

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

Институт машиностроения
(наименования института полностью)

Кафедра «Промышленная электроника»
(наименование)

11.03.04 Электроника и нанoeлектроника
(код и наименование направления подготовки, специальности)

Промышленная электроника
(направленность (профиль) / специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)**

на тему 3D сканер

Студент

С.В. Волков

(И.О. Фамилия)

(личная подпись)

Руководитель

к.т.н., доцент, А.В. Прядилов

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

Консультант

к.п.н., доцент, А.В. Кириллова

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

Тольятти 2020

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

Институт машиностроения
(наименования института полностью)

Кафедра «Промышленная электроника»
(наименование)

ЗАДАНИЕ
на выполнение бакалаврской работы

Студент Волков Сергей Владиславович, Элб-1601а

1. Тема 3D сканер
2. Срок сдачи студентом законченной бакалаврской работы _____
3. Исходные данные к бакалаврской работе _____
Источник питания: сеть 220 В, 50Гц
Максимальный размер сканируемого объекта по вертикали: 10 см
Максимальный диаметр сканируемой детали: 10 см
Время сканирования при разрешении 0,5 мм: 1 час
Измеряемое расстояние до объекта: 2-15 см
4. Содержание бакалаврской работы
Аннотация
Введение
 1. Состояние вопроса
 - 1.1. Формулирование актуальности, цели и задач проекта
 - 1.2. Обзор существующих решений
 2. Схемотехническая часть
 - 2.1. Разработка структурной схемы
 - 2.2. Разработка электрической принципиальной схемы
 3. Разработка конструкции
 4. Программная часть
 5. Исследовательская частьЗаключение
Список используемой литературы
5. Ориентировочный перечень графического и иллюстративного материала
 1. Обзорный лист
 2. Структурная схема устройства
 3. Электрическая принципиальная схема
 4. Конструкция устройства
 5. Блок-схема алгоритма программы
 6. Результаты сканирования
6. Дата выдачи задания «28» февраля 2020 г.

Руководитель бакалаврской работы

(подпись)

А.В. Прядилов
(И.О. Фамилия)

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

Институт машиностроения
(наименование института полностью)

Кафедра «Промышленная электроника»
(наименование)

**КАЛЕНДАРНЫЙ ПЛАН
выполнения бакалаврской работы**

Студент _____ С.В. Волков _____
по теме _____ 3D сканер _____

Наименование раздела	Плановый срок выполнения раздела	Фактический срок выполнения раздела	Отметка о выполнении раздела	Подпись руководителя
Анализ актуальности проекта, написание введения и обзорной главы. Разработка структурной схемы разрабатываемого устройства.	28.02.2020	24.04.2020		
Разработка электрической принципиальной схемы. Написание остальных разделов работы.	25.04.2020	29.05.2020		
Оформление пояснительной записки и плакатов.	30.05.2020	3.07.2020		

Руководитель бакалаврской работы _____
(подпись)

А.В. Прядилов
(И.О. Фамилия)

Аннотация

Объем бакалаврской работы 86 стр., 53 рисунков, 14 таблиц, 20 источников, 3 приложений.

В данной бакалаврской работе был разработан действующий макет 3D сканера, способного сканировать миниатюрные объекты и выводить информацию об объекте формате трехмерных координат.

Цель работы: разработка действующего макета 3D сканера, способного определять пространственные координаты объекта и генерировать на их основе облако точек для воссоздания виртуальной модели реального объекта (детской игрушки, миниатюрной детали и т.п.).

Задачи работы:

1. Анализ исходных данных и известных решений
2. Выбор элементов схемы;
3. Разработка структурной и принципиальной схемы;
4. Разработка программы, управляющей логикой работы устройства;
5. Изготовление и отладка макета устройства;
6. Экспериментальные исследования.

Работа состоит из пяти глав, в которых решены упомянутые задачи.

Для оформления чертежей воспользовался программным пакетом КОМПАС-3D V16.

Для разработки программы, управляющей логикой работы, использовалась среда разработки Arduino IDE.

Для разводки платы воспользовался программным пакетом EAGLE.

Областью применения данной работы являются экспериментальные исследования физических процессов 3D сканирования в образовательных целях, а также получение 3D моделей объектов деталей для дальнейшего проектирования и печати на 3D принтере.

Abstract

The title of the graduation work is: «3D scanner».

The graduation work consists of an explanatory note on 86 pages introduction, five parts, a conclusion, including 53 figures, 14 tables, list of 20 references including 5 foreign sources, three appendices and the graphic part on 6 A1 sheets.

The key issue of this work is the development of a 3D scanner that can quickly get the necessary virtual model with the geometry of a real physical object, without initial construction. We touch upon the problem of creating 3D models of various objects, since the manual design of three-dimensional objects takes a large amount of mental and time effort, so there is a need to simplify the design process. A 3D scanner allows you to solve this problem.

The aim of the work is the development of a working 3D scanner layout and creation of an algorithm for scanning and obtaining information about an object in the format of three-dimensional coordinates.

This work is divided into the following chapters: status of the issue; circuitry part; design development; software part; research part.

We also report the results of experiments conducted to scanning of volume objects.

In conclusion we'd like to stress that this topic of this work is one of the relevant directions in the creation of three-dimensional models of objects of various technical directions. These technical developments can be applied for educational purposes, as well as a demonstration sample and for obtaining models for further engineering and printing on a 3D printer.

Содержание

Содержание.....	2
Введение.....	3
1. Состояние вопроса	5
1.1. Формулирование актуальности, цели и задач проекта	5
1.2. Обзор существующих решений.....	6
1.3. Определение концепции разрабатываемого устройства.....	32
2. Схемотехническая часть.....	37
2.1. Разработка структурной схемы.....	37
2.2. Разработка электрической принципиальной схемы	39
2.3. Разработка силовой части схемы.....	41
2.4. Разработка системы управления.....	47
2.5. Разработка печатного узла	54
3. Разработка конструкции	59
3.1. Разработка каркаса	60
3.2. Разработка корпуса	64
4. Программная часть.....	68
4.1. Разработка кода	68
4.2. Отладка кода под 3D сканер	79
5. Исследовательская часть	81
5.1. Обработка результатов сканирования.....	81
5.2. Вывод 3D модели	82
Заключение	84
Список используемой литературы	85
Приложение А Перечень элементов	87
Приложение Б Выходная характеристика датчика расстояния	88
Приложение В Программа 3D сканера	89

Введение

В результате развития технологий получения трехмерных объектов появляются всё больше устройств, позволяющих получать информацию о структуре и геометрии внешнего мира и объектов, из которых он образован. Одним из таких устройств выступает 3D-сканер, представляющий собой специальное устройство, которое позволяет получать информацию о форме предметов, путём анализа их физической формы в геометрическом представлении и последующей её оцифровкой.

В дальнейшем полученные данные об объекте в виде трехмерных координат X, Y, Z выводятся в виде облака точек, расположение которых воссоздает форму отсканированного объекта.

Цель 3D-сканера состоит в том, чтобы быстро получить необходимую виртуальную модель с геометрией реального физического объекта, без исходного построения, путём пошагового измерения расстояния до объекта с помощью сканирующего устройства (лазера, инфракрасного дальномера, камеры и т.п.) с последующим переводом полученных данных в формат координат X, Y, Z , сборка и воспроизведение которых выведет 3D модель отсканированной детали.

В настоящее время существует проблема создания 3D моделей различных объектов, так как на ручное проектирование трехмерных объектов уходит большое количество умственных и временных затрат и усилий, то возникает потребность в упрощении процесса проектирования. Решить данную проблему позволяет 3D сканер.

На рынке достаточно много современных 3D сканеров, обладающих оптимальными характеристиками для непосредственного сканирования, т.е. возможности которых определяют высокую скорость обработки и преобразования полученной информации в необходимую геометрию, что

позволяет сэкономить время, и при этом не потерять производительной способности.

В рамках данной работы предполагается разработать, создать и отладить небольшую недорогую действующую модель 3D сканера на базе микроконтроллера Arduino, способный сканировать небольшие объекты, которые окружают нас в быту, которые можно вывести в трёхмерное пространство и распечатать на 3D принтере. Возможности 3D сканера ограничены его аппаратными составляющими (инфракрасным дальномером, размером стола и высотой установки).

Такая модель может быть использована в качестве демонстрационного образца, показывая возможности 3D сканирования на базе микроконтроллера Arduino, а также для изучения физического процесса сканирования в образовательных целях.

1. Состояние вопроса

1.1. Формулирование актуальности, цели и задач проекта

Актуальность данной работы заключается во всё большем охвате заинтересованности вокруг 3D сканеров. Универсальность 3D сканеров проявляется в создании виртуальных объемных моделей объектов промышленности, киноиндустрии, науки, медицины и физических предметов, используемых в жизни. Как устройства, 3D сканеры становятся все более доступным оборудованием, что делает его продуктом для массового использования. Это сопровождается увеличением конкуренции между субъектами рынка, разработчиками 3D сканеров, что служит основным катализатором к развитию самой тематики 3D сканеров и делает её все более актуальной. Поэтому для построения конкурентоспособного решения, необходимо разбираться в современных 3D сканерах: понимать принцип работы, устройство, технологии сканирования, чтобы основательно подойти к эксплуатации и разработке собственных решений.

Цель работы - разработка действующего макета 3D сканера, способного определять пространственные координаты объекта и генерировать на их основе облако точек для воссоздания виртуальной модели реального объекта (детской игрушки, миниатюрной детали и т.д.).

Для достижения этой цели поставлены и выполнены следующие **задачи**:

1. Анализ исходных данных и известных решений
2. Выбор элементов схемы;
3. Разработка структурной и принципиальной схемы;
4. Разработка программы, управляющей логикой работы устройства;
5. Изготовление и отладка макета устройства;
6. Экспериментальные исследования.

1.2. Обзор существующих решений

1.2.1. Применение 3D сканеров

Применение трехмерных сканеров заключается в решении ряда технических задач, которые возникают в следующих областях:

- Промышленности;
- Науке;
- Медицине;
- Архитектура и дизайн;
- Киноиндустрии и видеоиграх;
- Метрологии и контроле измерений.

Применение 3D сканеров основано на решении задач, когда необходимо за короткое время получить трехмерную модель реального объекта с высокой точностью. Здесь проявляется универсальность их применения в распознавании сложных геометрических форм. Благодаря этой способности значительно упрощается дальнейшее ручное проектирование.

Самое широкое применение 3D сканеров обрзается на предприятиях различного производственного назначения (машиностроения, авиастроения и т.п.), где преимущественно ставятся задачи обратного инжиниринга (реверс инжиниринга), когда необходимо отмасштабировать или изменить геометрию изделия, а конструкторская документация отсутствует. Таким образом, реальный объект сканируется, и на основе данных сканирования получается CAD-модель, т.е. происходит перевод материального объекта в цифровую форму.

Тем самым осуществляются задачи оцифровки физического объекта макета, для создания на его основе серийного изделия. Так, например, уникальные по своей геометрии поверхности автомобильных частей и изделий (рисунок 1), могут быть измерены, проанализированы и оцифрованы для последующего массового производства с использованием станков с ЧПУ.



Рисунок 1 - Сканирование деталей автомобиля

Как правило, отсканированные 3D данные для проектирования подобных изделий в серию не распространяется свободно, так как являются объектом интеллектуальной собственности владеющей ею компании, и могут передаваться только в коммерческих целях при сотрудничестве с другими компаниями.

В медицине 3D сканер выступает средством для сканирования воспроизведения моделей человеческих органов в образовательных целях, а также на практике (например, измерить зубы в челюсти человека). С его помощью можно воссоздавать ортопедические скобы, браслеты и анатомическую обувь для последующего проектирования в CAD системах.

Дизайнеры в архитектуре используют трехмерные сканеры для получения формы объекта, и возможности её доработки впоследствии. Зачастую, здесь возникают ситуации, когда необходимо произвести немало замеров по объекту (например, по макету реального здания), поэтому обычно обводят их по контуру, вращая на специальном столе. В конечном счёте, аппарат преобразует физический объект в цифровую модель на компьютере. После чего открывается возможность для их печати на 3D принтере или обработке на станках с ЧПУ.

В индустрии развлечений и игр применение 3D сканеров находит в разработке цифровых моделей игровых персонажей, моделей реальных актёров, а также объектов окружения и его декораций с последующим объединением в общую концепцию.

1.2.2. Основная концепция

Работа над изготовлением детали всегда начинается с 3D-модели, причём чтобы получить из заготовки или напечатать нужную деталь возникают следующие пути решения: разработать 3D модель в программе трехмерного моделирования или воспользоваться возможностями 3D-сканирования и получить модель уже существующей вещи, чтобы затем отредактировать ее и использовать. Очевидно, что по своей сути второй вариант будет более быстрым и возможно более качественным.

Например, при ручном замере геометрически сложной детали трудно добиться необходимой точности, и в дальнейшем передать все нелинейности её формы при проектировании математической модели самой детали. Вот здесь и возникает потребность в 3D сканере, который за короткий срок способен получить готовую математическую модель эквивалентную эталонной.

3D-сканер вычисляет расстояние до объекта, используя камеры с конусообразным полем зрения. Тем самым, 3D сканеру приходится дистанцироваться информацией о поверхностях в пределах его поля зрения, чтобы избежать потери информации. Также в дополнение к камерам используется подсветка (осветители), для повышения точности измерений.

Воспроизводимая с помощью 3D сканера картинка описывает расстояние до поверхности в каждой точке на изображении. Это позволяет получать выходные данные координаты X,Y,Z. Объем и качество данных зависят от применяемого 3D-сканера.

В конечном счёте, выдается трехмерное положение каждой точки на изображении. Тем самым визуализируется сосредоточенный набор точек по геометрии, который представляет из себя облако точек (рисунок 2).

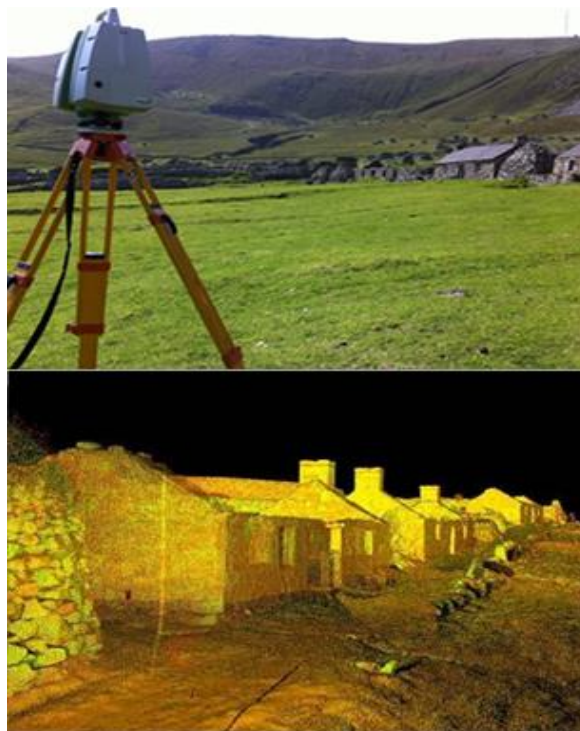


Рисунок 2 - Облака точек, получаемое в результате 3D-сканирования объектов

В результате сканирования все данные измерений в виде облака точек анализируются и выводятся на экран в виде трехмерного изображения с помощью компьютера по специальному программному обеспечению идущему к сканеру. Благодаря нему можно управлять процессом сканирования, изменять данные: выбирать разрешение и необходимые области для уточнения детализации, регистрации данных, а затем объединять их, чтобы создать полноценную 3D модель.

Таким образом, основным преимуществом 3D сканирования является сбор и получение данных об объемном объекте за короткое время и с хорошим качеством проработки деталей, особенно если сканер оборудован высококачественными сканирующими модулями с высоким разрешением и скоростью анализа.

1.2.3. Состав 3D сканера

Типовой состав 3D сканера приведен на рисунке 3.



Рисунок 3 – 3D сканер в полной комплектации

Основной состав комплектации 3D сканера (рисунок 3) строится из:

- Корпуса 1, определяемый функциональным назначением сканера;
- Несущей части (фундамента) 2. И подвижной части (вращающейся площадки), при вращении на которой объект распознаётся и анализируется во всём объеме и преобразуется в трехмерное пространство. Подвижная часть приводится в действие, благодаря шаговому двигателю;
- Дополнительно идущих треноги и штатива 3;
- Сканирующих сенсоров 4 (например, камеры, лазерных датчиков, инфракрасных сенсоров, а также контактных датчиков), с помощью которых происходит измерение расстояния до объекта и последующее преобразование измерений в трехмерное представление;
- Специальных осветителей 5, которые размещают рядом со сканирующими сенсорами для поддержания равномерного освещения во время сканирования, а также в некоторых случаях проецируют световые сетки непосредственно на объект (с помощью проектора), а камеры анализируют деформации на объекте и получают картинки;
- Базовых меток 6 для калибровки 3D сканера и привязки к исследуемому объекту во время сканирования.

В аппаратную часть сканера входит контроллер управления устройством, память, используемая для записи и хранения получаемой информации во время сканирования, а также блока питания под питание системы и USB переходника с 3D сканера для связи с компьютером.

К аппаратной части приходится специализированное программное обеспечение (ПО) с драйверами для самого устройства. С помощью него 3D сканер взаимодействует с компьютером (ноутбуком), передавая и воспроизводя данные (облако точек) на мониторе компьютера.

Определение форм фактора и метода сканирования 3D сканера выступают ведущими аспектами при построении конструкции и электрической составляющей 3D сканера. Определение необходимого качества сканирования зависит от следующих характеристик (таблица 1).

Таблица 1 - Характеристические особенности

Характеристика	Описание	Зависимость
Разрешение (разрешающая способность)	Наименьшее расстояние между сканируемыми точками в мм.	Обуславливается размерами сканируемой области, характеризующей разрешающей способностью и принципом работы сканера.
Точность	Суммарная ошибка измерения.	Включает в себя как случайную, так и систематические составляющие.
Шум	Случайная суммарная ошибка измерения. Устанавливается при повторных сканированиях путём сравнения с первым результатом.	Зависит от качества поверхности (гладкость/шероховатость, неоднородности) сканируемого объекта, расположения камер (относительно объекта и друг друга), освещения объекта и т.д.
Детализация	Размер объекта в мм, который может распознать сканер. Стандартное значение для 3D сканера: 1 мм.	Показатели детализации в зависимости от типа сканера могут варьироваться. В сравнение со стационарными, ручные сканеры имеют меньшие значения детализации.

Ключевой характеристикой из таблицы 1 выступает точность, которая показывает уровень соответствия полученной модели реальному образцу. Точность, как параметр, определяет погрешность, а также диапазон, в пределах которого могут сосредотачиваться результаты сравнения.

Для обеспечения идеальной точности требуется дорогостоящее оборудование. Однако не для всех изделий требуются высокие значения точности. В этом случае выгоднее использовать более бюджетные решения.

В общем случае при сканировании большинства других изделий достаточно точности в 0,1% от измеряемой длины. В большинстве задач, например, при сканировании скульптур и ювелирных изделий предъявляются элементарные требования к точности, поэтому в первую очередь следует снизить шум, улучшить детализацию, максимально проработать поверхность объекта. Примеры задач и требования по точности приведены в таблице 2.

Таблица 2 - Примеры задач и требования по точности

Пример задачи	Требования
Сканирование статуэтки, макета здания, скульптуры и т.п. Уменьшенные копии объектов предполагается изготавливать с помощью станков с ЧПУ или печатать на 3D принтере.	В этом случае необходимости в высокой точности нет. Большое значение имеет получение визуально идентичного результата, т.е. требуется высокая детализация, минимальный уровень шума. Конечный результат - получение STL файла, пригодного для дальнейшей работы с изготовителем.
Сканирование велосипедной вилки, педали, рамы. Результаты сканирования станут основой для моделирования другой вилки, педали, рамы.	Особую важность представляют места креплений, посадки и линии присоединения. В указанных местах необходима точность порядка 0.5 мм.
Сканирование стеклянной бутылки. На основании полученного результата будет создана новая форма.	Требования по точности минимальны. Форма горлышка и элементы резьбы берутся из CAD модели. Так как форма гладкая, значения уровня шума и детализации не значимы.
Сканирование фамильного перстня, серёжек, подвески.	Результат должен быть с высоким разрешением и детализацией, низким уровнем шума. Значения точности не важны.

Таким образом, увеличение качества сканирующей системы возникает чаще всего из-за необходимости в сканировании объекта, приоритетом в котором непосредственно ставиться качество высокой степени детализации, а где необходимости в этом нет, устанавливают более экономичные решения.

Так, в рамках данной ВКР не возникает необходимости в достижении высокой детализации, так как 3D сканирование предполагается использовать в образовательных целях.

1.2.4. Классификация 3D сканеров

В настоящее время 3D-сканеры делятся на два основных вида:

1. Контактный. Такой вид 3D сканера работает следующим образом: объект крепится к поверочной плите, устройство тактильно снимает показания (например, обводом модели специальным высокочувствительным щупом), после чего результаты показаний переносятся в трехмерное пространство в виде координат сканируемой модели (рисунок 4). Такие сканеры проводят излучение объекта напрямую.



Рисунок 4 - Пример сканирования щупом на контактном 3D сканере

В целом контактное сканирование не зависит от условий освещения.

По ходу эксплуатации и результату выдаваемым контактными 3D сканером можно подвести достоинства и недостатки этого типа:

Достоинства:

- Высокая детализация;
- Независимость от световых условий (освещения);
- Возможность сканирования призматической части объекта;
- Ценовая доступность.

Недостатки:

- Медленный процесс сканирования и анализа в целом (за один замер производится оцифровка только одной точки);
- Ограничение сканирования подвижных объектов, а также объектов имеющих сложную геометрическую форму.

2. Бесконтактный. Такой вид 3D сканера оказывает излучение или отражения светового потока (при помощи ультразвука и рентгеновских лучей), посредством которого предмет сканируется, т.е. подвергается цифровому исследованию.

Бесконтактные сканеры делятся на 2 вида: активные и пассивные.

а) Активные сканеры излучают на объект направленные волны в виде лазерного луча или структурированного света и обнаруживают его отражение, после чего анализируют (рисунок 5). Обычно здесь используется светодиодный или лазерный луч, реже - рентгеновские лучи, инфракрасное излучение или ультразвук.



Рисунок 5 - Пример бесконтактного активного 3D сканера

б) Пассивные сканеры обнаруживают отраженное окружающее излучение. Обычно обнаруживается видимый свет как естественное излучение окружающей среды (рисунок 6).



Рисунок 6 - Пример бесконтактного пассивного 3D сканера

Ключевыми компонентами 3D-сканеров пассивного типа выступают камеры, которые сводят материал в единую объемную фигуру по специальному программному обеспечению.

Классификация бесконтактных сканнеров проводится по следующим критериям:

- Ручной или стационарный (имеет платформу, на которой находится сканируемый объект или требует обхода вокруг объекта);
- Метод измерения (триангуляционный, времяпролетный, фазовый);
- Размер объекта сканирования;
- Скорость сканирования.
- Точность измерения;
- Разрешение сканирующих модулей (камеры, ИК-датчика, лазера).

Таким образом, контактные сканеры обладают преимуществом в качестве получаемых объектов, однако, не обладают способностью их цветопередачи. Их использование оказывается куда более трудоёмким, по сравнению с бесконтактными сканерами. Обладая хорошими скоростями сканирования, они позволяют получить необходимую модель за короткий срок с сохранением цвета объекта сканирования.

1.2.5. Технологии и методы 3D-сканирования

Исходя из п.1.2.4. решил раскрыть технологии, применяемые в 3D сканерах. В настоящем существует много различных 3D сканеров, которые работают по разным технологиям сканирования.

Выделяют следующие технологии трехмерного сканирования:

Для контактных 3D сканеров:

1. Контактная. Такая технология сканирования, или контроль геометрических размеров производится посредством подвода измерительного щупа к измеряемому объекту, регистрируя малейшие касания. Процесс такого сканирования самый медленный и самый точный.

