

Министерство образования и науки Российской Федерации  
Федеральное агентство по образованию  
Южно-Уральский государственный университет  
Кафедра «Электропривод и автоматизация  
промышленных установок»

681.5(07)  
Б825

А.М. Борисов, А.С. Нестеров, Н.А. Логинова

# **ПРОГРАММИРУЕМЫЕ УСТРОЙСТВА АВТОМАТИЗАЦИИ**

Учебное пособие

Челябинск  
Издательский центр ЮУрГУ  
2010

УДК 658.564(075.8) + 681.51(075.8)  
Б825

*Одобрено  
учебно-методической комиссией энергетического факультета.*

*Рецензенты:*  
В.С. Жабреев, В.М. Попов

Б825 Борисов, А.М.  
Программируемые устройства автоматизации: учебное пособие /  
А.М. Борисов, А.С. Нестеров, Н.А. Логинова. – Челябинск: Издательский  
центр ЮУрГУ, 2010. – 186 с.

Учебное пособие предназначено для студентов специальности 140604 «Электропривод и автоматика промышленных установок и технологических комплексов» при изучении дисциплины «Автоматизация типовых технологических процессов».

В пособии представлены конструкции, технические характеристики, используемые языки и принципы программирования программируемых логических контроллеров и сенсорных мониторов ведущих фирм Германии, Японии и США, нашедших широкое применение на российских предприятиях. Рассмотрено построение, конфигурирование и пользовательское программирование промышленной сети автоматизации PROFIBUS-DP.

Учебное пособие может быть использовано студентами других специальностей, а также инженерно-техническими работниками предприятий, занимающихся разработкой систем автоматизации.

УДК 658.564(075.8) + 681.51(075.8)

© Издательский центр ЮУрГУ, 2010

## ВВЕДЕНИЕ

Предлагаемое учебное пособие предназначено для освоения современных программируемых средств, используемых для автоматизации промышленного производства. В пособии представлены наиболее широко используемые на российских предприятиях изделия иностранных фирм, таких как Siemens (Германия), Omron (Япония), Automation Direct (США). Пособие основано (полностью) на технической документации фирм-изготовителей этих изделий.

Излагаемый в пособии материал представляется в виде лабораторных работ, используемых при изучении курса «Автоматизация типовых технологических процессов» в Южно-Уральском государственном университете. Однако, по объему представленный материал выходит за пределы материала необходимого для выполнения конкретной лабораторной работы. Например, в лабораторной работе используется только один язык программирования, а излагаются и другие, используемые для данного изделия, языки программирования и т.п.

В пособии по каждому изучаемому изделию дается конструкция, техническая характеристика изделия, принцип его программирования, примеры программирования и варианты задания для индивидуальности обучения студентов.

В пособии представлены зачастую схемы структурные, принципиальные и соединений изучаемых изделий и установок, которые могут оказать существенную помощь студентам при выполнении курсового и дипломного проектирования.

# 1. РАБОТА №1. ОСНОВЫ ПРОГРАММИРОВАНИЯ ПРОГРАММИРУЕМОГО КОНТРОЛЛЕРА SIMATIC S7-300

## 1.1. Цель работы

Ознакомиться с техническими характеристиками контроллера фирмы Siemens SIMATIC S7-300. Приобрести навыки программирования на лабораторном стенде с применением программного обеспечения STEP7 v.5.3.

## 1.2. Содержание работы

а) Изучить назначение, технические характеристики контроллера фирмы Siemens SIMATIC S7-300.

б) Изучить основы системы команд и принципы программирования контроллера.

в) Дома при подготовке к работе:

- выполнить синтез системы автоматизации согласно заданного варианта;
- составить программу на языке лестничных диаграмм.

г) В лаборатории:

- освоить графическую среду программного обеспечения STEP 7 v. 5.3;
- сконфигурировать центральную стойку контроллера;
- набрать на компьютере подготовленную дома программу и откомпилировать ее;
- ввести программу в контроллер и убедиться в правильности её работы;
- запрограммировать управление модулем с аналоговыми входами и выходами. Снять характеристики (вход/выход) одного канала ввода аналогового сигнала и одного канала вывода аналогового сигнала.

## 1.3. Описание лабораторного стенда

На рис. 1.1 представлена структура лабораторной установки для изучения программирования контроллера фирмы Siemens SIMATIC S7-300.

Лабораторная установка включает в себя:

- 1 – ПЭВМ типа IBM;
  - 2 – адаптер связи ПЭВМ с процессором (CPU) программируемого контроллера для записи и отладки программ;
  - 3 – контроллерный блок;
  - 4 – блок имитации сигналов и команд;
  - 5 – кабель связи между контроллерным блоком и блоком имитации.
- Контроллерный блок включает в себя:
- блок питания типа SITOP;
  - процессорный модуль CPU314 (заказной номер 314-1AE04-0AB0);

- модуль ввода дискретных сигналов SM321 DI32xDC24V (заказной номер 321-1BL00-0AA0) на 32 канала ввода сигналов постоянного тока напряжением 24 В;
- модуль вывода дискретных сигналов SM322 DI32xDO32В/0,5А (заказной номер 322-1BL00-0AA0) на 32 канала вывода сигналов постоянного тока напряжением 24 В с допустимым током нагрузки 0,5 А;
- модуль ввода/вывода аналоговых сигналов SM334 AI4/AO2x8bit (заказной номер 334-0CE01-0AA0) на 4 канала ввода и 2 канала вывода аналоговых сигналов;
- тумблер «Сеть» для питания контроллерного блока напряжением 220 В, 50 Гц;
- разъем для связи контроллерного блока с блоком имитации.

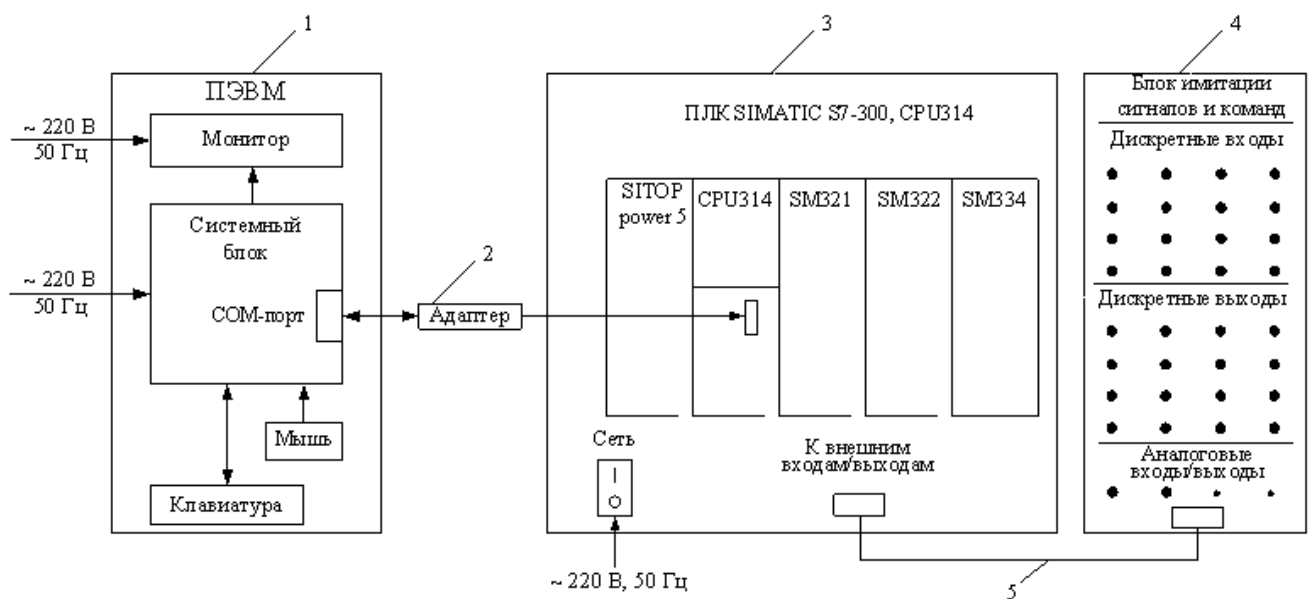


Рис. 1.1. Структурная смена лабораторной установки

Выше для каждого модуля контроллерного блока указан заказной номер, который потребуется для программирования контроллера при его конфигурировании.

**Процессорный модуль CPU314** имеет память ОЗУ 48 Кбайт и загрузочную вставную ММС максимум на 8 Мбайт. Языки программирования LAD/FBD/STL в английской аббревиатуре (KOP/FUP/AWL – в немецкой). Работает с модулями, которые могут размещаться в 4-х стойках. Максимальное количество модулей на каждой стойке 8. Максимальное количество дискретных сигналов ввода/вывода 1024, аналоговых сигналов максимум 256, таймеров 256 с диапазоном времени от 10 мс до 9990 с, счетчиков 256 с диапазоном счета от 0 до 999, промежуточные переменные (меркеры) – 256 байт.

В табл. 1.1 представлено распределение памяти контроллера.

Таблица 1.1

Область памяти	Символ	Биты, слова	Пояснения
Дискретные входы	I	I0.0...I127.7	Состояние входных устройств, подключенных к входным цепям
Аналоговые входы	PIW	PIW256...PIW766	Представленная в цифровой форме аналоговая величина
Дискретные выходы	Q	Q0.0...Q127.7	Состояние дискретных выходов
Аналоговые выходы	PQW	PQW256...PQW766	Представленная в аналоговой форме цифровая величина
Таймеры	T	T0...T255	Режим работы: формирователь импульса, формирователь удлиненного импульса, задержка включения, задержка включения с запоминанием, задержка выключения
Счетчики	C	C0...C255	Счетчики
Меркеры	M	M0.0...M255.7	Промежуточные (внутренние) переменные

Адресация сигналов и команд программируемых контроллеров SIMATIC S7-300 представлена детально в Приложении А к данной работе.

**Модуль ввода дискретных (цифровых) сигналов SM321** имеет четыре входных байта с адресами I0...I3 (рис. 1.2). В работе используются только входные байты I0 и I1.

Модуль имеет индикацию состояния зелеными светодиодами на каждом канале на лицевой панели модуля. Входной ток канала ввода при сигнале 1 и напряжении 24 В не превышает 7 мА.

**Модуль вывода дискретных (цифровых) сигналов** имеет четыре выходных байта с адресами Q4...Q7. В работе использованы только выходные байты Q4 и Q5. Как видно из схемы (рис. 1.3) состояние каждого канала выхода индицируется светодиодом, расположенном на лицевой панели модуля. В каждом канале предусмотрен стабилитрон для защиты цепей канала от перенапряжений при коммутации активно-индуктивной нагрузки. Коммутируемые напряжения и токи приведены в обозначении модуля. Диапазон сопротивления нагрузки от 48 Ом до 4 кОм.

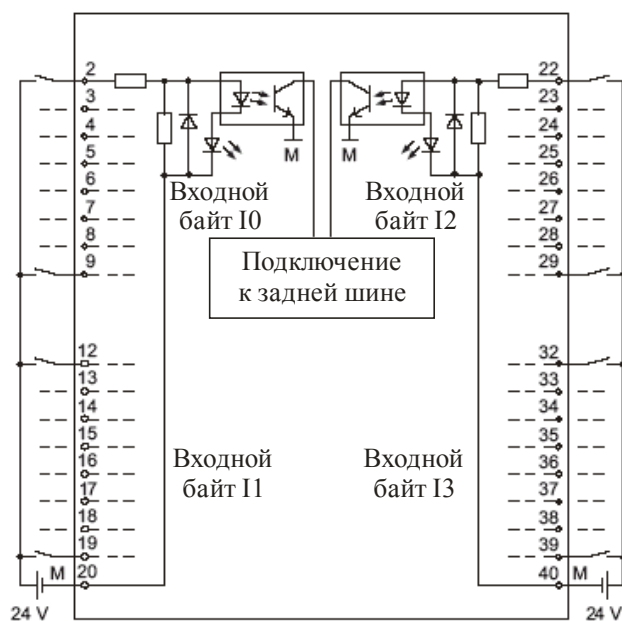


Рис. 1.2. Принципиальная схема цифрового модуля ввода SM321 DI32xDC24V

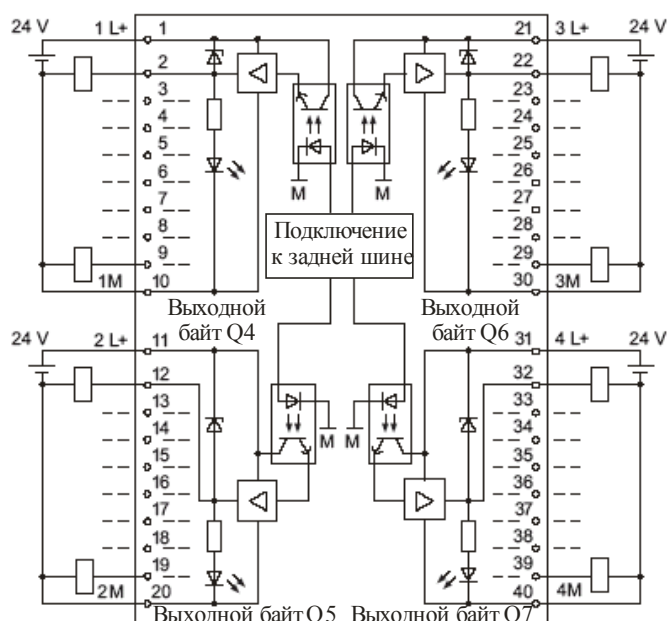


Рис. 1.3. Принципиальная схема цифрового модуля вывода SM322 DO32xDC24V/0,5A

**Аналоговый модуль ввода/вывода SM334** (рис. 1.4) не требует параметризации. Установка вида подачи входного сигнала (сигнал напряжения или токовый) и вида вывода обеспечивается путем подключения в соответствии со схемой рис. 1.4.

В лабораторной установке на аналоговые каналы подаются сигналы напряжения с потенциометрических резисторов, установленных на блоке имитации сигналов и команд. Поступающее с выходов напряжение индицируется изменением яркости свечения светодиодов, подключенных к соответствующим каналам

вывода. Указанные светодиоды также установлены на блоке имитации сигналов и команд.

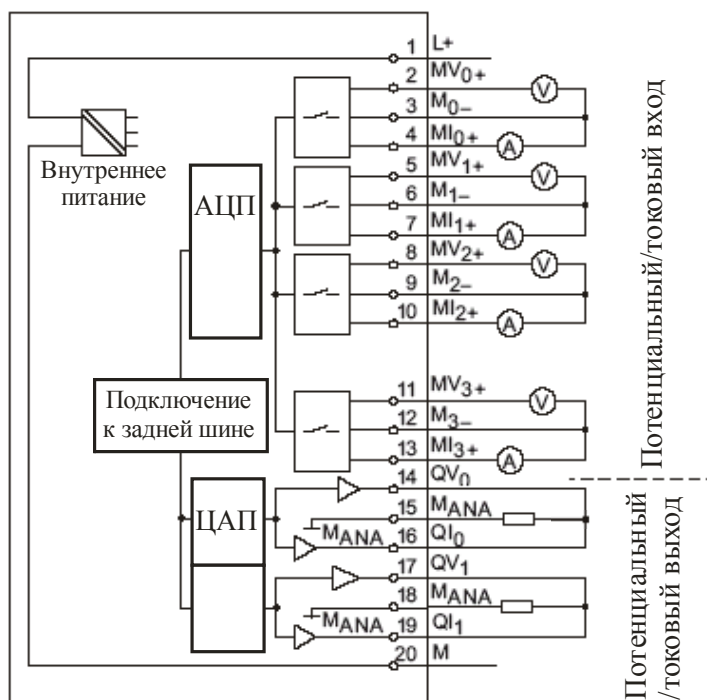


Рис. 1.4. Принципиальная схема аналогового модуля ввода/вывода SM334 AI4/AO2x8bit

**Блок имитации сигналов и команд** используется для выдачи дискретных и аналоговых сигналов на соответствующие входы контроллера и индикации выходных сигналов с контроллера. Дискретные входные сигналы подаются с 8 кнопок с адресами I0.0...I0.7 и 8 тумблеров с адресами I1.0...I1.7. Выходные дискретные сигналы подаются на 16 светодиодов с адресами Q4.0...Q4.7 и Q5.0...Q5.7. Как уже указывалось выше на блоке индикации установлено два потенциометрических резистора и два светодиода для работы с аналоговыми сигналами.

#### 1.4. Конфигурирование в STEP 7 v 5.3

Для подготовки программ при проведении лабораторных работ служит программа – «SIMATIC Manager». Для запуска «SIMATIC Manager» необходимо на рабочем столе Windows дважды щелкнуть курсором мыши по ярлычку



Откроется окно «SIMATIC Manager».

«SIMATIC Manager [Администратор SIMATIC]» – это центральное окно, которое становится активным при запуске STEP 7. По умолчанию запускается мастер STEP 7 (STEP 7 Wizard) (рис. 1.5), который оказывает помощь при создании проекта STEP 7. Структура проекта используется для надлежащего хранения и



размещения всех данных и программ. В предварительном обзоре (Preview) можно включать и выключать отображение структуры создаваемого проекта. Чтобы перейти к следующему диалоговому окну, щелкните на кнопке «Next [Дальше]».

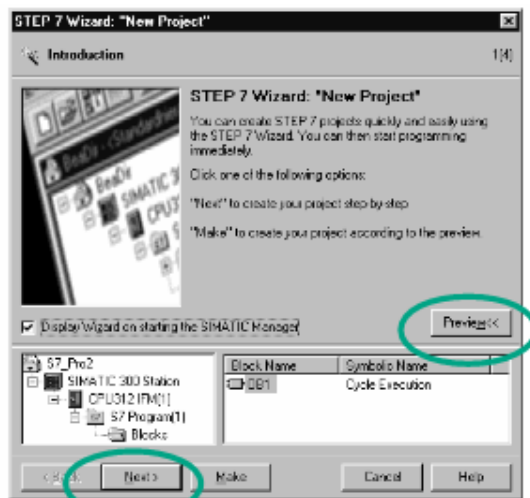


Рис. 1.5. Окно мастера STEP 7 (STEP 7 Wizard)

Выберите CPU (рис. 1.6) в соответствии с тем, который является мастером системы. Установка по умолчанию для адресов MPI равна 2. Щелкните на Next, чтобы подтвердить настройки и перейти к следующему диалоговому окну.

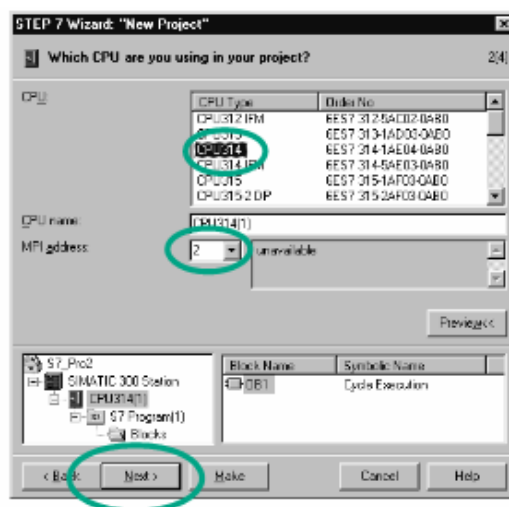


Рис. 1.6. Окно выбора процессора в STEP 7 Wizard

Выберите организационный блок OB1 (рис. 1.7). Выберите один из языков программирования: контактный план (LAD), список операторов (STL) или функциональный план (FBD). Подтвердите настройки кнопкой Next.

Дважды щелкните в поле «Project name [Имя проекта]» (рис. 1.8), чтобы выбрать предлагаемое имя, заменив его, например на «Master S7-300». Щелкните на кнопке «Make [Создать]», чтобы сгенерировать свой новый проект в соответствии с предварительным обзором.

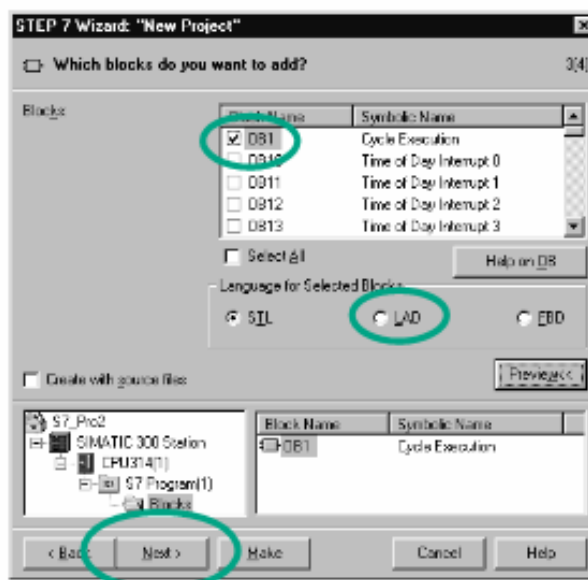


Рис. 1.7. Окно выбора языка программирования

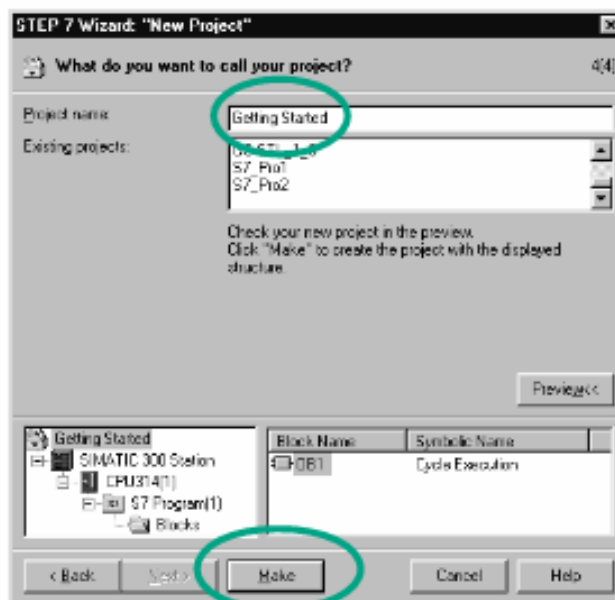
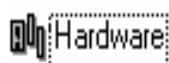


Рис. 1.8. Окно выбора имени проекта

Только что созданный проект отображается с выбранной S7 станцией и CPU (рис. 1.9). Щелкая на значке «+» или «-» открыть или закрыть папку.

Конфигурирование аппаратной части «Master`a S7-300» производится при нажатии на надпись «Master S7-300» в левой части окна. В правой части окна появится значок:



– [Оборудование]. Щелкните по нему, появится окно конфигурирования оборудования – «HW Config».

Добавление модулей производится из каталога «Catalog», содержащий список компонентов оборудования SIMATIC.

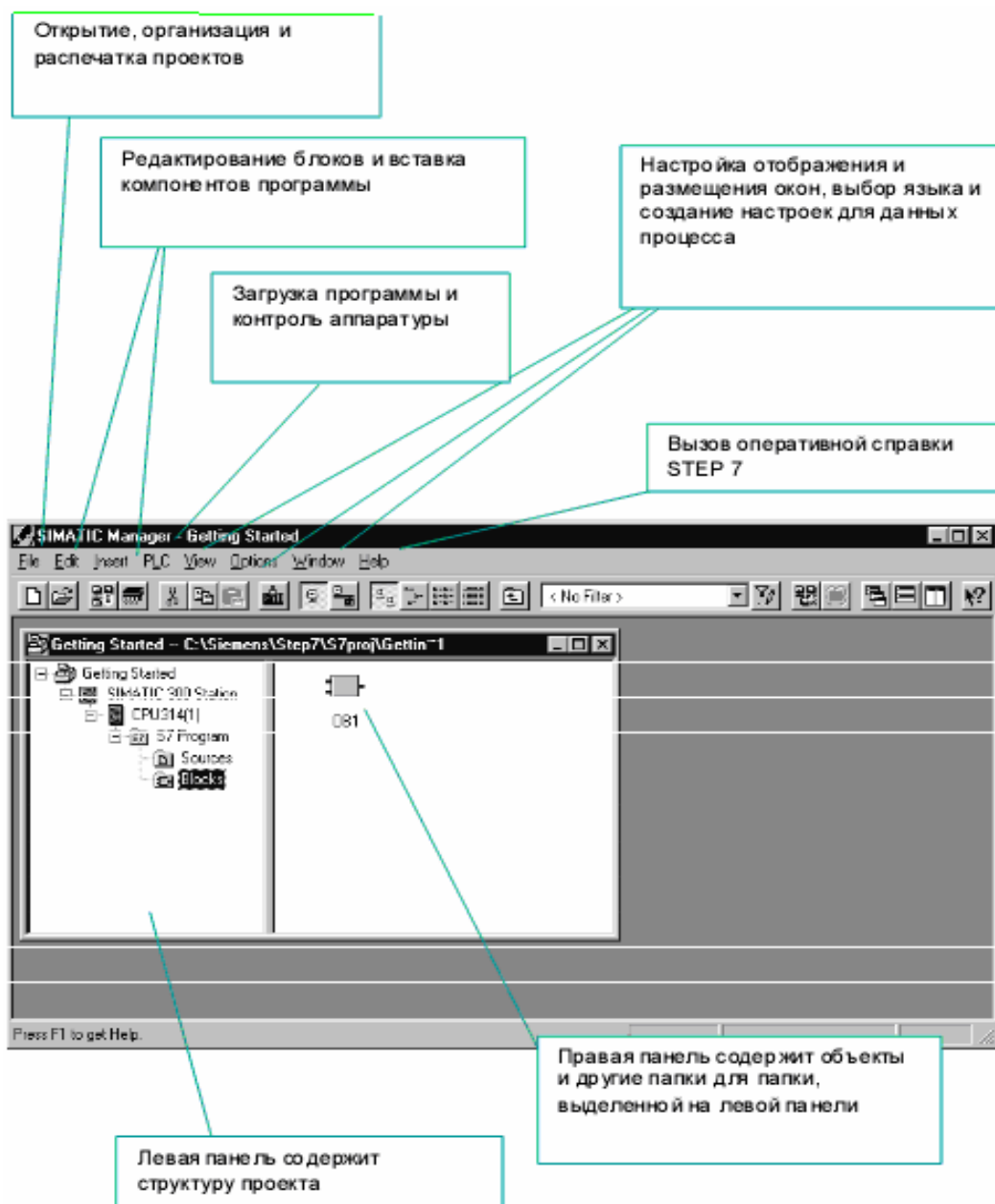


Рис. 1.9. Окно SIMATIC Manager

В каталоге в папке SIMATIC 300/RACK-300 выберите центральную стойку (профильную шину) «Rail». Отбуксируйте стойку, используя Drag&Drop, в окно станции. В нижней части окна появляется подробное представление стойки с дополнительными данными.

В качестве альтернативы можно дважды щелкнуть левой кнопкой мыши на выбранном модуле «Rail» и он появится в конфигурационной таблице.

Основное поле (рис. 1.10) содержит сведения о CPU и модулях системы:

- 1) Slot – позиция модуля в системе (3-я позиция используется только под интерфейсный модуль);
- 2) Module – тип модуля или CPU;
- 3) I address – возможные адреса модуля ввода;

- 4) Q address – возможные адреса модуля вывода;
- 5) Comment – строка для занесения комментариев.

Слот	Модуль	Номер для заказа	Адрес MPI	Комментарий
Slot	Module	Order number	MPI address	Comment
1	PS 307 10A	6ES7 307-1KA00-0AA0		
2	CPU 314	6ES7 314-1AE01-0AB0	2	
3				
4	DI8xAC120/230V	6ES7 321-1FF10-0AA0		
5	AI8x12Bit	6ES7 331-7KF02-0AB0		272...287
6	AI8x16Bit	6ES7 331-7NF10-0AB0		288...303
7	AI8xTC/4xRTD, Ex	6ES7 331-7SF00-0AB0		304...319
8	AO2x12Bit	6ES7 332-5HB00-0AB0		320...323
9	AO2x12Bit	6ES7 332-5HB81-0AB0		336...339
10				
11				

Рис. 1.10. Окно HW Config

На стойку устанавливается блок питания PS (при необходимости) и модули (например, CPU) из окна «Hardware Catalog». Слоты, в которые можно установить тот или иной модуль выделяются цветом. Выберите модули по заказным номерам (Order number), которые имеются на лицевой панели каждого модуля. Отбуксируйте модули, используя Drag&Drop, в соответствующие строки стойки (конфигурационная таблица).

STEP 7 проверяет, не нарушены ли правила для слотов (CPU S7-300, например, может быть установлен только в слоте 2).

⊘ – Символ нарушения правил для слотов.

Если выделить слот в стойке модулей, можно увидеть список всех возможных для установки модулей, открыв контекстно-зависимое меню (правой кнопкой мыши) «Insert Object [Вставить объект]» или «Replace Object [Заменить объект]». Эта возможность избавляет Вас от необходимости поиска в каталоге аппаратуры. Вы выбираете из всех модулей, которые имеются в текущем открытом профиле каталога.

!!! Помните, что модули в слотах следует располагать в той же последовательности, в какой они расположены физически и с теми заказными номерами, которые указаны на них.

Для проверки правильности проведенного конфигурирования необходимо в меню «Station [Станция]» выбрать и активировать позицию «Consistency Check [Проверить согласованность]». Появится соответствующее сообщение об отсутствии или наличии ошибок, и каких.

Покидание окна конфигурирования производится после сохранения данной станции «Station – Save» или сохранения с компиляцией «Station – Save and Compile».

## 1.5. Программирование пользовательской задачи

Программирование пользовательской задачи (создание пользовательской программы) реализуется при помощи программного обеспечения STEP7 v5.3. Обучающийся, в соответствии с поставленной задачей, разрабатывает алгоритм автоматизации управления объектом либо в виде логических уравнений, либо в виде схемы алгоритма. Входным, выходным сигналам и внутренним переменным (меркерам) присваиваются проектировщиком адреса в соответствии с конфигурационной таблицей.

Основная программа формируется в организационном блоке OB1. Для доступа к OB1 необходимо в левой панели окна SIMATIC Manager'a (рис. 1.9) дважды щелкнуть левой кнопкой мыши по знаку CPU..., затем по появившемуся значку S7 Program и наконец, по появившемуся значку Blocks. Появляется в правой части пиктограмма OB1. Дважды щелкните на пиктограмме OB1. Открывается окно программирования (рис. 1.11).

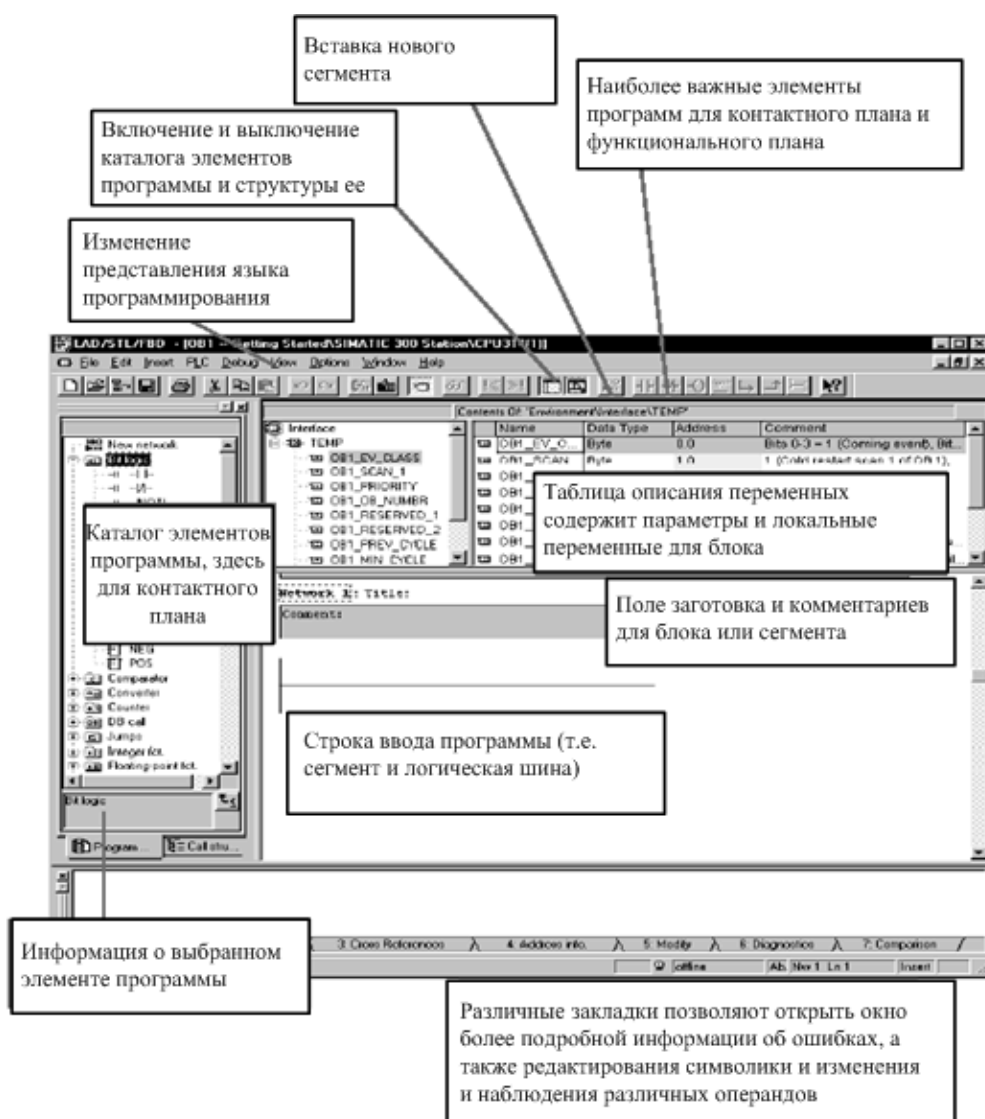


Рис. 1.11. Окно программирования

В меню «View [Вид]» можно выбрать язык программирования LAD/STL/FBD. Язык LAD (язык лестничных диаграмм) удобен для инженеров-электриков, язык STL (язык операторов) ориентирован для пользователей из мира компьютерных технологий (мнемокода) и, наконец, язык FBD (язык функциональных блоков) ориентирован на инженеров-схемотехников. STEP7 v5.3 позволяет в любое время изменить язык программирования.

Ниже рассматривается программирование на языке LAD.

При программировании лестничными диаграммами (LAD) в виде релейно-контактной схемы программа разделяется на сегменты. Каждый сегмент представляет собой отдельную цепь, по которой может протекать ток. Шина питания находится слева (вертикальная линия).

Для создания нового сегмента необходимо нажать на кнопку



– New network.

Рассмотрим ввод программы для реализации простого уравнения:

$$Q4.0 = (I1.1 \cdot \overline{I1.3}) + I1.2$$

1) Поставьте в токовую цепь (сегмент) реле (катушку) (рис. 1.12). Для этого нажмите на кнопку



– Coil [Катушка]



Рис. 1.12.

2) Щёлкните по тому месту (непосредственно по линии) куда, хотите вставить замыкающий (нормально открытый) контакт (рис. 1.13).

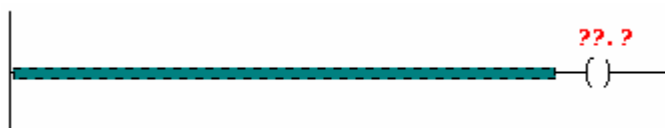


Рис. 1.13.

3) Вставьте замыкающий контакт, нажав на кнопку



– Normally Open Contact [Нормально открытый контакт] (рис. 1.14).



Рис. 1.14.

4) Создайте параллельную ветвь для замыкающего контакта. Для этого щелкните по месту, от которого будет ответвление (в нашем случае – шина питания), нажмите на кнопку

 – Open Branch [Создать ветвление] (рис. 1.15).

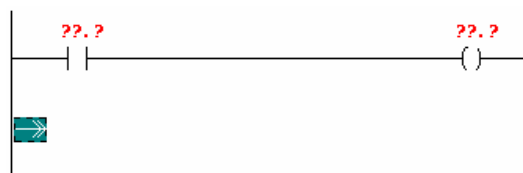


Рис. 1.15.

5) Вставьте нормально открытый контакт и завершите параллельную ветвь, нажав на кнопку

 – Close Branch [Завершить ветвление] (рис. 1.16).




Рис. 1.16.

6) Выделите место, куда необходимо поставить нормально замкнутый (размыкающий) контакт (рис. 1.17).



Рис. 1.17.

7) Вставьте размыкающий контакт, нажав на кнопку

 – Normally Closed Contact [Нормально закрытый контакт] (рис. 1.18).

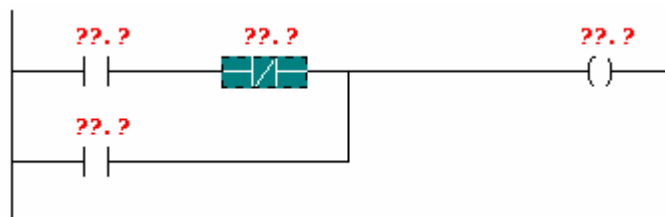


Рис. 1.18.

8) Теперь напишите адреса контактов. Для этого щёлкните по знакам ??? над контактом и введите адрес (ввод адреса завершается нажатием клавиши **ENTER**) (рис. 1.19).

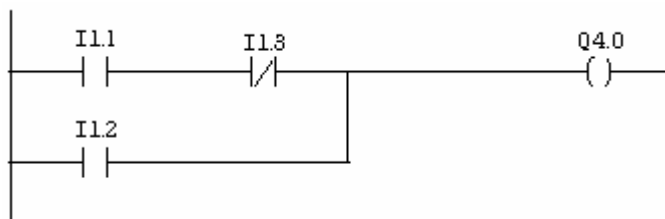


Рис. 1.19.

В среде STEP7 удобно пользоваться символической адресацией, когда любому адресу может быть присвоено удобное для пользователя символическое имя. Присвоим I1.1 – «Key1 [Кнопка 1]», I1.3 – «Key2 [Кнопка 2]», I1.2 – «Switch [Тумблер]», Q4.0 – «Lamp [Лампа]». Для этого необходимо открыть редактор символов, нажав одновременно Ctrl-Alt-T или выбрав Options→Symbol Table (выбрав в главном меню редактора программы Options и затем выбрав из выпадающего меню Symbol Table). Появится окно редактора символов (рис. 1.20).

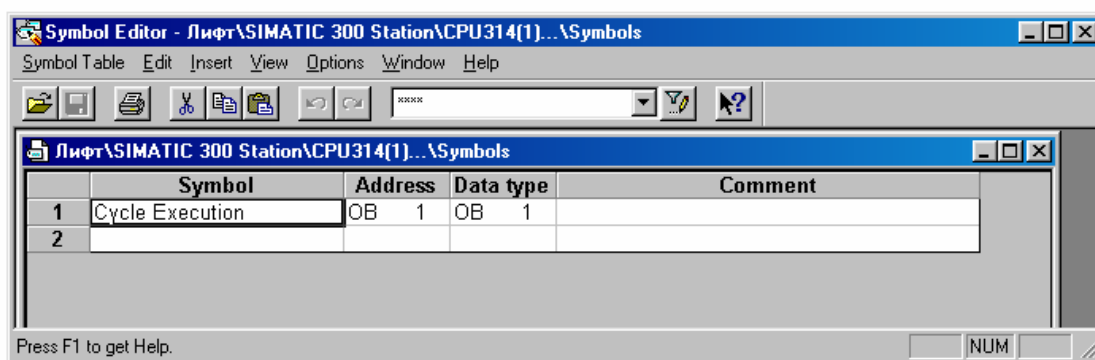


Рис. 1.20. Окно редактора символов

Заполните свободную строку в таблице символов:

- 1) в столбце символов (Symbol) вставьте: «Key1»;
  - 2) в строке адресов (Address) вставьте: «I1.0» (автоматически в столбце типа данных появится BOOL – тип данных «бит»);
  - 3) в столбце для комментариев (Comment) вставьте «Кнопка1».
- Введите символы для остальных адресов (см. рис. 1.21).



	Symbol	Address	Data type	Comment
1	Cycle Execution	OB 1	OB 1	
2	Key1	I 1.0	BOOL	Кнопка1
3	Key2	I 1.3	BOOL	Кнопка2
4	Switch	I 1.2	BOOL	Тумблер
5	Lamp	Q 4.0	BOOL	Лампа
6				

Рис. 1.21. Окно редактора символов

Сохраните таблицу символов (Symbol Table→Save). Закройте окно редактора символов. Теперь программа будет выглядеть следующим образом (рис. 1.22).

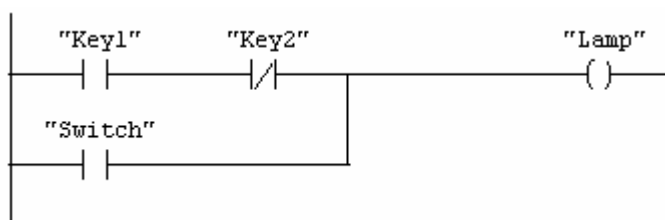





Рис. 1.22.

Для удаления какого-либо элемента программы необходимо его выделить и выбрать: Edit→Delete. Для отмены предыдущего действия необходимо выбрать: Edit→Undo.

В случае некорректного ввода адреса или символического имени программа высвечивает некорректное обозначение красным цветом.

Рассмотрим вставку функции на примере создания следующей программы: если кратковременно нажать на кнопку Key1, то зажётся лампа Lamp (адрес Q4.0), если кратковременно нажать на кнопку Key2, лампа Lamp (адрес Q4.2) погаснет. Для этого воспользуемся функцией SR триггера:

- 1) создайте новый сегмент нажав на кнопку ;
- 2) внесите в таблицу символов символ Lamp с адресом Q4.2;
- 3) нажмите в редакторе программы на  кнопку Program Elements – появится окно списка функций;
- 4) в окне списка функций выбрать: Bit logic→ SR (SR триггер) – щёлкнуть по этому значку быстро два раза;
- 5) над функцией SR триггера (рис. 1.23) вместо символов "???" вставьте символическое имя выхода;

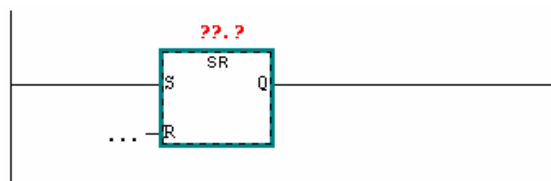


Рис. 1.23.

б) на входную линию функции S поставьте нормально открытый контакт Key1, а на место входа R – Key2 (рис. 1.24).

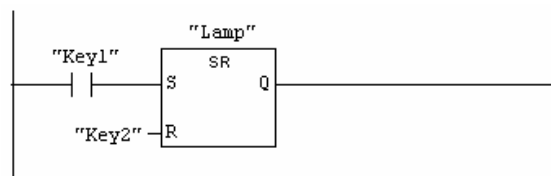


Рис. 1.24.

Для удаления функции необходимо её выделить, подвести к ней курсор, нажать на правую клавишу мыши и из выпадающего меню выбрать Delete.

## 1.6. Операции с таймерами

Таймеры имеют собственную область памяти в CPU. Для каждого из 256 таймеров резервируется 16-битное слово. Биты с 0 по 11 в таймерном слове выделены для записи установки времени, в битах 12 и 13 записывается база времени (код дискретности генератора импульсов), биты 14 и 15 не используются.

Уставка времени таймера может загружаться в двоичном, шестнадцатеричном или двоично-десятичном (BCD) коде.

Уставку времени можно задать в BCD коде с использованием следующего синтаксиса:

S5T#aH\_bbM\_ccS\_dddMS,

где a=часы, bb=минуты, cc=секунды и ddd= миллисекунды.

Диапазон уставки времени от 0 до 9990 с, что соответствует 2H\_46M\_30S.

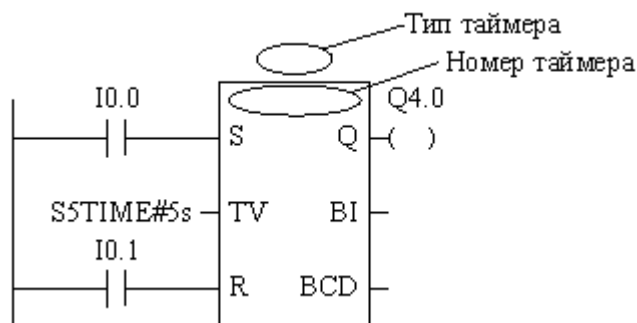
База времени устанавливается автоматически и значение округляется до ближайшего меньшего с этой базой времени. База времени определяет дискрету времени используемого генератора. Минимальная база времени равна 10 мс, максимальная 10 с. (см. табл. 1.3).

Таблица 1.3

База времени	Двоичный код для базы времени
10 мс	00
100 мс	01
1 с	10
10 с	11

В режиме RUN текущее значение таймера уменьшается на одну единицу через интервал времени, установленный базой времени, до тех пор, пока значение времени не станет равным нулю, что соответствует срабатыванию таймера.

В STEP 7 реализуется 5 разновидностей таймеров. В лестничных диаграммах блоки всех разновидностей обозначаются одинаково и различаются только обозначением типа таймера (рис. 1.25).



S – вход запуска, TV – уставка (заданное время),  
R – вход сброса, Q – выход (состояние таймера),  
BI – оставшееся время в двоичном формате,  
BCD – оставшееся время в двоично-десятичном формате

Рис. 1.25. Блок таймера и параметры входа/выхода

Блок таймера имеет вход R. Появление сигнала «1» на входе R во время работы таймера, заданного уставкой, сбрасывает таймер и выход отключается.

Имеется второй упрощенный вариант программирования таймеров, в котором отсутствуют вход R и выходы BI и BCD.

Ниже на рис. 1.26 – 30 приведены для каждого типа таймера оба варианта программирования и временные диаграммы их работы, из которых без дополнительных комментариев ясны особенности их работы.

Во втором варианте программирования таймеров возможен сброс таймеров подачей команды R (RESET) по адресу таймера (рис. 1.31).

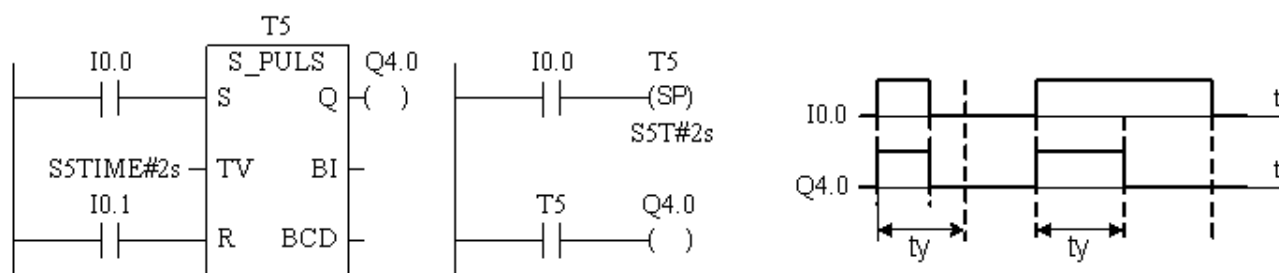


Рис. 1.26. Таймер формирователь импульса

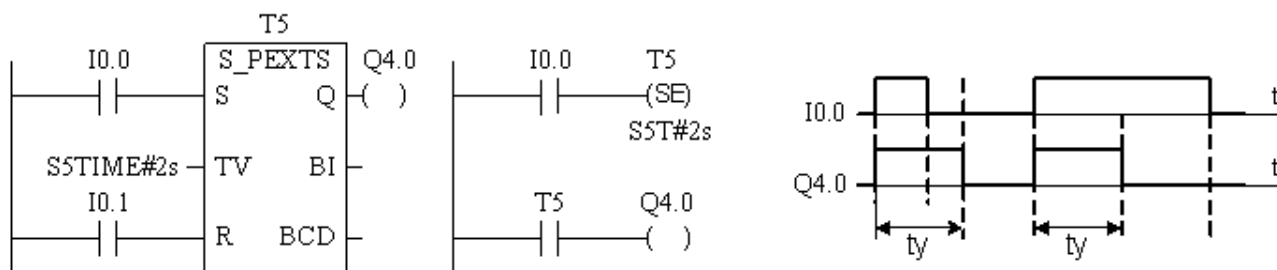


Рис. 1.27. Таймер формирователь удлиненного импульса

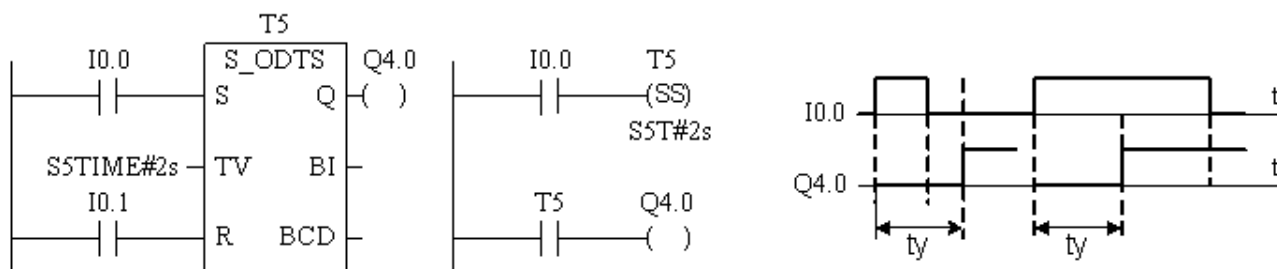


Рис. 1.28. Таймер формирователь задержки включения с запоминанием

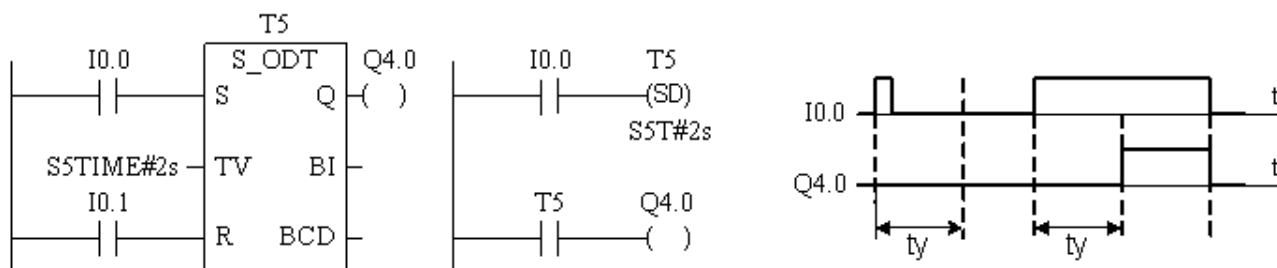


Рис. 1.29. Таймер формирователь задержки включения

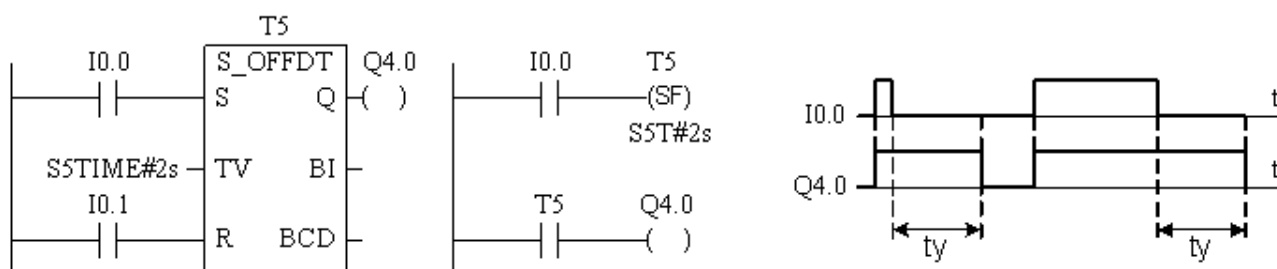


Рис. 1.30. Таймер формирователь задержки выключения

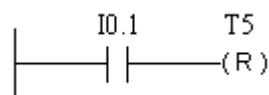


Рис. 1.31. Сброс таймера с адресом T5

## 1.7. Операции счета

Счетчики имеют собственную область памяти в CPU. Для каждого из 256 счетчиков резервируется слово длиной 16 бит, но для счета используется только биты с 0 по 11. Поэтому Счет возможен до 999, то есть уставка счетчика находит-

ся в интервале 0...999. Задание уставки, например, числа 127 осуществляется в следующем формате: C#127, где C# означает BCD код.

В контроллере реализованы 3 вида счетчиков:

- на сложение (прямой счет);
- на вычитание (обратный счет);
- реверсивный (прямой/обратный счет).

На рис. 1.32 представлены блоки всех 3-х видов счетчиков.

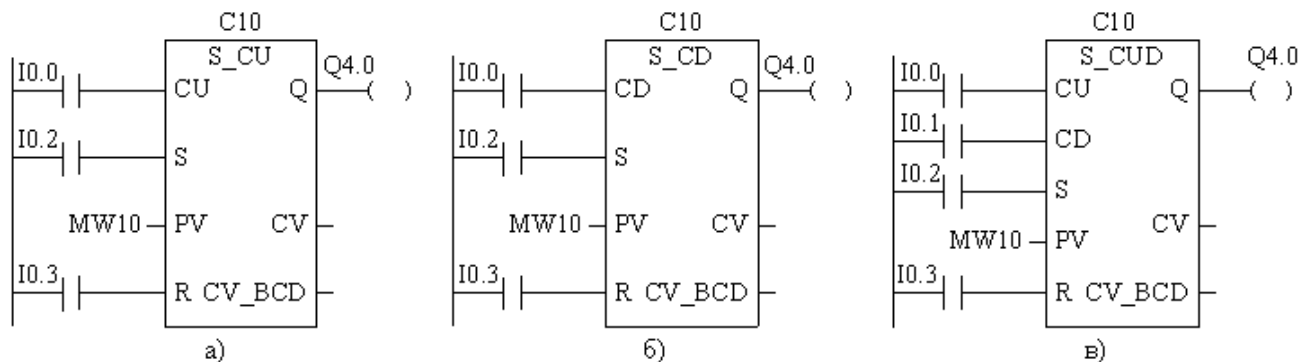


Рис. 1.32. Счетчики: а) на сложение, б) на вычитание, в) реверсивный

На блоках счетчиков рис. 1.32 указаны следующие обозначения: C10 – номер счетчика, S\_CU, C\_CD, S\_CUD – тип счетчика, CU – вход на сложение, CD – вход на вычитание, S – вход предустановки счетчика, PV – задание (уставка) счетчика, R – вход сброса, Q – выход (состояние) счетчика, CV – текущее значение счетчика в двоичном формате, CV\_BCD – текущее значение счетчика в формате BCD.

Уставка счетчика задается либо константой в формате C#<значение>, либо из области памяти (I, Q, M, D). Нарастающим фронтом сигнала на входе S в счетчик заносится уставка с входа PV. При нарастающем сигнале на входе CU значение счетчика увеличивается на «1», если значение счетчика меньше, чем 999. При нарастающем фронте сигнала на входе CD значение счетчика уменьшается на «1», если значение счетчика больше, чем «0». При нарастающем фронте сигнала на входе R счетчик сбрасывается (в счетчик записывается значение «0»).

Для реверсивного счетчика, если оба счетных входа имеют нарастающий фронт, то значение счетчика остается неизменным.

Для всех типов счетчиков выход Q равен «0», если значение счетчика равно «0», и устанавливается в «1», когда значение счетчика отлично от «0».

## 1.8. Операции с аналоговыми сигналами

CPU может обрабатывать аналоговые сигналы только в двоичном коде. Аналоговые входные модули преобразуют аналоговые сигналы в этот код.

Аналоговый модуль SM334, используемый в лабораторной работе, имеет более низкую разрешающую способность, чем другие аналоговые модули, и не имеет области отрицательных значений.

Входы и выходы модуля адресуются, начиная со стартового адреса модуля. Адрес канала получается из начального адреса модуля и адресного смещения.

Входам соответствуют следующие адреса каналов:

Таблица 1.4

Канал	Адрес
0	Начальный адрес модуля
1	Начальный адрес модуля + 2 байта адресного смещения
2	Начальный адрес модуля + 4 байта адресного смещения
3	Начальный адрес модуля + 6 байта адресного смещения

Выходам соответствуют следующие адреса каналов:

Таблица 1.5

Канал	Адрес
0	Начальный адрес модуля
1	Начальный адрес модуля + 2 байта адресного смещения

При выполнении лабораторной работы требуется снять характеристики вход/выход одного канала ввода и одного канала вывода аналогового сигнала. Для этого предлагается ввести программу, представленную на рис. 1.33.

Команда MOV в Network 1 обеспечивает считывание кода напряжения, поданного на канал ввода аналогового сигнала с адресом PIW288 и выдачу этого кода для светодиодной индикации модулем вывода дискретных сигналов по адресу QW5. Для снятия характеристики канала ввода аналогового сигнала мультиметром измеряется напряжение между клеммами – и + модуля SM334 и считывается по светодиодам код числа с байта QW5 модуля SM322 при изменении положения движка потенциометрического резистора, подключенного к каналу ввода.

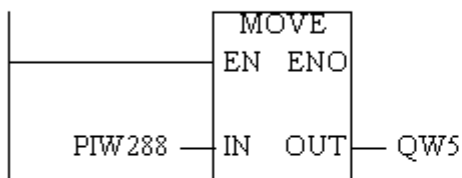
Команда MOV в Network 2 обеспечивает передачу кода набранного на входе IB1 (тумблерами на блоке имитации сигналов команд) на вход цифроаналогового преобразователя (ЦАП) канала вывода аналогового сигнала с адресом PQW288. Для построения характеристики вход/выход канала аналогового вывода считывается код, установленный на входе IB1, и мультиметром измеряется напряжение на аналоговом выходе.

OB1:

Снятие характеристик каналов ввода и вывода модуля SM334

Network: 1

Индикация кода преобразованного аналогового сигнала светодиодами модуля вывода дискретных сигналов по адресу QW5



Network: 2

Передача кода набранного на входах байта IB1 на канал вывода с адресом PQW288 для формирования соответствующего напряжения вывода

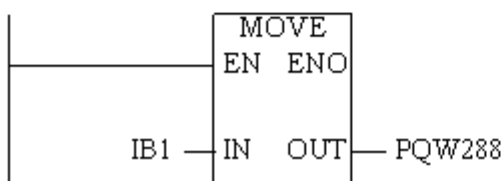


Рис.1.33. Программа для снятия характеристик каналов аналогового ввода и вывода

## 1.9. Программирование функциональных блоков и функций

Язык STEP7 предусматривает возможность упростить программирование, если возникает необходимость в программировании управления однотипными технологическими операциями. Это, например, рассматриваемые ниже испытания двигателей бензинового и дизельного. Испытания в принципе одинаковые, только аппараты (кнопки, датчики и устройства индикации) свои. Тогда централизованно программируется функциональный блок FB, который может многократно вызываться в организационном блоке OB1. Все формальные параметры и статические данные FB сохраняются в отдельном блоке (DB), назначаемом функциональному блоку.

Например, будем программировать функциональный блок FB1, которому дадим символическое имя «Двигатель». При кратковременном появлении команды «Vkl [Включить]» двигатель должен включиться и сформировать сигнал «Vkl\_on [Включен]».

При достижении фактической скорости Fakt заданного значения Zадание формируется сигнал Est. Двигатель отключается по сигналу «Otkl [Отключить]» или по сигналу «Neispr [Неисправность]». Здесь используется английский шрифт, потому что при программировании FB кириллица не воспринимается.

Прежде всего, необходимо определиться с символьным представлением сигналов и команд (табл. 1.6). Для организационного блока OB1 в табл. 1.6 имеется

ряд сигналов, которые будут использованы позднее при рассмотрении программирования функций.

Таблица 1.6

Описание переменных организационного блока OB1

Symbol	Address	Data type	Comment
BD_Bkl	I0.0	BOOL	Включить бензиновый
BD_Otkl	I0.1	BOOL	Отключить бензиновый
BD_Neisprr	I0.2	BOOL	Неисправность бензинового
BD_Fakt	MW0	INT	Фактическая скорость бензинового
BD_Bkl_on	Q4.0	BOOL	Бензиновый включен
BD_Est	Q4.1	BOOL	Заданная скорость достигнута
BD_Bent_on	Q4.2	BOOL	Вентилятор бензинового включен
BD_Timer_Bent	T1	TIMER	Таймер вентилятора бензинового
DD_Bkl	I1.0	BOOL	Включить дизель
DD_Otkl	I1.1	BOOL	Отключить дизель
DD_Neisprr	I1.2	BOOL	Неисправность дизеля
DD_Fakt	MW2	INT	Фактическая скорость дизеля
DD_Bkl_on	Q5.0	BOOL	Дизель включен
DD_Est	Q5.1	BOOL	Заданная скорость достигнута
DD_Bent_on	Q5.2	BOOL	Вентилятор дизеля включен
DD_Timer_Bent	T2	TIMER	Таймер вентилятора дизеля

Для создания FB1 в левой части окна проекта следует вызвать папку Blocks, правой кнопкой мыши вызвать всплывающее меню, в нем выбрать Insert New Object > Function Block и присвоить FB1 имя «Двигатель». В правой части окна проекта возникнет папка FB1.

Дважды щелкните левой кнопкой мыши на FB1. В появившемся диалоговом окне «Properties-Function Block» выберите язык, который хотите использовать, активизируйте триггерную кнопку «Multiple instance FB [Мультиэкземплярный FB]» и подтвердите остальные параметры настройки, щелкнув на ОК. FB1 вставлен в папку блоков Blocks.

Для программирования FB1 необходимо дважды щелкнуть на FB1. Открывается окно для программирования. Следует заполнить таблицу описания переменных (рис. 1.34). Тип переменной (Bool, Int и т.п.) выбирается с помощью всплывающего меню «Elementary Types [Элементарные типы]», используя правую кнопку мыши.



<input type="checkbox"/> Interface	Name	Data Type	Address	Initial Value	Comment
<input checked="" type="checkbox"/> IN	Bkl	Bool	0.0	False	Включить двигатель
•	Otkl	Bool	0.1	False	Отключить двигатель
•	Neispr	Bool	0.2	False	Неисправность
•	Fakt	Int	2.0	0	Текущая скорость

<input type="checkbox"/> Interface	Name	Data Type	Address	Initial Value	Comment
<input checked="" type="checkbox"/> Out	Bkl_on	Bool	4.0	False	Двигатель включен
•	Est	Bool	4.1	False	Есть заданная скорость
•					
•					

<input type="checkbox"/> Interface	Name	Data Type	Address	Initial Value	Comment
•					
•					
<input checked="" type="checkbox"/> STAT	Zadanie	Int	6.0	1500	Заданная скорость
•					

Рис. 1.34. Описание переменных функционального блока FB1

Затем программируется включение и отключение двигателя, как было описано выше. Выделите вопросительные знаки и введите соответствующие имена из таблицы описания переменных (знак «#» назначается автоматически). Помеченные знаком «#» локальные переменные действительны только в этом блоке.

Набранная программа на языке лестничных диаграмм имеет вид, представленный на рис. 1.35.

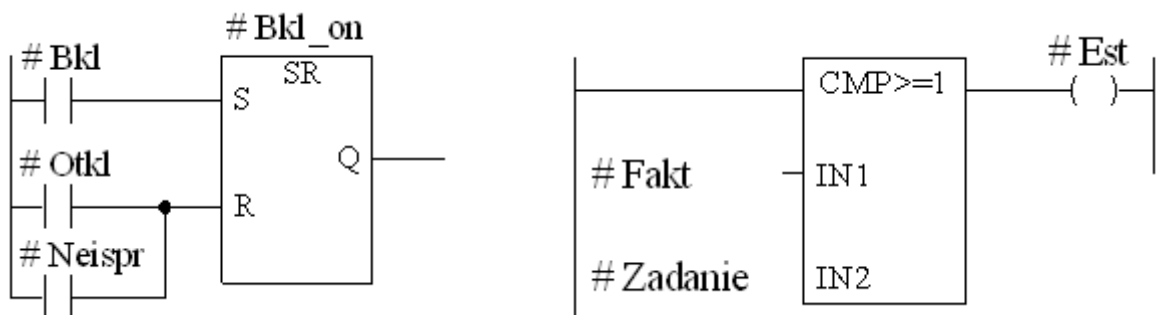


Рис. 1.35. Программа функционального блока FB1

Чтобы получить возможность вызова организованного блока FB1 в OB1 необходимо сгенерировать соответствующий блок данных. Экземплярный блок данных (DB) всегда ставится в соответствие функциональному блоку.

Рассматриваемый FB1 должен управлять и контролировать работу бензинового и дизельного двигателей. Различные заданные скорости хранятся в отдельных блоках данных.

В левой части окна проекта необходимо переместиться в папку Blocks и в правой половине окна щелкнуть правой кнопкой мыши. Используя всплывающее меню, выбрать Insert New Object > Data Block. Задайте имя блоку данных DB1 «Бензиновый», выберите тип блока «Instance DB [Блок экземпляра]» и поставьте ему в соответствие блок FB1. Примите все параметры настройки, отображаемые в диалоговом окне «Properties [Свойства]», щелкните на ОК. Блок данных DB1 добавляется к проекту. Дважды щелкните на значке DB1, чтобы открыть его. Подтвердите назначение блока экземпляра для FB1 с помощью ОК.

