

**Некоммерческое
акционерное
общество**

**АЛМАТИНСКИЙ
УНИВЕРСИТЕТ
ЭНЕРГЕТИКИ И
СВЯЗИ**

Кафедра электропривода
и автоматизации

МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ АЭП

Методические указания по выполнению расчетно-графических работ
для студентов специальности 5В071800 - Электроэнергетика

Алматы 2017

СОСТАВИТЕЛИ: С.Б. Алексеев, Ж.Ж. Тойгожинова.
Микропроцессорные средства и системы АЭП. Методические указания по выполнению расчетно-графических работ для студентов специальности 5В071800 - Электроэнергетика. – Алматы: АУЭС, 2017. – 30 с.

Методические указания содержат общие положения к выполнению расчетно-графических работ, указания по их выполнению и оформлению.

Методические указания предназначены для студентов специальности 5В071800.

Ил. 13, табл. 4, библи. – 6 назв.

Рецензент: доцент Б.К. Курпенов

Печатается по плану издания Некоммерческого акционерного общества «Алматинский университет энергетики и связи» на 2017 г.

©НАО «Алматинский университет энергетики и связи», 2017 г.

Содержание

Введение	4
1 Основные сведения о контроллерах SIMATIC S7-200, S7-1200.....	4
2 Основы программирования контроллера.....	11
3 Система команд микроконтроллера.....	13
4 Порядок выполнения расчетно-графических работ.....	18
Список литературы.....	30

Введение

Целью расчетно-графической работы является приобретение студентами практических навыков применения микропроцессорных программируемых контроллеров в схемах автоматизации технологических процессов, оснащенных современными электроприводами на основе полупроводниковых преобразующих устройств.

Областью наиболее эффективного их применения является управление робототехническими комплексами, автоматизированными линиями и цикловыми технологическими процессами.

В работе предлагается применить контроллеры S7-200, S7-1200 фирмы Siemens, являющейся одним из крупнейших производителей средств автоматизации. Для программирования контроллеров используется программное обеспечение STEP7. Указанные контроллеры имеются в лабораториях кафедры, и разработанные программы могут быть проверены и отредактированы.

1 Основные сведения о контроллерах SIMATIC S7-200, S7-1200

1.1 Технические характеристики контроллеров SIMATIC S7-200

Серия S7 –200– это ряд программируемых логических контроллеров, которые могут управлять разнообразными системами автоматизации.

На рисунке 1 представлен микроконтроллер S7–200. Компактная конструкция, расширяемость, низкая стоимость и мощная система команд микроконтроллеров S7–200 позволяют создавать на его базе простые и надежные системы управления малыми приложениями.

Кроме того, разнообразие типов процессоров и питающих напряжений обеспечивают необходимую гибкость при решении проблем автоматизации.

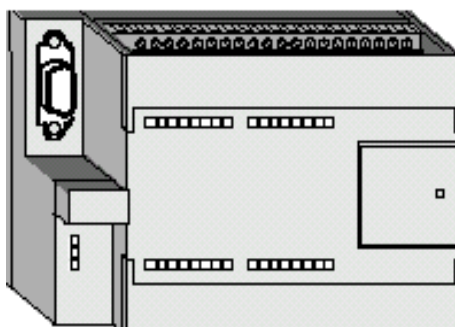


Рисунок 1.1 – Микроконтроллер S7–200

Для построения систем управления на базе контроллеров SIMATIC S7-200 может быть использовано 4 типа центральных процессоров (CPU),

каждый из которых имеет два варианта исполнения: с питанием $U = 24 \text{ В}$ и с питанием $\sim 220/230 \text{ В}$.

CPU-224 - центральный процессор, предназначенный для построения компактных систем автоматического управления высокой производительности, работающих автономно или в составе распределенных систем автоматического управления. Оснащен 14 встроенными дискретными входами и 10 дискретными выходами. Позволяет производить подключение до 7 модулей расширения ввода - вывода.

Центральные процессоры семейства S7-200 имеют следующие конструктивные особенности:

- встроенный источник питания $=24$ для питания внешних цепей и подключения датчиков;
- 1 или 2 (в CPU 226/CPU 226XM) встроенных PPI интерфейса;
- наличие шины расширения системы ввода (за исключением CPU 221);
- встроенные скоростные счетчики (30 кГц);
- 4 быстродействующих входа обработки сигналов аппаратных прерываний;
- 2 импульсных выхода (20 кГц) во всех моделях постоянного тока;
- переключатель выбора режимов работы;
- 1 (CPU 221 / CPU 222) или 2 (CPU 224/ CPU 226/ CPU 226XM) встроенных потенциометра аналогового задания цифровых параметров;
- опциональные (в виде submodule) или встроенные часы реального времени;
- опциональный модуль EEPROM памяти для хранения программ и данных;
- submodule буферной батареи для защиты данных в оперативной памяти при сбоях в питании контроллера;
- возможность использования имитаторов входных сигналов для отладки программы;
- возможность использования съемных терминальных блоков для замены модулей без демонтажа внешних цепей.

Все центральные процессоры семейства обеспечивают выполнение следующих функций:

- скоростной счет;
- обработка прерываний (останове, по времени, встречных, прерываний связи);
- прямое сканирование входов и выходов;
- выполнение операций целочисленной арифметики и арифметики с плавающей запятой, логических операций и другие;
- реализация алгоритмов ПИД регулирования.

1.2 Схема подключения входов и выходов контроллера на базе CPU 224

На рисунке 1.2 представлена схема с источником питания постоянного тока 24 В. Схема иллюстрирует порядок подключения источников питания, нагрузок и входных сигналов типа «сухой контакт» к зажимам контроллера.

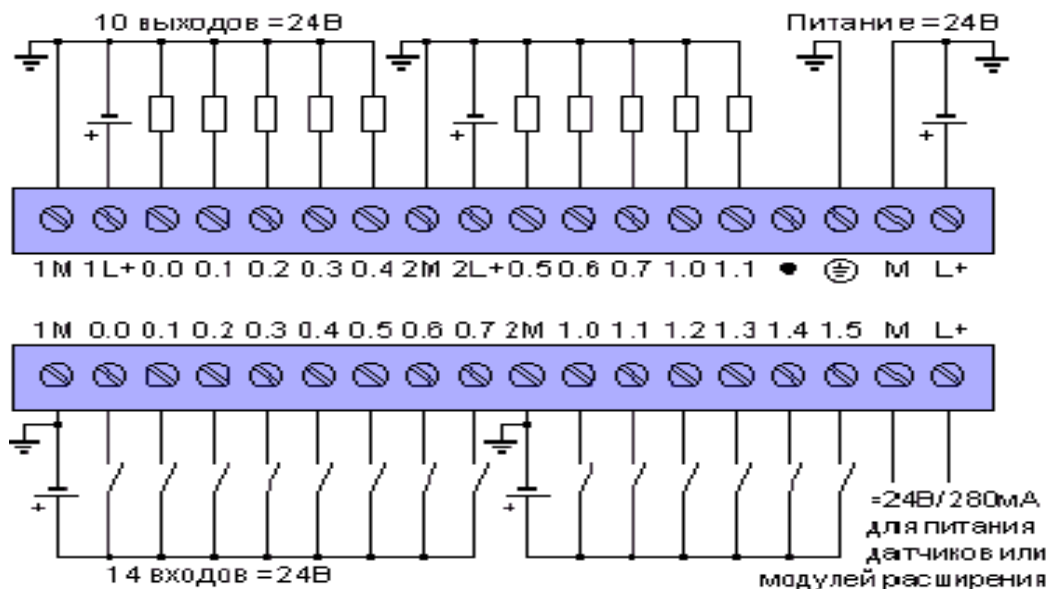


Рисунок 1.2

В качестве нагрузок к выходам могут быть подключены реле на 24 В, сигнальные лампы, а в качестве входных сигналов использованы контакты выключателей, реле, датчиков.

