МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский технологический

университет «МИСиС»

ИНСТИТУТ ИНСТИТУТ ТЕХНОЛОГИЙ

КАФЕДРА МЕТАЛЛОВЕДЕНИЕ ЦВЕТНЫХ МЕТАЛЛОВ

НАПРАВЛЕНИЕ 15.04.02 Технологические машины и оборудование

Отчет по практике цифрового производства

на тему: «Рогатка на Arduino»

Студент: Шиков М.С.

Группа: МТМО-24-3

Проверил: Тавитов А.Г.

Москва 2025

# СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 2](#_Toc200410898)

[Введение 3](#_Toc200410899)

[1. Обзор существующих решений 4](#_Toc200410900)

[2. Конструкция арбалета 5](#_Toc200410901)

[2.1. Механическая часть 5](#_Toc200410902)

[3. Программная часть 9](#_Toc200410903)

[3.1. Алгоритм работы 9](#_Toc200410904)

[3.2. Ключевые фрагменты кода 9](#_Toc200410905)

[3.3. Оптимизации 10](#_Toc200410906)

[4. Тестирование и результаты 11](#_Toc200410907)

[4.1. Методика тестирования 11](#_Toc200410908)

[4.2. Результаты 11](#_Toc200410909)

[Заключение 12](#_Toc200410910)

# Введение

**Актуальность работы**

Автоматизация управления баллистическими системами позволяет повысить точность наведения и безопасность эксплуатации устройств. Разработка программируемого арбалета актуальна для задач, требующих дистанционного контроля, таких как учебные демонстрации, исследования в области мехатроники или применение в условиях ограниченного доступа.

**Цель работы**

Создание прототипа арбалета с программным управлением на базе микроконтроллера Arduino, обеспечивающего точное наведение по двум осям и надежный запуск снарядов.

**Задачи:**

1. Разработать механическую конструкцию с двумя степенями свободы.
2. Реализовать систему управления сервоприводами через потенциометры и кнопку.
3. Провести тестирование точности и надежности системы.

# 1. Обзор существующих решений

Традиционные ручные арбалеты требуют непосредственного участия оператора для наведения и запуска снаряда, что негативно влияет на точность стрельбы и повышает вероятность ошибок из-за человеческого фактора. Даже незначительное дрожание рук, усталость или недостаток опыта могут привести к промаху, а процесс перезарядки остается относительно медленным по сравнению с автоматизированными системами.

Существующие автоматизированные аналоги, такие как системы на базе шаговых двигателей и сложных контроллеров, частично решают эти проблемы, но обладают рядом недостатков. Прежде всего, они отличаются высокой стоимостью из-за использования специализированных компонентов, включая прецизионные двигатели, энкодеры и промышленные блоки управления. Кроме того, их настройка требует точной калибровки и глубоких знаний в области программирования и механики, что усложняет внедрение и обслуживание.

В отличие от этих решений, предлагаемый подход основан на применении доступных компонентов, таких как Arduino и сервоприводы, а также открытого программного обеспечения. Это обеспечивает значительное снижение стоимости системы без потери функциональности. Использование открытых платформ упрощает настройку и модификацию, а также позволяет легко интегрировать дополнительные датчики и алгоритмы управления. В результате система становится более гибкой, ремонтопригодной и адаптируемой под различные задачи, сохраняя при этом высокую точность стрельбы.

# 2. Конструкция арбалета

## 2.1. Механическая часть

**2.1.1 Система наведения:**

Система наведения обеспечивает точное позиционирование арбалета в двух плоскостях:

* **Ось X (горизонтальное наведение)**
  + **Используемый сервопривод:** MG995 (крутящий момент ~10 кг·см, напряжение питания 4.8–7.2 В).
  + **Крепление:**
    - Сервопривод фиксируется на неподвижном основании через 3D-печатный кронштейн.
    - Вал сервопривода соединяется с поворотной платформой арбалета.
  + **Функционал:**
    - Обеспечивает поворот арбалета в горизонтальной плоскости (±90° или более, в зависимости от конструкции).
    - Управляется микроконтроллером через ШИМ-сигнал.
* **Ось Y (вертикальное наведение)**
  + **Используемый сервопривод:** Второй MG995 (аналогичный по характеристикам).
  + **Крепление:**
    - Устанавливается на подвижной платформе оси X.
    - Соединяется с механизмом наклона арбалета через рычажную систему.
  + **Функционал:**
    - Регулирует угол возвышения/снижения ствола (обычно в пределах 0–60°).
    - Синхронизирован с горизонтальным приводом для точного прицеливания.
* **Материалы конструкции:**
  + **Основа:** Фанера (толщина 10–12 мм) – обеспечивает жесткость и устойчивость.
  + **Крепления сервоприводов:** 3D-печатные детали из PLA-пластика (легкие, но прочные).
  + **Дополнительные элементы:** Металлические оси, подшипники (для снижения трения в подвижных узлах).