К примеру, по такой технологии работает 3D сканер, применяющийся для контроля пресс форм - Faro Arm Edge 9 (рисунок 7), представляющий из себя 7-ми осевую измерительную руку, оборудованную магнитным основанием, которое позволяет надежно закрепить сканер на любой ровной металлической поверхности.



Рисунок 7 - 3D сканер Faro Arm Edge 9

Данные сканеры достаточно широко применяются на высокоточных производствах для контроля геометрических размеров выпускаемой продукции.

Для бесконтактных активных 3D сканеров:

1. Лазерная. Это технология бесконтактного неразрушающего контроля, которая в цифровом формате фиксирует форму физических объектов с помощью лазерного луча. Лазерные 3D-сканеры измеряют мельчайшие детали и фиксируют произвольные формы для быстрого создания высокоточных облаков точек.

Лазерная технология включает в себе следующие методы сканирования:

а) Триангуляционный метод измерения расстояний

В данном методе происходит измерение расстояния с использованием лазерного модуля и оптического сенсора, которые находятся на некотором расстоянии и под углом α относительно друг от друга. Схематичная иллюстрация измерения расстояния приводиться на рисунке 8.

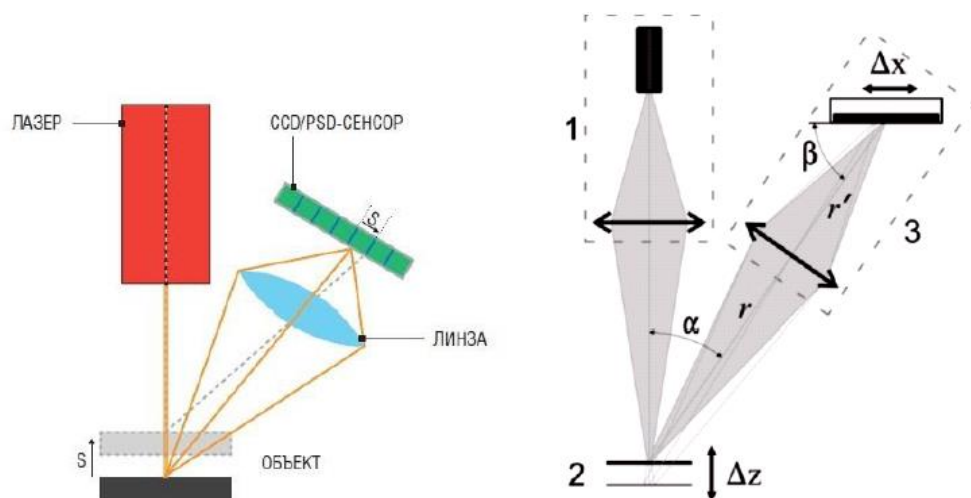


Рисунок 8 - Схема измерения расстояния триангуляционным методом.

По схеме измерения расстояния (рисунок 8) на поверхность объекта подаётся лазерный луч, поступающий с лазера. Затем он отражается от объекта и считывается сенсором, попадая в его поле зрения. После чего измеряется расстояния до объекта. Тем самым в результате одного измерения образуется треугольник в соотношении: «лазер – объект – сенсор».

Следует отметить, в зависимости от изменения расстояния до объекта также меняется местоположение лазерного следа на нём, что играет ключевую роль в формировании конечного результата сканирования.

Существует математическая зависимость между расстоянием на матрице и расстоянием до объекта, определяемая по формуле 1.

$$\Delta z = \frac{r \cdot \sin \varphi}{\sin(\alpha - \varphi)} \quad (1)$$

где r - расстояние до объекта;

α – фиксированный угол между лазером и камерой;

φ - фазовый сдвиг.

По формуле 1 рассчитывается разность расстояний, которая является искомым калибровочным расстоянием между платформой с закреплённым объектом сканирования, и лазером, испускающим на него луч.

Потенциал метода триангуляции заключается в измерении расстояния не только для одной точки, а до целой группы вертикальных точек. Это достигается при использовании специального лазерного модуля, фокус которого сосредоточен не в одной точке, а выводится целой вертикальной лазерной линией (рисунок 9).

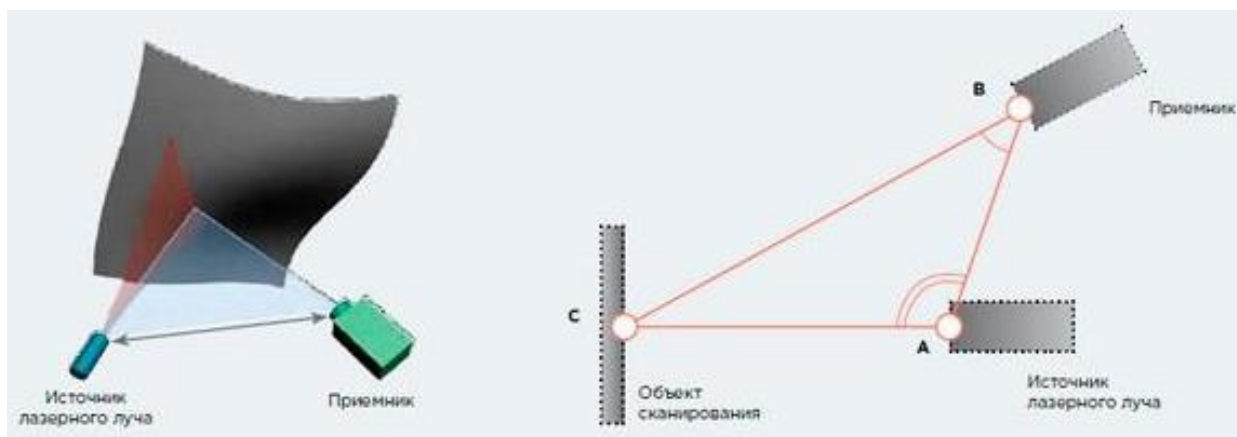


Рисунок 9 - Схема измерения лазерной линией расстояния до объекта

По схеме (рисунок 9) видно, что при измерении расстояния образуется треугольник с вершинами А (источник лазерного излучения), В (приёмник), С (объект сканирования), расположенные под углами, при изменении которых меняется расположение лазерного фокуса на объекте.

Несмотря на то, что 3D сканеры триангуляционного типа обладают лучшей точностью сканирования, они не способны распознавать дальние объекты, в отличие от сканеров, основанных на измерении времени отклика луча.

По такой технологии работают 3D сканеры, используемые для сканирования миниатюрных объектов, фигурок, детских игрушек, для последующей печати на 3D принтере: BQ Ciclop, Digitizer (MakerBot), которые приведены на рисунке 10.



а)



б)

Рисунок 10 – 3D сканеры, работающие по методу лазерной триангуляции:

а) BQ Ciclop; б) Digitizer (MakerBot)

По этой технологии работают также портативные ручные версии сканеров, обладающие лучшей скоростью сканирования.

Один из самых популярных ручных 3D сканеров, работающих по такой технологии: Shining 3D FreeScan X7 (рисунок 11).



Рисунок 11 - Ручной 3D сканер FreeScan X7, работающий по методу лазерной триангуляции

б) Времяпролетный метод измерения расстояний

Данный метод измерения расстояния заключается в измерении времени отклика лазерного луча от поверхности объекта, посредством испускания луча из излучателя, который в результате отражения от измеряемого объекта фиксируется приемником. В данном методе проверяется состояние приемника, путём создания высокой частоты обрабатывающего устройства. На рисунке 12 приводятся схематичное представление измерения расстояния.



Рисунок 12 - Схема времяпролетного метода измерения расстояния

Для данной схемы характерна зависимость расстояния от времени (таблица 3).

Таблица 3 – Зависимость расстояния от времени в численном виде

Расстояние до цели L	1 м	10 м	100 м	1 км	10 км	100 км
Время отклика t	6,7 нс	67 нс	0,67 мкс	6,7 мкс	67 мкс	0,67 мс

Искомое же расстояние до объекта L находится следующим образом (формула 2).

$$L = \frac{c \cdot t}{2 \cdot n}, \quad (2)$$

где c – скорость света,

t – время от момента испускания луча на объект до момента отражения от него и считыванием приемником,

n - показатель преломления среды.

Исходя из этого скорость распространения света в вакууме отличная от скорости распространения фотона света в конкретной среде.

Метод сканирования считается достаточно трудным, т.к. основная сложность вызывается с подсчетом времени отклика лазерного луча на малых расстояниях (менее метра).

С такой технологией работают 3D сканеры FARO Focus 3D и Leica HDS8800, которые приведены на рисунке 13.

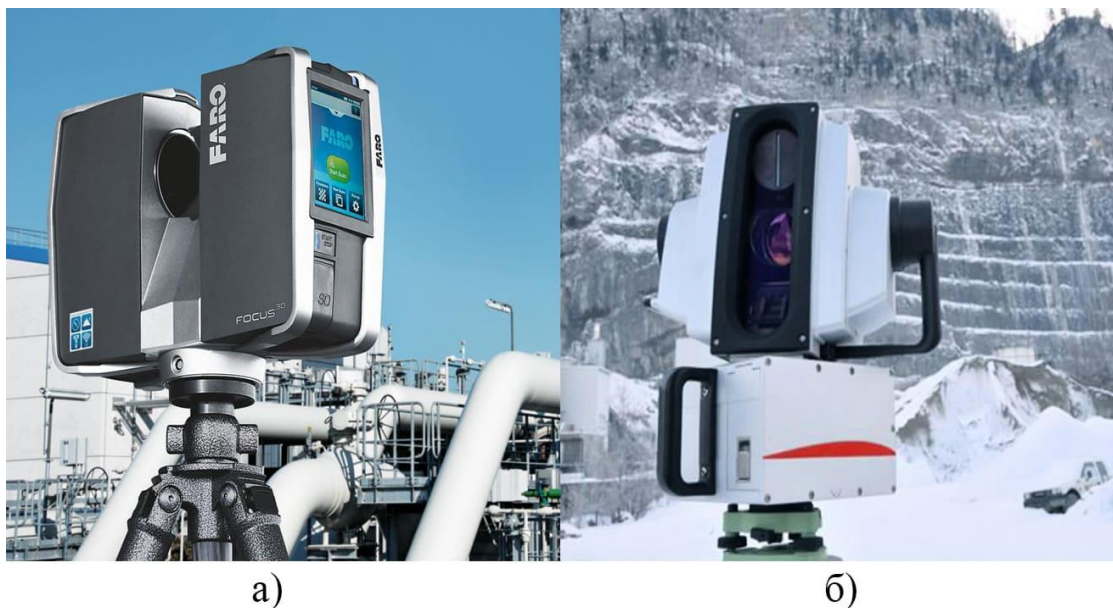


Рисунок 13 – 3D сканеры, работающие по времяпролетному методу:

а) Faro Focus 3D; б) Leica HDS8800

Эти сканирующие устройства идеально подходят для съемок местности и больших объектов и не предназначены для сканирования мелких деталей.

в) Фазовый метод измерения расстояний

Принцип действия данного метода измерения расстояния основан на измерении разности фаз прошедшего в приемник отражённого света от объекта. Многократно измеряется разность фаз для разных модулирующих частот. Затем по полученным значениям измерений решается уравнение для нахождения расстояния до объекта (формула 3):

$$D = \frac{c}{2f} \cdot \frac{\varphi}{2\pi}, \quad (3)$$

где φ – фазовый сдвиг,

c – скорость света,

f – модулирующая частота.

Формулой 3 можно пользоваться только в случае, если расстояние до объекта меньше половины длины волны модулирующего сигнала - $c/2f$. Иначе невозможно посчитать количество периодов волны, число которых укладывается в измеряемое расстояние. В этом случае, у лазера переключают частоту модуляции, после чего процесс измерения возобновляется.

Так как компаратор измеряет разность фаз, а разность в вольтах соответствует разности в амплитудах и в фазе, то в уравнение синусоиды можно поставить текущее значение амплитуды, максимальное значение и найти фазу.

Схема измерения расстояния фазовым методом, приведенная на рисунке 14, отражает физический смысл возникновения разности фаз.

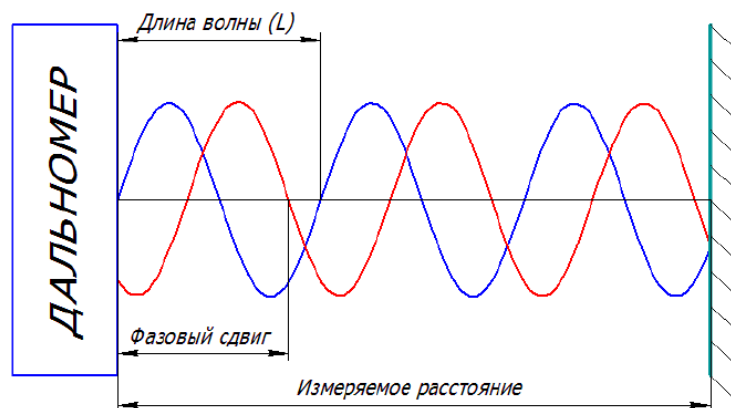


Рисунок 14 – Схема измерения расстояния фазовым методом

По данной схеме (рисунок 14) определяется искомое расстояние до объекта измерения. По длине волны, испускаемой дальномером, определяется сама точность метода измерения, после чего формируется окончательный результат измерения.

На фазовом методе сканирования основана работа 3D сканера Z+F IMAGER 5006 (рисунок 15).



Рисунок 15 – 3D сканер Z+F IMAGER 5006, работающий по фазовому методу

2. Оптическая технология. Такая технология позволяет достигать большой скорости сканирования, а также считывать объекты при движении.

Оптическая технология включает в себе следующий метод сканирования:

а) Метод сканирования структурированным светом. При таком методе проводится проекция световой сетки в виде «зебры», либо «черно-белых квадратов» на объект. В этот момент камеры начинают фотографировать объект и анализировать деформации на нём. На основе деформаций происходит генерация 3D модели сканируемого объекта. Технически, сетка проецируется на объект с помощью жидкокристаллического проектора, а камера, расположенная чуть в стороне от проектора, фиксирует форму самой сетки и вычисляет расстояние до каждой точки в поле зрения.

Недостатком данной технологии является ограничение по возможности сканирования крупных объектов. Недостаток можно устранить, если разметить объект специальными базовыми точками, которые позволяют сканировать большие объекты по частям с последующей «склейкой» модели.

По такой технологии работают 3D сканеры:

RangeVision Smart; RangeVision PRO; David SLS-3 (рисунок 16).



а)



в)



б)

Рисунок 16 – Стационарные 3D сканеры, работающие по методу структурированного света:

а)RangeVision Smart; б)RangeVision PRO; в)David SLS-3

По такому методу, также работают портативные ручные версии сканеров, обладающей большой точностью и скоростью сканирования.

Так, по такой технологии работают следующие модели 3D сканеров:

Artec Space Spider; Artec Eva, Creaform GO!SCAN 50, Shining 3D EinScan Pro 2X (рисунок 17).



Рисунок 17 – Ручные 3D сканеры, работающие по методу структурированного света: а) Artec Space Spider; б) Artec Eva; в) Creaform GO!SCAN 50; г) Shining 3D EinScan Pro 2X

Для бесконтактных пассивных 3D сканеров:

а) Стереоскопический метод сканирования. В сканерах, которые основаны на стереоскопическом методе сканирования, имеются две камеры, повернутые под не большим углом относительно друг друга. Анализируя разницу между двумя изображениями, строится трехмерная модель. Точность таких сканеров не высока, но зато позволяет получить цветную трехмерную модель.

По разнице в снимках с каждой камеры можно определить расстояние до каждой точки на изображении. Этот метод по своему принципу похож на стереоскопическое зрение человека.

С такой технологией измерения работают 3D сканеры Einscan-S от компании Shinning 3D: EinScan-S, EinScan-SE (рисунок 18).



Рисунок 18 - 3D сканеры Shinning 3D: а)EinScan-S; б)EinScan-SE, работающие по стереоскопическому методу сканирования

б) Фотометрический метод сканирования. Здесь используется фотограмметрическая съемка, а также в дополнении специальная система меток, позволяющая программе с большой точностью определить с какого ракурса и какая часть объекта была сфотографирована и как следствие - сделать более точную модель.

Наиболее качественное решение сейчас имеет 3D сканер AICON DPA от компании AICON (рисунок 19).



Рисунок 19 – 3D сканер AICON DPA, работающий по фотометрическому методу

Зачастую такая технология используется при сканировании больших объектов, а также выступает дополнительным атрибутом для повышения точности сканирования.

в) Метод силуэта. Заключается в производстве серии снимков и использовании на контрастном фоне контуров из полученных фотографий самого трехмерного объекта. Эти силуэты преобразуют для получения видимой оболочки объекта. При данном методе невозможно сканирование углубления в объектах (к примеру, внутреннюю полость кувшина).

Так же существуют другие технологии сканирования, например компьютерная томограмма (КТ) и МРТ, использующие рентгеновское излучение, а так же коноскопическая голография. Все эти методы сканирования довольно узкоспециализированные, и не имеют смысловой принадлежности к сути ВКР.

Таким образом, в результате рассмотрения существующих технологий и методов 3D-сканирования были раскрыты физические и технические особенности, которые будут учитываться при общем анализе и формировании собственного решения.

1.2.6. Анализ исходных данных и известных решений на рынке

Исходя из п.1.2.3-п.1.2.5 вышеприведенной классификации, методов и технологий 3D сканеров следует учитывать конструктивные и типовые особенности существующих 3D сканеров для определения рационального решения для последующей разработки. В начале для анализа этих особенностей произвёл поиск известных решений на рынке и охарактеризовал каждую из моделей по параметрам, составив сравнительную аналитику (таблица 4).

Таблица 4 – Сравнительная аналитика известных решений 3D сканеров на рынке

Виды сканеров	Технология сканирования	Метод сканирования	Характер применения	Назначение	Модели 3D сканеров	Цена, руб
Контактные	Контактная	Контактный	Стационарный	- Контроль геометрических размеров выпускаемой продукции	Faro Arm Edge 9	655 000
					ROMER Absolute Arm SE 7	1 880 000
Бесконтактные (активные)	Лазерная	Лазерная триангуляция	Стационарный	- Сканирование малогабаритных объектов, например статуэток или игрушек. - Использование для печати на 3D принтере или получения 3D модели для компьютерных играх.	BQ Ciclop,	24 890
					Digitizer (MakerBot)	93 100
			Ручной		Shining 3D FreeScan X7	3 250 000
					ScanTech KSCAN20	1 800 000
		Времяпролетный	Стационарный	- Съемки местности и больших объектов. - Ландшафтный дизайн, - Геодезические измерения, - Построение карт местности, - Сканирование памятников культуры.	FARO Focus 3D,	4 800 000
					Leica HDS8800	3 500 000
					Z+F IMAGER 5006	2 050 000
		Фазовый				
	Оптическая	Структурированным	Стационарный	- Реверс-инжиниринг,	RangeVision Smart	175 000

Продолжение таблицы 4

		светом		- Образование, - Компьютерные игры, -Протезирование.	RangeVision PRO	1 040 000
					David SLS-3	300 000
			Ручной	- Сканирование людей, - Архитектура, - Протезирование, - Музейное дело.	Artec Space Spider	1 740 000
					Artec Eva	1 210 000
					Creaform GO!SCAN 50	400 000
					Shining 3D EinScan Pro 2X	420 000
Бесконт актные (пассивн ые)	Бесконт актная	Стереоскоп ический	Стациона рный	- Не детализированны й реверс- инжиниринг	Shinning 3D EinScan-S	80 000
		Фотометри ческий		- Сканирование больших объектов, - Дополнительный атрибут для повышения точности сканирования.	AICON DPA	120 000
		Метод силуэта		- Сканирование не больших фигурок	-	-

По таблице 4 можно убедиться на сколько широкий выбор самых разных моделей по своим характерным особенностям и применению имеется на рынке. Далее из таблицы 4 выделил сканеры, которые наиболее удовлетворяют цели и задачам моего проекта. Здесь упор делался преимущественно на недорогие бесконтактные настольные решения, проводящие сканирование миниатюрных объектов. Данные 3D сканеры сгруппировал и занёс в таблицу 5.

Таблица 5 – Наиболее подходящие решения 3D сканеров на рынке

Наименование	Технология сканирования	Метод сканирования	Описание	Цена, руб.
BQ Ciclop	Активная (Лазерная)	Лазерная триангуляция	<ul style="list-style-type: none"> - Считывающее устройство - Logitech C270 HD веб камера; -2 лазера с апертурой и регулировочными винтами; -USB кабель; -Шаговый двигатель Nema с вращающей платформой; -Платы питания/управления — ZUM SCAN Shield/ ZUM BT-328; -Точность – 0.5-5мм от размеров детали; -Область сканирования: 205 мм; -ПО – Horus. 	24 890
Shinning 3D EinScan-S	Пассивная	Стереоскопический	<ul style="list-style-type: none"> - Сканирующий модуль; - Поворотный стол; - Кабель USB; - Прозрачная подложка с областями для сканера и стола; - Калибровочная панель; - Точность – 0,1 мм; - Специальное ПО Shining. 	80 000
Digitizer (MakerBot)	Активная (Лазерная)	Лазерная триангуляция	<ul style="list-style-type: none"> - Поворотная платформа, - Камера; - 2 лазера с апертурой и регулировочными винтами; - Кабель USB; - Блок питания; - Точность – 0,2 мм; - Специальное ПО Digitizer. 	93 100

С целью определения необходимых качеств, которыми должно обладать разрабатываемое устройство для решения поставленных задач, выявил основные достоинства и недостатки сканеров, которые свёл в таблицу 6.

Таблица 6 – Достоинства и недостатки 3D сканеров

Модель 3D сканера	Достоинства	Недостатки
BQ Ciclop	<ul style="list-style-type: none"> - Сканирование 2-мя лазерными лучами отличается оптимальными показателями передачи геометрии и структуры небольших и средних объектов - Автокалибровка камеры занимает несколько секунд; - Высокая точность и скорость сканирования; 	<ul style="list-style-type: none"> - Калибровка 3D сканера при сдвиге лазеров; - Размеры не могут быть меньше 200x200 мм и больше 500x500 мм, т.к. сканер считается настольным, и по этому, может сканировать объекты статичные, помещающиеся в область сканирования. - Нет возможности сканировать предметы с светящимися, блестящими, очень темными, с размытыми контурами (мягкие игрушки и пр.). А также не могут они быть прозрачными и двигающимися.
Shinning 3D EinScan-S	<ul style="list-style-type: none"> - Точность сканирования; - Поворотный стол, вращающей объект на 360 градусов; - Возможность сканировать как маленькие объекты, так и достаточно крупные; - 2 режима сканирования (автоматическое и свободное); - Отличная передача цвета в режиме текстурного сканирования. 	<ul style="list-style-type: none"> - Калибровка 3D сканера при очередном случайном или намеренном сдвиге сканирующего модуля; - Объекты сканирования должны иметь фиксированный размер, т.к. сканер считается настольным
Digitizer (MakerBot)	<ul style="list-style-type: none"> - Мобильность; - Расширенный функционал; - Точное сшивание полученных сканов; 	<ul style="list-style-type: none"> - Сложности при работе с блестящими, прозрачными и темными поверхностями; - Не может применяться для сканирования крупногабаритных предметов. - Отсутствует цветное сканирование.

Следует также отметить, что каждый из рассмотренных 3D сканеров, оборудуется собственным аппаратно-программным комплексом, который работает со специализированным программным обеспечением, которое совместимо только с моделями самой фирмы. Исходя из этого, с целью упрощения реализации управления устройством было принято решение

воспользоваться модифицируемым аппаратно-программным комплексом Arduino, под который можно написать и отладить код управления логикой работы устройства в специальной среде программирования Arduino IDE.

Таким образом, рассмотрев представленные решения, а также достоинства и недостатки, которые лежат в их основе, было решено разработать бесконтактный действующий макет 3D сканера, имеющий конструкцию с подвижным столом, приводимым в движение шаговым двигателем. В качестве датчика расстояния решил использовать инфракрасный датчик расстояния с триангуляционным методом измерения расстояния, который более дешевый и доступный среди своих аналогов, способных измерять расстояние от 2 до 15 см. Управление системой организуется, благодаря аппаратной платформе Arduino, как более простая и доступная в программировании и отладке микроконтроллерная система управления.

