

ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ, СИСТЕМЫ И СЕТИ

*Методические указания к лабораторным работам
для студентов бакалавриата направления 15.03.04*

**САНКТ-ПЕТЕРБУРГ
2019**

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

Кафедра автоматизации технологических процессов
и производств

ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ, СИСТЕМЫ И СЕТИ

*Методические указания к лабораторным работам
для студентов бакалавриата направления 15.03.04*

САНКТ-ПЕТЕРБУРГ
2019

УДК 004.7 (073)

ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ, СИСТЕМЫ И СЕТИ: Методические указания к лабораторным работам / Санкт-Петербургский горный университет. Сост.: *А.В. Бойков, Н.И. Котелева*. СПб, 2019. 46 с.

Содержат информацию о порядке проведения работ, краткие теоретические сведения, основные требования к содержанию отчетов.

Методические указания предназначены для студентов бакалавриата направления 15.03.04 «Автоматизация технологических процессов и производств».

Научный редактор проф. *В.Ю. Бажин*

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

ЛАБОРАТОРНАЯ РАБОТА № 1. ПНЕВМАТИКА FESTO. РАЗРАБОТКА АЛГОРИТМА УПРАВЛЕНИЯ

Цель работы: изучение работы и принципов функционирования программируемого логического контроллера FESTO FC-20 в совокупности с различными пневмоэлектрическими элементами.

Теоретические сведения

Контроллер FC-20 представляет собой микрокомпьютер для управления производственным оборудованием и позволяет программным способом задавать реакцию оборудования на состояние датчиков, отражающих стадии технологического процесса. Данный вариант контроллера фактически является программируемым реле. Каждый контроллер FC-20 содержит 12 двоичных входов (состояние 1 или 0) и 8 выходов.

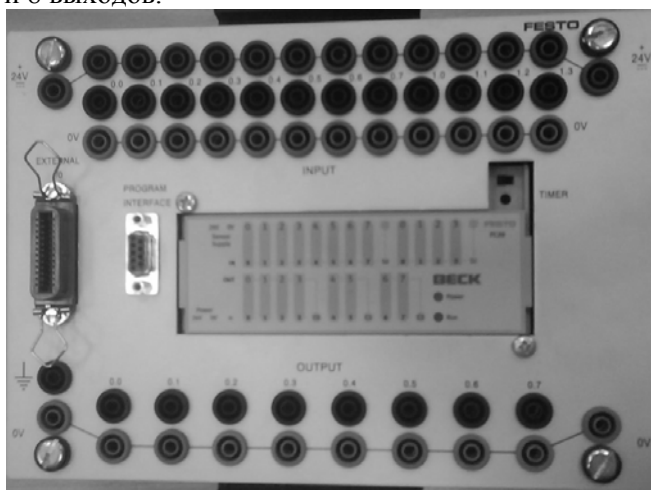


Рис. 1.1. Лабораторный ПЛК Festo FC-20

Лабораторный стенд представляет собой ПЛК FC-20, установленный в специальный внешний корпус, на котором расположены ряды клемм для подачи питания («земля» обозначена синим цветом, сигнал «+24В» - красным), клеммы входов и выходов с ПЛК, а также СОМ-порт для связи контроллера с компьютером при программировании.

Общий вид установки представлен на рисунке 1.2.

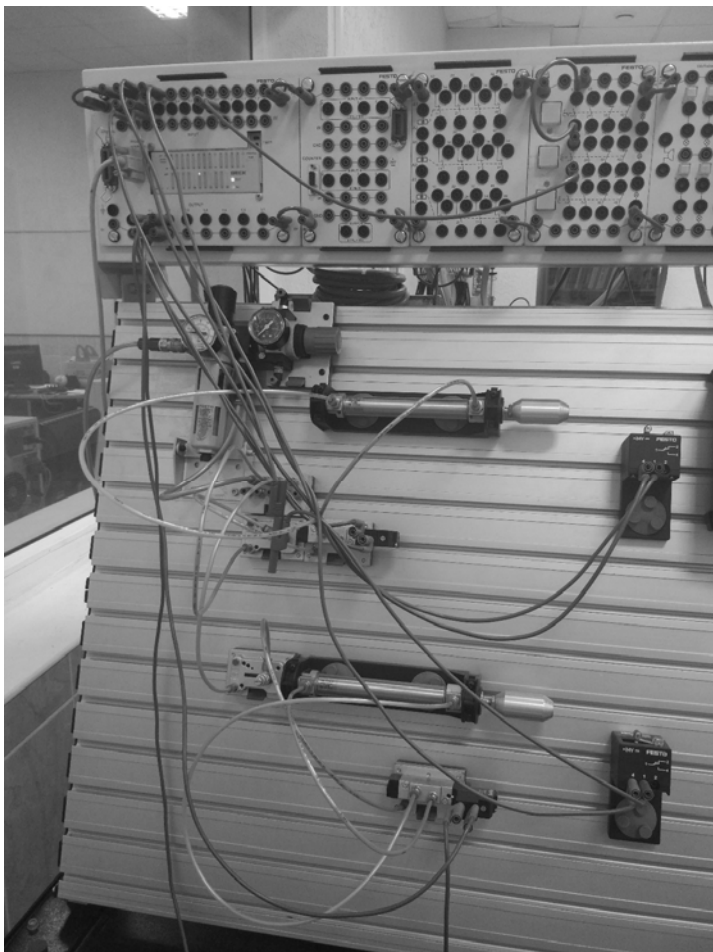


Рис.1.2 Общий вид установки Festo

Описание основных элементов лабораторной установки

1. Компрессорная установка с манометром. Компрессор является источником сжатого воздуха для всех пневматических устройств лабораторной установки. Компрессор подключен к ресиверу, который обеспечивает аккумуляцию сжатого воздуха. На выходе ресивера установлен влагоуловитель с манометром.



Рис. 1.3 Манометр Festo LFR-D-MINI

После манометра сжатый воздух подается на распределительное устройство, и далее на дроссель.

2. Дросселирующее устройство. Дроссель предназначен для коммутации внешних пневматических устройств с распределителем давления.

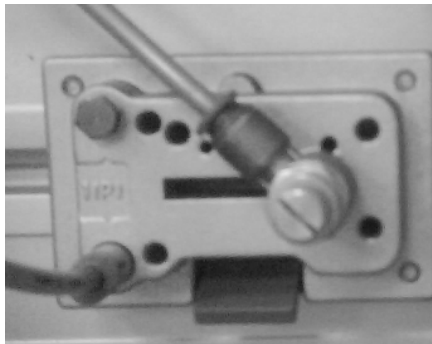


Рис.1.4 Дроссель

3. Реверсивный пневмоцилиндр. Пневмоцилиндр - исполнительное устройство, которое имеет два входа, обеспечивающее как выдвигание штока, так и его обратное движение.

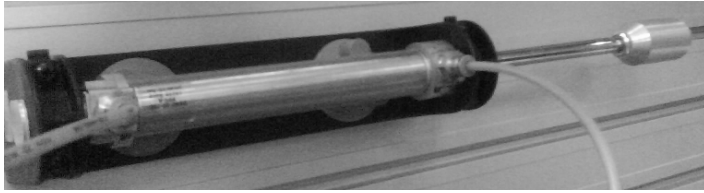


Рис. 1.5 Реверсивный пневмоцилиндр

4. Переключатель. Переключатель обеспечивает подачу электрического сигнала на один из двух выводов. В нормальном положении напряжение подаются с входа 1 на выход клеммы 2, в замкнутом – на 4.



Рис.1.6 Переключатель

Описание принципов программирования контроллера Festo FC-20

1. Открыть меню Пуск - Festo - FST 4.
2. В программе во вкладке Project выбрать New - New Project. В графе Name задаем имя проекту и нажимаем ОК.
3. В открывшемся диалоговом окне Project Setting выбираем контроллер FEC Compact.
4. В окне FST Project нажимаем правой клавишей мыши на Programs - New Program и выбираем Ladder Diagram, стартовая программа должна быть под номером 0.

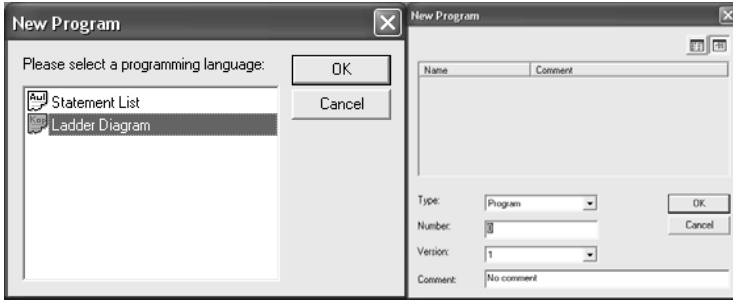




Рис.1.7 Настройки новой программы

В результате появится следующее окно - окно программы, куда можно добавлять необходимые элементы из библиотеки. Программирование схоже с работой в иных приложениях, реализующих работу с LD.

Типовые блоки Ladder Diagram


1. Контакты:

- 1.1  - нормально разомкнутый контакт;
- 1.2  - нормально разомкнутый контакт.

Контактам присваивается имя «I#», где # - номер входа на контроллере. Например, I0.0

2. Катушки:

- 2.1 Катушка без запоминания;
- 2.2 Катушка Set – запоминает в себе пришедшую единицу;
- 2.3 Катушка Reset – сбрасывает запоминание;
- 2.4 Катушка Increment – добавляет единицу в много-битовой операции;
- 2.5 Катушка Decrement – отнимает единицу в много-битовой операции.

Выбор типа катушки осуществляется нажатием на специальный символ . Катушкам присваивается имя «O#», где # - номер входа на контроллере. Например, O0.0

Пример работы алгоритма, использующего нормально разомкнутый контакт I0.1 и катушку без запоминания O0.1 приведен на рисунке 1.8.