1.3 Технические характеристики контроллеров SIMATIC S7-1200

Контроллер S7-1200 представляет собой более современную версию и его внешний вид представлен на рисунке 1.3.

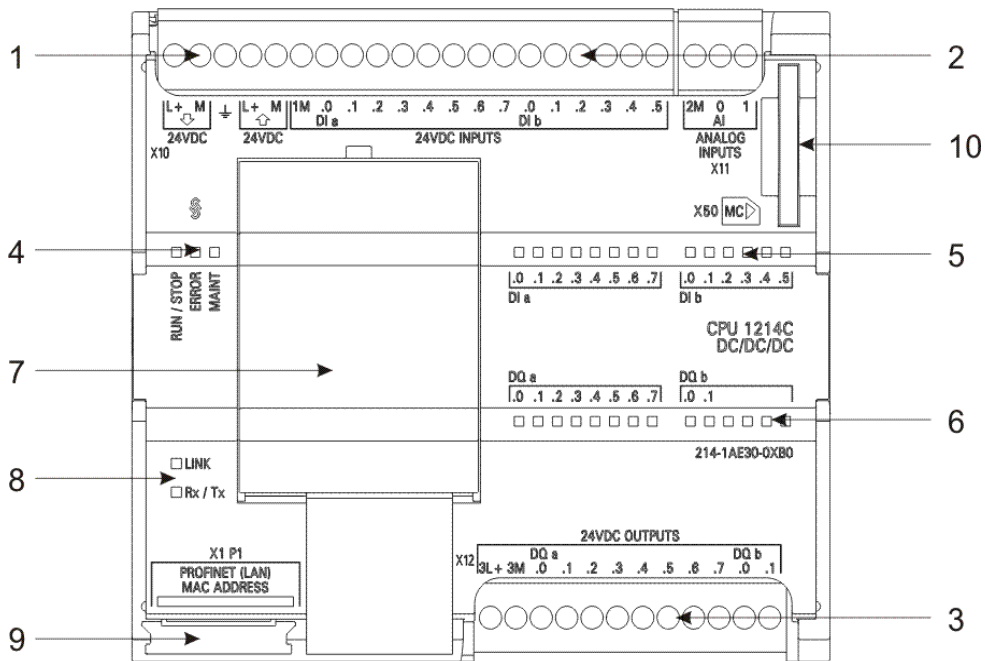


Рисунок 1.3 - Внешний вид ПЛК Siemens S7-1200 1214C (со снятыми защитными крышками)

В таблице 1.1 дается описание основных элементов контроллера.

Таблица 1.1 - Описание основных элементов ПЛК

№ п/п	Элемент	Функциональное назначение	
1	Разъем питания	Подключение питания контроллера	
2	Клеммный блок	Подключение входных цепей контроллера	
3	Клеммный блок	Подключение выходных цепей контроллера	
4	Индикаторы состояния ПЛК	STOP/RUN	Возможен постоянно горящий оранжевый или зеленый свет или мигающий (попеременно зеленый и оранжевый). Отображает режим работы CPU
		ERROR	Возможен постоянно горящий или мигающий красный свет. Отображает ошибки CPU
		MAINT	Возможен постоянно горящий или мигающий оранжевый свет. Отображает операции с картой памяти или необходимость обслуживания
5	Индикаторы состояния входов	Индикация состояния входов программируемого контроллера	
6	Индикаторы состояния выходов	Индикация состояния выходов программируемого контроллера	
7	Отсек для плат расширения	Установка либо коммуникационной платы расширения, либо платы расширения с дискретными или аналоговыми входами/выходами	
8	Индикаторы PROFINET	Индикация подключения и передачи данных PROFINET	
9	Разъем PROFINET	Подключение к сети PROFINET для связи к ПК либо с другими устройствами автоматизации	
10	Гнездо для карты памяти	Установка карты памяти	

Технические характеристики входных и выходных цепей контроллера представлены в таблице 1.2.

Таблица 1.2

Параметр	Значение
Число дискретных входов	14
Тип	Сток / Исток
Номинальное напряжение	24 В постоянного тока при 4 мА
Логическая 1 (мин.)	15 В постоянного тока при 2,5 мА
Логический 0 (макс.)	5 В постоянного тока при 1 мА
Потребление тока	4 мА/вход
Электрическая развязка (полевая сторона относительно логики)	500 В переменного тока в течение 1 минуты
Потенциально развязанные группы	1
Число аналоговых входов	2
Тип и диапазон (слово данных)	Напряжение, 0 – 10 В (0 – 27648)
Переопределение (слово данных)	32512 – 32767
Разрешение	10 битов
Электрическая развязка (полевая сторона относительно логики)	Нет
Точность	3,0% / 3,5% от всего диапазона
Число выходов	10
Тип	Транзисторный – MOSFET
Диапазон напряжений	от 20,4 до 28,8 В пост. тока

Ниже представлены схемы подключения входных и выходных цепей ПЛК S7-1200 1214C (рисунок 1.4).