Затем введите значение 1500 для бензинового двигателя в столбец «Actual Value [Фактическое значение]» в строке Zадание. Теперь определена максимальная скорость для этого двигателя. Сохранить DB1 и закройте блок.

Так же, как и в случае с DB1, сгенерируйте еще один блок данных DB2 для FB1. Теперь введите значение 1200 для дизельного двигателя.

Следующий шаг – это запрограммировать вызов функционального блока в OB1. Это демонстрируется на рис. 1.36.

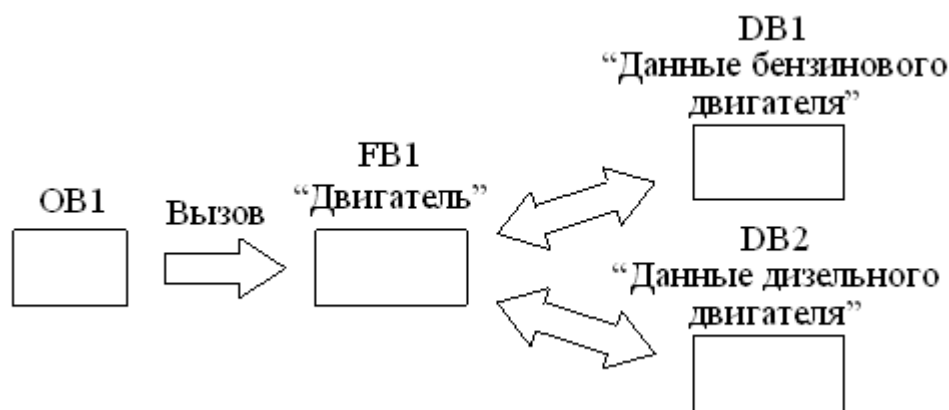


Рис. 1.36. Вызов блока FB1 в OB1

Необходимо открыть SIMATIC Manager с вашим проектом. Переместиться к папке Blocks и открыть OB1. Вставить сегмент программы. Затем перемещаетесь в каталог элементов программы, пока не достигнете папки функциональных блоков. Выделите FB1 и вставьте его в OB1.

Вставьте контакты перед каждым из следующих входов Bkl, Otkl и Neispr. Щелкните на знаке «???» над блоком «Двигатель», а затем, удерживая курсор в том же положении, щелкните правой кнопкой мыши, выберите во всплывающем меню Insert Symbol. Щелкните на блоке данных «Бензиновый». Этот блок затем автоматически вводится в рамку ввода в кавычках.

Выделяя вопросительные знаки, назначьте адреса всем остальным параметрам функционального блока, используя соответствующие символьные имена из табл. 1.4.

Вызовите новый сегмент и в нем вызовите функциональный блок «Двигатель» (FB1) с блоком данных «Дизель» и используйте для него соответствующие символьные адреса.

Набранные программы будут иметь вид, представленный на рис. 1.37.

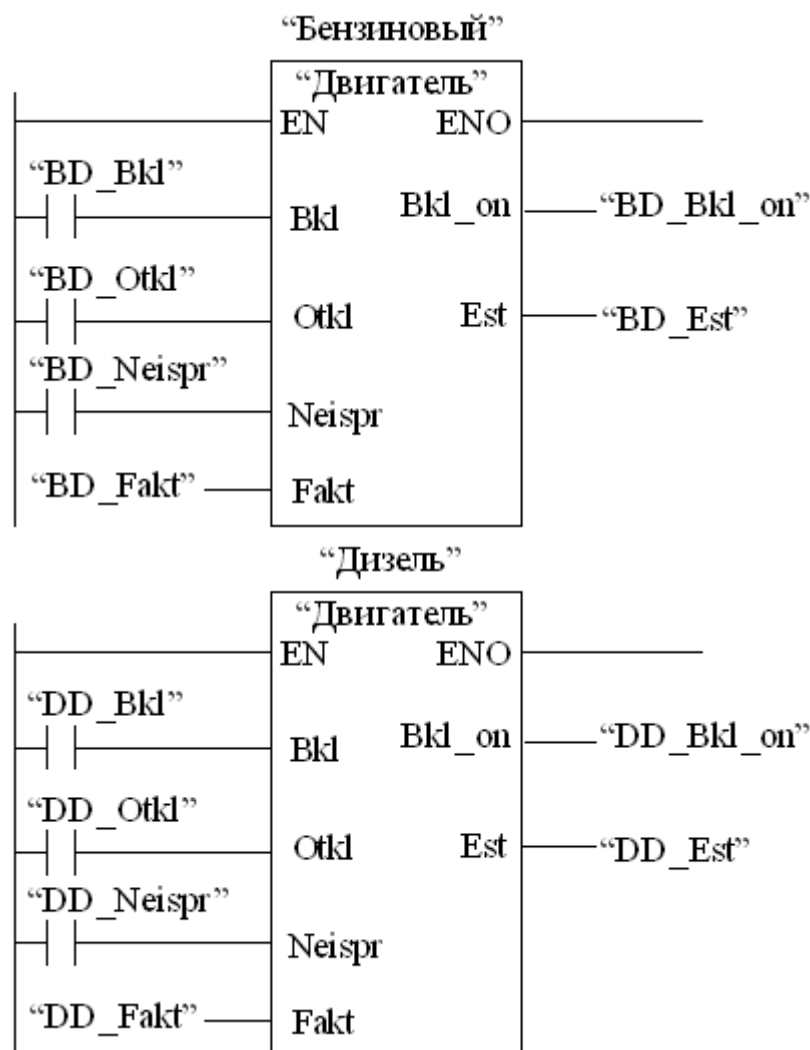


Рис. 1.37. Программирование вызова функционального блока FB1 в OB1

Сохраните свою программу и закройте блок OB1.

Функции, как и функциональные блоки, расположены в иерархии программы ниже организационного блока. Однако в отличие от функционального блока, функции не нужен блок данных. У функций параметры также перечисляются в таблице описания переменных, но статические данные не разрешаются.

Для создания функции в левой части окна проекта, переместитесь к папке Blocks и откройте ее. Щелкните правой кнопкой мыши в правой половине окна. Во всплывающем меню выбрать Insert New Object > Function. В диалоговом окне «Properties – Function [Свойства – Функция]» задайте имя FC1, в нашем случае «Вентилятор», и выберите нужный язык программирования. Подтвердите остальные параметры настройки с помощью ОК. Функция FC1 добавляется в папку Blocks.

В рассматриваемой задаче предусматривается, что каждый двигатель как бензиновый, так и дизель имеет свой вентилятор. Вентилятор включается при ра-

боте двигателя и продолжает еще работать 4 с после отключения двигателя. Для управления этими двигателями программируется функция FC1.

Дважды щелкните на папке FC1, чтобы открыть ее. Открывается окно для программирования. Введите описание переменных (рис. 1.38).

<input type="checkbox"/> Interface	Name	Data Type	Comment
<input type="checkbox"/> IN	Bkl_on	Bool	Двигатель включен
•	Timer_Bent	Timer	Задержка на отключение
•			
•			
•			

<input type="checkbox"/> Interface	Name	Data Type	Comment
<input type="checkbox"/> Out	Bent_on	Bool	Вентилятор включен
•			
•			
•			
•			

Рис. 1.38. Описание переменных функций

Затем выделите шину для ввода команд лестничных диаграмм и запрограммируйте управление вентилятором. Выделяйте вопросительные знаки и вводите соответствующие имена из таблицы описания переменных (знак # назначается автоматически). Набранная программа на языке лестничных диаграмм имеет вид, представленный на рис. 1.39.

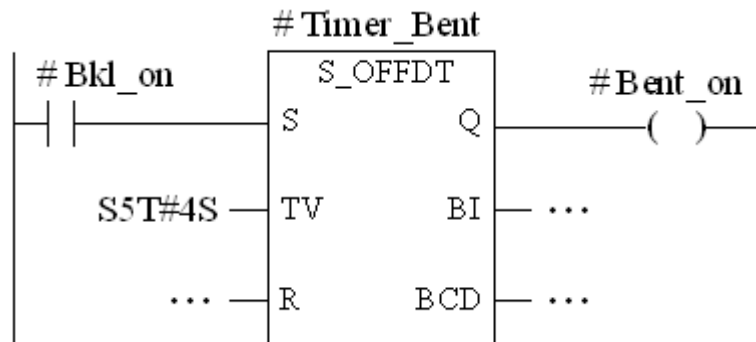


Рис. 1.39. Программирование функции FC1

Вызов функции FC1 в OB1 выполняется аналогично вызову функционального блока. Все переменные функции снабжаются в OB1 соответствующими адресами бензинового или дизельного двигателя, которые уже отражены в табл. 1.4.

Откройте свой проект SIMATIC Manager. В левой части окна проекта переместитесь к папке Blocks и откройте в правой части окна проекта OB1.

Вставьте новый сегмент программы. Затем перемещайтесь в каталоге элементов программы, пока не достигнете папки функций FC Blocks. Выделите FC1

и вставьте ее в OB1. Вставьте замыкающий контакт перед «Bkl\_on». Выделив вопросительные знаки при вызове FC1, введите символьные имена из табл. 1.4 для бензинового двигателя.

Вызовите очередной сегмент программы и запрограммируйте вызов функции FC1, используя адреса для дизельного двигателя.

Набранные программы будут иметь вид, представленный на рис. 1.40.

Сохраните блок и закройте окно.

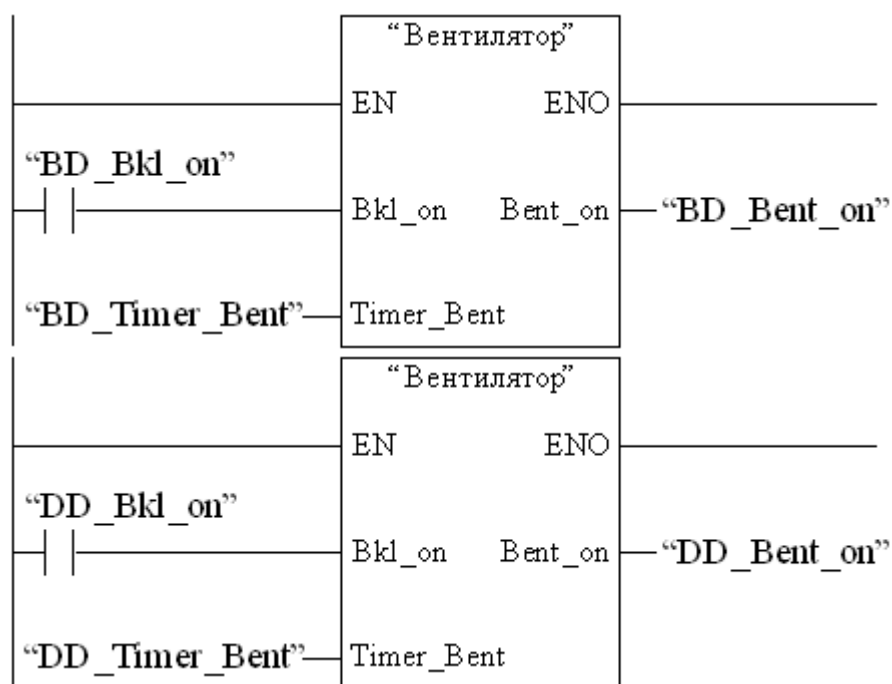


Рис. 1.40. Программирование вызова функции FC1 в OB1

### 1.10. Пример программирования контроллера SIMATIC S7-300

В качестве примера рассмотрим управление контроллером SIMATIC S7-300 движением тележки.

Тележка перемещается по прямолинейному пути. Имеется два фиксированных положения П0 и П1. Исходное положение тележки – П0.

При кратковременном нажатии на кнопку «Пуск» тележка движется вперед до положения П1, стоит в положении П1 5 с и возвращается назад в положение П0. В положении П0 немедленно возникает реверс привода тележки. Тележка вновь движется в положение П1, стоит в нем 5 с и возвращается в П0. В положении П0 вновь возникает команда на движение до положения П1 и т.д. после 10 циклов перемещения в положение П1 при возвращении тележки в положение П0 возникает сигнал «Конец цикла».

Для управления движением тележки следует предусмотреть счетчик (СЧ) со счетом до 10, задержку времени (таймер) для формирования задержанного сигнала  $\Pi 1^\uparrow$  и память  $P_\Pi$  о нажатии на кнопку «Пуск».

$$P_\Pi = (\text{Пуск} + P_\Pi) \cdot \overline{КЦ}$$

Команда на перемещение тележки вперед (В) возникает в положении  $\Pi 0$ , если отсутствует сигнал счетчика СЧ, и сохраняется при наличии сигнала памяти пуска  $P_\Pi$  до достижения положения  $\Pi 1$ . Этому описанию соответствует логическое уравнение

$$B = (\Pi 0 \cdot \overline{СЧ} + B) \cdot \overline{\Pi 1} \cdot P_\Pi$$

Команда на перемещение тележки назад (Н) возникает через 5 с нахождения в положении  $\Pi 1$  и сохраняется при наличии переменной  $P_\Pi$  до достижения положения  $\Pi 0$ . Тогда логическое уравнение для команды Н имеет вид

$$H = (\Pi 1^\uparrow + H) \cdot \overline{\Pi 0} \cdot P_\Pi$$

Сигнал конца цикла (КЦ) возникает в положении  $\Pi 0$  при наличии сигнала счетчика СЧ о завершении 10 циклов перемещения в положение  $\Pi 1$  и сигнала  $P_\Pi$ . Сигнал КЦ сохраняется до очередного нажатия на кнопку «Пуск». Тогда логическое уравнение для рассматриваемого сигнала имеет вид

$$КЦ = (\Pi 0 \cdot СЧ \cdot P_\Pi + КЦ) \cdot \overline{\text{Пуск}}$$

В ниже приведенной программе (рис. 1.41) использован счетчик на вычитание. При нажатии на кнопку «Пуск» в счетчик вводится уставка 10. Счетчик считает число циклов перемещения в положение  $\Pi 1$ , при достижении этого положения число в счетчике уменьшается на единицу. Счетчик сбрасывается по сигналу конца цикла КЦ. Признаком окончания цикла является число в счетчике равное нулю, при котором на выходе счетчика «0». Поэтому в уравнениях команд В и КЦ необходимо использовать инверсию сигнала СЧ по отношению к ранее записанным уравнениям

$$B = (\Pi 0 \cdot СЧ + B) \cdot \overline{\Pi 1} \cdot P_\Pi$$

$$КЦ = (\Pi 0 \cdot \overline{СЧ} \cdot P_\Pi + КЦ) \cdot \overline{\text{Пуск}}$$

Сброс счетчика производится сигналом конца цикла КЦ.

В табл. 1.7 представлена адресация команд и сигналов для контроллера SIMATIC S7-300.

Таблица 1.7

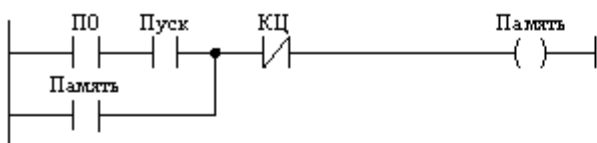
Команда, сигнал	$\Pi 0$	$\Pi 1$	Пуск	В	Н	КЦ	$P_\Pi$	СЧ	$\Pi 1^\uparrow$
Адрес	I0.0	I0.1	I0.2	Q4.0	Q4.1	Q4.2	M0.0	M0.1	M0.2
Символ	$\Pi 0$	$\Pi 1$	Пуск	Вперед	Назад	КЦ	Память	СЧ	Задержка

ОВ1:

Управление движением тележки

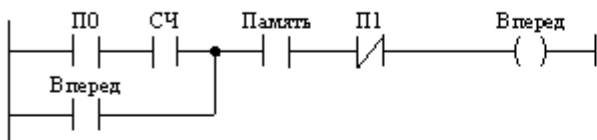
Network: 1

Память пуска



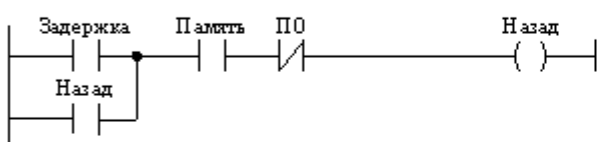
Network: 2

Движение вперед



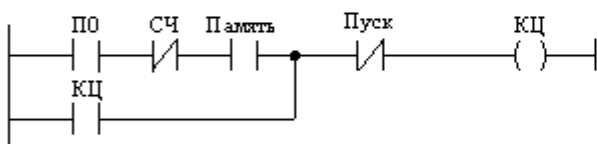
Network: 3

Движение назад



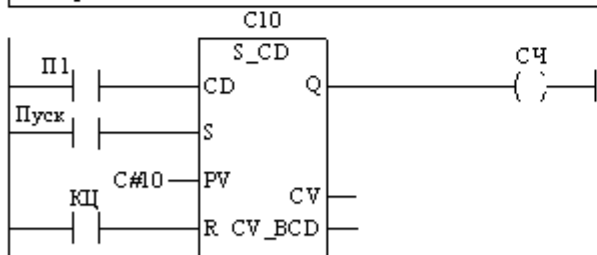
Network: 4

Конец цикла



Network: 5

Управление счетчиком



Network: 6

Управление таймером с задержкой на включение

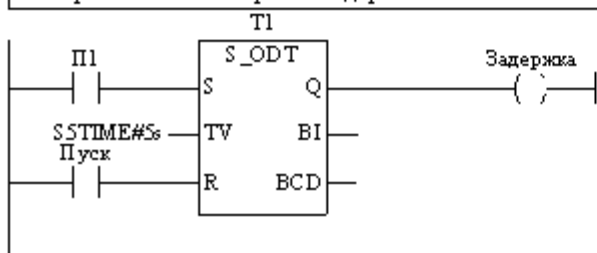


Рис. 1.41. Программа управления движением тележки

## 1.11. Загрузка программы в контроллер и запуск программы

Перед загрузкой программы необходимо сбросить память в CPU. Это выполняется в следующей последовательности:

– поверните переключатель режимов работы в положение MRES. Удерживайте переключатель в этом положении, пока светодиод STOP не загорится во второй раз и затем останется гореть (это занимает 3 с);

– в течение 3 с необходимо повернуть переключатель обратно в положение MRES. Светодиод STOP начинает быстро мигать и CPU производит сброс памяти. Когда светодиод STOP снова начинает гореть непрерывно, это означает, что CPU завершил сброс памяти.

Более подробно о стирании памяти, в частности, о стирании по запросу со стороны CPU и форматировании его сменной платы микропамяти SIMATIC (MMC) представлено в п.2.6 лабораторной работы №2 «Изучение промышленной сети PROFIBUS-DP».

Загрузка программы в контроллер осуществляется в следующей последовательности:

– переместитесь через SIMATIC 300 Station CPU314 к S7 Program. В SIMATIC Manager щелкните на символе «Blocks [Блоки]» в правой части окна. Ключ режима контроллера поставьте в положение STOP;

– из меню PLC (ПЛК) выберите команду «Download [Загрузить]», чтобы передать программу и конфигурацию аппаратуры в CPU. Щелкаете на Yes во всех появляющихся диалоговых окнах.

Для запуска программы необходимо перевести ключ режима работы контроллера в положение «RUN».

Во время работы программы можно следить за ходом ее выполнения. Переведя переключатель режима работы контроллера в положение «RUN» и нажав на кнопку с изображением очков.

При этом будут отображаться состояния контактов и катушек, а также значения счетчиков, таймеров в реальном времени.

## 1.12. Выполнение лабораторной работы

При подготовке к лабораторной работе студент должен в соответствии с табл. 1.8 выбрать свой вариант задания.

Таблица 1.8

Номер бригады	1 и 5			2 и 6			3 и 7			4 и 8		
	1	2	3	1	2	3	1	2	3	1	2	3
Номер варианта	1	2	3	4	5	6	7	8	9	10	11	12



Для заданного варианта необходимо разработать алгоритм ее решения. В соответствии с заданием определиться с адресацией входных и выходных переменных. Затем определиться с адресацией вводимых промежуточных переменных и с адресацией используемых таймеров и счетчиков. Составить программу для контроллера SIMATIC S7-300.

Запустить программу в работу и убедиться в правильности ее работы. При наличии ошибок в работе устранить их и продемонстрировать преподавателю работу правильно функционирующей программы.

Программу для снятия характеристик аналоговых каналов ввода/вывода можно вписать в программу своего варианта задания. Для этого следует учесть, что для индикации кода преобразованного напряжения канала аналогового ввода необходимо предусмотреть свободный байт модуля вывода дискретных сигналов, например, QW5.

*Вариант 1.* Перекладыватель служит для перемещения листов со стола на постоянно вращающийся рольганг. Перекладыватель имеет подъемник листов, установленный на тележке. Подъемник представляет собой поперечину, перемещающуюся по вертикальным стойкам сверху вниз и обратно. Поперечина имеет на конце электромагнит для притягивания листов.

В исходном положении перекладыватель стоит в положении П0 над столом, а его подъемник в крайнем верхнем положении КВ. При поступлении листа на стол подается сигнал наличия листа, подъемник включается для движения вниз. При касании листа подъемник останавливается и включается электромагнит. Через 2 с включается подъемник на подъем, достигает положения КВ и останавливается. Включается тележка и перемещается в положение П1 над рольгангом, где останавливается и электромагнит отключается. Лист падает на рольганг. Через 2 с тележка движется в положение П0. Цикл повторяется при поступлении очередного сигнала наличия листа. Включение системы в работу – нажатием кнопки «Пуск».

*Вариант 2.* В контроллере программным путем необходимо реализовать генератор импульсов. Время наличия импульса 1 с, время его отсутствия 2 с. При нажатии на кнопку «Пуск» начинается счет импульсов. При прохождении 10 импульсов загорается лампа Л1, при прохождении следующих 10 импульсов – лампа Л2, аналогично включаются лампы Л3 и Л4. После загорания лампы Л4 (т.е. после прохождения 40 импульсов) счет прекращается и все лампы горят до нажатия на кнопку «Стоп». Генератор импульсов реализовать программным путем.

*Вариант 3.* Автоматизировать перемещение пунсона штампованного пресса. Рабочий заправляет металлическую ленту (заготовку) и нажимает кнопку «Пуск». Пуансон из крайнего верхнего положения П0 движется вниз, при своем движении до положения П1 входит в матрицу, выбивая из ленты деталь необходимой фор-

мы, и возвращается в положение П0. Чисто механическим устройством лента перемещается на один шаг и через 2 с пуансон совершает очередное движение. Лента рассчитана на изготовление 10 деталей, поэтому через 10 ходов пуансона формируется сигнал «Конец цикла».

При подаче питания на систему автоматизации пуансон из любой точки должен прийти в положение П0.

*Вариант 4.* На программируемом контроллере реализовать систему управления толкателем с кривошипно-шатунным механизмом с реверсивным электродвигателем.

Толкатель имеет два фиксированных положения П0 и П1. При подаче питания на систему толкатель из любого положения движется вперед медленно до положения П0. При появлении кратковременного сигнала «Пуск» толкатель из положения П0 медленно движется до положения П1, стоит там в течение 2 с, а затем электродвигатель включается для движения назад быстро и возвращается в положение П0, где формируется сигнал «Конец цикла». Цикл повторяется при нажатии кнопки «Пуск».

При нажатии кнопки «Стоп» толкатель из любого положения на быстрой скорости возвращается в положение П0.

*Вариант 5.* Автоматизировать управление крышкой котла (типа большой кастрюли).

При кратковременном нажатии кнопки «Открыть» крышка поднимается до крайнего верхнего положения, в котором поворачивается, открывая доступ к содержимому котла.

При кратковременном нажатии на кнопку «Заккрыть» крышка поворачивается в положение точно над котлом и затем опускается, закрывая котел.

Для исключения ударов предусмотреть переход на пониженную скорость перед касанием крышкой котла при его закрывании.

*Вариант 6.* Автоматизировать управление дверью (воротами) въезда/выезда гаража.

Нормальное (исходное) положение двери закрытое. При кратковременном нажатии кнопки «Открыть» включается звонок и через 5 с включается привод двери на открывание. При полном открытии двери привод двери отключается сразу, а звонок звенит еще 2 с.

Закрывание дверей – аналогично: при кратковременном нажатии кнопки «Заккрыть» включается звонок, через 5 с включается привод двери на закрывание. При полном закрытии двери привод двери отключается, а звонок звенит еще 2 с.

*Вариант 7.* Печь сопротивлений содержит три нагревательных элемента (НЭ), каждый из которых питается от собственного тиристорного преобразователя соответственно ТП1...ТП3. Сигнал «1» на входе ТП соответствует максималь-

но допустимому току НЭ, сигнал «0» – минимально допустимому току НЭ. Синтезировать схему, обеспечивающую алгоритм работы группы ТП, представленный в табл. 1.9.

Таблица 1.9

Номер импульса генератора	-	1	2	3	4	5	6	7	8	9	...
Состояние ТП1	0	1	0	0	1	1	0	1	0	0	...
Состояние ТП2	0	0	1	0	0	1	1	0	1	0	
Состояние ТП3	0	0	0	1	1	0	1	0	0	1	

Начало работы осуществляется после нажатия на кнопку «Пуск».

Цикл управления печью сопротивления состоит из 6 тактов. Переход с одного такта на другой осуществляется при поступлении очередного импульса от генератора импульсов. Циклы управления непрерывно следуют друг за другом.

При поступлении команды «Стоп» прекращается подача импульсов и все ТП отключаются. Для возобновления работы необходимо нажать на кнопку «Пуск». Генератор импульсов реализуется программным путем.

*Вариант 8.* Манипулятор (рука со схватом) служит для подачи заготовок из накопителя в пресс. Рука манипулятора может перемещаться вперед/назад из крайнего заднего КЗ в крайнее переднее КП положение и назад, а также поворачиваться из крайнего правого положения КПр в крайнее левое положение КЛ и обратно. Схват имеет электромагнитный механизм зажима/разжима заготовок.

Исходное положение манипулятора перед накопителем (в положениях КЗ и КПр, схват под действием пружин зажат). При поступлении запроса со станка и наличии заготовки в накопителе включается электромагнит разжима схвата. Рука идет в крайнее переднее положение КП, через 2 с схват зажимает заготовку (электромагнит схвата отключается) и рука возвращается в положение КЗ. Рука поворачивается в положение КЛ и выдвигается вперед до КП. Схват разжимается (электромагнит схвата включается) и через 2 с рука идет в положение КЗ, затем поворачивается в положение КПр (исходное положение руки). Если есть запрос со станка, а накопитель пуст, то включается сигнальная лампа «Нет детали» и кратковременный звуковой сигнал (в течение 5 с). Световой сигнал снимается кнопкой «Сброс». Включение системы в работу – нажатием кнопки «Пуск».

*Вариант 9.* Автоматизировать процесс безупорной остановки длинномерного металла на рольганге Р1.

На рольганг Р1 металл передается с предыдущего рольганга Р0, наличие металла на котором контролируется датчиком Д0. Вдоль рольганга Р1 расположены датчики, контролирующие наличие металла последовательно в положениях П1, П2 и П3.

Нормальное положение рольганга Р1 – неподвижное. При наличии сигнала с датчика Д0 и отсутствии металла на рольганге Р1, рольганг Р1 включается и ме-

талл перемещается по рольгангу Р1. При достижении передним концом металла положения П1 происходит снижение скорости Р1 до «ползучей» скорости, а в положении П2 привод рольганга Р1 отключается. Если по каким-либо причинам металл достиг положения П3, то привод рольганга Р1 реверсируется и работает на пониженной скорости, пока металл не уйдет из положения П3. При остановке переднего конца металла между положениями П2 и П3 формируется сигнал разрешения работы механизмов уборки металла с Р1.

Включение системы в работу – при нажатии на кнопку «Пуск».

*Вариант 10.* Автоматизировать работу толкателя с кривошипно-шатунным механизмом с нереверсивным электродвигателем.

При нажатии кнопки «Пуск» кривошип из положения  $\alpha = 0^\circ$  движется до положения  $\alpha = 120^\circ$ . В этом положении происходит переход на пониженную (малую) скорость, кривошип идет до положения  $\alpha = 180^\circ$  и останавливается. Через 3с электродвигатель автоматически вновь включается и на повышенной скорости шатун идет в положение  $\alpha = 0^\circ$ . Стоит в нем 3с, включается и движется к положению  $\alpha = 120^\circ$  и на пониженной скорости до  $\alpha = 180^\circ$  и т.д. После 5 толканий толкатель останавливается в исходном положении (кривошип - в положении  $\alpha = 0^\circ$ ) и формируется сигнал «Конец цикла».

Очередной цикл из пяти толканий – после очередного нажатия на кнопку «Пуск».

При нажатии на кнопку «Стоп» толкатель из любого положения на повышенной скорости движется в положение  $\alpha = 0^\circ$  и останавливается.

*Вариант 11.* Тележка движется по кольцевому пути, на котором есть 3 фиксированных положения П0, П1 и П2. Привод тележки – от нереверсивного электродвигателя.

При подаче питания из любого положения тележка должна идти в положение П0. При нажатии на кнопку «Пуск» тележка без задержки движется в положение П1, стоит там в течение 3с, затем движется в П2, стоит там в течение 4с, движется в положение П0, стоит там в течение 5с, движется в П1, стоит там в течение 3с и т.д. То есть тележка непрерывно движется по кольцевому пути с остановками в фиксированных положениях.

При нажатии на кнопку «Стоп» тележка должна без остановки идти в П0 и там ожидать очередного нажатия кнопки «Пуск».

*Вариант 12.* Автоматизировать линию сортировки изделий. По конвейеру движутся низкие изделия, но редко могут встречаться и высокие. По ходу конвейера установлены две фотоголовки на расстоянии друг от друга, равном половине ширины изделия. Первая по ходу конвейера фотоголовка настроена на низкие изделия (нижняя фотоголовка), вторая – на высокие изделия (верхняя фотоголовка).

При прохождении низкой детали перекрывается только нижняя фотоголовка, а при высоких – вначале нижняя, а затем верхняя фотоголовки. На выходе кон-

вейера детали ударяются о сортировочную планку и высокие изделия поступают в правый накопитель (левое положение планки), низкие – в левый (правое положение планки). Планка перебрасывается пневмоприводом. Для переброски планки влево или вправо достаточно хотя бы кратковременно подать команду на ее перемещение влево или вправо. Включение системы в работу – нажатием кнопки «Пуск».

### 1.13. Требования к отчёту

Отчет должен содержать:

- а) цель работы;
- б) условия задачи по варианту, принятые обозначения переменных, логические функции для выходных и промежуточных переменных, при необходимости циклограмму работы оборудования;
- в) логические функции в адресах программируемого контроллера;
- г) программу для реализации системы управления;
- д) методику экспериментальной проверки функционирования реализованной системы управления и результаты проверки;
- е) характеристики вход-выход каналов ввода и вывода аналогового модуля SM 334;
- ж) выводы по работе.

### 1.14. Контрольные вопросы

1. Что означает термин «конфигурирование контроллера»?
2. Что означает термин «заказной номер модуля контроллера», где и как он используется?
3. В какой последовательности располагаются модули в слотах конфигурационной таблицы?
4. Что необходимо сделать при покидании окна конфигурирования?
5. Что такое таблица символов и как она используется при программировании контроллера?
6. Когда и как осуществляется сброс памяти CPU?
7. Можно ли в SIMATIC S7-300 реализовать таймер с уставкой времени 0,07 с?
8. При каких условиях на выходе счетчиков контроллера формируется сигнал «0» и сигнал «1»?
9. Как снять характеристику вход-выход аналогового канала ввода?
10. Как снять характеристику вход-выход аналогового канала вывода?
11. Что означает создание функционального блока (FB) при программировании контроллера?
12. Что означает создание и открытие функции (FC) при программировании контроллера?

## **2. РАБОТА №2. ИЗУЧЕНИЕ ПРОМЫШЛЕННОЙ СЕТИ PROFIBUS-DP**

### **2.1. Цель работы**

Ознакомиться с принципом построения промышленной сети PROFIBUS-DP и приобрести навыки программирования обмена информацией между компонентами сети.

### **2.2. Содержание работы**

а) Изучить состав представленной на лабораторном стенде промышленной сети PROFIBUS-DP, назначение и технические характеристики компонентов сети.

б) Изучить конфигурирование сети, программирование обмена информацией между компонентами сети и принципы программирования задач управления технологическими процессами.

в) Дома при подготовке к работе:

– выполнить для заданного варианта синтез системы автоматизации объектом согласно заданного варианта, сигналы управления которым поступают с разных компонентов сети, а информация о его работе также передается с объекта на разные компоненты сети;

– составить программу управления контроллерами Master'ом и Slave'ом.

г) В лаборатории:

– сконфигурировать контроллеры Master и Slave, записав для себя выставляемые при конфигурировании адреса компонентов сети и их модулей;

– учитывая установленные при конфигурировании адреса, откорректировать подготовленные дома программы, записать их в менеджере проектов «Simatic Manager» и запомнить;

– загрузить программы в программируемые контроллеры и запустить сеть в работу;

– используя кнопки и тумблеры «Блока имитационных сигналов» лабораторного стенда, подавать (имитировать) сигналы управляющих команд и датчиков и, наблюдая по светодиодам указанного блока ответную реакцию имитируемого объекта автоматизации, оценить правильность его функционирования.

### **2.3. Описание лабораторного стенда**

На рис. 2.1 представлена структура лабораторного стенда.

Лабораторный стенд позволяет производить изучение системы автоматизации, построенной на промышленной сети PROFIBUS-DP.

Лабораторный стенд включает в себя:

– модуль «Блок питания» на базе фирменного источника питания PS 307 на ток 5 А;

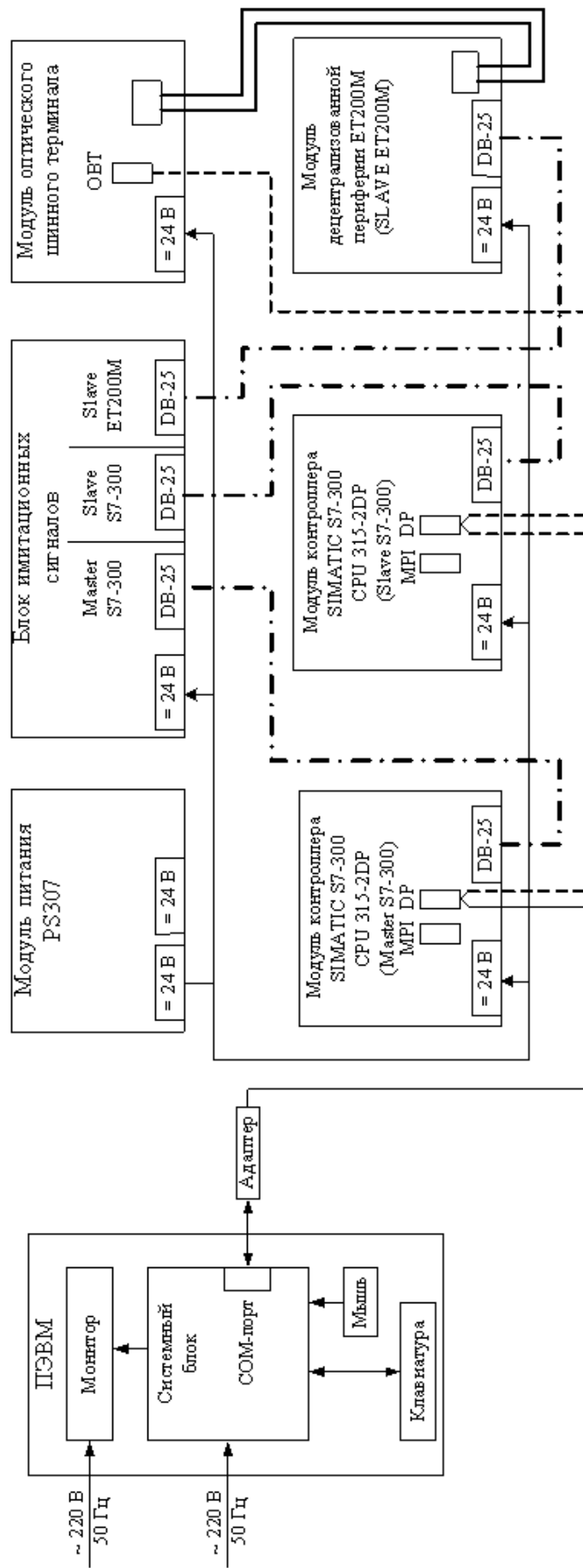


Рис. 2.1. Структурная схема лабораторной установки

– модуль «Контроллер SIMATIC S7-300 (Master S7-300)», включающий в себя CPU 315-2DP (заказной номер 315-2G10-OABO), модуль дискретного ввода/вывода SM323 DI8/DO8xDC24V (заказной номер 323-1BH01-OAAO) и модуль дискретного вывода SM322 DO8xDC24V/2A (заказной номер 322-1BF01-OAAO);

– модуль «Контроллер SIMATIC S7-300 (Slave S7-300)», включающий в себя CPU 315-2DP (заказной номер 315-2G10-OABO), модуль дискретного ввода DI16xDC24V (заказной номер 321-1BH50-OAAO) и модуль дискретного вывода SM322 DO16xDC24V/0,5A (заказной номер 322-1BH10-OAAO);

– модуль «Децентрализованная периферия ET200M (Slave ET200M)», включающий в себя интерфейсный модуль IM153-2F O (заказной номер 153-2BBOO-OXBO) и модуль ввода/вывода SM323 DI16/DO16xDC24V (заказной номер 323-1BL00-OAAO);

– модуль «Оптический шинный терминал ОВТ»;

– модуль «Блок имитационных сигналов»;

– ПЭВМ типа IBM (системный блок, монитор, клавиатура, мышь);

– адаптер для связи ПЭВМ с промышленной сетью;

– соединительные кабели.

Для полноты представления сети в лабораторной установке используются как проводные кабели типа витой пары интерфейса RS485, так и пластиково-оптический кабель. Оптический сетевой терминал ОВТ обеспечивает переход от проводного к пластиково-оптическому кабелю. На схеме рис. 2.1 показано:

– сплошными линиями – кабели низковольтного питания постоянным напряжением 24 В;

– штриховыми линиями кабели проводные (витая пара) сети PROFIBUS-DP;

– сплошной двойной толстой линией – пластиково-оптический кабель;

– штрихпунктирной линией – связь программируемых контроллеров и ET200M с блоком имитационных сигналов.

Программирование программируемых контроллеров осуществляется по сети через разъемы DP процессорных модулей CPU, поэтому адаптер связи с компьютером подключен непосредственно к сети.

## 2.4. Конфигурирование сети в STEP7 v 5.3

Работа с программным проектом осуществляется в менеджере проектов – «SIMATIC Manager». Для запуска «SIMATIC Manager» необходимо на рабочем столе Windows дважды щелкнуть курсором мыши по ярлыку:



–откроется окно «SIMATIC Manager» (рис. 2.2).

!!! Вам будет предложено воспользоваться мастером по созданию проекта одной системы (упрощенный вариант), необходимо отказаться, нажав «Cancel [отменить]».

Для создания нового проекта необходимо в главном меню выбрать «File [Файл]» «New...». Задайте имя проекта и покиньте окно. Откроется окно нового проекта.






Рис. 2.2. Окно SIMATIC Manager

Далее необходимо создать объекты: выделив проект в правой части окна и открыв контекстное меню с помощью правой клавиши мыши, выбрать «Insert new object [Вставить новый объект]». Вставляем в проект станцию «SIMATIC 300 Station». Вновь вставленный объект появится в правой половине. Здесь, так же, как у всех других объектов можно присваивать специфическое имя объекту. Присваиваем, например, «Master S7-300». Вставляем еще одну станцию «SIMATIC 300 Station» и сеть «PROFIBUS», даем имена «Slave S7-300» и «Profibus (1)» соответственно.

Конфигурирование аппаратной части «Master`a S7-300» производится при нажатии на надпись «Master S7-300» в левой части окна. В правой части окна появится значок:

 Hardware – [Оборудование]. Щелкните по нему, появится окно конфигурирования оборудования – «HW Config».

В правой части окна располагается окно «Hardware Catalog [Каталог аппаратуры]». Добавление модулей производится из этого каталога, содержащий список компонентов оборудования SIMATIC.

В каталоге выберите для вашей структуры в папке SIMATIC 300/RACK-300 центральную стойку «Rail [профильную шину]».

Отбуксируйте стойку, используя Drag&Drop, в окно станции. Стойка появляется в виде небольшой конфигурационной таблицы в верхней части окна станции. В нижней части окна станции появляется подробное представление стойки с дополнительными данными.

В качестве альтернативы можно дважды щелкнуть левой кнопкой мыши на выбранном модуле, и он появится в конфигурационной таблице.

Основное поле (рис. 2.3) содержит сведения о CPU и модулях системы:

- 1) Slot – позиция модуля в системы (3-я позиция используется только под интерфейсный модуль);
- 2) Module – тип модуля или CPU;
- 3) I address – возможные адреса модуля ввода или CPU;
- 4) Q address – возможные адреса модуля вывода или CPU;
- 5) Comment – строка для занесения комментариев.

На стойку (рис. 2.3) устанавливается блок питания PS (при необходимости) и модули (например, CPU) из окна «Hardware Catalog». Слоты, в которые можно установить тот или иной модуль выделяются цветом. Отбуксируйте модули, используя Drag&Drop или двойным щелчком левой кнопкой мыши, в соответствующие строки стойки (конфигурационная таблица).

!!! Когда левой кнопкой мыши выделяете модуль в каталоге, в нижней части каталога появляется заказной номер модуля, который нанесен на корпусе модуля.

Необходимо быть внимательным, чтобы заказной номер модуля, устанавливаемого в конфигурационную таблицу, соответствовал модулю, установленному реально на стенде.

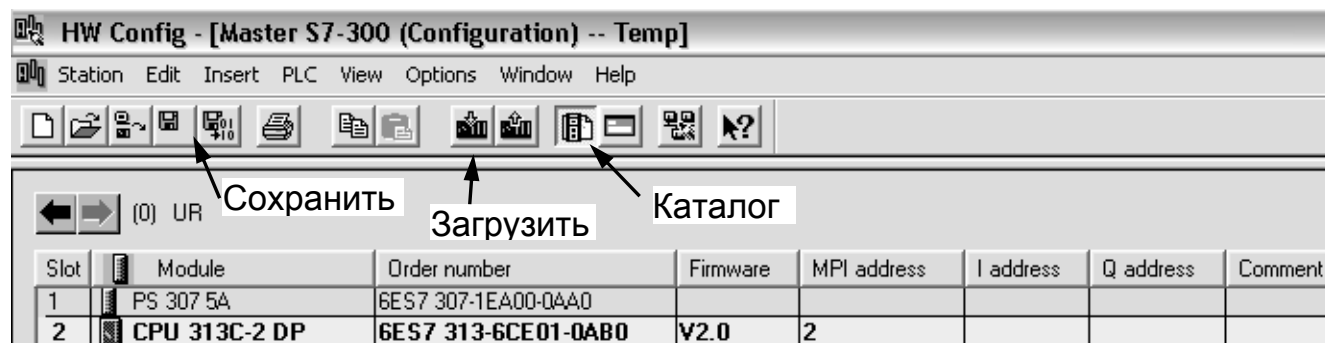


Рис. 2.3. Окно HW Config

STEP 7 проверяет, не нарушены ли правила для слотов, например CPU S7-300 может быть установлен только в слоте 2.

⊘ – Символ нарушения правил для слотов.

!!! В качестве альтернативы Вы можете также выделить соответствующую строку или несколько строк в конфигурационной таблице и дважды щелкнуть в окне каталога аппаратуры «Hardware Catalog» на желаемом модуле. Если выделено несколько строк, то все выделенные строки будут оснащены этим модулем сразу.

!!! Если выделить слот в стойке модулей, можно увидеть список всех возможных для установки модулей, открыв контекстно-зависимое меню (правой кнопкой мыши) «Insert Object [Вставить объект]» или «Replace Object [Заменить объект]». Эта возможность избавляет Вас от необходимости поиска нужного модуля в каталоге аппаратуры. Вы выбираете из всех модулей, которые имеются в текущем открытом профиле каталога.

!!! Помните, что модули в слотах следует располагать в той же последовательности, в какой они расположены физически и с теми номерами, которые указаны на них.

При установке CPU Master'a следует выбрать подключение к сети «Profibus (1)». Для этого левой кнопкой мыши выбрать строку DP процессорного модуля и в появившемся окне «Properties – PROFIBUS interface DP (RO/S2.1)» выделить строку «Profibus (1)» и нажмите ОК. В верхнем окне появится изображение шины (в просторечии «вешалка» или «якорь»), к которой будут подключаться компоненты сети.

При установке в конфигурационную таблицу модулей ввода/вывода STEP7 автоматически присваивает им адреса. Эти адреса необходимо учитывать в подготовленных дома программах.

Затем устанавливается на шину Profibus (1) Slave ET200M. Для этого в разделе PROFIBUS-DP каталога аппаратуры найдите раздел ET200M и из него методом Drag&Drop перетащите на изображение шины сам коммуникационный процессор

IM 153-2FD (заказной номер 153-2BB00-0XB0) и затем установить на него DI/DO-300 “SM323” (блок ввода-вывода дискретных сигналов).

Для проверки правильности проведенного конфигурирования необходимо в меню «Station [Станция]» выбрать и активизировать позицию «Consistency Check [Проверить согласованность]». Появляется соответствующее сообщение об отсутствии или наличии ошибок, и каких.

Покидание окна конфигурирования производится после сохранения данной станции «Station – Save» или сохранение с компиляцией «Station – Save and Compile».

«Slave S7-300» реализуется на программируемом контроллере SIMATIC S7-300 и поэтому является интеллектуальным Slave'ом (I-Slave'ом). Перед подключением его к системе DP-Master он должен быть вначале создан внутри проекта. Особенность его конфигурации заключается в том, что в нем должна быть предусмотрена промежуточная память для обмена информацией с Master'ом (рис. 2.4). CPU Slave'a берет на себя обмен данными с Master-устройством DP.

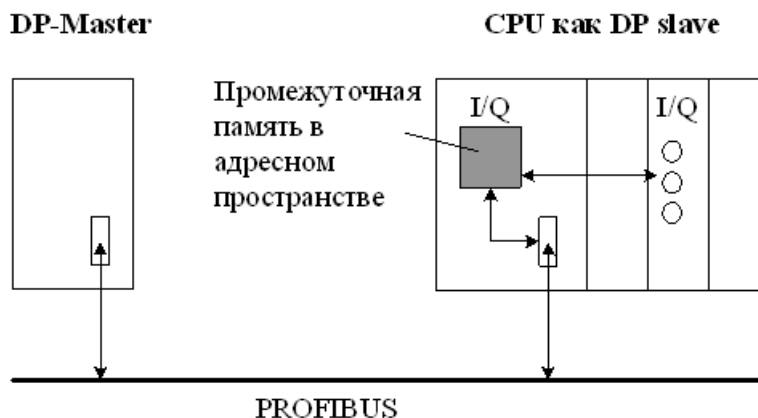


Рис. 2.4. Промежуточная память в CPU 315-DP, используемая как I-Slave

В промежуточной памяти проектируются адресные области входов (I) и выходов (Q). STEP7 для сетей PROFIBUS предусматривает возможность:

- запроктировать до 32 адресных областей входов и выходов;
- каждая из этих адресных областей может иметь величину (длину) до 32 байт;
- всего можно запроктировать максимум 244 байта входов и 244 байта выходов.

Следующая таблица (табл. 2.1) показывает принцип проектирования адресных областей. Для простоты для I и Q приняты одинаковые адреса.

Таблица 2.1

	Адрес Master-устройства	Адрес Slave-устройства	Длина	Единица
1	I100	Q100	2	Байт
2	Q100	I100	2	Байт

Указанные в табл. 2.1 адресные области CPU как Master-устройства DP и CPU как I-Slave-устройства DP задаются проектировщиком. Нельзя использовать адреса I и Q, используемые физическими каналами модулей ввода и вывода. Следует помнить, что в принципе для CPU315-2DP адресная область входов/выходов составляет 2048 байтов/2048 байтов.

Конфигурирование S7-300/CPU 315-2DP как Slave производится идентично изложенному ранее. Различие будет в указании работы DP интерфейса, то есть перевести из «DP-Master'a» (стоит по умолчанию) в «DP-Slave». Для этого выделите «DP» у CPU315-2DP. Откроется окно «Properties-DP Master». Откройте закладку «Operating Mode [Режим работы]» и установите режим работы «DP-Slave». После этого название «DP-Master» изменится на «DP-Slave». Далее перейдите на закладку «Configuration [Конфигурация]». Нажмите на кнопку New и в окне «Properties-DP – (Ro/S2.1) – Configuration – Row1» установите требуемый адрес промежуточной памяти для принимаемой информации, например, Input 100, Length: 2, Unit: Byte. После нажатия на кнопку ОК возвращаетесь в предыдущее окно, вновь нажимаете кнопку New и теперь устанавливаете адрес промежуточной памяти для информации выдаваемой, например, Output 100, Length: 2, Unit: Byte.

После нажатия на кнопку ОК возвращаетесь в окно HW Config.

Перед выходом из «HW Config» вновь используйте команду меню «Station», далее «Check Consistency», чтобы убедиться, что в конфигурации станции нет ошибок. Затем через команды «Save» или «Save and Compile» возвращаетесь в окно проекта SIMATIC Manager.

Теперь созданный «Slave S7-300» необходимо перенести на сеть «Profibus (1)». Для этого нужно перейти в окно «HW Config» для станции «Master' a S7-300». Откройте в каталоге оборудования папку «PROFIBUS-DP», а в ней подкаталог «Configured Stations», и подключите объект «CPU31x-2DP» к шине («вешалке») DP-Master'a методом Drag&Drop. При этом появляется окно «DP slave properties» с открытой закладкой «Connection», где показан перечень соответствующих Slave'ов. Выделите нужный и нажмите кнопку «Connect». Выбранный Slave исчезает из списка. Перейдите на закладку «Configuration» и внесите туда через функцию Edit адреса и длину промежуточной памяти для Master' a как показано на рис. 2.5.

Row	Mode	Partner DP a...	Partner addr	Local addr	Length	Consiste ...
1	MS	2	Q100	I100	2 Byte	Unit
2	MS	2	I100	Q100	2 Byte	Unit

Рис. 2.5. Информация на закладке «Configuration», «DP slave properties», «HW-Config»

Приведенные здесь адреса свободно выбраны для проекта-примера. Вы можете задавать другие области и адреса. Нужно только обратить внимание на то,

чтобы выходная область DP-Master' а всегда соответствовала входной области DP-Slave и наоборот.

Вернитесь в главное окно «HW Config [Master (Configuration)]» с помощью кнопки ОК. Slave в этом окне получит PROFIBUS-адрес. Свой адрес имеет каждый компонент подключенный к сети. Этот адрес присваивается STEP7 компонентам автоматически, но проектировщик его может изменить по своему усмотрению.

Компоненты распределенной периферии на лицевой панели имеют наборное поле для установки его адреса в сети. Необходимо следить, чтобы установленный адрес соответствовал адресу компонента в конфигурационной таблице.

Сохраняем проекты конфигурирования «Master S7-300» и «Slave S7-300» с компиляцией. На этом процесс конфигурирования закончен.

## 2.5. Программирование контроллеров

Для проверки работоспособности сконфигурированной сети необходимо загрузить в контроллеры проверочную программу. Как осуществить набор программы рассматривается в работе 1 «Программирование контроллера Simatic S7-300» этого пособия.

В лабораторной установке модули контроллеров Master'a и Slave'a выполнены на одинаковых CPU и отличаются только модулями ввода/вывода. В «Блоке имитационных сигналов» для них предусмотрены идентичные кнопки и тумблеры для имитации входных сигналов и команд и светодиоды для индикации выходных сигналов и команд. При конфигурировании контроллеров у обоих из них будут присутствовать байт входов IB0 и байт выходов QB4.

Ниже приводится два варианта программ, которые предусматривают при подаче сигналов на модуль ввода с адресом IB0 Master'a загораются индикаторы выходных сигналов адреса QB4 Slave'a и наоборот.

На рис. 2.6 представлены пользовательские программы в организационных блоках OB1 для Master'a (рис. 2.6а) и для Slave'a (рис. 2.6б) с использованием команды MOVE.

При передаче между Master'ом и Slave'ом больше четырех последовательных байт данных (консистентных данных) используются функции SFC14 и SFC15 (SFC – System Function Call).

Функция SFC14 применяется для чтения последовательных принимаемых данных (приемник) (рис. 2.7).

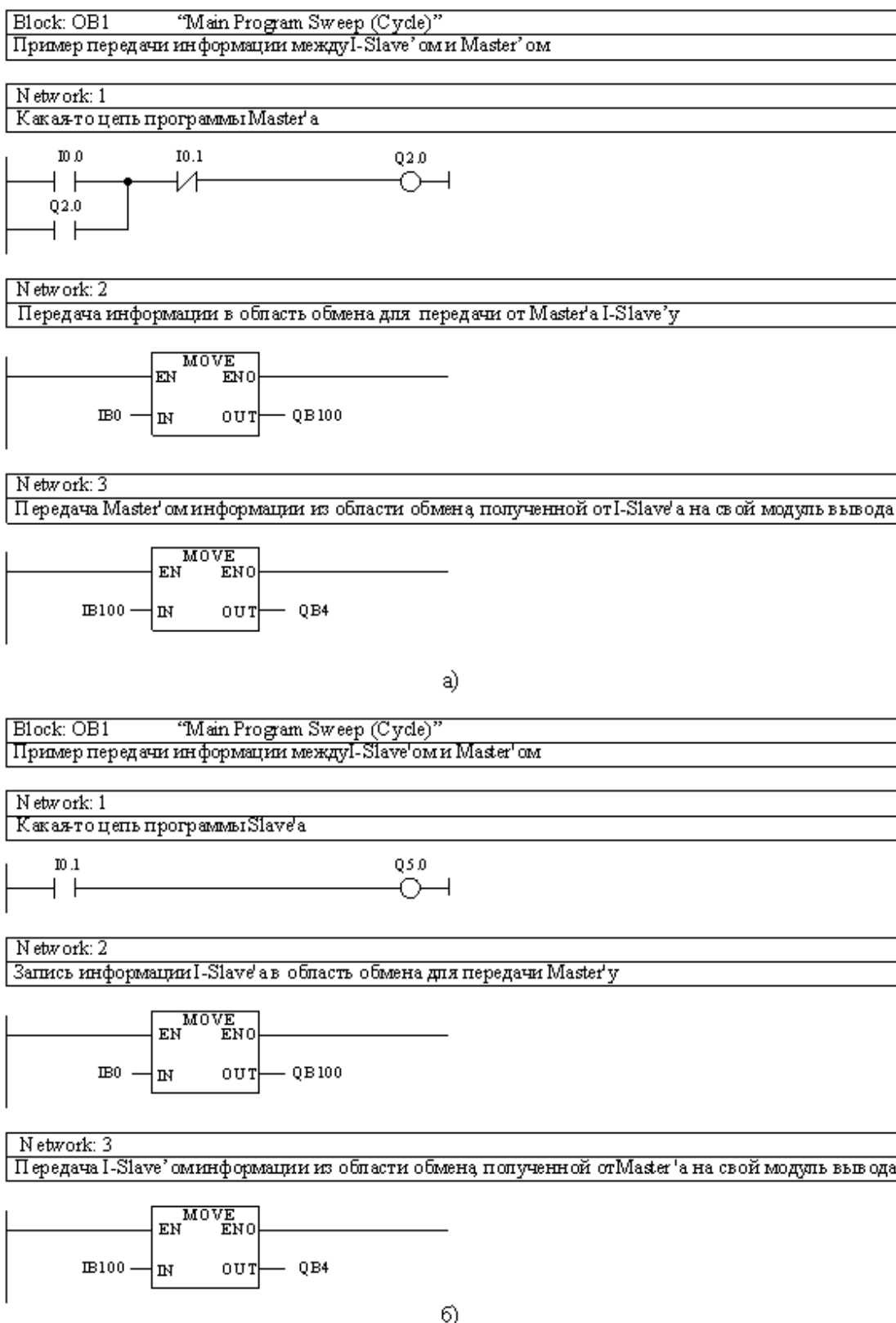


Рис. 2.6. Пример передачи информации между контроллерами с использованием команд MOVE: а) – программа в OB1 Master'a, б) – программа в OB1 I-Slave'a

Параметр LADDER (рис. 2.7) в шестнадцатеричном формате указывает начальный адрес области (промежуточной памяти) в которую поступают принимаемые данные, например w#16#64.

Параметр RECORD (рис. 2.7) указывает область, в которую направляется полученная информация, например P#M40.0 BYTE8.

Параметр RET\_VAL (рис. 2.7) указывает адрес памяти, в котором будет код ошибки при сбое передачи информации, например MW60.

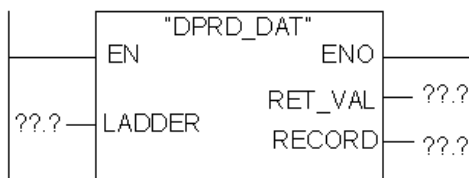


Рис. 2.7. SFC14 на языке лестничных диаграмм

Функция SFC15 применяется при передаче данных (передатчик) (рис. 2.8).

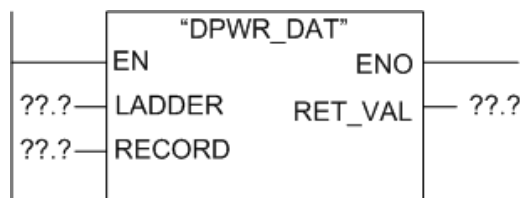


Рис. 2.8. SFC15 на языке лестничных диаграмм

Параметр LADDER (см. рис. 2.8) в шестнадцатеричном формате указывает начальный адрес области (промежуточной памяти) куда записать данные для отправки, например, w#16#64.

Параметр RECORD (см. рис. 2.8) указывает, откуда взять информацию, которую нужно передать, например P#M80.0 BYTE8.

Параметр RET\_VAL (см. рис. 2.8) указывает адрес памяти, в котором будет записан код ошибки при сбое передачи информации, например MW70.

На рис. 2.9 схематично представлен обмен данными между Master'ом и Slave'ом с использованием функций SFC14 и SFC15.

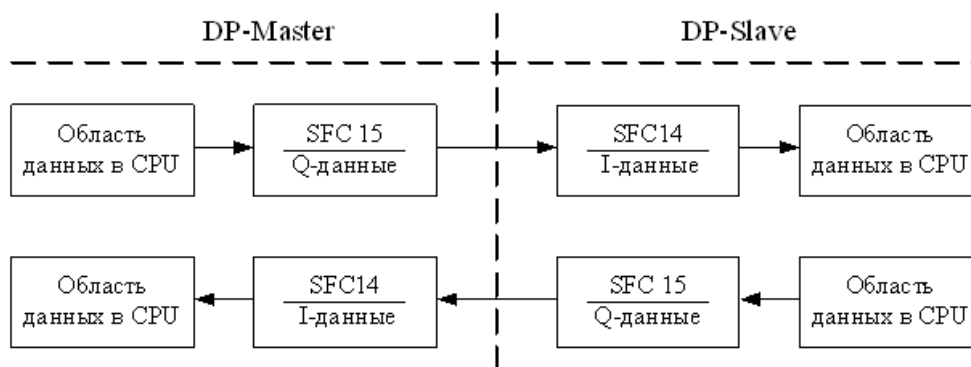


Рис. 2.9. Обмен входными/выходными данными через функции SFC14 и SFC15

На рис. 2.10 представлены программы для контроллеров те же самые, что и на рис. 2.6, но с использованием функций SFC14 и SFC15.

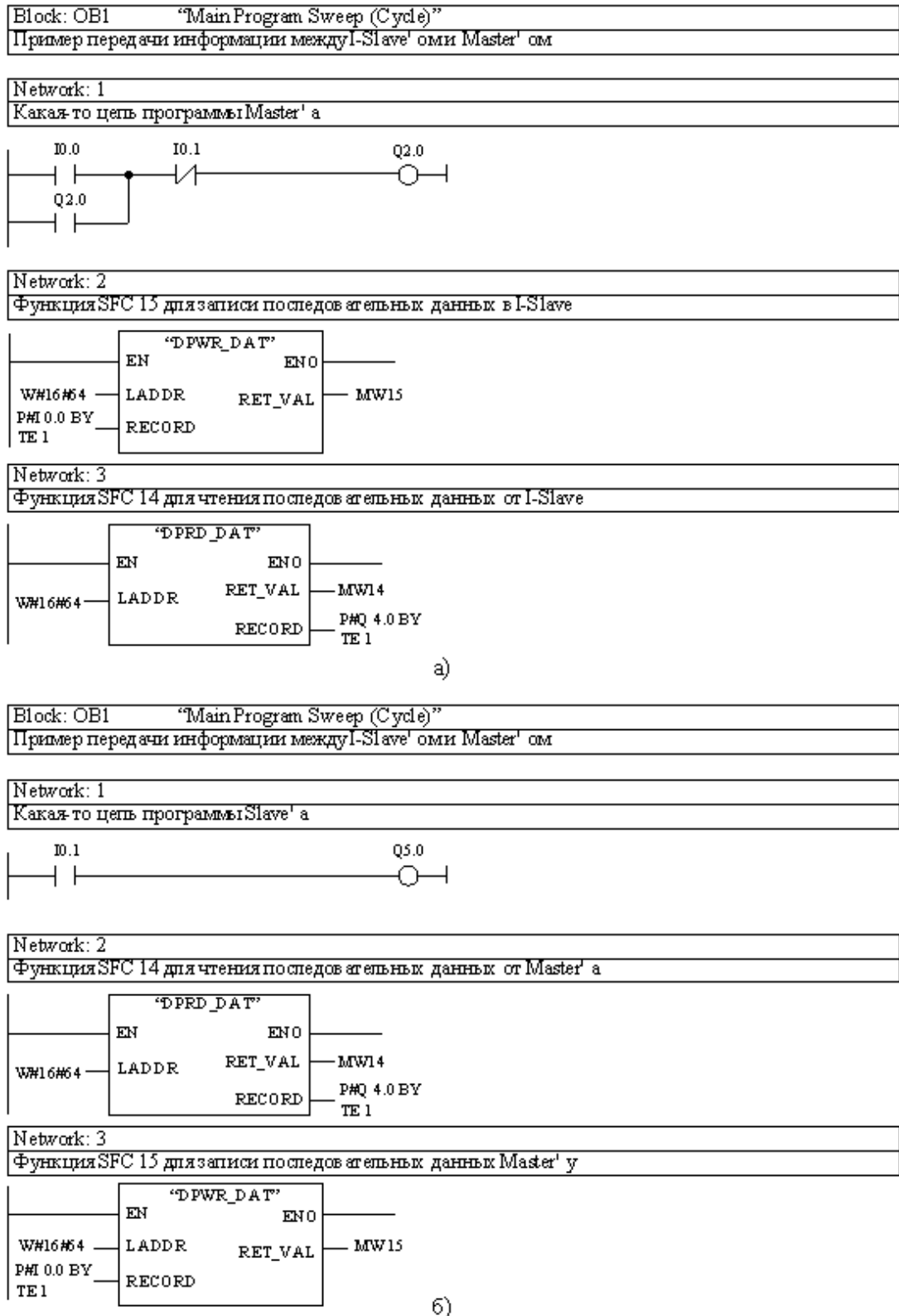


Рис. 2.10. Пример передачи информации между контроллерами с использованием функций SFC14 и SFC15:

а) – программа в OB1 Master'a, б) – программа в OB1 I-Slave'a



Программы демонстрируют, что функциями SFC14 и SFC15 можно передавать информацию с числом байт меньше четырех.

Slave'ами распределенной периферии, например, типа ET200M Master распоряжается как «собственными» модулями ввода/вывода, используя ту адресацию, которая определена при конфигурации сети. При этом не обращается внимание на промежуточные усилительно-повторительные элементы, проводную или оптоволоконную среду передачи. Если же необходимо использовать информацию с модулей распределенной периферии I-Slave'ом на программируемом контроллере, в нашем случае на SIMATIC S7-300/CPU 315-2DP, то необходимо ее получать или передавать через Master'a.

При вводе в сеть I-Slave'a на программируемом контроллере всегда как в программе Master'a, так и в программе Slave'a предусматривайте введение организационных блоков OB82 (диагностические сигналы) и OB86 (выход из строя носителя модулей). **Без этих блоков обмен между Master'ом и Slave'ом невозможен.**

## 2.6. Загрузка программ в программируемые контроллеры и включение в работу

Перед загрузкой программы в любой контроллер необходимо произвести стирание памяти его CPU. Для этого необходимо осуществить осмысленную последовательность операций с его переключателем режимов работы на лицевой панели CPU:

- перевести переключатель в положение STOP;
- перевести переключатель в положение MRES. Удерживайте переключатель в этом положении, пока светодиод STOP не загорится во второй раз. После этого отпустите переключатель. Он вернется в положение STOP;
- в течение 3 с необходимо снова перевести переключатель в положение MRES и держать так до тех пор, пока светодиод STOP мигает быстро (с частотой 2 Гц). Теперь переключатель можно отпустить. Когда CPU совершит общее стирание, светодиод STOP перестает мигать и горит ровным светом.

