Санкт-Петербургский государственный политехнический университет

Петра Великого

Институт прикладной математики и механики

**Кафедра «Теоретическая механика»**

**КУРСОВАЯ РАБОТА**

**Создание сортировщика-распаковщика на базе Arduino**

по дисциплине «Языки программирования»

Выполнил

студент гр. 23632/1 С.С.Барсуков

Руководитель

Ассистент А.Ю.Панченко

«\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2018 г.

Санкт-Петербург

2018

**Содержание**

1. Введение………………………………………………….. 3

1.1. Актуальность………………………………………... 4  
1.2. Поставленные цели…………………………………. 4  
1.3. Гипотеза………………………………….………….. 4

1.4. Проектная команда……………………….……….... 4

1. Основная часть…………………………………………... 5  
   2.1. Ход работы………………………………………….. 5   
   2.2. Использованное оборудование…………………….. 6  
   2.3. Arduino прошивка…………………………………… 6  
   2.4. Моделирование в Fusion 360……………………….. 7
2. Выводы по проекту……………………………………… 8
3. Список использованной литературы…………………… 9

**1.Введение**

Arduino

Arduino — торговая марка аппаратно-программных средств для построения простых систем автоматики и робототехники, ориентированная на непрофессиональных пользователей. Программная часть состоит из бесплатной программной оболочки (IDE) для написания программ, их компиляции и программирования аппаратуры. Аппаратная часть представляет собой набор смонтированных печатных плат, продающихся как официальным производителем, так и сторонними производителями. Полностью открытая архитектура системы позволяет свободно копировать или дополнять линейку продукции Arduino.

Микроконтроллеры для Arduino отличаются наличием предварительно прошитого в них загрузчика(bootloader). С помощью этого загрузчика пользователь загружает свою программу в микроконтроллер без использования традиционных отдельных аппаратных программаторов. Загрузчик соединяется с компьютером через интерфейс USB (если он есть на плате) или с помощью отдельного переходника UART-USB. Поддержка загрузчика встроена в Arduino IDE и выполняется в один щелчок мыши.

Язык программирования

Программирование ведется целиком через собственную программную оболочку (IDE), бесплатно доступную на сайте Arduino.

Язык программирования Arduino является стандартным C++ с некоторыми особенностями, облегчающими новичкам написание первой работающей программы.

- Программы, написанные программистом Arduino, называются наброски (или иногда скетчи и сохраняются в файлах с расширением ino. Эти файлы перед компиляцией обрабатываются препроцессором Ардуино. Также существует возможность создавать и подключать к проекту стандартные файлы C++.

- Обязательную в C++ функцию main() препроцессор Arduino создает сам, вставляя туда необходимые «черновые» действия.

- Программист должен написать две обязательные для Arduino функции setup() и loop(). Первая вызывается однократно при старте, вторая выполняется в бесконечном цикле.

- В текст своей программы (скетча) программист не обязан вставлять заголовочные файлы используемых стандартных библиотек. Эти заголовочные файлы добавит препроцессор Arduino в соответствии с конфигурацией проекта. Однако пользовательские библиотеки нужно указывать.

- Менеджер проекта Arduino IDE имеет нестандартный механизм добавления библиотек. Библиотеки в виде исходных текстов на стандартном C++ добавляются в специальную папку в рабочем каталоге IDE. При этом название библиотеки добавляется в список библиотек в меню IDE. Программист отмечает нужные библиотеки и они вносятся в список компиляции.

Arduino IDE не предлагает никаких настроек компилятора и минимизирует другие настройки, что упрощает начало работы для новичков и уменьшает риск возникновения проблем.

Простейшая Arduino-программа состоит из двух функций:

setup(): функция вызывается однократно при старте микроконтроллера.

loop(): функция вызывается после setup () в бесконечном цикле все время работы микроконтроллера.

Загрузка программы в микроконтроллер

Закачка программы в микроконтроллер Arduino происходит через предварительно запрограммированный специальный загрузчик (все микроконтроллеры от Ардуино продаются с этим загрузчиком). Загрузчик может работать через интерфейсы [RS-232](https://ru.wikipedia.org/wiki/RS-232), [USB](https://ru.wikipedia.org/wiki/USB) или [Ethernet](https://ru.wikipedia.org/wiki/Ethernet" \o "Ethernet) в зависимости от состава периферии конкретной процессорной платы. В некоторых вариантах, таких как Arduino Mini или неофициальной Boarduino, для программирования требуется отдельный переходник.

