



Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Инженерная школа информационных технологий и робототехники
Направление подготовки 15.03.06 Мехатроника и робототехника
ООП/ОПОП Интеллектуальные робототехнические и мехатронные системы
Отделение школы (НОЦ) Отделение автоматизации и робототехники

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Тема работы
Модернизация учебно-исследовательского стенда «Робот-бабочка»

УДК 007.52:004.415.2

Обучающийся

Группа	ФИО	Подпись	Дата
8E92	Го Цзыцзюнь		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Леонов Сергей Владимирович	К.Т.Н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Былкова Татьяна Васильевна	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД	Мезенцева Ирина Леонидовна	-		

Нормоконтроль

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОАР ИШИТР	Поберезкина Екатерина Евгеньевна	-		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Киселев Александр Викторович	К.Т.Н.		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП

Код компетенции	Наименование компетенции
Универсальные компетенции	
УК(У)-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач
УК(У)-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений
УК(У)-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде
УК(У)-4	Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(-ых) языке(-ах)
УК(У)-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни
УК(У)-7	Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности
УК(У)-8	Способен создавать и поддерживать в повседневной жизни и в профессиональной деятельности безопасные условия жизнедеятельности для сохранения природной среды, обеспечения устойчивого развития общества, в том числе при угрозе и возникновении чрезвычайных ситуаций и военных конфликтов
УК(У)-9	Способен проявлять предприимчивость в практической деятельности, в т.ч. в рамках разработки коммерчески перспективного продукта на основе научно-технической идеи
УК(У)-10	Способен принимать обоснованные экономические решения в различных областях жизнедеятельности
УК(У)-11	Способен формировать нетерпимое отношение к коррупционному поведению
Общепрофессиональные компетенции	
ОПК(У)-1	Способен представлять адекватную современному уровню знаний научную картину мира на основе знания основных положений, законов и методов естественных наук и математики
ОПК(У)-2	Владеет физико-математическим аппаратом, необходимым для описания мехатронных и робототехнических систем
ОПК(У)- 3	Владеет современными информационными технологиями, готовностью применять современные средства автоматизированного проектирования и машинной графики при проектировании систем и их отдельных модулей, а также для подготовки конструкторско-технологической документации, соблюдать основные требования информационной безопасности

ОПК(У)-4	Готов собирать, обрабатывать, анализировать и систематизировать научно-техническую информацию по тематике исследования, использовать достижения отечественной и зарубежной науки, техники и технологии в своей профессиональной деятельности
ОПК(У)-5	Способен использовать основы экономических знаний при оценке эффективности результатов своей профессиональной деятельности
ОПК(У)-6	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности
Профессиональные компетенции	
ПК(У)-1	Способен составлять математические модели мехатронных и робототехнических систем, их подсистем и отдельных элементов и модулей, включая информационные, электромеханические, гидравлические, электрогидравлические, электронные устройства и средства вычислительной техники
ПК(У)-2	Способен разрабатывать программное обеспечение, необходимое для обработки информации и управления в мехатронных и робототехнических системах, а также для их проектирования
ПК(У)-3	Способен разрабатывать экспериментальные макеты управляющих, информационных и исполнительных модулей мехатронных и робототехнических систем и проводить их экспериментальное исследование с применением современных информационных технологий
ПК(У)-4	Способен осуществлять анализ научно-технической информации, обобщать отечественный и зарубежный опыт в области средств автоматизации и управления, проводить патентный поиск
ПК(У)-5	Способен проводить эксперименты на действующих макетах, образцах мехатронных и робототехнических систем по заданным методикам и обрабатывать результаты с применением современных информационных технологий и технических средств
ПК(У)-6	Способен проводить вычислительные эксперименты с использованием стандартных программных пакетов с целью исследования математических моделей мехатронных и робототехнических систем
ПК(У)-7	Готов участвовать в составлении аналитических обзоров и научно-технических отчетов по результатам выполненной работы, в подготовке публикаций по результатам исследований и разработок
ПК(У)-8	Способен внедрять результаты исследований и разработок и организовывать защиту прав на объекты интеллектуальной собственности

ПК(У)-9	Способен участвовать в качестве исполнителя в научно-исследовательских разработках новых робототехнических и мехатронных систем
ПК(У)-10	Готов участвовать в подготовке технико-экономического обоснования проектов создания мехатронных и робототехнических систем, их подсистем и отдельных модулей
ПК(У)-11	Способен производить расчёты и проектирование отдельных устройств и подсистем мехатронных и робототехнических систем с использованием стандартных исполнительных и управляющих устройств, средств автоматики, измерительной и вычислительной техники в соответствии с техническим заданием
ПК(У)-12	Способен разрабатывать конструкторскую и проектную документацию механических, электрических и электронных узлов мехатронных и робототехнических систем в соответствии с имеющимися стандартами и техническими условиями
ПК(У)-13	Готов участвовать в проведении предварительных испытаний составных частей опытного образца мехатронной или робототехнической системы по заданным программам и методикам и вести соответствующие журналы испытаний
Профессиональные компетенции университета	
ДПК (У)-1	Способен проводить проверку технического состояния оборудования, обоснование экономической эффективности внедрения проектируемых модулей и подсистем мехатронных и робототехнических устройств, настройку системы управления и обработки информации с использованием соответствующих инструментальных средств



Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

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

УТВЕРЖДАЮ:
Руководитель ООП
_____ Киселев А.В.
(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

Обучающийся:

Группа	ФИО
8Е92	Го Цзыцзюнь

Тема работы:

Модернизация учебно-исследовательского стенда «Робот-бабочка»	
Утверждена приказом директора (дата, номер)	№34-87/с от 03.02.2023

Срок сдачи обучающимся выполненной работы:	04.06.2023
--	------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе</p> <p><i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Объектом исследования является учебно-исследовательского стенда «Робот-бабочка».</p> <p>Разработка системы управления учебно-исследовательского стенда «Робот-бабочка» должна производиться с учетом имеющейся базы оборудования в существующем стенде:</p> <ul style="list-style-type: none"> – промышленный камер Basler acA1300-200uc; – двигатель постоянного тока Maxon DC RE 50; – блок питания MEAN WELL NES-350-27; – энкодер Scanco SCA50; – сервоконтроллер Maxon ESCON 70/10.
---	--

<p>Перечень разделов пояснительной записки подлежащих исследованию, проектированию и разработке</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. Литературный обзор стенда «Робот-бабочка». 2. Разработка структурной и функциональной схем для стенда. 3. Выбор основных компонентов. 4. Разработка алгоритма обнаружения шарика. 5. Разработка алгоритмов передачи данных между Python и STM32. 6. Разработка алгоритмы для ПИД-регуляторов по скорости и по положению.
--	--

<p>Перечень графического материала</p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<ol style="list-style-type: none"> 1. Структурная схема. 2. Блок-схема алгоритма. 3. Конфигурация параметров STM32CubeMX. 4. Конфигурация параметров Escon Studio.
--	--

Консультанты по разделам выпускной квалификационной работы

(с указанием разделов)

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Былкова Татьяна Васильевна, доцент ОСГН ШБИП, к.э.н.,
Социальная ответственность	Мезенцева Ирина Леонидовна, старший преподаватель ООД, ШБИП

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	03.02.2023
---	------------

Задание выдал руководитель:

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

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8E92	Го Цзыцзюнь		

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Инженерная школа информационных технологий и робототехники
 Направление подготовки 15.03.06 «Мехатроника и робототехника»
 Уровень образования Бакалавриат
 Отделение школы (НОЦ) Отделение автоматизации и робототехники
 Период выполнения Весенний семестр 2022 /2023 учебного года

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Обучающийся:

Группа	ФИО
8E92	Го Цзыцзюнь

Тема работы:

Модернизация учебно-исследовательского стенда «Робот-бабочка»

Срок сдачи обучающимся выполненной работы:	04.06.2023
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
27.05.2023 г.	<i>Основная часть ВКР</i>	60
30.05.2023 г.	<i>Раздел «Социальная ответственность»</i>	20
30.05.2023 г.	<i>Раздел «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»</i>	20

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Леонов Сергей Владимирович	к.т.н.		03.02.2023

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Киселев Александр Викторович	к.т.н.		03.02.2023

Обучающийся

Группа	ФИО	Подпись	Дата
8E92	Го Цзыцзюнь		03.02.2023

Реферат

Выпускная квалификационная работа состоит из 101 страниц, 37 рисунков, 23 таблиц, 27 источников и 6 приложений.

Ключевые слова: учебно-исследовательский стенд «Робот-бабочка», микроконтроллер *STM32*, последовательная передача данных, ПИД-регулятор, компьютерное зрение, управление двигателем, модернизация.

Объектом исследования является учебно-исследовательский стенд «Робот-бабочка».

Цель работы – разработка системы управления стендом «Робот-бабочка» на основе системы *Windows*.

В процессе исследования был проведен литературный обзор. Также была разработана структурная схема, функциональная схема управления к стенду «Робот-бабочка». После этого разработали алгоритм обнаружения шарика, алгоритм передачи данных с компьютера на микроконтроллер *STM32* через *Python* и разработали ПИД-регуляторы по скорости и по положению для управления вращения двигателя.

В результате исследования были разработаны алгоритмы для управления учебно-исследовательского стенда «Робот-бабочка», которые позволяют управлять стендом «Робот-бабочка» на персональном компьютере с системой *Windows*, что снижает сложность использования стенда.

Область применения: Образовательные исследования, Разработка высокоточных алгоритмов.

Экономическая эффективность/значимость работы заключается в снижении стоимости использования учебно-исследовательского стенда «Робот-бабочка».

Выпускная квалификационная работа подготовлена в текстовом редакторе Microsoft Word 2019, графический материал – в пакете Microsoft Visio 2019.

Для выполнения выпускной квалификационной работы использовались следующие основные программы: Pycharm Community 2022.2.1, Escon Studio, STM32CubeMx, Keil uVision5.

Содержание

Введение.....	13
Определения, обозначения, сокращения	15
1 Литературный обзор	17
1.1 Теоретическая часть.....	17
1.2 Обзор аналогов	18
2 Описание технологического процесса.....	19
2.1 Общая информация о стенде «Робот-бабочка»	19
2.2 Автоматизированные задачи работы	20
2.3 Используемые технические средства.....	20
2.3.1 Промышленная камера	20
2.3.2 Энкодер	21
2.3.3 Микроконтроллер.....	24
2.3.4 Двигатель постоянного тока	25
2.3.5 Сервоконтроллер.....	26
3 Модернизация аппаратной и программной части учебно-исследовательского стенда «Робот-бабочка».....	28
3.1 Краткое описание	28
3.1.1 Модернизация аппаратной части стенда	28
3.1.2 Модернизация программной части стенда.....	28
4 Реализация и описания алгоритмов и методов	30
4.1 Распознавание шарика.....	30
4.1.1 Анализ существующих алгоритмов	30
4.1.2 Реализация алгоритма в проекте	33
4.2 Формирование потока данных для STM32.....	35
4.2.1 Реализация алгоритма на стороне Python.....	35

4.2.2	Реализация алгоритма на стороне STM32	37
4.3	Обратная связь – Отображение состояния вращения двигателя.....	38
4.4	Блок управления двигателем – Использование ПИД-регулятора.....	39
4.4.1	ПИД-регулятор	39
4.4.2	Реализация технологии ПИД-регулятора	41
5	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	47
5.1	Потенциальные потребители результатов исследования	47
5.2	Анализ конкурентных технических решений	48
5.3	SWOT-анализ.....	49
5.4	Планирование научно-исследовательских работ.....	53
5.4.1	Структура работ в рамках научного исследования	53
5.4.2	Определение трудоемкости выполнения работ и диаграмма Ганта... ..	54
5.5	Бюджет научно-технического исследования (НТИ)	57
5.5.1	Расчет материальных затрат НТИ	57
5.5.2	Расчет амортизационных отчислений на реализацию проекта.....	58
5.5.3	Основная заработная плата исполнителей темы	59
5.5.4	Дополнительная заработная плата исполнителей темы.....	60
5.5.5	Отчисления во внебюджетные фонды	61
5.5.6	Накладные расходы	62
5.5.7	Формирование бюджета затрат научно-исследовательского проекта	62
5.5.8	Определение ресурсной, финансовой и экономической эффективности исследования	63
5.6	Выводы по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение».....	66
6	Социальная ответственность.....	69
6.1	Введение.....	69
6.2	Правовые и организационные вопросы обеспечения безопасности	69

6.2.1 Особенности законодательного регулирования проектных решений	69
6.2.2 Организационные мероприятия при компоновке рабочей зоны	70
6.3 Производственная безопасность	71
6.3.1 Превышение уровня шума	72
6.3.2 Отсутствие или недостаток необходимого естественного освещения	72
6.3.3 Отклонение показателей микроклимата в рабочей зоне	73
6.3.4 Производственные факторы, связанные с напряженностью электрического поля	74
6.3.5 Производственные факторы, связанные с электрическим током	75
6.3.6 Производственные факторы, связанные со статическим электричеством	75
6.4 Экологическая безопасность	75
6.5 Безопасность в чрезвычайных ситуациях	76
6.6 Вывод по разделу «Социальная ответственность»	77
Заключение	78
Список использованных источников	79
Приложение А (обязательное) Структурная схема система	82
Приложение Б (рекомендуемое) Программный код на стороне Python	84
Приложение В (рекомендуемое) Код основного файла App.c на стороне STM32	87
Приложение Г (рекомендуемое) Код основного файла App1.c на стороне STM32	92
Приложение Д (обязательное) Конфигурация параметров STM32CubeMX	97
Приложение Е (обязательное) Конфигурация параметров Escon Studio	100

Введение

Современная робототехника становится все более и более продвинутой, и робототехника проникает в различные сферы повседневной жизни. Роботы могут заменить людей для работы в сложных условиях и работать с высокой точностью.

«Робот-бабочка» представляет собой учебно-исследовательский стенд, который в основном состоит из двигателя постоянного тока, сервоконтроллера, источника питания, промышленной камеры, микроконтроллера и энкодера. Объектив промышленной камеры направлен на панели в форме бабочки, что и послужило причиной ее названия «робот-бабочка». При вращении панели положение шара также будет меняться. Шарик каждый раз находится на разном расстоянии от центра и в каждый момент имеет разное ускорение. Промышленная камера фиксирует информацию о положении шарика, и информация будет передана в микроконтроллер, который использует встроенный алгоритм для расчета соответствующей скорости вращения для управления панелью, чтобы шарик оставался стабильным при вращении панелью.

«Робот-бабочка» – это комплекс программно-аппаратных. Комплекс поможет автоматизировать многие производственные операции и создать роботизированные ячейки для сверления, фрезерования, полировки, сборочных работ. С помощью программно-аппаратного комплекса студенты осваивают компетенции для разработки приложений в таких областях, как телемедицина, реабилитация, авиационная и ракетно-строительная техника и других [1].

Однако исходная система управления может использовать только компьютер с системой Linux для управления стенда «Робот-бабочка». По сравнению со знакомой нам системой Windows, система Linux более сложна, что, несомненно, увеличивает сложность обучения использованию «Робота-бабочки». При этом также отсутствуют алгоритмы управления стендом «Робот-бабочка» через компьютер с системой Windows.

В связи с чем целью работы является разработка системы управления стендом «Робот-бабочка» на основе системы *Windows*. Для реализации данной цели необходимо выполнить следующие задачи:

- исследовать существующие алгоритмы определения положения шарика;
- разработать алгоритм отправки данных со стороны Python на микроконтроллер STM32;
- исследовать методы управления скоростью двигателя и изучить ПИД-регулятор;
- реализовать системное управление стендом «Робот-бабочка».

Определения, обозначения, сокращения

В данной работе применены следующие термины с соответствующими определениями:

энкодер: Измерительный преобразователь угла поворота вращающегося объекта (например, вала) в цифровые или аналоговые сигналы, которые позволяют определить угол его поворота.

микроконтроллер: Микросхема, предназначенная для управления электронными устройствами.

преобразование Серкл Хафа (*Circle Hough Transform*): Метод извлечения признаков, используемый в цифровой обработке изображений для обнаружения кругов на несовершенных изображениях.

сервоконтроллер: Микропроцессорное устройство, способное управлять движением в реальном масштабе времени. Результатом работы сервоконтроллера является выдача управляющих сигналов на преобразователь частоты в какой-либо форме, которые меняются пропорционально требуемой скорости / перемещению / крутящему моменту.

ПИД-регулятор: Устройство в управляющем контуре с обратной связью. Он предназначен для поддержания установленных уровней задаваемых величин, например, температуры воздуха.

универсальный асинхронный или универсальный синхронно/асинхронный приемопередатчик: Удобный и простой последовательный интерфейс для организации информационного канала обмена микроконтроллера с внешним миром.

двигатель постоянного тока: Электрическая машина постоянного тока, преобразующая электрическую энергию постоянного тока в механическую энергию.

промышленная камера: Тип камеры. По сравнению с традиционными гражданскими камерами (камерами), обладают высокой стабильностью изображения, высокой пропускной способностью и высокой помехоустойчивостью и т. д.

В данной работе применены следующие сокращения и советуемые им расшифровки:

ПК – персональный компьютер;

СУ – система управления;

ПИД – пропорционально-интегрально-дифференцирующий;

USART – универсальный синхронно/асинхронный приемопередатчик;

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта.

1 Литературный обзор

1.1 Теоретическая часть

Учебно-исследовательский стенд «Робот-бабочка» относится к роботам, появившимся в последние годы, поэтому статей об исследовании стенда «Робот-бабочка» немного. Большинство статей представляют собой теоретический анализ движения мяча по панели и теоретический метод управления роботом-бабочкой.

Антон Ширяев и Леонид Фредович в своей работе "Кейс по нехватательной манипуляции: планирование и орбитальная стабилизация однонаправленных качения для стенда «Робот-бабочка»" Детально проанализировано движение стенда «Робот-бабочка». В статье указано, что «Робот-бабочка» состоит из двух одинаковых пластин в форме восьмерки, жестко расположенных параллельно друг другу на небольшом расстоянии, предназначен для манипулирования шариком, который может свободно катиться по границам пластин, как по рельсам. Шарик ненадежно прикреплен к раме, и движение шарика обеспечивается за счет силы тяжести. На основании известной информации авторы предлагают в исследовании следующие гипотезы:

- движение шарика, катящегося по раме, можно рассматривать как движение шарика по цилиндру;
- в любой момент существует только одна точка контакта между шаром и панелью, и панель не деформируется;
- шарик не будет скользить при движении по панели [2].

«Робот-бабочка» может помочь техническим специалистам разработать высокоточные алгоритмы для применения в аэрокосмической, медицинской и других областях. Можно сказать, что популярность стенда «Робот-бабочка» довольно важна.

Этот проект разрабатывает алгоритм, который может управлять стендом «Робот-бабочка» на ПК с системой Windows.

1.2 Обзор аналогов

Учитывая, что целью исследовательского проекта было обновление и модернизация исходной системы управления, можно предположить, что первоначальный робот «Бабочка» является наиболее подходящим аналогом. Ниже приведена сравнительная таблица характеристики двух приборов (разрабатываемое устройство и первоначальный робот).

Таблица 1 – Обзор аналогов разрабатываемого устройства

Показатели	Система управления	Микропроцессор	Язык для определения положения мяча	Язык микроконтроллера для чтения и вычисления информации об изображении
Разрабатываемое устройство	Windows	STM32	Python	C/C++
Первоначальный робот «Бабочка»	Linux	BeagleBone Black	Язык на базе Linux	Язык на базе Linux

Как видно из таблицы 1, по сравнению с первоначальным роботом «Бабочка», разрабатываемое устройство имеет больше преимуществ. Например, Система управления на базе Windows, позволяет управлять роботом «Бабочка» со своим компьютером. Это снижает барьеры для работы и повышает удобство использования. Языки Python и C/C++ менее сложны, чем языки на базе Linux.

2 Описание технологического процесса

2.1 Общая информация о стенде «Робот-бабочка»

«Робот-бабочка» — это платформа для разработки и обучения алгоритмов. Пользователи могут использовать учебно-исследовательский стенд «Робот-бабочка» для изучения и разработки адаптивных и нечетких алгоритмов управления. Мы можем использовать роботов-бабочек для разработки высокоточных алгоритмов, таких как разработка высокоточных алгоритмов управления медицинскими роботами для выполнения хирургических операций.

