

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский технологический
университет «МИСиС»

ИНСТИТУТ	ЭКОТЕХНОЛОГИЙ И ИНЖИНИРИНГА
КАФЕДРА	МЕТАЛЛОВЕДЕНИЕ ЦВЕТНЫХ МЕТАЛЛОВ
НАПРАВЛЕНИЕ	15.04.02 Технологические машины и оборудование

Практика цифрового производства

на тему: “Робот на радиоуправлении”

Студент: Иост Е.С.

Группа: МТМО-22-3

Проверил: Тавитов А.Г.

Москва 2023

Введение

Цель проекта: построить робота без колес с дистанционным управлением для прохождения полосы препятствий.

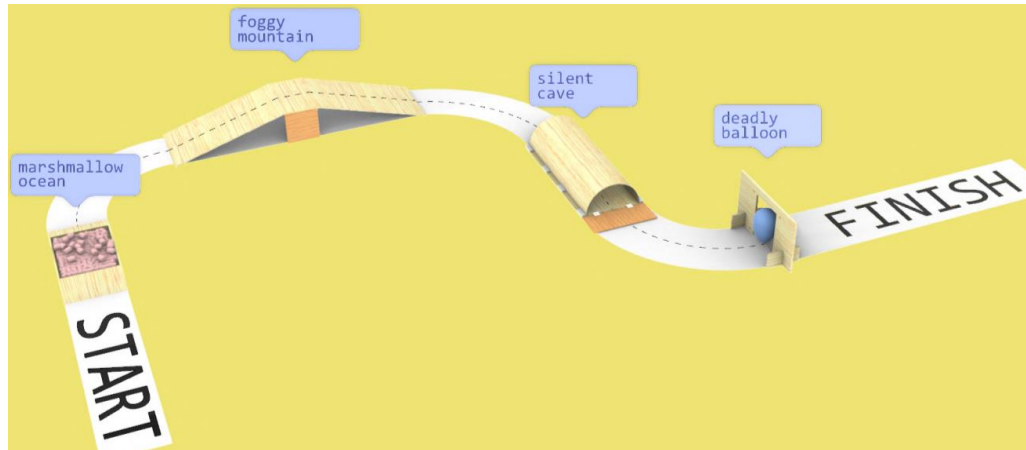


Рисунок 1 – Полоса препятствий

Компоненты, которые использовались в данном проекте:

- NRF
- NRF+
- Motor Shield
- Arduino Leonardo/Amperka Iskra Neo
- Arduino Pro Micro
- x2 18650
- x2 DC motor
- x2 joystick
- RGB LED strip
- potentiometer
- x2 bradboard
- powerbank
- vape mod
- servo 360

Скетчинг, эскизирование

Для начала работы было принято решение заскетчить все возможные варианты рабочих механизмов и расположения компонентов. Были сняты все размеры основных компонентов и важных узлов.

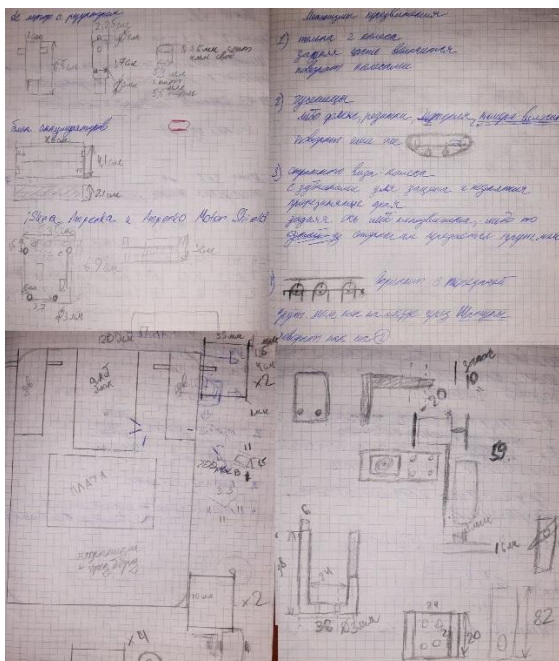


Рисунок 2 – Эскизы

По итогу было принято решение делать бота на гусеницах на прямоугольной платформе.

Моделирование и изготовление компонентов

Моделирование проводилось в программах CorelDRAW и SolidWorks. Компоненты изготавливались на 3D FFF принтере и лазерном ЧПУ станке.

Так как было принято решение делать бота на гусеницах необходимо сделать по 4 валька с каждой стороны. Основные ведущие приводные крепятся на валы моторов, вспомогательные служат для создания натяжения и трапецевидной формы гусеницы, а второстепенные такие же как основные только с круглым посадочным отверстием под болт.

