

**КАЗАНСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ**  
**ИНСТИТУТ ФИЗИКИ**  
***Кафедра радиоэлектроники***

**М.Н.ОВЧИННИКОВ, А.Г.ГАВРИЛОВ, А.И.ДЕРКАЧ,  
В.А.МАЦЕНКО**

**УЧЕБНО-ЛАБОРАТОРНЫЙ КОМПЛЕКС ПО  
ПРИМЕНЕНИЮ АВТОМАТИЗИРОВАННЫХ  
СИСТЕМ ДЛЯ ИССЛЕДОВАНИЯ ЯВЛЕНИЙ  
ПЕРЕНОСА В ПОРИСТЫХ СРЕДАХ**

**Учебно-методическое пособие**

**Казань – 2017**

**УДК 004.896:532.546:622.32**

**О–35**

*Принято на заседании кафедры радиоэлектроники  
Протокол № 5 от «10» апреля 2017 г*

**Рецензент:**  
канд. физ.-мат. наук, зав. лаб. **Я.В. Фаттахов**

**Овчинников М.Н.**

**Учебно-лабораторный комплекс по применению автоматизированных систем для исследования явлений переноса в пористых средах Учебно – методическое пособие / М.Н. Овчинников, А.Г. Гаврилов, А.И. Деркач, В.А. Маценко – Казань: Казанский (Приволжский) федеральный университет, 2017. – 79с.**

Настоящее учебно-методическое пособие адресовано студентам – бакалаврам, магистрантам и аспирантам, обучающимся по профилям «Радиофизика», «Радиофизические измерения», «Радиофизические методы по областям применения», «Разработка и эксплуатация нефтяных месторождений». Учебное пособие будет интересно также широкому кругу читателей, занимающихся проблемами проектирования, разработки и отладки автоматизированных систем контроля (ACK) в различных областях применения: в нефтяной и газовой промышленности, на транспорте, в энергетике, машиностроении, коммунальном хозяйстве.

Знания и практические навыки, полученные в Лаборатории подземной гидродинамики, где находится учебно – лабораторный комплекс, в области проектирования и создания автоматизированных систем управления производствами на базе современных комплектующих мировых ведущих производителей и стандартизованных, принятых в мировой практике языков программирования МЭК61131–13, будут востребованы в различных областях науки и производства.

© **Овчинников М.Н., Гаврилов А.Г.,  
Деркач А.И., Маценко В.А., 2017**  
© **Казанский университет, 2017**

# **Оглавление**

ВВЕДЕНИЕ .....	4
<b>РАЗДЕЛ 1. ЛАБОРАТОРНАЯ РАБОТА №1</b>	
«Основные принципы разработки прикладных программ для промышленного контроллера FASTWEL CPM902».....	7
<b>РАЗДЕЛ 2. ЛАБОРАТОРНАЯ РАБОТА №2</b>	
«Принцип работы и способы конфигурирования контроллера FASTWEL CPM902 в качестве подчиненного узла (сервера) сети MODBUS».....	33
<b>РАЗДЕЛ 3. ЛАБОРАТОРНАЯ РАБОТА №3</b>	
«Основные принципы разработки экранных форм графической панели WT3010. Организация взаимодействия панели WT3010 и контроллера CPM902 по протоколу MODBUS» .....	56
<b>СПИСОК ЛИТЕРАТУРЫ.....</b>	77
<b>ПРИЛОЖЕНИЕ А .....</b>	78
<b>ПРИЛОЖЕНИЕ В.....</b>	79

## **ВВЕДЕНИЕ**

История конструирования и создания автоматизированных систем контроля выработки нефтяного пласта в Казанском Государственном Университете (КГУ) уходит корнями в далекие пятидесятые – шестидесятые годы прошлого столетия. Тогда для радиодиспетчеризации Ромашкинского нефтяного месторождения и решения некоторых проблем, возникающих при добыче нефти, потребовалось участие физиков, математиков и радиофизиков КГУ. В 1955г. по инициативе областного комитета КПСС и объединения «Татнефть» были заключены первые в истории Казанского университета хозяйствственные договоры с нефтедобывающими организациями ТАССР.

В начале семидесятых годов прошлого столетия разработка Ромашкинского месторождения перешла в стадию поздней выработки и почти на всей территории месторождения велась с применением внутренконтурного и законтурного заводнения. На этой стадии, для управления оптимальной выработкой нефтяного пласта, необходимо постоянно следить за движением нефтяных и водяных потоков в разрабатываемом пласте. Наиболее эффективно это можно было сделать, применяя новый, активно развивающийся в то время, вид гидродинамических исследований пластов – метод фильтрационных волн давления. Идея использования метода фильтрационных гармонических волн давления (ФВД) для определения фильтрационных параметров пластов получила теоретическое и экспериментальное развитие в работах С.Н. Бузинова, И.Д. Умрихина, Ю.М. Молоковича, Н.Н. Непримерова и др. [1,2,3]. Она заключается в создании периодических изменений расхода на возмущающей скважине и регистрации откликов на него в виде изменений давления на возмущающей и реагирующей скважинах. В последнее время метод ФВД используется не только для исследования гидропроводности и пьезопроводности в около скважинных и межскважинных интервалах, но и как средство повышения нефтеотдачи пластов путем задания циклических режимов нагнетания и отбора.

В 90-е годы прошлого столетия на кафедре радиоэлектроники Казанского университета был разработан и неоднократно модернизировался программно-технический комплекс – «Автоматизированная система контроля и управления выработкой пластов» («АСКУ-ВП»), предназначенный для непрерывного контроля и совершенствования процесса добычи нефти. «АСКУ-ВП» позволил проводить весь спектр промысловых гидродинамических исследований на добывающих и нагнетательных скважинах.

В настоящее время идея комплекса «АСКУ-ВП» получила дальнейшее развитие в рамках организации учебного процесса для студентов старших курсов Института физики и Института геологии и нефтегазовых технологий, входящих в структуру Казанского (Приволжского) Федерального Университета. В ходе реализации Программы развития КФУ сотрудниками кафедры радиоэлектроники КФУ в содружестве со специалистами казанского филиала компании Прософт был создан «Учебно-лабораторный комплекс по применению автоматизированных систем для исследования явлений переноса в пористых средах».

В целом, комплекс технических средств и применяемое базовое ПО должны позволять создавать в лабораторных условиях прототипы систем автоматизации различных технологических объектов, в том числе и объектов нефтедобычи. Комплекс должен стать базой не только для подготовки молодых специалистов, но и для выполнения опытно-конструкторских и научно-исследовательских работ сотрудниками кафедры.

Нефтедобывающее производство включает в себя большое количество разнообразных технологических объектов: добывающие и нагнетательные скважины; индивидуальные и групповые замерные установки; блоки дозирования реагента; дожимные насосные станции; кустовые насосные станции; установки подготовки нефти; товарные парки и т.д. Как правило, эти объекты распределены по большой площади и зачастую труднодоступны. Оборудование находится на открытых площадках или в неотапливаемых помещениях. Технологический процесс добычи протекает непрерывно, без

остановок на праздники и выходные. Такие суровые условия эксплуатации предъявляют серьезные требования к комплексу технических средств систем автоматизации. Оборудование КТС должно быть надежным, рассчитанным на эксплуатацию в расширенном диапазоне температур, предоставлять возможности для решения широкого круга задач автоматизации. Немаловажными являются способность функционального и информационного наращивания возможностей созданных на основе комплекса систем автоматизации, наличие сертификатов средств измерения и разрешения Ростехнадзора на их применение. Все эти факторы учитывались при проектировании учебно-лабораторного комплекса.

Структура пособия построена на описании выполнения ряда лабораторных работ, последовательно раскрывающих функциональные возможности лабораторного комплекса.

# **РАЗДЕЛ 1. ЛАБОРАТОРНАЯ РАБОТА №1**

## **«ОСНОВНЫЕ ПРИНЦИПЫ РАЗРАБОТКИ ПРИКЛАДНЫХ ПРОГРАММ ДЛЯ ПРОМЫШЛЕННОГО КОНТРОЛЛЕРА FASTWEL CPM902»**

### **1.1 ОПИСАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ №1**

#### **Назначение:**

изучение среды разработки CoDeSys v.2.3 и основных принципов подготовки прикладных программ для промышленного контроллера Fastwel CPM902.

#### **Цель работы – освоение практических навыков:**

- разработки прикладной программы в среде CoDeSys;
- конфигурирования контроллера CPM902 и модулей Fastwel IO;
- эмуляции проекта средствами CoDeSys;
- загрузки программы в контроллер, её отладки и демонстрации.

#### **Используемое оборудование:**

- программируемый контроллер «Fastwel I/O CPM902» с набором функциональных модулей, расположенный в ШК1 и ШК2 в соответствии со структурной схемой комплекса (см. Приложение А), где модули соединены в единую конструкцию через пазы на корпусе каждого модуля в порядке, определенном в таблице 1 (Приложение В);
- кабель нуль-модемный;
- автоматизированные рабочие места (АРМ) студентов(4) и преподавателя.

#### **Используемое инструментальное ПО:**

среда разработки Fastwel IO CoDeSys Adaptation (установлена на АРМ студентов и преподавателя).

#### **Теоретический материал для допуска к выполнению лабораторной работы:**

- описание комплекса технических средств [4];
- описание комплекса программного обеспечения [5].

## 1.2 ОБЩИЕ СВЕДЕНИЯ

В лабораторной работе №1 приобретаются практические навыки подготовки прикладной программы контроллера СРМ902 в среде **CoDeSys v.2.3**. В ходе выполнения работы обучающийся должен разработать программу для реализации простого алгоритма управления 4-мя релейными выходами (каналами модулей DIM713) в зависимости от состояния 4-х соответствующих дискретных входов (каналов модуля DIM717).

**Последовательность операций при разработке прикладной программы:**

- создание проекта;
- создание и редактирование конфигурации контроллера;
- редактирование главной программы **PLC\_PRG POU**, компиляция проекта;
- отладка программы в режиме эмуляции;
- создание логического информационного канала между средой разработки и контроллером СРМ902;
- загрузка приложения в контроллер;
- онлайн отладка программы, просмотр и установка значений переменных.

## 1.3 СОЗДАНИЕ ПРОЕКТА

1.3.1. Средствами операционной системы Windows 7 загрузить на исполнение среду разработки CoDeSys: **Пуск – Все программы – 3S Software V2.3 – CoDeSys V2.3.**

1.3.2. В запущенном редакторе среды CoDeSys выполнить команду меню **Файл – Создать.**

1.3.3. В появившейся диалоговой панели **Настройки целевой платформы** из списка доступных систем выбрать конфигурацию «**Fastwel CPB902 WinCE Runtime**». Нажать кнопку **OK**.

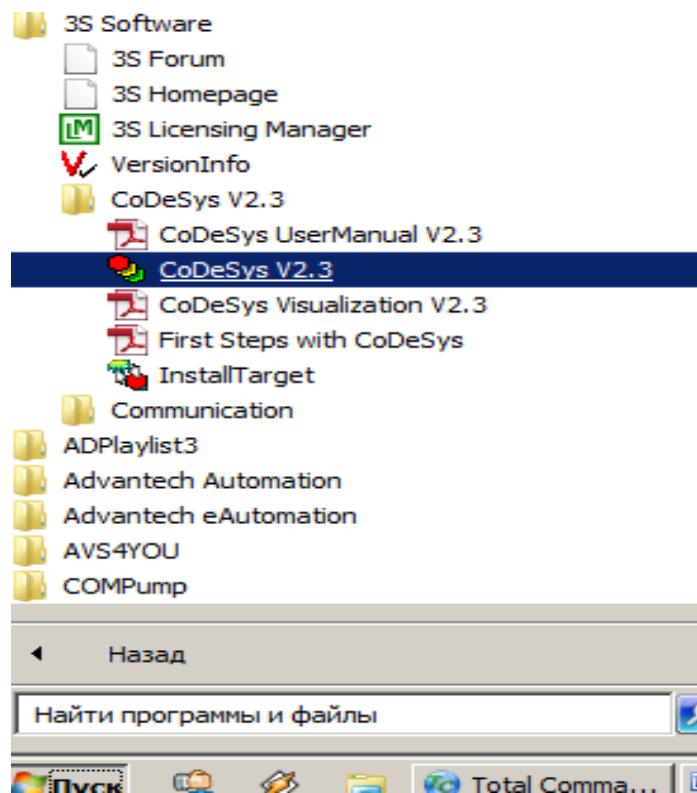


Рис. 1. Загрузка среды разработки CoDeSys

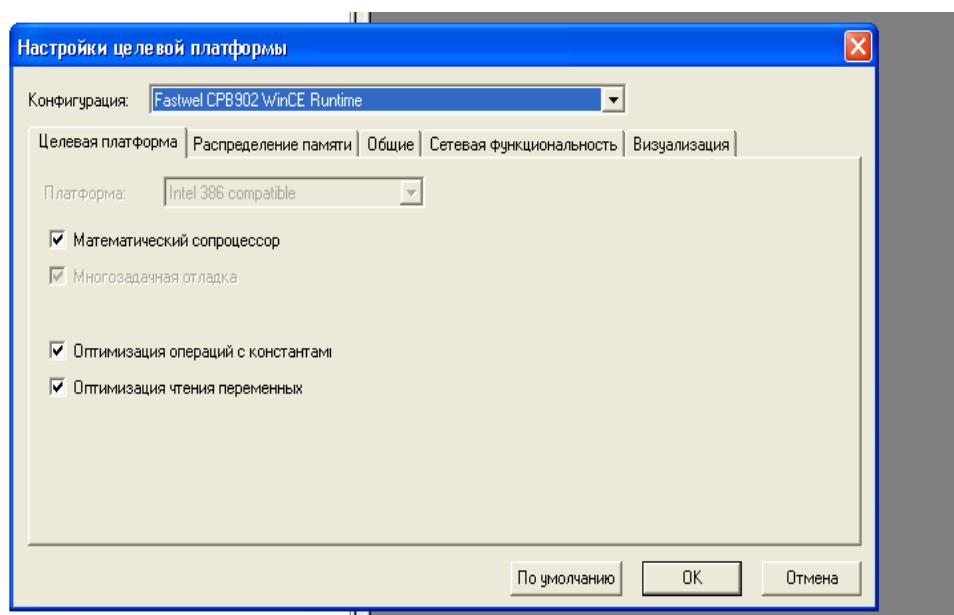


Рис. 2. Настройка целевой платформы

1.3.4. В появившейся диалоговой панели **Новый программный компонент POU** будет предложено создать программу с именем **PLC\_PRG** (язык ST по умолчанию). Нажать кнопку **OK**, тогда в проекте будет создана программа с таким именем.

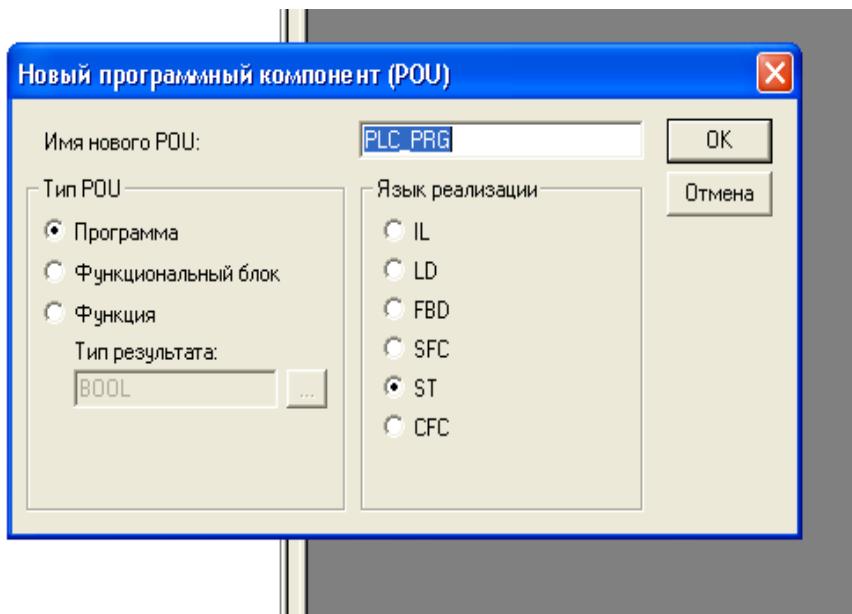


Рис. 3. Создание POU

1.3.5. Сохранить созданный проект под именем **Lab\_1.pro** на локальном диске в папке C:\Лаборатория гидродинамики \Labs\. При необходимости можно в название файла внести фамилию студента и номер учебной группы, например **Lab\_1\_Ivanov\_5101.pro**

## 1.4 СОЗДАНИЕ И РЕДАКТИРОВАНИЕ КОНФИГУРАЦИИ КОНТРОЛЛЕРА

1.4.1. Перейти на вкладку **Ресурсы** организатора объектов, расположенного в нижней области главного окна среды разработки CoDeSys, после чего дважды щелкнуть на названии ресурса **Конфигурация ПЛК**. В рабочей области появится окно с основным элементом конфигурации **CPB902 CPU Board**,

которое предназначено для создания и редактирования конфигурации контроллера Fastwel CPM902.

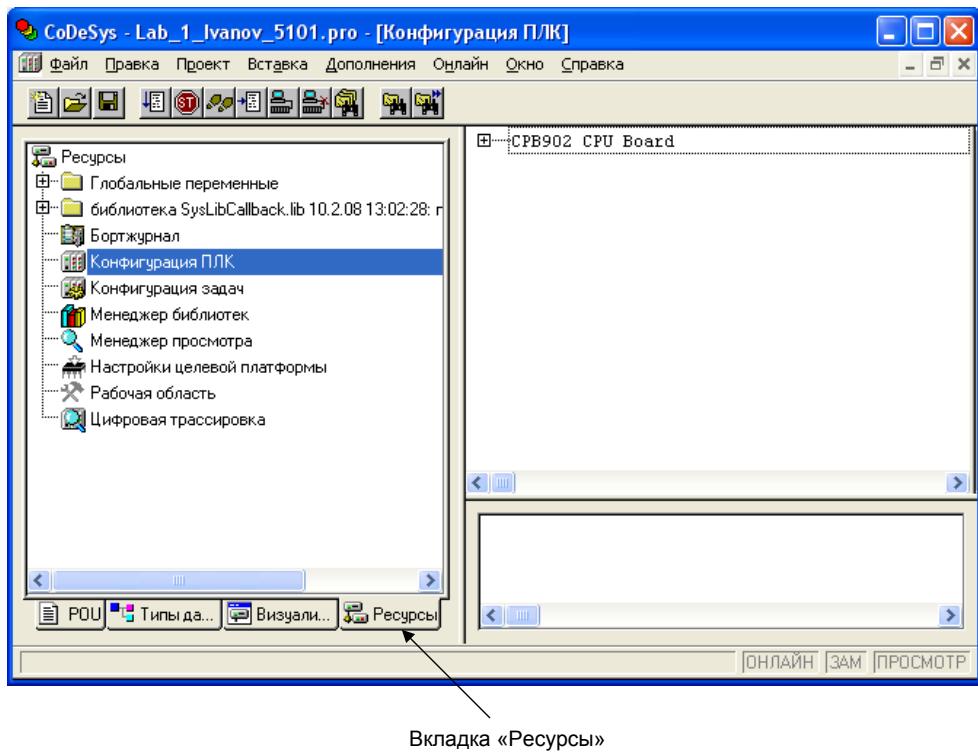


Рис. 4. Переход к ресурсам CoDeSys

1.4.2. Перейти в окно конфигурации **CPB902 CPU Board**. Этот корневой элемент конфигурации имеет поля для настройки первичных параметров контроллера.

Поле «Настройки»:

- автоматическое вычисление адресов;
- контроль перекрытия адресов;
- сохранение конф. файлов в проекте.

Поле «Параметры модуля»:

- **HotUpdateDisabled** – параметр, предназначенный для блокировки или активизации функции горячего обновления приложения;
- **Autostart** – параметр, позволяющий после загрузки приложения оставить контроллер в состоянии останова;

- **EventsRate** – период цикла "автоматической" циклической задачи. Если в окно ресурса **Конфигурация задач** не добавлено ни одной задачи, то этот параметр определит цикл автоматически создаваемой задачи **DefaultTask** и среда разработки CoDeSys ассоциирует с ней программу **PLC\_PRG**.

В первом ознакомительном проекте значения описанных выше параметров контроллера оставим без изменений, с настройками по умолчанию.

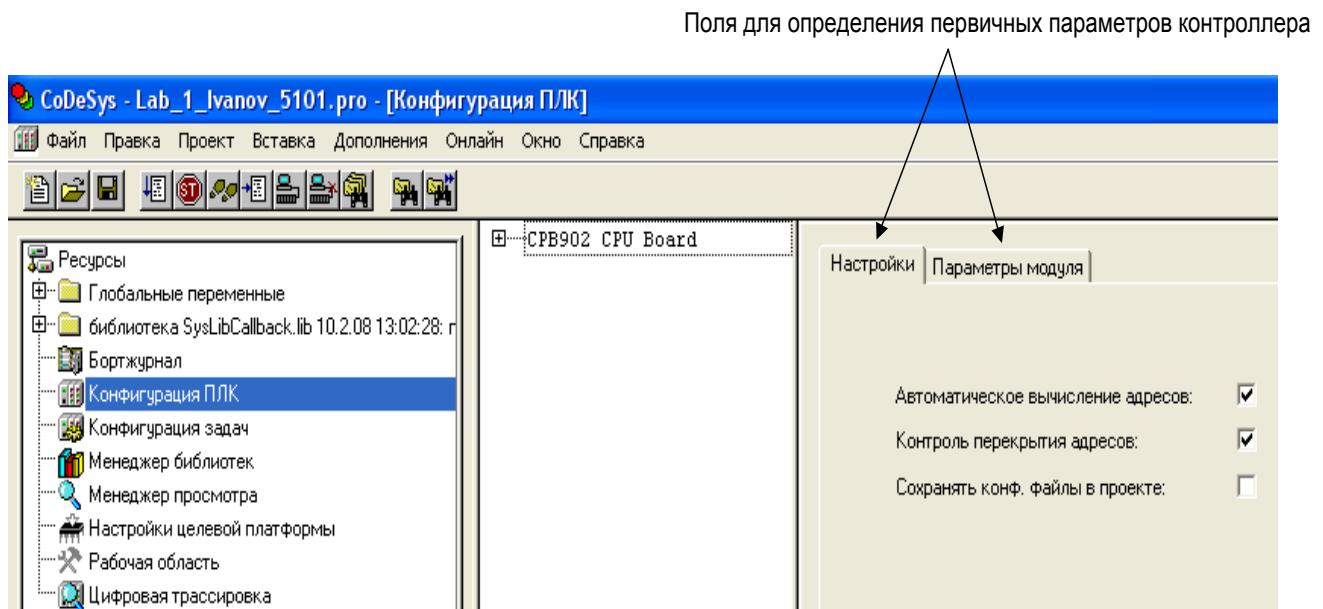


Рис. 5. Окно конфигурации первичных параметров контроллера

1.4.3. Раскрыть список конфигурации **CPB902 CPU Board**. Основными элементами списка конфигураций контроллера являются:

- **Application Diagnostics** – представление диагностической информации приложения;
- **I/O Modules** – конфигурация модулей ввода-вывода;
- **Ethernet1, Ethernet2** – конфигурация интерфейсов Ethernet;
- **COM2-COM6** – конфигурация коммуникационных портов.

