

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа Инженерная школа информационных технологий и робототехники
Направление подготовки 15.03.06 Мехатроника и робототехника
Отделение школы (НОЦ) Отделение автоматизации и робототехники

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Разработка и реализация алгоритмов автоматического управления для задач слежения за распознанными объектами

УДК 004.93.021.621.396.965.8

Студент

Группа	ФИО	Подпись	Дата
8E41	Войцеховский Алексей Алексеевич		

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ВКР	Зарницын Александр Юрьевич			
Руководитель ООП	Мамонова Татьяна Егоровна	к.т.н.		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП ТПУ	Петухов Олег Николаевич	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОКД ИШНКБ ТПУ	Авдеева Ирина Ивановна			

ДОПУСТИТЬ К ЗАЩИТЕ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ОАР	Леонов Сергей Владимирович	к.т.н.		

РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код рез-та	Результат обучения (выпускник должен быть готов)	Требования ФГОС, критериев и/или заинтересованных сторон
<i>Профессиональные</i>		
P1	применять глубокие естественно-научные, математические знания в области анализа, синтеза и проектирования для решения научных и инженерных задач производства и эксплуатации мехатронных и робототехнических устройств и систем, в том числе их систем управления.	Требования ФГОС (ПК-1, ПК-3, ОПК-1, ОПК-4, ОК-1, ОК-9), Критерий 5 АИОР (п. 1.1), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P2	воспринимать, обрабатывать, анализировать и обобщать научно-техническую информацию, передовой отечественный и зарубежный опыт в области теории, проектирования, производства и эксплуатации мехатронных и робототехнических устройств и систем, принимать участие в командах по разработке и эксплуатации таких устройств и систем.	Требования ФГОС (ПК-3, ПК-4, ПК-7, ОПК-1, ОПК-3, ОК-1, ОК-4, ОК-5, ОК-6, ОК-9), Критерий 5 АИОР (пп. 1.1, 1.2), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P3	применять и интегрировать полученные знания для решения инженерных задач при разработке, производстве и эксплуатации современных мехатронных и робототехнических устройств и систем (в том числе интеллектуальных) с использованием технологий мирового уровня, современных инструментальных и программных средств.	Требования ФГОС (ПК-2, ПК-3, ПК-4, ПК-5, ПК-15, ПК-18, ОПК-3, ОПК-6, ОК-1, ОК-5, ОК-6, ОК-7), Критерий 5 АИОР (пп. 1.2), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P4	определять, систематизировать и получать необходимую информацию в области проектирования, производства, исследований и эксплуатации мехатронных и робототехнических модулей, устройств и систем	Требования ФГОС (ПК-7, ПК-10, ПК-11, ПК-12, ПК-18, ОПК-4, ОПК-6, ОК-1, ОК-4, ОК-6, ОК-8), Критерий 5 АИОР (п. 1.3), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P5	планировать и проводить аналитические, имитационные и экспериментальные исследования для целей проектирования, производства и эксплуатации мехатронных и робототехнических средств и систем (в том числе интеллектуальных) с использованием передового отечественного и зарубежного опыта, уметь критически оценивать полученные теоретические и экспериментальные данные и делать выводы.	Требования ФГОС (ПК-1, ПК-2, ПК-3, ПК-4, ПК-5, ПК-6, ПК-13, ПК-17, ПК-18, ОПК-2, ОПК-3, ОК-1, ОК-3, ОК-4, ОК-6, ОК-7, ОК-8, ОК-9), Критерий 5 АИОР (п. 1.4), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P6	понимать используемые современные методы, алгоритмы, модели и технические решения в мехатронике и робототехнике и знать области их применения, в том числе в автоматизированных производствах.	Требования ФГОС (ПК-1, ПК-2, ПК-3, ПК-7, ОПК-1, ОПК-3, ОПК-4, ОК-5, ОК-9, ОК-10), Критерий 5 АИОР (п. 2.1), согласованный с требованиями международных стандартов EUR-ACE и FEANI

<i>Универсальные</i>		
P7	эффективно работать в профессиональной деятельности индивидуально и в качестве члена команды	Требования ФГОС (ПК-1, ПК-2 ПК-7, ПК-8, ПК-16, ПК-17, ОК-1, ОК-2, ОК-4, ОК-6, ОК-9), Критерий 5 АИОР (п. 2.1), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P8	владеть иностранным языком на уровне, позволяющем работать в интернациональной среде с пониманием культурных, языковых и социально-экономических различий	Требования ФГОС (ПК-4, ПК-8, ПК-9, ПК-16, ОПК-4, ОК-5), Критерий 5 АИОР (п. 2.2), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P9	проявлять широкую эрудицию, в том числе знание и понимание современных общественных и политических проблем, демонстрировать понимание вопросов безопасности и охраны здоровья сотрудников, юридических аспектов, ответственности за инженерную деятельность, влияния инженерных решений на социальный контекст и окружающую среду	Требования ФГОС (ПК-5, ПК-8, ПК-15, ПК-16, ПК-18, ОПК-1, ОПК-4, ОПК-5, ОК-3, ОК-4, ОК-5, ОК-6, ОК-8, ОК-9), Критерий 5 АИОР (пп. 1.6, 2.3.), согласованный с требованиями международных стандартов EUR-ACE и FEAN
P10	следовать кодексу профессиональной этики и ответственности и международным нормам инженерной деятельности	Требования ФГОС (ПК-8, ПК-11, ПК-16, ОПК-3, ОПК-6, ОК-4), Критерий 5 АИОР (пп. 2.4, 2.5), согласованный с требованиями международных стандартов EUR-ACE и FEANI
P11	понимать необходимость и уметь самостоятельно учиться и повышать квалификацию в течение всего периода профессиональной деятельности.	Требования ФГОС (ПК-4, ПК-8, ОПК-3, ОПК-4, ОК-5, ОК-6, ОК-7, ОК-8), Критерий 5 АИОР (2.6), согласованный с требованиями международных стандартов EUR-ACE и FEANI.

<p>Перечень подлежащих исследованию, проектированию и разработке вопросов</p> <p><i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<ol style="list-style-type: none"> 1. Провести обзор литературы 2. Построить модель объекта управления 3. Разработать алгоритм управления. 4. Написать программу на выбранном языке программирования.
<p>Перечень графического материала</p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<ol style="list-style-type: none"> 1. Блок-схемы алгоритма программы. 2. Диаграммы классов. 3. Дерево вызова процедур.
<p>Консультанты по разделам выпускной квалификационной работы</p> <p><i>(с указанием разделов)</i></p>	
<p>Раздел</p>	<p>Консультант</p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Петухов Олег Николаевич, доцент ОСГН, к.э.н.</p>
<p>Социальная ответственность</p>	<p>Авдеева Ирина Ивановна, ассистент ОКД</p>
<p>Названия разделов, которые должны быть написаны на русском и иностранном языках:</p>	
<p>Нет</p>	

<p>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</p>	<p>01.03.2018</p>
--	-------------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
ассистент ОАР	Зарницын Александр Юрьевич			01.03.2018

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Е41	Войцеховский Алексей Алексеевич		01.03.2018

Министерство образования и науки Российской Федерации
 федеральное государственное автономное образовательное учреждение
 высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа Инженерная школа информационных технологий и робототехники
 Направление подготовки (специальность) 15.03.06 Мехатроника и робототехника
 Уровень образования бакалавр
 Отделение школы (НОЦ) Отделение автоматизации и робототехники
 Период выполнения _____ (осенний / весенний семестр 2017/2018 учебного года)

Форма представления работы:

Бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
01.06.2018	Основная часть	60
24.05.2018	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	20
16.05.2018	Социальная ответственность	20

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
ассистент ОАР	Зарницын Александр Юрьевич			

СОГЛАСОВАНО:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОАР	Мамонова Татьяна Егоровна	к.т.н.		

Реферат

Выпускная квалификационная работа состоит из 91 страницы, включает в себя 25 рисунков, 25 таблиц и 5 приложений. При работе были использованы 14 источников литературы.

Ключевые слова: идентификация систем, алгоритм управления, тыловые роботы, мобильная робототехника, ПИД-регулятор, военная робототехника.

Цель работы – разработать алгоритм автоматического управления для задач слежения за распознанными объектами.

Объектом исследования является практическая часть разработки программного обеспечения для микроконтроллера, отвечающего за управление приводами мобильного робота.

Область внедрения данной работы – это программное обеспечение для тыловых роботов в воинских частях.

Определения

В пояснительной записке использовались следующие термины с соответствующими определениями:

Микроконтроллер – микросхема, содержащая на одном кристалле процессор, периферийные устройства, оперативное запоминающее устройство и постоянное запоминающее устройство.

Пропорционально-интегрально-дифференцирующий регулятор – устройство в управляющем контуре с обратной связью. ПИД-регулятор формирует управляющий сигнал, являющийся суммой трёх слагаемых, первое из которых пропорционально разности входного сигнала и сигнала обратной связи (сигнал рассогласования), второе — интеграл сигнала рассогласования, третье — производная сигнала рассогласования.

Широтно-импульсная модуляция – процесс управления мощностью, подводимой к нагрузке, путём изменения скважности импульсов, при постоянной частоте.

Обозначения и сокращения

ПИД – пропорционально-интегрально-дифференциальный

МК – микроконтроллер

ПК – персональный компьютер

ПО – программное обеспечение

СТЗ – система технического зрения

ШИМ – широтно-импульсная модуляция

Содержание

Определения	8
Обозначения и сокращения.....	8
Введение.....	13
1 Техническое задание.....	14
1.1 Основные задачи и цели.....	14
1.2 Назначение разрабатываемой программы.....	14
1.3 Требования к функциям программы.....	14
1.4 Требования к техническому обеспечению	15
2 Объект управления	16
2.1 Мотор-редукторы.....	16
2.2 Система технического зрения.....	17
2.3 Оптические энкодеры.....	18
2.4 Ультразвуковые дальномеры.....	19
2.5 Микроконтроллер	20
3 Обзор методов управления	21
3.1 ПИД-регулятор.....	21
3.2 Нейросетевой контроллер	22
3.3 Выбор метода управления.....	23
4 Идентификация электропривода.....	24
4.1 Получение экспериментальных данных.....	25
4.2 Идентификация параметров.....	27
4.3 Результаты идентификации	27
5 Синтез регуляторов.....	31
5.1 Синтез ПИД-регулятора скорости	31

5.2 Синтез ПИД-регулятора расстояния.....	32
5.3 Синтез ПИД-регулятора угла	34
6 Прием данных от системы технического зрения.....	36
6.1 Интерфейс передачи данных UART	36
6.2 Преобразование данных об отклонении объекта в угол отклонения	38
7 Реализация алгоритма.....	40
8 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	43
8.1 Потенциальные потребители результатов исследования	43
8.2 Анализ конкурентных технических решений.....	43
8.3 Технология QuaD	44
8.4 SWOT – анализ.....	45
8.5 Морфологический анализ	46
8.6 Планирование научно-исследовательских работ	49
8.6.1 Структура работ в рамках научного исследования.....	49
8.6.2 Определение трудоемкости выполнения работ.....	51
8.6.3 Разработка графика проведения научного исследования.....	55
8.7 Бюджет научно-технического исследования	56
8.7.1 Расчет материальных затрат	56
8.7.2 Основная заработная плата исполнителям темы.....	57
8.7.3 Дополнительная заработная плата	58
8.7.4 Отчисления во внебюджетные фонды.....	59
8.7.5 Накладные расходы	60
8.7.6 Контрагентные расходы	60

8.7.7 Формирование бюджета затрат научно-исследовательского проекта.....	60
8.8 Определение ресурсной, финансовой и экономической эффективности ресурсов	61
9. Социальная ответственность	64
9.1 Производственная безопасность	64
9.1.1 Микроклимат.....	64
9.1.2 Шум.....	66
9.1.3 Освещение	67
9.1.4 Психофизиологические факторы	68
9.1.5 Электробезопасность.....	69
10 Экологическая безопасность	70
10.1 Анализ воздействий объекта на литосферу и атмосферу.....	70
11 Безопасность в чрезвычайных ситуациях	70
12 Правовые и организационные вопросы обеспечения безопасности	72
13 Вывод по разделу «Социальная ответственность»	72
Заключение	74
Список использованных источников.....	75
Приложение А. Дерево вызова процедур.....	77
Приложение Б. Блок-схема главной программы	79
Приложение В. Исходный код программы для получения экспериментальных данных (МК).....	82
Приложение Г. Исходный код программы для получения экспериментальных данных (ПК).....	83

Приложение Д. Исходный код основной программы.....	84
--	----

Введение

Снаряжение современного солдата имеет массу в среднем равную 32 килограмма. Это накладывает ограничения на мобильность и скорость передвижения боевых подразделений. Одним из вариантов решения данной проблемы являются тыловые роботы. Данные робототехнические системы сопровождают подразделения и перевозят вооружение и припасы.

В то же время актуальным является создание роботов, которые могли бы автоматически следовать за солдатом. Это связано с тем, что во многих ситуациях, в том числе в ходе боевых действий, у солдат, как правило, не имеется возможности вести ручное управление роботом. В то же время удаленное управление обычно неосуществимо в связи с глушением всех видов радиосигналов в зоне боевых действий.

Многие современные разработки робототехнических устройств включают в себя применение алгоритмов распознавания образов и слежения за ними. Это обусловлено высоким качеством работы таких систем, а также возросшими вычислительными ресурсами современной компьютерной техники. В связи с этим актуальным является и разработка алгоритмов управления роботами, которые бы использовали в своей работе данные, полученные системами распознавания объектов.

1 Техническое задание

Для успешной разработки и реализации алгоритма необходимо до начала работ выставить четкие требования к результату работы, что позволит избежать лишних трудовых и временных затрат. В данном разделе поставлены требования к конечному программному обеспечению.

1.1 Основные задачи и цели

Целью данной работы является разработка программного обеспечения для системы автоматического управления, которое позволит управлять роботом при решении задачи слежения за человеком

Программа будет способна решать следующие задачи:

- 1) прием данных от вспомогательных систем и датчиков;
- 2) расчет величин управляющих сигналов на основе принятых данных;
- 3) формирование сигналов управления с помощью ШИМ.

1.2 Назначение разрабатываемой программы

Данная программа разрабатывается для применения в тыловых работах, под управлением которой они смогут следовать за солдатами в автоматическом режиме. Это позволит отказаться от ручного управления, и соответственно освободит солдат для выполнения основных обязанностей.

1.3 Требования к функциям программы

Основные функции программы представлены в виде нумерованного списка:

- 1) прием координат человека от системы технического зрения;
- 2) прием данных о расстоянии до человека от датчиков расстояния;
- 3) прием данных о скорости от инкрементальных энкодеров;
- 4) расчет величин управляющих сигналов, сформированных ПИД-регуляторами, на основе принятых данных;
- 5) формирование сигналов управления с помощью ШИМ;
- 6) поддержка заданного расстояния до объекта равного 2 метра;
- 7) слежение за объектом с точностью 15%.

1.4 Требования к техническому обеспечению

Для разработки программы необходим персональный компьютер или ноутбук. Для выполнения программы необходим микроконтроллер, оснащенный следующими периферийными устройствами: таймеры, блоки ввода-вывода, интерфейс UART . Также для работы алгоритма необходимы электроприводы на основе двигателя постоянного тока и червячного редуктора, инкрементальные оптические энкодеры на каждое колесо и датчики расстояния.

2 Объект управления

Объектом, для управления которым разрабатывается алгоритм управления, является прототип тылового робота. Данный прототип оснащен двумя мотор-редукторами с колесами, системой технического зрения, а также следующим набором датчиков: оптические инкрементальные энкодеры Autronics, дальномеры HC-SR04. Для управления электроприводами предназначен микроконтроллер STM32F4-Discovery. Внешний вид робототехнической платформы представлен на рисунке 1.