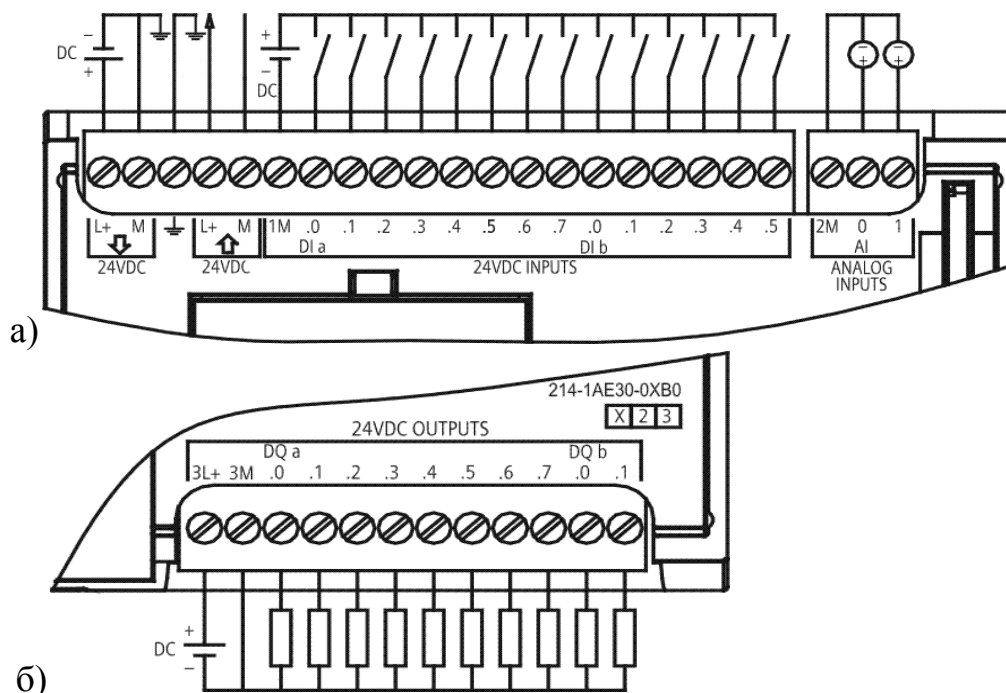


Рисунок 1.4- Схемы подключения входных (а) и выходных (б) цепей контроллера S7-1200 1214C

1.4 Цикл работы контроллера

Контроллер обрабатывает программу циклически. Цикл состоит из нескольких шагов, которые выполняются регулярно в строгой последовательности, и содержит следующие задачи (рисунок 1.3):

- считывание входов;
- обработка программы;
- обработка коммуникационных запросов;
- проведение самодиагностики в CPU;
- запись на выходы.



Рисунок 2.3 - Цикл CPU S7-200

1.4.1 Считывание цифровых входов.

В начале цикла считываются текущие значения цифровых входов и затем записываются в область отображения процесса на входах. В области отображения процесса на входах для CPU предусмотрены разделы по восемь битов (один байт). Если CPU или модуль расширения предоставляет в распоряжение физический вход не для каждого бита зарезервированного байта, то нельзя назначать такие биты последующим модулям в цепи «ввода-вывода» и использовать их в Вашей программе. Свободные входы в области отображения процесса обнуляются CPU в начале каждого цикла.

CPU автоматически не актуализирует аналоговые входы в качестве части цикла, а также не записывает в память отображение для аналоговых входов. Доступ к аналоговым входам должен производиться непосредственно через Вашу программу.

1.4.2 Обработка программы.

В этой фазе цикла CPU обрабатывает программу от первой до последней операции. Во время обработки главной программы или программы обработки прерываний можно прямо обращаться к входам и выходам и таким образом управлять ими. Если в программе используются прерывания, то программы обработки прерываний, поставленные в соответствие событиям прерываний, сохраняются как часть главной программы. Однако программы обработки прерываний не обрабатываются как составная часть цикла, а

обрабатываются только тогда, когда появляется событие прерывания (это возможно в любой точке цикла).

1.4.3 Обработка коммуникационных запросов.

В этой фазе цикла CPU обрабатывает все сообщения, принятые через коммуникационный порт.

1.4.4 Проведение самодиагностики в CPU.

В этой фазе цикла CPU проверяет программы в ПЗУ, память программ и состояние модулей расширения.

1.4.5 Запись на цифровые выходы.

В конце цикла значения из области отображения процесса на выходах записываются на цифровые выходы. В области отображения процесса на выходах для CPU предусмотрены разделы по восемь битов.

CPU автоматически не актуализирует аналоговые выходы в качестве части цикла, а также не записывает в память отображение для аналоговых выходов. Доступ к аналоговым выходам производится непосредственно через Вашу программу.

1.4.6 Прерывание цикла.

Если в программе используются прерывания, то программы обработки прерываний, поставленные в соответствие событиям прерываний, запоминаются как часть главной программы. Однако программы обработки прерываний не обрабатываются как составная часть нормального цикла, а обрабатываются только тогда, когда появляется событие прерывания (это возможно в любой точке цикла). CPU обрабатывает разблокированные прерывания асинхронно по отношению к циклу и выполняет программу обработки прерываний, когда появляется соответствующее событие прерывания. Обработка прерываний происходит в порядке их появления и в соответствии с их приоритетом.

1.4.7 Отображения процесса на входах и выходах.

Во время обработки программы доступ к входам и выходам происходит не прямо, а через соответствующие области отображения процесса. Имеются три важных основания для существования областей отображения процесса:

а) система в начале цикла опрашивает входы. Благодаря этому, значения этих входов синхронизируются и «замораживаются» на период обработки программы. После обработки программы через область отображения процесса актуализируются выходы. Это оказывает стабилизирующее воздействие на систему;

б) программа может обращаться к области отображения процесса гораздо быстрее, чем непосредственно к входам и выходам. Это ускоряет обработку программы;

в) входы и выходы являются битовыми объектами, доступ к которым должен производиться в битовом формате. К областям отображения процесса можно обращаться в формате бита, байта, слова и двойного слова. Поэтому области отображения процесса обеспечивают дополнительную гибкость.

Дополнительным преимуществом является то, что области отображения процесса достаточно велики для того, чтобы обрабатывать максимальное количество входов и выходов. Так как реальная система состоит из входов и выходов, то в области отображения процесса всегда есть неиспользуемые адреса, которые можно использовать как дополнительные внутренние маркеры.

1.4.8 Прямое управление входами и выходами.

С помощью операций прямого управления входами и выходами можно прямо обращаться к входу или выходу, хотя нормально в качестве источника и цели доступа к входам и выходам используются области отображения процесса. Если Вы обращаетесь прямо к входу, то соответствующий адрес в области отображения процесса на входах не изменяется. Если Вы обращаетесь прямо к выходу, то одновременно актуализируется соответствующий адрес в области отображения процесса на выходах.

2 Основы программирования контроллера

2.1 Редактор контактного плана (Ladder Logic)

Контроллеры S7-200, S7-1200 предоставляют различные типы команд, позволяющих решать широкий спектр задач автоматизации. Кроме того, ориентированное на ПК программное обеспечение для программирования STEP 7 предоставляет выбор различных редакторов, которые позволяют создавать управляющие программы с этими командами. Выбор типа редактора определяется самим пользователем. В расчетно-графической работе предлагается воспользоваться редактором Ladder Logic (контактный план). Редактор контактного плана (LAD) STEP 7 позволяет формировать программы, имеющие сходство с электрической монтажной схемой. По существу, программы в контактном плане позволяют CPU эмулировать протекание электрического тока от источника питания через ряд логических состояний входов, которые, в свою очередь, разблокируют логические состояния выходов. Логика обычно подразделяется на малые легко понимаемые сегменты, которые в англоязычной литературе часто называются rungs, т.е. ступеньки (в соответствии с английским названием контактного плана – Ladder Logic, что буквально означает «лестничная логика»), или networks – цепи. Программа выполняется по «цепям» слева направо и затем сверху вниз, как предписано программой. Как только CPU достигает конца программы, он начинает снова с вершины программы. Графическое

представление часто более легко понимается и является популярным во всем мире. Контактный план прост в использовании для начинающих программистов.

