Региональная конференция по научно-техническому творчеству школьников

«Лабиринты науки»

Разработка современного и безопасного пешеходного перехода на базе ArduinoUno

Автор проекта: Харитонова Варвара Сергеевна,

11 класс, МОУ «Средняя школа № 18»

Научный руководитель проекта: Шамахов Леонид Михайлович,

Ассистент кафедры «Строительные и Дорожные машины», ЯГТУ

Ярославль, 2023 год

Оглавление

Введение 3

Глава 1. Теоретическая часть 5

1.1. ПДД на пешеходных переходах и стандарты, по которым создаются переходы 5

1.2. Понятие Arduino и обзор требуемой электротехники 6

1.3. Изучение программирования на Arduino 8

1.4. Программа Компас-3D 9

1.5. Правила пайки 9

Глава 2. Создание пешеходного перехода будущего 11

2.1. Разработка программы для пешеходного перехода на базе Arduino Uno 11

2.2. Разработка деталей макета пешеходного перехода в Компас-3D 11

2.3. Создание площадки макета с помощью фрезерного станка с ЧПУ 12

2.4. Сборка макета 12

Заключение 13

Список литературы 14

Приложение 15

Введение

Актуальность темы: безопасное осуществление перехода людей через дорогу до сих пор не выполняется должным образом, так как существует очень печальный процент наезда на пешеходов в зоне специально отведенных для этого мест: пешеходных переходов, он составляет 40% от всех наездов. То есть 40% ДТП с участием пешеходов происходит на пешеходных переходах в связи с различными обстоятельствами. Это может быть как и нарушение ПДД водителем автомобиля, так и пешеходом. Многие пешеходы халатно относятся к переходу дороги и думают, что их все пропустят. Они переходят дорогу на красный сигнал светофора, не дождавшись зеленого, переезжают дорогу на самокатах или велосипедах н красный сигнал светофора, думая, что они быстрее. А на нерегулируемых пешеходных переходах чуть ли не всё, что вздумается. Поэтому нужно не только проводить профилактические беседы с населением, но и улучшить безопасность пешеходных переходов: требуется подавать не только световой и звуковой сигнал, но и ограничение движения пешеходов за счёт шлагбаума, а так же подавать текстовые сообщения

Проблема: малое количество регулируемых пешеходных переходов и недостаточная их безопасность

Цель: разработать программу управления регулируемого пешеходного перехода и реализовать его работу за счёт микроконтроллера ArduinoUno.

Задачи:

- Изучить правила дорожного движения и принцип работы светофора
- Изучить принцип работы программы ArduinoUno и Компас-3D
- Изучить необходимое оборудование
- Реализовать схему подключения на макетной плате
- Написать управляющую программу для микроконтроллера ArduinoUno
- Разработать проект пешеходного перехода в Компас -3D
- Изготовить детали макета на 3D принтере и фрезерном станке с ЧПУ

- Собрать макет пешеходного перехода вместе с электротехническим оборудованием

Объектная область: 3D моделирование, информатика, физика

Объект исследования: программирование и 3D моделирование

Предмет исследования:язык программирования «C++» на базе ArduinoUno**Практическая значимость:** материалы данного проекта могут использоваться для того, чтобы обезопасить переход людей через пешеходный переход

Методы исследования:

- Поиск информации в сети Интернет
- Анализ литературы
- Моделирование
- Тестирование

Глава 1. Теоретическая часть

1.1. ПДД на пешеходных переходах и стандарты, по которым создаются переходы

Рассмотрим правила дорожного движения [1], которые действуют в отношении пешеходных переходов:

1. Водители должны быть готовы остановиться перед пешеходным переходом, как только они заметили пешеходов, собирающихся перейти дорогу. Водители должны также уступить дорогу пешеходам, которые уже начали переходить дорогу.
2. Если на пешеходном переходе установлены светофоры для пешеходов, водители должны следовать сигналам светофора. Если светофор красный, водителям запрещено проезжать пешеходный переход, а если светофор зеленый, они могут продолжить движение.
3. В тех случаях, когда на пешеходном переходе нет светофоров или они не работают, водители должны предоставить пешеходам возможность безопасно перейти дорогу. В особенности следует уступить дорогу детям, пожилым людям, людям с ограниченными возможностями и беременным женщинам.
4. Водители должны учесть особенности пешеходных переходов, которые обозначены знаками дорожного движения. Пешеходные переходы могут быть оснащены подземными или надземными пешеходными переходами, ломаными или прямыми линиями на дороге, а также различными маркировками.
5. Водители должны увеличить осторожность в районах пешеходных переходов, особенно вблизи школ, детских садов, больниц и других мест, где пешеходный поток может быть высоким.

Правила ПДД часто не соблюдаются, не раз все мы замечали, что водители не уступают дорогу пешеходам, а проезжают на скорости около пешеходов,

создавая аварийную ситуацию. Поэтому важно с вниманием переходить дорогу и соблюдать правила, так как если их не будут соблюдать обе стороны, то аварий станет еще больше.

Известно, что практически все области жизнедеятельности человека регламентируется стандартами или как ещё называют: ГОСТам. ГОСТ - это **государственный стандарт**, который включает в себя требования государства к качеству продукции, его геометрические размеры, отклонения от эталона и т.д.

Так же для создания пешеходных переходов существуют определённые стандарты, например:

1. ГОСТ Р 52289-2004 "Улично-дорожные идентификационные и информационные системы. Пешеходные переходы" [2] - определяет основные требования к пешеходным переходам, включая размеры, цветовые решения, организацию освещения и другие характеристики.
2. ГОСТ 32944-2014 "Пешеходные переходы. Пешеходные идентификационные и дорожные знаки" [3] - устанавливает правила размещения дорожной разметки и знаков при обустройстве пешеходных переходов.
3. СНиП 11-02-96 "Обеспечение пешеходного движения" [4] - определяет требования к обустройству пешеходных зон, в том числе пешеходных переходов.
4. ПДД (Правила дорожного движения) - содержат общие требования к организации пешеходных переходов, включая правила приближения и перехода через дорогу для пешеходов.

1.2. Понятие Arduino и обзор необходимой электротехники

Для начала разберемся, что из себя представляет ArduinoUno [6]:

Arduino — это электронный конструктор, который позволяет любому человеку создавать разнообразные электро-механические устройства. Ардуино состоит из программной и аппаратной части. Программная часть

включает в себя среду разработки (программа для написания и отладки прошивок), множество готовых и удобных библиотек, упрощенный язык программирования. Аппаратная часть включает в себя большую линейку микроконтроллеров и готовых модулей для них. Благодаря этому, работать с Arduino очень просто!

С помощью Arduino можно обучаться программированию, электротехнике и механике. Но это не просто обучающий конструктор. На его основе вы сможете сделать действительно полезные устройства. Начиная с простых мигалок, метеостанций, систем автоматизации и заканчивая системой умного дома, ЧПУ станками и беспилотными летательными аппаратами. Возможности не ограничиваются даже вашей фантазией, потому что есть огромное количество инструкций и идей для реализации.

Для того, чтобы Arduino выполняло не только вычислительные работы, но и наглядно показывала выполнение програмного кода ей нужны дополнительные составляющие [6], перечислим те, которые будут задействованы в нашем проекте

1. Светодиоды - это полупроводниковый источник света, который излучает свет при протекании через него тока.
2. Сервоприводы - это электродвигатель с блоком управления, который за счёт обратной связи может точно поддерживать заданное положение вала или постоянную скорость вращения.
3. Динамик - это устройство, преобразующее электрический сигнал в звуковой.
4. Экран (дисплей) - электронное устройство, предназначенное для визуального отображения информации.
5. Резистор – это пассивный элемент электроцепей с постоянным или переменным сопротивлением. (Мы используем резистор на 1 кОм)

6. Кнопка - механическое устройство для передачи сигнала/ввода информации, элемент интерфейса человек-машина: элементарный физический механизм передачи электрического сигнала различным устройствам путём замыкания или размыкания двух или более контактов.
7. Соединительные провода

1.3.Изучение программирования на Arduino

Рассмотрим основные команды [5], которые используются при программировании на ArduinoUno:

#include – команда, которая позволяет подключить к нашему коду дополнительные библиотеки, в которых содержится информация о каких либо операциях

#define – это команда препроцессору заменить один набор символов на другой, то есть присвоить команде свой пин, с которого она будет вызываться

digitalWrite()— это команда, которая устанавливает состояние цифрового вывода пина Arduino в высокое или низкое состояние. Это используется для управления светодиодами, реле, моторами и другими устройствами, которые можно включать и выключать. Когда digitalWrite используется с аргументом HIGH, это устанавливает выход в состояние HIGH (положительное напряжение), когда он используется с аргументом LOW — устанавливает его в LOW (нулевое напряжение).

digitalRead() - считывает значение с заданного входа - HIGH или LOW.Если вход не подключен, то digitalRead может возвращать значения HIGH или LOW случайным образом.

analogRead() - считывает значение с указанного пина

delay() - останавливает выполнение программы на заданное в параметре количество миллисекунд

tone() - генерирует на порту вход/выхода сигнал — прямоугольную "волну", заданной частоты и с 50% рабочим циклом.

voidsetup() – функция, которая вызывается когда скетч начинает исполняться. Используется для определения режимов работы выводов, запуска используемых библиотек, установки первоначальных данных. Выполняется один раз после каждой подачи питания или перезагрузки платы.

void loop() - выполняет последовательный цикл, позволяет активно управлять платой.

