

**Р.Ю. ТКАЧЕВ  
Н.З. БОЙКО**

**РАЗРАБОТКА СИСТЕМЫ  
ОПЕРАТИВНОГО КОНТРОЛЯ  
И УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ  
ПРОЦЕССОМ НА БАЗЕ SCADA-систем**

**УЧЕБНО-МЕТОДИЧЕСКОЕ ПОСОБИЕ**

**Алчевск, 2020**

**Алчевск, 2020**

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
ЛУГАНСКОЙ НАРОДНОЙ РЕСПУБЛИКИ  
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«ДОНБАССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

КАФЕДРА АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ  
ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ

Р. Ю. Ткачев, Н. З. Бойко

**РАЗРАБОТКА СИСТЕМЫ ОПЕРАТИВНОГО КОНТРОЛЯ  
И УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ  
НА БАЗЕ SCADA-систем**

Курсовое проектирование  
по дисциплине  
«Программирование компьютерно-интегрированных систем»

Учебно-методическое пособие

Алчевск  
2020

УДК 621.6:049.77

ББК 32.79

Т48

**Ткачев Роман Юрьевич** — кандидат технических наук, доцент кафедры автоматизированного управления технологическими процессами ГОУ ВО ЛНР «ДонГТИ» (г. Алчевск);

**Бойко Николай Зельманович** — кандидат технических наук, доцент кафедры автоматизированного управления технологическими процессами ГОУ ВО ЛНР «ДонГТИ» (г. Алчевск).

***Рецензенты:***

*Д. В. Кобец* — кандидат технических наук, доцент кафедры автоматизированного управления технологическими процессами ГОУ ВО ЛНР «ДонГТИ» (г. Алчевск);

*А. Н. Баранов* — кандидат технических наук, доцент кафедры автоматизированного управления технологическими процессами ГОУ ВО ЛНР «ДонГТИ» (г. Алчевск);

*В. Д. Рябичев* — доктор технических наук, профессор, ректор ГОУ ВО ЛНР «ЛНУ им. В. Даля» (г. Луганск).

*Рекомендовано Ученым советом ГОУ ВПО ЛНР «ДонГТУ»  
(Протокол № 7 от 30.03.2018)*

**Ткачев Р. Ю.**

Т48

Разработка системы оперативного контроля и управления технологическим процессом на базе SCADA-систем. Курсовое проектирование по дисциплине «Программирование компьютерно-интегрированным систем»: учебно-методическое пособие / Р. Ю. Ткачев, Н. З. Бойко. — Алчевск : ГОУ ВО ЛНР «ДонГТИ», 2020. — 51 с.

Изложены цели, требования и содержание к оформлению, методические указания по выполнению курсового проекта, призванного углубить и закрепить знание курса «Программирование компьютерно-интегрированных систем».

УДК 621.6:049.77

ББК 32.79

© ГОУ ВО ЛНР «ДонГТИ», 2020

© Р. Ю. Ткачев, Н. З. Бойко, 2020

© Н. В. Чернышова, художественное оформление обложки, 2020

## СОДЕРЖАНИЕ

1. Цель курсового проектирования .....	4
2. Объем и структура курсового проекта .....	5
2.1. Структура пояснительной записки .....	5
2.2. Структура графической части .....	5
3. Последовательность работы над проектом и критерии оценивания ...	6
3.1. Контроль за выполнением курсового проекта .....	6
3.2. Критерии оценивания курсового проекта.....	8
3.3. Процедура защиты курсового проекта .....	10
4. Методические указания по выполнению курсового проекта .....	12
4.1. Титульный лист и лист задания .....	12
4.2. Реферат .....	12
4.3. Содержание .....	12
4.4. Введение .....	12
4.5. Основная часть .....	15
4.5.1. Принципы построения проекта в Trace Mode.....	15
4.5.2. Создание экранов АРМ (мнемосхем).....	17
4.5.3. Привязка элементов и переменных. Отладка связей .....	22
4.5.4. Создание программ и каналов .....	24
4.5.5. Создание узлов проекта и базы каналов .....	28
4.5.7. Имитация объекта регулирования. Интерфейс связи ОРС ....	44
4.6. Заключение .....	46
4.7. Перечень ссылок .....	46
5. Правила оформления курсового проекта .....	47
Перечень ссылок .....	48
Приложение А.....	50
Приложение Б .....	51

## 1. ЦЕЛЬ КУРСОВОГО ПРОЕКТИРОВАНИЯ

Целью данного курсового проекта по ПКИС является закрепление студентом знаний принципов создания (проектирования) многозадачных систем автоматического или автоматизированного управления технологическими процессами на базе SCADA-систем, освоения методов программирования компьютерно-интегрированных систем, организации обмена данными между различными программами с использованием стандартов OPC.

Курсовое проектирование играет большую роль в развитии самостоятельной работы студентов, способствует закреплению теоретических знаний, усвоению методов конструирования. Выполняя курсовой проект, студент должен практически освоить правила оформления технической документации, получить навыки работы со специальной литературой, проектными материалами, каталогами и справочниками. Курсовое проектирование помогает студенту подготовиться к выполнению более сложной инженерной задачи – дипломному проектированию и способствует:

- расширение технического кругозора: при изучении государственных стандартов на проектирование автоматизированных систем управления, руководящих технических материалов и другой специальной литературы;

- развитие творческой инициативы и самостоятельности при решении поставленных задач, стремление к поиску оригинальных решений; ознакомление с современными научно-техническими решениями в области проектирования АСУТП типовых технологических процессов;

- развитию навыков работы со специальной технической и справочной литературой, технически и литературно грамотного изложения материала;

- развитие чувства ответственности за выполненную работу, принятые решения и умения обосновать и защищать их.

## **2. ОБЪЕМ И СТРУКТУРА КУРСОВОГО ПРОЕКТА**

Курсовой проект состоит из пояснительной записки объемом в количестве 25–30 страниц и графической части в количестве двух листов формата А3.

### **2.1. Структура пояснительной записки**

Пояснительная записка (ПЗ) должна включать:

- титульный лист (1 с.);
- задание (1 с.);
- реферат (1 с.);
- содержание (1–2 с.);
- введение (1 с.);
- основную часть (10–23 с.);
- заключение (1 с.);
- перечень ссылок (1 с.);
- приложения (1–5 с.).

Основная часть ПЗ должна включать следующие разделы:

- создание экранов автоматизированного рабочего места АРМ (5 с.);
- создание программ и каналов (3–5 с.)
- создание узлов проекта и базы каналов (3–5 с.)
- создание имитационной модели технологического объекта (4–7 с.)
  
- имитация проекта (8 с.)

### **2.2. Структура графической части**

Графическая часть курсового проекта должна включать:

- монитор автоматизированного рабочего места диспетчера (оператора) технологического процесса и управления (А3);
- монитор настройки параметров регулятора(-ов) и трендов с имитацией работы компьютерно-интегрированной системы (А3).

### **3. ПОСЛЕДОВАТЕЛЬНОСТЬ РАБОТЫ НАД ПРОЕКТОМ И КРИТЕРИИ ОЦЕНИВАНИЯ**

#### **3.1. Контроль за выполнением курсового проекта**

Работа над курсовым проектом начинается с первой недели семестра и завершается за две недели до его окончания. На первой неделе семестра ведущий преподаватель выдает каждому студенту конкретное индивидуальное задание на выполнение проекта. Допускается выдача общего задания на курсовой проект для группы студентов (2–3 чел.) с обязательным выделением пунктов индивидуального выполнения для каждого участника группы. Ответственным за курсовое проектирование является преподаватель, читающий курс лекций по дисциплине «Проектирование компьютерно-интегрированных систем». В задании указывается тема проекта, и перечисляются поставленные перед студентом задачи; устанавливаются объемы пояснительной записки и графической части; приводится календарный план на выполнение составных частей и проекта в целом; приводится список рекомендуемой литературы.

Работа над курсовым проектом предполагает выполнение определенной последовательности этапов (табл. 1). Ход работы над КП контролируется преподавателем в отведенные для этого учебные часы

Первая ошибка незадачливого студента, начинающего впервые создавать проект автоматизированной системы контроля и управления и строить самостоятельно виртуальную модель системы управления состоит в том, что студент пытается сделать работу сразу и целиком. К этому его подталкивает кажущаяся простота рассмотренных им примеров моделей или проектов. Естественно, целиком и сразу такую работу выполнить, тем более качественно, не возможно. Проект создается маленькими частями, которые последовательно соединяются друг с другом, так, чтобы они бесконфликтно и согласованно работали параллельно. При этом неизбежны ошибки, которые нужно выявлять и исправлять. Поэтому в процессе выполнения проекта руководитель проекта проводит консультации для разъяснения возникающих при проектировании вопросов и контролирует своевременность и качество выполнения его составных частей в соответствии с календарным планом и гра-

фиком самостоятельной работы студентов. Для получения максимальной оценки каждая составная часть проекта должна быть представлена на проверку не позднее указанного в графике срока. По завершении проекта студент подписывает его и предъявляет руководителю. Если проект удовлетворяет предъявляемым к нему требованиям, руководитель допускает его к защите, о чем на чертежах и пояснительной записке делается соответствующая запись.

Таблица 1 — Последовательности этапов выполнения КП

№ этапа	Выполняемая работа	Результат
1	Анализ технического задания	Полное представление о требованиях, предъявляемых к проектируемой АСУТП (назначению, принципу работы и т.д.)
2	Выбор функциональной схемы автоматизации, элементов для реализации проекта и определение количества входных и выходных переменных. Разработка Экрана(-ов) монитора АРМ (мнемосхемы) ,	Обоснование элементов и узлов проекта. Allocation list (список) выбранных переменных. Экран(-ы) монитора АРМ оперативного персонала
3	Разработка алгоритмов обработки данных и управления, программ, каналов связей и узлов проекта и баз данных каналов	Описание работы разработанных алгоритмов и листинги разработанных программ или FBD /LD -диаграммы.
4	Разработка имитационной модели объекта регулирования в приложении МАТЛАБ и обеспечение интерфейса связи OPC –сервера модели объекта и SCADA-системы	Имитационная модель объекта регулирования в приложении МАТЛАБ. Программа обеспечения связи модели с OPC- сервером
5	Оформление ПЗ и графической части в соответствии с требованиями ГОСТ 2.105 -95 и ЕСКД	Пояснительная записка и чертежи



### 3.2. Критерии оценивания курсового проекта

Курсовой проект оценивается по стобалльной системе, в зависимости от уровня его выполнения и защиты.

Анализ результатов курсового проектирования проводится по следующим критериям:

1. Навыки самостоятельной работы с материалами, по их обработке, анализу и структурированию.

2. Умение правильно применять методы анализа и проектирования SCADA-системы.

3. Умение грамотно интерпретировать полученные результаты.

4. Способность осуществлять необходимые расчеты, получать результаты и грамотно излагать их в отчетной документации.

5. Умение выявить проблему, предложить способы ее разрешения, умение делать выводы.

Пункты с 1 по 5 дают до 32% вклада в итоговую оценку студента.

6. Умение оформить итоговый отчет в соответствии со стандартными требованиями.

Пункт 6 даёт до 18% вклада в итоговую оценку студента.

7. Умение защищать результаты своей работы, грамотное построение речи, использование при выступлении специальных терминов.

8. Способность кратко и наглядно изложить результаты работы.

Пункты 7,8 дают до 50% вклада в итоговую оценку студента.

Оценка «отлично» ставится студенту, который в срок, в полном объеме и на высоком уровне выполнил курсовой проект. При защите и написании работы студент продемонстрировал вышеперечисленные навыки и умения. Тема, заявленная в работе, раскрыта полностью, все выводы студента подтверждены материалами исследования и расчетами. Отчет подготовлен в соответствии с предъявляемыми требованиями.

Оценка «хорошо» ставится студенту, который выполнил курсовую работу, но с незначительными замечаниями, был менее самостоятелен и инициативен. Тема работы раскрыта, но выводы носят поверхностный характер, практические материалы обработаны не полностью.

Оценка «удовлетворительно» ставится студенту, который допускал просчеты и ошибки в работе, не полностью раскрыл заявленную те-

му, делал поверхностные выводы, слабо продемонстрировал аналитические способности и навыки работы с теоретическими источниками.

Оценка «неудовлетворительно» ставится студенту, который не выполнил курсовую работу, либо выполнил с грубыми нарушениями требований, не раскрыл заявленную тему, не выполнил практической части работы.

Таким образом, оценка курсового проекта состоит из трех составляющих: количества баллов за своевременность и инициативность выполнения (пп.1–5) (до 32 баллов), количества баллов за оформление пояснительной записки (до 18 баллов) и количества баллов, отведенных за защиту курсового проекта (до 50 баллов)

1) Критерии оценивания количества баллов за своевременность выполнения (до 32 баллов).

Графиком выполнения курсового проекта предусмотрена сдача на 13 неделе для окончательной проверки, а на 15 неделе его защита. В течение этих 14 недель проводилась консультация, и контроль выполнения студентом каждой части курсового проекта. Выполненная в поставленные сроки часть оценивалась по 8 баллов за 2 недели. Итого  $8 \times 4 = 32$  балла.

Студент, не предоставлявший своевременно на контроль выполненные им части курсового проекта не получает указанные баллы.

2) Критерии оценивания количества баллов за оформление пояснительной записки (п.6) (до 18 баллов).

В этой части курсовой проект оценивается по содержанию и оформлению пояснительной записки, состоящей из всех частей и графической части. Проект, не содержащий после окончательной проверки замечаний оценивается по максимуму в 18 баллов.

В курсовом проекте, содержащем замечания количество баллов снижается на 1 балл за ошибку. После выданных замечаний студенту дается срок (до 5 дней) для внесения исправлений в курсовой проект. За каждое исправленное замечание студенту возвращается 0,5балла.

3) Критерии оценивания количества баллов, отведенных за защиту курсового проекта (пп. 7–8) (до 50 баллов)

Студент, выполнивший курсовой проект и оформивший ПЗ в соответствии с требованиями допускается к защите.

### 3.3. Процедура защиты курсового проекта

Защита курсового проекта производится в присутствии студентов группы перед комиссией, состоящей из преподавателей выпускающей кафедры и руководителя проектирования.

Защита курсового проекта представляет собой устный публичный отчет студента, на который ему отводится 7–8 минут и ответы на 5 вопросов по теме. Устный отчет студента включает: раскрытие целей и задач проектирования, его актуальность, описание выполненного проекта, основные выводы и предложения, разработанные студентом в процессе курсового проектирования.

В основной части доклада следует большее внимание уделять постановке задач, обоснованию способа их экспериментального решения, т.е. ответам на вопросы «Зачем решается задача, какова цель?» и «Почему задача решалась именно так, а не иначе?», анализу полученных результатов и выводам. Не следует докладывать в форме: мы сначала сделали это, а потом это.

Может потребоваться запуск выполненного студентом проекта, поэтому его тоже нужно представить на карте памяти (флеш-карте).

За ответ на поставленный вопрос дается 0–10 баллов.

8–10 баллов студент получает, если: ответы построены логично, полные и правильные, содержат анализ, обобщение и систематичность. А сам студент умеет самостоятельно находить и пользоваться источниками информации, устанавливать причинно-следственные связи и делает аргументированные выводы. Умеет правильно и осознанно применять все виды технической, конструкторской и технологической документации в рамках учебной программы. Проявляет познавательно-творческий интерес к выбранной профессии, в своих ответах и суждениях выходит за рамки учебной программы. Ответ строит на уровнях самостоятельного и творческого мышления.

5–8 баллов студент получает, если: ответы в целом правильные, логичные и обоснованные.

