

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт энергетики и электротехники

(наименование института полностью)

Кафедра « Промышленная электроника »

(наименование кафедры)

27.03.04 Управление в технических системах

(код и наименование направления подготовки, специальности)

Системы и технические средства автоматизации и управления

(направленность (профиль)/специализация)

БАКАЛАВРСКАЯ РАБОТА

на тему Автоматизированный лабораторный стенд на базе пневматического промышленного манипулятора МП-11

Студент

А.В. Маслова

(И.О. Фамилия)

(личная подпись)

Руководитель

Д.Г. Токарев

(И.О. Фамилия)

(личная подпись)

Консультант

Н.В. Яценко

(И.О. Фамилия)

(личная подпись)

Допустить к защите

Заведующий кафедрой

к.т.н., доцент, А.А. Шевцов

(ученая степень, звание, И.О. Фамилия)

(личная подпись)

« » 20 г.

Тольятти 2018

Аннотация

Объем 63 с., 34 рис., 5 табл., 20 источников, 1 прил.

АВТОМАТИЗИРОВАННЫЙ ЛАБОРАТОРНЫЙ СТЕНД НА БАЗЕ ПНЕВМАТИЧЕСКОГО ПРОМЫШЛЕННОГО МАНИПУЛЯТОРА МП-11

Цель работы – создание автоматизированного лабораторного стенда на базе пневматического промышленного манипулятора МП-11

Цель достигается при выполнении следующих:

1. Разработать алгоритмы управления лабораторным стендом;
2. Разработать систему управления манипулятором;
3. Разработать структурную и принципиальную схемы системы управления стендом.

Работа состоит из четырех глав, в которых решены выше представленные задачи.

В главе 1 «Промышленный пневматический манипулятор МП-11» приведены общие сведения о промышленном пневматическом манипуляторе МП-11, рассмотрена структура и работа основных частей манипулятора.

В главе 2 «Разработка алгоритма управления промышленным манипулятором» описаны четыре разработанных сценария работы промышленного манипулятора, представлены блок-схемы алгоритмов управления согласно сценариям.

В главе 3 «Разработка системы управления манипулятором» в данной главе представлен сравнительный анализ микроконтроллеров, описана структурная схема управления манипулятором, представлена программа управления данным стендом.

В главе 4 «Комплектация шкафа управления» описывается подробно каждый компонент системы управления промышленным пневматическим манипулятором МП-11.

Abstract

The title of the graduation work is The Automated Laboratory Stand on the Basis of the Pneumatic Industrial Manipulator MP-11.

The graduation work consists of an explanatory note on 63 pages, introduction, including 34 figures, 5 tables, the list of 20 references including 5 foreign sources and 1 appendices, and the graphic part on 7 A1 sheets.

The aim of the work is to develop an automated laboratory stand.

The object of the graduation work is the pneumatic industrial manipulator MP-11.

We give full coverage to the algorithm of the manipulator operation is described in detail, the development of the control system of the automated laboratory stand, the creation of the virtual manipulator model.

We start with the statement of the problem and then logically pass over to its possible solutions. We first discuss on what scenarios the manipulator will work, in order to demonstrate its functioning. The manipulator can function with either one working clasp or two. We then analyze the existing microcontrollers and choose the most appropriate for us.

After the analysis of the existing microcontrollers of the required level, the best option was chosen – the microcontroller Arduino, which has a wide application in the projects of electronic automation. It allows you to control external devices. Thus, this microcontroller combines everything necessary to develop a control system.

In conclusion we'd like to stress all objectives are carried out, the aim is achieved. The automated laboratory stand on the basis of the pneumatic manipulator was created. The control system of this laboratory stand which carries out the work based on four algorithms is developed.

Оглавление	
Введение.....	5
1 Промышленный пневматический манипулятор МП-11	6
1.1 Структура и работа основных частей манипулятора МП-11	8
1.2 Функциональная схема механизма подъема	9
1.3 Модуль поворота.....	11
1.4 Поступательный модуль или рука.....	12
1.5 Демпфер	13
1.6 Пневматический привод манипулятора МП-11	14
2 Разработка алгоритма управления промышленным манипулятором.....	16
3 Разработка системы управления манипулятором.....	29
3.1 Аппаратное обеспечение системы управления.....	30
3.2 Структурная схема системы управления.....	37
3.3 Принципиальная схема системы управления.....	38
3.4 Программное обеспечение системы управления.....	39
4 Комплектация шкафа управления	50
4.1 Микроконтроллер Arduino Mega 2560	50
4.2 Arduino Sensor Shield	51
4.3 Реле 8 relay module	53
4.4 Реле 4 relay module	54
4.5 Матричная клавиатура 4x4.....	56
4.6 Дисплей LCD	57
Заключение	61
Список используемой литературы	62
Приложение А.....	64

Введение

Промышленные роботы широко используются на производстве, а именно в отрасли машиностроения. Так как в наше время производство все больше автоматизируется и с целью его усовершенствования используют промышленных роботов.

Промышленный робот – автоматическое устройство, которое состоит из манипулятора и перепрограммируемого устройства управления, применяющиеся для перемещения объектов в пространстве в различных производствах. Он является частью роботизированного технологического комплекса. Применение промышленных роботов значительно упрощает процесс производства.

Промышленные роботы заменяют человека там, где требуется тяжелый, монотонный труд. В условиях с повышенной влажностью и с повышенной температурой, вибрацией, шумом, взрывоопасностью. Промышленный робот представляющий собой перепрограммируемую автономную машину, способен выполнять аналогичные человеческим двигательные функции. Например, такие как перемещение предметов производства или технологической оснастки.

Роботы способствуют замене монотонного физического труда, повышению качества изделий, увеличению их выпуска.

Использование промышленных роботов в качестве лабораторных стендов в учебном процессе, позволяет студентам закрепить теоретические знания полученные в процессе изучения лекционного материала путем выполнения лабораторных и практических работ.

1 Промышленный пневматический манипулятор МП-11

Пневматический промышленный манипулятор МП-11 предназначен для выполнения и ориентирования деталей в автоматизированном технологическом процессе сборки, штамповки и другое.

Из шести степеней подвижности манипулятора четыре обеспечивают реализацию транспортирующих (региональных) движений, а две – ориентирующих (локальных). Для перемещения объекта манипулирования в требуемую позицию рабочей зоны достаточно только три транспортирующих степени подвижности, поэтому еще три степени подвижности являются избыточными. Их использование позволяет решить функциональные возможности робота. К основным степеням подвижности данного манипулятора следует отнести три транспортирующих:

- подъем рук;
- поворот рук;
- выдвижение одной из рук.

Избыточными степенями подвижности следует считать транспортирующее движение выдвижения второй руки (наличие которой определено конструктивным исполнением) и ориентирующие движения:

- ротация руки;
- сдвиг схвата.

Кроме того, следует сказать, что манипулятор оснащен пневматическим приводом, система управления обеспечивает программно – логический способ управления, обеспечивая ручной ввод программы, и манипулятор оснащен двумя механическими универсальными схватами клещевого типа.

Основные параметры технической характеристики промышленного пневматического робота МП-11 приведены в таблице 1.

Таблица 1 – Основные технические характеристики робота МП-11

Параметр	Значение
Номинальная грузоподъемность	2.0 кг
Полезная грузоподъемность	1.0 кг
Максимальная величина горизонтального перемещения	200 мм
Максимальный угол поворота в горизонтальной плоскости	120°
Максимальная величина вертикального перемещения (подъема) рук	65 мм
Максимальная величина сдвига схвата	25 мм
Максимальный угол поворота схвата вокруг продольной оси (ротация)	180°
Максимальный угол между руками	100°
Минимальный угол между руками	20°
Габаритные размеры: - длина - ширина - высота	990 мм 990 мм 445 мм
Рабочее давление пневмопитания	0.4...0.5 МПа
Напряжение управления электропневмо-распределителями - род тока - сила тока	$24_{-3.6}^{+2.4}$ В постоянный не более 0.5 А
Масса манипулятора	70 кг

Компоновка манипулятора реализована таким образом, что в стационарном каркасе (корпусе) смонтирован механизм подъема рук. Снизу к модулю механизма подъема крепится корпус модуля механизма поворота рук, поднимающийся вместе с ними. Сверху на модуле подъема один над другим (с углом между ними горизонтальной плоскости) установлены два поступательных модуля горизонтальных перемещений (выдвижение рук). На стержни этих модулей смонтированы модуль вращения (ротации) и модуль поступательный (сдвига) схвата, к которым непосредственно и крепятся схваты. Расположение других элементов конструкции узлов показано на рисунке 1.

1.1 Структура и работа основных частей манипулятора МП-11

Механизм подъем предназначен для перемещения рук манипулятора вдоль вертикальной оси Z. Модуль механизма подъема 1, в соответствии с рисунком 1, крепится к корпусу 2 манипулятора восьмью винтами 3. Так же в корпусе закреплена направляющая цилиндрической формы 4, на которой установлены упор 5, ограничивающие величину перемещения. На этом же рисунке показана установка в корпусе демпферов и механизма поворота.

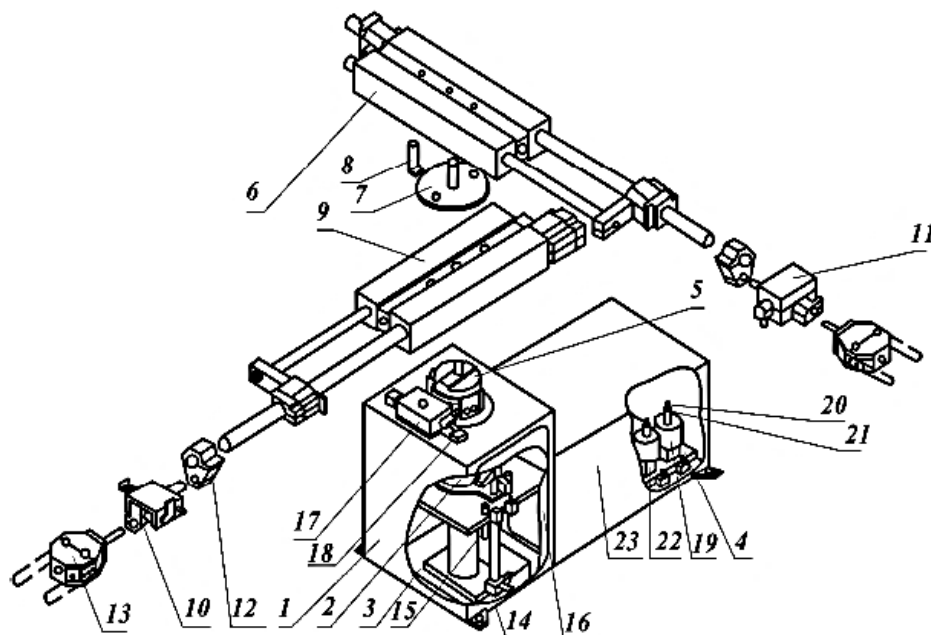


Рисунок 1 – Схема расположения основных узлов и элементов пневматического промышленного манипулятора МП-11

1 – корпус манипулятора; 2 – механизм подъема; 3 – механизм поворота; 4 – распределительная плита; 5 – муфта зубчатая; 6,9 – модуль поступательный (рука); 7 – фланец крепления рук; 8 – колонка фиксации рук относительно фланца; 10 – модуль поступательный сдвига схвата; 11 – модуль поворота (ротации) схвата; 12 – кронштейн; 13 – схват; 14 – упор; 15 – демпфер; 16 – крышка корпуса, 17 – амортизатор механизма поворота, 18 – контакт магнитоуправляемый КЭМ; 19 – электромагнитный регулятор (дроссель); 20 – игла; 21 – гайка фиксации иглы; 22 – пневмодроссель; 23 – корпус устройства распределения воздуха.

1.2 Функциональная схема механизма подъема

На рисунке 2 представлена схема функционирования модуля механизма подъема. Данный модуль состоит из корпуса (цилиндра) 1, полого штока – поршня 2 с уплотнительными манжетами

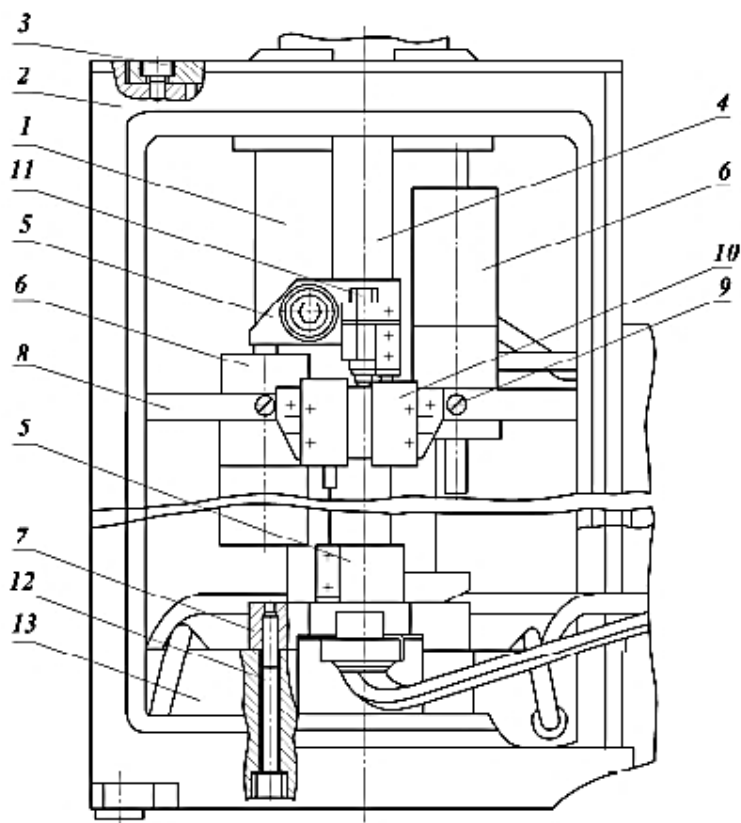


Рисунок 2 –Механизм подъема

1 – корпус модуля подъема пневмоцилиндра; 2 – корпус манипулятора; 3 – крепежные винты; 4 – цилиндрическая направляющая; 5 – упор; 6 – демпфер; 7 – пластина; 8 – перегородка корпуса; 9 – винт; 10 – блок контактов электромагнитных (КЕМ); 11 – винт тонкой регулировки положения упора; 12 – винт; 13 – корпус модуля поворота.

Представленный на рисунке 2 механизм, обеспечивает подъём и опускание рук манипулятора.

В корпусе данного механизма установлена цилиндрическая направляющая 4, к которой крепятся упоры 5. Демпферы 6 вворачиваются в пластину 8 корпуса 2 и стопорятся винтами 9.

Для регулировки механизма подъема устанавливается величина вертикального перемещения рук манипулятора и проверяется срабатывание блока переключателей 10, для чего:

- подберем положение микровинта 11, при котором надежно срабатывает переключатель 10;
- передвинем упоры 5 на необходимую величину, первоначально ослабив их затяжку к направляющей 4;
- заждем упоры 5.

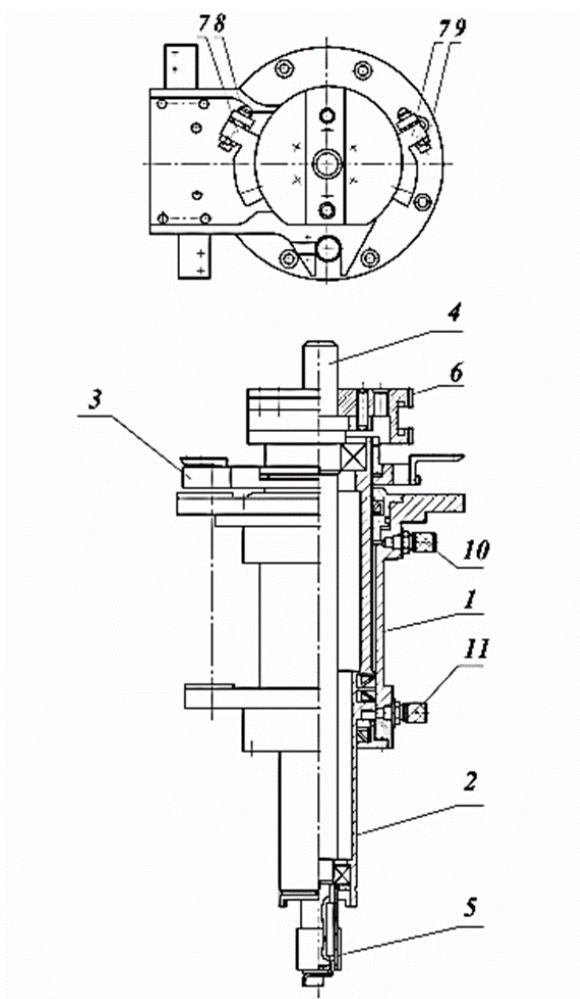


Рисунок 3 – Модуль подъема

1 – корпус пневмоцилиндра; 2 – шток-поршень; 3 – кронштейн; 4 – вал механизма поворота; 5 – колесозубчатое; 6 – муфта зубчатая; 7 – упоры; 8,9 – винты регулировки положений упоров; 10,11 – штуцер

При подаче сжатого воздуха через штуцер 10, изображенного на рисунке 3, в нижнюю полость пневмоцилиндра, происходит перемещение шток-поршня,

разделяющего полость цилиндра на две герметично изолированные друг от друга части, вверх подъем закрепленных на нем модулей и узлов, подача сжатого воздуха через верхний штуцер приводит к опусканию шток-поршня. Изготовление штоковых полостей пневмоцилиндра с различным сечением обусловлено соображениями улучшения динамических показателей работы модуля.

Для установки и фиксации на данном модуле механизма поворота, нижний конец шток-поршня выполняется с проточкой и выступом. На верхнем конце шток-поршня устанавливается кронштейн 3 для закрепления амортизатора механизма поворота.

Внутри полого шток-поршня закреплены опоры качения (подшипники) для установки в них вала 4 механизма поворота. На нижнем конце вала закреплено зубчатое колесо 5, взаимодействующее со шток-рейками механизма поворота. На верхнем конце вала закреплена зубчатая муфта 6 для крепления на ней и регулировки положений упоров 7, определяющих угол поворота рук манипулятора.