Различные команды в LAD представляются графическими символами и включают три основные формы (рисунки 2.1, 2.2):

а) контакты (I0.1, I.02) представляют логические состояния «входов», аналогичных выключателям, кнопкам, внутренним условиям и так далее;

б) катушки (Q0.4) обычно представляют логические результаты «выходов», аналогичных лампам, пускателям электродвигателей, промежуточным реле, состояниям внутренних выходов и так далее;

в) блоки представляют дополнительные команды такие, как таймеры, счетчики или математические команды (рисунок 2.2).

В LAD программа делится на сегменты (Network). Сегмент – это упорядоченное расположение контактов, катушек и блоков, которые соединены между собой и образуют цепь между левой и правой шиной. Step 7 позволяет создавать комментарии к сегментам программы.

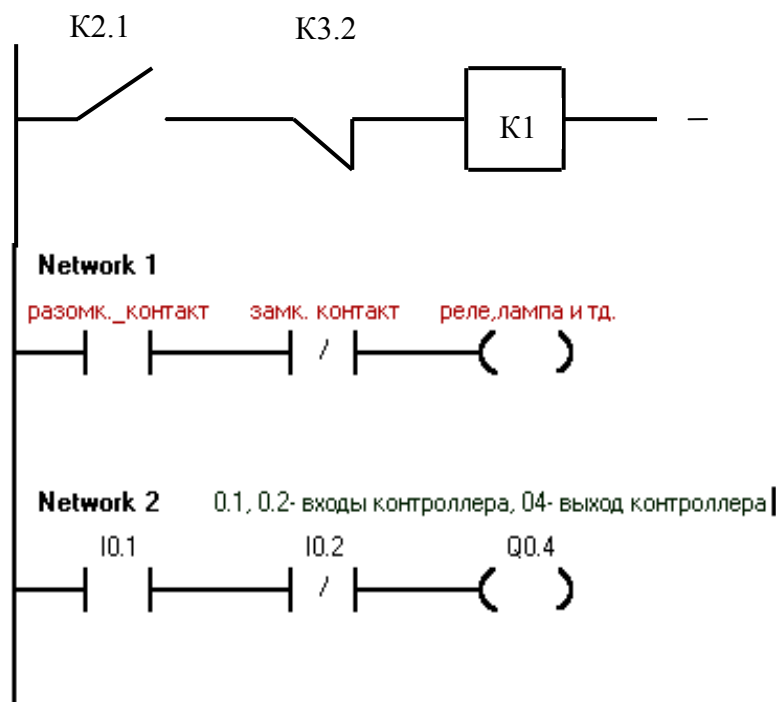


Рисунок 2.1 - Элементы электрической схемы и соответствующие им графические символы LAD при реализации программы

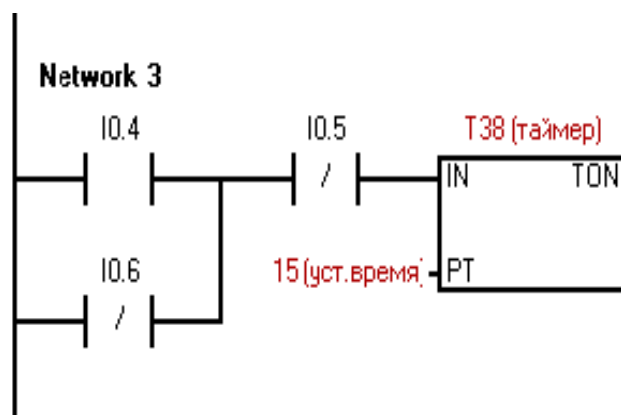


Рисунок 2.2 - Пример записи программы в форме LAD

3 Система команд микроконтроллера

3.1 Состав команд

Команды микроконтроллера можно разделить по функциональному назначению на следующие группы:

- а) «ввод - вывод»;
- б) управление программой;
- в) управление таймерами и счетчиками;
- г) арифметические операции;
- д) контроль и редактирование программы;
- е) тестовый контроль функциональных блоков.

3.2 Структура команд

Рисунок 3.1 иллюстрирует формат команды LAD в виде разомкнутого контакта. В LAD замкнутые контакты пропускают через себя сигнал к следующему элементу, а разомкнутые контакты блокируют сигнал.

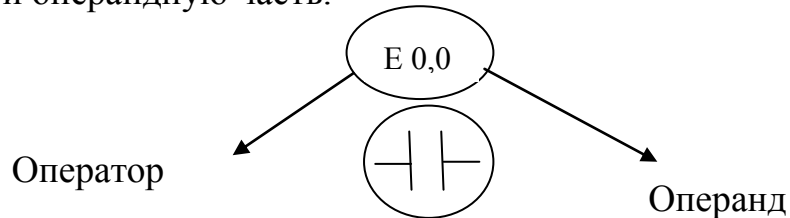
Команда содержит оператор (сам контакт) и операнд E0.0, иначе вход контроллера I0.0.

Значительная часть команд в LAD представлена в виде блоков (таймеры, счетчики и др.) на рисунке 2.2.

Значительная часть команд в LAD представлена в виде блоков (таймеры, счетчики и др.) на рисунке 2.2.

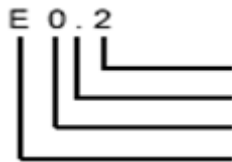
Управляющие инструкции

Управляющие инструкции (команды) являются наименьшими блоками в пользовательской программе для контроллера. Инструкция содержит операторную и операндную часть.



<p>(Что делать?)</p> <p>Операторная часть инструкции (в этом случае логической операции «И») определяет функцию, которая выполняется сама инструкция.</p>	<p>(Над чем делать?)</p> <p>Операторная часть инструкции (в данном случае вход 0.0) содержит дополнительную инструкцию для команды. Она содержит идентификатор операнда и параметр.</p>		
	<table border="1" style="width: 100%;"> <tr> <td style="text-align: center;"> <p>Идентификатор</p> <p>операнда определяет область в PLC. В примере операция выполняется над входом. Другие области включают выходы и маркеры.</p> </td> <td style="text-align: center;"> <p>Параметр</p> <p>является адресом операнда. Он содержит, например, адрес бита и байта.</p> </td> </tr> </table>	<p>Идентификатор</p> <p>операнда определяет область в PLC. В примере операция выполняется над входом. Другие области включают выходы и маркеры.</p>	<p>Параметр</p> <p>является адресом операнда. Он содержит, например, адрес бита и байта.</p>
<p>Идентификатор</p> <p>операнда определяет область в PLC. В примере операция выполняется над входом. Другие области включают выходы и маркеры.</p>	<p>Параметр</p> <p>является адресом операнда. Он содержит, например, адрес бита и байта.</p>		

Структура операнда

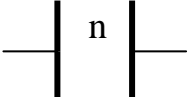


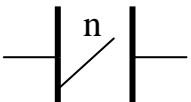
Адрес бита:
Точка:
Адрес байта:
Идентификатор области
Возможные области:

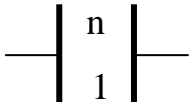
Номер бита в байте (0..7)
Отделяет адрес байта и номер бита
Номер группы из 8 бит

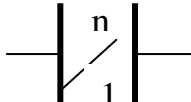
Входы, выходы, внутренние маркеры, специальные маркеры, маркеры переменных

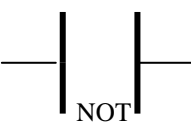
3.2.1 Операции над входами.