Студент достаточно осознанно пользуется справочной информацией, технической и конструкторско-технологической документацией. В ответах допускает незначительные ошибки, которые способен само-

стоятельно исправить, владеет основным учебным материалом и самостоятельно с пониманием воспроизводит его в устной, письменной и графической форме. Дает определение основных понятий, анализирует, сравнивает и систематизирует информацию.

3–5 балла студент получает, если: студент, при ответе допускает неточности и недостаточно правильные формулировки, нарушения последовательности в изложении материала. Основные понятия даются с ошибками, которые в полной мере самостоятельно исправить не может. Информация недостаточно обосновывается и сравнивается.

1–3 балла студент получает, если: при ответах допускает существенные ошибки, которые самостоятельно исправить не может, а воспроизводит при помощи преподавателя основной материал или его элементы на уровне знакомства, без понимания.

## **4. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КУРСОВОГО ПРОЕКТА**

### **4.1. Титульный лист и лист задания**

Титульный лист выполняется по установленной форме, общей для всех курсовых проектов, задание оформляется на специальном бланке, в соответствии с темой проекта.

### **4.2. Реферат**

Реферат курсового проекта должен содержать сведения об объеме ПЗ (количество страниц) и графической части (количество и формат листов), количестве иллюстраций, таблиц, использованных источников, перечень ключевых слов, включающий от 5 до 10 слов и словосочетаний, которые характеризуют содержание проекта, текст реферата.

Текст реферата должен содержать: объект разработки; цель работы, виды разработок; цель работы, виды разработок, полученные результаты и их новизну, основные конструктивные и технологические характеристики.

Стиль и последовательность изложения информации в реферате приведены в Приложении А.

### **4.3. Содержание**

Содержание включает номера и наименования разделов и подразделов ПЗ с указанием страниц, на которых они начинаются. Содержание включают в общее количество страниц ПЗ. Пример содержания ПЗ курсового проекта приведен в Приложении Б.

### **4.4. Введение**

Введение должно содержать назначение и область применения разработки, оценку современного состояния решаемой задачи, основание для разработки.

При решении задач реализации систем управления обычно используются специализированные программные пакеты, одним из разновидностей которых являются SCADA-системы.

SCADA-системы (Supervisory Control and Data Acquisition) — это системы диспетчерского контроля, управления и сбора данных о технологических процессах и работе участков, цехов и предприятия в целом.

SCADA-система состоит из компьютерной (виртуальной) части и физической. Компьютерная часть это программы, мнемосхемы, базы данных, позволяющие оператору (диспетчеру), а также в автоматическом режиме контролировать технологический процесс, управлять его ходом и документировать его параметры, осуществлять связь и обмен информацией с внешними элементами системы управления и другими компьютерами и контроллерами, участвующими в управлении.

Физическая часть SCADA-системы это оконечные устройства управления: датчики, преобразователи, исполнительные устройства, непосредственно воздействующие на объекты управления, которые, например, составляют технологическую цепочку, а также промежуточные управляющие устройства (встроенные в управление подсистемами компьютеры, программируемые логические контроллеры) и линии связи с этими устройствами (рис. 1).

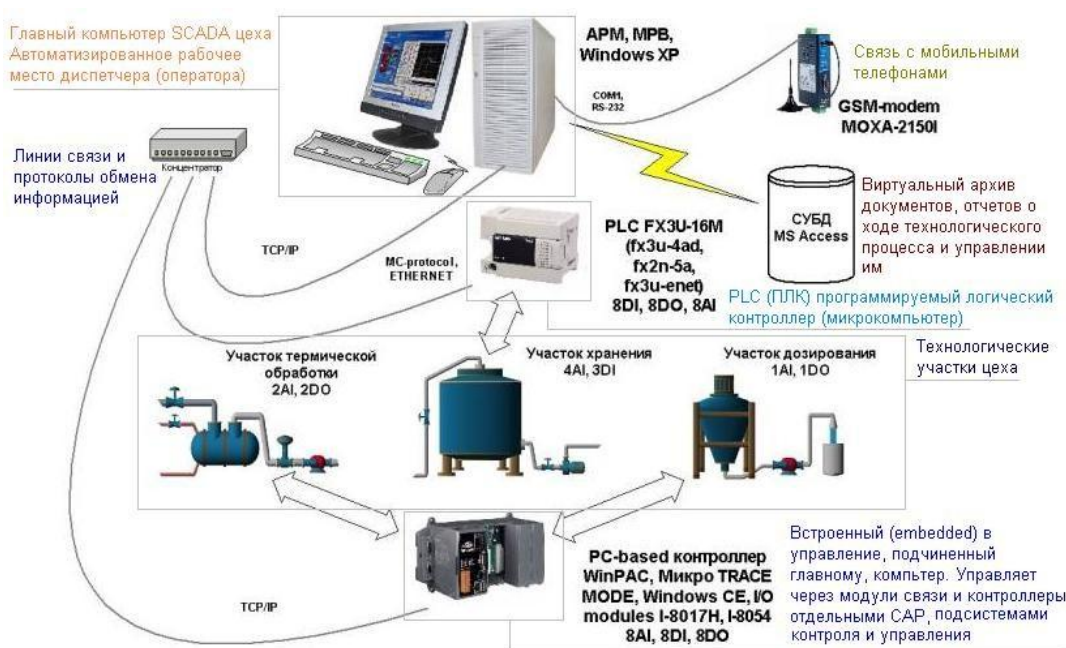


Рисунок 1 — Основные элементы физической части SCADA-системы

Проектирование систем автоматизации (АСУТП и АСУ) предполагает и создание проекта построения физической части системы SCADA, и ее виртуальной части.

Проектирование физической части осуществляется в соответствии с требованиями технологии, и состоит в выборе и определении предполагаемых мест размещения физических элементов автоматизации, моделировании и оптимизации параметров отдельных САР.

Проектирование виртуальной части SCADA системы проводится в специальной программе, редакторе проекта SCADA и состоит в выборе и задании свойств компонентов проекта SCADA-системы (мнемосхем, программ обработки информации и выработки сигналов и команд, протоколов и драйверов устройств и модулей связи, форм и порядка документирования хода процесса управления и др.) и соединении их посредством некоторых переменных (аргументов) друг с другом, а также с внешними устройствами.

Такого рода проекты создаются в общей среде, как для АРМ, так и для подчиненных им встроенных в управление (embedded) компьютеров и контроллеров.

Реализация виртуальной части проекта осуществляется путем запуска на компьютере АРМ диспетчера специальной программы, Монитора реального времени (МРВ), куда загружаются файлы, созданные на стадии проектирования в Редакторе проекта SCADA. Используя МРВ диспетчер, наблюдая за мнемосхемой (виртуальным диспетчерским пультом) имеет представление о текущем состоянии технологического процесса, возможность вмешаться в управление им, а также может получить и проанализировать его историю. Кроме того, и сам Монитор, в соответствии с логикой, заложенной на стадии проектирования виртуальной части SCADA, осуществляет контроль и управление технологическим процессом.

Обычно фирмы-разработчики предоставляют потребителю, проектировщику системы автоматизации, т.н. Интегрированную среду (ИС), т.е. совокупность Редактора проекта SCADA-системы и программу его реализации - Монитор реального времени (МРВ).

Существует довольно много программ для разработки и реализации SCADA-систем: Trace Mode (Москва, Россия), WinCC (Сименс),

PcVue (Франция), SIMP Light SCADA (Нижний Тагил, Россия), Wonderware HMI/SCADA (Klinkman) и др. К SCADA-системам иногда относят и известную и солидную программу LabView, которая в большей мере используется для автоматизации управления физическими техническими экспериментами.

## **4.5. Основная часть**

### **4.5.1. Принципы построения проекта в Trace Mode**

Вспомним основные возможности SCADA-систем:

- сбор первичной информации от устройств нижнего уровня;
- архивирование и хранение информации для последующей обработки (создание архивов событий, аварийной сигнализации, изменения технологических параметров во времени, полное или частичное сохранение параметров через определенные промежутки времени);
- визуализация процессов;
- реализация алгоритмов управления, математических и логических вычислений (имеются встроенные языки программирования типа VBasic, Pascal, C и др.), передача управляющих воздействий на объект;
- документирование как технологического процесса, так и процесса управления (создание отчетов), выдача на печать графиков, таблиц, результатов вычислений и др.;
- сетевые функции (LAN, SQL);
- защита от несанкционированного доступа в систему;
- обмен информацией с другими программами (например, Outlook, Word и др. через DDE, OLE и т.д.).

Таким образом, на момент создания проекта виртуальной части SCADA-системы должны быть известны:

- технологические процессы и оборудование, на котором они протекают;
- цели и задачи системы управления этими процессами, их контроля и документирования их работы;
- физическое оборудование и элементы системы управления (датчики, исполнительные устройства, модули связи, контроллеры и встроенные компьютеры и др.).



– функционально-технологическая или функционально-структурная схема, на которой должны быть отображены все информационные связи и все переменные и параметры технологического процесса, которые необходимы для его контроля и управления им.

Как уже отмечалось, функционально-структурные схемы могут быть представлены в самых разных формах. Главное, чтобы на них были отображены все элементы системы управления, как физические, так и виртуальные, и указано какую переменную (параметр управления) куда нужно передать и зачем.

Построение проекта виртуальной части SCADA-системы проводится в Редакторе проекта Trace Mode и состоит из четырех основных этапов:

– отображении виртуальных элементов системы управления (т.н. компонентов проекта);

– созданию связей (привязок значений источников и приемников к переменным) посредством виртуальных каналов связи;

- отладке в режиме имитатора работы виртуальной части SCADA-системы;

– подключении виртуальной части к реальным внешним устройствам и проверка состоятельности проекта, т.е. работоспособности, соответствия его работы во всех возможных режимах технологического процесса в соответствии с требованиями к управлению им.

Технология интегрированной разработки АСУ ТП объединяет программирование, как операторского интерфейса, так и промышленных контроллеров.

Инструментальная система состоит из следующих редакторов:

– редактор базы каналов;

– редактор представления данных (РПД);

– редактор шаблонов.

Редактор базы каналов необходим для разработки структуры проекта, а также включает математические основы обработки данных и управления (распределенная база реального времени):

– описываются конфигурации всех рабочих станций, УСО, контроллеров;

– настраиваются информационные потоки между ними;

- описываются входные, выходные сигналы, их связь с устройствами сбора данных и управления;

- настраиваются законы первичной обработки данных, технологические границы;

- осуществляется настройка архивирования и сетевого обмена и т.д.

РПД предназначен для разработки графической составляющей проекта:

- создание статических рисунков технологического процесса;

- динамические формы отображения и управления накладываются на статику.

Редактор шаблонов используется для разработки шаблонов документов.

Кроме того, в интегрированную среду разработки TRACE MODE® 6 встроены:

- редактор программ;

- построитель связей с СУБД;

- редактор паспортов оборудования (EAM);

- редактор персонала (HRM);

- редактор материальных ресурсов (MES).

#### **4.5.2. Создание экранов АРМ (мнемосхем)**

Принцип построения проекта виртуальной части SCADA состоит в отображении в Навигаторе проекта (специальных окнах) всех элементов системы управления, контроля, связи и документирования, включая экраны мнемосхем, программы обработки информации и управления, каналы связи, источники пробных сигналов для имитации работы виртуальной части системы и др. и связи их посредством переменных (аргументов) и специальных виртуальных каналов.

Построение любого проекта системы автоматизации осуществляется в четыре этапа:

- разработка функционально-структурной схемы проекта;

- разработка математических моделей элементов технологического процесса и алгоритма управления им;

- создание и отладка проекта в Trace Mode;

- запуск Монитора реального времени (реализация проекта).

В Trace Mode довольно много похожих или даже на первый взгляд совершенно одинаковых экранов и таблиц, что может запутать студента при освоении программы.

Итак, шаблоны экранов и каналы их вызова, чем они похожи и в чем их отличие. Шаблон экрана это графический экран, мнемосхема, на которой отображены статические и динамические Графические элементы, а также совокупность формальных и фактических переменных, которые могут быть связаны с Графическими элементами, с некоторыми их свойствами.

Графические элементы это картинки, которые не изменяются (статические), или изменяются (динамические) в соответствии с изменением значений привязанных к ним переменных при работе Монитора реального времени.

Переменные (аргументы) шаблона экрана задаются проектировщиком и могут быть формальными, т.е. не привязанными ни к какому источнику значений, а могут быть и фактическими, т.е. привязанными к некоторому конкретному источнику значений.

Шаблоны экрана называются так потому, что они могут быть использованы несколько раз, например, на производстве есть несколько участков с совершенно одинаковыми технологическими нитками. Поэтому у них и графические экраны практически одинаковые, за исключением привязки переменных к датчикам одного типа, но находящихся на разных участках, да названия участков отличаются. Шаблоны экранов можно копировать и вставлять в Навигаторе проекта. Тогда каждый из них можно редактировать отдельно.

Мнемосхемы являются важной частью SCADA-систем и эстетика их оформления играет не последнее роль в обеспечении качества работы диспетчера или оператора технологического процесса.

Мнемосхема - рисунок на экране, с кнопками, графиками, индикаторами, значения которых могут изменяться. Диспетчер наблюдает за ходом технологического процесса и поведением системы управления по мнемосхеме и может с помощью мышки и клавиатуры вмешиваться при необходимости в процесс управления.

Создание мнемосхемы, т.е. заполнение поля экрана графическими элементами. Часть из них может быть статическими, т.е. картинки и

тексты, а часть — иметь динамическую привязку, т.е. изменяться с течением времени по мере изменения переменных, к которым привязаны какие-то параметры этих графических элементов, например, текст в текстовом поле или значение кривой на графике.

На рисунке 2 представлен экран монитора системы управления водоохлаждением крышки установки печь-ковш.

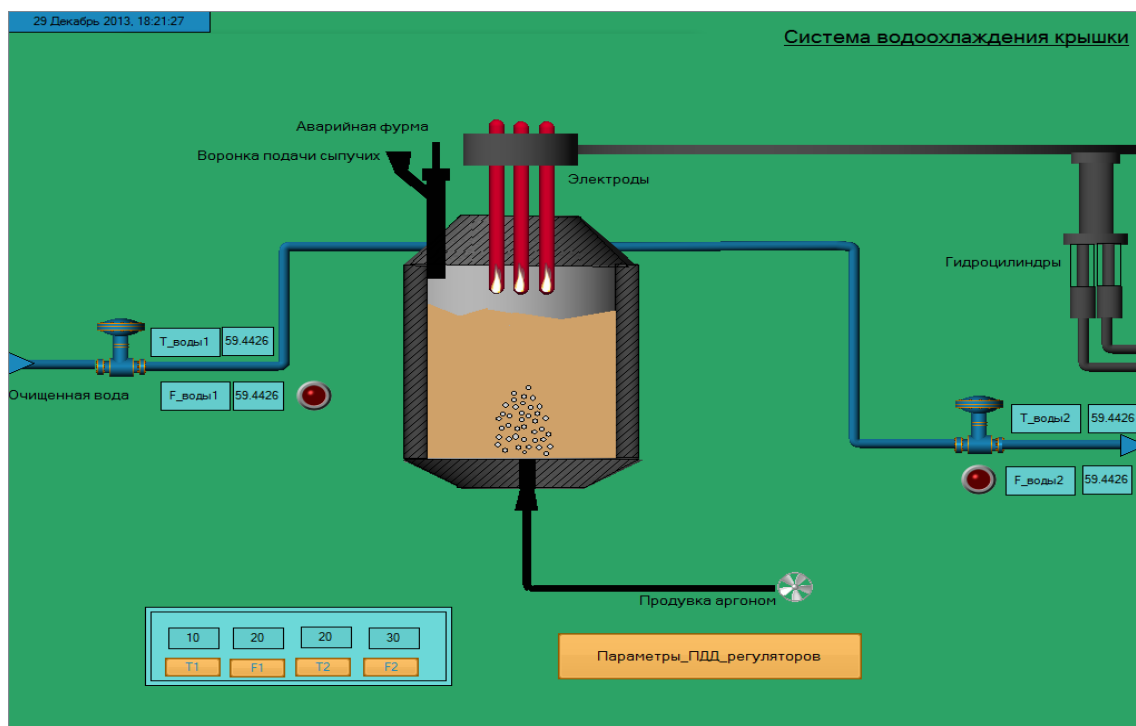


Рисунок 2 — Вид экрана АРМ «Система водоохлаждения крышки установки печь-ковш»

Канал вызова (С) шаблона экрана не только его вызывает, но и заменяет формальные переменные (аргументы) на фактические, т.е. осуществляет их привязку к конкретным элементам мнемосхемы или источникам значений. Чтобы не запутаться в процессе проектирования полезно, хотя и не обязательно, называть формальные переменные в шаблоне экрана и фактические переменные в канале по-разному.

