

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего
образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по выполнению лабораторных работ

по дисциплине «Интеллектуальные подсистемы САПР»
для студентов направления подготовки

11.03.04 Электроника и нанoeлектроника

Направленность (профиль)

Интеллектуальные программируемые системы и устройства

Ставрополь 2026 г.

ОГЛАВЛЕНИЕ

ЛАБОРАТОРНАЯ РАБОТА №1

"РАЗРАБОТКА ПРОЕКТА АСУ ТП В СРЕДЕ SCADA GENIE 3.0"	6
1 ВВЕДЕНИЕ	6
2 ЦЕЛЬ РАБОТЫ.....	6
3 КРАТКОЕ ОПИСАНИЕ SCADA-системы GENIE 3.0.....	6
3.1 Состав и архитектура системы GENIE	6
3.2 Редактор задач	6
3.3 Редактор форм отображения.....	12
3.4 Редактор отчетов	13
3.5 Редактор сценариев.....	14
4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ	15
5 КОНТРОЛЬНЫЕ ВОПРОСЫ.....	16
6 СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА.....	16

ЛАБОРАТОРНАЯ РАБОТА № 2

"ОСНОВНЫЕ ЭТАПЫ РАЗРАБОТКИ ПРОЕКТОВ АСУ ТП В СРЕДЕ SCADA-СИСТЕМЫ TRACE MODE 5.0"	17
1 ВВЕДЕНИЕ	17
2 ЦЕЛЬ РАБОТЫ.....	17
3 КРАТКОЕ ОПИСАНИЕ SCADA-системы TRACE MODE 5.0.....	17
3.1 Структура системы	17
3.2 Основы проектирования базы каналов	22
3.2.1 Основные понятия редактора базы каналов	22
3.2.2 Автопостроение баз каналов	26
3.2.3 Канал TRACE MODE	28
3.3 Основы разработка графического интерфейса	31
3.3.1 Основные понятия редактора представления данных	31
3.3.2 Графические элементы рисования.....	32
3.3.3 Графические элементы форм отображения	34
4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ	36
5 КОНТРОЛЬНЫЕ ВОПРОСЫ.....	38
6 СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА.....	38

ЛАБОРАТОРНАЯ РАБОТА № 3

"РАЗРАБОТКА АРМ-ОПЕРАТОРА ДЛЯ СУПЕРВИЗОРНОГО УПРАВЛЕНИЯ МИКРОПРОЦЕССОРНЫМ КОНТРОЛЛЕРОМ РЕМИКОНТ Р-130"	39
1 ВВЕДЕНИЕ	39
2 ЦЕЛЬ РАБОТЫ.....	39
3 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ	40
3.1 Значения канала	40
3.2 Классификация каналов	41
3.3 Обработка данных в канале	42
3.4 Классификация атрибутов канала	44

3.5 Первичная и выходная обработка	50
3.5.1 Масштабирование.....	52
3.5.2 Дрейф нуля	53
3.5.3 Подавление пиков.....	53
3.5.4 Апертура	54
3.5.5 Экспоненциальное сглаживание	54
3.5.6 Контроль шкалы.....	55
3.5.7 Анализ на недопустимость сочетания	56
3.5.8 Инверсия	56
3.5.9 Предустановка.....	56
4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ.....	57
4.1 Настройки SCADA-системы TRACE MODE.....	57
4.1.1 Настройка в редакторе базы каналов	57
4.1.2 Создание и настройка каналов	58
4.1.3 Настройки в редакторе представления данных	63
4.2 Настройка Ремиконт Р-130	67
4.3 Настройка шлюза	70
4.4 Обобщающие действия по настройке системы	70
4.5 Последовательность выполнения дополнительного задания.....	70
5 КОНТРОЛЬНЫЕ ВОПРОСЫ.....	71
6 СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА.....	71

ЛАБОРАТОРНАЯ РАБОТА № 4

" ПРИМЕНЕНИЕ СТАНДАРТНЫХ ЯЗЫКОВ

ПРОГРАММИРОВАНИЯ В SCADA-СИСТЕМЕ TRACE MODE

ДЛЯ МОДЕЛИРОВАНИЯ РАЗОМКНУТЫХ И ЗАМКНУТЫХ

АВТОМАТИЧЕСКИХ СИСТЕМ"

1 ВВЕДЕНИЕ	72
2 ЦЕЛЬ РАБОТЫ.....	72
3 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ	72
3.1 Языки программирования алгоритмов управления	72
3.1.1 Язык ТехноFBD.....	73
3.1.2 Вызов FBD-программ	79
3.1.3 Язык ТехноLD	80
3.1.4 Язык ТехноIL	82
3.2 Несколько примеров создания моделей объекта.....	83
3.2.1 Создание статической модели объекта	83
3.2.2 Создание динамической модели объекта.....	87
4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ.....	89
5 КОНТРОЛЬНЫЕ ВОПРОСЫ.....	92
6 СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА.....	92

ЛАБОРАТОРНАЯ РАБОТА № 5

" РАЗРАБОТКА РАСПРЕДЕЛЁННОЙ АСУ ТП ДЛЯ УПРАВЛЕНИЯ

СЛОЖНЫМ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ "

1 ВВЕДЕНИЕ	93
------------------	----

2 ЦЕЛЬ РАБОТЫ.....	93
3 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ	93
3.1 Идеология построения распределенных АСУ ТП в Трейс Моуд	94
3.2 Сетевой обмен данными между узлами проекта	96
3.2.1 Файловый обмен.....	96
3.2.2 Обмен «точка-точка»	97
3.2.3 Обмен «один ко многим»	98
3.2.4 Создание и настройка связи с каналами ОРС-сервера.....	102
4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ	105
5 КОНТРОЛЬНЫЕ ВОПРОСЫ.....	106
6 СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА.....	106
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	107

Предисловие

В настоящее время для предприятий, независимо от формы собственности, стала актуальной задача повышения эффективности производства и качества выпускаемой продукции, а также обеспечения нового качества управляемости за счет создания единого информационного пространства предприятия. Поэтому современная АСУ ТП обязательно должна предусматривать связь с корпоративными системами управления предприятием (АСУП), которые в современной терминологии на уровне предприятия называются MRP системами (Manufacturing Resource Planning) – планирование ресурсов производства, а на уровне производства называются MES системами (Manufacturing Execution System) - управление производством.

Основой программного обеспечения (ПО) верхнего уровня АСУ ТП составляет SCADA (Supervisory Control And Data Acquisition) программа, реализующая все основные функции визуализации измеряемой и контролируемой информации, передачи данных и команд системе контроля и управления.

Современный рынок предлагает большое количество различных SCADA систем, сильно отличающихся друг от друга. Но, не смотря на все их отличия, они имеют практически одинаковый алгоритм создания проекта для управления технологическим процессом.

В любой из SCADA систем присутствует компонент, при помощи которого осуществляется визуализация технологического процесса, то есть создание мнемосхем технологических объектов и настройка параметров их свойств. После того как мнемосхема создана, осуществляется настройка параметров динамических характеристик мнемосхем. Для формирования динамических характеристик основой любой SCADA системы является база данных реального времени. В данной базе постоянно записываются мгновенные значения, полученные от датчиков и от оператора, а при необходимости для каких либо параметров могут создаваться архивы.

Таким образом, помимо решения задач диспетчеризации, система SCADA является информационной основой для корпоративных систем управления предприятием.

Среди всего спектра SCADA систем в данном практикуме для студентов предлагается освоить две. Это самая простая система, представленная на Российском рынке система Тайваньского производства Genie 3.0 и вполне масштабная система Российского производства TRACE MODE 5.0.

ЛАБОРАТОРНАЯ РАБОТА №1

" РАЗРАБОТКА ПРОЕКТА АСУ ТП В СРЕДЕ SCADA GENIE 3.0"

1. ВВЕДЕНИЕ

Пакет GENIE, разработанный фирмой Advantech, является инструментальным средством для создания программного обеспечения сбора данных и оперативного управления (SCADA-система Supervisor Control And Data Acquisition System), исполняющегося в среде MS Windows 3.x, Windows 95 и Windows 98. GENIE текущей версии может применяться в проектах АСУ ТП масштаба технологического участка или цеха, с начальным или средним уровнем сложности.

2. ЦЕЛЬ РАБОТЫ

В результате выполнения лабораторной работы студент должен ознакомиться со структурой и возможностями SCADA-системы GENIE 3.0 и разработать систему управления процессом согласно варианту.

3. КРАТКОЕ ОПИСАНИЕ SCADA-системы GENIE 3.0

3.1. Состав и архитектура системы GENIE

Пакет GENIE состоит из двух основных программных модулей: Построитель стратегий GENIE (GENIE.EXE) и Исполнительная среда GENIE (GWRUN.EXE). Построитель стратегий используется для проектирования и тестирования проектов, называемых стратегиями, а Исполнительная среда — только для исполнения стратегий. Кроме того, в составе пакета имеется программа установки и настройки устройств ввода/вывода, а также набор динамически компокуемых библиотек (DLL), выполняющих различные функции в процессе разработки и выполнения программного обеспечения SCADA.

GENIE имеет модульно-ориентированную, открытую интегрированную архитектуру показанную на рисунке 1. Открытость архитектуры позволяет легко реализовывать взаимодействие GENIE с другими приложениями для совместного доступа к данным во время исполнения стратегий. Результат разработки приложения в GENIE сохраняется в файле стратегии. Файл стратегии (с расширением .GNI) представляет собой двоичный файл, содержащий всю информацию последнего сеанса редактирования.

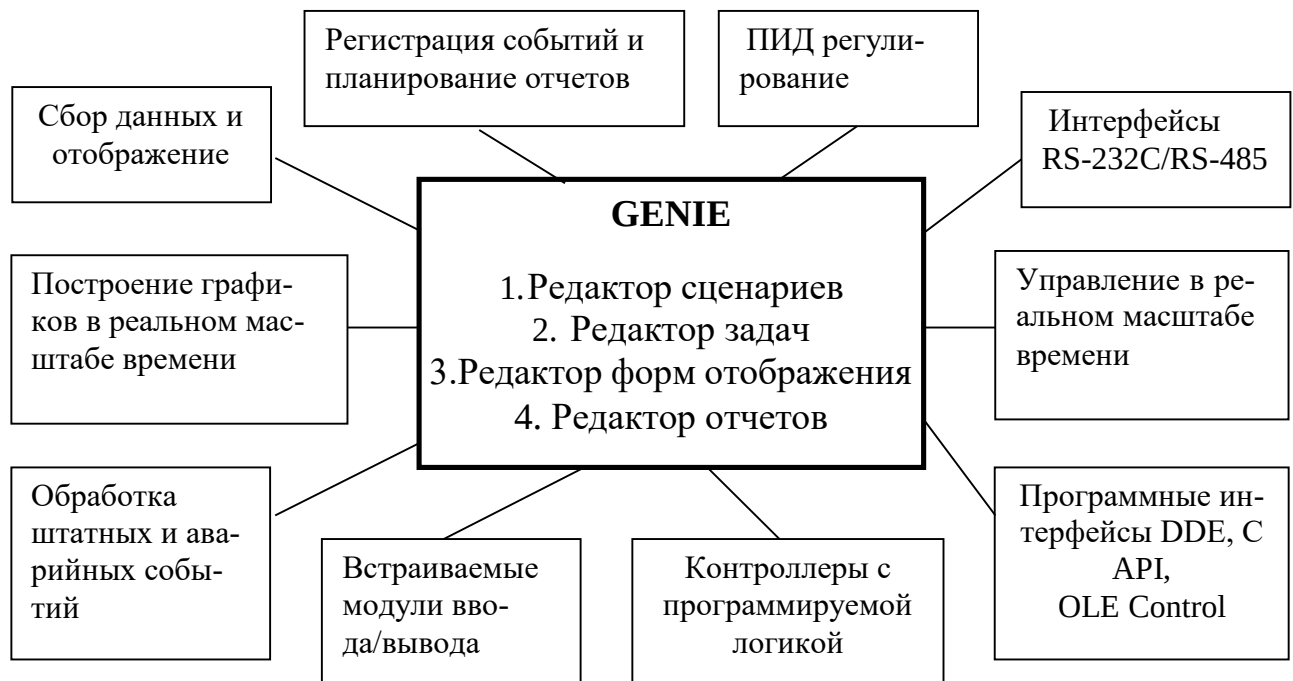


Рисунок 1 - Архитектура GENIE 3.0.

Стратегия - это совокупность одной или нескольких задач вместе с одной или большим количеством экранных форм, а также одним основным сценарием. Задача, экранная форма и основной сценарий являются тремя основными элементами, используемыми при проектировании стратегий. Простейшая стратегия имеет одну задачу с одной экранной формой и не имеет основного сценария.

Задача представляет собой набор функциональных блоков, отображаемых в окне задачи в виде пиктограмм. Экранная форма представляет собой набор элементов отображения и элементов управления. Пиктограммы функциональных блоков и элементы отображения/управления являются стандартными «кирпичиками» для построения стратегии. Они очень похожи по внешнему виду, за исключением того, что элементы отображения служат для реализации графического интерфейса пользователя, в то время как пиктограммы блоков скрыты во время исполнения стратегии.

Соединения между функциональными блоками в процессе разработки стратегии могут устанавливаться посредством видимых и невидимых связей. Соединения между пиктограммами блоков являются видимыми в окне *Редактора задач*. Такое соединение называется проводником, поскольку по своему назначению оно аналогично проводам в электрических схемах. Соединения между пиктограммами блока и элементами отображения невидимы. Поэтому они называются связями.

Каждая задача и экранная форма имеют собственные параметры. Впервые созданная новая задача или новая экранная форма имеют заданные по умолчанию параметры, которые могут быть изменены пользователем в соответствии с требованиями алгоритма. Задача имеет такие параметры, как период сканирования (интервал времени между текущим и следующим вызовом задачи для ис-

полнения), эффективный интервал исполнения (абсолютное время выполнения задачи), метод запуска и метод завершения. Минимальный период сканирования задачи, а значит, и связанных с ней параметров составляет 55мс. Экранная форма имеет такие параметры, как имя, стиль отображения, состояние и др.

В рамках стратегии может использоваться одновременно практически любое количество функциональных блоков и устройств ввода-вывода, которое ограничивается только быстродействием и размером памяти компьютера. Однако рекомендуется, чтобы количество функциональных блоков не превышало 500.

Построитель стратегий GENIE предоставляет в распоряжение пользователя четыре различных редактора:

- редактор задач,
- редактор форм отображения,
- редактор отчетов,
- редактор сценариев.

Указанные редакторы используются для создания, отладки и модификации задач, экранных форм, отчетов и специальных процедур соответственно. Поскольку в рамках одной стратегии может присутствовать несколько задач, то имеется возможность создания множества окон в *Редакторе задач*. По той же самой причине возможно создавать множество окон в *Редакторе форм отображения* для редактирования множества экранных форм. Но поскольку для любой системы может быть только один основной сценарий, то в Редакторе сценариев может быть открыто только одно окно.

3.2. Редактор задач

Редактор задач пакета GENIE (рисунок 2) использует информационно-поточную модель программирования, которая значительно удобнее для восприятия и алгоритмической интерпретации, чем традиционная линейная архитектура текстовых языков программирования. При разработке приложения сбора данных и управления пользователем создается блок-схема стратегии без обращения особого внимания различным логическим и синтаксическим соглашениям, принятым в стандартных языках программирования. Просто выберите объекты (пиктограммы функциональных блоков) из набора инструментов *Редактора задач* и соедините их проводниками для передачи данных от одного блока к другому.

Задачи, образующие стратегию GENIE и вызываемые (сканируемые) в процессе ее исполнения, разрабатываются при помощи *Редактора задач*. Для облегчения процесса разработки алгоритмов сбора данных и управления в *Редакторе задач* имеется набор инструментов, который содержит базовые функциональные блоки. Каждый функциональный блок предназначен для выполнения соответствующей встроенной функции обработки данных, поступающих от аппаратуры или вводимых пользователем.

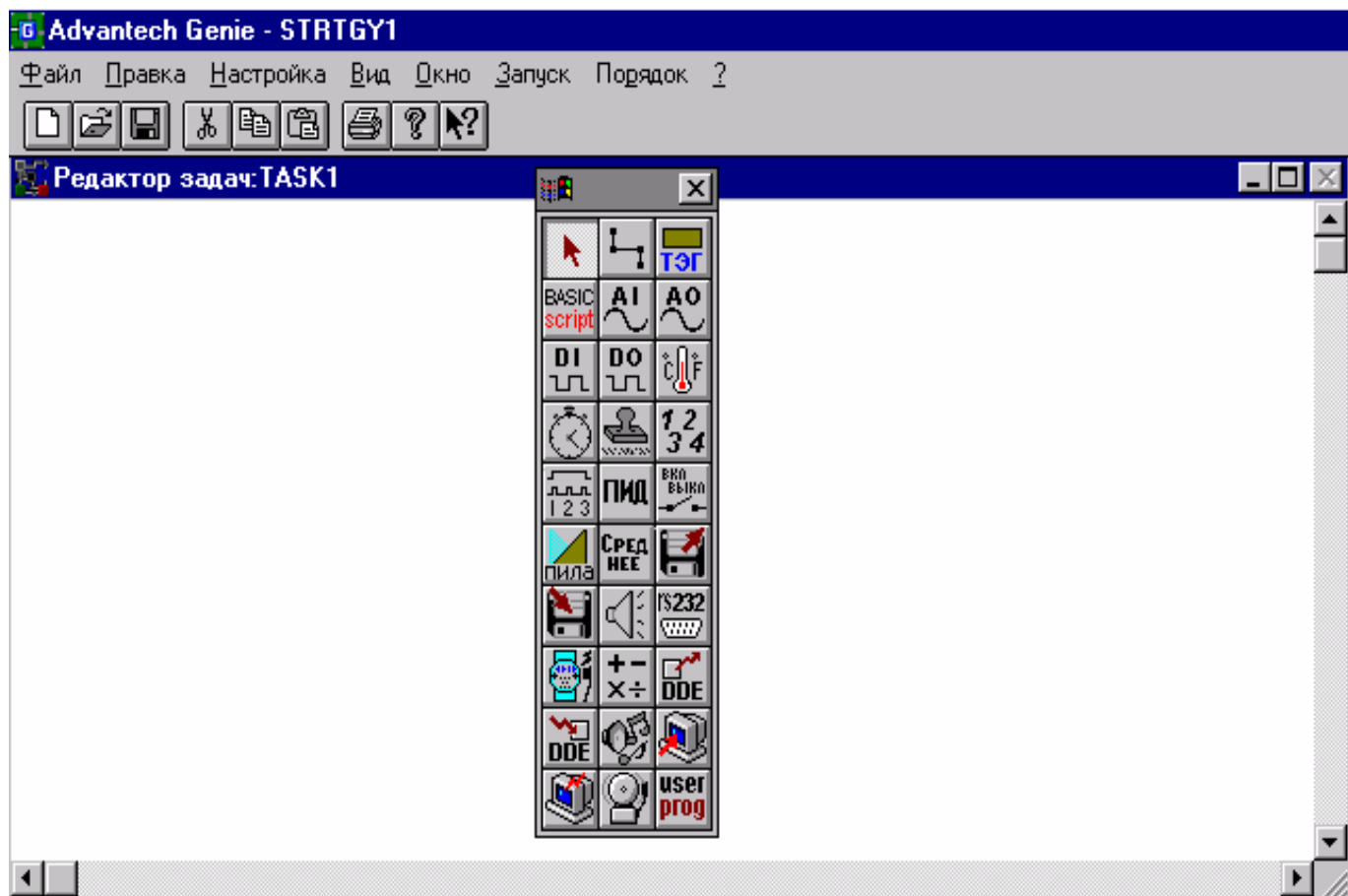


Рисунок 2 - Панель Редактора задач.

Ряд блоков позволяет организовывать взаимодействие непосредственно с низкоуровневыми драйверами аппаратуры.

Меню Редактора задач содержит набор команд, предназначенных для выполнения файловых операций, редактирования стратегии, настройки устройств ввода-вывода, запуска стратегии на исполнение, просмотра стратегии и манипуляций с окнами. Кроме того, меню содержит команду вызова справочной системы:

- *Файл* - содержит команды, позволяющие открывать, сохранять, закрывать, печатать и создавать файлы стратегии GENIE.
- *Правка* - содержит команды, позволяющие выполнять операции редактирования функциональных блоков стратегии GENIE. Перед выполнением любой команды меню Правка, функциональные блоки, подлежащие редактированию, должны быть выделены.
- *Настройка* - содержит команды, позволяющие выполнять установку, настройку или удаление устройств ввода-вывода, настройку параметров задач, входящих в стратегию, а также настройку параметров исполнительской среды GENIE.

- *Вид* - содержит команды, которые позволяют масштабировать размер пиктограмм функциональных блоков задачи, окно которой активно, таким образом, что все они помещаются в видимой области окна, скрывать и отображать строку состояния, панель инструментов и набор инструментов окна Построителя стратегии, а также позиционные обозначения функциональных блоков и порядок их исполнения.
- *Окно* - содержит команды, которые позволяют упорядочить взаимное расположение окон задач и экранных форм GENIE. В GENIE поддерживаются стандартные операции над окнами (Каскад, Расположить горизонтально без взаимного перекрытия, Упорядочить значки). Кроме того, с помощью данного меню имеется возможность переключения между различными окнами стратегии.
- *Запуск* - содержит команды, позволяющие запускать на исполнение текущую загруженную стратегию, останавливать ее исполнение и блокировать органы управления Windows на время исполнения стратегии без выхода из системы разработки стратегий (Построителя стратегии). В отличие от предыдущих версий GENIE, в которых при выполнении команд данного меню происходил запуск исполнительной среды и переход в ее окно, в GENIE версии 3.0 система исполнения и система разработки объединены в рамках одной программной оболочки, что обеспечивает возможность отладки и тестирования стратегии в процессе ее разработки.
- *Порядок* - содержит команды, позволяющие установить и изменить порядок исполнения функциональных блоков в задаче, окно которой активно. Имеется возможность изменения порядка исполнения нескольких блоков либо полного переопределения порядка исполнения всех блоков. Когда отмечена строка Изменить для всех блоков меню Порядок, однократный щелчок левой клавишей мыши на каждом функциональном блоке задачи переопределяет порядок их исполнения в соответствии с последовательностью, в которой были произведены щелчки левой клавишей мыши. Порядок исполнения отражается последовательностью чисел от 1 до N, которые отображаются на пиктограмме функционального блока при выборе строки Порядок исполнения меню Вид. Если требуется поменять местами порядок исполнения двух блоков, следует отметить строку Изменить для двух блоков меню Порядок и произвести однократный щелчок левой клавишей мыши на каждом из них.

Набор инструментов *Редактора задач* (рисунок 3) помимо пиктограмм функциональных блоков, с помощью которых строится алгоритм создаваемой стратегии, содержит инструменты для редактирования стратегии и установления связей между функциональными блоками.

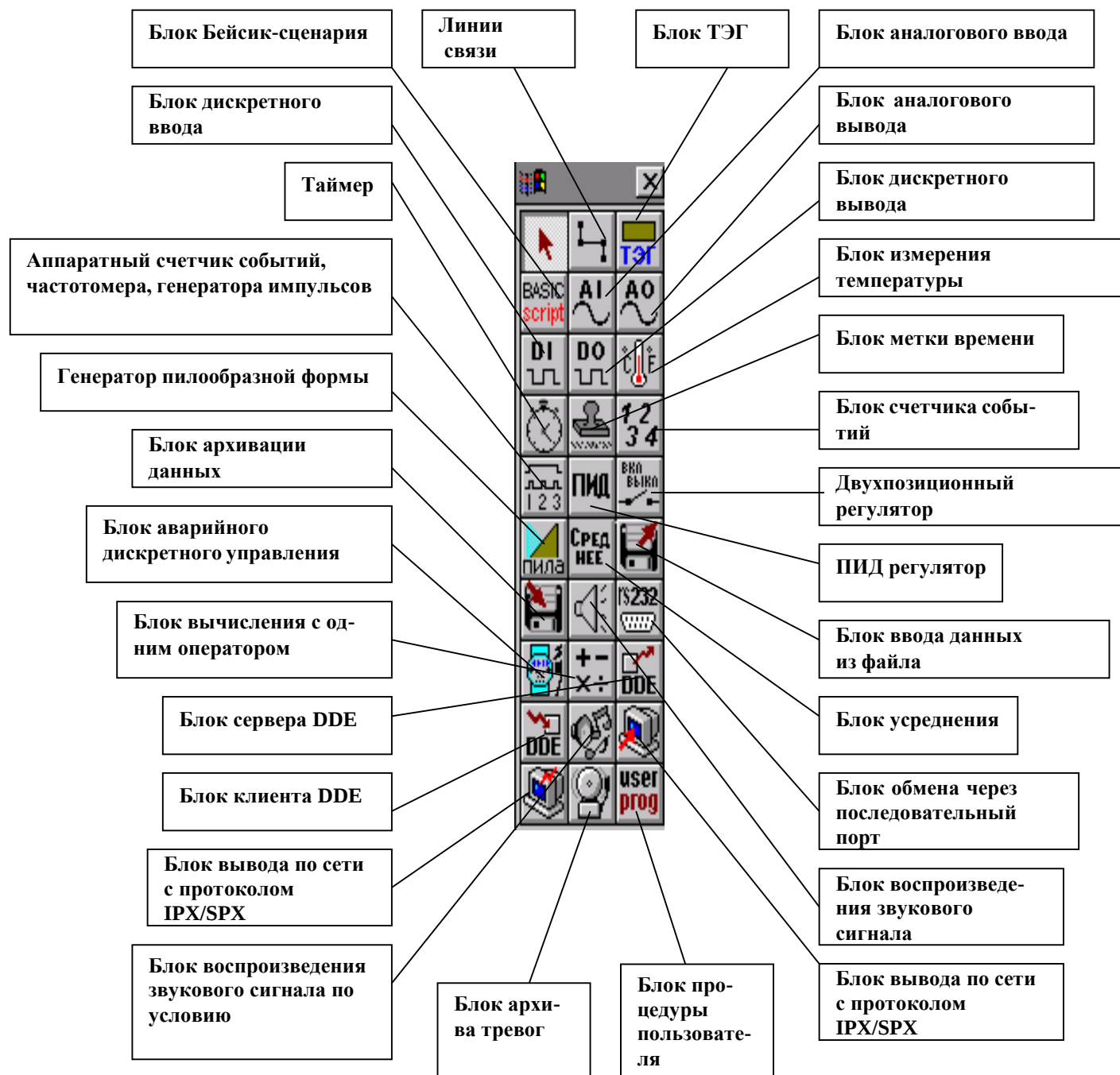


Рисунок 3 - Набор инструментов *Редактора задач*.

Более подробно назначение пиктограмм панели инструментов можно рассмотреть при вызове справки в программе GENIE.

3.3. Редактор форм отображения

Редактор форм отображения GENIE предназначен для создания графических мнемосхем автоматизированных рабочих мест оператора, обеспечивающих представление информации о контролируемом технологическом процессе в удобной для восприятия форме. Кроме того, он обеспечивает возможность использования растровых изображений, создаваемых пользователем, в качестве фоновых рисунков экранных форм. Экранные формы позволяют оператору следить за ходом контролируемого процесса и осуществлять управление его параметрами в процессе исполнения стратегии, созданной в *Редакторе задач*.

Назначение **меню** Редактора форм отображения аналогично меню Редактора задач.

Набор инструментов Редактора форм отображения (рисунок 4) позволяет разрабатывать мнемосхемы технологического процесса для рабочего места оператора.

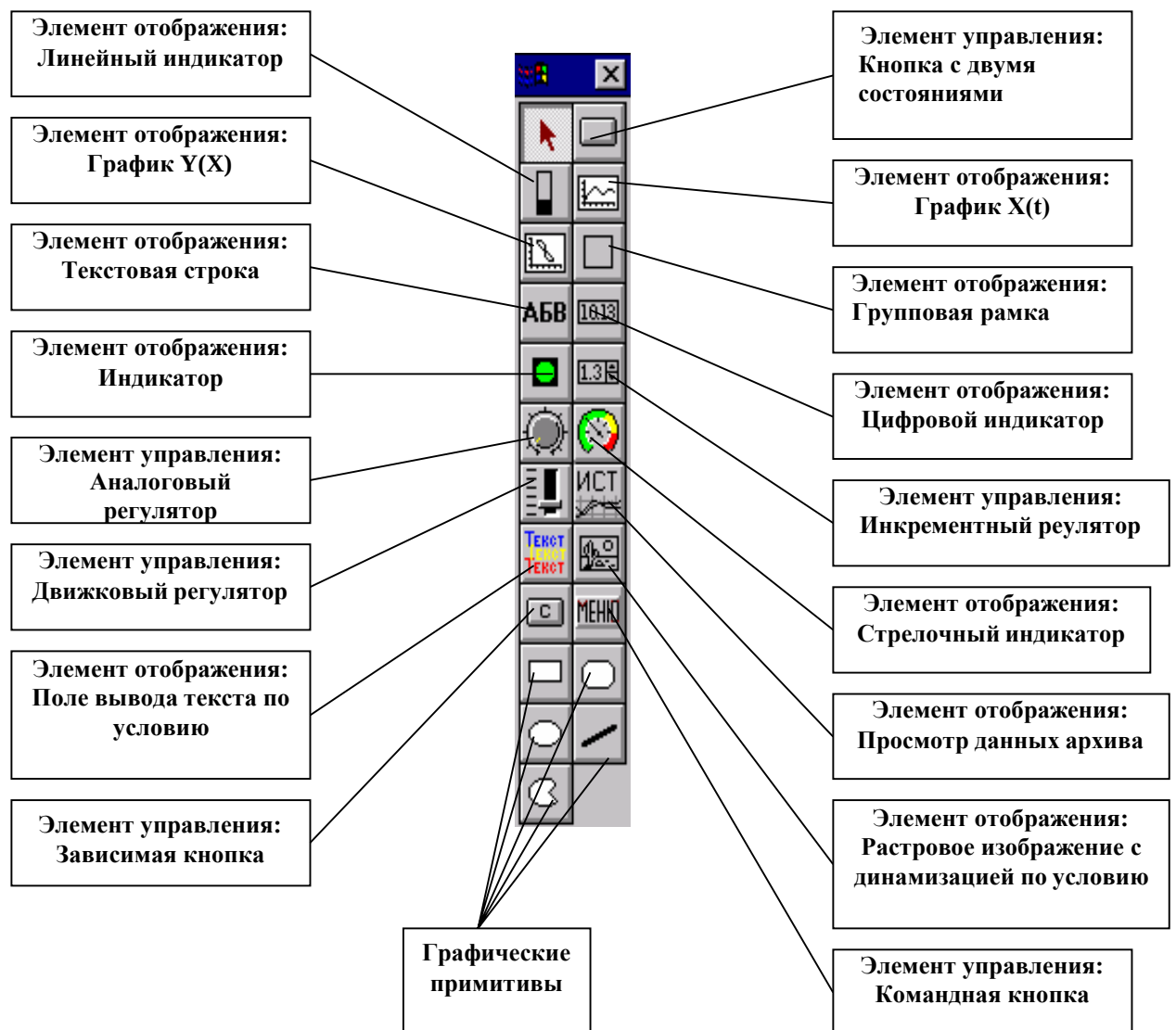


Рисунок 4 - Набор инструментов Редактора форм отображения.

3.4. Редактор отчетов

Редактор отчетов содержит графическую оболочку, в которой пользователь определяет вид и содержимое отчетов. В процессе исполнения стратегии подсистема генерации отчетов выполняет сбор данных, связанных с заданными на этапе разработки стратегии идентификаторами каналов функциональных блоков (тэгами), за predetermined интервалы времени и автоматический вывод отчетов на печать в заданные моменты времени. Средства взаимодействия с пользователем, входящие в *Редактор отчетов*, обеспечивают возможность печати отчетов по команде пользователя (вручную).

Редактор отчетов состоит из четырех основных компонентов:

1) Подсистема настройки параметров и формата отчета

Данная подсистема содержит диалоговые панели, которые позволяют пользователю настроить формат и время печати отчета. Записи о содержимом отчета помещаются в таблицу. Каждый столбец таблицы редактируется пользователем и содержит текст или специальные ключевые слова для каждого идентификатора канала, информация от которого подлежит включению в отчет. Информация о каждом используемом формате отчета заносится в специальный файл формата и извлекается во время генерации отчета.

2) Подсистема сбора данных

Подсистема сбора данных активизируется встроенным таймером *Редактора отчетов*, который настроен на 10-секундные интервалы. Подсистема сбора данных заносит информацию с заданных каналов в файл суточной базы данных в заданные пользователем моменты времени. Подсистема сбора данных *Редактора отчетов* разработана исключительно для генерации отчетов, тогда как высокоскоростное архивирование информации должно реализовываться с помощью других средств пакета GENIE (например, с помощью элемента отображения Просмотр данных архива или функционального блока архивации данных).

3) Планировщик отчетов

Производит фиксацию моментов времени, в которые должна быть выполнена печать отчетов, в течение суток. Планировщик отчетов активизирует подсистему генерации отчетов в заданные пользователем моменты времени для печати требуемого отчета. Кроме того, планировщик отчетов информирует пользователя о статусе печати отчета.

4) Подсистема генерации отчетов

Данная подсистема извлекает информацию из файла формата и файла суточной базы данных, после чего выводит требуемый отчет на печатающее устройство. В текущей версии пакета поддерживается только табличная форма отчетов. Отчеты в графической форме со встроенными суточными графиками будут реализованы в следующих версиях GENIE.

Подсистема сбора данных и планировщик отчетов скрыты от пользователя и активизируются только в процессе исполнения стратегии. Пользователь имеет возможность настройки параметров и формата каждого отчета на этапе разработки стратегии путем использования соответствующей подсистемы *Редактора отчетов*.

3.5. Редактор сценариев

Редактор сценариев предназначен для управления задачами, вычислениями и анализом данных в процессе исполнения стратегии.

Данное мощное средство создания сценарных процедур, совместимое с Microsoft Visual Basic, лицензировано у компании Summit Software Inc. Бейсик-сценарий является наиболее важным компонентом пакета GENIE, обеспечивающим возможность разработки специализированных фрагментов стратегии на языке программирования высокого уровня. Поскольку Visual Basic является одним из самых популярных и простых для изучения языков программирования, его наличие в составе пакета позволяет значительно упростить и повысить эффективность программирования в рамках GENIE.

Ядро подсистемы программирования и исполнения сценарных процедур пакета GENIE представляет собой набор библиотек динамической компоновки, с помощью которых выполняется предварительная компиляция сценарных процедур на этапе разработки и их исполнение в процессе выполнения всей стратегии. Бейсик-сценарий обеспечивает возможность не только управления задачами, входящими в стратегию, но и взаимодействия с DOS, Windows и другими приложениями посредством механизмов DDE, OLE, ODBC (SQL) и прямых вызовов функций интерфейса прикладного программирования Windows API. При использовании в Бейсик - сценарии базовых функций имеется возможность компиляции и исполнения программ, написанных на Visual Basic без каких-либо изменений. Номера ошибок и сообщения об ошибках в Бейсик - сценарии также совместимы с имеющимися в Visual Basic. В состав *Редактора сценариев* входит редактор диалоговых панелей, который позволяет разрабатывать собственные средства взаимодействия с оператором.

Наличие данного средства создания и исполнения сценарных процедур делает пакет GENIE одним из наиболее удобных и современных инструментов для разработки программного обеспечения верхнего уровня систем сбора данных и оперативного диспетчерского управления (SCADA).

Редактор сценариев представляет собой редактор текста с рядом удобных возможностей для редактирования кода сценария. Исходный текст сценария компилируется в псевдокод сразу после редактирования, так что нет необходимости компилировать его вновь при запуске программы. Редактирование основного сценария и сценариев, входящих в задачи стратегии, выполняется при помощи одного и того же редактора. Основной сценарий (Main Script), будучи включенным в стратегии осуществляет управление исполнением всей стратегии один раз в течении одного сеанса исполнительной среды GENIE. Основной сценарий может быть применен для выполнения таких операций, как запуск и остановка задач и т. п. Каждая сканируемая задача имеет предварительный сценарий (Pre-Task Script) и пост-сценарий (Post-Task Script). Указанные два типа сценариев при определенных условиях используются для инициализации или сброса значений, связанных с объектами, которые входят в стратегию. Основной сценарий исполняется один раз, тогда как предварительный и пост-сценарий задачи исполняются при каждом вызове задачи.

4. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Лабораторная работа выполняется на ПК "IBM PC" в следующей последовательности:

- 1) Включить ПК и двойным щелчком левой кнопки "мыши" запустить программу "Построитель стратегий GENIE" 
- 2) С помощью методического пособия ознакомиться со структурой основных редакторов системы GENIE. В отчёте отметить название и цель работы.
- 3) Получить у преподавателя один из 6-ти вариантов задания.
- 4) Изучить по рисунку принцип работы технологического участка производства.
- 5) Последовательно выполняя все действия выданного варианта создать проект, показанный на рисунке вначале задания.
- 6) После завершения программирования задачи в редакторе задач, обозначьте технологические параметры, отображаемые на экранной форме и запишите функциональные зависимости, связывающие их.
- 7) Покажите выполненную задачу и оформленный отчёт преподавателю и получите задание №2.
- 8) Обозначьте новые технологические параметры, требуемые для решения новой задачи, и
- 8) Используя рекомендации методического пособия, самостоятельно выполните задание №2.
- 9) Запишите новые функциональные зависимости и новую блок-схему программы, составленной в редакторе задач.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каково назначение SCADA-систем, и SCADA-системы Genie 3.0 в частности?
2. Опишите логическую архитектуру SCADA-системы Genie 3.0.
3. Назначение и структура редактора задач.
4. Назначение и структура редактора форм отображения.
5. Назначение и структура редактора отчётов.
6. Назначение и структура редактора сценариев.
7. Что такое Тег, и для чего он может использоваться?
8. Что такое Тренд?
9. Что вы понимаете под открытостью и интегрированностью архитектуры SCADA-системы GENIE 3.0?
10. Что такое стратегия?
11. Что такое задача, и из каких элементов она состоит?
12. Каково назначение и структура экранной формы?
13. Какова особенность информационно-поточной модели программирования?

6. СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА

Отчет оформляется в ученической или в общей тетради в соответствии с общими требованиями к оформлению текстовых документов. Отчет должен содержать:

- 1) наименование и цель работы;
- 2) номер варианта и название технологического процесса;
- 3) схема и краткое описание технологического процесса;
- 4) блок-схему программы, реализованной в редакторе задач для решаемого варианта;
- 5) обозначение технологических параметров, показанных на экранной форме и функциональные зависимости, связывающие их.
- 6) Задание №2, выданное преподавателем;
- 7) обозначение дополнительных технологических параметров, используемых в программе;
- 8) изменённые функциональные зависимости, с добавлением новых технологических параметров;
- 9) новую блок-схему программы, реализованной в редакторе задач.

ЛАБОРАТОРНАЯ РАБОТА № 2

"ОСНОВНЫЕ ЭТАПЫ РАЗРАБОТКИ ПРОЕКТОВ АСУ ТП В СРЕДЕ SCADA-СИСТЕМЫ TRACE MODE 5.0"

1. ВВЕДЕНИЕ

Одной из распространенных отечественных систем является SCADA – система TRACE MODE 5.0. Она широко используется студентами специальности «Автоматизация технологических процессов и производств» в курсовом и дипломном проектировании. TRACE MODE текущей версии может применяться в проектах АСУ ТП масштаба от технологического участка или аппарата, до целого производства со сложной, распределённой структурой АСУ ТП.

2. ЦЕЛЬ РАБОТЫ

В результате выполнения лабораторной работы студент должен:

- ознакомиться со структурой и возможностями SCADA-системы TRACE MODE 5.0.
- определиться с основными понятиями SCADA-системы TRACE MODE 5.0.
- познакомиться с основными этапами создания проекта.

3. КРАТКОЕ ОПИСАНИЕ SCADA-системы TRACE MODE 5.0

3.1. Структура системы

TRACE MODE (русское название – ТРЕЙС МОУД) – это программный комплекс, предназначенный для разработки, настройки и запуска в реальном времени систем управления технологическими процессами. Она включает в себя *инструментальную систему* разработки АСУ и *исполнительные (run-time) модули*. При помощи инструментальной системы осуществляется разработка всех проектов, создаваемых в ТРЕЙС МОУД. Исполнительные модули служат для запуска в реальном времени проектов, разработанных в инструментальной системе ТРЕЙС МОУД.

Инструментальная система ТРЕЙС МОУД состоит из двух систем: *Редактора базы каналов* и *Редактора представления данных*. В этих редакторах осуществляется разработка математической основы АСУ, графических экранных фрагментов для визуализации состояния технологического процесса и управления им.

В Редакторе базы каналов ТРЕЙС МОУД создается математическая основа системы управления: описываются конфигурации всех рабочих станций, контроллеров и УСО, используемых в системе управления, настраиваются информационные потоки между ними.



**Редактор
базы каналов**

Вход в Редактор базы каналов осуществляется либо двойным нажатием левой кнопки (ЛК) мыши на соответствующем ярлыке в папке ТРЕЙС МОУД 5.0, либо запуском исполнительного модуля chb.exe. Вид окна редактора базы каналов показан на рисунке 5.

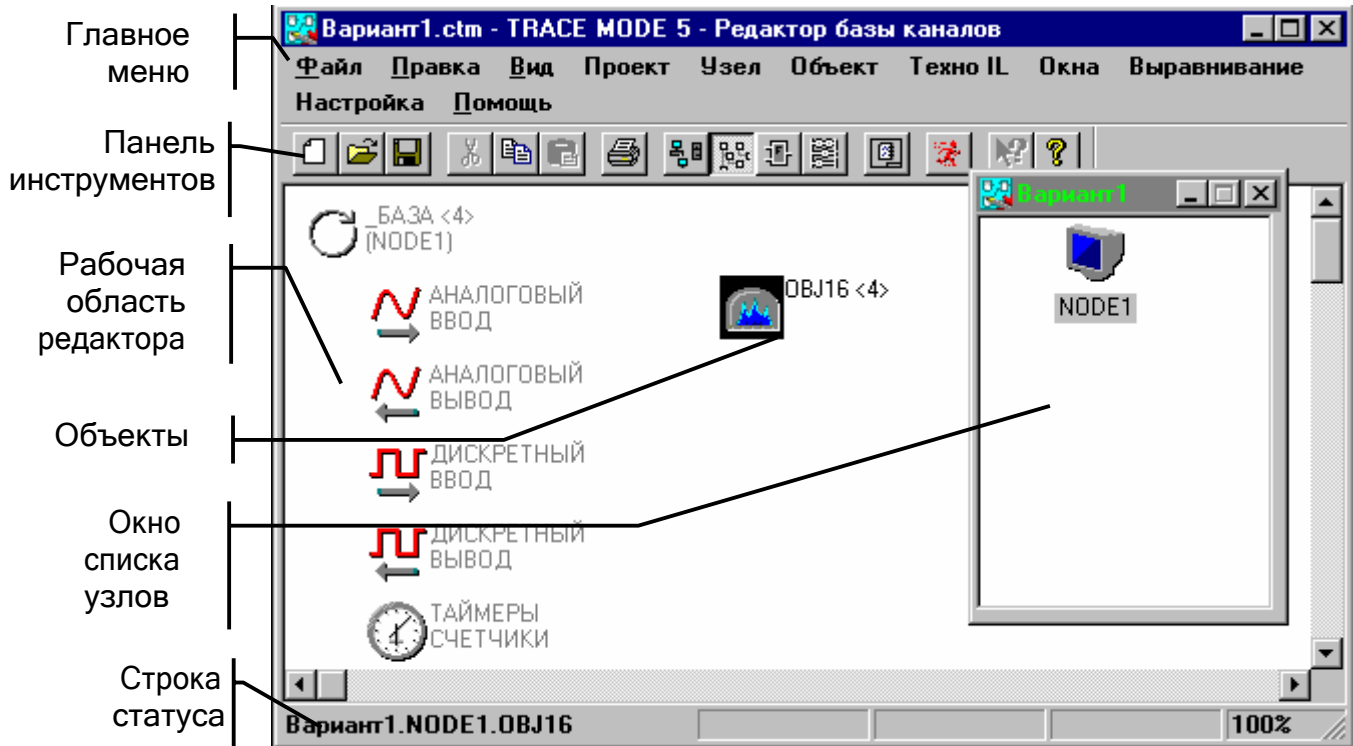


Рисунок 5 - Вид окна редактора базы каналов

В этом редакторе описываются входные и выходные сигналы и их связь с устройствами сбора данных и управления. В этом редакторе задаются периоды опроса или формирования сигналов, настраиваются законы первичной обработки и управления, технологические границы, структура математической обработки данных. Здесь устанавливается, какие данные и при каких условиях сохранять в различных архивах ТРЕЙС МОУД и настраивается сетевой обмен.

Кроме того, здесь же описываются задачи управления архивами, документированием, коррекции временных характеристик системы управления (периоды опроса параметров, время цикла системы и пр.), а также решаются некоторые другие задачи.

Результатом работы в этом редакторе является математическая и информационная структуры проекта АСУТП. Эти структуры включают в себя набор баз каналов и файлов конфигурации для всех контроллеров и операторских станций (узлов) проекта, а также файл конфигурации всего проекта.

Файл конфигурации проекта имеет расширение .cmt и сохраняется в ра-

бочей директории системы разработки. Для хранения всех остальных файлов проекта в рабочей директории создается каталог, имя которого совпадает с именем файла конфигурации. При этом базы каналов сохраняются в файлах с расширениями .dbb.

В Редакторе представления данных разрабатывается графическая часть проекта системы управления. При этом создается статичный рисунок технологического объекта, а затем поверх него размещаются динамические формы отображения и управления. Среди этих форм присутствуют такие, как поля вывода численных значений, графики, гистограммы, кнопки, области ввода значений и перехода к другим графическим фрагментам и т. д.



редактор
представления
данных

Вход в Редактор представления данных осуществляется либо двойным нажатием ЛК на соответствующем ярлыке в папке ТРЕЙС МОУД 5.0, либо запуском исполнительного модуля rcmap.exe. Вид окна редактора представления данных показан на рисунке 6.

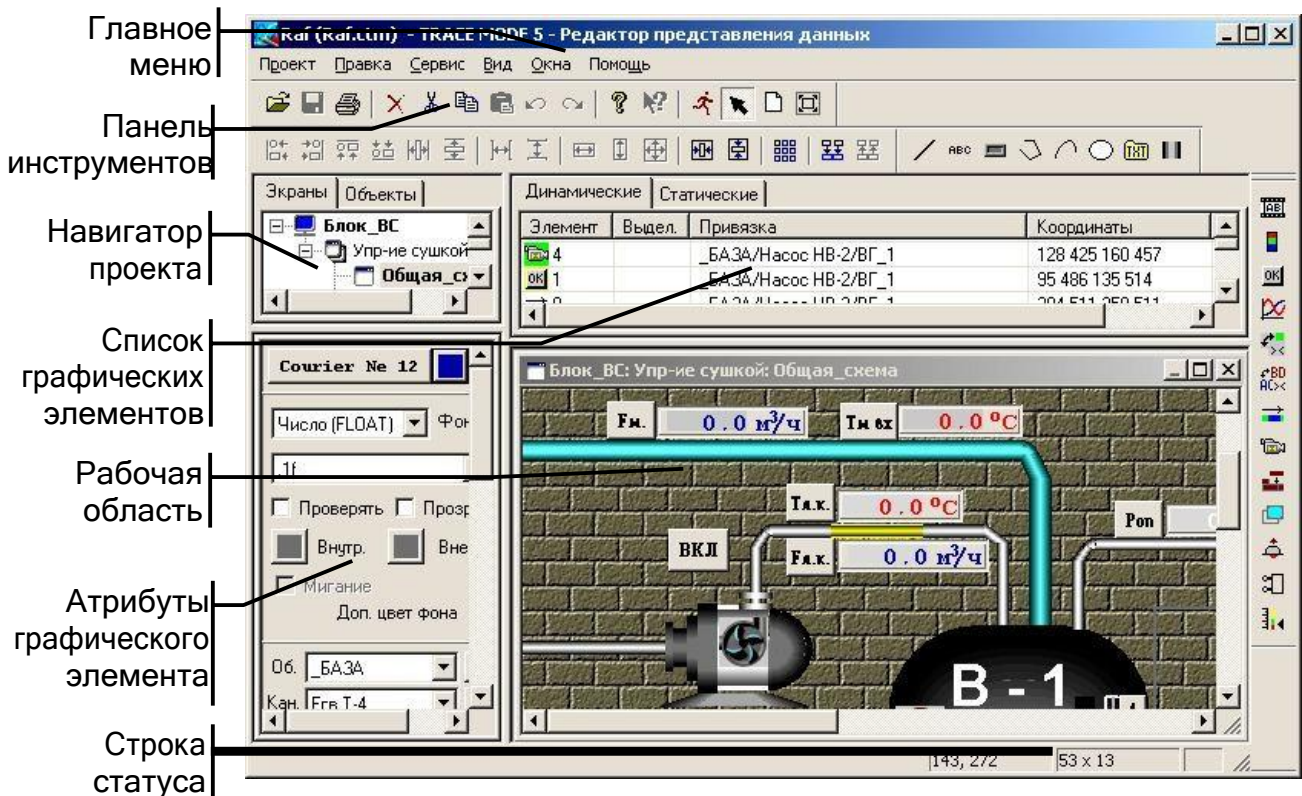


Рисунок 6 - Вид окна редактора представления данных

В этом редакторе разрабатывается графическая часть проекта системы управления. При этом создается статичный рисунок технологического объекта, а затем поверх него размещаются динамические формы отображения и управления. Среди этих форм присутствуют такие, как поля вывода численных значений, графики, гистограммы, кнопки, области ввода значений и перехода к другим графическим фрагментам и т. д.

Все формы отображения информации, управления и анимационные эффекты связываются с информационной структурой, разработанной в редакторе базы каналов.

Графические базы узлов проекта, созданные в редакторе представления данных, сохраняются в файлах с расширением dbg. Их сохранение осуществляется в соответствующих директориях проекта.

Исполнительные модули (RUNTIME) – это программы, под управлением которых запускается АСУ, созданная в инструментальной системе. В группу исполнительных модулей входят следующие программы:

- Монитор реального времени – **MPB**;
- Монитор реального времени – **NetLink MPB**;
- Монитор реального времени – **NetLink Light**;
- Монитор создания АРМ администратора – **SUPERVISOR**;
- Монитор глобального архива – **Глобальный регистратор**;
- Микро монитор реального времени – **Микро MPB**;
- Микро MPB с поддержкой обмена через модем – **Микро MPB Модем +**.

Первые пять мониторы предназначены для организации работы верхнего и административного уровней АСУ. **Микро MPB** и **Микро MPB Модем +** служат для работы в контроллерах нижнего уровня систем управления, при условии наличия в них операционной системы MS DOS.

Монитор реального времени MPB служит для запуска АРМ операторов, осуществляющих с его помощью супервизорный контроль и управление технологическим процессом. Под управлением MPB выполняются следующие задачи:

- *запрос данных о состоянии технологического процесса с контроллеров нижнего уровня по любому из встроенных протоколов или через драйвер;*
- *передача на нижний уровень команд управления по любому из встроенных протоколов или через драйвер;*
- *обмен данными с платами УСО;*
- *сохранение данных в архивах;*
- *обмен по сети с удаленными MPB;*
- *обмен по коммутируемым линиям с удаленными MPB;*
- *передача данных по сети на следующий уровень АСУ;*
- *представление оператору графической информации о состоянии технологического процесса;*
- *автоматическое и супервизорное управление технологическим процессом;*
- *обмен данными с другими приложениями WINDOWS через DDE/NetDDE/OPC;*
- *обмен с базами данных через ODBC;*
- *и другие функции.*

Монитор реального времени **NetLink MPB** может применяться только в составе систем управления, где обмен данными между узлами системы осуществляется по локальной сети.

Монитор реального времени **NetLink Light MPB** позволяет создавать дополнительные рабочие места операторов. Он не поддерживает функций обра-

ботки данных и автоматического управления. Этот монитор является дополнительной графической консолью, которая может быть подключена с удаленного компьютера к запущенному MPB. Таким образом, имея в сети один монитор реального времени, можно, используя NetLink Light, создать в сети требуемое количество рабочих мест, совершенно равноправных с MPB по функциям отображения и супервизорного управления.

SUPERVISOR предназначен для создания АРМ администратора. Он может получать данные только из архивов. Это могут быть либо локальные архивы MPB или NetLink MPB, либо глобальные архивы, которые создает Глобальный регистратор. С помощью SUPERVISOR невозможно осуществлять оперативное управление процессом. По функциям организации представления данных SUPERVISOR похож на NetLink Light. Для него требуется только создание графической базы (dbg файлы). Существенное отличие SUPERVISOR заключается в том, что он получает от MPB или Глобального регистратора архивные значения каналов, а NetLink Light - текущие данные.

Глобальный регистратор (ГР) - это специальный монитор ТРЕЙС МОУД, предназначенный для ведения глобального архива по всему проекту. Он архивирует данные посылаемые ему по сети мониторами реального времени. После сохранения данных в архив Глобальный регистратор может передавать их для просмотра мониторам SUPERVISOR. Архив Глобального регистратора реализует технологию хранилища данных. Он фиксирует значения технологических параметров при их изменении. В рамках одного проекта может поддерживаться только один такой архив.

Микро монитор реального времени (Микро MPB) предназначен для управления задачами сбора данных и управления в контроллерах нижнего уровня АСУТП. Он может быть использован в любых IBM-совместимых контроллерах. По возможностям математической обработки, управления, обмена данными с другими мониторами ТРЕЙС МОУД Микро MPB идентичен монитору реального времени. Однако для него не реализованы функции графического вывода информации. Задачи для Микро MPB разрабатываются в редакторе базы каналов. Поэтому при использовании IBM-совместимых контроллеров в рамках ТРЕЙС МОУД реализуется единая линия программирования задач верхнего и нижнего уровней систем управления.

Микро MPB Модем + идентичен по функциям с Микро MPB. Единственным его отличием является встроенная поддержка обмена данными с помощью модема по коммутируемым каналам, что позволяет использовать Микро MPB для создания удаленных пунктов сбора информации, обменивающихся данными через телефонную сеть.

Драйверы требуется, если протокол обмена данными с используемым устройством не встроен в систему. Основной функцией драйвера является обеспечение связи ТРЕЙС МОУД с внешними устройствами. Это могут быть устройства сбора, хранения, обработки, передачи данных (контроллеры, УСО, другой компьютер и т.д.) или какие-либо другие устройства. Драйвер осуществляет согласование форматов данных ТРЕЙС МОУД и аппаратуры, для связи с которой он разработан.

3.2. Основы проектирования базы каналов

Как можно понять из названия самого первого редактора инструментальной системы, канал является основным понятием системы TRACE MODE 5.0. *Канал - это базовое понятие системы.* Данные с внешних устройств записываются в каналы. Данные из каналов посылаются на внешние устройства и выводятся в различных формах на экран монитора. В каналы оператор заносит управляющие данные. Значения из каналов записываются в архивы, оперативные отчеты и все генерируемые документы. В каналах осуществляется преобразование данных. Меняя значения на системных каналах, можно управлять выводимой на экран информацией, звуковыми эффектами, архивами и т.д., то есть всей системой.

Что бы разобраться каким образом информация поступает в канал, как в нём преобразуется и как затем отображается на экране диспетчера, рассмотрим вначале основные понятия изучаемой SCADA-системы.

3.2.1 Основные понятия редактора базы каналов

Основные понятия, используемые в Трейс Моуд при разработке проектов автоматизации, описывают структуру информационных потоков, структуру данных и процедуры их обработки, основные элементы и подходы к разработке средств визуализации состояния технологического процесса и управления им (создание человекомашинного интерфейса). К ним относятся: проект; узел; канал; объект; алгоритм обработки; автопостроение; экран.

Проект системы управления в Трейс Моуд - это совокупность всех математических и графических компонентов ПО для операторских станций и контроллеров одной АСУ ТП, объединенных информационными связями и единой системой архивирования.

Проект включает в себя ПО для всех используемых АРМ и контроллеров - узлов проекта. Математические и графические компоненты их программного обеспечения одновременно загружаются в редакторы. Это делает проект прозрачным для разработчика и облегчает настройку взаимодействия узлов проекта и обмена данными.

Проект может быть масштабным и содержать сотни узлов, а может включать в себя только один контроллер или операторскую станцию.

Структура проекта описывается и редактируется в редакторе базы каналов (рисунок 5) и сохраняется в файле конфигурации проекта. Структура проекта включает в себя перечень узлов – операторских станций и контроллеров, работающих под управлением ТРЕЙС МОУД.

Создание и редактирование *графической составляющей проекта* осуществляется в редакторе представления данных (рисунок 6). Структура проекта представлена здесь в виде дерева, корневыми элементами которого являются имена узлов, а вложенными - имена групп и экранов.

Узел – это любое устройство, в котором запущено программное обеспечение ТРЕЙС МОУД, реализующее серверные функции. Это может быть контроллер, операторская или архивная станция.

Узлы одного проекта могут быть связаны между собой по локальной сети, по последовательным интерфейсам, по коммутируемым линиям или по радиоканалу.

Чтобы создать новый узел, надо выполнить команду «Создать» из меню «Узлы». При этом на экране появится диалог (рисунок 7), в котором для создаваемого узла надо указать тип и задать имя (до 20 символов без пробелов). Тип узла определяет, какой монитор будет использован для его запуска, а также механизмы автопостроения его базы каналов.

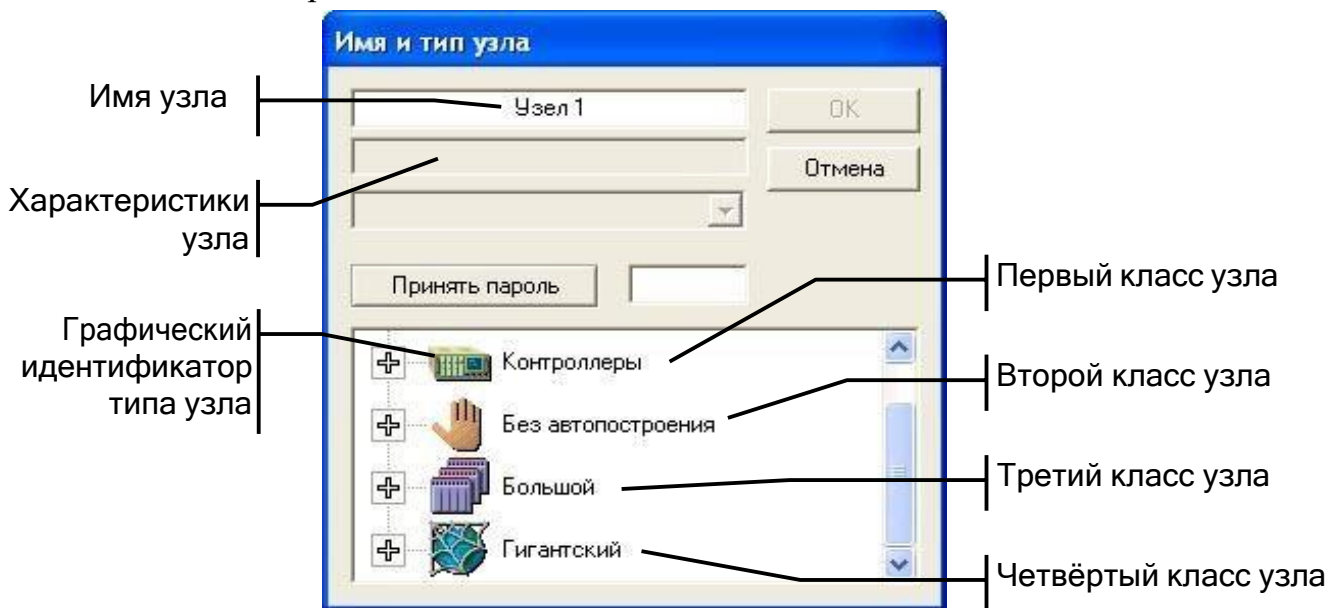


Рисунок 7 – Окно создания узла

Все узлы подразделяются на четыре класса. Класс узла определяет его информационную мощность и уровень системы управления, на котором он будет использоваться.

Первый класс включает в себя перечень узлов нижнего уровня систем управления. Для запуска этих узлов используются микроМРВ или микроМРВ Модем+. Все эти узлы имеют следующую информационную мощность: 4096 каналов, 1024 объекта, 1024 FBD-программы.

Второй класс включает в себя типы узлов, которые имеют малую и среднюю информационную мощность. Механизм автопостроения по умолчанию для них не предусмотрен. Узлы данного класса следует использовать для создания АРМ оператора небольших промышленных установок.

Третий класс содержит узлы большой информационной мощности. Их можно использовать для АРМ операторов и администраторов практически для любых объектов, за исключением, может быть, самых крупных. Методом автопостроения у них является связь с удаленными узлами. Тип связи определяется типом узла. Это может быть последовательный интерфейс (протокол M-LINK.), локальная сеть (протокол I-NET или OPC). Кроме того, в этот класс входит узел глобального регистратора, используемого для создания архивных станций. Для

запуска этих узлов могут использоваться MPB, NetLink MPB и монитор глобального регистратора (для архивных станций) соответствующей мощности.

Четвертый класс включает в себя самые мощные типы узлов. В него входят узлы для создания АРМ операторов и администраторов, архивных станций и глобальных регистраторов.

После создания нового узла в рабочем поле редактора появляется его изображение, включающее в себя его имя и графический идентификатор, соответствующий установленному типу (рисунок 8).



Рисунок 8 – Обозначение узла

Для настройки *параметров узла* или редактирования его базы каналов надо выделить его в окне структуры проекта. Чтобы настроить параметры выделенного узла, надо нажать правой кнопкой мыши (ПК) на его изображении, а для перехода к редактированию базы каналов - ЛК (рисунок 9).

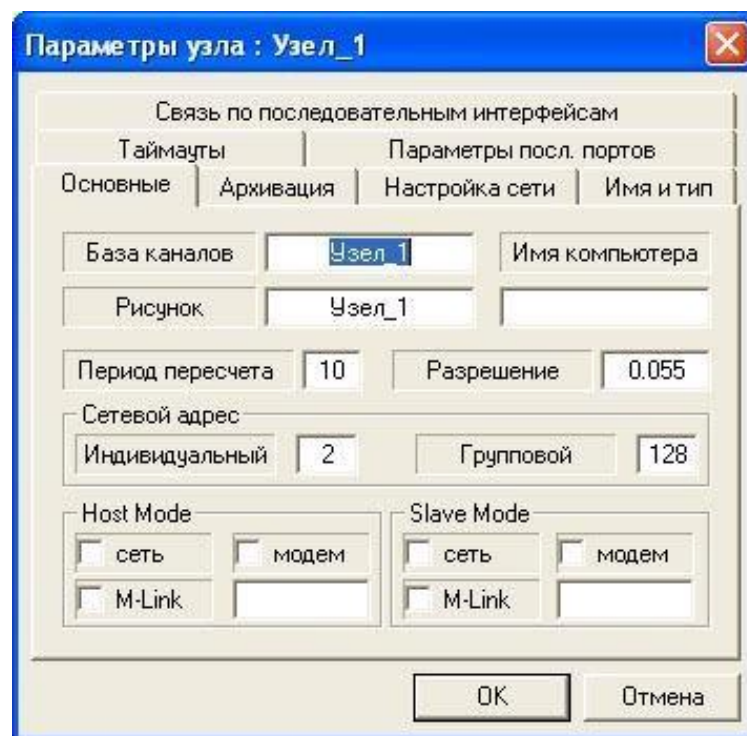


Рисунок 9 – Окно для настройки параметров узла

Для каждого узла настраиваются:

- скорость обработки параметров;
- параметры архивации событий;
- права доступа;
- сетевые параметры для связи с другими узлами проекта;
- параметры связи с нижним (контроллерным) уровнем, и т.д.

Для каждого узла проекта создается база данных реального времени. В ТРЕЙС МОУД она называется **базой каналов** и имеет иерархическую структуру. Основным элементом базы является **канал**.

Базой каналов в ТРЕЙС МОУД называется совокупность всех каналов, математических объектов, FBD-программ и IL-программ, созданных для одного конкретного узла.

Базы каналов сохраняются в файлы с расширением dbb. По умолчанию файлу базы каналов создаваемого узла присваивается имя, образованное из имени узла. Имена базы каналов и рисунка не должны содержать пробелов. Изменение имени файла базы каналов для узла доступно из бланка «Основные» диалога «Параметры узла».

При переходе к редактированию базы каналов запускается процедура автопостроения. После настройки новых связей для автопостроения осуществляется вход в окно редактирования узла, показанный на рисунке 5.

Каналы одной базы могут группироваться по заданным признакам или произвольно. Такие группы называются **объектами базы каналов**.

Объектом базы каналов называется совокупность каналов, которой приспан определенный набор свойств и атрибутов. Объекты допускают групповые операции над атрибутами каналов в процессе их редактирования и при работе в реальном времени.

Объект базы каналов имеет следующий набор параметров:

- *имя,*
- *графический идентификатор,*
- *подчиненность,*
- *загружаемость*
- *состояние его каналов при старте.*

Они настраиваются в диалоге «Параметры объекта» (рисунок 10).

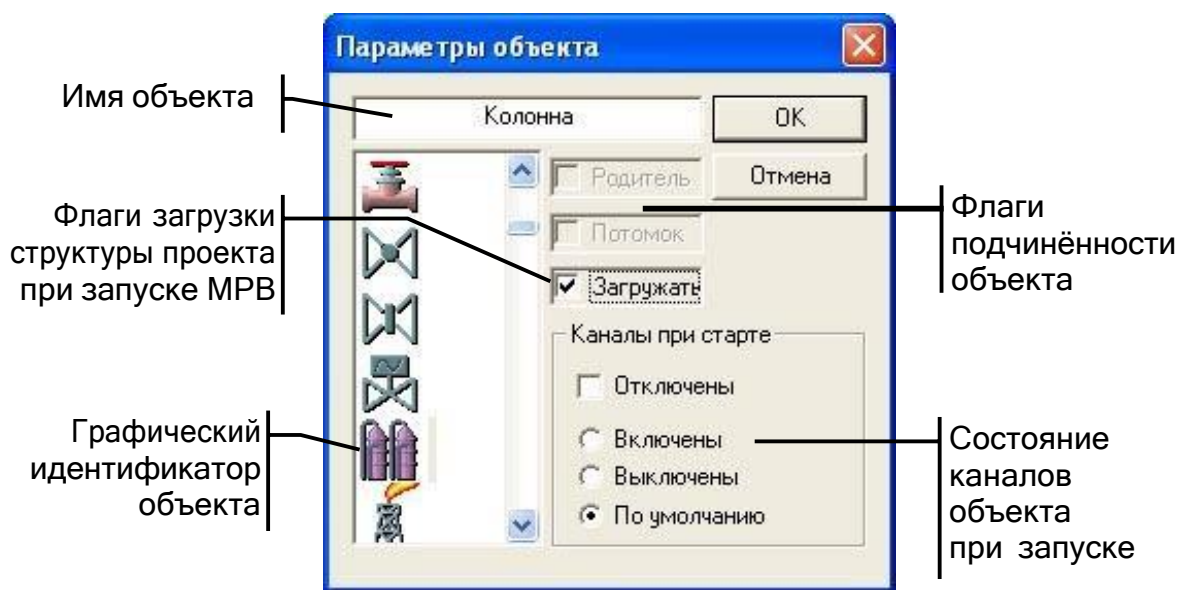


Рисунок 10 – Окно параметров объекта

3.2.2 Автопостроение баз каналов

В рамках разработки проекта автоматизации необходимо описать информационные потоки: для контроллеров надо создать и настроить каналы обмена данными с платами УСО, а для АРМ - каналы обмена данными с контроллерами и другими узлами проекта.

В крупных проектах задача заполнения баз, описывающих адреса источников и приемников данных, является очень трудоемкой. Это может привести к ошибкам, обусловленным объемом и рутинностью данной работы.

Для облегчения этой работы имеются механизмы автопостроения баз каналов:

- создание каналов обмена данными с удаленными узлами текущего проекта;
- создание каналов обмена данными с платами УСО;
- создание каналов обмена данными с контроллерами нижнего уровня (не РС-совместимыми);
- создание каналов при импорте баз данных;
- создание каналов обмена с OPC-серверами;
- создание каналов обмена данными с каналами объекта удаленного узла.

На рисунке 11 показаны диалоги настройки автопостроения для посадочных мест контроллера и для связи с удаленными узлами.

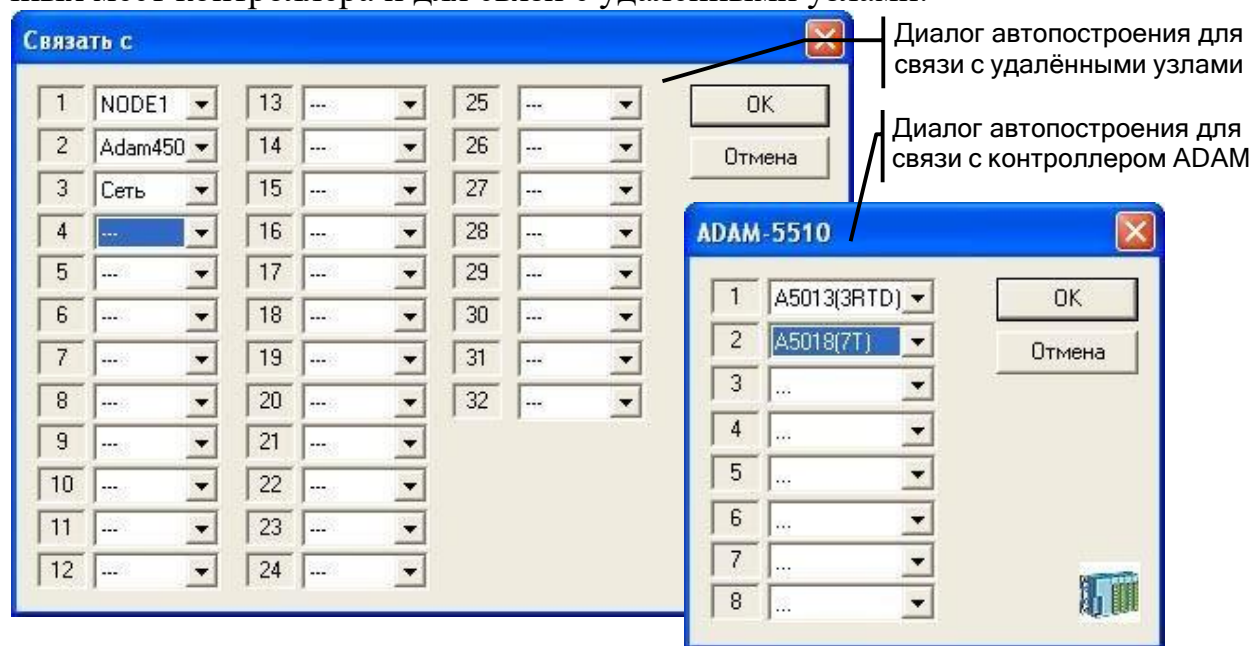


Рисунок 11 - Диалоги настройки автопостроения

Механизм «Автопостроения связей с узлами» автоматически запускается для любого узла класса «Большой» и «Гигантский» при первом входе в редактирование его базы каналов. Чтобы запустить этот механизм в следующий раз, надо выполнить команду! «Автопостроить» из меню «Узел».

Механизм «Автопостроение связей с платами УСО» определен для узлов класса «Контроллеры». Он автоматически запускается при первом входе в редактирование базы каналов любого узла данного класса.

Вид диалога настройки автопостроения зависит от типа узла (марки контроллера). В нем надо указать, в каких посадочных местах какие платы УСО ус-

тановлены. После этого в базе каналов создаются объекты, имена которых воспроизводят названия соответствующих плат УСО. В них помещаются каналы, настроенные на обмен данными с этими платами.

После настройки посадочного места изменить тип платы УСО невозможно. Чтобы его изменить, надо удалить соответствующий объект. При этом посадочное место освобождается.

Связи для автопостроения каналов в случае использования механизма «Автопостроение связей с контроллерами» описываются в бланке «Связь» по последовательным интерфейсам диалога «Параметры узла». Этот механизм автопостроения может применяться для любых узлов, кроме сетевых и архивных станций.

Механизм «Автопостроение по объектам» запускается командой «Связать с объектом узла» из меню «Узел». На экране надо указать узел и объект, с которым требуется построить связь, и выбрать тип связи. После этого в редактируемом узле появится объект, имя которого образуется из имен выбранного узла и объекта. В нем создаются каналы, настроенные на обмен с каналами указанного объекта.

Чтобы установить тип связи, можно выбрать один из следующих вариантов:

- *M-Link - обмен по протоколу M-Link без передачи времени изменения запрашиваемого параметра;*
- *M-Link(T) - обмен по протоколу M-Link с передачей времени последнего изменения запрашиваемого параметра;*
- *AutoNet - передача данных по сети;*
- *OPC - передача данных через механизм OPC;*
- *контроллер - при наличии в указанном объекте каналов обмена с контроллерами или платами УСО в редактируемом узле они воспроизводятся со всеми настройками.*

По умолчанию протокол определяется типом узла.

Чтобы запустить процедуру «Автопостроения обмена с OPC-серверами» следует, находясь в окне объектов настраиваемого узла, выполнить команду «Связать с OPC-сервером» из меню «Узел». При этом на экране появится диалог «Выбор сервера OPC».

При нажатии кнопки «Добавить» на экран выводится диалог, в котором можно выбрать сервер, зарегистрированный на локальной машине или на любом компьютере, присутствующем в сети. Указанный сервер добавляется в список предыдущего диалога.

При нажатии кнопки «Удалить» выделенный в списке сервер удаляется из окна. Кнопка «Изменить» используется для замены выделенного сервера. Она выводит на экран тот же диалог, что и кнопка «Добавить». Выбранный в нем сервер заменяет текущий.

Чтобы создать каналы для обмена с выделенным в списке сервером, надо нажать кнопку «Выбрать».

Трейс Моуд позволяет создавать базы каналов при импорте из технологической базы. Существует два способа импорта: - из текстовых файлов с разделителями; - через интерфейс ODBC из зарегистрированных источников данных.

3.2.3 Канал TRACE MODE

Добавление, удаление и редактирование каналов осуществляется из диалогового окна **Каналы объекта** любого из объектов (рисунок 12). В этом диалоге выводится список каналов, присутствующих в данном объекте, и информация по некоторым их атрибутам.

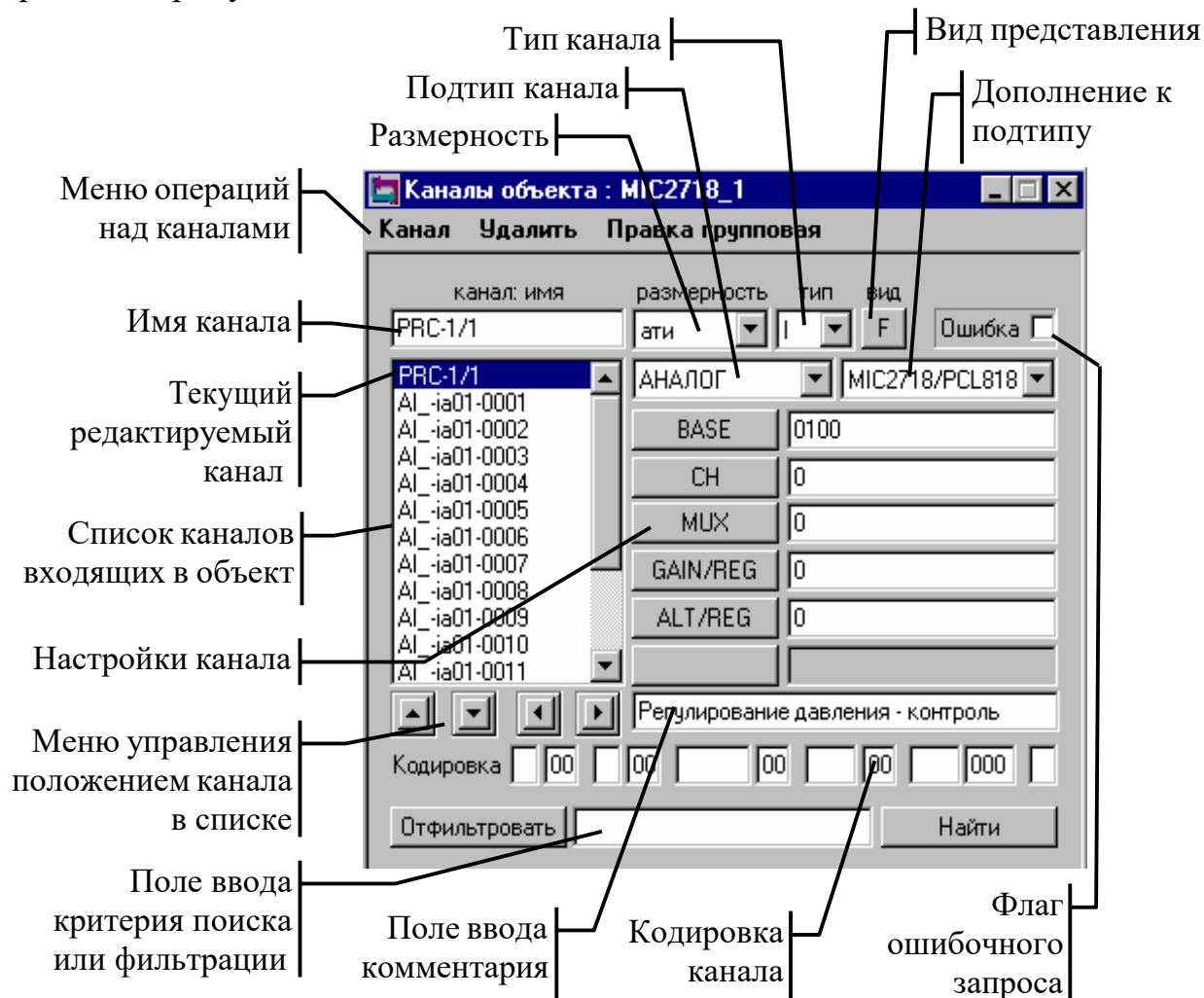


Рисунок 12 – Диалоговое окно «Каналы объекта»

Канал – это структура, состоящая из набора **переменных** и **процедур**, имеющая **настройки** на внешние данные, **идентификаторы** и **период** пересчета ее переменных.