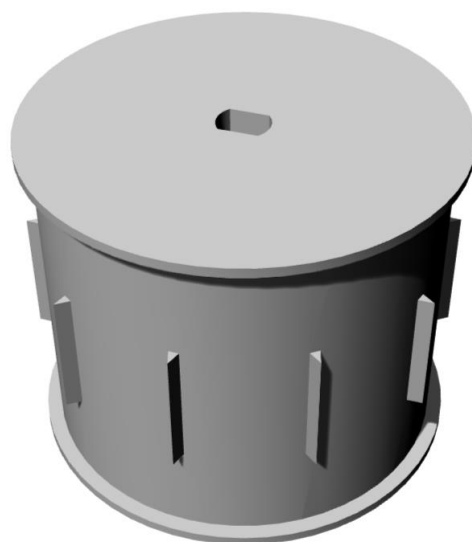


Рисунок 3 – 1 вариант модели основного валка

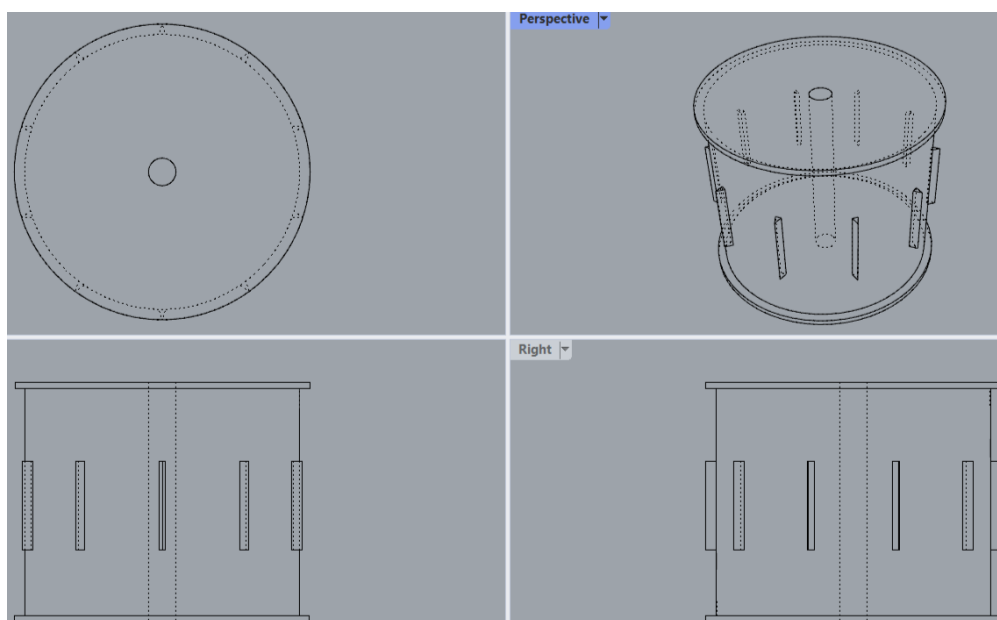


Рисунок 4 – 1 вариант модели второстепенного валка

Диаметр вала 40 мм, выступ 1 мм, гребенка 1 в высоту и 10 мм в длину, ширина 35 мм. Отверстие на основном вале по размерам выходного вала с мотора 3.5мм в высоту выреза и 5.5 мм по диаметру, на второстепенном 4 мм.

Далее по ходу сборки и тестирования аппарата выяснилось, что высота выступа в 1 мм слишком мала и гусеница постоянно соскакивала при передвижении. Было принято решение переделать модель и изготовить по-новому валки. Размер выступа в этот раз был изменен на 10 мм.

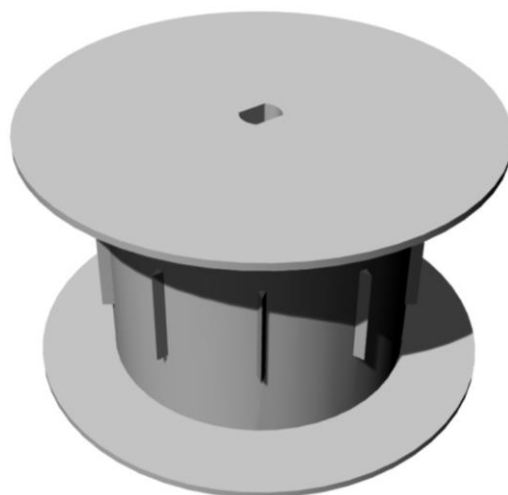


Рисунок 5 – 2 вариант модели основного валка

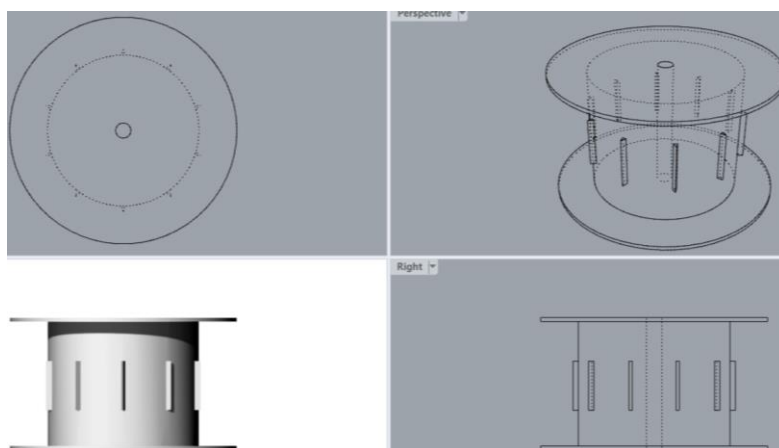


Рисунок 6 – 2 вариант модели второстепенного валка

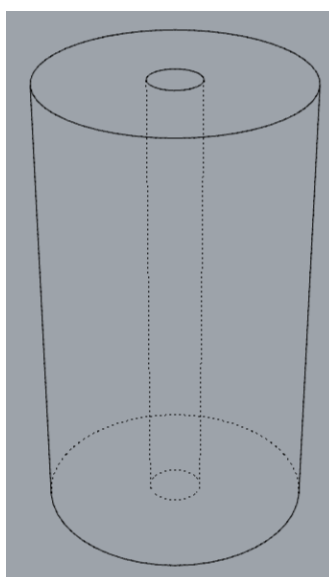


Рисунок 7 – Модель вспомогательного валка

Далее необходимо было произвести кронштейны крепления моторов и валов. По заранее снятым размерам с компонентов были сделаны модели и произведена печать на принтере.

