МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ «МИСИС»**

*ИНСТИТУТ* ТЕХНОЛОГИЙ

*КАФЕДРА* МЕТАЛЛОВЕДЕНИЯ ЦВЕТНЫХ МЕТАЛЛОВ

*НАПРАВЛЕНИЕ* 15.04.02 «Технологические машины и оборудование»

**КУРСОВАЯ РАБОТА**

**по предмету: Современные проблемы металлургии, машиностроения и материаловедения**

**на тему: АРБАЛЕТ \_**

*Выполнил:*

*Студент* Шиков М.С.

*Группа* МТМО 24-3

*Проверил*

Тавитов А.Г

*Оценка*

Москва, май 2025

# СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 2](#_Toc198824696)

[ВВЕДЕНИЕ 3](#_Toc198824697)

[1. ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ 4](#_Toc198824698)

[2. КОНСТРУКЦИЯ АРБАЛЕТА 5](#_Toc198824699)

[2.1. МЕХАНИЧЕСКАЯ ЧАСТЬ 5](#_Toc198824700)

[2.2. ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ 5](#_Toc198824701)

[3. ПРОГРАММНАЯ ЧАСТЬ 6](#_Toc198824702)

[3.1. АЛГОРИТМ РАБОТЫ 6](#_Toc198824703)

[3.2. КЛЮЧЕВЫЕ ФРАГМЕНТЫ КОДА 6](#_Toc198824704)

[3.3. ОПТИМИЗАЦИИ 7](#_Toc198824705)

[4. ТЕСТИРОВАНИЕ И РЕЗУЛЬТАТЫ 8](#_Toc198824706)

[4.1. МЕТОДИКА ТЕСТИРОВАНИЯ 8](#_Toc198824707)

[4.2. РЕЗУЛЬТАТЫ 8](#_Toc198824708)

[ЗАКЛЮЧЕНИЕ 9](#_Toc198824709)

# 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

Ручные арбалеты требуют непосредственного участия оператора для наведения и запуска, что снижает точность и повышает риск ошибок. Существующие автоматизированные аналоги (например, системы на базе шаговых двигателей) часто имеют высокую стоимость и сложность настройки. Преимущество данной работы — использование доступных компонентов (Arduino, сервоприводы) и открытого ПО для управления.

# 2. Конструкция арбалета

## 2.1. Механическая часть

* **Система наведения:**
  + Ось X: Горизонтальный сервопривод (например, MG995) для управления направлением.
  + Ось Y: Вертикальный сервопривод, закрепленный на оси X.
  + Материалы: Основа из фанеры, крепления сервоприводов напечатаны на 3D-принтере.
* **Пусковой механизм:**
  + Отдельный сервопривод, соединенный с спусковым крючком для натяжения и освобождения тетивы.

## 2.2. Электронные компоненты

* Микроконтроллер: Arduino Uno.
* Сервоприводы: 2 × MG995 (2 для осей X/Y), 1 × SG90 (для пуска).
* Датчики: 2 потенциометра (управление углом наклона), кнопка (пуск).
* Схема подключения (см. Приложение 1).

# 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, демонстрирующий возможность точного наведения (±1.5°) и надежного запуска. Система может быть использована в учебных целях для изучения основ робототехники и автоматизации.