Идентификаторами канала являются: **-имя, -комментарий, -кодировка**. Кроме того, каждый канал имеет числовой идентификатор, используемый внутри системы для ссылок на этот канал.

Среди переменных канала выделяются четыре основных значения:

- входное, - аппаратное, - реальное, - выходное.

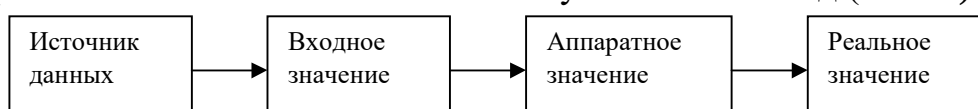
И ряд дополнительных переменных:

**- шесть границ; - гистерезис; - настройки процедур обработки;
- начальные параметры; - флаги архивирования и др.**

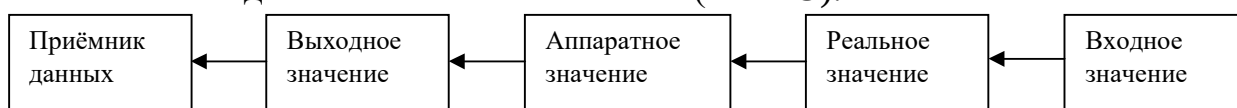
С помощью **настроек** входное значение канала связывается с источником данных, а выходное – с приемником. С помощью **процедур** входное значение канала преобразуется в аппаратное, реальное и выходное.

В зависимости от направления движения информации, т.е. от внешних источников (данные с контроллеров, УСО или системные переменные) в канал или наоборот, каналы подразделяются на *входные (тип INPUT)* и *выходные (тип OUTPUT)*.

Входной канал (тип Input - I) запрашивает данные у внешнего источника (контроллер, другой МРВ и пр.) или получает значение какой-нибудь системной переменной (например, код нажатой клавиши, счетчик ошибок обмена и пр.). Запрашиваемое каналом значение поступает на его вход (тип - I).



Выходной канал (тип Output - O) передает данные приемнику. Приемник может быть внешним (значение переменной в контроллере, в другом МРВ и пр.) или внутренним - одна из системных переменных, управляющая работой данного МРВ (номер проигрываемого звукового файла, номер экрана, выводимого на монитор и пр.). И внешние и внутренние приемники данных всегда связываются с выходными значениями каналов (тип - O).



Процедурами канала являются:

- **масштабирование** (умножение и смещение),
- **фильтрация** (подавление пиков, апертура (зона нечувствительности) и сглаживание),
- **логическая обработка** (предустановка, инверсия, контроль сочетаемости),
- **трансляция** (вызов внешней программы) и
- **управление** (вызов внешней программы).

*Переменные, настройки и идентификаторы канала образуют список его **атрибутов**.* Часть из них задается в редакторе базы каналов и не может быть изменена в реальном времени. Другие – могут иметь начальные значения и доступны для изменения.

Двойное нажатие ЛК на имени любого канала в списке выводит на экран диалог **Реквизиты** (рисунок 13 и 14), где осуществляется настройка большинства атрибутов этого канала и отладка обработки в нем данных.

Этот диалог позволяет ввести частоту и фазу пересчета канала, настроить и отладить первичную обработку сигнала в канале, ввести шкалу и аварийные границы канала, настроить вызов программ для более глубокой обработки сигнала или управления, а также настроить ряд других параметров канала.

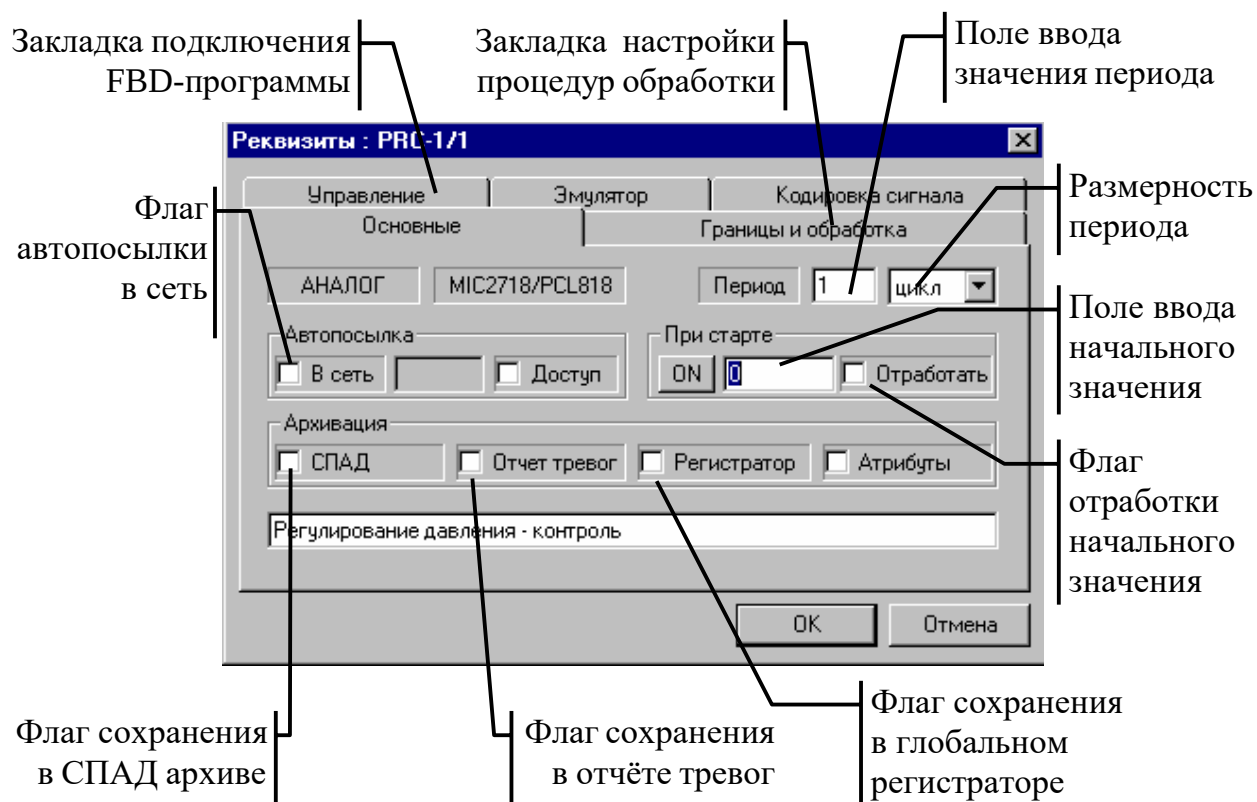


Рисунок 13 – Закладка «Основные» в окне реквизитов канала

Любой канал обязательно имеет следующий набор атрибутов:

Начальные установки: - *состояние*;
 - *значение*;
 - *кодировка*;
 - *размерность*.

Кроме того, для каждого канала можно установить ряд флагов, определяющих режимы пересчета и архивирования данных: - *флаг В СЕТЬ*;

- *флаг ДОСТУП*;
- *флаг ОТРАБОТАТЬ*;
- *флаг АРХИВ*;
- *флаг ТРЕВОГИ*;
- *флаг РЕГИСТРАТОР*;
- *флаг ЖУРНАЛ*.

Атрибут вид представления указывает, для работы с каким типом данных предназначен этот канал: аналоговые или дискретные. Набор атрибутов у этих каналов несколько отличается.

3.3. Основы разработки графического интерфейса

Все экраны в графических базах ТРЕЙС МОУД собраны в группы. В принципе можно использовать только одну группу в графической базе и размещать все экраны в ней. Однако в этом случае при большом количестве экранов будет сложно ориентироваться в них. Поэтому в редакторе представления данных реализована группировка экранов. Эту группировку удобно использовать исходя из функционального назначения экранов. Например, в одну группу можно собрать мнемосхемы, в другую - экраны настройки регуляторов, в третью - обзорные экраны и т. п. Можно также разбивать экраны, на группы исходя из стадий или участков автоматизируемого процесса.

Что бы разобраться в многообразии элементов редактора представления данных, выделим для начала основные понятия.

3.3.1 Основные понятия редактора представления данных

Графическая часть - это совокупность всех экранов для представления данных и супервизорного управления, входящих в графические базы узлов проекта.

Графическая база - это совокупность всех экранов, графических объектов, элементов рисования и форм отображения, используемых для отдельного узла проекта.

Графические базы создаются для всех операторских станций проекта.

Имя файла графической базы для узла задается в редакторе базы каналов. Для этого используется бланк «Основные» диалога «Параметры узла».

Графическая база любого узла состоит из списка экранов и размещенных на них графических элементов. Все графические элементы можно разделить на два типа: элементы рисования; - формы отображения.

Экраны в графических базах подразделены на группы. В принципе, можно использовать только одну группу в графической базе и размещать все экраны в ней. Однако в этом случае при большом числе экранов будет сложно ориентироваться в них. Поэтому в редакторе представления данных реализована группировка экранов.

Для доступа к графической базе любого узла проекта ее надо загрузить в редактор. Это осуществляется командой «Загрузить» из меню узлов бланка «Экраны» навигатора проекта. Перед выполнением этой команды надо выделить требуемый узел в этом бланке. После загрузки содержание графической базы выводится в бланке «Экраны» в качестве вложенных элементов для выбранного узла.

Экран - это графическое пространство фиксированного размера, на котором размещаются статический рисунок и формы отображения. Экран - это минимально адресуемая графическая информация, выводимая на монитор.

Одновременно на монитор может выводиться только один экран. Другие экраны можно просматривать на текущем экране с помощью окон - специальных форм отображения.

Каждый экран имеет свое имя и набор атрибутов (настроек). К таким атрибутам относятся:

- «Размер»;
- «Цвет фона»;
- «Обои»;
- «Права доступа»;
- «Спецификация окна просмотра отчета тревог».

Первые три атрибута по умолчанию устанавливаются по общим настройкам графической базы. Права доступа к экрану устанавливаются полными.

Разработка графических экранов осуществляется путем размещения на них графических элементов. Различают статические и динамические элементы.

На экранах можно размещать комплексы статических и динамических элементов, оформленных как графические объекты, используемые для тиражирования готовых решений в области создания интерфейса оператора.

Графическим объектом называется совокупность форм отображения и элементов рисования, которая оформлена как единый графический элемент. Оформленные в виде объектов типовые графические фрагменты могут вставляться в экраны графических баз любых проектов.

Существует два типа графических объектов: «Объект» и «Блок». Первый из них может ссылаться на 256 каналов, а второй - только на один.

Для создания и редактирования объектов используются такие же окна, как и при работе с экранами. Разработка объектов идентична процессу разработки экрана. Различие заключается лишь в настройке форм отображения на каналы. В объекте формы отображения связываются с его внутренними каналами. Эти каналы при размещении объекта на экране настраиваются на реальные каналы редактируемого узла.

Трейс Моуд позволяет осуществлять ряд операций с графическими объектами: копирование, сохранение и вставка в другие проекты или графические базы того же проекта, вывод в отдельные окна на других экранах и т. д.

Для хранения графических объектов используются графические библиотеки. Каждая библиотека имеет имя и список включенных в нее объектов. Чтобы в дальнейшем использовать созданную библиотеку, ее надо сохранить в файле. Для получения доступа к сохраненной ранее библиотеке надо ее загрузить в редактор представления данных.

3.3.2 Графические элементы рисования

Для разработки неизменяемой составляющей экранов используются статические графические элементы - **элементы рисования**. Они не связываются со значениями каналов и не реагируют на нажатие мыши. Их назначение - графическое оформление экрана.

Чтобы разместить элемент рисования на редактируемом экране, его надо сначала выбрать, затем настроить его атрибуты и выполнить операции по его масштабированию и позиционированию на экране.

Большинство элементов рисования могут быть динамизированы. В этом случае они будут выступать как формы отображения.

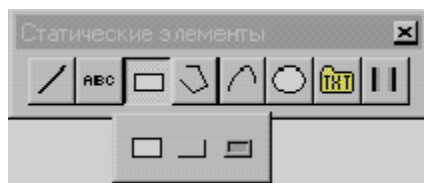
Выбор элементов рисования:

В редакторе представления данных для выбора элементов рисования предусмотрена специальная инструментальная панель. Эта панель имеет следующий вид



Каждая из иконок, присутствующих в этой панели, обозначает группу элементов рисования. При первом нажатии на любой из них на экран выводится диалог настройки атрибутов элемента рисования, текущего в данной группе.

Повторное нажатие на той же иконке открывает меню выбора элементов рисования данной группы



После выбора в этом меню требуемого элемента группы его изображение появится в инструментальной панели:  Как видно, для группы Прямоугольники установлен другой элемент рисования.

Описание элементов рисования:

Редактор представления данных позволяет использовать в графических экранах следующие графические элементы для разработки статичной части графических экранов:

- Линия;
- Статический текст;
- Прямоугольник/кнопка/рамка;
- Ломаная;
- Кривая;
- Эллипс/дуга;
- Внешние ресурсы(файлы);
- Объемная графика.

Размещение элементов рисования:

После настройки атрибутов элемента рисования можно перейти к его размещению на редактируемом экране. Для этого надо перевести курсор в область экрана и нажатием ЛК установить точку привязки.

Дальнейшие действия **отличаются** в зависимости от размещаемого элемента рисования:

- для большинства из них после установки точки привязки дальнейшее перемещение мыши выводит на экран их образ. Его размер определяется координатами точки привязки и текущего положения курсора. Повторное нажатие ЛК приводит к размещению элемента рисования на экране.
- некоторые элементы рисования устанавливаются в несколько этапов (например, многоугольник). Для этих элементов каждое нажатие ЛК по-

сле установки точки привязки является завершением этапа (для многоугольника - координаты вершины). Завершение размещения для этих элементов рисования осуществляется по нажатию ПК.

- *Размещение растровых изображений* осуществляется в момент установки точки привязки. Размер этого элемента рисования определяется размером размещаемого растрового изображения.

При установке всех графических элементов на экране отсчет координат осуществляется от левого нижнего угла экрана. Текущие координаты курсора выводятся в строке статуса.

Порядок расположения:

Любые элементы рисования всегда будут закрываться динамическими элементами – формами отображения.

Если области размещения элементов рисования пересекаются, то сверху будет находиться последний по порядку установки. Чтобы изменить последовательность элементов рисования, надо перейти в режим редактирования, выделить требуемый элемент и нажать на клавиатуре одно из следующих сочетаний клавиш:

CTRL+SHIFT+B – поверх всех элементов рисования;

CTRL+B – ниже всех элементов рисования.

3.3.3 Графические элементы форм отображения

Выбор **форм отображения** для их размещения на экранах осуществляется с помощью специальной инструментальной панели



Описание форм отображения:

Редактор представления данных позволяет использовать в графических экранах следующие динамические элементы для отображения информации о ходе технологического процесса и управления:

- *Текстовая форма отображения.*
- *Гистограммы (прямоугольная и произвольной формы).*
- *Кнопки различных типов.*
- *Тренды аналоговых и дискретных параметров.*
- *Цветовые индикаторы.*
- *Графические и текстовые индикаторы.*
- *Бегущие дорожки.*
- *Воспроизведение AVI-файлов.*
- *Отображение экранов в окнах.*
- *Свободные формы.*

Внимание! Форма отображения гистограмма произвольной формы и все цветовые индикаторы создаются путем динамизации уже размещенных элементов рисования. После настройки атрибутов этих форм надо перевести курсор в

рабочую область редактора, подвести его к нужному элементу и после изменения вида курсора нажать ЛК. Для гистограммы произвольной формы выбранный элемент обязательно должен быть замкнутым.

Каждая из иконок, присутствующих в этой панели, обозначает группу форм отображения. Первое нажатие ЛК на любой из них выводит на экран диалог настройки атрибутов формы, текущей в данной группе. Повторное нажатие на той же иконке открывает меню выбора форм отображения данной группы (так же как и для статических элементов).

После выбора формы отображения и настройки ее атрибутов можно перейти к размещению этой формы на редактируемом экране графической базы.

Механизм размещения:

Для размещения формы отображения (ФО) на экране следует перевести курсор в требуемую область экрана. Затем нажатием ЛК надо установить точку привязки формы отображения. Дальнейшее перемещение мыши приведет к выводу на экран контурного прямоугольника. Его размеры будут определяться координатами точки привязки и текущими координатами курсора. Контурный прямоугольник обозначает границы устанавливаемой формы отображения. Следующее нажатие ЛК зафиксирует размещаемую ФО на экране. Ее размер будет соответствовать величине контурного прямоугольника.

Некоторые ФО размещаются на экране при установке точки привязки..

Порядок расположения:

Любые формы отображения всегда размещаются поверх статических элементов. Если области размещения ФО пересекаются, то сверху будет находиться последняя по порядку установки.

4. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Лабораторная работа выполняется на ПК "IBM PC" в два этапа. Вначале в ознакомительном режиме по предложенным в справке урокам. Затем необходимо визуализировать технологический процесс, выданный в лабораторной работе №1.

Задание №1.

Для ознакомления с системой TRACE MODE 5.0 необходимо выполнить следующую последовательность действий:

- 1) С помощью методического пособия ознакомиться со структурой основных редакторов системы TRACE MODE 5.0. В отчёте отметить: - название работы, - цель работы, - название и назначение основных редакторов.

- 2) Включить ПК и двойным щелчком левой кнопки "мыши" открыть папку

- 3) Запустить справочную систему с помощью ярлыка

- 4) В открывшемся окне выбрать пункт «Быстрый старт», и открыть пункт «Урок 1». Созданный проект сохраните в отдельном каталоге, поименовав его своей фамилией.

- 5) Следуя указаниям справки последовательно выполнить три предлагаемых урока, записывая в тетрадь последовательность создания проекта.

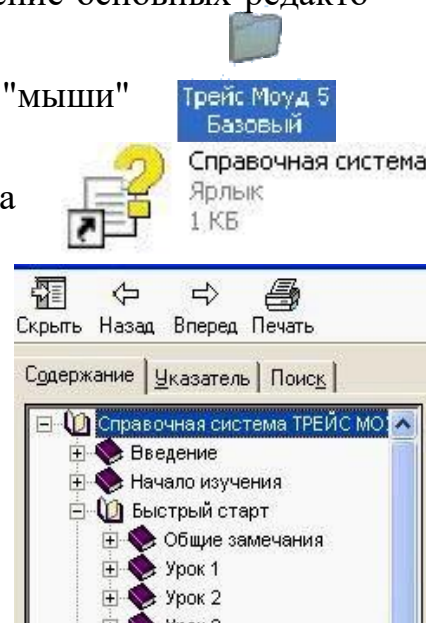
- 6) В созданную ранее папку скопируйте проект «БЫСТРЫЙ_СТАРТ2», находящийся в каталоге «C:\TraceMode5_Basic\». Копировать необходимо два элемента: а) файл «БЫСТРЫЙ_СТАРТ2.ctm»
б) каталог «БЫСТРЫЙ_СТАРТ2»

- 7) Следуя указаниям справки последовательно выполнить уроки 4, 5 и 6, записывая в тетрадь темы, рассматриваемые в каждом уроке. Урок 6 возможно выполнить только наполовину в связи с отсутствием сервера документирования в демо-версии системы TRACE MODE 5.0.

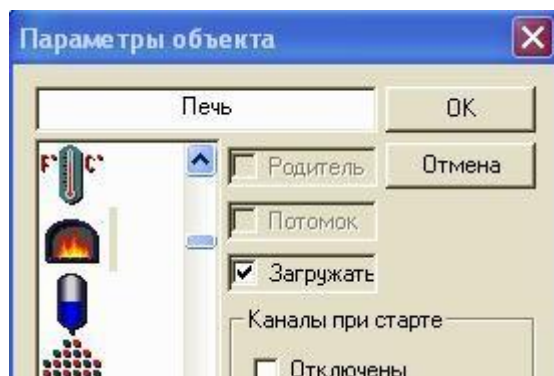
Задание №2.

Визуализация технологического процесса производится в следующей последовательности:

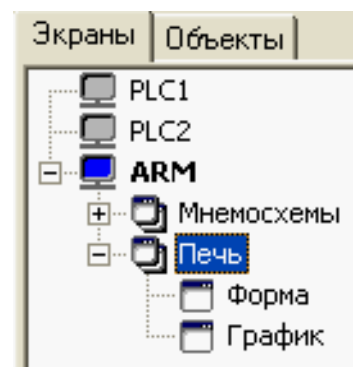
- 1) После завершения уроков и оформления отчёта откройте в редакторе базы каналов редактируемый вами проект «БЫСТРЫЙ_СТАРТ2».
- 2) Создайте в узле «АРМ» объект, пометив его точкой как «Пустой».



- 3) Войдите в параметры объекта, выделив его и нажав на нём правой клавишей мыши. В открывшемся окне введите новое имя, соответствующее имени вашего процесса выполненного в лабораторной работе №1, выберите подходящую для него картинку и поставьте галочку напротив слова «Загружать».



- 4) Войдите в созданный объект и в меню «Каналы» выберите пункт «Добавить». В открывшемся окне выберите имя любого канала, например «УРОВЕНЬ 1» и нажмите кнопку «ОК».
- 5) Поставьте курсор на добавленный канал и в меню «Каналы» выберите пункт «Создать по образцу». В поле «Имя канала» введите имя одной из переменных, используемых в вашем проекте.
- 6) Аналогичным образом создайте столько каналов, сколько переменных используется в вашем технологическом процессе, созданном во втором задании лабораторной работы № 1. Исходный канал «УРОВЕНЬ 1» удалите из объекта.
- 7) Сохраните проект и загрузите его в редакторе представления данных. Создайте новую группу экранов и создайте экранные формы для управления процессом, разработанным во втором задании первой лабораторной работы.
- 8) В основной экранной форме нарисуйте статический рисунок вашего процесса, представленный во втором задании лабораторной работы №1.
- 9) Во второй экранной форме выведите графики основных технологических переменных.
- 10) Создайте кнопки перемещения между формами.



- 11) На основной экранной форме добавьте динамические элементы, показывающие изменение всех технологических переменных вашего процесса.

Для гармоничного восприятия данные элементы предлагается создавать из трёх элементов: 1) Статический прямоугольник; 2) динамический текст; 3) Статический текст.

В результате получить следующий элемент:



- 12) Размерность технологической переменной укажите в поле «Всплывающая подсказка:» в настройке элемента «Статический прямоугольник».

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните структура и состав системы ТРЕЙС МОУД 5.0.
2. Каковы основные понятия редактора базы каналов системы ТРЕЙС МОУД 5.0 ?
3. Каковы основные понятия редактора представления данных ТРЕЙС МОУД 5.0 ?
4. Перечислите графические элементы системы ТРЕЙС МОУД 5.0. Как можно их классифицировать ?
5. Порядок работы с редактором базы каналов системы ТРЕЙС МОУД 5.0.
6. Поясните порядок работы с редактором представления данных системы ТРЕЙС МОУД 5.0.
7. Поясните последовательность создания проекта в системе ТРЕЙС МОУД 5.0.

6. СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА

Отчет оформляется в ученической или в общей тетради в соответствии с общими требованиями к оформлению текстовых документов. Отчет должен содержать:

- 1) наименование и цель работы;
- 2) название и назначение каждого из редакторов ТРЕЙС МОУД;
- 3) последовательность создания проекта в редакторе базы каналов, описанная в уроке №1;
- 4) назначение и принцип автопостроения, описанное в уроке №2;
- 5) основные параметры канала, настраиваемые в данной работе;
- 6) FBD-программа реализующая PID-регулятор, разработанная в уроке №3;
- 7) название и назначение основных графических элементов используемых в уроке №4;
- 8) последовательность настройки архивирования данных, описанная в уроке №5.

ЛАБОРАТОРНАЯ РАБОТА № 3

"РАЗРАБОТКА АРМ-ОПЕРАТОРА ДЛЯ СУПЕРВИЗОРНОГО УПРАВЛЕНИЯ МИКРОПРОЦЕССОРНЫМ КОНТРОЛЛЕРОМ РЕМИКОНТ Р-130"

1. ВВЕДЕНИЕ

Как уже говорилось, основная задача любой SCADA-системы – это обеспечение оператора информацией о ходе протекания технологического процесса и передача управляющих сигналов от диспетчера нижнему, контроллерному уровню.

Эти функции определяются во-первых особенностями выпускаемых в настоящее время контроллеров, а во-вторых, исходя из обеспечения надежности функционирования АСА ТП в целом.

Надёжность описанной выше системы заключается в том, что при нарушении связи между верхним и нижним уровнями, контроллер продолжает работать с настройками, переданными ему ранее.

Особенность контроллеров заключается в том, что в реальном времени нельзя изменить структуру программы, выполняемой в контроллере, но легко можно изменить почти любые переменные, используемые для настройки алгоритмов работы.

Таким образом, у диспетчера появляется дополнительная функция, позволяющая настраивать программы контроллера. Задачи, выполняемые контроллером можно разделить на две группы: это регулирование технологических параметров и управление оборудованием.

Задачу настройки этих задач принято называть супервизорным управлением. При этом оператор может изменять задание системе регулирования, изменять параметры настройки регуляторов, менять уставки для срабатывания систем блокировки и т.д.

2. ЦЕЛЬ РАБОТЫ

В результате выполнения лабораторной работы студент должен:

- ознакомиться с возможностями супервизорного управления;
- разработать проект, позволяющий производить супервизорное управление микропроцессорным контроллером Ремиконт Р-130;
- закрепить навыки создания проекта с TRACE MODE 5.0;
- изучить последовательность обработки информации в канале.

3. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Последовательность создание проекта была рассмотрена в предыдущей работе. В данной работе предполагается особое значение уделить последовательности обработки информации в канале: как значения поступают из внешних источников, как обрабатываются, и как передаются на нижний уровень.

Для начала остановимся подробнее на значениях канала, которые являются основными переменными, а затем рассмотрим каким образом возможно производить обработку информации в канале.

3.1. Значения канала

Каждый канал имеет четыре основных значения: входное, аппаратное, реальное и выходное. Они обозначаются следующим образом: in - входное; A - аппаратное; R. - реальное; O - выходное.

Значения канала могут иметь один из следующих форматов:

- *число с плавающей точкой одинарной точности;*
- *16-битовое целое число.*

Первый формат используется для аналоговых переменных, второй - для дискретных.

Под точкой ввода-вывода понимается входное или выходное значение канала, связанное с внешним источником или приемником данных по стандартным протоколам Трейс Моуд, Windows или по протоколам драйверов УСО. Такое значение канала типа FLOAT соответствует одной точке ввода-вывода, а типа HEX - *n* точкам, где *n* - разрядность значения.

Входное значение каналов типа INPUT может формироваться одним из следующих способов:

- *данными от внешних источников (управляющие контроллеры, УСО, данные с удаленных узлов и пр.);*
- *данными, запрашиваемыми у системы (системные переменные, значения других каналов и пр.).*

У каналов типа OUTPUT их **входное значение** формируется одним из следующих способов:

- *процедурой «Управление» данного канала;*
- *процедурами «Управление» или «Трансляция» других каналов;*
- *метапрограммой на языке Техно-IL;*
- *каналом удаленного узла (например, по сети);*
- *оператором с помощью управляющих графических форм.*

Аппаратное значение каналов типа INPUT формируется масштабированием (логической обработкой для дискретных каналов) входных значений. У каналов типа OUTPUT **аппаратное значение** получается из реального процедурой трансляции.

Аппаратные значения каналов имеют такое название, поскольку в них удобно получать величины унифицированных сигналов, с которыми работает аппарат ввода-вывода (4...20 мА 0...10 В и т. д.)

Реальные значения предназначены для хранения значений контролируемых параметров или сигналов управления в реальных единицах (например, в кг/час, °С, % и т. д.).

Для входных каналов (тип INPUT) реальные значения формируются из аппаратных процедурами *трансляции* и *фильтрации*.

Если канал является выходным (тип OUTPUT), то его реальное значение получается из входного после *фильтрации*.

Выходное значение определено только для каналов типа OUTPUT Оно пересчитывается из аппаратного значения. В аналоговых каналах используется процедура **масштабирования**. Для каналов, обрабатывающих дискретные сигналы, выходное значение формируется из аппаратного **логической обработкой**.

3.2. Классификация каналов

Каналы могут принимать данные с внешних устройств или управлять ими, контролировать или управлять работой системы в т. п. Назначение канала определяется его типом, подтипом и дополнениями к подтипу. Адресация к данным в рамках его назначения осуществляется с помощью настроек. Каналы могут обрабатывать аналоговые или дискретные сигналы. Это устанавливается с помощью параметра «Вид представления».

Параметры канала настраиваются в диалоге «Каналы объекта». Области настройки перечисленных параметров показаны на рисунке 12 (стр. 25).

Тип канала определяет направление движения информации в канале.

Входные значения каналов типа INPUT связываются с источниками данных.

Каналы типа OUTPUT передают свои выходные значения приемникам данных.

В поле задания типа канала диалог «Каналы объекта» для идентификации типа используется один символ: I - тип INPUT; O - тип OUTPUT.

В большинстве случаев тип канала выставляется автоматически при задании подтипа и дополнения к подтипу. Однако иногда его надо устанавливать вручную.

Подтип канала указывает класс источников или приемников Данных, с которыми будет связываться канал.

Для каналов типа INPUT подтип характеризует получаемую ими информацию («АНАЛОГ» - значение АЦП, считанное с платы УСО, «СИСТЕМНЫЙ» - состояние системы, «СВЯЗЬ» - данные с удаленных узлов проекта и пр.). Каналы OUTPUT имеют тот же набор подтипов, что и каналы INPUT. Однако для них подтип определяет класс приемников, а не источников данных («АНАЛОГ» - значение ЦАП, «СИСТЕМНЫЙ» - состояние системы, «СВЯЗЬ» - значения управляемых каналов на удаленных узлах проекта и т. п.).

Всего существует шестнадцать подтипов каналов. Все они могут задаваться как для входных, так и для выходных каналов.

Все **подтипы** каналов имеют **дополнение к подтипу**. С помощью дополнений уточняется тип источника или приемника данных. Например, для канала подтипа «СВЯЗЬ» с помощью дополнения задается среда передачи: локальная сеть, последовательным интерфейс или коммутируемые линии.

Для адресации данных в рамках установленного типа источника или приемника данных используются настройки канала. Их число зависит от подтипа и дополнения к подтипу. Примером таких настроек могут быть базовый адрес платы УСО и номер канала ввода на этой плате.

Канал может работать с аналоговыми параметрами (значения температур, расходов, напряжений и т. д.) и с дискретными (сигналы от концевых выключателей, магнитных пускателей, пороговых датчиков и т. д.). Тип данных, с которыми работает канал, определяется его видом представления:

- число с плавающей точкой одинарной точности;
- 16-битовое целое число.

Первый из них предназначен для аналоговых переменных, второй - для дискретных. Таким образом, канал может обрабатывать; либо один аналоговый сигнал, либо до 16 дискретных.

По умолчанию каналы имеют вид представления - число с плавающей точкой, что индицируется буквой F (float). При нажатии ЛК буква F сменится на букву H (hex), что соответствует виду представления для работы с дискретными сигналами.

Списки атрибутов у каналов с разным видом представления различаются.

3.3. Обработка данных в канале

Набор процедур в канале зависит от формата данных. Каналы, работающие с аналоговыми переменными, используют следующие процедуры:

- «**Масштабирование**»,
- «**Трансляция**»,
- «**Фильтрация**»
- «**Управление**».

В каналах, обрабатывающих дискретные параметры, применяются логическая обработка, трансляция и управление.

Порядок следования и содержание процедур может меняться в зависимости от типа канала (входной или выходной). На рисунке 14 представлены четыре возможные схемы каналов.

В каналах типа INPUT их процедуры обеспечивают первичную обработку данных (исправление ошибок датчиков, масштабирование коррекция температуры холодных спаев термопар и т. д.). **В каналах типа OUTPUT процедуры преобразуют** величину управляющего воздействия из реального формата (проценты, амперы и пр.) к виду, воспринимаемому внешними устройствами.

Процедура «**Масштабирование**» включает в себя две операции: «**Умножение**» и «**Смещение**». Последовательность этих операций меняется в зависимости от типа канала:

- У каналов типа INPUT входное значение умножается на заданный множитель и к полученному результату добавляется величина смещения. Результат присваивается аппаратному значению канала.
- У каналов типа OUTPUT к аппаратному значению добавляется величина смещения, затем эта сумма умножается на заданный множитель, а результат присваивается выходному значению канала.

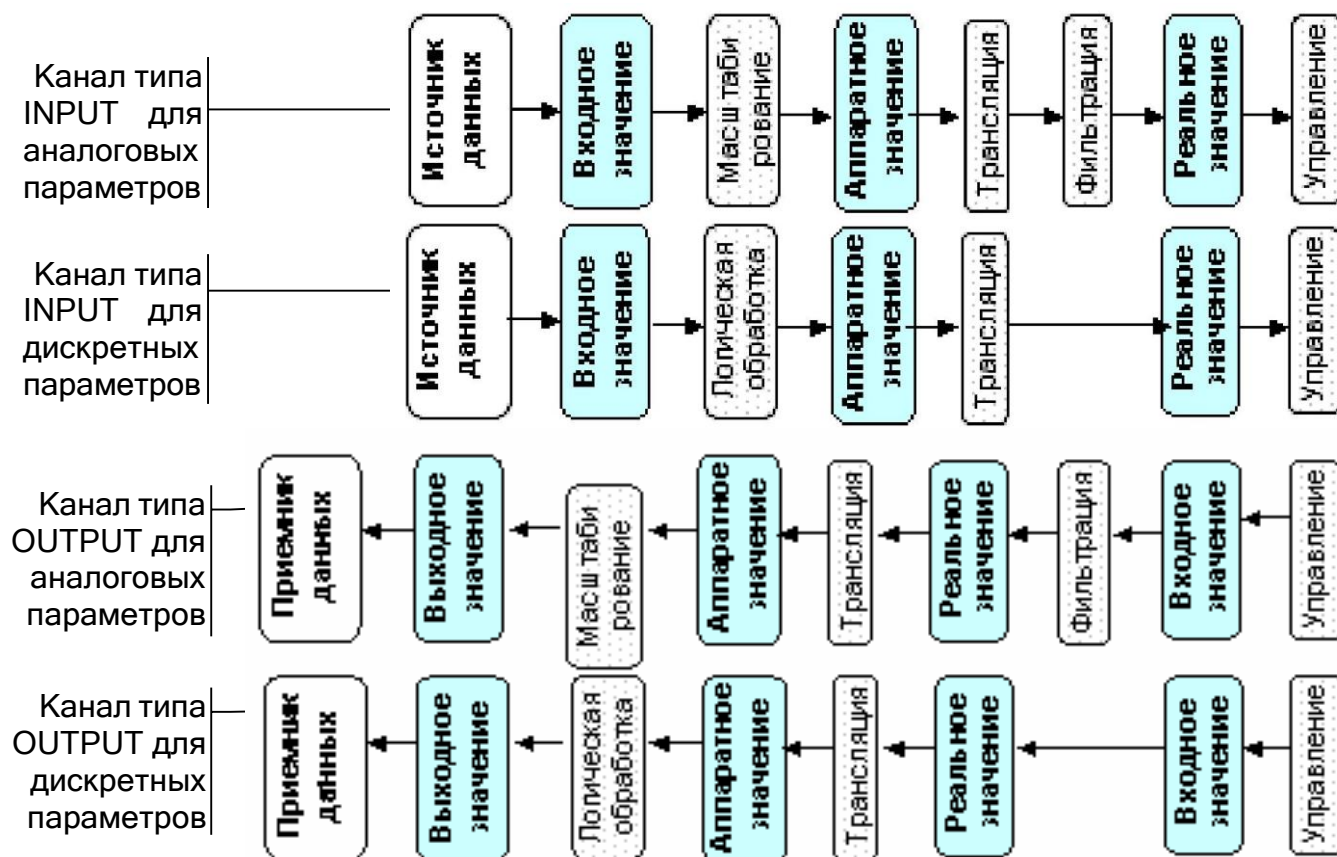


Рисунок 14 – Структура каналов для аналоговых и дискретных параметров

Процедура «**Логическая обработка**» в каналах типа INPUT по входному значению формирует аппаратное, а если тип OUTPUT, то по аппаратному значению - выходное. Канал работает с упакованными дискретными сигналами (до 16 сигналов). Для обработки этих сигналов можно задать три маски. Их использование рассмотрим в главе «обработка данных в канале»

Процедура «**Трансляция**» у входных каналов преобразует **аппаратное значение в реальное**, а у выходных - наоборот. Для этого вызывается FBD-программа, выбираемая при настройке процедуры. При настройке процедуры входные и выходные аргументы выбранной программы связываются с атрибутами текущего канала, а также любых других каналов из текущей базы. Поэтому процедура трансляции одного канала может также использоваться для формирования значений других каналов.

