

МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

МОСКОВСКИЙ ЭНЕРГЕТИЧЕСКИЙ ИНСТИТУТ

В.П. ЗВЕРЬКОВ, С.В. МЕЗИН

*Подготовка и реализация проектов АСУТП на базе ПТК
SPPA T-3000 фирмы Сименс*

ЛАБОРАТОРНЫЙ ПРАКТИКУМ

по курсу

«Информационное обеспечение систем управления»

для студентов, обучающихся по направлениям 220200 и
140100

УДК

621.398

З-434

УДК: 621.398.72.06.001.57(076)

Утверждено учебным управлением МЭИ в качестве учебного пособия для студентов

Подготовлено на кафедре автоматизированных систем управления тепловыми процессами.

Рецензент кафедры: доцент Кузищин В.Ф.

Зверьков В.П., Мезин С.В.

Подготовка и реализация проектов АСУТП на базе ПТК SPPA T-3000 фирмы Сименс: Лабораторный практикум – М.: Издательство МЭИ, 2010.– 88 с.

ISBN 5-7046-0929-5

Лабораторный практикум предназначен для знакомства с системой проектирования SPPA T-3000 фирмы Сименс, получения навыков работы и самостоятельной реализации простейших системы управления теплоэнергетических объектов. Приводится краткое описание возможностей программы и правила использования основных блоков и команд на примерах реализации системы сигнализации, позиционной, одноконтурной и каскадной систем регулирования.

Для студентов специальности 21.02 "Автоматизация технологических процессов и производств", а также специалистов, работающих в области автоматизации непрерывных технологических процессов.

ISBN 5-7046-0929-5

© Московский энергетический институт, 2010

СОДЕРЖАНИЕ

Практические занятия №1, 2. Проектирование алгоритмов управления в SPPA-T3000.....	4
Введение.....	4
Начало работы	5
Что доступно с рабочего стола?	6
Открытие вида проекта.....	7
Демонстрация работы готового проекта	8
Назначение исполняемого контейнера	9
Активизация.....	9
Запуск проекта.....	10
Операции управления	10
Окончание. Демонстрация	12
Практическое занятие №3. Проектирование алгоритмов управления в SPPA-T3000.....	15
Первый проект.....	15
Описание индикаторов и кнопок команд панели управления	27
Двухпозиционное регулирование уровня в баке	29
Практическое занятие № 4	38
Проектирование каскадной системы регулирования в SPPA-T3000.....	38
Теоретические основы.....	38
Задание	40
Выполнение задания	40
Описание и пути для элементов, используемых в схеме.....	45
Принцип режима регулирования с обратной связью.....	46
Режим слежения.....	47
Описание индикаторов и кнопок команд	54
Вопросы к защитах лабораторных работ	59

Практические занятия №1, 2. Проектирование алгоритмов управления в SPPA-T3000

Цель занятия: знакомство с системой проектирования и получение первых навыков работы на основе запуска демонстрационного проекта.

Введение

Применение микропроцессорных контроллеров для автоматизации технологических объектов управления требует использования специальных программ, облегчающих программирование контроллеров и создание интерфейса оператора. Это приводит к уменьшению времени на подготовку проектов, запуска системы в работу и на техническое обслуживание во время эксплуатации, сводя до минимума все затраты. Сбор, хранение и управление информацией по автоматизации объекта на протяжении всего периода эксплуатации является трудной задачей, которая, однако, легко решается при помощи SCADA-систем и систем технологического программирования. Как правило, каждая фирма - разработчик микропроцессорных средств автоматизации имеет такие программные средства.

Программные средства предназначены, с одной стороны, для подготовки, отладки и загрузки в контроллеры рабочих программ управления объектом и, с другой стороны, подготовки, отладки и загрузки в операторские станции видеogramм для отражения процесса функционирования объекта с возможностью воздействия на исполнительные устройства, задатчики и т.д. Системы проектирования ориентированы на международный стандарт МЭК-61131, в котором описаны основные языки программирования контроллеров. В системах проектирования используются графические редакторы и другие программные средства, позволяющие осуществить быстрый и экономичный переход от программирования технологических задач к высококачественному представлению схем технологических процессов и реализации управления.

Система проектирования SPPA-T3000 одна из таких систем, разработанная фирмой Сименс для своих микропроцессорных средств автоматизации и сетевых средств поддержки оператора.

Архитектура SPPA-T3000 основана на Web-технологии. Интернет стал стандартным методом предоставления распределенной информации на различных платформах, и его возможности только начинают использоваться. В близком будущем все больше приложений будет интегрироваться в Web-среду для осуществления возможности доступа и совместного использования информации во всем мире. SPPA-T3000 разработана таким образом, чтобы пользоваться преимуществами нынешних и будущих возможностей Web-технологий и легко представлять информацию в любом формате и в широком диапазоне компьютерных платформ.

Для визуализации процессов на всей станции и выполнения функций

управления и сигнализации необходим только стандартный Интернет-браузер, работающий под любой операционной системой. Выпадающие меню, всплывающие окна, возможности многооконной работы, простой поиск, всплывающие подсказки и многие другие функции делают использование системы операторами, обслуживающим персоналом и инженерами практически интуитивным. Благодаря этому достигается эффективность проектирования и эксплуатации, гарантирующая обоснованное принятие решений и безболезненное внесение проектных изменений и управление ими.

Система управления и контроля SPPA-T3000 – программно-технический комплекс, разработанный немецкой компанией Siemens, является системой управления производственными процессами, создающей уникальную открытую платформу для внедрения современных, экономически выгодных, ориентированных на будущее решений в области автоматизации производственных предприятий. Ее современное исполнение и архитектура обеспечивают возможность получения экономически выгодных рентабельных решений на всех этапах развития предприятия, включая проектирование, разработку, ввод в эксплуатацию, обслуживание и эксплуатацию, а также дальнейшее развитие.

Начало работы

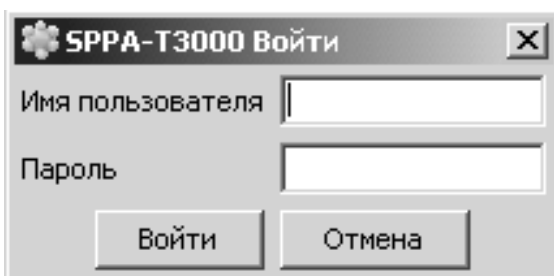


Рис. 1.1. Окно входа в систему

1. Для начала выполнения работы необходимо осуществить вход в систему. Для этого необходимо найти на рабочем столе ярлык SPPA-T3000 и запустить приложение.

2. В диалоговом окне **Вход** введите **имя пользователя** и **пароль**.

3. Нажмите кнопку **Войти**.

После этого откроется рабочий стол, показанный ниже, который является центром проектирования, и на который вызываются все остальные окна во время работы.

Сразу после открытия рабочего стола пользователь может сориентировать его или запустить автономную ориентацию по отношению к макету рабочего стола, узнать о том, как открывать и закрывать виды на рабочем столе, а также узнать о том, что делать в ре-

Рис. 1.2. Рабочий стол

жиме работы.

Что доступно с рабочего стола?

Рабочий стол является центральным видом или видом наивысшего уровня в системе SPPA-T3000. Все остальные виды вызываются с рабочего стола. Виды системы SPPA-T3000 предназначены для удовлетворения потребностей различных групп пользователей, например инженеров, операторов, обслуживающего персонала и т.д. Виды открываются как окна в рабочей области рабочего стола.

С рабочего стола можно открыть следующие виды, выбрав пункт меню **Приложения**:

- **Вид проекта (Project view):** это окно является основным для проектирования. Здесь отображается полная структура проекта, а также все данные проекта.
- **Вид иерархии видеограмм:** здесь задается структура, которой пользуется оператор для перехода между разными видеограммами.
- **Вид диагностики:** отображает информацию о системе и используется для анализа ошибок.
- **Экран ПСО:** на экране ПСО отображаются все сигналы в упорядоченном виде.
- **Редактор прототипов:** используется для создания и редактирования определяемых пользователем пиктограмм и панелей управления.
- **Отчеты:** используются для извлечения из архива и просмотра истории развития технологического процесса.
- **Выход:** завершение сеанса и закрытие приложения SPPA-T3000.

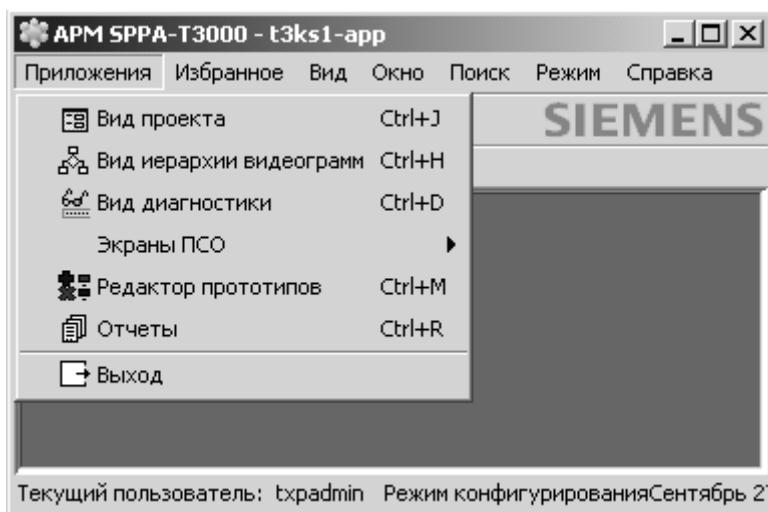


Рис. 1.3. Перечень видов системы SPPA-T3000

наблюдать реакцию проекта на введенные изменения.

Чтобы открыть вид проекта, вид иерархии видеограмм, экран ПСО, вид диагностики и т. д., в меню **Приложения** выберите нужный вид.

Другим необходимым пунктом главного меню рабочего стола является **Режим работы**. Выбрав этот пункт, пользователь получает доступ к переключению режимов работы: проектирование или эмуляция работы. В режиме эмулятора можно задавать значения переменных, вводить команды и

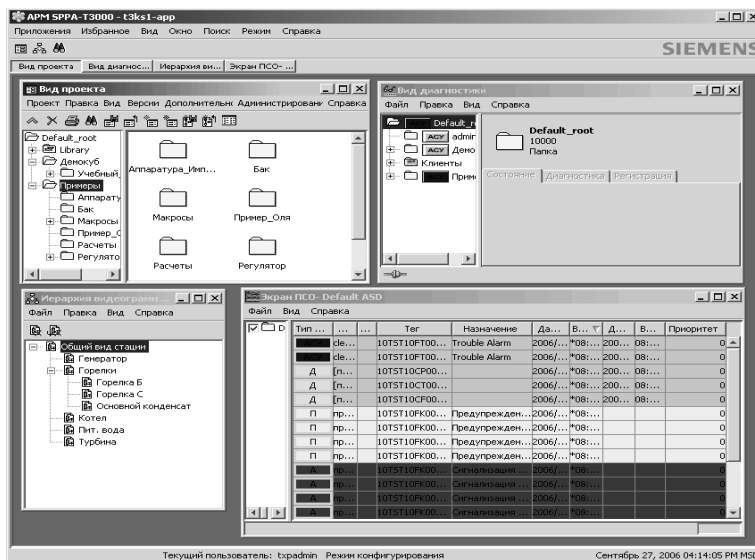


Рис. 1.4. Пример рабочего стола с несколькими открытыми видами проекта.

рование проекта.

Можно одновременно открывать и размещать на рабочем столе произвольное количество различных видов (здесь приводится пример рабочего стола с несколькими видами).

В третьей сверху строке рабочего стола для каждого открытого вида отображается своя кнопка. Если виды перекрывают друг друга, нажмите на необходимую кнопку, чтобы открыть нужный вид.

Открытие вида проекта

Выберите меню **Приложения** в строке меню рабочего стола, затем пункт **Вид проекта**. На экране появится окно **Вид проекта**, состоящее из двух рамок:

Левая рамка — это дерево объектов. Дерево объектов показывает расположение узлов в проекте. Узлы можно сравнить с папками, организующими проект. Эта организация называется иерархией проекта.

Существует два типа узлов.

- **Узлы проекта**: они содержат все настроенные для данного проекта схемы.



- **Узлы библиотеки (Library)**: здесь представлены все функции и элементы, которые можно получить для настройки проекта.



Правая рамка — это список объектов. Список объектов показывает содержание узла, выбранного в дереве объектов.

В узле, например, могут содержаться другие узлы проекта, представленные следующими элементами.

Ярлыки файлов функциональных схем и видеограмм

После этого вид открывается в рабочей области.

Из окна вида проекта открывается редактор схем, позволяющий сконфигурировать все прикладные программы и логику, а также видеограммы, используемые для работы с объектом. Исполняемый контейнер настраивается, и пользователь может вносить изменения в конфигурацию системы запуска. В окне вида проекта также выполняется администри-

Проект состоит из двух основных частей: функциональные схемы и видеодиаграммы.

Функциональные схемы представляют собой схему управления технологическим объектом, выполненную методом технологического программирования на базе модулей из библиотек Сименс.

Функциональные схемы отражают логику управления от входных

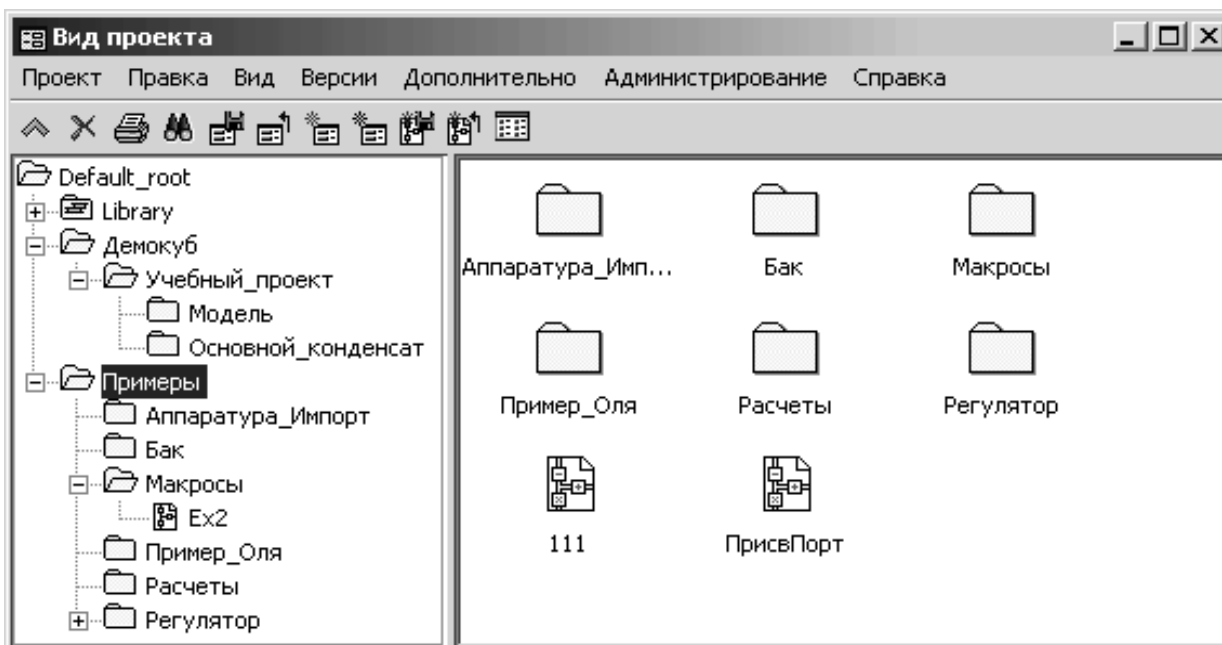


Рис.1. 5. Окно Вид проекта.

сигналов до управляющих воздействий. Здесь должны быть представлены все входные и выходные сигналы, сумматоры, ограничители, компораторы, блокировки, блоки вычислений и интерфейсов и другие модули и блоки, необходимые для управления объектом. Файлы функциональных схем имеют ярлыки, показанные на рисунке слева.

Видеодиаграммы представляют собой графические рисунки, отображающие, главным образом, технологические системы и позволяющие осуществлять наблюдение и управление процессом. Видеодиаграммы состоят из динамических и статических элементов. Элементы статического изображения включают в себя все элементы, которые не меняются при протекании технологического процесса. Файлы видеодиаграмм схем имеют ярлыки, показанные на рисунке слева.

Элементы динамического изображения включают в себя все элементы видеодиаграммы, которые отображают значения или состояния технологического процесса.

Демонстрация работы готового проекта

Для наилучшего представления возможностей проектирования различных схем управления технологическими объектами с помощью системы SPPA-T3000 рассмотрим демонстрационный пример. Откройте Вид проекта из пункта Приложения и выберите узел REFERENCE PLANTS, выберите узел BOILER, и в нем узел проекта Combustion. Этот проект содержит

четыре объекта: три функциональных схемы (AFC – Air flow control, BM – boiler master и INT - instruments) и одну видеодиаграмму (FUEL GAS SYSTEM). Дважды нажав на любой объект из проекта можно посмотреть логику построения для функциональных схем или видеодиаграмму технологического процесса сжигания топлива. Для запуска рассматриваемого проекта необходимо задать исполняемый контейнер и активировать объекты проекта (функциональные схемы и видеодиаграмма)

Организация исполняемых контейнеров — это специфика каждого отдельного проекта. Исполняемый контейнер – это раздел вычислительных ресурсов ПТК СПРА – Т3000, где будет исполняться программа соответствующая созданному проекту. Программа может выполняться либо на сервере, либо в технологическом контроллере. Как правило в контроллере исполняются отлаженные программы управления технологическим процессом. Необходимый исполняемый контейнер назначает преподаватель.

Назначение исполняемого контейнера

Выберите узел проекта “**Combustion**” в дереве объектов.

Выберите меню **Правка**, затем **Назначить исполняемый контейнер**.

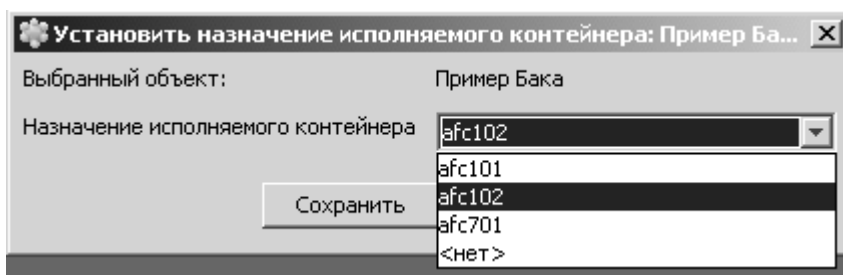


Рис. 1.6. Окно назначения исполняемого контейнера

Откроется диалоговое окно “**Назначить исполняемый контейнер**”.

Выберите исполняемый контейнер в раскрывающемся списке

“**Назначение исполняемого контейнера**”. В примере, приведенном выше, выбран исполняемый контейнер afc 102.

Нажмите на кнопку **Сохранить**.

Функциональные схемы узла проекта “**Combustion**” будут назначены выбранному исполняемому контейнеру.

Активизация

Выберите узел проекта “**Combustion**” в дереве объектов.

В меню **Правка** выберите пункт **Активизировать**.

Откроется диалоговое окно, требующее подтвердить выбранный узел проекта. Нажмите на кнопку **Активизировать**.

Откроется диалоговое окно, уведомляющее об успешной активизации схем.

Необходимо подтвердить активизацию.

В меню **Правка** выберите пункт **Подтверждение последней**

активизации / деактивизации.

Откроется диалоговое окно, уведомляющее об успешном подтверждении активизированных схем.

Проект **Combustion** (функциональные схемы и видеодиаграммы) теперь активизирован в системе.

Запуск проекта

Для запуска проекта необходимо из режима проектирования (конфигурирования) перейти в режим работы.

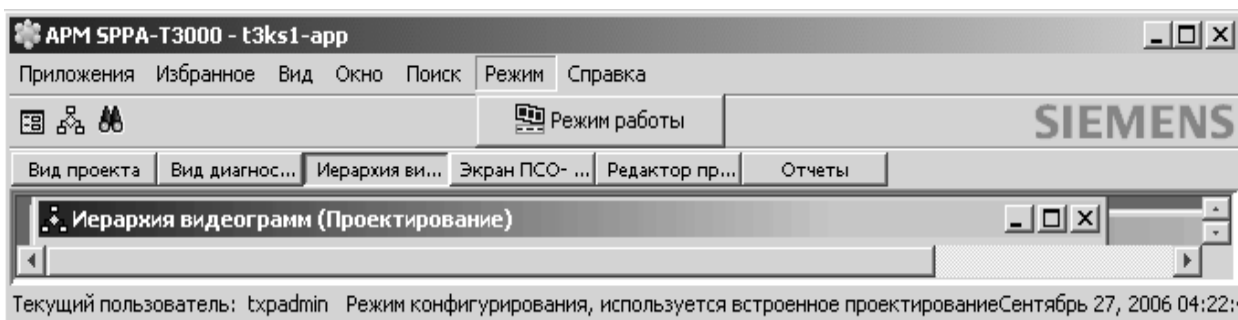


Рис. 1.7. Переключение режима работы

Для запуска (или остановки) режима работы проекта выберите пункт главного меню рабочего стола **Режим**, а затем — пункт **Режим работы**.

Появление индикаторов в верхней строке главного меню подтверждает переход в режим работы. В режиме работы можно задавать значения переменных, вводить команды и наблюдать реакцию проекта на введенные изменения.

Операции управления

Операциями управления процессом являются все вводы оператора в систему SPPA-T3000, которые прямо или косвенно воздействуют на процесс. Для этой цели используются панели управления (виртуальные блоки дистанционного управления), вызываемые с любого дисплея.

Панель управления — это рабочий интерфейс для оборудования объекта. Панели управления используются для эксплуатации оборудования и визуализации процессов.

