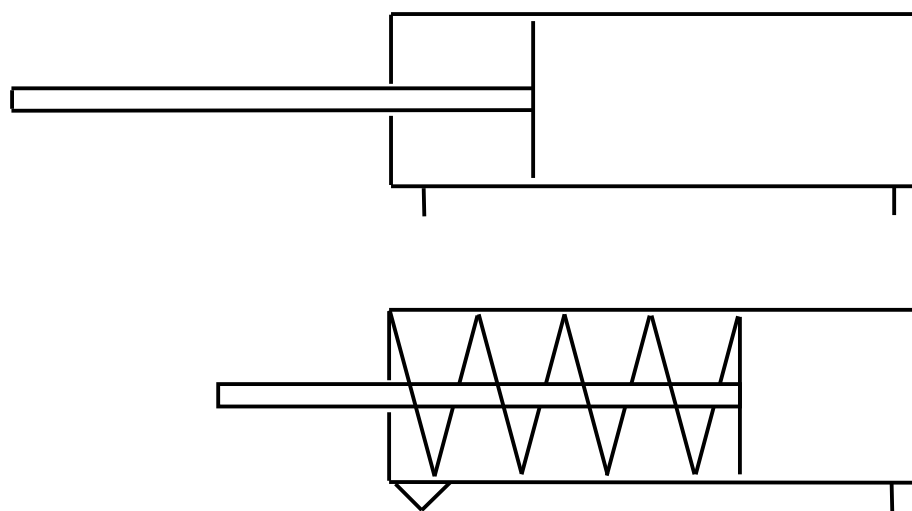


Новиков А.И., Дятлова Е.П.

Системы управления работой оборудования

Учебное Пособие



Санкт-Петербург

2013

УДК 676-5
ББК 35.77
Н 731

Новиков А.И., Дятлова Е.П.

Системы управления работой оборудования: учебное пособие /СПб
ГТУ РП. - СПб., 2013-83 с.

В пособии описаны технические средства пневмоавтоматики, широко используемые в системах контроля и управления в различных производственных процессах и оборудования, приведены алгоритмы работы оборудования, дано описание логического контроллера и описание программы SMS PneuDraw, используемой для проектирования пневматических схем. В пособии также даны рекомендации для выполнения лабораторной работы и курсового проекта.

Пособие предназначено студентам бакалаврам очной, заочной форм обучения, магистрам очной формы обучения по направлениям подготовки: 220400 «Управление в технических системах», 220700 «Автоматизация технологических процессов и производств», 220200, «Автоматизация и управление».

Авторы благодарят студентов 551 группы Зорину Е.В., Тимофеева И.В. и Пелевина А.В. за помощь в сборе и подготовке материалов.

Рецензенты: зав. кафедрой метрологии Горного института,
д-р техн. наук, профессор Б.Я.Литвинов
доцент кафедры информационно-измерительных
технологий и систем управления СПб ГТУ РП,
кандидат технических наук Г.А.Кнодель

Рекомендовано к изданию Редакционно-издательским советом
Университета в качестве учебного пособия.

© Новиков А.И, Дятлова Е.П., 2013

© Санкт-Петербургский

Государственный технологический университет
растительных полимеров, 2013

ВВЕДЕНИЕ

Автоматизация является важнейшим фактором роста производительности труда в промышленном производстве. К техническим средствам автоматизации относятся все устройства, входящие в систему контроля и управления, предназначенные для получения информации, ее передачи, хранения и преобразования, а также для осуществления управляющих и регулирующих воздействий на технологический объект управления.

В пособии уделяется внимание техническим средствам, использующим энергию сжатого воздуха. При разработке и внедрении систем автоматического контроля и управления технологическими процессами в химической, нефтяной, нефтехимической, лакокрасочной и ряде других отраслей промышленности целесообразно использование пневматических средств автоматизации.

Пневмоавтоматика – область технической кибернетики, охватывающая принципы и средства построения элементов, приборов и систем автоматического контроля и управления, использующих в работе различные эффекты газовой динамики. Пневмоавтоматика является одним из основных средств автоматизации производственных процессов. Выбор электрической или пневматической системы контроля или управления зависит от местных условий эксплуатации и предъявляемых к этой системе требований. В настоящее время разрабатываются оба вида технических средств, так как общий спрос на них примерно одинаков.

Разработка технических средств пневмоавтоматики значительно ускорилась после того, как был предложен элементный принцип конструирования пневматических приборов. Согласно этому принципу пневматический прибор собирается из пневматических элементов универсального назначения.

Сейчас трудно найти область промышленности, где не использовался бы сжатый воздух:

- автоматические линии по упаковке и дозированию сухих, жидких и пастообразных продуктов;
- металлообрабатывающие, деревообрабатывающие и ткацкие станки;
- линии по сборке автомобилей, электронных плат, часов;
- оборудование для тестирования, кузнечно-прессовое оборудование;
- строительная техника, металлургические прокатные станы;
- химические производства и многое другое.

Особое внимание в пособии сконцентрировано на дискретной логике и управлении циклическими процессами, что обусловлено тем, что в большинстве случаев элементы промышленного оборудования ориентированы на многократное выполнение элементарных повторяющихся операций.

Пособие предназначено для изучения дисциплины «Системы управления работой оборудования», проведения курсового проектирования и построения пневмосхем с использованием основных элементов пневмоавтоматики в программе **PneuDraw2.5**.

1. ЭЛЕМЕНТЫ ПНЕВМОАВТОМАТИКИ

1.1. Пневматические цилиндры

Пневматические цилиндры предназначены для преобразования энергии сжатого воздуха в механическое линейное перемещение. Пневмоцилиндры бывают одностороннего и двухстороннего действия, с односторонним или двухсторонним (проходным) штоком [1].

Конструкция цилиндра одностороннего действия показана на рис.1.1.1.



Рис. 1.1.1. Типовой цилиндр одностороннего действия

Цилиндр одностороннего действия имеет одно впускное отверстие (рис.1.1.2) и может совершать рабочее движение штока только в одном направлении. Это движение осуществляется при подаче сжатого воздуха в поршневую полость (ту, где отсутствует шток). Обратное в исходное положение шток поршня возвращается под действием установленной в цилиндре пружины или за счёт внешних сил. Штоковая полость (та, через которую проходит шток) постоянно сообщается с атмосферой. Цилиндр может быть «толкающего» или «тянущего» типа.

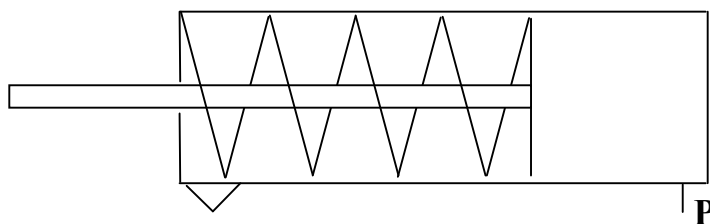


Рис. 1.1.2. Условное изображение цилиндра одностороннего действия

Для них характерно меньшее потребление воздуха по сравнению с цилиндрами двустороннего действия аналогичных размеров.

Цилиндры одностороннего действия применяются в зажимных и маркировочных устройствах, выталкивателях и т.п.

Такие цилиндры несколько проще конструктивно, так как в них не требуется уплотнения полости, сообщающейся с атмосферой. Соответственно и потери на трение в таких механизмах меньше, чем в цилиндрах двустороннего действия. Однако цилиндры одностороннего действия могут быть применены только в тех случаях, когда усилие, необходимое для возврата зажимных элементов в исходное положение, сравнительно невелико и может быть обеспечено пружиной. Кроме того, в цилиндрах одностороннего действия часть полезной мощности расходуется на сжатие возвратной пружины. Наличие пружины приводит также к увеличению общей длины цилиндра и ограничивает длину рабочего хода поршня.

Конструкция цилиндра двустороннего действия показана на **рис.1.1.3**.



Рис. 1.1.3. Типовой цилиндр двустороннего действия

В цилиндрах двустороннего действия осевое усилие передаётся как в направлении выталкивания, так и в направлении втягивания, поскольку давление сжатого воздуха прикладывается попеременно то с одной, то с другой стороны поршня (**рис.1.1.4**). Каждое из движений поршня

(втягивающее и толкающее) может быть использовано как рабочее. Осевое усилие при втягивании снижено вследствие меньшей полезной площади поршня, однако, данное обстоятельство имеет значение только в том случае, если цилиндр должен «тянуть» одну и ту же нагрузку в обоих направлениях.

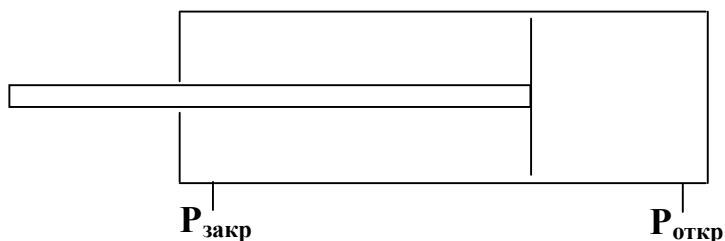


Рис. 1.1.4. Условное изображение цилиндра двустороннего действия

Гильза обычно изготавливается из бесшовной трубы, внутренняя рабочая поверхность которой может быть термически обработана и отполирована с целью сведения к минимуму трения. Торцевые крышки могут отливаться из алюминиевого сплава или изготавливаться из ковкого чугуна. Они могут крепиться либо при помощи стяжных шпилек или, при малых размерах цилиндров, вворачиваться в трубу гильзы на резьбе или обжиматься [1].

Для работы в химически агрессивных или опасных условиях корпус цилиндра может изготавливаться из алюминия, латуни, бронзы или нержавеющей стали.

1.2. Пневмораспределители

После выбора цилиндра встает вопрос, как им управлять, как осуществлять переключение, подачу воздуха и его сброс. Для этих целей служат пневматические распределители.

Пневматические распределители предназначены для изменения направления, пуска или остановки потоков сжатого воздуха в трубопроводах пневмосистемы [1].

На рис. 1.2.1 показано устройство пневмораспределителя.

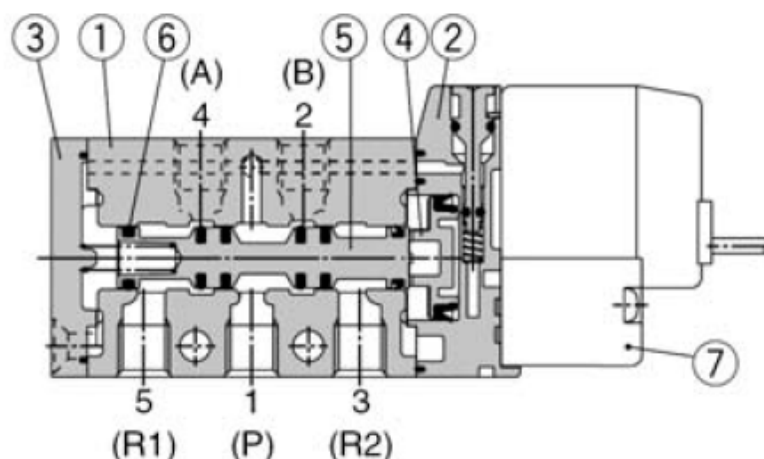


Рис. 1.2.1. Пневматический распределитель 5/2 с электропневматическим управлением:
1 - корпус; 2 - промежуточная деталь;
3 - концевая пластина; 4 - поршень управления;
5 - золотник; 6 - пружина; 7 - катушка;
А, В - выходные (рабочие) отверстия;
Р - отверстие подача питающего давления;
R1, R2 - выпускные отверстия (выхлопы).

Пневмораспределители характеризуются следующим показателями:

- количеством присоединительных отверстий;
- количеством позиций переключения;
- нормальной (не рабочей) позицией;
- способом управления распределителем.

Первые два пункта отображаются в виде цифр. Первая цифра обозначает количество присоединительных отверстий (исключая отверстие управления), а вторая цифра – количество возможных позиций переключения. Например: 3/2 пневмораспределитель нормально закрытый обозначает, что у него 3 присоединительных отверстия и два положения. В нормальном положении (когда нет сигнала управления) он закрыт (подача воздуха отсутствует).

1.3. Пневмоглушитель

Пневмоглушитель используется для снижения уровня шума при выхлопе в атмосферу отработанного воздуха из пневматических приводов и устройств. Пневмоглушители обязательно устанавливаются на выхлопные отверстия пневмораспределителей [1].

Материалом для изготовления корпуса служат латунь, медесодержащая сталь, полиэтилен. Для исполнения пористого элемента используются нержавеющая сталь, спеченная бронза или полиэтилен.

Принцип действия пневмоглушителей заключается в снижении энергии звуковых колебаний при прохождении отработанного воздуха сквозь пористый звукопоглощающий элемент. Простота конструкции обеспечивает высокую надежность эксплуатации. Пневмоглушители отличаются малыми габаритами и повышенной прочностью. Внешний вид пневмоглушителей и условное обозначение на схеме представлено на рис. 1.3.1.



Рис. 1.3.1. Пневмоглушитель

1.4. Конечные выключатели

Бесконтактные датчики положения используются в качестве конечных и путевых переключателей, реализуя обратную связь по положению штока цилиндра. Электрический сигнал поступает с датчика в систему управления (контроллер и т.п.), которая в зависимости от программы выдает управляющий сигнал на пневмораспределитель [1].

Герконовые датчики рекомендуются для общих случаев применения (геркон - герметический магнитно управляемый контакт [2]). Два язычковых контакта из железно-никелевого сплава заключены в стеклянный корпус.

Требуемый для замыкания контактов магнетизм крайне невелик из-за малого расстояния между язычками контакта. Магнитное кольцо на поршне приводит в действие датчик положения. Монтаж датчика выполняется непосредственно на гильзе цилиндра (рис. 1.4.1) с помощью хомута или скобы.

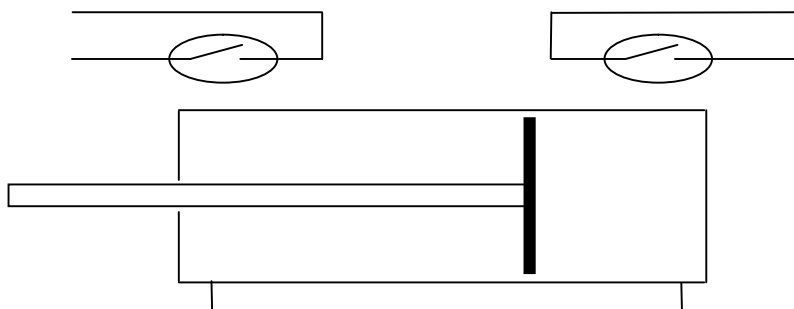


Рис. 1.4.1. Установка герконовых датчиков положения

Все типы датчиков положения оснащены системой защиты контактов, которая при использовании индуктивной нагрузки препятствует возникновению высоких пиков напряжения, которые привели бы к преждевременному обгоранию контактов герконов. Электронные датчики положения используются в случае больших токов (до 100 мА)[1].

1.5. Логический клапан «ИЛИ» (челночный клапан)

Данное устройство представляет собой клапан с тремя присоединительными отверстиями, два из которых служат входными для сигнального давления, а третье является выходным. Выход всегда соединен либо с одним из сигнальных входов (воздух под давлением подается только с одной стороны), либо с обоими сигнальными входами (воздух подается с обеих сторон). Если давление подается только на один вход, челночный затвор не допустит, чтобы поток воздуха с сигнальным давлением прошел через сигнальный вход на противоположной стороне. Устройство логического клапана представлено на рис.1.5.1.

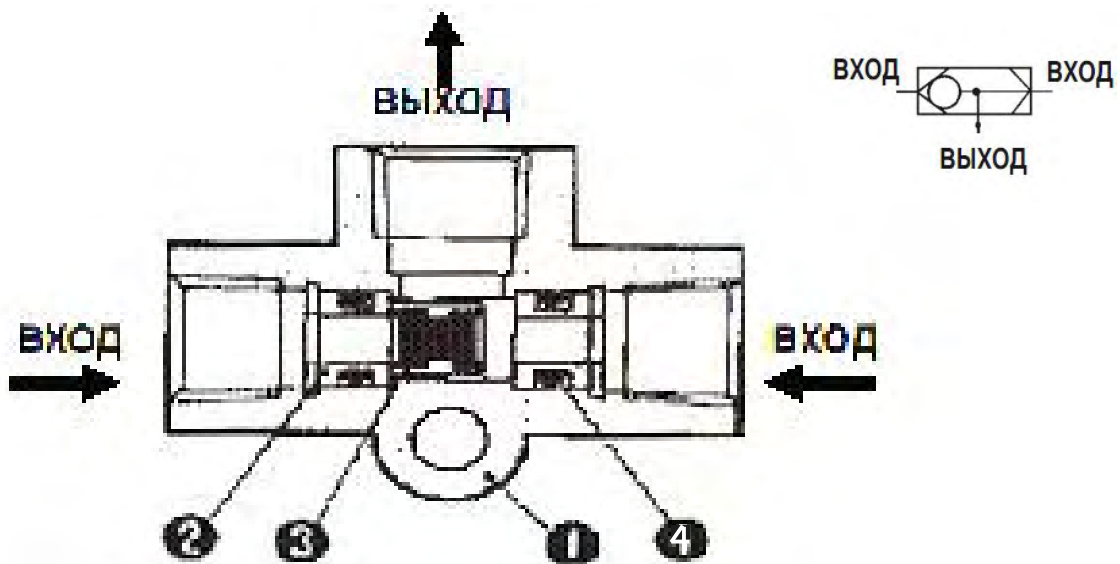


Рис. 1.5.1. Логический клапан «ИЛИ»:

1- корпус; 2 - седло клапана; 3 – клапан; 4 - кольцевая прокладка.

1.6. Логический клапан «И»

Данное устройство представляет собой клапан с тремя присоединительными отверстиями, два из которых служат входными для сигнального давления, а третье является выходным. Выход соединён всегда с двумя входными отверстиями. Если на одном из входов отсутствует давление, то клапан будет закрыт, даже если на втором входе имеется давление. На **рис. 1.6.1.** представлен принцип действия логического клапана «И».

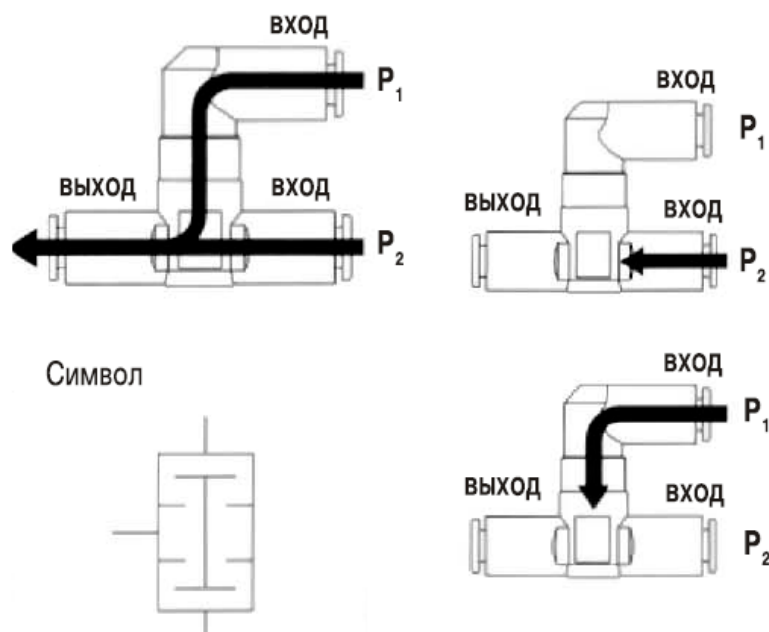


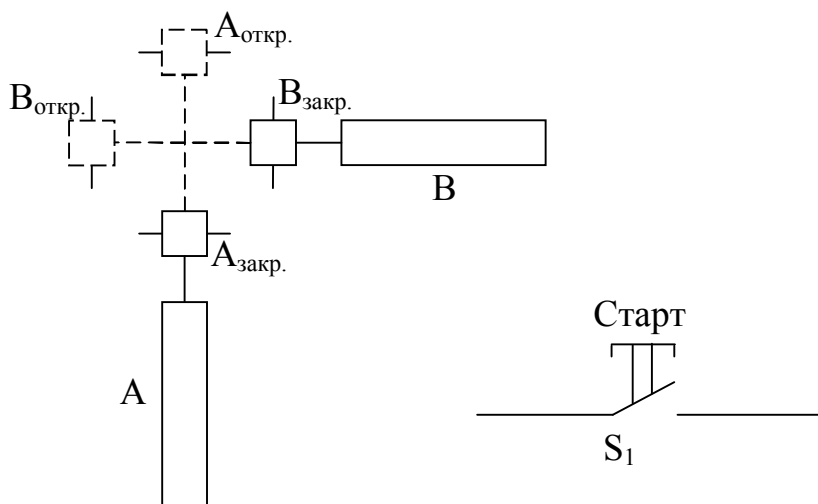
Рис. 1.6.1. Логический клапан «И»

2. АЛГОРИТМЫ РАБОТЫ ОБОРУДОВАНИЯ

Пример 1

Рассмотрим пример работы оборудования:

0. Начальное положение (*A – задвинут, B – задвинут*)

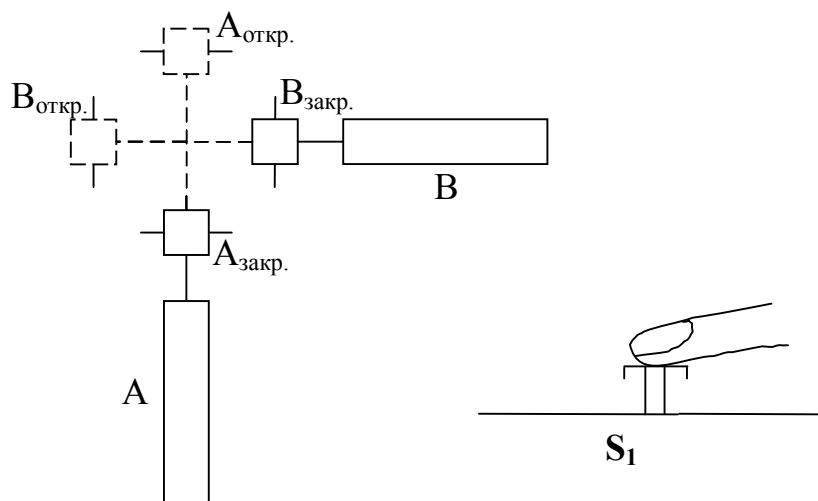


где A и B – пневматические цилиндры одностороннего действия с возвратной пружиной;

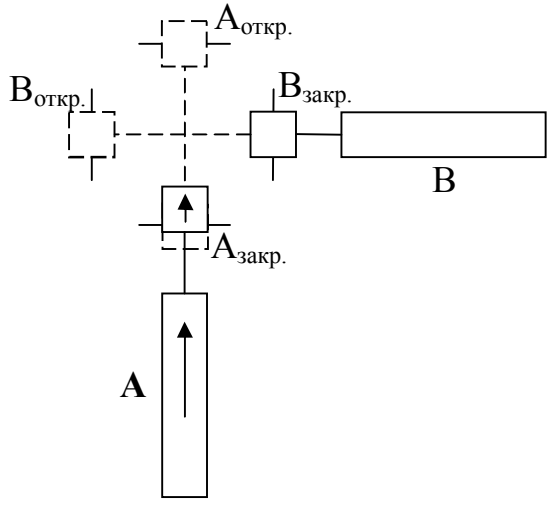
A_{откр.}, A_{закр.}, B_{откр.}, B_{закр.} – датчики конечных положений;

S₁ – кнопка «Старт».