Процедура «**Фильтрация**» используется в каналах работающих с аналоговыми переменными. Для входных и выходных каналов выполняется по-разному.

У каналов типа INPUT **фильтрация** осуществляется после процедуры трансляции до формирования реального значения и включает в себя операции:

- *подавление случайных всплесков в тракте измерения;*
- *подавление малых колебаний значения канала;*
- *экспоненциальное сглаживание;*
- *контроль шкалы* - отслеживание выхода реального значения канала за установленные границы шкалы.

У каналов типа OUTPUT процедура **фильтрация** формирует реальное значение по входному значению. При этом выполняются операции:

- *ограничение скорости изменения реального значения;*
- *подавление малых колебаний значения канала;*
- *экспоненциальное сглаживание;*
- *контроль шкалы* - усечение величины управляющего воздействия до границ шкалы канала.

Процедура «**Управление**» реализует функции управления. С её помощью вызывается FBD-программа, в которой можно запрограммировать требуемые алгоритмы управления. В качестве аргументов программе могут передаваться значения и атрибуты любых каналов из текущей базы. Эти аргументы могут быть как входными, так и формируемыми.

Формально процедура «Управление» связана с каналом только циклом пересчета. Она может вообще никак не участвовать в формировании его значений, а управлять другими каналами. Такая ситуация часто наблюдается при использовании процедуры управления на каналах типа INPUT.

3.4. Классификация атрибутов канала

Любой канал обязательно имеет следующий набор атрибутов:

- | | |
|---------------------------------------|---|
| - имя; | - состояние канала; |
| - тип канала; | - начальное значение; |
| - подтип канала; | - кодировку; |
| - дополнение к подтипу канала; | - размерность; |
| - вид представления; | - СПАД, регистратор, отчет тревог, атрибуты; |
| - период и фаза; | - время изменения. |
| - подключение; | |

Кроме того, для каждого канала можно установить ряд флагов, определяющих режимы пересчета и архивирования данных:

- «**В сеть**»; - «**Доступ**»; - «**Отработать**»; - «**СПАД**»;
- «**Тревоги**»; - «**Регистратор**»; - «**Атрибуты**».

Набор атрибутов канала зависит от его вида представления.

Для аналоговых каналов:

- **четыре аварийные границы;**
- **гистерезис;**
- **шкалу - две границы;**
- **коэффициент сглаживания;**
- **множитель;**
- **дрейф нуля;**
- **величину апертуры;**
- **порог фильтрации пиков.**

Для дискретных каналов:

- **Маска контроля сочетаемости;**
- **Маска инверсии;**
- **Маска предустановки.**

Кроме того, эти каналы имеют флаги включения операций

Для аналоговых каналов:

- «Сглаживание»,
- «Апертура»,
- «Фильтрация пиков»,
- «Умножение»,
- «Смещение (дрейф нуля)»,
- «Контроль границ шкалы».

Для дискретных каналов:

- «Контроль сочетаемости»,
- «Инверсия»,
- «Предустановка».

Кроме перечисленных выше, канал обладает еще рядом атрибутов. Они вычисляются системой или формируются по заданным алгоритмам. К таким атрибутам относятся:

- **Интервал;**
- **Тенденция;**
- **Достоверность;**
- **События.**

Основным идентификатором канала является его имя. Оно используется для ссылок на значения канала. К имени канала также привязываются динамические формы в редакторе представления данных.

При создании канала его имя формируется автоматически. При автопостроении оно отражает источник данных. Например, имя канала, связанного с пятым каналом платы аналогового ввода, расположенной в первом посадочном месте контроллера, будет «**AI_pp01-0005**». Если канал создан автопостроением запроса Данных с удаленных узлов, то его имя воспроизводит имя канала-источника.

При создании канала вручную ему присваивается следующее имя:

nen-xx00-*<число>*, де *<число>* - номер канала по порядку в объекте «БАЗА».

Этот параметр является числовым идентификатором канала и не изменяется при любых операциях с данным каналом (переименование, переопределение типа, подтипа и т. п.). Номер канала отображается в правом нижнем окне диалога «Каналы объектам редактора базы каналов».

Чтобы изменить имя канала, надо открыть диалог «Каналы объекта» того объекта, где он находится. Далее следует выделить этот канал в списке и внести требуемые изменения.

В имени канала можно использовать любые символы. Однако не рекомендуется использовать символы математических операций, знаки препинания и пробелы. Это важно, если организуются внешние ссылки на канал из других приложений и метапрограмм.

Имя канала не может быть менее шести символов. В противном случае оно дополняется до этого размера символами подчеркивания. Максимальный размер имени составляет тринадцать символов.

Период работы канала определяет частоту обновления данных и пересчета его значений. Значение периода может устанавливаться либо в единицах времени (секунды, минуты, часы), либо в периодах пересчета базы каналов (циклы системы).

Время, отводимое на цикл системы, называется **периодом пересчета**. Этот параметр определяет частоту выполнения основным системных задач (обмен данными с контроллерами, пересчет базы каналов, перерисовка экрана, сетевой обмен и пр.). Если задать период пересчета недостаточным для выполнения всех задач, то наименее приоритетные из них (например, обновление экрана монитора) будут тормозиться. Однако большое значение периода пересчета уменьшает скорость реакции системы. Таким образом, период пересчета надо задавать, исходя из выполнения всех ее задач и сохраняя при этом требуемую реактивность системы.

Значение и размерность периода вводятся в разделе «Период» бланка «Основные» диалога «Реквизиты» (рисунок 13, стр. 27).

Размерность периода канала выбирается из следующего списка:

- *цикл* - период в циклах;
- *сек* - период в секундах;
- *мин* - период в минутах;
- *час* - период в часах;
- *флаг* - период в циклах с отключением пересчета по системному флагу;
- *F1* - период в циклах с отставанием на один цикл;
- *F2* - период в циклах с отставанием на два цикла;
- *F3* - период в циклах с отставанием на три цикла;
- *F4* - период в циклах с отставанием на четыре цикла;
- *на старте* - канал отрабатывается один раз при запуске;
- *быстрый* - канал работает вне цикла системы.

Значение периода может изменяться в диапазоне от 1 до 100. По умолчанию оно задается в циклах и равно единице.

Период каналов, получающих данные от аппаратуры ввода-вывода, задает частоту опроса технологических параметров. Период выходных каналов задает частоту формирования управляющих сигналов.

В принципе можно задать одинаковую частоту опроса для всех контролируемых технологических параметров, ориентируясь на наиболее динамичные, или просто принять значения периодов всех каналов по умолчанию, что соответствует максимальной частоте. Однако в этом случае при создании крупных систем с большим числом контролируемых параметров может не хватить ресурсов компьютера для задач обработки и представления информации.

Атрибуты «СПАД», «Регистратор», «Отчет тревог», «Атрибуты» возвращают единицу при установке соответствующего флага в настройках канала, в противном случае их значение равно нулю.

Каждый канал может находиться в одном из двух **состояний** - «Включен» и «Выключен». Если канал включен (ON), то его значения пересчитываются с частотой, определяемой периодом канала. Когда канал выключен (OFF), значения в нем не пересчитываются и остаются неизменными.

Состоянием канала можно управлять в реальном времени. Для этого существуют следующие способы: другим каналом типа OUTPUT с подтипом «Канал» и дополнением к подтипу «Установить»; с помощью форм управления; из программ; по сети; по последовательному интерфейсу. В качестве управляемого атрибута указать состояние канала (C).

Для каждого канала можно установить **начальное состояние**. Это состояние, в которое канал будет переведен при запуске системы. По умолчанию начальное состояние канала - включен.

Начальное состояние канала задается в бланке «Основные» диалога «Реквизиты» нажатием ЛК на соответствующей кнопке раздела «При старте» (рисунок 13, стр. 27).

При настройке канала можно задать его начальное значение. Это значение при запуске монитора реального времени присваивается входному значению канала.

Чтобы начальное значение канала после присвоения было пересчитано процедурой трансляции, а в случае выходных каналов были произведены действия, определенные подтипом и дополнением к подтипу, в этом же меню следует установить флаг «Отработать».

Чтобы значение канала **записывалось** в любой из **архивов**, необходимо установить для него соответствующий флаг:

- флаг «СПАД»;
- флаг «Тревоги»;
- флаг «Регистратор».

Чтобы в архивах сохранялись изменения всех остальных атрибутов канала, необходимо установить флаг «Атрибутов».

В любом технологическом процессе есть параметры, по значению которых определяется состояние отдельных технологических узлов, стадий, участков, аппаратов. Выход значений этих параметров за заданные границы указывает на предаварийное или аварийное состояние процесса. При возникновении такой ситуации система управления должна выполнить действия, направленные на предотвращение возникновения или развития аварии (включение сигнализации, выдача рекомендации оператору или блокировка).

Чтобы обеспечить обработку аварийных ситуаций, каналы, работающие с аналоговыми переменными (вид представления F), имеют четыре **аварийные границы** и две **границы шкалы**.

Границы шкалы указывают возможный диапазон изменения контролируемого параметра. Например, если датчик позволяет измерять температуру в диапазоне от 0 до 400 °С, то его показания, лежащие вне данного диапазона, являются заведомо недостоверными. Если задать для канала границы шкалы, то при выходе за них его реального значения может автоматически формироваться признак недостоверности данных. Эта информация может быть доведена до оператора и зафиксирована в архивах.

Четыре аварийных границы подразделяются на две верхние и две нижние. Внутренние границы - предаварийные, внешние - аварийные. Значения границ задаются в разделе «Границы» бланка «Границы и обработка» диалога «Реквизиты» (рисунок 15).

Границы		Обработка	
Верхний предел	400	Множитель	2.5 ☒
ВГ_1	350	Дрейф нуля	16.8 ☒
Гистерезис	5	Пик	25 ☐
ВГ_0	300	Апертура	1 ☐
НГ_0	100	Сглаживание	0 ☐
НГ_1	50	Контр. шкалы	☒
Нижний предел	0		

Рисунок 15 – Окно для ввода границ и значений процедур обработки

В областях «ВГ1», «НГ1» вводятся значения верхней и нижней **внешних границ**;

В областях «ВГО», «НГО» - значения верхней и нижней **внутренних границ**.

Верхняя и нижняя границы шкалы задаются в областях «Верхний предел» и «Нижний предел» соответственно.

Значения границ шкалы и аварийных границ доступны для изменения в реальном времени. Это можно реализовать с помощью форм отображения, программ, а также по сети или последовательному интерфейсу из удаленного узла.

Аварийные границы и шкала разбивают диапазон изменения значения канала на семь интервалов. На рисунке 16 приведено графическое представление границ и интервалов значений канала.

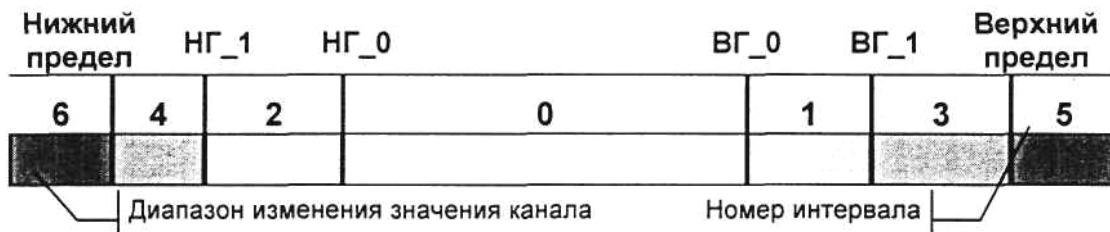


Рисунок 16 - Графическое представление границ и интервалов

На каждом цикле пересчета данных МРВ определяет номер интервала, в котором находится текущее значение канала, и формирует значение специальной переменной, являющейся атрибутом канала. Эта переменная называется **интервалом** и обозначается **P**.

В бланке «Границы и обработка» можно задать еще один параметр. Он влияет на отслеживание перехода значения канала через аварийные границы и называется **гистерезисом**. Введение гистерезиса позволяет убрать ненужный поток сообщений в отчет тревог при небольших колебаниях контролируемого параметра вблизи значения одной из границ. Величина этого параметра задается в области гистерезиса.

Для анализа технологических ситуаций и принятия решения часто необходимо знать не только значения параметров, но и то, как они изменяются в настоящий момент. Каждый канал имеет специальный атрибут, характеризующий тенденцию его изменения. **Тенденция (D)** оценивает изменение реального значения канала на текущем цикле по отношению к предыдущему.

В зависимости от вида представления канала формирование величины этого атрибута осуществляется разными способами.

Для каналов с видом представления F он принимает следующие значения:

- значение не изменилось;
- значение уменьшилось;
- значение увеличилось.

У каналов с видом представления H атрибут «Тенденция» представлен целым 16-битным числом. Каждый бит этого числа является индикатором изменения соответствующего бита реального значения. Если на текущем такте пересчета значение бита изменилось, то соответствующий бит тенденции устанавливается равным единице, в противном случае - равным нулю.

Каждому каналу можно присвоить **размерность**, в которой представлено его реальное значение. Установка размерности осуществляется в диалоге «Ка-

налы объекта». Для этого используется специальное поле данного диалога (см. рисунок 12, стр. 25).

Размерность выбирается из списка, который хранится в файле **razmer.ini** в директории INI. Если требуемая размерность в списке отсутствует, ее можно добавить, отредактировав указанный файл. Текст размерности не может содержать более 8 символов, а число строк в списке не должно превышать 255.

Для каждого канала можно задать **индивидуальный код**, который определяет его привязку к технологической схеме. Например, в кодировке могут указываться цех, участок, аппарат, устройство, тип сигнала.

С помощью кодировки можно построить иерархию объектов, которая позволяет легко ориентироваться в сложных проектах. Это облегчает для проектировщика выполнение декомпозиции задачи.

Для каждого канала можно задать комментарий: текстовую строку длиной до 38 символов. Текст комментария доступен в реальном времени, если канал не имеет кодировки. В этом случае в поля вывода кодировки подставляются первые 21 символ комментария.

Эта информация может выводиться на экраны мониторов, вставляться в отчеты и записываться в отчет тревог.

3.5. Первичная и выходная обработка

При измерении технологических параметров информация от датчиков поступает в аппаратуру ввода/вывода в виде унифицированных сигналов (0-10В, 4-20 мА и т. д.), сигналов от термопар, термометров сопротивления или в виде милливольтовых сигналов. Это означает, что реальной физической величине ставятся в соответствие такие параметры, как напряжение, сила тока, индуктивность или частота импульсов. С помощью устройств связи с объектом эти сигналы преобразуются в двоичные коды длиной от 8 до 16 разрядов. Чтобы провести анализ получаемой информации, необходимо преобразовать коды АЦП в масштаб реальных физических величин (мм, кг/час, ата, °С и т. д.). Кроме того, датчики могут иметь статические ошибки, нелинейные характеристики или зашумленный выходной сигнал.

Задачи масштабирования, линеаризации, коррекции ошибок датчиков и фильтрации составляют содержание первичной обработки данных. Другими словами, первичная обработка - это процедура получения корректных значений результатов измерения из двоичных кодов устройств связи с объектом.

При управлении ситуация выглядит аналогично. Однако здесь реализуется обратное движение информации. Рассчитанная величина управляющего воздействия переводится в цифровой код, который с помощью ЦАП преобразуется в стандартный унифицированный электрический сигнал. Этот сигнал подается на исполнительный механизм.

Чтобы реализовать требуемое управление и при этом защитить оборудование от возможных перегрузок при резких изменениях величины управляющих воздействий, необходимо реализовать выходную обработку. Такая обработка включает в себя масштабирование коррекции статической погрешности испол-

нительного механизма, ограничение скорости изменения и самой величины управляющего воздействия.

Первичная обработка

При первичной обработке процедура **трансляции** преобразует входные значения каналов, величины которых получены с аппаратуры ввода/вывода, в аппаратные и затем - в реальные. При этом реализуется следующий набор операций:

- масштабирование со сдвигом;
- фильтрация (три встроенных метода);
- установка битов (для дискретных сигналов);
- анализ на недопустимость сочетания (для дискретных сигналов);
- инверсия (для дискретных сигналов);
- вызов произвольной программы, написанной на языке функциональных блоков.

Выходная обработка

Выходная обработка преобразует входные значения каналов типа OUTPUT в реальные, затем - в аппаратные и после этого - в выходные значения, которые затем передаются на аппаратуру ввода/вывода. Эта обработка включает в себя следующие операции:

- ограничение величины управляющего воздействия;
- ограничение скорости изменения управляющего воздействия;
- масштабирование со сдвигом;
- анализ на недопустимость сочетания (для дискретных сигналов);
- инверсия (для дискретных сигналов);
- вызов произвольной программы, написанной на языке функциональных блоков.
-

Об этапах первичной и выходной обработки мы говорили в разделе 3.3 данной работы, поэтому перейдём к более подробному рассмотрению встроенных методов обработки.

Прежде чем перейти к описанию самих методов обработки, надо отметить, что настройка их осуществляется индивидуально для каждого канала. Это реализуется в диалоге **Реквизиты**.

Для каналов с видом представления F настройка обработки осуществляется в бланке **Границы и обработка** данного диалога (рисунок 17).

Для каналов с видом представления H - в бланке **Маски и эмуляция** (рисунок 18).

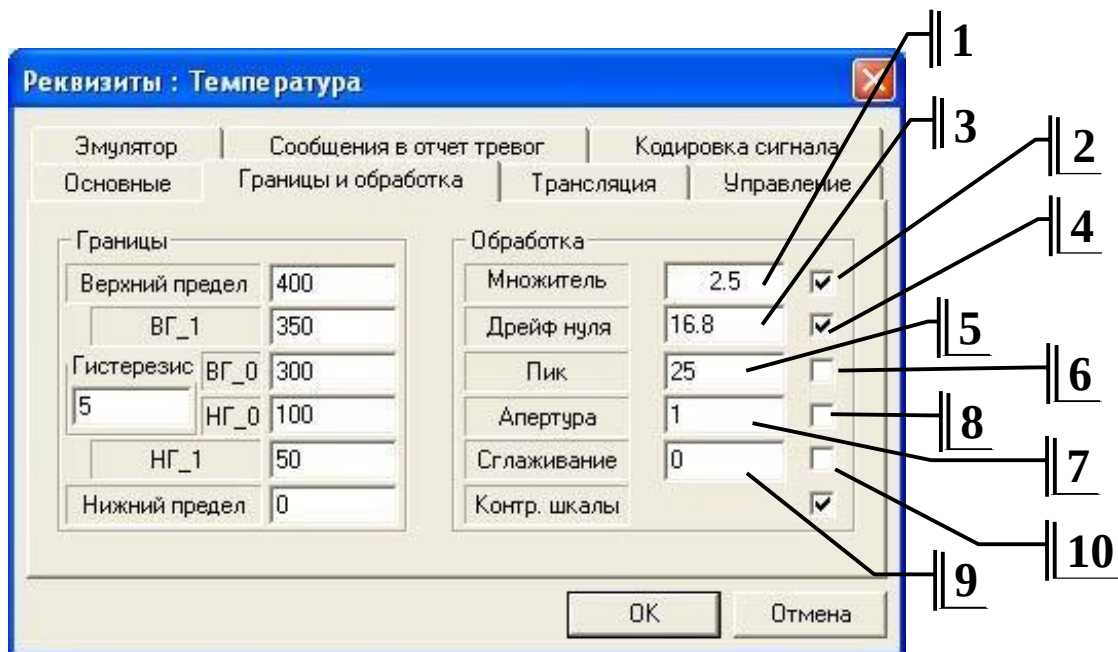


Рисунок 17 – Окно настройки обработки для аналоговых каналов

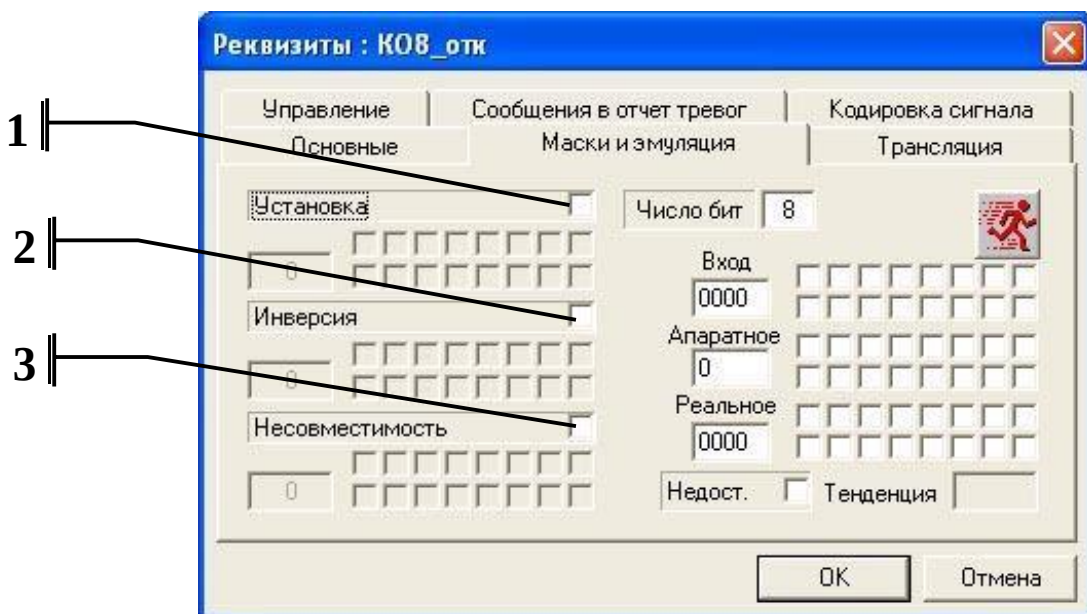


Рисунок 18 – Окно настройки обработки для дискретных каналов

3.5.1. Масштабирование

Этот метод обработки реализуется для каналов с видом представления — число с плавающей точкой (F).

Для реализации в канале операции масштабирования необходимо в поле **Множитель** (1 – рисунок 17) бланка **Границы и обработка** диалога **Реквизиты** задать значение коэффициента масштабирования. Кроме того, здесь же надо установить флаг включения данного метода (2 – рисунок 17).

При пересчете значений канала, для которого был задан этот метод обработки, осуществляется умножение входной величины на заданный коэффициент масштабирования.

3.5.2. Дрейф нуля

Этот метод обработки реализуется для каналов с видом представления (F). Он осуществляет **смещение** выходной величины по отношению к входной на заданное значение.

Для реализации этого метода обработки надо в поле **Дрейф нуля** (3 – рисунок 17) бланка **Границы и обработка** диалога **Реквизиты** задать значение величины смещения и установить флаг включения данного метода (4–рисунок 17).

Процедура **Масштабирование** и **Дрейф нуля** (смещение, сдвиг) действуют совместно по следующей схеме:

- Для **входных каналов** этот метод обработки реализуется до формирования аппаратного значения.

Получаемая от **входного значения**, т.е. от источника данных величина:

- а) сначала **масштабируется**,
- б) затем **сдвигается** (см. ниже)
- в) после этого записывается в **аппаратное значение**.

- Для **выходных каналов** данная операция выполняется после формирования аппаратного значения.

Величина, записанная в **аппаратное значение**:

- а) сначала **сдвигается**,
- б) затем **масштабируется**,
- в) после передается **выходному значению** и приемнику данных.

3.5.3. Подавление пиков

Назначение данного атрибута различно в зависимости от типа каналаб

Для каналов типа INPUT

Этот метод фильтрации следует применять для подавления сильных импульсных помех в измерительных трактах.

Для каналов типа OUTPUT

Исполнительные механизмы часто могут реализовать величину управляющего воздействия большую, чем это допускается по технологическим соображениям. Это может привести к возникновению аварийной ситуации.

Для предотвращения таких ситуаций реализован специальный метод выходной обработки. Каналам, формирующим аналоговые величины управляющих воздействий, можно установить ограничение на реальные значения.

Для реализации функции подавления пиков надо установить для канала соответствующий флаг (6–рисунок 17) и задать величину максимального изменения значения канала за один такт пересчета (5–рисунок

17) в бланке **Границы и обработка** диалога **Реквизиты**, как показано на следующем рисунке.

3.5.4. Апертура

Данный метод обработки реализует фильтрацию малых изменений входного сигнала. Он обеспечивает зону нечувствительности. Если изменение выходной величины по отношению к предыдущему значению меньше зоны нечувствительности, то выход не меняется.

Для отслеживания апертуры надо установить для канала соответствующий флаг и задать отличное от 0 значение зоны нечувствительности в бланке **Границы и обработка** диалога **Реквизиты** (7 и 8–рисунок 17).

Этот метод надо использовать для архивируемых каналов и для каналов, значения которых передаются удаленным узлам при их изменении. Введение этого метода позволяет существенно сократить интенсивность информационных потоков, а также увеличить глубину сохранения и скорость доступа к данным в архивах.

Этот метод обработки работает только в каналах с видом представления F. При этом тип канала может быть как INPUT, так и OUTPUT.

3.5.5. Экспоненциальное сглаживание

Этот метод вводит апериодичность изменения реального значения канала по отношению к аппаратному. Используя его, можно задать инерционность канала, то есть скорость изменения (реакции) реального значения канала при изменении аппаратного.

Данный метод фильтрации позволяет уменьшить величину случайных колебаний измеряемых значений параметров. Такие колебания могут быть обусловлены наличием шумов в измерительном тракте.

На рисунке 19 показано изменение значений канала типа INPUT, для которого включено экспоненциальное сглаживание. Здесь на вход канала подаются ступенчатое изменение (а) и сильно зашумленный сигнал от датчика (б). Толстой линией на обоих графиках показано истинное значение параметра. Тонкая линия демонстрирует измеренную величину (для аппаратного значения) и отфильтрованное значение (для реального значения).

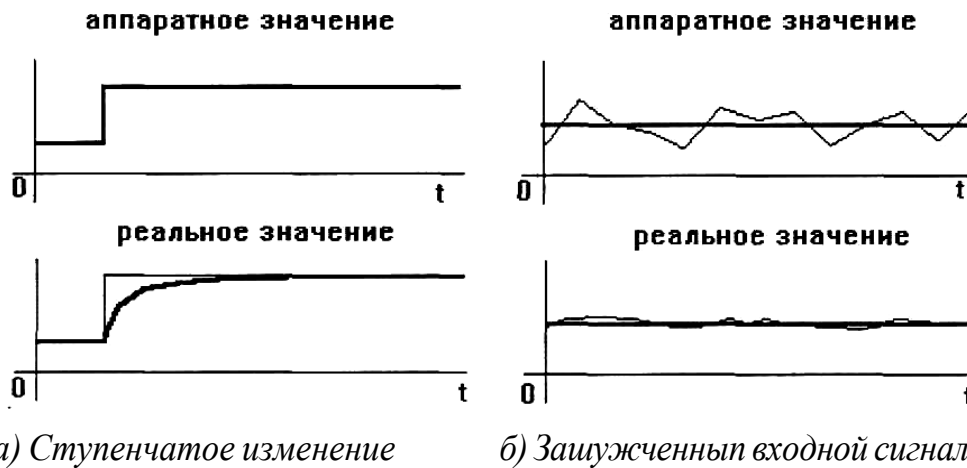


Рисунок 19 - Изменение значений канала типа INPUT, для которого включено экспоненциальное сглаживание

$$V = V_o * (1 - K) + V_p * K ,$$

где V - выходное значение канала;

V_p - выходное рассчитанное значение;

V_o - выходное значение канала на предыдущем цикле;

K - коэффициент фильтрации.

Чтобы в канале осуществлялся этот метод фильтрации, надо установить для него соответствующий флаг и задать отличное от 0 значение коэффициента сглаживания. Эти параметры задаются в бланке **Границы и обработка** диалога **Реквизиты (9 и 10 –рисунок 17)**.

3.5.6. Контроль шкалы

Этот метод обработки не требует задания каких-либо дополнительных атрибутов для канала, кроме соответствующего флага (**11 –рисунок 17**). Если флаг контроля шкалы установлен, то при выходе реального значения за границы шкалы каналу устанавливается признак недостоверности.

Используется только для аналоговых каналов.

Он имеет разное назначение в зависимости от типа канала.

- При его наличии у каналов типа INPUT устанавливается **признак недостоверности**, если реальное значение выходит за границы шкалы.
- Если канал OUTPUT, то флаг контроля шкалы задает **ограничение изменения его реального значения** рамками границ шкалы. При попытке присвоить каналу значение, выходящее за шкалу, оно будет обрезано до значения соответствующей границы.

3.5.7. Анализ на недопустимость сочетания

При управлении и контроле состояния устройств по значению дискретных сигналов можно выделить заведомо недопустимые их сочетания. Возникновение такого сочетания дискретных сигналов можно расценивать как некорректность в схемах их контроля.

Для индикации таких ситуаций каждому каналу, имеющему вид представления Н, можно задать маску, выделяющую биты, одно временная установка в 1 которых является признаком некорректной работы. В этом случае каналу будет установлен признак недоверности.

Для включения этой обработки используются соответствующие поля в бланке **Маски и эмуляция** диалога **Реквизиты (1)** –рисунок 18).

3.5.8. Инверсия

При обработке дискретных сигналов часто требуется использовать не полученное с датчика, а инвертированное значение. Операцию **инверсии** можно реализовать на стадии первичной обработки. Для этого каналу необходимо установить флаг инверсии и указать, значения каких битов его аппаратного значения надо инвертировать при выполнении процедуры трансляции.

Для включения этой обработки и выделения инвертируемых битов используются соответствующие поля в бланке **Маски и эмуляция** диалога **Реквизиты (2)** –рисунок 18). Расположение полей для установки флага инверсии значений дискретных сигналов и задания маски, выделяющей инвертируемые биты, показано на рисунке.

3.5.9. Предустановка

Последний метод первичной или выходной обработки предназначен для жесткой фиксации значений отдельных битов канала. Эта операция называется **Предустановкой**. Для ее реализации каналу следует установить соответствующий флаг и задать маску, выделяющую биты, значение которых будет всегда равно 1.

Для включения данной обработки и выделения битов, устанавливаемых в 1, используются соответствующие поля в диалоге **Реквизиты** бланка **Маски и эмуляция (3)** –рисунок 18). Расположение полей для установки флага предустановки значений дискретных сигналов и задания маски, выделяющей устанавливаемые биты, показано на рисунке.

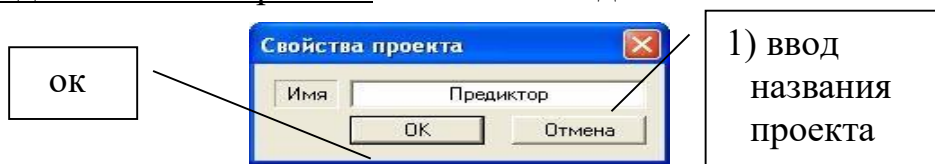
4. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Так как данная работа предусматривает подключение к реальному контроллеру, то помимо создания проекта в SCADA-системе необходимо произвести настройку микропроцессорного контроллера Ремиконт Р-130 и Шлюза, связывающего данный контроллер с персональным компьютером

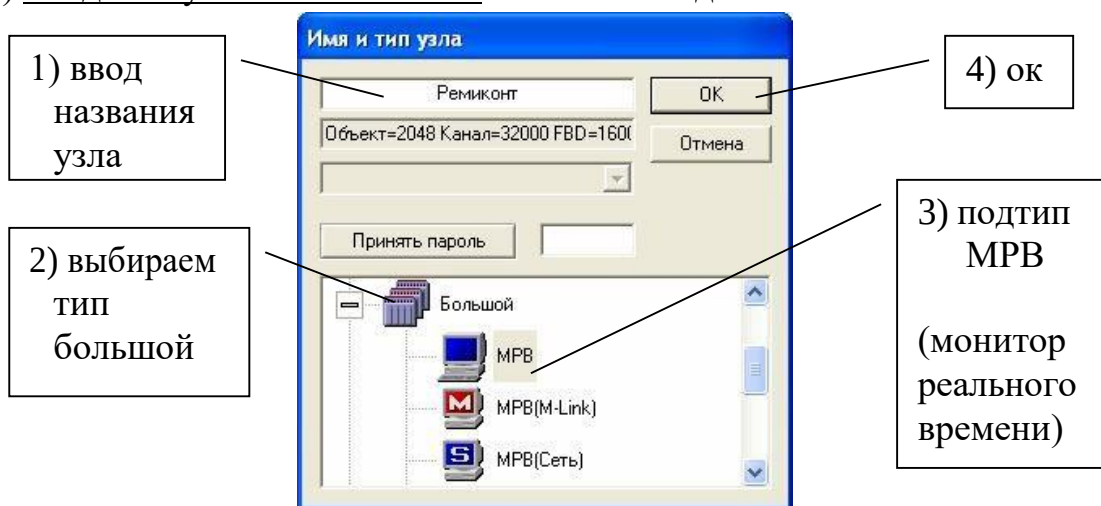
4.1 Настройки SCADA-системы TRACE MODE

4.1.1 Настройка в редакторе базы каналов

А) Создание нового проекта: `Файл` – `создать`



Б) Создание узла базы каналов: `Узел` – `создать`

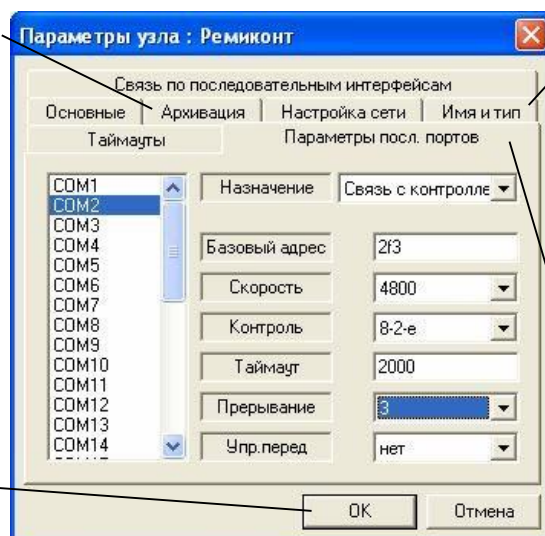


В) настройка параметров узла:

правый клик ПК на значке узла в окне узлов проекта



2) Вкладка `Архивация`:
вводим название будущего СПАД архива и его предположительный объем
(например:
Название - LAB3;
Объём - 5.0)



1) Вкладка `имя и тип`:
вводим или редактируем имя базы каналов
(Например: Ремиконт)

3) Вкладка `параметры посл. портов`:
Для связи со шлюзом используется порт COM2

Настройки порта COM2:

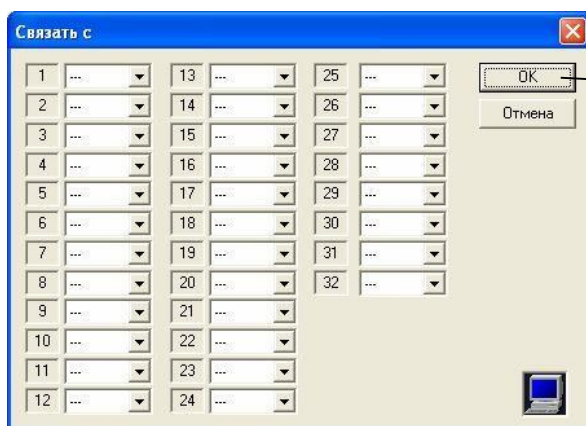
- **назначение** – связь с контроллером
- **базовый адрес** – любой отличный от 0 (Например 2f3)
- **скорость** – 4800
- **контроль (четности)** – 8-2-е
- **таймаут** – 2000 мс
- **прерывание** – 3
- **Упр. перед.** – нет

Г) Переходим в окно объектов выбранного узла:

Одним из трёх способов:

- 1) `окно` – `объекты`; 2) Alt – 2; 3) клик на значке

Если осуществляется первый вход в это окно, откроется промежуточное окно для указания связи с объектом



Нажмите `OK`, так как связь нам не требуется

Д) Разворачиваем объект база Р-130:

Одним из двух способов:

- 1) `объект` – `свернуть=развернуть`

- 2) один клик на значке



4.1.2 Создание и настройка каналов

Для выполнения лабораторной работы необходимо создать 18 каналов в объекте база каналов. Три канала будут принадлежать объекту **аналоговый ввод**, а остальные 15 объекту **аналоговый вывод**.

Откройте окно `Каналы объекта: \_БАЗА` для чего дважды кликните левой кнопкой мыши на значке



Создание каналов осуществляется в несколько этапов

- 1) меню открывшегося окна **‘Каналы объекта: _БАЗА_’**
‘канал’ – ‘создать’
- 2) ввод имени канала, для чего кликнуть мышкой в окошко **‘канал: имя’**, где вместо имени по умолчанию **‘new-xx00-0014’** ввести имя соответствующего канала, представленное в таблице 1.
- 3) Введите общие атрибуты каналов для связи с контроллером Ремиконт Р-130 (смотри на рисунке выше)

Общие атрибуты каналов для связи с контроллером Ремиконт Р 130:

подтип – контр_1

дополнение к подтипу – Ремиконт 130

№порта – 01

№контроллера – 01

Тип
Переменная
№алблока
№вх/вых

Смотри в таблице 1

- 4) Для каналов задание, возмущение и выход объекта необходимо установить флажок архивации в СПАД архиве, период обработки = 1 цикл. Откройте диалоговом окне **‘реквизиты канала’** (двойным кликом левой кнопки мыши на имени канала)

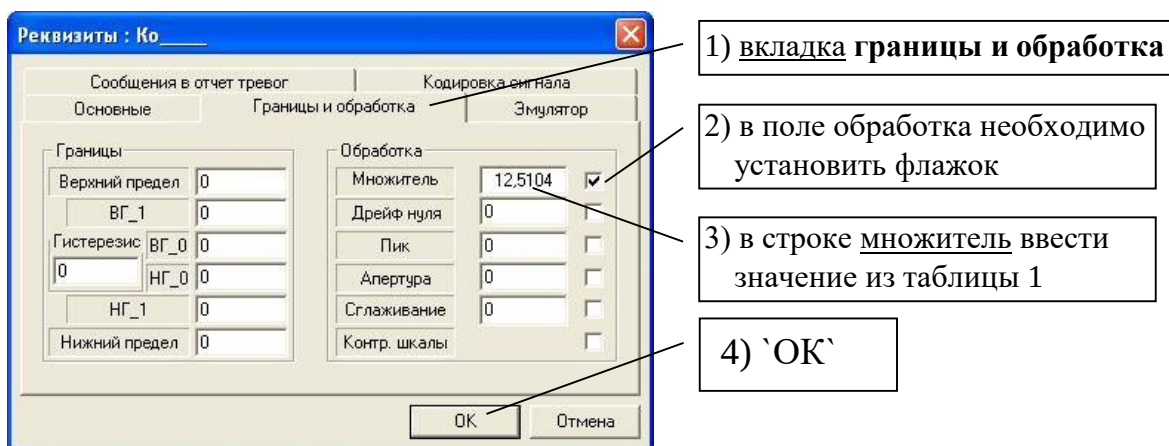
1) вкладка основные

3) период обработки = 1 цикл

2) флажок архивации в СПАД архиве

Для остальных каналов установить **период** обработки равный **1 с**, для чего вместо значения **‘цикл’** необходимо поставить **‘сек’**.

