МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМ. Н.Э. БАУМАНА

ОЛИМПИАДЫ ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ» МГТУ ИМ. Н.Э. БАУМАНА

КОНФЕРЕНЦИЯ «ШАГ В БУДУЩЕЕ.»

XXIX НАУЧНО-ОБРАЗОВАТЕЛЬНОЕ СОРЕВНОВАНИЕ МОЛОДЫХ ИССЛЕДОВАТЕЛЕЙ «ШАГ В БУДУЩЕЕ»

СБОРНИК ЛУЧШИХ РАБОТ

МОСКВА, 2026

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ 3
Баулин С.А. РАЗРАБОТКА МОБИЛЬНОЙ ПЛАТФОРМЫ НА OMNI-КОЛЁСАХ С BLDC МОТОРАМИ 7
Кравцов А.В. РАЗРАБОТКА ЛИНЕЙНОГО ЛАБОРАТОРНОГО БЛОКА ПИТАНИЯ 17
Измаилова А.Р. ДЕТСКАЯ КАРТА С РАСШИРЕННЫМ РОДИТЕЛЬСКИМ КОНТРОЛЕМ: ДЛЯ ВАШЕГО СПОКОЙСТВИЯ И ФИНАНСОВОЙ ГРАМОТНОСТИ РЕБЕНКА 33
Лобачев К.С. КОМПЛЕКС СИСТЕМ ДЛЯ ПРОИЗВОДСТВА ИЗДЕЛИЙ ПО ТЕХНОЛОГИИ FFF ИЗ ГРАНУЛ 46
Цветкова В.С. СКРИПТ ДЛЯ УПРАВЛЕНИЯ ПОЗИЦИЕЙ НА КРИПТОВАЛЮТНОЙ БИРЖЕ 93
Кузьмина С.В. ОЦЕНКА КАЧЕСТВА ВОЗДУХА С ПРИМЕНЕНИЕМ ДРОНА 141
Жук И.В. ШИФРОВАНИЕ МЕТОДОМ ГАММИРОВАНИЯ 164
Несмеянов А.В. JARVIS 1 - ПЕРСОНАЛЬНЫЙ АССИСТЕНТ ОБЩЕГО НАЗНАЧЕНИЯ 198
Семин А.В. РАЗРАБОТКА СИСТЕМЫ ПОСАДКИ КОСМИЧЕСКОГО АППАРАТА НА ПЛАНЕТУ МАРС 234
Качка М.Д. ПРОТОТИПИРОВАНИЕ ПРИБОРА ДЛЯ УДАЛЕННОГО УПРАВЛЕНИЯ ДАЧНЫМ УЧАСТКОМ С ПОМОЩЬЮ TELEGRAM-БОТА 264
Ганиева Д.Р. ПЕРСПЕКТИВЫ ПРИМЕНЕНИЯ ГИБРИДНЫХ КОМПОЗИЦИОННЫХ МАТЕРИАЛОВ ДЛЯ СОЗДАНИЯ ИНОПЛАНЕТНОЙ ИНФРАСТРУКТУРЫ НА ПОВЕРХНОСТИ ЛУНЫ 306
Егоров А.К. ЭКОЛОГИЧЕСКИЕ ПОКАЗАТЕЛИ ДИЗЕЛЬНОГО ДВИГАТЕЛЯ, РАБОТАЮЩЕГО НА СМЕСЯХ НЕФТЯНОГО ДИЗЕЛЬНОГО ТОПЛИВА И ОЛИВКОВОГО МАСЛА 325
Филимонова В.В. СОЗДАНИЕ И ИССЛЕДОВАНИЕ БИОРАЗЛАГАЕМОГО ДРЕВЕСНО-КРАХМАЛЬНОГО КОМПОЗИТА ДЛЯ УПАКОВКИ 349
Герасименко Г.А. АРХИБОТ MARKUS – АВТОНОМНЫЙ ГУСЕНЕЧНЫЙ РОБОТ ДЛЯ ИССЛЕДОВАНИЯ И 3D-КАРТОГРАФИИ ПЕЩЕР 357
Василенок А.В. РАЗРАБОТКА И СОЗДАНИЕ АВТОНОМНОГО МНОГОФУНКЦИОНАЛЬНОГО НАВИГАЦИОННОГО КОМПЛЕКСА «POLLUX NAVIGATOR» 376
Соловьев А.А. СБОРКА И ИСПЫТАНИЕ СОБСТВЕННОЙ МАГНИТНО-ИМПУЛЬСНОЙ УСТАНОВКИ ДЛЯ ЭЛЕКТРОМАГНИТНОЙ ФОРМОВКИ 410
Малахов А.Д. УСТРОЙСТВО КОНТРОЛЯ ИНФУЗИИ 430
Семенов А.А. ОБНАРУЖЕНИЕ ИЗМЕНЕНИЙ НА РАЗНОВРЕМЕННЫХ СНИМКАХ ГОРОДСКОЙ СРЕДЫ 449
Сырнева Д.Т. СОЗДАНИЕ И ИССЛЕДОВАНИЕ ХРОМАТОГРАФИЧЕСКИХ СВОЙСТВ АФФИННОГО СОРБЕНТА НА ОСНОВЕ АПИКАЛЬНОГО ДОМЕНА БЕЛКА ШАПЕРОНА GROEL 485
Чеглаков В.П. РАЗРАБОТКА МОБИЛЬНОГО УСТРОЙСТВА ДИСТАНЦИОННОЙ ТАКТИЛЬНОЙ ИНДИКАЦИИ ПРОСТРАНСТВА ДЛЯ НЕЗРЯЧИХ (УДТИП) 498

ДОРОГИЕ ДРУЗЬЯ!

В этот сборник вошли лучшие статьи участников Научно-образовательного соревнования молодых исследователей «Шаг в будущее», проходившего с 17 по 28 марта 2025 года в МГТУ им. Н.Э. Баумана. Статьи подготовлены школьниками в рамках конкурса проектных работ и публикуются в авторской редакции.

Основой задачей научно-образовательного соревнования «Шаг в будущее» является выявление и вовлечение в научно-исследовательскую работу, в сферу инженерного творчества учащихся образовательных учреждений, организация сотрудничества исследователей и ученых разных поколений, создание специальных условий в стенах МГТУ им. Н.Э. Баумана для воспитания профессионально-ориентированной, склонной к научной работе молодежи, осуществление комплексного подхода к оценке интеллектуального и творческого потенциала будущего студента. «Шаг в будущее» является уникальным методом профессиональной подготовки, которая осуществляется круглогодично в научно-исследовательских лабораториях школьников, созданных на базе кафедр Университета и ведущих предприятий России, и индивидуально с преподавателями МГТУ им. Н.Э. Баумана, которые выступают в качестве научных руководителей в течение одного, двух и более лет. Результатом такой подготовки становится научно-исследовательская работа, оформленная в соответствии с требованиями, соизмеримыми с требованиями на курсовые работы и отчеты по НИР, и представляемая для публичной защиты на научных секциях конференции, которые проводятся на кафедрах МГТУ им. Н.Э. Баумана при участии ведущих преподавателей, ученых и специалистов. Научными направлениями конференции являются: Машиностроительные технологии; Информатика и системы управления; Радиооптоэлектроника; Биомедицинская техника; Специальное машиностроение, Энергетика и экология, Автоматизация, робототехника и механика; Инженерный бизнес и менеджмент; Фундаментальные науки; Образовательно-реабилитационные технологии; биологические и химические технологии. В рамках ежегодного научно-образовательного соревнования проводится Выставка - конкурс программных разработок, которая за многие годы своего существования стала традиционным смотром достижений школьников, увлеченных захватывающим процессом создания компьютерных программ, и доказала свою состоятельность оригинальностью и новизной программных продуктов, участвовавших в Выставке и использовавшихся в дальнейшем в научных работах, учебном процессе, коммерческой реализации.

Научно-образовательное соревнование «Шаг в будущее» носит не только научный, образовательный и социальный характер, но представляет собой соревнование, на котором выбираются победители в различных номинациях. Всем участникам вручаются свидетельства участника научно-образовательного соревнования.

В МГТУ им. Н.Э. Баумана в рамках работы Центра довузовской подготовки проводятся следующие мероприятия:

- Олимпиады для школьников 8-11 классов, такие как Олимпиада школьников «Шаг в будущее» по шести профилям, олимпиада по программированию для школьников.
- «ТехноКубок» совместно с VK Company и МФТИ.
- Отраслевая олимпиада школьников «Газпром» совместно с ПАО «Газпром» и другими техническими университетами России.
- Экскурсии на предприятия ОПК проводятся совместно с организациями, сотрудничающими с Университетом по целевой подготовке кадров, на ведущие в своей области предприятия страны, с целью раскрытия связи профильного образования с овладением знаний для получения конкретной профессии и будущей трудовой деятельности. Создание системы взаимодействия между школами и предприятиями через учебные заведения высшего профессионального образования состоит в том, чтобы сформировать у молодежи полное представление об их профессиональном будущем, включая выбор будущей специальности, области деятельности, места работы и учебного заведения для получения соответствующего образования. Учащимся предоставляется возможность посетить музеи предприятий, увидеть производственные помещения, побывать на опытных площадках, а также задать все интересующие вопросы, касающиеся деятельности предприятия, условий работы, особенностей поступления и обучения по целевому набору и многое другое.
- Технический лабораторный практикум – школьным инженерным классам организуется в системе университетского образования для учеников 8-11-х классов с целью интеграции методики и технического оборудования университета в учебный процесс общеобразовательных учреждений; расширения и закрепления у обучающихся базовых знаний школьного курса физики, информатики и математики, получения навыков их использования на практике; приобретения опыта в обращении с техническим оборудованием и

навыков проведения экспериментов, а также популяризации научно-технических знаний. Цикл лабораторных работ выполняется на действующих современных образцах техники и технологических машин 16-ти экспериментальных лабораторий и 5-ти Научно-образовательных центров: Центр управления полетами малых космических аппаратов; Молодежный космический центр; лабораторный комплекс Дома Физики; НУЦ «Робототехника», НИЦ «Биомедицинская техника». Лабораторные работы сгруппированы по 5-ти разделам физики (Физика твердого тела, Термодинамика и Молекулярно-кинетическая теория, Электродинамика, Оптика, Механика), информатике и математике.

- Летняя практика: Организация и проведение летней научно-образовательной практики для обучающихся 8-10 классов общеобразовательных организаций города Москвы с инженерными классами на базе научно-образовательных центров, Военного института, лабораторий и кафедр следующих факультетов: Машиностроительные технологии, Специальное машиностроение, Энергомашиностроение, Робототехника и комплексная автоматизация, Фундаментальные науки, Радиоэлектроника и лазерная техника, Информатика и системы управления, Биомедицинская техника, Инженерный бизнес и менеджмент МГТУ им. Н.Э. Баумана. Каждому школьнику предоставляется возможность прохождения практики по индивидуальной образовательной траектории, которую он выбирает из вышеназванных подразделений Университета в соответствии с собственными предпочтениями.
- «Ресурсный центр политехнического образования» начал свою работу в 2018-2019 учебном году для учащихся 8-11 классов общеобразовательных организаций города Москвы с инженерными классами, в рамках которого проходят политехнические кружки по четырем научным направлениям: Занимательная электроника; Колесные машины; Инструментальная техника и технологии; Занимательная робототехника. На основании полученных знаний учащиеся могут выполнять самостоятельную проектную работу и представлять результаты своей деятельности на научных конференциях вузовского и городского уровня с участием ведущих специалистов в области техники и технологий.
- Проект «Университетские субботы» проводится для вовлеченности в технические специальности более широкого круга школьников 8-11 классов и

учащихся колледжей, техникумов, используя потенциал системы высшего профессионального образования. Обеспечивается управляемый доступ школьников к научно-техническим ресурса университета и работе профессиональных научных коллективов. Интеграция основного и дополнительного образования создает благоприятные условия для взаимодействия педагогов разных специальностей, формирования у подростков и старшеклассников целостной картины мира, комплексного педагогического влияния на личность школьника. В проекте проводятся следующие мероприятия: лекции, практические занятия, творческие занятия, мастер-классы по следующим направлениям: техника и технологии, экономика и управление, физико-математические науки, общественные и гуманитарные науки.

Такая система непрерывных профессиональных образовательных технологий «школа-вуз-предприятие» направлена на решение проблем, стоящих перед сегодняшней системой образования:

- устойчивая мотивация к приобретению профессиональных знаний в выбранной области, основанная на практической деятельности под руководством преподавателей вузов;
- ранняя активная и осознанная профессиональная ориентация;
- развитие сопровождающей научно-исследовательской и профессиональной подготовки в общеобразовательных учреждениях;
- разработка новых методик и новых обучающих технологий для развития современного образования.

Организаторы программы и составители данного сборника надеются, что такая форма работы заинтересует учителей и преподавателей учебных заведений Москвы и Московской области и поможет успешному развитию творческих возможностей довузовской молодежи в сфере научно-технической деятельности.

РАЗРАБОТКА МОБИЛЬНОЙ ПЛАТФОРМЫ НА OMNI-КОЛЕСАХ С BLDC МОТОРАМИ

Баулин Степан Алексеевич

Москва г, ГБОУ Школа №121, 11 класс

Секция: Многоцелевые роботы и летательные аппараты

Научный руководитель: Сухоцкий Владимир Андреевич, старший преподаватель, ГБОУ «Воробьёвы горы»

АННОТАЦИЯ

Рынок роботов для складской логистики продолжает развиваться в направлении увеличения времени работы без подзарядки аккумуляторов. Имея опыт конструирования роботов для спортивных соревнований, было решено разработать и протестировать мобильную платформу для складов на omni-колесах с приводом на BLDC бесколлекторных электромоторах. Анализ интернет-ресурсов в направлении исследований или применения приводов omni-колес на BLDC электромоторах не дал результатов.

Конструкция платформы и исследования были проведены исходя из возможности оборудования, имеющегося в лаборатории.

Тестовый вариант платформы разработан в трехколесном варианте.

Протестирована повторяемость при движении платформы по прямой и поворотах, построены графики потребляемой мощности при увеличении нагрузки.

При исследовании основное внимание было уделено определению крутящего момента срыва omni колеса. На основании этого предложен вариант расчёта передаточного отношения редукторов для BLDC электромоторов.

ВВЕДЕНИЕ

Рынок роботов на мобильных платформах для складской логистики, обслуживания массовых мероприятий, логистическое сопровождение в медицинских учреждения, автоматизация рутинных процессов продолжает развиваться в направлении увеличения времени работы робота без подзарядки аккумуляторов. Одно из направлений приводов платформы - использование бесколлекторных электродвигателей. Они широко используются в дронах. Поскольку налажено их массовое производство и драйверов для них, цена комплекта электромотор- драйвер постоянно снижается. Анализ интернет-ресурсов в направлении применения и даже исследований использования бесколлекторных электродвигателей для приводов омни-колес электромоторах не дал результатов. Поэтому в данной работе рассмотрен вопрос применения BLDC электромоторов для автономной мобильной платформы, которую можно дополнять в зависимости от задачи: датчиками препятствий, лидарами, манипуляторами и другим оборудованием.

Цель работы — создать мобильную платформу на омни-колёсах с приводом от BLDC моторов с целью проверки эффективности использования данных моторов в мобильной платформе.

Актуальность работы — в перспективе, полученные результаты помогут разработать более эффективных складских роботов и помочь в разработке мобильных платформ с приводом от BLDC двигателей.

Задачи:

1. Разработать конструкцию платформы в трехколесном варианте для увеличения маневренности.
2. Протестировать повторяемость траектории при движении платформы по прямой линии и поворотах.
3. Измерить потребляемую мощность при движении и сравнить с аналогичной платформой на шаговых двигателях.

ВЫБОР BLDC ЭЛЕКТРОМОТОРОВ

Расчётная масса платформы 10 килограмм. Для стабильного удержания робота на траектории при подборе BLDC моторов необходимо определить момент срыва ролика омни-колеса.

Для этого был изготовлен стенд (рис.1), в который фиксировался ролик омни-колеса и сверху нагружался, с помощью электронных весов измерялось усилие, при котором ролик начнёт проскальзывать.

Рис. 1. Стенд

Был проведен ряд измерений при разной нагрузке (таб. 1). На графике (рис. 2) показана линейная зависимость.

Таблица 1. Сравнительный анализ измерений при разной нагрузке

Сила (в килограммах)	Масса (в килограммах)	Сила (в ньютонах)
0,350	0,605	3,4335
0,365	0,725	3,5807
0,420	0,880	4,1202
0,530	1,240	5,1993
0,590	1,615	5,7879

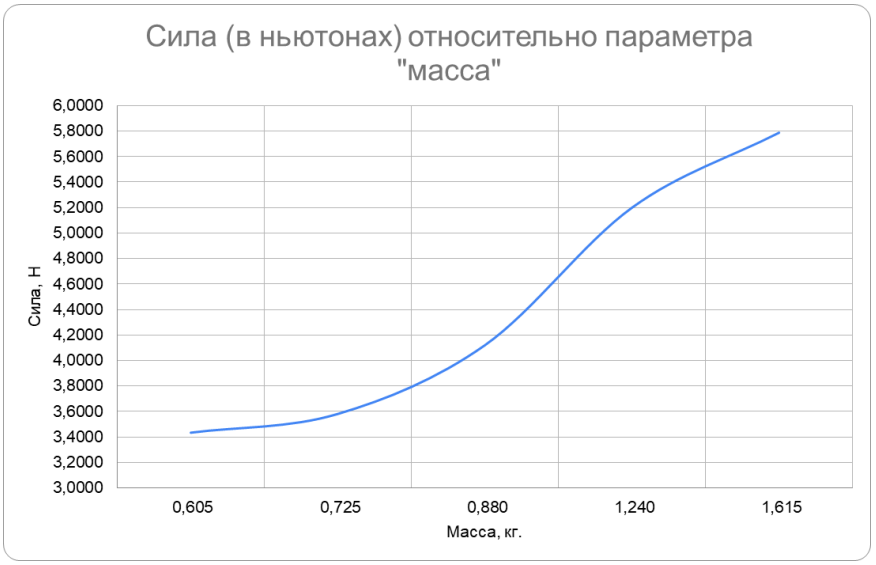


Рис. 2. Зависимость силы от массы

Видим на графике (рис. 2), что зависимость линейна, соответственно, мы можем предположить, что при массе платформы в 10 килограмм необходимая линейная сила будет 26.8 ньютона после пересчёта линейного момента в крутящий при диаметре колеса в 60 мм необходимый крутящий момент для срывая колеса 0.804 ньютон метра. Значит выбранные

моторы должны выдавать такой же или меньший крутящий момент.

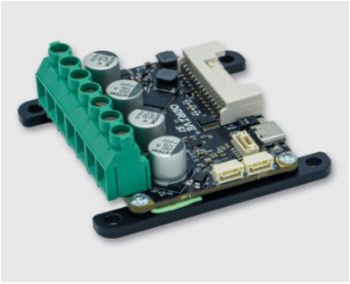
На основе тестов крутящего момента, найденных в интернете, были выбраны моторы тип размера 5010 (рис. 3), которые выдают крутящий момент в 0.2 ньютон метра.

Рис. 3. Мотор тип размера 5010

ЭЛЕКТРОНИКА

При выборе контроллеров BLDC моторов с обратной связью было рассмотрено три варианта изделия с учетом положительных/отрицательных факторов, необходимых для конструирования платформы (см. таб. 2 «Выбор контроллеров»).

Таблица 2. Выбор контроллеров

Odrive S1	Odesc V4.2	Moteus r4.11
		
Плюсы:		
Распространённость Высокая мощность (1600W) Низкий процент брака.	Низкая стоимость.	Есть «Экосистема». Хорошая документация. Встроенный магнитный энкодер. Большое количество, защит
Минусы:		
Высокая стоимость. Отсутствие «экосистемы».	Высокий процент брака. Отсутствие защит. Отсутствие «экосистемы».	Необходимо прошивать, достаточно сложная настройка

В итоге в качестве контроллеров для электромоторов были выбраны Mjbots moteus r4.11 (рис. 4). Одним из главных преимуществ данных контроллеров является “экосистема”, в

которую входит плата распределения питания Mjbots power dist (рис. 5).

Рис. 4. Mjbots moteus r4.11

Рис. 5. Mjbots power dist

При одновременном старте трёх электромоторов требуется запас энергии в конденсаторах суммарной ёмкостью 100 мкф. Чтобы компенсировать помехи для электроники от большого пикового тока, использована плата Mjbots power dist. В результате ток в цепи поднимается плавно за 10 мс и отсутствует резкий скачек.

Для передачи информации драйверы двигателей используют интерфейс CAN-FD, который отсутствует в Raspberry Pi (рис. 6). Поэтому дополнительно использована плата Mjbots Pi3Hat (рис. 7), которая преобразует интерфейс SPI в 5 CAN-FD шин.

Рис. 6. Raspberry Pi

Рис. 7. Mjbots Pi3Hat

Так же выше представлена структурная схема электроники платформы (рис. 8).

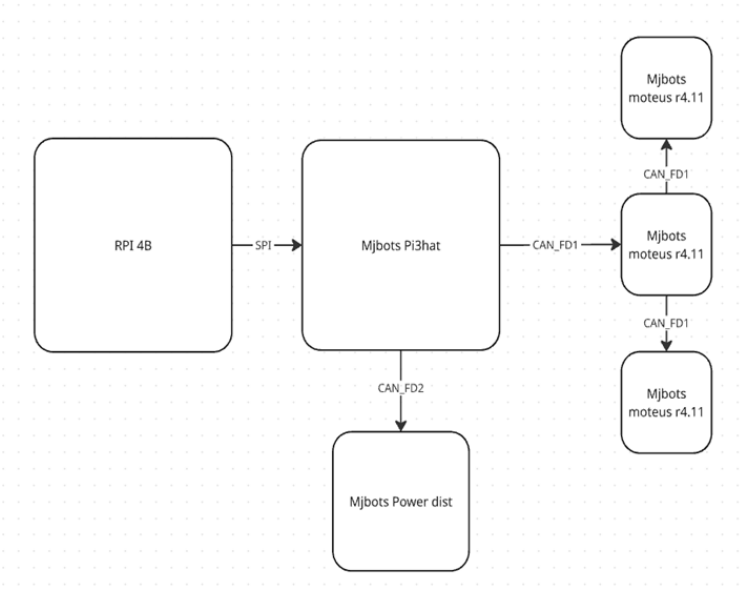


Рис. 8. Структурная схема электроники платформы

КОНСТРУКЦИЯ ПЛАТФОРМЫ

Проектирование конструкции платформы произведено в программе SolidWorks (рис. 9а, 9б, 9в). Детали основания изготовлены на лазерном станке из фанеры толщиной 6 мм. Нагруженные части напечатаны из ABS пластика с 15 процентами углеродного волокна, остальные из PETG пластика на 3D принтерах. Для крепления дополнительных модулей и в качестве крепления отсека с электроники использован стандартный конструкционный профиль тип размера 20 мм x 20 мм. В данный момент в качестве модуля установлен лидар для системы анти-столкновения (рис. 10).

Редуктор собран на основе закалённых цилиндрических валов толщиной 8 мм и стандартных подшипников.

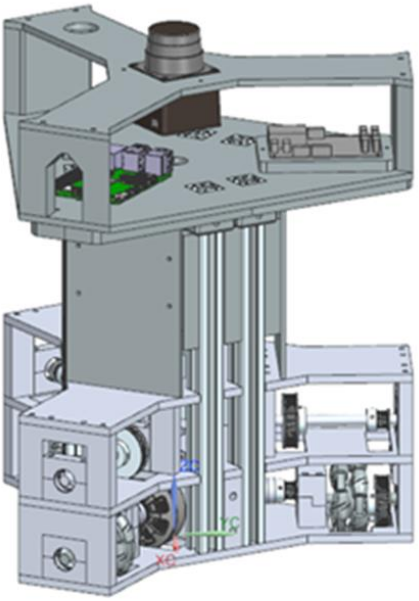


Рис. 9а. Общий вид платформы

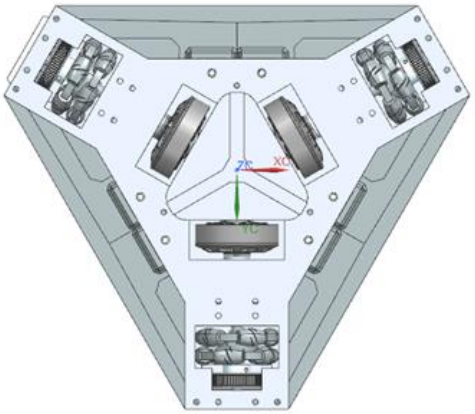


Рис. 9б. Вид снизу

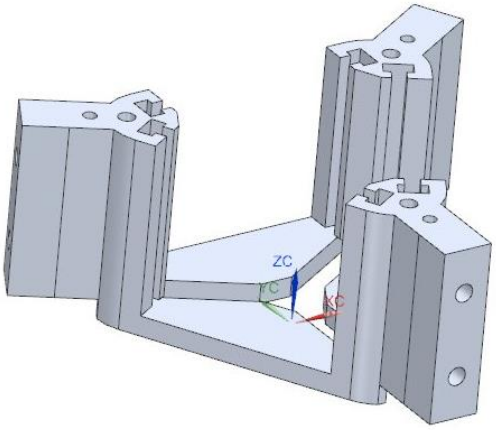


Рис. 9в. Крепление модулей мотора

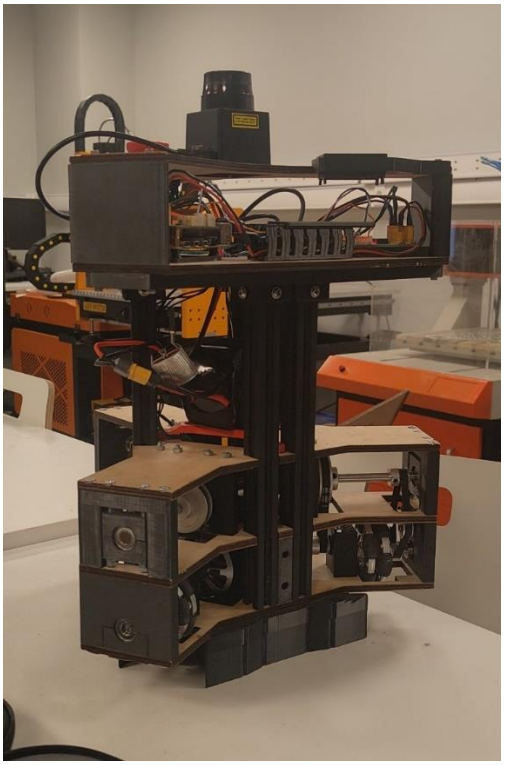


Рис. 10. Общий вид с установленным лидаром

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

На Raspberry Pi стоит операционная система Ubuntu 24.04 server, поверх которой установлен ROS2 Jazzy.

Было написано 3 библиотеки на языке программирования Python. Структурная схема работы программной части робота представлена ниже (рис.11). Moteus reader принимает значения, на которое надо повернуть каждый из моторов и отправляет эту команду через CAN-FD шину. ObstacieStopper - эта библиотека, которая принимает направление движения робота и данные лидара, и в случае, если по данным лидара есть препятствие, библиотека подаёт команду на остановку.

Omnipolar3 библиотека, которая принимает из основного файла градус - в каком направлении нужно поехать и расстояние, на которое нужно проехать и пересчитывает это всё в количество оборотов каждого из моторов.

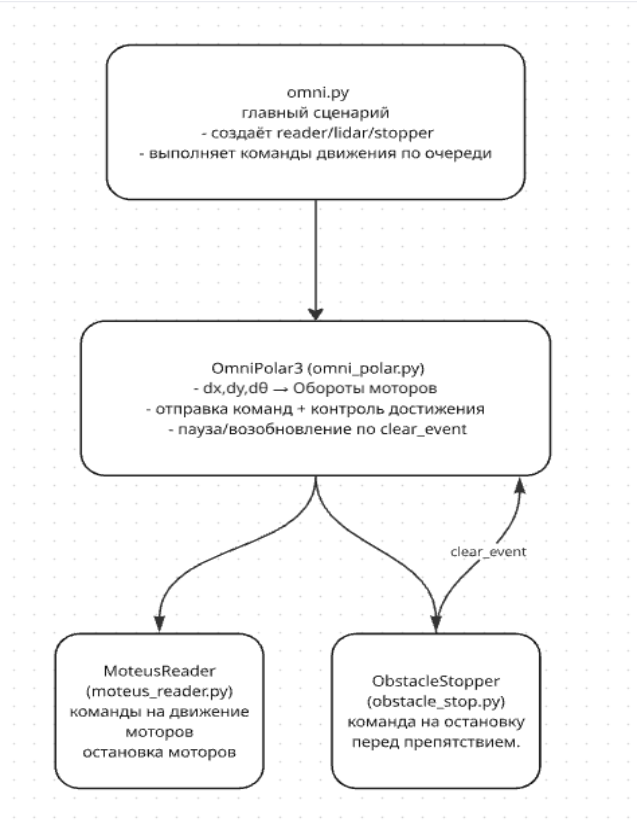


Рис. 11. Структурная схема работы программной части робота

ПРОВЕДЕНИЕ ТЕСТОВ И АНАЛИЗ РЕЗУЛЬТАТОВ

Для анализа повторяемости был проведён ряд тестов:

- движение по прямой на 0.5 метра с поворотом на 90 градусов;
- движение по прямой на 0.2 метра и возврат в точку старта:

Отклонение робота в обоих тестах составило в пределах 1 сантиметра.

Анализ рациональности использования BLDC двигателей по сравнению с шаговыми двигателями для увеличения времени работы:

Для сравнения был взят аналогичная по массе платформа на таких же омни-колёсах, но на шаговых двигателях тип размера nema 17 17HS4401 с драйверами TB6600. Было проведено измерения тока при равномерном прямолинейном движении платформы с одинаковой скоростью. Мощность в формуле:

$$P = U * I$$

Ниже представлена таблица с полученными данными.

Таблица 3. Результат полученных данных

Вид двигателей	Напряжение (В)	Ток(А)	Мощность (Вт)
BLDC	20	0.3	6
Nema 17	12	0.7	8,4

Вывод: BLDC моторы оказались на 40% более эффективны по сравнению с шаговыми двигателями.

ВЫВОДЫ

Платформы на омни-колесах с приводом на BLDC электромоторах целесообразно использовать для широкого спектра задач.

Платформа на BLDC моторах оказались на 40% более эффективна чем платформа шаговых двигателях по времени работы от аккумулятора.

В дальнейшем планируется разработать полноценную платформу и протестировать в складской логистике.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Mjbots https://github.com/mjbots

РАЗРАБОТКА ЛИНЕЙНОГО ЛАБОРАТОРНОГО БЛОКА ПИТАНИЯ

Кравцов Алексей Витальевич

Гулькевичи г, МАОУ СОШ №1 им. М.И. Короткова, 11 класс

Секция Технические системы и средства

Научный руководитель: Кузнецов Вадим Вадимович, к.т.н., доцент кафедры ИУК1, МГТУ им. Н.Э. Баумана

АННОТАЦИЯ

Линейный лабораторный блок питания (ЛБП) - вторичный источник электроэнергии, имеющий блоки регулировки выходного напряжения, тока, контроля работы, индикации режимов, схемы защиты, встроенные измерительные приборы; надёжное устройство с отсутствием радиопомех, редко выходящее из строя, легко поддающееся ремонту. Разработка ЛБП актуальна в настоящее время.

Цель: разработать линейный регулируемый малошумящий ЛБП с током нагрузки не менее 1 А, с защитой от короткого замыкания.

Задачи:

1) обобщить информацию о лабораторных блоках питания;
2) разработать электрическую и монтажную схемы ЛБП;
3) провести лабораторные испытания разработанного прибора.

Разработанный ЛБП можно использовать для питания широкого спектра устройств, требующих напряжения до 30 вольт: зарядки аккумуляторов. Он способен работать и с активно-индуктивной нагрузкой.

Преимущество разработанного ЛБП – невысокая себестоимость.

Новизна проекта заключается в том, что разработан полный комплект конструкторской документации, позволяющий самостоятельно изготовить линейный лабораторный блок питания с защитой от перегрузок даже начинающим радиолюбителям.

Работа имеет практическую значимость.

ВВЕДЕНИЕ

Линейный лабораторный блок питания - вторичный источник электроэнергии, имеющий блоки регулировки выходного напряжения, тока, контроля работы, индикации режимов, схемы защиты, встроенные измерительные приборы.

Лабораторный блок питания (ЛБП) очень востребованное оборудование, которое имеет множество применений. Наладчики, мастера и инженеры электроники для работы (проверки, калибровки) используют лабораторные блоки питания. Не малой мощности, недорогой линейный лабораторный блок питания может использоваться людьми многих профессий в их разнообразных сферах деятельности. Блоки питания применяются в научных лабораториях, сервисных отделах, на производстве, а также в учебных заведениях.

Линейный лабораторный блок питания — это надёжное устройство с отсутствием радиопомех, редко выходящее из строя, легко поддающееся ремонту. Разработка ЛБП актуальна в настоящее время.

Цель: разработать линейный регулируемый малошумящий лабораторный блок питания с током нагрузки не менее 1 А, с защитой от короткого замыкания.

Задачи:

1) обобщить информацию о лабораторных блоках питания;
2) разработать электрическую и монтажную схемы линейного ЛБП;
3) провести лабораторные испытания разработанного прибора.

Методы проектирования: анализ, синтез, абстрагирование, индукция, обобщение, дедукция, классификация, знаковое моделирование, описание, измерение, идеализация, формализация [1, с. 7].

Преимущество разработанного блока питания – невысокая себестоимость.

Новизна проекта заключается в том, что разработан полный комплект конструкторской документации, позволяющий самостоятельно изготовить линейный лабораторный блок питания с защитой от перегрузок даже начинающим радиолюбителям.

Работа имеет практическую значимость.

Краткое содержание работы по разделам:

Глава 1. Лабораторные блоки питания могут быть как линейными, так и импульсными.

В линейном БП в отличии от импульсного БП преобразование входного электрического тока в выходной электрический ток происходит за один этап: входной переменный электрический ток преобразуется в постоянный электрический ток.

Вывод: ИБП не смотря на свои некоторые преимущества, в частности: меньшие габариты и более высокий КПД, все же не могут заменить линейные БП в звуковой аппаратуре. У линейных БП есть неоспоримое преимущество - почти нет шумов независимо от шумов в

сети.

Глава 2. Линейный ЛБП.

Вывод: Разработанный блок питания предназначен для питания при ремонте и настройке радиоэлектронной аппаратуры: радиоприемников, раций, металлоискателей и других устройств, в том случае, когда нужно иметь низкий уровень электромагнитных помех.

Разработанный линейный ЛБП можно использовать для питания широкого спектра устройств, требующих напряжения до 30 вольт: зарядки аккумуляторов. Он способен работать и с активно-индуктивной нагрузкой.

ГЛАВА 1. ХАРАКТЕРИСТИКА И КЛАССИФИКАЦИЯ ЛАБОРАТОРНЫХ ИСТОЧНИКОВ ПИТАНИЯ.

1.1 Лабораторный блок питания — блок питания (БП) – вторичный источник электроэнергии (источник питания), дополненный блоками регулировки выходного напряжения и тока, контроля работы и индикации режимов, встроенным измерительным прибором (ампервольтметром), а также схемами защиты, что позволяет ему выполнять функции:

1) регулирование выходного напряжения в широких пределах;
2) ограничение по току;
3) защита от перегрева.

Лабораторные блоки питания могут быть как линейными, так и импульсными, а также иметь аналоговую или цифровую регулировку параметров.

Самые распространённые БП обеспечивают регулировку выходного напряжения в пределах от 0 до 36 вольт, имеют защиту от перегрузки. Нередко такие устройства являются полностью самодельными или построенными самостоятельно из модулей-заготовок.

Лабораторный блок питания излишки электрической энергии преобразует в тепловую.

Лабораторный блок питания – очень востребованное профессионалами оборудование. Он активно используется инженерами, занимающимися разработкой и ремонтом различных электронных устройств.

1.2 Линейные блоки питания

Линейные стабилизаторы постоянного напряжения находят широкое применение для питания радиоэлектронной аппаратуры (РЭА). Несмотря на то, что импульсные источники имеют ряд преимуществ и значительно более лучшие массогабаритные показатели, чем линейные, у последних в спектре выходного напряжении полностью отсутствуют высокочастотные составляющие, попадающие в диапазон до 30 МГц и выше. Поэтому линейные источники чаще используются для обеспечения постоянным напряжением

аппаратуры, к которой предъявляются повышенные требования к уровню помех в цепях питания, например, в приёмопередающих устройствах КВ и УКВ диапазона, а также в измерительной и звуковой технике где для достижения точностных и качественных характеристик, наряду с другими методами электромагнитной совместимости (ЭМС), важное значения имеет подавление пульсаций напряжения в цепях питания. По данной причине актуальна задача разработки оригинальных схемотехнических и конструкторских решений в данной области и улучшение параметров существующих [2, с. 36].

Традиционным считается линейный блок питания. Блок-схема линейного блока питания [3] представлена на рисунке 1:

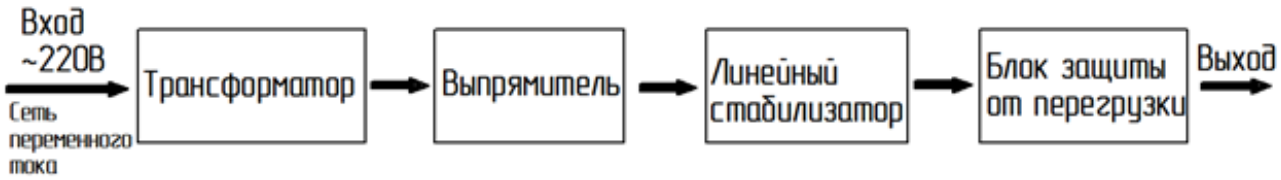


Рис. 1. Блок-схема линейного блока питания

Входной трансформатор приводит величину сетевого напряжения к заданному значению выходного напряжения, причём выходное напряжение может быть, как выше входного напряжения, так и ниже входного. В первом случае такой трансформатор называется повышающим, а во втором – понижающим.

Преобразование формы входного сигнала обеспечивает блок выпрямителей, который преобразует синусоидальный входной ток в последовательность положительных или отрицательных полупериодов с частотой следования импульсов в 50 герц при однополупериодном выпрямлении или 100 герц при двухполупериодном выпрямлении.

Такая значительная пульсация питающего напряжения не допустима для большинства потребителей. Уменьшение пульсации преобразованного напряжения осуществляется сглаживающим фильтром.

Величина напряжения после сглаживающего фильтра нормируется к величине напряжения необходимой самой нагрузке линейным стабилизатором, который обеспечивает заданное электрическое напряжение при электрическом токе определённым величиной внутреннего сопротивления нагрузки. Величина приложенного напряжения к нагрузке может меняться в некоторых пределах аналоговым методом.

Блок защиты от перегрузки запрещает работу линейного стабилизатора в случае подключения нагрузки большей, чем на которую рассчитан линейный стабилизатор или если в нагрузке, по каким-либо причинам, произошло короткое замыкание.

Работа линейного блока питания всегда сопровождается нагревом, так как часть электрической энергии при преобразовании сетевого напряжения в энергию потребляемой

нагрузкой выделяется в виде тепла. Количество электрической энергии, бесполезно растраченной на нагрев, определяется КПД каждого блока, входящего в его структуру. Нагрев требует организацию отвода тепла в пространства, а в некоторых случаях запрета работы линейного стабилизатора при перегреве - тепловая защита.

Вывод: В линейном БП преобразование входного электрического тока в выходной электрический ток происходит за один этап: входной переменный электрический ток преобразуется в постоянный электрический ток.

1.2.1 Преимущества и недостатки линейных блоков питания

Преимущества линейных блоков питания:

- простота и надежность конструкции;
- высокая ремонтопригодность, низкая себестоимость;
- отсутствие электромагнитных помех.

Недостатки линейных блоков питания:

- большой вес и габариты;
- обратная зависимость между КПД и стабильностью выходного напряжения;
- высокая металлоемкость конструкции [4].

1.3 Импульсные блоки питания

Принцип работы импульсного блока питания достаточно прост, блок-схема импульсного блока питания представлена на рисунке 2:

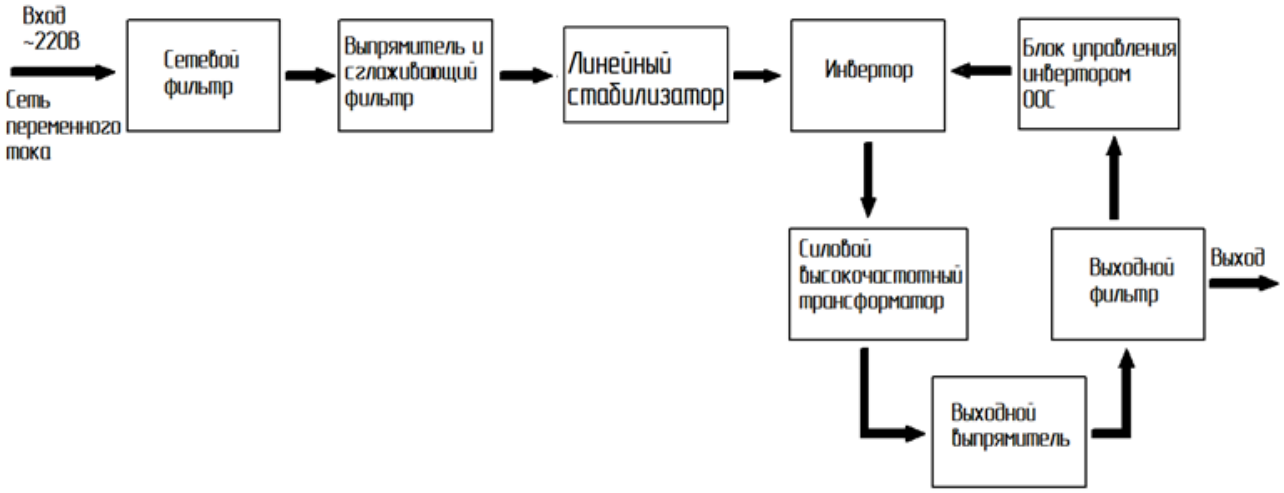


Рис. 2. Блок-схема импульсного блока питания

В отличие от линейного БП входным блоком ИБП является сетевой фильтр, а не силовой трансформатор, наличие фильтра обоснованно необходимостью подавления высокочастотных помех, возникающих при работе блока инвертора и способных проникать в питающую сеть нарушая работу других приборов.

Работа блока выпрямителя, сглаживающего фильтра и линейного стабилизатора ИБП принципиально не отличается от работы соответствующих блоков линейного БП.

Инвертор вырабатывает (генерирует) импульсы высокой частоты, поступающие через силовые ключи инвертора на высокочастотный трансформатор. Выходное напряжение высокочастотного трансформатора может быть, как ниже (чаще всего), так и выше амплитудного значения высокочастотных импульсов инвертора. Значение выходного напряжения высокочастотного трансформатора можно изменять в некоторых пределах за счёт изменения скважности импульсов от инвертора.

Работа выходного выпрямителя и выходного фильтра принципиально мало отличается от работы ранее описанных соответствующих блоков за исключением, что выпрямление и сглаживание пульсаций происходит не на частоте питающей сети, а на частоте работы инвертора. Выходной фильтр является заградительным фильтром, его задача не пропустить высокочастотные помехи к потребителю. Регулировка выходного напряжения и его стабилизация осуществляется с помощью отрицательной обратной связи (ООС), воздействующей на скважность импульсов инвертора.