1.4.Программа Компас-3D

Компас 3D - семейство систем автоматизированного проектирования, универсальная система автоматизированного проектирования, позволяющая в оперативном режиме выпускать чертежи изделий, схемы, спецификации, таблицы, инструкции, расчётно-пояснительные записки, технические условия, текстовые и прочие документы [7].

Её я буду использовать для создания деталей 3D макета, которые потребуются при сборке пешеходного перехода. Они в последствии будут напечатаны на 3D принтере с помощью вспомогательной программы Ultimaker Cura — это удобный в использовании слайсер, который генерирует G-код для различных моделей 3D-принтеров. Сама печать происходит за счёт пластика. Мы используем пластик PETG— ударопрочный пластик с высокой спекаемостью слоёв, диаметром 1,75 мм, основным его преимуществом является полное отсутствие запаха при печати.

1.5.Правила пайки

Паяние - это процесс соединения двух металлических элементов с помощью расплавленной металлической припойной пасты. Оно широко используется в электронике, ремонте и создании различных устройств.
Вот основные шаги и принципы паяния [8]:
1. Подготовка поверхностей: Прежде чем приступить к паянию, необходимо очистить поверхности элементов, которые нужно соединить. Для этого

использовался флюс - специальное средство для удаления окислов.
2. Подготовка припоя: Припой - это сплав металлов, который плавится при определенной температуре и используется для соединения элементов. Обычно припой поставляется в виде проволоки или пасты. Нужно выбрать припой, подходящий для ваших целей, и правильно подготовить его.
3. Подготовка паяльника: Паяльник - это инструмент, который нагревает припой и позволяет соединить элементы. Паяльник должен быть чистым и нагретым до нужной температуры.
4. Соединение элементов: Элементы выставляются в нужное положение и придерживаются припоем. Когда паяльник достигает нужной температуры, нужно прикоснуться им к припою и к элементам. Припой должен расплавиться и образовать надежное соединение.
5. Охлаждение и проверка: После того как соединение сделано, нужно дать ему остыть. Проверить соединение, и убедиться, что оно крепкое и надежное. Если нужно, удалить излишки припоя с помощью паяльника или специальных инструментов.
6. Термоусадка: После остывания на соединение одевается термоусадочная трубка, принцип работы которой заключается в изменение размеров под воздействием тепла. После нагревания она сжимается и плотно обхватывает паяные соединения, создавая прочное и защищенное соединение. Используется для защиты соединений от внешних воздействий, таких как влага, пыль или коррозия. Она обеспечивает электрическую изоляцию и укрепление соединений.
Важно помнить о безопасности при паянии. Нельзя касаться раскаленной части паяльника руками. Нужно работать в хорошо проветриваемом помещении и использовать защитные очки.

Глава 2. Создание пешеходного перехода будущего

2.1. Разработка программыдля пешеходного перехода на базеArduinoUno

Изучив необходимые материалы, мы переходим к написанию программы для пешеходного переходав программе ArduinoUno. Мы изучили элементы, которые будут задействованы в коде. Нужно написать программу, в которой зеленый сигнал светофора будет включаться по нажатию пешеходами кнопки, которая будет располагаться на светофоре. При нажатии этой кнопки зеленый сигнал светофора для автомобилей должен измениться на красный, а красный сигнал светофора для пешеходов на зеленый. Так же в код нужно добавить, что при загорании зеленого сигнала для пешеходов у нас должен подниматься шлагбаум, который будет открывать проход пешеходам (программа для сервопривода, который отвечает за поднятие и опускание шлагбаума). Добавить включение музыкального сопровождения на время горения зеленого сигнала для пешеходов (программа для динамика). И так же на время всего светофора написать команды для экрана, который на красный сигнал светофора для пешеходов будет выводить «stop», на время движения пешеходов (зеленый сигнал светофора) «gogogo», когда зеленый сигнал будет подходить к концу «potoropis» и опять возвращаться в «stop». Объединив все программы, мы получаем готовую программу для пешеходного перехода.[1]Чтобы убедиться в правильности, мысобрали схему[2] всех соединений на макетной плате и проверяла работу программы.

2.2.Разработка деталей макета пешеходного перехода в Компас-3D

Далее мы переходим к изготовлению деталей для самого макета. Для этого используем программу Компас-3D: начнём со шлагбаума[3], затем перейдём к созданию деталям, к которым будут крепиться сервопривод[4] и экраны[5].Так

[1] См. Приложение [1]
[2] См. Приложений [2]
[3] См. Приложение [3]
[4] См. Приложение [4]

же сделаем деталь самой площадки[6] макета, к которой будут крепиться все оставшиеся элементы и деталь светофор.[7]И чтобы понимать, как будет выглядеть наш итоговый макет, в этой же программе сделаем сборку[8] всего макета.Осталось только с помощью вспомогательной программы Ultimaker Cura подготовить файлы для печати деталей на 3D принтере. Для каждой детали будут использоваться следующие настройки[9]

2.3.Создание площадки макета с помощью фрезерного станка с ЧПУ. Изготовление площадки с помощью 3D печати привело к ненадлежащему виду, мы решили делать её с помощью фрезеровочной машины по дереву. Чтобы с её помощью вырезать площадку нужно написать программу. Рассмотрим основные команды [9]:

1. G0X0Y0Z15 – задает ускоренное перемещение инструмента
2. S700 – задаёт скорость вращения шпинделя
3. M03 – начинаем вращение шпинделя по часовой стрелке
4. G1F50 – задаёт скорость рабочей подачи
5. G0X0Y0 – задаёт начало координат
6. X0Y0Z-0.5 – задаёт глубину прореза
7. M5 – останавливает вращение шпинделя

Программа работает по координатам, поэтому для каждого движения шпинделя нужно задать координаты перемещения. Изучив основные команды,пишем программу[10] для фрезеровочной машины, благодаря которой получим площадку для нашего макета.