Пользователь может самостоятельно запрограммировать загрузчик в чистый микроконтроллер. Для этого в IDE интегрирована поддержка программатора на основе проекта [AVRDude](https://ru.wikipedia.org/wiki/AVRDude" \o "AVRDude). Поддерживается несколько типов популярных дешёвых программаторов.

**1.1.Актуальность**

К идее проекта подтолкнула проблема вскрытия упаковок со съедобным содержимым, связанная с отсутствием шероховатостей.

Данный проект был интересен с точки зрения реализации (желание опробовать оборудование в “ФабЛаб Политех” и научиться работе с микроконтроллерами).

**1.2.Поставленные цели**

Создать полуавтоматическую машину, которая будет открывать упаковки с пищей и делить поровну на заданное количество человек.

**1.3.Гипотеза**

Разрезание упаковки является эффективным методом её вскрытия.

**1.4.Проектная команда**

Асафов Руслан (студент группы 23632/1);

Барсуков Севастьян (студент группы 23632/1);

Вараев Владислав(студент группы 23632/1).

**2.Основная часть**

**2.1.Ход работы**

*| 23.03.18*

-Ознакомление с Arduino

-На данном этапе проблем не возникло

-Были разобраны материалы и видеоуроки по работе с Arduino, позволяющие приобрести необходимые для решения проблемы знания.