Стенд «Робот-бабочка» состоит из следующего оборудования и конструкций: панель в форме «Бабочка», двигатель постоянного тока maxim, промышленная камера Basler, энкодер sch50f, микроконтроллер BeagleBone Black, сервоконтроллер Escon 70/10 и система управления системой Linux. Стоит отметить, что в исходной системе управления для управления стенда «Робот-бабочка» должен использоваться компьютер с системой Linux.

Принцип работы робота-бабочки понять нетрудно: промышленная камера получает информацию о положении шарика и передает информацию на микроконтроллер, а микроконтроллер отправляет сигнал напряжения на двигатель постоянного тока для управления скоростью двигателя, так что шарик движется по панели оставаться стабильным [3-6].

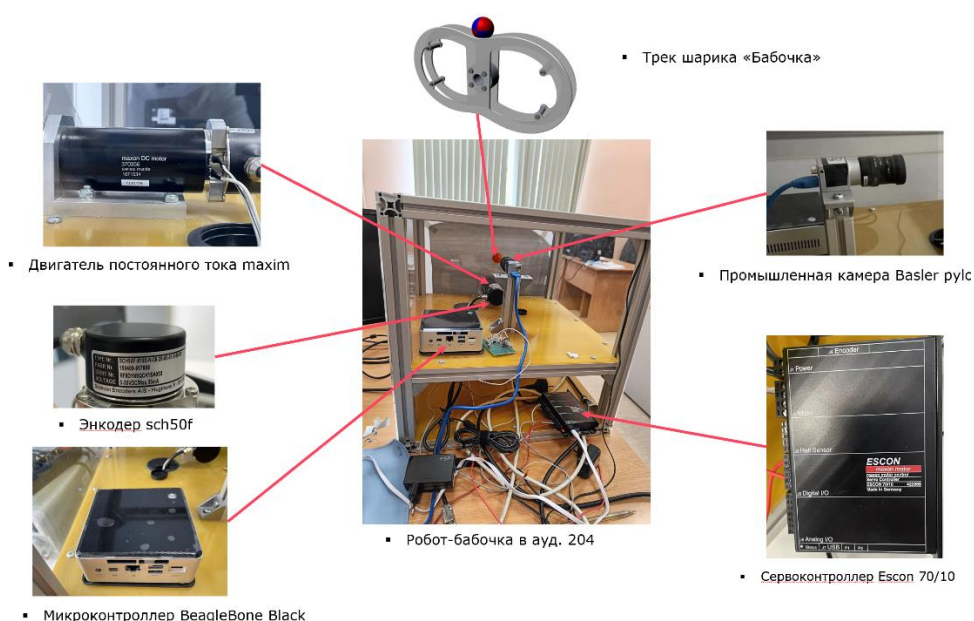


Рисунок 1 – Состав учебно-исследовательского стенда «Робот-бабочка»

2.2 Автоматизированные задачи работы

Как упоминалось ранее, стенд «Робот-бабочка» может работать только в системе Linux, что, несомненно, усложняет использование «Робот-бабочка». По сравнению с системой Windows, система Linux более сложна с точки зрения порога использования и сложности работы, а уровень ее использования также ниже по сравнению с системой Windows.

Это определило автоматизированные задачи: обновление оригинального учебно-исследовательской платформы «Робот-бабочка». Система управления была модернизирована с системы Linux до системы Windows, что позволяет пользователям управлять стендом «Робот-бабочка» через персональный ноутбук. Кроме того, микроконтроллер заменен с оригинального BeagleBlack на микроконтроллер STM32. Студенты и техники лучше знакомы с контроллером STM32, чем с исходным микроконтроллером, а микроконтроллер STM32 менее сложен в использовании. STM32 также обладает такими характеристиками, как высокая производительность, низкая стоимость, низкое энергопотребление и многофункциональность.

2.3 Используемые технические средства

2.3.1 Промышленная камера

В задаче требуется получение информации о положении мяча в режиме реального времени, но обычные камеры не могут снимать быстро движущиеся объекты. Для получения качественных и стабильных изображений промышленная камера на стенде, несомненно, лучший выбор. Промышленная камера *Basler acA1300-200uc* была использована в проекте. Промышленные камеры Basler отличаются высокой стабильностью изображения, широкой полосой пропускания и высокой помехоустойчивостью по сравнению с другими камерами.

Технические характеристики промышленной камеры *Basler acA1300-200uc* представлены в таблице 2:

Таблица 2 – Технические характеристики промышленной камеры

Сенсор промышленного камера		Характеристики камеры	
Сенсор	<i>PYTHON 1300</i>	Интерфейс	<i>USB 3.0</i>
Размер пикселя (Г x В)	4.8 μm x 4.8 μm	Цифровой вход	1
Частота кадров	203 fps	Цифровой выход	1
Тип сенсора	<i>CMOS</i>	Общий ввод/вывод	2
Монохромная / цветная	<i>Color</i>	Источник питания	<i>Via USB 3.0 interface</i>
Разрешение	1.3 MP	Требования к мощности (типичные)	3 W



Рисунок 2 – Промышленная камера *Basler ace1300-200uc*

2.3.2 Энкодер

Чтобы получить информацию о положении и скорости вращения двигателя, в проекте был использован энкодер *Scancon sch50f* [10-11]. Этот энкодер представляет собой высокоточный промышленный энкодер.

Технические характеристики энкодера *Scancon sch50f* представлены в таблице 3 [12]:

Таблица 3 – Технические характеристики энкодера *Scancon sch50f*

Вид	Значение/Диапазон	Вид	Значение/Диапазон
Диаметр	50 мм	Код	инкрементный
Разрешение	от 1 до 12 500 импульсов на оборот (импульсов на оборот)	Напряжение питания	4,5 В постоянного тока мин. до 30 В постоянного тока макс. (макс. 35 мА - без нагрузки)

Продолжение таблицы 3 – Технические характеристики энкодера *Scancon sch50f*

Вид	Значение/Диапазон	Вид	Значение/Диапазон
Выходное напряжение	Низкий: 500 мВ макс. при 10 мА Высокий: ($V_{in} - 0,6$) при -10 мА ($V_{вх} - 1,3$) при -25 мА	Выходной ток	30 мА макс. нагрузка на выходной канал
Выходной формат	Двухканальный (А, В) квадратурный с индексным (Z) и опциональным дополнительным (А-, В-, Z-) выходами	Чувство фазы	А ведет В по часовой стрелке (CW) от монтажного конца энкодера.
Масса	Энкодер: ~ 120 г (4,23 унции) Кабель: 60 гр/метр (2,12 унции/метр)	Скорость вала	12 000 об/мин (макс.)

Другая информация показана на рисунке ниже:

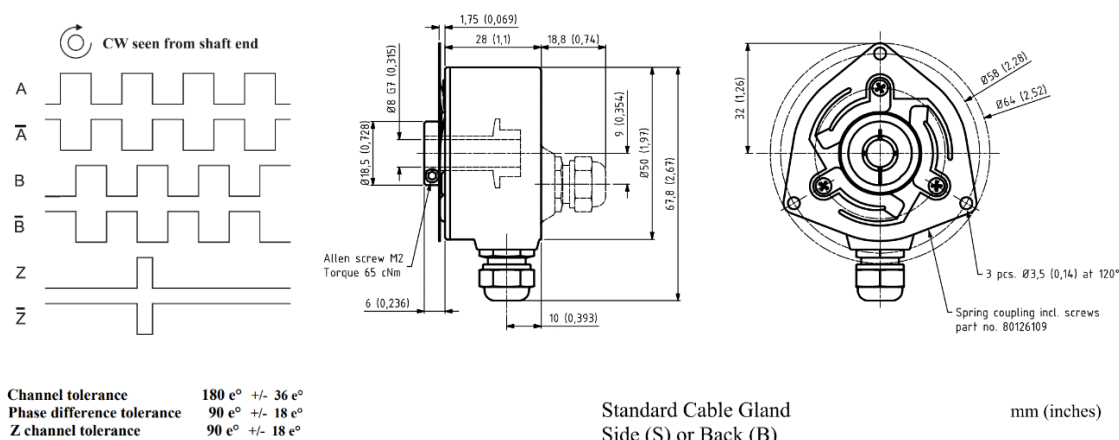


Рисунок 3 – Выходной сигнал и Механические размеры



Рисунок 4 – Энкодер *Scancon sch50f*

Сигналы, посылаемые прямым и обратным вращением двигателя, показаны на рисунке ниже (Красный — фаза А, желтый — фаза В):

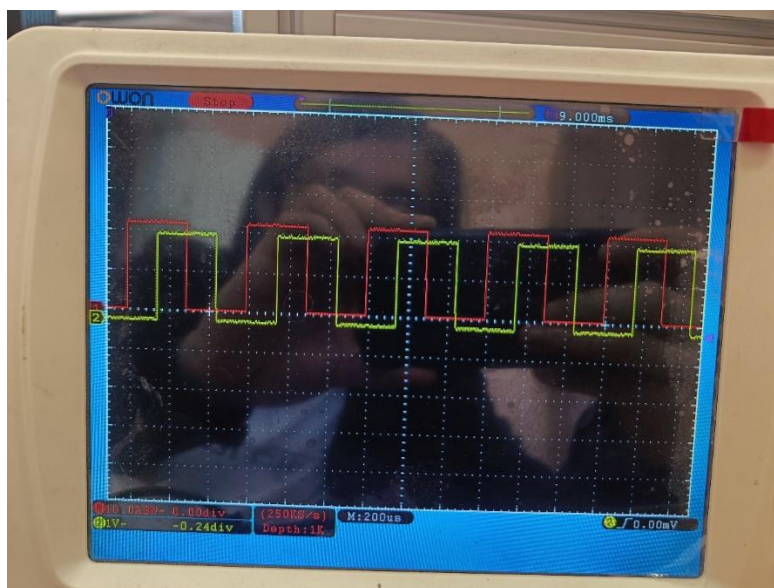


Рисунок 5 – Выходные сигналы (Вращение против часовой стрелки)

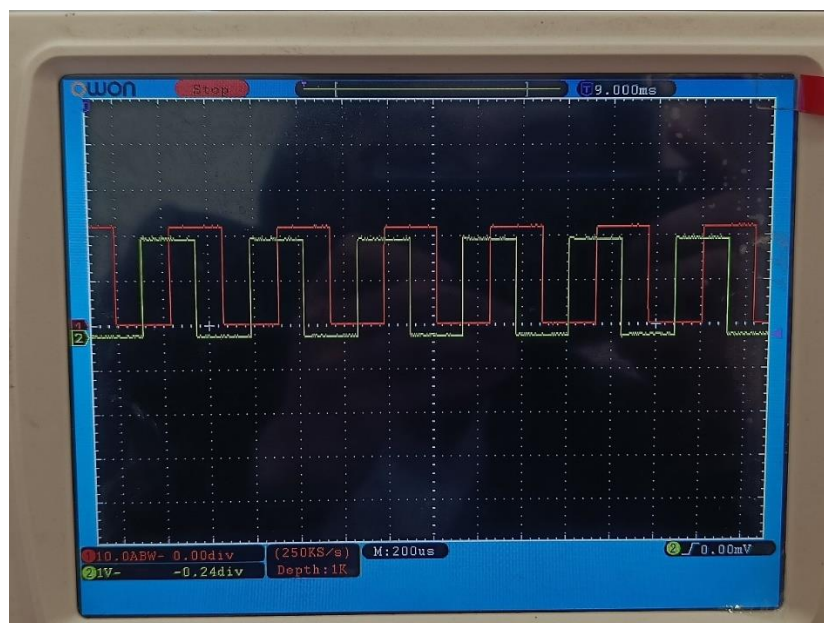


Рисунок 6 – Выходные сигналы (вращение по часовой стрелке)

2.3.3 Микроконтроллер

Ключевой частью исследовательской проекта является микроконтроллер, который должен получать информацию о положении мяча от компьютера и управлять вращением двигателя на основе этой информации. Учитывая применимость, удобство и эффективность микроконтроллера, в проекте был выбран микроконтроллер серии *STM32F3-Discovery* [13].

STM32F3-DISCOVERY позволяет пользователям легко разрабатывать приложения с помощью микроконтроллера смешанных сигналов серии STM32F3 на основе Arm® Cortex®-M4. Он включает в себя все необходимое для быстрого начала работы как новичков, так и опытных пользователей.

Основанный на микроконтроллере *STM32F303VCT6*, он включает в себя встроенный инструмент отладки ST-LINK/V2 или ST-LINK/V2-B, акселерометр, гироскоп и электронный компас ST MEMS, USB-соединение, светодиоды и кнопки.

Плата *STM32F3-DISCOVERY* предлагает следующие функции:

- STM32F303VCT6 Микроконтроллер на базе Arm®(a) с 256 Кбайт флэш-памяти, 48 Кбайт ОЗУ в корпусе LQFP100;
- Встроенный ST-LINK/V2 для печатной платы версии А или В или ST-LINK/V2-B для печатной платы версии С и новее;
- Питание платы: по шине USB или от внешнего источника питания 3 В или 5 В;
- Питание внешнего приложения: 3 В и 5 В;
- Датчик движения ST MEMS, 3-осевой гироскоп с цифровым выходом;
- Комплексная система ST MEMS с трехмерным цифровым датчиком линейного ускорения и трехмерным цифровым магнитным датчиком;
- Десять светодиодов:
 - LD1 (красный) для включения питания 3,3 В;
 - LD2 (красный/зеленый) для связи по USB;
 - Восемь пользовательских светодиодов: LD3/10 (красный), LD4/9 (синий), LD5/8 (оранжевый) и LD6/7 (зеленый).

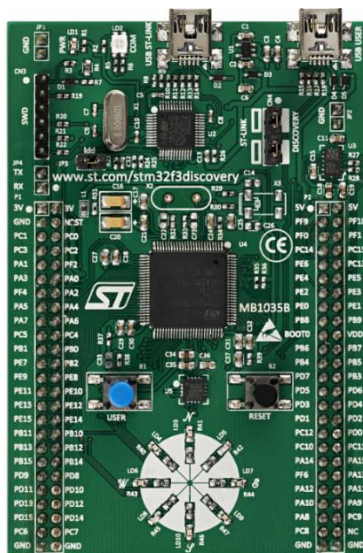


Рисунок 7 – Макетная плата *STM32F3-Discovery*

2.3.4 Двигатель постоянного тока

Двигатель является важной частью системы управления, он заставляет панель вращаться по часовой стрелке или против часовой стрелки. В этом проекте использовался двигатель постоянного тока *Maxon RE 50 Ø50 mm*. Этот двигатель постоянного тока имеет характеристики небольшого размера и мощного привода.

Технические характеристики и параметры промышленного камера представлены ниже, Рисунок 8 – 10.

Значения при номинальном напряжении

Стадия жизненного цикла	Не для новых разработок
Номинальное напряжение	48 В
Скорость холостого хода	4900 об/мин
Ток холостого хода	88.4 мА
Номинальная скорость	4630 об/мин
Номинальный момент (макс. длительный момент)	407 мНм
Номинальный ток (макс. длительный ток)	4.44 А
Пусковой момент	7370 мНм
Пусковой ток	78.9 А
Максимальный КПД	94 %

Рисунок 8 – Значения при номинальном напряжении двигателя

Параметры

Сопротивление между выводами	0.608 Ом
Индуктивность между выводами	0.423 мГн
Моментная постоянная	93.4 мНм/А
Скоростная постоянная	102 об/мин/В
Крутизна механической характеристики	0.666 об/мин/мНм
Механическая постоянная времени	3.78 мс
Инерция ротора	542 г·см ²

Рисунок 9 – Параметры двигателя

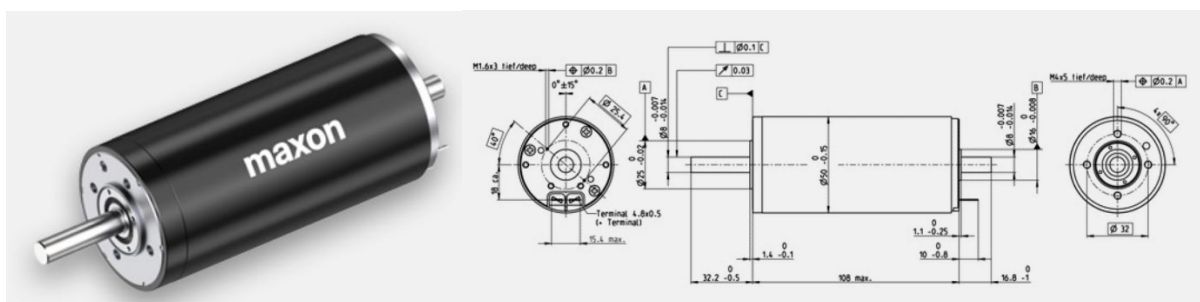


Рисунок 10 – Двигатель Maxon Maxon RE 50 Ø50 mm и его структура

2.3.5 Сервоконтроллер

Сервоконтроллер представляет собой микропроцессорное устройство, способное управлять движением в реальном масштабе времени. Результатом работы сервоконтроллера является выдача управляющих сигналов на преобразователь частоты в какой-либо форме. В проекте использовался сервоконтроллер maxon типа ESCON 70/10. Сервоконтроллер Escon 70/10 представляет собой высокопроизводительный сервоконтроллер небольшого размера, который может эффективно управлять двигателями постоянного тока мощностью около 700 Вт.

Технические характеристики и параметры промышленного камера представлены ниже, Рисунок 11.

Электрические характеристики

Рабочее напряжение V_{CC} , мин.	10 В
Рабочее напряжение V_{CC} , макс.	70 В
Макс. выходное напряжение (коэффициент к V_{CC})	0.95
Макс. импульсный выходной ток	30 А
Макс. время импульсного выходного тока	2 с
Макс. ток в продолжительном режиме, для режима постоянного тока	10 А
Частота ШИМ силового каскада	53.6 кГц
Частота дискретизации ПИ регулятора тока	53.6 кГц
Частота дискретизации ПИ регулятора скорости	5.36 кГц
Максимальный КПД	98 %
Макс. скорость (коллекторный)	150000 об/мин
Макс. скорость (бесколлекторный, 1 пара полюсов) блочная коммутация	150000 об/мин
Встроенный дроссель двигателя, на фазу	15 мкГн

Рисунок 11 – Электрические характеристики



Рисунок 12 – Сервоконтроллер Escon 70/10

3 Модернизация аппаратной и программной части учебно-исследовательского стенда «Робот-бабочка»

3.1 Краткое описание

Модернизация учебно-исследовательского стенда «Робот-бабочка» нашла отражение в обновлении программного и аппаратного обеспечения [9].

3.1.1 Модернизация аппаратной части стенда

В этом проекте вместо оригинального микроконтроллера *BeagleBone Black* для управления стендом «Робот-бабочка» использовалась отладочная плата микроконтроллера STM32. Программное обеспечение STM32CubeMX — это программное обеспечение, разработанное ST для настройки макетной платы STM32. Мы можем легко настроить макетную плату STM32 с помощью этого программного обеспечения.

3.1.2 Модернизация программной части стенда

Изначально для управления стендом «Робот-бабочка» требовался компьютер на базе Linux. Этот проект переводит систему управления из системы Linux в систему Windows, что позволяет персональным компьютерам управлять стендом «Робот-бабочка». Внедрение системы Windows повысило практичность различных устройств и снизило порог обучения на стенде.