Панель управления можно открыть двойным нажатием на ее динамический символ (пиктограмму). С помощью кнопок панели управления можно управлять оборудованием; информация о состоянии оборудования доступна на индикаторах панели управления.

Для панелей управления существует ряд основных принципов и методов работы, которые описаны ниже.

Для объяснения основных принципов ниже приведен пример панели управления для двигателя:

- Строка в верхней части окна содержит тег оборудования, иногда с разделителем тегов и внутренним идентификатором. Обычно этот тег совпадает с тегом, назначенным определенному оборудованию на схемах P&I. Таким образом, можно быстро идентифицировать физическое устройство.

- Под тегом расположено наименование устройства.

Кнопки заголовка (слева направо)

Текстовая панель управления



Открыть функциональную схему

В новом окне открывается динамическая функциональная схема, содержащая логику этого устройства

Символьная панель управления



Принудительно активизировать порты

Открывается диалоговое окно принудительной активизации портов функции автоматизации.



Открыть вид диагностики

Вид диагностики отображает подробную диагностическую информацию о функции автоматизации. Эта информация помогает диагностировать ошибки.



Открыть экран графиков

Экран графиков автоматически назначается системой и показывает predetermined signals of the control panel. If signals for trend display are not archived, curves start from the moment of opening the chart screen. The control panel graphics can be printed.



Открыть вид точки

Открывается диалог, содержащий вид точки устройства.



Открыть справку

Открывается документация, касающаяся данной функции.

Рабочая область панели управления расположена под заголовком. В этой области:

- отображается информация о конкретном состоянии устройства,
- выполняются рабочие действия.

Индикаторы для текстовых панелей управления представлены утопленными прямоугольниками с текстовой меткой.

Подсвеченная метка, например, **ОШИБКА**, говорит о том, что активен соответствующий сигнал; в данном случае, насос находится в состоянии сигнализации.

Функции индикации и управления могут объединяться в одном элементе, например, для имитирования работы кнопки с подсветкой.

- Подсвеченная метка, например, **СТОП**, говорит о том, что активен соответствующий сигнал, в данном случае насос остановлен.
- Неподсвеченные кнопки означают, что функция недоступна.

Для символьных панелей управления состояние устройства отображается символом пиктограммы, присутствующим на панели управления.

Большинство команд выполняются в два шага:

1. Нажатием на требуемую кнопку (например, ПУСК). Кнопка **ВЫПОЛНИТЬ** станет доступна.
2. Нажатием на кнопку **ВЫПОЛНИТЬ**.

Для некоторых кнопок команд, например, кнопки **СТОП** привода, для работы требуется только одно нажатие. После нажатия они становятся недоступными. Нажатие на кнопку **ВЫПОЛНИТЬ** не требуется.

После открытия панели управления можно задать её положение указате-

лем мыши. По умолчанию она открывается рядом с принадлежащей ей пиктограммой. Пользователь может установить положение открытия для панели управления. Положения открытия устанавливаются в соответствии с экранной сеткой (меню Окно).

После открытия панели управления можно изменить ее размер, как любого другого окна. При необходимости можно вернуть исходный размер панели управления.

Нажмите дважды левой клавишей мыши на функциональную схему Boiler master. На открывшейся функциональной схеме можно наблюдать текущие значения параметров функциональных алгоритмов реализующих логику управления котлом. На этой схеме блок SPADJE выполняет функции задатчика, а Boiler master - регулятора расхода газа. Оба блока имеют панели управления, которые открываются последовательным нажатием левой и правой кнопки мыши. В открывшемся окне выбрать пункт **открыть панель управления**. На появившейся панели управления можно изменять заданное значение и другие величины, причем каждое изменение параметра требует подтверждения (**Execute**). Измените заданное значение регулируемой величины с помощью панели управления задатчика (**SPADJ**) с помощью движка на панели **OUT**, либо нажатием кнопки внутреннее задание **INT SP**. Не забудьте подтвердить сделанные изменения (клавишей **Execute**). Блок управления имеет пиктограммы вызова дополнительных возможностей и свойств функционального элемента. Нажатием пиктограммы **Графика** вызывается окно изменяемых параметров блока. Наблюдайте работу проекта.

Окончание. Демонстрация

Для окончания демонстрации работы проекта необходимо выйти на узел проекта **Combustion** и вызвав пункт меню правка, выбрать команду **Деактивизировать**, затем подтвердить последнюю деактивацию и снять режим работы.

Библиотека SPPA-T3000

Технологическое программирование алгоритмов управления осуществляется на базе модулей библиотеки SPPA-T3000. Библиотека SPPA-T3000 содержит программные модули, используемые для реализации всех функций автоматизации и управления.

Одной из ключевых характеристик продукта SPPA-T3000 является сетевая основа системы управления технологическим



процессом с компонентной основой архитектуры программного обеспечения. Поэтому функции автоматизации АСУ (ФА) представлены отдельными компонентами.

Кроме того, компоненты АСУ выполняют не только традиционные функции автоматизации, но также функции, необходимые для управления, контроля, сигнализации, проектирования и диагностирования. Эта функциональность представлена стандартными интерфейсами.

Как следствие, использовавшийся ранее принцип независимости при создании систем автоматизации и заключавшийся в независимости различных модулей, выполняющих функции управления, проектирования и диагностирования является устаревшим.

Система SPPA-T3000 состоит из различных компонентов, взаимодействующих друг с другом и с устройствами. Функции управления, проектирования и диагностирования и т. д. являются разными представлениями одной, интегрированной системы. Поэтому каждый тип функции автоматизации состоит из идентичной стандартной структуры и имеет одинаковые интерфейсы для целей процесса управления. Это:

- Интерфейсы входа и выхода, позволяющие осуществлять обмен сигналами через порты с другими ФА и прокси аппаратуры.
- Операционный интерфейс, позволяющий осуществлять обмен сигналами через порты с панелями управления.
- Интерфейс сигнализации. Каждый двоичный порт выхода является потенциальным источником сигнализации. При объявлении дополнительного свойства **Сигнализация**, этот порт автоматически соединяется с системной сигнализацией.
- Интерфейс диагностики. Каждый ФА предоставляет порты выхода специальными свойствами для целей диагностирования.
- Интерфейс проектирования. Порты входа и выхода ФА можно сконфигурировать с помощью интерфейса проектирования.
- Интерфейс выполнения позволяет ФА работать в цикле программы исполняемого контейнера.

Ниже представлены программные модули библиотеки SPPA-T3000, используемые для реализации функций автоматизации и управления. Подробную информацию по каждому модулю можно найти в справочной системе SPPA.

- Аналоговые вычисления: ABS, ADD, ASIG, AVER, AXFR, COMP, CONV, DB, DERIV, DIV, EXP, INT, LEADLAG, LN, MAX, MIN, MINMAX, MONIT, MULTIPLY, NROOT, POLY, POWER, PT1, RAXFR, SUB, TRIGON;
- Дискретные вычисления: AND, BSEL, BSIG, BXFR, COUNTER, EXOR, JK_FF, LPULSE, MINPULS, NOT, OR, PULSE, RS_FF, T_LON, T_OFF, T_ON;
- Логическое управление: ACTUATOR, BRANCH, DCO, GC, MOTOR, MOTOR_2SR, PBO, PRESEL, SERVO, SLC, SLC_OP_SD,

SOV, STEP;

- Ограничение и контроль: ANA, ASEL2, ASEL3, ASMON, ASMON_R, BIN, BSMON, BSMON_R, GET_SET_QTY;
- Расчёт эффективности:
- AIRFLUEGAS, COAL_MASS_FLOW, COMBUSTION, CP_AIRFLUEGAS, DESULF, ETA_BOILER, FEEDER, FUEL_ENERGY, LOSS_ASH, LOSS_CO, LOSS_FLUEGAS, LOSS_RADIATION, U_COMBUST_COAL, U_COMBUST_GAS, USEFUL_HEAT;
- Регулирование: CCTRL, DELAY, PID, SP_AV, SPADJ, SPCTRL;
- Специальные функции: ASQMON, BSQMON, GAS, GASFLOW, GASMEAS, LEVEL, MEMORY, SAVE, STEAM, TIME, TIMEMON, WATER;
- Статистика и наработка: INT_AVER, MOV_AVER, OH_COUNT, SC_COUNT, STEADY;
- Функции технолог. процесса: DYNOPH, $H=f(P,T)$, $P=f(T_SAT)$, $S=f(P,T)$, $T=f(P_SAT)$, $T=f(P,H)$, $T=f(P,S)$, $V=f(P,T)$.

Практическое занятие №3. Проектирование алгоритмов управления в SPPA-T3000

Цель занятия: знакомство с системой проектирования, получение первых навыков работы и самостоятельная реализация простейшей системы управления.

Первый проект

Задание

Спроектировать систему сигнализации уровня воды в баке, подаваемой с помощью насоса. При достижении уровнем воды в баке нижнего значения (Сигнал: Низкий уровень) или при достижении уровнем воды в баке верхнего значения (Сигнал: Высокий уровень) осуществляется предупредительная сигнализация. При достижении уровнем воды в баке предельного значения (Сигнал: Остановка насоса) насос отключается. Значения уровня будем вводить вручную и смотреть на реакцию системы сигнализации.

Примерная технологическая схема системы сигнализации уровня воды в баке и аварийного отключения насоса подачи воды в бак показана

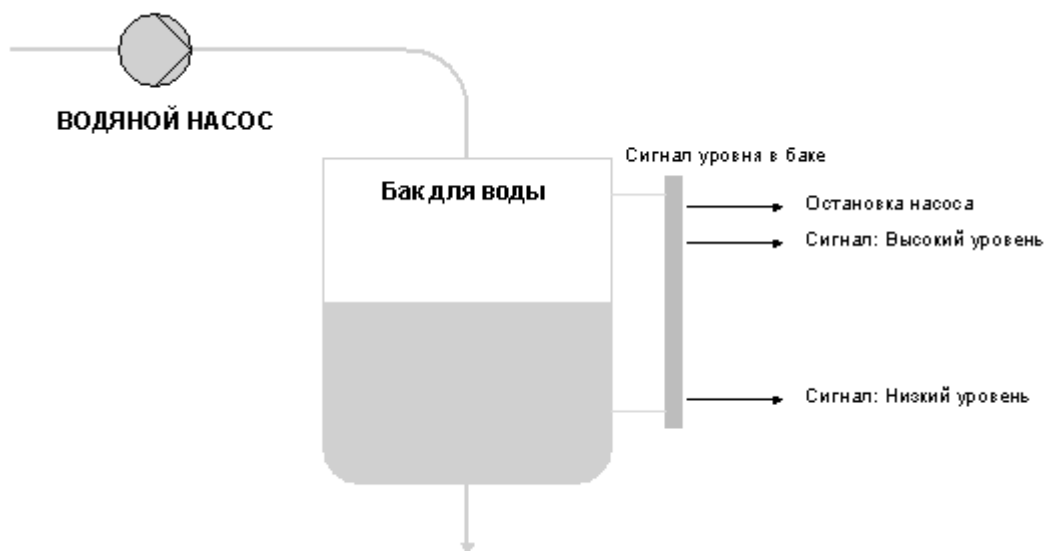


Рис. 2.1. Технологическая схема объекта управления

на рис.2.1.

Объект для примера включает в себя:

- бак для воды;
- насос, подающий воду в бак;
- аналоговый сигнализатор уровня воды в баке с пороговыми значениями для остановки насоса и сигнализаторы высокого и низкого уровня.

Для выполнения данного примера в SPPA-T3000 необходимо выполнить следующие действия:

1. Зарегистрироваться в системе и открыть рабочий стол.
2. Создать вид проекта и дополнительную структуру проекта.
3. Создание шаблонов для функциональной схемы и видеограммы.
4. Ввод информации в функциональную схему.
5. Ввод информации в видеограмму.
6. Назначить исполняемый контейнер и запустить пример на выполнение в эмуляторе.

1. Регистрация в системе SPPA – T3000.

Запустите SPPA – T3000 с помощью ярлыка на рабочем столе.

Введите логин и пароль (задает преподаватель). После открытия рабочего стола, нажмите на клавишу **Вид проекта**.

2. Создание узла проекта.

Выберите корневой узел в дереве объектов. Этот узел будет родительским узлом создаваемого узла.

Выберите меню **Проект**, затем **Новый**, затем **Узел проекта**.

Создан новый узел проекта с именем **“Новый Узел”**, под корневым узлом.

Затем новый узел проекта может быть переименован, для этого :

выберите узел **“НовыйУзел”** в дереве объектов, меню **Проект**, затем **Переименовать**. На имени узла появится курсор. Удалите имя старого узла и введите новое имя **“Пример Бака”** (или свою фамилию) затем нажмите на Enter.

Теперь внутри созданного узла можно создавать другие узлы вашего проекта, если он будет иметь сложную иерархическую структуру или просто другие проекты, не забывая присваивать новые имена каждому узлу проекта.

3. Создание шаблонов для элементов проекта.

3.1. Создание шаблона функциональной схемы.

Необходимо отметить, что сначала создается пустой для функциональной схемы, а затем он заполняется необходимыми элементами.

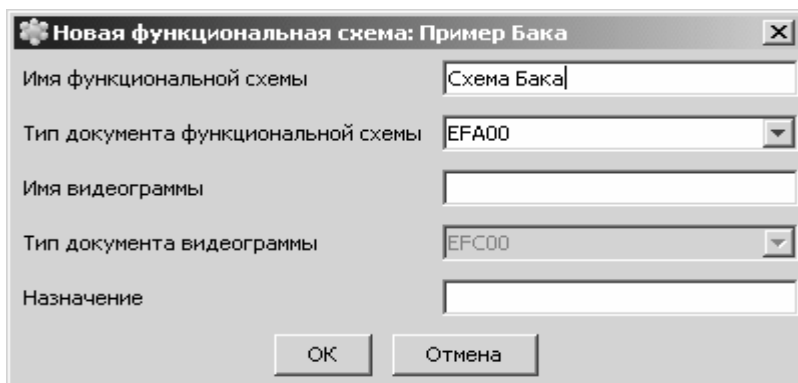


Рис. 2.2. Диалоговое окно **“Новая функциональная схема”**

Для создания пустого шаблона функциональной схемы выберите узел проекта **“Пример Бака”** в дереве объектов.

В меню **Проект** выберите пункт **Новый**, затем

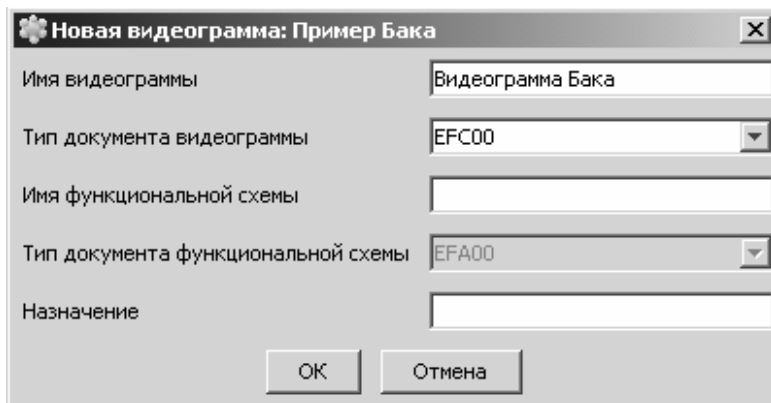
Функциональная схема. Откроется диалоговое окно **“Новая функциональная схема”**.

Введите имя в поле “Имя функциональной схема”, например, **Схема бака** . Поле **Видеограмма** оставьте пустым.

Нажмите на кнопку **ОК**. Создается пустой шаблон для функциональной схемы.

3.2. Создание шаблона видеограммы

Для создания пустого шаблона видеограммы выберите узел проекта



“**Пример бака**” в дереве объектов. В этом узле создается видеограмма. В меню **Проект** выберите пункт **Новый**, затем **Видеограмма**. Откроется диалоговое окно “**Новая видеограмма**”.

Рис. 2.3. Диалоговое окно “Новая видеограмма”

Введите имя для элемента”

Видеограмма”: **Видеограмма бака**. Поле **Функциональная схема** оставьте пустым.

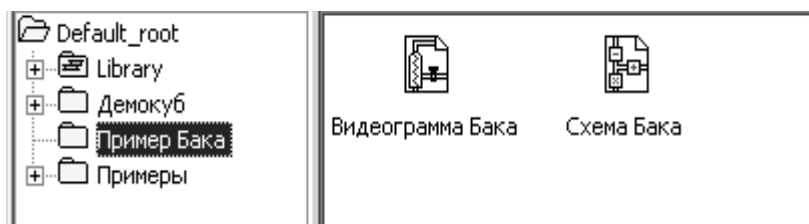


Рис. 2.4. Окно Вид проекта с созданными схемами

Нажмите на кнопку **ОК**. Создается пустой шаблон для видеограммы.

Итак, созданы два пустых шаблона для функциональной схемы и видеограммы

для выполняемого примера.

После создания пустых шаблонов для функциональной схемы и видеограммы в них необходимо внести необходимую информацию.

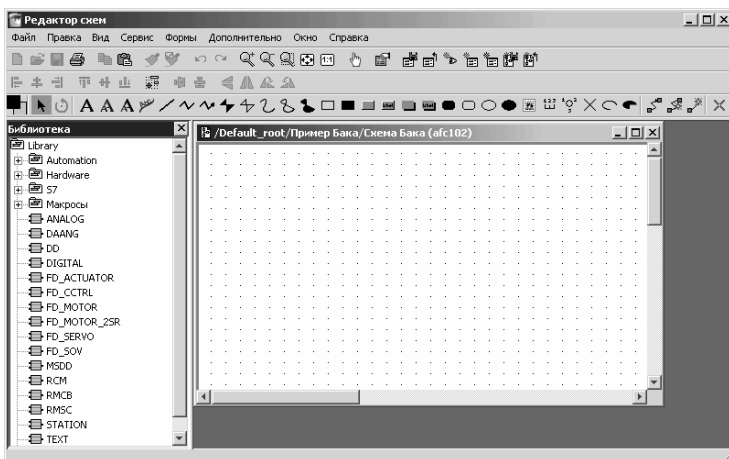


Рис. 2.5. Пустой шаблон функциональной схемы, открытой в режиме конфигурирования

4. Ввод информации в функциональную схему.

Ввод информации в функциональную схему (конфигурация) предполагает размещение блоков ФА, осуществление соединения блоков и настройка уставок или коэффициентов.

Выберите узел проекта “**Пример бака**” в дереве

объектов, а затем функциональную схему “Схема бака” в списке объектов. Выберите меню **Правка**, затем пункт меню **Открыть в режиме конфигурирования**. Редактор схем запускается и функциональная схема “Схема бака” открывается (схема является пустой, так как конфигурация еще не была выполнена).

4.1. Размещение блоков ФА

Логика автоматизации формируется путем размещения библиотечных алгоритмов на функциональной схеме.

Так как в задании предусмотрено включение и выключение двигателя насоса, то необходимо использовать библиотечный алгоритм **MOTOR**, который используется для управления и контроля двигателями приводов, используется для включения и выключения двигателя с помощью дистанционных и автоматических команд, а также команд защиты.

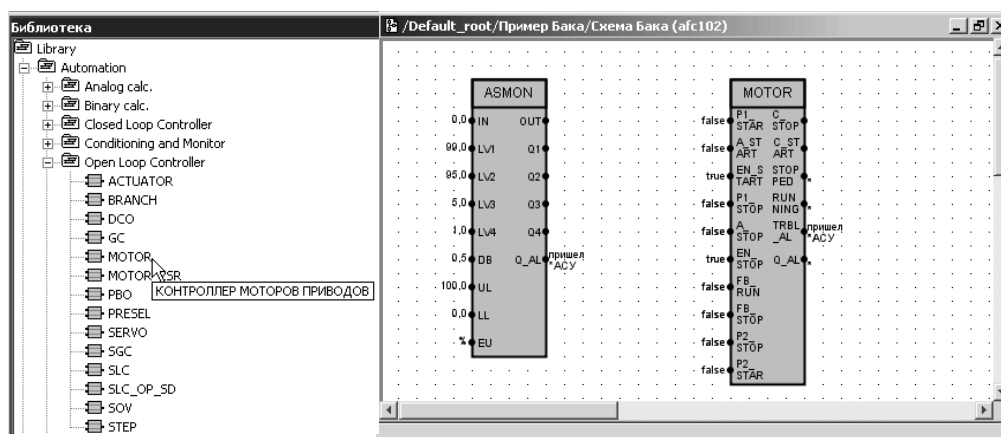


Рис. 2.6. Размещение ФА на функциональной схеме

Найдите блок **ФА MOTOR** в библиотеке. Он находится под **Library > Automation > Open Loop Controller**. Дважды нажмите на узел библиотеки «Automation». Будут отображены узлы библиотеки для функций автоматизации. Дважды нажмите на узел библиотеки **Open Loop Controller**. С помощью мыши выберите функцию автоматизации **MOTOR**. Переместите курсор в пустую функциональную схему “Схема бака”. Символ блока ФА перемещается вместе с курсором. Переместите символ блока ФА в положение, как показано на иллюстрации ниже, и установите его с помощью мыши. Символ будет размещен на схеме.

Для реализации предупредительной и аварийной сигнализации и выполнения необходимых действий, используется библиотечный алгоритм **ASMON** – монитор аналогового сигнала. Основным назначением функции монитора аналогового сигнала является выполнение до шести проверок аналоговой величины на достижение различных пределов. Повторив описанную выше процедуру, поместите на схему блок **ФА ASMON**. Блок **ФА ASMON** можно найти в библиотеке **Library > Automation > Conditioning and Monitor**.