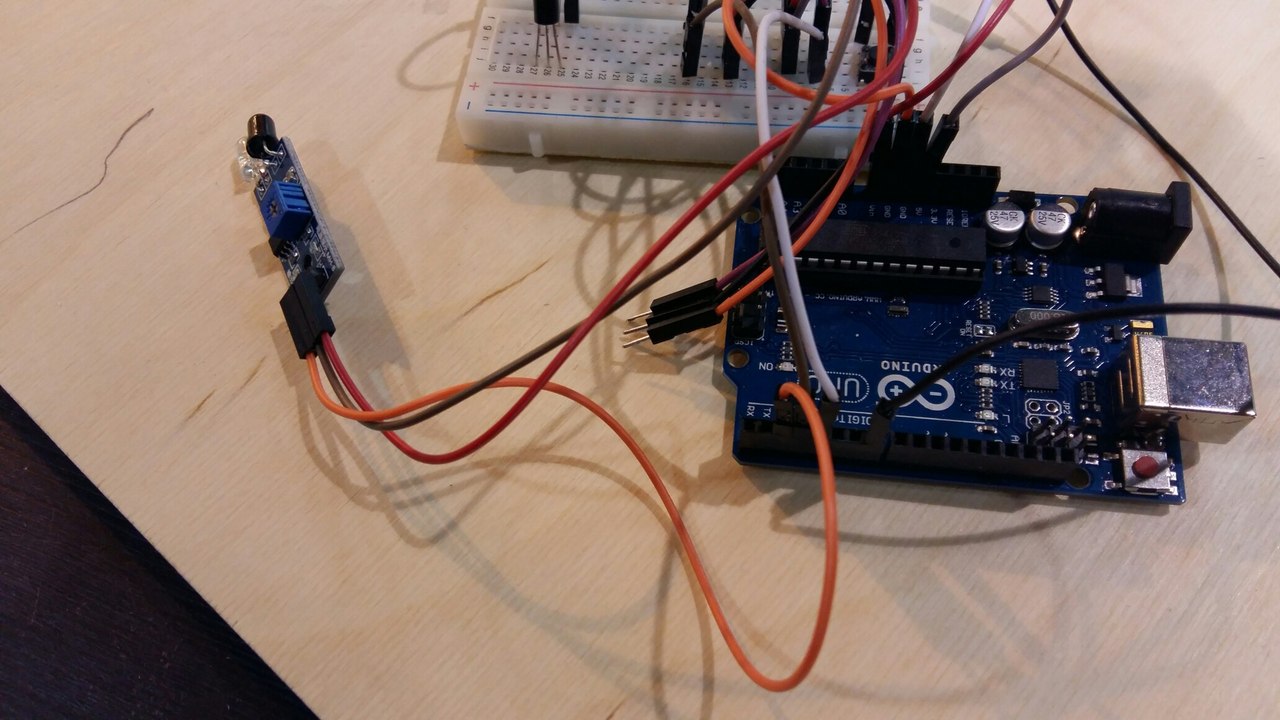
*| 01.04.18*

-Сборка основы конструкции – Arduino UNO c модулями

-Путём изменения кода программы была решена проблема с “залипанием” кнопки; При попытке подключении двигателя стала очевидной необходимость использования подключения двигателя через транзистор и резистор;

Нехватка портов подключения привела к покупке BreadBoard-а

-Собран "скелет" конструкции, основанный на Arduino UNO, и написана прошивка



*| 30.04.18*

-Работа с 3D принтером

-На данном этапе проблем не возникло

-С помощью Fusion 360 были смоделированы и распечатаны на принтере две заготовки(шестерни)

*| 07.05.18*

-Сборка механического каркаса

-Выяснилось, что мощности имеющегося двигателя не хватает для того, чтобы вращать шестеренки с натянутой лентой, а сама лента оказалась чрезмерно сопротивляющейся нагрузке на изгиб.

-Решено отказаться от использования двигателя постоянного тока (в связи с дороговизной более мощного)

*| 14.05.18 - 28.05.18*

-Сборка корпуса со всеми составляющими

-Вместо автоматического "открытия" упаковки было решено использовать механическое

-Корпус был собран

**2.2.Использованное оборудование**

-Arduino UNO

-Сервопривод SG90

-Датчик препятствий

-BreadBoard

-Кнопка, транзистор, резистор, комплект проводов

**2.3.Arduino прошивка**

Нам необходимо подключение библиотеки для работы с сервоприводом:

#include <Servo.h> // Библиотека для сервопривода

Объявим необходимые переменные. Переменная *servo1* объявляется особым типом переменных специфичным для работы с сервоприводом (*Servo*).

Servo servo1; // Объявление переменных

int timestep;

int timestep1;

boolean flag = 0;

boolean flag1 = 0;

boolean battle;

byte i;

byte j;

byte v;

byte c;

Функция *setup()* выполняется единожды при подключении Arduino к источнику питания. Настраиваем пины на вход и выход, сервопривод ставим в положение *0* градусов.

void setup() {

servo1.attach(4); // Задание пинов на вход и выход

pinMode(3,INPUT\_PULLUP);

pinMode(2,INPUT);

pinMode(9,OUTPUT);

Serial.begin(9600); // Открытие com-порта со скоростью передачи данных 9600 бит/с

servo1.write(0);

// digitalWrite(9,LOW);

}

Функция *loop()* выполняется многократно.

void loop() {

v = !digitalRead(2);

battle = !digitalRead(3);

Отрабатываем нажатие кнопки. Устранено “залипание” кнопки. Встроенная функция *millis()* возвращает время с начала работы программы.

if (battle==1 && flag==0 && (millis()-timestep)>300){ // Отрабатываем нажатие кнопки..

flag = 1;

i++;

Serial.println(i); // Вывод в порт (использовалось для отладки программы)

timestep = millis();

}

Отжатие кнопки.

if (battle==0 && flag==1){ // ..и отжатие

flag = 0;

}

Через 2.5 секунды после последнего нажатия кнопки запускается дальнейший скрипт.

if ((millis()-timestep)>2500 && i>1){

j = i;

