

Московская Командная Олимпиада по Робототехнике
Регламент «Юный Инженер»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к учебному проекту по разработке
колесного робота

Выполнил:
XXXXXX Роберт, 7 класс

Москва
2024

СОДЕРЖАНИЕ

1 Цель 2 Механическая часть

2.1 Проектирование

2.1.1 Корпус робота

2.1.2 Колёса

2.1.2.1 Диски

2.1.2.2 Шины

2.1.2.3 Передняя опорас

2.1.3 Крепления под электронику

2.1.3.1 Крепление для управляющей платы

2.1.3.2 Крепление для Датчиков

2.1.3.3 Крепление для Моторов

2.1.3.4 Крепление для тумблера

2.1.4 Механизм сбивания горизонтальной планки

2.2 Изготовление

2.2.1 Корпус робота

2.2.2 Колёса

2.2.2.1 Диски

2.2.2.2 Шины

2.2.3 Крепления под электронику

2.2.3.1 Крепление для управляющей платы

2.2.3.2 Крепление для Датчиков

2.2.3.3 Крепление для Моторов

2.2.3.4 Крепление для тумблера

2.2.4 Механизм сбивания горизонтальной планки

3 Электронная часть

3.1 Проектирование

3.1.1 Управляющая плата

3.1.2 Датчики

3.2 Изготовление

3.2.1 Выбор оптимального решения

3.2.2 Управляющая плата

3.2.3 Датчики

4 Алгоритмическая часть

5 Заключение

6 Приложения

1 Цель

1.1 Выполнить учебный проект. Изучить программирование, электронику, моделирование, сделать документацию и изготовить все составные части робота, способного проехать по линии, сбив одну из двух горизонтальных планок, находящихся на высоте 200 мм.

2 Механическая часть

2.1 Проектирование

Для проектирования деталей была использована Система Автоматизированного Проектирования (далее САПР) «КОМПАС-3D», по той причине, что он распространяется свободно и удобен в освоении.

2.1.1 Корпус робота должен быть достаточно универсален, так как предусматривается совместимость с разными версиями элементов, закрепляемых к нему. Из основных узлов, крепящихся к корпусу, можно выделить: батарейный отсек, управляющая плата, тумблер питания, датчик(и) линии, моторы, механизм сбивания планки. Трёхмерная модель финальной версии корпуса робота приведена на рисунке 1.1

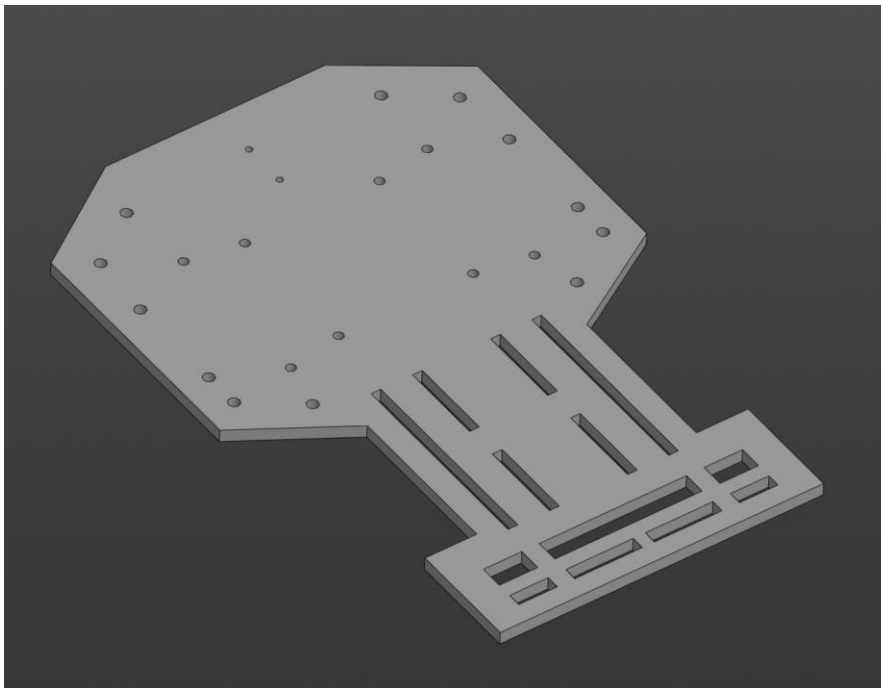


Рисунок 2.1 — финальная версия корпуса робота в окне САПР

2.1.2 Колёса должны жёстко и надёжно держаться на валу мотора и обладать хорошим сцеплением.

2.1.2.1 Колёсные диски имеют внешнюю канавку, для исключения проскальзывания шины. Финальная версия колёсных дисков - на рисунке 2.2.

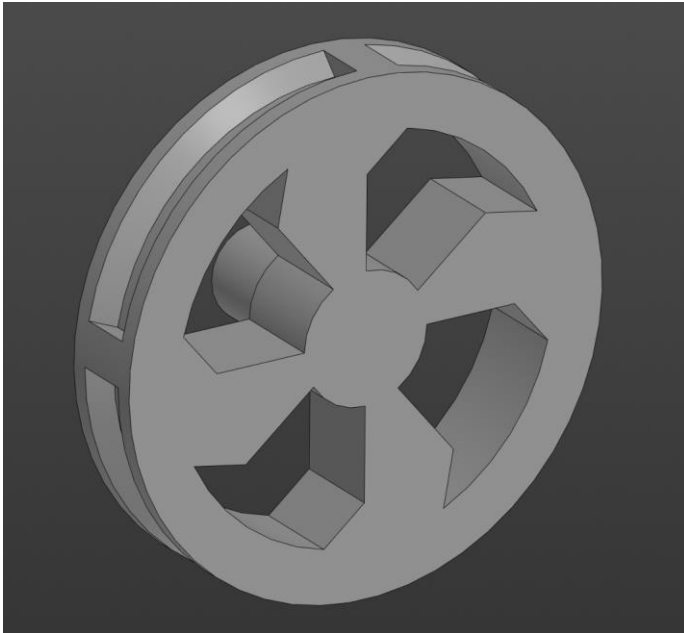


Рисунок 2.2 — финальная версия колёсных дисков в окне САПР

2.1.2.2 На шинах предусмотрена внутренняя канавка, соответная той, что на диске. Шины изготавливаются методом литья из силикона. Окончательную версию одноразовой формы для заливки силикона смотри на рисунке 2.3.

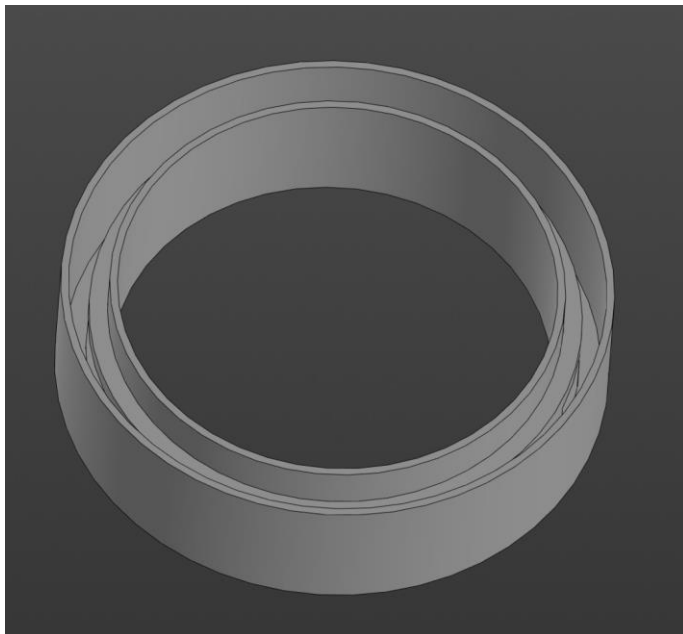


Рисунок 2.3 — финальная версия колёсных формы для заливки силикона в окне САПР

2.1.3 Крепления под электронику должны обеспечивать надёжное крепление к корпусу робота электроники (управляющая плата, датчики, моторы).

2.1.3.1 Управляющая плата крепится к корпусу робота стойками для печатных плат при помощи болтов М3.

2.1.3.2 Датчики линии крепятся к передней части корпуса робота через прямоугольные отверстия при помощи болтов и гаек М3.

2.1.3.3 Моторы крепятся к корпусу робота при помощи креплений и болтов и гаек М3. Крепление представляет из себя скобу с двумя отверстиями под болты М3. Финальная версия крепления моторов приведена на рисунке 2.4.

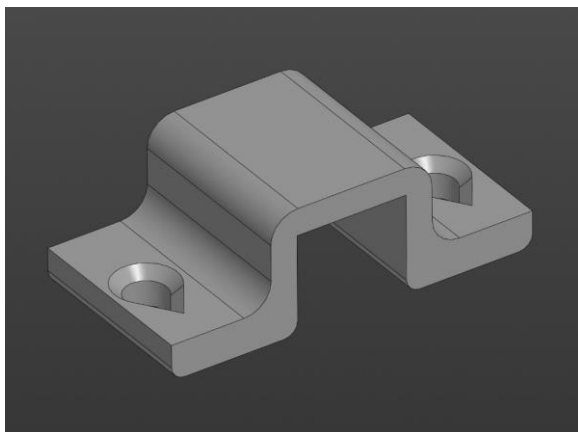


Рисунок 2.4 — финальная версия крепления моторов в окне САПР

2.1.4 Для устройства сбивания горизонтальной планки был выбран параллелограммный механизм. В качестве связующего элемента была избрана канцелярская скрепка, так как это доступный и удобный в обработке материал.

2.2 Изготовление

2.2.1 Корпус робота был изготовлен методом лазерной резки из фанеры толщиной 4 мм (Рисунок 2.5) (Приложение 2). Предусмотренные технологические отверстия под крепеж, в целом, справились со своей задачей.

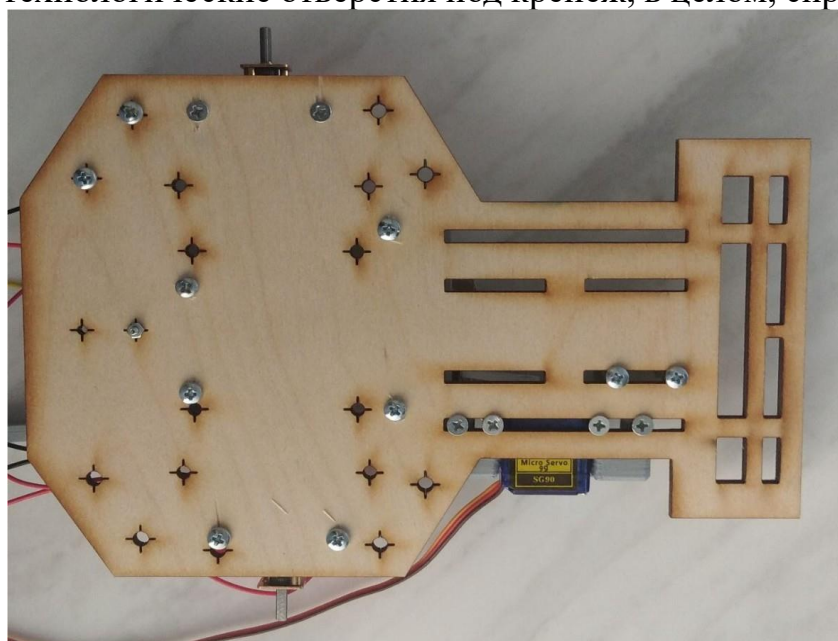


Рисунок 2.5 — корпус робота

2.2.2 Колёса имеют внешний диаметр $\varnothing 57.1$ мм, состоят из отдельных дисков и шин.