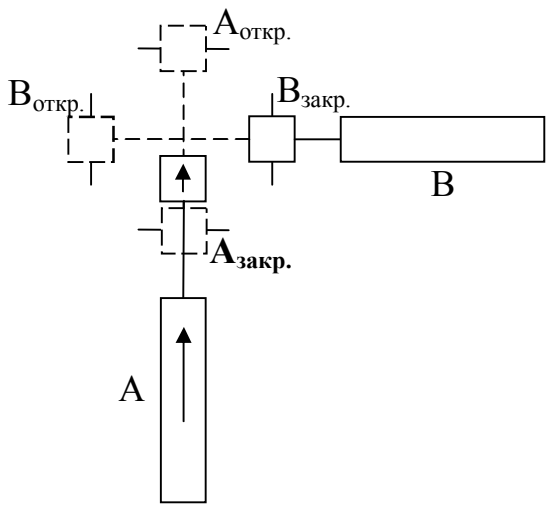
1. Нажатие кнопки «Старт» (S₁)



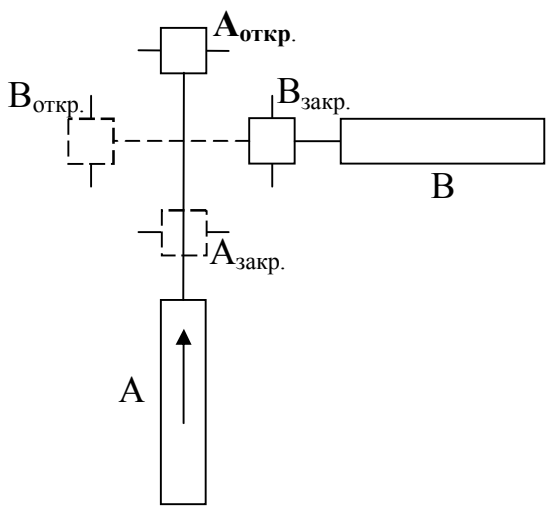
2. Начало движения цилиндра А



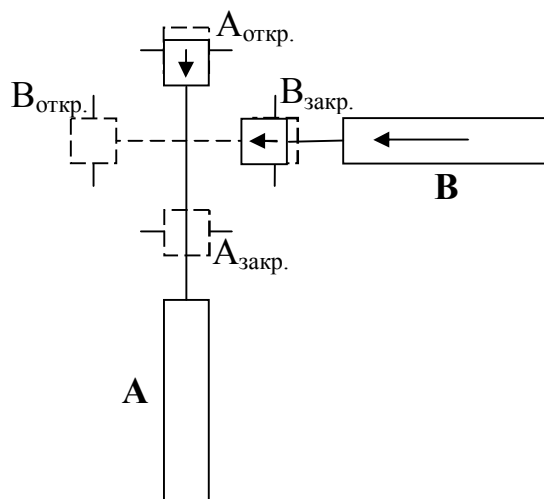
3. Выдвижение цилиндра А ($A_{закр.}$ - размыкается)



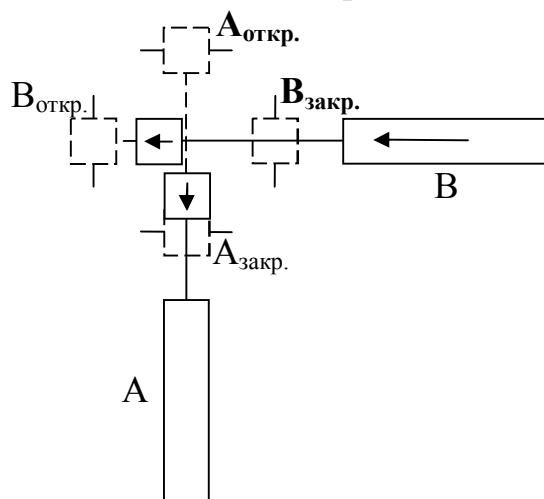
4. Цилиндр А выдвинут ($A_{откр.}$ - замыкается)



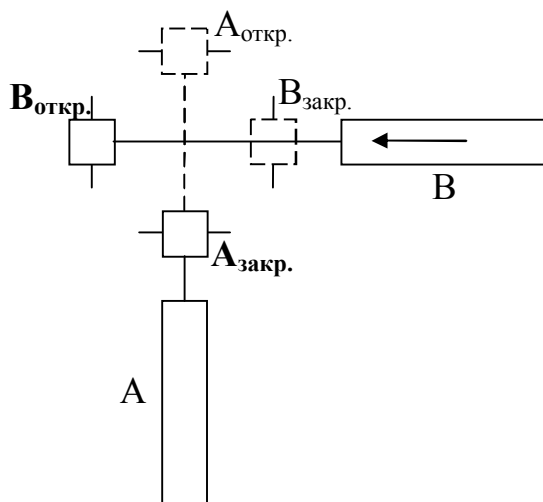
5. Подача сигнала на задвижение цилиндра А и выдвижение цилиндра В



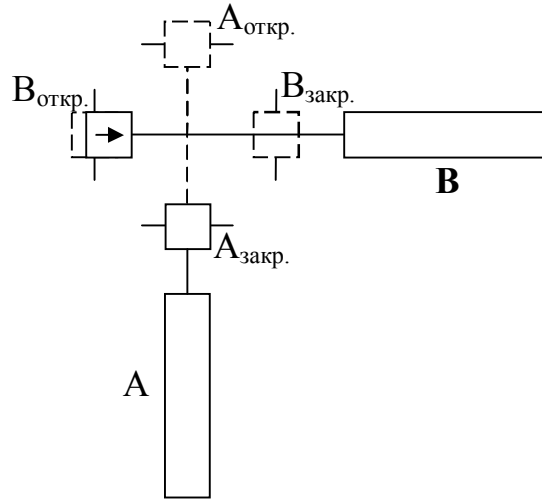
6. Цилиндр А задвигается, цилиндр В выдвигается ($A_{откр.}$ и $B_{закр.}$ замыкаются)



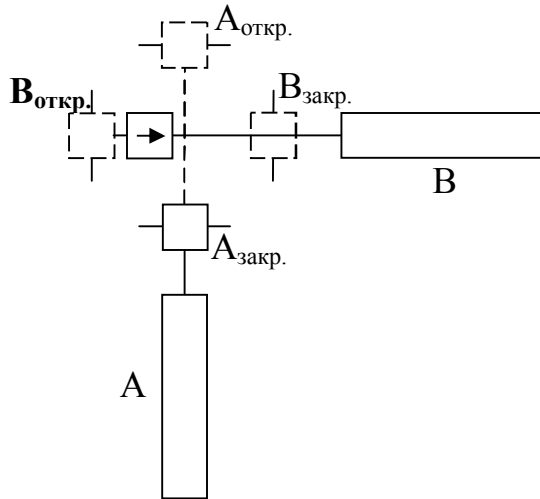
7. Цилиндр А задвинут, цилиндр В выдвинут ($A_{закр.}$ и $B_{откр.}$ замыкаются)



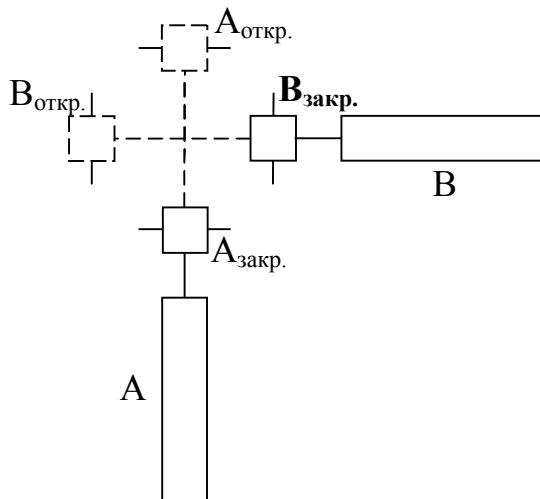
8. Подача сигнала на задвижение цилиндра В



9. Цилиндр В задвигается (*V_{откр.} размыкается*)



10. Цилиндр А задвинут, цилиндр В задвинут (*начальное положение*)



2.1. Циклограмма

Циклограмма (цикловая диаграмма) – графическое изображение циклического процесса, характеризующее изменения состояния оборудования во времени [3].

Дискретные сигналы имеют набор фиксированных уровней сигнала. Как правило, их два: **единица** (верхнее положение – есть сигнал) и **ноль** (нижнее положение – сигнал отсутствует), которые также можно интерпретировать как «Да» и «Нет», «Включен» и «Выключен», «Открыт» и «Закрыт» и т.п.

Составим циклограмму (рис. 2.1.1) для рассмотренного ранее примера 1. При составлении циклограммы сверху указываются такты (номера шагов, а в рассмотренном примере это также номера рисунков), а слева перечисляются все сигналы, причем входные сигналы (с датчиков) расположены выше выходных сигналов на исполнительные механизмы (ИМ).

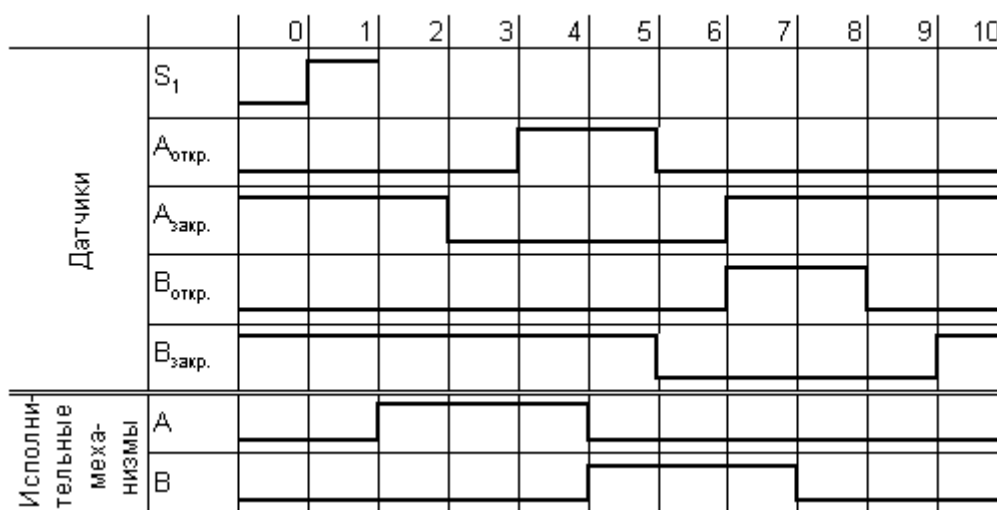


Рис. 2.1.1. Циклограмма

Пустые клетки означают «безразличные» состояния. Например, повторные нажатия кнопки запуска (S₁) при работающей системе ни к чему не приведут.

После завершения цикла система возвращается в исходное положение и готова к повторным запускам.

2.2. Последовательность действий

Система регулирования устанавливается на объекте для автоматического управления исполнительными механизмами (ИМ), в зависимости от текущего состояния оборудования.

Рассмотрим полученную ранее циклограмму (рис 2.1.1), построенную для примера 1. В данном случае у нас имеются два выходных параметра – исполнительные механизмы А и В, которыми необходимо управлять при помощи четырех датчиков конечных положений ($A_{откр.}$, $B_{откр.}$, $A_{закр.}$, $B_{закр.}$) и кнопки запуска S_1 .

Для построения последовательности действий необходимо:

1. Выявить (рис 2.2.3.А) все моменты переключения (рис 2.2.1) выходных сигналов:

- а) Переключение цилиндра А из нижнего состояния в верхнее в момент времени 2;
- б) Переключение цилиндра А из верхнего состояния в нижнее и цилиндра В из нижнего состояния в верхнее в момент времени 5;
- в) Переключение цилиндра В из верхнего состояния в нижнее в момент времени 8.

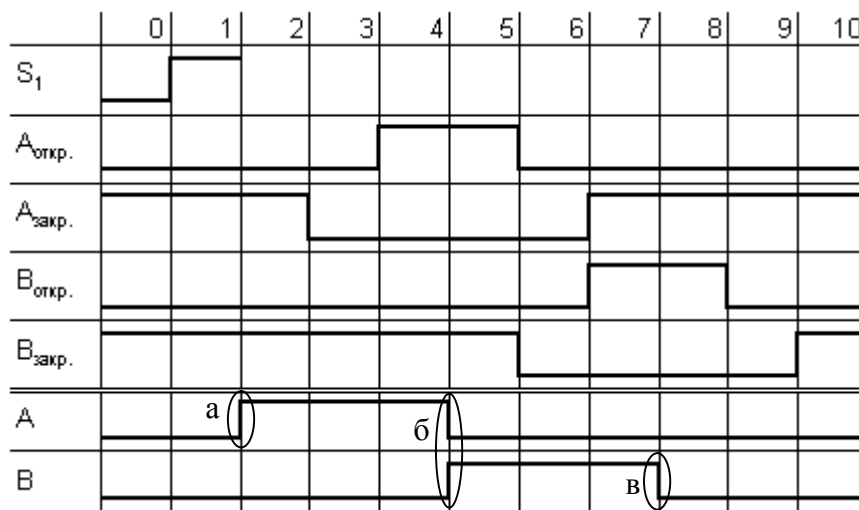


Рис. 2.2.1. Моменты переключения выходных сигналов

2. Далее с использованием циклограммы (и описания работы оборудования, если таковое имеется) определяется (рис 2.2.3.Б), вследствие срабатывания каких датчиков (рис 2.2.2) были выполнены данные переключения:

- а) Начало выдвижения цилиндра А является следствием нажатия кнопки S_1 ;
- б) Цилиндр А начинает задвигаться, а цилиндр В начинает выдвигаться, после полного открытия цилиндра А, т.е. после срабатывает датчик $A_{откр.}$;
- в) Цилиндр В начинает задвигаться, только после его полного выдвижения, т.е. после срабатывает датчик $B_{откр.}$.

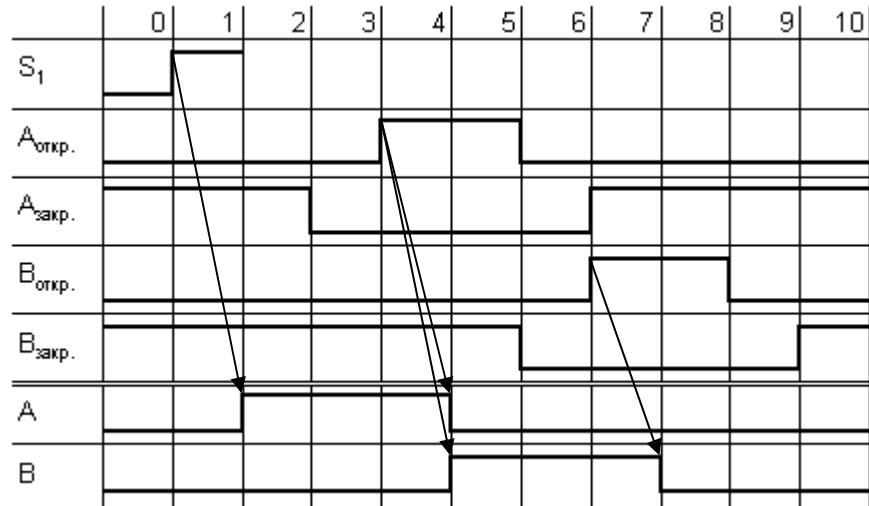


Рис. 2.2.2. Причины переключения выходных сигналов

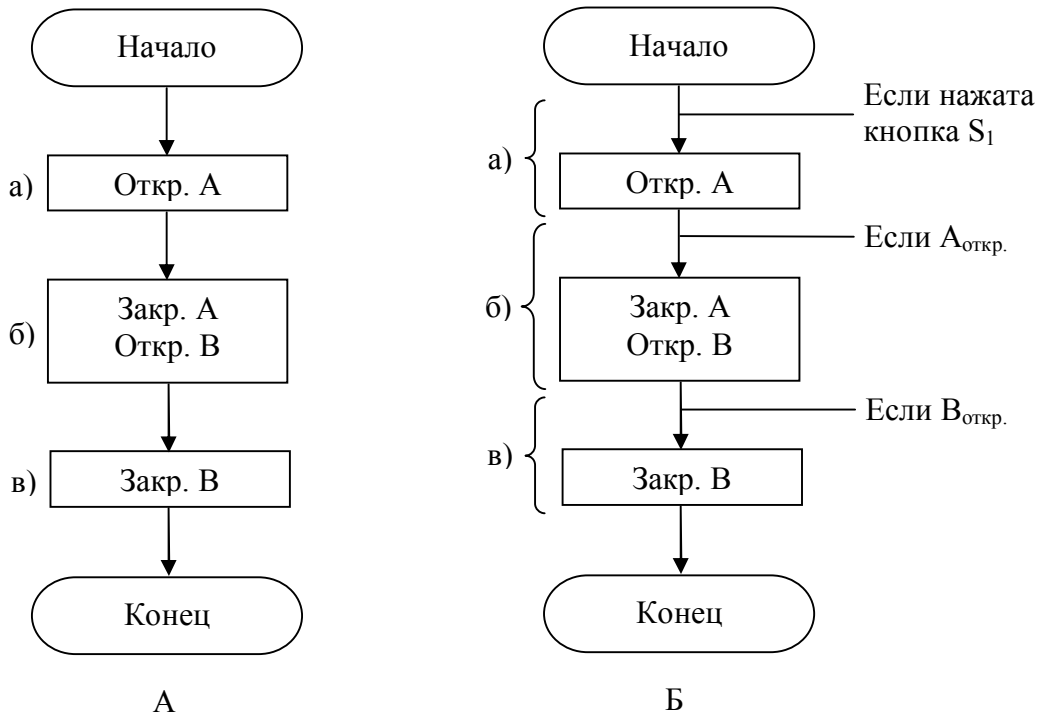


Рис. 2.2.3. Последовательность действий:

- А) без условий перехода к следующему действию;
- Б) с условиями перехода к следующему действию

Последовательность действий (рис 2.2.3) не является полным и достаточным алгоритмом для написания программы, так как изображена линейной и не содержит ветвлений. Входящие в последовательность действий альтернативы содержат лишь ветвь «Да». При невыполнении условия алгоритм повторяет проверку этого условия до тех пор, пока оно не наступит, т.е. данные условия являются лишь ожиданиями наступления определенных событий.

Последовательности действий могут строиться не только для всего процесса в целом, но и для отдельных его частей.

Несмотря на все сказанное, последовательность действий уже на данном этапе (без составления подробной блок-схемы) позволяет судить о том, какие из датчиков и исполнительных механизмов, возможно, не будут участвовать в системе регулирования. Так, датчики закрытого состояния ($A_{\text{закр.}}$ и $B_{\text{закр.}}$) в рассмотренном примере можно исключить из списка закупаемого оборудования и схемы подключения, так как они никак не отображены в последовательности действий (рис 2.2.3). Кроме того, нет необходимости резервировать дискретные входы контроллера для подключения указанных датчиков.

Подобная последовательность демонстрирует лишь наиболее вероятный ход событий, например, в отсутствие аварии, но она в дальнейшем упрощает составление блок-схемы.

ВЫВОД: построение последовательности действий начинается с конца циклограммы, т.е. с выходных параметров.

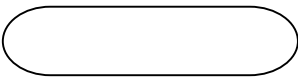
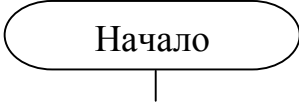
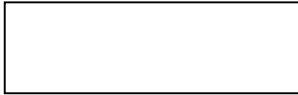
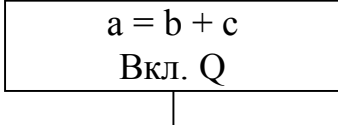
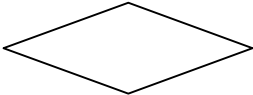
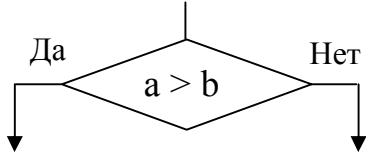

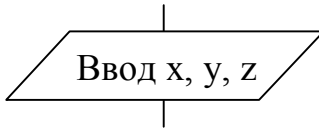

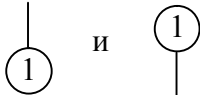
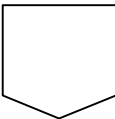
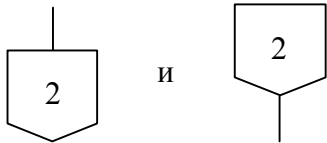
2.3. Блок-схема

Блок-схема — графическое представление алгоритма или процесса, изображающее шаги в виде блоков различной формы, соединенных между собой стрелками [4,5,6].

Наиболее часто используемые блоки [7] приведены в табл. 2.3.1, пример участка блок-схемы изображен на рис. 2.3.1.

Таблица 2.3.1

Основные обозначения по ГОСТ 19.003-80

Блок	Значение	Пример
	Вход и выход из внешней среды («Начало» и «Конец»)	
	Выполнение операции или нескольких операций. Операции записываются непосредственно внутри прямоугольника. В одном прямоугольнике может быть записано несколько операций, выполняемых одна за другой	
	Решение (условие). Альтернатива с двумя («Да» и «Нет») или более выводами	
	Ввод/вывод данных	
	Соединитель. Используется для организации разрывов	
	Разрыв между страницами	

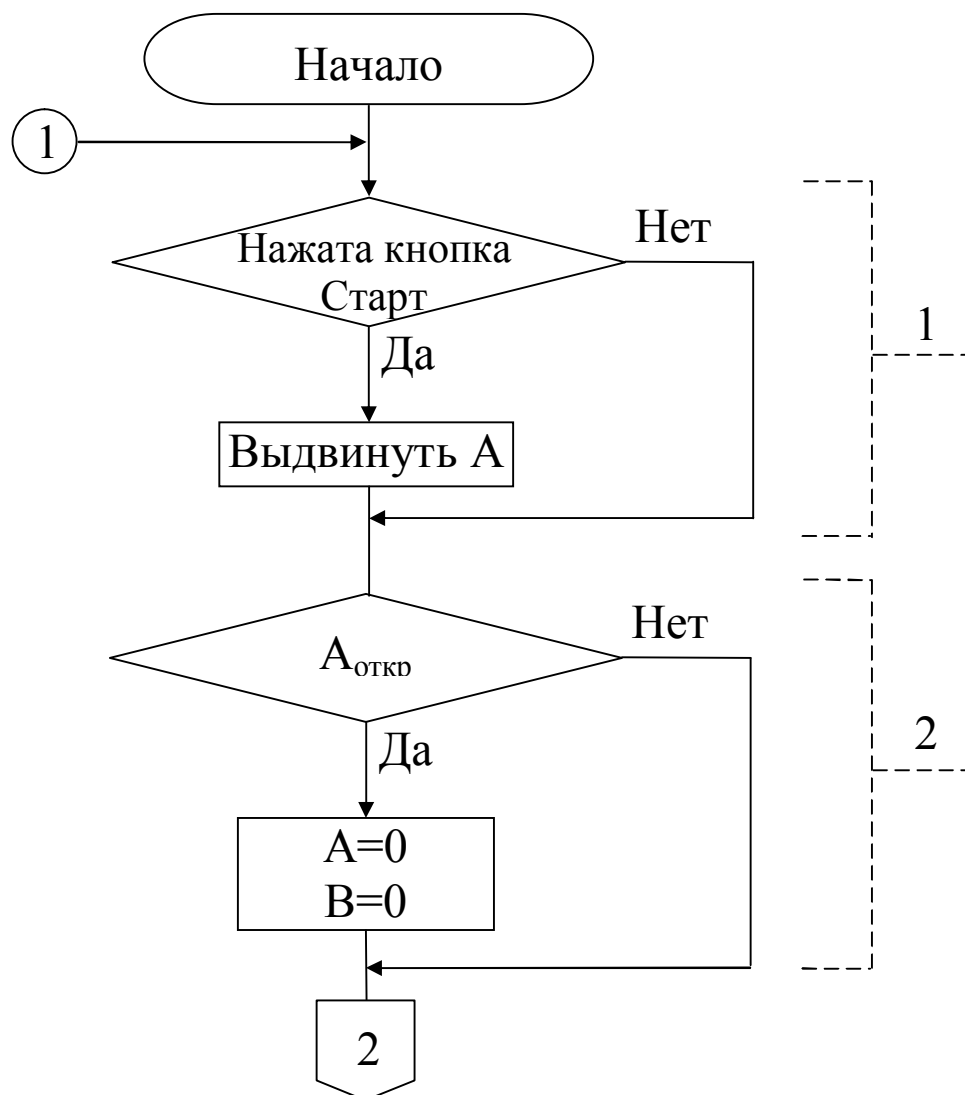


Рис. 2.3.1 Пример участка блок схемы

2.4. Написание программного кода по имеющейся блок-схеме

Отдельные участки блок-схемы однозначно соответствуют программному коду и наоборот. Ромб читается как «Если», ветвь «Да» как «то», а ветвь «Нет» как «иначе».

Рассмотрим несколько стандартных участков блок-схемы:

Альтернатива с двумя активными ветвями

На рис. 2.4.1 изображен участок блок-схемы и соответствующий ей программный код для альтернативы с двумя активными ветвями.