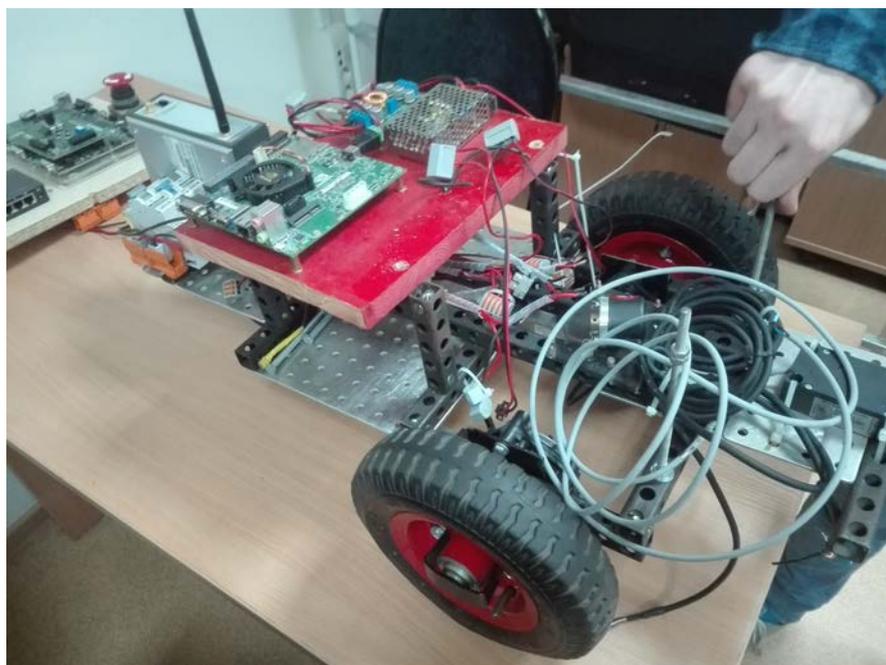


Рисунок 1 – Прототип мобильной робототехнической платформы

2.1 Мотор-редукторы

В качестве электропривода мобильного робота используется мотор-редуктор (рисунок 2) производства Калужского завода автомобильного электрооборудования. Мотор-редуктор состоит из двигателя постоянного тока и червячного редуктора.



Рисунок 2 – Мотор-редуктор

Главным его достоинством является низкая цена и достаточно высокий крутящий момент, обеспечиваемый использованием червячной передачи, что делает его хорошим вариантом для разработки бюджетной демонстрационной версии робота. Недостатком данного электропривода является отсутствие документации и информации о ряде характеристик электропривода. Характеристики электропривода отражены в таблице 1.

Таблица 1 – Характеристики электропривода

Параметр	Величина
Напряжение питания, В	12
Мощность, Вт	30
Момент на валу, Н * м	3
Масса, кг	0.95
Цена, руб	1 330

2.2 Система технического зрения

Система технического зрения робота включает в себя одноплатный микрокомпьютер NVidiaJetsonTK1 и камеруLogitechC310. Кроме того в состав СТЗ входит программное обеспечение, основанное на нейронных сетях и алгоритмах слежения, позволяющее распознать человека, за которым должен следовать робот, и определять его положение на кадре. Определив

положение человека, данное ПО вычисляет горизонтальное смещение центра человека относительно центра кадра и передает данное значение по интерфейсу UART в формате целого числа, характеризующее расстояние в пикселях. Характеристики СТЗ приведены в таблице 2.

Таблица 2 – Характеристики СТЗ

Параметр	Величина
Разрешение видеокамеры, пиксели	1280x720
Угол обзора видеокамеры, градусы	60

2.3 Оптические энкодеры

Для определения скорости вращения колес робота, на вал каждого электропривода установлен инкрементальный оптический энкодер производства компании Autonics серии E40 (рисунок 3).



Рисунок 3 – Энкодер Autonics E40

Характеристики данного энкодера представлены в таблице 3.

Таблица 3 – Параметры энкодера Autonics E40

Параметр	Величина
Напряжение питания, В	5
Разрешение, имп/об	3000
Максимально допустимая скорость	5000

вращения, об/мин	
Частота передачи сигнала, кГц	300

2.4 Ультразвуковые дальнометры

Данные датчики используются для определения расстояния до распознанного объекта. Характеристики ультразвукового датчика HC-SR04 (рисунок 4) представлены в таблице 4.



Рисунок 4 – Датчик HC-SR04

Таблица 4 – Параметры датчика HC-SR04

Параметр	Величина
Напряжение питания, В	5
Частота ультразвукового сигнала, кГц	40
Угол обзора, градусов	15
Максимальное измеряемое расстояние, м	5

Хотя данные датчики обладают небольшой дальностью и не очень высокой точностью, они подходят для применения в бюджетном прототипе. Принцип действия ультразвукового датчика представлен на рисунке 5.

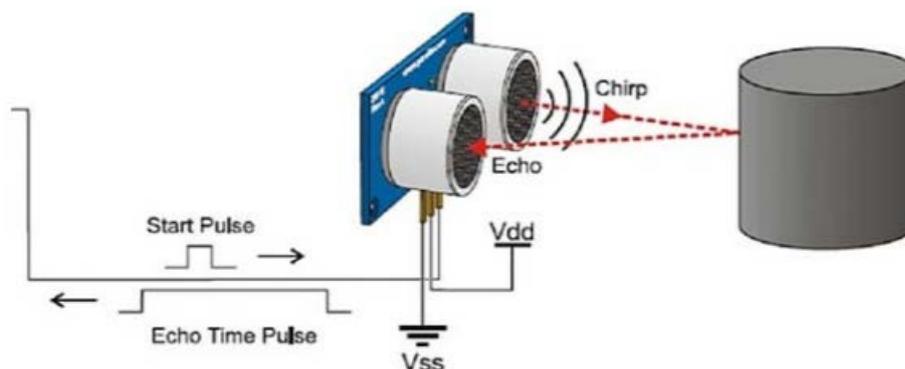


Рисунок 5 – Принцип действия ультразвукового датчика расстояния

2.5 Микроконтроллер

Используемый микроконтроллер STM32F4-Discovery (рисунок 6) предназначен для обработки сигналов с датчиков и управления электроприводами. Данный микроконтроллер построен на базе архитектуры ARM. Характеристики микроконтроллера представлены в таблице 5.



Рисунок 6 – Микроконтроллер STM32F4-Discovery

Таблица 5 – Параметры STM32F4-Discovery

Параметр	Величина
Напряжение питания, В	2...3.6
Тактовая частота, МГц	168
Число 16-битных таймеров с ШИМ	4
Разрядность, бит	32

3 Обзор методов управления

В современной теории автоматического управления накопился огромный опыт применения различных методов управления техническими объектами, в том числе и роботами. К ним можно отнести как уже давно зарекомендовавшие себя ПИД-регуляторы, так и довольно новые интеллектуальные регуляторы, основанные на нейронных сетях. Далее представлен краткий обзор данных методов, их преимуществ и недостатков.

3.1 ПИД-регулятор

ПИД-регулятором называют регулятор с обратной связью, который формирует управляющий сигнал, являющийся суммой трёх компонент, соответственно пропорциональных сигналу рассогласования, его интегралу и производной. Передаточная функция ПИД-регулятора имеет следующий вид:

$$W_{\text{ПИД}}(s) = K_p + \frac{K_I}{s} + K_D s$$

На практике Д-составляющую в таком виде получить невозможно в связи с принципом причинности, и её описывают следующая передаточная функция:

$$W_D(s) = \frac{K_D s}{\tau_d s + 1}$$

При этом, τ_d как правило, намного меньше, чем постоянная времени объекта и ею можно пренебречь.

Распространенность использования ПИД-регуляторов можно объяснить высоким качеством управления различными процессами в довольно широком диапазоне рабочих режимов. Кроме этого, ПИД-регуляторы имеют простой математический аппарат, а для их настройки существует большое количество методов и различного программного обеспечения.

Тем не менее, ПИД-регуляторы не лишены недостатков. На практике объекты управления могут быть нелинейными и/или

нестационарными. Реальные ПИД-регуляторы также нелинейны, т.к. их выходной сигнал имеет ограничения снизу и сверху.

3.2 Нейросетевой контроллер

Искусственная нейронная сеть (ИНС) — математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма. Состоит нейронная сеть из элементарных единиц – нейронов. Нейрон — это вычислительная единица, которая получает информацию, производит над ней простые вычисления и передает ее дальше. Они делятся на три основных типа (рисунок 7): входной (синий), скрытый (красный) и выходной (зеленый).

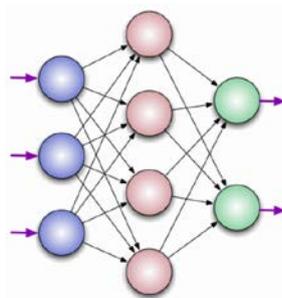


Рисунок 7 – Модель ИНС

Соответственно, есть входной слой, который получает информацию, n скрытых слоев, которые ее обрабатывают и выходной слой, который выводит результат.

При работе нейронных сетей существует два этапа – обучение и эксплуатация. В ходе первого происходит настройка весовых коэффициентов нейронов путем посредством расчета ошибки, полученной при прогоне через сеть обучающих данных. В процессе эксплуатации нейронной сети веса, как правило, остаются неизменными.

Схема применения дискретного нейроконтроллера представлена на рисунке 8.

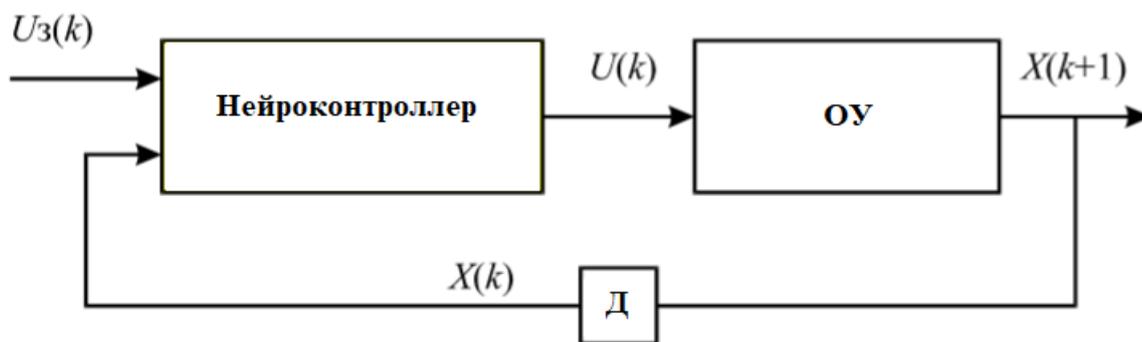


Рисунок 8 – Управление объектом с помощью нейроконтроллера

Для обучения нейроконтроллера может применяться как метод обратного распространения ошибки, так и генетические алгоритмы.

К достоинствам нейронного контроллера можно отнести нелинейность нейронных сетей, лежащих в его основе, что позволяет управлять сложными нелинейными объектами. Кроме того, настройка нейронной сети требует только обучающие данные и исключает ручную настройку.

К недостаткам же стоит отнести следующее:

1. Сложность при подборе архитектуры нейронной сети;
2. Сильная зависимость от обучающих данных;
3. Возможность переобучения;
4. Высокие требования к вычислительным ресурсам.

3.3 Выбор метода управления

Проанализировав два распространенных метода управления техническими объектами и сравнив их достоинства и недостатки, было принято решение использовать ПИД-регулирование. Данный выбор основывается на условии, что двигатель постоянного является линейной системой в рабочем диапазоне. Также учитывается относительная простота использования ПИД-регуляторов не меньшая вычислительная сложность по сравнению с двумя другими методами. Кроме того, в процессе обучения наибольший опыт работы был получен при работе именно с ПИД-регуляторами.

4 Идентификация электропривода

Так как параметры используемого электропривода неизвестны, для построения его математической модели требуется их установить.

Для построения математической модели неизвестных систем применяются методы идентификации систем. Данные методы применяются, когда параметры объекта управления заранее неизвестны, а также когда параметры могут меняться в течение времени.

Общий подход идентификации систем строится на следующих элементах:

1. Данные
2. Множество моделей-кандидатов
3. Правило, позволяющее выбрать модель для использования

Данные с входов и выходов часто регистрируются в процессе проведения экспериментов. При этом исследователь может выбрать сигналы и моменты времени для их измерения. Часть входных сигналов могут быть управляемыми. Основная задача планирования идентификационного эксперимента состоит в выборе наиболее информативных данных о сигналах системы, учитывая потенциальные ограничения. Стоит учитывать и то, что исследователь может быть лишен возможности влиять на ход эксперимента, а также должен учитывать условия нормальной эксплуатации объекта.

Множество моделей-кандидатов устанавливается фиксацией группы моделей, среди которых планируется искать подходящую. Данная часть является одной из самых важных. Также она является и наиболее трудоемкой. На данном этапе знание формальных свойств моделей соединяется с априорным знанием. Множество моделей иногда становятся результатом тщательного моделирования. После этого на основе законов физики и других знаний формируется модель, которая включает в себя параметры с неопределенными значениями. Альтернативным вариантом является использование стандартных линейных моделей без дополнительного обоснования. Множество таких моделей таких моделей,

параметры которых являются лишь средство подстройки и не отражают физики процесса, принято называть черным ящиком. Если же параметры допускают физическую интерпретацию, то такую модель называют серым ящиком.

Определение на основе экспериментальных данных лучшей модели множества является непосредственно методом идентификации. Оценка качества полученной модели связана с изучением поведения моделей в процессе их использования для воспроизведения измерений.

4.1 Получение экспериментальных данных

Для получения экспериментальных данных с мотор-редуктора, была использована плата Arduino Uno, отвечающая за считывание данных с инкрементального энкодера, установленного на вал электропровода. Для обработки данных были написаны две программы – для платы Arduino Uno на языке C++ (Приложение В), для ПК на языке Python (Приложение Г). Экспериментальные данные снимались с периодом 0.01 секунды в течение 5 секунд при подаче на электропривод напряжения величиной 12 вольт. График переходного процесса, полученный в ходе эксперимента представлен на рисунке 3.

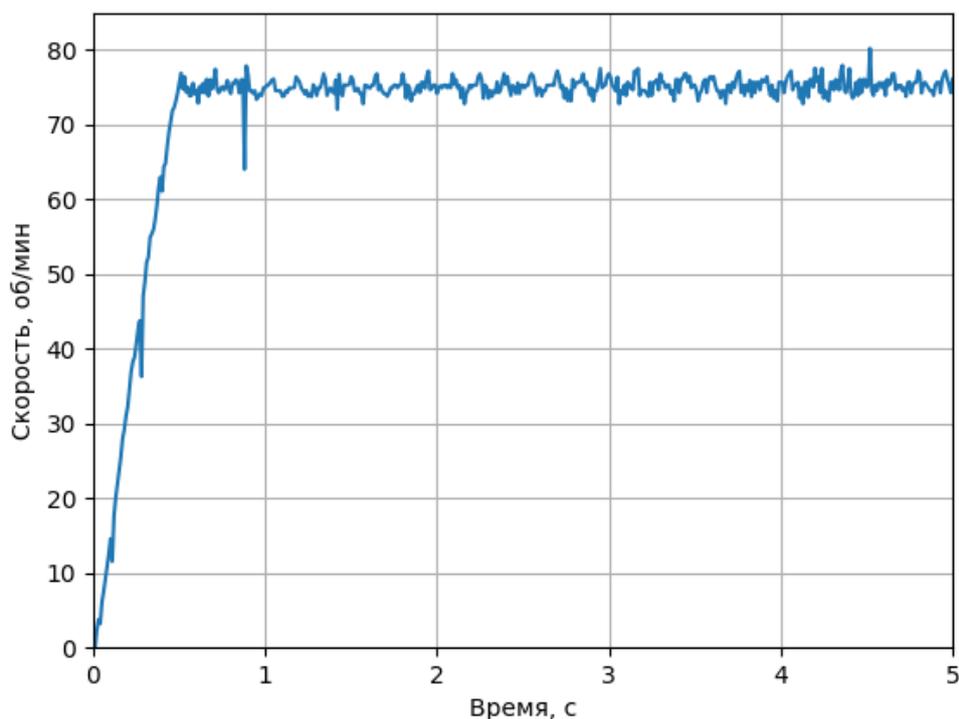


Рисунок 9 – Экспериментальные данные переходной характеристики скорости вращения электропривода

Структура электропривода на основе двигателя постоянного тока является априорной информацией, поэтому все модели-кандидаты являются серыми ящиками со следующими неизвестными параметрами: сопротивление и индуктивность якоря, постоянная ЭДС, момент инерции якоря, а также передаточное соотношение червячной передачи. Для идентификации электропривода и настройки регуляторов в пакете Simulink была построена его модель (рисунок 10).

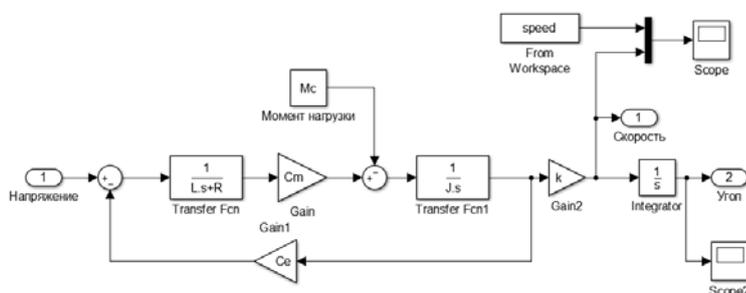


Рисунок 10 – Модель электропривода

4.2 Идентификация параметров

Сама идентификация параметров электропривода производилась с помощью пакета Simulink Design Optimization. В данном пакете решалась задача построения регрессионной модели с помощью нелинейного метода наименьших квадратов, общая формула которого имеет вид:

$$\frac{\partial SSE}{\partial b_j} = 2 \sum_{i=1}^n (y_i - f(x_i, b_j)) \frac{\partial f(x_i, b_j)}{\partial b_j} = 0, \quad (j = 1, 2, \dots, m),$$

где b_j – j -ый элемент вектора параметров, описывающий кривую,

y_i – i -ый элемент вектора истинных значений,

$f(x_i, b_j)$ – предсказанное значение,

SSE – суммарная квадратичная ошибка.