2.2.2.1 Колёсные диски были изготовлены методом FFF (далее 3D печать) пластиком PLA (рисунок 2.6)

(Приложение 3).

Первый изготовленный комплект дисков имел неточность — отверстие для оси мотора было немного больше необходимого. Второй комплект был напечатан с учетом усадки пластика и оказался удачным.

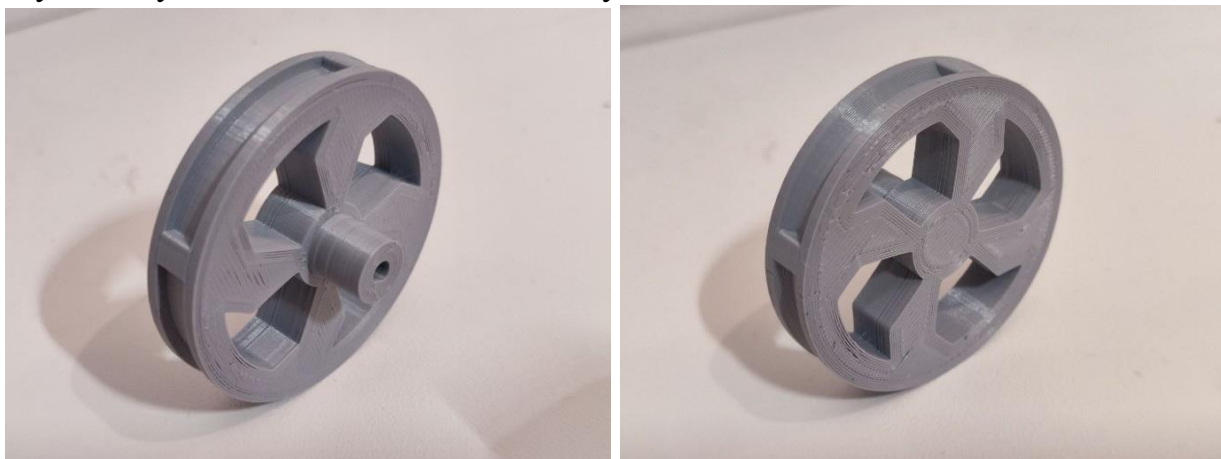


Рисунок 2.6 — колёсный диск с внешней и внутренней стороны

2.2.2.2 Шины изготовлены литьем силикона в формы, изготовленные методом 3D печати пластиком PLA.

Для изготовления шин, были испробованы три вида силикона:

- Санитарный силиконовый герметик Litokol SA цвета антрацит.
- Санитарный силиконовый герметик нейтральный Master Teks ProfiMaster Белый.
- Двухкомпонентный литьевой силикон Polypark SilSn на основе олова.

В результате эксперимента санитарный силиконовый герметик показал себя в целом неплохо, однако Litokol SA оказался излишне эластичным и с ярко выраженным запахом. Master Teks ProfiMaster Белый – достаточно плотный, практически без запаха. При этом, в обоих случаях главная проблема оказалась в заполняемости формы.

Как видно на Рисунке 2.7, в форме остались незаполненные полости, хотя при заливке принимались меры по уплотнению массы.



Рисунок 2.7 — Шины из санитарного силиконового герметика: слева Litokol SA цвета антрацит, справа - Master Teks ProfiMaster Белый

В разных источниках упоминаются способы разжижения силиконовых герметиков разбавлением бензином, силиконовым маслом, и прочими веществами, но было принято решение обратиться к специальным литьевым составам. PolyPark SilSn оказался относительно доступен по стоимости и срокам доставки.

В процессе литья опытным путем было установлено оптимальное соотношение отвердителя к силикону. При избыточном количестве отвердителя, быстрое сгущение силикона препятствовало однородной заливке. Состав отвердителя оказался не бесцветным, что помогло на уровне цветового оттенка определить нужную пропорцию компонентов (Рисунок 2.8).



(Рисунок 2.8). – пробные фрагменты заливки с разной пропорцией компонентов

Более темный образец (слева на Рисунке 2.8) затвердел менее чем за час, тогда как более светлому варианту (справа) на это понадобилось более 10 часов.

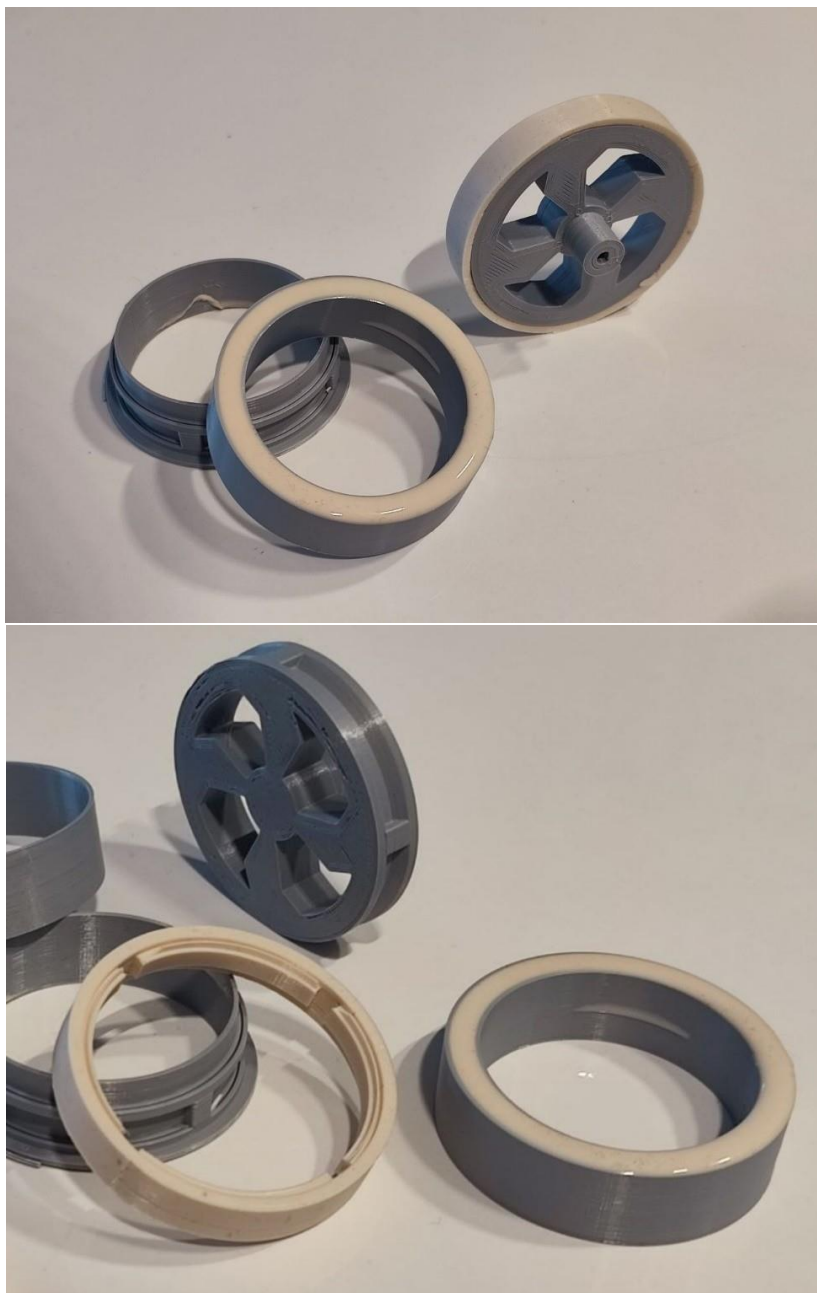


Рисунок 2.9 — Шина из двухкомпонентного силикона в форме и на колесном диске

Рисунок 2.10 — Шина из двухкомпонентного силикона в форме, извлеченная из формы и колесный диск

2.2.3 Крепления под электронику

2.2.3.1 Управляющая плата крепится к корпусу робота стойками для печатных плат при помощи болтов М3. Крепление для управляющей платы

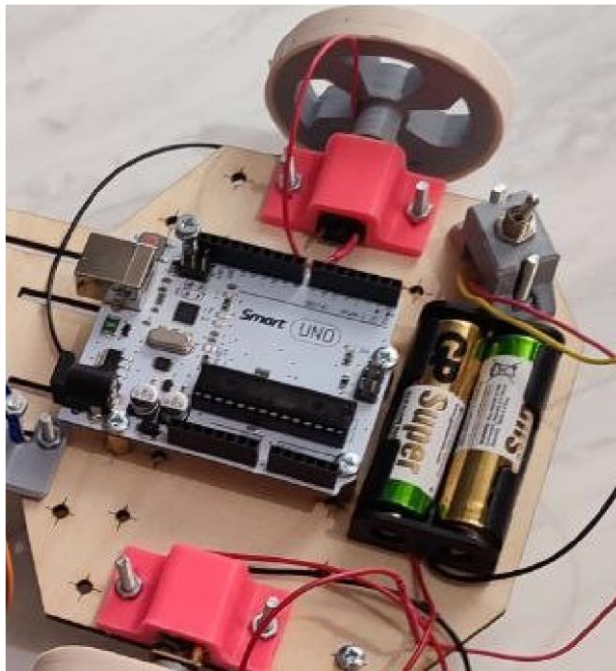


Рисунок 2.11 — Крепление для управляющей платы

2.2.3.2 Плата датчиков смонтирована на корпусе через болты М3. Крепление для Датчиков

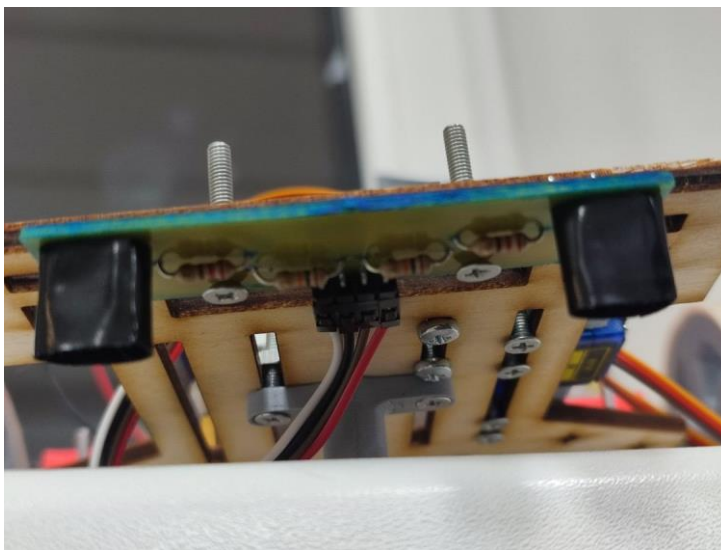


Рисунок 2.12 — Крепление для Датчиков

2.2.3.3 Скоба крепления мотора исполнена методом 3D печати пластиком PLA и смонтирована на болтах М3.