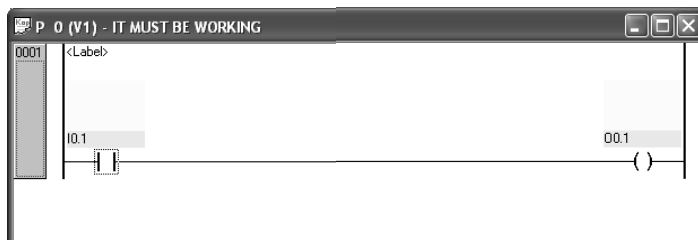


Рис.1.8 Пример программы

Для загрузки программы на контроллер надо нажать по очереди на панели инструментов на Compile Active Module, Make Project, Build Project, Download Project. По окончании загрузки необходимо перейти в режим выполнения (правая кнопка в окне программы – Run Mode) и запустить программу на выполнение – Run. Для выхода в режим редактирования нажмите правую кнопку в окне программы – Editor Mode.

3. Таймеры бывают нескольких видов:

- 3.1 TON – таймер задержки включения;
- 3.2 TOF – таймер задержки выключения;
- 3.2 TP – таймер пульса.

В программном пакете FST 4 работа таймеров реализуется следующим образом:

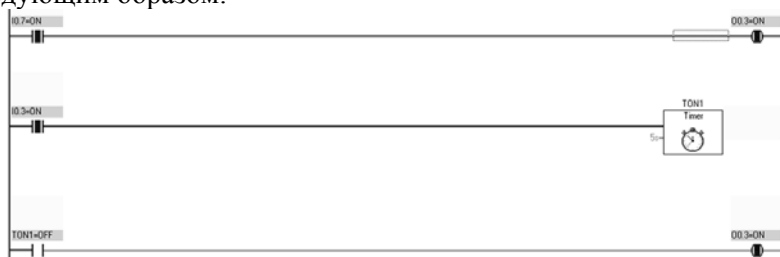


Рис.1.9 Пример работы таймера TON

Что бы задать имя таймера нужно написать его тип и номер таймера, например, TON1. Что бы получить значение с выхода таймера, нужно создать контакт с именем этого таймера.

4. Счетчики бывают двух видов: прямой (считает от 0 до заданного числа) и обратный (считает с заданного числа до 0).

В счетчиках используются 2 входа и 3 операнда.

Входы:

а) Питание/сброс. При подаче единицы сбрасывает значение счетчика.

б) Значение уставки. Сюда записывается значение счетчика, до которого он будет считать.

Операнды:

- 1) C0...C255 – операнд показывающий статус счетчика.
- 2) CP0...CP255 – мульти-битовый операнд, содержащий настройки счетчика (его конечное значение).
- 3) CW0...CW255 – мульти-битовый операнд содержащий текущее значение счетчика.

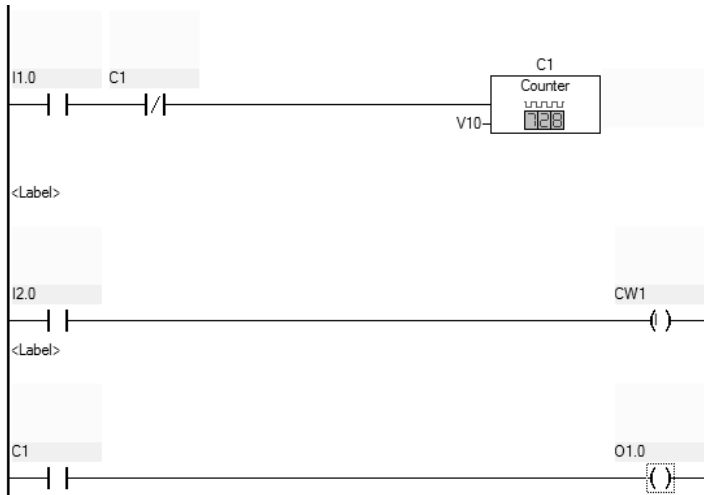


Рис. 1.10 Пример работы счетчика

На рисунке 1.10 показана программа реализации прямого счетчика. При поступлении единицы на контакт 1.0 счетчик C1 запускается. При этом значение 10 сохраняется в CP1 и хранится там, а значение CW1 сбрасывается на 0. Теперь при поступлении единицы на контакт 2.0 к CW1 добавится единица и так будет продолжаться до тех пор, пока CW1 не достигнет 10, после чего на контакт C1 замкнется и на выход 1.0 придет единица.

Задание на лабораторную работу

Пример задания: Реализовать процесс имитации сварки изделия при помощи двух пневмоцилиндров. Первый цилиндр по нажатию на кнопку выдвигает деталь на позицию, при этом включается звуковая и световая сигнализация. После чего происходит 4-х кратное выдвижение поршня номер 2 с задержкой в 1 секунду – имитирующее процесс сварки. Подсчет повторений реализовать с помощью счетчика. По окончании имитации сварки, первый поршень забирает деталь и отключает сигнализацию. После чего процесс готов к повторению в случае повторного нажатия на кнопку.

Отчет по лабораторной работе должен содержать:

- 1) Скриншот и описание работы программы;
- 2) Фотографию реализованной установки;
- 3) Выводы, обобщающие полученные результаты.

ЛАБОРАТОРНАЯ РАБОТА № 2. КОНТРОЛЛЕР АЛЬБАТРОС. КОНФИГУРИРОВАНИЕ И ПРОГРАММИРОВАНИЕ КОНТРОЛЛЕРА

Цель работы: изучение основ создания программ для КПК посредством системы программирования «Softlogic.P9 1», предназначенной для разработки, подготовки и загрузки программ логического управления в модуль процессора МП7/МП9 КПК ГАММА 11.

Теоретические сведения

Контроллер промышленный комбинированный ГАММА-11 имеет модульную структуру и предназначен для построения универсальных информационно-управляющих комплексов, обладающих гибкой структурой организации аналогового и цифрового ввода/вывода с программно-ориентированными исполняемыми функциями. Прибор может работать как автономно (в том числе с местной индикацией измеряемых параметров), так и в составе АСУ ТП совместно с верхним уровнем.

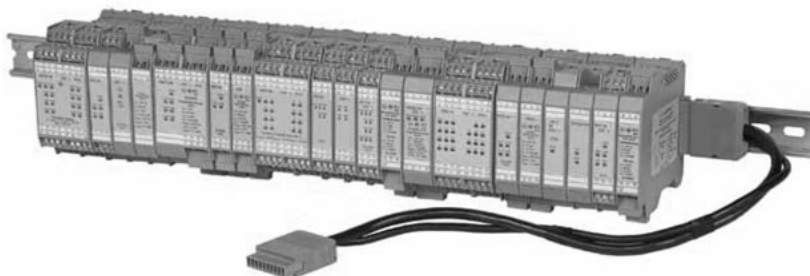


Рис. 2.1 Лабораторный КПК ГАММА 11

Выпускаемые семейства КПК отличаются типом используемого МП и максимальным количеством групп интерфейсных модулей (ГИМ), входящих в состав КПК. В МКПК используется МП7 и одна ГИМ, в СКПК используется МП9 с возможностью подключения до трёх ГИМ.

ГИМ состоит из интерфейсных модулей ввода/вывода (далее “интерфейсные модули”), часть которых имеет взрывозащищенное исполнение, а также блоков питания изолированных БПИЗ (количество БПИЗ определяется суммарной потребляемой мощностью модулей, подключенных к шинам питания ГИМ). Максимальное число интерфейсных модулей в ГИМ - 16. Для обеспечения питания и пе-

передачи данных интерфейсных модулей в ГИМ реализована внутренняя шина. В составе СКПК ГИМ должна комплектоваться модулем интерфейса МИ6, в задачи которого входит гальваническая развязка внутренней шины ГИМ от внешнего канала связи с МП9. В МКПК МП7 подключается непосредственно к внутренней шине ГИМ.

Обмен информацией между модулями, входящими в состав ГИМ, а также между ГИМ и МП осуществляется по последовательному CAN-интерфейсу. Каждый интерфейсный модуль прибора (кроме МСТ) выполнен на основе микроконтроллера (МКР) семейства PICmicro фирмы Microchip Technology Inc. Для реализации CAN-интерфейса МКР имеет встроенный аппаратный CAN-контроллер, согласование уровней сигналов МКР с уровнями на шине CAN-интерфейса осуществляется с помощью микросхем трансиверов.

Центральным узлом МКПК является МП7, построенный на основе МКР фирмы Silicon Laboratories, Inc., являющегося расширением МКР семейства MCS-51 фирмы Intel.

МП7 осуществляет связь с ЭВМ верхнего уровня по интерфейсу RS-485, хранит в своей ЭП параметры настройки и типы интерфейсных модулей прибора, обеспечивает сбор телеметрической информации от интерфейсных модулей и управление выходными сигналами интерфейсных модулей в соответствии с загруженным в МП7 алгоритмом. МП7 программируется посредством среды SoftLogic.

Установка и демонтаж модулей

Установка БПИЗ, МП7 и интерфейсных модулей МКПК осуществляется на один DIN-рельс. Для установки модуля заведите нижний паз узла крепления на DIN-рельс, как показано на рисунке, и поверните корпус в направлении стрелки – при этом модуль должен зафиксироваться на DIN-рельсе.

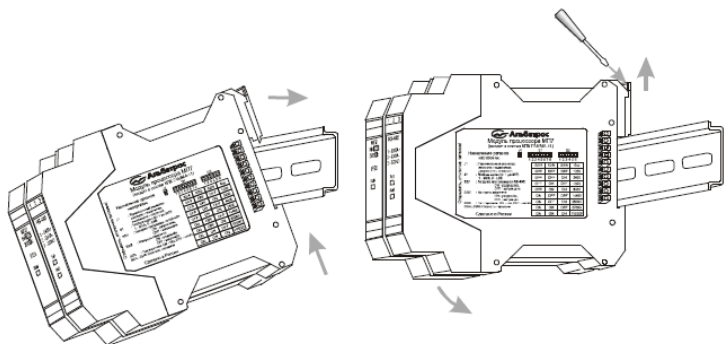


Рис. 2.2 Установка и демонтаж модулей

Подключение контроллера

Соединение контроллера с компьютером осуществляется через интеллектуальный двунаправленный конвертер интерфейсов RS-232 в RS-422/485, который преобразует данные с порта RS-232 (COM-порта) в физический интерфейс RS-422 или RS-485 и обратно.

