

ЛНИП г. Королев

Проектная работа

УНИВЕРСАЛЬНАЯ ПОСАДОЧНАЯ ПЛАТФОРМА ДЛЯ КВАДРОКОПТЕРОВ

Работа выполнена в рамках конкурсного задания на отбор в Артек 2024
«Реализация поставленного кейса на тему БЛА»

Выполнил:
Ученик 10 Б класса Куликов Юрий

Наставник:
Куликов А.П.

2024

Оглавление

0. Введение.....	2
1. Задание 1: Эскизное проектирование.....	4
2. Задание 2: 3D-моделирование.....	5
3. Задание 3: Робототехническая часть.....	9
Определение и разработка механической системы фиксации квадрокоптера на платформе после посадки. Определение необходимых дополнительных модулей и сенсоров с описанием их назначения. Описание мест для их крепления и подключения.....	9
4. Задание 4: Программирование.....	18
5. Задание 5: Производство.....	28
6. Задание 6. Бюджет.....	29
7. Заключение.....	30
8. По итогам анализа разработанного прототипа Платформы были выявлены следующие недостатки:.....	31
9. Эволюция и дальнейшие перспективы разработанной модели Платформы.....	31
10. Список использованных источников.....	31

0. Введение

0.1 Проблема

Посадка БЛА один из самых важных этапов полёта. На этом этапе происходит большинство случаев поломок БЛА, в том числе по следующим причинам:

- неровной поверхности на зоне посадки (склоны, рытвины, высокая трава и другие антропогенные факторы).
- неверное позиционирование зоны посадки оператором, в том числе через на FPV дронах
- неверное позиционирование посадочной зоны из-за плохих погодных условий – дождь, туман, тёмное время суток.

Также одной из востребованных задач после приземления дрона, является задача уменьшения времени «сворачивания» и смены позиции запуска дрона. Эта задача стоит перед обслуживающей дрон командой (расчётом).

0.2 Актуальность

В современных условиях активного использования дронов для решения различных задач, возникает потребность создания приспособления-устройства решить описанную проблему. Автономные взлетно-посадочные платформы для беспилотных летательных аппаратов используются для решения задач различного рода, в том числе для автоматической доставки продуктов, для военных задач, задач мониторинга и орошения в агропромышленном комплексе.

0.3 Цель

Необходимо разработать универсальную посадочную платформу для квадрокоптеров (далее- Платформа), предназначенную для различных применений.

0.4 Задачи

Задание 1. Эскизное проектирование:

Нарисовать эскиз будущей платформы (может быть нарисован с помощью чертёжных принадлежностей на листе бумаги, или графического планшета, или CAD-системы).

Задание 2. 3D-моделирование:

Создать 3D-модель(и) посадочной платформы с учетом габаритов и весовых характеристик квадрокоптеров различных типов и моделей.

Задание 3. Робототехническая часть:

3.1. Разработать механическую систему фиксации квадрокоптера на платформе после посадки.

3.2. Предусмотреть места для крепления и подключения дополнительных модулей и сенсоров, описать их назначение.

Задание 4. Программирование:

Разработать код для управления подсветкой посадочной платформы в зависимости от расстояния до приземляющегося квадрокоптера.

Задание 5. Производство:

Определить необходимые материалы для изготовления посадочной платформы с учетом прочности и легкости.

Задание 6. Бюджет:

Определить бюджет проекта.

0.5 Основная концепция

Основные допущения при создании универсальной посадочной платформы для квадрокоптеров:

- ✓ В части универсальности выбирается определённое семейство дронов размерами от DJI Phantom до агрокоптеров с вместимостью бака до 16л. Для квадрокоптеров большего типоразмера нужна доработка данного решения, с условиями масштабируемости. Для квадрокоптеров меньшего размера актуальность наличия Платформы является низкой. Но и для них может быть разработана уменьшенная версия Платформы.
- ✓ При выборе материалов для изготовления отдаётся предпочтение в сторону деталей изготавливаемых способом 3D печати. Количество деталей из других материалов минимизируется.
- ✓ Материалы из стали не используются из-за сложности их обработки.
- ✓ В части электронной базы и датчиков решается задача баланса между стоимостью и функциональностью. Компонент выбирается за минимальную стоимость с необходимой функциональностью.
- ✓ Также решается задача максимально быстрого создания рабочего прототипа изделия, без выхода на промышленное производство, для отработки работоспособности прототипа
- ✓ Автономность. Предполагается автономное питание и обслуживание-переноска одним человеком.
- ✓ Предполагается использование в закрытых помещениях и на открытом воздухе.
- ✓ Предполагается что Платформа будет эксплуатироваться в температурных режимах от -5 до +40 градусов. При более низких температурах нужно вводить элементы подогрева, смазку и менее хрупкие материалы. При более высоких температурах возможен перегрев электронного блока и нарушение геометрии деталей распечатанных методом 3d печати.

0.6 Требования к оформлению результатов проекта в виде файлов и документов:

1. Документация в формате PDF:

- ✓ В этом файле должны быть описаны этапы выполнения задания.
- ✓ Разработанные эскизы, скриншоты хода выполнения работы, а также фотографии, должны быть приложены в этот файл.
- ✓ В этом файле должен быть представлен весь программный код, который был использован для выполнения задания.

2. 3D-модели в формате STL:

- ✓ Каждая модель должна быть представлена в отдельном файле с соответствующим названием.
- ✓ Файлы должны быть сохранены в формате STL, который является стандартом для обмена 3D-моделями.

1. Задание 1: Эскизное проектирование.

Создание эскиза-чертежа будущей платформы с помощью CAD-системы.

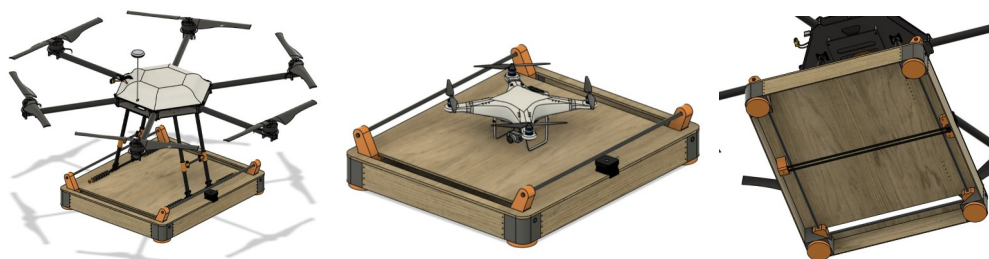


Рис. 1.1. Эскиз Платформы выполненный в Fusion360.

Описание принципов работы Платформы: Платформа устанавливается на поверхность, ножки позволяют производить небольшую регулировку для выравнивания. Включается питание платформы и переходит в режим ожидания прибытия БЛА. В это время для лёгкости обнаружения БЛА Платформа подсвечивается. Платформа оборудована дальномером и в момент обнаружения БЛА, готового для приземления, включается цветная подсветка. После приземления БЛА на платформу включаются механические захваты-толкатели, которые позиционируют и фиксируют БЛА. После нажатия кнопки оператором, захват освобождает БЛА для взлёта или снятия с Платформы. После этого Платформа заново переходит в режим готовности принятия нового БЛА.

Приведенные выше пластиковые детали (имеющие серый и оранжевый цвет на рендерах) были подготовлены к изготовлению на 3D принтере. Для этого модели деталей, разработанные в среде Fusion 360, были экспортированы в формат, использующийся в технологии быстрого прототипирования 3-D печати для хранения трехмерных моделей объектов stl. Полученный файл обрабатывается программой Prusa Slicer, которая и управляет режимом печати модели на принтере Prusa MK3S+. Далее сборка Платформы в формате step была импортирована в программу Компас-3Д для создания чертежа модели по ГОСТу.

Заключение: в ходе выполнения Задания 1 создан полноценный чертеж-эскиз Платформы:

- демонстрирующий идею проекта
- демонстрирующий принцип работы проекта,
- демонстрирующий его элементы
- эскиз образмерен и выполнен в цифровом виде в САД-системе
- чертёж оформлен по ГОСТу

2. Задание 2: 3D-моделирование.

Создание 3D-модель(и) посадочной платформы с учетом габаритов и весовых характеристик квадрокоптеров различных типов и моделей

Проведён анализ текущих видов распространённых квадрокоптеров и видов их формы для разработки универсального захвата (Рис. 2.1 – Рис.2.4). Также дополнительно, вне границы конкурсного задания рассмотрены мультикоптеры (Рис. 2.5)



Рис. 2.1 Квадрокоптеры типа Mavic 2 Mavic 3 Phantom 4.



Рис. 2.2 Матрис, АУТЕЛ, Autel 4T.

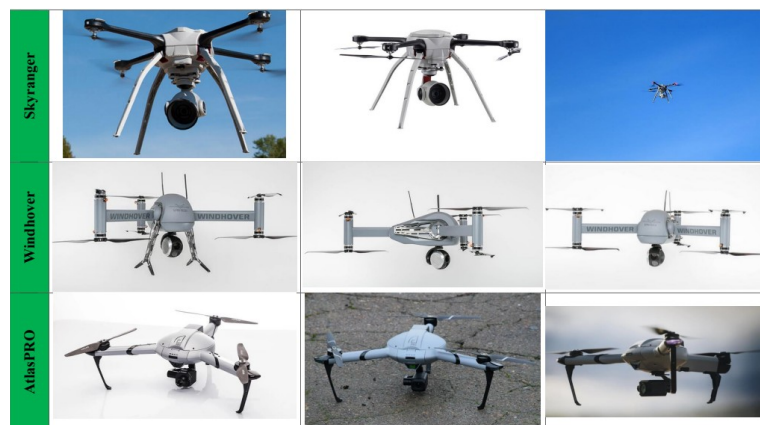


Рис. 2.3 Квадрокоптеры типа Skyranger, WindHover, AtlasPro.



Рис. 2.4 Квадрокоптеры типа Kargu m $\frac{1}{2}$ Skydio X2.

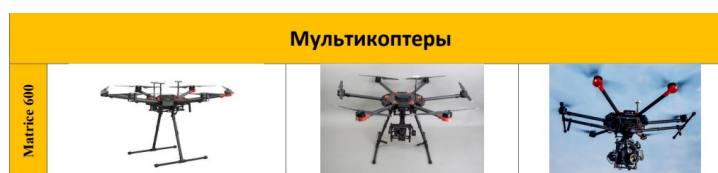


Рис. 2.5 Мультикоптеры Matrice 600.