4.2. Соединение блоков

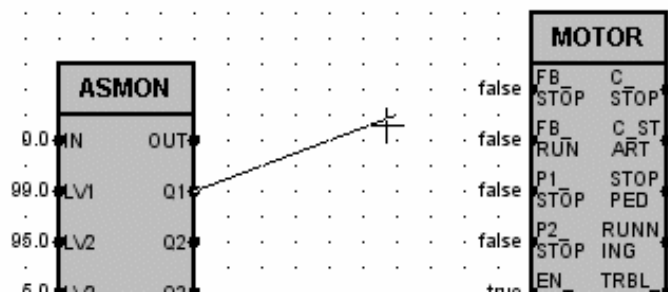



Рис. 2.7. Соединение портов блоков ФА

После размещения функций автоматизации (библиотечных алгоритмов) на схеме, требуется установить связь между ними. Для выключения насоса при достижении уровня воды в баке предельного значения требуется подать команду на

останов. Двоичный сигнал управления при достижении предельного значения генерируется в ФА блоке **ASMON**. Используется выход **ASMON Q1** (предельное значение 1). Для выполнения останова двигателя выходной порт **Q1 ASMON** необходимо подключить к входному порту **P1_STOP** (защита 1 Команда СТОП) мотора. Для создания данного соединения выполните следующие действия. Щелкните по значку **Соединение**  “(Обработчик графики)” на панели инструментов.

Чтобы увидеть доступные для соединения порты, проведите курсором мыши над блоком **ASMON**. В случае если порт доступен он будет подсвечен зеленым цветом. Щелкните по порту **Q1** блока **ASMON**. Щелкая по порту, щелкайте только по описанию порта внутри ФА блока. Не щелкайте по границе ФА блока. Линия подключается к порту источника. Линия привязана к курсору и двигается за ним. Чтобы увидеть доступные для соединения порты, проведите курсором мыши над блоком **MOTOR**. Щелкните по порту назначения **P1_STOP**. После чего соединение будет создано.

4.3. Настройка уставок срабатывания сигнализации.

У блоков, размещенных на функциональных схемах есть ряд связанных с ними свойств или настроек. Они называются параметрами, к которым можно обратиться в маске параметров. Маска параметров — это диалоговое окно, которое содержит все параметры символа. Параметры также используются для настройки сигналов. Сигналы используются для предупреждения обслуживающего персонала о состояниях, которые требуют внимания. Сигналы конфигурируются на функциональных схемах и просматриваются операторами в режиме работы. Сигналы, используемые в системе **SPPA – T3000**, называются тегами. Можно задать тип сигнализации (аварийная, предупредительная), также можно задать текст и описания сигнала. Сигналы, которые должны быть сконфигурированы для работы, приведены в табл. 1.

Таблица 1. Описание уставок для сигнализации

Имя тега	Уровень	Сигнал	Тип сигнализации
УРОВЕНЬ БАКА	90%	НИН	СИГНАЛ
УРОВЕНЬ БАКА	80%	ВЫС	ПРЕДУПРЕЖДЕНИЕ
УРОВЕНЬ БАКА	20%	НИЗК	ПРЕДУПРЕЖДЕНИЕ

Для открытия маски параметров выберите блок **ФА ASMON**. Щелчком правой кнопки мыши откройте контекстное меню. В контекстном меню выберите пункт **“Редактировать”**.

Тип	Имя	Значени...	Видимый порт	Видимый параметр	Архив <<	Сигнализация <<	Описание
BX	UL	100,0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	верхний предел
BX	LL	0,0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	нижний предел
BX	EU	%	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Единица измерения
BX	DB	0,5	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Абсолютная зона нечувствительно...
BX	LV1	90,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Предельное значение 1
BX	LSG1	Да	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Генерация предельного сигнала 1
BX	HI_LOW1	Высокий	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Верхний/Нижний предел 1
BX	LV2	80,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Предельное значение 2
BX	LSG2	Да	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Генерация предельного сигнала 2
BX	HI_LOW2	Высокий	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Верхний/Нижний предел 2
BX	LV3	20,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Предельное значение 3
BX	LSG3	Да	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Генерация предельного сигнала 3
BX	HI_LOW3	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Верхний/Нижний предел 3
BX	LV4	1,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Предельное значение 4
BX	LSG4	Да	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Генерация предельного сигнала 4
BX	HI_LOW4	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Верхний/Нижний предел 4
BX	LV5	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Предельное значение 5

Рис. 2.8. Маска параметров для блока ФА ASMON

Открывается маска параметров для блока **ФА ASMON**. Если колонка слишком узка, то ее можно расширить: наведите указатель на границу колонки заголовка таблицы, затем нажмите на левую кнопку мыши и растяните колонку до нужной ширины, после чего отпустите кнопку мыши. Убедитесь в том, что открыт лист маски «Набор параметров 1» (если нет, откройте его щелчком). В соответствии с таблицей 1, введите предельные значения для трех сигналов:

LV1: 90 - предельное значение №1,

LV2: 80 - предельное значение №2,

LV3: 20 - предельное значение №3.

Проверьте диапазон сигналов (который соответствует диапазону 0..100 %):

UL: 100 (Высший предел)

LL:0 (Низший предел)

Проверьте состояние функции контроля предельного сигнала. Функция контроля предельного сигнала отслеживает, не превысил ли входной сигнал IN предельное значение LVn. Нарушение предела указывается на соответствующем двоичном выходе Qn. Эта функция включается параметром LSGn.

Проверьте что значения LSG1, LSG2, LSG3 установлены (то есть, в параметре значение установлено **“Yes”**), а также верхнее или нижнее ограничение задано надлежащим образом:

HI_LOW1, HI_LOW2 установлены в положение High, что означает срабатывание сигнализации при превышении значения уставки.

HI_LOW3 установлен в положение Low, что означает срабатывание

сигнализации при значении параметра ниже значения уставки уставки.

Тип	Имя	Значени...	Ви...	Видим...	Архи...	Сиг...	П...	Тип сигнал...	Сообщение а...	Сообщение д...
ВХ	LSG5	Нет	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
ВХ	HI_LOW5	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
ВХ	LV6	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
ВХ	LSG6	Нет	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
ВХ	HI_LOW6	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
ВЫХ	OUT	0,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
ВЫХ	Q1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	АВАРИЙНАЯ	>НІНІ	<НІНІ
ВЫХ	Q2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	ПРЕДУПРЕ...	>НІ	<НІ
ВЫХ	Q3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	ПРЕДУПРЕ...	<LO	>LO
ВЫХ	Q4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	АВАРИЙНАЯ		
ВЫХ	Q5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	АВАРИЙНАЯ		
ВЫХ	Q6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	АВАРИЙНАЯ		
ВЫХ	CF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0	СООБЩЕН...	пришел	ушел
ВЫХ	Q_AL	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0	СООБЩЕН...	пришел	ушел
ВЫХ	OUT_GQ	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				

Рис. 2.9. Маска параметров для блока ФА ASMON

Сделайте прокрутку до нижней части маски параметров для выходных портов, после чего щелкните по заголовку колонки **Сигнализация** для раскрытия параметров сигнала. Включите сигнализацию, установив флажки в колонках **Сигнализация** для выходов Q1, Q2, Q3 и отредактируйте колонки **Активное сообщение** и **Неактивное сообщение** для выходов Q1, Q2, Q3:

Q1: >НІНІ и <НІНІ

Q2: >НІ и <НІ

Q3: <LO и >LO

Нажмите на кнопку «Сохранить». Теперь сконфигурированы три сигнала.

5. Ввод информации в шаблон видеодиаграммы.

Видеограмма, как известно, является визуальным отражением участка технологического процесса. Для создания видеодиаграммы используется набор графических примитивов, или так называемая свободная графика.

Свободная графика — это статичные элементы видеодиаграмм, используется для создания схематического представления объекта, в таком случае оператор может легко понять расположение и функции приводов или измерения, отображаемые на картинках. Видеодиаграммы состоят из статических и динамических элементов.

5.1. Создание статических элементов видеодиаграммы.

В соответствии с заданием (см. рис. 2.1) технологический процесс состоит из, бака и насоса соединенного между собой трубопроводами. Для того чтобы создать видеодиаграмму, выполните следующие действия:

В редакторе схем откройте видеодиаграмму **”Видеодиаграмма бака”**.

Начертите бак, для этого :


На панели инструментов редактора схем щелкните по значку 

“Скругленный прямоугольник с заливкой”. Двигайте курсор по схеме, курсор примет форму креста. Нарисуйте бак, следуя примеру на приведенной ниже картинке. Для создания угла бака нажмите и удерживайте левую кнопку мыши, затем переведите курсор на противоположный угол бака, отпустите кнопку мыши.

Начертите трубы:

На панели инструментов редактора схем нажмите на значок “Ломаная”.

Нажмите один раз на начальную точку линии, затем перемещайте курсор. Линия будет перемещаться вместе с курсором.

Нажмите один раз на угол линии, затем — два раза на конечную точку линии. Линия будет начерчена. Нажмите на значок “Ломаная со стрелкой”.  Начертите трубу на выходе бака.

Созданная картинка упрощенной схемы бака приведена на рис. 2.10.

5.2. Создание динамических элементов видеограммы.

Динамические элементы используются в видеограммах для предоставления оператору информации об объекте в реальном времени (например, о рабочем состоянии двигателя, температурах), и предоставления оператору возможности управлять объектом (например, приводами запуска или остановки).

Динамические элементы, расположенные в видеограммах, называются пиктограммами. Пиктограммы содержатся в своей собственной библиотеке. Сразу после размещения пиктограммы должно быть установлено рабочее соединение с соответствующим блоком ФА (в функциональной схеме).

В лабораторной работе динамические элементы должны быть размещены и подключены для следующих технологических элементов:

- насоса;
- индикации уровня бака.

Для размещения изображения насоса найдите пиктограмму

MOTOR_SYM5 в библиотеке прототипов и разместите её в центре линии. Путь к пиктограмме: **Library > Prototypes > Drives.**

Для подключения пиктограммы к блоку ФА для насоса дважды щелкните по пиктограмме насоса. Откроется маска

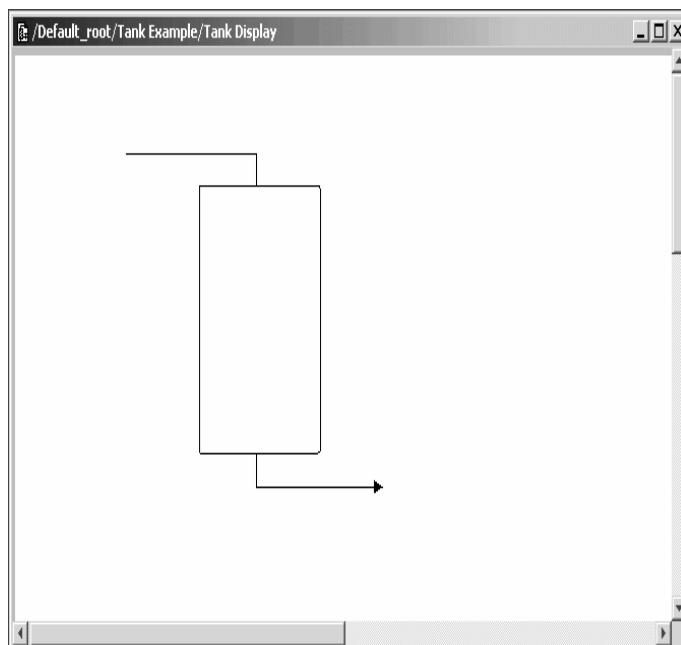


Рис. 2.10. Визуальное отражение бака на видеограмме

параметров пиктограммы, приведенная на рис. 2.11.

Блок ФА, к которому должно быть установлено рабочее соединение,

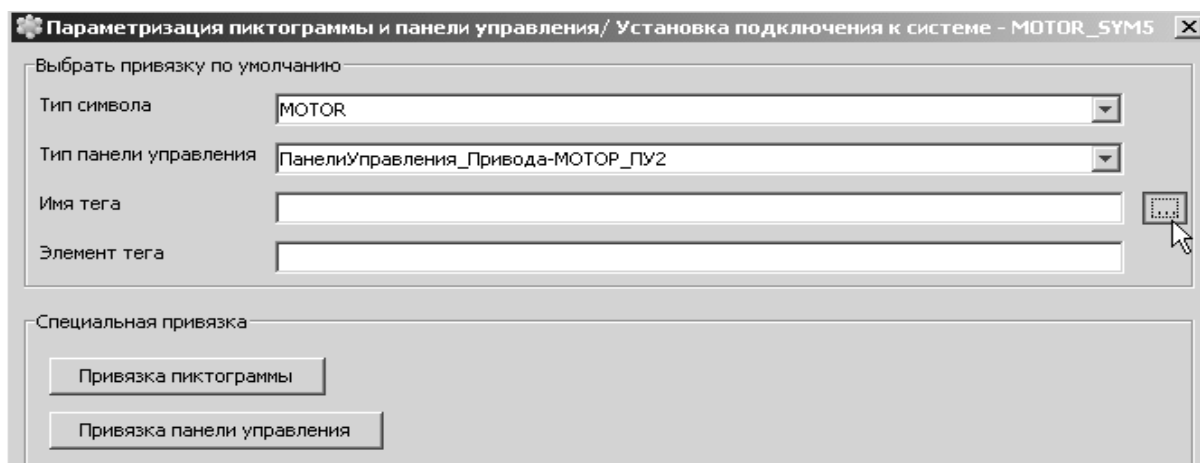


Рис. 2.11. Фрагмент маски параметров ФА MOTOR

находится с помощью браузера тегов, для того чтобы открыть его переместите курсор в поле “Имя тега” и нажмите на левую клавишу мыши. С правой стороны поля “Имя тега” появляется кнопка, после нажатия на которую откроется окно “Обзор тэгов”. Выберите щелчком узел “Пример бака”, браузер тегов, фрагмент окна которого приведен на рис. 2.12.

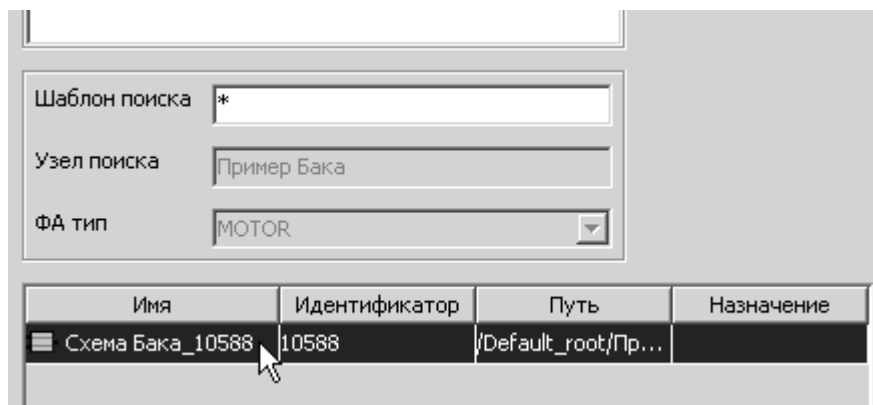


Рис. 2.12. Фрагмент окна браузера тэгов

Нажмите на кнопку “Начать поиск”. Теперь в браузере тегов начинается поиск функций

автоматизации типа **MOTOR** в выбранном узле. После окончания поиска откроется окно браузера тэгов. Результаты поиска отображаются в нижней части окна браузера тегов. Отдельный элемент списка соответствует насосу в примере выполнения. В данном случае идентификатор 10588 соответствует **ФА MOTOR** в узле “Пример бака”.

Выберите в списке элемент, щелкнув по нему. Нажмите на кнопку **ОК**.

В окне маски параметров **ФА MOTOR** в строке “элемент тега” появится номер идентификатора (соответствие с браузером тегов не обязательно).

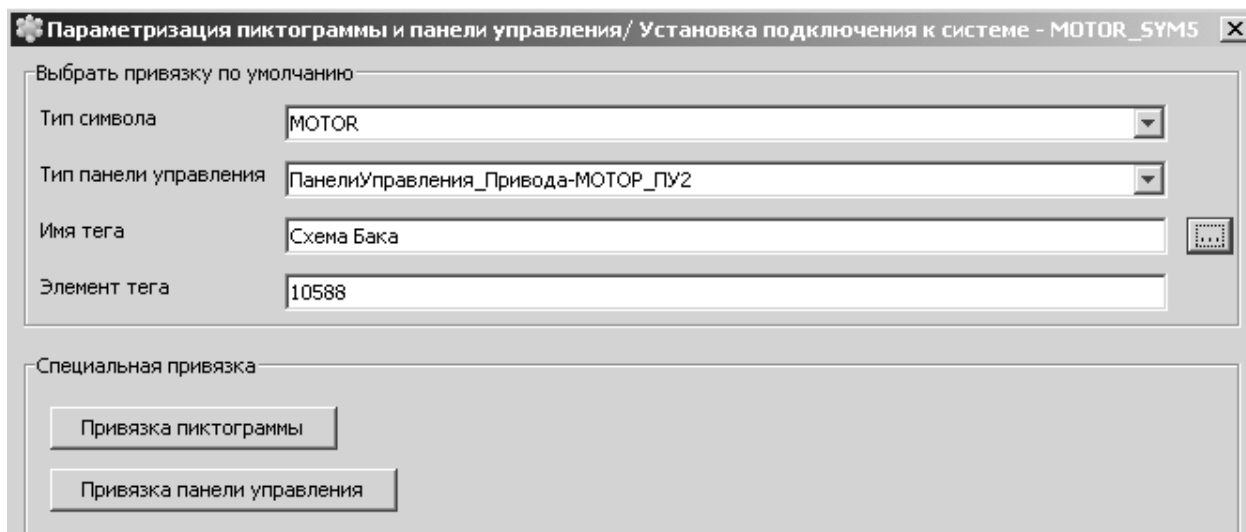


Рис. 2.13. Окончательный вид маски параметров **ФА MOTOR**

Теперь рабочее соединение с блоком **ФА** устанавливается автоматически. В маске параметров пиктограммы автоматически появляется имя тега и элемент тега блока **ФА**.

Нажмите на кнопку **Сохранить** для сохранения рабочего соединения, в противном случае нажмите на кнопку **Отмена**.

5.3. Создание индикации уровня в баке

Индикация уровня бака создается так же, как и для насоса.

Разместите пиктограмму индикатора уровня бака и установите рабочее соединение, как показано на рисунке.

Пиктограмма — **ASMON_SYM2** (путь: **Library > Prototypes > Monitoring**). Подключите пиктограмму к блоку **ФА ASMON** в функциональной схеме «Пример бака», так же как и для предыдущей пиктограммы. Итоговое визуальное отражение технологического процесса на видеодиаграмме показано на рис. 2.14.

6. Запуск примера на выполнение

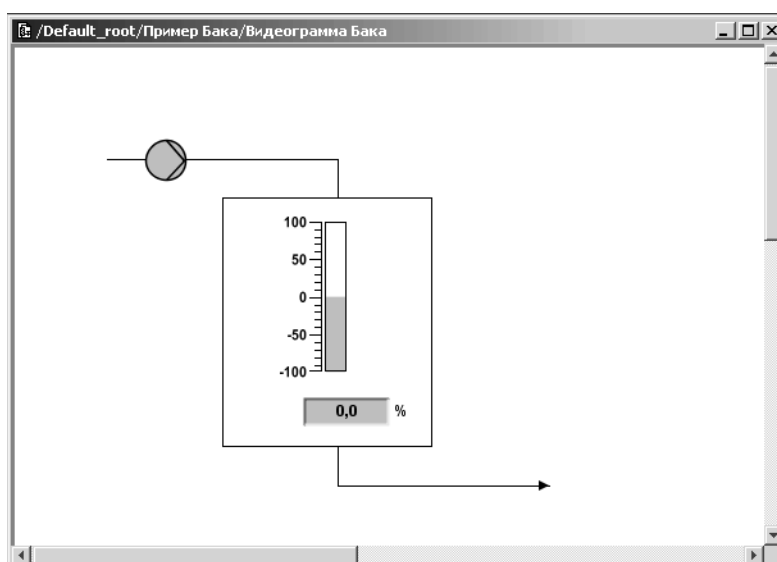


Рис. 2.14. Итоговое визуальное отражение технологического процесса на видеодиаграмме

Чтобы созданные функциональные схемы и видеодиаграммы выполняли в проекте предусмотренные действия необходимо подтвердить созданные схемы и активизировать их в одном из контейнеров: Для этого следует:

Открыть вид проекта и выбрать узел проекта “**Пример бака**” в дереве объектов.

Подтвердить функциональные схемы и видеодиаграммы, для чего:

В меню **Правка** выбрать пункт **Подтвердить**. Откроется диалоговое окно, где будут перечислены все схемы, которые будут подтверждены, нажать на кнопку **Подтвердить**.

Откроется диалоговое окно, уведомляющее об успешном подтверждении схем.

Для запуска рассматриваемого проекта необходимо задать исполняемый контейнер и активировать объекты проекта (функциональные схемы и видеодиаграмма)

Организация исполняемых контейнеров — это специфика каждого

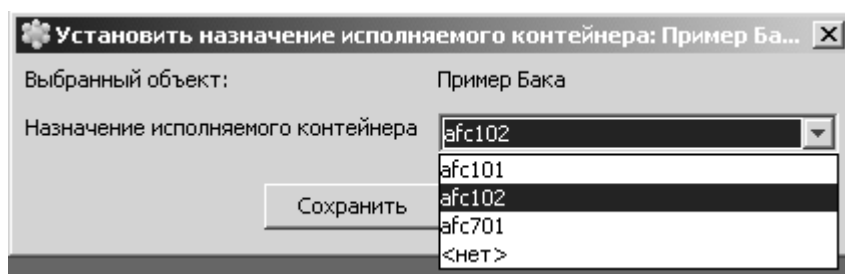


Рис. 2.15. Окно назначения исполняемого контейнера

отдельного проекта. Исполняемый контейнер – это раздел вычислительных ресурсов ПТК SPPA – Т3000, где будет исполняться программа соответствующая созданному проек-

ту. Программа может выполняться либо на сервере, либо в технологическом контроллере. Как правило, в контроллере исполняются отлаженные программы управления технологическим процессом. Необходимый исполняемый контейнер назначает преподаватель.

