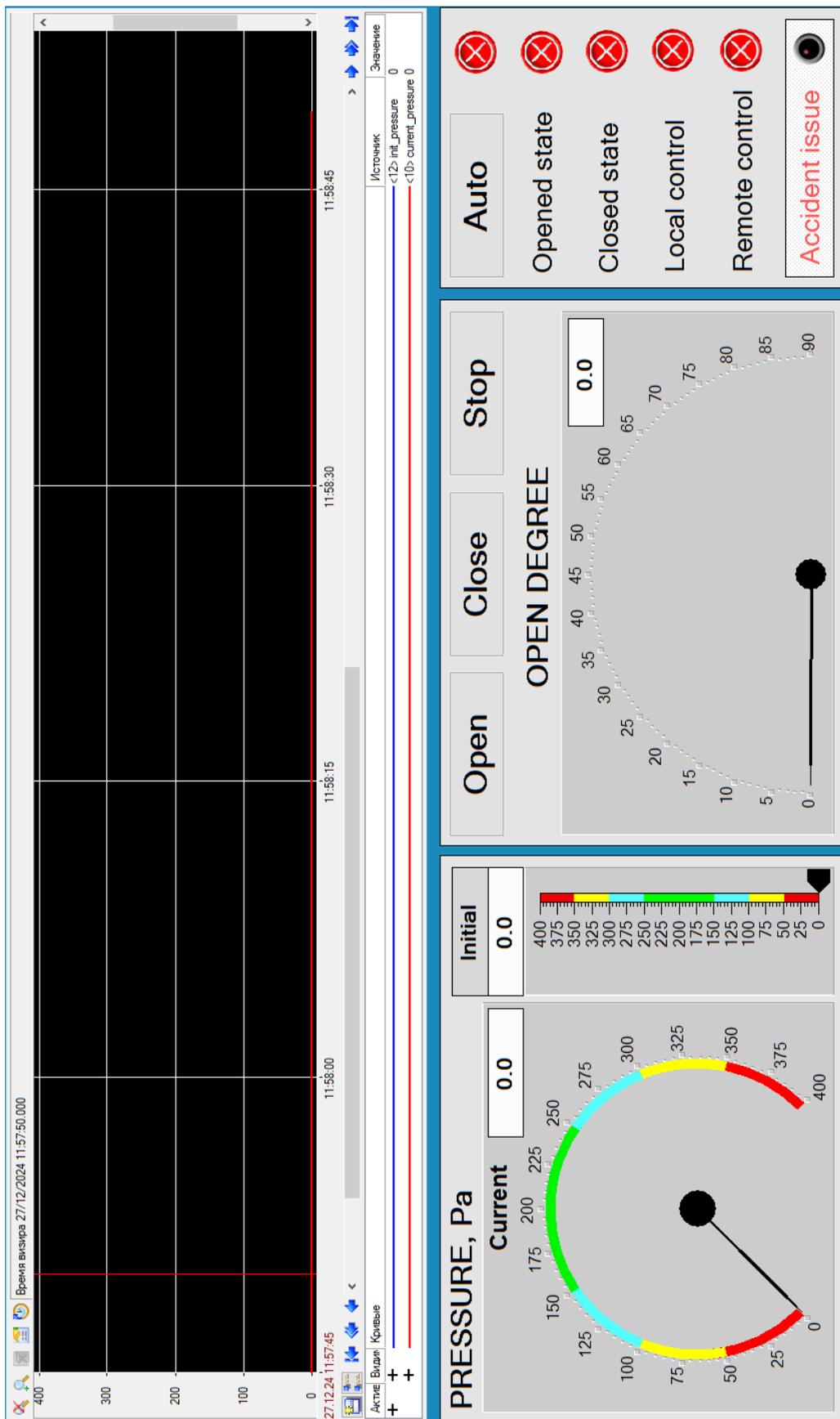


Рис. 63. Мнемосхема с элементами управления

2. *Добавьте каналы.* Для этого нужно добавить как входные, так и выходные каналы с типами HEX16 и FLOAT (рис. 64, 65). Примеры настроек каналов open_button и opened приведены на рис. 66, 67.

3. *Добавьте OPC группу и OPC сервер в данной группе (рис. 68, 69).*

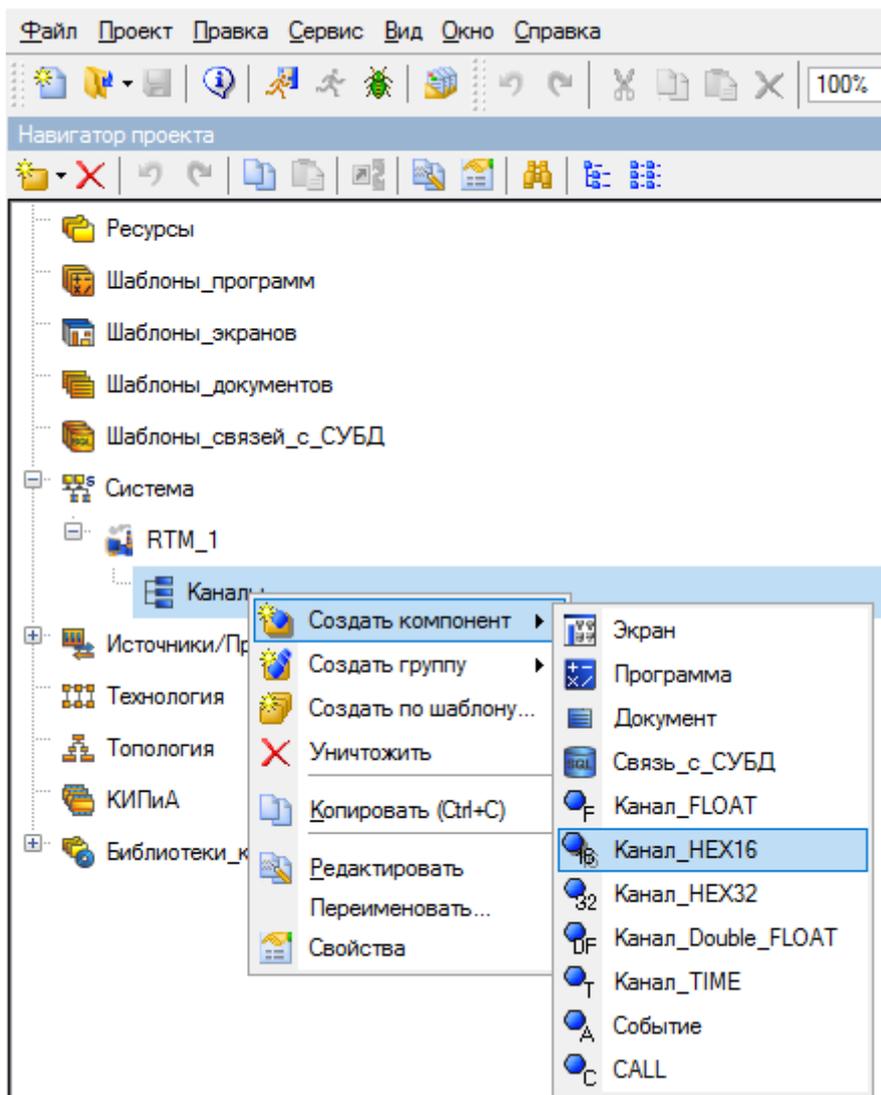


Рис. 64. Добавление нового канала

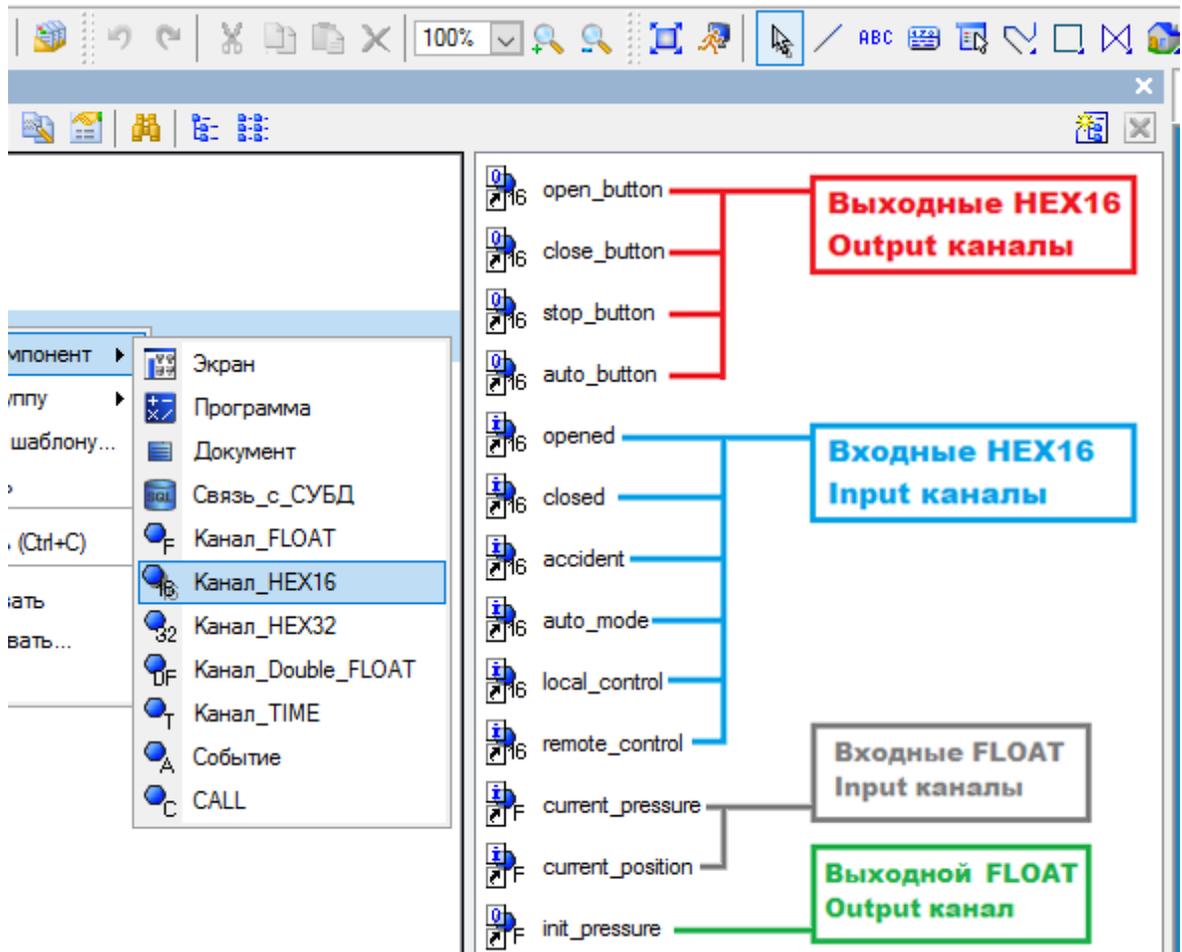


Рис. 65. Список всех необходимых каналов

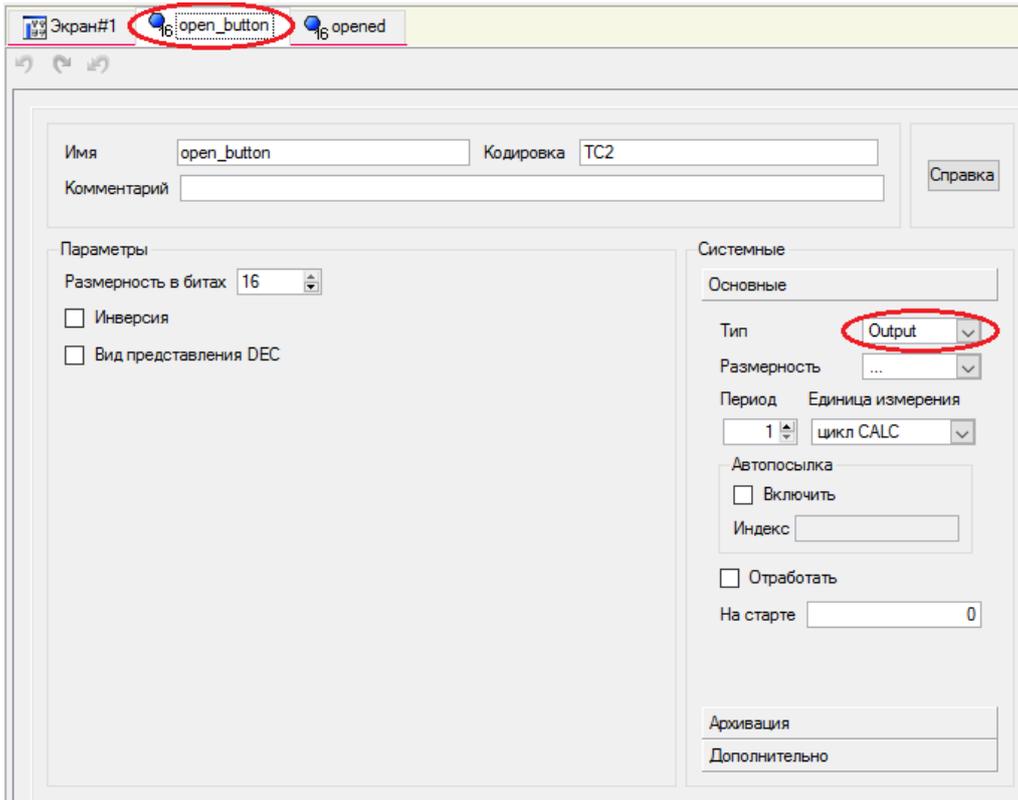


Рис. 66. Пример настроек выходного канала open_button

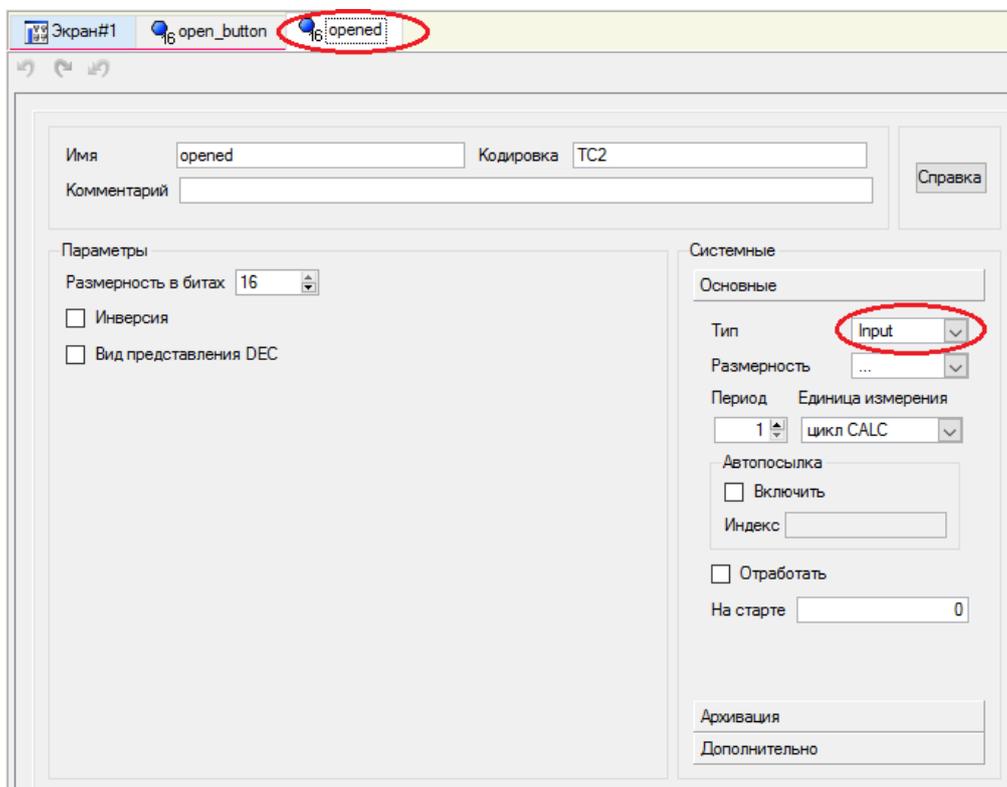


Рис. 67. Пример настроек входного канала opened

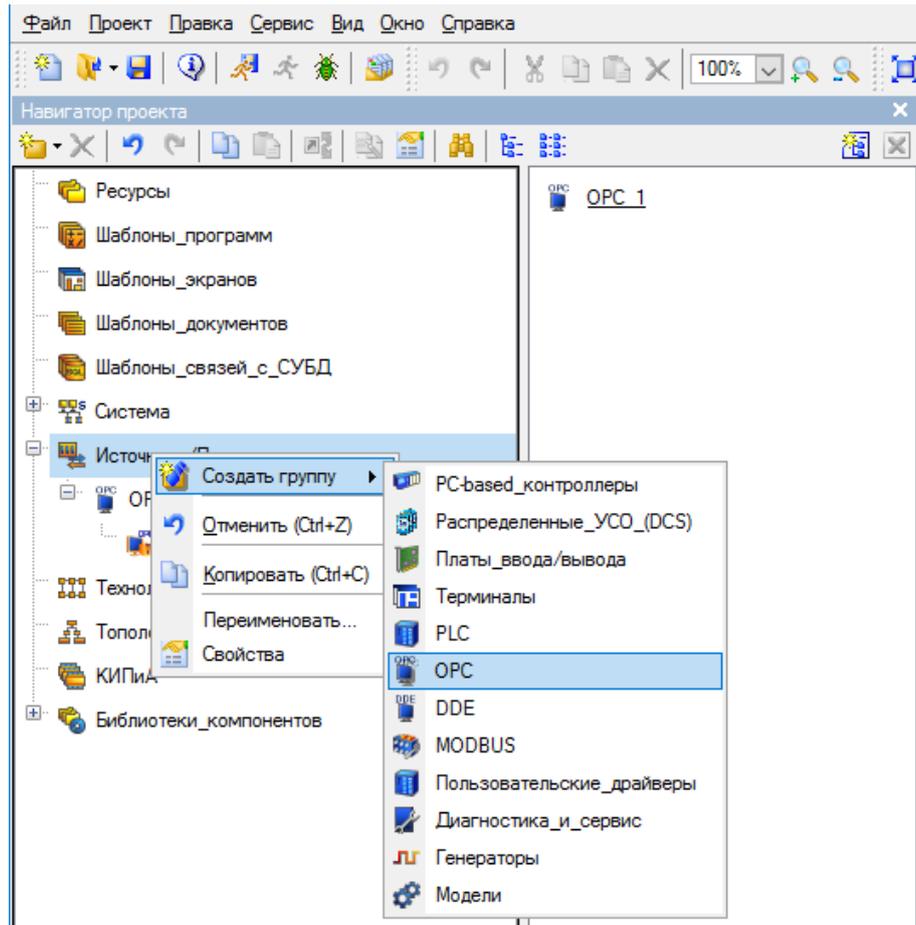


Рис. 68. Добавление OPC группы

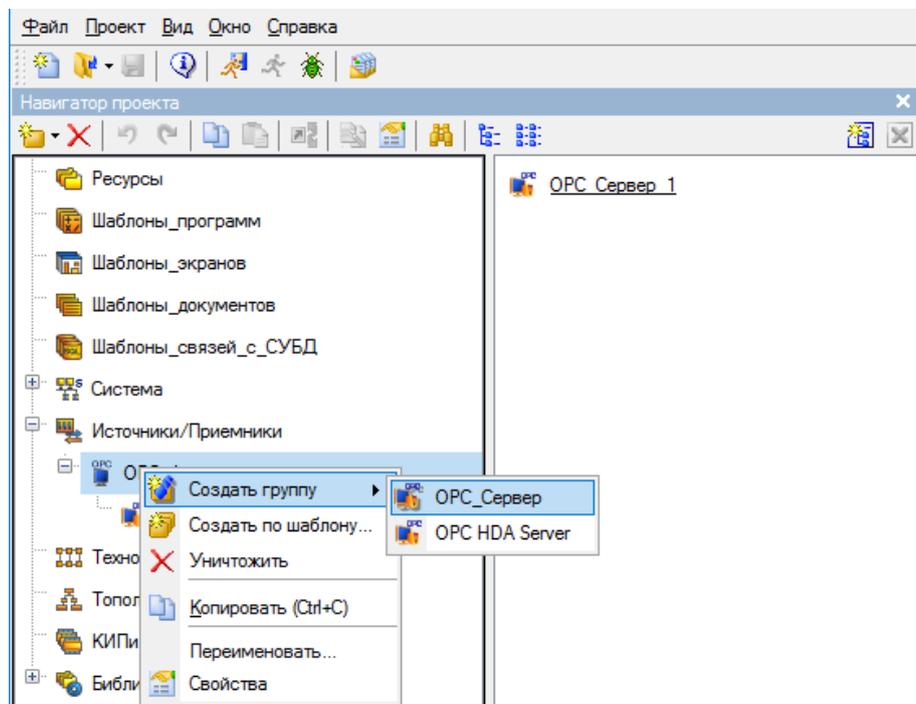


Рис. 69. Добавление OPC сервера

4. *Добавьте OPC источники и приёмники (рис. 70).* Для этого нужно добавить как дискретные, так и аналоговые источники/приёмники сигналов в OPC сервер. На рис. 71 приведён полный список источников/приёмников, для удобного восприятия их имена совпадают с именами ранее добавленных каналов (имя можно задать в окне редактирования после двойного клика клавишей мыши по источнику/приёмнику). К каждому источнику/приёмнику в дальнейшем будет привязан конкретный канал.