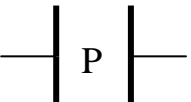
 Замыкающий контакт включен, если значение бита с адресом n равно 1.


 Размыкающий контакт включен, если значение бита с адресом n равно 0.

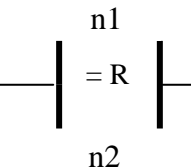
 Замыкающий контакт с непосредственным доступом включен, если значение бита заданного входа n равно 1.

 Размыкающий контакт с непосредственным доступом включен, если значение бита заданного входа n равно 0.

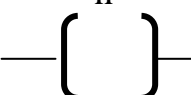
 Контакт «NOT» изменяет состояние потока сигнала. Если поток сигнала достигает контакта NOT, то он у контакта останавливается. Если поток сигнала не достигает контакта NOT, то у контакта создается поток сигнала.

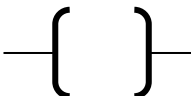
 Контакт «Обнаружение» нарастающего фронта пропускает поток сигнала в течение цикла при каждом нарастающем фронте.

 Контакт «Обнаружение» спадающего фронта пропускает поток сигнала в течение цикла при при каждом спадающем фронте.

 Операция «Сравнение» действительных чисел сравнивает два значения n1 и n2 друг с другом. Вы можете производить следующие сравнения: n1 = n2, n1 >= n2 и n1 <= n2. Операнды: n1, n2: VD, ED, AD, MD, SMD, AC. В LAD контакт замкнут, если результатом сравнения является истина.

3.2.2 Операции над выходами.

 Если выполняется операция «Присваивание», то заданный параметр (n) включается.

 Если выполняется операция «Прямое присваивание» значения биту, то заданный физический выход (n) непосредственно включается.

$\left[\begin{array}{c} S_BIT \\ S \\ N \end{array} \right]$	<p>Если выполняются операции «Установка и Сброс», то заданное количество (N) входов или выходов, начиная с S_BIT, включается или выключается.</p>
$\left[\begin{array}{c} S_BIT \\ R \end{array} \right]$	<p>Операнды: S_BIT: E, A, M, SM, T, Z, V, S N: EB, AB, MB, SMB, VB, AC.</p>
$\left[\begin{array}{c} S_BIT \\ S_1 \\ N \end{array} \right]$	<p>Если выполняются операции «Прямая установка» и «Прямой сброс», то заданное количество (N) входов или выходов, начиная с S_BIT, включается или выключается.</p>
$\left[\begin{array}{c} S_BIT \\ R_1 \end{array} \right]$	<p>Операнды: S_BIT: A N: EB, AB, MB, SMB, VB, AC.</p>

3.2.3 Команды управления программой.

$\left[\begin{array}{c} \\ END \end{array} \right]$	<p>Команда «Завершение программы» заканчивает работу главной программы в зависимости от результата предшествующей логической операции.</p>
$\left[\begin{array}{c} \\ STOP \end{array} \right]$	<p>Команда «STOP» заканчивает работу программы, немедленно, переводя CPU из режима RUN в режим STOP.</p>
$\left[\begin{array}{c} n \\ JMP \end{array} \right]$	<p>Команда «Переход на метку» выполняет ветвление программы к заданной метке перехода (n).</p>
$\left[\begin{array}{c} n \\ CALL \end{array} \right]$	<p>Команда «Вызов подпрограммы» передает управление подпрограмме (n).</p>
$\left[\begin{array}{c} \\ RET \end{array} \right]$	<p>Команда «Условное окончание» подпрограммы завершает подпрограмму в зависимости от результата предшествующей логической операции.</p>
$\left[\begin{array}{c} \\ RET \end{array} \right]$	<p>Любая подпрограмма должна заканчиваться командой «Абсолютное окончание подпрограммы».</p>

3.3 Таймеры и счетчики

3.3.1 Таймеры.

С помощью таймеров можно выполнять функции, управляемые временем. S7-200 предоставляет две разные таймерные операции: запуск таймера как формирователя задержки включения (TON) и запуск таймера как формирователя задержки включения с запоминанием (TONR). Эти два типа таймеров (TON и TONR) (рисунок 3.1) различаются по своей реакции на состояние входа разблокировки. Оба таймера TON и TONR отсчитывают значение времени, когда активизирован вход разблокировки. При выключенном входе разблокировки оба таймера не работают, однако, в то время как таймер TON автоматически сбрасывается, таймер TONR сохраняет свое последнее значение времени, а не сбрасывается. Поэтому лучше всего использовать таймер TON, когда нужен отдельный интервал времени, и использовать таймер TONR, когда необходимо накапливать несколько интервалов времени.

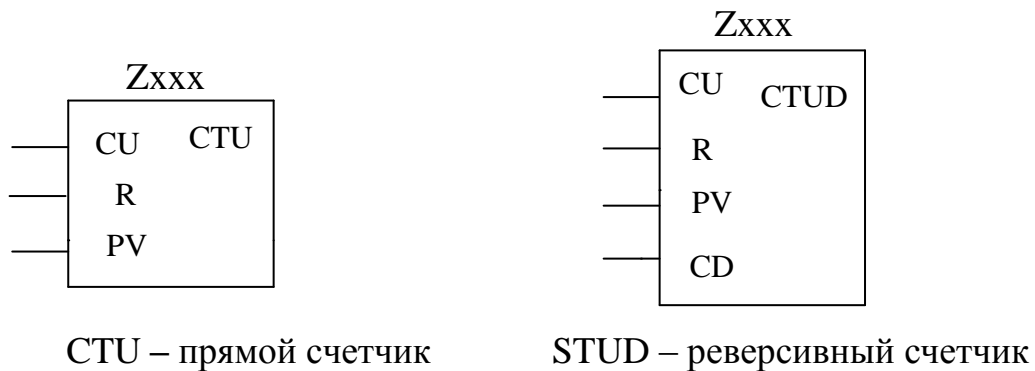


Рисунок 3.1 – Таймеры

3.3.2 Счетчики.

Операция «Прямой счет» выполняет счет в прямом направлении при нарастающем фронте на входе прямого счета (CU) до достижения максимального значения. Если текущее значение (Z_{xxx}) \geq предварительно установленному значению (PV), то включается бит счетчика (Z_{xxx}). Счетчик сбрасывается, если активизируется вход сброса.

Операция «Реверсивный счет» выполняет счет в прямом направлении при нарастающем фронте на входе прямого счета (CU). При нарастающем фронте на входе обратного счета (CD) выполняется счет в обратном направлении. Если текущее значение (Z_{xxx}) \geq предварительно установленному значению (PV), то включается бит счетчика (Z_{xxx}). Счетчик сбрасывается, если активизируется вход сброса.