Критерием оценки модели при этом служит минимизация суммарной квадратичной ошибки:

$$SSE = \sum_{i=1}^n (y_i - f(x_i))^2,$$

где y_i – i -ый элемент вектора истинных значений,

$f(x_i)$ – i -ый элемент вектора предсказанных значений.

Также для решения данной задачи возможно использование метода градиентного поиска, но в данном случае его применение дало более высокую величину суммарной квадратичной ошибки.

4.3 Результаты идентификации

В результате идентификации были получены параметры электропривода, модель которого наиболее приближена к реальному объекту.

Изменение параметров в процессе идентификации представлено на рисунке 11.

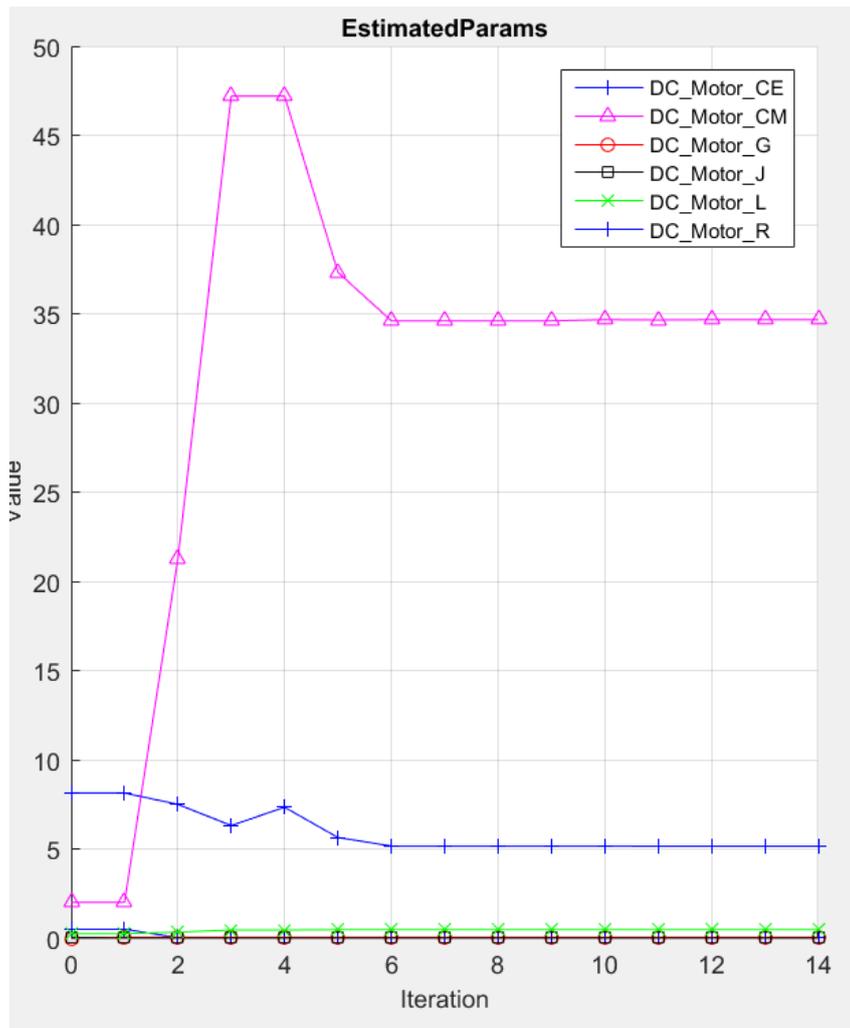


Рисунок 11 – Изменение идентифицируемых параметров

График изменения суммарной квадратичной ошибки идентификации показан на рисунке 12.

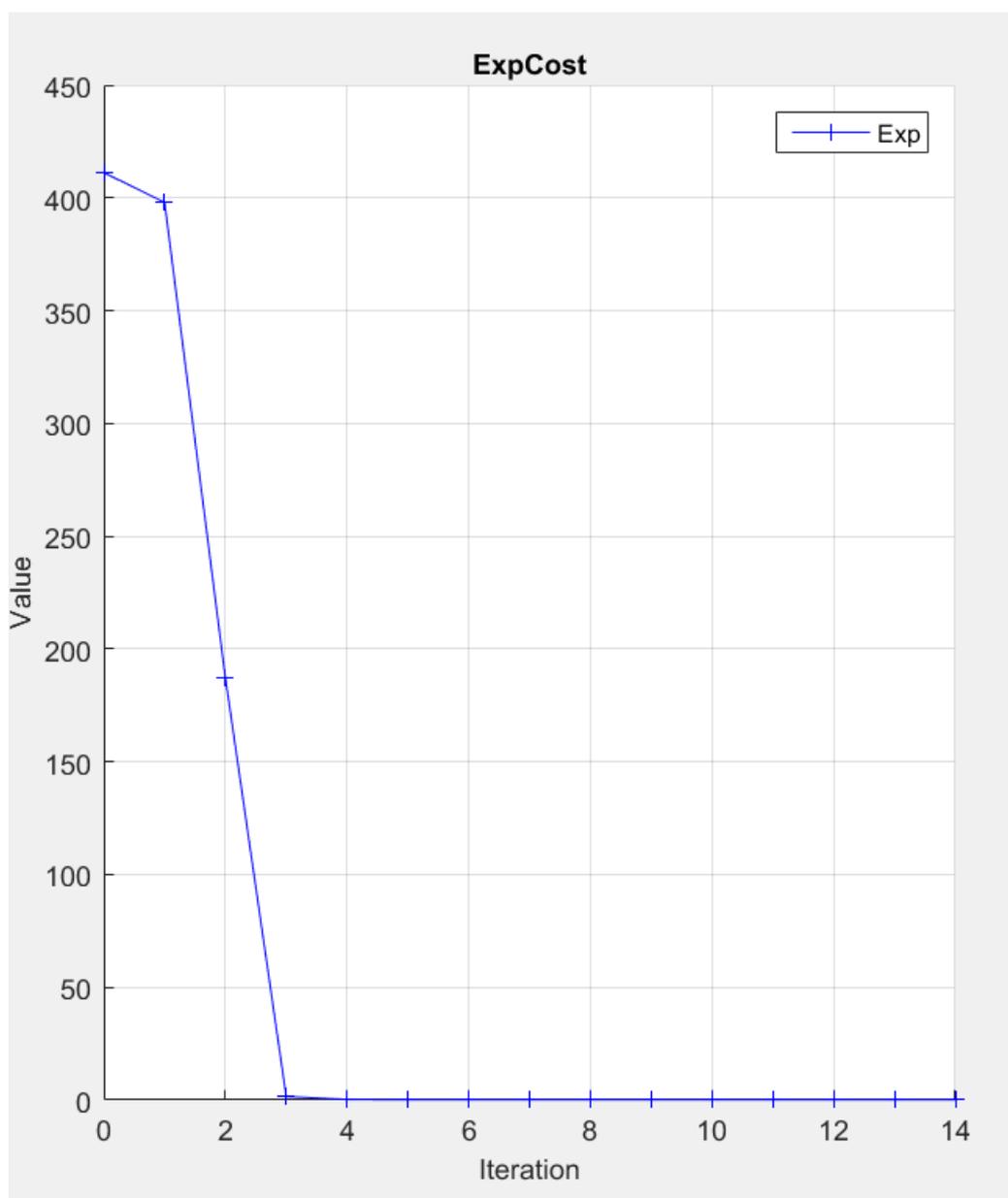


Рисунок 12 – Изменение суммарной квадратичной ошибки

Переходные характеристики при входящем напряжении равном 12 вольт модели (оранжевый график) и реального электропривода (синий график) представлены на рисунке 13.

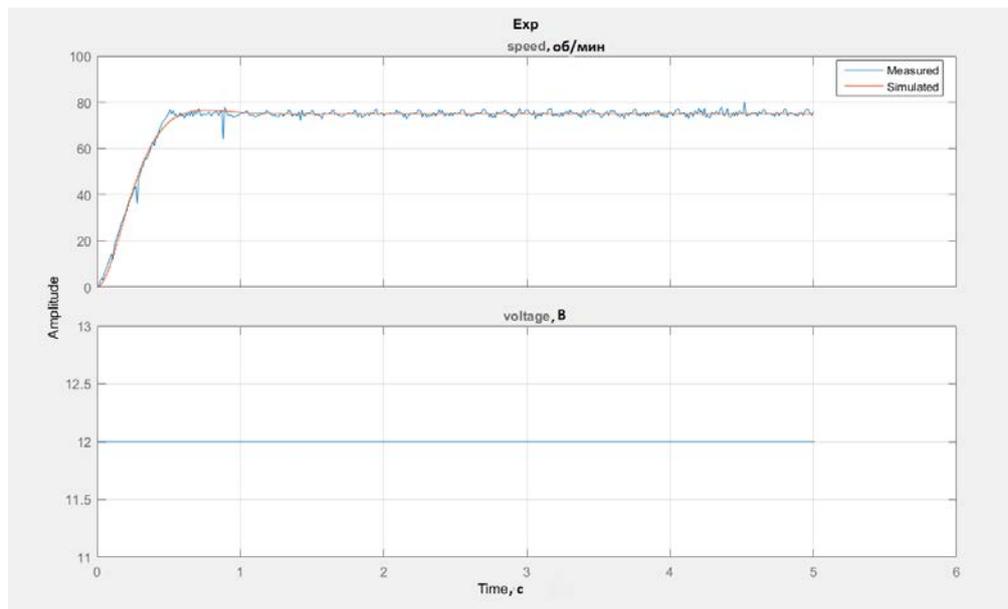


Рисунок 13 – Переходные характеристики модели и реального объекта
 Параметры, полученные в ходе идентификации, представлены в
 таблице 6.

Таблица 6 – Параметры электропривода.

Параметр	Величина
Сопротивление якоря, Ом	5.15
Индуктивность якоря, Гн	0.48
Электрическая постоянная, В/(рад/с)	0.234
Механическая постоянная, Н*м/А	34.67
Момент инерции якоря, кг/м ²	0.0128
Передаточное число червячной передачи	0.05

5 Синтез регуляторов

Для управления роботом было решено использовать ПИД-регуляторы для следующих величин: скорости вращения приводов, расстояние до объекта и угол отклонения объекта. Таким образом, алгоритм управления включает в себя четыре ПИД-регулятора.

Настройка ПИД-регуляторов с помощью математической модели электроприводов с использованием инструмента PID Tuner, включенного в пакет Simulink.

5.1 Синтез ПИД-регулятора скорости

Для синтеза регулятора скорости использовалась непосредственно модель электропривода, полученная выше. Схема управления скоростью показана на рисунке 14.

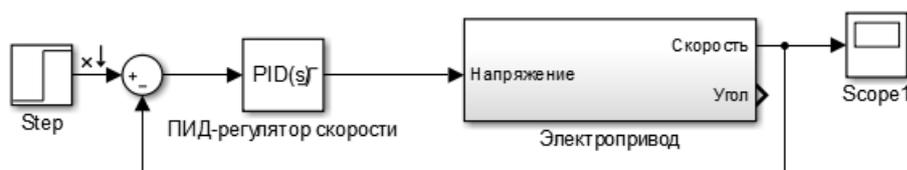


Рисунок 14 – Схема контура управления скоростью

В инструменте PID Tuner ПИД-регулятор настраивается путем передвижения ползунков характера переходного процесса и скорости переходного процесса. В данном случае регулятор был настроен на максимальную скорость переходного процесса (рисунок 15). При этом робастность была таким образом, чтобы минимизировать перерегулирование. В конечном итоге было получено время переходного процесса равное 0.0252 секунды и перерегулирование равное 4.53%. Данные показателя являются удовлетворительными для разрабатываемого алгоритма.

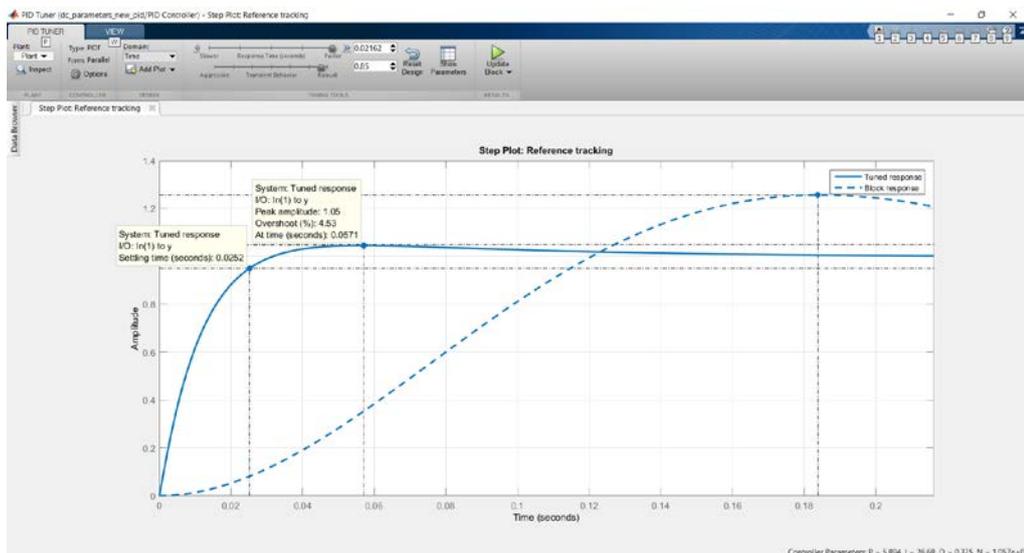


Рисунок 15 – Переходная характеристика после настройки регулятора скорости

Коэффициенты полученного регулятора скорости представлены в таблице 7.

Таблица 7 – Коэффициенты ПИД-регулятора скорости

Пропорциональный	Интегральный	Дифференциальный
5.9	26.68	0.352

5.2 Синтез ПИД-регулятора расстояния

Для синтеза регулятора расстояния к модели электропривода было добавлено вычисление расстояния, пройденного колесом робота. Было смоделировано приближение на 1 метр к объекту, удаленному на 2 метра (рисунок 16). Перемещение колесо было рассчитано по формуле:

$$S = 2 * \alpha * R,$$

где α – угол пройденный колесом,

$R = 0.1$ м – радиус колеса робота.

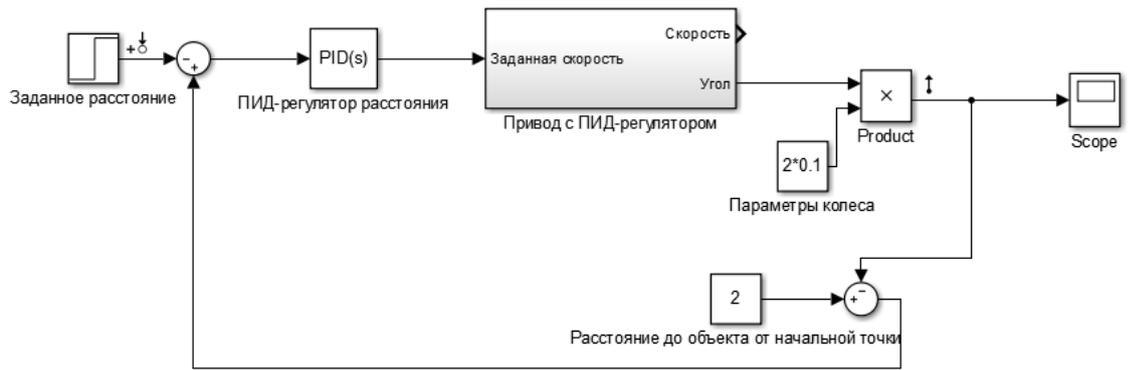


Рисунок 16 – Схема управления расстоянием

Настройка регулятора расстояния до объекта было произведена таким же образом, как и для регулятора скорости. Переходная функция контура расстояния показана на рисунке 17. Перерегулирование составило 1.05%, время переходного процесса – 0.0542 секунды.