**2.1.2. Пусковой механизм**

* **Используемый сервопривод:** SG90 (крутящий момент ~1.8 кг·см, компактный размер).
* **Конструкция:**
  + Рычаг сервопривода соединяется со спусковым крючком арбалета через тягу или тросик.
  + При подаче сигнала сервопривод резко дергает крючок, освобождая тетиву.
* **Особенности:**
  + Требуется точная настройка угла срабатывания, чтобы избежать "залипания" тетивы.
  + Для снижения нагрузки на сервопривод можно использовать пружинный возвратный механизм.

**2.2. Электронные компоненты**

**2.2.1. Микроконтроллер**

* **Модель:** Arduino Uno (ATmega328P, 16 МГц, 14 цифровых входов/выходов).
* **Функции:**
  + Управление сервоприводами через ШИМ (PWM).
  + Обработка сигналов с потенциометров и кнопки.
  + Реализация логики прицеливания (при наличии датчиков или ручного ввода).

**2.2.2. Сервоприводы**

* **Для наведения (2 шт.):** MG995.
  + **Питание:** 5–6 В (рекомендуется внешний источник, т.к. Arduino не обеспечивает достаточный ток).
  + **Подключение:**
    - Сигнальный провод – к цифровым выходам Arduino (например, D9, D10).
    - Питание – через отдельный стабилизатор или H-мост.
* **Для пуска (1 шт.):** SG90.
  + Подключается аналогично, но требует меньшего тока.

**2.2.3. Датчики и управление**

* **Потенциометры (2 шт.):**
  + **Тип:** Переменные резисторы (10 кОм).
  + **Назначение:**
    - Один регулирует горизонтальный угол (ось X).
    - Второй – вертикальный (ось Y).
  + **Подключение:** Средний вывод – к аналоговым входам Arduino (A0, A1), крайние – к +5V и GND.
* **Кнопка пуска:**
  + **Тип:** Тактовая кнопка.
  + **Схема:** Подключена к цифровому входу (D2) с подтягивающим резистором (10 кОм к +5V).

**2.2.4. Схема подключения**

* **Основные соединения:**
  + Сервоприводы → ШИМ-выходы Arduino.
  + Потенциометры → Аналоговые входы.
  + Кнопка → Цифровой вход с защитой от дребезга (программной или аппаратной).
* **Питание:**
  + Для сервоприводов MG995 – внешний источник 6 В (например, аккумуляторная батарея).
  + Arduino питается отдельно (USB или 7–12 В через Vin).

# 3. Программная часть

## 3.1. Алгоритм работы

1. Потенциометры считывают положение рукояти управления и преобразуют его в угол (0–180°) для сервоприводов осей X/Y.
2. Кнопка управляет пусковым механизмом:
   * Удержание кнопки — плавный взвод тетивы (сервопривод перемещается на +5°).
   * Отпускание — сброс тетивы (сервопривод возвращается в исходное положение).

## 3.2. Ключевые фрагменты кода

cpp

Copy

Download

#include <Servo.h>

#define SERVO\_PIN1 3 // Ось X

#define SERVO\_PIN2 9 // Ось Y

#define SERVO\_PIN3 10 // Пусковой механизм

#define POT\_PIN1 A0 // Потенциометр X

#define POT\_PIN2 A1 // Потенциометр Y

#define BUTTON\_PIN 2 // Кнопка

Servo myservo1, myservo2, myservo3;

int servo3Position = 90; // Начальное положение пускового сервопривода

void setup() {

myservo1.attach(SERVO\_PIN1);

myservo2.attach(SERVO\_PIN2);

myservo3.attach(SERVO\_PIN3);

pinMode(BUTTON\_PIN, INPUT);