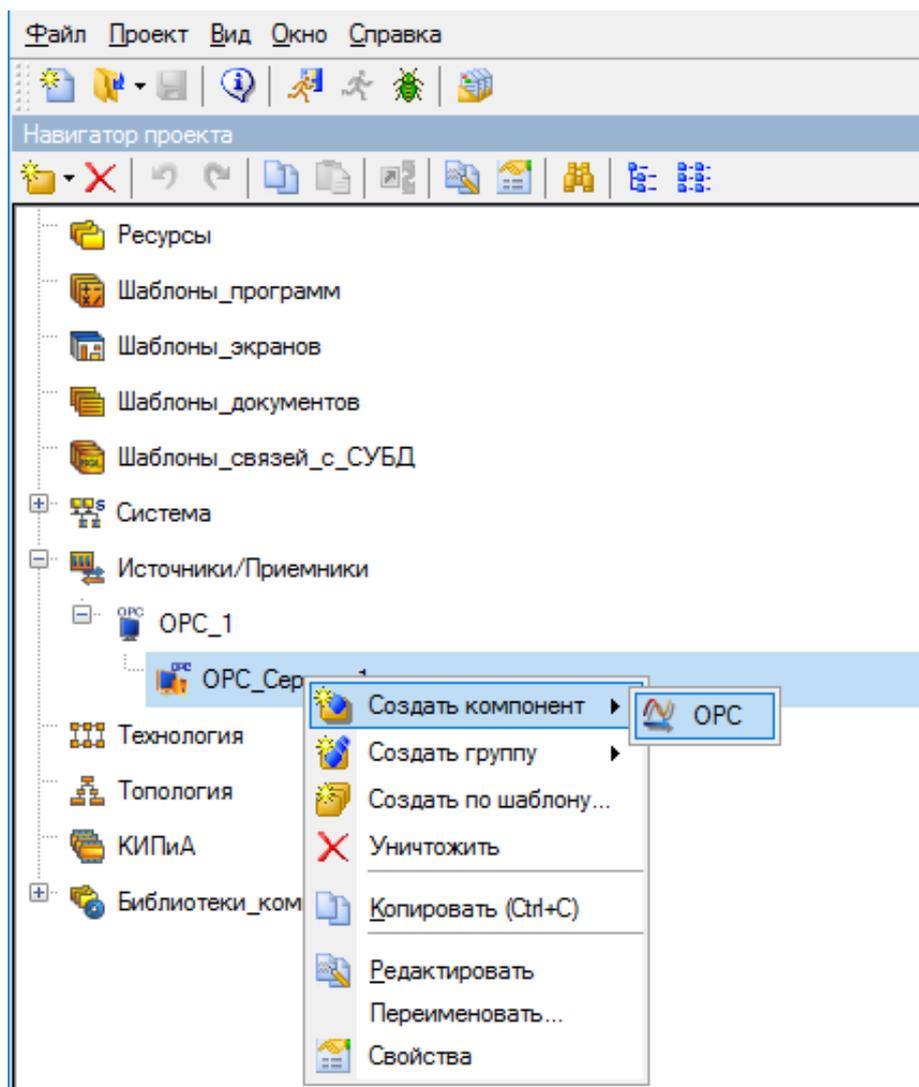


Рис. 70. Добавление OPC источника/приёмника

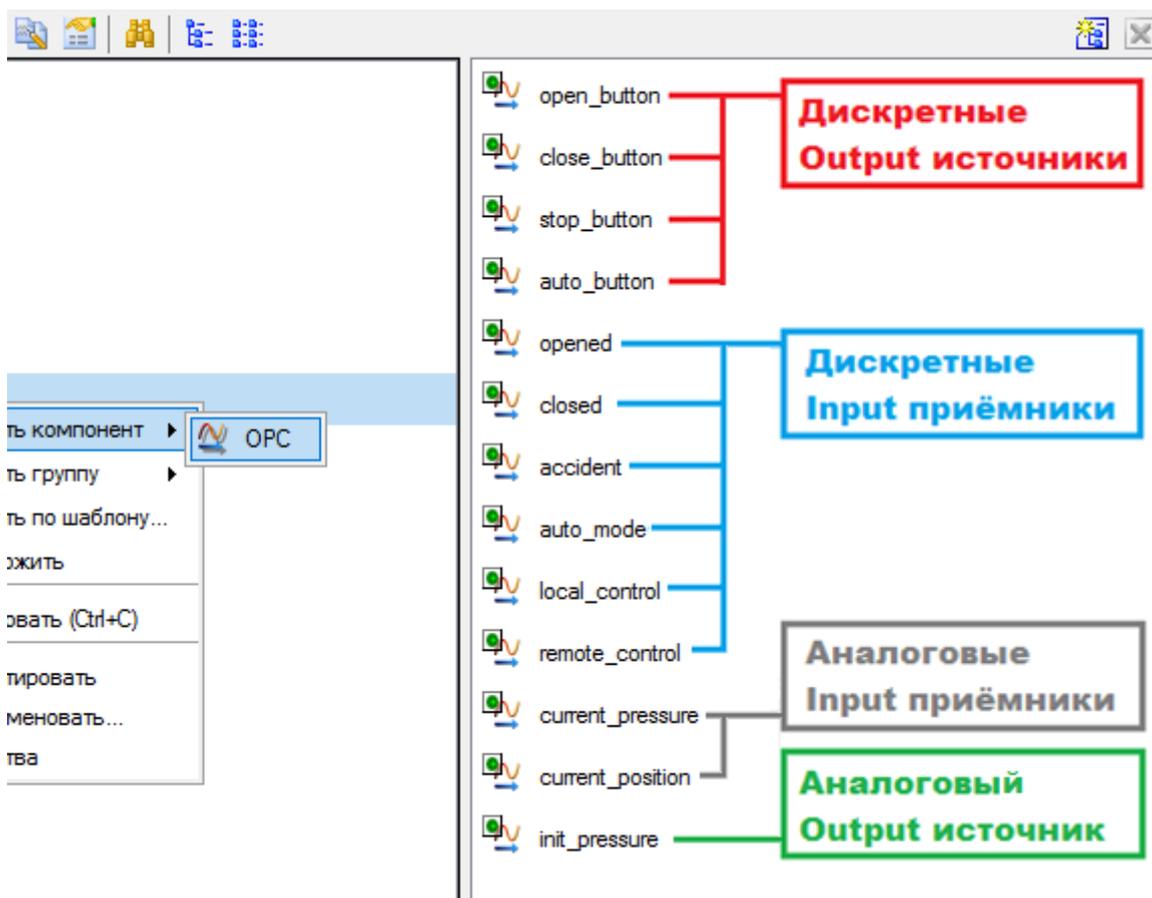


Рис. 71. Полный список источников/приёмников OPC сервера

Создавая нужный источник или приёмник, задайте ему следующие настройки:

- Имя;
- Сервер – CoDeSys.OPC.02 (выбирается кнопкой «Обзор» с последующим выбором тега символического файла, рис. 72);
- Идентификатор – устанавливается автоматически после выбора тега символического файла в OPC сервере.
- Направление (Output или Input, ориентируясь на рис. 71).
- Формат (Дискрет или Аналог, ориентируясь на рис. 71).

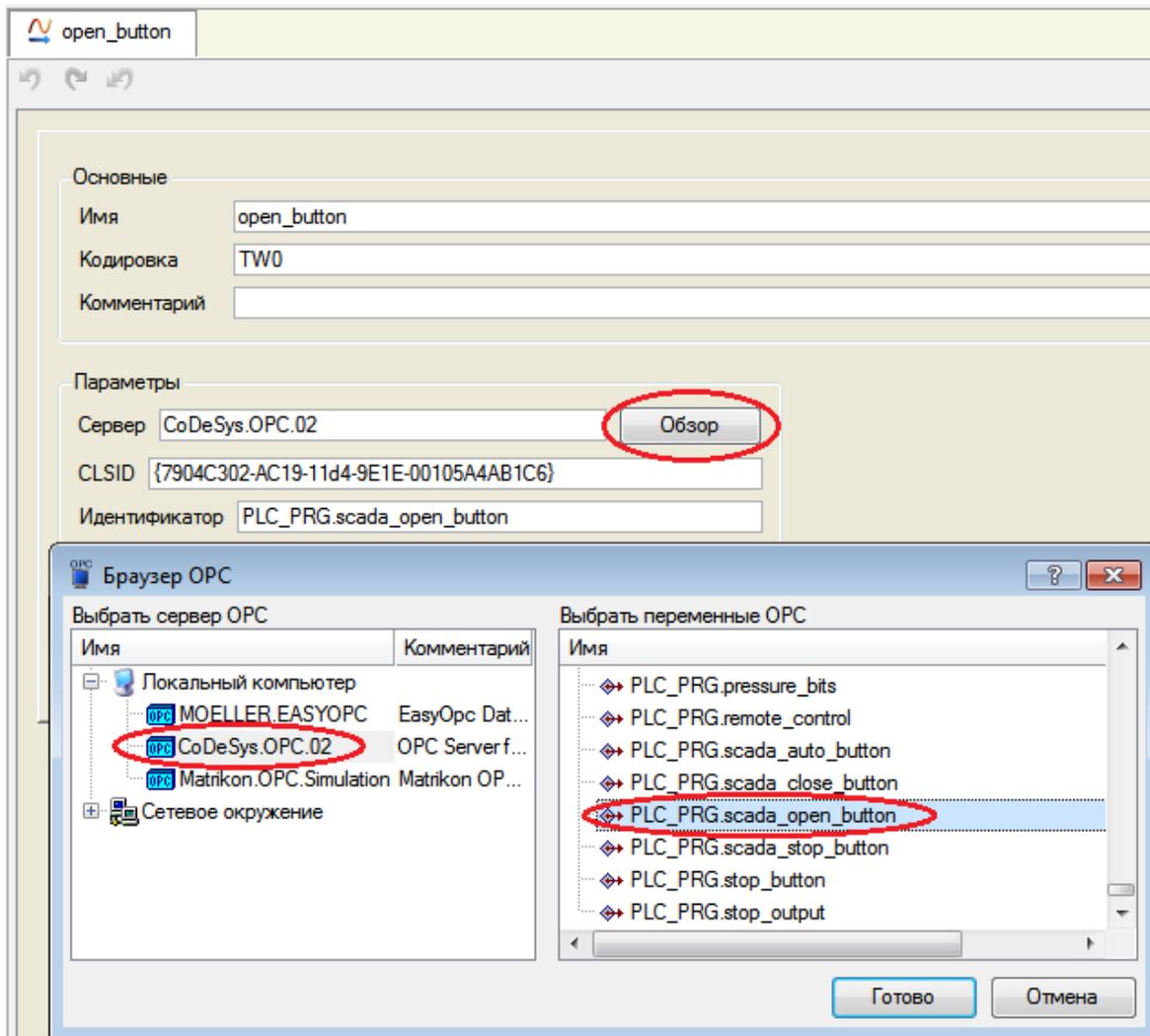


Рис. 72. Выбор тега символического файла в OPC сервере

ВНИМАНИЕ! Имя OPC тега в символическом файле, к которому привязывается источник или приёмник, может отличаться от имени самого источника или приёмника. Здесь главное понять к какому тегу нужно привязаться на уровне здравого смысла и понимания проекта.

Для удобства понимания, можно воспользоваться таблицей 3 для привязки OPC источников/приёмников SCADA системы к тегам символического файла.

Таблица 3

Привязка OPC источников/приёмников к тегам символьного файла

Имя OPC источника/приёмника в SCADA системе	Направление сигнала	Имя тега в символьном OPC файле
open_button	→	scada_open_button
close_button	→	scada_close_button
stop_button	→	scada_stop_button
auto_button	→	scada_auto_button
opened	←	opened
closed	←	closed
accident	←	accident
auto_mode	←	auto_mode
local_control	←	local_control
remote_control	←	remote_control
current_pressure	←	current_pressure
current_position	←	current_position
init_pressure	→	init_pressure

Пример настроек дискретного OPC источника «open_button» приведён на рис. 73, а пример дискретного OPC приёмника «opened» на рис. 74.

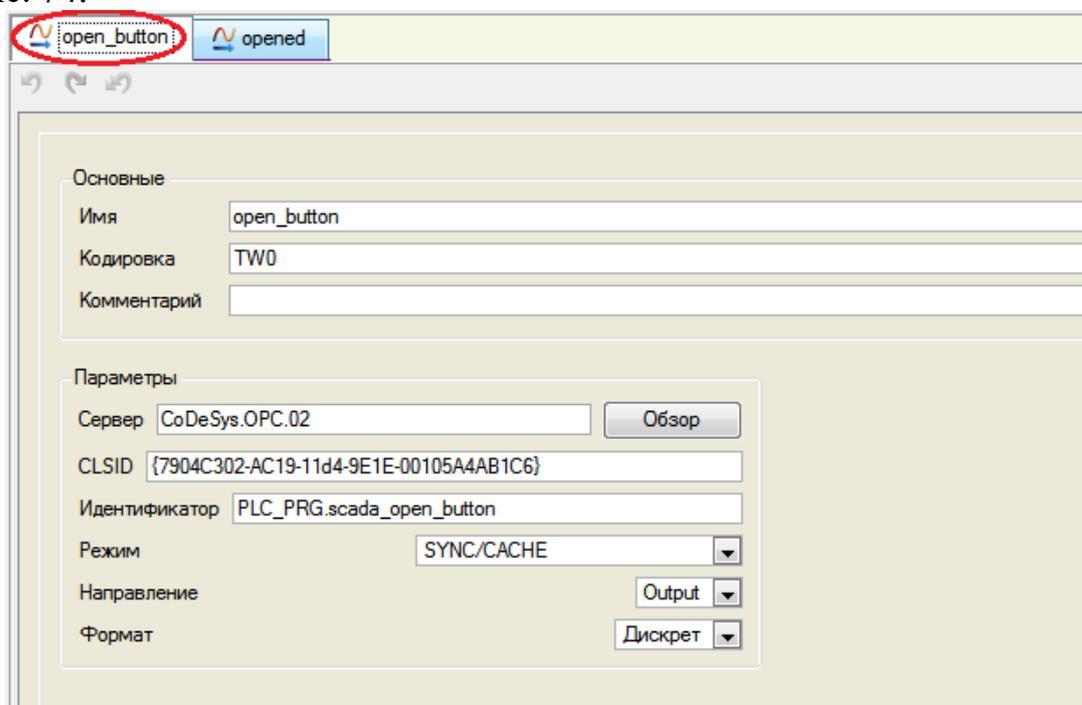


Рис. 73. Настройки OPC источника «open_button»

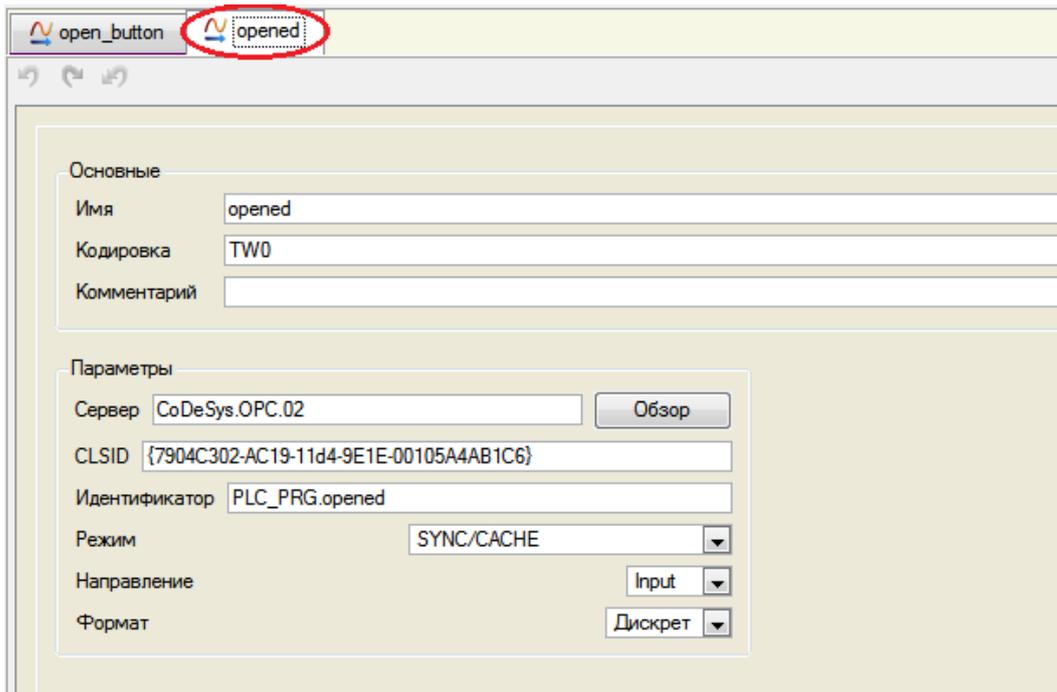


Рис. 74. Настройки OPC приёмника «opened»

5. Откройте свойства каждого канала и сделайте его привязку к одноимённым источникам/приёмникам (рис. 75). Пример привязки выходного канала «open_button» к OPC источнику «open_button» показан на рис. 76.

