

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
Санкт-Петербургский горный университет**

**Кафедра автоматизации технологических процессов
и производств**

ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

*Методические указания к практическим работам
для студентов магистратуры направления 15.04.04*

**САНКТ-ПЕТЕРБУРГ
2019**

УДК 004.9 (073)

ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ: Методические указания к практическим работам / Санкт-Петербургский горный университет. Сост. *А.В. Бойков*. СПб, 2019. 32 с.

Методические указания содержат краткие теоретические сведения, информацию о порядке выполнения работ, основные требования к содержанию отчетов.

Предназначены для студентов магистратуры направления 15.04.04 «Автоматизация технологических процессов и производств».

Научный редактор профессор *В.Ю. Бажин*

Рецензент канд. техн. наук *В.В. Васильев* (АСУТП ООО «ТОМС инжиниринг»)

© Санкт-Петербургский
горный университет, 2019

ПРАКТИЧЕСКАЯ РАБОТА № 1. ОСНОВЫ ПРОЕКТИРОВАНИЯ В СИСТЕМЕ MASTERSCADА

Цель работы: Изучить основы и приобрести навыки работы с системой сбора данных и оперативного диспетчерского управления MasterSCADA.

Теоретические сведения

MasterSCADA — программный пакет для проектирования систем диспетчерского управления и сбора данных (SCADA). Основными свойствами является модульность, масштабируемость и объектный подход к разработке. Система предназначена для сбора, архивирования, отображения данных, а также для управления различными логическими процессами. Помимо создания т. н. верхнего уровня, система позволяет программировать контроллеры с открытой архитектурой.

Порядок выполнения практической работы

1. Запустите OPC-сервер MasterOPC: Пуск – Программы – InSAT – MasterOPC – MasterOPC server. Выполните команду: Открыть – загрузите конфигурацию simulator.mbc. По правой кнопке мыши – Переименуйте тег SIN в Уровень.

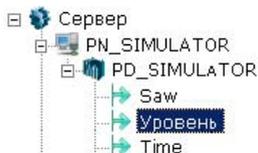


Рис. 1.1 Загрузка конфигурации OPC-сервера

Выполните команды: Сохранить + Сделать стартовым. Закрыйте OPC-сервер.

2. Запустите MasterSCADA – создайте новый проект Контроль уровня. На странице свойств (справа) – выберите вкладку Шкалы – Добавить – Имя: Уровень – Начало: 0, Конец: 8, Аварийные и предупредительные границы задайте согласно рис. 1.2.

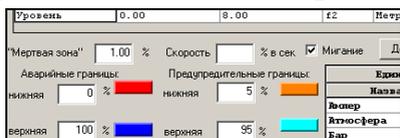


Рис. 1.2 Шкалы

3. По правой кнопке на Системе – Вставить – Компьютер. На странице свойств (справа) – поле Имя переименуйте Компьютер1 в АРМ оператора – Применить.

4. По правой кнопке на АРМ оператора – Поиск OPC DA серверов –InSAT Modbus OPC server DA. По правой кнопке на АРМ оператора – Вставить OPC сервер – InSAT Modbus OPC server DA. По правой кнопке на InSAT Modbus OPC server DA – вставить OPC переменные – Уровень, рис.1.3

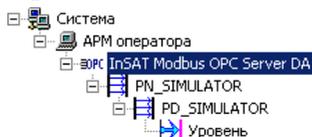


Рис. 1.3 OPC-переменные

5. Перейдите в дерево проектов (внизу слева) – по правой кнопке добавьте объект «Емкость». Справа внизу – выделите функциональный блок: Обработка сигналов – Повторитель – перетащите мышкой на объект Емкость – на странице свойств Общие (справа сверху) – переименуйте Повторитель в Уровень, рис. 1.4.

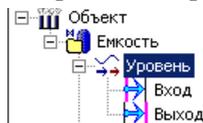


Рис. 1.4 Дерево объектов

6. Установка связи между переменной OPC-сервера Уровень и Входом элемента Уровень – Перетащите переменную (сверху) на Вход элемента Уровень (внизу). На Входе должна появиться розовая вертикальная черта, рис. 1.5. Выделите объект Емкость и по правой кнопке выполните: Вставить – Команду. На вкладке Общие задайте имя: ВПГ (верхняя пороговая граница), Тип: Аналоговый.

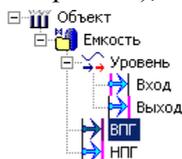


Рис. 1.5 Дерево объектов

7. На вкладке Опрос выхода установите флаг Восстановление

при рестарте. По правой кнопке выполните: Дублировать – переименуйте в НПГ (нижняя пороговая граница). Для команды ВПГ – вкладка Опрос выхода – Значение до опроса: 7. Для команды НПГ – вкладка Опрос выхода – Значение до опроса: 1. Свяжем команды с границами переменной Выход: выделите Выход – вкладка Контроль значений – в поле Верхняя предупредительная граница перетащите команду ВПГ, в поле Нижняя предупредительная граница – команду НПГ, рис. 1.6. Вкладка эл. Выход «Контроль значений».



Рис.1.6 Вкладка эл. Выход «Контроль значений»

8. Для сохранения предупредительных сообщений системы и действий оператора создадим Журнал сообщений: Выделите объект Емкость – вкладка Журналы – Тип: Новое окно – Свойства: флажок Поверх других окон, рис.1.7. Основной журнал.

Название	Тип	Свойства
Основной журнал	Новое окно	...

Рис. 1.7 Основной журнал.

9. Создание тренда: выделите вкладку Тренды – Добавить – Имя: Изменение уровня. По кнопке «Выбор перьев» (звезда)-добавьте переменную Выход, рис.1.8.

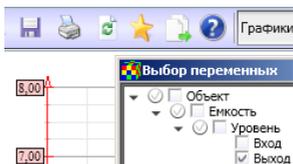


Рис. 1.8 Тренд

10. По правой кнопке мыши на оси Y выберите пункт Установки и установите флаги напротив всех границ, рис.1.9. Установки оси Y.

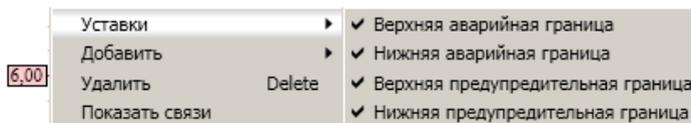


Рис. 1.9 Установки оси Y

Для возврата к менеджеру проектов нажмите кнопку Показать проект.

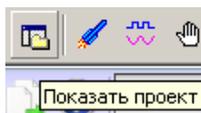


Рис. 1.10 Показать проект

11. На странице свойств Тренды тип задайте: Новое окно

Название	Тип	Свойства
Изменение уровня	Новое окно	...

Рис. 1.11 Создание нового окна

12. По вкладке Архив можно настроить срок хранения. По условию задачи 30 дней. Менять ничего не нужно, т.к. по умолчанию срок определен в 30 дней.