Для назначения исполняемого контейнера выберите узел проекта “**Пример бака**” в дереве объектов. Далее выберите меню **Правка**, затем **Назначить исполняемый контейнер**.

Откроется диалоговое окно “**Назначить исполняемый контейнер**”.

Выберите исполняемый контейнер в раскрывающемся списке “**Назначение исполняемого контейнера**”. В примере, приведенном выше, выбран исполняемый контейнер afc 102.

Нажмите на кнопку **Сохранить**.

Функциональные схемы узла проекта “**Пример бака**” будут назначены выбранному исполняемому контейнеру.

Для активизации выберите узел проекта “**Пример бака**” в дереве объектов.

В меню **Правка** выбрать пункт **Активизировать**.

Откроется диалоговое окно, требующее подтвердить выбранный узел проекта. Нажмите на кнопку **Активизировать**.

Откроется диалоговое окно, уведомляющее об успешной активизации схем. Затем следует подтвердить активизацию: в меню **Правка** выберите пункт **Подтверждение последней активизации / деактивизации**.

Откроется диалоговое окно, уведомляющее об успешном подтверждении активизированных схем.

Весь пример выполнения (функциональные схемы и видеограммы) теперь актуализирован в системе. Теперь схемы можно просматривать в режиме работы.

Запуск режима работы.

При открытом рабочем столе нажмите на кнопку **Режим** в верхней строке меню, а затем на кнопку **Режим работы** в открывшемся окне.

Запуск режима работы можно наблюдать по индикатору в левой нижней части экрана. Пример рабочего стола в режиме работы показан на рис. 2.16.

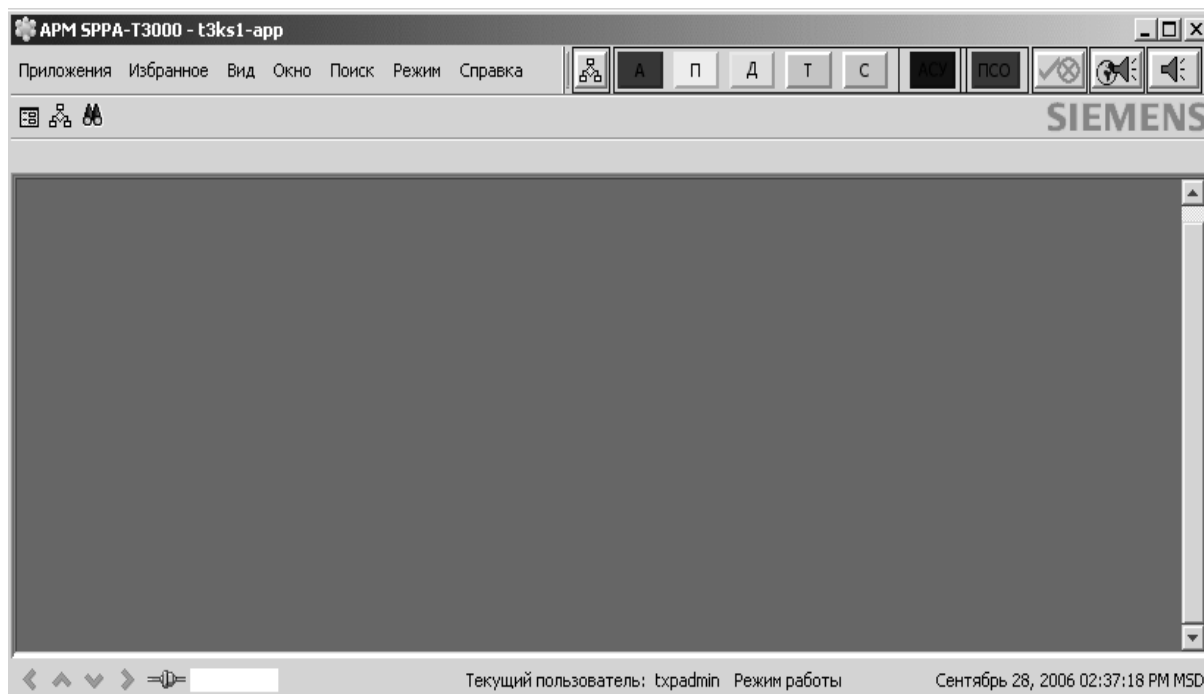


Рис. 2.16. Рабочий стол в режиме работы

Кнопка иерархии видеограмм и индикаторы общей сигнализации отображаются в строке меню (цветные кнопки). Справа от кнопки ПСО (последовательность сообщений оператору) размещены кнопки квитирования сигнализации.

Для открытия видеограммы бака в рабочем режиме необходимо щелкнуть в окне проекта по значку видеограммы правой кнопкой мыши и выбрать **Открыть в режиме работы**.

После запуска видеограммы в режиме работы необходимо последовательно выставить значения уровня в баке и проверить срабатывание сигнализации. Для проверки достижения значений уставок, используется принудительное выставление значений.

Для ввода входных значений в блок сигнализации необходимо открыть

функциональный блока **ASMON** в рабочем режиме. Для этого щелкнуть в окне проекта по значку блока правой кнопкой мыши и выбрать **Открыть в режиме работы**.

Откройте контекстное меню, нажав на правую кнопку мыши в любом месте схемы, и выберите пункт **Принудительно активизировать порты**.

Откроется диалоговое окно, показанное на рис. 2.17. и содержащее все порты блока **ФА ASMON**, которые могут быть принудительно активизированы.

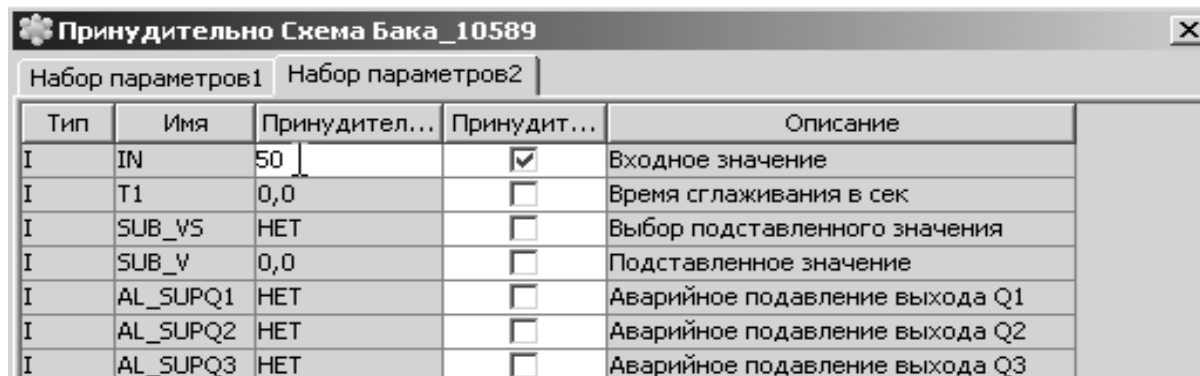


Рис. 2.17. Принудительная активация портов в маске ФА ASMON

Для принудительного выставления сигнала уровня в баке установите флажок порта **IN** в колонке **Принудительно**. Появляется флажок. Введите **Принудительное значение** на уровне 50.

Нажмите на кнопку **Сохранить**. Сигнал уровня бака принудительно выставляется на 50%.

Теперь на динамической функциональной схеме, панели управления и видеодиаграмме появляется принудительное значение 50% уровня бака.

Поменяйте выставленное значение сигнала уровня в баке последовательно на 85%, 95%, 5%. Напоминаем, что как было указано на первом занятии, открыть окно принудительной активации портов можно, нажав соответствующую пиктограмму на панели управления.



Рис.2.18. Панель управления мотором



Принудительно активизировать порты Открывается диалоговое окно принудительной активизации портов функции автоматизации.

Для прекращения мигания индикатора сигнализации необходимо квитировать всю сигнализацию выбранной видеодиаграммы, нажав клавишу справа от вызова экрана ПСО.

Обратите внимание на отключение насоса (загорается кнопка **START** (появляется возможность пуска мотора)).

Пример панели управления для моторов показан на рис.2.18.

Описание индикаторов и кнопок команд панели

управления

- Индикация состояния
 - **СИГНАЛИЗАЦИЯ (ошибка)**: индикация групповой сигнализации.
 - **ТЕСТ ПОЛОЖ (тест поз)**: если питатель мотора отображает индикацию тестового положения, т. е. контуры управления работают, но мотор отсоединен (вне корпуса) от питания, контакты можно контролировать с помощью логики управления мотора.

Если на панели управления отображается индикация тестового положения, сигналы обратной связи контрольного центра / коммутационной аппаратуры обрабатываются логикой управления мотора как выходные команды.

- **ЛОКАЛЬНЫЙ (по месту)**: локальный режим активируется с помощью внешнего сигнала, посланного логике контроллера. При работе в локальном режиме все команды автоматики и дистанционные команды заблокированы, обрабатываются лишь команды защиты.

- Кнопки команд

- **ПУСК**: действительно только для режима вручную. Запускает мотор.
- **СТОП**: действительно только для режима вручную. Останавливает мотор.
- **СБРОС**: используется для сброса состояния ошибки.
- **НАЛАДКА**: после активизации режима наладки сигналы команд автоматики и защиты не принимаются. Мотор будет принимать с панели управления только дистанционные команды.
- **ЗАПРЕТ**: если выбран режим запрета, выходы команд функции контроллера мотора заблокированы, ни одна из выходных команд не активирована. В данном случае команды защиты, автоматики и дистанционные команды приниматься не будут, т. е. контроллер мотора игнорирует все вводы команд. Тем не менее, все входы обратной связи будут обрабатываться и контролироваться. Режим запрета имеет самый высокий приоритет из всех режимов работы.
- **ДИСТ**: при работе в режиме вручную мотор принимает только дистанционные команды, тем не менее сигналы защиты обрабатываются, защитные отмены будут работать.
- **АВТО**: при работе в автоматическом режиме мотор принимает только команды автоматики, тем не менее сигналы защиты обрабатываются, защитные отмены будут работать.

Нажмите дважды на его изображение и запустите насос в работу. Откройте экран ПСО на рабочем столе и посмотрите перечень сообщений о срабатывании сигнализации. Экран ПСО (последовательность сообщений оператору) – экран, содержащий перечень сообщений сигнализации в хронологическом порядке.

Таким образом, первый проект закончен и Вы ознакомились с азами

использования системы SPPA-T3000, построения функциональных схем, работы с элементарной графикой и привязками графических элементов к элементам логики и запуска ее в режиме проверки.

Двухпозиционное регулирование уровня в баке

Задание

Спроектировать систему двухпозиционного управления подачей воды в бак с помощью включения и отключения насоса. После запуска системы насос включается и вода подается в бак. Уровень начинает увеличиваться. При достижении уровнем воды в баке верхнего предельного значения (Сигнал: Высокий уровень) мотор отключается. При этом, в связи со свободным сливом воды из бака, уровень начнет уменьшаться и при достижении уровнем воды в баке нижнего значения (Сигнал: Низкий уровень) мотор включается. Уровень опять начинает увеличиваться. При достижении уровнем воды в баке верхнего предельного значения мотор опять отключается и уровень опять начнет уменьшаться и т.д. При этом осуществляется предупредительная сигнализация. При вводе значения уровня воды в бак выше второго предельного значения (Сигнал: Остановка насоса), насос отключается по аварийной ситуации. Примерная технологическая схема системы управления подачей воды в бак (видеограмма) показана на рис.2.1. и остается без изменений.

Целью задания является создание элементарной функциональной схемы способной автоматически регулировать уровень в баке. Для создания такой схемы можно использовать пример из проекта №1, внося в него некоторые изменения. Один из возможных вариантов¹ реализации задачи приведен ниже.

Для модели уровня воды в баке можно использовать интегратор, реализуемый в программном задатчике SPADJ, задав в нем постоянное приращение ± 5 единиц при возможном изменении выходной величины от 0 до 100 %.

Для задания и отражения состояния мотора можно использовать триггер, выход которого подается на входы состояния мотора FB_RUN и FB_STOP. Включение мотора осуществляется подачей логического сигнала TRUE на вход FB_RUN и одновременно логического сигнала FALSE на вход FB_STOP. Для получения противоположного значения логического сигнала используется элемент NOT, реализующий логическую функцию отрицания «НЕ».

Выходы мотора подаются на соответствующие входы интегратора. Выход мотора Runned (Запущен) имеет логическое значение TRUE, если мотор включен и FALSE, если мотор выключен. Этот сигнал подается на вход A_UP программного задатчика SPADJ, разрешая или нет увеличение

¹ Обратите внимание на работу представленной схемы и исправьте ее логику в соответствии с заданием.

выхода блока на постоянное приращение вверх RR_UP. Аналогично, выход мотора Stopped (Остановлен) имеет логическое значение TRUE, если мотор выключен и FALSE, если мотор включен. Этот сигнал подается на вход A_Down программного задатчика SPADJ, разрешая или нет уменьшение выхода блока на постоянное приращение вниз RR_DOWN. Таким образом, сигнал на выходе блока SPADJ будет увеличиваться или уменьшаться на каждом такте работы системы в зависимости от значений сигналов, поступающих от блока MOTOR. Очевидно, что значения приращений RR_UP и RR_DOWN соответствуют постоянным интегрирования или скорости изменения уровня вверх и вниз.

Выход интегратора SPADJ подается на вход сигнализатора – измерителя ASMON, который показывает на своей шкале значение входного сигнала и вырабатывает TRUE или FALSE, если выполняются или нет условия, заданные при установлении значений уставок. Выходной логический сигнал сигнализатора ASMON подается на триггер, задача которого держать постоянное значение выхода, до тех пор, пока не изменится значение входного сигнала.

Функциональная схема реализации логики для выполнения задания, приведена на рис. 2.19.

Описания и пути для элементов, используемых в задании:

- **RS_FF** (*/LIBRARY/AUTOMATION/BINARY CALC.*) - RS триггер выполняет функцию бистабильной памяти, которая может иметь два внутренних состояния. RS триггер используется для хранения двоичных значений. Информация о состоянии значения доступна на выходе Q, о состоянии дополнительного значения — на выходе Q_N.

Изменение состояния RS триггера приведено ниже в таблице.

S	R	Q
0	0	Q _{n-1}
0	1	0
1	0	1
1	1	0
1	1	1

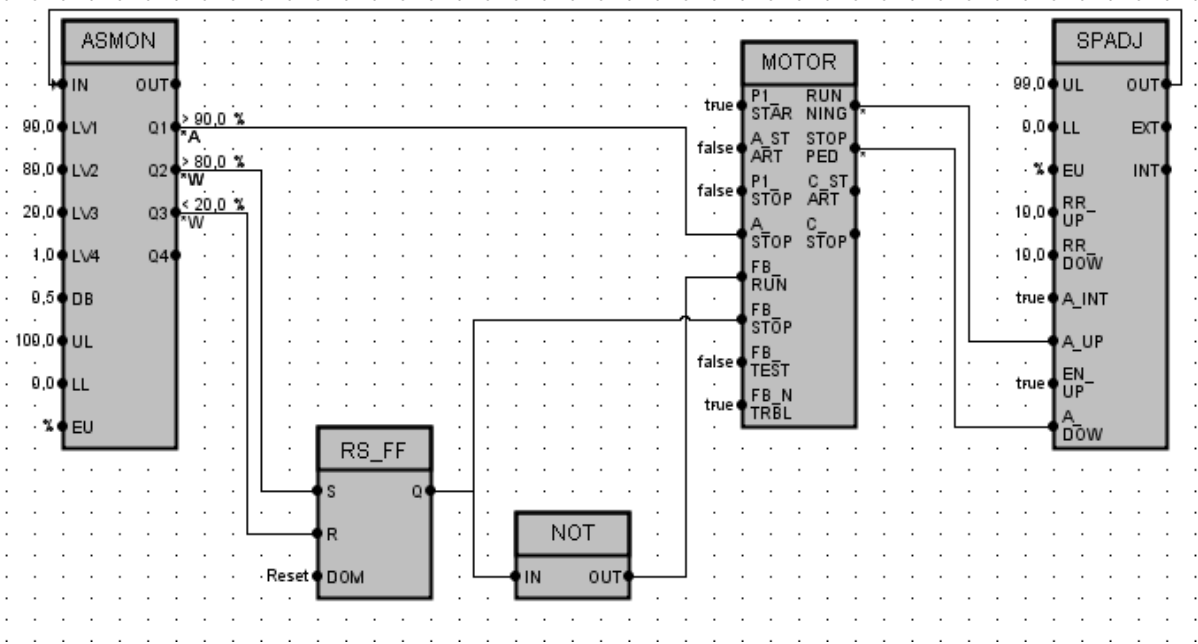


Рис. 2.19. Функциональная схема реализации логики системы двухпозиционного регулирования уровня в баке

- **NOT** (*/LIBRARY/AUTOMATION/BINARY CALC*) - Логика NOT меняет знак двоичного сигнала. Для двоичного выхода OUT устанавливается значение TRUE, если для входного сигнала установлено значение FALSE.
- **SPADJ** (*/LIBRARY/AUTOMATION/CLOSED LOOP CONTROLLER*) – Задатчик SPADJ.

Функция Задатчик используется для управления и отслеживания значения уставки. Может использоваться в трех режимах, но в задании используется в режиме ВНУТРЕННИЙ

Во внутреннем режиме интегратор уставки управляется подаваемыми вручную командами, автоматическими командами или командами защиты UP, DOWN и STOP. Он также может отслеживаться ручной уставкой M_SETP, поступающей с панели управления.

Выполнение задания:

1. Открыв функциональную схему задания №1, внесите на нее недостающие блоки, что бы получить схему как представлено на рис. 2.19.
2. Для блоков SPADJ, установите параметры для входов UL и LL(верхняя - нижняя граница), RR_UP (величина OUT УВЕЛИЧИВАЕТСЯ на RR_UNIT/сек) и RR_DOWN (RR_DOWN величина OUT УМЕНЬШАЕТСЯ на RR_UNIT/сек,) (то есть во сколько изменяется выходная величина в ед. времени, где RR_UNIT – размерность), проверить параметры в соответствии с рис.2.20.
3. Для работы схемы необходимо соединить функциональные блоки.

1) Выходы блока MOTOR - RUNNING и STOPPED соединяются соответственно с входами блока SPADJ - A_UP и A_DOWN. Таким образом, при состоянии мотора “Включен” численная величина на выходе задатчика OUT будет увеличиваться, при состоянии мотора “Выключен” численная величина на выходе задатчика OUT будет уменьшаться. Величина OUT

Тип	Имя	Значение <<	Ключ парам...	Видимый порт	Видимый па...	Архив >>	Сигнализация <<	Сообщение прише...	Сообщение ушедшей сигнал...	Режим кви...	Подавлят...
ВХ	LUL	99,0			<input checked="" type="checkbox"/>						
ВХ	LL	0,0			<input checked="" type="checkbox"/>						
ВХ	EU	%			<input checked="" type="checkbox"/>						
ВХ	RR_UP	5,0			<input type="checkbox"/>						
ВХ	RR_DOWN	5,0			<input type="checkbox"/>						
ВХ	RR_LIMIT	%			<input type="checkbox"/>						
ВХ	BASE	сек			<input type="checkbox"/>						
ВХ	TRACK_V	0,0			<input type="checkbox"/>						
ВХ	C_TRACK	<input type="checkbox"/>			<input type="checkbox"/>						
ВХ	EXT_SETUP	0,0			<input checked="" type="checkbox"/>						
ВХ	A_EXT	<input type="checkbox"/>			<input type="checkbox"/>						
ВХ	EN_EXT	<input type="checkbox"/>			<input type="checkbox"/>						
ВХ	A_INT	<input checked="" type="checkbox"/>			<input type="checkbox"/>						
ВХ	EN_INT	<input checked="" type="checkbox"/>			<input type="checkbox"/>						
ВХ	A_UP	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>						
ВХ	EN_UP	<input checked="" type="checkbox"/>			<input type="checkbox"/>						
ВХ	A_DOWN	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>						
ВХ	EN_DOWN	<input checked="" type="checkbox"/>			<input type="checkbox"/>						
ВХ	P1_DOWN	<input type="checkbox"/>			<input type="checkbox"/>						
ВХ	P1_UP	<input type="checkbox"/>			<input type="checkbox"/>						
ВХ	P2_UP	<input type="checkbox"/>			<input type="checkbox"/>						
ВХ	A_STOP	<input type="checkbox"/>			<input type="checkbox"/>						
ВЫХ	OUT	0,0			<input checked="" type="checkbox"/>						
ВЫХ	EXT	<input type="checkbox"/>			<input type="checkbox"/>			...0		Ушел - авт...	<input type="checkbox"/>
ВЫХ	INT	<input type="checkbox"/>			<input type="checkbox"/>			...0		Ушел - авт...	<input type="checkbox"/>
ВЫХ	TRACK	<input type="checkbox"/>			<input type="checkbox"/>			...0		Ушел - авт...	<input type="checkbox"/>
ВЫХ	RESET_P	<input type="checkbox"/>			<input type="checkbox"/>			...0		Manual Ope...	<input type="checkbox"/>

Рис. 2.20. Настройка блока ASMON

показывает уровень воды в баке.

2) Выходные порты блока ASMON подключаются в следующем порядке:

Выход Q1 (аварийная уставка >80%) подключается к входному порту блока MOTOR - A_STOP(команда автоматической остановки) для автоматического отключения мотора.

Выходы Q2 (предупредительная уставка >80%) и Q3(предупредительная уставка <10%) подключается к 2 портам разных блоков.

В первом случае для Q2 это вход SET блока RS_FF, при срабатывании уставки (Q2=TRUE) выход Q триггера получит значение TRUE. При этом выход триггера подключен к входному порту блока MOTOR - FB_STOP отвечающего за перевод состояния мотора в положение стоп, получает значение TRUE, а блока FB_RUN отвечающего за перевод состояния мотора в положение старт, получает значение FALSE.