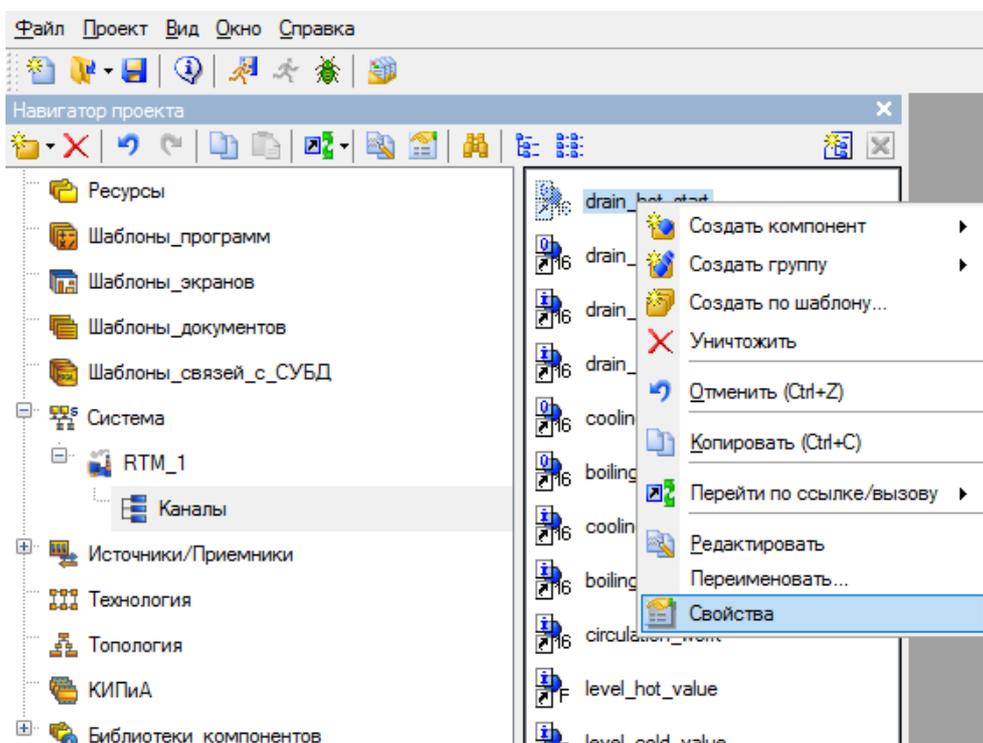


Рис. 75. Открытие свойств канала

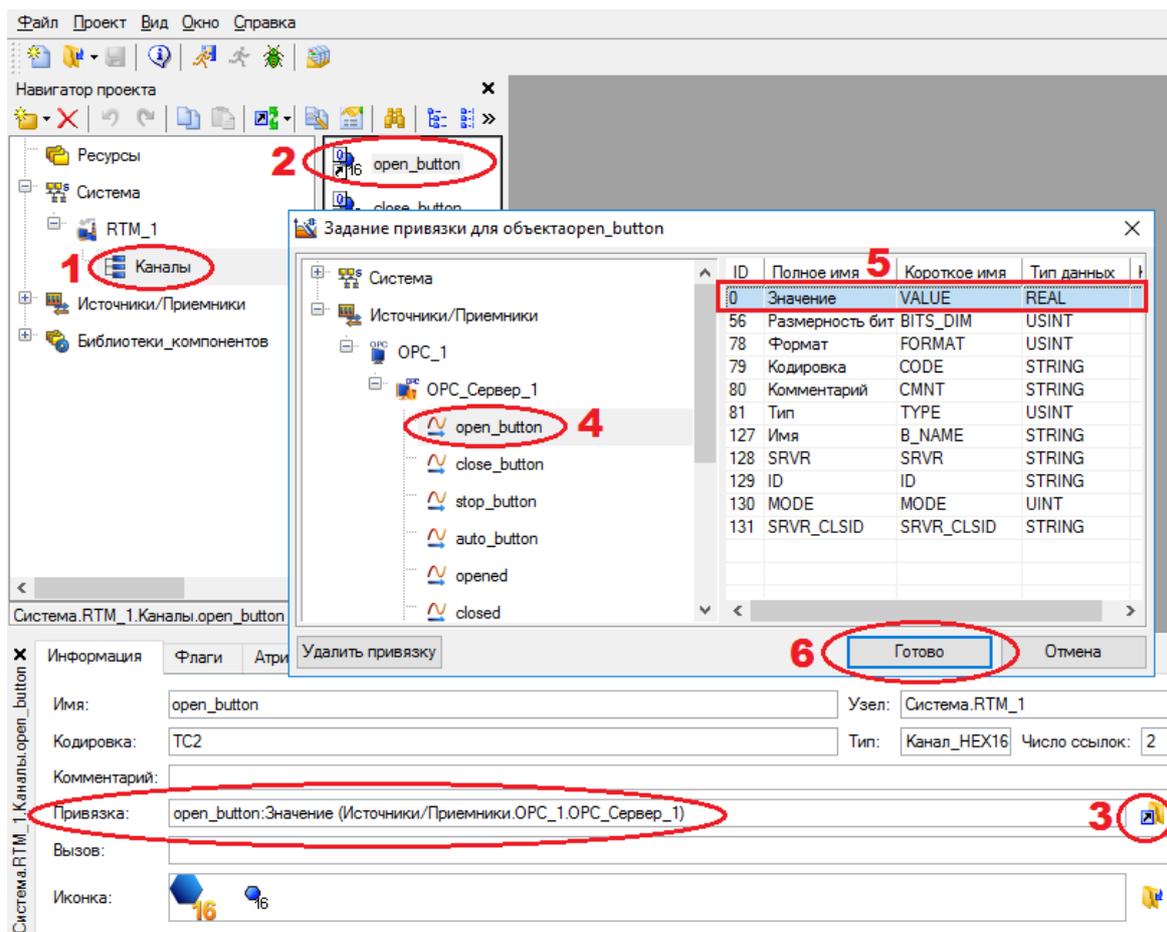


Рис. 76. Привязка канала к OPC источнику

6. Нарисуйте графические элементы (рис. 63), используя инструменты рисования (рис. 77).



Рис. 77. Элементы графического рисования

7. Сделайте настройки и привязки графических элементов к каналам. Пример настроек и привязки кнопки «Open» приведён на рис. 77, 78. Чтобы задать привязку, нужно кликнуть левой клавишей мыши по «Результат» и «Источник», далее откроется окно (рис. 79), в котором нужно добавить новую строку в таблице для текущей привязки, задать тип, тип данных и открыть окно привязки (рис. 80). **ВНИМАНИЕ!** В таблице (рис. 79) целый ряд привязок для отдельных графических элементов. Для конкретного графического элемента строка, после её добавления и выбора, будет подсвечиваться синим фоном. Т.е. это некая карта привязок графических элементов. **Обратите внимание** – в

таблице в колонке привязки может быть как входное значение, так и реальное значение, в зависимости от направления сигналов.

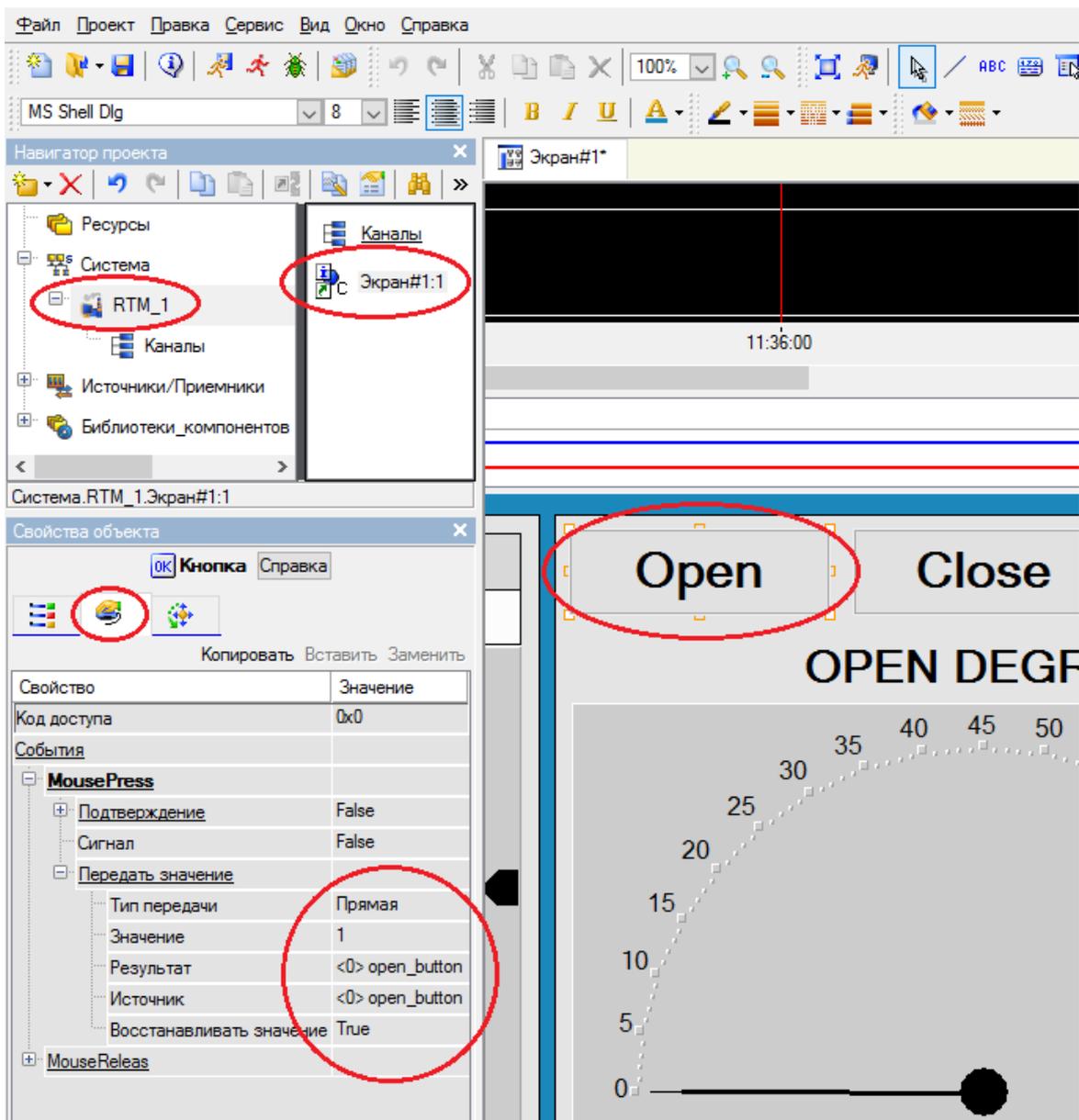


Рис. 78. Настройки кнопки «Open»

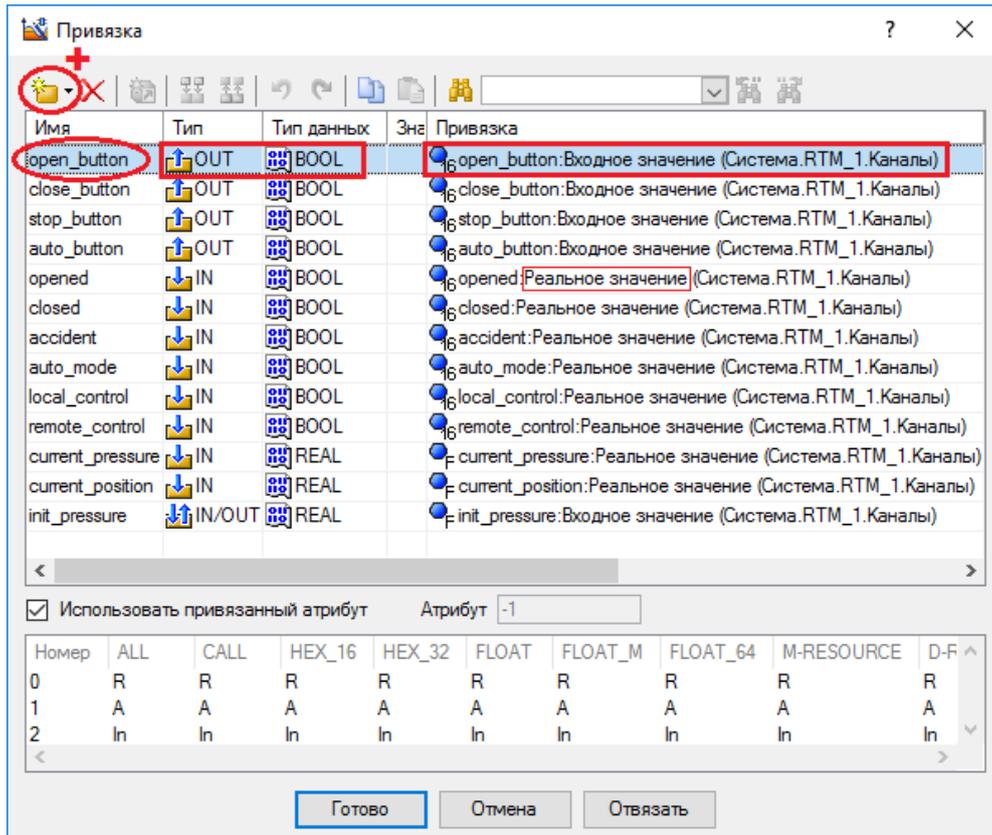


Рис. 79. Создание привязки графического элемента к каналу

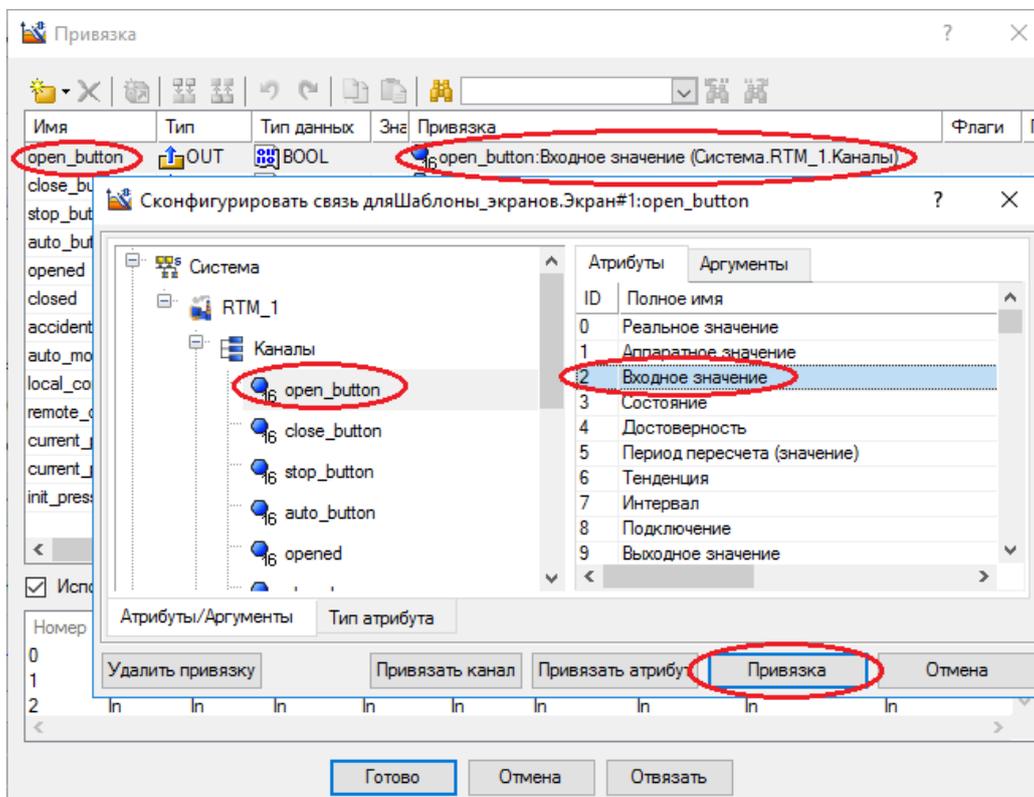


Рис. 80. Настройка привязки к каналу

На рис. 81-93 приведены примеры свойств различных графических элементов мнемосхемы.

Свойства объекта

Стрелочный прибор Справка

Копировать Вставить Заменить

Свойство	Значение
Отображаемая величина (<11> current_position)	0
Привязка	<11> current_position
Положение	По центру
Угол разворота	90
Заголовок	
Полоса	False
Шкала	True
Цвет текста	
Уровень 1	
Использовать	True
Число делений	9
Длина штриха	5
Показать значения	True
Шрифт	MS Shell Dlg,12
Десятичные знаки	0
Уровень 2	
Использовать	True
Число делений	2
Длина штриха	5
Показать значения	True
Шрифт	Arial,12
Десятичные знаки	0
Уровень 3	
Использовать	True
Число делений	5
Длина штриха	2
Показать значения	False
Шрифт	Arial,8
Десятичные знаки	1
Указатель	
Единицы	

22:11:00

Open Close

OPEN DEC

0 5 10 15 20 25 30 35 40 45

Рис. 81. Свойства стрелочного прибора показаний углового положения

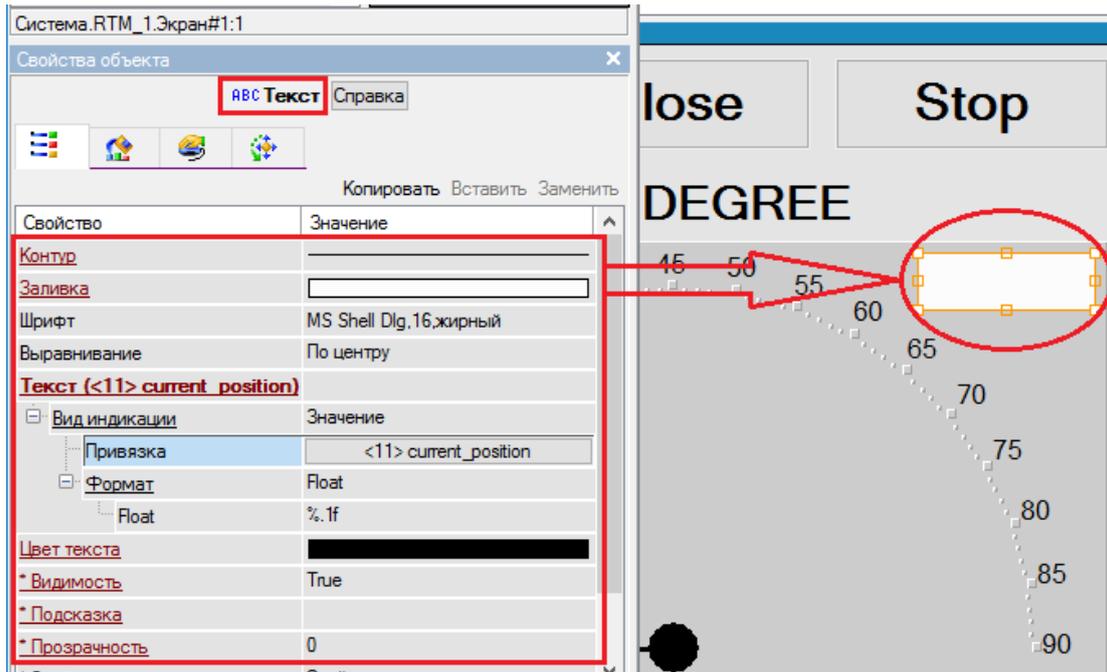


Рис. 82. Свойства текстового указателя углового положения

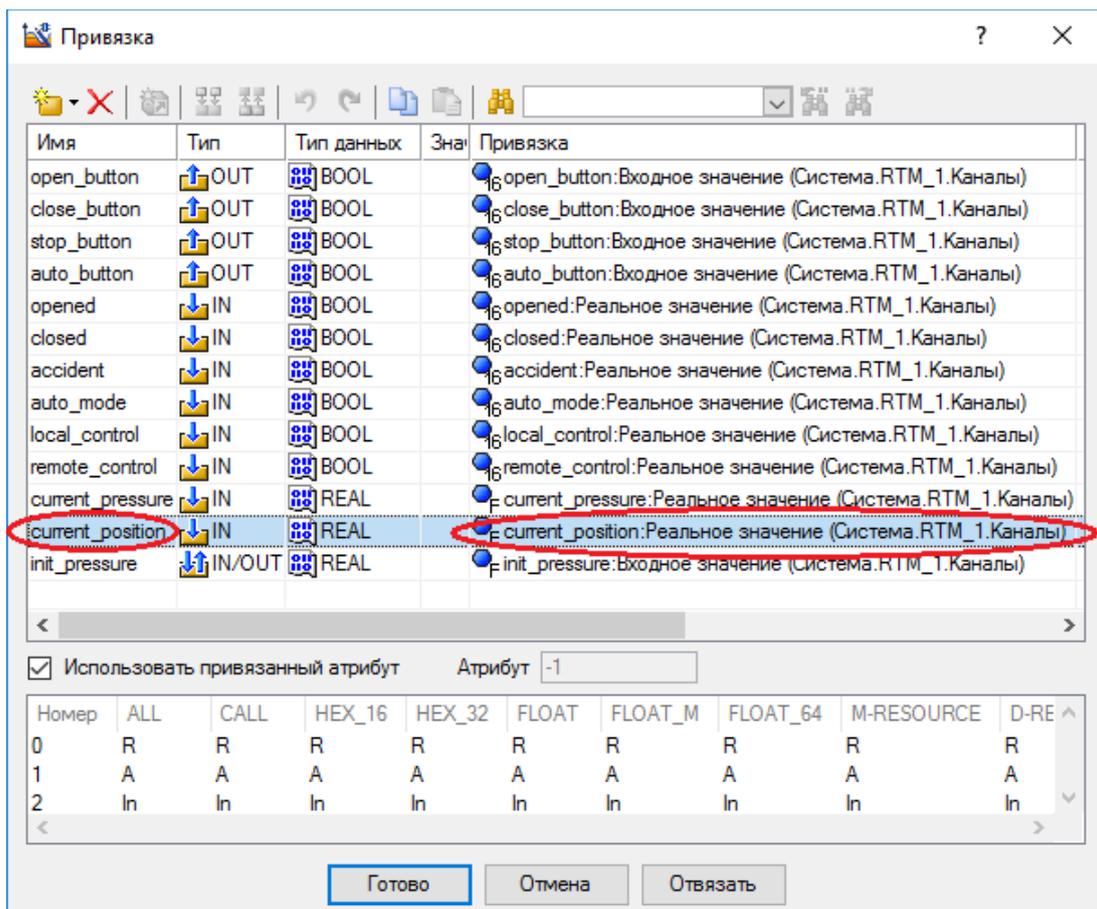


Рис. 83. Привязка стрелочного прибора и текстового указателя углового положения к каналу

