**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Сибирский государственный университет науки и технологий**

**имени академика М. Ф. Решетнева»**

ИНСТИТУТ машиноведения и мехатроники

НАПРАВЛЕНИЕ 15.03.06 «Мехатроника и робототехника»

НАПРАВЛЕННОСТЬ (ПРОФИЛЬ) «Мехатроника»

КАФЕДРА «Технология машиностроения»

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

Вид ВКР: бакалаврская работа

**РАЗРАБОТКА ЭЛЕКТРОМЕХАНИЧЕСКОГО МАНИПУЛЯТОРА**

наименование темы ВКР

**С КОМПЬЮТЕРНЫМ УПРАВЛЕНИЕМ**

|  |  |  |
| --- | --- | --- |
| Обучающийся | \_\_\_\_\_\_\_\_\_\_\_\_\_ | Б.В. Паршинцев |
|  | подпись | инициалы и фамилия |
| Обучающийся | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Э.А. Акелов |
|  | подпись | инициалы и фамилия |
| Руководитель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | В.А.Будьков |
|  | подпись | инициалы и фамилия |
| Ответственный за нормоконтроль | \_\_\_\_\_\_\_\_\_\_\_\_\_ | А.В. Скрипка |
|  | подпись | инициалы и фамилия |

Допускается к защите

Заведующий кафедрой\_\_\_\_\_\_\_\_\_\_\_\_\_/ Л.В. Ручкин /

«\_\_\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г.

Красноярск 2020 г.

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Сибирский государственный университет науки и технологий**

**имени академика М. Ф. Решетнева»**

Институт машиноведения и мехатроники

Технология маниностроения

УТВЕРЖДАЮ

Зав. кафедрой ТМС

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Л.В. Ручкин

«\_\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

**ЗАДАНИЕ**

**НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ**

**в форме бакалаврской работы**

Обучающиеся\_Б.В. Паршицнев, Э.А. Акелов\_\_\_\_\_\_\_\_\_\_\_\_

группа БМЕ16-01 Направления 15.03.06 «Мехатроника и робототехника»

Направленность (профиль) «Мехатроника»

Тема ВКР: Разработка электромеханического манипулятора с компьютерным управлением

Утверждена приказом по университету от 26 марта 2020г. №647

Руководитель ВКР – В.А. Будьков, старший преподаватель кафедры ТМС, СибГУ

Исходные данные ВКР: толщина листовой заготовки 5мм, перемещение детали вертикально 250мм, перемещение детали горизонтально 250мм, сталь Сталь 25 ГОСТ 1050-2013

Перечень разделов ВКР: Аналитическая часть, Конструкторская часть, Система управления, Программная часть

Перечень графического материала (с указанием обязательных чертежей при необходимости)

Манипулятор электромеханический. Чертеж общего вида – 1 лист А1

Манипулятор электромеханический. Сборочный чертеж – 2 листа А1

Привод поворота схвата. Сборочный чертеж – 1 лист А1

Привод вертикального перемещения. Сборочный чертеж – 1 лист А1

Привод поворота манипулятора. Сборочный чертеж – 1 лист А1

Манипулятор электромеханический. Схема функциональная – 1 лист А1

Система управления манипулятором. Схема электрическая соединений – 1 лист А1

Драйвер. Схема электрическая принципиальная – 1 лист А1

Система управления манипулятором. Алгоритм – 1 лист А1

Программа управления манипулятором. Формы кранные – 1 лист А1

Срок сдачи студентом первого варианта ВКР - « 16» июня 2020г.

Срок сдачи студентом окончательного варианта ВКР - «29» июня 2020г.

|  |  |  |
| --- | --- | --- |
| Руководитель ВКР | \_\_\_\_\_\_\_\_\_\_\_\_\_ | В.А.Будьков |
|  | подпись | инициалы и фамилия |
| Задание принял к исполнению | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Б.В. Паршинцев |
|  | подпись | инициалы и фамилия |
| Задание принял к исполнению | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Э.А. Акелов |
|  | подпись | инициалы и фамилия |
|  |  |  |
|  |  | «23» марта 2020г. |

**Аннотация**

В выпускной квалификационной работе представлен анализ исходных данных для разработки электромеханического манипулятора с компьютерным управлением. Произведен патентно-информационный поиск, выполнен анализ современных подходов к созданию электромеханических манипуляторов, выбрана оптимальная компоновка манипулятора. Выбрано захватное устройство Произведены расчеты: полный расчёт электроприводов поворота захватного устройства, модулей выдвижения, подъема и поворота манипулятора. Разработана система управления манипулятором. Составлены алгоритмы управления манипулятором, по которым написаны программа управления и прошивки для микроконтроллеров. Выполнен подбор всех необходимых вспомогательных устройств.

Выпускная квалификационная работа содержит:

Страниц 133, таблиц 16, рисунков 37, чертежей 93, использованных источников 8.

**Annotation**

In the final qualification work, an analysis of the initial data for the development of an electromechanical manipulator with computer control is presented. A patent information search and an analysis of modern approaches to the creation of electromechanical manipulators were performed. The optimal layout of the manipulator and the gripping device are selected. The calculations are made: a complete calculation of the rotation drives of the gripping device, calculations of extension, lifting and turning modules. A manipulator control system has been developed. The manipulator control algorithms are compiled, according to which the control program and firmware for microcontrollers are written. All necessary auxiliary devices have been selected.

Final qualification work includes:

Pages 133, tables 16, figures 37, drawings 93, sources 8.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 6](#_Toc43901496)

[1 АНАЛИТИЧЕСКАЯ ЧАСТЬ 7](#_Toc43901497)

[1.1 Анализ современных подходов к созданию электромеханических манипуляторов (систем компьютерного управления) 7](#_Toc43901498)

[1.2 Патентно-информационный поиск 11](#_Toc43901499)

[2 КОНСТРУКТОРСКАЯ ЧАСТЬ 20](#_Toc43901500)

[2.1 Выбор компоновки манипулятора 20](#_Toc43901501)

[2.2 Расчет захватного устройства 22](#_Toc43901502)

[2.3 Расчет привода поворота захватного устройства 24](#_Toc43901503)

[2.3.1 Энергетический расчет привода захватного устройства 24](#_Toc43901504)

[2.3.2 Расчет привода захватного устройства 26](#_Toc43901505)

[2.3.3 Устройство и принцип работы 35](#_Toc43901506)

[2.4 Расчет привода выдвижения руки манипулятора 35](#_Toc43901507)

[2.4.1 Энергетический расчет привода выдвижения руки манипулятора 36](#_Toc43901508)

[2.4.2 Расчет привода выдвижения руки манипулятора 37](#_Toc43901509)

[2.4.3 Устройство и принцип работы 38](#_Toc43901510)

[2.5 Расчет привода подъема манипулятора 38](#_Toc43901511)

[2.5.1 Энергетический расчет привода подъема манипулятора 38](#_Toc43901512)

[2.5.2 Расчет привода подъема манипулятора 40](#_Toc43901513)

[2.5.3 Устройство и принцип работы 45](#_Toc43901514)

[2.6 Расчет привода поворота манипулятора 46](#_Toc43901515)

[2.6.1 Энергетический расчет привода поворота манипулятора 46](#_Toc43901516)

[2.6.2 Расчет привода поворота манипулятора 47](#_Toc43901517)

[2.6.3 Устройство и принцип работы 68](#_Toc43901518)

[2.7 Устройство и принцип работы манипулятора 68](#_Toc43901519)

[3 СИСТЕМА УПРАВЛЕНИЯ 71](#_Toc43901520)

[3.1 Разработка функциональной схемы. Проработка архитектуры системы управления 71](#_Toc43901521)

[3.2 Разработка и выбор аппаратных средств сбора данных 72](#_Toc43901522)

[3.3 Разработка драйвера шагового двигателя 76](#_Toc43901523)

[3.3.1 Выбор основных компонентов 76](#_Toc43901524)

[3.3.2 Разработка схемы электрической принципиальной 78](#_Toc43901525)

[3.4 Разработка микропроцессорного блока 79](#_Toc43901526)

[3.4.1 Выбор основных компонентов 79](#_Toc43901527)

[3.4.2 Разработка схемы электрической принципиальной 81](#_Toc43901528)

[3.5 Разработка интерфейсных преобразователей 82](#_Toc43901529)

[3.6 Разработка схемы электрической соединений 85](#_Toc43901530)

[4 ПРОГРАММНАЯ ЧАСТЬ 89](#_Toc43901531)

[4.1 Составление алгоритмов работы 89](#_Toc43901532)

[4.1.1 Разработка алгоритма работы манипулятора 89](#_Toc43901533)

[4.1.2 Разработка алгоритма программы микроконтроллера приводов 92](#_Toc43901534)

[4.1.3 Разработка алгоритма программы технологического контроллера 95](#_Toc43901535)

[4.2 Разработка программы управления 98](#_Toc43901536)

[4.2.1 Разработка программы управления для микроконтроллеров приводов 98](#_Toc43901537)

[4.2.2 Разработка программы управления для технологического контроллера 105](#_Toc43901538)

[4.2.3 Разработка программы управления для персонального компьютера 112](#_Toc43901539)

[4.3 Инструкция оператора 126](#_Toc43901540)

[ЗАКЛЮЧЕНИЕ 130](#_Toc43901541)

[БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ 131](#_Toc43901542)

# ВВЕДЕНИЕ

В данной выпускной квалификационной работе представлена задача проектирования электрического манипулятора с компьютерным управлением. Мехатронные системы стремительно расширяют область применения. Они начинают интенсивно применяться для реализации высокотехнологичных процессов в различных отраслях промышленности, качественного улучшения характеристик широкого спектра разнообразных объектов. Манипулятор — это многозвенный механизм с приводами в каждом суставе. Предназначен для подъёма и переноса изделий, в большинстве случаев малогабаритных. Используются они в промышленности, медицине, в горном деле, в металлургии и в ядерной технике. Основу манипуляторов составляют пространственные механизмы со многими степенями свободы.

Компьютерное моделирование робототехнических систем имеет огромное значение в областях науки и техники. Огромное множество трудоёмкой физической работы человека в настоящее время заменено роботами, а компьютерные устройства, системы компьютерной математики значительно облегчили громоздкие вычисления и преобразования, сведя их к минимуму. Компьютерная модель - компьютерная программа, реализующая абстрактную модель некоторой системы. Компьютерные модели используются для получения новых знаний о моделируемом объекте или для приближенной оценки поведения математических систем, слишком сложных для аналитического исследования.

Управление манипулятором осуществляется либо при помощи оператора, либо заблаговременно написанной программой управления. Манипулятор нужен для автоматизации рабочего процесса.

Цели и задачи выпускной квалификационной работы:

1) Разработать манипулятор на базе шаговых двигателей.

2) Разработать и произвести расчеты приводов движения.

3) Разработать алгоритм работы манипулятора и реализовать данный алгоритм в виде прикладного программного обеспечения манипулятора.

4) Разработать алгоритмы работы микроконтроллеров приводов и технологического контроллера и написать программы согласно этим алгоритмам.

5) Разработать схему электрическую принципиальную манипулятора, драйвера шагового двигателя, микропроцессорного блока управления.

6) Разработать схему электрическую соединений и схему электрическую функциональную.

# 1 АНАЛИТИЧЕСКАЯ ЧАСТЬ

В аналитической части будет произведен анализ современных подходов к созданию электромеханических манипуляторов с системами компьютерного управления, тенденции развития и патентно-информационный поиск и анализ уже существующих на данный момент систем.

## 1.1 Анализ современных подходов к созданию электромеханических манипуляторов (систем компьютерного управления)

Направления развития (становления) мехатронных систем можно проследить с разных позиций. Например, одним из таких признаков является уровень интеграции (синергетического объединения) элементов, к числу которых относятся элементы механических, электромеханических, электронных и компьютерных (информационных) модулей (рисунок 1.1.1).

Синергия – это совместное действие, направленное на достижение единой цели.

Модуль –это унифицированная функциональная часть машины, конструктивно оформленная как самостоятельное изделие.



Рисунок 1.1.1 – Составные части мехатроники

В соответствии с признаком синергетического объединения можно исторически разделить мехатронные модули по уровням (определяющим ведущую тенденцию в современном машиностроении «от механики к мехатронике»).

Мехатронные модули первого уровня представляют собой объединение только двух исходных элементов. Типичным примером модуля первого поколения может служить «мотор-редуктор», где механический редуктор и управляемый двигатель выпускаются как единый функциональный элемент. Мехатронные системы на основе этих модулей нашли широкое применение при создании различных средств комплексной автоматизации производства (конвейеров, транспортеров, поворотных столов, вспомогательных манипуляторов).

Мехатронные модули второго уровня появились в 80-х годах в связи с развитием новых электронных технологий, которые позволили создать миниатюрные датчики и электронные блоки для обработки их сигналов. Объединение приводных модулей с указанными элементами привело к появлению мехатронных модулей движения, состав которых полностью соответствует введенному выше определению, когда достигнута интеграция трех устройств различной физической природы: механических, электротехнических и электронных. На базе мехатронных модулей данного класса созданы управляемые энергетические машины (турбины и генераторы), станки и промышленные роботы с числовым программным управлением.

Развитие третьего поколения мехатронных систем обусловлено появлением на рынке сравнительно недорогих микропроцессоров и контроллеров на их базе и направлено на интеллектуализацию всех процессов, протекающих в мехатронной системе, в первую очередь – процесса управления функциональными движениями машин и агрегатов.

Одновременно идет разработка новых принципов и технологий изготовления высокоточных и компактных механических узлов, а также новых типов электродвигателей (в первую очередь высокомоментных, бесколлекторных и линейных), датчиков обратной связи и информации. Синтез новых прецизионных, информационных и измерительных наукоемких технологий дает основу для проектирования и производства интеллектуальных мехатронных модулей и систем.

В дальнейшем мехатронные машины и системы будут объединяться в мехатронные комплексы на базе единых интеграционных платформ. Цель создания таких комплексов - добиться сочетания высокой производительности и одновременно гибкости технико-технологической среды за счет возможности ее реконфигурации, что позволит обеспечить конкурентоспособность и высокое качество выпускаемой продукции на рынках XXI века.

Наряду с синергетическим объединением (интеграцией) элементов мехатронных систем другими квалификационными признаками развития мехатронных систем, по-видимому, следует признать их интеллектуализацию и миниатюризацию.

В настоящее время в мехатронных системах объем функций распределен между механическими, электронными и компьютерными компонентами практически поровну. При этом доля компьютерной части возросла за последнее 10-летие вдвое, и есть все основания прогнозировать сохранение этой тенденции в технике будущего.

Принципиально важно подчеркнуть, что тенденция перехода от механических к мехатронным технологиям в современном машиностроении не «закрывает» механику. Наоборот, это стимулирует ее развитие к интеграции с интеллектуальными компонентами в рамках единой мехатронной системы.

Системный подход диктует новые требования к встроенным механическим и гибридным компонентам, что в свою очередь ведет к развитию новых технологий и конструкторских решений в области механики.



Рисунок 1.1.2 – Динамика изменения составляющих "Механика - электроника – информатика"

На рисунке 1.1.2 представлен характерный график (по данным журнала, выпущенного специально к Всемирной выставке «Промышленность, автоматизация и инновации» 2000 г.), который отражает динамику этого процесса в производ­ственных машинах за 30 лет, начиная с 70-х годов. Анализ показывает, что еще в начале 90-х годов подавляющее большинство функций машины (более 70 %) реализовывалось механическим путем. В последующие десятилетия происходило постепенное вытеснение механических узлов - сначала электронными, а затем и компьютерными блоками.

Интеллектуальные системы управления – основа интеллектуального управления сложными динамическими объектами (мехатронными системами). Они априорно ориентированы на работу в условиях неполноты или нечеткости исходной информации, неопределенности внешних возмущений и среды функционирования, требуют привлечения нетрадиционных подходов к управлению с использованием специальных методов и технологий искусственного интеллекта совершенно нового класса. К числу таких технологий можно отнести информационные технологии на основе экспертных систем, нейросетевых структур, ассоциативной памяти, нечеткой логики.

С течением времени развитие таких разделов искусственного интеллекта как инженерия знаний, компьютерная логика и лингвистика, методы и модели обучения, методы поиска и принятия решений и др. заложило теоретическую основу для создания высокоэффективных систем по обработке и использованию знаний для решения целого ряда прикладных задач, включая разработку систем, моделирующих творческие возможности человека. Такие системы и стали называть интеллектуальными.

В историческом аспекте, ранее при создании машин сначала выполняли базовое проектирование (с точки зрения физики и механики), а затем в качестве подсистемы разрабатывали системы управления. При этом обычные машины обладали свойством самоустойчивости и могли работать даже без информационных систем. Затем был разработан аппарат конфигурируемого управления – АКУ (*CCV – control configured vehicle*), согласно которому более эффективную машину можно спроектировать, если еще на этапе базового проектирования заложить основы системы управления (или хотя бы предположить существование такой системы). В отличие от обычной, спроектированная на основе использования концепции АКУ машина может реализовать свои функции только благодаря существованию системы управления. В случае АКУ существует механическая неустойчивость, которая часто оказывается необходимой для обеспечения соответствующих динамических характеристик.

Устойчивость достигается благодаря наличию контуров управления. В настоящее время концепцию АКУ широко используют в самолетостроении.

Для реализации концепции АКУ, несомненно, неизбежным является существование информационной системы и системы управления. В этом смысле концепцию АКУ можно отнести к основным руководящим принципам мехатроники и считать, что она относится к понятиям самого высокого ранга. Однако при этом необходимо иметь в виду, что концепция АКУ предъявляет повышенные требования к надежности информационных и управляющих систем.

Интерес к мехатронике, которая изначально понималась как некий симбиоз механики и электроники, в значительной степени связан с пересмотром ее фундаментальных основ с учетом принципов интеллектуального управления. Эта современная трактовка предполагает, что основой мехатронных систем, машин и т. д. является интеллектуальный мехатронный модуль (ИММ) или, точнее, конечный набор таких модулей.

Развитие концепции мехатронно-модульного построения сложных электромеханических систем различной конструкции и назначения предполагает комплексное решение целого ряда важнейших проблем. К их числу в первую очередь следует отнести разработку быстродействующих алгоритмов управления, обеспечивающих инвариантность к различного рода возмущениям (что особенно важно для прецизионных систем), принципов динамической развязки быстродействующих приводов с учетом компенсации взаимного влияния отдельных степеней подвижности, алгоритмов управления движением, инвариантных к типу кинематической схемы многозвенного мехатронно-модульного механизма, а также универсальных средств программирования, обеспечивающих возможность постановки прикладных задач на уровне описаний конечных технологических целей.

В наши дни управление на основе анализа внешних ситуаций (событий) остается одной из ключевых идей интеллектуального управления техническими и организационными системами. Другой базовой идеей является использование средств современной информационной технологии обработки знаний при поиске управленческих решений и формировании соответствующих управляющих воздействий.

По мере совершенствования систем, ориентированных на хранение, пополнение, обработку и использование знаний, начали создаваться системы, в которых результаты принятия решений приближаются по качеству к решениям, принятым человеком-оператором, а по скорости получения решений существенно превышают время реакции человека (особенно в непредсказуемых и непредвиденных ситуациях).

Возникла идея активизировать деятельность систем путем включения в их состав специальных дополнительных блоков формирования управляющих воздействий на основе принятых решений. Такие интеллектуальные системы, непосредственно подключенные к объекту, получили название «активные системы».

Интеллектуальные системы в последнее время стали весьма распространенным коммерческим продуктом, находящим широкий спрос пользователей-специалистов в самых разнообразных областях инженерно-технической и научно-технической сфер деятельности.

## 1.2 Патентно-информационный поиск

Учебный электромеханический робот с программным управлением.

Номер патента: 75086

Класс патента: *B25J9/00*

Заявка: 2008101234/22, 2008.01.09

Опубликовано: 2008.02.07

Авторы: Панов Сергей Сергеевич (*RU*)

Анфалов Владимир Михайлович (*RU*)

Мазеин Петр Германович (*RU*)

Патентообладатели**:** Государственное образовательное учреждение высшего профессионального образования "Южно-Уральский государственный университет" (RU)

Учебный электромеханический робот с программным управлением относится к области образования и предназначен для проведения лекционных занятий, практических, лабораторных, учебно- и научно-исследовательских работ по различным дисциплинам, связанным с технологией машиностроения, электроприводами, автоматизацией, электроавтоматикой, информационно-измерительными системами, числовым программным управлением, гибкими производственными системами, робототехникой и мехатроникой, микропроцессорными системами, вычислительной техникой и современными информационными технологиями и др.

Наиболее близким к заявляемому учебному роботу является учебный электромеханический робот, включающий шаговые электродвигатели, звенья, приводы поворота звеньев от шаговых электродвигателей, датчики нулевого положения звеньев робота, систему управления от компьютера (Информационные технологии в науке и образовании. Международная научно-практическая Интернет-конференция, ноябрь 2006 - март 2007. Железноводск. См. приложение). Движения от электродвигателей к звеньям передаются через тросики и зубчатые колеса, позиции отсчета нулевых положений датчиков могут устанавливаться вручную.

Недостатками указанного робота являются: низкая точность позиционирования, возможность нарушения положения датчиков, значительная сила трения и возможность потери управляющих импульсов, недостаточная виброустойчивость, неудовлетворительная повторяемость позиционирования из-за потери нулевого положения, а также недостатком робота является низкая грузоподъемность (не выше 1 кг). Все это существенно ограничивает применимость робота для оснащения учебных гибких производственных систем, гибких производственных модулей и автоматизированных сборочных комплексов.

Технической задачей предлагаемой полезной модели является устранение указанных недостатков, а именно, повышение точности и виброустойчивости, обеспечение повторяемости позиционирования, а также повышение грузоподъемности робота (не менее 3 кг) при сохранении габаритов и потребляемой мощности.

Для решения указанной задачи в учебном электромеханическом роботе с программным управлением, содержащем шаговые электродвигатели, звенья робота, соединенные приводами поворота с шаговыми электродвигателями, схват, датчики нулевых положений звеньев, блок управления с компьютерной системой ЧПУ, при этом, по крайней мере, первое из звеньев соединено с двигателе посредством червячной пары, согласно предложению, приводы поворота от шаговых электродвигателей ко второму и последующим звеньям робота выполнены в виде винтовых пар, а датчики нулевого положения жестко закреплены на звеньях робота.

По другому варианту в роботе все приводы поворота от шаговых электродвигателей ко второму и последующим звеньям робота выполнены в виде червячных пар, а датчики нулевого положения жестко закреплены на звеньях робота.

Выполнение приводов поворота от шаговых электродвигателей ко второму и последующим звеньям робота в виде винтовых или червячных пар обеспечивает грузоподъемность робота до 3 кг и обеспечивают виброустойчивость робота. Жесткое закрепление датчиков на звеньях роботов исключает потерю нулевого положения звеньев и гарантирует повторяемость позиционирования.

Робот работает следующим образом. При включении блока питания и компьютера в электросеть 220 В происходит обнуление памяти и звенья робота устанавливаются в нулевые положения, определенные нулевыми метками датчиков положения. Предлагаемый робот может быть использован для проведения лекционных занятий, лабораторных, учебно- и научно-исследовательских работ по различным дисциплинам, связанным с технологией машиностроения, электроприводами, электроавтоматикой, гибкими производственными системами, робототехникой и др.

Промышленный робот

Номер патента:2654096

Класс патента: *G25J11/00*

Заявка: 2015101655, 2015.01.20

Опубликовано: 2018.05.16

Авторы**:** Литвиненко Александр Михайлович (*RU*)

Бохер Роман Михайлович (*RU*)

Патентообладатели**:** Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования "Воронежский государственный технический университет" (RU)

Изобретение относится к области промышленной робототехники и может быть использовано при проектировании роботов с внешними магнитными системами, а также может использоваться для механизации и технологических операций. Промышленный робот содержит основание, руку манипулятора, установленную на платформе с шарнирами, внешнюю магнитную систему в виде статора с полюсами и обмотками и приводы поступательного перемещения. При этом приводы поступательного перемещения расположены в зазоре между полюсами статора, каждый привод содержит активные приводные элементы в виде спирально-пространственных токопроводящих элементов, неподвижный шток, соединенный шарниром с основанием, подвижный шток, соединенный шарниром с платформой, и пружины, закрепленные на корпусе привода и подвижном штоке. Изобретение направлено на повышение зоны обслуживания и уменьшение массы подвижных частей

Это достигается тем, что промышленный робот, содержащий основание, руку манипулятора, установленную на платформу с шарнирами, внешнюю магнитную систему в виде статора с полюсами и обмотками и привода поступательного перемещения, согласно изобретению приводы поступательного перемещения расположены в зазоре между полюсами статора, при этом каждый из привод содержит активные приводные элементы в виде спирально-пространственных токопроводящих элементов, неподвижный шток, соединенный шарниром с основанием, подвижный шток, соединенный шарниром с платформой, и пружины, закрепленные на корпусе привода и подвижном штоке.

Изобретение иллюстрируется чертежами, где на рисунке 1.2.1 показан заявленный робот, вид спереди в разрезе. Схема соединения основания с платформой в развертке изображена на рисунке 1.2.2. На рисунке 1.2.3 представлена конструкция привода поступательного перемещения.

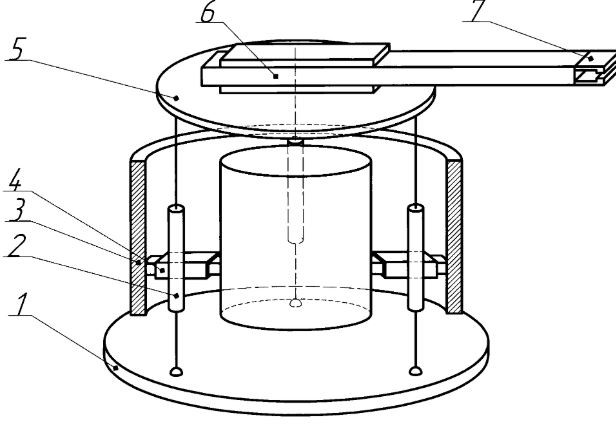


Рисунок 1.2.1 – Промышленный работа

Промышленный робот содержит основание 1, привод поступательного перемещения 2, статор с полюсами 3 и обмотками 4, платформу 5, на которой установлена рука 6 с схватом 7. Привода поступательного перемещения 2 соединены с платформой 5 с помощью подвижных штоков 8, 9, 10, соединяющиеся с платформой 5 шаровыми шарнирами 11, 12, 13.

Привод поступательного перемещения 2 имеет неподвижный шток 14, который соединен шаровыми шарнирами 15, 16, 17 с основанием 1. Привод поступательного перемещения содержит активный элемент, выполненный в виде спирально-пространственного элемента, содержащего ветвь 18 и ветвь 19. В месте их перегиба ветви 18 и 19 соединены с подвижным штоком 20. Так же конструкция привода поступательного перемещения содержит две пружины 21 и 22 закрепленные на корпусе привода 23 и подвижном штоке 20. Втулки с изоляцией 24 и 25 закреплены на корпусе 23, кабель 26 закреплен к неподвижному штоку 14.

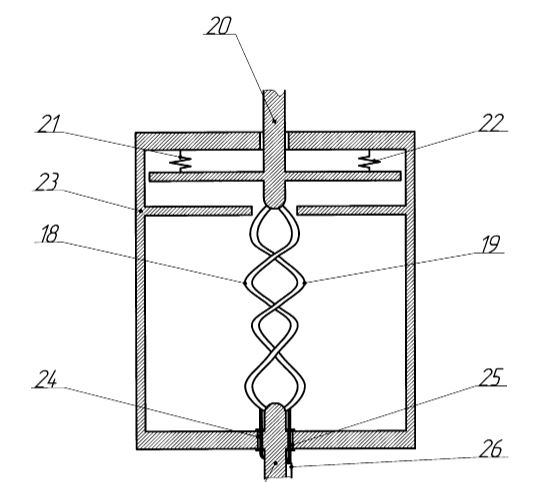
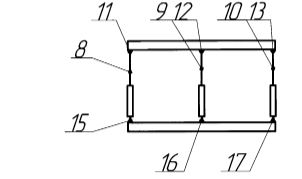


Рисунок 1.2.2 и Рисунок 1.2.3 - Схема соединения основания с платформой в развертке.

Промышленный робот работает следующим образом. Для перемещения платформы в требуемом направлении, необходимо подключить к источнику электрической энергии (не показан) обмотку 4 статора, который создаст магнитный поток в зазоре между полюсами 3, и привод поступательного перемещения 2. При включении источника электрической энергии ток в приводе поступательного перемещения протекает по активному элементу, представленному в виде спирально-пространственного токопроводящего элемента, при этом возникают электромагнитные поля. Так как ток в ветвях 18 и 19 имеет разное направление, то за счет магнитных полей происходит взаимоотталкивание ветвей, вследствие чего подвижный шток перемещается вниз. Расположение привода поступательного перемещения в зазоре между полюсами увеличивает тяговое усилие привода за счет воздействия внешних магнитных полей. Пружины 21 и 22 служат для возвратно поступательного движения привода, а электрический ток поступает к приводу поступательного перемещения по кабелю 26, через втулки с изоляцией 24, 25.

Введение основания, руки манипулятора, установленную на платформу с шарнирами, внешней магнитной системы и активных приводных элементов, позволяет увеличить зону обслуживания робота, а активные элементы выполнены в виде спирально-пространственных токопроводящих элементов, расположенных между платформой и основанием, позволяют снизить массогабаритные показатели робота.

Роботизированный сборочный стенд с компьютерным управлением

Номер патента: 78350

Класс патента: *G09B19/00*

Заявка: 2008111975/22, 2008.03.28

Опубликовано: 2008.11.20

Авторы: Смирнов Владимир Алексеевич (*RU*)

Петрова Лина Николаевна (*RU*)

Сайфутдинов Станислав Ровельевич (*RU*)

Мазеин Петр Германович (*RU*)

Патентообладатели**:** Государственное образовательное учреждение высшего профессионального образования "Южно-Уральский государственный университет" (*RU*)

Роботизированный сборочный стенд с компьютерным управлением относится к области образования, а именно к учебной технике и для научно-исследовательских работ по различным дисциплинам, связанным с технологией машиностроения, информационно-измерительными системами, числовым программным управлением, гибкими производственными станочными и сборочными системами, робототехникой и мехатроникой, микропроцессорными системами и др.

Технический результат - расширение технологических возможностей стенда за счет исключения предварительной ориентации сборочных элементов, повышения быстродействия операций схватывания и обеспечения точности сборки. Стенд снабжен установленными на основании роботом и видеокамерой, закрепленной с возможностью перемещения относительно поверхности основания и взаимосвязанной единой системой управления с роботом, маркерами системы координат основания, при этом на основании выполнена контрастная по цвету к его поверхности зона для размещения сборочных элементов, установлен контейнер для сборочных элементов и закреплены масштабные линейки.

Технической задачей предлагаемой полезной модели является устранение указанных недостатков, а именно, расширение технологических возможностей стенда за счет исключения предварительной ориентации сборочных элементов, повышения быстродействия операций схватывания и обеспечения точности сборки, возможности распознавать и выбирать сборочные элементы по конфигурации или другому признаку, например, цвету, удобства эксплуатации стенда, сокращение времени на наладку. возможность программирования процесса сборки известным методом "обучения" и, следовательно, расширение дидактических возможностей для подготовки специалистов по автоматизации сборки, робототехнике и мехатронике.

Указанная задача решается тем, что роботизированный сборочный стенд с компьютерным управлением, содержащий расположенные на основании электромеханический робот, соединенный с персональным компьютером единой системой управления, согласно предложению, он снабжен установленными на основании видеокамерой, закрепленной с возможностью перемещения относительно поверхности основания и взаимосвязанной с роботом единой системой управления и маркерами системы координат основания, при этом на основании выполнена контрастная по цвету к его поверхности зона для размещения сборочных элементов, установлен контейнер для сборочных элементов и закреплены линейки.

Особенностью заявляемого роботизированного сборочного стенда является создание системы технического зрения в виде видеокамеры и маркеров, физически определяющих систему координат основания для программы обработки видеоинформации, что позволяет осуществлять распознавание и выбор сборочных элементов без предварительной их ориентации. Взаимосвязь систем координат основания, робота и единой системы управления, обеспечивается по видеоизображению маркеров и распознавании их системой управления как базовых точек системы координат основания, что позволяет выполнять процесс распознавания сборочных элементов, их выбор, захват роботом и перемещение в место сборки в соответствии с управляющей программой пользователя, как в технологическом процессе сборки, так и в процессе обучения и подготовки специалистов по проектированию, наладке, программированию и управлению мехатронными системами.

Выполнение на основании контрастной по цвету к его поверхности зоны для размещения сборочных элементов из контейнера обеспечивает надежность распознавания, расположенных в контрастной зоне обзора сборочных элементов, а с помощью маркеров (выполненных, например, в виде светодиодов) обеспечивается привязка системы координат робота к системе координат основания и к системе координат единой системы управления, определяются координаты сборочных элементов. Полученная системой управления видеоинформация о конфигурации и положении всех размещенных в зоне обзора основания сборочных элементов и положении системы координат основания в общей систем координат, обрабатывается для последующей реализации роботом сборочного процесса в соответствии с управляющей программой, введенной в компьютер. Размещение на основании стенда контейнера и линеек обеспечивает удобство работы со стендом и сокращение времени на наладку процесса автоматизированной сборки. Применение линеек облегчает также выполнение программирования процесса сборки известным методом "обучения".

Стенд работает следующим образом. Сборочные элементы достаются из контейнера и располагаются на основании в зоне контрастного цвета в произвольном порядке и произвольном количестве (ограничение - возможность схвата робота взять деталь). Контрастная зона 2 может быть образована расположенным на основании 1 стенда ковриком. В начале работы по команде управляющей программы видеокамера передает изображение расположенных на коврике сборочных элементов и маркеров в компьютер системы управления, по маркерам устанавливается связь системы координат основания и робота с общей системой координат системы управления, распознаются сборочные элементы и определяются их координаты. После обработки в компьютере данных, полученных с помощью видеокамеры, робот осуществляет движение своих звеньев согласно управляющей программе, перемещает схват к нужному сборочному элементу, захватывает его и помещает в место сборки, затем схват разжимает сборочный элемент, оставляя его в месте сборки. Далее происходит перемещение в место сборки последующих элементов, как это предписывает управляющая программа для сборки данного узла. Располагая на контрастном коврике другие сборочные элементы, вводя другую управляющую программу пользователь может осуществлять расстановку сборочных элементов в другом месте и в другом порядке. Наличие контейнера на основании стенда позволяет содержать сборочные элементы вблизи контрастной зоны. Линейки могут использоваться для программирования процесса автоматизированной сборки методом "обучения".

Промышленная применимость. Основное назначение роботизированного сборочного стенда с элементами технического зрения: оснащение лекционных, лабораторных и практических работ по робототехнике и мехатронике, гибким производственным системам, гибким производственным модулям, роботизированным сборочным комплексам, в том числе, с техническим зрением и др. Роботизированный стенд может быть использован для изучения возможностей и алгоритмов технического зрения, изучения робота, его наладки и программирования, а также для наладки, программирования и моделирования процесса сборки изделий, кроме того, обеспечивается возможность тренажа навыков наладчика и программиста оборудования с числовым программным управлением, в том числе, мехатронного, расширяются знания в области применения современных информационных технологий в образовательном процессе и автоматизации производства. Кроме того, стенд может применяться в условиях автоматизированного производства для сборки изделий со сборочными элементами с массой до 0,3 кг.

Выводы по аналитической части

В данной главе был произведен анализ современных подходов к созданию электромеханических манипуляторов с использованием компьютерного управления было показано развитие мехатроники в разные периоды развитие, описаны базовые ее принципы и современные подходы и решение, используемые в данный момент. Был описан принцип синергетического объединения различных составляющих мехатронных систем.

Также был произведен патентно-информационный поиск, в результате которого были рассмотрены и проанализированы три патента схожих мехатронных систем.