Выход Q3 подключен к входу R триггера, при Q3=TRUE триггер сбрасывается, при этом выход триггера подключен к входному порту блока MOTOR - FB_STOP отвечающего за перевод состояния мотора в положение стоп получает значение FALSE, а блока FB_RUN отвечающего за перевод состояния мотора в положение старт получает значение TRUE.

4. Запуск задания

Подтвердите внесенные изменения. Активируйте схемы. Подтвердите последнюю активацию.

Запустите режим работы. В режиме работы откройте функциональную схему и видеограмму бака. На рис.2.21 приведены значения параметров функциональных блоков в режиме “работа”.

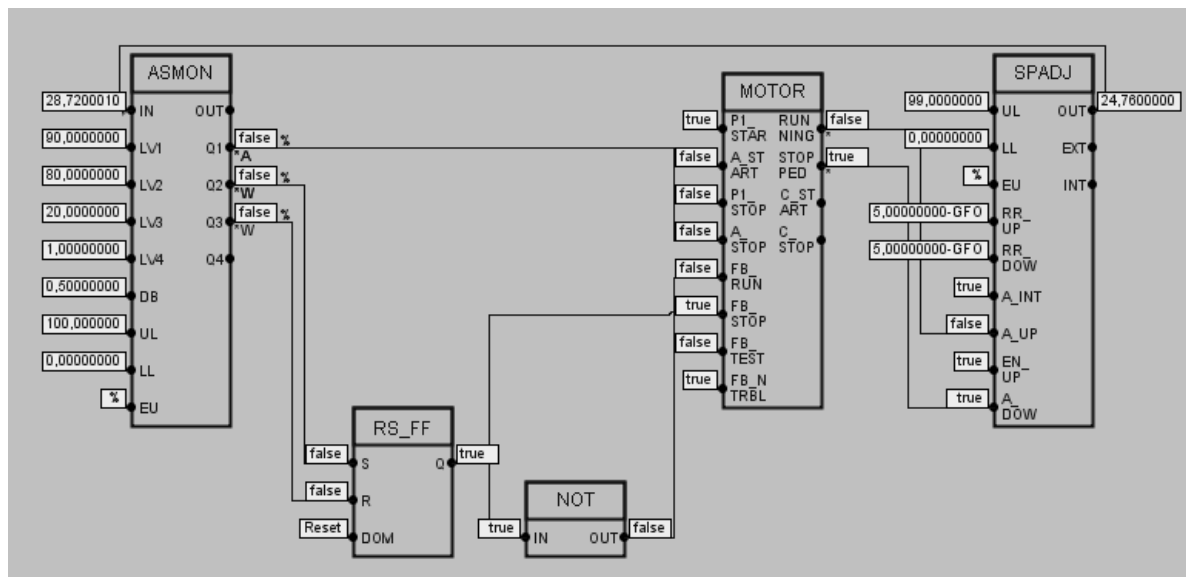


Рис. 2.21 Значения параметров функциональных блоков в режиме работа

Если схема собрана правильно, то результатом станет изменяющийся уровень в баке на видеограмме в блоке ASMON и периодически включающийся - выключающийся мотор. Во время работы функциональной схемы срабатывания уставок Q1, Q2, Q3 должны автоматически фиксироваться в экране ПСО (Экраны последовательности сообщений оператору). Экран ПСО запускается в режиме работы нажатием клавиши «ПСО» вверху экрана. На рис 2.22 показано меню рабочего стола в режиме работы, а на рис.2.23. последовательность сообщений оператору как результат работы выполняемого задания.



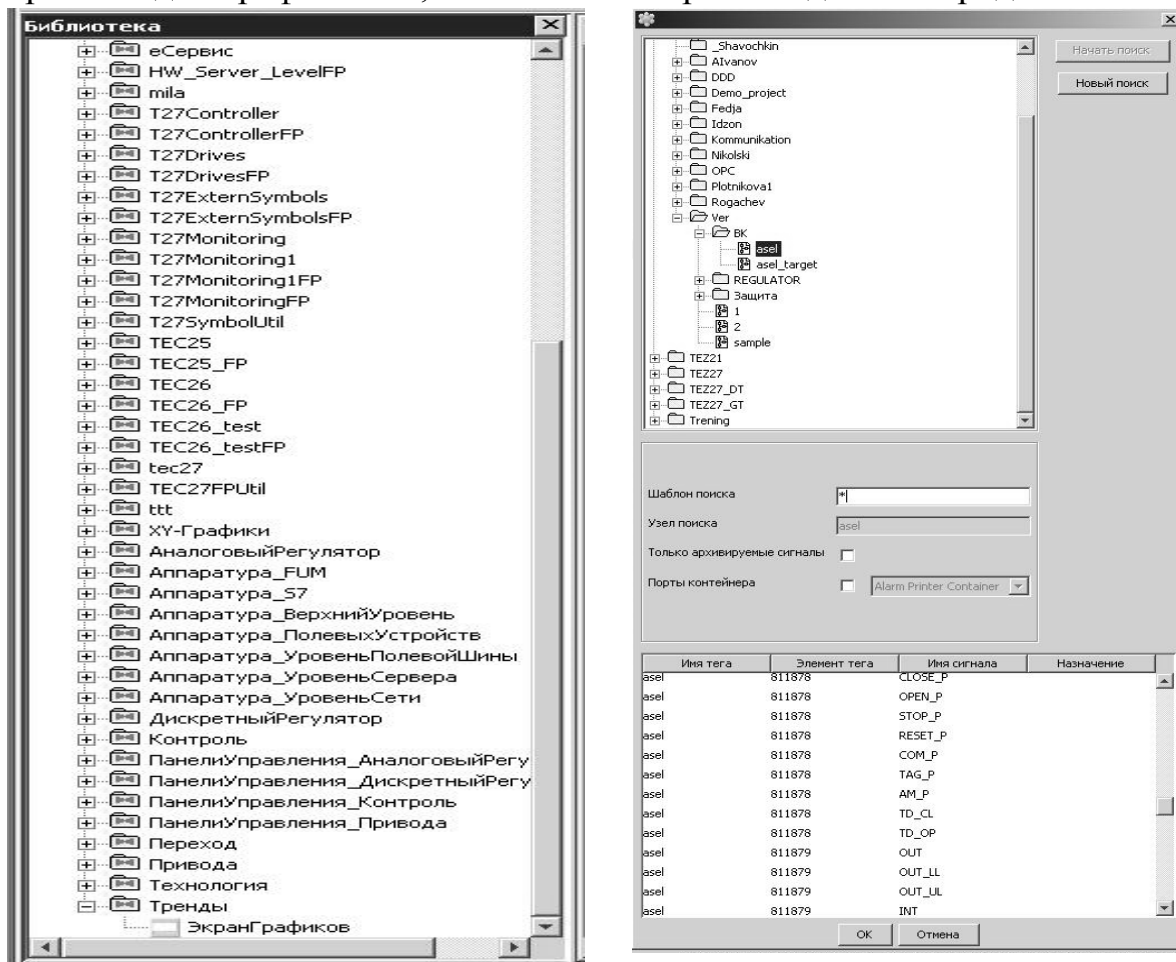
Рис.2.22. Меню рабочего стола в режиме работы.

Тип си...	Состо...	Кв...	Наименовани...	Тег	Назначение	Дат...	Вр...	Да...	Вре...	При...
АСУ
АСУ
АСУ
П	ПРЯЧЬСЯ		ПРЕДУПРЕДИТ...	БАК 10269 Q3	Предельный сигнал 3	2006/...	*08:5...			1
А	<300		АВАРИЙНАЯ	10LCA10FU001...	МОДЕЛЬ ОБЪЕКТА	2006/...	*08:5...			1
П	<600		ПРЕДУПРЕДИТ...	10LCA10FU001...	МОДЕЛЬ ОБЪЕКТА	2006/...	*08:5...			1
П	<20%		ПРЕДУПРЕДИТ...	10TST10CL001 ...	Предельный сигнал 3	2006/...	*08:5...			1
П			ПРЕДУПРЕДИТ...	БАК 10584 Q3	Предельный сигнал 3	2006/...	*08:5...			1

Рис.2.23. Последовательность сообщений оператору

7. Получение графиков изменения уровня

Графики отображают архивированные или текущие значения данных технологического процесса в виде линейной диаграммы. Ограничений по времени для графика нет; он может отображать данные продолительно-



стью от миллисекунд до нескольких лет. Операторы могут создавать группы графиков для быстрого доступа к информации о конкретных, заранее определенных точках.

Графики также можно встраивать в видеограммы в качестве компонентов; при этом они отображаются рядом с соответствующими процессами или оборудованием объекта.

Для работы с графиками создайте видеограмму с названием "График". Откройте созданную видеограмму и добавьте в нее прототип для графика из библиотеки прототипов Прототипы/Тренды/ЭкранГрафиков.

Перенесите экран графиков на пустое поле открытой видеограммы, для

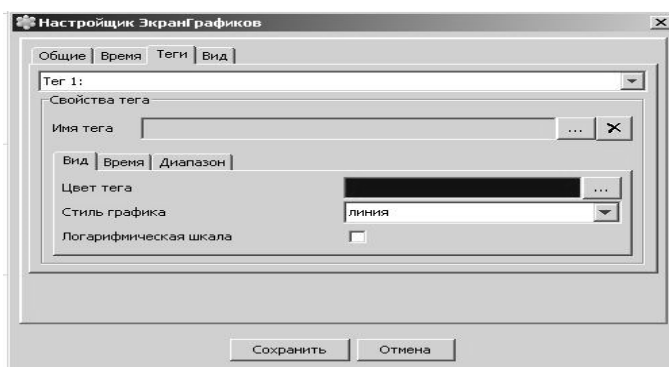


Рис. 2.26. Окно настройки графиков

выбора отображаемых на графике параметров выберете вкладку “Тренды”.

Щелкните на кнопке “...”. В появившемся окне выберите папку, в которой находится созданная функциональная схема, убедитесь, что в поле “Только архивируемые сигналы” нет флага. Нажмите кнопку “Начать поиск”. После окончания поиска в нижней части экрана появится таблица содержащая имена тегов, элементы тегов и имена сигналов. Обратите внимание на поля “Имя тегов”, “Элемент тега” и “Имя сигнала”, так как они однозначно указывают на сигнал функциональной схемы. Имя функциональ-

Тип	Имя	Значение >>	Видимый порт	Видимый па...	Архив >>	Сигнализация <<	...	Сообщение пр
BX	IN	0,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LV1	80,0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LSG1	Да	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	HI_LOW1	Высокий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LV2	50,0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LSG2	Да	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	HI_LOW2	Высокий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LV3	10,0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LSG3	Да	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	HI_LOW3	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LV4	1,0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LSG4	Да	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	HI_LOW4	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LV5	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LSG5	Нет	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	HI_LOW5	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LV6	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	LSG6	Нет	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	HI_LOW6	Низкий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
BX	DB	0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Рис.2.27. Определение уникального номера тега.

ной схемы отображается в поле “Имя тегов”. Поле “Элемент тега” является уникальным параметром. Что бы узнать “Элемент тега” для функционального блока необходимо вызвать функциональную схему и дважды щелкнуть по интересующему функциональному блоку. Имя тега будет указано в верхнем левом углу окна (числовое значение).

“Имя сигнала” указывает на выход функционального блока (например OUT)

Задание по формированию графиков:

Вывести на экран 2 графика.

1. Значение уровня в баке (ASMON - OUT).
2. Включение мотора (MOTOR - RUNNING).

Подтвердите внесенные изменения

Активируйте схемы

Подтвердите последнюю активации.

Откройте видеограмму “График” в режиме работы. Результаты работы составленной схемы проекта будут отражаться в открывшемся окне.

Для удобства работы с графиками следует изменить настройки длительности отображения.

Над областью отображения расположен список. Для открытия списка нажмите на стрелку справа от него.

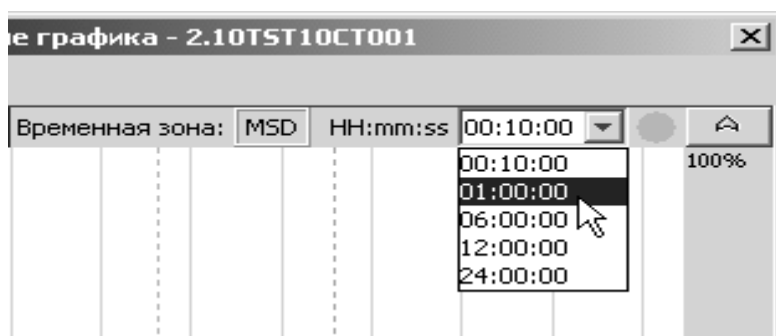
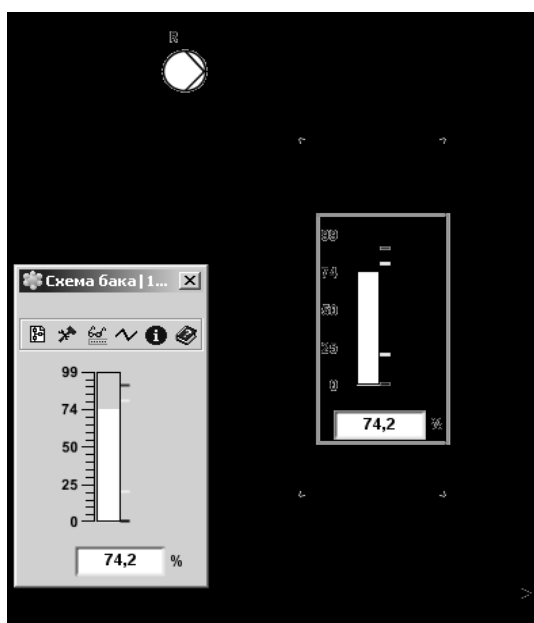
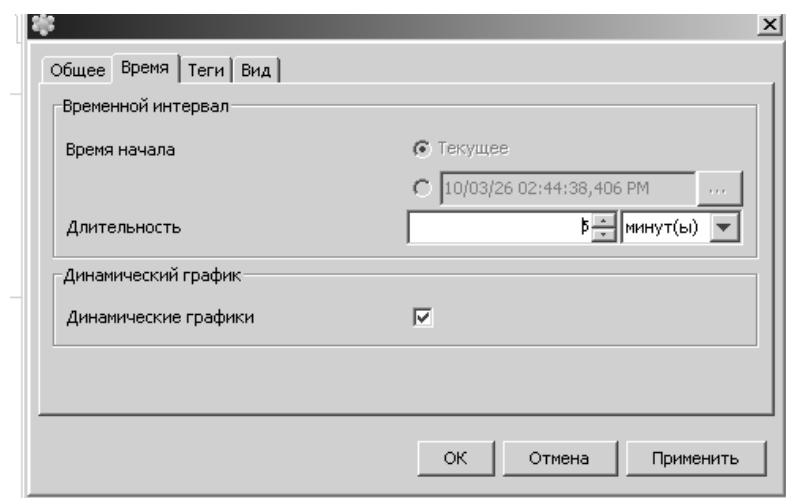


Рис. 2.28. Установка временных параметров графика



бака» с открытой панелью блока управления модулем SPADJ

этого необходимо дважды щелкнуть левой клавишей мыши на видеограмме сигнализатора SPADJ. На панели управления выбрать пункт «Вид точ-

Выберите длительность для области отображения 10 мин. Измените длительность отображения – 5 мин. Для этого нажмите кнопку «Редактирование конфигурации файла» на видеограмме «График» и в открывшемся окне выберите пункт «Время». Установите длительность 5 мин. И нажмите клавишу «ОК». Окно «Редактирование конфигурации файла» показано на рис. 2.29.

конфигурации файла

Последним этапом работы будет принудительное достижение аварийной уставки. Для этого в функциональной схеме откройте свойства функционального блока SPADJ отвечающего за расход воды поступающей в бак. (Дождаться пока задвижка будет закрыта и расход на сливе будет уменьшаться.)

Для достижения принудительного значения установите флажок в строке OUT столбца “Принудительно”, находящегося слева от столбца “Описание”. И выставите его значение 95. Наблюдать данные изменения на графике.

Такого же результата можно достичь, открыв панель управления функциональным блоком SPADJ, отвечающего за расход воды поступающей в бак. Для

ки» и в открывшемся окне установите флажок в строке OUT столбца “Принудительно”, находящегося слева от столбца “Описание”. Выставьте значение 95.

Установка предельного значения уровня воды в баке показана на рис. 2.31.

Тип	Имя	Значение	Качество	Временная отметка	Сигнализ...	Принудител...	Пр...	Описание
BX	UL	99,0	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	99,0	<input type="checkbox"/>	верхний предел
BX	LL	0,0	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	0,0	<input type="checkbox"/>	нижний предел
BX	EU	%	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	%	<input type="checkbox"/>	Единицы измерения
BX	RR_UP	5,0	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	1,0	<input checked="" type="checkbox"/>	Величина сдвига УВЕЛИЧ. в RR_UNIT/сек
BX	RR_DOWN	5,0	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	1,0	<input checked="" type="checkbox"/>	Величина сдвига УМЕНЬШАЕТСЯ в RR_UNIT/сек
BX	RR_UNIT	%	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	%	<input type="checkbox"/>	Единица величины сдвига
BX	BASE	сек	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	сек	<input type="checkbox"/>	Базовое время
BX	TRACK_V	0,0	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	0,0	<input type="checkbox"/>	Отслеживаемое значение
BX	C_TRACK	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Команда СЛЕЖЕНИЕ
BX	EXT_SETP	0,0	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	0,0	<input type="checkbox"/>	Внешнее задание
BX	A_EXT	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Ком.автоматики ВНЕШН.
BX	EN_EXT	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Разрешение ВНЕШН.
BX	A_INT	<input checked="" type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Ком.автоматики ВНУТРЕН.
BX	EN_INT	<input checked="" type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Разрешение ВНУТРЕН.
BX	A_UP	<input type="checkbox"/>	GOD	2008/04/11 11:16:40...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Ком.автоматики БОЛЬШЕ
BX	EN_UP	<input checked="" type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Разрешение БОЛЬШЕ
BX	A_DOWN	<input checked="" type="checkbox"/>	GOD	2008/04/11 11:16:40...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Ком.автоматики МЕНЬШЕ
BX	EN_DOWN	<input checked="" type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Разрешение МЕНЬШЕ
BX	P1_DOWN	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Защита 1 Команда МЕНЬШЕ
BX	P2_DOWN	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Защита 2 Команда МЕНЬШЕ
BX	P1_UP	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Защита 1 Команда БОЛЬШЕ
BX	P2_UP	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Защита 2 Команда БОЛЬШЕ
BX	A_STOP	<input type="checkbox"/>	GNC	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Ком.автоматики ОСТАНОВИТЬ Движение
ВЫХ	OUT	1,584	GOD	2008/04/11 11:17:19...	<input type="checkbox"/>	100,0	<input checked="" type="checkbox"/>	Выходное значение
ВЫХ	EXT	<input type="checkbox"/>	GOD	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Статус ВНЕШН.
ВЫХ	INT	<input checked="" type="checkbox"/>	GOD	2008/04/11 10:58:39...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Статус ВНУТР.
ВЫХ	TRACK	<input type="checkbox"/>	GOD	2008/04/11 10:52:23...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Режим СЛЕЖЕНИЯ
ВЫХ	RESET_P	<input type="checkbox"/>	GOD	2008/04/11 10:52:23...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Разрешен ДИСТ сброс

Рис. 2.31. Установка предельного значения уровня воды в баке

Все изменения в функциональных модулях и на графиках осуществляются только после нажатия клавиши «Применить» или «Сохранить».

На рис.2.32 показан примерный вид получаемого процесса двухпозиционного регулирования уровня в баке с помощью включения и отключения насоса, а также результат принудительного ввода значения уровня, превышающего значение аварийной уставки.

Таким образом, второй проект закончен и Вы значительно повысили свои знания по использованию системы SPPA-T3000 для построения функциональных схем, работы с элементарной графикой и привязками графических элементов к элементам логики и запуска ее в режиме проверки.

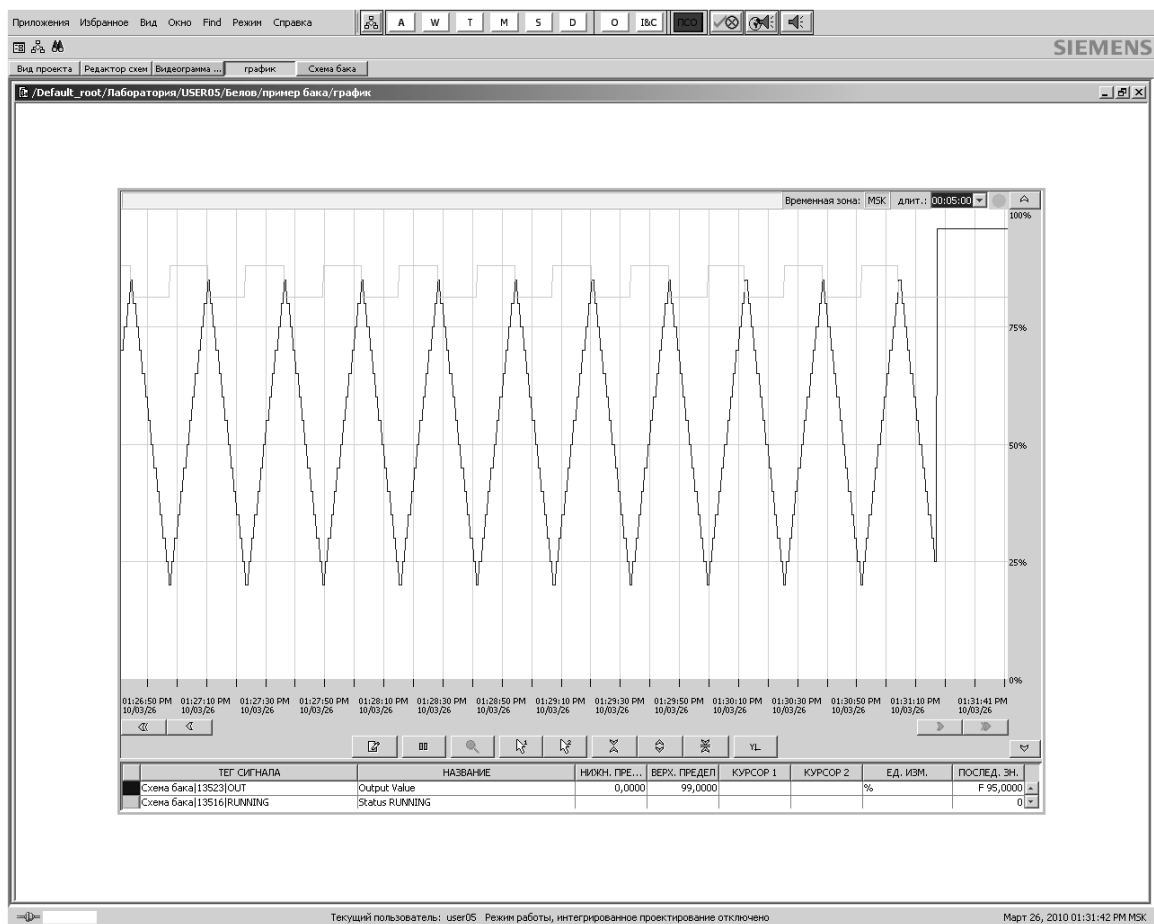


Рис.2.32. Графики заданных величин

Практическое занятие № 4

Проектирование каскадной системы регулирования в SPPA-T3000

Цель занятия: знакомство с системой проектирования, получение навыков работы и самостоятельная реализация каскадной системы регулирования.