Были проанализированы текущие решения посадочных платформ и типов захвата. Наиболее полно информация раскрыта статьёй на Хабре. [12]. Наиболее перспективным видом толкателей посчитал W и V образные толкатели. См. Рис. 2.6 Где слева: посадочная платформа с W-образными толкателями для позиционирования: 6 - БПЛА; 401 - толкатели W-образными кромками. Справа: Посадочная платформа с V-образными толкателями для позиционирования: 810 - ножка БПЛА; 220 - толкатели с V-образными кромками.

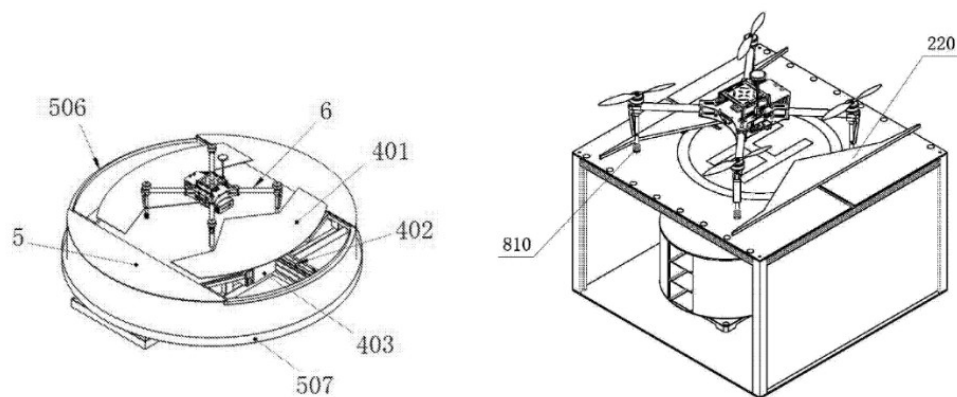


Рис. 2.6 W и V образные толкатели

Но учитывая, что большинство агродронов (см. Рис 2.7) используют опорную конструкцию в виде ползунков, такой вид толкателей не стал выбирать. Так как для ползунков данный вид захвата не эффективен



Рис. 2.7. «Агроскаут» и «Агродрон» от МАИ.

В итоге был выбран захват в виде двух толкателей прямой формы. Параллельные толкатели. Базовый принцип и примеры были изложены в статье на Хабре. [12]. Рис. 2.8. Толкатели в виде планок передвигают БЛА в требуемую зону позиционирования. Наиболее часто используются одна или две пары параллельных друг другу и работающих синхронно толкателей, сдвигающих БЛА за опоры (ножки) к центру посадочной площадки. Данный вид показался мне перспективным с учётом того, что в дальнейшем платформу можно оснастить зарядной станцией. Тогда будет удобно сдвигать БЛА к центру Платформы, где обычно расположены контакты питания. Как дополнительная возможность - толкатели могут содержать элементы крепления с электрическими контактами.



Рис. 2.8. Платформа Ewatt Aerospace. 1 - опора БЛА с электродами; 2 - посадочная площадка; 3 - толкатели с электродами

Процесс 3D моделирования:

Примечание: все операции выполнялись с помощью твердотельного моделирования. Детали серого и оранжевого цвета — пластиковые. Детали с текстурой древесины — из фанеры

-С учётом типоразмеров дронов была смоделирована площадка 650x550мм:



Рис. 2.9. Площадка

В качестве материала для площадки используется фанера 6мм. К модели применен соответствующий материал (текстура)

Толщина фанеры, мм	Вес на 1 кв. метр, кг
3	30-40
4	40-60
6	80-100
8	100-120
12	150-180
15	200-220
18	250-300

Она была выбрана по соотношению цены/прочности/обрабатываемости

- прочностные характеристики фанеры разной толщины

-Следующим шагом были смоделированы каретки хватателей:

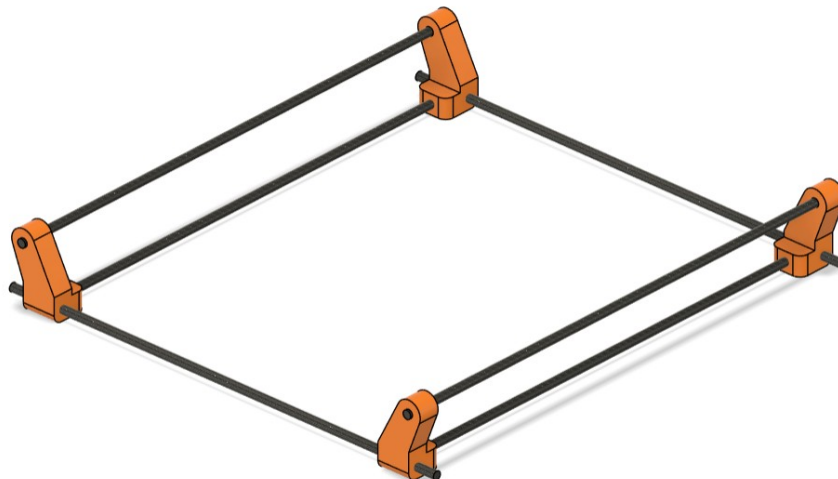


Рис. 2.10. Кинематика хватателей

В качестве направляющих были использованы трубки диаметром 10мм

-Далее были созданы 4 несущих угловых опоры с регулируемыми по высоте ножками

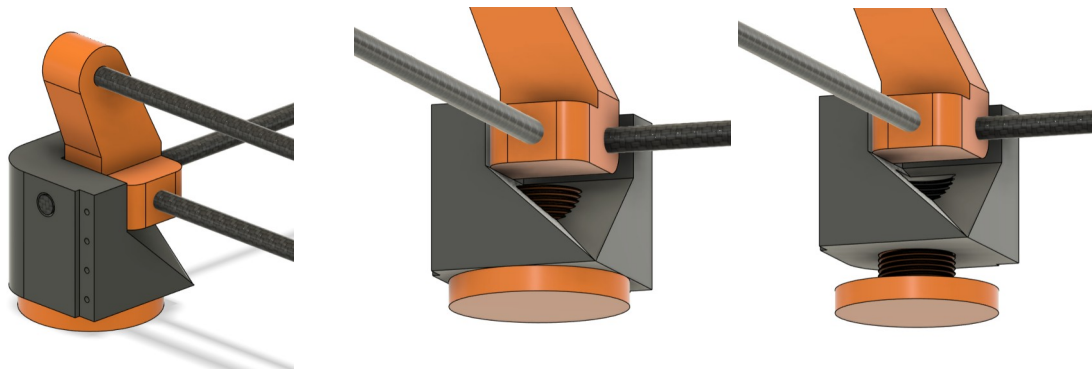


Рис. 2.11. Опора с регулируемой ножкой в 3х видах

Нижняя ножка вкручивается в опору с помощью резьбы. Эти детали с резьбой подготовлены для печати на 3д принтере.

-Следующим было создано крепление шагового двигателя NEMA 17:

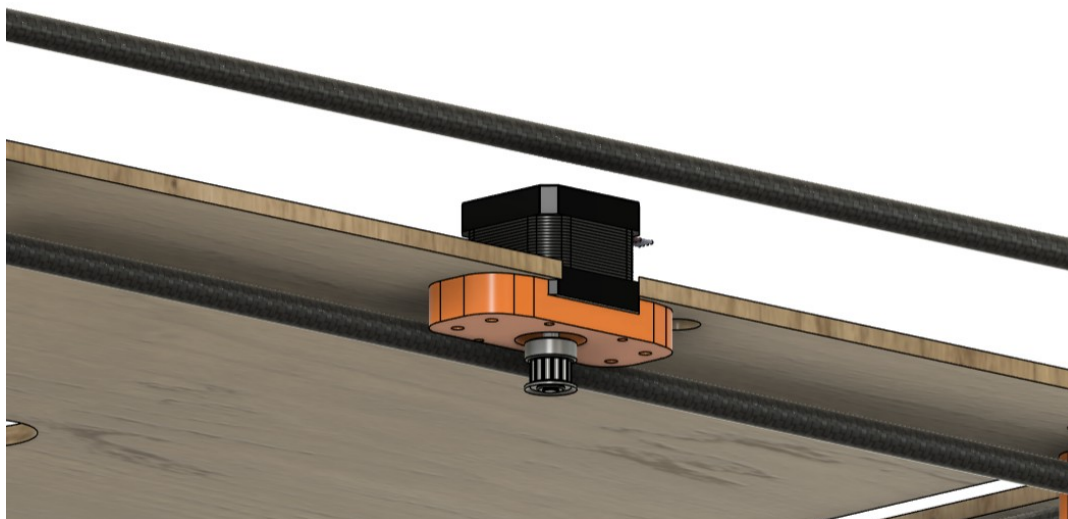


Рис. 2.12. Крепление шагового двигателя

-Смоделирована крепление шпильки для ремня, который приводит в движение толкатели:

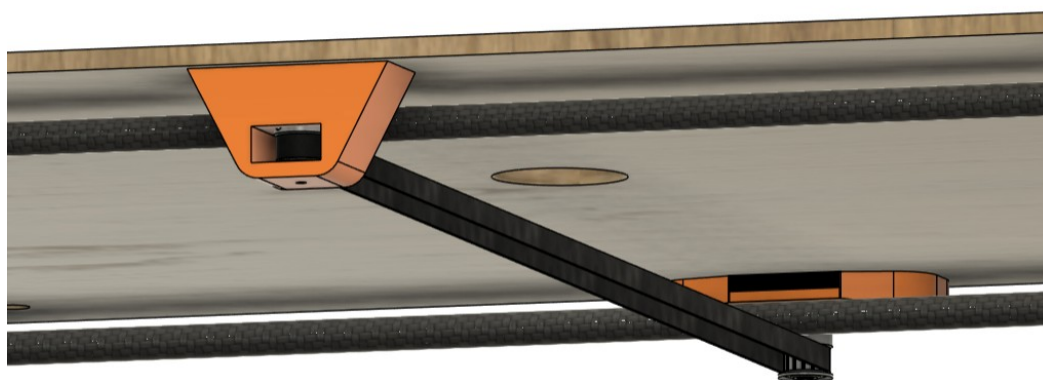


Рис. 2.13. Кронштейн для шпильки

-Далее была дорисована недостающая часть обшивки из 3мм фанеры:

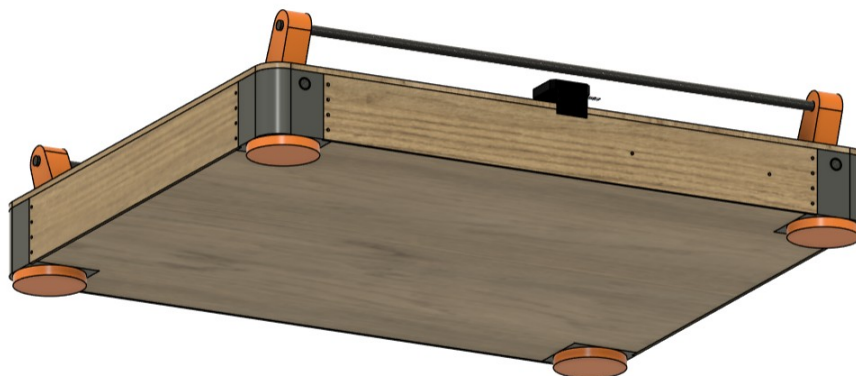


Рис. 2.14. Обшивка фанерой

Она соединяется с корпусом с помощью винтов М3. Необходимые отверстия смоделированы

-Следующим шагом создана коробка для электроники:

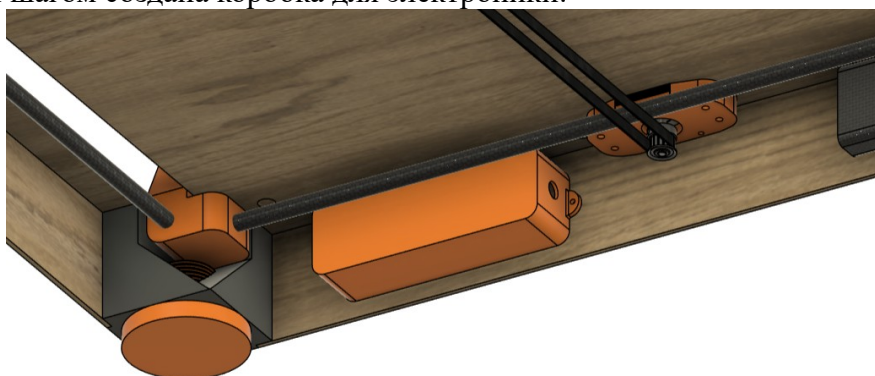


Рис. 2.15. Коробка для электроники

-Финальным шагом добавлены все необходимые отверстия и вырезы, а также место для LED ленты

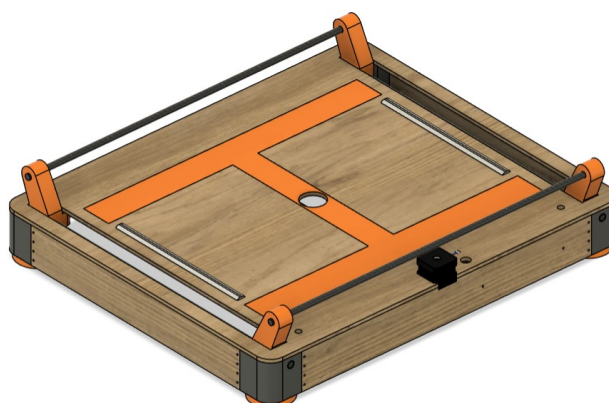


Рис. 2.16. Площадка в сборке

Таким образом была создана сборка всех деталей проекта.
 Все детали для 3D печати (серый/оранжевый) экспортированы в формат .stl
 Все детали для ЧПУ фрезеровки (фанера) экспортированы в формат .dxf

Заключение: в ходе выполнения Задания 2 созданы 3D-модель(и) Платформы:

- с учетом габаритов и весовых характеристик квадрокоптеров различных типов и моделей;
- 3D-модели разработаны с помощью полигонального или твердотельного моделирования;
- наличествует сборка разработанных деталей проекта в единую конструкцию;
- Применены материалы/текстуры, которые используют технические возможностей программ в добавлении текстур или материалов к 3D-моделям/сборке.

3. Задание 3: Робототехническая часть.

Определение и разработка механической системы фиксации квадрокоптера на платформе после посадки. Определение необходимых дополнительных модулей и сенсоров с описанием их назначения. Описание мест для их крепления и подключения.

Выбор микроконтроллера и аргументация выбора:

При выборе основного управляющего элемента были рассмотрены текущие предложения на рынке с учетом стоимости и их распространения. В результате выборки были выбраны две платформы Raspberry и Arduino. После сравнения выбор был остановлен на Arduino:

Arduino - это удобная платформа быстрой разработки электронных устройств и электронный конструктор для новичков и профессионалов. Платформа пользуется огромной популярностью во всем мире благодаря удобству и простоте языка программирования, а также открытой архитектуре и программному коду.

Arduino – комбинация аппаратной и программной частей для простой разработки электроники.

Аппаратная часть включает в себя большое количество видов плат Arduino со встроенными программируемыми микроконтроллерами, а также дополнительные модули.

Программная часть состоит из среды разработки (программы для написания скетчей и прошивки микроконтроллеров Arduino), упрощенного языка программирования, огромного множества готовых функций и библиотек.

Существует несколько версий платформ Arduino: Leonardo, Uno, Nano, Due.

Версия UNO является одной из самых популярных и широко используемой для небольших проектов.

Программирование микроконтроллеров Arduino осуществляется на языке программирования C/C++.

Контроллер Arduino Uno построен на платформе ATmega328, имеющей 14 цифровых входов/выходов (6 из которых могут использоваться как выходы ШИМ), 6 аналоговых входов, кварцевый генератор 16 МГц, разъем USB, силовой разъем, разъем ICSP и кнопку перезагрузки (см. таблицу №1). Для работы необходимо подключить платформу к компьютеру посредством кабеля USB, либо подать питание при помощи адаптера или батареи.

Таблица 3.1 характеристики контроллера

Технические характеристики Arduino Uno Rev 3	
Микроконтроллер	ATmega328P

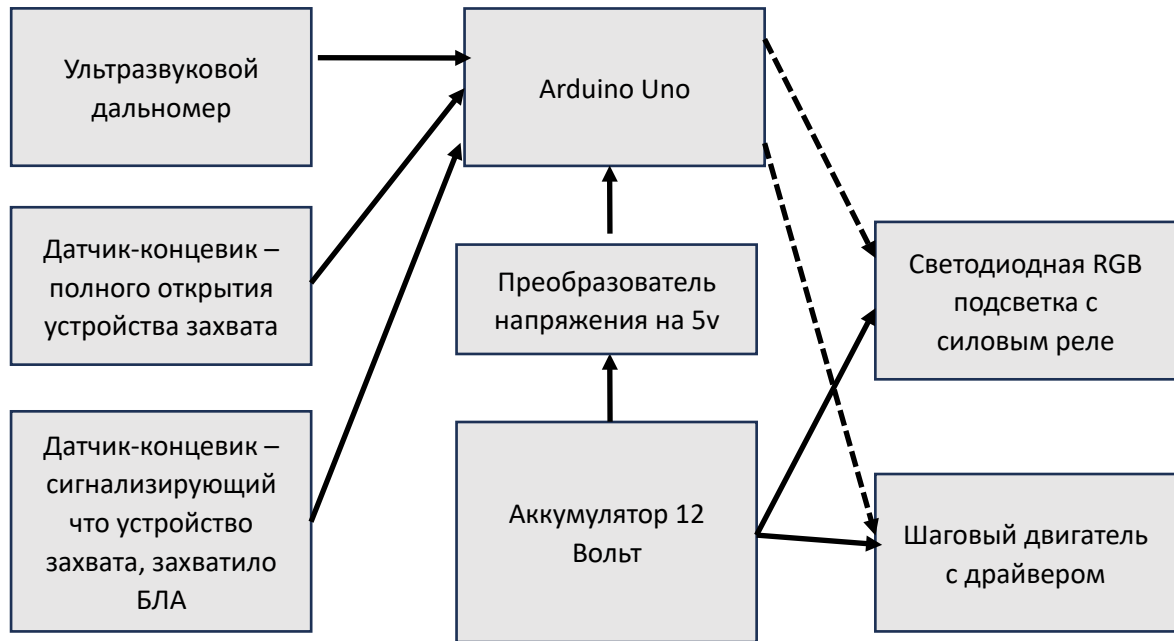


Рис.3.2. Структурная схема электр. элементов Платформы

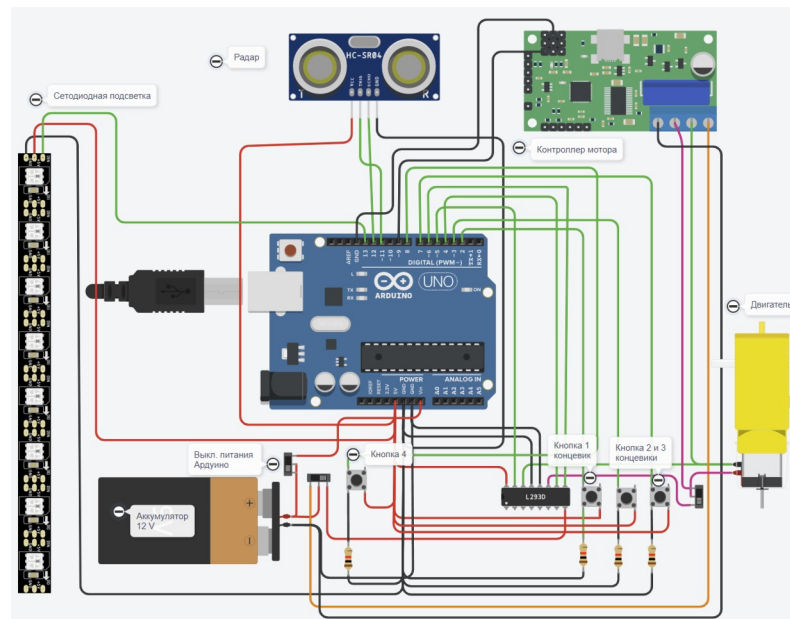


Рис. 3.3 Электронный макет проекта в ПО Tinkercad.