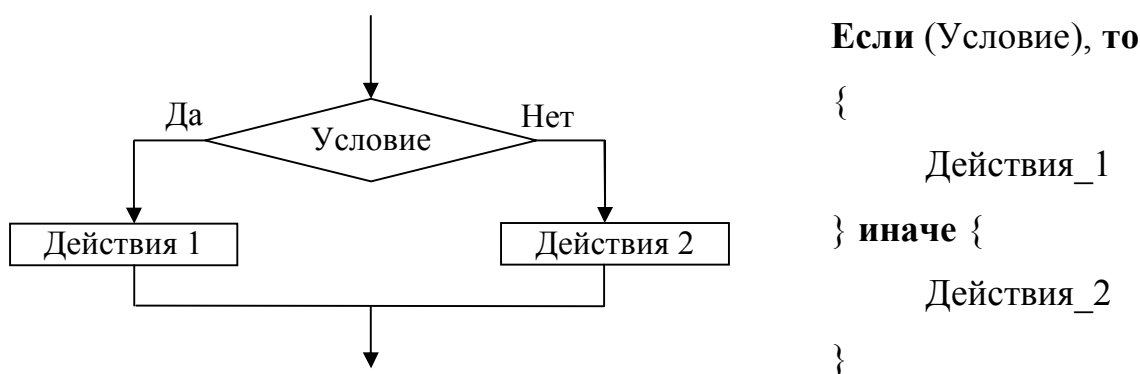


Рис. 2.4.1. Альтернатива с двумя активными ветвями

Альтернатива с одной активной ветвью

Альтернатива может содержать пустой набор действий в одной из ветвей. Альтернатива, содержащая пустой набор действий в ветви «Нет» (рис. 2.4.2), является наиболее распространенным частным случаем альтернативы с двумя активными ветвями. В этом случае нет необходимости в блоке «иначе».

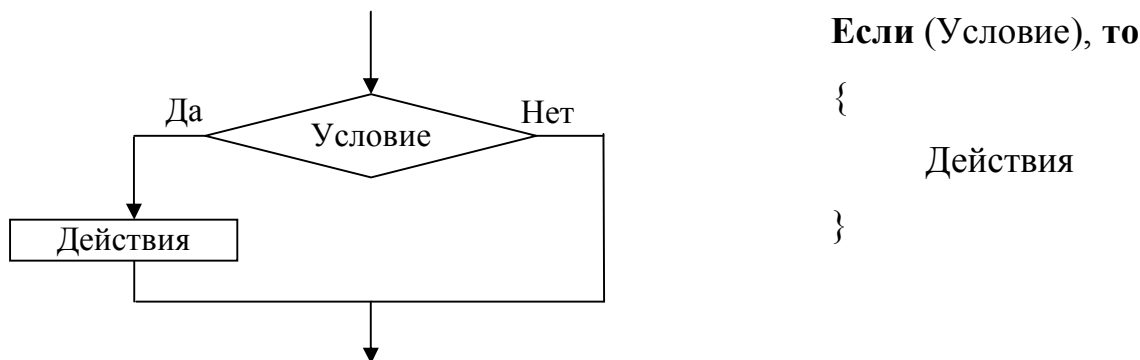


Рис. 2.4.2. Альтернатива с одной активной ветвью «Да»

Если действие отсутствует в ветви «Да» (рис. 2.4.3), то приведенная форма записи программного кода некорректна, так как содержит пустой блок действий («Ничего не делать»), а блок «то» является обязательным и сокращению не подлежит.

Правильно в данном случае поменять местами действия ветвей «Да» и «Нет» (рис. 2.4.4), используя отрицание «не» в условии. Черта над условием означает отрицание «не».

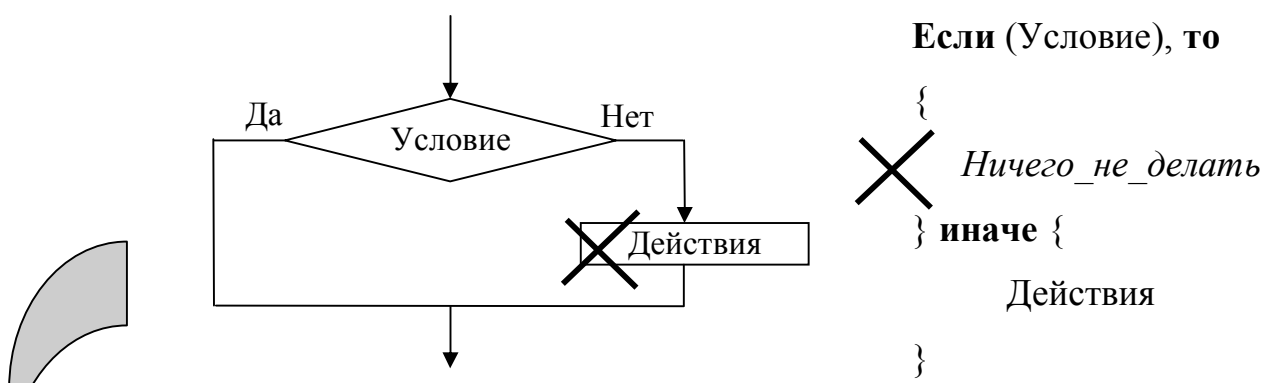


Рис. 2.4.3. Альтернатива с одной активной ветвью «Нет»

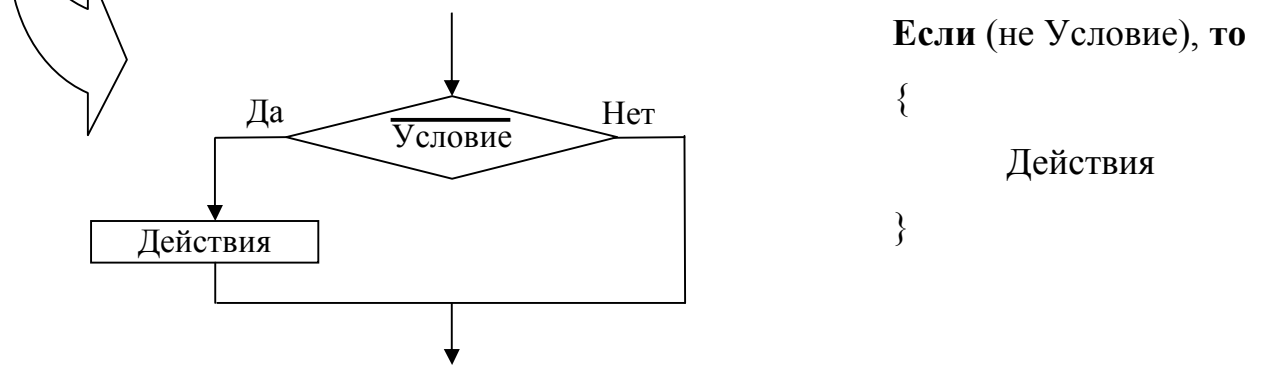


Рис. 2.4.4. Перестановка местами ветвей «Да» и «Нет»

Пример 2

Заготовки двигаются по транспортной ленте (рис. 2.4.5), которая запускается нажатием кнопки «**Пуск**». По достижении заготовкой датчика обнаружения **D** в конце транспортера, лента останавливается, и деталь сталкивается толкателем **A**. Следующая деталь подается после повторного нажатия кнопки «**Пуск**».

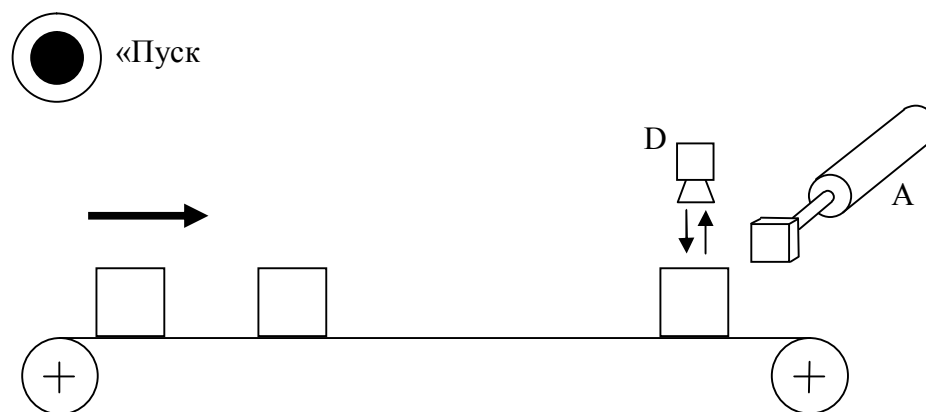


Рис. 2.4.5. Установка, обнаруживающая заготовки в конце транспортной ленты

Составим последовательность действий работы данной установки (рис. 2.4.6).

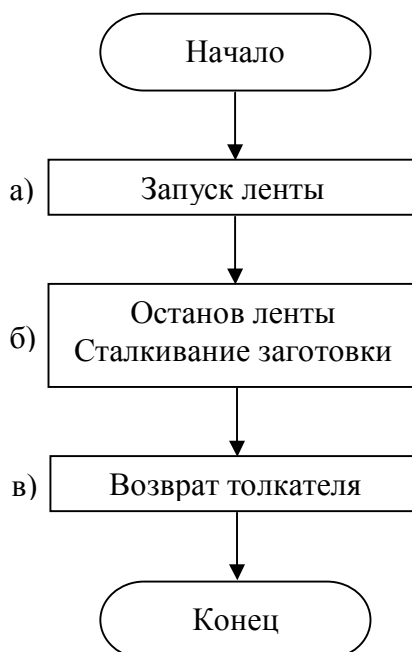


Рис. 2.4.6. Последовательность действий работы установки

Дополним последовательность действий (рис. 2.4.6) условиями переходов к следующему действию (рис. 2.4.7). Как видно из полученных условий, для работы установки необходим датчик открытого конечного положения ($A_{откр.}$) цилиндра А.

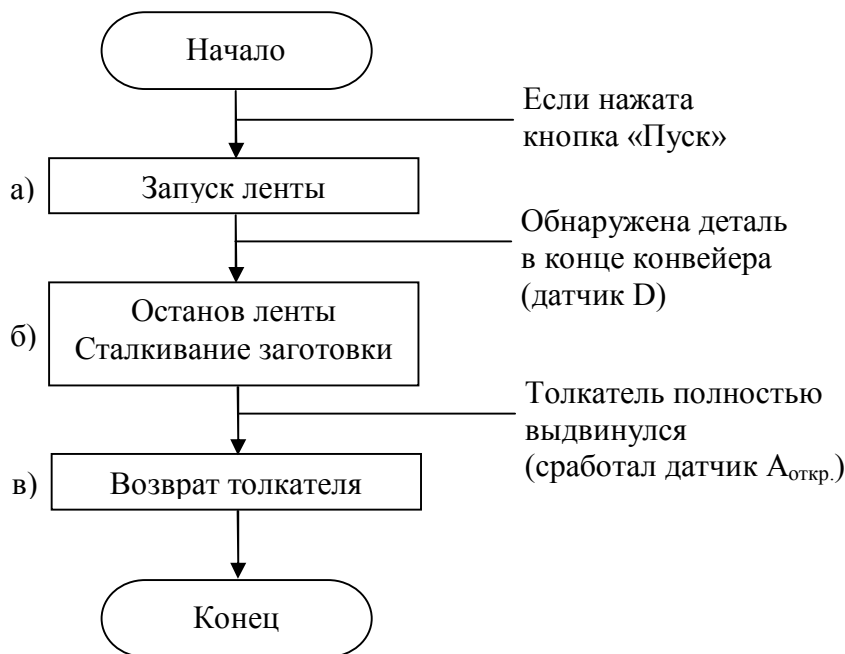


Рис. 2.4.7. Последовательность действий, дополненная условиями перехода

Изобразим циклограмму (рис. 2.4.8) рассматриваемой установки. Данная циклограмма полностью подтверждает полученную ранее последовательность действий (рис. 2.4.7), и содержит (рис. 2.4.9) те же действия (а, б, в) и условия переключений.

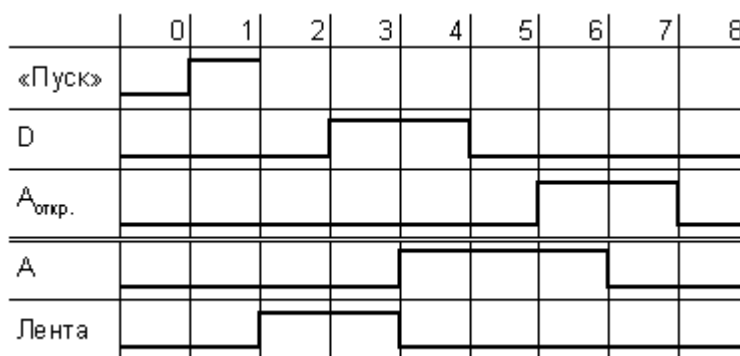


Рис. 2.4.8. Циклограмма работы установки

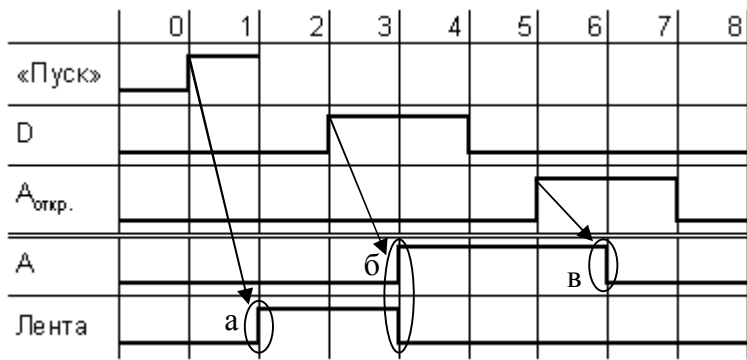


Рис. 2.4.9. Циклограмма с указанием моментов переключений

Составим блок-схему (**рис. 2.4.10**) рассматриваемой установки.

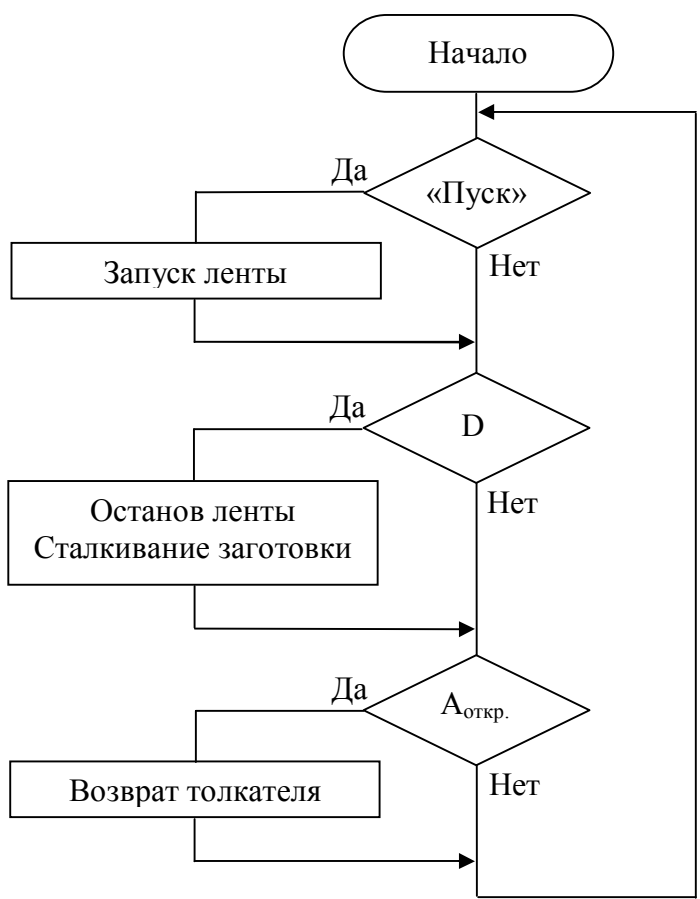
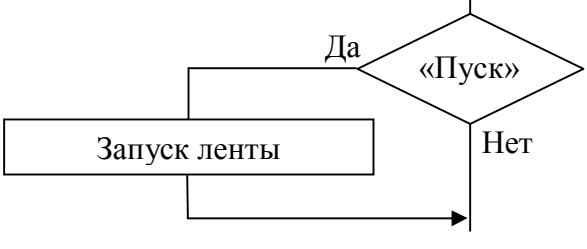
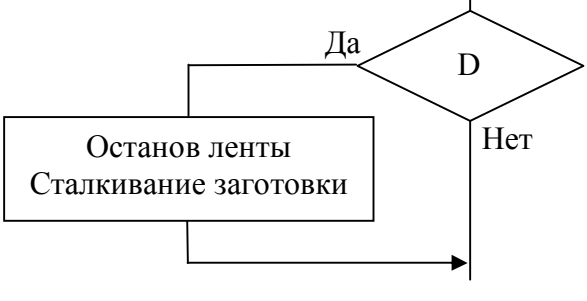
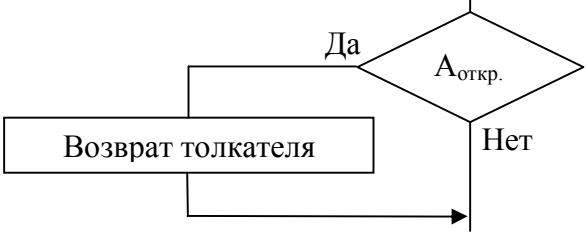


Рис. 2.4.10. Блок-схема работы установки

Данной блок-схеме однозначно соответствует программный код, который получается путем ее разбиения (**табл. 2.4.1**) на типовые участки. Каждому из трех полученных типовых участков с одной активной ветвью «Да» (**рис. 2.4.2**), однозначно соответствует приведенный программный код.

Разбиение блок-схемы на типовые участки

а)		Если («Пуск»), то { Запуск ленты }
б)		Если (D), то { Останов ленты Сталкивание заготовки }
в)		Если (A_{откр.}), то { Возврат толкателя }

Итоговый программный код получается после объединения этих трех участков:

Если («Пуск»), то

```
{
    Запуск ленты
}
```

Если (D), то

```
{
    Останов ленты
    Сталкивание заготовки
}
```

Если (A_{откр.}), то

```
{
    Возврат толкателя
}
```

2.5. Вложения

В качестве действий в блок-схемах может использоваться одна или несколько команд. Кроме того, в действия могут быть вложены (рис. 2.5.1) условия и подпрограммы, которые в свою очередь также могут содержать условия и подпрограммы.

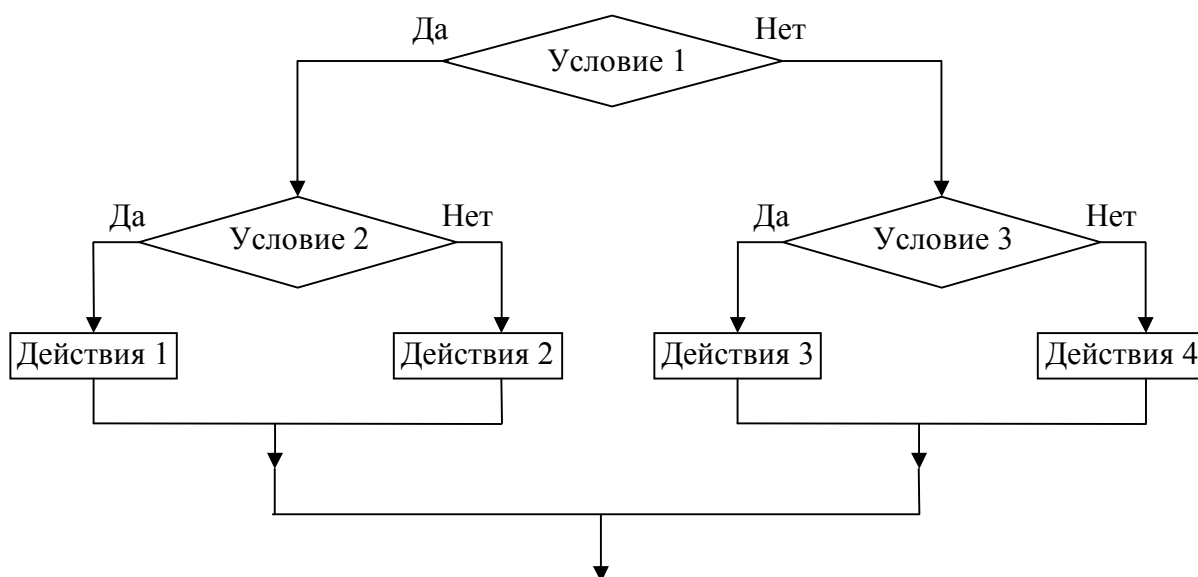


Рис. 2.5.1. Сложная структура программы с вложенными условиями

В этом случае необходимо разбить блок-схему на типовые участки, программный код для которых известен (рис. 2.5.2).

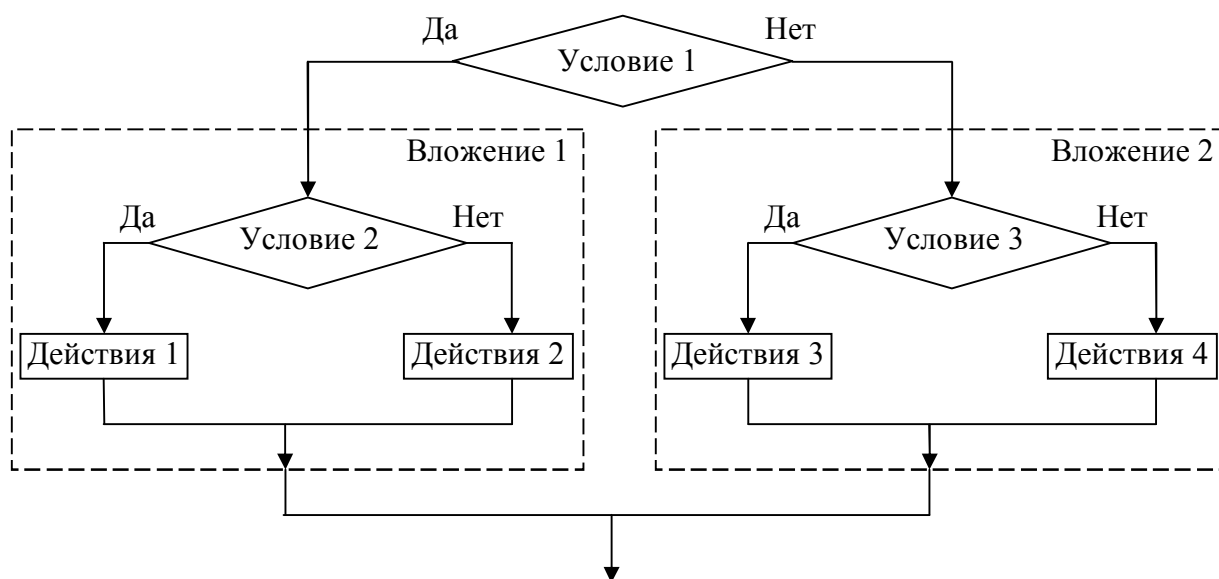
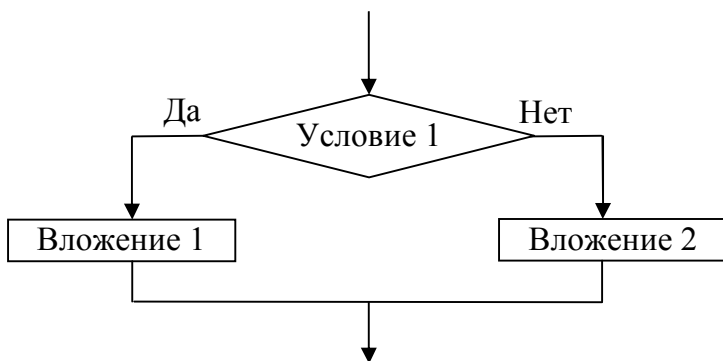


Рис. 2.5.2. Разбиение программы на типовые участки

В данном случае блок-схема основной программы будет иметь вид типового участка с двумя активными ветвями (рис. 2.5.3). Вложения также имеют вид типовых участков с двумя активными ветвями (рис. 2.5.4 и 2.5.5).