При автоматическом построении каналов переменных Редактор проекта Trace Mode сам уточняет названия фактических переменных так, чтобы они не совпадали с названием формальных переменных.

Свойства (правой кнопкой, см. рис.3 окна с таблицами переменных, представленных на вкладке Аргументы), шаблона экрана Экран#1

и Канала Экран#1:1 вызова этого экрана отличаются: У шаблона две вкладки, у канала — четыре; названия (имена) переменных (аргументов) в Свойствах могут как совпадать, так и отличаться. Переменные шаблона экрана могут не иметь привязки к источникам или приемникам их значений, а могут и иметь.

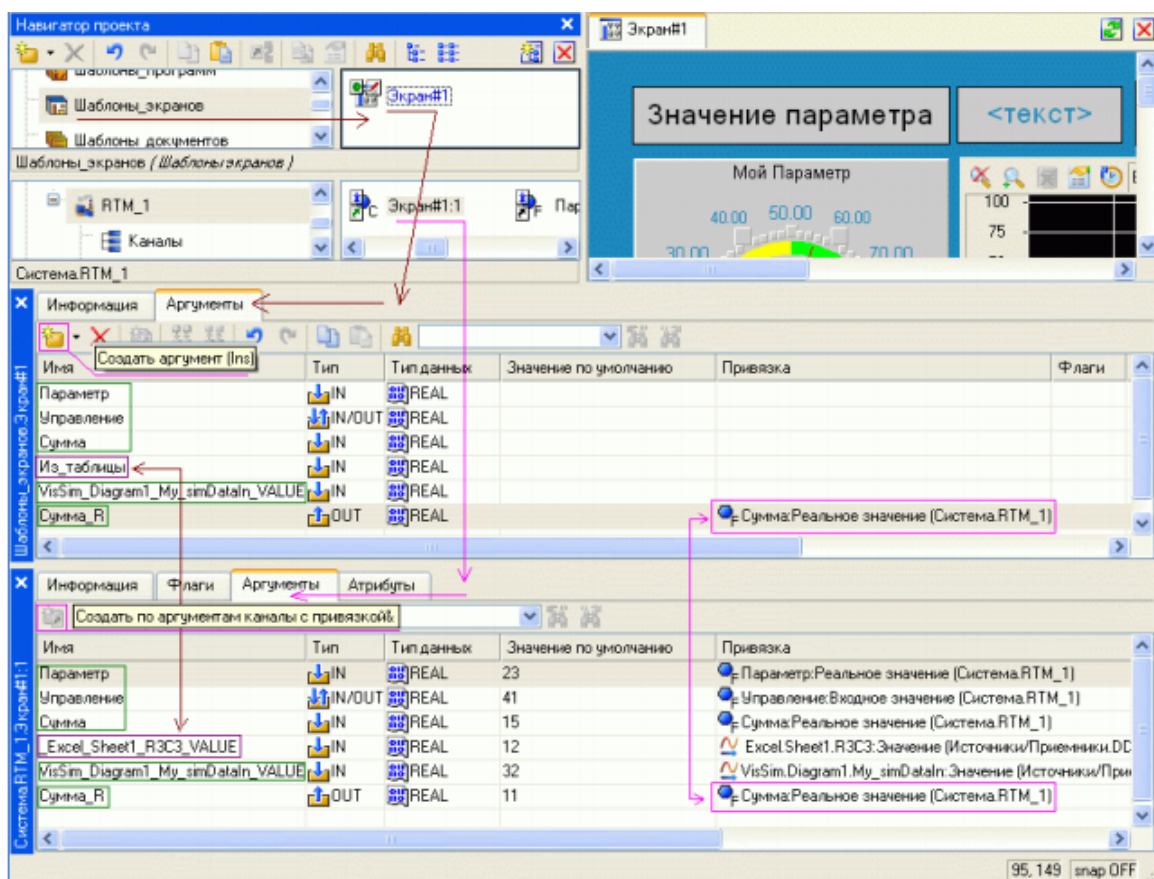


Рисунок 3 — Навигатора проекта Trace Mode.

Здесь на рисунке 3 вверху слева две копии шаблонов экранов, одна под другой. На верхней — отображен, открытый шаблон экрана Экран#1, на нижней — открыт канал вызова шаблона этого экрана (С) Экран#1:1. Справа вверху графический экран (мнемосхема), которая выводится только как шаблон, независимо от того, вызывается ли он щелчком по шаблону Экран#1 в Навигаторе или по каналу вызова Экран#1:1.

Переменные канала вызова (С) Экран#1:1 обязательно должны быть привязаны, иначе они не будут участвовать в информационном обмене.

Отметим, что в свойствах шаблона экрана (верхнее окно с таблицей) активная кнопка вверху слева позволяет создать аргумент (переменную) с нужным названием. В нижнем окне, Свойств Канала вызова экрана имеется очень похожая по внешнему виду кнопка, но она служит для автоматического создания каналов. Сейчас она не активна, поскольку все фактические переменные привязаны, но если, например, одну из привязок удалить, то кнопка станет активной и с ее помощью можно будет осуществить автоматическое построение канала. Конечно, и не пользуясь это кнопкой можно вручную привязать переменную к источнику (приемнику) значения или каналу.

Таким образом, в Свойствах шаблона экрана можно создавать новые переменные.

В свойствах канала вызова шаблона новые переменные создавать нельзя, но можно переименовывать переменные шаблона и осуществлять автоматическое построение каналов, т.е. создать в Навигаторе в том же поле, что и канал вызова экрана Экран#1:1, каналы с названиями фактических переменных, которые далее необходимо привязать вручную к источнику числовых значений.

Итак, при вызове мнемосхемы на экран компьютера в процессе работы Монитора реального времени, т.е. в процессе работы виртуальной, компьютерной части системы контроля и управления, канал вызова обращается к шаблону экрана, заменяет названия формальных переменных фактическими и обращается в соответствии с привязками к значениям величин источников или отправляет значения в приемники.

При автоматическом построении канала Trace Mode может заменить имя фактической переменной, совпадающей с именем формальной переменной на другое, несколько отличающееся, чтобы было легче различать что это, фактическая, относящаяся к шаблону экрана, или формальная, относящаяся к каналу вызова шаблона, переменная.

Окно свойств шаблона экрана с аргументами можно вызвать и из Свойств Графического элемента экрана, например, Текст, там ввести переменную и привязать ее. Из Свойств графического элемента вызывается только шаблон экрана и таблица его аргументов., в которой можно ввести новую переменную, формальную или фактическую, при этом по-

следнюю естественно, потребуется привязать к источнику или приемнику значения.

Более детальную информацию по созданию шаблона экрана можно просмотреть в справочной системе Trace Mode [1]. Дополнительно можно посмотреть примеры разработок проектов [2–7].

#### **4.5.3. Привязка элементов и переменных. Отладка связей**

Второй этап проектирования, который может перекрываться с первым, это установление виртуальных связей между источниками и приемниками сигналов посредством специальных переменных (т.н. аргументов). Осуществляется либо методом перетаскивания (drag-and-drop), что приводит к автоматическому созданию виртуальных каналов связи, либо вручную, с помощью табличных интерфейсов.

Привязка приемника числового значения может быть осуществлена непосредственно к источнику сигнала, а также, что делается чаще, посредством специальных виртуальных каналов связи.

Привязку нужно проводить очень аккуратно, именно здесь кроется источник возможных ошибок проекта. Поэтому на структурно-функциональной схеме системы управления, контроля и документирования должны быть в полной мере отображены все переменные, которые должны передаваться системой от источников к приемникам информации, например от выхода датчика работы насоса (Включен – выключен) к текстовому полю на мнемосхеме оператора.

На рисунке 4 показан результат автоматического создания канала (F) Text\_Value методом перетаскивания генератора пилы Пила#1 на канал (C) вызова шаблона экрана Экран#1:1. В результате посредством канала (F) передачи числового значения с плавающей точкой Text\_Value в канал Экран#1:1 вызова шаблона мнемосхемы группы виртуальных каналов Channel\_Gener\_To\_Text с помощью переменной Text\_Value происходит передача текущего значения сигнала генератора в текстовое поле Экрана (мнемосхемы).

В результате перетаскивания возникает канал (F) Text\_Value. Этот канал включен в цепь передачи значения сигнала генератора от виртуального генератора к текстовому полю мнемосхемы (приемнику этого значения). На мнемосхеме помещен всего один элемент, текстовый гра-

фический, в котором при запуске Монитора реального времени вместо надписи "Значение сигнала генератора" будет отображаться текущее численное значение сигнала виртуального генератора.

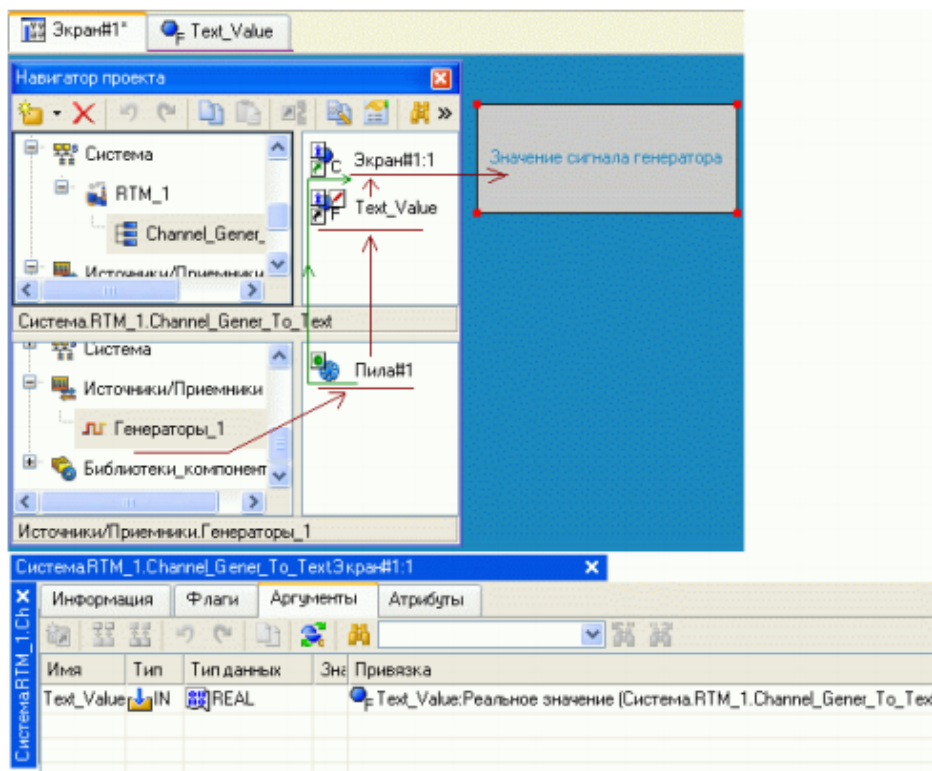


Рисунок 4 — Пример соединения источников и приемников проекта.

Пример отладки, т.е. поиска и исправления возможных ошибок в задании связей и привязок аргументов и переменных осуществляется в Trace Mode с помощью отладчика, запускаемого из панели инструментов Редактора проекта показан на рисунке 5.

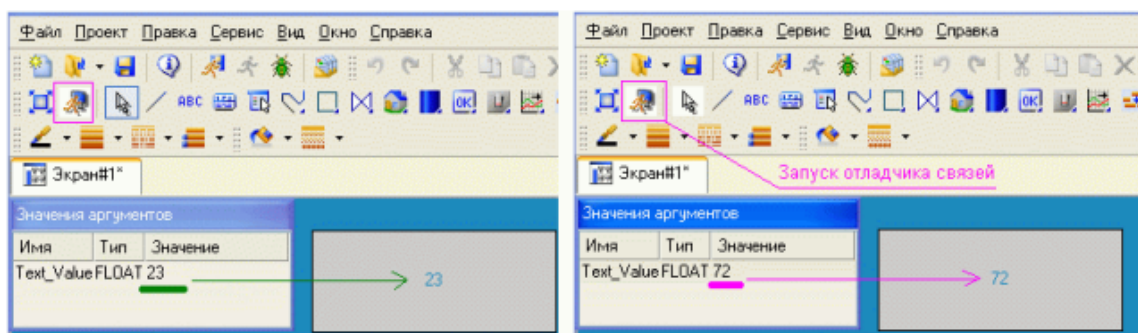


Рисунок 5 — Пример исправления возможных ошибок в задании связей и привязок аргументов с помощью отладчика связей



При запущенном отладчике связей проекта в его поле Значение можно вводить числа, с последующим нажатием клавиши Ввод на клавиатуре. При правильной привязке переменной в текстовом поле экрана мнемосхемы, где она должна отображаться, появляется то же самое число (см. рис. 5).

#### **4.5.4. Создание программ и каналов**

Другой важной частью SCADA-систем является аппарат анализа текущего состояния технологического процесса, прогнозирование его поведения, возможных аварийных ситуаций, которые нужно своевременно предотвратить и т.п. Такой аппарат опирается на программы на нескольких языках программирования.

Программы Trace Mode позволяют создавать имитаторы реальных объектов управления и тестировать SCADA-систему, без ее подключения к реальным объектам, что позволяет убедиться в ее работоспособности, или выявить и устранить ошибки проекта.

Эстетика мнемосхемы очень важна, но прежде всего проект должен быть состоятельным, работоспособным. Поэтому в методических указаниях, приводимые ниже мнемосхемы строятся с минимальным разнообразием Графических элементов и их настроек.

Основное внимание уделяется обеспечению правильности привязок переменных и правильности реализации алгоритмов при построении моделей объекта.

Программы в Trace Mode это инструменты анализа данных и построения имитаторов периферии (объектов управления). В Trace Mode проектировщик может воспользоваться шестью видами программ.

Каналы в Trace Mode это средства передачи значений переменных от источников к приемникам и первичной обработки данных.

##### ***4.5.4.1. Программа ST***

Программа ST представляет собой вид текста, в котором объявлены переменные программы, а также записаны операторы, осуществляющие действия с переменными и пояснения (рис. 6).