Указанные операции с переключателем режимов необходимы только тогда, когда пользователь сам хочет выполнить общее стирание CPU без запроса на эту операцию со стороны CPU (медленное мигание светодиода STOP). Если CPU сам запрашивает общее стирание, то достаточно кратковременно переключатель режимов работы перевести в положение MRES, чтобы запустить процесс общего стирания CPU. Если CPU после его успешного стирания вновь требует общего стирания, то в определенных случаях необходимо форматирование его сменной платы микропамяти SIMATIC (MMC).

Для форматирования платы MMC необходимо:

- переключатель режимов работы перевести в положение MRES и удерживать его в этом положении (около 9 с), пока светодиод STOP не загорится ровным светом;

– в течение следующих 3 с необходимо отпустить переключатель и вновь перевести его в положение MRES. Теперь светодиод STOP мигает только во время форматирования.

Всегда выполняйте эту последовательность операций в течение указанного времени, так как в противном случае MMC не форматировается, а возвращается в состояние общего стирания.

В сети у каждого контроллера будь то Master или Slave есть своя пользовательская программа.

В лабораторной установке используется контроллер Master и один I-Slave. Поэтому в левой части окна SIMATIC Manager, которая содержит структуру проекта, имеется два проекта «DP-Master» и «DP-Slave». Они должны загружаться каждый в свой контроллер.

Запись программ в контроллеры, как уже было указано выше, осуществляется по сети.

Для записи необходимо, чтобы:

– компьютер через адаптер был подключен к разъему DP CPU «DP-Master'a»;

– все компоненты сети должны иметь разные сетевые адреса и действительная конфигурация должна соответствовать реальной структуре;

– конфигурация может быть загружена только при согласованности и отсутствии ошибок.

Последовательность записи следующая:

– в меню SIMATIC Manager выбрать функцию «Option», затем «Set PG/PC Interface», а в ней режим «PC Adapter (Auto)»;

– переместитесь через «Master S7-300», CPU 315-2DP, S7 Program (1) и щелкните на символе «Blocks [Блоки]». Ключ режима контроллера поставьте в положение STOP;

– из меню «PLC [ПЛК]» выберите команду «Download [Загрузить]» для передачи программы и конфигурации аппаратуры в CPU. Щелкните на Yes во всех появляющихся окнах. Запись конфигурации и пользовательской программы Master'a записана в соответствующий контроллер;

– аналогично следует поступить с проектом «Slave S7-300», то есть через «Slave S7-300», CPU 315-2DP, S7 Program (2) и так далее.

Для запуска сети необходимо перевести переключатели режимов работы контроллеров в положение «Run».

Во время работы программы можно на экране монитора компьютера следить за ходом ее выполнения. Для этого при нахождении переключателей режимов работы контроллеров в положениях «Run» следует нажать на панели инструментов монитора кнопку с изображением очков «Monitor (on/off) [Монитор (включение/выключение)]». При этом изменением цвета будут отображаться состояния контактов и катушек, а также значения счетчиков и таймеров в реальном времени.

## 2.7. Выполнение лабораторной работы

Схема лабораторной установки перед началом работы полностью собрана и никаких изменений в процессе работы не предусматривается. Программы предыдущих бригад студентов удалены из компьютера.

В начале работы необходимо:

- ознакомиться с составом лабораторной установки, конструктивным исполнением компонентов сети и соединениями между ними. Обратить особое внимание на подключение кабелей проводных (витая пара) RS485 и оптического;
- включить компьютер, запустить программу «SIMATIC Manager [Администратор SIMATIC]»;
- провести конфигурирование сети, убеждаясь в ее правильности, используя команды меню «Station > Consistency Check», перед выходом из окна «HW Config» с использованием команд меню «Save and Compile [Сохранить и компилировать]»;
- в организационные блоки OB1 Master'a S7-300 и Slave'a S7-300 ввести соответствующие программы обмена, представленные на рис. 2.6 или рис. 2.10 (по выбору студентов) для обмена информацией между указанными компонентами сети. Не забыть вставить в программы организационные блоки **OB82 и OB86**;
- включить блок питания сети и убедиться в наличии питания всех компонентов сети;
- осуществить стирание памяти CPU контроллеров;
- записать подготовленные дома программы с конфигурацией в соответствующие контроллеры Master'a S7-300 и Slave'a S7-300;
- проверить правильность функционирования сети: при включении/выключении входов с адресом байта I0 контроллера Master'a S7-300 соответственно загораются/гаснут выходы байта Q4 Slave'a S7-300 и наоборот;
- если проверка прошла успешно, то можно приступить к выполнению индивидуальных заданий.

Вариант задания выбирается в соответствии с табл. 2.2.

Таблица 2.2

Номер бригады	1 и 5			2 и 6			3 и 7			4 и 8		
Член бригады	1	2	3	1	2	3	1	2	3	1	2	3
Вариант задания	1	2	3	4	5	6	7	8	9	10	11	12

В каждом варианте предусматривается ручное и автоматическое управление объектом. Объектом управляет контроллер DP-Slave' а 1, команды задания выбора режима работы (ручной/автоматический) и пуска/остановка автоматического режима работы подаются с контроллера DP-Master' а, а команды ручного управления – со Slave'a ET200M.

Ниже рассмотрен пример программирования управления тележкой, перемещающейся по прямолинейному пути от положения П1 к положению П2 и обратно (рис. 2.11).

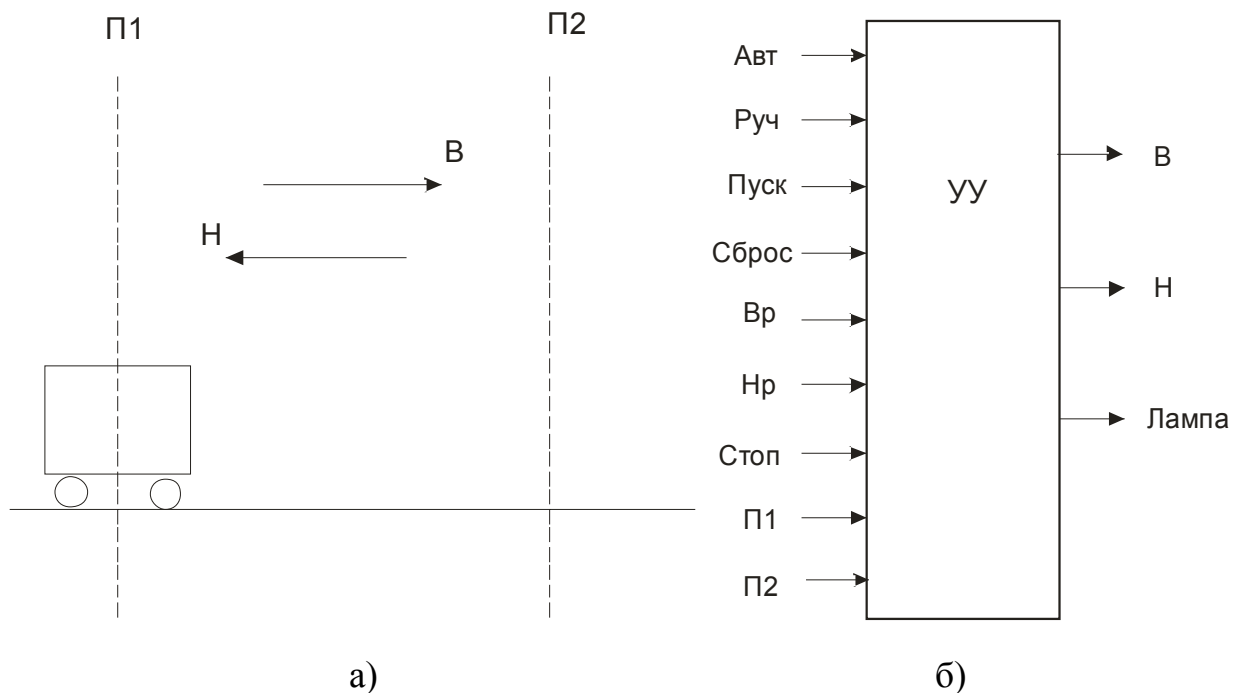


Рис. 2.11. К условию задачи: а) конструктивная схема механизма; б) входные и выходные сигналы/команды устройства управления УУ

Исходное положение тележки П1. В автоматическом режиме работы «Авт» при нажатии кнопки «Пуск» возникает команда на движение тележки вперед «В». Тележка доходит до положения П2, команда «В» отключается. Возникает команда на движение назад «Н». Тележка доходит до положения П1, команда «Н» отключается. Возникает команда «В», тележка движется до П2. Включается команда «Н» и т.д. Тележка непрерывно перемещается от положения П1 к П2 и обратно. При работе в автоматическом режиме включена световая индикация «Лампа».

При нажатии кнопки «Сброс», независимо от того где находилась тележка, она немедленно должна вернуться в положение П1. «Лампа» гаснет.

В ручном режиме работы «Руч» команды «В» и «Н» формируются при нажатии кнопок ручного управления соответственно «Вр» и «Нр». Останов тележки в ручном режиме – при нажатии кнопки «Стоп».

Для управления тележкой вводится память о включении автоматического режима « $p_a$ ». Она возникает в положении П1 при нажатии кнопки «Пуск» и сохраняется до появления сигнала памяти « $p_c$ » о нажатии кнопки «Сброс». Этому соответствует логическое уравнение:

$$p_a = Авт \cdot (П1 \cdot Пуск + p_a) \cdot \overline{p_c}$$

Память « $p_c$ » возникает в автоматическом режиме при любом положении тележки при нажатии кнопки «Сброс» и сохраняется до нажатия кнопки «Пуск»:

$$p_c = Авт \cdot (Сброс + p_c) \cdot \overline{Пуск}$$

Команда на движение вперед «В» возникает в автоматическом режиме в положении П1 и сохраняется до достижения тележкой положения П2. В ручном режиме она возникает в любом положении тележки при нажатии кнопки «Вр» и сохраняется до нажатия кнопки «Стоп» или до достижения положения П2. Этому соответствует логическое уравнение:

$$B = \left[ p_a \cdot (П1 + B) + p_a \cdot \overline{H} + Pуч \cdot (Вр + B) \cdot \overline{Стон} \right] \cdot \overline{П2}$$

Команда на движение тележки назад «Н» возникает в автоматическом режиме в положении П2 и сохраняется до достижения тележкой положения П1. Она возникает также в любом положении тележки при наличии сигнала памяти сброса и отсутствии движения вперед. В ручном режиме команда «Н» возникает в любом положении тележки при нажатии кнопки «Нр» и сохраняется до нажатия кнопки «Стоп» или до достижения положения П1. Этому соответствует логическое уравнение:

$$H = \left[ p_a \cdot (П2 + H) + p_a \cdot \overline{B} + Pуч \cdot (Нр + H) \cdot \overline{Стон} \right] \cdot \overline{П1}$$

Для исключения одновременного появления команд «В» и «Н» в уравнения этих команд вводится взаимная блокировка. Тогда:

$$B = \left[ p_a \cdot (П1 + B) + Pуч \cdot (Вр + B) \cdot \overline{Стон} \right] \cdot \overline{П2} \cdot \overline{H}$$

$$H = \left[ p_a \cdot (П2 + H) + p_a \cdot \overline{B} + Pуч \cdot (Нр + H) \cdot \overline{Стон} \right] \cdot \overline{П1} \cdot \overline{B}$$

«Лампа» включается сигналом « $p_a$ », т.е. по логическому уравнению:

$$Лампа = p_a$$

По условию задачи команды выбора режима «Авт», «Руч» и пуска/останова автоматического режима («Пуск», «Сброс») подаются с контроллера Master S7-300, а команды ручного управления «Вр», «Нр» и «Стоп» – со Slave'a ET200M. Механизмом управляет Slave S7-300 (рис. 2.12).

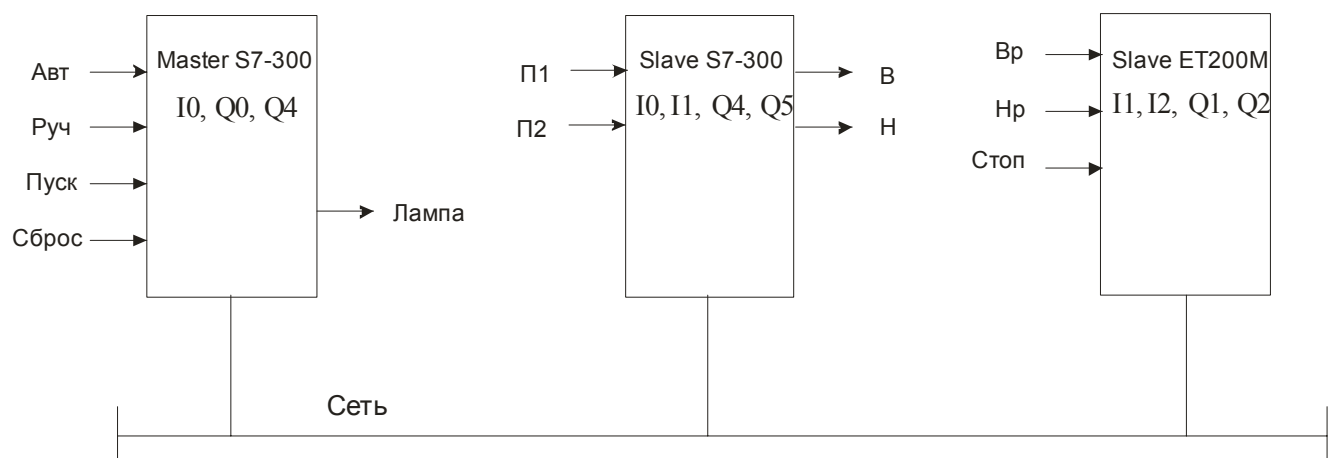


Рис. 2.12. Подключение входных/выходных сигналов к компонентам сети

После конфигурирования рассматриваемой сети автоматически были присвоены адреса байтов входов I и выходов Q модулям компонентов сети. Эти адреса указаны внутри корпусов компонентов сети на рис. 2.12. Обратите внимание, что входами/выходами децентрализованной периферии Slave'a ET200M контроллер Master S7-300 распоряжается как с входами/выходами своей стойки, но расположенными на удалении от этой стойки (через кабель с витой парой, оптический шинный терминал, оптоволоконный кабель и IM153-2 станции ET200M).

При программировании пользовательской программы следует всем входам и выходам рассматриваемого примера присвоить физические адреса. В примере принято, что у Master'a все входы подключены к байту I0, а выход к байту Q4. У Slave'a ET200M все выходы ручного управления подключены к байту I1. У Slave'a S7-300 сигналы с датчиков положения П1, П2 подключены к байту I0, а выходными команды В = Н – к байту Q4 выходных сигналов.

Воспользуемся возможностью символьного представления адресов контроллеров Simatic. Символы сигналам и командам присвоим те, которые использованы в логических уравнениях. В табл. 2.3 представлена адресация сигналов и команд Master'a S7-300.

Таблица 2.3

Адресация переменных Master'a S7-300

Символ сигнала, команды	Авт	Руч	Пуск	Сброс	Вр	Нр	Стоп	Лампа
Адрес физический	I0.0	I0.1	I0.2	I0.3	I1.0	I1.1	I1.2	Q4.0

Так как объектом управляет Slave S7-300, а обеспечивает Master S7-300, то в Slave S7-300 выделяется область памяти для обмена информацией с Master'ом S7-300. Воспользуемся адресной областью, рассмотренной в табл. 2.1, в которой для Master'a и Slave'a присвоено по два бита с адресов IB100 и QB100.

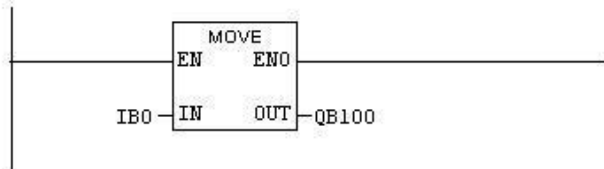
Тогда программа для Master'a S7-300 принимает вид, представленный на рис. 2.13.

Команды управления, поступающие с Master'a S7-300, будут размещаться в байтах соответственно IB100 и IB101 Slave'a S7-300. Выдаваемый со Slave'a S7-300 сигнал «Лампа» будет передаваться в составе байта QB100 Slave'a S7-300 на вход IB100 Master'a S7-300.

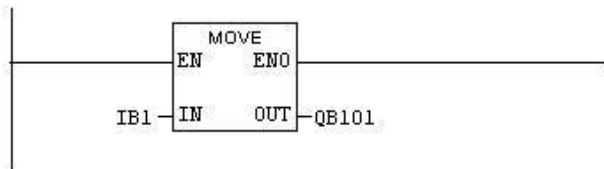
С учетом этого у Slave'a S7-300 адресация сигналов и команд принимается следующая (табл. 2.4).

OB1 : "Main Program Sweep (Cycle)"  
 Программа Master'a S7-300

**Network 1 : Title:**  
 Передача информации со входов Master'a в область для Slave'a



**Network 2 : Title:**  
 Передача информации со входов ET200M в область обмена для Slave'a



**Network 3 : Title:**  
 Прием информации из области обмена от Slave'a для включения сигнала "Лампа"

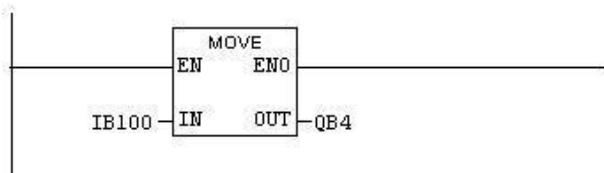


Рис. 2.13. Программа для Master'a S7-300

Таблица 2.4

Адресация переменных Slave'a S7-300

Символ сигнала, команды	Авт	Руч	Пуск	Сброс	Вр	Нр	Стоп
Адрес физический	M0.0	M0.1	M0.2	M0.3	M1.0	M1.1	M1.2

Символ сигнала, команды	Лампа	П1	П2	В	Н	р <sub>а</sub>	р <sub>с</sub>
Адрес физический	M2.0	I0.0	I0.1	Q4.0	Q4.1	M3.0	M3.1

Из табл. 2.4 видно, что в программе Slave'a S7-300 принимаемые от Master'a S7-300 команды должны заноситься в байты меркеров соответственно MB0 и MB1, а сигнал на включение лампы записывается в байт меркеров MB2 для передачи Master'у.

На рис. 2.14 представлена программа для Slave'a S7-300 на языке лестничных диаграмм.

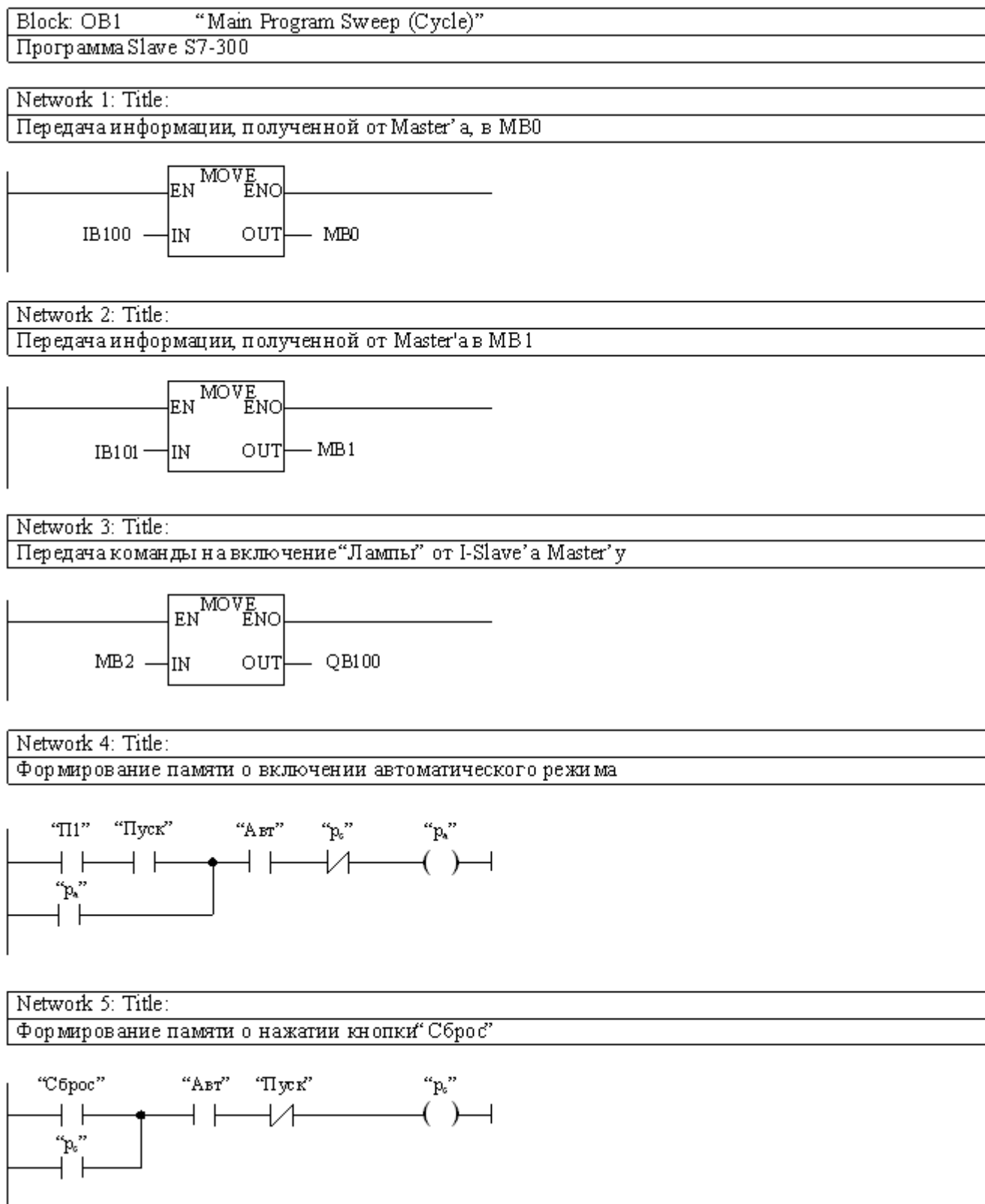


Рис. 2.14. Программа для Slave'a S7-300



*Вариант 1.* Автоматизировать управление дверью типа двери кабины лифта. Нормальное положение двери закрытое. Есть кнопка «Открыть», датчики открытого и закрытого положения двери соответственно До и Дз, а также датчик наличия препятствия Пр закрывания двери.

При нажатии кнопки «Открыть» происходит открытие двери. Через 5 с формируется команда на закрытие двери. Если при закрытии двери появился сигнал препятствия, закрывание двери отключается и дверь открывается. Через 5 с вновь формируется команда на закрывание двери. Если вновь появился сигнал препятствия, то дверь снова открывается на 5 с и так далее. После 3 таких закрываний/открываний, вызванных препятствием, дверь открывается и остается открытой, а на диспетчерском пункте возникает сигнал «Препятствие».

Для устранения неисправностей предусмотрена кнопка «Сброс» и кнопки ручного управления «Открыть» и «Заккрыть».

*Вариант 2.* Реализовать управление формированием выходной команды X, зависящей от входных сигналов A, B и C.

В автоматическом режиме при комбинации A=1 и B=1 выходной сигнал принимает значение 1. После исчезновения сигнала A или B выходной сигнал X принимает значение 0. При последующих появлениях комбинации A=1 и B=1 сигнал X не должен принимать значение 1 до тех пор, пока хотя бы кратковременно не появится сигнал C. То есть сигнал C возвращает схему в исходное состояние.

При ручном управлении любая комбинация A=1 и B=1 вызывает появление команды X.

*Вариант 3.* Реализовать управление светофором. Горение ламп в следующей последовательности: Красный, желтый, зеленый, желтый, красный, желтый и так далее. Время горения каждой лампы условно 5 с.

В ручном режиме работы (настроечном) каждая лампа управляется своим тумблером или кнопкой и можно включать их в любой комбинации.

*Вариант 4.* Автоматизировать процесс передачи детали с рольганга (роликового транспортера) 1 на транспортер 2 (рис. 2.15).

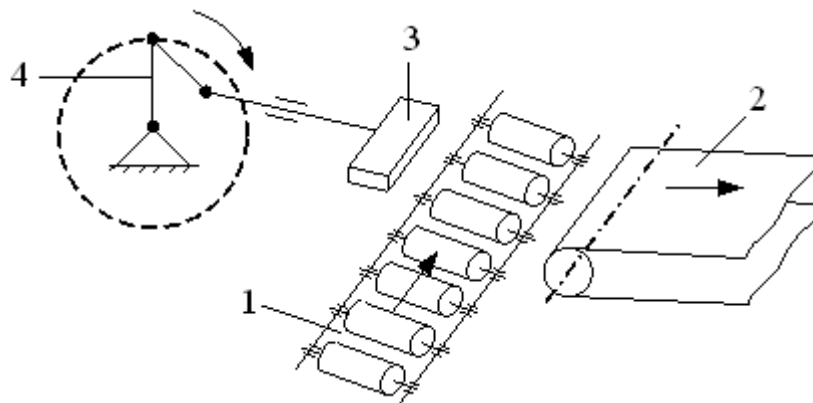


Рис. 2.15. Конструктивная схема автоматизируемого объекта

Рольганг 1 движется до появления детали перед толкателем 3 с кривошипно-шатунным механизмом 4. Рольганг останавливается, включается привод кривошипа толкателя, и толкатель сталкивает деталь на транспортер 2, который затем включается на 5 с, а кривошип толкателя, сделав полный оборот, возвращается в исходное положение. Через 2 с после прихода толкателя в исходное положение вновь включается рольганг 1.

При ручном управлении механизмы управляются при нажатии на соответствующие кнопки включено/отключено.

*Вариант 5.* Автоматизировать процесс распределения заготовок.

Машина М1 (рис. 2.16) производит изделия, которые необходимо по 5 штук поочередно передавать то к рабочему месту 1, то к рабочему месту 2. Подача изделий осуществляется рольгангами (роликовыми транспортерами) 1, 2 и 3.

Имеется направляющая планка 4, которая может занимать только 2 положения для направления заготовок к соответствующему рабочему месту. Положение планки зависит от числа переданных изделий. После передачи 5 изделий к одному рабочему месту она перебрасывается для подачи изделий к другому рабочему месту и наоборот.

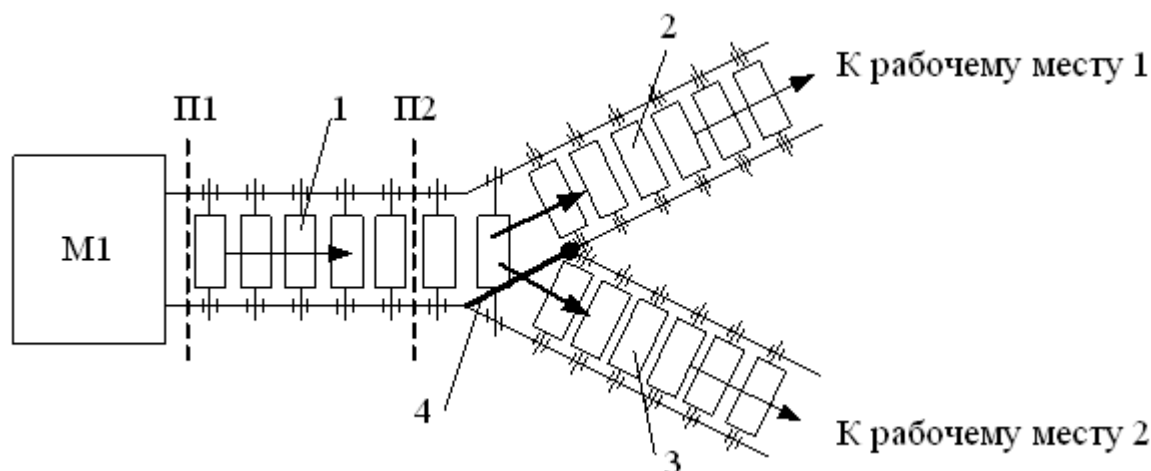


Рис. 2.16. Конструктивная схема распределения изделий

При подаче питания планка 4 должна занимать положение для направления изделий к рабочему месту 1. При выходе изделий из машины М1 по сигналу о положении П1 включается рольганг 1. При достижении положения П2 включается на 10 с, в зависимости от положения планки, рольганг 2 или рольганг 3. Рольганг 1 отключается через 3 с после достижения изделием положения П2. Исключить возможность удара изделия направляющей планкой при ее перебрасывании.

При ручном управлении механизмы управляются соответствующими кнопками.

*Вариант 6.* Автоматизировать процесс сортировки деталей.

Машина М1 (см. рис. 2.16) выдает на рольганг 1 попеременно то низкие, то высокие детали. Высокие детали необходимо отправлять рольгангом 2 к рабочему месту 1, низкие – рольгангом 3 к рабочему месту 2. Для этого имеется направляющая планка 4, которая может занимать только два положения для направления деталей к соответствующему рабочему месту.

При выдаче детали из машины М1 по сигналу положения П1 включается рольганг 1. При движении детали от положения П1 к положению П2 организовать контроль высоты детали и организовать переброс планки 4 в соответствующее положение в зависимости от высоты детали. В положении П2 включается соответствующий, в зависимости от высоты детали, рольганг 2 или 3. Рольганг 2 включается на 5с, рольганг 3 – соответственно на 7с. Рольганг 1 отключается через 3с после достижения деталью положения П2.

Организовать счет высоких и низких деталей. Через каждые 3 штуки выданных высоких деталей включается на 3с световой сигнал и счет начинается снова. Аналогично формируется световой сигнал на 3с после прохождения 3 штук низких деталей.

При ручном управлении механизмы управляются соответствующими кнопками включено/отключено.

*Вариант 7.* Кривошипно-шатунный толкатель имеет одно контролируемое положение П0. В автоматическом режиме при кратковременном нажатии на кнопку «Пуск» привод кривошипа немедленно включается и из положения П0 движется «Вперед». Кривошип совершает полный круг, возвращается в положение П0 и останавливается. По истечении 3с привод вновь включается, кривошип совершает полный круг, возвращается в положение П0, вновь стоит 3с, включается и так далее. После 5 толканий кривошип останавливается в положении П0 и формируется сигнал «Конец цикла».

В ручном режиме работы предусмотреть возможность движения кривошипного толкателя «Вперед» и «Назад».

*Вариант 8.* Тележка перемещается по кольцевому пути. Исходное положение П0, промежуточное положение П1. При кратковременном нажатии на кнопку «Пуск цикла» тележка немедленно движется «Вперед». В положении П1 останавливается на 3с, вновь включается и движется в положение П0. Стоит в П0 3с, включается «Назад», в положении П1 стоит 3с и затем продолжает двигаться «Назад» до положения П0. В П0 формируется сигнал «Конец цикла».

В ручном режиме управления предусмотреть движение тележки «Вперед» и «Назад».

*Вариант 9.* Тележка перемещается по кольцевому пути. Имеются 3 фиксированных положения П0...П2. При кратковременном нажатии на кнопку «Пуск цикла» привод тележки немедленно включается «Вперед» к положению П1. В П1

стоит 3 с и продолжает движение «Вперед» до положения П2. В положении П2 тележка стоит 3 с и включается «Назад» до положения П1. В положении П1 тележка стоит 3 с, включается «Вперед», останавливается в положении П2 на 3 с и затем продолжает движение «Вперед» до положения П0, где формируется сигнал «Конец цикла».

В ручном режиме управления предусмотреть движение тележки «Вперед» и «Назад».

*Вариант 10.* У шлагбаума при въезде на территорию гаража имеется датчик контроля препятствия в створе шлагбаума (машина, человек). У оператора для управления в автоматическом режиме есть кнопка «Открыть», для управления в ручном режиме кнопки «Открыть» и «Закрыть».

Нормальное положение шлагбаума закрытое. В автоматическом режиме при появлении автомобиля оператор нажимает кнопку «Открыть». Шлагбаум поднимается в крайнее верхнее открытое положение. Через 5 с он включается на закрытие, если нет сигнала датчика препятствия. Если при закрывании шлагбаума появится сигнал препятствия, то шлагбаум вновь откроется и вновь сделает попытку на закрывание через 5 с при отсутствии сигнала препятствия. Если сделано 5 таких попыток на закрытие, то шлагбаум остается открытым и автоматический режим сбрасывается.

При ручном управлении команды на открывание/закрывание шлагбаума осуществляется оператором, на которого возлагается контроль наличия препятствия.

*Вариант 11.* У двери кабинета офисного здания 2 замка: один механический, второй электромагнитный. Механический замок закрывается ключом на ночь, а днем этот замок открыт. Днем работает электромагнитный замок. При включенном электромагните замок открыт. Для управления замком на двери со стороны коридора стоит тумблер (включен/отключен) и светодиод, с внутренней стороны – одна кнопка «Открыть».

Нормальное положение замка под действием пружин на ригель закрытое. Электромагнит отключен. Для открытия замка со стороны коридора используется осмысленная последовательность действий с тумблером аналогичная последовательности при сбросе памяти CPU контроллера SIMATIC S7-300. Тумблер следует поставить в верхнее (включенное) положение. Начинается мигание светодиода. Когда он включится во второй раз нужно немедленно тумблер выключить. Электромагнит включается, открывая замок на 10 с, после которых он выключается. Отключенный замок не препятствует захлопыванию двери. С внутренней стороны замок на 10 с открывается при кратковременном нажатии на кнопку «Открыть».

*Вариант 12.* Автоматизировать тактовый стол для подачи заготовок к станку. Стол круглый с центральной осью вращения по типу стола капитал-шоу «Поле Чудес». Стол разбит на секторы, на которые устанавливаются заготовки. Есть

датчик, который реагирует на воздействующие элементы, закрепленные на каждом секторе. Когда робот снял очередную заготовку для ее установки на станке, он вырабатывает кратковременный импульс. По этому импульсу включается привод поворота стола, и движение продолжается до воздействия на датчик элемента очередного сектора.

При ручном режиме работы при нажатии кнопки «Вращение» стол вращается, не реагируя на сигналы датчика. При отпускании этой кнопки стол должен остановиться по сигналу датчика.

## 2.8. Требования к отчёту

Отчет должен содержать:

- а) цель работы;
- б) описание технологической задачи по варианту, конфигурацию или технологическую схему автоматизированного объекта, обозначение переменных, содержательное описание формирования логических функций, логические функции, соответствующие работе объекта с учетом ручного и автоматического режимов работы;
- в) физические и символические адреса переменных, отредактированные и проверенные программы для программируемых контроллеров Master S7-300 и Slave S7-300 с комментариями;
- г) описание методики и экспериментальной проверки работоспособности системы автоматизации в автоматическом и ручном режимах;
- д) выводы по работе.

## 2.9. Контрольные вопросы

1. Что означает термин PROFIBUS?
2. Какой «Slave» называется интеллектуальным слэйвом «I-Slave»?
3. В какой последовательности осуществляется конфигурирование сети PROFIBUS?
4. Как осуществляется проверка правильности проведенной конфигурации сети PROFIBUS?
5. Как осуществляется запись программ в контроллеры Master' а и Slave' а по сети PROFIBUS?
6. Как информацию с распределенной периферии передать в I-Slave?
7. Как в лабораторной работе проверить правильность работы сети PROFIBUS?
8. Как используется функция SFC14?
9. Как используется функция SFC15?

### **3. РАБОТА №3. ИЗУЧЕНИЕ СЕНСОРНОЙ ПАНЕЛИ ОПЕРАТОРА SIEMENS TP177A И ЕЕ ИСПОЛЬЗОВАНИЯ В СИСТЕМАХ АВТОМАТИЗАЦИИ**

#### **3.1. Цель работы**

Изучить функциональные возможности сенсорной панели оператора (сенсорного монитора) SIEMENS TP177A и приобрести навыки ее программирования для автоматизации управления заданным объектом.

#### **3.2. Содержание работы**

1. Изучить возможности и особенности лабораторного стенда с сенсорным монитором.

2. Изучить назначение и технические характеристики сенсорного монитора SIEMENS TP177A.

3. Изучить принципы программирования сенсорного монитора.

4. Дома, при подготовке к работе, подготовить:

– как минимум три картинки программируемого пульта управления виртуальным объектом по заданному варианту (выбор режима, ручное управление, автоматическое управление);

– программу для программируемого контроллера SIMATIC S7-300 для проверки функционирования подготовленного пульта управления совместно с контроллером.

5. В лаборатории:

– освоить графическую среду программного обеспечения WinCC flexible;  
– с помощью программы WinCC flexible на ПЭВМ набрать подготовленные картинки сенсорного монитора;

– записать проект в память сенсорного монитора;

– сконфигурировать центральную стойку контроллера SIMATIC S7-300;

– набрать на компьютере подготовленную дома программу и откомпилировать ее;

– ввести программу в контроллер и убедиться в правильности её работы;

– при совместной работе сенсорного монитора и программируемого контроллера проверить правильность функционирования пульта оператора;

– подготовить отчёт и сделать выводы по работе.

#### **3.3. Описание лабораторного стенда**

На рис. 3.1 представлена структура лабораторного стенда для изучения программирования сенсорного монитора и его совместной работы с программируемым контроллером SIMATIC S7-300. Лабораторный стенд включает в себя ПЭВМ типа IBM 1 и аппаратный блок 2.

В свою очередь аппаратный блок включает в себя:

- источник питания 3;
- сенсорную панель оператора 4;
- блок программируемого контроллера 5;
- устройство преобразования кодов (УПК) 6;
- элементы ввода/вывода дискретных сигналов 7;
- кнопка изменения режимов работы 8;
- кнопка сброса УПК в исходное состояние 9;
- кабельный разъем для связи ПЭВМ с УПК 10;
- кабель с адаптером для связи ПЭВМ с контроллером 12.

Блок контроллера включает в себя:

- процессорный модуль CPU314 (серийный номер 314-1AE04-0AB0);
- модуль ввода дискретных сигналов SM321 DI16xDC24V (заказной номер 321-1BH02-0AA0) на 16 каналов ввода сигналов постоянного тока напряжением 24 В;
- модуль вывода дискретных сигналов SM322 DO16xDC24V/0,5A (заказной номер 322-1BH01-0AA0) на 16 каналов вывода сигналов постоянного тока напряжением 24 В с допустимым током нагрузки 0,5 А.

На входы контроллера подаются сигналы с двух кнопок с фиксацией SB1 и SB2 и двух кнопок без фиксации SB3 и SB4. Дискретные выходы выводятся на светодиоды VD1...VD4.

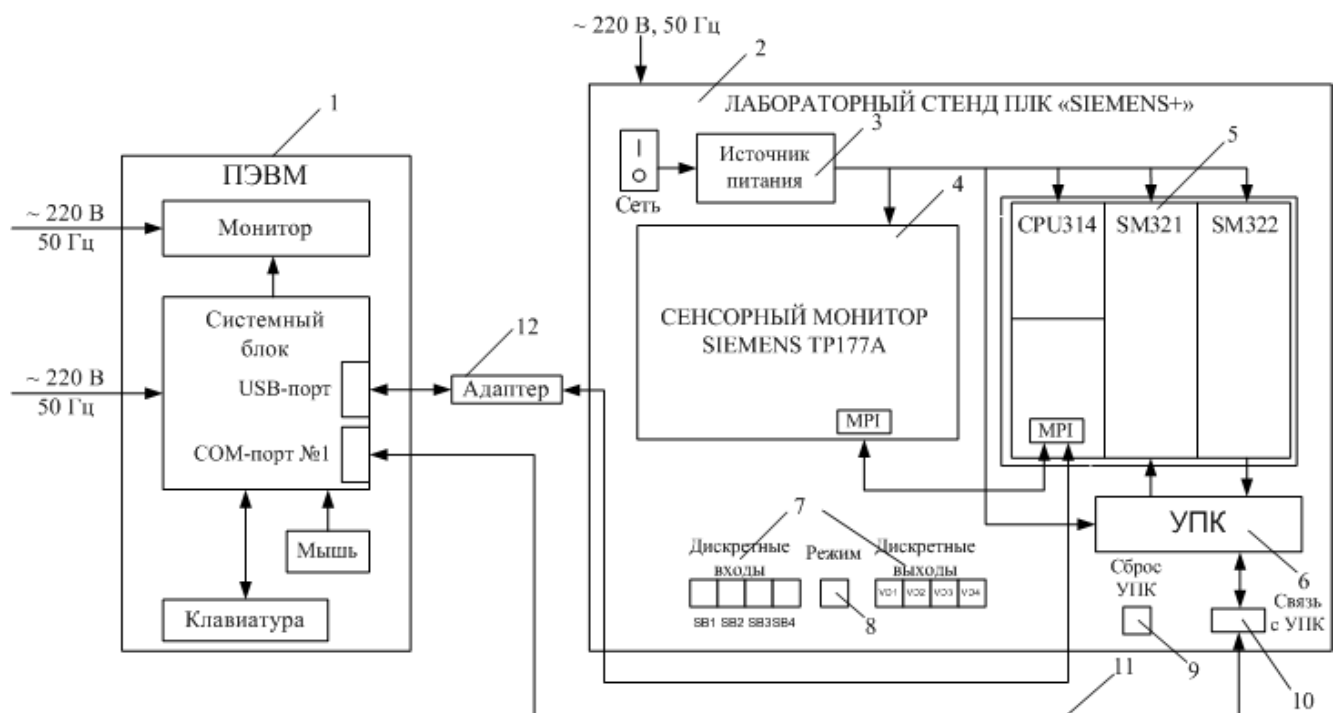


Рис. 3.1. Структурная схема стенда

На рис. 3.2 представлена схема электрическая принципиальная аппаратного блока. На схеме блок питания А3 обеспечивает питание напряжением постоянного

тока 24В сенсорного монитора А1, процессорного модуля CPU, модулей ввода и вывода программируемого контроллера А2, а также цепей устройства преобразования кодов (УПК) А4. Напряжение 5В необходимо для питания микроконтроллера УПК.

Цифровой модуль ввода SM321 имеет 2 байта каналов ввода с адресами I0.0 – I0.7 и I1.0 – I1.7. Входы I0.0 – I0.7 и I1.0 – I1.3 подключены к плате УПК, а с адресами I1.4 – I1.7 – к релейному блоку А5. При отключенной кнопке SB6 «Режим» эти входы подключаются к кнопкам входных сигналов на лицевой панели аппаратного блока.

Цифровой модуль вывода SM322 имеет 2 байта каналов вывода с адресами Q4.0 – Q4.7 и Q5.0 – Q5.7. Выходы с адресами Q4.0 – Q4.7 и Q5.0 – Q5.3 подключены к плате УПК для управления исполнительными устройствами виртуальных объектов, а с адресами Q5.4 – Q5.7 – к светодиодным индикаторам VD1 – VD4 на лицевой панели аппаратного блока.

Данная лабораторная работа ориентирована на ее проведение после выполнения лабораторной работы №1 по программированию контроллера SIMATIC S7-300. В работе №1 входные сигналы на контроллер подавались с тумблеров и кнопок, выходные сигналы – на светодиоды. Объект автоматизации как таковой отсутствовал и реализовывался мысленно.

Особенностью же данной лабораторной работы является наличие объекта автоматизации, имеющего датчики, контролирующие положение механизмов объекта и исполнительные устройства, обеспечивающие перемещение механизмов объекта с заданной скоростью. Объект автоматизации, его датчики и исполнительные устройства представлены виртуально (мультипликацией) на мониторе ПЭВМ и полностью имитируют работу реального объекта.

На экране монитора двигатели включаются (изменяют свой цвет), механизмы перемещаются, воздействуя на датчики, датчики включаются/отключаются (меняют цвет). Сигналы с датчиков виртуального объекта в виде 1,5 байтов информации из ПЭВМ в последовательном коде через СОМ-порт ПЭВМ передаются на УПК (рис.3.3). УПК преобразует последовательный код в параллельный, который затем поступает на 1,5 байта каналов модуля ввода контроллера SIMATIC S7-300. Контроллер по программе, которую составил и ввел обучаемый, формирует выходные команды, которые выдаются на 1,5 байта каналов модуля вывода контроллера. Схемно каналы модуля выхода параллельно подключены к УПК, которое преобразует параллельный код в последовательный и через СОМ-порт передает на исполнительные устройства виртуального объекта.



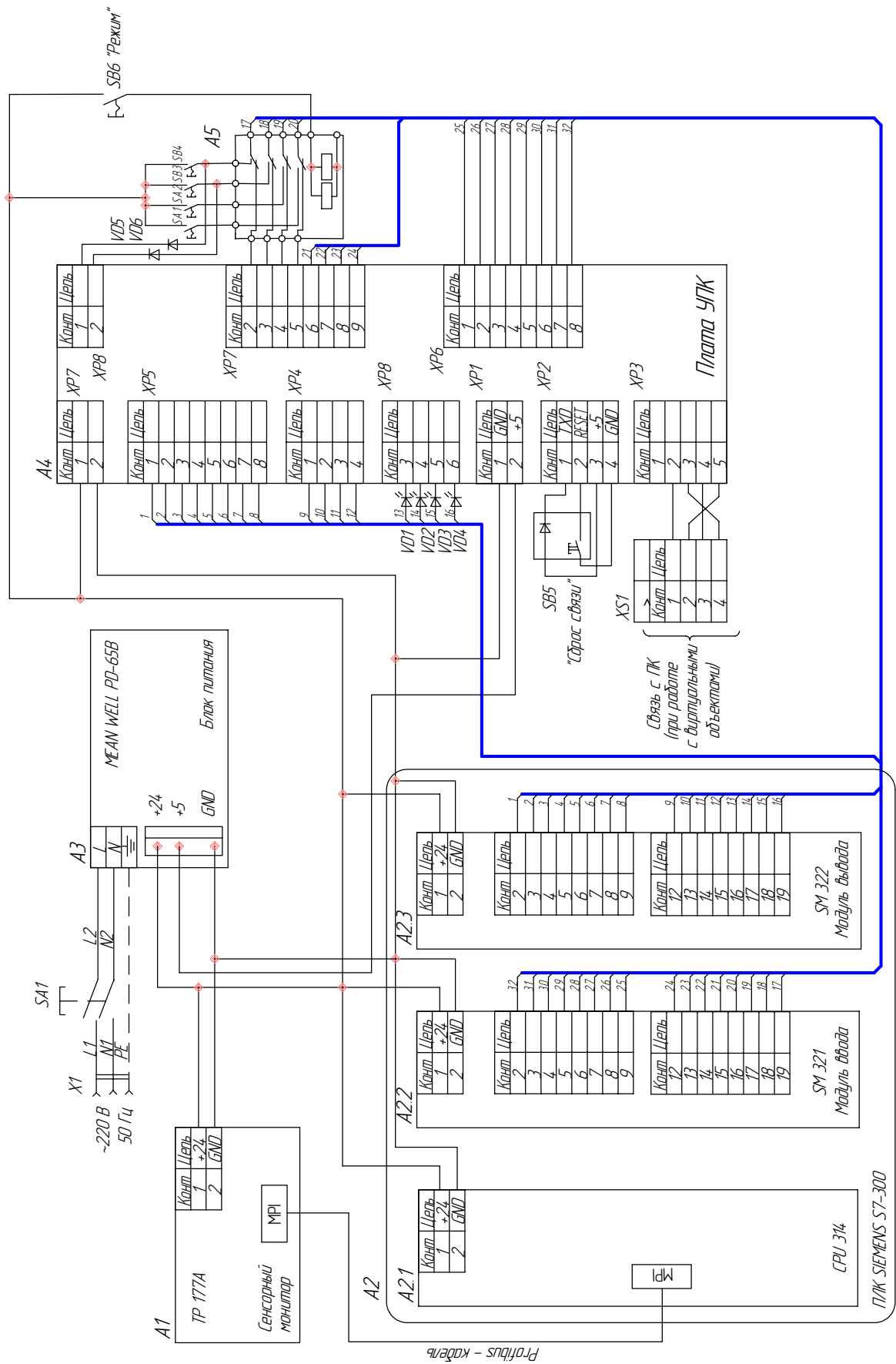


Рис. 3.2. Схема электрическая принципиальная лабораторного стенда

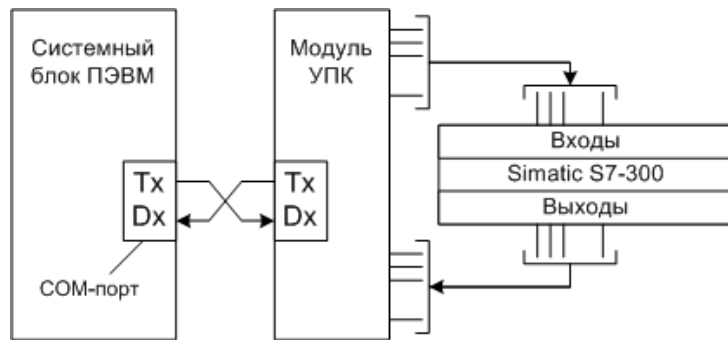


Рис. 3.3. Передача сигналов и команд при управлении виртуальными объектами

Варианты виртуальных технологических объектов представлены в Приложении Б.

При управлении виртуальным объектом СОМ-порт ПЭВМ подключается кабелем через разъем «Связь с ПК» (разъем DB-9) к плате УПК аппаратного блока. При выведении на экран монитора виртуального объекта, например, «Напольно-крышечной машины» (рис. Б.3 Приложения Б) и при включении режима управления «От контроллера» ПЭВМ берет всю инициативу по связи с контроллером на себя. ПЭВМ непрерывно циклически передает контроллеру через УПК сигналы с датчиков виртуального объекта и принимает от контроллера команды на исполнительные устройства. Обмен информацией между ПЭВМ и контроллером индицируется свечением встроенного в кнопку «Сброс связи» светодиода. Отсутствие связи возможно из-за плохого контакта в разъемах кабеля СОМ-порт – аппаратный блок, несоответствия адреса используемого СОМ-порта и установленного в настройках программы РК&МК управления виртуальными объектами и т.п.

В табл. 3.1 представлено соответствие входов и выходов контроллера элементам на лицевой панели аппаратного блока и сигналам с виртуального объекта и командам на виртуальный объект.

Таблица 3.1

Адрес		Назначение	
Входы	I0.0	X0	Виртуальный объект
	I0.1	X1	
	I0.2	X2	
	I0.3	X3	
	I0.4	X4	
	I0.5	X5	
	I0.6	X6	
	I0.7	X7	
	I1.0	X8	
	I1.1	X9	
	I1.2	X10	
	I1.3	X11	

Входы	I1.4	X12/SB1	Виртуальный объект/панель стенда
	I1.5	X13/SB2	
	I1.6	X14/SB3	
	I1.7	X15/SB4	
Выходы	Q4.0	Y0	Виртуальный объект
	Q4.1	Y1	
	Q4.2	Y2	
	Q4.3	Y3	
	Q4.4	Y4	
	Q4.5	Y5	
	Q4.6	Y6	
Выходы	Q4.7	Y7	Виртуальный объект
	Q5.0	Y8	
	Q5.1	Y9	
	Q5.2	Y10	
	Q5.3	Y11	
	Q5.4	Y12	
	Q5.5	Y13	Панель стенда
	Q5.6	Y14	
	Q5.7	Y15	

### 3.4. Технические характеристики панели оператора TP177A

**Сенсорные панели оператора SIMATIC серии 177** предназначены для построения систем человеко-машинного интерфейса (HMI) и решения задач оперативного управления и мониторинга. Панели могут использоваться с программируемыми контроллерами SIMATIC S7-200/S7-300/S7-400, системами компьютерного управления SIMATIC WinAC, программируемыми контроллерами других фирм.

Семейство объединяет в своем составе:

- сенсорные панели оператора SIMATIC TP 177A с монохромным дисплеем;
- сенсорные панели оператора SIMATIC TP 177B с монохромным или цветным дисплеем;
- панели оператора SIMATIC OP 177B с монохромным или цветным дисплеем и мембранной клавиатурой.

Работа панелей базируется на использовании операционной системы Windows CE, существенно расширяющей их функциональные возможности:

- шрифты True Type;
- мощные графические возможности: графические изображения, гистограммы;

- динамические объекты: поля ввода-вывода, выпадающие окна выбора объектов;
- тренды кривых (ТР 177В/ОР 177В);
- парольная защита;
- библиотеки элементов;
- система обработки сообщений;
- обработка рецептов (ТР 177В/ОР 177В);
- многоязыковая поддержка, включая русский язык.

Внешний вид сенсорной панели оператора SIMATIC TP 177A с лицевой и тыльной сторон представлен на рис. 3.4. В табл. 3.2 представлены основные технические характеристики используемой панели оператора.



Рис. 3.4. Панель оператора SIMATIC TP 177A

Таблица 3.2

Характеристика		Значение
Номинальное напряжение питания		=24В
Потребляемая мощность		6Вт
Дисплей	Тип	CCFL (Cold Cathode Fluorescence Lamps)
	Разрешающая способность, точек	320*240
	Размеры области отображения информации	116*87 мм
	Цветность	4 оттенка голубого цвета
	Наработка на отказ	50000 часов минимум
Клавиатура		Резистивная, 1000000 срабатываний
Микропроцессор		32-разрядный с RISC архитектурой
Операционная система		Microsoft Windows CE
Flash-EEPROM пользователя		512 КБ
Встроенные интерфейсы		RS485
Скорость передачи данных: MPI / PROFIBUS-DP		до 187.5 Кбит/с / до 1.5 Мбит/с
Количество и тип подключаемых контроллеров		1(S7-200/300/400/WinAC)
Пакет конфигурирования		WinCC flexible
Загрузка конфигурации		MPI / PROFIBUS-DP

Основные функции	Экраны	250
	Переменные	500
	Текстовые элементы	1000
	Графические объекты	1000
	Оперативные сообщения	1000
	Аварийные сообщения	1000
	Парольная защита доступа	Есть
	Количество интерактивных языков	5
	Драйверы для других контроллеров	Есть
Степень защиты		IP65-фронтальная панель, IP20-остальная часть корпуса
Габариты		212 x 156мм
Масса		0,7 кг
Диапазон температур: рабочий / хранения и транспортировки		0...+50 C / -20...+60 C

### 3.5. Операционная система сенсорной панели SIMATIC TP 177A

Как и большинство сенсорных панелей фирмы SIEMENS данный сенсорный монитор обладает собственной операционной системой.

Сразу после подачи питания на монитор TP177A появляется окно загрузчика Loader (рис. 3.5). Кнопки загрузчика запускают выполнение следующих функций:

- кнопка «Transfer [Передача проекта]» – запуск режима передачи проекта на устройство HMI;
- кнопка «Start [Запуск]» – запуск проекта, который хранится на устройстве HMI;
- кнопка «Control Panel [Панель управления]» – открытие панели управления устройства HMI.

Панель управления используется для задания различных параметров, например, параметров передачи данных.

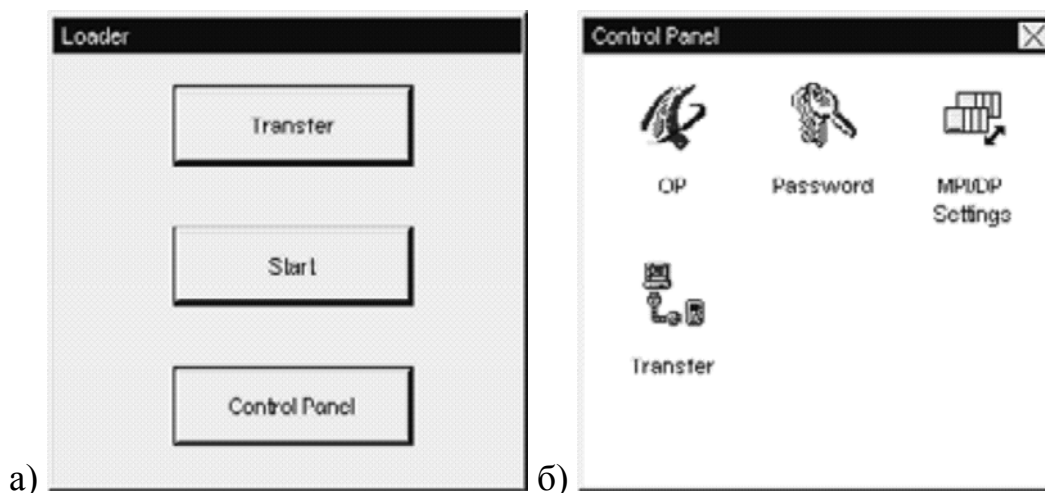


Рис. 3.5. Окна загрузчика (а) и панели управления (б) сенсорного монитора TP177A

Окно загрузчика открывается также при закрытии проекта.

Для изменения параметров настройки на панели управления необходимо выполнить следующие действия:

1. Перед изменением параметров на панели управления необходимо закрыть проект. Для этого используется соответствующий элемент операторского управления, созданный в проекте.

2. Открыть панель управления нажатием кнопки «Control Panel».

3. Открыть нужное диалоговое окно, дважды щелкнув на соответствующей иконке в панели управления.

4. Для изменения параметров настройки нужно коснуться соответствующего поля ввода или установить соответствующий флажок и, при необходимости, использовать отображенную на экране клавиатуру. Если панель управления защищена от неавторизованного доступа, необходимо ввести требуемый пароль. Изменить параметры настройки управления устройства НМІ в диалоговом окне.

5. Нажать  или , чтобы закрыть диалоговое окно.

6. Запустить проект с помощью загрузчика.

В панели управления особое внимание следует уделить меню настройки параметров панели оператора (OP) и меню передачи проекта (Transfer) (рис. 3.6). Первое меню используется для настройки контрастности экрана, ориентации экрана, задержки запуска проекта. На вкладке Device можно посмотреть информацию о мониторе TP177A. Второе меню необходимо для настройки канала передачи данных из ПЭВМ и ПК в монитор или обратно.

Для корректной работы сенсорного монитора в составе стенда также необходимо выставить следующие настройки в панели управления (рис. 3.7): в диалоговом окне «MPI-DP Settings» установить адрес устройства 1, в диалоговом окне «Transfer Settings» установить галочку напротив надписи «Enable Channel» в нижнем блоке «Channel 2».

Для корректной работы сенсорного монитора в составе стенда также необходимо выставить следующие настройки в панели управления (рис. 3.7): в диалоговом окне «MPI-DP Settings» установить адрес устройства 1, в диалоговом окне «Transfer Settings» установить галочку напротив надписи «Enable Channel» в нижнем блоке «Channel 2».

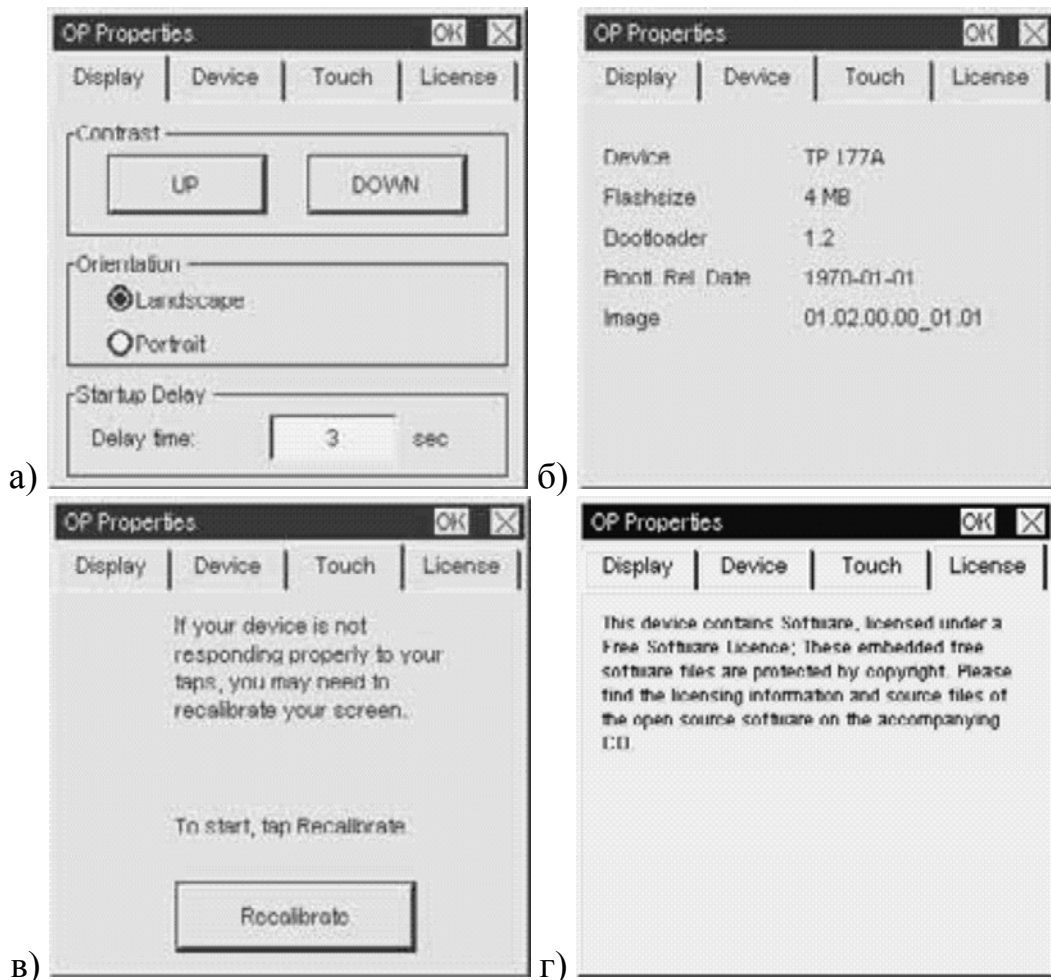


Рис. 3.6. Диалоговые окна панели управления сенсорного монитора TP177A

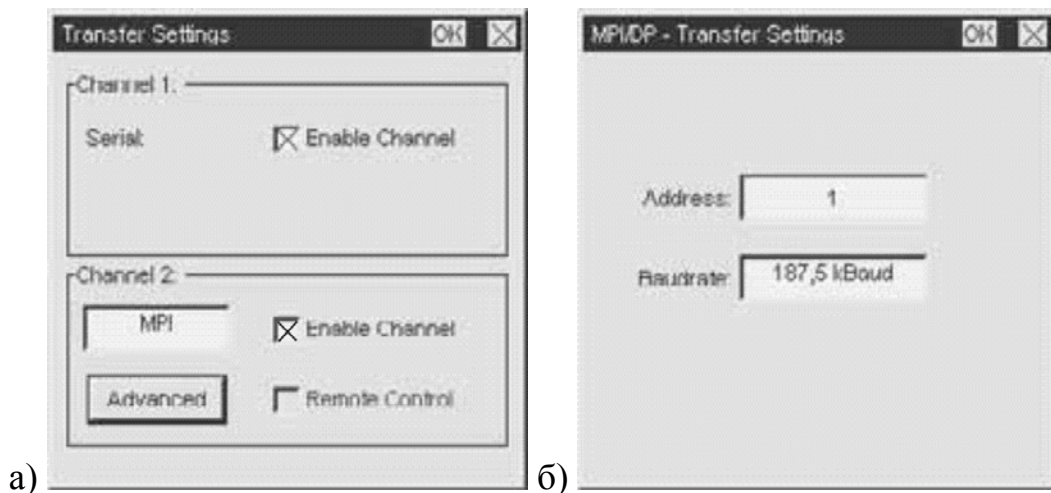


Рис. 3.7. Диалоговые окна панели управления сенсорного монитора TP177A

### 3.6. Программирование сенсорного монитора SIEMENS TP177A

#### 3.6.1 Общие положения

Программирование сенсорного монитора сводится к следующему:

- в программе WinCC flexible создается проект, с необходимым количеством экранов с возможностью их переключения;
- создается необходимое количество тегов и прописываются их адреса;
- из имеющегося набора графических инструментов на экране монитора формируется требуемое расположение, размер, цвет, функции и т.п. элементов виртуального пульта. Также есть возможность расположить на пульте необходимые надписи, рисунки;
- всем коммутационным и сигнальным элементам виртуального пульта назначаются соответствующие теги;
- проводится компиляция подготовленного проекта;
- проект по последовательному интерфейсу записывается в сенсорный монитор.

### 3.6.2 Начало работы с WinCC flexible. Создание нового проекта

Для программирования сенсорных панелей оператора TP177A фирмы SIEMENS используется специальная программа WinCC flexible. Ее запуск производится через соответствующий ярлык на рабочем столе или в меню «Пуск» персонального компьютера (Пуск → Все программы → SIMATIC → WinCC flexible 2007 → WinCC flexible). Проявится стартовая страница (рис. 3.8). Нажмите «Create a new project with the Project Wizard [Создать новый проект с помощью Мастера проектов]».