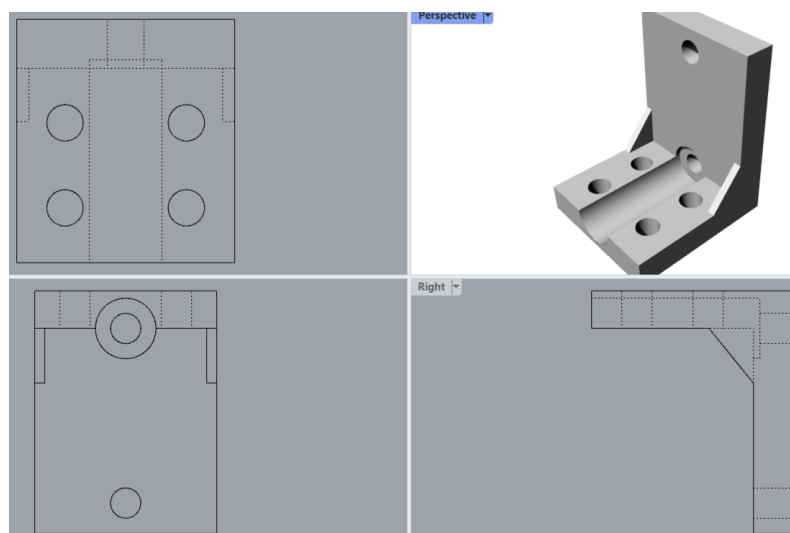


Рисунок 8 – Кронштейн крепления мотора

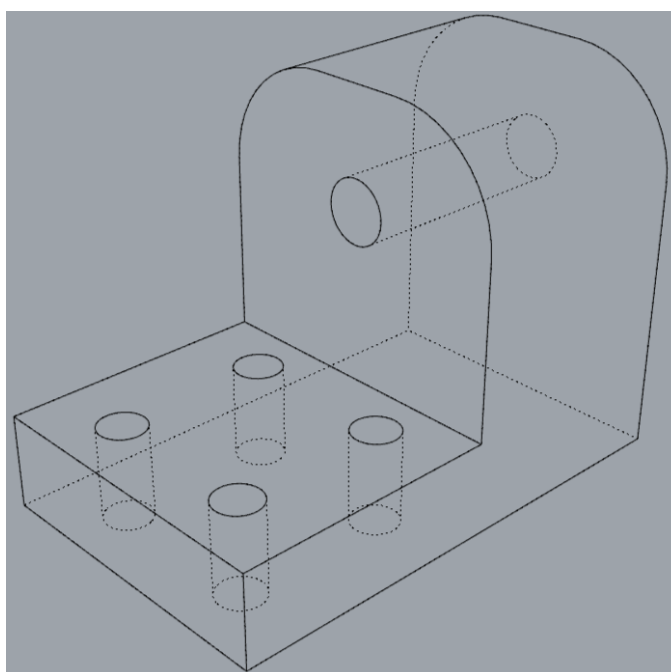


Рисунок 9 – Кронштейн крепления второстепенных валов

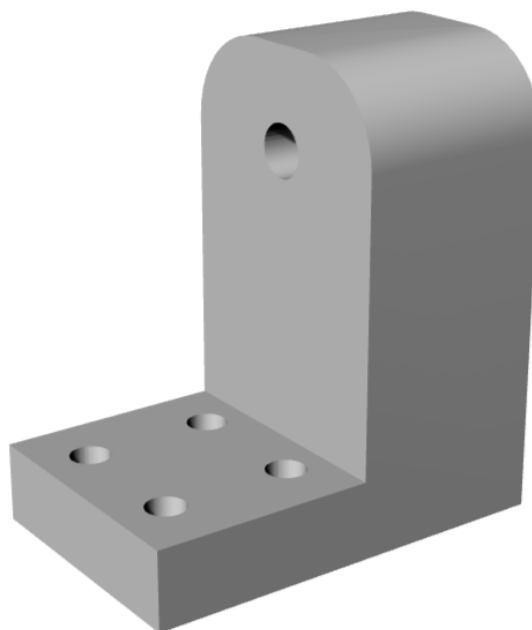


Рисунок 10 – Кронштейн крепления вспомогательных валов

Так как на полосе препятствий есть обязательное испытание – это лопнуть шарик и желательно это сделать интересным способом, было принято решение использовать `var mod` для этого. Для него тоже необходимо сделать крепление.

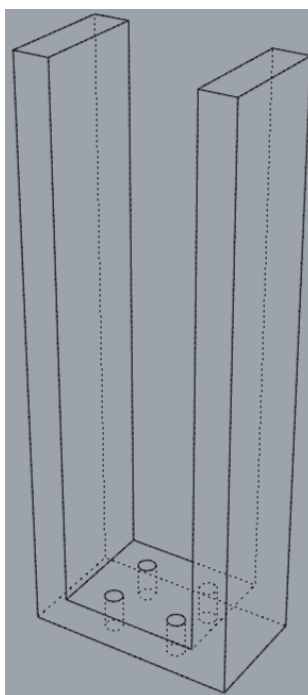


Рисунок 11 – Кронштейн крепления мода

Основная платформа вырезалась на лазерном станке из фанеры 6 мм, размерами 12x18 см.

Сборка компонентов

Для начала расположение компонентов размечалось на основной платформе. Далее из сложностей было, что 3D печать пластиком не идеальная технология и после печати необходимо дорабатывать детали, так нужно было отрывать поддержки, обрабатывать поверхности и рассверливать отверстия под нужные диаметры. Сборка кронштейном производилась с применением болтов М3, а в качестве осей вращения для валов было принято решение взять М4. Также для лучшего вращения валов использовались шайбы.