Вывод: в отличие от линейного БП в ИБП преобразование входного электрического тока в выходной электрический ток происходит в несколько этапов. Первый этап – входной переменный электрический ток преобразуется в постоянный электрический ток. Второй этап – постоянный электрический ток инвертором преобразуется в пульсирующий электрический ток большой частоты. Третий этап – пульсирующий электрический ток большой частоты преобразуется в выходной электрический постоянный ток.

1.3.1 Преимущества и недостатки импульсных источников

К достоинствам импульсных агрегатов нужно отнести:

- высокий коэффициент стабилизации;
- высокий коэффициент полезного действия;
- более широкий диапазон входных напряжений;
- более высокую мощность, по сравнению с линейными устройствами;
- отсутствие чувствительности к качеству электропитания и частоте входного напряжения;
- небольшие габариты;
- доступная цена [4].

К явным недостаткам импульсных БП можно отнести:

- импульсная система питания дает высокочастотные помехи;
- сложность схем, что негативно сказывается на надежности;
- низкая ремонтопригодность.

Вывод: ИБП не смотря на свои некоторые преимущества, в частности: меньшие габариты и более высокий КПД, все же не могут заменить линейные БП в звуковой аппаратуре. У линейных БП есть неоспоримое преимущество - почти нет шумов независимо от шумов в сети.

ГЛАВА 2. ЛИНЕЙНЫЙ ЛАБОРАТОРНЫЙ БЛОК ПИТАНИЯ

Разработанный линейный лабораторный блок питания имеет следующие характеристики:

1) Uвх ном 220 В;
2) Iвх ном 0,2 А;
3) Рвх ном 44 Вт;
4) Uвых 1,2 – 30 В;
5) Iвых мак 1 А.

Сравнение технических характеристик разработанного линейного ЛБП с аналогами [5], [6], [7] приведено в таблице 1.

Таблица 1. Сравнение технических характеристик

Наименование/ Параметры	Линейный ЛБП	XBL1503C	QJ1803C	GPS-71830D
Конструкторская документация	есть	нет	нет	нет
Возможность самостоятельного изготовления	есть	нет	нет	нет
Напряжение	(0-30) В	(0-15) В	(0-18) В	(0-18) В
Ток	(0-3) А*	(0-3) А	(0-3) А	(0-3) А
Защита от кз	да	да	да	да
Защита от перегрузок	да	да	-	-
Подключение активно-индуктивной нагрузки	да	-	-	-
Индикация значений напряжения и тока	да	да	да	да
Питание	(220±20) В	110/220 В ±10%	110/220 В ±10%	220 В

Комплектация	конструкторская документация, соединительные провода	-	инструкция, щупы	-
Наименование/ Параметры	Линейный ЛБП	XBL1503C	QJ1803C	GPS-71830D
Масса	2,3 кг	-	-	4,5 кг
Размеры	210x210x150	-	95x150x240	-
Активное охлаждение	да	да	-	-
Госреестр РФ	нет	да	нет	да

* с учётом мощности 60 Вт

Эскиз разработанного линейного лабораторного блока питания представлен в приложении Б.

Блок-схема разработанного линейного лабораторного блока питания представлена на рисунке 3:

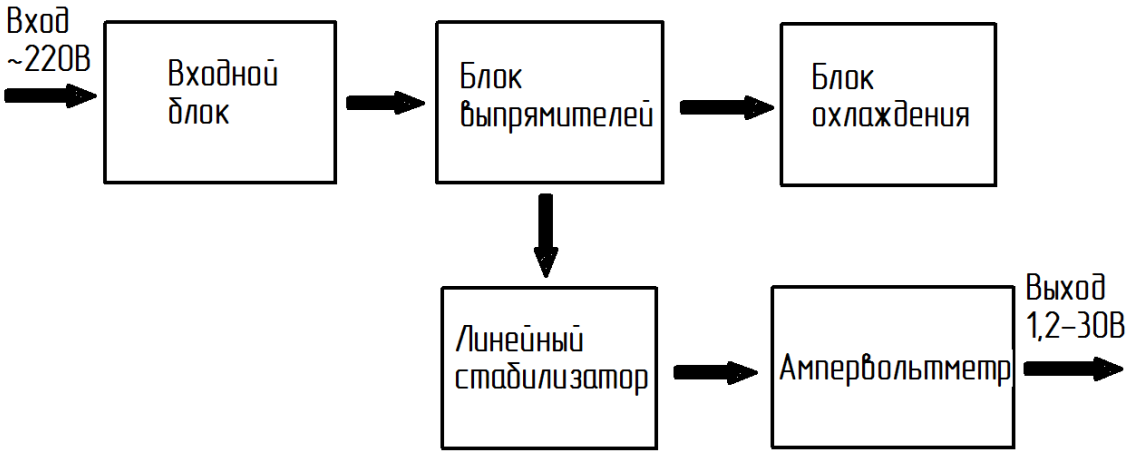


Рис. 3. Блок-схема линейного лабораторного блока питания

Электрическая схема линейного ЛБП размещена на рисунке 4, на рисунке 5 приведена электрическая схема, выполненная в программе EasyEDA:

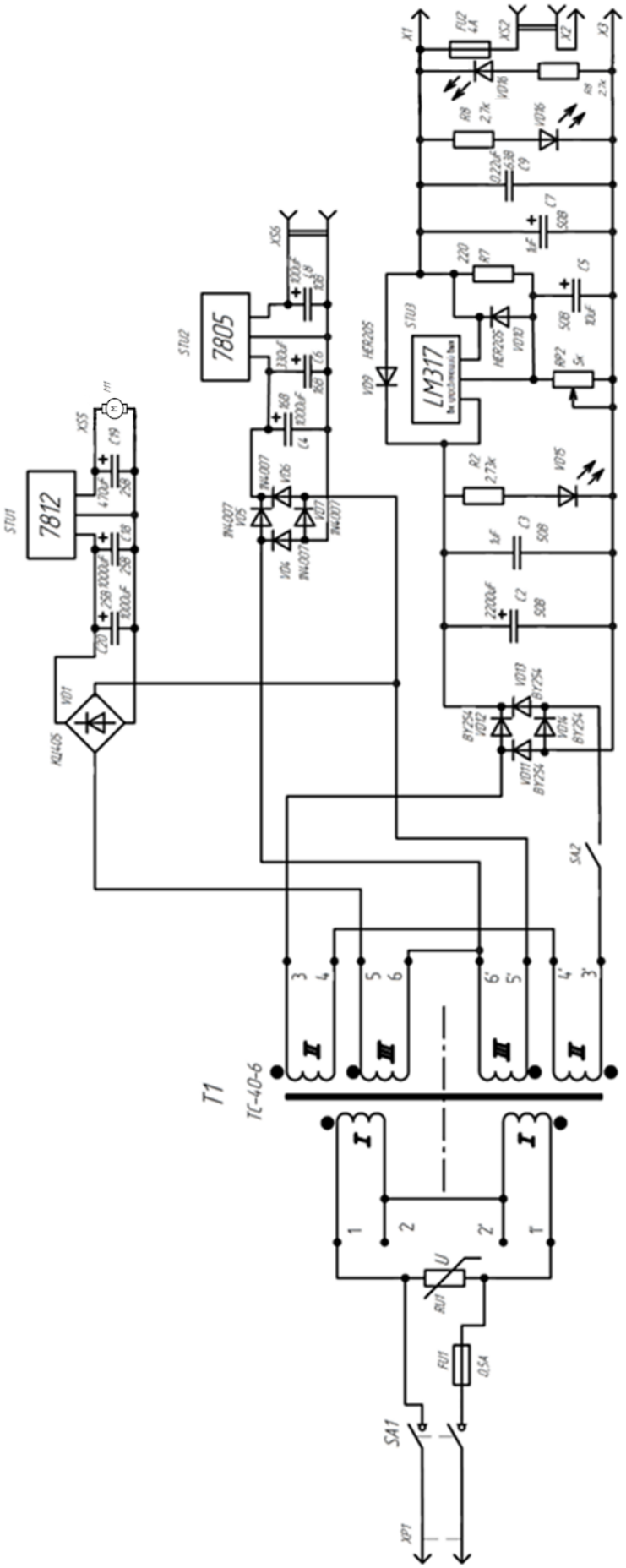


Рис. 4. Электрическая схема линейного ЛБП

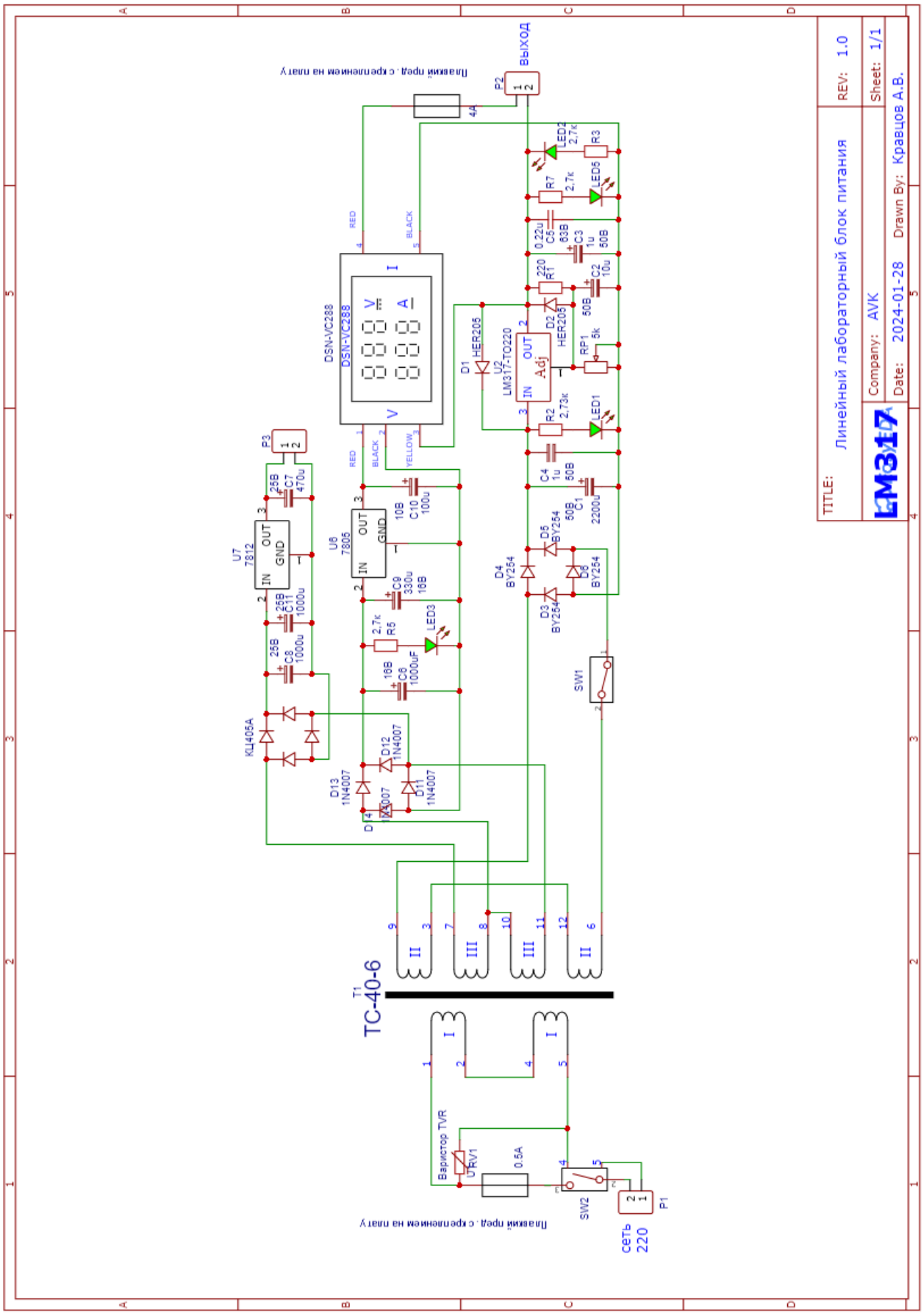


Рис. 5. Электрическая схема линейного ЛБП, созданная в EasyEDA

2.1 Входной блок

Входной блок выполнен на силовом трансформаторе ТС40-6 и имеет защиту от перенапряжения, реализованную на варисторе RU1 S14K250 и плавком предохранителе FU1 0.5 А (приложение В) [8]. Электрическая схема входного блока представлена на рисунке 6:

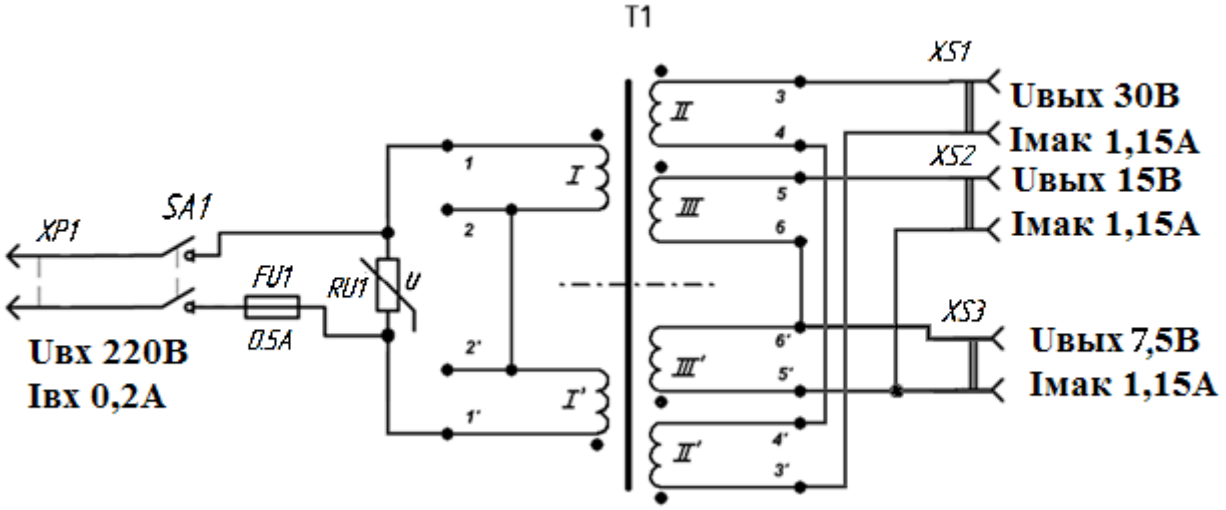


Рис. 6. Электрическая схема входного блока

Входной блок имеет характеристики:

1) Uвх ном 220 В;
2) Iвх ном 0,2 А;
3) Uвых1 30 В;
4) Iвых1 1,15 А;
6) Uвых2 15 В;
7) Iвых2 1,15 А;
8) Uвых2 7,5 В;
9) Iвых2 1,15 А.

Защита работает следующим образом: при превышении входного напряжения выше номинального напряжения первичной обмотки на 13,6 % приводит к пробою варистора и увеличению тока, проходящего через плавкий предохранитель выше его номинального значения, что, в свою очередь, вызывает выгорание плавкого предохранителя и обесточивание первичной обмотки силового трансформатора, рисунок 7 [9, с. 23]:

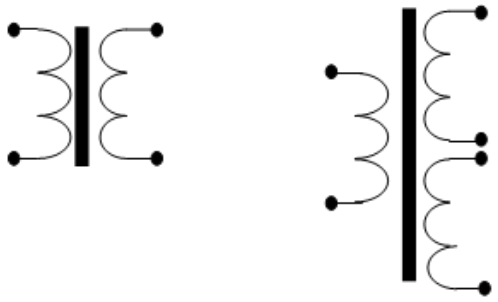


Рис. 7. Трансформатор:

а – с одной вторичной обмоткой, б – с двумя вторичными обмотками

2.2 Блок выпрямителей

Электрическая схема блока выпрямителей представлена на рисунке 8:

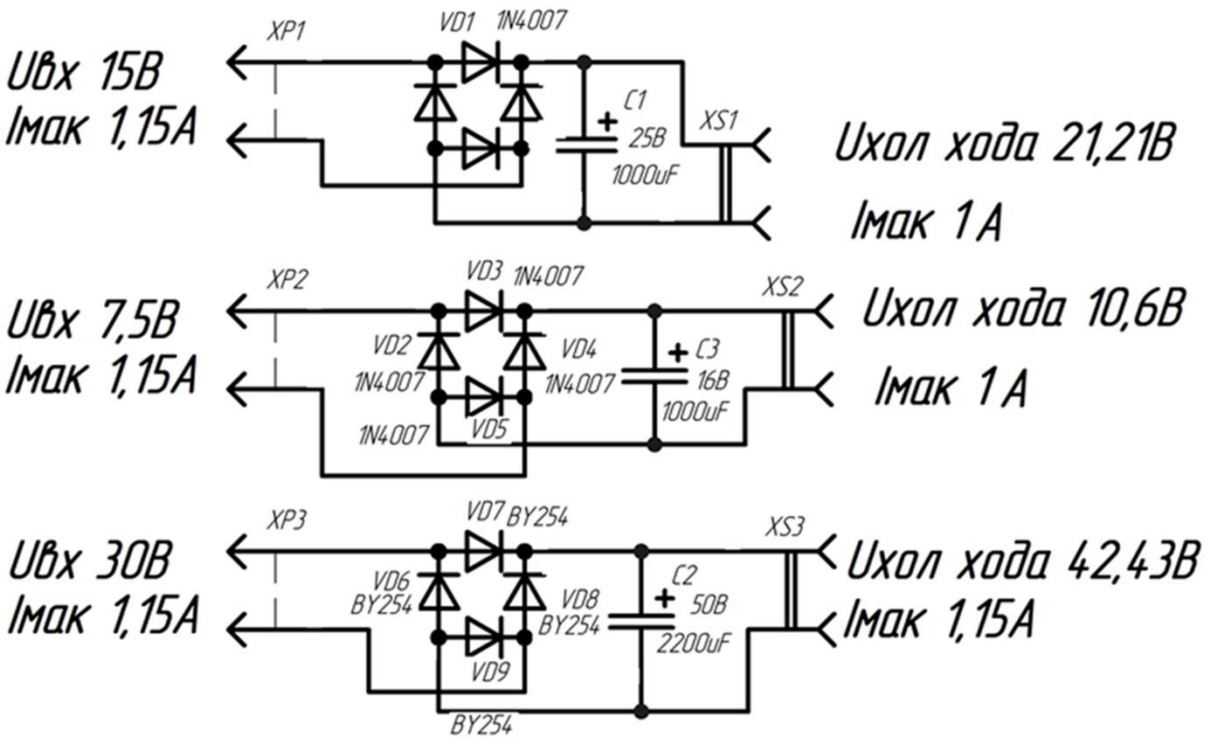


Рис. 8. Электрическая схема блока выпрямителей

Блок выпрямителей выполнен на диодах BY254, 1N4007 и конденсаторах C1, C2, C3.

Блок выпрямителей имеет характеристики:

1) Uвх1 15 В (~ входное напряжение выпрямителя блока охлаждения);
2) Iвх1 1,15 А (~ входной ток выпрямителя блока охлаждения);
3) Uхол хода1 21,21 В (⎓ выходное напряжение выпрямителя блока охлаждения на холостом ходу);
4) Iвых1 1 А (⎓ выходной ток выпрямителя блока охлаждения);
5) Iмак1 1 А (⎓ максимальный ток выпрямителя блока охлаждения);
6) Uвх2 7,5 В (~ входное напряжение выпрямителя блока питания ампервольтметра);

7) Iвх2 1,15 А (~ входной ток выпрямителя блока питания ампервольтметра);
8) Uхол хода2 10,6 В (⎓ выходное напряжение выпрямителя блока питания ампервольтметра на холостом ходу);
10) Iвых2 1 А (⎓ выходной ток выпрямителя блока питания ампервольтметра);
11) Iмак2 1 А (⎓ максимальный ток выпрямителя блока питания ампервольтметра);
12) Uвх3 30 В (~ входное напряжение выпрямителя линейного стабилизатора);
13) Iвх3 1,15 А (~ входной ток выпрямителя линейного стабилизатора);
14) Uхол хода3 42,43 В (⎓ выходное напряжение выпрямителя линейного стабилизатора на холостом ходу);
15) Iвых3 1,15 А (⎓ выходной ток выпрямителя линейного стабилизатора);
16) Iмак3 3 А (⎓ максимальный выходной ток выпрямителя линейного стабилизатора).

2.3 Линейный стабилизатор

Электрическая схема блока линейный стабилизатор представлена на рисунке 9:

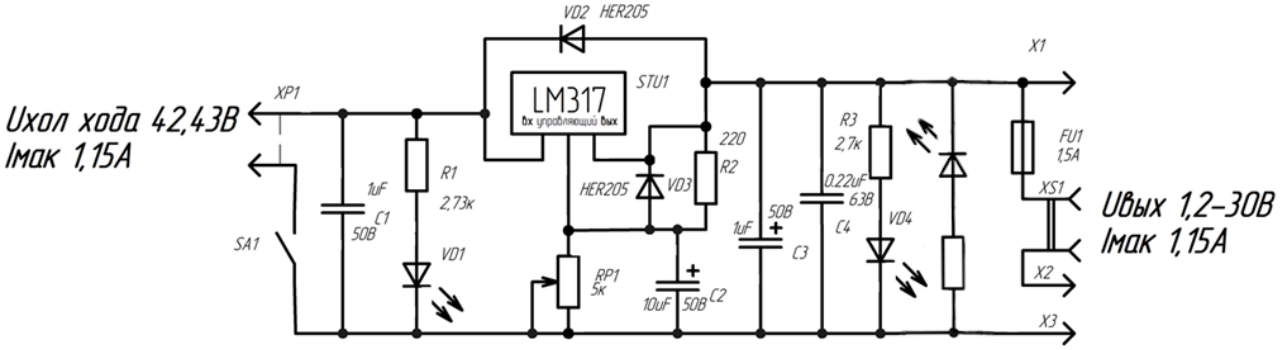


Рис. 9. Электрическая схема блока линейный стабилизатор

Линейный стабилизатор выполнен на микросхеме LM317 по стандартной схеме её подключения согласно документации (приложение Г) [10]. Регулировка выходного напряжения осуществляется переменным резистором RP1 5 кОм. Светодиоды VD1 и VD4 предназначены для индикации наличия питания до и после стабилизатора.

2.4 Ампервольтметр

Цифровой ампервольтметр DCN-VC288 предназначен для контроля выходного напряжения и тока, протекающего через нагрузку. Точность контроля параметров составляет ± 1 %. Питание ампервольтметра осуществляется напряжением 5 вольт от отдельного линейного стабилизатора, выполненного на микросхеме 7805, рисунок 10:

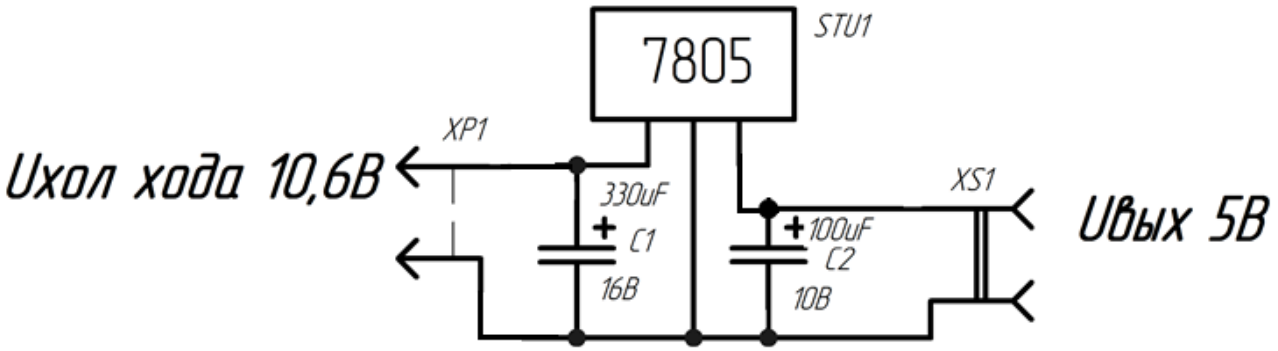


Рис. 10. Линейный стабилизатор ампервольтметра

2.5 Блок охлаждения

Электрическая схема блока охлаждения представлена на рисунке 11:

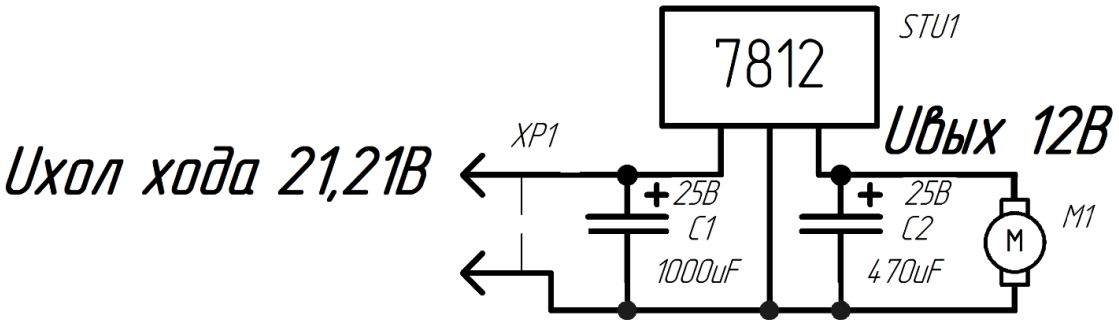


Рис. 11. Электрическая схема блока охлаждения

Блок питания вентилятора охлаждения выполнен на микросхеме 7812 по стандартной схеме и не имеет особенностей. Обдув греющихся элементов начинается сразу после включения линейного блока питания и заканчивается после его выключения.

Вывод: Разработанный блок питания предназначен для питания при ремонте и настройке радиоэлектронной аппаратуры: радиоприемников, раций, металлоискателей и других устройств, в том случае, когда нужно иметь низкий уровень электромагнитных помех.

Разработанный линейный ЛБП можно использовать для питания широкого спектра устройств, требующих напряжения до 30 вольт: зарядки аккумуляторов. Он способен работать и с активно-индуктивной нагрузкой.

ЗАКЛЮЧЕНИЕ

Целью работы была разработка линейного регулируемого малошумящего лабораторного блока питания с током нагрузки не менее 1 А, с защитой от короткого замыкания.

Был разработан лабораторный БП, имеющий лучшие характеристики, чем указано в задании (приложение Д).

Задачи проекта – выполнены, цель проекта достигнута.

Работа выполнена мной полностью самостоятельно. Собственноручно разработана

электрическая схема блока питания, конструкция и проведена отладка и изготовление опытного образца.

Разработанный блок питания могут использовать в школах, техникумах и других учебных заведениях, а также радиолюбители в качестве универсального многоцелевого источника питания. Именно линейный ЛБП очень хорошо подойдет для использования в учебных заведениях, т.к. он имеет простую конструкцию и принцип его работы проще понять.

Разработан полный комплект конструкторской документации, позволяющий самостоятельно изготовить линейный лабораторный блок питания с защитой от перегрузок даже начинающим радиолюбителям.

Работа имеет практическую значимость.

Преимущество разработанного БП – это его невысокая себестоимость.

В ходе работы над проектом был разработан, изготовлен и испытан (приложение Е) линейный регулируемый лабораторный блок питания с током нагрузки не менее 1 А, с защитой от короткого замыкания, имеющего широкий спектр применения (приложение Ж).

Выражаю особую благодарность моему научному руководителю Кузнецову Вадиму Вадимовичу, к.т.н., доценту кафедры ИУК1 за значимые замечания и важнейшие советы при оформлении проекта, а также учителям физики Натальной Ксении Вячеславовне и Фербер Светлане Владимировне, наставнику Шишкину Евгению Маленовичу, главе Гулькевичского района Шишикину Александру Александровичу.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Официальная документация Python:
2. Эйсмонт, Н. Г. Теоретические основы и практика научных исследований [Электронный ресурс]: учеб. пособие / Н. Г. Эйсмонт, В. В. Даньшина, С. В. Бирюков; Минобрнауки России, ОмГТУ. – Омск: Изд-во ОмГТУ, 2018. – Режим доступа: https://www.omgtu.ru/general_information/faculties/radio_engineering_department/department_of_quot_physics_quot/lib_pfys/280402280302/Teor_osnovi_prakt_nauch_issl.pdf. (дата обращения: 16.03.2025).
3. Схемотехническое моделирование подавления пульсаций в линейных стабилизаторах постоянного напряжения / В. В. Кузнецов, Н. М. Крючков, А. И. Козлов, И. Ю. Цирульников // Технологии электромагнитной совместимости. – 2023. – № 4(87). – С. 36-43. – EDN KVDUDN.
4. Схемы электрические структурные [Электронный ресурс]. – Режим доступа: https://digteh.ru/InjGraf/SxElektr/strukt/. (дата обращения: 15.09.2025).
5. Блок питания: импульсный или трансформаторный? Разбираем преимущества и недостатки [Электронный ресурс]. – Режим доступа: https://supereyes.ru/articles/power_supply/laboratornyy_blok_pitaniya_impulsnyy_ili_lineynyy_kakoy_vybrat/#impulsnye. (дата обращения: 17.03.2025).
6. БП GPS-71830D [Электронный ресурс]. – Режим доступа: https://www.chipdip.ru/product/istochnik-pitaniya-gps-71830d-gw-instek-8661389176?from=suggest_product. (дата обращения: 20.12.2025).
7. БП QJ1803C [Электронный ресурс]. – Режим доступа: https://www.chipdip.ru/product/qj1803c-istochnik-pitaniya-0-18v-3a-54w-qje-9000402103?from=suggest_product. (дата обращения: 20.12.2025).
8. БП XBL1503C [Электронный ресурс]. – Режим доступа: https://www.chipdip.ru/product/xbl1503c-istochnik-pitaniya-lineynyy-0-15v-3a-45w-xibo-9001528494?from=suggest_product. (дата обращения: 20.12.2025).
9. Трансформаторы силовые ТС-40 [Электронный ресурс]. - Режим доступа: https://vprl.ru/index/transformatory_ts_40/0-46. (дата обращения: 15.10.2025).
10. Горошков Б. И. Электронная техника: учеб. пособие для студ. сред. проф. образования / Б. И. Горошков, А. Б. Горошков. - 2-е изд., стер. - М.: Издательский центр «Академия», 2008. – 320 с.
11. LM317: Характеристики, виды и схемы [Электронный ресурс]. – Режим доступа: https://www.ruselectronic.com/lm317/. (дата обращения: 17.09.2025).

ДЕТСКАЯ КАРТА С РАШИРЕННЫМ РОДИТЕЛЬСКИМ КОНТРОЛЕМ: ДЛЯ ВАШЕГО СПОКОЙСТВИЯ И ФИНАНСОВОЙ ГРАМОТНОСТИ РЕБЕНКА

Измаилова Алсу Ринатовна

Москва г, ГБОУ Школа №444, 10 класс

Секция: Инженерный бизнес и менеджмент

Научные руководители: Ляхович Дмитрий Геннадьевич, старший преподаватель ИБМ3, МГТУ им. Н.Э. Баумана

Соколов Михаил Александрович, к.н., доцент каф. ИБМ7 МГТУ им. Н.Э. Баумана

АННОТАЦИЯ

Цель — сделать детскую карту, которая не позволит ребёнку купить вредное или опасное. Как это работает — мы даём родителям настройки, чтобы заблокировать оплату в определённых магазинах (например, фастфуда) или на определённые товары (например, сладости). Что сделали — провели опросы, изучили конкурентов и создали работающую систему. Вывод — наша карта не просто отслеживает траты, а заранее защищает от ненужных покупок, давая родителям настоящий контроль и спокойствие.

ВВЕДЕНИЕ

Актуальность темы исследования

За последние десятилетия финансовая система претерпевает глобальные изменения, связанные с всеобщий цифровизацией экономики. На данный момент доля безналичных платежей в России достигла максимумов за всё время, практически вытеснив наличные деньги из повседневного пользования. Эти изменения затронули все слои населения, включая детей и подростков. Современные школьники получают доступ к банковским картам значительно раньше, чем предыдущие поколения: персональная карта стала таким же предметом повседневности, как телефон.

Однако стремительное развитие финансовых технологий создало новую проблему: разрыв между технической возможностью тратить деньги и психологической готовностью ребенка к осознанным тратам. Отсутствие физического контакта с деньгами приводит к тому, что дети перестают ощущать их ценность, и начинают воспринимать средства на карте как абстрактные цифры, которые с легкостью можно тратить в онлайн-игре или на импульсивные покупки.

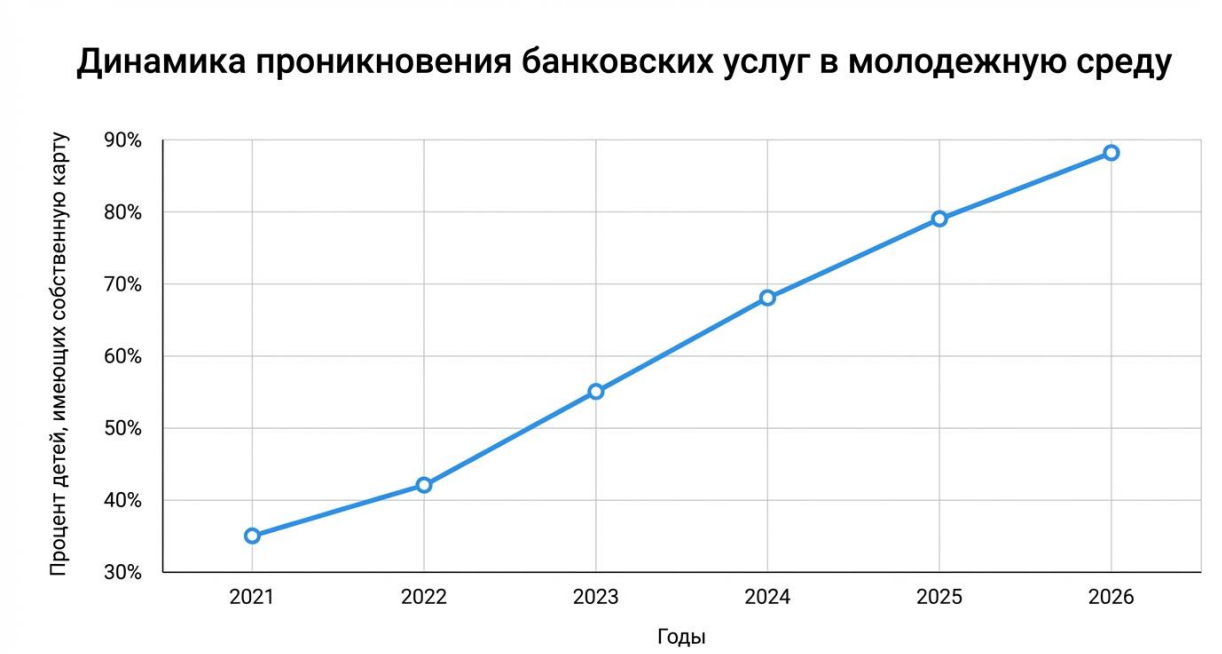


Рис. 1. Динамика проникновения банковских услуг в молодежную среду

Более того, цифровая среда несет в себе риски, с которыми родители зачастую не могут справиться традиционными методами воспитания. Существующие банковские приложения предлагают лишь базовый набор ограничений (лимит на месяц, ограничение на переводы и снятие наличных), что является недостаточным. Они не позволяют гибко регулировать категории покупок, блокировать сомнительные торговые точки или предотвращать транзакции в конкретных онлайн-сервисах. Родители оказываются в ситуации, когда узнают о нежелательной покупке постфактум, что приводит к конфликтам в семье и финансовым потерям.

Таким образом, разработка системы, обеспечивающей “умный” и гибкий родительский контроль, является не просто востребованной, а необходимой мерой для обеспечения финансовой безопасности и формирования правильной финансовой дисциплины у

подрастающего поколения.

Проблема исследования заключается в противоречии между широким распространением детских банковских карт и отсутствием эффективных технических инструментов для более чёткого родительского контроля и обучения детей финансовой грамотности.

Объект исследования: процесс формирования финансовой грамотности и безопасности несовершеннолетних пользователей банковских услуг.

Предмет исследования: программные алгоритмы и интерфейсные решения для создания банковской карты с расширенным родительским контролем «KidSafeKard».

Цель проекта: разработать концепцию и прототип мобильного приложения для детской банковской карты «KidSafeKard», которое позволит предотвращать нежелательные траты посредством гибкой системы фильтрации и повысит уровень осознанности потребления у детей. На начальном этапе обучение будет строиться на запретах, но в дальнейшем дети сами будут совершать осознанные покупки, не находясь под влиянием различных непроверенных людей, что научит их самостоятельному, здравому решению, что поможет и в дальнейшей жизни.

Для достижения поставленной цели необходимо решить следующие задачи:

- Теоретические:
- Изучить особенности финансового поведения современных подростков в условиях цифровой экономики.
- Провести сравнительный анализ существующих предложений детского банкинга (Сбер, Т-Банк, Альфа и др.) и выявить их функциональные недостатки.
- Рассмотреть психологические аспекты детского шопоголизма и влияния микротранзакции в играх.

Аналитические:

- Провести опрос среди родителей для выявления основных страхов, связанных с финансами детей.
- Определить перечень необходимых функций для MVP (минимально жизнеспособного продукта).

Практические:

- Создать архитектуру приложения и базы данных.
- Разработать пользовательский интерфейс (UI/UX) для двух вариантов: «Родитель» и “Ребенок”.
- Создать демонстрационный прототип системы «KidSafeKard».

Гипотеза проекта: внедрение системы превентивного контроля категорий покупок (на основе MCC-кодов) вместо простых лимитов на сумму позволит снизить количество нежелательных трат на 40% и снизит уровень тревожности родителей.

Методы исследования:

- Теоретические: анализ литературы и интернет-источников, сравнительный анализ аналогов.
- Эмпирические: анкетирование (Google Forms), интервьюирование целевой аудитории.

- Проектные: прототипирование интерфейсов, моделирование бизнес-процессов.

Практическая значимость:

Результаты работы могут быть использованы как основа для стартапа в сфере FinTech или предложены в качестве модуля для интеграции в существующие экосистемы крупных банков. Разработанный продукт «KidSafeKard» помогает родителям автоматизировать процесс финансового воспитания, а детям - безопасно учиться управлять личным бюджетом

ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ ДЕТСКОЙ ФИНАНСОВОЙ БЕЗОПАСНОСТИ

1.1. Финансовая грамотность детей в условиях цифровой экономики

В условиях цифровой экономики, которая к 2026 году стала доминирующей, понятие финансовой грамотности изменилось. Раньше под этим термином понималось умение считать сдачу и копить наличные деньги в копилке, то сегодня главное умение управления невидимыми, виртуальными активами.

Основной проблемой современного поколения детей -отсутствие физического ощущения расставания с деньгами. Психологи отмечают феномен снижения “боли оплаты” (pain of paying). Передача купюры вызывает микро-стресс, который даёт мозгу сигнал о потере ресурса. При оплате смартфоном, часами или картой этот сигнал отсутствует. Для ребенка баланс на карте представляется невещественной цифрой, которая обновляется родителями, а не результат, являющимся эквивалентом труда.

Цифровая среда негативно воздействует на психику несовершеннолетних. Маркетинговые стратегии сместились в сторону визуализации: приложения и онлайн-магазины используют яркие образы, таймеры обратного отсчета и системы вознаграждений, тем самым увеличивая желание приобрести различные товары. В этих условиях финансовая грамотность перестает быть просто знанием математики, а становится комплексом навыков цифрового и критического мышления.

Рис. 2. Сравнение прошлого и настоящего.

Таким образом, в современной цифровой экономике ребенок оказывается один на один с алгоритмами продаж, разработанными профессиональными маркетологами. Без специализированных инструментов контроля и обучения со стороны родителей, формирование

здоровых финансовых привычек становится практически невозможным, что ведет к искаженному восприятию реальности и потребительскому отношению к семейному бюджету.

1.2. Анализ существующего рынка детских банковских карт и их недостатки

К 2026 году на российском рынке представленное большое количество различных детских банковских карт. Крупнейшие из которых (Сбер, Т-Банк, Альфа-Банк, ВТБ) предлагают продукты для детей от 6 до 14 лет. Чаще всего это отдельные пластиковые карты с ярким дизайном, привязанные к счету родителя, и мобильные приложения.

Однако, проведя анализ функционала этих приложений, было выявлено ряд системных недостатков, которые делают не делают достаточно эффективными инструментами контроля.

1. Постфактум-контроль. Системы работают по принципу уведомления: родитель получает SMS уже после того, как транзакция совершена. Вернуть деньги в случае покупки цифрового контента или товаров в сомнительных магазинах крайне сложно.

2. Грубая система лимитов. Существующие настройки позволяют установить лишь общие лимиты (на день, на месяц) или полный запрет на снятие наличных/покупки в интернете. Например, родитель хочет, чтобы ребенок мог оплатить обед в школе картой, но не мог купить валюту в онлайн-игре. Стандартные банковские приложения не позволяют разделить эти оплаты, так как не предлагают гибкой настройки по MCC-кодам (Merchant Category Code - код категории продавца).

3. Отсутствие «Черных списков» конкретных точек. Родитель не может заблокировать оплату в конкретном оффлайн или онлайн магазинах.

Таблица 1. Сравнение «KidSafeKards» с конкурентами

Таблица сравнения «KidSafeKards vs Конкуренты»

	Обычная детская карта	KidSafeKards
Лимит на сумму:	✔ Есть	✔ Есть
Блокировка конкретных магазинов:	✕ Нет	✔ **Есть**
Фильтрация по категориям товаров (MCC):	Частичная	✔ **Полная**
Обучающие задания перед покупкой:	✕ Нет	✔ **Есть**

Исходя из этого, можно сделать вывод, что текущие рыночные решения ориентированы больше на привлечение клиентов красивым дизайном, нежели на обеспечение реальной безопасности и управляемости трат. Проект KidSafeKard призван заполнить этот технологический пробел.

1.3. Психологические и социальные риски бесконтрольных трат

Когда ребёнок имеет бесконтрольный доступ к электронным средствам это несет в себе различные угрозы, выходящие за рамки финансовых потерь семьи. Эти риски можно разделить на две большие группы: психологические и социальные.

Психологические риски: наибольшую опасность представляют микротранзакции в компьютерных играх и мобильных приложениях. Разработчики игр используют “темные паттерны” - интерфейсные уловки. Механика “лутбоксов” (случайных наград) воздействует на те же центры мозга, что и азартные игры, вызывая дофаминовую зависимость. Ребенок, имеющий карту без ограничений на категорию “Игры”, рискует стать жертвой игровой зависимости уже в раннем возрасте. Покупка виртуальных предметов становится способом снятия стресса или повышения социального статуса среди сверстников, замещая реальные достижения.

Социальные риски и безопасность: второй аспект касается физической безопасности и здоровья. Имея бесконтрольные средства, подростки часто приобретают товары, запрещенные или нежелательные для их возраста, но доступные в недобросовестных торговых точках. Кроме того, отсутствие контроля делает детей подвластными для мошенников. Переводы на “безопасные счета” или оплата несуществующих товаров на фишинговых сайтах - распространенные сценарии хищений.