1.3 Модуль поворота

На рисунке 4 изображен модуль поворота, предназначенный для поворота рук манипулятора вокруг вертикальной оси. Модуль образован корпусом 1, в котором выполнены полости двух пневмоцилиндров. В пневмоцилиндрах установлены шток-рейки 2, в зацеплении с которыми прямозубое цилиндрическое колесо, закрепленное на нижнем конце вала, изображенного на рисунке 3.

Через окно в днище корпуса манипулятора устанавливается модуль поворота, который с помощью пластины 7, как показана на рисунке 2, крепится винтами 12 к штоку модуля подъема. Цилиндрическая направляющая 4 одним концом закреплена в пластине 7, а вторым – в кронштейне 3, изображенном на рисунке 3. В результате она вместе со штоком модуля подъема и корпусом механизма поворота образует жесткую замкнутую конструкцию. Таким образом, предотвращается поворот штока модуля подъема. Демпферы 6 вворачиваются в горизонтальную перегородку 8 корпуса 2 и стопорятся винтами 9.

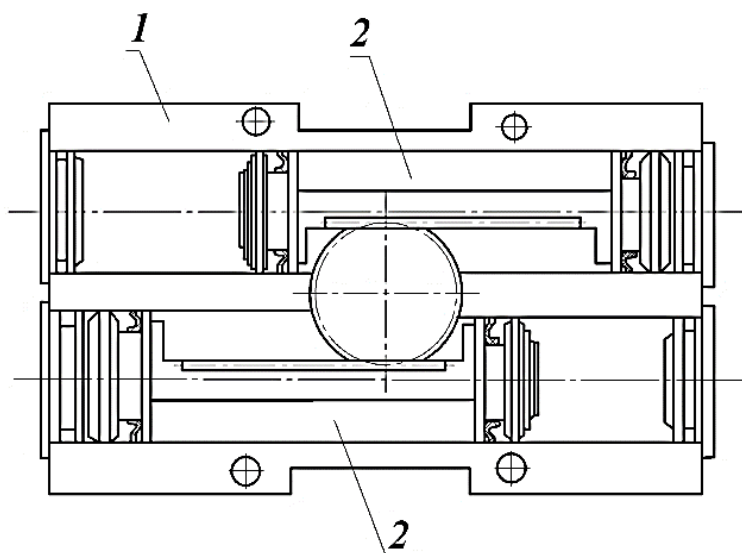


Рисунок 4 – Структура модуля поворота

1.4 Поступательный модуль или рука

Модуль поступательный или рука (рисунок 5), предназначена для горизонтального перемещения (выдвижения) схвата в процессе манипулирования объектом. Модуль образован корпусом 1, в котором расположен пневмоцилиндр двустороннего действия. Параллельно пневмоцилиндру в корпусе подвижно закреплен стержень 3. На стержне с двух сторон устанавливаются основные и регулировочные упоры 4 и 5, соответственно. Упоры, установленные со стороны схвата, соединены со штоком – поршнем. Все перемещаемые этим модулем объекты устанавливаются на стержне.

Полость цилиндра разделена штоком – поршнем 2 на две герметичные части. Для подачи сжатого воздуха в полости используют шток – поршень, который по этой причине выполнен полым. При подаче через один из штуцеров 8 или 9 сжатого воздуха выполняется перемещение штока – поршня 2 вместе со стержнем 3 и упорами 4,5. Вблизи конечного положения упор 4 нажимает на плунжер демпфера 6, что приводит к уменьшению скорости перемещения, обеспечивая плавность выхода в точку позиционирования. В конце хода срабатывает электромагнитный контакт 7, выдавая сигнал о завершении выполнения движения.

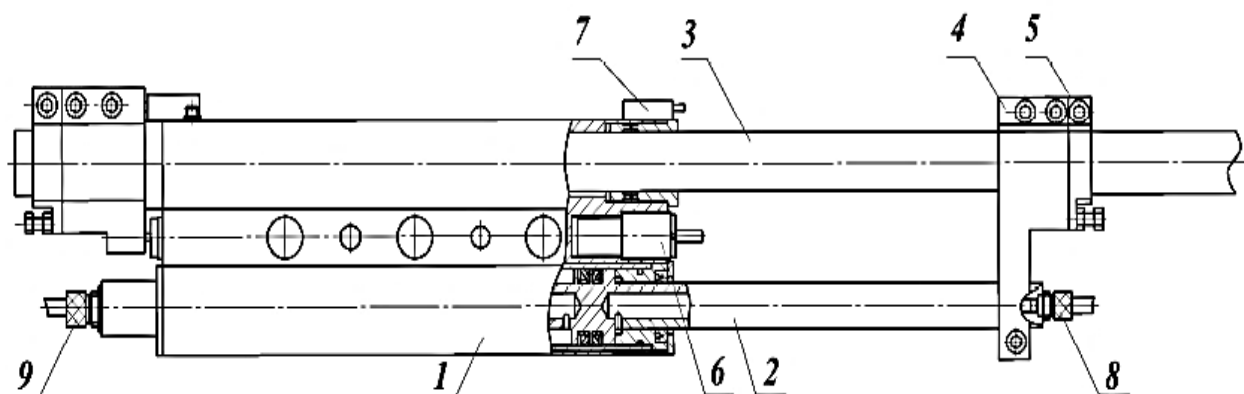


Рисунок 5 – Модуль поступательный (рука)

1 – корпус; 2 – шток-поршень; 3 – стержень; 4 – упор основной; 5 – упор регулировочный; 6 – демпфер; 7 – КЭМ (контакт электромагнитный); 8,9 – штуцер.

1.5 Демпфер

Демпфер, структура которого показана на рисунке 6, обеспечивает мягкий удар при выходе на упор, способствуя плавному выходу подвижного звена в точку позиционирования. Демпферы такой конструкции установлены на модулях горизонтальных перемещений (руках) и в механизме подъема.

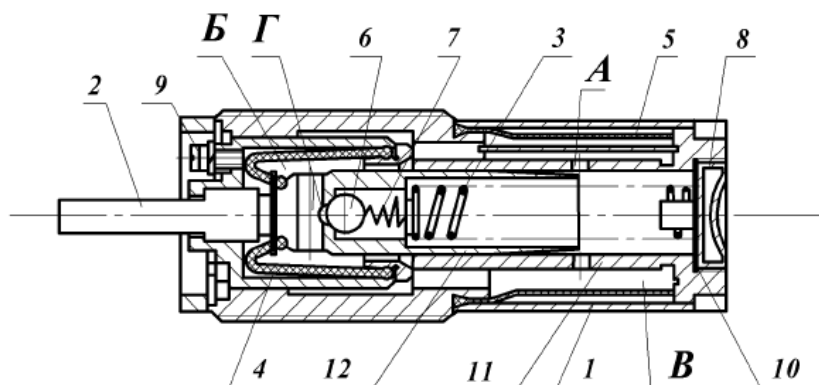


Рисунок 6 – Структура демпфера

1 – корпус; 2 – толкатель; 3 – пружина; 4 – колпачок; 5 – кожух; 6 – шарик; 7 – пружина; 8 – пробка; 9 – винт; 10 – прокладка; 11 – неподвижная (цилиндрическая) часть плунжера; 12 – подвижная (коническая) часть плунжера; А,Г – отверстия; Б,В – герметичные полости.

Демпфер состоит из корпуса 1, в котором установлен плунжер, поджимаемый пружиной 3 и имеющий четыре лыски с разными углами наклона, колпачок 4, кожух 5, герметично соединены плунжером и корпусом, образуя, две герметичных плоскости Б и В.

Так же в плунжере расположен шарик 6, поддерживаемый пружиной 7, обеспечивающий запираение перепускного отверстия Г.

При воздействии на толкатель 2 движение передается подвижной конической части плунжера 12, которая перемещаясь внутрь неподвижной цилиндрической части плунжера преодолевая силу упругости пружины, сокращает объем полый части плунжера, в которой находится масло.

1.6 Пневматический привод манипулятора МП-11

Так как грузоподъемность манипулятора невелика, а система управления робота цикловая, то перемещения его элементов используется пневматический привод. Структура пневмопривода образована узлом подготовки воздуха, изображенного на рисунке 7, узлом распределения воздуха, расположенным с тыльной стороны корпуса манипулятора и, закрываемого крышкой 23, как показано на рисунке 1, и соединительной арматурой (штуцеры, шланги и т.д.).

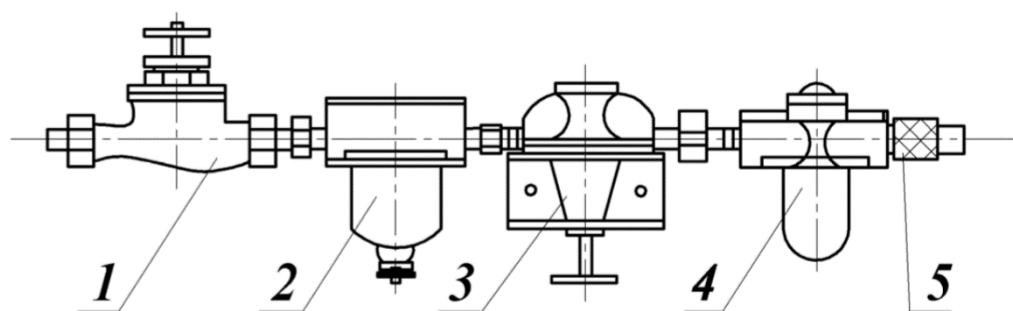


Рисунок 7 – узел подготовки воздуха (УПВ)

Узел подготовки воздуха предназначен для регулирования давления сжатого воздуха, подаваемого в манипулятор, очистки воздуха от твердых частиц и воды и распыления масла для смазки трущихся частей пневмоцилиндров манипулятора. Осушитель УВП обеспечивает влажность используемого сжатого воздуха, не превышающую 80%. Очиститель удаляет все механические включения с размером в поперечном сечении более 80 мкм. Устройство смазки распыляет одну – две капли масла в минуту.

Узел распределения воздуха образован двенадцатью пневмоцилиндрами, позиция 22 рисунка 1 (обозначенные на рисунке 8 символами «К1», «К2», ..., «К12»), установленными на распределительной плите. На них установлены

электромагнитные регуляторы (дрессели), позиция 19 на рисунке 1, обозначенные на рисунке 8 символами «ДР1», «ДР2», ..., «ДР12». Подключены они согласно пневматической принципиальной схеме, приведенной на рисунке 8. Электропневмораспределители реализуют срабатывание какой – либо степени подвижности, путем подсоединения шлангов, соединяющих соответствующую полость пневмоцилиндра с нагнетающей магистралью через каналы, выполненные в распределительной плите. Кроме того, они позволяют с помощью дресселей, изменяя вертикальные положения игл, позиция 20 рисунка 1, регулировать скорости срабатывания по каждой из степеней подвижности (отдельно в каждом направлении)

Принципиальная пневматическая схема работа приведена на рисунке 8.

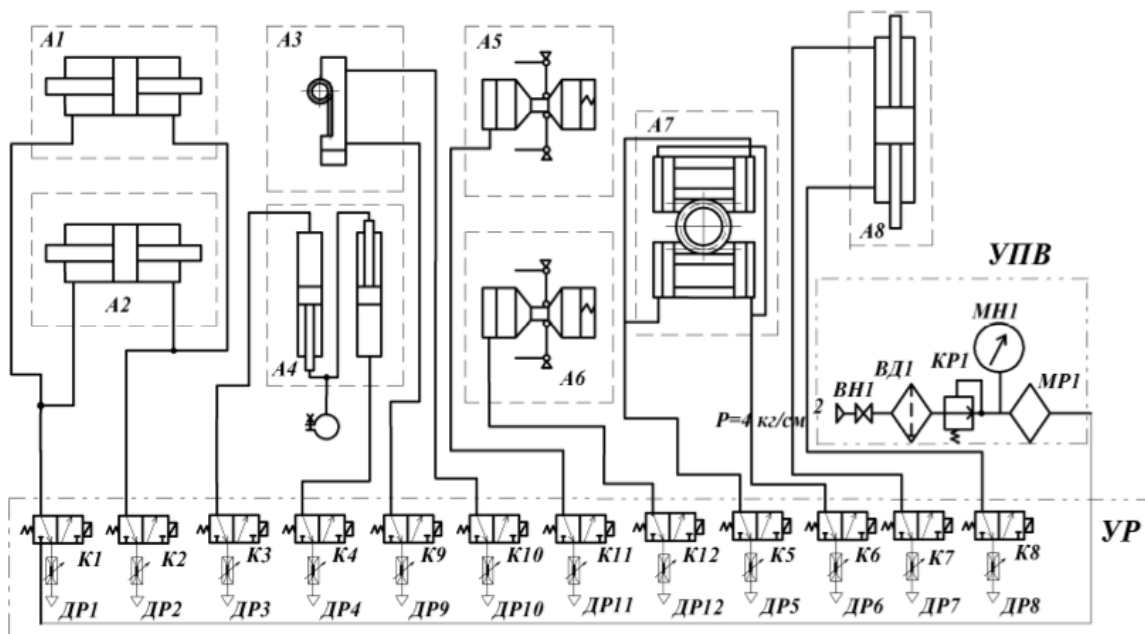


Рисунок 8 – Принципиальная пневматическая схема пневматического промышленного манипулятора МП – 11

A1, A2 – модули поступательные (руки); A3 – модуль вращательный (ротация схвата); A4 – модуль поступательный (сдвиг схвата); A5, A6 – схват; A7 – механизм поворота; A8 – механизм подъема; УР – узел распределения; ДР1, ДР2,...ДР12 – дрессели; К1, К2,...К12 – пневмораспределители; УВП – узел подготовки воздуха; ВН1 – вентиляльный запорный, ВД1 – фильтр – влагоотделитель, КР1 – пневмоклапан, МН1 – манометр, МР1 – маслораспылитель.

2 Разработка алгоритма управления промышленным манипулятором

Для проведения лабораторных работ было разработано четыре алгоритма функционирования промышленного пневматического манипулятора. Каждый алгоритм представляет собой определенную последовательность работы звеньев манипулятора.

Алгоритм 1 (рабочая нижняя рука)

Руки вперед → подъем манипулятора → разжатие схвата → руки вправо → зажим схвата → руки влево → опускание робота → ротация схвата → руки назад → разжим схвата

Описание алгоритма управления манипулятором МП-11 по первому сценарию

1. При нажатии кнопки «Пуск» подается питание на плату контроллера.
2. Включается перемещение манипулятора в горизонтальной плоскости.

При открытии электропневмоклапана воздух подается в пневмоцилиндр до тех пор, пока упор не включит концевой выключатель. Происходит переход к следующему шагу программы. Если сигнал включения отсутствует, то происходит аварийный останов программы. Указанные условия перехода к следующим шагам алгоритма аналогичны.

3. Далее действия выполняются в следующем порядке:
 - 1) включается перемещение манипулятора в вертикальной плоскости;
 - 2) включается разжатие схвата;
 - 3) включается перемещение манипулятора против часовой стрелки;
 - 4) включается зажим схвата;
 - 5) включается перемещение манипулятора по часовой стрелке;
 - 6) включается перемещение манипулятора в вертикальной плоскости;
 - 7) включается ротация схвата;
 - 8) включается перемещение манипулятора в горизонтальной плоскости;
 - 9) включается разжатие схвата.

При выполнении последнего шага программы происходит переход к началу программы.