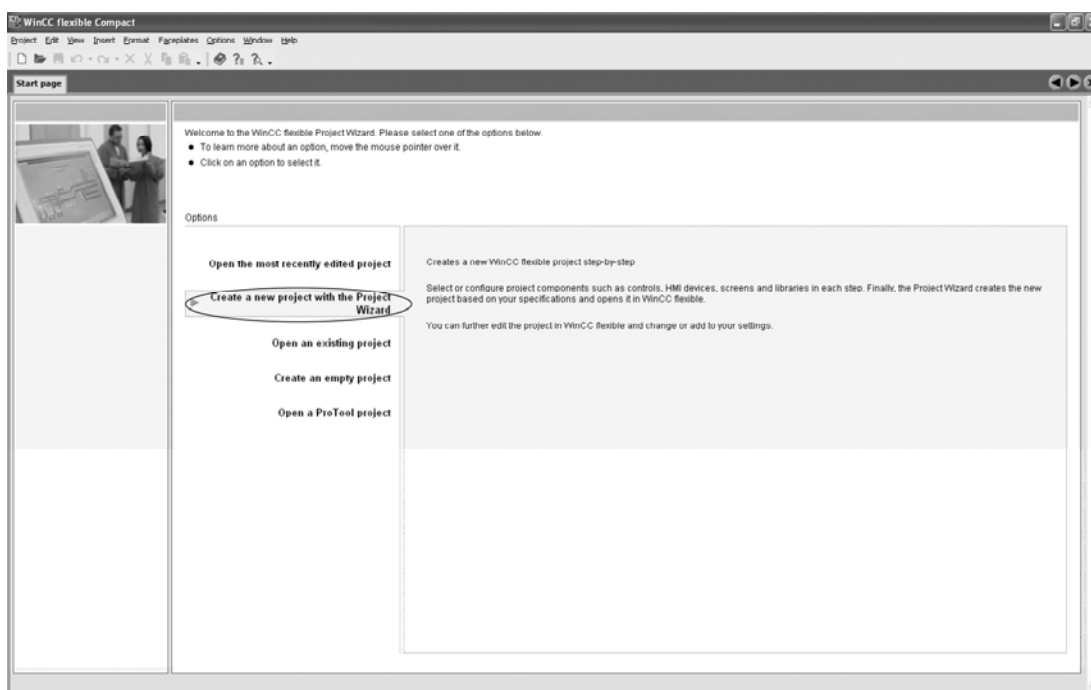


Рис. 3.8. Стартовая страница WinCC flexible

Выберите опцию «Small Machine [Малая установка]», затем нажмите «Next [Далее]» (рис. 3.9).



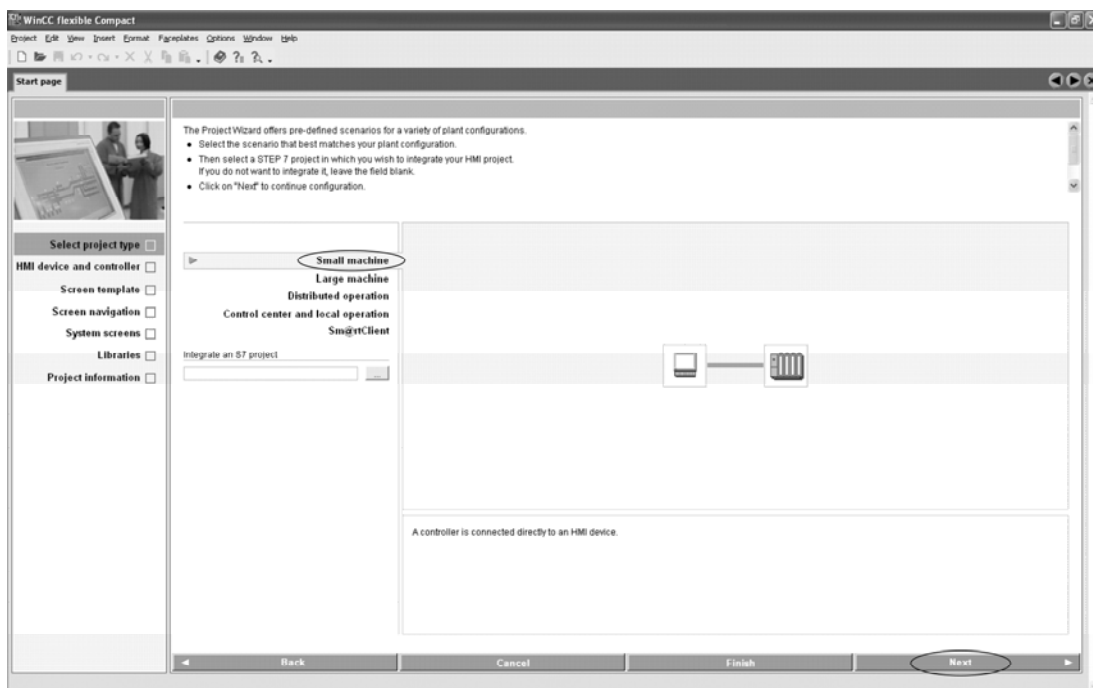


Рис. 3.9. Выбор типа проекта

Выберите в качестве устройства НМІ панель оператора «TP177A». При выборе контроллера укажите стандартный параметр «SIMATIC S7 300/400». Нажмите «Next [Далее]» (рис. 3.10).

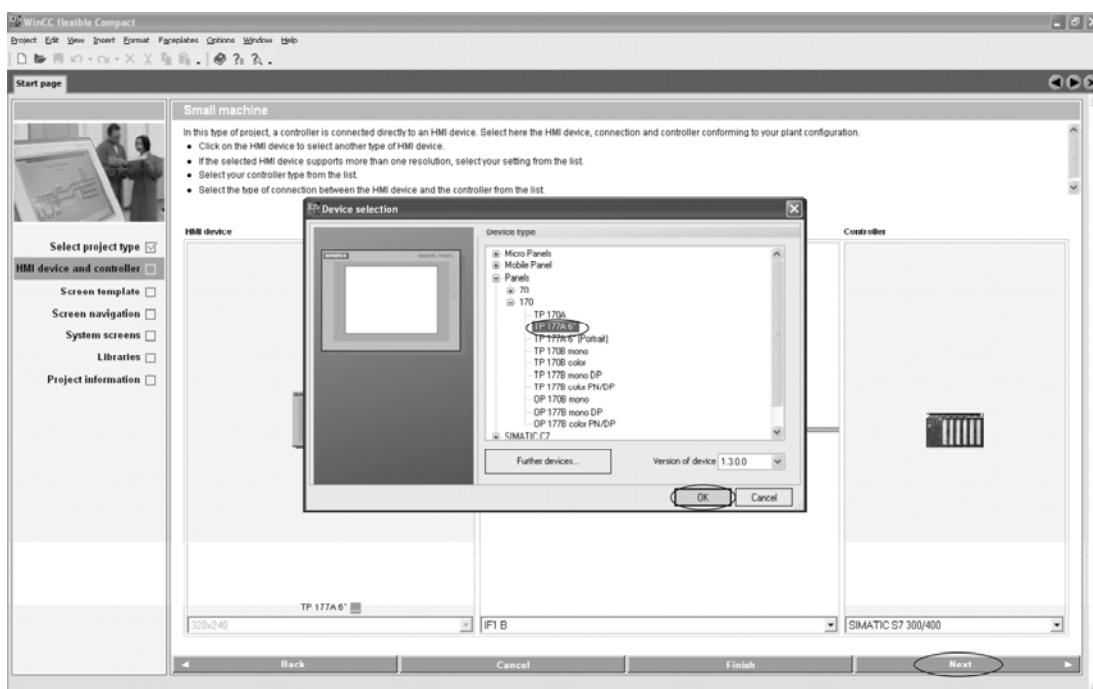


Рис. 3.10. Выбор устройства НМІ и контроллера

Щелкните на кнопке «Next [Далее]» для применения стандартных параметров настройки, приведенных на странице «Screen Templates [Шаблоны экранов]» (рис. 3.11).

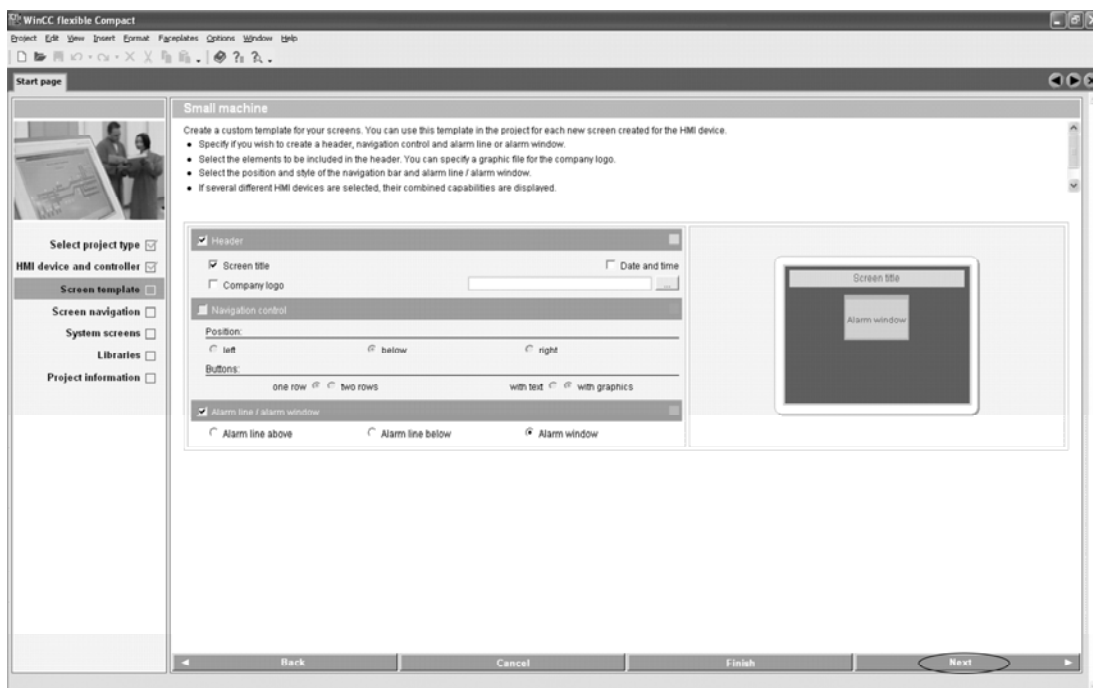


Рис. 3.11. Применение стандартных настроек

Определите структуру экранной навигации, создав требуемое количество экранов «Section screens [Экран секций]» (рис. 3.12). Возможно также и создание единственного стартового экрана.

Структура экранной навигации определяет фиксированную структуру экранов в проекте с соответствующими, определенными пользователем, сменами экранов, то есть переходами от экрана к экрану.

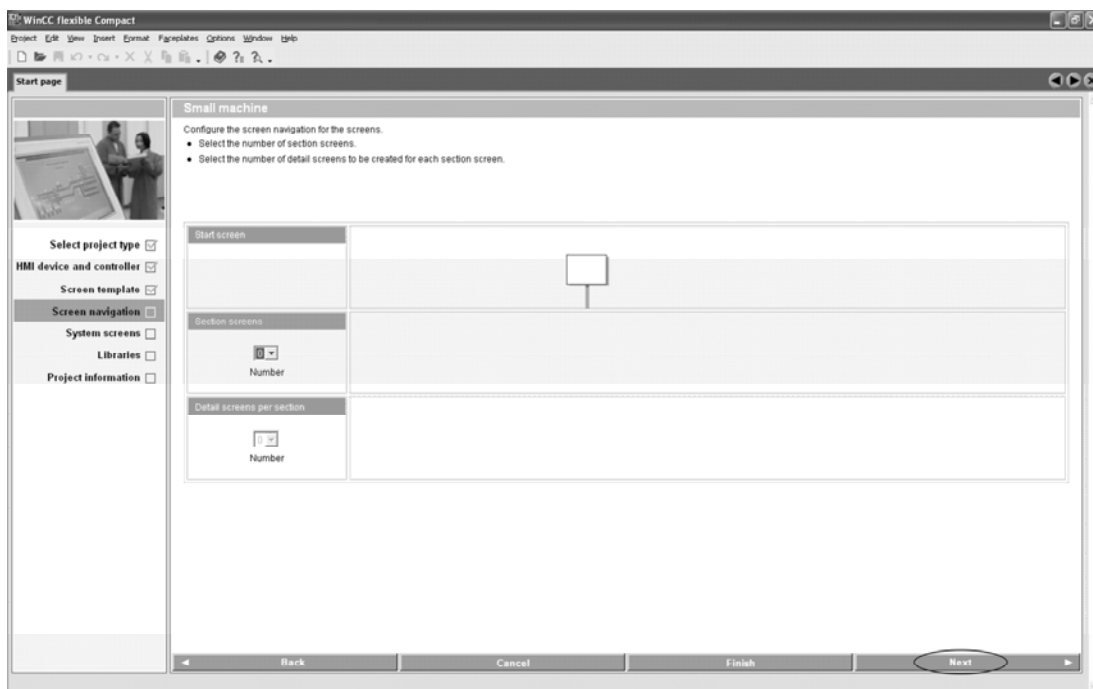


Рис. 3.12. Определение структуры экранной навигации

Щелкните по кнопке «Next [Далее]» для применения стандартных параметров настройки, приведенных на страницах «Libraries [Библиотеки]». Жмите «Next [Далее]», пока не появится окно, показанное на рис. 3.13. Затем, введя информацию о проекте (Project Name, Project author, Comments), нажмите «Finish».



Рис. 3.13. Ввод информации о проекте

Мастер проектов создает новый проект (рис. 3.14) в соответствии с заданной информацией, а затем открывает его в WinCC flexible. Слева располагается корневая структура проекта, содержащая все элементы, которые можно конфигурировать (рис. 3.15). Если ее нет, то необходимо выбрать в меню «View» позицию «Project».

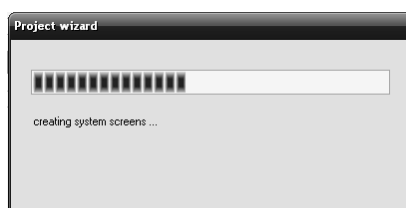


Рис. 3.14. Строка состояния создания нового проекта

Начальный экран для операторской панели TP177A открывается автоматически в рабочем пространстве, справа от корневой структуры проекта. Начальный экран содержит два окна сообщений, размещенных одно над другим (Окно сообщений об ошибках и окно системных сообщений). Серое затенение окна указывает на то, что это окно вставлено в шаблон. Для того, чтобы убрать серое затенение, снимите галочку  Use template  во вкладке «General» внизу проекта. Это делает

ся для того, чтобы фоновые сообщения не мешали создавать новые объекты на экране.

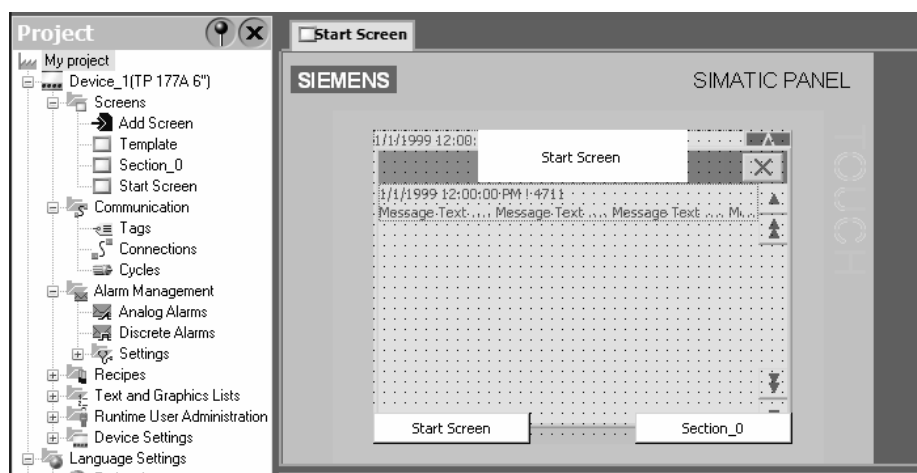


Рис. 3.15. Окно конфигурирования

В окне сообщений об ошибках отображаются вновь поступающие сообщения о процессе. В окне системных сообщений отображаются сообщения устройства HMI. Кроме того, уже определены параметры настройки соединения между устройством HMI и контроллером (рис. 3.16).

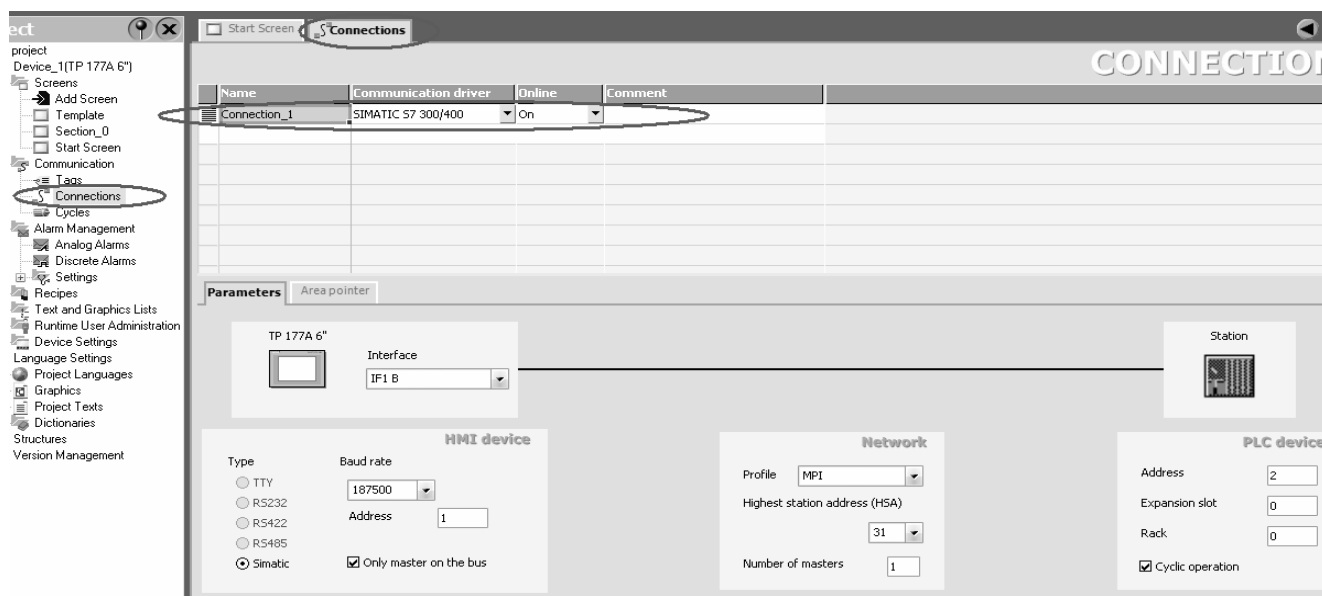


Рис. 3.16. Окно настройки соединения

При работе с выбранным экраном активизируется панель инструментов, расположенная над проектом (рис. 3.17). В табл. 3.3 дано краткое описание элементов этой панели.

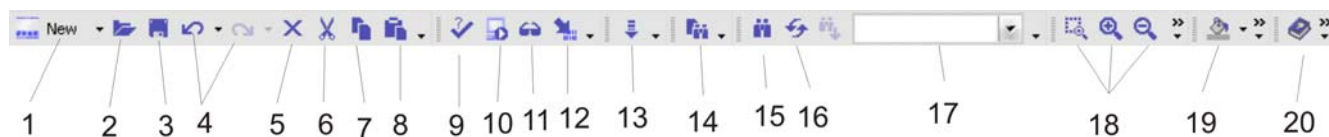


Рис. 3.17. Основные инструменты конфигурирования

Таблица 3.3

№ п/п	Описание	№ п/п	Описание
1	Новый проект	11	Компиляция и отладка
2	Открыть проект...	12	Компиляция и запуск симулятора
3	Сохранить проект	13	Загрузка проекта
4	Отменить очередную операцию проекта	14	Поиск объекта в проекте
5	Удалить объект	15	Найти текстовые строки
6	Вырезать объект	16	Заменить текстовые строки
7	Копировать объект	17	Область поиска
8	Вставить объект	18	Изменение масштаба
9	Сгенерировать	19	Панель «Рисование»
10	Запуск симулятора	20	Вызов справки

### 3.6.3 Работа с тегами

Все теги делятся на внешние и внутренние. Внешние теги служат для связи (обмена данными) между компонентами системы автоматизации процесса, например, между устройством НМІ и ПЛК. Внешний тег является образом определенной области (ячейки) памяти в ПЛК. Доступ к чтению и записи в эту область памяти осуществляется с НМІ-устройства и с ПЛК. Поскольку внешние теги являются образом области памяти в ПЛК, то возможные типы данных зависят от ПЛК, который связан с НМІ-устройством (табл. 3.4).

Таблица 3.4

Имя	Объект	Типы данных
Tag [Тег]	V	Char, Byte, Int, Word, DInt, DWord, Real, Bool, StringChar
Input [Вход]	I	
Output [Выход]	Q	
Bit memory [Битовая память]	M	
Timer [Таймер]	T	Timer
Counter [Счетчик]	C	Int

Внутренние теги не связаны с ПЛК и хранятся в памяти устройства НМІ. Поэтому доступ к чтению и записи внутренних тегов осуществляется только с устройства НМІ. Внутренние теги создаются, например, для выполнения локальных

вычислений. Для внутренних тегов возможны следующие типы данных: Char, Byte, Int, Uint, Long, Ulong, Float, Double, Bool, String и DateTime.

Все теги создаются в редакторе «Tags [Теги]». При создании тега для него определяется ряд основных свойств или параметров настройки. С помощью редактора «Tags [Теги]» свойства тега можно адаптировать к требованиям конкретного проекта (рис. 3.18). Для создания очередного тега необходимо двойным щелчком мыши кликнуть на пустую строку в закладке «Tags».

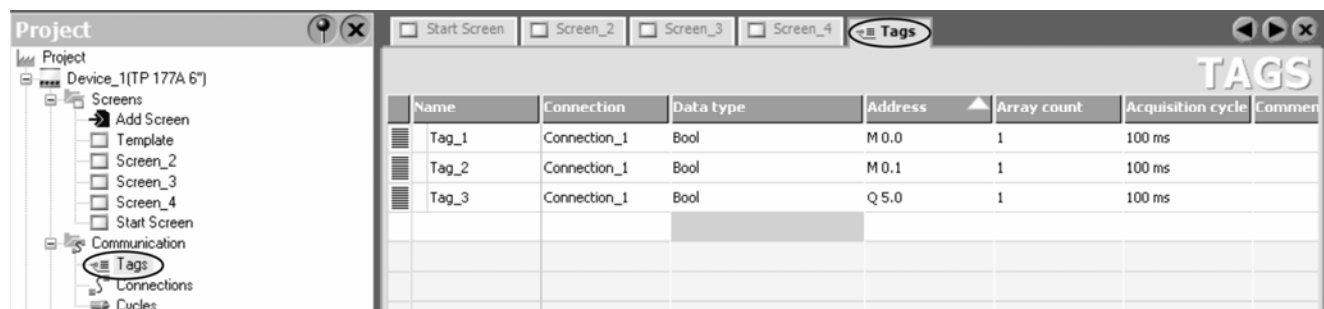


Рис. 3.18. Настройка тега

Для тега могут быть определены следующие свойства:

- «Name [Имя]». Каждый тег имеет имя, которое можно выбрать по своему усмотрению. Имя должно быть уникально в пределах проекта.

- «Connection [Соединение]» с ПЛК и «Logging cycle [Цикл архивации]» тега. Необходимо создать соединение с ПЛК для внешних тегов. Внешний тег является образом области памяти в ПЛК и служит для хранения значения, передаваемого из ПЛК. Путем задания цикла опроса можно определить, как часто и в какое время будет обновляться тег.

- «Data type [Тип данных]» и «Length [Длина]». Тип данных тега определяет тип значений, которые будут храниться в этом теге, внутренний способ хранения значений и максимальный диапазон возможных значений тега. Два простых примера типов данных – это «Int» для хранения целых значений и «String» для хранения символьных строк. Для тегов текстовых типов «String» или «StringChar» вы можете задать длину тега в байтах. Для других типов данных значение параметра «Length [Длина]» фиксировано.

- «Array count [Размер массива]». Можно составить тег из ряда элементов массива одного и того же типа. Элементы массива при этом хранятся в соседних ячейках памяти. Теги-массивы используются главным образом при работе с данными одного вида.

- «Comment [Комментарий]». Комментарий можно ввести для каждого тега.

- «Limits [Уставки]». Можно задать диапазон значений тега, определяемый верхней и нижней уставкой. Уставки будут использоваться для контроля значений процесса, для которых создан данный тег.

Все свойства, определенные при создании тега, можно редактировать позднее с помощью списка объектов.

### 3.6.4 Создание объектов на экране панели оператора

После того, как созданы теги и сделана общая подготовка экрана, можно приступить к созданию конкретных объектов экрана.

Пример реализации пульта оператора для управления простейшим технологическим процессом: необходимо запрограммировать пульт оператора для управления тележкой для транспортировки деталей между пролетами цеха. Имеется два фиксированных положения П0 и П1. На каждом положении расположена кнопка отправки в противоположное положение, причем каждая из этих кнопок является виртуальным объектом сенсорного монитора.

На основе описания технологического процесса составлены следующие логические уравнения:

$V = (K_{н.в} + V) \cdot \overline{П1}$  – Команда на движение тележки вперед;

$H = (K_{н.н} + H) \cdot \overline{П0}$  – Команда на движение тележки назад;

$K_{н.в}$  – кнопка для движения тележки вперед;

$K_{н.н}$  – кнопка для движения тележки назад.

На основе приведенных логических уравнений создается программа в среде STEP-7, которая в дальнейшем записывается в ПЛК SIEMENS SIMATIC S7-300. С подробным описанием загрузки программы в ПЛК можно ознакомиться в руководстве к лабораторной работе №1 по изучению ПЛК SIEMENS SIMATIC S7-300.

На панели оператора необходимо создать кнопки «Вперед» и «Назад»

В правом окне «Tools» представлены разные инструменты: Line, Ellipse и прочее. Необходимо выбрать строку «Button» и левой кнопкой мыши перетащить ее на панель оператора. Таким образом необходимо создать две кнопки (рис. 3.19).

Кликнув на кнопку, внизу отобразится вкладка настроек кнопки (рис. 3.20). Здесь можно придать кнопке особенный вид, задав в пункте «Graphic» конкретный вид, либо сделать на кнопке надпись, обозначенную в области «Text».

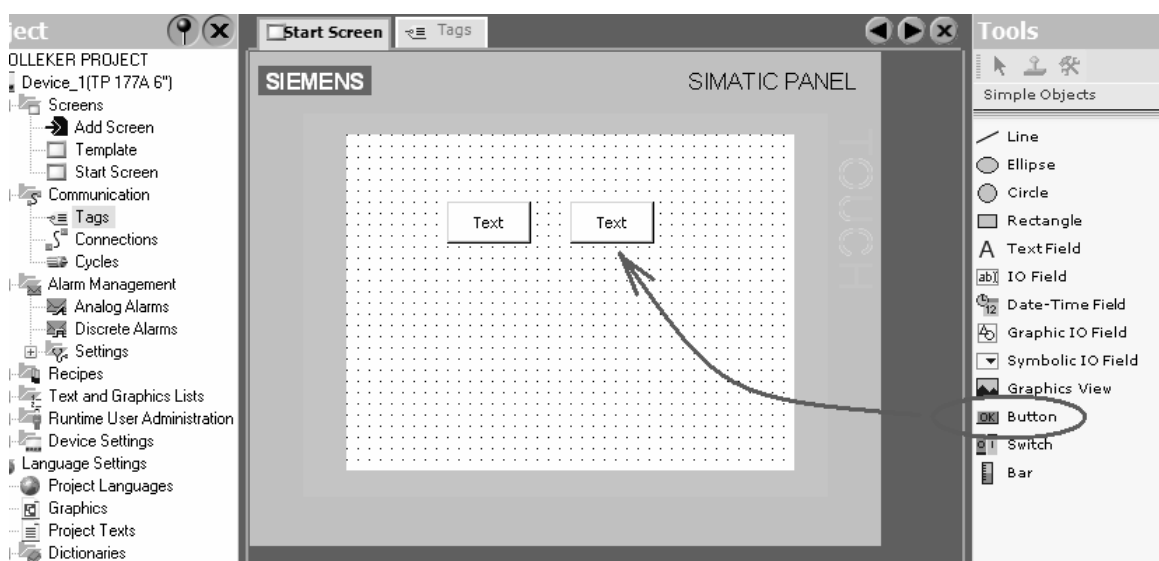


Рис. 3.19. Создание кнопок

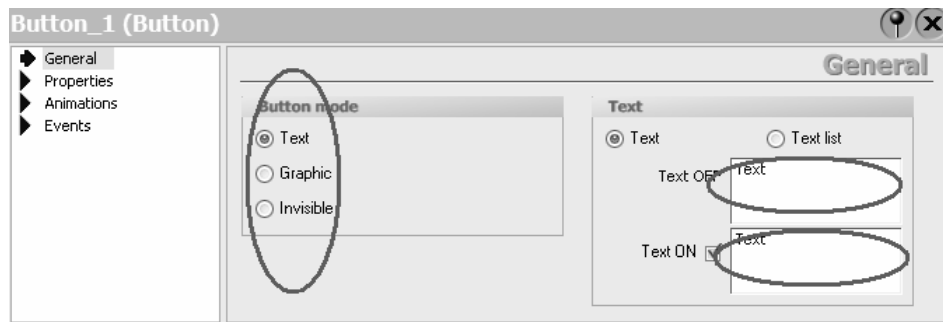


Рис. 3.20. Вкладка настроек кнопки

При необходимости перехода на русский алфавит необходимо на клавиатуре ПЭВМ нажать кнопки Shift+Ctrl или Alt+Shift (в зависимости от настройки клавиатуры).

Далее необходимо выбрать Events [События] → Press и обозначить в строке Edit bits > SetBit (рис. 3.21).

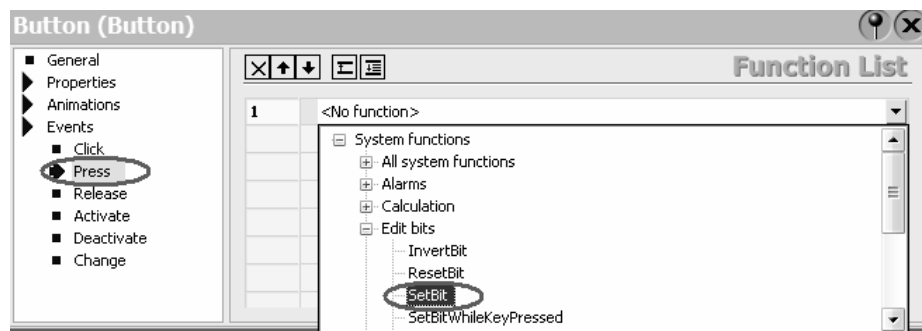


Рис. 3.21. Настройка кнопок

Далее необходимо выполнить аналогичную операцию: выбрать Events [События] → Release и обозначить в строке Edit bits → Reset bit.

В «Press» и «Release» нужно установить один и тот же заранее подготовленный тег (рис. 3.22). Для кнопок в тегах нельзя использовать адреса физических входов контроллера I, а следует устанавливать адреса промежуточных переменных (меркерров) M.

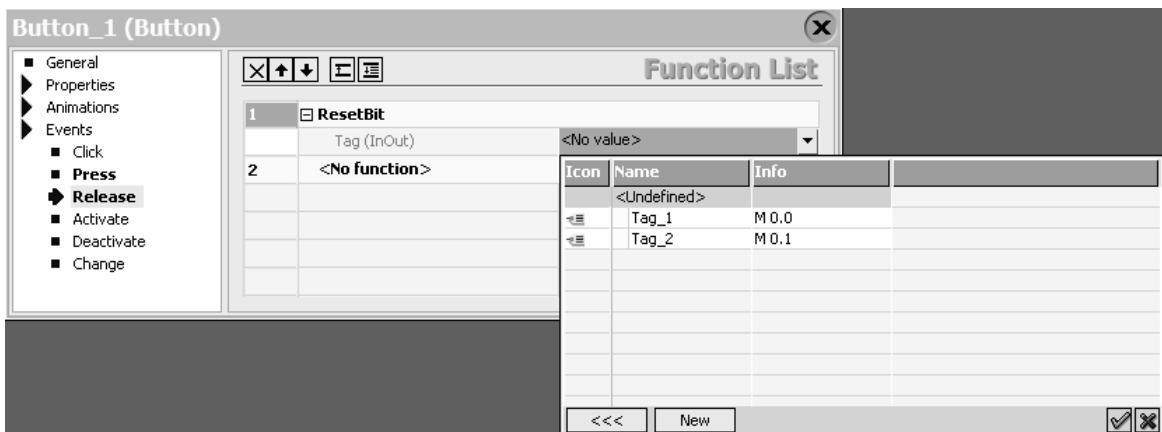


Рис. 3.22. Присвоение тега



Для кнопки «Назад» необходимо выполнить те же операции.

В пункте «Properties» можно присвоить кнопкам конкретные цвета (четыре голубых оттенка), а также и надписям на них. После выполненных операций отобразится экран, представленный на рис. 3.23.

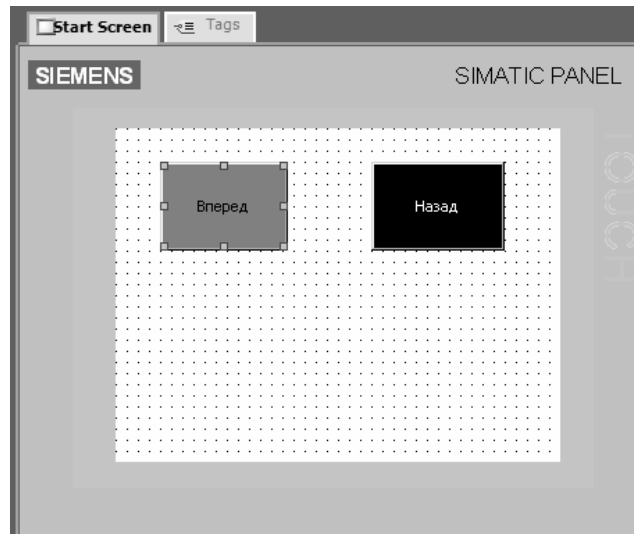


Рис. 3.23. Сконфигурированные объекты панели

Возможно программирование кнопки с функцией, когда первое нажатие и отпускание кнопки замыкает контакт, а второе нажатие и отпускание кнопки размыкает контакт. Для этого во вкладке настроек кнопки (см. рис. 3.21) необходимо выбрать Events → Click → InvertBit. Затем необходимо установить заранее подготовленный тег (рис. 3.22).

Далее необходимо создать кнопки перехода с экрана на экран.

Необходимо выбрать строку «Button» и левой кнопкой мыши перетащить ее на панель оператора (рис. 3.24).

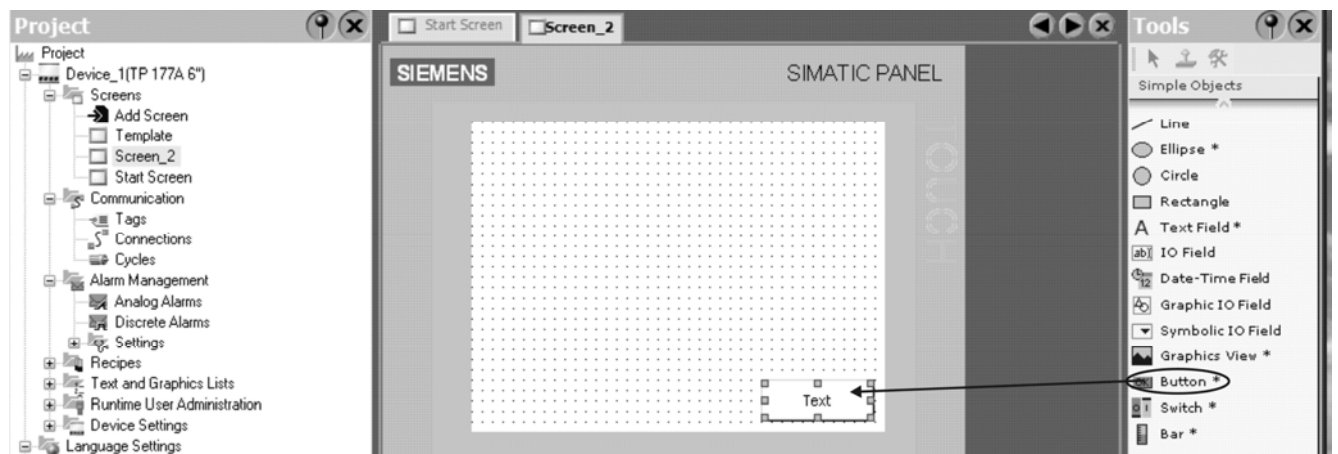


Рис. 3.24. Создание кнопки перехода на следующий экран

Кликнув на кнопку, внизу отобразится вкладка настроек кнопки (рис. 3.25). Здесь можно придать кнопке необходимый вид, задав в пункте «Graphic» конкретный вид, либо сделать на кнопке надпись, обозначенную в области «Text».

В структуре проекта в подкаталоге Screens отражены все Screen'ы проекта: Start Screen, Screen 1 и т.п. Следует эти названия заменить, например функциональными названиями: МЕНЮ, АВТ, РУЧ и т.п. Этим самым задаются внутренние теги, которые используются при формировании переходов.

На экранах создается необходимое количество кнопок переходов (рис. 3.24). Например, на экране МЕНЮ – две кнопки АВТ и РУЧ переходов на экраны соответственно автоматического и ручного режимов работы. На экране АВТ тоже устанавливаются две кнопки переходов на экраны МЕНЮ и РУЧ. Аналогично для экрана РУЧ – кнопки МЕНЮ и АВТ.

При нажатии на кнопку возникает вкладка настроек кнопки (рис. 3.25). Здесь задается надпись на кнопке, обозначенная в области «Text».

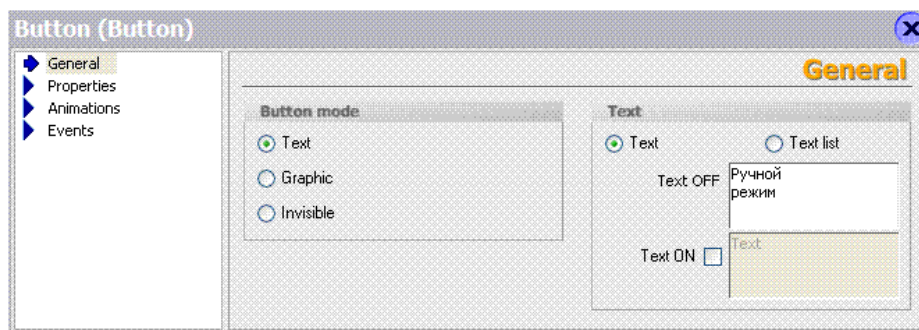


Рис. 3.25. Вкладка настроек кнопки

Далее необходимо выбрать Events → Click. В первой строке появившегося окна убрать надпись «No function» (рис. 3.26). Появляется вкладка, в которой следует выбрать функцию «ActivateScreen [Открыть экран]», кликнуть на поле «ScreenName [Имя экрана]». В поле значения отобразится кнопка выбора. С помощью этой кнопки откроется список тегов экранов, на которые обеспечивается переход. Далее необходимо выбрать требуемый тег. На этом заканчивается программирование кнопки перехода.

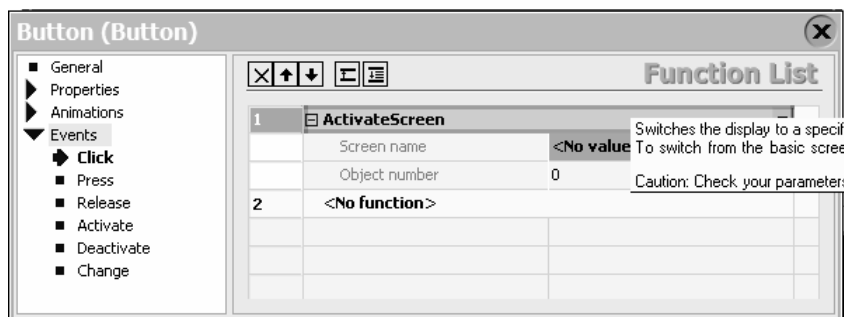


Рис. 3.26. Настройка кнопок

После выполненных операций отобразятся экраны, представленные на рис. 3.27.

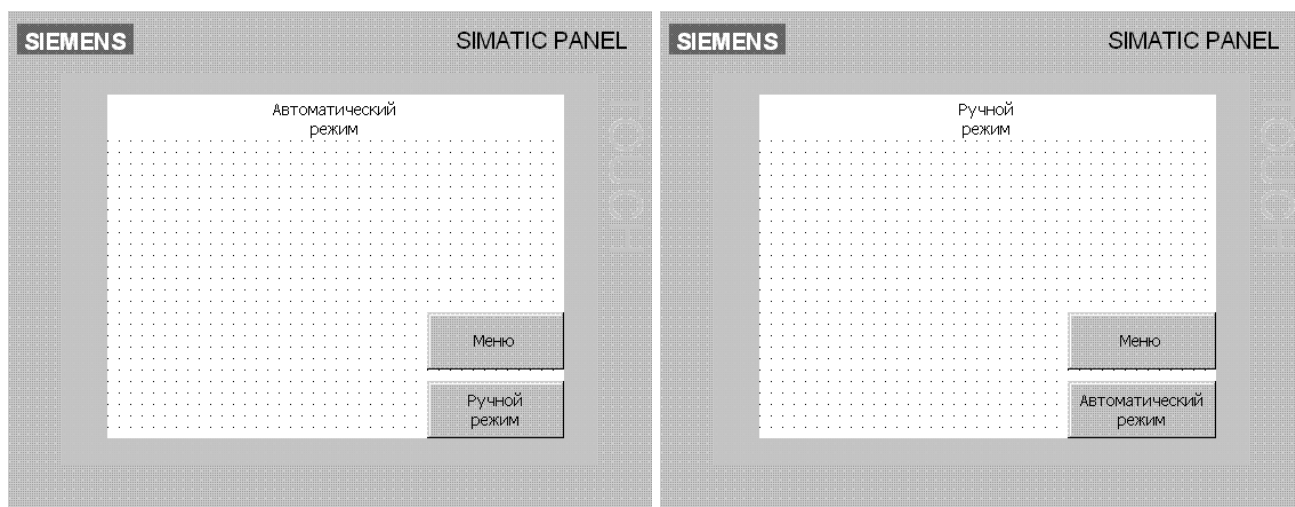


Рис. 3.27. Сконфигурированные кнопки перехода с экрана на экран

В связи с усеченной версией программы WinCC flexible, используемой в лабораторном стенде, возникают проблемы с программированием индикатора (лампочки) на экране монитора. Ее программирование ведется в следующей последовательности:

- в окне «Tools» выбрать строку «Circle» и левой клавишей мыши перетащить ее на панель оператора (рис. 3.28). Возникшую окружность увеличить или уменьшить до требуемых размеров;

- выбрать вновь строку «Circle», перетащить ее на панель оператора и поместить возникшую вторую окружность внутри первой, совместив их центры;

- для большой окружности не создается никаких тегов, а для внутренней предусматривается тег;

- при клике внутри малой окружности возникает вкладка настроек (рис. 3.29);

- выбрать Properties → Border color и установить белую границу окружности, а в позиции Fill color – черную заливку, в окне Fill style установить Solid;

- выбрать Animations → Visibility и установить заранее подготовленный тег (рис. 3.30), например, с адресом Q5.7. Поставить галочку (точку) в позиции «Hidden».

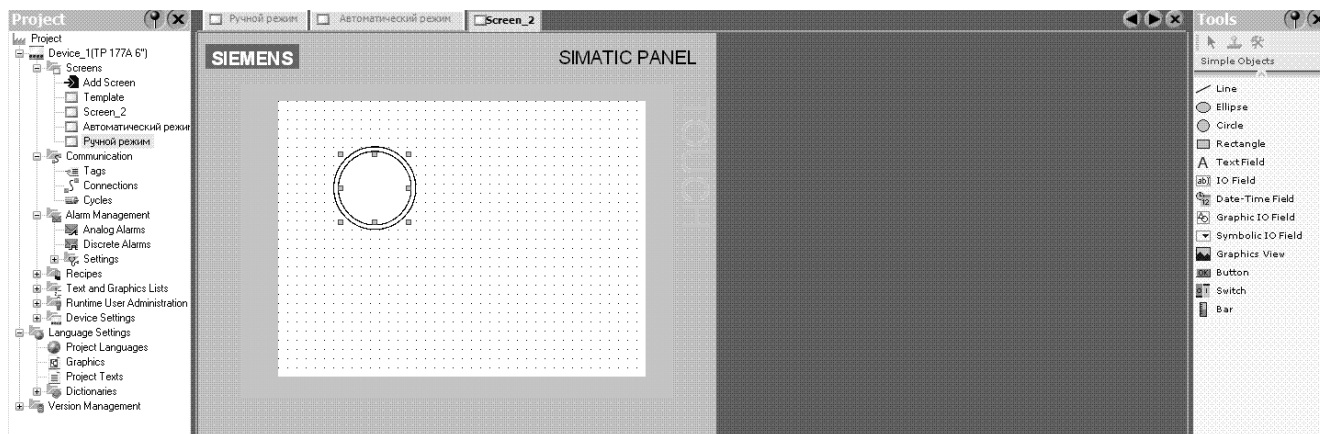


Рис. 3.28. Создание индикатора

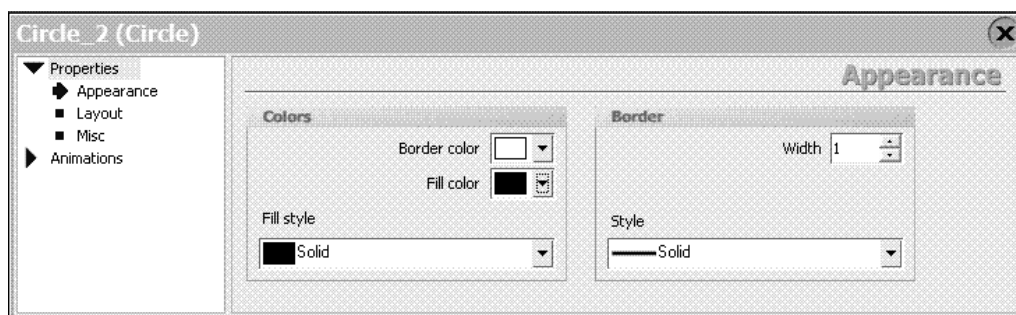


Рис. 3.29. Вкладка настроек

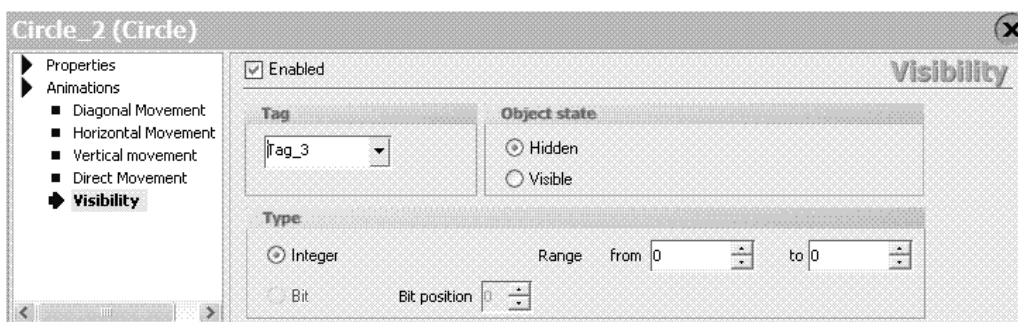





Рис. 3.30. Настройка управления индикатором

После того, как параметры панели сконфигурированы, необходимо скомпилировать проект. Перед компиляцией необходимо убедиться, что для всех объектов в пункте «Animations» настроек снята галочка в поле «Enabled».

Для компиляции проекта следует нажать на кнопку  «Start runtime system with simulator» в линейке инструментов. Если в линейке инструментов эта кнопка «спрятана», то необходимо через View → Toolbars поставить галочку напротив пункта «Compiler», нажав на него. В итоге на панели инструментов появятся кнопки для компиляции и симуляции проекта. После нажатия на  программа компилирует проект и, в случае исправной конфигурации, автоматически запускает симулятор монитора. Процесс компиляции отображается в окне «Output», доступ к которому осуществляется через View → Output в шапке проекта. При возникновении какой-либо ошибки, можно сразу ее увидеть.

### 3.6.5. Загрузка проекта в сенсорный монитор SIEMENS TP177A

Перед загрузкой проекта в монитор необходимо непосредственно настроить параметры связи с сенсорным монитором. Для определения параметров передачи надо выбрать команду меню  «Select transfer settings and start transfer to the device». Откроется диалоговое окно «Select devices for transfer [Выбор устройства для передачи]» (рис. 3.31). В появившемся окне необходимо установить следующие параметры: Mode – «MPI/DP», Station address – 1.

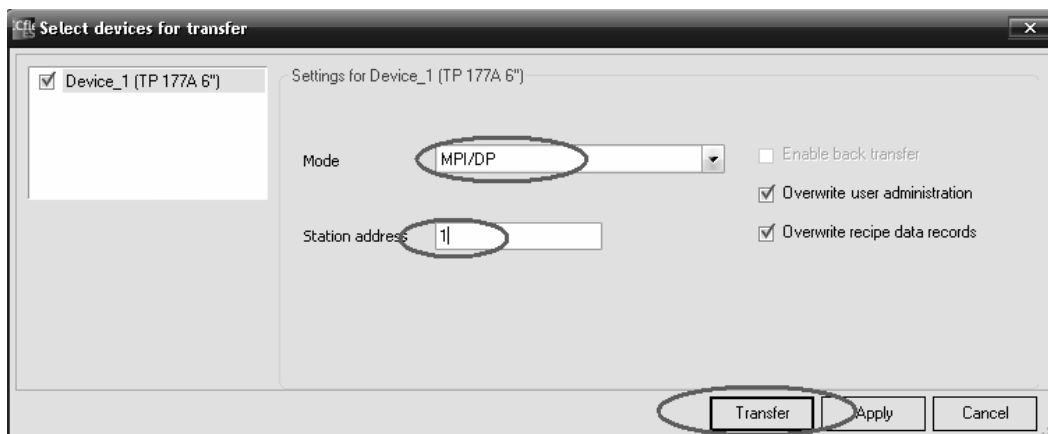
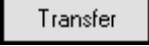


Рис. 3.31. Выбор устройства для передачи

Также необходимо удостовериться в подключении коммуникационного адаптера к порту MPI контроллера и порту USB компьютера. Далее нужно подать питание на сенсорную панель. Через некоторое время появится окно загрузчика (см. рис. 3.5а). Далее необходимо нажать кнопку  на самой панели. Только в этом режиме обеспечивается связь сенсорного монитора с персональным компьютером.

**ПРИМЕЧАНИЕ: во время обмена информацией между монитором и ПЭВМ запрещается выходить из данного режима.**

Для начала загрузки нужно в диалоговом окне «Select devices for transfer [Выбор устройства для передачи]» нажать кнопку «Transfer».

При правильном подключении и параметрах связи после этого начинается загрузка проекта в память сенсорного монитора, о чем сигнализирует следующее окно, изображенное на рис. 3.32.

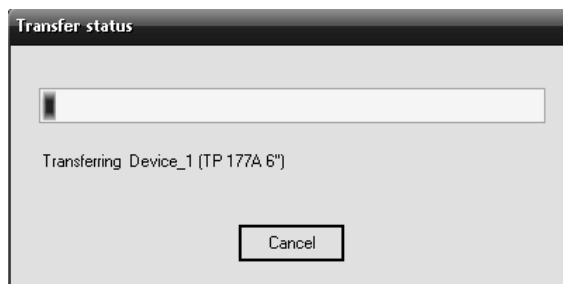


Рис. 3.32. Строка состояния загрузки проекта в память монитора

Во время загрузки возникает окно «Overwrite existing file», в котором будет задан вопрос о смене парольной защиты. Следует ответить «No [Нет]» (рис. 3.33), после чего загрузка проекта продолжится.

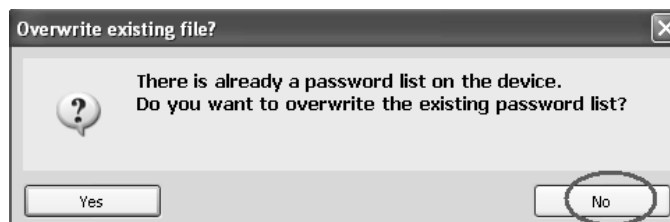


Рис. 3.33. Всплывающее окно при загрузке проекта

При неправильном подключении или параметрах связи на экране ПЭВМ появится окно с сообщением об ошибке. В этом случае нужно проверить настройки соединения как в программе WinCC flexible, так и в панели управления сенсорного монитора.

### 3.7. Выполнение лабораторной работы

Схема лабораторной установки перед началом работы полностью собрана и никаких изменений в процессе работы не предусматривается. Программы предыдущих бригад студентов удалены из компьютера.

В начале работы необходимо:

- ознакомиться с составом лабораторной установки, конструктивным исполнением компонентов стенда и соединениями между ними;
- включить компьютер, запустить программы «WinCC Flexible» и «SIMATIC Manager [Администратор SIMATIC]»;
- создать проект пульта управления и индикации для сенсорного монитора TP177A;
- провести конфигурирование программируемого контроллера, убеждаясь в ее правильности, используя команды меню Station → Consistency Check, перед выходом из окна «HW Config» с использованием команд меню «Save and Compile [Сохранить и компилировать]»;
- в организационный блок OB1 S7-300 ввести подготовленную дома программу;
- включить блок питания сети и убедиться в наличии питания всех компонентов стенда;
- осуществить стирание памяти CPU контроллера;
- записать подготовленные дома программы с конфигурацией в контроллер S7-300 и монитор TP177A;
- проверить правильность программирования контроллера и сенсорного монитора согласно выданного задания.

В табл. 3.4 представлены варианты задания для программирования сенсорного монитора. Варианты виртуальных объектов представлены в Приложении Б.

Таблица 3.4

Номер бригады	1			2			3			4		
Член бригады	1	2	3	1	2	3	1	2	3	1	2	3
Виртуальный объект	Вар. 1			Вар. 2			Вар. 3			Вар. 4		
Подвариант объекта	1	2	3	1	2	3	1	2	3	1	2	3
Номер бригады	5			6			7			8		
Член бригады	1	2	3	1	2	3	1	2	3	1	2	3
Виртуальный объект	Вар. 5			Вар. 6			Вар. 7			Вар. 8		
Подвариант объекта	1	2	3	1	2	3	1	2	3	1	2	3

### 3.8. Требования к отчёту

Отчет должен содержать:

- а) цель работы;
- б) описание технологической задачи по варианту, конфигурацию или технологическую схему автоматизированного объекта, обозначение переменных, содержательное описание формирования логических функций, логические функции, соответствующие работе объекта с учетом ручного и автоматического режимов работы;
- в) физические и символьные адреса переменных, отредактированные и проверенные программы для программируемого контроллера S7-300 и сенсорного монитора TP177A с комментариями к ним;
- г) рисунки с экранов программируемого пульта и комментарии к ним;
- д) описание методики и экспериментальной проверки работоспособности системы автоматизации в автоматическом и ручном режимах;
- е) выводы по работе.

### 3.9. Контрольные вопросы

1. Каково назначение сенсорного монитора?
2. Какие режимы работы есть у сенсорного монитора?
3. Какое максимальное количество картинок может содержать один проект?
4. Какое максимальное количество объектов может содержать одна картинка?
5. Как осуществляется переход от одной картинке к другой, если пульт включает в себя несколько картинок?
6. Как обеспечивается адресация элементов программируемого пульта управления?
7. Какой объект называется виртуальным?

8. Чем отличается программа программируемого контроллера для управления одним и тем же объектом при использовании программируемого пульта и при использовании обычного пульта с реальными кнопками, тумблерами и т.п.?

9. Каков порядок создания тега?

10. Каково назначение блока УПК в аппаратной части лабораторного стенда?

11. Как организован обмен информацией между виртуальным объектом и программируемым контроллером?



## **4. РАБОТА №4. ИЗУЧЕНИЕ СЕНСОРНОГО МОНИТОРА OMRON NT21**

### **4.1. Цель работы**

Изучить функциональные возможности сенсорного монитора и приобрести навыки программирования различных элементов на экране в соответствии с поставленной задачей.

### **4.2. Содержание работы**

1. Изучить назначение и техническую характеристику сенсорного монитора NT21.
2. Изучить принципы программирования сенсорного монитора .
3. Дома, при подготовке к работе, подготовить:
  - как минимум две картинку программируемого пульта управления виртуальным объектом по заданному варианту, обеспечивающего как ручное, так и автоматическое управление объектом;
  - программу для программируемого контроллера CPM2A для проверки функционирования подготовленного пульта управления совместно с контроллером.
4. В лаборатории:
  - с помощью программы NT-series Support Tool на ПЭВМ набрать подготовленные картинку сенсорного монитора;
  - записать проект в память сенсорного монитора;
  - с помощью программы CX-Programmer на ПЭВМ набрать, откомпилировать и записать в контроллер CPM2A подготовленную для него программу;
  - при совместной работе сенсорного монитора и программируемого контроллера проверить правильность функционирования пульта оператора при управлении виртуальным объектом;
  - подготовить отчёт и сделать выводы по работе.

### **4.3. Технические характеристики сенсорного монитора NT21**

Сенсорный монитор (сенсорная панель оператора) NT21 производится японской корпорацией OMRON. Данная модель обладает собственной операционной системой, широкими возможностями по программированию различных графических объектов на экране монитора, монохромным сенсорным экраном, большим перечнем совместимых программируемых контроллеров, причём не только «родных», но и других фирм (SIEMENS, Allen Bradley, GE-Fanuc), а также компактными размерами. Внешний вид сенсорного монитора NT21 с лицевой и тыльной сторон представлен на рис. 4.1. В табл. 4.1 представлено описание основных элементов сенсорного монитора.

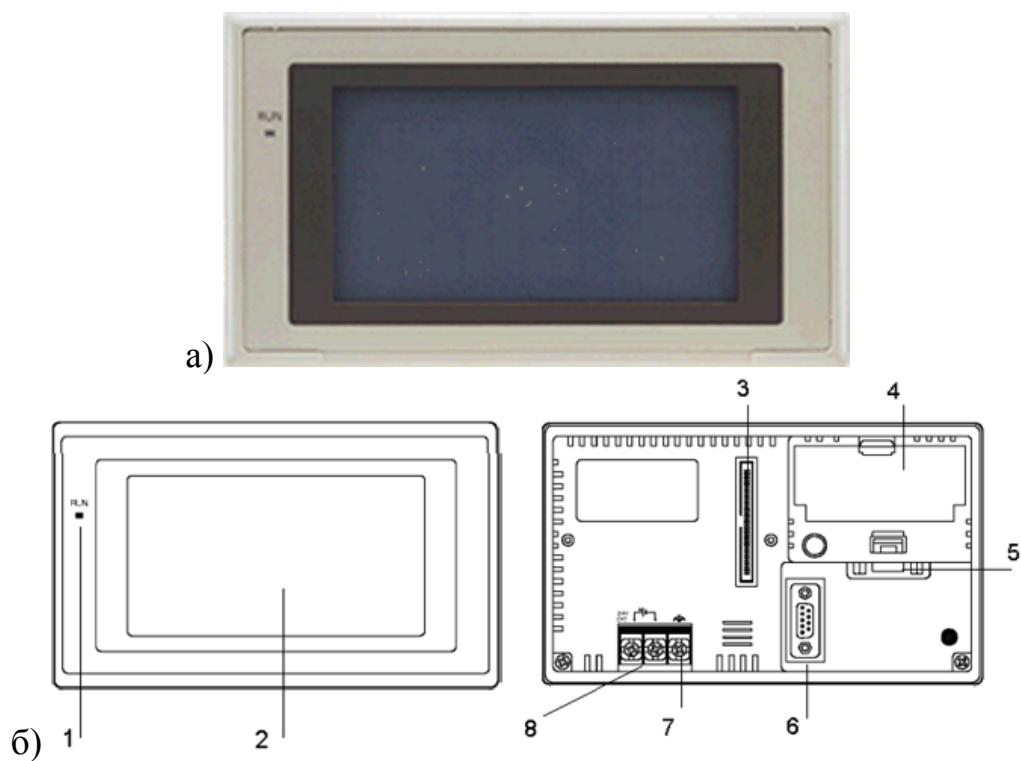


Рис. 4.1. Сенсорный монитор OMRON NT21:  
а) – внешний вид, б) – лицевая и тыльная сторона сенсорного монитора.

Таблица 4.1

№ п/п	Объект	Описание	
1	Индикатор рабочего состояния	Зелёный	В рабочем режиме
		Оранжевый или красный	При разрядке батареи
2	Дисплей	Рабочая поверхность для отображения и ввода информации (сенсорная панель)	
3	Разъём связи с модулем памяти	Используется для подключения дополнительного модуля памяти	
4	Отсек для батареи	В отсеке может находиться батарея	
5	Последовательный порт В	Используется для связи по протоколам RS-232C (связь с компьютером, программой Support Tool или адаптером связи NS-AL002)	
6	Последовательный порт А	Используется для связи по протоколу RS-232 C (связь с компьютером, программой Support Tool или устройством считывания штрихкода )	
7	Клемма защитного заземления	Подключение заземляющего провода	
8	Клеммы питания	Подключение питания	

Технические характеристики сенсорного монитора NT21 представлены в табл. 4.2.

Таблица 4.2

Характеристика		Значение	
Напряжение		DC, 24 В (колебания $\pm 15\%$ )	
Потребляемая мощность		7 Вт.	
Дисплей	Тип	Монохромный жидкокристаллический STN LCD	
	Эффективный размер	117 x 63	
	Углы обзора	Сверху/снизу	30 <sup>0</sup>
		Справа/слева	30 <sup>0</sup>
	Разрешение, пикселей	260 x 140	
	Срок службы	50,000 часов минимум	
Сенсорная панель	Тип	Резисторная матрица	
	Количество ключей	13 x 7	
Связь с внешними устройствами		последовательный порт А (9 конт.), последовательный порт В (9 конт.)	
Протоколы связи		Host Link, NT Link (1:1), NT Link (1:N), High-speed NT Link (1:N), Memory Link	
Поддерживаемые языки		Английский	
Flash-память		512 KB	
Количество пользовательских экранов		3999	

В табл. 4.3 представлены основные функции сенсорного монитора OMRON NT21.

Таблица 4.3

Функция	Описание
<b>Функции отображения информации</b>	
Символьный вывод	Отображаются различные символы, которым можно задавать различные атрибуты (мигание, цвет, размер, фон).
Вывод графических примитивов	Отображаются различные графические примитивы, также можно задавать различные атрибуты (мигание, толщина линии, размер, заливка).
Вывод данных из памяти	Отображается содержимое символьной и числовой таблицы памяти. Содержимое таблиц может выводиться из ПК.
Отображение индикаторов	Отображаются различные виды индикаторов с использованием числовой таблицы памяти
Отображение ламп	Отображаются лампы с заданными пользователем атрибутами (состояние ламп контролируется ПК).

Вывод аварийного листа событий	Тревожные сообщения автоматически отображаются в листе событий, также высвечивается время и номер сообщения
<b>Функции вывода информации</b>	
Зуммер	Обеспечивает звуковую сигнализацию при нажатии на сенсорные кнопки или во время отображения каких-либо экранов
<b>Функции ввода информации</b>	
Ввод с помощью сенсорных кнопок	Возможен ввод данных нажатием на сенсорные кнопки монитора
Всплывающие окна	Можно вызвать всплывающее окно одним нажатием на сенсорную кнопку на текущем экране
Цифровые /символьные строки	Можно создать цифровые и символьные клавиши. Вводимые данные будут записываться в цифровую /символьную строку и отображаться на дисплее, а также в ПК.
Ввод со считывателя штрихкодов	Данные со считывателя штрихкодов можно ввести в поле ввода символьной строки
<b>Функции связи</b>	
Несколько методов связи	Связь с внешними устройствами возможна с использованием пяти различных протоколов
<b>Функции операционной системы сенсорного монитора</b>	
Системное меню	Параметры монитора можно устанавливать в системных меню, высвечиваемых на экране
Функция создания экранов	С помощью программы NT Support Tool можно создавать экраны и пересылать их в память сенсорного монитора
Функция восстановления системы	Содержимое таблиц памяти и состояний записывается перед выключением монитора и восстанавливается при его включении
Функция «хранитель экрана»	Увеличивает срок работы подсветки, отключая её в моменты, когда сенсорный монитор не используется, но включён
Часы	Время отображается на дисплее синхронно с внешними устройствами (ПК)
Контрольный монитор	Когда монитор соединён с ПК по протоколам NT Link (1:1) и NT Link (1:N), он может быть использован для смены режима работы ПК, чтения сообщений об ошибках и других целей
Установка операционной системы	Операционная система может быть установлена с помощью программы NT Support Tool или внешнего модуля памяти
Функция записи графиков	Изменения в содержимом цифровых таблиц памяти можно отобразить в виде графика, который может быть записан (запись также может осуществляться в фоновом режиме).

Математические функции	Возможны арифметические, логические, битовые и другие операции, когда есть математические таблицы в данных экрана
------------------------	---

#### 4.4. Операционная система сенсорного монитора NT21

Как и большинство сенсорных панелей фирмы OMRON данный сенсорный монитор обладает собственной операционной системой. Монитор NT21 может работать в трёх режимах. В табл. 4.4 представлено краткое описание режимов работы сенсорного монитора.