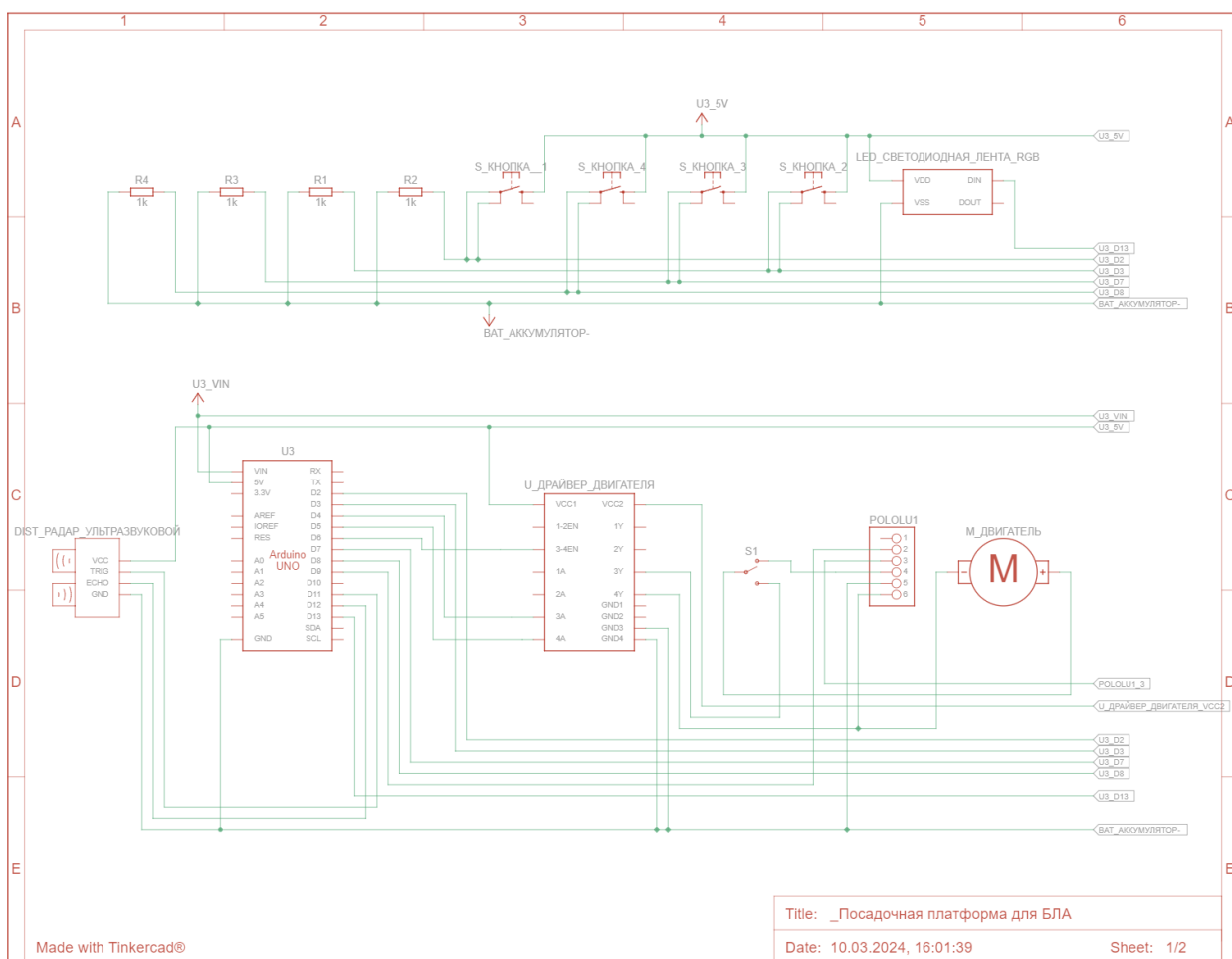


Рис. 3.4 Принципиальная схема устройства Лист 1 из 2.

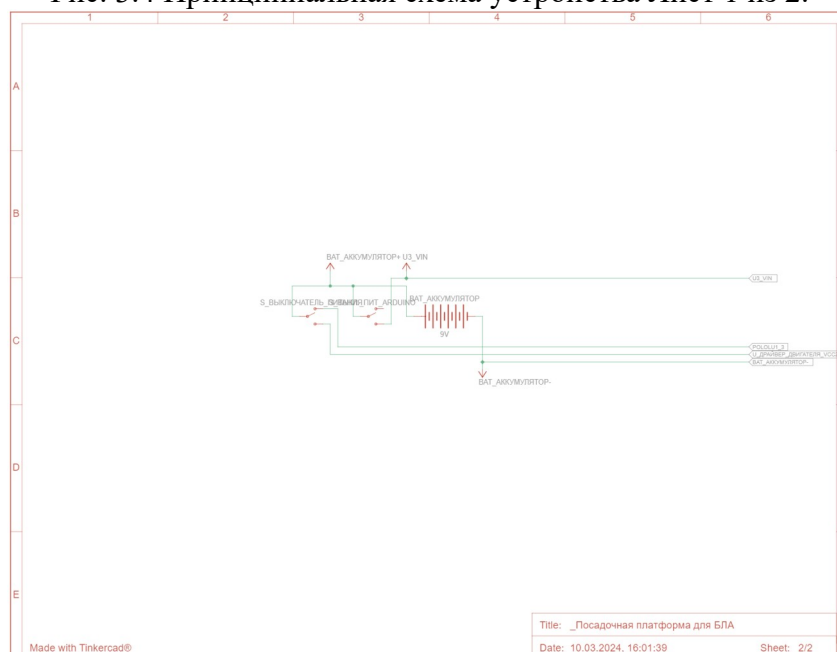


Рис. 3.4 Принципиальная схема устройства Лист 2 из 2.

Выбор датчиков:

Ультразвуковой дальномер HC-SR04

Для решения задачи определения приближающегося БЛА и активизации световой индикации Платформы необходимо выбрать подходящий тип датчика.

Проводили выбор датчика для измерения расстояния до квадрокоптера среди инфракрасных, лазерных, видекамеры.

Была выбран инфракрасный датчик из-за следующих его преимуществ:

- доступная стоимость;
- низкую заметность к обнаружению;
- относительно высокую защищенность от помех, перепадов температуры, влажности воздуха, освещенности.

Среди вариантов для Ардуино был выбран ультразвуковой дальномер HC-SR04

Ультразвуковой датчик способен определять расстояние до объектов от двух до 450 см.

Практические исследования показали, что наибольшая точность достигается при угле меньше 30 градусов. Разрешение 0,3см. Принцип действия заключается в следующем:

датчик отправляет ультразвуковые сигналы, которые отражаясь от объекта, возвращаются обратно. По задержке отраженного сигнала определяется расстояние до объекта. Точность и практичность этого ультразвукового датчика позволяют использовать его в качестве дальномера.

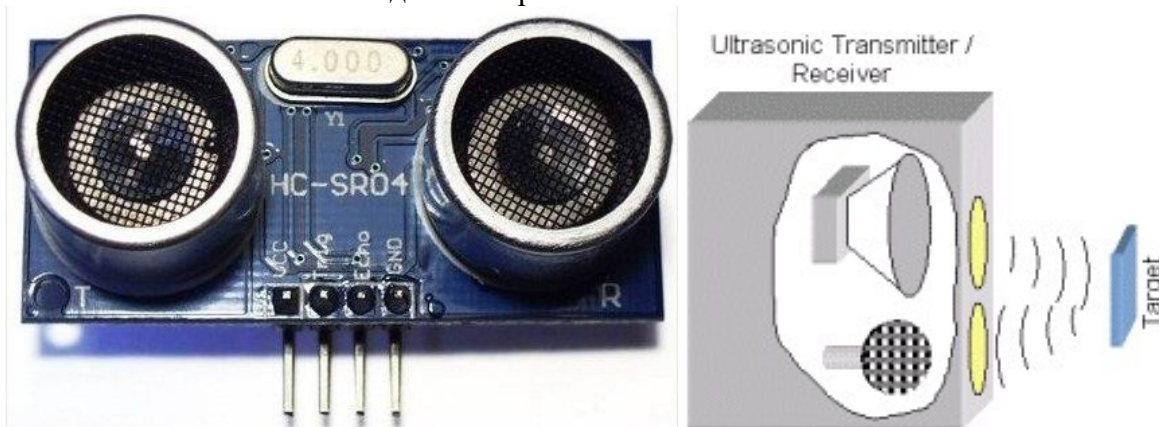


Рис. 3.6 Ультразвуковой дальномер

Разработка электрической схемы с датчиками была смоделирована в виртуальной среде отладки макетных плат Tinkercad [13] (см. рисунок 3.3).

Выбор системы позиционирования и захвата БЛА после посадки на платформу

Сравнивались три вида позиционирования:

- без устройств позиционирования;
- с активными устройствами позиционирования;
- с пассивными устройствами позиционирования;
- с комбинацией устройств позиционирования.

В ходе исследования была выбрана вид посадочной платформы с активными устройствами позиционирования в виде параллельных толкателей. В конструкцию толкателя входят шаговый двигатель и линейные направляющие. Толкатели в виде двух планок передвигают БЛА в требуемую зону позиционирования.

Для программирования системы позиционирования дрона на основе микроконтроллера Arduino Mega использовалась библиотека Stepper, позволившая управлять шаговым двигателем. Библиотека Stepper предоставляет удобный интерфейс управления биполярными и униполярными шаговыми двигателями. В программе применялась команда StepsPerRevolution. Для работы с шаговыми двигателями в библиотеке Stepper была задействована команда вида `stepper.set Speed (200)`. Затем, чтобы двигатель сделал один шаг, можно использовать команду `stepper.step(val)`. Количество шагов, которое должен сделать двигатель, определяется переменной «val». Чтобы шаговый двигатель сделал один шаг по часовой стрелке, необходимо использовать команду `stepper.step(1)`.

[Шаговый двигатель Nema 17, 34 мм, 26 см \(36,8 унций\), а, 12 В, Nema17, шаговый двигатель 42BYGH, 4-проводной ЧПУ Reprap 3D-принтер | AliExpress](#)



Рис. 3.7 Шаговый двигатель Nema

Драйвер управления шаговым двигателем HW-130 motor control shield for Arduino

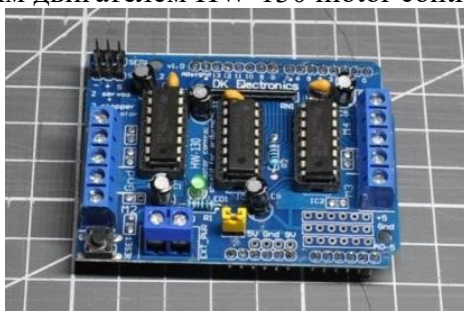


Рис. 3.8 Драйвер управления шаговым двигателем

Кнопка тактовая 12x12 KLS7-TS1204



Рис. 3.9 Кнопка - концевик

Преобразователя напряжения (12->5 В)

На Платформе смонтирован аккумулятор 12 вольт. Он же питает плату Ардуино, где предусмотрен вход Vin с диапазоном от +6V до +12V. См. Рис. 3.1 Поэтому дополнительного понижающего модуля питания не требуется.