# 2 КОНСТРУКТОРСКАЯ ЧАСТЬ

В конструкторской части будут рассмотрены характеристики электромеханического манипулятора и выбор его компоновки. Также будут произведены геометрические расчеты, расчеты на прочность, в том числе с использованием программного обеспечения КОМПАС 3D, энергетические расчет для всех приводов, из которых состоит манипулятор, описано их устройство и принцип работа, разработаны сборочные чертежи

## 2.1 Выбор компоновки манипулятора

Характеристики объекта автоматизации:

– номинальная грузоподъемность, кг 2;

– число степеней подвижности 4;

– наибольшие линейные перемещения, мм

по вертикальной оси 250;

по горизонтальной оси 250;

– наибольшее угловое перемещение, град

руки относительно вертикальной оси 315;

схвата относительно горизонтальной оси 180;

– диапазон скорости линейных перемещений, м/с:

по вертикальной оси 0.5;

по горизонтально оси 0.5;

– диапазон скорости угловых перемещений, град/с:

руки относительно вертикальной оси 100;

схвата относительно горизонтальной оси 60;

– масса, кг 30;

– габаритные размеры манипулятора, мм 700х625х405.

На этапе общей компоновки манипулятора выбирают количество, вид и взаимное расположение его степеней подвижности. С компоновкой манипулятора прежде всего связаны такие эксплуатационные характеристики робота, как форма, расположение и размеры рабочего пространства и рабочей зоны. Рабочее пространство характеризует ту часть объема, которую занимает робот и, следовательно, в которой не могут находиться остальное оборудование технологического комплекса и строительные конструкции. Рабочая зона характеризует досягаемость роботом тех или иных точек пространства.

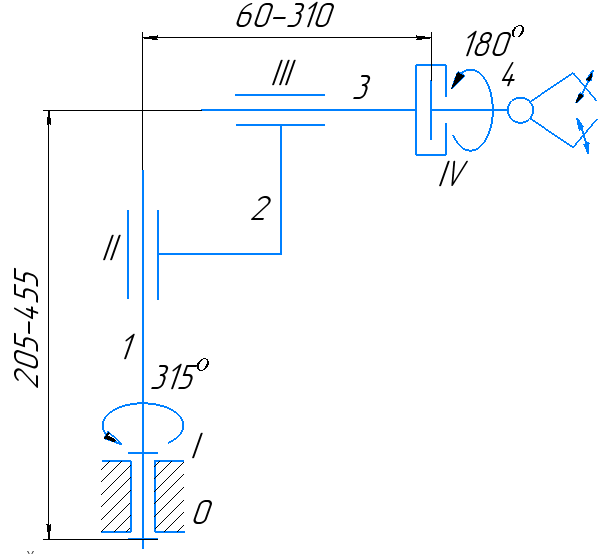


Рисунок – 2.1.1 - Структурная схема манипулятора

Выбираем 4 степени подвижности манипулятора. На рисунке 2.1.1 представлена структурная схема манипулятора [6].

0 – недвижимое звено

I, IV – вращательная (ротация) 5 класс

II, III – поступательная 5 класс

Число степеней подвижности W:

|  |  |
| --- | --- |
|  | (2.1.1) |

где *n* – число подвижных звеньев;

*Р5, Р4, Р3, Р2, Р1* – число кинематических пар, соответственно, пятого, четвертого, третьего, второго и первого классов [1].

В процессе общего проектирования манипулятора выбирают его компоновочную схему, определяют нагрузку на модули и точностные требования к ним. Затем поэтапно проектируют рабочий орган робота и модули. К основным задачам проектирования модуля относятся выбор состава, определение взаимодействия его основных функциональных элементов и конструктивных параметров деталей и агрегатов.

Манипулятор состоит из:

* захватное устройство;
* привод поворота ЗУ;
* привод горизонтального перемещения;
* привод вертикального перемещения;
* привод поворота;
* блок управления;
* блок питания;
* датчики углового перемещения.

Назначение составных частей:

Захватное устройство служит для захвата и удержания перфорированных пластин, заготовок со сложным лазерным раскроем, листового металла с вырезами и отверстиями, и аналогичных объектов.

Привод поворота ЗУ служит для поворота захватного устройства относительно горизонтальной оси и возможности захвата деталей с более сложной структурой.

Привод горизонтального перемещения служит для перемещения ЗУ в горизонтальной плоскости по направлению к основанию манипулятора или от него.

Привод вертикального перемещения служит для подъема или опускания ЗУ.

Привод поворота служит для поворота всего манипулятора относительно вертикальной оси.

Блок управления служит для обработки команд, полученных с ПК и их выполнения на каждом двигателе.

Блок питания необходим для питания всех устройств манипулятора.

Датчики углового перемещения служат для контроля над перемещениями каждого звена манипулятора.

## 2.2 Расчет захватного устройства

В качестве захватного устройства для манипулятора был выбран магнитный схват копании Schmalz модель *SGM-HP-20-G1/8-IG*. Он применяется для переноса перфорированных пластин, заготовок со сложным лазерным раскроем, листового металла с вырезами и отверстиями, и аналогичных объектов.

Характеристики данного схвата, согласно документации:

* максимальная сила удержания без фрикционного кольца, при толщине удерживаемой детали 1мм 28Н;
* максимальная сила удержания с фрикционным кольцом, при толщине удерживаемой детали 1мм 19Н;
* сила удержания без фрикционного кольца, при толщине удерживаемой детали 2мм 25Н;
* сила удержания с фрикционным кольцом, при толщине удерживаемой детали 2мм 16Н.

Габаритные размеры ЗУ указаны на рисунке 2.2.1.

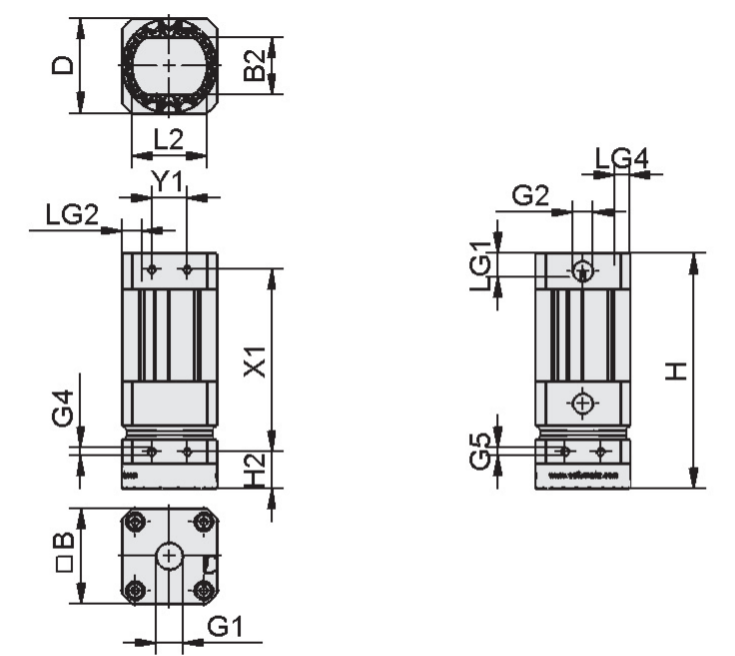


Рисунок 2.2.1 – Габаритные размеры магнитного захватного устройства *SGM-HP-20-G1/8-IG*

Масса транспортируемой детали 2кг.

Ускорение горизонтального движения [3]:

|  |  |
| --- | --- |
|  | (5.1.1) |

где – скорость перемещения;

– начальная скорость перемещения;

– время перемещения.

Сила тяжести:

|  |  |
| --- | --- |
|  | (5.1.2) |

где m - масса детали, кг;

g – ускорение свободного падения.

.

Сила инерции:

|  |  |
| --- | --- |
|  | (5.1.3) |

где – ускорение, м/с2.

Результирующая сила:

|  |  |
| --- | --- |
|  | (5.1.4) |

.

Результирующая сила меньше максимальной силы захвата у ЗУ.

## 2.3 Расчет привода поворота захватного устройства

В приводе поворота захватного устройства вращение двигателя через червячную передачу преобразуется во вращение ЗУ относительно горизонтальной оси.

2.3.1 Энергетический расчет привода захватного устройства

Двигатель передает момент через червячный вал к червячному колесу. При этом момент увеличивается на передаточное число, поэтому для выполнения условий используем шаговый двигатель FL20STH42-0804A который имеет крутящий момент M=55 Hм. Габаритные размеры приведены на рисунке 2.3.1.1. Техническая характеристика представлена в таблице 1.

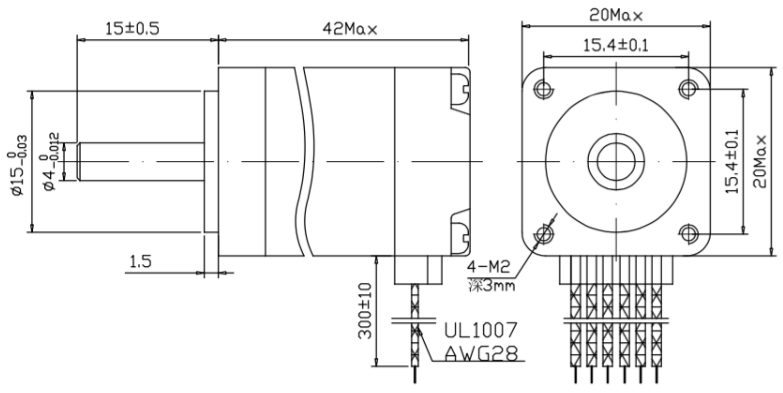


Рисунок 2.3.1.1 – Габаритные размеры шагового двигателя *FL20STH42-0804A*

На рисунке 2.3.1.2 представлена электрическая схема:



Рисунок 2.3.1.2 – Электрическая схема двигателя *FL57STH56-1006MB* с 4-мя выводами

Таблица 1 – Характеристики шагового двигателя *FL57STH56-1006MB*

|  |  |
| --- | --- |
| Параметры | Значения |
| Угловой шаг, град | 1.8 |
| Погрешность углового шага, % | ±5 |
| Погрешность сопротивления обмоток двигателя, % | 10 |
| Погрешность индуктивности обмоток двигателя, % | 20 |
| Рабочая температура, ºC | -20…50 |
| Максимальное радиальное биение вала двигателя, мм | 0,02 |
| Максимальное осевое биение вала двигателя, мм | 0,08 |
| Ток, А | 1 |

2.3.2 Расчет привода захватного устройства

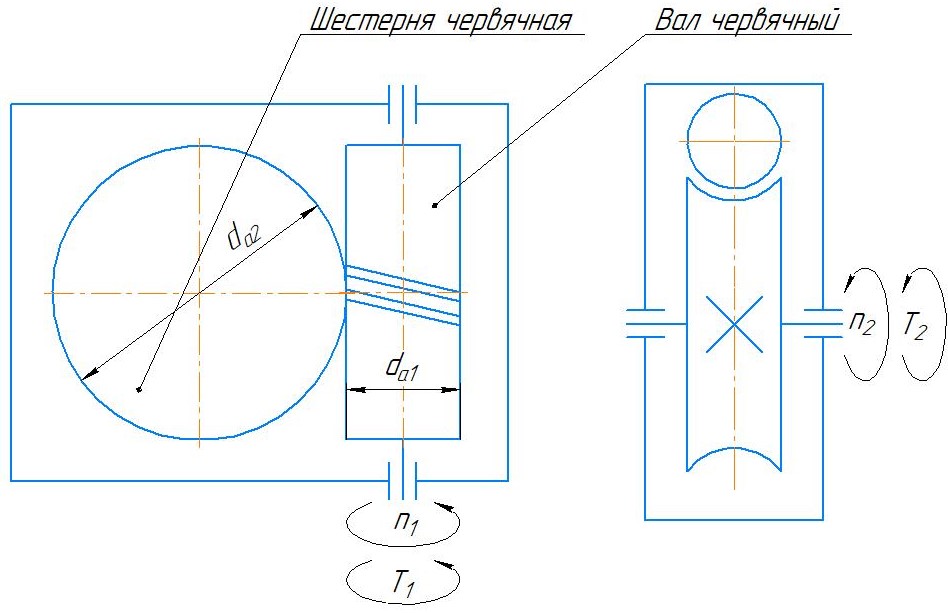


Рисунок 2.3.2.1 – Расчетная схема привода поворота захватного устройства

На рисунке 2.3.2.1 представлена расчетная схема привода поворота захватного устройства.

Выполним геометрический расчет червяной передачи.

Для червяков и колес червячных цилиндрических передач модуль *т,* мм, нормализован по ряду: 1,0; 1,25; 1,6; 2,0; 2,5; 3,15; 4,0; 5,0; 6,3; 8,0; 10,0; 12,5; 16,0; 20,0; 25,0. Выбираем модель равный 1.

Передаточное отношение:

|  |  |
| --- | --- |
|  | (2.3.2.1) |

где z1, z2 – число витков червяка и зубьев колеса соответственно. Число витков червяка принимают в зависимости от передаточного отношения передачи [2].

z1 = 1 при u > 35.5;

z1 = 2 при 18 < u < 35.5;

z1 = 4 при u < 18;

Для червячных передач номинальные значения передаточных чисел *и* стандартизованы ГОСТ2185-66 Номинальные значения передаточных чисел *и* для червячных редукторов следующие:

1-й ряд 1,00; 1,25; 1,60; 2,00; 2,50; 3,15; 4,0; 5,0; 6,3; 8,0; 10,0;12,5; 16;

2-й ряд 1,12; 1,40; 1,80; 2,24; 2,80; 3,55; 4,5; 5,6; 7,1; 9,0; 11,2; 14; 18;

Принимаем z1 = 2, z2 = 40

Размеры червячного колеса определяются по таким же расчетным зависимостям как для зубчатых колес. Для унификации стандартного инструмента, применяемого при нарезании червяков и червячных колес, отношение делительного диаметра d1 червяка к расчетному модулю m, называемое коэффициентом диаметра червяка q, нормализуют по ГОСТ 19672-74 в пределах . Рекомендуется принимать [5].

Стандартом установлено два ряда значений коэффициентов диаметра червяка *q* [5].

1-й ряд 6,3; 8; 10; 12,5; 16; 20; 25;

2-й ряд 7,5; 9; 11,2; 14; 18; 22,4;

Делительный угол подъема витков червяка и угла наклона зубьев колеса.

|  |  |
| --- | --- |
|  | (2.3.2.2) |

Делительные и начальные диаметры червяка и колеса:

|  |  |
| --- | --- |
| , | (2.3.2.3) |
|  | (2.3.2.4) |

Диаметры вершин червяка и колеса:

|  |  |
| --- | --- |
| , | (2.3.2.5) |
|  | (2.3.2.6) |

Диаметры впадин червяка и колеса [5]:

|  |  |
| --- | --- |
|  | (2.3.2.7) |
|  | (2.3.2.8) |

*.*

В червячной передаче без смещения высота зубьев и витков:

|  |  |
| --- | --- |
|  | (2.3.2.9) |

Для передачи без смещения делительное межосевое расстояние *а* и межосевое расстояние *aw:*

|  |  |
| --- | --- |
|  | (2.3.2.10) |

Модуль червячного зацепления проверяется по зависимости:

|  |  |
| --- | --- |
|  | (2.3.2.11) |

Наибольший диаметр червячного колеса определяется по формуле:

|  |  |
| --- | --- |
|  | (2.3.2.12) |

Условный угол обхвата червяка венцом зубчатого колеса определяется из условия:

|  |  |  |
| --- | --- | --- |
|  | (2.3.2.13) |  |

где b2 – ширина венца зубчатого колеса, принимаем равной 8мм.

Длину нарезной части червяка при z1 = 2 принимают [5]:

|  |  |
| --- | --- |
|  | (2.3.2.14) |

Принимаем b1 = 13.4 мм.

В таблице 2 представлены результаты геометрического расчета зубчатой передачи, выполненные в приложении для КОМПАС-3D – Валы и механические передачи 2D

Таблица 2 – Геометрический расчёт цилиндрической червячной передачи

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование и обозначение параметра | | Червяк \*1 | Червячное \*2 колесо |
| Исходные данные | | | |
| Число витков (зубьев) | z1, z2 | 2 | 40 |
| Модуль, мм | m | 1 | |
| Коэффициент диаметра червяка | q | 10 | |
| Вид червяка | − | ZA | |
| Исходный контур | − | ГОСТ 19036-94 | |
| Угол профиля исходного контура | αx | 20°00'00" | |
| Коэффициент высоты головки витка червяка исходного контура | ha1\* | 1 | |
| Коэффициент радиального зазора у поверхности впадин червяка (исходного контура червячного колеса) | c\* | 0,2 | 0,2 |
| Коэффициент расчётной толщины витка червяка | s\* | 1,571 | |

Продолжение таблицы 2

|  |  |  |  |
| --- | --- | --- | --- |
| Коэффициент радиуса кривизны переходной кривой витка червяка | ρ\*f1 | 0,3 | |
| Степень точности | − | 7-C | 7-C |
| Коэффициент смещения червяка | x | 0 | |
| Направление линии витка | − | правое | |
| Определяемые параметры | | | |
| Межосевое расстояние, мм | aw | 25+0,045 | |
| Передаточное число | u | 20 | |
| Делительный диаметр, мм | d | 10 | 40 |
| Начальный диаметр червяка, мм | dw1 | 10 | |
| Делительный угол подъема | γ | 11°18'36" | |
| Начальный угол подъема | γw | 11°18'36" | |
| Высота витка червяка, мм | h1 | 2,2 | |
| Высота головки витка червяка, мм | ha1 | 1 | |
| Диаметр вершин, мм | da | 12 | 42 |
| Диаметр впадин, мм | df | 7,6 | 37,4 |
| Наибольший диаметр червячного колеса, мм | dae2 | 43,5 | |
| Длина нарезанной части червяка, мм | b1 | 25 | |
| Ширина венца зубчатого колеса, мм | b2 | 8 | |
| Радиус кривизны переходной кривой червяка, мм | ρf1 | 0,3 | |
| Радиус выемки поверхности вершин зубьев червячного колеса, мм | Ra2 | 4 | |
| Радиус выемки впадин червячного колеса, мм | Rf2 | 6,3 | |
| Угол скашивания торцев зубчатого венца червячного колеса | Φ2 | 44°04'45" | |
| Контролируемы и измерительные параметры | | | |
| Расчётный шаг червяка, мм | p1 | 3,142 | |
| Ход витка, мм | pz1 | 6,283 | |
| Делительная толщина по хорде витка червяка, мм | sa1 |  | |

Окончание таблицы 2

|  |  |  |
| --- | --- | --- |
| Высота до хорды витка червяка, мм | ha1 | 1,002 |
| Диаметр измерительного ролика, мм | D | 1,732 |
| Размер червяка по роликам, мм | M1 |  |
| Условие M1>da1 (возможность измерения размера по роликам) | − | выполнено |
| Делительная толщина по хорде зуба червячного колеса в нормальном сечении, мм | sa2 | 1,54-0,06 |
| Высота до хорды зуба червячного колеса, мм | ha2 | 1,015 |
| Минимальное значение коэффициента смещения червяка | xmin | -1,34 |
| Условие отсутствия подрезания  зубьв червячного колеса x≥xmin | − | выполнено |
| Максимальное значение межосевого расстояния | awmax | 26,88 |
| Торцовая толщина по хорде на поверхности вершин зуба червячного колеса | Sat2 | 0,761 |
| Минимально рекомендованное значение торцовой толщины зуба на поверхности вершин, мм | 0,6·m | 0,6 |
| Условие отсутствия заострения aw≤awmax ,  зубьв червячного колеса Sat2≥0,6·m | − | выполнено |
| Наименьшая суммарная длина контактных линий | lmin | 12,238 |
| Коэффициент перекрытия | ε | 1,847 |
| Рекомендованное минимальное значение коэффициента перекрытия | − | 1,2 |

Выполним расчет на прочность червячной передачи привода поворота схвата.

Векторы окружных скоростей червяка v1 и v2 червячного колеса составляют между собой такой же угол, как угол, под которым перекрещиваются валы передачи, т. е. обычно угол, равный 90°. Каждая из скоростей определяется по соответствующей формуле:

|  |  |
| --- | --- |
|  | (2.3.2.15) |
|  | (2.3.2.16) |

где n1 и n2 максимальная частота вращения червяка и червячного колеса соответственно.

Скорость скольжения представляет собой геометрическую разность этих скоростей и определяется по формуле:

|  |  |
| --- | --- |
|  | (2.3.2.17) |

Сила взаимодействия между витками резьбы червяка и зубьями червячного колеса может быть разложена на три взаимно перпендикулярные составляющие: окружную, осевую и радиальную силы. Окружная сила червяка Ft1, равная и направленная противоположно осевой силе колеса *Fa2* [5]:

|  |  |
| --- | --- |
|  | (2.3.2.18) |

где T1 – момент на червяке.

|  |  |
| --- | --- |
| gL, | (2.3.2.19) |

где L – плечо силы тяжести.

Окружная сила Ft2 колеса равна осевой силе червяка Fa1, но направлена противоположно ей:

|  |  |
| --- | --- |
|  | (2.3.2.20) |

Радиальная сила Fr для червяка и колеса

|  |  |
| --- | --- |
|  | (2.3.2.21) |

где α=20 – стандартный угол профиля витков червяка.

Проверочный расчет зубьев червячных колес на контактную прочность выполняется по известным геометрическим параметрам передачи [5]:

|  |  |
| --- | --- |
|  | (2.3.2.22) |

где *KHβ* – коэффициент концентрации нагрузки;

*KHν* – коэффициент динамической нагрузки;

*[σH]* – допускаемые контактные напряжения.

KHβ определяет неравномерность распределения нагрузок по длине контактных линии в результате погрешностей в зацеплении и деформации зубьев колеса и витков резьбы червяка. При постоянной нагрузке *KHβ* = 1.

*KHν* – коэффициент динамической нагрузки, учитывающий динамическую нагрузку, возникающую в зацеплении, зависит от скорости скольжения и степени точности передачи. *KHν* = 1

Выбираем материал для зубьев червяка и колеса БрАЖ9-4 ГОСТ 18175-78 литьем в кокиль. Допускаемое контактное напряжение у данного материала *[σH]* = 400МПа

Коэффициент запаса по контактному напряжению:

|  |  |
| --- | --- |
|  | (2.3.2.23) |

Результаты расчета на прочность зубчатой передачи, выполненные в приложении для КОМПАС-3D – Валы и механические передачи 2D, представлены в таблице 3

Таблица 3– Расчёт на прочность цилиндрической червячной передачи при действии максимальной нагрузки

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование и обозначение параметра | | Червяк \*1 | Червячное \*2 колесо |
| Исходные данные | | | |
| Число витков (зубьев) | z1, z2 | 2 | 40 |
| Модуль, мм | m | 1 | |
| Коэффициент диаметра червяка | q | 10 | |
| Вид червяка | − | ZA | |
| Угол профиля исходного контура | αx | 20°00'00" | |
| Степень точности | − | 7-C | 7-C |
| Коэффициент смещения червяка | x | 0 | |
| Межосевое расстояние, мм | aw | 25 | |
| Передаточное число | u | 20 | |
| Твердость поверхности червяка, HRC | − | 45 | |
| Материал венца червячного колеса | − | БрАЖ9-4 ГОСТ 18175-78 | |
| Расчетная нагрузка  (крутящий момент на червячном колесе), Н\*м | Tmax | 2.943 | |
| Частота вращения червяка, об/мин | nmax | 200 | |
| Определяемые параметры | | | |
| Скорость скольжения, м/с | vs | 0,107 | |
| Расчёт на контактную прочность | | | |
| Расчётное контактное напряжение, МПа | σHmax | 256 | |
| Допускаемое контактное напряжение, МПа | σHPmax | 400 | |
| Коэффициент запаса по контактным напряжениям | nH | 1.56 | |
| Расчёт на прочность при изгибе | | | |
| Расчётное напряжение изгиба, МПа | σFmax | 10.15 | |
| Допускаемое напряжение изгиба, МПа | σFPmax | 160 | |
| Коэффициент запаса по напряжениям изгиба | nF | 15.76 | |

2.3.3 Устройство и принцип работы

Ни рисунке 2.3.3.1 вращение двигателя 1 через муфту 2 передается на червяк 3 червячной передачи. После чего червячное колесо 4 вращает магнитное захватное устройство 5 относительно своей оси. В качестве обратной связи служат индуктивные датчики конечного положения 6 и энкодер 7.

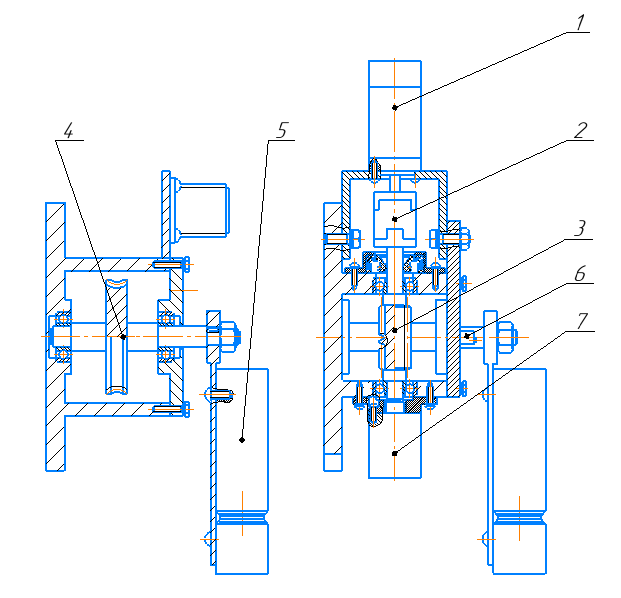


Рисунок 2.3.3.1 – Привод поворота схвата

## 2.4 Расчет привода выдвижения руки манипулятора

В приводе выдвижения руки манипулятора вращение двигателя через передачу винт-гайка качение (шарико-винтовой передачи) преобразуется в поступательное перемещение каретки, которая закреплена на приводе вертикального перемещения. В результате вращения двигателя привод вместе с захватным устройством перемещается в горизонтальном направлении.

2.4.1 Энергетический расчет привода выдвижения руки манипулятора

Двигатель передает момент через червячный вал к червячному колесу. При этом момент увеличивается на передаточное число, поэтому для выполнения условий используем шаговый двигатель *FL57STH56-1006MB* который имеет крутящий момент M=80 H·м. Габаритные размеры приведены на рисунке 2.4.1.1.

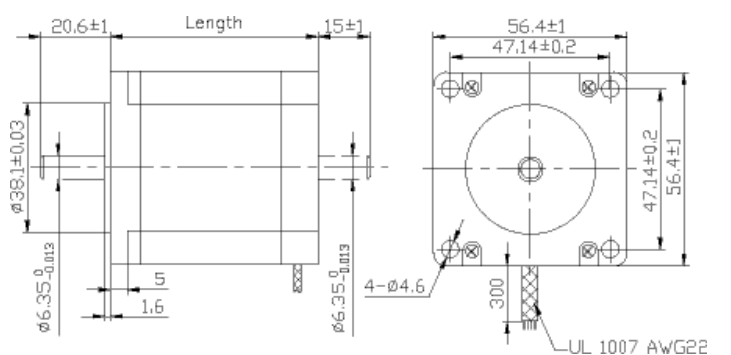


Рисунок 2.4.1.1 – Габаритные размеры шагового двигателя *FL57STH56-1006MB*

На рисунке 2.4.1.2 представлена электрическая схема:

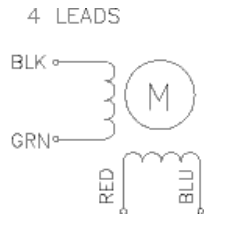


Рисунок 2.4.1.2 – Электрическая схема двигателя *FL57STH56-1006MB* с 4-мя выводами

Таблица 4 – Характеристики шагового двигателя *FL57STH56-1006MB*

|  |  |
| --- | --- |
| Параметры | Значения |
| Угловой шаг, град | 0.9 |
| Погрешность углового шага, % | ±5 |
| Погрешность сопротивления обмоток двигателя, % | 10 |
| Погрешность индуктивности обмоток двигателя, % | 20 |
| Рабочая температура, ºC | -20…50 |
| Максимальное радиальное биение вала двигателя, мм | 0,02 |
| Максимальное осевое биение вала двигателя, мм | 0,08 |
| Ток, А | 1 |

2.4.2 Расчет привода выдвижения руки манипулятора

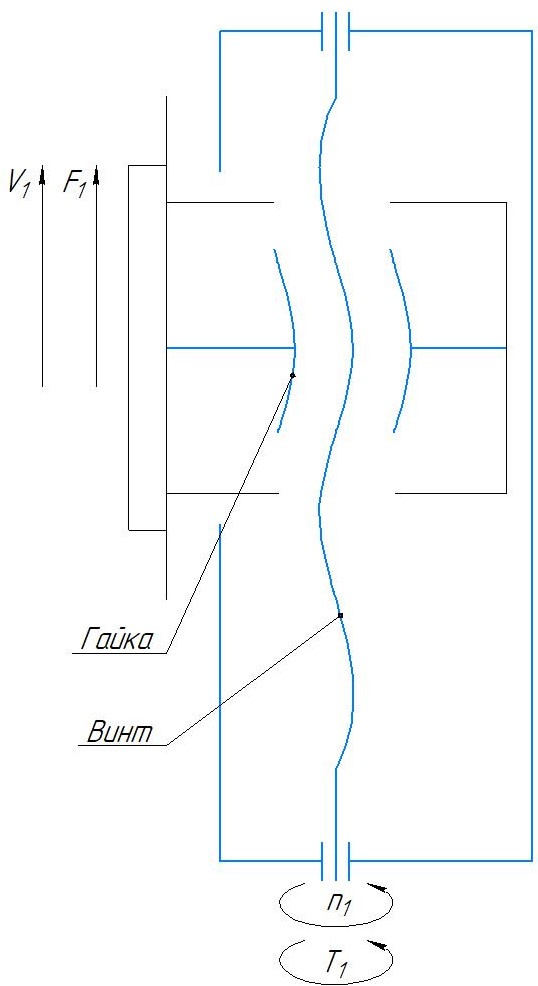


Рисунок 2.4.2.1 – Расчетная схема привода выдвижения руки манипулятора

Т.к. привод горизонтального перемещения, расчетная схема которого представлена на рисунке 2.4.2.1, по конструкции аналогичен с приводом подъема манипулятора, в нем используется передача-винт гайка с точно такими же характеристиками, какие рассчитаны в главе 2.5

2.4.3 Устройство и принцип работы

На рисунке 2.4.3.1 вращение двигателя 2 через муфту 3 передается на винт 4 шарико-винтовой передачи, преобразовываясь в поступательное движение гайки 6, которая двигает каретку 7 по направляющим 5. В качестве обратной связи служат энкодер 1 и механические датчики конечного положения 8.

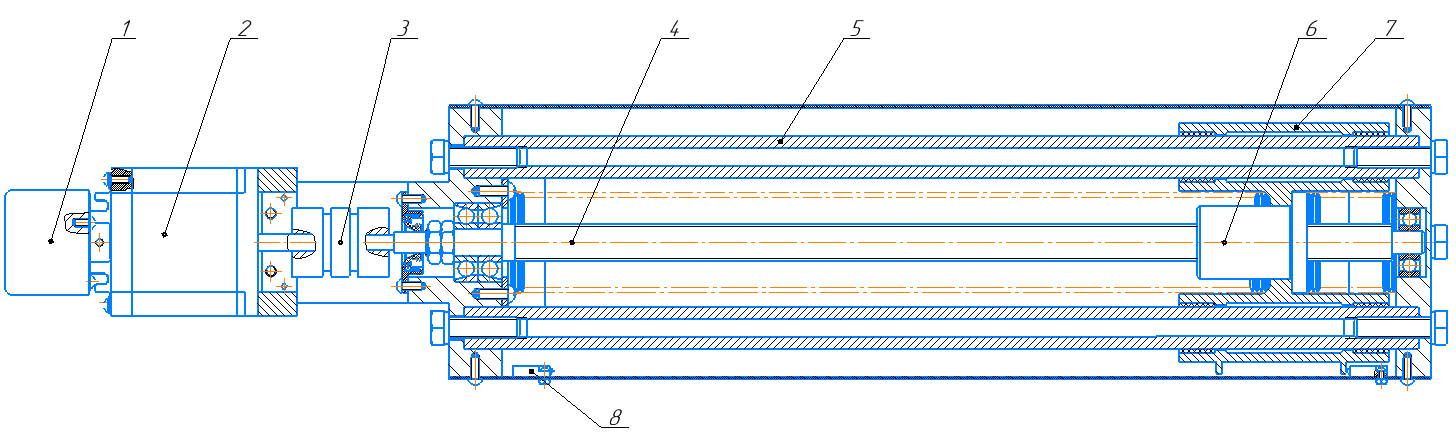


Рисунок 2.4.3.1 – Привод выдвижения руки манипулятора. Вид спереди

## 2.5 Расчет привода подъема манипулятора

В приводе вертикального перемещения вращение двигателя через передачу винт-гайка качение (шарико-винтовой передачи) преобразуется в поступательное перемещение каретки, на которой закреплен привод горизонтального перемещения. В результате вращения двигателя последний перемещается вверх и вниз по оси Z.

2.5.1 Энергетический расчет привода подъема манипулятора

Двигатель передает момент через червячный вал к червячному колесу. При этом момент увеличивается на передаточное число, поэтому для выполнения условий используем шаговый двигатель *FL57STH56-1006MB* который имеет крутящий момент *M*=80 H·м. Габаритные размеры приведены на рисунке 2.5.1.1.

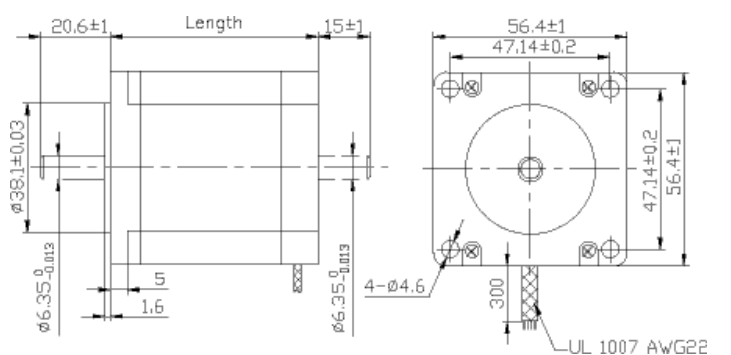


Рисунок 2.5.1.1 – Габаритные размеры шагового двигателя *FL57STH56-1006MB*

На рисунке 2.5.1.2 представлена электрическая схема:

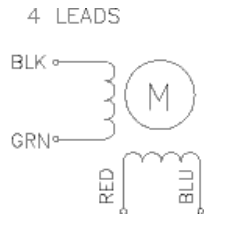


Рисунок 2.5.1.2 – Электрическая схема двигателя *FL57STH56-1006MB* с 4-мя выводами

Таблица 5 – Характеристики шагового двигателя *FL57STH56-1006MB*

|  |  |
| --- | --- |
| Параметры | Значения |
| Угловой шаг, град | 0.9 |
| Погрешность углового шага, % | ±5 |
| Погрешность сопротивления обмоток двигателя, % | 10 |
| Погрешность индуктивности обмоток двигателя, % | 20 |
| Рабочая температура, ºC | -20…50 |
| Максимальное радиальное биение вала двигателя, мм | 0,02 |
| Максимальное осевое биение вала двигателя, мм | 0,08 |
| Ток, А | 1 |

2.5.2 Расчет привода подъема манипулятора

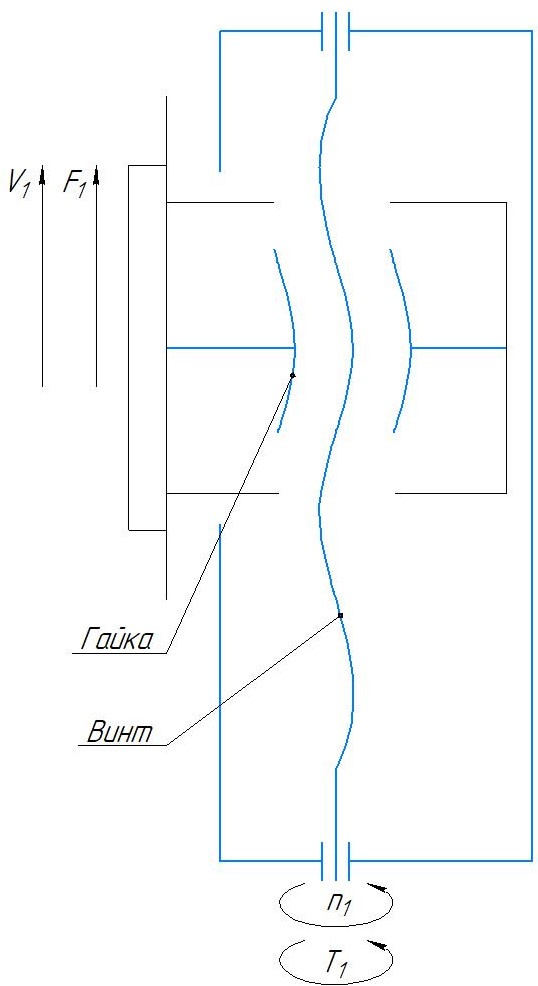


Рисунок 2.5.2.1 – Расчетная схема привода подъема манипулятора

На рисунке 2.5.2.1 представлена расчетная схема привода подъема манипулятора.