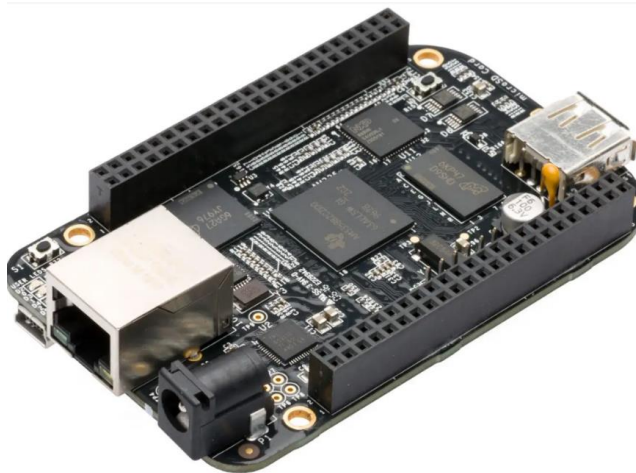


Рисунок 13 – Встроенная платформа *BeagleBone Black*

Следующие два рисунка могут более интуитивно увидеть разницу между до и после обновления:

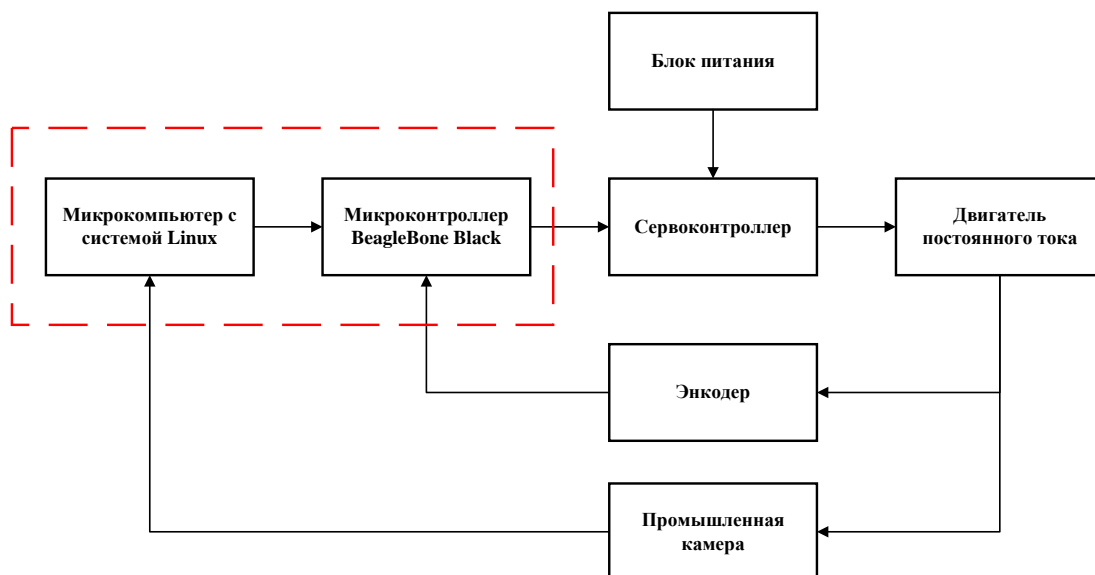


Рисунок 14 – Технический состав до обновления

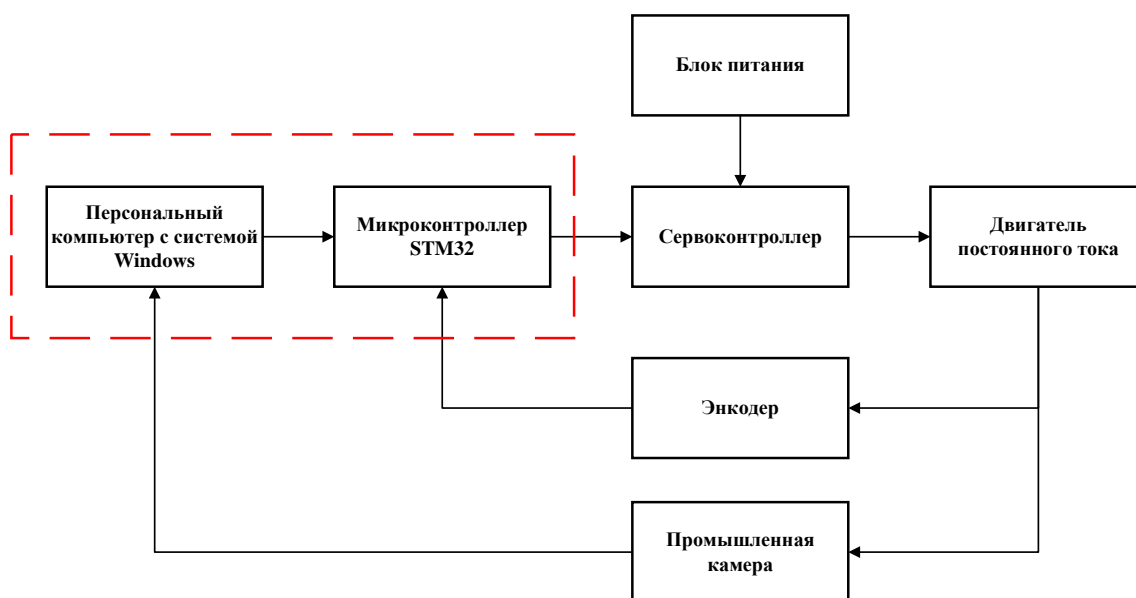


Рисунок 15 – Технический состав после обновления

4 Реализация и описания алгоритмов и методов

Разработку системы управления учебно-исследовательским стендом «Робот-бабочка» можно разделить на следующие четыре основные части:

- Распознавание шарика;
- Формирование потока данных для STM32;
- Обратная связь;
- Блок управления двигателем.

Отношения между четырьмя частями показаны на рисунке 16 ниже:



Рисунок 16 – Отношения между четырьмя частями

4.1 Распознавание шарика

В качестве первой части этого исследовательского проекта был разработан алгоритм определения положения шарика. Алгоритм распознавания мяча пропущен через библиотеку *Opencv* в среде *Python*.

4.1.1 Анализ существующих алгоритмов

Существует два основных метода, используемых для распознавания шарика в проекте: *Преобразование Хафа по кругу (СНТ)* и *Сегментация цветного изображения*. Оба метода могут реализовать распознавание шариков на изображении [7].

Преобразование Хафа по кругу – это базовый метод извлечения признаков, используемый в цифровой обработке изображений для обнаружения кругов на несовершенных изображениях. Кандидаты на круг создаются путем “голосования” в пространстве параметров Хафа, а затем выбора локальных

максимумов в матрице аккумуляторов. Преобразование Хафа по кругу — это процесс преобразования окружности в двумерном пространстве изображения в точку в трехмерном пространстве параметров, определяемую радиусом окружности и горизонтальными и вертикальными координатами центра окружности. Окружность, определяемая любыми тремя точками на окружности, преобразуется по Хафу. Последнему должна соответствовать точка в трехмерном пространстве параметров [8].

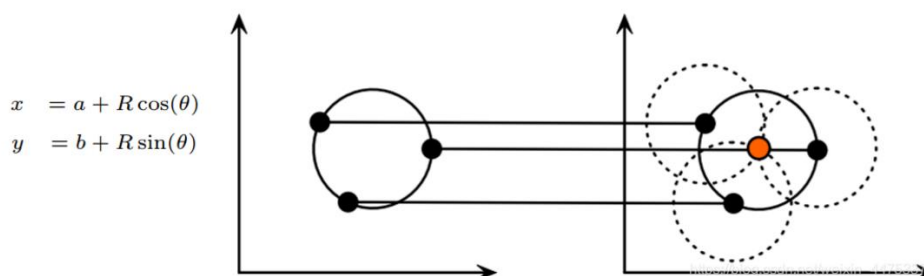


Рисунок 17 – Преобразование Хафа по кругу

```

1.  CVAPI(CvSeq*) cvHoughCircles( CvArr* image, void* circle_storage,
2.                                int method, double dp, double min_dist,
3.                                double param1 CV_DEFAULT(100),
4.                                double param2 CV_DEFAULT(100),
5.                                int min_radius CV_DEFAULT(0),
6.                                int max_radius CV_DEFAULT(0));

```

Рисунок 18 – Функция HoughCircles в библиотеке OpenCV в среде Python

Сегментация цветного изображения — используется в OpenCV для распознавания конкретных объектов / областей, имеющих определенный цвет. Для цветовой сегментации все, что нам нужно, это пороговые значения или знание нижней границы и верхней границы диапазона цветов в одном из цветовых пространств.

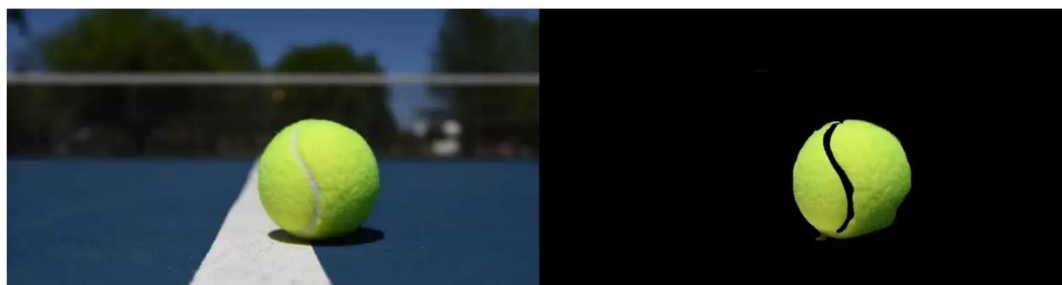


Рисунок 19 – Сегментированный теннисный мяч

В проекте использовались два метода для написания кода и запуска программы в программном обеспечении Pycharm и получил следующие результаты:

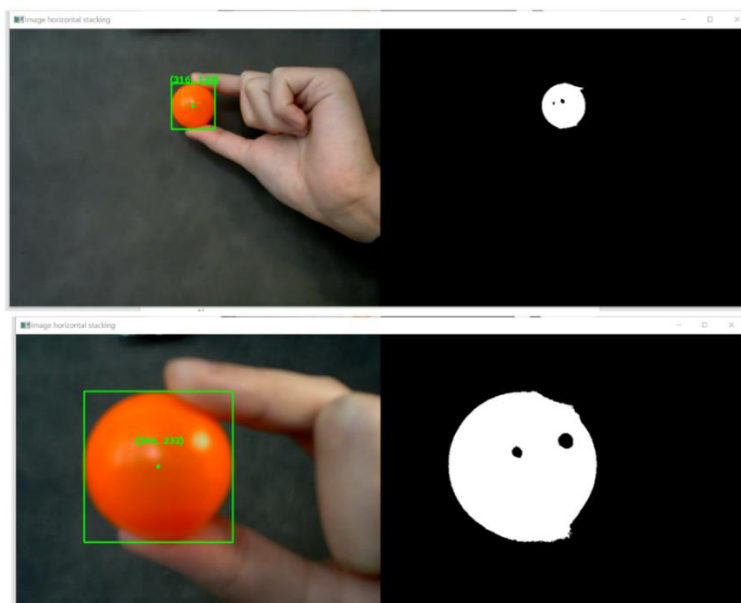


Рисунок 20 – Результат методом Сегментация цветного изображения
(Процесс приближения мяча к камере)

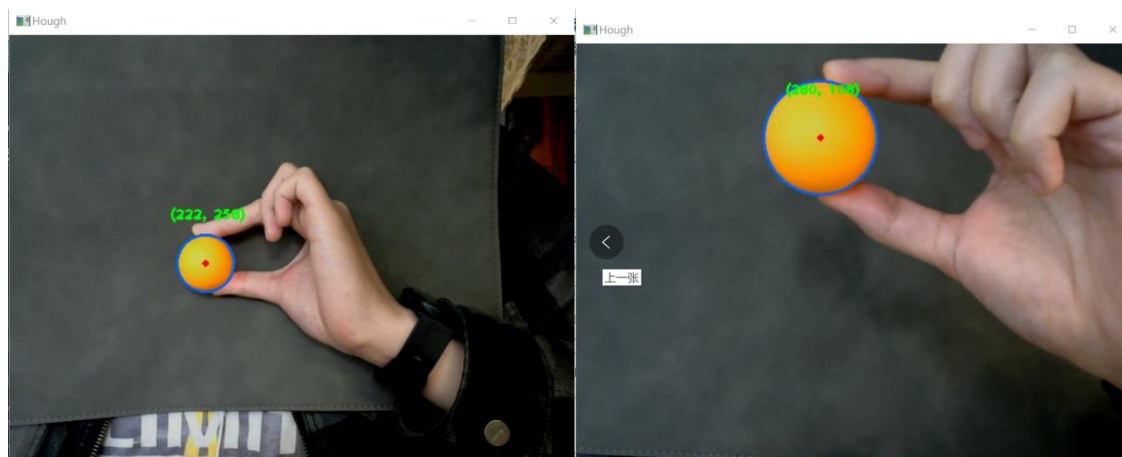


Рисунок 21 – Результат распознавания методом Преобразования Хафа по кругу
(Процесс приближения мяча к камере)

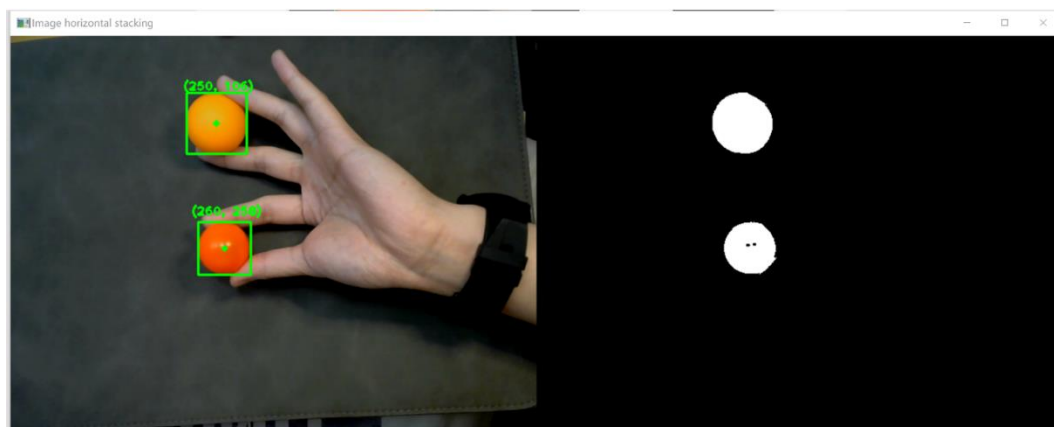


Рисунок 22 – Результат распознавания методом Сегментация цветного изображения (Распознавание 2 шарика одновременно)

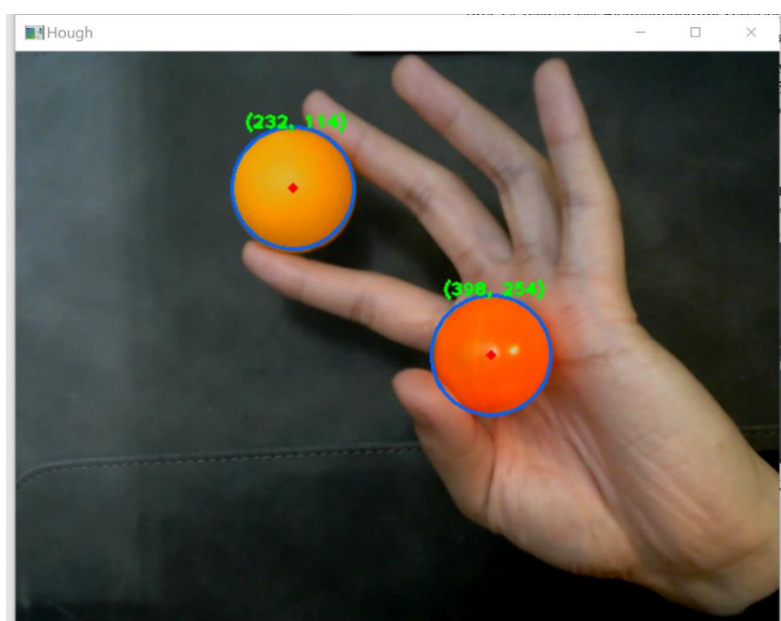


Рисунок 23 – Результат распознавания методом Преобразования Хафа по кругу (Распознавание 2 шарика одновременно)

Вывод: Оба метода могут распознавать шарик на изображении. Однако, учитывая, что в последующем нам необходимо получить координаты шарика на фоне разного цвета, чтобы исключить интерференцию цвета фона, выбрал метод **Преобразования Хафа по кругу** в качестве метода распознавания проекта.

4.1.2 Реализация алгоритма в проекте

На блок-схеме показана реализация алгоритма для распознавания шарика:

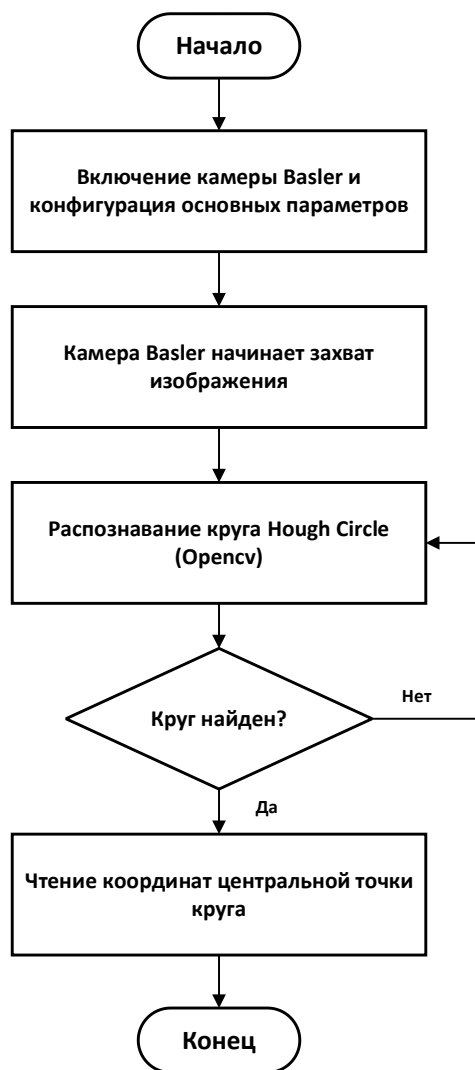


Рисунок 24 – Блок схема программы для распознавания шарика

Идентифицированная информация о координатах не является той информацией, которую необходимо отправить. Информация об угле и расстоянии, которую необходимо отправить, рассчитывается на основе положения шарика.

В соответствии с размером изображения и центральным положением панели выберите точку А (640,1080) в качестве начала координат. Горизонтальная правая сторона является положительным направлением по оси X, а вертикальная нисходящая сторона положительна по оси Y. Вычислите угол между положением шара и осью Y и расстояние между шаром и началом координат по формулам (3.1) и (3.2):

Угол между шаром и осью Y:

$$\varphi = \arctan \frac{x-660}{|1080-y|} \quad (3.1)$$

Расстояние между двумя точками:

$$R = \sqrt{(x - 660)^2 + (y - 1080)^2} \quad (3.2)$$

Результат показан на рисунке ниже:

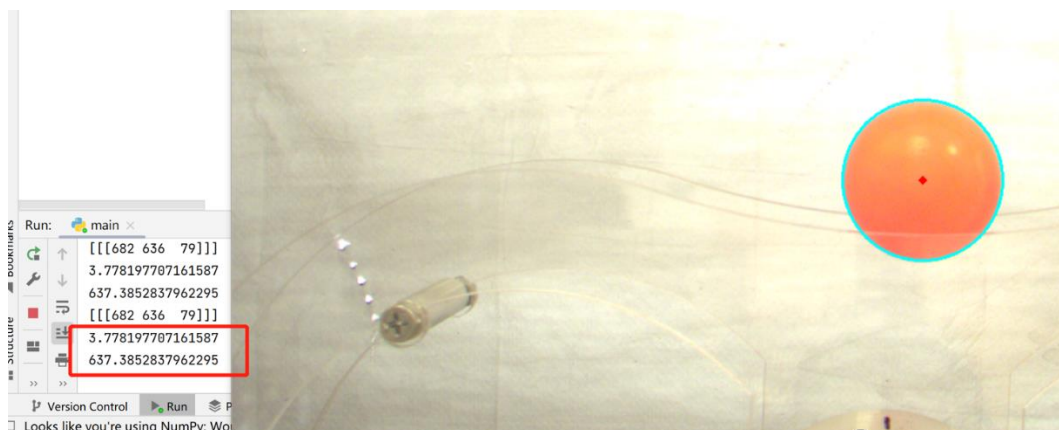


Рисунок 25 – Полученный результат

Из рисунка 25 видим, что программа может определить шарик на изображении. Координаты центральной точки шарика печатаются в окне Run.

4.2 Формирование потока данных для STM32

Чтобы отправить информацию о положении шарика на микроконтроллер STM32, необходимо найти метод для передачи данных со стороны Python на микроконтроллер STM32 [14].

4.2.1 Реализация алгоритма на стороне Python

В *Pycharm* используем библиотеку *pyserial* для передачи данных. Чтобы успешно отправить данные на STM32, необходимо установить ту же скорость передачи данных (115200).