Создание Мнемосхемы – Тип: Окно – включите флажок Стартовая мнемосхема компьютера. По кнопке Открыть – откройте мнемосхему – в Свойствах Цвет фона задайте 224;224;224. Свойства – Включить сетку: Истина. Внизу выберите вкладку Палитра – Графические примитивы – Текст: вверху мнемосхемы введите текст: Контроль уровня емкости. По правой кнопке мыши – Свойства – размер: 28, Цвет: синий. Палитра – Объемные элементы – Цилиндр: по правой кнопке – Свойства – вкладка Окончания – Начало: Срез. Выделите цилиндр – внизу вкладка Свойства – . Динамизация входных значений – выделите свойство Процент заливки – левой кнопкой мыши перетащите переменную Выход функционального блока Уровень. По кнопке (справа от Выхода) – задайте свойства Процента заливки. Процент заливки – Линейное преобразование: Истина.

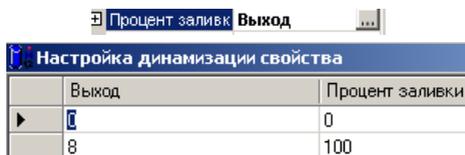


Рис. 1.12 Процент заливки

13. Оператор должен иметь возможность определять значения ВПГ и НПГ: перетащите данные команды на мнемосхему. Свойства

– Ширина границы: 3, см рис. 1.13. Перетащите Выход на цилиндр. Свойства: Ширина границы: 3, Цвет фона: White, Тип границы: Вогнутая граница



Рис. 1.13 Мнемосхема

14. По правой кнопке мыши перетащите объект Емкость на мнемосхему – Основной журнал – в Свойствах задайте – Подпись: Основной журнал. По правой кнопке мыши перетащите объект Емкость на мнемосхему – Изменение уровня – Свойства – Подпись – Тренд. По кнопке Пуск  проверим автоматизированное рабочее место (АРМ) оператора. Выделите Систему – Свойства (справа) – вкладка Операторы – Добавить: Помощник оператора – Посменно: флажок убрать! Операторы – вкладка Права доступа – задать как на рис. 1.14.

Должность Помощник оператора		Лок.	Глоб.	Журнал	Подтве
Категория	Действие				
Проект	Включить режим исполнения	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Отключить режим исполнения	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Навигация в дереве системы	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Рис. 1.14 Окно Операторы

15. АРМ оператора – Свойства – Операторы – выделить Помощник оператора – Добавить – свою фамилию, рис.1.15

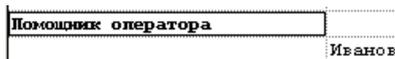


Рис. 1.15 Окно Оператора

16. На рабочем столе создайте ярлык для запуска АРМ оператора.

Задание на практическую работу:

На предприятии установлена емкость высотой 8 метров. Датчик измеряет уровень от 0 до 8 м. Нижняя аварийная граница составляет 5%, верхняя – 95%. Требования к системе:

1. В случае нарушения аварийных границ (верхней или нижней) выдать соответствующее предупреждение.
2. Мнемосхема должна отображать текущее значение уровня жидкости.
3. Предоставить возможность оператору изменять значения предупредительных границ.
4. Оператор должен иметь возможность посмотреть график изменения уровня жидкости по времени.
5. Значения измерений и сообщения о нарушениях должны храниться не менее 1 месяца.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №2. ДАТЧИКИ В СИСТЕМЕ MASTERSCADА

Цель работы: Приобрести навыки по проектированию датчиков на мнемосхеме в системе MasterSCADA.

Порядок выполнения практической работы

1. Создайте каталог /Проект-2 в своей директории. Получите файл конфигурации OPC сервера Симулятор.mbc – сохраните в директории /Проект-2/. Запустите OPC сервер: Пуск – Программы – InSAT – MasterOPC – MasterOPC server. Выполните команду: Открыть – загрузите конфигурацию Симулятор.mbc – Сохранить – Сделать стартовой – Закройте OPC сервер.

2. Запустите MasterSCADA – создайте новый проект Датчики напряжения. Система – по правой кнопке – вставить Компьютер 1. По правой кнопке на АРМ оператора – Вставить OPC сервер – InSAT Modbus OPC server DA. По правой кнопке на InSAT Modbus OPC server DA – вставить OPC переменные, рис. 2.1.

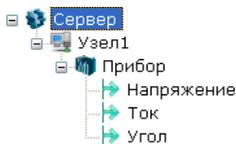


Рис. 2.1 ОРС переменные

3. Перейдите в дерево Объекты – по правой кнопке – Добавить Объект – Имя: Установка – Применить. В объект Установка добавьте подобъект Реактивная мощность – величина, характеризующая нагрузки, создаваемые в электротехнических устройствах колебаниями энергии электромагнитного поля в цепи синусоидального переменного тока, равна произведению среднеквадратичных значений напряжения и тока, умноженному на синус угла сдвига фаз между ними: $Q = U \cdot I \cdot \sin \varphi$. Единица измерения — вольт-ампер реактивный (ВАр).

4. В Палитре элементов выберите Вычисления – Синус Sin – поместите на объект Реактивная мощность.

$$\text{Синус} = \frac{\text{Диапазон}}{2} * \text{Sin}\left(\frac{2 * \text{Pi} * \text{Аргумент}}{\text{Период}}\right)$$

Функциональный вход Синус имеет три входа – *Период*, *Диапазон*, *Аргумент* и один выход – *Синус*, рис. 2.2.

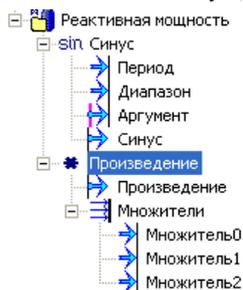


Рис. 2.2 Функциональный вход Синус

5. Выделите вход Период – вкладка Опрос: Константа – 360 – Применить. Выделите вход Диапазон – вкладка Опрос: Константа – 2 – Применить. Аргумент – это угол, поэтому установим связь с входом Угол (перетаскиванием указателя мыши, красная метка сигнализирует об установке связи). В объект Реактивная мощность добавим вычислительный элемент Произведение. Количество множителей зададим 3: Напряжение – вкладка Настройки = 3.

Свяжем Множитель0 с Напряжением дерева Системы, Множитель1 – Ток и Множитель2 – выходом Синус дерева Объектов. Создадим объект Активная мощность: Установка Активная мощность $Q = U \cdot I \cdot \sin\phi$ (Ватт), где U и I – среднеквадратичное значение напряжения и тока, ϕ – угол сдвига фаз между ними) – по правой кнопке добавим Объект – Имя: Активная мощность) – Применить. В Палитре элементов выберите Вычисления – Синус Cos – поместите на объект Активная мощность. Период – вкладка Опрос – 360 – Применить. Диапазон – Опрос – 2 – Применить. Свяжем вход Уголь дерева Системы с Аргументом косинус. По правой кнопке скопируем Произведение из блока Синус в буфер обмена и вставим из буфера в блок Косинус – на вопрос «Восстановление внешних связей» – Восстановить. Свяжем Косинус на Множитель2. Создадим объект Полная мощность: Установка – по правой кнопке – вставить Объект – Имя: Полная мощность (Полная мощность $S = U \cdot I$. Единица полной электрической мощности — вольт-ампер (В·А)). По правой кнопке добавим Расчет – вкладка Формула – перетащим из дерева Системы Напряжение и Ток в источники данных (рис. 2.3) – в поле формула перетащим Напряжение – знак умножения – Ток – Применить.