Расчет геометрических параметров ШВП при заданном передаточном отношении начинают с определения шага резьбы. При преобразовании вращательного движения вала в поступательное движение гайки шаг резьбы рассчитывают по формуле:

|  |  |
| --- | --- |
|  | (2.5.2.1) |

где *К* – число заходов резьбы, *uВП –* передаточноеотношение [3].

Рекомендуют принимать при преобразовании вращательного движения в поступательное *uВП =* 300…2000 м-1

Полученный шаг резьбы округляют до ближайшего из стандартного ряда: 1; 1,5; 2; 2,5; 3; 4; 5; 6; 8; 10; 12; 16; 20; 24. Округляем до 4.

Диаметр шариков равен [4]:

|  |  |
| --- | --- |
|  | (2.5.2.2) |

где *Кр* – коэффициент шага резьбы, равный 0,6.

Результат расчета округляют до ближайшего большего стандартного значения: 0,25; 0,30; 0,36; 0,635; 0,68; 0,7; 0,8; 0,84; 0,85; 1,0; 1,2; 1,3; 1,5; 1,588; 1,984; 2,0; 2,381; 2,5; 2,778; 3,0; 3,175; 3,5; 3,572; 4,0; 4,366; 4,5; 4,763; 5,0; 5,159; 5,5; 5,556; 5,8; 6,0; 6,35; 6,5; 6,747; 7,0; 7,144; 7,5; 7,938; 8,0; 8,334; 8,5; 8,731; 9,0; 9,575; 9,922; 10,0; 10,319; 10,716; 11,0; 11,112; 11,5; 11,509; 11,906; 12,0. Округляем до 2,5.

Внутренний диаметр винта равен:

|  |  |
| --- | --- |
|  | (2.5.2.3) |

где *КШ –* коэффициент диаметрашариков, равный 0,10…0,35.

Определяют предел текучести материала винта:

|  |  |
| --- | --- |
|  | (2.5.2.4) |

где *Fa* – осевая сила, Н;

[*n*]*Т* – требуемый коэффициент запаса по текучести, равный 1,5…2,5.

Материал винта выбирают из условия, чтобы его предел текучести был больше расчетного.

Длинные винты, работающие на сжатие, проверяют на устойчивость по условию

Допускаемая осевая сила:

|  |  |
| --- | --- |
|  | (2.5.2.5) |

где *Fкр* – критическая сила, Н, определяемая в зависимости от гибкости винта λ.

|  |  |
| --- | --- |
|  | (2.5.2.6) |

где *μ* – коэффициент приведения длины, зависящий от способов закрепления концов винта;

*l* – максимальная длина участка винта между серединами опор винта и гайки, мм;

*imin* – минимальный радиус инерции поперечного сечения винта.

Минимальный радиус инерции:

|  |  |
| --- | --- |
|  | (2.5.2.7) |

где *Jmin* – минимальный осевой момент инерции поперечного сечения винта, мм4.

*Аmin* – минимальная площадь поперечного сечения винта, мм2.

|  |  |
| --- | --- |
|  | (2.5.2.8) |
|  | (2.5.2.9) |

Если гибкость винта больше предельной λпред:

|  |  |
| --- | --- |
|  | (2.5.2.10) |

где σпц – предел пропорциональности материала винта, МПа;

*Е* = (2…2,2).105 МПа – модуль упругости первого рода материала винта, МПа.

то критическую силу определяют по формуле Л. Эйлера:

|  |  |
| --- | --- |
|  | (2.5.2.11) |

Если гибкость винта находится в пределах между начальной λ0 и предельной λпред гибкостью, то критическую силу определяют по формуле Ф.А. Ясинского:

|  |  |
| --- | --- |
|  | (2.5.2.12) |

где *а* и *b* – эмпирические коэффициенты.

Для сталей Ст.3, Ст.4, 25, 30, 40: а = 321, b = 1,16; для сталей Ст.5, Ст.6, 45, 50:а= 464, b = 3,62.

Длинные винты проверяют по критической частоте вращения:

|  |  |
| --- | --- |
|  | (2.5.2.13) |

где [*nкр*] – критическая частота вращения винта, мин-1.

|  |  |
| --- | --- |
|  | (2.5.2.14) |
|  | (2.5.2.15) |

где *k* = 0,5…0,8 – коэффициент запаса;

*ν* – коэффициент, зависящий от способов закрепления винта.

Критическую частоту вращения выбирают меньшей из двух полученных значений.

Расчет диаметра окружности расположения центров шариков, мм, ведут по формуле:

|  |  |
| --- | --- |
|  | (2.5.2.16) |

где *rж –* радиус профиля резьбы винта и гайки, мм;

*Вˈ* – смещение профиля резьбы в радиальном направлении, мм.

Для уменьшения трения радиусы *rж* должны быть больше радиуса шариков. При *dШ* ≤8 мм *rж*=0,51 *dШ*; при *dШ* >8 мм *rж*=0,53 *dШ.*

|  |  |
| --- | --- |
|  | (2.5.2.17) |

где *β –* угол контакта шариков с винтом и гайкой, град. Рекомендуют принимать близким к 45о.

Полученный результат *d*о округляют до ближайшего большего стандартного значения: 3,5; 4; 4,5; 5; 6; 8; 10; 12; 16; 20; 25; 32; 40; 50; 63; 80; 100. Округляем до 20мм.

Уточняют значение внутреннего диаметра винта:

|  |  |
| --- | --- |
|  | (2.5.2.18) |

Внешний диаметр винта, мм:

|  |  |
| --- | --- |
|  | (2.5.2.19) |

где *h*1 – глубина профиля резьбы у винта и гайки, мм.

|  |  |
| --- | --- |
|  | (2.5.2.20) |

Ширина канавки профиля резьбы, мм, равна:

|  |  |
| --- | --- |
|  | (2.5.2.21) |

Высота канавки, мм:

Число шариков в рабочей части ШВП с каналом возврата:

|  |  |
| --- | --- |
|  | (2.5.2.22) |

где *КВ* – число рабочих витков в одной замкнутой рабочей цепочке.

Округляем до ближайшего меньшего значение: zш = 50.

2.5.3 Устройство и принцип работы

На рисунке 2.5.3.1 вращение двигателя 2 через муфту 3 передается на винт 4 шарико-винтовой передачи, преобразовываясь в поступательное движение гайки 6, которая двигает каретку 7 по направляющим 5. В качестве обратной связи служат энкодер 1 и механические датчики конечного положения 8.

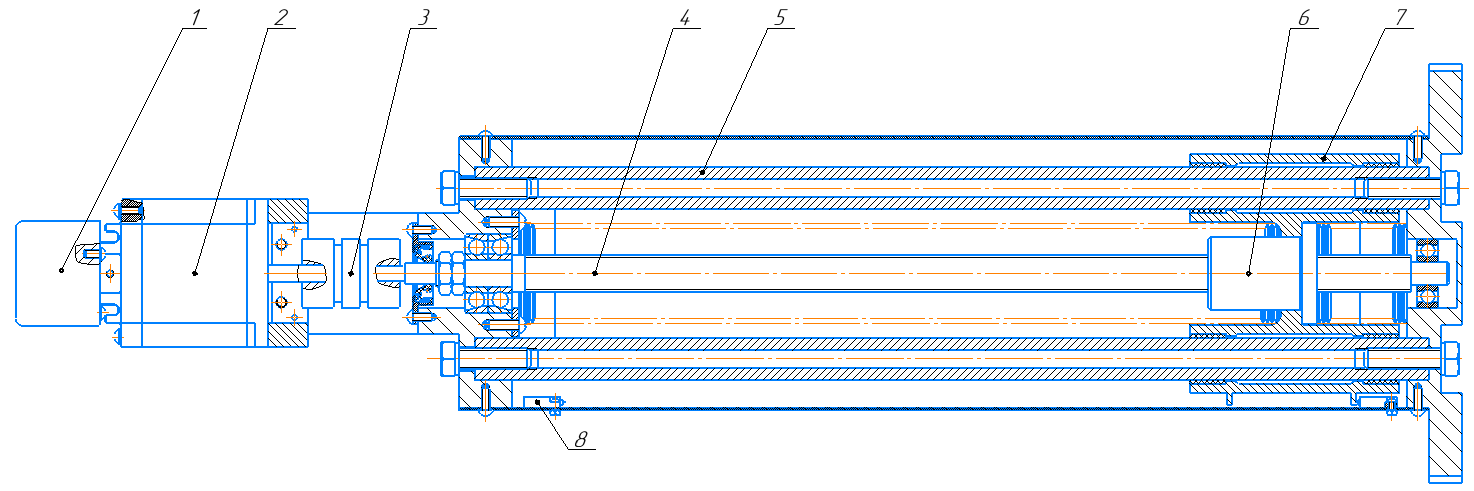


Рисунок 2.5.3.1 – Привод подъема манипулятора

## 2.6 Расчет привода поворота манипулятора

В приводе поворота вращение двигателя передается сперва через червячную передачу на зубчатую и, далее, вращается манипулятор.

2.6.1 Энергетический расчет привода поворота манипулятора

Необходимы момент двигателя равен рассчитанному моменту на червячном вале:

Двигатель передает момент через червячный вал к червячному колесу. При этом момент увеличивается на передаточное число, поэтому для выполнения условий используем шаговый двигатель *FL57STH56-1006MB* который имеет крутящий момент M=80 Hм. Габаритные размеры приведены на рисунке 2.6.1.1.

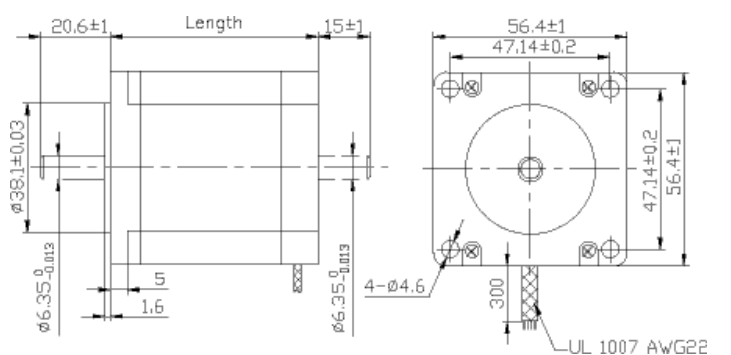


Рисунок 2.6.1.1 – Габаритные размеры шагового двигателя *FL57STH56-1006MB*

На рисунке 2.6.1.2 представлена электрическая схема:

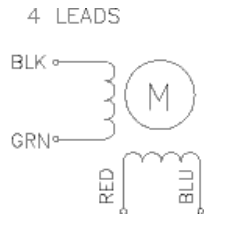


Рисунок 2.6.1.2 – Электрическая схема двигателя *FL57STH56-1006MB* с 4-мя выводами

Таблица 6 – Характеристики шагового двигателя *FL57STH56-1006MB*

|  |  |
| --- | --- |
| Параметры | Значения |
| Угловой шаг, град | 0.9 |
| Погрешность углового шага, % | ±5 |
| Погрешность сопротивления обмоток двигателя, % | 10 |
| Погрешность индуктивности обмоток двигателя, % | 20 |
| Рабочая температура, ºC | -20…50 |
| Максимальное радиальное биение вала двигателя, мм | 0,02 |
| Максимальное осевое биение вала двигателя, мм | 0,08 |
| Ток, А | 1 |

2.6.2 Расчет привода поворота манипулятора

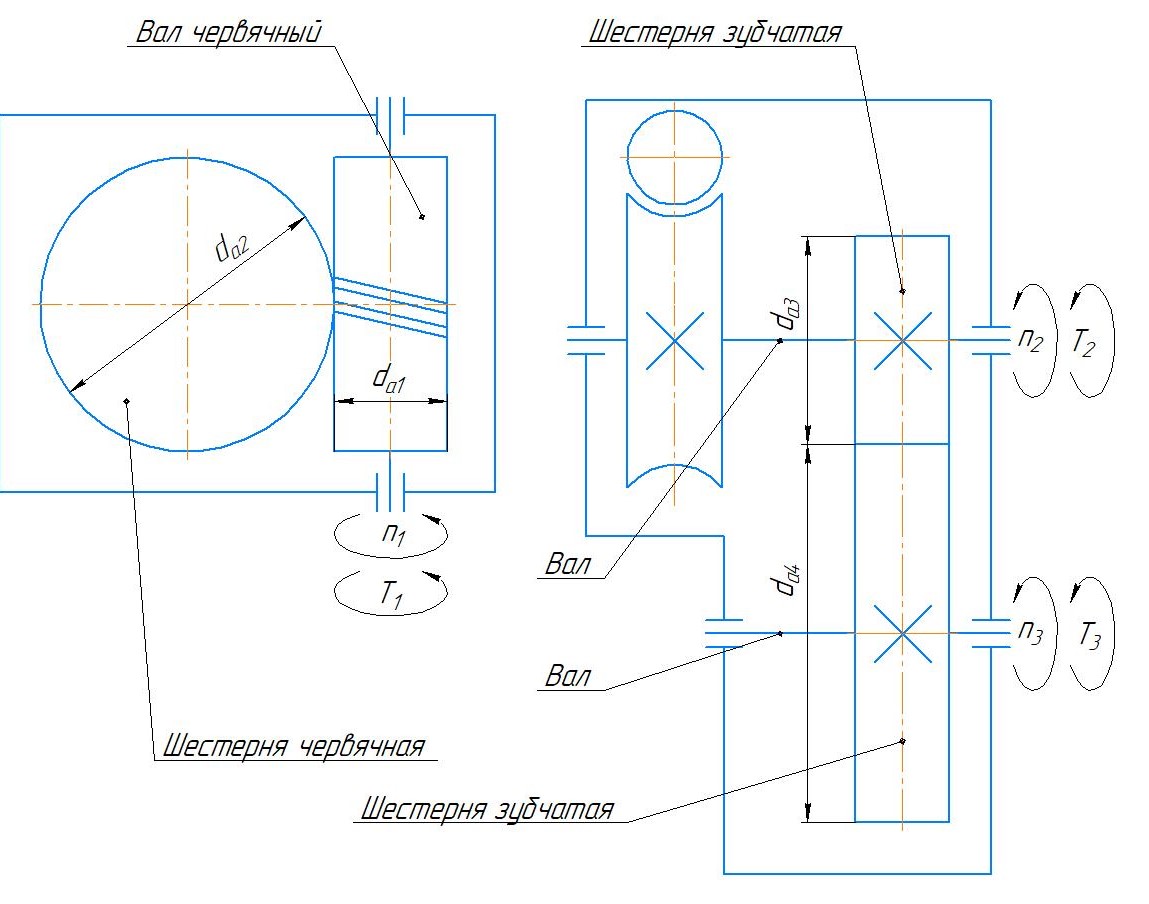


Рисунок 2.6.2.1 Расчетная схема привода поворота манипулятора

Выполним геометрический расчет зубчатой передачи привода поворота, расчетная схема которого представлена на рисунке 2.6.2.1.

Передаточное отношение:

|  |  |
| --- | --- |
|  | (2.6.2.1) |

где z1 и z2 число зубьев ведомого и ведущего колеса соответственно.

Номинальные значение передаточных чисел u зубчатых редукторов стандартизированы: 1-й ряд 1,00; 1,25; 1,60; 2,00; 2,50; 3,15; 4,0; 5,0; 6,3; 8,0; 10,0;12,5; 2-й ряд 1,12; 1,40; 1,80; 2,24; 2,80; 3,55; 4,5; 5,6; 7,1; 9,0; 11,2.

Расстояние между осями зубчатых колес цилиндрической передачи по межосевой линии называется межосевым расстоянием. Т.к. в данном случае используется зубчатое зацепление без смещения исходного контура, формула будет следующей:

|  |  |
| --- | --- |
|  | (2.6.2.2) |

где m - модуль зацепления.

M – основная характеристика размеров зубчатых и червячных колес. Для обеспечения взаимозаменяемости и унификации инструмента для изготовления, модули цилиндрических и конических эвольвентных зубчатых колес стандартизованы:.1-й ряд 1,0; 1,25; 1,5; 2; 2,5; 3; 4; 5; 6; 8; 10; 12; 16; 20; 25; 2-й ряд 1,125; 1,375; 1,75; 2,25; 2,75; 3,5; 4,5; 5,5; 7; 9; 11; 14; 18; 22; 28.

Выбираем модуль из первого ряда, равный 2

Межосевое расстояние цилиндрической зубчатой передачи, равное полусумме делительных диаметров ведомого и ведущего колеса при внешнем зацеплении, называется делительным межосевым расстоянием [4]:

|  |  |
| --- | --- |
|  | (2.6.2.3) |

Делительный диаметр:

|  |  |
| --- | --- |
|  | (2.6.2.4) |

где β – угол наклона зуба, в нашем случае 0

Начальный диаметр:

|  |  |
| --- | --- |
|  | (2.6.2.5) |

Диаметр вершин зубьев:

|  |  |
| --- | --- |
|  | (2.6.2.6) |

Диаметр впадин зубьев:

|  |  |
| --- | --- |
|  | (2.6.2.7) |

Основной диаметр:

|  |  |
| --- | --- |
|  | (2.6.2.8) |

где α угол профиля исходного контура.

Угол зацепления принимают равным углу профиля исходного контура:

|  |  |
| --- | --- |
|  | (2.6.2.9) |

В таблице 7 представлены результаты геометрического расчета зубчатой передачи, выполненные в приложении для КОМПАС-3D – Валы и механические передачи 2D.

Таблица 7 – Геометрический расчёт цилиндрической зубчатой передачи внешнего зацепления

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование и обозначение параметра | | Ведущее \*1 колесо | Ведомое \*2 колесо |
| Исходные данные | | | |
| Число зубьев | z1, z2 | 20 | 63 |
| Модуль, мм | mn | 2 | |
| Угол наклона зубьев на делительном цилиндре | β | 0°00'00" | |
| Исходный контур | − | ГОСТ 13755-2015 | |
| Угол профиля исходного контура | α | 20°00'00" | |
| Коэффициент высоты головки зуба исходного контура | ha\* | 1 | |
| Коэффициент радиального зазора исходного контура | c\* | 0,25 | |
| Коэффициент радиуса кривизны переходной кривой в граничной точке профиля зуба исходного контура | ρ\*f | 0,38 | |
| Ширина зубчатого венца, мм | b | 21 | 25 |
| Коэффициент смещения исходного контура | x | 0 | 0 |
| Степень точности | − | 7-C | 7-C |
| Определяемые параметры | | | |
| Передаточное число | u | 3,15 | |
| Межосевое расстояние, мм | aw | 83+0,045 | |
| Делительный диаметр, мм | d | 40 | 126 |

Продолжение таблицы 7

|  |  |  |  |
| --- | --- | --- | --- |
| Диаметр вершин зубьев, мм | da | 44 | 130 |
| Диаметр впадин зубьев, мм | df | 35 | 121 |
| Начальный диаметр, мм | dw | 40 | 126 |
| Основной диаметр, мм | db | 37,588 | 118,401 |
| Угол зацепления | αtw | 20°00'00" | |
| Контролируемые и измерительные параметры | | | |
| Постоянная хорда, мм | sc | 2,774-0-0,,0613 | 2,774-0-0,,0817 |
| Высота до постоянной хорды, мм | hc | 1,495 | 1,495 |
| Радиус кривизны разноимённых профилей зуба в точках, определяющих постоянную хорду, мм | ρs | 8,316 | 23,023 |
| Радиус кривизны активного профиля зуба в нижней точке, мм | ρp | 1,55 | 16,951 |
| Условиеρs > ρp (возможность измерения постоянной хорды) | − | выполнено | выполнено |
| Число зубьев в длине общей нормали | zWr | 3 | 8 |
| Длина общей нормали, мм | W | 15,321-0,055-0,125 | 46,047-0-0,,0715 |
| Радиус кривизны разноимённых профилей зубьев в точках, определяющих длину общей нормали, мм | ρw | 7,66 | 23,023 |
| Радиус кривизны профиля в точке на окружности вершин, мм | ρa | 11,436 | 26,838 |
| Условиеρp<ρw<ρa (возможность измерения длины общей нормали) | − | выполнено | выполнено |
| Диаметр измерительного ролика, мм | D | 3,464 | 3,464 |
| Угол профиля на окружности, проходящей через центр ролика | αD | 24°36'37" | 21°42'20" |
| Диаметр окружности, проходящей через центр ролика, мм | dD | 41,343 | 127,437 |

Продолжение таблицы 7

|  |  |  |  |
| --- | --- | --- | --- |
| Радиус кривизны разноимённых профилей зубьев в точках контакта поверхности ролика с главными поверхностями зубьев, мм | ρm | 6,877 | 21,834 |
| Условие ρp<ρm<ρa (возможность измерения размера по роликам) | − | выполнено | выполнено |
| Размер по роликам, мм | M | 44,807-0,153-0,273 | 130,861-0,219-0,381 |
| УсловиеdD+D>da (возможность измерения размера по роликам) | − | выполнено | выполнено |
| УсловиеdD−D>df (возможность измерения размера по роликам) | − | выполнено | выполнено |
| Нормальная толщина зуба по делительной окружности, мм | sn | 3,142-0-0,0613 | 3,142-0-0,0817 |
| Проверка качества зацепления по геометрическим показателям | | | |
| Коэффициент наименьшего смещения | xmin | -0,17 | -2,685 |
| Условие отсутствия подрезания зуба исходной производящей рейкой x≥xmin | − | выполнено | выполнено |
| Радиус кривизны в граничной точке профиля зуба, мм | ρl | 0,993 | 15,7 |
| Условие отсутствия подрезания ρl≥0 | − | выполнено | выполнено |
| Диаметр положения нижней точки активного профиля зуба, мм | dp | 37,715 | 123,159 |
| Диаметр положения точки пересечения эвольвенты с переходной кривой профиля зуба, мм | dП | 37,64 | 122,494 |
| Условие отсутствия интерференции ρl≤ρp | − | выполнено | выполнено |
| Нормальная толщина зуба на поверхности вершин, мм | sna | 1,39 | 1,576 |
| Минимально рекомендованное значение нормальной толщины зуба на поверхности вершин при поверхностном упрочнении зубьев, мм | 0,4·mn | 0,8 | |

Окончание таблицы 7

|  |  |  |  |
| --- | --- | --- | --- |
| Условие отсутствия заострения sna≥0,4·mn | − | выполнено | выполнено |
| Удельное скольжение профилей зубьев в нижних точках активных профилей зубьев | θp | -4,4984 | -1,12519 |
| Коэффициент торцового перекрытия | εα | 1,675 | |
| Рекомендованное минимальное значение коэффициента торцового перекрытия | − | 1,2 | |
| Коэффициент перекрытия | εγ | 1,675 | |
| Условие отсутствия самопересечения контура выреза зуба | − | выполнено | выполнено |

Выполним расчет на прочность зубчатой передачи привода поворота.

Для расчета на прочность необходимо найти максимальный крутящий момент на ведущем колесе. Для этого необходимо найти максимальный момент на ведомом колесе.

Максимальный момент силы на ведомом колесе:

|  |  |
| --- | --- |
|  | (2.6.2.10) |

где ε – угловое ускорение манипулятора;

I ‑ момент инерции манипулятора относительно оси, проходящей через ось ведомого колеса.

|  |  |
| --- | --- |
|  | (2.6.2.11) |

где *m* – масса подвижной части манипулятор;

*rmax* – максимальное расстояние от оси ведомого колеса до центра масс подвижной части манипулятора.

|  |  |
| --- | --- |
|  | (2.6.2.12) |

где *ωmax* – максимальная скорость поворота привода;

t – время разгона привода до максимальной скорости.

.

Окружная скорость колес:

|  |  |
| --- | --- |
|  | (2.6.2.13) |

где n1 – частота вращения ведущего колеса.

Окружная сила:

|  |  |
| --- | --- |
|  | (2.6.2.14) |

Радиальная сила:

|  |  |
| --- | --- |
|  | (2.6.2.15) |

Осевая сила:

|  |  |
| --- | --- |
|  | (2.6.2.16) |

Расчетное значение контактного напряжения:

|  |  |
| --- | --- |
|  | (2.6.2.17) |

где *ZE* – коэффициент, учитывающий механические свойства материалов сопряженных зубчатых колес, Мпа. Для стальных колес ;

*Zε* – коэффициент, учитывающий суммарную длину контактных линий;

*ZH* – коэффициент, учитывающий форму сопряженных поверхностей зубьев в полюсе зацепления. Для прямозубых цилиндрических передач ZH= 2.49;

*KH* – коэффициент нагрузки;

[σH] – расчетное допускаемое контактное напряжение, Мпа.

|  |  |
| --- | --- |
|  | (2.6.2.18) |

где εα – коэффициент торцевого перекрытия

|  |  |
| --- | --- |
|  | (2.6.2.19) |

|  |  |
| --- | --- |
|  | (2.6.2.20) |

где KHν – коэффициент, учитывающий динамическую нагрузку, возникающую в зацеплении;

KHβ – коэффициент, учитывающий неравномерность распределения нагрузки по длине контактных линий;

KHα – коэффициент, учитывающий распределение нагрузки между зубьями

Коэффициент динамической нагрузки KHν определяют в зависимости от степени точности передачи, окружной скорости и вида передачи. В нашем случае, при 7 степени точности, окружной скорости < 1 и прямозубой передачи KHν = 1.047.

Значения коэффициента KHβ принимают в зависимости от коэффициента ширины венца колеса относительно диаметра , схемы передачи и твердости зубьев. В нашем случае KHβ = 5.55

Для прямозубых передач KHα = 1

Допускаемое контактное напряжение определяется по материалу, из которого изготовлены шестерни. В нашем случае это Сталь 45Х ГОСТ 4543-2016 (нормализация, отпуск) с допустимым напряжением по контакту 2332 МПа.

Условие выполнено.

Результаты расчета на прочность зубчатой передачи, выполненные в приложении для КОМПАС-3D – Валы и механические передачи 2D, представлены в таблице 8

Таблица 8 – Расчёт на прочность при действии максимальной нагрузки цилиндрической зубчатой передачи внешнего зацепления (по ГОСТ 21354-87)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование и обозначение параметра | | Ведущее \*1 колесо | Ведомое \*2 колесо | |
| Исходные данные | |  |  |  |
| Число зубьев | z1, z2 | 20 |  | 63 |
| Модуль, мм | mn |  | 2 |  |
| Угол наклона зубьев на делительном цилиндре | β | 0°00'00" | | |
| Угол профиля исходного контура | α | 20°00'00" | | |
| Ширина зубчатого венца, мм | b | 21 |  | 25 |
| Коэффициент смещения исходного контура | x | 0 |  | 0 |
| Степень точности | − | 7-C |  | 7-C |
| Вариант схемы расположения передачи | − | 5 | | |
| Марка материала | \*1 | Сталь 45Х ГОСТ 4543-2016 | | |
| \*2 | Сталь 45Х ГОСТ 4543-2016 | | |
| Твердость активных поверхностей зубьев, HRC | − | 30 |  | 30 |
| Расчётная нагрузка  (крутящий момент на ведущем колесе), Н\*м | Tmax | 3,26 | | |
| Частота вращения ведущего колеса, об/мин | n1 | 52 | | |
| Определяемые параметры | |  |  |  |
| Окружная скорость в зацеплении, м/с | v | 0,09 | | |

Окончание таблицы 8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Расчёт на контактную прочность | | | | |
| Коэффициент, учитывающий неравномерность распределения нагрузки по длине контактных линий | KHβ | 2,354 | | |
| Удельная окружная динамическая сила, Н/мм | wHv | 0,158 | | |
| Коэффициент, учитывающий динамическую нагрузку, озникающую в зацеплении до зоны резонанса | KHv | 1,02 | | |
| Окружная сила на делительном цилиндре, Н | FtH | 163 | | |
| Удельная расчетная окружная сила, Н/мм | wHt | 18,639 | | |
| Расчётное контактное напряжение, МПа | σHmax | 326 | ,965 |  |
| Допускаемое контактное напряжение, МПа | σHPmax | 2332 |  | 2332 |
| Коэффициент запаса по контактным напряжениям | nH | 7,132 | 7,132 | |
| Расчёт на прочность при изгибе | |  |  | |
| Коэффициент, учитывающий неравномерность распределения нагрузки по длине контактных линий | KFβ | 3,76 | | |
| Удельная окружная динамическая сила, Н/мм | wFv | 0,42 | | |
| Коэффициент, учитывающий динамическую нагрузку, возникающую в зацеплении до зоны резонанса | KFv | 1,054 | | |
| Окружная сила на делительном цилиндре, Н | FtF | 163 | | |
| Удельная расчетная окружная сила, Н/мм | wFt | 30,766 | | |
| Расчётное напряжение изгиба, МПа | σFmax | 63,531 | 56,601 | |
| Допускаемое напряжение изгиба, МПа | σFPmax | 1139,025 | 1127,307 | |
| Коэффициент запаса по напряжениям изгиба | nF | 17,929 | 19,917 | |

Результаты расчета на долговечность зубчатой передачи, выполненные в приложении для КОМПАС-3D – Валы и механические передачи 2D, представлены в таблице 9

Таблица 9 – Расчёт на выносливость цилиндрической зубчатой передачи внешнего зацепления (по ГОСТ 21354-87)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование и обозначение параметра | | | Ведущее \*1 колесо | Ведомое \*2 колесо | |
| Исходные данные | | | | | |
| Число зубьев | | z1, z2 | 20 |  | 63 |
| Модуль, мм | | mn | 2 | | |
| Угол наклона зубьев на делительном цилиндре | | β | 0°00'00" | | |
| Угол профиля исходного контура | | α | 20°00'00" | | |
| Ширина зубчатого венца, мм | | b | 21 |  | 25 |
| Коэффициент смещения исходного контура | | x | 0 |  | 0 |
| Степень точности | | − | 7-C |  | 7-C |
| Вариант схемы расположения передачи | | − | 5 | |  |
| Марка материала | | \*1 | Сталь 45Х ГОСТ 4543-2016 | | |
| \*2 | Сталь 45Х ГОСТ 4543-2016 | | |
| Твердость активных поверхностей зубьев, HRC | | − | 30 |  | 30 |
| Базовое число циклов перемены напряжений, миллионы | контакт | NHlim | 23,375 | 23,375 | |
| изгиб | NFlim | 4 | 4 | |
| Предел выносливости по контакту, соответствующий базовому числу циклов, МПа | | σHlimb | 640 | 640 | |
| Коэффициент ограничения усталостных повреждений (контакт) | | αHG | 0,6 | | |
| Коэффициент безопасности (контакт) | | SH | 1,1 | 1,1 | |
| Показатель кривой выносливости (контакт) | | qH | 6 | 6 | |
| Предел выносливости по изгибу, соответствующий базовому числу циклов, МПа | | σFlimb | 498 | 498 | |
| Коэффициент ограничения усталостных повреждений (изгиб) | | αFG | 0,6 | | |

Окончание таблицы 9

|  |  |  |  |
| --- | --- | --- | --- |
| Коэффициент безопасности (изгиб) | SF | 1,7 | 1,7 |
| Показатель кривой выносливости (изгиб) | qF | 6 | 6 |
| Коэффициент, учитывающий шероховатость | ZR | 1 | 1 |
| Коэффициент, учитывающий влияние двухстороннего приложения нагрузки | YA | 1 | 1 |
| Планируемый ресурс работы, час | Lp | 90000 | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Режимы нагружения передачи | | | | | | | | |
| Расчётная нагрузка  (крутящий момент на ведущем колесе), Н•м | | Частота вращения ведущего колеса, об/мин | Продолжительность работы  передачи на данном режиме, % | Число циклов нагружения, миллионы | | Контактное напряжение,  МПа | Напряжение изгиба, МПа | |
| контакт | изгиб | Ведущее колесо\*1 | Ведомое колесо\*2 |
| THi | TFi | n1i | − | Nci | | σHi | σF1i | σF2i |
| 3.26 | 3.26 | 52 | 100 % | 280.8 | | 326,965 | 63,531 | 56,601 |
| Наименование и обозначение параметра | | | | | | | Ведущее колесо\*1 | Ведомое колесо\*2 |
| Определяемые параметры | | | | | | | | |
| Допускаемые напряжения по контакту, МПа | | | | | σHP | | 581,818 | 581,818 |
| Эквивалентные напряжения по контакту, МПа | | | | | σHE | | 0 | 0 |
| Допускаемые напряжения по изгибу, МПа | | | | | σFP | | 315,375 | 312,131 |
| Эквивалентные напряжения по изгибу, МПа | | | | | σFE | | 0 | 0 |
| Ресурс по контакту, час | | | | | LH | | 99999 | 99999 |
| Ресурс по изгибу, час | | | | | LF | | 99999 | 99999 |

Выполним геометрический расчет червячной передачи.

Для червяков и колес червячных цилиндрических передач модуль *т,* мм, нормализован по ряду: 1,0; 1,25; 1,6; 2,0; 2,5; 3,15; 4,0; 5,0; 6,3; 8,0; 10,0; 12,5; 16,0; 20,0; 25,0. Выбираем модель равный 1.

Передаточное отношение:

|  |  |
| --- | --- |
|  | (2.6.2.21) |

где z1, z2 – число витков червяка и зубьев колеса соответственно.

Число витков червяка принимают в зависимости от передаточного отношения передачи:

z1 = 1 при u > 35.5;

z1 = 2 при 18 < u < 35.5;

z1 = 4 при u < 18;

Для червячных передач номинальные значения передаточных чисел *и* стандартизованы ГОСТ2185-66 Номинальные значения передаточных чисел *и* для червячных редукторов следующие [5]:

1-й ряд 1,00; 1,25; 1,60; 2,00; 2,50; 3,15; 4,0; 5,0; 6,3; 8,0; 10,0;12,5; 16;

2-й ряд 1,12; 1,40; 1,80; 2,24; 2,80; 3,55; 4,5; 5,6; 7,1; 9,0; 11,2; 14; 18;

Принимаем z1 = 2, z2 = 40

Размеры червячного колеса определяются по таким же расчетным зависимостям как для зубчатых колес. Для унификации стандартного инструмента, применяемого при нарезании червяков и червячных колес, отношение делительного диаметра d1 червяка к расчетному модулю m, называемое коэффициентом диаметра червяка q, нормализуют по ГОСТ 19672-74 в пределах . Рекомендуется принимать .