Кроме того, также необходимо обработать отправленные данные перед отправкой. *Pyserial* отправляет данные в виде строк, так как каждое число в Python занимает один байт, одновременная отправка двух данных с плавающей запятой будет занимать много памяти, поэтому STM32 не может принимать такие данные [15].

Библиотека *Struct* — это модуль для упаковки и распаковки двоичных данных. Он позволяет вам связывать двоичные данные с различными типами данных, такими как целые числа, числа с плавающей запятой и т. д., а также сохранять и считывать их. Функция *struct.pack()* упаковывает два данных в двоичные данные, а после отправки их в STM32 распаковывает данные на стороне STM32, чтобы получить два данных типа с плавающей запятой. При упаковке данных заголовки и концы кадра вставляются для идентификации и проверки полученных данных на стороне STM32.

Блок-схема программы представлена на рисунке ниже:



Рисунок 26 – Блок-схема (передача данных: *Python* -> *STM32*)

Результат, полученный после запуска программы, показан на рисунке 26

ниже:

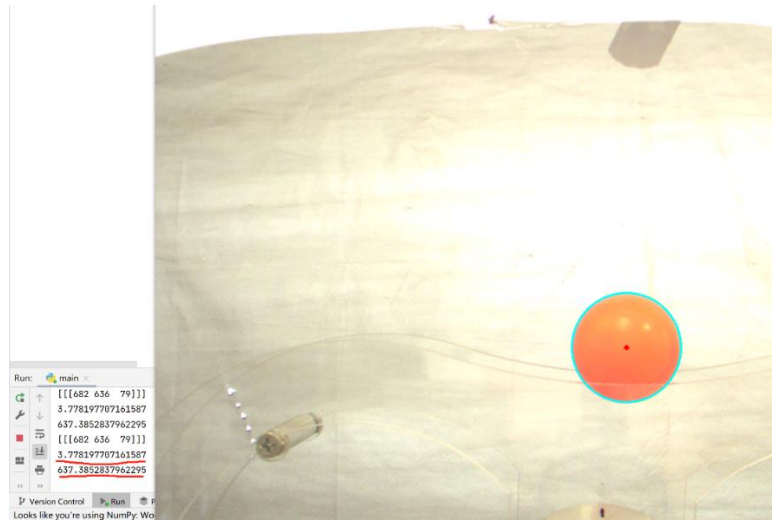


Рисунок 27 – Полученный результат в Pycharm

4.2.2 Реализация алгоритма на стороне STM32

На стороне STM32 нужно только разделить данные на две части (два данных) в соответствии с положением заголовка кадра и хвоста кадра, а затем сохранить их в две переменные соответственно. Блок-схема показана ниже:

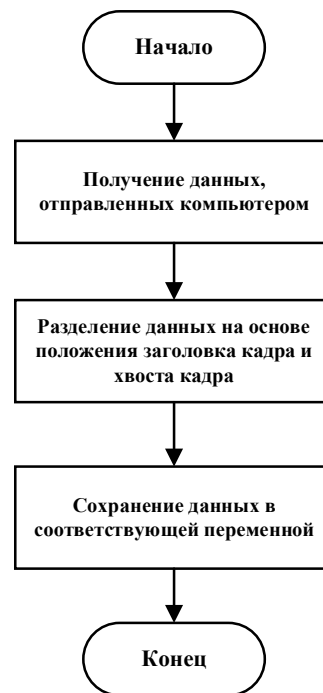


Рисунок 28 – Блок-схема на стороне STM32 (Получение данных)

Программы на стороне Python и на стороне STM32 запускаются одновременно, и можно получить следующие результаты:

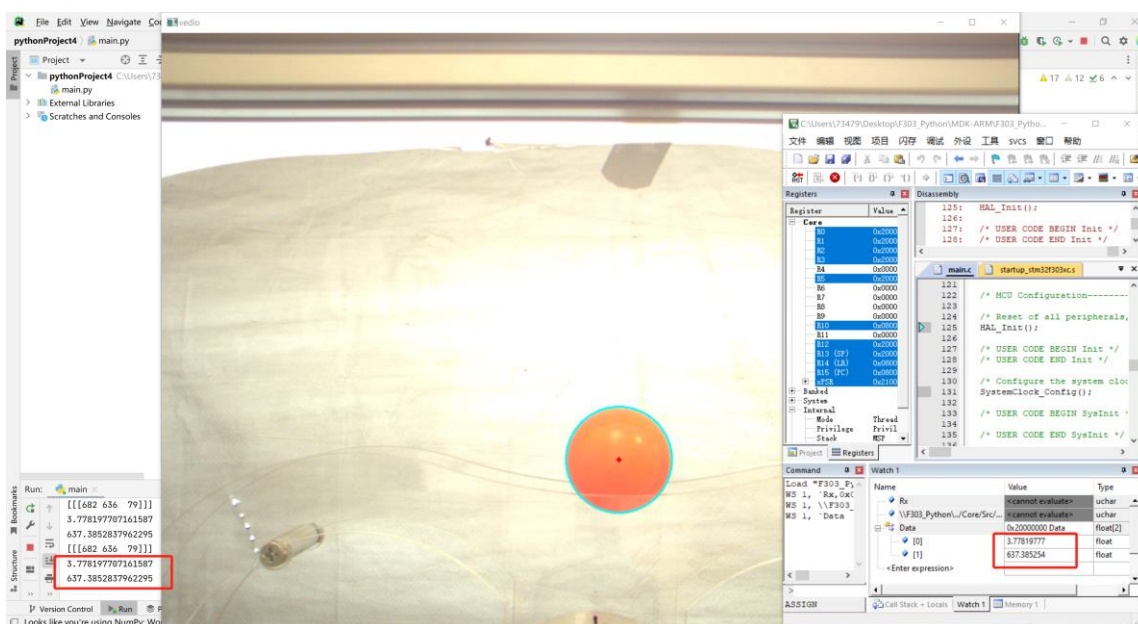


Рисунок 29 – Полученный результат передачи данных

Из результатов видим, что STM32 может получать данные, отправленные со стороны Python, что указывает на то, что поток данных между Python и STM32 успешно установлен. Положение шарика меняется, поэтому отправляемые данные тоже меняются. Данные, полученные терминалом STM32, изменяются по мере изменения положения шарика.

4.3 Обратная связь – Отображение состояния вращения двигателя

На плате *STM32F3-Discovery* имеется восемь светодиодных индикаторов, которые расположены по кругу, поэтому положение вращения двигателя можно наблюдать с помощью светодиодных индикаторов в соответствующих положениях.

Согласно техническим характеристикам энкодера, энкодер способен выдавать 8192 импульса на один оборот ротора двигателя. Следовательно, мы можем разделить эти 8192 импульса на восемь частей, и каждая лампочка светодиода соответствует одной части, то есть 1024 импульсам. Когда ротор двигателя поворачивается в соответствующее положение, соответствующий

светодиод горит, а другие светодиоды остаются выключенными. Когда двигатель продолжает вращаться, светодиоды также будут «вращаться» в соответствии с положением ротора двигателя.

Блок-схема алгоритма показана на рисунке ниже:



Рисунок 30 – Блок-схема раздела обратной связи

Энкодер подключен к двум последовательным портам платы STM32, а два последовательных порта подключены к фазе А и фазе В. Когда импульс фазы А поступает быстрее, чем импульс фазы В, двигатель вращается по часовой стрелке, в противном случае двигатель вращается против часовой стрелки. Последовательность включения светодиодов также будет последовательно загораться по часовой стрелке или против часовой стрелки.

4.4 Блок управления двигателем – Использование ПИД-регулятора

4.4.1 ПИД-регулятор

В этом проекте двигателю присваивается целевое значение положения в соответствии с положением шарика, а система обратной связи возвращает фактическое значение положения, но между ними возникает ошибка. Поэтому

нам нужен ПИД-регулятор, чтобы уменьшить ошибку. ПИД-регулятор регулирует входное значение на основе этой ошибки до тех пор, пока выходное значение не достигнет заданного значения [16].

Пропорционально-интегрально-дифференцирующий (ПИД) регулятор — устройство в управляющем контуре с обратной связью.

ПИД-регулятор состоит из трех компонентов: Пропорциональная составляющая, интегрирующая составляющая и дифференцирующая составляющая. У каждой части своя роль:

- пропорциональная: Пропорциональная составляющая вырабатывает выходной сигнал, противодействующий отклонению регулируемой величины от заданного значения, наблюдаемого в данный момент времени;
- интегрирующая: Интегрирующая составляющая пропорциональна интегралу по времени от отклонения регулируемой величины;
- дифференцирующая: Дифференцирующая составляющая пропорциональна темпу изменения отклонения регулируемой величины и предназначена для противодействия отклонениям от целевого значения, которые прогнозируются в будущем.

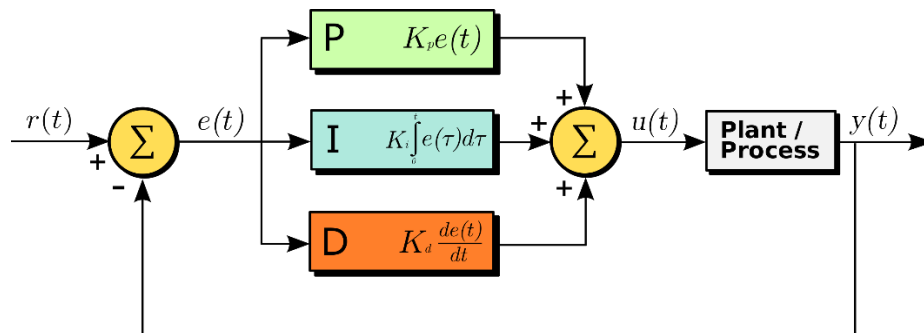


Рисунок 31 – Принципиальная схема ПИД-регулятора

4.4.1.1 ПИД-регулятор позиционного типа

ПИД-регулятор позиционного типа на самом деле является ПИД-регулятором, основанным на отклонении между фактическим положением текущей системы и положением, которого вы ожидаете достичь [17]. Формула расчета выглядит следующим образом (4.1):

$$U_k = K_p * e_k + K_i \sum_{j=0}^k e_k + K_d(e_k - e_{k-1}) \quad (4.1)$$

где K_p, K_i, K_d – коэффициенты усиления пропорциональной, интегрирующей и дифференцирующей составляющих регулятора соответственно.

Из приведенной выше формулы видно, что выход каждой операции ПИД-регулятора связан с прошлым состоянием, и ошибка интегрального члена будет накапливаться.

4.4.1.2 Инкрементальный ПИД-регулятор

Выход инкрементного ПИД-регулятора — это просто приращение управляющей величины. Когда количество управления, требуемое исполнительным механизмом, является инкрементным, тогда мы можем использовать алгоритм инкрементного ПИД-регулирования для управления. Формула расчета выглядит следующим образом (4.2):

$$\Delta U_k = K_p * (e(k) - e(k - 1)) + K_i * e_k + K_d * [e(k) - 2e(k - 1) + e(k - 2)] \quad (4.2)$$

где K_p, K_i, K_d – коэффициенты усиления пропорциональной, интегрирующей и дифференцирующей составляющих регулятора соответственно.

Для инкрементного PID при заданном входном значении отклонение между возвращаемым системой значением и установленным значением равно Err , а последнее отклонение $Last_Err$ и последнее отклонение $Previous_Err$ сохраняются в системе. Эти три входных значения можно использовать для расчета приращения ΔU_k вышеупомянутых управляющих переменных через инкрементный ПИД. Полученная управляющая переменная ΔU_k соответствует приращению последней ошибки положения, а не отклонению от фактического положения, то есть накопления ошибки нет. То есть контрольная сумма должна увеличиваться на основе последней контрольной суммы.

4.4.2 Реализация технологии ПИД-регулятора

Учитывая, что ключ проекта заключается в управлении положением вращения двигателя, выбран ПИД-регулятор позиционного типа.

В этом исследовательском проекте были разработаны два алгоритма для двух разных сценариев: шарик остается стабильным на панели и движение шарика по вращающейся панели.

4.4.2.1 Шарик остается стабильным на панели

Подобно самобалансирующемуся автомобилю, система должна управлять двигателем, чтобы он вращался в соответствующем положении в соответствии с положением шарика, чтобы качающийся шарик оставался стабильным и опирался на панель.

Для этого сценария была разработана двухконтурная система управления по положению и скорости. Контур скорости вложен в контур положения. Принципиальная схема показана на рисунке 32 ниже.

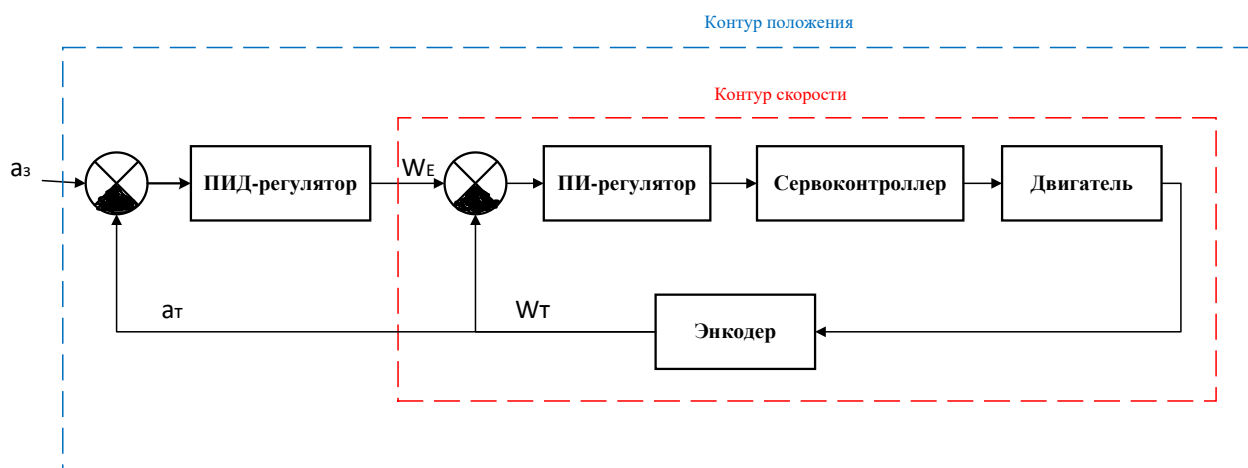


Рисунок 32 – Структурная схема двухконтурной системы управления

Описание: a_3 - целевой угол, который входит в контур положения как вход, и после действия ПИД-регулятора может быть получена ожидаемая скорость W_3 для достижения целевого положения. После этого в контур скорости входят скорость W_3 и силовая скорость двигателя, а после действия ПИД-регулятора снова регулируются скорость и положение вращения двигателя. Промышленные камеры и энкодеры используются в качестве обратной связи для определения положения шарика.

Запустим программу и осторожно переместите шарик рукой, чтобы он качался влево и вправо на панели. Система постоянно регулирует положение

вращения двигателя в соответствии с положением шарика и, наконец, стабилизирует шарик на панели (покое). Результат показан на рисунке 33 ниже.

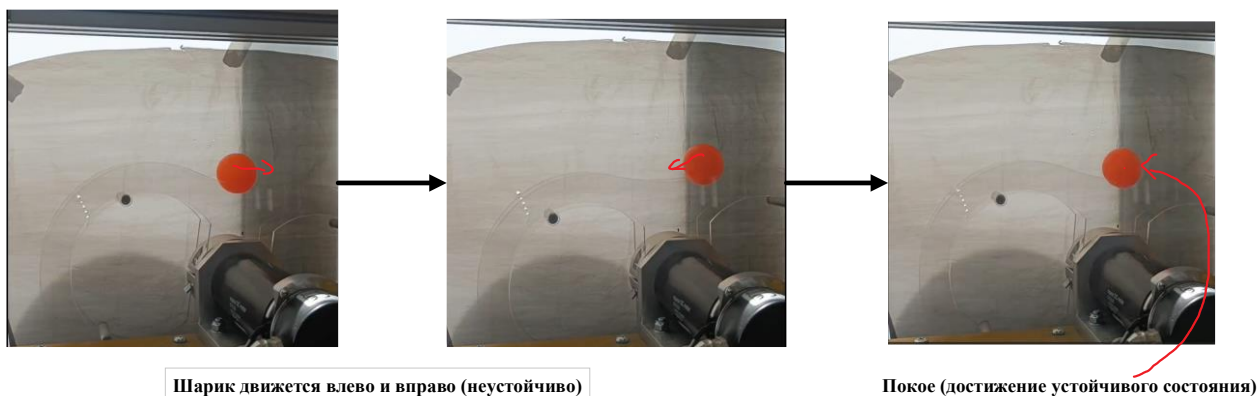


Рисунок 33 – Результат работы алгоритма

4.4.2.2 Движение шарика по вращающейся панели

В этой части устанавливается начальная скорость, чтобы двигатель вращался. Когда положение шарика распознано, двигатель изменит скорость в соответствии с различными положениями шарика, чтобы заставить шарик двигаться по панели.

Для этого сценария был разработан алгоритм ПИД-регулятора контура по скорости. Структурная схема этой части показана на рисунке 34.

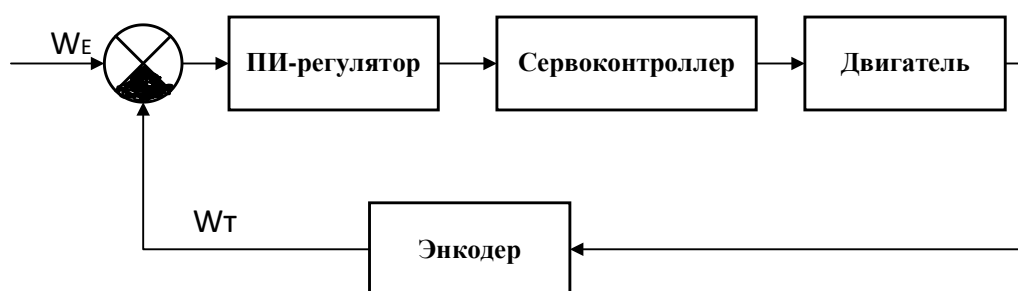


Рисунок 34 – Структурная схема управления двигателем

Так как движение мяча по панели очень сложное, это выходит за рамки моих возможностей на данном этапе, поэтому в этом проекте не разработан алгоритм, чтобы мяч двигался по вращающейся панели, не падая.

Алгоритм, разработанный в этом проекте, заставляет мяч двигаться 1,5 оборота по доске, а затем мяч падает. Конкретный метод алгоритма состоит в том, чтобы установить разные скорости для позиций с разными углами с

помощью нескольких попыток. Результат работы программы показан на рисунке 34 ниже:

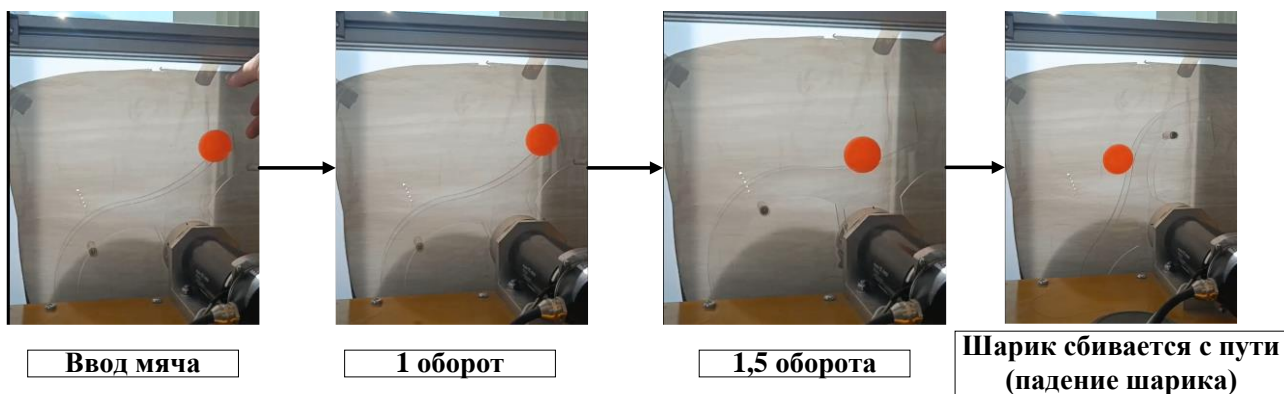


Рисунок 35 – Результат работы алгоритма

Математический алгоритм движения шарика слишком сложен, и на данном этапе невозможно реализовать идеальную ситуацию, когда шарик движется по вращающейся панели, не падая. Но цель проекта достигнута.