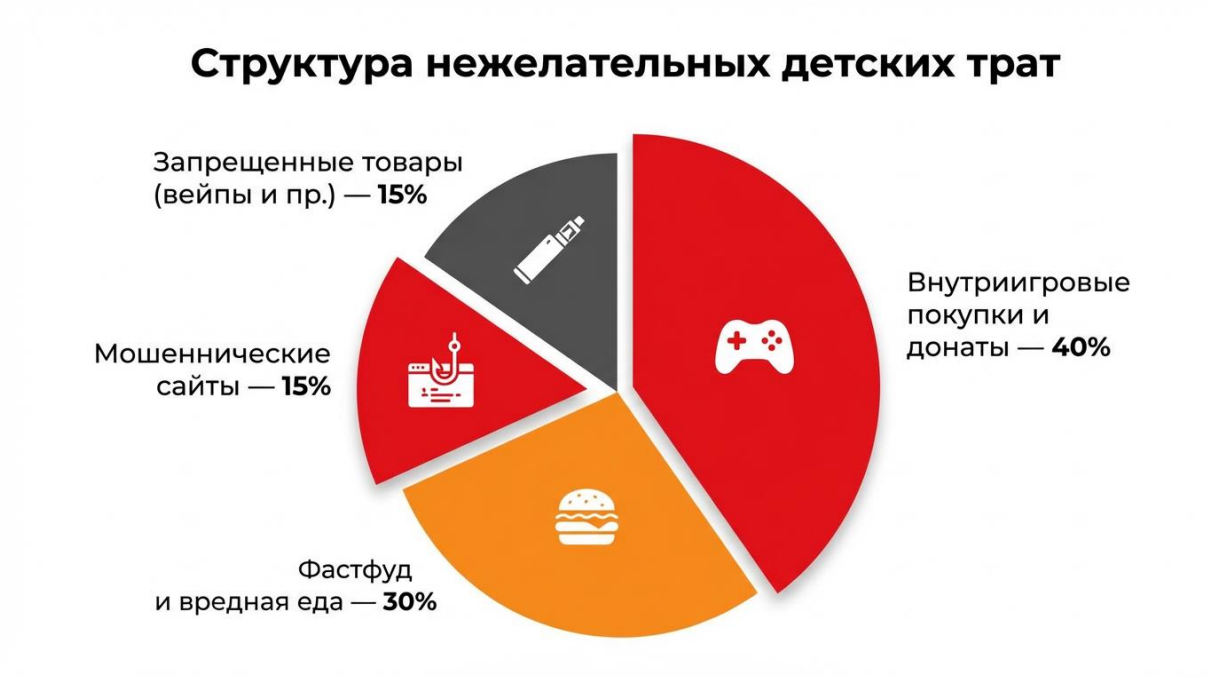


Рис. 3. Структура нежелательных детских трат

Таким образом, бесконтрольность финансовых потоков ребенка – это путь к формированию деструктивных привычек и угроза его здоровью. Внедрение превентивных мер защиты на уровне банковского процессинга, предлагаемое в проекте KidSafeKard, является необходимостью в современной социальной среде.

ГЛАВА 2. РАЗРАБОТКА ПРОЕКТА «KidSafeKard»

2.1. Концепция продукта: функционал и уникальные торговые предложения

Проект “KidSafeKard” представляет собой экосистему, состоящую из физической банковской карты и мобильного приложения с разделением на “Родитель” и “Ребенок”. В отличие от существующих на рынке аналогов, которые выполняют функцию простого

электронного кошелька с мониторингом, наш проект позиционируется как инструмент превентивного контроля.

Основная идея заключается в смене схемы: переход от модели «узнал о покупке - поругал» к модели «настроил правила - предотвратил нежелательную трату».

Ключевой функционал системы:

1. Фильтрация по MCC-кодам (White & Black Lists):

Родитель может не просто ставить лимит суммы, а управлять категориями. MCC (Merchant Category Code) - это четырехзначный номер, классифицирующий вид деятельности торговой точки.

Пример: можно разрешить траты в категории «Книжные магазины» (MCC 5942) безлимитно, но полностью заблокировать категорию «Видеоигры» (MCC 5816) или установить на нее лимит 500 рублей в месяц.

2. Система «Запрос – Одобрение» в реальном времени:

Если ребенок пытается купить товар в заблокированной категории или превышает лимит, транзакция не отклоняется молча. Ребенок получает уведомление с предложением отправить запрос родителю. Родитель в своем приложении видит: “Саша хочет купить товар в магазине «Steam» на сумму 300 р. Разрешить?”. Это учит ребенка диалогу и аргументации своих желаний.

3. Геолокационный контроль (Geo-fencing):

Возможность запретить покупки в определенных географических зонах (например, в районе неблагополучного рынка) или, наоборот, разрешить их только на территории школы и дома.

Уникальное торговое предложение (УТП):

Главным преимуществом KidSafeKard является детальный контроль. Мы предлагаем родителям максимальную точность в настройке финансовой свободы ребенка, в то время как конкуренты предлагают полную блокировку или общий лимит.

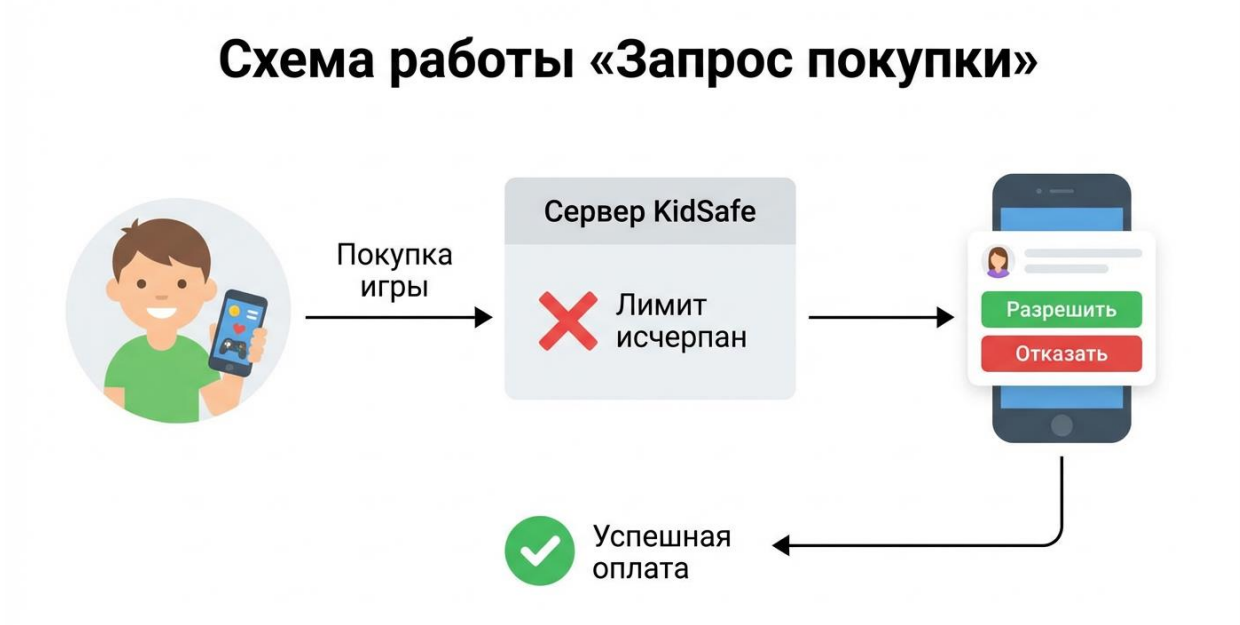


Рис.4. Схема работы

2.2. Технологический стек и архитектура приложения

Для реализации надежной и безопасной системы, работающей с финансами, был выбран современный стек технологий, актуальный для разработки ПО в 2026 году. От использования локальных баз данных (типа SQLite) было решено отказаться в пользу клиент-серверной архитектуры, так как данные должны синхронизироваться между двумя устройствами (родителя и ребенка) мгновенно.

Архитектура проекта:

1. Клиентская часть: мобильное приложение разрабатывается на фреймворке Flutter (язык Dart). Это позволяет создать единое приложение, работающее как на Android, так и на iOS, с высокой производительностью и плавными анимациями.
2. Серверная часть: логика обработки транзакций и хранения настроек реализована на языке Python с использованием фреймворка FastAPI. Python был выбран за скорость разработки, безопасность и широкие возможности работы с данными.
3. База данных: в качестве системы управления базами данных (СУБД) используется PostgreSQL. Она обеспечивает надежность транзакций (ACID), что очень важно, когда речь идет о деньгах.

В базе хранятся: профили пользователей, история транзакций, таблицы MCC-кодов, списки разрешенных/запрещенных категорий.

Алгоритм работы: при попытке совершения транзакции запрос поступает на наш сервер. Программа проверяет ID пользователя, сверяет MCC-код магазина с «Черным списком» родителя и проверяет остаток лимита для данной категории. Если условия соблюдены - сервер отправляет ответ «Одобрено». Весь процесс занимает менее 100 миллисекунд.

2.3. Дизайн интерфейса и пользовательский опыт (UI/UX)

Дизайн приложения KidSafeKard разработан с учетом психологии двух разных групп пользователей. Приложение имеет два режима отображения, которые переключаются при входе в аккаунт.

Интерфейс «Родитель»:

- Цветовая гамма: Сдержанная, используются темно-синие и белые тона, ассоциирующиеся с надежностью и спокойствием.
- Главный экран: Панель управления с графиками расходов ребенка за неделю.
- Ключевые функции: Кнопка «Настройки доступа», где представлен список категорий товаров с ползунками (Allow/Deny) и полями для ввода лимитов. Интерфейс интуитивно понятен и не перегружен лишними деталями.

Интерфейс «Ребенок»:

- Цветовая гамма: Яркая, с использованием градиентов (фиолетовый, оранжевый).
- Главный экран: Текущий баланс отображается крупным шрифтом. Под ним – кнопка “Попросить денег” и “Копилка”.
- Визуализация: при попытке купить что-то запрещенное появляется дружелюбный робот, который объясняет причину отказа, а не просто пишет “Ошибка”. Это снижает стресс сс у ребенка.

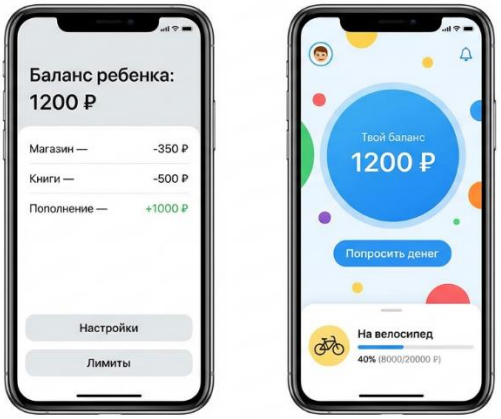


Рис.5. Интерфейс приложения

2.4. Результаты социологического опроса целевой аудитории

Для подтверждения актуальности разработки и улучшения функционала было проведено анкетирование. В опросе приняли участие 49 человек: 27 родителей и 22. Опрос проводился с использованием Google Forms.

Ключевые результаты:

1. Страхи родителей: 70% опрошенных родителей подтвердили, что боятся давать ребенку карту из-за риска бесцельного расходования средств. Самый популярный страх (60%) - донаты в онлайн-играх.
2. Потребность в контроле: на вопрос «Хотели бы вы запрещать конкретные категории покупок (например, фастфуд), оставляя доступ к другим?» 85% родителей ответили «Да» или «Скорее да». Это подтверждает нашу гипотезу о востребованности функции фильтрации по MCC-кодам.
3. Мнение детей: среди подростков 40% признались, что хотя бы раз тратили карманные деньги на вещи, о покупке которых жалели через пару дней. При этом 65% детей положительно оценили идею приложения, если оно поможет им накопить на крупную мечту (телефон, консоль) путем ограничения мелких трат.

Вывод по главе 2: разработанная концепция KidSafeKard технически реализуема на базе стека Python + PostgreSQL. Интерфейсные решения учитывают возрастные особенности, а результаты опроса доказывают, что продукт решает реальные проблемы как родителей, так и детей. Проект готов к стадии прототипирования

ЗАКЛЮЧЕНИЕ

В ходе выполнения командного проекта на тему «KidSafeKards» - детская карта с расширенным родительским контролем” была проведена комплексная работа по анализу проблем детской финансовой безопасности и разработке технического решения для их устранения.

Основные итоги работы:

1. Теоретический анализ. Изучение литературы и рыночной ситуации показало, что в условиях цифровой экономики 2026 года традиционные методы контроля расходов (наличные деньги) утратили эффективность. Существующие банковские продукты для детей, несмотря на свою популярность, имеют существенные недостатки: они работают по принципу постфактум-уведомлений и предлагают лишь базовые настройки лимитов, что не защищает ребенка от импульсивных трат и покупки нежелательных товаров.
2. Подтверждение актуальности. Проведенный социологический опрос подтвердил гипотезу исследования. 65% опрошенных родителей выразили готовность использовать инструменты для блокировки конкретных категорий покупок (например, донатов в играх или фастфуда), что доказывает востребованность функционала KidSafeKard.
3. Техническая реализация. Была разработана концепция и архитектура программного комплекса, состоящего из мобильного приложения на фреймворке Flutter и серверной части на языке Python с использованием базы данных PostgreSQL.
4. Практическая значимость. Главным результатом проекта стала разработка системы фильтрации транзакций по MCC-кодам. Данное решение позволяет гибко настраивать финансовую свободу ребенка: запрещать вредные категории, но оставлять доступ к полезным (транспорт, книги, питание), тем самым формируя навыки осознанного потребления без тотальных запретов.

Таким образом, цель проекта достигнута. Разработанный прототип «KidSafeKard» решает проблему «слепой зоны» в финансовых отношениях детей и родителей. В дальнейшем проект может быть масштабирован: планируется внедрение элементов искусственного интеллекта для анализа поведенческих паттернов ребенка и автоматических советов по накоплению средств. Проект готов к демонстрации и может рассматриваться как база для создания полноценного стартапа в сфере финансов.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

Нормативно-правовые акты:

1. «Уголовный кодекс Российской Федерации» от 13.06.1996 N 63-ФЗ (ред. от 29.12.2025) // Собрание законодательства РФ.
2. Стратегия повышения финансовой грамотности в Российской Федерации на 2024–2030 годы. - М.: Министерство финансов РФ, 2023.

Электронные ресурсы:

1. Документация к библиотеке FastAPI [Электронный ресурс]. - Режим доступа: https://fastapi.tiangolo.com/ru/tutorial/ (дата обращения: 15.01.2026).
2. Официальная документация PostgreSQL [Электронный ресурс]. - Режим доступ: (дата обращения: 20.01.2026).
3. Статистика использования безналичных платежей в РФ [Электронный ресурс] // Сайт Центрального Банка РФ. - Режим доступа (дата обращения: 10.01.2026).

ПРИЛОЖЕНИЕ А. МАКЕТЫ ИНТЕРФЕЙСА ПРИЛОЖЕНИЯ «KidSafeKard»

Рис. 6. Главный экран приложения для роли «Родитель» (Панель мониторинга)

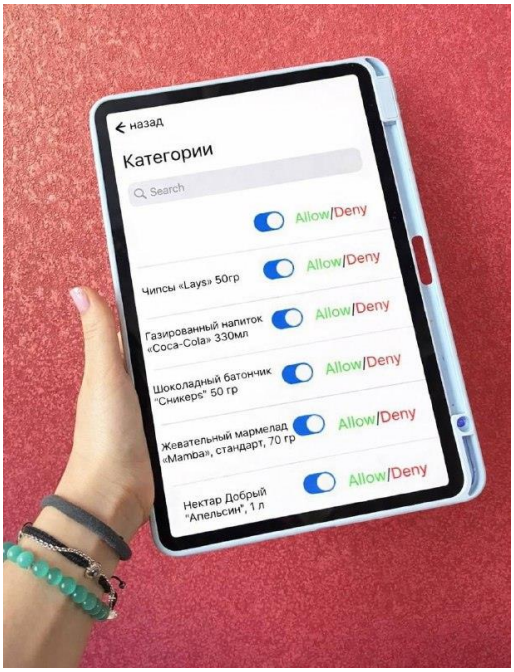


Рис. 7. Экран настройки ограничений (Уникальная функция MCC-фильтра)

ПРИЛОЖЕНИЕ Б. РЕЗУЛЬТАТЫ АНКЕТИРОВАНИЯ

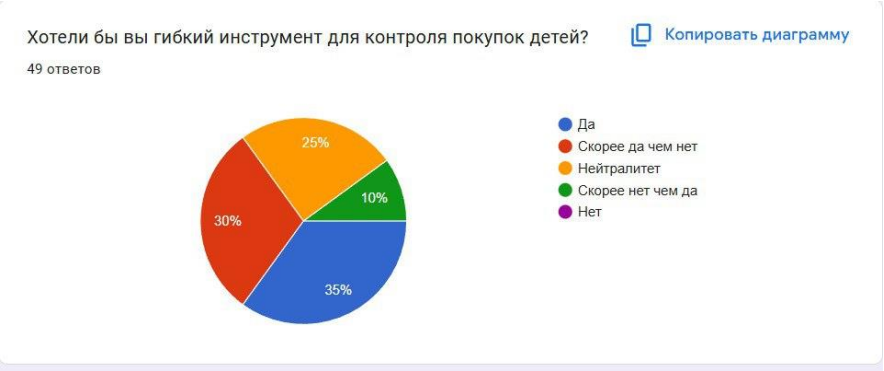


Рис 8. Ответ на вопрос: «Хотели бы вы гибкий инструмент для контроля покупок детей»

Таблица 2. Основные страхи родителей относительно детских финансов (по результатам)

Чего боятся родители?

№	Категория страха	Процент
1	Донаты в онлайн-играх и приложениях	60%
2	Покупка вредной еды (фастфуд, энергетики)	55%
3	Мошенничество в интернете	45%
4	Покупка вейпов/сигарет	30%
5	Потеря денег из-за невнимательности	20%

КОМПЛЕКС СИСТЕМ ДЛЯ ПРОИЗВОДСТВА ИЗДЕЛИЙ ПО ТЕХНОЛОГИИ FFF ИЗ ГРАНУЛ

Лобачев Константин Сергеевич

Москва г, ГБОУ Школа №1517, 11 класс

Секция: Робототехника и комплексная автоматизация

Научный руководитель: Смирнов Иван Алексеевич, преподаватель ГБОУ Школа №1517

АННОТАЦИЯ

В этом проекте я хочу изменить подход к печати филаментом. Сейчас к точности изготовления пластика для 3d печати предъявляются строгие требования (отклонение диаметра не более 0,05мм, а малейшая овальность считается недопустимой). Чтобы снизить требования к филаменту я разработал 2D датчик геометрии прутка, который позволяет принтеру автоматический корректировать подачу и компенсировать не идеальность филамента. Важной особенностью моего датчика является возможность учитывать овальность филамента. В результате, принтер, оборудованный таким датчиком, может печатать пластиком диаметром (1,650 ± 0.263) мм с таким же качеством, как и заводским (мой софт для поддержки датчика представляет собой написанный на языке Python модуль для прошивки Klipper). Такое упрощение требований к филаменту позволило мне разработать экструзионную линию более чем в два раза дешевле аналогов. Я разработал собственное ПО для линии, что позволило реализовать ряд важных защит.

ВВЕДЕНИЕ

Типовая экструзионная линия для производства филамента состоит из экструдера (который плавит гранулы пластика), системы охлаждения расплава, механизма протяжки (обеспечивающего движение филамента по линии), датчика геометрии (по которому корректируется скорость протяжки), намотчика и измерителя натяжения, по данным с которого корректируется скорость намотчика.

Диаметр сопла экструдера больше конечного диаметра прутка, филамент, проходя по линии растягивается до нужного диаметра. Шнек перемещает пластик из зоны загрузки (под бункером) к соплу.

Нагревается только зона плавления, зона загрузки остается холодной. Часто зона загрузки специально охлаждается, чтобы пластик не начал плавиться и слипаться на входе в экструдер. Underground 3D использует для этого систему водяного охлаждения из нескольких витков медной трубки, обернутых вокруг экструдера:

Михаил (автор меняет название канала чаще, чем выпускает видео, поэтому название может поменяться) использует два больших радиатора, приваренных к экструдеру.

Охлаждение

Воздушное	Водяное
Разогретый пруток обдувается потоками воздуха от вентиляторов	Пруток после растягивания попадает в воду и быстро охлаждается, что важно для пластиков, склонных к кристаллизации (например, PET)

Измерение диаметра и актуальность проекта

В этом проекте я хочу изменить подход к печати филаментом. Сейчас к точности диаметра предъявляются строгие требования, а малейшая овальность считается недопустимой. Такие требования делают промышленные экструзионные линии и сам филамент намного дороже, что уменьшает рентабельность 3D печати как способа производства.

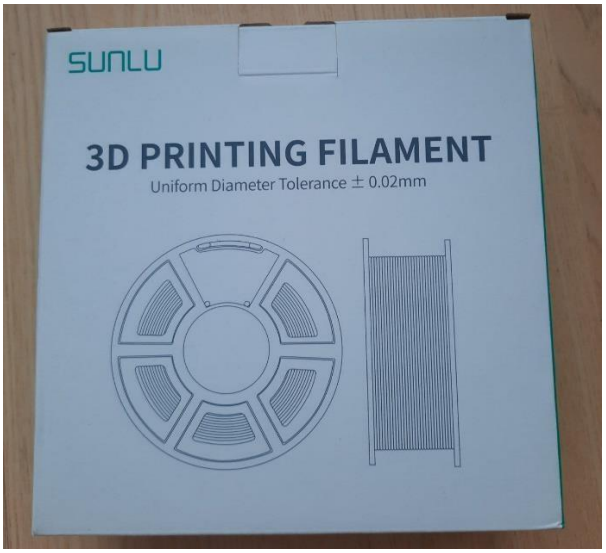


Рис. 1. Пластик для 3D печати

Я предлагаю снизить требования к допускам и перестать считать овальный филамент браком. Идея по реализации этого без ухудшения качества печати состоит в том, чтобы оборудовать принтер 2D датчиком сечения филамента и автоматически корректировать по его показаниям поток.

Следовательно, появляются особые требования к датчику обратной связи по геометрии прутка, работающего на экструзионной линии: он должен уметь работать с геометрией овала, не допускать превышения по большей полуоси, чтобы пруток не застрял в принтере, и сильного уменьшения меньшой полуоси. Я не нашел таких датчиков, поэтому разработал собственный на базе уже существующих датчиков диаметра круглого прутка.

Механизм протяжки

Механизм протяжки должен хорошо цеплять пруток тянуть филамент и практически через всю линию, мотор протяжки должен иметь стабильную регулируемую скорость.

Undeground 3D использовал для механизма полиуретановые ролики от погрузчика.

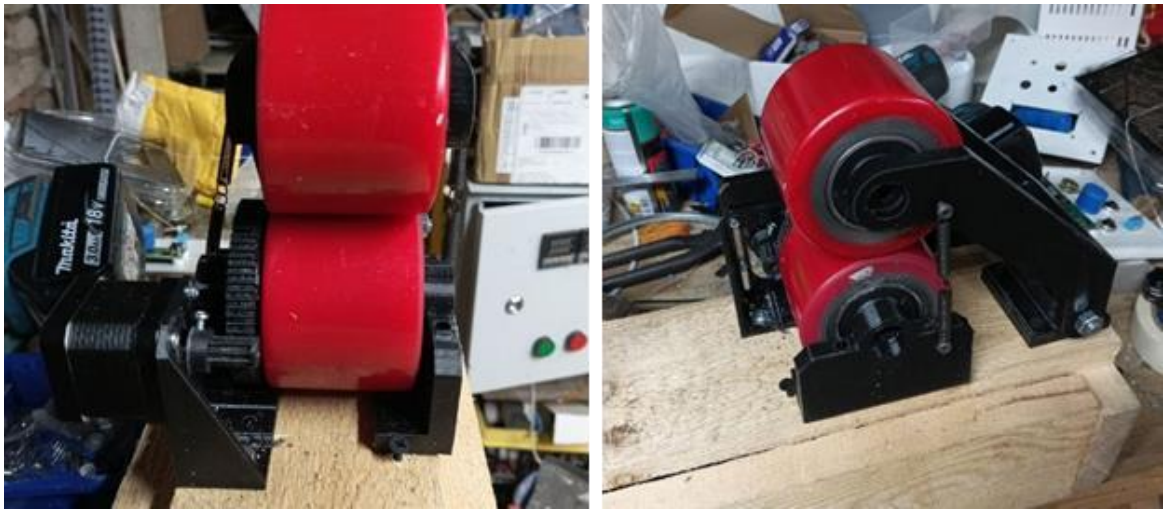


Рис. 2. Механизм

А в видео с канала «diy chen» автор нашел более простой способ: он использовал боуден фидор от 3д принтера. В моем случае такой фидер, в отличии от роликов может оставлять следы на пластике, которые могут помешать корректной работе датчика принтера.

Классический алгоритм работы протяжки: если диаметр филамента больше номинального, то скорость протяжки увеличивается, чтобы сильнее растягивать пластик на выходе из экструдера, если диаметр филамента меньше номинального, то скорость протяжки уменьшается, чтобы меньше растягивать пластик на выходе из экструдера. В данном проекте алгоритм будет усложнен в связи с возможной овальностью прутка.

Система намотки

Система намотки состоит из датчика натяжения филамента по показаниям которого корректируется скорость вращения наматываемой катушки и намотчика, вращающего катушку и укладывающего на нее готовый филамент.

Устройство намотчика (на примере перемотчика от Undeground 3D).

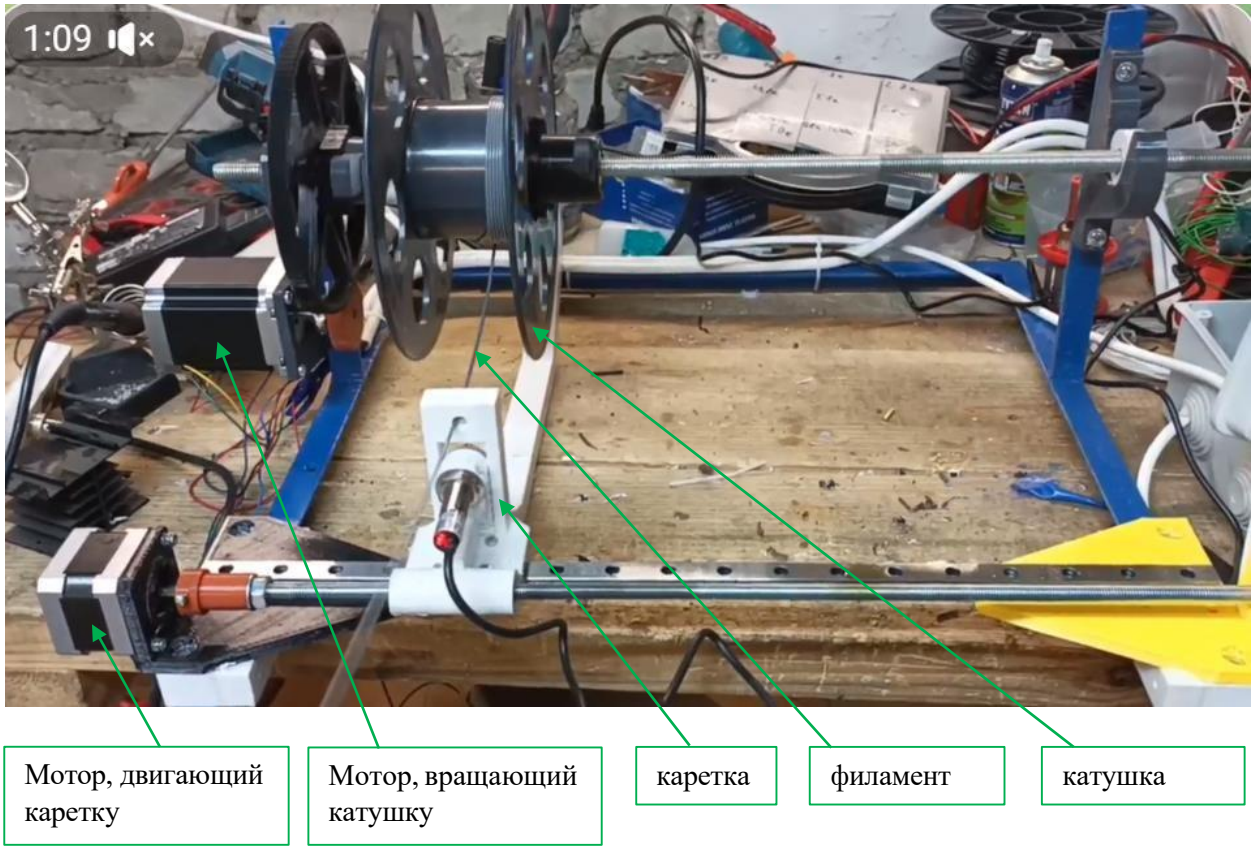


Рис. 3. Перемотчик от Underground 3D

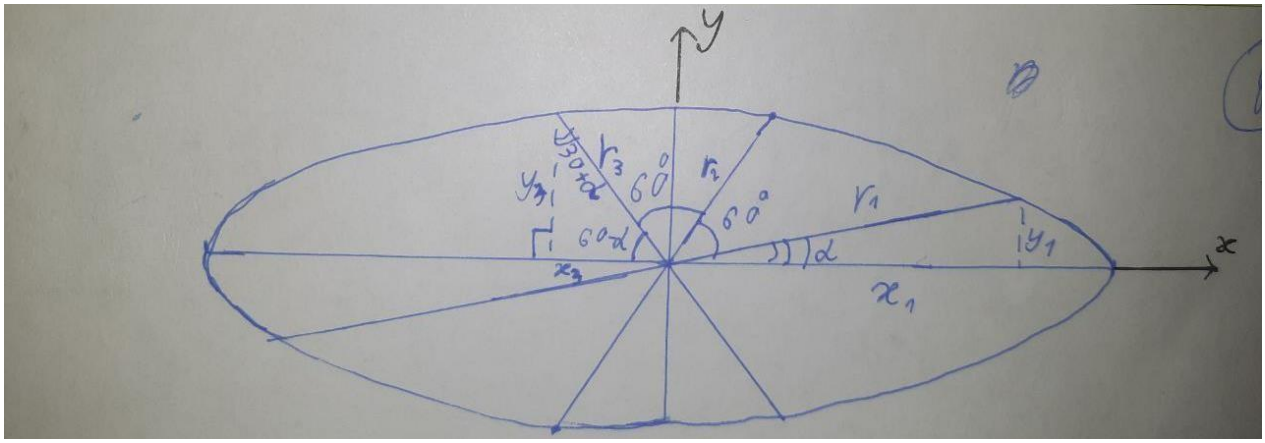
В каретке есть отверстие, через которое проходит пруток. Каретка, перемещаясь и направляя пруток, плотно укладывает ветки на катушку.

МЕТОДИКА ВЫПОЛНЕНИЯ РАБОТЫ

Датчик для принтера. Расчет геометрии.

Создание датчика я начал с вывода математических формул, позволяющих вычислять полуоси эллипса по трем его диаметрам. В основу легло уравнение

эллипса с центром в начале координат: $$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$



$x_1^2 + y_1^2 = r_1^2$ (теорема Пифагора)

$y_1^2 = r_1^2 - x_1^2$

$$\frac{r_1^2 - x_1^2}{b^2} + \frac{x_1^2}{a^2} = 1$$

$x_1 = r_1 \cdot \cos\alpha$

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

$$\Rightarrow \frac{r_1^2 - r_1^2 \cdot \cos^2\alpha}{b^2} + \frac{r_1^2 \cdot \cos^2\alpha}{a^2} = 1$$

$$\frac{r_1^2}{b^2} - \frac{r_1^2 \cdot \cos^2\alpha}{b^2} + \frac{r_1^2 \cdot \cos^2\alpha}{a^2} = 1$$

$$\frac{r_1^2}{b^2} - \frac{r_1^2 \cdot \cos^2\alpha}{b^2} + \frac{r_1^2 \cdot \cos^2\alpha}{a^2} = 1 \quad \Big| : (r_1^2)$$

$$\frac{1}{b^2} - \frac{\cos^2\alpha}{b^2} + \frac{\cos^2\alpha}{a^2} = \frac{1}{r_1^2}$$

$$-\frac{\cos^2\alpha}{b^2} + \frac{\cos^2\alpha}{a^2} = \frac{1}{r_1^2} - \frac{1}{b^2}$$

$$\cos^2\alpha \cdot \left(\frac{1}{a^2} - \frac{1}{b^2}\right) = \frac{1}{r_1^2} - \frac{1}{b^2}$$

$$\cos^2\alpha = \frac{\frac{1}{r_1^2} - \frac{1}{b^2}}{\frac{1}{a^2} - \frac{1}{b^2}} \quad \Big\lfloor a^2b^2$$

$$\cos^2\alpha = \frac{\frac{a^2b^2}{r_1^2} - a^2}{b^2 - a^2} \quad \Big\lfloor r_1^2$$

$$\boxed{\cos^2\alpha = \frac{a^2b^2 - a^2r_1^2}{r_1^2b^2 - a^2r_1^2}}$$

$$\sin^2\alpha + \cos^2\alpha = 1$$

$$\sin^2\alpha = 1 - \cos^2\alpha$$

$$\sin^2\alpha = \frac{r_1^2b^2 - a^2r_1^2 - a^2b^2 + a^2r_1^2}{r_1^2b^2 - a^2r_1^2}$$

$$\boxed{\sin^2\alpha = \frac{r_1^2b^2 - a^2b^2}{r_1^2b^2 - a^2r_1^2}}$$

$x_3^2+y_3^2=r_3^2$ (теорема Пифагора)

$y_3^2=r_3^2-x_3^2$

$$\frac{x_3^2}{a^2}+\frac{r_3^2-x_3^2}{b^2}=1 \quad (ф.1)$$

$x_3=r_3\cdot\sin(30+\alpha)$

$x_3=r_3\cdot(\cos 30^\circ\cdot\sin\alpha+\cos\alpha\cdot\sin 30^\circ)$

$x_3=r_3\cdot\left(\frac{\sqrt{3}}{2}\cdot\sin\alpha+\frac{1}{2}\cdot\cos\alpha\right)$

$x_3=r_3\cdot 0{,}5\cdot(\sqrt{3}\cdot\sin\alpha+\cos\alpha)$

$$\Rightarrow \frac{r_3^2\cdot 0{,}25\cdot(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{a^2}+\frac{r_3^2-r_3^2\cdot 0{,}25(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{b^2}=1 \quad \left|\cdot\left(\frac{4}{r_3^2}\right)\right.$$

$$\frac{(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{a^2}+\frac{4-(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{b^2}=\frac{4}{r_3^2}$$

$$\frac{(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{a^2}+\frac{4}{b^2}-\frac{(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{b^2}=\frac{4}{r_3^2}$$

$$\frac{(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{a^2}-\frac{(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2}{b^2}=\frac{4}{r_3^2}-\frac{4}{b^2}$$

$$(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2\cdot\left(\frac{1}{a^2}-\frac{1}{b^2}\right)=\frac{4}{r_3^2}-\frac{4}{b^2}$$

$$(\sqrt{3}\cdot\sin\alpha+\cos\alpha)^2=\frac{4a^2b^2-4a^2r_3^2}{r_3^2b^2-a^2r_3^2}$$

$$3\cdot\sin^2\alpha+\sin\alpha\cdot\cos\alpha\cdot2\sqrt{3}+\cos^2\alpha=$$

$$=\frac{4a^2b^2-4a^2r_3^2}{r_3^2b^2-a^2r_3^2}$$

$$\sin\alpha\cdot\cos\alpha\cdot2\sqrt{3}=\frac{4a^2b^2-4a^2r_3^2}{r_3^2b^2-a^2r_3^2}-$$

$$-3\cdot\sin^2\alpha-\cos^2\alpha$$

$$\sin^2\alpha\cdot\cos^2\alpha\cdot12=\left(\frac{4a^2b^2-4a^2r_3^2}{r_3^2b^2-a^2r_3^2}-3\cdot\right.$$

$$\left.\cdot\sin^2\alpha-\cos^2\alpha\right)^2.$$

Подставлю ранее полученные выражения sin^2(a) и cos^2(a)

$$\frac{a^2b^2-a^2r_1^2}{r_1^2b^2-a^2r_1^2}\cdot\frac{r_1^2b^2-a^2b^2}{r_1^2b^2-a^2r_1^2}\cdot 12=\left(\frac{4a^2b^2-4a^2r_3^2}{r_3^2b^2-a^2r_3^2}-\right.$$

$$\left.-\frac{3r_1^2b^2-3b^2a^2}{r_1^2b^2-a^2r_1^2}-\frac{a^2b^2-a^2r_1^2}{r_1^2b^2-a^2r_1^2}\right)^2 \quad \Big|\cdot(b^2-a^2)^2$$

$$\frac{a^2b^2-a^2r_1^2}{r_1^2}\cdot\frac{r_1^2b^2-a^2b^2}{r_1^2}\cdot 12=\left(\frac{4a^2b^2-4a^2r_3^2}{r_3^2}-\right.$$

$$\left.-\frac{3r_1^2b^2-3b^2a^2}{r_1^2}+\frac{a^2b^2-a^2r_1^2}{r_1^2}\right)^2 \quad \Big|\cdot(r_1^4\cdot r_3^4)$$

$$(a^2b^2-a^2r_1^2)\cdot(r_1^2b^2-a^2b^2)\cdot 12r_3^4=(4a^2b^2r_1^2-4a^2r_3^2r_1^2-$$

$$-r_3^2\cdot(3r_1^2b^2-3\cdot a^2b^2+a^2b^2-a^2r_1^2))^2$$

$$(a^2b^2-a^2r_1^2)\cdot(r_1^2b^2-a^2b^2)\cdot 12r_3^4=(4a^2b^2r_1^2-4a^2r_3^2r_1^2-$$

$$-r_3^2\cdot(3r_1^2b^2-2a^2b^2-a^2r_1^2))^2$$

$$(r_1^2a^2b^4-a^4b^4-r_1^4a^2b^2+r_1^2a^4b^2)\cdot 12r_3^4=(4a^2b^2r_1^2-$$

$$-4a^2r_3^2r_1^2-3r_1^2r_3^2b^2+2r_3^2a^2b^2+r_1^2r_3^2a^2)^2$$

$$(a^2b^2-a^2r_1^2)\cdot(r_1^2b^2-a^2b^2)\cdot 12r_3^4=(4a^2b^2r_1^2-4a^2r_3^2\cdot$$

$$\cdot r_1^2 - r_3^2 \cdot (3r_1^2b^2 - 2a^2b^2 - a^2r_1^2))^2$$

$$(r_1^2a^2b^4 - a^4b^4 - r_1^4a^2b^2 + r_1^2a^4b^2) \cdot 12r_3^4 = (4r_1^2a^2b^2 -$$

$$- 4r_1^2r_3^2a^2 - 3r_1^2r_3^2b^2 + 2r_3^2a^2b^2 + r_1^2r_3^2a^2)^2$$

$$12r_1^2r_3^4a^2b^4 - 12r_3^4a^4b^4 - 12r_1^4r_3^4a^2b^2 + 12r_1^2r_3^4a^4b^2 =$$

$$= (4r_1^2a^2b^2 - 3r_1^2r_3^2a^2 - 3r_1^2r_3^2b^2 + 2r_3^2a^2b^2)^2$$

$$12r_1^2r_3^4a^2b^4 - 12r_3^4a^4b^4 - 12r_1^4r_3^4a^2b^2 + 12r_1^2r_3^4a^4b^2 =$$

$$= 16r_1^4a^4b^4 + 9r_1^4r_3^4a^4 + 9r_1^4r_3^4b^4 + 4r_3^4a^4b^4 -$$

$$- 24r_1^4r_3^2a^4b^2 - 24r_1^4r_3^2a^2b^4 + 16r_1^2r_3^2a^4b^4 +$$

$$+ 18r_1^4r_3^4a^2b^2 - 12r_1^2r_3^4a^4b^2 - 12r_1^2r_3^4a^2b^4$$

$$0 = 16r_1^4a^4b^4 + 9r_1^4r_3^4a^4 + 9r_1^4r_3^4b^4 + 4r_3^4a^4b^4 -$$

$$- 24r_1^4r_3^2a^4b^2 - 24r_1^4r_3^2a^2b^4 + 16r_1^2r_3^2a^4b^4 +$$

$$+ 18r_1^4r_3^4a^2b^2 - 12r_1^2r_3^4a^4b^2 - 12r_1^2r_3^4a^2b^4 -$$

$$- 12r_1^2r_3^4a^2b^4 + 12r_3^4a^4b^4 + 12r_1^4r_3^4a^2b^2 - 12r_1^2r_3^4a^4b^2$$

$$0 = (16r_1^4 + 4r_3^4 + 16r_1^2r_3^2 + 12r_3^4) \cdot a^4b^4 +$$

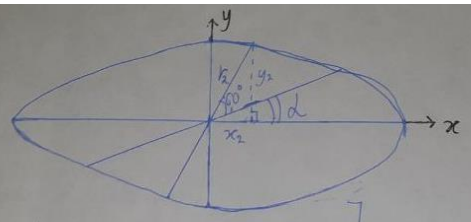
$$+ (-24r_1^4r_3^2 - 12r_1^2r_3^4 - 12r_1^2r_3^4) \cdot a^4b^2 +$$

$$+ (-24r_1^4r_3^2 - 12r_1^2r_3^4 - 12r_1^2r_3^4) \cdot a^2b^4 +$$

$$+(18r_1^4r_3^4+12r_1^4r_3^4)\cdot a^2b^2+9r_1^4r_3^4a^4+9r_1^4r_3^4b^4$$

$$\boxed{\begin{aligned}0=&(16r_1^4+16r_3^4+16r_1^2r_3^2)\cdot a^4b^4+\\&+(-24r_1^4r_3^2-24r_1^2r_3^4)\cdot a^4b^2+\\&+(-24r_1^4r_3^2-24r_1^2r_3^4)\cdot a^2b^4+30r_1^4r_3^4a^2b^2+\\&+9r_1^4r_3^4a^4+9r_1^4r_3^4b^4\end{aligned}}$$

Уравнение 1



$y_2=r_2\cdot\sin(60+\alpha)$

$r_2^2=y_2^2+x_2^2$ (теорема Пифагора)