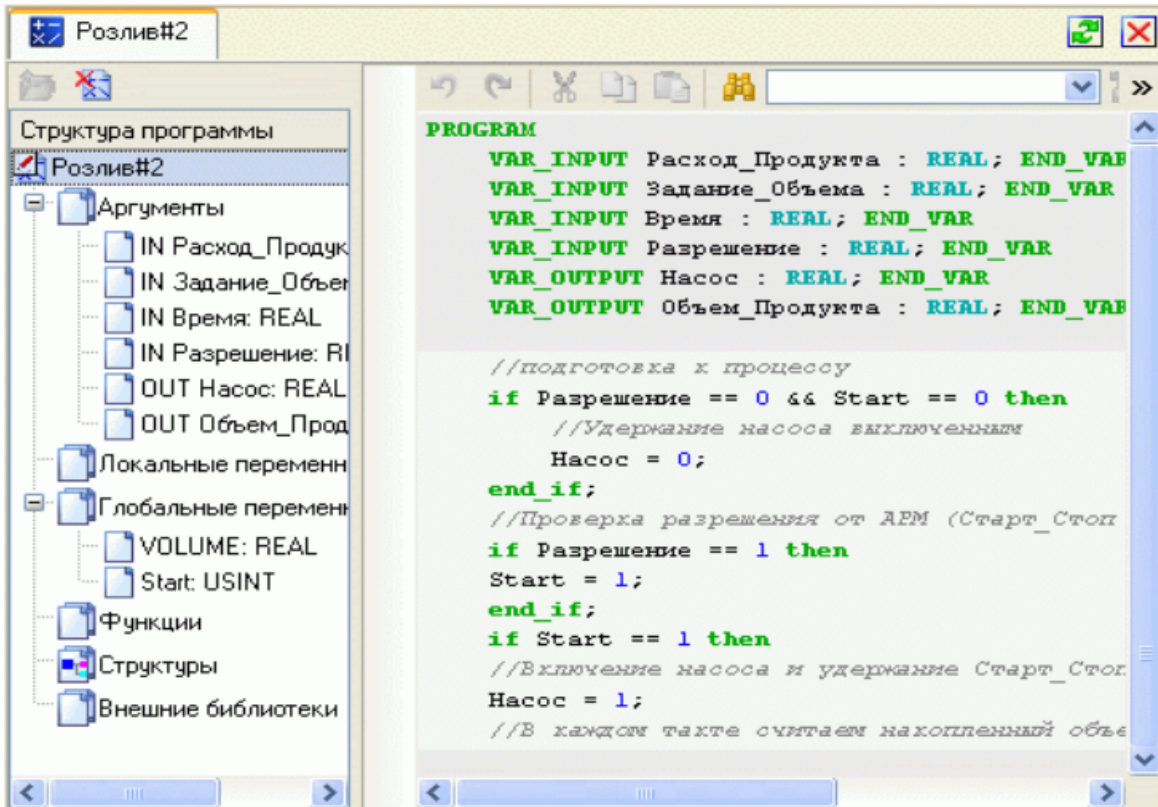


Рисунок 6 — Пример ST-программы Trace Mode.

Программа представляется в виде текста, содержащего операторы и пояснения

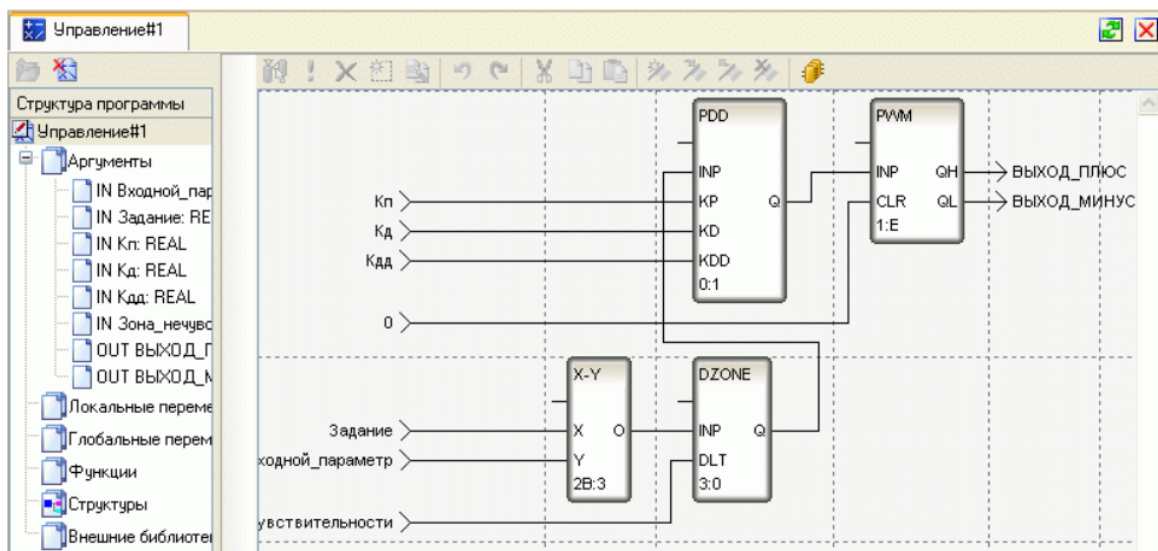


Рисунок 7 — Пример FBD - диаграммы (программы) Trace Mode, объектно-ориентированной модели.

Регулятор, реализующий закон управления ПДД. Математическая модель составляется из отдельных блоков библиотеки, аналогично построению структурных моделей, например, в программах Simulink, Fluidsim или LabView.

#### 4.5.4.2. Каналы и их свойства

Каналы в Trace Mode — это виртуальные средства передачи значений переменных от источников к приемникам и первичной обработки данных. Канал типа (C) – (Call) может вызывать Экран (мнемосхему) или программу и указывать привязку их переменных (аргументов), т.е. показывать, из какого источника нужно получить значение той или иной переменной, или к какому приемнику отправить значение переменной.

Ввод в проект (создание) канала: правой кнопкой по группе каналов, в выпавшем списке выбрать нужный канал (см. рис. 8). В правом окне Навигатора проекта показана группа каналов, с том числе класса (C) — вызова, а также класса (F) передачи значения в форме с плавающей запятой (см. рис. 9).

Каналы типа (F) (Float, с плавающей десятичной точкой) могут просто передавать значения переменных, каналы типа (H) (шестнадцатеричные) передают целочисленные значения.

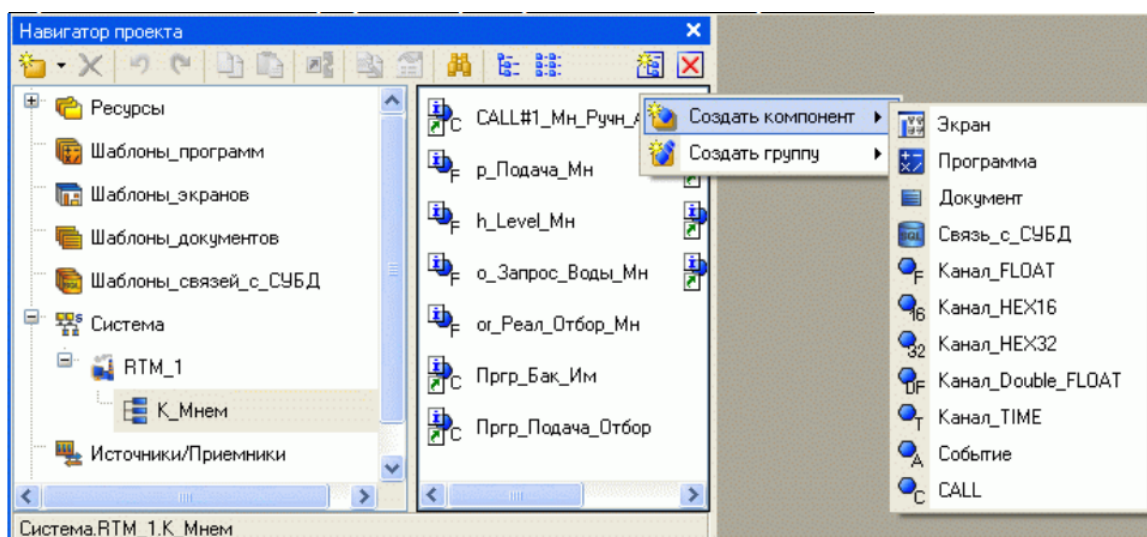


Рисунок 8 — Пример создания канала

Свойства канала типа (F) вызываются двойным щелчком по его названию в Навигаторе проекта. Окно свойств появляется в рабочей области как одна из вкладок. В окне свойств может быть введено название канала, задана предварительная обработка данных, например, задание ограничений на передаваемые значения (Границы), указан тип канала In или Out, введен комментарий и др.

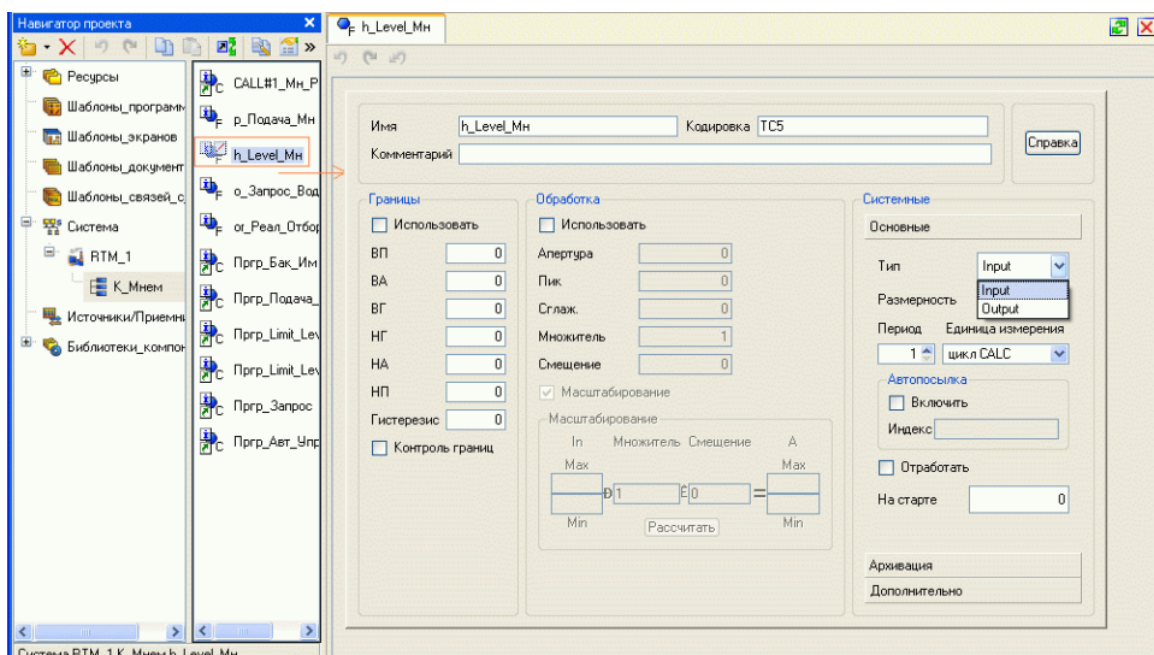


Рисунок 9 — Свойства канала типа (F)

Каналы могут быть однонаправленными: типа In подают в мнемосхему или программу значения, типа Out принимают из мнемосхемы или программы (или из других источников) значение переменных и куда-нибудь, если они привязаны, их отправляют. Каналы могут быть и двунаправленными, типа In/Out.

Указание источника или приемника значения переменной называется привязкой аргумента (переменной).

Созданный канал может быть перетаскиванием в Навигаторе проекта Trace Mode привязан к нужной мнемосхеме (шаблону экрана) или программе (шаблону программы).

Если проектировщиком создается канал вызова программы или экрана, то одновременно с каналом создается и соответствующий шаблон программы или экрана с тем же именем, которые могут быть изменены по желанию проектировщика.



Свойства шаблона экрана (программы) и свойства каналов их вызова отличаются (см. рис. 10).

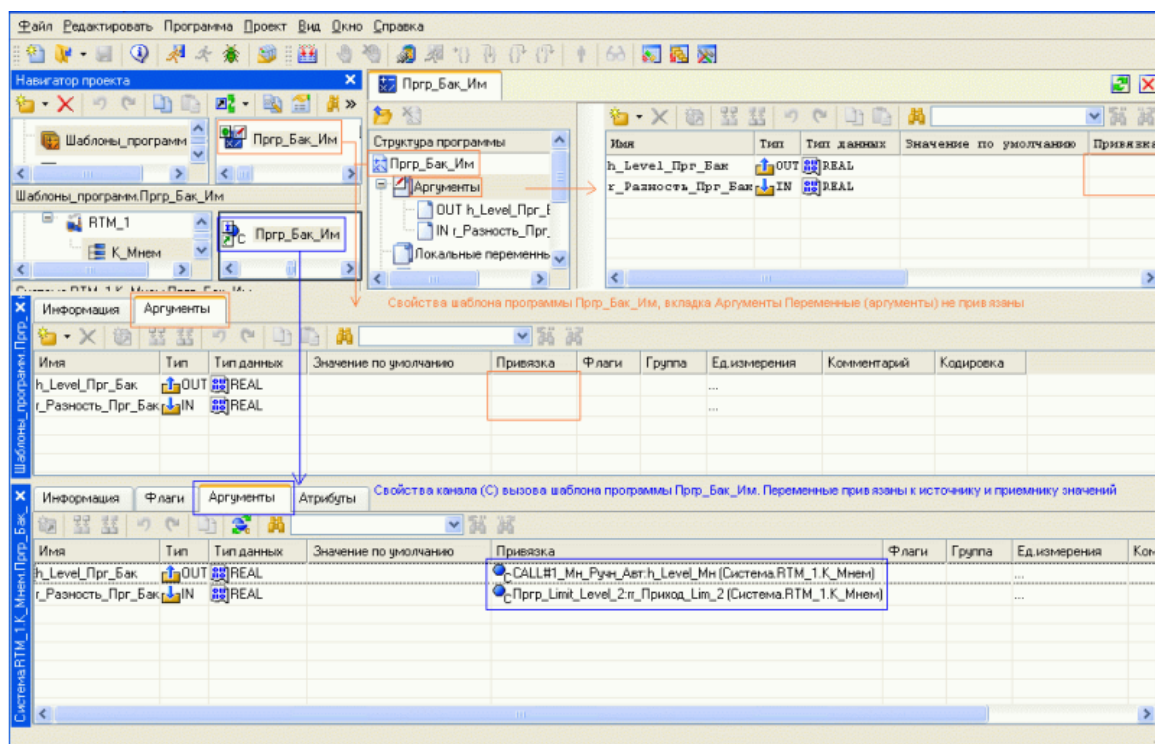


Рисунок 10 — Свойства шаблона экрана (программы) и свойства каналов

У шаблонов только две вкладки и их переменные могут быть не привязаны к источникам и приемникам. У каналов вызова шаблонов четыре вкладки и их переменные должны быть привязаны проектировщиком к источникам и приемникам, иначе они не будут иметь численных значений, т.е. не будут участвовать в процессе работы системы управления.

Переменные (аргументы) могут быть введены только в шаблонах программ (экранов), как и их тип. В окне Свойств каналов эти переменные только отображаются. Их имена в свойствах каналов могут быть изменены.

#### 4.5.5. Создание узлов проекта и базы каналов

После разработки шаблонов экранов и программ необходимо создать узлы проекта — АРМ и РС-based контроллера для которых в дальнейшем можно будет формировать базы каналов, используя механизм автопостроения.

Отметим, что каналы можно построить и вручную.