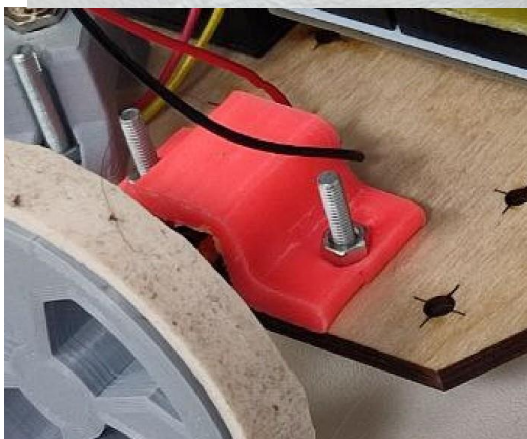


Рисунок 2.13 — Крепление для Моторов

2.2.3.4 Скоба крепления тумблера исполнена методом 3D печати пластиком PLA и смонтирована на болтах М3.



Рисунок 2.14 — Крепление для тумблера

2.2.4 Механизм сбивания горизонтальной планки.

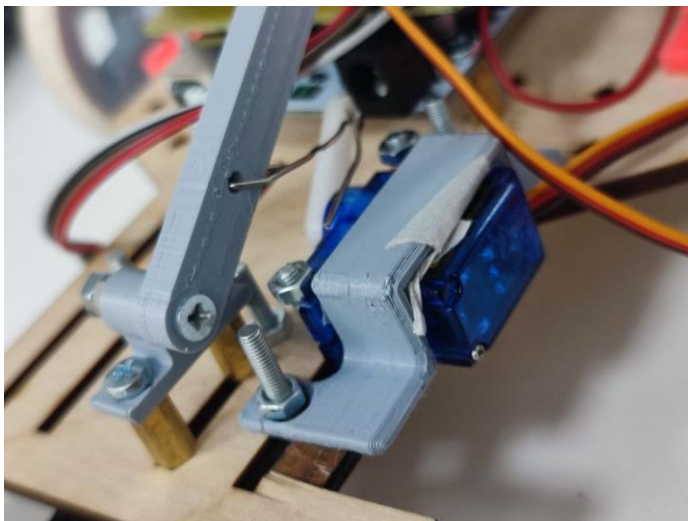


Рисунок 2.15 — Механизм сбивания горизонтальной планки

Состоит из сервопривода, смонтированного на корпусе скобой (3D печать). Привод соединен с рабочей частью через металлическую скрепку.

Вся конструкция смонтирована на корпусе болтами М3.

3 Электронная часть

3.1 Проектирование

Для проектирования, проверки и трассировки печатных плат была использована свободно распространяемая САПР «KiCAD».

3.1.1 Управляющая плата

Изначально предполагалось, что драйвер двигателей будет находиться на плате, установленной на корпусе отдельно от управляющего микроконтроллера (рисунок 3.1). Позже было решено выполнить драйвер на управляющей плате, в формате дополнительного модуля для платы Arduino Uno (аналог) (рисунок 3.2), так как такой формат не требует дополнительных креплений и проводов. На управляющей плате расположены:

- а) двухканальный драйвер моторов на n-канальных транзисторах типа MOSFET
- б) тактовая кнопка (предназначенная для отладки и запуска)
- в) индикационный светодиод
- г) контакты под:
 - 1) сервопривод
 - 2) датчики линии
 - 3) блок аккумуляторных батарей.

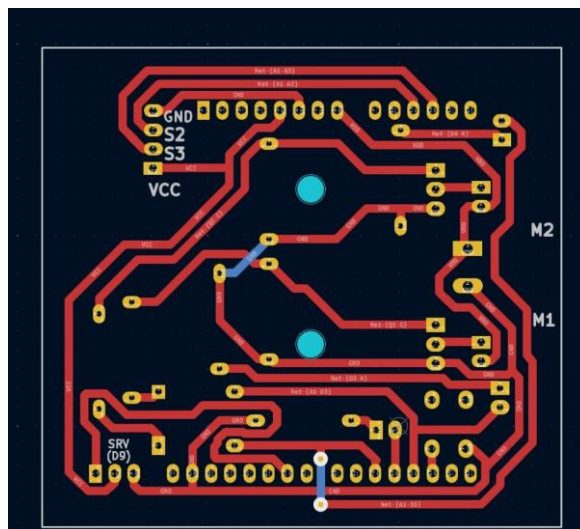
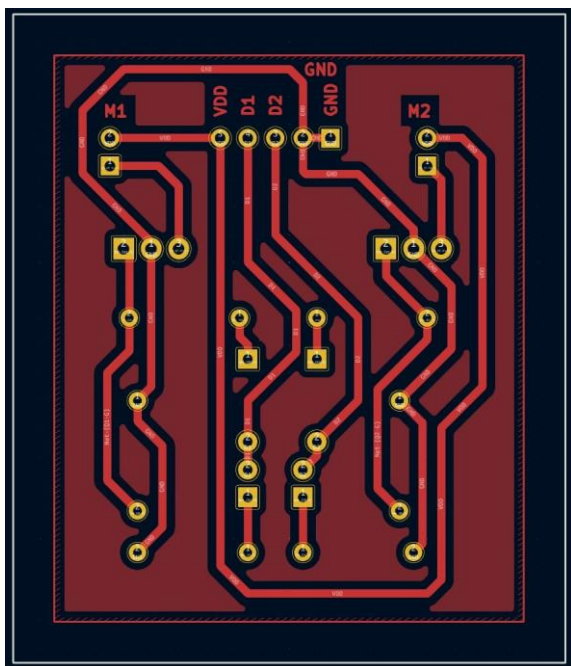


Рисунок 3.1 — первая версия в окне САПР Рисунок 3.2 — финальная версия драйвера управляющей платы с драйвером

3.1.2 Датчики линии

Изначально предполагалось, что датчики линии будут выполнены на отдельных платах (рисунок 3.3). Позже было решено выполнить 2 датчика линии на одной плате, так как такой формат обеспечивает правильное расстояние между датчиками (рисунок 3.4). Датчик представляет из себя оптопару, составленную из светодиода красного спектра и фоторезистора.

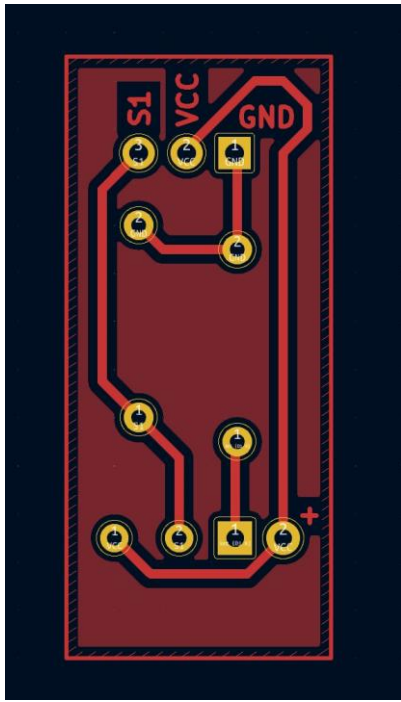


Рисунок 3.3 — первая версия платы датчика
линии в окне САПР

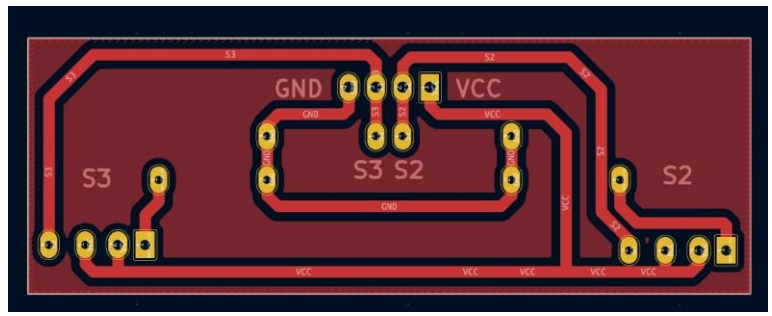


Рисунок 3.4 — финальная версия платы датчиков
линии в окне САПР

3.2 Изготовление

3.2.1 Для изготовления печатных плат рассматривались несколько технологий изготовления:

- а) Лазерно-Утюжная Технология (далее ЛУТ)
- б) метод фоторезиста (далее Фоторезист)
- в) метод фрезеровки плат.

Сравнение данных вариантов приведены в таблице 3.

Таблица 3 — сравнение методов изготовления печатных плат

| Метод | Плюсы | Минусы |
|------------|--|--|
| ЛУТ | Простота и доступность необходимых материалов | Невысокая точность, сложно предсказуемый результат, слабая повторяемость, человеческий фактор. |
| Фоторезист | Простота и возможность получить относительно высокое качество. | Много этапов, сложно предсказуемый результат, слабая повторяемость, человеческий фактор. |

| | | |
|------------------|--|---|
| Метод фрезеровки | Высокое качество, точность, скорость, повторяемость. | Необходимость наличия станка, требуется дополнительная подготовка файлов для ЧПУ. |
|------------------|--|---|

Изначально был выбран метод фоторезиста. В общей сложности этот процесс предусматривает около восьми этапов, каждый из которых сложно формализовать и приходится действовать опираясь на субъективные факторы:

- У разных производителей фоторезистивной пленки разные параметры ее экспонирования, разная степень реакции на промывочную жидкость, степень адгезии и т. д.
- На этапе прогрева пленки необходимо по наитию выбирать степень, время и температуру воздействия на пленку.
- Разные тонеры дают разную степень перекрытия пленки, а разные средства оптического уплотнения тонера также по-разному воздействуют с разными тонерами. и т. д.