1.3. Определение концепции разрабатываемого устройства

1.3.1. Определение принципа работы 3D сканера

Физический принцип работы 3D сканера заключается в цикле совокупной работы всех функциональных компонентов системы, которые взаимодействуют между друг другом при получении сигналов управления с микроконтроллера Arduino.

Сканирование происходит, начиная от фиксированного положения сканирующей платформы с инфракрасным дальномером, когда шаговый двигатель начинает вращать стол с предметом (рисунок 20).

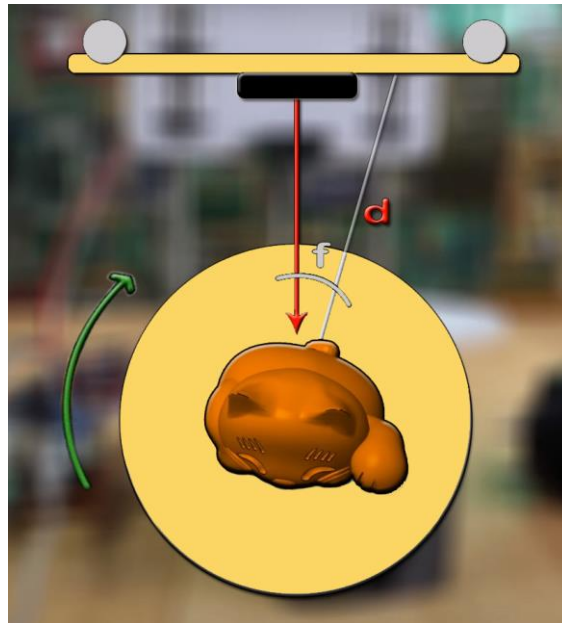


Рисунок 20 – Процесс вращения стола с объектом сканирования

В этот же момент времени задействуется инфракрасный дальномер, который производит измерение расстояния d до объекта параллельно вращению стола. Совершая один оборот, шаговый двигатель вращает стол на 360 градусов с определенным угловым шагом $f = 1.8$ градусов, после чего другой шаговый двигатель, отвечающий за положение сканирующей платформы с инфракрасным дальномером, передает направляющей оси с резьбой (оси Z) вращательное движение, тем самым поднимая платформу на фиксированный шаг вверх (рисунок 21).

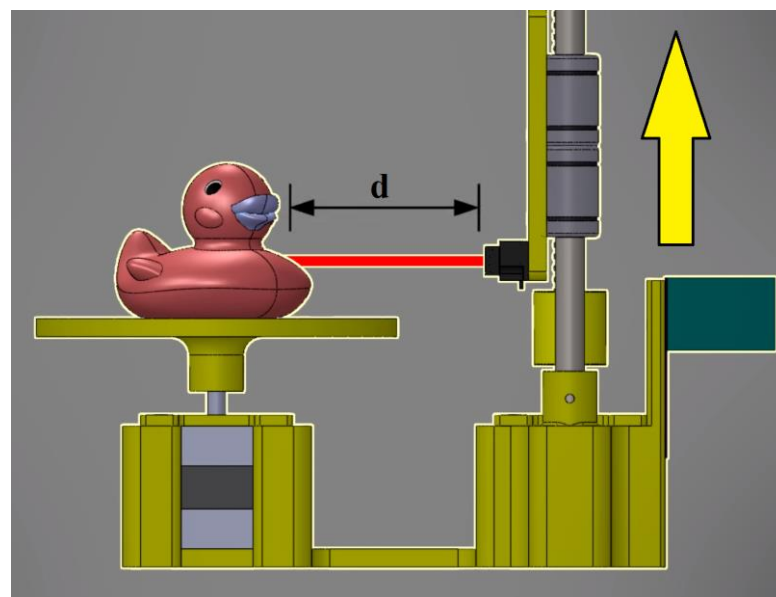


Рисунок 21 – Процесс движения платформы по оси Z

Данный цикл повторяется до достижения крайнего положение по высоте (оси Z), заданной в программе сканирования.

Сканирование производится, благодаря триангуляционному методу измерения расстояний (способу, при котором из излучателя на объект испускается луч, который после отражения от него фиксируется приемником). После процесса сканирования и перевода данных сканирования на SD карту необходимо преобразовать сформированное облако точек в трёхмерный объект. Это можно сделать с помощью специального программного обеспечения, базирующейся на подобных операциях (Meshlab).

В конечном счёте сканирование сводится к получению облака точек, расположение в пространстве которых воссоздает необходимое геометрическое представление просканированной физической модели для дальнейшей инженерной работы.

1.3.2. Условия использования устройства

Объекты, которые могут быть просканированы, должны быть размером не более 10 см в диаметре, и не более 10 см по вертикали. Также объекты должны обладать матовым покрытием, дабы избежать шумов внутри генерируемого облака точек.

Время сканирования при высоте в 10 см и разрешении 0,5 мм за полный цикл работы устройства составляет 1 час.

Измерение расстояния до объекта лежит в диапазоне 2-15 см.

Не все объекты могут быть просканированы в виду ограниченной остроты восприятия инфракрасным дальномером нестандартных геометрических особенностей объекта (пустот и полостей), которые требуют более точного сканирующего модуля. Поэтому устройство предполагает сканирование объектов с преобладанием преимущественно округлых геометрических форм.

1.3.3. Инструкция по использованию устройства

Эксплуатация устройства происходит в следующем порядке:

- 1) Активация 3D сканера при подключении устройства к сети питания;
- 2) Сканирование объекта от низа стола по всей высоте заготовки до высоты, указанной в программе;
- 3) Вывод информации о 3D объекте на SD карту;
- 4) Импорт файла с данными в среду 3D-моделирования;
- 5) Воспроизведение информации в трёхмерном пространстве, приведенной в файле с координатами X,Y,Z.

1.3.4. Определение электрических параметров 3D сканера

Электрические параметры устройства 3D сканера:

- Питание электрической схемы 3D сканера должно производиться от зарядного устройства, которое преобразует переменное напряжение 220 В, потребляемое из сети переменного тока, в постоянное напряжение 12 В с током 1,5 А;
- После зарядного устройства питание схемы должно распределяется по дорожкам разводной печатной платы, поступая на драйверы управления, которые потребляют 1,5 А;
- В схеме предусматривается 2-ух контактный выключатель, используемый для включения и выключения устройства;
- От драйверов управления производится питание шаговых двигателей, которое подаётся линией проводов от разъемов разводной схемы;
- Питание инфракрасного дальномера производится с гнезд разводной печатной платы и составляет 5В;
- На адаптер карт памяти microSD, который подключается к размеченным пинам на плате, подаётся напряжение в 5В;
- Питание микроконтроллера осуществляется по пину RAW, принимающий напряжение в диапазоне 6-18 В.

Выводы по разделу

Таким образом, после постановки цели ВКР и задач по её осуществлению, было проведено глобальное исследование темы 3D сканирования. Был осуществлен обзор и анализ существующих решений 3D сканеров, по основным параметрам которых удалось определить ключевую концепцию разрабатываемого устройства, опираясь на которую будет происходить его техническая реализация.

2. Схемотехническая часть

2.1. Разработка структурной схемы

В ходе схемотехнической части с учётом определенного типа и электрических характеристик разрабатываемого устройства создал структурную схему, изображенную на рисунке 22, которая даёт представление, из каких частей должна строиться электрическая схема устройства.

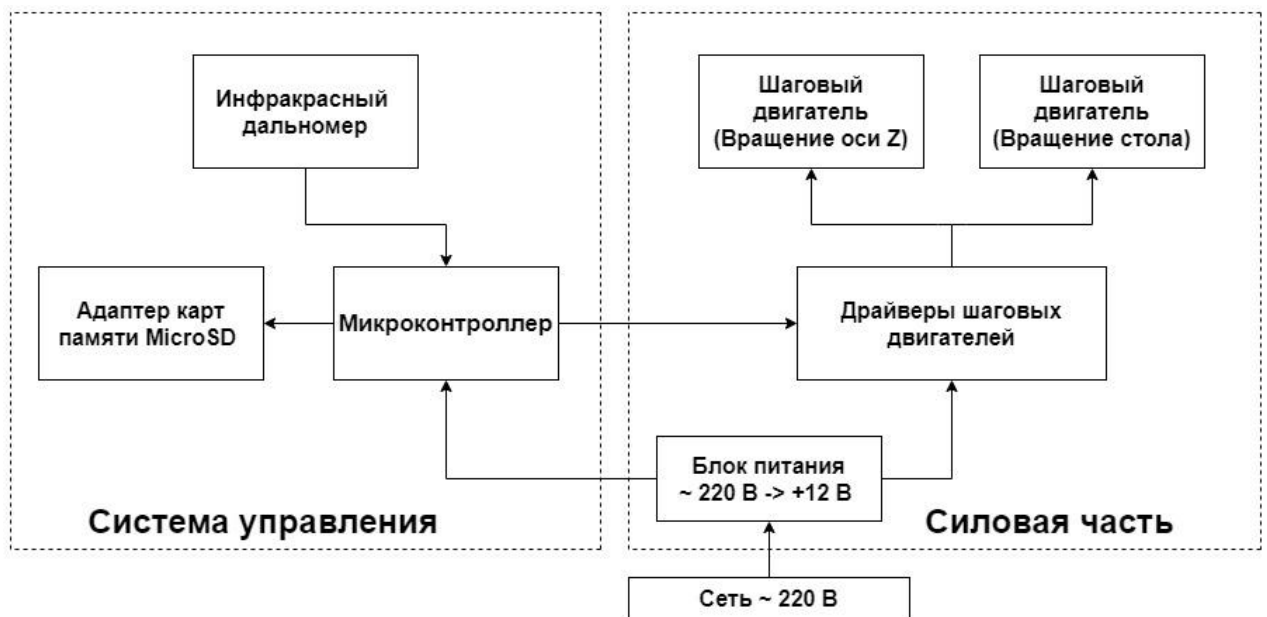


Рисунок 22 – Структурная схема устройства 3D сканера
для построения электрической схемы

Тем самым данная структурная схема отражает суть соединения и пути взаимодействия компонентов системы 3D сканера (рисунок 22).

Опираясь на данную структурную схему и определенные электрические параметры устройства, начал подбор элементов электронной компонентной базы.

В качестве сканирующего устройства выбрал инфракрасный датчик расстояния модели GP2Y0A51SK0F, который является наиболее доступным решением среди аналогов, которые слабо зависят от естественного освещения зоны сканирования.

В качестве силовых компонентов использовал 2 шаговых двигателя Nema17, имеющие оптимальную точность для 3D сканера и обладающие хорошими ходовыми качествами. В качестве драйвера для двигателей использовал 2 микросхемы EasyDriver A3967, которые позволяют организовать микрошаговый режим работы для шаговых двигателей.

В качестве устройства вывода использовал адаптер карт памяти формата microSD с интерфейсом SPI. Данный интерфейс позволяет организовать передачу данных синхронизировано с общим тактовым сигналом, генерируемым ведущим устройством (микроконтроллером). Ведомое устройство, в свою очередь, синхронизирует получение битовой последовательности с тактовым сигналом. Через адаптер памяти на SD карту осуществляется запись текстового файла с расширением .txt, в который заносятся рассчитанные программно отсчёты отсканированного объекта в формате трёхмерных координат X,Y,Z.

В качестве устройства управления выбрал микроконтроллерный модуль Arduino Pro Micro. Данный микроконтроллер обладает простыми аппаратным и программным интерфейсами, а также имеет гибкую скорость обработки данных, и обладает необходимым набором портов ввода/вывода. Данный микроконтроллер имеет компактный форм фактор, 24 контакта и преобразователь напряжения, который способен по пину RAW, принимать напряжение в диапазоне 6-18 В.

Устройство управление будет принимать данные о расстоянии с инфракрасного дальномера, анализировать, рассчитывать координаты облака точек (отсчёты) по геометрии сканируемого объекта, после чего создавать текстовый файл с расширением .txt на SD карте, в который будут заносятся полученные отсчёты. В тоже время микроконтроллер будет посылать сигналы управления на шаговые двигатели через драйверы, синхронизируя тем самым процесс сканирования с передвижением дальномера по высоте заготовки (по высоте сканируемой детали).

В качестве блока питания системы выбрал обычное зарядное устройство ROBITON с номиналом 12 вольт, который подает текущее значение напряжения на драйверы шаговых двигателей, а также осуществляет питание всей схемы и её компонентов в целом.

2.2. Разработка электрической принципиальной схемы

После выбора элементов схемы, опираясь на составленную структурную схему и определенные электрические параметры на устройство, разработал схему электрическую принципиальную 3D сканера (рисунок 23).

Схема собрана на микроконтроллере DD2, который управляет работой двух шаговых двигателей M2, M3, посредством подачи управляющих импульсов на их драйверы DD3, DD4. Другим соединением микроконтроллер DD2 принимает и пересчитывает данные, получаемые с оптического инфракрасного дальномера DA1, а затем записывает преобразованные данные через адаптер карт памяти microSD DD1 на саму карту. Драйверы шаговых двигателей DD3, DD4 питаются от зарядного устройства через вход принимающий от 6 до 30 В.

Питание схемы осуществляется через зарядное устройство A1, которое преобразует напряжение 220 В из сети переменного напряжения в необходимые +12 В для питания схемы. В разрыв между зарядным устройством A1 и гнездом X1.2 установлен ключ (выключатель) К для замыкания/размыкания цепи. Параллельно питанию подсоединяется вентилятор M1 для охлаждения электроники. Также имеется кнопка сброса SB, которая выполняет сброс всего цикла работы устройства.

Питание +5 В, подаваемые на инфракрасный дальномер DA1 и на адаптер карт памяти microSD DD1, осуществляется при преобразовании, поступающего на пин RAW, напряжения 12 В через встроенный регулятор напряжения. Перечень элементов, входящих в электрическую принципиальную схему привёл в приложении А.

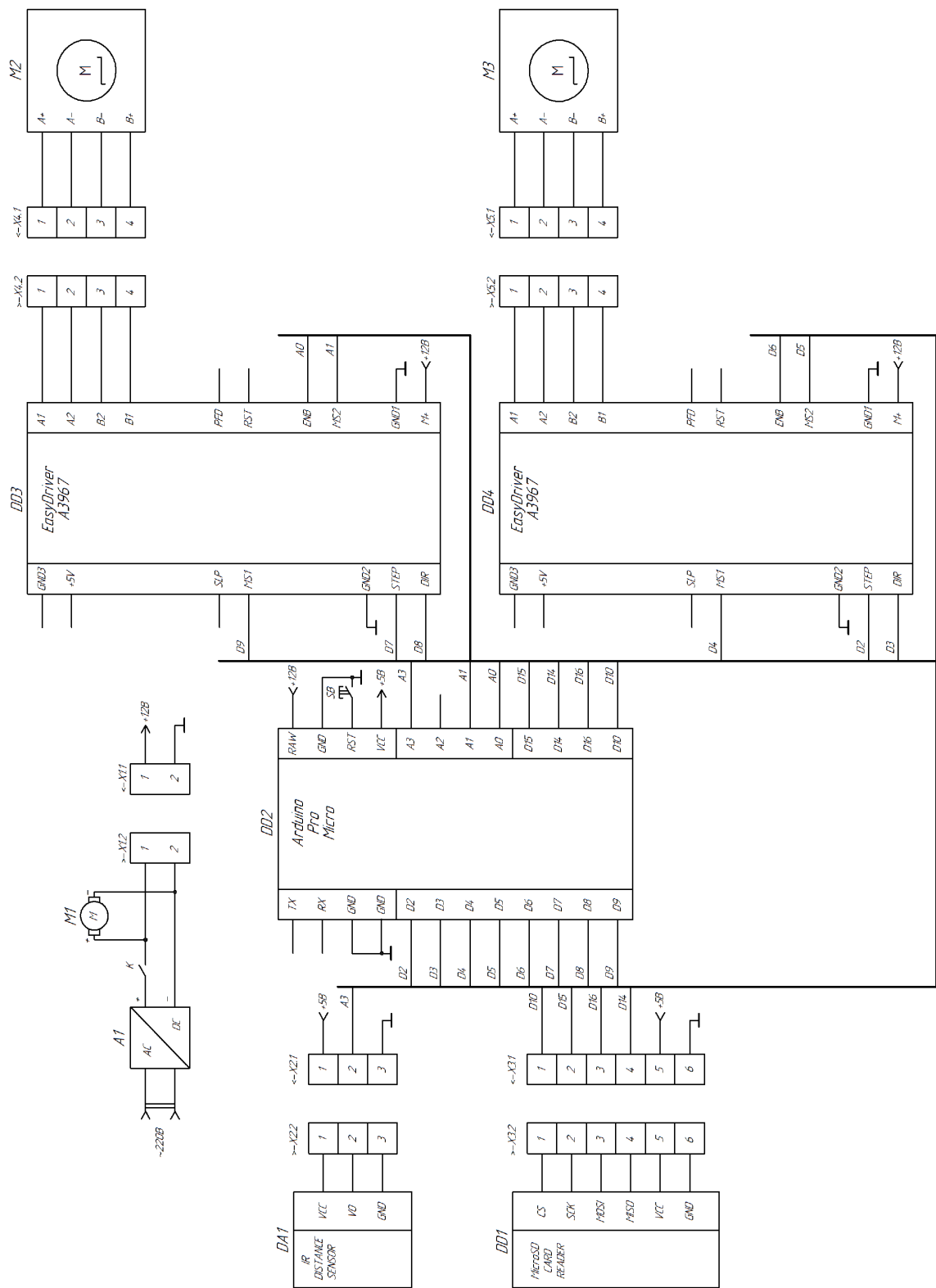


Рисунок 23 – Электрическая принципиальная схема 3D сканера

2.3. Разработка силовой части схемы

Силовая часть входит в электрическую схему, приведенную на рисунке 23. Она включает в себя зарядное устройство A1, которое подаёт напряжение в +12 В на драйверы шаговых двигателей DD3, DD4, которые по проводам соединяются с шаговыми двигателями M2, M3 через разъемы X4, X5.

Данная часть схемы рассчитана на преобразование и подачу напряжения поступающего с блока питания на силовые органы устройства (шаговые двигатели) через драйверы управления. Этот процесс происходит при получении сигнала управления, поступающего с логической части устройства (микроконтроллера Arduino Pro Micro) на непосредственно силовую часть устройства.

Первым из компонентов, входящий в силовую часть электрической схемы устройства (рисунок 23) является драйвер шагового двигателя EasyDriver A3967 (рисунок 24).

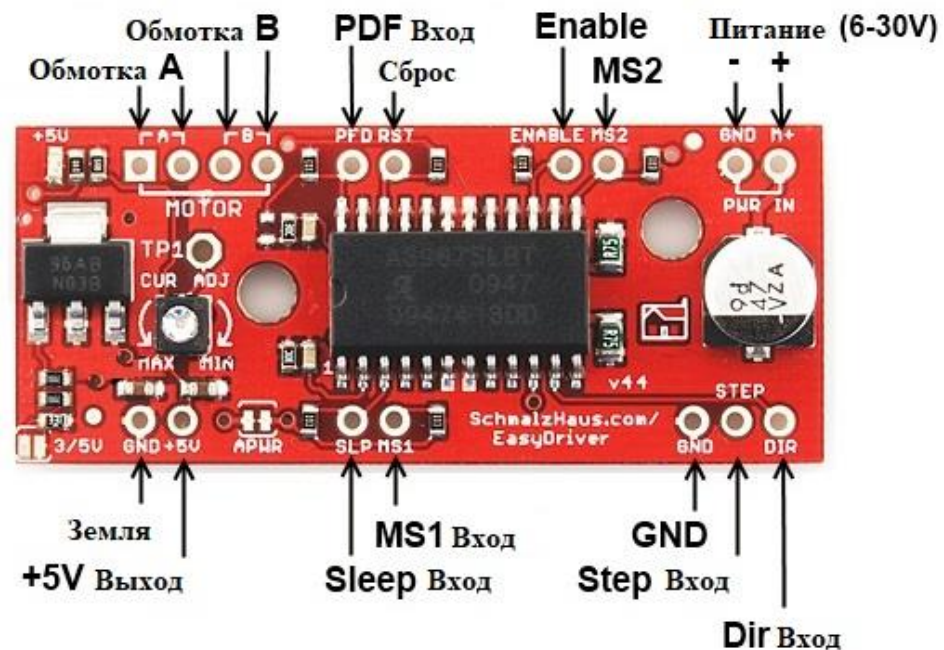


Рисунок 24 - Драйвер шагового двигателя EasyDriver A3967

Драйвер предназначен для управления биполярным шаговым двигателем с поддержкой «микрошага» и базируется на микросхеме A3967.

Драйвер имеет следующие выводы:

MOTOR и выходы A и B - Подключение обмоток шагового двигателя.
(A+ A- B+ B-).

- PFD - Вход скорость нарастания ШИМ.
- RST - Сброс драйвера, при низком уровне сбрасывает внутренний транслятор и отключает все выходные драйверы.
- ENABLE - При низком уровне, отключатся все выходы драйвера.
- MS1 и MS2 - Управление микрошаговым режимом. По умолчанию входы притянуты к питанию и выставлен шаг 1/8. Для установки полного шага, на оба входа нужно подать низкий уровень, для полушага только на MS2, для 1/4 шага, только на MS1. (полный шаг (0,0), полушаг (1,0), шаг 1/4 (0,1) и шаг 1/8 (1,1).
- PWR IN и вход M+ - Напряжение питания драйвера и моторов, также это напряжение подается на стабилизатор LM317 для питания логической части микросхемы.
- +5V - Выход напряжения со стабилизатора LM317, используемый для питания Arduino.
- SLP - Режим сна, если подать низкий уровень, будет отключена внутренняя схема для минимизации потребления энергии.
- STEP – Шаг, задаётся при переходе с низкого уровня на высокий, тогда драйвер делает один шаг или микрошаг в микрошаговом режиме.
- DIR - Состояние входа (высокий\низкий) определяет направление вращения двигателя.
- GND - Земля, все земли соединены.

Управление шаговым двигателем осуществляется заданием режима для его работы в полношаговом (рисунок 25) и микрошаговом (рисунок 26) режимах.

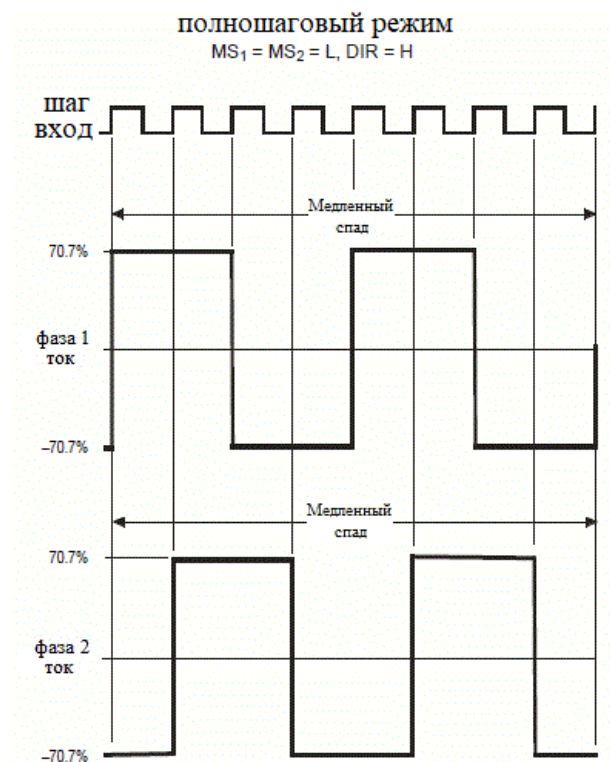


Рисунок 25 - График работы драйвера в полношаговом режиме

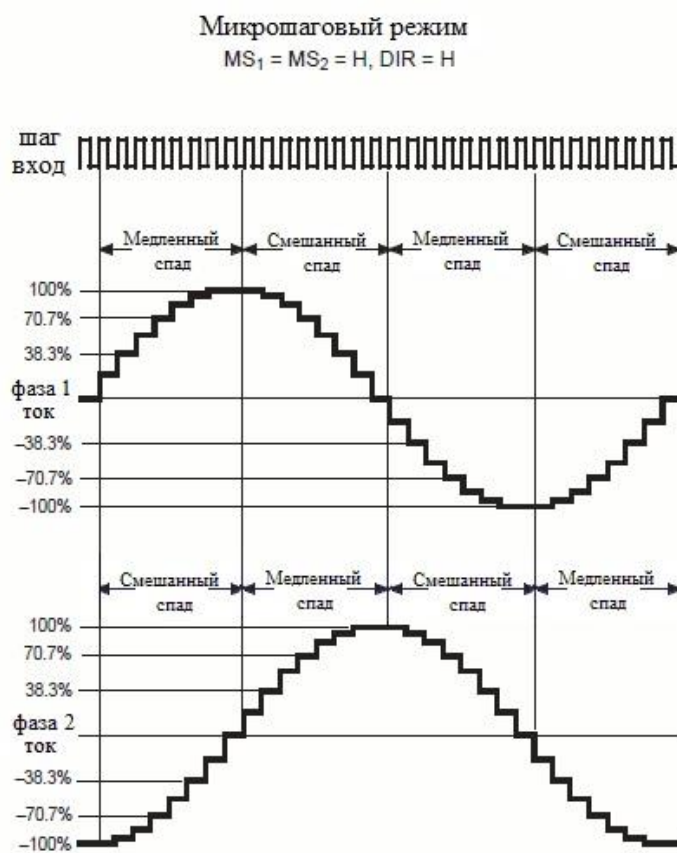


Рисунок 26 - График работы драйвера микрошаговом режиме

В полношаговом режиме, драйвер питает обмотки двигателя полным током, а направление тока в обмотках двигателя изменяется с каждым шагом. Данный режим считается штатным режимом работы двигателя. Главное достоинство - простота реализации. Как недостаток, двигатель сильнее подвержен вибрациям и резонирует на низких скоростях.