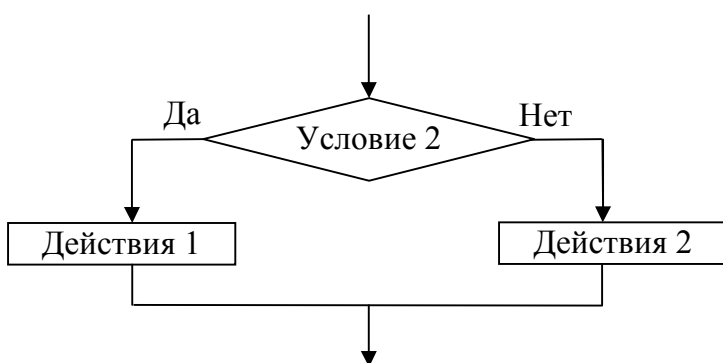


```

Если (Условие_1), то
{
    Вложение_1
} иначе {
    Вложение_2
}

```

Рис. 2.5.3. Основная программа с вложениями

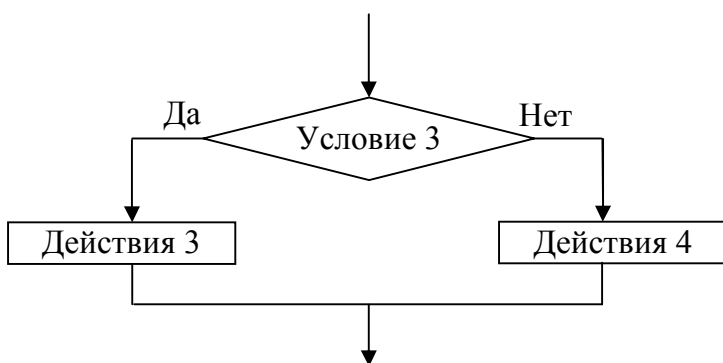


```

Если (Условие_2), то
{
    Действия_1
} иначе {
    Действия_2
}

```

Рис. 2.5.4. Вложение 1



```

Если (Условие_3), то
{
    Действия_3
} иначе {
    Действия_4
}

```

Рис. 2.5.5. Вложение 2

Подставляя вложения 1 и 2 (рис. 2.5.4 и 2.5.5) в основную программу (рис. 2.5.3), получаем итоговый код для исходной (рис. 2.5.1) структуры:

```
Если (Условие_1), то
{
    Если (Условие_2), то {
        Действия_1
    } иначе {
        Действия_2
    }
} иначе {
    Если (Условие_3), то {
        Действия_3
    } иначе {
        Действия_4
    }
}
```

Преобразование алгоритмов с вложениями

Блок-схема с вложениями может быть преобразована в блок-схему без вложений. Например, **рис. 2.5.1** может быть преобразован (**рис. 2.5.6**) к четырем последовательно соединенным участкам, не содержащим вложенных условий. При этом количество действий остается прежним, а условия объединяются в зависимости от маршрута, по которому можно попасть на соответствующие действия. Черта над условиями означает отрицание «Не».

Рассматриваемое преобразование имеет некоторые ограничения, и в случаях, когда в условиях и действиях имеется хотя бы одна общая переменная – может быть неэквивалентным.

В случае если во всех условиях присутствуют только входные переменные, а во всех действиях только выходные, то подобное преобразование может быть выполнено всегда.

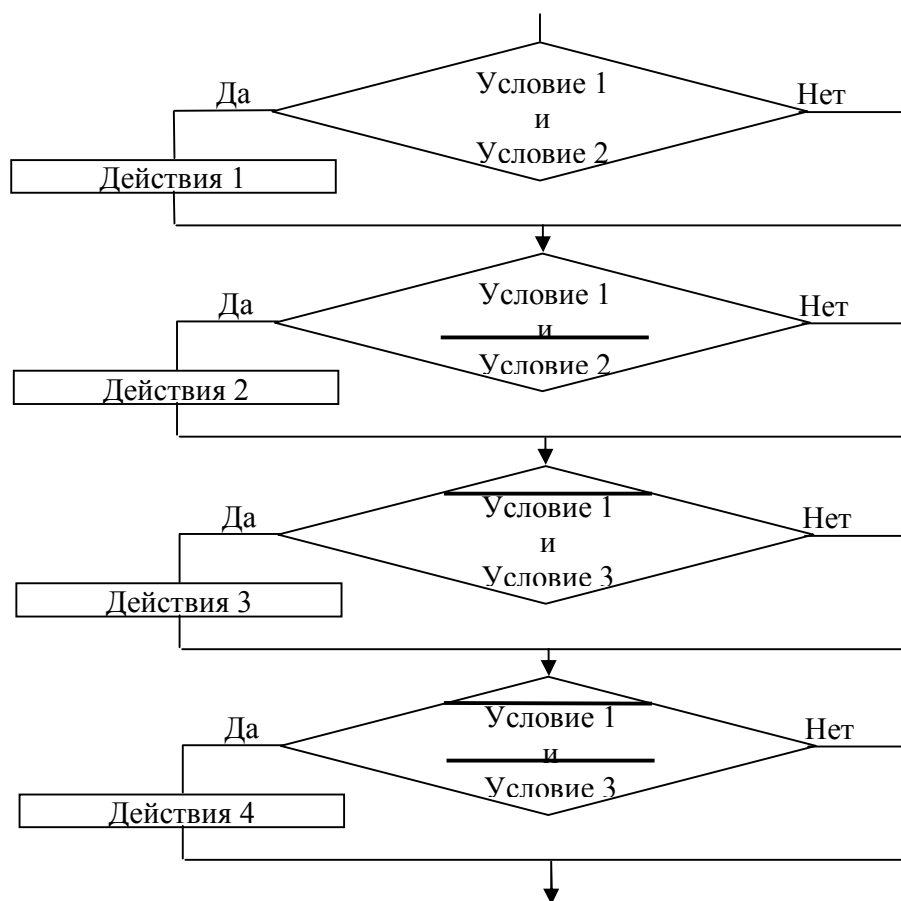


Рис. 2.5.6. Преобразование вложений

Программный код для данного преобразования имеет вид:

```

Если (Условие_1 & Условие_2), то
{
    Действия_1
}
Если (Условие_1 & (не Условие_2)), то
{
    Действия_2
}
Если ((не Условие_1) & Условие_3), то
{
    Действия_3
}
Если ((не Условие_1) & (не Условие_3)), то
{
    Действия_4
}
    
```

2.6. Подпрограммы

Подпрограммы (рис. 2.6.1) аналогичны вложениям, за исключением того, что подстановка подпрограмм (рис. 2.6.2 и 2.6.3) в главную программу не происходит. Каждая подпрограмма сохраняется в своем отдельном файле, имеющем начало и конец.

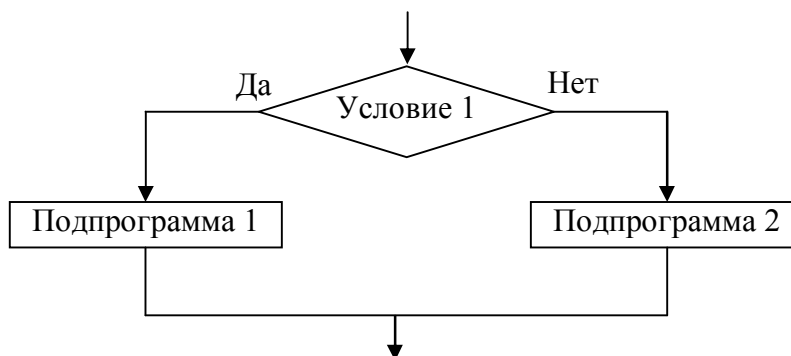


Рис. 2.6.1. Основная программа с подпрограммами

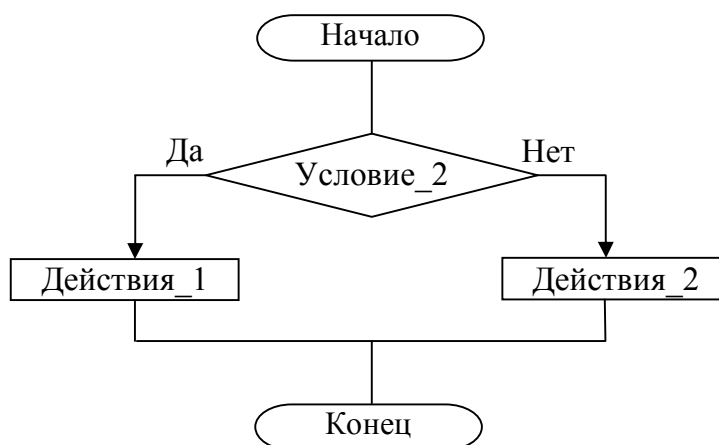


Рис. 2.6.2. Подпрограмма 1

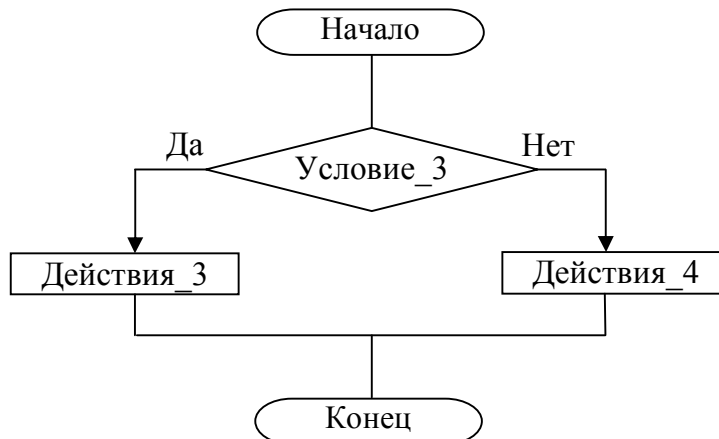


Рис. 2.6.3. Подпрограмма 2

Структура программы и подпрограммы идентична, так как подпрограмма фактически также является программой (аналогично персональному компьютеру, где любая расположенная в папке подпапка сама по себе также является папкой).

2.7. Шаги

Зачастую имеется необходимость в выполнении последовательно сменяемых локальных циклов (этапов или шагов). Каждый из таких локальных циклов выполняется независимо от других, а переход на следующий шаг происходит после выполнения определенного условия (рис. 2.7.1).

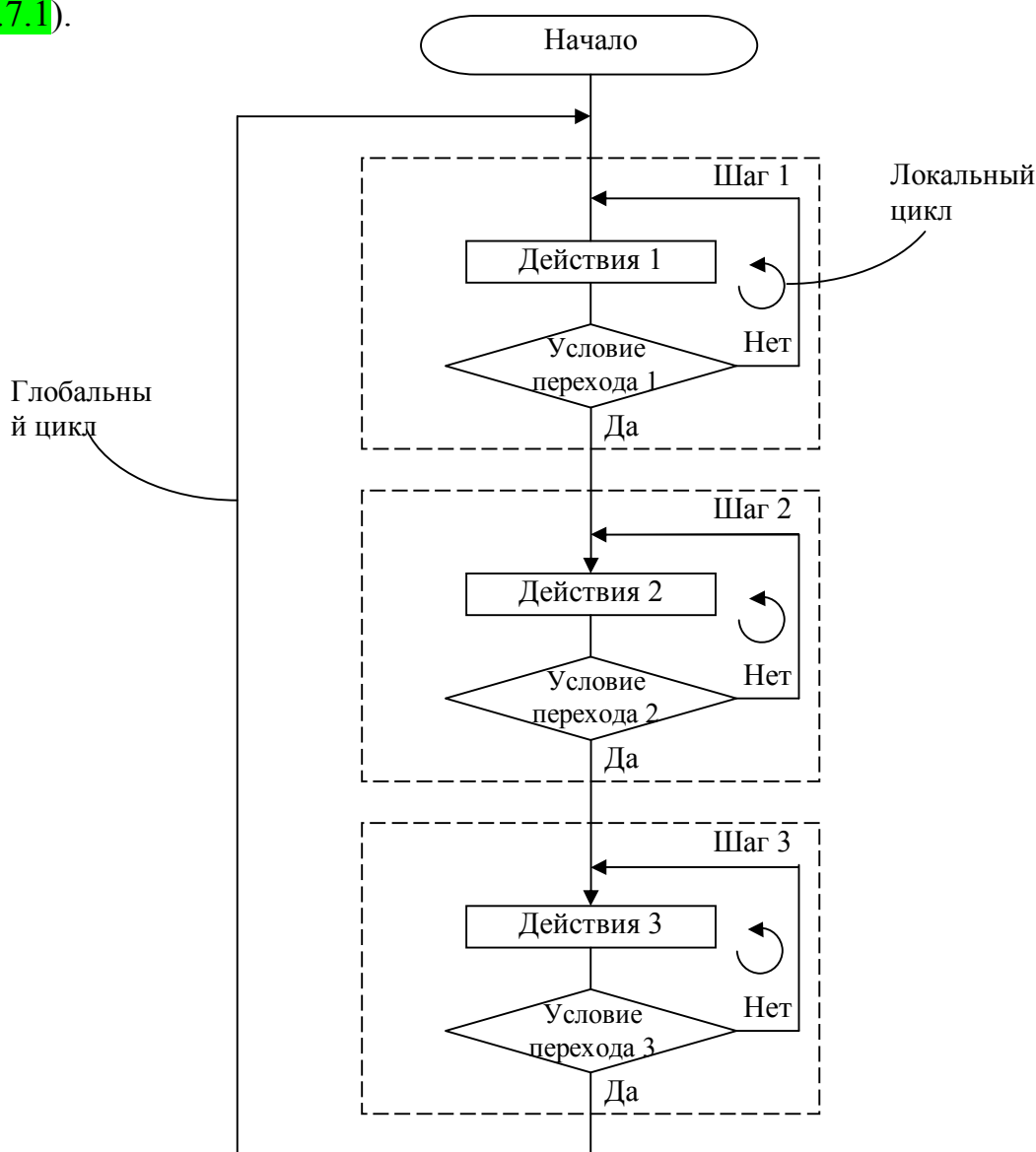


Рис. 2.7.1. Локальные циклы

Каждое из приведенных действий («Действия 1», «Действия 2», «Действия 3») может также включать вложенные условия и подпрограммы.

Шаги на языке FST

Программирование контроллера фирмы Festo производится на языке FST. Реализация шагов на языке FST осуществляется при помощи оператора **STEP**, содержащего имя или номер этого шага:

STEP имя|№

Например:

STEP 1

или

STEP End

При этом в качестве условия перехода выступает последнее из условий этого шага, например:

STEP 1

...

IF I0.0

THEN SET O1.1

STEP 2

...

В данном примере переход ко второму шагу осуществляется только после появления на дискретном входе **I0.0** сигнала.

Выполнение шагов осуществляется последовательно. Для перехода к произвольному шагу используется оператор **JMP TO** с указанием имени или номера шага, к которому осуществляется переход:

JMP TO 1

или

JMP TO End

Пример 3

Задача. Работа станка логически разделена на два этапа: подача заготовки и ее снятие. При появлении заготовки в начале конвейерной ленты срабатывает датчик **D_Start** и лента запускается. Когда заготовка доходит до конца конвейерной ленты, срабатывает датчик **D_End** и начинается сверление детали до заданной глубины. Глубина определяется датчиком **D_SverloDown**. После его срабатывания начинается следующий этап - сверло поднимается (определяется датчиком **D_SverloUp**) и отключается, загорается индикатор **HL**. Деталь снимается вручную, индикатор **HL** гаснет, и станок ожидает новой заготовки в начале конвейерной ленты.

Составим последовательность действий работы всего станка (рис. 2.7.2).

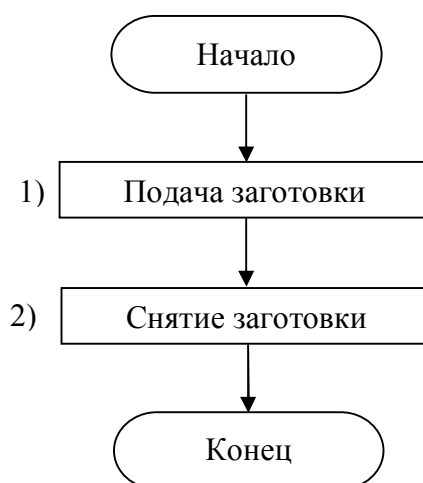


Рис. 2.7.2. Последовательность действий работы станка

Для каждого из этапов также можно составить собственные последовательности действий (рис. 2.7.3 и 2.7.4), при этом условием перехода на следующий этап является последнее из условий текущего этапа.

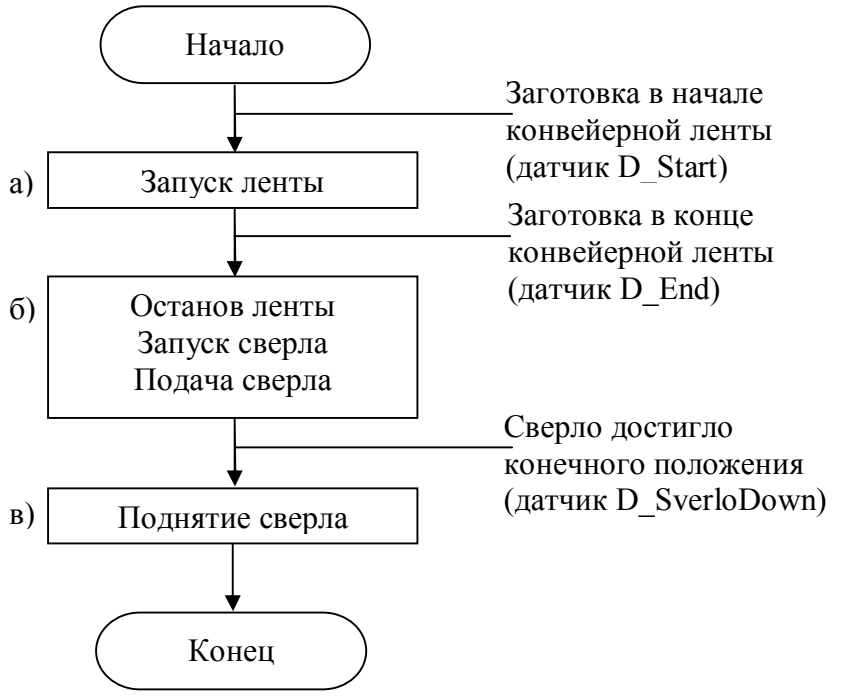


Рис. 2.7.3. Последовательность действий при подаче заготовки

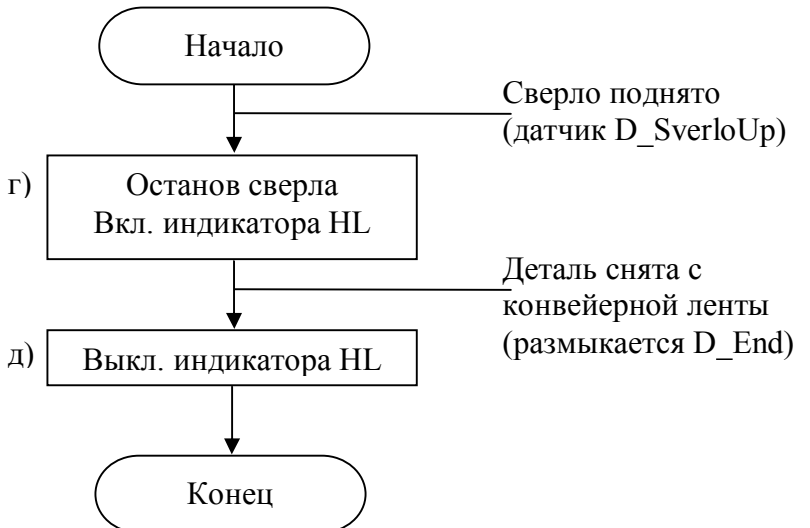


Рис. 2.7.4. Последовательность действий при снятии заготовки

Составим циклограмму работы станка (рис. 2.7.5), которая также логически разделена на два этапа. Этапы подписываются сверху.



Рис. 2.7.5. Циклограмма работы станка

Аналогично блок-схема (рис. 2.7.6) работы станка разделена на два шага. Каждый из шагов выделен пунктирной рамкой.

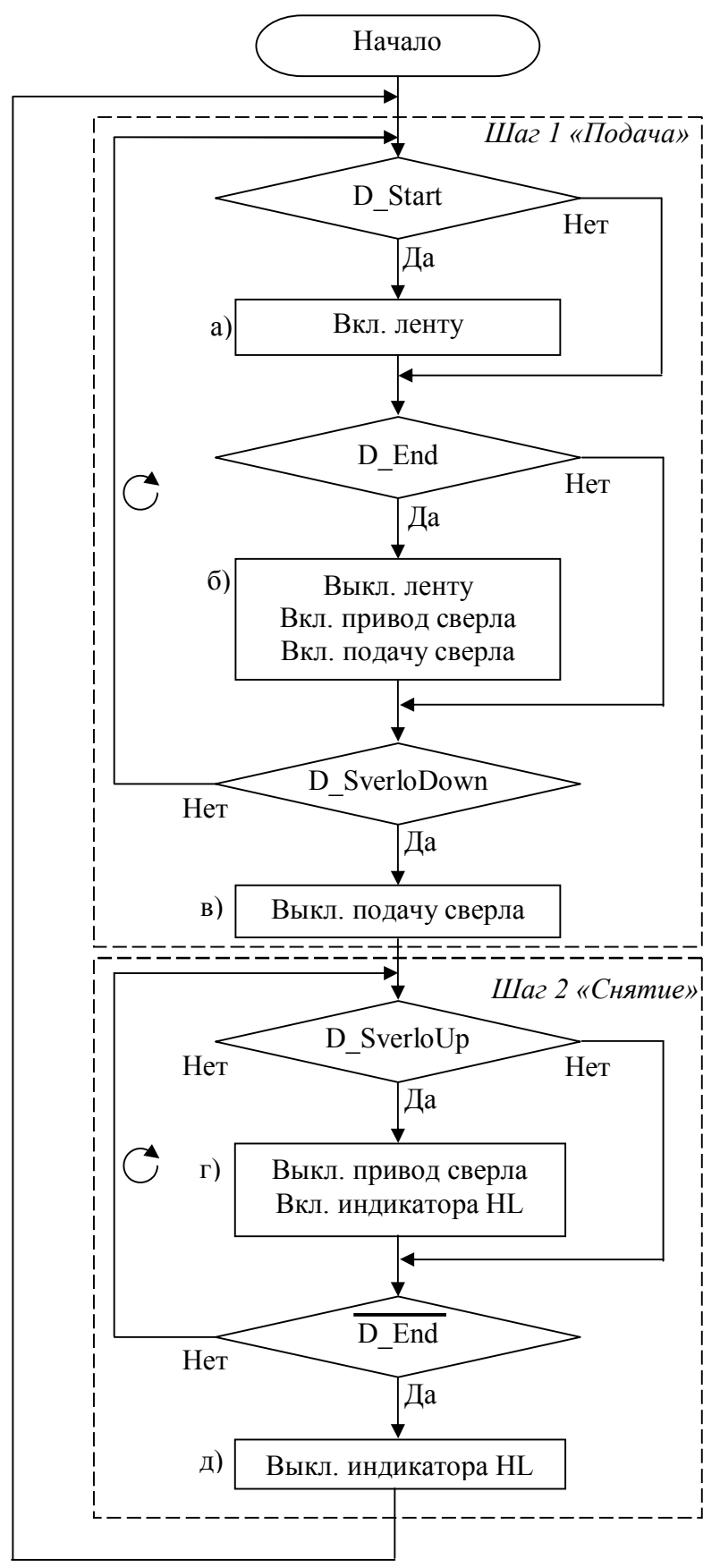


Рис. 2.7.6. Блок-схема работы станка (по шагам)

Остается только написать программный код для рассмотренной (рис. 2.7.6) блок-схемы:

STEP 1

```
IF      D_Start
THEN SET   Lenta

If      D_End
THEN RESET Lenta
        SET   Sverlo
        SET   PodachaSverla

IF      D_SverloDown
THEN RESET PodachaSverla
```

STEP 2

```
IF      D_SverloUP
THEN RESET Sverlo
        SET   HL

IF      N      D_End
THEN RESET HL

        JMP TO 1
```

3. КОНТРОЛЛЕР

3.1. Входные и выходные переменные

Для любого логического контроллера характерно наличие дискретных входов и выходов. На схеме (рис. 3.1.1) входы изображаются слева, а выходы справа. Как правило, число, как входов, так и выходов кратно восьми (8, 16, 32, 64 и т.д.). Каждый вход и выход имеет уникальный адрес. Адрес может состоять из нескольких частей, например двух частей разделенных точкой. В данном случае первая цифра означает номер банка данных (байта или слова) или порта, к которому осуществляется подключение. Вторая цифра (от 0 до 7 для байта или от 0 до 15 для слова) – порядковый номер в данном банке или порте.