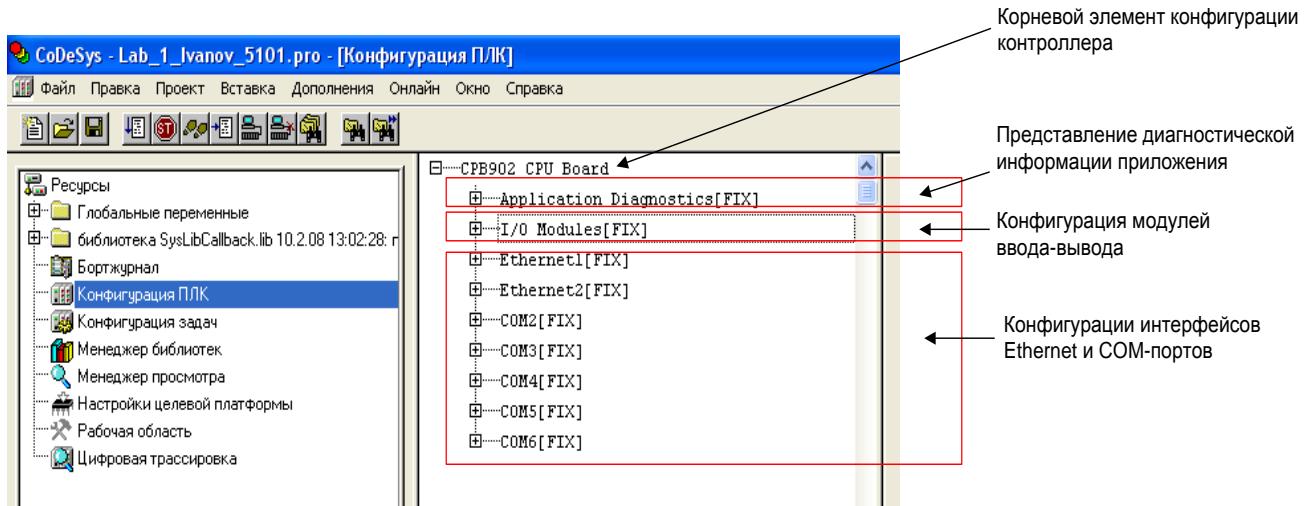


Рис. 5. Элементы конфигурации контроллера

1.4.4. Для того, чтобы модули **Fastwel I/O** были доступны из программы, следует добавить их конфигурацию. Для этого перейти к элементу **I/O Modules** описания модулей ввода-вывода, щелкнуть правой кнопкой мыши на элементе **I/O Modules** и выбрать в появившемся контекстном меню пункт «**Добавить подэлемент**». В списке найти описание нужного модуля и щелкнуть на него левой кнопкой мыши. Соответствующая конфигурация будет добавлена в список описания модулей ввода-вывода.

Начинать формирование списка следует с модуля, расположенного первым нашине FBUS. Однако следует учитывать, что в программную конфигурацию не добавляются вспомогательные модули OM796, OM758, OM759 и OM750, их описание в списке подэлементов отсутствует.

Набор функциональных модулей Fastwel I/O в составе контроллера и порядок установки приведены в Таблице 2 (Приложение В). В соответствии с ней сформируем список **I/O Modules**.

- 1) Первым на шине FBUS установлен модуль OM796 с порядковым номером N1 (см. Таблицу 2). Как было указано выше, его описание в списке элементов **I/O Modules** отсутствует, соответственно в программной конфигурации он не будет отображен.

2) В программную конфигурацию первым добавляется описание модуля **AIM722** как показано на рис 6. Конструктивно модуль расположен вторым и имеет порядковый номер N2 (см. Таблицу 2).

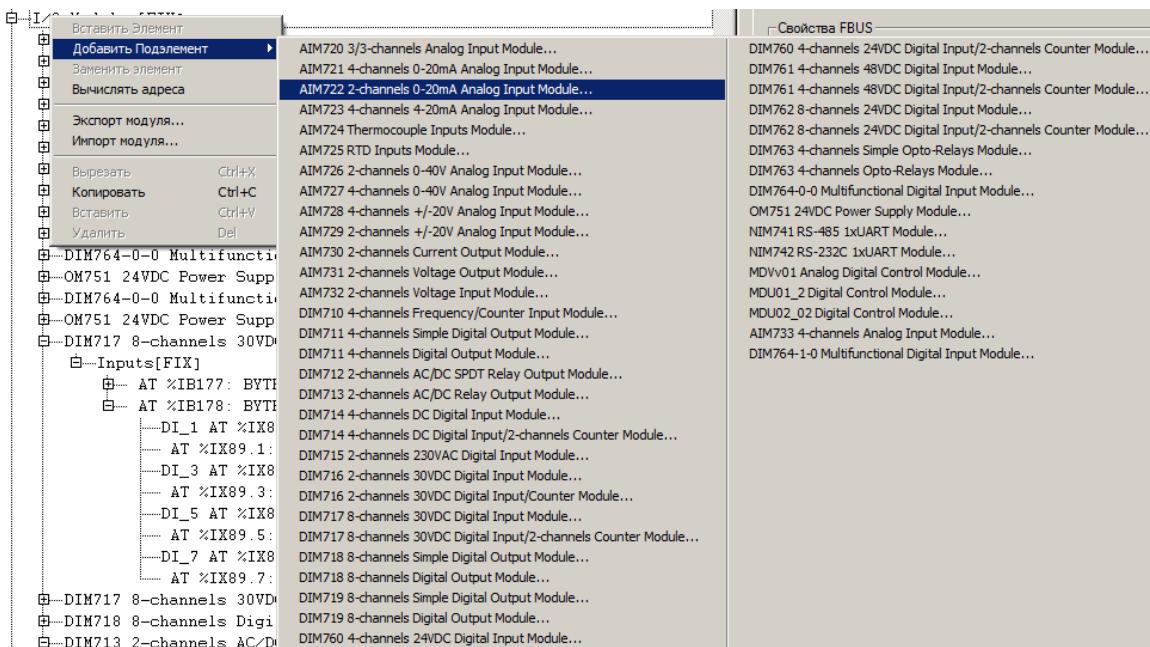


Рис. 6. Добавление описания модуля AIM722 в список I/O Modules

3) Следующим подэлементом так же будет модуль **AIM722**, поскольку два одинаковых модуля расположены в конструкции контроллера рядом. Затем добавляется модуль **AIM724** и т.д. Последними в программной конфигурации будут два одинаковых модуля **DIM713** (напомним, модуль **OM750** не добавляется, хотя физически он расположен последним в конструкции контроллера).

4) Итоговый список **I/O Modules**, соответствующий набору модулей Fastwel IO в составе контроллера (Приложение В) отображен на рис. 7.

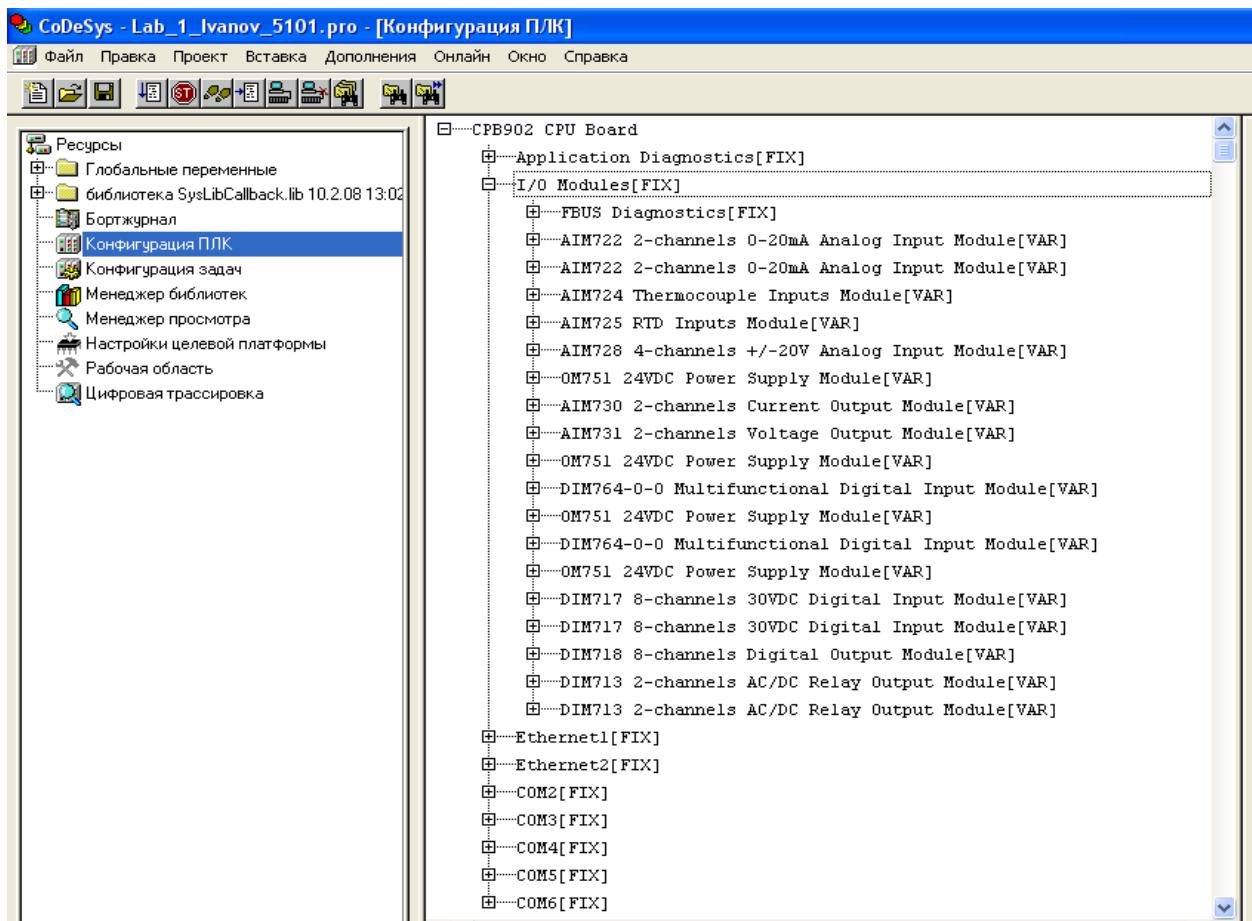


Рис. 7. Элементы I/O Modules описания модулей ввода-вывода

1.4.5. Для того, чтобы порт Ethernet1 контроллера был доступен из программы, следует его корректно сконфигурировать. Для этого перейти к элементу конфигурации **Ethernet1**. В появившемся справа окне (рабочей области) **Панель свойств Fastwel** отображаются сетевые параметры, которые будут транслированы в контроллер **CPM902** при загрузке в него прикладной программы.

В соответствии со структурной схемой учебно-лабораторного комплекса (Приложение А), за каждым узлом локальной сети учебно-лабораторного комплекса закреплен свой статический **IP-адрес**.

Контроллеры СPM902 имеют IP-адреса:

- 192.168.1.2** (маска подсети 255.255.255.0) – контроллер шкафа ШК1;
- 192.168.1.4** (маска подсети 255.255.255.0) – контроллер шкафа ШК2.

На рисунках 8 и 9 показано как должны выглядеть конфигурации интерфейса Ethernet1 для контроллеров СРМ902 шкафов ШК1 и ШК2. В соответствии с ними следует поменять настройки **Ethernet1** в своем проекте в зависимости от того, в какой из двух контроллеров будет перед загружена прикладная программа. Сохранить проект.

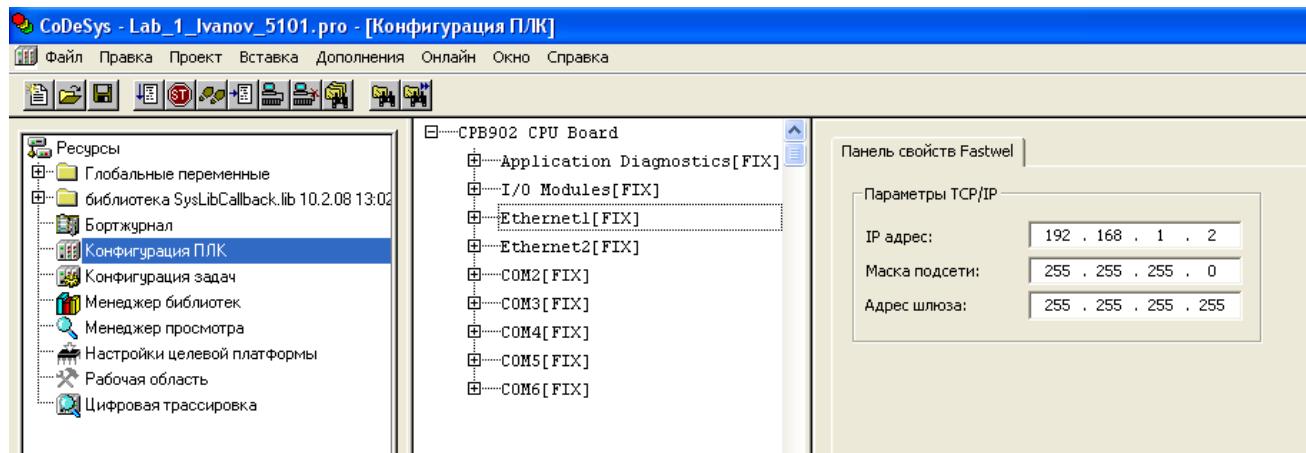


Рис. 8. Конфигурация контроллера СРМ902 (ШК1)

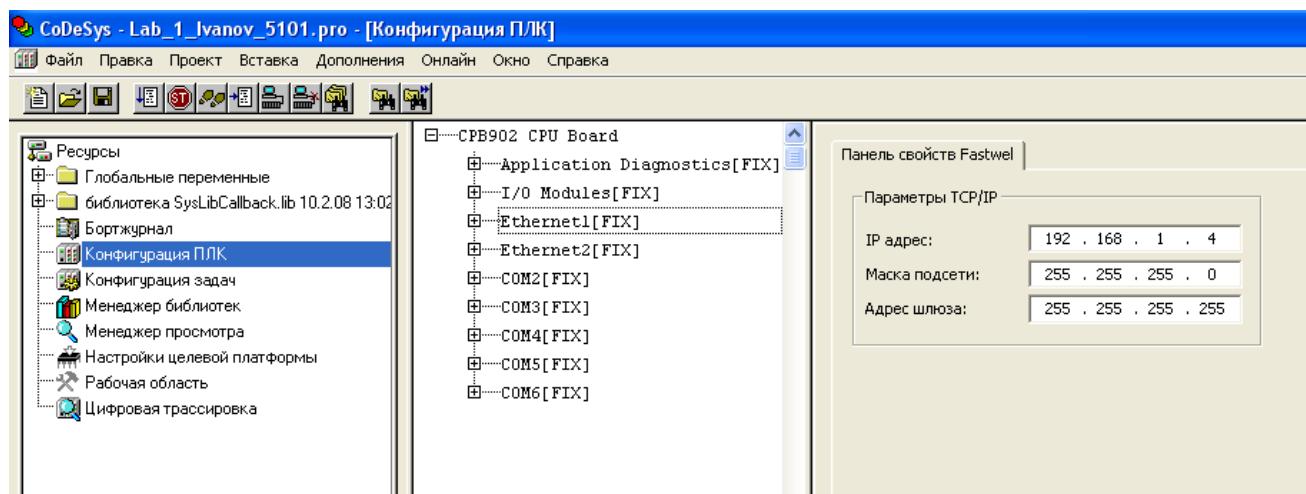


Рис. 9. Конфигурация контроллера СРМ902 (ШК2)

1.4.6. Сконфигурируем протокол сетевого обмена интерфейса Ethernet1. К аппаратному интерфейсу Ethernet1 можно подключить сервис сетевого обмена по протоколу Modbus (в качестве мастера и/или подчиненного узла). По умолчанию этот сервис не подключен и установлен элемент «**Not used**». Раскрыть список параметров элемента **Ethernet1**. Щёлкнуть правой кнопкой мыши на параметре «**Not used**», выбрать в появившемся контекстном меню пункт «**Заменить элемент...**» и установить «**Modbus TCP Slave**» (рис. 10). Этот сервис реализует функциональность подчиненного узла (сервера) протокола Modbus и используется для обмена данными между прикладной программой контроллера CPM902 и другим устройством – мастером сети Modbus.

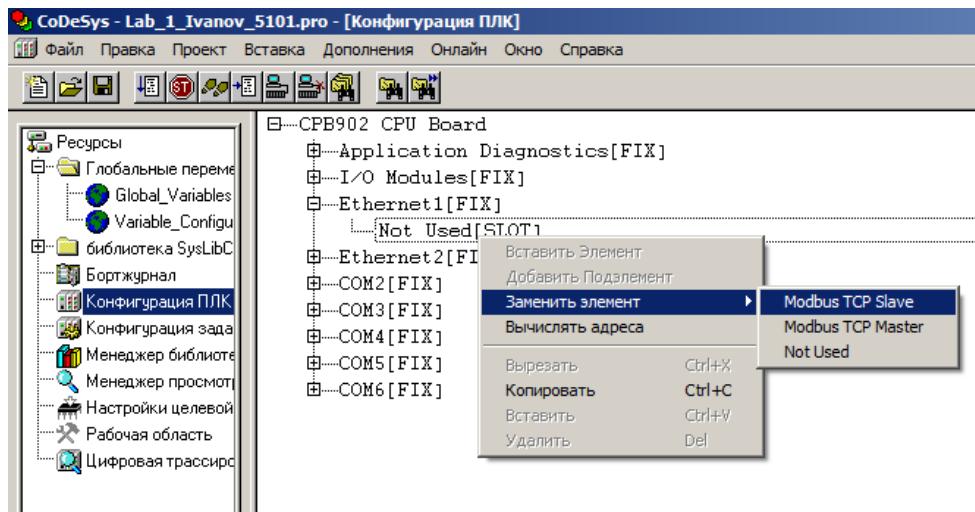


Рис.10. Конфигурирование сервиса ModbusTCP Slave на интерфейсе Ethernet1

Работа обоих контроллеров CPM902 (шкафов ШК1 и ШК2) в составе учебно-лабораторного комплекса в режиме подчиненного узла (сервера) Modbus является основным методом взаимодействия с другими узлами учебно-лабораторного комплекса. Функции мастеров сети MODBUS выполняют локальные панели операторов **WT3010**, а так же **Modbus OPC сервера**, установленные на АРМ студентов и преподавателя.

Однако эксплуатация учебно-лабораторного комплекса не исключает возможности использовать сервис мастера узла на контроллерах СРМ902, если это будет определено лабораторной задачей. Для добавления этого функционала нужно вместо элементов «**Not used**» и «**Modbus TCP Slave**» выбрать элемент «**Modbus TCP Master**».

## 1.5 ОРГАНИЗАЦИЯ СВЯЗИ ПРИКЛАДНОЙ ПРОГРАММЫ С ОБЛАСТЯМИ ВХОДНЫХ И ВЫХОДНЫХ ДАННЫХ ОКРУЖЕНИЯ ПРОГРАММЫ

Каналы модулей ввода-вывода и коммуникационные объекты внешней сети организуют окружение программы. Модель окружения представляется так называемым **образом процесса**, который состоит из двух областей памяти с непересекающимися адресами.

Первая область образа процесса, называемая **областью входных данных**, предназначена для хранения и обновления значений входных данных прикладной программы в процессе приёма информации от устройств ввода-вывода и сетевых интерфейсов.

Вторая область называется **областью выходных данных** и предназначена для хранения и обновления значений выходных данных программы в процессе выдачи информации устройствам ввода-вывода и в сетевые интерфейсы.

Среда разработки CoDeSys поддерживает три способа организации ссылок из прикладной программы на образ процесса:

- 1) Посредством создания символьных имен для каналов ввода-вывода в ресурсе **Конфигурация ПЛК**.
- 2) Путем использования конфигурационных переменных (VAR\_Config) в ресурсе **Глобальные переменные**.

3) Путем использования непосредственной адресации при объявлении и инициализации переменных.

Первый из описанных способов создания символьных имен является простым и удобным, хотя и имеет определенные ограничения – не позволяет выполнять отображение структур и массивов. В данной лабораторной работе не предполагается использование подобных структур, поэтому воспользуемся этим способом.

Создадим символьные имена для каналов модуля **DIM717** и двух модулей **DIM713**.

1.5.1. В ресурсе **Конфигурация ПЛК** вернуться к элементу **I/O Modules** и найти в списке описание модуля DIM717, которое соответствует реальному модулю с порядковым номером N18 в конструкции контроллера (в соответствии с Таблицей 2, Приложения В).

1.5.2. Раскрыть секцию **Input** внутри описания модуля и перейти к каналу "AT %IB178", отражающему состояние входов. По умолчанию данный канал имеет имя **InputStates** в поле комментариев. Для удобства можно в поле комментариев изменить имя на **N18\_InputStates** (Рис. 11).

1.5.3. Биты данного элемента отражают текущее состояние соответствующих входных каналов модуля DIM717. Выбрать бит, первый по списку: AT %IX89.0 : BOOL (\*Bit 0\*).

Дважды щелкнуть левой кнопкой мыши слева от надписи "AT%..." и ввести имя создаваемой переменной **DI\_1**.

Нажать клавишу «**Enter**».

1.5.4. Последовательно переходя на 3-ий, 5-ый и 7-ой биты, создать на входных каналах соответствующие символьные имена **DI\_3**, **DI\_5**, **DI\_7** (рис. 12).

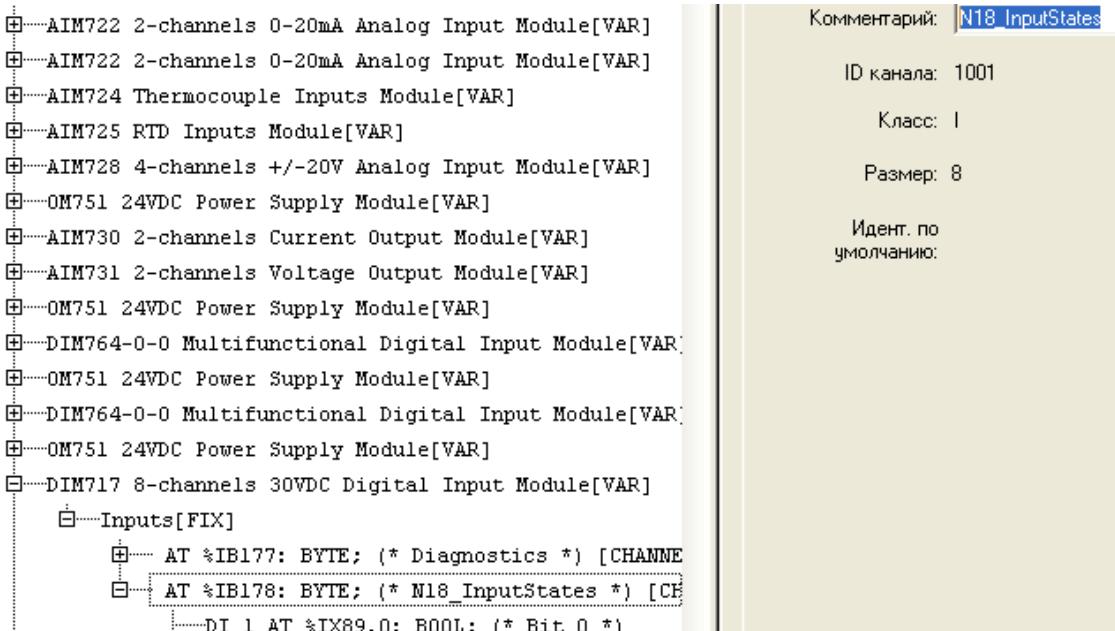


Рис. 11. Редактирование поля комментария для канала, отражающего состояние входов модуля DIM717

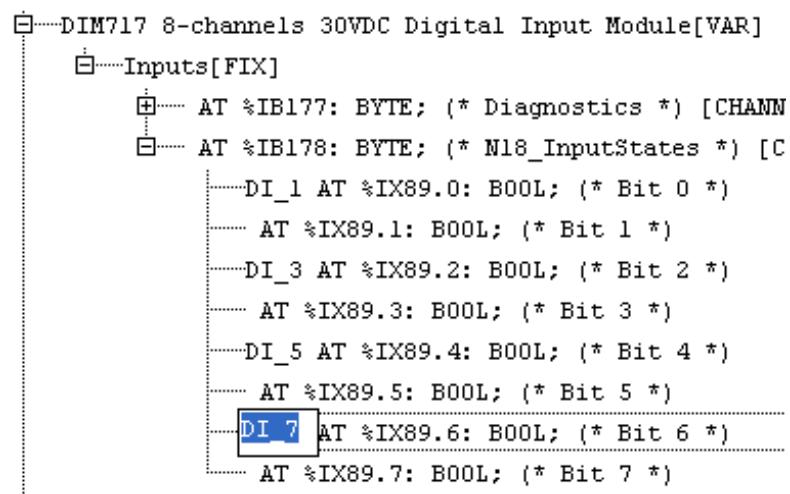


Рис. 12. Созданные символьные имена для каналов модуля DIM717 (переменные DI\_1, DI\_3, DI\_5, DI\_7)

1.5.5. Перейти к описанию модуля DIM713, который соответствует реальному модулю с порядковым номером N22 в конструкции контроллера (в соответствии с Таблицей 2, Приложение В).