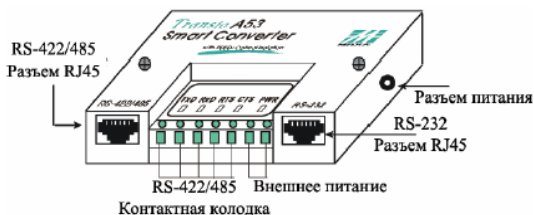


Рис. 2.3 Transio A53 Smart Converter Интеллектуальный двунаправленный конвертер интерфейсов RS-232 в RS-422/485

Настройка модульного процессора и интерфейсных модулей

Извлеките из корпуса блок, образованный ячейками модуля с установленными на них крышками, для чего выполните следующие действия:

- как показано на рисунке 2.4, приложите усилие отверткой соответствующего размера к фиксаторам крышек (маленькие серые стрелки на рисунке, два места сверху модуля и два места снизу

зу) – при освобождении фиксатора крышки слышен характерный щелчок;

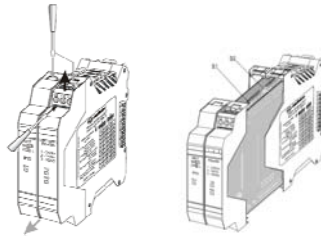


Рис. 2.4 Настройка модульного процессора и интерфейсных модулей

- осторожно выньте блок из корпуса, потянув его в направлении большой серой стрелки рисунка

Для модульного процессора с помощью секций переключателя S2 с номерами от 4 до 6 установите необходимую скорость передачи данных (значения скорости передачи в зависимости от состояния секций переключателя S2 приведены в таблице 2.1), а с номерами 2 и 3 установите наличие контроля паритета и его тип (значения устанавливаемого контроля паритета в зависимости от положений данных секций переключателя S2 приведены в таблице 2.2).

Таблица 2.1

Значения скорости передачи в зависимости от состояния секций переключателя S2

Номер секции переключателя S2 и ее положение			Скорость передачи, бод
4	5	6	
OFF	OFF	OFF	1200
OFF	OFF	ON	2400
OFF	ON	OFF	4800
OFF	ON	ON	9600
ON	OFF	OFF	14400
ON	OFF	ON	28800
ON	ON	OFF	57600
ON	ON	ON	115200

Примечание – При выпуске модуля из производства установлена скорость передачи 14400 бод.

Таблица 2.2

Значения устанавливаемого контроля паритета в зависимости от состояния секций переключателя S2

Номер секции	Положение секции	Значение
2	OFF	Контроль паритета запрещен
	ON	Контроль паритета разрешен
3	OFF	Контроль по нечетности
	ON	Контроль по четности

Примечание – При выпуске модуля из производства контроль паритета разрешен и задан контроль по четности.

Для каждого интерфейсного модуля необходимо установить уникальный позиционный номер в составе КПК. Значения позиционных номеров в зависимости от состояния секций переключателя S1 приведены в таблице 2.3.

Таблица 2.3

Значения позиционных номеров в зависимости от состояния секций переключателя S1

Номер секции переключателя S1 и ее положение				Позиционный номер
1	2	3	4	
OFF	OFF	OFF	OFF	1
ON	OFF	OFF	OFF	2
OFF	ON	OFF	OFF	3
ON	ON	OFF	OFF	4
OFF	OFF	ON	OFF	5
ON	OFF	ON	OFF	6
OFF	ON	ON	OFF	7
ON	ON	ON	OFF	8
OFF	OFF	OFF	ON	9
ON	OFF	OFF	ON	10
OFF	ON	OFF	ON	11
ON	ON	OFF	ON	12
OFF	OFF	ON	ON	13
ON	OFF	ON	ON	14
OFF	ON	ON	ON	15
ON	ON	ON	ON	16

Примечание – При выпуске модуля из производства установлено значение позиционного номера равное единице.

После проверки положений и, если необходимо, установки всех описанных механических органов настройки модуля, осторожно вставьте блок ячеек модуля по направляющим в корпус до характерного щелчка.

Перед началом работы необходимо удостовериться в работоспособности оборудования. Для этого открыть программу Тестирование КПК ГАММА-11. Во вкладке «Состояние системы» отобразятся подключенные модули и их состояние. Номер слота модуля должен совпадать с позиционным номером, который задается с помощью четырехразрядного механического переключателя S1, расположенного на плате одной из ячеек модуля.

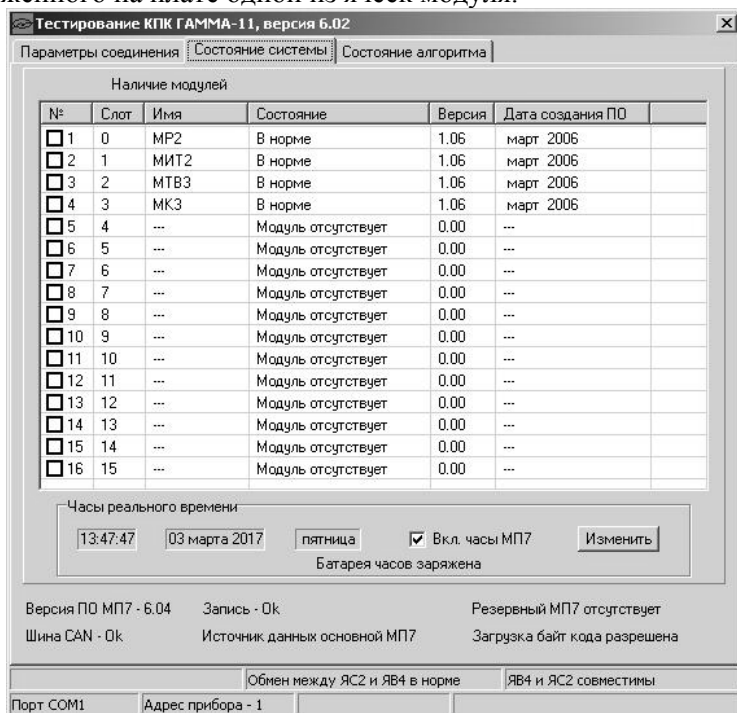



Рис. 2.5 Тестирование КПК ГАММА-11

Кликаем на «квадрат», соответствующий имени МК3. Появляется окно, имитирующее работу МК3. Для проверки работоспособности во вкладке «Групповая установка ключей» ставим галочку

напротив, например, ключа 5 и 13, и нажимаем кнопку записать . Если все было настроено правильно, на МКЗ загорятся два светодиода, которые будут соответствовать замкнутому состоянию ключей 5 и 13.

Запускаем редактор FBD «Альбатрос Гамма.FBD». Реализуем на рабочей области данную программу, используя элементы FBD. При программировании используются наборы библиотечных блоков и собственные блоки, также написанные на FBD или других языках МЭК 61131-3. Блок (элемент) — это подпрограмма, функция или функциональный блок (И, ИЛИ, НЕ, триггеры, таймеры, счётчики, блоки обработки аналогового сигнала, математические операции). Каждому элементу диаграммы необходимо присвоить определенные параметры. Для этого необходимо навести курсор «мыши» на элемент и нажать правую кнопку. На экране появится окно с параметрами элемента. Количество параметров и их значения зависят от типа выбранного элемента.

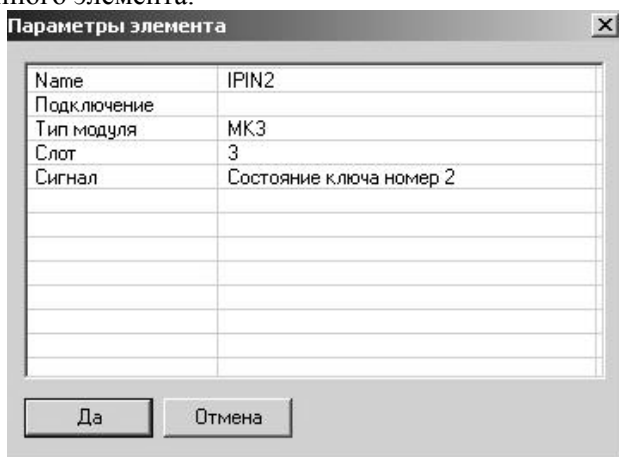


Рис. 2.6 Параметры элемента

После реализации программы необходимо её скомпилировать, для этого в пункте меню «Файл» нажимаем «Компилировать файл». Затем загрузить программу в МП7, для этого в пункте меню «Файл» нажимаем «Загрузить». Откроется окно «Параметры загрузки». Установить приведенные ниже параметры. Нажать «ОК».

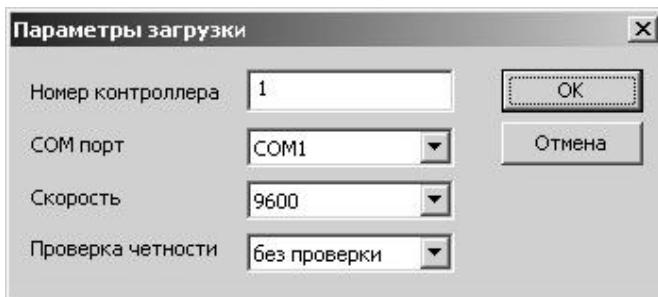


Рис. 2.7 Параметры загрузки

После успешной загрузки проекта в контроллер появится окно программы загрузчика. Программа загрузки после запуска загружает байт-код в КПК и выводит информацию о ходе загрузки. Кроме того, программа выводит информацию о состоянии и ошибках при выполнении алгоритма после загрузки, а также позволяет контролировать работу алгоритма в КПК (стартовать, останавливать, перезапускать).