LED RGB лента

В качестве источника подсветки была выбрана светодиодная лента.

Каждый светодиодный чип состоит из трех цветов и может изменять яркость в диапазоне от 0 до 255, модули могут объединяться для передачи сигналов по одной линии. Подсветка управляется с Ардуино через 1 управляющий контакт.

Модуль светодиодный адресный на чипах WS2812 5050 RGB 8 светодиодов Neopixel led С напряжением 5В и весом 3 г.



Рис. 3.10 LED RGB лента

Аккумулятор

Используются литиевые аккумуляторы, емкостью 2200 МА\Ч, с напряжением 12 В. Оценочно этого хватает на XXX часа в рабочем режиме. При таких характеристиках аккумулятор обладает малой массой и габаритами. Смена осуществляется оперативно, что помогает быстро менять батареи и не прерывать работу Платформы на длительное время.

Вспомогательных детали: резисторы, макетная плата.

Все эти детали нужны для согласования элементов Платформы и световой индикации работы Платформы

Монтажная плата (1 штука)

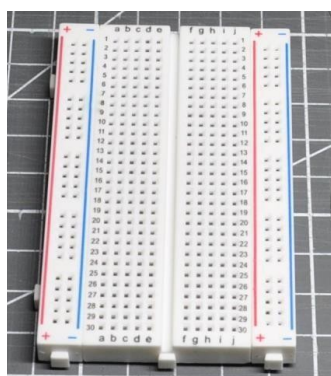


Рис. 3.11 Монтажная плата

Соединительные провода

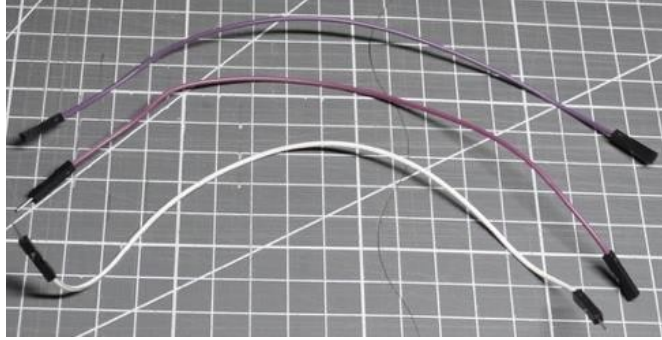


Рис. 3.12 Соединительные провода

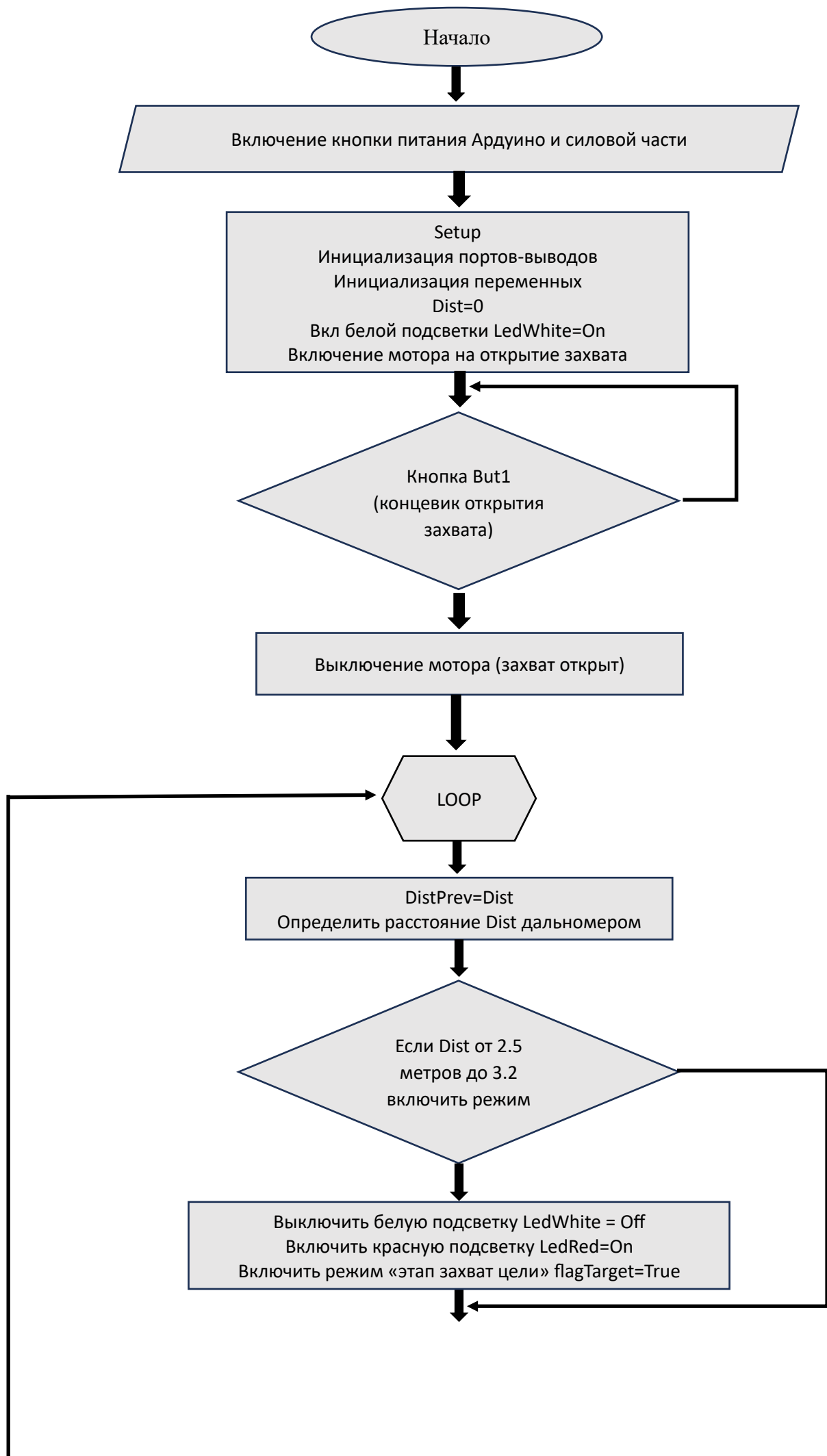
Заключение: в ходе выполнения Задания 3 создана робототехническая часть Платформы:

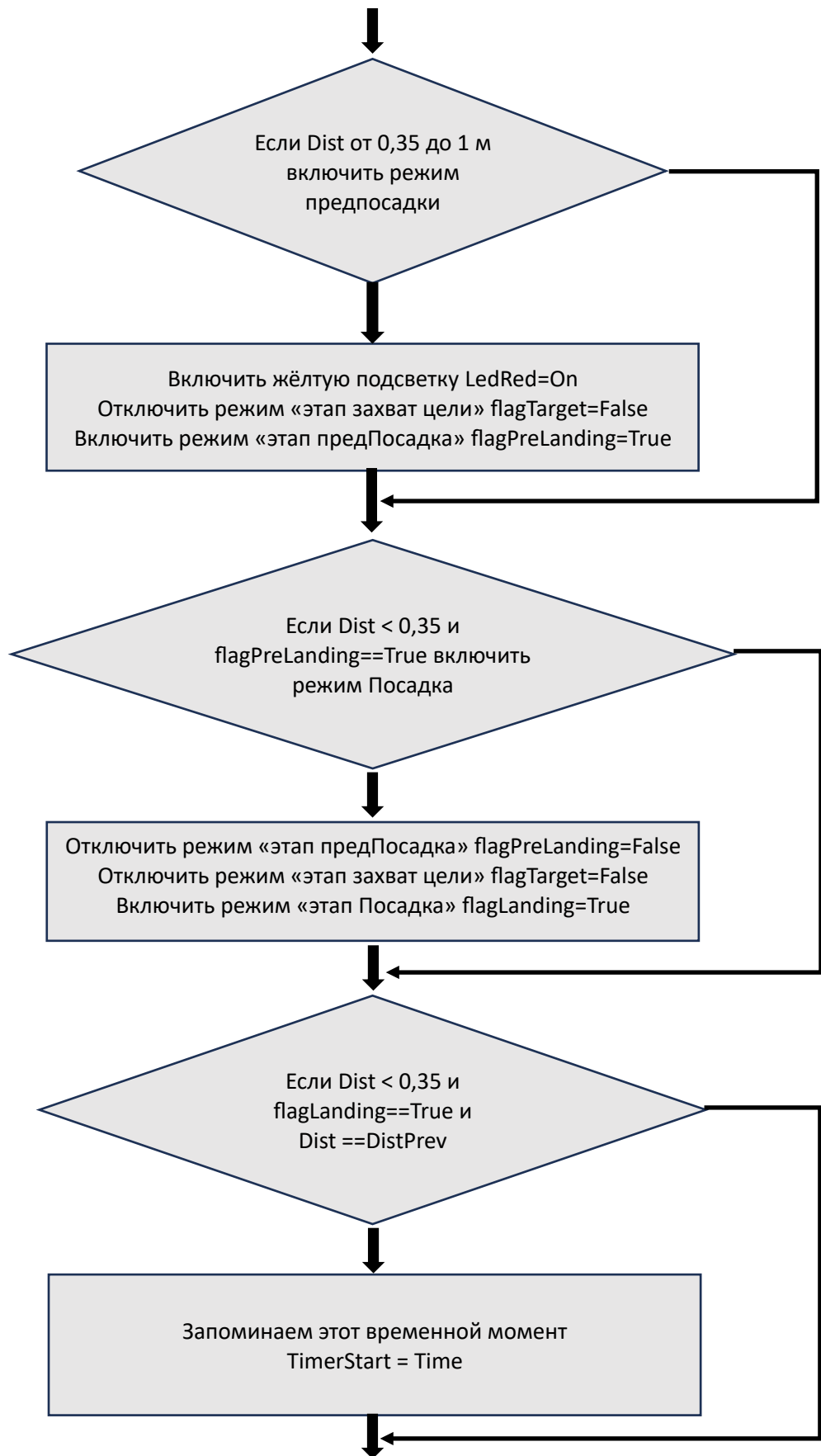
- разработана электрическая схема с применением микроконтроллерной платы Arduino;
- использованы элементы схемотехники;
- применены цифровые датчики;
- разработка выполнена в виртуальной среде.