1.5.6. Раскрыть список **Output** и перейти к элементу **OutputsControl**. Для удобства можно в поле комментариев изменить имя на **N22\_OutputsControl**.

1.5.7. Первые два бита данного элемента предназначены для управления 2-мя каналами релейной коммутации. Назначить им символьные имена **N22\_relay\_1** и **N22\_relay\_2** соответственно (рис. 13).

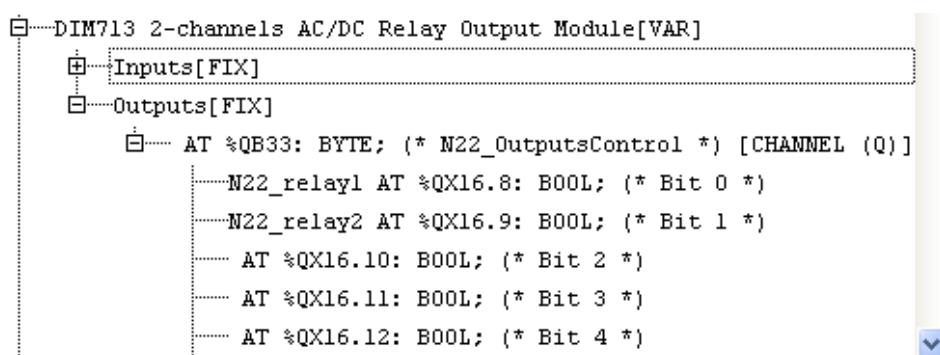


Рис. 13. Созданные символьные имена каналов модуля DIM713  
(переменные N22\_relay\_1 и N22\_relay\_2)

1.5.8. Перейти к описанию модуля DIM713, который соответствует реальному модулю с порядковым номером N23 в конструкции контроллера (в соответствии с Таблицей 2, Приложение В).

1.5.9. Повторяя действия п.1.5.5 и 1.5.6 создать переменные с символьными именами **N23\_relay\_1** и **N23\_relay\_2**.

Созданные в п. 1.5.1 – 1.5.7 переменные являются **глобальными для всего проекта**.

## 1.6 РЕДАКТИРОВАНИЕ ГЛАВНОЙ ПРОГРАММЫ PLC\_PRG POU

Перейти на вкладку **Ресурсы** организатора объектов, расположенного в нижней области главного окна среды разработки CoDeSys

1.6.1. Перейти на вкладку **POU**, расположенной в нижней области главного окна среды разработки CoDeSys.

1.6.2. Дважды щелкнуть на единственном элементе – программном модуле **PLC\_PRG (POU)**, который был определен при создании проекта (п. 1.3.4).

Редактор программного компонента POU **содержит поле ввода кода программы и раздел объявления переменных**, разделенные горизонтальной границей, как показано на рис. 14.

В данной лабораторной работе в разделе объявлений **PLC\_PRG POU** никакие переменные декларироваться не будут, поскольку в программе будем оперировать только переменными с символьными именами, созданными в п.п. 1.5.1 – 1.5.8.

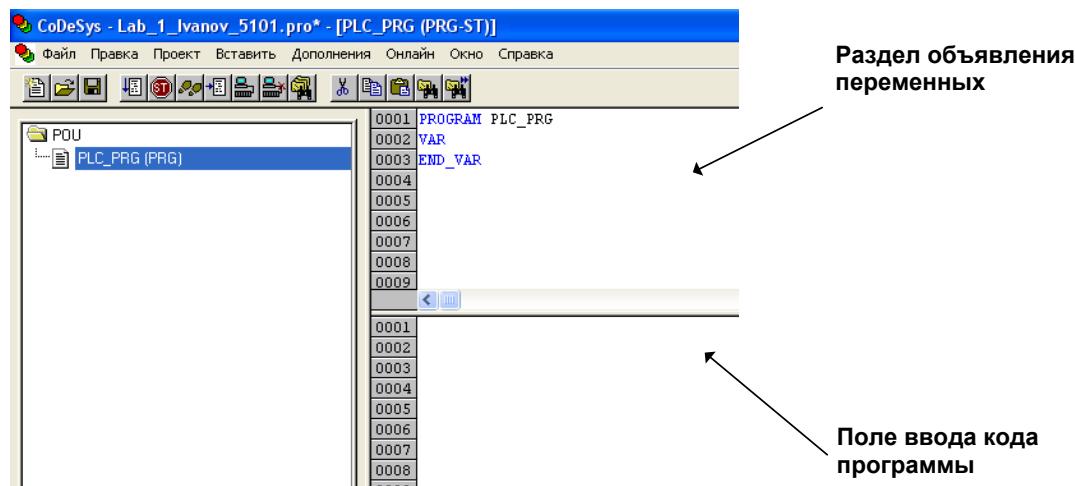


Рис. 14. Окно редактирования POU

1.6.3. Перейти в поле ввода кода.

В **PLC\_PRG POU** будет использоваться язык программирования ST (Structured Text), что было определено при создании проекта.

```
N22_relay1 := DI_1;      (* DIM713 (N22) -реле 1 управляетя от DIM717 (N18) -канал 1 *)
N22_relay2 := DI_3;
N23_relay1 := DI_5;      (* DIM713 (N23) -реле 1 управляетя от DIM717 (N18) -канал 4 *)
N23_relay2 := DI_7;
```

Как уже описывалось в разделе 1.5, созданные в **Конфигурации ПЛК** переменные с символьными именами доступны в главной программе **PLC\_PRG POU** как глобальные переменные. Для их ввода можно воспользоваться ассистентом ввода. Для этого нажать клавишу **F2** и выбрать раздел **Глобальные переменные**. Последовательно выбирая соответствующие переменные, ввести в PLC\_PRG программный код.

#### **Примечание:**

- Комментарий к программному коду помещается внутри комбинации символов **(\* и \*)**. Комментарий может находиться в любом месте поля ввода кода и содержать произвольный неформализованный текст.
- В окне ресурса **Конфигурация задач** не было добавлено ни одной задачи – примеры использования множества циклических и ациклических задач в одном проекте будут рассмотрены в следующих лабораторных работах. Поэтому среда разработки CoDeSys автоматически создаст описание циклической задачи с именем *DefaultTask* и ассоциирует с ней программу PLC\_PRG. Период цикла данной задачи будет равен значению параметра *CPB902 CPU Board:EventsRate* (по умолчанию – 10 мс) в окне ресурса **Конфигурация ПЛК**.

#### 1.6.4. Сохранить проект.

1.6.5. Компиляция проекта – выполнить команду меню **Проект** – **Компилировать (F11)**. Результат процесса компиляции будет указан в самом нижнем окне редактора CoDeSys. Если будут обнаружены ошибки в программе, то они будут указаны в этом же окне.

## 1.7 ОТЛАДКА ПРОГРАММЫ В РЕЖИМЕ ЭМУЛЯЦИИ

Во время эмуляции созданная программа выполняется не в ПЛК, а в компьютере, на котором запущена среда разработки CoDeSys. В этом режиме допустимы все функции онлайн, что позволяет проверить логическую правильность программ, не используя контроллер.

1.7.1. Выбрать пункт меню **Онлайн – Режим эмуляции**, и включить флагок, как показано на рис. 15.

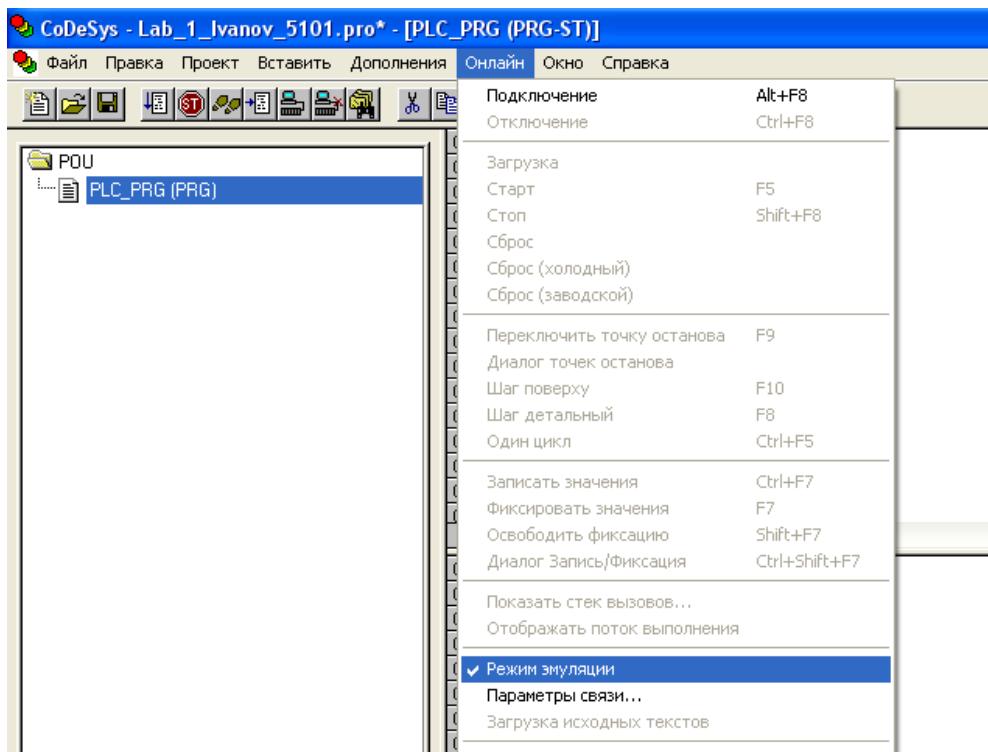


Рис. 15. Переход в режим эмуляции контроллера

1.7.2. Подключить среду разработки CoDeSys к «виртуальному» контроллеру, выполнив команду меню **Онлайн – Подключение** (Alt + F8).

1.7.3. Запустить исполнение программного кода через команду меню **Онлайн – Старт (F5)**.

1.7.4. Среда CoDeSys перейдет в режим эмуляции. Редактор программы **PLC\_PRG** примет вид как показано на рис.16: в левой части области кода ввода будет отображаться непосредственно код программы, а в правой – состояние её переменных.

1.7.5. Дважды щелкнуть на переменную **DI\_1** в правой части окна. Редактор укажет возможность замены значения **False** (логический «0») этой переменной на значение **True** (логическая «1»).

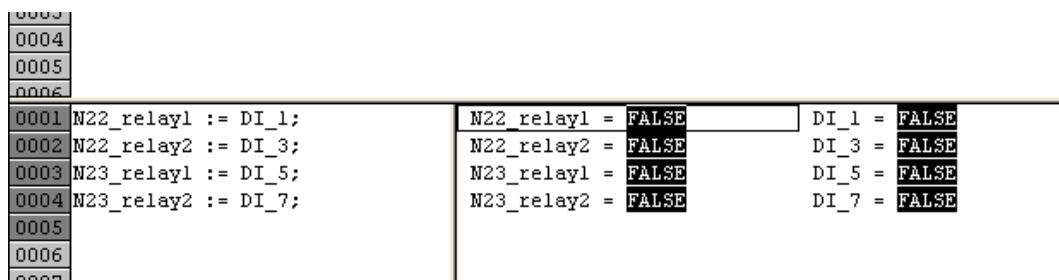


Рис. 16. Вид редактора PLC\_PRG POU в режиме эмуляции

1.7.6. Выполнить команду меню **Онлайн – Записать значение (Ctrl+ F7)**. Переменная **DI\_1** установится в значение **True**. После чего, в соответствии с алгоритмом программы, значение **True** примет и переменная **N22\_relay1**.

1.7.7. Провести действия п.п. 1.7.5. и 1.7.6 с переменными **DI\_3**, **DI\_5**, **DI\_7**, меняя их значения и управляя переменными, связанными с релейными выходами.

1.7.8. Просматривать и менять значения переменных, созданных через символьные имена для каналов ввода-вывода, можно и в ресурсе **Конфигурация ПЛК**.

Для этого перейти на вкладку **Ресурсы – Конфигурация ПЛК**, выбрать элемент **I/O Modules** и раскрыть списки модулей DIM717 (N18), DIM713 (N22), DIM713 (N23).

Перейти к каналу с символьным именем **DI\_1** и двойным щелчком мыши на квадрате слева поменять его цвет с синего на белый. Это будет означать, что значение **True** (логическая «1») переменной **DI\_1** сменилось на значение **False** (логический «0»). Поменяв цвет квадрата обратно с белого на синий, вернуться к значению **True**.

В соответствии с алгоритмом программы, переменная **N22\_relay1** так же меняет значение (цвет «своего» квадрата) в прямой зависимости от значения переменной **DI\_1** (рис. 17).

1.7.9. Завершение работы в режиме эмуляции осуществляется по команде меню **Онлайн – Отключение (Ctrl + F8)**.

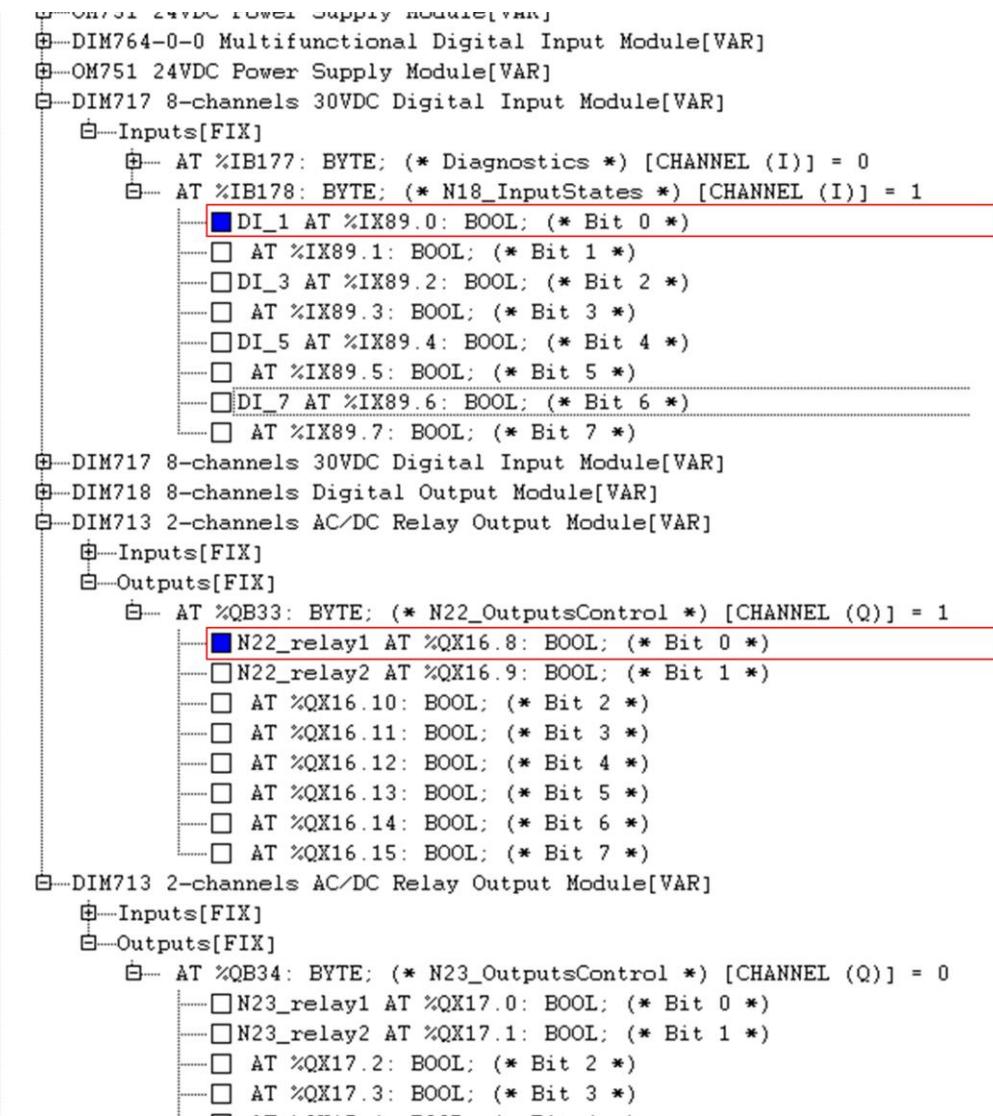


Рис. 17. Мониторинг и управление переменными из ресурса Конфигурация ПЛК

## 1.8 СОЗДАНИЕ ЛОГИЧЕСКОГО ИНФОРМАЦИОННОГО КАНАЛА МЕЖДУ СРЕДОЙ РАЗРАБОТКИ И КОНТРОЛЛЕРОМ СРМ902

Перед началом выполнения любых операций с удаленным контроллером, в том числе и загрузки в него прикладной программы, должна быть выполнена настройка параметров драйвера коммуникационного сервера **CoDeSys Gateway Server**. Этот сервер обеспечивает возможность работы через COM – порт (**P2P–соединение – «point-to-point»**) или через сеть **Ethernet** (как правило, по протоколу Modbus).

Настройка CoDeSys Gateway Server заключается в создании соответствующего логического информационного канала.

Создание логического информационного канала между средой разработки и контроллером по **ПОСЛЕДОВАТЕЛЬНОМУ КАНАЛУ СВЯЗИ (P2P)**:

1.8.1. При **выключенном питании (!!?)** контроллера CPM902 его порт COM1 соединить с последовательным портом компьютера COM2 при помощи нуль-модемного кабеля.

1.8.2. Выбрать команду меню **Онлайн-Параметры связи.** На экран будет выведена диалоговая панель **Communication Parameters.**

1.8.3. Нажать кнопку **New** и в появившейся диалоговой панели ввести имя создаваемого канала (или оставить имя по умолчанию '**localhost**'), а в списке **Device** выбрать строку « **P2P Fastwel P2P driver** », как показано на рис.18. Закрыть диалоговую панель нажатием кнопки **OK**.

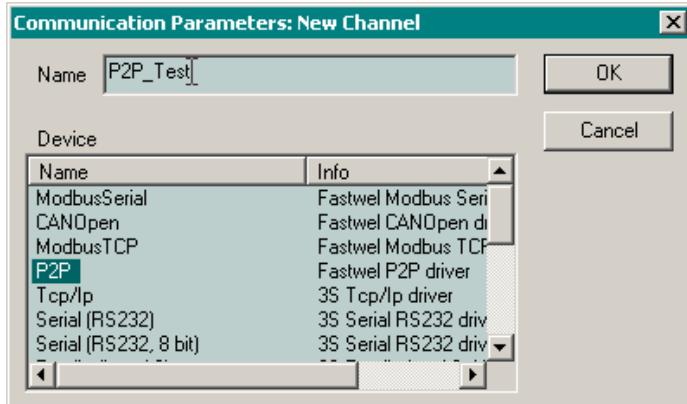


Рис. 18. Создание логического информационного канала  
по последовательному каналу связи (P2P)

1.8.4. В древовидном списке **Channels** появится элемент, соответствующий созданному каналу, а в таблице параметров канала справа – параметры созданного канала, как показано на рис. 19.

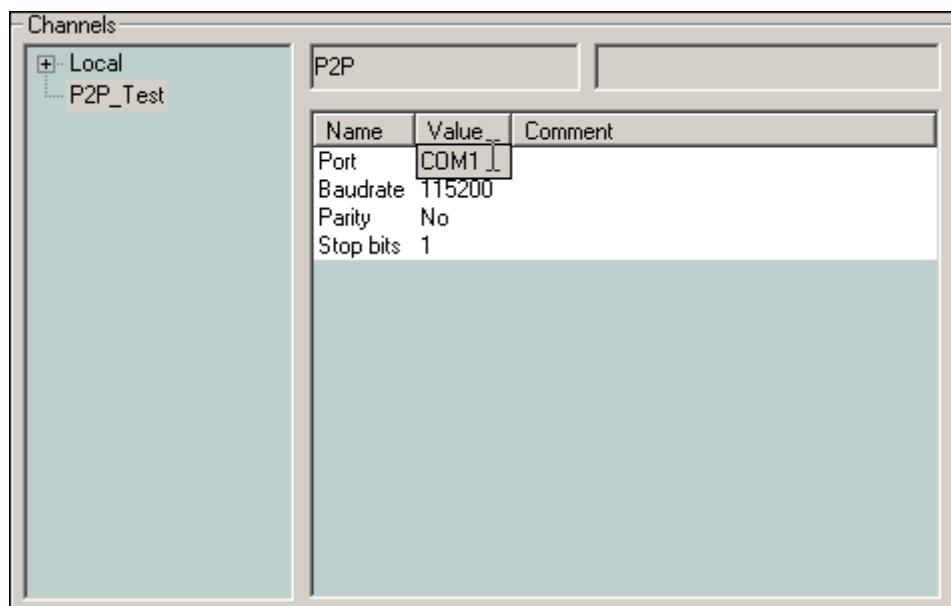


Рис. 19. Настройка параметров канала Р2Р

1.8.5. По умолчанию для связи с контроллером указан порт COM1.

В АРМ студентов комплекса используется порт COM2, поэтому необходимо дважды щелкнуть левой кнопкой мыши над именем **COM1** в таблице параметров и клавишами "стрелка вверх" или "стрелка вниз" выбрать последовательный порт компьютера порт **COM2**, через который будет осуществляться взаимодействие с контроллером в режиме "точка-точка", и нажать клавишу **Enter**.

Закрыть диалоговую панель **Communication Parameters** нажатием кнопки **OK**.

Создание логического информационного канала между средой разработки и контроллером по протоколу **MODBUS TCP**:

1.8.6. Выбрать команду меню **Онлайн-Параметры связи**. На экран будет выведена диалоговая панель **Communication Parameters**.

1.8.7. Нажать кнопку **New** и в появившейся диалоговой панели ввести имя создаваемого канала (или оставить имя по умолчанию 'localhost'), а в списке **Device** выбрать строку «**ModbusTCP Fastwel Modbus TCP driver**», как показано на рис. 20.

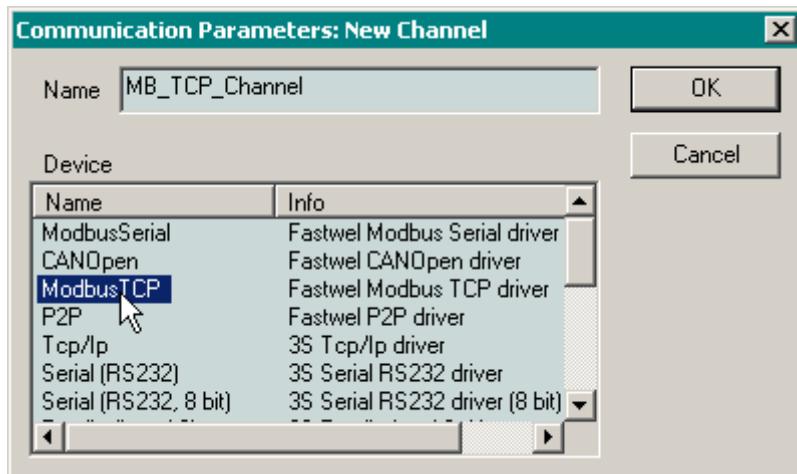


Рис. 20. Создание логического информационного канала  
по протоколу **MODBUS TCP**

1.8.9. Закрыть диалоговую панель нажатием кнопки **OK**. В древовидном списке **Channels** появится элемент, соответствующий созданному каналу, а в таблице параметров канала справа – параметры созданного канала с адресом 10.0.0.2 по умолчанию.

1.8.10. Дважды щелкнуть на значении параметра **Address** и ввести значение IP-адреса контроллера **СРМ902**, с которым предполагается установить связь.

Контроллеры **СРМ902** имеют IP-адреса:

**192.168.1.2** (маска подсети 255.255.255.0) – контроллер шкафа ШК1;

**192.168.1.4** (маска подсети 255.255.255.0) – контроллер шкафа ШК2.

Окно настройки канала примет вид, как показано на рис. 21. Нажать **OK**.