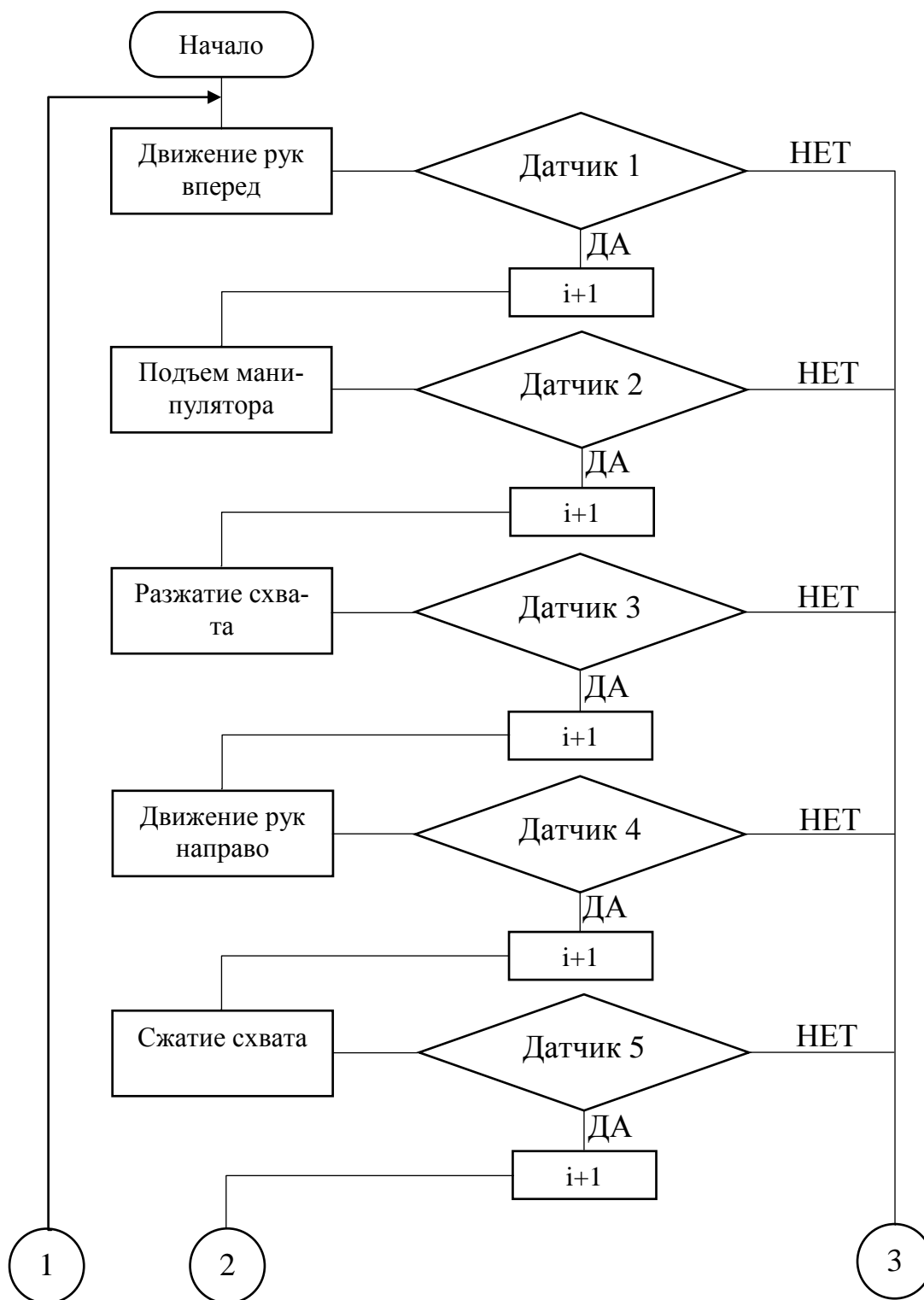


Рисунок 9 – Блок схема алгоритма 1

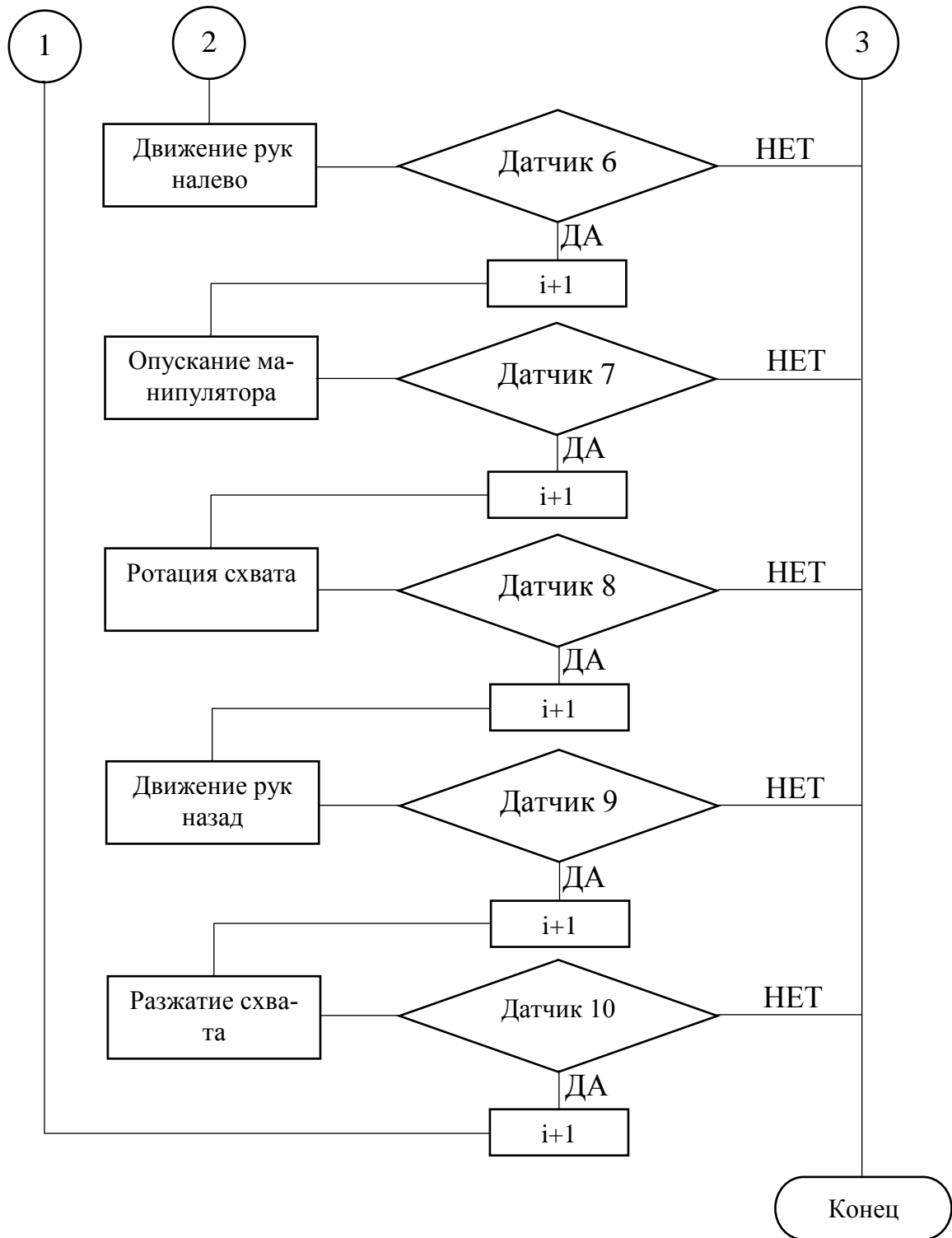


Рисунок 10 – Блок-схема алгоритма 1 (продолжение)

Алгоритм 2 (рабочая верхняя рука)

Разжатие схвата → руки направо → сжатие схвата → подъем → руки вперед → руки налево → движение схвата вправо → разжатие схвата → руки назад → опускание манипулятора

Описание алгоритма управления манипулятором МП-11 по второму сценарию

1. При нажатии кнопки «Пуск» подается питание на плату контроллера.

2. Включается разжатие схвата.

При открытии электропневмоклапана воздух подается в пневмоцилиндр до тех пор, пока упор не включит концевой выключатель. Происходит переход к следующему шагу программы. Если сигнал включения отсутствует, то происходит аварийный останов программы. Указанные условия перехода к следующим шагам алгоритма аналогичны.

3. Далее действия выполняются в следующем порядке:

- 1) включается перемещение манипулятора против часовой стрелки;
- 2) включается сжатие схвата;
- 3) включается перемещение манипулятора в вертикальной плоскости;
- 4) включается перемещение манипулятора в горизонтальной плоскости;
- 5) включается перемещение манипулятора по часовой стрелке;
- 6) включается движение схвата;
- 7) включается разжатие схвата;
- 8) включается перемещение манипулятора в горизонтальной плоскости;
- 9) включается перемещение манипулятора в вертикальной плоскости;

При выполнении последнего шага программы происходит переход к началу программы.

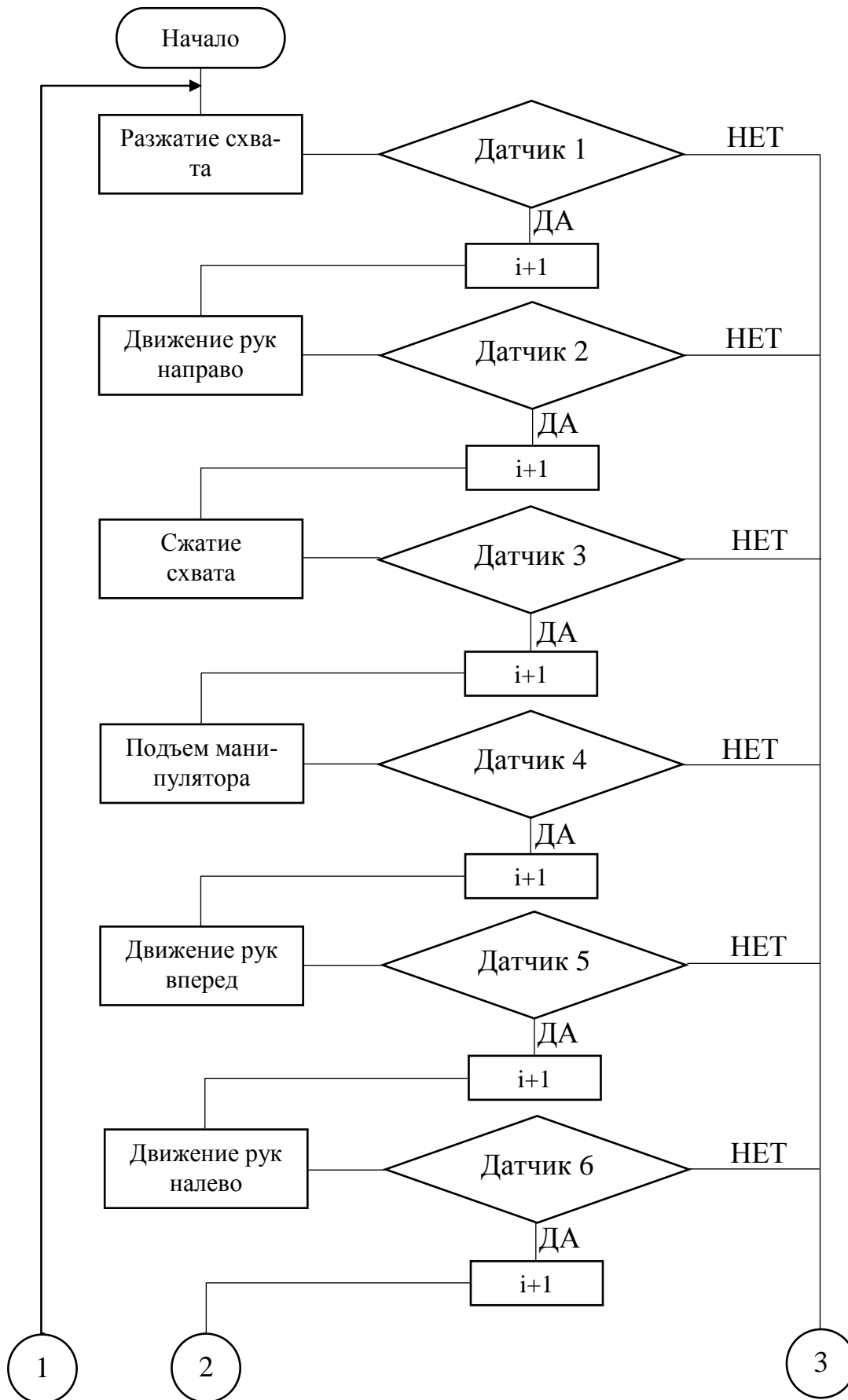


Рисунок 11 – Блок-схема алгоритма 2

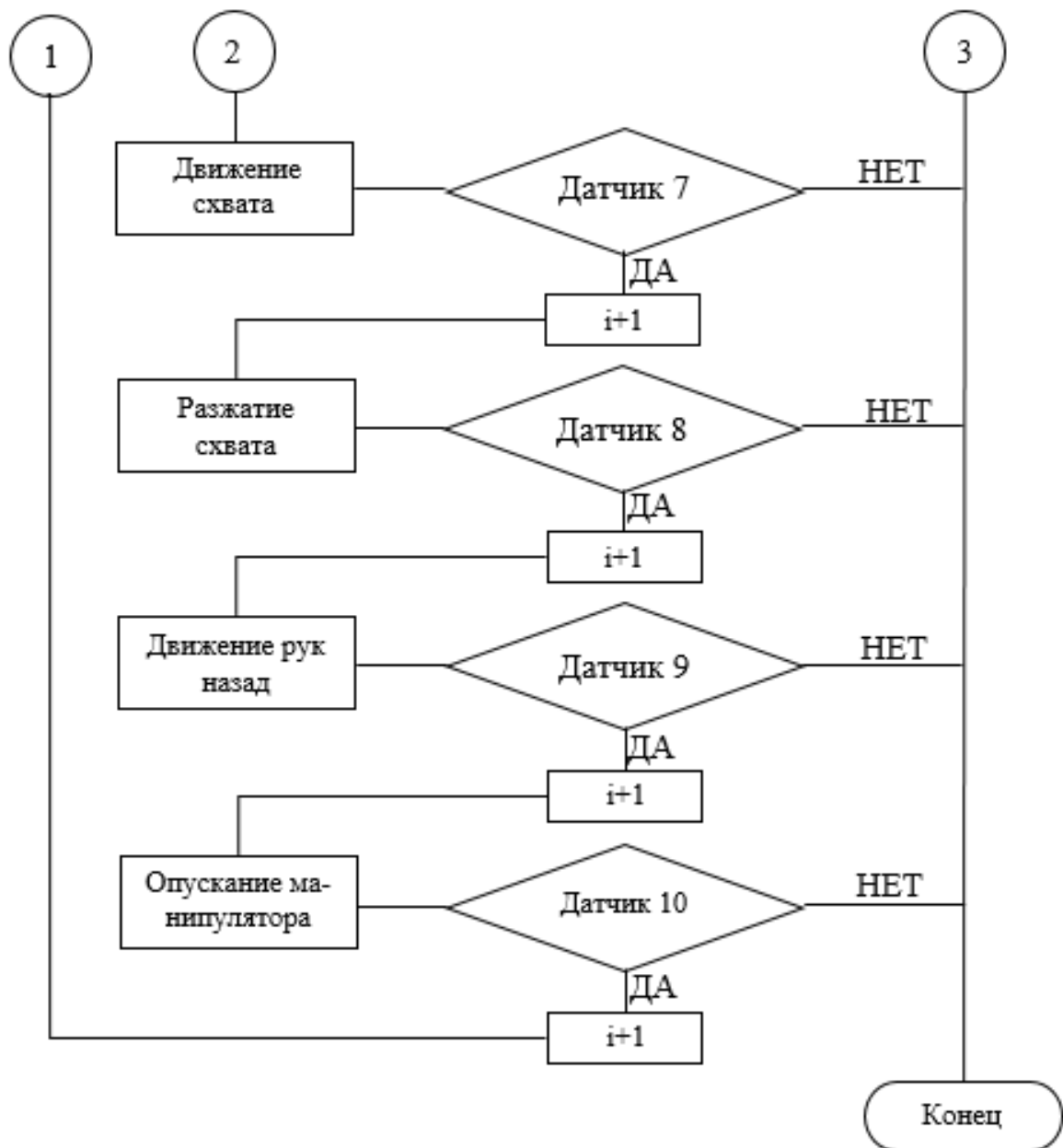


Рисунок 12 – Блок-схема алгоритма 2 (продолжение)

Алгоритм 3 (обе руки манипулятора рабочие)

Разжатие схватов → руки налево → подъем манипулятора → руки вперед → зажим нижнего схвата → руки направо → зажим верхнего схвата → руки назад → ротация схвата вправо → движение схвата направо → опускание манипулятора → руки налево → руки вперед → разжатие схватов → руки назад

Описание алгоритма управления манипулятором МП-11 по третьему сценарию

1. При нажатии кнопки «Пуск» подается питание на плату контроллера.

2. Включается разжатие схватов.

При открытии электропневмоклапана воздух подается в пневмоцилиндр до тех пор, пока упор не включит концевой выключатель. Происходит переход к следующему шагу программы. Если сигнал включения отсутствует, то происходит аварийный останов программы. Указанные условия перехода к следующим шагам алгоритма аналогичны.

3. Далее действия выполняются в следующем порядке:

- 1) включается перемещение манипулятора по часовой стрелке;
- 2) включается перемещение манипулятора в вертикальной плоскости;
- 3) включается перемещение манипулятора в горизонтальной плоскости;
- 4) включается зажим нижнего схвата;
- 5) включается перемещение манипулятора против часовой стрелки;
- 6) включается зажим верхнего схвата;
- 7) включается перемещение манипулятора в горизонтальной плоскости;
- 8) включается ротация схвата;
- 9) включается движение схвата;
- 10) включается перемещение манипулятора в вертикальной плоскости;
- 11) включается перемещение манипулятора по часовой стрелке;
- 12) включается перемещение манипулятора в горизонтальной плоскости;
- 13) включается разжатие схватов;
- 14) включается перемещение манипулятора в горизонтальной плоскости;

При выполнении последнего шага программы происходит переход к началу программы.