$x_2^2=r_2^2-y_2^2=r_2^2-r_2^2\cdot\sin^2(\alpha+60)$

$\frac{x^2}{a^2}+\frac{y^2}{b^2}=1$ (ф.1)

$$\rightarrow \frac{r_2^2-r_2^2\cdot\sin^2(\alpha+60)}{a^2}+\frac{r_2^2\cdot\sin^2(\alpha+60)}{b^2}=1$$

$$\frac{1-\sin^2(\alpha+60)}{a^2}+\frac{\sin^2(\alpha+60)}{b^2}=\frac{1}{r_2^2}$$

$$\frac{1}{a^2}-\frac{\sin^2(\alpha+60)}{a^2}+\frac{\sin^2(\alpha+60)}{b^2}=\frac{1}{r_2^2}$$

$$\sin^2(\alpha+60^\circ)\cdot\left(\frac{1}{b^2}-\frac{1}{a^2}\right)=\frac{1}{r_2^2}-\frac{1}{a^2}$$

$$(\sin\alpha\cdot\cos 60^\circ+\cos\alpha\cdot\sin 60^\circ)^2\cdot\left(\frac{1}{b^2}-\frac{1}{a^2}\right)=\frac{1}{r_2^2}-\frac{1}{a^2}$$

$$(\sin\alpha+\sqrt{3}\cdot\cos\alpha)^2\cdot\left(\frac{1}{b^2}-\frac{1}{a^2}\right)=\frac{4}{r_2^2}-\frac{4}{a^2}$$

$$(\sin\alpha+\sqrt{3}\cdot\cos\alpha)^2=\frac{\frac{4}{r_2^2}-\frac{4}{a^2}}{\frac{1}{b^2}-\frac{1}{a^2}}\quad\Big|\,a^2b^2$$

$$(\sin\alpha+\sqrt{3}\cdot\cos\alpha)^2=\frac{\frac{4a^2b^2}{r_2^2}-4b^2}{a^2-b^2}$$

$$(\sin\alpha+\sqrt{3}\cdot\cos\alpha)^2=\frac{4a^2b^2-4r_2^2b^2}{a^2r_2^2-r_2^2b^2}$$

$$\sin^2\alpha+2\sqrt{3}\cdot\cos\alpha\cdot\sin\alpha+3\cdot\cos^2\alpha=$$

$$=\frac{4a^2b^2-4r_2^2b^2}{a^2r_2^2-r_2^2b^2}$$

$$2\sqrt{3}\cdot\cos\alpha\cdot\sin\alpha=\frac{4a^2b^2-4r_2^2b^2}{a^2r_2^2-r_2^2b^2}-$$

$$-\sin^2\alpha-3\cdot\cos^2\alpha$$

$$12\cdot\cos^2\alpha\cdot\sin^2\alpha=\left(\frac{4a^2b^2-4r_2^2b^2}{a^2r_2^2-r_2^2b^2}-\right.$$

$$\left.-\sin^2\alpha-3\cdot\cos^2\alpha\right)^2$$

Подставлю ранее полученные выражения sin^2(a) и cos^2(a)

$$\frac{a^2b^2-a^2r_1^2}{r_1^2b^2-a^2r_1^2}\cdot\frac{r_1^2b^2-a^2b^2}{r_1^2b^2-a^2r_1^2}\cdot 12=\left(\frac{4a^2b^2-4r_2^2b^2}{a^2r_2^2-r_2^2b^2}-\frac{r_1^2b^2-a^2b^2}{r_1^2b^2-a^2r_1^2}-3\cdot\frac{a^2b^2-a^2r_1^2}{r_1^2b^2-a^2r_1^2}\right)^2 \quad \Big|\cdot(b^2-a^2)^2$$

$$\frac{a^2b^2-a^2r_1^2}{r_1^2}\cdot\frac{r_1^2b^2-a^2b^2}{r_1^2}\cdot 12=\left(\left(\frac{-4a^2b^2+4r_2^2b^2}{r_2^2}-\frac{r_1^2b^2-a^2b^2}{r_1^2}-3\cdot\frac{a^2b^2-a^2r_1^2}{r_1^2}\right)\cdot(-1)\right)^2\cdot(r_1^4\cdot r_2^4)$$

$$(a^2b^2-a^2r_1^2)\cdot(r_1^2b^2-a^2b^2)\cdot 12r_2^4=(4a^2b^2r_1^2-$$

$$-4r_1^2r_2^2\cdot b^2+r_1^2b^2r_2^2-a^2b^2r_2^2+3a^2b^2r_2^2-3a^2r_1^2r_2^2)^2$$

$$(a^2b^2-a^2r_1^2)\cdot(r_1^2b^2-a^2b^2)\cdot 12r_2^4=(4a^2b^2r_1^2-$$

$$-3r_2^2b^2r_1^2+2a^2b^2r_2^2-3a^2r_1^2r_2^2)^2$$

$$(r_1^2a^2b^4-a^4b^4-r_1^4a^2b^2+r_1^2a^4b^2)\cdot 12r_2^4=(4a^2b^2r_1^2-$$

$$-3r_2^2b^2r_1^2+2a^2b^2r_2^2-3a^2r_1^2r_2^2)(4a^2b^2r_1^2-$$

$$-3r_2^2b^2r_1^2+2a^2b^2r_2^2-3a^2r_1^2r_2^2)$$

$$12r_2^4r_1^2a^2b^4-12r_2^4a^4b^4-12r_2^4r_1^4a^2b^2+12r_2^4r_1^2a^4b^2=$$
$$=16r_1^4a^4b^4-12r_1^4r_2^2a^2b^4+8r_1^2r_2^2a^4b^4-12r_1^4r_2^2a^4b^2-$$
$$-12r_1^4r_2a^2b^4+9r_2^4r_1^4b^4-6r_1^2r_2^4a^2b^4+9r_1^4r_2^4a^2b^2+$$
$$+8r_1^2r_2^2a^4b^4-6r_1^2r_2^4a^2b^4+4r_2^4a^4b^4-6r_1^2r_2^4a^4b^2-$$
$$-12r_1^4r_2^2a^4b^2+9r_1^4r_2^4a^2b^2-6r_1^2r_2^4a^4b^2+9r_1^4r_2^4a^4$$

$$12r_2^4r_1^2a^2b^4-12r_2^4a^4b^4-12r_2^4r_1^4a^2b^2+12r_2^4r_1^2a^4b^2=$$
$$=16r_1^4a^4b^4-24r_1^4r_2^2a^2b^4+16r_1^2r_2^2a^4b^4-24r_1^4r_2^2a^4b^2+$$
$$+9r_2^4r_1^4b^4-12r_1^2r_2^4a^2b^4+18r_1^4r_2^4a^2b^2+4r_2^4a^4b^4-$$
$$-12r_1^2r_2^4a^4b^2+9r_1^4r_2^4a^4$$

$$0=16r_1^4a^4b^4-24r_1^4r_2^2a^2b^4+16r_1^2r_2^2\cdot a^4b^4-24r_1^4r_2^2\cdot$$
$$\cdot a^4b^2+9r_2^4r_1^4b^4-12r_1^2r_2^4a^2\cdot b^4+18r_1^4r_2^4a^2b^2+4r_2^4\cdot$$
$$\cdot a^4b^4-12r_1^2r_2^4a^4b^2+9r_1^4r_2^4a^4-12r_2^4r_1^2a^2b^4+$$

$$+12r_2^4a^4b^4+12r_2^4r_1^4a^2b^2-12r_2^4\cdot r_1^2a^4b^2$$

$$0=(16r_1^4+16r_1^2r_2^2+4r_2^4+12r_2^4)\cdot a^4b^4+$$
$$+(-24r_1^4r_2^2-12r_1^2r_2^4-12r_2^4r_1^2)\cdot a^2b^4+$$
$$+(-24r_1^4r_2^2-12r_1^2r_2^4-12r_2^4r_1^2)a^4b^2+$$
$$+(18r_1^4r_2^4+12r_2^4r_1^4)\cdot a^2b^2+9r_2^4r_1^4b^4+9r_1^4r_2^4a^4$$

$$0=(16r_1^4+16r_1^2r_2^2+16r_2^4)\cdot a^4b^4+$$
$$+(-24r_1^4r_2^2-24r_1^2r_2^4)\cdot a^2b^4+$$
$$+(-24r_1^4r_2^2-24r_1^2r_2^4)\cdot a^4b^2+30r_1^4r_2^4a^2b^2+$$
$$+9r_2^4r_1^4b^4+9r_1^4r_2^4a^4$$

Уравнение 2

С помощью выведенных формул и библиотеки sympy был написан код, позволяющий вычислить полуоси эллипса по трем его диаметрам в случае, если все они различны и больше 0. При равенстве всех трех диаметров эллипс считается кругом т.е. оби полуоси считаются равными половине диаметра. Случай, когда равны только 2 диаметра, также рассчитывается по своей формуле.

```python
def elips_sensor_helper(d1, d2, d3):  2 usages
    if r1 == r2:
        if r2 == r3:
            return r1, r1
        r1, r3 = r3, r1
    if r2 == r3:
        b = (0.75 * r2 ** 2 / (1 - (0.25 * r2 ** 2 / r1 ** 2))) ** 0.5
        return sorted([r1, b], reverse=True)
    # тут вычисления !!!!
```

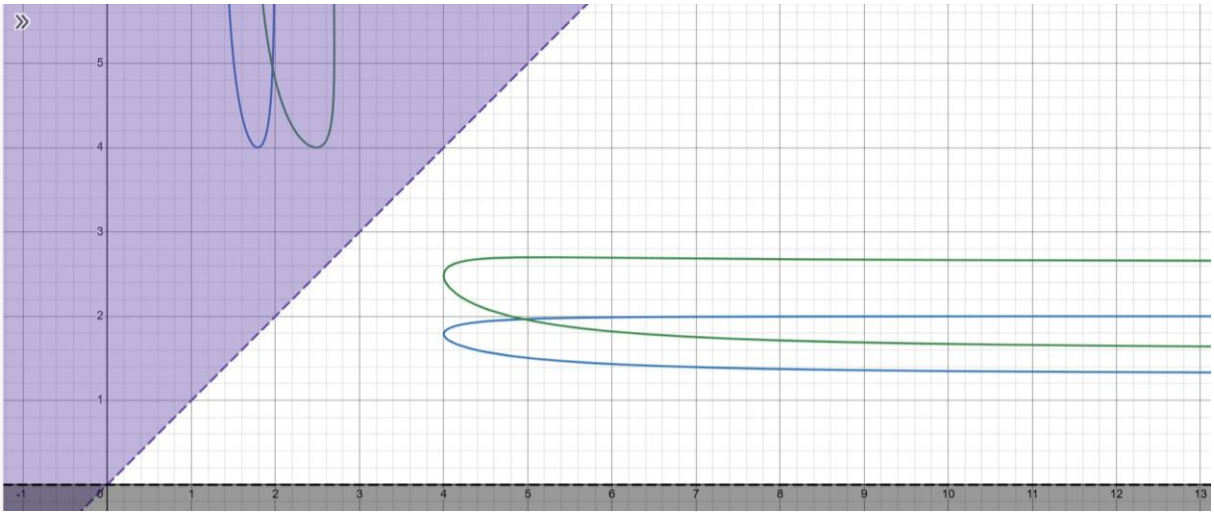


Рис. 5. Графики уравнений 1 (зеленый) и 2 (синий) при $r_1 > r_3 > r_2$.

На рисунке 5 по оси абсцисс отложены значения a, по оси ординат отложены значения b. Фиолетовым и серым цветами обозначены области, в которых значения a и b не удовлетворяют геометрическому смыслу. Точка пересечения графиков в белой области – искомое решение системы. Решим уравнение 2 как квадратное относительно $d=b^2$ (выразим больший корень)

$$0=(16r_1^4+16r_1^2r_2^2+16r_2^4)\cdot a^4b^4+$$
$$+(-24r_1^4r_2^2-24r_1^2r_2^4)\cdot a^2b^4+$$
$$+(-24r_1^4r_2^2-24r_1^2r_2^4)\cdot a^4b^2+30r_1^4r_2^4a^2b^2+$$
$$+9r_2^4r_1^4b^4+9r_1^4r_2^4a^4$$

$c=a^2$

$d=b^2$

$$0=16\cdot(r_1^4+r_1^2r_2^2+r_2^4)\cdot c^2d^2-$$
$$-24\cdot(r_1^4r_2^2+r_1^2r_2^4)\cdot cd^2-$$
$$-24\cdot(r_1^4r_2^2+r_1^2r_2^4)\cdot c^2d+30r_1^4r_2^4cd+$$
$$+9r_2^4r_1^4d^2+9r_1^4r_2^4c^2$$

$$D = \left(30 r_1^4 r_2^4 c - 24\left(r_1^4 r_2^2 + r_1^2 r_2^4\right)\cdot c^2\right)^2 - 4\cdot$$
$$\cdot\left(16\cdot\left(r_1^4 + r_1^2 r_2^2 + r_2^4\right)\cdot c^2 - 24\cdot\left(r_1^4 r_2^2 + r_1^2 r_2^4\right)\cdot c +\right.$$
$$\left. + 9 r_2^4 r_1^4\right)\cdot 9 r_1^4 r_2^4 c^2 =$$
$$= 900 r_1^8 r_2^8 c^2 - 1440\left(r_1^8 r_2^6 + r_1^6 r_2^8\right)\cdot c^3 + 576 c^4\cdot$$
$$\cdot\left(r_1^8 r_2^4 + 2 r_1^6 r_2^6 + r_1^4 r_2^8\right) - 576\left(r_1^8 r_2^4 + r_1^6 r_2^6 + r_1^4 r_2^8\right)\cdot c^4 +$$

$$d = \frac{24\left(r_1^4 r_2^2 + r_1^2 r_2^4\right)\cdot c^2 - 30 r_1^4 r_2^4 c}{32\left(r_1^4 + r_1^2 r_2^2 + r_2^4\right)\cdot c^2 - 48\left(r_1^4 r_2^2 + r_1^2 r_2^4\right) c + 18 r_1^4 r_2^4} +$$
$$+ \frac{\sqrt{576 r_1^8 r_2^8 c^2 - 576\left(r_1^8 r_2^6 + r_1^6 r_2^8\right) c^3 + 576 r_1^6 r_2^6 c^4}}{32\left(r_1^4 + r_1^2 r_2^2 + r_2^4\right) c^2 - 48\left(r_1^4 r_2^2 + r_1^2 r_2^4\right) c + 18 r_1^4 r_2^4} =$$
$$= \frac{12\left(r_1^4 r_2^2 + r_1^2 r_2^4\right) c^2 - 15 r_1^4 r_2^4 c}{16\left(r_1^4 + r_1^2 r_2^2 + r_2^4\right) c^2 - 24\left(r_1^4 r_2^2 + r_1^2 r_2^4\right) c + 9 r_1^4 r_2^4} +$$
$$+ \frac{12 r_1^3 r_2^3 c\sqrt{r_1^2 r_2^2 - \left(r_1^2 + r_2^2\right) c + c^2}}{16\left(r_1^4 + r_1^2 r_2^2 + r_2^4\right) c^2 - 24\left(r_1^4 r_2^2 + r_1^2 r_2^4\right) c + 9 r_1^4 r_2^4}$$

$$d = \frac{12\left(r_1^4 r_3^2 + r_1^2 r_3^4\right) c^2 - 15 r_1^4 r_3^4 c - 12 r_1^3 r_3^3 c\sqrt{r_1^2 r_3^2 - \left(r_1^2 + r_3^2\right) c + c^2}}{16\left(r_1^4 + r_1^2 r_3^2 + r_3^4\right) c^2 - 24\left(r_1^4 r_3^2 + r_1^2 r_3^4\right) c + 9 r_1^4 r_3^4}$$

Аналогично из уравнения 1:

$$+ 864\left(r_1^8 r_2^6 + r_1^6 r_2^8\right)\cdot c^3 - 324 r_1^8 r_2^8 c^2 =$$
$$= 576 r_1^8 r_2^8 c^2 - 576\left(r_1^8 r_2^6 + r_1^6 r_2^8\right)\cdot c^3 + 576\cdot r_1^6 r_2^6 c^4$$

При помощи выведенных формул можно подобрать максимально близкое к реальному целое значение c. Для этого в цикле перебираются возможные значения c, для каждого c вычисляется d по двум, описанным выше формулам. Значение c, при котором расхождение в значениях d будет минимально, считается верным.

Для увеличения точности нахождения c используется «коэффициент точности» (в коде обозначается precision): при расчете геометрии используется единичный отрезок равный 1мм/«коэффициент точности».

```python
class EllipseFilamentWideSensorHelper:  1 usage

    def __call__(self, last_epos=None):
        if last_epos is None:
            self.d1, self.d2, self.d3 = (self.sensor1.smoother.get_value(), self.sensor2.smoother.get_value(), self.sensor3.smoother.get_value())
        else:
            self.d1, self.d2, self.d3 = (self.sensor1.get_value(last_epos), self.sensor2.get_value(last_epos), self.sensor3.get_value(last_epos))
        r1, r3, r2 = sorted([self.d1 / 2, self.d2 / 2, self.d3 / 2], reverse=True)
        # тут вычисления !!!!
```

```
class EllipseFilamentWideSensorHelper:
    def __call__(self, last_epos=None):

        k1 = 12 * (((r1 ** 4) * (r3 ** 2)) + ((r1 ** 2) * (r3 ** 4)))
        k2 = 15 * ((r1 * r3) ** 4)
        k3 = 12 * ((r1 * r3) ** 3)
        k4 = (r1 * r3) ** 2
        k5 = (r1 ** 2) + (r3 ** 2)
        k6 = 16 * ((r1 ** 4) + k4 + (r3 ** 4))
        k7 = 2 * k1
        k8 = 9 * ((r1 * r3) ** 4)

        for c in range(int(r1 ** 2) - 1, int(max_c) + 2):
            d1 = (((p1 * (c ** 2)) - (p2 * c) + (p3 * c * ((p4 - (p5 * c) + (c ** 2)) ** 0.5))) /
                  (p6 * (c ** 2) - (p7 * c) + p8)).real
            d2 = (((k1 * (c ** 2)) - (k2 * c) - (k3 * c * ((k4 - (k5 * c) + (c ** 2)) ** 0.5))) /
                  (k6 * (c ** 2) - (k7 * c) + k8)).real
            if d1 <= 0 or d2 <= 0:
                continue
            if d1 >= c:
                continue
            delta = d1 - d2
            if delta < 0:
                delta = -delta
            if delta < perfect_delta_d:
                perfect_c = c
                perfect_delta_d = delta
                d = (d1, d2)
        self.a, self.b = (perfect_c ** 0.5) / self.precision, (((d[0] + d[1]) / 2) ** 0.5) / self.precision
        # тут вычисления !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        return (self.a * self.b) ** 0.5 * 2
```

Рис. 7. Код вычисления геометрии филамента

ПРОГРАММНОЕ ОБЕСПЕЧЕНИ

Я написал модуль для популярной прошивки 3D принтеров Klipper, позволяющий измерять 3 диаметра филамента, вычислять по этим данным геометрию пластика (по описанному выше алгоритму) и корректировать движения экструдера, чтобы принтер вне зависимости от геометрии прутка выдавливал нужное количество пластика в любой момент времени.

Для установки необходимо поместить файлы модуля в каталог klipper/klippy/extras на хост компьютере принтера, после этого станут доступны включение и настройка работы датчика геометрии филамента через файл printer.cfg.

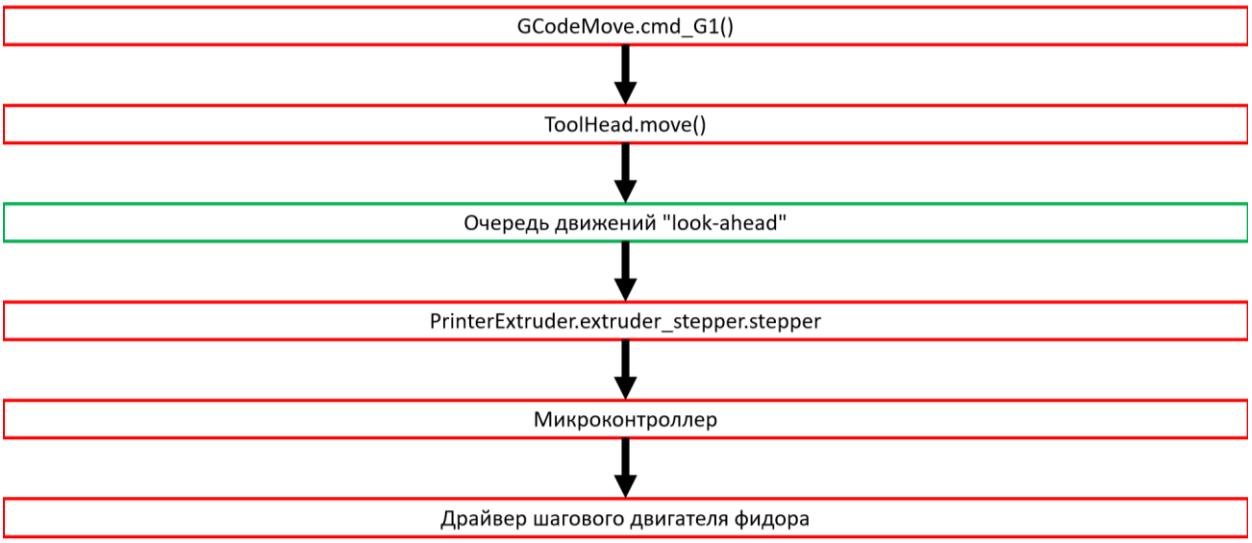


Рис.8. Упрощенная схема выполнения команды перемещения

Модуль поддерживает компенсацию расстояния между датчиком и хотэндом:

результаты вычислений сперва складываются в массив и используются только тогда, когда участок филамента, которому они соответствуют, доходит до хотэнда. Координата, к которой будет привязано значение измерения по выбору пользователя может быть взята из ToolHead или из extruder_stepper (последний вариант предпочтительнее т.к. учитывает буферизацию движений в прошивке). Аналогично работает компенсация расстояния между измерителями диаметров внутри датчика

Мой софт поддерживает два варианта корректировки движений экструдера:

1) Выполнение команды M221 для изменения движений средствами GCodeMove.
2) EMoveCuter. Это разработанный мной модификатор движений, позволяющий корректировать подачу и дробить движения (чтобы за одно движение ось экструдера перемещалась на расстояние, не превышающее заданное в конфигурации). Данный метод повышает нагрузку на электронику.

Если не использовать EMoveCuter, то при выполнении большой команды перемещения (с экструдированием) часть измеренных значений может не быть использована т.к. каждое движение корректируется с учетом геометрии одного сечения филамента.

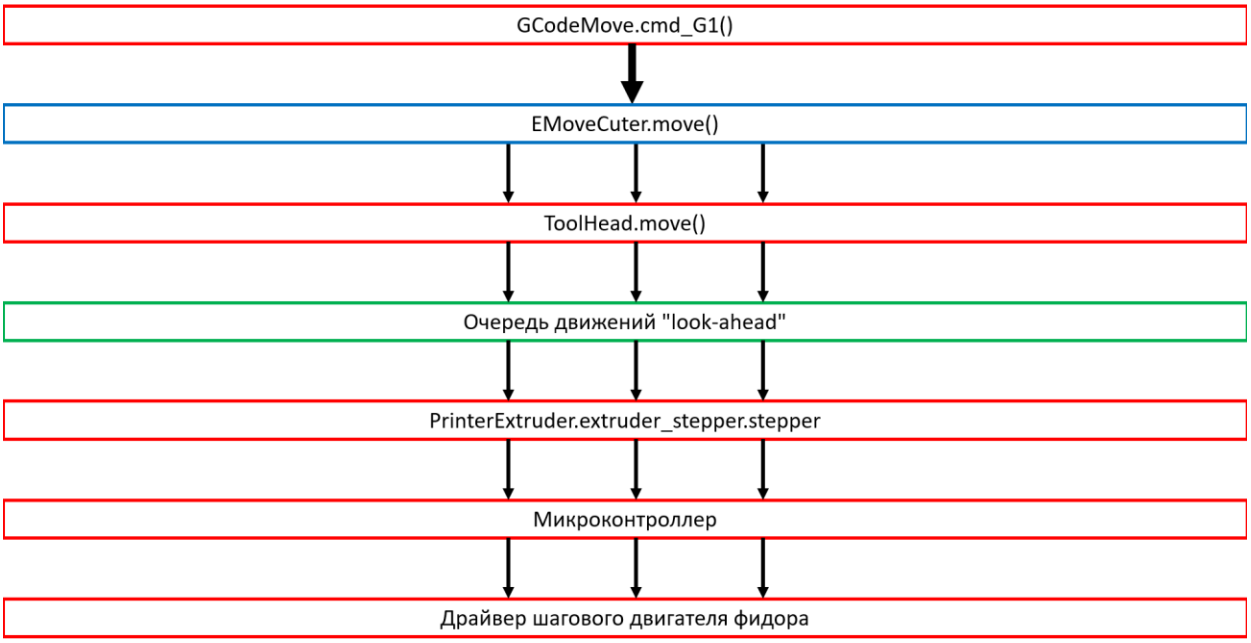


Рис. 9. Схема выполнения команды перемещения с использованием EMoveCuter

Если толщина филамента окажется вне допустимого диапазона модуль может выполнить заданный пользователем набор команд, например, поставить печать на паузу.

Команды

QUERY_FILAMENT_WIDTH: возвращает текущие показания датчика
RESET_FILAMENT_WIDTH_SENSOR: очищает массив
DISABLE_FILAMENT_WIDTH_SENSOR: отключить датчик
ENABLE_FILAMENT_WIDTH_SENSOR: включить датчик
QUERY_RAW_FILAMENT_WIDTH: возвращает текущие показания АЦП и значение RAW для калибровки
ENABLE_FILAMENT_WIDTH_LOG: Включить запись логов
DISABLE_FILAMENT_WIDTH_LOG: Выключить запись логов

Рис. 10. ПО для 3d принтера. Команды Конструкция

Реализация измерения диаметра была частично взята от датчика, поддерживаемого в Klipper:

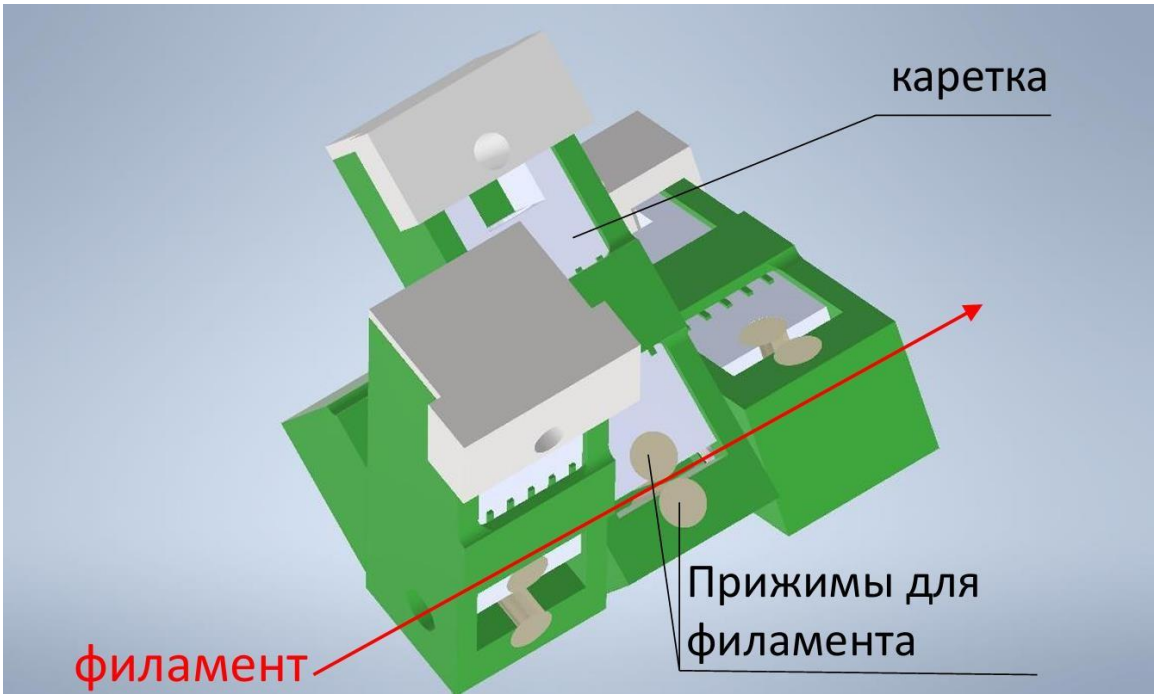


Рис. 11. Устройство датчика геометрии филамента.

Подпружиненная каретка зажимает пруток между прижимами. С помощью двух датчиков Холла определяется положение магнита, закрепленного на каретке. Таким образом, по показаниям датчиков Холла определяется каждый из трех измеряемых диаметров филамента.

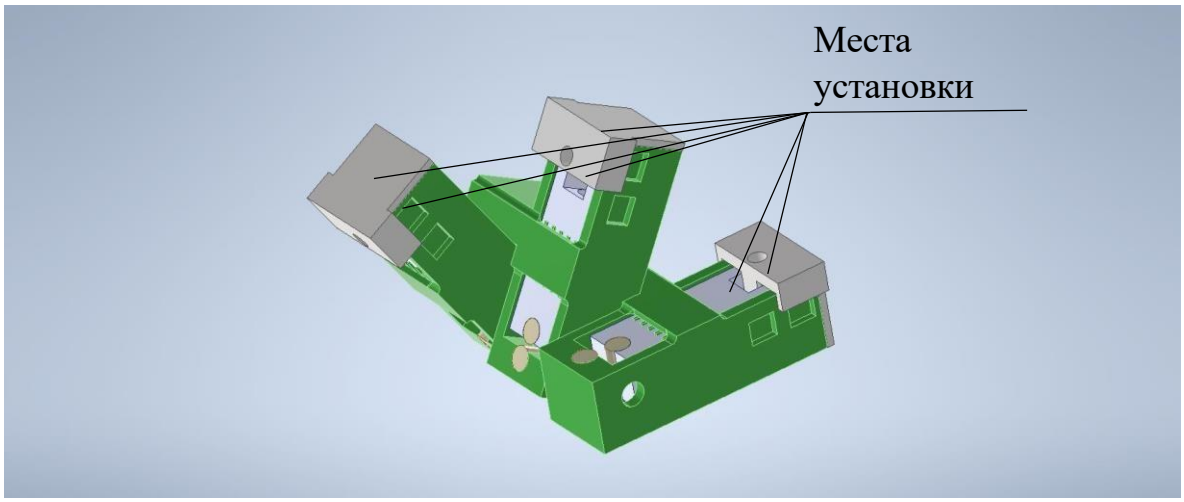


Рис. 12. Устройство датчика геометрии филамента.

ЭКСТРУДЕР

В качестве шнека был выбран самый бюджетный вариант – сверло по дереву.

В качестве двигателя был использован шаговый nema23 мотор т.к. он легко может управляться основным контроллером линии (через специальный драйвер), что позволяет реализовать ряд важных зажит, а также сделать использование линии более удобным.

Ствол был сделан из металлической трубы на 1 дюйм. Для подачи гранул в стволе сделано специальное отверстие. Ствол крепится к двигателю на шпильки М5 через промежуточную деталь.

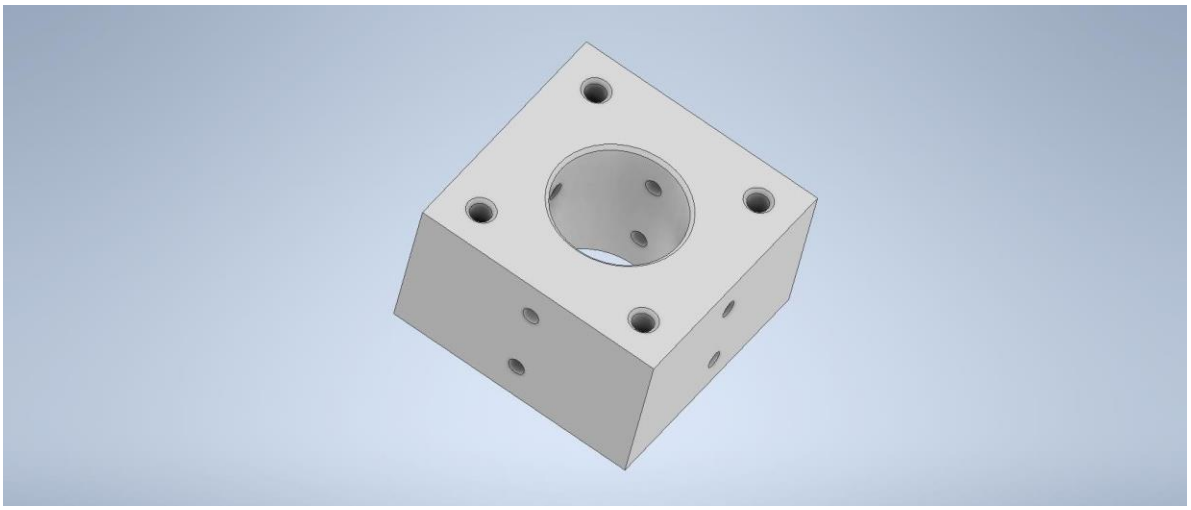


Рис. 13. Деталь для крепления ствола к шпилькам.

Бункер состоит из трех частей: основная часть (из PETG), горловина и скоба для фиксации горловины (из ABS)

Сопло было изготовлено из латунной водопроводной заглушки на 1 дюйм.

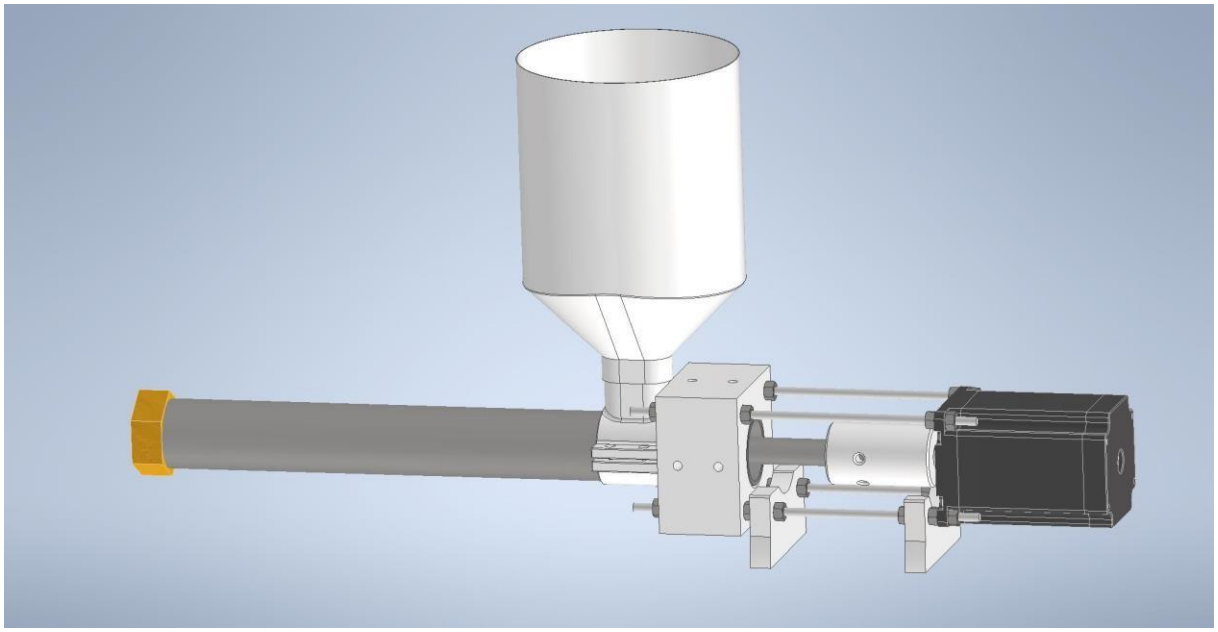


Рис. 14. Механика экструдера первой версии.

Для нагрева использовался 1 кольцевой нагреватель на 90W, управляемый через твердотельное реле. Обратная связь по температуре осуществлялась при помощи термистора HT-NTC 100K B3950. Охлаждение зоны загрузки (для предотвращения расплавления материала в бункере и выхода из строя крепления ствола) производилось при помощи змеевика из медной трубки. К змеевику для повышения уровня безопасности был припаян заземляющий проводник.

В первом прототипе были выявлены существенные недостатки:

1) Мощности нагревателя не хватало для поддержания температуры
2) Перегрев драйвера основного мотора
3) Нехватка момента для вращения шнека

ЭКСТРУДЕР. ВТОРОЙ ПРОТОТИП

Для увеличения мощности нагрева были добавлены еще 2 нагревателя.

Драйвер основного мотора был заменен на DM860A (с активным охлаждением и большей мощностью)

Рис. 15. Сравнение драйвера основного мотора первого прототипа и второго.

Для увеличения крутящего момента в конструкцию был добавлен понижающий планетарный редуктор с передаточным соотношением 4:1.

Рис. 16. Редуктор для nema23

В экструдерах есть значительные силы, действующие на шнек в направлении от сопла: сопротивление (разрезанию) гранулы, попавшей в питающем отверстии (отверстии в стволе под бункером) между стволом и шнеком, и сопротивление расплава выдавливанию (расплавленный пластик имеет большую вязкость). На редукторе установлен радиальный подшипник, плохо несущий упорные нагрузки. Также, упорные нагрузки несут риск поломки механизма редуктора: они могут сместить выходной вал. Для снятия с вала редуктора упорных нагрузок в конструкцию был добавлен упорный подшипник (внутренний диаметр 25, наружный диаметр 47, ширина 15мм).

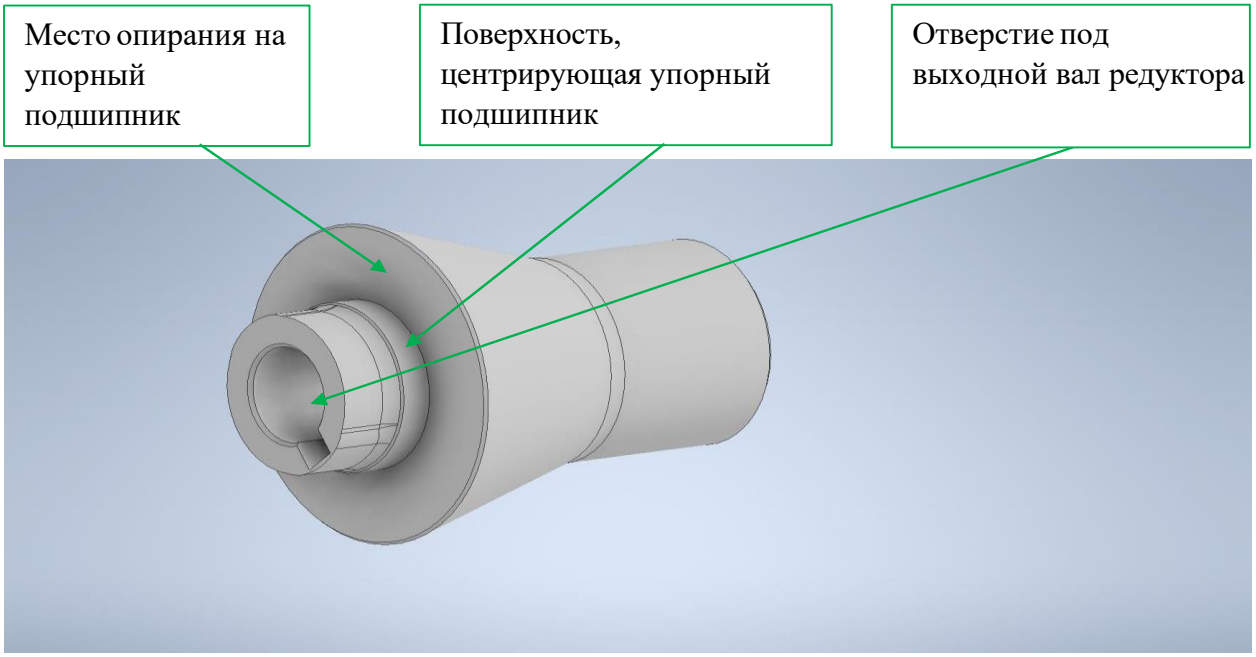


Рис.17. Муфта для соединения шнека с редуктором

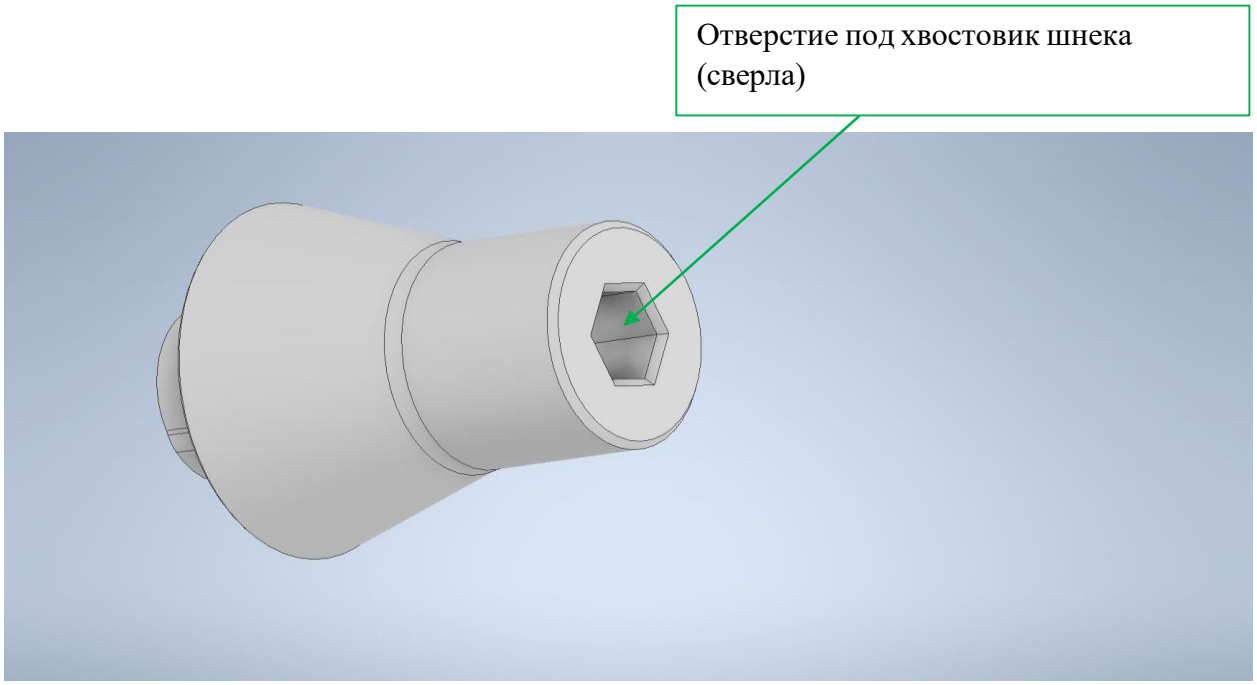


Рис. 18. Муфта для соединения шнека с редуктором

Специальная муфта передает крутящий момент с редуктора на шнек и перераспределяет упорные нагрузки на подшипник, а также центрирует подвижную половину корпуса последнего.

Неподвижная половина корпуса опирается на металлический фланец редуктора и центрируется специальной деталью.