Стандартом установлено два ряда значений коэффициентов диаметра червяка *q:*

1-й ряд 6,3; 8; 10; 12,5; 16; 20; 25;

2-й ряд 7,5; 9; 11,2; 14; 18; 22,4;

Делительный угол подъема витков червяка и угла наклона зубьев колеса.

|  |  |
| --- | --- |
|  | (2.6.2.22) |

Делительные и начальные диаметры червяка и колеса:

|  |  |
| --- | --- |
| , | (2.6.2.23) |
|  | (2.6.2.24) |

Диаметры вершин червяка и колеса:

|  |  |
| --- | --- |
|  | (2.6.2.25) |
|  | (2.6.2.26) |

Диаметры впадин червяка и колеса:

|  |  |
| --- | --- |
|  | (2.6.2.27) |
|  | (2.6.2.28) |

.

В червячной передаче без смещения высота зубьев и витков:

|  |  |
| --- | --- |
|  | (2.6.2.29) |

Для передачи без смещения делительное межосевое расстояние *а* и межосевое расстояние *aw:*

|  |  |
| --- | --- |
|  | (2.6.2.30) |

Модуль червячного зацепления проверяется по зависимости:

|  |  |
| --- | --- |
|  | (2.6.2.31) |

Наибольший диаметр червячного колеса определяется по формуле:

|  |  |
| --- | --- |
|  | (2.6.2.32) |

Условный угол обхвата червяка венцом зубчатого колеса определяется из условия:

|  |  |
| --- | --- |
|  | (2.6.2.33) |

где b2 – ширина венца зубчатого колеса, принимаем равной 8мм.

Длина нарезной части червяка при z1 = 2 принимают:

|  |  |
| --- | --- |
|  | (2.6.2.34) |

Принимаем b1 = 13.4 мм.

В таблице 10 представлены результаты геометрического расчета зубчатой передачи, выполненные в приложении для КОМПАС-3D – Валы и механические передачи 2D

Таблица 10 – Геометрический расчёт цилиндрической червячной передачи

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование и обозначение параметра | | Червяк \*1 | Червячное \*2 колесо |
| Исходные данные | | | |
| Число витков (зубьев) | z1, z2 | 2 | 40 |
| Модуль, мм | m | 1 | |
| Коэффициент диаметра червяка | q | 10 | |
| Вид червяка | − | ZA | |
| Исходный контур | − | ГОСТ 19036-94 | |

Продолжение таблицы 10

|  |  |  |  |
| --- | --- | --- | --- |
| Угол профиля исходного контура | αx | 20°00'00" | |
| Коэффициент высоты головки витка червяка исходного контура | ha1\* | 1 | |
| Коэффициент радиального зазора у поверхности впадин червяка (исходного контура червячного колеса) | c\* | 0,2 | 0,2 |
| Коэффициент расчётной толщины витка червяка | s\* | 1,571 | |
| Коэффициент радиуса кривизны переходной кривой витка червяка | ρ\*f1 | 0,3 | |
| Степень точности | − | 7-C | 7-C |
| Коэффициент смещения червяка | x | 0 | |
| Направление линии витка | − | правое | |
| Определяемые параметры | | | |
| Межосевое расстояние, мм | aw | 25+0,045 | |
| Передаточное число | u | 20 | |
| Делительный диаметр, мм | d | 10 | 40 |
| Начальный диаметр червяка, мм | dw1 | 10 | |
| Делительный угол подъема | γ | 11°18'36" | |
| Начальный угол подъема | γw | 11°18'36" | |
| Высота витка червяка, мм | h1 | 2,2 | |
| Высота головки витка червяка, мм | ha1 | 1 | |
| Диаметр вершин, мм | da | 12 | 42 |
| Диаметр впадин, мм | df | 7,6 | 37,4 |
| Наибольший диаметр червячного колеса, мм | dae2 | 43,5 | |
| Длина нарезанной части червяка, мм | b1 | 25 | |
| Ширина венца зубчатого колеса, мм | b2 | 8 | |
| Радиус кривизны переходной кривой червяка, мм | ρf1 | 0,3 | |
| Радиус выемки поверхности вершин зубьев червячного колеса, мм | Ra2 | 4 | |

Окончание таблицы 10

|  |  |  |
| --- | --- | --- |
| Радиус выемки впадин червячного колеса, мм | Rf2 | 6,3 |
| Угол скашивания торцев зубчатого венца червячного колеса | Φ2 | 44°04'45" |
| Контролируемы и измерительные параметры | | |
| Расчётный шаг червяка, мм | p1 | 3,142 |
| Ход витка, мм | pz1 | 6,283 |
| Делительная толщина по хорде витка червяка, мм | sa1 |  |
| Высота до хорды витка червяка, мм | ha1 | 1,002 |
| Диаметр измерительного ролика, мм | D | 1,732 |
| Размер червяка по роликам, мм | M1 |  |
| Условие M1>da1 (возможность измерения размера по роликам) | − | выполнено |
| Делительная толщина по хорде зуба червячного колеса в нормальном сечении, мм | sa2 | 1,54-0,06 |
| Высота до хорды зуба червячного колеса, мм | ha2 | 1,015 |
| Минимальное значение коэффициента смещения червяка | xmin | -1,34 |
| Условие отсутствия подрезания  x≥x  зубьв червячного колеса min | − | выполнено |
| Максимальное значение межосевого расстояния | awmax | 26,88 |
| Торцовая толщина по хорде на поверхности вершин зуба червячного колеса | Sat2 | 0,761 |
| Минимально рекомендованное значение торцовой толщины зуба на поверхности вершин, мм | 0,6·m | 0,6 |
| Условие отсутствия заострения aw≤awmax ,  зубьв червячного колеса Sat2≥0,6·m | − | выполнено |
| Наименьшая суммарная длина контактных линий | lmin | 12,238 |
| Коэффициент перекрытия | ε | 1,847 |

Выполним расчет на прочность червячной передачи привода поворота.

Векторы окружных скоростей червяка v1 и v2 червячного колеса составляют между собой такой же угол, как угол, под которым перекрещиваются валы передачи, т. е. обычно угол, равный 90°. Каждая из скоростей определяется по соответствующей формуле:

|  |  |
| --- | --- |
|  | (2.6.2.35) |
|  | (2.6.2.36) |

где n1 и n2 максимальная частота вращения червяка и червячного колеса соответственно.

Скорость скольжения представляет собой геометрическую разность этих скоростей и определяется по формуле:

|  |  |
| --- | --- |
|  | (2.6.2.37) |

Сила взаимодействия между витками резьбы червяка и зубьями червячного колеса может быть разложена на три взаимно перпендикулярные составляющие: окружную, осевую и радиальную силы. Окружная сила червяка *Ft1*, равная и направленная противоположно осевой силе колеса *Fa2*:

|  |  |
| --- | --- |
|  | (2.6.2.38) |

где T1 – момент на червяке.

|  |  |
| --- | --- |
|  | (2.6.2.39) |

где T2 – момент на колесе

Окружная сила *Ft2* колеса равна осевой силе червяка *Fa1*, но направлена противоположно ей:

|  |  |
| --- | --- |
|  | (2.6.2.40) |

Радиальная сила Fr для червяка и колеса

|  |  |
| --- | --- |
|  | (2.6.2.41) |

где α=20 – стандартный угол профиля витков червяка.

Проверочный расчет зубьев червячных колес на контактную прочность выполняется по известным геометрическим параметрам передачи:

|  |  |
| --- | --- |
|  | (2.6.2.42) |

где *KHβ* – коэффициент концентрации нагрузки, которым определяется неравномерность распределения нагрузок по длине контактных линии в результате погрешностей в зацеплении и деформации зубьев колеса и витков резьбы червяка. При постоянной нагрузке KHβ = 1;

*KHν* – коэффициент динамической нагрузки, учитывающий динамическую нагрузку, возникающую в зацеплении, зависит от скорости скольжения и степени точности передачи. *KHν* = 1;

*[σH]* – допускаемые контактные напряжения.

Выбираем материал для зубьев червяка и колеса БрАЖ9-4 ГОСТ 18175-78 литьем в кокиль. Допускаемое контактное напряжение у данного материала *[σH]* = 400МПа

Коэффициент запаса по контактному напряжению:

|  |  |
| --- | --- |
|  | (2.6.2.43) |

Результаты расчета на прочность зубчатой передачи, выполненные в приложении для КОМПАС-3D – Валы и механические передачи 2D, представлены в таблице 11

Таблица 11 – Расчёт на прочность цилиндрической червячной передачи при действии максимальной нагрузки.

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование и обозначение параметра | | Червяк \*1 | Червячное \*2 колесо |
| Исходные данные | | | |
| Число витков (зубьев) | z1, z2 | 2 | 40 |
| Модуль, мм | m | 1 | |
| Коэффициент диаметра червяка | q | 10 | |
| Вид червяка | − | ZA | |
| Угол профиля исходного контура | αx | 20°00'00" | |
| Степень точности | − | 7-C | 7-C |
| Коэффициент смещения червяка | x | 0 | |
| Межосевое расстояние, мм | aw | 25 | |
| Передаточное число | u | 20 | |
| Твердость поверхности червяка, HRC | − | 45 | |
| Материал венца червячного колеса | − | БрАЖ9-4 ГОСТ 18175-78 | |
| Расчетная нагрузка  (крутящий момент на червячном колесе), Н\*м | Tmax | 1,035 | |
| Частота вращения червяка, об/мин | nmax | 1040 | |
| Определяемые параметры | |  | |
| Скорость скольжения, м/с | vs | 0,555 | |
| Расчёт на контактную прочность | |  | |
| Расчётное контактное напряжение, МПа | σHmax | 132,901 | |

Окончание таблицы 11

|  |  |  |
| --- | --- | --- |
| Допускаемое контактное напряжение, МПа | σHPmax | 400 |
| Коэффициент запаса по контактным напряжениям | nH | 3,01 |
| Расчёт на прочность при изгибе | |  |
| Расчётное напряжение изгиба, МПа | σFmax | 7,449 |
| Допускаемое напряжение изгиба, МПа | σFPmax | 160 |
| Коэффициент запаса по напряжениям изгиба | nF | 21,478 |

2.6.3 Устройство и принцип работы

На рисунке 2.6.3.1 вращение двигателя 2 через муфту 3 передается на червяк 5 червячной передачи. После чего с червячной шестерни 6 через вал 7 передается на зубчатую шестерню 8 и далее через зубчатую передачу с зубчатой шестерни 9 на вал 10, который вращается закрепленное на нем основание 11. В качетсве обратной связи служат энкодер 1 и механические датчики конечного положения 4 [8].

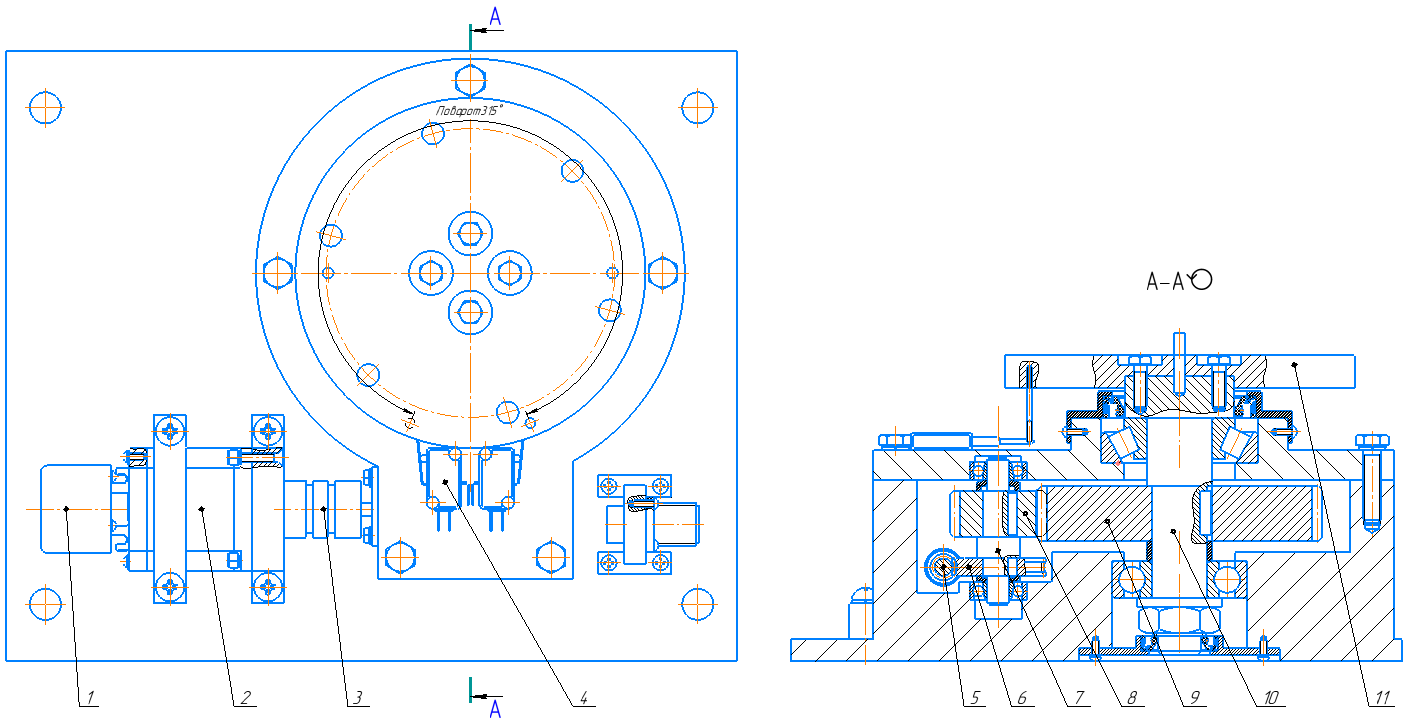


Рисунок 2.6.3.1 – Привод поворота манипулятора

## 2.7 Устройство и принцип работы манипулятора

На рисунке 2.7.1 момент от двигателя 1 передается через муфту 2 на червячную передачу, с которой момент передается на зубчатую передачу, находящиеся в приводе поворота 3. Благодаря этому происходит вращение привода вертикального перемещения 4, в котором происходит преобразование при помощи передачи винт-гайка качения (ШВП) вращательного движения двигателя 5 в постпупательное перемещение каретки, на которой закреплен привод горизонтального перемещения 6. Вращательное движение двигателя 7 через передачу винт-гайка качения (ШВП) преобразуется в постпупательное движение, в следствии чего происходит перемещения привода поворота схвата 8 в горизонтальном направлении. Вращение двигателя 9 через червячную передачу, находящуюся в приводе поворота схвата 8 передается на вращение магнитного схвата 10. На каждом приводе в обоих крайних положения установлены датчики конечного положения. Также на каждом двигателе установлен энкодер. Блок управления 11 необходим для управления электромеханическим манипулятором.

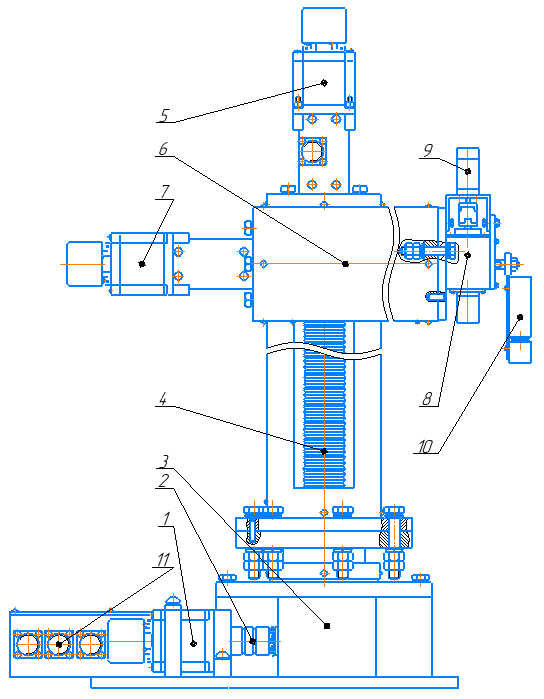


Рисунок 2.7.1 – Электромеханический манипулятор

Выводы по конструкторской части

В данном разделе был спроектирован электромеханический манипулятор на базе шаговых двигателей. Указали состав системы и назначение этих элементов в установке. Задали характеристики объекту автоматизации.

Далее выбрали компоновку манипулятора и описали составляющие части манипулятора, а также принцип его работы на производстве.

Также было выбрано захватное устройство, указаны его характеристики, проведен расчет захватного устройства. Были произведены геометрические расчеты, расчеты на прочность червячной и зубчатой передачи для привода поворота, передачи винт-гайка качения для приводов вертикального и горизонтального перемещения и червячной передачи для привода поворота схвата, в том числе с использованием программного обеспечения КОМПАС 3D. Произведя энергетический расчет для всех приводов, из которых состоит манипулятор, выбрали двигатель для каждого привода.

Разработаны сборочные чертежи для всех приводов и спецификации к ним. Разработан сборочный чертеж электромеханического манипулятора и спецификация к нему. Разработан чертеж общего вида манипулятора.

# 3 СИСТЕМА УПРАВЛЕНИЯ

В данной главе будет проработана и описана архитектура системы управления, разработана ее функциональная схема, выбраны аппаратные средства сбора данных. Будет разработан драйвер шагового двигателя, будут выбраны основные компоненты, разработана схема электрическая принципиальная. Также будет разработан микропроцессорный блок, выбраны основные компоненты для него, разработана схема электрическая принципиальная. Будут выбраны и разработаны интерфейсные преобразователи, разработана схема электрическая соединений мехатронного модуля.

## 3.1 Разработка функциональной схемы. Проработка архитектуры системы управления

Персональный компьютер последовательно выполняет интерпретацию G кода, аппроксимацию, расчет промежуточных точек, решает обратную задачу кинематики и преобразует полученные значению в число импульсов энкодера на каждом приводе. После отправляет вычисленные значению на технологический контроллер, представленные в виде микроконтроллера *ATMega 328P* через *USB-UART* преобразователь. Технологический контроллер распределяет полученные данные по приводам посредство шины SPI и ждет выполнения задания. Для этого используется шина синхронизации. После чего «говорит» компьютеру, что готов принять новую порцию данных.

Микроконтроллеры приводов получив задание, сравнивают с текущим положением привода, которое получают, всегда считая импульсы энкодера, определяют необходимое направление вращения и подают управляющий сигнал на драйвер двигателя. Продолжая при этом получать обратную информацию в виде показаний энкодера. При достижении необходимого положения микроконтроллеры останавливают движение приводов и сообщают по шине синхронизации технологическому контроллеру о завершении выполнения задания. При срабатывании датчика конечного положения в любой момент времени всё движение двигателей немедленно останавливается и микроконтроллер уходит в аварийный режим, при котором дальнейшее движение привода невозможно. Исключением является этап калибровки привода, когда микроконтроллер запускает двигатель до тех пор, пока не сработает датчик конечного положения и в этот момент обнуляет счетчик импульсов энкодера.

Таким образом обеспечивается обратная связь каждого привода и повышается точность, безопасность и авария устойчивость системы.

На рисунке3.1.1 приведена функциональная схема системы управления манипулятором с учетом всех вышеперечисленных данных.

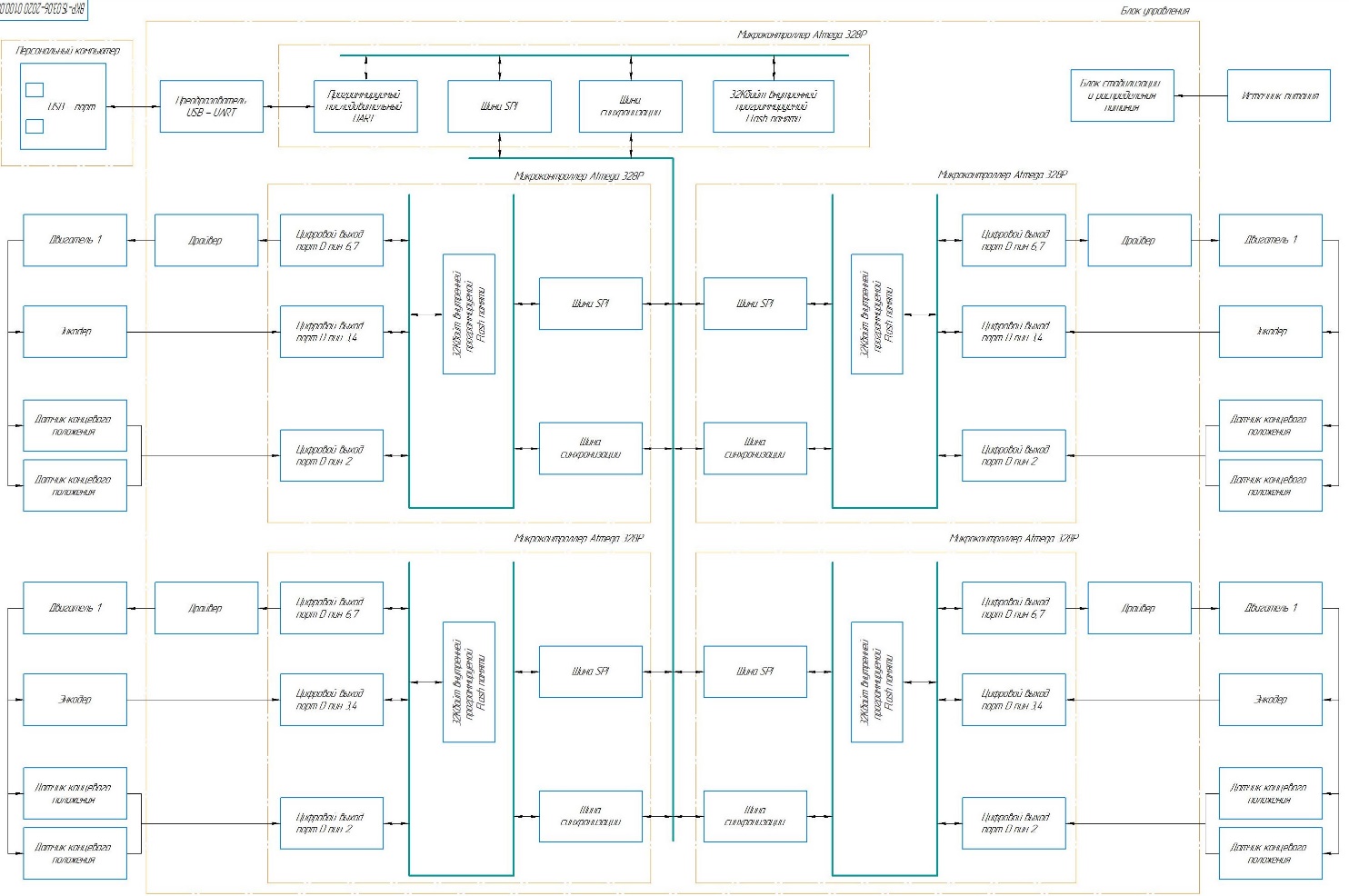


Рисунок 3.1.1 – Функциональная схема системы управления манипулятором

## 3.2 Разработка и выбор аппаратных средств сбора данных

В качестве датчиков обратной связи будут использоваться четыре датчика угла поворота и 8 концевых датчиков.

При выборе датчика угла поворота нужно ориентироваться на число шагов за 1 оборот двигателя. У двигателей *FL57STH56-1006MB*, которые используются на 3-х приводах это 0.9º.

Количество шагов за один оборот:

|  |  |
| --- | --- |
|  | (3.2.1) |

где – величина одного шага двигателя.

Для данных 3-х двигателей используем инкрементальный энкодер как датчик угла поворота, выполненный с полым сквозным валом. Они будут установлен на каждый второй вал шагового двигателя.

Был выбран инкрементальный энкодер *Autonics E40H6-3000-3-V-5*, габаритные размеры которого представлены на рисунке 3.2.1.

Таблица 12 – Техническая характеристика энкодера *E40H6-3000-3-V-5*

|  |  |
| --- | --- |
| Параметры | Значения |
| Количества импульсов | 3000 |
| Максимальное число оборотов, об/мин | 5000 |
| Нагрузка на подшипник, Н | 2кгс - радиальная, 1кгс - максимальная |
| Выходные фазы | A, B, Z |
| Предельная частота, кГц | 200 |
| Рабочее напряжение, В | 5 |
| Рабочая температура, °C | От -10 до +70 |
| Диаметр под вал, мм | 6 |

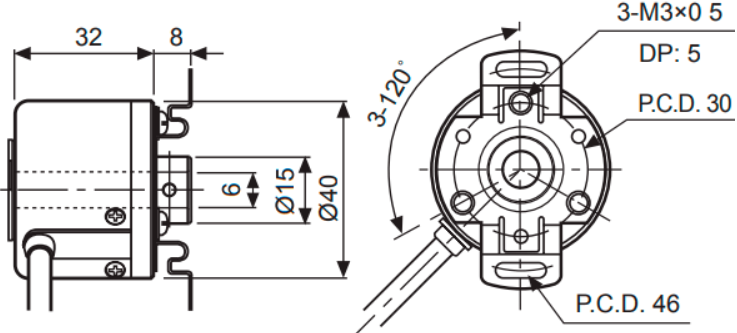


Рисунок 3.2.1 – Габаритные размеры энкодера

У двигателя *FL20STH42-0804A*, который используется в приводе поворота схвата угловой шаг равен 1.8:

Количество шагов за один оборот:

Для данного двигателя также используется инкрементальный энкодер как датчик угла поворота, выполненный с полым сквозным валом. Он будут установлен на второй вал шагового двигателя.

Был выбран инкрементальный энкодер *Autonics E20HB 3-360-3-N-5-R*, габаритные размеры которого представлены на рисунке 3.2.2.

Таблица 13 – Техническая характеристика энкодера E20HB 3-360-3-N-5-R

|  |  |
| --- | --- |
| Параметры | Значения |
| Количества импульсов | 360 |
| Максимальное число оборотов, об/мин | 6000 |
| Нагрузка на подшипник, Н | 2кгс - радиальная, 2кгс - максимальная |
| Выходные фазы | A, B, Z |
| Предельная частота, кГц | 100 |
| Рабочее напряжение, В | 5 |
| Рабочая температура, °C | От -10 до +70 |
| Диаметр под вал, мм | 3 |

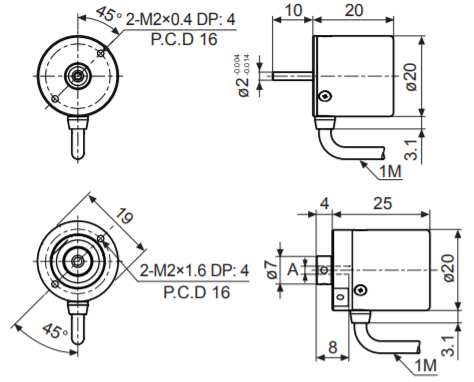


Рисунок 3.2.2 – Габаритные размеры энкодера

В качестве концевых датчиков на 3х приводах выбраны *SHL-W55*. На рисунке 3.2.3 представлены габаритные размеры датчика и его техническая характеристика

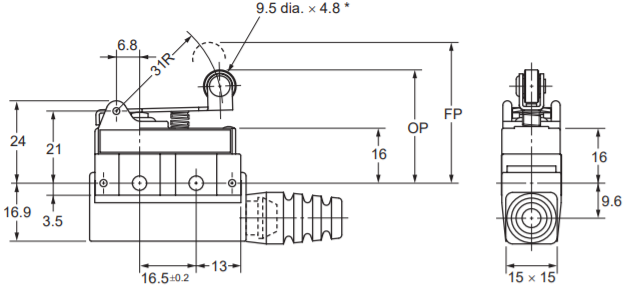


Рисунок 3.2.3 – Габаритные размеры концевого датчика *SHL-W55*

Таблица 14 – Техническая характеристика датчика *LJ12A3-4-Z/BX*

|  |  |
| --- | --- |
| Параметры | Значения |
| Напряжение питания, B | 5-30 |
| Ток, мА | 1500 |
| Полярность выхода | NPN |
| Частота срабатывания, 1/мин | 30 |
| Вес, г | 62 |

В качестве концевого датчика для привода поворота схвата принимаем индуктивный датчик конечного положения *IQ05-0B8NS-ZU1*. На рисунке 3.2.4 представлены габаритные размеры датчика.

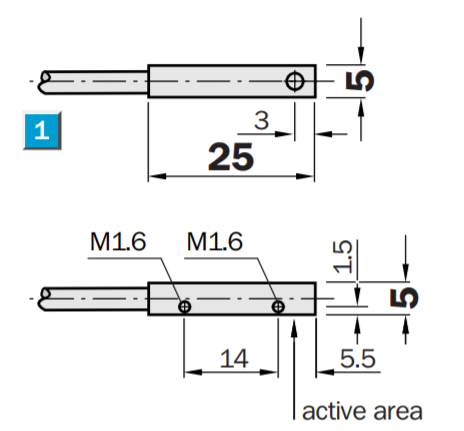


Рисунок 3.2.4 – Габаритные размеры индуктивного концевого датчика *IQ05-0B8NS-ZU1*

Таблица 15 – Техническая характеристика датчика *LJ12A3-4-Z/BX*

|  |  |
| --- | --- |
| Параметры | Значения |
| Напряжение питания, B | 5-30 |
| Диапазон срабатывания, мм | 0.8 |
| Ток, мА | 200 |
| Полярность выхода | NPN |
| Максимальная частота срабатывания, 1/сек | 5000 |
| Рабочая температура, ºС | От -25 до +70 |
| Вес, г | 26 |

## 3.3 Разработка драйвера шагового двигателя

В данном разделе будут разработан драйвер шагового двигателя, выбраны все необходимые компоненты, разработана схема электрическая принципиальная.

3.3.1 Выбор основных компонентов

В качестве драйвера для шагового двигателя было решено выбрать связку из микросхем *L297* и *L298*. Микросхема *L297* содержит логику формирования временных последовательностей, а *L298* представляет собой мощный сдвоенный *H*-мост.

На рисунке 3.3.1.1 показано назначение выходов микросхемы *L297*.

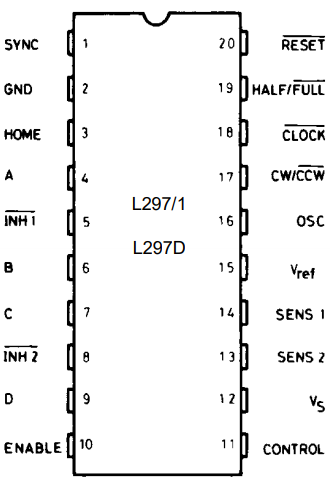


Рисунок 3.3.1.1 – Распиновка микросхемы *L297*

На рисунке 3.3.1.2 показана логика формирования импульсов в полушаговом режиме в зависимости от входных сигналов.

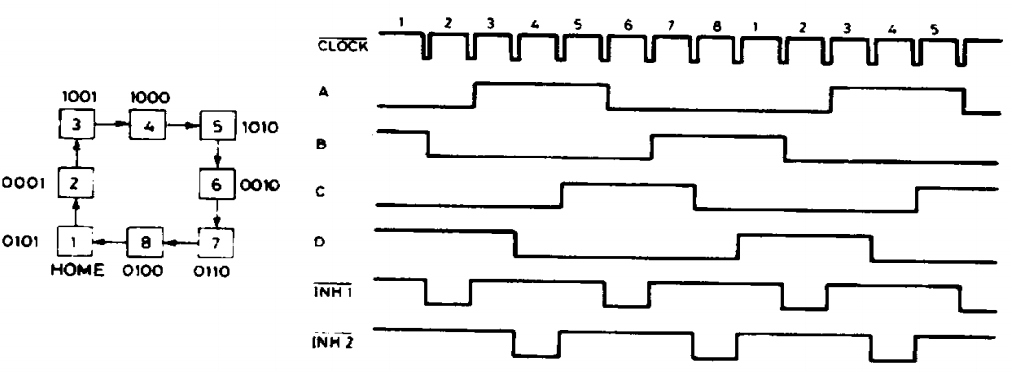


Рисунок 3.3.1.2 – Диаграмма формирование управляющих импульсов на микросхеме *L297* в режиме деления шага

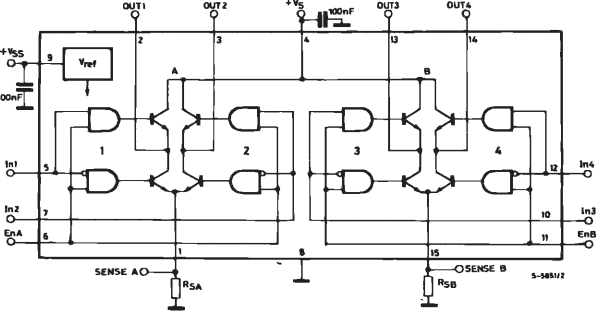


Рисунок 3.3.1.3 – Блок-диаграмма устройства микросхемы *L298*

На рисунке 3.3.1.3 показано внутренне устройство микросхемы *L298,* а на рисунке 3.3.1.4 показано рекомендуемая схема соединения микросхемы *L297* с *L298*.

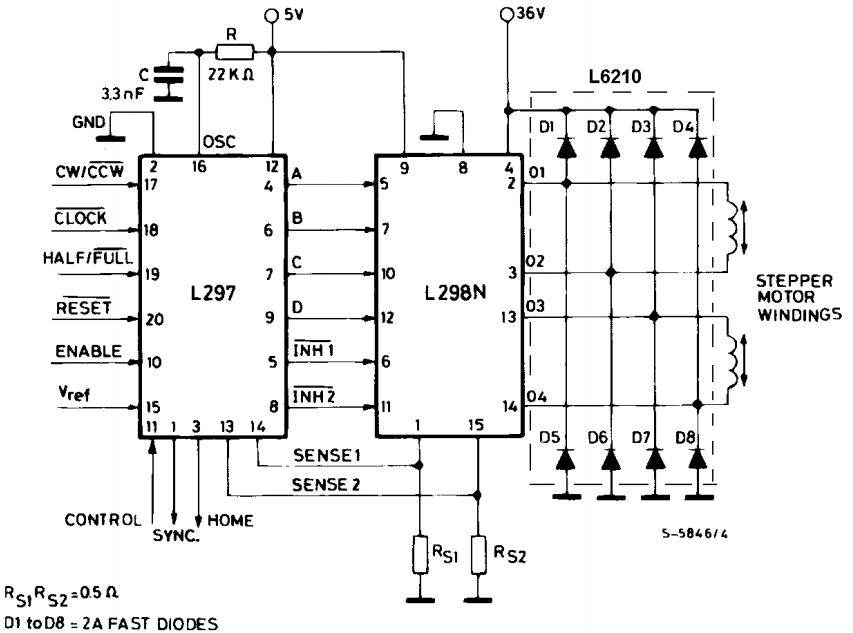


Рисунок 3.3.1.4 – Рекомендуемая схема подключения микросхем *L297* и *L298*

3.3.2 Разработка схемы электрической принципиальной

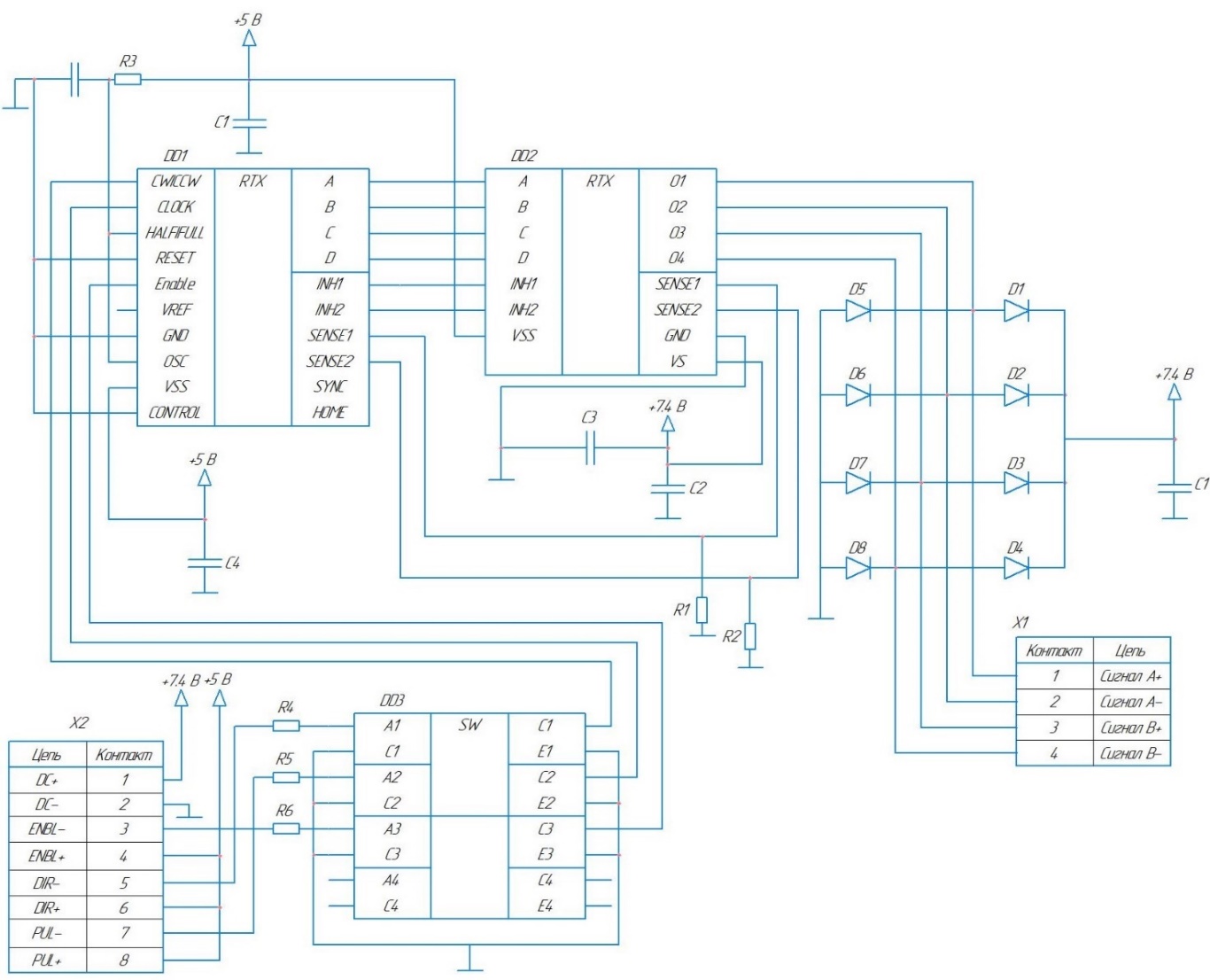


Рисунок 3.3.2.1 – Драйвер шагового двигателя. Схема электрическая принципиальная



Рисунок 3.3.2.2 – Драйвер шагового двигателя. Таблица компонентов

## 3.4 Разработка микропроцессорного блока

В данной главе представлен процесс разработки микропроцессорного блока, выбор основных компонентов и разработка схемы электрической принципиальной.