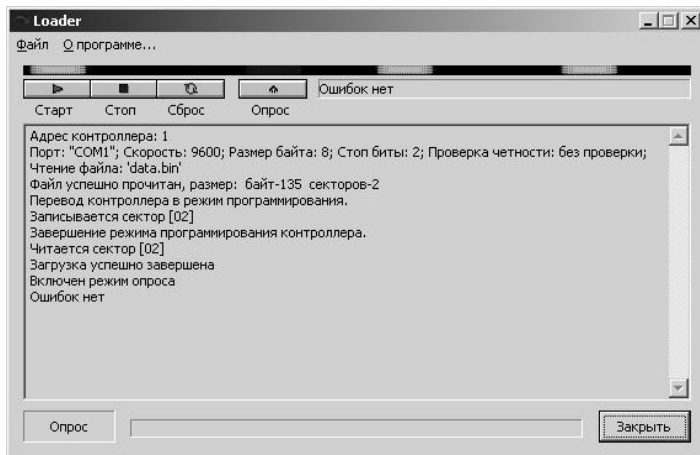


Рис. 2.8 Окно загрузки

По окончании загрузки программы в контроллер закрываем окна всех программ и снова открываем программу тестирования КПК ГАММА-11 и проверяем работоспособность загруженной программы.

Задание на лабораторную работу

Реализовать алгоритм программы, заданной преподавателем.

Пример задания изображен на рисунке 2.9:

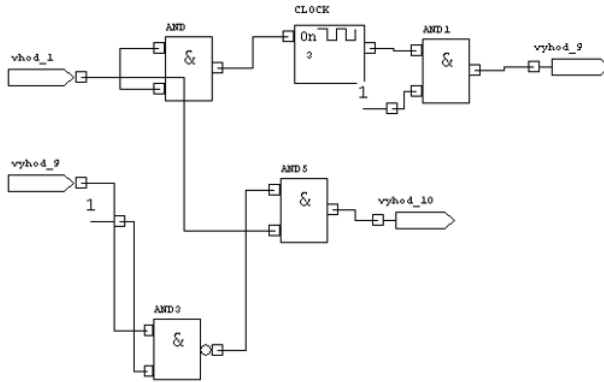


Рис. 2.9 Пример программы

Отчет по лабораторной работе должен содержать:

- 1) Скриншот и описание работы программы;
- 2) Фотографию реализованной установки;
- 3) Выводы, обобщающие полученные результаты.

ЛАБОРАТОРНАЯ РАБОТА № 3. ОБМЕН ДАННЫМИ МЕЖДУ КОНТРОЛЛЕРОМ AUTOLOG И SCADA СИСТЕМОЙ IFIX

Цель работы: освоить методы настройки драйвера Modbus RS-232 для коммуникации SCADA системы iFIX с ПЛК Autolog 64 по протоколу Modbus через интерфейс RS-232, язык программирования ALPro контроллера AutoLog и научиться практически применять для программирования контроллера, установленного на стенде.

Теоретические сведения

Программируемые логические контроллеры "Autolog" фирмы "FF – Automation" (Финляндия) применяются в информационно-вычислительных системах основных отраслей промышленности для преобразования аналоговых и дискретных электрических величин в цифровой код, программной обработки и накопления результатов преобразования и их представления в аналоговой или цифровой форме, удобной для потребителя и для передачи по каналам связи. Для данных контроллеров в качестве языка программирования используется язык программирования ALPro.

Действия на ПЛК Autolog.

1. Включаем питание контроллера.
2. Включаем персональный компьютер и загружаем операционную систему MS DOS.

3. Порядок входа в ALPro:

C:\>dir ↵; C:\>dir /w↵; далее находим frirus и выполняем следующую команду: C:\>frirus↵; C:\>cd dn↵; C:\>dir /w↵; C:\>dn↵

4. Работа в DOS NAVIGATOR. Создаем свою папку (3 лат. буквы) с помощью кнопки F7; находим в каталоге файл alpro.exe и запускаем его.

5. Работа в ALPro:

Запускаем *ALPro v.2.21* → *AL 32/36/64* и проверяем, что *PLS is connected*.

Существует две основные команды: **STR** и **EQ**

Пример:

STR I 0 – данная команда считывает в аккумулятор с нулевого входа с адресом 0;

EQ O 2 – данная команда записывает на аккумулятор с выхода с адресом 2.

Для использования ПЛК Autolog в сети Modbus по витой паре требуется задать в контроллере специальные ячейки оперативной памяти, где необходимо задать параметры COM порта (RS-232) и задать уникальный адрес контроллера в сети Modbus. Для этого:

1. На языке алрго создаем программу, которая будет иметь следующий вид:

```
STR R C 4  
EQ R M 327  
STR R C 3  
EQ R M 345  
STR R C 1  
EQ R M 363  
STOP
```

STR R C 4 и *EQ R M 327* - запись в ячейку 327 константы 4, говорит о том, что связь реализуется через протокол MODBUS

STR R C 3 и *EQ R M 345* - запись в ячейку 345 константы 3, говорит о том что связь производится на скорости 9600



STR R C 1 и *EQ R M 363* - запись в ячейку 363 константы 1, говорит о том что номер контроллера в сети 1.

Чтобы откомпилировать программу, необходимо нажать **Compile**→**Compile only**. Для выполнения программы используются следующие функции: **Transfer**→**Write program to PLS** и **Transfer**→**Execute PLS program**.

Для наглядного просмотра используется функция **Debug** и **Watch**→**Watch Loop**. При необходимости можно добавить или удалить переменные, используемые при отладке с помощью функций Add/Remove Variable.

Действия на компьютере с ОС Windows, соединенного с контроллером Autolog с помощью витой пары через порт COM1, в SCADA системе iFIX.

Необходимо запустить iFIX 4.5 в ДЕМО-режиме, после чего вызвать подпрограмму iFIX Workspace, а в ней через панель «При-

ложения Toolbar» утилиту системного конфигурирования (), администратор базы данных ().

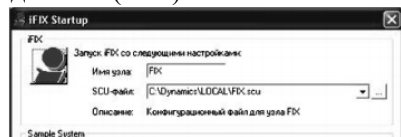


Рис. 3.1 Запуск iFIX в DEMO-режиме

1) Действия в утилите системного конфигурирования iFIX



Рис. 3.2 Утилита системного конфигурирования iFIX

Из списка драйверов выбрать драйвер Modbus – условное обозначение **MB1**, затем ввести параметры соединения в окне настройки драйвера (рисунок 3.3).

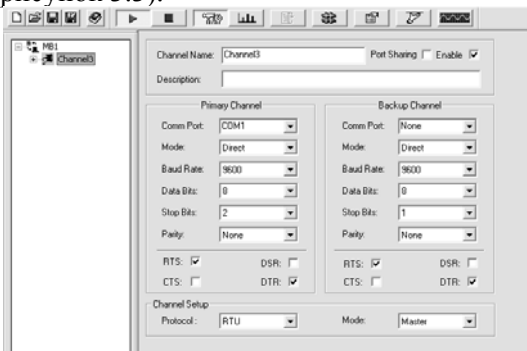


Рис. 3.3 Параметры настройки драйвера MB1

Создать/подключить новое устройство и указать его имя (например, Device3), оно в дальнейшем будет использоваться при обращении к адресам внутренних переменных контроллера (рисунок 3.4).

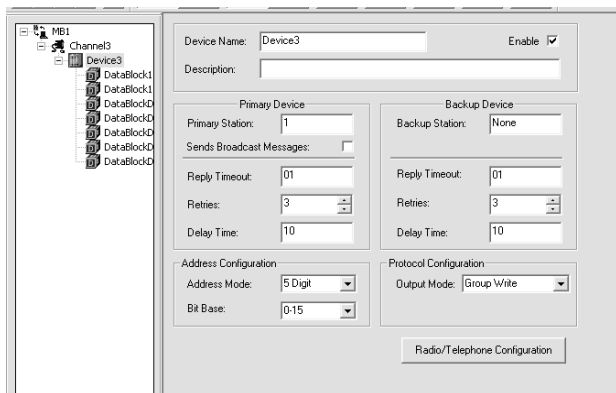


Рис. 3.4 Создание/подключение нового устройства

Для обозначения адресов датчиков и исполнительных механизмов, подключенных к модулям ввода-вывода контроллера по протоколу Modbus, используется специальная система адресов, где первая цифра в адресе обозначает тип модуля, а остальные цифры - порядковый номер точки в общем поле номеров точек для однотипных модулей. Например, обозначение адреса для входа/выхода номер 1 для различных модулей представлено на следующей таблице 3.1.

Таблица 3.1

Обозначение адреса для входа/выхода номер 1 для различных модулей

Тип адреса	Полная форма представления адреса	Упрощенная форма представления адреса
Дискретный выход	00001	0:1
Дискретный вход	10001	1:1
Аналоговый вход	30001	3:1
Аналоговый выход, или регистр. ячейка памяти	40001	4:1

2) Действия в Администраторе базы данных iFIX.

В Администраторе базы данных создать и сохранить свою базу реального времени для iFIX. Данные реального времени ведутся в базе данных iFIX, которая состоит из функциональных блоков раз-

личных типов. Состав и тип блоков зависит от технологического процесса, под который разрабатывается SCADA-система. Блоки могут выполнять следующие функции:

- принимать значения из других блоков, из OPC-серверов либо непосредственно из драйверов ввода/вывода;
- манипулировать значениями в соответствии со своей конфигурацией;
- сравнивать вводимые значения с заранее заданными пределами;
- масштабировать производственные величины в заданный диапазон;
- выполнять вычисления;
- выводить значения обратно в драйвер ввода/вывода или OPC-сервер.

Блоки подразделяются на первичные и вторичные.