В микрошаговом режиме происходит деление шага (например, 1/8, 1/4, 1/2), с каждым шагом обмотки запитаны не полным током, а уровнем изменяемым по синусоидальному закону. В этом режиме вал фиксируется в промежуточных положениях между шагами, что позволяет увеличить количество шагов и точность позиционирования вала двигателя, и уменьшить вибрацию двигателя, особенно на низких скоростях.

В данной работе стационарного (полношагового) режима вполне достаточно для реализации оптимального вращения заготовки.

Основные параметры драйвера шагового двигателя EasyDriver A3967 сведены в таблице 7.

Таблица 7 – Основные параметры драйвера шагового двигателя EasyDriver A3967

Параметр	Значение
Функциональное назначение	Драйвер
Максимальный ток на одну фазу, мА	750
Напряжение привода двигателя, В	7-30
Возможность ограничения выходного тока, мА	150-750
Микрошаговые режимы работы	1/8, 1/4, 1/2

Драйвер EasyDriver A3967 управляет шаговым двигателем Nema 17HS4401, который представляет из себя биполярный двигатель с высоким крутящим моментом.

Данный двигатель способен поворачиваться на заданное число шагов. За один шаг совершает оборот на $1,8^\circ$, соответственно полный оборот на 360°

осуществляет за 200 шагов. Шаговый двигатель Nema 17HS4401 приведен на рисунке 27.



Рисунок 27 – Шаговый двигатель Nema 17HS4401

Биполярный двигатель имеет две обмотки, по одной на каждую фазу, полярности на которых изменяется драйвером для изменения направления магнитного поля. Соответственно, от двигателя отходят четыре провода от обмоток шагового двигателя (рисунок 28).

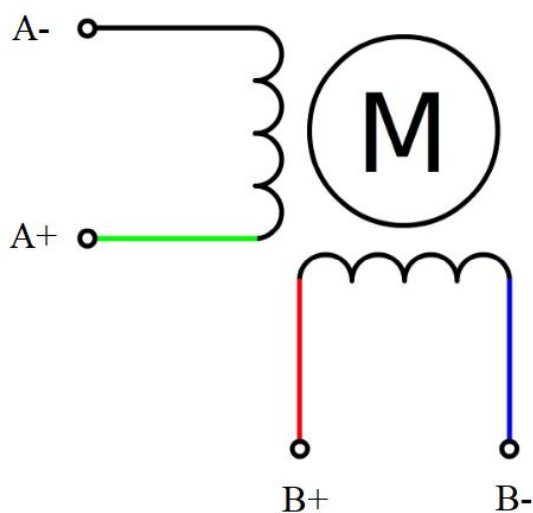


Рисунок 28 - Обмотки шагового двигателя

В таблице 8 приведены циклы полярности обмоток шагового двигателя, где за каждый цикл(шаг) соблюдается определенная полярность, т.е. на выводах обмоток присутствует знак полярности плюс или минус.

Таблица 8 – Циклы полярности обмоток шагового двигателя

№	A-	B+	A+	B-
1	+	+	-	-
2	-	+	+	-
3	-	-	+	+
4	+	-	-	+

Основные параметры шагового двигателя Nema 17HS4401 сведены в таблице 9.

Таблица 9 – Основные параметры шаговых двигателей Nema17 17HS4401

Параметр	Значение
Функциональное назначение	Двигатель
Тип	Биполярный
Форма контактов Сопротивление фазы, Ом	1.5
Крутящий момент, кг x см	5.5
Момент удержания, кг x см	2.8
Индуктивность, мГн;	2.8
Длина вала, мм	24
Угловой шаг, °	1.8 (200 шагов на оборот)
Макс. ток обмотки, А	1.7
Диаметр вала, мм	5
Число фаз, шт	2
Габариты корпуса, мм	42×42×48

2.4. Разработка системы управления

Система управления также входит в электрическую схему, приведенную на рисунке 23. Система управления состоит из управляющего устройства DD2 (микроконтроллера Arduino Pro Micro), устройства сканирования DA1 (инфракрасного дальномера) и устройства получения DD1 (адаптера карт памяти формата microSD).

Основой системы управления служит плата Arduino Pro Micro на микроконтроллере ATmega32U4, который программируется благодаря встроенному загрузчику, который позволяет загружать код в плату без использования внешнего программатора.

Питание платы можно проводить по micro USB с напряжением 5 В. Также на плате предусмотрен регулятор напряжения, который может принимать напряжение до 18 В. Данным напряжением осуществляется питание схемы, которое поступает по пину RAW.

Плата Arduino Pro Micro имеет следующие выводы:

- 18 цифровых выводов, которые могут работать как входы, так и как выходы. Напряжение на выводах 3,3 В, при токе в 40мА;
- последовательный интерфейс с пинами TX и RX;
- I2C интерфейс с пинами SDA и SCL;
- ШИМ выводы: 3, 5, 6, 9, 10;
- SPI интерфейс с пинами MISO, MOSI и SCK;
- Светодиоды сигнализирующие: наличие питания, RX и TX;
- 9 аналоговых входов A0-A3 и A6-A10;
- RESET – вывод для перезагрузки микроконтроллера, аналогичен физической кнопки сброса.

Модуль Arduino Pro Micro приведен на рисунке 29.

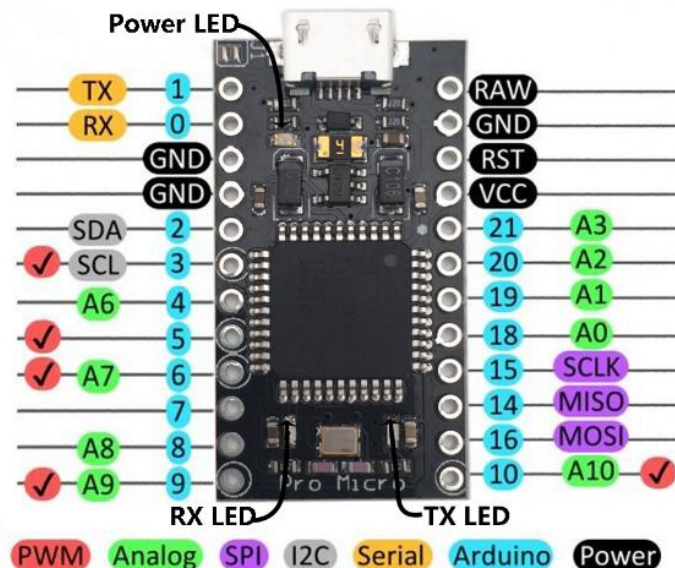


Рисунок 29 – Модуль Arduino Pro Micro

К заданным пинам микроконтроллера подводятся периферийные элементы, согласно заданной схеме подключения и суть процесса управления основывается на подаче импульсов, которые приводят к выполнению к той или иной логической операции для совершения работы какого то функционального блока (например, шагового двигателя, через драйвер).

Управление органами управления(шаговыми двигателями) в момент сканирования происходит посредством подачи сигналов управления с микроконтроллера ATmega32U4. На его основе можно организовать синхронную подачу управляющих импульсов на шаговые двигатели, и параллельно считывать данные со сканирующего устройства (инфракрасного дальномера).

Основные параметры микроконтроллера сведены в таблице 10.

Таблица 10 – Основные параметры микроконтроллера Arduino Pro Micro

Параметр	Значение
Микроконтроллер	ATmega32U4
Тип ядра	AVR
Разрядность, бит	8

Продолжение таблицы 10

Тактовая частота, МГц	16
Объем флеш-память, Кбайт	32
Объем ОЗУ-памяти, Кбайт	2,5
Количество цифровых входов/выходов	18
Количество аналоговых входов	9
Шим выводы	5
Рабочее питание, В	3.3
Входное напряжение (вывод RAW), В	6-18
Максимальный общий ток, мА	500 мА
Максимальный ток вывода, мА	10 мА
USB-разъем	Micro USB

Сканирующее устройство представлено инфракрасным датчиком расстояния Sharp GP2Y0A51SK0F (рисунок 30), с выходным аналоговым напряжением, который работает на методе триангуляции, рассчитанный на измерение расстояния в диапазоне 2 – 15 см.

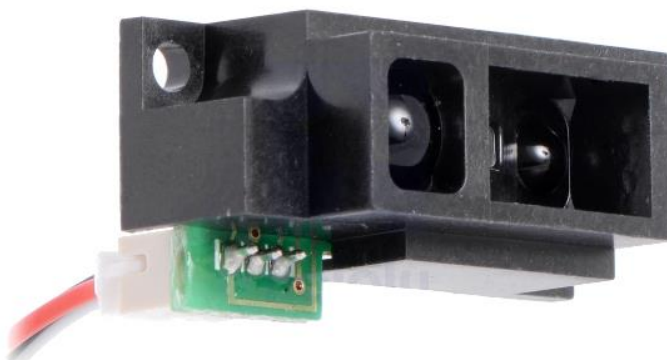


Рисунок 30 – ИК датчик расстояния Sharp GP2Y0A51SK0F

Этот инфракрасный датчик является более экономичным решением по сравнению с ультразвуковым дальномером, а также обеспечивает лучшее быстродействие по сравнению с другими инфракрасными датчиками. Связь с микроконтроллером реализуется за счёт соединения аналогового выхода с аналого-цифровым преобразователем для измерения значений.

Работа данного датчика основывается на методе триангуляции. Данный метод менее подвержен влиянию со стороны интерференции излучения, а также слабо зависит от отражения от поверхностей, которые могут быть из различных материалов и иметь разный цвет.

Конструктивно ИК датчик расстояния Sharp GP2Y0A51SK0F состоит из излучателя - инфракрасного светодиода (LED) с линзой, а также приёмника, выполненного на позиционно-чувствительном фотоэлементе (Position-Sensitive Detector, PSD) с линзой.

Физический смысл работы датчика заключается в том, что излучатель испускает световой луч на объект, который отражается от него и попадает на приёмник, проходя через его линзу. В результате цикла измерения испускаемый и отраженный лучи образуют треугольник «излучатель — объект отражения — приемник».

На рисунке 31 приводится, которая показывает путь отраженного луча на различных расстояниях.

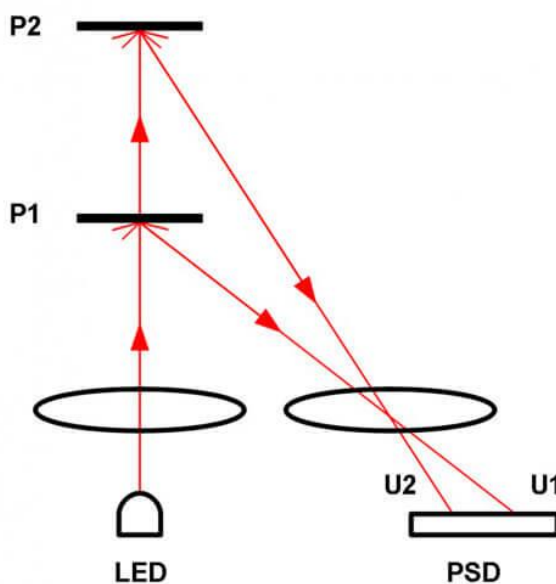


Рисунок 31 – Иллюстрация, поясняющая метод определения расстояния с помощью ИК датчика

Определенное расстояние до объекта напрямую зависит угол отражения от него. В зависимости от местоположения падающего на PSD луча меняется его проводимость. Затем происходит преобразование

проводимости в напряжение, которое оцифровывается аналого-цифровым преобразователем микроконтроллера, после чего вычисляется расстояние до объекта.

Ниже представлена временная диаграмма отражающая время измерения расстояния за цикл и с каким запаздыванием до следующего измерения оно может следовать (рисунок 32).

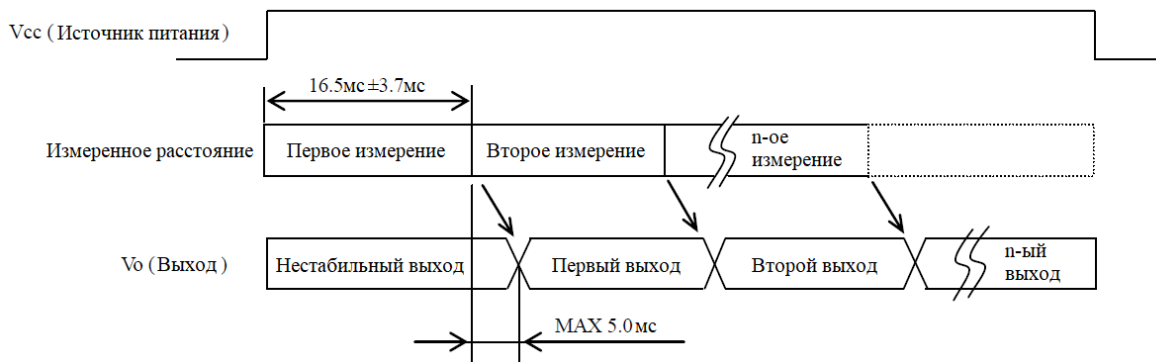


Рисунок 32 - Временная диаграмма работы ИК датчика расстояния
GP2Y0A51SK0F

Отношение выходного напряжения датчика к обратно пропорционально изменяемому расстоянию практически линейно на всём диапазоне измерений. По графику изменения аналогового выходного напряжения как функции от обратно пропорционально изменяющегося расстояния до отражённого объекта можно определять приблизительное расстояние, сопоставляя выходное напряжение (V) с расстоянием (см) (Приложение Б).

Технические параметры на датчик приведены в таблице 11.

Таблица 11 – Основные параметры инфракрасного датчика расстояния GP2Y0A51SK0F

Параметр	Значение
Тип датчика	расстояния
Вид датчика	инфракрасный
Напряжение питания(постоянное), В	4.5 - 5.5
Диапазон измерения расстояния, см	2 - 15

Устройством приёма и получения данных выступает адаптер карт памяти microSD с интерфейсом SPI(Serial Peripheral Interface) для Arduino (рисунок 33). Данный модуль позволяет подключить карту формата microSD к Arduino и использовать ее для логирования показаний с инфракрасного дальномера, хранения конфигурационного файла с данными показаниями в преобразованной форме.

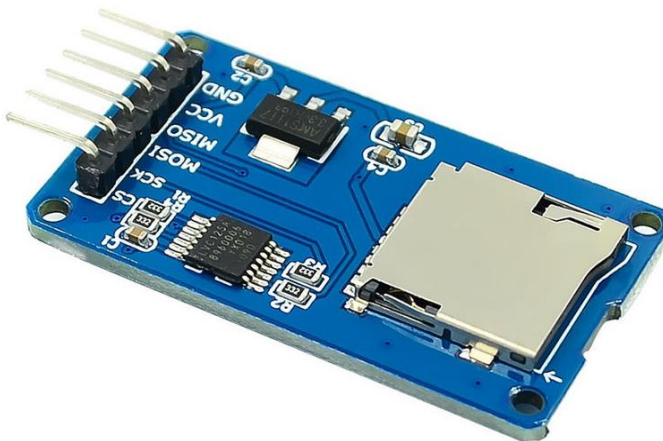


Рисунок 33 – Адаптер карт памяти microSD с интерфейсом SPI

Адаптер microSD карты имеет шесть контактов из которых 2 используются для питания схемы модуля:

- GND - питание модуля «земля»;
- VCC - питание модуля «3.3 В — 5В».

А другие четыре контакта используются для подачи цифровых сигналов и входят в SPI интерфейс:

- MOSI — выход ведущего, вход ведомого. Служит для передачи данных от ведущего устройства(микроконтроллера) ведомому(карте памяти);
- MISO — вход ведущего, выход ведомого. Служит для передачи данных от ведомого устройства ведущему;
- SCK — последовательный тактовый сигнал. Служит для передачи тактового сигнала для ведомых устройств;
- CS — выбор микросхемы, выбор ведомого.

Технически SPI интерфейс является последовательным синхронным стандартом передачи данных, предназначенным для обеспечения сопряжения микроконтроллера и периферии (рисунок 34).

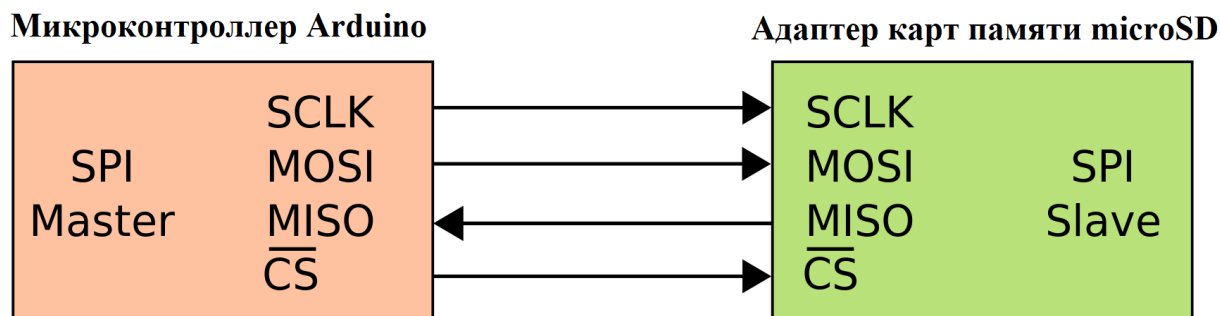


Рисунок 34 - Ведущее (Master) и ведомое (Slave) устройства интерфейса SPI

Ведущее устройство (микроконтроллер) посылает сигнал CS для того ведомого, обмен данными с которым будет осуществляться, а также задаёт скорость обмена данными, полярность и фазу тактирования. Обмен данными происходит в режиме полного дуплекса, что позволяет устройствам по шине одновременно передавать и принимать данные.

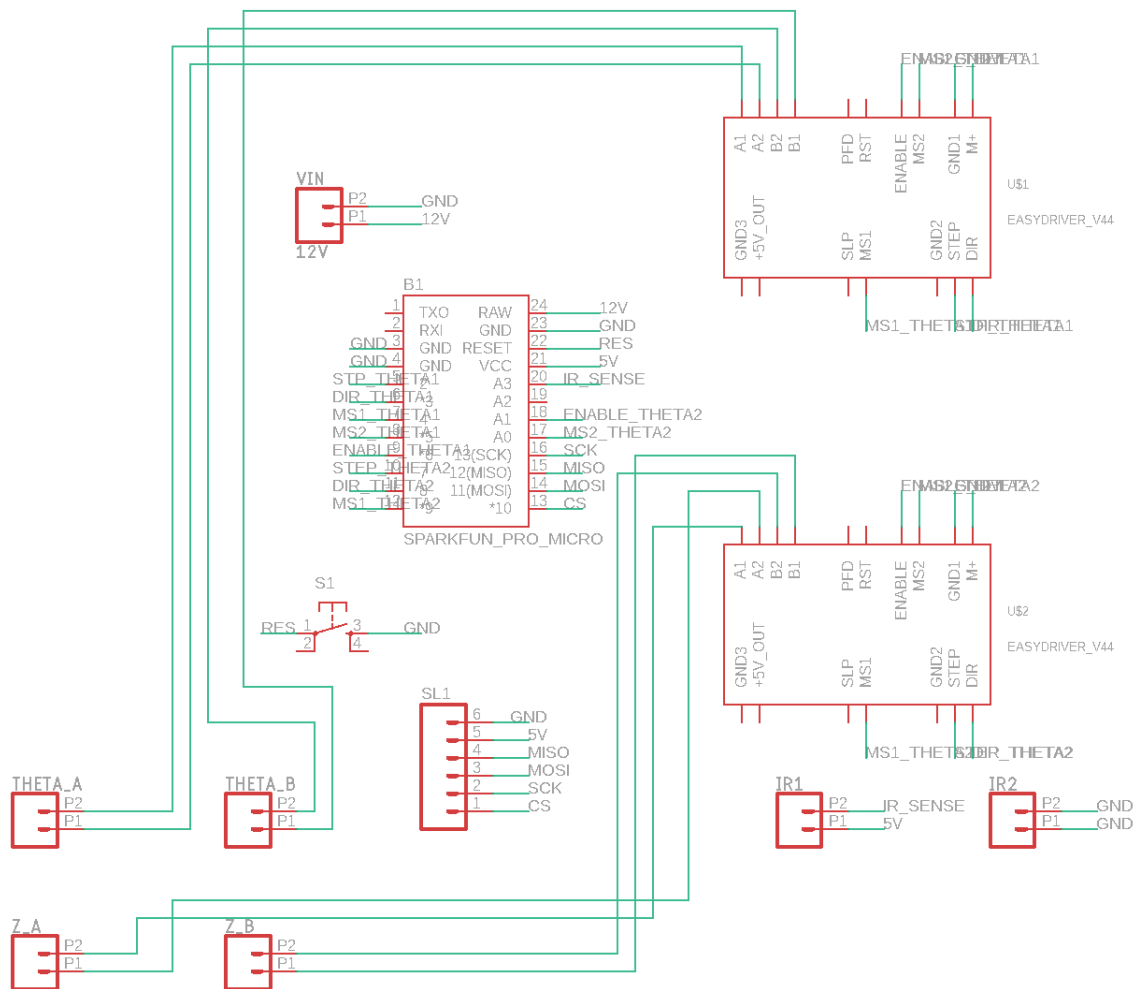
Технические параметры на адаптер карт памяти microSD приведены в таблице 12.

Таблица 12 – Основные параметры адаптер карт памяти microSD

Параметр	Значение
Поддержка карт	micro SD, micro SDHC
Интерфейс	SPI
Напряжение питания, В	4.5 - 5.5, или 3.3
Напряжение интерфейса SPI, В	5, 3.3
Габариты, мм	42x24x12

2.5. Разработка печатного узла

Для изготовления устройства 3D сканера и упрощения связи между составляющими его компонентами, разработал двухслойную разводную печатную плату, представляющую собой основание из текстолита с проводящими дорожками и отверстиями для монтажа элементов и крепления. Разводку печатной платы произвёл в программном пакете EAGLE. Первым шагом с помощью встроенного редактора схем собрал электрическую схему устройства, связав между собой функциональные элементы, которые взял из сторонней библиотеки элементов (рисунок 35а).