3.4.1 Выбор основных компонентов

Вся логика управления драйверами шаговых двигателей и общения с компьютером реализована с помощью пяти микроконтроллеров *ATMega 328P*, блок-диаграмма которого представлена на рисунке 3.4.1.1, объединенных шиной SPI и шиной синхронизации. Рассмотрим характеристика данного микроконтроллера.

– высокопроизводительные, мало потребляющие *AVR* 8-битные микроконтроллеры;

– развитая RISC архитектура;

* 1. 131 исполняемых команд, большинство за один машинный такт;
  2. 32 рабочих регистра общего назначения;
  3. полностью статический режим работы;
  4. производительность до 16 *MIPS* при 16 МГц;
  5. встроенный 2-х тактовый умножитель.

– энергонезависимая память программ и данных;

* 1. 32 Кбайт внутрисистемной самопрограммируемой *FLASH* памяти с количеством циклов перепрограммирования до 10 000;
  2. Опционное загрузочная область памяти с независимыми ключевыми битами, внутрисистемное программирование встроенной загрузочной программой, правильное чтение в процессе записи;
  3. 1 Кбайт *EEPROM* с допустимым количеством циклов стирания записи до 100 000;
  4. 2Кбайт внутренней *SRAM*;
  5. программируемый ключ защиты программ.

– периферийные функции;

* 1. два 8-битных таймера/счётчика с программируемым предделителем и режимом сравнения;
  2. один 16-битный таймер/счётчик с программируемым предделителем, режимом сравнения и захвата;
  3. счётчик реального времени с программируемым генератором;
  4. шесть ШИМ канала;
  5. 8-и канальный, 10-и битный АЦП;
  6. программируемый *USART*;
  7. *Master/Slave SPI* последовательный интерфейс;
  8. Программируемый *Watchdog* таймер с программируемым генератором;
  9. Встроенный аналоговый компаратор.

– специальные функции;

* 1. *Reset* по включению питания и выключение при снижении напряжения питания;
  2. Внутренний калиброванный RC генератор;
  3. Внешние и внутренние источники прерывания;
  4. Шесть экономичных режимов: *Idle*, подавления шумов АЦП, экономичный, режим ожидания и режим расширенного ожидания.

– 23 программируемых линии ввода-вывода;

– 32 выводной корпус *TQFP* или 32-контактный *MLF*;

– напряжение питания: 2.7 В до 5.5 В;

– тактовая частота: 0-16 МГц.

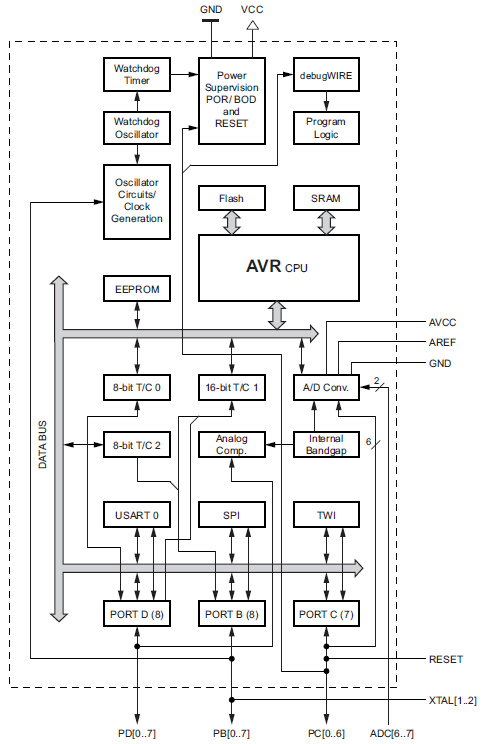


Рисунок 3.4.1.1 – Блок-диаграмма внутренней логики микроконтроллера *ATMega328P*

3.4.2 Разработка схемы электрической принципиальной

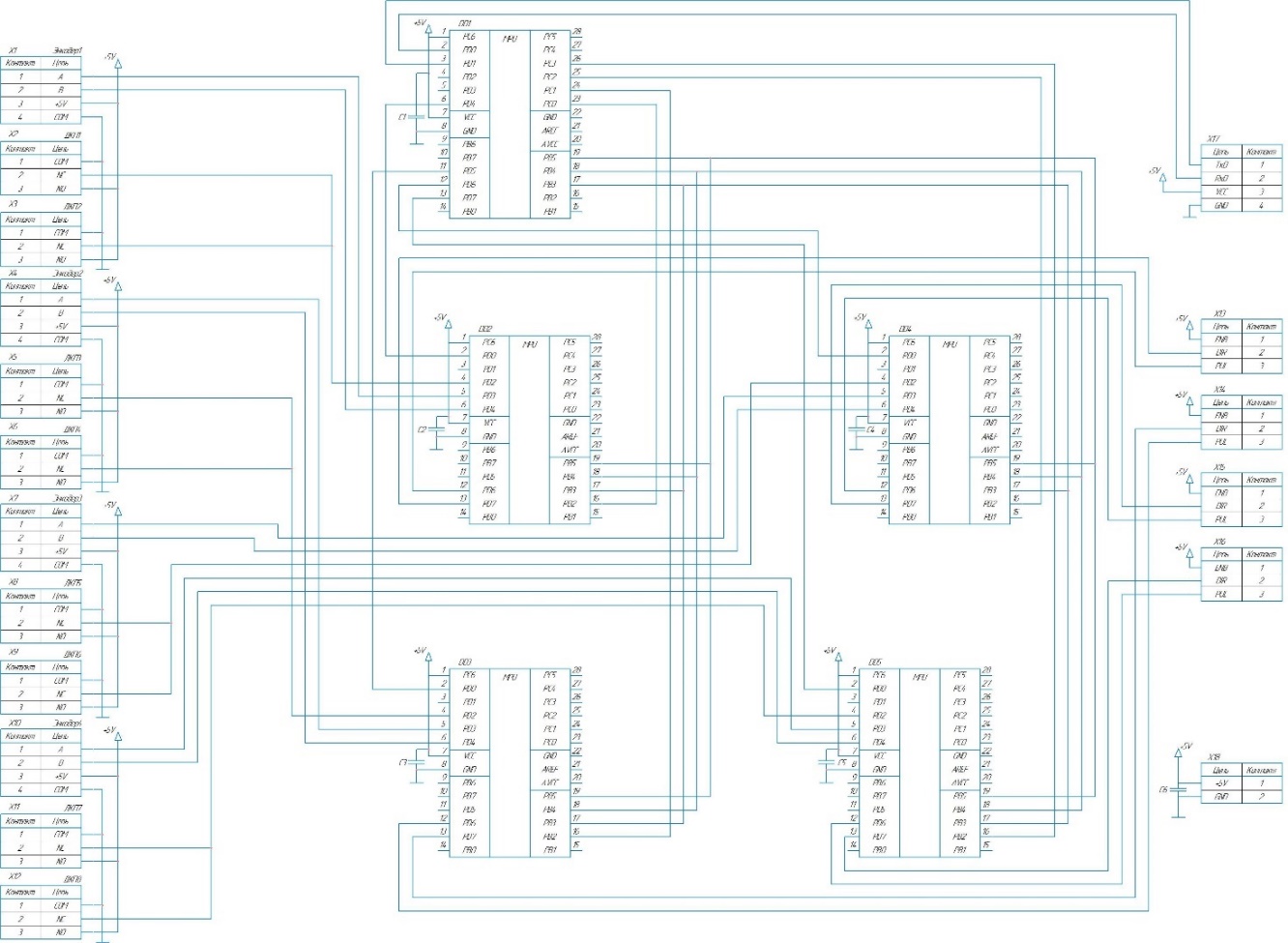


Рисунок 3.4.1.1 – Микропроцессорный блок управления. Схема электрическая принципиальная

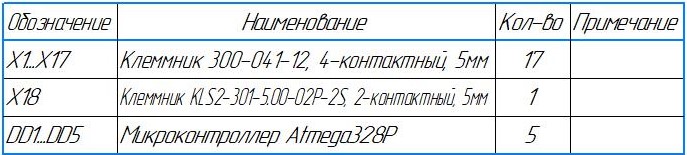


Рисунок 3.4.1.2 – Микропроцессорный блок управления. Таблица компонентов

## 3.5 Разработка интерфейсных преобразователей

В качестве интерфейсного преобразователя для общения персонального компьютера с микроконтроллером была выбрана микросхема *CH341*.

*CH341* – мультифункциональный конвертер из *USB 2.0* в *UART, EPP, I2C* и *SPI*. В режиме параллельного порта реализован интерфейс *EPP* и эмуляция принтера (позволяющая подключать принтеры с интерфейсом *LPT* напрямую к *USB* без написания дополнительных драйверов). В последовательном режиме поддерживаются интерфейсы *I2C* и *SPI*. Все поддерживаемы интерфейсы представлены на рисунке 3.5.1.

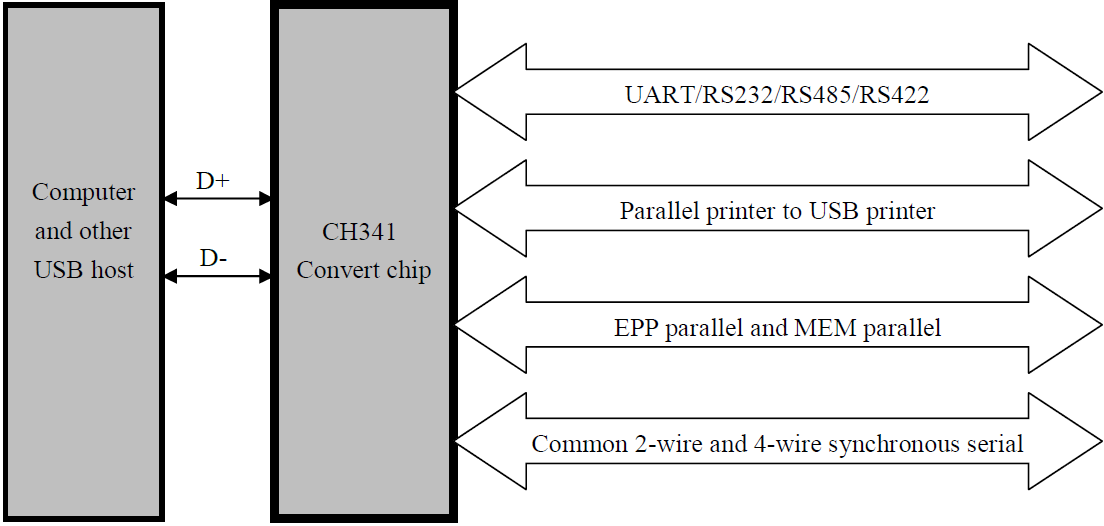


Рисунок 3.5.1 – *CH341*, поддерживаемы интерфейсы

При работе с *UART* интерфейсом поддерживаются следующие функции:

* имитация стандартного последовательного порта, используемого в периферийном оборудовании или для возможности расширения дополнительного последовательного порта через *USB*;
* полностью совместим с программами, использующими последовательный интерфейс на компьютере с операционной системой *Windows*;
* аппаратный полнодуплексный последовательный интерфейс, встроенный преобразователь и приемный буфер, поддержка 50 бит/с ~ 2 Мбит/с скорости передачи данных;
* поддержка пяти, шести, семи или восьми битов данных, поддержка одного, двух стоп битов или без них;
* поддержка интерфейса *RS232*, *RS485* и *RS422* путем добавления внешнего оборудования для изменения логического уровня;
* поддержка косвенного доступа к внешней последовательной памяти *EEPROM* через стандартный последовательный интерфейс.

На рисунках 3.5.2, 3.5.3, 3.5.4 показаны типовые схемы подключения *CH341* в режиме совмести с *RS232*, *RS485* и *UART* соответственно.

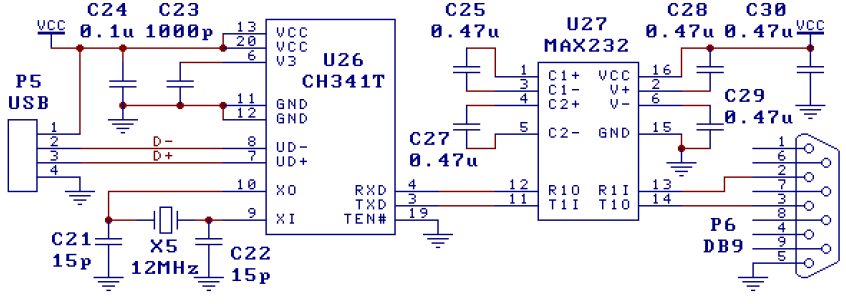


Рисунок 3.5.2 – Схема подключение *CH341* в режиме совместимости с *RS232*

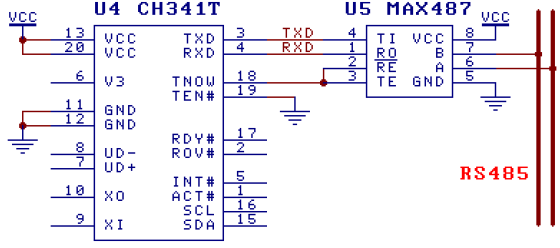


Рисунок 3.5.3 – Схема подключения *CH341* в режиме совместимости с *RS485*

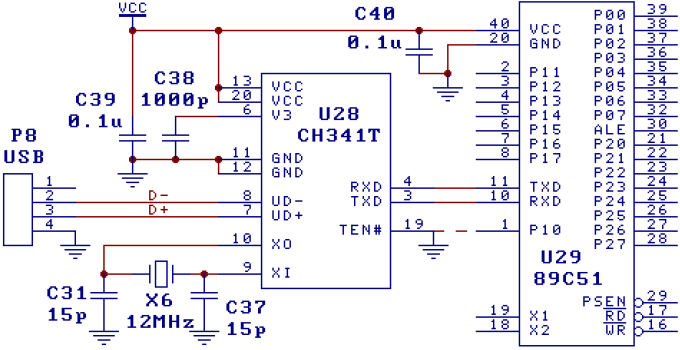


Рисунок 3.5.4 – Пример подключения *CH341* в режиме работы с *UART*

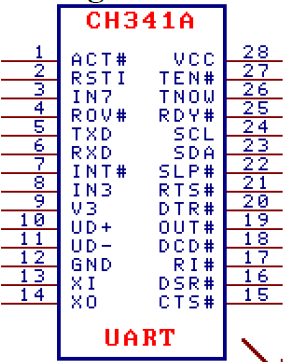


Рисунок 3.5.5 – Распиновка чипа *CH341A*

## 3.6 Разработка схемы электрической соединений

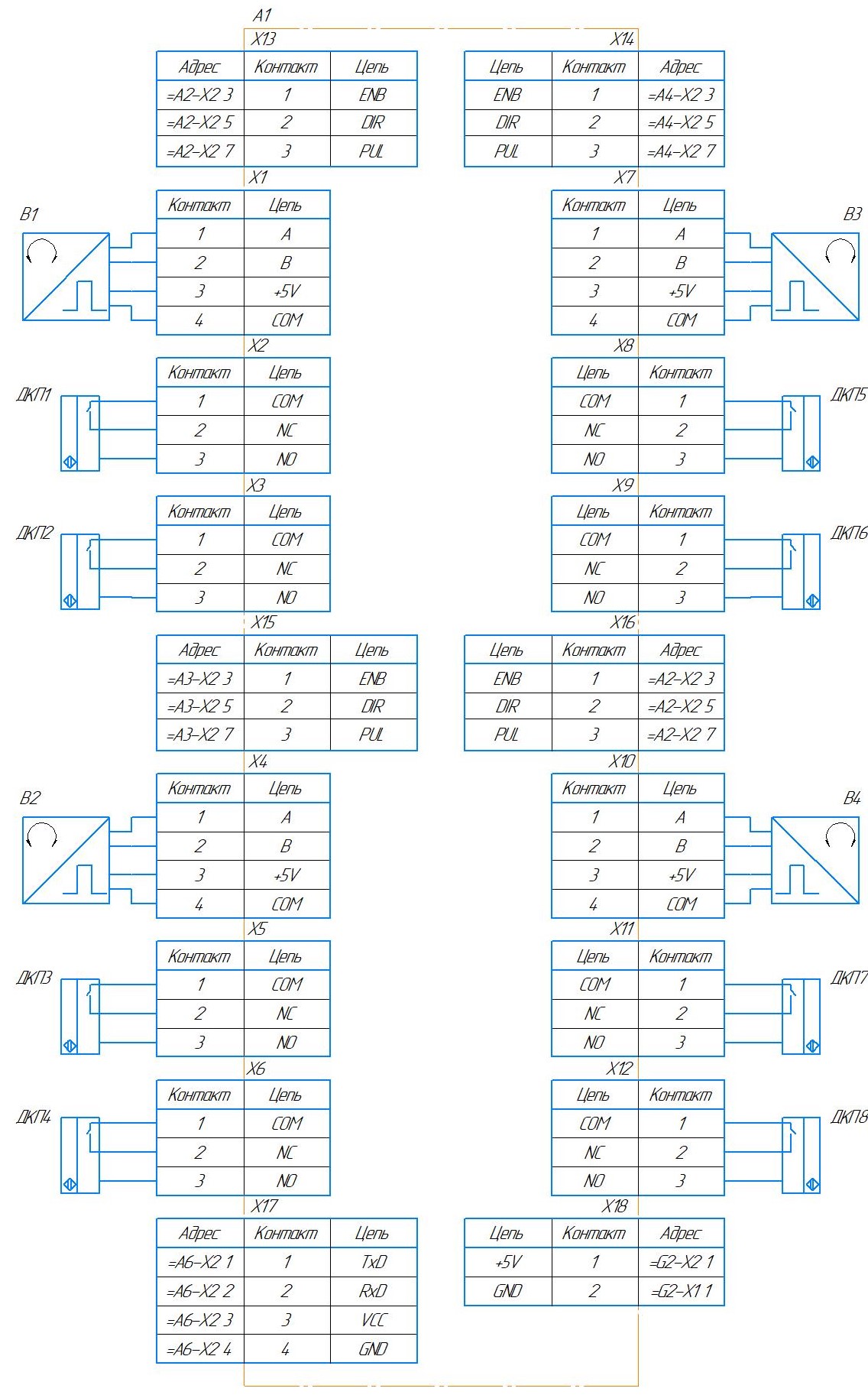


Рисунок 3.6.1 – Схема электрическая соединений. Блок управления

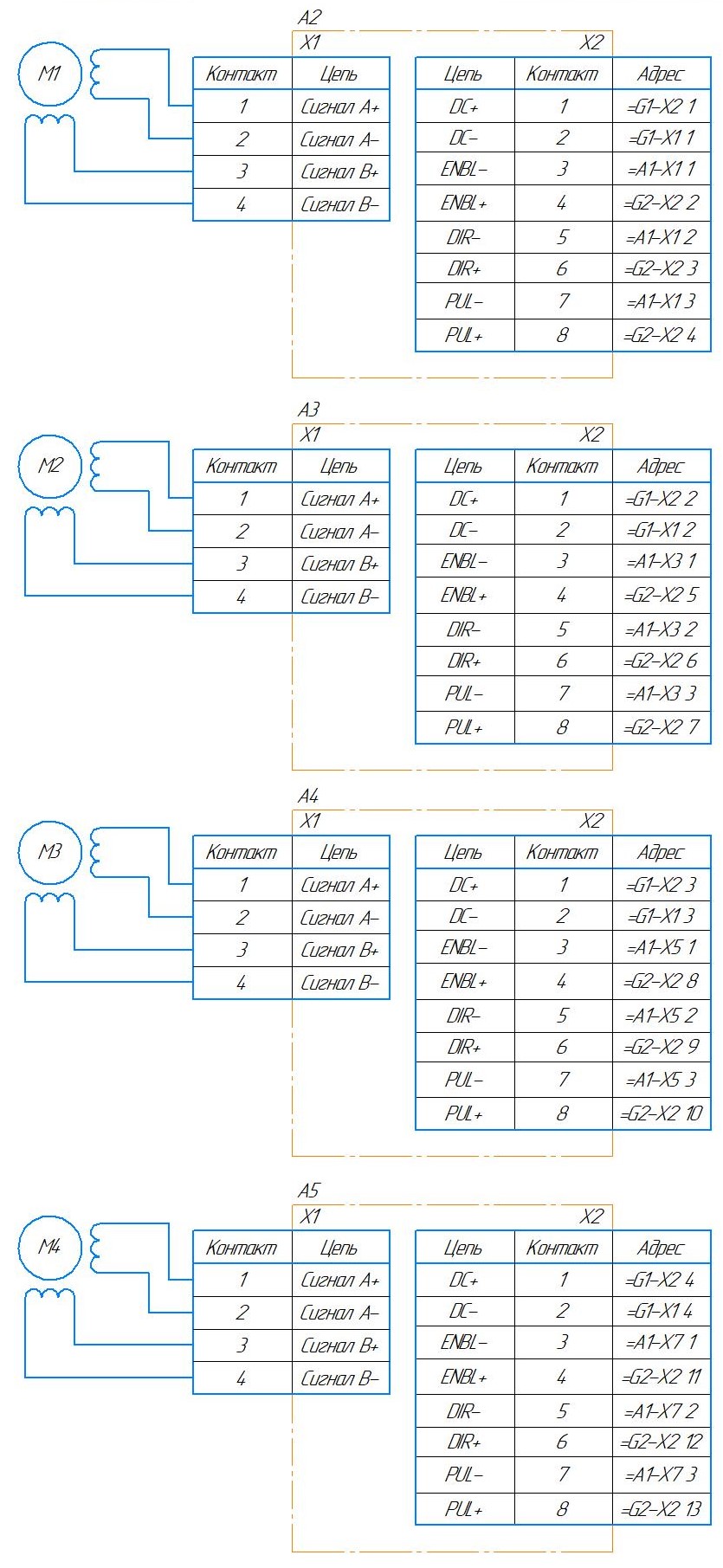


Рисунок 3.6.2 – Схема электрическая соединений. Подключение драйверов шагового двигателя

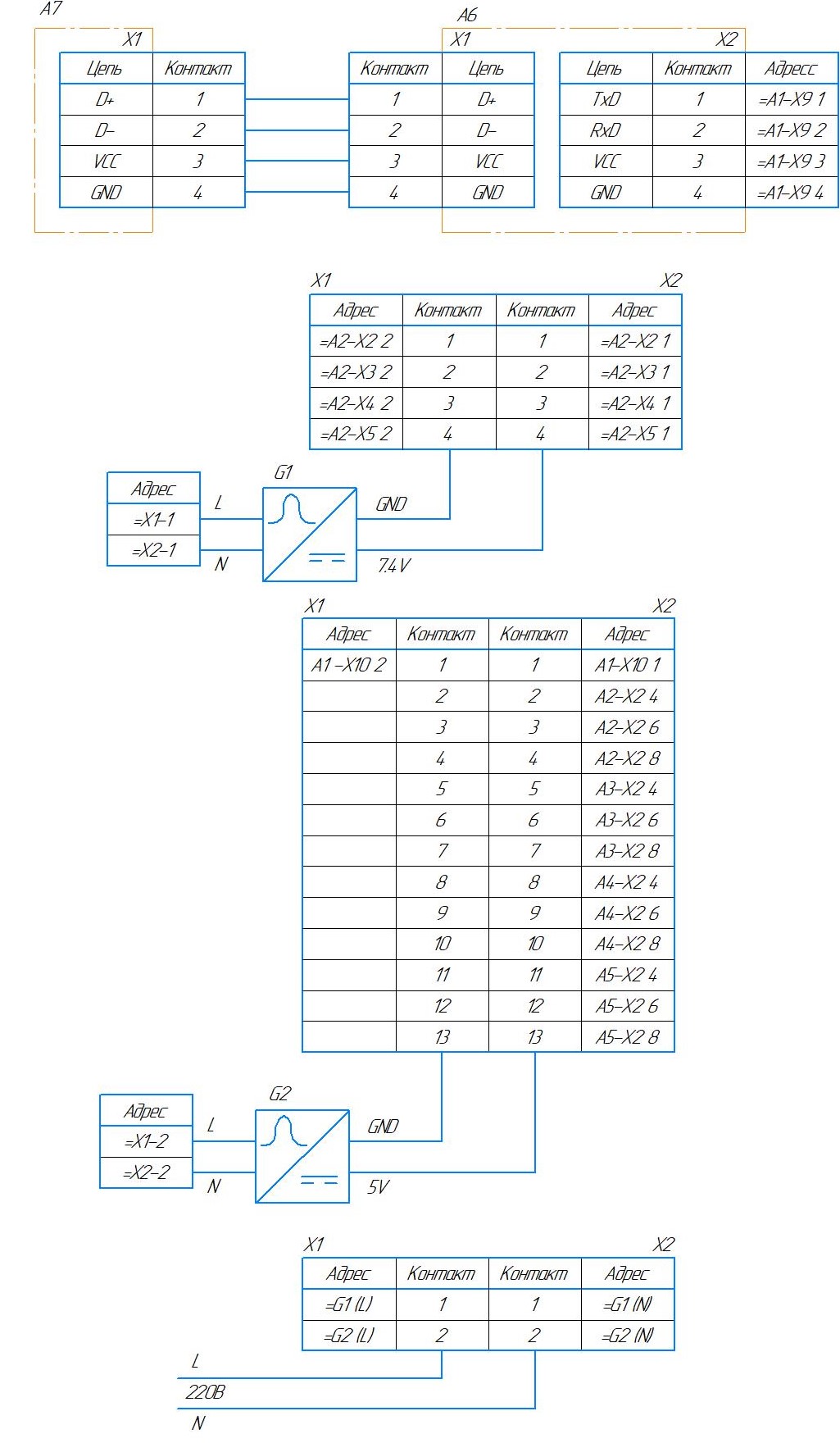


Рисунок 3.6.3 – Схема электрическая соединений. Подключение персонального компьютера, преобразователя USB-USART, блоков распределения питания и источника питания



Рисунок 3.6.4 – Схема электрическая соединений. Таблица компонентов

Выводы по системе управления

Проработана и описана архитектура системы управления, разработана ее функциональная схема, выбраны аппаратные средства сбора данных. Был разработан драйвер шагового двигателя, были выбраны основные компоненты, разработана схема электрическая принципиальная. Также разработан микропроцессорный блок, выбраны основные компоненты для него, разработана схема электрическая принципиальная. Были выбраны и разработаны интерфейсные преобразователи, разработана схема электрическая соединений мехатронного модуля.

# 4 ПРОГРАММНАЯ ЧАСТЬ

В данном разделе представлен процесс разработки программного обеспечения всех составных частей манипулятора. Представлены алгоритмы работы управляющей программы, программы технологического контроллера, программы микроконтроллера привода.

## 4.1 Составление алгоритмов работы

В данном разделе будут составлены алгоритмы работы программы для всех составных частей системы управления, а именно для управляющего манипулятором персонального компьютера, для технологического контроллера, который обеспечивает общение компьютера и микроконтроллеров приводов и для самих МК приводов, которые управляют движением своего привода в соответствии с датчиками обратной связи.

4.1.1 Разработка алгоритма работы манипулятора

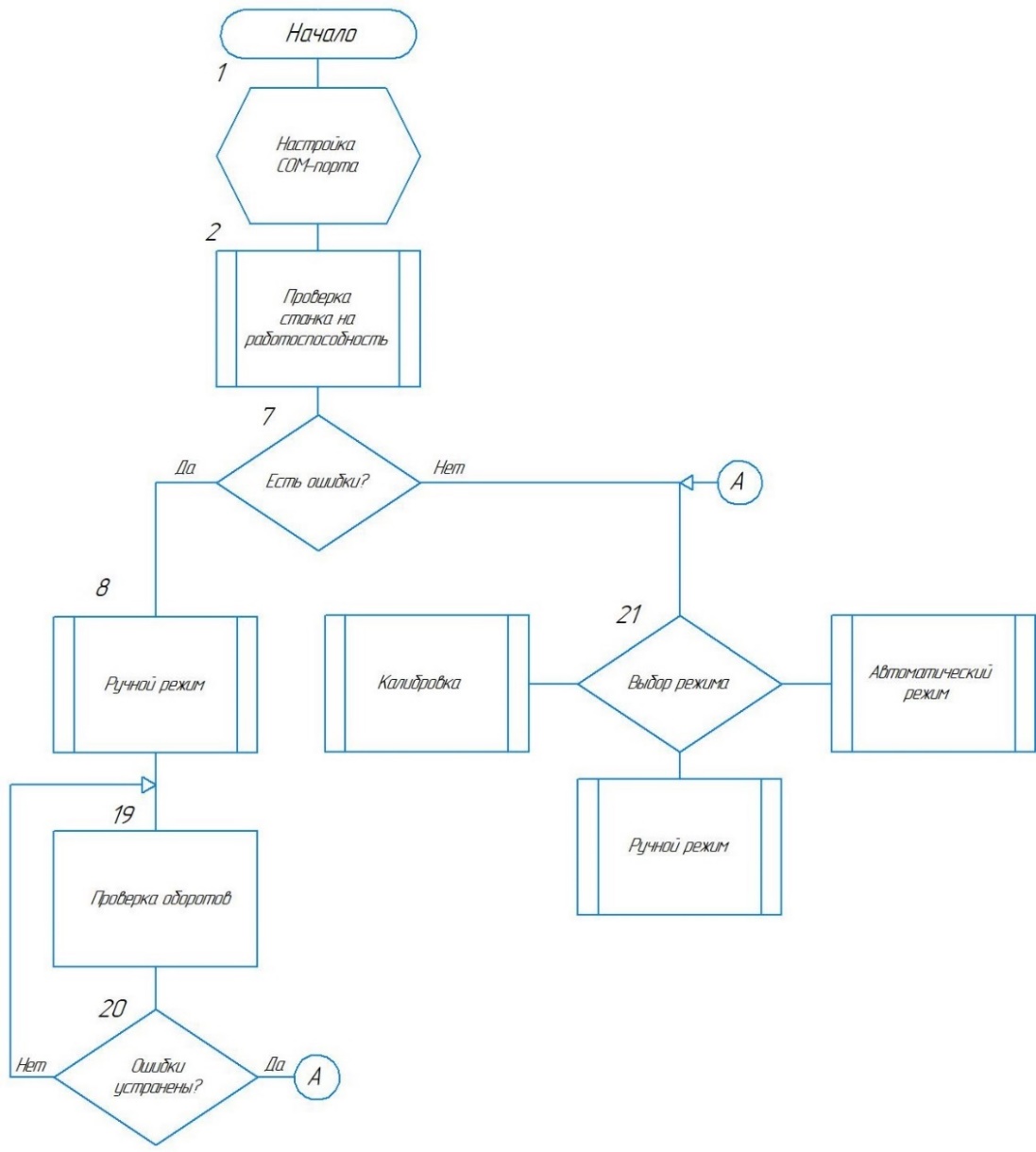


Рисунок 4.1.1.1 – Алгоритм работы манипулятора. Начало работы программы управления