Система.RTM_1.Экран#1:1

Свойства объекта

Стрелочный прибор Справка

Копировать Вставить Заменить

Свойство	Значение
Отображаемая величина (<10> current_pressure)	0
Привязка	<10> current_pressure
Положение	По центру
Угол разворота	135
Заголовок	
Полоса	True
Шкала	True
Цвет текста	
Уровень 1	
Использовать	True
Число делений	8
Длина штриха	5
Показывать значения	True
Шрифт	MS Shell Dlg,12
Десятичные знаки	0
Уровень 2	
Использовать	True
Число делений	2
Длина штриха	5
Показывать значения	True
Шрифт	Arial,12
Десятичные знаки	0
Уровень 3	
Использовать	True
Число делений	5
Длина штриха	2
Показывать значения	False
Шрифт	Arial,8
Десятичные знаки	1
Указатель	
Единицы	

08.01.25 22:10:30 22:10:45

Актив Видим Кривые

+

+

+

PRESSURE, Pa

Current

0 25 50 75 100 125 150 175 200 225 250 275 300 325 350 375 400

Рис. 84. Свойства стрелочного прибора показаний фактического давления

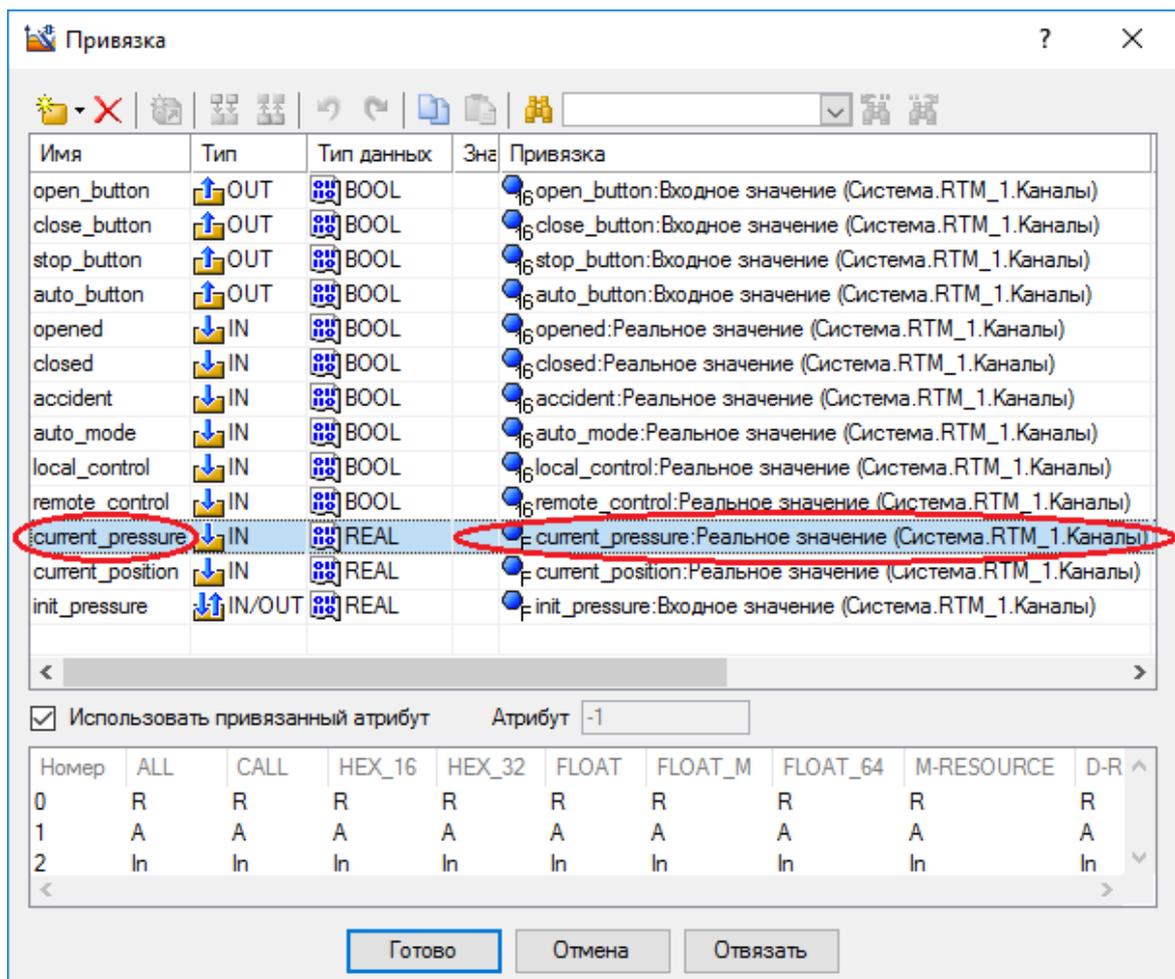


Рис. 85. Привязка стрелочного прибора показаний фактического давления к каналу

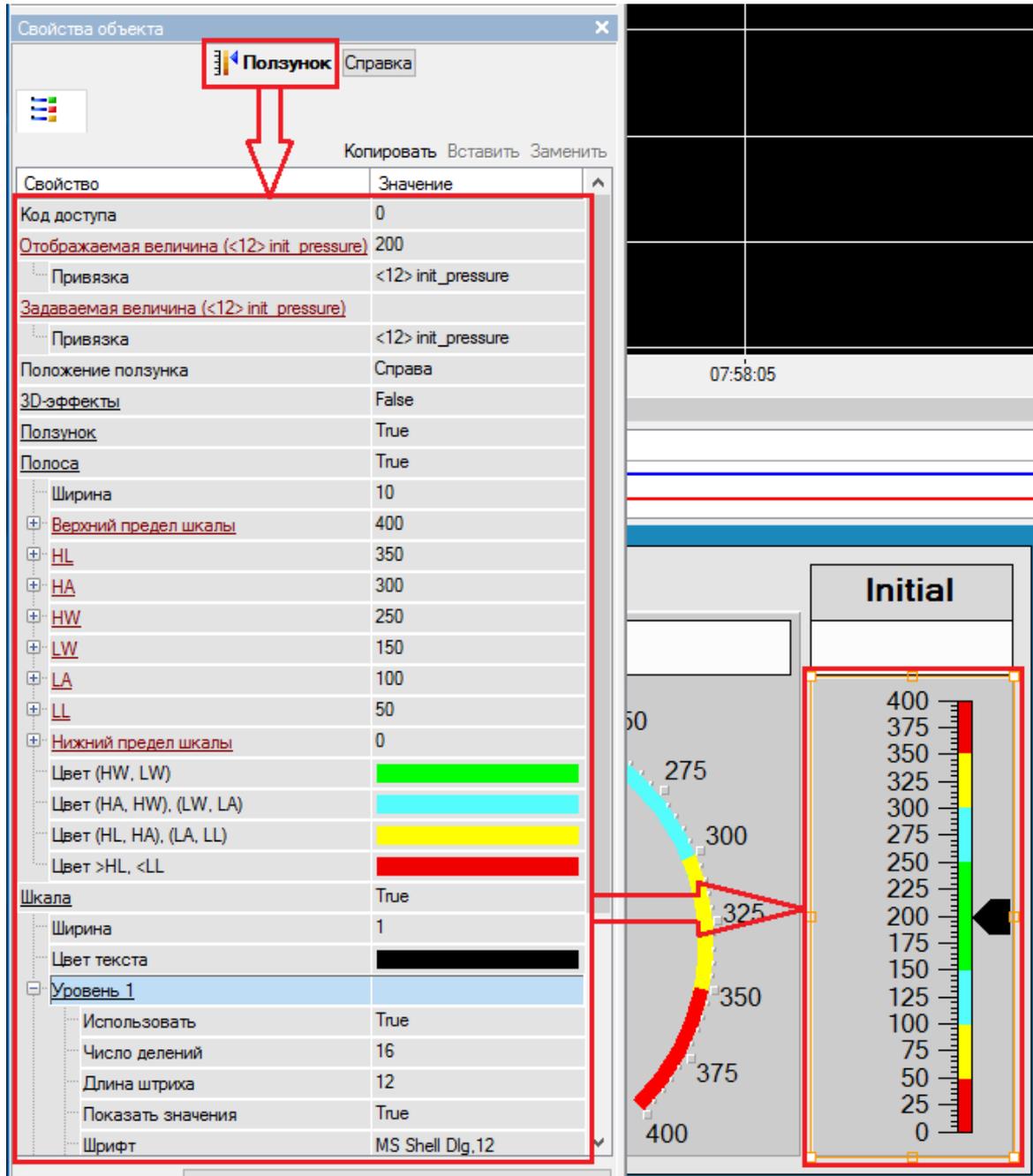


Рис. 86. Свойства задатчика начального давления

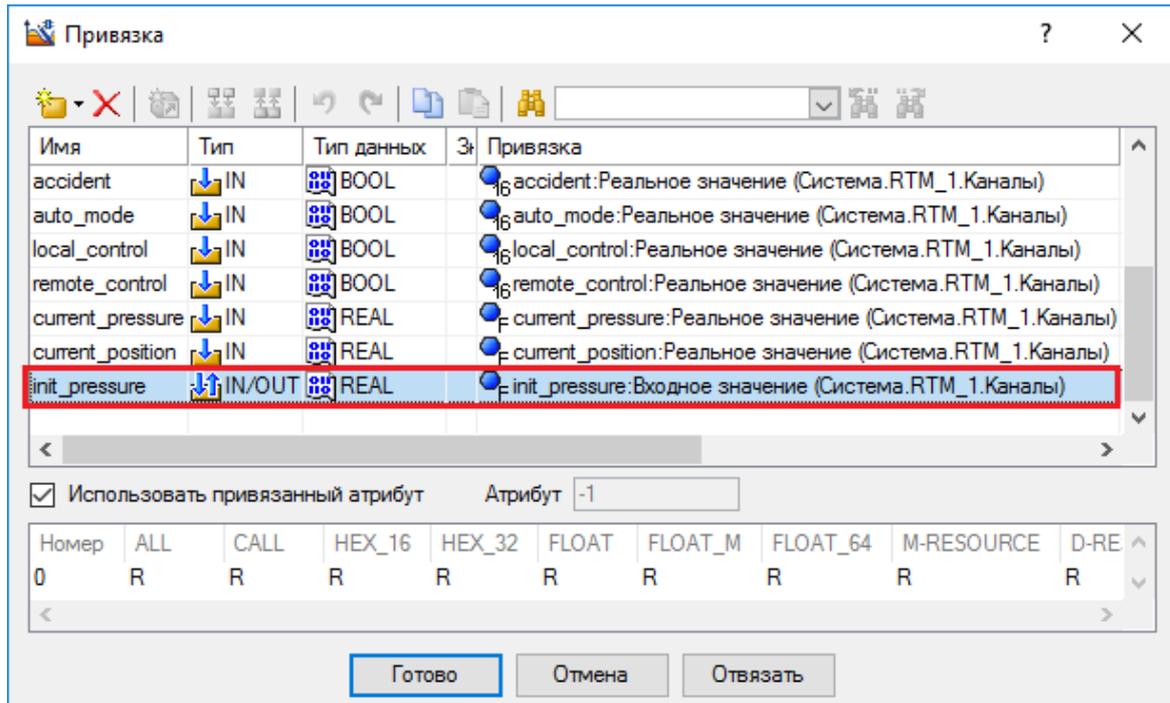