Теоретические основы

Если одноконтурная система регулирования с ПИД-регулятором недостаточно эффективно подавляет действие возмущений, то это скорее всего свидетельствует о необходимости усложнения информационной структуры системы, т.е. о необходимости введения добавочных каналов связи регулятора с объектом. Объясняется это тем что во всякой информационной структуре (естественно – и в одноконтурной) имеется предел достижимой точности регулирования, обусловленной в частности, наличием запаздывания в регулирующем канале объекта. Преодолеть этот предел, оставаясь в рамках той же структуры, принципиально невозможно никаким усовершенствованием алгоритмов регулирования. Так, среднеквадратичное отклонение регулируемой величины в системе с оптимально настроенным регулятором является минимально возможным в одноконтурной структуре системы с запаздыванием в регулирующем канале объекта. Естественно, что если предельно допустимая точность не удовлетворяет требованиям

технологической работоспособности системы, одноконтурная система становится принципиально неприменимой, сколь бы совершенным ни был алгоритм работы регулятора. Именно такая ситуация возникает чаще всего на практике, когда качество работы одноконтурной системы с ПИД-регулятором оказывается неудовлетворительным.

Одним из способов усложнения информационной структуры системы является ввод в регулятор добавочной информации об изменении некоторых специально подобранных величин более оперативно, чем управляемая величина, характеризующих изменение текущего состояния объекта – такие системы называются системами со вспомогательными регулируемыми величинами. Системы со вспомогательными регулируемыми величинами – это многоконтурные системы, поскольку введение каждой вспомогательной величины образует новый контур.

Два основных практически применяемых варианта структурной схемы двухконтурной системы показаны на рис.3.1 и рис.3.2.

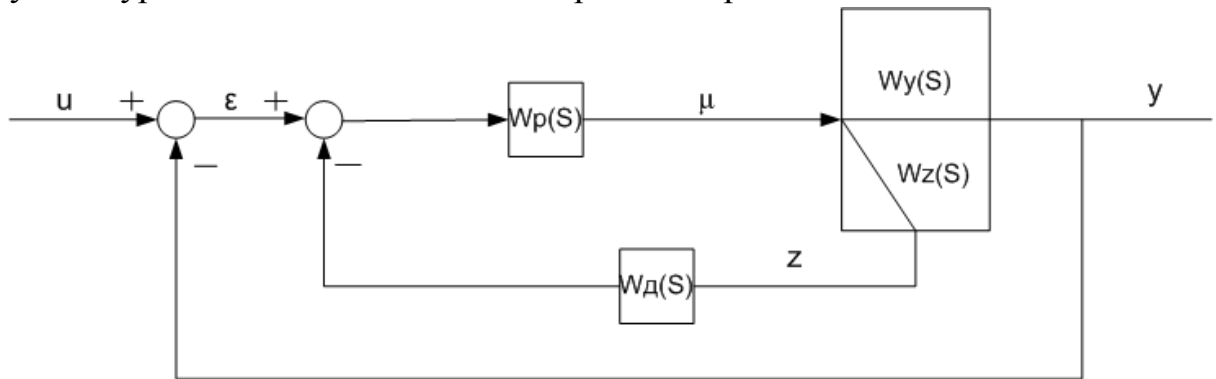


Рис.3.1. Двухконтурная схема с дифференциатором.

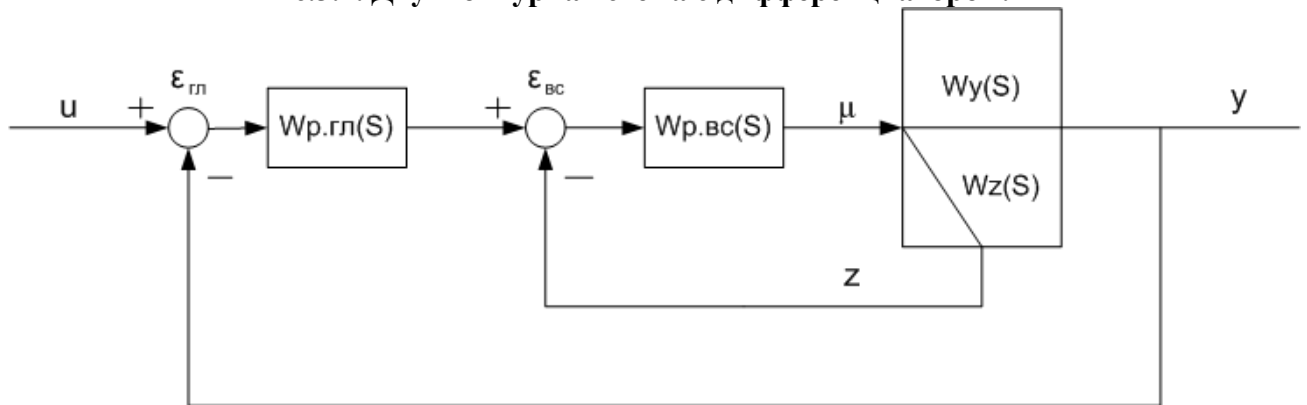


Рис.3.2. Двухконтурная схема с двумя регуляторами.

Схема, изображенная на рис.3.1 имеет только один регулятор с передаточной функцией $W(S)$, но на его вход подается добавочный сигнал от вспомогательной регулируемой величины, предварительной преобразованной в блоке дифференцирования с передаточной функцией $Wд(S)$. Необходимость дифференцирования вспомогательной регулируемой величины прежде всего определяется требованиями к статике системы регулирования – вспомогательный сигнал на входе в регулятор должен исчезнуть по окон-

чании процесса регулирования, так как в противном случае регулятор будет поддерживать не заданное значение регулируемой величины, а сумму заданного значения и вспомогательной регулируемой величины. Кроме того, дифференцирование вспомогательного сигнала может также способствовать улучшению динамики системы регулирования.

Другой вариант двух контурной системы приведен на рис.3.2. Эта система имеет два соподчиненных регулятора: вспомогательный регулятор внутреннего контура стабилизирует вспомогательную величину $z(t)$, а командное воздействие этому регулятору формирует регулятор внешнего контура, который называют корректирующим или главным. На схеме приняты обозначения: $W_y(S)$ – передаточная функция объекта по каналу основной регулируемой величины, $W_z(S)$ – передаточная функция объекта по каналу вспомогательной регулируемой величины, $W_{p.гл}(S)$ – передаточная функция главного регулятора, $W_{p.вс}(S)$ – передаточная функция вспомогательного регулятора.

Задание

Спроектировать каскадную систему регулирования температуры перегретого пара с двумя ПИ-регуляторами. Основным регулирующим воздействием является $D_{ВПР}$ – расход впрыска. Основной регулируемой величиной является $\theta_{ПП}$ – температура перегретого пара, вспомогательной регулируемой величиной $\theta_{ВПР}$ - температура впрыска.

Передаточные функции объекта по основному и вспомогательным каналам имеют вид:

$$W_{\mu y}(s) = \frac{e^{-5s}}{(76s + 1)(1 + 23.2s)^2} \quad (1)$$

$$W_{\mu z}(s) = \frac{(77.7s + 1)}{(104s + 1)(1 + 8.09s)^2} \quad (2)$$

Для выполнения данного примера в SPPA-T3000 необходимо выполнить следующие действия:

1. Зарегистрироваться в системе и открыть рабочий стол.
2. Создать вид проекта и дополнительную структуру проекта.
3. Создание шаблонов для функциональной схемы и видеограммы.
4. Ввод информации в функциональную схему.
5. Ввод информации в видеограмму.
6. Назначить исполняемый контейнер и запустить пример на выполнение в эмуляторе.

Выполнение задания

1. Создание узла проекта.

В папке со своей фамилией создайте новый узел и переименуйте его в «Каскадная система» (рис.3.3).

Если у вас нет своей папки, то необходимо создать ее, проделав следующие операции:

Выберите меню **Проект**, затем **Новый**, затем **Узел проекта** .

Создан новый узел проекта с именем “**Новый Узел**”, под корневым узлом.

Затем новый узел проекта может быть переименован, для этого :
выберите узел “**НовыйУзел**” в дереве объектов, меню **Проект**, затем **Переименовать**. На имени узла появится курсор. Удалите имя старого узла и введите свою фамилию затем нажмите на Enter.

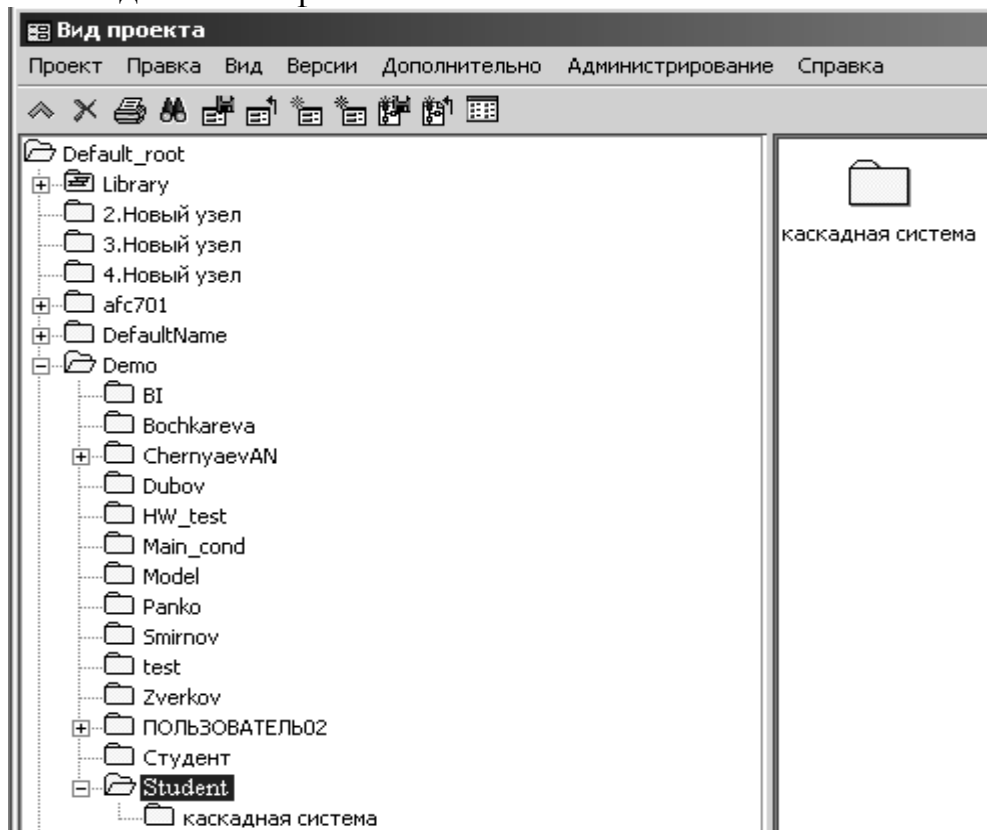


Рис.3.3. Создание узла проекта.

2. Создание шаблонов для элементов проекта.

3.1. Создание шаблона функциональной схемы.

В узле «Каскадная система» создайте функциональную схему, для этого щелчком правой клавиши по пустому полю вызовите контекстное меню и выберете пункт «Создать новую функциональную

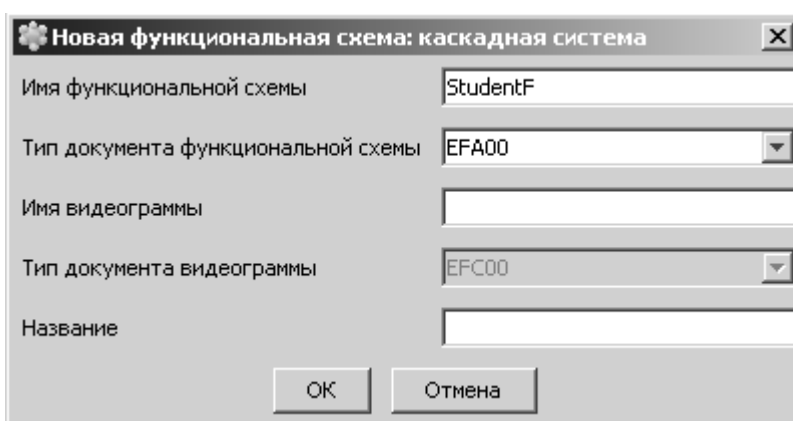


Рис.3.4. Диалоговое окно “Новая функциональная схема”

схему» Откроется диалоговое окно “**Новая функциональная схема**”. Введите в поле “**Имя функциональной схема**” свою фамилию и на конце добавьте букву F, как показано на рис.3.4. Поле **Видеограмма** оставьте пустым.

Нажмите на кнопку **ОК**. Создается пустой шаблон для функциональной схемы.

3.2. Создание шаблона видеограммы

В узле «Каскадная система» создайте видеограмму, для этого щелчком правой клавиши по пустому полю вызовите контекстное меню и выберите пункт «Создать новую видеограмму». Откроется диалоговое окно «**Новая видеограмма**». Введите в поле «**Имя видеограммы**» свою фамилию и на конце добавьте букву V, как показано на рис.3.5. Поле **Функциональная схема** оставьте пустым.

Нажмите на кнопку **ОК**. Создается пустой шаблон для видеограммы.

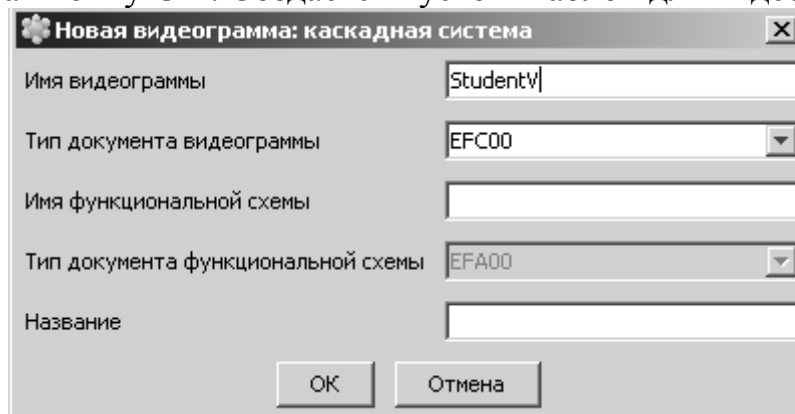


Рис.3.5. Диалоговое окно «Новая видеограмма»

Таким образом, вы создали два пустых шаблона для функциональной схемы и видеограммы. Эти два шаблона находятся в узле «Каскадная система», как показано на рис.3.6.

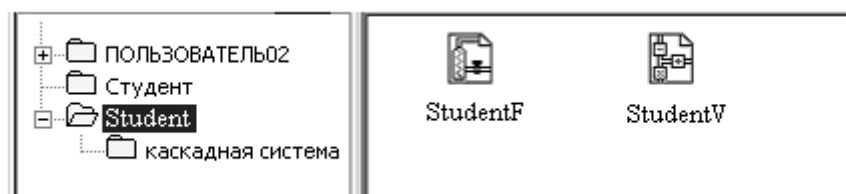


Рис.3.6. Окно Вид проекта с созданными схемами

4. Ввод информации в функциональную схему.

Откройте функциональную схему в режиме конфигурирования. Для этого выделите функциональную схему, выберите меню **Правка**, затем пункт меню **Открыть в режиме конфигурирования**. Редактор схем запускается и функциональная схема «**StudentF**» открывается (схема является пустой, так как конфигурация еще не была выполнена).

Соберите схему в соответствии с рис.3.10.

Модель объекта по каналу основной регулируемой величины, соответствующая передаточной функции (1) и модель объекта по каналу вспомогательной регулируемой величины, соответствующая передаточной функции (2), представлены на рис.3.8. и рис.3.9. соответственно.

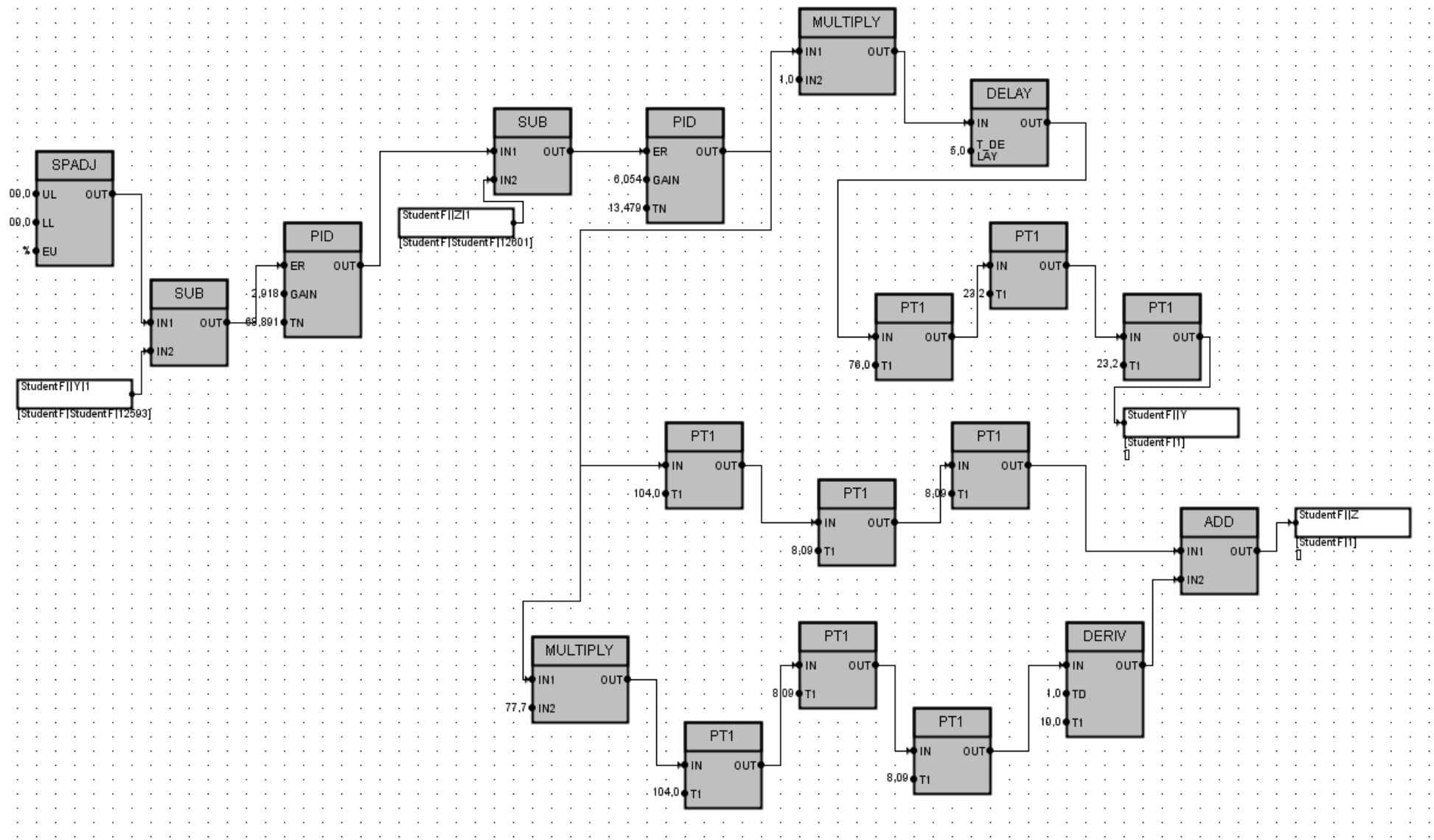


Рис.3.10. Структурная схема каскадной системы регулирования.

Описание и пути для элементов, используемых в схеме

1. **SPADJ** (*/LIBRARY/AUTOMATION/CLOSED LOOP CONTROLLER*) – Задатчик SPADJ.

Функция Задатчик используется для управления и отслеживания значения уставки. У нее есть три режима работы:

- Внутренний режим — автоматические команды и подаваемые вручную команды регулируют величину выходного значения с помощью интегратора;
- Внешний режим — внешняя уставка пересылается на выход;
- Режим слежения — выход повторяет отслеживаемое значение, этот режим отменяет другие режимы.

2. **SUB** (*/LIBRARY/AUTOMATION/ANALOG CALC.*) - Функция вычитания выполняет арифметическое вычитание двух аналоговых значений и предоставляет результат (разница) на выходе.