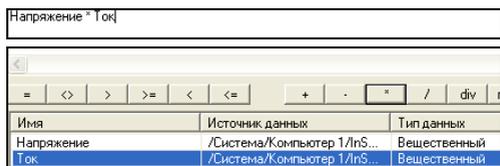


Рис. 2.3 Формула расчета

6. Создадим Мнемосхему: выделим объект Установка – вкладка Окна – Создать. В Палитре (внизу) – выберем Датчики – Стрелочный прибор – поместим на Мнемосхему. Выберем Свойства (слева от Палитры) – перетащим выход Произведение на Страницу свойств. В свойстве Текст заголовка введите Реактивная. Аналогично разместите еще два стрелочных прибора: Активная (вход Произведение) и Полная (Расчет). Настроим шкалы: Система – Шкалы – Добавить – Реактивная – начало:0, конец: 2500, ед. измерения ВАр (Рис. 2.4). Аналогично добавить Активная (ед. изм. Ватт) и Полная (ед. изм. ВА). Примечание: Аварийные и Предупредительные границы убрать.

Уровень 100	0.0	100.0	f1	Процент
Реактивная	0.0	2500.0	f2	Реактивная мощность
Активная	0.0	2500.0	f2	Ватт
Полная	0.0	2500.0	f2	Полная мощность

Четвертая зона" % Скорость % в сек Мигание

Аварийные границы: Предупредительные границы:

Единицы измерения	
Название	Обозначение
Сантиметр в минуту	см/мин
Сантиметр в секунду	см/с
Реактивная мощность	ВАр
Полная мощность	ВА

Рис. 2.4 Настройки стрелочного прибора

7. Назначим выходу Производство блока Реактивная мощность – вкладка Общие – диапазон изменения: Реактивная. Назначим выходу Производство блока Активная мощность – вкладка Общие – диапазон изменения: Активная. Назначим выходу Расчет блока Полная мощность – вкладка Общие – диапазон: Полная. Откроем Мнемосхему: Установка – Окна – Открыть – щелкните поочередно по датчикам – результат: индикаторы поменяли шкалу. Запустите мнемосхему.

8. Из палитры мнемосхемы – Окна – Тренд: разместите на оставшейся части окна. Перетаскиванием на график добавим переменные: выход Реактивного производства, выход Активного производства и Расчет полной мощности. По правой кнопке зададим Уровень: Базовый – Легенду переименуем в Реактивный, Активный и Полный соответственно.

Задание на практическую работу:

На мнемосхеме необходимо отражать: активную, реактивную и полную мощность. Напряжение, ток и угол ф (фи) поступают через OPC сервер. Мощность необходимо рассчитать.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №3. ПРОГРАММА УПАКОВКИ 8 БИТ В БАЙТ

Цель работы: Познакомиться с системой программирования логических контроллеров MasterPLC Designer.

Теоретические сведения

Промышленный контроллер обладает входом в 8 бит и выходом в 1 байт. Требуется перевести параллельный двоичный код в десятичное число (выход).

Для перевода двоичного числа в десятичное, необходимо это число представить в виде суммы произведений степеней основания двоичной системы счисления на соответствующие цифры в разрядах двоичного числа.

Пример: Дано двоичное число $1010\ 0010_2$

Перевод двоичного числа в десятичное: $1010\ 0010_2 = 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 128 + 32 + 2 = 162_{10}$

Порядок выполнения практической работы

1. Запустите программу MasterPLC Designer – Новый проект: Наименование: Project_1; Тип: Стандартный

2. В разделе Система раскройте список фильтров и задайте «Полное дерево (нет фильтров)», рис.3.1

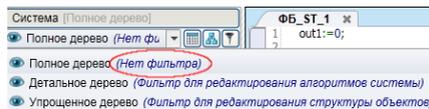


Рис. 3.1 Настройка фильтра дерева Системы

3. В разделе Система по правой кнопке Добавить – Контроллер Windows – Переименуйте в Контроллер 1

4. Раскройте группу Контроллер 1 – Ресурсы – ФБ (функциональный блок) и по правой кнопке Добавить – ФБ ST.

5. Переименуйте программу ФБ ST_1 в rack8bits

6. Раскройте группу rack8bits – Параметры – по правой кнопке Добавить – Параметр, рис.3.2

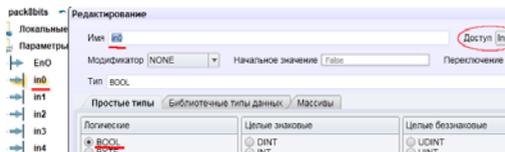


Рис. 3.2 Создание входов и выходов программы

7. По правой кнопке мыши – Дублировать – продублируйте переменную in0 7 раз. Аналогично создадим переменную out1: Доступ: Output (выход) – Тип данных: USINT (от 0 до 255, размер 1

байт).

8. В редакторе программы задается тело программы, см. Листинг 1. Код упаковки 8 бит (in0, in1,...,in7) в 1 байт (out1)

```
out1:=0;  
IF(in0)THEN out1:=out1+1;END_IF;  
IF(in1)THEN out1:=out1+2;END_IF  
IF(in2)THEN out1:=out1+4;END_IF  
IF(in3)THEN out1:=out1+8;END_IF;  
IF(in4)THEN out1:=out1+16;END_IF;  
IF(in5)THEN out1:=out1+32;END_IF  
IF(in6)THEN out1:=out1+64;END_IF  
IF(in7)THEN out1:=out1+128;END_IF;
```

9. Сохраните проект. Компиляция проекта: Исполнение – Проверить. В случае обнаружения синтаксических ошибок кода программы – проверить наличие переменных, тип данных, комментаторные скобки и т.д. Чтобы программа выполнялась в контроллере ее нужно поместить в основную задачу – Перетащите программу pack8bits в Основную задачу.

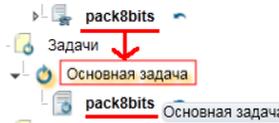


Рис. 3.3 Запуск программы на контроллере

10. Выделите группу Основная задача и задайте период запуска основной задачи 1 секунда (1000 мс) в окне Настройки. Для запуска проекта щелкните кнопку Эмуляция – раскройте правый (Вход) и левый (Выход) клемник – задайте TRUE для in1, in5 и in7 значение TRUE.

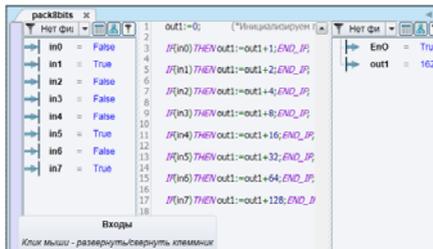


Рис. 3.4 Эмуляция

Примечание. При ручном изменении значения, переменная автоматически блокируется пиктограмма Блокировка (Замок)

11. Симуляция контроллера: По правой кнопке на ФБ добавьте ФБ FBD – перетащите в группу Основная задача -откройте окно блок схем FBD – перетащите в него программу pack8bits

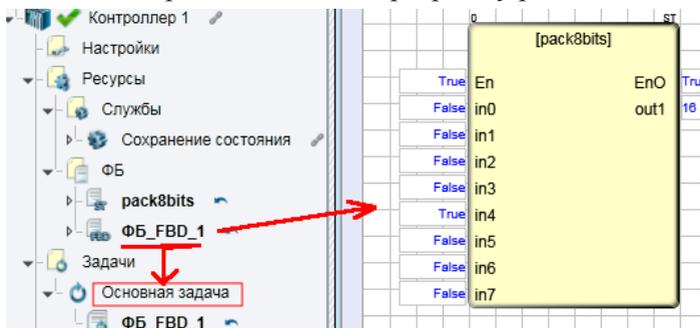


Рис. 3.5 Симуляция контроллера

12. Запустите Эмуляция – протестируйте результат работы – сохраните проект.