Если счетчик сбрасывается посредством операции “Сброс”, то сбрасывается как бит счетчика, так и текущее значение счетчика. Обращение к текущему значению, а также к биту счетчика осуществляется с помощью номера счетчика.

Указание.

Так как каждый счетчик обладает собственным текущим значением, то не назначайте одинаковый номер нескольким счетчикам (прямой и реверсивный счетчики имеют доступ к одному и тому же текущему значению).

4 Порядок выполнения расчетно-графических работ

4.1 Техническое задание

Для выполнения расчетно-графической работы студентам выдается задание в виде алгоритма управления или технологической схемы автоматизируемого процесса с описанием последовательности работы элементов схемы, а также в виде принципиальной релейно-контакторной схемы, требующей модернизации.

В результате выполнения расчетно-графической работы студент должен представить пояснительную записку содержащую:

- а) исходное задание;
- в) программу решения поставленной задачи на языке, присущем данному контроллеру с подробным описанием ее работы;
- д) схему подключения контроллера SIMATIC S7-200, S7-1200 к датчикам и исполнительным элементам технологического процесса (электроприводы, пускатели, контакторы и тд.).

4.2 Тематика расчетно-графических работ

В процессе изучения курса студент выполняет три расчетно-графические работы. Первая работа направлена на приобретение навыков программирования контроллеров по заданному алгоритму. Вторая и третья работы связаны с автоматизацией заданного технологического процесса или с

модернизацией релейно-контактных схем управления электроприводами и механизмами.

4.3 Последовательность выполнения расчетно-графической работы

На основании алгоритма, схемы технологического процесса и технического задания необходимо проанализировать поставленную задачу с учетом технических возможностей S7-200, S7-1200. Обосновать достаточность числа входов и выходов для подключения исполнительных элементов и датчиков, выбрать контроллер для решения поставленной задачи. Далее дается описание последовательности работы элементов электрической схемы и датчиков, затем на основании алгоритма работы с использованием компьютерной программы Step 7 составляется программа на языке LAD. Для этого необходимо, войдя в программу, открыть папку «новый проект» и используя символику редактора набрать программу в виде цепей Network.

Заканчивается работа разработкой принципиальной электрической схемы подключения контроллера к элементам схемы и датчикам на основании технического описания контроллеров и раздела 1.2, 1.3 настоящего указания.

4.4 Примеры программирования контроллеров

4.4.1 Пример программы с применением таймеров.

Допустим, согласно заданию, некоторую нагрузку (Q0.0) необходимо подключать к сети на время - t_p , а затем отключать на время - t_n , при этом процесс носит циклический характер. Команда на включение и отключение процесса подается с пульта управления выключателем – К (I0.7).

Для решения поставленной задачи можно применить два таймера: один для задания времени работы – t_p , другой для задания времени паузы – t_n и их управляющие контакты. На рисунке 4.1 представлена программа, реализующая заданный алгоритм.

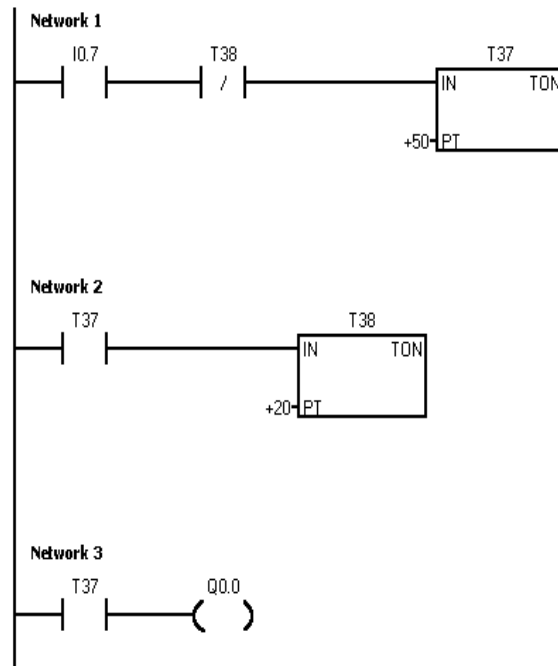


Рисунок 4.1

4.4.2 Пример программы управления движением тележки.

В качестве примера представлена программа управления 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 секунд и возвращается назад в положение П0. В положении П0 немедленно возникает реверс привода тележки. Тележка вновь движется в положение П1, стоит в нем 5 секунд 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 П1↑ и память РП о нажатии на кнопку «Пуск»:

$$P_{II} = (Пуск + P_{II}) \cdot \overline{КЦ}. \quad (4.1)$$

Команда на перемещение тележки вперед (В) возникает в положении П0, если отсутствует сигнал счетчика СЧ, и сохраняется при наличии сигнала памяти пуска РП до достижения положения П1. Этому описанию соответствует логическое уравнение:

$$B = (П0 \cdot \overline{СЧ} + B) \cdot \overline{П1} \cdot P_{II}. \quad (4.2)$$

Команда на перемещение тележки назад (Н) возникает через 5 секунд нахождения в положении П1 и сохраняется при наличии переменной РП до достижения положения П0. Тогда логическое уравнение для команды Н имеет вид:

$$H = (П1 \uparrow + H) \cdot \overline{П0} \cdot P_{П}. \quad (4.3)$$

Сигнал конца цикла (КЦ) возникает в положении П0 при наличии сигнала счетчика СЧ о завершении 10 циклов перемещения в положение П1 и сигнала РП. Сигнал КЦ сохраняется до очередного нажатия на кнопку “Пуск”. Тогда логическое уравнение для рассматриваемого сигнала имеет вид:

$$КЦ = (П0 \cdot СЧ \cdot P_{П} + КЦ) \cdot \overline{Пуск}. \quad (4.4)$$

В ниже приведенной программе (рисунок 4.2) использован счетчик на вычитание. При нажатии на кнопку «Пуск» в счетчик вводится уставка 10. Счетчик считает число циклов перемещения в положение П1, 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:

$$B = (П0 \cdot \overline{СЧ} + B) \cdot \overline{П1} \cdot P_{П}, \quad (4.5)$$

$$КЦ = (П0 \cdot \overline{СЧ} \cdot P_{П} + КЦ) \cdot \overline{Пуск}. \quad (4.6)$$

Сброс счетчика производится сигналом конца цикла КЦ.

В таблица 4.1 представлена адресация команд и сигналов для ПЛК S7-1200.