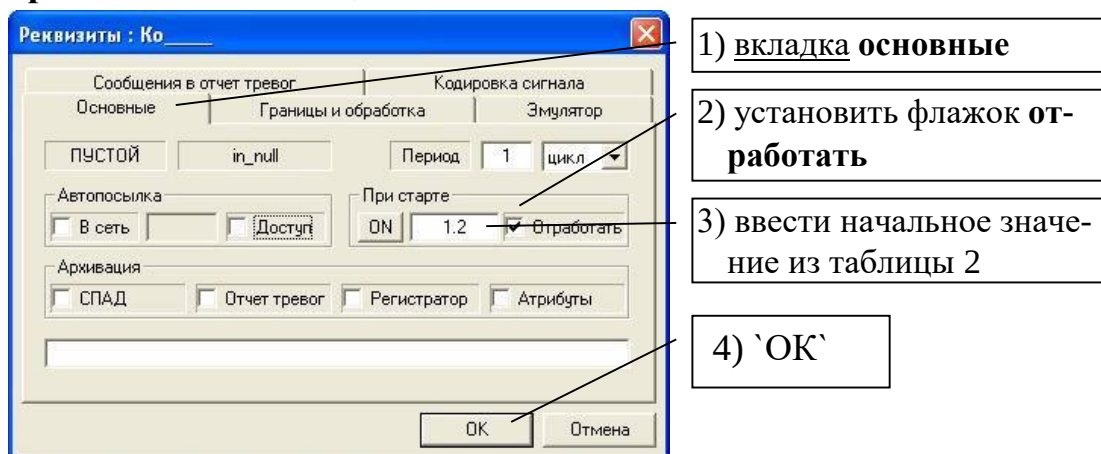
- 5) Для части каналов необходимо ввести поправочный коэффициент (множитель) для соответствия с видом представления сигналов в контроллере Ремиконт Р 130. Множитель устанавливается в диалоговом окне **‘реквизиты канала’** (которое открывается двойным кликом левой кнопки мыши на имени канала)



Список необходимых каналов, их назначение, атрибуты и множитель указаны в таблице 1.

- б) Для того, что бы при старте проекта значения объекта установились на заданном значении необходимо их задать

Начальные значения каналов устанавливаются в диалоговом окне **'реквизиты канала'**,



Необходимо установить флажок отработать и ввести начальное значение. Для каналов Км1 и Км2 (Масштабные коэффициенты) установить значения = 0, тем самым при старте будет отключена модель объекта и получим САР с обычным регулятором.

Значения параметров настройки каналов при старте проекта даны в таблице 2.

Таблица 1 - Настройки атрибутов каналов (для схемы №1)

Имя канала	Назначение	Тип Input/Output	Пере - менная	№Алго- блока	№Вх /Вых	Множитель
Задание	Запрос сигнала задающего воздействия	Input	Вых	6	1	нет
Возмущение	Запрос сигнала возмущающего воздействия	Input	Вых	5	1	нет
Выход объекта	Запрос сигнала на выходе объекта	Input	Вых	12	1	нет
Ко	Коэффициент передачи объекта k_0	Output	Вх	11	2	12,5104
То	Постоянная времени объекта T_0	Output	Вх	11	3	0,2442
T2	Постоянная времени объекта T_2	Output	Вх	11	4	0,2442
to	Время запаздывания объекта $T = T_0/m$, $m = 12$	Output	Вх	12	5	0,2442
1Км	Коэффициент передачи модели Смита k_m	Output	Вх	13	2	12,5104
2Км	Коэффициент передачи модели Смита k_m	Output	Вх	15	2	12,5104
1Тм	Постоянная времени модели Смита T_m	Output	Вх	13	3	0,2442
2Тм	Постоянная времени модели Смита T_m	Output	Вх	15	3	0,2442
1T2	Постоянная времени T_2 модели Смита	Output	Вх	13	4	0,2442
2T2	Постоянная времени T_2 модели Смита	Output	Вх	15	4	0,2442
tcm	Время запаздывания модели Смита $T = T_m/m$, $m = 12$	Output	Вх	14	5	0,2442
Км1	Масштабный коэффициент	Output	Вх	16	3	12,5104
Км2	Масштабный коэффициент	Output	Вх	16	5	12,5104
Кр	Коэффициент передачи регулятора k_p для одноконтурной САР	Output	Вх	7	6	1,5673
Ти	Время интегрирования T_u для одноконтурной САР	Output	Вх	7	7	0,2442

Таблица 2 - Настройки атрибутов каналов (для схемы №2)

Имя канала	Назначение	Тип Input/Output	Пере - менная	№Алго- блока	№Вх /Вых	Множитель
Задание	Запрос сигнала задающего воздействия	Input	Вых	6	1	нет
Возмущение	Запрос сигнала возмущающего воздействия	Input	Вых	5	1	нет
Выход объекта	Запрос сигнала на выходе объекта	Input	Вых	18	1	нет
Ко	Коэффициент передачи объекта k_0	Output	Вх	16	2	12,5104
T3	Постоянная времени объекта T_3	Output	Вх	16	3	0,2442
T4	Постоянная времени объекта T_4	Output	Вх	17	3	0,2442
to	Время запаздывания объекта $T = T_o/m$, $m = 12$	Output	Вх	18	5	0,2442
1Км	Коэффициент передачи модели Смита k_m	Output	Вх	21	2	12,5104
2Км	Коэффициент передачи модели Смита k_m	Output	Вх	24	2	12,5104
1Тм3	Постоянная времени модели Смита T_{m3}	Output	Вх	21	3	0,2442
2Тм3	Постоянная времени модели Смита T_{m3}	Output	Вх	24	3	0,2442
1Тм4	Постоянная времени T_{m4} модели Смита	Output	Вх	22	4	0,2442
2Тм4	Постоянная времени T_{m4} модели Смита	Output	Вх	25	4	0,2442
tcm	Время запаздывания модели Смита $T = T_m/m$, $m = 12$	Output	Вх	23	5	0,2442
Км1	Масштабный коэффициент	Output	Вх	15	3	12,5104
Км2	Масштабный коэффициент	Output	Вх	15	5	12,5104
Кр	Коэффициент передачи регулятора k_p для одноконтурной САР	Output	Вх	7	6	1,5673
Ти	Время интегрирования T_u для одноконтурной САР	Output	Вх	7	7	0,2442

Таблица 3 - Значения каналов при старте

Имя канала	Значение
Ко	1,2
То	30
T2	0
to	1,5
1Км	1,5
2Км	1,5
1Тм	30
2Тм	30
1Т2	0
2Т2	0
tcm	1,2
Км1	Для одноконтурной САР = 0, Для САР с упредителем Смита = -1
Км2	Для одноконтурной САР = 0, Для САР с упредителем Смита = 1
Кр	1,5
Ти	35

4.1.3 Настройки в редакторе представления данных.

- А) Открываем и загружаем проект.
- Б) Добавляем группу экранов (В поле экраны правый клик на значке проекта – добавить группу – ввод имени группы)
- В) Добавляем экран (правый клик на значке группы экранов – добавить экран – ввод имени экрана)
- Г) На экране размещаем необходимые графические элементы и экранные формы для проведения отслеживания хода лабораторной работы, аналогично показанным на рисунке 20 и рисунке 21.

В данной лабораторной работе объект регулирования представлен виртуальной моделью смоделированной про помощи алгоритмов ДИН и ЗАП библиотеки алгоритмов микропроцессорного контроллера «Ремиконт Р – 130». Примером реального объекта регулирования может служить трубчатый теплообменник. Функциональная схема САР температуры выходного потока жидкости теплообменника смешения, выполненная в редакторе представления данных для наглядного проведения лабораторной работы показана на рисунке 22.

Параметры объекта регулирования		Параметры модели объекта предиктора Смита	
Коэффициент передачи	0,0	Коэффициент передачи	13 15
Постоянная времени T_0	0,0	модели Смита	0,0 0,0
Постоянная времени T_2	0	Постоянная времени T_m	0,0 0,0
Время запаздывания T	0,0	Постоянная времени T_2	0 0

Масштабные коэффициенты	
Km1	0
Km2	0

Масштабные коэффициенты:
для одноконтурной САР
Km1 = 0
Km2 = 0

для САР с предиктором Смита
Km1 = -1
Km2 = 1

Параметры алгоритма ДИН в алгоблоке 15 аналогичны	
Время запаздывания модели Смита	0,0

Параметры настройки регулятора для одноконтурной САР	
Коэффициент передачи	0,0
Время интегрирования	0,0

Рисунок 20 - Экранные формы для настройки параметров регулирования.



Рисунок 21 - Экранная форма – тренды

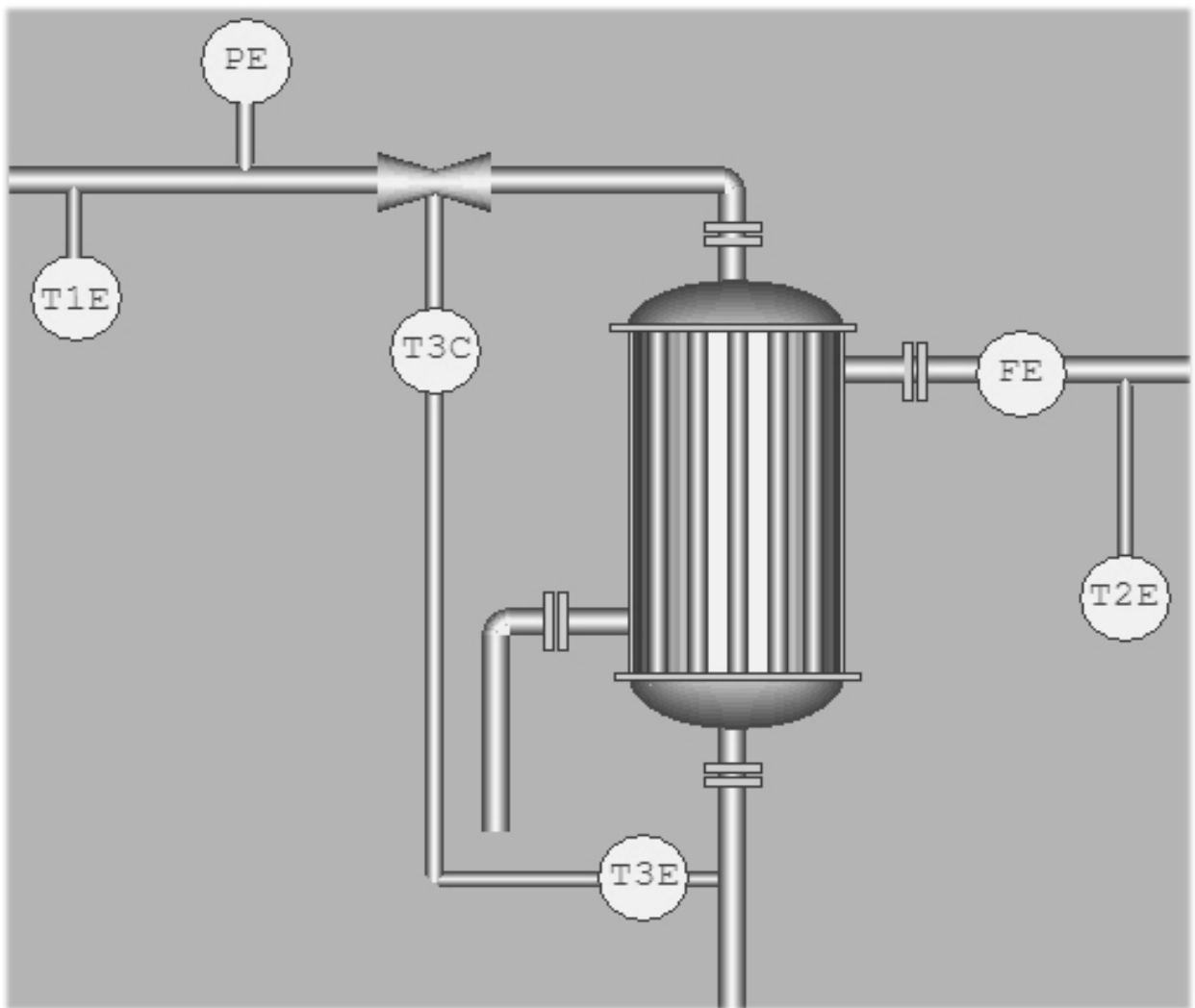


Рисунок 22 - Функциональная схема САР.

Рекомендации :

- 1) Изменение параметров задания, возмущающего воздействия удобнее всего отслеживать на графике (**‘Экранная форма’ - ‘тренды’**) 
- 1) Значения параметров настройки и графики изменения значения на выходе объекта регулирования разместить на разных экранах. Для переключения между экранами назначить специальные кнопки.
- 2) Отображение и изменение коэффициентов объекта регулирования и параметров регулятора осуществить с помощью **‘динамического текста’** 
- 3) Управление включением модели объекта можно осуществить как с помощью **‘динамического текста’**, так и с помощью специальных кнопок.

Параметры настройки кнопок для управления включением модели объекта:

- **«Экранная форма» - «кнопки»**



флажок показать
необходимо поставить

- назначить 2-е функции нажав на кнопку '+'

- в открывшемся окне указать посылку значения в канал —

1) вкладка прямая

2) нажмите кнопку '...' для выбора канала

а) Канал 1Км

б) Атрибут Реальное

в) 'OK'

3) значение 1

4) 'OK'

Для кнопки включения модели объекта: $Km1 = -1$, для канала $Km2 = 1$
 Для кнопки отключения модели объекта: $Km1 = 0$, для канала $Km2 = 0$

4) Рекомендуемые параметры для динамического текста:

- Вид отображения: – число (FLOAT) – .1F
- атрибуты всех каналов – Реальное
- флажок показать - поставить

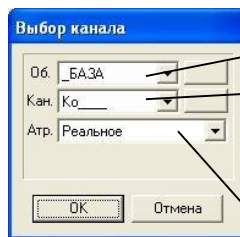
5) Для того чтобы изменить значения коэффициентов в алгоблоках Ремиконт Р-130 используются соответствующие каналы объекта базы каналов аналоговый вывод.

Для `динамического текста` с отображением параметра необходимо назначить Функцию Посылка значения

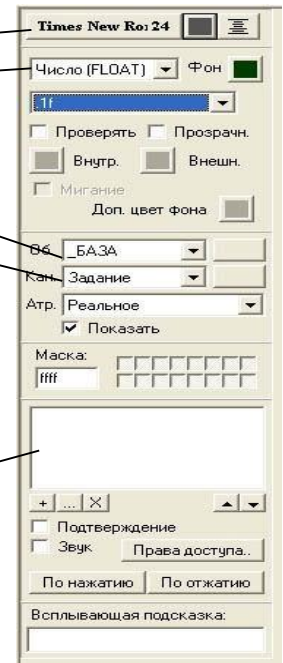


1) Вкладка **Ввод и посылка**

2) нажмите кнопку `...`
Для выбора канала



- Объект – **_БАЗА**
- Канал – Выбрать необходимый канал объекта аналоговый вывод
- Атрибут – **реальное**



4.2 Настройка Ремиконт Р-130

В открытой сети Транзит абонент имеет возможность:

- 1) Запросить значение сигнала на любом выходе любого алгоблока в любом контроллере.
- 2) Запросить значение любого параметра настройки (как константы, так и коэффициента) любого алгоблока в любом контроллере.
- 3) Изменить значение любого коэффициента любого алгоблока в любом контроллере.
- 4) Запросить значения оперативных параметров любого контура или логической программы в любом контроллере.
- 5) Изменить оперативные параметры любого контура или логической программы в любом контроллере.
- 6) Запросить наличие ошибок в любом контроллере.

В контроллере необходимо сконфигурировать программу, представленную на рисунке 23 и настроить коэффициенты из таблицы 3.

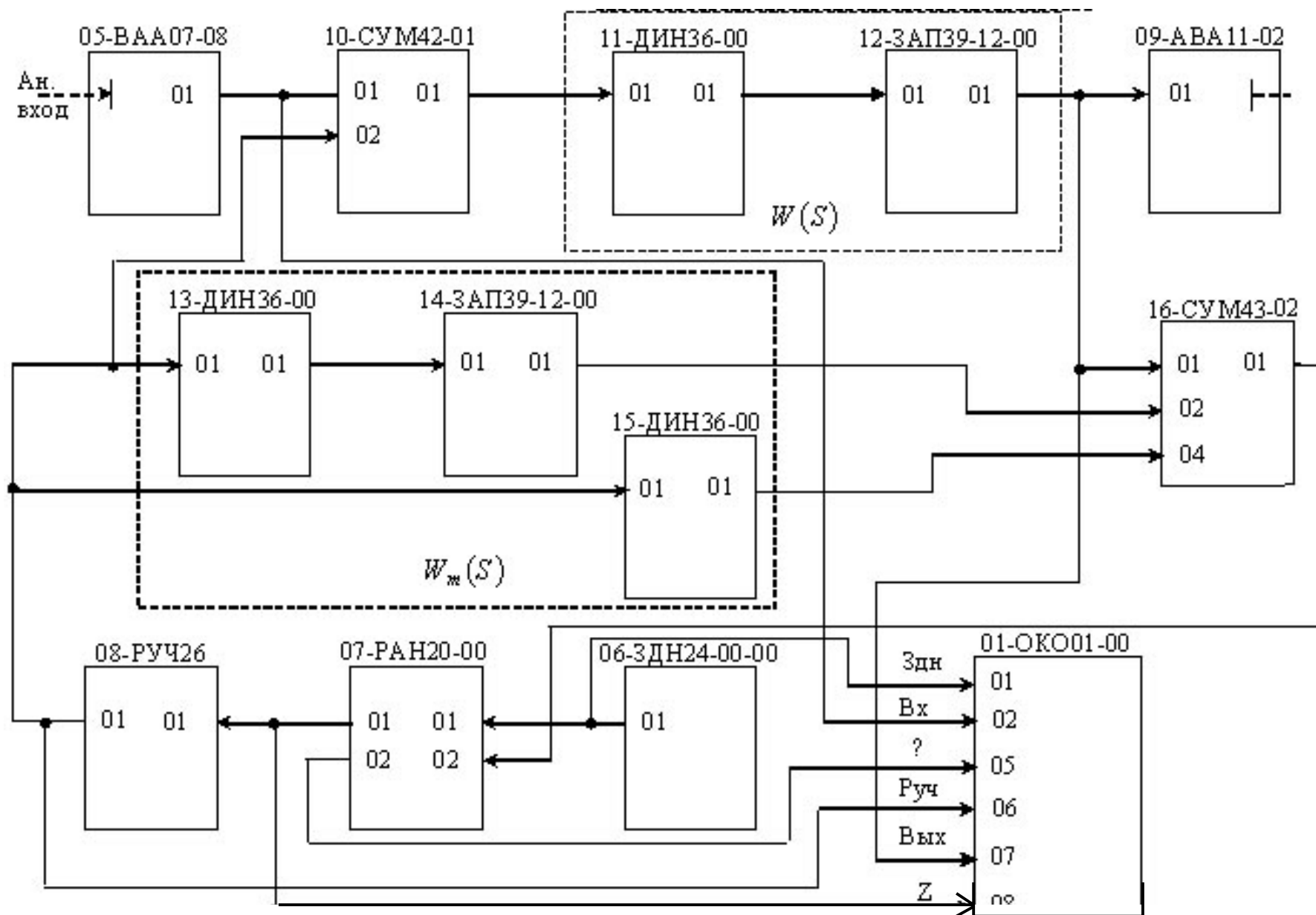


Рисунок 23 – Блок-схема программы, реализованной в контроллере Ремиконт Р-130 (для схемы №1

Таблица 3 - Параметры настройки контроллера Ремиконт Р-130

Алгоритм	№ входа	Наименование	Значение
11 - ДИН	02	Коэффициент передачи объекта k_o	1,2
	03	Постоянная времени объекта T_o , с	30
	04	Постоянная времени T_2 , с	0
12 – ЗАП	05	Время запаздывания объекта $T = \tau_o / m$, с ($m=12$)	1,5
13 – ДИН	02	Коэффициент передачи модели Смита k_m	1,2
	03	Постоянная времени T_m , с	30
	04	Постоянная времени T_2 , с	0
14 – ЗАП	05	Время запаздывания модели Смита $T = \tau_m / m$, с ($m=12$)	1,5
15 – ДИН	02	Коэффициент передачи модели Смита k_m	1,2
	03	Постоянная времени T_m , с	30
	04	Постоянная времени T_2 , с	0
16 – СУМ		Масштабные коэффициенты для одноконтурной САР:	
	03	k_{m1}	0
	05	k_{m2}	0
		Масштабные коэффициенты для САР с упредителем Смита:	
	03	k_{m1}	-1
	05	k_{m2}	1
07 – РАН	06	Для одноконтурной САР: Коэффициент передачи регулятора k_p	1,5
	07	Время интегрирования T_u , с	35

При работе с оперативными параметрами абонент может запрашивать и изменять те же параметры, которые наблюдает и изменяет оператор, работая с лицевой панелью контроллера. Поэтому при настройке микропроцессорного контроллера Р-130 вводить значения коэффициентов не обязательно.

Для связи с абонентом необходимо установить следующие параметры:

- В процедуре `**Системные параметры**` необходимо установить командный режим, и системный номер(02)
- В процедуре `**конфигурация**` параметры свободных входов алгоблоков необходимо установить как коэффициенты.

4.3 Настройка шлюза

Для связи с абонентом используется канал последовательной связи RS232C

В процедуре **‘Приборные параметры’** используется только *общее обнуление* **Код 00**

В процедуре **‘Системные параметры’** установить вид абонента:

- Код 01,
- N = 01 – Связь с ЭВМ

В процедуре **‘Системный номер’**:

- код 00 ,
- N = 00 – 15,
- сетевой номер в сети транзит(01)

Для того чтобы абонент мог выполнять функции оперативного управления и вести настройку, а также получить информацию с выходов и входов алгоритмов задавать в шлюзе какие-либо алгоритмы не требуется.

4.4 Обобщающие действия по настройке системы.

А. Скопировать разработанный проект на сервер, расположенный в лаборатории 1311 и, убедившись в работоспособности и конфигурации микропроцессорного контроллера Ремиконт Р-130 и шлюза, запустить его в профайлере.

Б. Убедиться в том, что информация передаётся и принимается от шлюза и микропроцессорного контроллера Ремиконт Р-130.

В. В случае работоспособности созданной системы оформить отчёт и, представив его преподавателю, получить дополнительное задание.

4.5 Последовательность выполнения дополнительного задания.

1. Получить у преподавателя индивидуальную блок-схему программы для микропроцессорного контроллера Ремиконт Р-130.
2. Выделить в этой программе переменные, которые можно использовать для супервизорного управления и представить их в виде таблицы, аналогичной таблице 1.
3. Разработать проект в редакторе базы каналов.
4. Разработать несколько экранных форм, позволяющие как можно точнее представить всю информацию из микропроцессорного контроллера Ремиконт Р-130
5. Обнулить программу в микропроцессорном контроллере Ремиконт Р-130 и сконфигурировать в нём выданную вам программу.
6. Запустить и протестировать разработанный проект.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Назначение и принцип супервизорного управления.
2. Параметры и переменные контроллера Ремиконт Р-130, допустимые для изменения через Шлюз.
3. Классификация каналов TRACE MODE 5.0.
4. Основные переменные канала TRACE MODE 5.0.
5. Перечень и назначение атрибутов канала TRACE MODE.
6. Последовательность обработки информации в каналах TRACE MODE 5.0.
7. Назначение и настройка процедур первичной и выходной обработки информации в канале TRACE MODE 5.0.

6. СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА

Отчет оформляется в ученической или в общей тетради в соответствии с общими требованиями к оформлению текстовых документов. Отчет должен содержать:

- 1) наименование и цель работы;
- 2) структурная схема программы, реализованной в микропроцессорном контроллере Ремиконт Р-130.
- 3) список переменных, передаваемых в микропроцессорный контроллер, с целью обеспечения супервизорного управления;
- 4) документированное оформление созданного проекта из окна просмотра, вызываемого комбинацией клавиш <Alt + 5>.
- 5) блок схема дополнительного задания.
- 6) таблицу с списком переменных для индивидуального проекта
- 7) документированное оформление индивидуального проекта из окна просмотра, вызываемого комбинацией клавиш <Alt + 5>.

ЛАБОРАТОРНАЯ РАБОТА № 4

"ПРИМЕНЕНИЕ СТАНДАРТНЫХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ В SCADA-СИСТЕМЕ TRACE MODE ДЛЯ МОДЕЛИРОВАНИЯ РАЗОМКНУТЫХ И ЗАМКНУТЫХ АВТОМАТИЧЕСКИХ СИСТЕМ"

1. ВВЕДЕНИЕ

Реализованные в инструментальной системе TRACE MODE 5 язык схем на функциональных блоках (Техно FBD) и язык инструкций (Техно IL) существенно расширены по сравнению с базовыми требованиями стандарта, включают набор из более чем 150 элементарных и библиотечных функций. Среди встроенных алгоритмов ПИД, ПДД, нечеткое, позиционное регулирование, ШИМ преобразование, статистические функции, функции расчета техника-экономических показателей (ТЭП) и т.д.

Существенным развитием стандарта является добавление ряда интегрированных функциональных блоков управления типовыми технологическими объектами (клапан, задвижка, привод, мотор, насос, группа моторов и т.д.). Кроме того, проектировщик имеет возможность наращивать библиотеки языков своими собственными функциями, учитывающими особенности задач, решаемых в его проектах. Разработки пользовательских функциональных блоков можно осуществлять на языке Техно IL, или C.

2. ЦЕЛЬ РАБОТЫ

В данной работе студенты должны:

- во-первых, освоить программирование моделей поведения объектов с помощью стандартного языка функциональных блоков.
- во-вторых, научиться настраивать и связывать каналы между собой в пределах одного узла;
- в-третьих, разработать законченную задачу управления системой автоматического регулирования на диспетчерском уровне.

3. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

3.1. Языки программирования алгоритмов управления

Встроенных функций первичной и выходной обработки каналов может быть недостаточно для выполнения сложных алгоритмов обработки данных, регулирования и управления. Такие алгоритмы разрабатываются в виде отдельных программ и могут вызываться процедурами канала. Для разработки таких программ в Трейс Моуд предусмотрены языки: язык функциональных блоков (ТехноFBD) со встроенным языком релейно-контактных схем ТехноLD и язык инструкций (ТехноIL).

Языки ТехноFBD и ТехноLD предназначены для разработки алгоритмов в виде схем функциональных блоков. Созданные программы могут вызываться из процедур каналов («Трансляция» и «Управление»).

Программы на ТехноIL записываются в виде последовательности инструкций. Этот язык позволяет программировать функциональные блоки для языка ТехноFBD и создавать метапрограммы, которые запускаются параллельно с пересчетом базы каналов.

Языки реализуют стандарт МЭК-61131/3. Кроме функций, описанных в этом стандарте, в них встроено большое число дополнительных функций.

3.1.1 Язык ТехноFBD

Язык функциональных блоков является языком визуального программирования. При создании программы на языке ТехноFBD функциональные блоки с заданными функциями размещаются в поле редактирования, осуществляется настройка их входов и выходов, а также связь их между собой в диаграмму, реализующую требуемую функцию. Программа, созданная на этом языке, называется FBD-программой.

Разработка и отладка FBD-программ и LD-программ осуществляется в специальных окнах редактора базы каналов:

- для входа в одно из них выполняется командой **«FBD-программы»** меню **«Окна»** или нажатием сочетания клавиш **ALT-3**, или кнопкой ;
- для входа во второе - командой **LD-программы** меню **«Окна»**, или нажатием сочетания клавиш **ALT-6**.

При этом на экране появляется диалог, вид которого показан на рисунке 24. В этом диалоге выбирается программа для редактирования. Можно изменить ее имя, создать новую и удалить существующую программу, сохранить ее в файле или выбрать из файла.

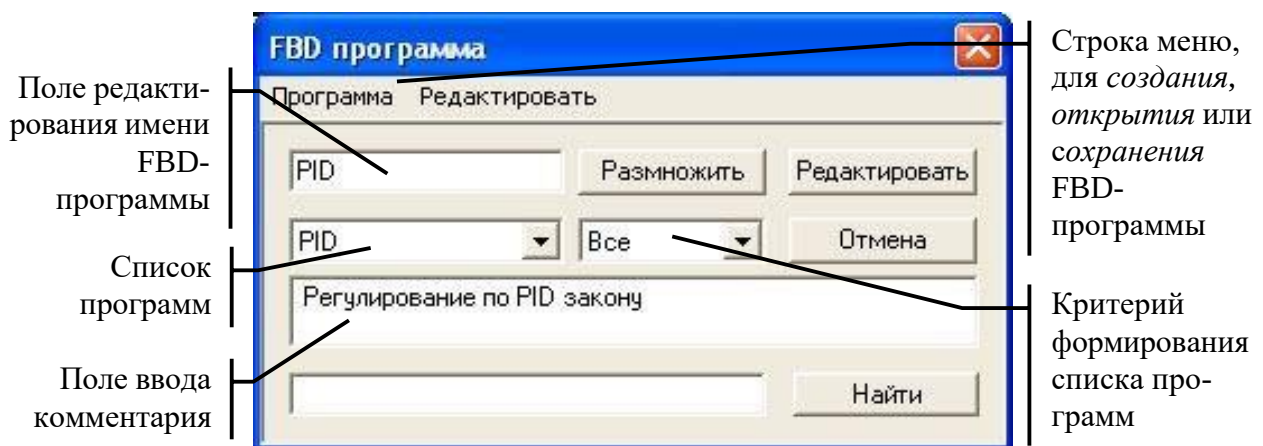


Рисунок 24 – Диалог выбора FBD-программы

В окне **FBD-программа**, можно:

- создать новую программу (команду **Создать** из меню **Программа**);
- удалить существующую программу;

- выбрать программу для редактирования;
- изменить имя программы;
- сохранить программу в файл с расширением **.crm**;
- сохранить программу в виде текста;
- вставить программу из файла;

FBD-программа имеет **имя** и **комментарий**

- **Имя** FBD-программы используется для ссылок на нее.
- **Комментарий** – это дополнительная информация о назначении программы. Он выводится во всех диалогах, ссылающихся на FBD-программы.

По умолчанию FBD-программа создается для одного узла. Чтобы использовать созданную программу на всех узлах проекта, нужно выполнить команду «Размножить» диалога FBD-программы. Полученные с помощью данной процедуры копии программы могут в дальнейшем редактироваться независимо друг от друга.

Нажатием ЛК мыши на кнопку «**Редактировать**» осуществляется переход в окно редактирования, показанное на рисунке 25.

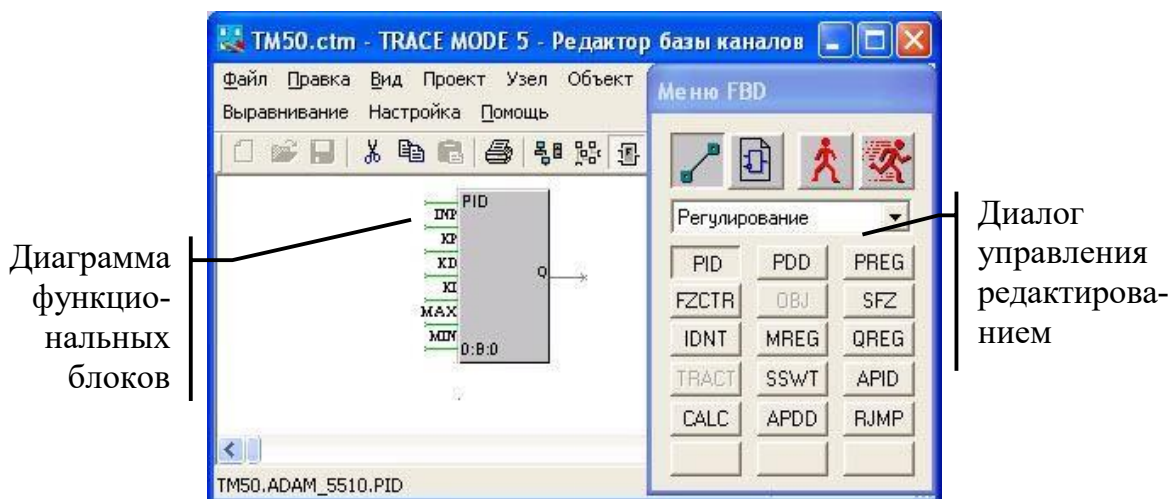


Рисунок 25 – Окно редактирования FBD-программы

Элементарным звеном на языке ТехноFBD является функциональный блок, пример которого, показан на рисунке 26.

- ♦ **Блок** - это графическое изображение вызова одной из функций, реализованных в языке. Это могут быть либо стандартные, встроенные в систему функции, либо разработанные пользователем на языке Техно IL.

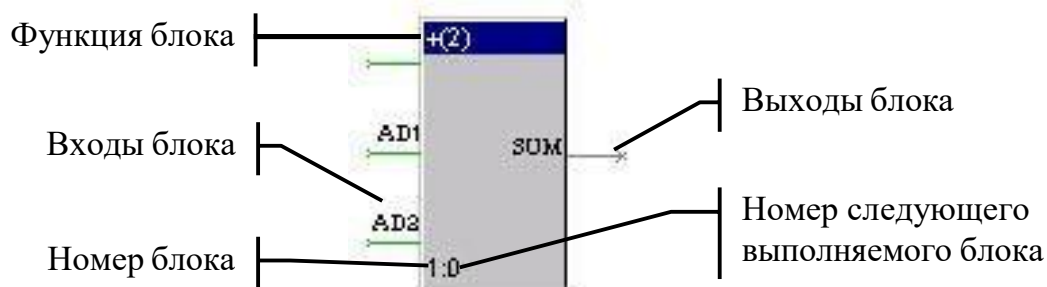


Рисунок 26 – Пример функционального блока

Существуют три режима работы в окне редактирования FBD-программ:

- редактирование связей;
- размещение блоков;
- эмуляция работы функции.

Выбор режима осуществляется из диалога управления редактированием. Этот диалог называется **Меню FBD** и появляется на экране при входе в окно редактирования FBD-программ. Вид этого диалога показан на рисунке 37.

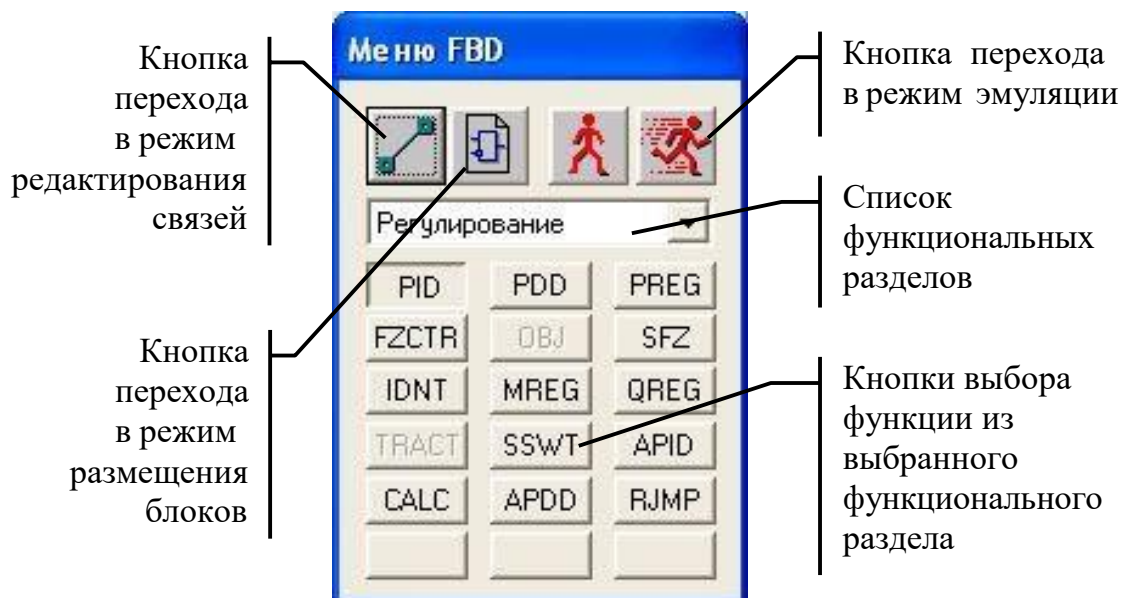


Рисунок 27 – Окно диалога для управления редактированием

Чаще всего приходится работать в первом из этих режимов, но на самом первом этапе всё-таки нужно разместить функциональные блоки. Рассмотрим же все три режима работы окна редактирования FBD-программ последовательно:

1) Режим размещение блоков

Для входа в режим размещения новых функциональных блоков надо нажать ЛК на иконке диалога Меню FBD .

После этого любое нажатие ЛК на свободном месте рабочего поля окна размещает новый функциональный блок.

Каждый функциональный блок имеет следующие атрибуты:

- выполняемую функцию;
- номер;
- входы и выходы.

А) наименование выполняемой функции выводится в его верхней части. Функция блока определяет действия, выполняемые с его входами при формировании значений выходов. Функция блока выбирается перед его размещением. Изменение функции блока после его размещения недоступно.

Б) номер функционального блока устанавливается автоматически

при его размещении в рабочем поле и недоступен для изменения. Он используется только для индикации очередности выполнения функциональных блоков при вызове программы.

В) **номер следующего выполняемого блока** определяется автоматически. Он записывается за номером текущего блока через символ «:». Для первого блока в этой области перед номером записывается символ **В**, а для последнего - **Е**.