2.4. Сборка макета. Получив все необходимые детали начинаем собирать макет. Паяем провода[11], удлиняем их, чтобы схему подключения реализовать на макете.Спаяв все провода, приступаем к сборке сначала элементов по-отдельности, а потом собираем всё воедино и получаем готовый макет[12]

---

[5] См.Приложение [5]
[6] См. Приложение [6]
[7] См. Приложение [7]
[8] См. Приложение [8]
[9] См. Приложение [9]
[10] См. Приложение [10]
[11] См. Приложение [11]
[12] См. Приложение [12]

Заключение

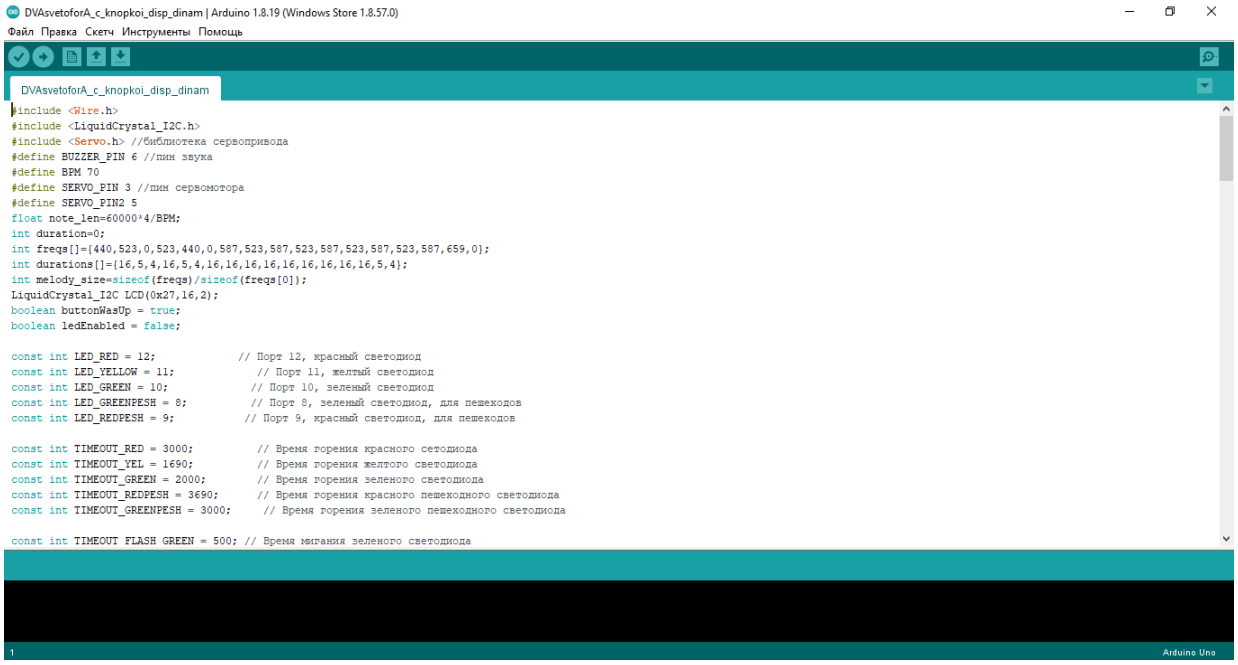
Работая над этим проектом, мне удалось разработать управляющую программу современного и безопасного пешеходного перехода и реализовать схему подключения на изготовленном макете. Все задачи, которые были поставлены мной в начале работы, успешно выполнены, я изучила необходимую теорию, написала программу, разработала макет и реализовала схему подключения оборудования. Разработанный проект отличается от существующих аналогов тем, что звуковое сопровождение является более приятным, текстовая трансляция требуемых действий выводится на экран, но самое главное – это шлагбаумы, не дающие пешеходам выбежать на проезжую часть на красный сигнал светофора. Вместе с тем я научилась программировать ArduinoUno на упрощённом C++, паять электронные компоненты, создавать детали в Компас-3D, пользоваться программой Ultimaker Cura для 3D печати и так же основам написания программы для фрзерного станка с ЧПУ. В ходе изучения и испытания на себе всех этих элементов проекта я могу с уверенностью сказать, что цель моего проекта достигнута. Я написала программу для пешеходного перехода будущего и реализовала схему его подключения с помощью микроконтроллера ArduinoUno.

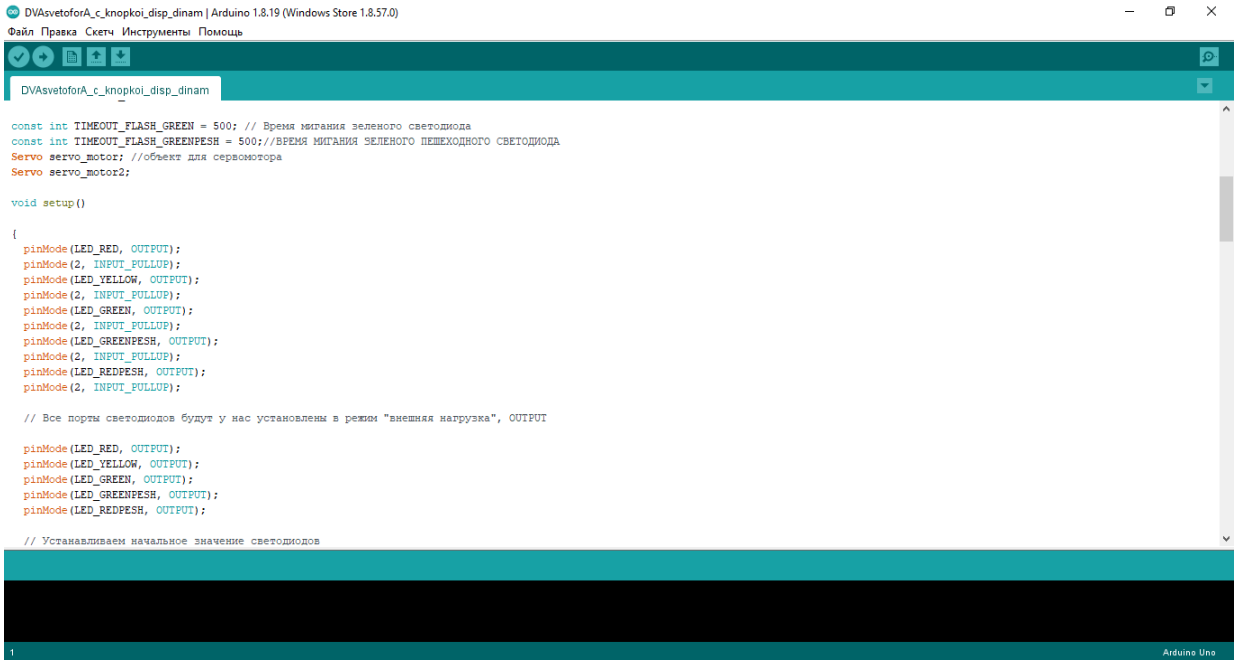
Список литературы

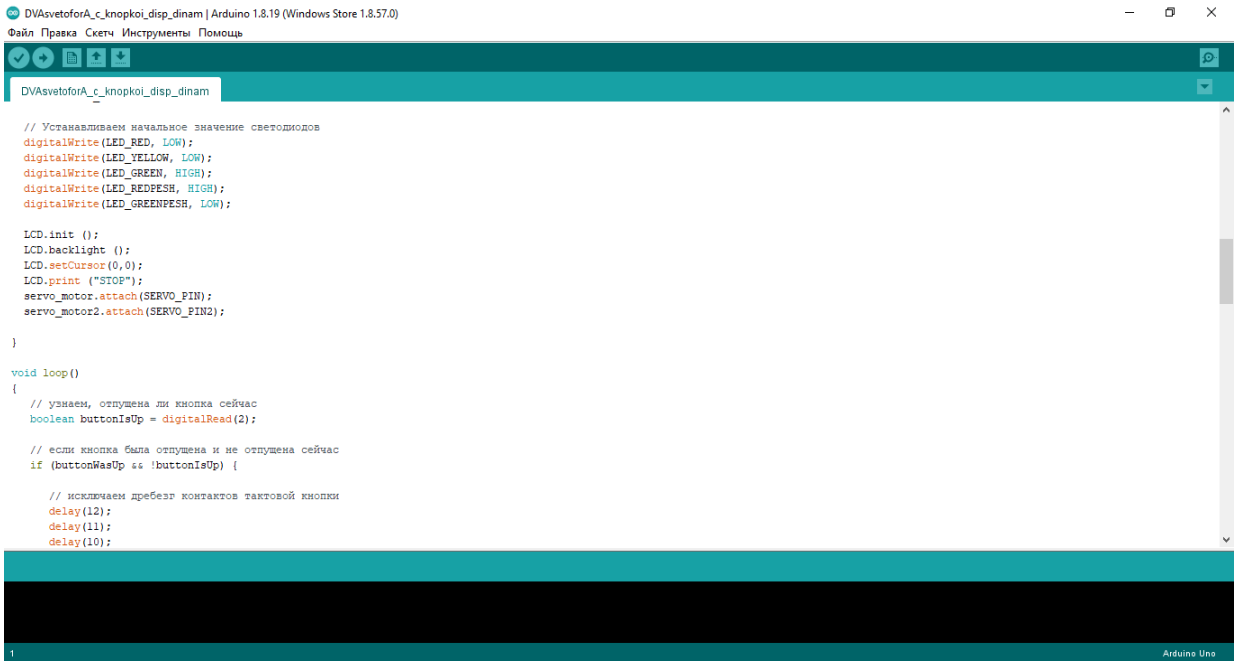
1. ПДД на пешеходных переходах [Электронный ресурс]- Режим доступа: https://www.consultant.ru/document/cons_doc_LAW_2709/7c45508e360f5b7b8ae1443d73feb01f52a6199d/
2. ГОСТ Р 52289-2004 “Правила применения дорожных знаков, разметки, светофоров, дорожных ограждений и направляющих устройств” [Электронный ресурс]- Режим доступа: https://docs.cntd.ru/document/1200038798
3. ГОСТ 32944-2014 "Пешеходные переходы. Пешеходные идентификационные и дорожные знаки" [Электронный ресурс]- Режим доступа: https://gostassistent.ru/doc/9b6cf695-b8ad-4687-b312-2c7f6d58937c
4. СНиП 11-02-96 "Обеспечение пешеходного движения" [Электронный ресурс]- Режим доступа: https://geo-rf.ru/wp-content/uploads/2020/03/snip11-02-96.pdf
5. Программирование на Arduino. [Электронный ресурс]- Режим доступа: https://arduino.ru/
6. Arduino и необходимая электротехника [Электронный ресурс]- Режим доступа: https://electrik.info/arduinomanual.pdf
7. Компас-3D [Электронный ресурс]- Режим доступа: https://kompas.ru/kompas-3d/about/
8. Правила пайки [Электронный ресурс]- Режим доступа: https://www.asutpp.ru/kak-pravilno-payat.html
9. Команды для фрезеровочной машины [Электронный ресурс]- Режим доступа: https://3d-stanki.ru/spravochnik/programmnoe-obespechenie-dlya-stankov-s-chpu/opisanie-g-i-m-kodov-dlya-programmirovaniya-chpu-cnc-stankov-2/