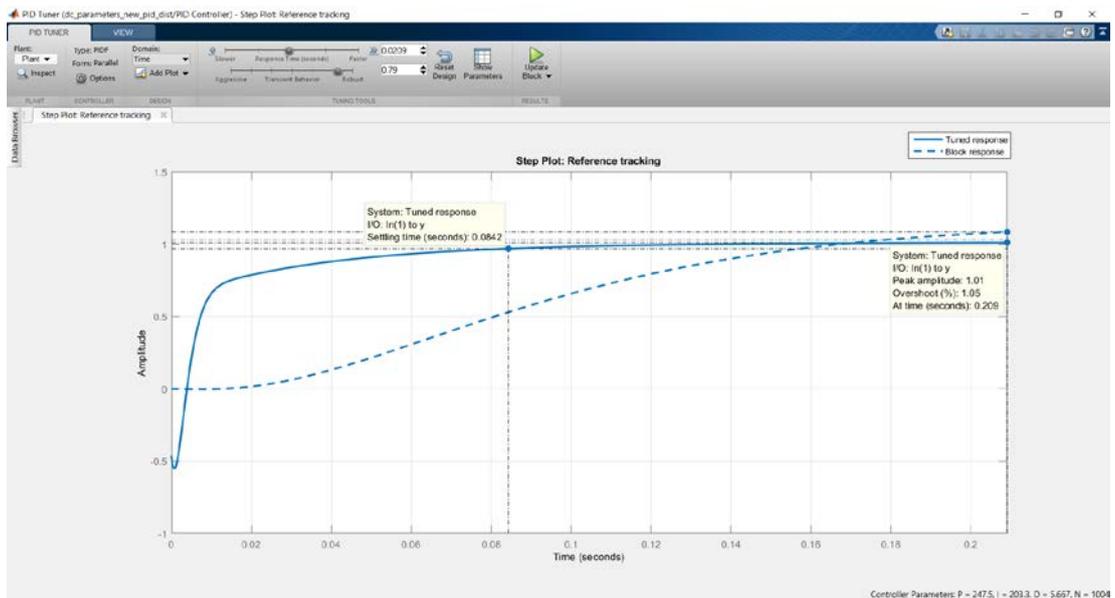


Рисунок 17 – Переходная характеристика после настройки регулятора расстояния

Настроенные коэффициенты ПИД-регулятора расстояния представлены в таблице 8.

Таблица 8 – Коэффициенты ПИД-регулятора расстояния

Пропорциональный	Интегральный	Дифференциальный
49.47	98.39	-0.87

5.3 Синтез ПИД-регулятора угла

Регулятор угла настраивался в последнюю очередь с помощью модели (рисунок 18), включающей в себя два электропривода.

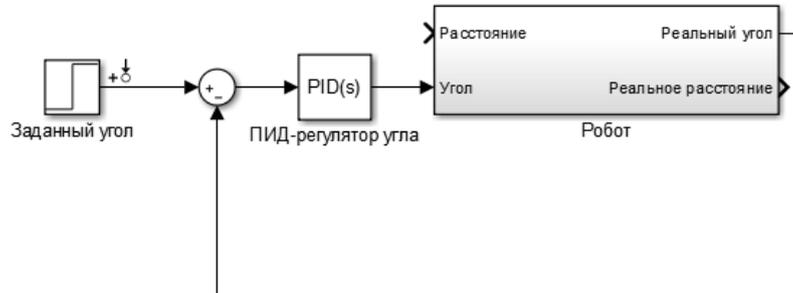


Рисунок 18 – Схема управление углом поворота робота

Также было смоделирован поворот робота (рисунок 19) робота. Расчет смещения робота был произведен по формуле:

$$\theta = R(2 * b),$$

где $R = 0.1$ м – радиус колеса робота;

$b = 0.2$ м – расстояние между колесами робота.



Рисунок 19 – Модель поворота робота

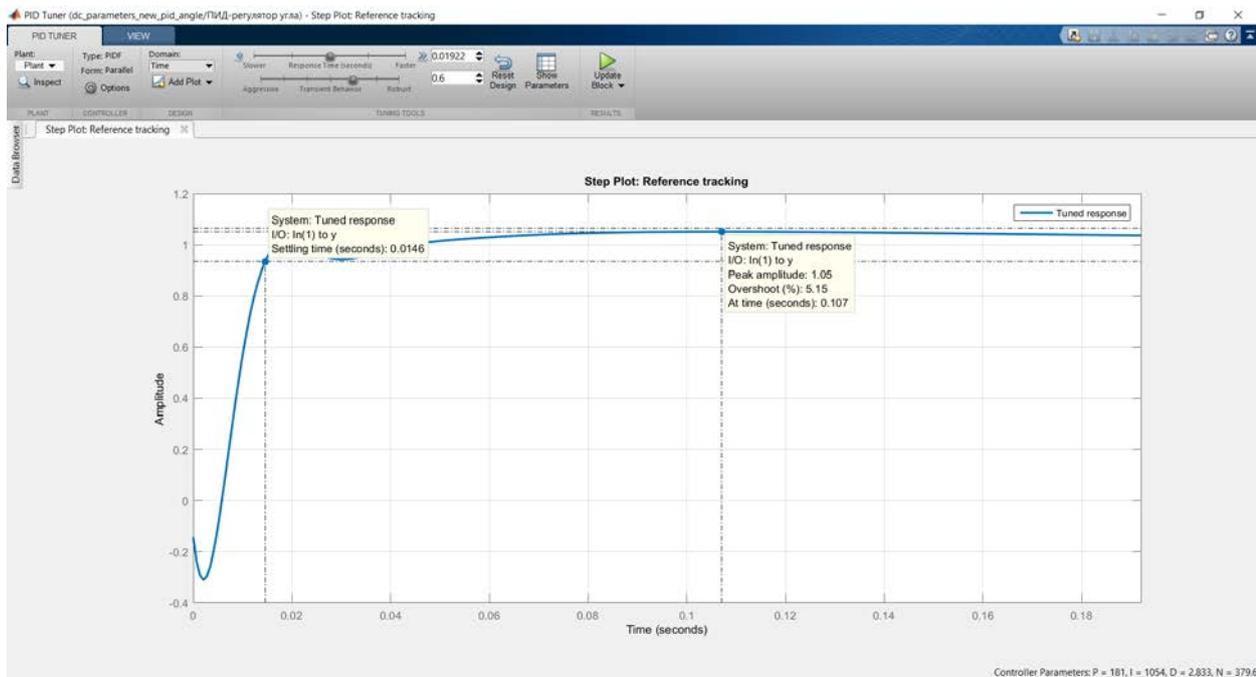


Рисунок 20 – Переходная характеристика после настройки регулятора угла

Коэффициенты настроенного регулятора угла представлены в таблице 9.

Таблица 9 – Коэффициенты ПИД-регулятора скорости

Пропорциональный	Интегральный	Дифференциальный
181	1053	2.833

6 Прием данных от системы технического зрения

Для приема данных об угле отклонения объекта слежения относительно мобильного робота применяется интерфейс UART. Данный модуль отвечает за прием данных о расстоянии до объекта от датчика расстояния и координаты смещения объекта относительно оптического центра камеры от микрокомпьютера NVidia JetsonTK1. Прием данных от микрокомпьютера осуществляется по стандарту RS-232, который представляет собой асинхронный интерфейс UART.

6.1 Интерфейс передачи данных UART

Универсальный асинхронный приёмопередатчик (англ. UniversalAsynchronousReceiver-Transmitter, UART) – узел вычислительных устройств, предназначенный для организации связи с другими цифровыми устройствами. Преобразует передаваемые данные в последовательный вид так, чтобы было возможно передать их по одной физической цифровой линии другому аналогичному устройству. Метод преобразования хорошо стандартизован и широко применяется в компьютерной технике, в том числе во встраиваемых устройствах. UART может представлять собой отдельную микросхему или являться частью микроконтроллера.

Передача данных в UART осуществляется по одному биту в равные промежутки времени. Этот временной промежуток определяется заданной скоростью UART и для конкретного соединения указывается в бодах, что соответствует величине бит в секунду. Существует общепринятый ряд стандартных скоростей: 300; 600; 1200; 2400; 4800; 9600; 19200; 38400; 57600; 115200; 230400; 460800; 921600 бод.

Помимо информационных бит, UART автоматически вставляет в поток синхронизирующие метки, так называемые стартовый и стоповый биты. При приёме эти лишние биты удаляются из потока. Обычно стартовый и стоповый биты обрамляют один байт информации, т.е. 8 бит, при этом младший информационный бит передаётся первым, сразу после стартового.

Встречаются реализации UART, передающие по 5, 6, 7, или 9

информационных бит. Обрамленные стартом и стопом биты являются минимальной посылкой. Некоторые реализации UART используют два стоповых бита при передаче для уменьшения вероятности рассинхронизации приёмника и передатчика при плотном трафике. Приёмник игнорирует второй стоповый бит, воспринимая его как короткую паузу на линии.

Принято соглашение, что пассивным (в отсутствие потока данных) состоянием входа и выхода UART является логическая 1. Стартовый бит всегда логический 0, поэтому приёмник UART ждёт перепада из 1 в 0 и отсчитывает от него временной промежуток в половину длительности бита (середина передачи стартового бита). Если в этот момент на входе всё ещё 0, то запускается процесс приёма минимальной посылки. Для этого приёмник отсчитывает 9 битовых длительностей подряд (для 8-битных данных) и в каждый момент фиксирует состояние входа. Первые 8 значений являются принятыми данными, последнее значение проверочное (стоп-бит). Значение стоп-бита всегда 1, если реально принятое значение иное, UART фиксирует ошибку. Формат передаваемых по UART данных представлен на рисунке 21.



Рисунок 21 – Формат данных UART

Для формирования временных интервалов передающий и приёмный UART имеют источник точного времени (тактирования). Точность этого источника должна быть такой, чтобы сумма погрешностей (приёмника и передатчика) установки временного интервала от начала стартового

импульса до середины стопового импульса не превышала половины (а лучше хотя бы четверти) битового интервала.[1] Для 8-битной посылки $0,5/9,5 = 5\%$ (в реальности не более 3%). Поскольку эта сумма ошибок приёмника и передатчика плюс возможные искажения сигнала в линии, то рекомендуемый допуск на точность тактирования UART — не более 1,5%.

Поскольку синхронизирующие биты занимают часть битового потока, то результирующая пропускная способность UART меньше скорости соединения. Например, для 8-битных посылок формата 8-N-1 синхронизирующие биты занимают 20% потока, что при физической скорости линии 115 200 бод означает полезную скорость передачи данных 92 160 бит/с или 11 520 байт/с.

6.2 Преобразование данных об отклонении объекта в угол отклонения

По интерфейсу UART передается знаковое целое число, характеризующее расстояние отклонения объекта от центра кадра в пикселях. Для использования данной информации в алгоритме управления следует преобразовать полученную информацию в величину угла отклонения. Для этого следует воспользоваться подобием схемой получения кадра изображения (рисунок б). Зная b – ширина кадра, A_1B_1 – отклонение объекта от центра кадра и q – угол обзора камеры, мы можем найти угол отклонения объекта.

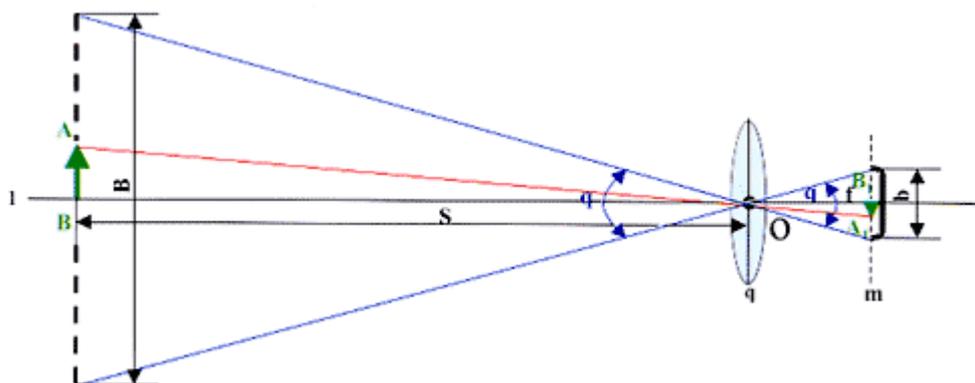


Рисунок 22 – Схема получения кадра видеокамерой

Тогда:

$$\frac{tg\varphi}{tg\frac{q}{2}} = \frac{A_1B_1}{\frac{b}{2}},$$

где φ – угол отклонения объекта.

Преобразуем выражение в следующее:

$$\varphi = arctg\left(tg\left(\frac{q}{2}\right) \frac{2A_1B_1}{b}\right)$$

Используемая в СТЗ камера имеет угол обзора равный 60° и ширину кадра равную 1280пикселей. Подставим данные значения в выражение, приведенное выше, для получения расчетной формулы:

$$\varphi = arctg\left(tg\left(\frac{60}{2}\right) \frac{A_1B_1}{\frac{1280}{2}}\right)$$

$$\varphi = arctg(0.0009 * A_1B_1)$$

Полученная формула используется в основной части разрабатываемой программы.

7 Реализация алгоритма

Для реализации алгоритма на базе микроконтроллера была выбран язык программирования C, т.к. он более высокоуровневый, чем Assembler, и требует меньших временных затрат на разработку и отладку программ. Кроме того на языке C написана удобная библиотека CMSIS, позволяющая уменьшить количество рутинных действий при работе с периферией микроконтроллера.

Также для реализации алгоритма на STM32F4-Discovery можно использовать относительно новый язык программирования Rust. Он обладает скоростью исполнения сопоставимой с языком C, а кроме того он более безопасен при работе с памятью. Тем не менее, в связи со своей молодостью, данный язык не обладает достаточной базой библиотек и других наработок.

Исходный код реализованной программы представлен в приложении Д.

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8Е41	Войцеховский Алексей Алексеевич

Институт	ИШИТР	Отделение школы (НОЦ)	ОАР
Уровень образования	Бакалавриат	Направление/специальность	15.03.06 Мехатроника и робототехника

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	- Тариф на электроэнергию – 5,257 руб./кВт·ч.; - Оклад студента – 1850 руб. в месяц; - Оклад руководителя проекта – 13824 руб. в месяц. - Человеческие ресурсы – 2 человека (руководитель и студент-дипломник).
2. <i>Нормы и нормативы расходования ресурсов</i>	- Годовая норма амортизации составляет 40 %
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	На основании пункта 1 ст.58 закона №212-ФЗ для учреждений осуществляющих образовательную и научную деятельность вводится пониженная ставка – 27,1%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	- Методы коммерциализации результатов инженерных решений; - SWOT-анализ
2. <i>Планирование и формирование бюджета научных исследований</i>	- Определение трудоемкости выполнения работ; - Расчет материальных затрат НИИ; - Основная и дополнительная зарплата исполнителей темы; - Отчисления во внебюджетные фонды; - Накладные расходы; - Проведение анализа безубыточности проекта
3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	- Расчет интегрального показателя финансовой эффективности.

Перечень графического материала (с точным указанием обязательных чертежей):

1. Оценка конкурентоспособности технических решений
2. Матрица SWOT
3. Альтернативы проведения НИ
4. График проведения и бюджет НИ
5. Оценка ресурсной, финансовой и экономической эффективности НИ

Дата выдачи задания для раздела по линейному графику	01.03.2018
---	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент	Петухов Олег Николаевич	К.Э.Н.		01.03.2018

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Е41	Войцеховский Алексей Алексеевич		01.03.2018

8 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Целью экономического раздела является проведение детального анализа проекта по критериям конкурентоспособности и ресурсоэффективности, оценка перспективности проекта, определение трудоемкости и графика работ, а также расчёт интегрального показателя ресурсоэффективности.

8.1 Потенциальные потребители результатов исследования

Для того, чтобы определить потенциальных потребителей, необходимо определить целевой рынок и провести его сегментирование. Целевым рынком является рынок военной робототехники. В качестве конечных потребителей выступают военные части.

8.2 Анализ конкурентных технических решений

Конкурентные решения используют различные датчики для определения лидера. Среди датчиков есть GPS-приемники, лидары, системы технического зрения (СТЗ). При этом для следования за лидером в большинстве случаев используются избыточное количество оборудования. С целью уменьшения количество бортового оборудования в данной работе предлагается новый способ следования. По сравнительному анализу конкурентных решений была составлена таблица 10. Полужирным шрифтом выделен сегмент, на который направлена данная работа.

Таблица 10 – Карта сегментации рынка

		Дополнительное использование СТЗ	
		Да	Нет
Первичный датчик	Лидар	Lockheed Martin SMSS	BigDog
	GPS-приёмник	HDT Global Protector	Northrop Grumman

			Camel
	СТЗ	-	Отсутствуют

8.3 Технология QuaD

Технология QuaD (QUality ADvisor) - это инструмент измерения характеристик, который описывает качество новой разработки, а также ее перспективность на рынке. Технология позволяет принимать решение о целесообразности вложения капитала в НИР и может использоваться при проведении различных маркетинговых исследований, существенным образом снижая их трудоемкость и повышая точность и достоверность результатов. Оценочная карта представлена в таблице 11.