Таблица 4.4

Режим	Описание
Рабочий режим	В этом режиме монитор контролируется ПК, возможны различные дисплейные операции и операции ввода /вывода
Установка системы	Установка операционной системы монитора с помощью программы NT Support Tool или внешнего модуля памяти
Режим системного меню	Используется для изменения системных параметров сенсорного монитора или данных рабочей программы (данных экранов)

При включении питания сенсорный монитор автоматически включается в рабочий режим при наличии готового проекта в его памяти. Для перехода в системное меню из рабочего режима необходимо нажать на два любых угла экрана сенсорного монитора, а для перехода в режим установки системы из рабочего режима необходимо нажать на верхний левый угол при включении питания сенсорного монитора (рис. 4.2, рис. 4.3).

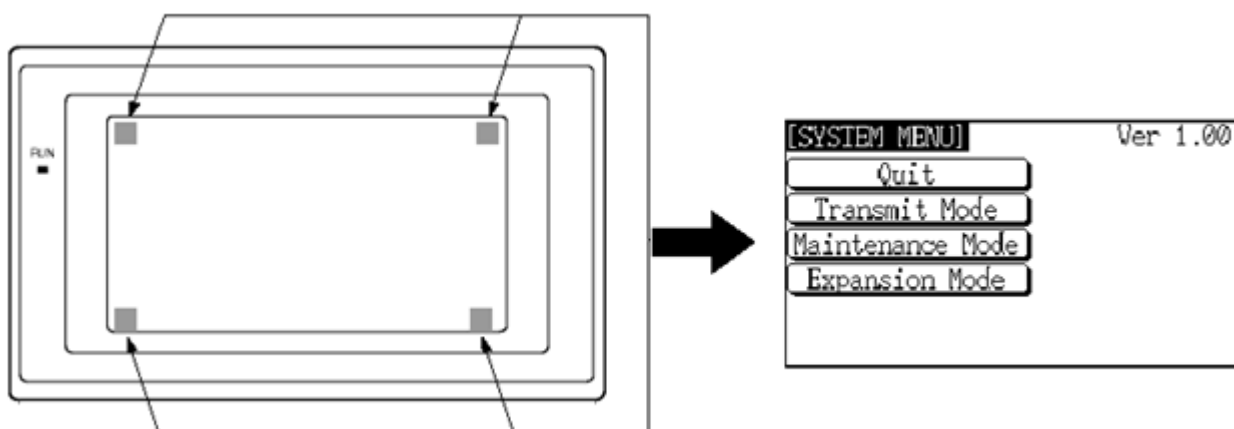


Рис. 4.2. Переход в системное меню и экран системного меню

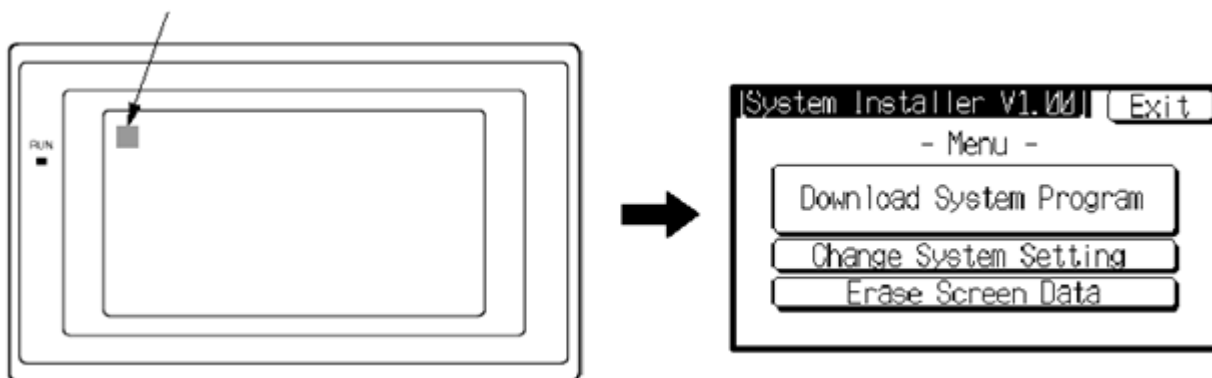


Рис. 4.3. Переход в режим установки системы и экран установки системы

Перемещение по пунктам меню в обоих режимах производится нажатием на название требуемого пункта, отображаемое на дисплее сенсорного монитора. Так как в лабораторной работе не требуется менять операционную систему монитора, то далее работа в этом режиме рассматриваться не будет. В системном меню особое внимание следует уделить меню передачи данных (Transmit Mode) и меню настройки параметров (Maintenance Mode). Первое меню используется для пересылки данных из ПЭВМ в монитор или обратно. Второе меню необходимо для проверки состояния сенсорного монитора, сброса всех параметров, проверки записанных данных, выбора требуемых параметров связи с внешними устройствами, а также ряда других установок. Режим Expansion mode используется для работы с контроллером (программирование, наблюдение и другое)

#### 4.5. Программирование сенсорного монитора OMRON NT21

Сенсорный монитор OMRON NT21 содержит различную виртуальную коммутационную аппаратуру, а также различные средства для ввода, обработки и вывода на дисплей монитора информации. Для обеспечения всех этих возможностей сенсорный монитор обладает flash-памятью, которая разбита на таблицы, в которых размещаются все необходимые данные

Данный сенсорный монитор обладает широкими программными возможностями, описание которых приведено в табл. 4.5. Также приведен объём памяти, занимаемый конкретным объектом.

Таблица 4.5

Элемент	Описание
Фиксированные символы (Character)	Не занимает таблиц памяти. Отображаются фиксированные символьные данные до 65535 статичных элементов на экране).
Фиксированные символьные строки (Character string)	Занимает 1 символьную таблицу памяти + 1 косвенную таблицу памяти. Отображаются фиксированные символьные строки. До 256 элементов на экране, до 32 байт в строке.

Числовые строки (Numeral displays)		Занимает 1 числовую таблицу памяти + 1 косвенную таблицу памяти. Отображаются фиксированные числовые строки. До 256 элементов на экране, до 10 знаков в строке.
Гистограммы (Bar graph)		Занимает 1 числовую таблицу памяти. Отображается текущее значение заданного слова ПК. До 50 элементов на экране. Возможно отображение текущего значения в процентах и в абсолютной величине.
Фиксированные маркеры (Mark displays)		Не занимает таблиц памяти. До 65535 статичных элементов на экране.
Лампы (Lamps)		Занимает 1 бит ПК. Загорается при наличии 1 по заданному адресу ПК. До 256 элементов на экране.
Аналоговые индикаторы (Analogue meter)		Занимает 1 числовую таблицу памяти. Отображается текущее значение заданного слова ПК. До 50 элементов на экране. Возможно отображение текущего значения в процентах и в абсолютной величине.
Кнопки (Touch switches)	Без фиксации (Momentary)	Замыкает контакт при нажатии сенсорной кнопки.
	С фиксацией (Alternative)	Замыкает контакт при первом нажатии сенсорной кнопки, размыкает его – при повторном нажатии .
	Установки (Set)	Устанавливает 1 по заданному адресу.
	Сброса (Reset)	Устанавливает 0 по заданному адресу.
	Переключения экранов (Screen switch)	Переключает текущий экран на заданный.
	Вызова окон	Включает окна в текущем экране.
	Ввода данных (Input key)	Применяется для ввода данных с сенсорной клавиатуры.
	Курсорные (Cursor move)	Используются для перемещения по пунктам расширенного меню.
Тренд - графики (Trend graph)		Количество занимаемых таблиц памяти зависит от количества линий. Отображается текущее значение заданного слова ПК. Максимум 1 кадр на экране, 50 графиков в проекте (без записи в память), 8 графиков в проекте (с записью в память).
Пунктирные графики (Broken line graph)		Количество занимаемых ячеек памяти зависит от количества линий. Отображается текущее значение заданного слова ПК. Максимум 1 кадр на экране , 256 графиков в кадре, до 260 точек в графике.
Картинки (Image data)		Не занимает таблиц памяти. Различные статичные изображения. До 256 картинок на экране.
Статическая графика (Graphic displays)		Не занимает таблиц памяти. Отображаются различные статичные элементы (ломаная линия, прямоугольник, многоугольник, круг, дуга, сектор). До 65535 статичных элементов на экране.
Библиотеки (Library data)		Не занимает таблиц памяти. Различные статичные изображения. До 256 картинок на экране.

Числовой ввод (Numeral inputs)	Клавиши (Numeric key)	Для ввода числовых данных. До 256 элементов на экране.
	Дисковый (Thumbwheel)	Занимает 1 числовую таблицу памяти. Для ввода числовых данных. До 26 элементов.
Символьные строки ввода (Character string inputs)		Занимает 1 символьную таблицу памяти. Отображается вводимая информация с сенсорной клавиатуры, до 256 элементов.
Листы происшествий (Alarm lists)		Не занимает таблиц памяти. До 4 групп на экране.
История происшествий (Alarm histories)		
Часы (Clock display)		Отображается время с помощью числового дисплея.
Обычные экраны (Normal screen)		В проекте может быть до 3999 экранов, создаваемых пользователем.
Связанные экраны (Overlapping screens)		Родительский экран может состоять из нескольких дочерних (до 8) экранов.
Окна (Window screens)		Может включаться до 3 окон одновременно на одном экране.
Экраны истории дисплея (Display history screens)		Сортировка по событиям (до 1024 экранов), сортировка по частоте возникновения (до 255 раз).
Максимальное число экранов		3999.
Системные экраны		0: нет экрана ; 9000: экран «Initializing system»; 9001: экран списка событий по времени; 9002: экран списка событий по повторяемости; 9200: экран консоли для программирования; 9999: экран возвращения к предыдущему.

Так как сенсорный монитор содержит виртуальную коммутационную и сигнальную аппаратуру, то он может принимать и выдавать информацию. При совместной работе с программируемым контроллером сенсорный монитор находится в режиме прямого доступа в память ПК.

Допустим, необходимо реализовать виртуальный пульт, содержащий одну кнопку и одну сигнальную лампу. Во-первых, необходимо определиться с ячейками памяти в программируемом контроллере, куда будут записываться состояния кнопок и откуда будет взят сигнал на лампу. Обмен между сенсорным монитором и ПК идет в основном через промежуточные переменные (внутренние выходы). Пусть в основной программе ПК OMRON CPM-30CDR использованы промежуточные переменные IR019.00 и IR019.01. Кнопке будет соответствовать адрес IR019.00, а лампе адрес IR019.01. При нажатии на сенсорную кнопку на экране монитора в ПК по заданному адресу установится 1 и, в зависимости от выбранного типа кнопки, при отпускании кнопки 1 по заданному адресу сбросится (Momentary), установится 1 до следующего нажатия (Alternative) или установится в 1 постоянно. При появлении 1 по адресу IR019.01 будет выдана команда на включение виртуальной лампы находящейся на дисплее сенсорного монитора.

Программирование сенсорного монитора сводится к следующему:



– из имеющегося набора графических инструментов на экране монитора формируется требуемое расположение, размер, цвет и функции и т. п. элементов виртуального пульта. Также есть возможность расположить на пульте необходимые надписи, рисунки;

– всем коммутационным и сигнальным элементам виртуального пульта присваиваются адреса, находящиеся в памяти ПК;

– проводится компиляция подготовленного проекта;

– проект по последовательному интерфейсу записывается в сенсорный монитор.

#### 4.6. Работа с программой NT-series Support Tool

Для программирования сенсорных панелей оператора серии NT фирмы OMRON используется специальная программа NT-series Support Tool. Запуск NT Support Tool производится через соответствующий ярлык на рабочем столе или в меню «Пуск» персонального компьютера (Пуск > Программы > OMRON > NTST > NT-series Support Tool).

После запуска программы перед пользователем открывается пустое рабочее поле (рис. 4.4а). Для дальнейшей работы необходимо открыть ранее сохраненный или создать новый проект. Для открытия существующего проекта необходимо выбрать в меню File пункт Open. После этого появится список существующих проектов (рис. 4.4б), из которого надо выделить курсором нужный и нажать кнопку ОК или отменить действие кнопкой Cancel. После этих операций должно появиться два окна проекта. В левом окне расположено дерево проекта, в котором можно выбирать необходимые объекты для просмотра или редактирования. В правом окне расположен перечень объектов, расположенных в папке, выбранной в левом окне (рис. 4.5).

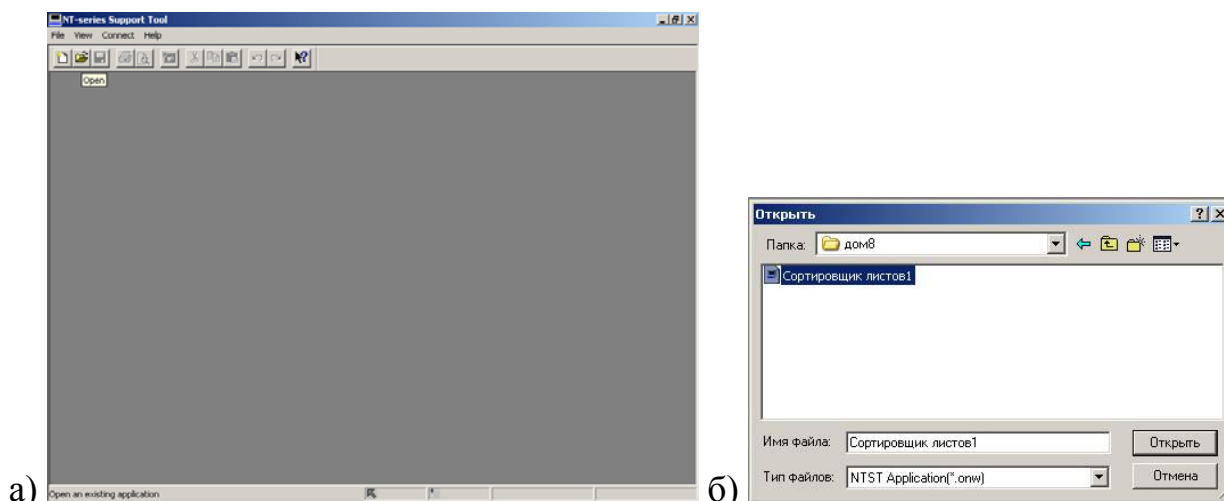


Рис. 4.4. Окна программы при запуске приложения NT-series Support Tool и выборе файла проекта

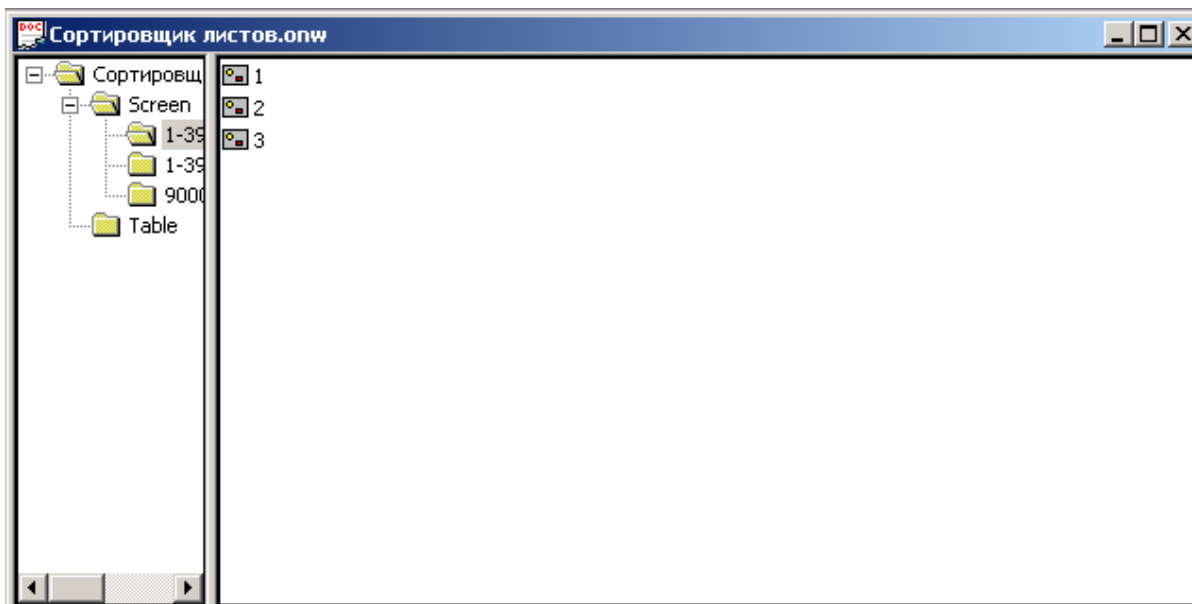


Рис. 4.5. Рабочее окно программы NT-series Support Tool

Для создания нового проекта нужно выбрать в меню File пункт New. В появившемся окне в разделе PT Type (рис. 4.6а) в строке PT model выбрать тип панели оператора NT21, в строке PLC Vendor выбрать OMRON, а также, если требуется написать комментарий в соответствующей строке. В разделе Control/Notify Area (рис. 4.6б) в соответствующих полях ввести начальные адреса контрольной области памяти (Control Area), информационной области памяти (Notify Area) и области Window Control Area. При этом длина области Control Area составляет 5 слов, области Notify Area – 2 слова, а области Window Control Area – 9 слов. После этого необходимо подтвердить правильность введённых данных нажатием кнопки ОК.

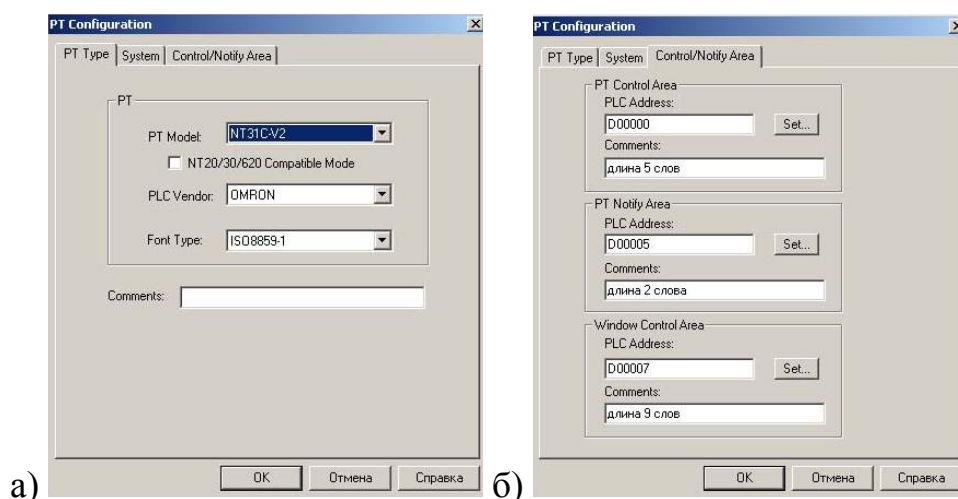


Рис. 4.6. Окна выбора параметров проекта

В появившемся далее окне «Clock Address (Адресация времени)» рис. 4.7 можно ничего не устанавливать и просто нажать ОК.

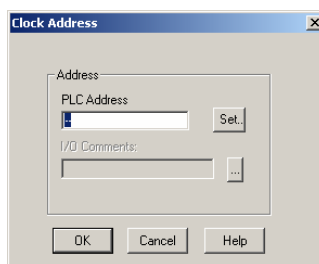


Рис. 4.7. Окно установки адреса часов

После проведения этих операций перед пользователем откроется проект, в котором по умолчанию для редактирования будет готов первый экран (рис. 4.8). для увеличения масштаба экрана необходимо выбрать в меню View пункт Zoom и установить масштаб 400%.

Всего в проекте может быть до 3999 экранов, созданных пользователем. Каждый экран может содержать до различных 256 объектов.

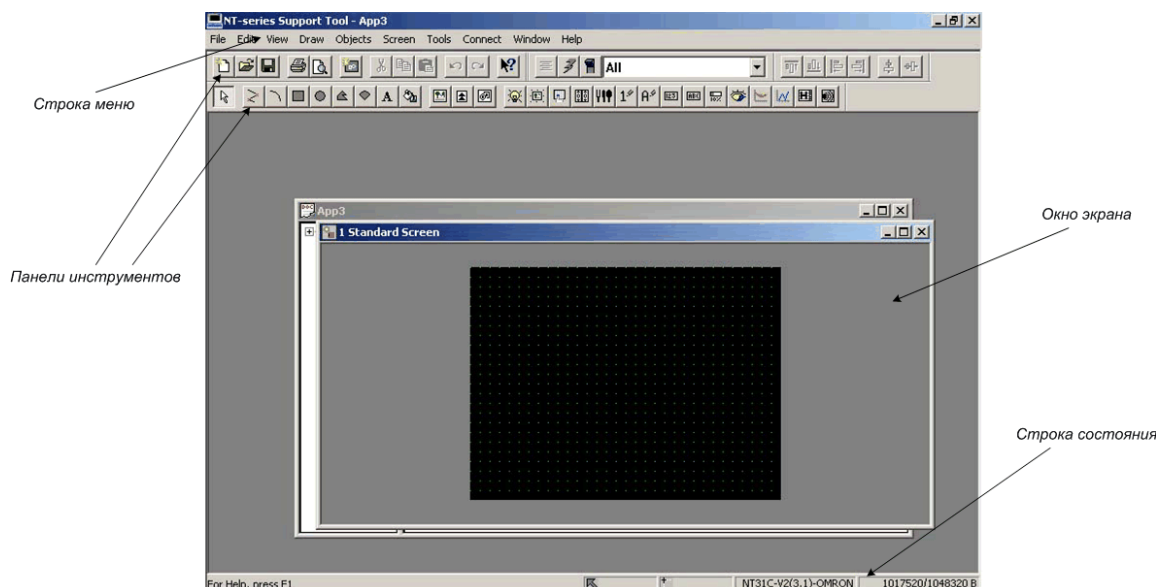


Рис. 4.8. Внешний вид проекта NT-series Support Tool

При работе с проектом возможно редактирование одновременно только одного экрана. Если экран уже существует, то его можно выбрать в правом окне программы. При этом в левом окне программы должен быть выбран необходимый тип экрана.

Для создания нового экрана необходимо в меню Screen выбрать пункт New. При этом появится окно, в котором нужно выбрать тип экрана (Обычный – Standard, Главный – Parent, Окно – Window, Экран связи с ПК – Host Connect). После этого указать номер экрана и, если необходимо, ввести комментарий. При работе с экраном возможны следующие действия: удаление текущего экрана (меню Screen>пункт Delete), изменение масштаба изображения экрана (меню View>пункт Zoom), наложение и отключение сетки (меню Screen>пункт Grid), а также изменение свойств текущего экрана (меню Screen>пункт Properties).

При работе с выбранным экраном активизируется панель инструментов, расположенная над проектом (рис. 4.9).



Рис. 4.9. Панель инструментов NT-series Support Tool

В табл. 4.6 дано краткое описание элементов этой панели.

Таблица 4.6

№ п\п	Описание	№ п\п	Описание
1	Указатель	15	Сенсорная кнопка
2	Ломаная линия Дуга	16	Дисковый числовой ввод
3	Дуга	17	Способ ввода
4	Прямоугольник	18	Числовой ввод
5	Круг	19	Символьный ввод
6	Многогранник	20	Числовая строка
7	Сектор	21	Символьная строка
8	Текст	22	Гистограмма
9	Заливка	23	Аналоговый индикатор
10	Рисунок	24	Тренд - график
11	Маркер	25	Ломанный график
12	Библиотечный объект	26	История предупреждений
13	Стандартная лампа	27	Лист предупреждений
14	Фигурная лампа		

С её помощью (так же как и меню Objects) можно редактировать количество, размер, положение и другие параметры объектов, расположенных на экране. В меню Objects расположено восемь основных разделов, на которые рассортированы все элементы. Для редактирования экрана (установки, модификации или удаления элементов экрана) выполняется следующая последовательность действий:

- в меню Objects или на панели инструментов выбрать необходимый элемент (при этом выбранный элемент как бы прилипает к курсору);
- навести курсор на нужное место экрана (курсор при этом имеет вид крестика);
- нажать левую кнопку мыши и, удерживая её, установить необходимый размер объекта;
- отпустить левую кнопку мыши (при этом появится окно со свойствами выбранного объекта);
- настроить все требуемые параметры редактируемого объекта, после чего нажать кнопку ОК.

В дальнейшем можно изменять размер объекта. Для этого необходимо сперва выделить нужный объект, щелкнув на него левой кнопкой мыши. При этом по

краям объекта появляются маркеры, при наведении на которые курсор изменяет свой вид на стрелки. Нажав при этом левую клавишу мыши и удерживая ее, можно изменять размер объекта. При достижении желаемого размера клавишу следует отпустить.

Если навести курсор на свободный участок объекта и удерживать левую кнопку мыши, то курсор поменяет вид на перекрестье стрелок. В этом режиме можно перемещать объект в пределах картинки. По окончании перемещения кнопку мыши следует отпустить.

Для проведения операций над выделенным объектом нужно нажать правую клавишу мыши или в меню Edit и Draw на панели инструментов. В открывшемся при этом контекстном меню доступны следующие операции:

- вырезать (Cut) объект в буфер памяти;
- копировать (Copy) объект в буфер памяти;
- вставить (Paste) объект из буфера памяти;
- удалить (Delete) выделенный объект;
- вынести вперед (Bring to Front) объект;
- удалить на задний план (Send to Back) объект;
- группировать (Group) несколько выделенных объектов;
- разгруппировать (Ungroup);
- выравнивание (Align) нескольких выделенных объектов.

После установки положения и размера необходимо установить параметры объекта. Для первоначальной установки свойств объекта надо при установке отпустить левую кнопку мыши. Для изменения свойств необходимо дважды быстро щелкнуть на выбранном объекте или в меню Draw выбрать пункт Properties, при этом появляется окно, в котором можно выставить большое количество параметров объекта. Большинству элементов присущи общие параметры, приведённые в табл. 4.7.


Таблица 4.7

Параметр	Описание
Рамка (Frame)	Выбирается с рамкой, или без рамки.
Форма объекта (Shape)	Выбирается одна из предлагаемых форм
Тип линии (Line type)	Выбирается один из 4 типов линий
Тип изображения (Attribute)	Выбирается тип отображения объекта: Обычный, инверсный, мигающий, инверсный мигающий
Основной цвет объекта (Foreground)	Выбирается один из 2 цветов
Фоновый цвет объекта (Background)	Выбирается один из 2 цветов
Адрес (PLC Bit Address)	Устанавливается необходимый адрес контроллера
Размер шрифта (Scale)	Устанавливается размер шрифта
Сглаживание (Smoothing)	Выбирается режим сглаживания текста
Тип дисплея (Display type)	Выбирается формат отображения данных на цифровом дисплее (Десятичный или шестнадцатеричный)
Формат числа (Format)	Целое или с десятичной точкой

Для большинства элементов виртуального пульта (кнопки, лампы, графики) необходимо указать адрес ячейки памяти программируемого контроллера, с которым данный элемент будет связан в ходе работы. Для кнопки, во-первых, необходимо выбрать её тип (Notify bit, Switch screen, Cursor move, Input key, Copy Setting, Print screen). Во-вторых, в зависимости от выбранного типа кнопки, установить адрес (Notify bit), по которому будет устанавливаться 1 при нажатии на виртуальную кнопку, установить номер экрана (Switch screen), на который будет происходить переключение при нажатии, установить параметры перемещения (Cursor move), установить выполняемую функцию (Print screen), установить параметры ввода (Input key), или параметры копирования (Copy Setting). Далее необходимо установить параметры включения кнопки (адрес, цвет), то есть её можно использовать как лампу, а также установить надпись на кнопке.

Для лампы необходимо выбрать следующие параметры: адрес программируемого контроллера, цвет лампы в выключенном и включенном состояниях, надпись на лампе, форму лампы.

После выбора всех параметров необходимо нажать кнопку «Применить» и далее «ОК», или, в случае отмены выбора, – кнопку «Отмена».

Для ввода нового экрана нужно в меню Screen выбрать пункт New или же на панели инструментов нажать значок . При этом появляется окно выбора экрана (рис. 4.10а), в котором необходимо выбрать тип экрана. После этого появится следующее окно (рис. 4.10б), в котором будет предложено ввести номер экрана, а также комментарий, если это стандартный экран или всплывающее окно. Если же это родительский экран, то необходимо указать номер родительского экрана, а также номера дочерних экранов (рис. 4.11).

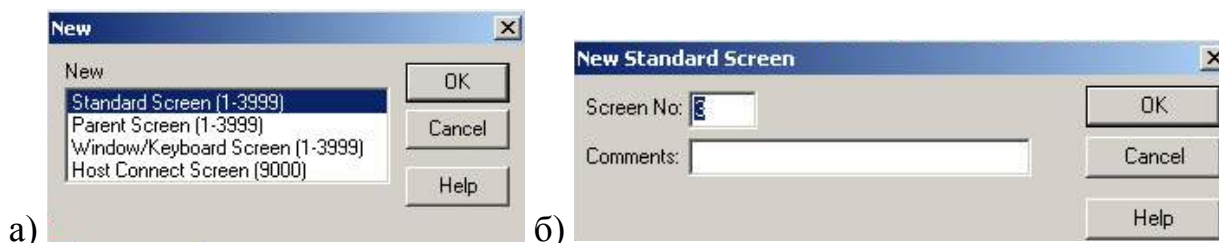


Рис. 4.10. Окна выбора экранов

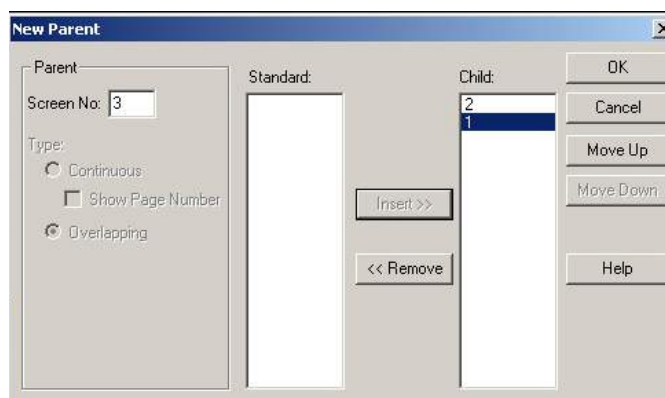


Рис. 4.11. Окно выбора родительского и дочерних экранов

## 4.7. Примеры программирования сенсорного монитора NT21

*Пример 1.* Программирование кнопки с ярлыком установки единицы по адресу IR019.00.

Текст ярлыка 0 – «OFF» шрифтом 2x2, расположенный по центру. Цвет букв – белый, цвет фона – чёрный.

Текст ярлыка 1 – «ON» шрифтом 2x2, расположенный по центру. Цвет букв – чёрный, цвет фона – белый.

Программирование ведётся в следующей последовательности:

- выбрать пункт Objects >Touch Switch;
- установить курсор в нужном месте, нажать и отпустить левую кнопку мыши;
- в закладке General в поле OFF выбрать чёрный цвет, в поле ON – белый;
- в закладке Settings в поле Function выбрать Notify Bit, в поле Action Type – Set, в поле PLC Bit Adress установить 0001900;
- в закладке Label установить галочку напротив Label и в поле Label Type выбрать On Off Static;
- нажать Edit, в поле Off Label ввести OFF, в поле On Label – ON, в поле Scale – 2x2, в поле OFF – белый цвет, в поле ON – чёрный и нажать кнопку ОК;
- для подтверждения выбора параметров нажать ОК.

*Пример 2.* Создание на экране кнопок перехода на требуемый экран.

Программирование ведётся в следующей последовательности:

- выбрать пункт Objects >Touch Switch;
- установить курсор в нужном месте, нажать и отпустить левую кнопку мыши;
- в закладке Settings в поле Function выбрать Switch Screen, в поле Screen No. – требуемый номер экрана;
- для подтверждения выбора параметров нажать ОК.

Для выбора других параметров (цвет кнопки, ярлык и другое) необходимо воспользоваться примером 1.

*Пример 3.* Создание статического текста «Picture #1» шрифтом 2x2 черным цветом на белом фоне.

Программирование ведётся в следующей последовательности:

- выбрать пункт Objects >Fixed Displays >Text;
- установить курсор в нужном месте, нажать и отпустить левую кнопку мыши;
- в поле Description ввести текст «Picture #1», в поле Scale – 2x2, для настройки цвета и фона текста в поле Foreground выбрать черный цвет, в поле Background – белый;
- для подтверждения выбора параметров нажать ОК.

*Пример 4.* Создание стандартной лампы круглой формы, сигнализирующей о состоянии бита IR010.02.

В выключенном состоянии лампа горит белым цветом, во включенном – мигает черным. На лампе – надпись PUSK шрифтом 2x2 черного цвета. Программирование ведётся в следующей последовательности:

- выбрать пункт Objects >Lamps >Standart;
- установить курсор в нужном месте, нажать и отпустить левую кнопку мыши;
- в закладке General в поле Shape (форма) выбрать Circle, в поле OFF – белый цвет, в поле ON – черный цвет;
- в закладке Light Function в поле PLC Bit Adress установить 0001002, в поле On Type – Flash;
- в закладке Label установить галочку напротив Label и в поле Label Type выбрать Static;
- нажать Edit, в поле Label ввести PUSK, в поле Scale – 2x2, в поле OFF – черный цвет, в поле ON – также черный и нажать кнопку ОК;
- для подтверждения выбора параметров нажать ОК.

*Пример 5.* Создание цифрового дисплея, показывающего содержимое слова IR010 в десятичном коде (4 разряда до точки, 2 – после).

Цифры размером 2x2 со сглаживанием, цвет цифр – черный на белом фоне.

Программирование ведётся в следующей последовательности:

- выбрать пункт Objects >Numeral Display;
- установить курсор в нужном месте, нажать и отпустить левую кнопку мыши;
- в закладке General в поле Scale выбрать 2x2, напротив Smoothing поставить галочку, для настройки цвета и фона в поле Foreground выбрать черный цвет, в поле Background – белый;
- в закладке Settings в поле Display Type поставить точку напротив Decimal, в полях Integer и Decimal выставить соответственно 4 и 2;
- для выбора слова, значение которого будет считываться, нажать кнопку в поле Reference, при этом появится окно настройки (рис. 4.12);
- в графе PLC Address указать слово, значение которого будет считываться, и нажать кнопку Close;
- для подтверждения выбора параметров нажать ОК.



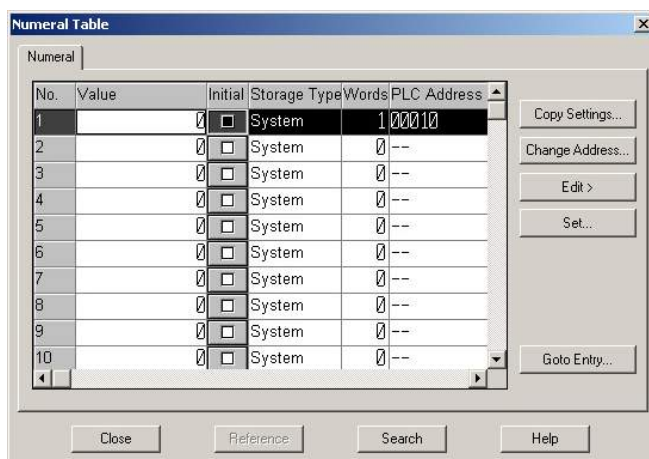


Рис. 4.12. Окно настройки таблиц данных

#### 4.8. Загрузка проекта в сенсорный монитор OMRON NT21

Перед загрузкой проекта в память сенсорного монитора компиляции проекта не требуется и программой NT-series Support Tool эта процедура не предусматривается. Поэтому перед загрузкой необходимо непосредственно настроить параметры связи с сенсорным монитором. Для этого в меню Connect нужно выбрать пункт Comms. Setting. При этом появляется окно настройки параметров связи (рис. 4.13). В окне параметров связи необходимо выбрать COM3, 115 Kbps и для подтверждения этого нажать ОК.

Также необходимо удостовериться в правильности подключения шнура связи к порту панели оператора и порту компьютера COM3. Далее нужно подать питание на сенсорную панель. Через некоторое время должен появиться инициализирующий экран. Для перехода в системное меню необходимо нажать два любых угла экрана монитора. При этом появится системное меню (рис. 4.3а).

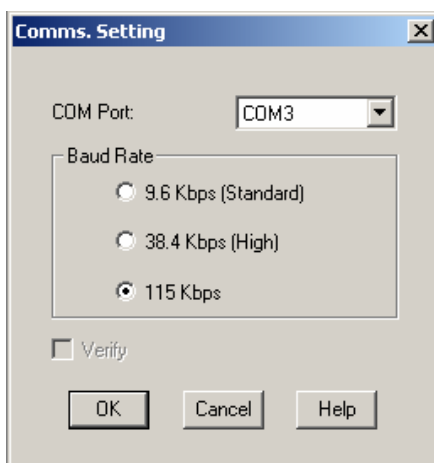


Рис. 4.13.

Далее необходимо выбрать пункт Transmit Mode. Только в этом режиме обеспечивается связь сенсорного монитора с персональным компьютером. Во

время обмена информацией между монитором и ПЭВМ запрещается выходить из данного режима.

Для начала загрузки нужно в программе NT-series Support Tool выбрать строку Connect >Download (NT-series Support Tool >PT)>Application (рис. 4.14).

При правильном подключении и параметрах связи после этого начинается загрузка проекта в память сенсорного монитора, о чём сигнализирует следующее окно, представленное на рис. 4.15а.

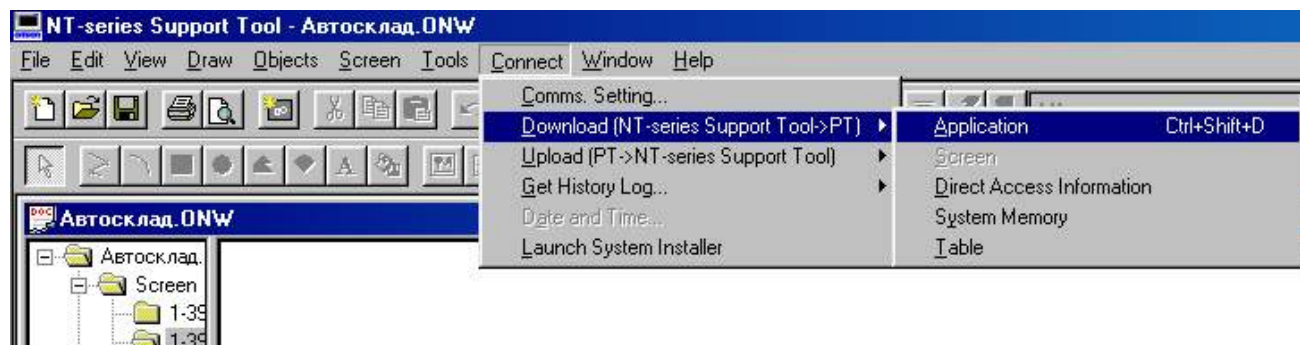


Рис. 4.14.

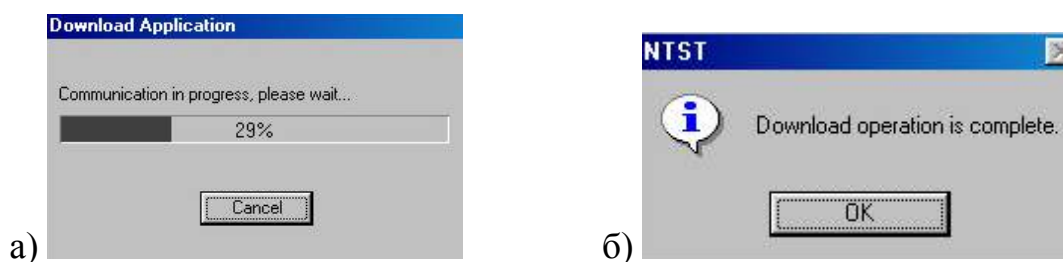


Рис. 4.15. Отображение процесса записи проекта в память монитора

По окончании загрузки проекта в сенсорный монитор (рис. 4.15б) следует нажать кнопку «ОК».

При неправильном подключении или параметрах связи на экране ПЭВМ появится окно с сообщением об ошибке (рис. 4.16а), в котором предлагается повторить попытку связи или отказаться от этой операции. В том случае, когда номер COM - порта ПЭВМ, выбранного в настройках не соответствует действительному соединению СМ и ПЭВМ на экране монитора появится окно (рис. 4.16б), в котором будет предложено сменить номер COM - порта.

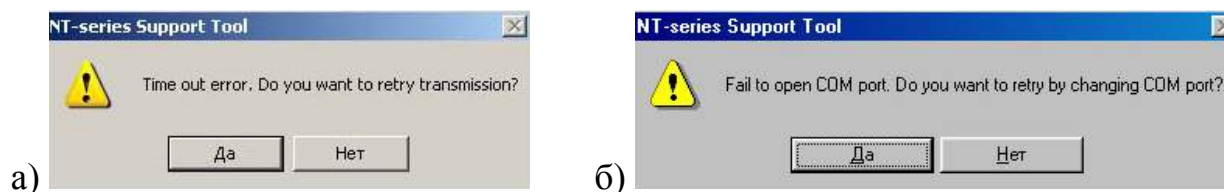


Рис. 4.16. Окна ошибок при записи проекта в память монитора

В качестве дополнительной функции программы NT-series Support Tool стоит отметить печать проекта. Для этого необходимо нажать кнопку на панели инст-

рументов, после чего появится окно (рис. 4.17а), в котором можно выбрать данные, отправляемые на печать. В этом окне в поле Report Type необходимо выбрать пункт Screen Image, при этом окно меняет вид (рис. 4.17б). Далее необходимо в поле Include поставить галочки напротив тех данных, которые необходимо отправить на печать. После этого должно появиться стандартное окно настройки параметров принтера, где пользователь самостоятельно выбирает необходимые свойства и отправляет данные на печать.

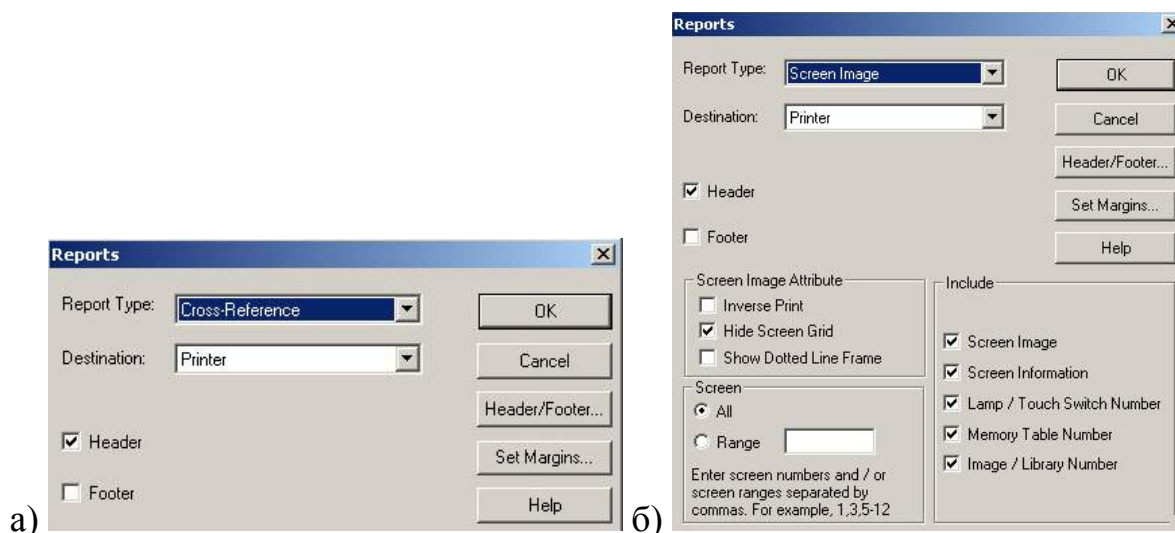


Рис. 4.17. Настройка параметров печати

## 4.9. Выполнение лабораторной работы

Данная лабораторная работа должна проводиться после выполнения лабораторной работы по изучению программируемого контроллера СРМ2А, так как предусматривается использование сенсорного монитора совместно с программируемым контроллером для управления виртуальным объектом. Описание виртуальных объектов представлено в Приложении Б. Так как подготовить и отладить программу автоматизации объекта в полном объеме довольно трудоемкая работа, то в каждом варианте объекта предусмотрены подварианты. Именно эти подварианты и будут использоваться в данной работе, и именно для них будут проектироваться виртуальные пульта управления.

В табл. 4.8 представлены варианты задания для программирования сенсорного монитора.

Как указывалось в программе работ (раздел 4.2), необходимо реализовать как минимум две картинки программируемого пульта, обеспечивающего управление объектом, как в автоматическом, так и в ручном режимах управления.

При разработке пульта управления в основном используются следующие типы программируемых объектов: переключатель или кнопки выбора режима работы «Авт-Руч», кнопки с фиксацией и без фиксации, кнопки перехода на другую картинку, индикаторы.

На всех программируемых объектах необходимо предусмотреть надписи или рисунки, поясняющие функциональное назначение объекта, например: Авт., Руч., Пуск, Сброс, Стоп, Вверх, Вниз, Влево, Вправо и т.п.

Таблица 4.8

Номер бригады	1			2			3			4			5			6			7			8					
Член бригады	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3			
Виртуальный объект	Вар. 1			Вар. 2			Вар. 3			Вар. 4			Вар. 5			Вар. 6			Вар. 7			Вар. 8					
Подвариант объекта	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3

Перед работой с виртуальными объектами необходимо проверить подключение программируемого контроллера к порту СОМ1 ПЭВМ, подключение встроенной в контроллерный модуль платы «Устройство преобразования кодов» (УПК) к порту СОМ2, а панели оператора – к СОМ3. Кроме того, необходимо проверить подключение сенсорного монитора к программируемому контроллеру. Если нет связи между ПЛК и СМ, то на лицевой панели ПЛК не горит светодиод СОММ (при отсутствии связи ПЛК с ПЭВМ), а на экране панели светится надпись «System Initializing».

«Устройство преобразования кодов» преобразует сигналы управления, формируемые программируемым контроллером, в последовательный код и через СОМ-port компьютера выдает их на исполнительные органы виртуального объекта. Он же преобразует сигналы датчиков виртуального объекта, поступающие через СОМ-port, из последовательного кода в параллельный и подает их на входы контроллера. В табл. 4.9 представлено соответствие входов и выходов контроллера сигналам с датчиков виртуального объекта и командам на виртуальный объект.

При работе с программой виртуальных объектов необходимо выполнить следующее:

- запустить с рабочего стола ярлык РК&МК;
- в меню «Настройки» выбрать пункт «Настройки СОМ порта», где если даже уже установлен порт СОМ2, повторить установку порта СОМ2;
- в «МЕНЮ» выбрать пункт «Выбор виртуального объекта», где задать требуемый виртуальный объект;
- для начала работы выбранного виртуального объекта от программируемого контроллера установить переключатель в нижнем правом углу экрана в положение «От контроллера». При этом светодиод на кнопке УПК модуля программируемого контроллера должен мигать, кроме того, на лицевой части контроллера должны загореться светодиоды, соответствующие определенным входным сигналам.

Таблица 4.9

Адрес		Назначение		Адрес		Назначение	
Входы	IR000.01	X0	Виртуальный объект	Выходы	IR010.00	Y0	Виртуальный объект
	IR000.02	X1			IR010.01	Y1	
	IR000.03	X2			IR010.02	Y2	
	IR000.04	X3			IR010.03	Y3	
	IR000.05	X4			IR010.04	Y4	
	IR000.06	X5			IR010.05	Y5	
	IR000.07	X6			IR010.06	Y6	
	IR000.08	X7			IR010.07	Y7	
	IR000.09	X8			IR011.00	Y8	
	IR000.10	X9			IR011.01	Y9	
	IR000.11	X10			IR011.02	Y10	
	IR001.00	X11			IR011.03	Y11	
	IR001.01	X12					
	IR001.02	X13					
	IR001.03	X14					
	IR001.04	X15					
IR001.05	X16						

Дополнительную информацию по программированию сенсорного монитора и его совместной работы с программируемым контроллером при управлении виртуальными объектами можно получить в описании к работе №3 в учебном пособии «Борисов А.М. Средства автоматизации и управления/ А.М. Борисов, А.С. Нестеров, Челябинск: Изд-во ЮУрГУ, 2007. – 207 с.».

#### 4.10. Требования к отчёту

Отчет по работе должен содержать:

- а) цель работы;
- б) краткое описание виртуального объекта автоматизации, перечень необходимых элементов управления (кнопок, тумблеров, устройств индикации и т.д.);
- в) картинки программируемого пульта управления;
- г) программу для контроллера СРМ2А, использованную при проверке правильности работы разработанного программируемого пульта управления;
- д) методику экспериментальной проверки работоспособности проекта и результаты проверки;
- е) выводы по работе.

#### 4.11. Контрольные вопросы

1. Каково назначение сенсорного монитора?
2. Какие режимы работы есть у сенсорного монитора?

3. Какое максимальное количество картинок может содержать один проект?
4. Какое максимальное количество объектов может содержать одна картинка?
5. Если пульт включает в себя несколько картинок, то, как осуществляется переход от одной картинке к другой?
6. Как обеспечивается адресация элементов программируемого пульта управления?
7. Чем отличается программа ПК для управления одним и тем же объектом при использовании программируемого пульта и при использовании обычного пульта с реальными кнопками, тумблерами и т.п.?

## **5. РАБОТА №5. ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРОВ DL05 И DL06 ПРОГРАММИРУЕМЫХ КОНТРОЛЛЕРОВ ФИРМЫ AUTOMATION DIRECT**

### **5.1. Цель работы**

Ознакомиться с устройством и техническими характеристиками микроконтроллеров DO-05DD и DO-06DD1 соответственно серий DL05 и DL06 контроллеров фирмы Automation Direct. На базе микроконтроллера DO-05DD приобрести навыки программирования задач автоматизации, требующих выполнения логических, временных, счётных и других функций.

### **5.2. Содержание работы**

1. Изучить устройство и технические характеристики микроконтроллеров серий DL05 и DL06, систему команд, языки их программирования.

2. Дома, при подготовке к работе, выполнить синтез системы автоматизации для заданного варианта задачи. Составить программу на языке лестничных диаграмм для ввода в микроконтроллер.

3. В лаборатории:

– пройти тестирование по системе команд и принципу программирования контроллера DO-05DD;

– освоить графическую среду программного обеспечения «DirectSoft32»;

– набрать на ПЭВМ подготовленную дома программу, запустить её на исполнение. Убедиться в правильности функционирования подготовленной дома программы. При необходимости откорректировать программу;

– записать программу в контроллер;

– запустить программу в работу, проверить правильность функционирования системы управления с помощью имитатора внешних устройств, продемонстрировать работу системы преподавателю;

– при успешном выполнении работы преподаватель даёт задание на подготовку и проверку той же программы на языке стадий или с использованием барабанного командоаппарата.

### **5.3. Технические характеристики изучаемых микроконтроллеров**

#### **5.3.1. Общие положения**

Семейство контроллеров фирмы Automation Direct включает следующие разновидности контроллеров:

DirectLOGIC DL05 – серия экономичных моноблочных микроконтроллеров;

DirectLOGIC DL06 – серия моноблочных микроконтроллеров;

DirectLOGIC DL105 – моноблочный малогабаритный микроконтроллер;

DirectLOGIC DL205 – модульный миниконтроллер;  
 DirectLOGIC DL305 – модульный контроллер;  
 DirectLOGIC DL405 – модульный контроллер с большой вычислительной мощностью.

Серия микроконтроллеров фирмы Automation Direct DL05 включает в себя шесть различных моделей. Они отличаются параметрами каналов ввода/вывода информации (родом тока и уровнем допустимых напряжения и тока). Все они имеют один и тот же внешний вид и одинаковые характеристики процессора. Они имеют небольшие размеры, но включают систему команд, обычно присущую более большим, дорогостоящим системам.

Микроконтроллеры серии DL06 являются обновлённой версией микроконтроллеров серии DL05. Серия DL06 включает в себя уже восемь различных моделей, также различающихся параметрами каналов ввода/вывода информации. Они имеют большее количество каналов ввода/вывода, чем микроконтроллеры DL05. Все модели микроконтроллера имеют одинаковый внешний вид и одинаковые характеристики процессора.

### 5.3.2. Устройство микроконтроллеров DL06

На рис. 5.1 представлен микроконтроллер серии DL06 исполнения DO-06DD1. На нём показаны основные элементы микроконтроллера (табл. 5.1).

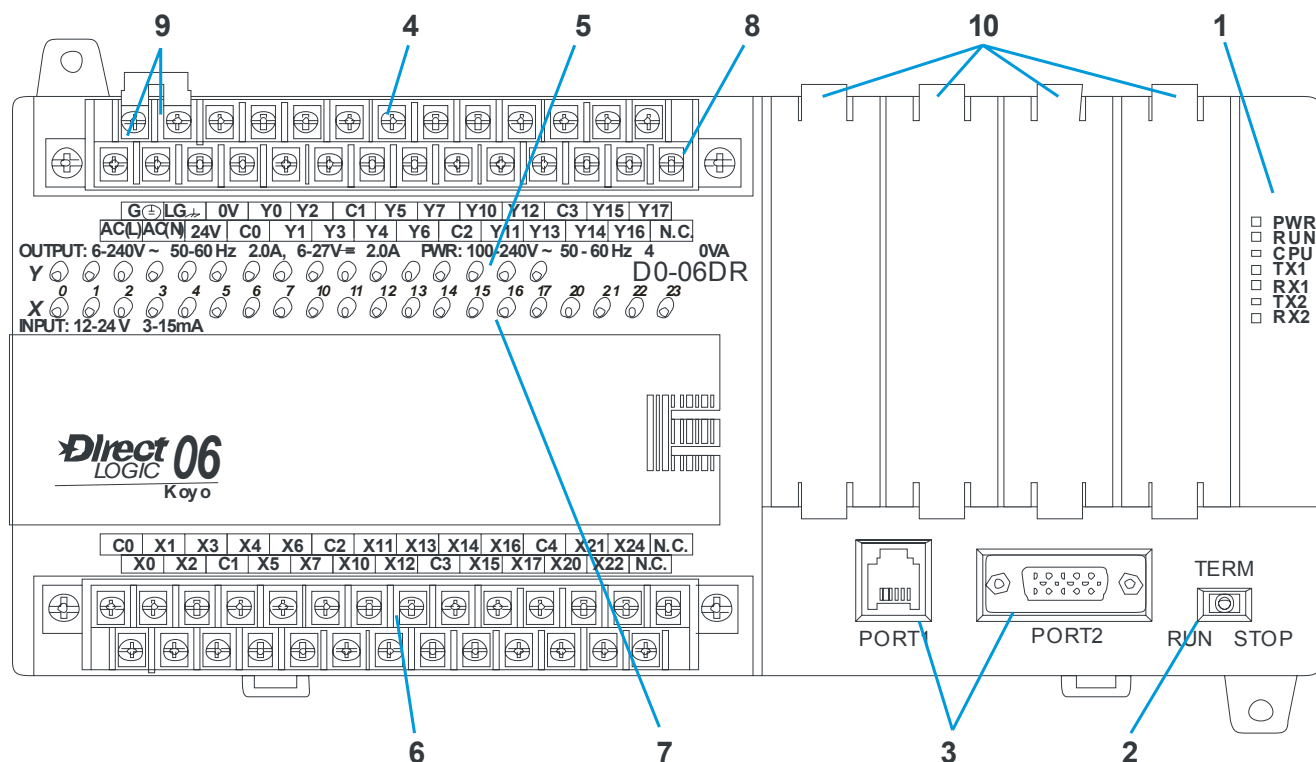


Рис. 5.1. Общий вид микроконтроллера DO-06DD1

Питание контроллера осуществляется от сети переменного тока напряжением 100...240 В частотой 50/60 Гц. Потребляемая мощность не более 40 ВА.



Таблица 5.1

№ п/п	Объект	Описание	
1	Индикаторы состояния контроллера	Питание (PWR)	Отображает состояние питания системы
		Работа (RUN)	Отображает состояние работы: ВКЛ, ВЫКЛ, STOP.
		Ошибка (CPU)	Отображает неполадки в работе процессора
		Передача данных (TxD1, TxD2)	Отображают процесс передачи информации при программировании либо при связи с другим контроллером
		Прием данных (RxD1, RxD2)	Отображают процесс приема информации при программировании либо при связи с другим контроллером
2	Переключатель режимов	Переключатель режимов работы контроллера - RUN: выполнение программы - STOP: останов выполнения программы - TERM: программирование контроллера	
3	Коммуникационные порты	Порты для связи с ПЭВМ (при программировании) или с другим контроллером (при параллельной работе)	
4	Дискретные выходы	Контакты выходов контроллера	
5	Индикаторы состояния выходов	Отображают состояние выходов контроллера	
6	Дискретные входы	Контакты входов контроллера	
7	Индикаторы состояния входов	Отображают состояние входов контроллера	
8	Контакт для подключения внешнего напряжения +24В	Питание входов/выходов контроллера	
9	Контакты для подключения напряжения ~220В	Питание внутренних цепей контроллера	
10	Модули расширения	Дополнительные модули дискретных входов контроллера (в данном исполнении их три)	

### 5.3.3. Общие характеристики микроконтроллеров DL06

В табл. 5.2 приведены характеристики единые для всех контроллеров DL06.

Таблица 5.2

Характеристика	Описание
Инструкций (команд)	229, в том числе тригонометрические
Память программы (слов)	7680
Память данных (слов)	7616
Входы (бит)	512
Выходы (бит)	512
Внутренние переменные (бит)	1024
Стадии (бит)	1024
Таймеры (бит)	256
Счетчики (бит)	128
Режимы работы	Работа (RUN), Стоп (STOP), Программирование (TERM)
Дополнительно	Встроенные часы реального времени
	Арифметика с плавающей запятой

### 5.3.4. Входы и выходы микроконтроллеров DL06

Технические характеристики входов микроконтроллера сведены в табл. 5.3.

Таблица 5.3

Количество входов	50 (включая 30 входов дополнительных модулей)	
Потенциальное разделение	Оптопара	
Тип входов	Транзисторные (потребитель или источник)	
	X0-X3	X4-X23, X100-X150*
Постоянное напряжение	10.8/26.4 В	
Ток	6/13 мА	4/8.5 мА
Напряжение/ток логической 1	10 В / 5 мА и выше	10 В / 4 мА и выше
Напряжение/ток логического 0	2 В / 0,5 мА и ниже	2 В / 0,5 мА и ниже
Входное сопротивление	1,8 кОм	2,8 кОм
Время срабатывания	70 мкс	4 мс

\*Адресация дополнительных модулей, начинается с цифры 100.

Внешний источник питания присоединяется одним полюсом к общей точке (общим точкам), а другим непосредственно к входу (входам) контроллера. На 50 входов имеется 11 общих изолированных точек (C0, C1...): 5 точек – по 4 входа на каждую (процессорный модуль), 6 точек – по 5 входов на каждую (модули дополнительные).

На рис. 5.2 показан вариант подключения входов контроллера. Полярность подключения значения не имеет и определяется пользователем при решении конкретной задачи.

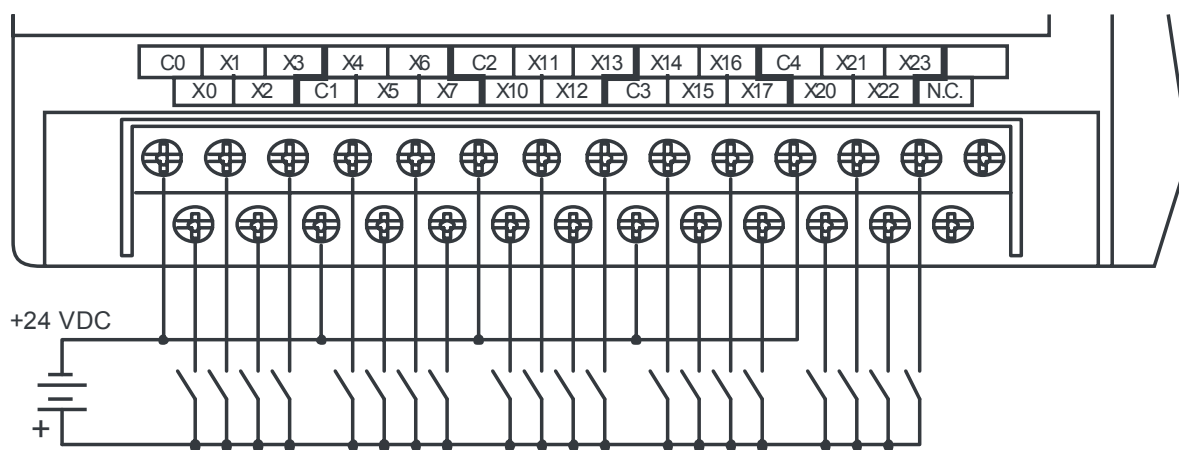


Рис. 5.2. Вариант подключения входов микроконтроллера DL06

Дискретные входы X0-X3 несколько отличаются от стандартных входов X4-X150 (рис. 5.3) характеристиками и внутренним устройством и могут быть использованы как высокоскоростные.

Каждый дополнительный модуль добавляет 10 дискретных входов. Подключение входов дополнительных модулей расширения осуществляется аналогично основным входам (рис. 5.3).

Технические характеристики выходов микроконтроллера сведены в табл. 5.4.

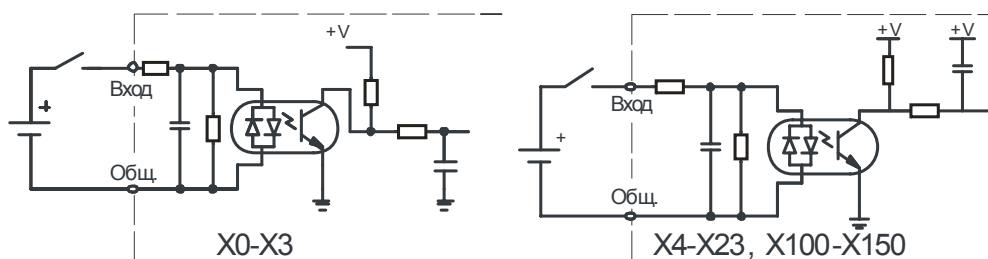


Рис. 5.3. Дискретные входы микроконтроллера

Таблица 5.4

Количество выходов	16	
Тип выходов	Транзисторные (потребитель)	
	Y0-Y1	Y2-Y17
Постоянное напряжение	5/30 В	
Ток	0.5/1 А	
Ток утечки	15 мкА (30 В)	
Время переключения	10/20 мкс	10/60 мкс

На выходах общие точки (C0-C3) имеют следующее распределение:

- C0: для выхода Y0-Y3;
- C1: для выхода Y4-Y7;
- C2: для выходов Y10-Y13;
- C3: для выходов Y14-Y17.

Внешний источник питания подсоединяется отрицательным полюсом к общей точке (общим точкам), а положительным полюсом непосредственно к нагрузке контроллера и контакту +V. На рис. 5.4 показан возможный вариант подключения нагрузки на выходы контроллера.

Внимание! Соблюдение полярности – обязательно!

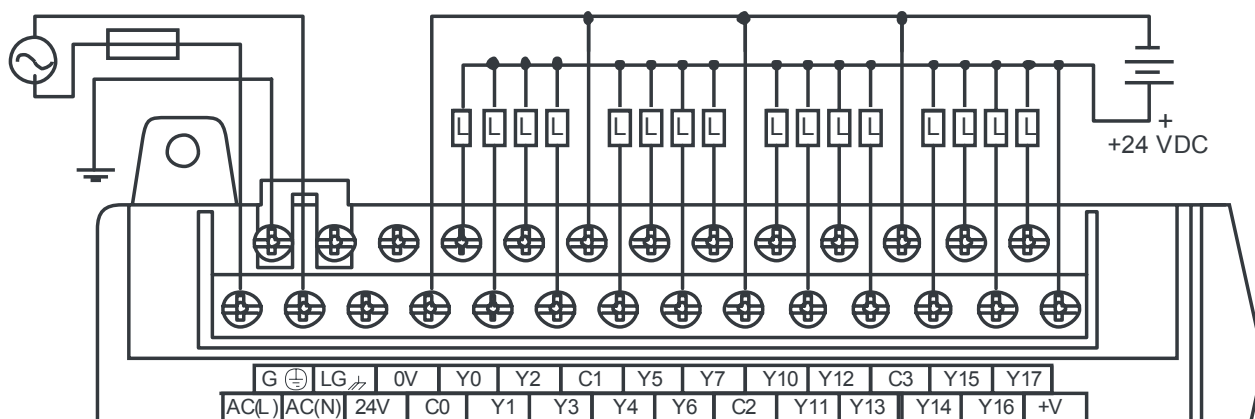


Рис. 5.4. Вариант подключения выходов микроконтроллера DL06

Дискретные выходы Y0-Y1 (рис. 5.5) несколько отличаются от стандартных выходов Y2-Y17 характеристиками и внутренним устройством, и могут быть использованы как импульсные выходы.