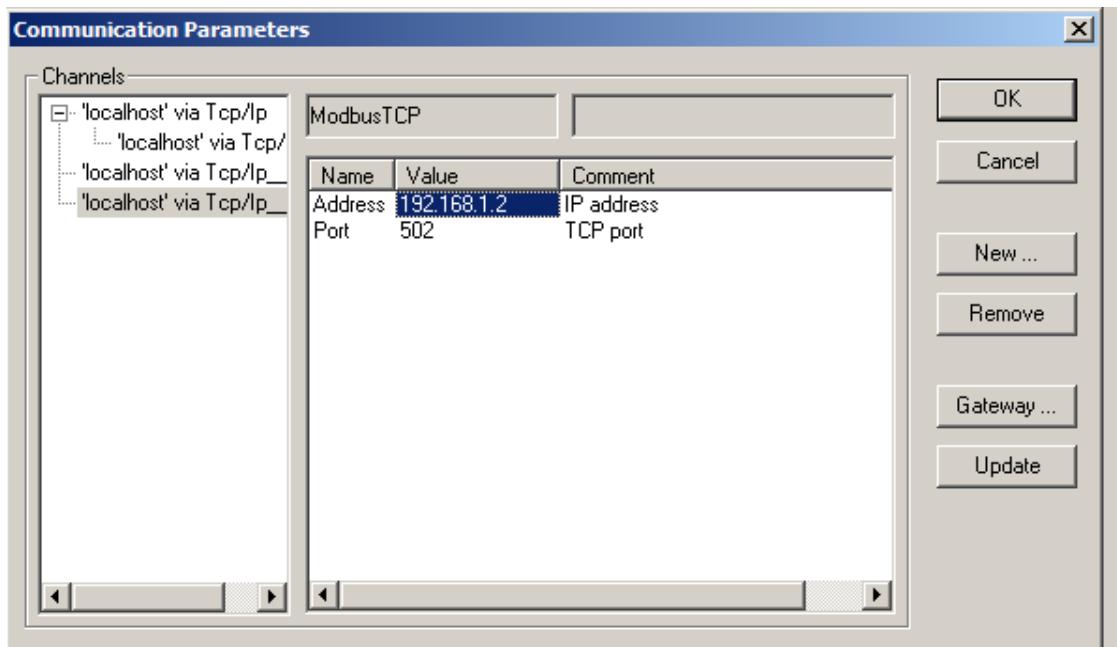


Рис. 21. Настройка параметров интерфейса Ethernet

1.8.11. Закрыть диалоговую панель **Communication Parameters** нажатием кнопки **OK**.

В ходе выполнения **Лабораторной работы №1** для приобретения соответствующих навыков следует самостоятельно создать оба логических информационных канала, проверить их работоспособность путем загрузки через каждый прикладной программы в контроллер СРМ902 (раздел 1.9). Однако в обычной практике, как правило, удобнее пользоваться каналом через интерфейс **Ethernet**, используя протокол **Modbus**. В последующих работах именно он и будет использоваться. Связь по последовательному каналу связи через **P2P** будет применяться в исключительных случаях, например, тогда, когда в контроллер загружена программа, содержащая неправильный IP – адрес или некорректный код.

## 1.9 ЗАГРУЗКА ПРИЛОЖЕНИЯ В КОНТРОЛЛЕР

1.9.1. Выбрать пункт меню **Онлайн – Режим эмуляции**, и выключить флажок. Теперь выполнение команды **Онлайн-Подключение** будет сопровождаться попыткой подключения к удаленному контроллеру.

1.9.2. Выбрать пункт меню **Онлайн-Параметры связи**. Создать или выбрать ранее созданный логический информационный канал (раздел 1.8), который будет использоваться для связи с контроллером.

Если будет выбрано P2P – соединение, то выключить питание!!!! контроллера СРМ902 и соединить его порт COM1 с последовательным портом компьютера при помощи нуль-модемного кабеля.

1.9.3. Выбрать команду **Онлайн – Подключение (Alt + F8)** для подключения к контроллеру. При успешном соединении на экран будет выведена диалоговая панель, требующая подтвердить замену программы в контроллере. Нажать **Да**.

На экран будет выведено окно, отображающее ход загрузки программы.

1.9.4. По завершении загрузки программы в контроллер, выполняется конфигурирование в соответствии с содержимым секций **Ресурсы** и оборудование переключается на новое приложение.

## 1.10 ОНЛАЙН ОТЛАДКА ПРОГРАММЫ, ПРОСМОТР И УСТАНОВКА ЗНАЧЕНИЙ ПЕРЕМЕННЫХ

Предполагается, что прикладная программа загружена в контроллер, а в среде CoDeSys открыт проект, в котором разрабатывалась данная программа. Для просмотра значений переменных подключить среду разработки CoDeSys к контроллеру СМР902, выполнив команду меню **Онлайн – Подключение (Alt + F8)**.

Онлайн отладка программы, просмотр и установка значений переменных проводятся по тем же принципам, как это делалось в режиме эмуляции (раздел 7). Отличие будет состоять лишь в том, что необходимо подавать реальное воздействие на входные каналы модуля DIM717 (N18) в строгом соответствии со схемой подключения и отслеживать соответствующее изменение состояния релейных выходов модулей DIM713 (N22, N23) в конструкции контроллера.

**РАЗДЕЛ 2. ЛАБОРАТОРНАЯ РАБОТА №2**

**«ПРИНЦИП РАБОТЫ И СПОСОБЫ КОНФИГУРИРОВАНИЯ  
КОНТРОЛЛЕРА FASTWEL CPM902 В КАЧЕСТВЕ ПОДЧИНЕННОГО  
УЗЛА (СЕРВЕРА) СЕТИ MODBUS»**

**2.1 ОПИСАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ №2**

**Назначение:**

изучение принципов сетевого взаимодействия по протоколу Modbus.

**Цель работы – освоение практических навыков:**

- конфигурирования контроллера CPM902 в качестве подчиненного узла (сервера) Modbus для сетевого обмена данными с другими устройствами комплекса;
- использования Fastwel Modbus OPC-сервера для мониторинга и отладки работы сервера Modbus.

**Используемое оборудование:**

- программируемый контроллер Fastwel IO CPM902 с набором функциональных модулей, расположенный в ШК1 и ШК2 в соответствии со структурной схемой комплекса (Приложение А). Модули соединены в единую конструкцию через пазы на корпусе каждого модуля в порядке, определенным в таблице 2 (Приложение В);
- кабель нуль-модемный;
- АРМ студента или преподавателя.

**Используемое инструментальное ПО:**

среда разработки **Fastwel IO Codesys Adaptation** (установлена на АРМ студентов и преподавателя).

**Используемое прикладное ПО:**

**Fastwel Modbus OPC Server** (установлена на АРМ студентов и преподавателя).

## **Теоретический материал для допуска к выполнению лабораторной работы:**

- описание комплекса технических средств [5];
- описание комплекса программного обеспечения [6].

### **2.2 ОБЩИЕ СВЕДЕНИЯ**

Основным методом сетевого взаимодействия оборудования учебно-лабораторного комплекса является обмен данными по стандартному промышленному протоколу **ModbusTCP** – версии протокола Modbus для IP сетей. Протокол Modbus реализован по принципу «клиент – серверного» опроса. В сети имеется **клиент**, так называемое «главное» (**master**) устройство, и несколько **серверов** – «подчиненных» (**slaves**) устройств (рис. 22). Главное устройство инициирует транзакции (передаёт запросы). Клиент Modbus может адресоваться индивидуально к подчиненному или инициировать передачу широковещательного сообщения для всех подчиненных устройств. Подчиненное устройство отвечает на запрос, адресованный именно ему. Версия протокола **ModbusTCP** и созданная на его основе сеть характерны тем, что допускают присутствия нескольких мастеров.

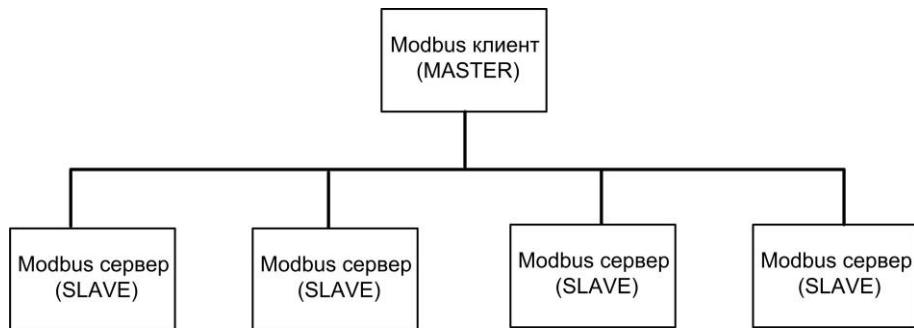


Рис. 22. Структура сети при сетевом взаимодействии по протоколу Modbus

Спецификация протокола Modbus определяет следующие коммуникационные объекты:

- **Input Register** (входной регистр) – объект, доступный только для чтения и представляющий 16-разрядное значение из области выходных данных подчиненного узла сети Modbus;
- **Holding Register** (выходной регистр) — объект, доступный для записи и чтения, и представляющий 16-разрядное значение в области входных данных подчиненного узла сети Modbus;
- **Discrete Input** (дискретный вход) — объект, доступный только для чтения и представляющий битовое поле в области выходных данных подчиненного узла Modbus;
- **Coil** (дискретный выход) – объект, доступный для записи и чтения, и представляющий битовое поле в области входных данных подчиненного узла Modbus.

В лабораторной работе №2 рассматривается интеграция контроллера СРМ902 в учебно-лабораторный комплекс в качестве сервера Modbus.

Работа обоих контроллеров СРМ902 (шкафов ШК1 и ШК2) в составе учебно-лабораторного комплекса в режиме подчиненного узла (сервера) Modbus определена структурной схемой комплекса. Функции клиентов (master) Modbus при этом выполняют:

- 1) Прикладное ПО – **Fastwel Modbus OPC Server**, установленное на АРМ студентов и преподавателя.
- 2) Интеллектуальные локальные панели оператора «**WT3010**», размещенные в шкафах ШК1 и ШК2.

Принципы взаимодействия этих Modbus – клиентов с Modbus – сервером контроллера СРМ902, отображены на рисунках 23 и 24.

Fastwel Modbus OPC Server в данной лабораторной работе используется как средство мониторинга и отладки работы сервера Modbus на контроллерах СРМ902 (раздел 1.10). Конфигурирование панели WT3010 и её интеграция в работу комплекса подробно описаны в лабораторной работе № 3.

В качестве базовой программы для контроллера СРМ902 будем использовать приложение, разработанное в лабораторной работе №1. Сохраняются все основные настройки и общая задача управления 4-мя релейными выходами (каналами модулей DIM713). Однако изменится сам принцип управления – команды на включение и отключение релейных выходов будут передаваться на контроллер в виде данных по протоколу Modbus от **мастера** сети (в процессе отладки – от **Fastwel Modbus OPC Server**).

#### **Последовательность операций при запуске контроллера СРМ902 в качестве подчиненного узла (сервера) сети Modbus:**

- создание проекта;
- редактирование конфигурации контроллера;
- редактирование параметров коммуникационных объектов сервера Modbus;
- редактирование главной программы **PLC\_PRG POU**;
- компиляция проекта;
- загрузка приложения в контроллер;
- мониторинг и отладка работы сервера Modbus.

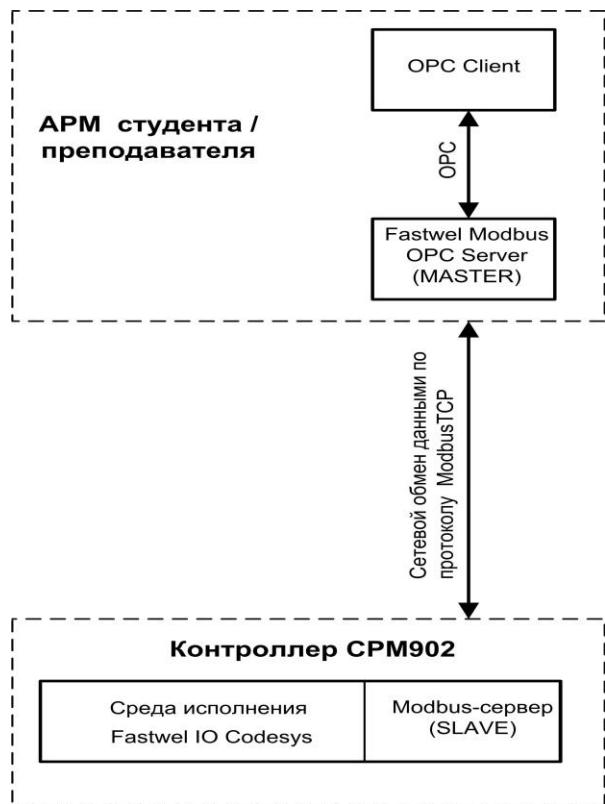


Рис. 23. Взаимодействие ПО Fastwel Modbus OPC Server и контроллера CPM902

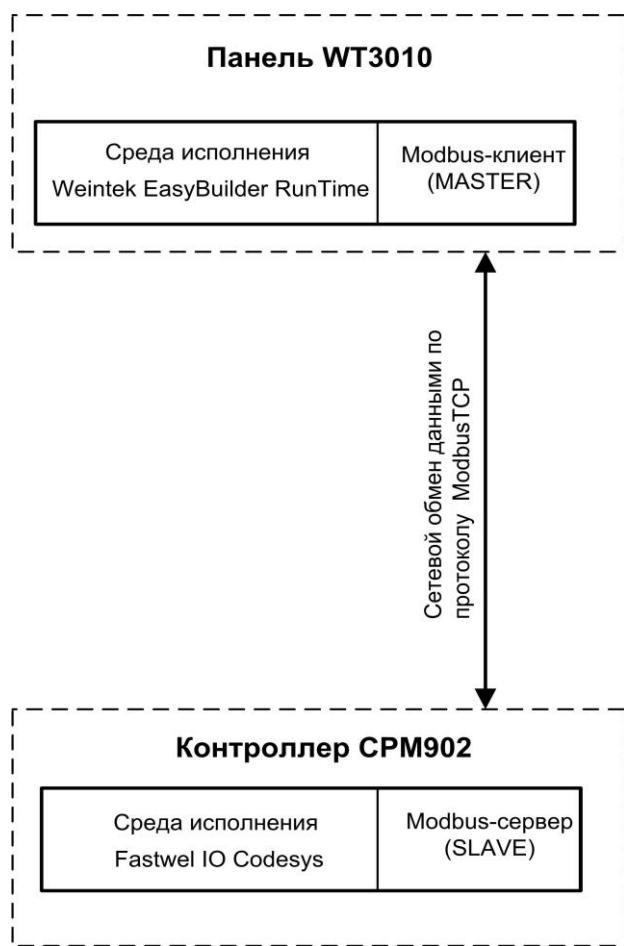


Рис. 24. Взаимодействие панели WT3010 и контроллера CPM902

## 2.3 СОЗДАНИЕ ПРОЕКТА

Как уже указывалось ранее, в данной работе в качестве базовой программы для контроллера СРМ902 будем использовать приложение, разработанное в лабораторной работе №1.

2.3.1. Средствами операционной системы **Windows 7** загрузить на исполнение среду разработки CoDeSys: **Пуск – Все программы – 3S Software V2.3 – CoDeSys V2.3.**

2.3.2. В запущенном редакторе среды Codesys выполнить команду меню **Файл – Открыть** и найти в файловой системе ПК собственный проект, который был ранее создан и сохранен в ходе выполнения лабораторной работы №1 (рис.25).

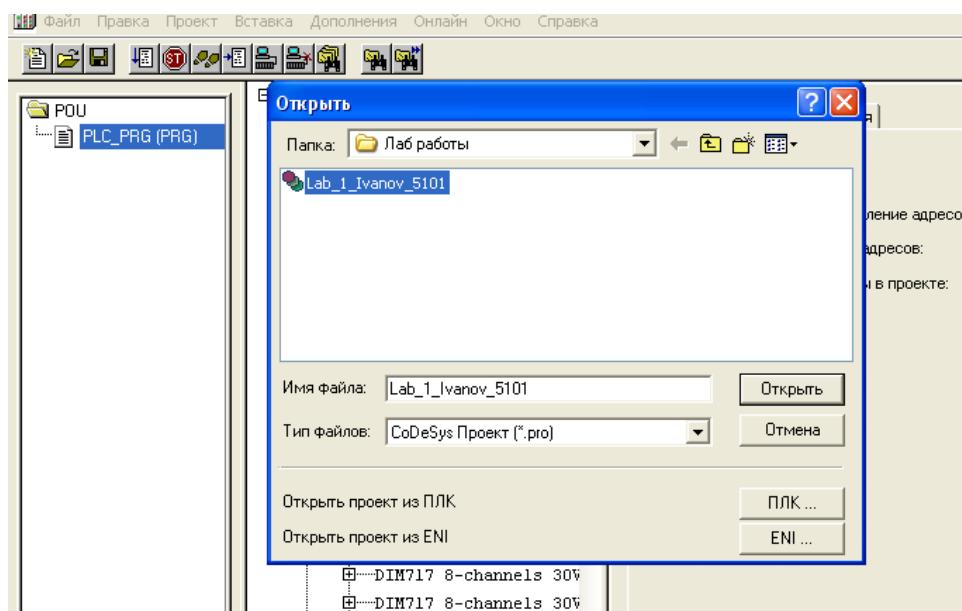


Рис. 25. Запуск ранее созданного проекта

2.3.3. Выполнить команду меню **Файл – Сохранить как** и поместить в тот же каталог проект с уже измененным именем **Lab\_2\_Ivanov\_5101.pro**.

## 2.4 РЕДАКТИРОВАНИЕ КОНФИГУРАЦИИ КОНТРОЛЛЕРА

2.4.1. Перейти на вкладку **Ресурсы**, расположенной в нижней области главного окна среды разработки CoDeSys, после чего дважды щелкнуть на названии ресурса **Конфигурация ПЛК**. В правой части редактора Codesys появится окно с основным элементом конфигурации **CPB902 CPU Board**, которое предназначено для создания и редактирования конфигурации контроллера Fastwel CPM902.

2.4.2. Перейти к элементу конфигурации **Ethernet1** и проверить соответствие настроек сетевых параметров. Контроллеры CPM902 имеют IP – адреса:  
**192.168.1.2** (маска подсети 255.255.255.0) – контроллер шкафа ШК1;  
**192.168.1.4** (маска подсети 255.255.255.0) – контроллер шкафа ШК2.

2.4.3. Раскрыть список параметров интерфейса **Ethernet1**. В исходном проекте к интерфейсу Ethernet1 уже был установлен параметр «**Modbus TCP Slave**», т.е. подключен сервис, реализующий функциональность подчиненного узла (сервера) Modbus. Проверить, что данный параметр не изменен (рис.26).

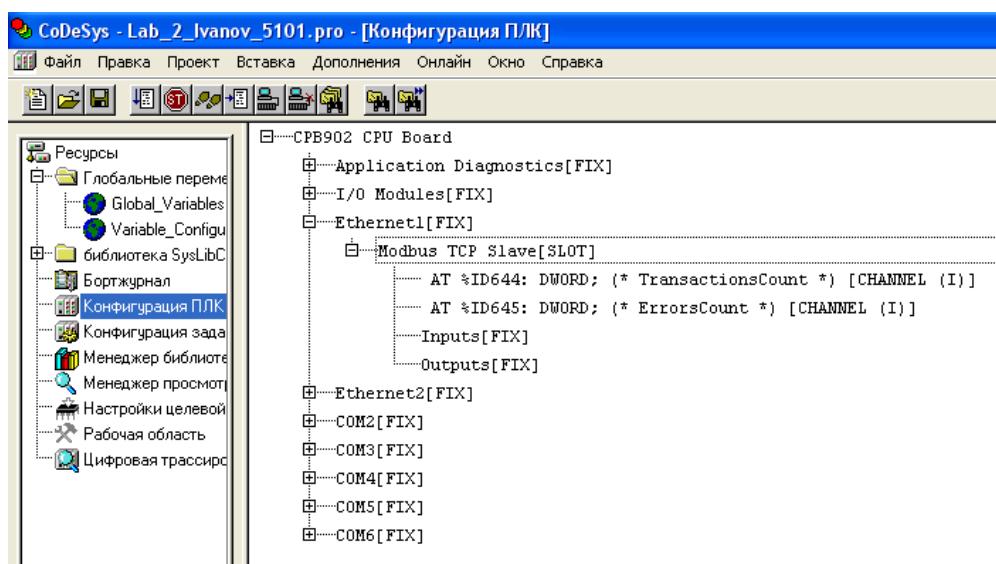


Рис. 26. К интерфейсу Ethernet1 подключен сервис подчиненного узла (сервера) Modbus

## 2.5 РЕДАКТИРОВАНИЕ ПАРАМЕТРОВ КОММУНИКАЦИОННЫХ ОБЪЕКТОВ СЕРВЕРА MODBUS

Области памяти контроллера СРМ902, отображаемые на множества регистров и битовых полей сервера MODBUS, описываются в секциях **Inputs** и **Outputs** в древовидном списке конфигурации сервера (рис. 26). Секция **Inputs** содержит список объектов доступа к данным, поступающим по сети от удаленных клиентов. Секция **Outputs** содержит список объектов доступа к данным для передачи в сеть. Перечень типов и назначение объектов данных области памяти контроллера СРМ902, отображаемых на множества регистров и битовых полей сервера MODBUS приведен в табл. 1.

Таблица 1.

Тип секции	Тип объекта	Область данных объекта
Outputs	<b>WORD Output</b>	Двухбайтовый выходной канал типа WORD
	<b>DWORD Output</b>	Четырехбайтовый выходной канал типа DWORD
	<b>REAL Output</b>	Четырехбайтовый выходной канал типа REAL
	<b>LREAL Output</b>	Восьмибайтовый выходной канал типа LREAL
	<b>2-Bytes Output</b>	2 однобайтовых выходных канала типа BYTE
Inputs	<b>WORD Input</b>	Двухбайтовый входной канал типа WORD
	<b>DWORD Input</b>	Четырехбайтовый входной канал типа DWORD
	<b>REAL Input</b>	Четырехбайтовый выходной канал типа REAL
	<b>LREAL Input</b>	Восьмибайтовый входной канал типа LREAL
	<b>2-Bytes Input</b>	2 однобайтовых входных канала типа BYTE

В соответствии с задачей лабораторной работы №2 команды на включение и отключение 4-х релейных выходов будут передаваться на контроллер в виде данных по протоколу Modbus от мастера сети. Для формирования списка объектов доступа к этим данным используется секция **Inputs**.

2.5.1. Перейти в секцию **Inputs**, щелкнуть правой кнопкой мыши и в контекстном меню выбрать команду «**Добавить Подэлемент**». Затем в списке выбрать тип объекта **2 – Bytes Input**, который позволяет принимать значения отдельных битовых полей.

2.5.2. Раскрыть иерархический список добавленного объекта **2 – Bytes Input**, который представляет собой описание 2-х отдельных байтов (рис. 27).

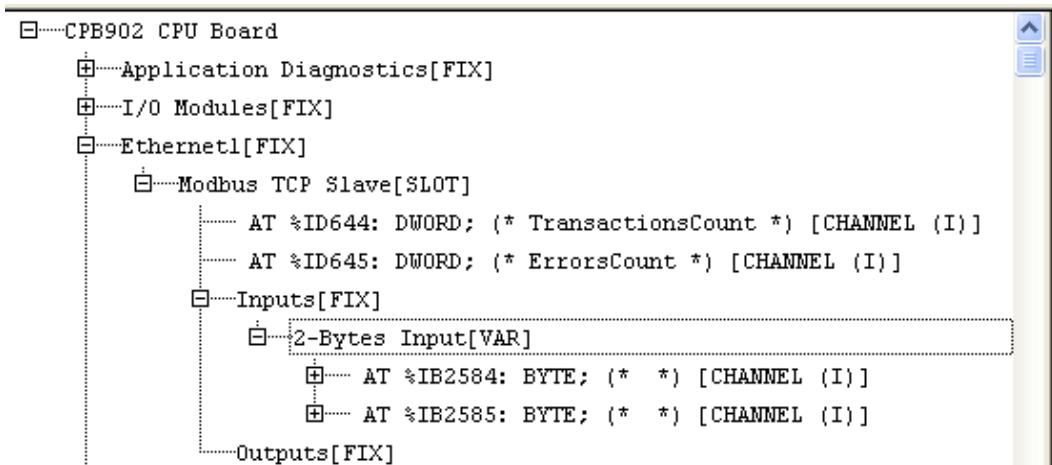


Рис.27. Список объекта 2-Bytes Input

2.5.3. Перейти к первому байту "АТ %IB2584". Предполагая, что этот канал будет использоваться только для управления 4-мя релейными выходами (каналами модулей DIM713), в поле комментария (в окне справа) ввести название канала **N22\_N23\_relays\_management**. (рис. 28).