Таблица 11 - Оценочная карта для сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение (3/4)	Средневзвешенное значение (5x2)
1	2	3	4	5	
Показатели оценки качества разработки					
1. Энергоэффективность	0,05	80	100	0,80	0,04
2. Помехоустойчивость	0,04	60	100	0,60	0,024
3. Надежность	0,07	80	100	0,80	0,056
4. Мобильность	0,16	70	100	0,70	0,16
5. Персонализация	0,06	100	100	1,00	0,06
6. Безопасность	0,16	70	100	0,70	0,16
7. Функциональная мощность	0,09	80	100	0,80	0,072
8. Простота эксплуатации	0,09	100	100	1,00	0,09
9. Качество интеллектуального интерфейса	0,08	40	100	0,40	0,034
10. Ремонтпригодность	0,08	50	100	0,50	0,044
Показатели оценки коммерческого потенциала разработки					
11. Перспективность	0.05	100	100	1,00	0,05

рынка					
12. Цена	0,07	100	100	1,00	0,07
Итого	1			9,9	0,823

Оценка качества и перспективности по технологии QuaD определяется по формуле:

$$P_{cp} = \sum P_i \cdot 100 = 0,823 \cdot 100 = 82,3, \text{ где:}$$

1) P_{cp} – средневзвешенное значение показателей качества и перспективности научной разработки;

2) P – средневзвешенное значение показателя.

Значение P_{cp} позволяет говорить о перспективах разработки и качестве проведенного исследования. Если значение показателя P_{cp} получилось от 100 до 80, то такая разработка считается перспективной. Если от 79 до 60 – то перспективность выше среднего. Если от 69 до 40 – то перспективность средняя. Если от 39 до 20 – то перспективность ниже среднего. Если 19 и ниже – то перспективность крайне низкая. Из таблицы можно сделать вывод, что разработку можно считать перспективной.

8.4 SWOT – анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – это комплексный анализ научно-исследовательского проекта. Такой анализ применяют для исследования внешней и внутренней среды проекта. Составленная матрица SWOT представлена в таблице 12.

Таблица 12 - SWOT-анализ

	Сильные стороны научно-исследовательского проекта:	Слабые стороны научно-исследовательского проекта:
	С1. Невысокая стоимость. С2. Мобильность. С3. Надежность. С4. Экономичность.	Сл1. Недостаток конструкторского опыта разработки мобильных роботов. Сл2. Отсутствие

		финансирования. Сл3. Сложность создания первого прототипа робота. Сл4. Отсутствие репутации на рынке.
Возможности: В1. Использование инновационной инфраструктуры ТПУ. В2. Заинтересованность со стороны государства. В3. Продажа продукта по частям, не только в качестве робота, но и отдельно программное обеспечение.	При использовании инфраструктуры ТПУ можно снизить стоимость проекта. Новизна разработки состоит именно в заложенном алгоритме управления, что позволяет продавать только алгоритм, а не робота целиком.	Появление дополнительного спроса на рынке может способствовать финансированию проекта. Проводя исследования на базе ТПУ можно разработать прототип и получить известность на рынке.
Угрозы: У1. Отсутствие спроса на более производительный алгоритм следования У2. Низкая скорость изготовления. У3. Введение требований к продаваемому ПО. У4. Несвоевременное финансовое обеспечение научного исследования со стороны государства.	Невысокая цена и хорошая функциональность способствуют повышению спроса. Требования государства могут быть перекрыты заданными ограничениями по надежности.	Необходимо разработать прототип для того, чтобы выйти на рынок и заполучить репутацию.

8.5 Морфологический анализ

Морфологический подход основан на систематическом исследовании всех теоретически возможных вариантов, которые вытекают из закономерностей объекта исследования. Анализ охватывает все возможные варианты. Путем комбинирования вариантов можно получить большое количество различных решений, ряд которых представляет практический

интерес. Составим таблицу 13, в которой будут отражены возможные варианты исполнения по различным проблемам разработки.

Таблица 13 - Морфологическая матрица для алгоритма компьютерного зрения

Характеристика	Варианты исполнения		
	1	2	3
Язык программирования	C	Matlab	Rust
Вычислительная платформа	Персональный компьютер	Система на чипе	Микроконтроллер
Датчик угла поворота вала электродвигателя	Оптический энкодер	Магнитный энкодер	Механический энкодер

К алгоритмам автоматического управления предъявляются требования к обработке информации с периферийных устройств в реальном времени, таких как датчики. Для этих целей требуется применение микроконтроллеров, т.к. операционные системы, работающие на ПК и системах на чипе, не позволяют в реальном времени взаимодействовать с периферийными устройствами. Для программирования микроконтроллеров применяются компилируемые языки программирования, такие как C или Rust. В программной среде Matlab имеется возможность генерации программного кода на языке C. Язык программирования Rust является перспективной заменой C с большей безопасностью и стабильностью, но на данный момент данный язык молодой и проигрывает C в количестве библиотек и поддержке микроконтроллерами. Ситуация с библиотеками имеется и в языке Matlab. Механические энкодеры имеют проблему дребезга контактов, а магнитные энкодеры более чувствительны к помехам, по

сравнению с оптическими. Поэтому в качестве варианта исполнения берётся язык программирования С, микроконтроллер и оптический энкодер.

8.6 Планирование научно-исследовательских работ

8.6.1 Структура работ в рамках научного исследования

Для организации НИР применяются различные методы экономического планирования с целью более эффективного использования времени и рабочей силы, а также снижения трудозатрат. Планирование НИР заключается в:

1. составлении перечня работ, необходимых для достижения поставленной задачи;
2. определении участников;
3. установлении продолжительности в рабочих днях;
4. построения линейного графика и его оптимизации.

Примерный порядок этапов и работ, распределение исполнителей по данным видам работ приведен в таблице 14. В таблице и далее используются следующие сокращения:

1. НР – научный руководитель;
2. С – студент.

Таблица 14 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ этапа	Содержание работ	Исполнитель
Разработка задания	1	Постановка задачи	НР
Выбор направления исследования	2	Обзор научно-технической базы	НР, С
	3	Разработка и утверждение ТЗ	НР, С
	4	Составление календаря проекта	С
	5	Разработка вариантов исполнения проекта	НР, С
Теоретические исследования	6	Построение модели электропривода	С

	7	Разработка алгоритма работы	С
	8	Программирование и отладка алгоритма	С
Экспериментальные исследования	9	Тестирование работы алгоритма	С
	10	Обработка полученных результатов	НР, С
Оформление отчета по НИР	11	Составление пояснительной записки	С
	12	Оформление графического материала	С

На первом этапе происходит постановка цели и задачи исследования – разработка и реализация алгоритмов распознавания изображений по видеопотоку в задачах слежения за объектами. Тематика выбирается научным руководителем и обсуждается со студентом.

На втором этапе студент производит поиск научной литературы по предоставленной тематике для ознакомления и изучения необходимого материала. В дальнейшем данный материал будет использоваться для проведения исследований и разработки устройства.

На третьем этапе студент совместно с научным руководителем разрабатывают общее содержание ВКР. Данный документ является основополагающим при проведении дальнейшего исследования и разработки.

На четвертом этапе составляется календарный план выполнения работ с учётом линейного графика обучения.

На пятом этапе осуществляется разработка архитектуры программного продукта. Для этого составляются требования к функционалу программы. Далее выбирается парадигма программирования. Для упрощения

внесения будущих изменений функционал программы разбивается на несколько независимых друг от друга компонент. Для данных компонент составляются диаграммы классов и блок-схемы функций для визуализации порядка работы алгоритма.

На шестом этапе студентом пишется код для программы на выбранном языке программирования. Затем на этапе тестирования происходит оценка работы алгоритма: быстродействие, точность, робастность.

На восьмом, девятом и десятом этапе студент, под руководством научного руководителя занимается интерпретацией и обработкой результатов, а также оформлением пояснительной записки и графического материала (графические материалы результатов исследования, презентация проекта).

8.6.2 Определение трудоемкости выполнения работ

Наиболее ответственной частью экономических расчетов по теме является расчет трудоемкости работ, так как трудовые затраты составляют основную часть стоимости НИР. Под трудоемкостью работ понимают максимально допустимые затраты труда в человеко-днях на выполнение НИР с учетом организационно технических мероприятий, обеспечивающих наиболее рациональное использование выделенных ресурсов.

Так как отсутствует нормативная база по проводимым работам, а также достоверная информация о процессе выполнения подобных работ иными исполнителями, воспользуемся экспертным способом оценки продолжительности выполнения запланированных работ.

Определим ожидаемое время проведения работ, длительность этапов в рабочих и календарных днях, по формулам, воспользовавшись формулой:

$$t_{ож} = \frac{3 \cdot t_{\min} + 2 \cdot t_{\max}}{5}, \text{ где:}$$

1) $t_{ож}$ – ожидаемое время выполнения i -го этапа работ;

2) t_{\min} - минимально возможная трудоемкость выполнения заданной i -ой работы (оптимистическая оценка: в предположении наиболее благоприятного стечения обстоятельств);

3) t_{\max} - максимально возможная трудоемкость выполнения заданной i -ой работы (пессимистическая оценка: в предположении наиболее неблагоприятного стечения обстоятельств).

Ожидаемое, минимальное и максимальное время исполнения в предложенной выше формуле, оцениваются в рабочих днях на человека. Произведем перевод этих величин в календарные дни, воспользовавшись следующей формулой:

$$T_{КД} = T_{РД} \cdot T_K, \text{ где:}$$

1) $T_{КД}$ – продолжительность выполнения этапа в календарных днях;

2) T_K – коэффициент календарности, позволяющий перейти от длительности работ в рабочих днях к их аналогам в календарных днях.

Рассчитывается по формуле:

$$T_K = \frac{T_{КЛ}}{T_{КЛ} - T_{ВД} - T_{ПД}}, \text{ где:}$$

1) $T_{КЛ}$ – календарные дни ($T_{КЛ} = 365$);

2) $T_{ВД}$ – выходные дни ($T_{ВД} = 52$);

3) $T_{ПД}$ – праздничные дни ($T_{ПД} = 10$).

В свою очередь рабочие дни рассчитываются по следующей формуле:

$$T_{РД} = \frac{t_{ожс}}{K_{ВН}} \cdot K_D, \text{ где:}$$

1) $K_{ВН}$ – коэффициент выполнения работ, учитывающий влияние внешних факторов на соблюдение предварительно определенных длительностей;

K_D – коэффициент, учитывающий дополнительное время на компенсацию непредвиденных задержек и согласование работ ($K_D = 1 - 1.2$; в этих границах конкретное значение принимает сам исполнитель).

Для простоты расчетов примем K_D и $K_{ВН}$, равными единице. Тогда формула для расчета календарных дней преобразуется в следующую:

$$T_{КД} = T_{РД} \cdot T_K = t_{ож} \cdot T_K = \frac{3 \cdot t_{\min} + 2 \cdot t_{\max}}{5}$$

Воспользовавшись данными из таблицы 4, приведенными выше формулами, произведем расчет продолжительности выполнения работ студентом в календарных днях. Результаты расчетов представлены в таблице 15.

Таблица 15 - Временные показатели проведения научного исследования

Название работы	Трудоемкость работ, чел-дни						Длительность работ в рабочих днях $T_{РД}$		Длительность работ в календарных днях $T_{КД}$	
	t_{\min}		t_{\max}		$t_{ож}$		Одновременное выполнение работ		Одновременное выполнение работ	
	Студент	Научный руководитель	Студент	Научный руководитель	Студент	Научный руководитель	Студент	Научный руководитель	Студент	Научный руководитель
Постановка задачи	5	3	8	6	6,2	4,2	3,1	2,1	5	1
Обзор научно-технической базы	7	2	10	4	12	2,8	4,5	1,4	5	3
Разработка и утверждение	7	1	12	2	9	1,2	4,5	0,7	7	2

ТЗ										
Составление календаря проекта	3	0	5	0	3,8	0	3,8	0	6	0
Разработка вариантов исполнения проекта	9	2	16	7	11,8	3,2	5,9	2,6	8	4
Построение модели электропривода	3	0	8	0	5	0	5	0	8	0
Разработка алгоритма работы	7	0	14	0	9,8	0	9,8	0	15	0
Программирование и отладка алгоритма	7	0	10	0	8,2	0	8,2	0	13	0
Тестирование работы алгоритма	3	0	7	0	4,6	0	4,6	0	7	0
Обработка полученных результатов	6	4	14	8	9,2	5,6	4,6	2,8	7	5
Составление пояснительной записки	5	0	10	0	7	0	7	0	11	0
Оформление	10	5	14	8	11,6	6,2	5,8	3,1	9	5

графического материала										
Итого									114	34

8.6.3 Разработка графика проведения научного исследования

По данным из таблицы 15 создадим диаграмму Ганта.

Таблица 16 - Календарный план-график проведения НИОКР

№	Этап	Исполнитель	$T_{кд}$	Продолжительность выполнения работ														
				Февраль		Март		Апрель		Май		Июнь						
1	Постановка задачи	НР	1	1														
		С	5	2														
2	Обзор научно-технической базы	НР	3		1													
		С	7	1	2													
3	Разработка и утверждение ТЗ	НР	2			1												
		С	7	1	2													
4	Составление календаря проекта	С	6			1	1											
5	Разработка вариантов исполнения проекта	НР	4				1											
		С	8	1	2													
6	Построение модели электропривода	С	8				1	1										
7	Разработка алгоритма работы	С	15						1	1								
8	Программирование и отладка алгоритма	С	13							1	1							
9	Тестирование работы алгоритма	С	7								1	1						
10	Обработка	НР	5									1						
		С	7										1	1				

	полученных результатов																	
11	Составление пояснительной записки	С	11															
12	Оформление графического материала	НР	5															
		С	9															

8.7 Бюджет научно-технического исследования

Бюджет научно-технического исследования должен быть основан на достоверном отображении всех видов расходов, связанных выполнением проекта. В процессе формирования бюджета разработки используется следующая группировка затрат по статьям:

1. материальные затраты разработки;
2. основная заработная плата исполнителей темы;
3. дополнительная заработная плата исполнителей темы;
4. отчисления во внебюджетные фонды (страховые отчисления);
5. затраты на научные и производственные командировки;
6. накладные расходы.

8.7.1 Расчет материальных затрат

Для вычисления материальных затрат воспользуемся следующей формулой:

$$Z_M = (1 + k_t) \cdot \sum_{i=1}^m C_i \cdot N_{расх.}, \text{ где:}$$

- 1) m – количество видов материальных ресурсов;
- 2) C_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов;
- 3) $N_{расх.}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования;

4) k_t – коэффициент, учитывающий транспортно-заготовительные расходы.

Таблица 17 - Материальные затраты

Наименование	Единица измерения	Количество	Цена, руб.
ПК	шт.	1	25000
Лист А4	шт.	100	200
Ручка	шт.	3	90
Микроконтроллер	шт.	1	2000
Итого, руб.	27290		

8.7.2 Основная заработная плата исполнителям темы

Статья включает основную заработную плату работников, непосредственно занятых выполнением НИИ, (включая премии, доплаты) и дополнительную заработную плату. Она рассчитывается по формуле

$$Z_{ЗП} = Z_{осн.} + Z_{доп.}, \text{ где:}$$

- 1) $Z_{осн.}$ – основная заработная плата;
- 2) $Z_{доп.}$ – дополнительная заработная плата (12-20 % от $Z_{осн.}$).

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{дн.} = \frac{Z_M \cdot M}{F_D}, \text{ где:}$$

- 1) Z_M – месячный должностной оклад работника, руб.;
- 2) M – количество месяцев работы без отпуска в течение года;
- 3) F_D – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн.

Таблица 18 - Баланс рабочего времени

Показатели рабочего времени	Руководитель	Студент
-----------------------------	--------------	---------

Календарное число дней	365	365
Количество нерабочих дней - выходные дни - праздничные дни	120	120
Потери рабочего времени - отпуск - невыходы по болезни	48	72
Действительный годовой фонд рабочего времени	197	173

Месячный оклад работника рассчитывается по формуле:

$$Z_M = Z_{TC} \cdot (1 + k_{np} + k_{\delta}) \cdot k_p, \text{ где:}$$

- 1) Z_{TC} – заработная плата по тарифной ставке, руб.;
- 2) k_{np} – премиальный коэффициент, равный 0,3 (т.е. 30% от Z_{TC});
- 3) k_{δ} – коэффициент доплат и надбавок составляет примерно 0,2 – 0,5 (в НИИ и на промышленных предприятиях – за расширение сфер обслуживания, за профессиональное мастерство, за вредные условия: 15-20% от Z_{TC});
- 4) k_p – районный коэффициент, равный 1,3 (для Томска).