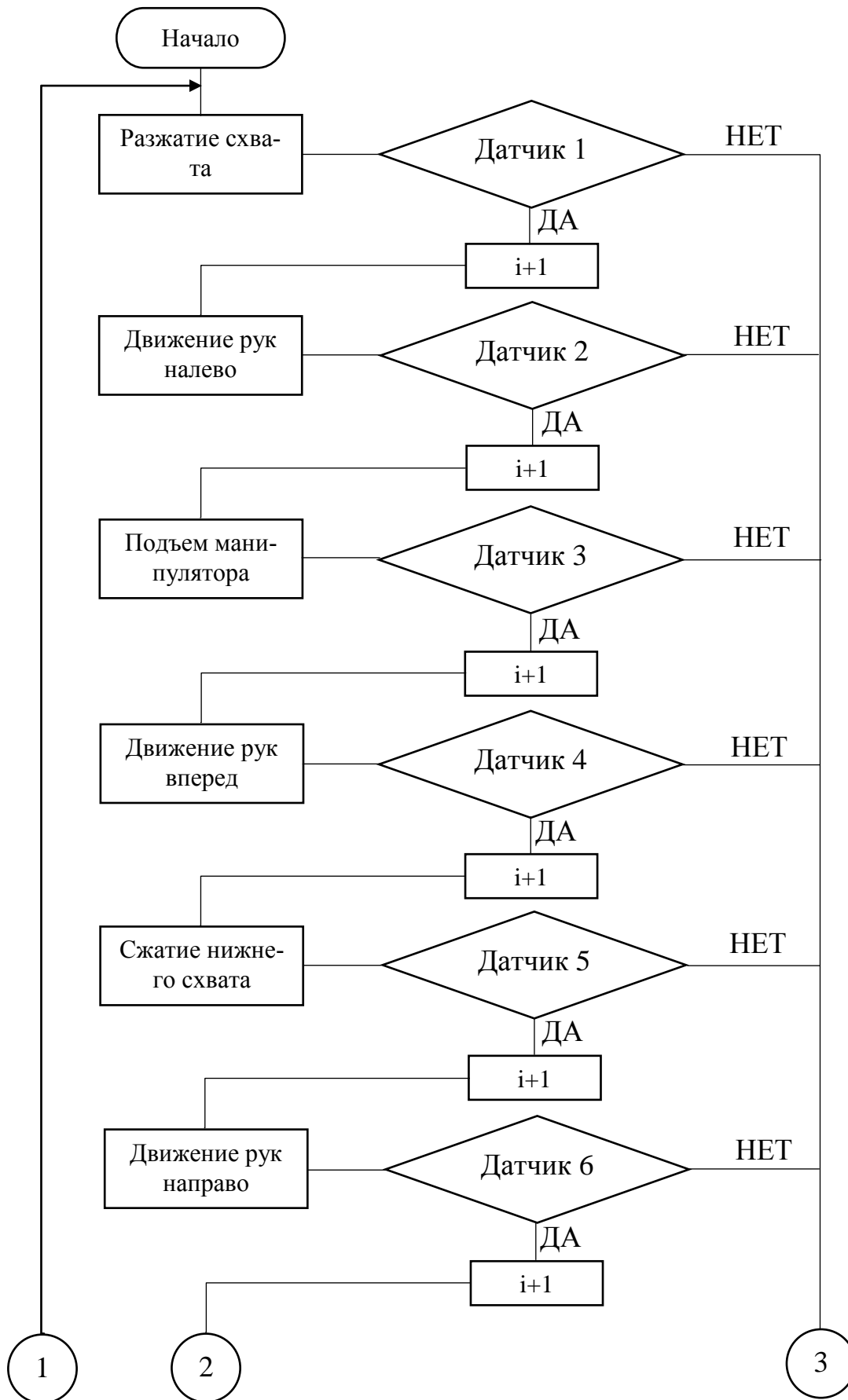


Рисунок 13 – Блок-схема алгоритма 3

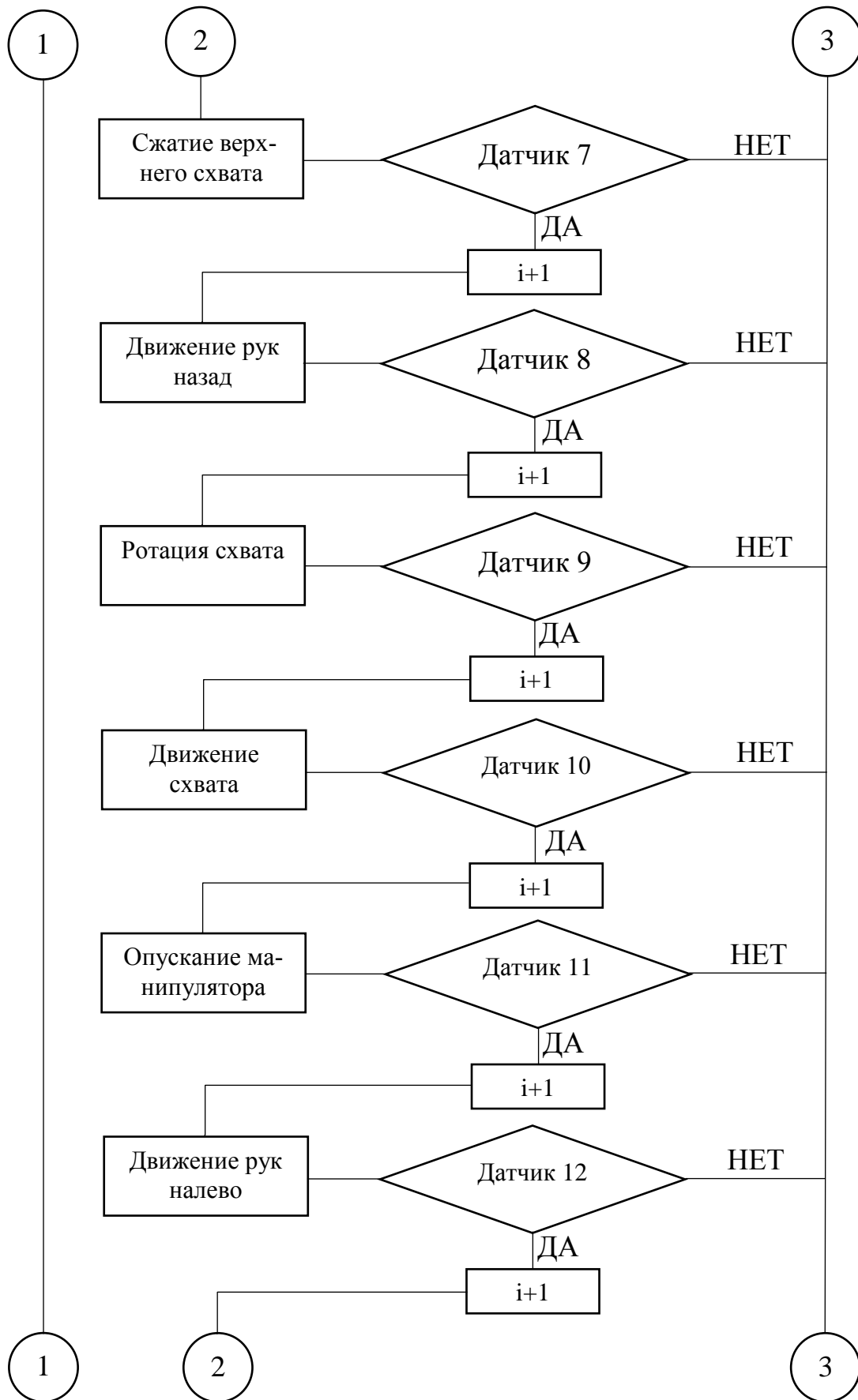


Рисунок 14 – Блок-схема алгоритма 3 (продолжение)

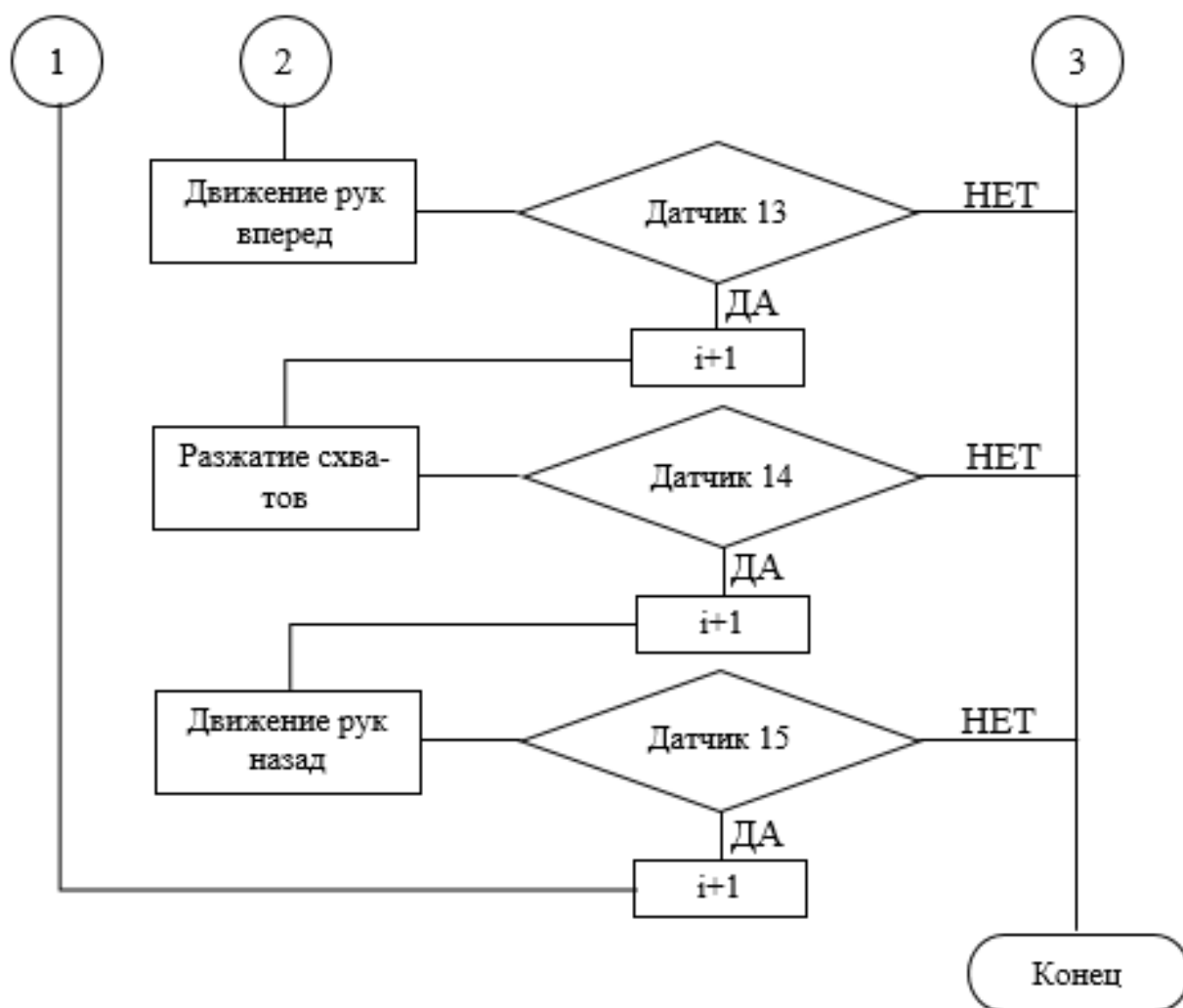


Рисунок 15 – Блок-схема алгоритма 3 (продолжение)

Алгоритм 4 (обе руки манипулятора рабочие)

Подъем манипулятора → разжатие схватов → руки направо → руки вперед → зажим схватов → руки назад → руки налево → опускание манипулятора → руки вперед → ротация схвата вправо → движение схвата налево → разжатие схватов → руки назад

Описание алгоритма управления манипулятором МП-11 по четвертому сценарию

1. При нажатии кнопки «Пуск» подается питание на плату контроллера.
2. Включается перемещение манипулятора в вертикальной плоскости.

При открытии электропневмоклапана воздух подается в пневмоцилиндр до тех пор, пока упор не включит концевой выключатель. Происходит переход

к следующему шагу программы. Если сигнал включения отсутствует, то происходит аварийный останов программы. Указанные условия перехода к следующим шагам алгоритма аналогичны.

3. Далее действия выполняются в следующем порядке:

- 1) включается разжатие схватов;
- 2) включается перемещение манипулятора против часовой стрелки;
- 3) включается перемещение манипулятора в горизонтальной плоскости;
- 4) включается зажим схватов;
- 5) включается перемещение манипулятора в горизонтальной плоскости;
- 6) включается перемещение манипулятора по часовой стрелке;
- 7) включается перемещение манипулятора в вертикальной плоскости;
- 8) включается перемещение манипулятора в горизонтальной плоскости;
- 9) включается ротация схвата;
- 10) включается движение схвата;
- 11) включается разжатие схватов;
- 12) включается перемещение манипулятора в горизонтальной плоскости.

При выполнении последнего шага программы происходит переход к началу программы.

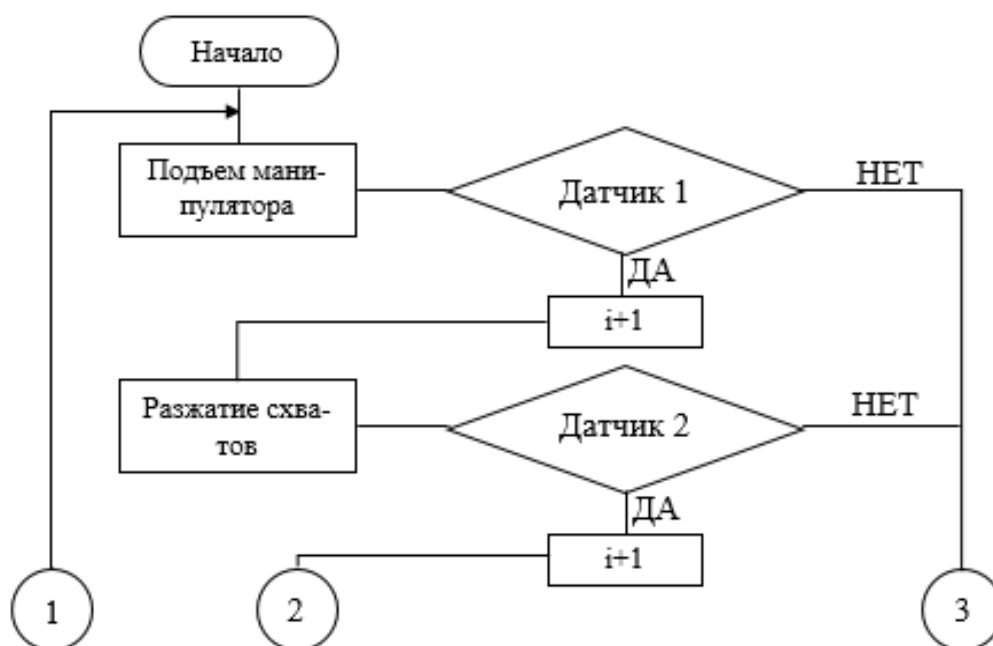


Рисунок 16 – Блок-схема алгоритма 4

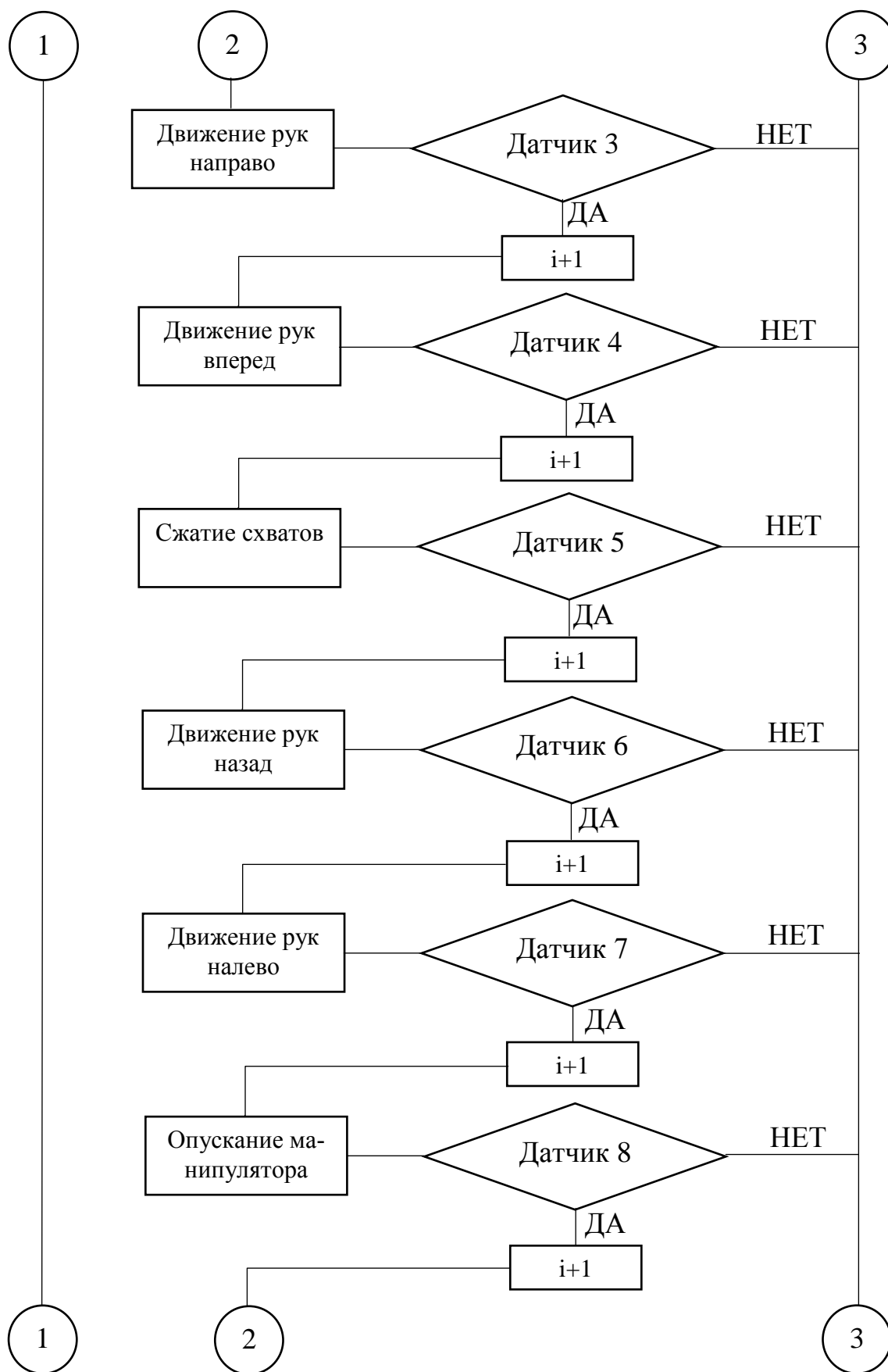


Рисунок 17 – Блок-схема алгоритма 4 (продолжение)

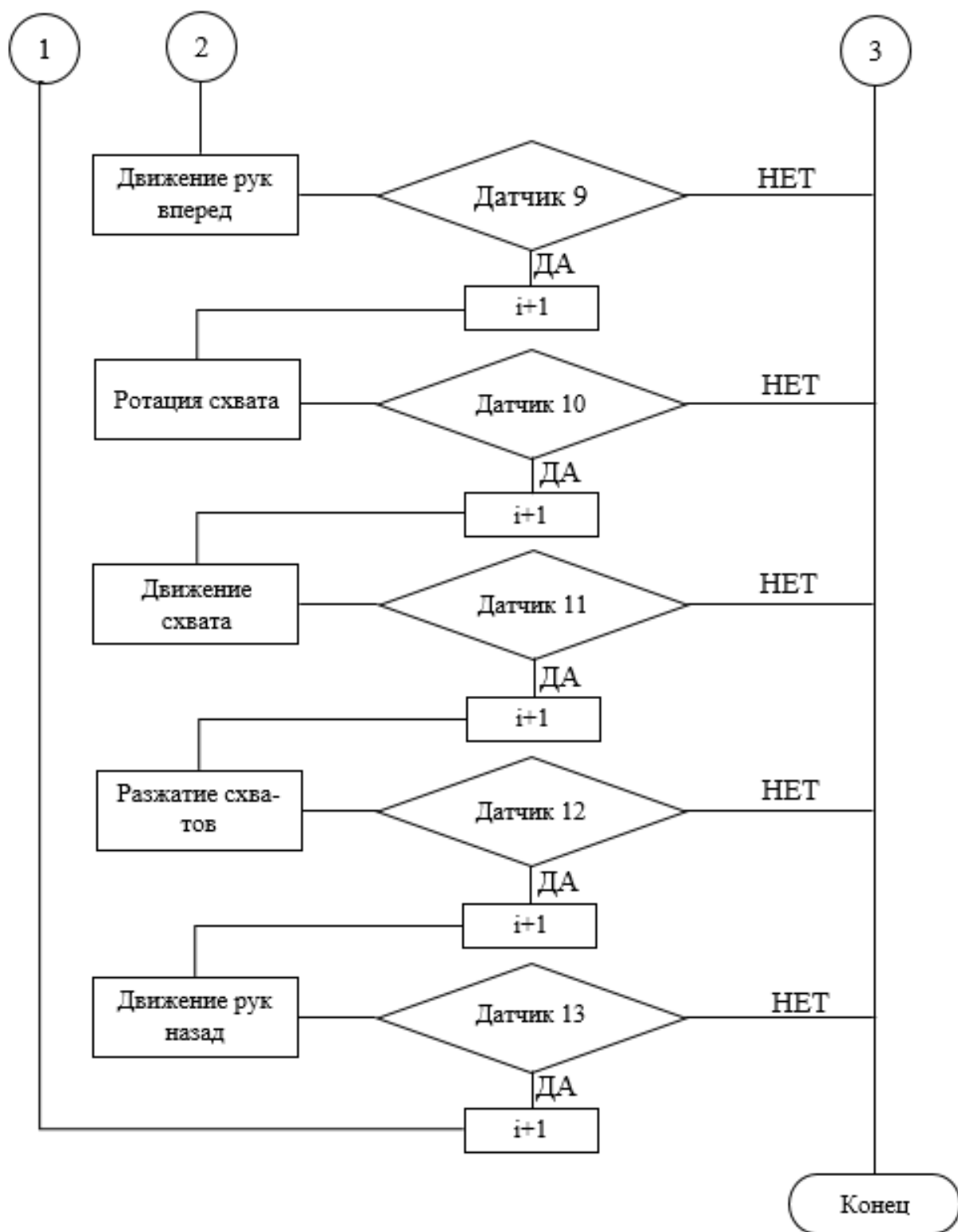


Рисунок 18 – Блок-схема алгоритма 4 (продолжение)

3 Разработка системы управления манипулятором

Ранее управление промышленным пневматическим манипулятором МП-11 осуществлялось с помощью устройства управления ЭЦПУ – 6030, предназначенного для управления манипуляторами с позиционированием по упорам.