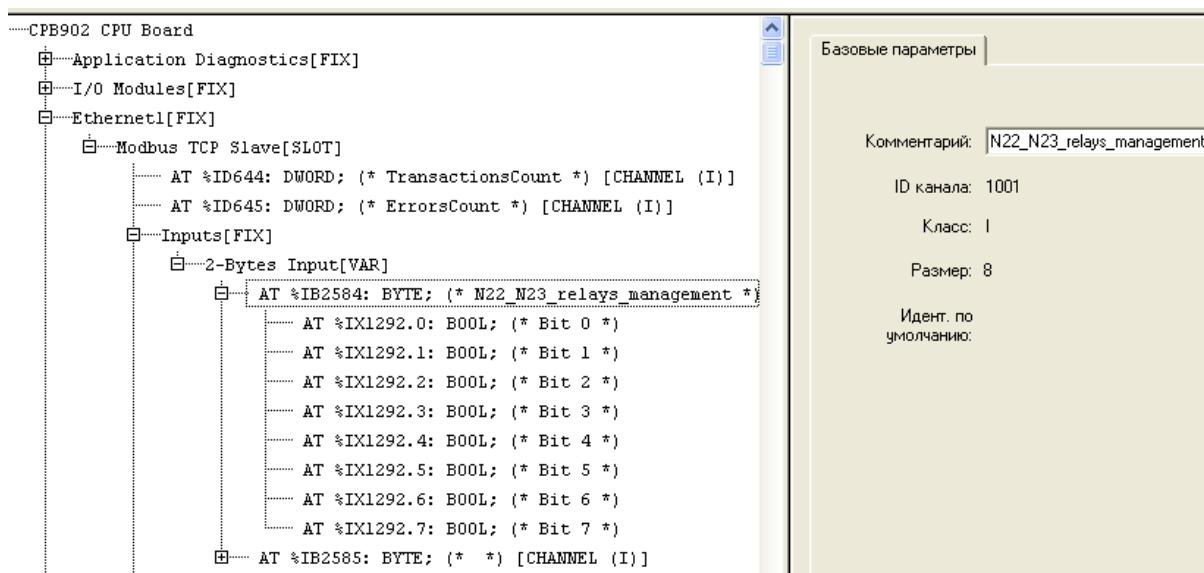


Рис 28. Редактирование поля комментария для канала первого байта объекта 2–Bytes Input

2.5.4. Внутри байта можно получить доступ к отдельным битовым полям. Раскрыть список битовых полей канала **N22\_N23\_relays\_management** (рис. 28). Первые 4 бита данного элемента и будут использоваться для управления 4-мя релейными выходами модулей DIM713. Выбрать бит, первый по списку: **AT %IX1292.0 : BOOL (\*Bit 0\*)**. Дважды щелкнуть левой кнопкой мыши слева от надписи "AT%..." и ввести символьное имя создаваемой переменной: **mb\_relay1**. Нажать клавишу «Enter».

2.5.5. Последовательно переходя на следующие три бита, создать символьные имена:

**mb\_relay2;**

**mb\_relay3;**

**mb\_relay4** (рис. 29).

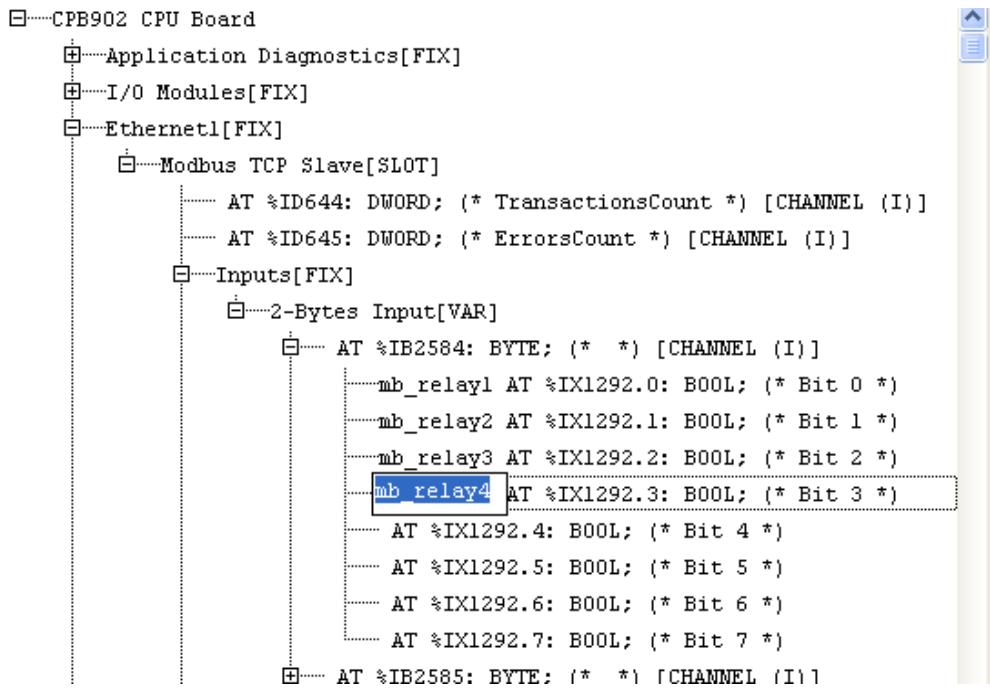


Рис. 29. Созданные символьные имена для отдельных битовых полей объекта 2–Bytes Input (переменные mb\_relay1...4).

Созданные в п. 2.5.4 – 2.5.5 переменные являются **глобальными** для всего проекта.

**2.5.6.** Теперь можно получить отображение используемого объекта **2 – Bytes Input** на множество регистров и битовых полей сервера Modbus.

- 1) Выбрать корневой элемент объекта – секцию **Inputs** и, нажав правую клавишу, выбрать команду **Вычислять адреса**.
- 2) Опять вернуться на элемент объекта **2 – Bytes Input** и в диалоговом окне справа получить параметры доступа **Modbus Access Properties** (рис. 30).

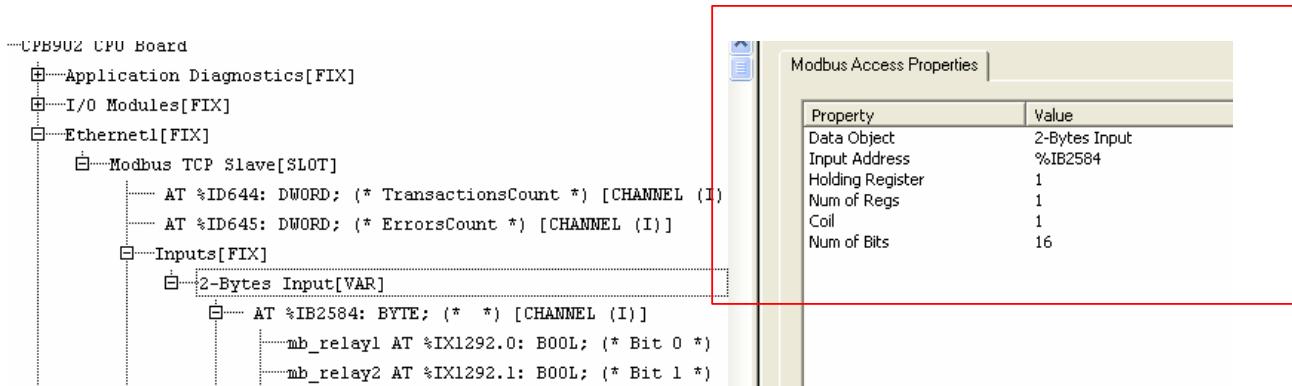


Рис. 30. Отображение используемого объекта 2-Bytes Input на множество регистров и битовых полей сервера Modbus

Из описания этих параметров следует, что мастер сети Modbus может выполнять запросы записи и чтения, используя адрес **1** в операциях доступа к выходным регистрам (***Holding Registers***).

Шестнадцать битовых полей (***Num of Bits***) данного регистра могут адресоваться мастером сети Modbus следующим образом:

адрес первого битового поля: 1;

-----  
адрес шестнадцатого битового поля: 16.

(В общем виде адрес битового поля вычисляется по формуле:

***Адрес битового поля = (Адрес регистра – 1)\*16 + Номер битового поля (начиная с 1).***

Нас интересуют только первые четыре бита выходного регистра с адресами от **1** до **4**, соответствующие каналам переменных **mb\_relay1 ... 4**. Соответственно со стороны клиента Modbus нужно будет формировать запросы удаленному контролеру на запись значений в битовые поля делимого регистра (тип COIL) с адресами от **1** до **4**.

## 2.6 РЕДАКТИРОВАНИЕ ГЛАВНОЙ ПРОГРАММЫ PLC\_PRG POU

2.6.1. Перейти на вкладку **POU**, расположенной в нижней области главного окна среды разработки CoDeSys и выбрать программу **PLC\_PRG (POU)**.

2.6.2. Перейти в поле ввода кода. Пользуясь ассистентом ввода (**F2**) изменить программный код следующим образом:

```
N22_relay1 := mb_relay1; (* DIM713 (N22) - реле 1 управ-ся переменной mb_relay1 /Modbus COIL: 1*)
N22_relay2 := mb_relay2;
N23_relay1 := mb_relay3; (* DIM713 (N23) - реле 1 управ-ся переменной mb_relay3 /Modbus COIL: 3*)
N23_relay2 := mb_relay4;
```

2.6.3. Сохранить проект.

2.6.4. Компиляция проекта – выполнить команду меню **Проект – Компилировать (F11)**. Результат процесса компиляции будет указан в самом нижнем окне редактора Codesys. Если будут обнаружены ошибки в программе, то они будут указаны в этом же окне.

## 2.7 ОТЛАДКА ПРОГРАММЫ В РЕЖИМЕ ЭМУЛЯЦИИ

Отладка программы в режиме эмуляции происходит по тем же принципам, что были описаны в Лабораторной работе №1, в разделе 1.7.

Просматривать и менять значения переменных для каналов ввода-вывода и сервера Modbus можно и в поле ввода кода (вкладка **POU**), и в ресурсе **Конфигурация ПЛК** (вкладка **Ресурсы**).

## 2.8 ЗАГРУЗКА ПРИЛОЖЕНИЯ В КОНТРОЛЛЕР

Способы создания логических информационных каналов для подключения и загрузки приложения в контроллер были подробно описаны в Лабораторной работе №1, раздел 1.8 и 1.9.

## 2.9 ОНЛАЙН ОТЛАДКА ПРОГРАММЫ, ПРОСМОТР И УСТАНОВКА ЗНАЧЕНИЙ ПЕРЕМЕННЫХ

Предполагается, что прикладная программа загружена в контроллер, а в среде CoDeSys открыт проект, в котором разрабатывалась данная программа. Для просмотра значений переменных подключить среду разработки CoDeSys к контроллеру СМР902, выполнив команду меню **Онлайн – Подключение (Alt + F8)**.

## 2.10 МОНИТОРИНГ И ОТЛАДКА РАБОТЫ СЕРВЕРА MODBUS

В разделах **2.3 – 2.9** было описано конфигурирование и запуск контроллера СМР902 в качестве подчиненного узла (сервера) сети Modbus. Для мониторинга и отладки работы этого сервиса будем использовать прикладное ПО **Fastwel Modbus OPC Server**, установленное на АРМ студентов и преподавателя.

Fastwel Modbus OPC Server реализует функции мастера протоколов Modbus и Modbus TCP, выполняя операции чтения и записи данных между компьютером, на котором он установлен, и, подчиненными узлами сети. Сервер поддерживает объекты прикладного уровня протокола Modbus:

- Input Register (входной регистр);
- Holding Register (выходной регистр);
- Discrete Input (дискретный вход);

- Coil (дискретный выход).

2.10.1. Средствами операционной системы Windows 7 загрузить на исполнение сервер: **Пуск- Все программы – Fastwel Modbus OPC Server - Modbus OPC Server** (рис. 31).

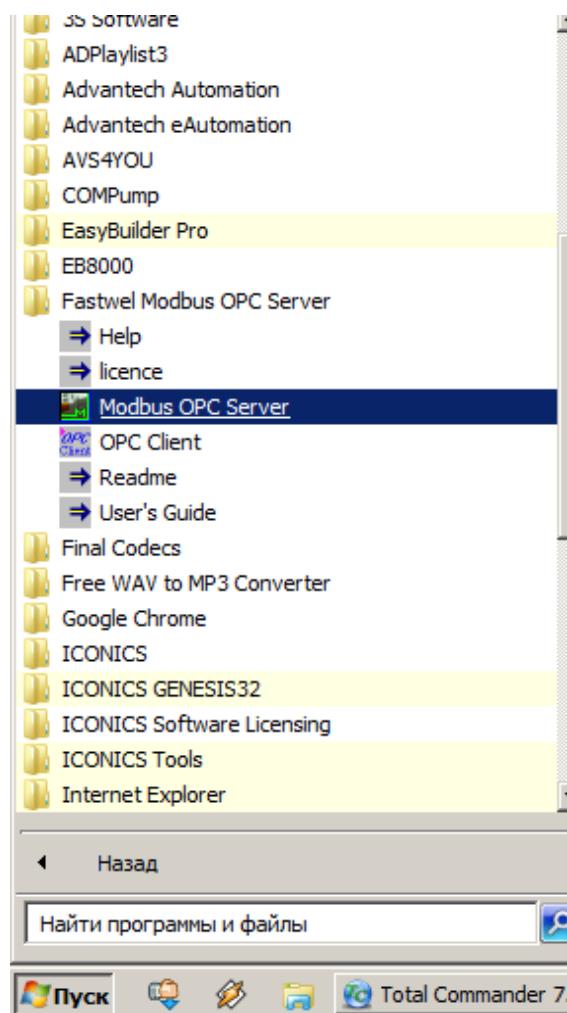


Рис. 31. Запуск ПО Fastwel Modbus OPC Server

2.10.2. В открывшемся окне программы создать конфигурацию сервера. Для этого в главном меню сервера выбрать пункт **Добавить – «Новое устройство.»** (или нажать соответствующую кнопку на панели инструментов, рис 32).

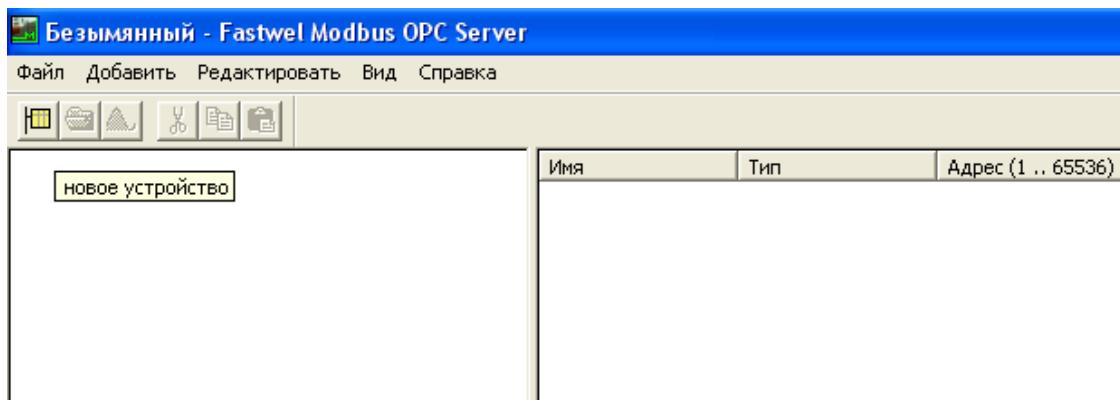


Рис. 32. Создание конфигурации устройства Modbus

2.10.3. При этом откроется окно программной настройки параметров устройства, как показано на рис.33. Оно подразумевает конфигурирование программного объекта, представляющего в сервере реальное физическое устройство. Выбрать тип устройства ModbusTCP и заполнить его параметры:

- **Имя** – указать имя создаваемого элемента в конфигурации OPC сервера. По умолчанию – это **Node1**, но можно изменить на имя, которое четко ассоциируется с удаленным устройством (например, оно может быть «**Fastwel\_CPM902**»). При изменении рекомендуется использовать только сочетания латинских букв и цифр.
- **IP-адрес** удаленного подчиненного узла Modbus:  
**192.168.1.2** - контроллер шкафа ШК1;  
**192.168.1.4** - контроллер шкафа ШК2.
- **Адрес** – индивидуальный адрес устройства в сети Modbus. Оставить значение по умолчанию – **1**. Для Modbus TCP, как правило, используется только это значение.
- **Таймаут** – время, отведенное на ответ устройства. (по умолчанию – **500 мс**).

Нажать **OK**, после чего в конфигурацию будет добавлено новое устройство.

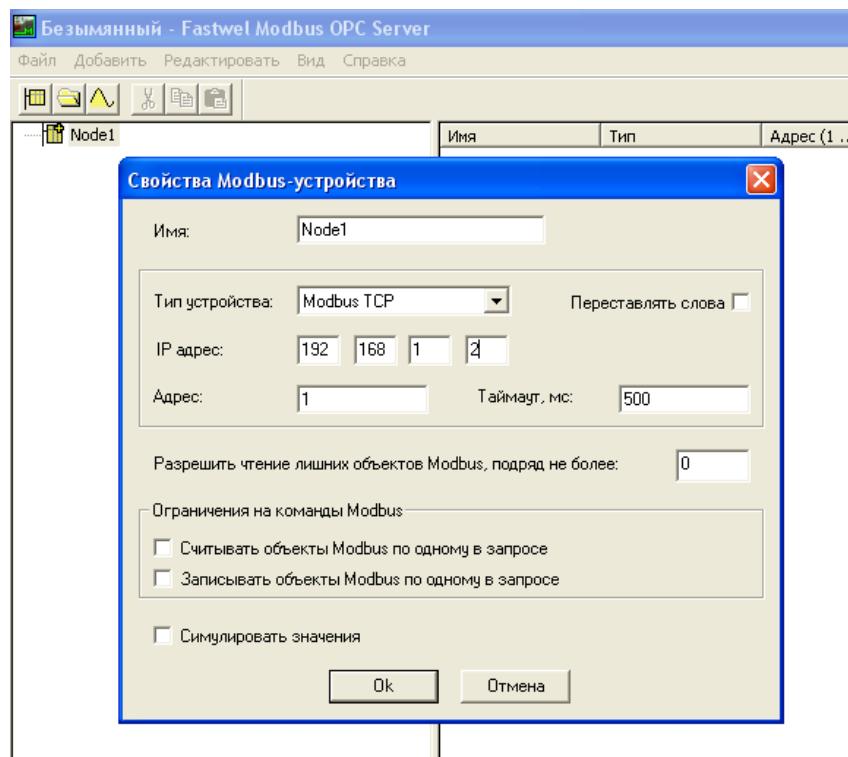


Рис. 33. Редактирование конфигурации удаленного устройства Modbus

**2.10.4. Создание тегов.** Доступные по сети Modbus объекты – регистры, дискретные входы и выходы представляются для OPC-клиентов в виде **тегов**.

Чтобы создать тег, следует выделить созданную конфигурацию устройства и, щелкнув правой кнопкой мыши, выбрать команду «**Новый тег**» или нажать на соответствующую иконку на панели инструментов. Появится диалог свойств тега (рис. 34).

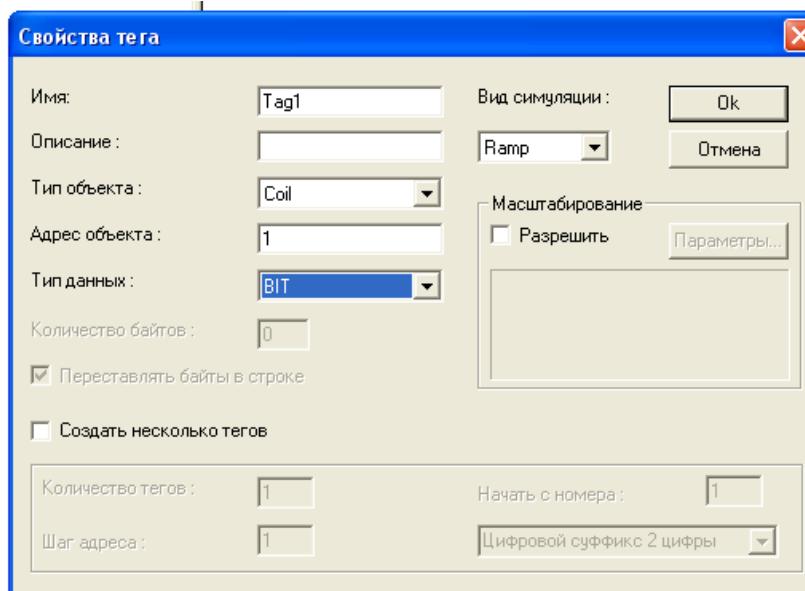


Рис. 34. Диалог свойств создаваемого тега

#### 2.10.5. Заполнить поля свойств тега (рис. 13):

- **Имя** – имя тега. Как и в имени устройства не рекомендуется применять символы, отличные от латинских букв, цифр и подчеркивания. Переименовать имя первого тега на *Relay1*.
- **Описание** – поле комментария, является необязательным.
- **Тип объекта**. Из списка выбрать тип **COIL** (дискретный выход). Как было определено в п.5.6, для управления первым релейным выходом со стороны Modbus нужно формировать запрос на изменение состояния отдельных разрядов делимого регистра – COIL с адресом **1**. Поэтому следует установить адрес объекта **1**.
- **Тип данных**, передаваемых в запросе Modbus. При использовании типа объекта COIL из списка следует выбрать тип данных **BIT**.

Нажать **OK**, после чего в конфигурацию будет добавлен новый тег.

2.10.6. Повторяя последовательность действий п.п. 2.10.4. и 2.10.5, создать еще три подобных тега (рис. 14):

**Relay2 – COIL** – адрес **2** – тип данных **BIT**;

**Relay3 – COIL** – адрес **3** – тип данных **BIT**;

**Relay4 – COIL** – адрес **4** – тип данных **BIT**.



Рис. 14. Набор созданных тегов в конфигурации сервера

2.10.7. Сохранить созданную конфигурацию через пункт меню **Файл-Экспорт в csv**. При необходимости можно будет открыть её командой **Файл – Импорт из csv**.

2.10.8. Выполнить команду **Вид->Монитор** в главном меню. Значения тегов в реальном времени будут отображаться на правой панели главного окна. Теги типа **BIT** могут принимать значения **off** (логический “0”) или **on** (логическая “1”).

Если значения всех тегов отображаются как **Bad** (рис. 15), то вероятнее всего, что удаленный сервер Modbus не доступен. Следует проверить доступность удаленного контроллера по IP-адресу, используя утилиту **ping**.

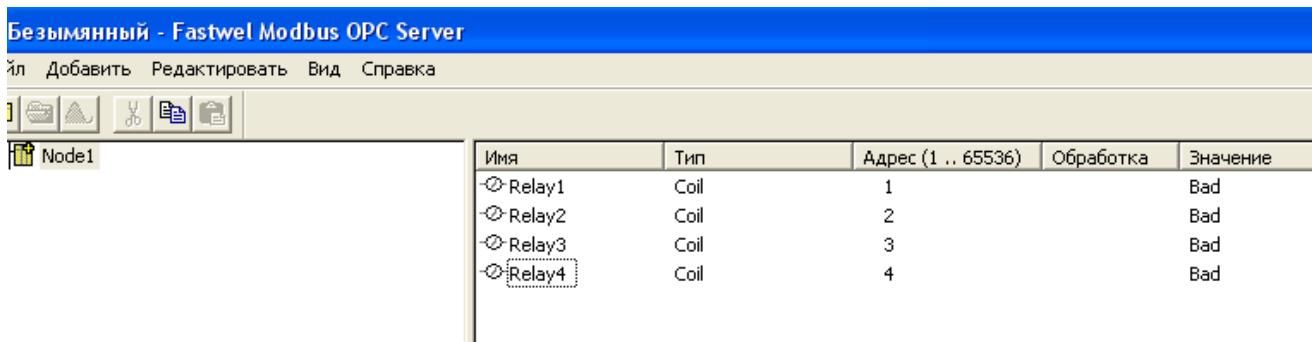


Рис.15. ПО Fastwel Modbus OPC Server не удаётся подключиться к удаленному серверу Modbus (значения всех тегов «Bad»)

Возможные причины отсутствия ответа на эхо – запросы по IP-адресу и действия по их устранению:

- контроллер CPM902 выключен – проверить питание оборудования;
- контроллер CPM902 включен, но его сетевой сервис неверно сконфигурирован – проверить конфигурацию ПЛК;
- контроллер CPM902 включен, но находится в режиме работы, отличном от нормального – проверить световую индикацию контроллера (LED1, LED2), действовать в соответствии с ней;
- в сети Ethernet обрыв – проверить исправность сетевого кабеля (патч-корда) и работоспособность сетевых коммутаторов.