Расчёт основной заработной платы приведён в таблице 19.

Таблица 19 - Расчет основной заработной платы

Исполнители	Разряд	Z_{TC} , руб.	k_{np}	k_{δ}	k_p	Z_M , руб.	$Z_{осн.}$, руб.	T_P , раб. дн.	$Z_{осн.}$, руб.
Руководитель	1	9489	0,3	0,2	1,3	18503,55	976,83	24	19862,52
Студент		1854	0	0	1,3	2410,2	130,95	137	13346,43
Итого $Z_{осн.}$									33208,95

8.7.3 Дополнительная заработная плата

Дополнительная заработная плата включает заработную плату за не отработанное рабочее время, но гарантированную действующим

законодательством. Расчет дополнительной заработной платы ведется по формуле:

$$З_{доп.} = k_{доп.} \cdot З_{осн.}, \text{ где:}$$

1) $k_{доп.}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 – 0,15).

$k_{доп.}$ равен 0,12. Результаты по расчетам дополнительной заработной платы сведены в таблицу 20.

Таблица 20 - Затраты на дополнительную заработную плату

Исполнители	Основная зарплата (руб.)	$k_{доп.}$	Дополнительная зарплата (руб.)
Руководитель	19862,52	0,12	2383,50
Студент	13346,43	0,12	1601,57
Итого			3985,07

8.7.4 Отчисления во внебюджетные фонды

Величина отчислений во внебюджетные фонды определяется исходя из формулы:

$$З_{внеб.} = k_{внеб.} \cdot (З_{осн.} + З_{доп.})$$

где $k_{внеб.}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

На 2018 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30%. На основании пункта 1 ст.58 закона №212-ФЗ для учреждений, осуществляющих образовательную и научную деятельность в 2018 году пониженная ставка – 27,1%. Отчисления во внебюджетные фонды представлены в таблице 21.

Таблица 21 - Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.	Дополнительная заработная плата, руб.
Руководитель проекта	19862,52	2383,50
Студент	13346,43	1601,57
Коэффициент отчислений во	27,1%	

внебюджетные фонды	
Итого	
Руководитель	5382,74
Студент	3616,88
Итого	8998,88

8.7.5 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов, оплата услуг связи, электроэнергии и т.д. Расчет накладных расходов определяется по формуле:

$$Z_{\text{нак.}} = \sum Cm \cdot k_{\text{нр}}, \text{ где:}$$

- 1) $k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы;
- 2) Cm – затраты по статьям накладных расходов.

Величину коэффициента накладных расходов можно взять в размере 10%.

$$Z_{\text{нак.}} = (27290 + 33208,95 + 3985,07 + 8998,88) \cdot 0,10 = 7347,29 \text{ руб.}$$

8.7.6 Контрагентные расходы

Контрагентные расходы включают затраты, связанные с выполнением каких-либо работ по теме сторонними организациями (контрагентами, субподрядчиками). В данном проекте отсутствует необходимость в стороннем подрядчике.

8.7.7 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта. Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в таблице 22.

Таблица 22 – Расчет бюджета затрат НТИ

Наименование статьи	Сумма, руб.
---------------------	-------------

1. Материальные затраты НИИ	27290,00
2. Затраты на заработную плату научному руководителю	22246,02
3. Затраты на заработную плату студенту	14948,00
4. Затраты на отчисления во внебюджетный фонд	8998,88
5. Накладные расходы	7347,29
Бюджет затрат НИИ	80830,19

8.8 Определение ресурсной, финансовой и экономической эффективности ресурсов

В результате исследования были определены затраты на проект по разработке алгоритма компьютерного зрения. Бюджет составляет 80 тыс. руб. Учитывая все конкурентные преимущества данного программного обеспечения, можно предположить, что продукт будет конкурентоспособным и будет иметь спрос на рынке.

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа	ФИО
8Е41	Войцеховский А.А.

Школа	ИШИТР	Отделение	ОАР
Уровень образования	Бакалавриат	Направление/специальность	Мехатроника и робототехника 15.03.06

Исходные данные к разделу «Социальная ответственность»:

1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения

Объектом исследования является алгоритм автоматического управления мобильным роботом для задач слежения за распознанными объектами. Алгоритм выполняется на микроконтроллере, используемом в мобильном роботе. Рабочим местом является место за персональным компьютером, отведенное для выполнения работы. Основным оборудованием, на котором производится работа, является ПК.

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Производственная безопасность

Анализ выявленных вредных факторов:

- Шум
- Микроклимат
- Освещение
- Напряженная зрительная работа
- Монотонность труда

Анализ выявленных опасных факторов:

- Электрический ток (источником является ПК и микроконтроллер)
- Статическое электричество
- Короткое замыкание

2. Экологическая безопасность:

Анализ воздействия на литосферу и атмосферу (образование отходов при поломке ПК, перегорании люминесцентных ламп)

3. Безопасность в чрезвычайных ситуациях:

Чрезвычайной ситуацией, которая может возникнуть на рабочем месте, является возникновение пожара.

<p>4. Правовые и организационные вопросы обеспечения безопасности.</p>	<p>Требования к организации рабочего места:</p> <ul style="list-style-type: none"> – Трудовой кодекс; – Федеральный закон N 426-ФЗ “О специальной оценке условий труда”; Приказ Минтруда N 33н "Об утверждении Методики проведения специальной оценки условий труда”; – Приказ Минздравсоцразвития N 302н "Об утверждении перечней вредных и (или) опасных производственных факторов и работ, при выполнении которых проводятся обязательные предварительные и периодические медицинские осмотры”;
---	---

Дата выдачи задания для раздела по линейному графику	01.03.2018
--	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
ассистент	Авдеева И.И.			01.03.2018

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Е41	Войцеховский А.А.		01.03.2018

9. Социальная ответственность

В данном разделе выпускной квалификационной работы рассматриваются вопросы, касающиеся социальной ответственности при разработке и реализации алгоритмов автоматического управления для задач слежения за распознанными объектами. Алгоритмы автоматического управления, как правило, исполняются с помощью микроконтроллеров, встраиваемых в системы управления.

Предлагаемые в данной работе алгоритмы позволят управлять мобильным роботом на основании информации об объекте слежения с видеокамеры, установленной на роботе.

9.1 Производственная безопасность

В процессе разработки и реализации алгоритма управления на человека могут воздействовать следующие вредные факторы:

- Микроклимат
- Шум
- Освещение
- Психофизиологические факторы: напряженная зрительная работа, монотонность труда

Опасные производственные факторы:

- Электрический ток (источником является ПК и микроконтроллер)
- Статическое электричество;
- Короткое замыкание.

9.1.1 Микроклимат

Микроклимат помещения, как правило, характеризуют тремя основными параметрами:

- температура окружающего воздуха, °С;
- относительная влажность воздуха, %;
- скорость движения воздуха, м/с;

Неблагоприятные метеоусловия в помещении могут вызывать перенапряжение механизмов терморегуляции, перегрев или переохлаждение организма, что в свою очередь негативно влияет здоровье и продуктивность труда сотрудника.

Сочетание температуры воздуха, относительной влажности воздуха и его скорости движения определяют микроклимат в помещении.

В таблице 23 представлены оптимальные параметры микроклимата согласно СанПиН 2.2.4.548–96 в помещениях для легкой категории работ, к которой относятся работы, выполняемые сидя и не требующие физического напряжения. Работа персонала в данном случае относится к категории работ Ia, с интенсивностью энергозатрат до 139 ккал/ч. Работы производятся сидя и сопровождаются незначительным физическим напряжением.

Таблица 23 – Оптимальные параметры микроклимата

Пе риод года	Темпе ратура, °С	Относи тельная влажность, %	Скорость движения воздуха, м/с, не более
Хо лодный	22-24	40-60	0,1
Те плый	23-25	40-60	0,1

К допустимым микроклиматическим нормам относят величины, которые способны вызвать изменение теплового состояния организма. При этом не возникает нарушений состояния здоровья, но могут ощущаться дискомфорт, ухудшение самочувствия и понижение работоспособности. Допустимые нормы устанавливаются в тех производственных помещениях, где по каким-либо причинам нет возможности обеспечить оптимальные условия.

Допустимые параметры для легкой категории работ представлены в таблице 24.

Таблица 24 – Допустимые параметры микроклимата

Пе	Температура	Относи	Скорость движения
----	-------------	--------	-------------------

Период года	t, °C		Относительная влажность, %	Скорость воздуха, м/с, не более	
	Нижнее оптимальное значение	Верхнее оптимальное значение		Нижнее оптимальное значение	Верхнее оптимальное значение
Холодный	2 0-21	2 4,1-25	15-75	0,1	0,1
Теплый	2 1-22	2 5,1-26	15-75	0,1	0,1

Нормативные значения параметров микроклимата в помещении, где находится рабочее место, в течение всего года поддерживаются благодаря установленному кондиционеру и центральному отоплению.

9.1.2 Шум

При выполнении работ устройством работник может подвергаться воздействию шума от работающего персонального компьютера. Продолжительное воздействие шума может привести к шумовой болезни – заболеванию организма, которое сопровождается поражением органов слуха и центральной нервной системы. Также шум вызывает головную боль, раздражительность, быструю утомляемость, потерю слуха, повышение артериального давления. Допустимые шумовые характеристики рабочих мест регламентируются СанПин 2.2.4/2.1.8.562-96 и представлены в таблице 25.

Таблица 25 – Допустимые уровни звукового давления

Полоса частот, Гц	1,5	3	25	50	100	200	500	1000	2000
Уровни звукового давления, дБ	6	1	1	4	9	5	2	0	8
Уровни	50								

звука, дБА	
Эквивалентные уровни звука, дБА	50

Работа в данном случае относится к творческой деятельности (проектирование и программирование) и максимальный уровень шума не должен превышать 50 дБА. Этим требованиям удовлетворяет рабочее место, оснащенное современным персональным компьютером с низким уровнем шума.

9.1.3 Освещение

Освещение на рабочем месте должно отвечать условиям работы и быть оптимальным по величине. Спектр освещения должен стремиться к спектру дневного света, т.к. физиология человека лучше адаптирована к последнему. В случае недостатка освещения человек быстро утомляется, снижается работоспособность, возрастает опасность ошибок и несчастных случаев. Слишком высокая освещенность, как и недостаточная, вызывает быстрое утомление глаз и снижение остроты зрения. Также освещенность должна быть равномерной по площади.

В итоге требования к производственному освещению сводятся к следующим:

- величина освещенности должна соответствовать гигиеническим нормам, учитывающим условия зрительной работы;
- состав спектра искусственного света, должен стремиться к спектру естественного освещения;
- должна быть обеспечена равномерность уровня освещенности в помещении, чтобы избежать переутомления зрения.

В помещении, в котором ведется разработка алгоритма, соблюдены все требования к освещению, приведенные в СанПиН 2.2.1/2.1.1.1278-03, и имеются искусственные и естественные источники освещения.

9.1.4 Психофизиологические факторы

При разработке и реализации алгоритмов разработчик сталкивается с напряженной зрительной работой и умственной нагрузкой, т.к. большая часть рабочего времени занята работой за персональным компьютером. В процессе работы с компьютером необходимо соблюдать правильный режим труда и отдыха. В противном случае у персонала отмечаются значительное напряжение зрительного аппарата с появлением жалоб на неудовлетворенность работой, головные боли, раздражительность, нарушение сна, усталость и болезненные ощущения в глазах.

Работа разработчика алгоритмов относится к III категории работы с ПЭВМ. Таким образом, регламентированное время перерывов составляет 70 минут. В время перерывов рекомендуется выполнять гимнастику для глаз, позволяющую снять зрительное напряжения. Кроме этого, эффективность перерывов повышается при организации специального помещения для отдыха персонала с удобной мягкой мебелью, аквариумом, зеленой зоной и т.п.

Также работа разработчика алгоритмов связана с монотонной работой за компьютером. При этом человек находится в фиксированной позе, которая не является физиологически естественной, что вредно для опорно-двигательного аппарата, а также приводит к застоям крови в сосудах и внутренних органах и другим заболеваниям. Наиболее известен в среде пользователей компьютеров тендовагинит запястных сухожилий, связанный с длительными однообразными движениями при вводе информации посредством мыши и клавиатуры. Для уменьшения негативного влияния монотонности труда рекомендуется во время перерывов заниматься какой-нибудь физической активностью: походить, размяться и т.д. Также это позволит отдохнуть от монотонных действий, таких как работа с мышью и клавиатурой.

Рабочее место разработчика алгоритма, при соблюдении правил работы за ПК, полностью соответствует требованиям для нормального психического и физиологического функционирования человека.

9.1.5 Электробезопасность

В процессе разработки алгоритма управления возможно поражение электрическим током, проходящим от сети к ПК, а также от ПК к микроконтроллеру. Электрические установки представляют для работника высокий уровень потенциальной опасности. В зависимости от условий в помещении опасность поражения человека электрическим током увеличивается или уменьшается. Не стоит работать с электрооборудованием в условиях повышенной влажности (длительное время более 75%), высокой температуры (более 35°C), наличии токопроводящей пыли, токопроводящих полов и возможности одновременного соприкосновения к имеющим соединение с землей металлическим элементам и металлическим корпусом электрооборудования. Исходя из этого, работа с ПК и микроконтроллером может проводиться только в помещениях без повышенной опасности, и возможность поражения током может быть только при прикосновении непосредственно с элементами ПК и выводам микроконтроллера.

Обязательны следующие меры предосторожности:

- перед началом работы нужно убедиться, что выключатели и розетка закреплены и не имеют оголенных токоведущих частей;
- при обнаружении неисправности оборудования и приборов необходимо не делая никаких самостоятельных исправлений сообщить ответственному за оборудование;
- запрещается загромождать рабочее место лишними предметами.

К мероприятиям по предотвращению возможного поражения электрическим током:

- соблюдение соответствующих расстояний до токоведущих частей или путем закрытия, ограждения токоведущих частей;
- применение блокировки аппаратов и ограждающих устройств для предотвращения ошибочных операций и доступа к токоведущим частям;
- применение предупреждающей сигнализации, надписей и плакатов;
- применение устройств для снижения напряженности электрических и магнитных полей до допустимых значений;

- использование средств защиты и приспособлений, в том числе для защиты от воздействия электрического и магнитного полей в электроустановках, в которых их напряженность превышает допустимые нормы. Повышение электробезопасности в установках достигается применением систем защитного заземления, защитного зануления и защитного отключения.

Все требуемые меры защиты соблюдаются в помещении, где ведется разработка алгоритма.

10 Экологическая безопасность

В настоящее время проблема загрязнения окружающей среды носит серьезный характер. Литосфера, гидросфера и атмосфера загрязнены различными продуктами хозяйственно деятельности человека. Современным трендом в науке и технике является внедрение технологий безотходного производства. Современные технологии позволяют уменьшить выбросы разнообразных веществ в окружающую среду. Несмотря на это, для уменьшения влияния на окружающую среду необходимо постоянно совершенствовать технологии добычи, потребления, переработки сырья, использования и утилизации оборудования.

10.1 Анализ воздействий объекта на литосферу и атмосферу

Для уменьшения негативного воздействия на окружающую среду необходимо заранее рассмотреть варианты утилизации неисправного и устаревшего оборудования, перегоревших люминесцентных ламп. Утилизация электрооборудования должна производиться в специализированных учреждениях. Кроме того, регулярная диагностика оборудования позволит предотвратить преждевременные поломки.

11 Безопасность в чрезвычайных ситуациях

Наиболее вероятной чрезвычайной ситуацией при разработке алгоритма является пожар на рабочем месте. Причиной пожара могут служить короткое замыкание, перегрузки по току, статическое электричество и т.д. В качестве противопожарных мероприятий должны быть применены следующие меры:

- должно использоваться только исправное оборудование;
- должны проводиться периодические инструктажи по пожарной безопасности;

- пути и проходы для эвакуации людей должны содержаться в свободном состоянии;

- в помещении должны находиться средства тушения пожара, средства связи; электрическая проводка электрооборудования и осветительных приборов должна быть исправна;

- все сотрудники должны знать место нахождения средств пожаротушения и уметь ими воспользоваться, средств связи и номера экстренных служб.

В связи с возможностью возникновения пожара разработан следующий план действий:

- в случае возникновения пожара сообщить о нем руководителю, постараться устранить очаг возгорания имеющимися силами при помощи первичных средств пожаротушения (огнетушитель порошковый, углекислотный О-1ПО (з)-АВСЕ);

- привести в действие ручной пожарный извещатель, если очаг возгорания потушить не удастся;

- сообщить о возгорании в службу пожарной охраны по телефону 01 или 010, сообщить адрес, место и причину возникновения пожара;

- принять меры по эвакуации людей и материальных ценностей;

- встретить пожарную охрану, при необходимости сообщить всю необходимую информацию и оказать помощь при выборе наилучшего подхода к очагу возгорания.