Функциональный состав устройства и характерные связи между блоками изображен на рисунке 19. Данное устройство построено по принципу синхронного программного автомата с жестким циклом управления. Основные блоки и узлы устройства управления:

- пульт управления, задает режимы работы устройства, и осуществляет ручное управление звеньями манипулятора;
- блок управления, обрабатывает информацию по заданному режиму и выдает управляющие воздействия на манипулятор;
- блок усилителей, выдает управляющие команды на клапаны манипулятора;
- блок питания, осуществляет питание системы управления и электропневмоклапанов;
- программноноситель, предназначен для набора и хранения требуемой программы работы манипулятора.



Рисунок 19 – ЭЦПУ-6030 схема структурная

С пульта управления задается один из следующих режимов работы устройства:

- ручной;
- команда;
- цикл;
- автомат.

Режим работы «ручной»

Подача команд осуществляется с пульта управления, команды поступают на блок усилителей, а затем на манипулятор для управления его подвижными органами. Контроль положения исполнительных органов манипулятора осуществляется с помощью табло индикации состояния звеньев манипулятора.

Режим работы «команда»

Устройство обеспечивает обработку одного кадра программы, набранной на программноносителе. После обработки команд, заданных в кадре, происходит остановка устройства.

Режим работы «цикл»

Устройство осуществляет однократную обработку всех кадров программы.

Режим работы «автомат»

Устройство осуществляет многократную обработку рабочего цикла робота.

Штатное устройство управления неисправно, его восстановление в условиях развития автоматизации нерационально, поэтому в результате анализа принято решение разработать систему управления на базе одного из современных микроконтроллеров.

3.1 Аппаратное обеспечение системы управления

Рассмотрим два семейства контроллеров AVR и PIC.

AVR микроконтроллеры, производимые фирмой ATMEL – это семейство 8-битных и более новых 32-битных микроконтроллеров с архитектурой RISC, которые совмещают в себе вычислительное ядро, Flash-память и разнообразную периферию (аналоговые и цифровые входы и выходы, интерфейсы и т.п.) на одном кристалле.

PIC (Peripheral Interface Controller) – контроллеры периферийных интерфейсов, выпускающиеся фирмой MicroChip и являющиеся типичными представителями RISC- процессоров.

Таблица 2 – Сравнительные характеристики семейств контроллеров

Характеристики	PIC	AVR
Электрические параметры		
Напряжение питания (V)	2.0 – 5.5	2.7 – 6.0 4.0 – 6.0
Нагрузочная способность портов В/В (ма)	25 8.5 -3.0	40 20.0 -3.0
Максимальная тактовая частота (МГц)	40	16
Архитектура		
Организация процессора	RISC, Гарвардская	RISC, Гарвардская
Число тактов на 1 командный цикл	4	1
Стек	Аппаратный, 8 уровней для PIC16 32 уровня для PIC18	Программный, ограничивается емкостью ОЗУ
Типы памяти	RAM, ROM	Регистры (32), RAM, ROM
Доступ к памяти программ и данных	Страничный	Полный
Прерывания, способ определения источника	PIC16 – программный (polling) PIC18 – векторный	Векторный
Уровень приоритета прерываний	PIC16 – определяется программой обработки PIC18 – переменный	Фиксированный
Сброс флажков прерываний	Программный	Аппаратный, при переходе по вектору. Возможен программный.
Вложенность прерываний	PIC16 – отсутствует PIC18 – имеется	Возможна
Флажки признаков в регистре состояний	PIC16 – C, Z, DC PIC18 – C, Z, DC, OV, N	C, Z, N, V, S, H
Система команд		
Количество инструкций	PIC16 – 35 PIC18 – 77	AT90S – 118 ATmega – 133
Количество методов адресации	PIC16 – 3 PIC18 – 7	8
Возможность доступа к данным в программной памяти	PIC16 – отсутствует PIC18 – полный	AT90S – частичный ATmega – полный

Микроконтроллеры AVR гораздо устойчивее к помехам, надежные в эксплуатации, чем микроконтроллеры семейства PIC. Микроконтроллеры семейства AVR выполняют один командный цикл за один такт, а PIC – за четыре такта. AVR также имеет большее количество методов адресации.

Проведя анализ существующих микроконтроллеров (таблица 2) необходимого уровня был выбран наилучший вариант – контроллер семейства AVR.

Система управления стендом реализована на платформе Arduino. Аппаратная платформа Arduino имеет широкое применение в проектах электронной автоматики. Она позволяет осуществлять управление внешними устройствами (реле, датчики, флэш-память, LCD-экраны, световая индикация и т.д.). Таким образом, данная аппаратная платформа объединяет все необходимое для разработки системы управления.

Рассмотрим три наиболее распространенные аппаратные платформы Arduino такие как Arduino UNO, Arduino LEONARDO, Arduino MEGA.

Arduino UNO – плата оснащена микроконтроллером ATmega328. Аппаратная платформа имеет 6 аналоговых входов, 14 цифровых вход/выходов, тактовая частота 16 МГц, USB разъем, силовой разъем, ICSP разъем и кнопку перезагрузки. Для работы, платформу необходимо подключить к компьютеру используя USB кабель, либо подать питание при помощи AC/DC адаптера или батареи.

Arduino LEONARDO – плата оснащена микроконтроллером ATmega32u4. Платформа имеет 20 цифровых вход/выходов, тактовая частота 16 МГц, микро-USB разъем, силовой разъем, ICSP разъем и кнопку перезагрузки. Для работы, платформу необходимо подключить к компьютеру посредством кабеля USB, либо подать питание при помощи адаптера AC/DC или батареи.

Arduino MEGA – плата оснащена микроконтроллером ATmega2560. Плата имеет 16 аналоговых входов, 54 цифровых входа/выходов, 4 последовательных порта UART, тактовая частота 16 МГц, USB коннектор, разъем питания, ICSP разъем и кнопка перезагрузки. Для работы, платформу необходимо подключить к компьютеру посредством USB кабеля или подать питание при помощи адаптера AC/DC, или аккумуляторной батареи.

Таблица 3 – Сравнительные характеристики платформ

Параметр	Arduino UNO	Arduino LEONARDO	Arduino MEGA
Микроконтроллер	ATmega328	ATmega32u4	ATmega2560
Рабочее напряжение	5В	5В	5В
Входное напряжение (рекомендуемое)	7-12В	7-12В	7-12В
Входное напряжение (предельное)	6-20В	6-20В	6-20В
Цифровые Входы/Выходы	14	20	54
Аналоговые входы	6	12	16
Постоянный ток через вход/выход	40 mA	40 mA	40 mA
Постоянный ток для вывода 3.3 В	50 mA	50 mA	50 mA
Флеш-память	32 Kb	32 Kb	256 Kb
ОЗУ	2 Kb	2 Kb	8 Kb
Энергонезависимая память	1 Kb	1 Kb	4 Kb
Тактовая частота	16 MHz	16 MHz	16 MHz

По данным приведенным в таблице 3 видно, что аппаратная платформа Arduino Mega (рисунок 20) имеет наибольшее количество цифровых входов/выходов, так же флеш-память, ОЗУ и энергонезависимую память. Исходя из вышеперечисленного для данного автоматизированного лабораторного стенда выбираем аппаратную платформу Arduino Mega 2560.

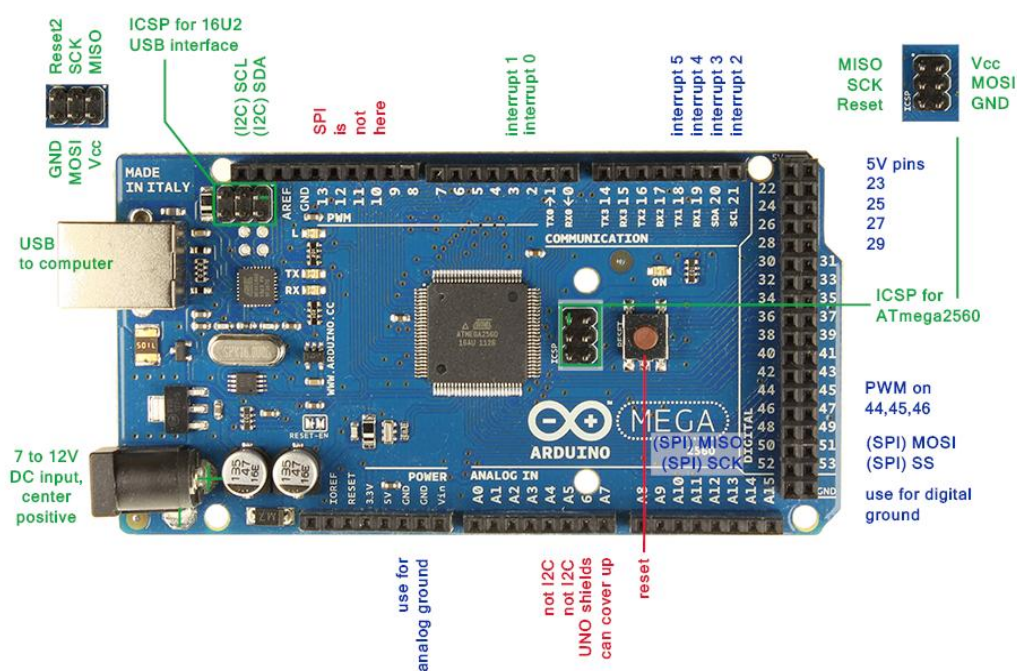


Рисунок 20 – плата ArduinoMega 2560

Пины платы Arduino Mega 2650

Таблица 4 – Цифровые пины

Пин	Адресация в скетче	Специальное назначение	ШИМ
Цифровой пин 0	0	RX (Serial)	
Цифровой пин 1	1	TX (Serial)	
Цифровой пин 2	2	Вход для прерываний 0	ШИМ
Цифровой пин 3	3	Вход для прерываний 1	ШИМ
Цифровой пин 4-12	4-12		ШИМ
Цифровой пин 13	13	Встроенный светодиод	ШИМ
Цифровой пин 14	14	TX (Serial3)	
Цифровой пин 15	15	RX (Serial3)	
Цифровой пин 16	16	TX (Serial2)	
Цифровой пин 17	17	RX (Serial2)	
Цифровой пин 18	18	TX (Serial1) Вход для прерываний 5	
Цифровой пин 19	19	RX(Serial1) Вход для прерываний 4	
Цифровой пин 20	20	I2C SDA Вход для прерываний 3	
Цифровой пин 21	21	I2C SCL Вход для прерываний 2	
Цифровой пин 22-43	22-43		
Цифровой пин 44-49	44-49		ШИМ
Цифровой пин 50	50	MISO	
Цифровой пин 51	51	MOSI	
Цифровой пин 52	52	SCK	
Цифровой пин 53	53	SCL	

Аналоговые пины

У платы Arduino Mega 16 аналоговых входов. Каждый из них соединен с 10-разрядным АЦП, поэтому в скетче можно получить 1024 уровней значений с помощью функции `analogRead()`. Диапазон значений напряжения по отношению к земле на аналоговых пинах по умолчанию равен 0-5 В. Этот диапазон можно изменить с помощью функции `analogReference()` и пина AREF.

Соединение с компьютером

Подключение к компьютеру и загрузка скетча осуществляется стандартно для большинства плат семейства Arduino. С помощью микросхемы ATmega16U2 устройство определяется как COM-порт, через который могут передаваться данные и осуществляется перепрошивка микроконтроллера.

Непосредственное управление лабораторным стендом осуществляется при помощи матричной клавиатуры (рисунок 21).

Клавиатура выполнена в виде матрицы 4x4 (рисунок 21), каждая кнопка является областью воздушного зазора между двумя диэлектрическими слоями с нанесенными на них токопроводящими покрытиями. Дорожки токопроводящего покрытия одного слоя нанесены горизонтально (выводы 1-4), а другого вертикально (выводы 5-8). Нажатие на кнопку приводит к соединению дорожки одного слоя с дорожкой другого, и как следствие, к замыканию одного из выводов 1-4 с одним из выводов 5-8.

Таблица 5 – Технические характеристики

Параметр	Значение
Габариты, мм	68×77×0,8
Рабочее напряжение, В	до 12
Максимальный ток, мА	100
Сопротивление изоляции, МОм	>100
Сопротивление контактов, Ом	<200
Дребезг контактов, мс	<5
Рабочая температура, °С	от 0 до +70
Вес, г	10

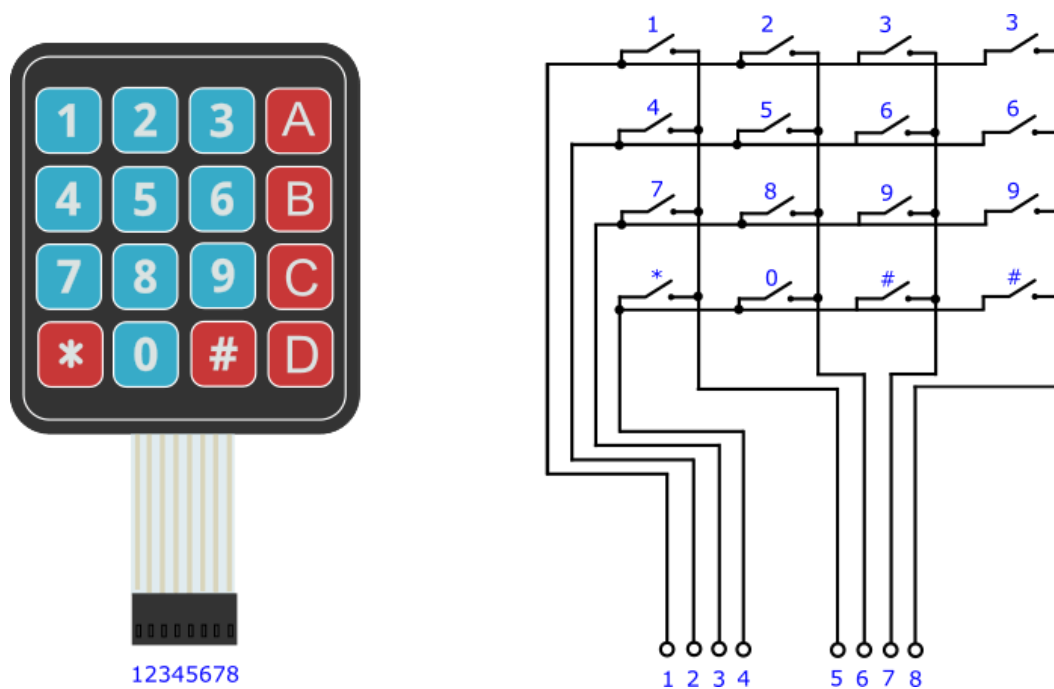


Рисунок 21 – Клавиатура управления манипулятором

На рисунке 21, клавиши 0-9, *, # – движения манипулятора, А-D – сценарии работы манипулятора.

1 – Подъем манипулятора

2 – Поворот манипулятора налево

3 – Поворот манипулятора направо

4 – Движение рук вперед

5 – Движение рук назад

6 – Разжатие верхнего схвата

7 – Разжатие нижнего схвата

8 – Ротация схвата вправо

9 – Ротация схвата влево

0 – Движение верхнего схвата вправо

* – Движение верхнего схвата влево

– Сброс

А – Сценарий 1

В – Сценарий 2

С – Сценарий 3

Д – Сценарий 4

3.2 Структурная схема системы управления

Разработана структурная схема управления промышленным манипулятором МП-11 (рисунок 22).