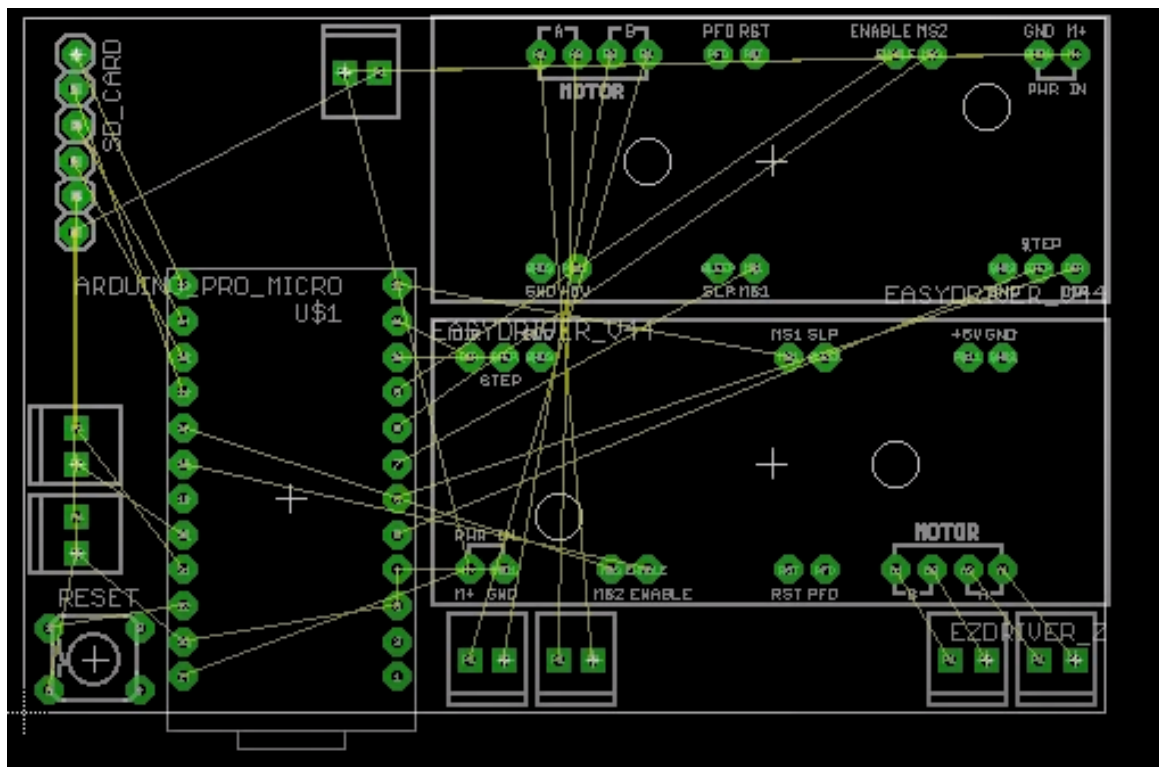


Рисунок 35б – Печатная плата с корпусами элементов в EAGLE

Далее выполнил авто трассировку печатной платы, определив расположение проводников на печатной плате (рисунок 35в).

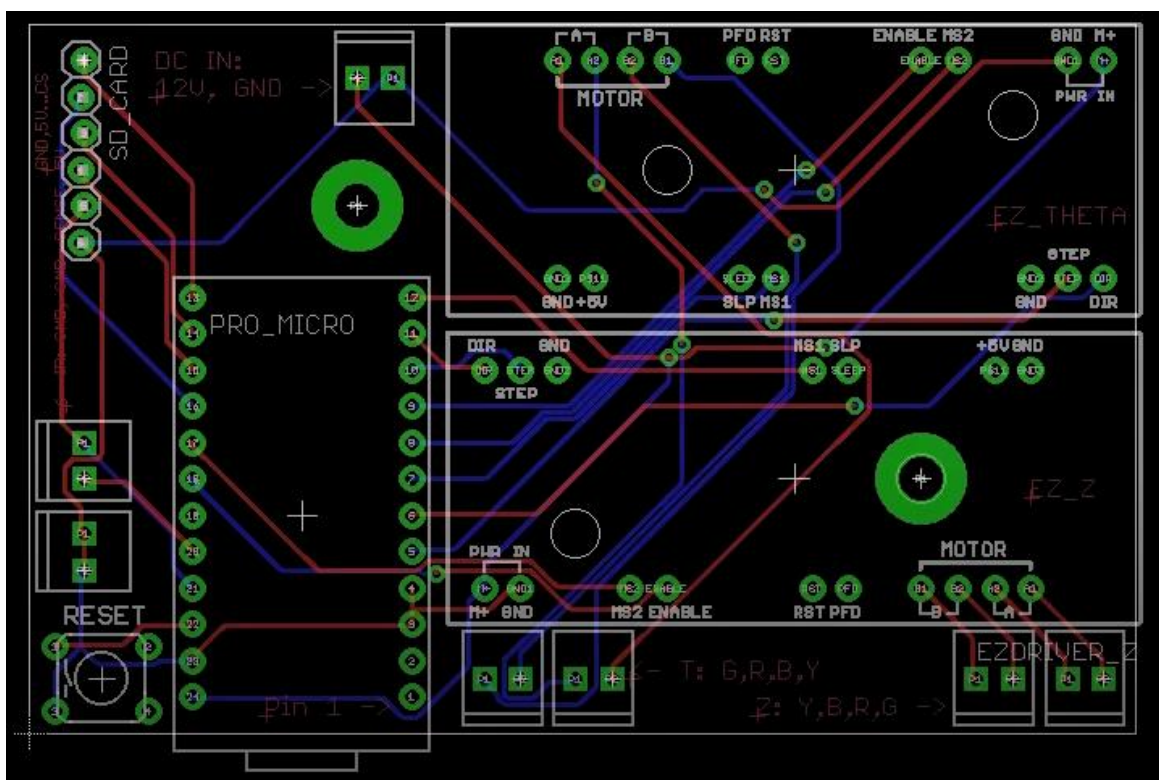


Рисунок 35в – Печатная плата после авто трассировки в EAGLE

Технические параметры платы приведены в таблице 13.

Таблица 13 – Технические параметры разводной печатной платы

Параметр	Значение
Количество проводящих слоёв	2
Способ монтажа	в отверстия на плату
Способ крепления	по отверстиям
Напряжение питания, В	12
Рабочий ток, А	0,05

По нажатию кнопки сброса, которая монтируется на разводной печатной плате, осуществляется сброс системы микроконтроллера.

В качестве разъемов для монтажа, к которым будут подсоединяться провода от внешних устройств, выбрал разъемы kf120-2.54-2P, имеющие 2,5 мм расстояние между выводами (рисунок 36).



Рисунок 36 – Разъемы kf120-2.54-2P

После завершения процесса разводки печатной платы в EAGLE, сохранил полученный результат в файл с расширением .CXF и открыл программой КОМПАС-3D V16, в которой произвёл окончательное оформление чертежа (рисунок 37, а-в).

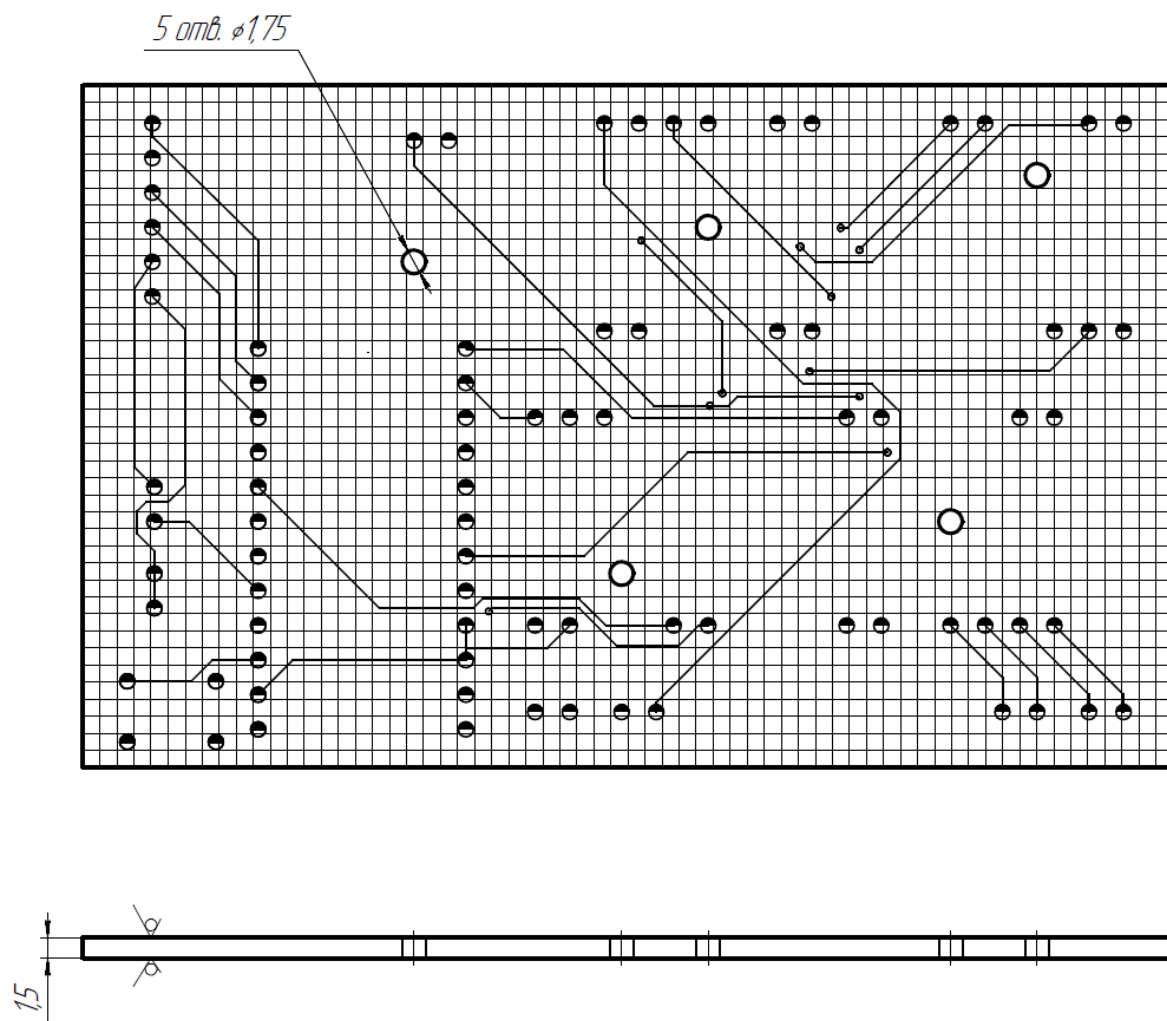


Рисунок 37а – Вид печатной платы со 1 стороны

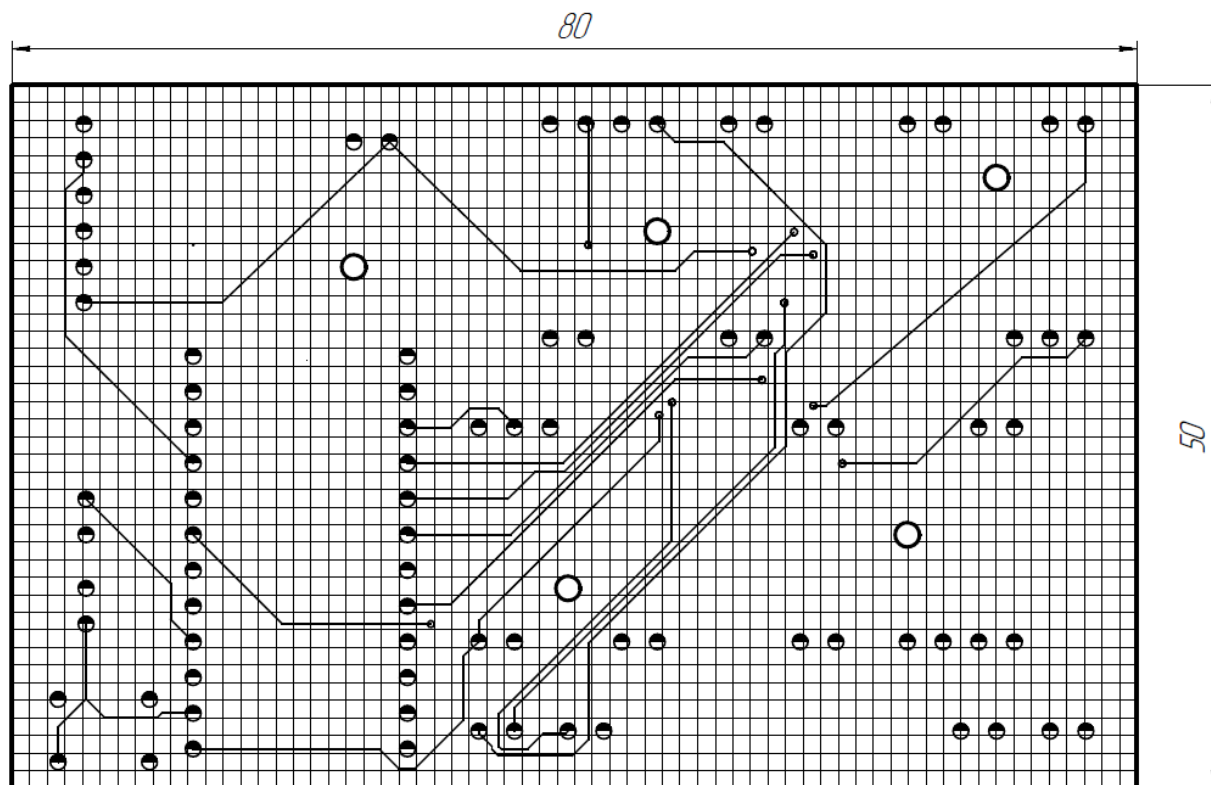


Рисунок 37б – Вид печатной платы со 2 стороны

<i>Обозначение отв.</i>	<i>Диаметр отв., мм</i>	<i>Диаметр контактной площадки, мм</i>	<i>Количество отв.</i>
●	1,1	1,5	82

1. Плату изготовить комбинированным способом
2. Плата должна соответствовать требованиям ГОСТ 23752-79.
3. Шаг координатной сетки 1,25 мм
4. Минимальная ширина проводника 0,8 (1,25) мм.
5. Минимальное расстояние между проводниками 0,3 мм.
6. Шероховатость $R_z=4,0$

Рисунок 37в – Требования к изготовлению печатной платы.

Выводы по разделу

Таким образом, в ходе разработки схемотехнической части устройства была составлена структурная схема, по которой удалось создать электрическую принципиальную схему, состоящую из силовой и управляющей части. Также, с целью упрощения соединения компонентов схемы между собой, была разработана печатная плата.

3. Разработка конструкции

В предисловие для разработанной электрической схемы устройства необходимо было спроектировать оптимальное конструктивное решение для её работоспособности, исходя из определенной концепции устройства и его принципа работы.

В качестве инструмента для проектирования каркаса 3D сканера, воспользовался CAD (Computer-Aided Design) программой – Autodesk PowerShape. В данной системе 3D моделирования удалось спроектировать каркас устройства.

3.1. Разработка каркаса

Разработка конструкции устройства заключалась в задумке оптимальных механических частей для цельной стыковки и слаженной работы устройства в конечном. Пользуясь методами каркасного и поверхностного моделирования, были разработаны элементы каркаса 3D сканера, общий вид которых приведен на рисунке 38.

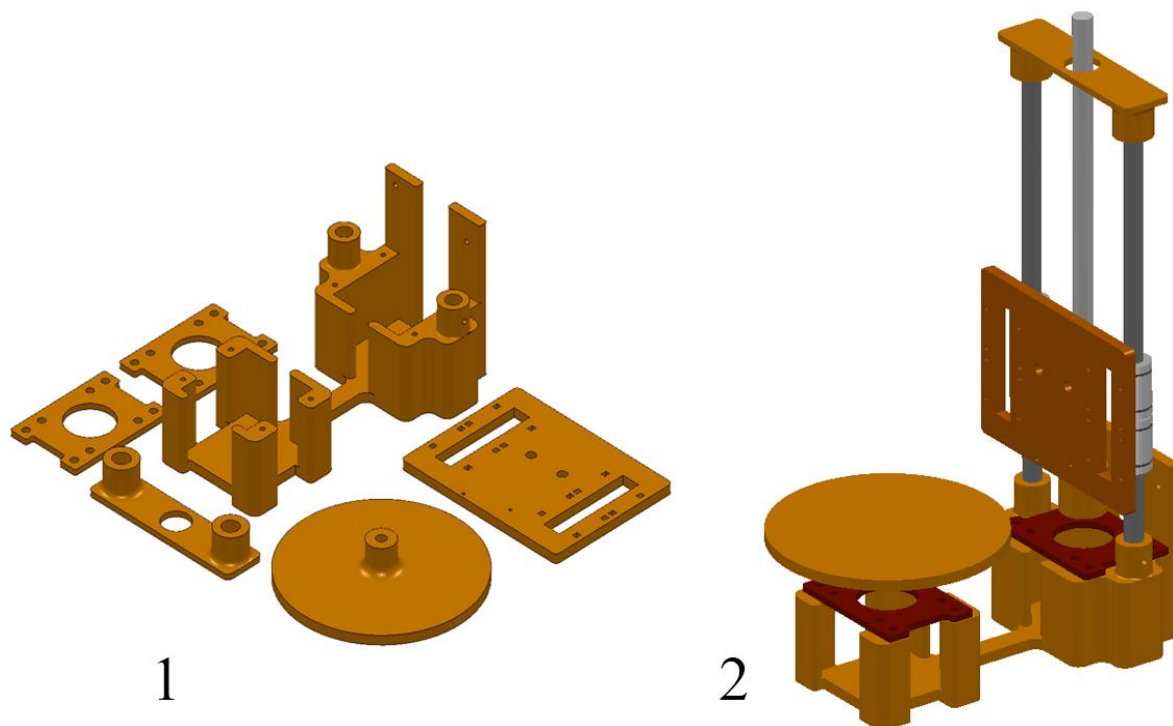


Рисунок 38 – Проектирование каркаса в PowerShape: 1 – Части каркаса;
2- Каркас в собранном виде

Каркас 3D сканера состоит из следующих конструктивных элементов:

- 1) Базы с ячейками для шаговых двигателей и отверстиями для вставки направляющих стержней;
- 2) Стол с отверстием под насадку на вал шагового двигателя;
- 3) 2-ух опорных стержней с длиной 200 мм и диаметром 8 мм;
- 4) Концевого колпака с выемками под опорные стержни;
- 5) Сканирующей платформы с отверстиями под крепление к резьбовой оси Z и инфракрасного дальномера, а также крепления под стяжки;
- 6) Резьбовой оси Z длиной 8 мм на 215 мм с кронштейном под крепление сканирующей платформы. По данной оси будет осуществляться передвижение сканирующего сенсора.
- 7) Скользящих подшипников диаметром 8 мм, выступающими направляющими механизма передвижения по оси Z.
- 8) Покрышек с отверстиями под крепления на шаговые двигатели.

Для печати на 3D принтере экспортировал спроектированные модели частей каркаса в формате STL из CAD системы в слайсер Simplify3D - программу, разбивающую трехмерную модель на слои, тем самым подготавливая её к печати на 3D-принтере (рисунок 39).

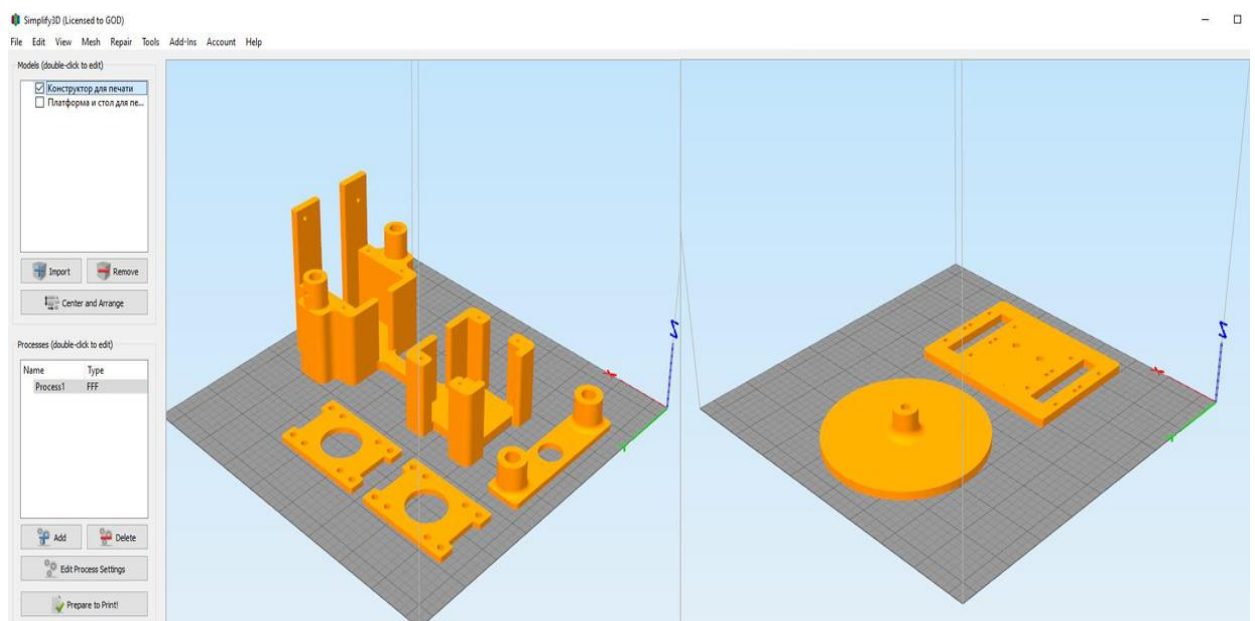


Рисунок 39 – Подготовка основных частей каркаса к печати на 3D принтере

В слайсере Simplify3D 3D-модели деталей программно подготовил к процессу печати, задав необходимые параметры 3D печати: ширину экструзии, т.е. величину выходящего материала (пластика) из экструдера, а также скорость перемещения осей и температуры подогрева стола, после чего сформировал управляющую программу на языке G-Code (стандартном языке для ЧПУ систем). Визуально результат вывода управляющей программы представлен набором виртуальных траекторий прохода, рассчитанных по 3D моделям самих частей каркаса (рисунок 40).

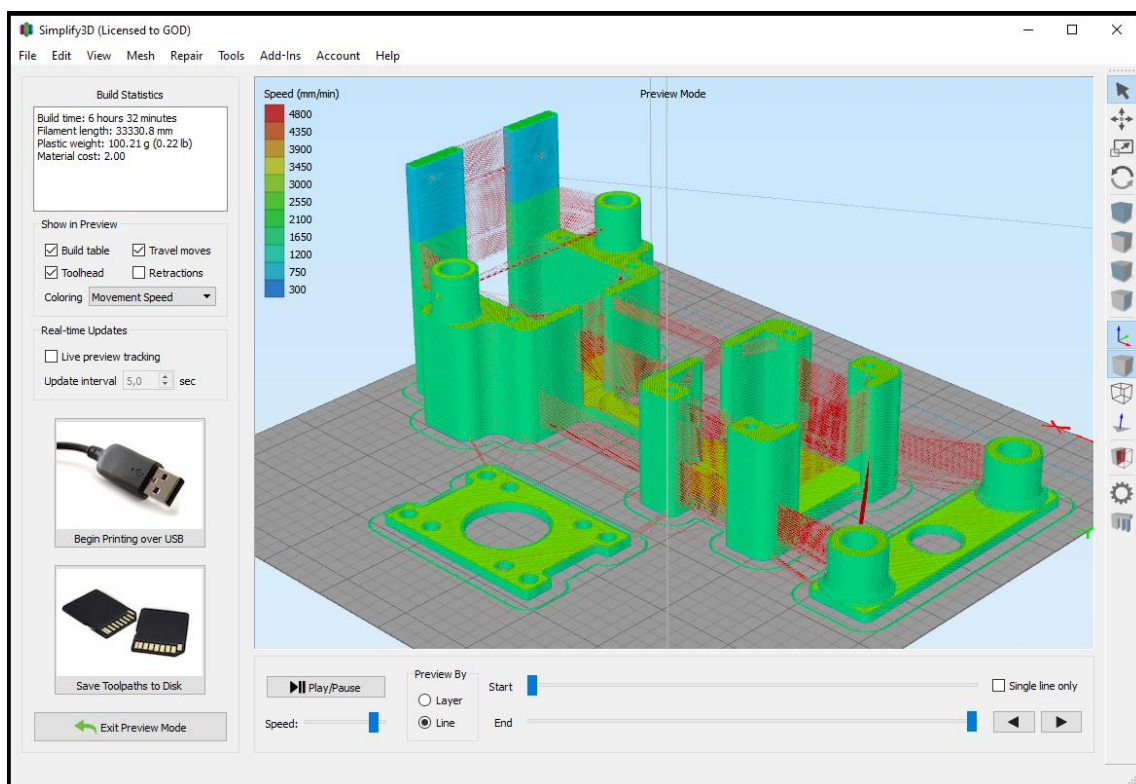


Рисунок 40 – Результат вывода управляющей программы
для печати каркаса

Аналогичным образом вывел управляющие программы для остальных частей спроектированного каркаса (поворотного стола и сканирующей платформы).