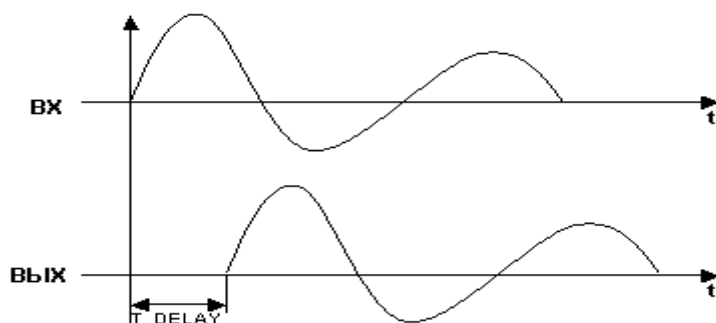
Значение на выходе (разница) получается в соответствии со следующим правилом вычисления: $ВЫХ = ВХ1 - ВХ2$.

3. **ADD** (*/LIBRARY/AUTOMATION/ANALOG CALC.*) – Функция сложения выполняет арифметическое сложение двух аналоговых значений и предоставляет результат (сумма) на выходе.

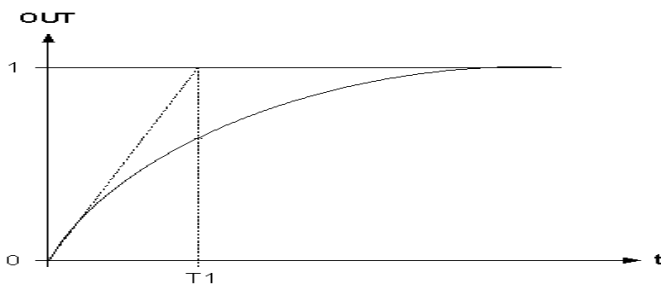
Значение на выходе (сумма) получается в соответствии со следующим правилом вычисления: $ВЫХ = ВХ1 + ВХ2$.

4. **MULTIPLY** (*/LIBRARY/AUTOMATION/ANALOG CALC.*) - Функция умножения выполняет арифметическое умножение двух входных значений и предоставляет результат (произведение) на выходе.

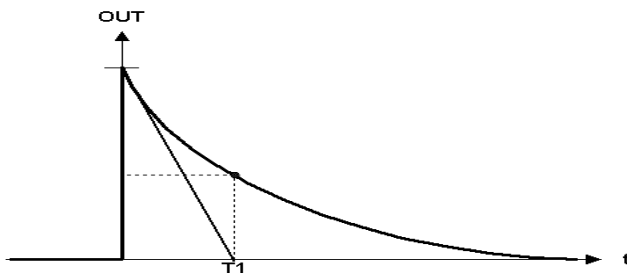
5. **DELAY** (*/LIBRARY/AUTOMATION/CLOSED LOOP CONTROLLER*)- Функция задержки обеспечивает выходное значение, следующее за входным с настраиваемой временной задержкой.



6. **PT1** (*/LIBRARY/AUTOMATION/ANALOG CALC.*) - Функция задержки первой команды выполняет задержку нарастания сигнала на постоянную времени T1. Аperiodическое звено.



7. **DERIV** (/LIBRARY/AUTOMATION/ANALOG CALC.) – функция дифференцирования использует правило трапеции для определения скорости изменения аналогового сигнала. Звено дифференцирования.



1. **PID** (/LIBRARY/AUTOMATION/CLOSED LOOP CONTROLLER) – Контроллер.

Контроллер PID имеет два основных режима работы:

- **Режим регулирования с обратной связью**

Контроллер PID использует активный алгоритм.

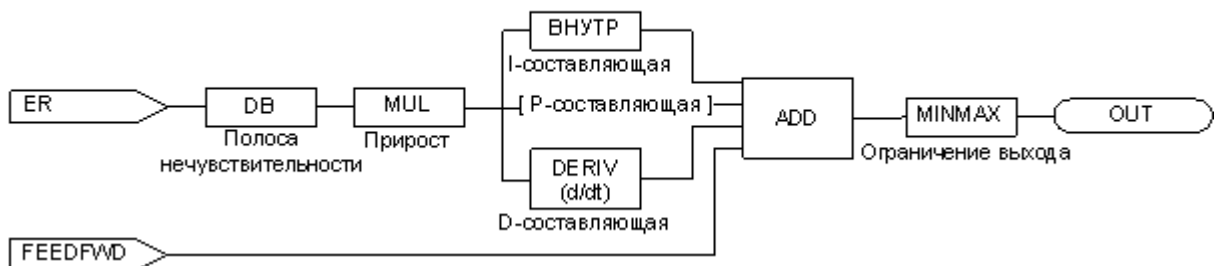
- **Режим слежения**

В этом режиме контроллер PID передает входные значения TRACK_V на выходы OUT.

Принцип режима регулирования с обратной связью

Принцип работы в этом режиме показан на следующей схеме.

Контроллер PID Режим ЗАМКНУТОГО КОНТУРА:



После прохождения сигналом ошибки (отклонения) ER установленной полосы нечувствительности DB сигнал умножается на значение GAIN. Результатом является пропорциональное значение (P-составляющая), которое также является базовым для вычисления интегрального (I-составляющая) и производного (D-составляющая) значения. Роль каждого значения контролируется параметрами как описано ниже.

Полоса нечувствительности

Если отклонение сигнала ER находится внутри полосы нечувствительности DB, значение этого сигнала равно нулю.

P-составляющая

P-составляющая равна отклонению сигнала, умноженному на значение GAIN.

P-составляющая может быть деактивирована установкой сигнала P_OFF в значение ИСТИННО.

I-составляющая

I-составляющая изменяется пропорционально усиленному сигналу ER.

I-составляющая может быть деактивирована сигналом I_OFF.

D-составляющая

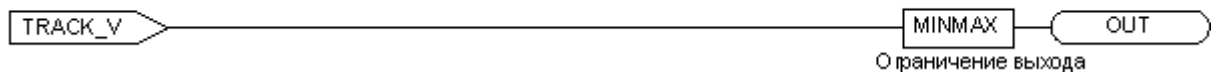
D-составляющая вычисляет скорость изменения усиленного сигнала отклонения ER.

D-составляющая может быть деактивирована сигналом D_OFF.

РЕЖИМ СЛЕЖЕНИЯ

Режим слежения активируется установкой значения ИСТИННО для C_TRACK. В этом режиме контроллер PID передает входные значения TRACK_V на выходы OUT. Выходное значение ограничено величинами UL и LL.

Контроллер PID Режим СЛЕЖЕНИЯ



Примечание: более подробное описание функциональных блоков можно найти в справке к SPPA-T3000.

Коннекторы

После того, как все необходимые блоки перенесены в поле функциональной схемы, необходимо соединить их, как показано на рис.3.11. Для упрощения схемы SPPA позволяет разорвать соединения с помощью коннекторов сигнала. Коннекторы сигнала делятся на *Коннекторы источника* и *Коннекторы назначения*.

В приведенном ниже примере было установлено внешнее соединение между двумя функциональными схемами. *Коннектор источника* (рис.3.11) соединен с источником сигнала, *коннектор назначения* (рис.3.12) соединен с назначением сигнала. Коннекторы содержат информацию о сигнале.

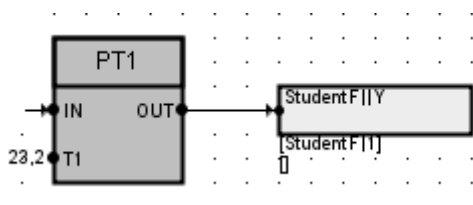


Рис.3.11. Коннектор источника

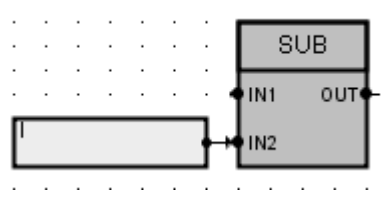



Рис.3.12. Коннектор назначения

Для создания *коннектора источника* выполните следующие действия. Откройте функциональную схему в редакторе схем. В схеме должен быть расположен символ источника сигнала.

1. Щелкните по значку **Соединение (Обработчик внешних коннекторов)** .
2. Наведите курсор на блок источника. Будут выделены доступные для соединения порты.
3. Щелкните по порту источника. Выбранный порт будет выделен.
4. Наведите курсор на схему. Вместе с курсором перемещается блок, обозначающий контур коннектора источника.
5. Щелкните мышью, когда коннектор источника займет нужное положение.

Коннектор может быть отредактирован. Для этого

1. Выберите коннектор источника, который требуется отредактировать.
2. Откройте маску коннектора.
3. Откроется диалоговое окно **«Редактировать определение сигнала»**. На рис.3.13. приведен пример диалогового окна

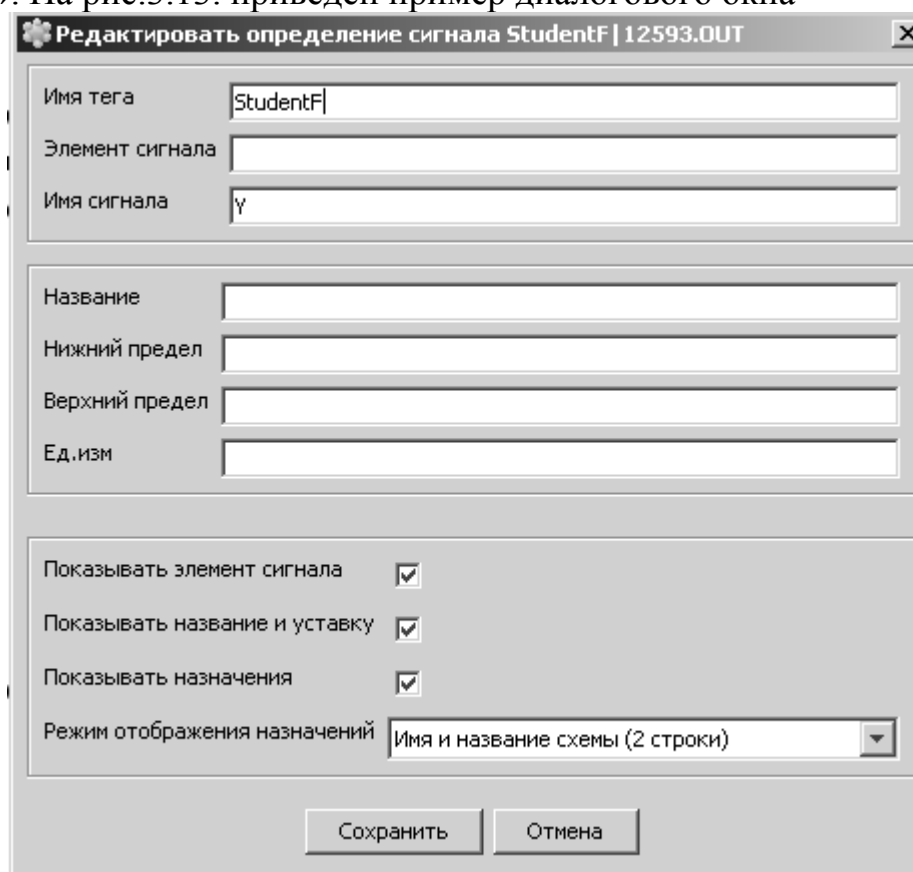



Рис.3.13. Диалоговое окно «Редактировать определение сигнала»

Для создания коннектора назначения выполните следующие действия. Откройте функциональную схему в редакторе схем. В схеме должен быть расположен символ назначения сигнала.

1. Щелкните по значку **Соединение (Обработчик внешних коннекторов)** .
2. Наведите курсор на блок назначения. Будут выделены доступные для соединения порты.

3. Щелкните по порту назначения. Выбранный порт будет выделен.
4. Наведите курсор на схему. Вместе с курсором перемещается блок, обозначающий контур коннектора назначения.
5. Щелкните мышью, когда коннектор назначения займет нужное положение.

При создании коннектора назначения определение сигнала создается пустым и может быть отредактировано. Для редактирования необходимо:

1. Выберите коннектор назначения, предназначенный для редактирования.
2. Откройте маску коннектора.
3. Откроется диалоговое окно «**Редактировать сигнал источника**». На рис.3.14 приведен пример диалогового окна/

Редактировать сигнал источника StudentF | 12586.IN2

Имя тега: StudentF

Элемент сигнала:

Имя сигнала: γ

Название:

Нижний предел:

Верхний предел:

Ед.изм:

Показывать элемент сигнала:

Показывать название и уставку:

Показывать источник:

Способ отображения источника: Имя схемы и имя модуля (1 строка)

Сохранить Отмена

Рис.3.14. Диалоговое окно «Редактировать сигнал источника»

4. Если редактируемый сигнал подключен, т. е. существует соответствующий коннектор источника, то выполняются следующие условия.
 - Если изменено поле в определении сигнала (имя тега, элемент сигнала или имя сигнала), соединение будет открыто.
 - Установки сигнала (назначение, верхний или нижний предел, единицы измерения или настройка) изменить нельзя, они зависят от коннектора источника.
5. Если сигнал не подключен, то есть не существует соответствующего коннектора источника, то можно изменить любое поле без влияния на коннекторы источников.

6. Поля можно редактировать либо вручную, либо используя **Просмотрщик сигналов** для поиска и подключения к существующему источнику сигнала.

1. Нажмите на кнопку справа от поля **Имя тега**.



2. Откроется просмотрщик сигналов. (рис.3.15).

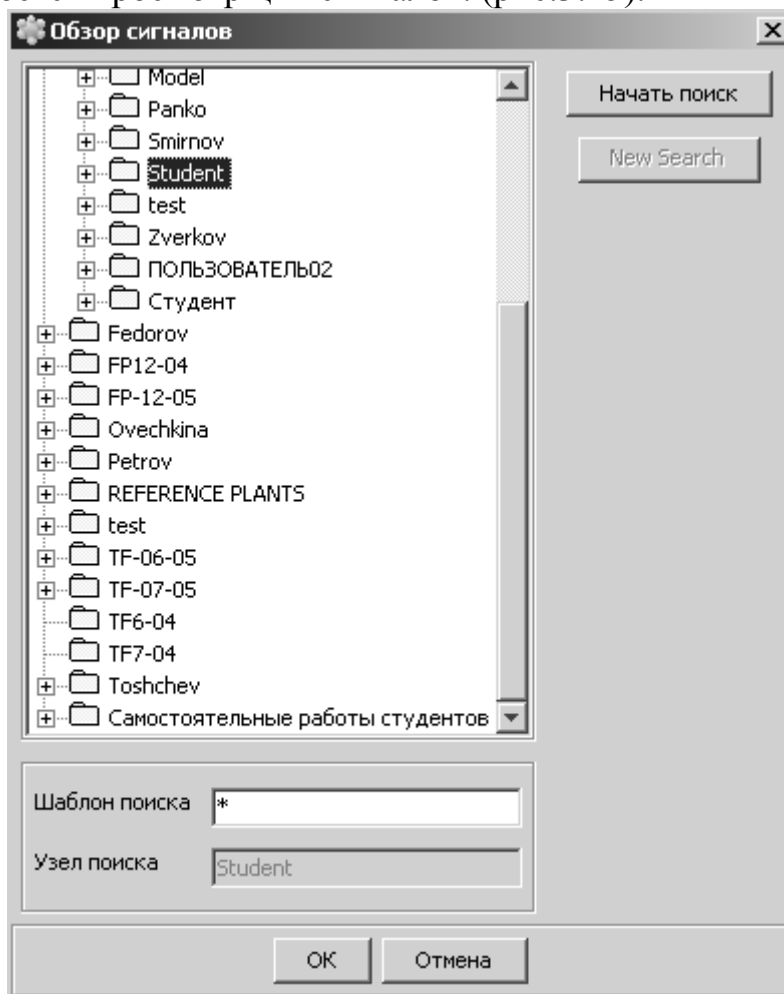


Рис.3.15. Просмотрщик сигналов

3. В левой рамке диалогового окна просмотрщика сигналов выберите узел проекта или схему, поиск которых будет осуществляться (при необходимости перемещаясь по дереву). Узел или схема появится в поле **Узел поиска**.

4. Введите шаблон поиска в поле **Шаблон поиска**. Этот шаблон соответствует **имени тега** сигнала. Шаблон поиска может содержать звездочку (*) для поиска по маске.

5. Нажмите на кнопку **“Начать поиск”**.

6. Найденные сигналы выводятся в нижней части диалогового окна (если сигналы не найдены, будет выведено соответ-

ствующее сообщение). В любой момент можно начать новый поиск нажатием на кнопку “Новый поиск”.

Имя тега	Элемент тега	Имя сигнала	Название
StudentF		Z	
StudentF		Y	

7. Выберите требуемый сигнал из списка, щелкнув по нему.
 8. Нажмите на кнопку **ОК**. Поля в диалоговом окне **Редактировать сигнал источника** будут заполняться автоматически.
7. Для сохранения любых изменений нажмите на **Сохранить**, для отказа от них нажмите на **Отмена**.

Внести информацию в блоки модели объекта в соответствии с передаточными функциями (1) и (2) и рис.3.8 и 3.9.

Произвести настройку корректирующего и стабилизирующего регуляторов, как это показано на рис.3.16 и рис.3.17 соответственно. Для открытия маски параметров выберите блок регулятора. Щелчком правой кнопки мыши откройте контекстное меню. В контекстном меню выберите пункт “Редактировать”.

Символ StudentF | 12587

Свойства Набор параметров 1 Набор параметров 2

Тип	Имя	Значение >>	Видимый порт	Видимый параметр	Архив >>	Сигнал >>
BX	ER	0,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	DB	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	DB_OFF	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	FEEDFWD	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	TRACK_V	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	C_TRACK	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	P_OFF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	GAIN	2,918	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	I_OFF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	TN	68,891	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	D_OFF	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	TD	1,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	T1	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	UL	100,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	LL	-100,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	ANTI_W	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	EN_UP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	EN_DOWN	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ВЫХ	OUT	0,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ВЫХ	OUT_UL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ВЫХ	OUT_LL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ВЫХ	TRACK	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ВЫХ	CF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Рис.3.16. Маска настройки корректирующего регулятора.

Символ StudentF 12613						
Свойства		Набор параметров 1		Набор параметров 2		
Тип	Имя	Значение >>	Видимый порт	Видимый параметр	Архив >>	Сигнал >>
BX	ER	0,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	DB	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	DB_OFF	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	FEEDFWD	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	TRACK_V	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	C_TRACK	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	P_OFF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	GAIN	6,054	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	I_OFF	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	TN	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	D_OFF	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	TD	1,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	T1	0,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	LUL	1000,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	LL	-1000,0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	ANTI_W	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	EN_UP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BX	EN_DOWN	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BbX	OUT	0,0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BbX	OUT_UL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BbX	OUT_LL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BbX	TRACK	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BbX	CF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Рис.3.17. Маска настройка стабилизирующего регулятора

Ввод информации в видеограмму

Откройте пустую видеограмму StudentV и добавьте в нее прототип для графика из библиотеки прототипов Прототипы/Тренды/ЭкранГрафиков (рис.3.18). Это окно выбора экрана графиков.

Перенесите экран графиков на пустое поле открытой видеограммы, для выбора отображаемых на графике параметров в режиме редактирования выберете вкладку “Теги”(рис.3.19).

Щелкните на кнопке “...”. В появившемся окне (рис.3.20) выберите папку, в которой находится созданная функциональная схема, убедитесь, что в поле “Только архивируемые сигналы” нет флага. Нажмите кнопку “Начать поиск”.



Рис.3.18. Экран выбора экрана графиков

После окончания поиска в нижней части экрана появится таблица содержащая имена тегов, элементы тегов и имена сигналов. Обратите внимание на поля “Имя тегов”, “Элемент тега” и “Имя сигнала”, так как они однозначно указывают на сигнал функциональной схемы. Имя функциональной схемы отображается в поле “Имя

тегов”. Поле “Элемент тега” является уникальным параметром. Что бы узнать “Элемент тега” для функционального блока необходимо вызвать функциональную схему и дважды щелкнуть по интересующему функциональному блоку будет указано в верхнем левом углу окна.

“Имя сигнала” указывает на выход функционального блока (например OUT).

Выведите на экран 3 графика:

- Задание (SPADJ – OUT).
- Изменение основной регулируемой величины. (Y)
- Изменение вспомогательной регулируемой величины (Z).

Также на панель управления необходимо вывести панель управления задатчиком eSPADJ_FP1 (*LIBRARY/PROTOTYPES/eCloseLoopControllerFP*) (рис.21) и панели управления регуляторами PID_FP (*LIBRARY/PROTOTYPES/CloseLoopControllerFacePlates*) (рис.22). Общий вид рабочего экрана видеогаммы **StudentV** представлен на рис.20.