13. Закомментируйте код программы и найдите сумму произведений степеней по формуле, используя стандартную функцию преобразования типов:

$$\begin{aligned} out1 := & bool_to_usint(in0) \cdot 1 + bool_to_usint(in1) \cdot 2 + bool_to_usint(in2) \cdot 4 + \\ & bool_to_usint(in3) \cdot 8 + bool_to_usint(in4) \cdot 16 + \\ & bool_to_usint(in5) \cdot 32 + bool_to_usint(in6) \cdot 64 + \\ & bool_to_usint(in7) \cdot 128. \end{aligned}$$

Задание на практическую работу:

Реализовать программу упаковки 8 бит в 1 байт.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 4. СОЗДАНИЕ ПРОЕКТА В TRACE MODE

Цель работы: освоить методику создания системы мониторинга, содержащую один узел APM с использованием механизма автопостроения каналов TRACE MODE методом «от шаблона экрана».

Порядок выполнения практической работы

Разработка любого проекта автоматизации всегда начинается с запуска Интегрированной среды разработки (ИСР). Для ее запуска необходимо выполнить команду TRACE MODE IDE 6 (base) из меню Программы WINDOWS или двойным щелчком ЛК мыши по иконке  рабочего стола Windows. С помощью иконки  с инструментальной панели создать новый проект, при этом в открывшемся на экране диалоге рис. 4.1. Выберем Простой стиль разработки.

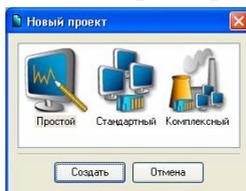


Рис. 4.1 Создание нового проекта

После нажатия левой клавиши мыши (ЛК) на экранной кнопке Создать, в левом окне Навигатора проекта появится дерево проекта с созданным узлом АРМ RTM_1 рис. 4.2. Откроем узел RTM_1 двойным щелчком ЛК, в правом окне Навигатора проекта отобразится содержимое узла – пустая группа Каналы и один канал класса Вызов Экран#1, предназначенный для отображения на узле АРМ графического экрана.

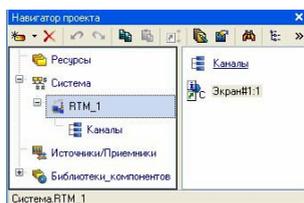


Рис. 4.2 Навигатор проекта

Создание графического экрана. Двойным щелчком ЛК на компоненте Экран#1 открыть окно графического редактора.

Создание статического текста. На панели инструментов графического редактора выделить иконку графического элемента (ГЭ) , на поле редактора установить прямоугольник ГЭ, для чего зафиксировать ЛК «точку привязки»; развернуть прямоугольник движением курсора и зафиксировать выбранный ГЭ. Разместим в

левом верхнем углу экрана статический текст – надпись: «Значение параметра».

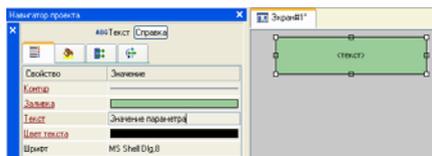


Рис. 4.3 Свойства статического текста

Создание динамического текста, создание аргумента экрана в процессе настройки динамического текста. Подготовим на экране вывод динамического текста для отображения численного значения какого-либо источника сигнала – внешнего или внутреннего путем указания динамизации атрибута ГЭ. Определим назначение аргумента шаблона экрана. Создать и разместить новый ГЭ ABC справа от ГЭ с надписью «Значение параметра». Двойным щелчком ЛК на строке Текст вызвать меню Вид индикации. В правом поле строки нажать ЛК и вызвать список доступных типов, выбрать тип Значение и свойства Привязка. В открывшемся окне Свойство привязки, нажав кнопку на его панели инструментов, создать аргумент экрана рис. 4.4. Двойным щелчком ЛК выделить имя аргумента и изменить его, введя с клавиатуры «Параметр» (завершить ввод нажатием клавиши Enter). Подтвердить связь с этим аргументом нажатием кнопки Готово. Закрыть окно свойств ГЭ.

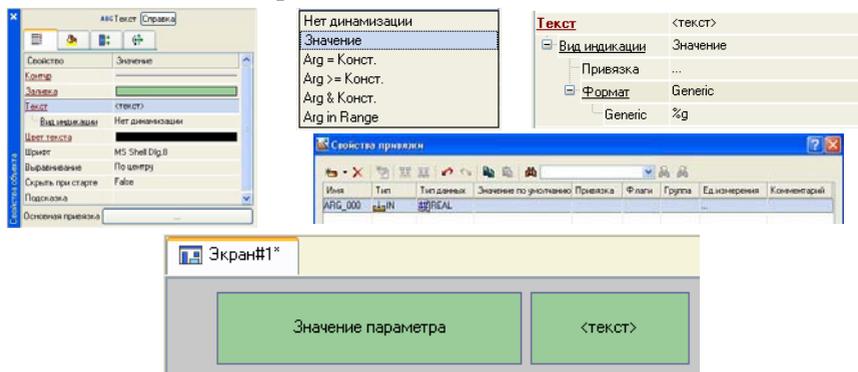


Рис. 4.4 Свойства динамического текста

Создание стрелочного прибора, привязка к тому же аргументу. Применим для отображения параметра новый тип ГЭ – Стрелочный прибор рис. 4.5. Выделить двойным щелчком ЛК на инструментальной панели редактора графики иконку  и выбрать в появившемся меню иконку стрелочного прибора . Установить ГЭ Стрелочный прибор, выбрав его размер таким, чтобы все элементы графики и текста на нем было разборчивы и симметричны.



Рис. 4.5 Стрелочный прибор

Перейти в режим редактирования и открыть окно свойств ГЭ Стрелочный прибор. Щелчком ЛК на кнопке Основная привязка открыть окно табличного редактора аргументов. ЛК выбрать аргумент шаблона экрана Параметр. Подтвердить выбор ЛК на кнопке Готово. Двойным щелчком ЛК открыть свойство Заголовок и в строке Текст ввести слово «Параметр». Закрыть окно свойств.

Автопостроение канала. Для создания канала в узле проекта по аргументу шаблона экрана воспользуемся процедурой автопостроения. В слое Система открыть узел RTM_1. С помощью ПК вызвать контекстное меню и ЛК открыть свойства компонента Экран#1. Выбрать вкладку Аргументы, выделить ЛК аргумент Параметр и с помощью иконки  создать канал класса Float типа Input с именем Параметр.