Метод фоторезиста был испробован на практике, в какой-то степени можно назвать опыт успешным (рисунок 3.4)



Рисунок 3.4 — Метод фоторезист: лазерная печать макета на прозрачной пленке



Рисунок 3.5 — Метод фоторезист: перенос фоторезистивной пленки и шаблона на стеклотекстолит



Рисунок 3.6 — Метод фоторезист: экспонирование композита в ультрафиолете

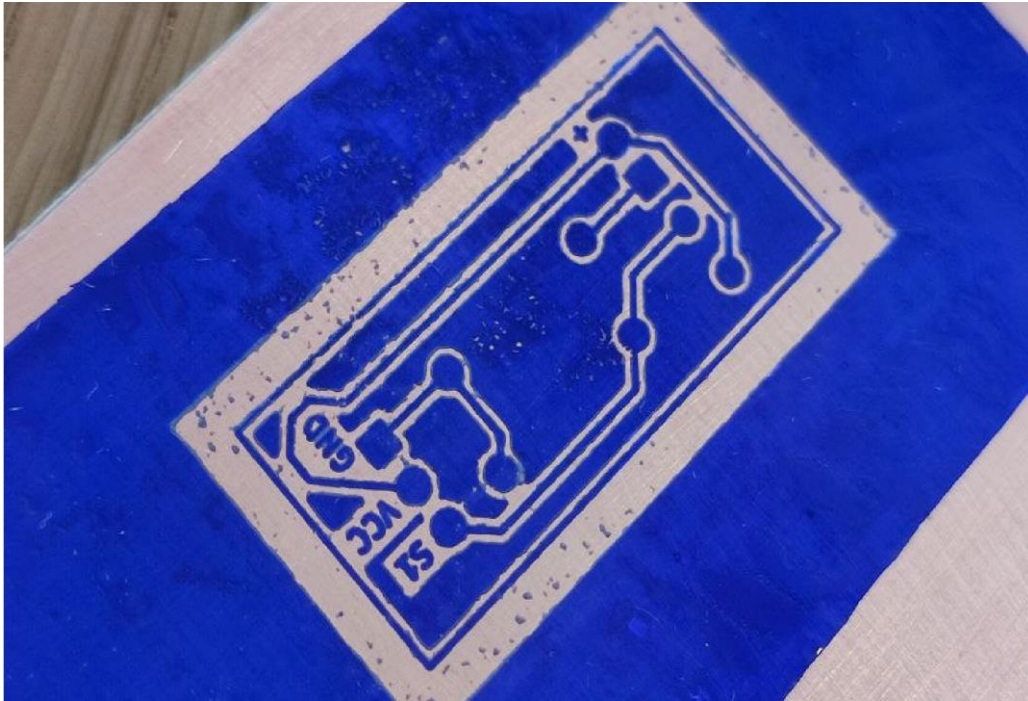


Рисунок 3.7 — Метод фоторезист: полимеризированные участки пленки после смывки

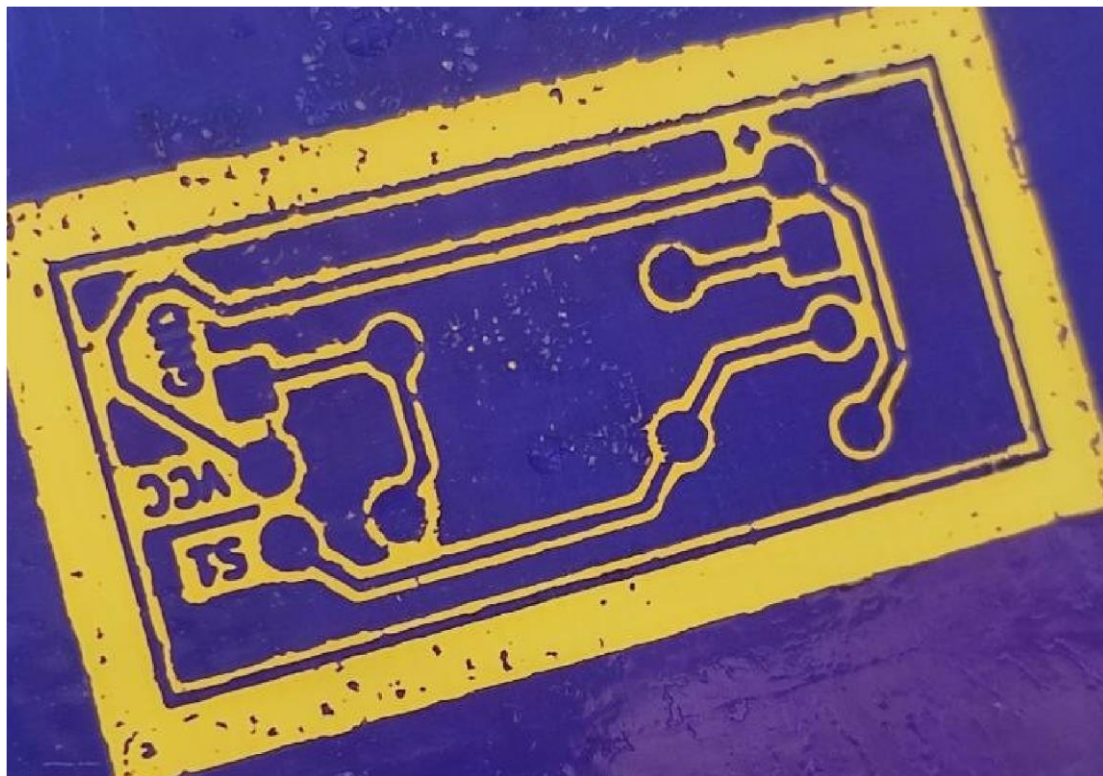


Рисунок 3.8 — Метод фоторезист: Протравленные участки платы

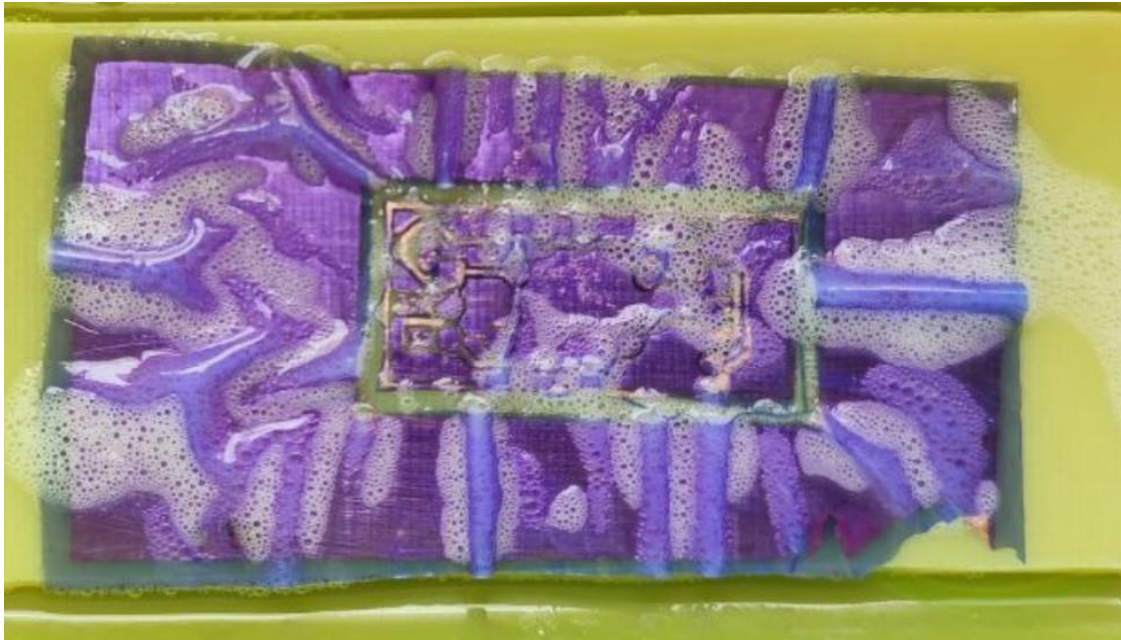


Рисунок 3.9 — Метод фоторезист: финальная смывка фоторезиста

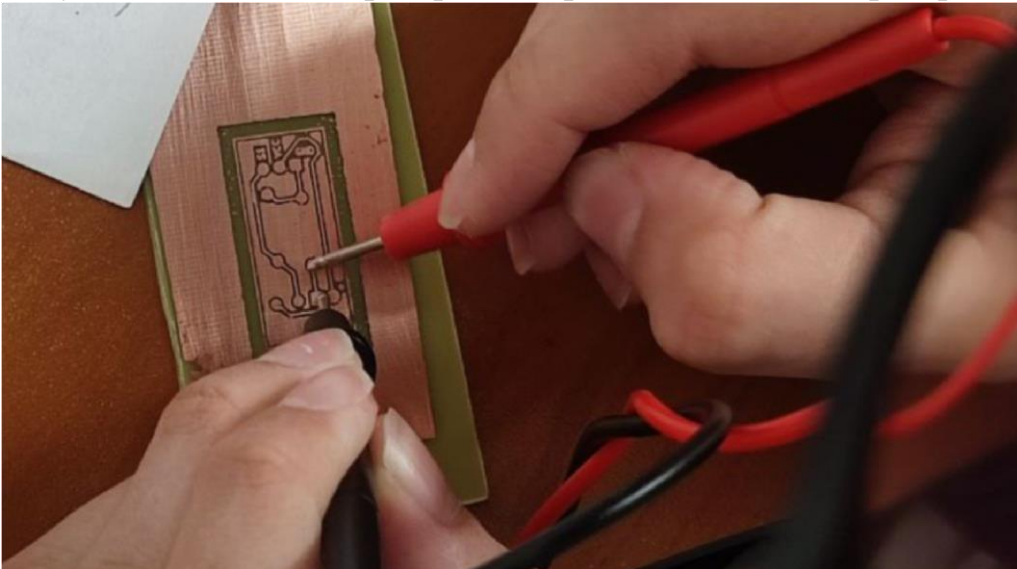


Рисунок 3.10 — Метод фоторезист: проверка контактов готовой платы

В фотоотчете отсутствуют некоторые этапы, такие как прогрев пленки на текстолите, охлаждение после экспонирования, и некоторые другие.

Повторюсь, в целом опыт удался, однако, в силу вышеуказанных причин, которые ведут к слабой предсказуемости результата, от этого способа было решено отказаться.

Для следующего метода нам понадобится фрезеровальный станок. Я использовал популярную настольную модель CNC 3018.



Рисунок 3.11 — Метод фрезеровки: сборка станка

3.2.2 Изготовление управляющей платы.

Для подготовки G-кода я использовал свободно распространяемую программу GGEasy, разработанную российским энтузиастом. Я пробовал до этого встроенные возможности графического редактора Inkscape, но сочетание функционала и простоты GGEasy делает этот вариант **незаменимым**.