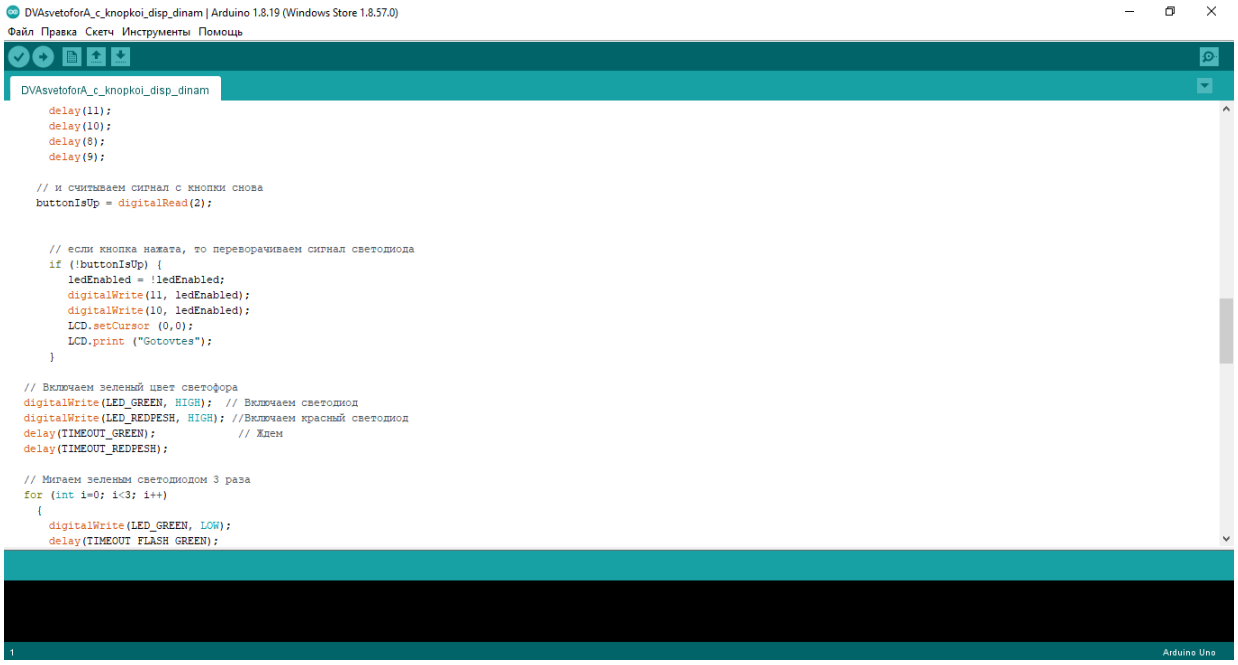
Приложение

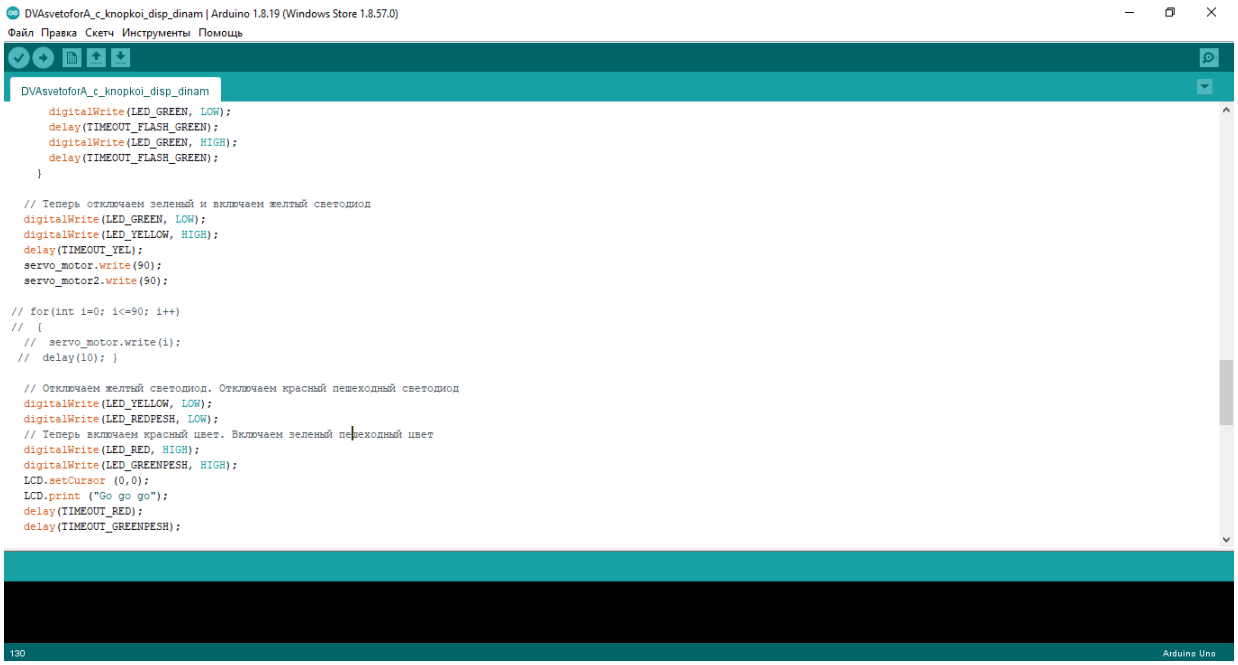
Приложение [1]