Г) каждый блок в зависимости от выполняемой функции имеет определенное количество **входов** и **выходов**. Входы всегда расположены слева, а выходы - справа. Назначение входов и выходов блока определяется его функцией. Исключением является первый вход. Он всегда управляет пересчетом блока. При вычислении блока над переменными, связанными с его входами, осуществляются действия, определенные функцией блока. Полученные в результате значения присваиваются переменным, связанным с выходами блока.

В ТРЕЙС МОУД имеется около 150 стандартных функций. Они разбиты на 13 функциональных разделов. Кроме того, предусмотрено еще три раздела для произвольно программируемых функциональных блоков. Выбор функции для блока осуществляется в два этапа. На первом этапе выбирается функциональный раздел, на втором - функция.

Функциональный раздел выбирается из списка, расположенного под кнопками установки режима. На функциональных кнопках диалога Меню FBD выводятся наименования соответствующих им функций в выбранном, разделе. Нажатие ЛК на любой из кнопок устанавливает ее функцию в качестве текущей.

Функции могут быть: а) встроенные в систему; б) разработанные пользователем на языке Техно IL; в) как внешние DLL модули.

При размещении нового блока список переменных программы автоматически увеличивается. Число добавляемых переменных равно количеству выходов на новом блоке и зависит от его функции.

2) Режим редактирования связей

Для входа в режим редактирования связей надо нажать ЛК на иконке диалога **Меню FBD**



В режиме редактирования связей можно настроить структуру обработки данных. В этом режиме доступны следующие операции:

- *настройка входов и выходов функциональных блоков;*
- *связывание входов и выходов функциональных блоков;*
- *удаление связей;*
- *удаление блоков;*
- *перемещение блоков.*

А) для **настройки входа или выхода** функционального блока надо

дважды нажать на нем ЛК. При этом на экране появится диалог **Описание переменной**, показанный на рисунке 28. Этот диалог позволяет задать тип переменной, вид представления и значение.

Кроме того, в этом диалоге можно ввести комментарий к настраиваемому входу или выходу. Этот комментарий выводится в бланках диалога **Реквизиты** для переменных тип **аргумент** и **константа** при настройке вызова программы.

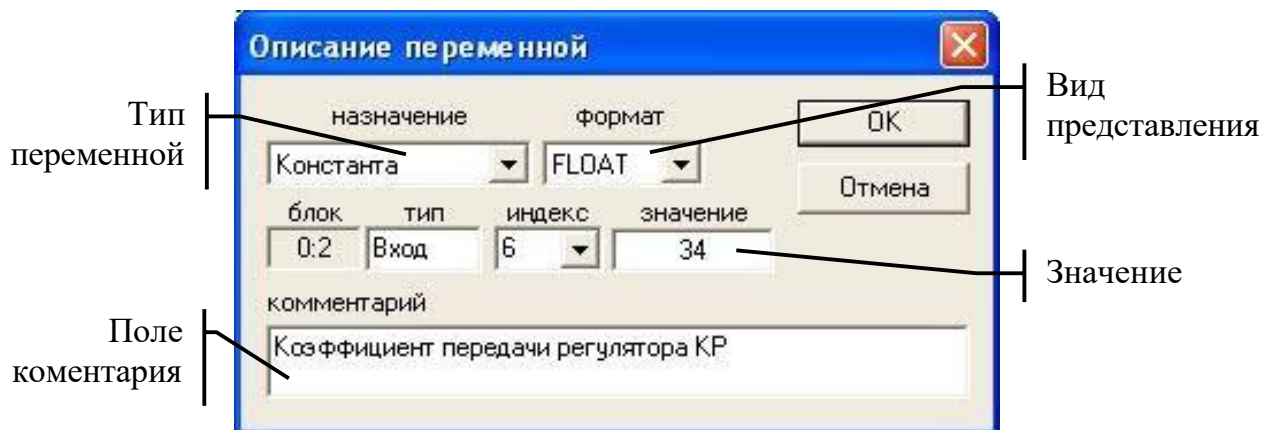


Рисунок 28 – Окно описания переменной

Тип переменной определяет переменную, с которой он будет связан. Это может быть либо внутренняя переменная, либо константа, либо внешняя переменная – аргумент. Входы и выходы блоков могут иметь один из следующих типов:

- *Свободен*, устанавливается автоматически для всех незадействованных входов. Этот тип недоступен для установки вручную;
- *Блок*, устанавливается для всех выходов при размещении блока. Вход или выход, имеющий данный тип, связан с внутренней переменной программы. Этот тип автоматически устанавливается для входа, связанного с выходом любого блока;
- *Константа*, может быть задан только для входа. Такой вход имеет фиксированное значение и не может меняться в процессе выполнения программы;
- *Аргумент*, это вход или выход программы, связываемые с внешними переменными. Такими переменными являются атрибуты каналов.

Б) Связывание функциональных блоков. При разработке алгоритмов, выполняющих несколько последовательных действий, необходимо передать результаты вычислений одного блока другому. Для этого надо связать соответствующие входы и выходы этих блоков.

Для установления такой связи следует нажать ЛК на одном из ее концов и, удерживая кнопку мыши нажатой, перевести

курсор в область другого конца связи. При этом соответствующие вход и выход функциональных блоков соединятся линией. Связанным входам и выходам автоматически присваивается тип **блок**.

Связывать можно не только вход с выходом, но и вход со входом. В этом случае линии соединения не проводятся, однако связь устанавливается. При выделении любого из связанных между собой входов цвет всех остальных также меняется на красный.

Чтобы посмотреть все связи выхода функционального блока, следует нажать на нем ЛК. При этом линии связи этого выхода с другими блоками выделяются красным цветом. Нажатие ЛК на входе блока не выделяет его связи. Однако при этом выделяются все входы других блоков, связанных с тем же источником.

Для удаления связи между блоками надо нажать ЛК на входе в блок. Нажатие клавиши DEL после этого приводит к удалению выделенной связи. После удаления связи входам блоков устанавливается тип **свободен**, а выходам - **блок**.

В) Удаление блока. Чтобы удалить функциональный блок из редактируемой программы, его надо сначала выделить. Выделение блока осуществляется нажатием ЛК на его изображении. Нажатие клавиши DEL после этого приводит к удалению из программы выделенного блока. Все связи этого блока также удаляются. Входам блоков, связанных с выходами удаляемого блока, устанавливается тип **свободен**, а выходам - **блок**.

Г) Перемещение блока. Чтобы сделать программу читаемой и менять последовательность пересчета блоков, предусмотрена возможность перемещения блока с сохранением всех определенных для него связей.

Для перемещения блока надо сначала выделить его нажатием на нем ЛК. Затем, удерживая нажатой ЛК, следует перевести курсор мыши в требуемую область рабочего поля. После того как ЛК будет отпущена, выделенный блок переместится в указанное место.

3) Режим эмуляции работы программы

Этот режим работы редактора предназначен для отладки разрабатываемой программы. Переход в режим эмуляции осуществляется по нажатию ЛК на соответствующей иконке диалога **Меню FBD** 

После перехода в этот режим каждое нажатие ЛК в рабочем поле приводит к однократному выполнению редактируемой функции.

При работе в режиме эмуляции рядом с каждым входом и выходом всех функциональных блоков выводятся их значения. Величины всех

входов, имеющих тип **константа**, при эмуляции остаются неизменными, равными заданным для них значениям. Входы с типом **аргумент** принимают значение 0.

Чтобы изменить значения входов типа **константа** и задать значения входов типа **аргумент**, используется специальный диалог **Значение переменной**. Этот диалог появляется на экране при нажатии ЛК на любом входе любого функционального блока. Заданное в этом диалоге значение присваивается соответствующему входу и остается неизменным до нового ввода.

Текущие значения всех переменных и констант выводятся в режиме отладки рядом с соответствующими входами и выходами блоков. Чтобы отслеживать изменение какой-либо одной переменной, следует выделить соответствующий ей вход или выход. При этом его текущее значение выводится в строке статуса окна редактора базы каналов.

3.1.2 Вызов FBD-программ

FBD-программы вызываются процедурами каналов: **трансляция** и **управление**. Существует подтип каналов **УПРАВЛЕНИЕ**. Он не имеет значений и атрибутов, кроме состояния, имени и кодировки. Для каналов этого подтипа определена только процедура **управление**, позволяющая вызвать FBD-программу.

При настройке вызова FBD-программы используются соответствующие бланки диалога **Реквизиты**. В них задается имя программы, настраивается связь ее аргументов с атрибутами каналов текущей базы и задаются значения констант, передаваемых в программу (рисунок 29).



Рисунок 29 – Окно реквизитов для вызова FBD-программы

После выбора программы в специальном окне выводится для настройки

список ее аргументов и констант. Цифры в квадратных скобках соответствуют выходам и входам блоков программы.

Каждая строка списка аргументов программы содержит характеристику аргумента и связанной с ним переменной базы каналов. Она имеет следующий формат:

nn <имя объекта>:<имя канала>_<атрибут>,

где **nn** – характеристика аргумента:

=> - значение аргумента формируется программой;

<= - значение аргумента передается программе.

Для изменения канала и атрибута, связанного с аргументом FBD-программы или для изменения значения константы надо **дважды нажать ЛК** на соответствующей строке. При этом появится окно, показанное на рисунке 30.

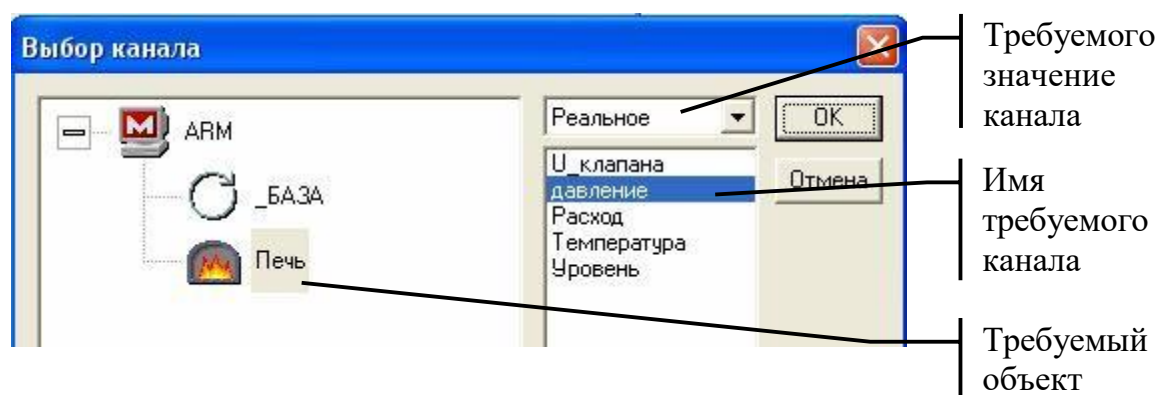


Рисунок 30 – Окно для **выбора значения канала** в качестве атрибута

3.1.3. Язык ТехноLD

В окне LD-программы блоки имеют представление и средства редактирования, соответствующие стандарту языка LD.

В окне **LD программы** для блоков раздела **LD-функции** существуют дополнительные элементы соответствующие стандарту языка LD, не используемые в окне FBD-программы.

Перечислим эти средства редактирования:

- **Отображение связи входов.**

Соединенные входы принимают тип, индекс и значение входа, к которому проведена связь.

Если значение соединенных входов равно нулю или тип таких входов - **аргумент**, связь невидима. Кроме того, связь видима только в том случае, если блоки расположены один под другим. Чтобы удалить связь, нужно вручную изменить индекс одного из соединенных входов.

Это свойство может быть использовано для прорисовки левой ши-

ны питания, предусмотренной стандартом языка LD. Для этого надо расположить блоки один под другим, задать ненулевое значение входа одного из них, установить тип этого входа **константа** и далее провести к этому входу связь от входа другого блока. Результат описанной процедуры показан на рисунке 31.

Рисунок 31 – Часть LD-программы с видимой общей шиной

- Правая шина питания.

Эту функцию выполняет блок **LOR**.

- Особенности комментария к связанной переменной.

Связанная переменная настраивается таким же образом, как вход или выход блока.

Комментарий выводится на блоке вместо заданного по умолчанию слова **"VAR"**. Знак ";" выполняет в комментарии функцию разделителя – на блок выводится текст до этого знака (не более 10 символов). Например, если задать комментарий как

MY_VAR_3; set by (S)

на блоке отобразится MY_VAR_3

- Установка привязки нескольких блоков к одной переменной.

Связанные переменные можно соединять так же, как входы или выходы блоков. Результат такого действия может быть различным.

При соединении переменных блоков-контактов итоговая переменная принимает тип, индекс и значение переменной, к которой проведена связь.

При соединении переменных контакта и катушки итоговая переменная принимает тип и индекс переменной катушки.

При соединении переменных катушек в диаграмму блоков добавляется блок **LOR**, к входам которого автоматически подключаются соединяемые переменные.

- Удаление блока. Как и в окне **FBD-программы**, чтобы удалить блок, его надо выделить нажатием ЛК и нажать клавишу DEL. Однако удаление не произойдет, если переменная блока выделена (красным цветом). Чтобы снять выделение переменной, нужно нажать ЛК на изображении функции блока.

3.1.4. Язык ТехноIL

Язык инструкций (ТехноIL) - это текстовый язык для разработки программ, реализующих функции обработки данных и управления. Он является расширением языка IL.

Программы IL разрабатываются в текстовом виде как последовательность инструкций, содержащих команды, операнды и операторы.

Разработанные и отлаженные в Трейс Моуд IL-программы могут использоваться другими инструментальными средствами программирования контроллеров. Можно использовать расширенные возможности Техно1Б и для оформления программ.

Программы на языке ТехноIL разрабатываются в специальном диалоге, вход в который выполняется командой «Создать» меню «ТехноIL». Редактирование программы осуществляется непосредственным вводом текста с клавиатуры в левом окне.

Тип разрабатываемой программы указывается в специальном поле диалога «ТехноГБ». Здесь при разработке функционального блока следует установить значение FB, а для метапрограмм - значение PRG.

Разработанные на ТехноГБ функциональные блоки помещаются в один из трех разделов: «ТехноIL_1», «ТехноIL_2», «ТехноIL_3». Заполнение функций этих разделов осуществляется последовательно, по мере добавления новых блоков.

Метапрограммы управления и обработки данных могут ссылаться на значения атрибутов каналов, а также вызывать в качестве подпрограмм стандартные функциональные блоки. Максимальное количество таких программ равно 16.

Чтобы подключить IL-программу к системе, следует произвести ее трансляцию, используя кнопку «Трансляция». Сообщения о результатах трансляции выводятся в правом окне диалога «ТехноIL». Если трансляция программы завершена успешно, то ее можно добавить в систему кнопкой «Добавить». Если программа имеет тип FB, то в один из трех разделов (ТехноIL_1, ТехноIL_2 или ТехноIL_3) будет добавлен новый блок, имя которого совпадает с именем IL-программы. Если тип программы был PRG, то после выполнения операции добавления к базе каналов текущего узла будет подключена новая метапрограмма.

Для редактирования добавленных в систему IL-программ надо выполнить команду «Редактировать» из меню «ТехноIL».

3.2. Несколько примеров создания моделей объекта.

В настоящее время всё больше проектов АСУ ТП имеют сложную, чаще всего распределённую архитектуру. Совмещение в себе функций административного уровня только добавляют количество узлов в создаваемых проектах.

В таких сложных системах, для повышения надёжности систем управления функции сбора информации и визуализации хода технологического процесса пытаются разделить, создав для них отдельные узлы. Таким образом оператор производит наблюдение и супервизорное управление с АРМ, на котором нет драйверов и задач сбора информации от контроллеров нижнего уровня и от плат УСО. Эти задачи решаются на отдельных узлах, которые условно можно назвать серверами сбора информации.

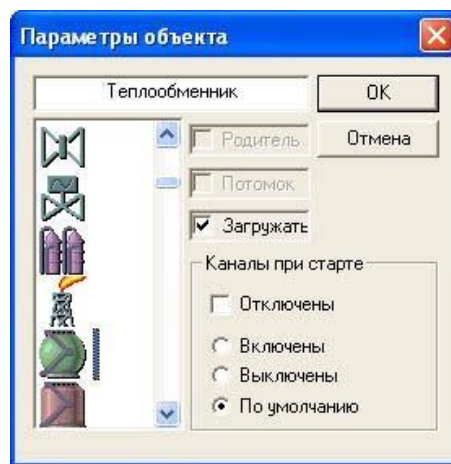
Другой задачей, помимо надёжности, является то, что алгоритмы управления желательно отлаживать до внедрения проекта в реальное производства, а это возможно только при моделировании моделей поведения объекта.

Поэтому остановимся на моделировании статических и динамических свойств объекта поподробнее.

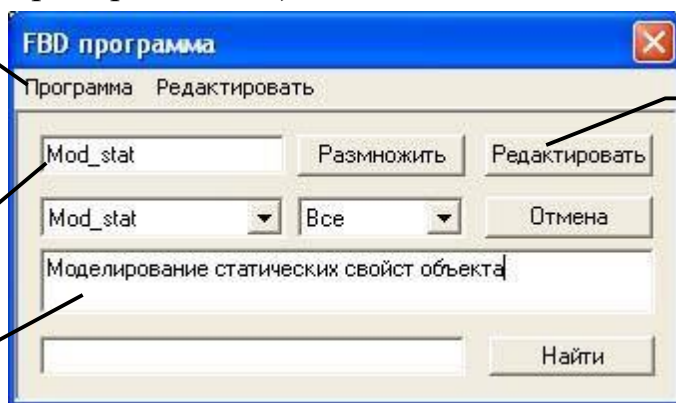
3.2.1. Создание статической модели объекта

Данная модель будет случайным образом изменять значение параметра вокруг в пределах зоны, обозначенной средним значением

1. Создайте в базе каналов объект, и переименуйте его названием моделируемого аппарата или участка.
2. Создайте в этом объекте два канала:
 - а) один по названию канала, например 1_Tempreg;
 - б) второй – среднее значение этого же параметра - 1_Temp_sred.
3. Перейдите в окно редактирования FBD-программ (ALT-3) и создайте программу, переименовав её (например Mod_stat):



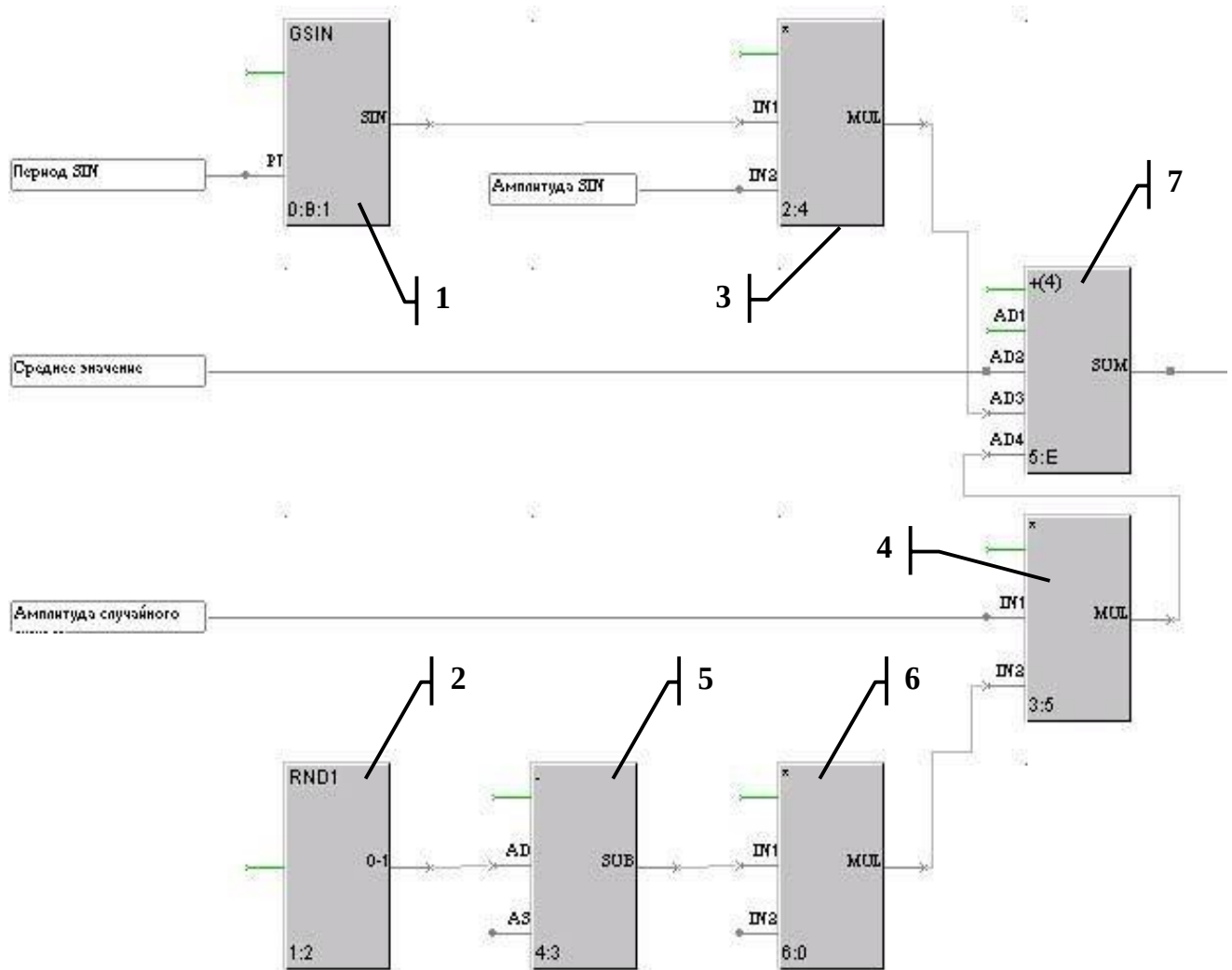
- а) Меню «программа», «создать»
 - б) Переименовать, набрав новое имя
 - в) Пояснения к программе
 - г) Нажать кнопку «Редактировать» Для перехода в окно редактирования FBD-программы
4. Перейдите в режим размещения блоков  и последовательно распо-



ложить в окне редактора 7 функциональных блоков:

- а) «GSIN» и «RND1» из группы «Генераторы» для генерирования сигнала синуса и случайной единицы (1 и 2);
- б) два блока «*» из группы «Арифметические» для масштабирования амплитуды Синуса и Случайной единицы (3 и 4);
- в) один блок «-» и один блок «*» из группы «Арифметические» для приведения диапазона колебания случайного числа от -1 до 1 (5 и 6);
- г) один блок «+(4)» из группы «Арифметические» для сложения влияния синуса и случайного числа и поднятия до уровня среднего значения (7).

5. Перейдите в режим редактирования связей  и соедините функциональные блоки как указано ниже:



6. Настройте параметры входов для блоков:

Входы, которые далее не описаны используются для межблочного соединения и имеют атрибут *блок*.

1 блок: вход PT – *константа* (Период SIN)

2 блок: входы не настраиваются

3 блок: вход INT2 – *константа* (Амплитуда SIN)

4 блок: вход INT1 – *константа* (Амплитуда случайного сигнала)

5 блок: вход AS – *константа* – значение = 0.5 (Вычитание половины единицы для нормирования случайной единицы вокруг нуля)

6 блок: вход INT2 – *константа* – значение = 2 (Увеличение амплитуды случайной единицы в 2 раза для диапазона от -1 до 1)

7 блок: вход AD1 – не используется

вход AD2 – *аргумент* (Среднее значение)

выход SUM – *аргумент* (Выход сгенерированного случайного сигнала)

7. Перейдите в окно редактирования базы каналов (Alt-2) и откройте созданный ранее объект.

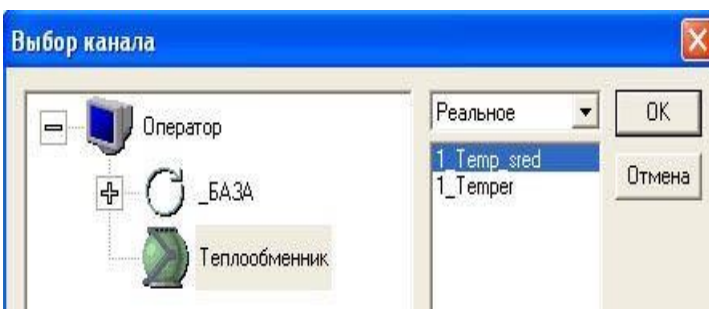
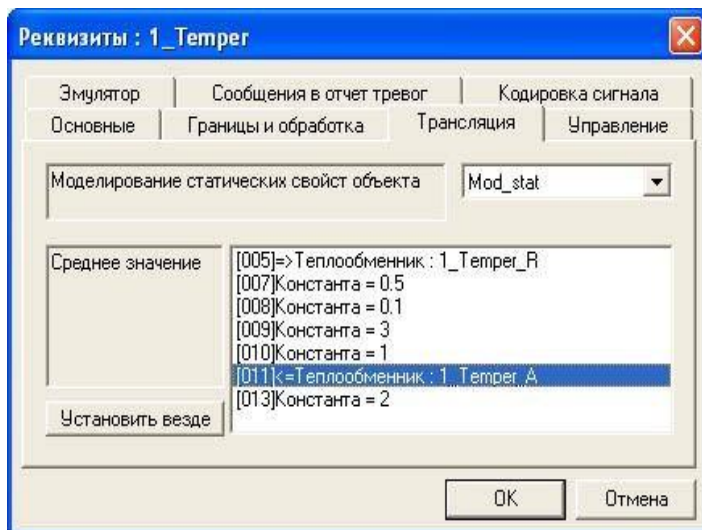
8. Двойным нажатием ЛК мыши на имени канала 1_Temper откройте окно его реквизитов.

9. Перейдите в закладку трансляция и подключите программу, созданную ранее.

10. Выход программы (строка с символом “>=”) свяжите с реальным значением текущего канала (1_Temper)

Вход “(011)<=” с реальным значением второго канала (1_Temper_sred).

Для этого дважды кликните на данной строке ЛК мыши и откроется окно выбора канала и его атрибута.



11. Аналогичным образом задайте константы для амплитуды синуса и случайной единицы с учётом, что их общая сумма должна быть равна половине *диапазона изменения технологического параметра*.

12. Сохраните проект и закройте редактор базы каналов.

13. Откройте редактор представления данных и загрузите в нём созданный ранее проект.

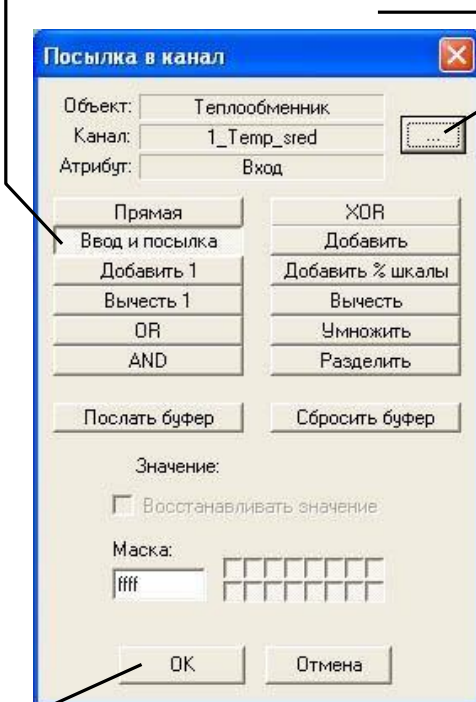
14. На экранной форме, в месте, где необходимо показать значение смо-



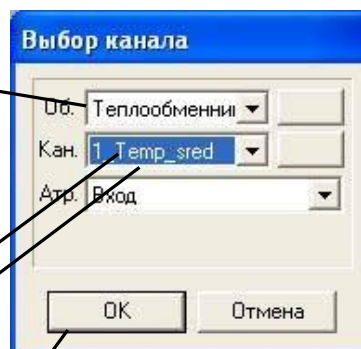
делированного технологического параметра, создайте динамический объект типа «Динамический текст»

15. В свойствах этого объекта укажите:

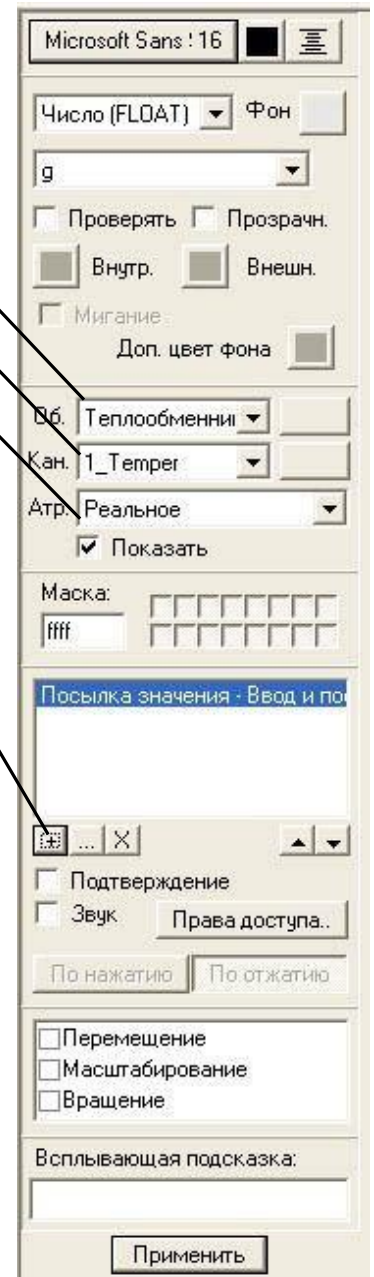
- а) имя объекта
- б) имя канала
- в) значение канала
- г) добавить действие «посылка значения»
- д) настроить «ввод и посылку»
- е) нажать клавишу «...» для изменения канала



- ж) в открывшемся окне укажите: имя объекта
- имя канала для изменения среднего значения (1_Temp_sred)
- входное значение



- з) нажатием клавиши «OK» подтвердите свои действия



16. Нарисуйте динамический объект (или примените изменения), сохраните проект и запустите его на выполнение.

3.2.2. Создание динамической модели объекта

Для создания самой простой модели поведения объекта в динамике необходимо так же создать два канала. Только в отличии от предыдущего примера, первый канал будет являться входом объекта, а второй канал выходом объекта.

Для примера возьмём тот же теплообменник, где вода подогревается с помощью подачи пара. Где $F_{\text{пара}}$ – расход пара;



$T_{\text{воды}}$ – температура воды на выходе из теплообменника.

1. Создайте в базе каналов объект, и переименуйте его названием моделируемого аппарата или участка.
2. Создайте в этом объекте два канала:
 - а) один по названию входного канала, например **$F_{\text{пара}}$**
 - б) второй по названию выходного канала, например **$T_{\text{воды}}$** .
3. Перейдите в окно редактирования FBD-программ (ALT-3) и создайте программу, переименовав её (например АП1_ЗЧЗ).
4. Перейдите в режим размещения блоков  и расположить в окне редактора один функциональный блок «**OBJ**» из группы «Регулирование».
5. Перейдите в режим редактирования связей  и настройте параметры входов и выхода расположенного блока:

вход INP – **аргумент** (Вход объекта)

вход K – **константа** (Коэффициент передачи объекта)

вход T – **константа** (Постоянная времени объекта)

вход N – **константа** (Время запаздывания объекта - задаётся в количестве циклов отставания выхода от входа может быть от 0 до 4)

вход IS – **константа** (код помехи)

выход Q – **аргумент** (Выход объекта)

6. Перейдите в окно редактирования базы каналов (Alt-2) и откройте созданный ранее объект.
7. Двойным нажатием ЛК мыши на имени канала **$T_{\text{воды}}$** откройте окно его реквизитов.
9. Перейдите в закладку трансляция и подключите программу, созданную ранее.
10. Выход программы (строка с символом “>=”) свяжите с реальным значением текущего канала (**$T_{\text{воды}}$**).
Вход программы (строка с символом “>=”) свяжите с реальным значе-

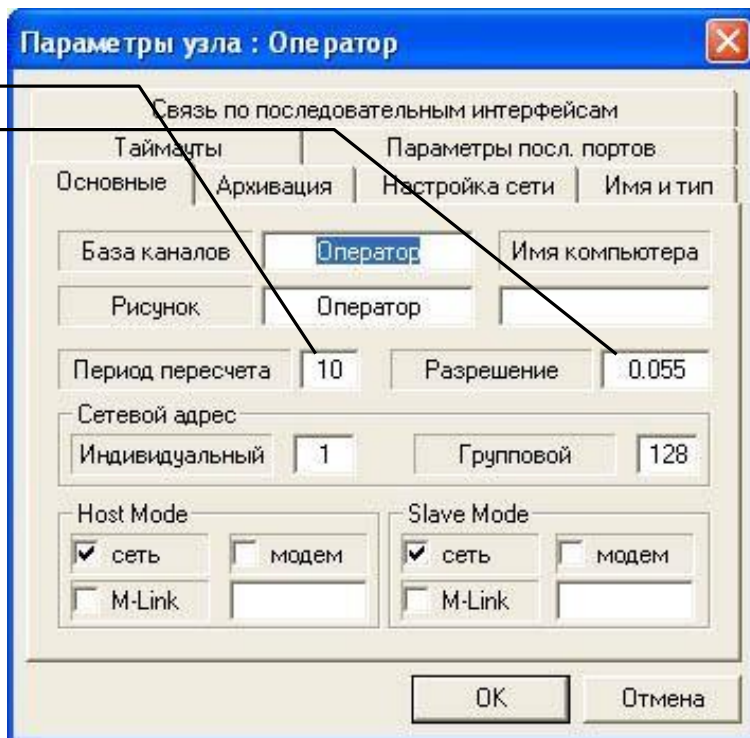
нием канала **F_пара**, являющимся входным для нашего объекта.

11. Аналогичным образом задайте константы для параметров вашего объекта (коэффициент передачи, постоянную времени и время запаздывания).

12. Откройте окно параметров узла, кликнув на значке узла ПК мыши.

13. В открывшемся окне задайте такой период пересчёта и разрешение, что бы их произведение было кратно вашему времени запаздывания.

Это произведение определит время пересчёта базы канала. Все временные константы будут помножаться на это значение. Запаздывание будет указываться в количестве циклов пересчёта, на которое значение будет отставать, и это значение должно быть целым числом.



14. Сохраните проект и закройте редактор базы каналов.

15. Откройте редактор представления данных и загрузите в нём созданный (или отредактированный) ранее проект.

16. На экранной форме, в месте, где необходимо показать значение входного технологического параметра **F_пара**, создайте динамический объект типа «Динамический текст».

17. Свяжите его с **реальным** значением канала **F_пара** и задайте действие для ввода **входного** значения этого же канала.

18. В месте, где необходимо отображать выходное значение технологического параметра **T_воды**, создайте такой же динамический объект, и свяжите его с **реальным** значением канала **T_воды**. Действия для этого канала задавать не нужно.

19. В свободном месте экрана создайте Тренд, в котором укажите для отображения реальные значения входного и выходного каналов.

4. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Данная работа состоит из 2-х частей. Вначале необходимо разработать локальный проект, в котором должно быть как минимум один статический и один динамический объекты. При этом можно начать новый проект выданный преподавателем, а можно продолжить работу с проектом «БЫСТРЫЙ_СТАРТ2», доработанный во второй части

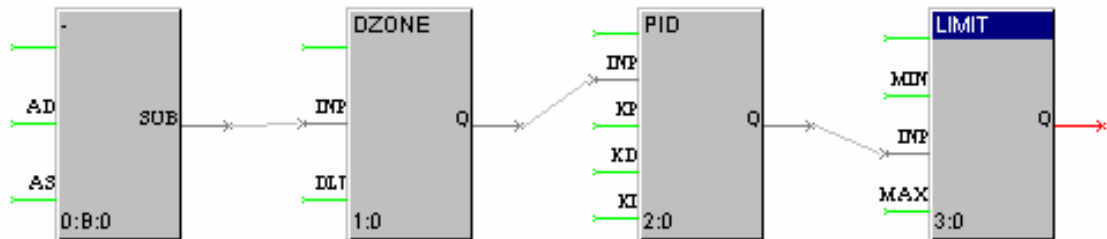
Задание №1.

1. Уточнить у преподавателя описание и схему технологического процесса.
2. Разработать для данного процесса приблизительную структуру проекта, представив её в виде совокупности экранов. Схему представить графически, перечислив назначение каждого экрана и функции оператора, которые можно сделать с её помощью.
3. Выбрать из разработанной схемы участок процесса, представляемый одной экранной формой, но имеющий как минимум один контур регулирования, и нарисовать приблизительную схему расположения аппаратов и динамических объектов отображения и ввода информации, которые предполагается на ней разместить.
4. Перечислить все технологические параметры, которые будут отображаться на этой экранной форме. Можно это сделать в виде таблицы контролируемых параметров. Для каждого из параметров обязательно укажите диапазон изменения и размерность.
5. Создать проект в редакторе базы каналов и для каждого из технологических параметров создать канал, сокращённо обозначив его.
6. Создайте программу моделирования статических свойств объекта.
7. Выберите один из каналов и подключите к нему созданную программу с помощью процедуры трансляции.
8. Используя значения диапазонов изменения для этого параметра, представленный выше в таблице, настройте константы таким образом, что бы значение канала изменялось в заданном диапазоне.
9. Настройте необходимое количество каналов как в пунктах 7 и 8. Затем сохраните проект и загрузите его в редакторе представления данных.
10. Создайте экранную форму и разместите на ней аппараты и динамические объекты, которые указаны вами на схеме, нарисованной в пункте 3.
11. Подключите к динамическим объектам созданные вами каналы, как это указано в примере (раздел 3.2.1)
12. В случае возникновения ошибок откройте заново редактор базы каналов и запустите эмуляцию пересчёта базы каналов.
13. В завершение задания распечатайте отчёт из окна «отчёт» (ALT-5).

Задание №2.