Создание генератора синуса и привязка его к каналу. Введем в состав проекта источник сигнала – внутренний генератор синусоиды, свяжем его с созданным каналом и опробуем выполненные средства отображения. Открыть слой Источники/Приемники и через ПК создать в нем группу Генераторы. Двойным щелчком ЛК открыть группу Генераторы и через ПК создать в ней компонент Синусоида рис. 4.6. Захватить с помощью ЛК созданный источник и, не отпуская ЛК, перетащить курсор на узел RTM_1 в слое Система,

а затем, в открывшемся окне компонентов RTM_1, на канал Параметр. Отпустить ЛК.

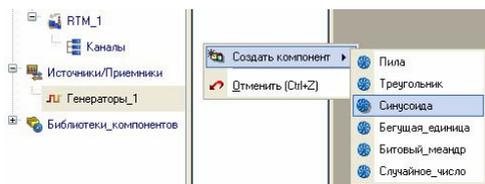


Рис. 4.6 Создание компонента Синусоида

Запуск проекта. Закрыть окно графического редактора. Сохранить проект с помощью иконки .

На инструментальной панели выбрать команду  и скомпилировать проект для запуска в реальном времени. ЛК выделить в слое Система узел RTM_1, выбрать

иконку  на инструментальной панели и запустить режим исполнения. В открывшемся окне ГЭ справа от надписи «Значение параметра» должно показываться изменение синусоидального сигнала. То же значение должен отображать и стрелочный прибор.

Добавление функции управления. Введем в состав графического экрана средство, позволяющее реализовать ввод числовых значений с клавиатуры. Создадим новый аргумент шаблона экрана для их приема. Вызвать графический экран на редактирование. Выбрать на инструментальной панели графического редактора иконку ГЭ Кнопка – . С помощью мыши разместить его в поле экрана под ГЭ Стрелочный прибор. Перейти в режим редактирования, вызвать окно свойств ГЭ. В поле Текст ввести «Управление». Открыть бланк События  и ПК раскрыть меню По нажатию (pressed). Выбрать из списка команду Добавить Send Value, раскрыть меню настроек выбранной команды. В поле Тип передачи (Send Type) выбрать из списка Ввести и передать (Enter & Send). ЛК в поле Результат вызвать табличный редактор аргументов. Создать еще один аргумент и задать ему имя Управление. Изменить тип аргумента на IN/OUT, кнопкой Готово подтвердить привязку атрибута ГЭ к этому аргументу.

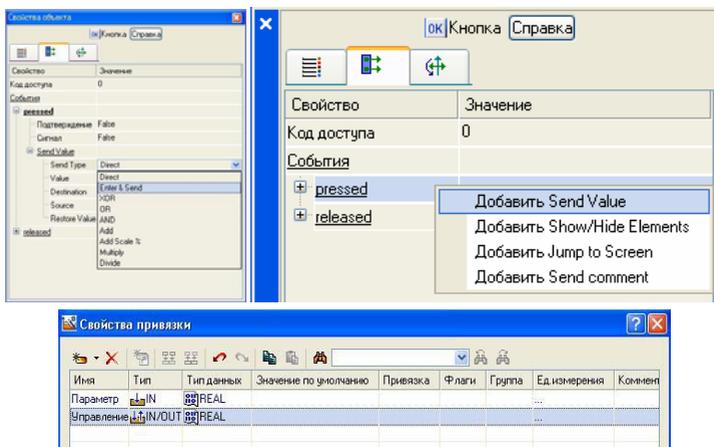


Рис. 4.7 Настройка графического элемента Кнопка

Закреть окно свойств ГЭ с помощью щелчка ЛК по иконке . Выделить ЛК ГЭ Текст, служащий для отображения значения Параметра. С помощью иконки  на панели инструментов или комбинацией клавиш Ctrl+C скопировать выделенный ГЭ Текст в буфер обмена. Далее с помощью иконки  или комбинацией клавиш Ctrl+V извлечь копию ГЭ из буфера обмена и поместить ее на графический экран. Переместить, удерживая нажатой ЛК, копию ГЭ Текст справа от размещенного на экране ГЭ Кнопка. Двойным щелчком ЛК на перемещенном ГЭ Текст открыть окно его свойств. Двойным щелчком ЛК на строке Текст вкладки основных свойств  перейти к настройке динамизации данного атрибута ГЭ. В правом поле строки Привязка щелчком ЛК открыть табличный редактор аргументов шаблона экрана. Выделить ЛК в списке аргумент Управление и щелчком ЛК по экранной кнопке Готово подтвердить привязку атрибута ГЭ Текст к данному аргументу шаблона экрана. Закреть окно свойств ГЭ Текст.

Привязка аргумента экрана к каналу. Создадим по аргументу Управление шаблона экрана новый канал, отредактируем его привязку. В слое Система открыть узел RTM_1. С помощью ПК вызвать через контекстное меню свойства компонента Экран#1. Выбрать вкладку Аргументы, ЛК выделить аргумент Управление и с

помощью иконки  создать новый канал. В результате, в узле RTM_1, будет автопостроен канал с именем Управление. Двойным щелчком в поле Привязка аргумента Управление вызвать окно настройки связи, выбрать в нем атрибут Входное значение канала Управление и кнопкой Привязка подтвердить связь аргумента экрана Управление с атрибутом Входное значение канала Управление.

Размещение ГЭ Тренд. Дополним созданный экран новым ГЭ для совместного просмотра изменений значений каналов узла во времени и отслеживании предыстории – трендом. В правой части графического экрана разместим ГЭ Тренд для вывода значений Параметр и Управление. Основные свойства ГЭ оставим заданными по

умолчанию. Перейдем во вкладку  и, выделив ЛК строку. Кривые, с помощью ПК создадим две новые кривые. Настроим их привязки к аргументам, толщину и цвет линий рис. 4.8.

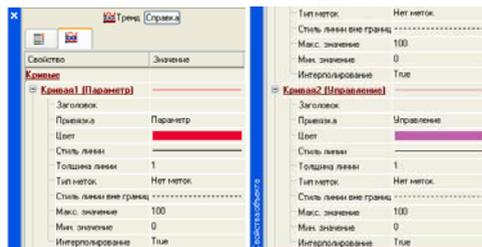


Рис. 4.8 Свойства графического элемента

Простейшая обработка данных. С помощью создания нового компонента проекта – шаблона программы свяжем операцией сложения два имеющихся канала. Будем суммировать реальные значения каналов Параметр и Управление, а результат помещать во вновь созданный аргумент экрана Сумма (с отображением на ГЭ Текст и Тренд) без создания дополнительного канала в узле проекта. Скопировать два первых ГЭ – «Значение параметра» и «text» и разместить их ниже ГЭ Кнопка. Изменить статический текст первого ГЭ на «Сумма:». А динамику второго ГЭ привязать к третьему аргументу экрана типа IN с именем Сумма, который создать в процессе привязки рис. 4.9. Добавить еще одну кривую на тренд с привязкой к аргументу Сумма.