Таблица 4.1

Команда, сигнал	П0	П1	Пуск	В	Н	КЦ	РП	СЧ	П1↑
Адрес	I0.0	I0.1	I0.2	Q0.1	Q0.2	Q0.3	M0.0	M0.1	M0.2
Символ	П0	П1	Пуск	Вперед	Назад	КЦ	Память	СЧ	Задержка

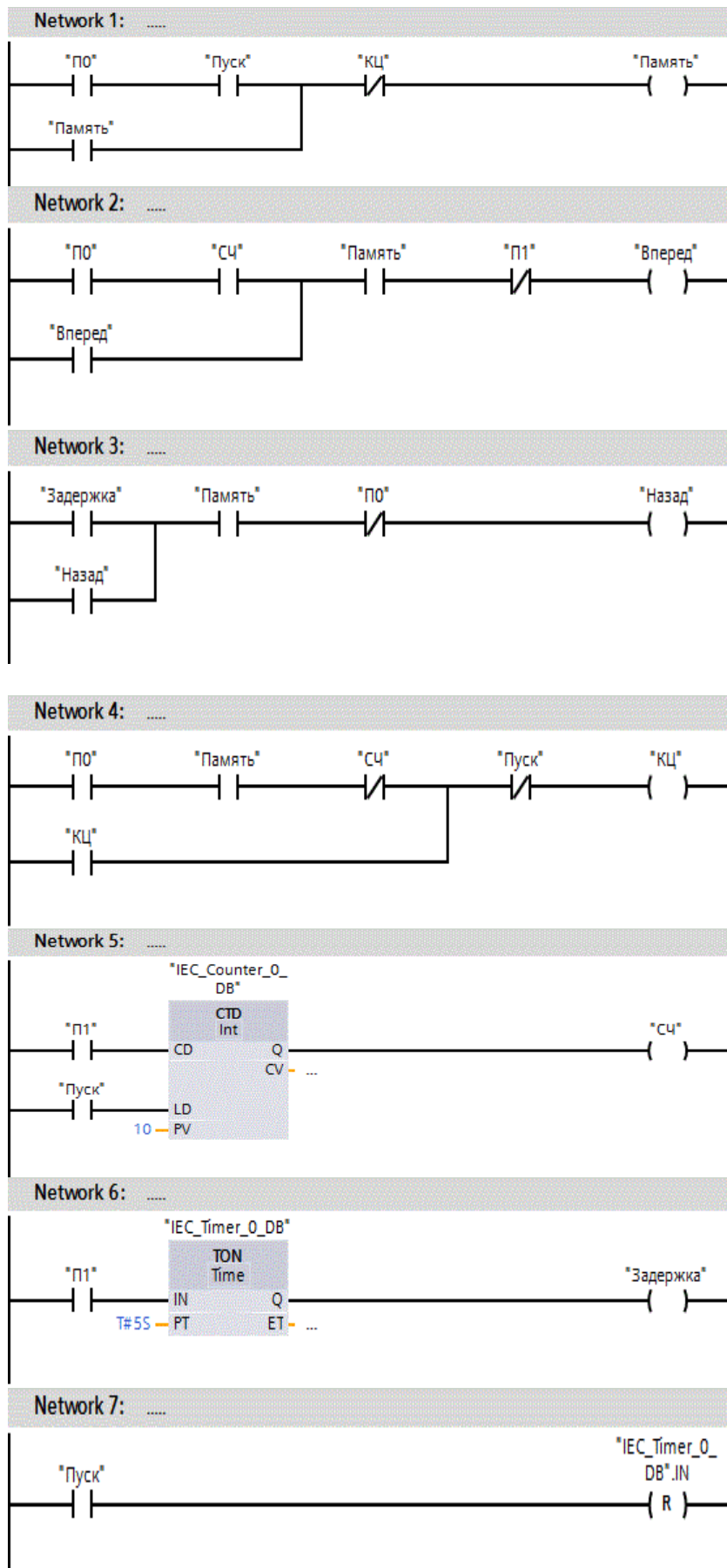


Рисунок 4.2 - Программа управления движением тележки

4.4 Задание к РГР1

Цель: освоить основные команды языка LAD и получить навыки разработки простых программ.

Разработать программу, обеспечивающую заданную временную последовательность включения электрических ламп, подсоединенных к выходам контроллера (Q0.0...) (рисунки 1.2, 1.4). Для запуска и останова программы и организации циклов использовать входные дискретные сигналы (I.0...). Вариант временной диаграммы задается преподавателем.

4.5 Задание к РГР 2

Цель: освоить принципы программного управления частотой вращения регулируемого электропривода.

Механизм рабочей машины (РМ) приводится в движение от асинхронного электродвигателя, управляемого преобразователем частоты (ПЧ), согласно заданной преподавателем временной диаграммы. При этом для задания необходимой частоты вращения, пуска и реверса используются входы ПЧ, позволяющие выполнять указанные действия, при подаче на соответствующий вход дискретного сигнала от управляющего контроллера (рисунок 4.2).

Требуется на основе программного обеспечения Step-7 и языка LAD разработать:

- 1) Программу управления двигателем.
- 2) Принципиальную электрическую схему подключения контроллера к управляющим входам ПЧ (рисунки 1.2, 1.4).

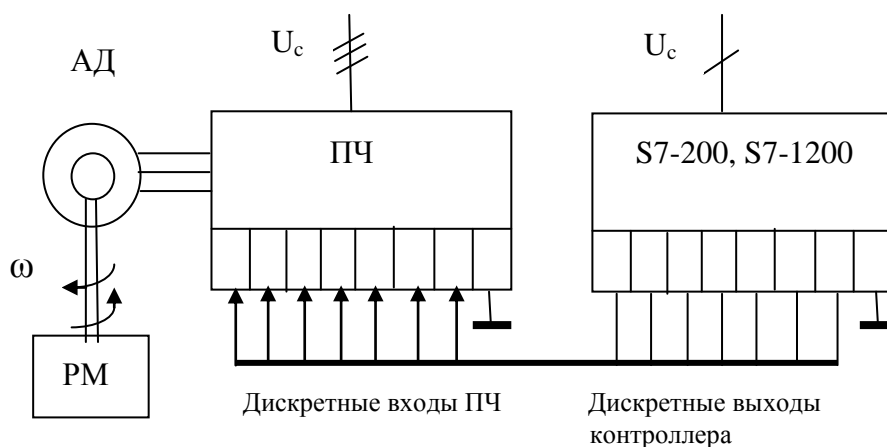


Рисунок 4.2

4.6 Задание к РГР 3

Цель: освоить программирование контроллера для решения задач автоматизации технологических процессов и модернизации систем управления электроприводами.

Расчетно-графическая работа 3 выдается в виде задания на модернизацию релейно-контакторной схемы управления электродвигателем, либо в виде задачи на автоматизацию технологического процесса. Ниже представлены возможные варианты. Студент решает одну из задач исходные данные для которой выдаются преподавателем.

4.6.1 Модернизация релейно-контакторной схемы пуска электродвигателя.

Для электрической релейно-контакторной схемы пуска электродвигателя постоянного тока в функции времени (рисунок 4.3) разработать аналог реализации с применением контроллера SIMATIC S7-200:

1) Используя словесный алгоритм пуска электродвигателя постоянного тока и редактор LAD, составить программу работы контроллера, реализующую пуск двигателя по заданному алгоритму. Произвести расчет выдержек времени таймеров для каждой пусковой ступени.

2) Составить схему подключения контроллера к источникам питания, исполнительным элементам (контакторы, пускатели, реле) и датчикам.

3) Проверить работу программы на лабораторном стенде.