По взрывопожарной и пожарной опасности помещения подразделяются на категории А, Б, В1 - В4, Г и Д, а здания - на категории А, Б, В, Г и Д.

Согласно НПБ 105-03 помещение, где производится разработка алгоритма, относится к категории В – горючие и трудногорючие жидкости, твердые горючие и трудногорючие вещества и материалы (в том числе пыли и волокна), вещества и материалы, способные при взаимодействии с водой, кислородом воздуха или друг с другом только гореть, при условии, что помещения, в которых они имеются в наличии или обращаются, не относятся к категориям А или Б.

По степени огнестойкости данное здание относится к 1-й степени огнестойкости по СНиП 2.01.02-85 – здания с несущими и ограждающими конструкциями из естественных или искусственных каменных материалов, бетона

или железобетона с применением листовых и плитных негорючих материалов (выполнено из кирпича, которое относится к трудно сгораемым материалам)

В здании места работы над алгоритмом соблюдены требования пожаробезопасности и имеются необходимые средства пожаротушения.

12 Правовые и организационные вопросы обеспечения безопасности

Согласно Статьям 91 и 108 ТК РФ, регламентированное время продолжительности рабочего времени не должно превышать 40 часов в неделю, в течение рабочего дня работодатель обязан предоставить работнику перерыв для отдыха и питания. Предоставляемое работнику время для отдыха и питания располагается во временном промежутке от 30 минут до 2 часов, в зависимости от работодателя.

Кроме того, в соответствии с СанПиН 2.2.2.542-96, рабочее время, проведенное за компьютером не должно превышать 6 часов при 8-часовом рабочем дне, а непрерывная работа не должна превышать 2 часов. Регламентированные перерывы следует устанавливать через каждые 1.5-2 часа продолжительность 20 минут каждый или 15 минут через каждый час работы. Суммарное время перерывов должно составлять 70 минут.

Требования к организации рабочих мест предъявляются следующее. Рабочее место должно быть организовано с учетом эргономических требований согласно ГОСТ 12.2.032-78 «Система стандартов безопасности труда. Рабочее место при выполнении работ сидя. Общие эргономические требования» и ГОСТ 12.2.061-81 «ССБТ. Оборудование производственное. Общие требования безопасности к рабочим местам».

Обязанности по обеспечению безопасных условий и охраны труда возлагаются на работодателя и отражены в статье 212 ТК РФ.

13 Вывод по разделу «Социальная ответственность»

В ходе выполнения работы над разделом «Социальная ответственность» были выявлены опасные и вредные факторы, воздействию которых может подвергнуться работник, разрабатывающий и реализующий алгоритмы автоматического

управления для задач слежения за распознанными объектами. Среди этих факторов: шум, уровень освещённости, показатели микроклимата, психофизиологические факторы и электрический ток. По результатам работы, эти показатели не превышают допустимых значений. Также был проведен анализ нормативной документации.

Основываясь на результатах проделанной работы, был предложен ряд мер, для исключения или уменьшения влияния опасных и вредных факторов на человека и окружающую среду. Таким образом, рабочее место разработчика алгоритма соответствует нормативным требованиям по безопасности.

Заключение

В результате выполнения данной работы был разработан алгоритм управления для задач за распознанными объектами. Также данный алгоритм был реализован на базе микроконтроллера STM32F4.

Разработанное ПО обладает всем функционалом, который был задан в технических требованиях.

Список использованных источников

1. Кудрявцева Т. Методы дискретизации линейных систем непрерывного времени: [Электронный ресурс]. URL:https://a-lab.ee/edu/sites/default/files/Kudrjajtseva_BSc.pdf (Дата обращения: 13.05.2018)
2. Куо Б. Теория и проектирование цифровых систем управления. М.: Машиностроение, 1986.
3. Применение методов нейронных сетей и генетических алгоритмов в решении задач управления электроприводами. Электротехника, № 5, 1999. – С. 2–6.
4. Дорф В, Бишоп Р. Современные системы управления. М: Лаборатория Базовых Знаний, 2002.
5. Льюнг Л. Идентификация систем. Теория для пользователя. М: Главная Редакция Физико-Математической Литературы, 1991.
6. Иванов А.А. Основы робототехники: учебное пособие. Москва: Высшее образование, 2012. Изд. Высшее образование. 223 с.
7. Разработка алгоритма слежения за распознанными по видеопотоку объектами / А. А. Волков [и др.] // Молодежь и современные информационные технологии : сборник трудов XV Международной научно-практической конференции студентов, аспирантов и молодых учёных, 04-07 декабря 2017 г., г. Томск. — Томск : Изд-во ТПУ, 2017. — [С. 132-133].
8. СанПиН "Гигиенические требования к персональным электронно-вычислительным машинам и организации работы." от 21 июня 2003 г. № 2.2.2/2.4.1340-03 // Российская газета.
9. СНиП "Естественное и искусственное освещение." от 1 января 1996 г. № 23-05-95 // Российская газета.
10. СН "Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории застройки." от 31 октября 1996 г. № 2.2.4/2.1.8.562

– 96 // Российская газета.

11. ГОСТ "Здания жилые и общественные. Параметры микроклимата в помещениях."

от 1 января 2013 г. № 30494—2011 // Российская газета.

12. Инструкция "Инструкция по организации работ, охране труда и экологической

безопасности при работе на ПЭВМ /ПК/ в издательствах и на

полиграфических предприятиях Госкомпечати России" от 17 июня 2011

г. Российская газета.

13. ГОСТ Р "Безопасность оборудования информационной технологии, включая электрическое конторское оборудование." № 50377-92 (МЭК

950-86) // Российская газета.

14. ГОСТ "Система стандартов безопасности труда (ССБТ). Рабочее место

при выполнении работ сидя. Общие эргономические требования." №12.2.032-

78 // Российская газета.

Приложение А. Дерево вызова процедур

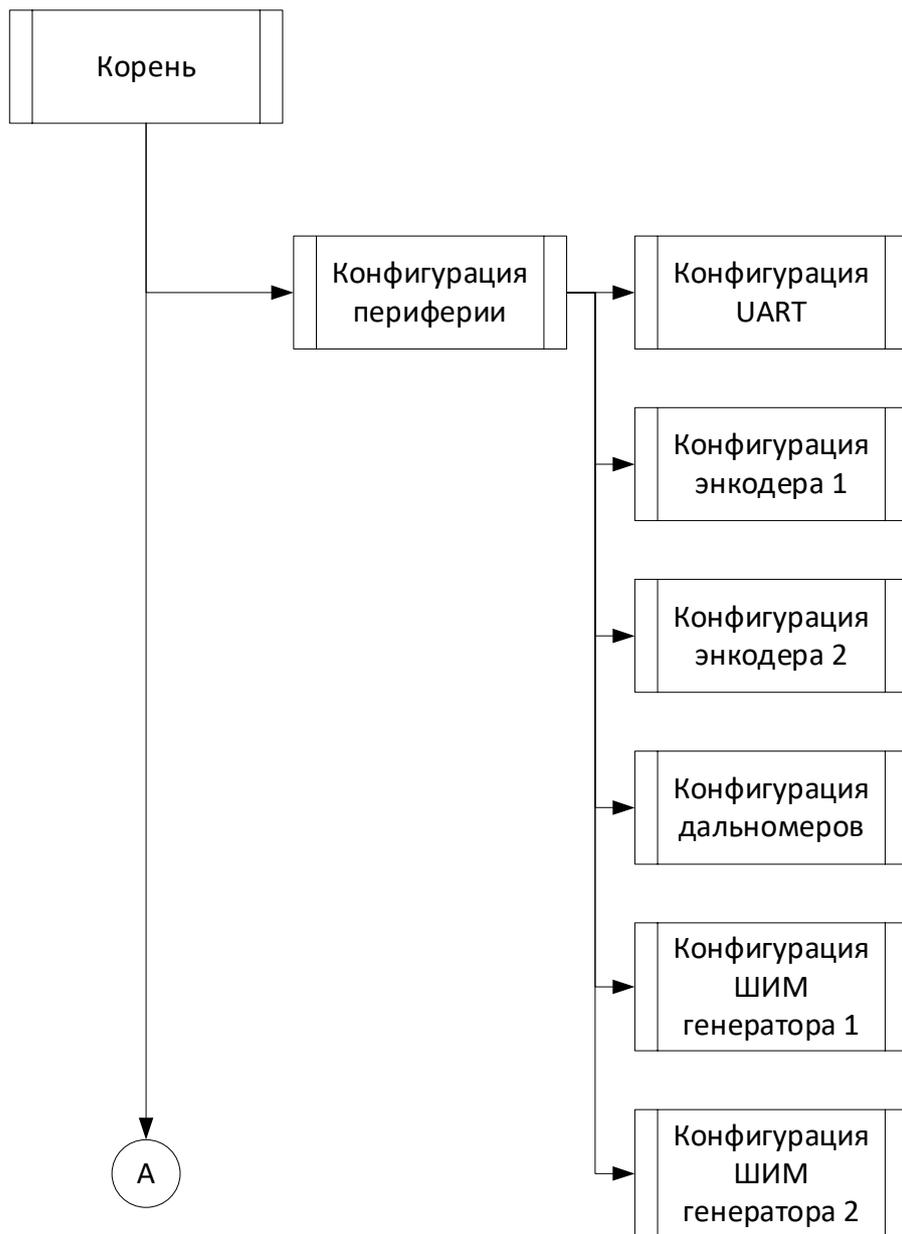


Рисунок – Дерево вызова процедур, часть 1



Рисунок 18– Дерево вызова процедур, часть 2

Приложение Б. Блок-схема главной программы

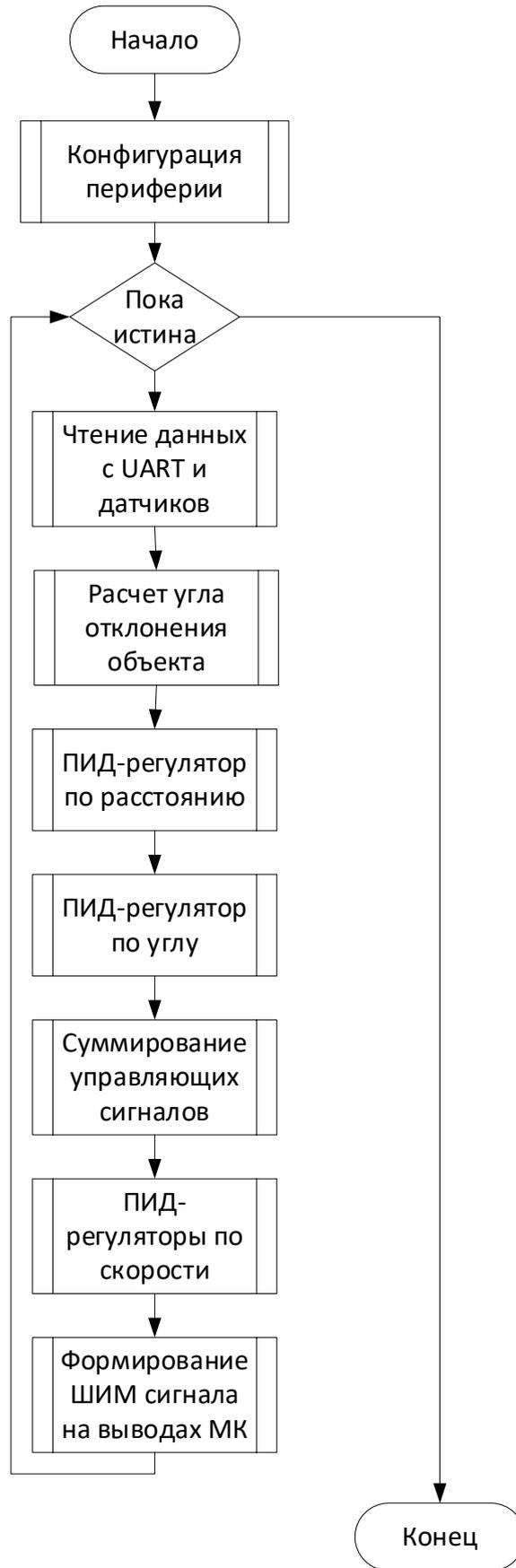


Рисунок 19 – Блок-схема главной программы

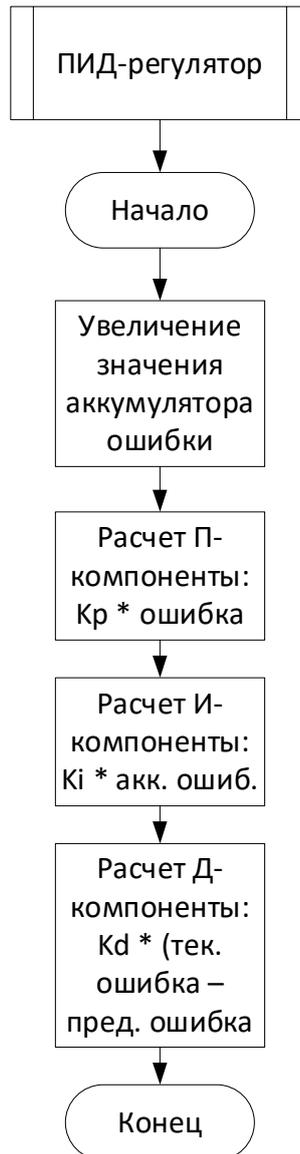


Рисунок 20 – Блок-схема функции ПИД-регулятора

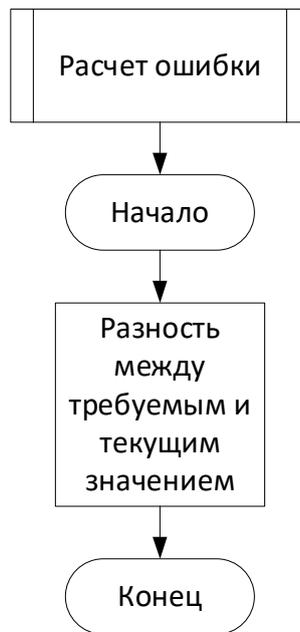


Рисунок 21 – Блок-схема функции расчета ошибки

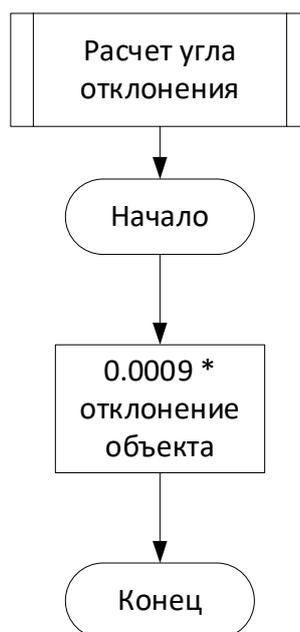


Рисунок 22 – Блок-схема функции расчета угла отклонения

Приложение В. Исходный код программы для получения экспериментальных данных (МК)

```
#include <TimerOne.h>
#include <Encod_er.h>

Encod_er encoder( 5, 6, 4);

void setup() {
  Serial.begin(250000); // инициализируем порт, скорость 250000
  Timer1.initialize(0.1); // инициализация таймера 1, период 0.1 мкс
  Timer1.attachInterrupt(timerInterrupt, 0.1); // задаем обработчик прерываний
}
double rotation = 0;
double pos = 0;
void loop() {
  if(encoder.timeRight != 0) {
    Serial.println(encoder.read());
    encoder.timeRight = 0;
    pos = 0;
  }
  if(encoder.timeLeft != 0) {
    Serial.println(encoder.read()); // вывод текущего положения
    encoder.timeLeft = 0;
  }
}

// обработчик прерывания 0.1 мкс
void timerInterrupt() {
  encoder.scanState();
}
```

Приложение Г. Исходный код программы для получения экспериментальных данных (ПК)

```
import serial
import time
import pandas as pd
import termcolor
flag = True
while flag:
try:
    ser = serial.Serial('COM5', 250000, timeout=0)
    flag = False
except:
print("Can't connect to target device... Try again")
time0 = time.time()
time1 = 0
tick0 = 0
ticks = 0
dr = 0.01
iteration = 0
iterations = 500
experiment_data = {'time': [], 'speed': []}
while iteration <= iterations:
    data = ser.readline()
try:
    ticks = int(data.decode("utf-8"))
    time1 = time.time() - time0
if time1 > dr:
with open('parameters.json') as parameters_file:
if (ticks - tick0) < 0:
    experiment_data['time'].append(iteration * dr)
    experiment_data['speed'].append(None)

continue

speed = round((ticks - tick0) * 60 / (3000 * time1), 2)
    experiment_data['time'].append(iteration * dr)
    experiment_data['speed'].append(speed)
print(f"Iteration: {iteration} \nSpeed = {speed} rpm \n--\n")
    iteration += 1
time0 = time.time()
    time1 = 0
tick0 = ticks
except:
pass
print(termcolor.colored('Experiment successfully finished!', color='green'))
out_data = pd.DataFrame(data=experiment_data)
out_data.to_csv('experiment_1.csv')
```