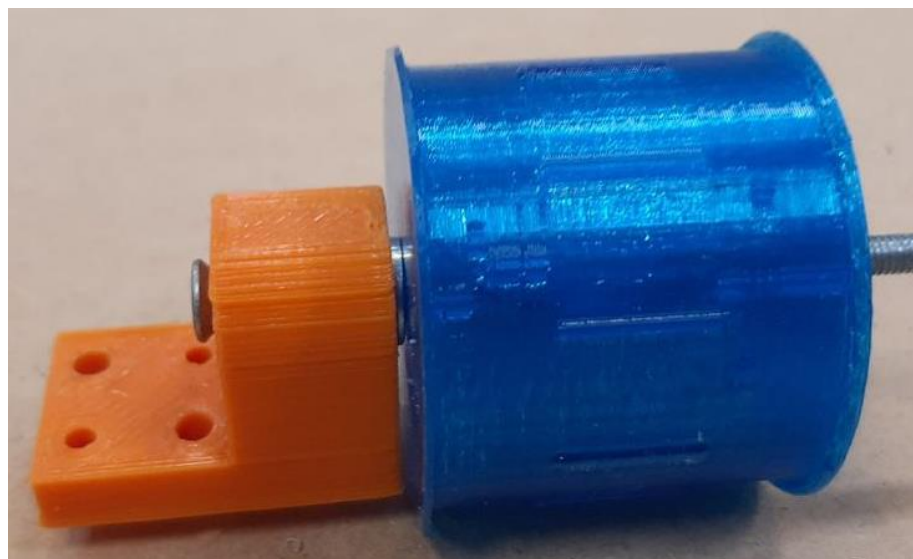


Рисунок 12 – Вал с кронштейном в сборе

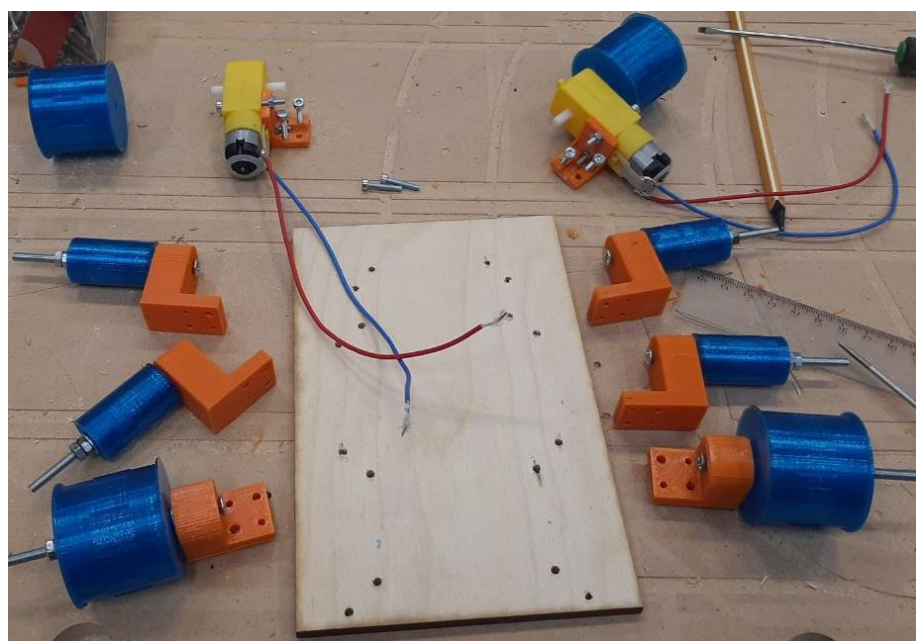


Рисунок 13 – Сборка готовых узлов

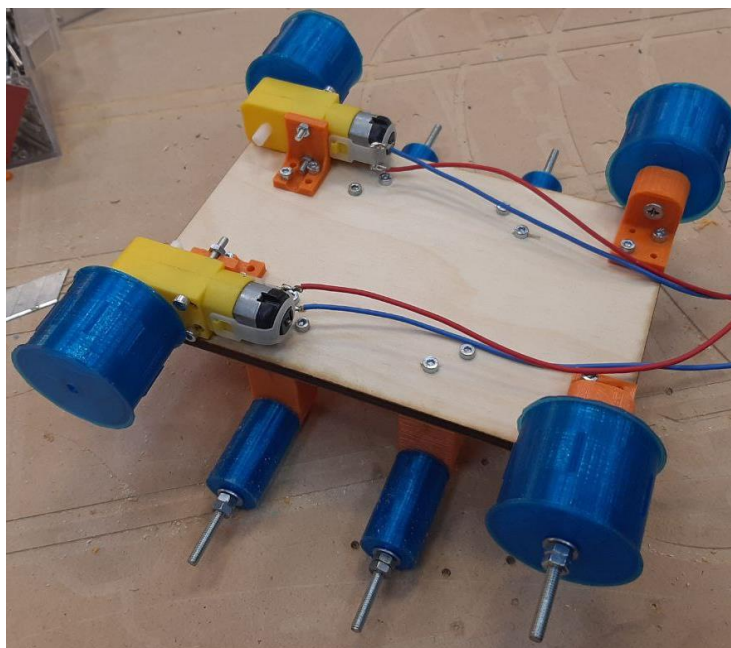


Рисунок 14 – Готовая сборка первичных компонентов

После долгого принятия решения по поводу изготовления гусениц, за основу была взята велосипедная камера. Она была разрезана на лоскуты шириной 3.5 см по ширине валов. Длина измерялась практическим методом. Для скрепления двух концов использовался степлер и клей.