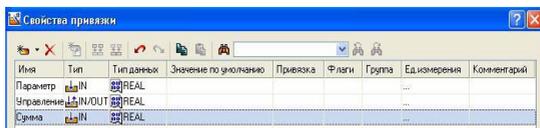


Рис. 4.9 Свойства привязки

Создание программы на языке ST. Создадим программу, в которой сумма двух аргументов, связанных с атрибутами Реальное значение каналов Параметр и Управление, будет помещается в третий аргумент с именем Сумма. В дальнейшем, воспользуемся возможностью связывания аргументов шаблонов для вывода на экран результата работы программы без создания дополнительного канала. Двойным щелчком ЛК открыть узел RTM_1 и создать в нем компонент Программа, рис. 4.10. Выделить компонент Программа#1 и ПК вызвать контекстное меню, выбрав в котором ЛК пункт Редактировать шаблон, перейти в режим редактирования программы.

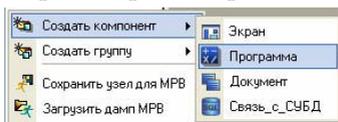


Рис. 4.10 Режим редактирования программы

Выделением ЛК в дереве шаблона Программа#1 строки Аргументы вызвать табличный редактор аргументов. Иконкой  создать в редакторе аргументов три аргумента с именами Параметр, Управление и Сумма. При этом первые 2 аргумента должны быть типа IN, а третий – OUT. Выделить в дереве шаблона строку Программа#1 и в открывшемся диалоге Выбор языка выбрать язык ST. По нажатию экранной кнопки Принять в открывшемся окне редактора программ с объявленными переменными набрать следующую строку рис. 4.11.

```
PROGRAM
VAR_INPUT Параметр : REAL; END_VAR
VAR_INPUT Управление : REAL; END_VAR
VAR_OUTPUT Сумма : REAL; END_VAR

Сумма=Параметр+Управление;

END_PROGRAM
```

Рис. 4.11 Вид программы на языке ST

С помощью иконки  на инструментальной панели редактора или «горячей клавишей» F7 скомпилировать программу и убе-

даться в успешной компиляции в окне Выход (Output), вызываемого из инструментальной панели с помощью иконки .

Привязка аргументов программы. Выполним привязку аргументов программы к атрибутам каналов. Вызвать свойства компонента Программа#1 через контекстное меню. Выбрать вкладку Аргументы. Двойным нажатием в поле Привязка привязать аргументы программы к атрибутам каналов – аргумент Параметр к реальному значению канала Параметр, аргумент Управление к реальному значению канала Управление рис. 4.12.

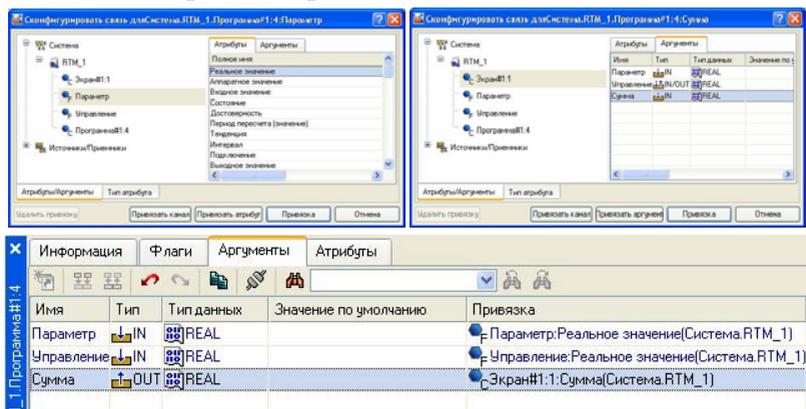


Рис. 4.12 Привязка аргументов программы к атрибутам каналов (пример)

Задание на практическую работу:

Освоить методику создания системы мониторинга, содержащую различные элементы ввода-вывода, индикации, генерации и обработки сигналов в TRACE MODE методом «от шаблона экрана».

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 5. РЕАЛИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ ПРИ ПОМОЩИ SCADA-СИСТЕМЫ TRACE MODE

Цель работы: освоить методику программирования логических функций при помощи SCADA-системы TRACE MODE на языке FBD.

Порядок выполнения практической работы

Рассмотрим реализацию следующей логической функции $f = x_1x_2x_4 \cup x_1x_2 \cup x_1x_3 \cup x_2x_4$ при помощи SCADA-системы TRACE MODE на языке FBD.

Двойным щелчком ЛК на компоненте Экран#1 открыть окно графического редактора. Подготовим на экране вывод динамического текста для отображения численного значения входных переменных X1, X2, X3, X4 и выходного значения Y путем указания динамизации атрибута графического элемента (ГЭ). Определим назначение аргумента шаблона экрана. Создадим и разместим пять ГЭ для ввода/вывода Значения переменных. А также создадим пять аргументов экрана 4 входных и один выходной рис. 5.1.

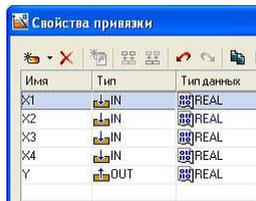


Рис. 5.1 Создание аргументов экрана

Двойным щелчком ЛК выделить имя аргумента и изменить его, введя с клавиатуры «X1» (завершить ввод нажатием клавиши Enter). Подтвердить связь с этим аргументом нажатием кнопки Готово; Аналогичным образом настроить все входные и выходные аргументы.

Введем в состав графического экрана средство, позволяющее реализовать ввод числовых значений с клавиатуры. Создадим новый аргумент шаблона экрана для их приема. Выбрать на инструментальной панели графического редактора иконку ГЭ Кнопка –

Перейти в режим редактирования первой кнопки, вызвать окно свойств ГЭ рис. 5.2. В поле Текст ввести «X1». Открыть бланк События и ПК раскрыть меню По нажатию (pressed). Выбрать из списка команду Добавить Передать значение, раскрыть меню настроек выбранной команды. В поле Тип передачи (Send Type) вы-

брать из списка Ввести и передать (Enter & Send). ЛК в поле Результат вызвать табличный редактор аргументов и произвести привязку к аргументу X1. Аналогичным образом настроить остальные кнопки x2, x3, x4.

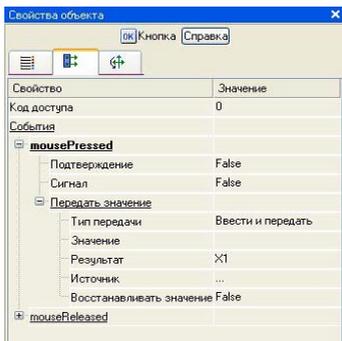


Рис. 5.2 Свойства графического элемента X1

Создадим по аргументам X1, X2, X3, X4 и выходу Y шаблона экрана новые каналы и отредактируем их привязку, рис. 5.3.

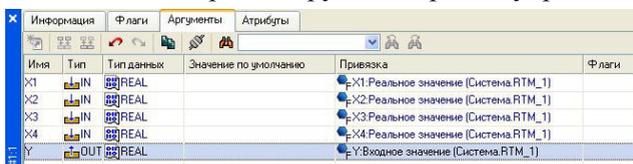


Рис. 5.3 Окно свойств экрана

Выбрать вкладку Аргументы, ЛК выделить аргументы X1, X2, X3, X4 и выход Y и с помощью иконки  создать новый канал. В результате, в узле RTM_1, будут автопостроены каналы X1, X2, X3, X4 и Y.