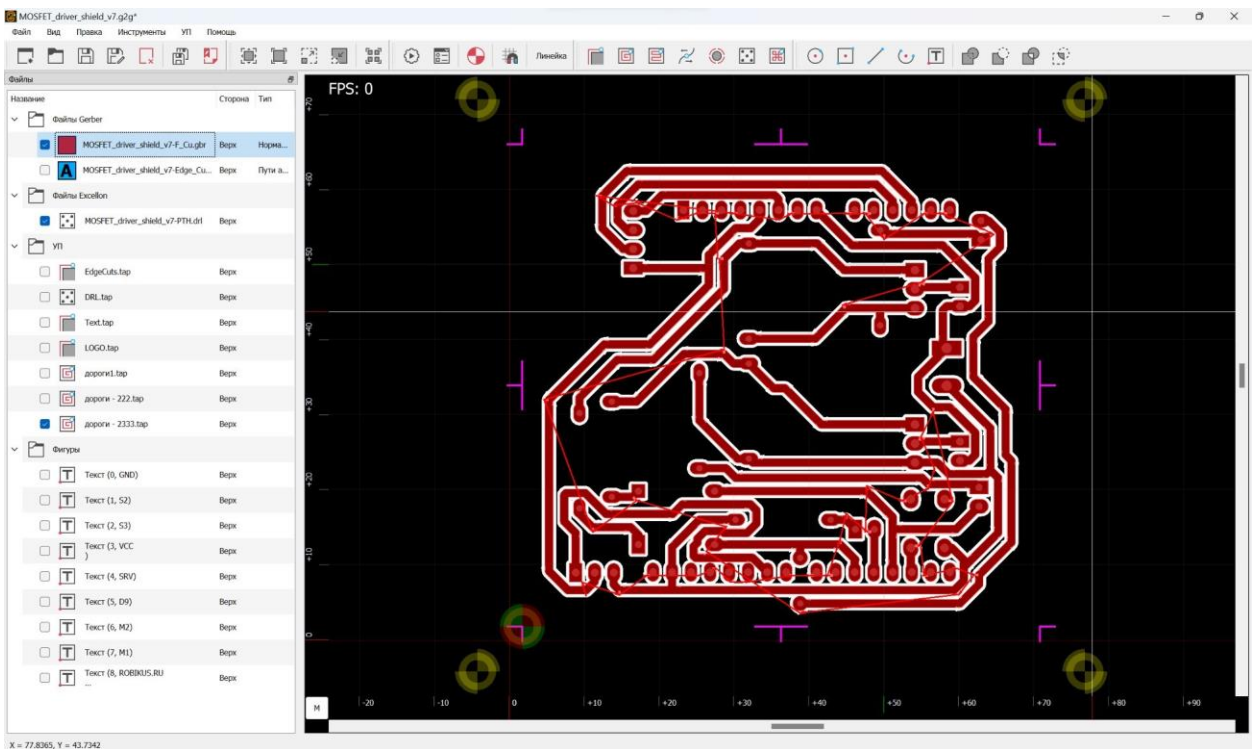


Рисунок 3.11 — Метод фрезеровки: интерфейс программы GGEasy

Итак, в случае с методом фрезеровки, для получения нужного мне результата, я совершил следующие шаги:

1. Подготовил схему цепи в KiCad.
2. Экспортировал соответствующие файлы для:
 - дорожек
 - сверловки
 - текста
 - обрезной рамки
3. Импортировал данные в GGEasy (Рисунок 3.11)
4. Настроил параметры материала, фрез и сверла (Рисунок 3.12)
5. Экспортировал G-код для каждого из файлов.
6. Импортировал G-код в управляющую программу станка (в моем случае Candle).
7. Установил и закрепил на фрезерном столе стеклотекстолит.
8. Установил и закрепил в цанге нужный фрезерный нож.
9. С помощью Z-щупа построил карту высот для нужного участка.
10. Запустил программу резки для дорожек.
11. Повторил п.8-10 для файлов G-кода текста, сверловки и обрезной рамки.

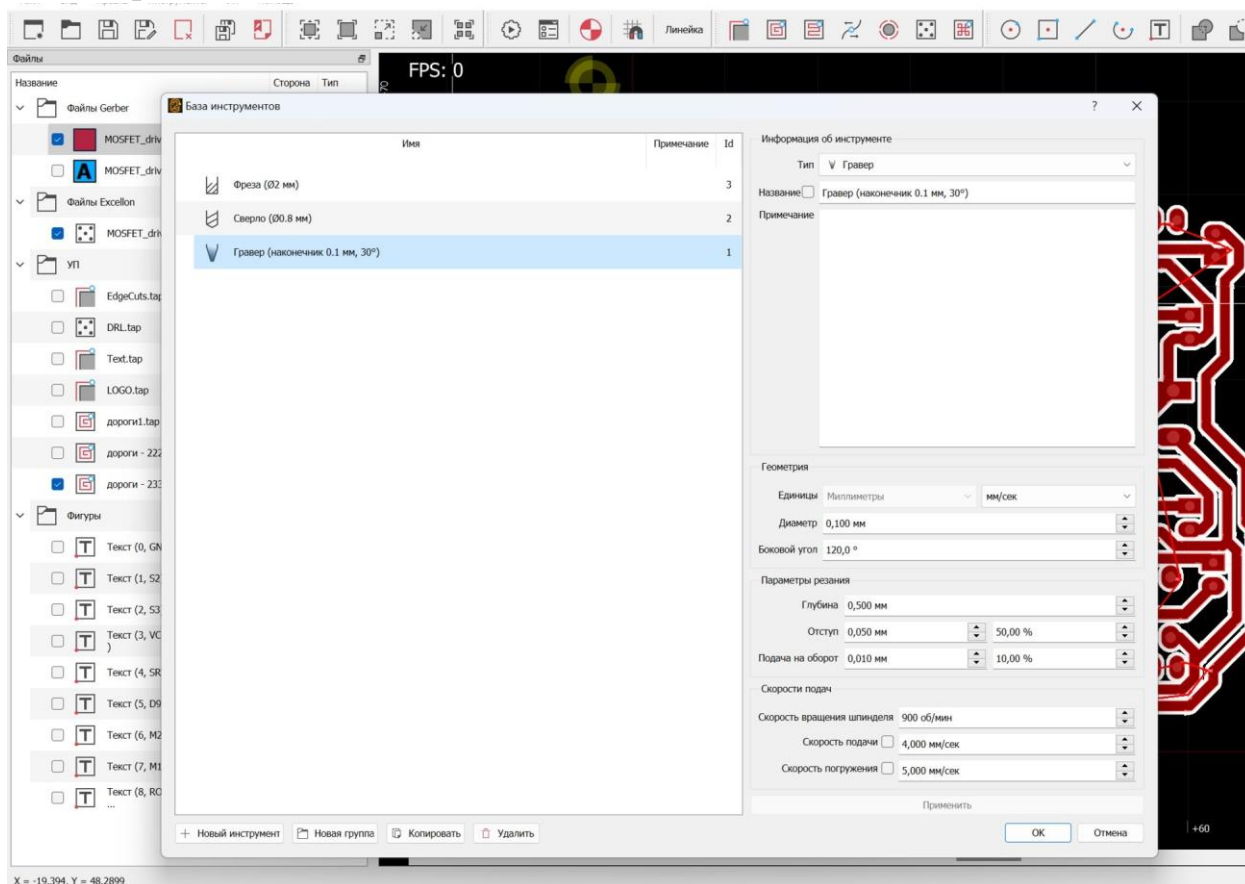


Рисунок 3.12 — Метод фрезеровки: интерфейс программы GGEasy. Настройка инструмента

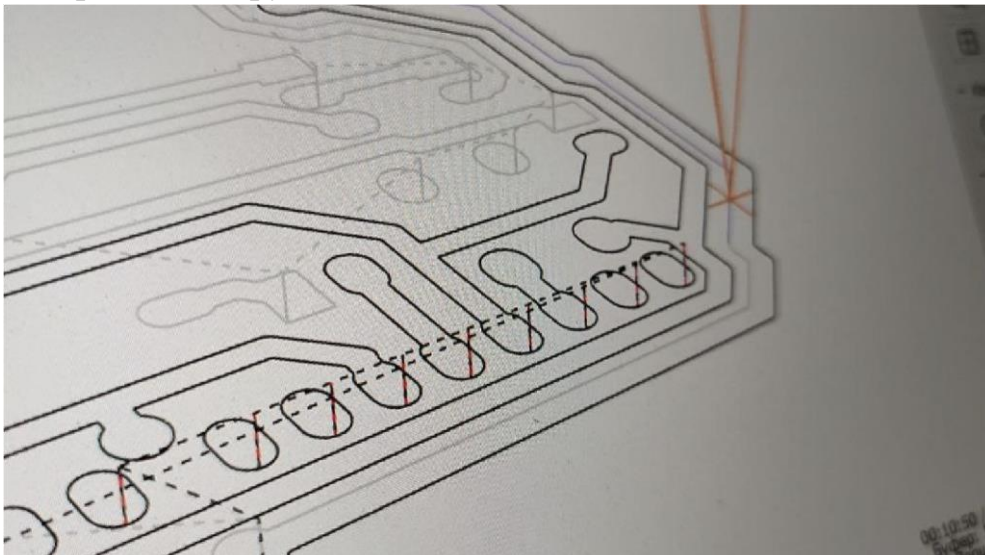


Рисунок 3.13 — Метод фрезеровки: интерфейс программы Candle. Визуальное сопровождение работы станка в режиме резки/фрезеровки

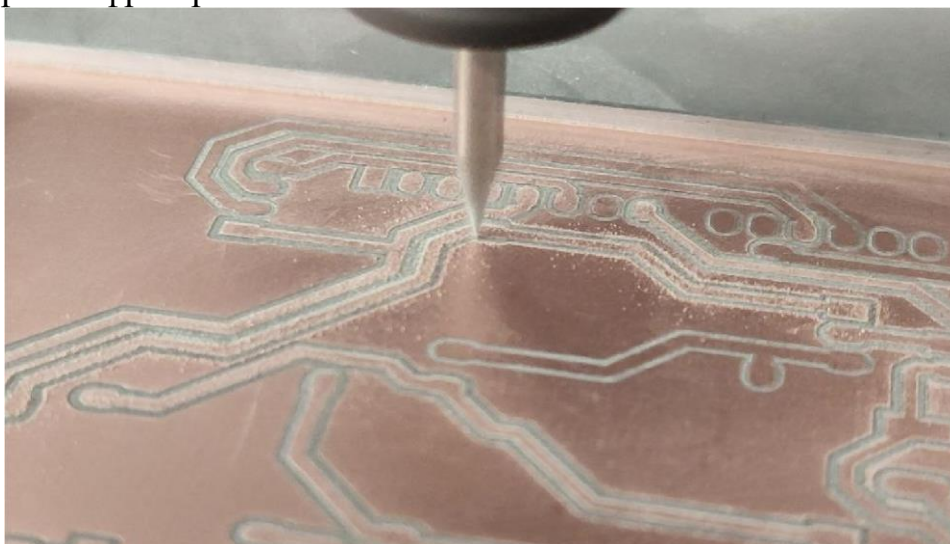


Рисунок 3.14 — Метод фрезеровки: процесс фрезеровки платы

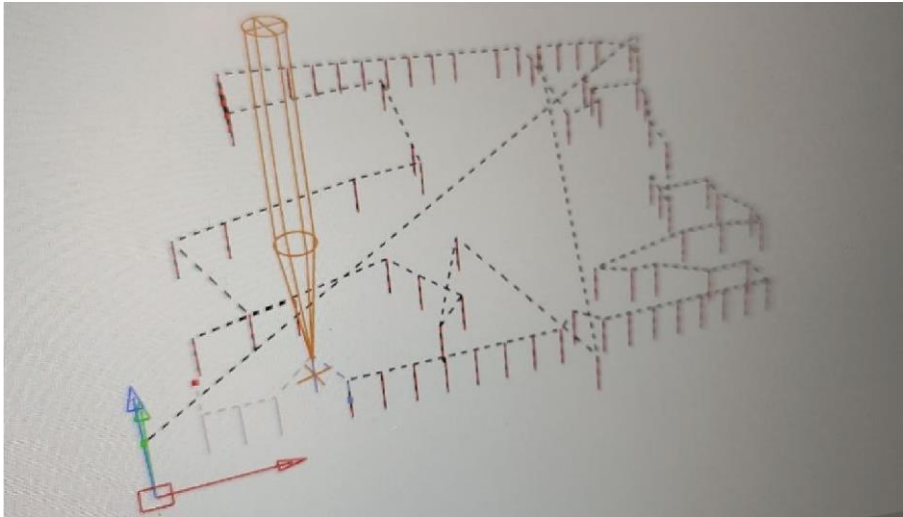


Рисунок 3.15 — Метод фрезеровки: интерфейс программы Candle. Визуальное сопровождение работы станка в режиме сверловки