Приложение Д. Исходный код основной программы

```
#include "stm32f4xx.h"
#include "stm32f4xx_tim.h"
#include "stm32f4xx_gpio.h"
#include "stm32f4xx_rcc.h"
#include "stm32f4xx_usart.h"
#include "math.h"
#include "stdlib.h"

TIM_TimeBaseInitTypeDef TIM_TimeBaseStructure;
TIM_ICInitTypeDef TIM_ICInitStructure;
TIM_HandleTypeDef htim1;
TIM_HandleTypeDef htim8;
GPIO_InitTypeDef GPIO_InitStructure;
USART_InitTypeDef USART_InitStructure;

uint8_t receivedData[8];
uint8_t bytesToReceive = 16;
uint8_t receivedDataCounter = 0;
float distance = 0;
float angle = 0;
float speed_left = 0;
float speed_right = 0;
uint32_t counter = 0;
uint8_t direction;
uint32_t turn_counter = 0;
// параметры ПИД-регуляторов
struct PID_motor = {
    float err = 0;
    float sum_err = 0;
    float prev_err = 0;
    float kp = 5.9;
    float ki = 26.68;
    float kd = 0.352;
    float control = 0;
    float target = 0;
}

struct PID_angle = {
    float err = 0;
    float sum_err = 0;
    float prev_err = 0;
    float kp = 181;
    float ki = 1053;
    float kd = 2.833;
    float control = 0;
    float target = 0;
}
```

```

struct PID_dist = {
    float err = 0;
    float sum_err = 0;
    float prev_err = 0;
    float kp = 49.47;
    float ki = 98.39;
    float kd = -0.87;
    float control = 0;
    float target = 2;
}

void main(void) {

    // производим настройку периферии
    config();
    // основной цикл программы
    while(1) {
        readEncoder();
        readUART();
        readAngle();
        pid_angle.err = calculateErr(pid_angle.target, angle);
        pid_dist.err = calculateErr(pid_dist.target, distance);
        pid_angle.control = pidRegulator(pid_angle.err, pid_angle.sum_err,
pid_angle.prev_err, pid_angle.kp, pid_angle.ki, pid_angle.kd);
        pid_dist.control = pidRegulator(pid_dist.err, pid_dist.sum_err,
pid_dist.prev_err, pid_dist.kp, pid_dist.ki, pid_dist.kd);
        pid_left.target = pid_dist.control + pid_angle.control;
        pid_right.target = pid_dist.control - pid_angle.control;
        pid_left.err = calculateErr(pid_left.target, speed_left);
        pid_right.err = calculateErr(pid_right.target, speed_right);
        pid_left.control = pidRegulator(pid_left.err, pid_left.sum_err,
pid_left.prev_err, pid_left.kp, pid_left.ki, pid_left.kd);
        pid_right.control = pidRegulator(pid_right.err, pid_right.sum_err,
pid_right.prev_err, pid_right.kp, pid_right.ki, pid_right.kd);
        generatePWM(pid_left.control, pid_right.control);
    }
}

float calculateErr(float target, float current) {
    float err = 0;
    err = target - current;
    return err;
}

void config() {
    configUART();
    configEncoder_1();
    configEncoder_2();
    configDist();
    configPWM_left;
}

```

```

    configPWM_right;
    USART_ITConfig(USART1, USART_IT_TC, ENABLE);
    struct PID_motor pid_left;
    struct PID_motor pid_right;
    struct PID_angle pid_angle;
    struct PID_dist pid_dist;
}

/* инициализируем advanced-таймер для первого энкодера */
void configEncoder_1() {
    TIM_Encoder_InitTypeDef sConfig;
    TIM_MasterConfigTypeDef sMasterConfig;

    htim1.Instance = TIM1;
    htim1.Init.Prescaler = 0;
    htim1.Init.CounterMode = TIM_COUNTERMODE_UP;
    htim1.Init.Period = 65535;
    htim1.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
    htim1.Init.RepetitionCounter = 0;
    sConfig.EncoderMode = TIM_ENCODERMODE_TI12;
    sConfig.IC1Polarity = TIM_ICPOLARITY_RISING;
    sConfig.IC1Selection = TIM_ICSELECTION_DIRECTTI;
    sConfig.IC1Prescaler = TIM_ICPSC_DIV4;
    sConfig.IC1Filter = 15;
    sConfig.IC2Polarity = TIM_ICPOLARITY_FALLING;
    sConfig.IC2Selection = TIM_ICSELECTION_DIRECTTI;
    sConfig.IC2Prescaler = TIM_ICPSC_DIV4;
    sConfig.IC2Filter = 15;
    if (HAL_TIM_Encoder_Init(&htim1, &sConfig) != HAL_OK) {
        Error_Handler();
    }

    sMasterConfig.MasterOutputTrigger = TIM_TRGO_RESET;
    sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
    if (HAL_TIMEx_MasterConfigSynchronization(&htim1, &sMasterConfig) != HAL_OK)
    {
        Error_Handler();
    }

    HAL_TIM_Encoder_Start_IT(&htim1, TIM_CHANNEL_1 | TIM_CHANNEL_2);
    HAL_TIM_Base_Start(&htim1);
}

/* инициализируем advanced-таймер для второго энкодера */
void configEncoder_2() {
    TIM_Encoder_InitTypeDef sConfig;
    TIM_MasterConfigTypeDef sMasterConfig;

    htim8.Instance = TIM8;
    htim8.Init.Prescaler = 0;

```

```

htim8.Init.CounterMode = TIM_COUNTERMODE_UP;
htim8.Init.Period = 65535;
htim8.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
htim8.Init.RepetitionCounter = 0;
sConfig.EncoderMode = TIM_ENCODERMODE_TI12;
sConfig.IC1Polarity = TIM_ICPOLARITY_RISING;
sConfig.IC1Selection = TIM_ICSELECTION_DIRECTTI;
sConfig.IC1Prescaler = TIM_ICPSC_DIV4;
sConfig.IC1Filter = 15;
sConfig.IC2Polarity = TIM_ICPOLARITY_FALLING;
sConfig.IC2Selection = TIM_ICSELECTION_DIRECTTI;
sConfig.IC2Prescaler = TIM_ICPSC_DIV4;
sConfig.IC2Filter = 15;
if (HAL_TIM_Encoder_Init(&htim8, &sConfig) != HAL_OK) {
    Error_Handler();
}

sMasterConfig.MasterOutputTrigger = TIM_TRGO_RESET;
sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
if (HAL_TIMEx_MasterConfigSynchronization(&htim8, &sMasterConfig) != HAL_OK)
{
    Error_Handler();
}

HAL_TIM_Encoder_Start_IT(&htim8, TIM_CHANNEL_1 | TIM_CHANNEL_2);
HAL_TIM_Base_Start(&htim8);
}

void configDist() {
    RCC->APB2ENR = RCC_APB2ENR_IOPAEN|RCC_APB2ENR_AFIOEN; //Датчик подключен к
порту A, также включаем тактирование модуля AFIO для внешних прерываний
    GPIOA->CRL = 0x8A; //Нога 0 порта A настраивается на выход Push-Pull, нога 1
- на вход альтернативной функции с подтяжкой
    RCC->APB1ENR = RCC_APB1ENR_TIM2EN|RCC_APB1ENR_TIM3EN; //Включаем тактирование
таймеров 2 и 3

    TIM2->PSC = SystemCoreClock/1000000-1; //Предделитель таймера настраивается
так, чтобы счет шел каждую микросекунду
    TIM2->ARR = 60000-1; //Период сигнала, мкс
    TIM2->CCR1 = 10; //Длительность импульса, мкс
    TIM2->CR1 |= TIM_CR1_ARPE; //Включен режим предварительной записи регистра
автоперезагрузки
    TIM2->CCMR1 |= TIM_CCMR1_OC1PE; //Включен режим предварительной загрузки
регистра сравнения
    TIM2->CCMR1 |= TIM_CCMR1_OC1M_2|TIM_CCMR1_OC1M_1; //Режим PWM
    TIM2->CR1 |= TIM_CR1_CEN; //Включаем таймер
    TIM2->CCER |= TIM_CCER_CC1E; //Выход канала захвата/сравнения включен - нога
PA0

```

```

    TIM3->PSC = SystemCoreClock/1000000-1; //Предделитель настроен на счет каждую
микросекунду
    TIM3->ARR = 50000; //Считаем до 50 мс

    AFIO->EXTICR[0] = AFIO_EXTICR1_EXTI1_PA; //Настраиваем прерывания на ногу 1
порта A
    EXTI->IMR = EXTI_IMR_MR1; //Разрешаем прерывания с этой ноги
    EXTI->RTSR = EXTI_RTSR_TR1; //Прерывания по нарастающему сигналу
    EXTI->FTSR = EXTI_FTSR_TR1; //...и по спаду сигнала
    NVIC_EnableIRQ(EXTI1_IRQn);
}

/* производим конфигурацию UART */
void configUART() {
    // включение прерываний
    __enable_irq();
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_USART1, ENABLE);
    RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOA, ENABLE);

    // Инициализация нужных пинов контроллера, для USART1 -
    // PA9 и PA10
    GPIO_StructInit(&GPIO_InitStructure);

    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF;
    GPIO_InitStructure.GPIO_Pin = GPIO_Pin_9;
    GPIO_InitStructure.GPIO_Speed = GPIO_Speed_2MHz;
    GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
    GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_UP;
    GPIO_Init(GPIOA, &GPIO_InitStructure);

    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF;
    GPIO_InitStructure.GPIO_Pin = GPIO_Pin_10;
    GPIO_InitStructure.GPIO_Speed = GPIO_Speed_2MHz;
    GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
    GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_UP;
    GPIO_Init(GPIOA, &GPIO_InitStructure);

    // И вот еще функция, которой не было при работе с STM32F10x,
    // но которую нужно вызывать при использовании STM32F4xx
    GPIO_PinAFConfig(GPIOA, GPIO_PinSource9, GPIO_AF_USART1);
    GPIO_PinAFConfig(GPIOA, GPIO_PinSource10, GPIO_AF_USART1);

    // А теперь настраиваем модуль USART
    USART_StructInit(&USART_InitStructure);
    USART_InitStructure.USART_Mode = USART_Mode_Rx | USART_Mode_Tx;
    USART_InitStructure.USART_BaudRate = 250000;
    USART_Init(USART1, &USART_InitStructure);

    // Включаем прерывания и запускаем USART
    NVIC_EnableIRQ(USART1_IRQn);
}

```

```

    USART_Cmd(USART1, ENABLE);
}

void configPWM_left() {
    RCC->APB1ENR |= RCC_APB1ENR_TIM3EN;
    RCC->APB2ENR |= RCC_APB2ENR_IOPAEN;
    RCC->APB2ENR |= RCC_APB2ENR_AFIOEN;

    GPIOA->CRL &= ~GPIO_CRL_CNF1;
    GPIOA->CRL |= GPIO_CRL_CNF1_1;

    GPIOA->CRL &= ~GPIO_CRL_MODE1;
    GPIOA->CRL |= GPIO_CRL_MODE1;

    TIM3->PSC = 24 - 1;
    TIM3->ARR = 1000;
    TIM3->CCR2 = 0;

    TIM3->CCMR1 |= TIM_CCMR1_OC2M_1 | TIM_CCMR1_OC2M_2;
    TIM3->CCER |= TIM_CCER_CC2E;
    TIM3->CCER &= ~TIM_CCER_CC2P;

    TIM3->CR1 &= ~TIM_CR1_DIR;
    TIM3->CR1 |= TIM_CR1_CEN;
    RCC->APB2ENR |= RCC_APB2ENR_IOPCEN;
    RCC->APB2ENR |= RCC_APB2ENR_IOPAEN;
for port A // enable clock

    GPIOC->CRH &= ~GPIO_CRH_CNF8;
    GPIOC->CRH |= GPIO_CRH_MODE8_0;

    GPIOC->CRH &= ~GPIO_CRH_CNF9;
    GPIOC->CRH |= GPIO_CRH_MODE9_0;

    GPIOC->CRL &= ~GPIO_CRL_CNF0;
    GPIOC->CRL &= ~GPIO_CRL_MODE0;
}

void configPWM_right() {
    RCC->APB1ENR |= RCC_APB1ENR_TIM4EN;
    RCC->APB2ENR |= RCC_APB2ENR_IOPAEN;
    RCC->APB2ENR |= RCC_APB2ENR_AFIOEN;

    GPIOA->CRL &= ~GPIO_CRL_CNF1;
    GPIOA->CRL |= GPIO_CRL_CNF1_1;

    GPIOA->CRL &= ~GPIO_CRL_MODE1;
    GPIOA->CRL |= GPIO_CRL_MODE1;

    TIM4->PSC = 24 - 1;

```

```

TIM4->ARR = 1000;
TIM4->CCR2 = 0;

TIM4->CCMR1 |= TIM_CCMR1_OC2M_1 | TIM_CCMR1_OC2M_2;
TIM4->CCER |= TIM_CCER_CC2E;
TIM4->CCER &= ~TIM_CCER_CC2P;

TIM4->CR1 &= ~TIM_CR1_DIR;
TIM4->CR1 |= TIM_CR1_CEN;
RCC->APB2ENR |= RCC_APB2ENR_IOPCEN;
RCC->APB2ENR |= RCC_APB2ENR_IOPAEN; // enable clock
for port A

GPIOC->CRH &= ~GPIO_CRH_CNF10;
GPIOC->CRH |= GPIO_CRH_MODE10_0;

GPIOC->CRH &= ~GPIO_CRH_CNF11;
GPIOC->CRH |= GPIO_CRH_MODE11_0;

GPIOC->CRL &= ~GPIO_CRL_CNF0;
GPIOC->CRL &= ~GPIO_CRL_MODE0;
}

/* обработка данных от энкодеров */
int readEncoder() {
    counter_1 = __HAL_TIM_GET_COUNTER(&htim1);
    direction_1 = __HAL_TIM_IS_TIM_COUNTING_DOWN(&htim1);
    counter_2 = __HAL_TIM_GET_COUNTER(&htim8);
    direction_2 = __HAL_TIM_IS_TIM_COUNTING_DOWN(&htim8);
    speed_left = counter_1 * 60 / (3000 * 65535 / 72000000);
    speed_right = counter_2 * 60 / (3000 * 65535 / 72000000);
}

/* обработка данных от датчика расстояния */
float readDistance() {
    return distance;
}

/* преобразование данных угла */
float readAngle() {
    float angle = 0;
    int pixels = atoi(receivedData);
    angle = atan2f(0.0009 * pixels);
}

/* генерация ШИМ сигналов */
void generatePWM(left, right) {
    TIM3->CCR2 = left;
    TIM4->CCR2 = right
}

```

```

/* ПИД-регулятор */
float pidRegulator(float err, float sum_err, float prev_err, float kp, float ki,
float kd) {
    float control = 0;
    float p = kp * err;
    sum_err += err;
    float i = ki * sum_err;
    float d = kd * (err - prev_err);
    control = p + i + d
    return control;
}

/* прерывание для обработки сигнала с датчиков */
void EXTI1_IRQHandler(void)
{
    EXTI->PR |= EXTI_PR_PR1; //Сбрасываем флаг прерывания
    if(GPIOA->IDR & GPIO_IDR_IDR1) TIM3->CR1 |= TIM_CR1_CEN; //Проверяем, если на
ноге PA1 лог 1, запускаем таймер
    else{
//... а если 0, то:
        TIM3->CR1 &= ~TIM_CR1_CEN; //Останавливаем таймер
        distance = TIM3->CNT/58/100; //Записываем в глобальную переменную значение
регистра CNT/58 - это и будет расстояние в мм
        TIM3->CNT = 0; //Обнуляем таймер
    }
}

/* прерывание для приема данных по UART */
void USART1_IRQHandler()
{
    // Убеждаемся, что прерывание вызвано новыми данными в регистре данных
    if (USART_GetITStatus(USART1, USART_IT_RXNE) != RESET)
    {
        // Чистим флаг прерывания
        USART_ClearITPendingBit(USART1, USART_IT_RXNE);

        // То ради чего все затеяли - принимаем данные
        receivedData[receivedDataCounter] = USART_ReceiveData(USART1);

        // Увеличиваем значение счетчика
        receivedDataCounter++;

        // выходим из прерывания по приему данных
        if (receivedDataCounter == bytesToReceive)
        {
            USART_ITConfig(USART1, USART_IT_RXNE, DISABLE);
        }
    }
}
}

```