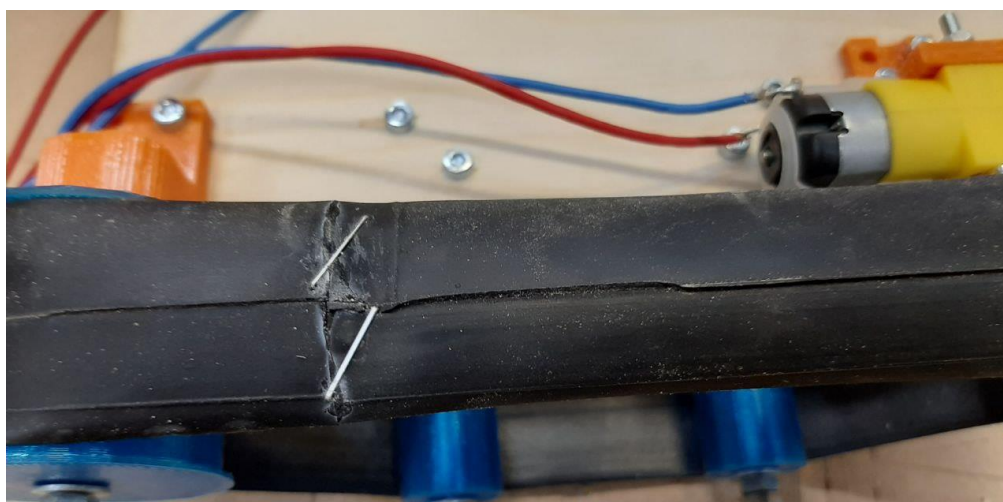


Рисунок 15 – Стык гусеницы

Для дальнейшей работы сначала необходимо было произвести тестовую сборку электроники без радио модуля. Все компоненты ввода были подключены на прямую к плате. Производилась отладка работы двигательных систем и работы сервопривода с написанием кода для платы управления Arduino Leonardo.

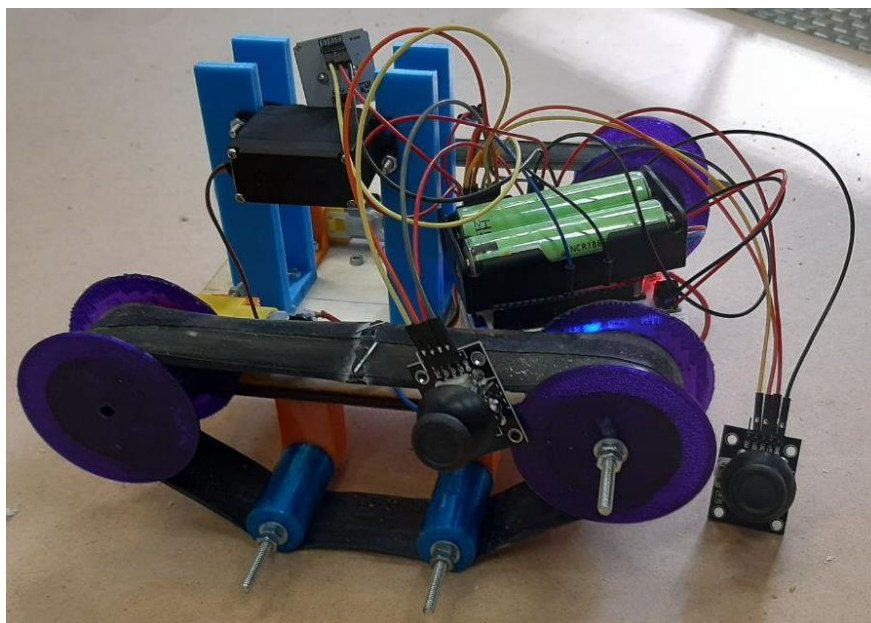


Рисунок 16 – Предварительная сборка электроники

После предварительной отладки работы приступили к подключению радио модулей и изготовлению мультя управления. Для простоты и меньшей траты ресурсов, было принято решение производить сборку и взять за основу пульта длинный bradboard. Компоненты крепились на саму основу, или на стяжки, или на двухсторонний скотч. Выбиралось эргономичное расположение и производился кабель менеджмент.

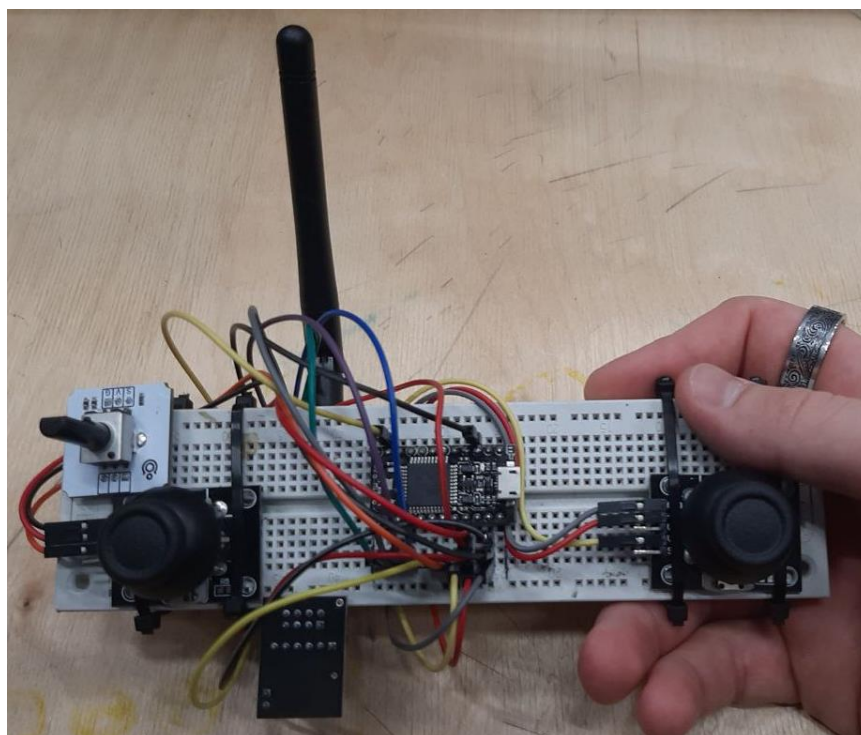


Рисунок 17 – Готовый пульт управления

Далее производилась конечная сборка компонентов робота. Элементы электроники крепились на двухсторонний скотч к основе, светодиодная лента на термоклей в передней части бота. Аккумуляторный блок закрепили под основной платформой. За механизм для лопанья шарика был взят vare mod. На его дрипку была накручена спираль вытянутой формы, на кронштейнах закреплен сервопривод, который нажимает на кнопку мода при считывании сигнала с потенциометра, спираль при нажатии начинает разогреваться.