Для обмена с различными устройствами в TRACE MODE используются следующие механизмы:

- обмен по протоколам, встроенным в мониторы. К таким протоколам относятся:

- DCS – протоколы контроллеров/серий модулей LAGOON, ROBO, NuDAM-6000, I-7000, ADAM-4000, ADAM-5000/485, RIO-2000 и т.п. при связи по последовательному интерфейсу;

- MODBUS RTU при связи по последовательному интерфейсу;

- MODBUS TCP/IP при связи по сети;

- протоколы плат ввода/вывода (УСО), устанавливаемых в системные шины компьютеров/контроллеров;

- обмен через поставляемые драйверы;

- обмен через драйверы, разработанные и подключенные пользователем по правилам, описанным в разделе Разработка драйверов. Интерфейс TCOM.

Обмен через драйверы реализуется при связи как по стандартным интерфейсам (в том числе полевым шинам), так и при использовании дополнительных устройств, реализующих необходимые интерфейсы, – коммуникационных плат, преобразователей интерфейсов и т.п.

Для всех видов обмена в ИС предусмотрены соответствующие шаблоны каналов (см. Шаблоны каналов обмена [1]).

Для конфигурирования обмена с поддерживаемым оборудованием в TRACE MODE предусмотрены шаблоны числовых каналов, которые различаются подтипом и дополнением к подтипу (см. Подтипы каналов и Атрибуты каналов, отображаемые профайлером [1]).

В большинстве случаев шаблоны каналов обмена с платами/модулями создаются в навигаторе автоматически (см. Назначение групп источников (приемников)), при этом для них задается ряд параметров. Эти параметры могут быть изменены в соответствующих редакторах. Примерный вид одного из таких редакторов показан на рисунке (см. также Редакторы источников (приемников)).

Рисунок 11 — Пример окна настройки канала

Параметры аппаратных тегов задаются в соответствии с документацией на устройство, образуя удаленный адрес (см. Удаленный адрес и разновидности драйверов и Атрибуты каналов, отображаемые профайлером [1]).

Групповое редактирование параметров шаблонов каналов возможно в базовом редакторе (см. Базовый редактор группы компонентов [1]).

IP-адрес может задаваться в редакторе в нескольких форматах, например IP 192.168.77.1 .

Если в канале обмена по RS номер COM-порта задан равным 255, MPV использует для обмена первый COM-порт, который удалось инициализировать.

Узлы проекта создаются как корневые группы слоя Система. Предопределенное название узла указывает на семейство мониторов, для которых данный узел предназначен (см. Модификации мониторов [1]). Узел может содержать только те компоненты, которые поддерживаются мониторами соответствующего семейства.

В общем случае, узлы могут выполняться под управлением различных мониторов.

Как правило, узел выполняется на отдельном аппаратном средстве. В случае запуска двух и более узлов на одном аппаратном средстве оно должно быть оборудовано соответствующим количеством сетевых карт.

Параметры узлов задаются в соответствующем редакторе (см. Редактор параметров узла).

Узел RTM предназначен для запуска на компьютере под управлением исполнительных модулей семейства RTM (MPB) – мониторов с поддержкой отображения графических экранов оператора, поддержкой обмена по последовательному интерфейсу и сети с различным оборудованием и выполняющего пересчет каналов всех классов, кроме каналов T-FACTORY.

Монитор – многопоточный процесс. Приоритеты потоков заданы по умолчанию, однако их можно изменить на вкладке Дополнительно редактора узла (см. Задание параметров узла ) или в файле \*.cnf (см. Задание параметров работы мониторов ) [1].

Ниже указаны номера потоков и выполняемые в них задачи (см. также Принцип работы монитора. Канал TRACE MODE 6 [1]):

0 – CALC, основной поток, выполняемый монитором циклически. Один цикл включает следующие последовательно выполняемые этапы: последовательный анализ всех включенных каналов узла (по возрастанию ID) и установка флага SV (недоступен для пользователя) каналам, требующим пересчета (см. Период пересчета канала).

Каналы с типом пересчета цикл FAST и FAST EXE, цикл IDLE и IDLE EXE, цикл TF анализируются соответственно в потоке FAST, IDLE и TF.

Пересчет всех каналов (кроме каналов CALL) типа INPUT, которые должны пересчитываться в основном потоке, и, в некоторых случаях, — отработка этих каналов:

- если канал связан с УСО – установка каналу флага необходимости отработки Q (недоступен для пользователя), отработка (чтение данных) и пересчет (вычисление атрибутов канала — т.е. выполнение внутренних алгоритмов и трансляции). Сброс для канала флагов SV и Q;



- если канал принимает данные асинхронно (сеть, RS) — установка флага Q и пересчет. Сброс флага SV, установленный флаг Q является признаком необходимости отработки канала в соответствующем потоке;
- пересчет и отработка каналов класса CALL основного потока;
- пересчет каналов типа OUTPUT, которые должны пересчитываться в основном потоке, и анализ их выходного значения. Установка флага Q каналам, выходное значение которых изменилось. Дальнейшие действия выполняются по следующему алгоритму:
  - если канал связан с УСО — отработка (передача значения) и сброс флага Q;
  - если канал передает данные асинхронно (сеть, RS), он не обрабатывается и флаг Q не сбрасывается.

Не сброшенный в основном потоке флаг SV является признаком необходимости пересчета канала в соответствующем потоке.

Если используются приоритеты потоков по умолчанию, и для цикла установлено время (см. Время цикла монитора), недостаточное для выполнения всех его задач, система будет работоспособной (будет идти обмен по RS и сети, выполняться программы и т.п.), однако заданные временные характеристики пересчета/отработки каналов будут нарушаться (т.е. быстродействие системы снизится). Например, в такой ситуации при запуске узла под Windows CE может никогда не отработаться вызов графической панели.

#### **4.5.6. Имитация объекта регулирования. Интерфейс связи OPC**

Поскольку с помощью программ могут быть построены модели (имитаторы) объектов регулирования, а с помощью каналов они могут быть, как, впрочем, и напрямую, подключены к системе управления, то это позволяет строить, исследовать и отлаживать проекты SCADA-систем в безопасном для реального оборудования режиме.

##### ***4.5.6.1. Модели объектов управления и построение их имитаторов в Trace Mode***

Математическое, виртуально-аналитическое моделирование объектов и элементов системы управления, исследование моделей и их оптимизация осуществляются с целью обеспечения наилучшего выполне-

ния системой управления поставленных перед ней задач. На рисунке 12 показан пример реализации простой модели емкости для воды, построенной в программе FBD которая может быть использована как имитатор в виртуальной SCADA-системе.

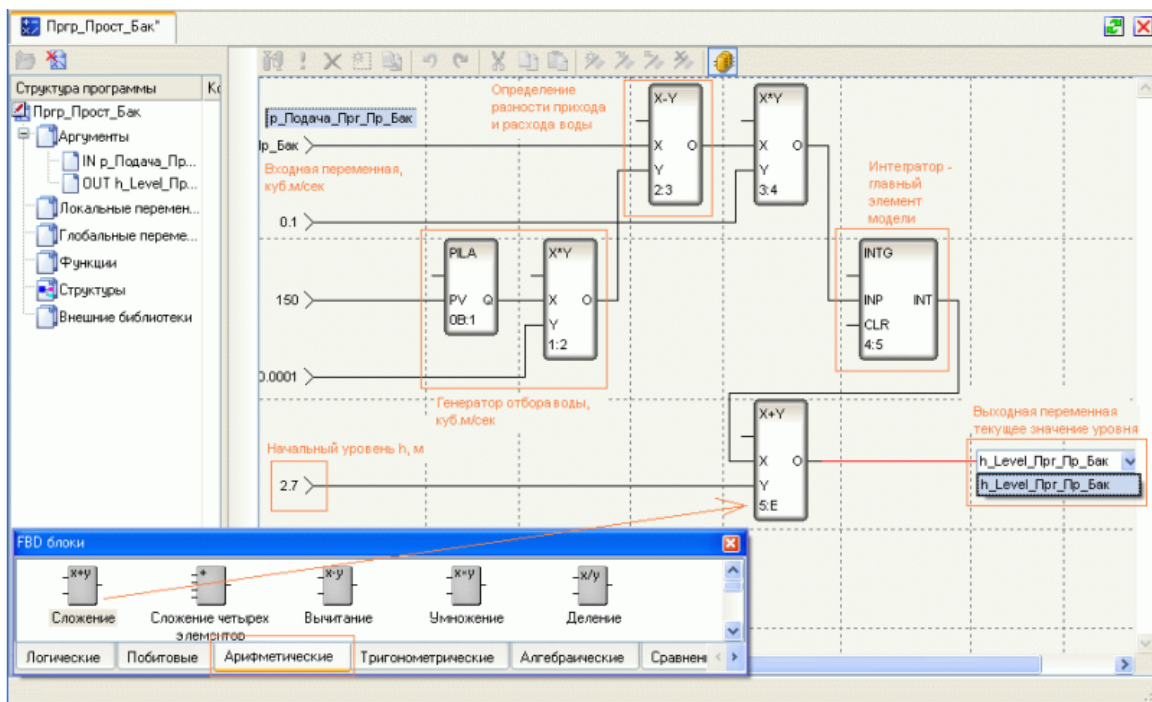


Рисунок 12 — Пример простой модели емкости для воды, построенной в программе FBD

Основной результат математического моделирования - структура локальных систем управления и перечень управляемых и контролируемых параметров, параметров управляющих элементов и их оптимальные значения.

#### 4.5.6.2. Модели объектов управления в других программах аналитического и имитационного моделирования

В Trace Mode наряду с собственным аналитическим аппаратом, в качестве инструментально-компьютерных средств можно использовать связи с другими программами имитационного моделирования такими как MatLab с приложением Simulink, или Fluidsim, или LabView — программ аналитического и виртуально-объектного моделирования.

Теория и практика моделирования, исследования и автоматизации рассматривалась в курсах идентификации объектов и систем и теории автоматического управления.

Пакет Matlab Simulink, который является библиотекой системы Matlab, включает в себя набор стандартных функциональных блоков для моделирования математических моделей различных объектов управления, логических схем, различных преобразований, источников сигналов и отображения выходной информации. Однако для установления связи с Трейс Мод потребуется произвести настройку OPC - серверов как один из вариантов интеграции используемых разнородных программ [10–16].

Стандарт OPC (OLE for Process Control) разрабатывался с целью унификации взаимодействия контроллеров с программным обеспечением (драйверами и SCADA-системами). Разработчики TRACE MODE наделили систему возможностью связи с УСО, контроллерами и приложениями посредством технологии OPC (OLE for Process Control). В рамках технологии OPC возможна разработка приложений (OPC Server), выступающих в роли имитационных моделей объектов управления.

Технология OPC предлагает стандарты для обмена технологическими данными, в которые заложены самые широкие возможности. Учитывая большой авторитет вовлеченных в данную деятельность фирм, можно ожидать, что технология OPC будет набирать силу. Это перспективная технология для интеграции разнородных систем. Хотя процесс становления еще далеко не завершён и есть много проблем, которые предстоит решить.

Суть OPC — предоставить разработчикам промышленных программ универсальный фиксированный интерфейс (то есть набор функций) обмена данными с любыми устройствами. В то же время разработчики устройств предоставляют программу, реализующую этот интерфейс (набор функций) [11, 14].

Для осуществления моделирования и программирования алгоритма управления была использована технология OPC, позволившая обмениваться TRACE MODE 6 и Simulink информацией друг с другом. На

рисунке 13 можно увидеть взаимодействие программных комплексных средств TRACE MODE 6 и Simulink.

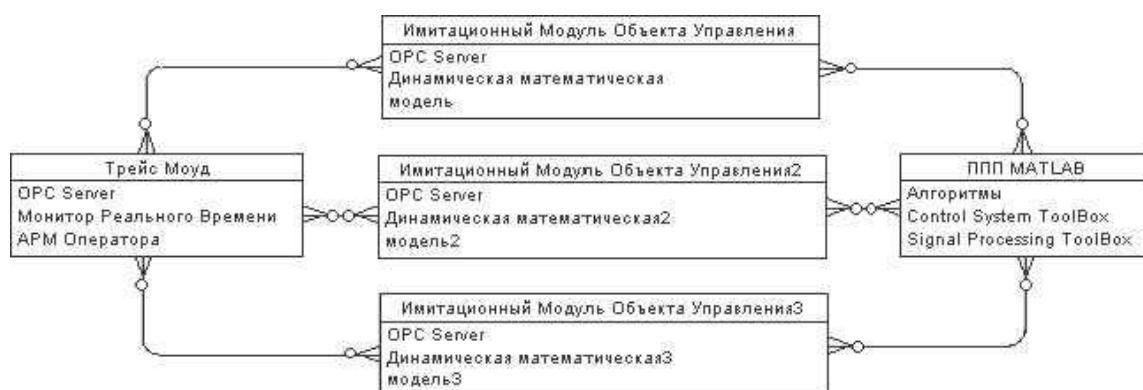


Рисунок 13 — Схема интеграции TRACE MODE и ППП MATLAB

Стандарт обмена данными OPC базируется на распространенной общепринятой схеме Клиент-Сервер. Эта архитектура позволяет подключить множество клиентов к одному серверу. И наоборот, данный стандарт позволяет использования одним клиентом различных OPC-серверов [10, 12].

Таким образом, возможно создание трехуровневой системы (рис.13) на базе SCADA системы TRACE MODE, ППП MATLAB, и приложений (OPC Server) — имитационных моделей объектов управления, функционирующей в рамках технологии COM (Component Object Model), отвечающей стандарту OPC, и операционной системы Microsoft Windows NT/2000/XP.

#### 4.5.7.3. *Настройка OPC — сервера TRACE MODE 6*

Для доступа OPC-клиента к каналам узла узел должен исполняться под управлением специализированного монитора – OPC-сервера TRACE MODE 6.

4.5.7.3.1. *Регистрация OPC-сервера TRACE MODE 6.* Для регистрации OPC-сервера TRACE MODE 6 в ОС нужно выполнить следующие процедуры (указанные ниже файлы находятся в директории установки монитора):

запустить командный файл register\_opc\_proxy\_stub.cmd (однократная процедура, после которой сервер можно регистриро-

вать/разрегистрировать). Для отмены нужно запустить командный файл `unregister_opc_proxy_stub.cmd`;

– запустить командный файл `register_server.cmd` (для отмены регистрации нужно запустить файл `unregister_server.cmd`);

– выполнить следующую команду:

```
%ОПС-сервер%/tm6opc das.exe %узел%/*.dbb
```

При выполнении этой команды сервер запускается.

Запуск ОПС-сервера TRACE MODE 6

Если ОПС-сервер зарегистрирован в ОС, для его запуска вручную достаточно выполнить следующую команду:

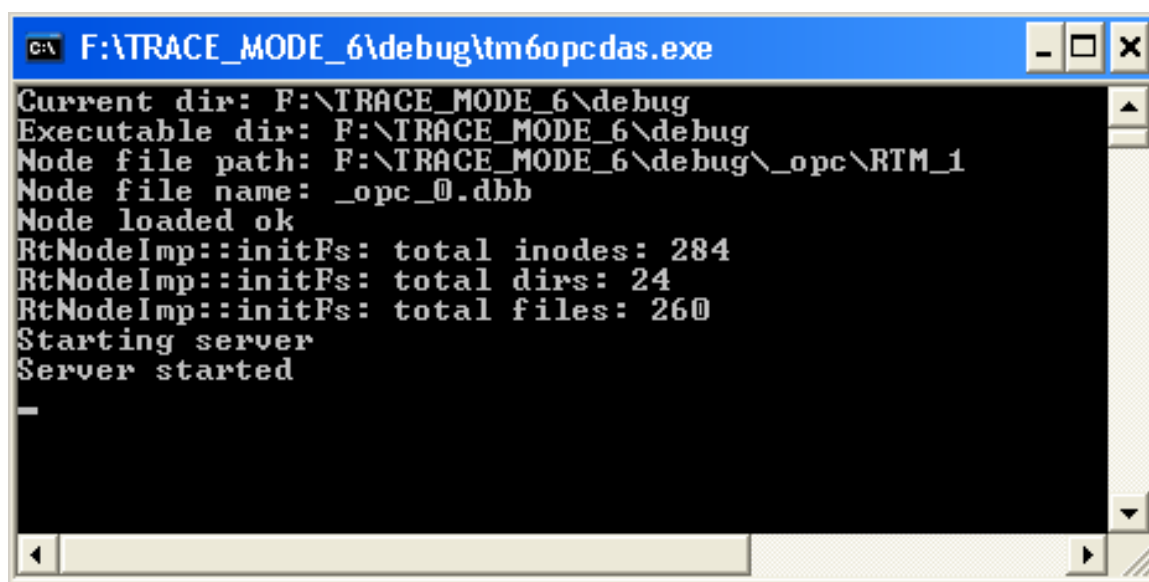
```
%ОПС-сервер%/tm6opc das.exe
```

При запросе клиента ОПС-сервер запускается автоматически.