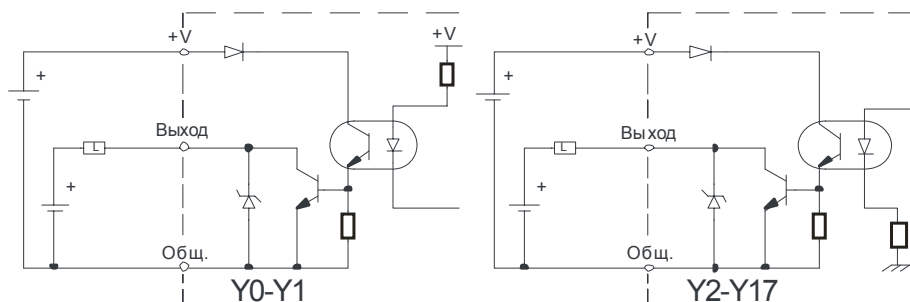


Рис. 5.5. Дискретные выходы микроконтроллера

Микроконтроллеры DL05 имеют всего 8 входов и 6 выходов, подключение которых аналогично микроконтроллерам DL06.

### 5.3.5. Распределение памяти микроконтроллеров

Вся доступная пользователю память (V-память) разделена на 16-разрядные регистры, т.е. доступна по ссылке V [восьмеричный порядковый номер регистра].

Все переменные, используемые при программировании, имеют свой адрес, который отражен в карте памяти контроллера DL06. Объем памяти и ее распределение DL05 такое же, как у DL06. Но так как входов/выходов у DL05 значительно меньше, чем у DL06, то число используемых ячеек памяти меньше. Начальные же адреса ячеек используемых областей памяти такие же, как у DL06. Распределение памяти микроконтроллеров DL06 и DL05 (для DL05 – в нижней строке в круглых скобках), обозначение и количество используемых сигналов представлено в табл. 5.5.

Формат представления информации – восьмеричный. Например, ниже показана ячейка с адресом V40400, в которой размещаются, биты состояния входов с адресами X0...X17:

Биты состояния входов	X17	X16	X15	X14	X13	X12	X11	X10	X7	X6	X5	X4	X3	X2	X1	X0	Адрес ячейки V40400
-----------------------------	-----	-----	-----	-----	-----	-----	-----	-----	----	----	----	----	----	----	----	----	---------------------------

Аналогично в своих ячейках располагаются биты состояния Y, C, S, T и SP.

Для каждого таймера кроме бита состояния выделяется еще ячейка для текущего значения счета таймера. Например, для 0-го таймера выделяется бит T0 в ячейке V41100 и ячейка V0000 для хранения текущего значения счета:

Биты состояния таймеров	T17	T16	T15	T14	T13	T12	T11	T10	T7	T6	T5	T4	T3	T2	T1	T0	Адрес ячейки V41100
Текущее значение счета																	Адрес ячейки V00000

Аналогично реализуются и счетчики. Таймер считает импульсы внутреннего генератора, а счетчик считает внешние события.

Таблица 5.5

Тип памяти	Восьмеричный указатель дискретной памяти	Область памяти	Кол-во, бит	Обозначение
Входы	X0 – X777 (X0 – X377)	V40400...V40437 (V40400...V40417)	512 (256)	X0 ┌─┴─┐
Выходы	Y0 – Y777 (Y0 – Y377)	V40500...V40537 (V40500...V40517)	512 (256)	Y0 ┌─( )─┐
Внутренние переменные	C0 – C1777 (C0 – C777)	V40600...V40677 (V40600...V40637)	1024 (512)	C0    C0 ┌─┴─┐ ┌─( )─┐
Специальные реле	SP0 – SP777 (SP0 – SP777)	V41200...V41237 (V41200...V41237)	512	SP0 ┌─┴─┐
Таймеры	T0 – T377 (T0 – T177)	V41100...V41117 (V41100...V41107)	256 (128)	┌─ TMR    T0    ─┐ └─ K100           ─┘    T0 ┌─┴─┐
Текущее значение таймера	Нет	V0000...V0377 (V0000...V0177)	256 слов (128 слов)	V0 K100 ┌─┴─┐
Счетчики	CT0 – CT177 (CT0 – CT177)	V41140...V41147 (V41140...V41147)	128	┌─ CNT    CT0    ─┐ └─ K10           ─┘    CT0 ┌─┴─┐
Текущее значение счетчика	Нет	V1000...V1177 (V1000...V1177)	128 слов	V1000 K100 ┌─┴─┐
Слова данных	Нет	V1200...V7577 (V1200...V7377)	3328 слов (3196 слов)	Используются со многими командами
Слова данных, сохраняемые	- Нет	- (V7400...V7577)	- (128)	Используются со многими командами
Стадии	S0 – S1777 (S0 – S777)	V41000...V41077 (V41000...V41017)	1024 256	┌─ SG            ─┐ └─ S001           ─┘    SP1 ┌─┴─┐
Системные параметры	Нет	V7600...V7777	128	Используется для разных целей
Глобальные входы (удаленные)	GX0 – GX3777 -	V40000...V40177 -	2048 -	GX0 ┌─┴─┐
Глобальные выходы (удаленные)	GY0 – GY3777 -	V40200...V40377 -	2048 -	GY0 ┌─( )─┐
Системные параметры	Нет	V7600...V7777	128 слов	

Программное обеспечение контроллера позволяет считывать текущие значения таймеров и счетчиков и использовать их для сравнения с содержимым любой другой ячейки памяти на равно, не равно, меньше, больше или равно.

Особенностью контроллеров D0-06DD1 является наличие специальных реле SP. Это реле, выполняющие определенную функцию. Например, реле SP1 имеет всегда замкнутый контакт, реле SP5 в течение 50 мс включено и в течение 50 мс отключено. Реле SP7 включается в каждое другое сканирование. Реле SP3 включается при низком напряжении батареи и т.п. Контакты этих реле можно без ограничений использовать в программе.

Наличие стадий S используется в языке RLL<sup>PLUS</sup> при составлении программ не по логическим уравнениям, а фактически по циклограмме работы объекта. Вся программа может быть разбита на нумерованные стадии (до 1024) и с каждой из них можно связать реле S. Когда стадия активна, то выполняются все инструкции соответствующей стадии, и наоборот, если стадия пассивна, то пропускаются все действия внутри стадии.

## 5.4. Программирование микроконтроллеров

### 5.4.1. Общие положения

Микроконтроллеры DL05 и DL06 можно программировать с помощью графического пакета DirectSoft32. Для программирования доступно два языка:

- язык RLL (Relay Ladder Logic) – язык релейной логики;
- язык RLL<sup>PLUS</sup>, объединяющий стандартный язык релейной логики с возможностями стадийного программирования.

Микроконтроллеры используют восьмеричную систему счисления.

### 5.4.2. Программирование на языке RLL

Программное обеспечение RLL обеспечивает программирование на языке лестничных диаграмм LD (Ladder Diagram), на языке инструкций IL (Instruction List) и с помощью барабанного командоаппарата.

Язык RLL ориентирован на программирование по уравнениям алгебры Буля (по логическим уравнениям). Он требует знания синтеза таких уравнений и преобразования их в релейный эквивалент.

В языке RLL есть 5 групп команд:

- булевы (булевские) команды;
- команды таймера;
- команды счетчика;
- команды сумматора;
- табличные команды.

Ниже в указанном порядке будет рассмотрено использование перечисленных команд. Полный перечень всех команд представлен в Приложении А данного раздела. Язык лестничных диаграмм использует представление программы в виде своеобразной релейно-контактной схемы. В табл. 5.6 представлены контакты, используемые в программах микроконтроллеров, являющиеся основой группы **булевых команд**.

Для сокращения записей адресов ячеек V-памяти при программировании в программном пакете DirectSoft32 предусмотрена возможность вместо адресов ячеек присваивать на символьное обозначение TA00...TA17, использующее восьмеричный формат нумерования. Всего этих символьных обозначений – 16. Доступ к ячейкам TA осуществляется через контекстное меню Tools/Memory Editor, либо при одновременном нажатии клавиш Ctrl+Y. Ячейки памяти, имеющие символьное обозначение TA00...TA17, могут содержать текущее значение таймеров,

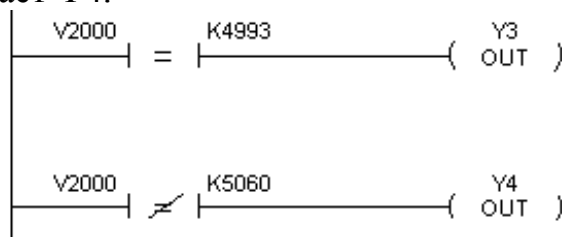
счетчиков, а также могут быть использованы для непосредственной записи в них чисел, используемых в математико-арифметических операциях.

При операциях сравнения могут быть использованы ячейки памяти, содержащие текущее значение таймеров, счетчиков, V-память и т.д.

Таблица 5.6

Контакт	Описание
	Замыкающий контакт (нормально разомкнутый)
	Размыкающий контакт (нормально замкнутый)
$\begin{matrix} \text{TA0} & & \text{TA1} \\ \text{---} & < & \text{---} \end{matrix}$	Контакт замкнут, при условии, что первая переменная (TA0) меньше второй (TA1)
$\begin{matrix} \text{TA0} & & \text{TA1} \\ \text{---} & \neq & \text{---} \end{matrix}$	Контакт замкнут, при условии, что первая переменная (TA0) не равна второй (TA1)
$\begin{matrix} \text{TA0} & & \text{TA1} \\ \text{---} & = & \text{---} \end{matrix}$	Контакт замкнут, при условии, что первая переменная (TA0) равна второй (TA1)
$\begin{matrix} \text{TA0} & & \text{TA1} \\ \text{---} & \geq & \text{---} \end{matrix}$	Контакт замкнут, при условии, что первая переменная (TA0) больше или равна второй (TA1)
	Контакт нормально разомкнутый, мгновенного действия (в первом скане)
	Контакт нормально замкнутый, мгновенного действия (в первом скане)
	Контакт, инвертирующий выход
	Контакт, срабатывающий по переднему фронту (при переходе сигнала 0 → 1)
	Контакт, срабатывающий по заднему фронту (при переходе сигнала 1 → 0)
$\left( \begin{matrix} \text{Y0} \\ \text{OUT} \end{matrix} \right)$	Контакт вывода (устанавливает соответствующий выход)
$\left( \begin{matrix} \text{Y0} \\ \text{SET} \\ \text{Y7} \end{matrix} \right)$	Фиксирование вывода (диапазона выводов)
$\left( \begin{matrix} \text{Y0} \\ \text{RST} \\ \text{Y7} \end{matrix} \right)$	Расфиксирование вывода (диапазона выводов)

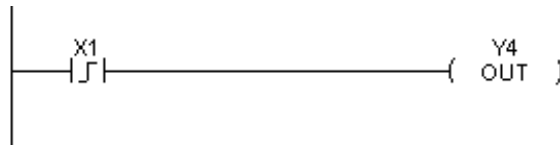
В следующем примере показано, что при значении в ячейке памяти V2000=4993, срабатывает выход Y3. Если же значение в этой в этой же ячейке V2000≠5060, то срабатывает Y4.



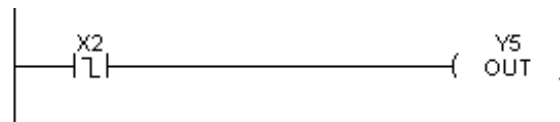
Здесь К означает представление константы. Вместо адресов ячеек, как уже указывалось выше, может быть использовано присвоенное им символьное обозначение ТА0...ТА17.

Состояния контактов мгновенного действия при формировании выходного воздействия берутся не из области данных, а непосредственно с входа контроллера.

В следующем примере показано что, каждый раз, когда X1 переходит из состояния Выкл. в состояние Вкл., обмотка Y4 будет замыкаться на время одного цикла сканирования.



В следующем примере при каждом переходе X2 переходит из состояния Вкл. в состояние Выкл. Обмотка Y5 будет замыкаться на время одного цикла сканирования.



Язык лестничных диаграмм (LD) прост и нагляден. Язык инструкций (IL) используется в основном лишь при корректировке программ с ручного программатора на месте установки контроллера в производственных условиях. Программу на языке LD программное обеспечение DirectSoft32 на ПЭВМ автоматически переводит на язык IL.

Язык IL использует булевый стек, имеющий 8 уровней. На рис. 5.6 представлено решение логического уравнения вида

$$X0 \cdot X1 + X2 \cdot X3 + X4 \cdot \overline{X5} = Y0$$

с использованием стека типа «первый пришел – последний вышел» и рабочего регистра.

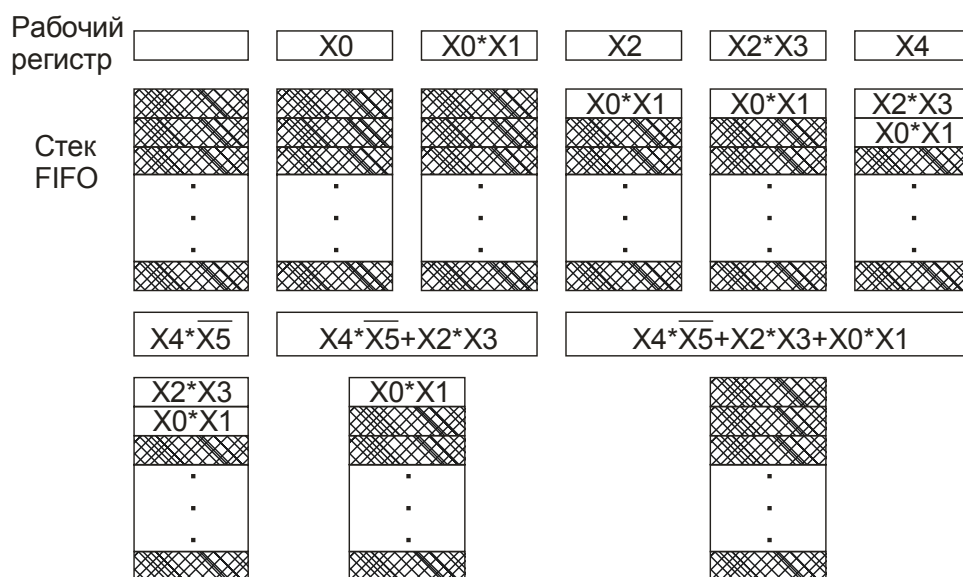


Рис. 5.6. Использование стека для решения логического уравнения



На рис. 5.7 приведены программы решения упомянутого уравнения на языках LD и IL.

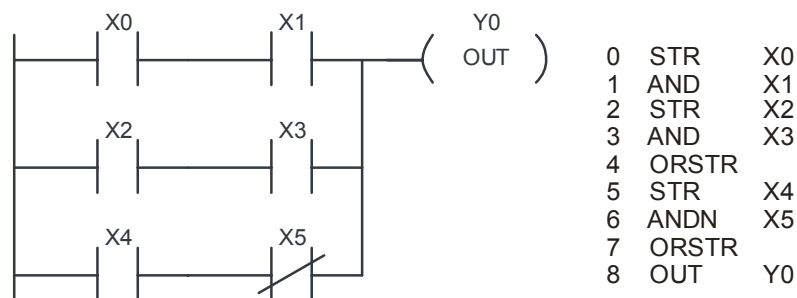


Рис. 5.7. Программа решения логического уравнения на языках LD и IL

При программировании на языке IL каждый очередной раз когда вводится команда STR (Store - память), эта команда перемещает данные из рабочего регистра в стек. Команды ANDSTR и ORSTR извлекают информацию из стека и совершают требуемые логические операции с содержимым рабочего регистра. Команда OUT (Output) отправляет содержимое рабочего регистра на выход.

Программа, использующая команды сравнения на языках LD и IL, представлена на рис. 5.8.

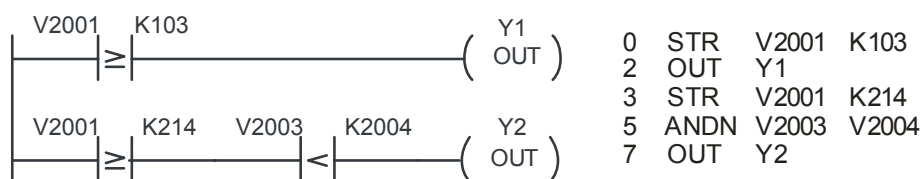


Рис. 5.8. Фрагмент программы, использующей контакты сравнения

Рассматриваемые микроконтроллеры реализуют два вида таймеров:

- простые таймеры с задержкой только переднего фронта входного сигнала;
- накапливающие таймеры для счета длительного включенного состояния входного сигнала.

Таймеры представляют собой счетчики, которые считают импульсы генератора, следующие с определенной частотой. У микроконтроллеров есть два генератора. Один генератор с дискретой времени следования импульсов 0,1 с (условно назван «медленный таймер»), второй – с дискретой времени 0,01 с (условно – «быстрый таймер»).

**Команды таймера** включают в себя:

- обозначение типа таймера. Простые таймеры обозначаются TMR (медленный таймер) и TMRF (быстрый таймер), накапливающие таймеры соответственно TMRA и TMRFA;
- номер таймера T0...T177 (для DL05), T0...T377 (для DL06);
- уставка таймера. Уставка задается десятичным числом от 0000 до 9999 для простых таймеров и от 0000 до 99999999 – для накапливающих.

Счет числа импульсов располагается в V-ячейках памяти, выделенных соответственно каждому таймеру (табл. 5.5). Когда текущее значение счета таймера достигнет уставки, выход таймера включится.

На рис. 5.9 представлена программа управления простым медленным таймером с номером T1 и уставкой времени 3 с на языке LD (рис. 5.9а) и на языке ПЛ (рис. 5.9б). Временные диаграммы (рис. 5.9в) показывают, что через 3 с после замыкания контакта X1 (входной сигнал) включается выход T1 таймера, что вызывает включение выхода Y0. Отключение таймера и сброс его ячейки V-памяти V0001 происходит при размыкании контакта X1.

Максимальная задержка переднего фронта входного сигнала для таймера TMR составляет 999,9 с, для таймера TMRF – 99.99 с.

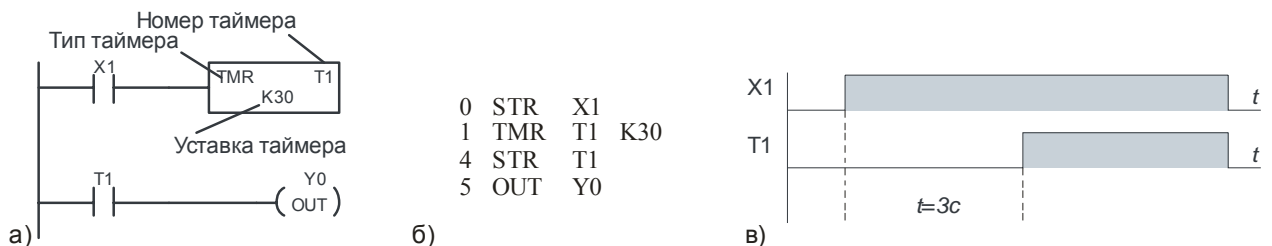


Рис. 5.9. Программирование простого таймера

На рис. 5.10 представлена программа управления накапливающим медленным таймером с номером T2 и уставкой времени 30 с на языке LD (рис. 5.10а) и на языке ПЛ (рис. 5.10б). Таймер имеет два входа. При замыкании контакта X1 (рис. 5.10в) происходит счет импульсов генератора. При размыкании контакта X1 счет прекращается, но подсчитанное число импульсов сохраняется. При следующем замыкании X1 счет продолжается и т.д. Когда накопленное число счета станет равным уставке – включается выход T2 таймера, что вызывает включение выхода Y1. При последующих включениях X1 счет продолжается, а выход сохраняется во включенном состоянии. Выход таймера отключается и происходит сброс накопленного значения при замыкании контакта X2 (сброс).

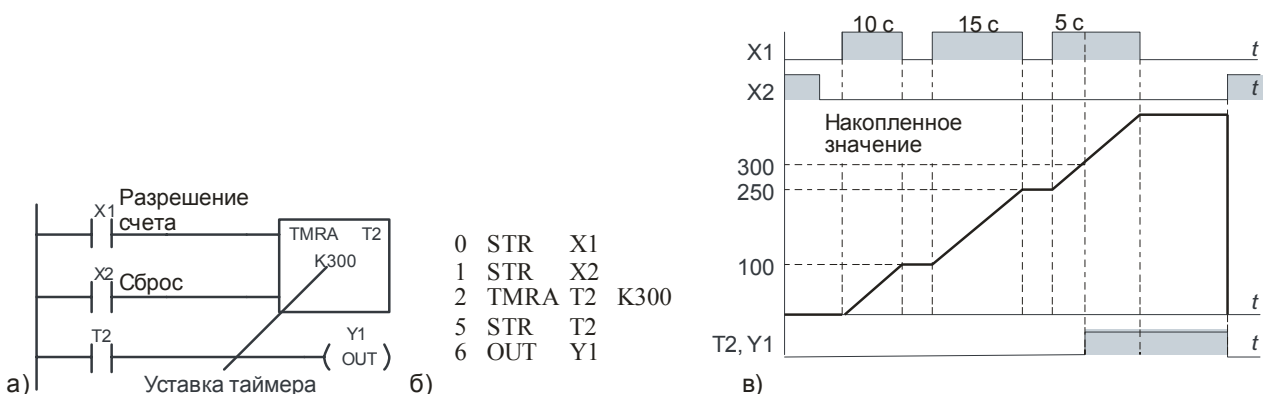


Рис. 5.10. Программирование накапливающего таймера

Накапливающему таймеру для счета импульсов выделяется две последовательных ячейки V-памяти общей емкостью счета до 99999999 в BCD-формате. В рассматриваемом случае выделяются ячейки V002 и V003.

Максимальное задание уставки таймера TMR A составляет 9999999,9 с а таймера TMRFA – 999999,99 с.

В микроконтроллерах счетчики реализуются подобно таймерам, только считают они не импульсы генератора, а события. Счет изменяется в случае, если сигнал на входе изменяется с 0 на 1, т.е. счетчик реагирует на передний фронт сигнала на счетном входе.

**Команды счетчика** включают в себя:

- обозначение типа счетчика. Рассматриваемые микроконтроллеры имеют два типа счетчиков: счетчик на сложение (обозначение «CNT») и реверсивный (обозначение «UDC»);
- номер счетчика CT0...CT177 для микроконтроллеров DL05 и DL06;
- уставка счетчика. Уставка задается десятичным числом от 0000 до 9999 для счетчика на сложение и от 0000 до 99999999 для реверсивного счетчика.

Счет числа событий располагается в V-ячейках памяти, выделенных соответственно каждому счетчику (табл. 5.5). Когда текущее значение счета достигает уставки, выход счетчика включается.

На рис. 5.11 представлена программа управления счетчиком на сложение CNT с номером CT1 и уставкой 3 на языке LD (рис. 5.11а) и на языке IL (рис. 5.11б). Временные диаграммы (рис. 5.11в) показывают, что после 3-х включений контакта X1 включается выход счетчика CT1, что вызывает включение выхода Y2. Отключение счетчика и сброс ячейки V-памяти V1001 происходит при замыкании контакта X2.

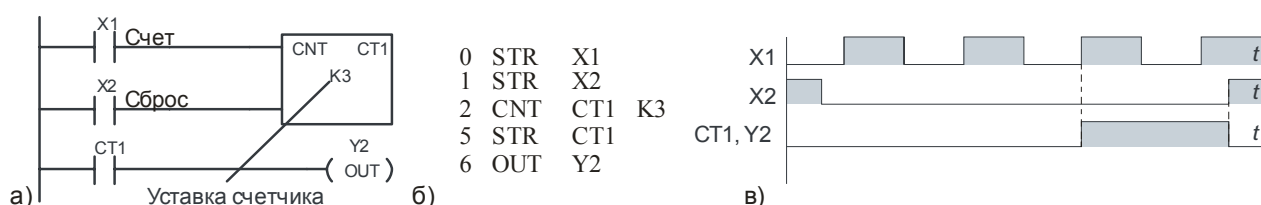


Рис. 5.11. Программирование счетчика на сложение

На рис. 5.12 представлена программа управления реверсивным счетчиком UDC с номером CT2 и уставкой 3 на языке LD (рис. 5.12а) и на языке IL (рис. 5.12б). Реверсивный счетчик имеет дополнительный вход на вычитание (Счет-). На рис. 5.12в представлены временные диаграммы работы счетчика UDC. Этот счетчик подсчитывает разницу в количестве срабатываний контактов X1 и X2. Когда она достигает 3-х, срабатывает выход счетчика CT2. замыкание контакта X3 сбрасывает выход счетчика и текущее значение соответствующих ячеек V-памяти. В рассматриваемом примере это ячейки V1002 и V1003.

При счете на увеличение достижение максимального значения (99999999 в BCD-формате) останавливает счет. То же – при достижении нуля при счете на уменьшение.

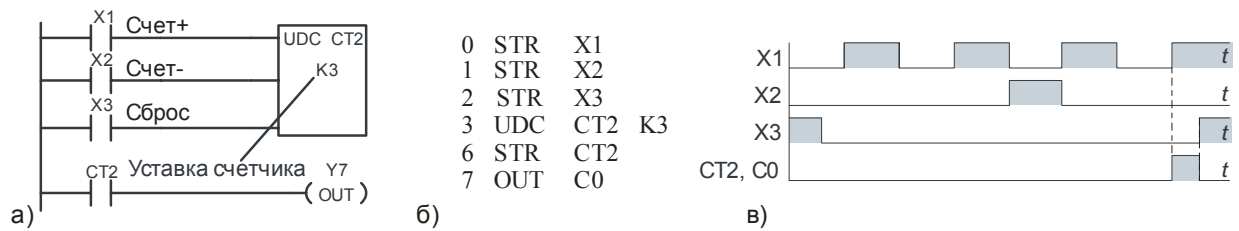


Рис. 5.12. Программирование реверсивного счетчика

**Команды сумматора** предназначены для реализации математических и логических операций, перемещения информации и т.п. У рассматриваемых микроконтроллеров есть 32 разрядный аккумулятор (сумматор), работающий с 16 разрядными ячейками V – памяти. Команды сумматора в полном объеме представлены в Приложении А данного раздела.

Пусть необходимо константу 4935 переместить из ячейки с адресом V1400 в ячейку с адресом V1410. Это обеспечивается программой рис. 5.13.



Рис. 5.13. Программа пересылки одинарного слова:  
а) программа; б) осуществление операции

На рис. 5.13а при замыкании контакта X1 командой LD (Load) содержимое ячейки V1400 помещается в аккумулятор, а командой OUT – из аккумулятора в ячейку с адресом V1410.

Следующая программа (рис. 5.14) содержимое двух последовательных ячеек V1401 и V1400 командой LDD (Load Double) помещает в аккумулятор, а команда OUTD (Out Double) пересылает в две последовательный ячейки V1411 и V1410 содержимое аккумулятора.

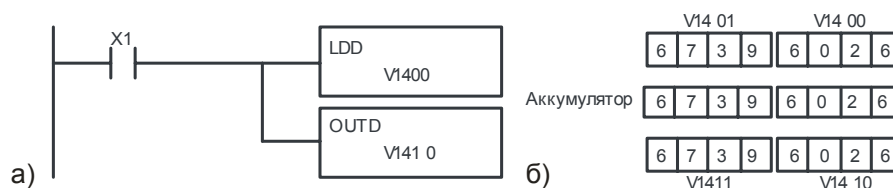


Рис. 5.14. Программа пересылки двойного слова:  
а) программа; б) осуществляемые операции

Аккумулятор имеет 32-х разрядный стек на 8 уровней. Каждый раз, когда значение командой LD или LDD помещается в стек, другое значение в стеке смещается вниз на один уровень. Если в стек записать больше 8 уровней, то значение из 8-го уровня удаляется из стека и может быть потеряно.

Команда POP сдвигает значение стека вверх. После команды POP значение, которое было в аккумуляторе, удаляется, а то, которое было в стеке, помещается в аккумулятор. Ниже (рис. 5.15) приведена программа засылки констант в ячейки V1400, V1401 и V1402.

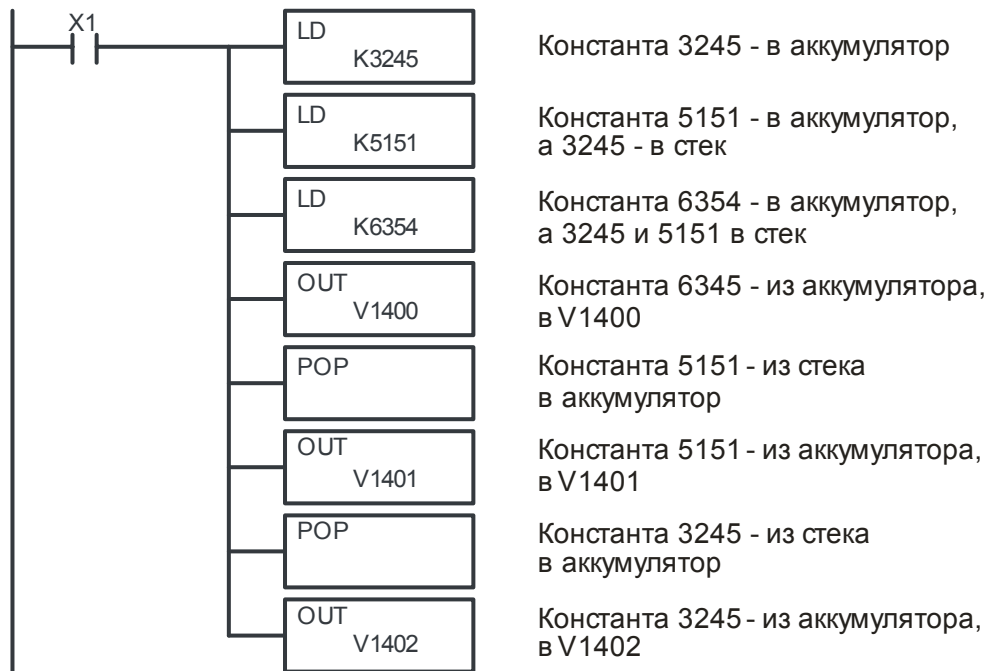


Рис. 5.15. Программа пересылки данных

Пример программы, выполняющей вычисление  $Y = ((V2000 + V2001) * V2002) / V2003$  и помещающей результат в ячейку V2004 на языках LD и IL приведен на рис. 5.16.

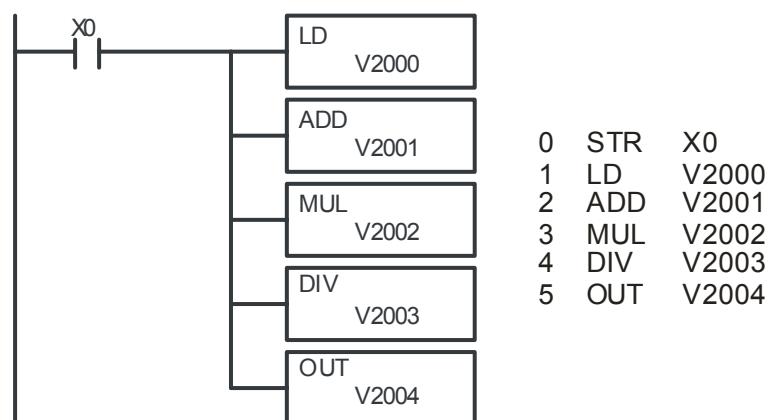


Рис. 5.16. Пример выполнения арифметических действий

Примечание:

- операции осуществляются с 16-ричными числами;
- все переменные хранятся в ячейках памяти V2000...V2003;
- вычисление происходит при нажатии кнопки Пуск (адрес X0)\$
- результат арифметических операций сохраняется в 16-ти младших разрядах 32-битного аккумулятора, а флаги хранятся в следующих специальных регистрах (реле) памяти:

SP60 – если число в аккумуляторе меньше числа в ячейке V-памяти;

SP61 – если число в аккумуляторе равно числу в ячейке V-памяти;

SP62 – если число в аккумуляторе больше числа в ячейке V-памяти;

SP63 – при нулевом значении результата операции (флаг Z);

SP66 – при превышении значения результата операции 16 бит;

SP67 – при превышении значения результата операции 32 бит;

SP70 – при отрицательном значении результата операции (флаг S).

**Табличные команды**, которых всего три (Приложение В), обеспечивают перемещение из одной области V-памяти в другую область V-памяти и т.п.

**Программирование с помощью барабанного командоаппарата** позволяет сэкономить время при программировании и отладке.

Команды барабанного командоаппарата, работающие по событиям программным образом, имитируют электромеханический барабанный командоаппарат (рис. 5.17). Команды барабанного командоаппарата наилучшим образом подходят для циклических процессов, состоящих из конечного количества шагов. Они могут простым образом выполнить работу нескольких цепей релейной логики.

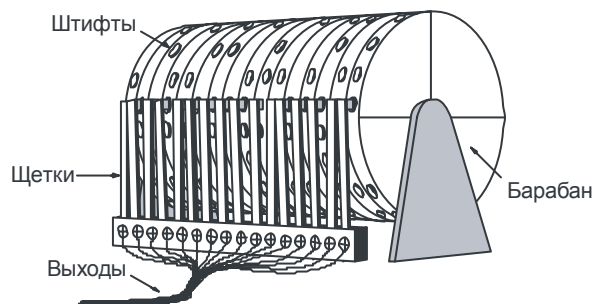


Рис. 5.17. Барабанный командоаппарат

Барабанный командоаппарат с переходом по времени обеспечивает 16 шагов и 16 выходов (рис. 5.18). Переход с шага на шаг происходит только по времени, заданным числом единиц счета на шаг. Неиспользуемые шаги должны программироваться со значением «единиц счета на шаг»=0. Дискретными точками выхода может быть независимо назначен тип X, Y и C, или они могут быть оставлены неиспользуемыми.

Как только будет инициализирован вход «Старт», станет доступным таймер барабанного командоаппарата. Он останавливается при завершении последнего шага или при инициализации входа «Сброс».

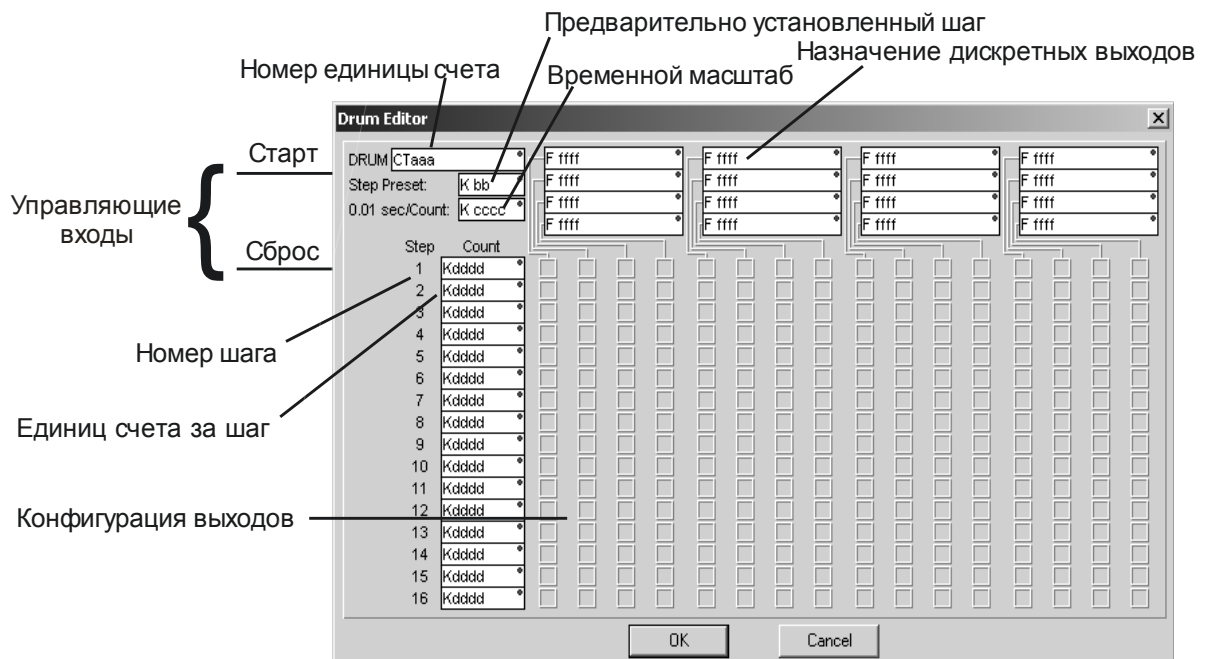


Рис. 5.18. Барабанный командоаппарат с дискретными входами и переходом по времени (DRUM)

Барабанный командоаппарат с дискретными входами и переходом по событию обеспечивает 16 шагов и 16 выходов (рис. 5.19). Переход шага происходит по времени или по событию. Вход «Переход» также вызывает переход шага при каждом включении. Время задается в единицах счета за шаг, а события задаются как дискретные контакты. Тип дискретных точек выхода может быть назначен независимо.



Рис. 5.19. Барабанный командоаппарат с дискретными входами и переходом по событию (EDRUM)

Как только будет инициализирован вход «Старт», станет доступным таймер барабанного командоаппарата. Таймер работает, пока имеет место событие для данного шага. Когда номер единицы шага становится равным значению единиц счета за шаг, командоаппарат переходит к следующему шагу. Этот процесс останавливается при завершении последнего шага или при инициализации входа «Сброс». Командоаппарат переходит к выбранному предварительно установленному шагу при переходе процессора в Рабочий режим программы, а также при каждой инициализации входа «Сброс».

### 5.4.3. Примеры программирования контроллера на языке RLL

В качестве примера рассматривается управление воротами (дверью) въезда/выезда на территорию гаража.

Для управления открытием/закрытием ворот имеется только одна кнопка  $K_H$ . Открытое и закрытое положения ворот контролируются соответственно датчиками  $D_O$  и  $D_3$ . Начальное (исходное) положение ворот закрытое (есть сигнал с датчика  $D_3$ ). Для открытия ворот кратковременно нажимается, а затем отпускается кнопка  $K_H$ . Команда на открытие ворот  $O$  возникает только после отпускания кнопки и сохраняется до срабатывания датчика открытого положения ворот  $D_O$ . Одновременно с появлением команды  $O$  включается предупредительный звонок  $Зв$ , который должен звенеть в течение 18 с. Закрывание ворот аналогично: при кратковременном нажатии и последующем отпускании кнопки  $K_H$  возникает команда  $З$  на закрывание ворот и включается на 18 с предупредительный звонок. При полном закрытии ворот (срабатывание датчика  $D_3$ ) привод ворот отключается. Звонок отключается по истечении задержки 18 с.

Для программирования на языке RLL следует составить логические уравнения выходных команд и сигналов, а также внутренних выходов (промежуточных переменных).

Вводится промежуточная переменная  $P_K$  о нажатии кнопки управления  $K_H$ . Она возникает при нажатии кнопки  $K_H$  и сохраняется до появления команды  $O$  или  $З$ :

$$P_K = (K_H + P_K) \cdot \overline{(O + З)} = (K_H + P_K) \cdot \overline{O} \cdot \overline{З}$$

Команда  $O$  возникает при закрытом положении ворот, наличии памяти  $P_K$  и отпускании кнопки. Возникшая команда сохраняется до полного открытия ворот. Этому соответствует логическое уравнение:

$$O = (D_3 \cdot P_K \cdot \overline{K_H} + O) \cdot \overline{D_O}$$

Аналогично, команда  $З$  возникает при открытом положении ворот, наличии памяти  $P_K$  и отпускании кнопки. Возникшая команда сохраняется до полного закрытия ворот. Этому соответствует логическое уравнение:

$$З = (D_O \cdot P_K \cdot \overline{K_H} + З) \cdot \overline{D_3}$$

Команда на включение звонка  $Зв$  возникает при появлении команд  $O$  и  $З$  и сохраняется до включения таймера с задержкой времени 18 с. Командой на рабо-



ту таймера является сигнал о включении звонка Зв. То есть сигнал Зв запускает таймер, который через 18 с отключает звонок. Этому соответствуют логические уравнения:

$$Зв = (O + З + Зв) \cdot \overline{T}^{\uparrow}$$

$$T = Зв$$

В табл. 5.7 представлена адресация переменных в адресах программируемого микроконтроллера DL06 или DL05.

В табл. 5.8 представлена логические уравнения в адресах программируемого микроконтроллера.

Таблица 5.7

Команда, сигнал	К <sub>Н</sub>	Д <sub>0</sub>	Д <sub>3</sub>	О	З	Зв	Р <sub>К</sub>	Т
Адрес	X0	X1	X2	Y0	Y1	Y2	C0	T0

Таблица 5.8

Команда, сигнал	Уравнения в адресах ПЛК DL06
Р <sub>К</sub>	$(X0 + C0) \cdot \overline{Y0} \cdot \overline{Y1} = C0$
О	$(X2 \cdot C0 \cdot \overline{X0} + Y0) \cdot \overline{X1} = Y0$
З	$(X1 \cdot C0 \cdot \overline{X0} + Y1) \cdot \overline{X2} = Y1$
Зв	$(Y0 + Y1 + Y2) \cdot \overline{T0} = Y2$
Т	$Y0 = TMR$

Программа на языке лестничных диаграмм представлена на рис. 5.20.

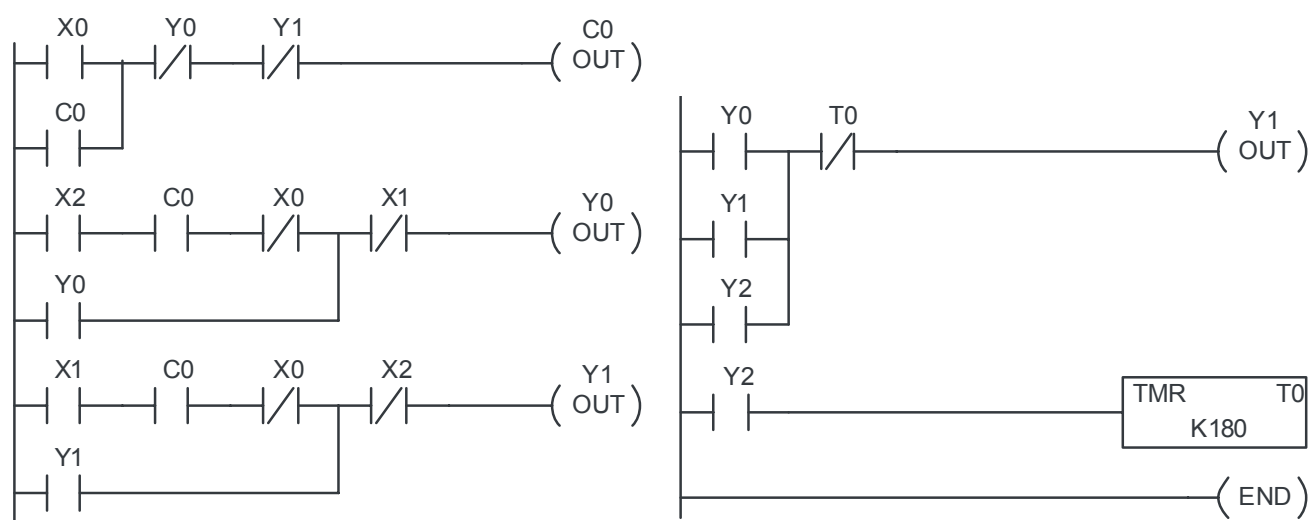


Рис. 5.20. Программа управления воротами гаража на языке лестничных диаграмм

#### 5.4.4. Программирование на языке RLL<sup>PLUS</sup>

Программирование на языке стадий RLL<sup>PLUS</sup> (стадийное программирование RLL<sup>PLUS</sup>) не требует составления логических уравнений. Это практически программирование по циклограмме технологического процесса. Такое программирование сокращает время одного скана программы.

В программе, составленной по логическим уравнениям, в пределах одного скана решаются все логические уравнения. При стадийном программировании весь процесс управления разбивается на ряд стадий. Стадия может выполняться в данном скане, если она активна, и не выполняться, если она пассивна (неактивна). Наличие пассивных стадий и сокращает время скана программы.

При стадийном программировании следует учитывать следующее:

а) стадии включаются, когда программа переходит от стадии к стадии с помощью команды JMP;

б) стадии можно включать и отключать командами SET и RST точно также как использовали эти команды при управлении выходами;

в) активная стадия автоматически отключается при переходе к другой стадии;

г) стадии инициализации (ISG) автоматически включаются, когда контроллер переходит от режима Program к режиму Run. Таких стадий может быть несколько, и они одновременно могут быть активными;

д) можно использовать одну и ту же ситуацию для перехода к нескольким стадиям;

е) при выходе из стадии выходы, которые были включены командой OUT, автоматически отключаются. Если нужно сохранить выход, то используется команда SET.

Язык RLL<sup>PLUS</sup> в соответствии со стандартом МЭК61131 можно отнести к языку последовательных функциональных схем SFC (Sequential Function Chart), в котором любая SFC схема составляется из элементов, представляющих шаги и условия переходов.

На рис. 5.21 представлена программа на языке стадий RLL<sup>PLUS</sup> для рассмотренной выше задачи управления воротами въезда/выезда на территорию гаража. В этой программе приняты обозначения переменных в соответствии с табл. 5.7.

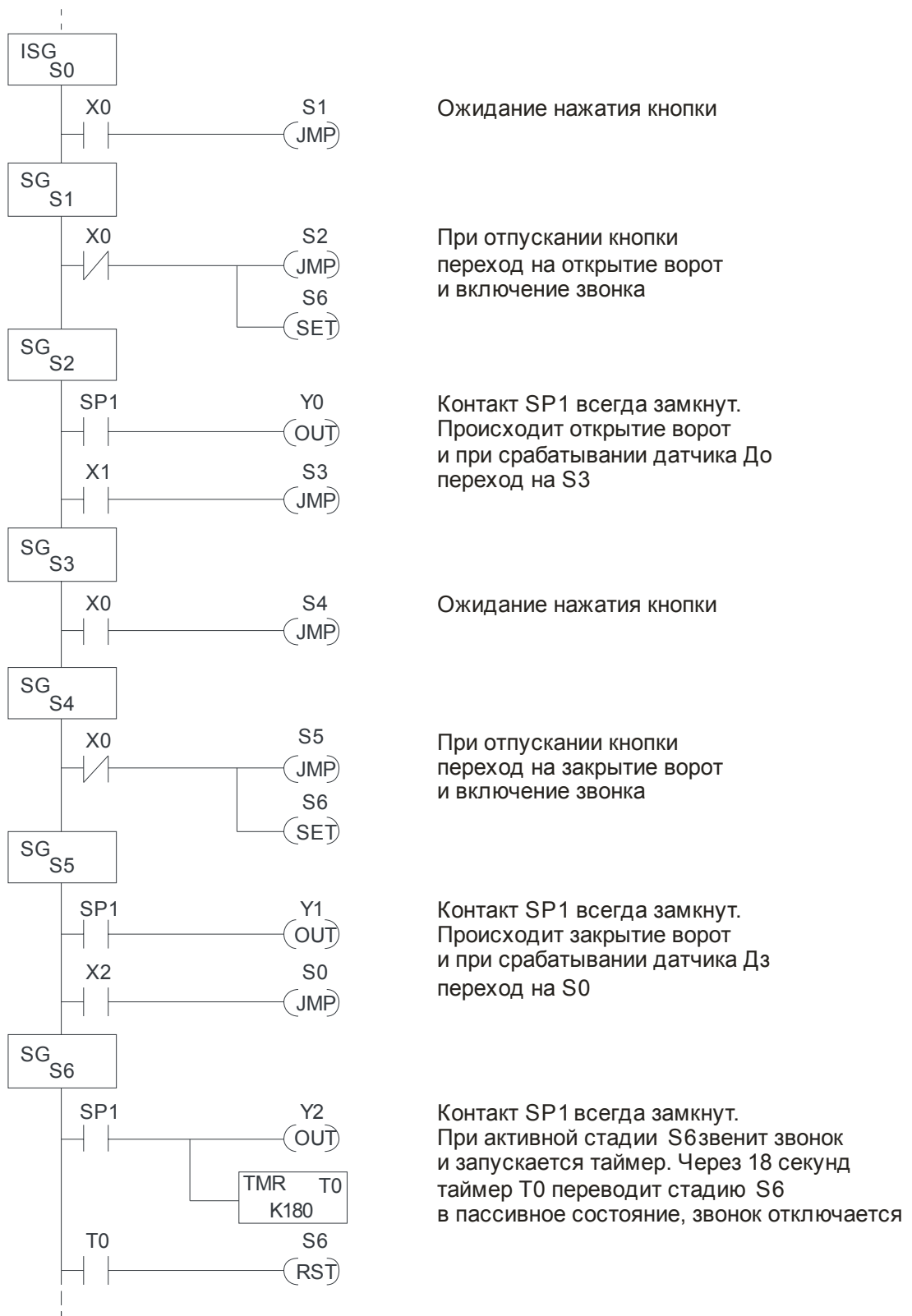


Рис. 5.21. Программа управления воротами гаража на языке RLL<sup>PLUS</sup>

## 5.5. Подготовка программы с помощью пакета программирования DirectSoft32

Для подготовки программ для микроконтроллеров DL служит специальный пакет программ DirectSoft32. Запуск DirectSoft32 осуществляется через соответствующий ярлык на рабочем столе или в меню «Пуск» персонального компьютера.

После запуска программы автоматически выводится окно создания нового проекта (рис. 5.22).

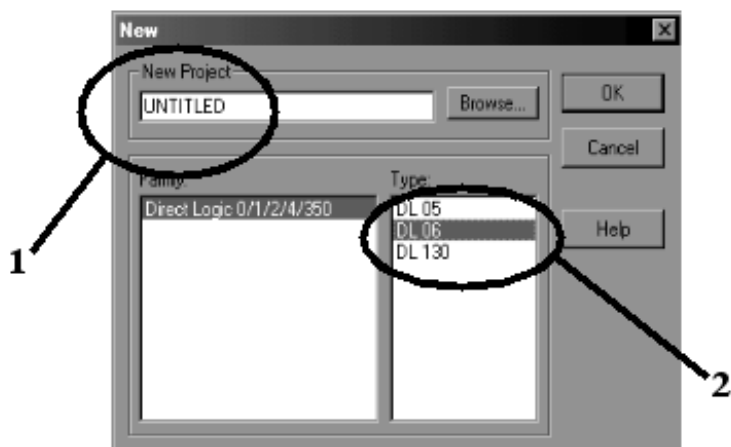


Рис. 5.22.


В появившемся окне (см. рис. 5.22) указываются имя проекта (1) и тип контроллера (2). По окончании ввода параметров следует провести подтверждение, нажав кнопку «ОК». После подтверждения появляется собственно само окно программы, которое представляет собой чистое (незаполненное) поле в виде цепей лестничной диаграммы.

Находясь в окне программы, можно выбирать компоненты из контекстного меню либо использовать значки на панели инструментов (рис. 5.23).



Рис. 5.23. Панель инструментов

Краткое описание назначения всех компонентов панели инструментов представлено в табл. 5.9.

Для начала программирования нужно перейти в режим редактирования (, при котором появляется дополнительная панель инструментов снизу экрана (рис. 5.24).

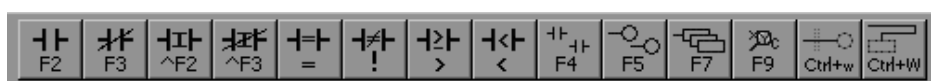
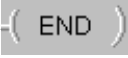

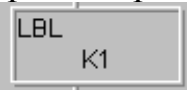


Рис. 5.24. Дополнительная панель инструментов

Таблица 5.9

Значок панели инструментов	Контекстное меню	Пояснения
	File/Read Program/from Disk File/Write Program/to Disk	Считать, хранящуюся на диске программу/Сохранение памяти программы на диске
	File/New Project/Offline	Создать новый автономный проект
	File/Open Project	Открыть уже существующий проект
	File/Backup Project	Создать резервную копию проекта
	Edit/Edit Mode	Перейти в режим редактирования
	Edit/Accept	Компиляция изменений
	Edit/Cut	Вырезать выделенные цепи в буфер обмена. Цепи выделяются клавишами-стрелками при удерживаемой клавише [Shift]
	Edit/Copy	Копировать выделенные цепи в буфер обмена
	Edit/Paste	Вставить из буфера обмена вырезанные или скопированные цепи
	Search/Find	Найти элемент программы
	Search/Find Again	Снова найти элемент программы
	Search/Element Browser	Запустить окно просмотра элементов
	View/Options	Управление различными параметрами визуального отображения программы
	View/Zoom In View/Zoom Out	Увеличить размер экрана Уменьшить размер экрана
	Help/Help Index	Помощь (на англ. языке)

Описание кнопок реле приведены выше, поэтому в табл. 5.10 представлено краткое описание назначения только некоторых компонентов данной панели инструментов.

**ВНИМАНИЕ!** Написанная программа обязательно должна заканчиваться ячейкой () , и не в коем случае (во избежание неполадок с центральным процессором), программа не должна заикливаться командами () и () .

Значок панели инструментов	Пояснения
	Посмотреть и выбрать различные типы контактов
	Посмотреть и выбрать различные типы выходных катушек
	Посмотреть и выбрать различные типы команд блока
	Вставить горизонтальное соединение с выходной колонкой
	Нарисовать соединение между контактом и блоком стадии

Для набора цепи, показанной на рис. 5.25 мышью или клавишами управления курсором подвести курсор к началу цепи. Выбрать мышью в нижней линейке инструментов замыкающий контакт и щёлкнуть левой кнопкой мыши. Появляется на схеме обозначение контакта. Вместо имеющейся подписи у контакта записать X0 и щёлкнуть левой кнопкой мыши на обозначение «галочки» в окне изображения контакта. Курсор автоматически смещается вправо на следующую позицию.

Аналогично вводится контакт X1. Стрелкой курсора при нажатой клавише «Ctrl» изобразить линию, шунтирующую контакт X1, с длиной достаточной для изображения 2-х контактов. Курсор переметить на место установки контакта T0.

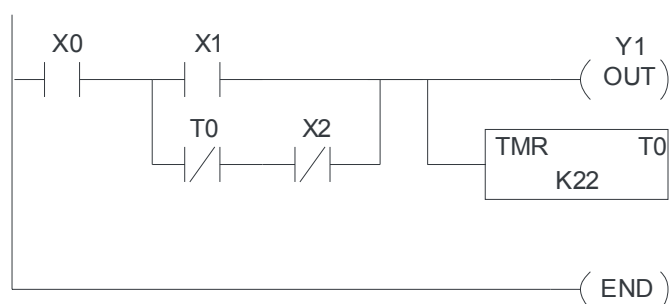




Рис. 5.25.

Выбрать мышью в линейке инструментов размыкающий контакт и аналогично предыдущему ввести контакт T0. Курсор автоматически смещается вправо на место записи второго контакта. Вводится размыкающий контакт X2. Курсор устанавливается на обозначение выхода. Нажатием клавиши «Delete» убирается бывшее изображение. Напечатать Y1 и щёлкнуть мышью на обозначение «галочки». Поставить курсор слева от выхода Y1, и, используя клавишу «Ctrl» и клавиши курсора. Организовать параллельную выходную цепь. Курсор стоит на позиции выхода. В линейке инструментов выбираем клавишу . В появившемся окне выбрать режим Vox и выбрать обозначение требуемого таймера. Мышью нажать на клавишу «ОК». В появившемся обозначении напечатать требуемый тип таймера и номер его контакта. Нажать клавишу «Enter» или на изображение «галочки». Перевести курсор на изображение выхода следующей второй строки программы.

Щёлкнуть мышью по клавише . В появившемся окне выбрать режим Program Control, а затем выбрать обозначение END. После нажатия мышью на «ОК» появится изображение схемы с внесённым выходом END. Можно не обращаться к окну, а сразу печатать слово END и дважды нажать на клавишу Enter. Это справедливо и для ввода других символов.

Всем элементам, используемым при программировании, можно присвоить условные буквенно-цифровые обозначения (по выбору пользователя). Для этого нужно нажать пункт меню Tools/Documentation Editor.

При необходимости можно поменять обозначение элемента, нажав на него два раза левой кнопкой мыши. Либо, если нужно поменять не только обозначение, но и тип элемента, то необходимо выбрать изменяемый элемент, нажав левую кнопку мыши, и нажать на новый тип элемента в нижней панели инструментов (либо выбрать другой элемент из списка, нажав ).

Вертикальные соединительные линии рисуются клавишами-стрелками при удерживаемой клавише «Ctrl», стираются они при удержании клавиш «Shift + Ctrl».

Для удаления отдельного элемента лестничных диаграмм (например, реле) необходимо в режиме редактирования выбрать его, нажав на него левой кнопкой мыши, и нажать клавишу «Delete».

Для удаления всей цепи лестничной диаграммы необходимо в режиме редактирования нажать левой кнопкой мыши на незанятую элементами часть цепи, нажать клавишу «Delete» и подтвердить удаление, нажав на «ОК».

Выделение цепей лестничных диаграмм производится клавишами-стрелками при удерживаемой клавише «Shift».

Для сохранения проекта необходимо нажать следующие пункты контекстного меню: File/Save Project/to Disk или File/Save Project As... [Сохранить как].

При нажатии правой кнопки мыши появляется меню, в котором можно выбрать тип отображения программы:

- 1) Ladder View – лестничные диаграммы;
- 2) Stage View – стадии;
- 3) Mnemonic View – мнемоническое представление (инструкции);
- 4) Cross Reference View – представление в виде перекрестных ссылок.

Для выхода из программы DirectSoft32 можно нажать File/Exit в контекстном меню, либо воспользоваться стандартной кнопкой закрытия приложения Windows в виде крестика в верхнем правом углу окна приложения.

При этом если с момента последнего сохранения программы на диск произошли какие-либо изменения, появится окно (рис. 5.26), предупреждающее о том, что внесенные изменения не были откомпилированы.

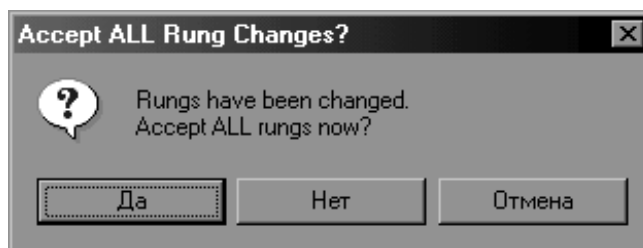


Рис. 5.26.

Откомпилировать программу можно, нажав кнопку «Да». Если же последние изменения сохранять не нужно, то нажать «Нет», Нажав «Отмена» – вы вернетесь обратно в программу. После нажатия кнопки «Да» (рис. 5.27) либо при выходе из программы после компилирования, появляется окно, предупреждающее о том, что программа была изменена, откомпилирована, но не сохранена.

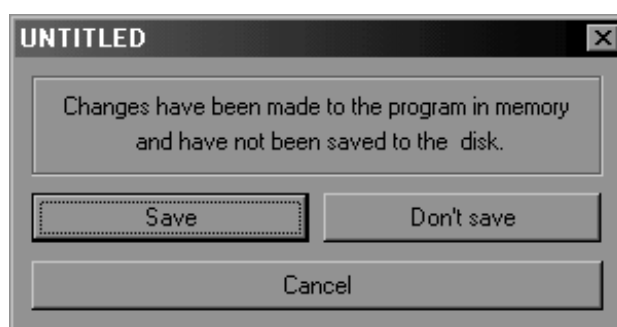



Рис. 5.27.

Можно либо подтвердить сохранение (кнопка «Save»), либо отказаться от сохранения (кнопка «Don't save»), либо отказаться от выхода из программы DirectSoft32 (кнопка «Cancel»).

Перед записью программы в контроллер её необходимо откомпилировать, для чего надо нажать кнопку (  ) либо Edit/Асcept. Если в программе обнаружены ошибки, то появится сообщение с их перечислением. Все выявленные ошибки следует исправить. Признаком успешной компиляции является зелёная вертикальная полоса, отмечающая цепи набранной схемы, включая цепь с командой END.

## 5.6. Ввод программы в микроконтроллер

После успешной компиляции программа готова к загрузке. Для работы с микроконтроллером необходимо включить его и установить режим TERM. Порядок загрузки и работы следующий:

- для связи с контроллером нажать пункты меню PLC/Connect...;
- в появившемся окне (рис. 5.28), поставить галочку (1) и нажать кнопку Select (признаком успешной связи является появление дополнительных кнопок на верхней панели инструментов);



- при успешном соединении с микроконтроллером появится окно запроса источника программы отображаемой при дальнейшей работе (рис. 5.29). Для записи в микроконтроллер составленной программы нажмите «Use Disk», для чтения программы из микроконтроллера нажмите «Use PLC»;
- нажмите на клавишу «Use Disk»;
- при появлении окна с сообщением об ошибке (рис. 5.30), следует убедиться в правильности подключения кабеля связи между ПК и ПЭВМ, а также в том, что микроконтроллер включен (светится индикатор PWR);

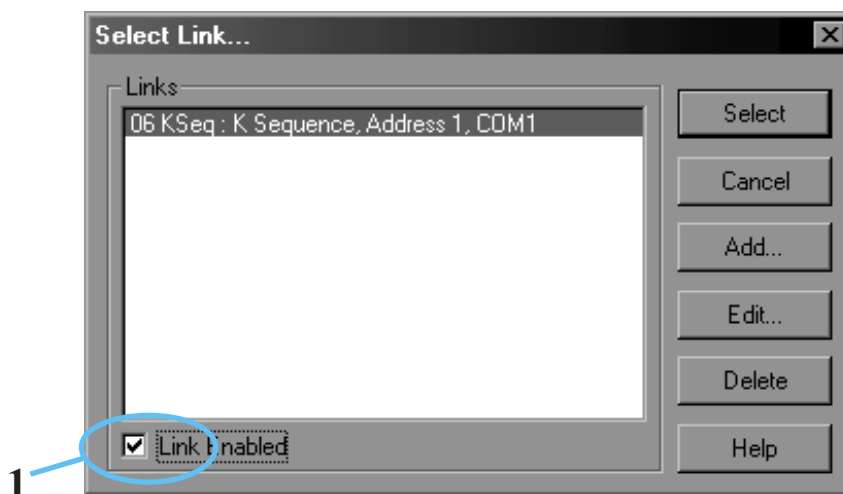


Рис. 5.28.

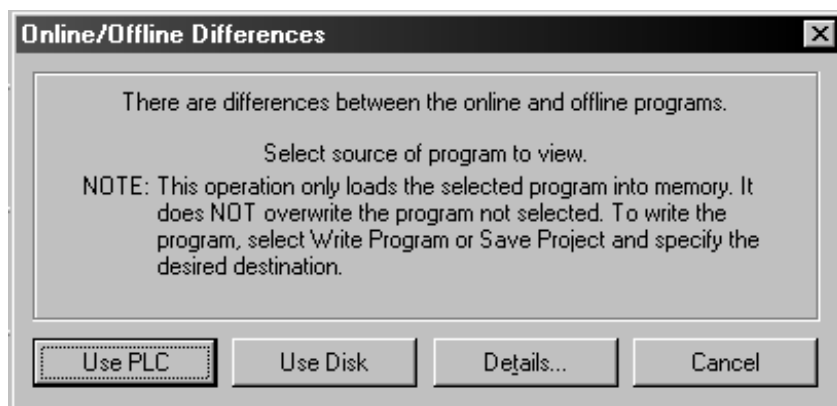



Рис. 5.29.



Рис. 5.30.

- при отсутствии ошибок появляется изображение схемы и дополнительные функциональные клавиши. Нажатием функциональной клавиши с изображением светофора выбрать в появившемся окне режим Program и нажать мышью клавишу «OK». При нажатии клавиши с изображением контроллера с упирающейся в него стрелкой происходит запись программы в контроллер. При этом переключатель контроллера должен находиться в среднем положении Term. Вновь нажать функциональную клавишу с изображением светофора, в появившемся окне выбрать режим RUN и после подачи мышью команды «OK» появляется схема программы ,

а на контроллере загорается дополнительная индикация RUN. Можно приступить к проверке работы программы на контроллере.

ПЭВМ позволяет наблюдать текущее состояние входов/выходов контроллера. Для этого необходимо мышью нажать на функциональную клавишу  в строке клавиш со светофором.

## 5.7. Вывод программы на печать

Для вывода программы на печать необходимо нажать пункт меню File/Print. В появившемся окне (рис. 5.31) можно установить следующие параметры печати:

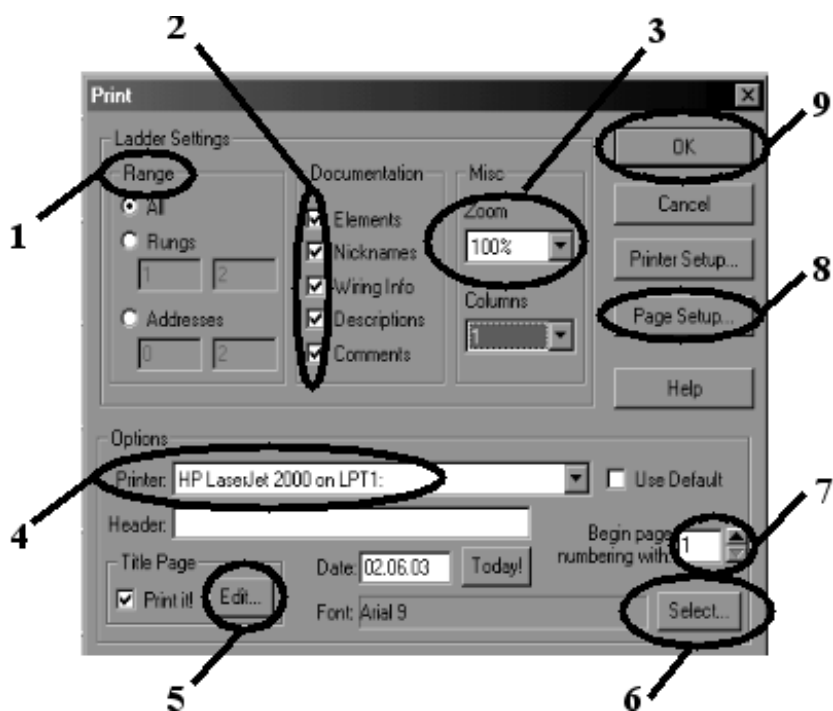


Рис. 5.31.