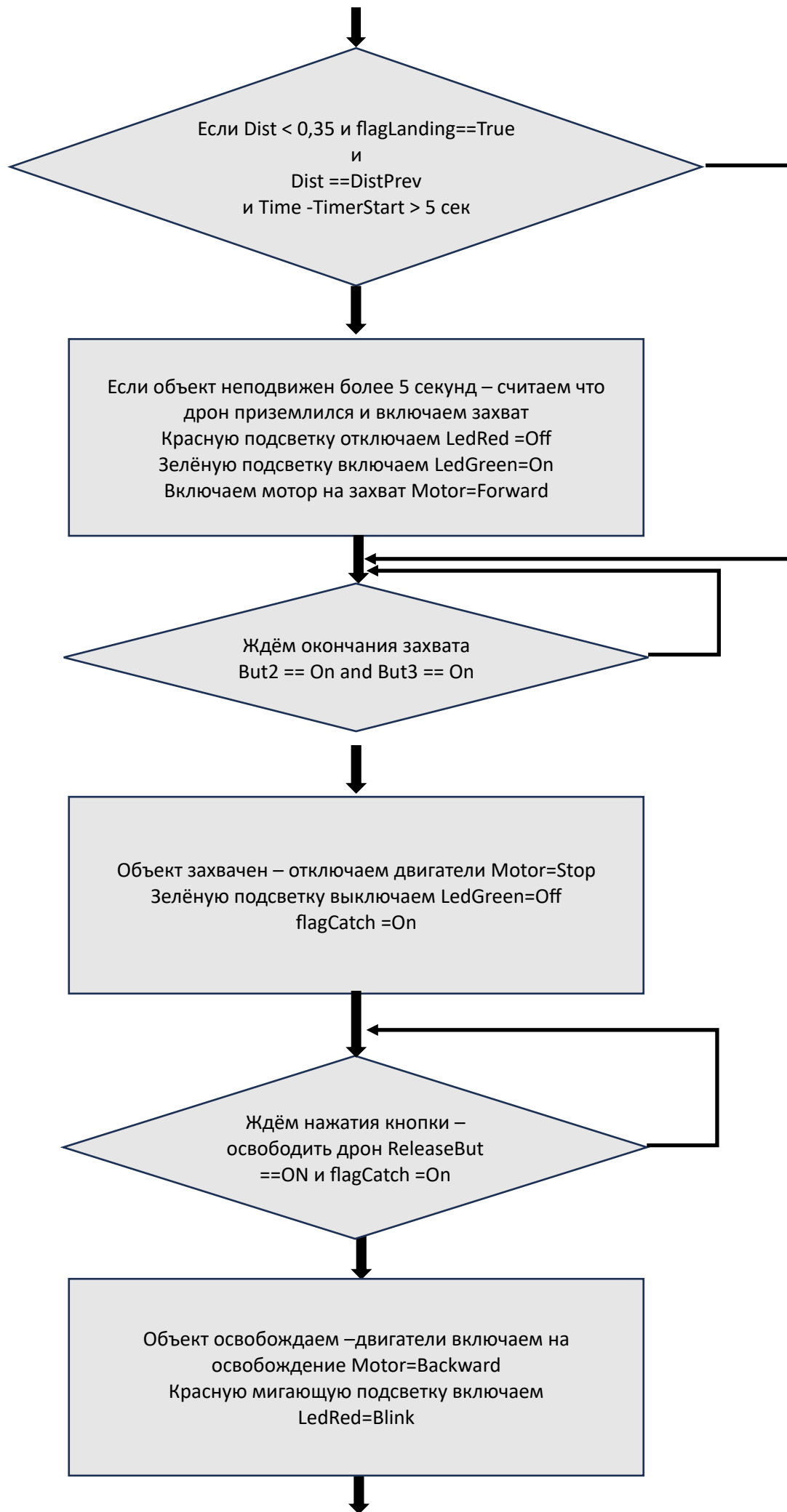
4. Задание 4: Программирование.

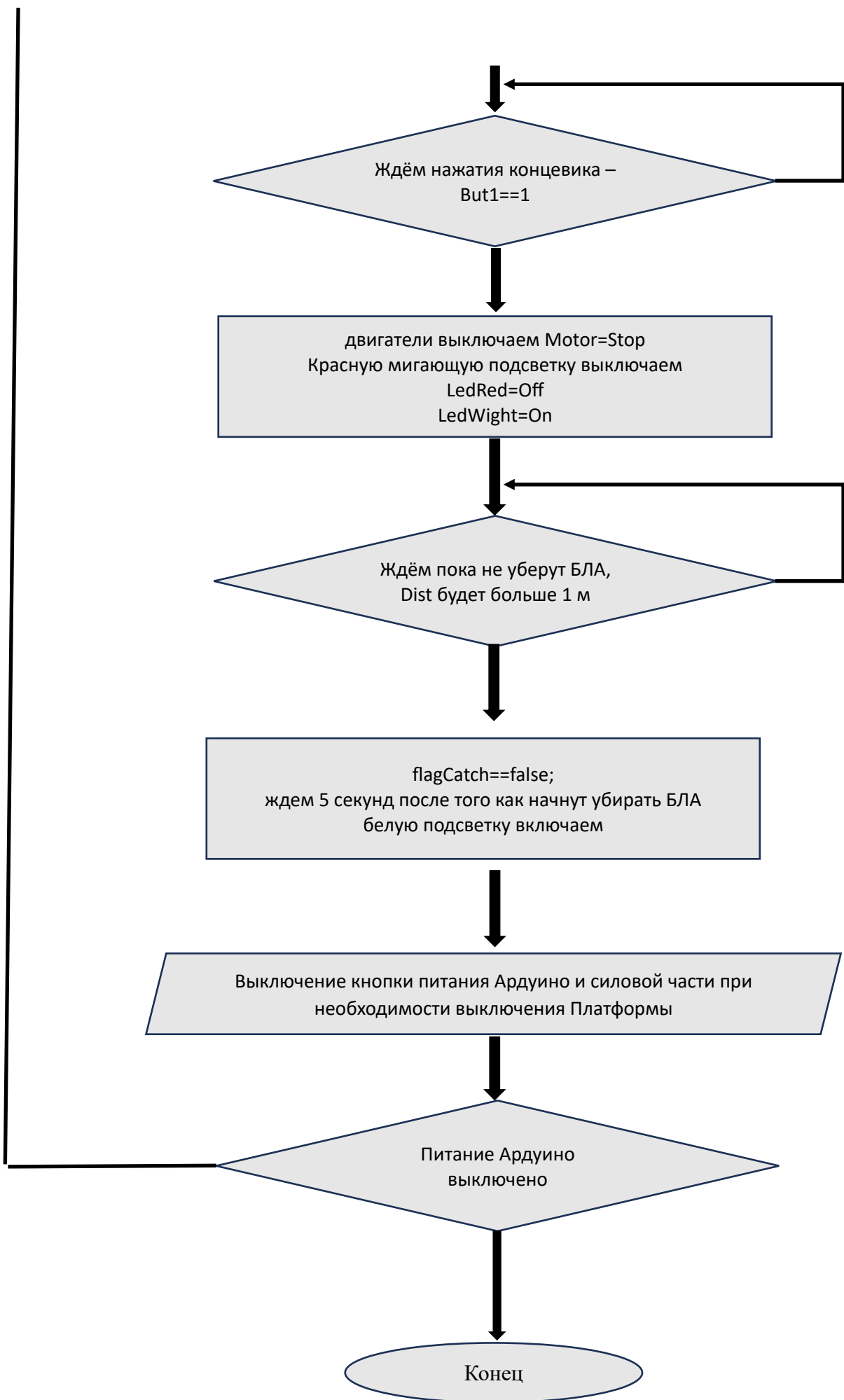
Разработка кода для управления Платформы, включая управление подсветкой посадочной платформы в зависимости от расстояния до приземляющегося квадрокоптера.