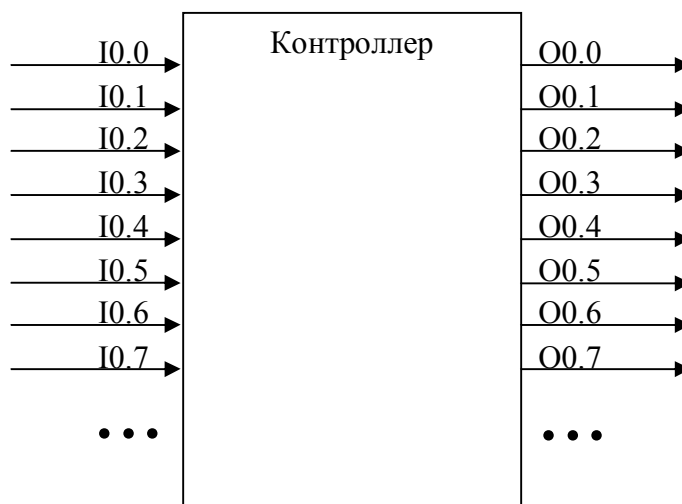


Рис. 3.1.1. Дискретные входы и выходы контроллера

Получить значение входной переменной можно указав ее адрес с приставкой «I» («Input») или, в зависимости от марки контроллера, «DI» («Digital Input»), например:

I0.0

Получить значение выходной переменной можно указав ее адрес с приставкой «O» («Output») или, в зависимости от марки контроллера, «DO» («Digital Output»), реже «Q», например:

O1.7

3.2. Ввод/Вывод контроллера

Ввод/Вывод в контроллере реализован на аппаратном уровне (рис. 3.2.1). Обновление значений входных и выходных переменных происходит через заданные интервалы времени (например, 1 секунда или 1/100 секунды). На блок-схеме [6,7] ввод/вывод обозначается в виде «параллелограмма» с наклонными боковыми сторонами. В пользовательской программе (блок-схеме) нет необходимости использовать данный блок (рис. 3.2.1), поскольку, как уже было сказано, ввод/вывод реализован в контроллере на аппаратном уровне и происходит без участия пользователя.

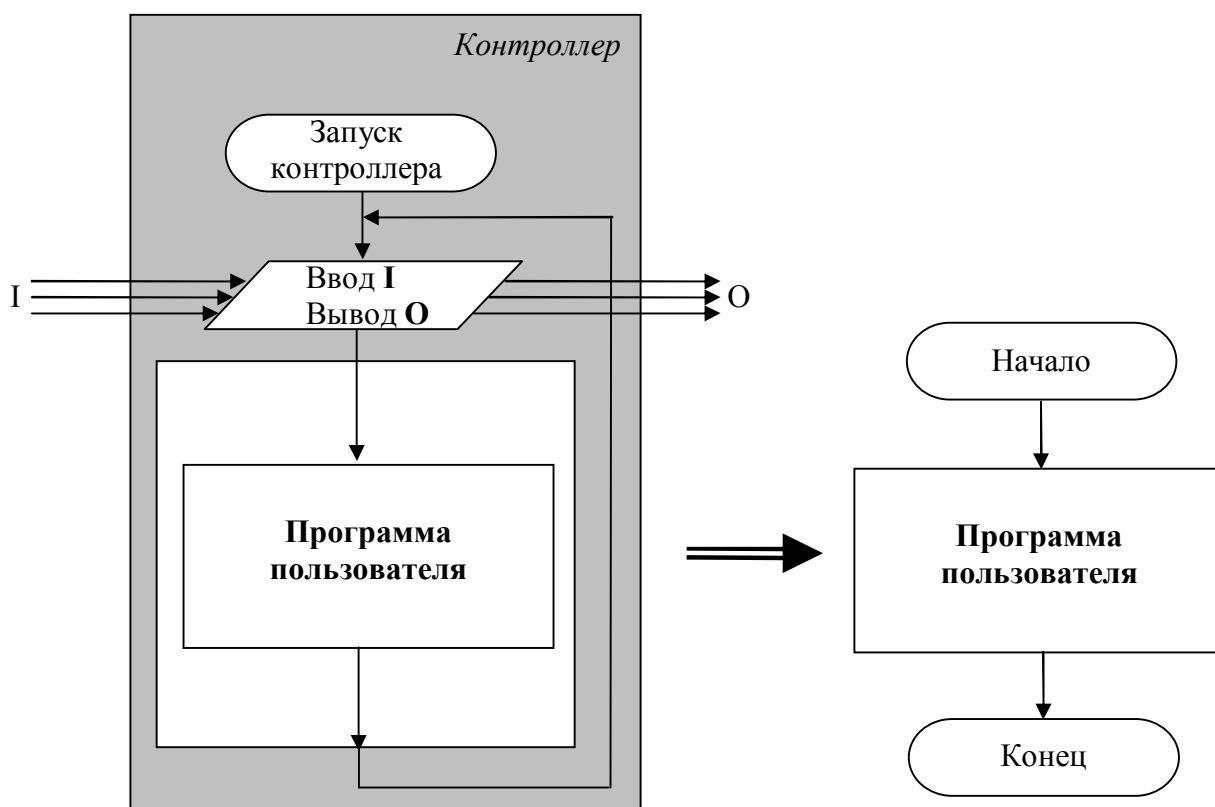


Рис. 3.2.1. Реализация процесса ввода/вывода контроллера

Входные переменные предназначены только для чтения и не могут быть записаны из контроллера. Значение входной переменной изменяется только в результате изменения электрического сигнала на соответствующем входе

контроллера. Входные переменные используются в основном в условиях («ромб» на блок-схеме [6,7]).

Выходные переменные, в зависимости от устройства контроллера, могут быть или только записаны, или записаны и прочитаны (в данном случае речь идет о чтении последнего записанного в них значения, а не о проверке изменения электрического сигнала на соответствующем выходе). Выходные переменные используются в основном для реализации действий («прямоугольник» на блок-схеме [6,7]). Как правило, значения выходных переменных (до первой записи в них значения) по умолчанию установлены в «ноль».

3.3. Элементы памяти

Обычно элементы памяти («ячейки памяти», «внутренние переменные» или просто «переменные») изображают (рис. 3.3.1) как выходы контроллера, соединенные с его входами, что означает сохранение значения между тактами выполнения программы (перенос значений на следующий шаг).

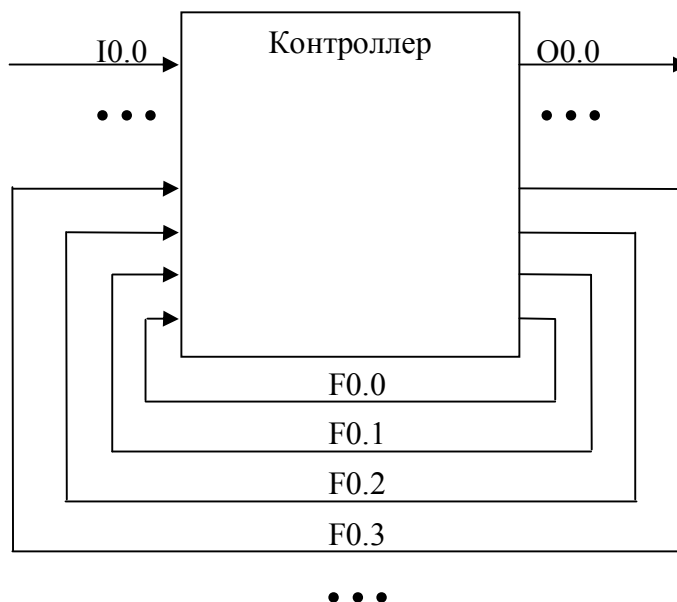


Рис. 3.3.1. Элементы памяти контроллера

Из этого следует, что элементы памяти обладают свойствами как входных, так и выходных переменных и предназначены для использования как в действиях («прямоугольник» на блок-схеме [6,7]), так и в условиях («ромб» на блок-схеме [6,7]). Вначале необходимо записать значение переменной в память, для того, чтобы его можно было прочитать в дальнейшем. Как правило, значения элементов памяти (до первой записи в них значения) по умолчанию установлены в «ноль», но чтение не присвоенных ячеек может вызвать «неопределенность» со случайным значением.

Элементы памяти, в зависимости от марки контроллера, могут иметь имена, начинающиеся с «X», «F» (Flag), «M» (Memory) и др., или даже задаваться пользователем.

На циклограмме (см. рис. 3.3.6) элементы памяти изображаются между входными и выходными переменными. На блок-схеме и в последовательности действий для каждого используемого элемента памяти должны присутствовать процедуры, как записи, так и чтения – **в противном случае использование данного элемента памяти теряет смысл!**

Существуют ситуации, когда по состоянию только входных сигналов (датчиков) нельзя однозначно судить о состоянии выходных переменных (ИМ). В этом случае используются элементы памяти. Разберем пример, наглядно иллюстрирующий их использование:

Пример 4

Установка (рис. 3.3.2) после нажатия кнопки «Start» подает подряд по две заготовки, если они имеются в наличии (определяется датчиком **D_Start**).

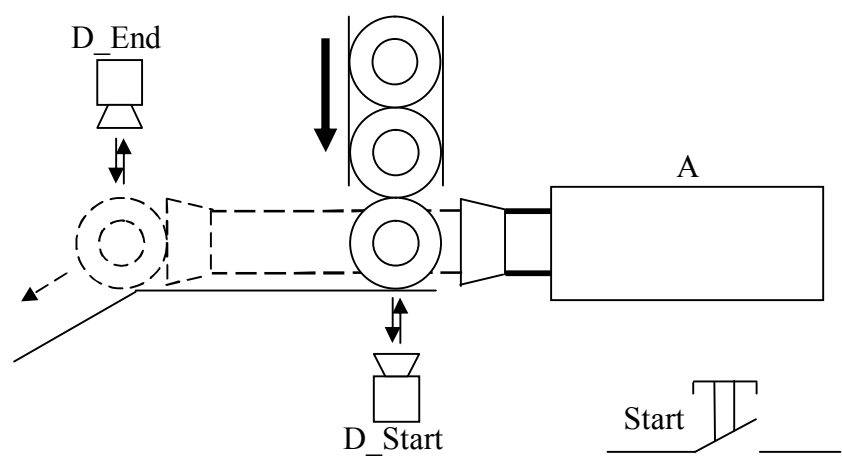


Рис. 3.3.2. Установка, подающая подряд по две заготовки:

- Start – кнопка запуска;
- A – цилиндр, подающий заготовки;
- D_Start – датчик наличия заготовки вначале;
- D_End – датчик конечного положения заготовки

Изобразим последовательность действий работы установки (**рис 3.3.3**). Она состоит из двух одинаковых комбинаций (выдвижение-задвижение цилиндра A), соединенных последовательно.

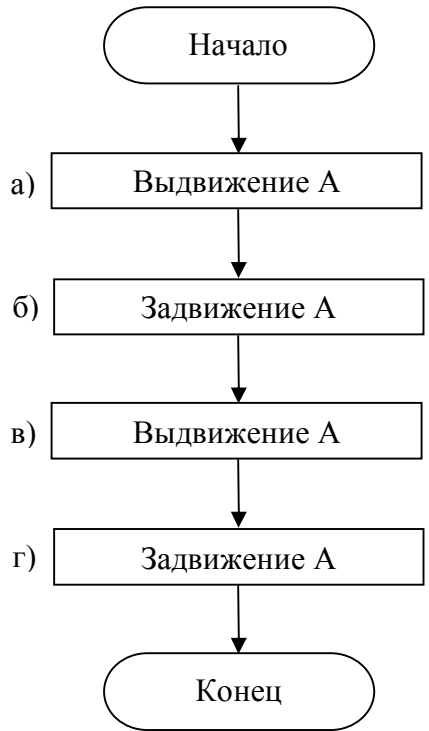
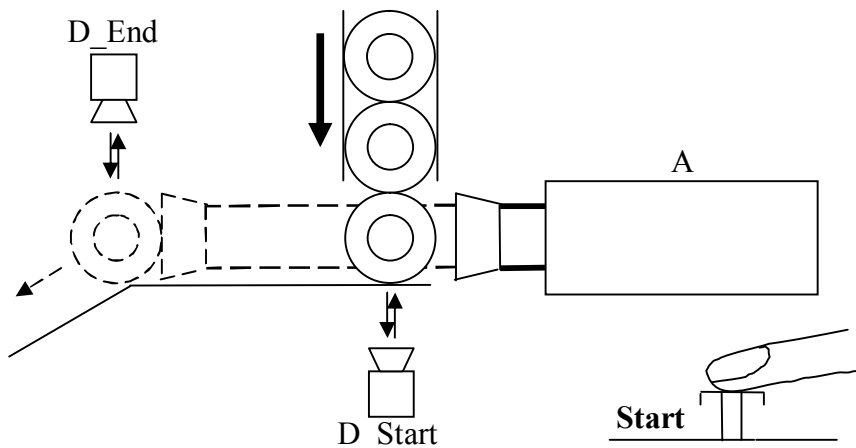


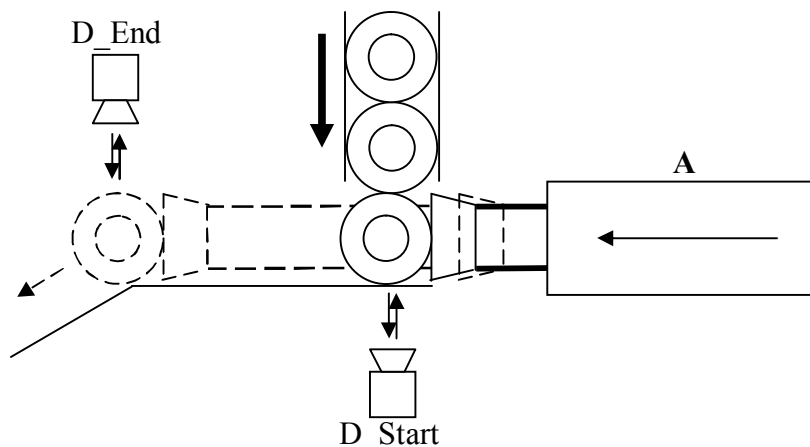
Рис. 3.3.3. Последовательность действий работы установки, подающей подряд по две заготовки

Составим (рис. 3.3.4) циклограмму работы установки, и изобразим каждый такт в виде отдельного (по кадрам) рисунка:

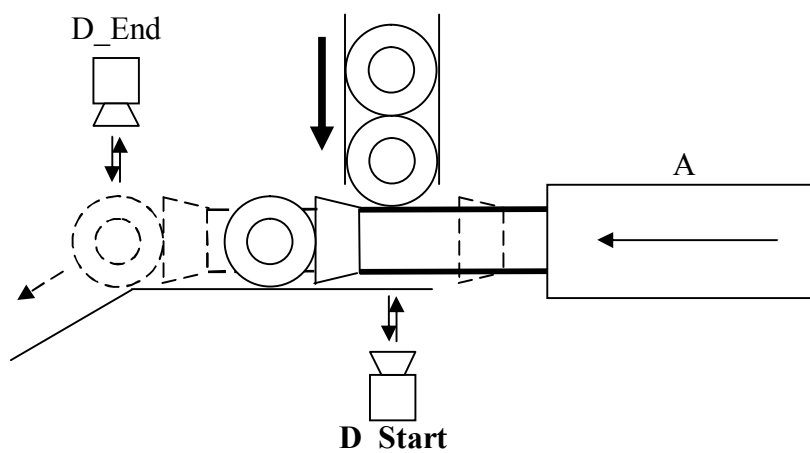
1. Нажатие кнопки запуска («Start»)



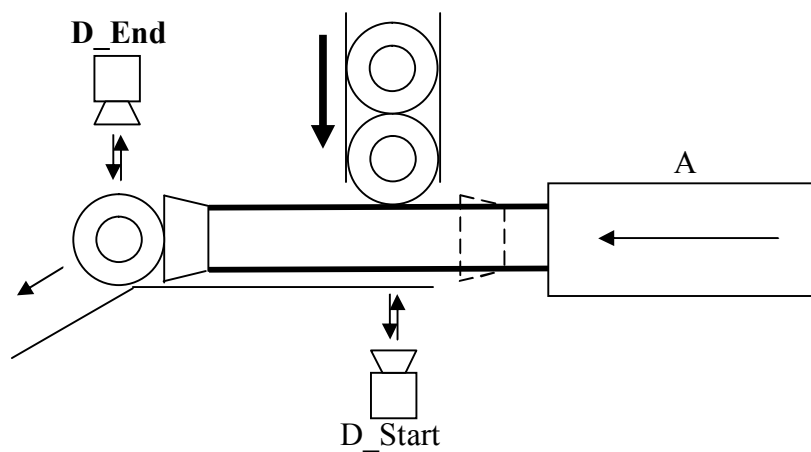
2. Начало движения цилиндра А



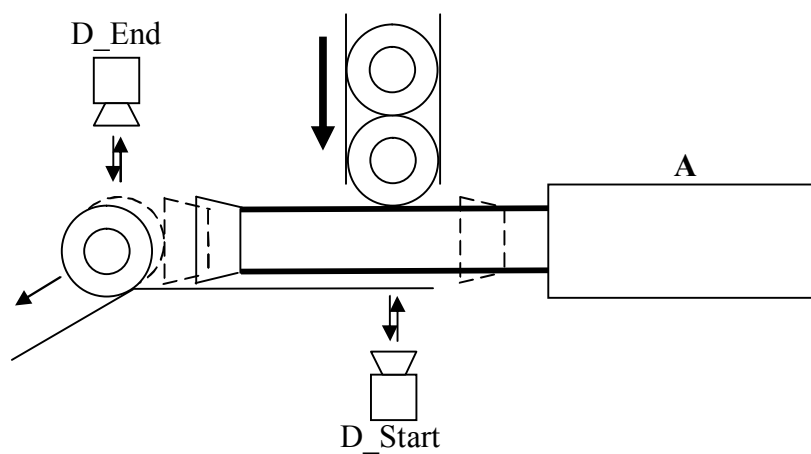
3. Перемещение заготовки (*D_Start размыкается*)



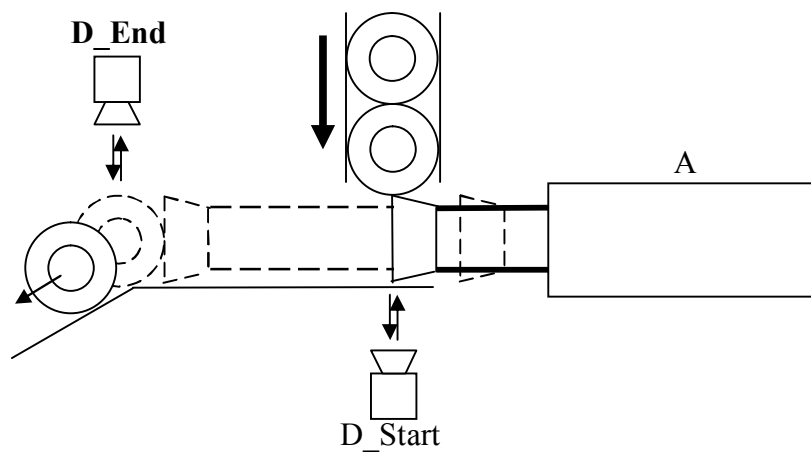
4. Заготовка достигла конца (D_End замыкается)



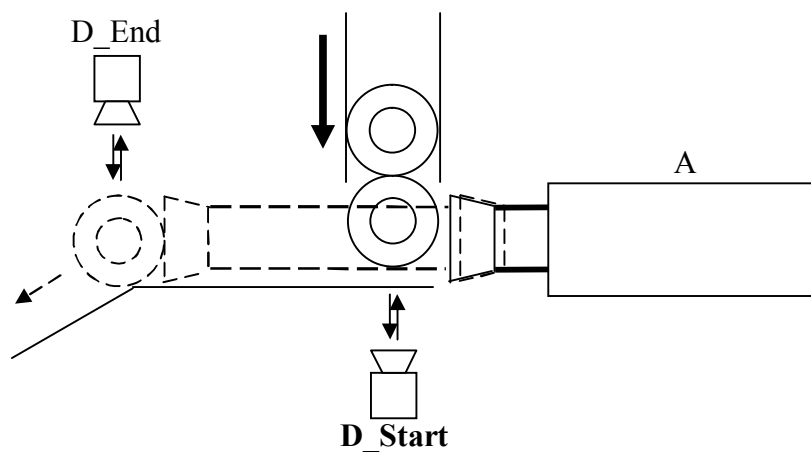
5. Возврат толкателя



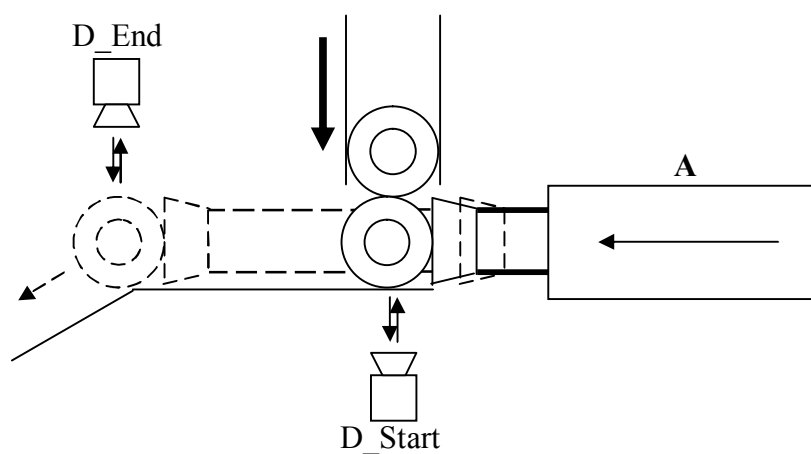
6. Падение заготовки (D_End размыкается)



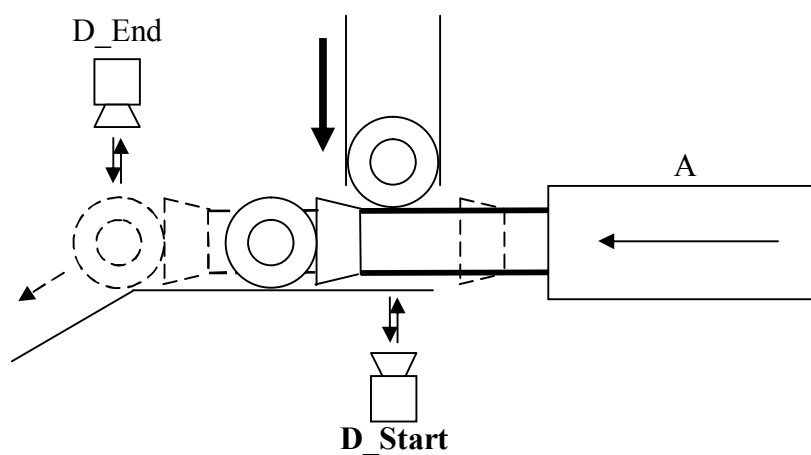
7. Поступление следующей заготовки (D_Start замыкается)



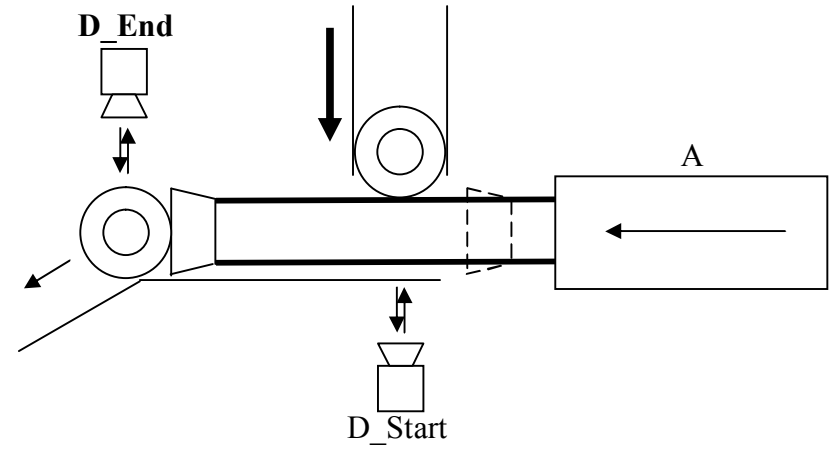
8. Повторное выдвигание цилиндра А



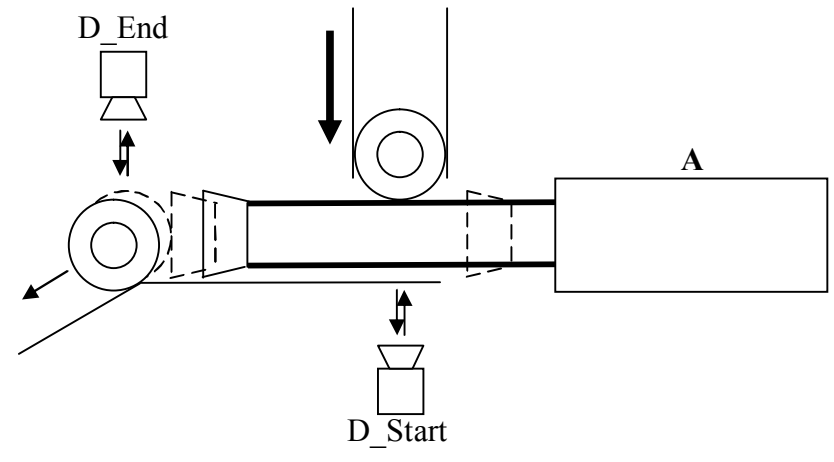
9. Перемещение заготовки (D_Start размыкается)