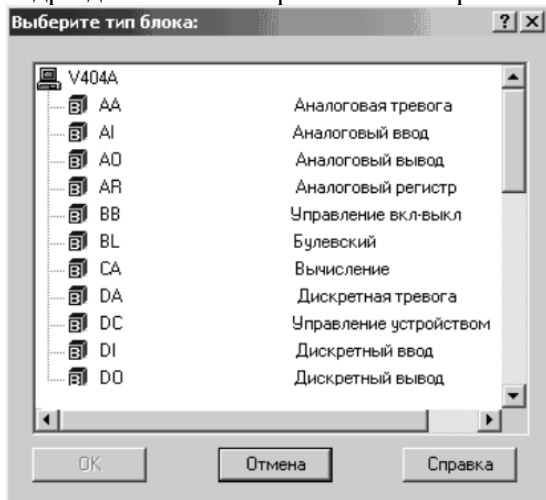


Рис. 3.5 Окно «Выбор типа блока»

Данные поступают в первичные блоки. К ним относятся, например, блоки типа аналоговый и дискретный ввод (AI и DI), аналоговая и дискретная тревога (AA и DA), аналоговый и дискретный регистры (AR и DR). Сканирование первичных блоков осуществляется по времени или по изменениям в принимаемых данных. Для

приема данных реального времени первичные блоки настраиваются на какой-либо драйвер реального времени (например, MB1 – Modbus).

Чтобы каждый первичный блок в базе данных получил данные, необходимо установить связь с точками ввода/вывода через драйвер ввода/вывода. В iFIX встроены два имитационных драйвера: SIM и SIM2, которые можно использовать для тестирования цепочек блоков до соединения с реальным оборудованием. Имитационный драйвер – эта матрица адресов. Если один блок пишет по заданному адресу, другие могут считывать значения по такому же адресу.

В регистрах SIM генерируется определенная случайная или заданная последовательность чисел. Для выбора регистра необходимо ввести соответствующее двухбуквенное обозначение в поле «Адрес ввода/вывода»: RA, RB, RC, RD, RE, RF, RG, RH, RI, RJ, RK, RX, RY, RZ.

Примеры регистров данных:

1. SIM регистр RA ступенчато изменяет значение от 0 до 100% диапазона ИНВ со скоростью, управляемой регистром RY. Значения регистра RA могут только читаться.

2. SIM регистр RD генерирует синусоидальную волну от 0 до 100% диапазона ИНВ со скоростью, управляемой регистром RY. Значения регистра RD могут только читаться. Для включения генерации значений регистра RD необходимо использовать регистр RX.

3. SIM регистр RG генерирует случайные величины от 25 до 75% диапазона ИНВ. Значения регистра RG могут только читаться. Для включения генерации значений регистра RG необходимо использовать регистр RX.

4. SIM регистр RH производит ступенчатое изменение значения от 0 до 100% диапазона ИНВ и снова до 0% со скоростью, управляемой регистром RJ. Значения регистра RH могут только читаться. Для включения генерации значений регистра RH необходимо использовать регистр RK.

5. SIM регистр RX включает и отключает генерацию значений в других регистрах. В процессе работы можно ввести 1 для включения всех регистров и 0 – для отключения всех регистров.

	Имя тега	Тип	Описание	Период	Устр.	Адр. I/O	Тек. знач.
1	POLOGENIE	AO		—	SIM	2	0,00
2	SPEED	AO		—	SIM	1	3,00
3	CALC	CA		—	—	—	0,00
4	SYNCH	AI		1	SIM	0	0,00
5	DI144322	DI		1	MB1	Device3:10001	CLOSE
6	DO17	DO	1	—	MB1	Device3:00017	OPEN

Рис. 3.6 Пример сформированной базы данных

3) Действия в редакторе операторского экрана iFIX WorkSpace.

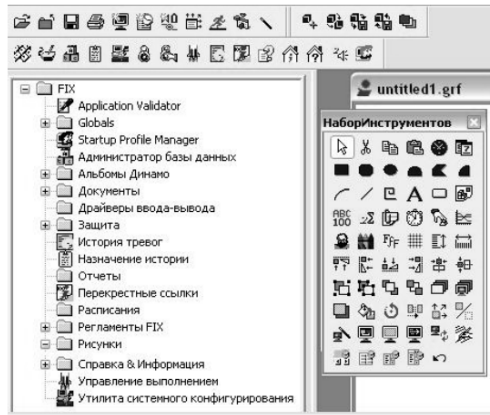


Рис. 3.7 Окно WorkSpace

Основным инструментом для работы со SCADA-системой iFIX является среда разработки, конфигурирования и выполнения Intellution WorkSpace. Окно WorkSpace (рис. 3.7) можно условно разделить на три основные зоны: 1) панель инструментов (вверху); 2) системное дерево (слева); 3) рабочую зону (справа). WorkSpace совмещает в себе две среды: среду конфигурирования (среду разработчика) и среду выполнения (среду пользователя). В среде конфигурирования разработчик создает экранные формы, т. е. рисунки, которые наглядно отображают аппараты, трубопроводы и другие компоненты технологической установки. Среда выполнения предназначена для непосредственной визуализации протекания процесса.

Зона «Системное дерево». Отображает в иерархическом виде файлы, имеющиеся на локальном узле. При первом запуске

WorkSpace системное дерево показывает имя локального узла на вершине дерева, а также папки и приложения.

Зона «Рабочая зона». Предназначена для рисования и создания объектов, отображающих технологические аппараты, трубопроводы, текущие значения параметров процесса и др. Эта зона сразу после запуска содержит два окна: окно рисования (открывается с названием «untitled1») и окно «Набор Инструментов».

С помощью набора инструментов можно добавлять в рисунок различные объекты. При наведении курсора на любой элемент в окне «**Набор Инструментов**» (рис. 3.7) выплывает подсказка, которая содержит краткое описание добавляемого объекта. Объект добавляется в рисунок нажатием на обозначающую его кнопку. Некоторые формы рисуются путем задания с помощью мыши последовательности опорных точек. После добавления фигуры в рисунок ее можно преобразовывать, т. е. менять ее цвет, размер, стиль заполнения переднего плана. Добавленному в рисунок объекту автоматически присваивается название, с которым этот объект отображается в системном дереве. Чтобы изменить название объекта, нужно либо открыть окно «Properties», как было указано выше, и изменить строку «Name», либо произвести двойной щелчок по объекту и в открывшемся окне «Диалог Основные Анимации» (рис. 3.7) в поле «Имя объекта» ввести желаемое имя. При наименовании объектов следует помнить, что их названия не должны совпадать с названием рисунка.

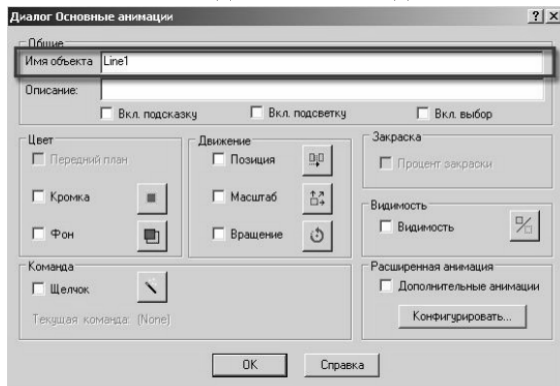


Рис. 3.8 Окно WorkSpace

Задание на лабораторную работу

Реализовать алгоритм программы, заданной преподавателем.

Пример задания: Создать экранную форму в iFIX объекта автоматизации – три резервуара смешения, перемешиваемых с помощью мешалок. Возможный вид экранной формы представлен на рисунке 3.9.

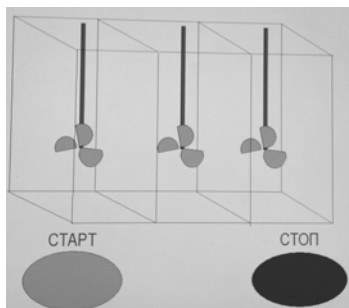


Рис. 3.9 Пример Операторское окно в режиме редактирования

Отчет по лабораторной работе должен содержать:

- 1) Скриншот и описание работы программы;
- 2) Фотографию реализованной установки;
- 3) Выводы, обобщающие полученные результаты.

ЛАБОРАТОРНАЯ РАБОТА № 4. ПРОГРАММИРОВАНИЕ СЕНСОРНОЙ ПАНЕЛИ MAGELIS

Цель работы: ознакомиться с принципами программирования сенсорных панелей Magelis. Осуществить обмен данными между контроллером и сенсорной панелью.

Теоретические сведения

В лабораторной работе используется сенсорная графическая панель Magelis XBTGT4340 французского производителя Schneider Electric. Назначение данной графической панели – индикация текстовых сообщений, графических объектов, управление и параметрирование данных. Перед началом работы необходимо подать питание на сенсорную панель (24 В) и подключить ее к сети посредством COM-порта. Программирование осуществляется в среде Vijeo-Designer V5.0 - программное обеспечение для конфигурирования панелей оператора.

1) Запустить программу Vijeo-Designer V5.0 и в появившемся окне выбрать пункт – создание нового проекта (Create new Project) и нажать далее. В следующем окне необходимо выбрать имя проекта и можно сделать краткое описание. В любом проекте может быть использовано несколько устройств, поэтому предлагается ввести имя панели (Target Name). Далее надо выбрать тип и модель сенсорной панели (рис. 4.1).