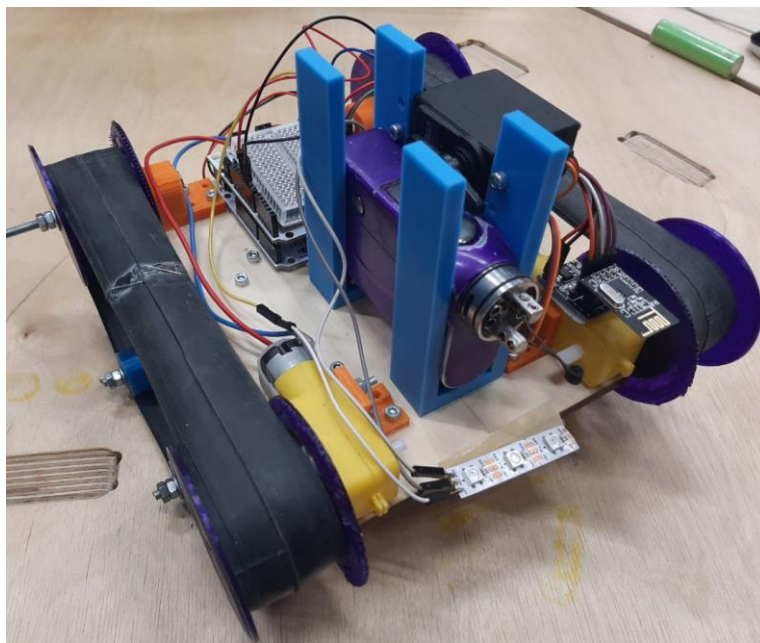


Рисунок 18 – Готовый бот

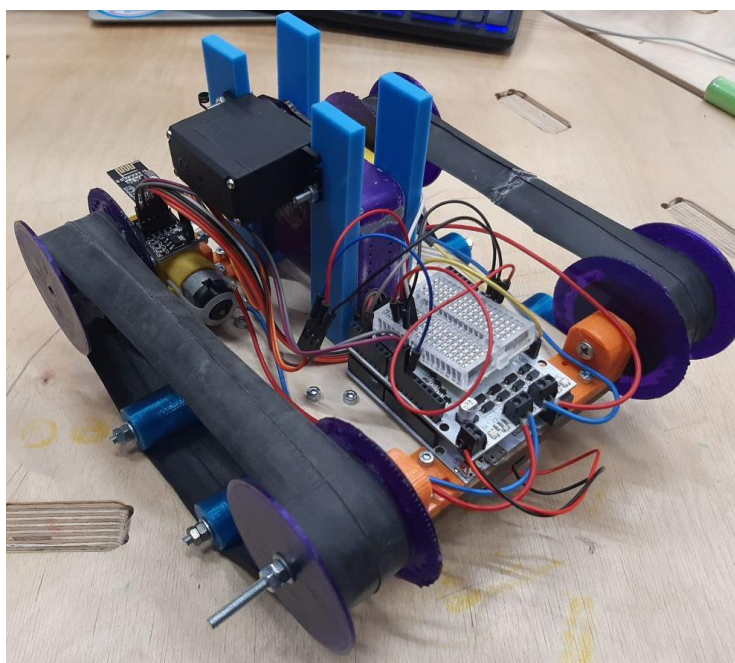


Рисунок 19 – Готовый бот

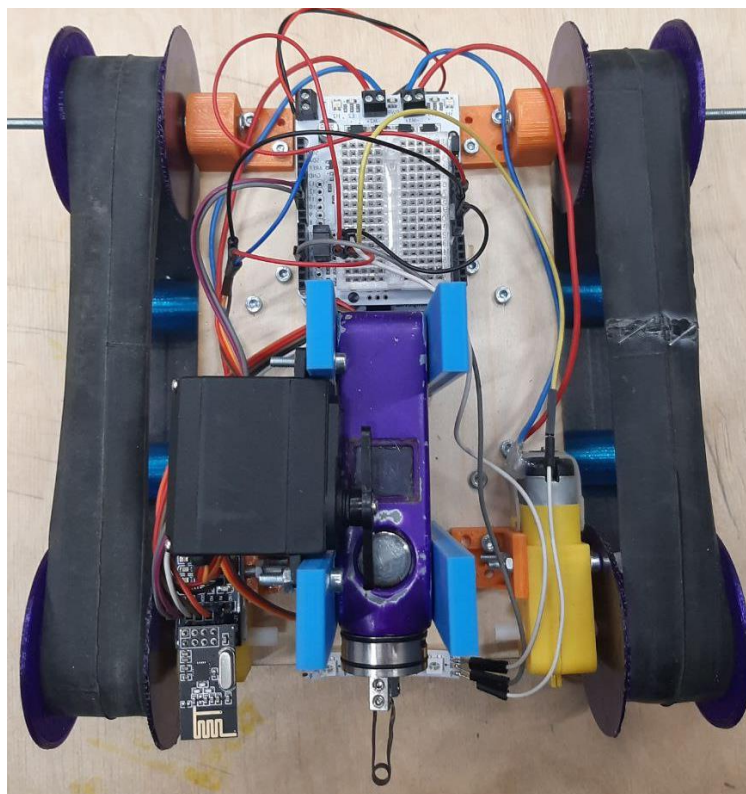


Рисунок 20 – Готовый бот

Схемы подключения электроники

Вся электроника подключалась либо на прямую к плате, либо с использованием платы bradboard. Управляющая плата на боте Arduino Leonardo/Amperka Iskra Neo с Motor Shield, а на пульте управления Arduino Pro Micro. Связь настроена через радио модули NRF и NRF+. Далее будут представлены схемы подключения основных узлов электроники и готовых агрегатов.