Рис. 87. Привязка задатчика начального давления к каналу

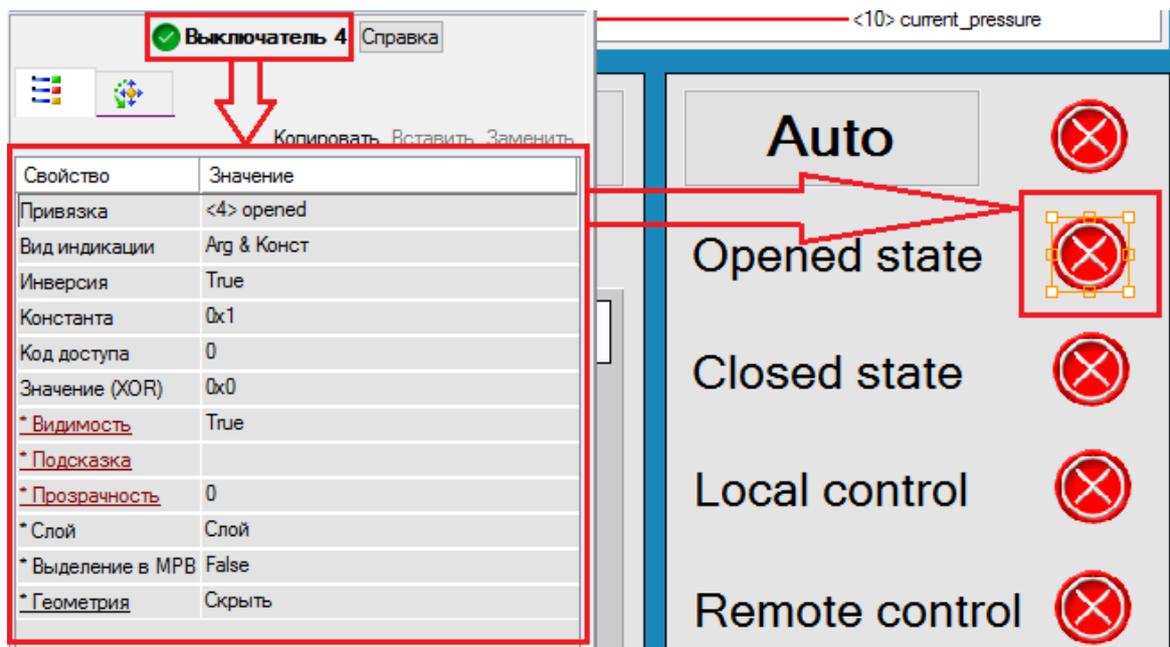


Рис. 88. Свойства сигнализатора открытого положения

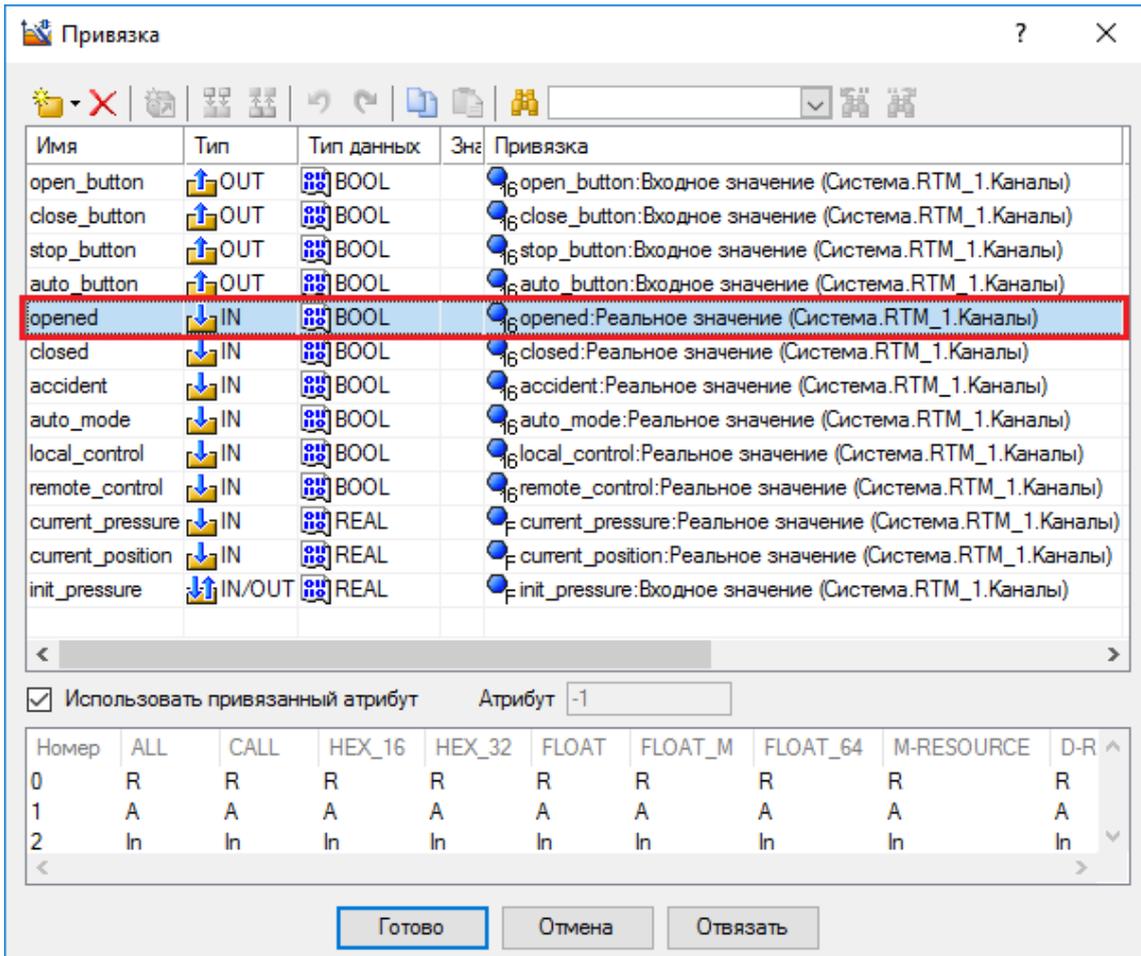


Рис. 89. Привязка сигнализатора открытого положения к каналу

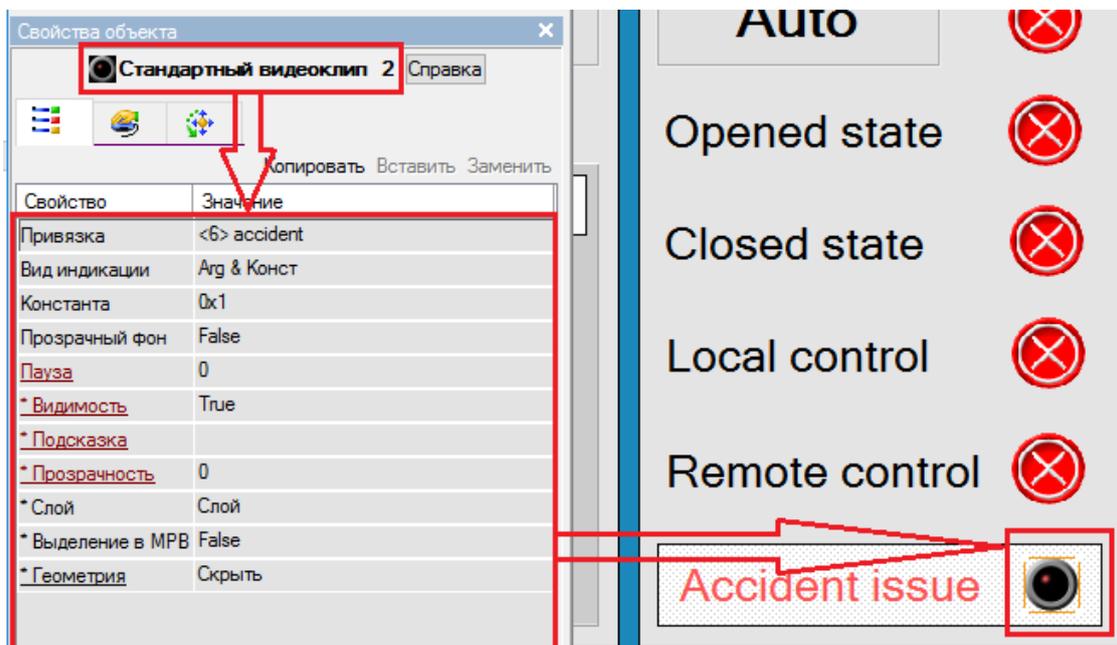


Рис. 90. Свойства сигнализатора аварии

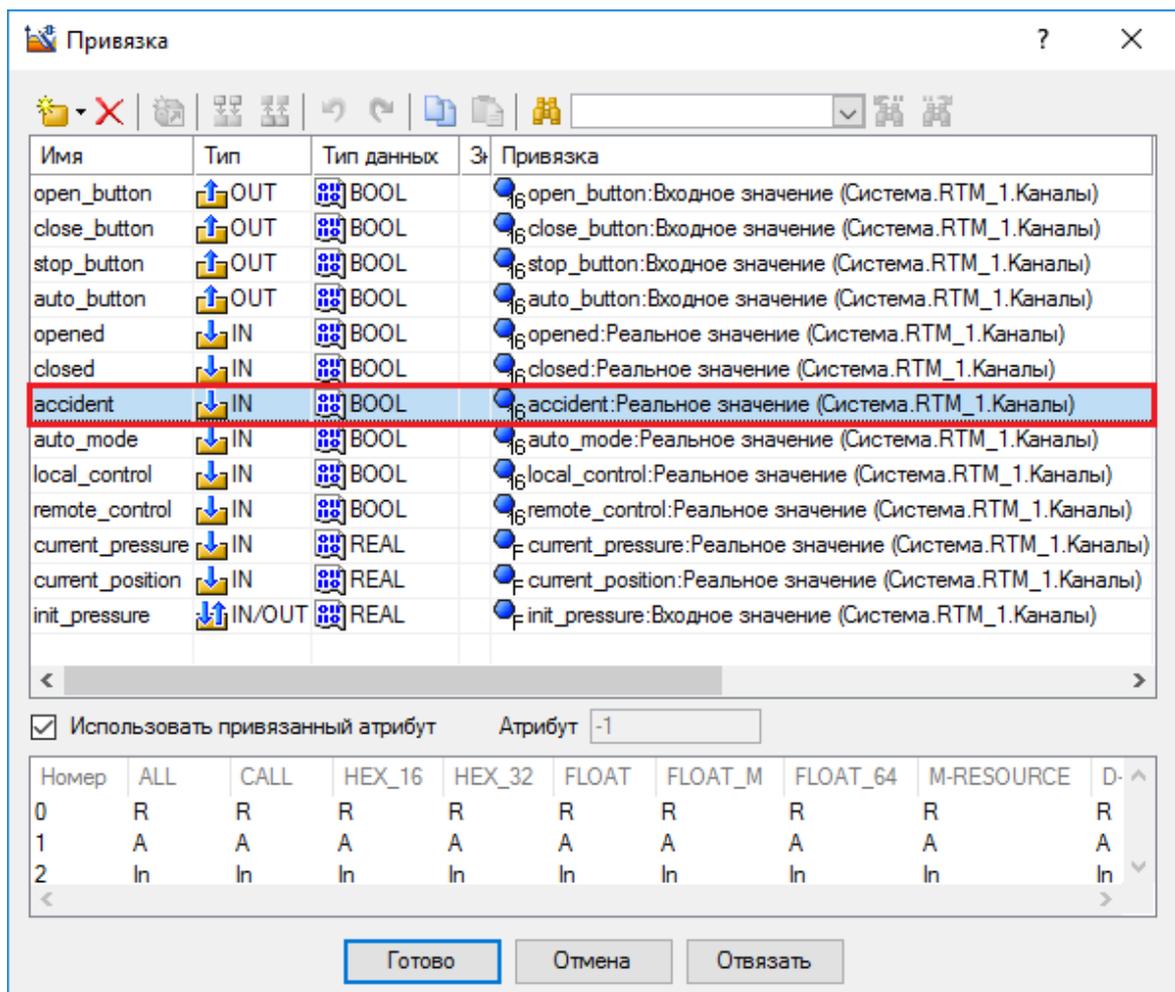


Рис. 91. Привязка сигнализатора аварии к каналу

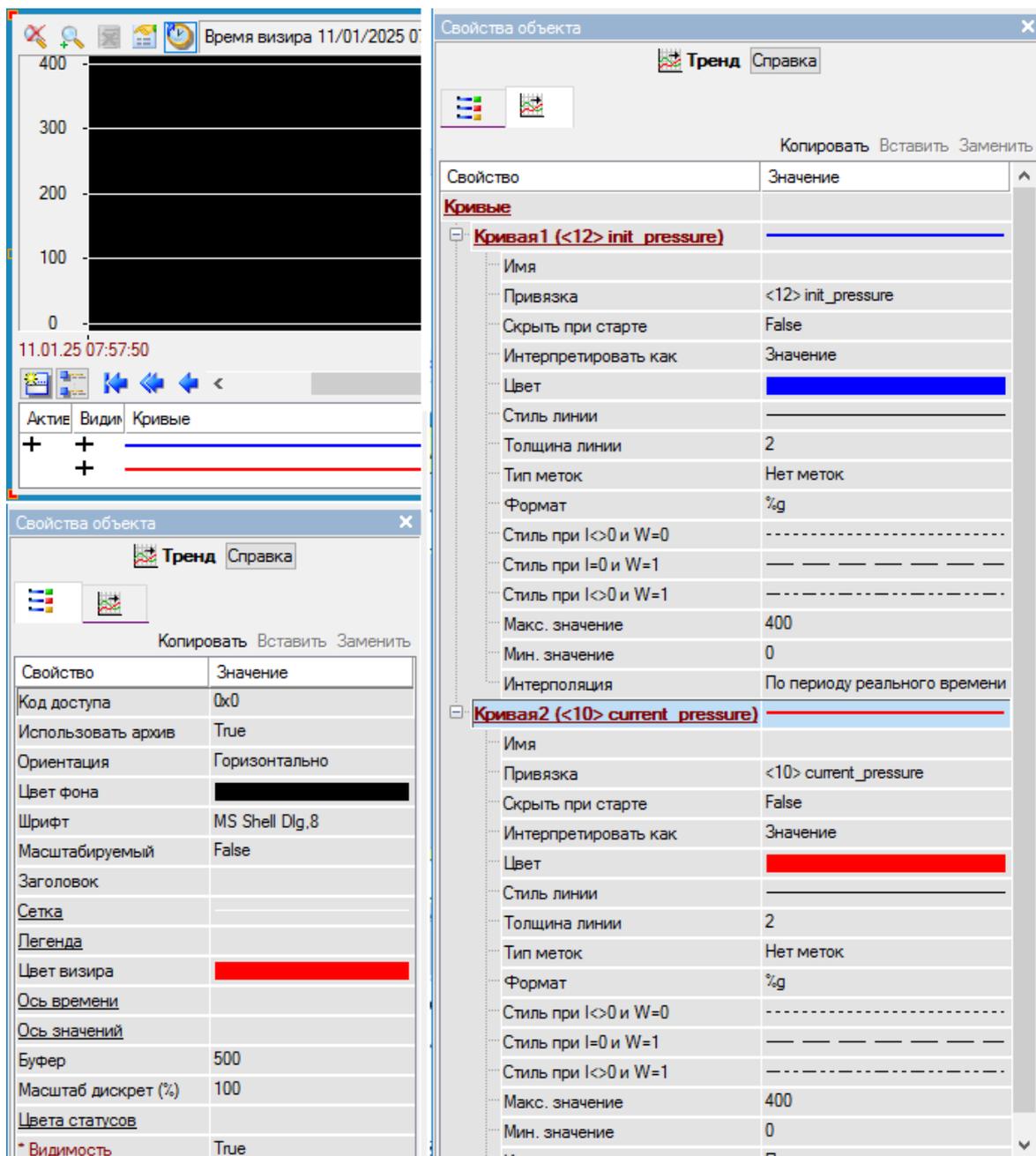


Рис. 92. Свойства элемента трендов

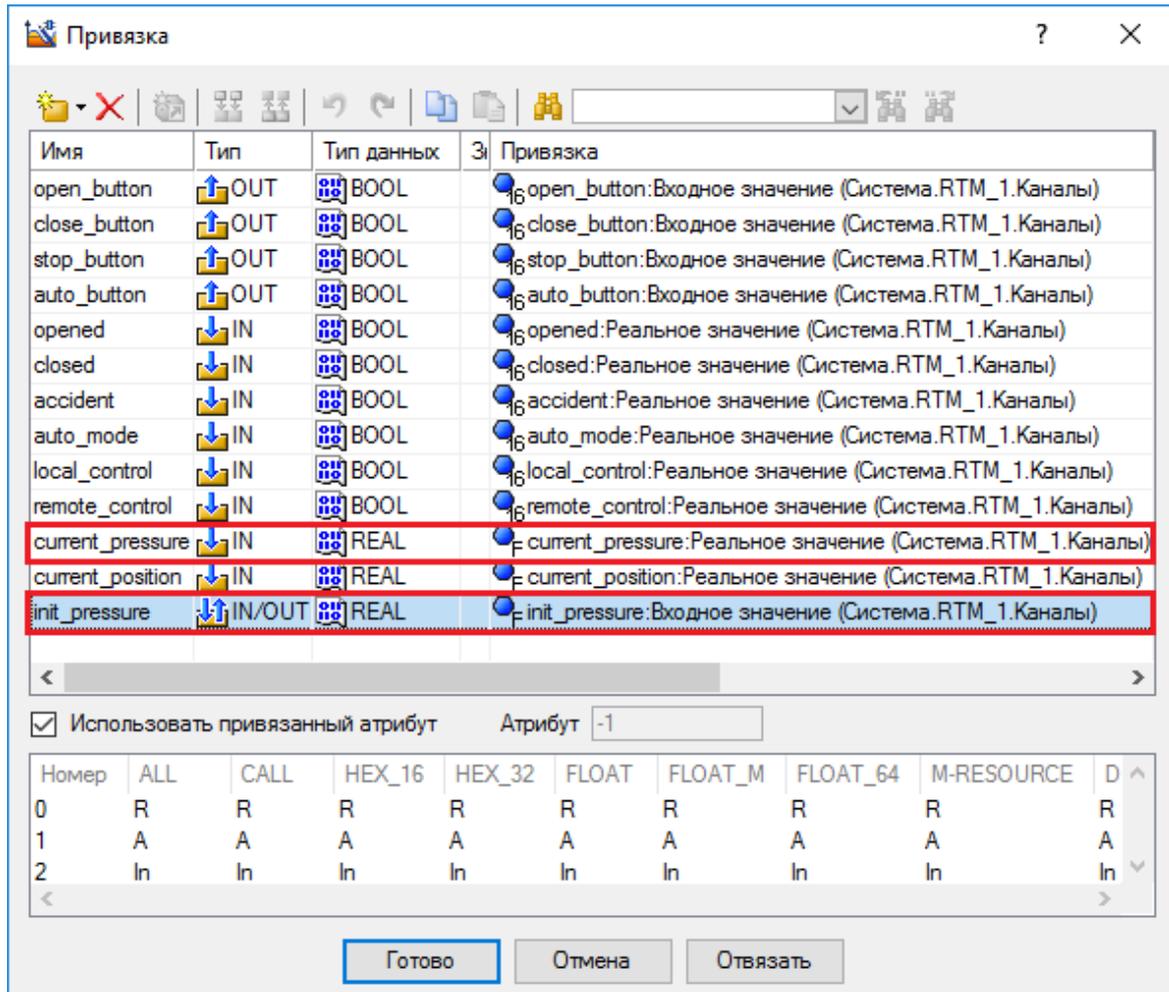


Рис. 93. Привязка кривых элемента трендов к каналам

8. Проверьте работоспособность мнемосхемы в режиме эмуляции (рис. 94).

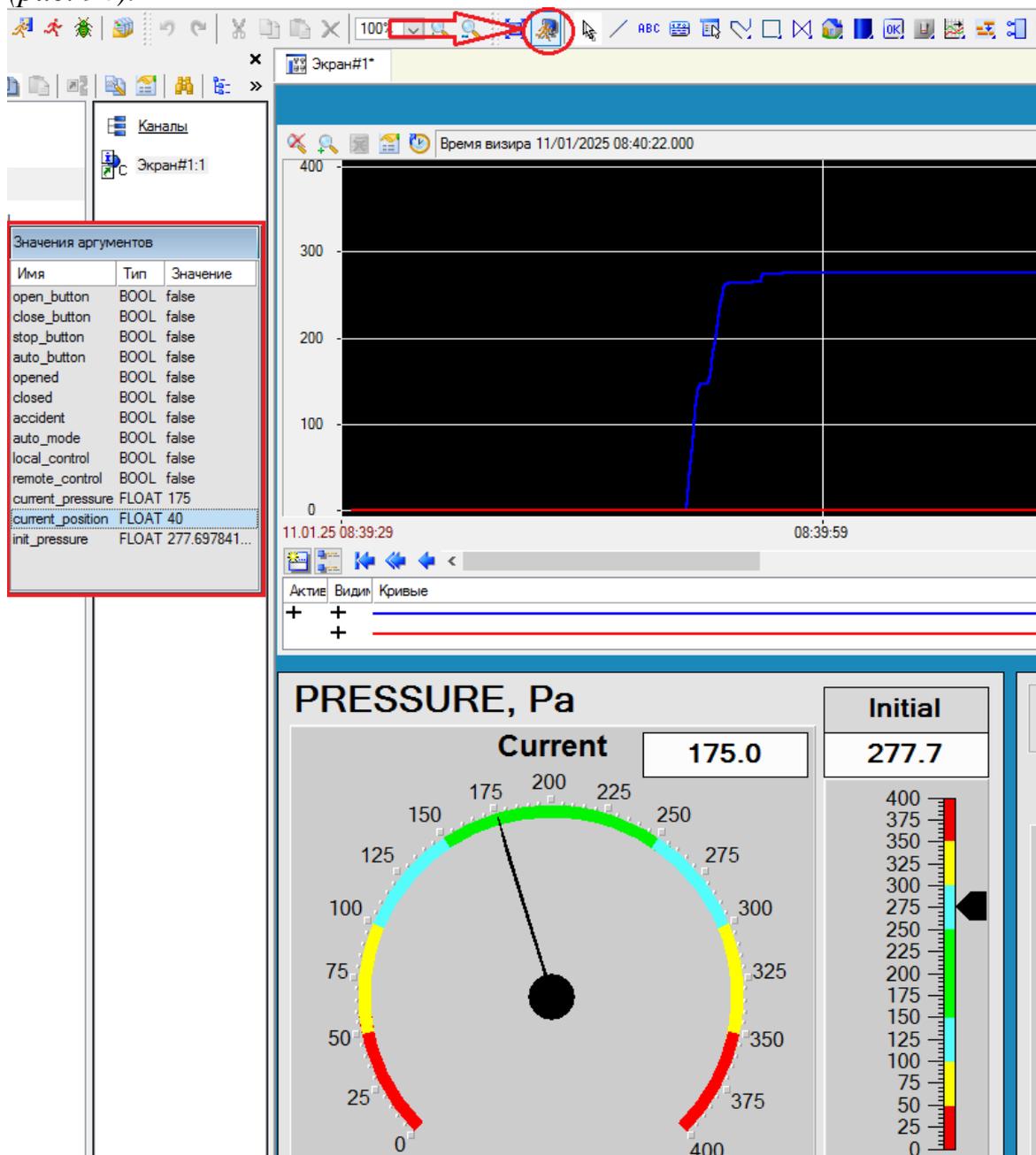


Рис. 94. Проверка мнемосхемы в режиме эмуляции

2.7. Настройка OPC сервера

Чтобы OPC сервер «знал» с какими контроллерами нужно взаимодействовать, необходимо добавить их в конфигурацию OPC.

1. Запустите *OPCConfig* и добавьте контроллер (рис. 95).

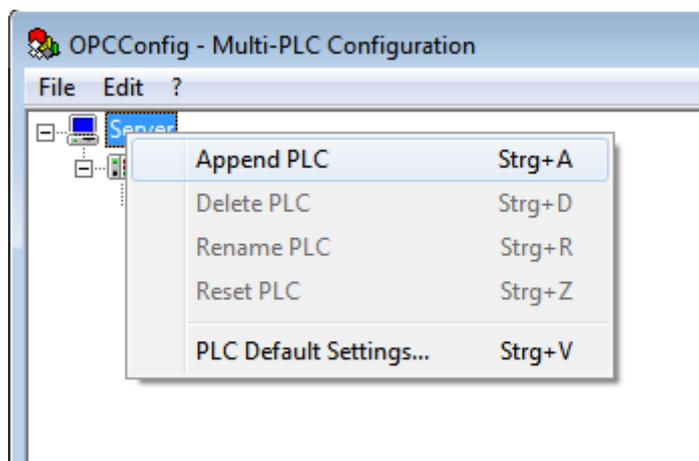


Рис. 95. Добавление контроллера в OPC конфигурацию

2. Задайте настройки контроллера (рис. 96).