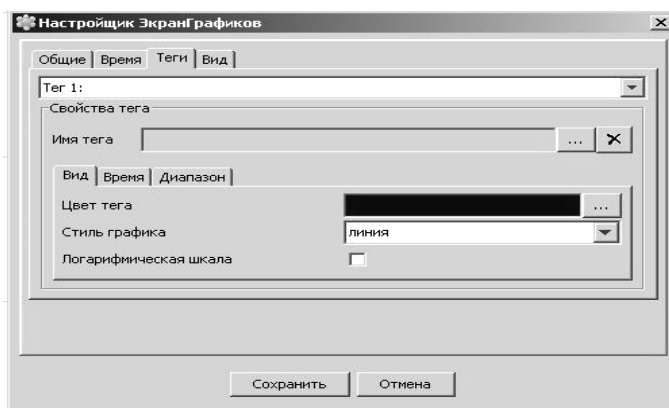


Рис.3.19. Окно настройки графиков

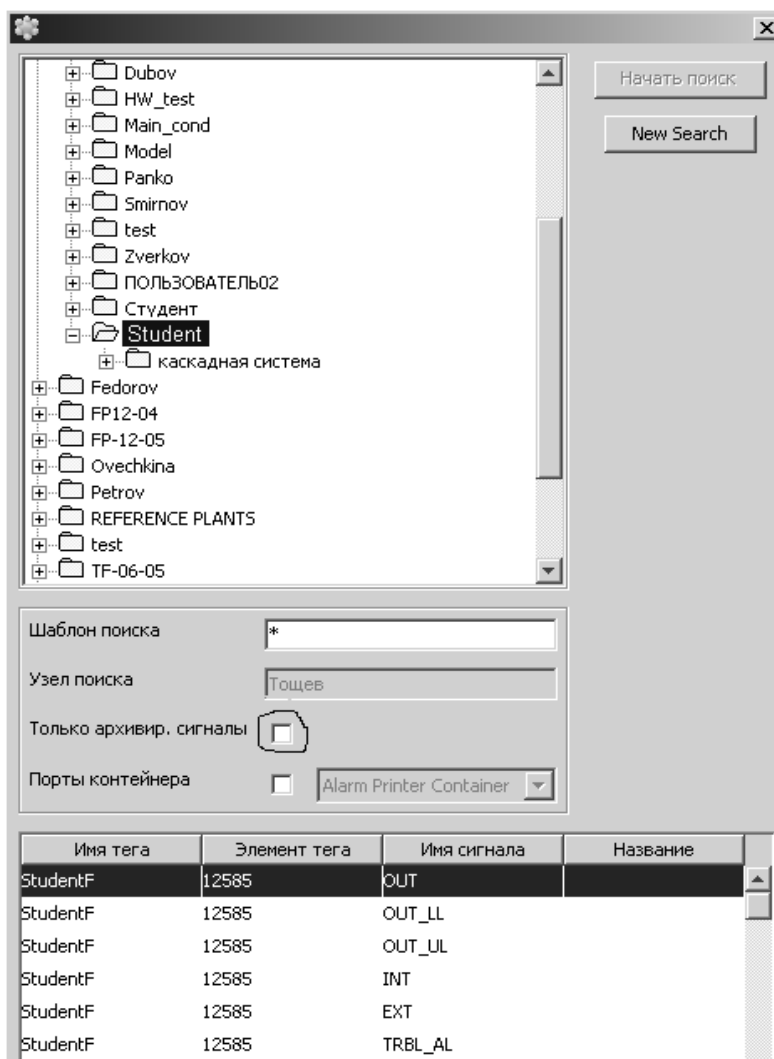


Рис.20. Окно назначения сигналов графикам.

Описание индикаторов и кнопок команд

Обозначение аналоговых величин:

- 3. и 4. **OUT (ВЫХ)**: текущее значение выходного сигнала, выводимое в виде шкалы и цифрового значения.
- 5. **INT SP (ВНУТР ЗД)**: значение внутренней настраиваемой уставки задатчика.
- 6. **EXT SP (ВНЕШ ЗД)**: значение, принятое на вход от внешнего источника.
- Состояние (индикация состояния осуществляется посредством символа пиктограммы на панели управления):

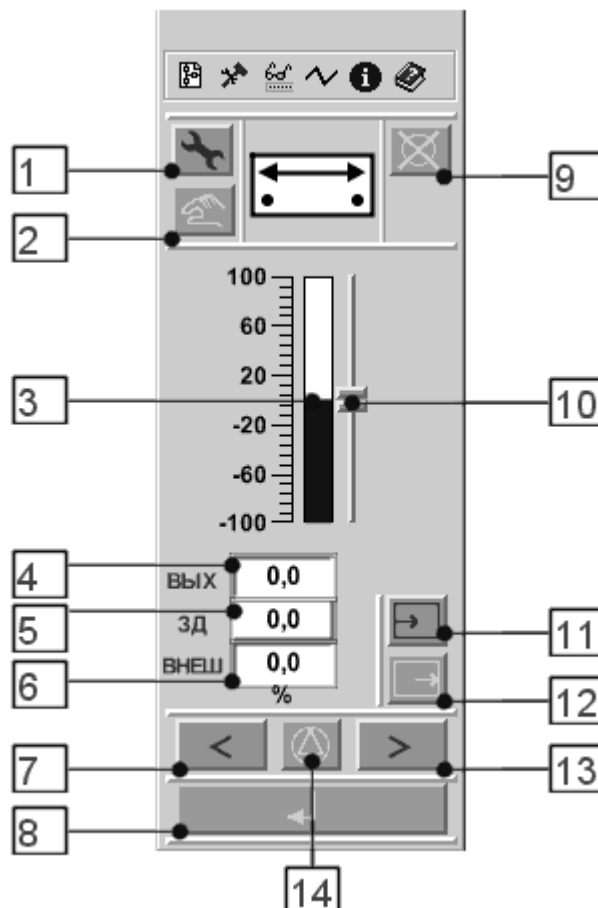


Рис.3.21. Панель управления задатчиком

- **СИГНАЛИЗАЦИЯ:**

Сигнализация ошибки запускается при срабатывании защиты, при ошибке вычисления или при переходе во внутренний режим, вызванном внешней ошибкой.

Кнопки команд

- 13, 7 и 14 **UP / DOWN / STOP (ВВЕРХ) / (ВНИЗ) / (СТОП)**: Для изменения текущей уставки используйте кнопки ВВЕРХ или ВНИЗ, а затем кнопку 8 **ВЫПОЛН**. Сигнал будет изменяться с установленной скоростью перехода до тех пор, пока не будет нажата кнопка **СТОП**. Команда **СТОП** будет выполнена незамедлительно без дополнительного нажатия **ВЫПОЛН**.
- 12. **INT (ВНУТР)**: Во внутреннем режиме уставка может быть выставлена в панели управления при помощи кнопок 13 **ВВЕРХ**, 7 **ВНИЗ** или элемента 10 (**бегунок**), если включен режим ручного управления.
Если включен автоматический режим, задатчик подчиняется только командам автоматики.
- 11. **EXT (ВНЕШ)**: во время внешнего режима задатчик управляется сигналом с внешнего источника.
- 2. **MANUAL / AUTO (ВРУЧНУЮ / АВТО)**: переключатель между автоматическим и ручным режимами управления.

В ручном режиме управление задатчиком осуществляется только по командам, поданным вручную.

В автоматическом режиме управление осуществляется только по автоматическим сигналам.

- 8. **EXECUTE (ВЫПОЛН)**: Подтверждает запуск команд на исполнение.
- 1. **COM (НАЛАДКА)**: В режиме наладки (возможность перехода в этот режим зависит от имени пользователя) принимаются поданные вручную команды, даже если активен сигнал разрешения или присутствует сигнал защиты.
- 9. **RESET (СБРОС)**: Индикатор блокировки внутренней логики для ручных и автоматических команд, которая происходит вследствие срабатывания внешней или внутренней защиты. Для отмены блокировки нажмите кнопку СБРОС, а затем **ВЫПОЛН**. Если причина срабатывания исчезла (блокировка снята), индикация будет снята, иначе индикация останется без изменений, а сброс будет невозможен до тех пор, пока причина не будет устранена.

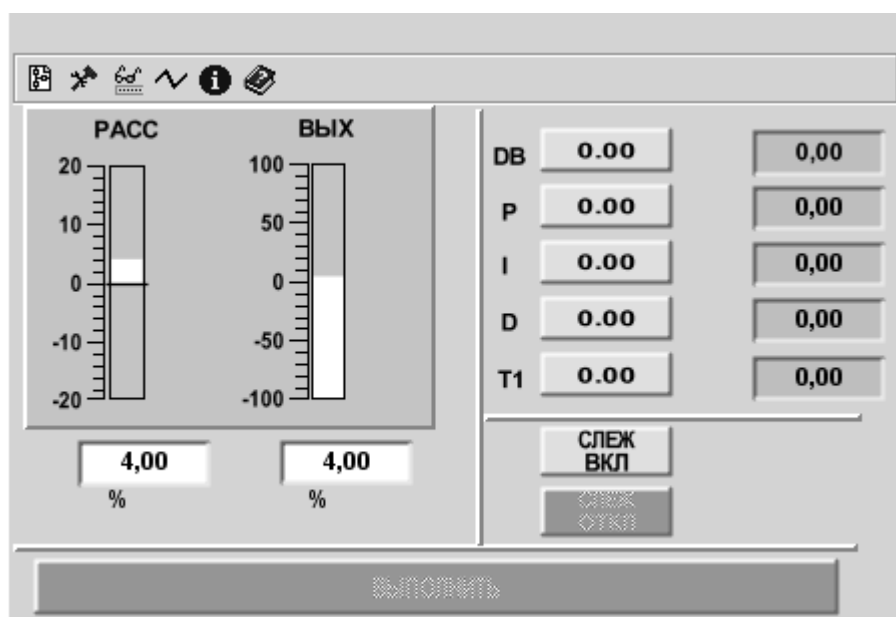


Рис.3.22. Панель управления регулятором.

Описание индикаторов и кнопок команд

- Отображение аналогового значения:
 - **ВЫХ**: текущее значение сигнала выхода отображается в виде линейки и цифрового значения.
 - **ОТКЛ**: текущее значение сигнала об ошибке на входе контроллера, которое отображается в виде линейки и цифрового значения.
- Кнопки команд:
 - **СЛЕЖ ВКЛ / СЛЕЖ ОТКЛ**: Позволяет осуществить выбор нужного режима слежения.

В режиме слежения управляемая переменная следует за сигналом слежения от внешнего источника. Внутренний ограничитель устанавливает верхний и нижний пределы значения сигнала.

- **ПОЛОСА НЕЧУВСТВ**: Позволяет настраивать полосу нечувствительности для отклонения сигнала. До тех пор, пока текущее значение отклонения, как показано на шкале **ОТКЛ**, ниже значения полосы нечувствительности (DB), PID-контроллер не меняет текущий выходной сигнал. Если текущее отклонение превышает значение полосы нечувствительности (DB), выходной сигнал меняет свое значение.
- **P**: Позволяет выполнить настройку коэффициента увеличения (пропорциональная часть) функции PID.
- **I**: Позволяет выполнить настройку времени интегрирования функции PID.
- **D**: Позволяет выполнить настройку разницы времени (производная часть) функции PID.
- **T1**: Позволяет выполнить настройку времени задержки функции PID.

6. Назначить исполняемый контейнер и запустить пример на выполнение в эмуляторе.

Откройте видеogramму **StudentV** в режиме работы. Для удобства работы с графиками следует изменить настройки длительности отображения. Над областью отображения расположен список. Для открытия списка нажмите на стрелку справа от него (рис.3.23).

Выберите длительность для области отображения 10 мин.

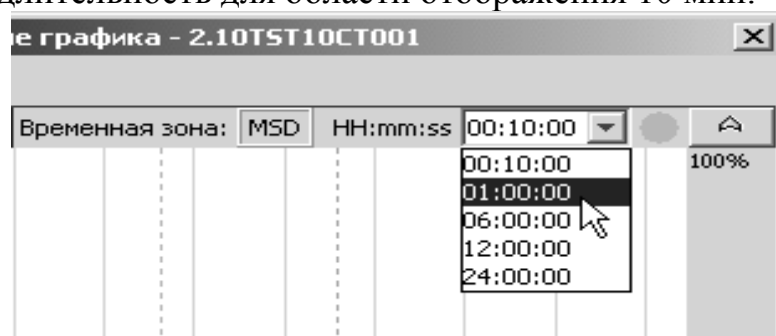


Рис. 3.23. Установка временных параметров графика.

Изменить уставку задатчика на +10. Будет наблюдаться процесс, показанный на рис.20.

Последним этапом работы предлагается поменять настройки регуляторов и добиться лучшего затухания переходного процесса, например как на рис.3.24.

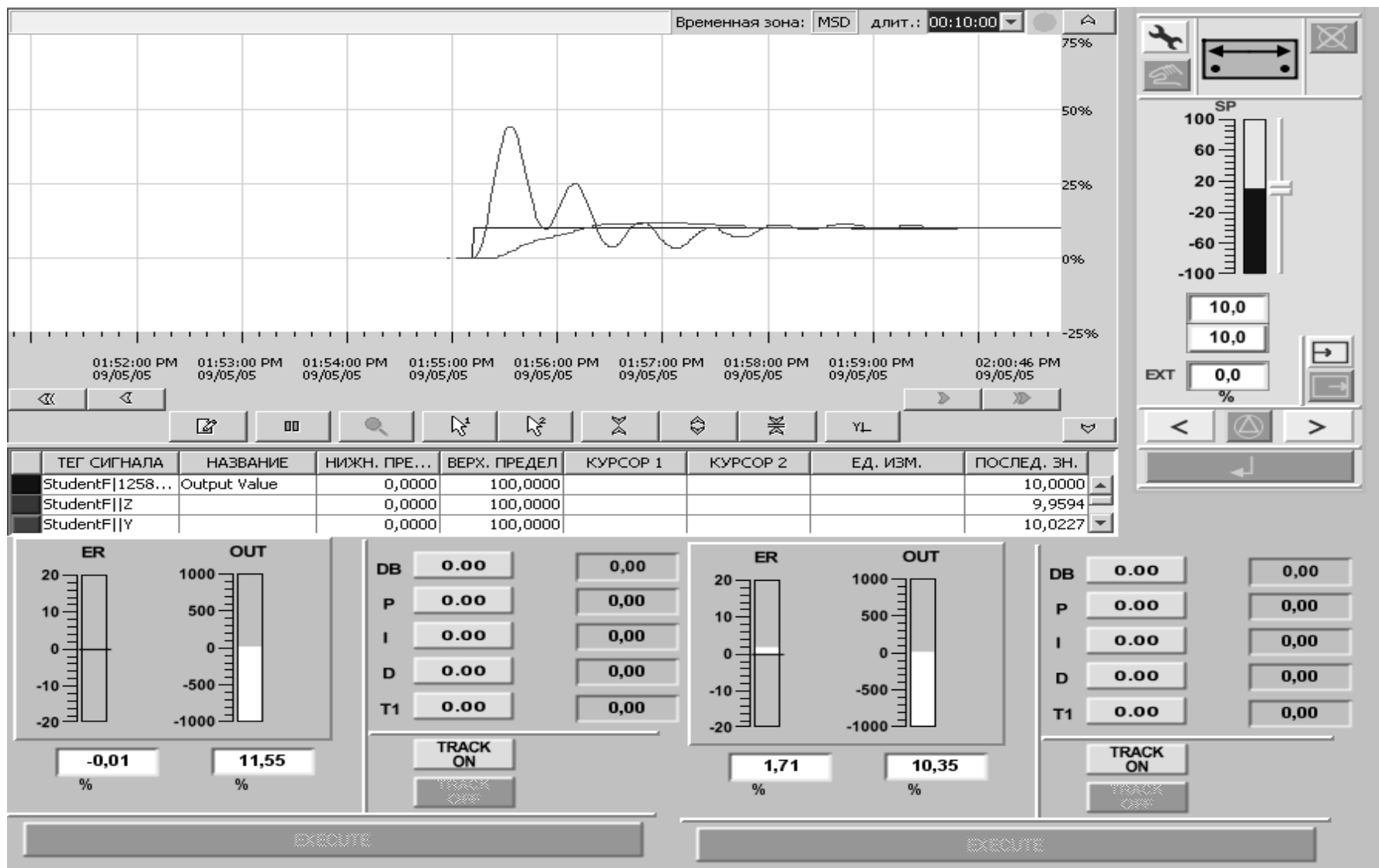


Рис.3.24. Общий вид рабочего экрана видеозаписи и переходные процессы во внешнем и внутренних контурах при возмущении заданием.

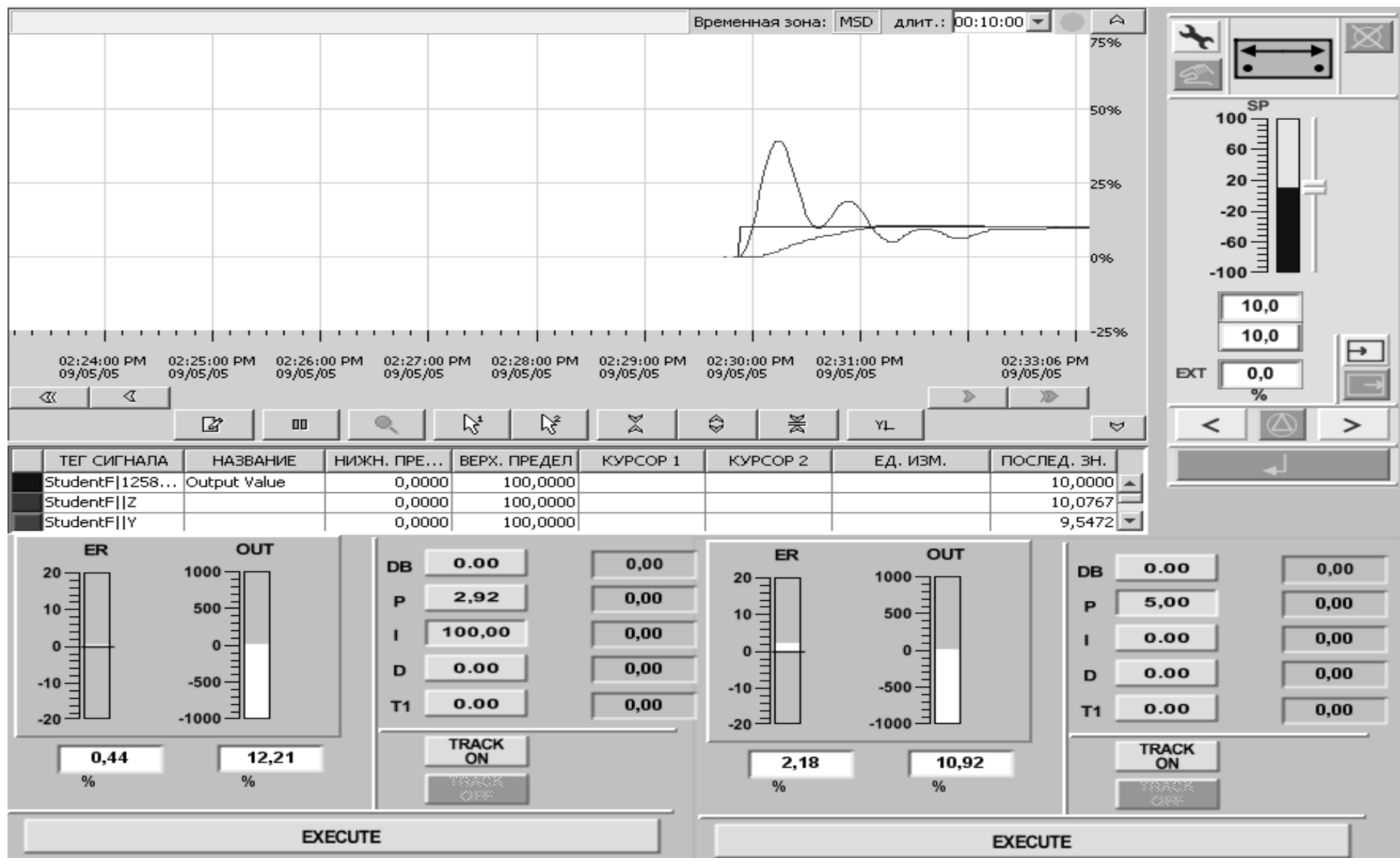


Рис.3.25. Переходные процессы при других настройках регуляторов.

Вопросы к защитам лабораторных работ

1. Объясните основное назначение системы SPPA-T3000.
2. В чем суть технологии “тонких клиентов” в SPPA-T3000?
3. Перечислите основные стили технологического программирования по стандарту МЭК–61131.
4. Какие интерфейсы входят в систему SPPA-T3000.
5. Как открыть вид проекта, что это такое?
6. Как разместить библиотечные алгоритмы на функциональной схеме?
7. Назначение исполняемого контейнера.
8. Как активизировать проект и запустить режим работы?
9. Как деактивизировать проект и выйти из режима работы?
10. Что такое экран ПСО, как его открыть?
11. Что такое узлы проекта и узлы библиотеки?
12. Каковы две основные части проекта?
13. Что такое функциональная схема?
14. Что есть исполняемый контейнер?
15. Как назначить исполняемый контейнер?
16. Что такое панель управления объекта?
17. Как открыть вид диагностики на панели управления?
18. Приведите примеры программных модулей библиотеки SPPA-T3000.
19. Как привязать функциональную схему к видеограмме?
20. Как создать шаблоны для функциональной схемы и видеограммы?
21. Что такое макрос, как его создать?
22. Объясните принцип работы блока MOTOR из библиотеки Open Loop Controller SPPA-T3000.
23. Объясните принцип работы блока ASMON из библиотеки Conditioning and Monitor SPPA-T3000.
24. Как на функциональной схеме связать два блока?
25. Что такое маска параметров для функционального блока?
26. Перечислите статические элементы видеограммы.
27. Перечислите динамические элементы видеограммы.
28. Раскройте смысл термина “тег” в SPPA-T3000.
29. В чем разница между AFC и RTC контейнерами?
30. Какой функциональный блок отвечает за логику задатчика?
31. Какие блоки удобно использовать для построения модели технологического объекта?
32. Виды коннекторов, их различия.

Учебное издание

Зверьков Владимир Петрович

Мезин Сергей Витальевич

Подготовка и реализация проектов АСУТП на базе ПТК

SPPA T-3000 фирмы Сименс

ЛАБОРАТОРНЫЙ ПРАКТИКУМ

по курсу

«Информационное обеспечение систем управления»

для студентов, обучающихся по направлениям 220200 и 140100

Редактор издательства

ЛР № 020528 от 05.06.97

Темплан издания МЭИ

Подписано к печати Печать офсетная

Формат 60×84/16 Физ. печ. л. 8

Тираж 300 экз. Изд. № 118 Заказ

Цена руб.

Издательство МЭИ, 111250, Москва, Красноказарменная ул., д. 14

Отпечатано в типографии ЦНИИ «Электроника»,

117415, Москва, просп. Вернадского, д. 39