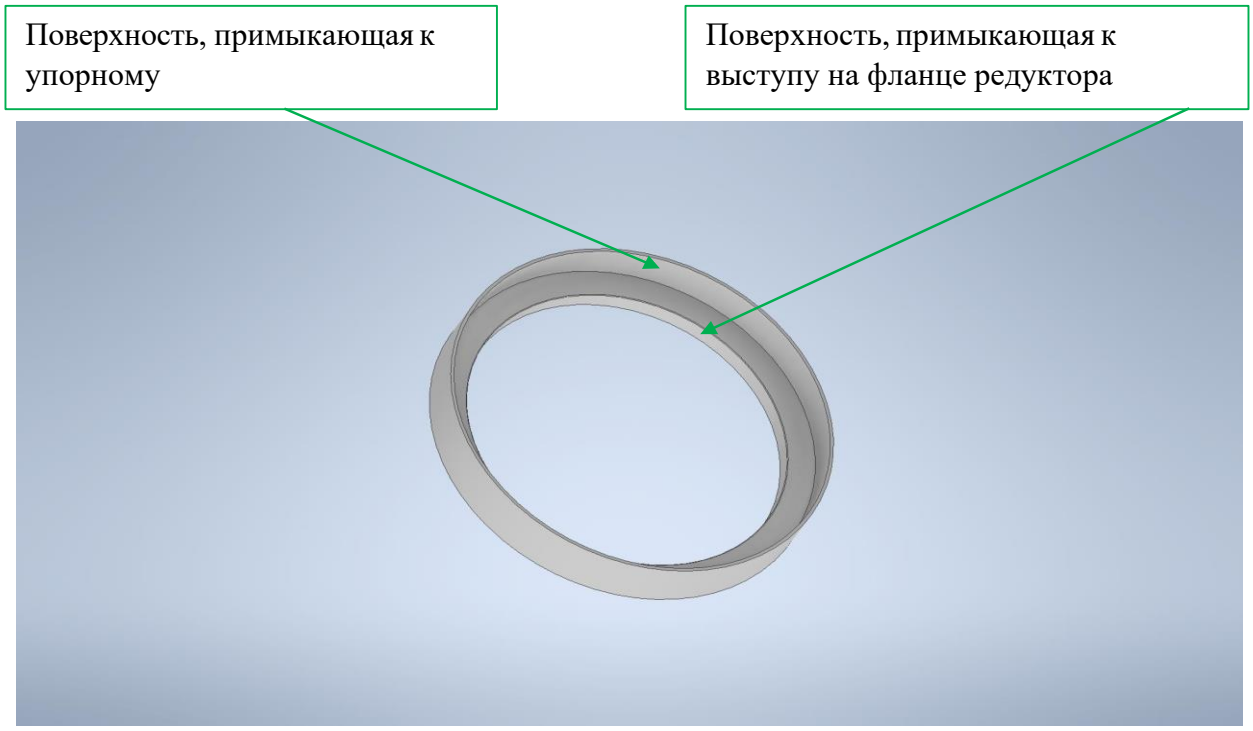


Рис. 19. Деталь, центрирующая неподвижную часть упорного подшипника

В второй версии без изменений остались:

- Система охлаждения зоны загрузки (зоны под бункером)
- Ствол и шнек
- Сопло
- Деталь крепления ствола к шпилькам
- Бункер
- Датчик температуры

Рис. 20. Второй прототип экструдера

ЭКСТРУДЕР. ТЕПЛОИЗОЛЯЦИЯ И НАГРЕВАТЕЛИ

Было разработано две версии нагрева экструдера:

- На основе покупных кольцевых нагревателей
- На основе нихромовой проволоки

Кольцевые нагреватели питаются напрямую от 220В, что позволяет разгрузить основной блок питания, но увеличивает электроопасность системы.

Нагреватель на основе нихромовой проволоки требует намотки, но может питаться от более низкого напряжения.

Рис. 21. Намотка нагревателя из нихромовой проволоки

Ствол экструдера сначала обматывается диэлектрическим материалом (в

данном случае каптоновым скотчем и стеклотканью), после чего на него наматывается греющая проволока и фиксируется каптоновым скотчем. Отводы для подачи напряжения я сделал из термостойкого провода в стеклотканевой изоляции, также возможно изготовление отводов из нихромовой проволоки, сложенной в несколько раз (для увеличения суммарного сечения отвода и уменьшения его тепловыделения)

Для теплоизоляции экструдера я использовал каолиновую вату (она обладает высокой термостойкостью, низкой теплопроводностью и нетоксична). Такая теплоизоляция позволяет более чем в 3 раза сократить мощность нагрева, необходимую для поддержания температуры свыше 210°

С. Теплоизолятор оклеил фольгированным скотчем (для фиксации и предотвращения отделения частиц (пыли) от теплоизолятора).

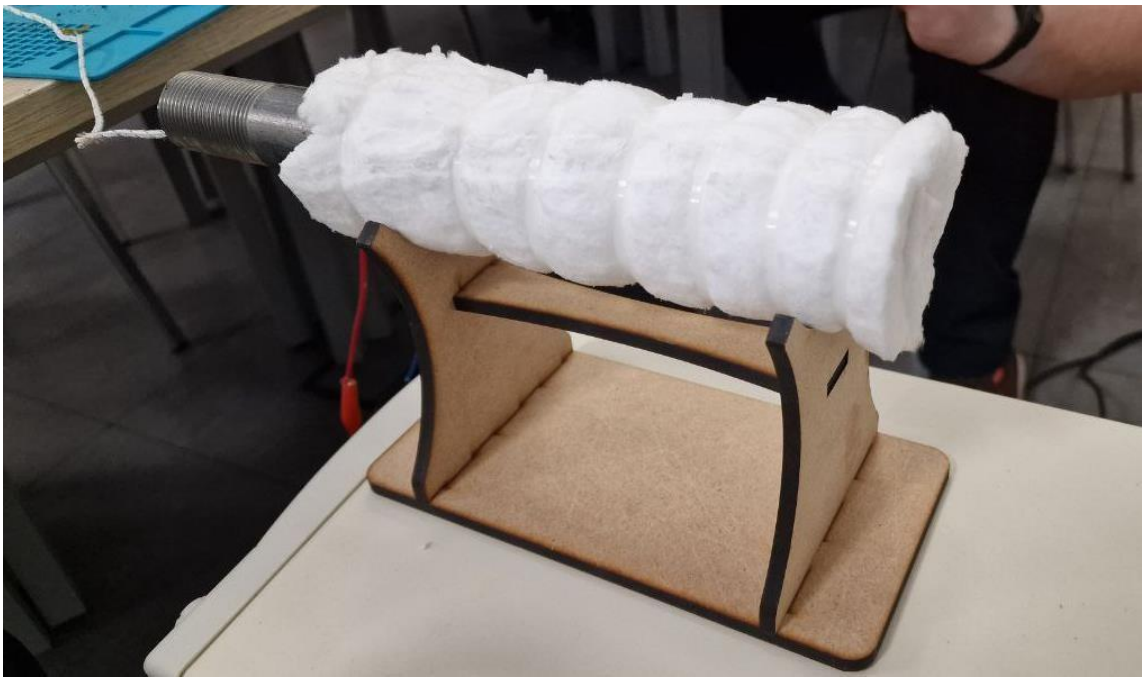


Рис. 22. Теплоизоляция.

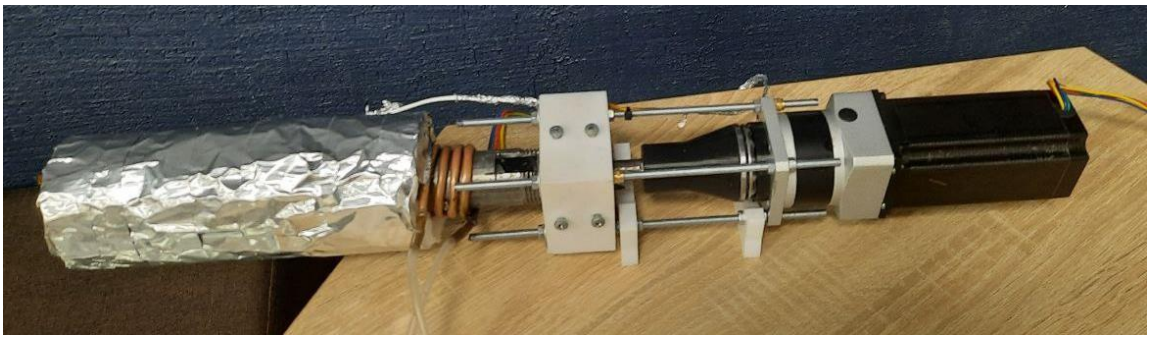


Рис. 23. Экструдер с теплоизоляцией.

Рис. 24. Экструдер с теплоизоляцией. Вывод провода термистора.

ОХЛАЖДЕНИЕ

Был выбран наиболее универсальный и компактный (что крайне важно для домашнего использования) вариант – водяное охлаждение (ванна).

Ванна имеет центральную часть, где происходит охлаждение прутка водой. Для входа и выхода филамента предусмотрены отверстия (отверстие для входа прутка больше, чем отверстие для выхода), через эти отверстия вытекает вода. Таким образом, пруток заходит в воду (в струю), проходит под водой и выходит из нее, не изгибаясь. Вода, вытекшая из центральной части, сперва попадает в боковые части ванны, перетекает в резервуар, расположенный под ванной, а потом откачивается помпой обратно в центральную часть.

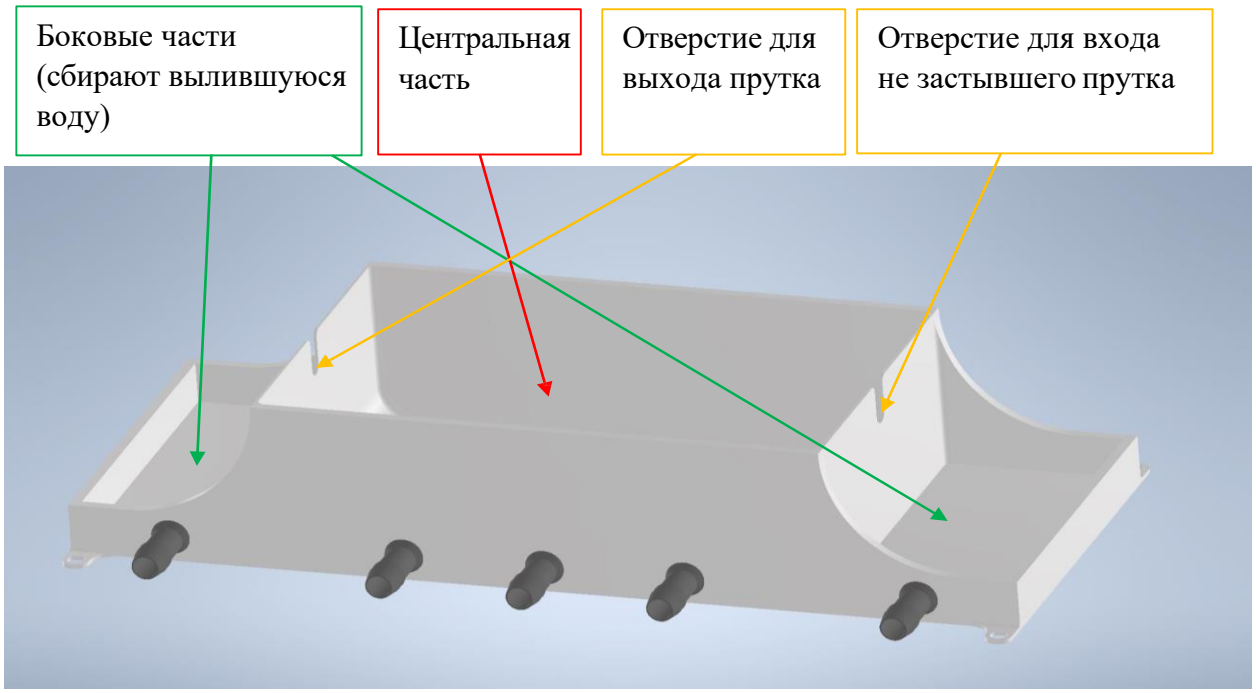


Рис. 25. Первая версия ванны

В основную деталь ванны вклеиваются штуцеры (такое разделение сильно упрощает печать ванны). На штуцеры надеваются трубки. Трубки от крайних штуцеров идут в резервуар, расположенный под ванной, а остальные штуцеры через разветвитель подключаются к помпе (разделение потока воды от помпы на 3 трубки позволяет сделать потоки внутри ванны более ламинарными).

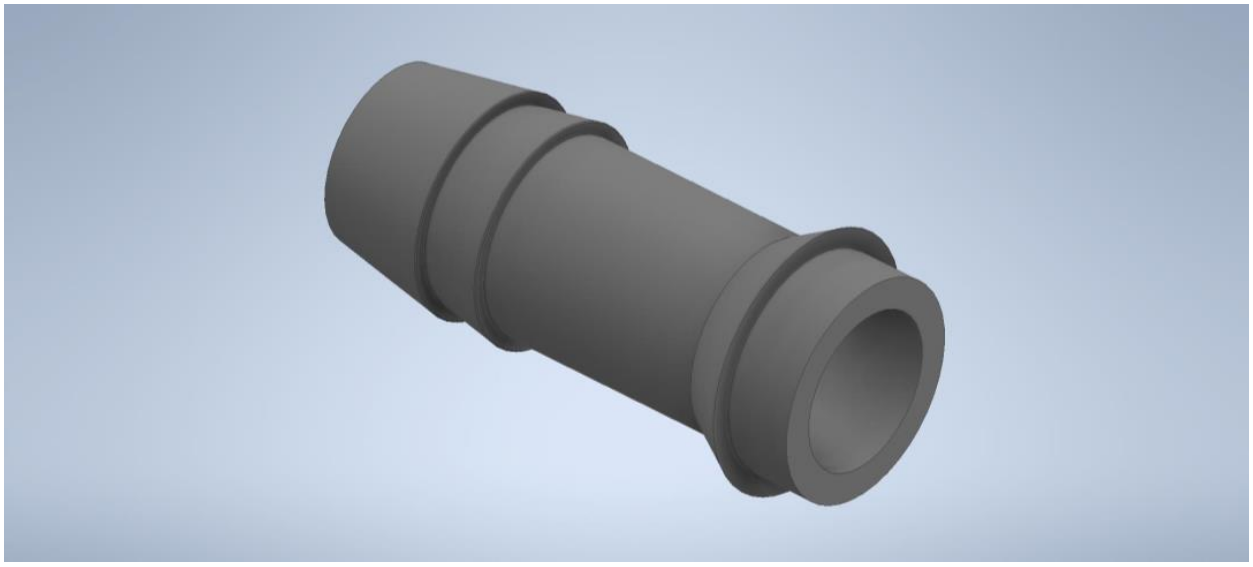


Рис. 26. Штуцер

В ходе испытаний была выявлена нехватка производительности сливных трубок.

ОХЛАЖДЕНИЕ. ВТОРОЙ ПРОТОТИП

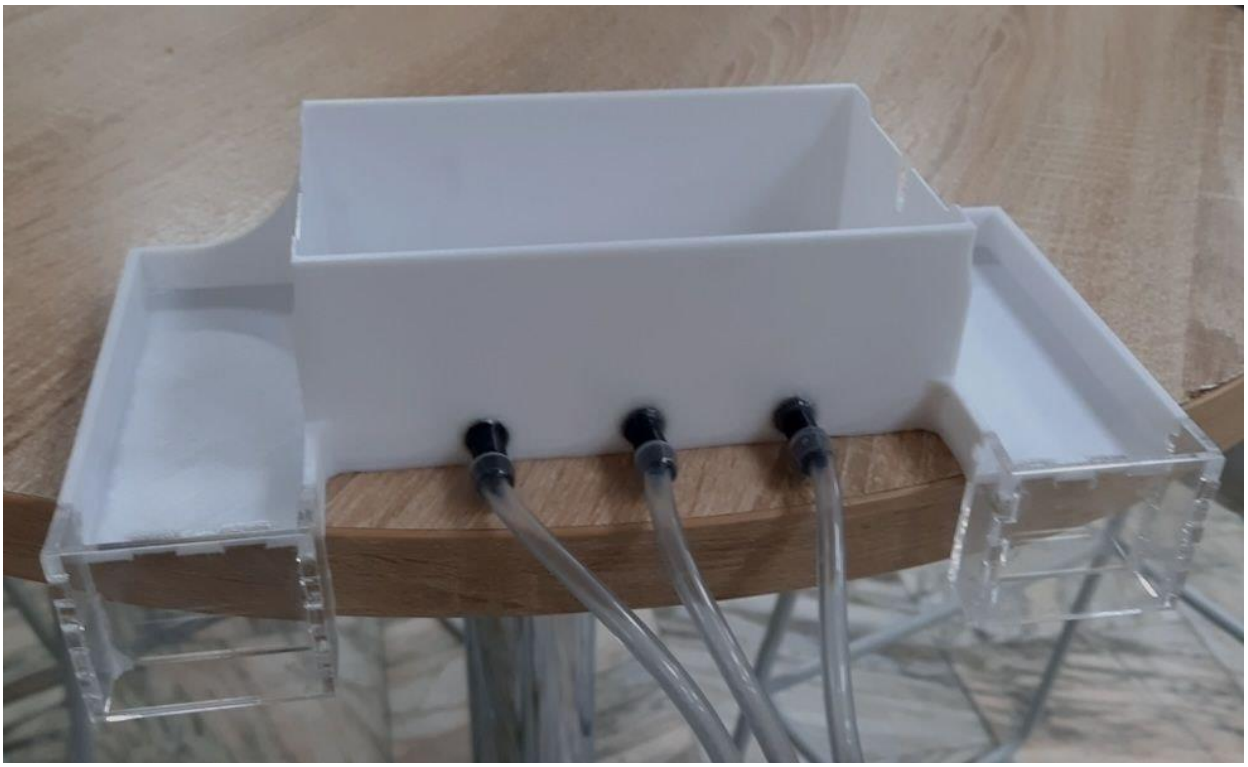


Рис. 27. Вторая версия ванны.

Трубки были заменены на жесткие прямые сливы, сбрасывающие воду в резервуар, стоящий прямо под ними. Детали сливов имеют соединение шип- паз и крепятся при помощи клея.

Прозрачные детали были вырезаны на лазерном гравере из листа акрила толщиной 3мм. (Также они могут быть напечатаны на 3D принтере.)

В этой версии была увеличена высота наружных стенок центральной части ванны.

Таким образом, удалось решить проблему с выливанием воды из системы охлаждения.

ПРОТЯЖКА

В качестве двигателя протяжки используется шаговый nema17 мотор (из-за простоты управления).

Филамент зажимается между двумя роликами, имеющими противоскользящие покрытие из полиуретана.

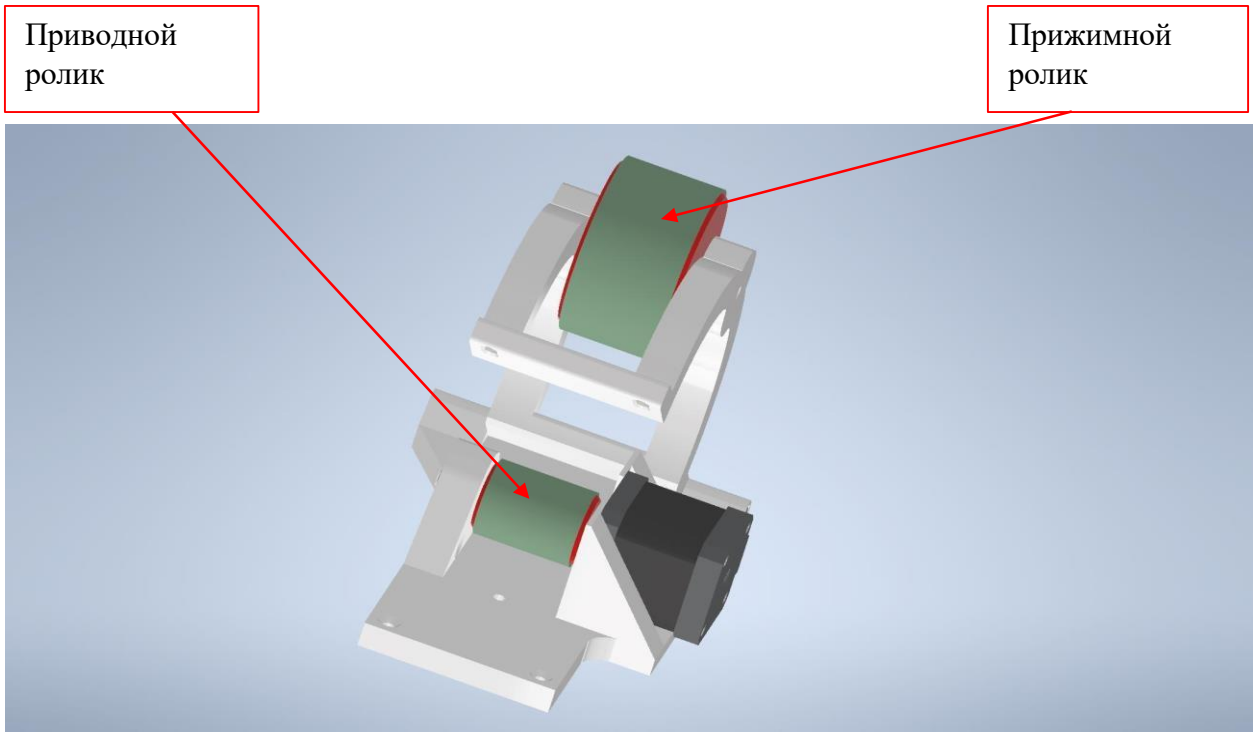


Рис. 28. Протяжка. Вид сбоку.

Приводной ролик с одной стороны плотно насаживается на вал двигателя, а с другой стороны через впрессованный в него радиальный подшипник опирается на винт м4, прикрепленный к основанию протяжки.

Прижимной ролик через впрессованные в него радиальные подшипники опирается на шпильку м4, продетую через специальные отверстия в подвижной рамке и зафиксированную гайками.

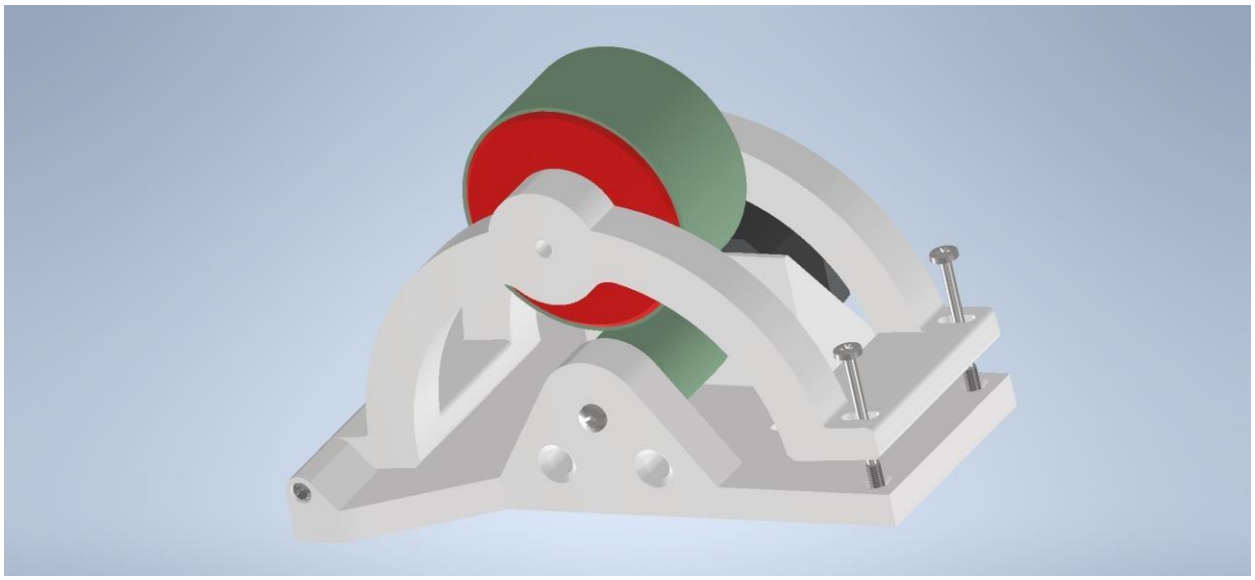


Рис. 29. Протяжка. Вид спереди.

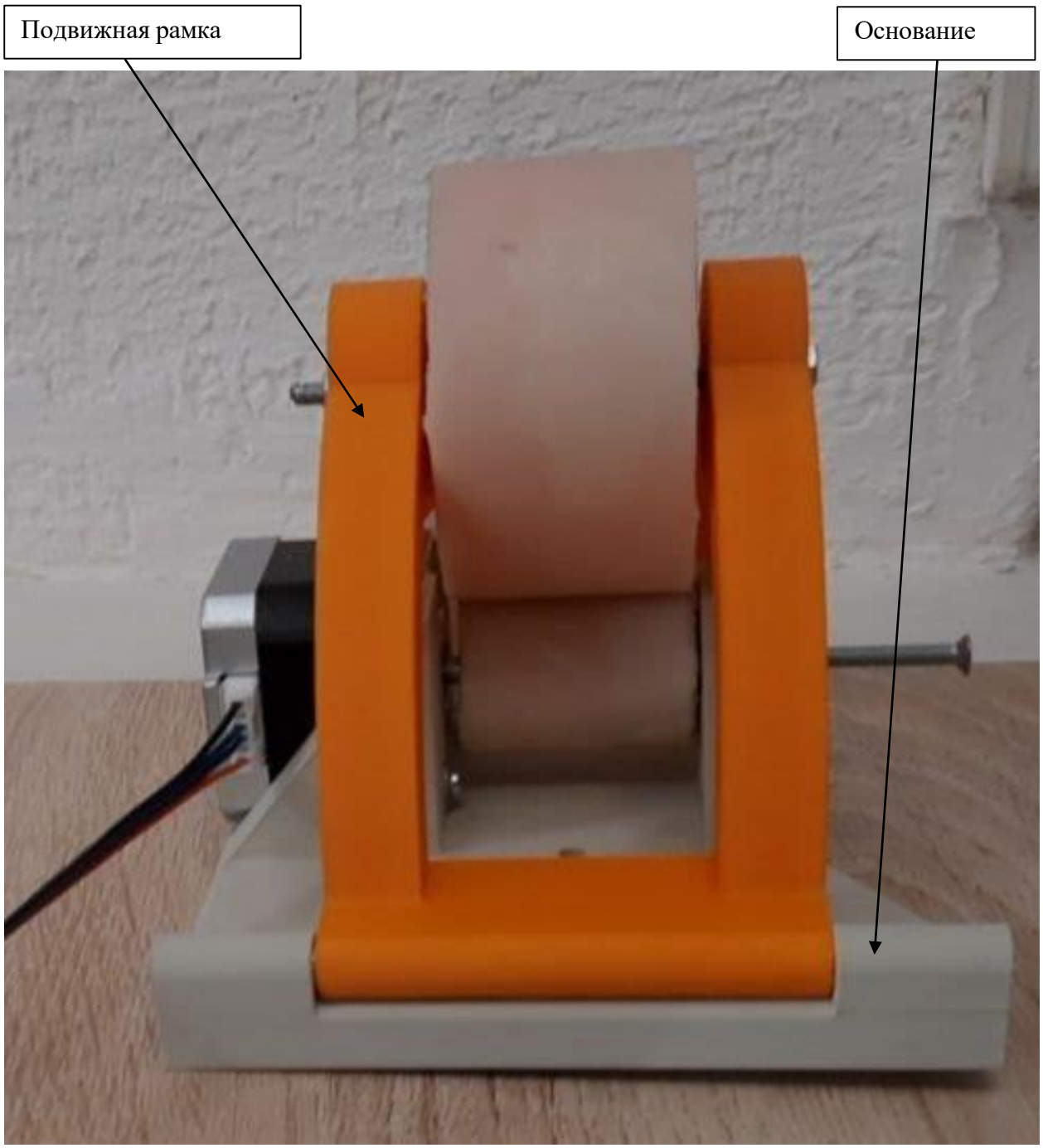


Рис. 30. Протяжка. Фото сбоку.

Подвижная рамка крепится к основанию через петлю, в качестве пальца используется вал диаметром 6 мм.

В правую часть основания вплавляются втулки М4, в которые (через пружины и отверстия в подвижной рамке) вкручиваются винты М4х50. При помощи этих винтов можно регулировать усилие прижатия роликов к прутку

ИЗГОТОВЛЕНИЕ РОЛИКОВ ПРОТЯЖКИ

Основная часть ролика печатается на FDM 3D принтере, после чего шкурится и обезжиривается.

На SLA 3D принтере (дающем более высокое качество поверхности) печатается форма для заливки двухкомпонентного полиуретана.

Рис. 31. Печать формы на LCD 3D принтере

Форма обрабатывается разделительным составом, после чего в нее запрессовывается ролик (ролик насаживается углублением/отверстием под подшипник на специальный выступ на форме).

Для защиты центрального отверстия от затекания полиуретана в него вставляется трубка.

Полиуретан замешивается и заливается в форму.

В вакуумной камере чередуются вакуум (для выхода воздуха из заливки) и нормальное давление (для заполнения пустот, откуда вышел воздух, полиуретаном).

Для заливки был выбран полиуретан твердостью по Шору 95А.

Рис. 32. Приводной ролик. Подготовка к заливке

После полимеризации форма разламывается (для этого в ней предусмотрен пропил (в большем из ребер жесткости) и ролик извлекается. Лишний полиуретан срезается.

Рис. 33. Покрытие из полиуретана

ДАТЧИК НАТЯЖЕНИЯ

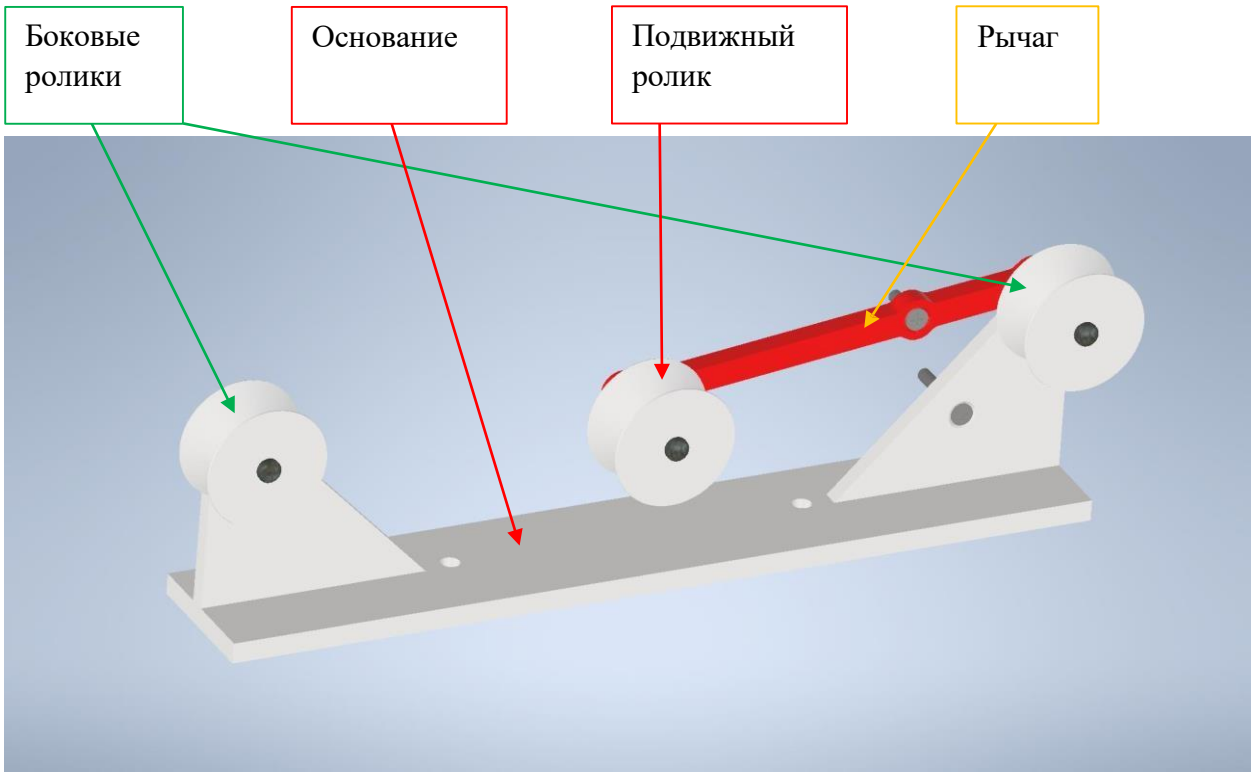


Рис. 34. Измеритель натяжения

Пластик проходит над боковыми роликами и под подвижным роликом.

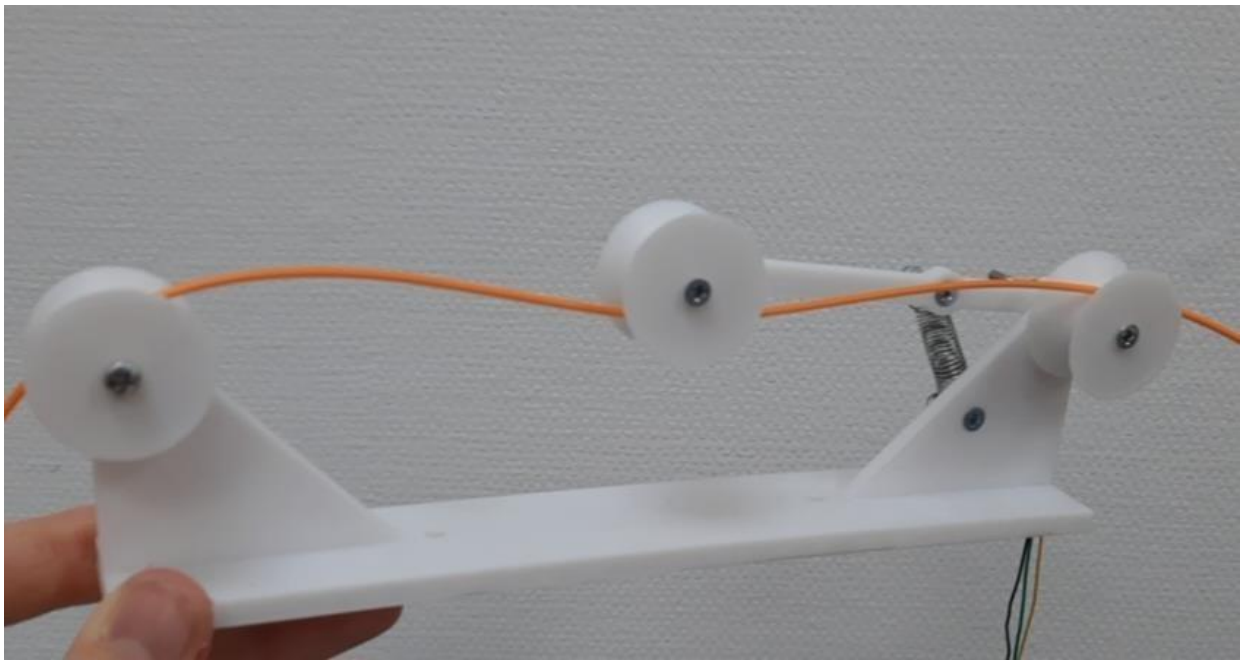


Рис. 35. Измеритель натяжения. Фото спереди

Пружина стремится опустить рычаг, а пластик, натягиваясь, стремится его поднять. Чем выше поднимается рычаг, тем больше натяжение пластика.

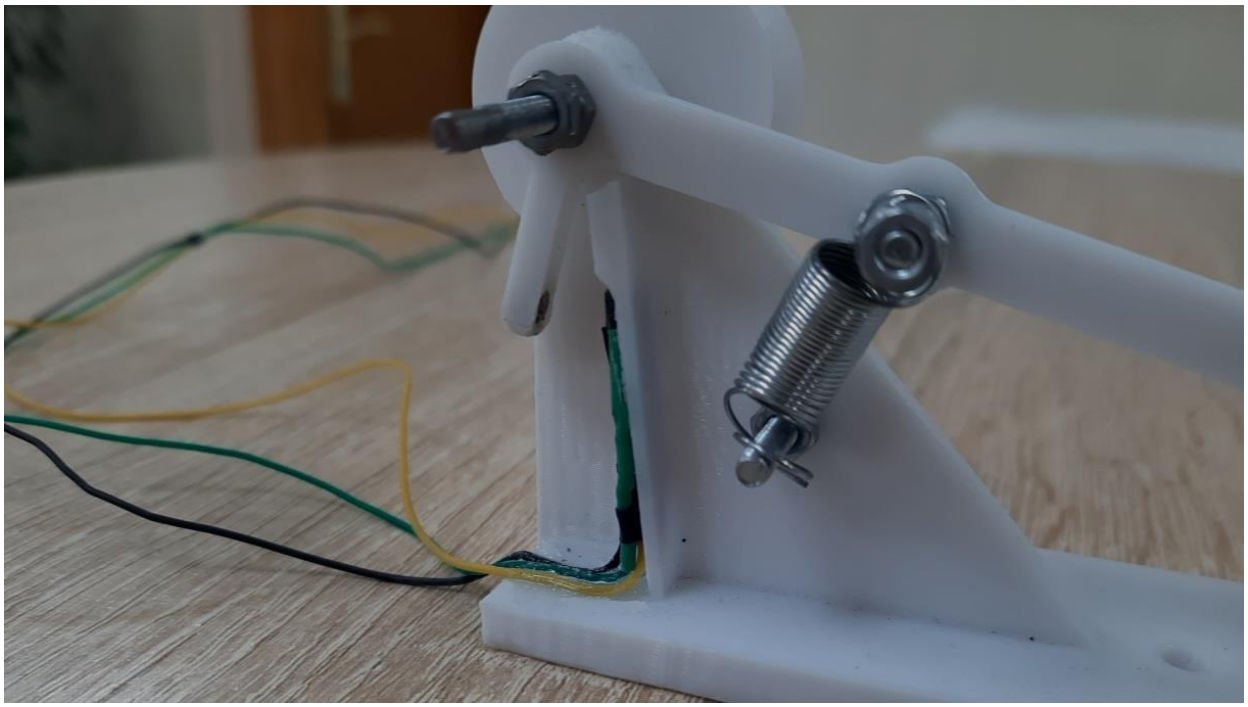


Рис. 36. Измеритель натяжения. Фото сзади

Рычаг имеет малое плечо, в котором закреплен магнит. На основании закреплен аналоговый датчик Холла (ss49e).

Когда увеличивается натяжение, основное плече рычага поднимается и плече с магнитом приближается к датчику Холла. Когда натяжение ослабевает, магнит отдаляется от датчика Холла.

Микроконтроллер, зная натяжение нити после протяжки, может управлять намотчиком филамента или устройством для нарезки пластиковой нити (последнее применяется для гранулирования пластика с целью его дальнейшей переработки литьем под давлением или другими технологиями).

ЭЛЕКТРОННАЯ СХЕМА И ПО ЭКСТРУЗИОННОЙ ЛИНИИ

В качестве основного микроконтроллера был выбран esp32 из-за большего количества портов GPIO и низкой стоимости.

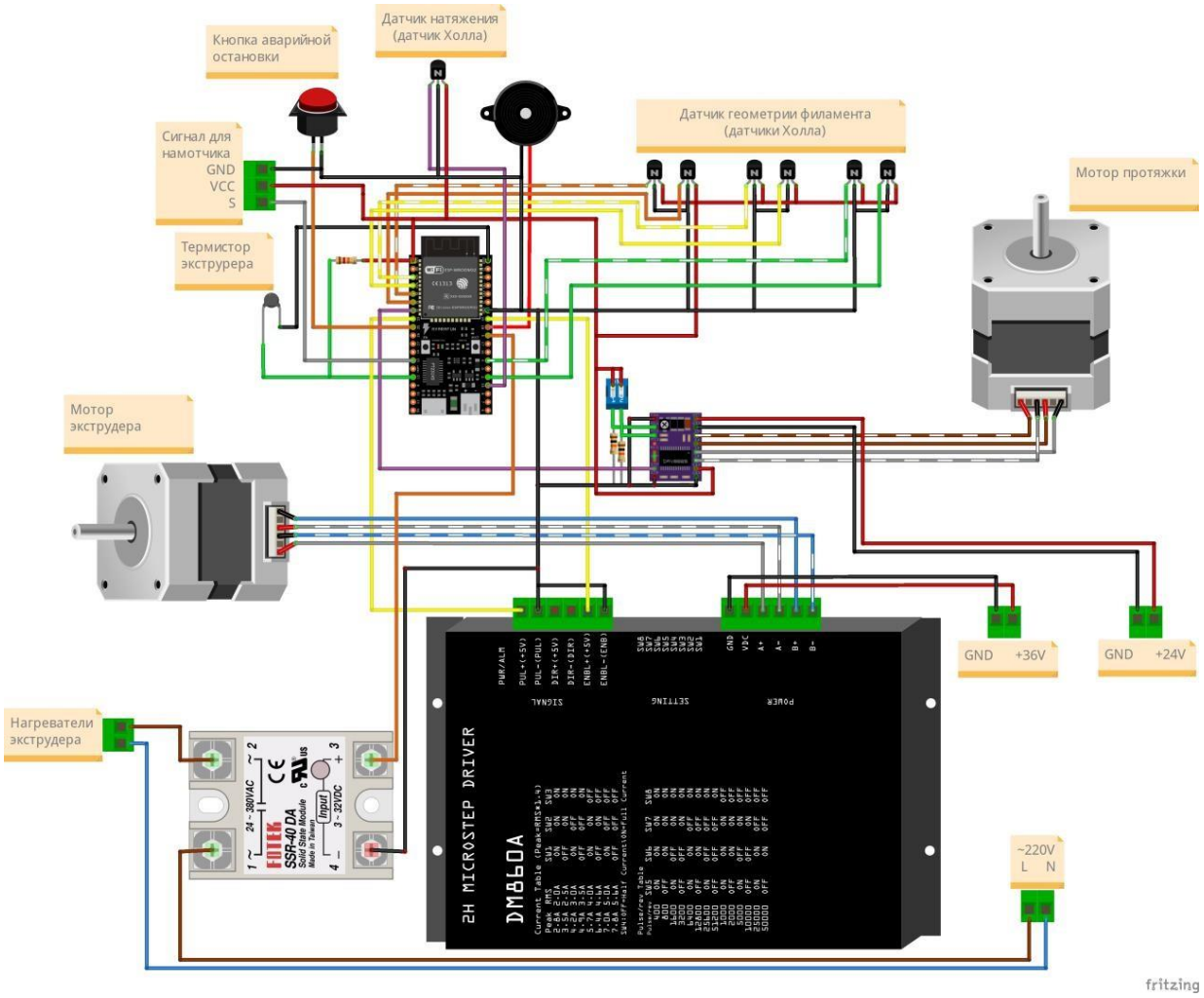


Рис. 37. Электрическая схема (для питания нагревателей от 220В переменного тока)

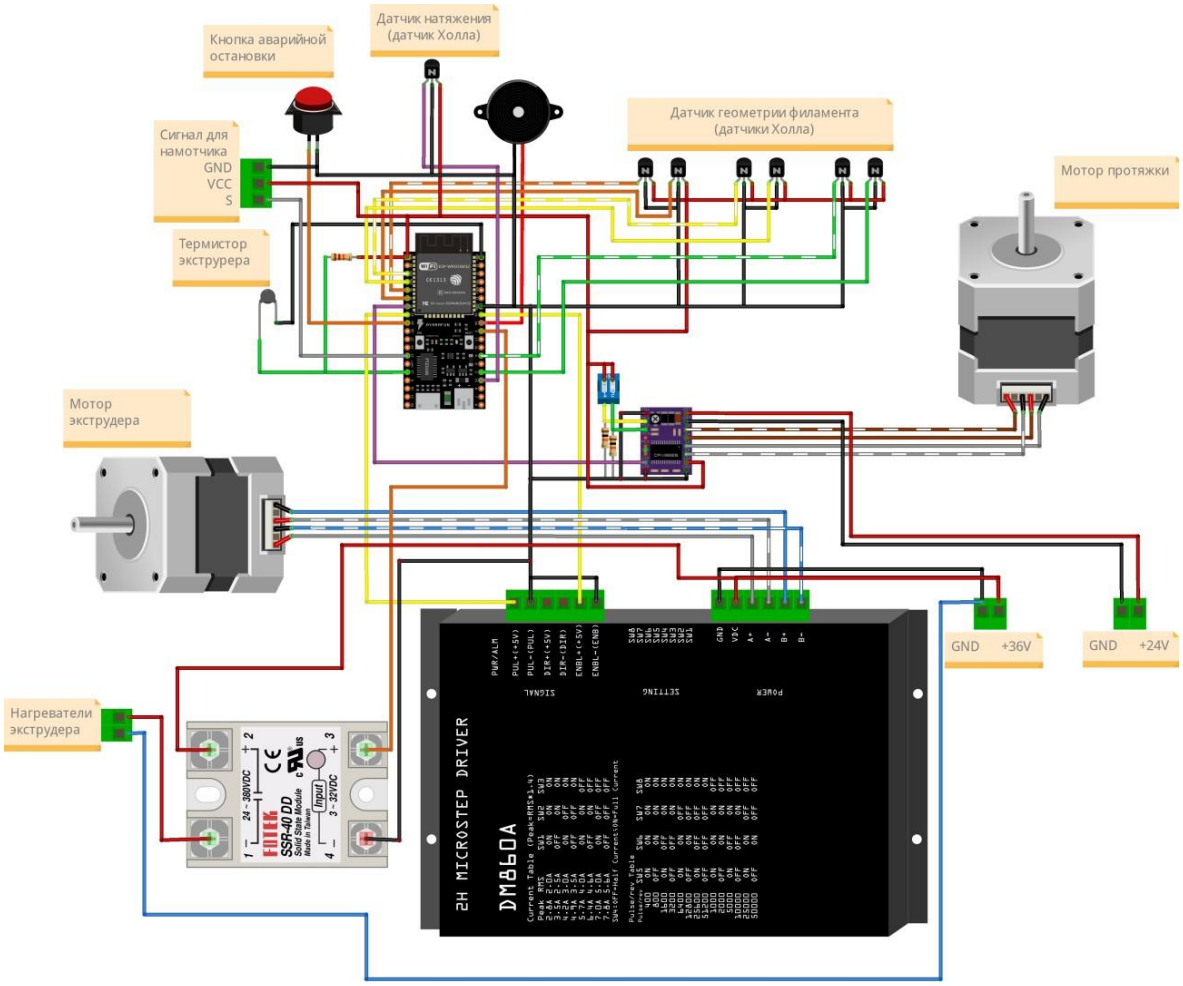


Рис. 38. Электрическая схема (для питания нагревателей от 36В постоянного тока)

Микроконтроллер питается от компьютера через USB и получает команды по этой же шине. Такое решение позволяет перенести на ПК тяжелые вычисления геометрии для обратной связи и предоставить оператору (пользователю) удобный интерфейс взаимодействия.