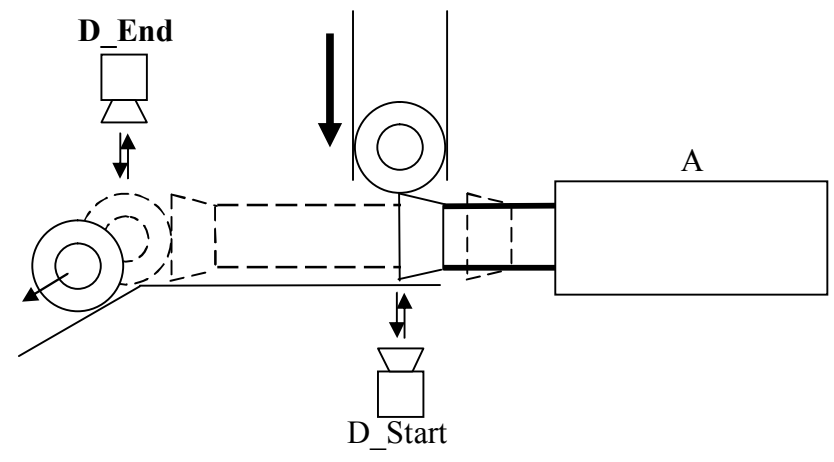
10. Заготовка достигла конца (D_End замыкается)



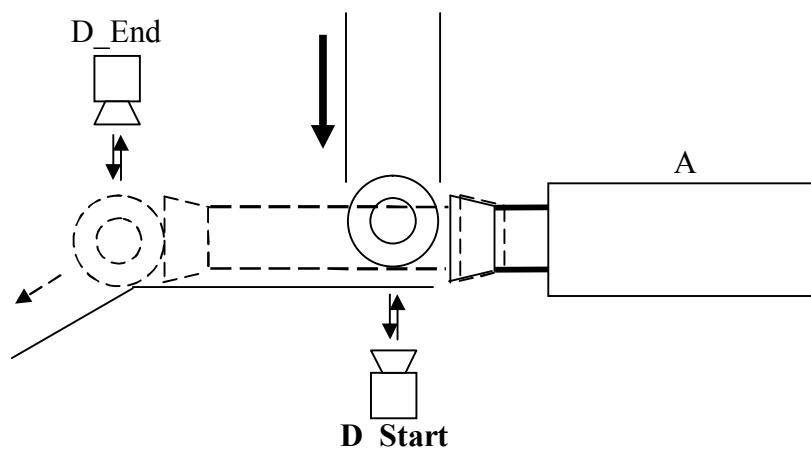
11. Возврат толкателя



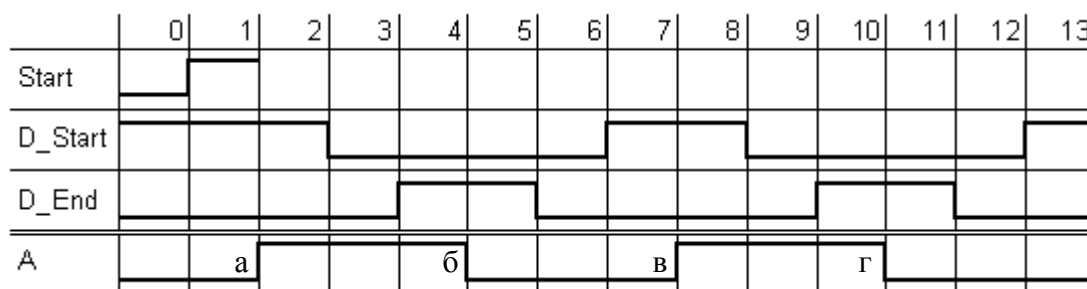
12. Падение заготовки (D_End размыкается)



13. Поступление следующей заготовки (D_Start замыкается)



Последний (13-й) такт совпадает с начальным положением (см. [рис. 3.3.2](#)) и мы можем запустить подачу следующей пары заготовок повторным нажатием кнопки «Start».



[Рис. 3.3.4](#). Циклограмма работы установки, подающей подряд по две заготовки

Казалось бы, используя данную циклограмму ([рис. 3.3.4](#)), уже можно дополнить имеющуюся ([рис. 3.3.3](#)) последовательность действий условиями перехода к следующему действию ([рис. 3.3.5](#)). Но в данном случае необходимо сравнить состояния сигналов на 0-м (а также 1-м) и 7-м тактах. Мы можем видеть идентичные положения сигналов, а это означает, что на 0-м такте установка будет подавать следующую заготовку по условию **D_start** (**в**) независимо от нажатия кнопки «Start».

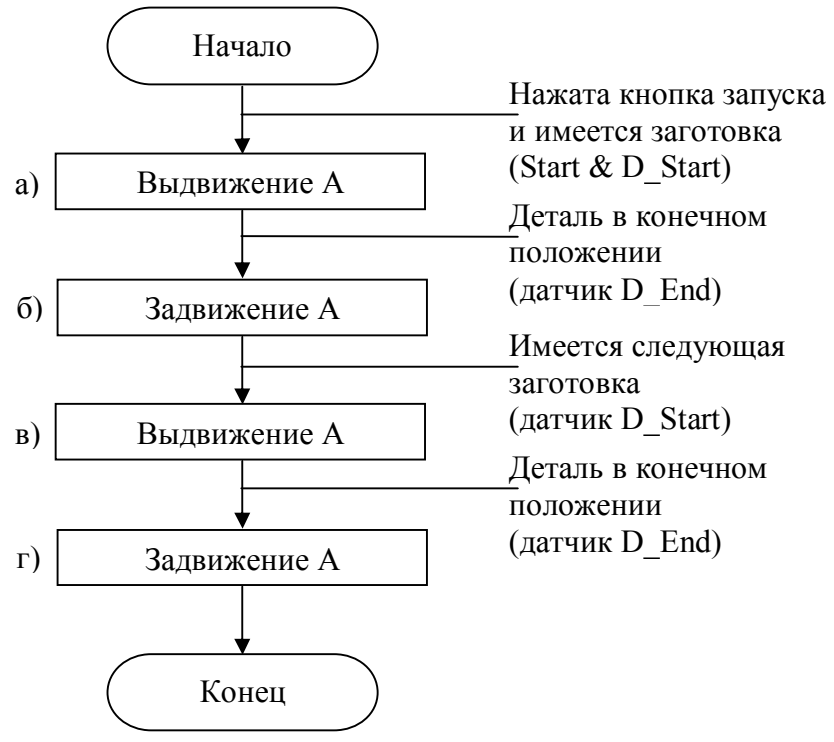


Рис. 3.3.5. Последовательность действий работы установки с предполагаемыми условиями перехода к следующему действию

Правильно в данном случае будет добавить (**рис. 3.3.6**) на циклограмму элемент памяти **X**, сохраняющий номер текущей заготовки: **X=0**, если заготовка первая, изменяющий свое состояние на **X=1**, если следующей будет вторая заготовка. После подачи второй заготовки элемент памяти снова сбрасывается в **X=0**. Переменная **X** может иметь и более осмысленное название, например **Step2** – «Подача второй заготовки».

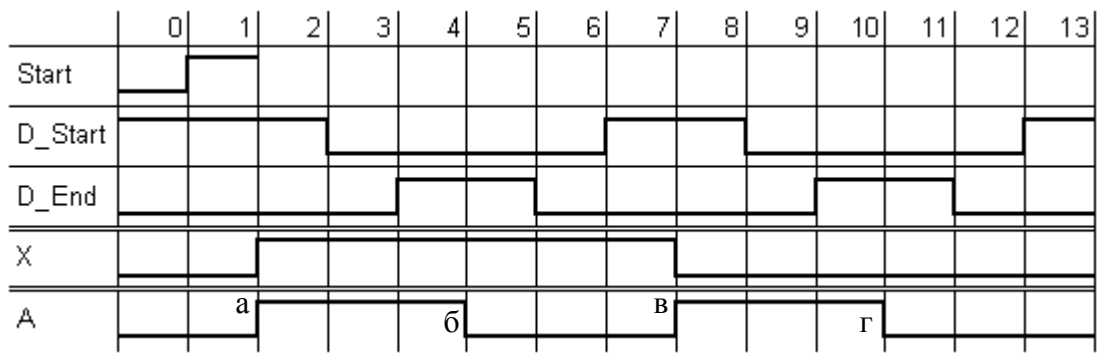


Рис. 3.3.6. Циклограмма, дополненная элементом памяти

Составим последовательность действий (рис. 3.3.7) для новой циклограммы (рис. 3.3.6), дополненной элементом памяти.

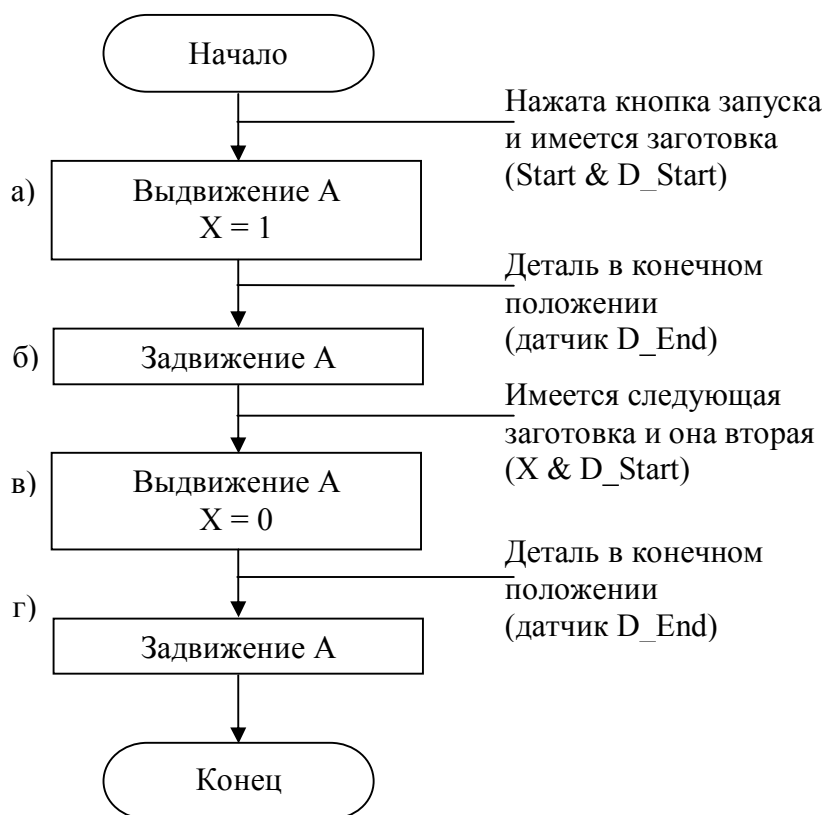


Рис. 3.3.7. Последовательность действий работы установки, дополненная элементом памяти

Используя данную последовательность действий (рис. 3.3.7), можно составить блок-схему (рис. 3.3.8) работы установки, подающей по две заготовки подряд, которую можно упростить, исключив повторяющиеся участки (участок б аналогичен участку г).

В рассмотренном варианте предполагается, что по умолчанию при включении установки $X=0$.

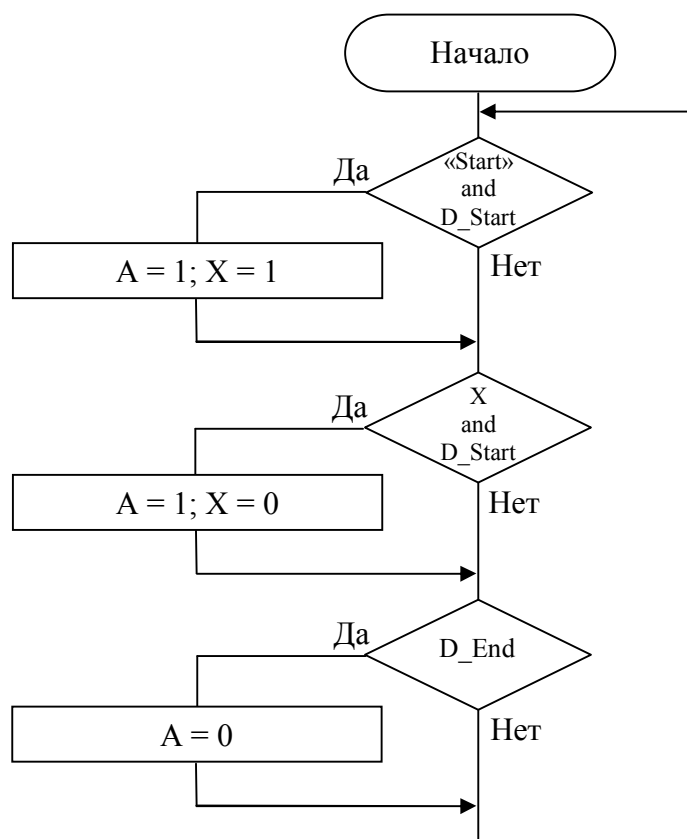


Рис. 3.3.8. Блок-схема работы установки, подающей подряд по две заготовки

По данной блок-схеме можно составить следующий программный код:

```

Если (Start и D_Start), то {
    Вкл. А
    Вкл. Х
}
Если (X и D_Start), то {
    Вкл. А
    Выкл. Х
}
Если (D_End), то {
    Выкл. А
}
  
```

Другой возможный способ решения данной задачи – разбиение на шаги (рис. 3.3.9 и 3.3.10). При этом первым шагом будет подача первой заготовки, а вторым – подача второй заготовки.

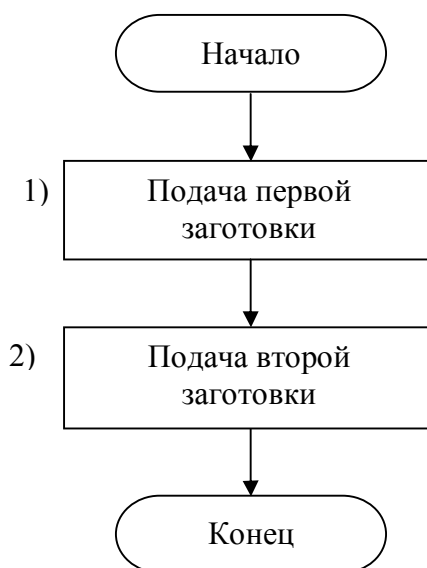


Рис. 3.3.9. Последовательность действий работы установки (по шагам)



Рис. 3.3.10. Циклограмма работы установки (по шагам)

Как и прежде, условием перехода к следующему шагу является последнее из условий текущего шага (рис. 3.3.11 и 3.3.12).

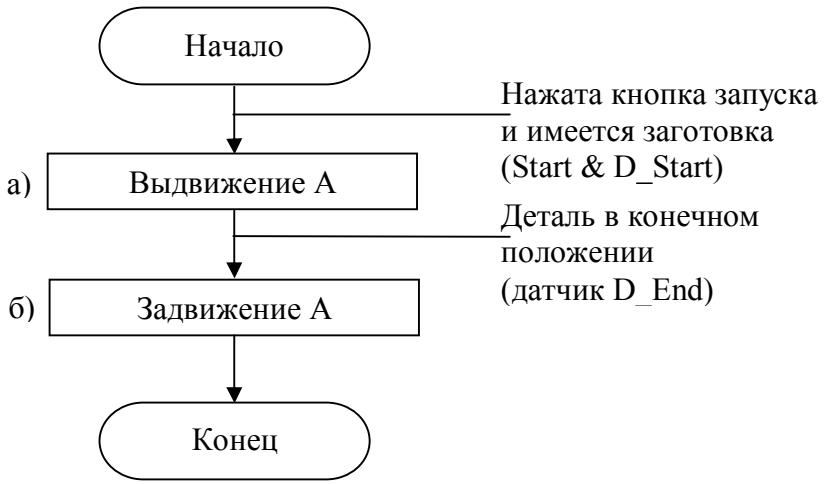


Рис. 3.3.11. Шаг 1. Подача первой заготовки

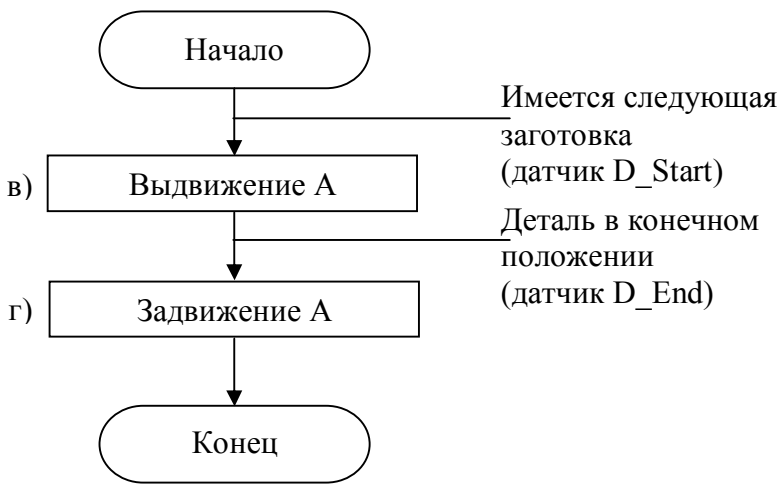


Рис. 3.3.12. Шаг 2. Подача второй заготовки

В данном случае оба варианта (рис. 3.3.6 и 3.3.10) практически идентичны, только в первом случае за разбиение на шаги отвечает переменная X, и изображается на циклограмме (рис. 3.3.6) между входными и выходными параметрами, а во втором специальная переменная с номером шага, которая изображается (рис.3.3.10) вверху циклограммы.

Используя данную последовательность действий, можно составить блок-схему работы установки, состоящую из двух шагов (рис. 3.3.13).

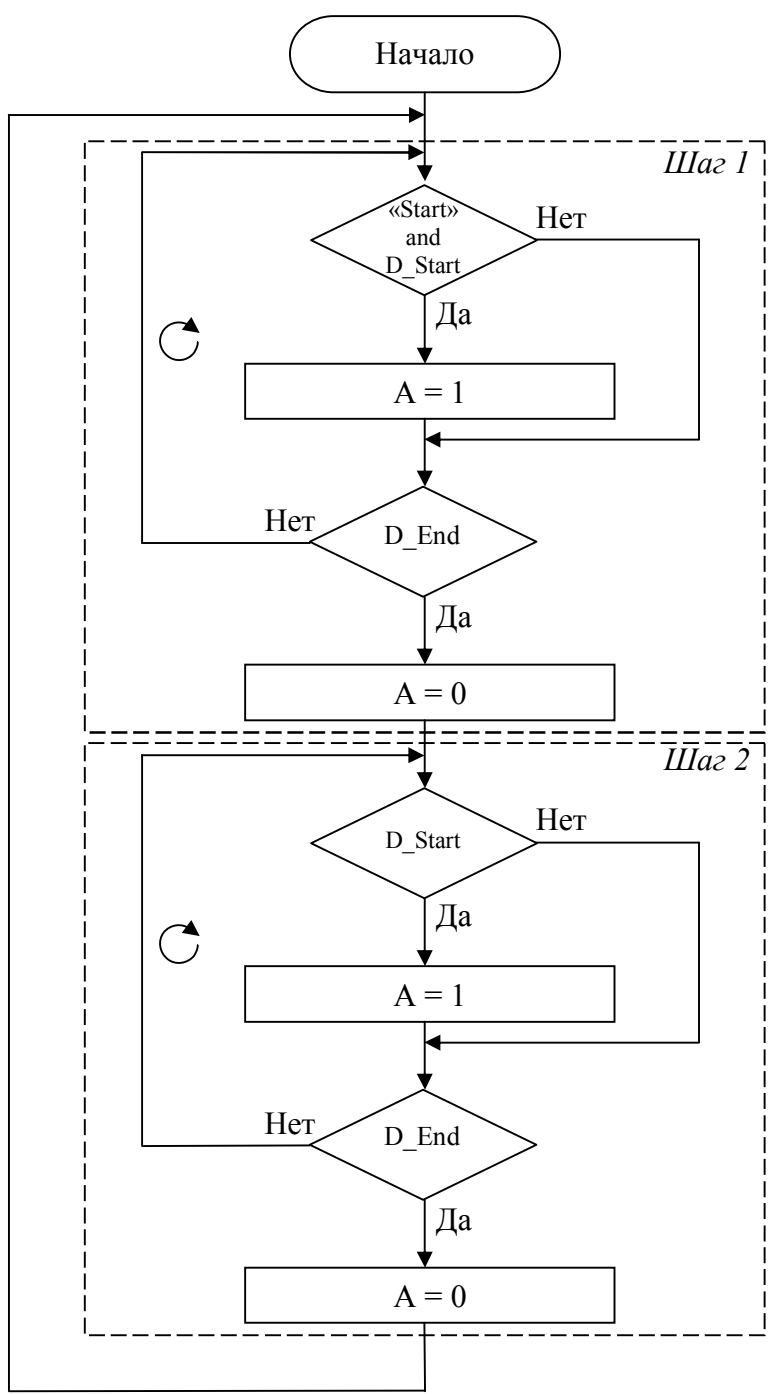


Рис. 3.3.13. Блок-схема работы установки, подающей подряд по две заготовки (по шагам)

По данной блок-схеме несложно получить следующий программный код:

Шаг 1

Если (Start и D_Start), то {

 Вкл. А

}

Если (D_End), то {

 Выкл. А

}

Шаг 2

Если (D_Start), то {

 Вкл. А

}

Если (D_End), то {

 Выкл. А

 ПЕРЕЙТИ НА ШАГ 1

}

3.4. Таймеры

Таймер – это устройство для отсчета времени. Для начала работы таймера его необходимо инициализировать (подготовить к работе, активировать), т.е. задать [8] требуемые параметры (такие как интервал времени), после чего запустить. На блок-схеме инициализация таймера осуществляется (рис. 3.4.1) в блоке действий («прямоугольник» [6,7]).



Рис. 3.4.1. Блок инициализации таймера

Таймер работает следующим образом (рис. 3.4.2):

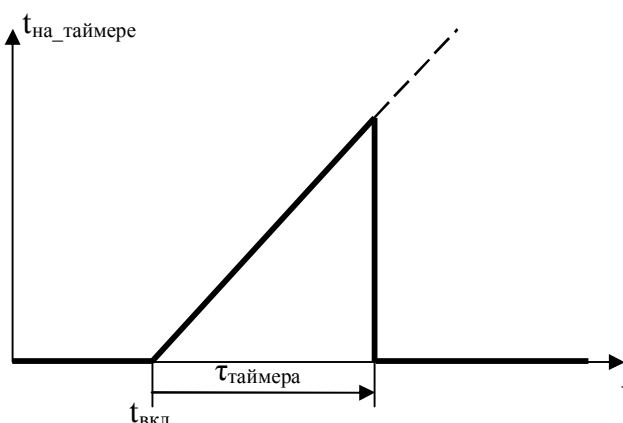


Рис. 3.4.2. Принцип работы таймера с прямым отсчетом

Вначале время на таймере равно нулю. При включении таймера в момент времени $t_{\text{вкл}}$ время на таймере ($t_{\text{на_таймере}}$) начинает увеличиваться. По прошествии времени $\tau_{\text{таймера}}$, таймер останавливается.

Возможны также варианты таймеров с обратным отчетом (рис. 3.4.3), в которых время уменьшается от заданного значения $\tau_{\text{таймера}}$ до нуля.

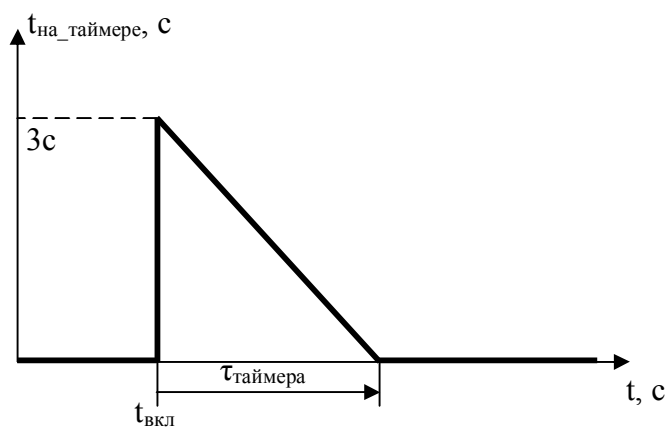


Рис. 3.4.3. Принцип работы таймера с обратным отсчетом

Состояние таймера

Состояние таймера (**рис. 3.4.4**) является дискретной величиной, т.е. таймер может быть либо «**ВКЛЮЧЕН**», либо «**ВЫКЛЮЧЕН**».