После печати на 3D принтере заданных частей каркаса и сборки их воедино с электрическими и механическими компонентами устройства получился следующий результат (рисунок 41).

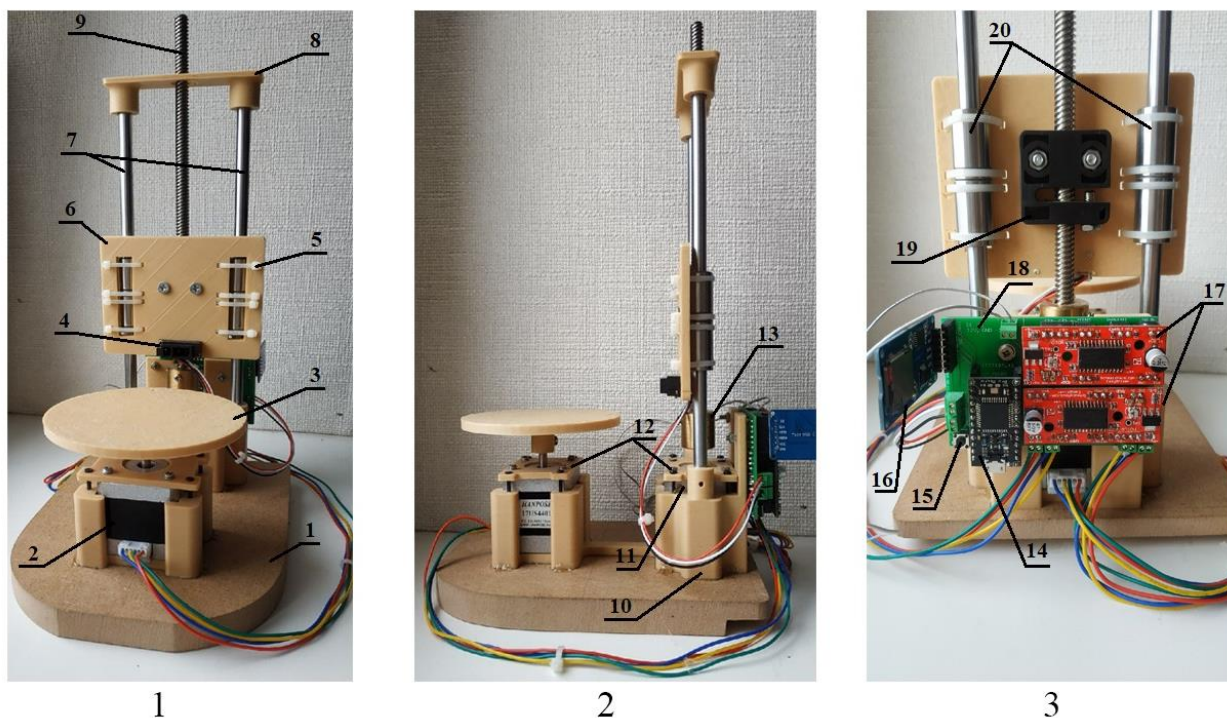


Рисунок 41 – Каркас 3D сканера в собранном виде:

1 - Вид спереди; 2 - Вид сбоку; 3 – Вид сзади

Устройство каркаса 3D сканера (рисунок 41) состоит в следующем:

Вся несущая конструкция 10, представляющая из себя базу с ячейками для размещения электрических и механических компонентов устройства, располагается на подложке из бруска древесины 1. На шаговый двигатель поворотного стола 2 насаживается сам стол 3, после чего подтягивается болтом к валу двигателя. Инфракрасный датчик расстояния 4 крепится на затяжных винтах к сканирующей платформе 6, которая подсоединяется, благодаря кронштейну 19 к резьбовой оси Z 9. В кронштейне также предусмотрено отверстие с резьбой, по которому сканирующая платформа 6 провинчивается по оси Z 9. Для стабилизации перемещения по ней предусмотрены 2 опорных стержней 7 с длиной 200 мм и диаметром 8 мм, на которые насаживается сканирующая платформа 6, благодаря направляющим скользящим подшипникам 20. Они затягиваются стяжками 5 к самой сканирующей платформе через предусмотренные отверстия. Перемещение сканирующей платформы осуществляется, благодаря шаговому двигателю оси Z 11, который передаёт вращательное движение

резьбовой оси через соединитель вала с двумя затяжными винтами 13. Для фиксации шаговых двигателей на конструкции предусмотрены крышки с отверстиями под крепежные винты 12. Для ограничения планки сканирования есть концевой колпак 8 с выемками под опорные стержни. Вся электроника размещается на разводной печатной плате 18, которая крепится сзади конструкции. К разъёмам платы подключаются электронные модули: модуль Arduino Pro Micro 14, адаптер карт памяти MicroSD 16, драйвер шаговых двигателей 17, а также кнопка сброса 15. К клеммным разъёмам подводятся провода от шаговых двигателей 2, 11, инфракрасного датчика расстояния 4, зарядного устройства.

3.2. Разработка корпуса

Для каркаса было решено разработать переносной корпус с целью упрощения перемещения конструкции, а также для изоляции 3D сканера во время сканирования. Разработанный корпус приведен на рисунке 42.

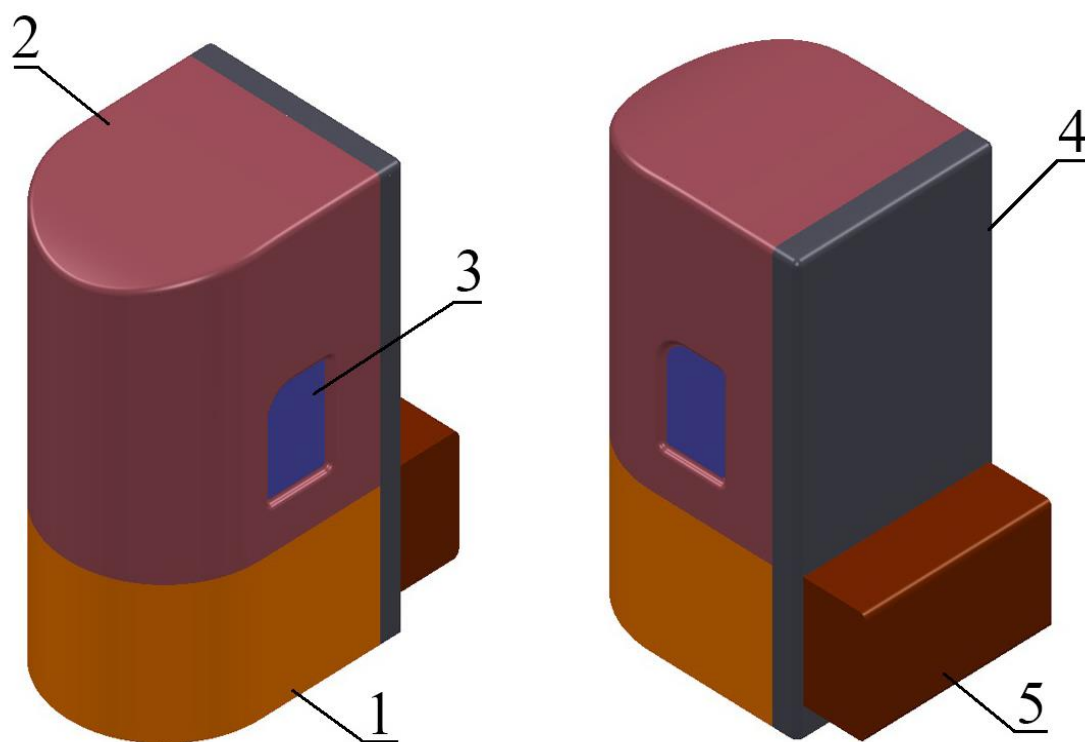


Рисунок 42 – Разработанный корпус в PowerShape

Корпус конструктивно состоит из основания 1, внутри которого будет размещаться устройство, навесной крышки 2 для изоляции рабочего пространства, 3 – окошка для наблюдения за процессом сканирования, 4 – вертикальной крышки для жесткости конструкции, 5 – крышки под электронику.

После непосредственного процесса проектирования и моделирования корпуса устройства, получил файл с 3D моделью, которую в качестве математической модели использовал под написание управляющих программ под обработку на станке с ЧПУ (Числовым Программным Управлением) с помощью специальной CAM(Computer-Aided Manufacturing) программы PowerMill.

Полученные программы в PowerMill экспортировал в корень проекта с разрешением .cut и пропустил через постпроцессор ЧПУ, т.е. модуль, преобразующий созданные траектории в управляющей программы на языке G-code (рисунок 43).

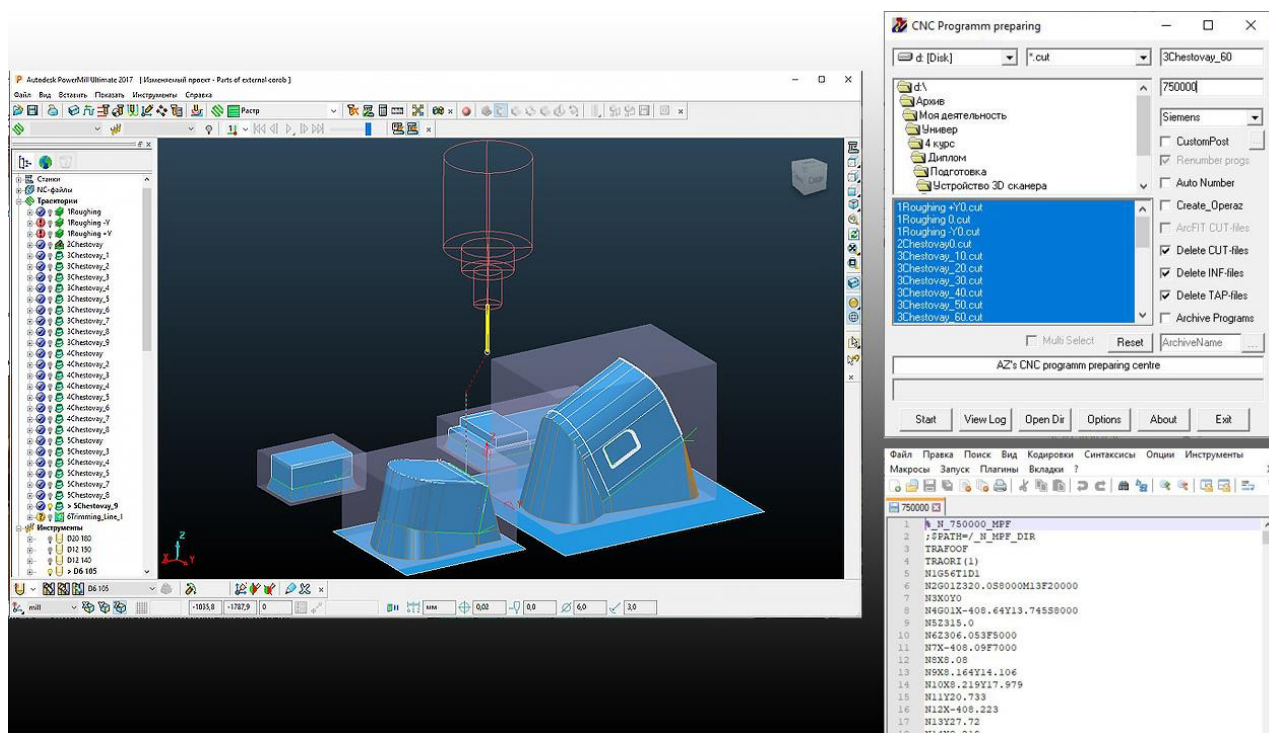


Рисунок 43 – Процесс преобразования траекторий в управляющие программы на языке в G-code

По данным управляющим программам, написанным на языке G-code, будут изготовлены части корпуса на ЧПУ станке. Результат обработки заготовки, составленной из листов MDF (древесноволокнистых плит) выглядит следующим образом (рисунок 44).



Рисунок 44 – Результат изготовления частей корпуса на фрезерном станке с ЧПУ

После обработки на станке с ЧПУ заготовки произвели формовку обработанной заготовки для получения необходимых пластиковых частей корпуса. После их получения собрали корпус и поместили туда сам каркас 3D сканера вместе с предусмотренной для его крепления подложкой. Далее в крышке проделали отверстия под переключатель вкл/выкл и отверстия под разъем питания системы и крепление вентилятора охлаждения самой электроники во время работы 3D сканера. В результате сборки всех механических элементов каркаса, монтажа и размещения электроники, и компоновки всего каркаса во внешний корпус получили следующий результат (рисунок 45).

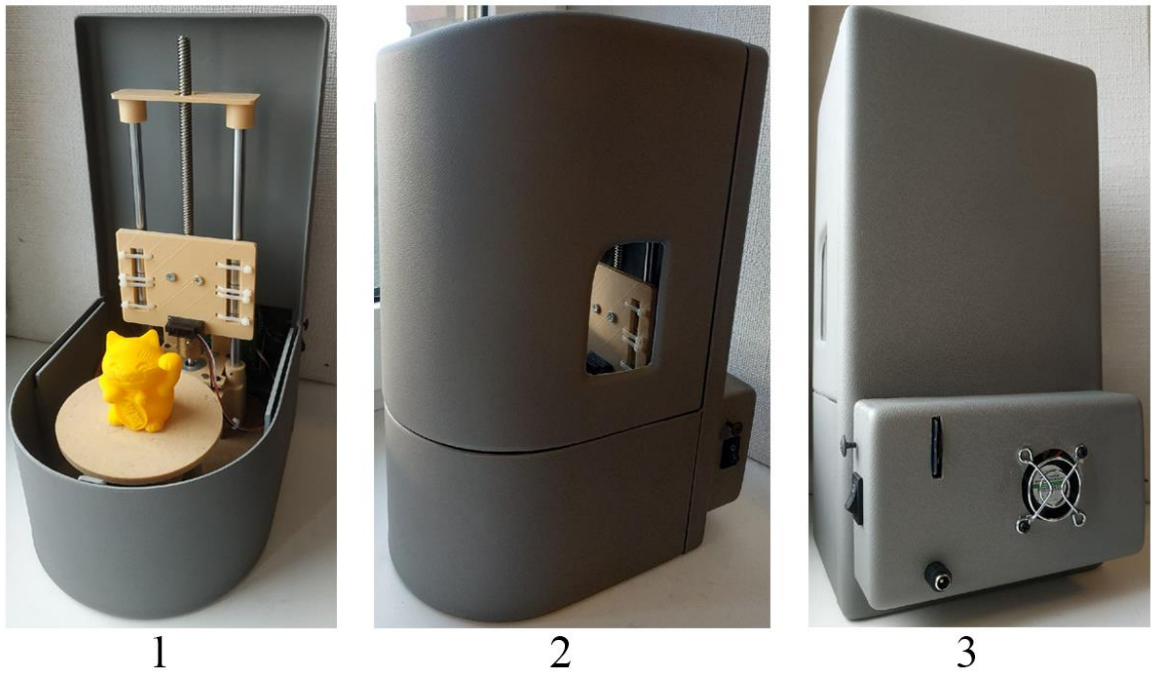


Рисунок 45 – Проект 3D сканера в собранном виде

На рисунке 45 изображен проект 3D сканера в собранном виде в 3 ракурсах. На 1-ом представлен каркас, непосредственно размещенный в корпусе, на 2-ом корпус с одетой крышкой, а также 3-й ракурс с видом на крышку под электронику, в которой выделил места под ключ замыкания вкл/выкл, вентилятор с решеткой для охлаждения системы, а также вырез под SD карту и штекер для питания 3D сканера.

После сборки проекта приступил к написанию кода программы под управление логикой работы устройства 3D сканера.

Выводы по разделу

Таким образом, опираясь на определенную концепцию устройства и с учётом расположения его функциональных компонентов, в частях была разработана конструкция устройства: была спроектирована и реализована несущая часть (каркас), в которой удалось разместить все электрические компоненты, а также переносной корпус под его оснащение.

4. Программная часть

4.1. Разработка кода

Программу управления логикой работы 3D сканера реализовал в виртуальной среде Arduino IDE на языке программирования, основанном на языках C/C++.

Таким образом, в результате разработки кода, были написаны инструкции для цикла работы 3D сканера с возможностью задания параметров сканирования. Например, одним из таких параметров выступает шаг, задаваемый шаговому двигателю для вращения сканирующей платформы по оси Z, т.е. задаётся значение 0,5 мм, которое учитывается при расчёте общего количества шагов, а затем используется в цикле, как элемент условия для вращения шагового двигателя оси Z. При уменьшении данного параметра точность получаемых сканов увеличивается, вследствие уменьшения расстояния между слоями сканирования.

В приложении В представлена программа, полностью отражающая работу 3D сканера с комментариями о её работе. Для наглядного объяснения ключевых этапов работы данной программы была составлена блока схема алгоритма (БСА), приведенная на рисунках 46, а-г.

Первым действием подключаем стороннюю библиотеку SPI для работы с SPI интерфейсом, а также библиотеку SD для осуществления чтения и записи на SD карту. Также определяем изменяемые переменные - параметры сканирования (например, высоту сканирования «int z_axis_height = 9) и константы под выводы драйверов и объявляются расчётные переменные.

Затем вызывается функция «void setup ()», в которой задаются начальные установки работы 3D сканера, такие как скорость обмена данными через последовательный порт, устанавливается тип источника опорного напряжения, инициализируются библиотеки и назначаются режимы работы входов.

Здесь ведется расчёт угла в радианах Rad с учётом углового шага двигателя (1,8 градуса), а также количества шагов p для совершения полного оборота в 360 градусов (формула 4).

$$Rad = \left(\frac{\pi}{180}\right) \cdot \left(\frac{360}{p}\right) = \left(\frac{3,1415926535}{180}\right) \cdot \left(\frac{360}{200}\right) = 0,0314, \quad (4)$$

где $p = 200$ – количество шагов, которое необходимо проделать двигателю поворотного стола, чтобы совершить полный оборот.

Также первично рассчитывается количество шагов по оси Z, которое необходимо сделать, путем деления суммы высоты слоя по оси Z z_layer , общего количества шагов p и количества оборотов для совершения подъема на 1 см $lead_rot$ (формула 5).

$$Steps_Z = \left(\frac{z_layer + p + lead_rot}{10}\right) = \left(\frac{1 + 200 + 10}{10}\right) = 21,1, \quad (5)$$

где $z_layer = 1$ см – заданная высота слоя по оси Z, с которой

осуществляется подъём до заданной высоты;

$p = 200$ - количество шагов для совершения полного оборота;

$lead_rot = 10$ - количество оборотов для совершения подъёма

на 1 см;

Функция «void setup ()» вызывается один раз после запуска устройства.