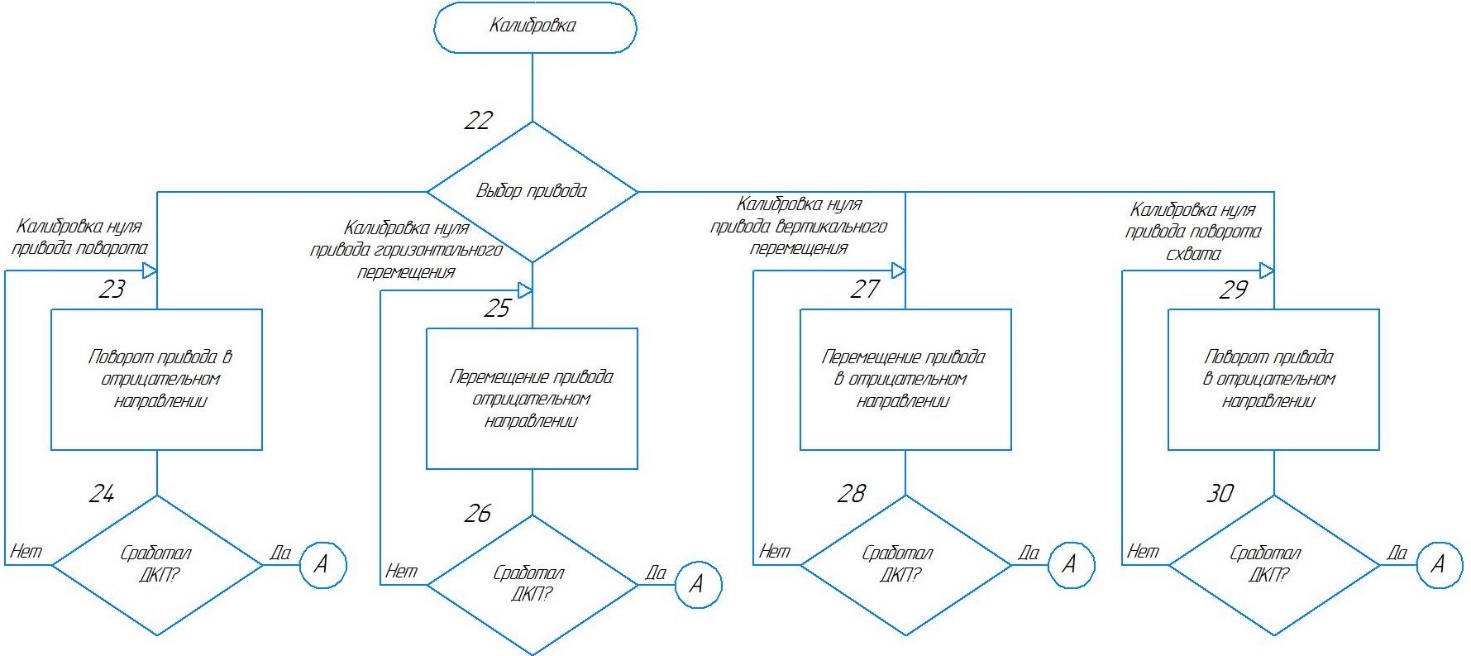


Рисунок 4.1.1.2 – Алгоритм работы манипулятора. Калибровка

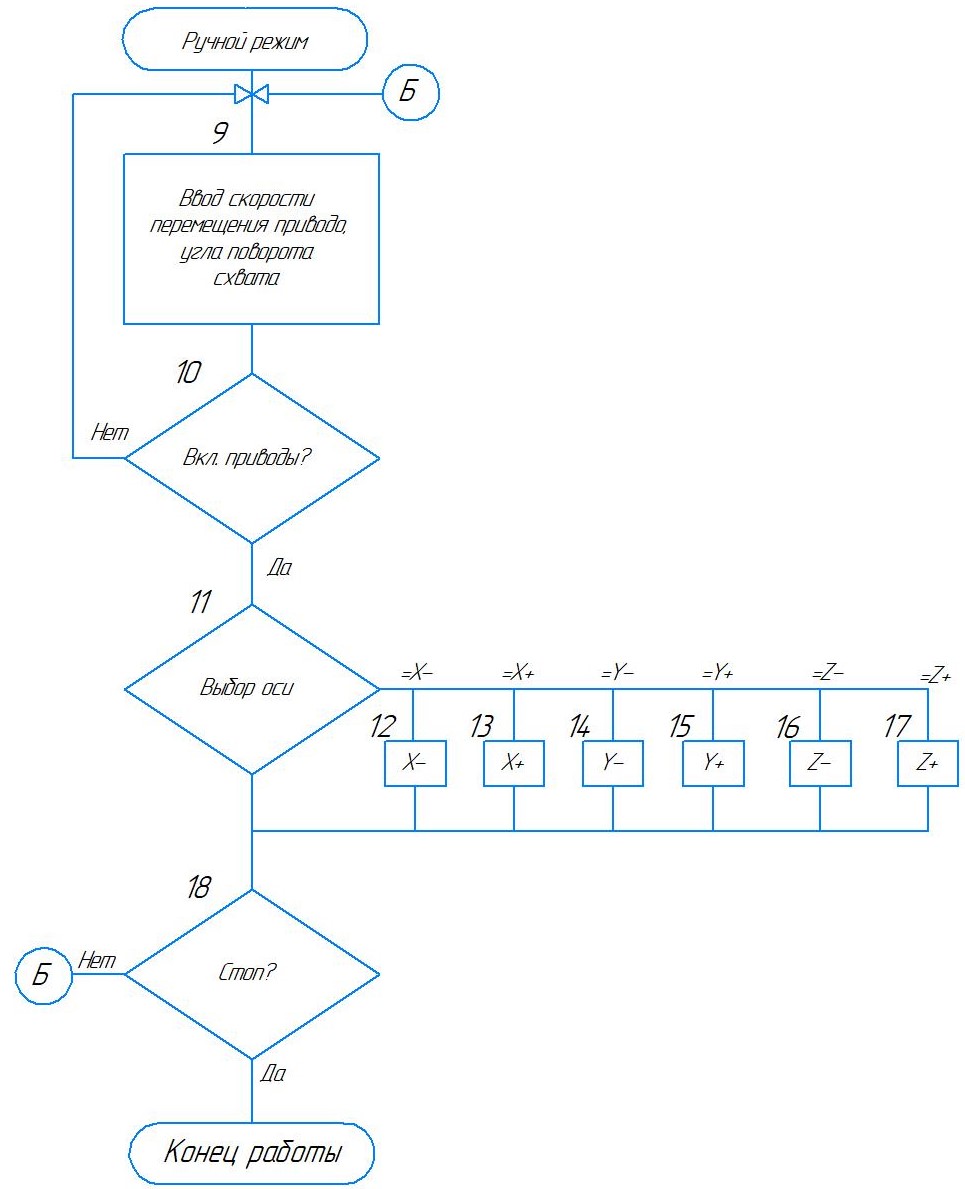


Рисунок 4.1.1.3 – Алгоритм работы манипулятора. Ручной режим

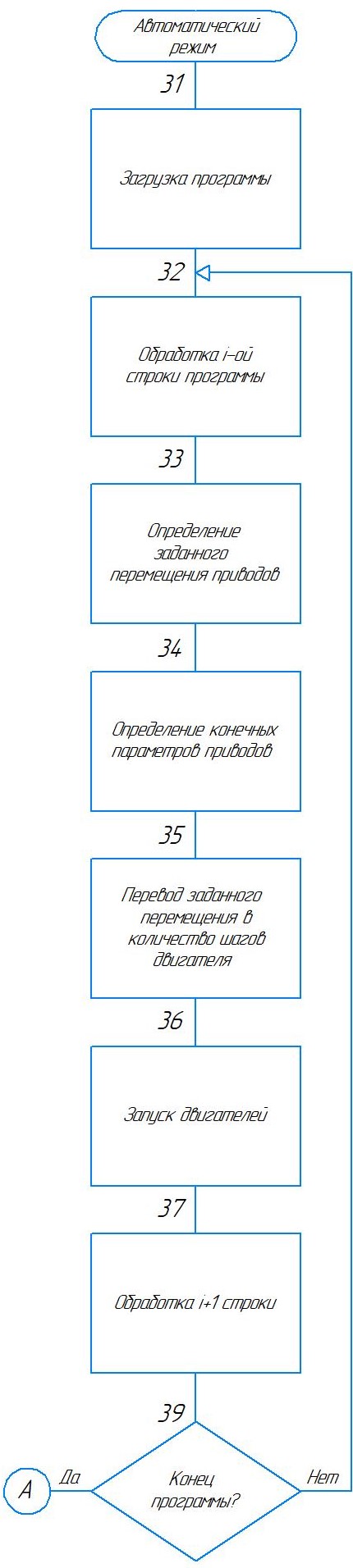


Рисунок 4.1.1.4 – Алгоритм работы манипулятора. Автоматический режим

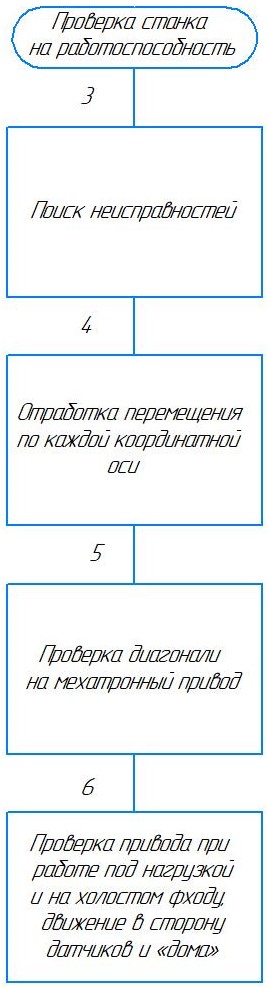


Рисунок 4.1.1.5 – Алгоритм работы манипулятора. Проверка станка на работоспособность

4.1.2 Разработка алгоритма программы микроконтроллера приводов

В начале работы программы происходит инициализации необходимой периферии, настройки портов ввода-вывода, калибровка привода. После чего начинается бесконечный цикл, в котором отслеживается состояние флага срабатывания ДКП и ожидание задания от технологического контроллера. После получения задания, начинается его выполнение.

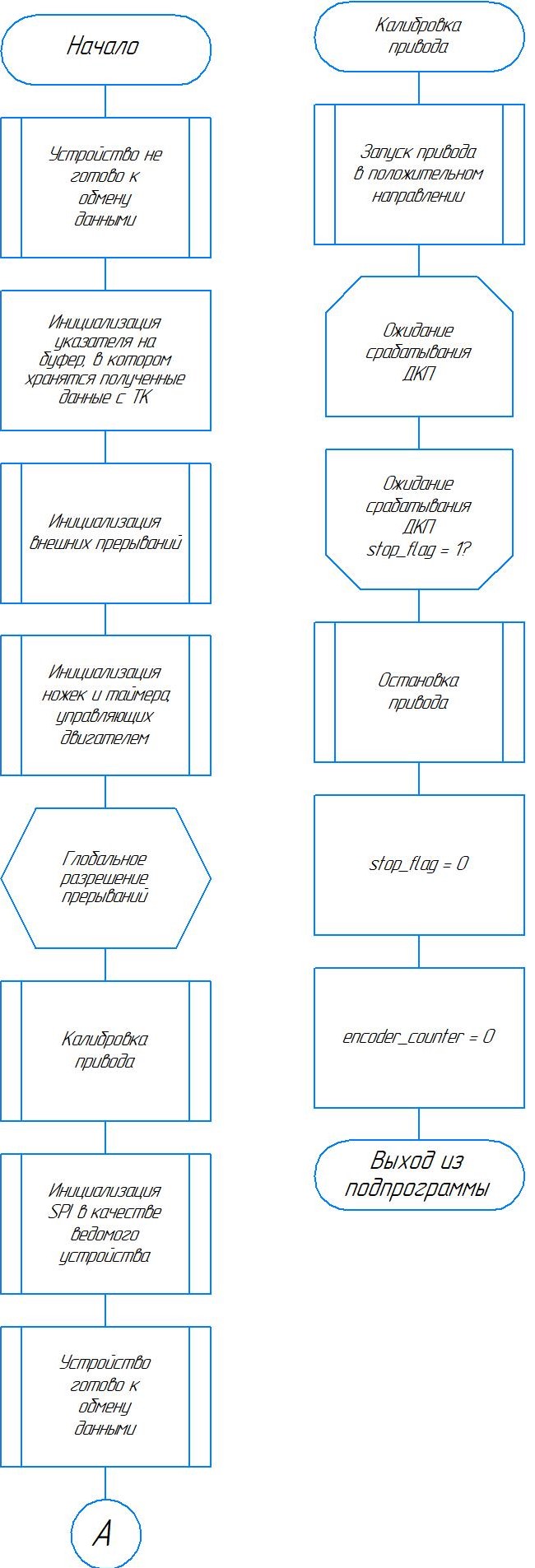


Рисунок 4.1.2.1 – Алгоритм работы программы микроконтроллера привода. Точка входа в программу и функция калибровки привода

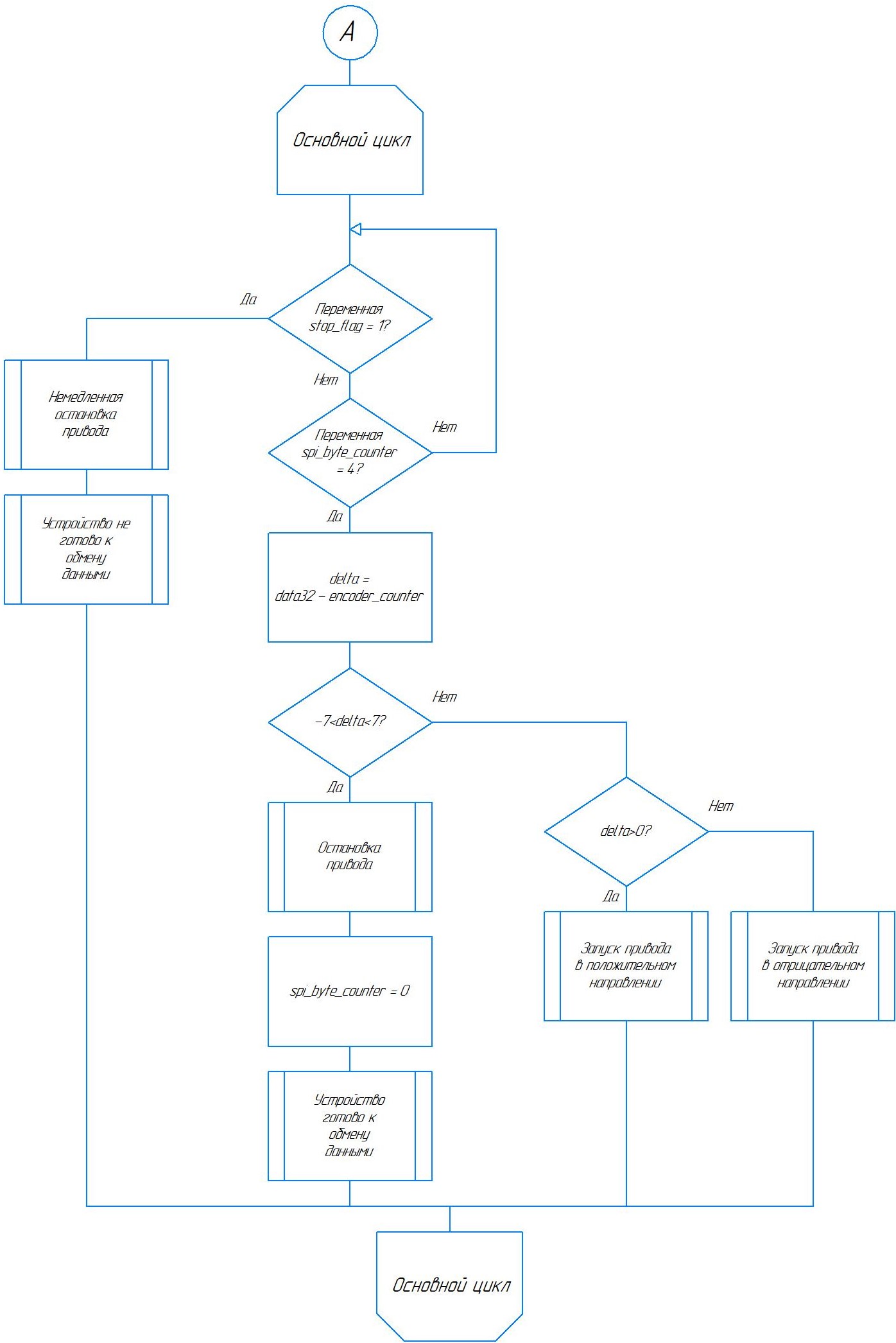


Рисунок 4.1.2.2 – Алгоритм работы программы микроконтроллера привода. Основной цикл

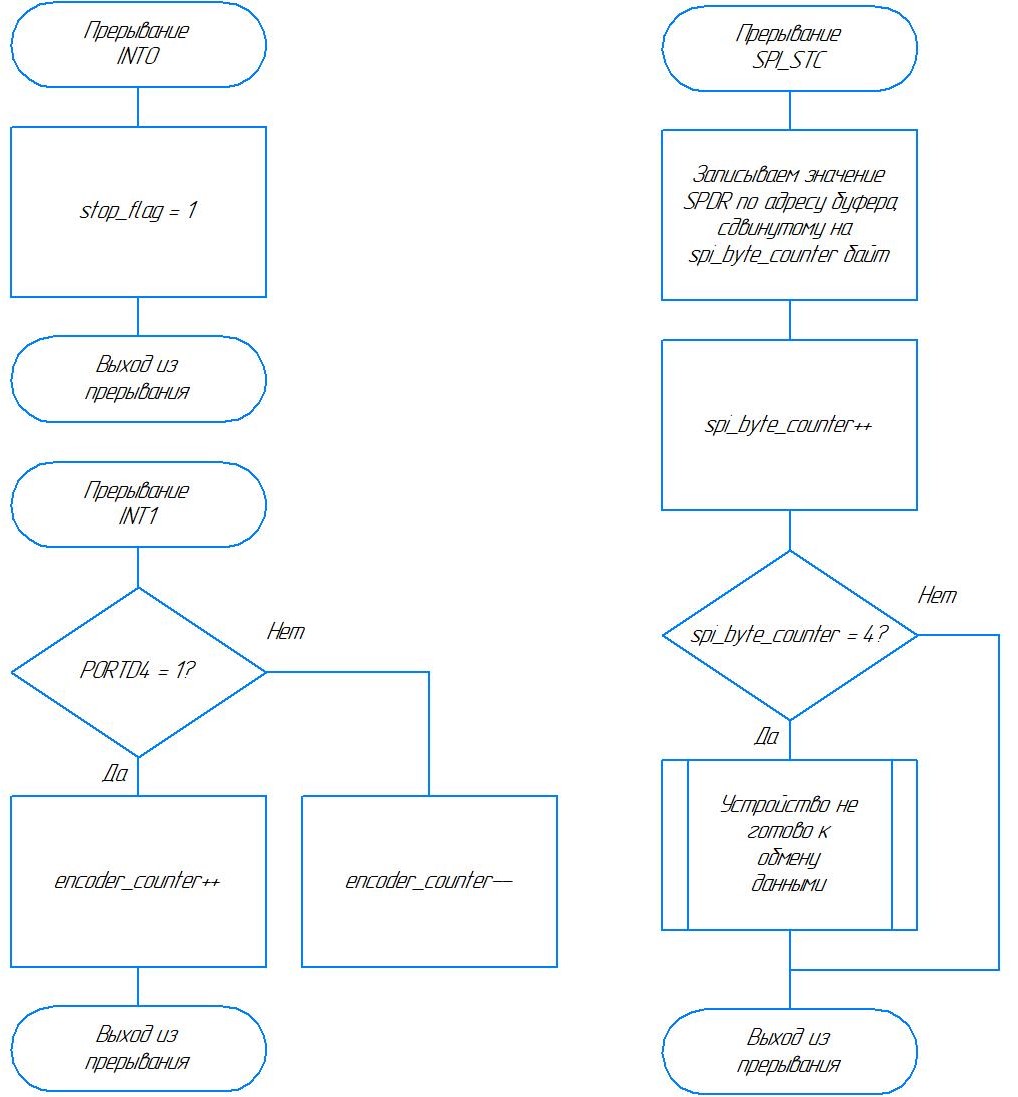


Рисунок 4.1.2.3 – Алгоритм работы программы микроконтроллера привода. Обработчики прерываний

4.1.3 Разработка алгоритма программы технологического контроллера

В начале работы программы инициализируется вся необходимая периферия, настройка портом ввода-вывода, разрешение прерываний. После чего компьютеру посылается число 120, говорящее, что ТК готов к обмену данными. По получению 16-ти байт данных (по 4 на каждый привод) и при завершении работы всех приводов, новые данные отправляются на каждый привод.

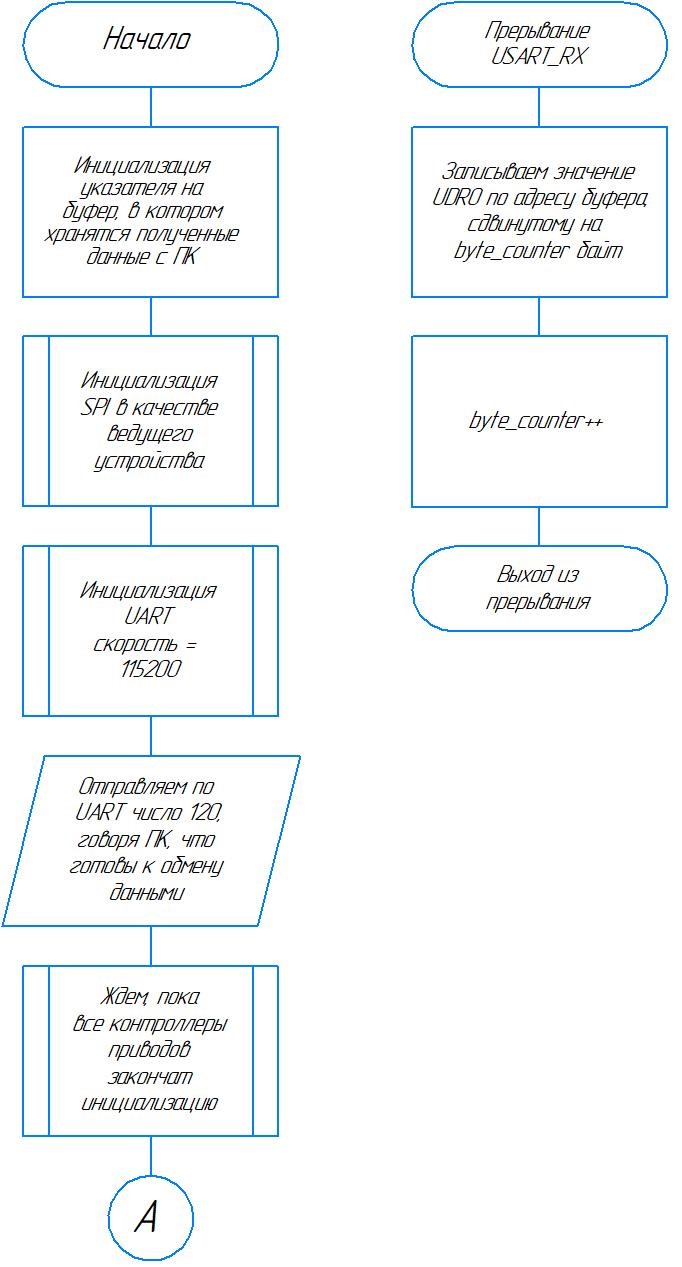


Рисунок 4.1.3.1 – Алгоритм программы технологического контролера. Точка входа в программу и обработчик прерывания по приему данных по *UART*

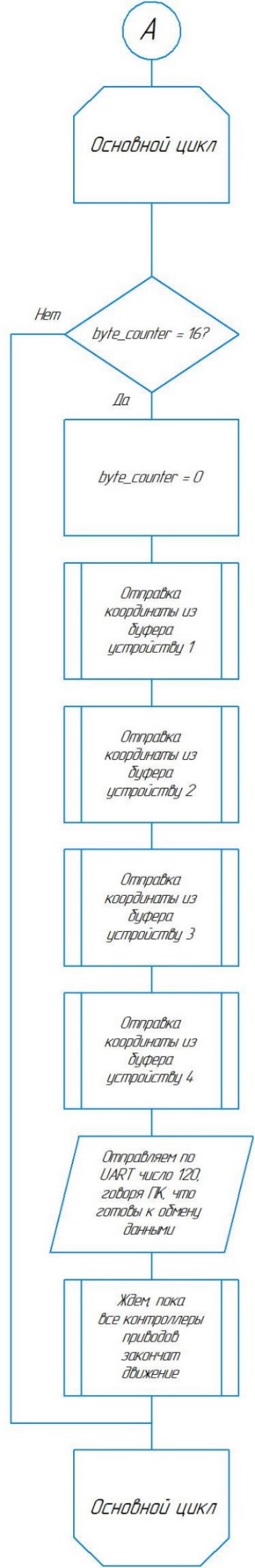


Рисунок 4.1.3.2 – Алгоритм программы технологического контролера. Основной цикл

## 4.2 Разработка программы управления

Разработаны программы для микроконтроллеров приводов и технологического контроллера, представлен исходный код с подробными комментариями и описанием. Также представлена реализация программы управления мехатронным модулем для персонального компьютера с использованием COM-порта.

4.2.1 Разработка программы управления для микроконтроллеров приводов

Программа управления микроконтроллером приводов была написана на языке программирования Си с использованием проприетарной интегрированной среды разработки (IDE) Atmel Studio 7.

Программа представлена несколькими модулями, состоящими из заголовочных файлов (.h) и Си файлов, содержащих исполняемый код (*.с*). Рассмотрим главный файл прошивки *“micro\_0.c”.*

В самом начале производится включение в модуль заголовочного файла *“main.h”,* в котором производятся подключение уже всех остальных модулей и определение макросов.

*#include "Libraries/main.h"*

Далее определяется дефайновая подстановка, содержащия 8-битовое значение, которым задается частота тактирования ножки, подключенной к драйверу шагового двигателя.

*#define MOTOR\_SPEED 255*

Объявляются все переменные, видимые глобально в этом модуле.

*uint8\_t SPI\_byte\_counter = 0*; // Переменная для счета принятых байтов по SPI

*uint8\_t* *stop\_flag = 0*; // Переменная, для хранения состояния ДКП

*uint8\_t* *motor\_dir*; // Флаг, для определения направления вращения двигателя

*uint32\_t* *data32*; // Переменная для полученных данных с технологического контроллера

*uint32\_t* *encoder\_counter = 0*; // Заводим 32-байтовую переменную под счетчик для энкодера.

*int32\_t* *delta = 0*; // Переменная для подсчета ошибки

*uint8\_t \* p\_data32*; // Указатель на первый байт этих данных

Т.к. функции, использованные в модуле, объявлены после точки входа, нам необходимо объявить их прототипы.

*void ext\_INTERRUPT\_init(void);* // Прототип функции, для возможности ее вызова

*void device\_is\_busy(uint8\_t is\_busy);* // Функция для сообщения технологическому контроллеру о занятости

*void motor\_driver\_init(void);* // Функция инициализации управления драйвером

*void motor\_driver\_start(uint8\_t OCR0A\_value, uint8\_t direction);* // Функция запуска двигателя

*void motor\_driver\_stop(void);* // Функция остановки двигателя

*void motor\_calibration(void);* // Функция калибровки двигателя

Точка входа представляет собой функцию main, в которой инициализируется вся необходимая перефирия, разрешаются прерывания, производится калибровка моторов, после чего запускается бесконечный цикл, в котором происходит управление двигателем, в зависимости от того, какие данные получены с переферии.

*int main(void)*

{

*device\_is\_busy(1);* // Говорим, что устройство занято инициализацией

*p\_data32 = (uint8\_t \*) &data32;* // Инициализируем указатель на область памяти с data32

*ext\_INTERRUPT\_init();* // Инициализация внешних прерываний

*motor\_driver\_init();* // Инициализация таймера

*sei();* // Разрешаем прерывания

*motor\_calibration();* // Калибровка привода

*SPI\_SL\_init();* // Инициализируем SPI в качестве ведомого

*device\_is\_busy(0);* // Устройство готово к работе

*while (1)*

{

//------------------------------------------------------------------------------------------

*if (stop\_flag)* // Если сработал ДКП, все останавливаем

{

*motor\_driver\_stop();* // Если сработал ДКП останавливаем двигатель

*device\_is\_busy(1);* // Говорим, что устройство занято

*continue;*

}

//----------------------------------------------------------------------------------------

*if (SPI\_byte\_counter == 4)* // Если закончился прием данных...

{

*delta = data32 - encoder\_counter;* // Высчитываем дельту

*if (delta < 7 && delta > -7)* // Если дельта в пределах погрешности, то

{

*motor\_driver\_stop();* // Останавливаем двигатель

*SPI\_byte\_counter = 0;* // Обнуляем счетчик SPI

*device\_is\_busy(0);* // Говорим, что устройство готово к обемену

*continue;*

}

*else if (delta > 0)* // Если дельта больше нуля, то...

{

*motor\_driver\_start(MOTOR\_SPEED, 1);* // Вращаем в положительном направлении

}

*else* // Если дельта меньше нуля, то...

{

*motor\_driver\_start(MOTOR\_SPEED, 0);* // Вращаем в отрицательном направлении

}

}

}

}

Далее идет обработчик внешнего прерывания на ножке *PD2* (при срабатывание концевого датчика).

*ISR(INT0\_vect)* // Обработчик внешнего прерывания *INT0*

{

*stop\_flag = 1;* // Устанавливаем флаг остановки привода

}

Обработчик внешнего прерывания на ножке PD3 (работа энкодера).

*ISR(INT1\_vect)* // Обработчик внешнего прерывания *INT1*

{

*if (BitIsSet(PIND, 4)) encoder\_counter++;* // Если вторая ножка энкодера в 1, значит прибавляем

*else encoder\_counter--;* // Если вторая ножка энкодера в 0, значит отнимаем

}

Обработчик прерывания по приему данных на шине SPI.

*ISR(SPI\_STC\_vect)* // Обработчик прерывания по приему байта данных

{

*\*(p\_data32 + SPI\_byte\_counter) = SPDR;* // Записываем полученный байт в память

*SPI\_byte\_counter++;* // Увеличиваем счетчик адреса

*if (SPI\_byte\_counter == 4)*

{

*device\_is\_busy(1);* // Если приняли 4 байта, то приступаем к выполнению движения

}

}

Далее идет объевление функций, прототипы которых мы описали выше. Функция калибровки привода.

*void motor\_calibration(void)*

{

*motor\_driver\_start(MOTOR\_SPEED, 0);* // Запускаем двигатель в обратном направлении

*while(1)*

{

\_delay\_us(1);

*if (stop\_flag)* // Если сработал ДКП...

{

*motor\_driver\_stop();*  // Останавливаем двигатель

*stop\_flag = 0;* // Возвращаем в 0 флаг остановки

*encoder\_counter = 0;*  // Калибруем переменную-счетчик импульсов

*break;* // Калибровка завершена

}

*else* // Если нет, то ждем

{

*continue;*

}

}

}

Функция инициализации пинов и таймера, необходимых для работы двигателя.

*void motor\_driver\_init(void)*

{

*SetBit(DDRD, 7);* // Ножка драйвера "dir" на выход

*SetBit(DDRD, 6);* // OC0A на выход

*TCCR0A = (0<<COM0A1)|(1<<COM0A0)|(1<<WGM01)|(0<<WGM00);* // Выбираем режим СТС и переключение ножки OC0A при совпадении

}

Функция начала движения двигателя, которая принимает значение для регистра сравнения таймера 0 и направление (0 – назад, 1 – вперед).

*void motor\_driver\_start(uint8\_t OCR0A\_value, uint8\_t direction)*

{

*if (direction) SetBit(PORTD, 7);* // Если direction = 1, то пин dir в высокое состояние

*else ClearBit(PORTD, 7);* // Иначе в низкое

*OCR0A = OCR0A\_value;* // Задаем скорость

*TCCR0B = (1<<CS02)|(0<<CS01)|(1<<CS00);* // Запускаем таймер, предделитель 1024

}

Функция остановки двигателя.

*void motor\_driver\_stop(void)*

{

*TCCR0B = (0<<CS02)|(0<<CS01)|(0<<CS00);* // Останавливаем таймер

*TCNT0 = 0;* // Обнуляем счетчик таймера

}

Функция для общения микроконтроллера с технологическим контроллером, которая сообщает, выполнил ли микроконтроллер задание или нет, принимающее необходимое состояние контроллера (0 – микроконтроллер готов к работе, 1 – микроконтроллер занят).

*void device\_is\_busy(uint8\_t is\_busy)*

{

*if (is\_busy)*

{

*SetBit(DDRD, 0);* // Опускаем статус-шину, сообщая, что устройство занято

*ClearBit(PORTD, 0);*

}

*else*

{

*ClearBit(DDRD, 0);* // Отпускаем статус-шину, сообщая, что устройство готово к обмену данными

*SetBit(PORTD, 0);* // Подтягиваем ее к 1 через резистор

}

}

Функция инициализации внешних прерываний.

*void ext\_INTERRUPT\_init(void)* // Инициализация внешних прерываний INT0 и INT1

{

*ClearBit(DDRD, 2);* // Настраиваем ножку с прерыванием по срабатыванию ДКП на вход

*SetBit(PORTD, 2);* // Подтяжка к 1

*EICRA = (1<<ISC01)|(0<<ISC00);* // Прерывание при спадающем фронте на PD2

*SetBit(EIMSK, INT0);* // Разрешаем прерывание INT1

*ClearBit(DDRD, 3);* // Настраиваем ножку на вход

*SetBit(PORTD, 3);* // Подтяжка к 1

*EICRA = (1<<ISC11)|(0<<ISC10);* // Прерывание при спадающем фронте на PD3

*SetBit(EIMSK, INT1);*  // Разрешаем прерывание INT1

}

Далее будет рассмотрено содержимое файла “main.h” весь файл заключен в структуру условной компиляции, чтобы определение файла компилятором производилось только один раз.

*#ifndef MAIN\_H\_*

*#define MAIN\_H\_*

*#endif /\* MAIN\_H\_ \*/*

Определяем частоту тактирования микроконтроллера.

*#define F\_CPU 16000000UL* // Частота процессора

Подключаем стандартные библиотеки.

*#include <avr/io.h>*

*#include <stdio.h>*

*#include <util/delay.h>*

*#include <avr/interrupt.h>*

*#include <stdlib.h>*

Подключает заголовочный файл модуля для работы с шиной SPI.

*#include "SPI.h"* // Подключаем библиотеку работы с LCD

Определяем макросы побитной записи / проверки битов в регистре.

*#define SetBit(reg, bit) reg |= (1<<(bit))* // установка бита в регистре

*#define ClearBit(reg, bit) reg &= (~(1<<bit))* // очистка бита в регистре

*#define BitIsClear(reg, bit) ((reg & (1<<bit)) == 0)* // проверка бита на 0

*#define BitIsSet(reg, bit) ((reg & (1<<bit)) != 0)* // проверка бита на 1

*#define InvBit(reg, bit) reg ^= (1<<(bit))* // инвентировать бит регистра

Также определяем макросы доступа к регистрам *DDR* и *PIN* портов ввода-вывода, по регистру *PORT*

*#define DDR(port) (\*((&port) - 1))*

*#define PIN(port) (\*((&port) - 2))*

Файл SPI.h также заключен в директиву условной компиляции.

*#ifndef SPI\_H\_*

*#define SPI\_H\_*

*#endif /\* SPI\_H\_ \*/*

Подключение заголовочного файла “main.h”

*#include "main.h"*

В данном файле производятся объявление прототипов функций, реализация которых представлена в файле “SPI.c”

/\*Инициализация *SPI* интерфейса в качестве ведомого устройства\*/

*void SPI\_SL\_init(void);*

Файл *SPI.c* начинается с подключения заголовочного файла *SPI.h*

*#include "SPI.h"*

Далее идет реализации функций, описанных в данном модуле. Функция инициализации интерфейса SPI в качестве ведомого устройства.

*void SPI\_SL\_init(void)*

{

*DDRB &= ~((1<<PORTB2)|(1<<PORTB3)|(1<<PORTB5));* // MOSI, SCK, SS на вход

*SetBit(DDRB, 4);* // MISO на выход

*ClearBit(PORTB, 4);*  // 0 на MISO

*SetBit(SPSR, SPI2X);* // Включаем удвоение частоты тактирования SPI

*SPCR = (1<<SPIE)|(1<<SPE)|(0<<MSTR);* // Включаем SPI, прерывания, выбираем контроллер в качетсве ведомого

}

Для успешной компиляции проекта необходимо также подключить файлы с расширением .c в *Solution Explorer* в среде *Atmel Studio*. После чего структура модулей будет выглядеть как показано на рисунке 4.2.1.1.

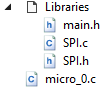


Рисунок 4.2.1.1 – Структура программы для микроконтроллера

4.2.2 Разработка программы управления для технологического контроллера

Программа управления микроконтроллером приводов была написана на языке программирования Си с использованием проприетарной интегрированной среды разработки *(IDE) Atmel Studio 7*.

Программа представлена несколькими модулями, состоящими из заголовочных файлов (*.h*) и Си файлов, содержащих исполняемый код (*.с*). Рассмотрим главный файл прошивки *“main\_micro.c”.*

В самом начале производится включение в модуль заголовочного файла *“main.h”,* в котором производятся подключение уже всех остальных модулей и определение макросов.

*#include "Libraries/main.h"*

Объявляются все переменные, видимые глобально в этом модуле.

*uint8\_t byte\_counter = 0;* // Счетчик принятый байтов

*uint32\_t data32;* // Переменная для данных

*uint32\_t* *coordinates[4];* // Массив для данных, полученных по UART

*uint8\_t* *\* p\_coordinate\_1;* // Указатель на первый байт первой координаты

Т.к. функции, использованные в модуле, объявлены после точки входа, нам необходимо объявить их прототипы.

*void wait\_while\_devices\_busy(void);*

Точка входа представляет собой функцию main, в которой инициализируется вся необходимая периферия, разрешаются прерывания, после чего запускается бесконечный цикл, в котором происходит отправка координат по всем приводам, в зависимости от того, какие данные получены по *UART*.