ОПС-сервер TRACE MODE 6 поддерживает спецификацию OPCDA 2.0 – это следует учитывать, если в ИС создается узел – клиент ОПС (см. Вкладка ‘Базовый редактор’ [1]).

#### 4.5.7.3.2. Интерфейс оператора ОПС-сервера TRACE MODE 6

Интерфейс оператора ОПС-сервера TRACE MODE 6 включает консоль служебных сообщений сервера и графическую оболочку (см. рис. 14 и рис. 15).



```
C:\> F:\TRACE_MODE_6\debug\tm6opc das.exe
Current dir: F:\TRACE_MODE_6\debug
Executable dir: F:\TRACE_MODE_6\debug
Node file path: F:\TRACE_MODE_6\debug\_opc\RTM_1
Node file name: _opc_0.dbb
Node loaded ok
RtNodeImp::initFs: total inodes: 284
RtNodeImp::initFs: total dirs: 24
RtNodeImp::initFs: total files: 260
Starting server
Server started
```

Рисунок 14 — Запуск консоли служебных сообщений

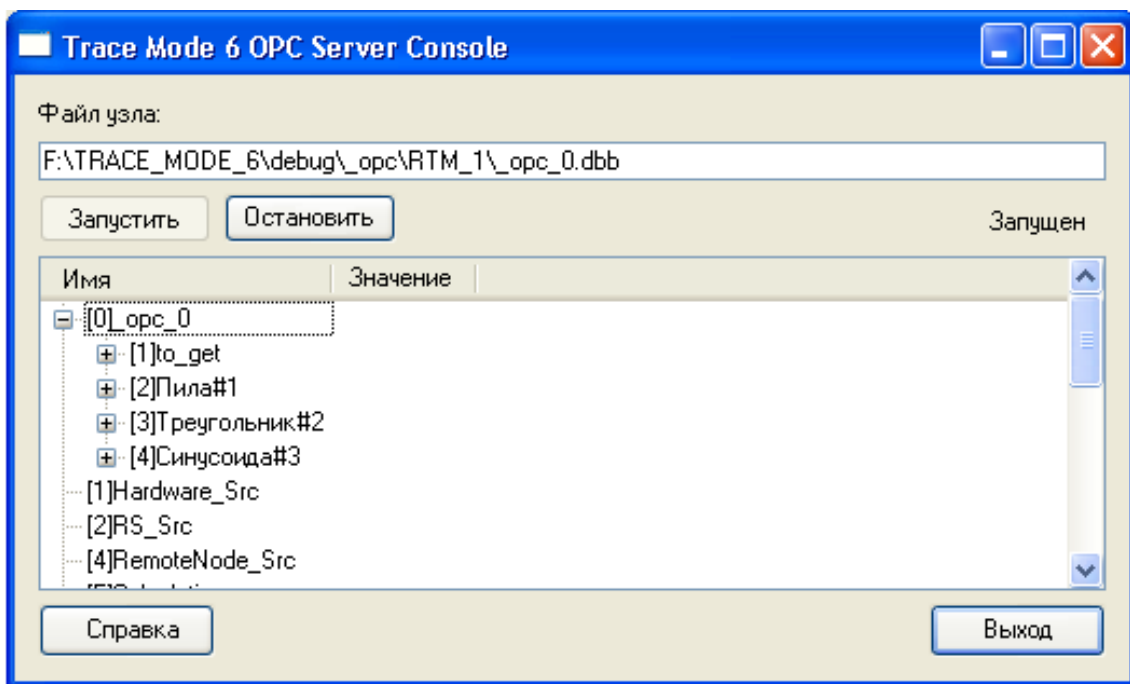


Рисунок 15 — Графическая оболочка OPC-сервера

Аналогично оболочке профайлера без поддержки графических экранов (см. Профайлер без поддержки графических экранов, [1]), оболочка OPC-сервера отображает каналы узла.

В поле Файл узла отображается полный путь к файлу \*.dbb узла.

Кнопки Запустить и Остановить предназначены соответственно для запуска и останова сервера. Информация о состоянии сервера отображается в строке этих кнопок справа.

При нажатии кнопки Выход сервер останавливается и выгружается.

*4.5.7.3.3. Настройка безопасности ОС.* Если необходим удаленный доступ к OPC-серверу TRACE MODE 6, то на компьютере, где сервер должен быть запущен, может потребоваться конфигурирование параметров ряда служб (в том числе DCOM, брандмауэра и самого сервера).

Подробнее о настройке OPC-сервера TRACE MODE 6 см.[16].

*4.5.7.3.4. Библиотека OPC Toolbox Matlab Simulink* предоставляет набор интерфейсов для работы с OPC-серверами. OPC Toolbox позволяет считывать и записывать данные в OPC-сервер, а также предоставляет возможность анализа качества входных данных [12–13].

В качестве примера используем Matlab Simulink для создания модели технологического объекта в виде передаточной функции апериодического звена первого порядка с запаздыванием. Для реализации связи TRACE MODE + InTouch используем чтение и запись для TRACE MODE OPCDA 2.0. Для получения переходной характеристики в реальном времени используем режим симуляции.

Создаем новый файл модели.

Выбираем в меню File -> New -> Model.

Для реализации обмена данными MatLab <-> TRACE MODE, используем OPC Toolbox пакета Simulink (рис. 16).

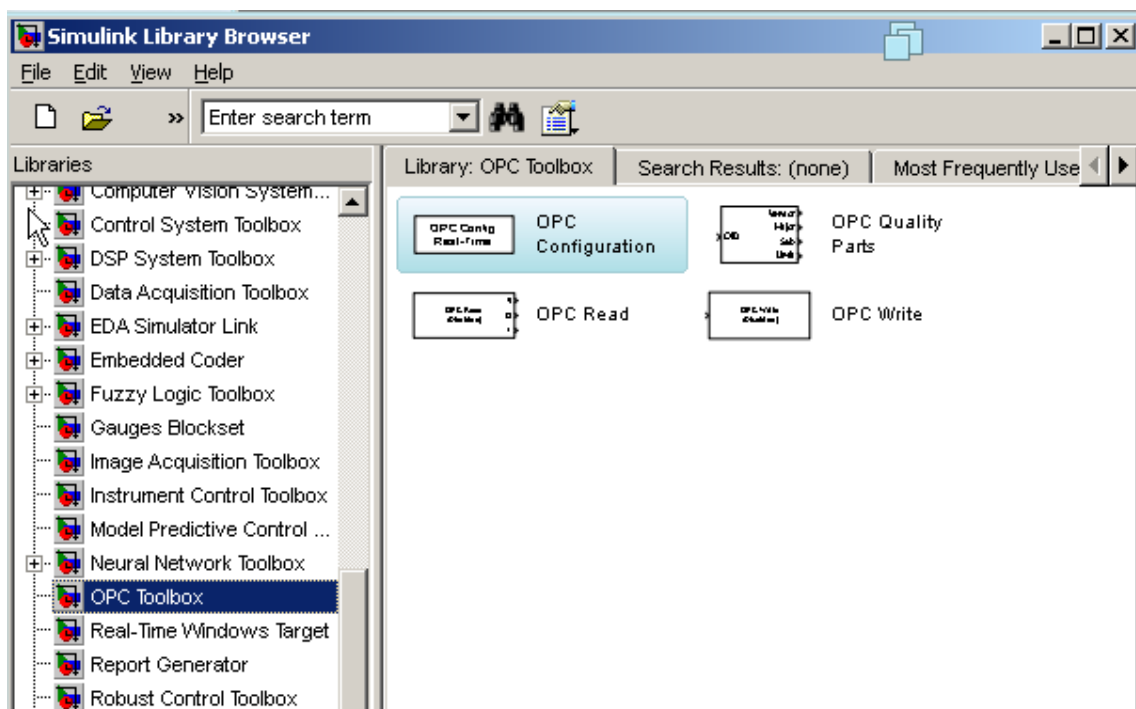


Рисунок 16 — Вид библиотеки OPC Toolbox пакета Simulink

Перетаскиваем на область модели (рис. 17) блоки OPC Configuration, OPC Read и OPC Write (для OPC Read отключаем выходы Q и T).

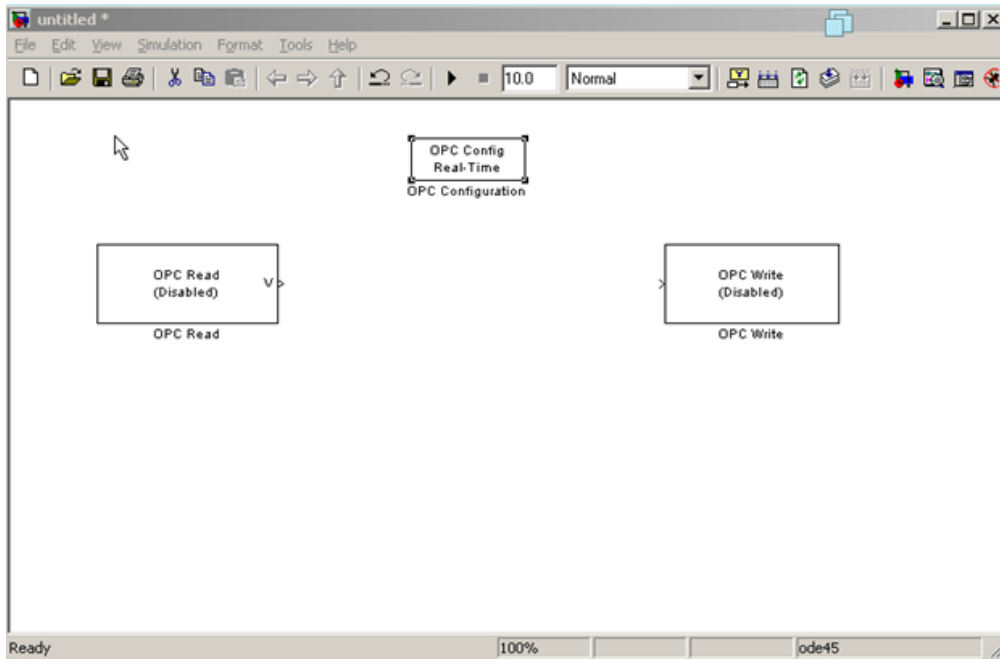


Рисунок 17 — Вид окна будущей модели

Далее настраиваем блок OPC Config Real-Time (рис. 18). Нажимаем кнопку Configure OPC Clients...

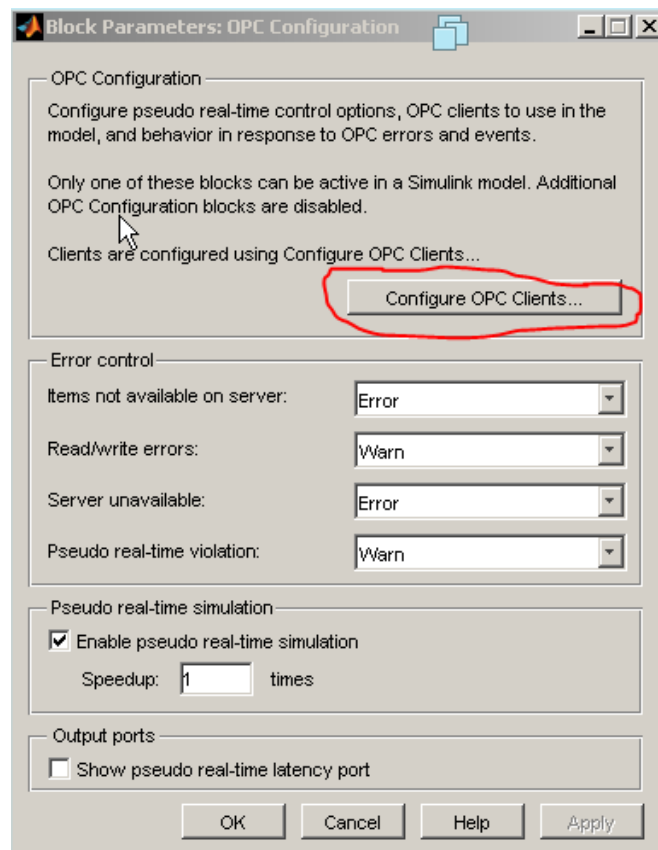


Рисунок 18 — Настройка блока OPC Config Real-Time



В появившемся окне нажимаем Add (рис. 19).

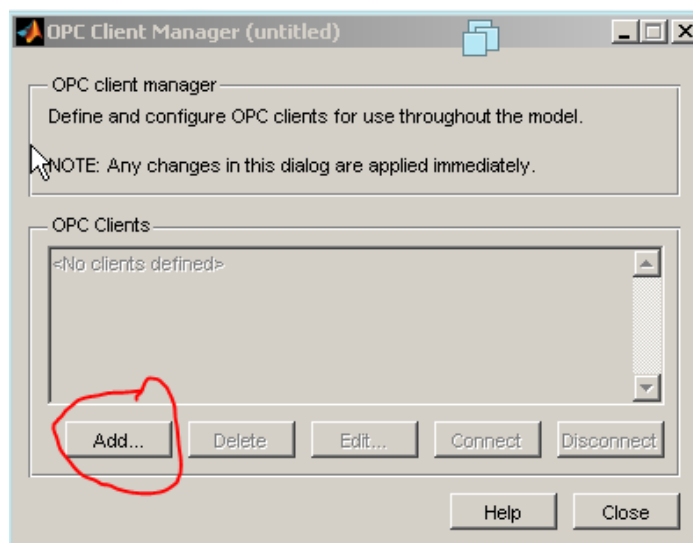


Рисунок 19 — Добавление OPC – клиентов

Далее кнопкой Select просматриваем список зарегистрированных OPC серверов (рис. 20). Выбираем TRACE MODE OPCDA 2.0 и нажимаем 2 раза Ok, Close, Ok.

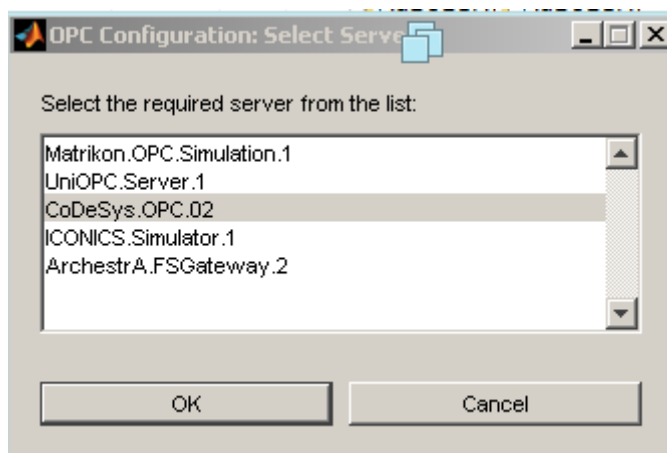


Рисунок 20 — Список зарегистрированных OPC серверов

Далее заходим в настройки блока OPC Read (рис. 21). Добавляем теги, кнопка Add. В поле Enter Item ID(s) необходимо прописать полный путь к переменной TRACE MODE. В нашем случае путь состоит из имени контроллера, названии программы и названия внутренней переменной TRACE MODE.