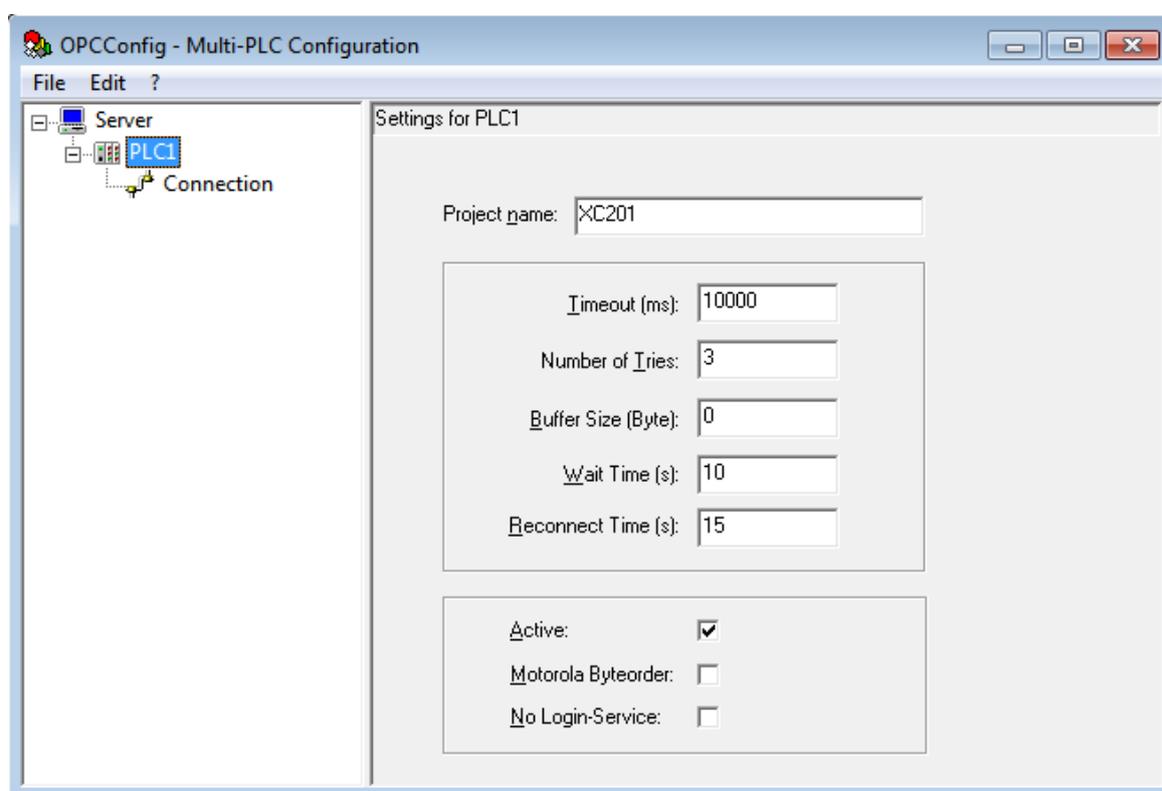


Рис. 96. Настройки контроллера

3. *Отредактируйте соединение с контроллером (рис. 97). Задайте его IP адрес, по которому OPC сервер будет обращаться к контроллеру в соответствии с картой IP адресов (рис. 3), в данном случае 192.168.119.13.*

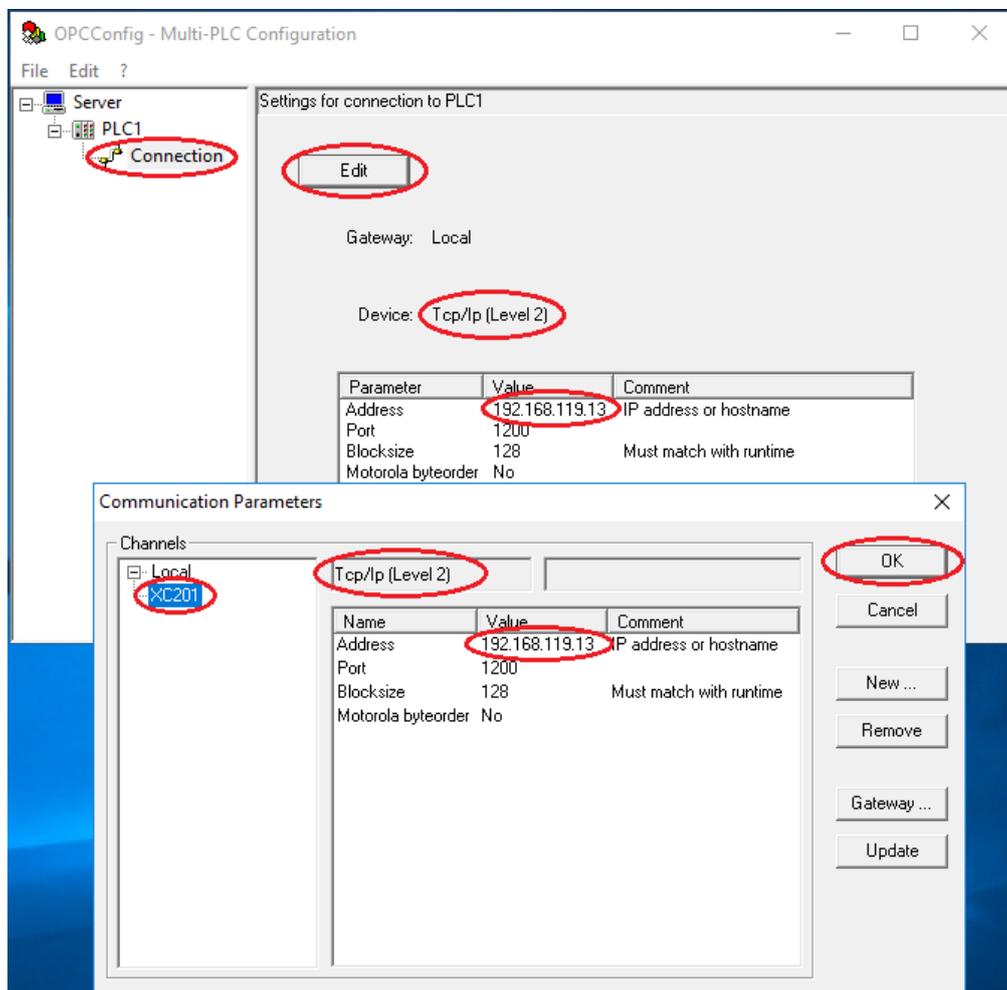


Рис. 97. Редактирование соединения с контроллером

2.8. Апробация созданной системы управления

1. *Создайте электрическое соединение лабораторной установки с контроллером. Соедините необходимые провода между датчиками физической модели и входами «I» контроллера (рис. 98). Также соедините необходимые провода между входами исполнительных элементов физической модели и выходами «Q» контроллера. Данные соединения должны производиться в соответствии с назначением переменных физических входов/выходов в редакторе объявлений PLC_PRG. Объявленные переменные соответствуют определённым датчикам и исполнительным элементам физической модели. Соедините модульный*

контроллер XC-CPU201 с HMI панелью специальным проводом для передачи данных по протоколу CAN Open.

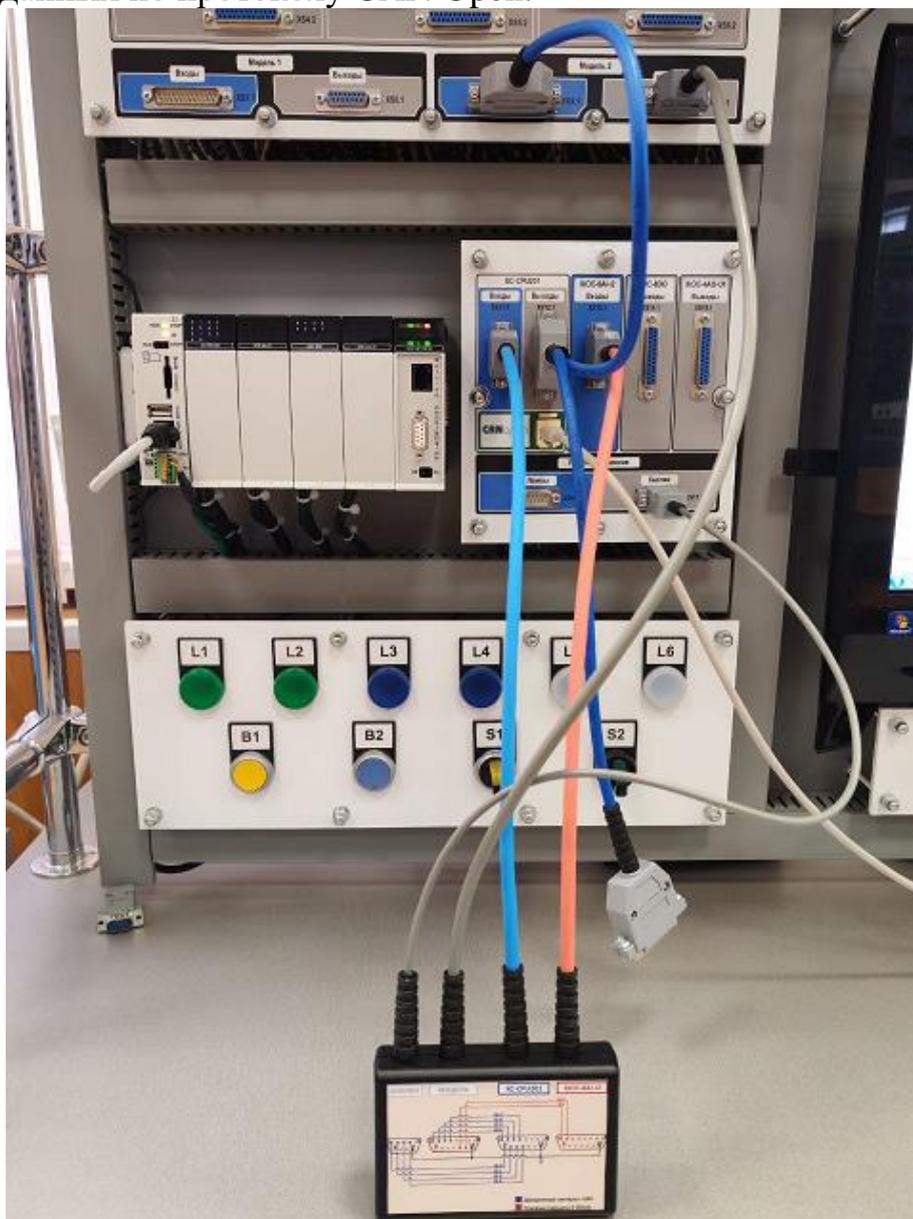


Рис. 98. Подключение контроллера к модели нефтяной задвижки

2. Проверьте соединение с контроллером XC-CPU201. Во вкладке «Онлайн» (Online) откройте диалог «Параметры соединения» (Communication parameters) и удостоверьтесь в наличии настройки TCP/IP соединения с IP-адресом в соответствии с картой IP адресов (рис. 3), в данном случае 192.168.119.13. Установите соединение с контроллером: «Онлайн/Соединение» (Online/Login). CODESYS попросит вас подтвердить загрузку (download) кода проекта. Также во вкладке «Онлайн» (Online) выберите опцию «Создать загрузочный проект» (Create boot project).

4. *Запустите контроллер XC-CPU201: «Онлайн/Запуск» (Online/Run).*

5. *Проверьте соединение с внутренним контроллером HMI панели XV102. Во вкладке «Онлайн» (Online) откройте диалог «Параметры соединения» (Communication parameters) и удостоверьтесь в наличии настройки TCP/IP соединения с IP-адресом в соответствии с картой IP адресов (рис. 3), в данном случае 192.168.119.14. Установите соединение с контроллером: «Онлайн/Соединение» (Online/Login). CODESYS попросит вас подтвердить загрузку (download) кода проекта. Также во вкладке «Онлайн» (Online) выберите опцию «Создать загрузочный проект» (Create boot project).*

6. *Запустите внутренний контроллер HMI панели XV102: «Онлайн/Запуск» (Online/Run).*

7. *Запустите профайлер среды SCADA TRACE MODE в режиме реального времени. Для этого сначала сохраните проект для монитора реального времени (рис. 99), а затем запустите профайлер (рис. 100).*

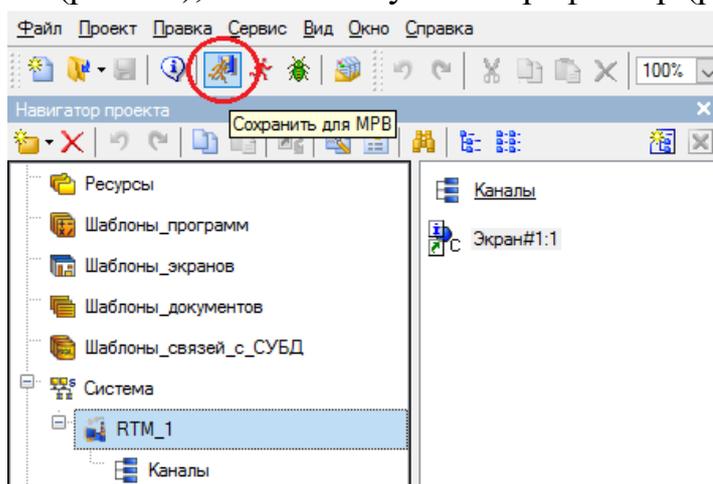


Рис. 99. Сохранение для монитора реального времени

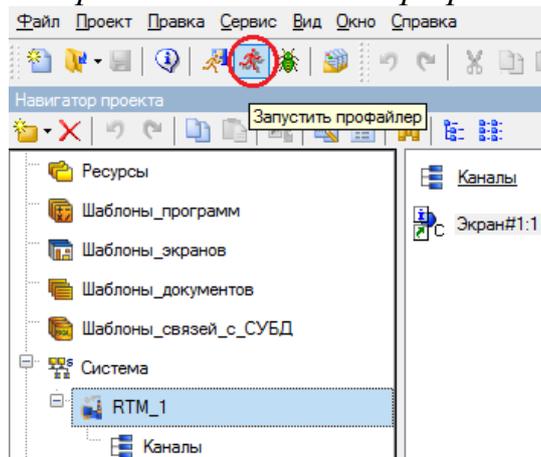


Рис. 100. Запуск профайлера TRACE MODE

В итоге запустится профайлер с работающим монитором реального времени (МРВ), где можно просматривать состояние компонентов и полноценно управлять всей системой нефтяной задвижки, нажимая на кнопки, задавая нужные пределы давления (рис. 101).

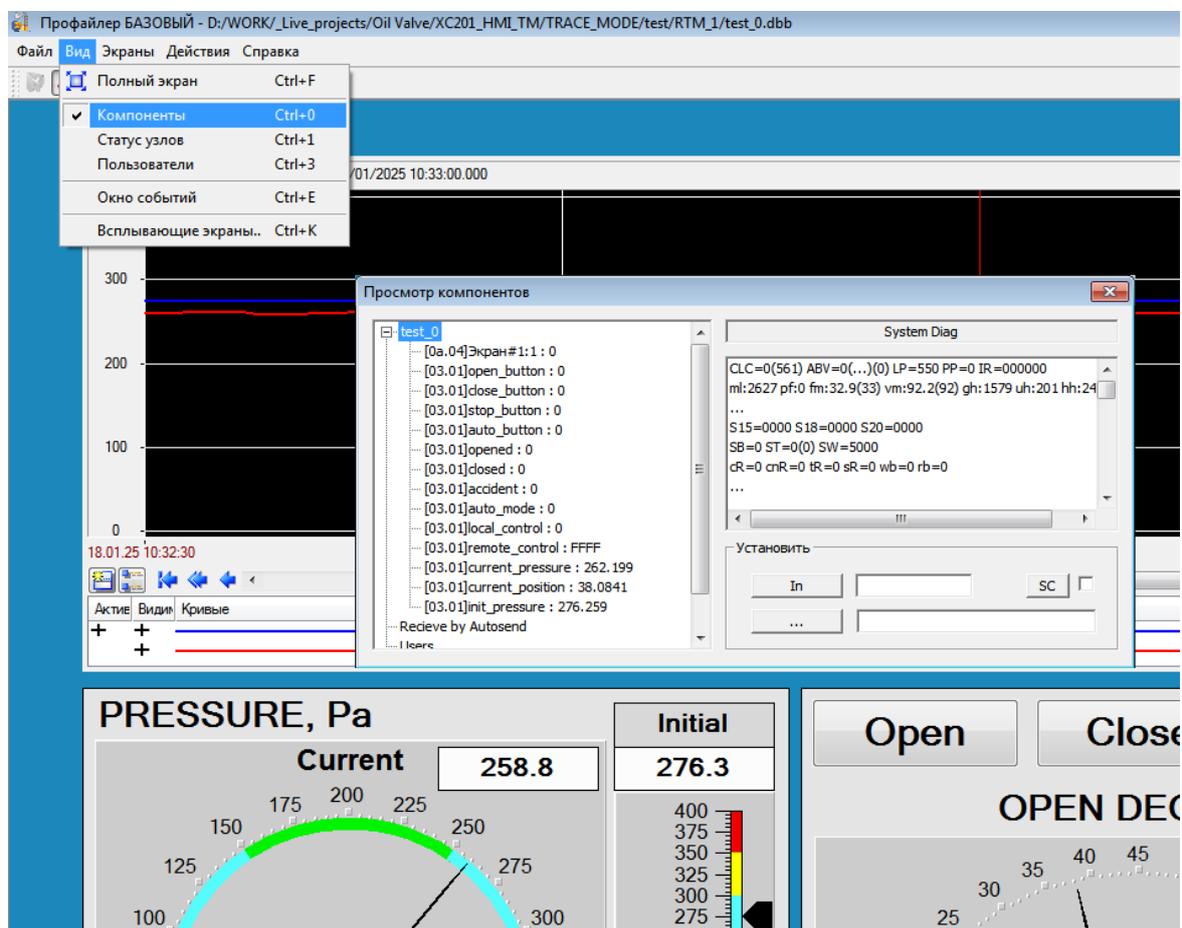


Рис. 101. Работающий монитор реального времени (МРВ)

8. Проверьте работоспособность системы. Включите лабораторную установку нефтяной задвижки и на её панели управления с помощью магнитного ключа активируйте дистанционный режим управления. С кнопок В1 и В2 на универсальном стенде включите режимы открывания и закрывания заслонки в нефтяной задвижке. С кнопки S1 на учебном стенде можно останавливать процесс открывания/закрывания в любой момент времени.

Проверьте работоспособность системы, нажимая на кнопки открывания/закрывания или остановки на профайлере SCADA системы в режиме реального времени.

Переведите систему в автоматический режим нажатием кнопки S2 на учебном стенде или кнопки Auto на профайлере SCADA системы. Установите с помощью задатчика начального давления нужный его

уровень на SCADA системе. В результате система должна автоматически поддерживать нужный уровень давления в трубопроводе.

Результат всех выполняемых манипуляций должен отображаться на объекте визуализации (рис. 35) и на мнемосхеме в SCADA системе TRACE MODE (рис. 63) в виде конкретных значений текущего давления и углового положения заслонки нефтяной задвижки.

Дополнительно проверьте передачу значений тегов (переменных) через OPC сервер. Для этого запустите OPC обозреватель (MatrikonOPC Explorer) и установите соединение с OPC сервером CoDeSys.OPC.02 (рис. 102).