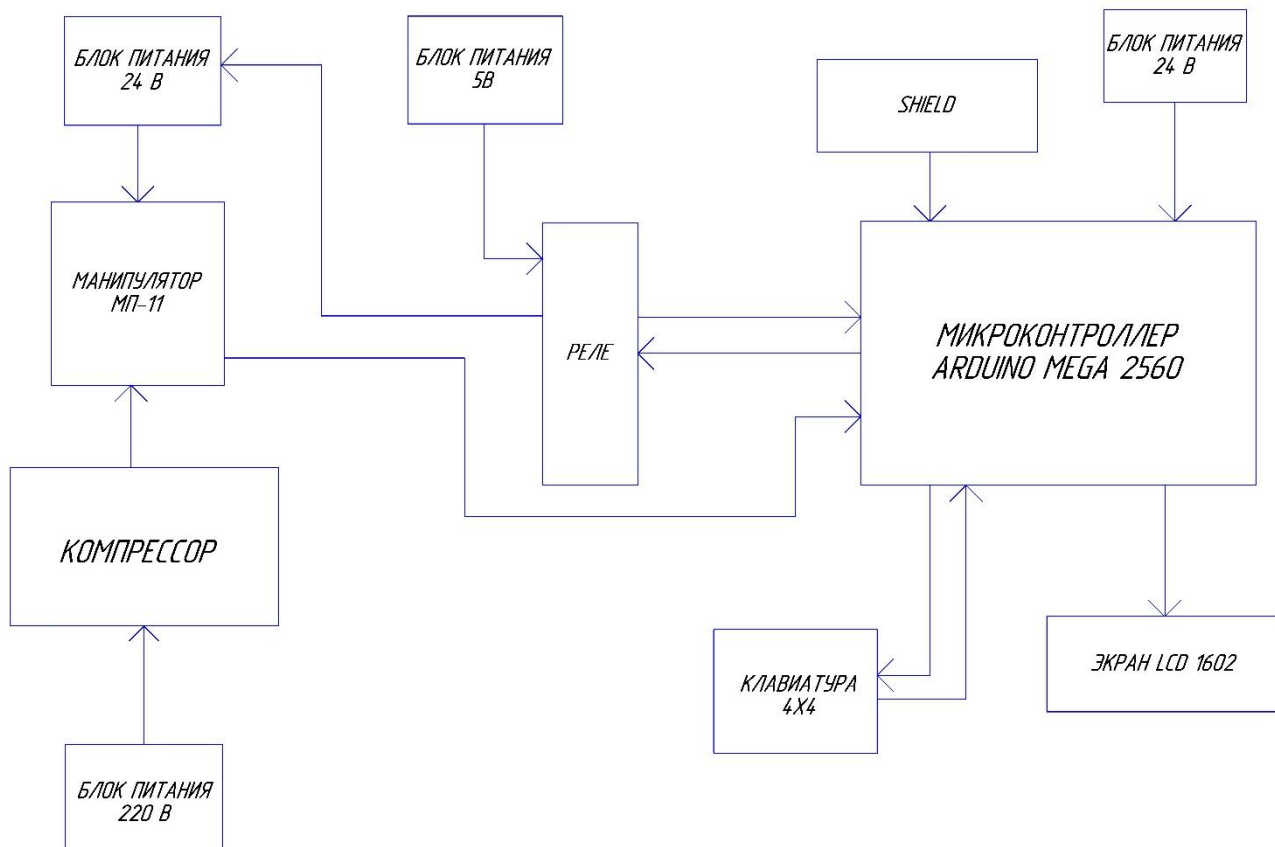


Рисунок 22 – Структурная схема управления пневматическим промышленным манипулятором МП-11

Компрессор питается от напряжения 220В. Воздушные магистрали, подающие воздух на пневмоцилиндры звеньев манипулятора, переключаются с помощью электропневмоклапанов (ЭПК), для питания ЭПК необходимо 24В. Сигнал с ЭПК передается на микроконтроллер Arduino Mega 2560, для питания платы необходимо 24В. С помощью клавиатуры на плату приходит сигнал о выполнении какого-либо действия, затем с помощью двухжильных проводов передается сигнал на реле, питание которых 5В, далее сигнал передается на электропневмоклапаны, выполняется действие. К микроконтроллеру Arduino подключен экран I2C, при нажатии кнопки на экране отображается какое действие выбрано пользователем.

3.3 Принципиальная схема системы управления

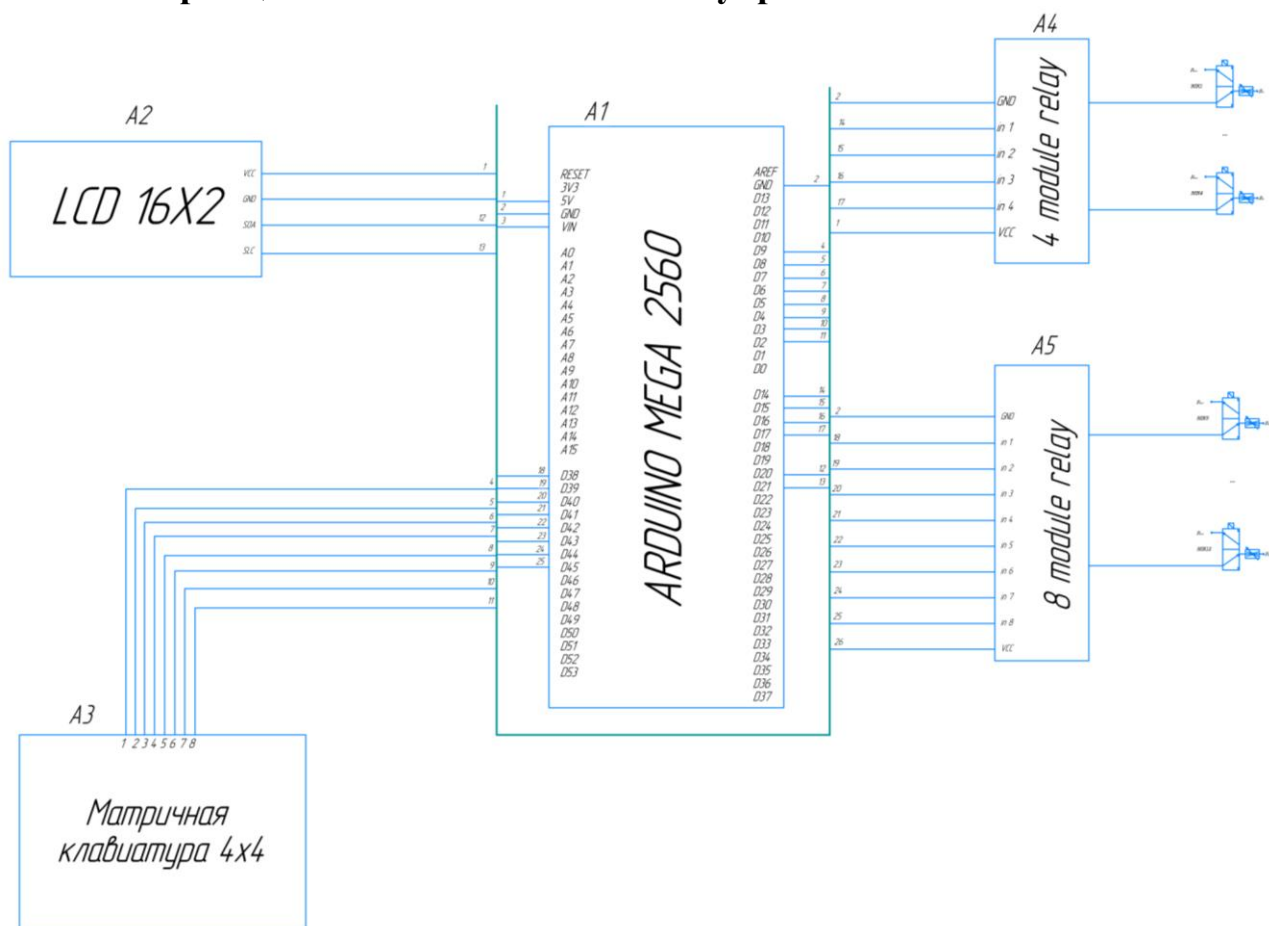


Рисунок 23 – Принципиальная схема управления пневматическим промышленным манипулятором МП-11

Микроконтроллер Arduino пошагово осуществляет выполнение заданной программы. Выполнение каждого шага обеспечивается контролем состояния датчиков (концевых выключателей). При перемещении управляемого звена манипулятора в крайнюю заданную точку траектории, соответствующий концевой выключатель замыкается, на входе контроллера появляется управляющий сигнал, происходит переход к выполнению следующей команды. Если сигнал концевого выключателя отсутствует, выполнение программы останавливается. Выбор команды осуществляется оператором с помощью матричной клавиатуры. При нажатии на кнопку клавиатуры выполняется действие, либо определенный цикл, на экран выводится выбранное действие. К микроконтроллеру Arduino так же подключены два модуля реле (4 и 8 шт.). К данным модулям

подключены электропневмоклапаны промышленного пневматического манипулятора МП-11.

3.4 Программное обеспечение системы управления

Плата Arduino программируется на специальном языке программирования, основанном на C/C++. Язык программирования Arduino является разновидностью C++, другими словами, отдельного языка программирования для Arduino не существует.

Дело в том, что все написанные скетчи компилируются с минимальными изменениями в программу на языке C/C++. Компилятор Arduino IDE значительно упрощает написание программ для этой платформы и создание устройств на Arduino становится намного доступней, тем кто не имеет больших познаний в языке C/C++. Ниже представлен код для управления лабораторным стендом.

```
// Подключаем библиотеку для работы с клавиатурой
#include <Keypad.h>
// Подключаем библиотеку для работы с шиной I2C
#include <Wire.h>
// Подключаем библиотеку для работы с LCD дисплеем по шине I2C
#include <LiquidCrystal_I2C.h>
// Объявляем объект библиотеки, указывая параметры дисплея (адрес I2C
= 0x27, количество столбцов = 16, количество строк = 2)
LiquidCrystal_I2C lcd(0x27, 16, 2);
// Количество строк клавиатуры
const byte ROWS = 4;
// Количество столбцов клавиатуры
const byte COLS = 4;
// Определение клавиатуры
char keys[ROWS][COLS] = {
//Расположение названия клавиш, как на клавиатуре
{
  '1','4','7','*'  }
, {
  '2','5','8','0'  }
,
{
  '3','6','9','#'  }
,

```

```

    {
      'A','B','C','D'  }
    };
    // Подключение строк клавиатуры ROW0, ROW1, ROW2 и ROW3 к кон-
тактам Arduino
    byte rowPins[ROWS] = {
      5, 4, 3, 2 };
    // Подключение столбцов клавиатуры COL0, COL1 and COL2 к контактам
Arduino
    byte colPins[COLS] = {
      9, 8, 7, 6};
    // Создание клавиатуры
    Keypad kpd = Keypad( makeKeypad(keys), rowPins, colPins, ROWS, COLS
);
    int relay12 = 45; // Подъем робота
    int relay11 = 44; // Движение схвата вправо
    int relay10 = 43; // Разжатие нижнего схвата
    int relay9 = 42; // Поворот робота по часовой стрелке
    int relay8 = 41; // Ротация схвата влево
    int relay7 = 40; // Ротация схвата вправо
    int relay6 = 39; // Разжатие верхнего схвата
    int relay5 = 38; // Движение верхнего схвата влево
    int relay4 = 17; // Движение рук манипулятора назад
    int relay3 = 16; // Движение рук манипулятора вперед
    int relay2 = 15; // Поворот робота против часовой стрелки
    int relay1 = 14; // Опускание манипулятора
    void setup() {
      pinMode(relay1, OUTPUT);
      pinMode(relay2, OUTPUT);
      pinMode(relay3, OUTPUT);
      pinMode(relay4, OUTPUT);
      pinMode(relay5, OUTPUT);
      pinMode(relay6, OUTPUT);
      pinMode(relay7, OUTPUT);
      pinMode(relay8, OUTPUT);
      pinMode(relay9, OUTPUT);
      pinMode(relay10, OUTPUT);
      pinMode(relay11, OUTPUT);
      pinMode(relay12, OUTPUT);
      Serial.begin(9600);
      digitalWrite(relay1,HIGH);
      digitalWrite(relay2,HIGH);
      digitalWrite(relay3,HIGH);
      digitalWrite(relay4,HIGH);

```



```

digitalWrite(relay5,HIGH);
digitalWrite(relay6,HIGH);
digitalWrite(relay7,HIGH);
digitalWrite(relay8,HIGH);
digitalWrite(relay9,HIGH);
digitalWrite(relay10,HIGH);
digitalWrite(relay11,HIGH);
digitalWrite(relay12,HIGH);
// Инициализация экрана
lcd.begin();
// Включение подсветки экрана
lcd.backlight();
// Устанавливается позиция, с которой выводится текст
lcd.setCursor(6, 0);
lcd.print("MP-11");
delay (3000);
lcd.clear();
}
void loop() {
char key = kpd.getKey();
if (key){
Serial.println(key);
switch (key) {
case '1':
lcd.setCursor(6, 0);
lcd.print("lifting robot");
digitalWrite(relay1,HIGH);
digitalWrite(relay2,HIGH);
digitalWrite(relay3,HIGH);
digitalWrite(relay4,HIGH);
digitalWrite(relay5,HIGH);
digitalWrite(relay6,HIGH);
digitalWrite(relay7,HIGH);
digitalWrite(relay8,HIGH);
digitalWrite(relay9,HIGH);
digitalWrite(relay10,HIGH);
digitalWrite(relay11,HIGH);
digitalWrite(relay12,LOW); // Подъем манипулятора
break;
case '2':
lcd.setCursor(6, 0);
lcd.print("turn left");
digitalWrite(relay1,HIGH);
digitalWrite(relay2,LOW); // Поворот робота против часовой стрелки

```

```

digitalWrite(relay3,HIGH);
digitalWrite(relay4,HIGH);
digitalWrite(relay5,HIGH);
digitalWrite(relay6,HIGH);
digitalWrite(relay7,HIGH);
digitalWrite(relay8,HIGH);
digitalWrite(relay9,HIGH);
digitalWrite(relay10,HIGH);
digitalWrite(relay11,HIGH);
digitalWrite(relay12,HIGH);
break;
case '3':
  lcd.setCursor(6, 0);
  lcd.print("turn right");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,HIGH);
  digitalWrite(relay7,HIGH);
  digitalWrite(relay8,HIGH);
  digitalWrite(relay9,LOW); // Поворот работа по часовой стрелке
  digitalWrite(relay10,HIGH);
  digitalWrite(relay11,HIGH);
  digitalWrite(relay12,HIGH);
  break;
case '4':
  lcd.setCursor(6, 0);
  lcd.print("forward");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,LOW); // Движение рук манипулятора вперед
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,HIGH);
  digitalWrite(relay7,HIGH);
  digitalWrite(relay8,HIGH);
  digitalWrite(relay9,HIGH);
  digitalWrite(relay10,HIGH);
  digitalWrite(relay11,HIGH);
  digitalWrite(relay12,HIGH);
  break;
case '5':

```

```

lcd.setCursor(6, 0);
lcd.print("backward");
digitalWrite(relay1,HIGH);
digitalWrite(relay2,HIGH);
digitalWrite(relay3,HIGH);
digitalWrite(relay4,LOW); // Движение рук манипулятора назад
digitalWrite(relay5,HIGH);
digitalWrite(relay6,HIGH);
digitalWrite(relay7,HIGH);
digitalWrite(relay8,HIGH);
digitalWrite(relay9,HIGH);
digitalWrite(relay10,HIGH);
digitalWrite(relay11,HIGH);
digitalWrite(relay12,HIGH);
break;
case '6':
  lcd.setCursor(6, 0);
  lcd.print("unclasp grip1");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,LOW); // Разжатие верхнего схвата
  digitalWrite(relay7,HIGH);
  digitalWrite(relay8,HIGH);
  digitalWrite(relay9,HIGH);
  digitalWrite(relay10,HIGH);
  digitalWrite(relay11,HIGH);
  digitalWrite(relay12,HIGH);
  break;
case '7':
  lcd.setCursor(6, 0);
  lcd.print("unclasp grip2");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,HIGH);
  digitalWrite(relay7,HIGH);
  digitalWrite(relay8,HIGH);
  digitalWrite(relay9,HIGH);
  digitalWrite(relay10,LOW); // Разжатие нижнего схвата

```

```

digitalWrite(relay11,HIGH);
digitalWrite(relay12,HIGH);
break;
case '8':
  lcd.setCursor(6, 0);
  lcd.print("rotation right");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,HIGH);
  digitalWrite(relay7,LOW); // Ротация схвата вправо
  digitalWrite(relay8,HIGH);
  digitalWrite(relay9,HIGH);
  digitalWrite(relay10,HIGH);
  digitalWrite(relay11,HIGH);
  digitalWrite(relay12,HIGH);
  break;
case '9':
  lcd.setCursor(6, 0);
  lcd.print("rotation left");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,HIGH);
  digitalWrite(relay7,HIGH);
  digitalWrite(relay8,LOW); // Ротация схвата влево
  digitalWrite(relay9,HIGH);
  digitalWrite(relay10,HIGH);
  digitalWrite(relay11,HIGH);
  digitalWrite(relay12,HIGH);
  break;
case '0':
  lcd.setCursor(6, 0);
  lcd.print("grip right");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,HIGH);

```

```

digitalWrite(relay7,HIGH);
digitalWrite(relay8,HIGH);
digitalWrite(relay9,HIGH);
digitalWrite(relay10,HIGH);
digitalWrite(relay11,LOW); // Движение верхнего схвата вправо
digitalWrite(relay12,HIGH);
break;
case '*':
  lcd.setCursor(6, 0);
  lcd.print("grip left");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,LOW); // Движение верхнего схвата влево
  digitalWrite(relay6,HIGH);
  digitalWrite(relay7,HIGH);
  digitalWrite(relay8,HIGH);
  digitalWrite(relay9,HIGH);
  digitalWrite(relay10,HIGH);
  digitalWrite(relay11,HIGH);
  digitalWrite(relay12,HIGH);
  break;
case '#':
  lcd.setCursor(6, 0);
  lcd.print("stop");
  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);
  digitalWrite(relay5,HIGH);
  digitalWrite(relay6,HIGH);
  digitalWrite(relay7,HIGH);
  digitalWrite(relay8,HIGH);
  digitalWrite(relay9,HIGH);
  digitalWrite(relay10,HIGH);
  digitalWrite(relay11,HIGH);
  digitalWrite(relay12,HIGH);
  break;

```

// Сценарий 1, рабочая нижняя рука

```

case 'A':
  lcd.setCursor(6, 0);
  lcd.print("algorithm 1");