1. В разработанном выше проекте определите, какой из параметров можно считать управляемым (регулируемым) параметром.
2. Определите, какой параметр для этого, выбранного параметра, будет являться входным, т.е. регулирующим.
3. Определите, какие из каналов будут оказывать возмущающее воздействие на выбранный регулируемый параметр.
4. Нарисуйте параметрическую схему полученного объекта регулирования.
5. Утвердите вашу схему у преподавателя.
6. Загрузите редактор базы каналов и в узле АРМ создайте новый объект, назвав его «Регулирование».
7. В данном объекте создайте:
 - два канала для динамической модели канала управления, вход модели и выход модели;
 - по два канала для каждого из возмущающих параметров, вход модели и выход модели.
8. Создайте программу для моделирования динамических свойств объекта (раздел 3.2.2, п. 3-5).
9. Определитесь с динамическими свойствами каждого из каналов нарисованного в пункте 4 объекта. Возмущающие каналы должны обладать более инерционными свойствами чем канал управления.
Запишите все эти передаточные функции в тетрадь с отчётом.
10. Подключите к каждому из каналов, отвечающему за выход динамической модели программу, моделирующую динамические свойства объекта. Константы со свойствами каналов возьмите из записанных ранее передаточных функций.
11. Создайте программу, суммирующую столько параметров, сколько вами было создано каналов для выходов динамической модели.
12. Подключите эту программу к каналу, созданному для регулируемого параметра, таким образом, что бы **реальное** значение этого параметра было равно сумме выходов всех созданных вами моделей.
13. Сохраните проект и загрузите его в редакторе представления данных. Создайте для каждого из каналов, созданных для входа динамических моделей, по одному «динамическому тексту» таким образом, что бы можно было бы вводить их значения с клавиатуры (раздел 3.2.2, п.17).
14. Для регулируемого параметра создайте «тренд», который будет показывать изменение его значения во времени.
15. Запустите и протестируйте проект, вводя значения каналов созданных для входа динамических моделей. Покажите его преподавателю.

16. Теперь закройте редактор представления данных и загрузите ваш проект в редакторе базы каналов.
17. Создайте в созданном ранее объекте «Регулирование» каналы, необходимые для настройки системы регулирования:
 - задание регулируемой переменной;
 - коэффициент передачи регулятора;
 - коэффициент интегрирования регулятора;
 - коэффициент дифференцирования регулятора.
18. Создайте программу регулятора представленную на рисунке:



Настройте входы и выходы блоков программы в соответствии с таблицей 4.

Таблица 4 – Настройка входов и выходов программы PID-регулятора

Блок	Вход, выход	Тип	Комментарий
(-)	AD	аргумент	Вход регулятора
	AS	аргумент	Задание регулятора
DZONE	DLT	константа	Зона нечувствительности
PID	KP	аргумент	Коэф-т пропор-ной составляющей
	KD	аргумент	Коэф-т пропор-ной составляющей
	KI	аргумент	Коэф-т интегральной составляющей
LIMIT	MIN	константа	Минимум управления
	MAX	константа	Максимум управления
	Q	аргумент	Выход регулятора

19. Подключите созданную программу в качестве управляющей процедуры к каналу, созданному для регулирования (см. пункт 12) . В качестве выходного аргумента программы, т.е. выход блока LIMIT подключите к **входному значению** канала, являющегося входом динамической модели (см. пункт 7).
20. Запустите режим эмуляции и проверьте работу вашей замкнутой системы. Значения регулятора определите с помощью системы IPC-CAD, но пересчитайте коэффициенты с учётом передаточных функций регулятора в ТРЕЙС МОУДе и в IPC-CAD.
21. Сохраните проект и загрузите его в редакторе представления данных.
22. Создайте отдельную экранную форму и оформите супервизорное управление системой регулирования.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие языки программирования используются в Трейс Моуд 5.0 ?
Охарактеризуйте каждый из них.
2. Перечислите и охарактеризуйте режимы работы окна редактирования FBD-программ.
3. Что такое функциональный блок?
4. Перечислите и расшифруйте атрибуты функционального блока.
5. Каких типов могут быть входы и выходы функционального блока?
6. Каким образом может быть вызвана FBD-программа в Трейс Моуд 5.0?

6. СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА

Отчет оформляется в ученической или в общей тетради в соответствии с общими требованиями к оформлению текстовых документов. Отчет должен содержать:

- 1) наименование и цель работы;
- 2) схема технологического процесса и краткое его описание;
- 3) схема расположения аппаратов, динамических объектов отображения и ввода информации, которые предполагается располагать на экранной форме;
- 4) таблица с перечнем всех технологических параметров, которые будут отображаться на создаваемой экранной форме. В этой таблице помимо названия параметров обязательно указать краткое обозначение, среднее значение и диапазон изменения для каждого из параметров;
- 5) документированное оформление проекта по выполнению первого из окна просмотра, вызываемого комбинацией клавиш <Alt + 5>
- 6) параметрическая схему объекта регулирования с пояснением всех параметров на ней;
- 7) передаточные функции для всех каналов объекта регулирования (управляющего и возмущающих);
- 8) документированное оформление проекта, вызываемого комбинацией клавиш <Alt + 5>
- 9) Структурная схема САР с указанием всех каналов, созданных для её реализации. Так же представьте список всех этих каналов в табличной форме.

ЛАБОРАТОРНАЯ РАБОТА № 5

"РАЗРАБОТКА РАСПРЕДЕЛЁННОЙ АСУ ТП ДЛЯ УПРАВЛЕНИЯ СЛОЖНЫМ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ"

1. ВВЕДЕНИЕ

В обычных SCADA системах разработка проекта привязана к одной операторской станции. Поэтому при разработке сетевых комплексов сначала создаются БД РВ для отдельных ПК, и лишь потом они объединяются в сеть. Однако современные промышленные АСУ живут и развиваются десятки лет, имеют тенденцию к интеграции как между собой, так и с АСУ финансово-хозяйственных служб. За это время меняется технология, добавляются и заменяются датчики, АРМ, вводятся новые функции и т.д. Вместе с этим, неизбежно, развивается и модифицируется ПО АСУ. Поддерживать и развивать системы, состоящие из многих обособленных ПК и контроллеров, каждый из которых ничего не знает о других, и трудно и дорого.

TRACE MODE 5 имеет распределенную БД РВ, поэтому распределенная АСУ, включающая в себя несколько ПК и контроллеров, рассматривается системой как единый проект. Каждый узел (ПК или контроллер) в распределенной АСУТП, работающей под управлением TRACE MODE 5, имеет информацию об остальных узлах системы, и в случае его модификации автоматически обновляет соответствующие БД на других узлах. При этом АСУ можно создавать как в архитектуре клиент-сервер, так и в виде распределенной системы управления (DCS) - технология разработки АСУТП как единого проекта будет одинаково эффективна.

2. ЦЕЛЬ РАБОТЫ

В данной работе студенты должны:

- во-первых закрепить навык разработки проекта в среде TRACE MODE.
- во-вторых, используя сетевые протоколы научиться совмещать несколько проектов в один распределённый проект;
- в-четвёртых, опираясь на созданные модели и распределённую структуру проекта, научиться разрабатывать и визуализировать супервизорное управление замкнутыми системами регулирования.

3. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Трейс Моуд позволяет создавать многоуровневые, иерархически организованные, резервированные АСУ ТП и имеет средства для создания распределенных АСУ ТП, включающих в себя до трех уровней в иерархии.

Расширенные возможности языков программирования стандарта МЭК-61131/3 позволяют легко создавать математическую основу пересчёта базы каналов.

3.1 Идеология построения распределенных АСУ ТП в Трейс Моуд

В рамках идеологии построения АСУ ТП в Трейс Моуд можно выделить три уровня иерархии (см. рисунок 32):

- уровень контроллеров - нижний уровень;
- уровень операторских станций - верхний уровень,
- административный уровень.

На **уровне контроллеров** реализуется сбор данных от датчиков и непосредственно от цифровых устройств. Для создания этого уровня предусмотрены мониторы: микроМРВ, микроМРВ Мо-дем+, микроМРВ GSM+. Первый предназначен для запуска в контроллерах, связанных с верхним уровнем по локальной сети или последовательному интерфейсу, второй - для связи по коммутируемым телефонным линиям или радиоканалам, а третий - для связи по GSM-сети. При использовании выделенных телефонных линий или радиоканалов следует применять микроМРВ.

Для **уровня операторских станций** предусмотрены мониторы МРВ, NetLink МРВ и NetLink Light. Они позволяют создавать рабочие станции оперативного управляющего персонала.

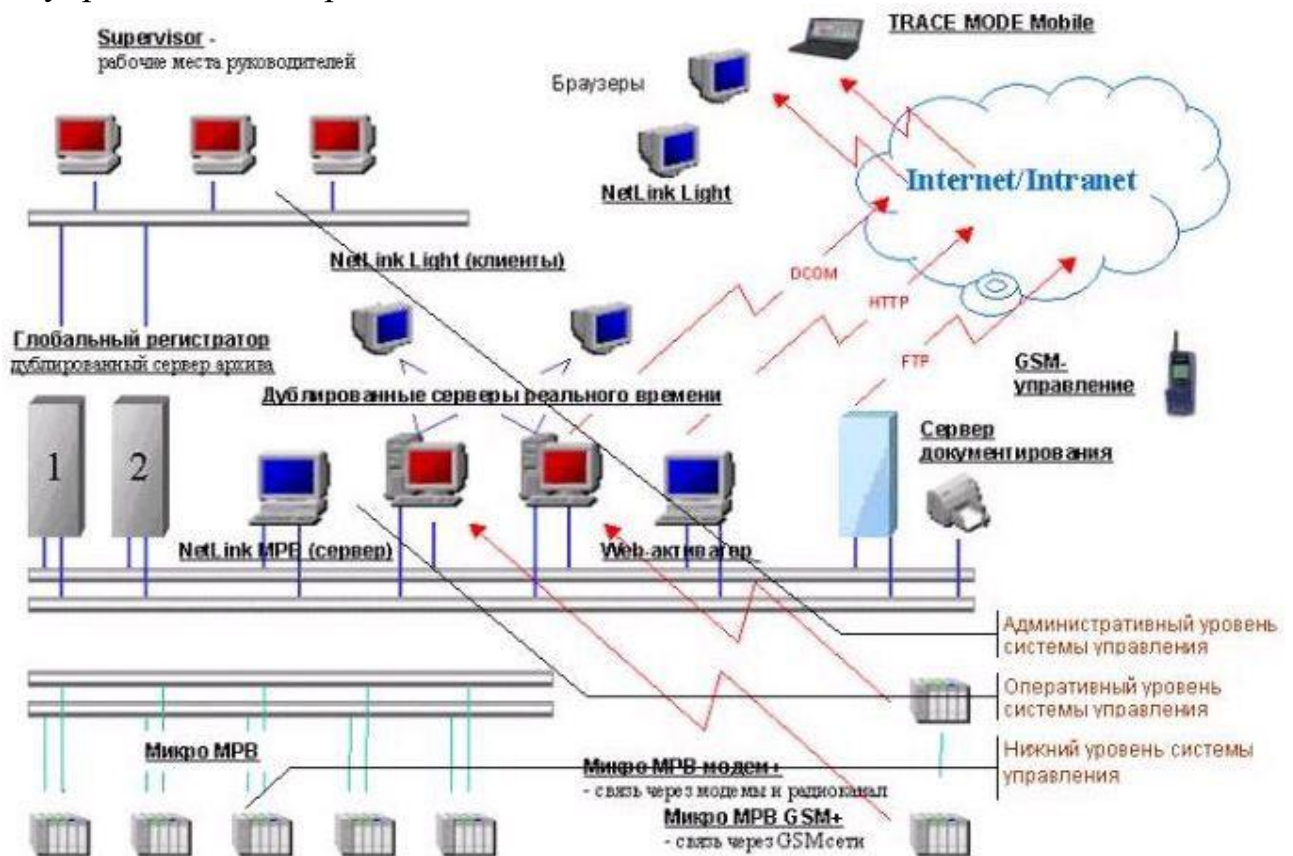


Рисунок 32 – Многоуровневая АСУ ТП на базе Трейс Моуд

Задачей **административного уровня** управления является контроль текущего состояния производственных процессов и анализ функционирования производства по архивным данным. Для решения задач данного уровня предусмотрен монитор SUPERVISOR. Он является специализированной графической кон-

солью, которая может подключаться к серверу математической обработки МРВ, NetLink МРВ или глобальному регистратору. В первых двух случаях просматривается локальный СПАД, а в последнем - глобальный архив. Кроме того, SUPERVISOR можно переключить в режим реального времени. Тогда он работает как консоль NetLink Light и может использоваться для управления процессом.

При работе с архивами SUPERVISOR реализует следующие функции:

- отображение последних изменений значений каналов;
- просмотр архивов в режиме Playback;
- просмотр на заданное архивное время с пошаговым переходом по времени.

МРВ могут **обмениваться данными по следующим линиям:**

- локальная сеть;
- последовательный интерфейс RS-232, RS-485, RS-422;
- радиоканал;
- выделенная телефонная линия; коммутируемые телефонные линии;
- сети GSM.

По этим носителям организовываются информационные потоки всех уровней системы управления. При этом могут реализоваться как вертикальные связи (между уровнями), так и горизонтальные (между узлами одного уровня).

Обмен по всем линиям, кроме локальной сети, осуществляется через последовательный порт по протоколу M-LINK. Для перехода на любую из них нужен соответствующий конвертер или модем.

Узлы в сети M-LINK неравноправны:

- один имеет статус MASTER,
- остальные - SLAVE.

Такие сети следует применять для связи между операторскими станциями и контроллерами.

Для построения **локальной сети** могут использоваться различные адаптеры: Ethernet, Arcnet, Token Ring и др.

Объединенные в локальную сеть операторские станции образуют **оперативный уровень управления**. На нем могут создаваться рабочие места, дублирующие друг друга. Для этого можно использовать различные комбинации мониторов.

Первый вариант - это использование дублированных МРВ. Для них постоянно поддерживается контроль работоспособности. В случае необходимости осуществляется их автоматическое переключение с «горячего» резерва на рабочее состояние.

Второй вариант - тиражирование рабочих мест - предполагает использование на одной операторской станции МРВ, а на остальных - NetLink Light. Связь с контроллерами и архивирование осуществляет МРВ. Остальные станции являются графическими терминалами. Они связываются по сети с базой каналов МРВ.

Для *связи оперативного уровня с административным* необходимо использовать локальную сеть. В этом случае SUPERVISOR запрашивает данные из архивов на удаленном диске.

Если автоматизируемый объект *распределен на большой территории* и нет возможности использовать локальную сеть, следует применять сети на базе протокола M-LINK. В этих случаях можно использовать радиоканал, коммутируемые линии или GSM-сеть.

3.2. Сетевой обмен данными между узлами проекта

Трейс Моуд позволяет создавать крупные распределенные системы. В одном проекте может присутствовать до 200 сетевых серверов. Это могут быть контроллеры, работающие под управлением микроMPB, и операторские станции, работающие под управлением MPB и NetLink MPB. В этой же сети могут присутствовать дублированные глобальные регистраторы, а также неограниченное число архивных станций, работающих под управлением SUPERVISOR, И консольных операторских станций, работающих под управлением NetLink Light.

Все серверы реального времени имеют уникальный сетевой номер, который присваивается узлу при его добавлении в проект.

Сетевой номер задается в диапазоне 1...199. Он является идентификатором узла в рамках проекта и не связан с идентификаторами сетевой ОС. Глобальные регистраторы всегда имеют сетевой номер 200. Что касается SUPERVISOR и NetLink Light, то они являются клиентами. Поэтому их число в сети теоретически не ограничено.

Для обмена данными по сети между мониторами Трейс Моуд рекомендуется использовать сети Windows. Минимальная конфигурация для обмена данными между мониторами, включает в себя: службу «Клиент для сетей Microsoft»; драйвер сетевой платы; сетевой протокол NetBEUI.

Трейс Моуд поддерживает следующие режимы сетевого обмена: **файловый обмен; точка-точка; один ко многим.**

3.2.1. Файловый обмен

Файловый обмен предполагает запись одним или несколькими мониторами данных в файл, который могут считывать другие мониторы. Данный тип обмена используется в следующих случаях:

- при сохранении данных в локальном архиве на диске удаленного компьютера или при чтении данных из этого файла;
- при сохранении отчета тревог на диске удаленного компьютера или при просмотре этого файла с помощью специальных форм отображения или утилит;
- при сохранении состояния системы на удаленном диске или при считывании файла состояния при запуске системы.

3.2.2. Обмен «точка-точка»

Данный обмен данными по сети предназначен для передачи оперативной информации от одного монитора к другому.

Режим обмена «точка-точка» обеспечивает одиночную связь значения атрибута канала одного узла (пассивного) с входом или выходом канала другого (активного).

В этом обмене один узел является пассивным, а второй - активным. Направление передачи информации определяется типом канала активного узла. Если он имеет тип INPUT, то данные будут запрашиваться у пассивного узла, а если OUTPUT - передаваться.

Для связи в режиме "точка-точка" на активном узле используются каналы подтипа СВЯЗЬ (Рисунок 24). Для приема данных значение дополнения к подтипу канала следует указать IN NET, для передачи - OUT NET или OUT MAP-NET. Пассивный узел при этом не должен иметь каких-либо специальных каналов. Активный узел может обратиться к любому атрибуту любого канала пассивного узла.

Настройка канала NN определяет номер сетевого адаптера. Если она равна нулю, то используется основной сетевой адаптер, в противном случае - второй. При нажатии ЛК на любую другую настройку на экран выводится диалог «Выбор канала» (рисунок 25). В нем следует указать опрашиваемый или управляемый канал удаленного узла.

Существует стандартный объект базы каналов, в который автоматически заносятся все каналы подтипа «СВЯЗЬ» с описанными дополнениями к подтипу. Он имеет имя «СЕТЕВОЙ ВВОД/ВЫВОД».

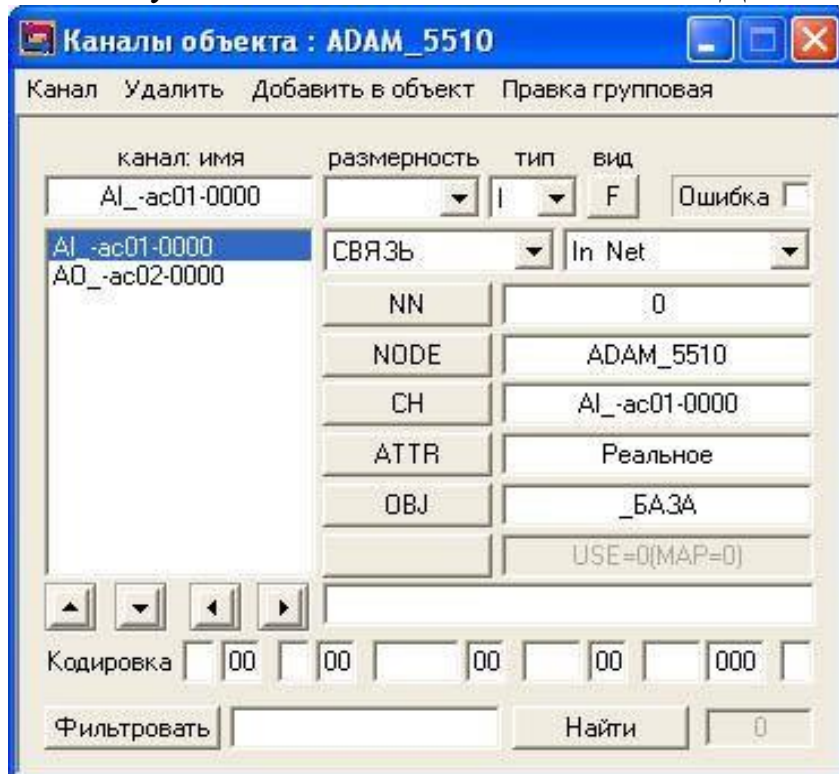


Рисунок 33 – Настройки канала типа «Связь»

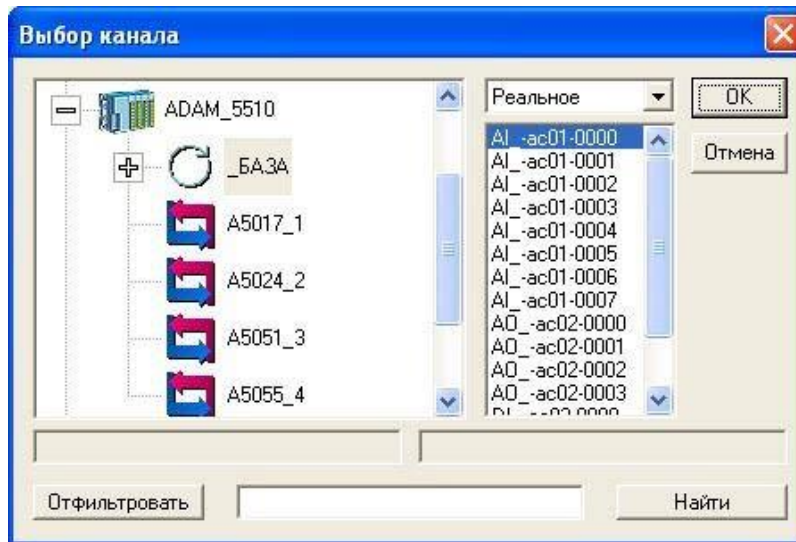


Рисунок 34 – Окно для выбора канала при связи «точка-точка»

Для связи мониторов в режиме «точка-точка» можно воспользоваться также стандартным механизмом OPC. Это менее скоростной протокол, но он дает больше гарантий доставки данных. Для запроса данных по OPC используются каналы подтипа «СВЯЗЬ» с дополнением IN OPC, а для управления - OUT OPC. Кроме того, при такой связи для каждого узла следует указать сетевое имя или IP-адрес компьютера, на котором он будет запускаться. Это делается в бланке «Основные» диалога «Параметры узла».

3.2.3. Обмен «один ко многим»

В этом режиме при изменении реального значения канала оно посылается в сеть. Его могут принимать несколько других узлов, присутствующих в сети. Такой режим используется при групповых рассылках, групповом управлении и передаче данных в резервированные глобальные регистраторы.

Для группового управления используются каналы подтипа «СВЯЗЬ» с дополнением OUT Net(group). Они настраиваются так же, как для управления отдельным узлом: указывается удаленный узел, канал и атрибут. Однако их выходное значение посылается не на индивидуальный сетевой номер указанного узла, а на групповой. Все узлы с таким же групповым номером принимают эти послыки и присваивают полученное значение каналам с теми же номерами, что и каналы, указанные в настройках. Групповые сетевые номера задаются для узлов в редакторе базы каналов. Для этого используется раздел «Сетевой адрес» бланка «Основные» в диалоге «Параметры узла».

Групповые рассылки используются, если одни и те же данные должны получать несколько узлов. Например, информация из контроллеров передается нескольким операторским станциям. Для этого опрашиваемые каналы надо объявить **сетевыми переменными**. Максимальное число таких рассылок в одном проекте равно 32 000.

Чтобы значение канала стало сетевой рассылкой, для него надо установить флаг «В сеть» в бланке «Основные» диалога «Реквизиты». При этом каналу присваивается номер сетевой рассылки. Он выводится рядом с полем установки флага (Рисунок 26). *При каждом изменении реального значения этого канала оно посылается в сеть.*

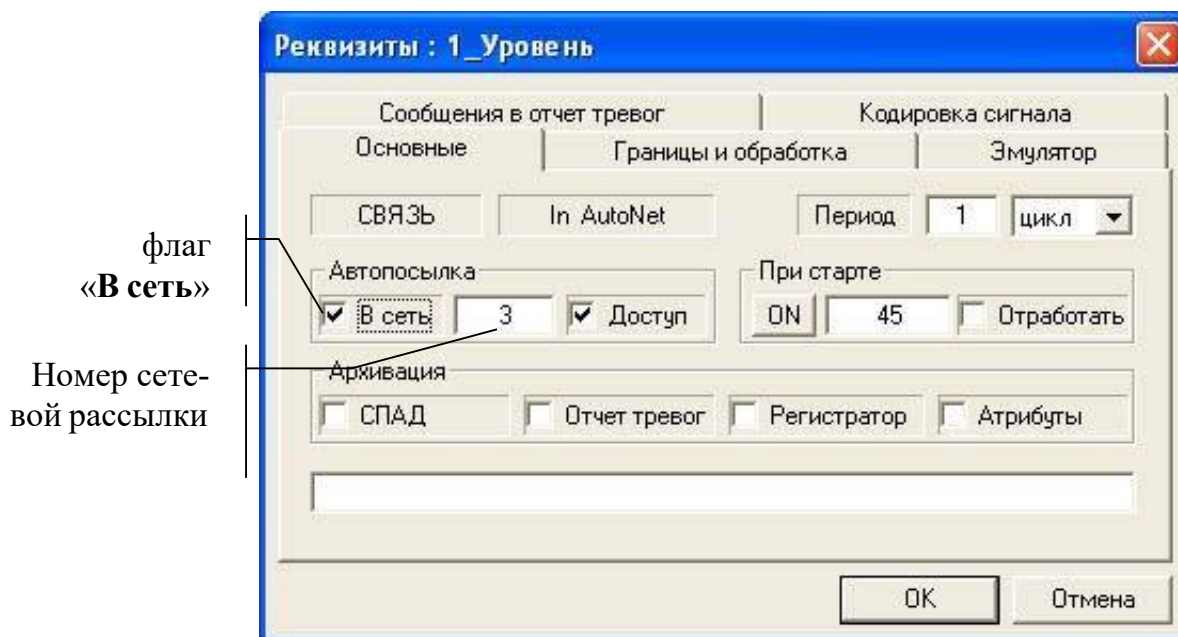


Рисунок 35 - Задание сетевых номеров при групповом управлении

Для его приема на другом узле надо создать канал, принимающий рассылку. Этот канал имеет подтип «СВЯЗЬ» и дополнение IN AutoNET. В качестве настроек для него устанавливается узел и канал на нем, посылающий данные в сеть (Рисунок 36).



Рисунок 36 – Канал, принимающий авторассылки

Чтобы автопосылка в узле-приемнике принималась в каналы, подтип или дополнение которых не совпадают с подтипом «СВЯЗЬ» или **In AutoNet**, надо создать в узле-приемнике канал подтипа «СВЯЗЬ» с дополнением In_Net с любым каналом узла-источника (этот канал можно даже выключить).

Профайлер принимает от себя автопосылки (без условий) и данные, передаваемые с помощью канала Out MapNet (если групповые номера узла-источника и узла-приемника совпадают). Прием можно отменить соответствующим битом канала «Сеть,DDE» подтипа «СИСТЕМНЫЙ».

Все присутствующие в проекте сетевые рассылки доступны для просмотра в стандартном объекте «ПРИЕМ РАССЫЛОК» любого узла (рисунок 37). Напротив сетевых рассылок, принимаемых на текущем узле, выводятся имена каналов-приемников. Каналы подтипа «СВЯЗЬ» с дополнением к подтипу IN AutoNET автоматически попадают в этот объект. Любой канал типа INPUT может принимать сетевые рассылки. Принимаемые им значения копируются в его атрибут «Выход». Если канал отключить от источника, то значение выхода копируется в реальное значение.

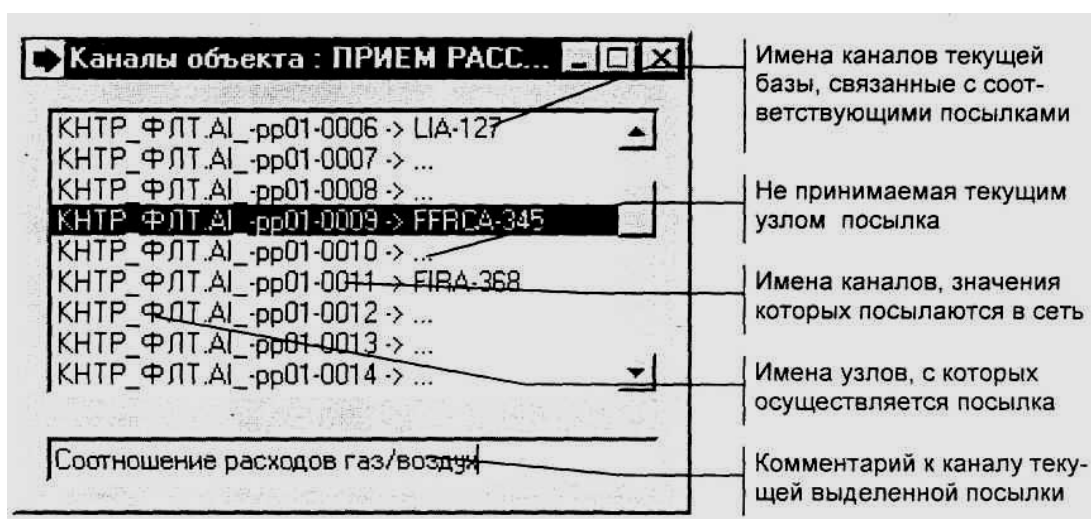


Рисунок 37 - Обзор сетевых рассылок

Чтобы связать канал текущей базы с данными, посылаемыми по сетевой рассылке, надо выбрать требуемую рассылку в списке. При этом на экран выводится диалог **выбора канала** (см. рисунок 25). Имя указанного в этом диалоге канала вносится в правую часть списка сетевых рассылок. Для отказа от приема данных надо выделить в списке требуемую рассылку и нажать ЛК на кнопку «Разъединить».

Чтобы МРВ поддерживал обмен данными по сети, для него надо установить флаги **HOST MODE** и **SLAVE MODE**. Первый из них необходим для формирования запросов по сети, а второй - для ответа текущего монитора на формируемые к нему запросы. Эти флаги устанавливаются в бланке «Основные» диалога «Параметры» узла (Рисунок 38).

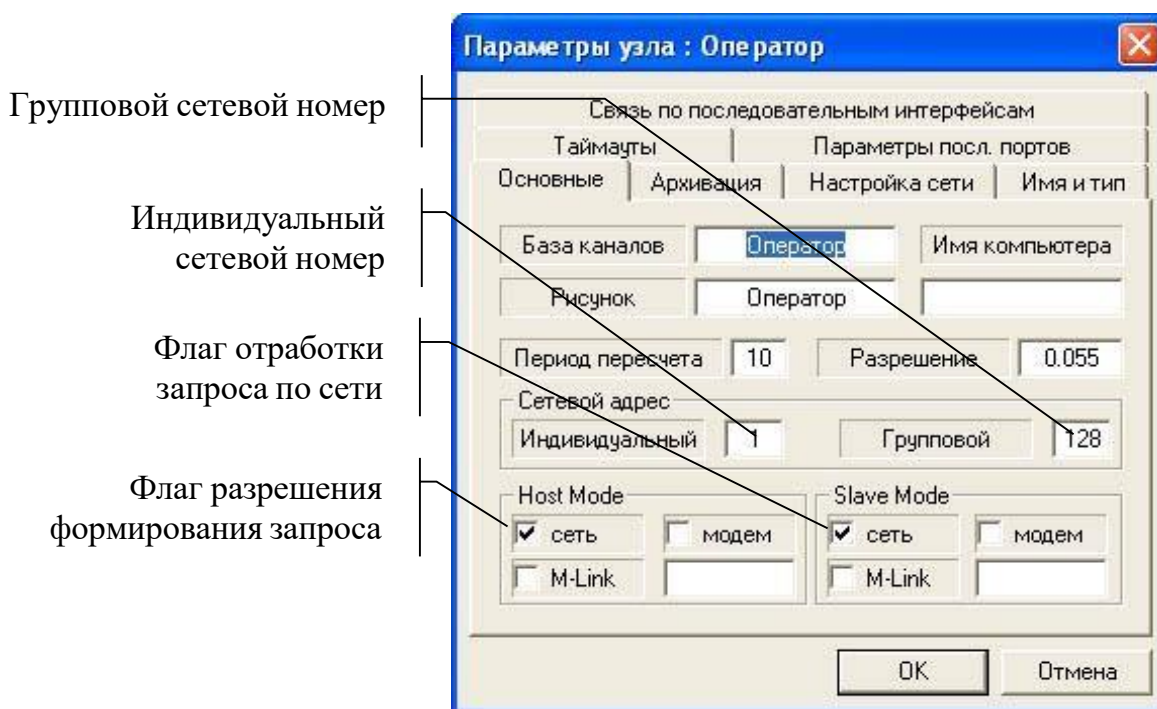


Рисунок 38 – Настройка параметров сетевого обмена

Основными параметрами сетевого обмена каждого узла проекта являются его **индивидуальный** и **групповой сетевые номера**. Сетевой номер присваивается узлу автоматически при его создании. Он используется при обращении к этому узлу другими узлами или внешними приложениями. Групповой номер применяется при групповом приеме данных и управлении. Дублирующие друг друга узлы должны принимать и обрабатывать одни и те же команды управления. Для этого им необходимо установить одинаковый групповой номер.

Для контроля сетевого обмена в Трейс Моуд предусмотрены следующие каналы:

- подтип «ДИАГНОСТИКА» с дополнением «Сеть»;
- подтип «ДИАГНОСТИКА» с дополнением «Код сети»;
- подтип «ДИАГНОСТИКА» с дополнением «Свб. NCB»;
- подтип «ДИАГНОСТИКА» с дополнением «NCB (прием)»;
- подтип «ДИАГНОСТИКА» с дополнением «NCB (отсылка)».

Значение канала «ДИАГНОСТИКА» с дополнением «Сеть» определяет следующие ситуации:

- никаких действий не выполняется;
- запуск операции;
- выполнение операции;
- ожидание;
- завершение операции;
- ошибка выполнения операции;
- нормальное завершение операции.

Канал подтипа «ДИАГНОСТИКА» с дополнением «Код сети» позволяет контролировать сообщения сетевой ОС. В него копируется байт статуса сетевого обмена.

3.2.4. Создание и настройка связи с каналами OPC-сервера

Протокол OPC сравнительно медленный по сравнению с представленными выше способами связи, но он зарекомендовал себя как наиболее надёжный, поэтому стоит на нём остановиться подробнее.

Каналы для связи с OPC-сервером создаются процедурой автопостроения. Чтобы запустить ее, следует, находясь в окне объектов настраиваемого узла, выполнить команду **Связать с OPC-сервером** из меню **Узел** или нажать сочетание клавиш **Alt+L**. При этом на экране появится диалог, показанный на рисунке 39.



Рисунок 39 – Окно выбора OPC-сервера

Здесь выводится список используемых в проекте серверов. Чтобы редактировать этот список, предусмотрены кнопки: **Добавить**, **Удалить** и **Изменить**. По нажатию первой из них на экран выводится диалог выбора сервера. По нажатию кнопки **Удалить** выделенный в списке сервер удаляется. По нажатию кнопки **Изменить** на экране появляется диалог выбора сервера. Однако в этом случае выбранный сервер не добавляется в список, а заменяет текущий.

Чтобы создать каналы для обмена с любым сервером из списка, надо выделить его и нажать ЛК на кнопке **Выбрать** (рисунок 40).

В левом окне выводятся каналы OPC-сервера. Чтобы установить связь с любым из них надо выделить его и нажатием ЛК на соответствующей кнопке переместить его в правое окно. После выхода из этого диалога в окне базы каналов появится новый объект, имя которого совпадает с идентификатором OPC-сервера. В нем создаются каналы для обмена данными с выбранными каналами сервера.

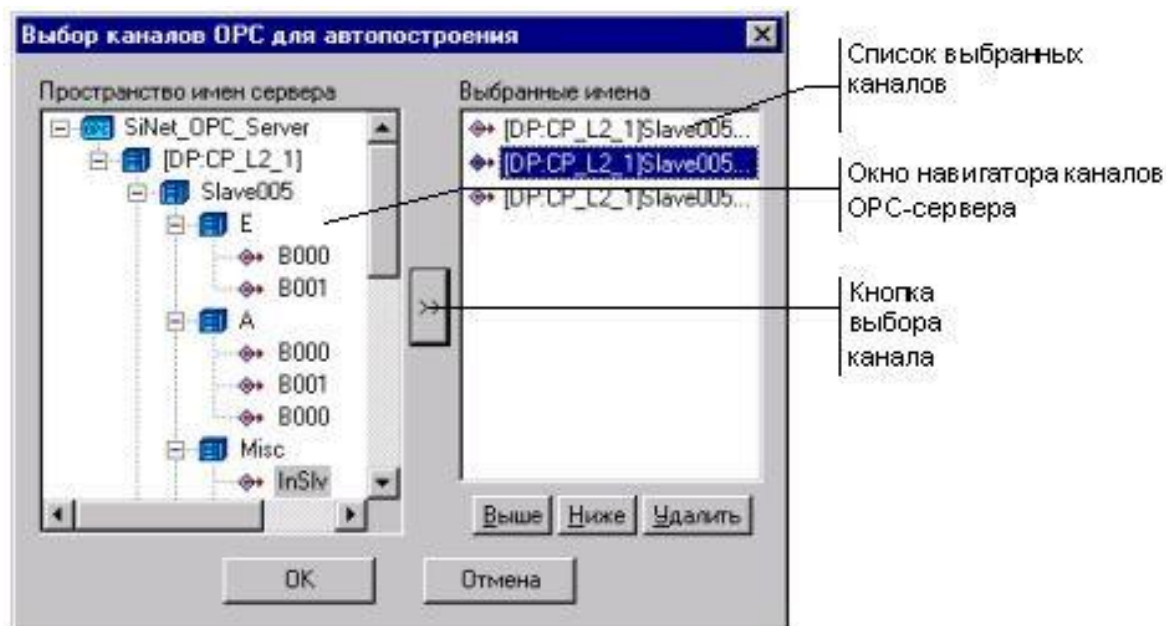


Рисунок 40 – Окно выбора каналов OPC-сервера для автопостроения

Каналы связи с OPC-сервером имеют три настройки (рисунок 41).