Создание программы на языке FBD. Создадим программу, которая будет реализовывать заданную логическую функцию. Двойным щелчком ЛК открыть узел RTM_1 и создать в нем компонент Программа. Выделением ЛК в дереве шаблона Программа#1 строки Аргументы вызвать табличный редактор аргументов иконкой  создать в редакторе аргументов четыре аргумента X1, X2, X3, X4 и выход Y. При этом первые 4 аргумента должны быть типа IN, а последний – OUT. Выделить в дереве шаблона строку Программа#1 и

в открывшемся диалоге Выбор языка выбрать язык FBD. По нажатию экранной кнопки Принять в открывшемся окне редактора программ с объявленными переменными создать программу в соответствии с заданием. Для выбора палитры FBD блоков необходимо ЛК мыши нажать на кнопку  после чего появится окно выбора FBD блоков рис. 5.4.

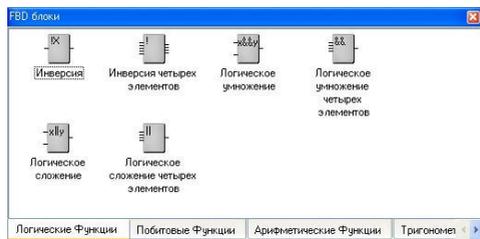


Рис. 5.4 Палитра FBD блоков

При разработке программы верхние входы FBD блоков не используются т.к. они предназначены для изменения порядка пересчета блоков, а информационными входами, являются входы начиная со второго. Для реализации логической функции выберем следующие блоки: из раздела Логические Функции FBD блоки инверсия (!X), логическое умножение (X&&Y и &&), логическое сложение (||). После размещения всех блоков для рассматриваемого примера программа будет выглядеть следующим образом рис. 5.5.

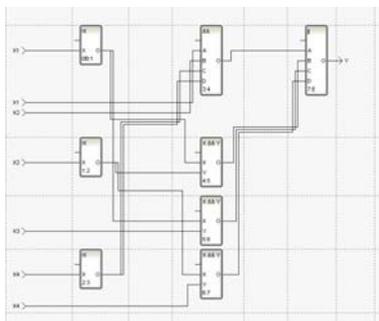


Рис. 5.5 Программа на языке FBD

Выполним привязку аргументов программы к атрибутам каналов. Вызвать свойства компонента **Программа#1** через контекстное меню. Выбрать вкладку **Аргументы**. Двойным нажатием в поле

Привязка привязать аргументы программы к атрибутам каналов – аргументы X1, X2, X3, X4 к реальному значению каналов X1, X2, X3, X4.

Имя	Тип	Тип данных	Значение по умолчанию	Привязка
X1_R	IN	REAL		FX1:Реальное значение (Система.RTM_1)
X2_R	IN	REAL		FX2:Реальное значение (Система.RTM_1)
X3_R	IN	REAL		FX3:Реальное значение (Система.RTM_1)
X4_R	IN	REAL		FX4:Реальное значение (Система.RTM_1)
Экран_1_1_Y	OUT	REAL		FXЭкран#1:1:Y (Система.RTM_1)

Рис. 5.6 – Окончательная настройка связи

Задание на практическую работу:

Получить у преподавателя логическую функцию и реализовать ее при помощи SCADA–системы TRACE MODE на языке FBD.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 6. РЕАЛИЗАЦИЯ ОДНОКОНТУРНОЙ САР ПРИ ПОМОЩИ SCADA-СИСТЕМЫ TRACE MODE

Цель работы: освоить методику программирования одноконтурной системы автоматического регулирования при помощи SCADA–системы TRACE MODE на языке FBD.

Порядок выполнения практической работы

Рассмотрим реализацию одноконтурной системы автоматического регулирования при помощи SCADA–системы TRACE MODE на языке FBD.

Двойным щелчком ЛК на компоненте Экран#1 открываем окно графического редактора.

Подготовим на экране вывод динамического текста для отображения численного значения входных переменных аperiodического звена первого порядка: коэффициента усиления объекта ($K_{об}$), постоянной времени объекта ($T_{об}$), времени запаздывания ($T_{ау}$); ПИД регулятора: коэффициент усиления регулятора (K_r), времени

интегрирования (T_i), времени дифференцирования (T_d); и величины задания (Zad). Выбрать на инструментальной панели графического редактора иконку ГЭ Кнопка – . С помощью мыши разместить семь элементов в поле экрана. Перейти в режим редактирования нажатием  кнопки, затем двойным щелчком ЛК вызвать окно свойств первого ГЭ. В поле Текст ввести «К объекта». Открыть бланк Действия  и (правой кнопкой мыши) ПК раскрыть меню События По нажатию (mouse Pressed). Выбрать из списка команду Добавить Передать значение, раскрыть меню настроек выбранной команды. В поле Тип передачи (Send Type) выбрать из списка Ввести и передать (Enter & Send).

ЛК в поле Результат вызвать табличный редактор аргументов. В открывшемся окне Свойство привязки, нажав кнопку  на его панели инструментов, создать восемь аргументов экрана 7 входных и один выходной. Входной (IN) или Выходной (OUT) будет аргумент, выбирается из низпадающего меню, появляющегося при двойном щелчке ЛК по типу аргумента. Двойным щелчком ЛК выделить имя аргумента и изменить его, введя с клавиатуры «Kob» (завершить ввод нажатием клавиши Enter). Подтвердить связь с этим аргументом нажатием кнопки Готово Аналогичным образом настроить все остальные аргументы. Аналогичным образом настроить остальные кнопки Tob, Tau, Kr, T_i , T_d , Zad .

Для отображения числового значения входных аргументов создадим и разместим семь ГЭ  нажатием. Для этого, активировав ГЭ нажатием ЛК, рисуем области вывода динамического текста, задавая левый верхний и правый нижний углы щелчком ЛК.

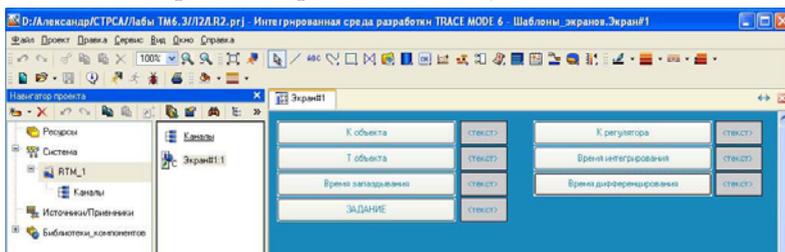


Рис. 6.1 Создание графических элементов

После этого необходимо переключиться в режим редактирования активацией ГЭ . Двойным щелчком ЛК на строке Текст вызвать меню Вид индикации. В окне «Свойства объекта», двойным щелчком ЛК по свойству Текст развернуть его. В появившемся подпункте Вид индикации щелкнув по значению Нет динамизации выбрать его тип Значение. Выбрать свойство Привязка и связать с Kob.