Рисунок 3.16 — Метод фрезеровки: процесс сверловки платы

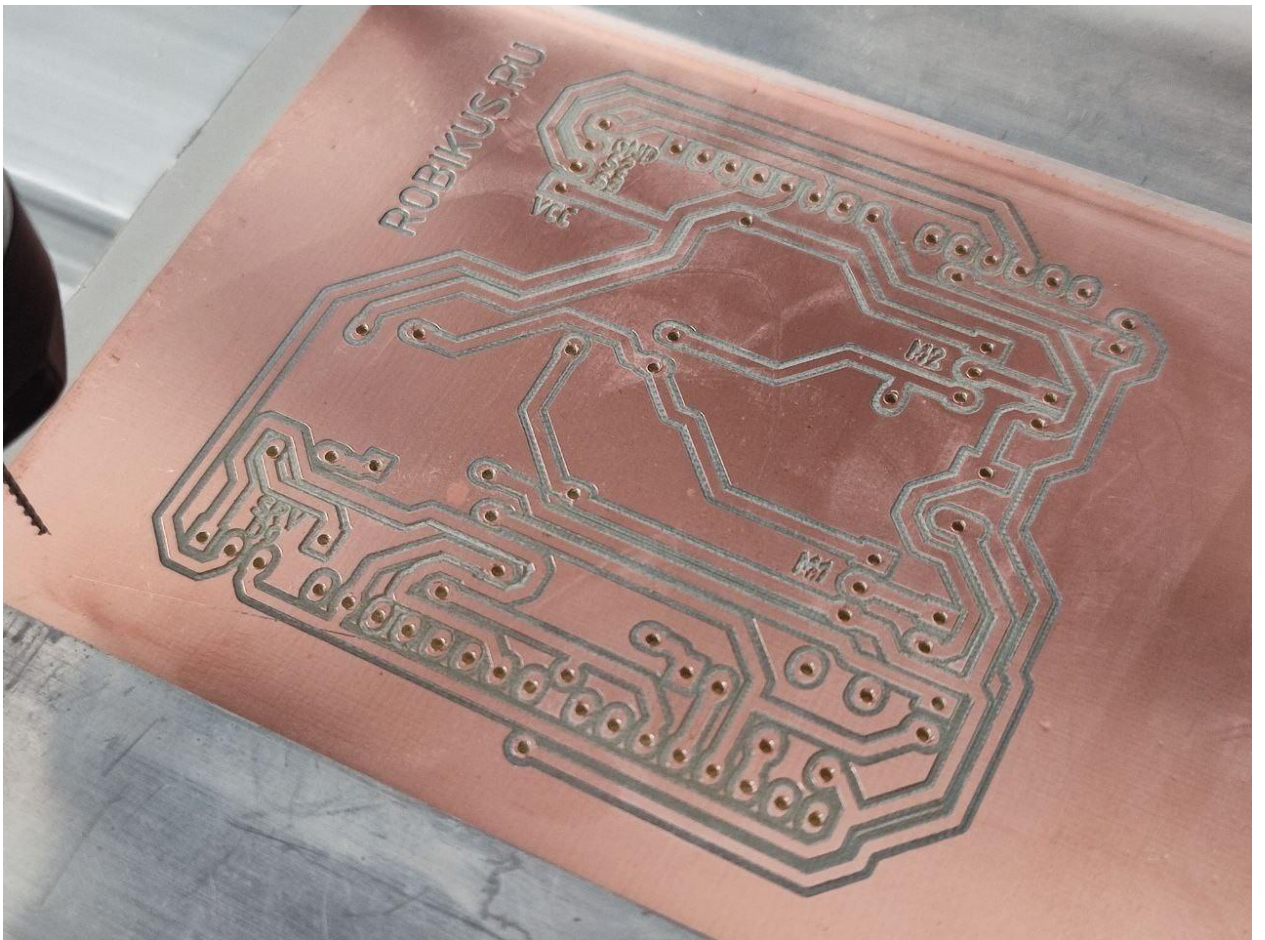


Рисунок 3.17 — Метод фрезеровки: результат фрезеровки файлов дорожек и текста, а также сверловки

Конечно, в процессе производства платы методом фрезеровки возникали трудности и был брак и испорченный материал, но это больше связано с привыканием к новому оборудованию и технологии.



Рисунок 3.18 — часть неудачных или ошибочных вариантов плат. На мой взгляд, основная проблема при фрезеровке плат заключается в толщине границы, отделяющей контактную дорожку от остальной поверхности и соседних дорожек.

Однако теперь, уверенно освоив метод фрезеровки, и научившись задавать нужную мне ширину границы я у меня есть надежный вариант производства односторонних плат. С двухсторонними могут возникнуть проблемы с позиционированием изделия на столе, но я этот вопрос еще не изучал.

Еще один шаг к готовому изделию в рамках производства печатных плат — паяльная маска. Ее нанесение также связано с экспонированием фоточувствительных материалов в ультрафиолете под печатной маской. Однако, этот процесс оказался для меня относительно предсказуемым.

3.2.3 Изготовление управляющей платы.

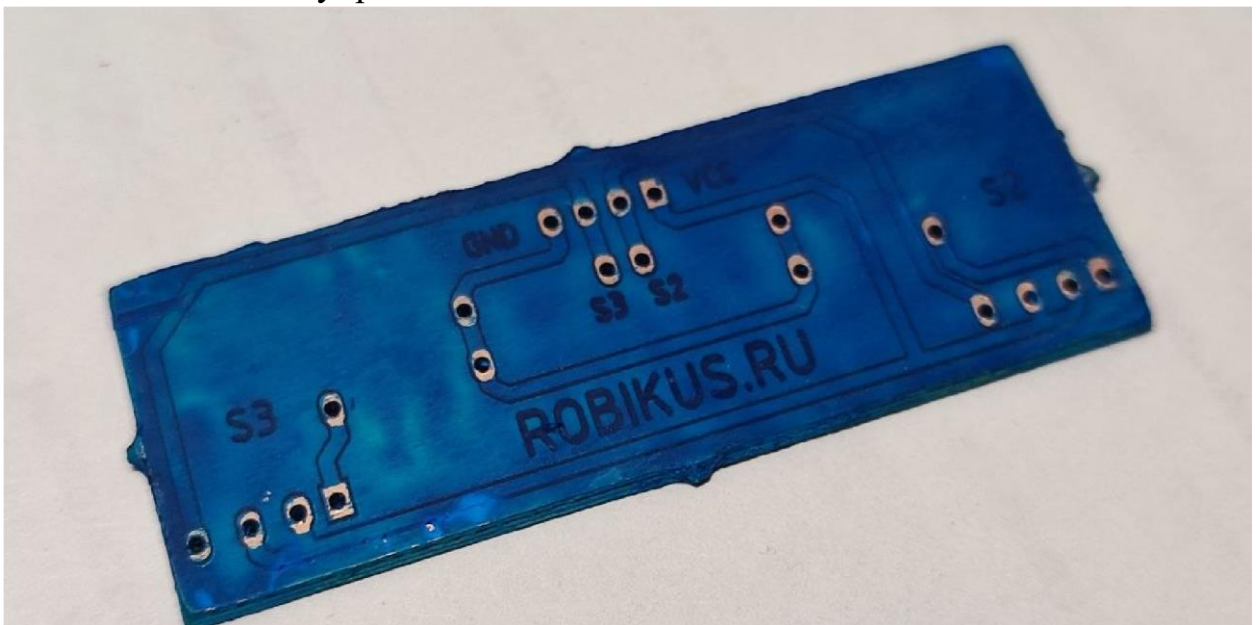


Рисунок 3.19 — Изготовление платы датчика: результат изготовления платы методом фрезеровки с последующим нанесением паяльной маски

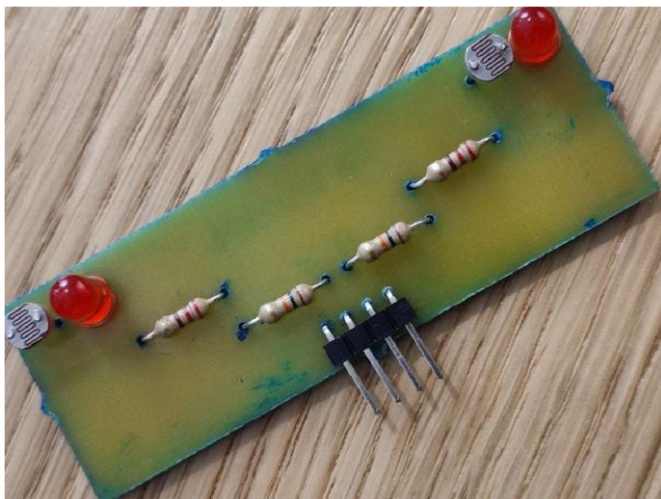


Рисунок 3.20 — Последний шаг — лужение площадок и пайка компонентов

4 Алгоритмическая часть

4.1 В качестве среды программирования была выбрана среда разработки «Arduino IDE» (рисунок 4.1), так как она совместима с отладочной платой Arduino Uno. В качестве регулятора для движения по линии был избран

Пропорционально-

Дифференциальный регулятор (далее PD-регулятор). Все функции, отвечающие за движение и расчёт управляющего воздействия на основе PD-регулятора были вынесены в отдельную библиотеку (Приложение 4).