Программа для ПК написана на языке Python с использованием библиотек pyQt6 и pySerial и является кроссплатформенной.

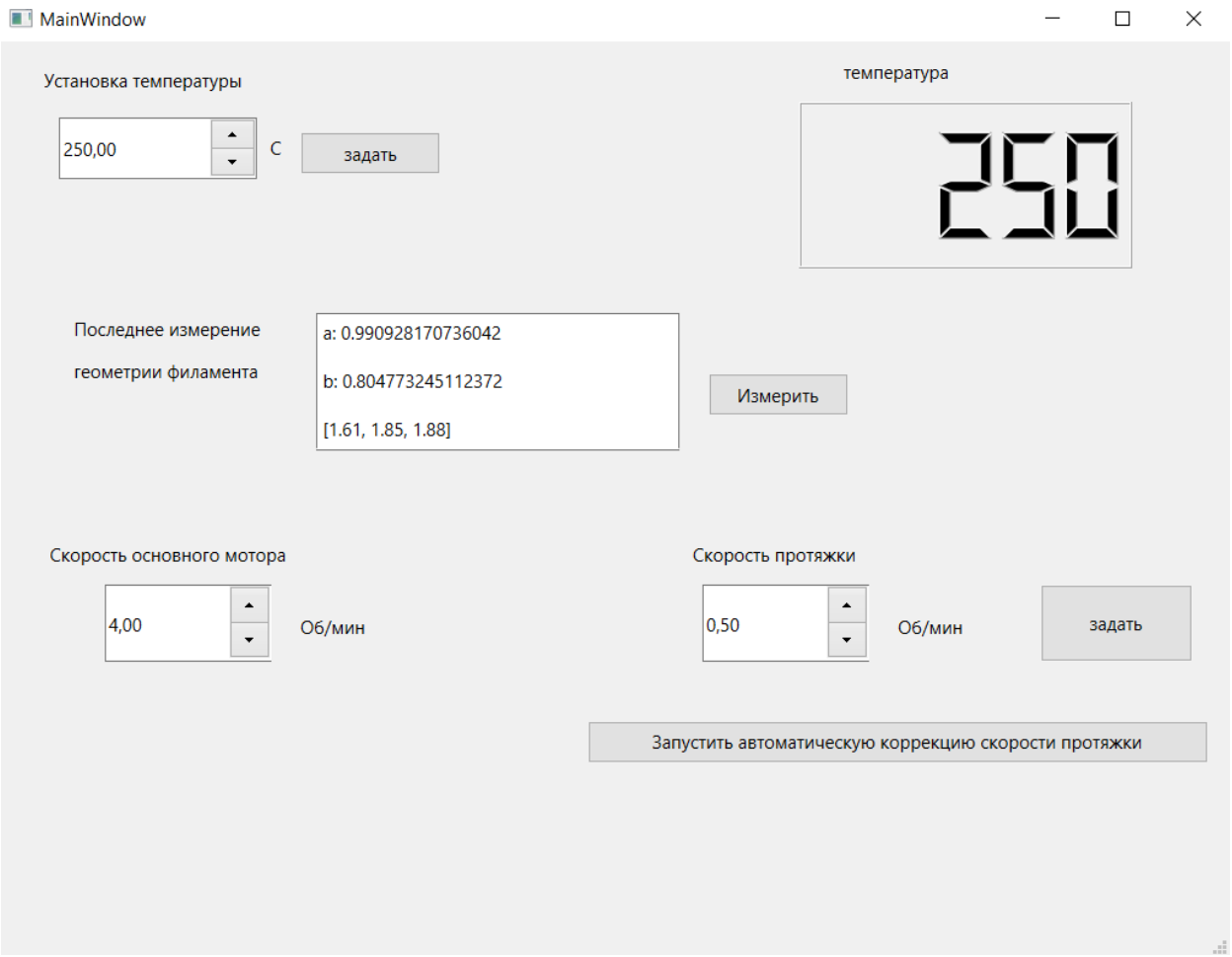


Рис. 39. Скриншот пользовательского интерфейса

ЗАКЛЮЧЕНИЕ

Датчик для принтера

Основной задачей датчика для принтера является борьба с неравномерным экструдированием (вызванным дефектами филамента). Тест на дефекты экструдирования представляет собой тонкостенную (в одну линию) башню, дефекты геометрии филамента (если их не компенсировать) проявляются в виде неповторяющихся горизонтальных полос и наплывов.

Тесты были распечатаны при помощи одного G-code файла из пластика собственного производства. При съемке фотографий (рисунки 40 и 41) положения камеры и источника света не менялись.

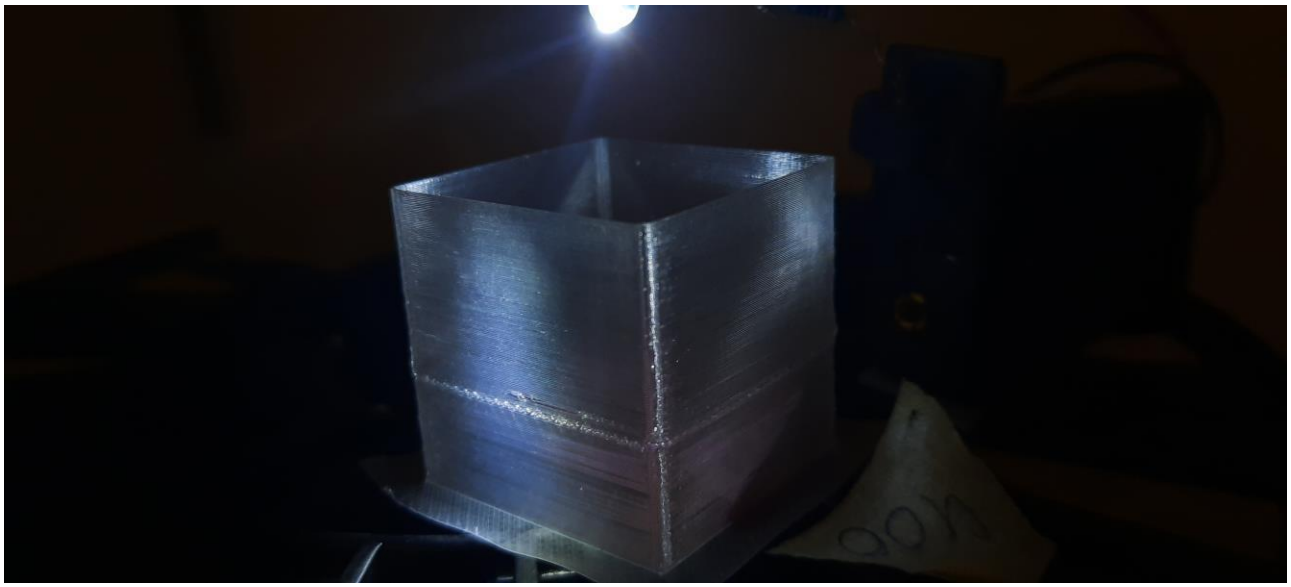


Рис. 40. Тест на неравномерность экструдирования без датчика геометрии филамента.

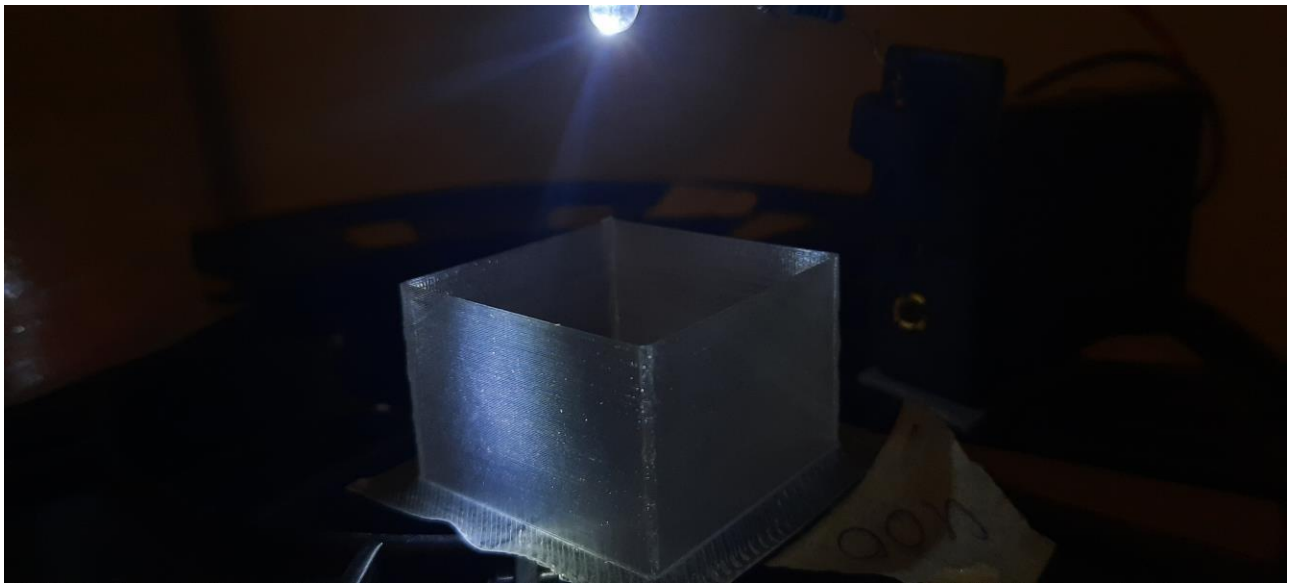


Рис. 41. Тест на неравномерность экструдирования с датчиком геометрии филамента.

На рисунках 40 и 41 видно, что разработанный мной датчик геометрии филамента убирает дефекты, связанные с неточностью изготовления прутка.

Экструзионная линия

Ниже приведена таблица со стоимостью (на 5 февраля 2026 года) основных компонентов разработанной в ходе проекта экструзионной линии.

Таблица 1. Стоимость основных компонентов

Узел	Компонент	Стоимость, ₽
Экструдер	Радиатор	561
	основной двигатель	4 280
	редуктор основного мотора	2 222
	драйвер основного двигателя	1 826
	помпа охлаждения	379
	охлаждение зоны загрузки (медная трубка)	300
	трубки силиконовые	440
	Заготовка под цилиндр	205
	Сопло	93
	Шнек (сверло)	573
	твердотельное реле	889
	термистор	119
	Понижаюший преобразователь	102
	Блок питания	1 350
	Итого:	**13 339**
Охлаждение	помпа	379
	Понижаюший преобразователь	102
	Выключатель	136
	Трубки	148
	Итого:	**765**
Протяжка	Мотор	441
	Драйвер	727
	Итого:	**1 394**

Датчик геометрии	Датчики Холла	374
	Пружины	132
	Итого:	**506**
Материнская плата	Микроконтроллер ESP32	363
	Клеммы	184
	Итого:	**547**
Итого:		**16 551**

Стоимость основных компонентов составила 16 551₽, что более чем в 3 раза дешевле аналогов.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Канал об экструзионных линиях «Underground 3D» URL:https://t.me/underground_3d
2. YouTube канал «Михаил»

URL:https://youtube.com/channel/UC4e7aQhjuWZ- 4DddZReNAkQ?si=nlLig2_l9P_yn3yw

3. Видео с канала «diy chen» про экструдер

URL:https://www.youtube.com/watch?v=MRQKmRvfZMI

4. Статья «Эллипс»

URL:https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%BB%D0%B8%D0%BF%D1%81

5. Исходный код прошивки «Klipper»

URL:https://github.com/Klipper3d/klipper

6. «Klipper 3D» документация

URL:https://www.klipper3d.org

СКРИПТ ДЛЯ УПРАВЛЕНИЯ ПОЗИЦИЕЙ НА КРИПТОВАЛЮТНОЙ БИРЖЕ

Цветкова Варвара Сергеевна

Москва г, ГБОУ Инженерная Школа №1581, 11 класс

Секция: Системы обработки информации

Научный руководитель: Цветков Сергей Юрьевич, MS, MCSE

АННОТАЦИЯ

Работа посвящена тому, как снизить риски потерь при инвестировании в криптовалюты через специальные платформы децентрализованных финансов (DeFi), например, Uniswap (версия V3).

В таких сервисах пользователи могут предоставлять свои криптовалюты в общий «пул» и получать за это комиссии от обменных операций. Однако из-за резких колебаний цен инвестор может столкнуться с так называемыми «непостоянными потерями» — ситуацией, когда его доход оказывается меньше, чем если бы он просто держал криптовалюту и ничего с ней не делал.

Цель работы — разработать способ уменьшить такие потери и создать веб-приложение, которое помогает автоматически управлять инвестициями в криптовалютах.

В исследовании:

- построена математическая модель, показывающая, как меняется стоимость вложений при росте или падении цены криптовалюты;
- рассчитано, насколько инвестор может потерять по сравнению с обычным хранением актива;
- предложены три способа защиты от потерь;
- разработан алгоритм и веб-приложение, которые помогают инвесторам уменьшить риски при работе с пулами ликвидности в DeFi.

Практическая реализация результатов выполнена в виде программного комплекса «Мастер ликвидности», включающего серверную часть на Python (с использованием библиотек ccxt, numpy, pandas, matplotlib), фронтенд на HTML, CSS и JS, API реализованную при помощи фреймворка FastAPI, и систему уведомлений через Telegram-бота. Тестирование алгоритмов проводилось на децентрализованной бирже Hyperliquid и централизованной бирже Bybit через API-интерфейсы данных площадок. В ходе разработки решены задачи устойчивого подключения к биржам, автоматического перезапуска при сбоях и сохранения состояния

стратегии.

Полученные результаты подтверждают возможность значительного снижения влияния непостоянных потерь на итоговую доходность позиции при использовании алгоритмического хеджирования. Работа имеет практическую значимость для частных инвесторов и разработчиков DeFi-инструментов и может служить основой для дальнейших исследований в области автоматизированного управления рисками на криптовалютном рынке.

ВВЕДЕНИЕ

Криптовалютная экономика берет свое начало в 2008 году, когда Сатоши Накамото опубликовал статью «Биткойн: P2P электронные деньги». За прошедшее время криптовалюты превратились в важнейшее средство хранения и передачи активов. Капитализация криптовалютного рынка исчисляется триллионами долларов, основные монеты торгуются на ведущих фондовых биржах и все больше стран размещают свои валютные резервы в криптовалютах.

Криптовалюта – это валюта в цифровой форме; для защиты транзакций в криптовалюте используется шифрование (криптография). Не существует центрального органа по выпуску или регулированию криптовалют. Для записи транзакций и выпуска новых единиц используется децентрализованная система, основанная на технологии Блокчейн.

Блокчейн – распределенный реестр, используемый для записи и хранения цифровых транзакций. Данные в блокчейне хранятся в блоках. Блоки соединяются друг с другом в линейном, хронологическом порядке, создавая цепочку. Данные являются хронологически последовательными, поскольку нельзя удалять или изменять цепочку без консенсуса со стороны сети.

Благодаря своей децентрализации, криптовалюты породили совершенно новый тип финансовых инструментов – децентрализованные финансы (Decentralized Finance, DeFi).

Децентрализованные финансы – это финансовые инструменты и сервисы, функционирующие на основе технологии блокчейн и использующие в своей работе смарт контракты. Благодаря своей децентрализации DeFi инструменты не нуждаются в посреднике или регуляторе, что позволяет снизить издержки и повысить скорость проведение транзакций.

Смарт-контракт («умный контракт») — это компьютерный алгоритм (код), автоматически исполняющий условия договора между сторонами, записанный в блокчейне. Он заменяет посредников (нотариусов, банки), гарантируя выполнение сделки только при наступлении заранее прописанных условий, что делает процесс прозрачным, безопасным и необратимым.

Перечислим основные способы инвестирования в DeFi.

1. Предоставление ликвидности в пулы ликвидности – внесение двух или более различных активов в пул для предоставления ликвидности в обменных операциях с целью получения вознаграждения в виде комиссий за обмен.
2. Предоставление ликвидности в долг (lending) и получение ликвидности в заем (borrowing) – предоставление ликвидности лендинговым протоколам с цель получения доходности в виде годового процента (APR) и использование внесенной ликвидности в качестве залога для получения займа под процент.

3. Стейкинг (staking) – поддержание работы блокчейна с помощью блокировки в нем активов за вознаграждение.

В данной работе мы рассмотрим принципы работы пулов ликвидности с точки зрения инвестора. Хотя пулы ликвидности и обещают большой процент доходности, внесение в них ликвидности сопряжено со значительными рисками убытков из-за колебаний курсов внесенных криптовалют. Возникающие таким образом убытки называются непостоянными потерями (Impermanent Loss, далее сокращенно ImL).

Пул ликвидности (Liquidity Pool) — это автоматический обменник криптовалют, управляемы смарт-контрактами в блокчейне и получающий ликвидность для осуществления обмена от сторонних инвесторов. Инвесторы зарабатывают за счет комиссий, которые взымаются с клиентов при обмене.

Непостоянные потери — это риск возникновения финансовых потерь при взаимодействии с пулами ликвидности на DeFi-платформах. Такая ситуация возникает при изменении стоимости активов, отправленных инвестором в пул ликвидности. В данной работе мы будем исследовать непостоянные потери в пулах ликвидности, работающих по алгоритму Uniswap V3 (алгоритм концентрированной ликвидности), и рассмотрим возможные способы устранения или уменьшения этих потерь.

Актуальность данной работы состоит в том, с момента появления в 2020 году алгоритма Uniswap V3, пулы ликвидности превратились в один из самых популярных и распространенных способов инвестирования в DeFi. Но в то же время, для проблемы непостоянных убытков так и не было найдено удовлетворительного решения. Различные предлагаемые методы позволяют лишь несколько уменьшить размеры непостоянных убытков или сделать их менее заметными за счет снижения доходности. Мы в данной работе предложим универсальный способ устранения непостоянных потерь методом хеджирования волатильного актива в пуле ликвидности.

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

В данной работе мы будем исследовать непостоянные потери в пуле ликвидности Uniswap V3 при внесении в него двух активов – одного волатильного и одного стабильного (Stable Coin). Суммарная стоимость активов в пуле при стартовой цене волатильного актива , определяется следующим образом:

$$V_0 = x \cdot p_0 + y \qquad (2.1)$$

где V_0 — это весь объем внесенной в пул ликвидности, p_0 - начальная цена волатильного актива, x - колличество волатильных монет, y - колличество стейблов.

В Uniswap V3 ликвидность предоставляется только в определенных интервалах цен. Такая ликвидность называется «концентрированной» и работает только в ценовом диапазоне $[p_a, p_b]$,

где p_a – нижняя граница ценового диапазона, а p_b – верхняя граница.

Непостоянные потери – это разность между начальной стоимость внесенных в пул активов и их стоимость при текущей цене p_x , и могут быть рассчитаны по формуле:

$$IL = V_0 - (x' + y') =$$
$$= x \cdot p_x + y - L \cdot \frac{\sqrt{p_b} - \sqrt{p_x}}{\sqrt{p_x \cdot p_b}} - L \cdot (\sqrt{p_x} - \sqrt{p_a}) \quad (2.2)$$

На рис. 1 показано изменение активов в пуле ликвидности под влиянием непостоянных потерь при изменении цены p_x . Также на графике показано изменение стоимости активов при пассивном владении, без внесения в пул ликвидности.

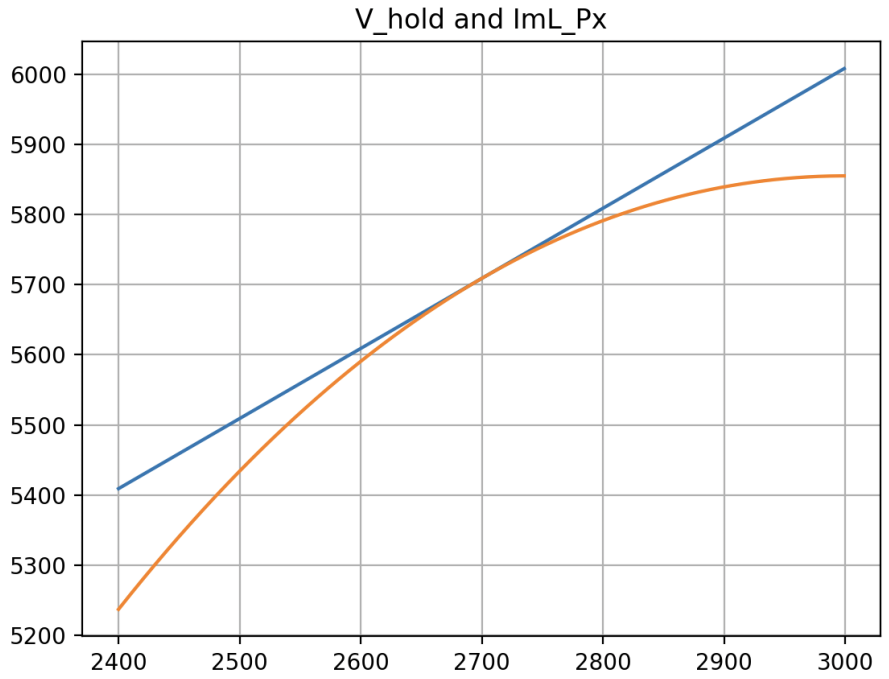


Рис. 1. Сравнение изменения стоимости активов при внесении в пул Uniswap V3

Моделирование проводилось для следующих параметров:

```
Px = 2700 #entry price
Pa = 2400 #low range
Pb = 3000 #top range
```

Как видно из графика, при росте цены, в диапазоне цен $p_0 < p_x < p_b$, стоимость активов в пуле растет, но уступает по объему активам при пассивном владении. А при падении цены, в диапазоне цен $p_a < p_x < p_0$, непостоянные потери приводят к уменьшению ликвидности в пуле и убытки в этом интервале цен опережают убытки в случае пассивного владения активами.

Кроме того, необходимо отметить, что количество монет x и y в пуле изменяется при изменении цены. При приближении цены к нижней границе диапазона p_a, активы в пуле перетекают в дешевеющий актив x, а при возрастании цены – в стабильный актив y. На рис. 2 показано изменение активов при изменении цены p_x:

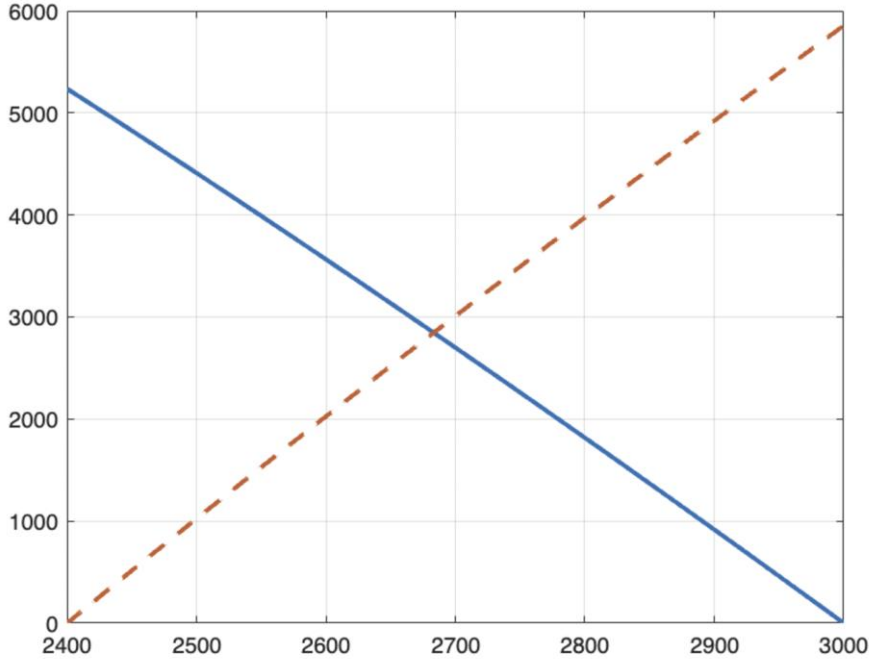


Рис. 2. Изменение активов в пуле ликвидности при изменении цены p_x

Проведенное математическое моделирование показало, что при изменении цены количество активов в пуле и их совокупная стоимость меняется. Кроме того, непостоянные потери опережают потери в стоимости активов при пассивном владении. Далее исследуем возможные способы устранения данной проблемы.

РАЗРАБОТКА АЛГОРИТВОМ ХЕДЖИРОВАНИЯ

В данной работе мы будем исследовать способы устранения непостоянных потерь при помощи хеджирования.

Хеджирование — это защита финансовой инвестиции за счет снижения рисков через сделки, которые покроют возможные убытки от основной инвестиции. Для нашей задачи хеджирование — это нахождение и реализация инвестиционной позиции, которая при сложении с кривой изменения стоимости активов в пуле ликвидности даст постоянную величину при любом изменении цены (рис. 3).

Графически задачу нахождения хеджирующей позиции можно представить как сложение основного графика изменения активов с такой искомой функцией, что в результате мы получим график константы равной первоначальному объему внесенной ликвидности V_0:

Рис. 3. Графический результат применения хеджирующей позиции

Для построения хеджирующей функции нам доступны только стандартные биржевые фьючерсы — длинная позиция (лонг, long) и короткая позиция (шорт, short). Это требование является фундаментальным ограничением при разработке хеджирующего алгоритма. Графики доходности длинной и короткой позиций являются линейными функциями и показаны на рис. 4.

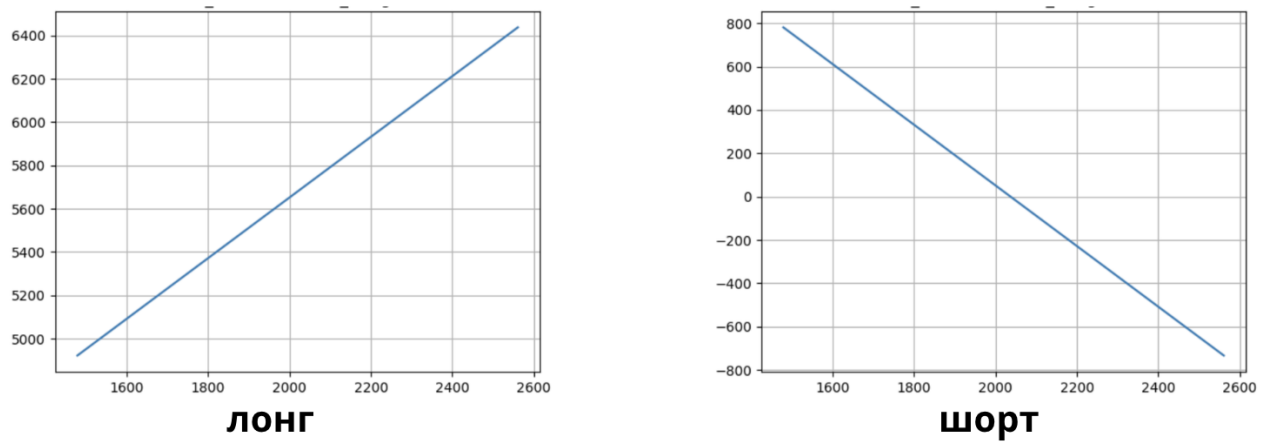


Рис. 4. Графики доходности длинной и короткой позиций

Как нетрудно заметить, лонг и шорт являются друг для друга идеальными хеджирующими позициями. Но, к сожалению, в данном исследовании нам нужно найти хеджирующую функцию для нелинейной кривой изменения ликвидности в пуле и готовой функции для решения данной задачи не существует.

В ходе работы над данным проектом нами были предложены три способа нахождения хеджирующей функции.

МЕТОД ОТКРЫТИЯ КОРОТКОЙ ФЬЮЧЕРСНОЙ ПОЗИЦИИ

Для хеджирования непостоянных потерь по данному методу мы будем открывать короткую позицию с помощью бессрочного фьючерса на сторонней торговой площадке используя дополнительный объем ликвидности.

Короткая фьючерсная позиция (short futures) – это вид биржевой сделки, при которой

инвестор зарабатывает на снижении цены актива. Бессрочный фьючерс имеет линейный график доходности на всем ценовом диапазоне, которая рассчитываете по следующей формуле:

$$PnL_{short} = d \cdot (p_x - p_0) \cdot L * x_{futures} \qquad (3.1)$$

Где d – направление фьючерса (1 – лонг, -1 – шорт), p_x – текущая цена, p_0 – начальная цена (цена открытия сделки), $x_{futures}$ – размер позиции.

Пример моделирование доходности короткого фьючерса показано на рис. 5.

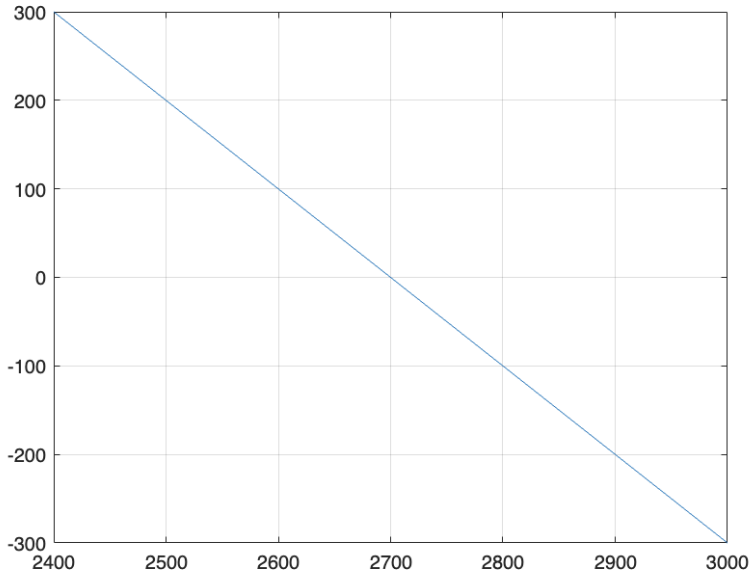


Рис. 5. Моделирование доходности короткой фьючерсной позиции

Значения непостоянных потерь при хеджировании коротким фьючерсом могут быть найдены п формуле:

$$ImL_{short} = PnL_{short} + ImL \qquad (3.2)$$

Принцип хеджирования состоит в вложении двух графиков доходности. Результат применения хеджирования шортом показан на рис.6:

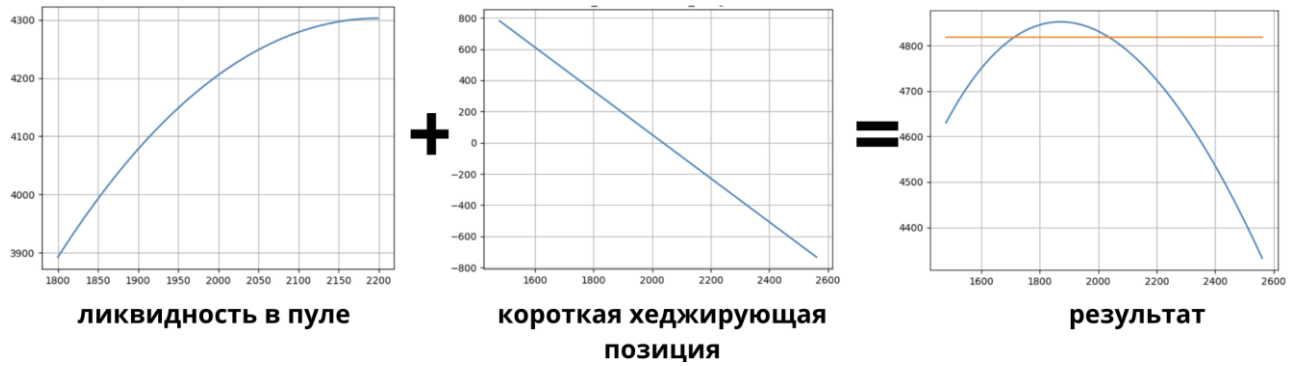


Рис. 6. Применение короткого фьючерса для хеджирования

Как видно из графика, нам не удалось полностью избавиться от непостоянных потерь при данном методе хеджирования. Хотя в диапазоне цен $p_a < p_x < p_0$ непостоянные потери уменьшились, а при увеличении плеча или размера короткого фьючерса, они вообще могут быть сведены к нулю в отрицательном диапазоне цен, в диапазоне цен $p_0 < p_x < p_b$ короткий фьючерс

только увеличивает потери. Таким образом данный метод хеджирования может изменить форму графика доходности итоговой позиции (красный график).

График изменения ликвидности в пуле при данном методе хеджирования в сравнении с пассивным владением и инвестированием в пул без хеджирования показано на рис. 7.

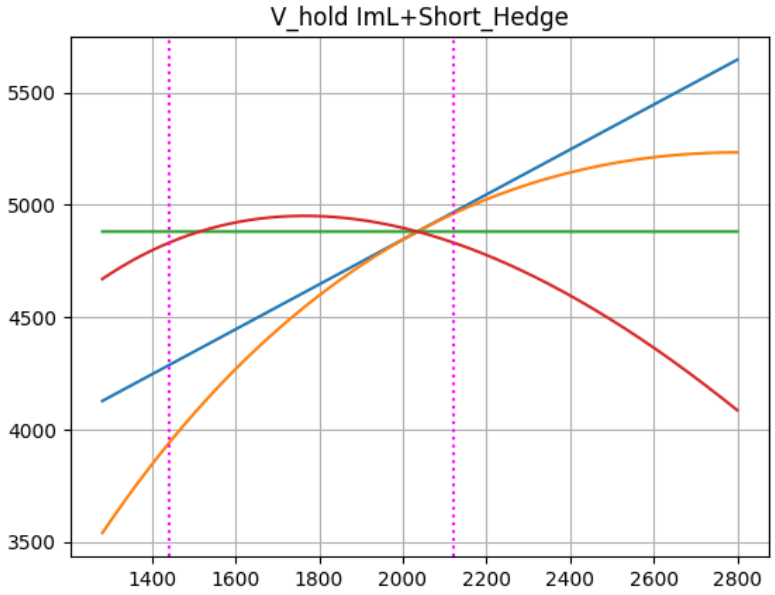


Рис. 7. Изменение непостоянных потерь при хеджировании коротким фьючерсом

МЕТОД ДИНАМИЧЕСКОГО ХЕДЖИРОВАНИЯ С ИЗМЕНЕНИЕМ РАЗМЕРА ХЕДЖИРУЮЩЕЙ ПОЗИЦИИ

Метод динамического хеджирования с изменением размера хеджирующей позиции заключается в динамическом изменении размера хеджирующего шорта для его соответствия размеру ликвидности волатильного актива в пуле.

Поскольку в данном методе мы постоянно подстраиваем размер хеджирующего депозита под объем ликвидности, график доходности данного шорта не может быть построен, т.к. он постоянно изменяется. При выборе алгоритма работы данного метода следует обратить внимание на другой параметр – выбор уровня дискретности изменения хеджирующего шорта и алгоритм определения момента для изменения его объема.

Возможны два способа определения момента для изменения размера шорта – при изменении размера основной позиции на определенный процент и метод фиксированных интервалов, при котором шорт автоматически подстраивается под размер ликвидности в пуле при достижении заранее заданных значений цены.

Недостатком данного метода является необходимость регулярного изменения размера хеджирующего шорта, что влечет за собой неизбежные потери из-за периодической фиксации отрицательного PnL шортового фьючерса. На рис.8 показан график изменения общей

стоимости хеджирующего шорта и ликвидности в пуле в методе динамического хеджирования (монеты VIRTUAL/ETH).

Как видно из графика из-за волатильности общая сумма (пул+хедж) неуклонно уменьшается с течением времени из-за волатильности и необходимости изменять размер хеджирующего шорта.

Блок-схема алгоритма хеджирования с автоматическим изменением размера хеджирующей позиции показана на рис. 9

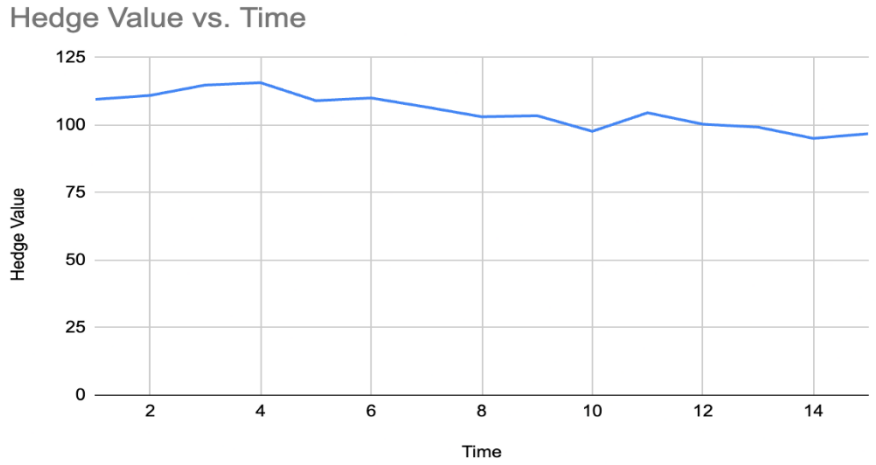


Рис. 8. Изменение общей стоимости хеджирующего шорта и ликвидности в пуле в методе динамического хеджирования

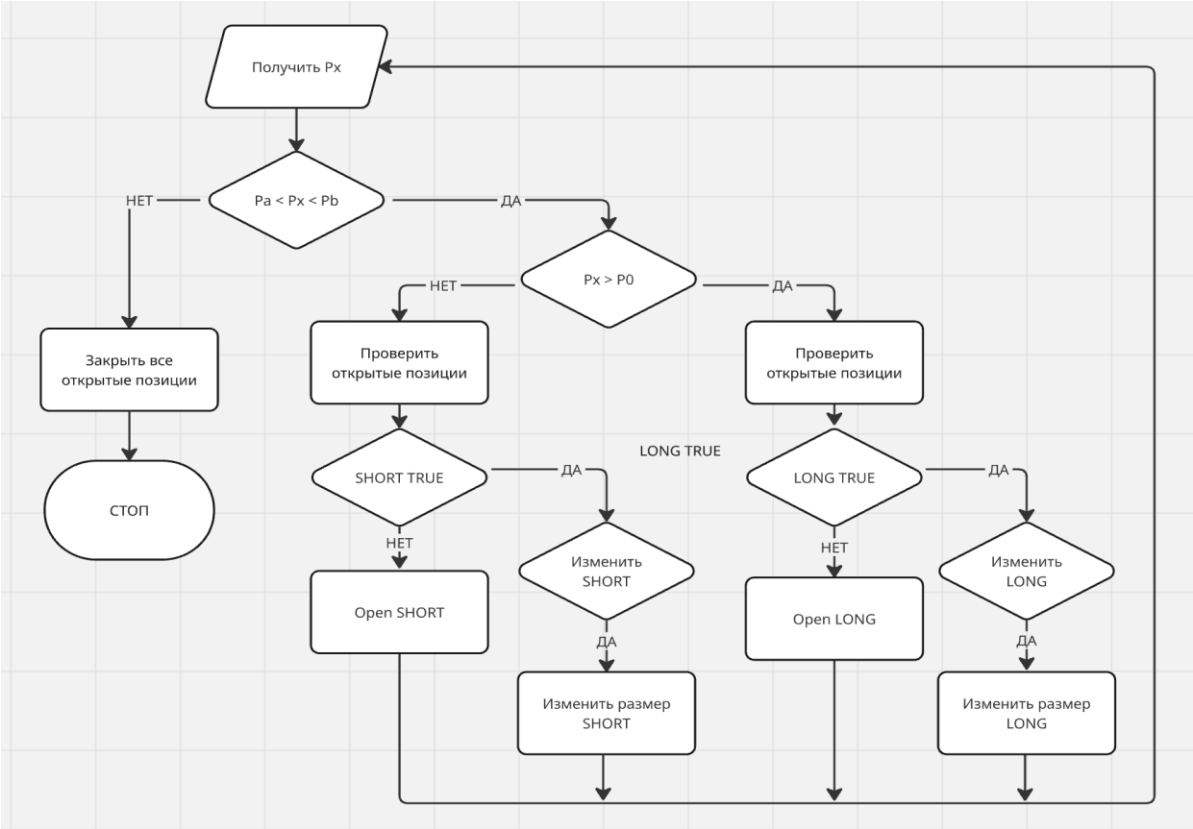


Рис. 9. Блок-схема алгоритма хеджирования с автоматическим изменением размера хеджирующей позиции

МЕТОД ДИНАМИЧЕСКОГО ХЕДЖИРОВАНИЯ СО СМЕНОЙ ПОЗИЦИИ ПРИ ПЕРЕХОДЕ ЧЕРЕЗ НАЧАЛЬНУЮ ЦЕНУ (P0)

В ходе проведения данного исследования нами был разработан метод динамического хеджирования, суть которого состоит в следующем:

в диапазоне цен $p_a < p_x < p_0$ - мы открываем короткую позицию

в диапазоне цен $p_0 < p_x < p_b$ – мы открывает длинную позицию

Иными словами, если цена от точки входа двинулась вверх, мы открываем длинную позицию, если вниз — короткую. Данный метод по своей природе подразумевает периодическое открытие и закрытие фьючерсных позиций по мере того, как цена волатильного актива пересекает значение начальной цены p_0. Использование для этих целей традиционных бирж (как централизованных CEX бирж, так и децентрализованных DEX бирж) неизбежно влечет дополнительные расходы в виде комиссий за открытие и закрытие фьючерсных сделок. Поскольку цена может достаточно часто пересекать точку входа p_0, издержки в виде комиссий могут быть значительными при использовании данного метода.

С целю уменьшения расходов на комиссии мы изучили возможность хеджирования при помощи открытия коротких позиций на лендинговом протоколе, вообще без использования криптовалютных бирж. И хотя в целом такой способ возможен, наше исследование показало, что и в этом случае есть скрытые комиссии и никого выигрыша использование децентрализованных инструментов не дает.