Рисунок 46а – БСА программы

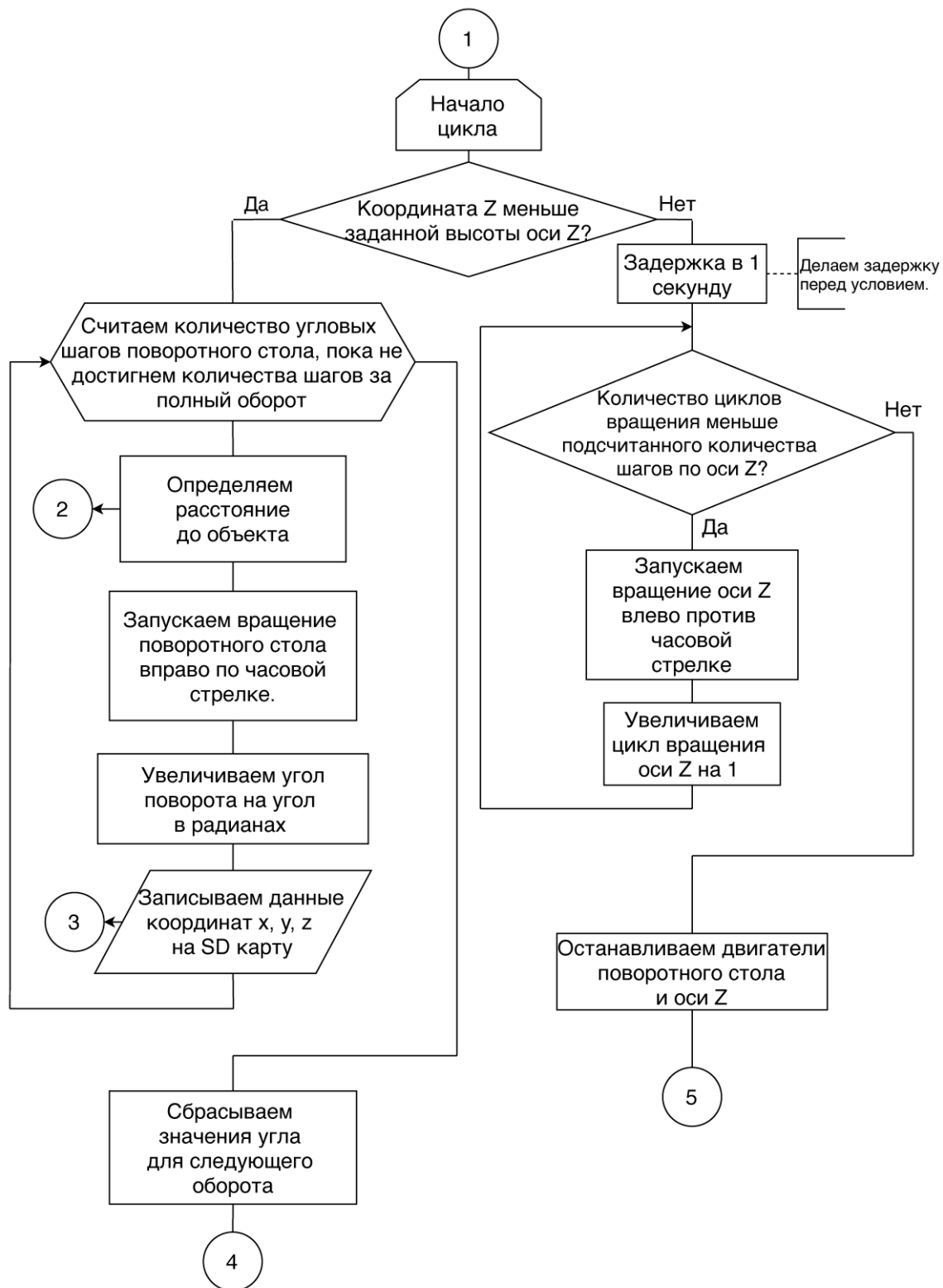


Рисунок 466 – БСА программы

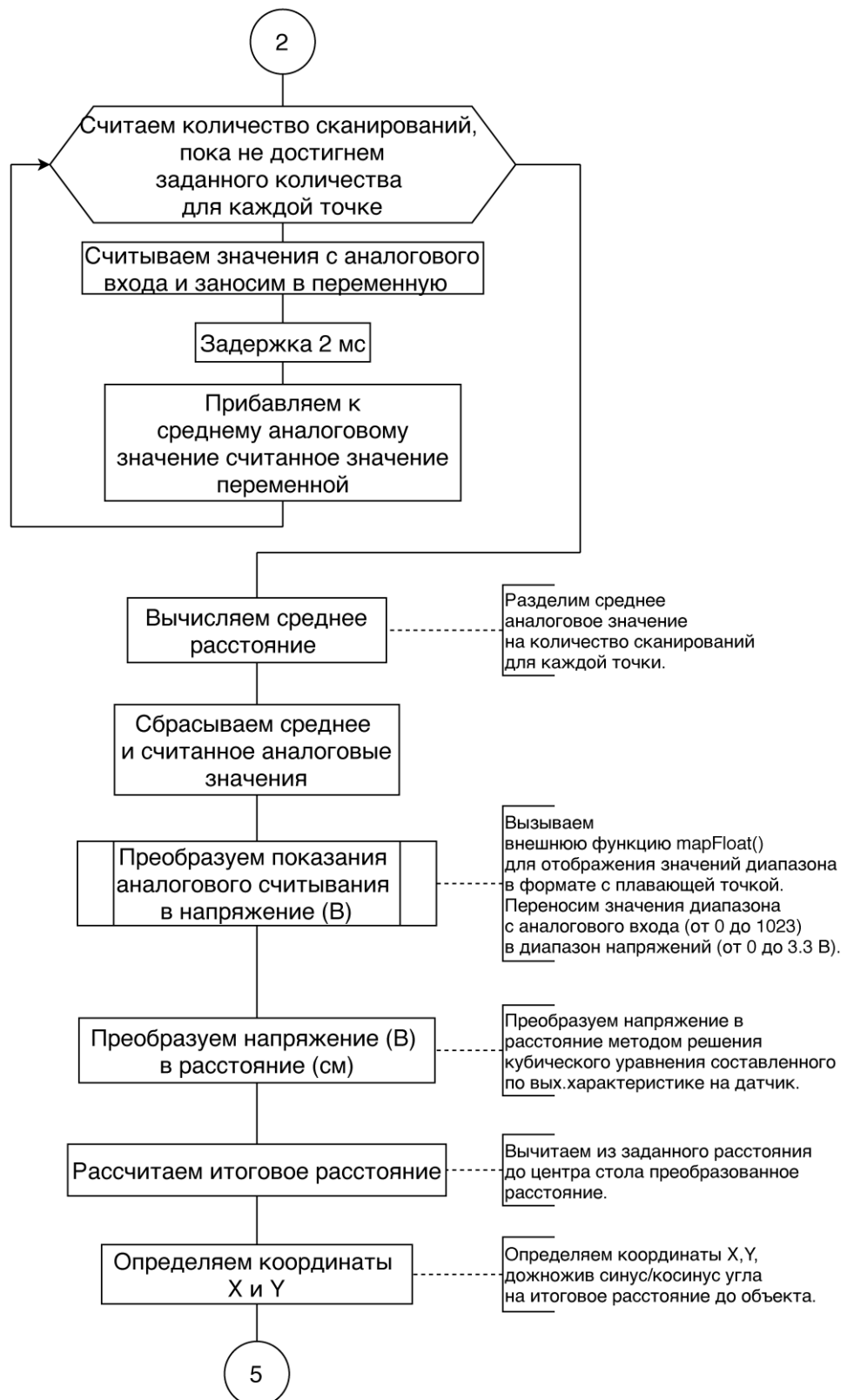


Рисунок 46в – БСА программы

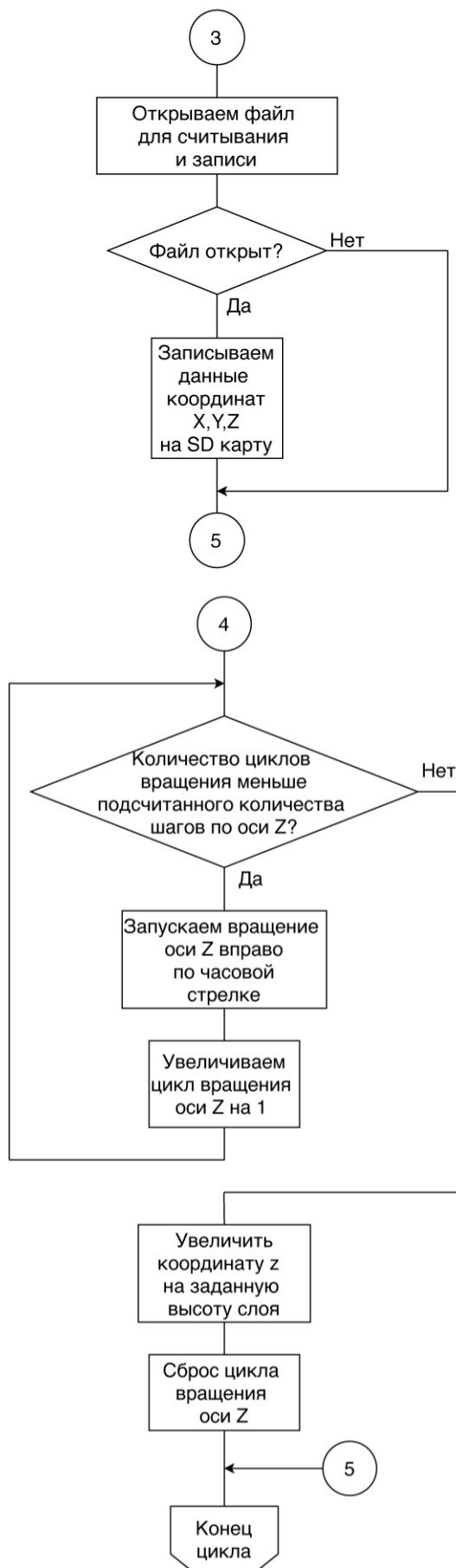


Рисунок 46г – БСА программы

С вызовом функции «void loop()» запускается основной цикл работы 3D сканера. В цикле программы прописывается логика алгоритма, по которой он будет следовать. Здесь прописываются логические операции с переменными (сравнения, условия, счётчики), а также вызываются подпрограммы, выполняющие определенные преобразования и вычисления.

Вследствие того, что основные операции программы работы 3D сканера осуществляются в цикле и они следуют в определённом порядке, то возникает необходимость совершать данный цикл после каждого запуска работы устройства. Поэтому операции данного цикла следует проводить с использованием условного оператора «if», а также операторов цикла «for» для повторения определенных команд внутри заданного цикла, и «while» с выполнением операций по условию.

Таким образом, в начале цикла, когда процесс сканирования запустился, ставим условие: меньше ли текущая координата Z заданной высоты по оси Z. Если результат – истина, то входим в цикл подсчёта количества шагов, пока не будет достигнуто количество угловых шагов за полный оборот. Рассмотрим ситуацию за 1 цикл работы.

Так, согласно алгоритму шаговый двигатель сначала поворачивает поворотный стол с заготовкой на угловой шаг в 1,8 градусов, с которым вращается двигатель в штатном режиме.

После чего задействуется подпрограмма, отвечающая за определение расстояния до объекта `getDistance()`, в которой запускается счётчик количества измерений расстояния до заданной величины, т.е. происходит считывание показаний с аналогового входа датчика до заданной планки. Когда счётчик завершается, определяется среднее значения среди считанных показаний (формула 6).

$$senseDistance = analog/scan_amount, \quad (6)$$

где *analog* – сумма всех показаний аналогового считывания;

scan_amount – количество измерений расстояния.

Это среднее значение преобразуются из диапазона аналогового считывания (0..1023) в диапазон напряжений (0..3.3В) с помощью математической функции `mapFloat()` (формула 7).

$$\text{senseVoltage} = \text{mapFloat}(\text{senseDistance}, 0.0, 1023.0, 0.0, 3.3) = (\text{senseDistance} - 0.0) \cdot (3.3 - 0.0) / (1023.0 - 0.0) + 0.0 \quad (7)$$

где *senseDistance* – показание аналогового считывания.

С целью корректного пересчёта в расстояние преобразованных показаний с датчика, решил определить калибровочное уравнение (уравнение полинома) по выходной характеристике на данной датчик из документации (приложение Б).

Для его определения потребовалось по выходной характеристике, представляющей из себя график зависимости аналогового выходного напряжения от расстояния до отражённого объекта, определить значения напряжения в предусмотренном диапазоне измеряемых расстояний (2-15 см). Полученные значения занёс в таблицу, соотнеся выходное напряжение (V) с расстоянием (см) (таблица 14).

Таблица 14 – Соотношение выходного напряжения (V) с расстоянием (см)

V, В	D, см
2,08	2
1,7	3
1,35	4
1,05	5
0,92	6
0,8	7
0,75	8
0,65	9
0,58	10
0,55	11
0,5	12
0,45	13
0,42	14
0,4	15

Для преобразования значений напряжения, получаемых после оцифровки АЦП поступающих аналоговых значений, и перевода их в расстояние, решил оценить исходную зависимость, путём вывода уравнения полинома, т.е. определить корректирующее уравнение под определение расстояния датчиком. Для этого воспользовался программным пакетом MATLAB со встроенным модулем для вывода графика. После задания значений из таблицы 13 вывел следующий результат (рисунок 47).

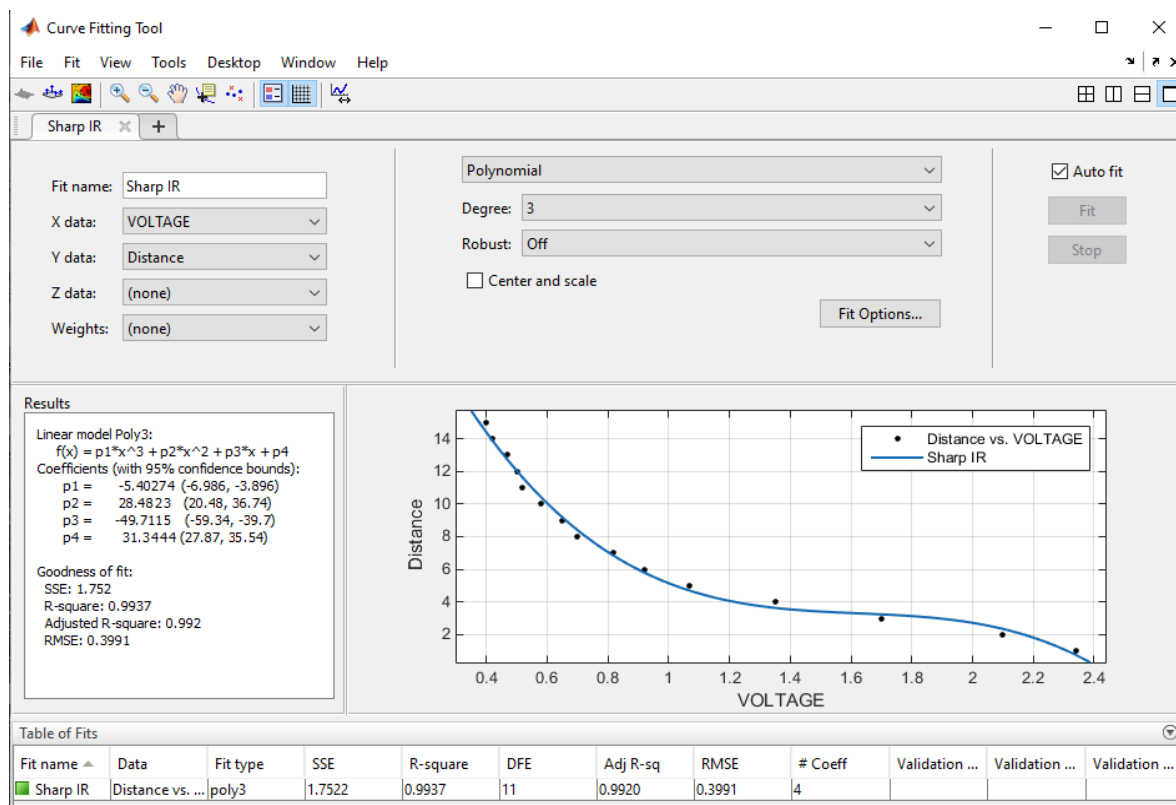


Рисунок 47 – Вывод уравнения полинома для пересчёта напряжения в расстояние

В результате оценки зависимости датчика расстояния было определено уравнение полинома, которое соответствует зависимости расстояния от выходного напряжения (выходной характеристике датчика) (формула 8).

$$f(x) = -5.40274 \cdot x^3 + 28.4823 \cdot x^2 - 49.7115 \cdot x + 31.3444 \quad (8)$$

где $p1 = -5.40274$; $p2 = 28.4823$; $p3 = -49.7115$; $p4 = -49.7115$; $p5 = 31.3444$ – константы уравнения полинома, определенные по зависимости датчика расстояния; x – корень уравнения, в который будут подставляться текущие значение напряжения.

Порядок уравнения полинома (3-ий порядок) был подобран, так как при нём достигается оптимальная точность повторения выходной характеристики.

Полученное уравнение для преобразования поступающего напряжения в расстояние адаптировал в коде программы 3D сканера (формула 9).

$$Distance = -5.40274 * pow(senseVoltage, 3) + 28.4823 * pow(senseVoltage, 2) - 49.7115 * senseVoltage + 31.3444, \quad (9)$$

где *senseVoltage* – показатель напряжения;

pow(senseVoltage, n) – функция возведения значения *senseVoltage* в степень *n*;

Затем находится разница между расстоянием до центра поворотного стола и пересчитанным расстоянием (формула 10), т.е. определяется расстояние от центра стола до точки, с которой сенсор снял показание.

$$d' = distance_to_center - Distance, \quad (10)$$

где *distance_to_center* – расстояние до центра поворотного стола;

Distance – пересчитанное расстояние;

После чего определяются координаты X и Y при текущем угле и расстоянии, определённом на данном угле (формула 11-12).

$$X = \cos(f) \cdot d', \quad (11)$$

$$Y = \sin(f) \cdot d', \quad (12)$$

где *f* – текущий угол поворота;

d – определенное расстояние, при данном угле;

Определив координаты в данной точке, поворачиваем стол с тем же шагом вправо по часовой стрелке, после чего увеличиваем угол поворота на угол в радианах.

Затем происходит запись данных в формате координат X,Y,Z на SD карту, благодаря подпрограмме write_to_SD. В ней сначала генерируется файл, затем происходит его открытие для считывания и записи, и в последующем ставится условие на его открытие. Если результат истина – осуществляем запись, после чего закрываем файл, с проверкой сохранения данных на SD карту.

Графическое пояснение проделанных математических операций проиллюстрировано рисунком 48.

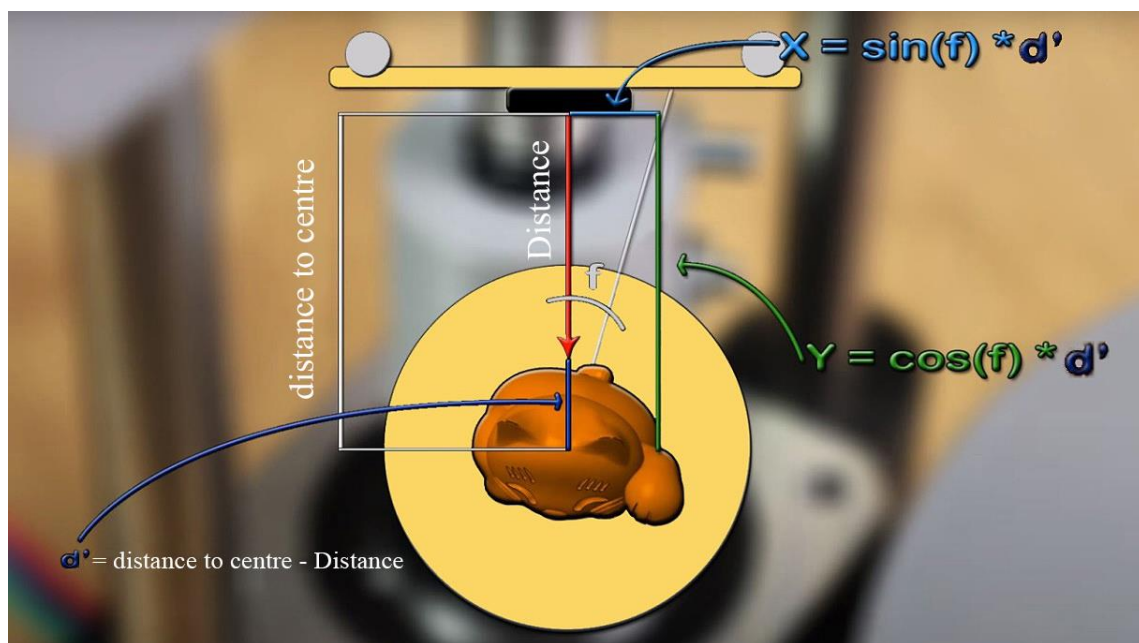


Рисунок 48 – Математическая иллюстрация работы кода

Когда в цикле будет достигнуто количество угловых шагов за полный оборот, значение угла сбрасывается для следующего оборота.

Затем переходим к условию, меньше ли количество циклов вращения подсчитанного количества шагов по оси Z. Если результат истина, то запускаем вращение оси Z вправо по часовой стрелке. После чего увеличиваем цикл вращения на 1 и по новой ставим условие. Когда результат условия – ложь, то увеличиваем координату Z на заданную высоту слоя, после чего сбрасываем цикл вращения оси Z.

В конце работы программы, когда текущая координата Z станет больше заданной высоты по оси Z, делаем задержку, после чего ставим условие, аналогично подъему на заданную высоту. Только в этом случае происходит вращение оси Z влево, на сделанное количество шагов вниз, в исходное положение до непосредственного начала цикла. После этого останавливаем двигатели – процесс сканирования можно считать завершенным.

4.2. Отладка кода под 3D сканер

Так же имеется возможность для собственной модернизации и доработки алгоритмов работы, посредством постановки и введения необходимых параметров в код для управляющего контроллера Arduino. При подключении к компьютеру по micro USB через последовательный интерфейс в среде разработки IDE Arduino можно отслеживать переменные, дающие информацию о текущем состоянии процесса сканирования с помощью функции Serial.print() (рисунок 49).

```
Serial.print(loop_cont);    Serial.print("  ");  
Serial.print(angle);        Serial.print("  ");  
Serial.print(senseDistance); Serial.print("  ");  
Serial.print(x);            Serial.print("  ");  
Serial.print(y);            Serial.print("  ");  
Serial.print(z);            Serial.println("  ");
```

Рисунок 49 – Вывод заданных переменных
в последовательный порт

На рисунке 49 выведены строчки кода, которые выводят заданные переменные в последовательный порт (рисунок 50), где:

- Loop_cont – порядковый номер шага;
- Angle – угол в радианах;
- SenseDistance – расстояние до объекта;
- X,Y,Z – определяемые координаты объекта.

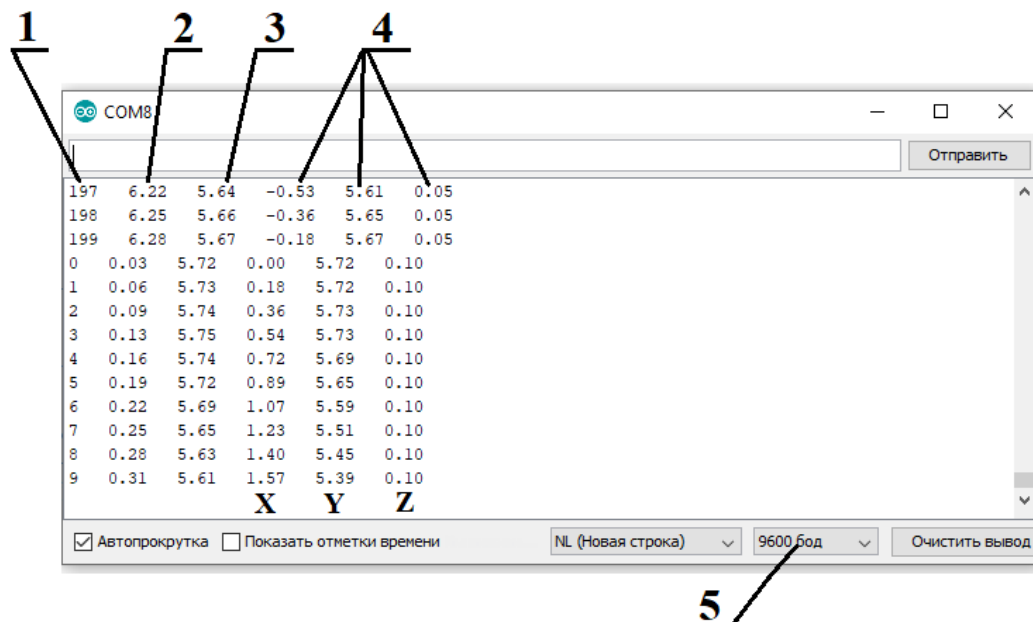


Рисунок 50 – Процесс отслеживания параметров сканирования 3D сканера:

1 - Порядковый номер шага; 2 - Угол в радианах; 3 - Расстояние до объекта;
4 - Координаты объекта X,Y,Z, 5 - Скорость последовательного порта.

При анализе отслеживаемых параметров, обратился к параметрам сканирования (рисунок 51), значения которых можно изменять, чтобы повлиять на работу сканера. Например, чтобы увеличить скорость шагового двигателя поворотного стола следует изменить количество сканирований (измерений) за один угловой шаг.

```
int scan_amount = 50;
String file="scan_001.txt";
int z_axis_height = 9; //in cm
int step_delay = 2000; //in mc
float z_layer_height = 0.1; //in cm
int lead_screw_rotations_per_cm = 10;
int steps_per_rotation_for_motor = 200;
int distance_to_center = 6;
```

Рисунок 51 – Калибровка параметров сканирования 3D сканера

Выводы по разделу

Таким образом, в результате реализации программной части и отладки кода под 3D сканер удалось добиться наиболее удобоваримого качества сканирования, после чего приступил к обработке результатов сканирования – файла с данными в формате координат X, Y, Z.

5. Исследовательская часть

5.1. Обработка результатов сканирования

После достижения необходимого качества сканирования объекта при разрешении 0,5 мм и 1 мм удалось вывести результаты в виде сканов (облака точек) в трехмерное пространство. Вывод полученных данных в виде облака точек осуществил с помощью специального программного обеспечения(Meshlab). Результаты сканирования, выведенные в трехмерное пространство, приведены на рисунке 52.

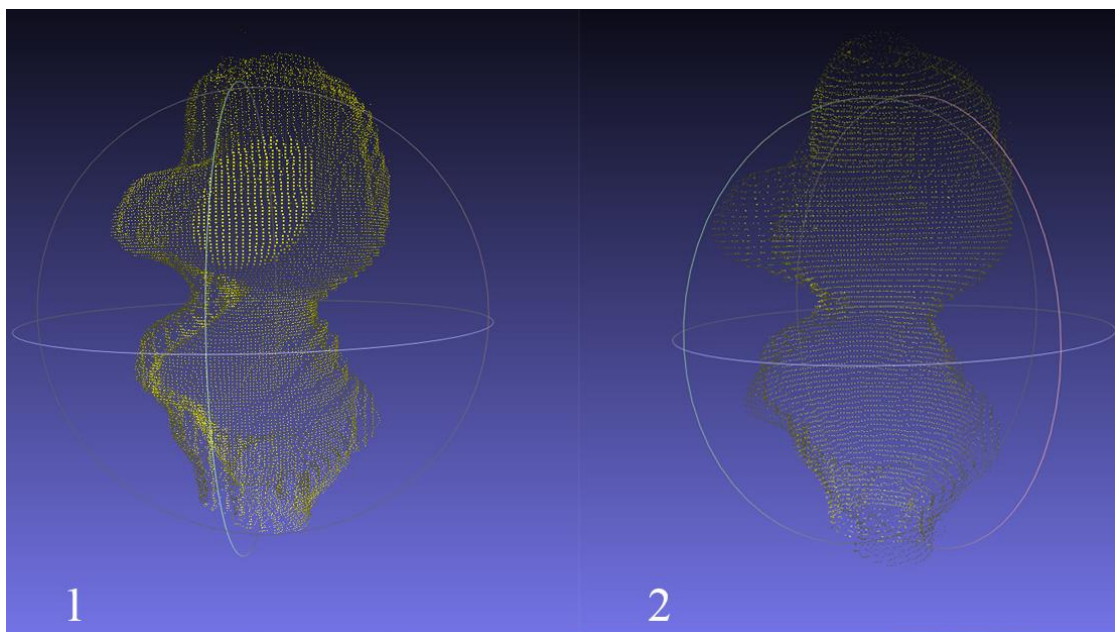


Рисунок 52 – Результаты сканирования объекта в виде облака точек при разрешении: 1) 0,5 мм; 2) 1 мм

Для корректного отображения облака точек, необходимо было вычислить нормали для набора точек, после чего получил корректное отображение облака точек в трехмерном пространстве.