- 1) диапазон цепей лестничной диаграммы, который будет выведен на печать;
- 2) дополнительная информация (элементы, их обозначения, название цепей, пояснения, комментарии);
- 3) масштаб графического отображения программы;
- 4) выбор печатающего устройства;
- 5) заглавие страниц;
- 6) выбор шрифта выводимой на печать текстовой информации;
- 7) нумерация страниц (с какой будет начата);
- 8) задание полей печати (отступов от краев страницы);
- 9) подтверждение вывода на печать.

Существует возможность предварительного просмотра выводимой на печать программы. Для этого нужно нажать пункт меню File/Print Preview. В появившемся меню можно экспериментально подобрать все необходимые параметры и при

нажатии на кнопку «ОК», просмотреть результат будущей распечатки на экране монитора.

## 5.8. Дополнительные возможности микроконтроллеров

Возможности микроконтроллеров DL05 и DL06 многообразны. Руководство пользователя, например, микроконтроллера DL06 (DirectSoft32: руководство пользователя (файл DSoft32\_rus, каталог dl06user)) по объему составляет более 500 страниц. Поэтому в представленном пособии дан тот объем информации, который обеспечивает знакомство с основными функциями контроллера, позволяет запрограммировать на различных языках программирования задачу по управлению заданным объектом и проверить работу объекта по подготовленной программе.

В упомянутом руководстве пользователя можно ознакомиться с дополнительными возможностями контроллеров DL06:

- высокоскоростными счетчиками;
- использованием высокоскоростных входов и импульсных выходов;
- высокоскоростным прерыванием;
- аналоговыми входами;
- ПИД-регуляторами;
- картриджами памяти/часами реального времени.

## 5.9. Выполнение лабораторной работы

Лабораторная работа проводится на стенде, представляющем собой совокупность ПЭВМ типа IBM (системный блок, монитор, клавиатура, мышь, принтер) и контроллерный блок.

Контроллерный блок (рис. 5.32) конструктивно представляет собой корпус 1 с вмонтированным в нем программируемым контроллером 2 типа D0-05DD. На лицевой панели корпуса располагаются тумблеры 5 для ввода команд в контроллер, восемь светодиодов 4 индикации возбужденных (включенных) входов и шесть светодиодов 3 индикации возбужденных (включенных) выходов. На торце корпуса имеется доступ к переключателю 6 режимов контроллера RUN–TERM–STOP [работа–программирование–стоп] и к коммутационным портам 7 (Port1 и Port2) для связи с компьютером и с периферийными устройствами.

Питание программируемого контроллера осуществляется подачей переменного напряжения 220В при включении тумблера SA1 (рис. 5.33). Встроенный блок питания, включающий в себя понижающий трансформатор TV1, выпрямитель на диодах VD1–VD4, стабилизатор DA1, фильтрующие конденсаторы C1–C3 и делитель напряжения R1 и R2, служит для питания цепей входов и выходов программируемого контроллера. Тумблеры SA2–SA9 обеспечивают подачу входных сигналов на контроллер. При замыкании контактов тумблеров на входы контроллера подается напряжение уровня логической единицы. При этом загорается

соответствующий светодиод VD11–VD18. Светодиоды VD5–VD10 включены на выходы контроллера и включаются при выводе логической единицы на соответствующий выход. Резисторы R3–R16 – токоограничивающие. Функции светодиодов VD11–VD18 продублированы на самом контроллере светодиодами Input, а светодиоды VD5–VD10 – светодиодами Output на контроллере.

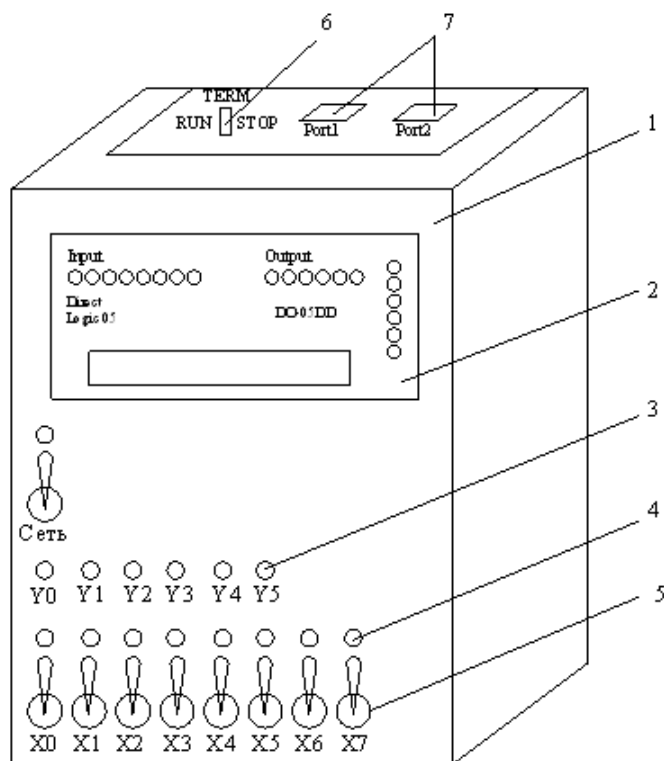


Рис. 5.32. Внешний вид контроллерного блока

При подготовке к лабораторной работе студент должен в соответствии с табл. 5.11 выбрать свой вариант задания для реализации на микроконтроллере D0-05DD.

Таблица 5.11

Номер бригады	1 и 5			2 и 6			3 и 7			4 и 8		
Член бригады	1	2	3	1	2	3	1	2	3	1	2	3
Номер варианта	1	2	3	4	5	6	7	8	9	10	11	12

Для заданного варианта необходимо разработать алгоритм его решения. В соответствии с заданием определиться с адресацией входных, выходных, вводимых промежуточных переменных и с адресацией таймеров и счетчиков. Подготовить программу для её ввода в ПЭВМ. В лаборатории запустить программу и убедиться в правильности её работы. При наличии ошибок в работе устранить их и продемонстрировать преподавателю работу правильно функционирующей программы.

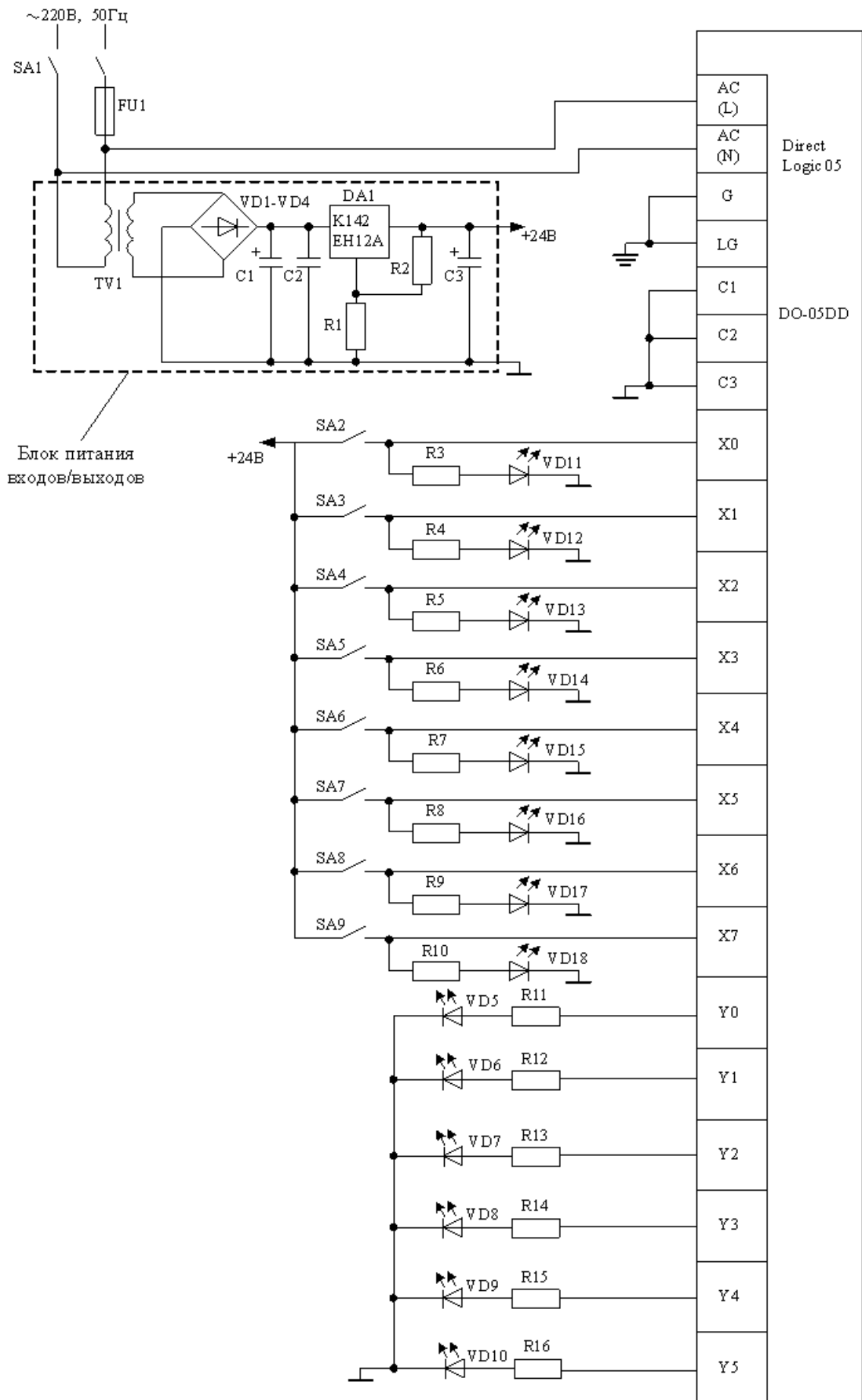


Рис. 5.33. Схема контрольного блока

*Вариант 1.* Запрограммировать управление замком двери офисного кабинета. Замок имеет электромагнит, при включении которого дверь открывается. На двери установлены 3 кнопки Кн1, Кн2 и Кн3. Нормальное положение двери закрытое (электромагнит отключен).

Для открытия двери необходимо осуществить осмысленные действия:

– нажать кнопку Кн1 и удерживать в нажатом состоянии. Начинает мигать лампочка. Отсчитать 2 мигания и кнопку Кн1 отпустить;

– нажать кнопку Кн2 и удерживать. Отсчитать 3 мигания лампочки и, не отпуская кнопку Кн2, нажать кнопку Кн3. Отсчитать еще 3 мигания лампочки. Отпустить обе кнопки Кн2 и Кн3.

Если все сделано правильно, то должен на 5 с включиться электромагнит, открывая замок. Если же требуемая последовательность не соблюдена, но было 3 нажатия кнопок в любой последовательности, то лампа начинает быстро мигать до нажатия на кнопку «Сброс», которая приводит схему в исходное состояние.

*Вариант 2.* Запрограммировать управление гирляндой из 5-ти ламп HL1...HL5. При нажатии на кнопку «Пуск» начинается загорание и погасание ламп (бег) в последовательности HL1, HL2, HL3, HL4, HL5, HL1, HL2, HL2, HL3 и т.д. Через 5 циклов бега в одном направлении направление бега изменяется на противоположное, т.е. на последовательность HL5, HL4, HL3, HL2, HL1, HL5, HL4 и т.д. Через каждые 5 с направление бега автоматически меняется на противоположное.

*Вариант 3.* С валом исполнительного механизма манипулятора связан импульсный датчик перемещения (при работе имитировать включением/отключением входа, например, Х0). За цикл работы манипулятор совершает поворот вокруг вертикальной оси на  $360^\circ$ , т.е. начинает работу из исходного положения и возвращается вновь в исходное положение. При повороте на  $360^\circ$  импульсный датчик выдает 20 импульсов.

Работа начинается по сигналу «Пуск цикла». После 5-ти импульсов манипулятор останавливается до поступления кратковременного сигнала «Деталь установлена», разрешающего дальнейшее движение. После 10-го импульса манипулятор останавливается до поступления сигнала «Конец обработки». Манипулятор вновь включается и движется до поступления 15-го импульса. Манипулятор останавливается в этой позиции для снятия детали на 5 с, вновь включается, и по 20-му импульсу он останавливается. Формируется сигнал «Конец цикла».

*Вариант 4.* Перекладыватель служит для перемещения листов со стола на постоянно вращающийся рольганг. Перекладыватель имеет подъемник листов, установленный на тележке. Подъемник представляет собой поперечину, перемещающуюся по вертикальным стойкам сверху вниз и обратно. Поперечина имеет на конце электромагнит для притягивания листов.

В исходном положении переключатель стоит в положении П0 над столом, а его подъемник в крайнем верхнем положении КВ. При поступлении листа на стол подается сигнал наличия листа, подъемник включается для движения вниз. При касании листа подъемник останавливается и включается электромагнит. Через 2 с включается подъемник на подъем, достигает положения КВ и останавливается. Включается тележка и перемещается в положение П1 над рольгангом, где останавливается и электромагнит отключается. Лист падает на рольганг. Через 2 с тележка движется в положение П0. Цикл повторяется при поступлении очередного сигнала наличия листа. Включение системы в работу – нажатием кнопки «Пуск».

*Вариант 5.* Автоматизировать механизм закрывания-открывания нагревательного колодца слитков обжимного прокатного стана.

Нагрев слитков осуществляется газом в колодце, который закрывается напольно-крышечной машиной. Крышка расположена на тележке и может опускаться и подниматься. Сама же тележка с поднятой крышкой может отъезжать в сторону, позволяя извлекать клещевым краном слитки из колодца.

При поступлении кратковременной команды «Открыть» механизм подъема крышки включается на подъем. Крышка поднимается до крайнего верхнего положения и тележка с крышкой отъезжает в крайнее открытое положение и останавливается.

При поступлении кратковременной команды «Закреть» тележка с крышкой возвращается в положение над колодцем и после ее остановки крышка опускается до крайнего нижнего положения, закрывая колодец.

Предусмотреть снижение скорости тележки при подходе ее к крайним положениям и снижение скорости опускания крышки при ее подходе к крайнему нижнему положению.

*Вариант 6.* Разработать информационное табло склада на 20 крупногабаритных пакета. Конвейер 1 доставляет пакеты в зону складирования. Фотоэлектрический датчик в конце конвейера 1 погрузки определяет сколько пакетов поступает на склад. Конвейер 2 транспортирует пакеты из зоны хранения в зону погрузки на автомобили для доставки клиентам. Фотоэлектрический датчик в конце конвейера 2 определяет сколько пакетов покидают зону складирования. Информационное табло должно отображать уровень заполнения склада:

- склад пустой;
- зона складирования не пуста;
- зона складирования заполнена на 50%;
- зона складирования заполнена на 80%;
- зона складирования заполнена полностью.

*Вариант 7.* Автоматизировать управление открыванием/закрыванием двери гаража. Есть только одна кнопка управления. Для открывания двери следует на-

жать кнопку, включается звонок, но движение на открытие двери начинается только при отпускании кнопки. При полном открытии двери привод двери отключается, но звонок звенит еще 2 с. Для закрывания двери аналогично следует нажать кнопку, включается звонок, но движение на закрытие двери начинается только при отпускании кнопки. При полном закрытии двери привод двери отключается, но звонок звенит еще 2 с. Если при закрытии двери встретилось препятствие, то привод двери реверсируется и дверь открывается. После устранения препятствия следует снова нажать и отпустить кнопку.

*Вариант 8.* Автоматизировать грузовой подъёмник библиотеки, обеспечивающий связь между подвальным книгохранилищем и читальными залами на 1 и 2 этажах. В подвале установлены кнопки Кп1 и Кп2 отправки из подвала соответственно на 1-ый и 2-ой этажи. На этажах установлены кнопки отправки подъемника с этажа в подвал.

На всех этажах установлены двери шахты. Сам подъёмник двери не имеет. Движение подъемника разрешается только при всех закрытых дверях шахты.

При приходе подъемника на заданный этаж кратковременно включается звонок, установленный в подъемнике. Оператор вручную открывает двери шахты на этаже нахождения подъемника, разгружает и загружает подъемник, закрывает вручную дверь шахты и нажимает кнопку отправки в подвал.

*Вариант 9.* Тележка перемещается по прямолинейному пути, имеющему 4 фиксированных положения П0...П3 где П0 и П3 – крайние положения, а П1 и П2 – промежуточные. При нажатии на кнопку «Пуск» тележка из положения П0 движется до положения П1, стоит в нем 1 с, затем движется в положение П3, где стоит в течение 3 с и возвращается в положение П2, стоит в нем 2 с и возвращается в положение П0. В положении П0 формируется сигнал «Конец цикла». Этот сигнал снимается при очередном нажатии кнопки «Пуск». Нажатие кнопки «Сброс» должно прервать выполнение цикла и вызвать движение тележки в положение П0 без формирования сигнала «Конец цикла».

*Вариант 10.* Необходимо автоматизировать перемещение тележки. Исходное положение тележки П0. По сигналу «Пуск» тележка быстро перемещается до положения П1, где останавливается на 2 с, в течение которых на неё укладывается заготовка. Грузеная тележка на пониженной скорости перемещается от положения П1 к положению П2, где в течение 4 с происходит выгрузка заготовки. Порожняя тележка быстро движется до П1, где в нее в течение 2 с вновь грузится очередная заготовка. Затем на пониженной скорости тележка перемещается в положение П2 в течение 4 с происходит выгрузка заготовки.

Затем тележка быстро перемещается от положения П2 к положению П0, где стоит до очередного сигнала «Пуск». После окончания цикла выдается световой сигнал «Конец цикла».



*Вариант 11.* Обеспечить автоматизацию станка для сверления платы. При нажатии на кнопку «Пуск» из крайнего верхнего положения П0 сверло движется в крайнее нижнее положение П1. По достижении П1, сверло возвращается в положение П0, где находится в течение 2 с. За это время плата чисто механическим путем перемещается на заданное расстояние, после чего движение сверла повторяется. В плате сверлятся 3 отверстия, после чего сверло остается в положении П0 до очередного нажатия на кнопку «Пуск». В этот промежуток времени выдается сигнал «Конец цикла».

*Вариант 12.* Тележка имеет три контролируемых положения П0, П1, П2. Положения П0 и П2 соответственно левое и правое положения, а П1 – среднее.

Исходное положение – П1. При нажатии на кнопку «Пуск» тележка движется вправо до положения П2, где останавливается и стоит в течение 2 с. Затем тележка включается для движения влево, останавливается на 2 с в положении П1, движется до положения П0, где останавливается. Через 3 с тележка возвращается в исходное положение П1 и формируется сигнал «Конец цикла».

## 5.10. Требования к отчёту

Отчет должен содержать:

- а) цель работы;
- б) условие задачи по варианту, принятые обозначения, логические функции для выходных и промежуточных переменных, при необходимости циклограмму работы оборудования;
- в) логические функции в адресах программируемого микроконтроллера;
- г) программу для программируемого микроконтроллера;
- д) методику экспериментальной проверки функционирования реализованной системы управления и результаты проверки;
- е) выводы по работе.

## 5.11. Контрольные вопросы

1. Почему полярность подключения напряжения для входов микроконтроллера не имеет значения, а для выходов соблюдение полярности обязательно?
2. Каково назначение специальных реле?
3. Сколько и какие языки программирования используются в микроконтроллере D0-06DD1?
4. Какое значение напряжения соответствует уровню логической единицы входов и выходов микроконтроллера?
5. Что означает фиксирование вывода контроллера командой «Set»?
6. Какие функции возлагаются на булевый стек при программировании на языке инструкций?
7. Каково назначение и использование контактов сравнения?

8. Какая разрядность ячеек V-памяти и аккумулятора?
9. Как осуществляется пересылка информации из одной ячейки V-памяти в другую ячейку V-памяти?
10. Каково назначение и сколько уровней имеет стек аккумулятора?
11. Какое принципиальное отличие накапливающего таймера от таймера с задержкой на включение?
12. Какое назначение предустановки (уставки) реверсивного счетчика UDC?
13. Что такое барабанный командоаппарат и для программирования каких процессов его целесообразнее использовать?
14. Какое принципиальное отличие стадийного программирования от программирования по лестничным диаграммам?

## ПРИЛОЖЕНИЯ

### ПРИЛОЖЕНИЕ А. АДРЕСАЦИЯ СИГНАЛОВ И КОМАНД ПРОГРАММИРУЕМЫХ КОНТРОЛЛЕРОВ SIMATIC S7-300

На рис. А.1 представлена максимальная конфигурация контроллера S7-300 на четырех стойках (контроллеры с CPU 312, 312IFM, 312C и 313 могут использоваться только для конфигураций с одной стойкой). Для монтажа на нескольких стойках требуются интерфейсные модули (IM), которые продолжают шину CPU на задней панели S7-300 от одной стойки к другой. CPU всегда находится на стойке 0. Передающий IM360 размещается на стойке 0, которая называется центральным устройством (CU). Остальные стойки называются устройствами расширения (EM). В них устанавливаются принимающие IM361. Длина соединительных кабелей типа 368 до 10 м.

Места размещения модулей в стойках называются слотами. В стойке 0 в слоте 1 всегда размещается блок питания, в слоте 2 – процессорный модуль CPU, в слоте 3 – интерфейсный модуль. На одной стойке не может быть более 11 слотов. В слотах 4...11 может быть размещено общим числом не более 8 модулей (сигнальных модулей SM, функциональных модулей FM, коммуникационных процессоров CP). Количество размещаемых модулей (SM, FM, CP) ограничено допустимым потреблением тока из шины на задней панели S7-300. Общее потребление тока на один ряд не может превышать 1,2 А.

Каждый вход и выход имеют абсолютный адрес, предопределённый конфигурацией аппаратуры. Если первый цифровой (дискретный) модуль установлен в слоте 4 стойки 0, то по умолчанию начальный адрес равен 0. Начальный адрес каждого следующего цифрового модуля увеличивается на 4.

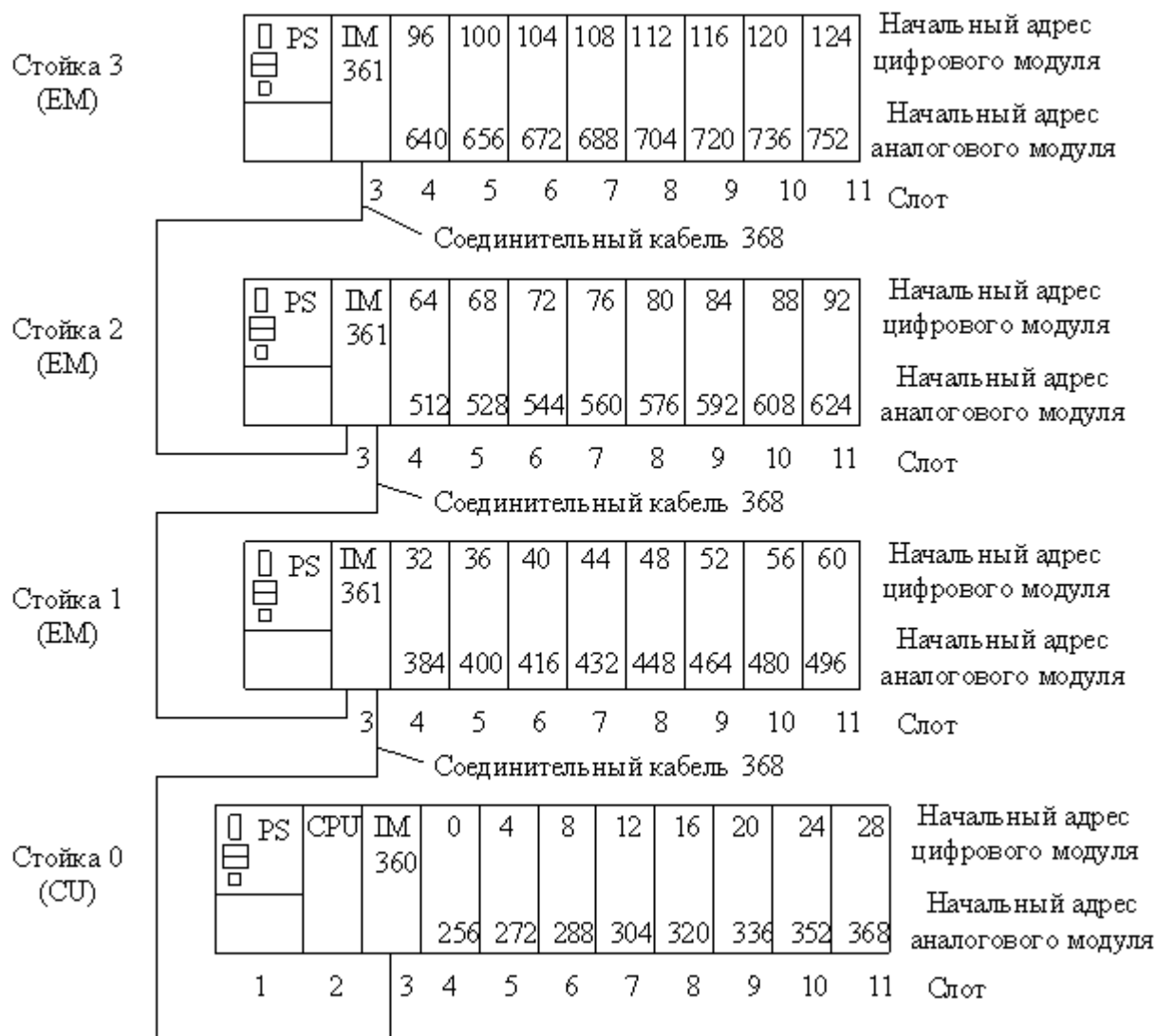


Рис. А.1. Максимальная конфигурация S7-300

Пусть, например, цифровой модуль ввода типа S7-321 DI32xDC24V на 32 канала ввода сигналов постоянного тока напряжением 24 В установлен в слоте 4 стойки 0. На рис. А.2 указано разделение входных каналов по байтам и в качестве примеров указаны адреса модуля ввода цифровых (дискретных) сигналов S7-321 DI32xDC24V, установленного в слоте 4 стойки 0 для четырех каналов ввода.

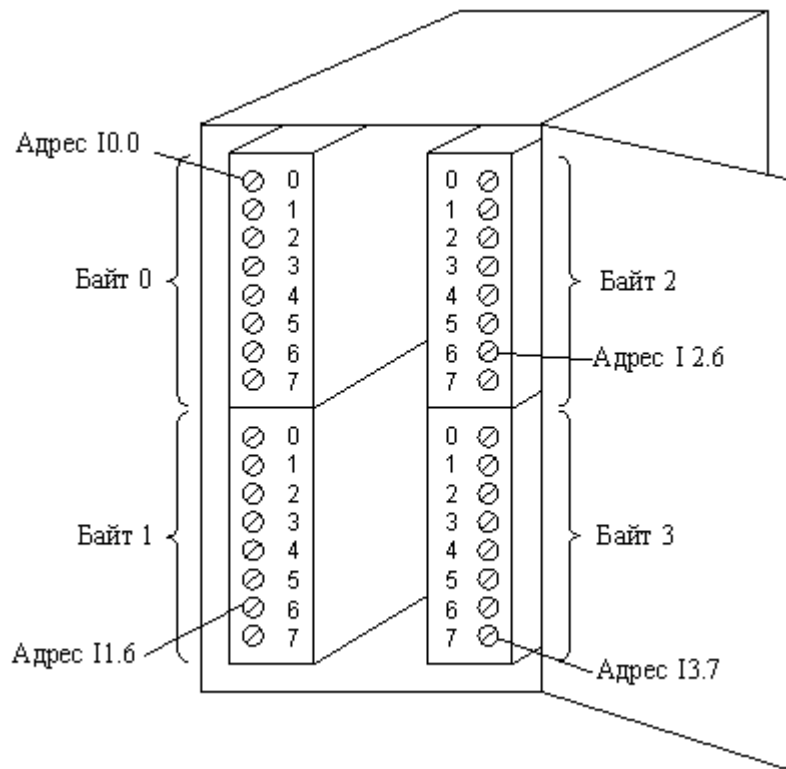


Рис. А.2. Принцип адресации каналов модуля ввода

На рис. А.3 дано разъяснение формирования адреса I1.6., канала ввода.



Рис. А.3. Формирование абсолютного адреса

По такому же принципу формируются адреса выходов Q, меркеров (внутренних переменных) M и локальных данных L. В табл. А.1 представлены идентификаторы адресов и диапазоны значений для битов (BOOL), байтов (BYTE), слов (WORD) и двойных слов (DWORD) для CPU314.

Таблица А.1

Идентификаторы адресов и диапазоны значений для CPU314

Объем информации	Входы	Выходы	Меркеры (промежуточные переменные)	Локальные данные
BOOL	I0.0...I127.7	Q0.0...Q127.7	M0.0 ...M255.7	L0.0...L509.7
BYTE	IB.0...IB127	QB.0...QB127	MB0.0 ...MB255	LB0.0...LB509
WORD	IW.0...IW126	QW.0...QW126	MW0.0 ...MW254	LW0.0...LW508
DWORD	ID.0...ID124	QD.0...QD124	MD0.0 ...MD252	LD0.0...LD506

Адресация каналов модулей аналоговых входов и выходов расположена в пункте 1.8 настоящего пособия. Эти каналы работают с байтами, словами и двой-

ными словами. Поэтому идентификаторами адреса входов являются переменный входной байт PIV, периферийное входное слово PIW и периферийное входное двойное слово PID. Для модулей аналогового выхода – соответственно, периферийный входной байт PQB, периферийное выходное слово PQW и периферийное выходное двойное слово PQD.

Перед программированием контроллера необходимо определиться, в какой последовательности (в каких слотах) устанавливаются модули в стойке (стойках). Затем необходимо полученную конфигурацию занести в конфигурационную таблицу (конфигурационное окно “HW Config”).

Пусть в центральном устройстве (стойке 0) размещаются следующие модули (рис. А.4):

- модуль питания PS-307;
- процессорный модуль CPU314;
- модуль ввода дискретных сигналов SM321 DI32xDC24V на 32 канала ввода сигналов постоянного тока напряжением 24 В;
- модуль вывода дискретных сигналов SM322 DO32xDC24V/0.5A на 32 канала вывода сигналов постоянного тока напряжением 24 В с допустимым током 0,5 А;
- модуль ввода/вывода дискретных сигналов SM323 DI16/DO16x24V/0.5A на 16 каналов ввода сигналов постоянного тока напряжением 24 В и 16 каналов вывода сигналов постоянного тока напряжением 24 В с допустимым током 0,5 А;
- модуль SM334 AI4/AO2x8/8bit на 4 канала ввода и 2 канала вывода аналоговых сигналов;
- модуль SM331 AI8x12bit на 8 каналов ввода аналоговых сигналов;
- модуль SM332 AO8x12bit на 8 каналов вывода аналоговых сигналов.

PS-307	CPU 314		SM321 DI32x DC24V	SM322 DO32x DC24V/ 0.5A	SM DI16/ DO16x 24V/0.5A	SM AI4/ AO2x 8/8bit	SM AI8x 12bit	SM AO8x 12bit
1	2	3	4	5	6	7	8	9

Слот

Рис. А.4. Размещение модулей контроллера в центральном устройстве (стойке 0)

При заполнении конфигурационной таблицы (табл. А.2) компьютер автоматически присваивает значения I Adress’ов и Q Adress’ов.

Таблица А.2

Slot	Modul	Order Number	MPI Adress	I Adress	Q Adress
1	PS307 2A	6ES7 307-1BA00-0AA0			
2	CPU314	6ES7 314-1AE04-0AB0	2		
3					
4	DI32xDC24V	6ES7 321-1BL00-0AA0		0...3	
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0			4...7
6	DI16/DO16x24V/0.5A	6ES7 323-1BL00-0AA0		8...9	8...9
7	AI4/AO2x8/8bit	6ES7 334-OCE01-0AA0		304...311	304...307
8	AI8x12bit	6ES7 331-7KF00-0AB0		320...335	
9	AO8x12bit	6ES7 332-SHF00-0AB0			336...351

Анализируя присвоенные в табл. А.2 автоматически I Adress'a и Q Adress'a и сопоставляя их с размещением модулей по рис. А.1 и рис. А.4, можно получить четкое представление об адресации каналов использованных в примере модулей.

Заносимый в конфигурационную таблицу Order Number (Заказной номер) нанесен на лицевой панели каждого модуля.

## **ПРИЛОЖЕНИЕ Б.**

### **ВАРИАНТЫ ТЕХНОЛОГИЧЕСКИХ ОБЪЕКТОВ УПРАВЛЕНИЯ**

#### **Б.1. Общие положения**

Б.1.1. Ниже представлены варианты технологических объектов управления (объектов автоматизации), записанные в ПЭВМ. Для каждого варианта представлена конструктивная схема объекта в том виде, в котором она вызывается на экран дисплея при выборе данного варианта. Так как на схеме показаны все необходимые датчики и исполнительные элементы, достаточные для функционирования системы в соответствии с поставленной задачей, поэтому обучающиеся должны пользоваться именно указанными на рисунках (на экране) обозначениями и конструктивными назначениями элементов.

Б.1.2. На экране монитора над изображением механизма представлены разряды (биты) команд на исполнительные механизмы и разряды (биты) сигналов датчиков. Они позволяют контролировать изменение команд и сигналов в процессе работы механизмов. Наличие команды или сигнала сопровождается появлением в соответствующем разряде 1, отсутствие команды или сигнала – 0.

Эту же (дублирующую) информацию получает пользователь при анализе рисунка объекта управления. В частности, если фон исполнительного элемента и датчика серый (синий, белый), то это соответствует отсутствию команды и сигнала. Если фон красный (желтый) – соответствует включенному состоянию.

В табл. Б.1 представлено соответствие буквенно-цифровых позиционных обозначений сигналов датчиков в описании вариантов задач с сигналами ПЭВМ, подаваемых в контроллер, и с их адресом в контроллере. В табл. Б.2 представлена аналогичная информация для команд управления исполнительными механизмами.

Для каждого варианта дается описание технологического процесса, последовательность работы механизмов и назначение датчиков технологической информации. Это не означает, что в каждом случае студенту необходимо создавать систему автоматизации объекта в полном объеме. Так как есть управляемые механизмы и датчики, контролируемые их положение, то можно задачу на автоматизацию сузить, например, исключив переходы на пониженную (или повышенную) скорость и т.д.

Учитывая ограниченное время выполнения лабораторной работы, в каждом варианте сформировано по три более простых подварианта задач, которые и выполняются при проведении лабораторной работы.

Б.1.3. Для задания малой скорости перемещения объекта необходимо формировать команды направления движения, например, Тл (движение влево) и Тп (движение вправо) и команду на снижение скорости Тм. Тогда, например, при одновременном наличии команд Тл и Тм движение будет происходить влево на малой скорости.



Таблица Б.1

Номер бита (датчика) в ПЭВМ	№1 Роботизированный комплекс	№2 Методическая печь	№3 Нагревательный колодец	№4 Сортировка листов металла	№5 Участок транспортировки труб	№6 Станок для сверления отверстий	№7 Линия химической обработки	№8 Упаковочная машина	№9 Лифт
0	К1	ПР0	ВП	Г	П0	КВ	П1	КЛТ	Д1
X1	К2	ПР1	ПС	Б	П1	П0	П2	КПТ	Д1в
X2	К3	ПР2	НП	КВ	П2	П1	П3	НИ	Д2н
X3	К4	ПР3	КВЛ	ПС	П3	П2	П5	П1	Д2
X4	К5	ПТ0	П0	ПМ	П4	П3	ПС	П2	Д2в
X5	К6	ПТ1	П1	КЛ	П5	П4	ПЭ	КЛ	Д3н
X6	К7	ПТ2	П2	П1	П6	КН	КВ	П3	Д3
X7	К8	ПТ3	КВП	П2	П7		ПН	П4	3
X8		П30		П3	П8		П4	КП	О
X9		П33		П4	П9		ПМ	НТ	КН
X10		Ф		П5	П10		ПБ	КВ	КВ
X11				П6	С1		КП	КН	
X12				П7	С2		КЛ	Д1	
X13				КП					
X14				КСЛ					
X15				КСС					
X16									

Таблица Б.2

Номер бита (команды) в ПЭВМ	№1 Роботизиро- ванный комплекс	№2 Методическая печь	№3 Нагревате- льный колодец	№4 Сортировка листов металла	№5 Участок транспорти- ровки труб	№6 Станок для сверления отверстий	№7 Линия химической обработки	№8 Упаковочная машина	№9 Лифт
У0	Вв	Рв	Вверх	Тл	Р4В	В	П	ТолП	В
У1	Вн	Рн	Вниз	Тп	Р3В	Н		ТолЛ	Н
У2	Л	Рм	Мк	Тм	Р2В	М	Л	ПодВн	ЗД
У3	П	Тв	Пр	Вп	Р1В	Шпм	В	ПодВв	ОД
У4	Зажим	Тн	Л	Нп	Р0В	Шпб	Н	Зажим	М
У5	Разжим	З0	Мг	Мп	ТМВ	-		ШтВв	Э1
У6	Тр1	З3		Э	ТМН	КЦ		ШтВн	Э2
У7	Тр2	Р1			ТВ			ТелП	Э3
У8	Тр3	Р3			ТН		Вр	ТелЛ	
У9								Тран	К1
У10							СБ		К2
У11									К3

## Б.2. ВАРИАНТ 1. РОБОТИЗИРОВАННЫЙ КОМПЛЕКС ТРАНСПОРТИРОВКИ ИЗДЕЛИЙ

Роботизированный комплекс (рис. Б.1) предназначен для переноса изделий с автоматизированных линий (транспортеров) 5 и 6 на транспортер 7.

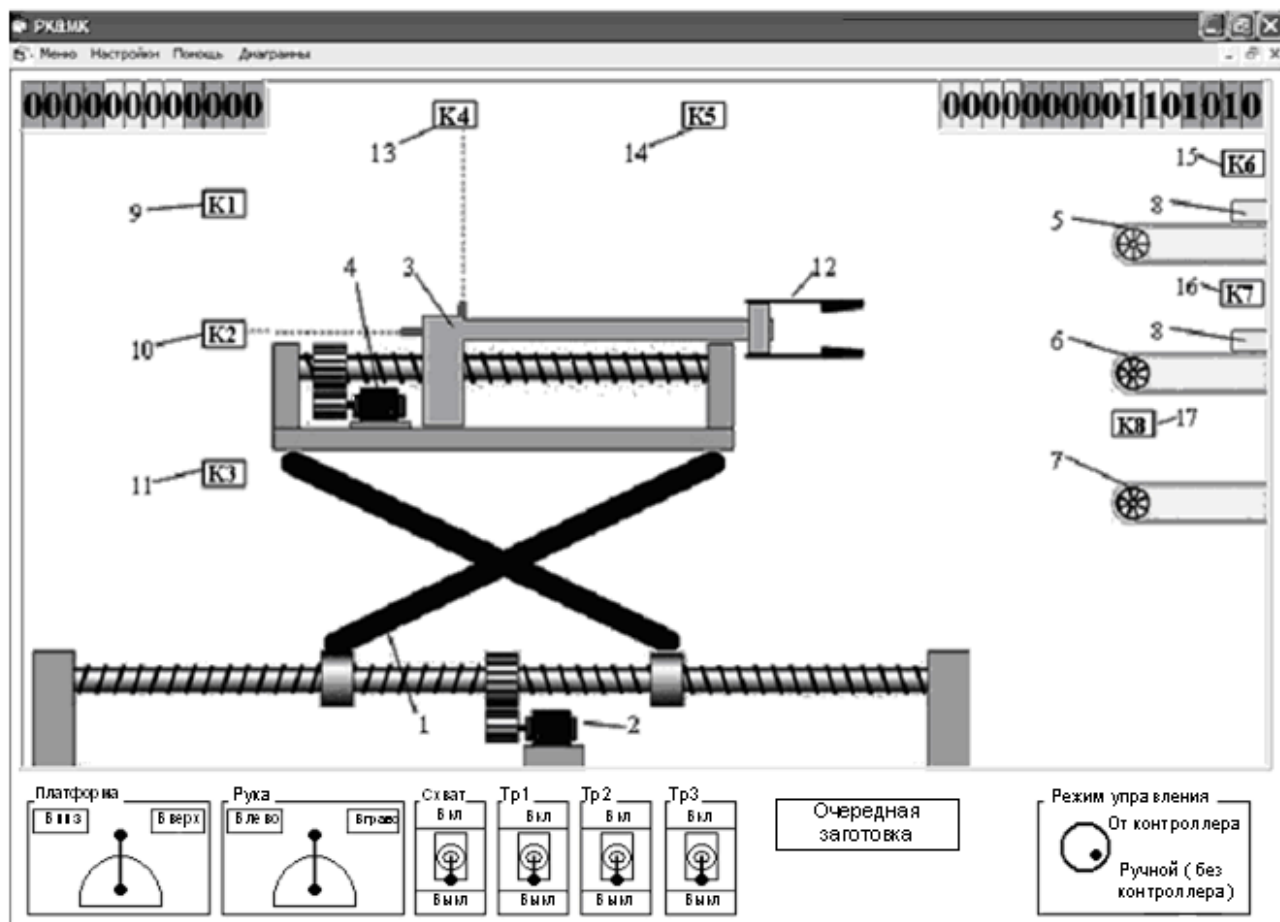


Рис. Б.1. Роботизированный комплекс транспортировки изделий

Непосредственно робот содержит механизм подъема 1, управляемый от двигателя 2, привод выдвижения руки 3 с двигателем 4, схват 12, управляемый пневмоклапаном. Вертикальные перемещения манипулятора контролируются тремя индуктивными датчиками 9, 10, 11 (датчики К1, К2, К3), установленные прямо напротив соответствующих линий подачи заготовки. По горизонтали привод руки контролируется двумя датчиками 13 и 14 (датчики К4, К5) крайних левого и правого положений. Наличие заготовок определяется срабатыванием датчиков 15, 16 и 17 (датчики К6, К7, К8). Перемещение заготовок выполняется нереверсивными транспортерами 5, 6 и 7, при этом два первых транспортера 5 и 6 включаются для подачи заготовок влево, а транспортера 7 – направо.

Цикл работы системы следующий. При нажатии кнопки «Очередная заготовка» (в нижней части экрана ПЭВМ) по закону случайных чисел появляется заготовка на подающем транспортере 5 или 6 и перемещается без участия манипуля-

тора до срабатывания датчика К6 или К7. Система должна включить соответствующий транспортер (команды Тр1 или Тр2) до отключения того же датчика. После этого происходит перемещение манипулятора по вертикали и горизонтали к данной заготовке (исходное положение схвата предполагается разжатое) до срабатывания датчиков К1/К2 и К5, зажим схвата (команда Зажим) и отход манипулятора назад до положения К4. Если на третьем транспортере 7 отсутствует заготовка, манипулятор перемещается вниз (Вн) и вправо (П) до срабатывания датчиков К3 и К5, в противном случае система ждет освобождения этого транспортера (отсутствия сигнала датчика К8). После установки руки манипулятора около третьего транспортера 7 схват разжимается (Разжим), и при наличии сигнала датчика К8 включается транспортер Тр3 и работает до исчезновения сигнала датчика К8. Далее цикл повторяется.

*Подвариант 1.1.* Необходимо автоматизировать работу двух механизмов – привода подъема манипулятора и привода выдвижения руки. В ручном режиме управления манипулятор устанавливается в положения К1, К4. В автоматическом режиме по команде «Пуск» с кнопки сенсорного монитора происходит перемещение только по одной координате: по горизонтали до положения К5, по вертикали до положения К3, по горизонтали до положения К4, по вертикали до положения К1 и далее повтор. Таким образом, манипулятор описывает по часовой стрелке в пространстве прямоугольник. При нажатии кнопки «Стоп» манипулятор устанавливается в положения К1, К4 и останавливается.

*Подвариант 1.2.* Необходимо автоматизировать работу двух механизмов – привода подъема манипулятора и схвата. При запуске манипулятор устанавливается в положение К2, К5, схват разжат. Затем происходит перемещение только по вертикали: вниз до положения К3 и затем зажать схват, по вертикали до положения К1 и затем схват разжать и так далее.

*Подвариант 1.3.* Автоматизировать работу двух механизмов привода подъема манипулятора и привода выдвижения руки. В ручном режиме управления манипулятор устанавливается в положения К3 и К4. При нажатии на кнопку «Пуск» в автоматическом режиме включается одновременно оба привода для движения до положений соответственно К1 и К5. При достижении этих положений формируются команды для движения обратно в положения К3 и К4. В К3 и К4 вновь формируются команды движения в положения К1 и К5 и т.д. Эти движения непрерывно продолжают до нажатия кнопки «Стоп».

### Б.3. ВАРИАНТ 2. УЧАСТОК ТРАНСПОРТИРОВКИ ЗАГОТОВОК В МЕТОДИЧЕСКУЮ ПЕЧЬ ДЛЯ ИХ НАГРЕВА ПЕРЕД ПРОКАТКОЙ (МЕТОДИЧЕСКАЯ ПЕЧЬ)

В методической печи 1 (рис. Б.2) осуществляется нагрев заготовок постоянных геометрических размеров до температуры проката.

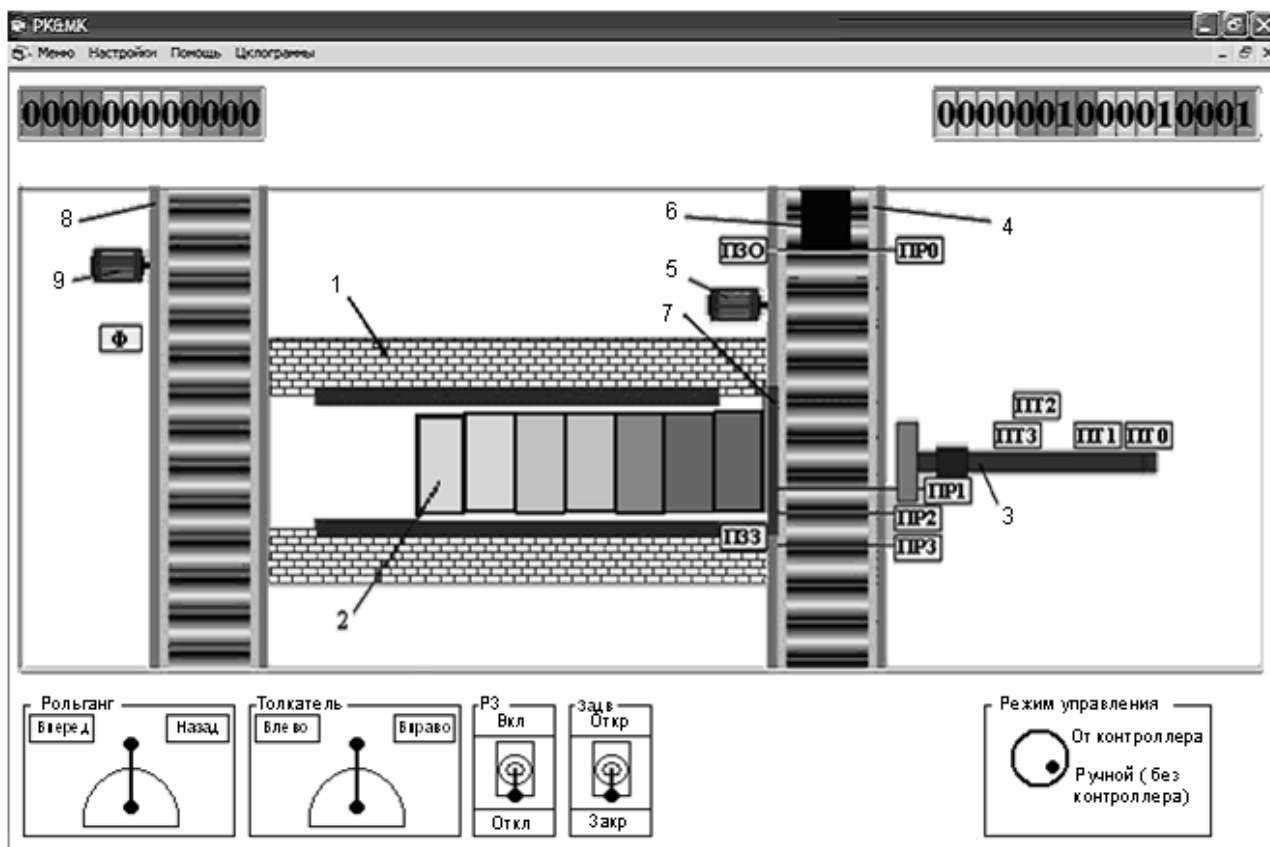


Рис. Б.2. Участок транспортировки заготовок в методическую печь

Вначале толкатель 3 находится в исходном (крайнем заднем) положении ПТ0, заслонка 7 переднего окна печи закрыта (есть сигнал датчика ПЗ3), заготовка перед печью отсутствует. При появлении заготовки 6 в положении ПР0 подающего рольганга 4 (Р1) включается рольганг Р2 вперед (команда Рв) и заготовка движется к переднему окну печи. При достижении передним концом заготовки положения ПР1 осуществляется снижение скорости рольганга Р2 до ползучей скорости (есть команды Рв и Рм), с которой осуществляется движение заготовки до положения ПР2. Рольганг Р2 отключается. Если по инерции передний конец дошел до положения ПР3 или перешел его, то рольганг Р2 включается для движения назад на ползучей скорости (команды Рн и Рм), движение сразу прекращается, если заготовка уйдет из положения ПР3.

Заготовка стоит перед печью до прихода сигнала (запрос) с кнопки на экране сенсорного монитора на выдачу заготовки из печи. В этом случае включается толкатель вперед (команда Тв) и движется до положения ПТ1. В этом положении

толкателя включается двигатель заслонки 7 на ее открытие (команда 30). Заслонка уходит из закрытого положения заслонки ПЗЗ и при достижении открытого положения заслонки ПЗО останавливается. Вновь включается толкатель для движения вперед. При своем движении толкатель сталкивает заготовку с рольганга и подает ее в печь 1, передвигая при этом все нагреваемые заготовки в печи. В режиме работы с полной загрузкой печи толкатель должен проделать путь, при котором крайняя заготовка выталкивается из печи и по склизу попадает на отводящий рольганг 8 (РЗ). При этом засвечивается фотоэлектрический датчик Ф, при срабатывании которого включается отводящий рольганг (РЗ), транспортируя заготовку к прокатному стану, а толкатель реверсируется (Тн) и движется в исходное положение. При прекращении засветки датчика Ф отводящий рольганг останавливается.

Если идет еще только первоначальная загрузка печи и при движении толкателя вперед крайняя заготовка в принципе не может быть вытолкнута на отводящий рольганг, то толкатель реверсируется при достижении им положения ПТЗ.

Когда толкатель при своем ходе назад (команда Тн) проходит положение ПТ2, формируется команда на закрытие (ЗЗ) заслонки. Она закрывается до тех пор, пока заслонка не придет в положение ПЗЗ.

При остановке толкателя в положении ПТ0 возможна подача к печи очередной заготовки.

Рольганг Р2 приводится в движение двигателем 5. Привод заслонки условно не показан. Привод отводящего рольганга приводится в движение двигателем 9. Толкатель приводится в движение двигателем, который условно не показан.

*Подвариант 2.1.* Необходимо автоматизировать работу двух механизмов - толкателя и задвижки. При поступлении с экрана сенсорного монитора кратковременной команды «Запрос оператора» электропривод толкателя включается и из исходного положения ПТ0 толкатель движется к промежуточному положению ПТ1, где останавливается. В этот момент начинает открываться заслонка, и после ее открытия толкатель вновь движется вперед. При достижении толкателем положения ПТ3 происходит реверс и он возвращается в исходное состояние ПТ0. Как только толкатель при движении назад проходит положение ПТ2, заслонка закрывается. Цикл повторяется при повторном нажатии на кнопку «Запрос оператора» на экране сенсорного монитора.

*Подвариант 2.2.* При наличии заготовки в положении ПР0 и нажатии на экране сенсорного монитора кнопки «Пуск» происходит включение рольганга Р2 и движение заготовки до положения ПР2, где и останавливается. По сигналу со следующей кнопки сенсорного монитора осуществляется возвращение заготовки на повышенной скорости рольганга Р2 до прекращения воздействия на датчик ПР1. Происходит останов рольганга. Через 0.5 с заготовка возвращается в положение ПР2. Формируется сигнал «Конец цикла».

*Подвариант 2.3.* При наличии заготовки в положении ПР0 и нажатии на экране сенсорного монитора кнопки «Пуск» включается рольганг Р2 и сразу открывается заслонка. При достижении заготовкой положения ПР2 рольганг останавливается и включается привод толкателя. При достижении толкателем положения ПТ3 привод толкателя реверсируется и толкатель возвращается в исходное положение ПТ0. При движении назад и проходе толкателем положения ПТ1 подается команда на закрытие заслонки.

#### **Б.4. ВАРИАНТ 3. УЧАСТОК НАГРЕВАТЕЛЬНОГО КОЛОДЦА ОБЖИМНОГО ПРОКАТНОГО СТАНА**

Нагрев слитков перед прокаткой на обжимном прокатном стане осуществляется в нагревательных колодцах, отапливаемых смесью доменного и коксового газов или природным газом. Колодец закрывается крышкой. Для открытия и закрытия колодца крышкой существует напольно-крышечная машина (рис. Б.3), включающая в себя два механизма: механизм подъема/опускания крышки и механизм перемещения тележки, на которой располагается крышка с механизмом подъема/опускания.

При нагреве слитков 1 до температуры 1200 °С к колодцу подаются слитковоз и клещевой кран (на рис. Б.3 не показаны). По сигналу «Открыть» с кнопки сенсорного монитора включается (Вверх) двигатель 4 (Д1) механизма подъема крышки колодца, установленный на тележке 3. Двигатель Д1 через механическую передачу осуществляет подъем крышки 2 до верхнего положения ВП, контролируемого соответствующим датчиком. Затем включается двигатель 5 (Д2) и тележка, приводимая в движение указанным двигателем, движется в позицию ожидания П2 (по рис. Б.3 движение вправо (Пр)) и стоит там до тех пор, пока с сенсорного монитора не поступит сигнал «Закрыть». В этом случае тележка движется влево (команда Л). При движении влево в положении П1 осуществляется снижение скорости движения тележки до ползучей скорости (команда Мт), с которой тележка движется до положения ПО. В положении ПО происходит останов тележки и осуществляется движение крышки вниз (команда Вниз). При наличии сигнала с датчика ПС при движении вниз происходит снижение скорости опускания крышки до ползучей (команда Мк), с которой происходит затем движение до полного закрытия колодца. Нижнее положение крышки контролируется датчиком НП (закрыт колодец).

При загрузке нагревательного колодца слитками все операции осуществляются аналогично, только вместо слитковоза подается состав со слитками.

В программе ПЭВМ предусмотрена остановка движения тележки при достижении ею крайних положений КВЛ и КВП (конечный выключатель левый и конечный выключатель правый соответственно) независимо от режима работы напольно-крышечной машины.

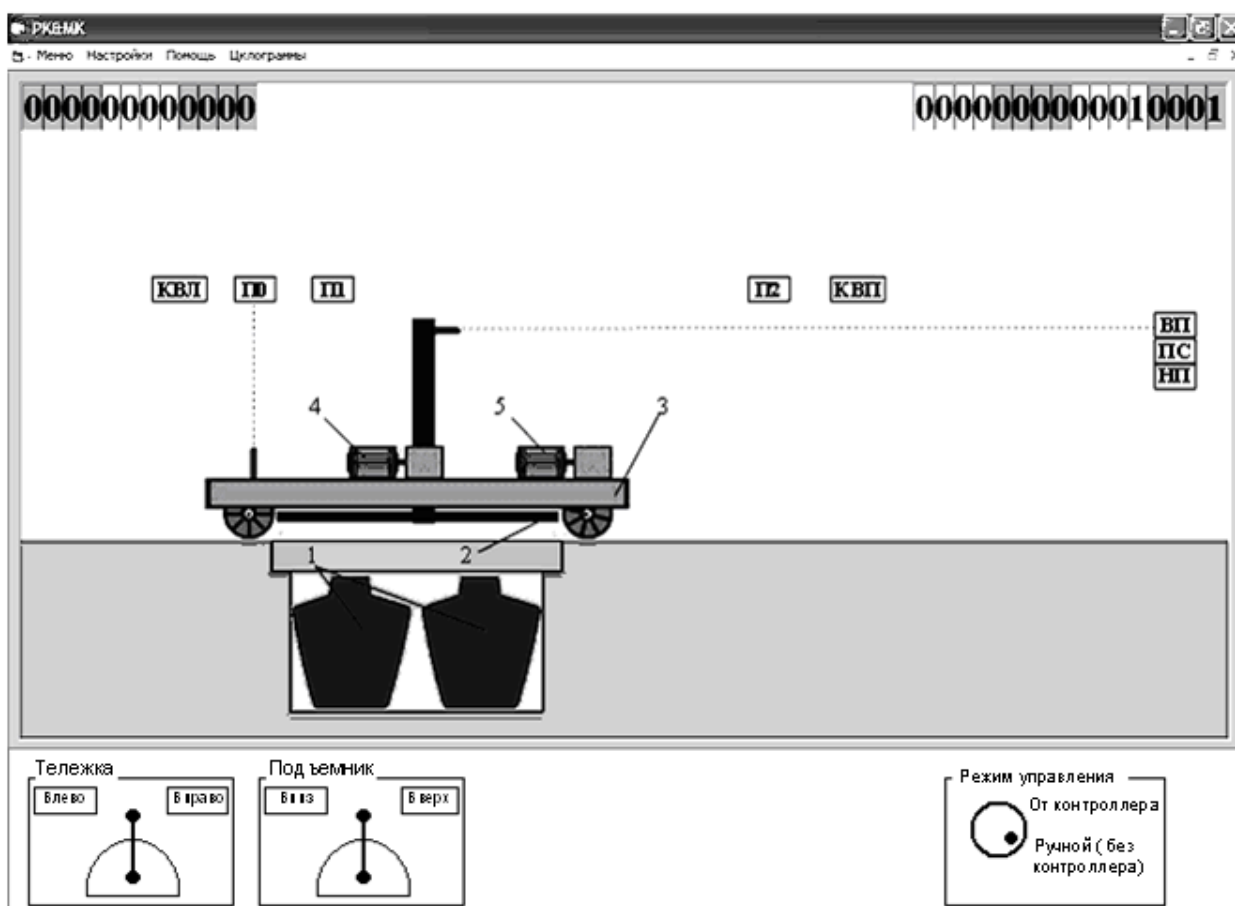


Рис. Б.3. Механизм управления крышкой нагревательных колодцев

*Подвариант 3.1.* Автоматизировать работу механизма управления крышкой нагревательных колодцев. При подаче сигнала «Пуск» (с кнопки на экране сенсорного монитора) открывается крышка колодца и тележка движется вправо до П2, где через 1 с происходит реверс. В положении П1 тележка переходит на малую скорость и возвращается в исходное состояние П0. Крышка колодца закрывается. При достижении крышкой положения НП цикл заканчивается.

*Подвариант 3.2.* При первом нажатии на кнопку «Пуск» открывается крышка нагревательного колодца до положения ВП, где останавливается на 2с и возвращается в исходное положение НП. Затем крышка доходит до положения ПС, где останавливается на 3с и возвращается в исходное положение. Движение «Вниз» от положения ПС к положению НП осуществляется на медленной скорости. Далее цикл непрерывно повторяется до нажатия на кнопку «Стоп».

*Подвариант 3.3.* В режиме ручного управления «Без контроллера» поставить крышку колодца в крайнее верхнее положение ВП. Установить режим управления «От контроллера». После первого нажатия на экране сенсорного монитора на выбранную кнопку «Пуск» тележка движется вправо до П2, где стоит 2 с. После окончания выдержки времени тележка перемещается влево до положения П1, пе-



реходит на ползучую скорость и подходит к П0, где останавливается до очередного нажатия кнопки «Пуск». Если в процессе работы нажималась кнопка «Стоп», то движение тележки прекращается, но если затем нажать кнопку «Пуск», то движение тележки продолжается с прерванного места.

### Б.5. ВАРИАНТ 4. УЧАСТОК СОРТИРОВКИ И ПАКЕТИРОВАНИЯ ГОДНЫХ И БРАКОВАННЫХ ЛИСТОВ МЕТАЛЛА

Обработанные и нарезанные на мерные длины стальные листы металла проходят через устройство контроля качества, которое выдает информацию о качестве листа. При качественном (годном) листе выдается сигнал годности Г, при бракованном листе выдается сигнал брака Б (рис. Б.4).

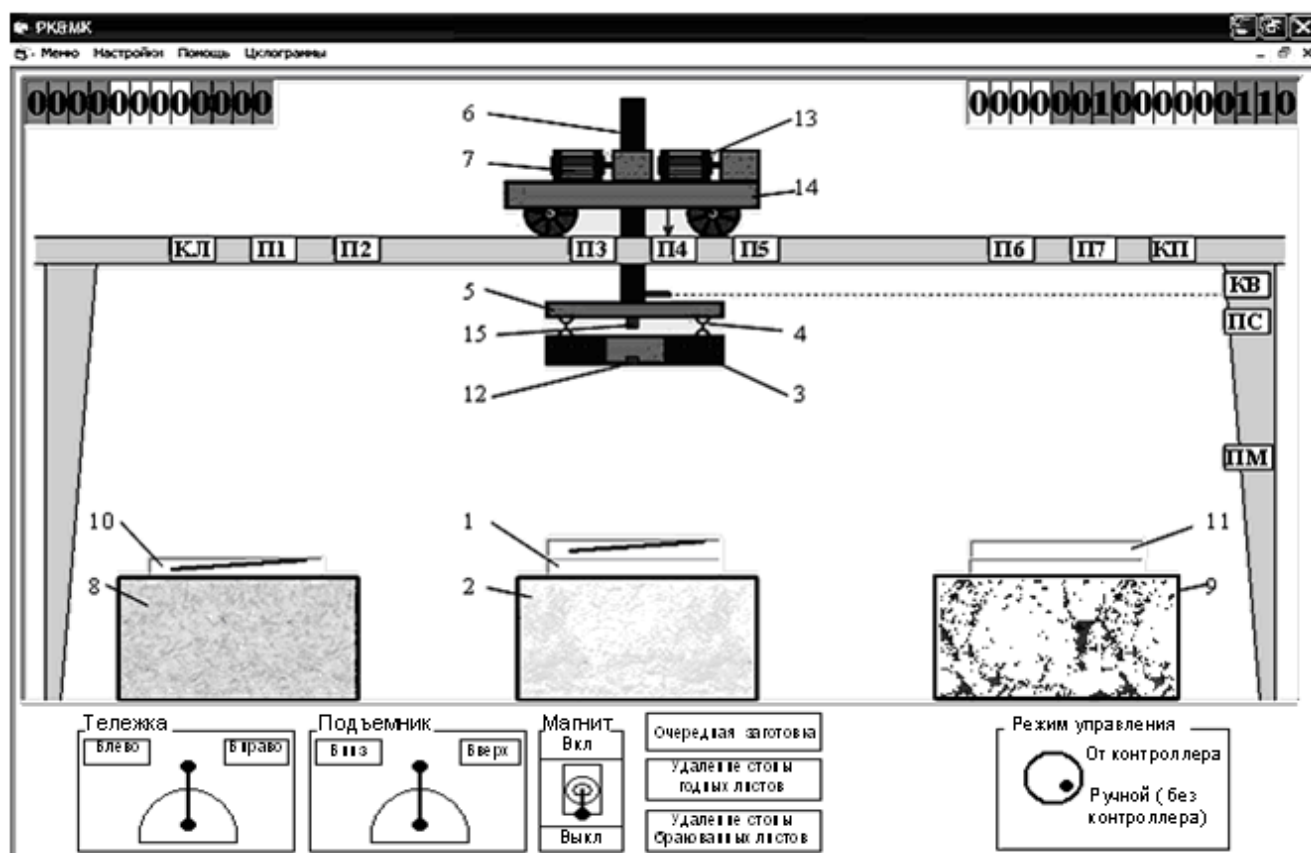


Рис. Б.4. Участок сортировки и пакетирования листов

Лист 1 подается на стол 2 раскладывателя при нажатии на кнопку «Очередная заготовка» в нижней части экрана монитора. Захват и перенос листа со стола осуществляется электромагнитом 3, подвешенном цепями 4 к траверсе 5. Если лист захвачен (есть сигнал датчика касания листа 12 (КСЛ)), то включается двигатель 7 (Д1) подъемника и осуществляется подъем (Вп) штанги 6, а следовательно, и электромагнита с листом. Подъем происходит до срабатывания датчика крайнего верхнего положения КВ. После отключения двигателя Д1 включается двигатель 13 (Д2) перемещения тележки 14 подъемника. Тележка движется влево (Тл) к

столу 8, если лист бракованный, или вправо (Тп) к столу 9, если лист годный. Тележка разгоняется до заданной скорости. При приближении к столу пакетирования по сигналу датчика П2 (или П6) происходит снижение скорости движения тележки до ползучей скорости (Тм) и по сигналу датчика П1 (или П7) происходит отключение двигателя Д2.