*int main(void)*

{

*p\_coordinate\_1 = (uint8\_t \*)coordinates;* // Инициализируем указатель на

*SPI\_MS\_init();* // Инициализируем SPI как ведущее устрйство

*USART\_Init(16);* // Инициализируем USART, скорость 115200

*USART\_Transmit(120);* // Просим у комьютера координаты для приводов

*wait\_while\_devices\_busy();* // Ждем пока устройства освободятся

*while (1)*

{

*if (byte\_counter == 16)*

{

*byte\_counter = 0;*

*SPI\_MS\_send\_uint32\_t(p\_coordinate\_1, SPI\_slave1\_pin);*  // Передаем координаты первому устройству

*SPI\_MS\_send\_uint32\_t((p\_coordinate\_1 + 4), SPI\_slave2\_pin);* // Передаем координаты второму устройству

*SPI\_MS\_send\_uint32\_t((p\_coordinate\_1 + 8), SPI\_slave3\_pin);* // Передаем координаты третьему устройству

*SPI\_MS\_send\_uint32\_t((p\_coordinate\_1 + 12), SPI\_slave4\_pin);* // Передаем координаты четвертому устройству

*USART\_Transmit(120);* // Говорим компьютеру, что готовы получить следующие данные

*wait\_while\_devices\_busy();* // Ждем, пока все устройства выполнят свою работу

*\_delay\_us(20);*

}

}

}

Обработчик прерывания по получению байта данных по *UART*.

*ISR(USART\_RX\_vect)*

{

*\*(p\_coordinate\_1 + byte\_counter) = UDR0;*

*byte\_counter++;*

}

Объявление функции, которая ожидает, пока все микроконтроллеры сообщат о завершении прошлого задания.

*void wait\_while\_devices\_busy(void)*

{

*DDRD = 0b00001111;* // Ножки 4-7 на вход

*PORTD |= 0b11110000;* // Подтяжка к 1 на ножках 4-7

*while(1)* // Ждем пока все устройства будут готовы

{

*\_delay\_us(20);*

*if (((PIND & 0b11110000) == 0b11110000))*

{

*break;*

}

}

}

Далее будет рассмотрено содержимое файла “main.h” весь файл заключен в структуру условной компиляции, чтобы определение файла компилятором производилось только один раз.

*#ifndef MAIN\_H\_*

*#define MAIN\_H\_*

*#endif /\* MAIN\_H\_ \*/*

Определяем частоту тактирования микроконтроллера.

*#define F\_CPU 16000000UL* // Частота процессора

Подключаем стандартные библиотеки.

*#include <avr/io.h>*

*#include <stdio.h>*

*#include <util/delay.h>*

*#include <avr/interrupt.h>*

*#include <stdlib.h>*

Подключает заголовочный файл модуля для работы с шиной SPI и USART.

*#include "SPI.h"* // Подключаем библиотеку работы с LCD

*#include "USART.h"*  // Библиотека для настроек USART

Определяем макросы побитной записи / проверки битов в регистре.

*#define SetBit(reg, bit) reg |= (1<<(bit))* // установка бита в регистре

*#define ClearBit(reg, bit) reg &= (~(1<<bit))* // очистка бита в регистре

*#define BitIsClear(reg, bit) ((reg & (1<<bit)) == 0)* // проверка бита на 0

*#define BitIsSet(reg, bit) ((reg & (1<<bit)) != 0)* // проверка бита на 1

*#define InvBit(reg, bit) reg ^= (1<<(bit))* // инвентировать бит регистра

Также определяем макросы доступа к регистрам *DDR* и *PIN* портов ввода-вывода, по регистру *PORT*

*#define DDR(port) (\*((&port) - 1))*

*#define PIN(port) (\*((&port) - 2))*

Файл SPI.h также заключен в директиву условной компиляции.

*#ifndef SPI\_H\_*

*#define SPI\_H\_*

*#endif /\* SPI\_H\_ \*/*

Подключение заголовочного файла “main.h”

*#include "main.h"*

Определяем подстановок с определением ножек, к которым будут подключены ножки синхронизации ведомых устройств для облегчения процесса отладки и изменения прошивки.

/\* Определение ножек, на которых будут висеть ведомые устройства\*/

*#define SPI\_slaves\_port PORTC* // Порт, к которому подключены ножки SS ведомых устройств

*#define SPI\_slave1\_pin 0* // Ножка с первым ведомым устрйоством

*#define SPI\_slave2\_pin 1* // Ножка со вторым ведомым устрйоством

*#define SPI\_slave3\_pin 2* // Ножка с третьим ведомым устрйоством

*#define SPI\_slave4\_pin 3* // Ножка с четвертым ведомым устрйоством

В данном файле производятся объявление прототипов функций, реализация которых представлена в файле *“SPI.c”*

/\*Инициализация *SPI* интерфейса в качестве мастера\*/

*void SPI\_MS\_init(void);*

/\*Передача мастером байта данных. Возвращает полученный от ведомого байт\*/

*uint8\_t SPI\_MS\_send\_byte(uint8\_t byte);*

/\*Передача мастером 4 байта данных. Возвращает полученный от ведомого байт\*/

*void SPI\_MS\_send\_uint32\_t(uint8\_t \* p8\_data32, uint8\_t slaves\_pin);*

Файл SPI.c начинается с подключения заголовочного файла *SPI.h*

*#include "SPI.h"*

Далее идет реализации функций, описанных в данном модуле. Функция инициализации интерфейса *SPI* в качестве ведущего устройства.

*void SPI\_MS\_init(void)*

{

*DDRB |= (1<<PORTB3)|(1<<PORTB5);* // *MOSI*, *SCK* на выход

*PORTB &= ~((1<<PORTB3)|(1<<PORTB5));* // Низкий уровень на *MOSI*, *SCK*

*DDR(SPI\_slaves\_port) |= (1<<SPI\_slave1\_pin) | (1<<SPI\_slave2\_pin) | (1<<SPI\_slave3\_pin)| (1<<SPI\_slave4\_pin);* // На выход ножки, к которым подключены пины SS ведомых устройств

*SPI\_slaves\_port |= (1<<SPI\_slave1\_pin)|(1<<SPI\_slave2\_pin)|(1<<SPI\_slave3\_pin)| (1<<SPI\_slave4\_pin);*  // Высокий уровень на всех ножках с SS

*ClearBit(DDRB, 4);* // MISO на вход

*SetBit(SPSR, SPI2X);* // Включаем удвоение частоты тактирования SPI

*SPCR = (1<<SPE)|(1<<MSTR);* // Включаем SPI выбираем контроллер в качестве мастера

}

Функция отправки одного байта данных по SPI. Принимает байт данных, который необходимо отправить.

uint8\_t SPI\_MS\_send\_byte(uint8\_t byte)

{

*SPDR = byte;* // Начинаем передачу данных

*while(BitIsClear(SPSR, SPIF));* // Ждем пока завершится передача данных

*return SPDR;* // Возвращаем полученный байт

}

Функция отправки 4-х байтовой переменной uint32-t определенному ведомому устройству. Принимает указатель на первый байт данных и номер пина, к которому подключено ведомое устройство.

*void SPI\_MS\_send\_uint32\_t(uint8\_t \* p8\_data32, uint8\_t slaves\_pin)*

{

*ClearBit(SPI\_slaves\_port, slaves\_pin);* // Говорим, с каким устройством мы хотим общаться

*SPI\_MS\_send\_byte(\*p8\_data32++);*  // отправляем первый байт

*\_delay\_us(delay\_between\_bytes\_us);* // Задрежка, чтобы контроллер успел обработать данные

*SPI\_MS\_send\_byte(\*(p8\_data32++));* // отправляем второй байт

*\_delay\_us(delay\_between\_bytes\_us);* // Задрежка, чтобы контроллер успел обработать данные

*SPI\_MS\_send\_byte(\*(p8\_data32++));* // отправляем третий байт

*\_delay\_us(delay\_between\_bytes\_us);* // Задрежка, чтобы контроллер успел обработать данные

*SPI\_MS\_send\_byte(\*(p8\_data32));* // отправляем четвертый байт

*SetBit(SPI\_slaves\_port, slaves\_pin);* // Возвращаем высокий уровень на ножку *SS*

}

Файл USART.h также заключен в директиву условной компиляции.

*#ifndef USART\_H\_*

*#define USART\_H\_*

*#endif /\* USART\_H\_ \*/*

Подключение заголовочного файла “main.h”

*#include "main.h"*

В данном файле производятся объявление прототипов функций, реализация которых представлена в файле *“USART.c”*

/\*Инициализация *USART* с выбором скорости через регистр *UBBR*\*/

*void USART\_Init(unsigned int reg\_UBRR\_value);*

/\*Отправляет один байт данных в *UART*\*/

*void USART\_Transmit(unsigned char data);*

/\*Отправляем последовательно набор символов из переданной строки\*/

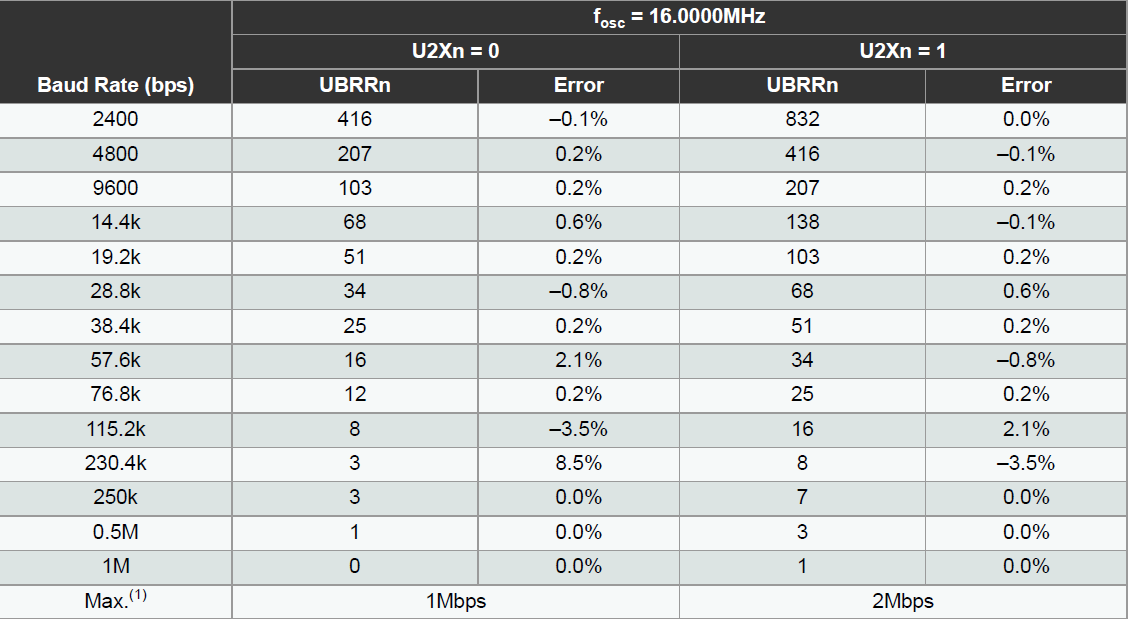
*void USART\_sendStr(char \*string);*

Файл *USART.c* начинается с подключения заголовочного файла *USART.h*

*#include "USART.h"*

Далее идет реализации функций, описанных в данном модуле. Функция инициализации интерфейса *USART*, которая принимает значение регистра *UBRR*, задающего скорость *USART*. Зависимость скорости работы *USART* от значения *UBRR* при частоте тактирования 16МГц представлена в таблице 16.

Таблица 16 – Зависимость скорости работы *USART* от значения регистра *UBRR*.



*void USART\_Init(unsigned int reg\_UBRR\_value)* // Инициализация *USART*

{

*UBRR0 = reg\_UBRR\_value;*

*UCSR0B = (1<<RXEN0)|(1<<TXEN0);* // Включаем прием и передачу данных по USART

*UCSR0B |= (1<<RXCIE0);* // Разрешаем прерывания при приеме

*UCSR0A |= (1<<U2X0);* // Удваиваем скорость передачи

*UCSR0C = (1<<UCSZ01)|(1<<UCSZ00);* // Выбираем 8-битный режим предачи данных

*UCSR0C |= (1<<USBS0);* // Выбираем два стоп-бита

*asm("nop");*

}

Функции отправки одного байта данных по *USART*. Принимает байт, необходимый для отправки.

*void USART\_Transmit(unsigned char data)*

{

*while ( !(UCSR0A & (1<<UDRE0)) );* // Ждем пока регистр приема-передачи данных освободится

*UDR0 = data;* // Начало передачи данных

}

Функция отправки строчного массива по *USART*. Принимает указатель на первый байт массива (название массива).

*void USART\_sendStr(char \*string)*

{

*while (\*string)* // Пока не наткнемся на 0, означающий конец строчного массива

{

*USART\_Transmit(\*string++);* // Передаем в *UART* текущий байт нашего массива, после чего инкрементируем указатель

}

}

Для успешной компиляции проекта необходимо также подключить файлы с расширением .c в *Solution Explorer* в среде *Atmel Studio*. После чего структура модулей будет выглядеть как показано на рисунке 4.2.2.2.

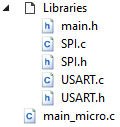


Рисунок 4.2.2.2 – Структура программы для технологического контроллера

4.2.3 Разработка программы управления для персонального компьютера

Программа управления манипулятором была написана в среде разработки *LabVIEW*.

Лицевая панель представлена на рисунке 4.2.3.1 и рисунке 4.2.3.2



Рисунок 4.2.3.1 – Лицевая панель программы. Автоматический режим

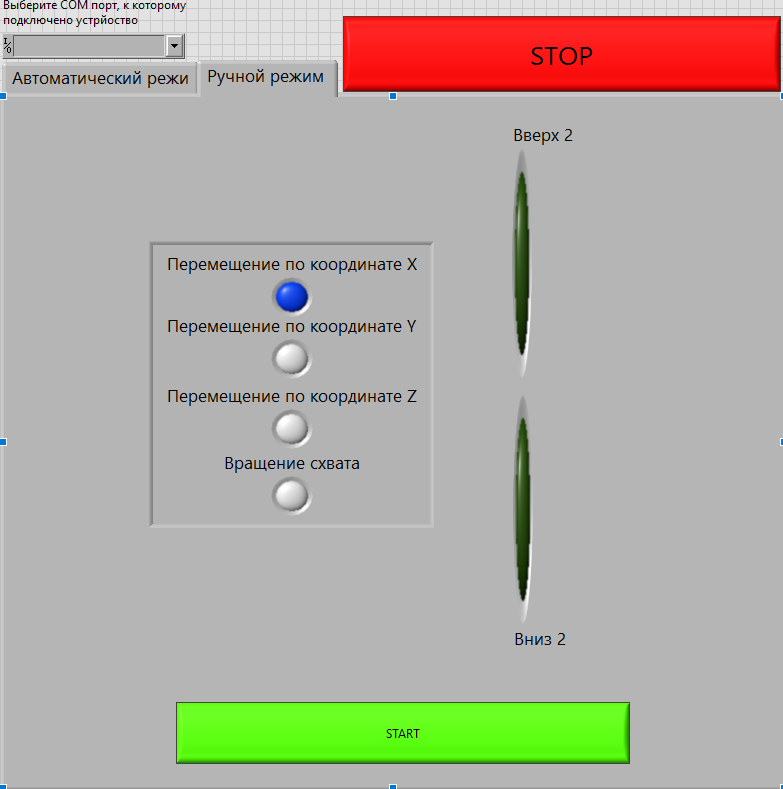


Рисунок 4.2.3.2 – Лицевая панель программы. Ручной режим

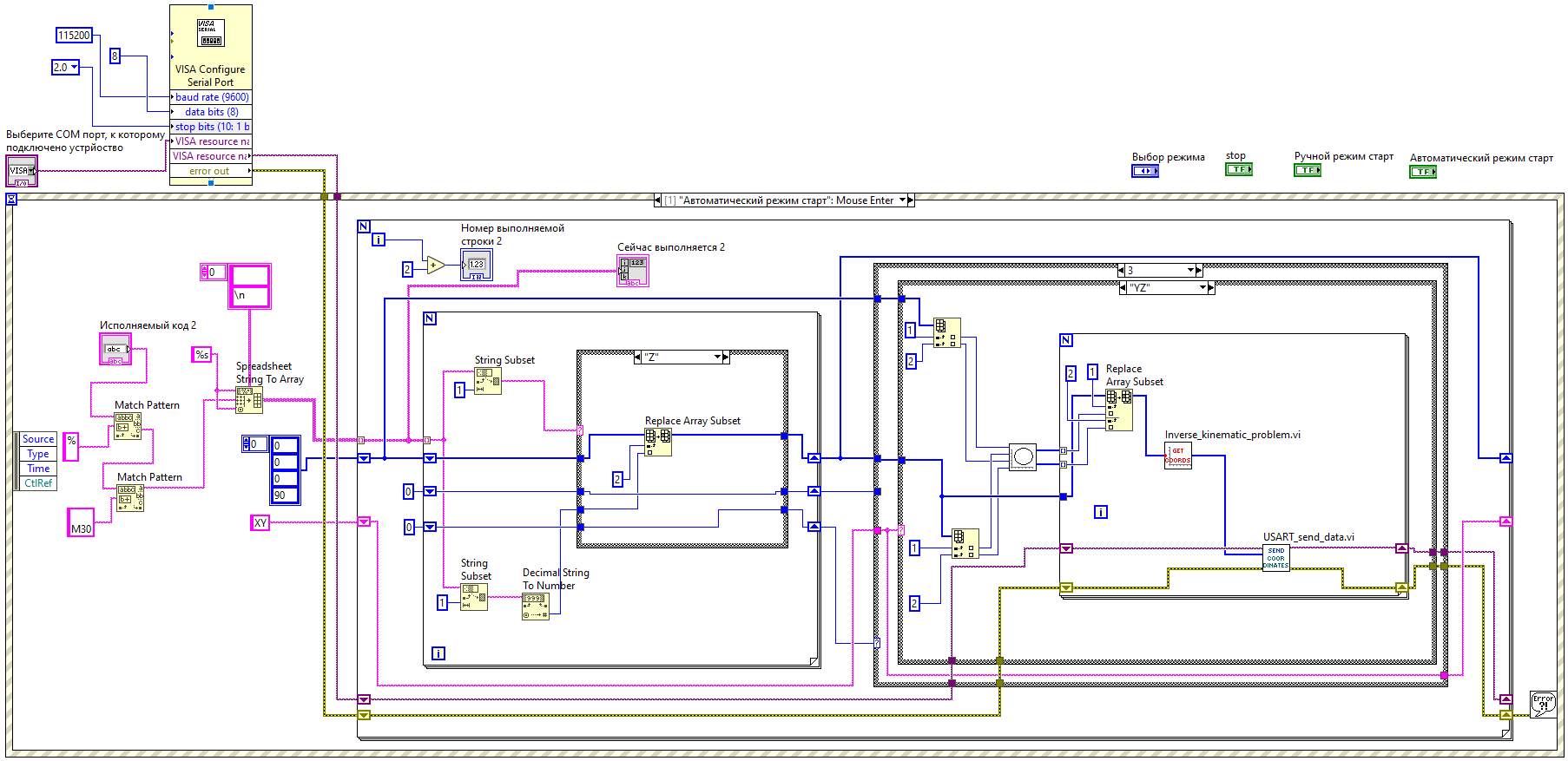


Рисунок 4.2.3.3 – Блок-диаграмма программы управления манипулятором

На рисунке 4.2.3.3 представлен внешний вид программы управления, рассмотрим по частям все этапы выполнения программы.

В начале идет конфигурация порта, где мы задаем его характеристики, для общения с микроконтроллером. Показано на рисунке 4.2.3.4.

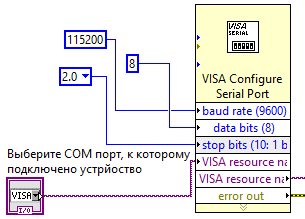


Рисунок 4.2.3.4 – Настройка *COM* порта

Далее, на рисунке 4.2.3.5, вся программа заключена в структуру *Event Structure*, которая выполняет запуск той или иной части кода, в зависимости от события нажатия кнопок на лицевой панели. При нажатии кнопки *“Start”* на лицевой панели в автоматическом режиме. Выполняется следующий код:

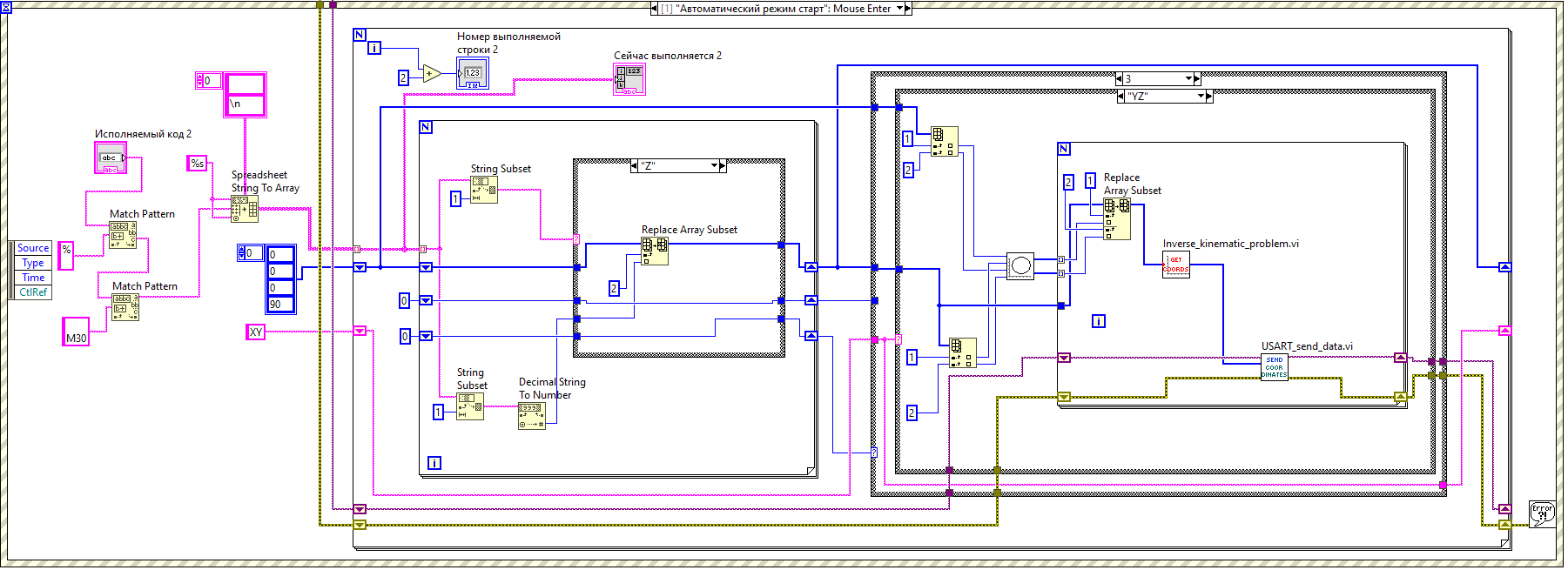


Рисунок 4.2.3.5 – Блок-диаграмма. Автоматический режим

При нажатии кнопки *“Start”* на лицевой панели в ручном режиме, запустится бесконечный цикл, представленный на рисунке 4.2.3.6:

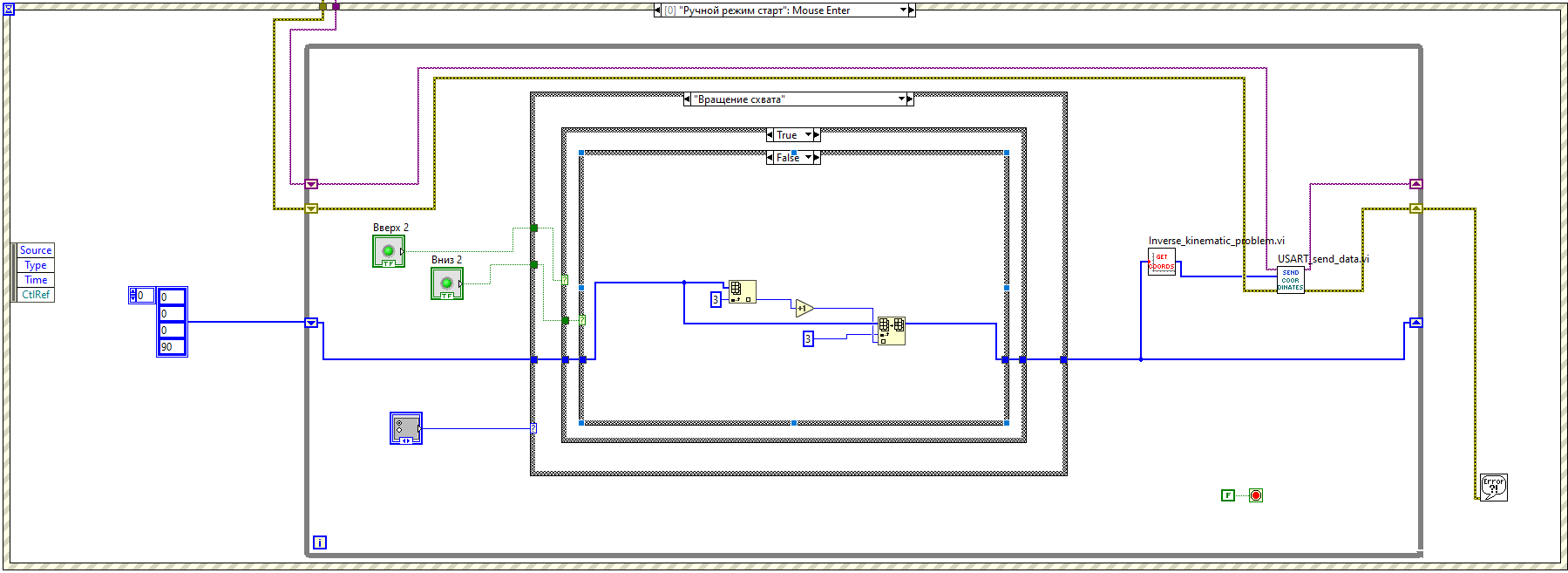


Рисунок 4.2.3.6 – Блок-диаграмма. Ручной режим

При нажатии кнопки *“Stop”* на лицевой панели выполнение кода остановится и пользователю выдастся сообщение с возможными дальнейшими действиями, как показано на рисунке 4.2.3.6.

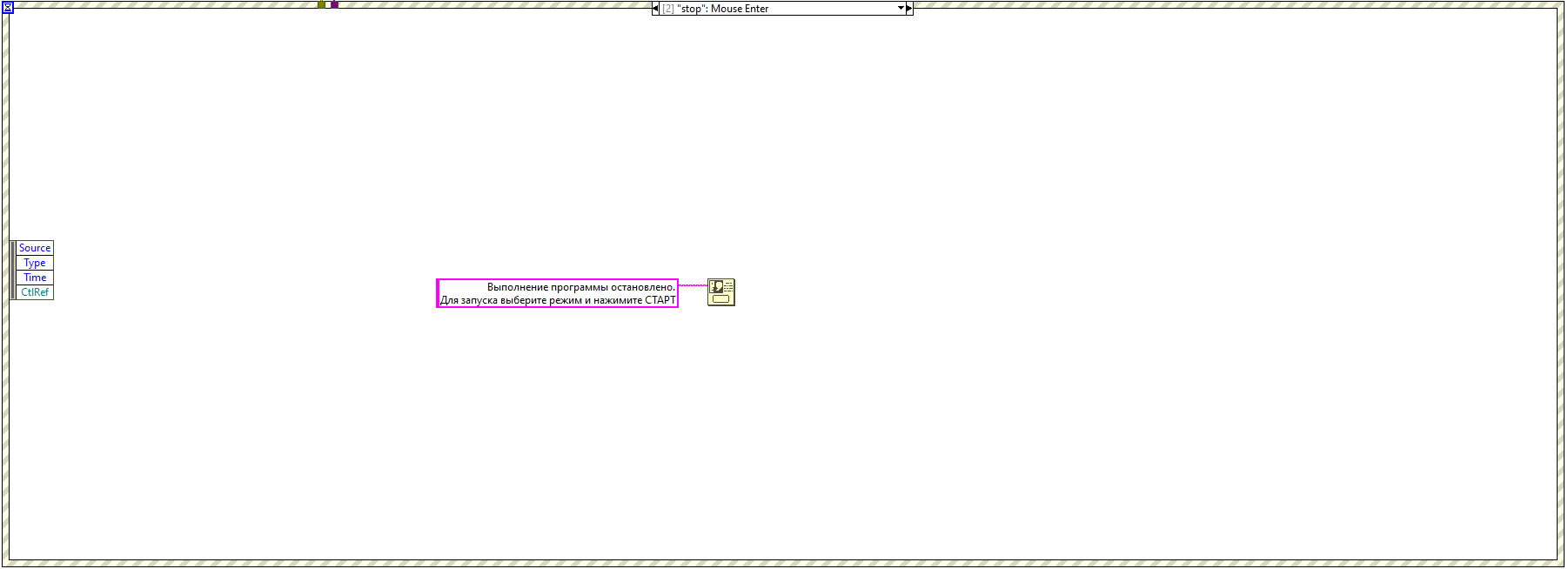


Рисунок 4.2.3.6 – Блок-диаграмма. Режим стоп

Рассмотрим автоматический режим, показанный на рисунке 4.2.3.7. Вначале происходит работа с введенным *G* кодом представленном в виде строки. Определяется начало и конец программы, а происходит разбиение строки на строковый массив сперва по символу переноса строки, а потом по символу пробела. На выход получается двухмерный строковый массив. Также инициализируется координаты начального положения и рабочая плоскость.

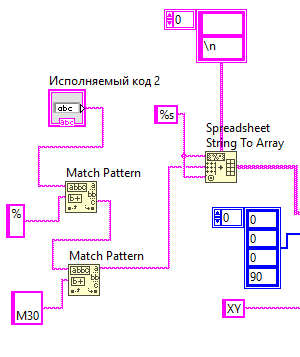


Рисунок 4.2.3.7 – Парсинг *G* кода и инициализация начальных параметров

Далее, как показано на рисунке 4.2.3.8, идет структура *For loop*, которая проходится по всем строкам массивы (которые были разбиты ранее по символу переноса строки. То есть мы проходимся по каждой отдельной команде из G кода.

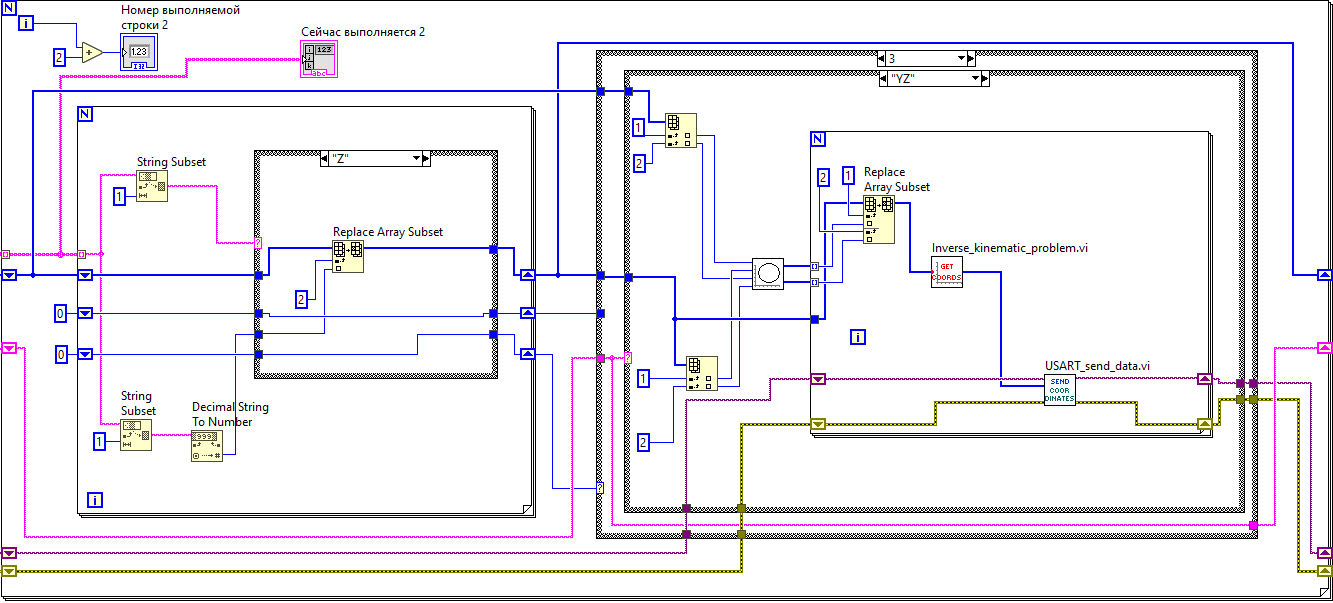


Рисунок 4.2.3.8 – Обработка G команд

Далее, как показано на рисунке 4.2.3.9, идет еще одна структура *For loop* которая проходит по каждому элементу полученного одномерного строчного массива (по строке, которая разбита пробелами). В данном цикле определяется номер команды *G*-кода, а также входные параметры для этой команды, по букве, которая идет перед параметром.

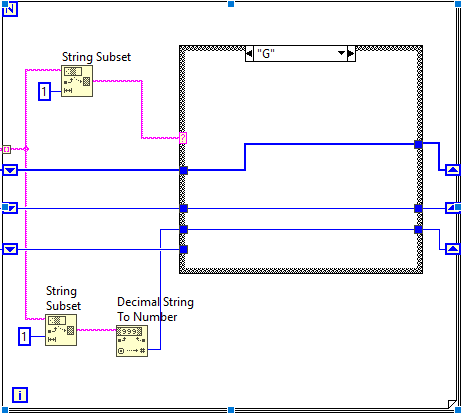
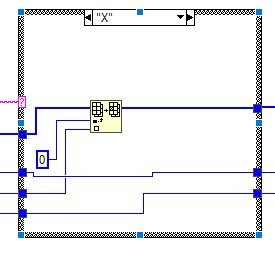
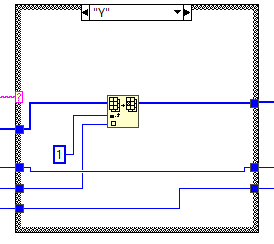
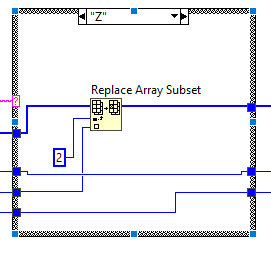
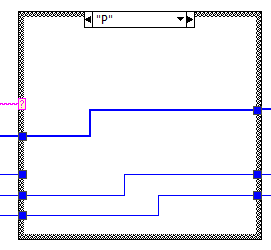


Рисунок 4.2.3.9 – Структура, определяющая номер *G* команды и ее параметры

Далее, на рисунке 4.2.3.10, представлены все действия в структуре выбора, выполняемые в зависимости от имени параметра:

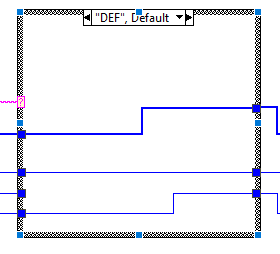


Рисунок 4.2.3.10 – Варианты действий в структуре выбора, в зависимости от имени входного параметра

Далее, на рисунке 4.2.3.11, идет еще одна структура выбора, которая уже непосредственно исполняет *G* команду в соответствии с ее номером и параметрами:

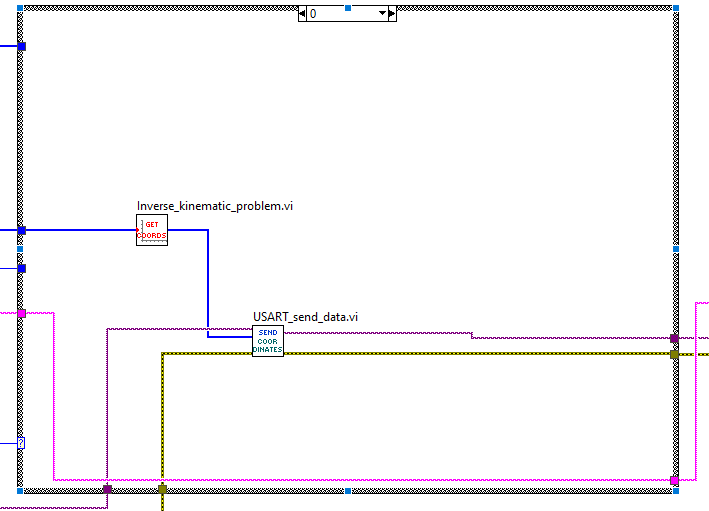


Рисунок 4.2.3.11 – Структура выбора, выполняющая *G* команду с номером 0

На рисунке 4.2.3.11 представлено выполнение *G* команды 0 (холостой ход к заданной точке). Здесь также представлено два ВПП. Один решает обратную задачу кинематики, для перехода от координат *XYZ* к обобщенным координатам манипулятора. А другой отправляет вычисленные значение на технологический контроллер

Рассмотрим подробнее выполнение остальных *G* – команд.