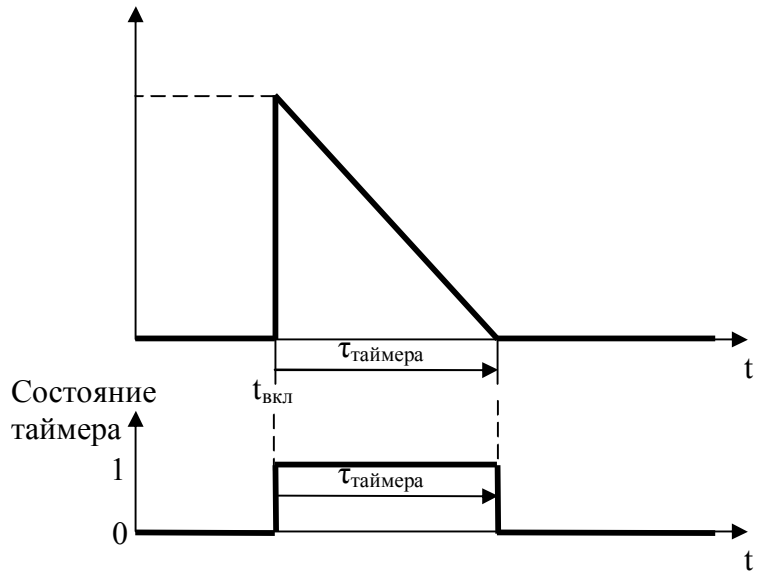


Рис. 3.4.4. Состояние таймера

Именно состояние таймера используется при проверке условий срабатывания таймера, и именно состояние таймера используется при составлении циклограммы. На циклограмме таймер изображается в виде отдельной переменной, между входными и выходными параметрами. Время

работы таймера указывается непосредственно на циклограмме, между моментами его включения и выключения (см. **примеры 5 и 6**).

Пример 5

Необходимо запустить конвейерную ленту **через 10 секунд** после нажатия кнопки «**Start**».

Циклограмма работы данного примера изображена на **рис. 3.4.5**.

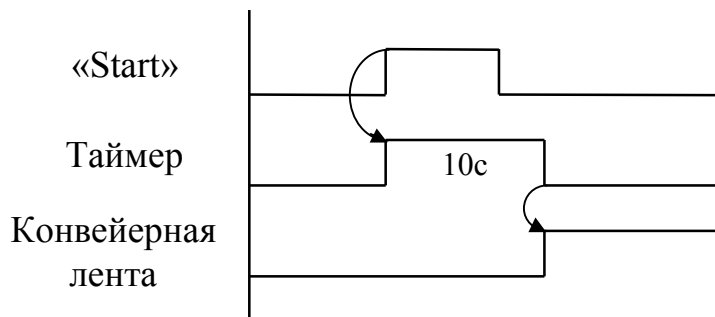


Рис. 3.4.5. Циклограмма запуска конвейерной ленты через 10 секунд

Пример 6

Необходимо запустить конвейерную ленту **на 10 секунд** нажатием кнопки «**Start**».

Циклограмма работы данного примера изображена на **рис. 3.4.6**.

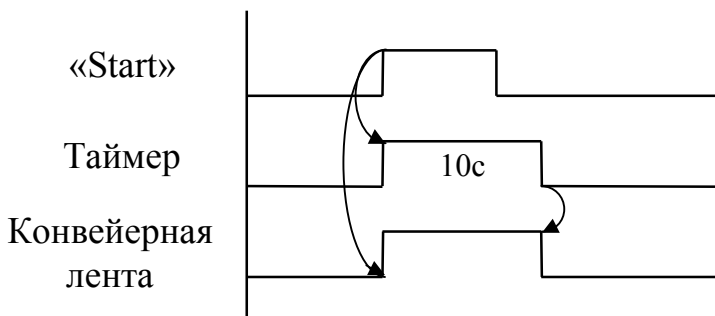


Рис. 3.4.6. Циклограмма работы конвейерной ленты в течение 10 секунд

Двойственность таймера.

Таймер как внешнее устройство

Таймер может быть рассмотрен в виде внешнего устройства с заранее заданным интервалом времени (рис. 3.4.7), при этом выходной сигнал с контроллера запускает таймер, по истечении заданного времени сигнал на входе контроллера пропадает (или, в зависимости от его модификации, появляется). Естественно, что при такой схеме подключения мы не можем задавать время работы таймера ($\tau_{\text{таймера}}$) при запуске, и оно должно быть заранее задано в самом таймере.

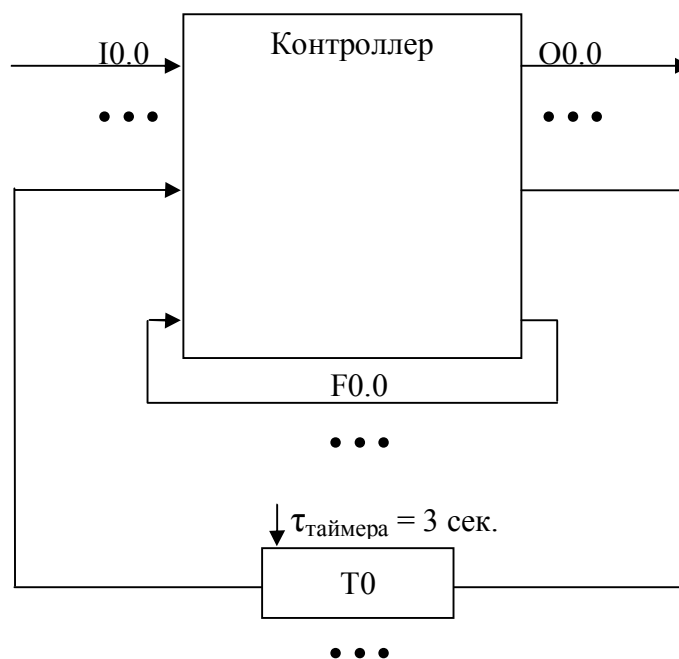


Рис. 3.4.7. Таймер как внешнее устройство

Таким образом, таймер, как и элементы памяти, имеет двойственную природу, и обладает свойствами как выходных (запуск таймера), так и входных (проверка состояния таймера) переменных.

Таймер обычно имеет обозначение «Т», «Tmr» или «Timer».

4. ОПИСАНИЕ ПРОГРАММЫ SMC PNEUDRAW2.5

Программа SMC PneuDraw разработана японской компанией SMC-Pneumatik как средство проектирования пневматических схем. Интерфейс программы PneuDraw основан на семействе программного обеспечения MS Visio [9].

4.1. Интерфейс программы

На [рис. 4.1.1](#) изображен интерфейс программы PneuDraw. Он состоит из четырех главных частей.

- „Главное меню” – Содержит элементы управления программой, вызов подменю, сохранение и открытие проектов.
- „Панель инструментов” – Содержит разделенные по группам кнопки, нажатием на которые, можно получить доступ ко многим функциям программы. Все кнопки дублируют функции, находящиеся в главном меню, но к ним можно получить доступ одним нажатием мышки.
- „Библиотека графических элементов” – Элементы распределены в библиотеке по группам. Отображение подгрупп происходит при нажатии на символ «+» рядом с имени группы. Отображение элементов групп происходит при нажатии на группу или подгруппу в нижней части области. Обозначение пневматических элементов и их графическое отображение в программе см. в [Приложении](#).
- „Область рисования” – Область, куда происходит перемещение графических объектов, связей между ними. Для перемещения объекта выберите его в нижней части библиотеки графических элементов и, не отпуская клавиши мыши, перетащите на панель рисования.

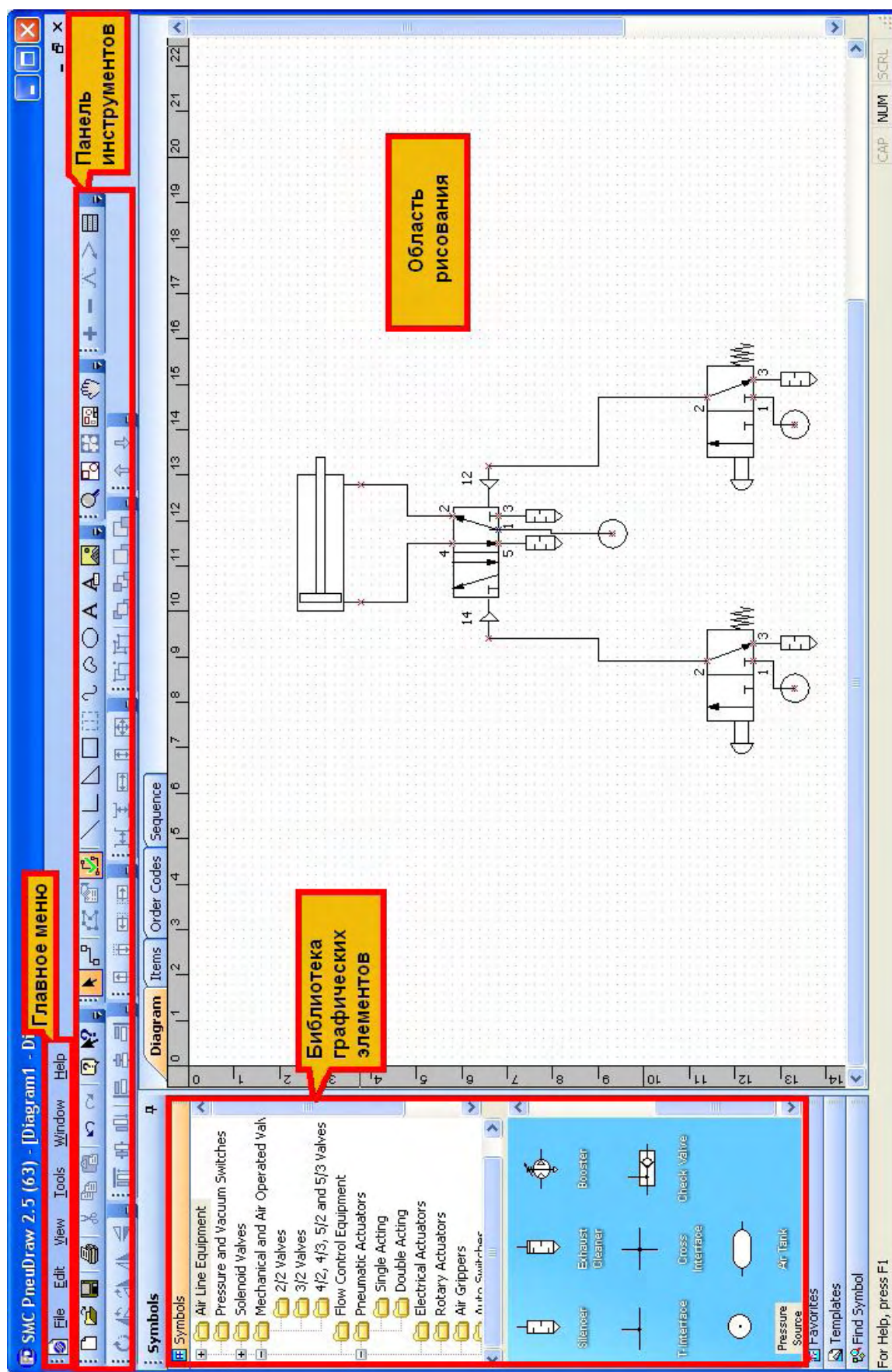




Рис. 4.1.1. Интерфейс программы PneuDraw 2.5


4.2. Работа в программе

В лабораторных работах ограничимся составлением простейших пневматических схем.


Для начала создайте новый проект, нажав в главном меню на надпись «File». Из выпавшего списка выберите «New». Либо нажмите клавишу  на панели инструментов.

Для открытия сохраненного проекта выберите в главном меню пункт «File», далее нажмите в выпавшем списке пункт «Open». Либо нажмите клавишу  в панели инструментов.

Далее расположите пневматические элементы на области рисования. Для этого выберите в библиотеке элементов интересующую вас группу, затем выберите в нижней части библиотеки нужный вам элемент и «перетащите» его на область рисования.

Для создания связей между объектами щелкните на клавишу  панели инструментов. Далее щелкните по месту присоединения на первом объекте (при этом, при приближении курсора к месту присоединения, программа обозначит его красным квадратом). Щелкните по месту связи на втором объекте для завершения процедуры создания связи.

Некоторые объекты (например, глушители) могут присоединяться напрямую к объектам. Для этого при «перетаскивании» объекта из библиотеки, подведите его к месту присоединения нужного объекта.

Для сохранения проекта нажмите клавишу  в панели инструментов, далее выбрав директорию сохранения и имя файла. Либо выберите пункт «Save» или «Save As» в меню «File» главного меню.

4.3. Пример выполнения работы

Постановка задачи: Машина для обрезки листов бумаги (рис. 4.3.1) до заданного формата снабжена пневматическим приводом. Для обеспечения безопасности работы оператора пуск должен производиться только при нажатии двух кнопок. Возврат резака осуществляется автоматически после выполнения рабочей операции.

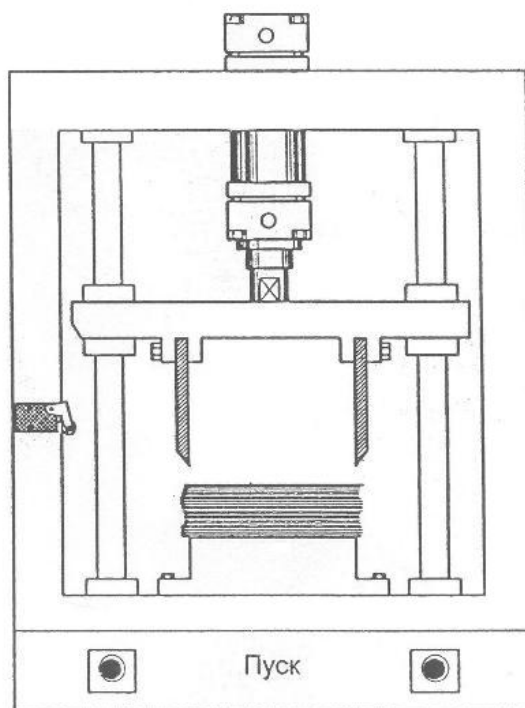


Рис. 4.3.1. Бумагорезательная машина

Задание: Разработать принципиальную пневматическую схему бумагорезательной машины на базе пневмоцилиндра двухстороннего действия.

Выполнение работы:

Запустите программу PneuDraw2.5 двойным щелчком мыши на ярлыке SmcPneuDraw.

По умолчанию откроется новый проект. Для загрузки старого или создания нового проекта щелкните мышкой на соответствующих пиктограммах панели инструментов.

Далее перетащим необходимые компоненты из библиотеки элементов. Обратите внимание, что отдельно кнопок, пружин, толкателей, рычагов и т.п. в библиотеке нет. Такие элементы уже присоединены к пневмораспределителям.

В постановке задачи сказано, что для управления бумагорезательной машиной необходимо две кнопки.

В библиотеке программы распределители с различными элементами управления находятся в группе “Mechanical and Air Operated Valves”. Далее они распределены по подгруппам “2/2 Valves”, “3/2 Valves”, “4/2, 4/3, 5/2 and 5/3 Valves”.

Для правильной работы бумагорезательной машины необходимо, чтобы при нажатии двух кнопок происходило выдвижение пневмоцилиндра, а при окончании операции происходил возврат штока пневмоцилиндра обратно, т.е. воздух из пневмоцилиндра должен выходить через распределитель, для этого используется пневмораспределитель 3/2 с возвратной пружиной. Управление распределителем осуществляется при помощи кнопки. Данный пневмораспределитель находится в подгруппе “3/2 valves”, и называется “3/2 mechanical valve – (см. **рис. 4.3.2**). «Перетащим» два таких распределителя из библиотеки на область рисования.



Рис. 4.3.2. Расположение распределителей 3/2:

- 1 - пневмораспределитель с рычажным управлением;
- 2 - пневмораспределитель с кнопочным управлением и возвратной пружиной

Возврат штока пневмоцилиндра по завершении операции можно выполнить посредством установки концевого выключателя, который при выдвигении штока пневмоцилиндра на 100% “дает команду” штоку задвинуться. В роли концевого выключателя может выступать пневмораспределитель 3/2 с рычагом. При выдвигении пневмоцилиндра шток нажимает на рычаг, и воздух поступает в полость закрытия пневмоцилиндра.

В пневмосхемах, у распределителей входы и выходы, к которым можно привести линии присоединения, отображают нормальное (нерабочее) состояние распределителя.

При подаче возмущения, например, нажатием на кнопку (для распределителя 3/2 с кнопочным управлением и возвратной пружиной) распределитель “смещается” в сторону нажатия, т.е. вправо. При этом смещаются линии соединения.

На рис. 4.3.3 показан пневмораспределитель в нерабочем положении (1) и тот же распределитель, но при нажатой кнопке (2). Стрелками показано возможное направление движения воздуха.

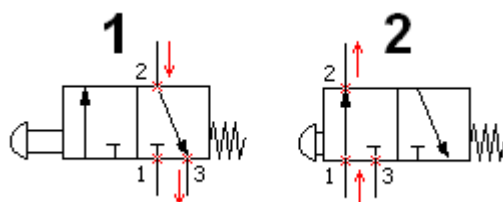


Рис. 4.3.3. Распределитель 3/2 в обычном положении и при нажатой кнопке:

- 1- нерабочее положение;
- 2- рабочее состояние

Вынесите пневмораспределитель с рычажным управлением на панель рисования. Для того чтобы пневмоцилиндр открывался только при нажатии двух кнопок, необходим логический элемент «и».

Напомним, что логический элемент «и» срабатывает только тогда, когда на два его входа подается рабочий воздух. В библиотеке элементов он располагается в группе “Flow Control Equipment” и называется “AND Valve” (рис. 4.3.4). Вынесем на панель рисования логический элемент «и».

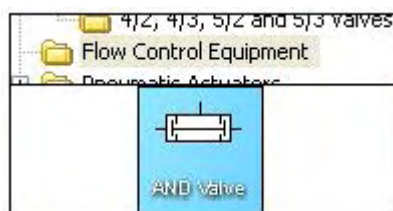


Рис. 4.3.4. Логический элемент «И»

Пневмоцилиндр двустороннего действия расположен в библиотеке элементов в группе “Pneumatic Actuators” в подгруппе “Double Acting”. Вытащим на панель рисования обычный пневмоцилиндр двустороннего действия. В библиотеке элементов он назван “Without Magnet”.

Для того чтобы пневмоцилиндр выдвигал шток при нажатии двух кнопок и возвращал шток в начальное положение, необходим ещё один пневмораспределитель, который управляется посредством сжатого воздуха.

При подаче воздуха на одно из двух управляющих отверстий (они также могут называться пилотными клапанами, а управляющий воздух - пилотным) данный распределитель переключается на соответствующее положение.

На **рис. 4.3.5** показан пример работы данного пневмораспределителя. При подводе воздуха к пилотному клапану 14 (1), распределитель переключается в «левое» положение. Воздух выходит из отверстия 4, отверстие 2 выпускает воздух.

При подводе управляющего воздуха в пилотный клапан 12 (2), распределитель переключается в другое положение, и подача воздуха осуществляется наоборот. При подключении к такому распределителю двухстороннего пневмоцилиндра, он будет открываться или закрываться при подаче воздуха на управляющие отверстия 12 или 14.

Следует отметить, что при подаче управляющего воздуха в оба пилотных клапана, приоритет будет иметь то положение, которое было до одновременной подачи.

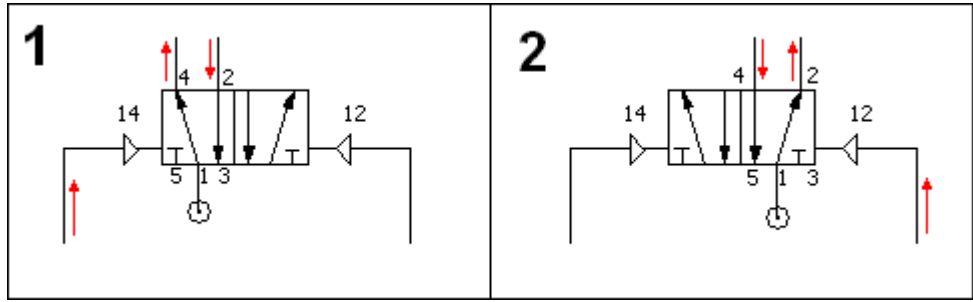


Рис. 4.3.5. Работа пневмораспределителя 5/2:
 1- Подача пилотного воздуха в клапан 14;
 2- Подача пилотного воздуха в клапан 12

Нужный пневмораспределитель находится в группе “Mechanical and Air Operated Valves”, подгруппа “4/2, 4/3, 5/2 and 5/3 Valves”. Называется “5/2 Air Operated Valve”. Вытащите его на панель рисования.

Для работы пневмооборудования необходим источник сжатого воздуха. На схеме его также нужно указать. Данный элемент находится в группе “Air Line Equipment” и называется “Pressure source” (рис. 4.3.6).

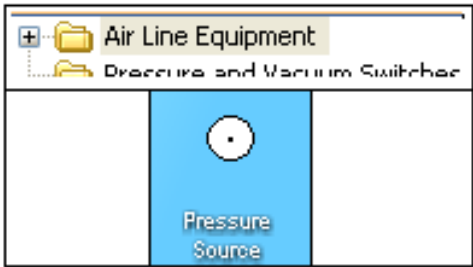


Рис. 4.3.6. Источник сжатого воздуха

Все необходимые элементы находятся на панели рисования. Сгруппируем их для облегчения чтения пневмосхемы. Например, как показано на рис. 4.3.7.

Так как пневмооборудование в процессе работы производит много шума, на него желательно ставить пневмоглушители.

Поставим пневмоглушители (группа “Air Line Equipment”, название “Silencer”) на все отверстия пневмораспределителей из которых выходит воздух. Элементы с пневмоглушителями показаны на рис. 4.3.8.