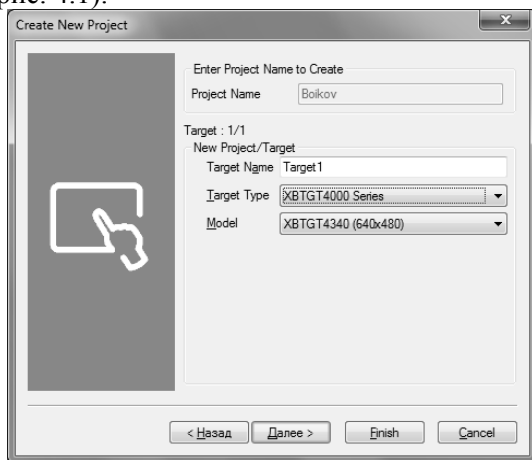


Рис. 4.1 Окно выбора серии и модели панели

2) Закончив с настройками проекта, необходимо настроить соединение с панелью. Для этого в окне Navigation левой кнопкой мыши 1 раз нажать на название устройства Target1, а затем в окне Property Inspector вкладке Download ввести настройки соединения с панелью (рис. 4.2).

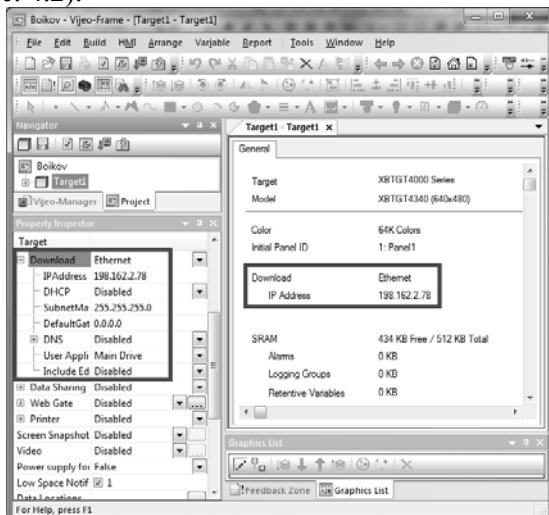


Рис. 4.2 Настройки соединения с панелью

3) Соединение с контроллером настраивается в окне Navigation выбираем пункт IO Manager → New Driver (рис. 4.3)

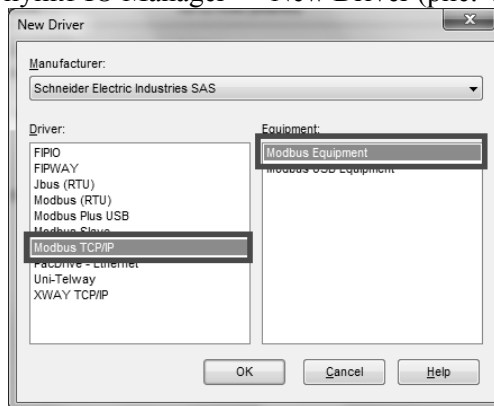


Рис. 4.3 Выбор типа соединения

Далее в IO Manager → ModbusTCP/IP01 необходимо проверить сетевые настройки панели (рис. 4.4).

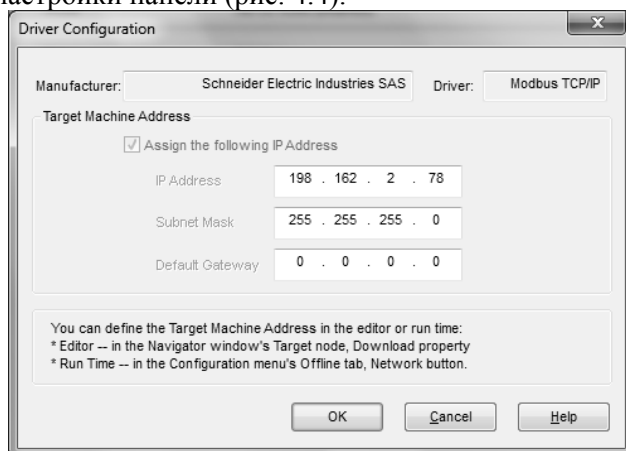


Рис. 4.4 Сетевые настройки панели

Далее в IO Manager → Equipment01 установить настройки контроллера (рис. 4.5): IP адрес контроллера и параметры IEC61131 Syntax.

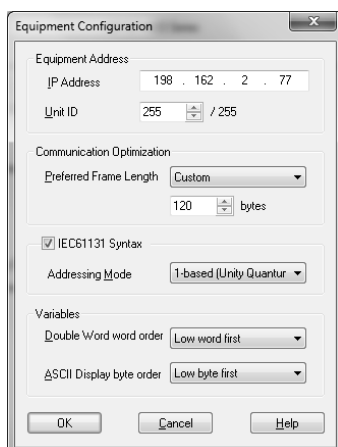


Рис. 4.5 Пример настройки соединения с контроллером

4) Теперь можно приступить к созданию визуального оформления панели.

Создадим две переменные: одну для считывания с контроллера, а другую для записи в него. Для этого в окне Target1 → Variable добавить новые переменные (рис.4.6).

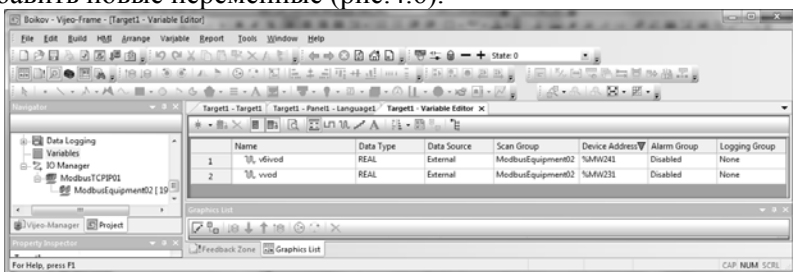


Рис. 4.6 Создание переменных

Визуализация переменных выполняется в окне Graphic Panels → Base Panels → Panel1 с помощью инструмента цифровой дисплей. При создании визуализации возникает окно настройки цифрового дисплея, где необходимо выбрать переменную и вид ее отображения (рис. 4.7). Если переменную можно изменять с панели необходимо поставить галочку – Enable Input Mode.

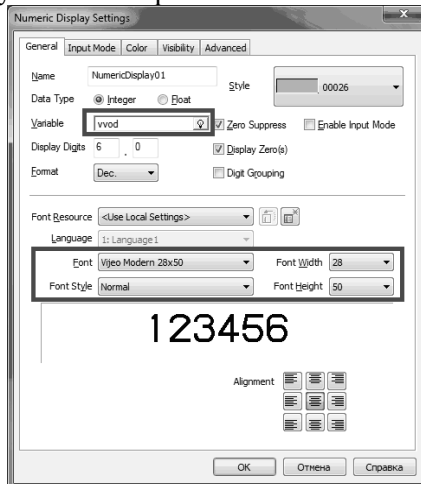


Рис. 4.7 Настройки числового дисплея

Таким образом, итоговый вид панели может выглядеть следующим образом (рис. 4.8).



Рис. 4.8 Итоговый вид панели

Прежде чем загрузить экранную форму на панель, проект надо проверить на ошибки с помощью Build → Validate Target и скомпилировать для загрузки Build → Build Target, после чего проект можно загрузить на панель Build → Download Target.

Задание на лабораторную работу

Если после перезагрузки панели появились два поля для ввода/вывода, то для проверки работоспособности создать проект в Unity Pro по заданию преподавателя.

Пример задания: Создать экранную форму и алгоритм расчета уровня воды в баке на основе изменения разности расходов: задаваемого входного расхода и фиксированного выходного. Результаты вывести на сенсорную панель. Возможный вид экранной формы представлен на рис. 4.9.

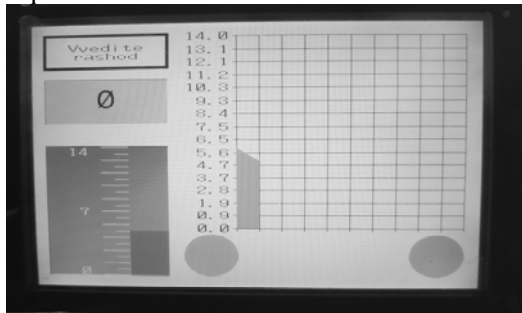


Рис. 4.9 Пример визуализации экранной формы

Отчет по лабораторной работе должен содержать:

- 1) Скриншот и описание работы программы;
- 2) Фотографию дисплея и программы контроллера в процессе выполнения;
- 3) Выводы, обобщающие полученные результаты.

ЛАБОРАТОРНАЯ РАБОТА № 5. ОБМЕН ДАННЫМИ МЕЖДУ КОНТРОЛЛЕРАМИ MODICON

Цель работы: осуществить обмен данными между контроллерами Modicon-Modicon.

Теоретические сведения

Процессорные модули платформы автоматизации Modicon M340 предназначены для комплексного управления станцией ПЛК, в состав которой могут входить дискретные модули входов-выходов, аналоговые модули входов-выходов, счетные модули. Модули размещаются на одном шасси.

Программирование контроллеров осуществляется в программе UnityPro XL4.1. В UnityPro XL4.1 под каждый контроллер необходимо создать новый проект, модель контроллера Modicon необходимо посмотреть непосредственно на контроллере. В разделе проекта **Configuration → 0: PLC Bus** также необходимо собрать корзину в соответствии с реальным оборудованием.

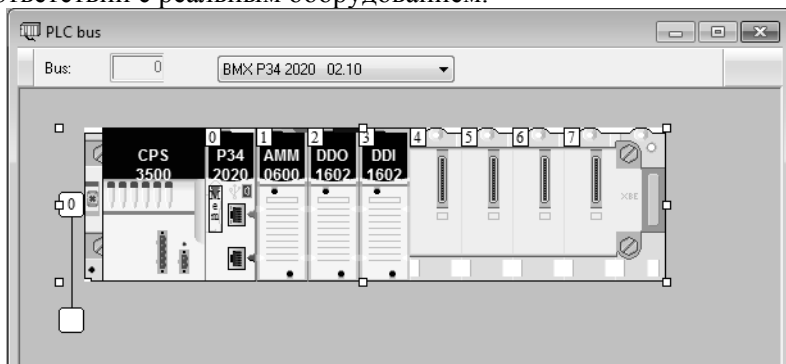


Рис.5.1 Пример корзины контроллера

Процессорные модули BMX P34 2020/2030 поддерживают коммуникацию по сети Ethernet. Данные процессорные модули имеют встроенный коммуникационный канал, предназначенный для коммуникации по сети Ethernet. Сетевые настройки настраиваются в разделе проекта **Communication → Networks**, где двойным щелчком создаем новую сеть с Ethernet (название Ethernet_1 появится автоматически, при желании его можно заменить).