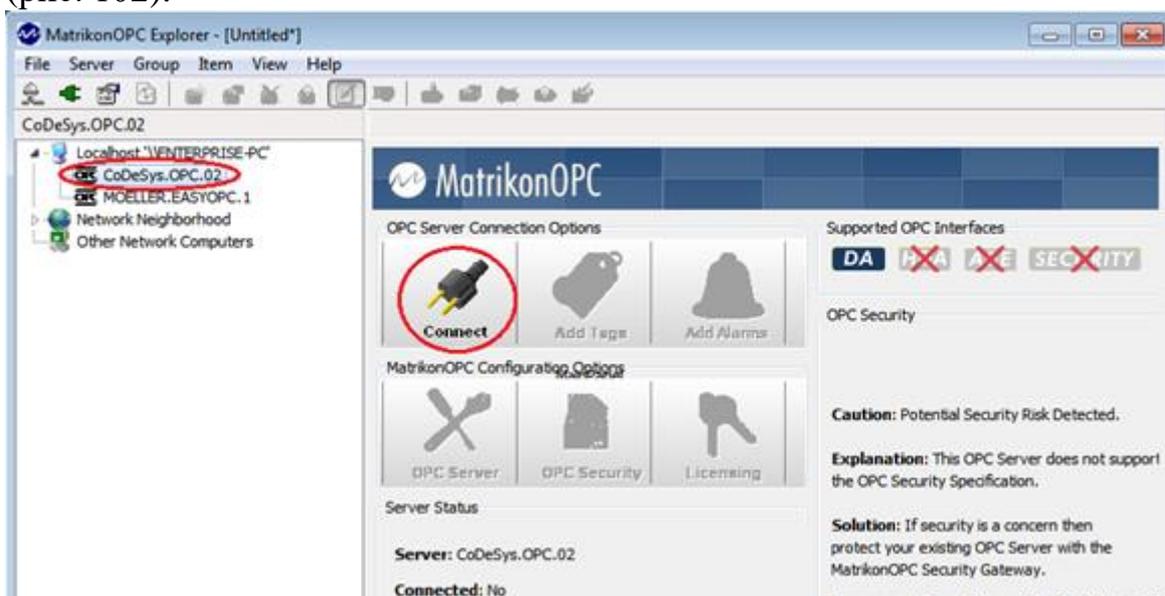


Рис. 102. Соединение с OPC сервером через MatrikonOPC Explorer

Добавьте нужные теги для отображения (рис. 103, 104). Далее можно просматривать состояние добавленных тегов в режиме реального времени (рис. 105).

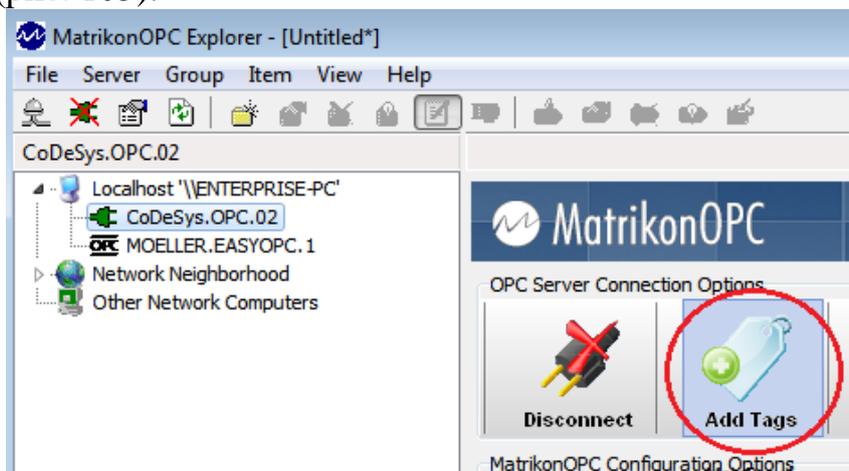


Рис. 103. Вход в раздел добавления тегов

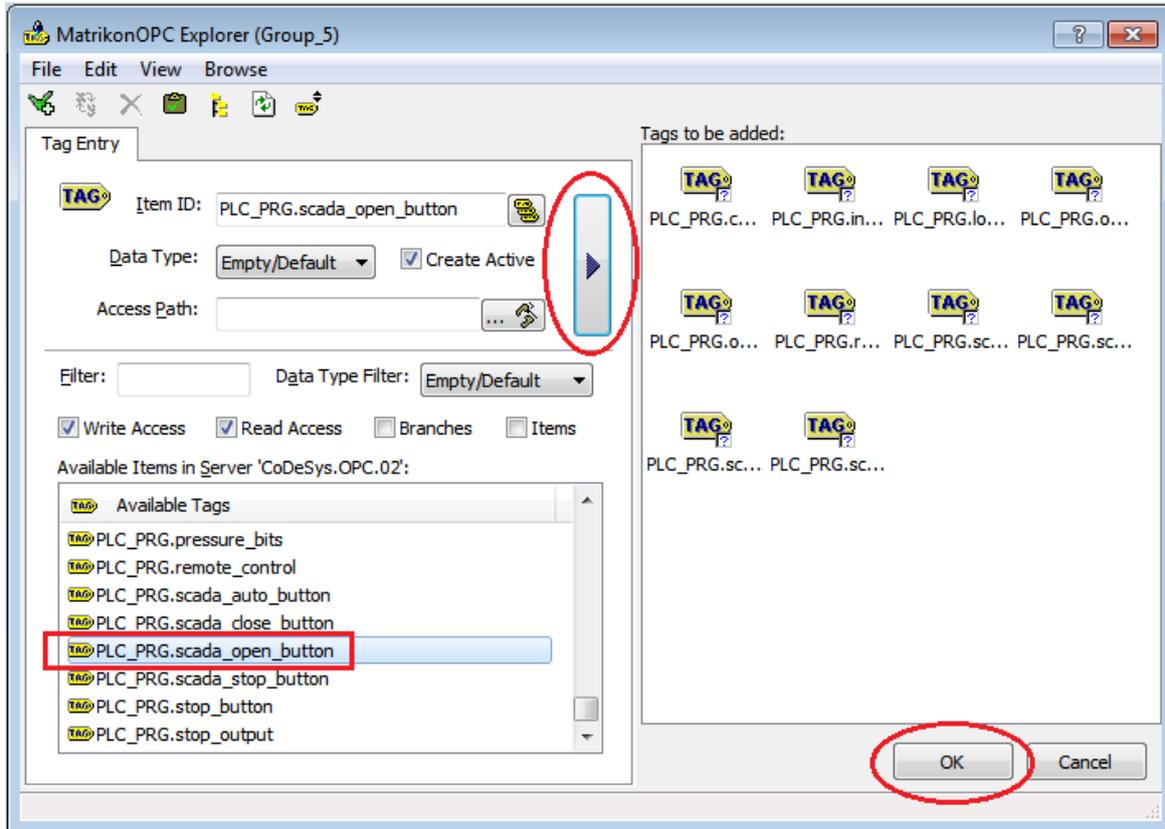


Рис. 104. Добавление тегов в обозреватель

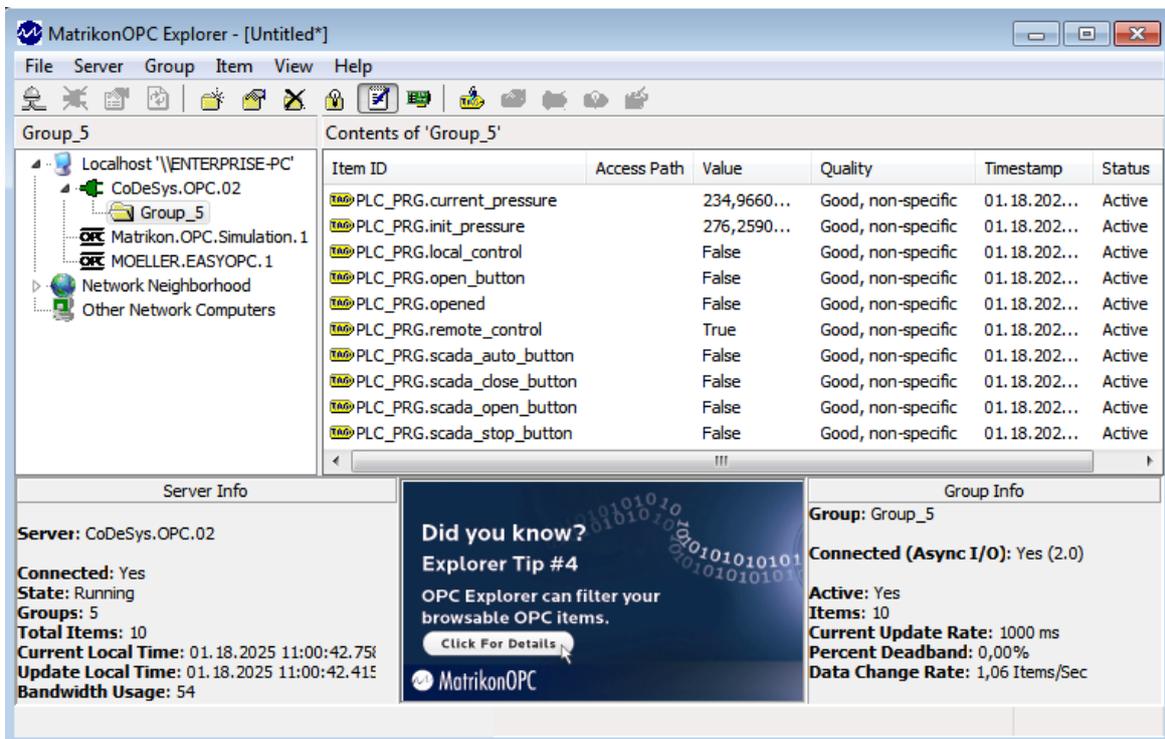


Рис. 105. Отображение состояния тегов в режиме реального времени

3. УСТАНОВКА ПОГРАММНОГО ОБЕСПЕЧЕНИЯ

Внимание! В аудитории учебного центра на стендах данную процедуру выполняет только преподаватель один раз. Обучающийся, при наличии данного программного обеспечения, может его установить только на своём личном компьютере.

3.1. Установка и настройка CODESYS 2.3.9 SP8

1. Запустите файл *Setup_XSOFT_CODESYS_V2.3.9_SP8.exe*.
2. Если автоматически не запустился установщик таргетов *Setup_XC_XV_Targets_V2.3.9_SP8.exe*, запустите его.
3. Активируйте таргеты для контроллера XC-CPU201 и HMI панели оператора XV102. Для этого откройте специальную CODESYS утилиту InstallTarget и следуйте следующим шагам (рис. 106-108).

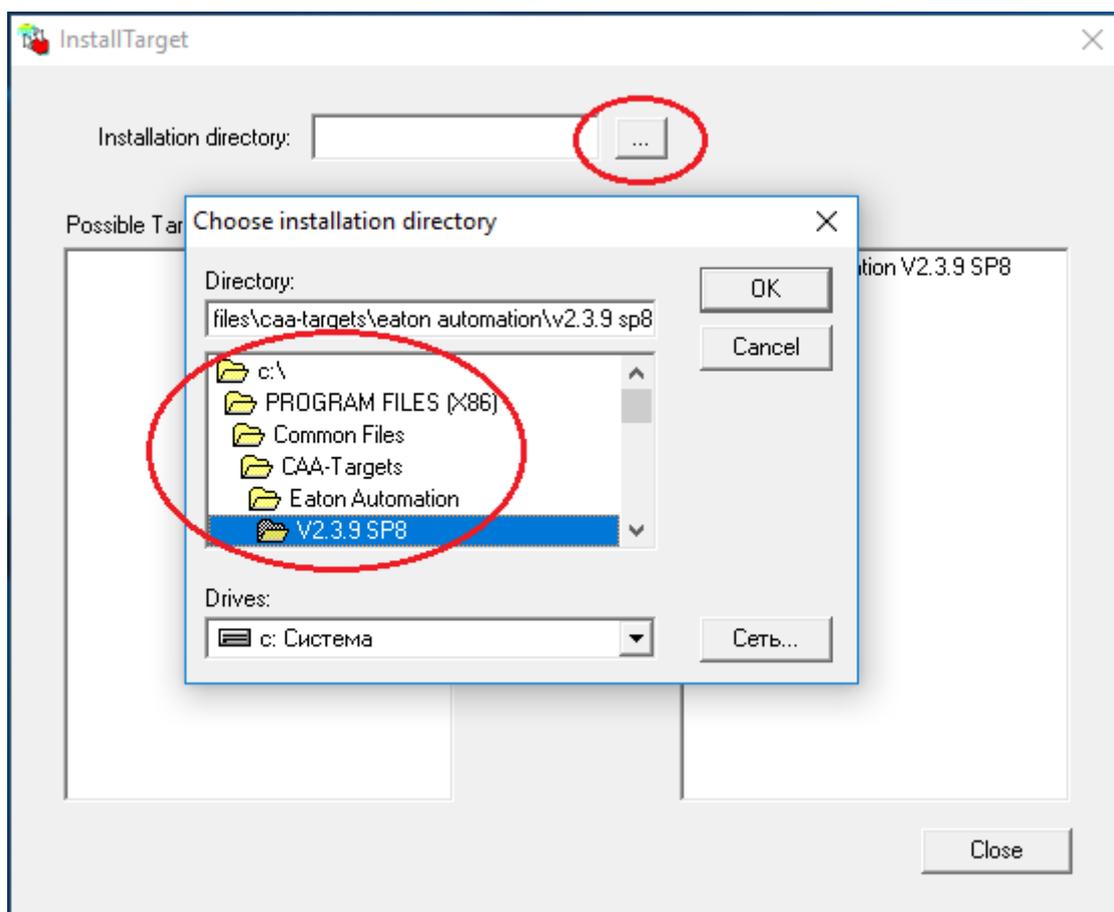


Рис. 106. Определение области видимости выбора таргет-файла

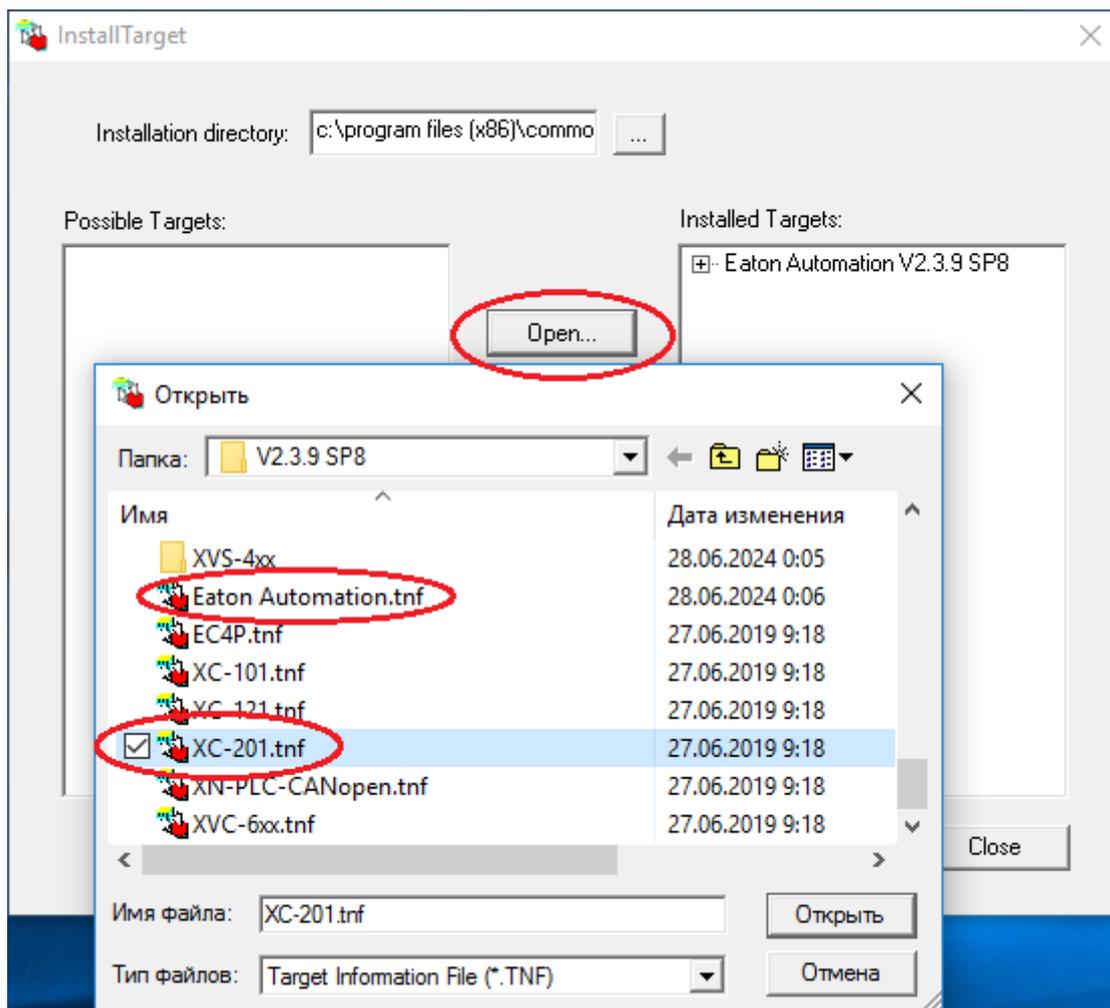


Рис. 107. Выбор TNF таргет-файла для установки таргетов

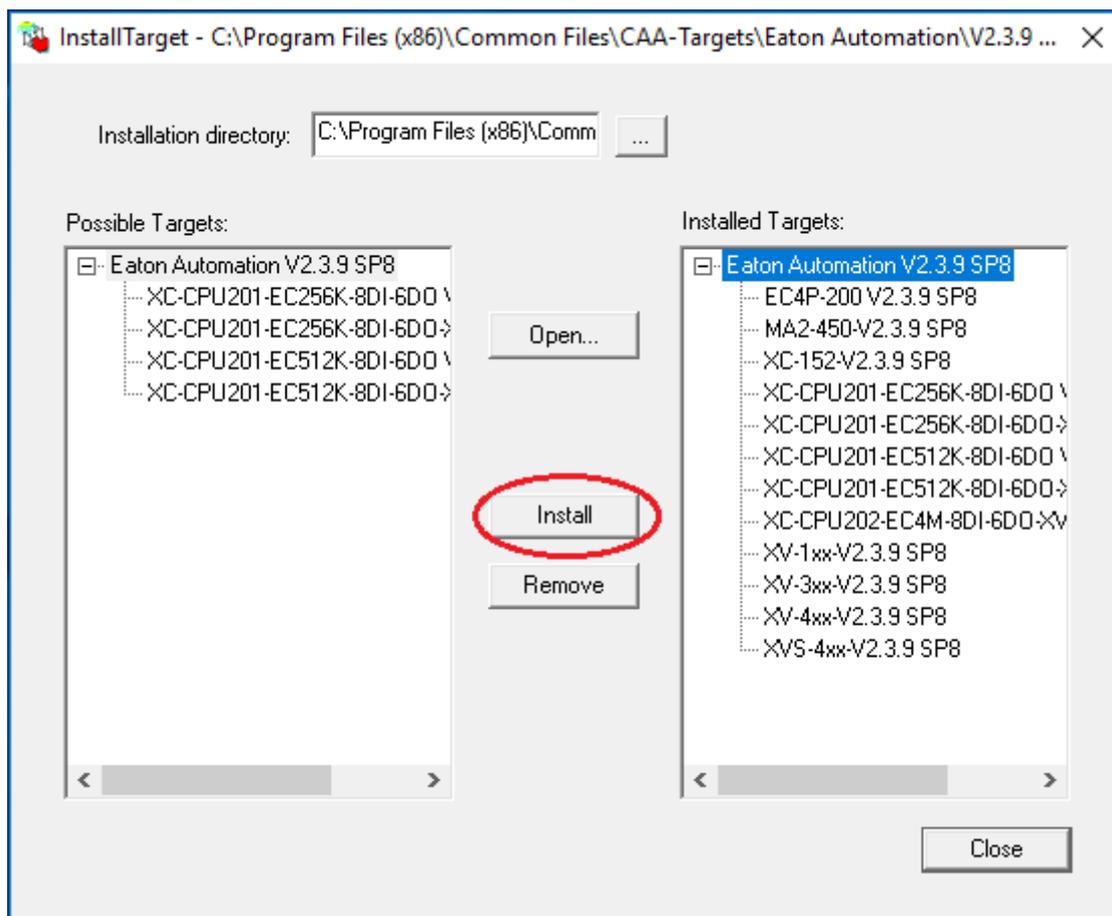


Рис. 108. Установка таргетов для контроллеров

3.2 Установка Trace Mode 6.09

1. Установка программного комплекса Trace Mode 9.09 достаточно проста. Запустите установщик «Setup» и следуйте его дальнейшим инструкциям.