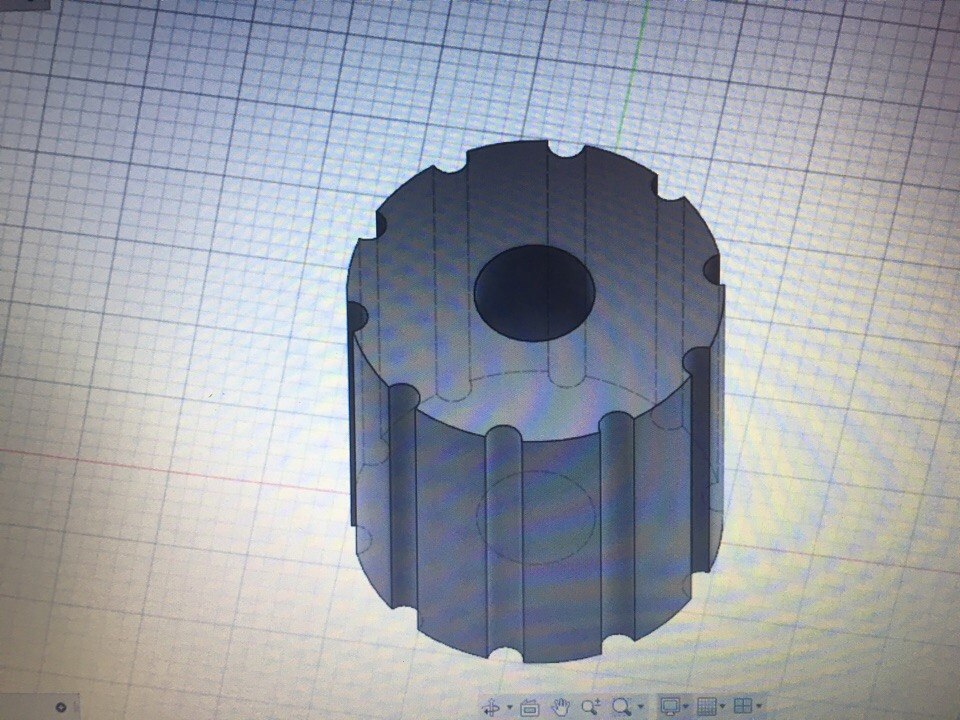
}

**2.4. Моделирование в Fusion 360**

Fusion 360TM - это первый инструмент 3D CAD, CAM и CAE в своем роде. Он объединяет весь процесс разработки продукта в единой облачной платформе, которая работает как на Mac, так и на ПК.

Комплекс представляет собой полноценный CAD/CAE/CAM инструмент. Применяется в сфере промышленного дизайна и производства, позволяет спроектировать практически всё, что бывает необходимо.

В ходе проекта велась работа в Fusion 360 с целью моделирования шестерней для вала двигателя постоянного тока, способных передавать вращающий момент на ленту с помощью зубчатого закрепления



**3. Выводы по проекту**

Наблюдается выполнение основных функций установкой. Для большего эффекта требуется увеличить количество тестирований с целью выявления мест конструкции, требующих усовершенствований. Например, более рациональным и зрелищным способом вскрытия упаковки может оказаться её сдавливание и дальнейшее расхождение по швам, вероятность которого возможно увеличить путём расчёта места приложения усилия. Проблем, связанных с написанным скетчем к Arduino не обнаружено, сама программа не занимает много места в памяти Arduino UNO и обрабатывается быстро. Таким образом, использование микроконтроллера Arduino отлично подходит для реализации проекта такого рода. В итоге часть работ, отведённая Барсукову С.С., по написанию прошивки для платы Arduino и по сборке установки была выполнена, что позволило привести установку к конечному виду и добиться её работоспособности.

1. **Список использованной литературы**
2. Видео уроки по Arduino // [Официальный сайт YouTube канала [AlexGyver](https://www.youtube.com/channel/UCgtAOyEQdAyjvm9ATCi_Aig)**]** / Copyright AlexGyver Technologies 2015-2018 – URL: <http://alexgyver.ru/arduino_lessons/> . – (дата обращения: 23.03.2018).
3. Лекции по Arduino // [Информационный портал BOTEON] / "BOTEON" 2012-2013 высокие технологии, робототехника, энергосбережение, интернет технологии– URL: <http://boteon.com/blogs/obuchayuschie-lekcii-po-arduino> . – (дата обращения: 23.03.2018).
4. Интернет-класс обучения работы с Arduino // [Информационный портал robotclass] /– URL: <http://robotclass.ru> . – (дата обращения: 23.03.2018).
5. Р. Лафоре Объектно-ориентированное программирование в С++ – Издательство “Питер” – 4 издание; Санкт-Петербург: “Питер”, 2004. – С.168 – 217.