```

```

digitalWrite(relay3,LOW); // Движение рук манипулятора вперед
    delay(2000);
digitalWrite(relay12,LOW); // Подъем манипулятора
    delay(2000);
digitalWrite(relay10,LOW); // Разжатие нижнего схвата
    delay(2000);
digitalWrite(relay9,LOW); // Поворот манипулятора по часовой стрелке
    delay(2000);
digitalWrite(relay10,HIGH); // Сжатие нижнего схвата
    delay(2000);
digitalWrite(relay9,HIGH); // Поворот манипулятора по часовой
стрелке
    delay(1000);
digitalWrite(relay2,LOW); // Поворот манипулятора против часовой
стрелки
    delay(2000);
digitalWrite(relay12,HIGH); // Опускание манипулятора
    delay(2000);
digitalWrite(relay8,LOW); // Ротация схвата влево
    delay(2000);
digitalWrite(relay3,HIGH); // Движение рук манипулятора вперед
    delay(1000);
digitalWrite(relay4,LOW); // Движение рук манипулятора назад
    delay(2000);
digitalWrite(relay10,LOW); // Разжатие нижнего схвата
    delay(4000);
digitalWrite(relay10,HIGH); // Сжатие нижнего схвата
    delay(1000);
digitalWrite(relay8,HIGH); // Ротация схвата влево
    delay(1000);
digitalWrite(relay4,HIGH); // Движение рук манипулятора назад
    delay(1000);
digitalWrite(relay2,HIGH); // Поворот робота налево
    delay(4000);
break;
// Сценарий 2, рабочая верхняя рука
case 'B':
    lcd.setCursor(6, 0);
    lcd.print("algorithm 2");
    digitalWrite(relay6,LOW); // Разжатие верхнего схвата
    delay(2000);
    digitalWrite(relay9,LOW); // Поворот робота по часовой стрелке
    delay(2000);
    digitalWrite(relay6,HIGH); // Сжатие верхнего схвата

```

```

delay(2000);
digitalWrite(relay12,LOW); // Подъем манипулятора
delay(2000);
digitalWrite(relay3,LOW); // Движение рук манипулятора вперед
delay(2000);
digitalWrite(relay9,HIGH); // Поворот манипулятора по часовой стрелке
стрелки
digitalWrite(relay2,LOW); // Поворот манипулятора против часовой
стрелки
delay(2000);
digitalWrite(relay11,LOW); // Движение верхнего схвата вправо
delay(2000);
digitalWrite(relay6,LOW); // Разжатие верхнего схвата
delay(2000);
digitalWrite(relay3,HIGH); // Движение рук манипулятора вперед
digitalWrite(relay4,LOW); // Движение рук манипулятора назад
delay(2000);
digitalWrite(relay12,HIGH); // Опускание манипулятора
delay(2000);
digitalWrite(relay6,HIGH); // Сжатие верхнего свата
digitalWrite(relay11,HIGH); // Движение верхнего свата вправо
digitalWrite(relay4,HIGH); // Движение рук манипулятора назад
digitalWrite(relay2,HIGH); // Поворот манипулятора против часовой
стрелки
delay(4000);
break;
// Сценарий 3, работают оба схвата
case 'C':
  lcd.setCursor(6, 0);
  lcd.print("algorithm 3");
  digitalWrite(relay6,LOW); // Разжатие верхнего схвата
  digitalWrite(relay10,LOW); // Разжатие нижнего схвата
  delay(2000);
  digitalWrite(relay2,LOW); // Поворот робота против часовой стрелки
  delay(2000);
  digitalWrite(relay12,LOW); // Подъем манипулятора
  delay(2000);
  digitalWrite(relay3,LOW); // Движение рук вперед
  delay(2000);
  digitalWrite(relay10,HIGH); // Сжатие нижнего схвата
  delay(2000);
  digitalWrite(relay2,HIGH); // Поворот манипулятора против часовой
стрелки
digitalWrite(relay9,LOW); // Поворот робота по часовой стрелке

```



```

delay(2000);
digitalWrite(relay6,HIGH); // Сжатие верхнего схвата
delay(2000);
digitalWrite(relay3,HIGH); // Движение рук манипулятора вперед
digitalWrite(relay4,LOW); // Движение рук манипулятора назад
delay(2000);
digitalWrite(relay11,LOW); // Движение верхнего схвата вправо
digitalWrite(relay7,LOW); // Ротация нижнего схвата вправо
delay(2000);
digitalWrite(relay12,HIGH); // Опускание манипулятора
delay(2000);
digitalWrite(relay9,HIGH); // Поворот манипулятора по часовой стрелке
ке
digitalWrite(relay2,LOW); // Поворот манипулятора против часовой
стрелки
delay(2000);
digitalWrite(relay4,HIGH); // Движение рук назад
digitalWrite(relay3,LOW); // Движение рук вперед
delay(2000);
digitalWrite(relay6,LOW); // Разжатие верхнего схвата
digitalWrite(relay10,LOW); // Разжатие нижнего схвата
delay(2000);
digitalWrite(relay3,HIGH); // Движение рук манипулятора вперед
digitalWrite(relay4,LOW); // Движение рук манипулятора назад
delay(2000);
digitalWrite(relay4,HIGH); // Движение рук манипулятора назад
digitalWrite(relay6,HIGH); // Сжатие верхнего схвата
digitalWrite(relay10,HIGH); // Сжатие нижнего схвата
digitalWrite(relay2,HIGH); // Поворот манипулятора против часов
стрелки
digitalWrite(relay7,HIGH); // Ротация нижнего схвата вправо
digitalWrite(relay11,HIGH); // Движение верхнего схвата вправо
delay(4000);
break;
// Сценарий 3, работают оба схвата
case 'D':
  lcd.setCursor(6, 0);
  lcd.print("algorithm 4");
  digitalWrite(relay12,LOW); // Подъем манипулятора
  delay(2000);
  digitalWrite(relay6,LOW); // Разжатие верхнего схвата
  digitalWrite(relay10,LOW); // Разжатие нижнего схвата
  delay(2000);
  digitalWrite(relay9,LOW); // Поворот робота по часовой стрелке

```



```

delay(2000);
digitalWrite(relay3,LOW); // Движение рук манипулятора вперед
delay(2000);
digitalWrite(relay6,HIGH); // Сжатие верхнего схвата
digitalWrite(relay10,HIGH); // Сжатие нижнего схвата
delay(2000);
digitalWrite(relay3,HIGH); // Движение рук манипулятора вперед
digitalWrite(relay4,LOW); // Движение рук манипулятора назад
delay(2000);
digitalWrite(relay9,HIGH); // Поворот манипулятора по часовой стрелке
digitalWrite(relay2,LOW); // Поворот манипулятора против часовой стрелки
delay(2000);
digitalWrite(relay12,HIGH); // Опускания манипулятора
delay(2000);
digitalWrite(relay4,HIGH); // Движение рук манипулятора назад
digitalWrite(relay3,LOW); // Движение рук манипулятора вперед
delay(2000);
digitalWrite(relay7,LOW); // Ротация схвата вправо
digitalWrite(relay5,LOW); // Движение верхнего схвата влево
delay(2000);
digitalWrite(relay6,LOW); // Разжатие верхнего схвата
digitalWrite(relay10,LOW); // Разжатие нижнего схвата
delay(2000);
digitalWrite(relay3,HIGH); // Движение рук манипулятора вперед
digitalWrite(relay4,LOW); // Движение рук манипулятора назад
delay(2000);
digitalWrite(relay4,HIGH); // Движение рук манипулятора назад
digitalWrite(relay6,HIGH); // Сжатие верхнего схвата
digitalWrite(relay10,HIGH); // Сжатие нижнего схвата
digitalWrite(relay5,HIGH); // Движение верхнего схвата влево
digitalWrite(relay7,HIGH); // Ротация схвата вправо
digitalWrite(relay2,HIGH); // Поворот манипулятора против часовой стрелки
delay(4000);
break;
}
}
}

```

4 Комплектация шкафа управления

4.1 Микроконтроллер Arduino Mega 2560

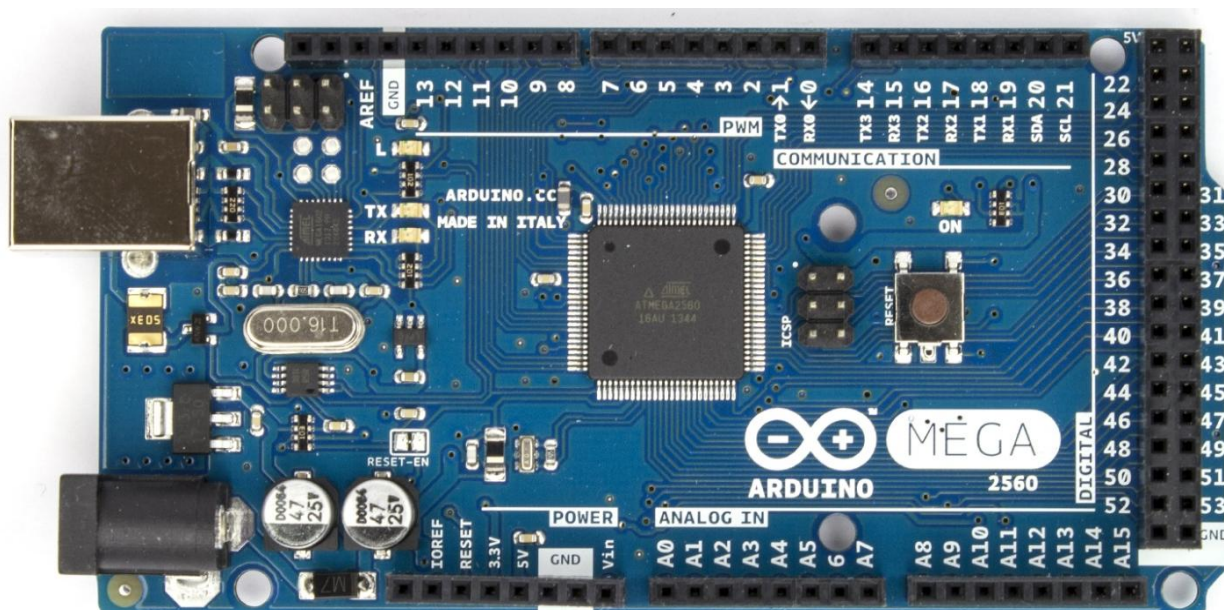


Рисунок 24 – Микроконтроллер Arduino Mega 2560

Технические характеристики, цифровые и аналоговые пины, подключение к компьютеру были рассмотрены в пункте 3.1.

Питание аппаратной платформы Arduino Mega 2560

Обеспечить питание для платформы можно как через подключение по USB, так и от внешнего источника питания. Источник питания выбирается автоматически.

Внешнее питание (не USB) может подаваться через преобразователь напряжения AC/DC (блок питания) или аккумуляторной батареей. Преобразователь напряжения подключается посредством разъема 2.1 мм с положительным полюсом на центральном контакте. Провода от батареи подключаются к выводам GND и VIN разъема питания (POWER).

Платформа может работать при внешнем питании от 6В до 20В. При напряжении питания ниже 7В, вывод 5V может выдавать менее 5В, при этом платформа может работать нестабильно. При использовании напряжения выше 12В регулятор напряжения может перегреться и повредить плату. Рекомендуемый диапазон от 7В до 12В.

Аппаратная платформа Arduino Mega 2560, в отличие от предыдущих версий, не использует FTDI USB микроконтроллер. Обмен данными по USB осуществляется с помощью микроконтроллера Atmega8U2.

Выводы питания:

VIN. Вход используется для подачи питания от внешнего источника. Подача напряжения питания происходит через данный вывод. Если питание подается на разъем 2.1 мм, то на этот вход можно подать питание.

5V. Используется для питания микроконтроллера и компонентов на плате. Питание может подаваться от вывода VIN через регулятор напряжения, либо от разъема USB, либо от другого регулируемого источника напряжения 5В.

3V3. Напряжение на выводе 3.3В генерируемое микросхемой FTDI на платформе. Максимальное потребление тока 50 мА.

GND. Выводы заземления.

4.2 Arduino Sensor Shield

Shield достаточно прост – его основная задача предоставить более удобные варианты подключения к плате Arduino. Это осуществляется за счет дополнительных разъемов питания и земли, выведенных на плату к каждому из аналоговых и цифровых пинов. Также на плате можно найти разъемы для подключения внешнего источника питания (для переключения нужно установить перемычки), светодиод и кнопка перезапуска.

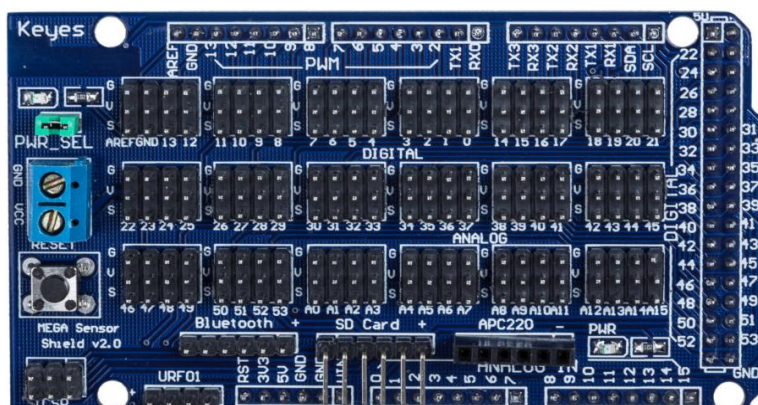


Рисунок 25 – Arduino Sensor Shield

Коммутационная плата Mega Sensor Shield V2.0 предназначена для расширения функциональности контроллеров на платформе Arduino Mega. Плата

позволяет облегчить подключение к контроллеру Arduino MEGA большого количества простых периферийных устройств. Это удобно в проектах с использованием множества датчиков, светодиодов, кнопок, двигателей, реле.

На плате предусмотрена кнопка сброса или перезагрузки RESET.

Есть индикатор включения питания, подписанный PWR.

Контактные площадки Mega Sensor Shield V2.0 совместимы с Arduino Mega 2560 и Mega 1280.

Интерфейс для подключения внешнего источника питания (блока питания, батареи). Обозначение выводов: GND, VCC.

Колодка питания Arduino, обозначение выводов: RESET, 3V3, 5V, GND, VIN.

Установлено 16 аналоговых входов, обозначенных 0 – 15.

Analog IO ports (интерфейс аналоговых входов-выходов) имеет 15 выводов. Обозначение выводов: A0 – A15. Каждый вход/выход имеет 3 контакта GND, VCC, Signal.

ICSP (внутрисхемное программирование) – для подключения программаторов. Имеет 6 выводов, первый контакт обозначен на схеме интерфейсов платы.

Входы и выходы интерфейса RS232 (последовательная шина COM).

ACP220 wireless module interface (интерфейс для подключения внешнего Wi-Fi модуля ACP220) имеет 6 входов. Обозначение входов: GND, VCC, не задействован, D18, D19, не задействован.

Входы-выходы Arduino контроллера, обозначенные: 0 – 13, GND, AREF. Вывод AREF (опорное напряжение для аналоговых входов).

Питание платы осуществляется или с Arduino, или от внешних источников питания (блоков питания, батарей). Переключение с источника питания Arduino на внешний источник осуществляется с помощью переключки SEL_PWR. Для питания платы нужно напряжение 5В.

4.3 Реле 8 relay module



Рисунок 26 – Плата реле (8 модулей)

Модуль построен на базе реле «JQC-3FF-S-Z». Провода, идущие от реле, надёжно закрепляются клеммниками при помощи винтов. За счёт наличия на каждом клеммнике трёх контактов, можно выбирать для каждого канала начальное состояние: нормально разомкнутое (NO) или нормально замкнутое (NC). Модулем можно управлять, непосредственно широким диапазоном микроконтроллеров, такими как для Arduino, AVR, PIC.

На выходе канала реле имеется три вывода, средний вывод всегда соединён с одним из соседних, а с каким именно, зависит от логического уровня на ВХОДЕ «IN» этого канала:

Если на вход «IN» канала подать уровень логической «1», то средний вывод выхода канала будет соединён так как это нарисовано на тыльной стороне платы. Такое же соединение будет при отсутствии питания модуля на выводах «VCC» и «GND».

Если на вход «IN» канала подать уровень логического «0», то средний вывод выхода канала будет соединён НЕ так как это нарисовано на тыльной стороне платы.

Таким образом вы можете использовать реле модуля: либо как нормально замкнутое, либо как нормально разомкнутое, либо как переключающее.

4.4 Реле 4 relay module

Электромеханическое 4-х канальное реле способно параллельно управлять 4-мя каналами, не зависящими друг от друга. С помощью такого реле станет возможным размыкание и замыкание нагрузочных цепей с напряжением до 250 В и током до 10 А. Но рекомендуется коммутировать цепи с током до 7 А. Модуль построен на базе реле «SRD-05-VDC-SL-C», чем и обеспечивается его коммутационная способность.

Провода, идущие от реле, надёжно закрепляются клеммниками при помощи винтов. За счёт наличия на каждом клеммнике трёх контактов, можно выбирать для каждого канала начальное состояние: нормально разомкнутое (NO) или нормально замкнутое (NC). Модулем можно управлять, подключив его к Arduino, уровень логической «1» выводов которой равен 5 В.

Основные достоинства

- Компактные размеры
- Взаимодействие с Arduino через любой цифровой контакт

На выходе канала реле имеется три вывода, средний вывод всегда соединён с одним из соседних, а с каким именно, зависит от логического уровня на ВХОДЕ «IN» этого канала:

Если на вход «IN» канала подать уровень логической «1», то средний вывод выхода канала будет соединён так как это нарисовано на тыльной стороне платы. Такое же соединение будет при отсутствии питания модуля на выводах «VCC» и «GND».

Если на вход «IN» канала подать уровень логического «0», то средний вывод выхода канала будет соединён НЕ так как это нарисовано на тыльной стороне платы.

Таким образом вы можете использовать реле модуля: либо как нормально замкнутое, либо как нормально разомкнутое, либо как переключающее.

Питание:

Входное напряжение питания 5В постоянного тока, подаётся на выводы «Vcc» и «GND» модуля.

Дополнительно, на плате модуля имеется разъём из двух выводов: «JD-VCC» и «VCC», на котором установлена переключатель. Если убрать эту переключатель и подать отдельное (дополнительное) питание 5В постоянного тока на выводы «JD-VCC» и «GND», то питание логики модуля будет осуществляться с выводов «JD-VCC» и «GND», а питание обмоток реле (каналов) модуля будет осуществляться с выводов «VCC» и «GND», а питание обмоток реле (каналов) модуля будет осуществляться с выводов «JD-VCC» и «GND». Такой способ подключения позволяет избежать появление помех на шине питания «VCC» и «GND», которые могут повлиять на работу других модулей на этой шине питания.