Блок - схема алгоритма программы









Код программы автоматического управления Платформой: механизмами захвата и подсветкой.

```
// C++ code
#define __DEBUG 0
#include <Servo.h>

unsigned long TimerStart=millis(); // переменная времени в миллисекундах от начала работы Ардуино

boolean flagStandby = false; //Переменная режима ожидания
boolean flagTarget = false;
boolean flagPreLanding = false;
boolean flagLanding = false;
boolean flagCatch = false;

String strDebug ="";

int Dist = 0; //Переменная для считывания показаний с датчика
int DistPrev = 0; //Переменная для показаний с датчика в предыдущий момент времени

int But1 = 2; // 1-й датчик касания -концевик полного открытия устройства захвата
int But2 = 3; // 2-й датчик касания -концевик срабатывания устройства захвата на 1-й рамке
int But3 = 7; // 3-й датчик касания -концевик срабатывания устройства захвата на 2-й рамке
int But4 = 8; // 4-й датчик касания – кнопки управления – включить движение – освободить захваченный БЛА

// подключение драйвера двигателя для управления двигателем
int IN1Pin = 4; //микросхемы и управления
int IN2Pin = 5; //пины для подключения микросхемы
int Enable1Pin = 6; //обмоткой двигателя

int pinTrig = 11;
int pinEcho = 12;
// функция определения расстояния в сантиметрах датчиком
long getDist() {
    long duration, distance;
    digitalWrite(pinTrig, LOW);
    delayMicroseconds(2);
    digitalWrite(pinTrig, HIGH);
    delayMicroseconds(10);
    digitalWrite(pinTrig, LOW);
    duration=pulseIn(pinEcho, HIGH);
    return (duration/2) / 29.1;
}

//int cm = 0;

#include <Adafruit_NeoPixel.h>
int LedPin = 13; // Вывод управления включением/выключением светодиодами
#define PIN 13 // input pin Neopixel is attached to

#define NUMPIXELS 8 // number of neopixels in strip

Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMPIXELS, LedPin, NEO_GRB + NEO_KHZ800);

int delayval = 100; // timing delay in milliseconds
```

```

int redColor = 0;
int greenColor = 0;
int blueColor = 0;
// setColor()
// picks random values to set for RGB
void setColor(int redColor,int greenColor, int blueColor){
  for (int i=0; i < NUMPIXELS; i++) {
    // pixels.Color takes RGB values, from 0,0,0 up to 255,255,255
    pixels.setPixelColor(i, pixels.Color(redColor, greenColor, blueColor));
    // This sends the updated pixel color to the hardware.
    pixels.show();
    // Delay for a period of time (in milliseconds).
    //delay(delayval);
  }
}

int buttonState = 0;
int ButTest = 8;
Servo motor;
#define PIN_PWM 9
//*****
//метод закрытия захвата
void MotorForward (){
  digitalWrite(Enable1Pin,HIGH);//включаем обмотку двигателя
  //подаем напряжение так, чтобы двигатель вращался в одну сторону
  digitalWrite(IN1Pin,HIGH);
  digitalWrite(IN2Pin,LOW);
  TimerStart =millis(); // Запоминаем этот временной момент
  while(digitalRead(But2)!= HIGH && digitalRead(But3)!= HIGH ){
    //while(digitalRead(But2)!= HIGH ){
  delay(100);
  if ( millis()-TimerStart > 10000){
    break;
  }
  } // ждем пока захват полностью не захватит БЛА и не нажмет обе кнопки концевиков одновременно и
  ставим задержку чтобы избавиться отдребезга кнопки
  digitalWrite(Enable1Pin,LOW); //отключаем двигатель
}
//метод открытия захвата
void MotorBackward (){
  digitalWrite(Enable1Pin,HIGH);//включаем обмотку двигателя
  //подаем напряжение так, чтобы двигатель вращался в обратную сторону
  digitalWrite(IN1Pin,LOW);
  digitalWrite(IN2Pin,HIGH);

  if (__DEBUG !=0) {
    Serial.println(digitalRead(But1));
  }
  while(digitalRead(But1) != HIGH){
    delay(100);
  } // ждем пока захват полностью не откроется и не нажмет кнопку концевика
  digitalWrite(Enable1Pin,LOW); //отключаем двигатель
}

//////////
void setup()
{
  pinMode(But1, INPUT ); //Объявление порта кнопки1
  pinMode(But2, INPUT ); //Объявление порта кнопки2

```

```

pinMode(But3, INPUT ); //Объявление порта кнопки3
pinMode(But4, INPUT ); //Объявление порта кнопки4

pinMode(pinTrig, OUTPUT); //Объявление порта дальномера - источник
pinMode(pinEcho, INPUT); //Объявление порта дальномера – локатор-приёмник

Serial.begin(9600);      // открываем последовательный порт - для вывода отладочной
информации
motor.attach(PIN_PWM);
pixels.begin(); // Initialize the NeoPixel library.
setColor(255,255,244); //включаем белую подсветку

//устанавливаем все пины двигателя как выводы
pinMode(Enable1Pin, OUTPUT);
pinMode(IN1Pin, OUTPUT);
pinMode(IN2Pin, OUTPUT);

digitalWrite(Enable1Pin,LOW); //отключаем обмотку
MotorBackward (); // инициализация - открываем захват

// pinMode(ButTest, INPUT);
// pinMode(LED_BUILTIN, OUTPUT);
}
////////////////////
void loop()
{
  uint32_t sec = millis() / 1000ul;    // полное количество секунд
  int timeHours = (sec / 3600ul);      // часы
  int timeMins = (sec % 3600ul) / 60ul; // минуты
  int timeSecs = (sec % 3600ul) % 60ul; // секунды
  //Serial.print("Time sec:");
  //Serial.print(timeSecs);
  //Serial.println("Loop start ");

  DistPrev=Dist;
  Dist = getDist();
  // Если Dist от 2.5 метров до 3.2 включить режим захват цели
  if (250 <= Dist && Dist <= 320){
    setColor(255,0,0); // Включить красную подсветку LedRed=On
    flagTarget=true; //Включить режим «этап захват цели» flagTarget=True
    strDebug = strDebug + " Mode = flagTarget";
  }
  // Если Dist от 0,35 до 1 м включить режим предПосадка
  if ( 35 <= Dist && Dist <= 100){
    setColor(255,255,0); // Включить желтую подсветку
    flagTarget=false; // Отключить режим «этап захват цели» flagTarget=False
    flagPreLanding=true; //Включить режим «этап предПосадка»
    strDebug = strDebug + " Mode = flagPreLanding";
  }
  // Если Dist < 0,35 и flagPreLanding==True включить режим Посадка
  if (Dist < 35 && flagPreLanding==true){
    //flagPreLanding =false; //
    flagTarget=false;
    flagLanding=true;
    strDebug = strDebug + " Mode = flagLanding";
  }
  // Если Dist < 0,35 и flagLanding==True и Dist ==DistPrev
  if (Dist < 35 && flagLanding==true && Dist ==DistPrev && flagPreLanding ==true){
    TimerStart =millis(); // Запоминаем этот временной момент
  }
}

```



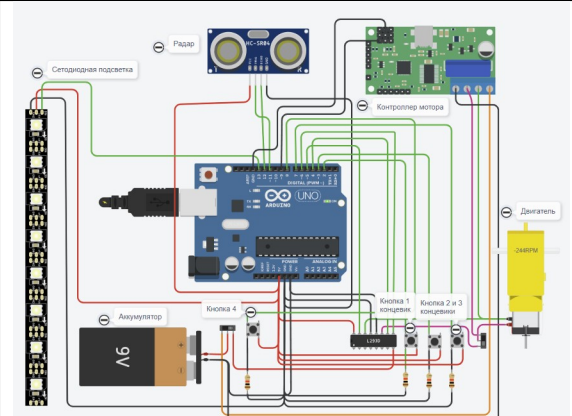
```

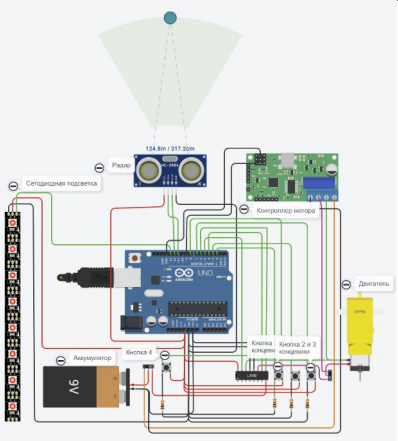
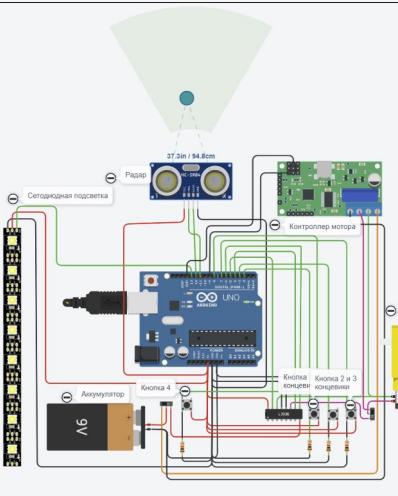
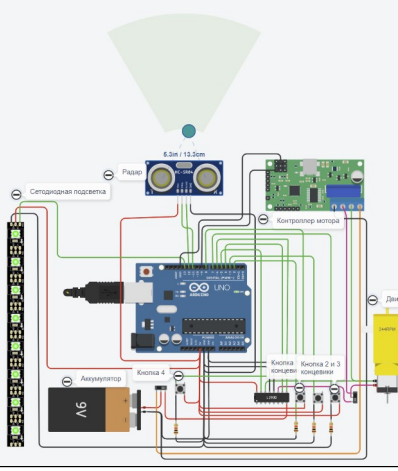
    flagPreLanding =false;
}
// Если Dist < 0,35 и flagLanding==True и Dist ==DistPrev и Time -TimerStart > 5 сек
if (Dist < 35 && flagLanding==true && Dist ==DistPrev && millis()-TimerStart > 5000){
    // Если объект неподвижен более 5 секунд – считаем что дрон приземлился и включаем захват
    setColor(0,255,0); // Зелёную подсветку включаем LedGreen=On
    MotorForward (); // Включаем мотор на захват Motor=Forward
    setColor(0,0,0); //Зелёную подсветку выключаем LedGreen=Off
    flagLanding=false;
    flagPreLanding =false;
    flagCatch = true;
    strDebug = strDebug + " Mode = flagCatch";
}
// Ждём нажатия кнопки – чтобы освободить дрон ReleaseBut ==ON
if (flagCatch==true && digitalRead(But4)== HIGH){
    MotorBackward (); //Объект освобождаем –двигатели включаем на освобождение Motor=Backward
    // ждём пока не уберут БЛА, ждем по безопасности 5 секунд, чтобы уйти в режим ожидания БЛА
    while(getDist() > 100){
        delay(5000);
    }
    flagCatch==false;
    setColor(254,254,254); //белую подсветку включаем
    strDebug = strDebug + " Mode = StandBy";
}
if (strDebug != ""){
    Serial.println (timeSecs + strDebug);
}
strDebug = "";
delay(10); // Delay a little bit to improve simulation performance
}
/***** конец кода программы *****/

```

Демонстрация работоспособности электроники и программных алгоритмов в виртуальной среде Tinkercad представлена в Таблице 4.1

Таблица 4. Фотографии/снимки экрана основных 4-х этапов работы эл. части Платформы

№п/п	Фото	Описание
1		<p>Этап включения питания – инициализации:</p> <ul style="list-style-type: none"> - подсветка светится белым - двигатель работает в обратном направлении –244RPM – захват раскрывается для перехода в режим ожидания

2		<p>Включился режим «этап захват цели» <code>flagTarget=True</code>:</p> <ul style="list-style-type: none"> - радар показывает дальность до БЛА 317,2 см - подсветка светит красным
3		<p>включился режим предПосадка:</p> <ul style="list-style-type: none"> - радар показывает дальность до БЛА 94,8 см - подсветка светит жёлтым
4		<p>Режим захвата БЛА:</p> <ul style="list-style-type: none"> - радар показывает дальность до БЛА 13,3 см и неподвижность БЛА (сел) - подсветка светит зелёным - двигатель работает в прямом направлении 244RPM – толкатели сдвигаются для захвата и удержания БЛА

Заключение: в ходе выполнения Задания 4 разработано программное обеспечение для работы Платформы:

- написан код работы микроконтроллера с взаимодействием с основными элементами схемы, включая управление подсветкой посадочной платформы в зависимости от расстояния до приземляющегося квадрокоптера;
- разработан автоматический алгоритм работы Платформы, включающий программные сценарии.