Моделирование доходности динамического фьючерса показано на рисунке 10. Блок схема алгоритма для данного метода показана на рис. 11

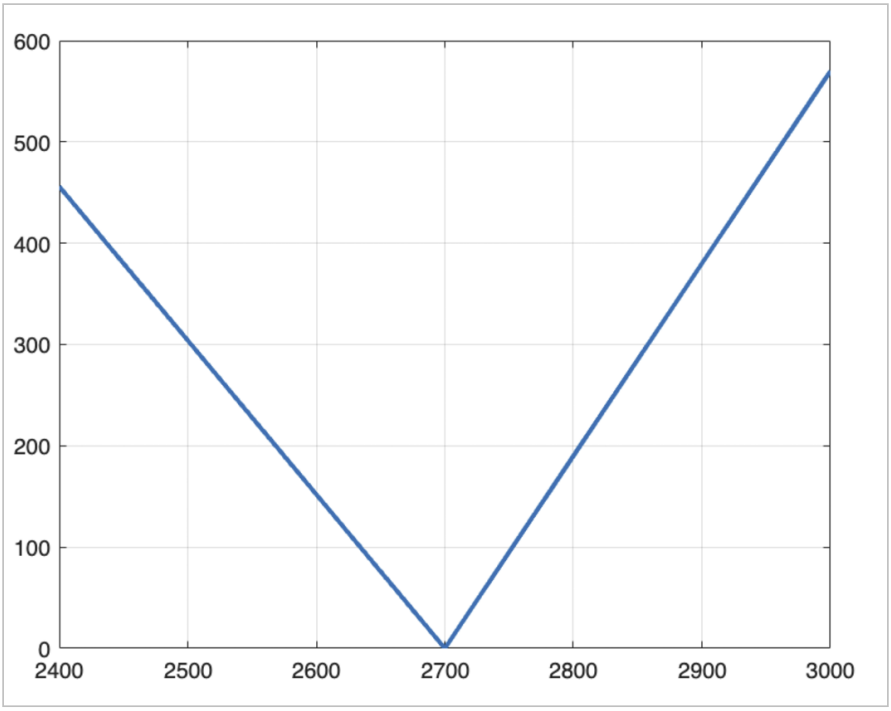


Рис. 10. Доходность алгоритма динамического хеджирования

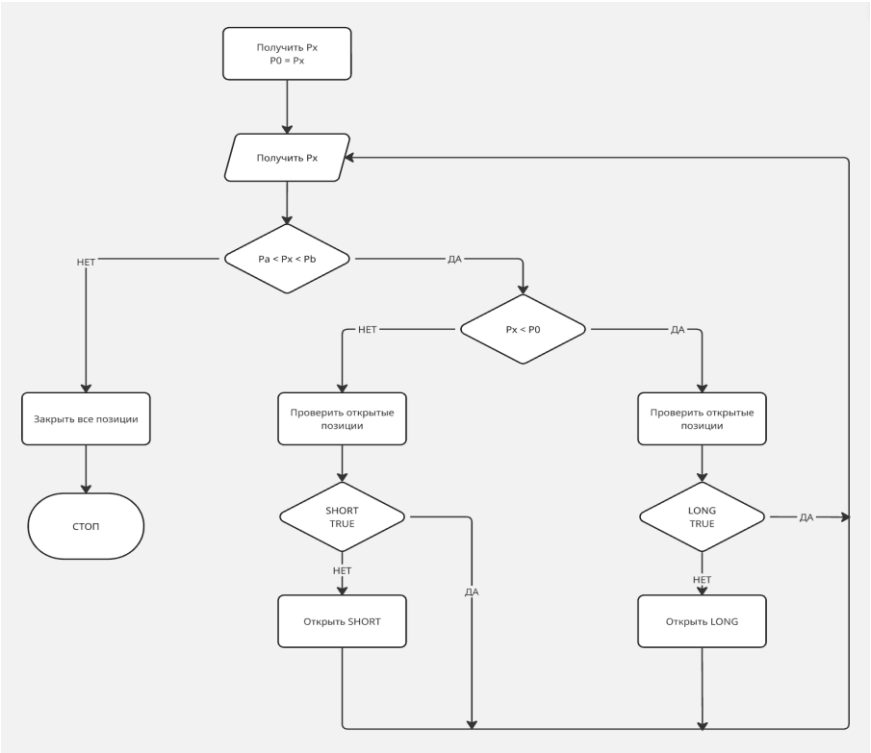


Рис. 11. Блок схема алгоритма метода динамического хеджирования

Результат применения динамического хеджирования показан на рис. 11. Также на графике показано изменение стоимости активов при пассивном владении, без внесения в пул ликвидности.

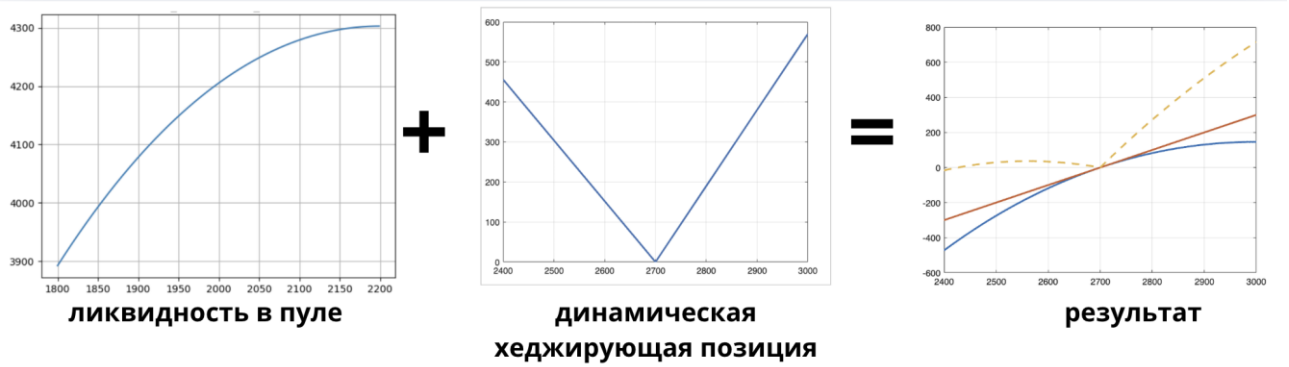


Рис. 12. Результат применения динамического хеджирования

Как видно из графика, нам удалось полностью хеджировать непостоянные потери во всех ценовых диапазонах.

Здесь необходимо отметить, что хотя данный метод хеджирования в теории выглядит весьма многообещающим, на практике мы столкнулись с серьезной проблемой в работе хэджирующей позиции. Суть проблемы состоит в том, что цена может очень часто проходить через точку входа p_0 хеджирующей позиции, что влечет за собой частую уплату комиссий за закрытие и открытие позиций и, что гораздо важнее, фиксация отрицательной доходности фьючерсной позиции в момент его закрытия после пересечения цены p_0. Таким образом, хотя в теории данный метод показывает отличные результаты, на практике он должен применяться крайне осторожно и только после тщательного анализа открываемой позиции.

РАЗРАБОТКА ПРИЛОЖЕНИЯ

Используем стек

При разработке приложения использовался следующий стек технологий:

Математическое моделирование. Моделирование проводилось с использованием языка программирования Python и библиотеки matplotlib.

Скрипты управления ликвидностью. Все скрипты написаны на языке Python (библиотеки ccxt, numpy, pandas, math, time). Для взаимодействия в криптовалютными биржами используется библиотека ccxt.

Фронтенд. Фронтенд реализован с использованием HTML, CSS, JavaScript. Все страницы фронтенда используют общий CSS файл.

Бэкенд. Взаимодействие между фронтендом и бэкендом реализовано при помощи API на фреймворке FastAPI.

Стартовый скрипт. Для запуска приложения создан shell скрипт.

Оповещение пользователя. Обратная связь с пользователем также дублируется с помощью Telegram бота.

Архитектура приложения

Структурно приложение состоит из следующих элементов (рис. 13).

Фронтенд

Браузерное приложение, состоящее из отдельных html страниц со встроенными JS скриптами и использующими общий CSS файл (styles.css). Фронтенд связывается с бэкендом посредством GET и POST запросов. Включает в себя следующие страницы:

- Страница для моделирования Непостоянных потерь.
- Страница для моделирования хеджирования методом Короткого фьючерса.
- Страница для моделирования хеджирования методом Динамического хэджирования.
- Страница для открытия Новой позиции

На всех страницах используется JS скрипты для получения значений из полей ввода (<input>) и их использования в других полях. При этой последовательно используются следующие JS функции:

`document.getElementById('p0')` - для чтения введенного значения их поля

`parseFloat(inputFieldP0.value)` - для получения числового значения ввода

`isNaN(x0)` - для проверки на валидность введенных данных

`addEventListener()` - для назначения обработчика события к элементу

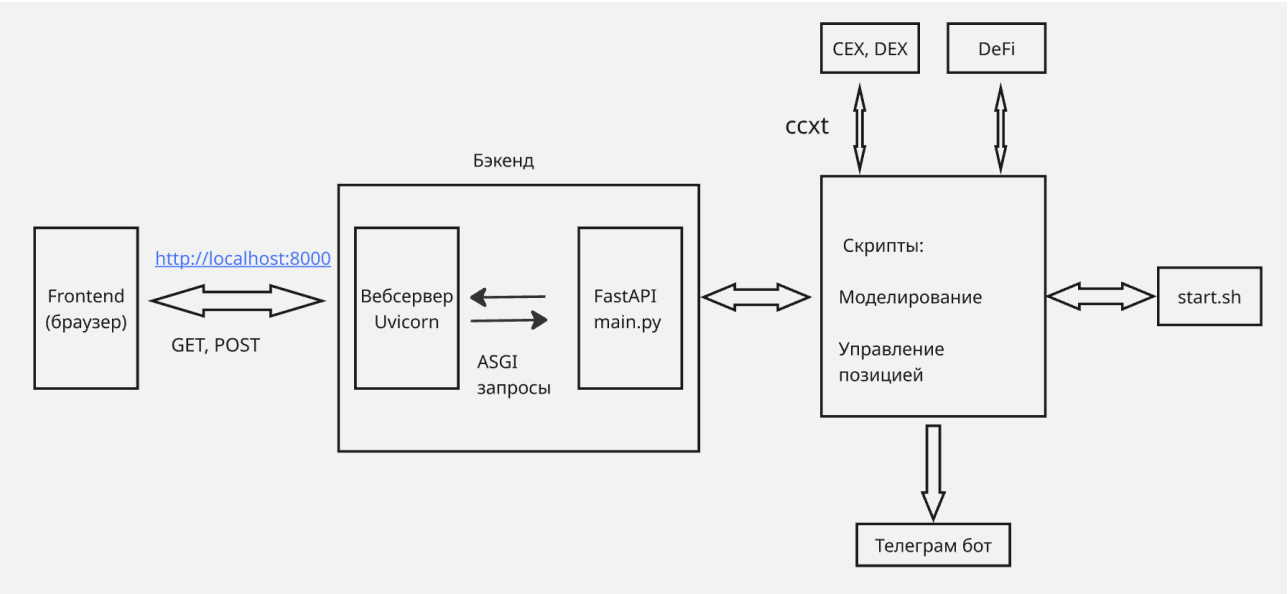


Рис.13. Архитектура приложения

Для взаимодействия с бэкендом используется фреймворк FastAPI. Поскольку это ключевой элемент для взаимодействия фронтенда и бекенда приложения, рассмотрим его работу подробнее. Рассмотрим работу FastAPI на примере страницы Sim_ImL.html. На фронтенде создана асинхронная функция `async function calculate()`. Асинхронные функции не блокируют основной поток, что в нашем случаем может оказаться критично, т. к. мы обращаемся к удаленным серверам бирж и на получение ответа может потребоваться значительное время. Функция выполняет следующие операции:

Получение данных из полей и выделение числовой части:

```
document.getElementById('имя поля');
parseFloat(document.getElementById('имя поля').value)
```

Затем происходит валидация данных:

```
isNaN(переменная)
```

И если все в порядке, формируются данные для отправки *requestData* (тип данных в JS «объект»). Далее функция отправляется запрос на бэкенд:

```
const response = await fetch('http://localhost:8000/calculate', {
        method: 'POST',
        headers: {
        'Content-Type': 'application/json',
        },
        body: JSON.stringify(requestData)
});
```

В запросе используется встроенная функция браузера *fetch()* для отправки http запросов. Далее указывается метод http запроса (*POST)* для отправки на сервер. В заголовке http запроса серверу указывается, что запрос содержит JSON. После получения запроса FastAPI

автоматически распарсит JSON в тип данных Python словарь. В теле запроса JavaScript объект requestData преобразуется в JSON строку.

После выполнения запроса приходит ответ *response* - объект, содержащий ответ от FastAPI сервера. Далее следует проверка ответа на содержание ошибок осуществляется:

```
if (response.ok) {}
```

Далее процедура парсит ответ JSON и формирует результат:

```
const responseData = await response.json();
```

Полный листинг страниц фронтенда находится в Приложении.

Бэкенд

Бэкенд (файл main.py) отвечает за взаимодействие с фронтендом посредством POST и GET запросов через FastAPI и вызов модулей для расчета моделей и управления позициями. Для его работы необходима установка и запуск ASGI сервера Uvicorn. Опишем основные компоненты бэкенда.

```
app = FastAPI()
```

Создаем экземпляр FastAPI приложения.

Затем решаем вопрос с CORS ошибкой, которая не позволит принимать запросы из неизвестных источников:

```
app.add_middleware(
CORSMiddleware,
allow_origins=["*"],
allow_credentials=True,
allow_methods=["*"],
allow_headers=["*"],
)
```

Далее используем импортированную модель Pydantic для валидации входящих данных первого эндпоинта (обработчика):

```
class CalculationData(BaseModel):
    variable: type
```

И запускаем эндпоинт для расчета непостоянных потерь:

```
@app.post("/calculate")
async def calculate(data: CalculationData):
```

Асинхронная функция `async def calculate(data: CalculationData):` принимает данные класса CalculationData. Далее вызываем импортированную функцию `ImpermLossCalc()` из внешнего модуля для расчета непостоянных потерь:

```
Xa, Yb, Y0, V0, LOSSA, LOSSB, image = ImpermLossCalc(data.X0, data.P0, data.P1,
```

```
data.P2, data.APR, data.Loss)
```

И, наконец, функция возвращает JSON клиенту:

```
return {
    "status": "success",
    "results": {
    "Xa": Xa,
    "Yb": Yb,
    "Y0": Y0,
    "V0": V0,
    "LOSSA": LOSSA,
    "LOSSB": LOSSB,
    "image": image,
}
```

По такому алгоритму работают все обработчики в бэкенде. Документация реализованного API с активными обработчиками (handlers) доступна по адресу http://localhost:8000/docs и показана на рис. 14.

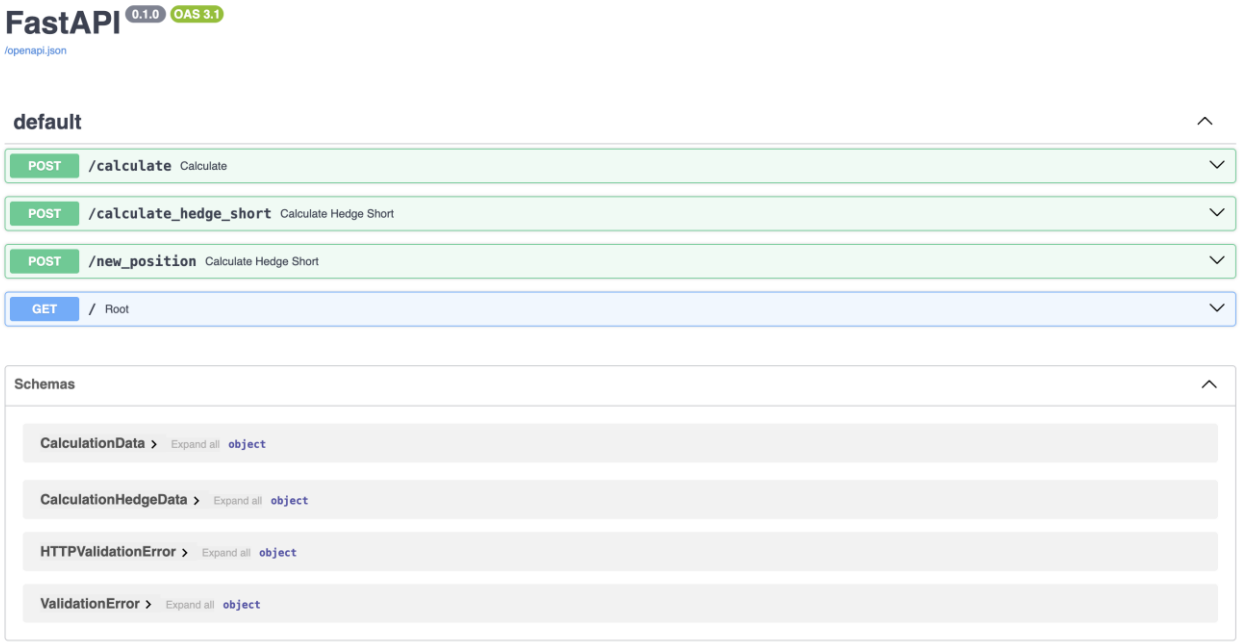


Рис. 14. Документация FastAPI со списком используемых обработчиков

В API доступны следующие обработчики (часто также называемые «ручками»):

calculate — получение данных из фронтенда для вызова процедуры ImpermLossCalc(), вызов процедуры, отправка результатов на фронтенд.

calculate_hedge_short - получение данных из фронтенда для вызова процедуры ShortHedgeCalc( ) , вызов процедуры, отправка результатов на фронтенд.

new_position - получение данных из фронтенда для вызова процедуры NewPosition() , вызов процедуры, отправка результатов на фронтенд.

Скрипты для моделирования и управления

Скрипты для моделирования управления позициями представляют собой программы, написанные на Python. Модули импортированы в бэкенд main.py. Все процедуры запускаются непосредственно из main.py. Включают следующие модули:

- Скрипт с процедурами для моделирования Непостоянных потерь (ImL_Calc.py). Функция ImpermLossCalc().
- Скрипт с процедурами для моделирования Хэджирования (ImL_Hedge_Short.py). Функция ShortHedgeCalc().
- Скрипт для открытия новой позиции (New_Position.py). Функция NewPosition().

Для работы с криптовалютными биржами используется библиотека ccxt. Для обращения к бирже сначала создается объект, для чего потребуется публичный ключ криптокошелька и приватный ключ API для доступа к бирже. Покажем на примере подключения в децентрализованной бирже Hyperliquid:

```
dex = ccxt.hyperliquid({
"walletAddress": "0x1111111111111111111111111111111111111111", # /!\ Public address of your account/wallet
"privateKey": "0x1111111111111111111111111111111111111111",
})
```

Далее можно пользоваться методами созданного объекта. Например, чтобы запросить список всех открытых позиций по монете используется метод fetch_positions:

```
positions = dex.fetch_positions([symbol])
```

Скрипт для Телеграм кода

Данный скрипт вызывается из бэкенда при наступлении критически важных событий:

- Открытие позиции
- Изменение размера позиции
- Выход из диапазона
- Закрытие позиции

Для инициализации бота задаются BOT_TOKEN и CHAT_ID. Данный бот используется только для отправки данных. Из соображений безопасности было решено исключить возможность получения данных телеграм ботом.

Скрипт для перезапуска бэкенда

В ходе работы над скриптом мы столкнулись с проблемой обеспечения стабильной связи с API криптобиржи или децентрализованного протокола. При обращении к удаленной системе команды периодически завершилась с ошибкой, и работа скрипта прекращалась. Для решения данной проблемы была использования программа watchdog. Данная программа

представляет собой shell скрипт, который вызывает основную программу в бесконечном цикле. В случае аварийного прерывания исполнения основной программы, скрипт перезапускает основную программу. Поскольку при перезапуске, программа должна продолжить свою работу с теми параметрами, с которыми она была прервана, при каждом итерации текущие параметры скрипта записывайся в файл. При перезапуске текущие параметры считываются из файла.

Запуск скрипта для открытия новой позиции происходит из bash скрипта. Скрипт исполняет бесконечный цикл пока основная программа работает. Если проверка на закрытие программы дает положительный ответ, скрипт перезапускает программу. Команда проверки завершения программы:

```
EXIT_CODE=$?
```

При этом так же отправляет уведомление пользователю в Телеграм бот.

Исходные коды всех модулей программы находятся в Приложении 1.

Описание некоторых основных страниц фронтенда.

Страница моделирования Непостоянных потерь.

На странице «Impermanent Loss Simulator» (рис. 15) пользователь вводит следующие параметры:

- Начальная цена: P0
- Размер позиции: X0
- Нижняя цена диапазона: Pa (рассчитывается % от цены P0)
- Верхняя цена диапазона: Pb (рассчитывается % от цены P0)
- Годовая доходность позиции: APR
- Допустимые потери для выбора точки закрытия позиции: (Loss)

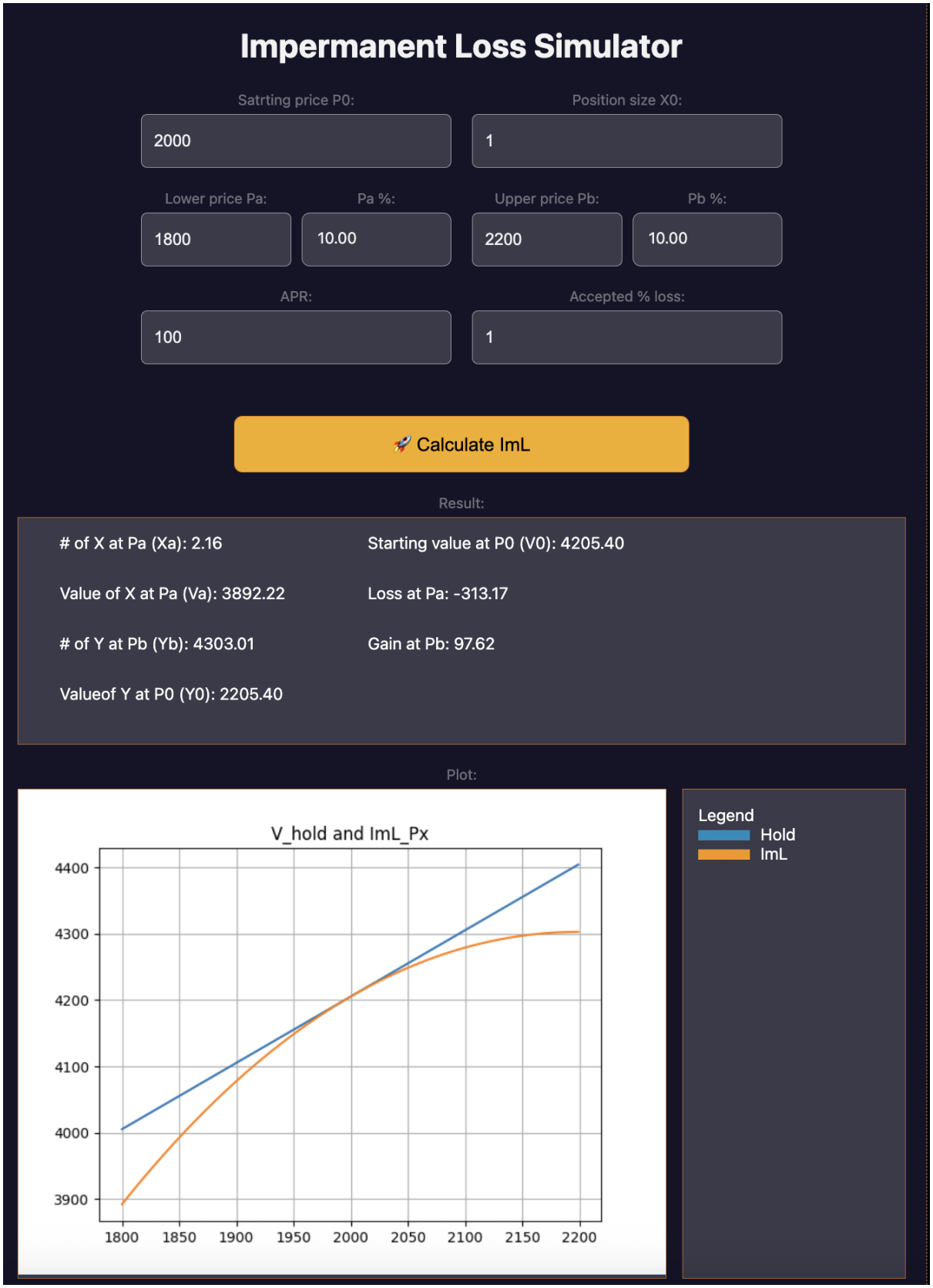


Рис.15. Страница моделирования Непостоянных потерь

Кнопка «Рассчитать ImL» (Calculate ImL) вызывает процедуру ImpermLossCalc() модуля Sim_ImL.html. На выходе пользователь получает расчет Непостоянных потерь в пуле ликвидности и графики с доходностью Непостоянных потерь (ImL) и пассивного владения

(Hold). Приложение рассчитывает следующие параметры:

- Количество актива X в нижней границе ценового диапазона: Xa
- Общий объем позиции при цене Pa: Va
- Количество актива Y в верхней границе ценового диапазона: Yb
- Общий объем позиции при цене Pb: Vb
- Количество актива Y в точке входа (функция от X0): Y0
- Общий объем позиции в точке входа (функция от X0): V0
- Потери при цене Pa: Loss_Pa
- Прибыль при цене Pb: Gain_Pb

Страница моделирования хеджирования Коротким фьючерсом

Рис. 16. Страница моделирования Хеджирования коротким фьючерсом

На странице «Short Hedging Simulator» пользователь вводит следующие параметры:

- Начальная цена: P0
- Размер позиции: X0
- Нижняя цена диапазона: Pa (рассчитывается % от цены P0)
- Верхняя цена диапазона: Pb (рассчитывается % от цены P0)
- Годовая доходность позиции: APR
- Допустимые потери для выбора точки закрытия позиции: (Loss)
- Плечо для хеджирующей позиции: Leverage
- Коэффициент размера хеджа: K

Кнопка «Рассчитать ImL + Хеджирование» (Calculate ImL + Hedge) вызывает процедуру ShortHedgeCalc( ) из модуля SimHedgeShort.html. На выходе пользователь получает расчет Непостоянных потерь в пуле ликвидности и графики с доходностью Непостоянных потерь (ImL), пассивного владения (Hold), доходность позиции с хеджированием и цены, при которых достигаются допустимые потери. Приложение рассчитывает следующие параметры:

- Количество актива X в нижней границе ценового диапазона: Xa
- Общий объем позиции при цене Pa: Va
- Количество актива Y в верхней границе ценового диапазона: Yb
- Общий объем позиции при цене Pb: Vb
- Количество актива Y в точке входа (функция от X0): Y0
- Общий объем позиции в точке входа (функция от X0): V0
- Потери при цене Pa: Loss_Pa
- Прибыль при цене Pb: Gain_Pb
- Потери в точке допустимых потерь: X_delta_abs
- Размер позиции в точке допустимых потерь: X_limit
- Нижняя цена точки допустимых потерь: Pa_loss
- Верхняя цена точки допустимых потерь: Pb_loss
- Число дней для выхода в безубыток: days

Графики при моделировании создаются в процедурах при помощи библиотеки mstplotlib. Для передачи изображения на фронтенд изображения форматируются в base64, сохраняются в buf, и т.о. передаются на фронтенд без создания дополнительных файлов на диске.

Страница для открытия новой позиции

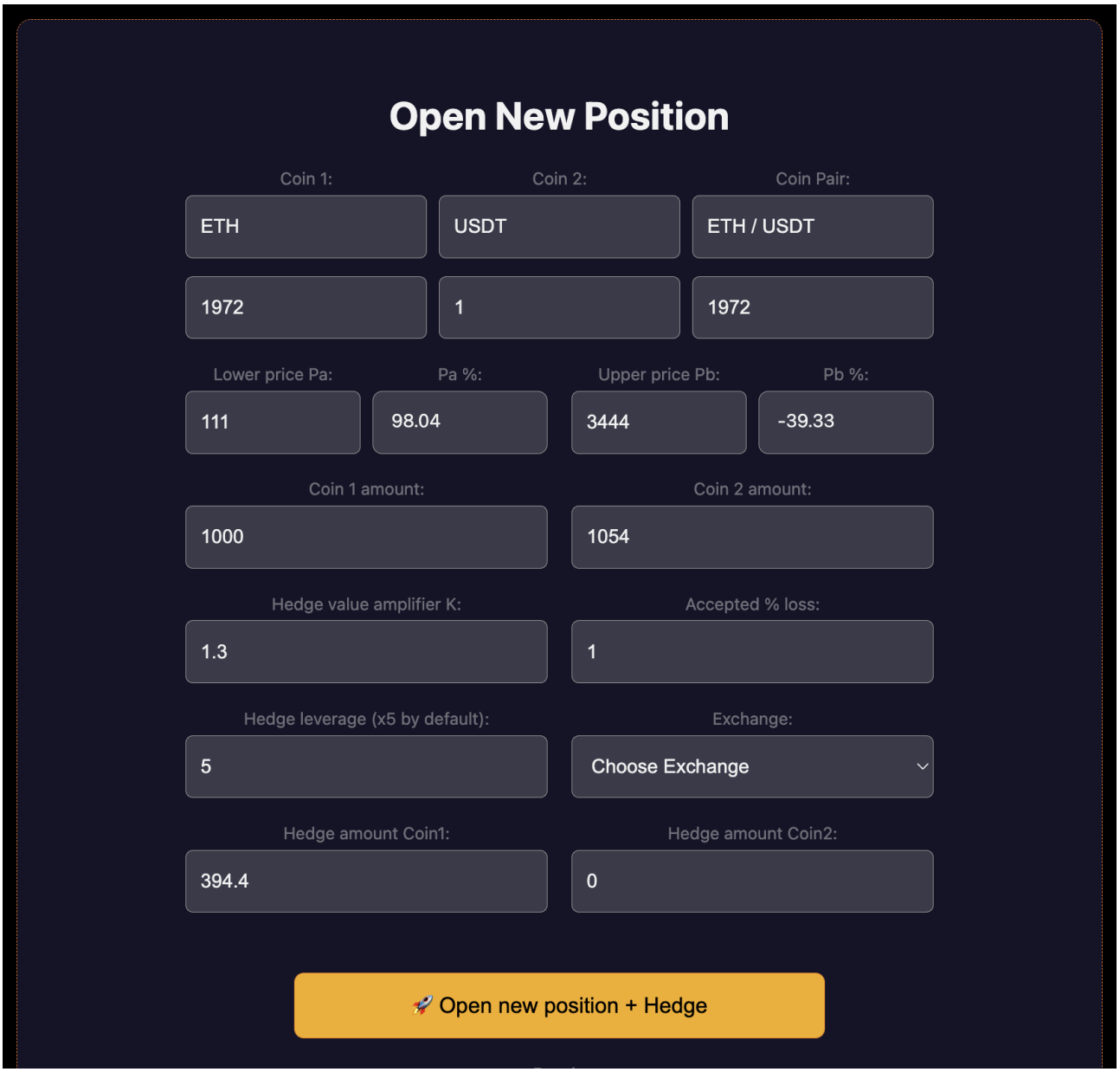


Рис. 17. Страница открытия новой позиции

На данной странице пользователь вводит следующие данные:

- Первая и вторая монеты в пуле: Coin1 и Coin2. На основе введенных данных запрашивается актуальная цена данных монет (запрос через API отправляется на сайт Coinmarketcap) и рассчитывается кросс-курс для данной пары активов P0.
- Нижняя цена диапазона: Pa (также рассчитывается % от цены P0)
- Верхняя цена диапазона: Pb (также рассчитывается % от цены P0)
- Число монет Coin1 (число монет Coin2 рассчитывается автоматически)
- Годовая доходность позиции: APR
- Допустимые потери для выбора точки закрытия позиции: (Loss)
- Плечо для хеджирующей позиции: Leverage

- Коэффициент размера хеджа: K
- Биржа для размещения хеджирующей позиции (CEX / DEX)

В поле «Размер хеджирующей позиции» (Hedge amount Coin1:) автоматически подставляется объем ликвидности, необходимой для открытия хеджирующей позиции (рассчитывается на основании данных об объеме позиции, плеча и хеджирующего коэффициента).

После нажатия кнопки «Открыть новую позицию + Хеджирование» (Open new position + Hedge) запускается процедура NewPosition() из модуля NewPositionHedge.html. Данная процедура работает как непрерывный цикл, блок схема алгоритма показана на рис. 9.

ЗАКЛЮЧЕНИЕ

В данной работе были проанализированы три метода хеджирования непостоянных потерь. Нам удалось показать, что метод хеджирования короткой фьючерсной позицией и метод динамического хеджирования позволяет снизить или частично устранить непостоянные потери. В то же время, данные методы не могут идеально хеджировать Непостоянные потери из-за их нелинейной природы. В рассмотренных методах короткие позиции открывались при помощи разработанного нами приложения «Мастер Ликвидности». Тестирование проводилось на децентрализованной криптовалютной бирже (DEX) Hyperliquid и на централизованной криптовалютной бирже (CEX) Bybit.

Разработанные методы не лишены недостатков. К недостаткам метода хеджирования короткой фьючерсной позицией следует отнести невозможность полного устранения Непостоянных потерь. По мере приближения цены к границам диапазона потери увеличиваются и должны быть компенсированы за счет заработанных комиссий. Данный метод может оказаться убыточным в случае быстрого выхода цены из рабочего диапазона.

Метод динамического хеджирования позволяет сколь угодно точно воспроизвести нелинейный характер графика доходности позиции в пуле ликвидности и обеспечить максимально точное хеджирование Непостоянных потерь. Точность данного метода определяется частотой дискредитации хеджирующего графика, что позволяет обеспечить любую требуемую степень приближения к нелинейному графику изменения ликивдности в пуле. Недостатком данного метода является неизбежные потери в хеджирующей позиции из-за необходимости постоянно менять размер шорта и как следствие регулярно фиксировать отрицательную доходность позиции.

Все разработанные методы хеджирования реализуются в автоматическом режиме при помощи приложения «Мастер Ликвидности». В ходе работы над данный инструментов были решены задачи подключения к DEX и CEX биржам через API, обеспечение стабильности работы приложения при обрыве соединения через API, получение обратной связи о работе приложения через Телегам-бот, реализована связь фронтенда и бэкенда через fastAPI и управление настройками приложения через интерфейс на фронтенде.

Также в работе намечены дальнейшие направления исследований по данной теме, такие как использование Искусственного Интеллекта для анализа рыночной ситуации и выбора оптимального в данный момент способа хеджирования. Также в планах использование Docker контейнеров для обеспечения стабильности приложения и адаптация приложения для работы с различными аккаунтами пользователей.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Hayden Adams, Noah Zinsmeister, Moody Salem, River Keefer, Dan Robinson Uniswap v3 Core. 2021
   https://app.uniswap.org/whitepaper-v3.pdf
2. What Is GammaSwap? It's The Degen's Dream
   https://www.blocmates.com/articles/gammaswap-it-s-the-degen-s-dream
3. Dave WhiteDan, Robinson Zubin, Koticha Andrew, Leone Alexis, Gauba Aparna Krishnan. Power Perpetuals. 2021
   https://www.paradigm.xyz/2021/08/power-perpetuals
4. Hedging Impermanent Loss with Power Perpetuals
   https://docs.deri.io/library/academy/power-perpetuals/hedging-impermanent-loss-with-power-perpetuals-1
5. Peteris Erins.
6. Impermanent Loss in Uniswap V3. 2021
   https://medium.com/auditless/impermanent-loss-in-uniswap-v3-6c7161d3b445
7. Uniswap V3 Development Book
   https://uniswapv3book.com/milestone_1/providing-liquidity.html
8. Как устроен AMM — Automated Market Maker
   https://habr.com/ru/companies/metalamp/articles/749538/
9. Непостоянные потери в Uniswap V2 + V3
   https://habr.com/ru/articles/841586/

ПРИЛОЖЕНИЕ

Исходный код бекенда main.py

```
from fastapi import FastAPI

from fastapi.middleware.cors import CORSMiddleware
from pydantic import BaseModel
import uvicorn
from ImL_Calc import ImpermLossCalc
from ImL_Hedge_Short import ShortHedgeCalc
import base64

app = FastAPI()

# Настройка CORS
app.add_middleware(
CORSMiddleware,
allow_origins=["*"],
allow_credentials=True,
allow_methods=["*"],
allow_headers=["*"],

)
# Модель для входящих данных - только P0, Pa, Pb, X0
class CalculationData(BaseModel):
P0: float
P1: float
P2: float
X0: float
APR: float
Loss: float
#Leverage1: float
#Leverage2: float


@app.post("/calculate")
async def calculate(data: CalculationData):
"""
Эндпоинт для расчета - принимает P0, P1, P2 и возвращает результат
"""
print ('from main.py')
print(f"📥 Получены значения: P0={data.P0}, P1={data.P1}, P2={data.P2}, X0={data.X0}, APR={data.APR}, Loss={data.Loss}" )
# Call external def
Xa, Yb, Y0, V0, LOSSA, LOSSB, image = ImpermLossCalc(data.X0, data.P0, data.P1, data.P2, data.APR, data.Loss)

print ('ImL Calculation Data ')
print(f"✅ Результат Xa: {Xa}")
print(f"✅ Результат Yb: {Yb}")
print(f"✅ Результат V0: {Y0}")
print(f"✅ Результат V0: {V0}")
print(f"✅ Результат LOSSA: {LOSSA}")
print(f"✅ Результат LOSSB: {LOSSB}")

return {
```

```
"status": "success",
"input_values": {
"P0": data.P0,
"P1": data.P1,
"P2": data.P2,
"X0": data.X0,

},
"results": {
"Xa": Xa,
"Yb": Yb,
"Y0": Y0,
"V0": V0,
"LOSSA": LOSSA,
"LOSSB": LOSSB,
"image": image,
},
"message": "ImL calculated"
}
#------------------------- Short Hedge ------------------------
class CalculationHedgeData(BaseModel):
P0: float
P1: float
P2: float
X0: float
APR: float
Loss: float
Leverage: float
Hedge_k: float

@app.post("/calculate_hedge_short")
async def calculate_hedge_short(data: CalculationHedgeData):
print ('Yo!')
print(f"📥 Получены значения: P0={data.P0}, P1={data.P1}, P2={data.P2}, X0={data.X0}, APR={data.APR}, Loss={data.Loss},
Leverage1{data.Leverage}, Hedge_k{data.Hedge_k}" )
# Call external def
Xa, Yb, Y0, V0, LOSSA, LOSSB, image_base64, X_delta_abs, X_limit, Pa_loss, Pb_loss, days = ShortHedgeCalc(data.X0, data.P0,
data.P1, data.P2, data.APR, data.Loss, data.Leverage, data.Hedge_k)
#print ('Starting Value of X (X0) = ', X0*P0)
print ('Hedge Calculation Data ')
print(f"✅ Результат Xa: {Xa}")
print(f"✅ Результат Yb: {Yb}")
print(f"✅ Результат V0: {Y0}")
print(f"✅ Результат V0: {V0}")
print(f"✅ Результат LOSSA: {LOSSA}")
print(f"✅ Результат LOSSB: {LOSSB}")
print(f"✅ Результат X_delta_abs: {X_delta_abs}")
print(f"✅ Результат X_limit: {X_limit}")
print(f"✅ Результат Pa_loss: {Pa_loss}")
print(f"✅ Результат Pb_loss: {Pb_loss}")
print(f"✅ Результат days: {days}")


return {
```

```
"status": "success",
"input_values": {
"P0": data.P0,
"P1": data.P1,
"P2": data.P2,
"X0": data.X0,

},
"results": {
"Xa": Xa,
"Yb": Yb,
"Y0": Y0,
"V0": V0,
"LOSSA": LOSSA,
"LOSSB": LOSSB,
"image": image_base64,
"X_delta_abs": X_delta_abs,
"X_limit": X_limit,
"Pa_loss": Pa_loss,
"Pb_loss": Pb_loss,
"days": days
},
"message": "ImL + Short Hedge calculated"
}

#X_delta_abs, X_limit, Pa_loss, Pb_loss
#------------- new position -----------------------------------

class NewPosition(BaseModel):
P0: float
P1: float
P2: float
X0: float
APR: float
Loss: float
Leverage: float
Hedge_k: float

@app.post("/new_position")
async def calculate_hedge_short(data: CalculationHedgeData):

P0 = 1
P1 = 1
P2 = 1
X0 = 1
return {
"status": "success",
"input_values": {
"P0": data.P0,
"P1": data.P1,
"P2": data.P2,
"X0": data.X0,

},
```

```
"results": {
"Xa": P0,
"Yb": P1,
"Y0": P2,
"V0": X0,

},
"message": "ImL + Short Hedge calculated"
}
#-----------------------------------------------------------------
@app.get("/")
async def root():
return {
"message": "Сервер работает",
"endpoints": {
"POST /calculate": "Отправить P0, P1, P2 для расчета"
}
}


if __name__ == "__main__":
uvicorn.run("main:app", reload=True)
```