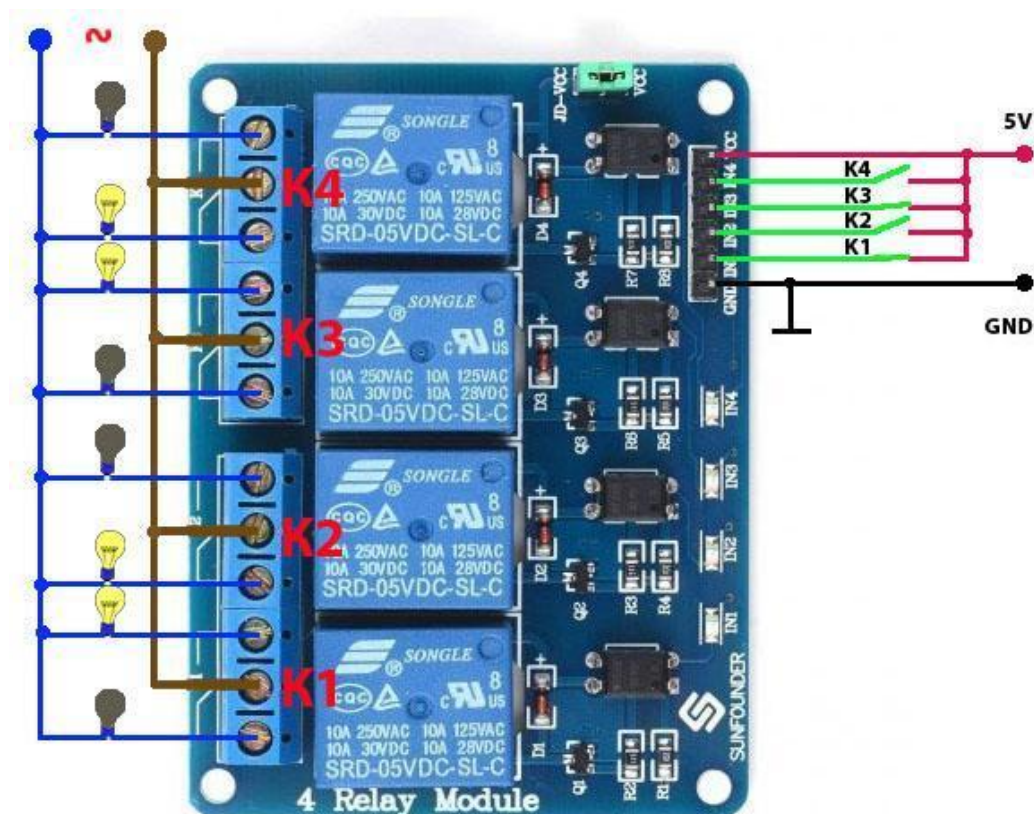


Рисунок 27 – 4 relay module

Характеристики:

Напряжение питания модуля: 5В постоянного тока.

Ток, потребляемый модулем: до 75 мА на каждый включённый канал.

Коммутируемые модулем выходные цепи: до 30В постоянного тока 10 А; до 250В переменного тока 10 А

Сопротивление обмотки реле: $70 \Omega \pm 10\%$.

Сопротивление изоляции реле: выше 100 МОм

Время срабатывания реле при включении: до 10 мс.

Время срабатывания реле при выключении: до 5 мс.

Скорость механических переключений: до 300 операций / мин.

Материал контактов реле: AgCdO.

Рабочая температура: -25 ... +70 °С

Рабочая влажность: 45 ... 85%

4.5 Матричная клавиатура 4x4



Рисунок 28 – Матричная клавиатура

Мембранная клавиатура 4x4 матрица может использоваться для управления различными устройствами, например, для набора кодов в кодовых замках или для программирования.

Для работы с мембранной клавиатурой, её необходимо подключить к устройству, которым нужно управлять, создать программу управления и подключить питание.

Принцип работы – замыкание определенных пар проводов осуществляется при нажатии одной из клавиш матричной клавиатуры.

Данная клавиатура имеет один управляемый интерфейс для подключения к устройству: 8-пиновый интерфейс типа «мама».

Питание матричной клавиатуры производится от устройства, которым нужно управлять или от внешнего источника. потребляемый ток 100 мА, напряжение питания до 35 В.

4.6 Дисплей LCD

LCD дисплей – частый гость в проектах Arduino. Но в сложных схемах может возникнуть проблема недостатка портов Arduino из-за необходимости подключить экран, у которого очень много контактов. Выходом в этой ситуации может стать I2C /PC переходник, который подключает практически стандартный для Arduino экран 1602 к платам Uno, Nano или Mega всего лишь при помощи 4 пинов.

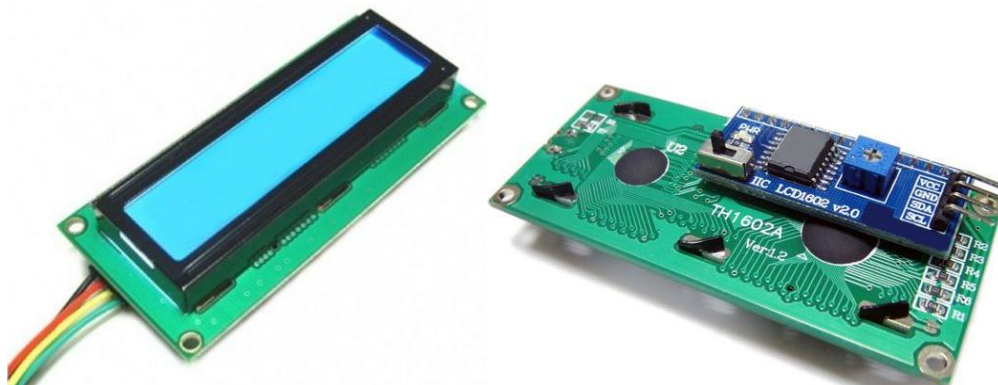


Рисунок 29 – Дисплей LCD

Технические характеристики дисплея:

- Символьный тип отображения, есть возможность загрузки символов;
- Светодиодная подсветка;
- Контроллер HD44780;
- Напряжение питания 5В;
- Формат 16x2 символов;
- Диапазон рабочих температур от -20°C до $+70^{\circ}\text{C}$, диапазон температур хранения от -30°C до $+80^{\circ}\text{C}$;
- Угол обзора 180 градусов.

Контакты на этом дисплее пронумерованы от 1 до 16. Нанесены они на задней части платы. Как именно они подключаются к Arduino, показано на рисунке 30.

Дисплей 1602		Arduino	Дисплей 1602		Arduino
1	GND	GND	9	DB2	Не подключается
2	VDD	5V	10	DB3	Не подключается
3	Contrast	К GND через резистор	11	DB4	Пин 2
4	RS	Пин 0	12	DB5	Пин 3
5	R/W	GND	13	DB6	Пин 4
6	Enable	Пин 1	14	DB7	Пин 5
7	DB0	Не подключается	15	Back LED+	К 5V через резистор
8	DB1	Не подключается	16	Back LED-	GND

Рисунок 30 – Подключение контактов LCD 1602 к Arduino

Соединительные провода мама-папа

Для соединения всех комплектующих системы управления автоматизированным лабораторным стендом на базе промышленного пневматического манипулятора МП-11 осуществляется с помощью двухжильных проводов мама-папа (рисунок 31).

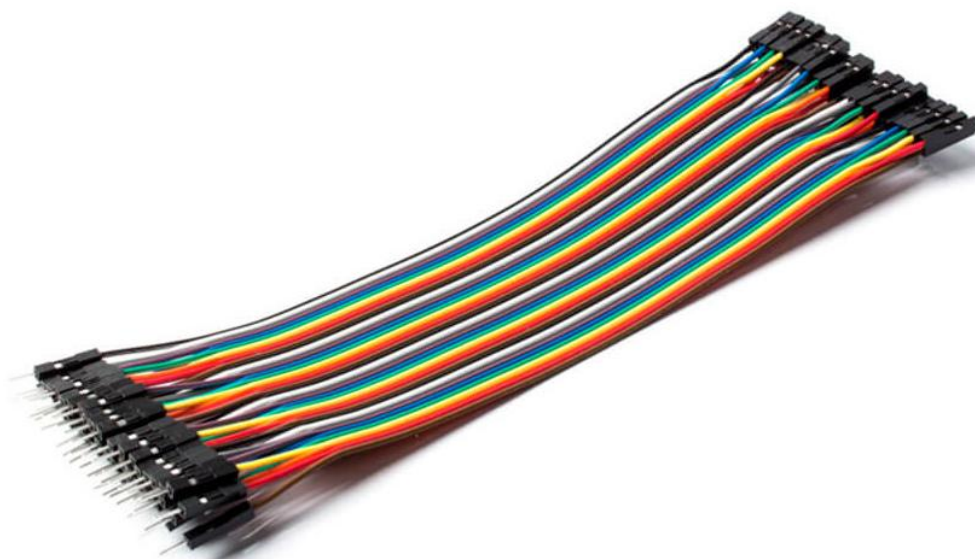


Рисунок 31 – Соединительные провода мама-папа

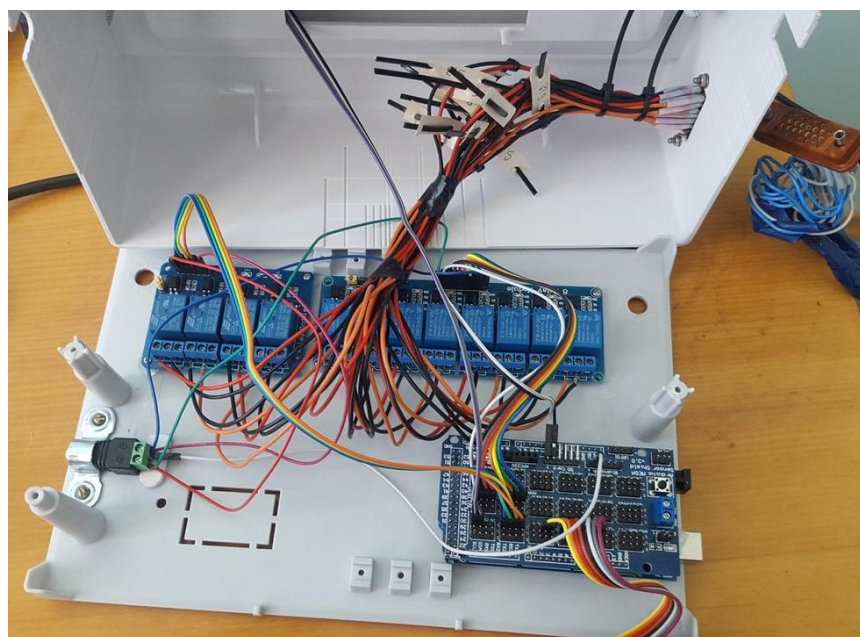


Рисунок 32 – Шкаф управления автоматизированным лабораторным стендом
(крышка открыта)

Клапаны промышленного пневматического манипулятора МП-11 соединены с реле в шкафу управления через разъем RS-232.


Характеристики RS-232:

- поддерживаемая топология соединения - "точка-точка";
- направление передачи – дуплексная и симплексная;
- режим обмена – асинхронный (в последовательных каналах ПЭВМ) и синхронный (реализуется с помощью специальных адаптеров SDLC);
- метод кодирования информации – NRZ;
- используемые сигналы - потенциальные (несимметричные), уровень логического нуля: 3..25В, уровень логической единицы: -3..-25В, в ПЭВМ используются уровни $\pm 12В$;

- скорость передачи информации - от 110 бит/с до 115200 бит/с;
- количество линий передачи информации и служебных сигналов - до 25

RS-232 – интерфейс для осуществления передачи информации с одного устройства на другое на расстоянии до 20 м. Информация передается по проводам с уровнями сигналов, которые отличаются от стандартных 5В, для обеспе-

чения большей устойчивости к помехам. Асинхронная передача данных осуществляется с установленной скоростью при синхронизации уровнем сигнала стартового импульса.



Контакт	Обозн.	Направление	Описание
1	SHIELD	---	Shield Ground - защитная земля, соединяется с корпусом устройства и экраном кабеля
2	TXD	-->	Transmit Data - Выход передатчика
3	RXD	<--	Receive Data - Вход приемника
4	RTS	-->	Request to Send - выход запроса передачи данных
5	CTS	<--	Clear to Send - вход разрешения терминалу передавать данные
6	DSR	<--	Data Set Ready - вход сигнала готовности от аппаратуры передачи данных
7	GND	---	System Ground - сигнальная (схемная) земля
8	CD	<--	Carrier Detect - вход сигнала обнаружения несущей удаленного модема
9-19	N/C	-	-
20	DTR	-->	Data Terminal Ready - выход сигнала готовности терминала к обмену данными
21	N/C	-	-
22	RI	<--	Ring Indicator - вход индикатора вызова (звонка)
23-25	N/C	-	-

Рисунок 33 – Контакты разъема RS-232

Корпус шкафа системы управления автоматизированным лабораторным стендом на базе пневматического промышленного манипулятора представлен на рисунке 34.



Рисунок 34 – Корпус шкафа управления

Заключение

Целью выпускной бакалаврской работы было создание автоматизированного лабораторного стенда на базе пневматического промышленного манипулятора МП-11. Для достижения поставленной цели были решены следующие задачи:

1. Разработан алгоритм управления лабораторным стендом
2. Разработана система управления лабораторным стендом
3. Разработана структурная схема системы управления
4. Разработана принципиальная схема системы управления

В процессе выполнения выпускной бакалаврской работы были разработаны алгоритмы управления автоматизированным лабораторным стендом с одним и двумя рабочими схватами. Созданы блок-схемы алгоритмов управления для наглядности работы лабораторного стенда. Разработано программное обеспечение на базе аппаратной платформы Arduino Mega 2560. Собран шкаф управления автоматизированным лабораторным стендом.

Цель, поставленная для выпускной бакалаврской работы достигнута, все задачи выполнены.

Список используемой литературы

1. Arduino [Электронный ресурс]. URL: <http://robocraft.ru/blog/arduino/> (дата обращения: 12.04.2016).
2. Emanuel Todorov, Chunyan Hu, Alex Simpkins and Javier Movellan, Identification and control of a pneumatic robot, 2010, , (36941 символов с пробелами), <http://ieeexplore.ieee.org/document/5627779/figure>
3. Jordi Lucero, Robotics Potential Fields, 2009, (9247 символов с пробелами) <http://scholarcommons.usf.edu/ujmm/vol2/iss1/3/>
4. Nabil M. Hewahi, Abed Alkader A. Almobayed, Swarm Robotics with Circular Formation Motion Including Obstacles Avoidance, 2017, (33350 символов с пробелами) <http://www.edusoft.ro/brain/index.php/brain/article/view/695/774>
5. Sandra Mattsson, Åsa AB Fasth, Fast-Berglund, Johan Stahre, Describing Human-Automation Interaction in production, 2012, (37286 символов с пробелами) https://www.researchgate.net/profile/Sandra_Mattsson/publication/236944823_Describing_Human-Automation-Interaction_in_production/links/0c96051a4e4debd10d000000/Describing-Human-Automation-Interaction-in-production.pdf
6. Sho Maeda, Nobutaka Tsujiuchi, Takayuki Koizumi, Mitsumasa Sugiura and Hiroyuki Kojima, Development and Control of a Pneumatic Robot Arm for Industrial Fields, 2012, (23812 символов с пробелами), <http://journals.sagepub.com/doi/full/10.5772/50538>
7. Баранов, В.Н. Применение микроконтроллеров AVR: схемы, алгоритмы, программы / В.Н.Баранов – М.: Издательский дом «Додэка– XXI», 2004. – Мировая электроника) – 288 с
8. Булгаков А.Г. Промышленные роботы : кинематика, динамика, контроль и управление:/ А.Г. Булгаков, В.А. Воробьев. – Москва : Солоно-Пресс. 2007. -488с.

9. Воротников С.А. Информационные устройства робототехнических систем. М.:Изд-во МГТУ им. Н.Э. Баумана, 2005. – 384 с.
10. Головин, И.Г. Языки и методы программирования: Учебник для студентов учреждений высшего профессионального образования / И.Г. Головин, И.А. Волкова. – М.: ИЦ Академия, 2012. – 304 с.
11. Зенкевич С.Л. Управление роботами: учеб. для вузов / С.Л. Зенкевич, А.С. Ющенко.– М: МГТУ им. Н.Э. Баумана, 2000. 400 с
12. Зенкевич С.Л., Ющенко А.С. Основы управления манипуляционными роботами: учебник для вузов. – 2-е изд., исправ. И доп. М.: Изд-во МГТУ им. Н.Э.Баумана, 2004. – 480 с.
13. Калашников В.И. Электроника и микропроцессорная техника: Учебник для студ. учреждений высш. проф. обр. / В.И. Калашников, С.В. Нефедов. – М.: ИЦ Академия, 2012. – 368 с
14. Канцедал С.А. Алгоритмизация и программирование : Учебное пособие / С.А. Канцедал. – М.: ИД ФОРУМ, НИЦ ИНФРА– М, 2013. – 352 с.
15. Климов А. С. Современные роботы в машиностроении: учебное пособие / А. С. Климов, О. В. Бойченко. – Тольятти : ТГУ, 2005. – 132 с.
16. Козырев Ю.Г. Промышленные роботы: справочник. – М.:Машиностроение, 1988, 392 с.
17. Кравцов А.Г. Изучение конструкций манипуляторов промышленных роботов РБ – 241 и МП – 11.01: методические указания к лабораторной работе / А.Г. Кравцов. – Оренбург: ОГУ, 2013. – 57 с.
18. Медведев А. Е. Автоматизация производственных процессов : учеб. пособие / А. Е. Медведев, А. В. Чупин ; Кузбас. гос. техн. ун-т. – Кемерово, 2009. – 325 с.
19. Петин В.А. Проекты с использованием контроллера Arduino. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2015. – 464 с.
20. Челпанов И.Б. Устройство промышленных роботов : учеб. для учащихся приборостроит. Техникумов / И.Б. Челпанов. – 2-е изд., перераб. и доп. – СПб.: Политехника, 2001. – 203 с.