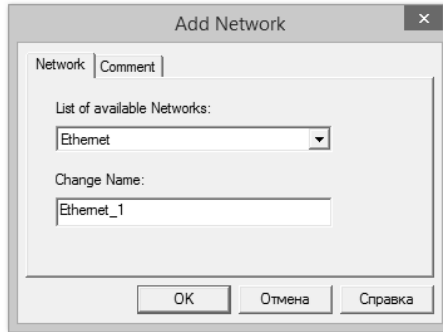


Рис. 5.2 Создание новой сети Ethernet

После этого настраиваем подключение Ethernet. Для этого слева двойным щелчком выбираем **Communication** → **Networks** → **Ethernet_1**. В открывшемся окне необходимо: в поле Model Family выбрать версию и модель своего контроллера (CPU) и установить IP-адрес контроллера (написан на специальной наклейке на боковой стороне контроллера) и маску подсети (ее можно узнать, вызвав командную строку **cmd** и выполнив команду **ipconfig**).

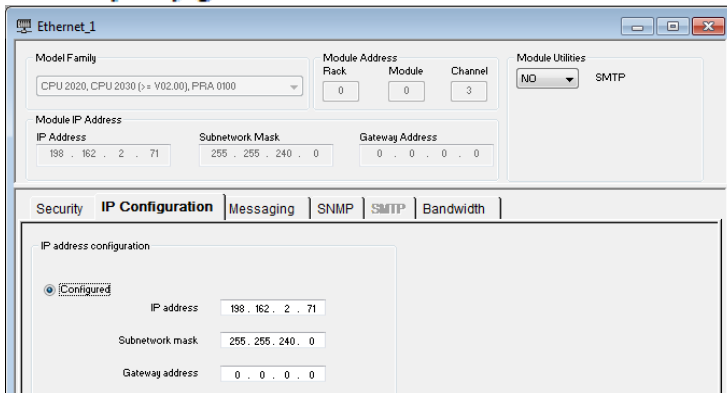


Рис. 5.3 Окно настройки сети Ethernet

Соединение Ethernet необходимо также подключить к самому контроллеру. Для этого двойным щелчком выбираем **Configuration** → **0: PLC Bus** → **BMX X**. В открывшейся корзине необ-

ходимо найти и дважды нажать на Ethernet разъем RJ45, расположенный непосредственно на контроллере. В открывшемся окне необходимо выбрать *Channel 3* и установить следующие параметры: **Function — ETH TCP IP, Net Link — Ethernet_1**.

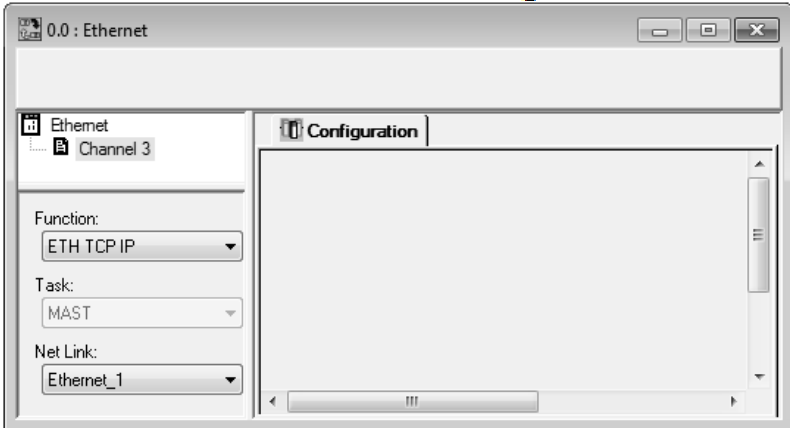


Рис. 5.4 Подключение сети Ethernet в контроллере

В разделе **Program → Tasks → MAST → Sections** можно создать два программы на языке **FBD**.

Первая программа — мультивибратор, функциональная блок-диаграмма которого выглядит следующим образом:

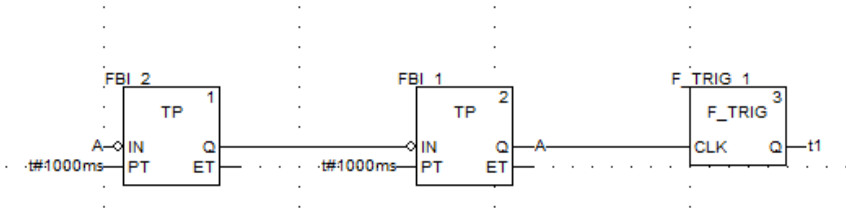


Рис. 5.5 Программа мультивибратора

Данный мультивибратор необходим лишь для создания импульсов тактирования процессов записи и чтения, период импульсов равен 2 секундам. На выходе данного мультивибратора формируется переменная **t1**, которая является результатом его работы.

Вторая программа — блок обмена, функциональная блок-диаграмма которого выглядит следующим образом:

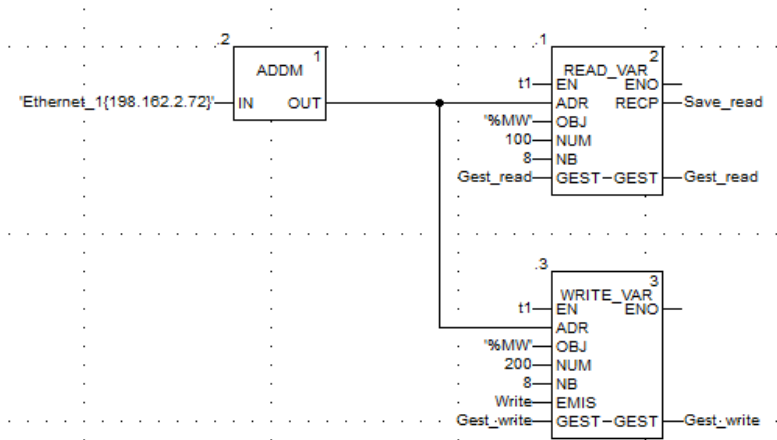


Рис. 5.6 Программа обмена между контроллерами

Рассмотрим подробно каждый из трех блоков:

Блок **READ_VAR**.

- **EN** — контроль работы коммуникации. Если не поставить галочку в настройках данного блока (**ПКМ** → **Properties** → **Show EN/ENO**), то по умолчанию при каждом сканировании контроллера будет выполняться функция **READ**. При поступлении сигнала (1) на этот вход, выполняется функция **READ**, то есть, мы имеем возможность контролировать процесс работы блока.

- **ADR** — адрес удаленного устройства. По умолчанию адрес формируется массивом, состоящим из 7 элементов (для контроллеров Modicon) и из 5 элементов (для контроллеров Premium). Формат данного массива сложен для формирования, поэтому для данного входа используется дополнительный блок (его входы и выходы описаны ниже), который преобразовывает понятный для восприятия адрес в массив, необходимый для блока **READ**. Для Modicon эта функция — **ADDM**, для Premium — **ADDR**.

- **OBJ** — тип данных, который нужно считать. Это может быть либо внутреннее слово (%MW), либо внутренний бит (%M).

- **NUM** — адрес первого элемента, который будем считывать.
- **NB** — количество элементов, которые будем считывать.
- **GEST** — массив, в который заносится служебная информация. Там можно отследить ошибки, активность коммуникации, таймауты и многое другое. Обо всем можно прочитать в справке.
 - **RECP** — область памяти, в которую будут сохраняться данные, которые считываются с удаленного узла. Данный массив должен быть того же размера, что и массив, описанный на входе этого блока (см. п. **OBJ**, **NUM** и **NB**)
 - **GEST** — тот же массив, в который заносится служебная информация.

Блок ADDM.

- **IN** — адрес удаленного IP-узла. Данный адрес можно задать двумя способами: через имя сети (**Ethernet_1**) или же с помощью топологического адреса на корзине. Первый способ намного проще и понятнее, поэтому воспользуемся именно им.
- **OUT** — преобразованный адрес удаленного IP-узла. Этот выход просто подается на вход **ADR** блока **READ**

Блок WRITE_VAR.

- **EN** — контроль работы коммуникации. Аналогично блоку **READ_VAR**.
- **ADR** — адрес удаленного устройства. Аналогично блоку **READ_VAR**.
- **OBJ** — тип данных, с которым будем работать. Это может быть либо внутреннее слово (%MW), либо внутренний бит (%M).
- **NUM** — адрес первого элемента, который будем отправлять.
- **NB** — количество элементов, которые будем отправлять.
- **EMIS** — массив, в который вводится отправляемая информация.
- **GEST** — тот же массив, в который заносится служебная информация.

По всем переменным можно создать анимационную таблицу, чтобы отслеживать работу обмена данными. Для этого в разделе **Variables & FB Types → Elementary Variables** выделить все переменные (Ctrl+A) и либо правой клавишей мыши (Initialize New Animation Table), либо сочетанием Ctrl+Shift+T создаем новую анимационную таблицу. Для удобства можно открыть колонку Address (**ПКМ → Customize Columns → Address**).