3.3. Установка и лицензирование Galileo 8.1

1. Получите лицензионный ключ для использования его при установке Galileo. Для выполнения этой операции необходимо иметь лицензионный сертификат, который необходимо ввести на сайте <http://www.eaton-automation.com/license>, вместе с личными данными (рис. 109). В ответном письме на указанную электронную почту будет выслан лицензионный ключ.

2. Установите Galileo 8.1.10. Запустите файл GalileoV8110.exe и в ходе установки введите лицензионный ключ.



Рис. 109. Процесс получения лицензионного ключа

3.4. Обновление прошивки HMI панели XV102

1. Подайте электрическое питание на панель оператора. После загрузки Windows CE, нажмите кнопку *Start* и выберите в меню *Programs/Control Panel* (рис. 110).



Рис. 110. Открывание Control/Panel

2. Два раза кликните по вкладке *Network*.
3. В появившемся окне два раза кликните по вкладке *ONBOARD1* (рис. 111).



Рис. 111. Открывание Onboard1

4. При этом откроется окно *FEC Internet Driver*, в котором будут отображены сетевые настройки контроллера панели оператора (рис. 112).



Рис. 112. Сетевые настройки контроллера панели оператора

5. Задайте IP адрес 192.168.119.14 и нажмите «ОК».
6. Откройте Codesys 2.3.9. При выборе целевой платформы, необходимо указать XV-1xx-V2.3.9 SP8.
7. Перейдите во вкладку ресурсов, зайдите в конфигурацию ПЛК, выберите Firmware и нажмите кнопку Start (рис. 113).

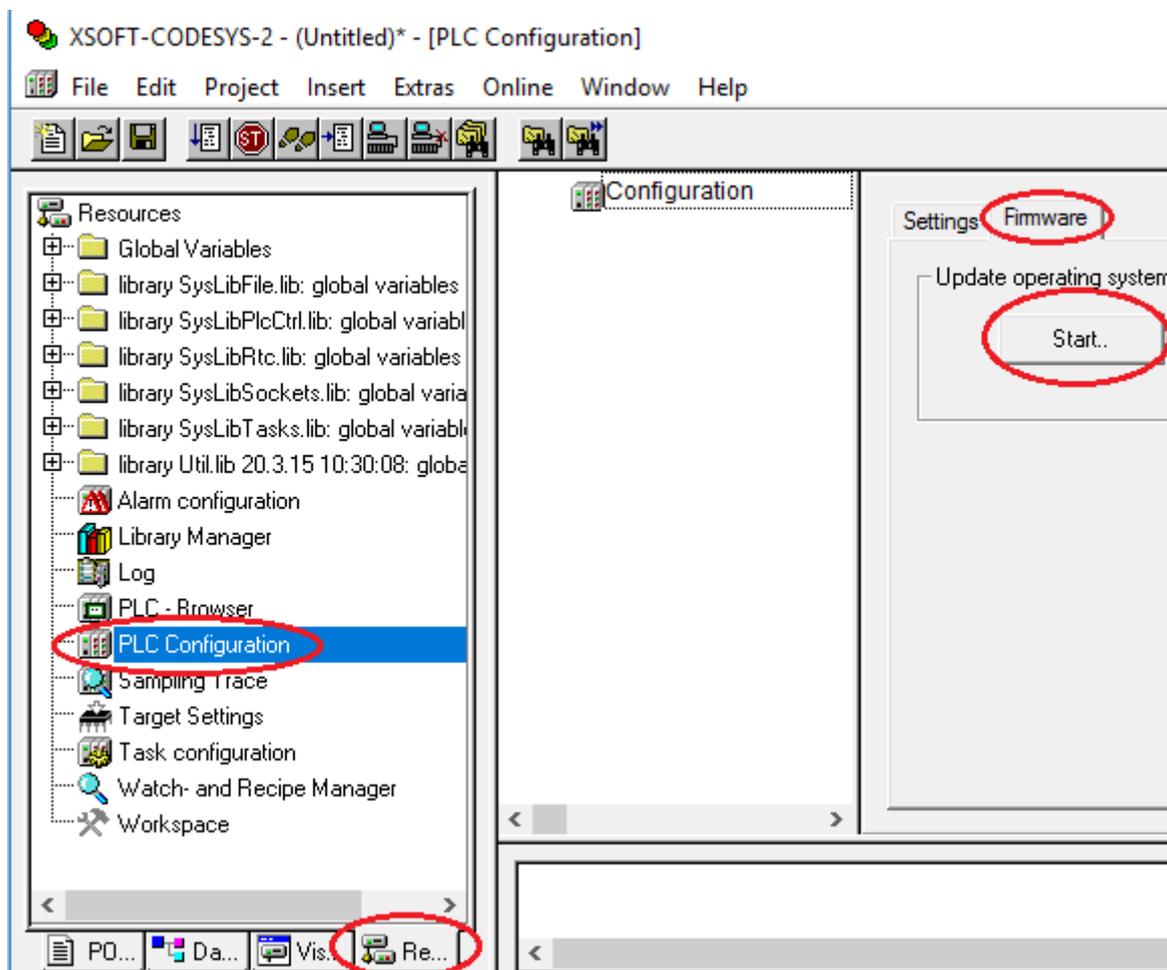


Рис. 113. Запуск прошивки HMI панели

8. В появившемся окне выберите файл TargetFirmwareWinCE_V2.4.21, расположенный по адресу Program Files\Common Files\CAA-Targets\Eaton Automation\V2.3.9 SP8\Firmware\XV-1xx.

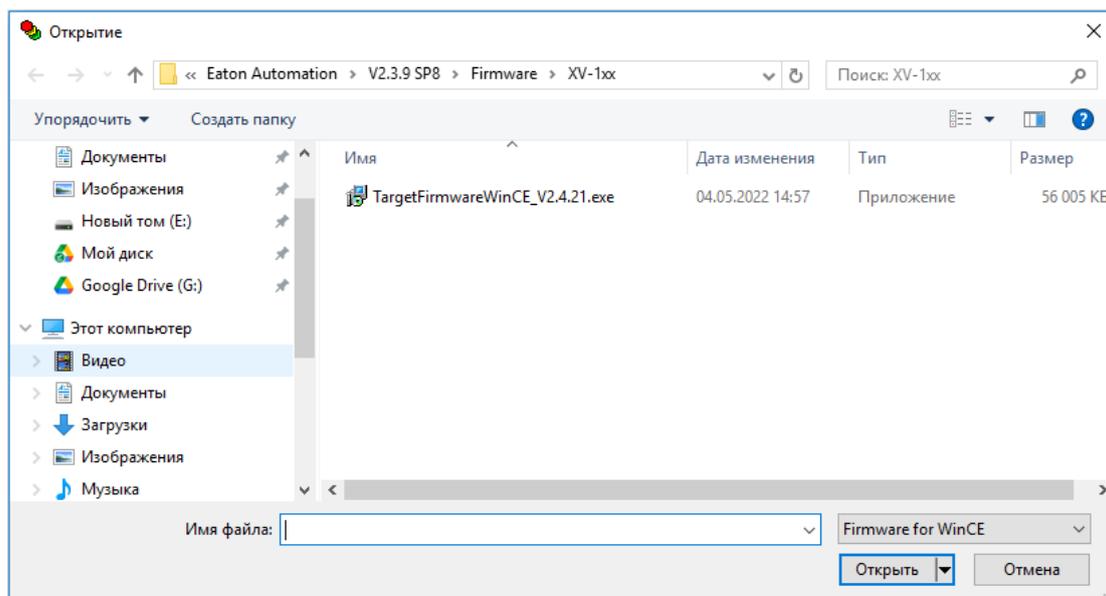


Рис. 114. Выбор TargetFirmwareWinCE_V2.4.21

9. После этого запустится окно инсталлятора. В первом приветственном окне, мы нажимаем кнопку Next. В следующем выбираем режим FTP-Installation и нажимаем кнопку Next (рис. 115).

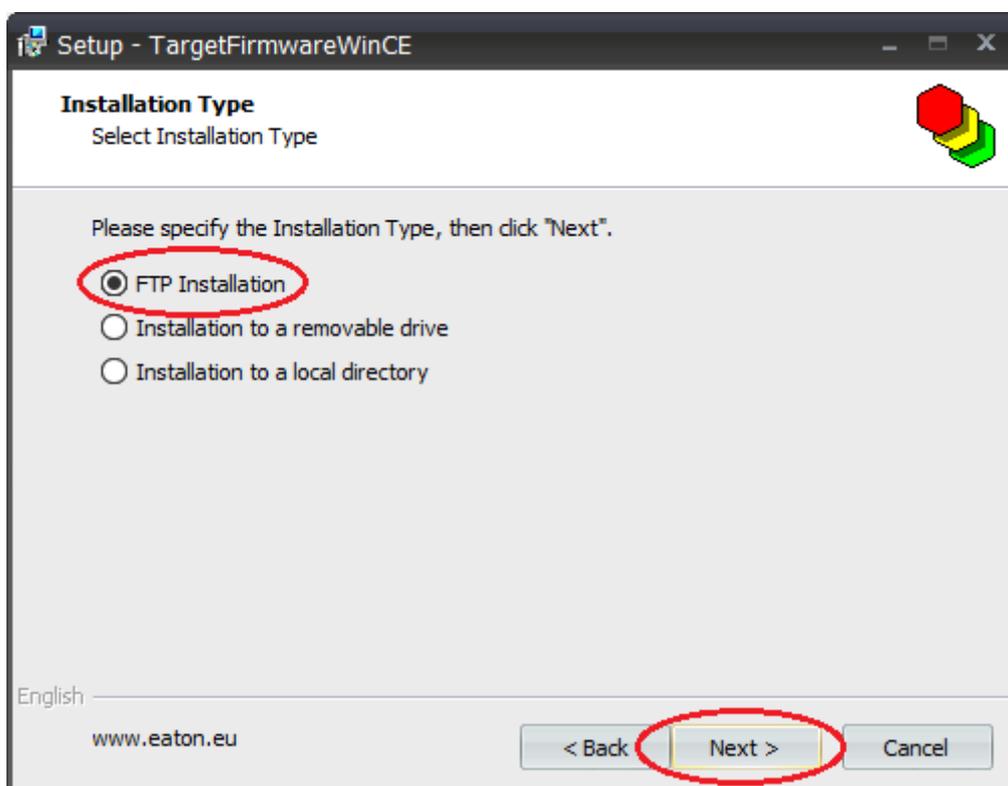


Рис. 115. FTP установка

10. В следующем окне введите IP-адрес панели оператора (192.168.119.14), остальные поля оставьте неизменным (рис. 116).

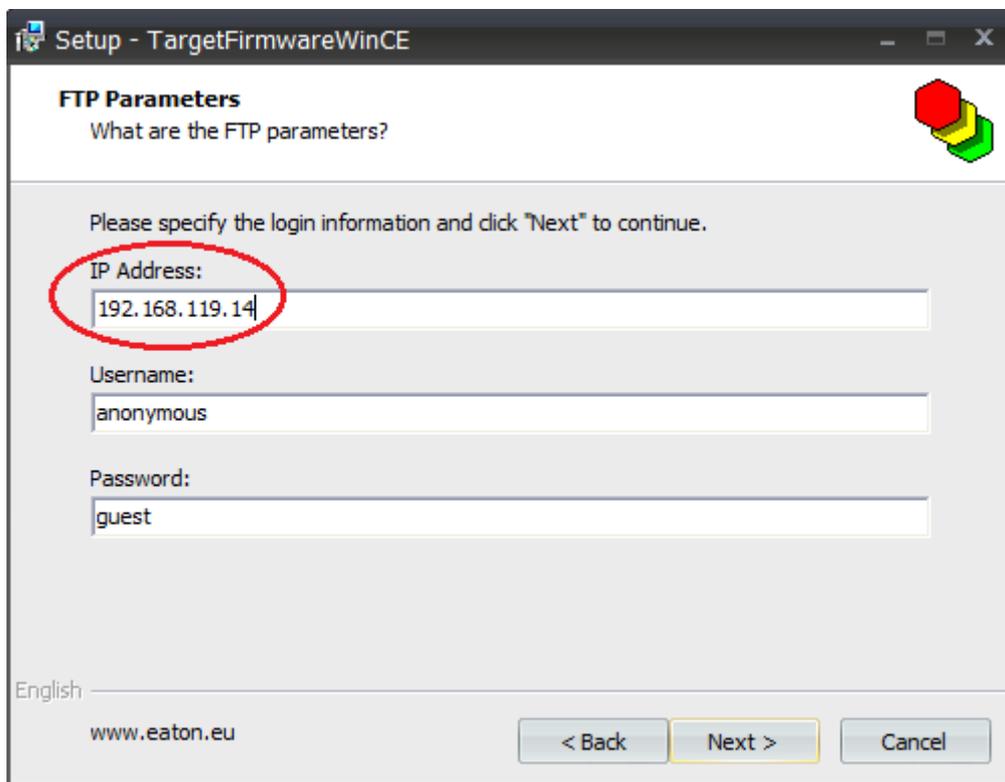


Рис. 116. Задание IP адреса для FTP установки

11. Перед выполнением следующего шага необходимо запустить FTP сервер на панели оператора, Start/Programs/Communication/FTP server/ОК.
12. Если IP-адрес был указан верно и в панели оператора запущен FTP-сервер, то инсталлятор автоматически определит тип панели (рис. 117). Далее нажмите кнопку Next.

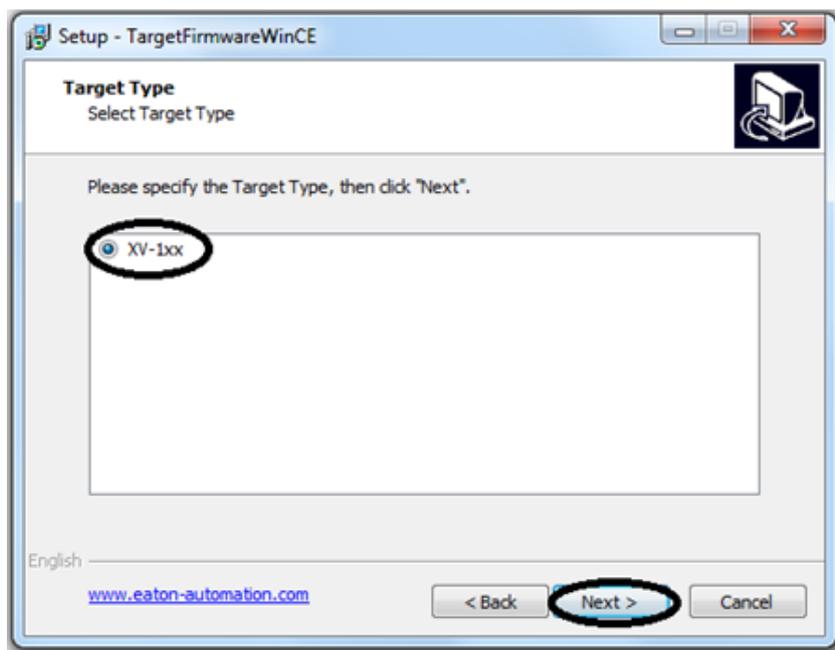


Рис. 117. Автоматическое определение типа НМІ панели

13. Далее происходит выбор устанавливаемых компонентов. Выберите все компоненты и нажмите кнопку *Next* (рис. 118).

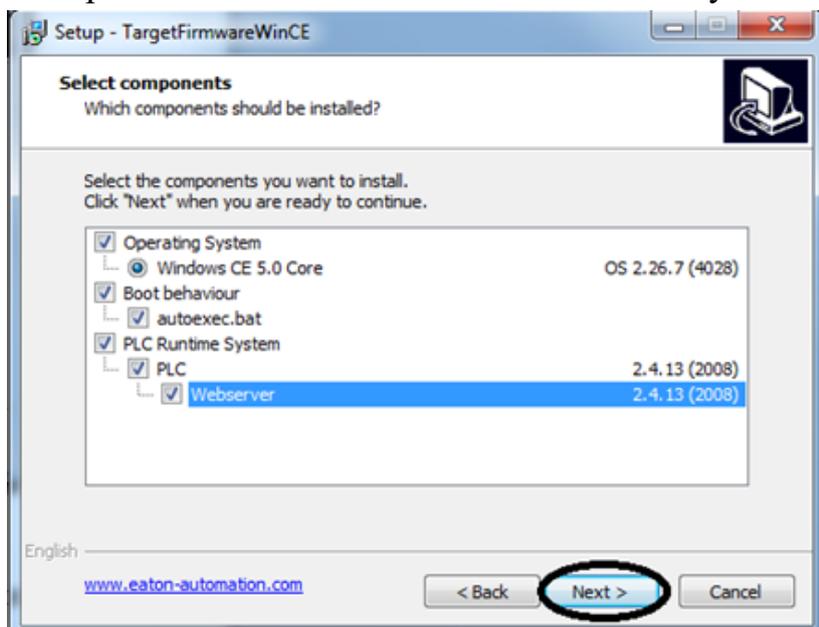


Рис. 118. Выбор устанавливаемых компонентов

14. Завершающим этапом подтвердите установку выбранных компонентов нажатием кнопки *Install*. Дождитесь завершения установки, панель оператора должна автоматически перезагрузиться.

СПИСОК ЛИТЕРАТУРЫ

1. Гирник А.С., Федянин А.Л., Шилин А.А. Системы автоматки и управления на базе программируемых логических контроллеров: учебное пособие [Электронный ресурс] / А.С. Гирник. – Томск : Изд-во ТПУ, 2024. – Заглавие с титульного экрана. – URL: https://portal.tpu.ru/departments/otdel/publish/catalog/2024/method_2024/Tab/GirnikFedyaninShilin.pdf .
2. Руководство пользователя по программированию ПЛК в CoDeSys 2.3. 3S – Smart Software Solutions GmbH 2008. – URL: https://owen.ru/uploads/373/cds23_manual_v2.8.pdf.
3. Первые шаги с CoDeSys. 3S – Smart Software Solutions GmbH 2004. – URL: https://owen.ru/uploads/373/cds23_firststeps.pdf.
4. Визуализация CoDeSys. Дополнение к руководству пользователя по программированию ПЛК CoDeSys 2.3. 3S – Smart Software Solutions GmbH 2008. – URL: https://owen.ru/uploads/373/cds23_visu_v1.7.pdf.