**ЗАДАНИЕ К РАЗДЕЛУ
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Обучающемуся:

Группа	ФИО
8E92	Го Цзыцзюнь

Школа	ИШИТР	Отделение школы (НОЦ)	Отделение автоматизации и робототехники
Уровень образования	Бакалавриат	Направление/специальность	15.03.06 Мехатроника и робототехника

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

<i>1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Стоимость материальных ресурсов определялась по средней рыночной стоимости. Оклады в соответствии с окладами сотрудников организации.
<i>2. Нормы и нормативы расходования ресурсов</i>	Тариф на электричество – 3,5 руб./кВт·ч, 30% районный коэффициент, 30% премиальный коэффициент.
<i>3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Коэффициент отчислений во внебюджетные фонды - 30 %

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<i>1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	Оценить коммерческий потенциал и перспективность проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения.
<i>2. Планирование и формирование бюджета научных исследований</i>	Определить трудоемкость выполнения проекта. Разработать план и рассчитать затраты.
<i>3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	Определить интегральный показатель эффективности исследования. Оценить сравнительную эффективность научного исследования (НИ).

Перечень графического материала:

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 к разделу в соответствии с календарным учебным графиком

Задание выдал консультант по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Былкова Татьяна Васильевна	К.Э.Н.		

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8E92	Го Цзыцзюнь		

5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является проектирование и создание конкурентоспособно разработанных технологий, отвечающих современным требованиям в области ресурсоэффективности и ресурсосбережения [18].

5.1 Потенциальные потребители результатов исследования

Для анализа потребителей результатов исследования необходимо рассмотреть целевой рынок и провести его сегментирование.

Целевым рынком проекта являются университеты, проводящие исследования в области автоматического управления, и центры исследования алгоритмов, разрабатывающие и эксплуатирующие оборудование автоматического управления. Сегментация рынка осуществляется на основе двух основных критериев: направления деятельности и размера организации. Карта сегментирования представлена в таблице 4 ниже:

Таблица 4 – Карта сегментирования

		Направления деятельности		
		Научные исследования	Образовательная деятельность	Проектирование оборудования
Размер организации	Мелкие			
	Средние			
	Крупные			

Согласно карте сегментирования рынка, можно сделать вывод, что для реализации разработки подходят образовательные учреждения, а также крупные организации, проводящие научные исследования и проектирующие оборудование в области автоматического управления. Для образовательных учреждений и технических исследовательских организаций стандарт является обязательным аппаратным требованием для проведения исследований.

5.2 Анализ конкурентных технических решений

Существуют следующие основные конкуренты:

- Стенд «робот бабочка» компании «Robotikum» (к1);
- Лабораторно-исследовательский комплекс «Робот Бабочка» компании «Образовательная робототехника» (к2).

Позиция разработки и конкурентов оценивается по каждому показателю экспертным путем по пятибалльной шкале, где 1 – наиболее слабая позиция, а 5 – наиболее сильная. Оценка конкурентоспособности технических решений представлена в таблице 5.

Таблица 5 – Оценочная карта для сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б _ф	Б _{к1}	Б _{к2}	К _ф	К _{к1}	К _{к2}
1	2	3	4	5	6	7	8
Технические критерии оценки ресурсоэффективности							
1. Безопасность	0,06	4	4	5	0,24	0,24	0,30
2. Простота использования	0,06	5	4	3	0,30	0,24	0,18
3. Улучшение производительности	0,10	5	4	3	0,50	0,40	0,30
4. Надежность	0,04	4	5	5	0,16	0,20	0,20
5. Энергоэкономичность	0,02	4	4	4	0,08	0,08	0,08
6. Уровень автоматизации	0,06	5	4	3	0,30	0,24	0,18
7. Возможность подключения к ПК	0,12	5	5	5	0,60	0,60	0,60
8. Качество интеллектуального интерфейса	0,04	4	3	3	0,16	0,12	0,12
Показатели оценки коммерческого потенциала разработки							
9. Конкурентоспособность продукта	0,12	5	4	4	0,60	0,48	0,48
10. Цена	0,12	5	4	4	0,60	0,48	0,48
11. Предполагаемый срок эксплуатации	0,10	4	4	4	0,40	0,40	0,40
12. Уровень проникновения на рынок	0,06	2	4	4	0,12	0,24	0,24
13. Срок выхода на рынок	0,04	3	5	4	0,12	0,20	0,16
14. Послепродажное обслуживание	0,06	4	4	4	0,24	0,24	0,24
Итого	1	59	58	55	4,42	4,16	3,96

По результатам оценочной карты разработанный стенд имеет наивысший общий балл. Основным преимуществом данного стенда перед конкурентами является простота использования, улучшение производительности, уровень автоматизации, конкурентоспособность и цена. Однако, он уступает двум другим продуктам по сроку выхода на рынок и проникновению на рынок.

5.3 SWOT-анализ

SWOT-анализ – это Метод стратегического планирования и стратегического управления, используемый, чтобы помочь человеку или организации определить Сильные и слабые стороны, Возможности и Угрозы, связанные с деловой конкуренцией или планированием проекта.

С помощью этого метода можно обозначить основные проблемы проекта, определить пути решения и перспективу развития. Первым шагом является описание сильных и слабых сторон проекта, а также выявление возможностей и угроз. Результаты первого этапа SWOT-анализа представлены в таблице 6.

Таблица 6 – Матрица SWOT

<p>Сильные стороны:</p> <p>С1. Снижение сложности обучения</p> <p>С2. Низкая стоимость оборудования</p> <p>С3. Актуальность разработки</p> <p>С4. Возможность использования в учебных целях</p> <p>С5. Наличие разных режимов работы.</p>	<p>Слабые стороны:</p> <p>Сл1. Наличие импортных компонентов в системе</p> <p>Сл2. Медленный вывод на рынок разработанного стенда</p> <p>Сл3. Большие габариты установки стенда</p> <p>Сл4. Медленный вывод на рынок разработанного стенда</p>
--	---

Продолжение таблицы 6 – Матрица SWOT

<p>Возможности:</p> <p>В1. Исследование и разработка более эффективных алгоритмов</p> <p>В2. Доступ к исследовательским грантам</p> <p>В3. Использование стенда для исследования автоматизации технологических процессов</p> <p>В4. Замена сложных компонентов стендов</p> <p>В5. Модернизировать оборудование стенда «Робот-бабочка»</p>	<p>Угрозы:</p> <p>У1. Отсутствие спроса на рынке</p> <p>У2. Развивающаяся конкуренция на рынке</p> <p>У3. Малый объем рынка сбыта</p> <p>У4. Санкции на оборудования стенда</p>
--	--

Вторым этапом является выявление соответствия сильных и слабых сторон научно-исследовательского проекта внешним условиям окружающей среды. Интерактивные матрицы проекта представлены в таблицах 7 – 10.

Таблица 7 – Интерактивная матрица сильных сторон и возможностей проекта

	Сильные стороны проекта					
		C1	C2	C3	C4	C5
Возможности проекта	B1	+	-	-	-	+
	B2	-	-	-	+	-
	B3	+	-	-	-	+
	B4	-	+	-	-	-
	B5	-	+	+	-	-

Таблица 8 – Интерактивная матрица слабых сторон и возможностей проекта

	Слабые стороны проекта				
		Сл1	Сл2	Сл3	Сл4
Возможности проекта	В1	-	+	-	+
	В2	0	-	-	-
	В3	+	-	-	+
	В4	+	-	-	-
	В5	+	+	-	0

Таблица 9 – Интерактивная матрица сильных сторон и угроз проекта

	Сильные стороны проекта					
		С1	С2	С3	С4	С5
Угрозы проекта	У1	-	+	+	+	-
	У2	-	+	-	-	-
	У3	-	-	-	-	-
	У4	-	-	-	-	-

Таблица 10 – Интерактивная матрица слабых сторон и угроз проекта

	Слабые стороны проекта				
		Сл1	Сл2	Сл3	Сл4
Угрозы проекта	У1	+	+	-	+
	У2	-	0	-	-
	У3	0	0	-	-
	У4	+	+	-	-

В рамках третьего этапа была составлена итоговая матрица SWOT-анализа. Итоговая матрица представлена в таблице 11.

Таблица 11 – SWOT-анализ

	<p>Сильные стороны:</p> <p>С1. Снижение сложности обучения</p> <p>С2. Низкая стоимость оборудования</p> <p>С3. Актуальность разработки</p> <p>С4. Возможность использования в образовательных целях</p> <p>С5. Наличие разных режимов работы.</p>	<p>Слабые стороны:</p> <p>Сл1. Наличие импортных компонентов в системе</p> <p>Сл2. Медленный вывод на рынок разработанного стенда</p> <p>Сл3. Большие габариты установки стенда</p> <p>Сл4. Отсутствие прототипа</p>
<p>Возможности:</p> <p>В1. Исследование и разработка более эффективных алгоритмов</p> <p>В2. Доступ к исследовательским грантам</p> <p>В3. Использование стенда для исследования автоматизации технологических процессов</p> <p>В4. Замена сложных компонентов стендов</p> <p>В5. Модернизировать оборудование стенда «Робот-бабочка»</p>	<p>Снижение сложности обучения помогает разработать более эффективные алгоритмы и автоматизировать процесс использования стенда.</p> <p>Невысокая стоимость оборудования позволяет производить замену сложных частей стенда и программно-аппаратную модернизацию оборудования на стенде.</p> <p>Поскольку стенд может быть использован для технического образования, он помогает образовательным и исследовательским учреждениям получать исследовательские гранты.</p>	<p>Использование импортного оборудования может стать проблемой для внедрения разработки в отечественную промышленность. В то же время отсутствие прототипов может сказаться на использовании и модернизации стенда.</p>

Продолжение таблицы 11 – SWOT-анализ

<p>Угрозы:</p> <p>У1. Отсутствие спроса на рынке</p> <p>У2. Развивающаяся конкуренция на рынке</p> <p>У3. Малый объем рынка сбыта</p> <p>У4. Санкции на оборудования стенда</p>	<p>Низкая стоимость оборудования стенда, актуальность стенда и возможность его использования в образовательных целях позволяют повысить спрос на стенды на рынке.</p>	<p>Наличие импортных компонентов, Медленный вывод на рынок разработанного стенда и отсутствие прототипа являются причинами отсутствия рыночного спроса. На использование импортного оборудования могут повлиять санкции.</p>
--	---	--

В результате SWOT-анализа были рассмотрены сильные и слабые стороны модернизации учебно-исследовательского стенда, выявлены возможности, которые позволят систему сделать лучше по сравнению с конкурентами и рассмотрены варианты угроз, которые могут этому помешать. Для уменьшения угроз и борьбы со слабыми сторонами необходимо:

- Снизить зависимость от импортных компонентов оборудования. Максимально заменить импортные оборудования системы на отечественные;
- Производить анализ деятельности конкурентов на рынке и действовать на опережение, расширяя функционал системы и повышая качество стенда.

5.4 Планирование научно-исследовательских работ

5.4.1 Структура работ в рамках научного исследования

Для реализации научного проекта необходимы два исполнителя – руководитель (Р) и инженер (И). Проектная работа делиться на этапы, по каждому виду запланированных работ установлена соответствующая должность исполнителей. Этапы реализации проекта представлены в таблице 12.

Таблица 12 – Этапы реализации проекта

Основные этапы	№	Содержание работ	Исполнитель
Выбор направления исследования	1	Выбор направления научного исследования	Р, И
	2	Постановка основных целей и задач	Р
Разработка технического задания	3	Составление и утверждение технического задания	Р, И
Анализ предметной области	4	Обзор технической литературы	И
	5	Календарное планирование работ	Р, И
Теоретические и экспериментальные исследования	6	Подбор оборудования стенда	И
Теоретические и экспериментальные исследования	7	Разработка алгоритма опроса датчика положения шарика	И
	8	Разработка алгоритма программы передачи координат со стороны Pthon на STM32	И
	9	Разработка алгоритмов для управления двигателям	И
	10	Отладка параметров	И
	11	Оптимизация алгоритмов	И
	12	Сборка исследовательского стенда	И
Дополнительные разделы	13	Написание раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»	И
	14	Написание раздела «Социальная ответственность»	И
Проверка результатов	15	Проверка работы руководителем	Р
Оформление отчета по НИР	16	Составление пояснительной записки	И

5.4.2 Определение трудоемкости выполнения работ и диаграмма

Ганта

В большинстве случаев, затраты на оплату труда составляют большую часть стоимости разработки, поэтому неотъемлемым действием определения ресурсоэффективности является определение трудоемкости работ участников проектирования. Рассчитываем среднее значение трудоемкости $T_{ожі}$ по следующей формуле (5.1):

$$T_{ожі} = \frac{3 \cdot T_{mini} + 2 \cdot T_{maxi}}{5} \quad (5.1)$$

где $T_{ожі}$ – ожидаемая трудоемкость выполнения i -ой работы, чел/дн.;

T_{mini} – минимальная трудоемкость i -ой работы, чел/дн.;

T_{maxi} – максимальная трудоемкость i -ой работы, чел/дн.

Далее рассчитываем продолжительность одной работы по формуле (5.2):

$$T_{pi} = \frac{T_{ожі}}{Ч_i}, \quad (5.2)$$

где T_{pi} – продолжительность одной работы, раб. дн.;

$T_{ожі}$ – ожидаемая трудоемкость выполнения одной работы, чел.-дн.;

$Ч_i$ – количество исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

Длительность каждого из этапов работ из рабочих дней необходимо перевести в календарные дни согласно формуле (5.3):

$$T_{ki} = T_{pi} \cdot k_{кал} \quad (5.3)$$

где T_{ki} – продолжительность выполнения i -й работы, календ. дн.;

T_{pi} – продолжительность выполнения i -й работы, раб. дн.;

$T_{кал}$ – коэффициент календарности.

Коэффициент календарности определяется по формуле (5.4):

$$k_{кал} = \frac{T_{кал}}{T_{кал} - T_{вых} - T_{пр}}, \quad (5.4)$$

Таким образом, коэффициент календарности равен:

$$k_{кал} = \frac{365}{365 - 118} = 1,480$$

Результаты по трудоемкости выполнения работ представлены в таблице

13.

Таблица 13 – Временные показатели проведения исследования

№ работы	Трудоёмкость работ						Длительность работ в рабочих днях T_{pi}		Длительность работ в календарных днях T_{ki}	
	T_{min} , чел- дни		T_{max} , чел- дни		$T_{ожі}$, чел- дни		Инженер	Руководитель	Инженер	Руководитель
	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель				
1	2	2	5	5	3,2	3,2	1,6	1,6	2,4	2,4
2	0	2	0	5	0	3,2	0	3,2	0	4,7
3	5	1	10	3	7	1,8	3,5	0,9	5,2	1,3
4	14	0	30	0	20,4	0	20,4	0	30,2	0
5	1	1	3	3	1,8	1,8	0,9	0,9	1,3	1,3
6	1	0	5	0	2,6	0	2,6	0	3,8	0
7	1	0	5	0	2,6	0	2,6	0	3,8	0
8	7	0	14	0	9,8	0	9,8	0	14,5	0
9	3	0	7	0	4,6	0	4,6	0	6,8	0
10	7	0	14	0	9,8	0	9,8	0	14,5	0
11	7	0	14	0	9,8	0	9,8	0	14,5	0
12	14	0	30	0	20,4	0	20,4	0	30,2	0
13	3	0	7	0	4,6	0	4,6	0	6,8	0
14	2	0	5	0	3,2	0	3,2	0	4,7	0
15	0	8	0	15	0	10,8	0	10,8	0	16
16	1	0	3	0	1,8	0	1,8	0	2,7	0
Итого	68	14	152	31	102	21	96	17	141	26

Согласно полученным результатам, разработка учебно-исследовательского стенда на основе «Робот-бабочка» займет 17 рабочих день у руководителя и 96 рабочих дня у инженера.

Диаграмма Ганта – Это популярный тип столбчатых диаграмм, который используется для иллюстрации плана, графика работ по какому-либо проекту. По полученным данным была построена диаграмма Ганта, представленная на рисунке 36.

№	Вид работ	Исполнители	Ткi, кал. Дни	Продолжительность выполнения работ															
				Февраль			Март			Апрель			Май			Июнь			
				1	2	3	1	2	3	1	2	3	1	2	3	1	2		
1	Выбор направления научного исследования	Р, И	2	■															
2	Постановка основных целей и задач	Р	5	■															
3	Составление и утверждение технического задания	Р, И	5		■														
4	Обзор научно-технической литературы	И	30		■	■	■												
5	Календарное планирование работ	Р, И	1				■												
6	Подбор оборудования	И	4					■											
7	Разработка алгоритма опроса датчика положения	И	4						■										
8	Разработка алгоритма программы передачи координат из Python в STM	И	5							■									
9	Разработка алгоритмов для управления двигателями	И	7								■								
10	Отладка параметров	И	15									■							
11	Оптимизация алгоритма	И	15										■						
12	Сборка стенда	И	30											■	■				
13	Написание раздела «финансовый менеджмент, ресурсоэффективность и ресурсосбережение»	И	7													■			
14	Написание раздела «социальная ответственность»	И	5														■		
15	Проверка работы руководителем	Р	16															■	
16	Составление пояснительной записки	И	3																■

■ – Инженер и руководитель; ■ – Инженер; ■ – Руководитель;

Рисунок 36 – Диаграмма Ганта

Как показано на рисунке выше, больше времени было выделено на сборку стенда и обзор литературы.

5.5 Бюджет научно-технического исследования (НТИ)

5.5.1 Расчет материальных затрат НТИ

Для разработки данного продукта необходимы следующие ресурсы:

- Персональный компьютер (ПК);
- Материальные ресурсы: оборудование для сборки стенда (датчики и энкодеры и т.д.), расходные материалы (канцелярия, провода изоляция и т.д.).

Расчет материальных затрат представлен в таблице 14.

Таблица 14 – Материальные затраты

Наименование	Единица измерения	Количество	Цена за ед., руб.	Цена (всего), руб.
Промышленная камера <i>Basler</i>	Шт.	1	39 101,3	39 101,3
Двигатель <i>Maxon</i>	Шт.	1	54 619	54 619
Сервоконтроллер	Шт.	1	38 409	38 409
Энкодер	Шт.	1	40 983	40 983
Микроконтроллер	Шт.	1	11 580	11 580
Кабель для передачи данных с фотоаппарата на компьютер	Шт.	1	1 500	1 500
Кабель для передачи данных USB	Шт.	2	1 500	3 000
Маленький шарик	Шт.	1	50	50
Оргстекло	Шт.	9	1 000	9 000
Итого, руб.				198 242,3

По расчетам материальные затраты для модернизации стенда «Робот-бабочка» составляет 198 242,3 рублей.

5.5.2 Расчет амортизационных отчислений на реализацию проекта

Разработка проекта производится в течение 5 месяцев с использованием персонального компьютера первоначальной стоимостью 65 000 рублей. Срок его полезного использования составляет 3 года.

Норма амортизации рассчитывается согласно формуле (5.5):

$$N_A = \frac{1}{T} \cdot 100\% \quad (5.5)$$

где T – срок полезного использования, лет.

Тогда для используемого персонального компьютера составит:

$$N_A = \frac{1}{3} \cdot 100\% = 33.3\%$$

Годовые амортизационные отчисления:

$$A_{\text{год}} = \frac{65000 \times 33.3}{100} = 21645 \text{ руб.}$$

Амортизационные отчисления за 5 месяцев составят:

$$A = \frac{21645 \times 5}{12} = 9018,75 \text{ руб.}$$

По результатам расчетов амортизационные отчисления на реализацию проекта в течение 5 месяцев составили 9018,75 рублей.

5.5.3 Основная заработная плата исполнителей темы

В этом разделе рассчитывается основная заработная плата.

Основная плата работников, непосредственно занятых выполнением НИИ равняется произведению среднедневной платы работника и количества рабочих дней (5.6):

$$Z_{\text{осн}} = Z_{\text{дн}} \times T_{\text{раб}} \quad (5.6)$$

где $Z_{\text{дн}}$ – среднедневная заработная плата, руб.;

$T_{\text{раб}}$ – продолжительность работ, раб. дн.