Name	Value	Type	Comment	Address
A		BOOL		
Gest_read		ARRAY[0..3] OF INT		%MW150
Gest_read[0]		INT		%MW150
Gest_read[1]		INT		%MW151
Gest_read[2]		INT		%MW152
Gest_read[3]		INT		%MW153
Gest_write		ARRAY[0..3] OF INT		%MW170
Gest_write[0]		INT		%MW170
Gest_write[1]		INT		%MW171
Gest_write[2]		INT		%MW172
Gest_write[3]		INT		%MW173
Save_read		ARRAY[0..7] OF INT		%MW110
Save_read[0]		INT		%MW110
Save_read[1]		INT		%MW111
Save_read[2]		INT		%MW112
Save_read[3]		INT		%MW113
Save_read[4]		INT		%MW114

Рис 5.7 Пример анимационной таблицы

Готовый проект необходимо проверить на ошибки (*Build – Analysis*) и скомпилировать (*Build – Build Changes*), после чего загрузить на контроллер с помощью следующей последовательности действий *PLC-Connect* и *PLC-Transfer Project to PLC* и *PLC - Run*, предварительно убедитесь в правильности настроек подключения в разделе *PLC-Standard Mode* и *PLC – Set Address*.

Задание на лабораторную работу

Реализовать обмен данными между двумя контроллерами в соответствии с заданием преподавателя.

Примеры заданий:

1. Записать данные из ячеек на первом контроллере 40:8 и 180:10 в ячейки второго контроллера 140:8 и 80:10 соответственно.
2. Записать данные из ячейки на первом контроллере 25:8 в ячейку второго контроллера 200:8, а после снова вернуть данные в первый контроллер из ячейки 200:8 в ячейку 100:8.

3. Записать данные из ячеек на первом контроллере 60:5 и 75:3 в ячейки второго контроллера 160:5 и 25:3 соответственно.

4. Записать данные из ячейки на первом контроллере 111:11 в ячейку второго контроллера 222:11, а после снова вернуть данные в первый контроллер из ячейки 222:11 в ячейку 33:11.

5. Записать данные из ячеек на первом контроллере 65:10 и 105:8 в ячейки второго контроллера 165:10 и 125:8 соответственно.

6. Записать данные из ячейки на первом контроллере 95:4 в ячейку второго контроллера 170:4, а после снова вернуть данные в первый контроллер из ячейки 170:4 в ячейку 155:4.

Отчет по лабораторной работе должен содержать:

- 1) Скриншот и описание работы программы;
- 2) Фотографию реализованной установки;
- 3) Выводы, обобщающие полученные результаты.

ЛАБОРАТОРНАЯ РАБОТА № 6. SIEMENS/ALLEN BRADLEY. РАЗРАБОТКА АЛГОРИТМА УПРАВЛЕНИЯ

Цель работы: освоение системы программирования RSLogic 500 Starter и разработка с ее помощью программ управления промышленными контроллерами Allen Bradley для решения задач автоматизации управления технологическими процессами в реальном времени.

Теоретические сведения

Язык LD (Ladder Diagram) или РКС (Релейно-Контактные Схемы) представляет собой графическую форму записи логических выражений в виде контактов и обмоток реле. Предназначен для программирования промышленных контроллеров (ПЛК). Синтаксис языка удобен для замены логических схем, выполненных на релейной технике. Ориентирован на инженеров по автоматизации, работающих на промышленных предприятиях. Обеспечивает наглядный интерфейс логики работы контроллера, облегчающий не только задачи собственно программирования и ввода в эксплуатацию, но и быстрый поиск неполадок в подключаемом к контроллеру оборудовании.

Слева и справа схема ограничена вертикальными линиями - шинами питания. Между ними расположены цепи, образованные контактами и обмотками реле, по аналогии с обычными электронными цепями. Слева любая цепь начинается набором контактов, которые посылают слева направо состояние «ON» или «OFF», соответствующие логическим значениям ИСТИНА или ЛОЖЬ. Каждому контакту соответствует логическая переменная. Если переменная имеет значение ИСТИНА, то состояние передается через контакт. Контакты могут быть соединены параллельно, тогда соединение передает состояние логическое ИЛИ. Если контакты соединены последовательно, то соединение передаёт логическое И.

Контроллеры Allen Bradley настраиваются и программируются через RSLogic 500 Starter. В лабораторной работе используется контроллер MicroLogix 1500 LSP Series A.

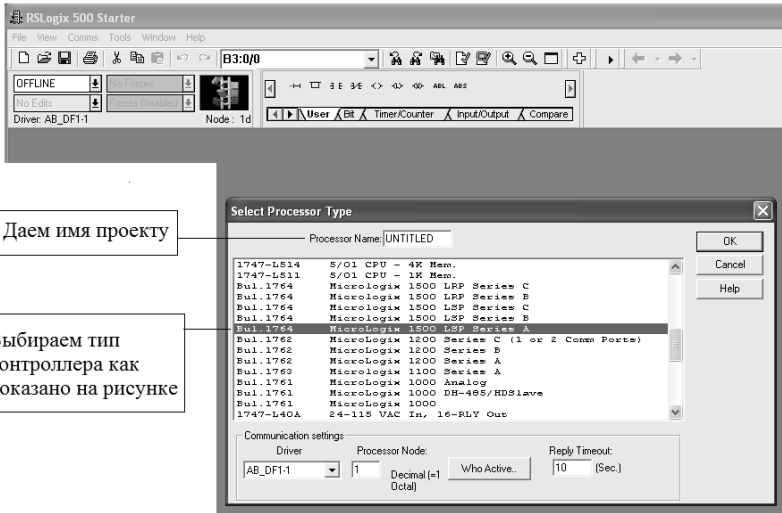


Рис 6.1 Создание проекта в RSLogix 500 Starter

Библиотека элементов содержит все необходимые элементы для написания программы на языке LAD.



Рис 6.2 Библиотека элементов RSLogix 500 Starter

На языке LD на базе контроллера Allen Bradley имя входного контакта задается следующим образом – I:0/0, а выходного – O:0/0 (где вторая цифра — это номер входного/выходного контакта). Таймеры обозначаются – T4:0, а счетчики – C5:0 (где вторая цифра — это порядковый номер таймера/счетчика).

Загрузка в контроллер осуществляется с помощью вкладки Command – Download.



Рис 6.3 Компиляция и загрузка программы на контроллер

Перед загрузкой программы в контроллер запускается компилятор (проверка на ошибки), и если есть ошибки, можно их определить с помощью клавиш F3(предыдущая ошибка) или F4(следующая ошибка).



Рис 6.4 Запуск программы на контроллер

Задание на лабораторную работу

Реализовать программу для контроллера в соответствии с заданием преподавателя.

Примеры задания:

1. Разработать программу управления лентой транспортера, которая может приводиться в движение с помощью электродвигателя. В начале транспортера имеются две кнопки: S1 для запуска и S2 для остановки. В конце транспортера тоже имеются две кнопки: S3 для запуска и S4 для остановки. Транспортер можно запускать или останавливать с любого конца. Также датчик S5 останавливает транспортер, когда предмет, находящийся на ленте, достигает конца ленты.

2. Разработать программу для управления установкой непрерывного литья. Установка представляет собой ленту транспортера с установленными на ней формами. При доставке формы в зону разлива в течение 6 секунд осуществляется заливка в форму расплавленного материала.

Требования к системе управления установкой:

- В начале транспортера имеются две кнопки: S1 для запуска и S2 для остановки. В конце транспортера тоже имеются две кнопки: S3 для запуска и S4 для остановки. Транспортер можно запускать или останавливать с любого конца.

- поступление формы в зону разлива фиксируется фотоэлектрическим датчиком.

- на время разлива транспортер останавливается;

- при завершении процесса разлива движение транспортера продолжается до поступления следующей формы;

- при отсутствии поступления форм в зону разлива более 15 секунд транспортер останавливается и выдается сигнал тревоги.

Отчет по лабораторной работе должен содержать:

- 1) Скриншот программы и описание работы программы
- 2) Выводы, обобщающие полученные результаты.

РЕКОМЕНДУЕМЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Бошляков, А.А.* Проектирование алгоритмического и программного обеспечения мехатронных систем: Учеб. пособие / А.А. Бошляков, С.В. Овсянников. Москва: МГТУ им. Н.Э. Баумана, 2007, 56 с.

2. *Кадыров Э.Д.* Программируемые логические контроллеры. Программирование и конфигурирование: Учеб. пособие / Э.Д. Кадыров, А.Н. Кравченко, А.Ю. Фирсов. СПб.: СПГИ, 2007.

3. *Назаров, С. В.* Локальные вычислительные сети: Справочник. В 3-х кн. Кн. 2: Аппаратные и программные средства / С.В. Назаров, В.П. Поляков, А.В. Луговец и др. М.: Финансы и статистика, 1994, 264 с.

4. *Паринов А.В.* Сети связи и системы коммутации: Учебное пособие / А.В. Паринов, С.В. Ролдугин, В.А. Мельник. Воронеж: Научная книга, 2016, 178 с.

5. *Стетина В.В.* Архитектура ЭВМ и вычислительные системы: Учебник. М.: КУРС: ИНФРА-М, 2018, 384 с.

6. *Шишов О.В.* Программируемые контроллеры в системах промышленной автоматизации: Учебник. М.: НИЦ ИНФРА-М, 2016, 365 с.

СОДЕРЖАНИЕ

Лабораторная работа № 1. Пневматика FESTO. Разработка алгоритма управления.....	3
Лабораторная работа № 2. Контроллер АЛЬБАТРОС. Конфигурирование и программирование контроллера.....	11
Лабораторная работа № 3. Обмен данными между контроллером AUTOLOG и SCADA системой iFIX.....	20
Лабораторная работа № 4. Программирование сенсорной панели MAGELIS.....	29
Лабораторная работа № 5. Обмен данными между контроллерами MODICON.....	34
Лабораторная работа № 6. SIEMENS/ALLEN BRADLEY. Разработка алгоритма управления.....	41
Рекомендуемый библиографический список.....	45

ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ, СИСТЕМЫ И СЕТИ

*Методические указания к лабораторным работам
для студентов бакалавриата направления 15.03.04*

Сост.: *А.В. Бойков, Н.И. Котелева*

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 27.06.2019. Формат 60×84/16.
Усл. печ. л. 2,7. Усл.кр.-отт. 2,7. Уч.-изд.л. 2,3. Тираж 100 экз. Заказ 624. С 224.

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