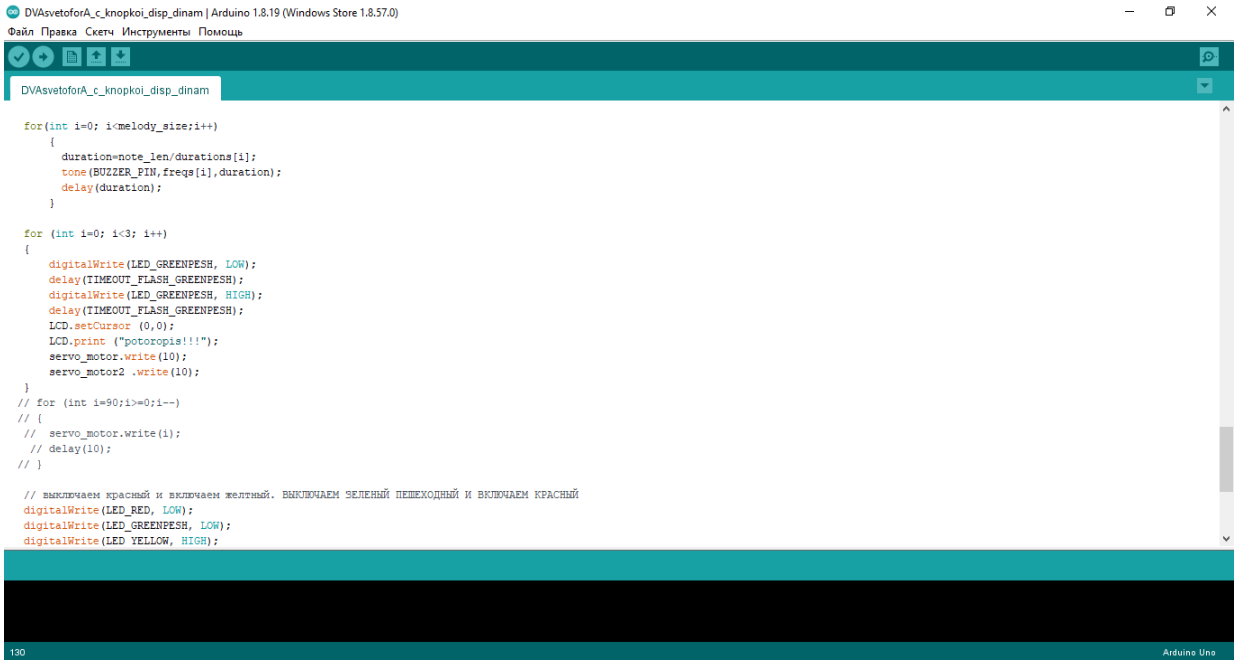





DVAsvetoforA_c_knopkoi_disp_dinam | Arduino 1.8.19 (Windows Store 1.8.57.0)
Файл Правка Скетч Инструменты Помощь
DVAsvetoforA_c_knopkoi_disp_dinam
// Устанавливаем начальное значение светодиодов
digitalWrite(LED_RED, LOW);
digitalWrite(LED_YELLOW, LOW);
digitalWrite(LED_GREEN, HIGH);
digitalWrite(LED_REDPESH, HIGH);
digitalWrite(LED_GREENPESH, LOW);
LCD.init ();
LCD.backlight ();
LCD.setCursor(0,0);
LCD.print ("STOP");
servo_motor.attach(SERVO_PIN);
servo_motor2.attach(SERVO_PIN2);
}
void loop()
{
// узнаем, отпущена ли кнопка сейчас
boolean buttonIsUp = digitalRead(2);
// если кнопка была отпущена и не отпущена сейчас
if (buttonWasUp && !buttonIsUp) {
// исключаем дребезг контактов тактовой кнопки
delay(12);
delay(11);
delay(10);
1
Arduino Uno


DVAsvetoforA_c_knopkoi_disp_dinam | Arduino 1.8.19 (Windows Store 1.8.57.0)
Файл Правка Скетч Инструменты Помощь
DVAsvetoforA_c_knopkoi_disp_dinam
delay(11);
delay(10);
delay(8);
delay(9);
// и считываем сигнал с кнопки снова
buttonIsUp = digitalRead(2);
// если кнопка нажата, то переворачиваем сигнал светодиода
if (!buttonIsUp) {
ledEnabled = !ledEnabled;
digitalWrite(11, ledEnabled);
digitalWrite(10, ledEnabled);
LCD.setCursor (0,0);
LCD.print ("Gotovtes");
}
// Включаем зеленый цвет светофора
digitalWrite(LED_GREEN, HIGH); // Включаем светодиод
digitalWrite(LED_REDPESH, HIGH); //Включаем красный светодиод
delay(TIMEOUT_GREEN); // Ждем
delay(TIMEOUT_REDPESH);
// Мигаем зеленым светодиодом 3 раза
for (int i=0; i<3; i++)
{
digitalWrite(LED_GREEN, LOW);
delay(TIMEOUT_FLASH_GREEN);
1
Arduino Uno


DVAsvetoforA_c_knopkoi_disp_dinam | Arduino 1.8.19 (Windows Store 1.8.57.0)
Файл Правка Скетч Инструменты Помощь
DVAsvetoforA_c_knopkoi_disp_dinam
digitalWrite(LED_GREEN, LOW);
delay(TIMEOUT_FLASH_GREEN);
digitalWrite(LED_GREEN, HIGH);
delay(TIMEOUT_FLASH_GREEN);
}
// Теперь отключаем зеленый и включаем желтый светодиод
digitalWrite(LED_GREEN, LOW);
digitalWrite(LED_YELLOW, HIGH);
delay(TIMEOUT_YEL);
servo_motor.write(90);
servo_motor2.write(90);
// for(int i=0; i<=90; i++)
// {
// servo_motor.write(i);
// delay(10); }
// Отключаем желтый светодиод. Отключаем красный пешеходный светодиод
digitalWrite(LED_YELLOW, LOW);
digitalWrite(LED_REDPESH, LOW);
// Теперь включаем красный цвет. Включаем зеленый пешеходный цвет
digitalWrite(LED_RED, HIGH);
digitalWrite(LED_GREENPESH, HIGH);
LCD.setCursor (0,0);
LCD.print ("Go go go");
delay(TIMEOUT_RED);
delay(TIMEOUT_GREENPESH);
130
Arduino Uno


DVAsvetoforA_c_knopkoi_disp_dinam | Arduino 1.8.19 (Windows Store 1.8.57.0)
Файл Правка Скетч Инструменты Помощь
DVAsvetoforA_c_knopkoi_disp_dinam
for(int i=0; i<melody_size;i++)
{
duration=note_len/durations[i];
tone(BUZZER_PIN,freqs[i],duration);
delay(duration);
}
for (int i=0; i<3; i++)
{
digitalWrite(LED_GREENPESH, LOW);
delay(TIMEOUT_FLASH_GREENPESH);
digitalWrite(LED_GREENPESH, HIGH);
delay(TIMEOUT_FLASH_GREENPESH);
LCD.setCursor (0,0);
LCD.print ("potoropis!!!");
servo_motor.write(10);
servo_motor2 .write(10);
}
// for (int i=90;i>=0;i--)
// {
// servo_motor.write(i);
// delay(10);
// }
// выключаем красный и включаем желтный. ВЫКЛЮЧАЕМ ЗЕЛЕНЫЙ ПЕШЕХОДНЫЙ И ВКЛЮЧАЕМ КРАСНЫЙ
digitalWrite(LED_RED, LOW);
digitalWrite(LED_GREENPESH, LOW);
digitalWrite(LED_YELLOW, HIGH);
130
Arduino Uno

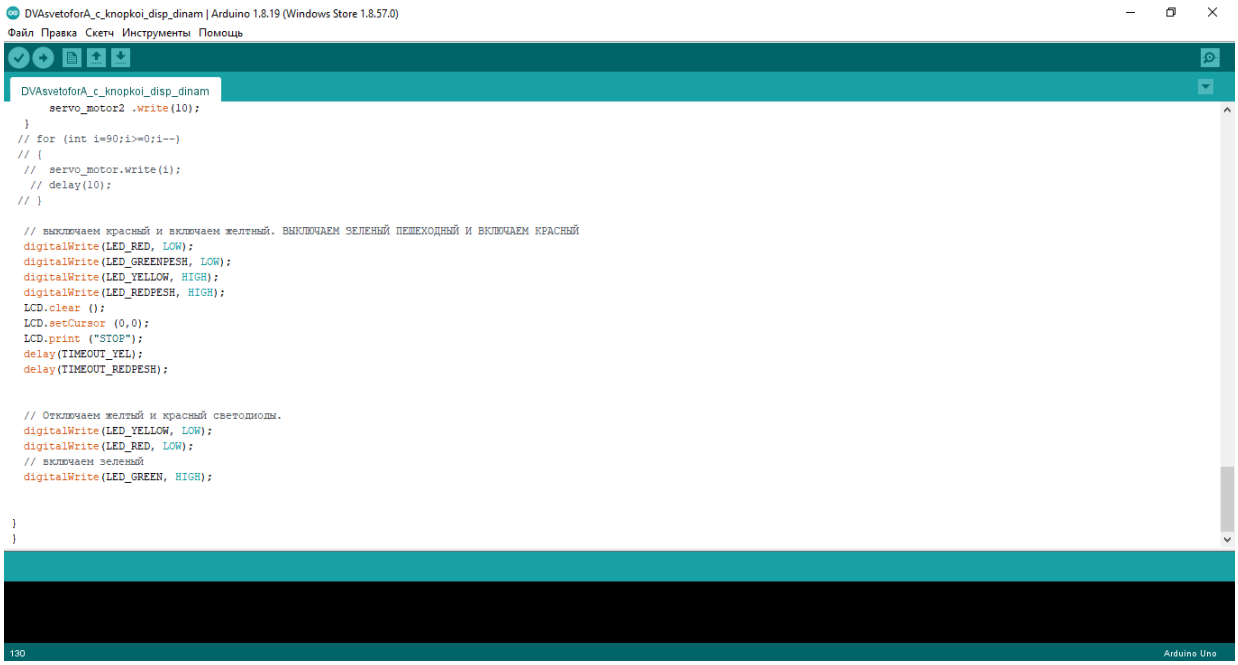


Рисунок 1-7. Программа пешеходного перехода

Приложение [2]