2.10.9. При успешном подключении ПО Fastwel Modbus OPC Server к удаленному серверу Modbus (рис. 16) значения тегов типа BIT отображаются как **off** (логический “0”)/по умолчанию/ или как **on** (логическая “1”).

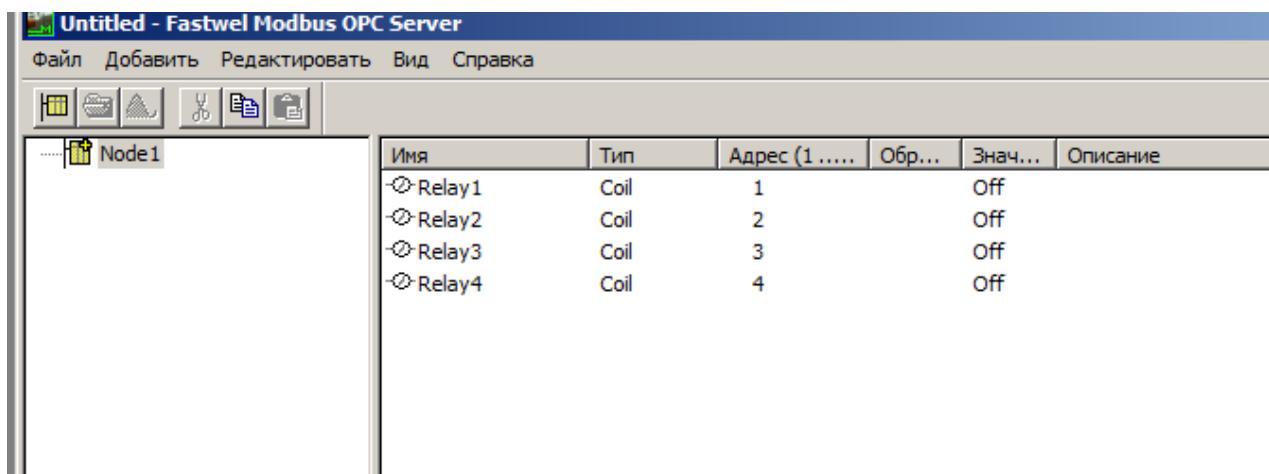


Рис.16. ПО Fastwel Modbus OPC Server успешно взаимодействует с удаленным сервером Modbus

2.10.10. Для управлением релейными выходами к ПО Fastwel Modbus OPC Server следует подключить OPC клиента – прикладную программу на ПК, имеющую интерфейс OPC. В данной лабораторной работе для отладки будет использоваться ПО **OPC client**, входящий в комплект установки ПО Fastwel Modbus OPC Server. Взаимодействие этих программных компонентов отображено на рис. 2 (раздел 2).

2.10.11. Средствами операционной системы Windows 7 загрузить клиента OPC: **Пуск- Программы – Fastwel Modbus OPC Server - OPC Client.**

2.10.12. В открывшемся окне программы **Fastwel OPC Client** через меню выполнить команду **OPC-Connect** и выбрать сервер OPC с именем **Fastwel.Modbusopc** (рис. 17). Нажать **OK**.

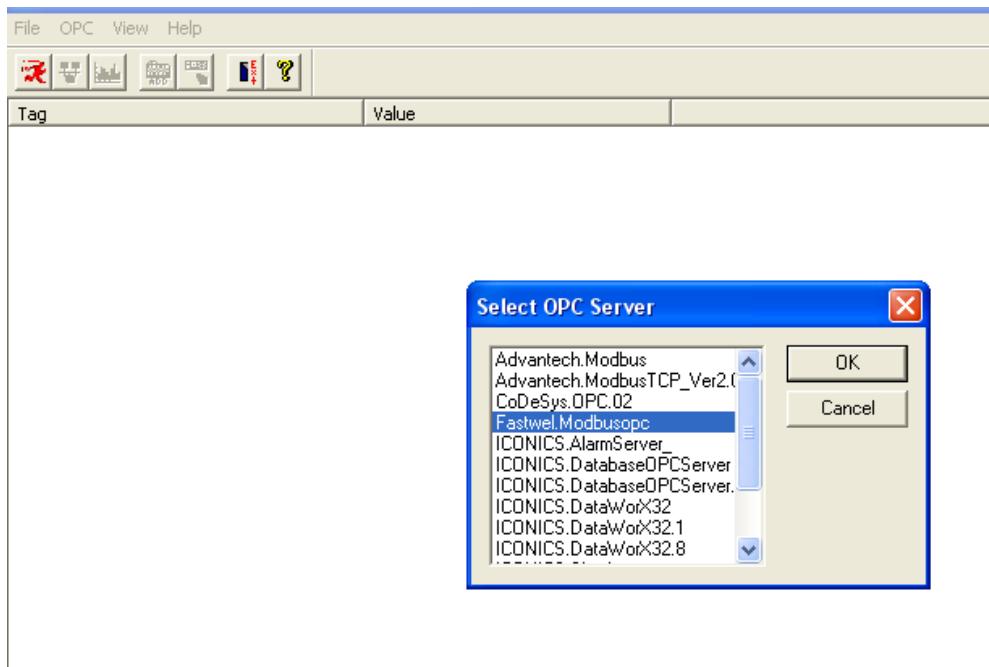


Рис.17. Модуль OPC Client – окно выбора сервера OPC

2.10.13. На панели инструментов нажать кнопку **Add Item**. В появившемся окне отобразится текущая конфигурация Fastwel Modbus OPC Server.

Выбрать и нажатием кнопки **OK** добавить в конфигурацию OPC клиента тег **Relay1**. Затем таким же образом добавить теги **Relay2..4** (рис. 18).

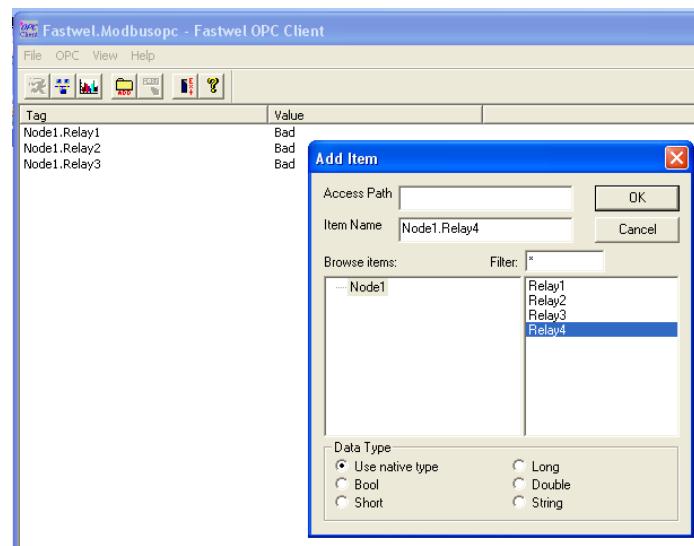


Рис.18. Добавление в конфигурацию OPC Client набора тегов для мониторинга и записи значений

2.10.14. Выделить тег ***Relay1*** в созданной конфигурации OPC Client. Нажать кнопку **Write Item** на панели инструментов (рис. 19). В окне записи с клавиатуры установить 0 или 1, нажать **OK**.

Это будет соответствовать команде записи этого значения в соответствующий тег конфигурации Fastwel Modbus OPC Server, который в свою очередь отправит удаленному устройству Modbus (контроллеру СРМ902) запрос на запись значения в битовое поле делимого регистра с адресом «1». После того, как сервер Modbus примет изменение, в соответствии с алгоритмом программы произойдет выключение/ включение выходного реле 1 модуля DIM713 с порядковым номером N22. Проверить изменение состояния этого релейного выхода на соответствующем контроллере СРМ902 (в шкафу ШК1 или ШК2).

2.10.15. Проверить функционирование всех 4-х тегов и соответствующих им каналов:

Тег ***Relay1*** – реле 1 DIM713 (N22);

Тег ***Relay2*** – реле 2 DIM713 (N22);

Тег ***Relay3*** – реле 1 DIM713 (N23);

Тег ***Relay4*** – реле 2 DIM713 (N23).

**РАЗДЕЛ 3. ЛАБОРАТОРНАЯ РАБОТА №3**

**«ОСНОВНЫЕ ПРИНЦИПЫ РАЗРАБОТКИ ЭКРАННЫХ ФОРМ  
ГРАФИЧЕСКОЙ ПАНЕЛИ WT3010. ОРГАНИЗАЦИЯ  
ВЗАИМОДЕЙСТВИЯ ПАНЕЛИ WT3010 И КОНТРОЛЛЕРА СРМ902 ПО  
ПРОТОКОЛУ MODBUS»**

**3.1 ОПИСАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ №3**

**Назначение:**

Изучение среды Weintek EasyBuilder 8000 для разработки приложений графической панели WT3010 и принципов сетевого взаимодействия по протоколу Modbus.

**Цель работы освоение практических навыков:**

- создания экранных форм графической панели WT3010;
- конфигурирования панели в качестве мастера (клиента) Modbus для сетевого обмена данными с другими устройствами комплекса.

**Используемое оборудование:**

- графическая панель оператора WT3010, расположенная в ШК1 / ШК2 в соответствии со структурной схемой комплекса (Приложение А);
- программируемый контроллер Fastwel IO СРМ902 с набором функциональных модулей. Модули соединены в единую конструкцию через пазы на корпусе каждого модуля в порядке, определенном в таблице 2 (Приложение В);
- АРМ студента или преподавателя.

**Используемое инструментальное ПО (установлено на АРМ студентов и преподавателя):**

- среда разработки Weintek EasyBuilder 8000;

- среда разработки Fastwel IO Codesys Adaptation.

## **Обязательный изученный теоретический материал для допуска к лабораторной работе:**

- описание комплекса технических средств;
- описание комплекса программного обеспечения.

### **3.2 ОБЩИЕ СВЕДЕНИЯ**

В лабораторной работе №3 приобретаются практические навыки создания простых экранных форм графической панели WT3010, а также конфигурирования панели в качестве мастера (клиента) Modbus.

В результате обучающийся должен реализовать процесс управления с панели WT3010 релейными выходами контроллера CPM902 (4-мя каналами модулей DIM713) по протоколу Modbus. Соответствующая этим задачам прикладная программа для CPM902, функционирующего в качестве сервера Modbus, была подготовлена в ходе выполнения Лабораторной работы №2. Взаимодействие панели WT3010 и контроллера CPM902 по протоколу Modbus отображено на рис.1.

В ходе выполнения лабораторной работы №3 следует строго соблюдать, чтобы панель WT3010, размещенная в шкафу ШК1, управляла работой контроллера CPM902, размещенного в том же шкафу. Тоже самое касается и оборудования шкафа ШК2.

Для этого при редактировании системных настроек панелей WT3010 следует учитывать распределение IP-адресов в соответствии со структурной схемой комплекса.

ШК1:

**192.168.1.2** – Контроллер CPM902;

**192.168.1.3** – Графическая панель WT3010.

ШК2:

**192.168.1.4** – Контроллер CPM902;

**192.168.1.5** – Графическая панель WT3010.

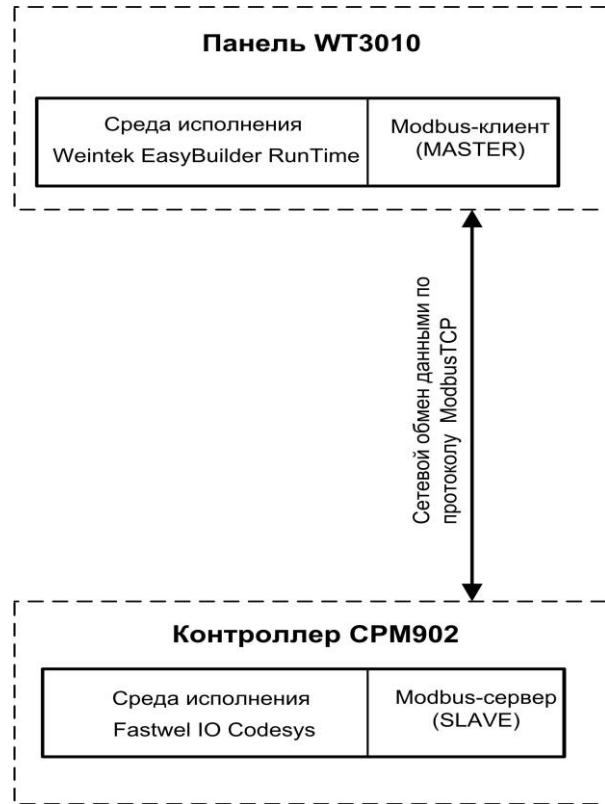


Рис. 1 Взаимодействие панели WT3010 и контроллера  
CPM902 протоколу Modbus

Последовательность операций при запуске графической панели WT3010 в качестве мастера (клиента) сети Modbus для управления работой контроллера CPM902:

- создание проекта в редакторе EasyBuilder8000;
- редактирование системных настроек проекта;
- редактирование экранной формы;
- отладка программы в режиме оффлайн-симуляции;
- отладка программы в режиме онлайн-симуляции;
- редактирование сетевых параметров панели WT3010, загрузка в нее созданного проекта;

- проверка управления графической панелью работой контроллера CPM902.

### 3.3 СОЗДАНИЕ ПРОЕКТА

3.3.1. Средствами операционной системы Windows 7 загрузить на исполнение редактор проектов EasyBuilder8000 (рис.2): **Пуск – Все программы – EB8000 – EasyBuilder8000.**

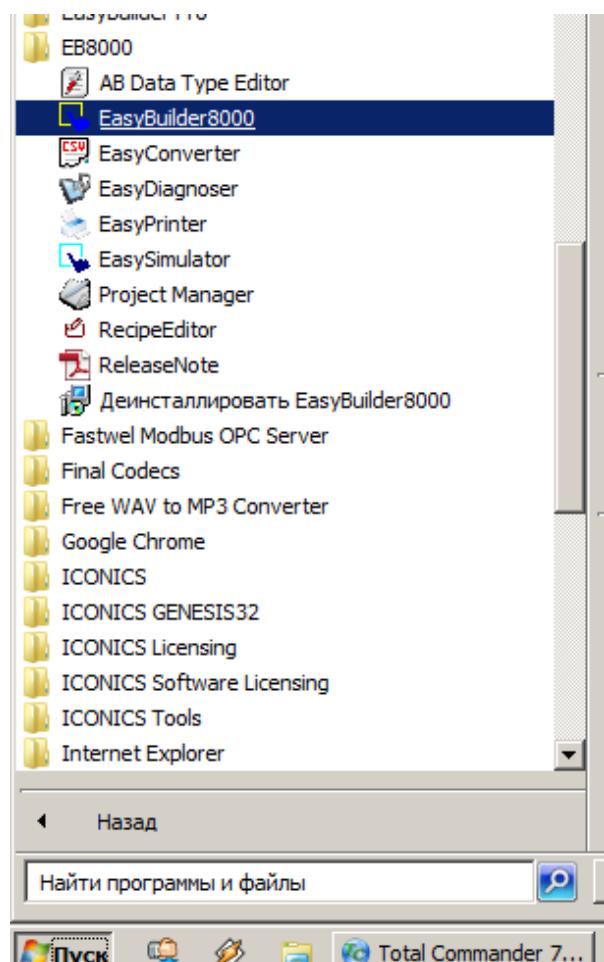


Рис. 2. Загрузка редактора проектов **EasyBuilder8000**

3.3.2. В появившейся диалоговой панели (рис. 3) выбрать пункт «Создать новый проект». Нажать кнопку **OK**.

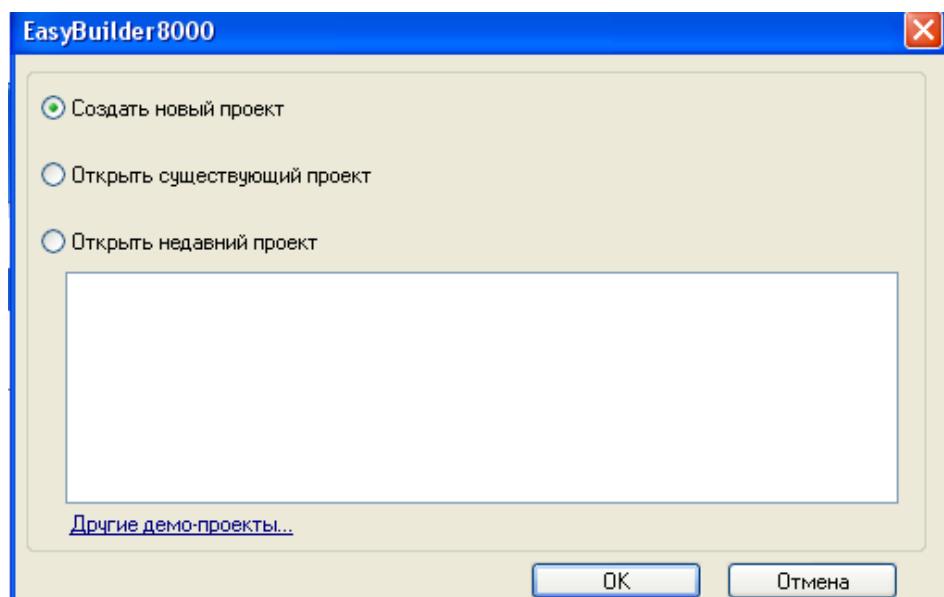


Рис. 3. Создание нового проекта

3.3.3. В появившемся диалоговом окне раскрыть спускающийся список моделей графических панелей Weintek. Выбрать строку, в которой присутствует панель WT3010, как показано на рис.4.

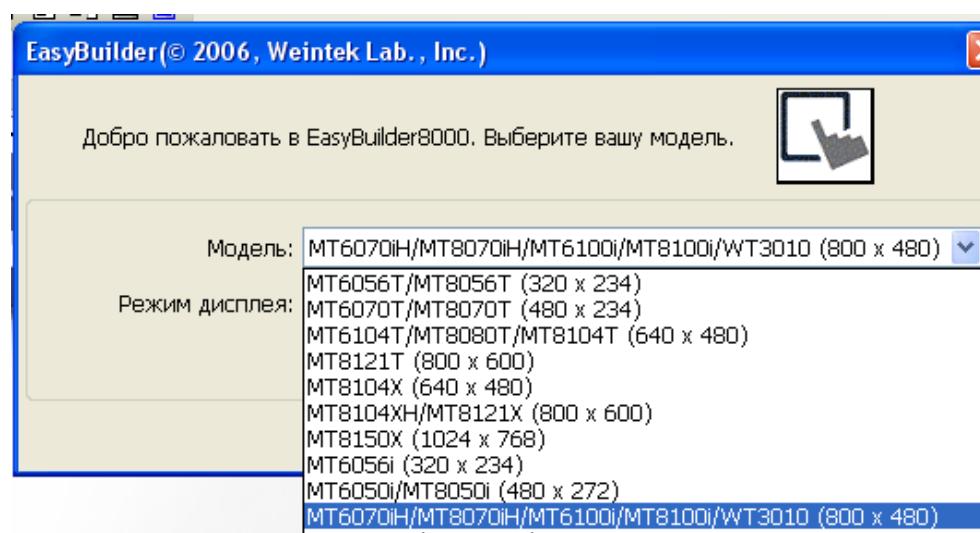


Рис. 4. Выбор модели графической панели

3.3.4. В том же окне выбрать режим дисплея – «Ландшафтный» (горизонтальная ориентация экрана), как показано на рис. 5. Нажать кнопку **OK**.

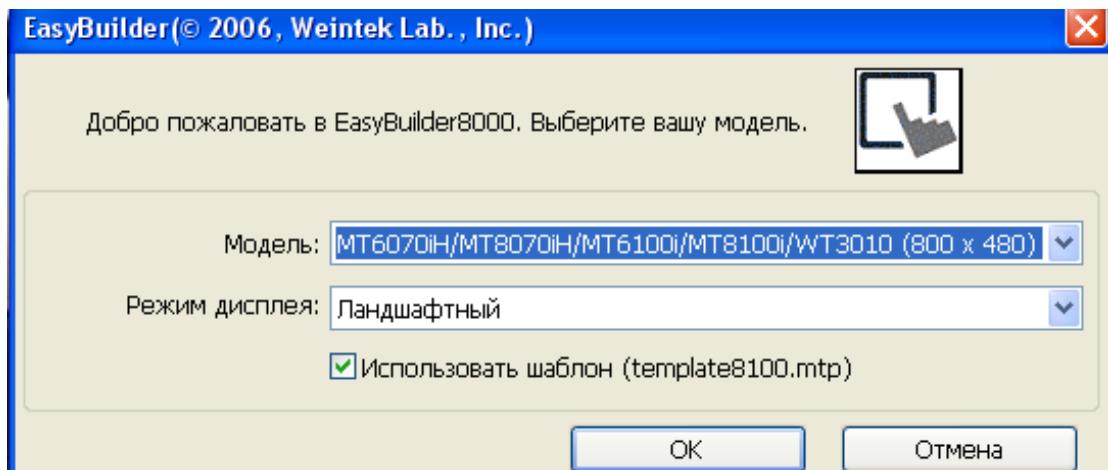


Рис. 5. Выбор режима дисплея

3.3.5. Появится окно редактирования системных настроек (рис.6). Нажать кнопку **ОТМЕНА**, поскольку изменение настроек будет проводиться позже, в разделе 4.

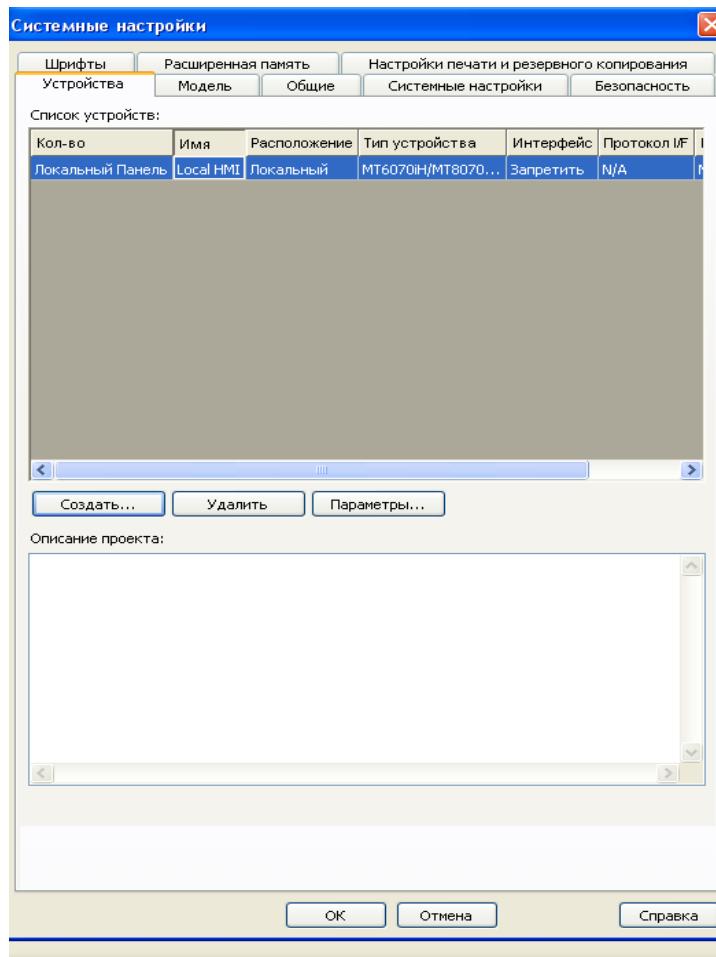


Рис. 6. Окно редактирования системных настроек

3.3.6. Сохранить созданный проект под именем **Lab\_4.pro** на локальном диске в папке **С:\Лаборатория гидродинамики \Labs\**. При необходимости можно в название файла внести фамилию студента и номер учебной группы, например **Lab\_4\_Ivanov\_5101.pro**.