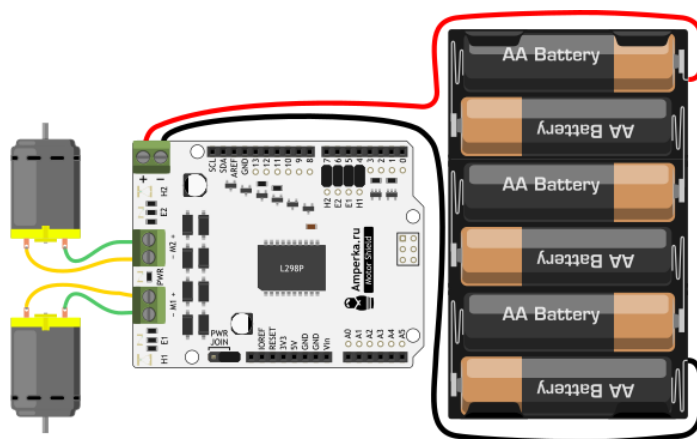


Рисунок 21 – Схема подключения аккумуляторного блока и моторов к Motor Shield

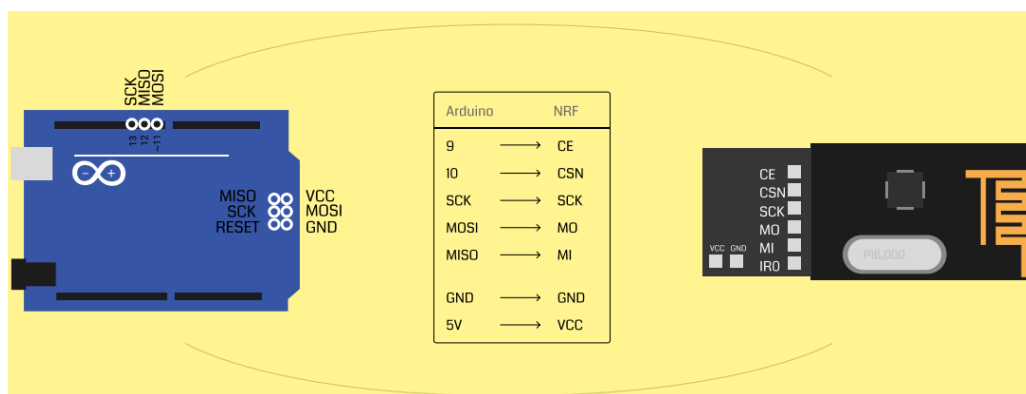


Рисунок 22 – Схема подключения радио модуля к Arduino Leonardo

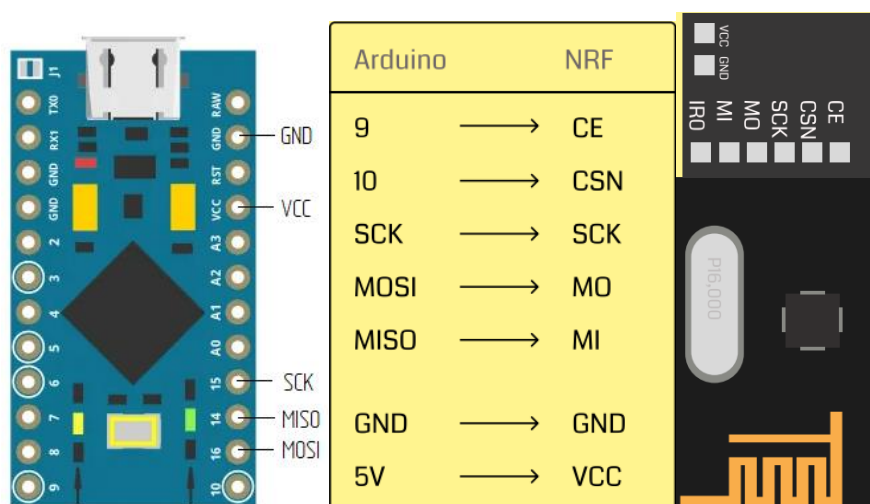


Рисунок 23 – Схема подключения радио модуля к Arduino Pro Micro

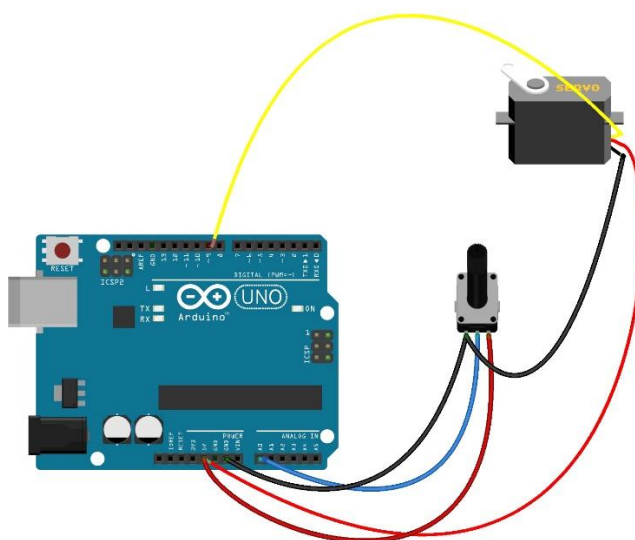


Рисунок 24 – Схема подключения потенциометра с сервоприводом

Код Arduino

Написание кода проводилось в программе Arduino IDE. Необходимо было установить дополнительные библиотеки для управления сервоприводом, светодиодной лентой и радио модулями. На пульте управления считываются входные данные с устройств ввода, создается массив данных и передается на приемник на боте, там из массива достаются данные и распределяются на конкретные агрегаты.

```
1  #include <SPI.h>
2  #include <nRF24L01.h>
3  #include <RF24.h>
4
5  RF24 radio(8, 10);
6  int data[3];
7  #define PIN_VRX A2
8  #define PIN_VRY A3
9
10 void setup() {
11     Serial.begin(9600);
12     radio.begin();
13     radio.setChannel(1);
14     radio.setDataRate(RF24_1MBPS);
15     radio.setPALevel(RF24_PA_HIGH);
16     radio.openWritingPipe(0x2234567890LL);
17     radio.stopListening();
18     pinMode(A1, INPUT);
19 }
20
21 void loop() {
22     int potval = analogRead(A1);
23     radio.write(&data, sizeof(data));
24     delay(50);
25     Serial.println(potval);
26     int valX = analogRead(PIN_VRX);
27     int valY = analogRead(PIN_VRY);
28     Serial.print("X = ");
29     Serial.println(analogRead(PIN_VRX));
30     Serial.print("Y = ");
31     Serial.println(analogRead(PIN_VRY));
32     data[0] = potval;
33     data[1] = valX;
34     data[2] = valY;
35 }
36
```

Рисунок 25 – Код для пульта управления

```

1  #include <SPI.h>
2  #include <nRF24L01.h>
3  #include <RF24.h>
4  #include <Servo.h>
5  Servo servo;
6  #define servoPin 11
7  int data[3];
8  RF24 radio(9, 10);
9  #define SPEED_1 5
10 #define DIR_1 4
11 #define SPEED_2 6
12 #define DIR_2 7
13 #define LED_PIN 3
14 #define LED_NUM 3
15 #include "FastLED.h"
16 CRGB leds[LED_NUM];
17 byte counter;
18 void setup(){
19     Serial.begin(9600);
20     radio.begin();
21     radio.setChannel(1);
22     radio.setDataRate(RF24_1MBPS);
23     radio.setPALevel(RF24_PA_HIGH);
24     radio.openReadingPipe(0, 0x2234567890LL);
25     radio.startListening();
26     servo.attach(servoPin);
27     for (int i = 4; i < 8; i++) {
28         pinMode(i, OUTPUT);
29         FastLED.addLeds<WS2812, LED_PIN, GRB>(leds, LED_NUM);
30         FastLED.setBrightness(200);
31     }
32 }
33 void loop(){
34     if (radio.available()) {