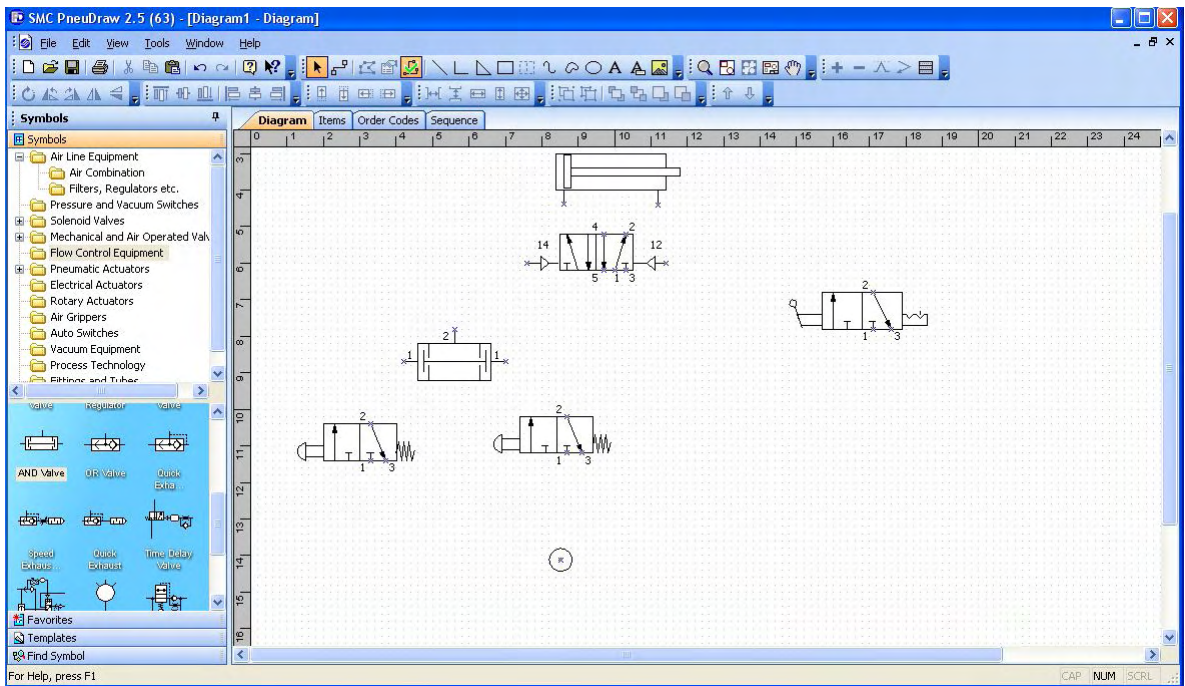


Рис. 4.3.7. Пример расположения элементов

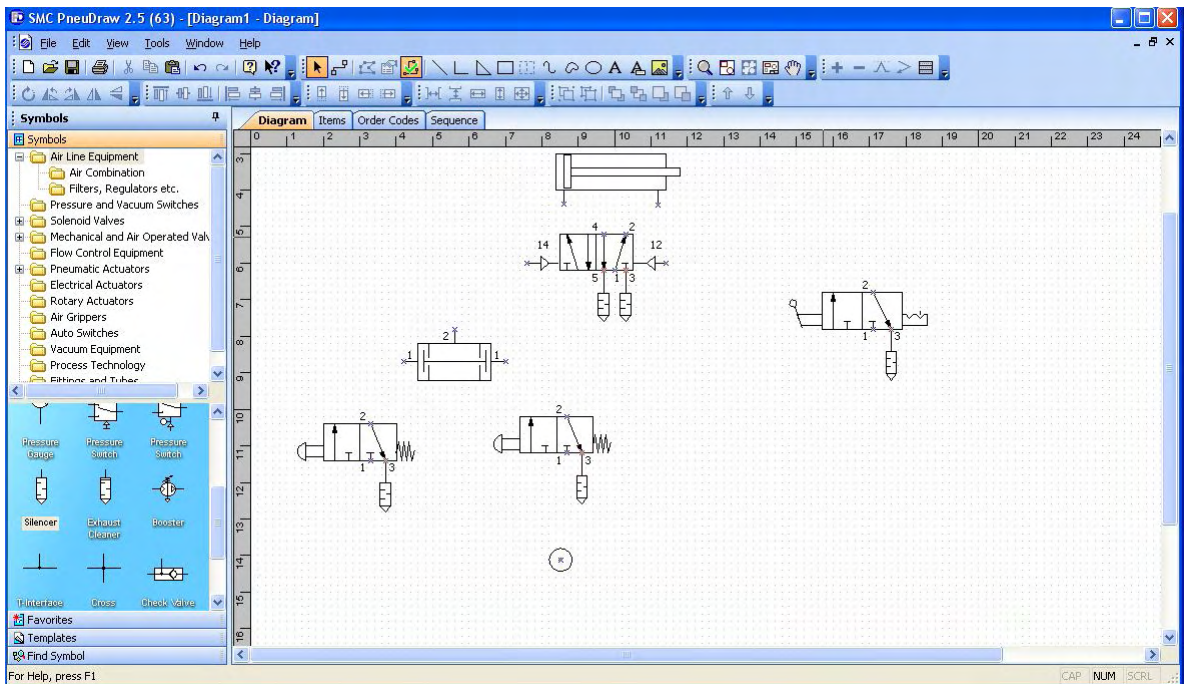


Рис. 4.3.8. Пример расположения пневмоглушителей

Все элементы необходимо связать между собой. Линии связи на схемах - это трубки и фитинги в реальной модели. Для облегчения чтения схемы на ней можно расположить несколько источников сжатого воздуха. Пример выполненной работы в программе PneuDraw см. на **рис. 4.3.9.**

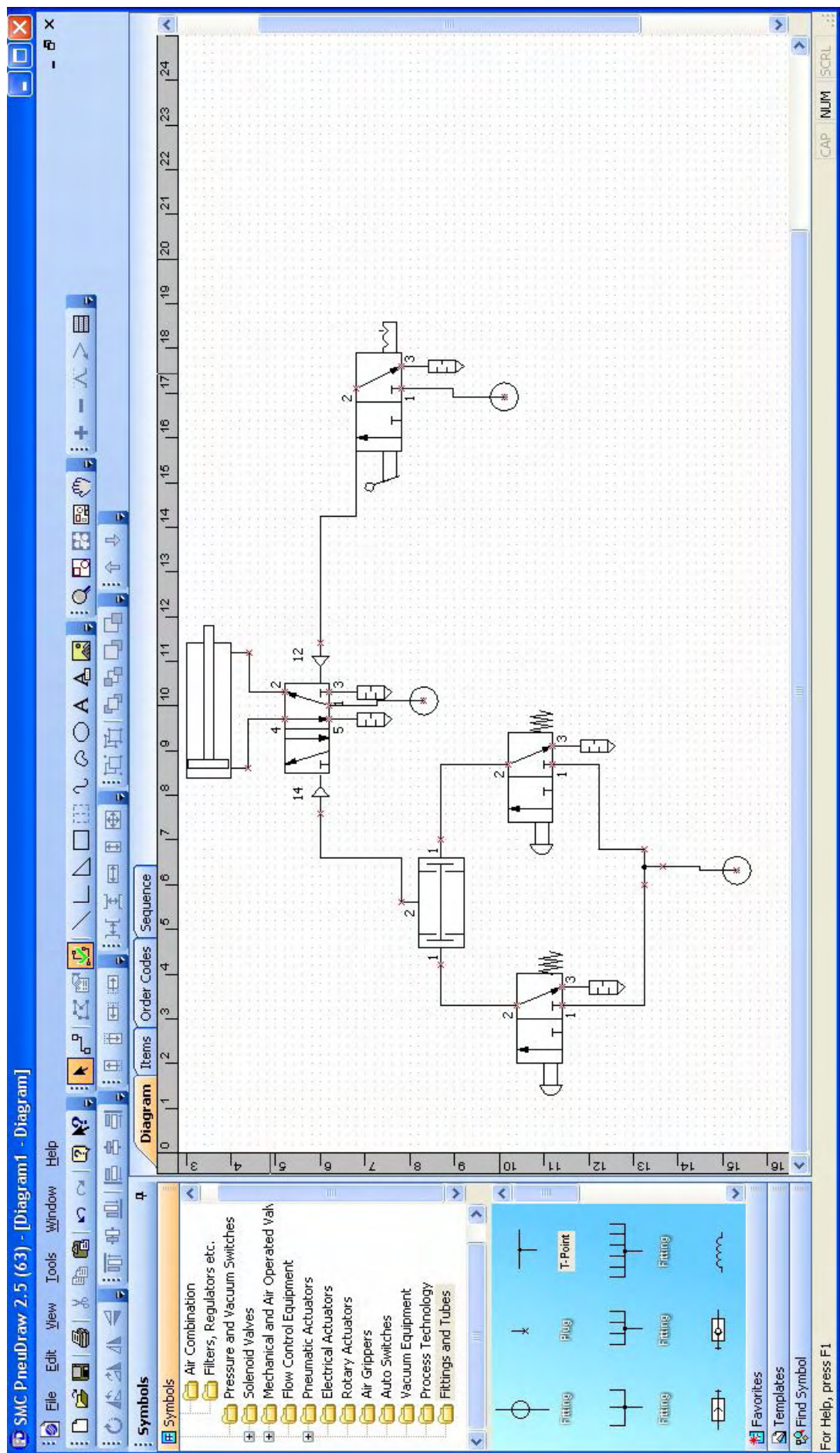


Рис. 4.3.9. Пример выполненной работы

Задания для выполнения лабораторной работы

Для выполнения лабораторных работ по проектированию пневматических схем в программе SMC PneuDraw задание выдается для двух лабораторных работ варианты, которых изложены ниже.

Вариант 1

Постановка задачи: Отсортированная бумага в брикетах подается на транспортный конвейер с двух рабочих мест посредством пневматического толкателя.

Выдвижение штока толкателя должно производиться с левого или правого рабочего места при кратковременном нажатии на соответствующую пневматическую кнопку «Пуск» (рис. 4.3.10).

Возврат толкателя в исходную позицию осуществляется автоматически.

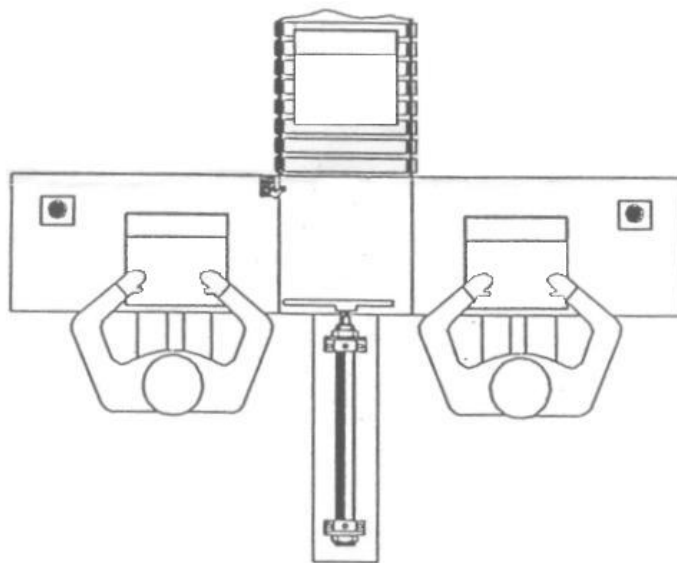


Рис. 4.3.10. Пневматический толкатель на два рабочих места

Задание: Разработать принципиальную пневматическую схему пневматического толкателя с двумя альтернативными пусковыми кнопками на базе пневмоцилиндра двухстороннего действия.

Написать отчет по выполненной работе.

Вариант 2

Постановка задачи: В муфельной печи дверь должна приводиться в движение пневмоприводом на базе пневмоцилиндра двухстороннего действия. Закрытие и открытие двери должно происходить кратковременным нажатием соответствующих пневматических кнопок (рис.4.3.11).

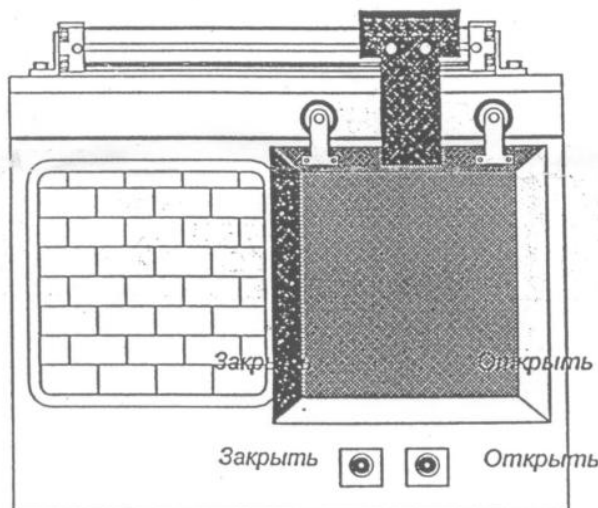


Рис. 4.3.11. Муфельная печь с пневматической дверью

Задание: Разработать принципиальную пневматическую схему привода муфельной печи на базе пневмоцилиндра двухстороннего действия. Написать отчет по выполненной работе.

Оформление отчета

Отчет по лабораторной работе должен содержать:

1. Титульный лист.
2. Задание на выполнение работы (вариант 1 или вариант 2).
3. Список используемых элементов и их условные обозначения (выбранные из библиотеки элементов программы PneuDraw).
4. Распечатку построенной в программе PneuDraw принципиальной пневматической схемы.
5. Обоснование выбора пневматических элементов.

Содержание курсового проекта (работы)

Курсовая работа должна содержать:

1. Титульный лист.
2. Задание.
3. Описание используемого оборудования.
4. Таблицы входных и выходных переменных.
5. Схему подключения оборудования к контроллеру.
6. Циклограмму работы оборудования.
7. Блок-схему системы управления работой оборудования.
8. Код программы управления.

Примерный список тем экзамена

Билет состоит из теоретической и практической (задачи) частей:

Теоретическая часть

1. Последовательность действий.
2. Блок-схема.
3. Таймеры.
4. Элементы памяти.
5. Входные и выходные переменные.
6. Ввод-Вывод контроллера.
7. Написание программного кода по имеющейся блок-схеме.
8. Вложенные действия.
9. Подпрограммы.
10. Шаги.
11. Описание работы пневматического оборудования. Цилиндры.
12. Циклограмма.
13. Конечные выключатели. Датчики обнаружения.
14. Контроллер фирмы Festo. Язык FST. Синтаксис языка FST [8].
15. Сортировочная станция фирмы Festo [8].

Практическая часть

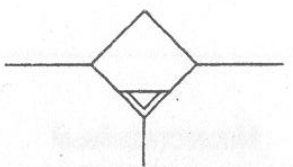
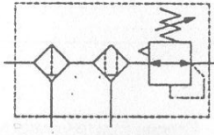
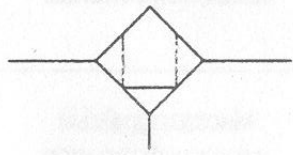
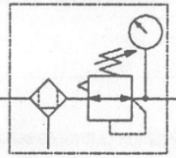
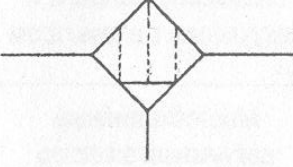

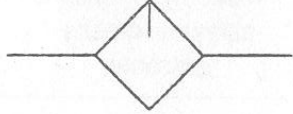
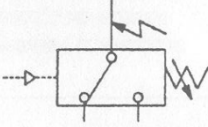
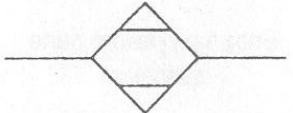
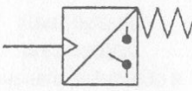
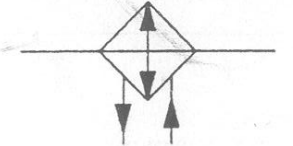
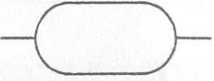


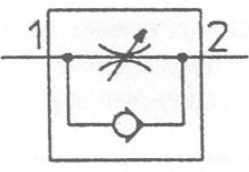
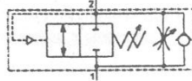
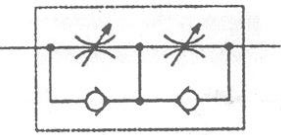
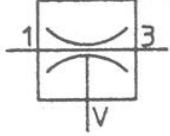
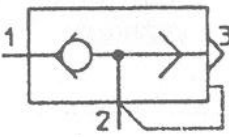
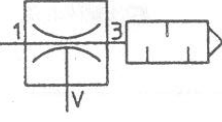
1. Изобразить циклограмму работы оборудования.
2. Изобразить блок-схему и программный код.

Приложение

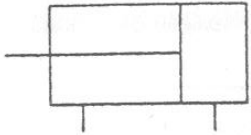
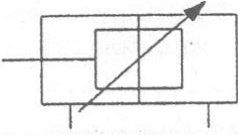
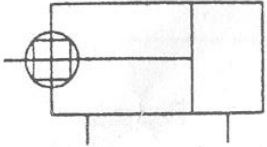
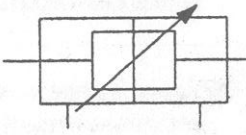
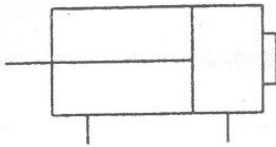
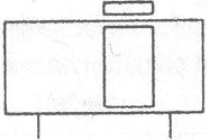
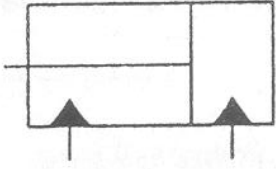
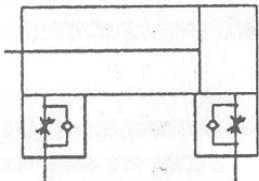
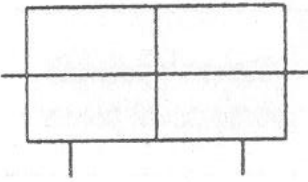
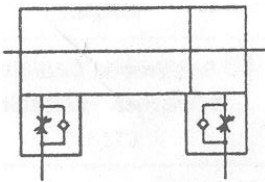
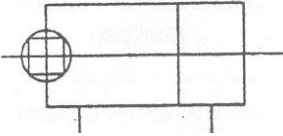
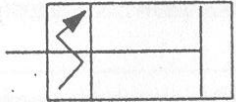
Символы для обозначения пневмооборудования

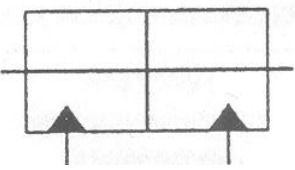
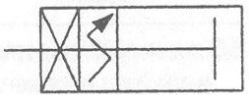
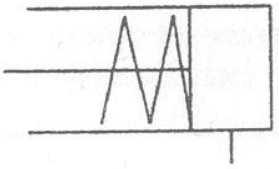
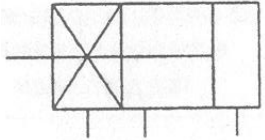
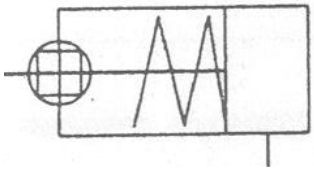
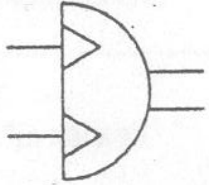
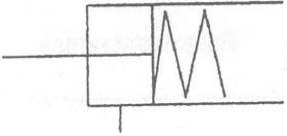
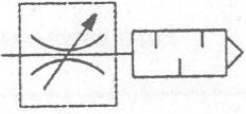
	2/2 пневмораспределитель, нормально закрытый		Ручное управление (обобщенный символ)
	2/2 пневмораспределитель, нормально открытый		Нажимная кнопка
	3/2 пневмораспределитель, нормально закрытый		Рычаг
	3/2 пневмораспределитель, нормально открытый		Педадь
	3/3 пневмораспределитель, в среднем положении закрыт		Механический толкатель
	4/2 пневмораспределитель		Возвратная пружина
	4/3 пневмораспределитель, в среднем положении закрыт		Роликовый рычаг
	4/3 пневмораспределитель, в среднем положении открыт		Односторонний роликовый рычаг
	5/2 пневмораспределитель		Электромагнит с одной обмоткой

	5/3 пневмораспределитель, в среднем положении закрыт		Электромагнит с двумя противодействующими обмотками в одном узле
	5/3 пневмораспределитель, в среднем положении открыт		Непрямое электромагнитное управление
	5/3 пневмораспределитель, в среднем положении под давлением		Управление посредством сжатого воздуха
	Клапан «ИЛИ»		Фиксатор
	Клапан «И»		Указатель давления
	Реле давления		Пневмоглушитель
	Редукционный клапан со сбросом давления		Блок подготовки воздуха (упрощенное значение)
	Фильтр-влагоотделитель		Блок подготовки воздуха (фильтр, редукционный клапан, маслораспылитель)
	Фильтр-влагоотделитель с автоматическим отводом конденсата		Фильтр/Регулятор (комбинация воздушного фильтра и редукционного клапана)

	Устройство автоматического отвода конденсата		Блок подготовки воздуха (фильтр, микрофильтр, редукционный клапан)
	Микрофильтр		Блок подготовки воздуха (микрофильтр, редукционный клапан, манометр)
	Субмикрофильтр		Усилитель давления с ручной регулировкой
	Маслораспылитель		Реле давления регулируемое
	Осушитель		Реле давления нерегулируемое
	Охладитель		Ресивер
	Манометр		Обратный клапан без пружины
	Дроссель с обратным клапаном		Блок управления скоростью исполнительного звена (для регулирования входящего потока)
	Блок из двух дросселей с обратными клапанами для управления потоком в обоих направлениях		Вакуумный эжектор
	Клапан быстрого выхлопа		Вакуумный эжектор с пневмоглушителем

	<p>Клапан быстрого выхлопа с пневмоглушителем</p>		<p>Многостадийный вакуумный эжектор с фильтром и пневмоглушителем</p>
	<p>Клапан быстрого выхлопа с дросселем и пневмоглушителем</p>		<p>Многостадийный вакуумный эжектор с фильтром, пневмоглушителем и вакуумным манометром</p>
	<p>Дроссель с обратным клапаном и клапаном сброса остаточного давления</p>		<p>Многостадийный вакуумный эжектор с фильтром, пневмоглушителем и вакуумным реле давления</p>
	<p>Пневмозамок с дросселем и обратным клапаном</p>		<p>Блок вакуумного реле давления</p>
	<p>Блок управления скоростью исполнительного звена (для регулирования выходящего потока)</p>		<p>Фильтр</p>
	<p>Пневмоцилиндр одностороннего действия</p>		<p>Пневмоцилиндр одностороннего действия тянущего типа с пружинным возвратом и непроворотным штоком</p>

	<p>Пневмоцилиндр двустороннего действия</p>		<p>Пневмоцилиндр двустороннего действия с регулируемым пневматическим демпфером</p>
	<p>Пневмоцилиндр двустороннего действия с неповоротным штоком</p>		<p>Пневмоцилиндр двустороннего действия с регулируемым пневматическим демпфером и двусторонним штоком</p>
	<p>Пневмоцилиндр двустороннего действия с задней монтажной опорой и односторонним штоком</p>		<p>Бесштоковый магнитный пневмоцилиндр</p>
	<p>Пневмогидравлич еский цилиндр двустороннего действия</p>		<p>Пневмоцилиндр двустороннего действия с односторонним штоком и встроенными дресселями</p>
	<p>Пневмоцилиндр двустороннего действия с двусторонним штоком</p>		<p>Пневмоцилиндр двустороннего действия со сквозным штоком и встроенными дресселями</p>
	<p>Пневмоцилиндр двустороннего действия с двусторонним неповоротным штоком</p>		<p>Пневмоцилиндр с односторонним штоком и датчиком координаты поршня</p>

	<p>Пневмогидравлический цилиндр двустороннего действия с двусторонним штоком</p>		<p>Пневмоцилиндр с односторонним штоком, датчиком координаты поршня и прецизионным стопором</p>
	<p>Пневмоцилиндр одностороннего действия толкающего типа с пружинным возвратом</p>		<p>Пневмоцилиндр двустороннего действия с пневмоуправляемым стопором</p>
	<p>Пневмоцилиндр одностороннего действия толкающего типа с пружинным возвратом и неповоротным штоком</p>		<p>Поворотный исполнительный механизм двустороннего действия</p>
	<p>Пневмоцилиндр одностороннего действия тянущего типа с пружинным возвратом</p>		<p>Регулируемый пневмодроссель с пневмоглушителем</p>

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Электронный каталог пневматического оборудования фирмы SMC.
2. МАЙЕР К.Ф. Контроль параметров систем автоматики цеха листорезных бумаг: учебное пособие / СЛПК - Сыктывкар, 2010. – 376 с.
3. Токмакова Л.И. Системы управления электроприводов. Учебное пособие. Часть 1 / Дальневосточный государственный технический университет им. В.В. Куйбышева. – Владивосток, 2002.
4. ГОСТ 19.701-90. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.
5. ГОСТ 19.002-80. Схемы алгоритмов и программ. Правила выполнения.
6. <http://ru.wikipedia.org/wiki/Блок-схема> (дата обращения: 01.10.2011).
7. ГОСТ 19.003-80. Схемы алгоритмов и программ. Обозначения условные графические.
8. Ремизова И.В., Новиков А.И. Программирование и наладка логического контроллера фирмы FESTO - Лабораторный практикум / СПбГТУРП. - СПб., 2009.
9. Материалы фирмы SMS, СПб, 2003г.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
1. ЭЛЕМЕНТЫ ПНЕВМОАВТОМАТИКИ	5
1.1. Пневматические цилиндры	5
1.2. Пневмораспределители	7
1.3. Пневмоглушитель	9
1.4. Конечные выключатели	9
1.5. Логический клапан «ИЛИ» (челночный клапан)	10
1.6. Логический клапан «И»	11
2. АЛГОРИТМЫ РАБОТЫ ОБОРУДОВАНИЯ	12
2.1. Циклограмма	16
2.2. Последовательность действий	17
2.3. Блок-схема	20
2.4. Написание программного кода по имеющейся блок-схеме	22
2.5. Вложения	28
2.6. Подпрограммы	32
2.7. Шаги	33
3. КОНТРОЛЛЕР	40
3.1. Входные и выходные переменные	40
3.2. Ввод/Вывод контроллера	41
3.3. Элементы памяти	42
3.4. Таймеры	57

4. ОПИСАНИЕ ПРОГРАММЫ SMC PNEUDRAW2.5 61

 4.1. Интерфейс программы 61

 4.2. Работа в программе 63

 4.3. Пример выполнения работы 64

Задания для выполнения лабораторной работы 72

Оформление отчета 73

Содержание курсового проекта (работы) 74

Примерный список тем экзамена 74

Приложение..... 75

БИБЛИОГРАФИЧЕСКИЙ СПИСОК 81

Учебное издание

Александр Игоревич Новиков
Елена Павловна Дятлова

Системы управления работой оборудования

Учебное пособие

Редактор и корректор Т.А.Смирнова

Техн. редактор Л.Я.Титова

Темплан 2012 г., поз.10

Подп. к печати 27.02.13. . Формат 60x84/16. Бумага тип. № 1.

Печать офсетная. 5,5 печ.л.; 5,5 уч.-изд. л.

Тираж 100 экз. Изд. № 10. Цена «С» Заказ

Ризограф Санкт-Петербургского государственного технологического университета растительных полимеров, 198095, СПб., ул. Ивана Черных, 4