G – команда под номером 4 выполняет задержку, заданную параметром *P*

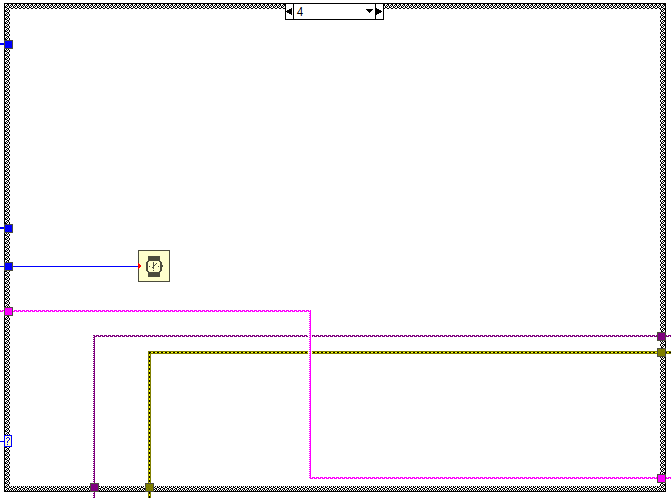


Рисунок 4.2.3.12 – Исполнение *G04*

*G* – команда под номером 17 выполняет смену рабочей плоскости на *XY*. *G* – команда под номером 18 выполняет смену рабочей плоскости на *XZ*

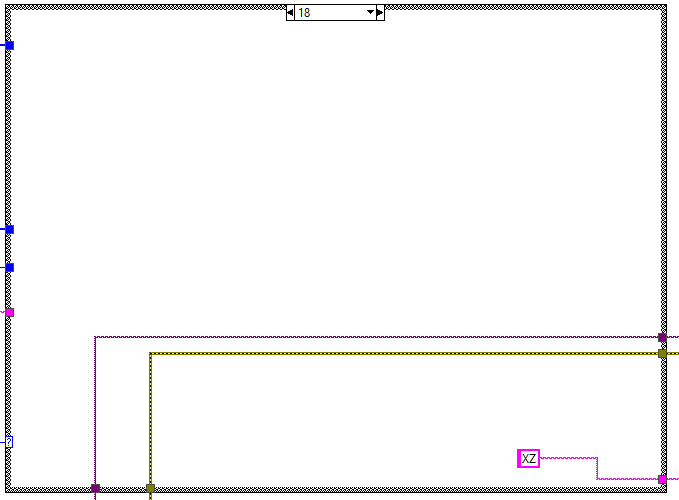
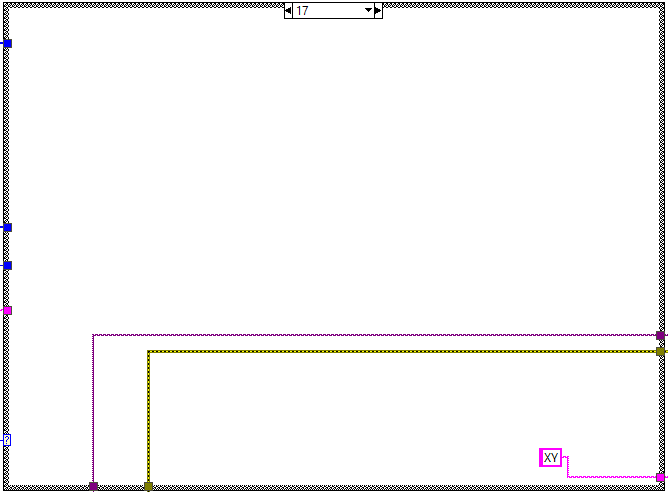


Рисунок 4.2.3.13 – Выполнение команды *G17* и *G18*

*G* – команда под номером 19 выполняет смену рабочей плоскости на *YZ.*

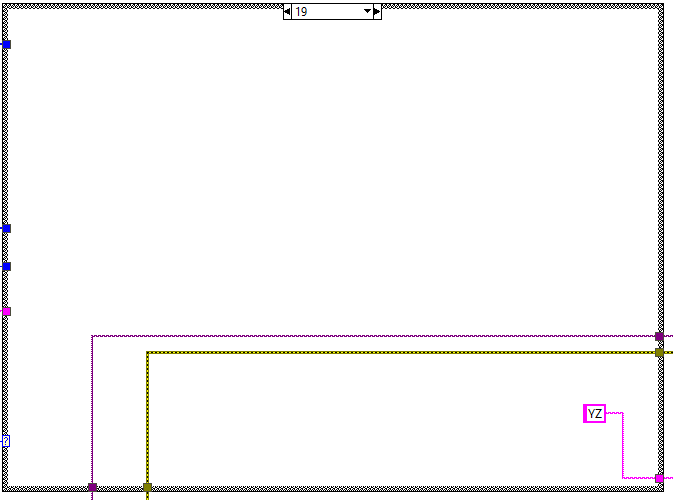


Рисунок 4.2.3.13 – Выполнение команды *G19*

При вводе команды, которая не поддерживается системой управления, пользователю выдастся предупреждение, c выбором дальнейших действий.

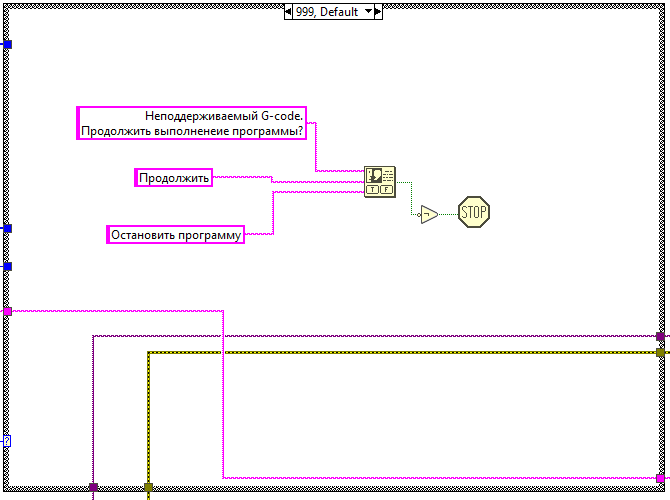


Рисунок 4.2.3.14 – Случай не распознанной команды

Команда под номером 1 представляет из себя линейную интерполяцию, в рабочей плоскости, между заданными координатами в параметрах и предыдущим состоянием системы. Рассмотрим ее реализацию.

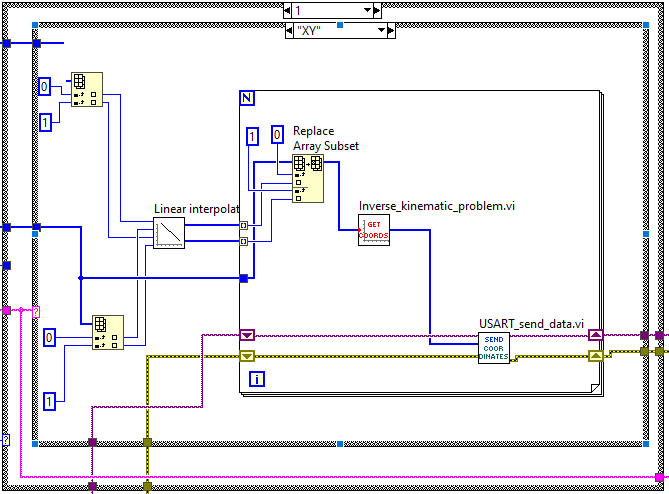


Рисунок 4.2.3.15 – Выполнение линейной интерполяции

Как видно на рисунке 4.2.3.15, есть структура выбора, которая зависит от текущей рабочей плоскости (их можно менять командами *G17-19*). Сперва мы достаем из массива с новыми координатами и из массива с прошлого состояния (для этого используется сдвиговый регистр) и передаем их в функцию линейной интерполяции, которая возвращает массив из координт *X* и *Y*. Далее идет структура For loop, которая проходится по всем элементам этих массивов, на каждом элементе запуская манипулятор для достижения всех промежуточных точек. Реализация для плоскостей *XZ* и *YZ* ничем не отличается, за исключением выборки координат из входных массивов. Реализация данных случаев представлена на рисунке 4.2.3.16.

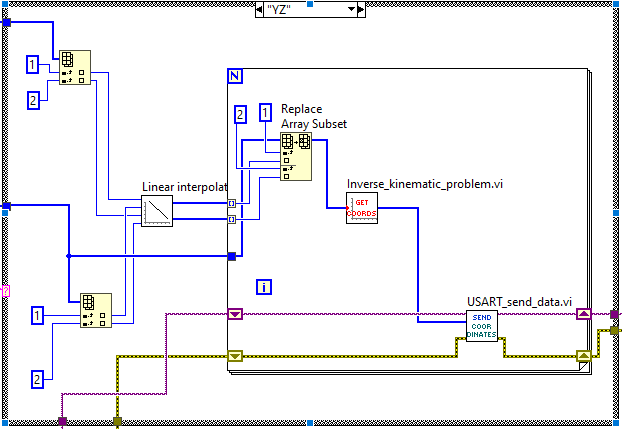
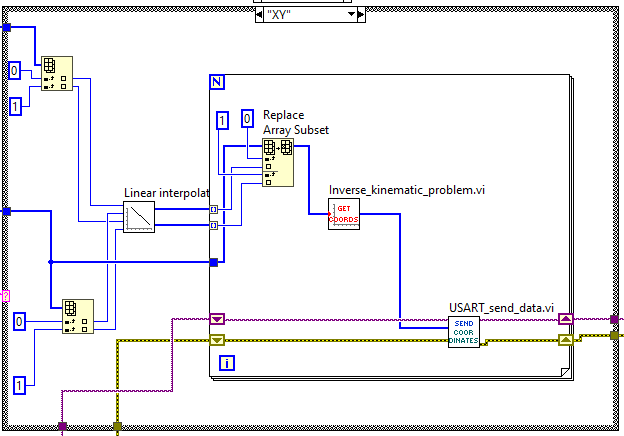


Рисунок 4.2.3.16 – Линейная интерполяция в плоскостях *XY* и *YZ*

Команда под номером 1 представляет из себя круговую интерполяцию, в рабочей плоскости, между заданными координатами в параметрах и предыдущим состоянием системы. Рассмотрим ее реализацию.

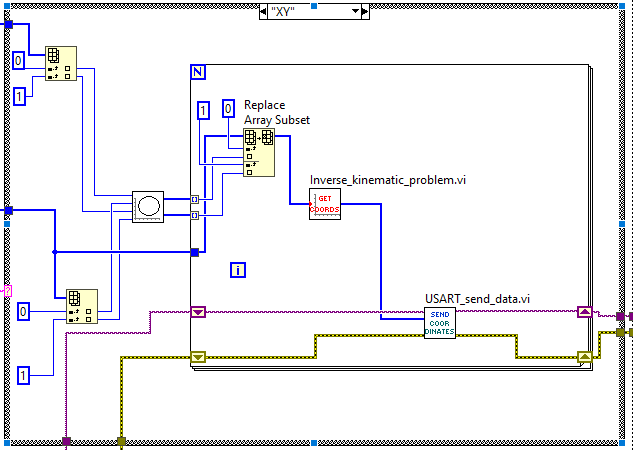


Рисунок 4.2.3.17 – Реализация круговой интерполяции

Реализация очень схожа с линейной интерполяцией, за исключением только того, что координаты передаются в другую функцию, которая реализует круговую интерполяцию, возвращает также два массива с координатами *X* и *Y*. Далее действия аналогичны, как и при линейной интерполяции.

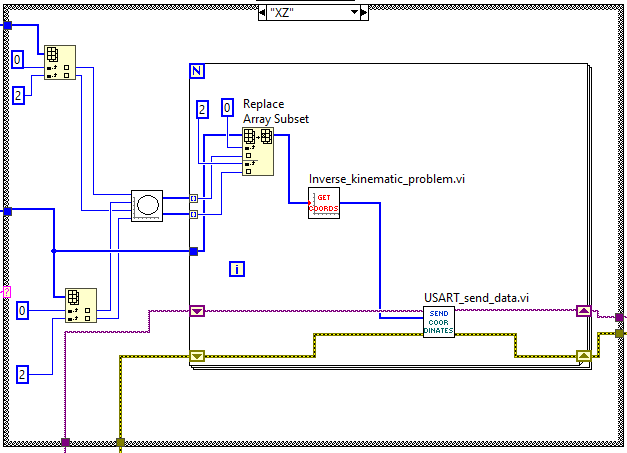
 

Рисунок 4.2.3.18 – Круговая интерполяция в плоскостях *XY* и *YZ*

Рассмотрим реализацию обратной задачи кинематики. Блок-диаграмма ВПП, реализовывающего данный функционал для манипулятора представлена на рисунке 4.2.3.19. Т.к. кинематика устройства довольна проста для расчетов, то было решено что применять такие методы решения обратной задачи кинематики, как метод обратных преобразований, метод штрафных функций, метод интервалов или общий подход к решению обратной задачи кинематики с помощью конечного поворота или смещения не целесообразно. Было принята простая модель, где координата *z* совпадает с обобщенной координатой привод подъема манипулятора, а при переводе координат *X* и *Y* в полярную систему координат, получившиеся угол и радиус и есть ни что иное, как обобщенные координаты привода поворота и привода выдвижения [7].

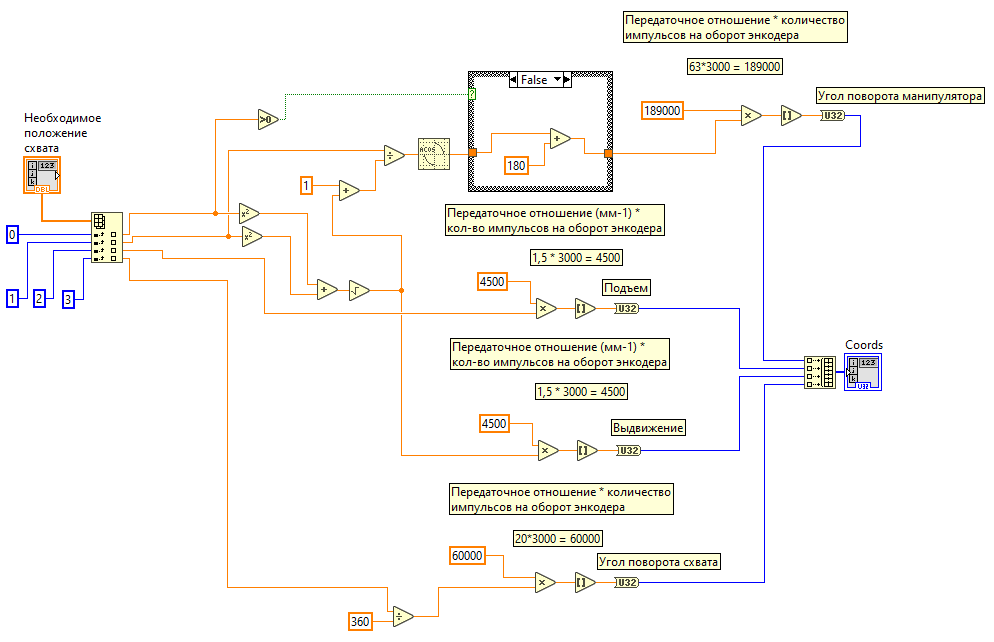


Рисунок 4.2.3.19 – ВПП решающий обратную задачи кинематики для манипулятора

Рассмотрим реализацию функции отправки координат на технологический контроллер. Сперва ожидается прием байта подтверждения, который представляет число 120, означающий, что технологический контроллер готов принять новые координаты, после чего идет их отправка.

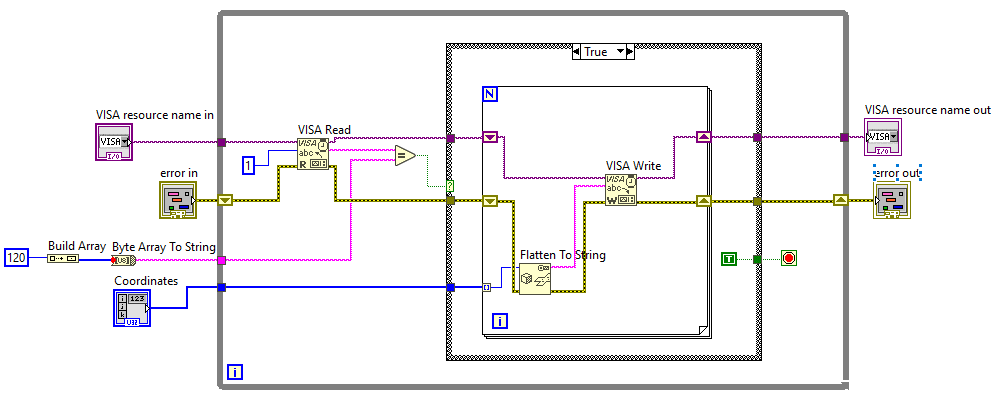


Рисунок 4.2.3.20 – ВПП, реализующий отправку четырех координат на технологический контроллер

Рассмотрим реализацию функции линейной интерполяции:

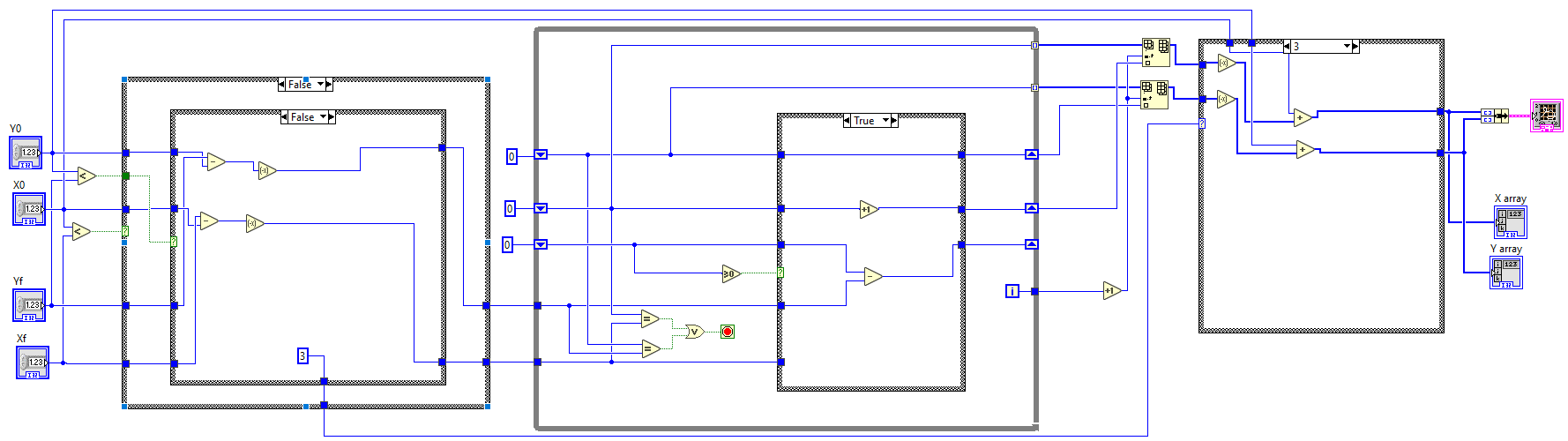


Рисунок 4.2.3.21 – ВПП, реализующий функцию линейной интерполяции

Функция круговой интерполяции:

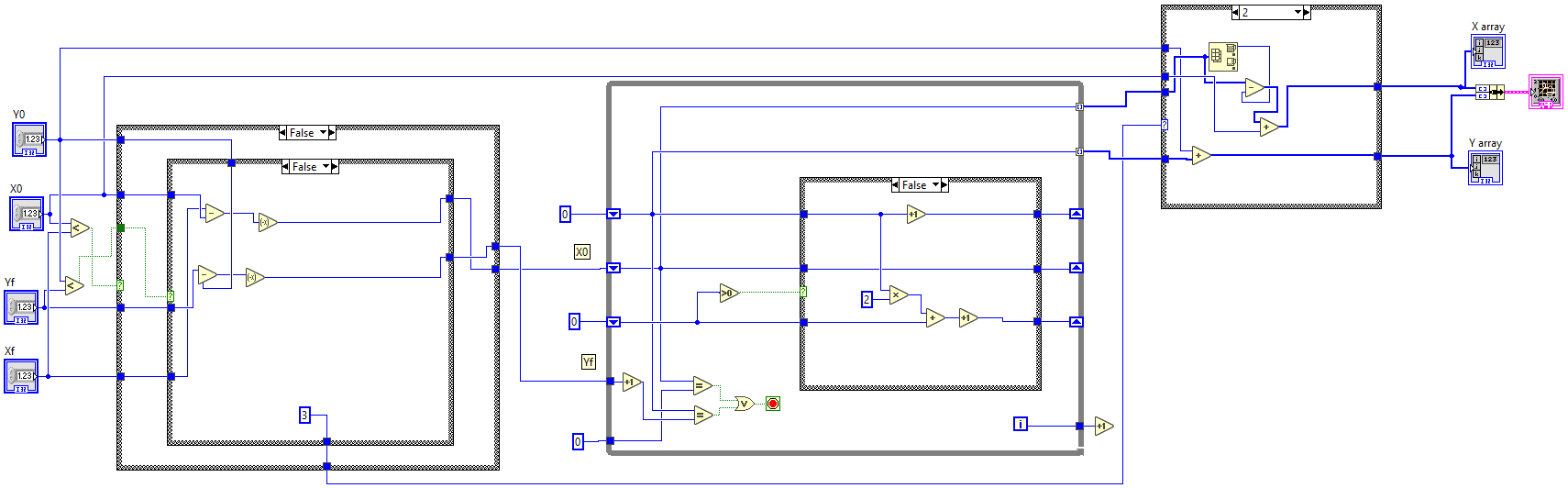


Рисунок 4.2.3.22 – ВПП, реализующий функцию круговой интерполяции

Рассмотрим ручной режим. В ручном режиме запускается бесконечный цикл, котором, в зависимости от факта нажатия кнопки вверх или вниз на лицевой панели, происходит перемещение привода на 1мм по выбранной оси.

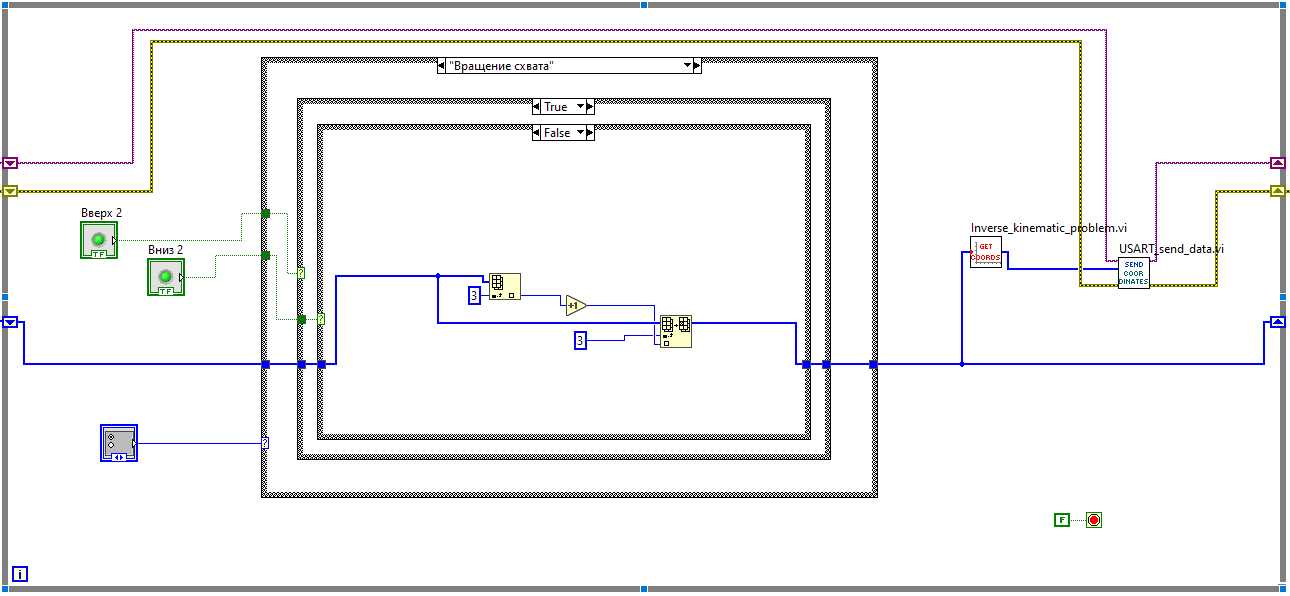


Рисунок 4.2.3.23 – Бесконечный цикл в ручном режиме управления манипулятором

Внутри присутствует структура выбора, в которой выбирается, по какой координате будет происходить перемещение, после чего данные отправляются на ТК.

## 4.3 Инструкция оператора

На рисунке 4.3.1 показана лицевая панель программы управления приводом.

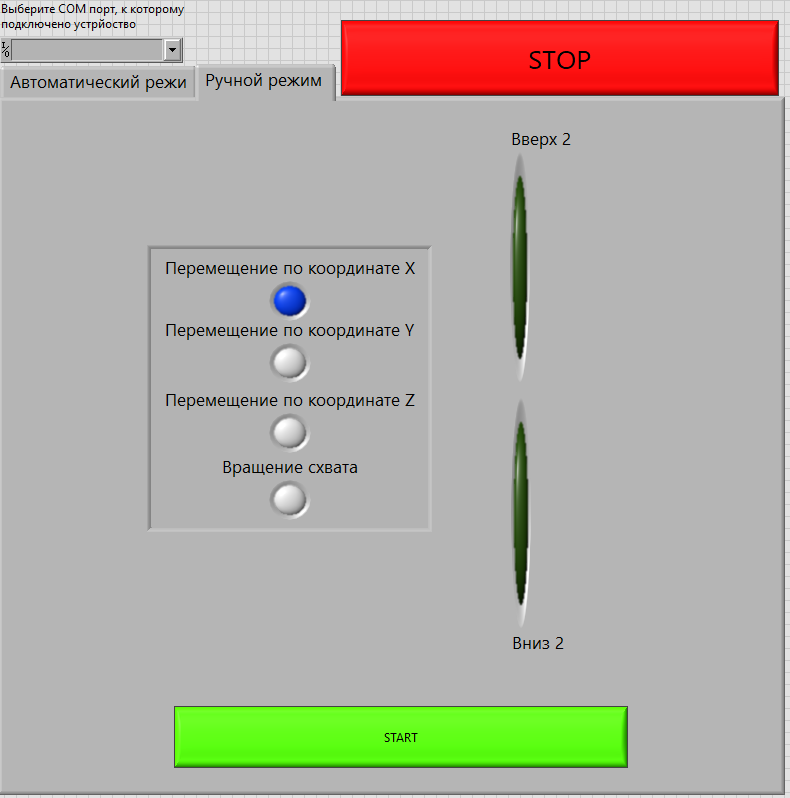


Рисунок 4.3.1 – Лицевая панель программы управления манипулятором

Сперва необходимо выбрать из выплывающего списка при нажати, *COM* порт, как показано на рисунке 4.3.2, к которому подключен манипулятор:

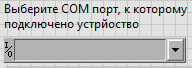


Рисунок 4.3.2 – Выбор *COM* порта

Далее, как показано на рисунке 4.3.3 необходимо выбрать какой режим вам нужен, ручной или автоматический.



Рисунок 4.3.2.3 – Выбор режима

Если выбран ручной режим, необходимо нажать кнопку старт. После чего можно выбирать координату, по которой должно происходить перемещение и нажатием кнопок вверх либо вниз управлять перемещение схвата.

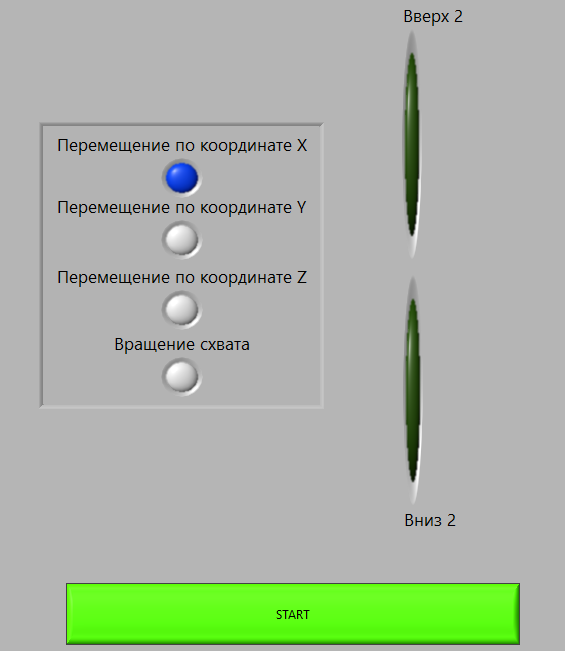


Рисунок 4.3.2.4 – Ручной режим

Если выбран автоматический режим, как показано на рисунке 4.3.2.5, то необходимо в окне исполняемый код ввести программу движения манипулятора в виде *G* кода и нажать кнопку старт. В ходе выполнения программы будет отображаться, какая по номеру сейчас выполняется строчка и выводиться сама эта строчка.

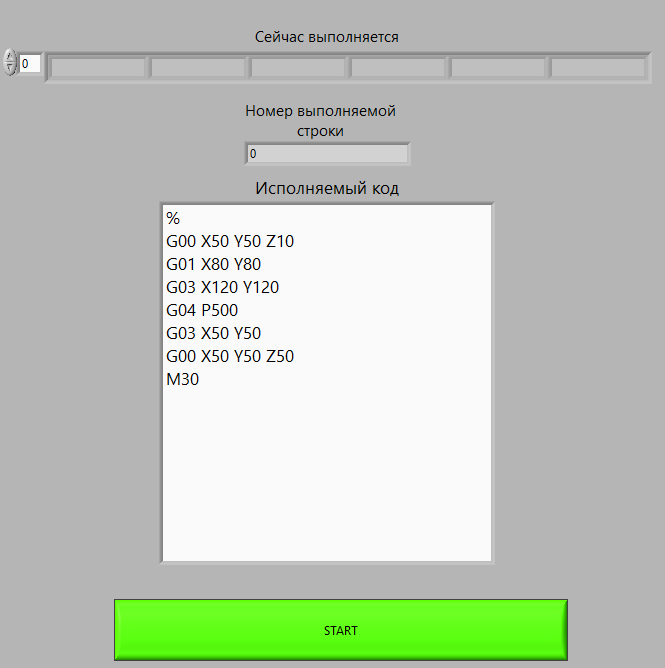


Рисунок 4.3.2.5 – Автоматический режим

В случае возникновения внештатной ситуации или при необходимости остановить любое движение манипулятора, нажать кнопу *Stop*.



Рисунок 4.3.2.6 – Кнопка Stop для аварийной остановки

Выводы по системе управления

В процессе работы над программным обеспечением мехатронного модуля были составлены алгоритмы работы манипулятора, программы для микроконтроллеров приводов и технологического контроллера. Далее была разработаны и описаны программы управления для микроконтроллеров приводов и технологического контроллера на языке программирования Си в среде *Atmel Studio*, собран проект, скомпилирована прошивка. Был представлен исходный код программ с подробными комментариями и описанием. Далее была разработано программное обеспечение для управления модулям на персональном компьютере с использование среды разработки *LabVIEW*, в которой была решена обратная задача кинематики и создан интерпретатор G кода.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы был спроектирован электромеханический манипулятор на базе шаговых двигателей. Указан состав системы и назначение этих элементов в установке. Заданы характеристики объекта автоматизации. Далее выбрали компоновку манипулятора и описали составляющие части манипулятора, а также принцип его работы на производстве. Также было выбрано захватное устройство, указаны его характеристики, проведен расчет захватного устройства. Рассчитана червячная и зубчатая передачу для привода поворота, передача винт-гайка качения для приводов вертикального и горизонтального перемещения и червячная передачу для привода поворота схвата. Выбрали двигатель для каждого привода. Разработаны сборочные чертежи для всех приводов и спецификации к ним. Разработан сборочный чертеж электромеханического манипулятора и спецификация к нему. Разработан чертеж общего вида манипулятора.

Выла проработана и описана архитектура системы управления, разработана ее функциональная схема, выбраны аппаратные средства сбора данных. Был разработан драйвер шагового двигателя, были выбраны основные компоненты, разработана схема электрическая принципиальная. Также разработан микропроцессорный блок, выбраны основные компоненты для него, разработана схема электрическая принципиальная. Были выбраны и разработаны интерфейсные преобразователи, разработана схема электрическая соединений мехатронного модуля.

В процессе работы над программным обеспечением мехатронного модуля были составлены алгоритмы работы манипулятора, программы для микроконтроллеров приводов и технологического контроллера. Далее была разработаны и описаны программы управления для микроконтроллеров приводов и технологического контроллера на языке программирования Си в среде Atmel Studio, собран проект, скомпилирована прошивка. Был представлен исходный код программ с подробными комментариями и описанием. Далее была разработано программное обеспечение для управления модулям на персональном компьютере с использование среды разработки *LabVIEW*, в которой была решена обратная задача кинематики и создан интерпретатор *G* кода.

# БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

1 Соломенцев, Ю.М.;. — Промышленные роботы в машиностроении: Альбом схем и чертежей. Учеб. пособие для вузов./ Ю.М. Соломенцев, К.П. Жуков, Ю.А. Павлов и др. — М.: Машиностроение, 1986. — 140 с.: ил.

2 Орлов, П.И. Основы конструирования: Справочно-методическое пособие. В 2-х кн. Кн. 1/Под ред. П.Н. Учаева. – Изд. 3-е, испр – М.: Машиностроение, 1988. – 560с.: ил.

3 Орлов, П.И. Основы конструирования: Справочно-методическое пособие. В 2-х кн. Кн. 2/Под ред. П.Н. Учаева. – Изд. 3-е, испр – М.: Машиностроение, 1988. – 544с.: ил.

4 Дунаев, П.Ф. Конструирование узлов и деталей машин: Учеб. пособие для студ. техн. спец. вузов/П.Ф. Дунаев, О.П. Леликов. – 8-е изд., перераб. и доп. – М.: Издательский центр «Академия», 2004. – 496 с.

5 Аввакумов, М.В. Расчет червячных передач: методические указания. /А. Б. Коновалов, М. В. Аввакумов; СПб ГТУРП. – СПб., 2012.-37с.

6 Ручкин, Л.В. Основы мехатроники и робототехники: методические указания. /Л.В. Ручкин, Л.П. Сысоева: СибГАУ им М.Ф. Решетнева, 2015.

7 Ручкин, Л.В. Методы моделирования мехатронных систем: учеб. пособие / Л.В. Ручкин, Н.Л. Ручкина; СибГУ им. М.Ф. Решетнева. – Красноярск, 2018. -80 с.

8 Анурьев, В.И. Справочник конструктора – машиностроителя. В 3-х томах. Т.1. – 6е изд., перераб и доп. / В. И. Анурьев– М.: Машиностроение, 1982.-736 с.:ил.