### 3.4 РЕДАКТИРОВАНИЕ СИСТЕМНЫХ НАСТРОЕК ПРОЕКТА

3.4.1. Выбрать в меню редактора EasyBuilder8000 **Правка – Системные настройки**. Появится окно редактирования системных настроек, как было отображено на рис.6. Вкладка «Устройства» отображает список устройств, для подключения к которым созданы соответствующие конфигурации. В исходном виде в этом списке присутствует только одно устройство с именем **Local HMI**, через которое предоставляется доступ к ячейкам внутренней памяти панели.

3.4.2. Для редактирования параметров подключения к удаленному контроллеру СРМ902 по протоколу Modbus следует нажать кнопку **Создать**, как показано на рис. 7.

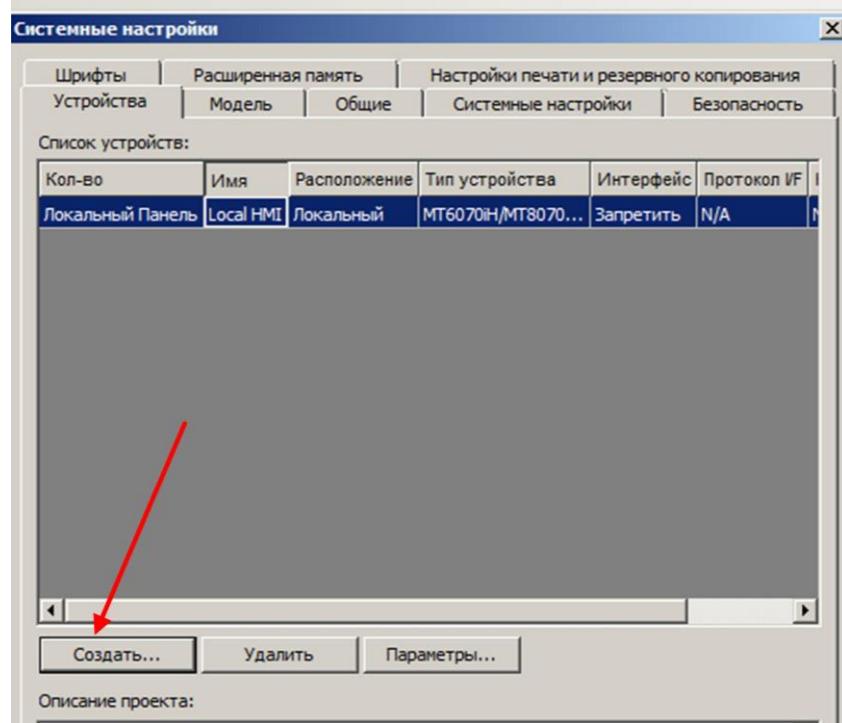


Рис. 7. Создание конфигурации устройства, к которому будет осуществляться подключение

3.4.3. В появившемся окне «Параметры устройства» (рис. 8) изменить параметры нового устройства.

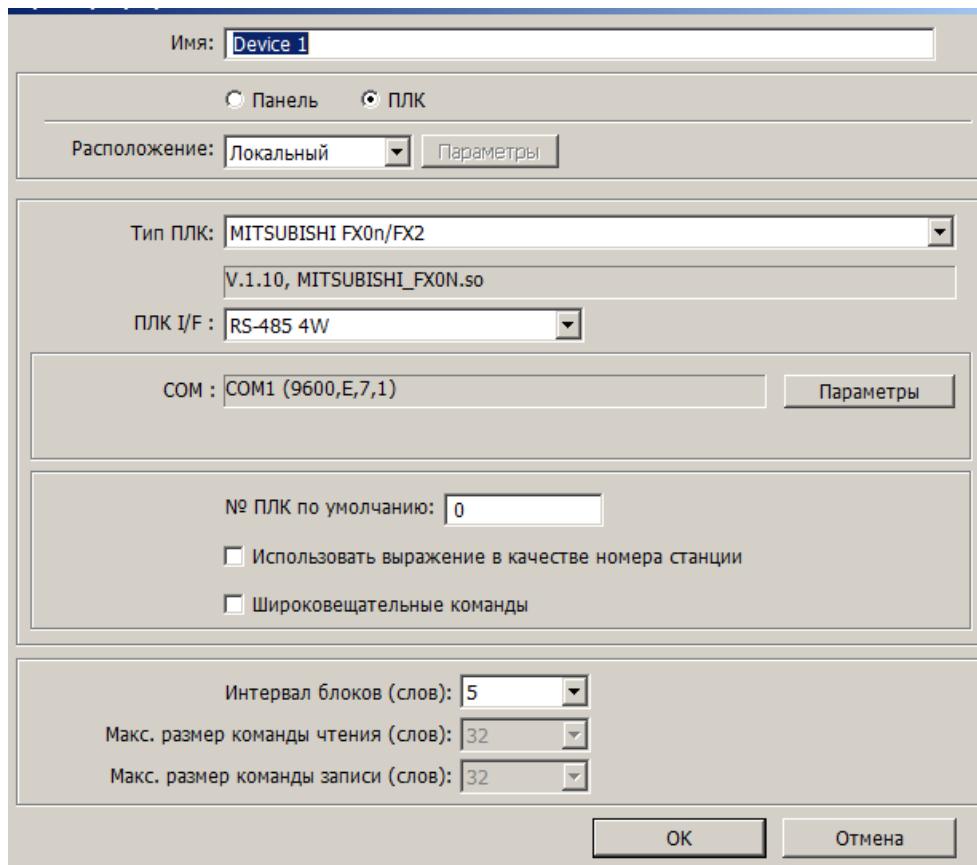


Рис. 8. Редактирование параметров нового устройства

- **Имя** – имя устройства (по умолчанию «Device 1»). Можно поменять имя на любое другое, например «Fastwel\_CPM902». При этом не рекомендуется применять символы, отличные от латинских букв, цифр и подчеркивания.
- **Расположение** – «Локальный». Оставить без изменения, поскольку подключение к контроллеру CPM902 будет осуществляться напрямую с панели. Альтернативное значение «Удаленный» выбирается, если предполагается работа через другую удаленную панель.
- **Тип ПЛК**. Выбор протокола обмена или типа устройства. Панели Weintek поддерживают большое количество коммуникационных протоколов и могут работать с разным оборудованием [3]. Раскрыть спускающийся список и выбрать строку **Modbus TCP/IP (Ethernet)**, как показано на рис. 9.

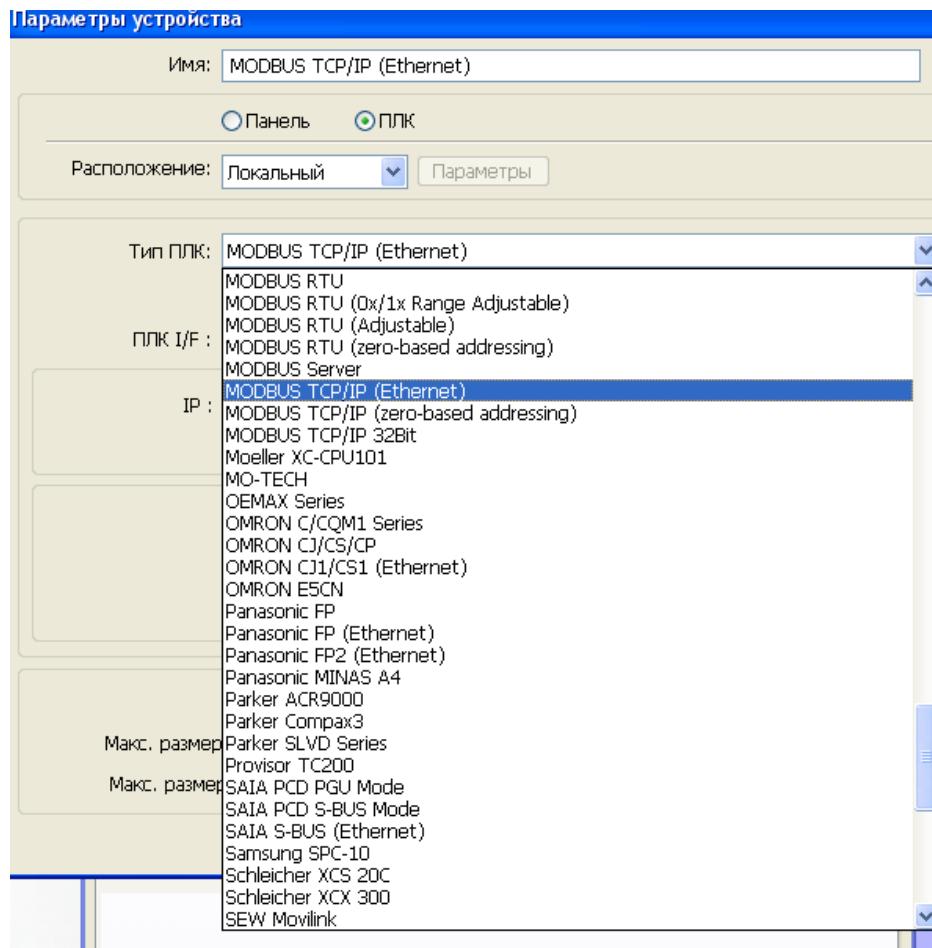


Рис. 9. Выбор коммуникационного протокола Modbus TCP /IP

- **ПЛК I/F.** Выбор интерфейса связи. Протокол Modbus TCP предполагает реализацию через порт Ethernet, поэтому выбирается интерфейс «Сеть».
- **Параметры** – сетевые параметры устройства, к которому будет осуществляться подключение. Следует установить IP-адрес контроллера CPM902, с которым предстоит работать (рис. 10):
  - **192.168.1.2** – контроллер CPM902 ШК1;
  - **192.168.1.4** – контроллер CPM902 ШК2.

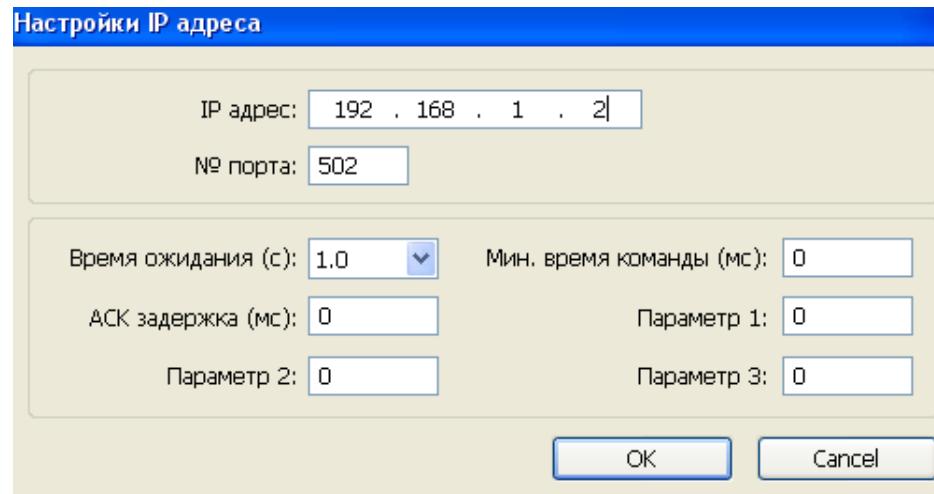


Рис. 10. Редактирование сетевых параметров для связи с контроллером СPM902

- **№ ПЛК по умолчанию** - индивидуальный адрес устройства в сети Modbus. Оставить значение по умолчанию – 1. Для Modbus TCP, как правило, используется только это значение.

Нажать **OK**, после чего будет добавлено новое устройство – конфигурация подключения к контроллеру СPM902, как показано на рис. 11. Выйти из окна системных настроек.

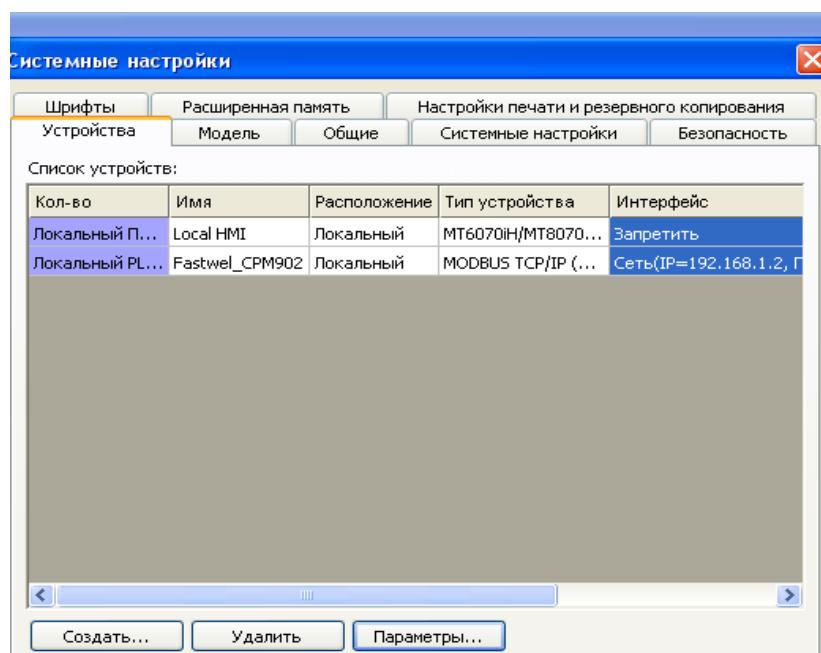


Рис. 11. Обновленный список устройств, содержащий конфигурацию подключения к контроллеру СPM902

### 3.5 РЕДАКТИРОВАНИЕ ОКНА ЭКРАННОЙ ФОРМЫ

Окно – базовый компонент проекта. С помощью окон пользователь может увидеть на экране панели все виды информации: в виде объектов, изображений или текста. До 1997 окна, пронумерованные от 3 до 1999, могли быть использованы в редакторе EasyBuilder 8000.

По умолчанию пользователю предлагается редактировать основное окно проекта: **№0010 с именем WINDOW\_010** (рис. 12). Именно оно и будет отображаться в первую очередь при запуске панели.

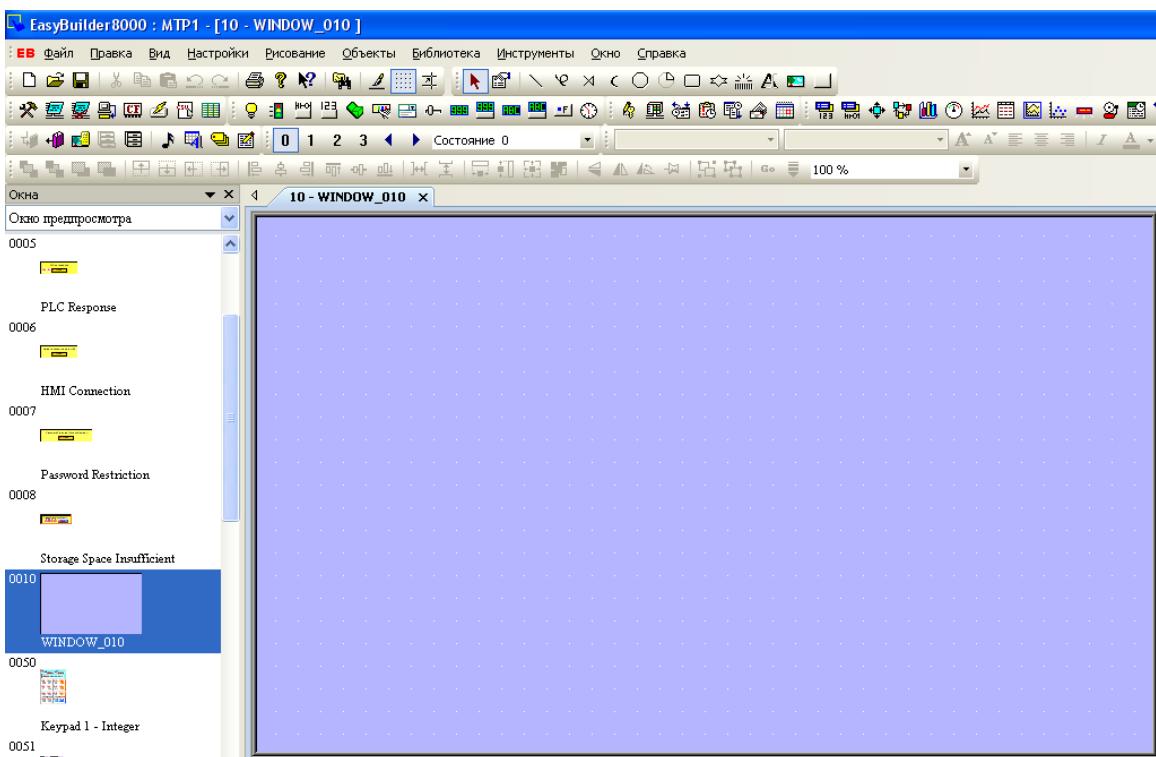


Рис. 12. Основное окно проекта: №0010 с именем WINDOW\_010

Для управления 4-мя релейными выходами контроллера СРМ902 с экрана графической панели будем использовать 4 элемента в виде тумблеров, которые разместим в окне WINDOW\_010.

3.5.1. На панели инструментов выбрать пиктограмму элемента Битовый переключатель, как показано на рис. 13. Откроется окно редактирования параметров этого элемента (рис. 14).

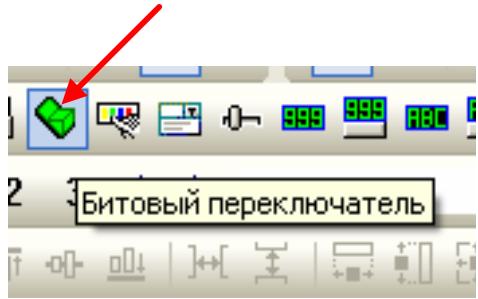


Рис. 13. Пиктограмма элемента Битовый переключатель

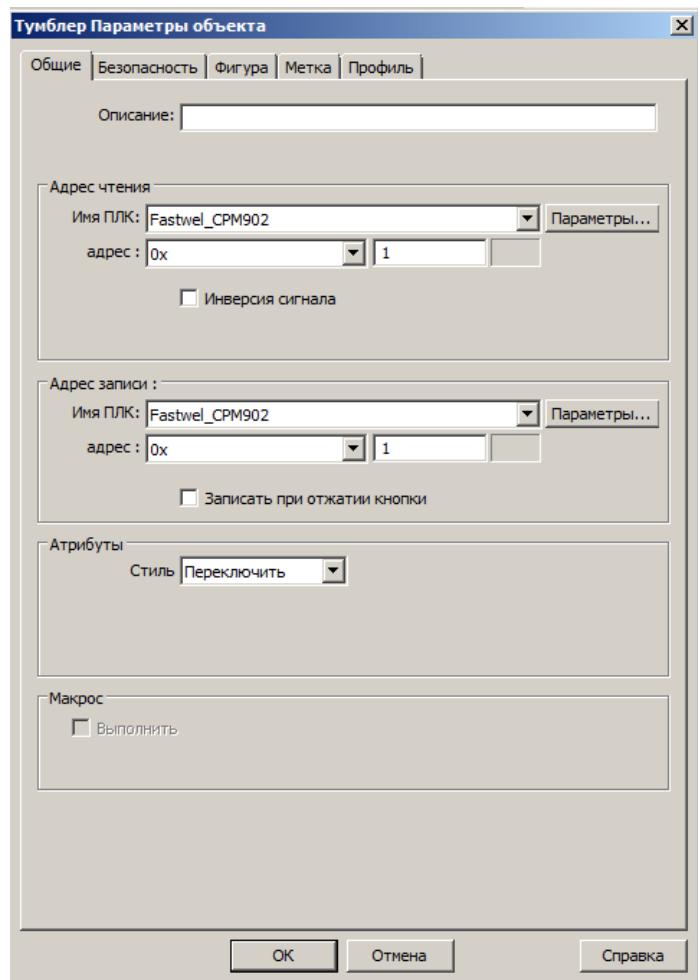


Рис. 14. Параметры элемента Битовый переключатель

3.5.2. Изменить параметры адреса записи:

– **Имя ПЛК** – выбрать из списка устройство «Fastwel\_CPM902», которое было определено в системных настройках проекта (раздел 4).

– **Адрес:**

- В первом окне выбрать код функции Modbus: **0x** – запись значения одного флага (Force Single Coil);
- Во втором окне указать адрес регистра Modbus (битового поля) для записи. Как было определено в Лабораторной работе №2, для управления первым релейным выходом со стороны мастера Modbus нужно формировать запрос на изменение состояния битового поля делимого регистра с адресом **1**. Поэтому следует установить адрес **1**.

3.5.3 Параметры адреса чтения сделать такими же.

3.5.4. Параметр **Атрибуты** определяет действие, которое будет выполняться при нажатии оператором. Выбрать «**Переключить**».

3.5.5. Перейти на вкладку **Фигура**, поставить флажок на значении «**Исп. изображение**» (рис 15).

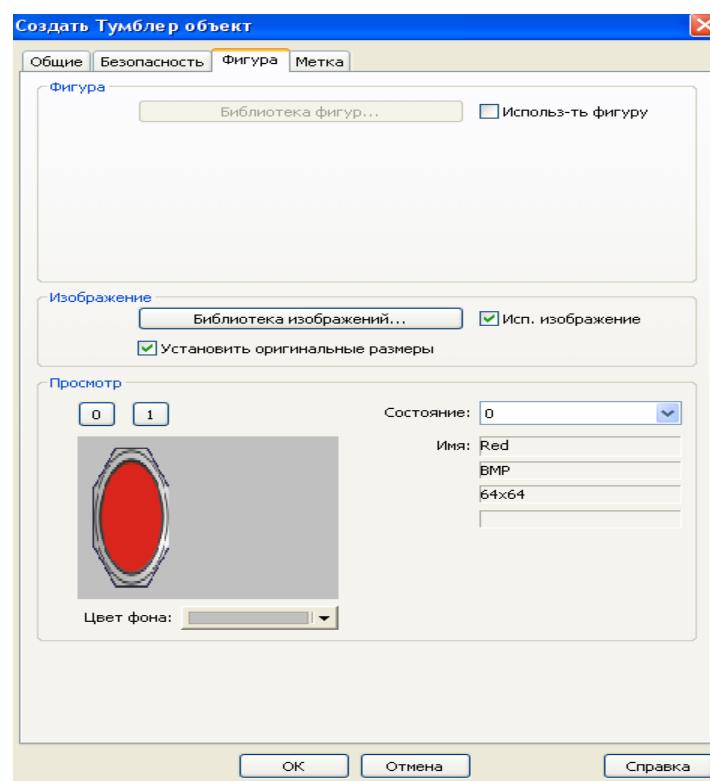


Рис. 15. Параметры элемента Битовый переключатель

3.5.6. Изменить изображение по умолчанию на изображение тумблера. Для этого перейти по кнопке к Библиотеке изображений и открыть библиотеку «**MT800\_Demo\_640x480\_0.flb**», как показано на рис. 16. После этого библиотека будет доступна в проекте.