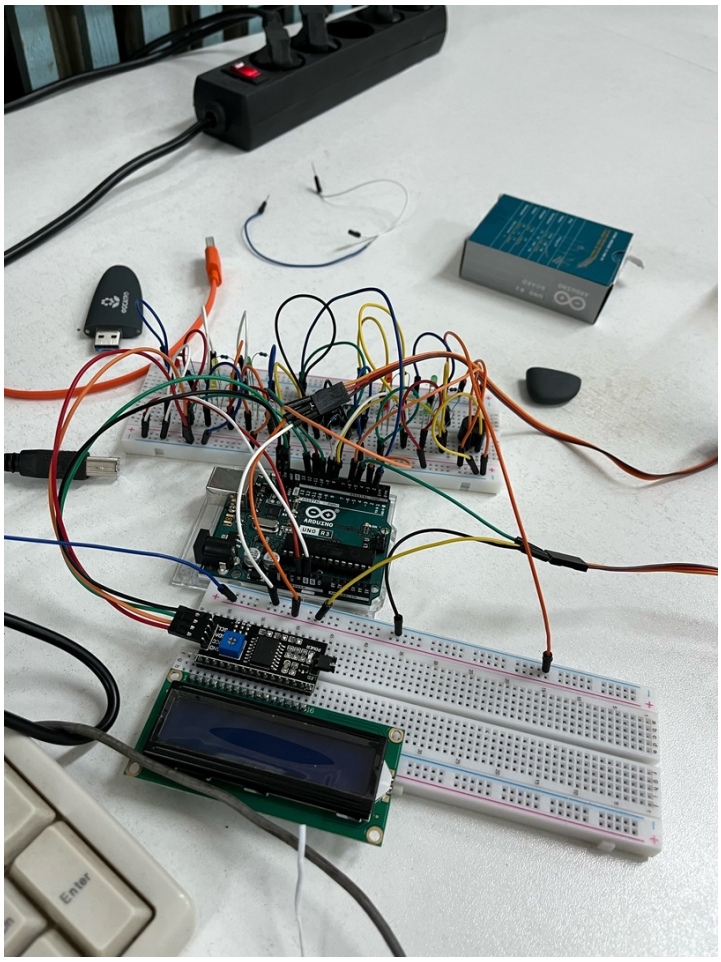


Рисунок 8. Монтажная схема подключения

Приложение [3]

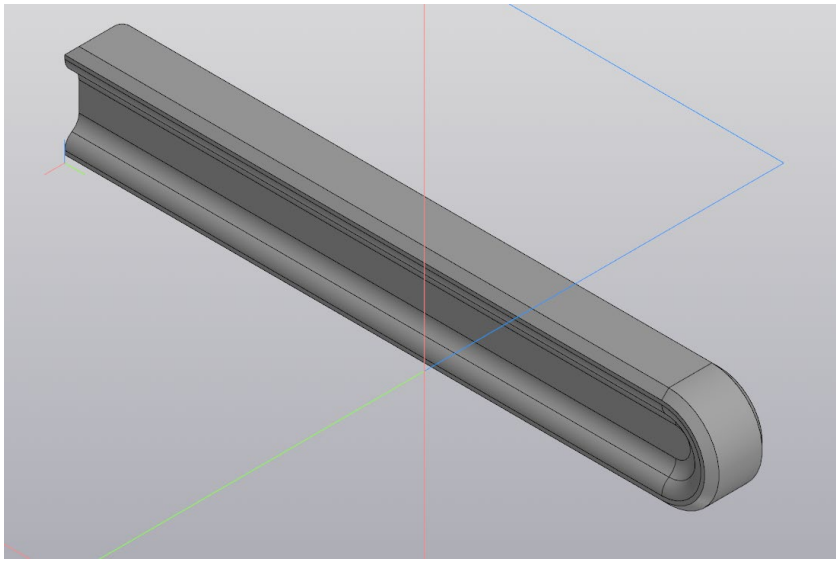


Рисунок 9. Шлагбаум

Приложение [4]

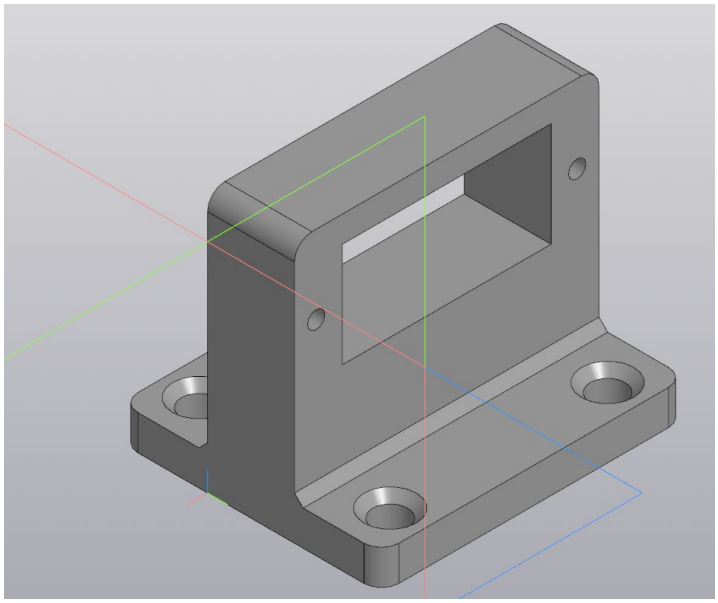


Рисунок 10. Крепление сервопривода

Приложение [5]

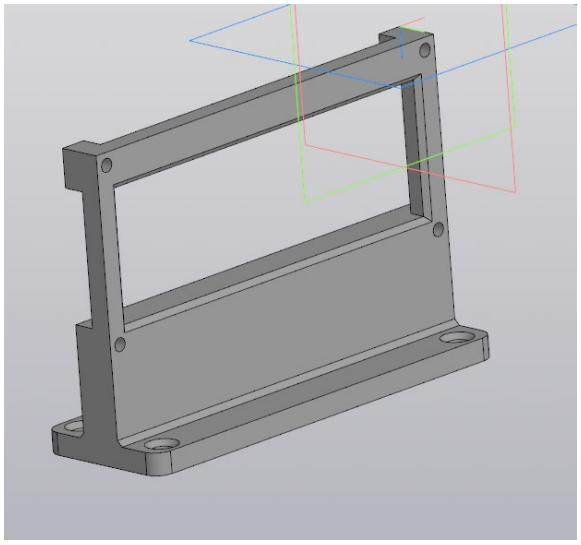


Рисунок 11. Крепление экрана

Приложение [6]