Подъемник опускается (Нп) для укладки листа в стопу. При соприкосновении листа со стопой его опускание с электромагнитом прекращается, но опускание траверсы 5 продолжается. На траверсе закреплен датчик касания стопы 15 (КСС). При срабатывании этого датчика прекращается опускание подъемника и снимается питание с электромагнита. Подъемник возвращается в положение КВ.

Вновь включается привод тележки раскладывателя и тележка движется в положение П4 над столом 2. По сигналу датчиков П3 или П5 (в зависимости от направления подхода к положению П4) происходит снижение скорости движения тележки до ползучей скорости (Тм). По сигналу датчика П4 происходит останов тележки.

При поступлении очередного листа и наличии сигнала о его качестве из положения КВ подъемник опускается (Нп) до положения ПМ, в котором осуществляется снижение скорости подъемника до ползучей скорости (Мп), с которой подъемник опускается до касания листа (до срабатывания датчика КСЛ). Происходит включение электромагнита (Э) и цикл работы повторяется.

Необходимо предусмотреть формирование сигналов, выдаваемых на сигнальные лампы на экране сенсорного монитора, о переполнении стоп бракованных 10 и годных 11 листов. Максимальная высота стоп годных и бракованных листов одинаковая и контролируется датчиком переполнения ПС.

На рис. Б.4 принята некоторая условность: датчики КВ, ПС и ПМ изображены расположенными у правой стойки раскладывателя. На самом же деле они расположены на тележке 14 и перемещаются вместе с ней. Воздействие на них осуществляется флажком, закрепленным на штанге 6.

В программе ПЭВМ предусмотрено случайное генерирование информации годного или бракованного листа. Очередной лист на столе 2 появляется при нажатии кнопки «Очередная заготовка» в нижней части экрана монитора. Уборка стоп бракованных и годных листов осуществляется нажатием соответственно кнопок «Удаление стопы бракованных листов» и «Удаление стопы годных листов» в нижней части экрана монитора.

Начало автоматической работы возможно при наличии сигнала КВ и расположении тележки в положении П4. В указанные положения механизмы устанавливаются при вызове программы рассматриваемого варианта или приводятся в указанные положения в ручном режиме управления «Без контроллера».

В программе ПЭВМ предусмотрена остановка движения подъемника при достижении им положения КВ и остановка движения тележки при достижении ею крайних положений КЛ и КП независимо от режима работы установки.

*Подвариант 4.1.* Требуется автоматизировать процесс перемещений тележки раскладывателя. Если траверса не находится в верхнем положении, то необходимо осуществить подъем траверсы в ручном режиме работы. Исходное положение тележки П4. При первом нажатии на кнопку «Пуск» (кнопка на экране сенсорного монитора) тележка из положения П4 движется в положение П2, стоит в нем в течение 2 с и возвращается в положение П4. При втором нажатии на кнопку «Пуск» тележка из положения П4 движется в положение П7, стоит в нем в течение 3 с и возвращается в положение П4.

В момент достижения тележкой положения П4 включается сигнал «Конец цикла», который выключается очередным сигналом «Пуск» или подачей сигнала «Сброс».

*Подвариант 4.2.* Требуется автоматизировать процесс перемещения листа со стола раскладывателя (см. рис. Б.4) на правый стол, независимо от качества листа.

Из исходного состояния при наличии листов на столе (тележка находится в положении П4, подъемник в верхнем положении КВ) по команде «Пуск» (кнопка на экране сенсорного монитора) подъемник опускается вниз на ползучей скорости до момента срабатывания датчика касания листа КСЛ. При срабатывании этого датчика выключается опускание подъемника и включается питание электромагнита, тем самым происходит захват листа. Далее на повышенной скорости осуществляется подъем штанги и электромагнита с листом до верхнего положения КВ. Затем тележка на ползучей скорости перемещается из положения П4 в положение П7, где останавливается.

Подъемник опускает лист на стол 9 и оставляет его там (опускание осуществляется до срабатывания датчика КСС). Затем подъемник возвращается в исходное состояние КВ, а тележка в положение П4.

При возвращении раскладывателя в исходное состояние загорается сигнальная лампа «Конец цикла» на экране сенсорного монитора. Повторение цикла – при повторном нажатии кнопки «Пуск».

*Подвариант 4.3.* Тележка стоит в положении П4, подъемник в верхнем положении КВ. По команде «Пуск» (с экрана сенсорного монитора) тележка движется до положения П1, затем реверсируется, движется до П4 стоит 3 с, движется до положения П7 и возвращается в положение П4, в котором формируется сигнал «Конец цикла». Перед остановками обеспечивается переход на медленную скорость.

## **Б.6. ВАРИАНТ 5. УЧАСТОК ТРАНСПОРТИРОВКИ ТРУБ БОЛЬШОГО ДИАМЕТРА**

Транспортировочное устройство трубоэлектросварочного участка (рис. Б.5) производит передачу труб с линии сварки (рольганги Р3 и Р4) на линию отделки (рольганги Р1 и Р2).

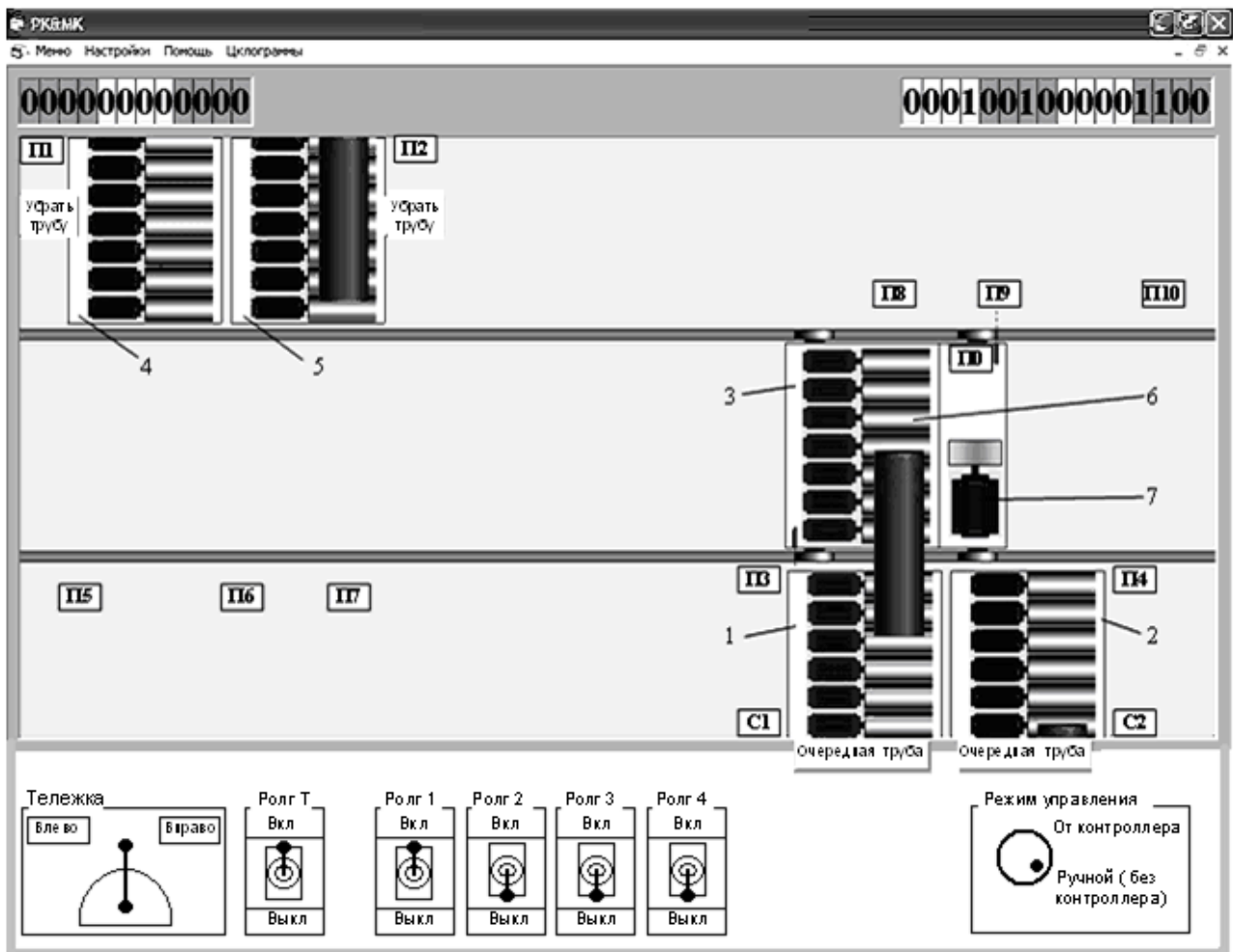


Рис. Б.5. Участок транспортировки труб большого диаметра

После сварки на трубоэлектросварочных станах трубы подаются для дальнейшей транспортировки рольгангами 1 (P3) и 2 (P4). К линии отделки трубы транспортируются передвижным рольгангом, представляющим собой приводную тележку 3 (Т) с рольгангом 6 (P0) и двигателем 7 для приема и выдачи труб. Команды на движение тележки вперед (влево) обозначены ТМВ (медленно) и ТВ (быстро), а на движение тележки назад (вправо) соответственно ТМН и ТН. Команды на включение рольгангов обозначены в соответствии с номерами рольгангов следующим образом: P0В, P1В, P2В, P3В, P4В.

Передача труб с линии сварки выполняется с той же очередностью, с которой они поступают на рольганги P3 и P4 (если труб на рольгангах P3 и P4 нет, то тележка всегда останавливается у P4). Тележка Т останавливается против очередного выходного рольганга P3 или P4, включаются оба рольганга (P0 и P3 или P4), и труба перемещается на передвижной рольганг P0 до положения П0. После остановки трубы на рольганге P0 включается двигатель привода тележки Т и она транспортирует трубу к рольгангам линии отделки.

Прием труб с передвижного рольганга в отделку производится всегда на рольганг 5 (P2), если он свободен. Если рольганг P2 занят, то прием осуществля-

ется свободным рольгангом 4 (P1). При занятых рольгангах P1 и P2 тележка останавливается у рольганга P2.

После останова тележки передвижного рольганга против выбранного рольганга линии отделки включается рольганг P0 и рольганг линии отделки (P1 или P2). После ухода трубы с рольганга P0, о чем свидетельствует достижение трубой положений П1 или П2 на рольгангах линии отделки, тележка направляется к выходным рольгангам линии сварки и цикл работы повторяется.

Движение тележки между рольгангами линии сварки и линии отделки осуществляется на повышенной скорости с переходом на ползучую скорость перед выбранным рольгангом. Если движение идет к рольгангу P2, то переход на ползучую скорость происходит в положении П7, если же к рольгангу P1, то на ползучей скорости осуществляется перемещение от положения П6 до П5. При движении к рольгангу P3 переход движения на ползучую скорость осуществляется в положении П8, если же идет движение к рольгангу P4, то на ползучей скорости происходит перемещение от положения П9 до П10.

В положениях П5, П6, П9 и П10 осуществляется совмещение оси передвижного рольганга с осями соответствующих рольгангов.

Сигналы С1 и С2 о выходе трубы со сварочных станков подаются кнопками «Очередная труба» у изображений рольгангов P3 и P4.

Команды на удаление труб с рольгангов P1 и P2 подаются кнопками «Убрать трубу» у изображения рольгангов P3 и P4.

Приводы всех рольгангов нереверсивные и нерегулируемые.

*Подвариант 5.1.* Тележка находится в исходном состоянии, то есть стоит у рольганга P3 (положение П9) и трубы на ней нет. Также отсутствуют трубы на рольгангах P3 и P4.

Если труба вначале появляется от первого сварщика (С1), то должен включиться рольганг P3 и затем P0 (см. условие работы в основном варианте). Как только труба полностью заходит на тележку, должна загореться сигнальная лампочка на экране сенсорного монитора «Загрузка закончена».

Если же труба появляется вначале от второго сварщика (С2), то труба полностью заходит на рольганг P4 (до П4), затем тележка на ползучей скорости перемещается к рольгангу P4 и осуществляется загрузка трубы на тележку. В конце загрузки должен появиться сигнал «Загрузка закончена».

*Подвариант 5.2.* Из исходного положения (П9 или П10) по команде «Пуск» (кнопка на экране сенсорного монитора) тележка ускоренно перемещается до П7, переходит на ползучую скорость и доходит до положения П6, где стоит в течение 3с. После этого тележка возвращается к рольгангу P3, если на нем есть труба и эта труба появилась раньше (то есть она первоочередная), чем труба на рольганге P4. Если же первоочередной является труба на рольганге P4, то тележка должна идти к этому рольгангу. Если труб на рольгангах P3 и P4 нет, то тележка должна занять положение у рольганга P3 (П9).

*Подвариант 5.3.* При первом нажатии на кнопку «Пуск» (кнопка на экране сенсорного монитора) тележка должна из исходного положения (П9) переместиться к рольгангу Р2 (П6), при этом переход от П7 к П6 должен осуществляться на пониженной скорости. У рольганга Р2 тележка должна стоять 3 с и затем возвратиться в исходное положение (П9). Переход от П8 к П9 осуществляется также на пониженной скорости.

При втором нажатии на кнопку «Пуск» тележка осуществляет переход к рольгангу Р1 (от положения П6 до П5 на пониженной скорости), стоит также 3 с у рольганга Р1 и возвращается в исходное положение. Последующие нажатия кнопки «Пуск» не вызывают никаких действий до тех пор, пока не будет нажата кнопка «Сброс» на экране сенсорного монитора.

## **Б.7. ВАРИАНТ 6. СТАНОК ДЛЯ СВЕРЛЕНИЯ ГЛУБОКИХ ОТВЕРСТИЙ**

Сверление глубоких отверстий связано с периодическим выведением сверла 3 из изделия 4 для извлечения стружки (рис. Б.6).

Конструктивно станок состоит из стойки 1, по направляющим которой перемещается шпиндельная бабка 2. Привод инструмента 3 осуществляется от электродвигателя 9. Стойка установлена на основании 6. На основании устройствами 5 осуществляется зажим обрабатываемой детали 4. Подача шпиндельной бабки (ШБ) осуществляется ходовым винтом 8, приводимым в движение двигателем 10 через редуктор 7.

Контроль положения ШБ осуществляется датчиками КВ, КН, П0...П4. Датчики КВ и КН ограничивают крайнее верхнее и крайнее нижнее положения соответственно. Датчики П0...П4 контролируют промежуточные положения ШБ при сверлении детали.

В автоматическом режиме работы последовательность обработки следующая. ШБ находится в положении П0 (датчик П0 светится красным цветом). На экране сенсорного монитора нажимается кнопка, которой присвоена функция кнопки «Пуск». Включаются двигатели 9, 10. ШБ должна быстро идти к детали. В положении П1 двигатель переключается на пониженную рабочую подачу, с которой идет перемещение ШБ до положения П2. В положении П2 осуществляется реверс двигателя 10 и ШБ движется вверх до положения П1. В положении П1 вновь реверсируется двигатель 10 и ШБ ускоренно перемещается до положения П2, в котором скорость снижается до рабочей скорости подачи.

Совершается сверление до прихода ШБ в положение П3. Из положения П3 вновь поперечина ускоренно движется в положение П1, из которого ускоренно возвращается в положение П3 и затем на рабочей подаче осуществляется перемещение ШБ от положения П3 до положения П4. Нахождение ШБ в положении П4 свидетельствует о завершении сверления отверстия. Из положения П4 ШБ ускоренно перемещается в положение П0, где необходимо сформировать сигнал «Конец цикла», которым на экране монитора должна быть включена сигнальная лампа.

Для смены детали необходимо нажать кнопку «Смена детали» в нижней части экрана монитора.

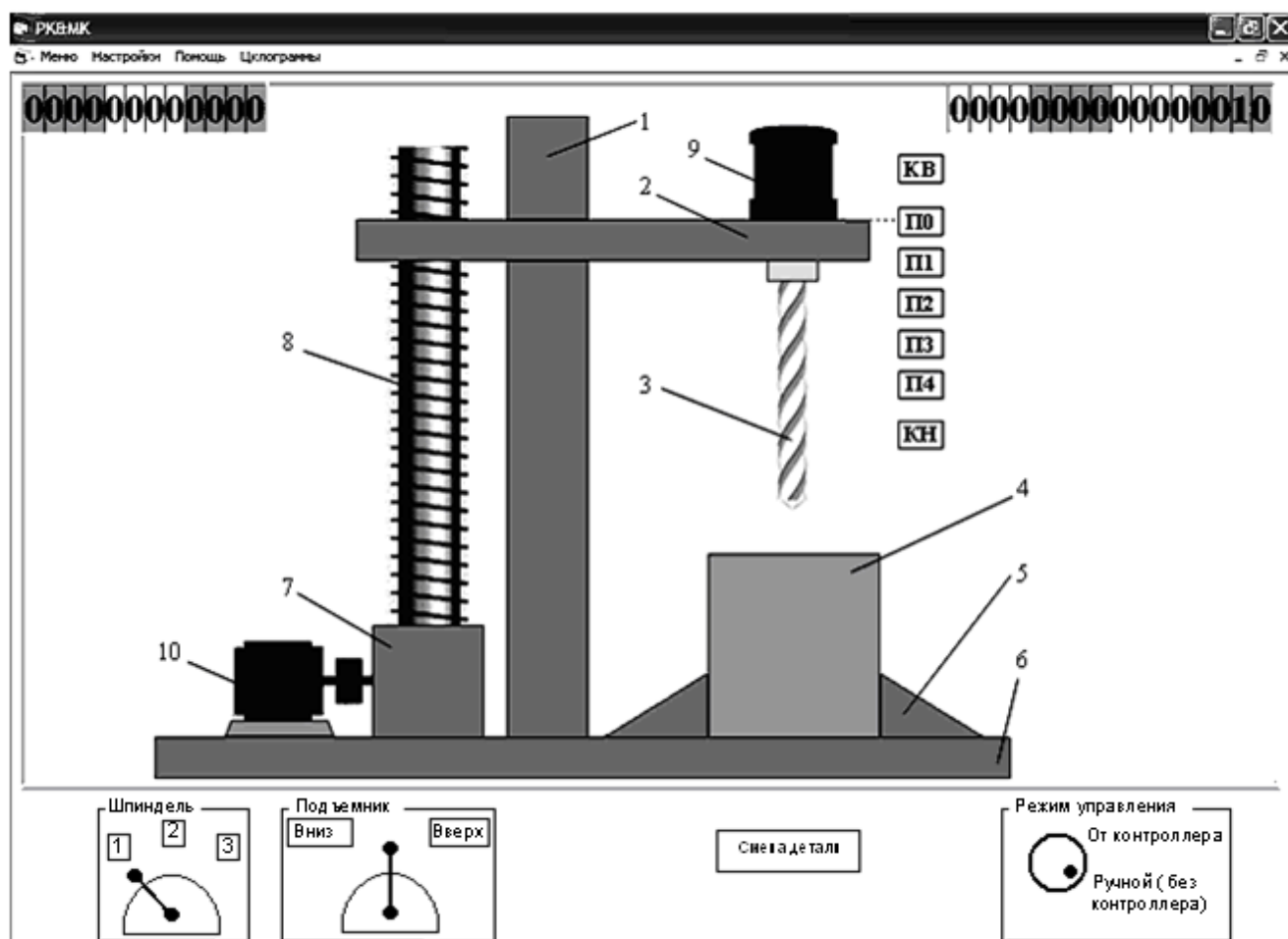


Рис. Б.6. Станок для сверления глубоких отверстий

В соответствии с табл. Б.2 введены следующие обозначения выходных команд: движение ШБ вниз – Н; вверх – В; снижение скорости – М; вращение шпинделя (сверла) медленно – Шпм; вращение шпинделя (сверла) быстро – Шпб; сигнал «Конец цикла» – КЦ.

В программе ПЭВМ предусмотрена остановка движения ШБ при достижении ею крайних положений KB и КН независимо от режима работы станка.

*Подвариант 6.1.* После кратковременного появления команды «Пуск» (кнопка на экране сенсорного монитора) ШБ из верхнего положения П0 ускоренно движется до положения П1. От П1 до П2 ШБ движется на малой скорости с включенным шпинделем (Шпб). В положении П2 происходит реверс двигателя 10 и шпиндельная бабка ускоренно возвращается в П0. По пути при достижении ею положения П1 отключается вращение сверла.

Повторение цикла осуществляется при следующем появлении команды «Пуск».

*Подвариант 6.2.* После кратковременного появления команды «Пуск» (кнопка на экране сенсорного монитора) включается двигатель 9 вращения сверла на медленную скорость и ШБ из верхнего положения П0 движется вниз до положения П2, затем возвращается в П0. Через 1 секунду ШБ из верхнего положения П0 движется вниз до П3, далее реверсируется и вновь возвращается в П0. Двигатели 9 и 10 отключаются и система готова к новому циклу работы.

*Подвариант 6.3.* После кратковременного появления команды «Пуск» с выбранной кнопки блока управления из верхнего положения П0 ускоренно движется до положения П2. От П2 до П4 ШБ движется на малой скорости с включенным шпинделем с малой скоростью (Шпм). В положении П4 происходит реверс двигателя 10 и шпиндельная бабка ускоренно возвращается в П0. По пути при достижении ею положения П2 отключается вращение сверла.

Повторение цикла осуществляется при следующем появлении команды «Пуск».

## **Б.8. ВАРИАНТ 7. ЛИНИЯ ХИМИЧЕСКОЙ ОБРАБОТКИ ДЕТАЛЕЙ**

Линия химической обработки деталей предназначена для последовательной обработки деталей в ваннах 1...5 с химическими растворами (рис. Б.7), а также для сушки деталей в сушильной камере 8. Подача нагретого воздуха осуществляется вентилятором 7 с приводом от двигателя 6.

Обрабатываемые детали вручную загружаются оператором в барабан 10, когда он находится на ложементе 9 (положение ПЭ). На тележке 11 расположены механизм подъема и опускания барабана 12 и механизм передвижения тележки 13. С целью ускорения химической обработки деталей, а также улучшения качества обработки предусмотрен привод вращения барабана 14.

Для улучшения наглядности работы механизмов здесь и далее будем считать, что 1 мин. реального технологического процесса будет соответствовать 1-3 с на лабораторном комплексе.

После ручной загрузки барабана и нажатии на кнопку «Пуск» (кнопка на экране сенсорного монитора) барабан с ложементов поднимается (команда В) в крайнее верхнее положение (КВ). Затем тележка 11 перемещается (команда П) к ванне 1 (положение П1). Барабан опускается вниз (Н).

При достижении крайнего нижнего положения (КН) включается вращение барабана (Вр) на 3 с в обезжиривающем растворе. После паузы 2 с вращение барабана возобновляется еще на 3 с. По окончании операции барабан переносится (команда Л) к ванне 2. Движение тележки вправо (П) или влево (Л) возможно только при нахождении барабана в крайнем верхнем положении (КВ).

В ванне 2 производится промывка деталей в проточной промышленной воде в течение 2 с. После промывки барабан перемещается к следующей ванне.

В ванне 3 осуществляется процесс травления деталей в течение 8 с. Барабан вращается в течение 2 с с паузами 2 с.



В следующей ванне 4 детали промываются в проточной холодной воде в течение 2 с с неподвижным барабаном.

В ванне 5 производится покрытие деталей оксидной пленкой в течение 5 с. Во время этой операции производится постоянное вращение барабана.

Затем производится промывка деталей в проточной воде (ванна 4) в течение 2 с. По окончании промывки барабан поднимается над ванной и стоит без вращения 2 с для стекания воды.

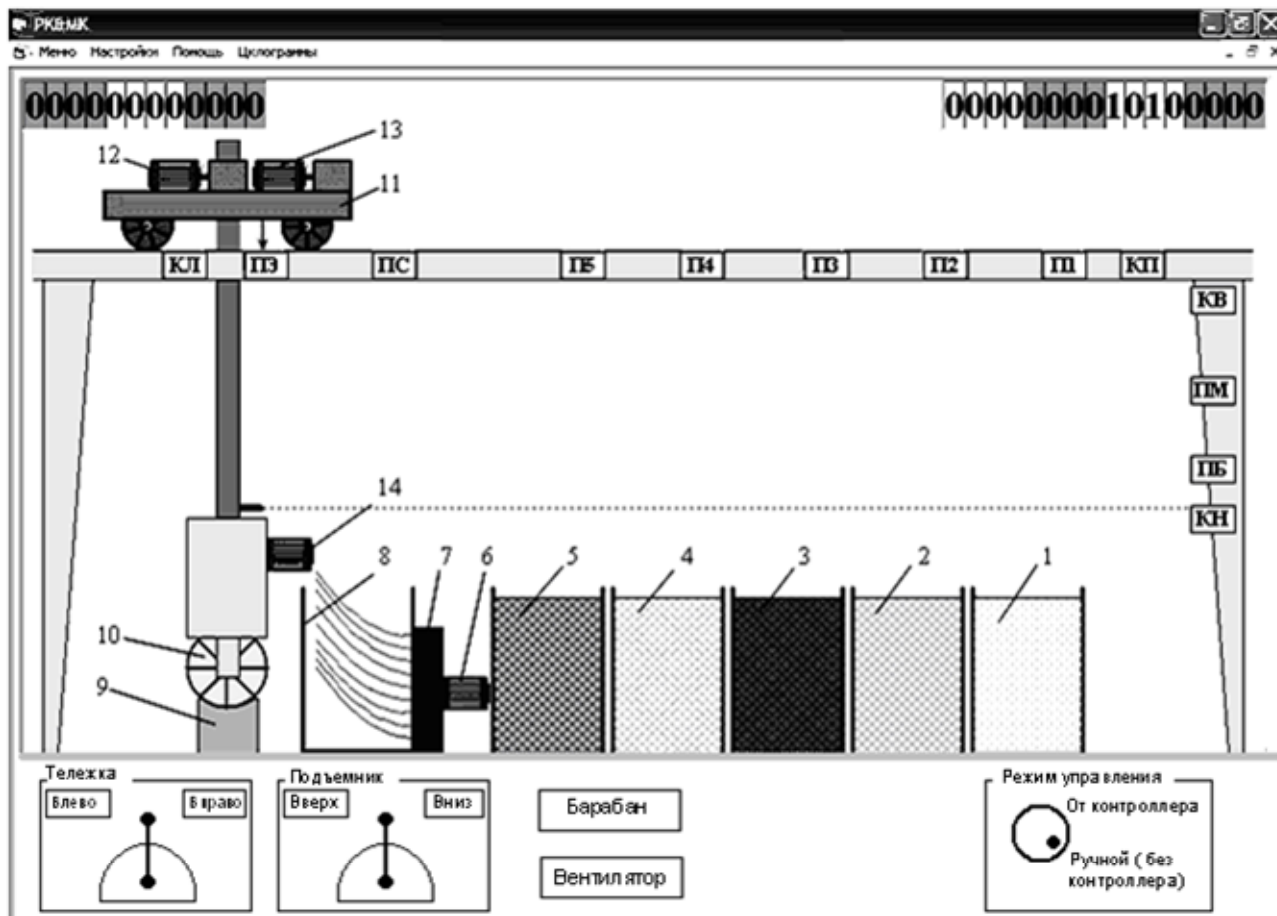


Рис. Б.7. Линия химической обработки деталей

Промытые детали переносятся в камеру сушки. Вентилятор включается командой СБ. Время сушки 3 с. Затем барабан в верхнем положении остывает в течение 2 с и переносится к месту разгрузки на ложемент. С окончанием последней операции формируется сигнал «Конец цикла» и выдается на сигнальную лампу на экране сенсорного монитора.

При нажатии на кнопку «Стоп» (на экране сенсорного монитора) система автоматически должна прекратить свою работу в том состоянии, в котором находится. Окончание работы с прерванного места осуществляется только вручную. Для включения системы в автоматическом режиме надо привести механизмы в исходное состояние, то есть барабан должен находиться на ложементе.

Положения ПМ и ПБ дают преподавателю возможность разнообразить работу рассматриваемой линии.

*Подвариант 7.1.* После кратковременного появления команды «Пуск» (кнопка на экране сенсорного монитора) барабан с деталями поднимается до положения КВ и переносится до ванны 3 (положение ПЗ). Далее барабан погружается в ванну и вращается в растворе по 2 с с паузой 2 с в течение 8 с. Затем барабан переносится в зону сушки. Включается двигатель вентилятора на 4 с. После чего барабан возвращается в исходное положение и зажигается лампа «Конец цикла» на экране сенсорного монитора. Система готова к новому циклу работы.

*Подвариант 7.2.* Требуется автоматизировать процесс контроля работоспособности всех элементов линии химической обработки. В режиме «Тест» после нажатия кнопки «Пуск-Т» (кнопка на экране сенсорного монитора) барабан поднимается механизмом подъема до положения КВ. Затем тележка перемещается вправо до датчика КП и возвращается в положение сушки ПС. Здесь барабан опускается в нишу с вентилятором. Вентилятор командой СБ включается на 2 с. После отключения вентилятора барабан поднимается до положения КВ и передвигается влево до датчика КЛ, а затем возвращается в исходное состояние. При исправности всех датчиков и работоспособности всех механизмов загорается сигнал «Конец цикла проверки» на экране сенсорного монитора. При составлении программы для контроллера считать, что команды на включение вентилятора и привода вращения барабана являются подтверждением их работоспособности.

*Подвариант 7.3.* Имеется два типа деталей с различными химическими процессами обработки. Тип деталей задается переключателем (можно использовать кнопку с отключением при повторном нажатии) на экране сенсорного монитора. При нажатии на кнопку «Пуск» (на экране сенсорного монитора) барабан с деталями первого типа поднимается до положения КВ и перемещается в положение ПЗ. Барабан опускается в ванну 3 на 3 с, а затем возвращается в исходное состояние.

Для второго типа деталей после нажатия кнопки «Пуск» барабан поднимается до положения КВ и движется к ванне 1 (П1). Далее барабан опускается на 4 с в химический раствор и затем возвращается в исходное положение.

Цикл обработки данного типа деталей должен заканчиваться независимо от изменения задания типа детали.

## **Б.9. ВАРИАНТ 8. УЧАСТОК УПАКОВКИ**

Упаковочная машина предназначена для установки заготовок или коробок (например, коробок с обувью) друг на друга и дальнейшей укладке полученной стопы в емкость (рис. Б.8). Установка содержит толкатель 4, перемещающий коробки 1 в горизонтальной плоскости, подъемник 5, подъемник стопы с захватом 2,

перемещаемый по горизонтали и вертикали тележкой 3 с двигателями 6 и 7, транспортер 8. При работе системы необходимо сложить заготовки (коробки) друг на друга по 3 штуки, переместить полученную стопу с помощью захвата на транспортер для дальнейшей упаковки.

Упаковочная машина работает следующим образом. Исходное положение толкателя КЛТ, подъемника КВ, захват разжат и находится в положении П1, тележка находится в положении ПЗ. На транспортере установлена пустая емкость 9 для размещения в ней коробок.

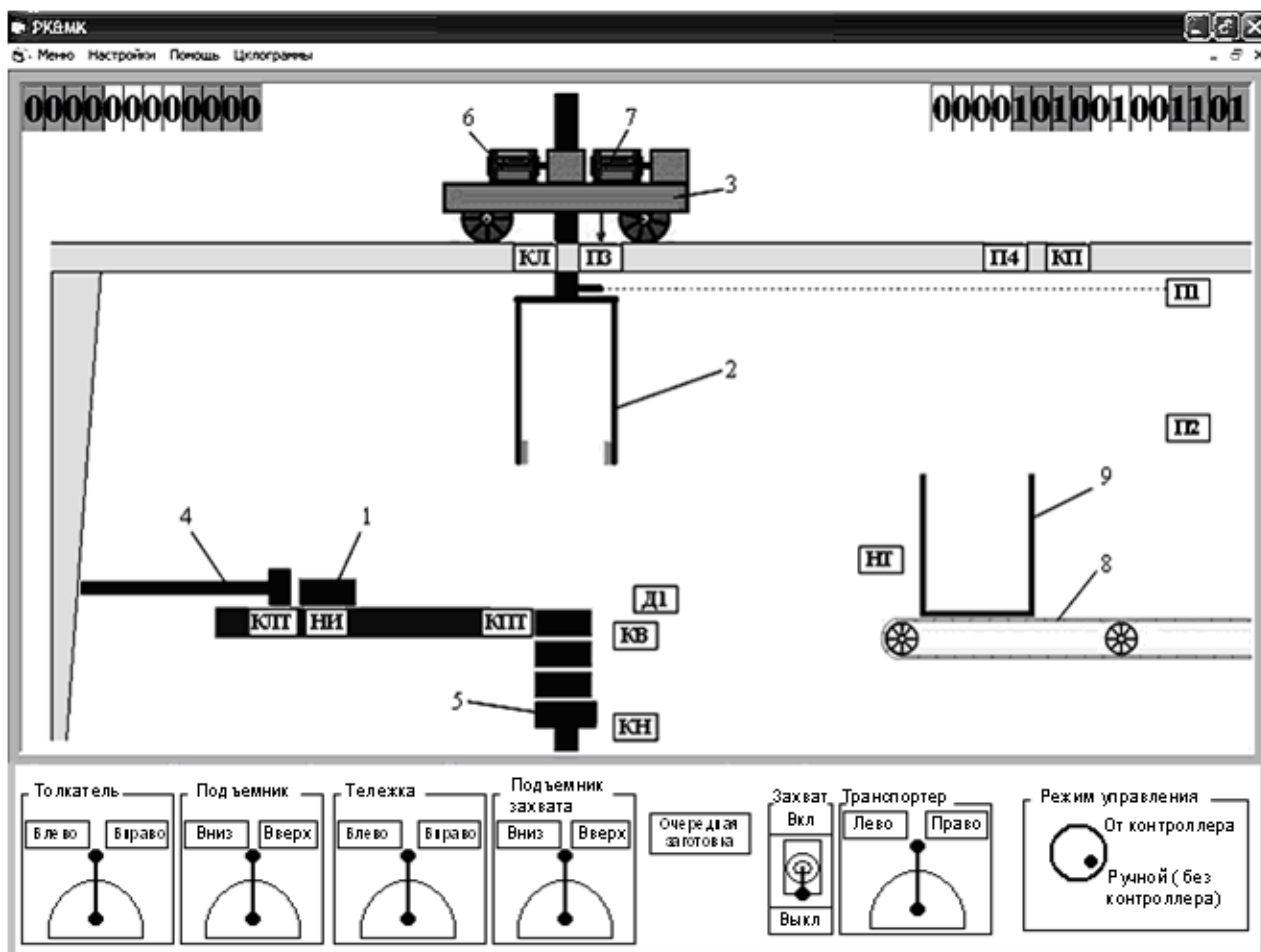


Рис. Б.8. Участок упаковки

При нажатии кнопки «Очередная заготовка» в нижней части монитора ПЭВМ на столе подачи появляется заготовка 1, наличие которой контролируется датчиком НИ. Толкатель 4 движется (ТолП) в крайнее правое положение КЛТ, перемещая коробку на подъемник 5. Далее толкатель по команде ТолЛ возвращается в исходное крайнее левое положение КЛТ. Заготовки опускаются (ПодВн) на высоту заготовки (пока не отключится датчик Д1). Далее работа повторяется до получения на подъемнике 5 стопы максимум из 3 заготовок. Подъемник 5 с тремя заготовками по команде ПодВв поднимается до положения КВ. После этого опускается захват (ШтВн) до положения П2, происходит его зажатие (Зажим), подъем

(ШтВв) до положения П1. Далее по команде ТелП тележка из положения П3 перемещается в положение П4. Происходит опускание (ШтВн) захвата и его разжатие – заготовки размещаются в емкости (таре) 9. Наличие тары контролируется датчиком НТ. Захват поднимается (ШтВв) в верхнее положение П1. Включается транспортер 8 (Тран) и емкость с заготовками увозится транспортером. Включается тележка для движения влево (ТелЛ) и устанавливается в исходное положение П3. В дальнейшем цикл повторяется.

Конечные выключатели КЛ и КП контролируют крайние соответственно левое и правое положения тележки 3. При нормальной работе автоматики движение тележки ограничивается датчиками П3 и П4. В положениях КЛ и КП тележка может оказываться только при сбоях автоматики.

*Подвариант 8.1.* Автоматизация захвата. Исходное положение: захват разжат и находится в положении П1, тележка – в положении П3. После нажатия кнопки «Пуск» (на экране сенсорного монитора) захват перемещается вниз в положение П2, далее происходит его зажим. После этого захват поднимается до верхнего положения П1 и затем его разжим. Далее цикл работы непрерывно повторяется до нажатия кнопки «Сброс» (на экране сенсорного монитора).

*Подвариант 8.2.* Автоматизация толкателя, захвата, тележки и транспортера (не работает только привод подъемника 5, т.е. происходит перемещение коробок по одной). При нажатии кнопки «Очередная заготовка» в нижней части монитора ПЭВМ появляется заготовка. Толкатель переносит ее на подъемник, находящийся в верхнем положении. Опускается захват в разжатом состоянии, происходит зажим заготовки, подъем захвата до верхнего положения. Далее включается тележка для движения направо до положения П4 и опускание захвата с заготовкой на транспортер. После подъема захвата транспортер включается и увозит заготовку. В верхнем положении захвата включается тележка для движения налево до П3. Цикл закончен и далее он повторяется при нажатии кнопки «Очередная заготовка» в нижней части монитора ПЭВМ. На экране сенсорного монитора располагаются кнопки «Пуск», «Стоп» и сигнальная лампа о движении тележки вправо.

*Подвариант 8.3.* Тестовый контроль приводов датчиков тележки с захватом. Остальные механизмы не работают.

Исходное положение тележки 3 в положении П3, разжатый захват 2 в положении П1. При нажатии кнопки пуска теста на экране сенсорного монитора, захват опускается до положения П2. Происходит зажатие захвата и затем подъем его до положения П1. Далее тележка из положения П3 движется в положение П4. Происходит опускание захвата до положения П2 и его разжим. Захват поднимается в верхнее положение. Включается тележка и устанавливается в исходное положение П3. Если все движения прошли в полном соответствии с заданием, то на экране сенсорного монитора загорается сигнальная лампа. Если же были сбои в работе, то через 10с после нажатия на кнопку пуска теста загорается другая сиг-

нальная лампа. Исправность датчиков КЛ и КП в данном подварианте не проверяется.

### Б.10. ВАРИАНТ 9. ЛИФТ ПАССАЖИРСКИЙ

Пассажирский лифт – хорошо знакомый всем объект, не требующий пояснений по назначению и работе. В рассматриваемом варианте лифт (рис. Б.9) рассчитан на 3 этажа.

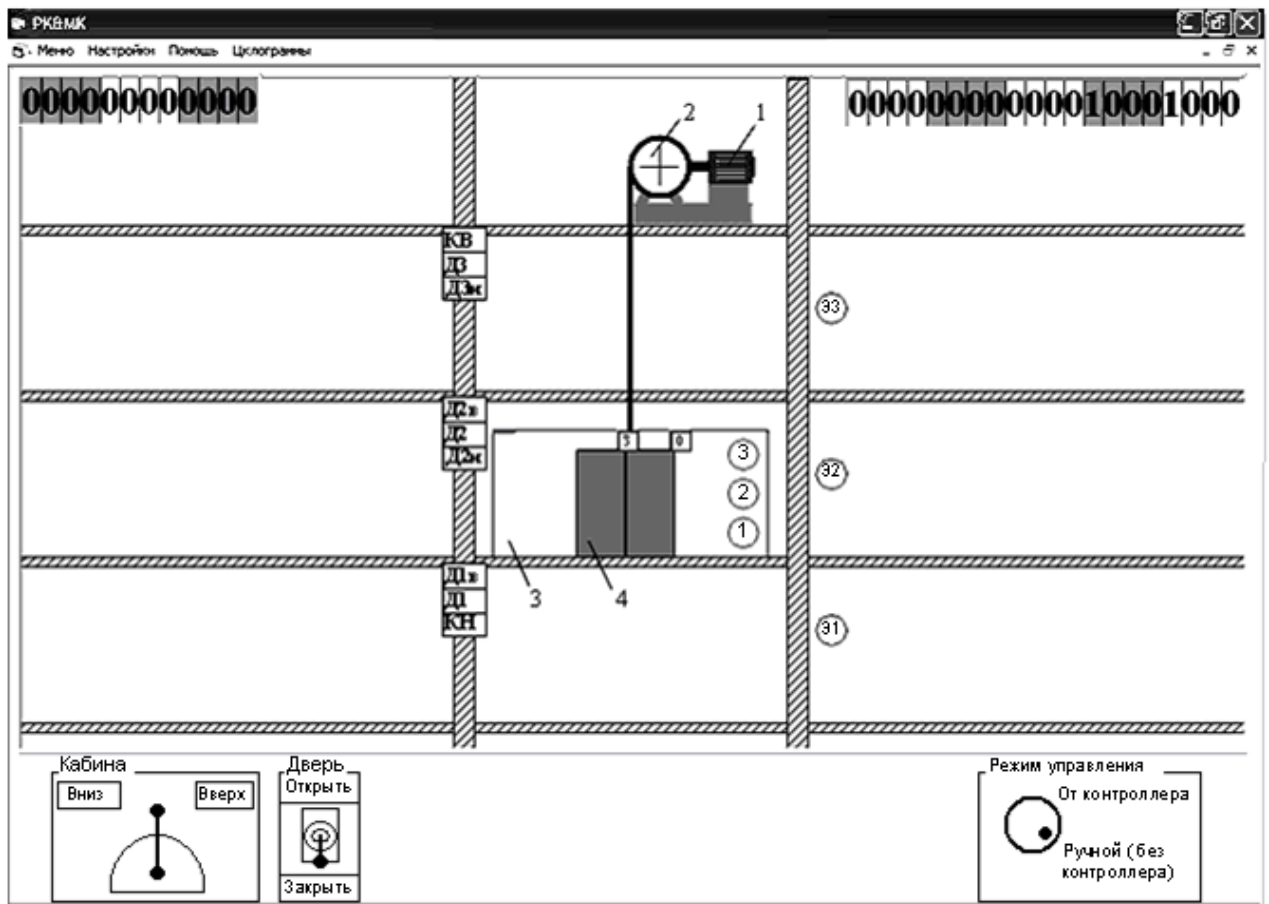


Рис. Б.9. Пассажирский лифт

В машинном отделении располагается силовая установка для подъема/опускания кабины: двигатель 1 и шкив 2. Кабина лифта 3 имеет двухстворчатую дверь 4, приводимую в движение двигателем, который условно не показан. Крайние положения кабины контролируются конечными выключателями верхним и нижним соответственно КВ и КН. Точная остановка кабины на этажах обеспечивается по сигналам от этажных датчиков Д1, Д2, Д3. При подходе к заданному этажу скорость кабины лифта снижается до ползучей. Снижение скорости при подходе к 3-му этажу происходит по сигналу датчика ДЗн, при подходе к 1-му этажу – по сигналу датчика Д1в. На втором этаже установлены два датчика для организации снижения скорости, если кабина движется на заданный второй этаж.

Датчик Д2в снижает скорость кабины при ее движении на второй этаж сверху, а Д2н – при движении снизу.

Управление лифтом осуществляется кнопками Э1, Э2, Э3, расположенными на этажных площадках соответственного 1-го, 2-го и 3-го этажей. Они служат для вызова кабины на заданный этаж.

Кнопки 1, 2 и 3 расположены в кабине и являются кнопками приказов движения кабины на требуемый этаж.

Система управления лифтом должна формировать управляющие команды на приводы кабины и двери. На привод кабины система управления должна вырабатывать команды движения «Вверх» (В), «Вниз» (Н) и замедления скорости (М).

При одновременном наличии команд В и М осуществляется движение вверх медленно, при одновременном наличии команд Н и М – движение вниз медленно. На привод двери подаются команды «Открыть» (ОД) и «Заккрыть» (ЗД). Положение двери контролируется датчиками открытого и закрытого положения двери соответственно О и З. Команды вызова Э1, Э2, Э3 и команды приказов 1, 2, 3 подаются обучаемым с кнопок на экране сенсорного монитора. Команды вызовов и приказов обеспечивают подсветку кнопок на экране монитора ПЭВМ.

Алгоритм управления лифтом предполагается следующим. Движение кабины возможно лишь при закрытой двери, т.е. при наличии сигнала З. Кабина идет на тот этаж, на котором раньше нажали кнопку вызова с этажной площадки. При подходе на заданный этаж кабина автоматически замедляет свой ход и на ползучей скорости подходит к месту остановки. При остановке кабины дверь автоматически открывается. Пассажир входит в кабину и нажимает кнопку приказа. Дверь кабины закрывается, включается привод кабины и кабина движется на заданный этаж (с замедлением перед остановкой). При остановке кабины дверь автоматически открывается, пассажир выходит из кабины. Через запрограммированную задержку времени дверь автоматически закрывается. Дверь автоматически закроется и в случае, если пассажир зашел в кабину и долго не нажимает на кнопку приказа. Если дверь кабины закрылась, а кнопка приказа не нажата, но есть сигнал запроса с какого-то этажа, то лифт будет идти на этаж, с которого поступил запрос. Таким образом, пассажир, вошедший в кабину, должен нажать кнопку приказа до закрытия двери.

На экране сенсорного монитора должны быть представлены кнопки вызовов на этажи и кнопки приказов из кабины лифта. Изображение лифта на мониторе ПЭВМ также имеет кнопки вызовов и приказов, и подавать команды вызовов и приказов можно, щелкая левой клавишей мыши по соответствующей кнопке. То есть кнопки на экране сенсорного монитора и кнопки на изображении виртуального лифта как бы соединены параллельно.

*Подвариант 9.1.* При нажатии и отпускании кнопки Э1 кабина совершает непрерывные движения вверх и вниз (от Д1 до Д3 и обратно), останавливаясь на 1...2 с на каждом этаже. Движение осуществляется на малой скорости. При нажатии на кнопку Э2 движение прекращается.






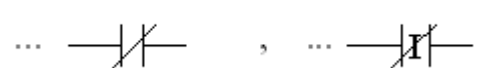
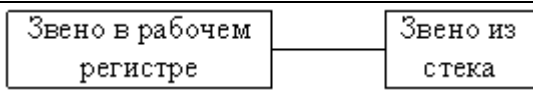
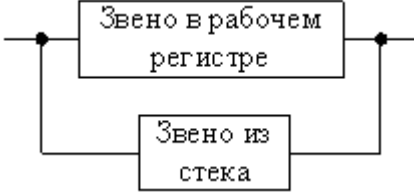
*Подвариант 9.2.* При нажатии на кнопку Э3 кабина совершает непрерывные движения на медленной скорости вверх и вниз (от Д1 до Д3 и обратно) без остановки на 2-м этаже. При нажатии на кнопку Э2 движение кабины прекращается. При нажатии на кнопку Э1 кабина должна идти на 1-ый этаж и там остановиться. При повторном нажатии кнопки Э3 описанная работа лифта повторяется.

*Подвариант 9.3.* При включении системы в работу кабина, где бы она ни была, движется автоматически до 1-го этажа и останавливается. При первом нажатии на кнопку Э1 кабина движется на 2-й этаж, стоит там в течение времени  $t_1$  и возвращается на 1-й этаж. При втором нажатии на кнопку Э1 кабина движется на 3-й этаж, стоит там в течение времени  $t_2$  и возвращается на 1-й этаж. При третьем нажатии на кнопку Э1 кабина движется на 2-й этаж, стоит там в течение времени  $t_1$  и возвращается на 1-й этаж. При четвертом нажатии на кнопку Э1 кабина движется на 3-й этаж, стоит там в течение времени  $t_2$  и возвращается на 1-й этаж. То есть при каждом нечетном нажатии кнопки Э1 кабина совершает «ходку» на 2-й этаж, а при каждом четном нажатии – на 3-й этаж. Движение кабины осуществляется на малой скорости.

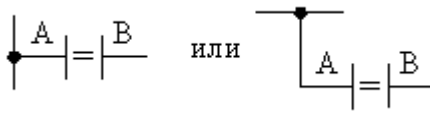
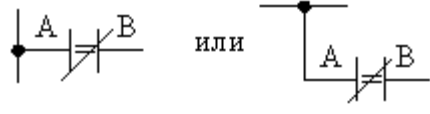
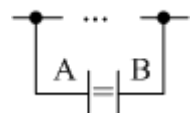
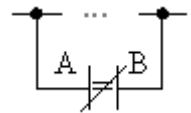
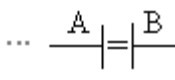
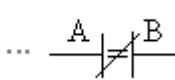
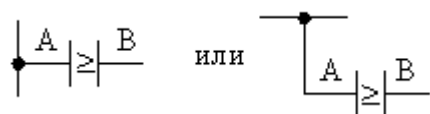
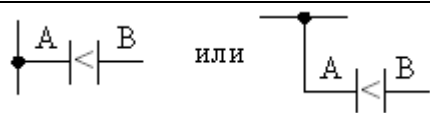
## ПРИЛОЖЕНИЕ В. СИСТЕМА КОМАНД МИКРОКОНТРОЛЛЕРОВ

Представленная ниже система команд ориентирована на программирование как с использованием пакета Direct Soft, так и ручного программатора.

### Булевы команды

Мнемоника	Обозначения в DirectSoft32	Описание
STR, STRI		Начинает новое звено или следующую цепь с замыкающим контактом.
STRN, STRNI		То же, но с размыкающим контактом
OR, ORI		Присоединение замыкающего контакта параллельно другому контакту в звене
ORN, ORNI		То же, но с размыкающим контактом
AND, ANDI		Последовательное включение замыкающего контакта
ANDN, ANDNI		То же, но с размыкающим контактом
ANDSTR		Последовательное соединение двух звеньев
ORSTR		Параллельное соединение двух звеньев
OUT	-(OUT)	Вывод состояния звена в регистр состояния
OUTI	-(OUTI)	Вывод состояния звена немедленный (при выполнении команды в ходе сканирования)
OROUT	-(OROUT)	Выводит состояние звена в регистр состояния. Можно многократно ссылаться на один и тот же канал

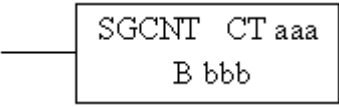
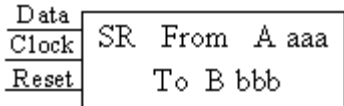


OROUTI	-(OROUTI)	Вывод состояния звена немедленный при выполнении команды в ходе сканирования. Можно многократно ссылаться на один и тот же канал
SET, SETI	-(SET), -(SETI)	Установить (включить) и зафиксировать канал или группу каналов
RST, RSTI	-(RST), -(RSTI)	Сбросить (отключить) канал или группу каналов, которые были установлены (включены) командой SET, SETI
PAUSE	-(PAUSE)	Прекращает обновление заданных сигналов вывода
STRE		Начинает новое звено или следующую ветвь с замыкающим контактом. Контакт замкнут, если A=B
STRENE		То же, но с размыкающим контактом. Контакт замкнут, если A≠B
ORE		Присоединение замыкающего контакта параллельно другому контакту. Контакт замкнут, если A=B
ORNE		То же, но с размыкающим контактом. Контакт замкнут, если A≠B
ANDE		Последовательное включение замыкающего контакта. Контакт замкнут, если A=B
ANDNE		То же, но с размыкающим контактом. Контакт замкнут, если A≠B
STR		Начинает новое звено или следующую цепь, с замыкающим контактом. Контакт замкнут, если A≥B
STRN		То же. Контакт замкнут, если A<B

OR		Присоединение замыкающего контакта параллельно другому контакту. Контакт замкнут, если $A \geq B$
ORN		То же. Контакт замкнут, если $A < B$

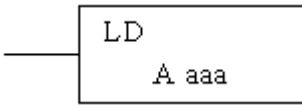
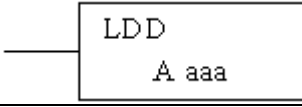

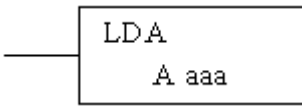
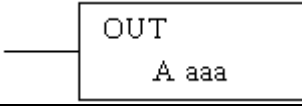
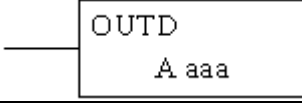
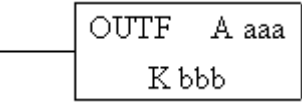
### Команды таймеров и счетчиков

Мнемоника	Обозначения в DirectSoft32	Описание
TMR		Таймер с дискретностью 0,1 с. Максимальная установка 999,9 с (при $K=9999$ ). Номер таймера T0...T377 (T0...T177)
TMRF		Быстрый таймер с дискретностью 0,01 с. Максимальная установка 99,9 с (при $K=9999$ ). Номер таймера T0...T377 (T0...T177)
TMRA		Накапливающий (аккумулирующий) таймер с дискретностью 0,1 с. Максимальная установка 9999999,9 с (при $K=99999999$ ). Номер таймера T0...T377 (T0...T177)
TMRFA		Накапливающий (аккумулирующий) быстрый таймер с дискретностью 0,01 с. Максимальная установка 999999,9 с (при $K=99999999$ ). Номер таймера T0...T377 (T0...T177)
CNT		Суммирующий счетчик. Уставка счетчика 0...9999. Номер счетчика CT0...CT377 (CT0...CT177)
UDC		Реверсивный счетчик. Уставка счетчика 0...99999999. Номер счетчика CT0...CT377 (CT0...CT177)

SGCNT		Счетчик стадий (суммирующий)
SR		Сдвиг регистра

В средней колонке таблицы для всех элементов условно указан один и тот же номер 1 и задание уставки константой  $K=30$ .

### Команды загрузки аккумулятора и вывода данных

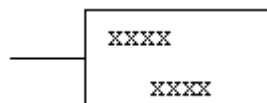
Мнемоника	Обозначения в DirectSoft32	Описание
LD		Загрузка 16-битного слова из ячейки $V_{xxxx}$ в аккумулятор
LDD		Загрузка двойная. Загружает 32-битное слово из ячеек $V_{xxxx}$ и $V_{(xxxx+1)}$ в аккумулятор
LDF		Загрузка форматированная. Загружает в аккумулятор заданное количество последовательных дискретных битов памяти
LDA		Загрузка адреса. Загружает аккумулятор 16-ричным значением восьмеричной константы (адреса)
OUT		Вывод. Копирует младшие 16 бит аккумулятора в ячейку с адресом $V_{xxxx}$
OUTD		Вывод двойной. Копирует значения в аккумуляторе в ячейки с адресами $V_{xxxx}$ и $V_{(xxxx+1)}$
OUTF		Вывод форматированный. Выводит заданное число бит (1-32) из аккумулятора в заданные дискретные области памяти
POP		Выталкивание из стека аккумулятора. Все значения в стеке сдвигаются вверх на один уровень. Вытолкнутое значение записывается в аккумулятор, а значение, бывшее в аккумуляторе, удаляется

### Логические и математические команды

Логические и математические команды реализуют операции с числами (данными). Числа могут располагаться в аккумуляторе А, ячейках V-памяти, представляться константами К.

В командах указывается, где храниться первое число, где храниться второе число и куда помещается результат. Как правило, первое число хранится в А и там же в А сохраняется результат операции.

Ниже в таблице не указывается обозначение команды в программном пакете Direct Soft, так как они имеют один и тот же следующий вид:



В левом верхнем углу пишется мнемоника команды, а внизу посередине – адрес ячейки V-памяти  $V_{xxxx}$  или константа  $K_{xxxx}$ .

Мнемоника	Наименование	Описание
AND	Логическое И	Логическое И для младших 16 бит А и содержимого V-памяти. Результат в А
ANDD	Логическое И двойное	Логическое И значения в А и 8-значной константы. Результат в А
OR	Логическое ИЛИ	Логическое ИЛИ младших 16 бит А и содержимого V-памяти. Результат в А
ORD	Логическое ИЛИ двойное	Логическое ИЛИ значения в А и 8-значной константы. Результат в А
XOR	Исключающее ИЛИ	Логическое исключающее ИЛИ младших 16 бит А и содержимого V-памяти. Результат в А
XORD	Исключающее ИЛИ двойное	Логическое исключающее ИЛИ значения в А и 8-значной константы. Результат в А
CMP	Сравнить	Сравнивает значение младших 16 бит А с содержимым V-памяти. Устанавливает признаки: равно (=), больше (>), меньше (<)
CMPD	Сравнить двойное	То же, но 32 бита А с двумя последовательными ячейками V-памяти или 8-значной константой
ADD	Сложение	Суммирует значение 16 бит А в BCD (двоично-десятичном) формате с содержимым V-памяти. Результат в А
SUB	Вычитание	Вычитает из значения 16 бит А в BCD формате содержимое V-памяти или 4-значную BCD константу. Результат в А
MUL	Умножение	Умножает значение младших 16 бит А в BCD формате на содержимое ячейки V-памяти или 4-значную BCD константу. Результат в А
DIV	Деление	Делит значение младших 16 бит А в BCD формате на содержимое ячейки V-памяти или 4-значную BCD константу. Результат в А

ADDD SUBD MULD DIVD	Сложение двойное Вычитание двойное Умножение двойное Деление двойное	Эти команды аналогичны командам ADD, SUB, MUL и DIV, но операции осуществляются с 32 разрядами A в BCD формате и с двумя последовательными ячейками V-памяти или 8-значной константой
ADDB SUBB MULB DIVB	Сложение двойное Вычитание двойное Умножение двойное Деление двойное	Эти команды аналогичны командам ADD, SUB, MUL и DIV, но информация в A, ячейках V-памяти и константы – в двоичном представлении

### Команды работы с разрядами и преобразования чисел

Мнемоника	Наименование	Описание
INC	Инкремент	Увеличивает величину, хранящуюся в BCD формате в ячейке V-памяти, на 1 при каждом вызове команды
DEC	Декремент	Уменьшает величину, хранящуюся в BCD формате в ячейке V-памяти, на 1 при каждом вызове команды
INCB DECB	Инкремент двоичный Декремент двоичный	Команды аналогичны командам INC и DEC, но информация в ячейках V-памяти в двоичном представлении
BIN	Команда двоичная	Преобразует число в A в BCD формате двоичное представление. Результат в A
BCD	Двоично-десятичное кодирование	Преобразует число в A в двоичном представлении в BCD формате. Результат в A
INV	Инверсия	Дает дополнительный код 32-битного значения A. Результат в A
GRAY	Из кода Грея в BCD	Преобразует число в 16-битном формате Грея в BCD формате. Результат в A
SUM	Сумма	Вычисляет число бит равных 1 в A. Результат в формате HEX помещается в A
SHFL	Сдвиг влево	Сдвигает биты в A влево на заданное число разрядов
SHFR	Сдвиг вправо	То же, но сдвиг вправо

### Табличные команды

Мнемоника	Наименование	Описание
MOV	MOVE	Перемещает значения из одной области V-памяти в другую область V-памяти с такой же длиной.
MOVMC LDLBL	Move Memory Cartridge Load Label	Копирование данных между V-памятью и памятью программы
LDLBL MOVMC	Copy Data From a Data Label Area to	Данные копируются из области метки данных (Data Label Area) в V-память

## ОГЛАВЛЕНИЕ

Введение.....	3
1. Работа №1. Методические указания по программированию контроллера Simatic S7-300	
1.1. Цель работы.....	4
1.2. Содержание работы.....	4
1.3. Описание лабораторного стенда.....	4
1.4. Конфигурирование в STEP 7 v 5.3.....	8
1.5. Программирование пользовательской задачи.....	13
1.6. Операции с таймерами.....	18
1.7. Операции счета.....	20
1.8. Операции с аналоговыми сигналами.....	21
1.9. Программирование функциональных блоков и функций.....	23
1.10. Пример программирования контроллера SIMATIC S7-300.....	29
1.11. Загрузка программы в контроллер и запуск программы.....	32
1.12. Выполнение лабораторной работы.....	32
1.13. Требования к отчёту.....	37
1.14. Контрольные вопросы.....	37
2. Работа №2. Изучение промышленной сети PROFIBUS-DP	
2.1. Цель работы.....	38
2.2. Содержание работы.....	38
2.3. Описание лабораторного стенда.....	38
2.4. Конфигурирование сети в STEP7 v 5.3.....	40
2.5. Программирование контроллеров.....	45
2.6. Загрузка программ в программируемые контроллеры и включение в работу.....	49
2.7. Выполнение лабораторной работы.....	51
2.8. Требования к отчёту.....	61
2.9. Контрольные вопросы.....	61
3. Работа №3. Изучение сенсорной панели оператора Siemens TP177A и ее использования в системах автоматизации	
3.1. Цель работы.....	62
3.2. Содержание работы.....	62
3.3. Описание лабораторного стенда.....	62
3.4. Технические характеристики панели оператора TP177A.....	67
3.5. Операционная система сенсорной панели SIMATIC TP 177A.....	69
3.6. Программирование сенсорного монитора SIEMENS TP177A.....	71
3.6.1. Общие положения.....	71
3.6.2. Начало работы с WinCC flexible. Создание нового проекта.....	72
3.6.3. Работа с тегами.....	77

3.6.4.	Создание объектов на экране панели оператора.....	79
3.6.5.	Загрузка проекта в сенсорный монитор SIEMENS TP177A.....	85
3.7.	Выполнение лабораторной работы .....	86
3.8.	Требования к отчёту.....	87
3.9.	Контрольные вопросы.....	87
4.	Работа №4. Изучение сенсорного монитора OMRON NT21	
4.1.	Цель работы.....	89
4.2.	Программа работы.....	89
4.3.	Технические характеристики сенсорного монитора NT21.....	89
4.4.	Операционная система сенсорного монитора NT21 .....	93
4.5.	Программирование сенсорного монитора OMRON NT21.....	94
4.6.	Работа с программой NT-series Support Tool.....	97
4.7.	Примеры программирования сенсорного монитора NT21 .....	103
4.8.	Загрузка проекта в сенсорный монитор OMRON NT21.....	105
4.9.	Выполнение лабораторной работы.....	107
4.10.	Требования к отчёту.....	109
4.11.	Контрольные вопросы.....	109
5.	Работа №5. Программирование микроконтроллеров DL05 и DL06 программируемых контроллеров фирмы AUTOMATION DIRECT	
5.1.	Цель работы.....	111
5.2.	Содержание работы.....	111
5.3.	Технические характеристики изучаемых микроконтроллеров.....	111
5.3.1.	Общие положения.....	111
5.3.2.	Устройство микроконтроллеров DL06.....	112
5.3.3.	Общие характеристики микроконтроллеров DL06.....	113
5.3.4.	Входы и выходы микроконтроллеров DL06.....	114
5.3.5.	Распределение памяти микроконтроллеров.....	116
5.4.	Программирование микроконтроллеров.....	118
5.4.1.	Общие положения.....	118
5.4.2.	Программирование на языке RLL.....	118
5.4.3.	Примеры программирования контроллера на языке RLL.....	128
5.4.4.	Программирование на языке RLL <sup>PLUS</sup> .....	130
5.5.	Подготовка программы с помощью пакета программирования DirectSoft32.....	132
5.6.	Ввод программы в микроконтроллер .....	136
5.7.	Вывод программы на печать .....	138
5.8.	Дополнительные возможности микроконтроллеров.....	139
5.9.	Выполнение лабораторной работы .....	139
5.10.	Требования к отчёту.....	145
5.11.	Контрольные вопросы.....	145

Приложения	
Приложение А. Адресация сигналов и команд программируемых контроллеров SIMATIC S7-300.....	147
Приложение Б. Варианты технологических объектов управления	
Б.1. Общие положения.....	152
Б.2. Вариант 1. Роботизированный комплекс транспортировки изделий.....	155
Б.3. Вариант 2. Участок транспортировки заготовок в методическую печь для их нагрева перед прокаткой (методическая печь).....	157
Б.4. Вариант 3. Участок нагревательного колодца обжимного прокатного стана.....	159
Б.5. Вариант 4. Участок сортировки и пакетирования годных и бракованных листов металла.....	161
Б.6. Вариант 5. Участок транспортировки труб большого диаметра.....	163
Б.7. Вариант 6. Станок для сверления глубоких отверстий.....	166
Б.8. Вариант 7. Линия химической обработки деталей.....	168
Б.9. Вариант 8. Участок упаковки.....	170
Б.10. Вариант 9. Лифт пассажирский.....	173
Приложение В. Система команд микроконтроллеров .....	176



А.М. Борисов, А.С. Нестеров, Н.А. Логинова

# **ПРОМЫШЛЕННЫЕ ПРОГРАММИРУЕМЫЕ КОНТРОЛЛЕРЫ**

Учебное пособие

Техн. редактор

Издательство Южно-Уральского государственного  
университета

---

---

УОП Издательства. 454080, г. Челябинск, пр. им. В.И. Ленина, 76