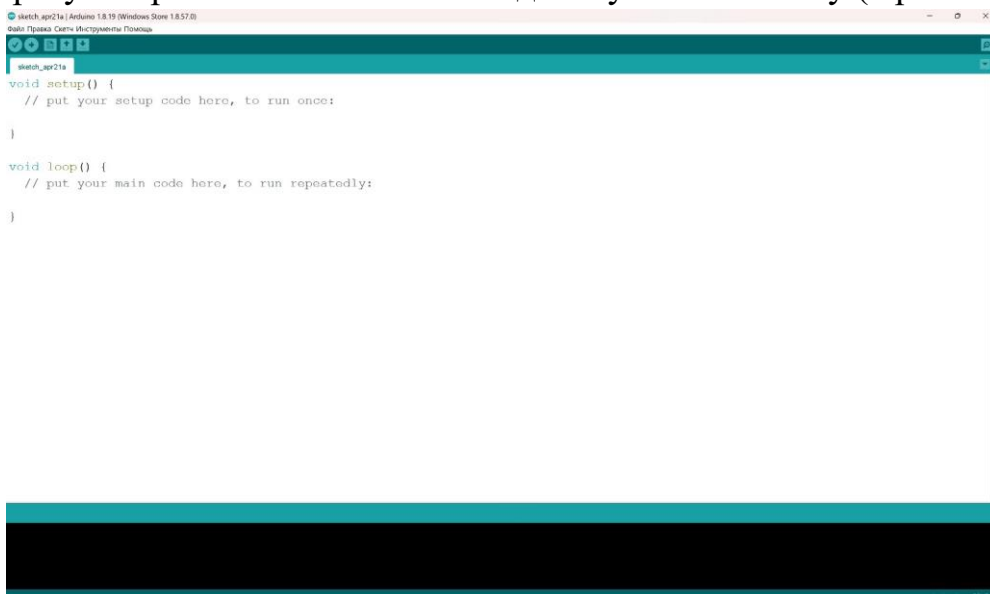


Рисунок 4.1 — интерфейс среды «Arduino IDE»

5. Заключение

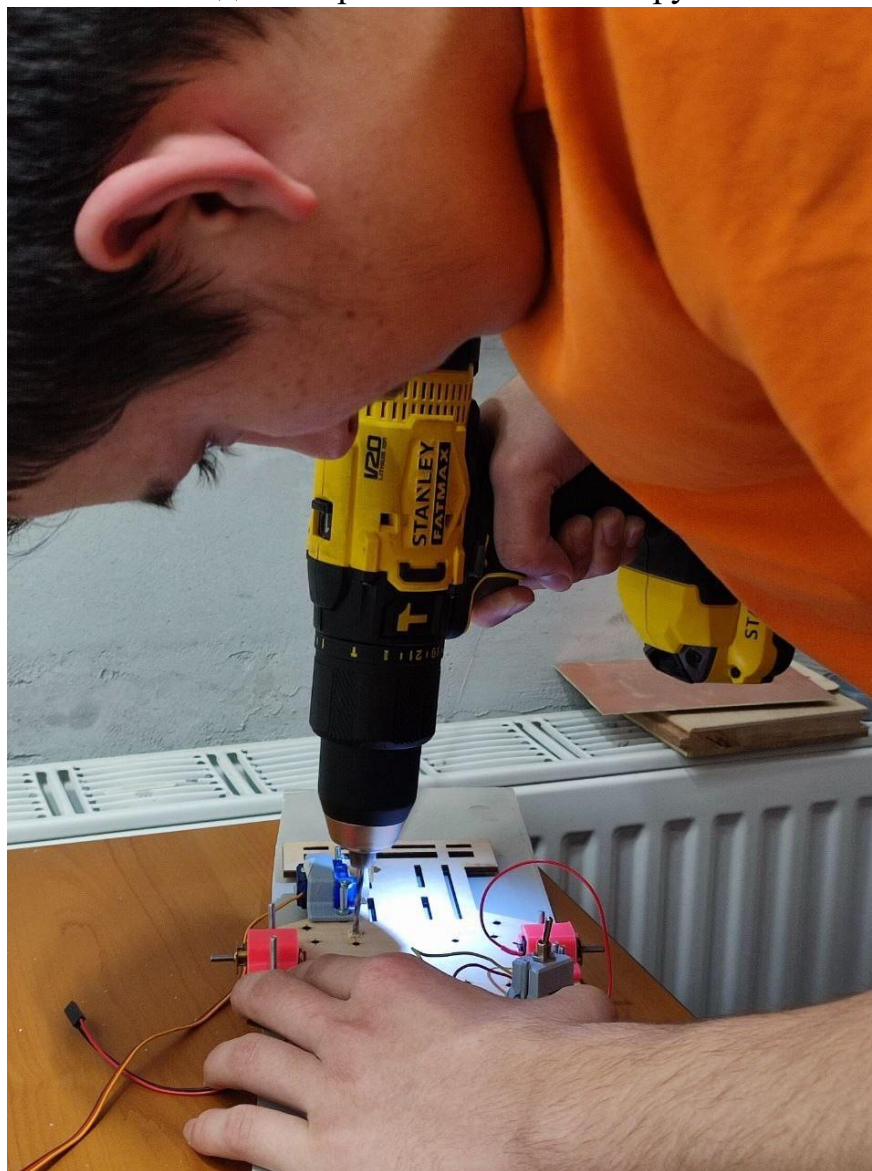
За время работы над проектом я узнал много новой, интересной и полезной информации.

Я узнал, что обычные кухонные ингредиенты могут помочь разрезать металл, и что средство для прочистки труб не такое скучное как выглядит, и что мамина ультрафиолетовая лампа для ногтей оказывается нужна в инженерном хозяйстве.

Я активно пользовался инструментами, подручными средствами, обзавелся тисками, наждачкой, аккаунтом на сайте Амперки, и даже научился пользоваться фрезерным станком с ЧПУ!

Я научился лучше паять, понял, что не все припои одинаково полезны, и ещё раз убедился, что сроки иногда кажутся оптимистичней чем на самом деле.

Самое яркое впечатление от проекта — осознание того, что этот робот едущий по полю — я сделал практически своими руками!



6. Приложения:

Приложение 1 — чертёж крепления тумблера

КОМПАС-3D v21 Учебная версия © 2022 ООО "АСКОН-Системы проектирования", Россия. Все права защищены.

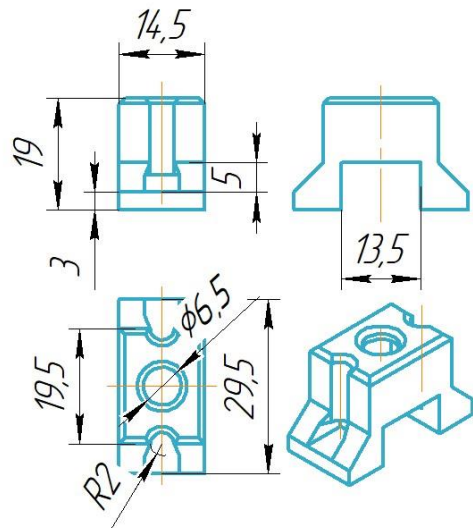
Справ. №

Взам. инв. № Инв. № дубл. Подп. и дата

Подп. и дата

Не для коммерческого использования

Перв. примен.



| Изм. | Лист | № докум. | Подп. | Дата |
|----------|------|----------|-------|------|
| Разраб. | | | | |
| Проб. | | | | |
| Т.контр. | | | | |
| Н.контр. | | | | |
| Утв. | | | | |

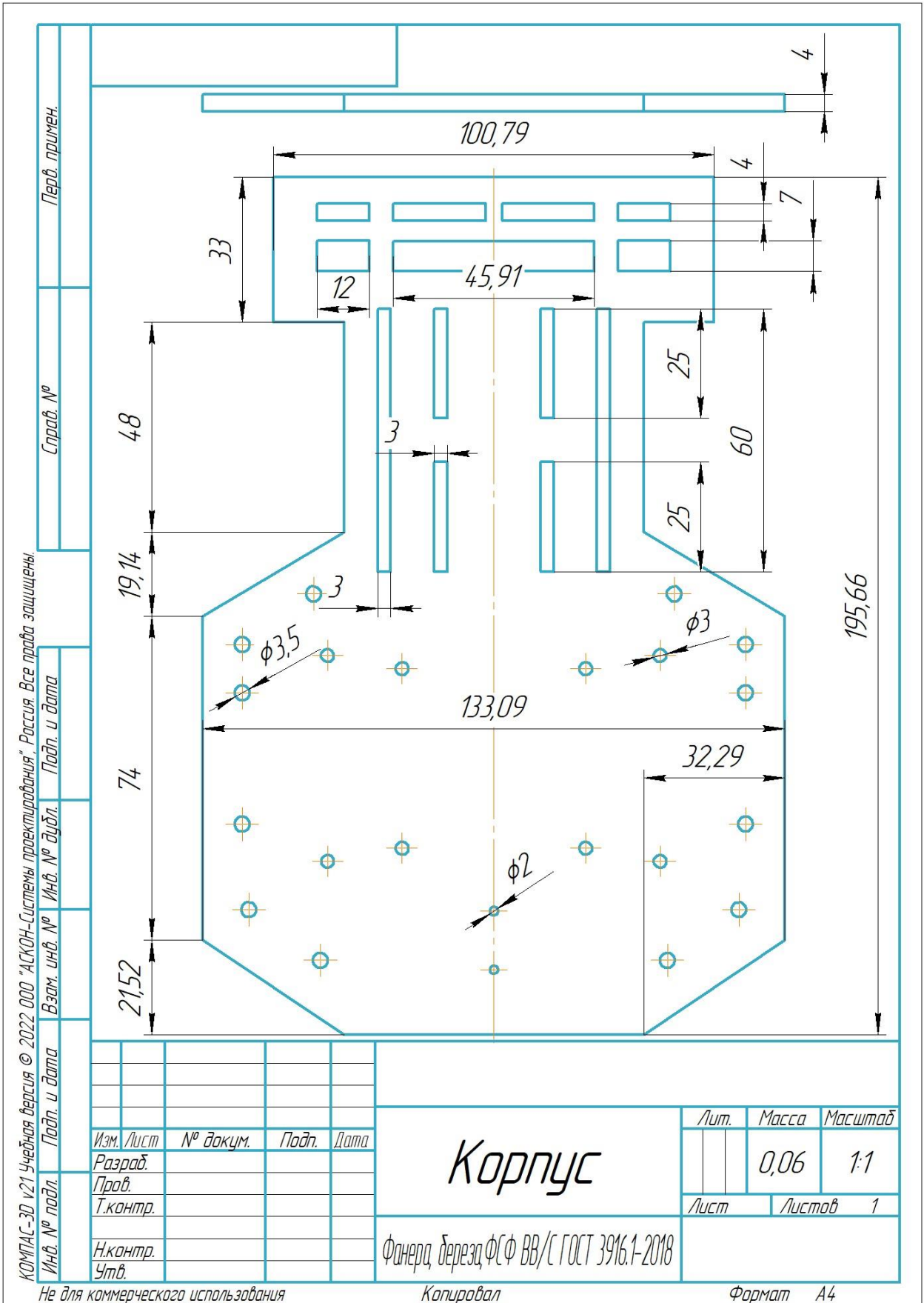
Крепление тумблера
Полипропилен ГОСТ 26996-86

| Лист | Масса | Масштаб |
|------|--------|---------|
| | 0 | 1:1 |
| Лист | Листов | 1 |