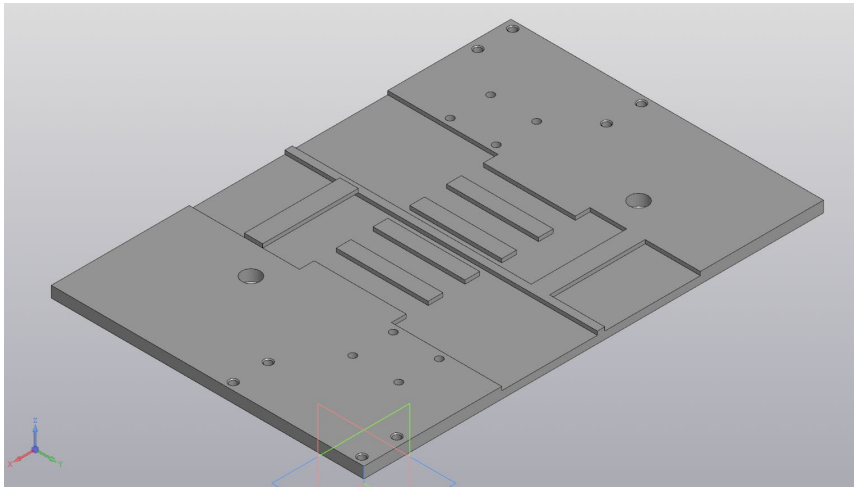


Рисунок 12. Площадка

Приложение [7]

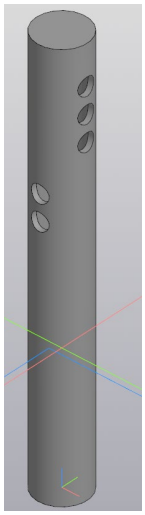


Рисунок 13. Светофор

Приложение [8]

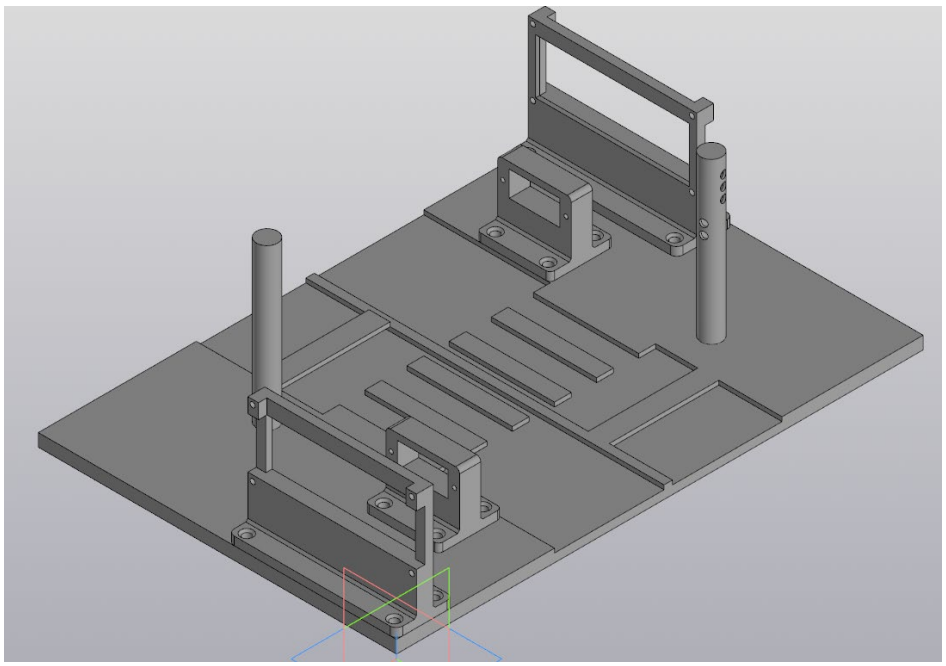
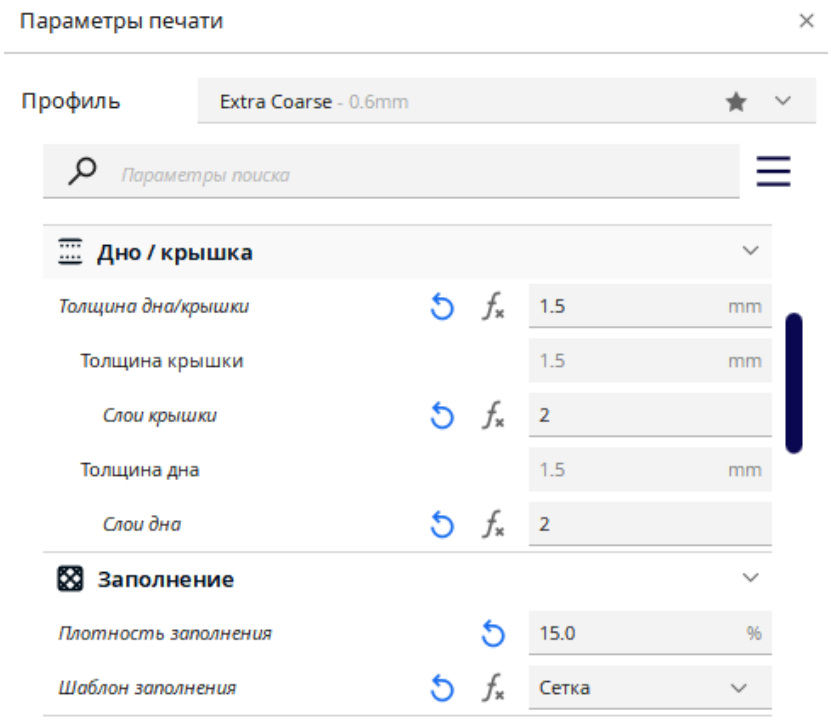


Рисунок 14.Сборка всего макета

Приложение [9]



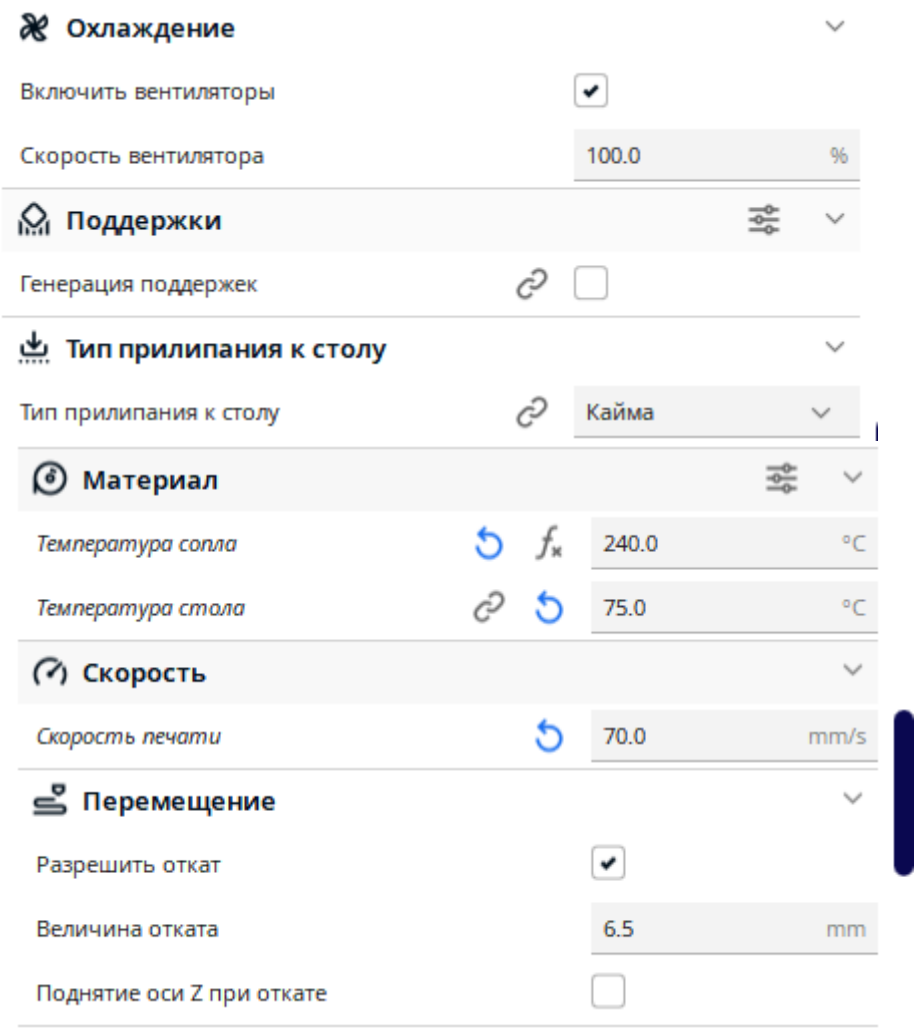
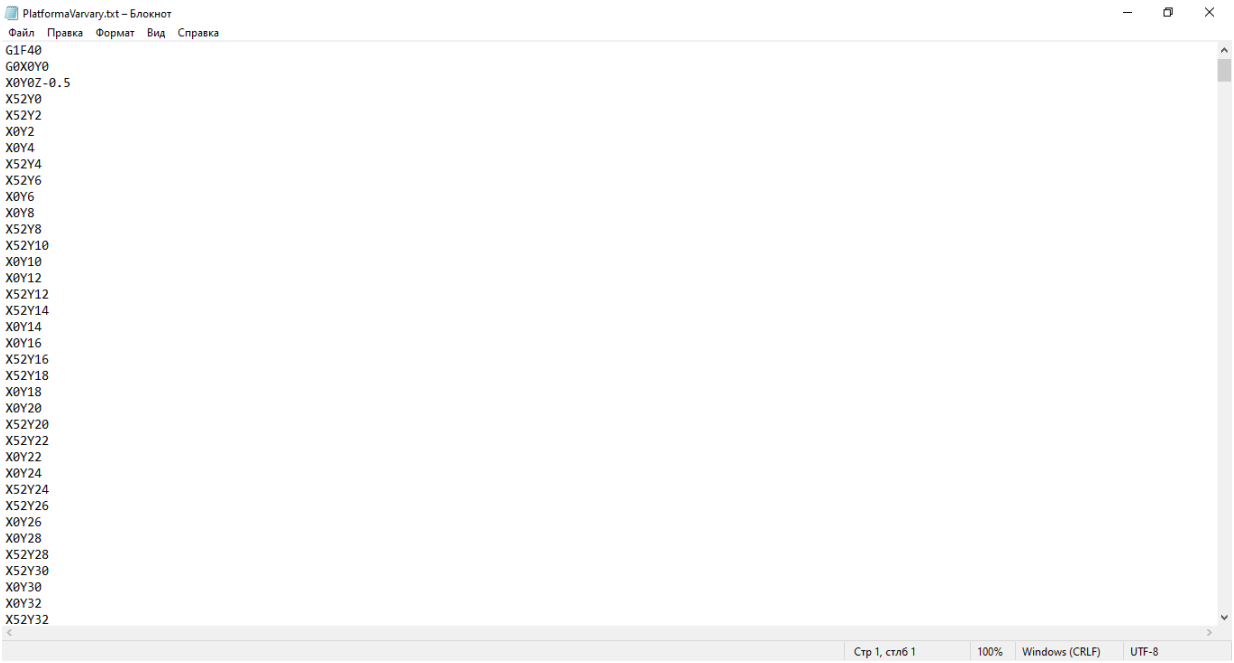