Среднедневная заработная плата рассчитывается по формуле (5.7):

$$Z_{\text{дн}} = \frac{Z_{\text{м}} \times \text{М}}{F_{\text{д}}} \quad (5.7)$$

где $Z_{\text{м}}$ – месячный должностной оклад работника, руб.;

М – количество месяцев работы без отпуска в течение года;

$F_{\text{д}}$ – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн.

Месячный должностной оклад работника рассчитывается согласно следующей формуле (5.8):

$$Z_{\text{м}} = Z_{\text{ТС}} \times (1 + K_{\text{пр}} + K_{\text{д}}) \times K_{\text{р}} \quad (5.8)$$

где $Z_{\text{ТС}}$ – заработная плата по тарифной ставке, руб.;

$K_{\text{пр}}$ – премиальный коэффициент, равный 0,3;

$K_{\text{д}}$ – коэффициент доплат и надбавок, равный примерно 0,2 – 0,5;

$K_{\text{р}}$ – районный коэффициент, равный 1,3 (для Томска).

Баланс рабочего времени для руководителя и инженера представлен в таблице 15.

Таблица 15 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Инженер
Календарное число дней	365	365
Количество нерабочих (выходных и праздничных) дней	118	118
Потери рабочего времени – отпуск – невыходы по болезни	48	72
Количество месяцев без отпуска	10,6	9,8
Действительный годовой фонд рабочего времени	199	175

Действительный годовой фонд рабочего времени руководителя и инженера составляют 199 и 175 дней соответственно.

Результаты основной заработной платы приводятся в таблице 16.

Таблица 16 – Расчет основной заработной платы

Исполнители	Оклад, руб.	$k_{пр}$	$k_{д}$	K_p	Z_m , руб.	$Z_{дн}$	Траб, дни	$Z_{осн}$, руб.
Руководитель	38 000	0,3	0,2	1,3	74 100	3 947,04	17	136 281,6
Инженер	13 000	0,3	0,2	1,3	25 350	1 419,6	96	67 099,6

Как показано в таблице выше, основная заработная плата составит 67 099,6 рубля для инженера и 136 281,6 для руководителя соответственно.

5.5.4 Дополнительная заработная плата исполнителей темы

В этом разделе рассчитывается дополнительная заработная плата.

Расчет дополнительной заработной платы ведется по следующей формуле (5.13):

$$Z_{доп} = Z_{осн} \times K_{доп} \quad (5.9)$$

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

Примем $K_{доп} = 0,12$, тогда дополнительная плата руководителя и инженера будет рассчитываться следующим образом:

$$З_{\text{доп.Р}} = 136\,281,6 * 0,12 = 16\,353,79 \text{ руб.}$$

$$З_{\text{доп.И}} = 67\,099,6 * 0,12 = 8\,051,95 \text{ руб.}$$

Дополнительная заработная плата составит 16353,79 рубля для инженера и 8051.95 для руководителя соответственно.

Следовательно, итоговая заработная плата для руководителя и инженера:

$$З_{\text{зп.Р}} = З_{\text{осн.Р}} + З_{\text{доп.Р}} = 136\,281,6 + 16\,353,79 = 152\,635,39 \text{ руб.}$$

$$З_{\text{зп.И}} = З_{\text{осн.И}} + З_{\text{доп.И}} = 67\,099,6 + 8\,051,95 = 75\,151,55 \text{ руб.}$$

5.5.5 Отчисления во внебюджетные фонды

Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы (5,10):

$$З_{\text{внеб}} = K_{\text{внеб}} * (З_{\text{осн}} + З_{\text{доп}}) \quad (5.10)$$

где $K_{\text{внеб}}$ – коэффициент отчислений на уплату во внебюджетные фонды.

Размер страховых взносов равен 30 % от заработной платы. Сюда включены взносы на пенсионное страхование – 22 %, на медицинское страхование – 5,1 %, а также на соцстрахование – 2,9 %. Отчисления во внебюджетные фонды представлены в таблице 17.

Таблица 17 – Отчисление во внебюджетные фонды

Исполнитель	Заработная плата, руб.	Коэффициент отчислений во внебюджетные фонды, %	Отчисления во внебюджетные фонды, руб.
Руководитель	152 635,39	30	45 790,62
Инженер	75 151,55	30	22 545,47
Итого			68 336,09

Сумма отчислений во внебюджетные фонды для руководителя и инженера составит 68336,09 руб.

5.5.6 Накладные расходы

Накладные расходы включают в себя затраты на управление, хозяйственное обслуживание, эксплуатацию и ремонт оборудования и составляют 15-20% от суммы заработной платы и отчислений (5.11):

$$C_H = K_H \times (Z_{зпр} + Z_{зпи}) \quad (5.11)$$

где C_H – накладные расходы, руб.;

K_H – коэффициент накладных расходов;

$Z_{зпи}$ – заработная плата инженера, руб.;

$Z_{зпр}$ – заработная плата руководителя, руб.

Выбираем $K_H = 20\%$, и вычислим накладные расходы:

$$C_H = 0,2 * (152\,635,39 + 75151,55) = 45\,557,39 \text{ руб.}$$

Таким образом, накладные расходы проекта составят 45 557,39 руб.

5.5.7 Формирование бюджета затрат научно-исследовательского проекта

Определение бюджета затрат на проект приведен в таблице 18.

Таблица 18 – Расчет бюджета затрат НИИ

Наименование статьи		Сумма, руб.
Материальные затраты		198 242,3
Амортизационные расходы		9 018,75
Наименование статьи	Сумма, руб.	Наименование статьи
Затраты по заработной плате работников	- руководитель	152 635,39
	- инженер	75 151,55
Отчисления во внебюджетные фонды	- руководитель	45 790,62
	- инженер	22 545,47
Накладные расходы		45 557,39
Бюджет затрат НИИ		548 941,47

Согласно расчетам, бюджет затрат научно-исследовательской работы составил 548 941,47 рублей. Соотношение каждой части следующее: материальные затраты – 36,1 %, амортизационные расходы – 1,6 %, затраты по

заработной плате работников – 41,5 %, отчисления во внебюджетные фонды – 12,4 % и накладные расходы – 8,4 %.

5.5.8 Определение ресурсной, финансовой и экономической эффективности исследования

Интегральный финансовый показатель разработки определяется по формуле (5.12):

$$I_{\text{финр}}^{\text{испл.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}} \quad (5.12)$$

где $I_{\text{финр}}^{\text{испл.}i}$ – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

На рынке есть два аналога:

– Стенд «Butterfly» компании «Robotikum» – 472 830 рублей;

– Лабораторно-исследовательский комплекс «Butterfly» компании «Образовательная робототехника» – 747 740 рублей.

Реализация разрабатываемого стенда составит 548 941,47 рублей.

Вычислим интегральный финансовый показатель разработки относительно конкурентов:

$$I_{\Phi}^{\text{разр.}} = \frac{548\,941,47}{747\,740} = 0,73$$

$$I_{\Phi}^{a1} = \frac{472\,830}{747\,740} = 0,63$$

$$I_{\Phi}^{a2} = \frac{548\,941,47}{747\,740} = 1$$

Видим, что аналог 2 дороже разрабатываемого стенда.

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить по формуле (5.13):

$$I_{pi} = \sum a_i \cdot b_i \quad (5.13)$$

где I_{pi} – интегральный показатель ресурсоэффективности для i -го варианта исполнения разработки;

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Расчет интегрального показателя ресурсоэффективности проводит в таблице 19. (Аналог 1 – стенд «Butterfly-Robot» компании «Robotikum», Аналог 2 – Лабораторно-исследовательский комплекс «Butterfly» компании «Образовательная робототехника»)

Таблица 19 – Сравнительная оценка характеристик проекта

Критерии / Оценки	Весовой коэффициент параметра	Текущий проект	аналог 1	аналог 2
1. Энергоэффективность	0.05	4	3	3
2. Помехоустойчивость	0.2	3	3	4
3. Надежность	0.2	5	4	4
4. Унифицированность	0.1	3	4	4
5. Удобство эксплуатации	0.1	5	4	3
6. Простота эксплуатации	0.15	5	4	3
7. Безопасность	0,2	4	4	5
ИТОГО	1	29	26	26

$$I_m^{\text{разр.}} = 0,05 * 4 + 0,2 * 3 + 0,2 * 5 + 0,1 * 3 + 0,1 * 5 + 0,15 * 5 + 0,2 * 4 = 4,15$$

$$I_m^{a1} = 0,05 * 3 + 0,2 * 3 + 0,2 * 4 + 0,1 * 4 + 0,1 * 4 + 0,15 * 4 + 0,2 * 4 = 3,75$$

$$I_m^{a2} = 0,05 * 3 + 0,2 * 4 + 0,2 * 4 + 0,1 * 4 + 0,1 * 3 + 0,15 * 3 + 0,2 * 5 = 3,90$$

По результатам можно делать вывод, что расчетов разрабатываемый проект ресурсоэффективнее обоих аналогов.

Интегральный показатель эффективности вариантов исполнения определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле (5.14):

$$I_{\text{исп.}i} = \frac{I_{pi}}{I_{\text{испл.}i}^{\text{финр}}} \quad (5.14)$$

По формуле вычислим и получим:

$$I_{\text{исп.разр.}} = \frac{4,15}{0,73} = 5,68$$

$$I_{\text{исп.а1}} = \frac{3,75}{0,63} = 5,95$$

$$I_{\text{исп.а2}} = \frac{3,90}{1} = 3,90$$

Сравнительная эффективность вариантов исполнения рассчитывается по формуле (5.15):

$$\mathcal{E}_{\text{ср}} = \frac{I_{\text{разраб}}}{I_{\text{ан.}i}} \quad (5.15)$$

где $\mathcal{E}_{\text{ср}}$ – сравнительная эффективность проекта;

$I_{\text{разраб}}$ – интегральный показатель эффективности разработки;

$I_{\text{ан.}i}$ – интегральный показатель эффективности аналога.

$$\mathcal{E}_{\text{ср1}} = \frac{5,68}{5,95} = 0,95$$

$$\mathcal{E}_{\text{ср2}} = \frac{5,68}{3,90} = 1,46$$

Расчет сравнительной эффективности разработки представлен в таблице 20.

Таблица 20 – Сравнительная эффективность разработки

Показатели	Разрабатываемый проект	Аналог 1	Аналог 2
Интегральный финансовый показатель разработки	0,73	0,63	1
Интегральный по казатель ресурсоэффективности разработки	4,15	3,75	3,90
Интегральный показатель эффективности	5,68	5,95	3,90
Сравнительная эффективность вариантов исполнения	-	0,95	1,46

5.6 Выводы по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

В ходе разработки данного раздела были выявлены потенциальные потребители стенда «Робот-бабочка». В разделе анализа конкурентных технических решений были определены конкуренты разработки. Также создана диаграмма SWOT-анализа для анализа сильных сторон, слабых сторон, возможностей и угроз проекта. Кроме того, были произведены расчеты трудоемкости НИИ. По диаграмме Ганта больше времени было выделено на сборку стенда и обзор литературы. Согласно расчетам, бюджет затрат научно-исследовательской работы составил 548 941,47 рублей. Соотношение каждой части следующее: материальные затраты – 36,1 %, амортизационные расходы – 1,6 %, затраты по заработной плате работников – 41,5 %, отчисления во внебюджетные фонды – 12,4 % и накладные расходы – 8,4 %.

В процесс анализа ресурсной, финансовой и экономической эффективности демонстрирует преимущества разработанного стенда перед конкурентами.

**ЗАДАНИЕ К РАЗДЕЛУ
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Обучающемуся:

Группа	ФИО
8Е92	Го Цзыцзюнь

Школа	ИШИТР	Отделение (НОЦ)	Отделение автоматизации и робототехники
Уровень образования	Бакалавриат	Направление/специальность	15.03.06 Мехатроника и робототехника

Тема ВКР:

Модернизация учебно-исследовательского стенда «Робот-бабочка»	
Исходные данные к разделу «Социальная ответственность»:	
<p>Введение</p> <ul style="list-style-type: none"> – Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения. – Описание рабочей зоны (рабочего места) при разработке проектного решения/при эксплуатации 	<p><i>Объект исследования:</i> учебно-исследовательского стенда «Робот-бабочка».</p> <p><i>Область применения:</i> учебные задачи и задачи разработки алгоритма.</p> <p><i>Рабочая зона:</i> лаборатории и аудитории.</p> <p><i>Размеры помещения:</i> 40 м².</p> <p><i>Рабочее место:</i> кабинет 204 отделения ОАР, ИШИТР.</p> <p><i>Количество и наименование оборудования рабочей зоны:</i> Персональный компьютер, учебно-исследовательский стенд «Робот-бабочка» и осциллограф.</p> <p><i>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне:</i> замена и использование аппаратного оборудования, написание алгоритма управления и отладка алгоритма на стенде «Робот-бабочка».</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p>1. Правовые и организационные вопросы обеспечения безопасности при эксплуатации:</p> <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<p>Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 19.12.2022).</p> <p>ГОСТ 12.2.032-78 Рабочее место при выполнении работ сидя.</p>
<p>2. Производственная безопасность при эксплуатации:</p> <ul style="list-style-type: none"> – Анализ выявленных вредных и опасных производственных факторов 	<p>Вредные факторы:</p> <ol style="list-style-type: none"> 1. Превышение уровня шума; 2. Отсутствие или недостаток необходимого естественного освещения; 3. Отклонение показателей микроклимата в рабочей зоне; 4. Производственные факторы, связанные с напряженностью электрического поля.

	<p>Опасные факторы:</p> <p>1. Производственные факторы, связанные со статическим электричеством;</p> <p>2. Производственные факторы, связанные с электрическим током.</p> <p>Требуемые средства коллективной и индивидуальной защиты от выявленных факторов: беруши, защитные наушники, защитные очки, устройство защиты питания, осветительное устройство, коллективная защита (вентиляция, заземление).</p>
3. Экологическая безопасность при эксплуатации:	<p>Воздействие на литосферу: промышленные отходы, образующиеся в процессе производства.</p> <p>Воздействия на атмосферу: вредный газ, образующийся при потребляемой мощности зарядного устройства; выхлопные газы сварочного процесса.</p> <p>Воздействие на селитебную зону: шум при работающем двигателе.</p>
4. Безопасность в чрезвычайных ситуациях при эксплуатации:	<p>Возможные ЧС: пожар, взрыв из-за технической аварии.</p> <p>Наиболее типичная ЧС: пожар из-за неправильной эксплуатации.</p>

Дата выдачи задания к разделу в соответствии с календарным учебным графиком	
---	--

Задание выдал консультант по разделу «Социальная ответственность»:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД	Мезенцева Ирина Леонидовна	-		

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8Е92	Го Цзыцзюнь		

6 Социальная ответственность

6.1 Введение

С непрерывным развитием науки и техники с каждым годом увеличивается интенсивность использования компьютерных технологий и технологий компьютерного зрения в сфере жизнедеятельности человека. В этой исследовательской работе в стенде «Робот-бабочка» используются обе эти технологии.

В данной исследовательской работе были проведены аппаратные и программные обновления учебно-исследовательского стенда «Робот-бабочка». Конструкция стенда представляет собой каркас кубической конструкции. На стенде представлено различное оборудование, в том числе промышленная камера, двигатель постоянного тока, блок питания, сервоконтроллер, микроконтроллер и энкодеры. Посередине стенда находится панель в форме бабочки, основной материал панели - оргстекло.

При работе с компьютером и стендом «робот-бабочка» человек подвергается воздействию многих опасных и вредных производственных факторов: коротких замыканий, шума и вибраций. В данном разделе будут рассмотрены и проанализированы эти факторы, влияющие на здоровье человека.

6.2 Правовые и организационные вопросы обеспечения безопасности

6.2.1 Особенности законодательного регулирования проектных решений

Государственный надзор и контроль за организациями любых организационно-правовых форм и форм собственности проверяются специально уполномоченными государственными органами в соответствии с федеральным законом [19]. Этими специально уполномоченными государственными органами являются: Федеральная инспекция труда, Государственная экспертиза условий труда, Федеральная служба по труду и занятости населения, Федеральная служба по экологическому, технологическому и атомному надзору (Госгортехнадзор, Госэнергонадзор, Госатомнадзор России) Федеральная служба по надзору в

сфере защиты прав потребителей и благополучия человека (Госсанэпиднадзор России) и др. В стране также нормально функционирует Единая государственная система предупреждения и ликвидации чрезвычайных ситуаций (РСЧС), положения которой утверждены Российской Федерацией постановлением Правительства РФ, в соответствии с которым система объединяет органы управления, силы и значит.

6.2.2 Организационные мероприятия при компоновке рабочей зоны

Основным видом работы, выполняемой с использованием учебно-исследовательского стенда «Робот-бабочка», была работа на персональном компьютере с небольшим использованием оборудования на стенде или прямыми манипуляциями с стендом. Рабочее место напрямую влияет на производительность. Согласно нормативному документу *ГОСТ 12.2.032-78* к рабочему месту предъявляются следующие требования [20]:

- Рабочее место должно занимать площадь не менее 6 м², высота помещения должна быть не менее 4 м, а объем – не менее 20 м³ на человека;
- Высота над уровнем пола рабочей поверхности, за которой работает оператор, составляет 720 мм;
- Рабочие стулья должны быть сконструированы таким образом, чтобы поддерживать правильную рабочую осанку при работе с ПК;
- Оптимальный размер для рабочего стола 1600x1000 кв. мм;
- Под столом должно быть пространство для ног с глубиной 650 мм.

Также рабочий стол должен быть стабильным.

Рабочее место этого проекта — 204 кабинет корпуса №10 ТПУ. Данное рабочее место соответствует требованиям *ГОСТ 12.2.032-78*.

В дополнение к этому необходимо держать наиболее часто используемые предметы на расстоянии вытянутой руки, что сведет к минимуму время, затрачиваемое на повседневные операции. на рабочем месте оборудованы рабочие столы, которые соответствуют рисунку 37.



Рисунок 37 – Размещение составляющих ПК

6.3 Производственная безопасность

В нормативном документе «ГОСТ 12.0.003-2015 ССБТ. Опасные и вредные производственные факторы. Классификация» даны основания для классификации и метод классификации по некоторым вредным и опасным признакам в производственном процессе [21]. В следующей таблице 21 перечислены возможные вредные и опасные производственные факторы и соответствующие нормативные документы в процессе разработки проекта.

Таблица 21 – Возможные вредные и опасные факторы и соответствующие нормативные документы

Факторы (по ГОСТ 12.0.003-2015)	Нормативные документы
Вредные	ГОСТ 12.1.003-2014 Система стандартов безопасности труда. Шум. Общие требования безопасности. СНиП 23-05-95* Естественное и искусственное освещение. СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания.
1. Превышение уровня шума	
2. Отсутствие или недостаток необходимого естественного освещения	
3. Отклонение показателей микроклимата в рабочей зоне	
4. Производственные факторы, связанные с напряженностью электрического поля	ГОСТ 12.1.038-82 ССБТ. Электробезопасность. Предельно допустимые значения напряжений прикосновения и токов. ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление. Зануление.
Опасные	
1. Производственные факторы, связанные со статическим электричеством	ГОСТ 12.1.038-82 ССБТ. Электробезопасность. Предельно допустимые значения напряжений прикосновения и токов. ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление. Зануление.
2. Производственные факторы, связанные с электрическим током	

6.3.1 Превышение уровня шума

Одним из важных и распространенных факторов, влияющих на работу, является шум. Оборудование в рабочей зоне является основным источником шума, а работа оборудования сопровождается шумом. Длительное нахождение в шумной обстановке не только снижает качество работы, но и вызывает дискомфортные симптомы, такие как головная боль, раздражительность, ухудшение памяти и повышенная утомляемость, и даже вызывает необратимое поражение ушей и других органов. Было задокументировано, что длительное воздействие громкого шума (выше 80 дБ(А)) вызывает частичную или полную потерю слуха у людей. Допустимые уровни звукового давления на рабочем месте представлены в таблице 22.

Таблица 22 – Допустимые уровни звукового давления на рабочем месте

Уровни звукового давления, дБ, в октавных полосах со среднегеометрическими частотами, Гц									Эквивалентные уровни звука, дБА
31,5	63	125	250	500	1000	2000	4000	8000	
107	95	87	82	78	75	73	71	69	80

Ношение наушников и берушей может уменьшить воздействие шума на организм человека.