По результатам сканирования (рисунок 52) можно заметить, что кучность облака точек при разрешении 1 мм меньше, чем при 0,5 мм, вследствие меньшего расстояния между точками сканирования, за счёт чего достигается большая степень детализации объекта. Текущее исследование проводилось при одинаковой точности, т.е. при разных разрешениях

осуществлялось 50 измерений на точку, из которых определялось среднее значение для каждой точки.

Исходя из сравнения результатов сканирования, для создания 3D модели решил взять облако точек с большим разрешением.

5.2. Вывод 3D модели

Для создания 3D модели необходимо было реконструировать и придать текстурированное представление для формы облака точек, после чего сформировать полноценную 3D модель. В конечном преобразовании облака точек в 3D модель был получен следующий результат (рисунке 53).

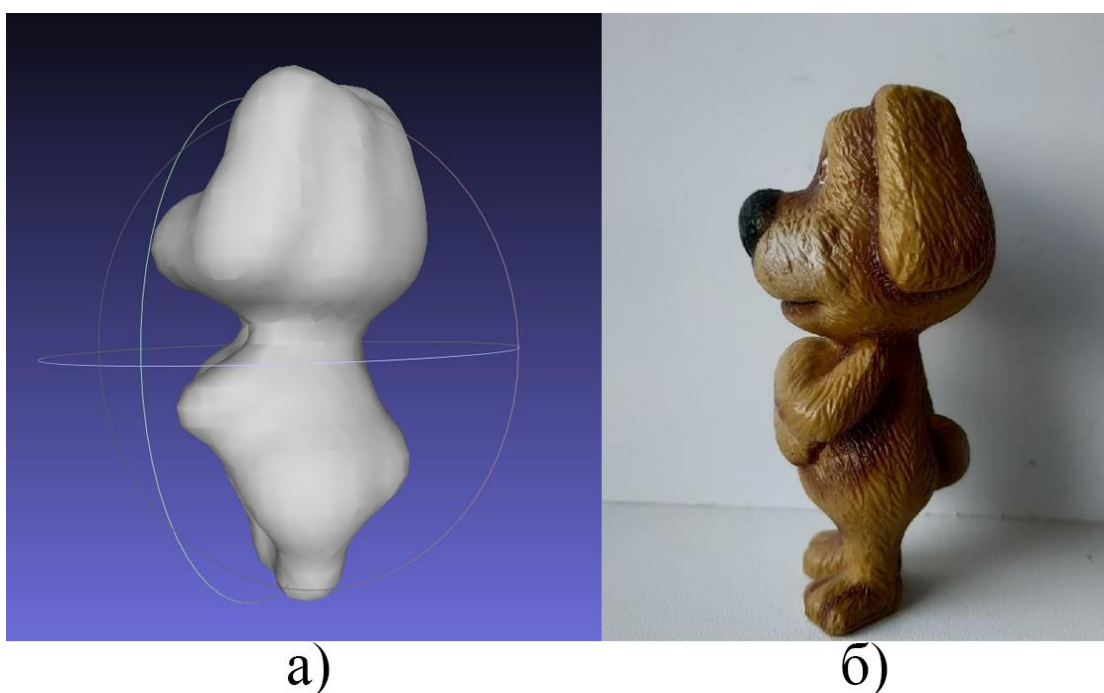


Рисунок 53 – Результат преобразования облака точек в 3D модель:

а) 3D модель; б) Исходная модель (объект сканирования)

По результату видно, что модель представляет из себя примитивное повторение формы объекта сканирования (формы исходной модели), в которой прослеживаются ключевые акценты силуэта, но происходит потеря в детализации мелких частей объекта. Также не учитываются участки с поднутрениями, которые находятся в закрытой зоне от сканирования.

Это говорит о недостатке измерения инфракрасным дальномером нестандартных геометрических форм объекта (изгибов, наконечников), которые требуют более точного сканирующего модуля, что в свою очередь делает вопрос разработки 3D сканера открытым для дальнейших исследований в области трехмерной графики.

Выводы по разделу

Таким образом, в результате исследовательской части был экспериментально исследован объемный объект при разном разрешении (0,5 мм и 1 мм). После обработки и визуализации полученных результатов в среде Meshlab - удалось получить 3D модель объекта, по которой можно в действительности оценить качество сканирования устройством.

Заключение

В данной бакалаврской работе был разработан действующий макет 3D сканера, способного сканировать миниатюрные физические объекты и выводить информацию об их форме в формате трехмерных координат. Выполнив поставленные задачи, была достигнута основная цель работы.

В первой части работы проведен обзор существующих решений, а также основной концепции и применение 3D сканеров в разных инженерных областях. Было исследовано основное устройство 3D сканеров, проведена их классификация, а также рассмотрены технологии 3D сканирования и физические методы на которых они основываются. В результате анализа известных решений была окончательно определена концепция устройства.

Во второй части была создана структурная схема, иллюстрирующая суть соединения компонентов системы устройства. С её учётом был произведен выбор элементной базы, по которой удалось разработать электрическую принципиальную схему устройства. С целью упрощения связи между компонентами схемы, была разработана печатная плата.

В третьей части работы описывается разработка конструкции устройства: несущего каркаса и корпуса, которые сведены в основные этапы её разработки.

В четвертой части приводиться реализация программы управления работой 3D сканера. На данном этапе была написана программа и составлена блок-схема алгоритма, которая графически отражает суть работы программы.

В последней части работы проводились экспериментальные исследования в работе 3D сканера. Были получены результаты сканирования при разрешении 0,5 мм и 1 мм, в котором удалось выяснить, что улучшение детализации происходит при увеличении количества измерений на одну точку, и сокращении длины шага движения по оси Z. А по существу стоит задуматься над аппаратной модернизацией проекта.

Список используемой литературы

- 1.Буянов С.С. Перспективы использования 3D-технологий для развития информационно-аналитической платформы «История современной России» // NB:Исторические исследования. — 2014. - № 6. - С.75-97.
- 2.Прямыцын И.Б., Челпанов И. Б. Лазерные сканеры: распознавание и воспроизведение в 3D-модели мелких подробностей рельефа // Интернет журнал Науковедение - 2012. - № 4.
- 3.Лысыч М. Н., Шабанов М. Л., Жадобкина В. В. Современные системы 3D сканирования // Молодой ученый. - 2014. - №20. - С. 167-171.
- 4.Сделай сам 3D-сканер // КомпьютерПресс № 11'2002.
- 5.Дмитрий Ошкин. To be 3D or not to be... // CADmaster № 40/5.2007
- Хоровиц, Хилл: Искусство схемотехники. М.: Издательство БИНОМ 2014. -704 с.,ил. Берзан В.П., Геликман Б.Ю., Гураевский М.Н. и др.
- 6.Интернет магазин электронных компонентов «Импульс» [Электронный ресурс]. URL: <https://www.impulsi.ru> (дата обращения 15.04.2020).
- 7.Каталог электронных компонентов [Электронный ресурс]. URL: [http:// www.chip-dip.ru](http://www.chip-dip.ru) (дата обращения: 14.05.2020)
- 8.Электроника и наноэлектроника, управление в технических системах, электроэнергетика и электротехника. Выполнение бакалаврской работы / сост. Позднов М.В., Прядилов А.В. - Тольятти: ТГУ, 2020. - 43 с.
- 9.Семёнов Б.Ю. Силовая электроника: от простого к сложному. – 2 изд., испр - М.: СОЛОН-Пресс, 2015. – 416с.: ил.
- 10.ГОСТ 2.701-84. Единая система конструкторской документации.
- 11.Схемы. Виды и типы. Общие требования к выполнению. – Введ. 1985-07-01. – М.: Госстандарт СССР: Изд-во стандартов, 1984.- 11с.: ил.
- 12.ГОСТ 2.710-81 Единая система конструкторской документации (ЕСКД). Обозначения буквенно-цифровые в электрических схемах.

13. Структура и правила оформления. – Введ. 2002-07-01. – М.: Госстандарт РФ: Изд-во стандартов, 2001.- 23с.
14. 3D today.ru – портал о 3D печати [Электронный ресурс] – Режим доступа: www.3dtoday.ru/wiki/15.
15. Александров А.А. Электротехнические чертежи и схемы / Александров К.К., Кузьмина Е.Г.- М.:Энергоатомиздат, 1990. - 288с.
16. Remondino, F.; Menna, F. (2008). "Image-based surface measurement for close-range heritage documentation". The International Archives of the Photogrammetry. Retrieved 28 April 2017.
17. Franca, J. G. D., Gazziro, M. A., Ide, A. N., & Saito, J. H. (2005, September). A 3D scanning system based on laser triangulation and variable field of view. In Image Processing, 2005. ICIP 2005. IEEE International Conference on (Vol. 1, pp. I-425). IEEE.
18. Brian Curless (November 2000). "From Range Scans to 3D Models". ACM SIGGRAPH Computer Graphics. 33 (4): 38–41.
19. Christian Teutsch (2007). Model-based Analysis and Evaluation of Point Sets from Optical 3D Laser Scanners (PhD thesis).
20. P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, G. Ranzuglia(2008). "MeshLab: an Open-Source Mesh Processing Tool". Visual Computing Lab, ISTI - CNR, Pisa Italy.

Перечень элементов

					20-110304.113/09.297.03 ПЭ			
Изм.	Лист	№ докум.	Подп.	Дата				
Разраб.		Волков С.В.				Лит.	Лист	Листов

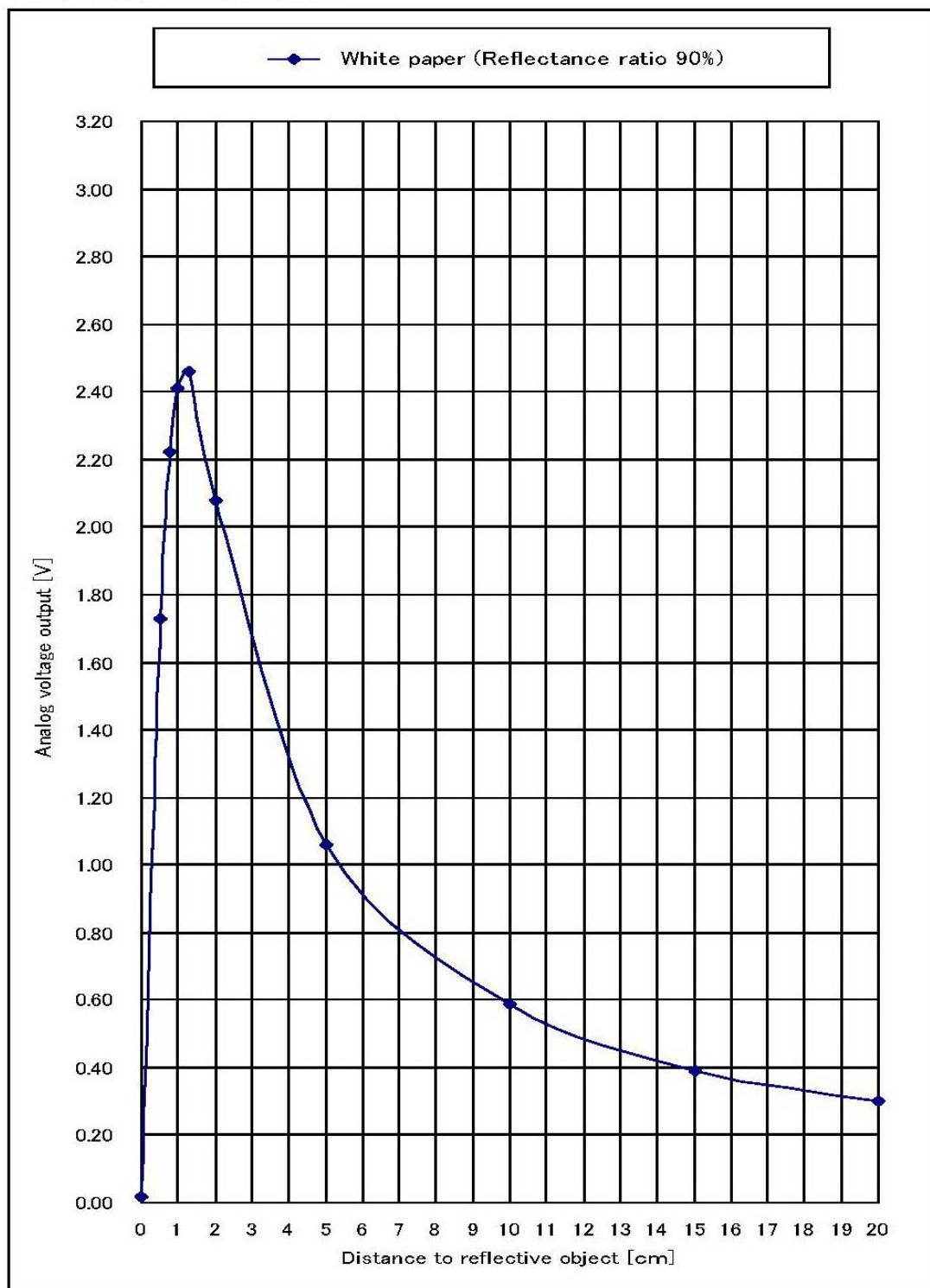
Пров.	Прядилов А.В.			3D сканер Перечень элементов				1	1
					ТГУ, ЭЛБ-1601а				
Н.контр.	Прядилов А.В.								
Утв.	ШЕВЦОВ А.В.								

Приложение Б

Выходная характеристика датчика расстояния

■Supplements

- Example of output distance characteristics



Приложение В

Программа 3D сканера

```
#include <SPI.h> //Подключаем библиотеку SPI.h для работы с SPI интерфейсом
```

```
#include <SD.h> //Подключаем библиотеку SD.h для осуществления чтения и записи на SD карту
```

```
//Изменяемые переменные
```

```
int scan_amount = 50; //Количество сканирований(измерений) для каждой точки. Результатом является среднее значение. Увеличение значения ведёт к задержке для каждого сканирования - приводит к снижению скорости вращения  
String file="scan_001.txt"; //Имя сохраняемого файла на SD карту  
int z_axis_height = 9; //in cm //Максимальная высота сканирования по оси Z  
int step_delay = 2000; //in us //Задержка для каждого шага шагового двигателя в микросекундах  
float z_layer_height = 0.1; //in cm //Высота слоя по оси Z. Принимается в сантиметрах для каждого слоя.  
int lead_screw_rotations_per_cm = 10; //Количество оборотов, которое необходимо сделать ходовому винту, чтобы поднять на 1см.  
int steps_per_rotation_for_motor = 200; //Количество шагов, которое необходимо проделать двигателю поворотного стола, чтобы совершить полный оборот.  
int distance_to_center = 6; //Расстояние от датчика до центра поворотного стола в см.
```

```
//Объявляем константы и присваиваем им значения выводов драйвера управления поворотным столом
```

```
int dir_r = 3;  
int step_r = 2;  
int tMS1=4;  
int tMS2=5;  
int enable_r = 6;
```

```
//Объявляем константы и присваиваем им значения выводов драйвера управления осью Z
```

```
int dir_z = 8;  
int step_z = 7;  
int zMS1=9;  
int zMS2=18;  
int enable_z = 19;
```

```
// Задание основных переменных
```

```
File file_values; //Присвоение классу File переменной file_values. Используется в подпрограмме write_to_SD для SD карты.  
float senseDistance = 0; //Измеренное расстояние  
float angle = 0; //Угол поворота при каждом цикле (0°-360°)  
float x = 0; //Координаты X,Y,Z  
float y = 0; //  
float z = 0; //  
int z_loop = 0; //Переменная, используемая для вращения двигателя оси Z  
int r_loop = 0; //Переменная, используемая для вращения двигателя поворотного стола
```

Продолжение приложения В

```
float measured_analog = 0; //Считанное аналоговое значение с датчика расстояния  
float analog = 0; //Среднее аналоговое значение
```

```

float RADIANS = 0.0;    //Угол в радианах. Рассчитывается в void setup

int steps_z_height = 0; //Переменная, используемая для подсчёта количества шагов по
оси Z

void setup() {
  Serial.begin(9600);    //Задаём скорость обмена данными через последовательный порт
  pinMode(A3, INPUT);    //Устанавливаем аналоговый пин A3 как вход
  analogReference(INTERNAL); //Устанавливаем тип источника опорного напряжения как
INTERNAL(встроенный) - 1.1В ,т.к. микроконтроллер на ATmega32.
  SD.begin(10);          //Инициализируем библиотеку SD и цифровой контакт D10,
подключенный к CS-линии SD-карты

  //Устанавливаем константы оси Z, как выход
  pinMode(dir_z, OUTPUT);
  pinMode(step_z, OUTPUT);
  pinMode(zMS1, OUTPUT);
  pinMode(zMS2, OUTPUT);
  pinMode(enable_z, OUTPUT);

  //Устанавливаем константы поворотного стола, как выход
  pinMode(dir_r, OUTPUT);
  pinMode(step_r, OUTPUT);
  pinMode(tMS1, OUTPUT);
  pinMode(tMS2, OUTPUT);
  pinMode(enable_r, OUTPUT);

  digitalWrite(enable_z,HIGH); //Отключаем драйвер управления осью Z
  digitalWrite(enable_r,HIGH); //Отключаем драйвер управления поворотным столом

  //Отключаем микрошаговый режим на двигателе поворотного стола
  digitalWrite(tMS1,LOW);
  digitalWrite(tMS2,LOW);

  //Отключаем микрошаговый режим на двигателе оси Z
  digitalWrite(zMS1,LOW);
  digitalWrite(zMS2,LOW);

  //Рассчитываем переменные угла в радианах(RADIANS) и количество шагов по оси Z
  RADIANS = (3.141592 / 180) * (360/steps_per_rotation_for_motor);
  steps_z_height = (z_layer_height * steps_per_rotation_for_motor *
lead_screw_rotations_per_cm)/10;

}

```

Продолжение приложения В

```

void loop() {

```

```

//Когда процесс сканирования запустился ставим условие процесса, когда будет
достигнута максимальная высота
if(z < z_axis_height)
{
    for(int loop_cont = 0 ; loop_cont < steps_per_rotation_for_motor; loop_cont++)
//Устанавливаем счётчик подсчёта количества шагов, пока не будет достигнуто
количество шагов за полный оборот
    {
        getDistance(); //Определяем расстояния до объекта
        digitalWrite(enable_r,LOW); //Включаем драйвер поворотного стола
        digitalWrite(dir_r,HIGH); //Заставляем вращаться стол вправо по часовой стрелке
        digitalWrite(step_r,HIGH); //Задаём шаг вращения стола
        delayMicroseconds(step_delay); //Вводим задержку между шагами
        digitalWrite(step_r,LOW);
        delayMicroseconds(step_delay);
        angle = angle + RADIAN; //Увеличиваем угол поворота на угол в радианах
        write_to_SD(x,y,z); //Записываем данные координат x, y, z на SD карту

//Выводим заданные переменные в последовательный порт для мониторинга и калибровки
        Serial.print(loop_cont); Serial.print(" ");
        Serial.print(angle); Serial.print(" ");
        Serial.print(senseDistance); Serial.print(" ");
        Serial.print(x); Serial.print(" ");
        Serial.print(y); Serial.print(" ");
        Serial.print(z); Serial.println(" ");
    }
    angle = 0; //Сбрасываем значение угла для следующего оборота

//Во время того, пока цикл вращения меньше подсчитанного количества шагов по оси Z
    while(z_loop < steps_z_height)
    {
        digitalWrite(enable_z,LOW); //Включаем драйвер оси Z
        digitalWrite(dir_z,LOW); //Заставляем вращаться ось Z вправо
        digitalWrite(step_z,HIGH); //Задаём шаг вращения оси Z
        delayMicroseconds(step_delay); //Вводим задержку между шагами
        digitalWrite(step_z,LOW);
        delayMicroseconds(step_delay);
        z_loop = z_loop+1; //Увеличиваем цикл вращения оси Z на 1
    }
    z = z + z_layer_height; //Увеличить координату z на высоту слоя
    z_loop = 0; //Сброс значения переменной вращения оси Z

} //Конец условия if, когда z > z_axis_height

else
{
    delay(1000);

```

Продолжение приложения В

```

while(z_loop < steps_z_height)
{

```



```

digitalWrite(enable_z,LOW);    //Включаем драйвер оси Z
digitalWrite(dir_z,HIGH);      //Заставляем вращаться ось Z влево
digitalWrite(step_z,HIGH);     //Задаём шаг вращения оси Z
delayMicroseconds(step_delay); //Вводим задержку между шагами
digitalWrite(step_z,LOW);
delayMicroseconds(step_delay);
z_loop = z_loop+1;
}
//Останавливаем драйверы
digitalWrite(enable_r,HIGH); //Отключаем драйвер управления поворотным столом
digitalWrite(enable_z,HIGH); //Отключаем драйвер управления осью Z

// Когда процесс сканирования закончен, то останавливаем драйверы

} //Конец цикла подпрограммы void loop()

//Вводим Функцию, которая получает показания расстояния с датчика расстояния в
действительной форме
double getDistance()
{
    for (int aa=0; aa < scan_amount; aa++) //Организуем счётчик подсчёта количества
сканирований для каждой точки. Результатом является среднее значение.
    {
        measured_analog = analogRead(A3); //Считываем значения с аналогового пина A3 и
заносим в переменную measured_analog
        delay(2);
        analog = analog + measured_analog; //Увеличиваем среднее аналоговое значение на
считанное значение переменной measured_analog
    }
    senseDistance = analog/scan_amount; //Вычисляем среднее расстояние. Разделим среднее
аналоговое значение на количество сканирований для каждой точки.
    analog=0; //Сбрасываем среднее аналоговое значение
    measured_analog = 0; //Сбрасываем считанное аналоговое значение
    senseDistance = mapFloat(senseDistance,0.0,1023.0,0.0,3.3); //Преобразуем показания
аналогового считывания в напряжение
    senseDistance = -5.40274*pow(senseDistance,3)+28.4823*pow(senseDistance,2)-
49.7115*senseDistance+31.3444; //Преобразуем напряжение в расстояние методом
решения кубического уравнения, составленного по выходной характеристике на датчик
    senseDistance = distance_to_center - senseDistance; //Рассчитываем итоговое расстояние =
расстояние от датчика до центра поворотного стола - измеренное расстояние
    y = (cos(angle) * senseDistance); //Определяем координаты Y, дожножив косинус угла
на итоговое расстояние до объекта
    x = (sin(angle) * senseDistance); //Определяем координаты X, дожножив синус угла на
итоговое расстояние до объекта
}

```

Продолжение приложения В

//Функция map(), которая отображает значения диапазона в формате с плавающей точкой

```

float mapFloat(float fval, float val_in_min, float val_in_max, float val_out_min, float
val_out_max)
{
    return (fval - val_in_min) * (val_out_max - val_out_min) / (val_in_max - val_in_min) +
    val_out_min; //Переносим значения диапазона с аналогового входа (от 0 до 1023) в
диапазон напряжений (от 0 до 3.3 В)
}

//Функция записи данных координат X,Y,Z на SD карту
void write_to_SD(float SDx, float SDy, float SDz)
{
    file_values = SD.open(file, FILE_WRITE); //Открываем файл для считывания и записи

//Если файл открыт, записываем данные координат X,Y,Z на SD карту, после чего
закрываем файл, с проверкой сохранения данных на SD-карту.

    if (file_values)
    {
        file_values.print(x); file_values.print(",");
        file_values.print(y); file_values.print(",");
        file_values.println(z);
        file_values.close();
    }
}

```