5. Задание 5: Производство.

Определение необходимых материалов для изготовления посадочной платформы с учетом прочности и легкости.

Подробный перечень необходимых для изготовления Платформы материалов указан в [Таблице 6.1. Стоимость материалов и работ](#).

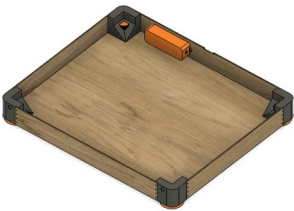
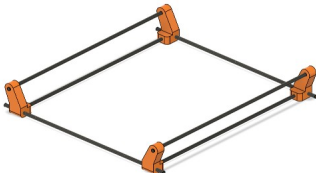



Таблица 5.1. Основные используемые материалы и аргументация их выбора:

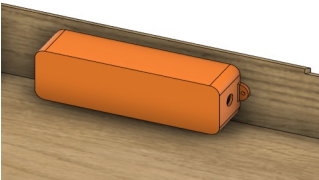
№ п/п	Наименование	Область применения	Аргументация выбора	Фото/эскиз
1	Многослойная фанера 6 мм	используется для изготовления основной площадки -стола Платформы	<ul style="list-style-type: none"> • Выдерживает вес крупных БЛА, • при этом оставаясь легким материалом, • легка в обработке • изготовление через 3d печать неоптимально 	
2	Элементы формируемые 3d печатью PLA пластиком	Держатели, муфты, крепёжные площадки, футляры для электронных компонентов	<ul style="list-style-type: none"> • точность и быстрота изготовления детали сложной формы • уменьшение разнообразия используемых материалов для унификации и организации производства 	
3	Плата Arduino Uno	Управляющий контроллер	<ul style="list-style-type: none"> • недорогая • есть необходимая функциональность 	
4	Электронные компоненты (дальномер)	Датчики и элементы управления	<ul style="list-style-type: none"> • совместимость с Arduino • обладает необходимой функциональностью 	
5	Шаговый двигатель	Кинематика хватателей	обладает достаточным усилием для позиционирования и фиксации	

Сборка Платформы

Таблица 5.2. Полноценная технологическая карта изготовления Платформы с последовательностью этапов изготовления.

№п/п, шаг операции	Описание технологической операции	Эскиз	Материалы, Оборудование, инструменты и приспособления	Время, ч
1	Печать деталей на 3d принтере		Пластик PLA 3d Принтер	3
2	Резка листов многослойной фанеры для изготовления корпуса		Многослойная фанера и электролобзик (пильный инструмент)	0,5
3	В подготовленных листах		Фреза по дереву и	1

	многослойной фанеры делаются необходимы технологические вырезы и отверстия ИЛИ фрезеровка на ЧПУ станке		сверлильный станок (дрель) ИЛИ Лазерный/Фрезерный станок с ЧПУ	
4	Сборка каркаса платформы из деталей изготовленных на шаге 1 и шаге 3		Ручной инструмент и электроинструмент	1.5
4.1	Соединение дна и боковых стенок на винты М3		42 винта М3	0.5
4.2	Сборка кинематики хватателей		-	0.1
4.3	Установка хватателей на верхнюю платформу. Установка креплений для шагового двигателя и ремня		5 винтов М3 4 винта М4 Шпилька М3	0.2
4.4	Установка шагового двигателя. Установка ремня		4 винта М4	0.3
4.5	Установка LED ленты и сборка компонентов, полученных на шаге 4.4 и 4.1 Наклейка оранжевой виниловой пленки в форме буквы Н		4 винта М4	0.4
5	Пайка эл. элементов		Паяльная станция и необходимые компоненты (провода, припой и др.)	1

6	Установка электрических компонентов и их взаимное подключение		2 винта М3	1
7	Программирование Ардуино – заливка ПО		Компьютер, ПО	0,1
8	Тестовое включение, проверка базовой функциональности			0,1

Заключение: в ходе выполнения Задания 5 описан этап производства Платформы:

- описаны выбранные материалы
- аргументирован выбор материалов
- указано необходимое оборудование;
- создана полноценная технологическая карта изготовления Платформы с последовательностью этапов изготовления.

6. Задание 6. Бюджет.

Определение бюджета проекта.

Стоимость материалов и работ по изготовлению 1 шт Платформы

Таблица 6.1. Стоимость материалов и работ

№п/п	Наименование (ссылка на магазин)	Ед.	Кол-во	Цена за 1 ед., руб.	Стоимость, руб
	Материалы				
1	PLA пластик, чёрный	кг	1	1350	1350
2	Многослойная фанера 6 мм 600*600*6	шт	1	1000	1000
3	Arduino Uno R3	шт	1	600	600
4	Комплект проводов, резисторы и разъёмы		1	1000	1000
5	Н-мост Драйвер управления шаговым двигателем motor control shield for Arduino	шт	1	900	900
6	Шаговый двигатель Nema 17, 34 мм, 26 см (36,8 унций), а, 12 В, Nema17, шаговый двигатель 42BYGH, 4-проводной ЧПУ Reprap 3D-принтер AliExpress	шт	1	900	900
7	Ультразвуковой датчик HC-SR04	шт	1	57	57
8	Макетная плата	шт	1	147	147
9	Кнопка тактовая 12x12 KLS7-TS1204	шт	4	5	20
10	Выключатель четырехконтактный KCD1-104	шт	1	35	35
11	Светодиодная лента RGB, 5v Модуль светодиодный Neopixel High Density 8 RGB	шт	1	624	624

	WS2812 5050				
12	Силовое реле - модуль реле: HJR-4102-L-5V Реле шилд 4х канальный	шт	1	504	504
13й	Литиевые аккумуляторы, емкостью 2200 МА\Ч, с напряжением 12 В	шт	1	2300	2300
	<i>Итого материалы</i>				9497
	<i>Работы</i>				
	Стоимость механика по сборке	5	ч	500	2500
	Стоимость работ по монтажу электроники	2	ч	1000	2000
	<i>Итого работы</i>				4500
	Итого				13997

Итого: 14 000 рублей

Заключение: в ходе выполнения Задания 6 рассчитана экономическая часть проекта, с составлением экономической сметы.

7. Заключение

Цель и задачи, поставленные в работе, выполнены. В результате данной работы исследованы варианты построения Платформы и управление её через Arduino.

В процессе разработки Платформы были решены 6 заданий:

- ✓ Эскизное моделирование (чертёж);
- ✓ 3D-моделирование;
- ✓ Робототехническая часть;
- ✓ Программирование;
- ✓ Производство;
- ✓ Бюджет.

В конце работы опробован прототип Платформы.

Таким образом, можно сделать вывод, что разработанная Платформа выполняет заявленную цель проекта и решает поставленные задачи.

Основные технические характеристики разработанной Платформы:

Масса — 5 кг

Габариты:

Длина- 650 мм

Ширина- 550 мм

Высота- 140 мм

8. По итогам анализа разработанного прототипа Платформы были выявлены следующие недостатки:

- ✓ Чувствительность аккумуляторов к холоду (резкое ухудшение характеристик аккумуляторов при низких температурах)
- ✓ Недостаточная защита от влаги. Корпус не герметичен, и при активных осадках в блок электроники может попасть вода и вывести оборудование из строя.

9. Эволюция и дальнейшие перспективы разработанной модели Платформы

Дальнейшее развитие платформы возможно в сторону повышения точности позиционирования и захвата, а также расширения области применений и ограничений по

эксплуатации (влагозащита, температуростойкость, ударопрочность). Потребуется интеграция дополнительных датчиков.

Возможная модернизация:

- ✓ Датчики контакта приземления БЛА
- ✓ Лазерный дальномер
- ✓ Модуль WiFi
- ✓ Радиомаячок
- ✓ Радиомаячок

10. Список использованных источников

Список литературы

1. Сборник лучших работ 2022 года, МГТУ им. Баумана с XXV научно-образовательное соревнования «Шаг в будущее» (сборник лучших работ-2022.pdf)
2. Сборник лучших работ 2023, МГТУ им. Баумана, Центр довузовской подготовки МГТУ им. Баумана, Олимпиада школьников «Шаг в будущее» конференция «Шаг в будущее. космонавтика» XXVI научно-образовательное соревнование молодых исследователей «шаг в будущее» (best_2023.pdf)
3. Сборник, в который вошли лучшие статьи участников Научно- образовательного соревнования молодых исследователей «Шаг в будущее, Москва», проходившего 16-20 марта 2015 года в МГТУ им. Н.Э. Баумана. (bestworks2015_t2.pdf)
4. Сборник, в который вошли лучшие статьи участников Научно- образовательных соревнований молодых исследователей «Шаг в будущее, Москва» и «Шаг в будущее, Космонавтика», проходившего 12-16 марта 2018 года в МГТУ им. Н.Э. Баумана (Сборник лучших работы_2017-2018.pdf)
5. Сборник, в который вошли лучшие статьи участников Научно - образовательного соревнования молодых исследователей «Шаг в будущее», проходившего с 15 по 19 марта 2021 года в МГТУ им. Н.Э. Баумана (Сборник лучших работ-2021.pdf)
6. Проектная работа «Разработка счётчика посетителей» Е.Куреева, 2019 г.

Интернет – сайты:

7. Официальный сайт Ардуино URL: <https://www.arduino.cc> (дата обращения 08.03.2024)
8. Все об Ардуино URL: <https://all-arduino.ru/> (дата обращения 08.03.2024)
9. Подключение ультразвукового датчика к Ардуино URL: <http://роботехника18.рф/ультразвуковой-датчик-к-ардуино/> (дата обращения 08.03.2024)
10. Подключение Arduino, установка драйвера URL: <https://www.youtube.com/watch?v=E5a6iLrxAug> (дата обращения 08.03.2024)
11. Онлайн редактор макетов плат Fritzing URL: <https://fritzing.org/> (дата обращения 08.03.2024)
12. Домики для дронов: какие посадочные платформы БПЛА бывают, что внутри, чем отличаются, кто делает и как выбрать нужную. Frontwise 9 окт 2020 URL: <https://habr.com/ru/companies/innopolis/articles/521948/>
13. <https://www.tinkercad.com/>