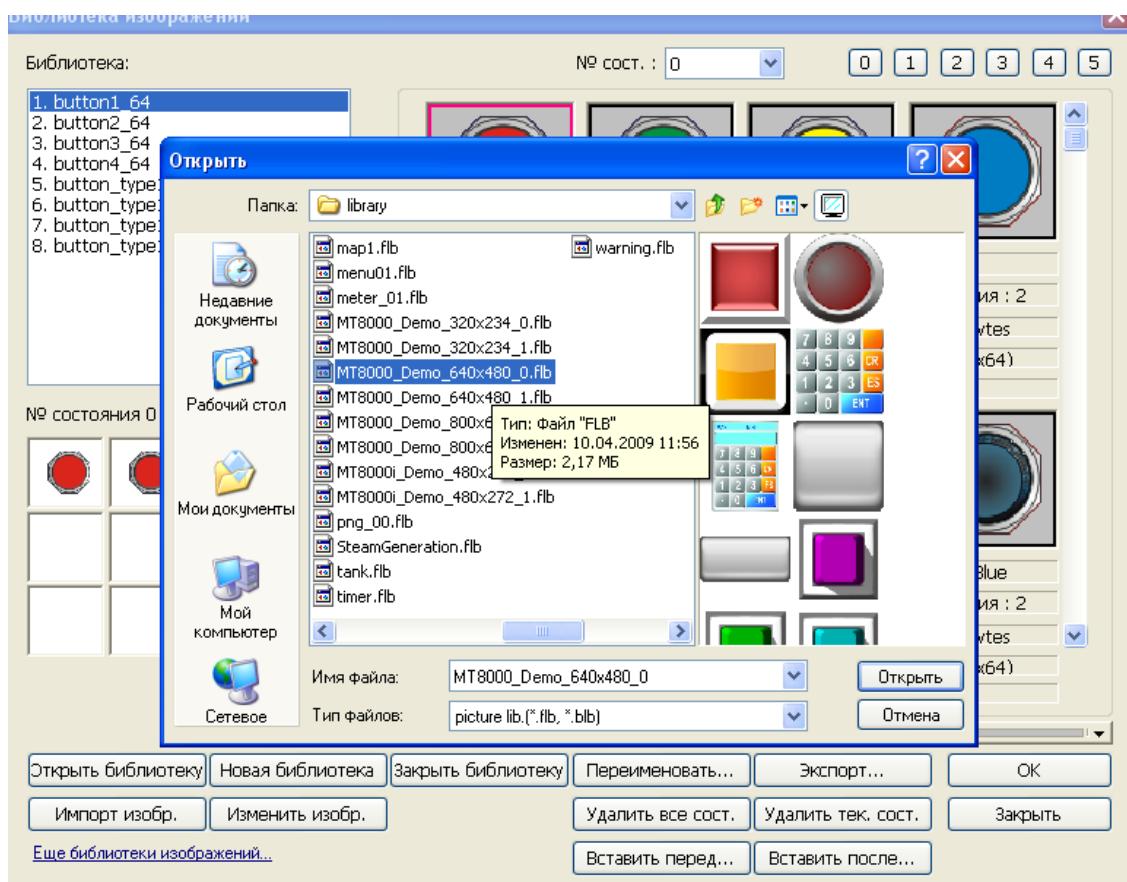


Рис. 16. Добавление библиотеки изображений в проект

3.5.7. Найти изображение тумблера (рис. 17). Нажать **OK**. Элемент Битовый переключатель с привязанным к нему изображением тумблера появится в поле окна экранной формы (рис. 18).

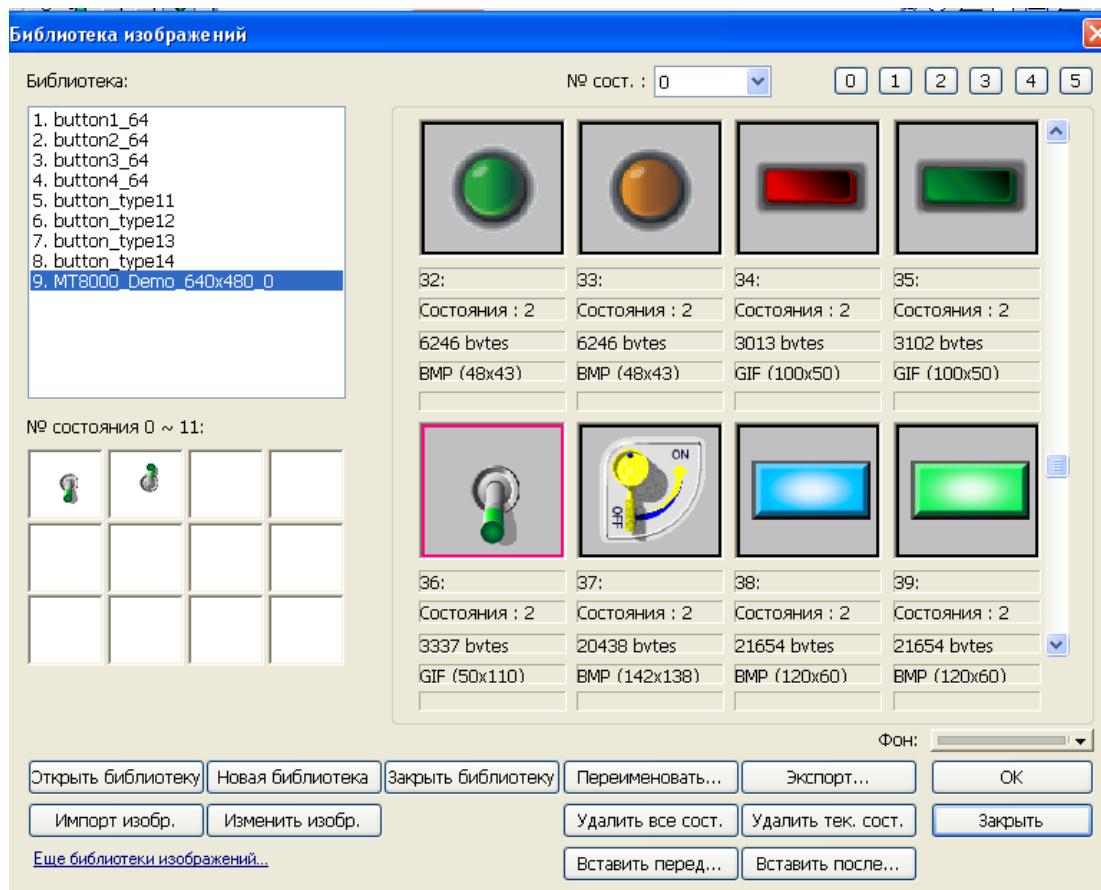


Рис. 17. Выбор изображения тумблера для элемента Битовый переключатель

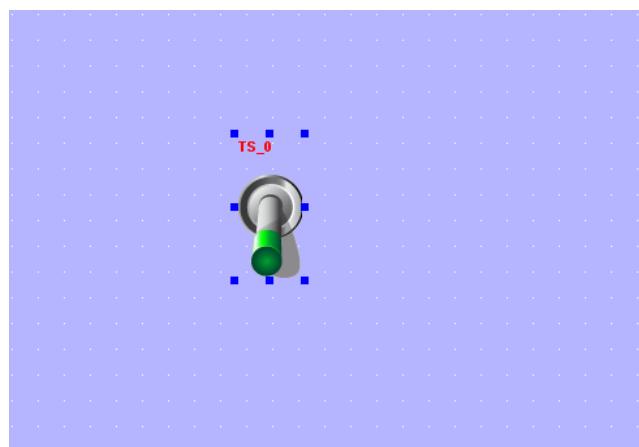


Рис. 18. Элемента Битовый переключатель с привязанным к нему изображением тумблера

3.5.8. Выделить созданный элемент. Нажать правой кнопкой мыши и выполнить копирование элемента. Вставить на свободное место поля экранной формы еще три таких тумблера (рис. 19).

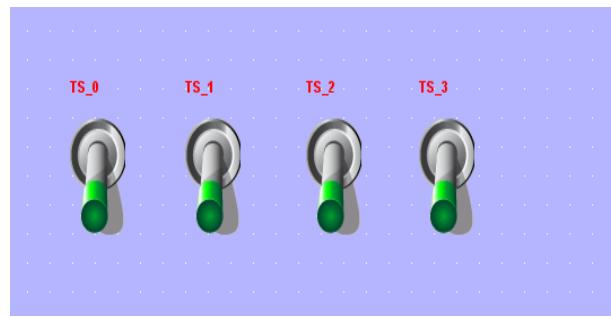


Рис. 19. Элементы для управления релейными выходами контроллера СРМ902

3.5.9. Редактировать параметры каждого элемента в соответствии с задачей лабораторной работы:

- второй тумблер должен иметь адрес «2» для записи и чтения;
- третий – адрес «3» для записи и чтения;
- четвертый – адрес «4» для записи и чтения.

3.5.10. Выбирая на панели инструментов пиктограммы «Прямоугольник» и «Текст», добавить декоративную рамку и надпись «УПРАВЛЕНИЕ РЕЛЕЙНЫМ ВЫХОДОМ» (рис. 20).



Рис. 20. Добавление надписи в окно экранной формы

3.5.11. Сохранить проект. Провести компиляцию через команду меню: **Инструменты–Компиляция.**

### 3.6 ОТЛАДКА ПРОГРАММЫ В РЕЖИМЕ ОФФЛАЙН-СИМУЛЯЦИИ

Существует два режима имитации работы панели WT3010:

**ОФЛАЙН** (полностью виртуальное исполнение без привязки к реальному оборудованию);

**ОНЛАЙН** (с подключением к компьютеру, с установленным ПО EasyBuilder8000, к реальному контроллеру, например СРМ902).

Для перехода к онлайн-симуляции выполнить команду меню **Инструменты – Оффлайн-симуляция**.

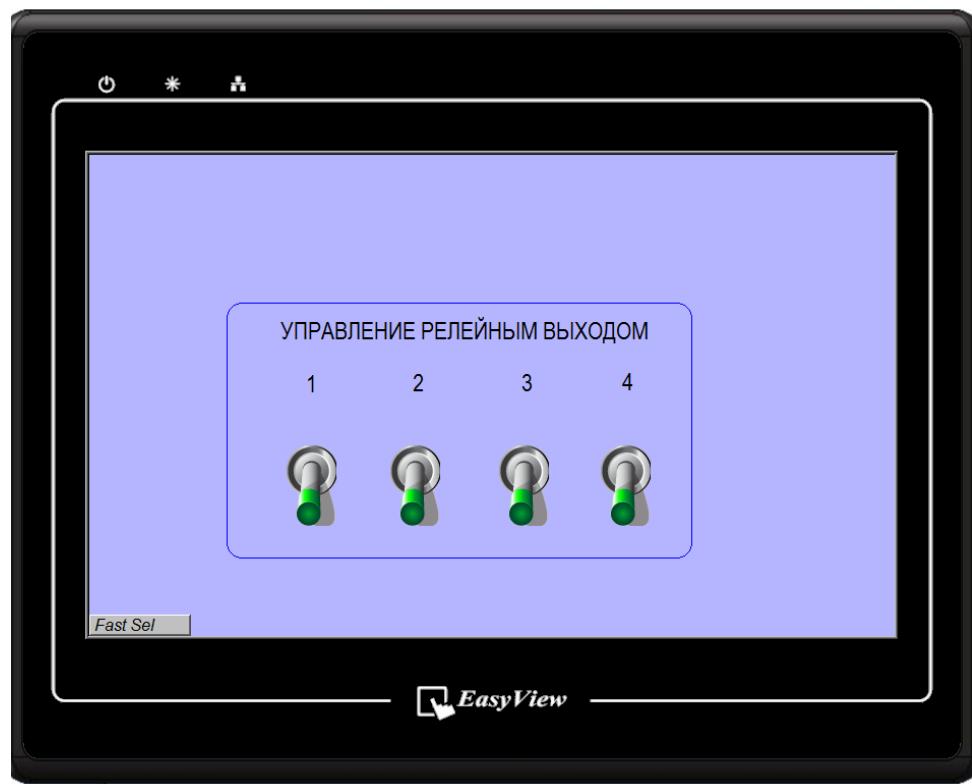


Рис. 21. Работа редактора EB8000 в режиме симуляции

## 3.7 ОТЛАДКА ПРОГРАММЫ В РЕЖИМЕ ОНЛАЙН-СИМУЛЯЦИИ

Режим имитации ОНЛАЙН выполняется при наличии соединения с контроллером и точно заданных параметрах соединения. Существует временное ограничение работы в этом режиме – **10 минут**.

Для перехода к офлайн-симуляции выполнить команду меню **Инструменты – Онлайн-симуляция**. Окно редактора EB8000 в обоих режимах (онлайн или офлайн) выглядит одинаково, как показано на рис. 21.

## 3.8 РЕДАКТИРОВАНИЕ СЕТЕВЫХ ПАРАМЕТРОВ ПАНЕЛИ WT3010

3.8.1. Для редактирования сетевых параметров панели WT3010 необходимо её включить и дождаться загрузки. После этого перейти к панели инструментов (System Toolbar), расположенной в нижней правой части экрана (рис. 22).



Рис. 22. Панель инструментов (System Toolbar), расположенная в нижней правой части экрана панели WT3010

3.8.2. Нажать кнопку и ввести пароль – по умолчанию: **111111**.



Рис. 23. Поле ввода пароля

3.8.3. Перейти на вкладку «**System settings**» (рис. 24) , установить флајок на «**IP address get from below**» и указать IP–адрес:

**192.168.1.3** для графической панели в ШК1;

**192.168.1.5** для графической панели в ШК2.



Рис. 24. Вкладка для редактирования сетевых параметров панели

3.8.4. Нажать последовательно кнопки **Apply** и **OK**.

### 3.9 ЗАГРУЗКА СОЗДАННОГО ПРОЕКТА В ПАНЕЛЬ

Для загрузки созданного ранее проекта в панель в редакторе EB8000 выполнить команду меню **Инструменты – Загрузка**.

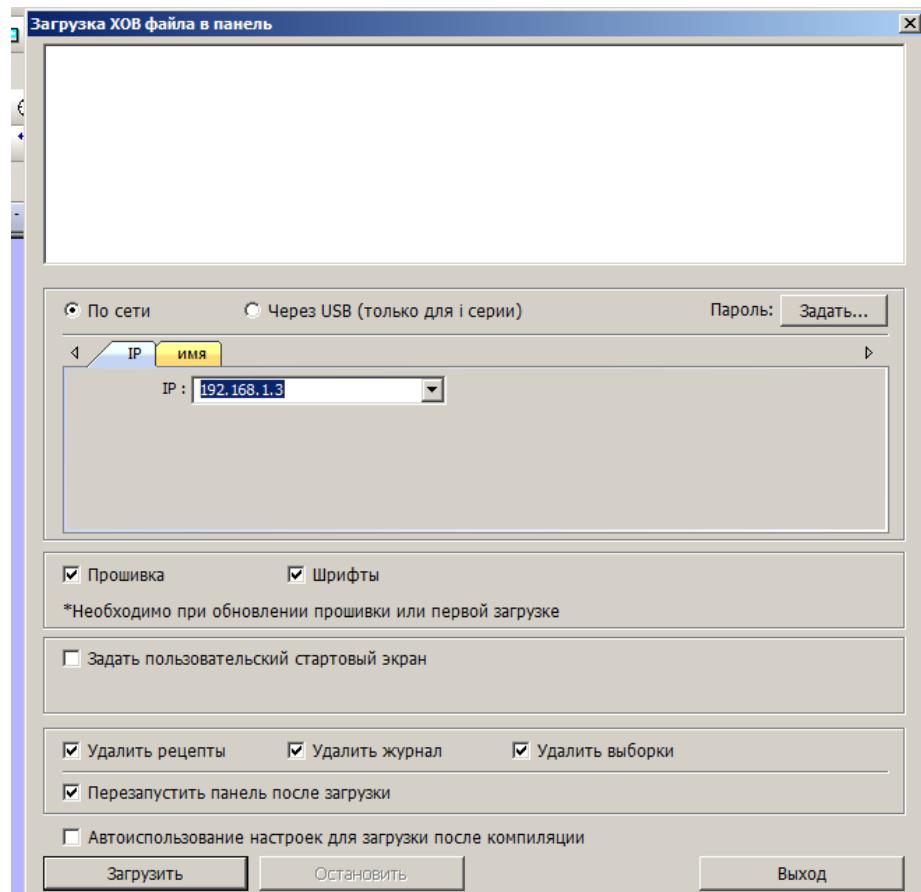


Рис. 25. Загрузка созданного проекта в панель

В появившемся окне поставить флажок, указывающий, что загрузка проекта будет выполнена по сети (рис. 25 ) и установить IP-адрес панели:

**192.168.1.3** для графической панели в ШК1;

**192.168.1.5** для графической панели в ШК2.

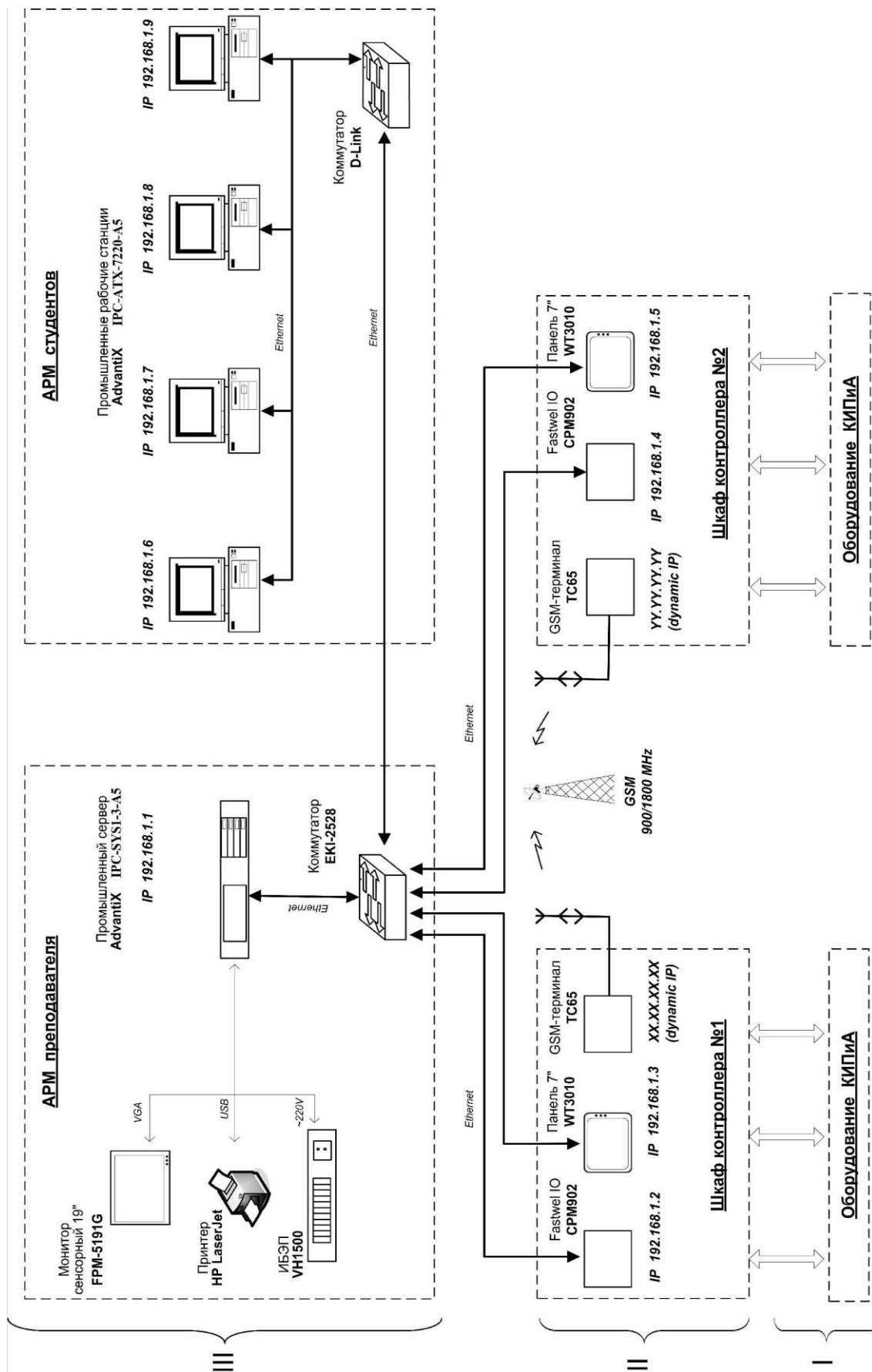
После загрузки проекта панель автоматически будет перезапущена.

## **СПИСОК ЛИТЕРАТУРЫ**

1. Бузинов С.Н., Умрихин И.Д. Исследование нефтяных и газовых скважин и пластов. М., Недра, 1984, 265с.
2. Молокович Ю.М., Непримеров Н.Н., Пикуза В.И., Штанин А.В. Релаксационная фильтрация. Казань, Изд-во Казанского университета, 1980, 136с.
3. Разработка нефтяного месторождения как комплексная междисциплинарная технология / Овчинников М.Н., Гаврилов А.Г., Непримеров Н.Н., Штанин А.В. // Наукоемкие технологии. –2004.– Т.5, №4.– с. 20–26.
4. Weintek. PLC Connection Guide.  
[http://www.weintek.com/Download/PLC\\_Connect\\_Guide/eng/PLC\\_connection\\_guide.pdf](http://www.weintek.com/Download/PLC_Connect_Guide/eng/PLC_connection_guide.pdf)
5. Описание комплекса технических средств. «Учебно-лабораторный комплекс по применению автоматизированных систем для исследования явлений переноса в пористых средах».
6. Описание комплекса программного обеспечения. «Учебно-лабораторный комплекс по применению автоматизированных систем для исследования явлений переноса в пористых средах».

## ПРИЛОЖЕНИЕ А

Структурная схема учебно-лабораторного комплекса



## ПРИЛОЖЕНИЕ В

Таблица 2. Набор модулей Fastwel I/O в составе шкафа контроллера

Номер п/п	Тип модуля	Назначение	Режим измерения
1	OM796	Модуль расширения шины FBUS	
2	AIM722 (1)	2-канальный модуль аналогового ввода сигнала 0...20 мА	Д
3	AIM722 (2)	2-канальный модуль аналогового ввода сигнала 0...20 мА	Д
4	AIM724	2-канальный модуль аналогового ввода сигнала термопар ( В - от 600°C до 1800°C, Е - от -100°C до 1000°C, J - от -100°C до 1200°C, K - от -100°C до 1370°C, N - от -100°C до 1300°C, R - от 0°C до 1700°C, S - от 0°C до 1700°C, T -100°C до 400°C)	Д
5	AIM725	2-канальный модуль аналогового ввода сигнала термосопротивлений ;2-/3-проводная соединение Pt50, Pt100, Pt200, Pt500, Pt1000, Ni100, Ni120, Cu50	Д
6	AIM728	4-канальный модуль аналогового ввода напряжения от -20 до +20 В DC;	О
7	OM751 (1)	Модуль ввода питания 24 В DC с предохранителем и диагностикой	
8	AIM730	2- канальный модуль аналогового вывода; 0-20 мА или 4-20 мА устанавливается программно;	О
9	AIM731	2-канальный модуль аналогового вывода; от минус 10 до плюс 10 В или от 0 до 10 В устанавливается программно;	О
10	OM751 (2)	Модуль ввода питания 24 В DC с предохранителем и диагностикой	
11	OM758 (1)	Модуль размножения потенциала 0 В шины питания	
12	DIM764 (1)	8-канальный универсальный модуль дискретного ввода (счётчик импульсов, измеритель частоты импульсов, измеритель временных интервалов)	О
13	OM751 (3)	Модуль ввода питания 24 В DC с предохранителем и диагностикой	
14	OM758 (2)	Модуль размножения потенциала 0 В шины питания	
15	DIM764 (2)	8-канальный универсальный модуль дискретного ввода (счётчик импульсов, измеритель частоты импульсов, измеритель временных интервалов)	О
16	OM751 (4)	Модуль ввода питания 24 В DC с предохранителем и диагностикой	
17	OM759	Модуль размножения потенциала 24 В или 48 В постоянного тока шины питания	
18	DIM717 (1)	8-канальный модуль дискретного ввода 24 В постоянного тока (переключение напряжения высокого уровня) с индикацией/ 2-канальный счётчик импульсов с частотой следования до 300 Гц	О
19	DIM717 (2)	8-канальный модуль дискретного ввода 24 В постоянного тока (переключение напряжения высокого уровня) с индикацией/ 2-канальный счётчик импульсов с частотой следования до 300 Гц	О
20	OM758 (3)	Модуль размножения потенциала 0 В шины питания	
21	DIM718	8-канальный модуль дискретного вывода 24 В постоянного тока/0,5 А (драйвер нижнего плеча) с защитой от КЗ, диагностикой и ШИМ-генерацией	
22	DIM713 (1)	2-канальный релейный выходной модуль 250 В переменного тока/30 В постоянного тока, 2 А, нормально разомкнутые контакты	
23	DIM713 (2)	2-канальный релейный выходной модуль 250 В переменного тока/30 В постоянного тока, 2 А, нормально разомкнутые контакты	
24	OM750	Оконечный модуль – заглушка шины FBUS	

Д – дифференциальный режим измерения, О – однопроводный режим измерения