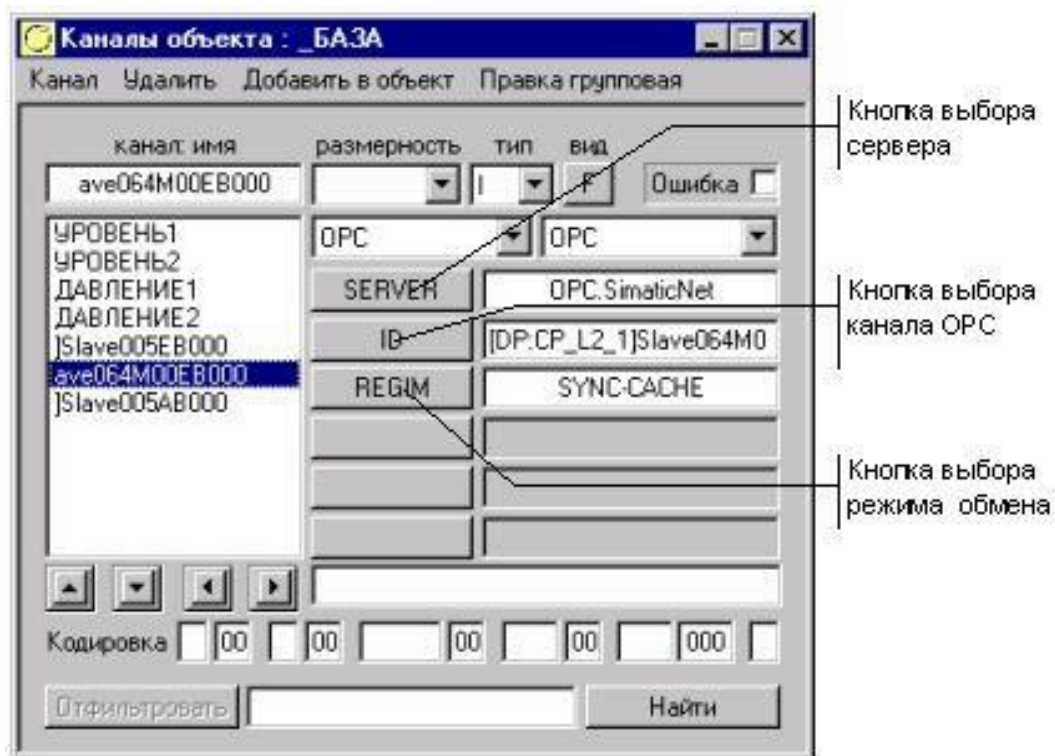


Рисунок 41 – Окно канала с настройками OPC-сервера

Для выбора режима обмена надо нажать ЛК на кнопке **MODE**. При этом на экране появляется диалог для выбора режима.

Кнопка **SERVER** выбрать OPC-сервер. Кнопка **ID** служит для выбора канала OPC-сервера. При ее нажатии на экран выводится диалог показанный на рисунке 42.

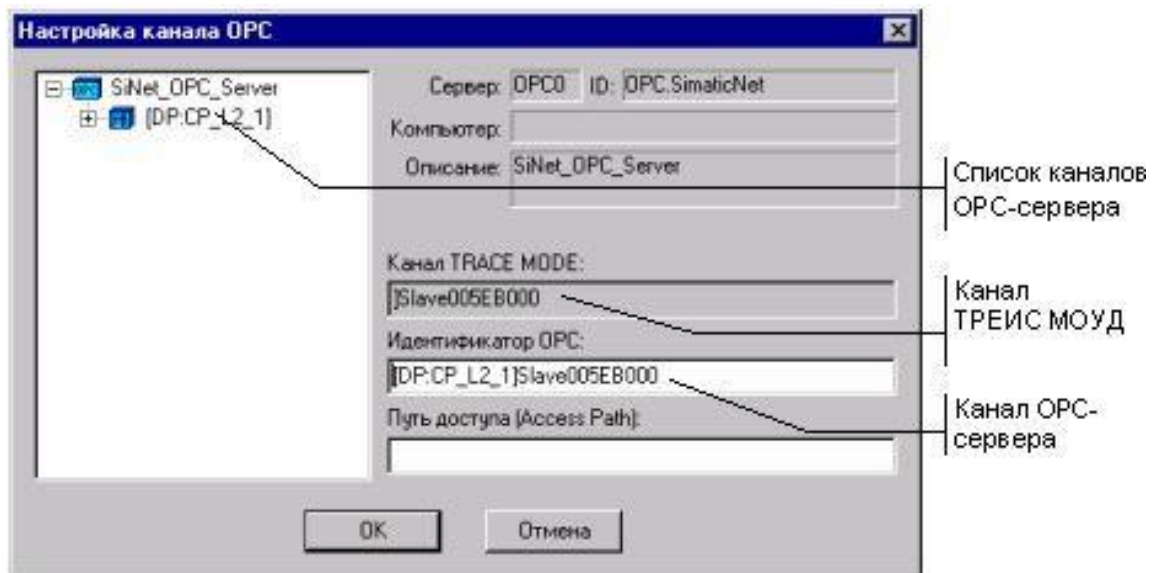


Рисунок 42 – Окно выбора канала с конкретного OPC-сервера

В поле **Идентификатор OPC** задается канал OPC-сервера для связи с редактируемым каналом ТРЕЙС МОУД. Его можно ввести с клавиатуры или выбрать с помощью навигатора в левом окне.

Канал подтипа OPC может быть типа INPUT или типа OUTPUT. В первом случае он будет запрашивать информацию с OPC-сервера, а во втором случае изменять значение канала на OPC-сервере

4. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Для выполнения данной работы необходимо завершить создание проекта процесса, выданного в лабораторной работе №4 и объединиться с несколькими товарищами.

На базе одного из проектов, создайте один большой проект., включающий в себя несколько узлов, на каждом из которых реализуются какие либо функции. Один из этих узлов рекомендуется сделать сервером, моделирующим поведение реального объекта, как это описано в двух предыдущих заданиях.

Для этой цели обозначим тот проект, к которому добавляем каналы - *первым*, а тот из которого данные добавляются – *вторым*. Объединение проектов производите в следующей последовательности:

1. Сохраните все объекты созданные вами в базе каналов из второго проекта в отдельные файлы.
2. Скопируйте эти созданные файлы с компьютера, где расположен второй объект на компьютер, где расположен первый объект.
3. Загрузите первый проект в редакторе базы каналов и откройте новые объекты из файлов.
4. Таким же образом можно скопировать все недостающие программы из второго проекта в первый.
5. Для загруженных каналов необходимо заново осуществить подключение всех программ.
6. Используя библиотеку объектов все графические объекты из второго проекта можно скопировать в первый проект.
7. Создайте в редакторе базы каналов ещё один узел и настройте в нём сетевой обмен <точка в точку>.
8. Для каждого из узлов создайте в редакторе представления данных свои экранные формы. На сервере желательно побольше графиков, а на АРМ-оператора побольше визуализации аппаратов.
9. Когда оба узла будут готовы, то скопируйте проект на другой компьютер в такую же точно директорию, в какой он находится на первом компьютере.
10. Запустите в профайлере на одном компьютере этот проект для первого узла, с моделями, а на втором компьютере этот же проект, но со вторым узлом.
11. Оттестируйте сетевой обмен между узлами одного распределённого проекта.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Уровни иерархии построения АСУ ТП в Трейс Моуд 5.0, назначение каждого из них.
2. Какие линии обмена данными используются в Трейс Моуд 5.0.
3. Назначение и область применения протокола M-LINK.
4. Что представляет из себя оперативный уровень управления?
5. Какое назначение локальной сети в Трейс Моуд 5.0?
6. Перечислите и охарактеризуйте режимы сетевого обмена.
7. Для чего предназначен файловый обмен?
8. Для чего предназначен обмен «точка-точка»?
9. Каковы особенности обмена «один ко многим»?
10. Какие языки программирования используются в Трайс Моуд 5.0 ?

6. СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА

Отчет оформляется в ученической или в общей тетради в соответствии с общими требованиями к оформлению текстовых документов. Отчет должен содержать:

- 1) наименование и цель работы;
- 2) блок схема распределённого проекта АСУ ТП, с пояснением назначения каждого узла и каждой экранной формы.
- 3) документированное оформление проекта для каждого из узлов, вызываемого комбинацией клавиш <Alt + 5>
- 4) таблица основных каналов узла, являющегося серверов с указанием всех программ, подключаемых к каждому каналу.
- 5) таблица основных каналов узла, являющегося клиентом (АРМ оператора).

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Деменков Н.П. SCADA-системы как инструмент проектирования АСУ ТП: Учеб.пособие. – М.: изд-во МГТУ им. Н.Э. Баумана, 2004. – 328с.:ил.
2. Аристова Н.И., Корнеева А.И. Промышленные программно-аппаратные средства на отечественном рынке АСУ ТП. М.:Научтехиздат, 200. – 399 с.
3. Матвейкин В.Г., Фролов С.В., Шахтман М.Б. Применение SCADA-систем при автоматизации технологических процессов: Учеб. Пособие. В.; Тамбов: Машиностроение, 200. – 176 с.
4. Анзимилов Л. В. Интегрированная SCADA и Softlogic система TRACE MODE 5.0 в 2002 году. Ж. Приборы и системы. Управление, контроль, диагностика. 2002. №1, с. 7-13.
5. Руководство пользователя Трейс Моуд. Версия 5.0. М.: AdAstra Research Group, Ltd. 2000. 814 с.
6. Локотков А. Что должна уметь система SCADA. Ж. Современные технологии автоматизации. 1998. №3, с. 44-46.
7. Соболев О.С Прогресс в области SCADA-систем и проблемы пользователей // Мир компьютерной автоматизации. 1999. №3. С. 20-24.
8. Деменков Н.П. SCADA-системы как инструмент проектирования АСУ ТП // Приложение к журналу «Информационные технологии». 2002. №11. 24 с.
9. Ицкович Э.Л., Соловьев Ю.А., Мурзенко И.В. Опыт использования открытых SCADA-программ // Промышленные АСУ и контроллеры. 1999. № 11. С. 36-38
10. Куцевич К А. SCADA-системы и муки выбора // Мир компьютерной автоматизации. 1999. № 1. С. 72-78.
11. Качадин А.Ю. Выбор SCADA-системы: надежность или простота? // Промышленные АСУ и контроллеры. 2001. № 3. С. 50-52.
12. SCADA-системы: проблемы выбора / В. Бунин, В. Анопренко, А. Ильин, О. Салова, Н. Чибисова, А. Якушев // Современные технологии автоматизации. 1999. № 4. С. 6-24.
13. Куцевич И.А. SCADA-системы. Взгляд со стороны // Промышленные АСУ и контроллеры. 1999. № 4. С. 22-28.
14. Потапова Т.Б. Уроки выбора SCADA-программ // Промышленные АСУ и контроллеры. 2001. № 1. С. 41-42.
15. Соловьев Ю.А. Уроки выбора SCADA-программ (продолжение темы) // Промышленные АСУ и контроллеры. 2001. № 5. С. 43.
16. Деменков Н.П. Проблемы сравнительного анализа SCADA-систем // Промышленные АСУ и контроллеры. 2001. № 4. С. 43-46.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
для по организации и проведению самостоятельной работы
по дисциплине «Устройства СВЧ»
для студентов направления подготовки
11.03.04 Электроника и нанoeлектроника
Направленность (профиль):
Интеллектуальные программируемые системы и устройства

Ставрополь
2026

Содержание

Введение

Самостоятельная работа как важнейшая форма учебного процесса

Цели и основные задачи СРС

Виды самостоятельной работы

Организация СРС

Общие рекомендации по организации самостоятельной работы

Самостоятельная работа студента - необходимое звено становления исследователя

Методические рекомендации для студентов по отдельным формам самостоятельной работы

Учебно-методическое и информационное обеспечение дисциплины

ВВЕДЕНИЕ

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины является формирование набора профессиональных компетенций будущего бакалавра по направлению подготовки 11.03.04 Электроника и наноэлектроника, путем ознакомления с теоретическими основами передачи сигналов СВЧ, параметрами и характеристиками пассивных и активных компонентов СВЧ устройств

Задачи освоения дисциплины:

- ознакомление с теоретическими основами передачи сигналов СВЧ;
- изучение параметров и характеристик пассивных компонентов СВЧ устройств;
- изучение параметров и характеристик активных компонентов СВЧ устройств.

2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесённых с планируемыми результатами освоения образовательной программы

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
ПК-5 Способен к монтажу и наладке технических средств автоматизации технологических процессов производства электронных приборов, блоков и систем различного функционального назначения	ИД-1 _{ПК-5} Использует методы наладки измерительного, диагностического и технологического оборудования, используемого в области электроники и наноэлектроники	Использует методы наладки и диагностики параметров компонентов, элементов и узлов СВЧ устройств
ПК-6 Способен к сервисному обслуживанию измерительного, диагностического и технологического оборудования, к управлению программно-аппаратными средствами информационных служб инфокоммуникационной системы предприятия	ИД-3 _{ПК-6} Осуществляет мониторинг диагностического, технологического оборудования	Демонстрирует знание правил мониторинга и технического обслуживания компонентов, элементов и узлов СВЧ устройств

САМОСТОЯТЕЛЬНАЯ РАБОТА КАК ВАЖНЕЙШАЯ ФОРМА УЧЕБНОГО ПРОЦЕССА

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения. Обучение в ВУЗе включает в себя две, практически одинаковые по объему и взаимовлиянию части – процесса обучения и

процесса самообучения. Поэтому СРС должна стать эффективной и целенаправленной работой студента.

Концепцией модернизации российского образования определены основные задачи профессионального образования - "подготовка квалифицированного работника соответствующего уровня и профиля, конкурентоспособного на рынке труда, компетентного, ответственного, свободно владеющего своей профессией и ориентированного в смежных областях деятельности, способного к эффективной работе по специальности на уровне мировых стандартов, готового к постоянному профессиональному росту, социальной и профессиональной мобильности".

Решение этих задач невозможно без повышения роли самостоятельной работы студентов над учебным материалом, усиления ответственности преподавателей за развитие навыков самостоятельной работы, за стимулирование профессионального роста студентов, воспитание творческой активности и инициативы.

К современному специалисту общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных способностей и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной финансовой ситуации. Формирование такого умения происходит в течение всего периода обучения через участие студентов в лабораторных занятиях, выполнение контрольных заданий и тестов, написание курсовых и выпускных квалификационных работ. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

Формы самостоятельной работы студентов разнообразны. По данной дисциплине «Основы нейронных сетей» предусмотрены следующие:

Использование материала учебно-методического комплекса дисциплины

На первом этапе необходимо ознакомиться с обязательным минимумом содержания основной образовательной программы по изучаемой дисциплине. Далее необходимо изучить рабочую программу дисциплины, в которой рассмотрено содержание тем дисциплины лекционного курса, взаимосвязь тем лекций с лабораторными занятиями, темы и виды самостоятельной работы. По каждому виду самостоятельной работы предусмотрены определённые формы отчетности, представленные в учебной программе по дисциплине «Основы нейронных сетей».

Работа с литературой

Для успешного освоения дисциплины, необходимо самостоятельно детально изучить представленные темы по рекомендуемым источникам информации, которые представлены в учебной программе.

Самостоятельная работа приобщает студентов к научному творчеству, поиску и решению актуальных современных проблем.

ЦЕЛИ И ОСНОВНЫЕ ЗАДАЧИ СРС

Ведущая цель организации и осуществления СРС должна совпадать с целью обучения студента – подготовкой бакалавра с высшим образованием. При организации СРС важным и необходимым условием становятся формирование умения

самостоятельной работы для приобретения знаний, навыков и возможности организации учебной и научной деятельности.

Целью самостоятельной работы студентов является овладение фундаментальными знаниями, профессиональными умениями и навыками деятельности по профилю, опытом творческой, исследовательской деятельности. Самостоятельная работа студентов способствует развитию самостоятельности, ответственности и организованности, творческого подхода к решению проблем учебного и профессионального уровня.

Задачами СРС являются:

- систематизация и закрепление полученных теоретических знаний и практических умений студентов;
- углубление и расширение теоретических знаний;
- формирование умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развитие познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развитие исследовательских умений;
- использование материала, собранного и полученного в ходе самостоятельных занятий на семинарах, на лабораторных занятиях, при написании курсовых и выпускной квалификационной работ, для эффективной подготовки к итоговым зачетам и экзаменам.

ВИДЫ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

В образовательном процессе высшего профессионального образовательного учреждения выделяется два вида самостоятельной работы – аудиторная, под руководством преподавателя, и внеаудиторная. Тесная взаимосвязь этих видов работ предусматривает дифференциацию и эффективность результатов ее выполнения и зависит от организации, содержания, логики учебного процесса (межпредметных связей, перспективных знаний и др.):

Аудиторная самостоятельная работа по дисциплине выполняется на учебных занятиях под непосредственным руководством преподавателя и по его заданию.

Внеаудиторная самостоятельная работа выполняется студентом по заданию преподавателя, но без его непосредственного участия.

Основными видами самостоятельной работы студентов без участия преподавателя по данной дисциплине являются:

- формирование и усвоение содержания конспекта лекций на базе рекомендованной лектором учебной литературы, включая информационные образовательные ресурсы (электронные учебники, электронные библиотеки и др.);
- подготовка к лабораторным работам, их оформление.

Основными видами самостоятельной работы студентов с участием преподавателей являются:

- текущие консультации;
- прием и защита лабораторных работ (во время проведения л/р).

ОРГАНИЗАЦИЯ СРС

Методика организации самостоятельной работы студентов зависит от структуры, характера и особенностей дисциплины «Основы нейронных сетей», объема часов на ее изучение, вида заданий для самостоятельной работы студентов, индивидуальных качеств студентов и условий учебной деятельности.

Процесс организации самостоятельной работы студентов включает в себя следующие этапы:

- подготовительный (определение целей, составление программы, подготовка методического обеспечения, подготовка оборудования);
- основной (реализация программы, использование приемов поиска информации, усвоения, переработки, применения, передачи знаний, фиксирование результатов, самоорганизация процесса работы);
- заключительный (оценка значимости и анализ результатов, их систематизация, оценка эффективности программы и приемов работы, выводы о направлениях оптимизации труда).

Деятельность студентов по формированию и развитию навыков учебной самостоятельной работы

В процессе самостоятельной работы студент приобретает навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Выполняя самостоятельную работу под контролем преподавателя студент должен:

- освоить минимум содержания, выносимый на самостоятельную работу студентов и предложенный преподавателем в соответствии с образовательными стандартами высшего образования;
- планировать самостоятельную работу в соответствии с графиком самостоятельной работы, предложенным преподавателем;
- самостоятельную работу студент должен осуществлять в организационных формах, предусмотренных учебным планом и рабочей программой преподавателя;
- выполнять самостоятельную работу и отчитываться по ее результатам в соответствии с графиком представления результатов, видами и сроками отчетности по самостоятельной работе студентов.

студент может:

сверх предложенного преподавателем (при обосновании и согласовании с ним) и минимума обязательного содержания, определяемого программой по данной дисциплине:

- самостоятельно определять уровень (глубину) проработки содержания материала;
- предлагать темы и вопросы для самостоятельной проработки;
- в рамках общего графика выполнения самостоятельной работы предлагать обоснованный индивидуальный график выполнения и отчетности по результатам самостоятельной работы;
- предлагать свои варианты организационных форм самостоятельной работы;
- использовать для самостоятельной работы методические пособия, учебные пособия, разработки сверх предложенного преподавателем перечня;

– использовать не только контроль, но и самоконтроль результатов самостоятельной работы в соответствии с методами самоконтроля, предложенными преподавателем или выбранными самостоятельно.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего бакалавра, она планируется студентом самостоятельно. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

ОБЩИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Основной формой самостоятельной работы студента является изучение конспекта лекций, их дополнение, рекомендованной литературы, активное участие на лабораторных занятиях. Но для успешной учебной деятельности, ее интенсификации, необходимо учитывать следующие субъективные факторы:

1. Знание школьного программного материала, наличие прочной системы знаний, необходимой для усвоения основных вузовских курсов. Это особенно важно для математических дисциплин. Необходимо отличать пробелы в знаниях, затрудняющие усвоение нового материала, от малых способностей. Затратив силы на преодоление этих пробелов, студент обеспечит себе нормальную успеваемость и поверит в свои способности.

2. Наличие умений, навыков умственного труда:

а) умение конспектировать на лекции и при работе с книгой;
б) владение логическими операциями: сравнение, анализ, синтез, обобщение, определение понятий, правила систематизации и классификации.

3. Специфика познавательных психических процессов: внимание, память, речь, наблюдательность, интеллект и мышление. Слабое развитие каждого из них становится серьезным препятствием в учебе.

4. Хорошая работоспособность, которая обеспечивается нормальным физическим состоянием. Ведь серьезное учение — это большой многосторонний и разнообразный труд. Результат обучения оценивается не количеством сообщаемой информации, а качеством ее усвоения, умением ее использовать и развитием у себя способности к дальнейшему самостоятельному образованию.

5. Соответствие избранной деятельности, профессии индивидуальным способностям. Необходимо выработать у себя умение саморегулировать свое эмоциональное состояние и устранять обстоятельства, нарушающие деловой настрой, мешающие намеченной работе.

6. Овладение оптимальным стилем работы, обеспечивающим успех в деятельности. Чередование труда и пауз в работе, периоды отдыха, индивидуально обоснованная норма продолжительности сна, предпочтение вечерних или утренних занятий, стрессоустойчивость на экзаменах и особенности подготовки к ним,

7. Уровень требований к себе, определяемый сложившейся самооценкой.

Адекватная оценка знаний, достоинств, недостатков - важная составляющая самоорганизации человека, без нее невозможна успешная работа по управлению своим поведением, деятельностью.

Одна из основных особенностей обучения в высшей школе заключается в том, что постоянный внешний контроль заменяется самоконтролем, активная роль в обучении принадлежит уже не столько преподавателю, сколько студенту.

Зная основные методы научной организации умственного труда, можно при наименьших затратах времени, средств и трудовых усилий достичь наилучших результатов.

Эффективность усвоения поступающей информации зависит от работоспособности человека в тот или иной момент его деятельности.

Работоспособность - способность человека к труду с высокой степенью напряженности в течение определенного времени. Различают внутренние и внешние факторы работоспособности.

К внутренним факторам работоспособности относятся интеллектуальные особенности, воля, состояние здоровья.

К внешним:

- организация рабочего места, режим труда и отдыха;
- уровень организации труда - умение получить справку и пользоваться информацией;
- величина умственной нагрузки.

Выдающийся русский физиолог Н. Е. Введенский выделил следующие условия продуктивности умственной деятельности:

- во всякий труд нужно входить постепенно;
- мерность и ритм работы. Разным людям присущ более или менее разный темп работы;
- привычная последовательность и систематичность деятельности;
- правильное чередование труда и отдыха.

Отдых не предполагает обязательного полного бездействия со стороны человека, он может быть достигнут простой переменой дела. В течение дня работоспособность изменяется. Наиболее плодотворным является *утреннее время* (с 8 до 14 часов), причем максимальная работоспособность приходится на период с 10 до 13 часов, затем *послеобеденное* - (с 16 до 19 часов) и *вечернее* (с 20 до 24 часов). Очень трудный для понимания материал лучше изучать в начале каждого отрезка времени (лучше всего утреннего) после хорошего отдыха. Через 1-1,5 часа нужны перерывы по 10 - 15 мин, через 3 - 4 часа работы отдых должен быть продолжительным - около часа.

Составной частью научной организации умственного труда является овладение техникой умственного труда.

Время, которым располагает студент для выполнения учебного плана, складывается из двух составляющих: одна из них — это аудиторная работа в вузе по расписанию занятий, другая - внеаудиторная самостоятельная работа. Задания и материалы для самостоятельной работы выдаются во время учебных занятий по расписанию, на этих же занятиях преподаватель осуществляет контроль за самостоятельной работой, а также оказывает помощь студентам по правильной организации работы.

Самостоятельные занятия потребуют интенсивного умственного труда, который необходимо не только правильно организовать, но и стимулировать. При этом очень важно уметь поддерживать устойчивое внимание к изучаемому материалу. Выработка

внимания требует значительных волевых усилий. Именно поэтому, если студент замечает, что он часто отвлекается во время самостоятельных занятий, ему надо заставить себя сосредоточиться. Подобную процедуру необходимо проделывать постоянно, так как это является тренировкой внимания. Устойчивое внимание появляется тогда, когда человек относится к делу с интересом.

Следует правильно организовать свои занятия по времени: 50 минут - работа, 5-10 минут - перерыв; после 3 часов работы перерыв - 20-25 минут. Иначе нарастающее утомление повлечет неустойчивость внимания. Очень существенным фактором, влияющим на повышение умственной работоспособности, являются систематические занятия физической культурой. Организация активного отдыха предусматривает чередование умственной и физической деятельности, что полностью восстанавливает работоспособность человека.

САМОСТОЯТЕЛЬНАЯ РАБОТА СТУДЕНТА - НЕОБХОДИМОЕ ЗВЕНО СТАНОВЛЕНИЯ ИССЛЕДОВАТЕЛЯ

Прогресс науки и техники, информационных технологий приводит к значительному увеличению научной информации, что предъявляет более высокие требования не только к моральным, нравственным свойствам человека, но и в особенности, постоянно возрастающие требования в области образования – обновление, модернизация общих и профессиональных знаний, умений специалиста.

Всякое образование должно выступать как динамический процесс, присущий человеку и продолжающийся всю его жизнь. Овладение научной мыслью и языком науки является необходимой составляющей в самоорганизации будущего специалиста исследователя. Под этим понимается не столько накопление знаний, сколько овладение научно обоснованными способами их приобретения. В этом, вообще говоря, состоит основная задача вуза.

Специфика вузовского учебного процесса, в организации которого самостоятельной работе студента отводятся все больше места, состоит в том, что он является как будто бы последним и самым адекватным звеном для реализации этой задачи. Ибо во время учебы в вузе происходит выработка стиля, навыков учебной (познавательной) деятельности, рациональный характер которых будет способствовать постоянному обновлению знаний высококвалифицированного выпускника вуза.

Однако до этого пути существуют определенные трудности, в частности, переход студента от синтетического процесса обучения в средней школе, к аналитическому в высшей. Это связано как с новым содержанием обучения (расширение общего образования и углубление профессиональной подготовки), так и с новыми, неизвестными до сих пор формами: обучения (лекции, семинары, лабораторные занятия и т.д.). Студент получает не только знания, предусмотренные программой и учебными пособиями, но он также должен познакомиться со способами приобретения знаний так, чтобы суметь оценить, что мы знаем, откуда мы это знаем и как этого знания мы достигли. Ко всему этому приходят через собственную самостоятельную работу.

Это и потому, что самостоятельно приобретенные знания являются более оперативными, они становятся личной собственностью, а также мотивом поведения, развивают интеллектуальные черты, внимание, наблюдательность, критичность, умение оценивать. Роль преподавателя в основном заключается в руководстве накопления знаний (по отношению к первокурсникам), а в последующие годы учебы, на старших курсах, в

совместном установлении проблем и заботе о самостоятельных поисках студента, а также контролирования за их деятельностью. Отметим, что нельзя ограничиваться только приобретением знаний, предусмотренных программой изучаемой дисциплины, надо постоянно углублять полученные знания, сосредотачивая их на какой-нибудь узкой определенной области, соответствующей интересам студента. Углубленное изучение всех предметов, предусмотренных программой, на практике является возможным, и хорошая организация работы позволяет экономить время, что создает условия для глубокого, систематического, заинтересованного изучения самостоятельно выбранной студентом темы.

Конечно, все советы, примеры, рекомендации в этой области, даваемые преподавателем, или определенными публикациями, или другими источниками, не гарантируют никакого успеха без проявления собственной активности в этом деле, т. е. они не дают готовых рецептов, а должны способствовать анализу собственной работы, ее целей, организации в соответствии с индивидуальными особенностями. Учитывая личные возможности, существующие условия жизни и работы, навыки, на основе этих рекомендаций, возможно, выработать индивидуально обоснованную совокупность методов, способов, найти свой стиль или усовершенствовать его, чтобы изучив определенный материал, иметь время оценить его значимость, пригодность и возможности его применения, чтобы, в конечном счете, обеспечить успешность своей учебы с будущей профессиональной деятельности

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ СТУДЕНТОВ ПО ОТДЕЛЬНЫМ ФОРМАМ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Система вузовского обучения подразумевает значительно большую самостоятельность студентов в планировании и организации своей деятельности.

Работа с литературой

При работе с литературой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги.

Важно помнить, что рациональные навыки работы с книгой — это всегда большая экономия времени и сил.

Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу.

Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода).

При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа.

Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях)

дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем.

Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались.

Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента.

Различают два вида чтения; первичное и вторичное. *Первичное* - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах. После него не должно остаться ни одного непонятного слова. Содержание не всегда может быть понятно после первичного чтения.

Задача *вторичного* чтения полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым).

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Выделяют **четыре основные установки в чтении научного текста**:

1. информационно-поисковый (задача – найти, выделить искомую информацию)
1. усваивающая (усилия читателя направлены на то, чтобы как можно полнее осознать и запомнить как сами сведения, излагаемые автором, так и всю логику его рассуждений)
2. аналитико-критическая (читатель стремится критически осмыслить материал, проанализировав его, определив свое отношение к нему)
3. творческая (создает у читателя готовность в том или ином виде – как отправной пункт для своих рассуждений, как образ для действия по аналогии и т.п. – использовать суждения автора, ход его мыслей, результат наблюдения, разработанную методику, дополнить их, подвергнуть новой проверке).

С наличием различных установок обращения к научному тексту связано существование и нескольких **видов чтения**:

1. библиографическое – просматривание карточек каталога, рекомендательных списков, сводных списков журналов и статей за год и т.п.;
2. просмотровое – используется для поиска материалов, содержащих нужную информацию, обычно к нему прибегают сразу после работы со списками литературы и каталогами, в результате такого просмотра читатель устанавливает, какие из источников будут использованы в дальнейшей работе;
3. ознакомительное – подразумевает сплошное, достаточно подробное прочтение отобранных статей, глав, отдельных страниц, цель – познакомиться с характером информации, узнать, какие вопросы вынесены автором на рассмотрение, провести сортировку материала;
4. изучающее – предполагает доскональное освоение материала; в ходе такого чтения проявляется доверие читателя к автору, готовность принять изложенную информацию, реализуется установка на предельно полное понимание материала;

5. аналитико-критическое и творческое чтение – два вида чтения близкие между собой тем, что участвуют в решении исследовательских задач. Первый из них предполагает направленный критический анализ, как самой информации, так и способов ее получения и подачи автором; второе – поиск тех суждений, фактов, по которым или в связи, с которыми, читатель считает нужным высказать собственные мысли.

Из всех рассмотренных видов чтения основным для студентов является изучающее – именно оно позволяет в работе с учебной литературой накапливать знания в различных областях. Вот почему именно этот вид чтения в рамках учебной деятельности должен быть освоен в первую очередь. Кроме того, при овладении данным видом чтения формируются основные приемы, повышающие эффективность работы с научным текстом.

Основные виды систематизированной записи прочитанного:

1. Аннотирование – предельно краткое связное описание просмотренной или прочитанной книги (статьи), ее содержания, источников, характера и назначения;

1. Планирование – краткая логическая организация текста, раскрывающая содержание и структуру изучаемого материала;

2. Тезирование – лаконичное воспроизведение основных утверждений автора без привлечения фактического материала;

3. Цитирование – дословное выписывание из текста выдержек, извлечений, наиболее существенно отражающих ту или иную мысль автора;

4. Конспектирование – краткое и последовательное изложение содержания прочитанного.

Конспект – сложный способ изложения содержания книги или статьи в логической последовательности. Конспект аккумулирует в себе предыдущие виды записи, позволяет всесторонне охватить содержание книги, статьи. Поэтому умение составлять план, тезисы, делать выписки и другие записи определяет и технологию составления конспекта.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта.

2. Выделите главное, составьте план.

3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора.

4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли.

В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля.

Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Лабораторные занятия

Следует подчеркнуть, что только после усвоения лекционного материала с определенной точки зрения (а именно с той, с которой он излагается на лекциях) он будет закрепляться на лабораторных занятиях как в результате обсуждения и анализа лекционного материала, так и с помощью выполнения заданий лабораторной работы, поставленных задач. При этих условиях студент не только хорошо усвоит материал, но и научится применять его на практике, а также получит дополнительный стимул (и это очень важно) для активной проработки лекции.

При самостоятельном выполнении заданий нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что выполнение каждого учебного задания должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Выполнение заданий данного типа нужно продолжать до приобретения твердых навыков в их решении.

Самопроверка

После изучения определенной темы по записям в конспекте и учебнику, а также решения достаточного количества соответствующих заданий на лабораторных занятиях и самостоятельно студенту рекомендуется, используя лист опорных материалов, воспроизвести по памяти определения, выводы формул, формулировки основных положений и доказательств.

В случае необходимости нужно еще раз внимательно разобраться в материале.

Иногда недостаточность усвоения того или иного вопроса выясняется только при изучении дальнейшего материала. В этом случае надо вернуться назад и повторить плохо усвоенный материал. Важный критерий усвоения теоретического материала - умение решать задачи или пройти тестирование по пройденному материалу. Однако следует помнить, что правильное решение задачи может получиться в результате применения механически заученных формул без понимания сущности теоретических положений.

Консультации

Если в процессе самостоятельной работы над изучением теоретического материала или при выполнении заданий у студента возникают вопросы, разрешить которые самостоятельно не удастся, необходимо обратиться к преподавателю для получения у него разъяснений или указаний. В своих вопросах студент должен четко выразить, в чем он испытывает затруднения, характер этого затруднения. За консультацией следует обращаться и в случае, если возникнут сомнения в правильности ответов на вопросы самопроверки.

Подготовка к экзамену по дисциплине

Изучение многих общепрофессиональных и специальных дисциплин завершается экзаменом. Подготовка к экзамену способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания.

Экзамен - форма промежуточной аттестации студентов по результатам освоения теоретических знаний, приобретения практических навыков и компетенций, целью которой является контроль результатов освоения студентами учебного материала по программе конкретной дисциплины, проверка и оценка знаний, полученных за семестр (курс), их прочности, развития творческого мышления, приобретения навыков самостоятельной работы, умения применять теоретические знания при решении практических задач. Экзамен может быть проведен в устной форме, в устно-письменной форме, в форме письменной работы или тестирования.

Число вопросов, включаемых в экзаменационный билет, должно быть не менее двух и не более пяти, при этом вопросы могут носить как теоретический, так и прикладной характер. На экзамен могут выноситься типовые задачи, проработанные в течение семестра на аудиторных занятиях и в процессе самостоятельной работы. Содержание вопросов и задач, включаемых в экзаменационный билет, должно соответствовать учебной программе дисциплины и фондам оценочных средств. Перечень примерных вопросов, заданий и критерии оценки уровня обученности и уровня сформированности компетенций доводятся до сведения обучающихся в начале семестра изучения дисциплины.

Систематическая подготовка к занятиям в течение семестра позволит использовать время экзаменационной сессии для систематизации знаний.

Текущая аттестация студентов проводится преподавателями, ведущими лабораторные занятия по дисциплине в следующих формах: собеседование при защите лабораторной работы с предоставлением письменного отчета. Допуск к лабораторным работам происходит при наличии у студентов печатного варианта отчета. Защита отчета проходит в форме собеседования. Максимальное количество баллов студент получает, если оформление отчета соответствует установленным требованиям, а отчет полностью раскрывает суть работы. Основанием для снижения оценки являются:

- не качественно выполненная работа на персональном компьютере,
- письменный отчет выполнен в небрежной форме и с ошибками,
- студент не дает исчерпывающий ответ на вопрос преподавателя,
- путается или затрудняется в демонстрации работы на компьютере.

Отчет может быть отправлен на доработку в следующих случаях:

- небрежное выполнение и с ошибками,
- сокращенное содержание отчета.

Критерии оценивания собеседования приведены в Фонде оценочных средств по дисциплине Основы нейронных сетей.

УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

8.1. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля)

8.1.1. Перечень основной литературы:

1. Скворцова Т.И. Компьютерные коммуникации и сети Электронный ре-курс / Скворцова Т.И.: учебно-методическое пособие. - Москва: РТУ МИ-РЭА, 2020. - 223 с., экземпляров неограниченно.

2. Берлин А.Н. Высокоскоростные сети связи Электронный ресурс: учебное пособие / А.Н. Берлин. - Высокоскоростные сети связи. - Москва: Ин-тернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 437 с. - Книга находится в базовой версии ЭБС IPRbooks., экземпляров неограниченно.

3. Проскуряков А.В. Компьютерные сети: основы построения компьютерных сетей и телекоммуникаций: учебное пособие / А.В. Проскуряков; Министерство науки и высшего образования Российской Федерации; Федеральное государственное автономное образовательное учреждение высшего образования «Южный федеральный университет»; Инженерно-технологическая академия. - Ростов-на-Дону|Таганрог: Издательство Южного федерального университета, 2018. - 202 с.: ил. - <http://biblioclub.ru/>. - Библиогр.: с. 195-196. - ISBN 978-5-9275-2792-2, экз-земпляров неограниченно.

8.1.2. Перечень дополнительной литературы:

1. Шишова Н.А. Основы построения инфокоммуникационных систем и сетей [Электронный ресурс]: учебное пособие / Н. А. Шишова. — Электрон. текстовые данные. — М.: Московский технический университет связи и информатики, 2015. — 43 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/61512.html>, экземпляров неограниченно.

2. Олифер В.Г. Компьютерные сети: принципы, технологии, протоколы: учеб. пособие для вузов / В.Г. Олифер, Н.А. Олифер. - 4-е изд. - СПб. [и др.]: Питер, 2014. - 944 с. : ил. - (Учебник для вузов). - Гриф: Рек. МО. - Библиогр.: с. 917. - Алф. указ.: с. 918-943. - ISBN 978-5-469-00004-8 – 14 экземпляров.

3. Оливер Ибе Компьютерные сети и службы удаленного доступа Электронный ресурс: Учебное пособие / Ибе Оливер; пер. И. В. Сеницын. - Компьютерные сети и службы удаленного доступа. - Саратов: Профобразование, 2017. - 333 с. - Книга находится в премиум-версии ЭБС IPR BOOKS. - ISBN 978-5-4488-0054-2, экземпляров неограниченно.

4. Зиангирова Л.Ф. Сетевые технологии Электронный ресурс: Учебно-методическое пособие / Л.Ф. Зиангирова. - Саратов: Вузовское образование, 2017. - 100 с. - Книга находится в премиум-версии ЭБС IPR BOOKS. - ISBN 978-5-906172-30-3, экземпляров неограниченно.

5. Методические рекомендации по выполнению лабораторных работ по дисциплине "Компьютерные сети": Направление подготовки 44.03.05 Педагогическое образование. Профиль подготовки - Информатика и Иностранный язык (английский). Квалификация выпускника - бакалавр. Форма обучения - очная. Учебный план 2018г. Изучается в 7 и 8 семестре. / Сев.-Кав. федер. ун-т. - Ставрополь: СКФУ, 2017. - 206 с., экземпляров неограниченно.

8.2. Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине (модулю)

8.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

NetCracker Professional Users Guide and Reference Manual [электронный ресурс] // <http://www.netcracker.com/>.

Единое окно доступа к образовательным ресурсам: [портал]. URL: <http://window.edu.ru/>

Национальный открытый университет ИНТУИТ. URL: <http://www.intuit.ru/>