Копировал

Формат А4

Приложение 2 — чертёж корпуса робота

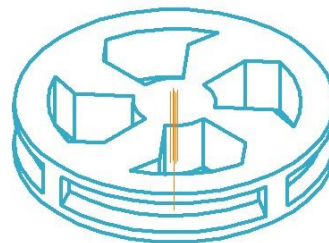
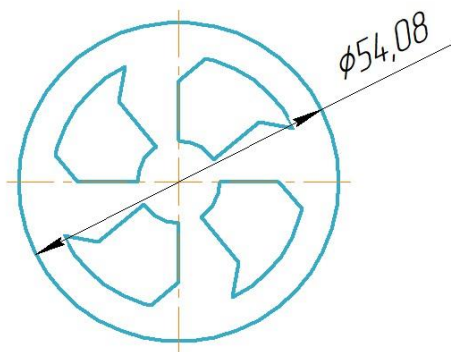
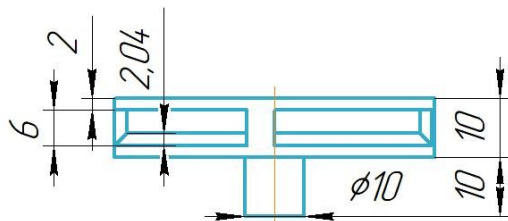


Приложение 3 — чертёж колёсного диска

КОМПАС-3D v21 Учебная версия © 2022 ООО "АСКОН-Системы проектирования", Россия. Все права защищены.

Справ. №

Перв. примен.



Изм. № подл. Подп. и дата

Изм. Лист

№ докум.

Подп.

Дата

Колёсный диск
(правый)

Лист

Масса

Масштаб

0,01

1:1

Лист

Листов

1

Инд. № подл.

И.контр.

Утв.

Полипропилен

ГОСТ

26996-86

Не для коммерческого использования

Копировал

Формат А4

Приложение 4 — код библиотеки движения робота

```
sketch_apr21a | Arduino 1.8.19 (Windows Store 1.8.57.0)
Файл Правка Скетч Инструменты Помощь

sketch_apr21a.g
#include <Arduino.h>
// Библиотека для управления лодочной частью робота для Кинго инженера
// Автор: Кирилл Коберя, 2024

// Основные константы
#define M1S 3 // Левый мотор
#define M2S 5 // Правый мотор
#define S1 аналогA1(A1) // показания левого датчика линии (аппаратный)
#define S2 аналогA1(A2) // показания правого датчика линии (аппаратный)
#define SENS1 map(S1,490,520,0,100) // Показания левого датчика линии (с нормализацией)
#define SENS2 map(S2,490,520,0,100) // Показания левого датчика линии (с нормализацией)

// Переменные для ПИ-регулятора:
float P = 1.4; //Пропорциональный
float I = 64 // Дифференциальный
int lastError = 0; //Провал ошибки

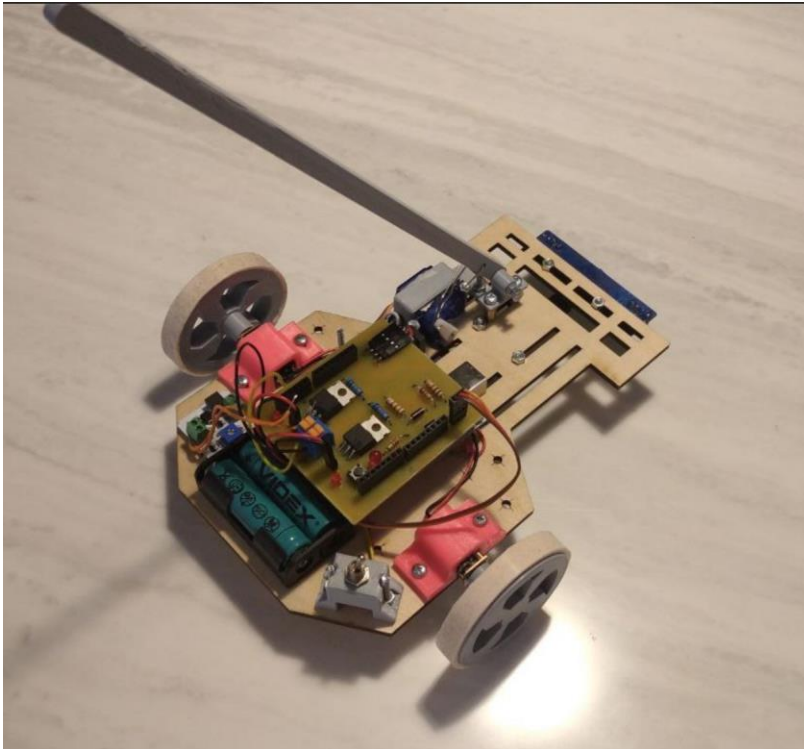
void init_drive() { // Инициализация портов моторов
  pinMode(M1S, OUTPUT);
  pinMode(M2S, OUTPUT);
}

void driveBC(int MB, int MC) { // Функция подачи необходимого напряжения на моторы (в процентах)
  analogWrite(M1S, map(constrain(MB,0,100),0,100,0,255));
  analogWrite(M2S, map(constrain(MC),0,100),0,100,0,255));
}

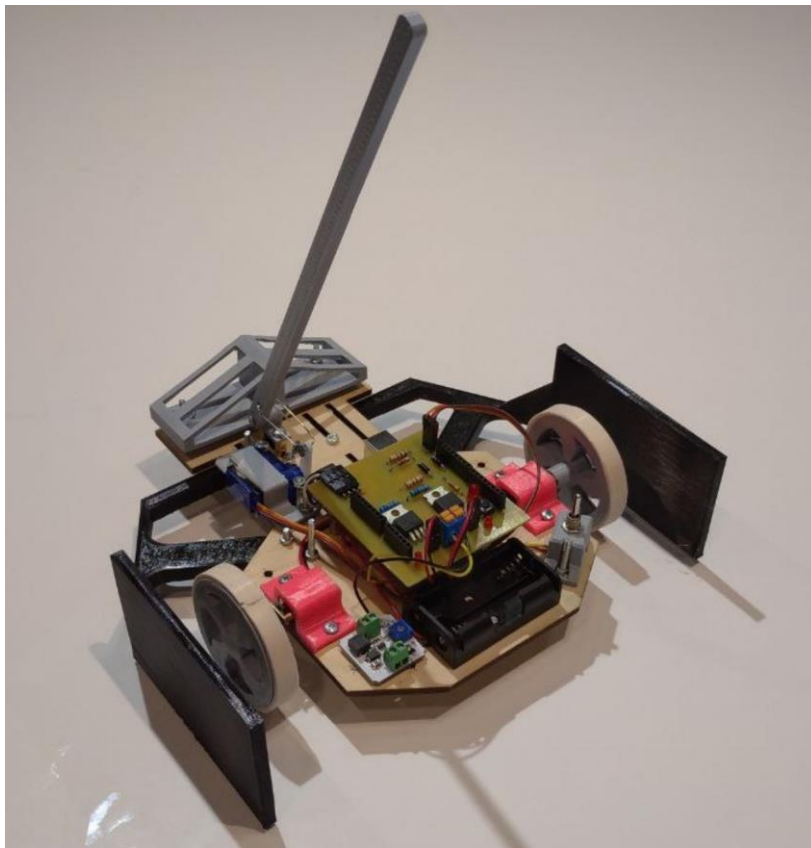
int PReg(int par1, int par2) { // одна итерация ПИ-регулятора
  int error = par1 - par2; // Ошибка
  float P = error * P; // Пропорциональная составляющая
  float I = error * lastError * I; // Дифференциальная составляющая
  lastError = error; //Обновление прошлой ошибки;
  return P + I; // Сумма составляющих
}

void line_millis(int t, int V) { // движение по линии в течении t миллисекунд со скоростью V
  int tStart = millis(); // время начала
  while (millis() - tStart < t) { // Повторять пока не пройдет t миллисекунд
    int UV = PReg(SENS1, SENS2); // Управление лодочной частью
    driveBC(UV, UV * V); // Подача расчетного напряжения
    delay(5); // Небольшая задержка для верной работы дифференциальной составляющей
  }
}
```

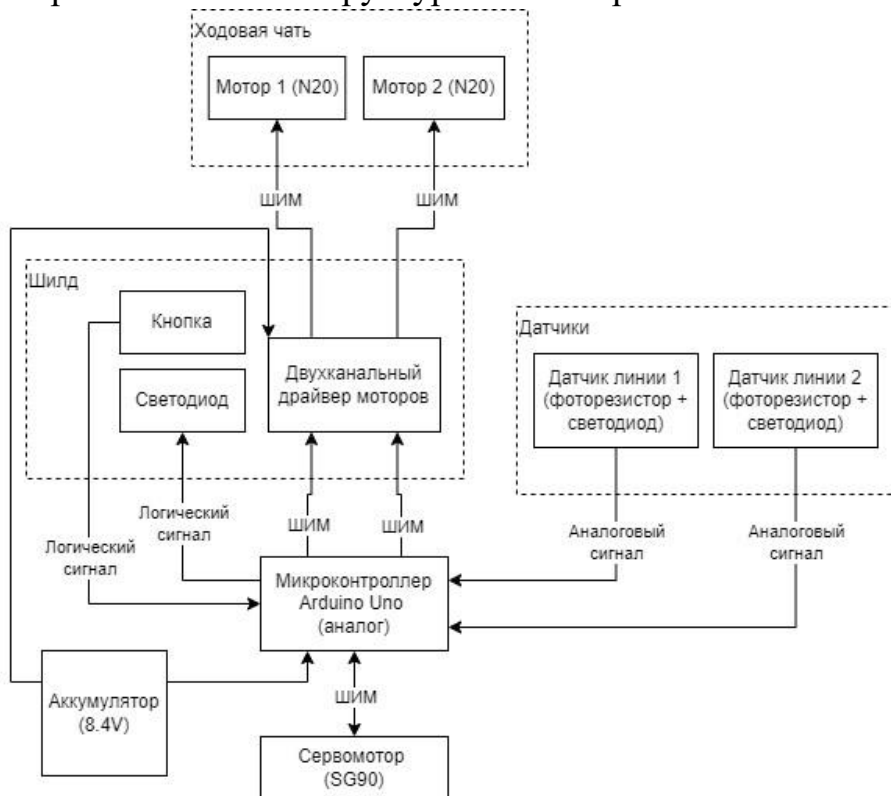
Приложение 5 — общий вид робота на 19.04.2024



Приложение 6 — общий вид робота на 06.06.2024



Приложение 7 — структурная схема робота



Приложение 8 — полная принципиальная схема робота