Основными источниками шума проекта являются:

- вращение охлаждающего вентилятора персонального компьютера;
- вращение двигателя постоянного тока стенда;
- вращение энкодера.

Уровни шума, издаваемые этими источниками шума, ниже 50 децибел, что намного ниже указанного максимально допустимого уровня шума.

6.3.2 Отсутствие или недостаток необходимого естественного освещения

Достаточное необходимое естественное освещение на рабочем месте является одним из необходимых условий труда, а недостаточное освещение

влияет на качество работы. Освещение в основном влияет на зрение людей, а работа при слабом освещении может привести к утомлению, перенапряжению глаз, головным болям, что может привести к снижению общей работоспособности и производительности труда. В рабочих зонах, предназначенных для использования с ПК, система общего освещения должна освещать рабочую поверхность не менее 300 лк.

Способы регулирования освещения на рабочих местах и в производственных помещениях:

- носить защитные очки;
- установить светозащитные устройства;
- установить осветительные приборы и т.д.

Для снижения влияния этого фактора на здоровье и соблюдения требований по освещению необходимо работать в рабочей зоне с достаточным естественным освещением и осветительными приборами. Кроме того, необходимо делать перерывы в работе.

Рабочая зона, где разрабатывался проект, имеет окна, через которые в рабочую зону попадает естественный свет. При этом в рабочей зоне располагаются осветительные приборы типа настольных ламп. Рабочая зона соответствует требованиям.

6.3.3 Отклонение показателей микроклимата в рабочей зоне

В рабочей зоне на изменение микроклимата будет влиять количество оборудования в рабочем состоянии, вентиляция, температура за окном и т.д. Состояние внутренней среды на рабочем месте напрямую влияет на здоровье организма человека, а комфортная микроклиматическая среда позволяет повысить эффективность труда.

По тяжести ручного труда работа инженеров стенда «Робот-бабочка» относится к категории 1а, то есть к легкой работе. В таблице 23 приведены допустимые значения показателей микроклимата.

Таблица 23 – Допустимые значения показателей микроклимата на рабочем месте

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
	Диапазон ниже оптимальных величин	Диапазон выше оптимальных величин			Диапазон ниже оптимальных величин	Диапазон выше оптимальных величин
Холодный	20,0 – 21,9	24,1 – 25,0	19-26	15-75	0,1	0,1
Теплый	21,0 – 22,9	25,1 – 26,0	20-29	15-75	0,1	0,2

Изменить микроклимат рабочего места помогают следующие устройства и методы:

- открыть окна для проветривания;
- установка системы отопления помещения;
- установка кондиционера.

Кроме того, время пребывания на рабочем месте должно быть ограничено.

6.3.4 Производственные факторы, связанные с напряженностью электрического поля

В этом исследовательском проекте работа персонального компьютера и кабины робота-бабочки увеличивает силу окружающего электрического поля и вызывает деионизацию окружающего воздушного пространства.

Повышенная напряженность электрического поля, несомненно, вредна для человеческого организма и может привести к таким симптомам, как дегенерация мозга и потеря памяти.

Согласно «СанПиН 2.2.2/2.4.1340-03 Гигиенические требования к персональным электронно-вычислительным машинам и организации работ» допускается нахождение в течение рабочего дня в электрическом поле напряженностью не более 5 кВ/м [22]. Необходимо время от времени открывать окна, чтобы проветрить комнату, и делать перерыв.

6.3.5 Производственные факторы, связанные с электрическим током

Поражение электрическим током является чрезвычайно опасным производственным фактором, поскольку ток не виден человеческому глазу. Преходящие эффекты воздействия относительно высокой интенсивности, вызванные прохождением через организм электрического тока, часто приводят к летальному исходу [23].

Во избежание поражения электрическим током рабочее место должно быть защищено и заземлено в соответствии с техническими требованиями эксплуатации. Кроме того, согласно «ГОСТ Р 12.1.019-2017 Система стандартов безопасности труда. Электробезопасность. Общие требования и номенклатура видов защиты» все, что питается от промышленной сети, должно сопровождаться предупредительными знаками, чтобы не совершать неправильных действий и движений персонала. Перед началом работы со стендом «Робот-бабочка» необходимо ознакомиться с инструкцией по эксплуатации стенда и изучить правила электробезопасности, описанные в Требованиях электробезопасности [24].

6.3.6 Производственные факторы, связанные со статическим электричеством

Статическое электричество – это совокупность явлений, связанных с возникновением, сохранением и релаксацией свободного электрического заряда на поверхности или в объёме диэлектриков или на изолированных проводниках. Статическое электричество может возникнуть, когда тело человека соприкасается с устройствами, работающими от электрического тока (например, персональные компьютеры, стенд «Робота-бабочка»). Статическое электричество может привести к травмам и пожару.

6.4 Экологическая безопасность

В этом разделе необходимо рассмотреть характер воздействия на окружающую среду при использовании стенда «Робот-бабочка».

В ходе разработки проекта, когда батарея ПК села, его нужно было зарядить. Во время зарядки зарядное устройство потребляет электроэнергию, что представляет опасность для атмосферы. Для работы стенда «Робот-бабочка» также должна быть подключена к источнику питания.

Одним из процессов в проекте является сварка, которая может привести к загрязнению воздуха. При сварке необходимо измерять уровень загрязнения воздуха для соблюдения нормативного документа [25].

Воздействие на литосферу в основном от промышленных отходов. Во время производства может произойти повреждение оборудования или персональных компьютеров. После завершения работ необработанные отходы должны быть переработаны и подвергнуты предварительной обработке, чтобы уменьшить загрязнение почвы [26].

Воздействие на селитебную зону оказывает шум двигателя, работающего на стенде. Согласно нормативному документу *СанПиН 2.2.1/2.1.1.1200-03* необходимо построить надлежащее рабочее пространство, чтобы уменьшить воздействие на жителей [28].

Воздействие на гидросферу пренебрежимо мало.

6.5 Безопасность в чрезвычайных ситуациях

Во время разработки стенда «Робот-бабочка» возможной чрезвычайной ситуацией стал пожар. Короткое замыкание в цепи или неправильная эксплуатация инженером может привести к пожару. Учитывая, что при разработке проекта необходимо паять множество электронных компонентов, существует вероятность возгорания в процессе пайки.

При возникновении пожара сохраняйте спокойствие и примите следующие меры:

- немедленно выключить питание;
- вызвать по телефону пожарную команду;
- покинуть помещение необходимо согласно плану эвакуации.

На кабинете 204 десятого корпуса ТПУ установлены датчики дыма. При возникновении пожара срабатывают датчики дыма, которые посылают предупреждающий сигнал на все здание [27].

6.6 Вывод по разделу «Социальная ответственность»

В данном разделе были изучены и исследованы государственные стандарты и нормы. Также было выявлено источники вредных и опасных факторов.

На этой основе выявляются и анализируются вредные и опасные факторы. Перечислены актуальные действенные меры помощи инженерам в рабочей зоне от выявленных вредных и опасных факторов.

Наконец, в этом разделе изучаются правовые и организационные аспекты обеспечения безопасности и организационные мероприятия при планировке рабочих мест.

Заключение

В ходе выполнения основной части работы был произведен литературный обзор, после которого было проанализировано основное оборудование стенда «Робот-бабочка» и предложен план модернизации стенда.

Вся работа по модернизации стенда «Робот-бабочка» разделена на две основные части. Первая часть – это аппаратная часть. В аппаратной части для замены микроконтроллера Beagle Black на тестовом стенде был выбран микроконтроллер STM32. В то же время был выбран персональный компьютер с системой Windows для замены исходного микрокомпьютера с системой Linux. Вторая часть – исследование алгоритмов управления стендом «Робот-бабочка». В этой части для получения информации о положении мяча выбирается алгоритм обнаружения круга Хафа, между Python и STM32 формируется поток данных, а для управления вращающимся двигателем используется двухконтурный ПИД-регулятор положения и скорости. При этом светодиод на плате STM32 используется для обратной связи, чтобы удобнее было наблюдать за вращением мотора. После запуска программы микроконтроллер STM32 может управлять вращением двигателя в соответствии с информацией о положении шарика, чтобы шарик устойчиво стоял на панели.

«Робот-бабочка» может реализовать разработку высокоточных алгоритмов сенсорных операций в различных областях и предоставляет платформу для обучения алгоритмам для студентов и технических специалистов в области медицины, авиации и других областях.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. «Робот Бабочка» взлетел на мировой уровень [электронный ресурс] / Агентство стратегических инициатив: <https://asi.ru/news/59081/>
2. Maksim Surov, Anton Shiriaev, Leonid Freidovich, Sergei Gusev, Leonid Paramonov Case study in non-prehensile manipulation: planning and orbital stabilization of one-directional rollings for the “Butterfly” robot //Proceedings - IEEE International Conference on Robotics and Automation 2015 [электронный ресурс]: <https://ieeexplore.ieee.org/document/7139385>
3. Робот «Бабочка»: как научить машину чувствовать [электронный ресурс] / Итмо News: <https://news.itmo.ru/ru/archive/news/5110/>
4. Роман Усатов-Ширяев, Robotikum: о роботе-бабочке, хирурге — и о том, что у них общего [электронный ресурс] / Хайтек: <https://hightech.fm/2020/02/12/robotikum>
5. С ПОМОЩЬЮ РОБОТА «БАБОЧКИ» ПОЛИТЕХНИКИ УЧАТСЯ УПРАВЛЯТЬ «ЧУВСТВАМИ» РАКЕТ, АВТОМОБИЛЕЙ И ТЕХНИКИ [электронный ресурс] / Научная Россия: <https://scientificrussia.ru/articles/s-pomoshchyu-robota-babochki-politehniki-uchatsya-upravlyat-chuvstvami-raket-avtomobilej-i-tehniki>
6. Робот-бабочка поможет заменить вредный ручной труд в российском авиапроме [электронный ресурс] / Тасс наука: <https://nauka.tass.ru/nauka/4010654>
7. Преобразование круга Хафа [электронный ресурс] / OpenCV-Python Tutorials: https://opencv24-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_houghcircles/py_hough_circles.html
8. Преобразование Хафа с помощью OpenCV (C++/Python) [электронный ресурс] / LearnOpenCV: <https://learnopencv.com/hough-transform-with-opencv-c-python/>
9. Модернизация учебно-исследовательской установки «Робот-бабочка» / С.В. Леонов, к.т.н., доцент ОАР, Ян Жун (Китай), ТПУ // XIX

Международной научно-практической конференции студентов, аспирантов и молодых ученых (Томск, 21–25 марта 2022 г.)

10. Что такое энкодер: типы и принцип работы [электронный ресурс] / Инженерная компания 555: <https://ik555.ru/statyi/chto-takoe-encoder/>

11. Назначение и виды энкодеров [электронный ресурс] / Техпривод: <https://tehprivod.su/poleznaya-informatsiya/naznachenie-i-vidy-encoderov.html>

12. Инкрементальный энкодер SCH50F [электронный ресурс] / Willtec: <https://www.scancon.dk/products/incremental/standard-encoders/hollow-shaft/sch50f/>

13. Комплект Discovery с микроконтроллером STM32F303VC [электронный ресурс] / ST: <https://www.st.com/en/evaluation-tools/stm32f3discovery.html>

14. STM32 UART. Прием и передача данных по UART в STM32CubeMx [электронный ресурс] / MicroTechnics: <https://microtechnics.ru/stm32-uart-priem-i-peredacha-dannyh-po-uart-v-stm32cubemx/>

15. UART (USART) на STM32L (STM32) [электронный ресурс] / EasyElectronics: <http://we.easyelectronics.ru/STM32/uart-usart-na-stm32l-stm32.html>

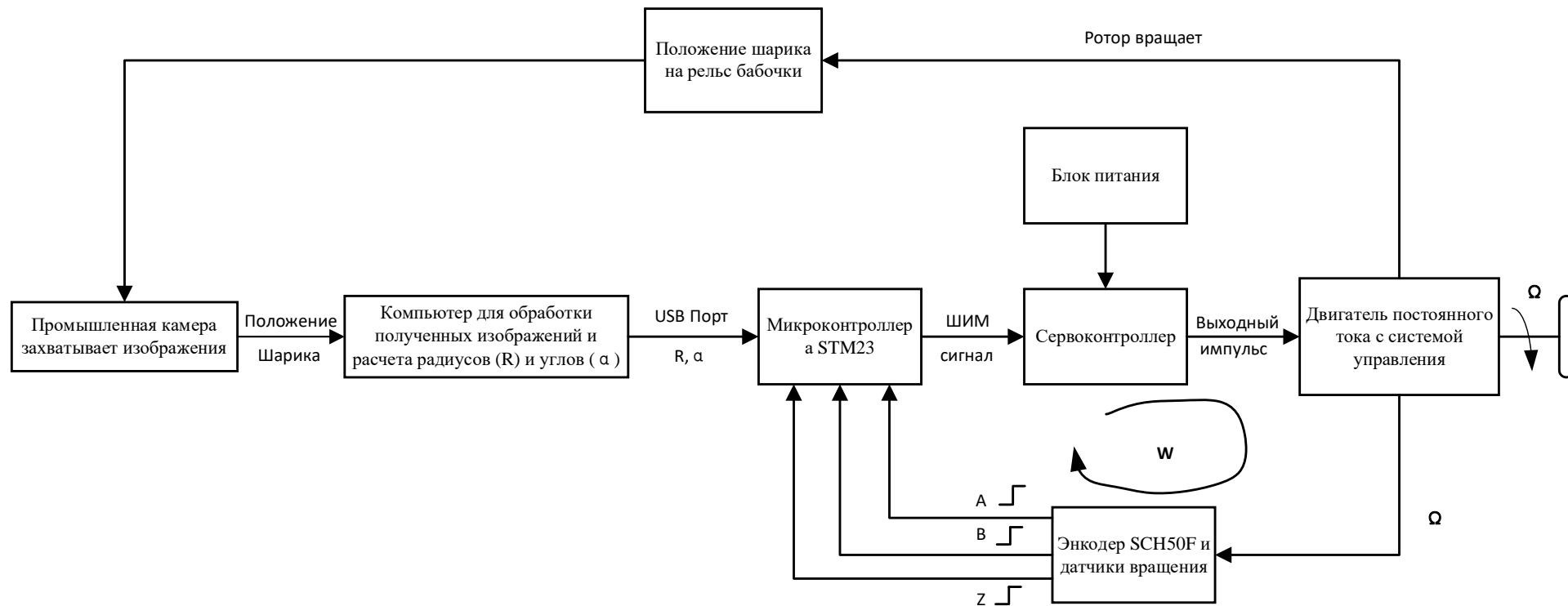
16. Проект ПИД-регулятора с микроконтроллером STM32 с использованием балансировочного стола для шариков [электронный ресурс] / ACROME: <https://acrome.net/post/pid-controller-design-for-stm32-microcontrollers>

17. Анализ процесса управления контуром положения щеточного двигателя постоянного тока и двойным контуром управления скоростью положения (позиционный ПИД-регулятор) [электронный ресурс] / CSDN: <https://blog.csdn.net/bugeilunajiusong/article/details/125923912>

18. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение: учебно-методическое пособие / И.Г. Видяев, Г.Н. Серикова, Н.А. Гаврикова, Н.В. Шаповалова, Л.Р. Тухватулина, З.В. Криницына; Томский политехнический университет. – Томск: изд-во ТПУ, 2014. – 36с. 18. Районный коэффициент.

19. Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 05.02.2018).
20. *ГОСТ 12.2.032-78* Рабочее место при выполнении работ сидя. (Дата обращения 30.05.2023)
21. *ГОСТ 12.0.003-2015 ССБТ*. Опасные и вредные производственные факторы. Классификация. (Дата обращения 30.05.2023)
22. *СанПиН 2.2.2/2.4.1340-03* Гигиенические требования к персональным электронно-вычислительным машинам и организации работ. (Дата обращения 30.05.2023)
23. *ГОСТ 12.0.003-2015. ССБТ*. Опасные и вредные производственные факторы. Классификация. (Дата обращения 30.05.2023)
24. *ГОСТ Р 12.1.019-2017* Система стандартов безопасности труда. Электробезопасность. Общие требования и номенклатура видов защиты. (Дата обращения 30.05.2023)
25. *ГОСТ 17.2.3.01-86* Охрана природы (ССОП). Атмосфера. Правила контроля качества воздуха населенных пунктов: дата введения 1987-01-01. (Дата обращения 30.05.2023)
26. *ГОСТ 17.4.3.04-85* Охрана природы (ССОП). Почвы. Общие требования к контролю и охране от загрязнения: дата введения 1986-07-01. (Дата обращения 30.05.2023)
27. *ГОСТ 12.1.004-91* «ССБТ Пожарная безопасность» :дата введения 2014-06-19. (Дата обращения 30.05.2023)

Приложение А
(обязательное)
Структурная схема система



Приложение Б

(рекомендуемое)

Программный код на стороне Python

Листинг 1 – Программный код на стороне Python

```
1 from pypylon import pylon
2 import cv2 as cv
3 import numpy as np
4 import serial
5 import math
6 import struct
7
8 ser = serial.Serial(port="com5", baudrate=115200)
9
10 camera = pylon.InstantCamera(pylon.TlFactory.GetInstance().CreateFirstDevice())
11
12 camera.StartGrabbing(pylon.GrabStrategy_LatestImageOnly)
13 converter = pylon.ImageFormatConverter()
14
15 converter.OutputPixelFormat = pylon.PixelType_BGR8packed
16 converter.OutputBitAlignment = pylon.OutputBitAlignment_MsbAligned
17
18 while camera.IsGrabbing():
19     grabResult = camera.RetrieveResult(5000, pylon.
    TimeoutHandling_ThrowException)
20
21     if grabResult.GrabSucceeded():
22         image = converter.Convert(grabResult)
23         img = image.GetArray()
24         grey_img = cv.cvtColor(img, cv.COLOR_BGRA2GRAY)
25         mimg = cv.medianBlur(grey_img, 7)
26
27         #img = cv.GaussianBlur(grey_img, (9, 9), 0)
28
29         canny = cv.Canny(mimg, 50, 100)
30         circles = cv.HoughCircles(canny, cv.HOUGH_GRADIENT, 1, 30, param1=100,
    param2=25, minRadius=70, maxRadius=80)
31         # x = circles.astype('int')
32         x = np.array(circles)
33
34         if x.all(None):
35             circles = np.uint16(np.around(circles))
36
37             print(str(circles))
38
39             numcircle = 0
40             for i in circles[0, :]:
41                 cv.circle(img, (i[0], i[1]), i[2], (255, 255, 0), 2)
42                 cv.circle(img, (i[0], i[1]), 2, (0, 0, 255), 3)
43                 dx1 = i[0] - 640
44                 dy1 = 1080 - i[1]
45                 cv.imshow("vedio", img)
46                 size = img.shape
47                 h = size[0]
48                 w = size[1]
49                 #print(h)
50                 #print(w)
51                 if cv.waitKey(1) == 27:
```

Продолжение листинга 1 – Программный код на стороне Python

```
52     break
53
54     #angle1=math.atan2(dx1,dy1)/(math.pi/180)
55     angle1 = math.atan2(dx1, dy1)
56     angle1_r = round(angle1, 3)
57     #angle_en = struct.pack('<f', angle1)
58     print(angle1_r)
59     dsquare= pow(dx1,2)+pow(dy1,2)
60     R=math.sqrt(dsquare)
61     R_r = round(R, 3)
62     angle_en = struct.pack('<f', angle1_r)
63     R_en = struct.pack('<f', R_r)
64     print(R_r)
65     a1 = b'\xfb'
66     a2 = b'\xbf'
67     if abs(angle1) > 0.0:
68         ser.write(a1 + a2 + angle_en + R_en)
69         #tded.write(str(angle1).encode("gbk"))
70     else:
71         cv.imshow("vedio", img)
72         # cv.waitKey(0)
73         if cv.waitKey(1) == 27:
74             break
75
76
77     # cv.namedWindow('title', cv.WINDOW_NORMAL)
78     # cv.imshow('title', canny)
79     #
80     # k = cv.waitKey(1)
81     # if k == 27:
82     #     break
83     grabResult.Release()
84
85     camera.StopGrabbing()
86
87     cv.destroyAllWindows()
```

Приложение В

(рекомендуемое)