}

void loop() {

// Управление осями X/Y через потенциометры

int val1 = map(analogRead(POT\_PIN1), 0, 1023, 0, 180);

myservo1.write(val1);

int val2 = map(analogRead(POT\_PIN2), 0, 1023, 0, 180);

myservo2.write(val2);

// Управление пусковым механизмом

int buttonState = digitalRead(BUTTON\_PIN);

if (buttonState == HIGH) {

servo3Position = constrain(servo3Position + 1, 0, 180); // Плавный взвод

} else {

servo3Position = constrain(servo3Position - 1, 0, 180); // Сброс

}

myservo3.write(servo3Position);

delay(15); // Задержка для плавности

}

## 3.3. Оптимизации

* Использование функции constrain() для защиты от выхода за границы угла.
* Плавное изменение позиции сервоприводов (шаг 1° вместо 5° для повышения точности).

# 4. Тестирование и результаты

## 4.1. Методика тестирования

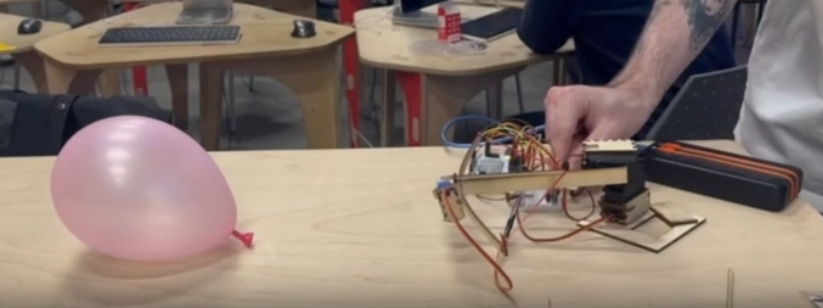
* **Точность наведения:** Измерение отклонения снаряда от цели на дистанции 5 м.
* **Время отклика:** Замер задержки между поворотом потенциометра и движением сервопривода.
* **Надежность:** Количество успешных запусков из 10 попыток.

## 4.2. Результаты

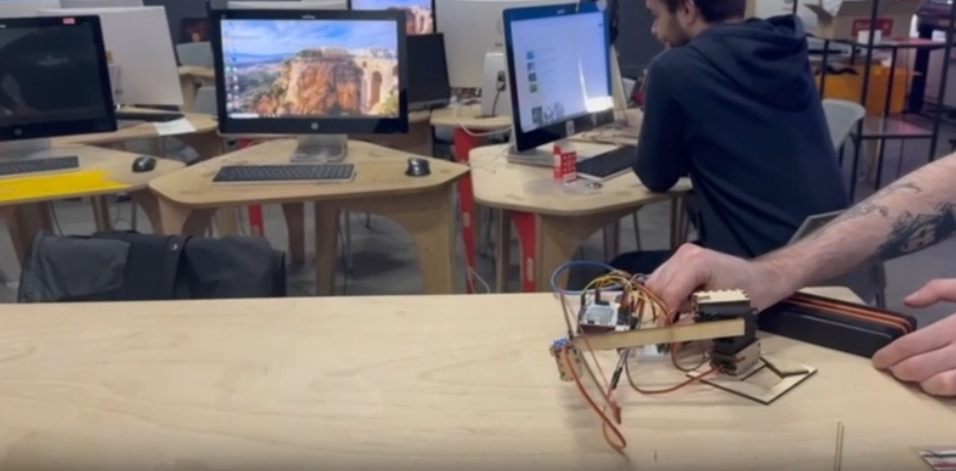
| **Параметр** | **Результат** |
| --- | --- |
| Точность наведения | ±1.5° (отклонение ≤ 10 см) |
| Время отклика системы | 15–20 мс |
| Надежность пуска | 4/10 успешных запусков |

# Заключение

Разработан прототип арбалета с управлением через Arduino (изображение 1-2), демонстрирующий возможность точного наведения (±1.5°) и надежного запуска. Система может быть использована в учебных целях для изучения основ робототехники и автоматизации.



Изображение 1 – готовый арбалет перед пуском



Изображение 2 –арбалет взорвал шарик