Работа релейно-контакторной схемы

Одним из способов автоматического пуска электродвигателей постоянного тока является пуск в функции времени с ограничением тока с помощью пусковых резисторов, включаемых последовательно с якорем двигателя. Такая схема, предусматривающая также и динамическое торможение, представлена на рисунке 2.1. Схема содержит контактор пуска КМ1, контакторы ускорения КМ2 и КМ3, контактор торможения КМ4, реле времени К2, К4, реле динамического торможения К5. Для защиты электродвигателя от аварийных режимов в схеме предусмотрены реле максимального тока К1 и реле обрыва поля К3. Ниже представлен алгоритм работы схемы в режиме пуска и торможения.

При подаче питания на силовую схему и схему управления автоматом QF срабатывает реле К3 и реле времени К4. Реле К3 замыкает свой контакт К3.1 в цепи контактора КМ1 и тем самым подготавливает цепь пуска. Реле времени размыкает свой контакт в цепи контактора ускорения КМ2. При нажатии кнопки SB2 срабатывает контактор КМ1 и своим силовым контактом КМ1.1 подает напряжение на электродвигатель, в то же время меняют свое состояние все его контакты в цепях управления. По цепи электродвигателя

начинает протекать ток, вызывая срабатывание реле времени K2 и размыкание контакта K2.1 в цепи контактора ускорения KM3. С размыканием контакта KM1.3 в цепи реле времени K4 начинается отсчет выдержки времени работы двигателя на первой ступени пуска с двумя пусковыми резисторами R1, R2. По окончании выдержки времени K4 замыкает свой контакт в цепи контактора KM2, который своим контактом KM2.1 шунтирует пусковой резистор R3, и двигатель продолжает разгоняться на второй ступени.

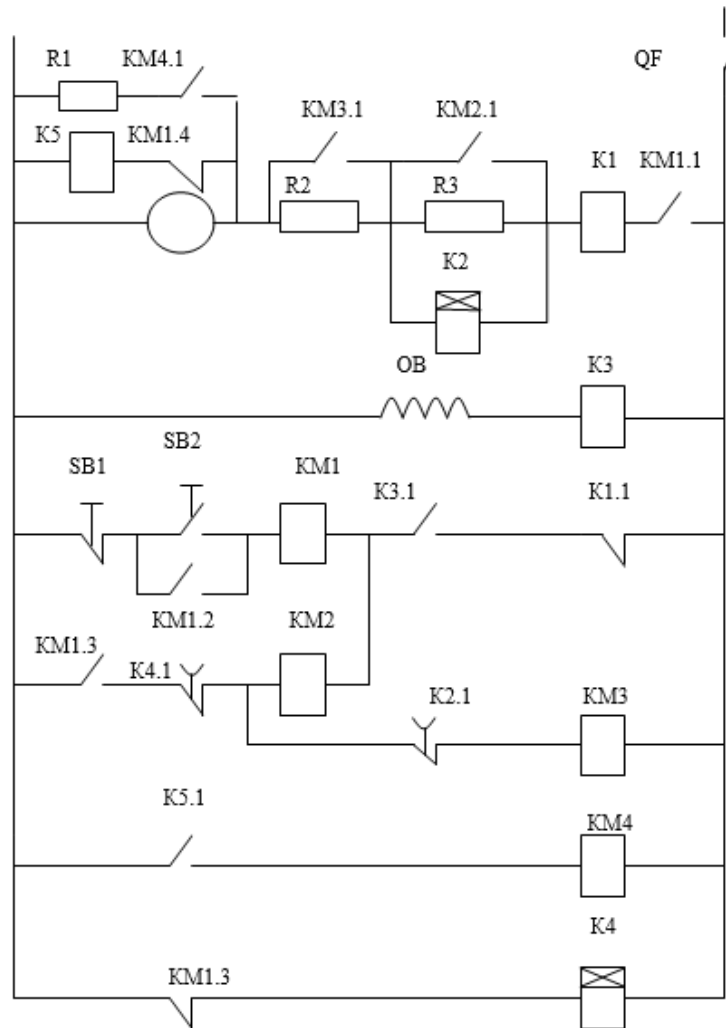


Рисунок 4.3

Одновременно теряет питание реле времени K2 и начинается отсчет его выдержки времени. По истечении выдержки времени замыкается его контакт в цепи контактора KM3, который своим контактом KM3.1 шунтирует резистор R3 и двигатель, продолжая разгоняться, выходит на естественную характеристику.

4.6.2 Автоматизация технологических процессов.

Задача 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».

Задача 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.

Задача 3. Автоматизировать перемещение пуансона штампованного прессы. 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 формы, 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.

Задача 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.

Задача 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 «Закрыть» 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.

Задача 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 «Закрыть» включается звонок, через 5 с включается привод двери на закрывание. При полном закрытии двери привод двери отключается, а звонок звенит еще 2 с.

Задача 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 максимально допустимому току НЭ, сигнал «0» – минимально допустимому току НЭ. Синтезировать схему, обеспечивающую алгоритм работы группы ТП, представленный в таблице 4.2.

Таблица 4.2

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.

Задача 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».

Задача 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 металл 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».

Задача 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 3 с электродвигатель автоматически вновь включается, и на повышенной скорости шатун идет в положение $\alpha = 0^\circ$. Стоит в нем 3 с, включается и движется к положению $\alpha = 120^\circ$ и на пониженной скорости до $\alpha = 0^\circ$ 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.

Задача 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 3 с, затем движется в П2, стоит там в течение 4 с, движется в положение П0, стоит там в течение 5 с, движется в П1, стоит там в течение 3 с 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».

Задача 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 конвейера 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».

Список литературы

- 1 Кудряшов В.С., Иванов А.В. и др. Основы программирования микропроцессорных контроллеров в цифровых системах управления технологическими процессами: учебное пособие. ВГУИТ- 2014. - 144 с.
- 2 Костров Б.В. Микропроцессорные системы и микроконтроллеры: учеб. Пособие – М.: Десс., 2007. - 320 с.
- 3 Погребницкий М. Я. Микропроцессорные системы управления электротехнологическими установками- МЭИ. – М.: МЭИ, 2003. - 72 с.
- 4 Белов А.В. Конструирование устройств на микроконтроллерах. Изд. Наука и техника. 2005.
- 5 Алексеев С.Б. Микропроцессорные средства и системы. Методические указания к выполнению лабораторных работ. АИЭС, 2005.
- 6 Hans Berger. Automating with SIMATIC- Nuremberg, March, 2000.

Сергей Борисович Алексеев
Тойгожинова Жанар Жумакановна

МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ АЭП

Методические указания по выполнению расчетно-графических работ
для студентов специальности 5В071800 - Электроэнергетика

Редактор Л.Т. Сластихина
Специалист по стандартизации Н.К. Молдабекова

Подписано в печать _____
Тираж 25 экз.
Объем 1,88 уч. – изд. л.

Формат 60x84 1/16
Бумага типографическая №1
Заказ _____ Цена 940 тг.

Копировательно – множительное бюро
некоммерческого акционерного общества
«Алматинский университет энергетики и связи»
050013 Алматы, ул. Байтурсынова, 126