Код основного файла App.c на стороне STM32

Листинг 2 – Код основного файла App.c на стороне STM32

```

1  #include "App.h"
2
3  pid_t Motor_Speed;
4  pid_t Motor_Position;
5
6  float Speed_Max=150.0f;
7
8  #define PI 3.1415926f
9
10 uint8_t Rx_Buff[10];
11 uint8_t Rx_Data[10];
12
13 float Data[2]={0.0f,0.0f};
14
15 static void Filter_Data(uint8_t _header,uint8_t _id,uint8_t* _input,uint8_t* _output,uint8_t _size)
16 {
17     if(_input[0]==_header&&_input[1]==_id)
18     {
19         memcpy(_output,_input,_size);
20     }
21     else
22     {
23         if(_input[_size-1]==_header&&_input[0]==_id)
24         {
25             _output[_size-1]=_input[0];
26             memcpy(_output,_input+1,_size-1);
27         } else
28         {
29             uint8_t i=1;
30             for ( i = 1; i < _size-1; ++i)
31             {
32                 if(_input[i]==_header&&_input[i+1]==_id)
33                 {
34                     memcpy(_output,_input+i,_size-i);
35                     memcpy(_output+_size-i,_input,i);
36                     break;
37                 }
38             }
39         }
40     }
41 }
42
43 void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart)
44 {
45     if(huart->Instance == USART1) //
46     {
47         Filter_Data(0XFB,0XBF,Rx_Buff,Rx_Data,10);
48         memcpy(Data,&Rx_Data[2],8);
49     }
50 }
51
52
53
54 void System_Init()
55 {
56
57     HAL_UART_Receive_DMA(&huart1,Rx_Buff,10);
58
59     HAL_TIM_PWM_Start(&htim4,TIM_CHANNEL_1);
60     HAL_TIM_PWM_Start(&htim4,TIM_CHANNEL_2);
61
62     HAL_TIM_Encoder_Start(&htim3, TIM_CHANNEL_ALL);
63     TIM3->CNT=Encoder_initial_value; //
64
65     HAL_TIM_Base_Start_IT(&htim6);
66     PID_struct_init(&Motor_Speed,POSITION_PID,1000.0f,200.0f,7.0f,0.007f,0.01f,10000.0f,0.01f);
67     PID_struct_init(&Motor_Position,POSITION_PID,Speed_Max,110.0f,5.0f,0.00001f,0.00001f,1000.0f,0.01f);
68
69 }
70
71
72 int Round_Cnt=0;
73 int64_t Position_Sum=0;
74 int64_t Position_Sum_Last=0;
75 int Position_Now=0;
76 int Position_Last=Encoder_initial_value;
77 float Speed=0.0f;
78

```


Продолжение листинга 2 – Код основного файла App.c на стороне STM32

```

79  float Position_radian=0.0f;
80
81
82  uint64_t Time_Last=0;
83  uint64_t Time=0;
84
85  void Get_Motor_Position_Speed()
86  {
87
88      Position_Sum_Last=Position_Sum;
89      Position_Now=(int) (TIM3->CNT);
90      if(Position_Now-Position_Last>40000)
91      {
92          Round_Cnt--;
93      }else if(Position_Now-Position_Last<-40000)
94      {
95          Round_Cnt++;
96      }
97      Position_Sum=Round_Cnt*65535+(Position_Now-Encoder_initial_value);
98      Position_Last=Position_Now;
99
100     Speed=(Position_Sum-Position_Sum_Last)*60000/16384;
101     Position_radian=(float) Position_Sum*2*PI/16384.0f;
102
103 }
104
105
106 float Speed_E=0.0f;
107 float Speed_W=0.0f;
108
109 float PWM=0.0f;
110
111 float Position_E= 0.0f;
112
113
114 void Motor_Speed_Control()
115 {
116
117     static uint8_t time=0;
118     Get_Motor_Position_Speed();
119
120
121     if(time<5)
122     {
123         PWM=pid_calc (&Motor_Speed, Speed, Speed_E+Speed_W);
124     }else
125     {
126         Motor_Position.MaxOutput=Speed_Max;
127
128         Speed_E=pid_calc (&Motor_Position, Position_radian, Position_E);
129         time=0;
130     }
131     time++;
132
133     if(PWM>0.01f)
134     {
135         TIM4->CCR1=(uint32_t) PWM;
136         TIM4->CCR2=0;
137     }
138     else if(PWM<-0.01f)
139     {
140         TIM4->CCR2=(uint32_t) (-PWM);
141         TIM4->CCR1=0;
142     }
143     else
144     {
145         TIM4->CCR1=TIM4->ARR;
146         TIM4->CCR2=TIM4->ARR;
147     }
148 }
149
150
151 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
152 {
153     if(htim->Instance==TIM6)
154     {
155         Motor_Speed_Control();
156         LED_Step_clockwise((Position_Sum%16384)/2048);

```

Продолжение листинга 2 – Код основного файла App.c на стороне STM32

```
157
158     Position_E= - (Data[0]);
159 }
160
161 }
162
163
164 void LED_Step_clockwise(uint8_t step)
165 {
166     switch(step)
167     {
168     case 7:
169     {
170         HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_SET);
171         HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_RESET);
172         HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_RESET);
173         HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_RESET);
174         HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_RESET);
175         HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_RESET);
176         HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_RESET);
177         HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_RESET);
178     }
179     break;
180     case 0:
181     {
182         HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_SET);
183         HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_RESET);
184         HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_RESET);
185         HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_RESET);
186         HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_RESET);
187         HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_RESET);
188         HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_RESET);
189         HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_RESET);
190     }
191     break;
192     case 1:
193     {
194         HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_SET);
195         HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_RESET);
196         HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_RESET);
197         HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_RESET);
198         HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_RESET);
199         HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_RESET);
200         HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_RESET);
201         HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_RESET);
202     }
203     break;
204     case 2:
205     {
206         HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_SET);
207         HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_RESET);
208         HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_RESET);
209         HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_RESET);
210         HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_RESET);
211         HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_RESET);
212         HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_RESET);
213         HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_RESET);
214     }
215     break;
216     case 3:
217     {
218         HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_SET);
219         HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_RESET);
220         HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_RESET);
221         HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_RESET);
222         HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_RESET);
223         HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_RESET);
224         HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_RESET);
225         HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_RESET);
226     }
227     break;
228     case 4:
229     {
230         HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_SET);
231         HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_RESET);
232         HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_RESET);
233         HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_RESET);
234         HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_RESET);
```

Продолжение листинга 2 – Код основного файла App.c на стороне STM32

```
235     HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_RESET);
236     HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_RESET);
237     HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_RESET);
238 }
239 break;
240 case 5:
241 {
242     HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_SET);
243     HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_RESET);
244     HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_RESET);
245     HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_RESET);
246     HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_RESET);
247     HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_RESET);
248     HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_RESET);
249     HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_RESET);
250 }
251 break;
252 case 6:
253 {
254     HAL_GPIO_WritePin(GPIOE,LED8_Pin,GPIO_PIN_SET);
255     HAL_GPIO_WritePin(GPIOE,LED1_Pin,GPIO_PIN_RESET);
256     HAL_GPIO_WritePin(GPIOE,LED2_Pin,GPIO_PIN_RESET);
257     HAL_GPIO_WritePin(GPIOE,LED3_Pin,GPIO_PIN_RESET);
258     HAL_GPIO_WritePin(GPIOE,LED4_Pin,GPIO_PIN_RESET);
259     HAL_GPIO_WritePin(GPIOE,LED5_Pin,GPIO_PIN_RESET);
260     HAL_GPIO_WritePin(GPIOE,LED6_Pin,GPIO_PIN_RESET);
261     HAL_GPIO_WritePin(GPIOE,LED7_Pin,GPIO_PIN_RESET);
262 }
263
264 }
265 }
266
```

Приложение Г

(рекомендуемое)

Код основного файла App1.c на стороне STM32

Листинг 3 – Код основного файла App1.c на стороне STM32

```

1  #include "App.h"
2
3  pid_t Motor;
4
5  uint8_t Rx_Buff[10];
6  uint8_t Rx_Data[10];
7
8  float Data[2]={0.0f,0.0f};
9
10 static void Filter_Data(uint8_t _header,uint8_t _id,uint8_t* _input,uint8_t* _output,uint8_t _size)
11 {
12     if(_input[0]==_header&&_input[1]==_id)
13     {
14         memcpy(_output,_input,_size);
15     }
16     else
17     {
18         if(_input[_size-1]==_header&&_input[0]==_id)
19         {
20             _output[_size-1]=_input[0];
21             memcpy(_output,_input+1,_size-1);
22         } else
23         {
24             uint8_t i=1;
25             for ( i = 1; i < _size-1; ++i)
26             {
27                 if(_input[i]==_header&&_input[i+1]==_id)
28                 {
29                     memcpy(_output,_input+i,_size-i);
30                     memcpy(_output+_size-i,_input,i);
31                     break;
32                 }
33             }
34         }
35     }
36 }
37
38 void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart)
39 {
40     if(huart->Instance == USART1) //
41     {
42         Filter_Data(0XFB,0XBF,Rx_Buff,Rx_Data,10);
43         memcpy(Data,&Rx_Data[2],8);
44     }
45 }
46
47
48
49 void System_Init()
50 {
51
52     HAL_UART_Receive_DMA(&huart1,Rx_Buff,10);
53
54     HAL_TIM_PWM_Start(&htim4,TIM_CHANNEL_1);
55     HAL_TIM_PWM_Start(&htim4,TIM_CHANNEL_2);
56
57     HAL_TIM_Encoder_Start(&htim3, TIM_CHANNEL_ALL);
58     TIM3->CNT=Encoder_initial_value;
59
60     HAL_TIM_Base_Start_IT(&htim6);
61     PID_struct_init(&Motor,POSITION_PID,1000.0f,200.0f,2.2f,0.2f,0.01f,10000.0f,0.01f);
62
63 }
64
65
66 __STATIC_INLINE uint32_t LL_SYSTICK_IsActiveCounterFlag()
67 {
68     return ((SysTick->CTRL & SysTick_CTRL_COUNTFLAG_Msk) == (SysTick_CTRL_COUNTFLAG_Msk));
69     //SysTick->CTRL
70     //SysTick_CTRL_COUNTFLAG_Msk
71 }
72 uint64_t micros()
73 {
74     /* Ensure COUNTFLAG is reset by reading SysTick control and status register */
75     LL_SYSTICK_IsActiveCounterFlag();
76     uint32_t m = HAL_GetTick();
77     uint32_t tms = SysTick->LOAD + 1;
78     __IO uint32_t u = tms - SysTick->VAL;

```

Продолжение листинга 3 – Код основного файла App1.c на стороне STM32

```

79     if (LL_SYSTICK_IsActiveCounterFlag())
80     {
81         m = HAL_GetTick();
82         u = tms - SysTick->VAL;
83     }
84     return (m * 1000 + (u * 1000) / tms);
85 }
86
87
88 int Round_Cnt=0;
89 int64_t Position_Sum=0;
90 int64_t Position_Sum_Last=0;
91 int Position_Now=0;
92 int Position_Last=Encoder_initial_value;
93 float Speed=0.0f;
94
95
96 uint64_t Time_Last=0;
97 uint64_t Time=0;
98 int flag = 0;
99
100
101 void Get_Motor_Position_Speed()
102 {
103     Time_Last=Time;
104     Time=micros();
105     Position_Sum_Last=Position_Sum;
106     Position_Now=(int) (TIM3->CNT);
107     if(Position_Now-Position_Last>40000)
108     {
109         Round_Cnt--;
110     }else if(Position_Now-Position_Last<-40000)
111     {
112         Round_Cnt++;
113     }
114     Position_Sum=Round_Cnt*65535+(Position_Now-Encoder_initial_value)
115     Position_Last=Position_Now;
116     if(Time-Time_Last>=1)
117     {
118         Speed=(Position_Sum-Position_Sum_Last)*60000/(8192*2);
119     }
120 }
121
122
123 float Set_Speed=-35.0f;
124 float PWM=0.0f;
125 void Motor_Speed_Control()
126 {
127     Get_Motor_Position_Speed();
128     PWM=pid_calc(&Motor,Speed,Set_Speed);
129     if(PWM>0.01f)
130     {
131         TIM4->CCR1=(uint32_t)PWM;
132         TIM4->CCR2=0;
133     }else if(PWM<-0.01f)
134     {
135         TIM4->CCR2=(uint32_t) (-PWM);
136         TIM4->CCR1=0;
137     }else
138     {
139         TIM4->CCR1=TIM4->ARR;
140         TIM4->CCR2=TIM4->ARR;
141     }
142 }
143
144 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
145 {
146     if(htim->Instance==TIM6)
147     {
148         Motor_Speed_Control();
149         if (Position_Sum >= 0)
150         {
151             LED_Step_clockwise((Position_Sum%(8192*2))/2048);
152         }
153         else
154         {
155             LED_Step_clockwise(7-(-Position_Sum%(8192*2))/2048);
156         }
157     }
158 }

```

Продолжение листинга 3 – Код основного файла App1.c на стороне STM32

```
157     }
158
159 }
160
161 void LED_Step_clockwise(uint8_t step)
162 {
163     flag = step;
164     switch(step)
165     {
166     case 7:
167     {
168         HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_SET);
169         HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_RESET);
170         HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_RESET);
171         HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_RESET);
172         HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_RESET);
173         HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_RESET);
174         HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_RESET);
175         HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_RESET);
176     }
177     break;
178     case 0:
179     {
180         HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_SET);
181         HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_RESET);
182         HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_RESET);
183         HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_RESET);
184         HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_RESET);
185         HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_RESET);
186         HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_RESET);
187         HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_RESET);
188     }
189     break;
190     case 1:
191     {
192         HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_SET);
193         HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_RESET);
194         HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_RESET);
195         HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_RESET);
196         HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_RESET);
197         HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_RESET);
198         HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_RESET);
199         HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_RESET);
200     }
201     break;
202     case 2:
203     {
204         HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_SET);
205         HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_RESET);
206         HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_RESET);
207         HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_RESET);
208         HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_RESET);
209         HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_RESET);
210         HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_RESET);
211         HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_RESET);
212     }
213     break;
214     case 3:
215     {
216         HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_SET);
217         HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_RESET);
218         HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_RESET);
219         HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_RESET);
220         HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_RESET);
221         HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_RESET);
222         HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_RESET);
223         HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_RESET);
224     }
225     break;
226     case 4:
227     {
228         HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_SET);
229         HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_RESET);
230         HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_RESET);
231         HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_RESET);
232         HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_RESET);
233         HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_RESET);
234         HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_RESET);
```

Продолжение листинга 3 – Код основного файла App1.c на стороне STM32

```
235     HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_RESET);
236   }
237   break;
238   case 5:
239   {
240     HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_SET);
241     HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_RESET);
242     HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_RESET);
243     HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_RESET);
244     HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_RESET);
245     HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_RESET);
246     HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_RESET);
247     HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_RESET);
248   }
249   break;
250   case 6:
251   {
252     HAL_GPIO_WritePin(GPIOE, LED8_Pin, GPIO_PIN_SET);
253     HAL_GPIO_WritePin(GPIOE, LED1_Pin, GPIO_PIN_RESET);
254     HAL_GPIO_WritePin(GPIOE, LED2_Pin, GPIO_PIN_RESET);
255     HAL_GPIO_WritePin(GPIOE, LED3_Pin, GPIO_PIN_RESET);
256     HAL_GPIO_WritePin(GPIOE, LED4_Pin, GPIO_PIN_RESET);
257     HAL_GPIO_WritePin(GPIOE, LED5_Pin, GPIO_PIN_RESET);
258     HAL_GPIO_WritePin(GPIOE, LED6_Pin, GPIO_PIN_RESET);
259     HAL_GPIO_WritePin(GPIOE, LED7_Pin, GPIO_PIN_RESET);
260   }
261 }
262 }
263 }
```


Приложение Д

(обязательное)

Конфигурация параметров STM32CubeMX

Pinout & Configuration
Clock Configuration
Project Manager
Tools

Software Packs
Pinout

Categories A-Z

System Core >

Analog >

Timers >

- RTC
- TIM1
- TIM2
- ✓ TIM3
- ✓ TIM4
- ✓ TIM6
- TIM7
- TIM8
- TIM15
- TIM16
- TIM17

Connectivity >

Multimedia >

Computing >

Middleware >

TIM3 Mode and Configuration

Mode

Slave Mode Disable

Trigger Source Disable

Clock Source Disable

Channel1 Disable

Channel2 Disable

Channel3 Disable

Channel4 Disable

Combined Channels Encoder Mode

ETR IO as Clearing Source

XOR activation

One Pulse Mode

Configuration

Reset Configuration

✓ NVIC Settings
✓ DMA Settings
✓ GPIO Settings

✓ Parameter Settings
✓ User Constants

Search Signals

Pin Na...	Signal on ...	GPIO outp...	GPIO mode	GPIO Pull...	Maximum ...	Fast Mode
PE2	TIM3_CH1	n/a	Alternate ...	No pull-up ...	Low	n/a
PE3	TIM3_CH2	n/a	Alternate ...	No pull-up ...	Low	n/a

Pinout view
System view

STM

STM32F303VCTx

LQFP100

Pinout & Configuration

Clock Cor

▼ Softw

Categories A->Z

- System Core >
- Analog >
- Timers >
 - RTC
 - TIM1
 - TIM2
 - ✓ TIM3
 - ✓ **TIM4**
 - ✓ TIM6
 - TIM7
 - TIM8
 - TIM15
 - TIM16
 - TIM17
- Connectivity >
- Multimedia >
- Computing >
- Middleware >

TIM4 Mode and Configuration

Mode

Slave Mode

Trigger Source

Clock Source

Channel1

Channel2

Channel3

Channel4

Combined Channels

ETR IO as Clearing Source

XOR activation

One Pulse Mode

Configuration

✓ DMA Settings ✓ GPIO Settings

✓ Parameter Settings ✓ User Constants ✓ NVIC Settings

Search Signals

Pin Na...	Signal on ...	GPIO outp...	GPIO mode	GPIO Pull...	Maxi
PD12	TIM4_CH1	n/a	Alternate ...	No pull-up ...	Low
PD13	TIM4_CH2	n/a	Alternate ...	No pull-up ...	Low

Pinout & Configuration

Clock Cor

▼ Softw

Categories A->Z

- System Core >
- Analog >
- Timers >
 - RTC
 - TIM1
 - TIM2
 - ✓ TIM3
 - ✓ TIM4
 - ✓ **TIM6**
 - TIM7
 - TIM8
 - TIM15
 - TIM16
 - TIM17
- Connectivity >
- Multimedia >
- Computing >
- Middleware >

TIM6 Mode and Configuration

Mode

Activated

One Pulse Mode

Configuration

✓ NVIC Settings ✓ DMA Settings

✓ Parameter Settings ✓ User Constants

Configure the below parameters :

▼ Counter Settings

Prescaler (PSC - 16 bit... 64-1

Counter Mode Up

Counter Period (AutoRe... 1000-1

auto-reload preload Disable

▼ Trigger Output (TRGO) Parame...

Trigger Event Selection Reset (UG bit from TIMx_EGR)

Приложение Е

(обязательное)

Конфигурация параметров Escon Studio

ESCON Studio 2.2 - [Controller Monitor]

File View Tools Window Help

Active Controller ESCON 70/10 - USB0

Tools

- Configuration
 - Startup Wizard
 - Regulation Tuning
 - Firmware Update
 - Diagnostics
- Operation
 - Controller Monitor
 - Parameters
 - Data Recorder

Dynamic Help

- Contents
- Startup Wizard
 - Motor / Sensors
 - Controller
 - Inputs / Outputs

Controller Monitor

ESCON Enabled

The diagram illustrates the control loop for a maxon DC motor. It starts with an input signal A11 (-0.002 V) entering the Set Value block (0.0 rpm). This signal passes through a Speed Ramp block (0.0 rpm/s) and a Current Limit block (30.0000 A) into the Speed Controller (Open Loop) block (0.0 rpm). The Speed Controller's output goes to the Power Stage block, which is also influenced by an IxR Factor (500) and an Enable signal (High) from DI2. The Power Stage block includes status indicators for Output voltage saturated, Amplifier Limitation, and Motor I2t Limitation. The final output is connected to a maxon DC motor, with a feedback current of 0.1762 A shown.

Overview Controller Properties

Status

0 Errors 0 Warnings Hide

Type	Name	Description	Solution
------	------	-------------	----------

Ready Standard Controller Online