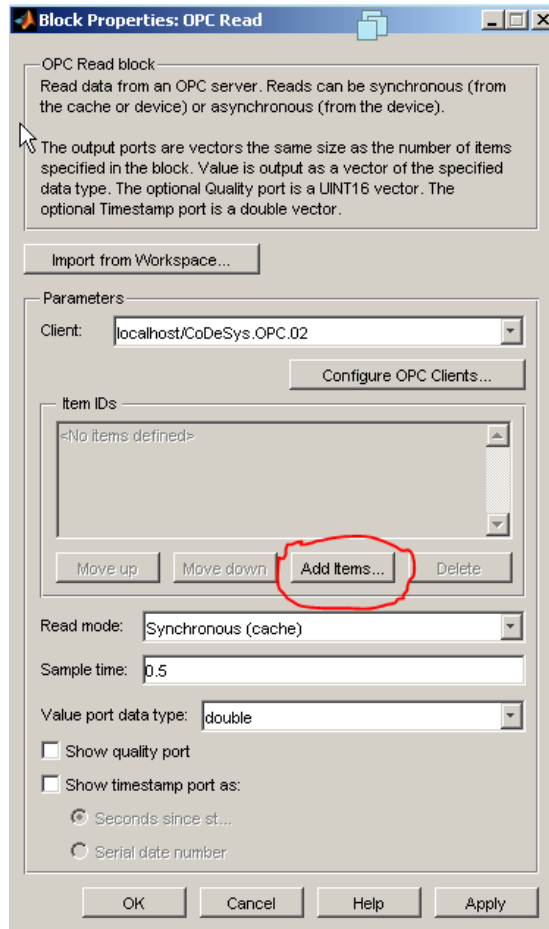


Рисунок 21 — Настройки блока OPC Read

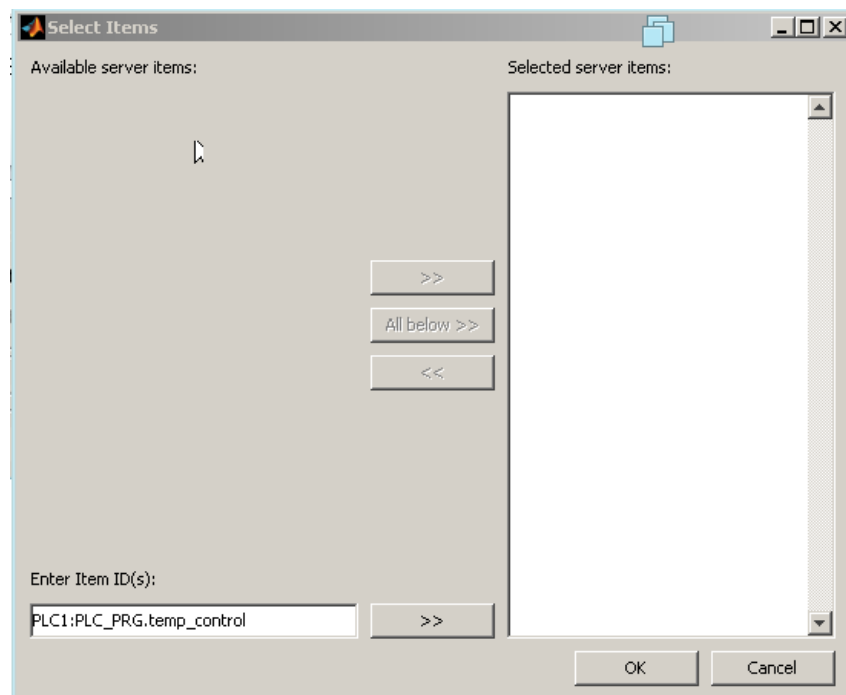


Рисунок 22 — Установка параметра Enter Item ID(s)

Нажимаем на стрелки, чтобы перенести переменную и нажимаем Ok. Обязательно не забываем нажимать Apply и Ok в окне настройки OPC Read и видим отображение имени соответствующего контроллера — названия внутренней переменной TRACE MODE (рис. 23).

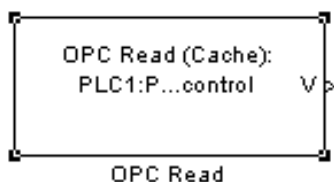


Рисунок 23 — Вид блока OPC Read после настройки

Важно не перепутать вход с выходом. В нашем случае входом модели является выход ПИД-регулятора, а выход модели подается на вход регулятора.

Настройка блока записи аналогична блоку чтения (рис. 24).

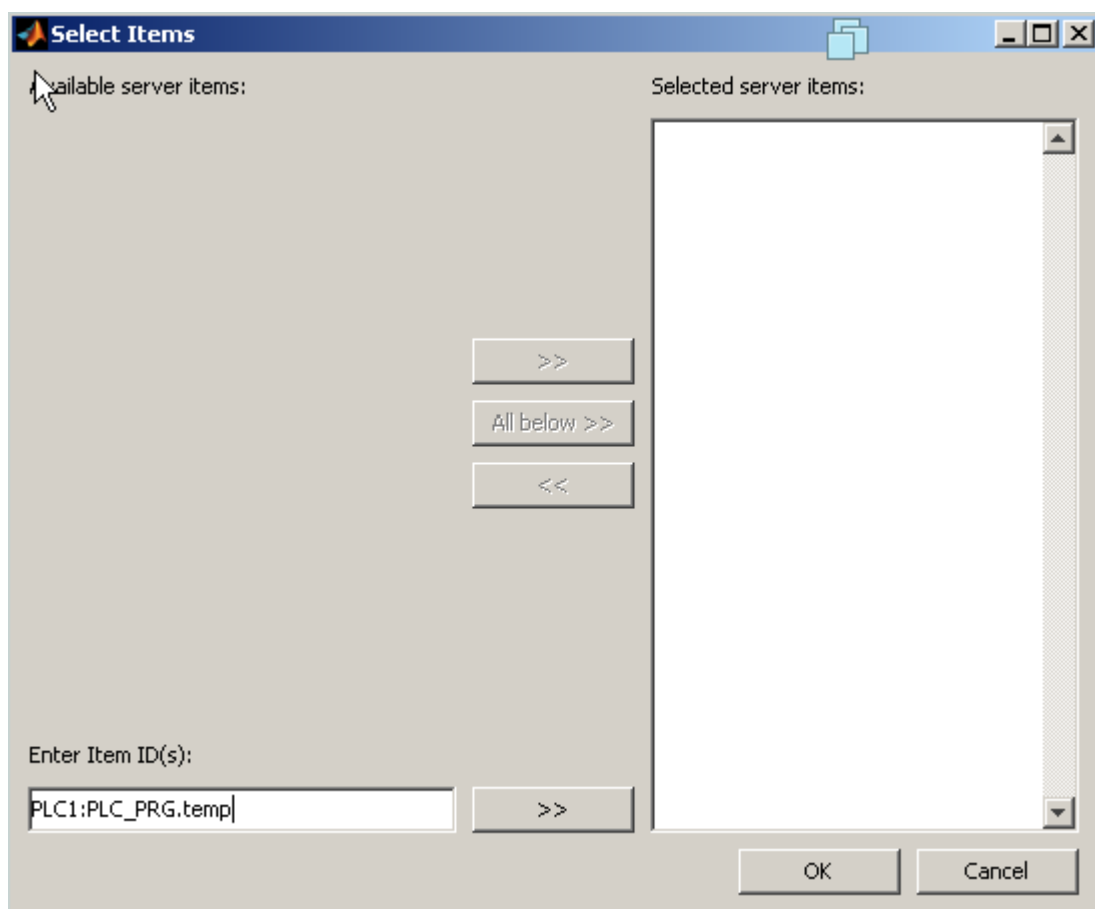


Рисунок 24 — Установка параметра Enter Item ID(s) блока записи

Теперь создадим имитацию теплового объекта. В разделе Simulink выбираем пункт Continuous и перетаскиваем в область модели блоки Transfer Fcn и Transport Delay. Соединяем стрелками блоки (рис. 25).

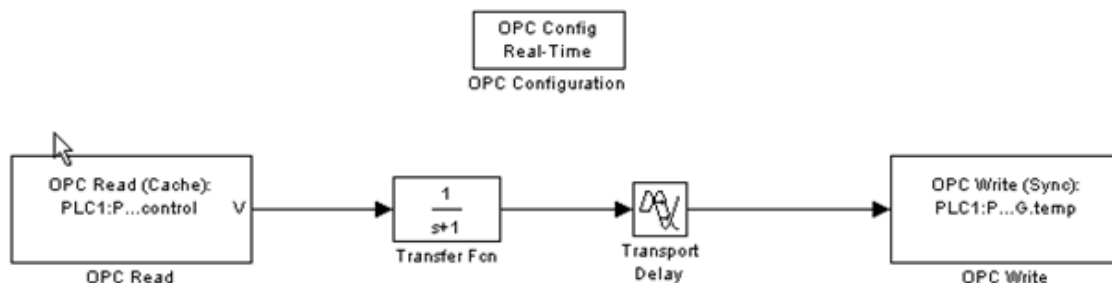


Рисунок 25 — Имитационная модель технологического объекта с настроенными блоками OPC Toolbox

Зададим произвольные параметры для нашего объекта. Если необходимо достичь определённого качества регулирования — необходимо будет произвести настройку регулятора – рассчитать параметры К и Т. В нашем случае это не принципиально.

**ВНИМАНИЕ!** Перед началом симуляции объекта необходимо убедиться, что контроллер, который работает с моделью объекта, запущен (Online -> Login), иначе код обработки чтения/записи вылетит по exception-у и работа Matlab Simulink будет прекращена без сохранения данных.

В параметрах настройки моделирования выставляем время симуляции бесконечность inf и запускаем модель (рис. 26).

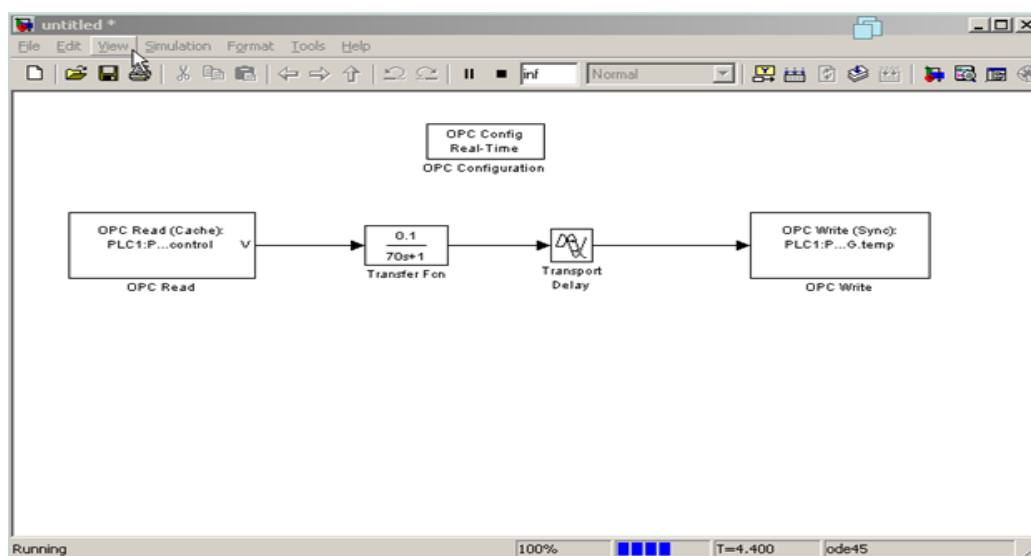


Рисунок 26 — Вид окна модели в процессе симуляции

Если добавить Scope (Simulink -> Sinks -> Scope) (рис. 27), то можно будет увидеть, как изменяется регулируемый параметр (рис. 28).

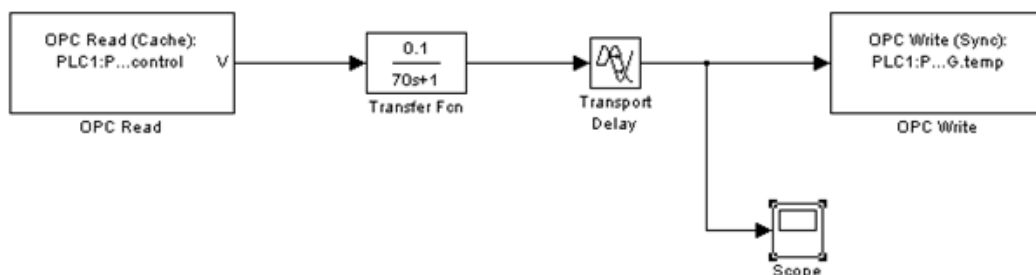


Рисунок 27 — Имитационная модель технологического объекта с блоком Scope

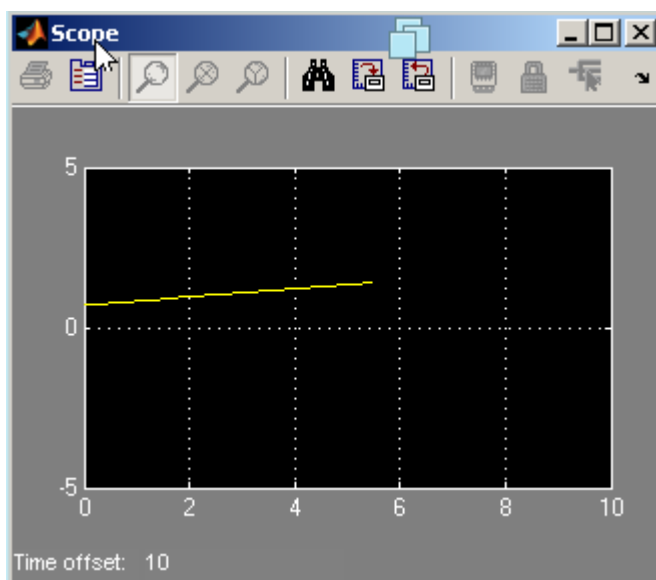


Рисунок 28 — Осциллограмма переходного процесса технологического объекта в Matlab Simulink .