35         radio.read(&data, sizeof(data));
36         Serial.println(String(data[0]) + " "; " + String(data[1]) + " "; " + String(data[2]));
37
38         int potVal = data[0];
39         servo.write(map(potVal, 0, 1023, 0, 180));
40         delay(50);
41         int valX = data[1];
42         int valY = data[2];
43         if (valX < 112) {
44             digitalWrite(DIR_1, LOW);
45             digitalWrite(DIR_2, HIGH);
46             digitalWrite(SPEED_1, 255);
47             digitalWrite(SPEED_2, 255);
48         }
49         else if (valX > 912) {
50             digitalWrite(DIR_1, HIGH);
51             digitalWrite(DIR_2, LOW);
52             digitalWrite(SPEED_1, 255);
53             digitalWrite(SPEED_2, 255);
54         }
55         else if (valY < 112) {
56             digitalWrite(DIR_1, LOW);
57             digitalWrite(DIR_2, LOW);
58             digitalWrite(SPEED_1, 255);
59             digitalWrite(SPEED_2, 255);
60         }
61         else if (valY > 912) {
62             digitalWrite(DIR_1, HIGH);
63             digitalWrite(DIR_2, HIGH);
64             digitalWrite(SPEED_1, 255);
65             digitalWrite(SPEED_2, 255);
66         }
67         else if (112 < valX < 912) {
68             digitalWrite(DIR_1, HIGH);
69             digitalWrite(DIR_2, HIGH);
70             digitalWrite(SPEED_1, 0);
71             digitalWrite(SPEED_2, 0);
72         }
73         else if (112 < valY < 912) {
74             digitalWrite(DIR_1, HIGH);
75             digitalWrite(DIR_2, HIGH);
76             digitalWrite(SPEED_1, 0);
77             digitalWrite(SPEED_2, 0);
78         }
79         for (int i = 0; i < LED_NUM; i++) {
80             leds[i].setHue(counter + i * 255 / LED_NUM);
81         }
82         counter++;
83         FastLED.show();
84     }
}

```

Рисунок 26 – Код для бота

Заключение

По итогу проделанной работы над проектом была достигнута поставленная цель. Были смоделированный и изготовлены компоненты и детали, произведена сборка и отладка робота и пульта управления, подключены электронные компоненты и написан код для управления данными агрегатами.

К сожалению, по завершению прохождения препятствий выяснилось, что конструктивно у робота получился большой вес, в связи с этим моторам не хватало крутящего момента, для передвижения под углом, из-за слабой натяжки гусеницы, не получалось преодолеть ступени для подъема, всё

упиралось в мощность моторов. Также неисправность сервопривода привело плохой работе механизма нажатия кнопки. На полосе было пройдено $1/4$ препятствий и было затрачено 9.45 минут. В дальнейших проектах будут учтены выводы.