Дополним созданный экран новым ГЭ для совместного просмотра изменений значений каналов узла во времени и отслеживания предыстории – трендом. На экране разместим ГЭ Тренд для вывода значений Zad и Vihod. Основные свойства ГЭ оставим заданными по умолчанию. Перейдем во вкладку и, выделив ЛК строку Кривые, с помощью ПК создадим две новые кривые. Настроим их привязки к аргументам, толщину и цвет линий: Двойным щелчком ЛК мыши в поле тренда открываем окно Свойства тренда и переходим на вкладку Кривые. Нажимая ПК мыши в поле кривые задаем две кривые Zad и Vihod, а в поле привязка привязываем созданные кривые к заданию и выходу.

Создадим по аргументам Kob, Tob, Tau, Kr, Ti, Td, Zad, Vihod, шаблона экрана новые каналы и отредактируем их привязку. В слое Система открыть узел RTM_1. С помощью ПК вызвать контекстное меню свойства компонента Экран#1 и выбрать команду Свойства рис. 6.18. Выбрать вкладку Аргументы, ЛК, при нажатой клавише Cntrl, выделить аргументы Kob, Tob, Tau, Kr, Ti, Td, Zad, Vihod и с помощью иконки создать новые каналы. В результате, в узле RTM_1, будут автопостроены следующие каналы.

Создадим программу, которая будет реализовывать одноконтурную систему автоматического регулирования. Для этого ПК по узлу RTM_1 вызвать контекстное меню выбрать подменю «Создать компонент» и выбрать ЛК в нем компонент Программа. Выделением ЛК в дереве шаблона Программа#1 строки Аргументы вызвать табличный редактор аргументов иконкой  и создать в редакторе аргументов следующие аргументы программы Kob, Tob, Tau, Kr, Ti, Td, Zad,. При этом аргументы Kob, Tob, Tau, Kr, Ti, Td, Zad должны быть типа IN, а Vihod – OUT.

Щелкнув ЛК в дереве шаблона строку Программа#1 и в открывшемся диалоге Выбор языка выбрать язык FBD рис. 6.2. Для

создания системы управления выберем следующие блоки: из раздела Арифметические Функции FBD блок вычитание (X–Y); из раздела Регулирование FBD блок модель объекта (OBJ) и звено PID (PID).

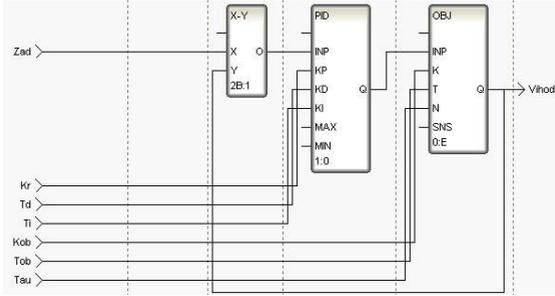


Рис. 6.2 – Программа реализации системы управления

С помощью иконки на инструментальной панели редактора или «горячей клавишей» F7 скомпилировать программу и убедиться в успешной компиляции в окне Выход (Output), вызываемого из инструментальной панели с помощью иконки.

Выполним привязку аргументов программы Kob, Tob, Tau, Kr, Ti, Td, Zad к атрибутам каналов. Вызвать свойства компонента Программа#1 через контекстное меню. Выбрать вкладку Аргументы.

Задания для практической работы

Реализовать одноконтурную систему автоматического регулирования при помощи SCADA–системы TRACE MODE на языке FBD. Вариант задания выбирается исходя из номера буквы в алфавите по инициалам студента. Номер первой буквы Фамилии в алфавите – коэффициент усиления объекта, Имени – постоянная времени объекта, Отчества – время запаздывания.

Например, Иванов Сергей Васильевич И–10, С–19, В–3.

Получаем передаточную функцию $W(s) = \frac{10 \cdot e^{-2s}}{19s+1}$

Расчёт настроек регулятора производится по формулам:

$$Kr = \frac{1,2}{K \cdot T}, T_{из} = 2\tau, T_{пр} = 0,4\tau$$

$$T_i = \frac{T_{из}}{Kr}, T_d = T_{пр}Kr$$

В Trace Mode вместо времени интегрирования вводится коэффициент, равный $\frac{1}{T_i}$.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 7. РАЗРАБОТКА SCADA-ЭКРАНА УЧАСТКА ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ

Цель работы: Индивидуальное задание для закрепления умений, приобретенных в ходе выполнения практических работ в SCADA-системе MasterScada или Trace Mode.

Задание на практическую работу:

Разработать операторский экран цеха/участка промышленного предприятия или отразить технологический процесс в соответствии со знаниями, полученными во время производственной практики.

Операторский экран должен содержать:

1) мнемосхему цеха/участка промышленного предприятия или фрагмент технологического процесса с указанием входных и выходных параметров (не менее 10 шт.);

2) графики наиболее важных параметров (не менее 2 шт.) в режиме реального времени;

3) обработка аналоговых (не менее 1 шт.) или цифровых (не менее 1 шт.) сигналов с датчиков.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

РЕКОМЕНДУЕМЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Деменков Н.П.* Системы автоматического управления на основе программируемых логических контроллеров. Техническая коллекция Schneider Electric. Выпуск № 16, 2008.
2. *Карнов А.Г.* Цифровые системы автоматического регулирования: учебное пособие. Томск: ТУСУР, 2015.
3. Основы программирования микропроцессорных контроллеров в цифровых системах управления технологическими процессами / В.С. Кудряшов, А.В. Иванов, М.В. Алексеев и др. Воронеж: ФГБОУ ВПО «Воронежский государственный университет инженерных технологий», 2014.
4. *Сергеев А.И.* Программирование контроллеров систем автоматизации: Учебное пособие / А.И. Сергеев, А.М. Черноусова, А.С. Русяев. – Оренбург: Оренбургский государственный университет, 2016.
5. *Шишов, О.В.* Элементы систем автоматизации: контроллеры, операторные панели, модули удаленного доступа: Лабораторный практикум Москва: Директ-Медиа, 2015

СОДЕРЖАНИЕ

Практическая работа № 1. Основы проектирования в системе MasterSCADA	3
Практическое занятие №2. Датчики в системе MasterSCADA.....	8
Практическое занятие №3. Программа упаковки 8 бит в байт.....	11
Практическое занятие № 4. Создание проекта в TraceMode	14
Практическое занятие № 5. Реализация логических функций при помощи SCADA-системы TraceMode	22
Практическое занятие № 6. Реализация одноконтурной САР при помощи SCADA-системы TraceMode	26
Практическое занятие № 7. Разработка SCADA-экрана участка промышленного предприятия.....	30
Рекомендуемый библиографический список	31

ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

*Методические указания к практическим работам
для студентов магистратуры направления 15.04.04*

Сост. *А.В. Бойков*

Печатается с оригинал-макета, подготовленного кафедрой
автоматизации технологических процессов и производств

Ответственный за выпуск *А.В. Бойков*

Лицензия ИД № 06517 от 09.01.2002

Подписано к печати 13.09.2019. Формат 60×84/16.
Усл. печ. л. 1,9. Усл.кр.-отт. 1,9. Уч.-изд.л. 1,7. Тираж 50 экз. Заказ 805. С 299.

Санкт-Петербургский горный университет
РИЦ Санкт-Петербургского горного университета
Адрес университета и РИЦ: 199106 Санкт-Петербург, 21-я линия, 2