#### 4.5.7. Имитация проекта

Для отладки проекта ИС снабжена следующими механизмами:

- автономная отладка шаблонов — см. разделы, посвященные разработке шаблонов;

- запуск выделенного узла (слоя Технология) под управлением одного из отладочных мониторов (профайлеров) из интегрированной среды по команде Отладка (см. Меню ‘Файл’ и главная панель инстру-

ментов ИС и Задание общих настроек ИС) с протоколированием работы в текстовый файл;

– использование функции шпион — эта функция обеспечивает получение в режиме редактирования реальных данных с работающих узлов проекта. Для ее использования нужно выполнить команду «Шпион» (рис. 20), при этом в ИС открывается одноименное окно:

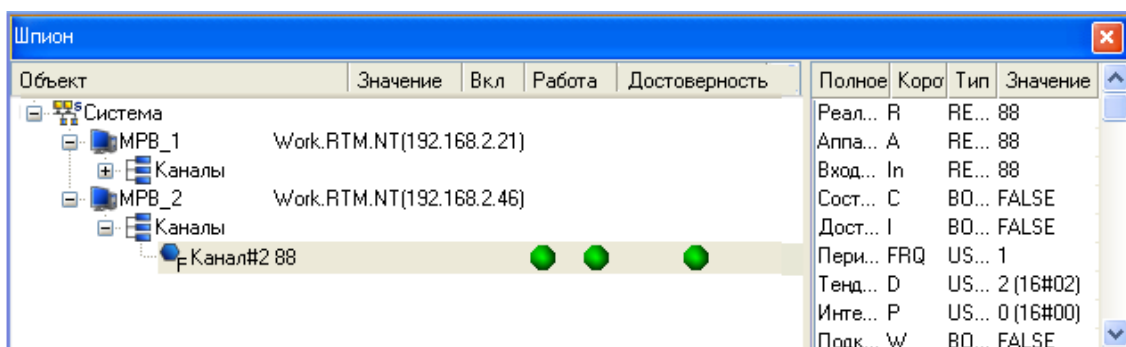


Рисунок 29 — Окно «Шпиона»

Функция шпион конфигурируется в диалоге Настройки (см. Вкладка ‘Интегрированная среда разработки’). Данная функция не работает при связи с узлом по RS.

Для отладки узла (слоя Технология) его можно запустить (в том числе из ИС) под управлением одного из следующих отладочных мониторов:

- профайлера с поддержкой графических экранов (rtc.exe);
- профайлера без поддержки графических экранов (rtmg32.exe).

При конфигурировании ИС можно указать профайлер, который должен запускаться из ИС (см. Задание общих настроек ИС [1]). Пример работы профайлера приведен на рисунке 30.

Профайлеры записывают протокол своей работы в файл <имя файла ргj>\_<порядковый номер узла>.txt, который сохраняется в папке узла. Степень детализации отладочной информации, выводимой в файл, может быть задана. Кроме того, все мониторы записывают информацию о своей работе в файл tm6\_log.txt (см. Файл tm6\_log.txt).

**ВНИМАНИЕ!** В списке профайлера каналы отображаются по возрастанию ID, а аргументы канала – по возрастанию порядкового номера.

Более детальную информацию по отладке проекта можно посмотреть в справочной системе TRACE MODE [1].

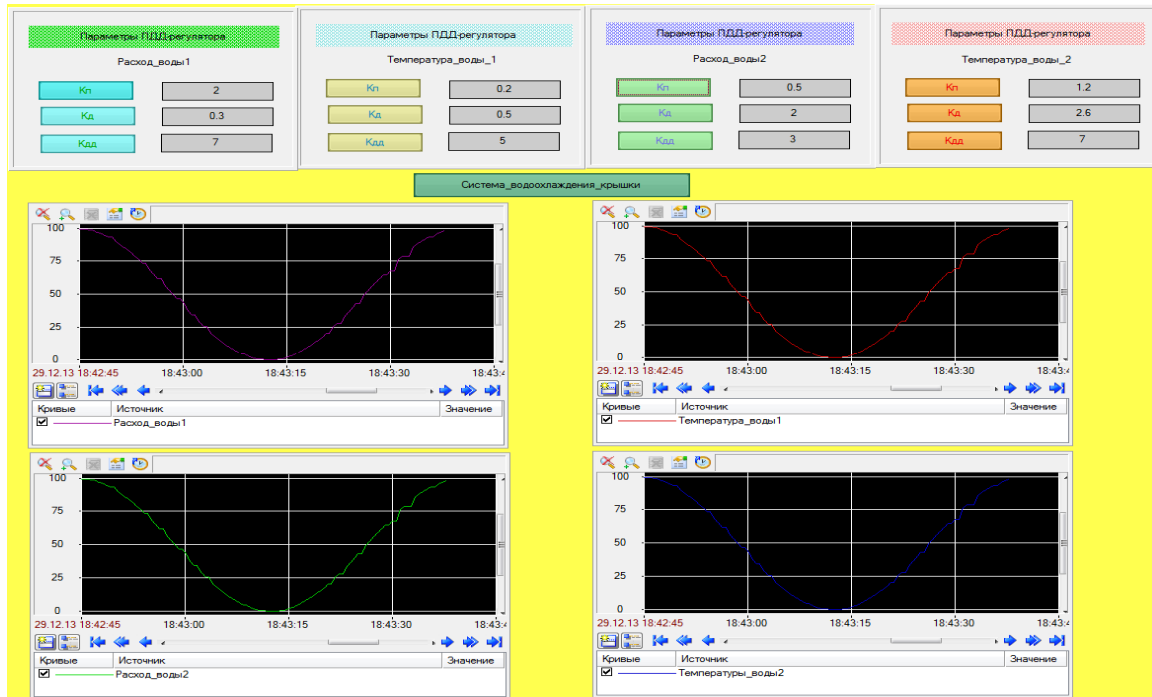


Рисунок 30 — Вид экрана АРМ «Параметры ПДД-регуляторов» через профайлер

## 4.6. Заключение

Заключение должно содержать краткие выводы по разделам основной части проекта, предложения по использованию выполненных разработок.

## 4.7. Перечень ссылок

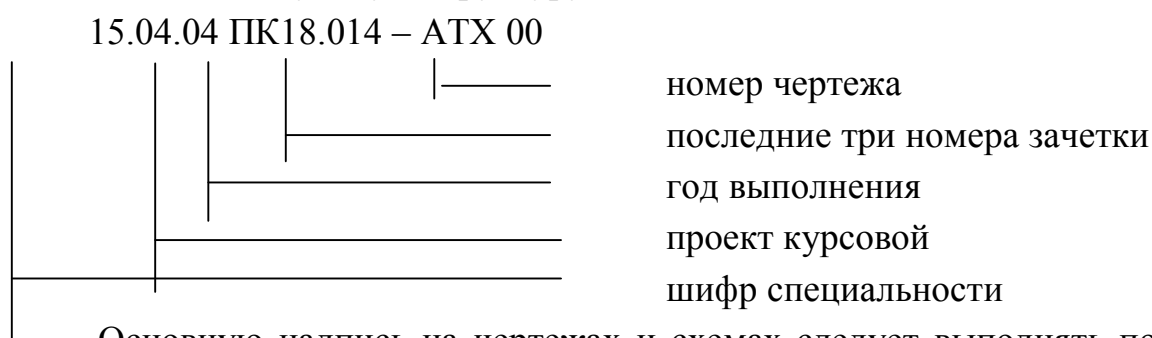
Перечень помещают непосредственно после основного текста ПЗ. Он должен содержать перечень источников, использованных при выполнении работы. Источники следует располагать в порядке появления ссылок в тексте. Сведения об источниках, включенных в перечень, необходимо давать в соответствии с [17].

## 5. ПРАВИЛА ОФОРМЛЕНИЯ КУРСОВОГО ПРОЕКТА

Пояснительная записка должна быть написана разборчивым почерком или выполнена машинописным (машинным) способом на одной стороне листа белой бумаги формата А4 (210 х 297 мм) через 6,5 мм с высотой букв и цифр не менее 1,8 мм. Необходимо соблюдать следующие размеры полей: верхнее, левое и нижнее не менее 20 мм, правое – не менее 10 мм. ПЗ должна быть оформлена в соответствии с требованиями [18].

Чертежи и схемы графической части проекта выполнены на двух листах белой бумаги формата А 3 (420 х 297 мм). Чертежи выполняют в соответствии с требованиями стандартов ЕСКД.

Обозначения документов, входящих в состав курсового проекта, должен иметь следующую структуру:



Основную надпись на чертежах и схемах следует выполнять по форме 1 и 1а [19].

Основную надпись заполняют с учетом следующих особенностей:

- в графе 2 обозначают документ в соответствии с приведенной выше структурой;
- в графе 4 указывают литеру У (учебный);
- в графе 9 указывают наименование ВУЗа (ДонГТУ) и шифр группы ( в две строки).



## ПЕРЕЧЕНЬ ССЫЛОК

1. Быстрый старт 1. Справка Trace Mode 6.
2. Релиз SCADA TRACE MODE 6.06.2. — Режим доступа: [http://www.adastra.ru/news/6062\\_scada/](http://www.adastra.ru/news/6062_scada/)
3. Пьявченко Т.А., Финаев В.И. Автоматизированные информационно-управляющие системы. — Таганрог : Изд-во ТРТУ, 2007. — 271 с. Режим доступа: <http://window.edu.ru/resource/205/61205/files/PiavchenkoT.pdf>
4. Шифрин Б.М. Пример разработки проекта системы водоотлива в Trace Mode 6. — СПб. : Лесотехническая академия, 2012. — Режим доступа: [http://shifrinb.ucoz.ru/load/lta/pas/primer\\_razrabotki\\_proekta\\_sistemy\\_vodootliva\\_v\\_trace\\_mode\\_6/21-1-0-40](http://shifrinb.ucoz.ru/load/lta/pas/primer_razrabotki_proekta_sistemy_vodootliva_v_trace_mode_6/21-1-0-40). Документ Word. 1.53 МБ. 2012.
5. Шифрин Б.М. Проектирование автоматизированных систем. — СПб. : Лесотехническая академия, 2012. — Режим доступа: <http://shifrinb.ucoz.ru/load/lta/pas/21>
6. Сундуков Е. Ю. Интегрированные системы проектирования и управления [Электронный ресурс] : сб. описаний лаб. работ для студентов направления подготовки дипломированных специалистов: 220200.65 «Автоматизация и управление» специальности 220301 «Автоматизация технологических процессов и производств» и направления подготовки бакалавров техники и технологии: 220200 «Автоматизация и управление» : самост. учеб. электрон. изд. / Сыкт. лесн. ин-т. — Электрон. дан. (1 файл в формате pdf : 4 Мб). — Сыктывкар : СЛИ, 2009. — Режим доступа: <http://lib.sfi.komi.com>. (дата обращения: 01.04.2013)
7. Кабанов А. А. Программирование в среде ТРЕЙС МОУД : Методические указания. — Тамбов : Изд-во Тамб. гос. техн. ун-та, 2006. — 32 с.
8. Dr. Bob Davidov. Компьютерные технологии управления в технических системах // Портал научно-практических публикаций [Электронный ресурс]. URL: <http://portalnp.ru/author/bobdavidov> (дата обращения: 25.10.2017)
9. Яковлев О.Г. Программирование ПЛК. — М., 2000–2001 гг. — 89 с.
10. Применение стандарта OPC. — Электронный ресурс. URL: <http://www.pikprogress.ru/opc.php>. (дата обращения: 14.04.2013)

11. Бергер Г. Автоматизация посредством STEP 7 с использованием STL и SCL и программируемых контроллеров SIMATIC S7-300/400. Siemens AG. — Нюрнберг, 2001. — 654 с.
12. Dr. Bob Davidov. Simulink OPC DA клиент. OPC Toolbox // Портал научно-практических публикаций [Электронный ресурс]. URL: <http://portalnp.ru/2013/09/1031> (дата обращения: 25.10.2017)
13. OPC Toolbox в Matlab Simulink // [Электронный ресурс]. URL: <http://autoworks.com.ua/programmirovanie-kontrollerov/matlab-simulink-opc-toolbox/> (дата обращения: 25.10.2017)
14. Martin Stianko. Технология OPC — Основные принципы и преимущества// [Электронный ресурс]. URL: <http://shtrm.ru/local/myopc/stat/2.htm> (дата обращения: 25.10.2017)
15. Связь Matlab и Simatic S7 по OPC // [Электронный ресурс]. URL : <https://simaticby.wordpress.com/2011/03/24/связь-matlab-и-simatic-s7-300-по-opc/> (дата обращения: 25.10.2017)
16. OPC-сервер: подключение к SCADA TRACE MODE. Учебный фильм. [Электронный ресурс]. URL: <https://www.youtube.com/watch?v=He9qiSNBDik> (дата обращения: 25.10.2017)
17. ГОСТ 7. 1 – 2006 Библиографическая запись. Библиографическое описание. Общие требования и правила составления. — Переиздание 2011. — М. : ФГУП «Стандартинформ». 2011. — 62 с.
18. ГОСТ 2. 105 – 95 Общие требования к текстовым документам. Изд с поправкой (ИУС 12–2001) — М. : ФГУП «Стандартинформ». 2005. — 30 с.
19. ГОСТ 2. 104 – 68 ЕСКД Основные надписи. — Изд с поправкой (ИУС 12–2002) — М. : ФГУП «Стандартинформ». 2002. — 12 с.

## ПРИЛОЖЕНИЕ А

### Пример составления реферата курсового проекта

#### РЕФЕРАТ

Пояснительная записка 30 с., 1 рисунок, 1 таблица, 12 источников, 5 приложений, графическая часть — 2 листа формата А 3.

Объектом разработки является АРМ оператора контроля и управления основными параметрами установки печь-ковш.

Цель работы — создать операторский интерфейс (человеко-машинный интерфейс) системы мониторинга, содержащий один узел АРМ (автоматизированное рабочее место) оператора технологического процесса установки ковш-печь.

На основании анализа объекта автоматизации выбрана централизованная одноуровневая структура управления, составлены перечни параметров, подлежащих автоматическому контролю, регулированию, сигнализации. Выбрана структура для реализации АСР, разработаны экраны мониторов и программы имитации работы объекта и регуляторов. Разработан операторский интерфейс системы мониторинга и управления, содержащий один узел АРМ (автоматизированное рабочее место) оператора технологического процесса. Проведены отладка, настройка и имитация работы проекта при помощи профайлера (внутреннего или сетевого).

ТЕХНОЛОГИЧЕСКИЙ ПРОЦЕСС, ОБЪЕКТ  
АВТОМАТИЗАЦИИ, СТРУКТУРА УПРАВЛЕНИЯ, SCADA-система,  
УЗЕЛ RTM, КАНАЛЫ СВЯЗИ, МНЕМОСХЕМА.

## ПРИЛОЖЕНИЕ Б

### Пример оформления содержания курсового проекта

#### СОДЕРЖАНИЕ

Введение	4
1 Создание экранов АРМ	5
2 Создание шаблонов программ	14
3 Создание узлов проекта и базы каналов	20
4 Разработка имитационной модели объекта и обеспечение связи через OPC- сервер	23
5 Имитация проекта	28
Заключение	30
Перечень ссылок	31
Приложение А Структурная схема шаблона программы «расходом воды»	33

УЧЕБНОЕ ИЗДАНИЕ

**Роман Юрьевич Ткачев  
Николай Зельманович Бойко**

**РАЗРАБОТКА СИСТЕМЫ ОПЕРАТИВНОГО КОНТРОЛЯ  
И УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ  
НА БАЗЕ SCADA-систем**

Курсовое проектирование  
по дисциплине  
«Программирование компьютерно-интегрированных систем»

Учебно-методическое пособие

В авторской редакции

Художественное оформление обложки

Н. В. Чернышова

---

Заказ № 346. Формат 60x84 <sup>1</sup>/<sub>16</sub>.

Бумага офс. Печать RISO.

Усл. печат. л. 3 Уч.-изд. л. 2,6

Издательство не несет ответственность за содержание  
материала, предоставленного автором к печати.

Издатель и изготовитель:

ГОУВПО ЛНР «Донбасский государственный технический университет»  
пр. Ленина, 16, г. Алчевск, ЛНР, 94204  
(ИЗДАТЕЛЬСКО-ПОЛИГРАФИЧЕСКИЙ ЦЕНТР, ауд. 2113, т/факс 2-58-59)

Свидетельство о государственной регистрации издателя, изготовителя  
и распространителя средства массовой информации

МИ-СГР ИД 000055 от 05.02.2016