Рисунок 15-17. Настройки для печати

Приложение [10]



PlatformaVarvary.txt – Блокнот

Файл Правка Формат Вид Справка

```
X104Y130Z5
X104Y140Z5
X104Y140Z-1
X104Y170
X104Y170Z5
X106Y0Z5

X106Y0Z-1
X106Y60
X106Y60Z5
X106Y110Z5
X106Y110Z-1
X106Y130
X106Y130Z5
X106Y140Z5
X106Y140Z-1
X106Y170
X106Y170Z5
X108Y0Z5

X108Y0Z-1
X108Y60
X108Y60Z5
X108Y110Z5
X108Y110Z-1
X108Y130
X108Y130Z5
X108Y140Z5
X108Y140Z-1
X108Y170
X108Y170Z5
X0Y0Z15


M5
```

Стр 1, стлб 1 | 100% | Windows (CRLF) | UTF-8

Рисунок 18-19. Код для площадки

Приложение [11]

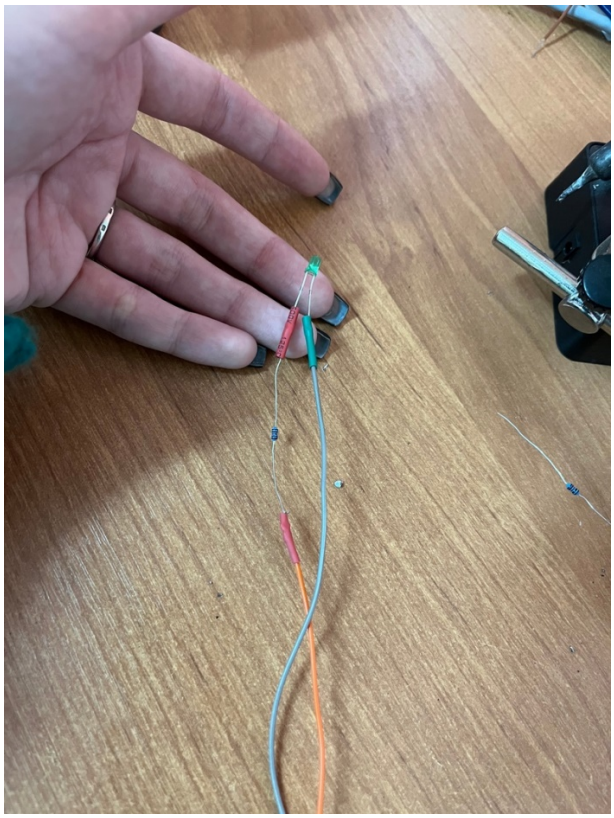


Рисунок 20. Спаянные провода

Приложение [12]

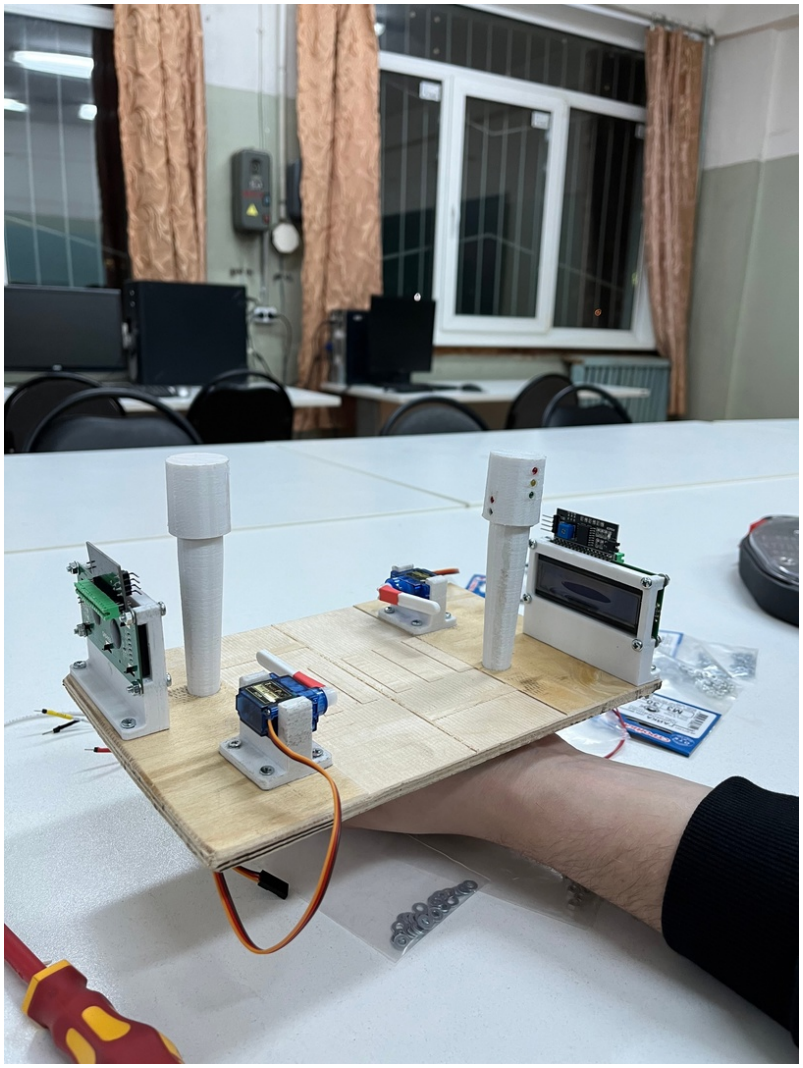


Рисунок 21. Готовый макет