Исходный код фронтенда страницы Sim_ImL.html

```
<!DOCTYPE html>

<html lang="ru">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="./styles.css">
<title>Liquidity Modeling | DeFi Yield Simulator</title>


</head>

<body>

<header>
<div class="header-container">
<img src="./pics/LM_logo.png" alt="Liquidity Master Logo" class="logo">

<nav>
<div class="dropdown">
<a href="#" class="menu-item">Simulate ▼</a>
<div class="submenu">
<a href="./Sim_ImL.html">Impermanent Loss Calc</a>
<a href="./Sim_Hedge_Short.html">Short Hedging</a>
<a href="./Sim_Hedge_Dyn.html">Dinamic Hedging</a>
</div>
```

```
</div>

<a href="./New_position.html" class="menu-item">New position</a>
<a href="#" class="menu-item">My positionы</a>
<a href="#" class="menu-item">Docs</a>
</nav>

</div>

</header>

<main>
<div class="content-placeholder">
<h1>Impermanent Loss Simulator</h1>
<p style="margin-top:20px;color:var(--muted);">
<!---------------------------- new Input fields ---------------- -->
<div class="input-group">
<div style=" display:flex; align-items: center; justify-content:center; gap: 20px;">
<div class="input-row" >
<div class="field_name">Satrting price P0:</div>
<input class="PInput" type="number" id="p0" step="any" placeholder="Enter P0" >
</div>

<div class="input-row" >
<div class="field_name">Position size X0:</div>
<input class="PInput" type="number" id="x0" step="any" placeholder="Enter X0" >
</div>
</div>

<!------------------------------------------------ Pa, Pb input row. ---------------------------------------------- -->
<div style=" display:flex; align-items: center; justify-content:center; gap: 20px;">
<div style=" display:flex; align-items: center; justify-content:center; gap: 10px;">
<div class="input-row">
<div class="field_name">Lower price Pa:</div>
<input class="PInput_range" type="number" id="p1" step="any" placeholder="Enter Pa" >
</div>
<div class="input-row">
<div class="field_name">Pa %:</div>
<div class="Poutput" id="Pa%" placeholder="Pa %" step="any" type="number" >
<p style="text-align: center; color: #807979; " > Pa %</p>
</div>
</div>
</div>

<div style=" display:flex; align-items: center; justify-content:center; gap: 10px;">
<div class="input-row">
<div class="field_name">Upper price Pb:</div>
<input class="PInput_range" type="number" id="p2" step="any" placeholder="Enter Pb" >
</div>
<div class="input-row">
<div class="field_name">Pb %:</div>
<div class="Poutput" id="Pb%" placeholder="Pb %" step="any" type="number" >
<p style="text-align: center; color: #807979; " > Pb %</p>
</div>
```

```
</div>
</div>

</div>
<!------------------------------------------------------------ ------------------------------------------ -->


<div style=" display:flex; align-items: center; justify-content:center; gap: 20px;">
<div class="input-row">
<div class="field_name">APR:</div>
<input class="PInput" type="number" id="apr" step="any" placeholder="Enter APR" >
</div>
<div class="input-row">
<div class="field_name">Accepted % loss:</div>
<input class="PInput" type="number" id="loss" step="any" placeholder="% loss" >
</div>
</div>

<!-- ---------------------------------- % calculation script ---------------------------------- -->


<script> /*change Pa from abs to % */
const inputFieldP0 = document.getElementById('p0');
const inputFieldPa = document.getElementById('p1');
const outputFieldPa = document.getElementById('Pa%');
const inputFieldPb = document.getElementById('p2');
const outputFieldPb = document.getElementById('Pb%');

inputFieldPa.addEventListener('input', () => {
const val_p0 = parseFloat(inputFieldP0.value) ;
const val_pa = parseFloat(inputFieldPa.value) ;

if (isNaN(val_pa) || isNaN(val_p0) || val_p0 === 0) {
outputFieldPa.textContent = "0.00";
} else {
const percentage = ((val_p0 - val_pa) / val_p0 * 100).toFixed(2);
outputFieldPa.textContent = percentage;
}
});


inputFieldPb.addEventListener('input', () => {
const val_p0 = parseFloat(inputFieldP0.value) ;
const val_pb = parseFloat(inputFieldPb.value) ;

if (isNaN(val_pb) || isNaN(val_p0) || val_p0 === 0) {
outputFieldPb.textContent = "0.00";
} else {
const percentage = ((val_pb - val_p0) / val_p0 * 100).toFixed(2);
outputFieldPb.textContent = percentage;
}
});
</script>
```

```
</div>
<!-- -------------- end of Input ------------------------------- -->


<!----------------------------------- Calc button ------------------->
<div style=" gap: 10px; align-items: center; padding:15px 40px;">
<button class="execute-button" id="calcBtn" onclick="calculate()">🚀 Calculate ImL</button>
</div>

<!------------------------------------- "result_window" --------------------->
<div style="align-items: center; justify-content:center; ">
<div class="input-row">
<div class="field_name">Result:</div>
<div class="result_window" id="result" >
<p style="text-align: center; color: #807979; padding:15px 40px;" > Calculation results</p>
</div>
</div>
<div class="input-row">
<div class="field_name">Plot:</div>
<div style=" display:flex; gap: 15px;">
<div class="plot_window">
<img id="plotImage" style="max-width:100%;" />
</div>
<div class="legend_window">
<p>Legend</p>
<div><img src="./pics/lines/blue.png" width="50" height="10" style="margin-right: 10px;">Hold</div>
<div><img src="./pics/lines/orange.png" width="50" height="10" style="margin-right: 10px;">ImL</div>
</div>
</div>



</div>

</div>

<div style="margin-top: 20px;">
<!-- h4 style="margin-bottom: 5px;">📥 Servre response:</h4 -->
<div style="display: none;" id="responseData" class="json-view">{}</div>
</div>

<script>
// Основная функция расчета
async function calculate() {
const button = document.getElementById('calcBtn');
const resultDiv = document.getElementById('result');
const sentDataDiv = document.getElementById('sentData');
const responseDataDiv = document.getElementById('responseData');
const plotDiv = document.getElementById('plot');
// Получаем значения из полей
const p0 = parseFloat(document.getElementById('p0').value);
const p1 = parseFloat(document.getElementById('p1').value);
const p2 = parseFloat(document.getElementById('p2').value);
const x0 = parseFloat(document.getElementById('x0').value);
const apr = parseFloat(document.getElementById('apr').value);
```

```
const loss = parseFloat(document.getElementById('loss').value);
// Проверка на валидность
if (isNaN(p0) || isNaN(p1) || isNaN(p2) || isNaN(x0) || isNaN(apr) || isNaN(loss)) {
resultDiv.className = 'result error';
resultDiv.innerHTML = '<h3>❌ Error</h3><p>Fill all fields</p>';
return;
}
// Формируем данные для отправки
const requestData = {
P0: p0,
P1: p1,
P2: p2,
X0: x0,
APR: apr,
Loss: loss,
};
// Показываем отправляемые данные
//sentDataDiv.textContent = JSON.stringify(requestData, null, 2);
// Блокируем кнопку
button.disabled = true;
button.textContent = '⏳ Расчет...';
// Показываем статус
resultDiv.className = 'result';
resultDiv.innerHTML = '<h3>⏳ Выполняется расчет...</h3>';
console.log('📤 Отправляем данные:', requestData);
try {
// Отправляем POST запрос
const response = await fetch('http://localhost:8000/calculate', {
method: 'POST',
headers: {
'Content-Type': 'application/json',
},
body: JSON.stringify(requestData)
});
// Получаем ответ
const responseData = await response.json();
console.log('📥 Получен ответ:', responseData);
// Показываем ответ
responseDataDiv.textContent = JSON.stringify(responseData, null, 2);
if (response.ok) {
// Успешный расчет
resultDiv.className = 'result success';
resultDiv.innerHTML = `<h3>❌ Got results ----------</h3>`;

// Форматируем результат
//const result = responseData.result;
const Xa = responseData.results.Xa;
const Yb = responseData.results.Yb;
const Y0 = responseData.results.Y0;
const V0 = responseData.results.V0;
const LOSSA = responseData.results.LOSSA;
const LOSSB = responseData.results.LOSSB;
const Va = responseData.results.Xa * p1;
const formattedXa = typeof Xa === 'number' ? Xa.toFixed(2) : Xa;
```

```
const formattedVa = typeof Va === 'number' ? Va.toFixed(2) : Va;
const formattedYb = typeof Yb === 'number' ? Yb.toFixed(2) : Yb;
const formattedY0 = typeof Y0 === 'number' ? Y0.toFixed(2) : Y0;
const formattedV0 = typeof V0 === 'number' ? V0.toFixed(2) : V0;
const formattedLOSSA = typeof LOSSA === 'number' ? LOSSA.toFixed(2) : LOSSA;
const formattedLOSSB = typeof LOSSB === 'number' ? LOSSB.toFixed(2) : LOSSB;

resultDiv.innerHTML = `
<div class="result_window" id="result" >
<div>
<p style="text-align: left; padding:15px 40px;" > # of X at Pa (Xa): ${formattedXa}</p>
<p style="text-align: left; padding:15px 40px;" > Value of X at Pa (Va): ${formattedVa}</p>
<p style="text-align: left; padding:15px 40px;" > # of Y at Pb (Yb): ${formattedYb}</p>
<p style="text-align: left; padding:15px 40px;" > Valueof Y at P0 (Y0): ${formattedY0}</p>
</div>
<div>
<p style="text-align: left; padding:15px 40px;" > Starting value at P0 (V0): ${formattedV0}</p>
<p style="text-align: left; padding:15px 40px;" > Loss at Pa: ${formattedLOSSA}</p>
<p style="text-align: left; padding:15px 40px;" > Gain at Pb: ${formattedLOSSB}</p>
</div>
</div>

`;
} else {
// Ошибка сервера
resultDiv.className = 'result error';
resultDiv.innerHTML = `
<h3> ❌ Server error</h3>
<p>Статус: ${response.status}</p>
<p>${responseData.detail || 'Неизвестная ошибка'}</p>
`;
}
document.getElementById("plotImage").src =
"" + responseData.results.image;

} catch (error) {
console.error(' ❌ Error:', error);
resultDiv.className = 'result error';
resultDiv.innerHTML = `
<h3> ❌ Connection error</h3>
<p>${error.message}</p>
<p style="font-size: 14px; color: #666;">Убедитесь, что сервер запущен на http://localhost:8000</p>
`;
responseDataDiv.textContent = error.message;
} finally {
// Разблокируем кнопку
button.disabled = false;
button.textContent = '🚀 Calculate ImL';
}

}
// Добавляем возможность отправки по Enter
document.addEventListener('keypress', function(e) {
if (e.key === 'Enter') {
```

```
calculate();
}
});
</script>

</p>
</div>
</main>

<hr>

<footer>
<a href="https://t.me/LiquidityMaster" target="_blank">Telegram</a>
<a href="https://www.youtube.com/channel/UCrNBI0Dpk90PUmM44JOO9xQ" target="_blank">Youtube</a>
</footer>


</body>
</html>
```

Исходный код фронтенда страницы Sim_Hedge_Short.html

```
<!DOCTYPE html>

<html lang="ru">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="./styles.css">
<title>Liquidity Modeling | DeFi Yield Simulator</title>
<!-- link rel="icon" href="./favicon.png" type="image/png" -->
</head>

<body>

<header>
<div class="header-container">
<img src="./pics/LM_logo.png" alt="Liquidity Master Logo" class="logo">

<nav>
<div class="dropdown">
<a href="#" class="menu-item">Simulate ▼</a>
<div class="submenu">
<a href="./Sim_ImL.html">Impermanent Loss Calc</a>
<a href="./Sim_Hedge_Short.html">Short Hedging</a>
<a href="./Sim_Hedge_Dyn.html">Dinamic Hedging</a>
</div>
</div>

<a href="./New_position.html" class="menu-item">New position</a>
<a href="#" class="menu-item">My positionы</a>
<a href="#" class="menu-item">Docs</a>
</nav>
```

```
</div>

</header>

<main>
<div class="content-placeholder">
<h1>Short Hedging Simulator</h1>
<p style="margin-top:20px;color:var(--muted);">
<!-- -------------- new Input 8 fields ---------------- -->
<div class="input-group">

<div style=" display:flex; align-items: center; justify-content:center; gap: 20px;">
<div class="input-row" >
<div class="field_name">Satrting price P0:</div>
<input class="PInput" type="number" id="p0" step="any" placeholder="Enter P0" >
</div>

<div class="input-row" >
<div class="field_name">Position size X0:</div>
<input class="PInput" type="number" id="x0" step="any" placeholder="Enter X0" >
</div>
</div>


<div style=" display:flex; align-items: center; justify-content:center; gap: 20px;">
<div style=" display:flex; align-items: center; justify-content:center; gap: 10px;">
<div class="input-row">
<div class="field_name">Lower price Pa:</div>
<input class="PInput_range" type="number" id="p1" step="any" placeholder="Enter Pa" >
</div>
<div class="input-row">
<div class="field_name">Pa %:</div>
<div class="Poutput" id="Pa%" placeholder="Pa %" step="any" type="number" >
<p style="text-align: center; color: #807979; " > Pa %</p>
</div>
</div>
</div>

<div style=" display:flex; align-items: center; justify-content:center; gap: 10px;">
<div class="input-row">
<div class="field_name">Upper price Pb:</div>
<input class="PInput_range" type="number" id="p2" step="any" placeholder="Enter Pb" >
</div>
<div class="input-row">
<div class="field_name">Pb %:</div>
<div class="Poutput" id="Pb%" placeholder="Pb %" step="any" type="number" >
<p style="text-align: center; color: #807979; " > Pb %</p>
</div>
</div>
</div>
</div>

<div style=" display:flex; align-items: center; justify-content:center; gap: 20px;">
<div class="input-row">
```

```
<div class="field_name">APR:</div>
<input class="PInput" type="number" id="apr" step="any" placeholder="Enter APR" >
</div>
<div class="input-row">
<div class="field_name">Accepted % loss:</div>
<input class="PInput" type="number" id="loss" step="any" placeholder="% loss" >
</div>
</div>


<script> /*change Pa from abs to % */
const inputFieldP0 = document.getElementById('p0');
const inputFieldPa = document.getElementById('p1');
const outputFieldPa = document.getElementById('Pa%');
const inputFieldPb = document.getElementById('p2');
const outputFieldPb = document.getElementById('Pb%');

inputFieldPa.addEventListener('input', () => {
const val_p0 = parseFloat(inputFieldP0.value) ;
const val_pa = parseFloat(inputFieldPa.value) ;

if (isNaN(val_pa) || isNaN(val_p0) || val_p0 === 0) {
outputFieldPa.textContent = "0.00";
} else {
const percentage = ((val_p0 - val_pa) / val_p0 * 100).toFixed(2);
outputFieldPa.textContent = percentage;
}
});


inputFieldPb.addEventListener('input', () => {
const val_p0 = parseFloat(inputFieldP0.value) ;
const val_pb = parseFloat(inputFieldPb.value) ;

if (isNaN(val_pb) || isNaN(val_p0) || val_p0 === 0) {
outputFieldPb.textContent = "0.00";
} else {
const percentage = ((val_pb - val_p0) / val_p0 * 100).toFixed(2);
outputFieldPb.textContent = percentage;
}
});
</script>


<div style=" display:flex; align-items: center; justify-content:center; gap: 20px;">
<div class="input-row">
<div class="field_name">Hedge leverage (x5 by default):</div>
<input class="PInput" type="number" id="leverage" step="any" value="5" placeholder="Hedge leverage" >
</div>
<div class="input-row">
<div class="field_name">Hedge value amplifier K:</div>
<input class="PInput" type="number" id="hedge_k" step="any" placeholder="Hedge amplifier K" >
</div>
</div>
```

```
</div>



<!---------------------------------- Calc button ------------------->
<div style=" gap: 10px; align-items: center; padding:15px 40px;">
<button class="execute-button" id="calcBtn" onclick="calculate_hedge_short()">🚀 Calculate ImL + Hedge</button>
</div>

<!------------------------------------- "result_window" --------------------->
<div style=" align-items: center; justify-content:center; ">
<div class="input-row">
<div class="field_name">Result:</div>
<div class="result_window" id="result" >
<p style="text-align: center; color: #807979; padding:15px 40px;" > Impermanent Loss + Short</p>
</div>
</div>
<div class="input-row">
<div class="field_name">Plot:</div>
<div style=" display:flex; gap: 15px;">
<div class="plot_window">
<img id="plotImage" style="max-width:100%;" />
</div>
<div class="legend_window">
<p>Legend</p>
<div><img src="./pics/lines/red.png" width="50" height="10" style="margin-right: 10px;">ImL+Hedge</div>
<div><img src="./pics/lines/blue.png" width="50" height="10" style="margin-right: 10px;">Hold</div>
<div><img src="./pics/lines/green.png" width="50" height="10" style="margin-right: 10px;">Base</div>
<div><img src="./pics/lines/orange.png" width="50" height="10" style="margin-right: 10px;">ImL</div>
<div><img src="./pics/lines/purple_dot.png" width="50" height="10" style="margin-right: 10px;">Loss $</div>


</div>
</div>

</div>

<div style="margin-top: 20px;">
<!-- h4 style="margin-bottom: 5px;">📥 Servre response:</h4-->
<div style="display: none;" id="responseData" class="json-view">{}</div>
</div>

<script>
// Основная функция расчета
async function calculate_hedge_short() {
const button = document.getElementById('calcBtn');
const resultDiv = document.getElementById('result');
const sentDataDiv = document.getElementById('sentData');
const responseDataDiv = document.getElementById('responseData');
const plotDiv = document.getElementById('plot');
// Получаем значения из полей
const p0 = parseFloat(document.getElementById('p0').value);
const p1 = parseFloat(document.getElementById('p1').value);
```

```
const p2 = parseFloat(document.getElementById('p2').value);
const x0 = parseFloat(document.getElementById('x0').value);
const apr = parseFloat(document.getElementById('apr').value);
const loss = parseFloat(document.getElementById('loss').value);
const leverage = parseFloat(document.getElementById('leverage').value);
const hedge_k = parseFloat(document.getElementById('hedge_k').value);
// Проверка на валидность
if (isNaN(p0) || isNaN(p1) || isNaN(p2) || isNaN(x0) || isNaN(apr) || isNaN(loss)) {
resultDiv.className = 'result error';
resultDiv.innerHTML = '<h3>❌ Error</h3><p>Fill all fields</p>';
return;
}
// Формируем данные для отправки
const requestData = {
P0: p0,
P1: p1,
P2: p2,
X0: x0,
APR: apr,
Loss: loss,
Leverage: leverage,
Hedge_k: hedge_k
};
// Показываем отправляемые данные
//sentDataDiv.textContent = JSON.stringify(requestData, null, 2);
// Блокируем кнопку
button.disabled = true;
button.textContent = '⏳ Расчет...';
// Показываем статус
resultDiv.className = 'result';
resultDiv.innerHTML = '<h3>⏳ Выполняется расчет...</h3>';
console.log('📤 Отправляем данные:', requestData);
try {
// Отправляем POST запрос
const response = await fetch('http://localhost:8000/calculate_hedge_short', {
method: 'POST',
headers: {
'Content-Type': 'application/json',
},
body: JSON.stringify(requestData)
});
// Получаем ответ
const responseData = await response.json();
console.log('📥 Получен ответ:', responseData);
// Показываем ответ
responseDataDiv.textContent = JSON.stringify(responseData, null, 2);
if (response.ok) {
// Успешный расчет
resultDiv.className = 'result success';
// Форматируем результат
//const result = responseData.result;
const Xa = responseData.results.Xa;
const Yb = responseData.results.Yb;
const Y0 = responseData.results.Y0;
```

```
const V0 = responseData.results.V0;
const LOSSA = responseData.results.LOSSA;
const LOSSB = responseData.results.LOSSB;
const X_delta_abs = responseData.results.X_delta_abs;
const X_limit = responseData.results.X_limit;
const Pa_loss = responseData.results.Pa_loss;
const Pb_loss = responseData.results.Pb_loss;
const Va = responseData.results.Xa * p1;
const Days = responseData.results.days;
const formattedXa = typeof Xa === 'number' ? Xa.toFixed(2) : Xa;
const formattedVa = typeof Va === 'number' ? Va.toFixed(2) : Va;
const formattedYb = typeof Yb === 'number' ? Yb.toFixed(2) : Yb;
const formattedY0 = typeof Y0 === 'number' ? Y0.toFixed(2) : Y0;
const formattedV0 = typeof V0 === 'number' ? V0.toFixed(2) : V0;
const formattedLOSSA = typeof LOSSA === 'number' ? LOSSA.toFixed(2) : LOSSA;
const formattedLOSSB = typeof LOSSB === 'number' ? LOSSB.toFixed(2) : LOSSB;
const formattedX_delta_abs = typeof X_delta_abs === 'number' ? X_delta_abs.toFixed(2) : X_delta_abs;
const formattedX_limit = typeof X_limit === 'number' ? X_limit.toFixed(2) : X_limit;
const formattedPa_loss = typeof Pa_loss === 'number' ? Pa_loss.toFixed(2) : Pa_loss;
const formattedPb_loss = typeof Pb_loss === 'number' ? Pb_loss.toFixed(2) : Pb_loss;
const formattedDays = typeof Days === 'number' ? Days.toFixed(2) : Days;

resultDiv.innerHTML = `
<div class="result_window" id="result">
<div>
<p style="text-align: left; padding:15px 40px;" > # of X at Pa (Xa): ${formattedXa}</p>
<p style="text-align: left; padding:15px 40px;" > Value of X at Pa (Va): ${formattedVa}</p>
<p style="text-align: left; padding:15px 40px;" > # of Y at Pb (Yb): ${formattedYb}</p>
<p style="text-align: left; padding:15px 40px;" > Valueof Y at P0 (Y0): ${formattedY0}</p>
</div>
<div>
<p style="text-align: left; padding:15px 40px;" > Starting value at P0 (V0): ${formattedV0}</p>
<p style="text-align: left; padding:15px 40px;" > Loss at Pa: ${formattedLOSSA}</p>
<p style="text-align: left; padding:15px 40px;" > Gain at Pb: ${formattedLOSSB}</p>
<p style="text-align: left; padding:15px 40px;" > Loss at delta X_delta_abs: ${formattedX_delta_abs}</p>
</div>

<div>
<p style="text-align: left; padding:15px 40px;" > Price X_limit: ${formattedX_limit}</p>
<p style="text-align: left; padding:15px 40px;" > Price Pa_loss: ${formattedPa_loss}</p>
<p style="text-align: left; padding:15px 40px;" > Price Pb_loss: ${formattedPb_loss}</p>
<p style="text-align: left; padding:15px 40px;" > Days to break even: ${formattedDays}</p>
</div>
</div>

`;
} else {
// Ошибка сервера
resultDiv.className = 'result error';
resultDiv.innerHTML = `
<h3> ❌ Serger error</h3>
<p>Статус: ${response.status}</p>
<p>${responseData.detail || 'Неизвестная ошибка'}</p>
`;
```

```
}
document.getElementById("plotImage").src =
"" + responseData.results.image;



} catch (error) {
console.error(' ❌ Error:', error);
resultDiv.className = 'result error';
resultDiv.innerHTML = `
<h3> ❌ Connection error</h3>
<p>${error.message}</p>
<p style="font-size: 14px; color: #666;">Убедитесь, что сервер запущен на http://localhost:8000</p>
`;
responseDataDiv.textContent = error.message;
} finally {
// Разблокируем кнопку
button.disabled = false;
button.textContent = '🚀 Calculate ImL + Hedge';
}

}



// Добавляем возможность отправки по Enter
document.addEventListener('keypress', function(e) {
if (e.key === 'Enter') {
calculate();
}
});
</script>



</p>
</div>
</main>

<hr>

<footer>
<a href="https://t.me/LiquidityMaster" target="_blank">Telegram</a>
<a href="https://www.youtube.com/channel/UCrNBI0Dpk90PUmM44JOO9xQ" target="_blank">Youtube</a>
</footer>


</body>
</html>
```

Исходный код модуля ImL_Calc.py

```
import math

import numpy as np
import matplotlib.pyplot as plt
import io
import base64

def ImpermLossCalc(X0, P0, Pa, Pb, APR, Loss):
L = X0 * math.sqrt(P0 * Pb)/(math.sqrt(Pb) - math.sqrt(P0))
Y0 = L * (math.sqrt(P0) - math.sqrt(Pa))
V0 = X0*P0+Y0 #starting Value in the Pool
Xa = L * (math.sqrt(Pb) - math.sqrt(Pa)) / math.sqrt(Pa*Pb) # of X at Pa
Yb = L * (math.sqrt(Pb) - math.sqrt(Pa)) #value at Pb

Px = np.arange(Pa*1.0, Pb*1.0, 1)
Px_plot = np.arange(Pa*1.0, Pb*1.0, 1)
X_Px = L * (np.sqrt(Pb) - np.sqrt(Px)) / np.sqrt(Px*Pb)
Y_Px = L * (np.sqrt(Px) - np.sqrt(Pa))
ImL_Px = X_Px*Px + Y_Px
V_const=np.ones(len(Px))*V0
V_hold = Px*X0 +Y0

plt.figure()
#plt.title('V_hold and ImL_Px')
#plt.plot(Px, V_const)
plt.plot(Px, V_hold, Px, ImL_Px)
plt.grid(True)
plt.savefig('V_hold.jpg', dpi=300, bbox_inches='tight')
#plt.show()

buf = io.BytesIO()
plt.savefig(buf, format="png")
plt.close()
image_base64 = base64.b64encode(buf.getvalue()).decode("utf-8")

LossA = (Xa*Pa-V0)
LossB = (Yb-V0)
print('------------from def ImLCalc ----------')
print ("Xa = ", Xa)
print ("Yb = ", Yb)
print ("Y0 = ", Y0)
print ("V0 = ", V0)
print ("LossA = ", LossA)
print ("LossB = ", LossB)
print('------------end of def----------')

return Xa, Yb, Y0, V0, LossA, LossB, image_base64



X_delta_percent = 1 #allowed loss (Risk Management)
Loss = 1
APR = 200 #APR in LP
k_short = 1.2 # short coefficicent
```

```
P0 = 1000
Pa = 500
Pb = 1500
X0 = 1 #starting quantity of ETH

Xa, Xb, Y0, V0, LOSSA, LOSSB, image = ImpermLossCalc(X0, P0, Pa, Pb, APR, Loss)

#Short plot
#plt.figure()
#plt.title('V_hold')
#plt.plot(Px, V_hold)
#plt.grid(True)
#plt.show()

#Hold vs Impermanent loss
```

Исходный код модуля ImL_Hedge_Short.py

```
import math

import numpy as np
import matplotlib.pyplot as plt
import io
import base64

def ShortHedgeCalc(X0, P0, Pa, Pb, APR, Loss, Leverage, Hedge_k):
L = X0 * math.sqrt(P0 * Pb)/(math.sqrt(Pb) - math.sqrt(P0))
Y0 = L * (math.sqrt(P0) - math.sqrt(Pa))
V0 = X0*P0+Y0 #starting Value in the Pool
Xa = L * (math.sqrt(Pb) - math.sqrt(Pa)) / math.sqrt(Pa*Pb) # of X at Pa
Yb = L * (math.sqrt(Pb) - math.sqrt(Pa)) #value at Pb

Px = np.arange(Pa*1.0, Pb*1.0, 1)
Px_plot = np.arange(Pa*1.0, Pb*1.0, 1)
X_Px = L * (np.sqrt(Pb) - np.sqrt(Px)) / np.sqrt(Px*Pb)
Y_Px = L * (np.sqrt(Px) - np.sqrt(Pa))
ImL_Px = X_Px*Px + Y_Px
V_const=np.ones(len(Px))*V0
V_hold = Px*X0 +Y0

#-------------------------SHORT----------------------------------------
#Ideal hedging short
V_short = X_Px * (np.ones(len(Px)) * P0 - Px)

# classic short
V_long = Px*X0 - P0*X0
V_short_classic = Hedge_k*X0 * (np.ones(len(Px)) * P0 - Px)
V_long_classic = Hedge_k*X0 * (np.ones(len(Px)) * P0 + Px)
PnL_ImL_Short = np.add(ImL_Px, V_short_classic) #PnL of ImL_Px + hedging Short
```

```
#calculate new range boundaries for X_delta_percent
X_delta_abs = (Loss / 100) * V0 #allowed Value loss
X_limit = V0 - X_delta_abs #Value at percent prices
V_limit = V_const - X_delta_abs


ImL_Short_diff = []
ImL_Short_diff=np.ones(len(Px))
pos_index = []

for i in range(len(X_Px)):
ImL_Short_diff[i] = PnL_ImL_Short[i] - V_limit[i] #where PnL_ImL_Short is within % range
if ImL_Short_diff[i] > 0:
pos_index.append(i)

i_limit_a = pos_index[0]
i_limit_b = pos_index[-1]

print('pos_indexA =', pos_index[0])
print('pos_indexB =', pos_index[-1])

print('----- ImL_Hedge_Short')
Pa_loss = Px[i_limit_a] # low price of accepted loss
Pb_loss = Px[i_limit_b] # high price of accepted loss
print ('Value loss at allowed percent X_delta_abs = ', X_delta_abs)
print('Value at allowed percent X_limit = ', X_limit)
print ('Price Pa_loss', Pa_loss)
print ('Price Pb_loss', Pb_loss)
print('----- X_delta_abs, X_limit, Pa_loss, Pb_loss ')

print('PnL_ImL_Short_A = ',PnL_ImL_Short[i_limit_a] )
print('PnL_ImL_Short_B = ',PnL_ImL_Short[i_limit_b] )

PnL_ImL_Short_A = float(PnL_ImL_Short[i_limit_a])

#calculate # of days for selected %
days = (V0 - (PnL_ImL_Short_A) ) /((V0/100)* APR / 365)

print ('days to break even:', days)




#------------- plot generation -------------------------------------------
plt.figure()
plt.title('V_hold ImL+Short_Hedge')
#plt.plot(Px, V_hold, Px, ImL_Px, Px, V_const, Px, PnL_ImL_Short)
plt.plot(Px, PnL_ImL_Short, Px, V_const,)


#plt.vlines(x=[Px[i_limit_a], Px[i_limit_b]], ymin=0, ymax=1,
# transform=plt.gca().get_xaxis_transform(),
# color='magenta', linestyle=':', linewidth=1.5, alpha=1)
```

```
plt.grid(True)
plt.savefig('V_Iml_Short.jpg', dpi=300, bbox_inches='tight')
#plt.show()
buf = io.BytesIO()
plt.savefig(buf, format="png")
plt.close()
image = base64.b64encode(buf.getvalue()).decode("utf-8")
#end of image


LossA = (Xa*Pa-V0)
LossB = (Yb-V0)
print ('Def Hedge Calculation Data ')
print ("Xa = ", Xa)
print ("Yb = ", Yb)
print ("Y0 = ", Y0)
print ("V0 = ", V0)
print ("LossA = ", LossA)
print ("LossB = ", LossB)
print ("days = ", days)

return Xa, Yb, Y0, V0, LossA, LossB, image, X_delta_abs, X_limit, Pa_loss, Pb_loss, days

#X_delta_percent = 1 #allowed loss (Risk Management)
#Loss = 1
#APR = 200 #APR in LP
#k_short = 1.2 # short coefficicent
#P0 = 1000
#Pa = 500
#Pb = 1500
#X0 = 1 #starting quantity of ETH
#Leverage = 5
#Hedge_k = 1

#Xa, Xb, Y0, V0, LOSSA, LOSSB, image = ShortHedgeCalc(X0, P0, Pa, Pb, APR, Loss, Leverage, Hedge_k)

#Short plot
#plt.figure()
#plt.title('V_hold')
#plt.plot(Px, V_hold)
#plt.grid(True)
#plt.show()

#Hold vs Impermanent loss
```

Исходный код модуля ImL_Calc.py

```
import math

import numpy as np
import matplotlib.pyplot as plt
import io
```

```
import base64

def ImpermLossCalc(X0, P0, Pa, Pb, APR, Loss):
L = X0 * math.sqrt(P0 * Pb)/(math.sqrt(Pb) - math.sqrt(P0))
Y0 = L * (math.sqrt(P0) - math.sqrt(Pa))
V0 = X0*P0+Y0 #starting Value in the Pool
Xa = L * (math.sqrt(Pb) - math.sqrt(Pa)) / math.sqrt(Pa*Pb) # of X at Pa
Yb = L * (math.sqrt(Pb) - math.sqrt(Pa)) #value at Pb

Px = np.arange(Pa*1.0, Pb*1.0, 1)
Px_plot = np.arange(Pa*1.0, Pb*1.0, 1)
X_Px = L * (np.sqrt(Pb) - np.sqrt(Px)) / np.sqrt(Px*Pb)
Y_Px = L * (np.sqrt(Px) - np.sqrt(Pa))
ImL_Px = X_Px*Px + Y_Px
V_const=np.ones(len(Px))*V0
V_hold = Px*X0 +Y0

plt.figure()
#plt.title('V_hold and ImL_Px')
#plt.plot(Px, V_const)
plt.plot(Px, V_hold, Px, ImL_Px)
plt.grid(True)
plt.savefig('V_hold.jpg', dpi=300, bbox_inches='tight')
#plt.show()

buf = io.BytesIO()
plt.savefig(buf, format="png")
plt.close()
image_base64 = base64.b64encode(buf.getvalue()).decode("utf-8")

LossA = (Xa*Pa-V0)
LossB = (Yb-V0)
print('------------from def ImLCalc ----------')
print ("Xa = ", Xa)
print ("Yb = ", Yb)
print ("Y0 = ", Y0)
print ("V0 = ", V0)
print ("LossA = ", LossA)
print ("LossB = ", LossB)
print('------------end of def----------')

return Xa, Yb, Y0, V0, LossA, LossB, image_base64



X_delta_percent = 1 #allowed loss (Risk Management)
Loss = 1
APR = 200 #APR in LP
k_short = 1.2 # short coefficicent
P0 = 1000
Pa = 500
Pb = 1500
X0 = 1 #starting quantity of ETH
```

```
Xa, Xb, Y0, V0, LOSSA, LOSSB, image = ImpermLossCalc(X0, P0, Pa, Pb, APR, Loss)

#Short plot
#plt.figure()
#plt.title('V_hold')
#plt.plot(Px, V_hold)
#plt.grid(True)
#plt.show()

#Hold vs Impermanent loss
```

Исходный код start.bh

```
while true; do
echo "Start the main program..."
# python3 /home/ubuntu/HedgeBotCC/Code/resume_test.py
/usr/local/bin/python3 /Users/meister/Hyperliquid_bot/CCXT_bot/2_Cryptos_prod/Hedge_HL_2CC_try_diff_1.py
# Если программа завершилась (сбой или нормально)
EXIT_CODE=$?
echo "Program finished with code: $EXIT_CODE"
# don't restart if everything is OK (код 0)
if [ $EXIT_CODE -eq 0 ]; then
echo "Program finished OK. Exit."
break
fi
echo "Restart in 5 sec..."
sleep 5
done
```

ОЦЕНКА КАЧЕСТВА ВОЗДУХА С ПРИМЕНЕНИЕМ ДРОНА

Кузьмина Светлана Васильевна

Якутск г, МАНОУ ДДТ им. Ф.И. Авдеевой, 10 класс

Секция: Многоцелевые роботы и летательные аппараты

Научный руководитель: Данилов Денис Алексеевич, педагог дополнительного образования МАНОУ ДДТ им. Ф.И. Авдеевой

АННОТАЦИЯ

Осведомление о степени загрязненности воздуха предотвратит пагубное воздействие частиц на здоровье горожан. В проекте рассматривается способ экологической оценки качества воздуха с применением беспилотного летательного аппарата (далее – дрон), оснащенного датчиками для измерения концентрации загрязнителей. Традиционные методы мониторинга имеют следующие ограничения: зачастую они расположены в ограниченных географических зонах и не могут отражать полную картину загрязнения. Дроны позволяют перемещаться на большие расстояния и собирать больше данных для анализа качества воздуха, требуя при этом минимальные ресурсы. В ходе проекта проведено исследование контролируемых величин и их влияния на организм, проведен обзор и анализ готовых аналогов, собран прототип, в который внедрены датчики для определения следующих параметров: диоксид углерода, угарный газ, твердые частицы, влажность, температура, изучено изменение качества воздуха с высотой, сделаны тестирование данного способа измерения качества воздуха, тестирование устройства при низких температурах. Проект также включает в себя информационные кампании для повышения осведомленности населения о проблемах экологии и их влиянии на здоровье.

Ключевые слова: качество воздуха, загрязняющие вещества, дрон, датчики, измерительная станция, стратификация

ВВЕДЕНИЕ

Здоровье горожан напрямую зависит от качества воздуха, данный фактор влияет на уровень жизни. Применяя меры по защите организма от загрязнения воздуха, можно снизить риск возникновения таких заболеваний, как инсульт, сердечно-сосудистые болезни, рак легких и других серьезных недугов [1]. Однако существующая сеть станций мониторинга в нашем регионе слишком разрежена и не отражает полную картину загрязнения. При этом оснастить территорию достаточным количеством таких станций дорого и ресурсоёмко. Использование дронов предполагает экономичный и мобильный подход: вместо множества статичных датчиков мы создаём перемещаемую измерительную станцию на базе дрона. Это позволяет максимально эффективно использовать имеющиеся у нас ограниченные ресурсы, превращая один комплект газоанализаторов в инструмент для оперативного обследования всей площади. Существующие исследования показывают, что использование дронов для мониторинга качества воздуха начало развиваться во всем мире. Однако многие вопросы в исследуемой области остаются нерешенными: недостаточно изучен данный способ оценки качества воздуха, влияние различных факторов на точность измерений.

Цель*:* разработать способ экологической оценки качества воздуха с применением дрона.

Задачи:

- Исследовать контролируемые величины и их влияние на человека
- Найти готовые аналоги
- Программировать датчики
- Сделать конструкцию для установки датчиков на дрон
- Программировать дрон
- Провести испытания на работоспособность
- Тестировать данный способ измерения качества воздуха
- Тестировать устройства при низких температурах
- Создать сайт
- Создать справочник-путеводитель, постер

Гипотеза: Созданный нами дрон-эколог сможет заменить работу целой станции, автоматизирует данный процесс и за короткий промежуток времени можно получить необходимую информацию об экологическом состоянии воздуха.

Практическая ценность: Получаемые данные можно использовать с целью создания информационной основы принятия решений в области охраны природы, контроля предприятий природопользования и экологической безопасности Использование дронов позволяет

перемещаться на большие расстояния по территории города, используя минимальные ресурсы, мобильность измерительной станции позволяет получать больше данных для анализа.

Методы исследования: анализ и синтез, наблюдение, сравнение, измерение, эксперимент, тестирование.

1. ИССЛЕДОВАНИЕ КОНТРОЛИРУЕМЫХ ВЕЛИЧИН И ИХ ВЛИЯНИЯ НА ОРГАНИЗМ ЧЕЛОВЕКА

Диоксид углерода (IV)- кратковременное вдыхание здоровыми людьми двуокиси углерода в этих концентрациях вызывает отчетливые сдвиги в функции внешнего дыхания, кровообращении и значительные ухудшения электрической активности головного мозга [2].

Оксиды азота - взаимодействуя с парами воды в воздухе, разрушают легочную ткань, вызывая хронические заболевания, отек легких. Наибольшую опасность оксиды азота представляют в качестве активного компонента смога, приводя к раздражению дыхательной системы, осложнению заболеваний, затруднению дыхания, кашлю и удушью, тошноте, боли в животе. Контакт кожи и глаз вызывает раздражение и ожоги [3].

Твердые частицы представляют собой жидкие и твердые частиц, взвешенные в воздухе. Эти вещества могут проникать глубоко в легкие и затем в кровоток. Это может привести к астме, бронхиту, раку легких и крови. Они могут варьироваться от микроскопических частиц до дыма, сажи, жидких частиц и пыли [3, 4].

Угарный газ может повлиять на память, работоспособность мозга, поведение и сознание. Высокая концентрация газа незамедлительно приводит к смерти. При небольшом количестве газа может появиться пульсация в висках, слабость, головная боль, нарушение равновесия, звон в ушах, мерцание перед глазами, слабость в ногах, тошнота, рвота, слезы, позывы к кашлю [4, 5].

2. АНАЛИЗ ИТОГОВ ПЕРВЫХ СПРИНТОВ

Из найденных датчиков я отобрала подходящие и перешла к этапу тестирования и программирования следующих датчиков: паров спирта, угарного газа, метана, водорода, температуры. Изначально работа велась с дроном Призрак-2 (Рис.1). С данным прототипом я участвовала в Лаврентьевских чтениях и поняла, что данные компоненты не подходят для реализации проекта: при получении данных с датчиков не выстраивается «полная картина» экологической ситуации. По итогам первого спринта была выявлена проблема: недостаточное количество деталей для реализации проекта.

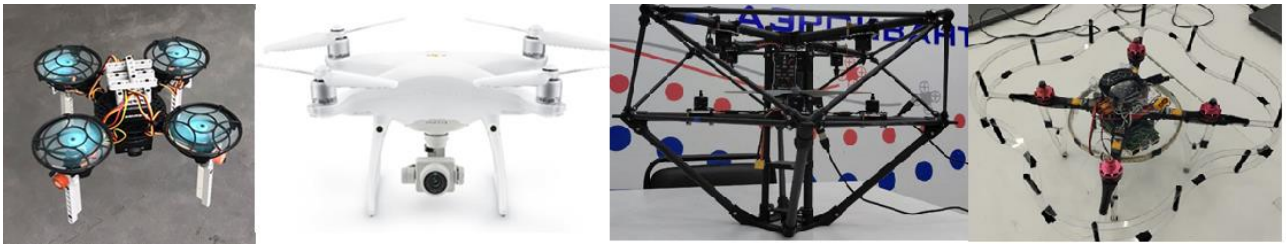


Рис. 1. Дроны: Призрак-2, Фантом-4, октокоптер, Клевер-4

Для решения данной проблемы я купила недостающие детали из интернет-магазинов, составили смету (Таблица 1). В качестве замены Призрака-2 ввиду недостаточной грузоподъемности рассматривались дроны Фантом-4, октокоптер и Клевер-4 (Рис.1). но из-за ошибок подключения с первыми двумя моделями начали работать, и я приступила к Клеверу-4.

Таблица 1. Смета первого закупа

Наименование	Кол-во, шт.	Цена, руб	Сумма, руб	Ссылка
1. Датчик углекислого, качества воздуха, температуры и влажности	1	429	428	https://aliexpress.ru
2. Датчик пыли	1	1790	1790	https://amperka.ru
3. Датчик обнаружения угарного газа Arduino	2	88	176	https://aliexpress.ru
4. Troyka Shield	1	740	740	https://amperka.ru
5. Набор модулей беспроводной платы SIM900	1	683	683	https://aliexpress.ru
6. Плата	1	246	246	https://aliexpress.ru
7. Доставка	1	342	342	https://aliexpress.ru
Сумма, руб	4405			

Так как масса дрона Клевер-4 с собранным полезным грузом была близка к предельной взлетной массе мы рассмотрели более мощные модели и выбор пал на Phantom 4 Pro.

3.1 ПРОГРАММИРОВАНИЕ ДАТЧИКОВ

Для программирования датчиков был освоен инструментарий среды разработки Arduino IDE. В рамках подготовки был проведён тест отдельных датчиков и модулей. В ходе тестирования были выявлены следующие проблемы: отсутствие устойчивого сигнала в GSM-модуле, а также утеря датчика температуры и влажности DHT22. На основании этого были

дополнительно приобретены необходимые компоненты, составлена смета (Таблица 2). Перед началом измерений все датчики прошли процедуру прогрева ("прожиг") в течение 48 часов для стабилизации выходных сигналов. Газовые анализаторы были дополнительно откалиброваны с помощью специального кода в условиях проветриваемого помещения. Программная часть реализовывалась поэтапно. Сначала датчики были последовательно соединены в единый код. На заключительном этапе к объединённой программе сбора данных был подключен GSM-модуль для передачи данных (Рис. 2)

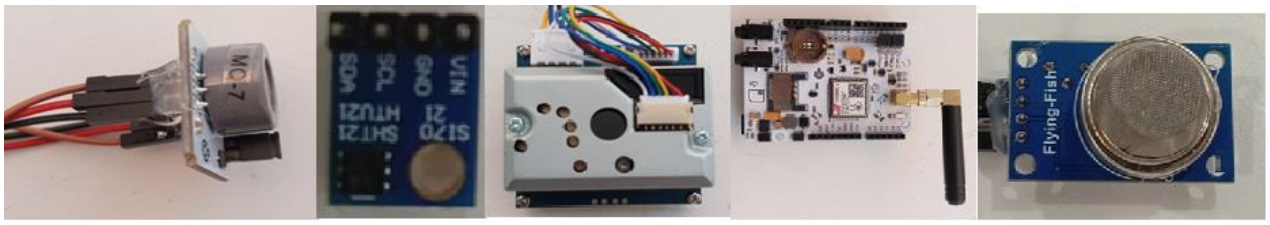


Рис. 2. Компоненты: MQ-7, HTU21, Dust sensor. GSM-модуль, MQ-135.

3.2 ПРОГРАММИРОВАНИЕ ДРОНА

Изначально у нас возникли трудности с подключением пульта к дрону Phantom 4, однако после того как мы нашли ему замену, работа наконец-то началась. Далее мы последовательно выполнили все необходимые этапы. Сначала была проведена настройка через приложение DJI GO. Затем мы перешли к подготовке самого дрона: установили пропеллеры, вставили и активировали аккумулятор. После этого подключили и включили новый пульт управления. Следующим ключевым шагом стало сопряжение пульта с дроном для обеспечения устойчивой связи. Далее мы приступили к важной процедуре калибровки датчиков, в рамках которой откалибровали компас и инерциальный модуль (IMU). Финальным этапом стала детальная настройка параметров полёта: мы установили ограничения по высоте и дальности, выбрали подходящий режим полёта и настроили параметры камеры для съёмки.

4. КОНСТРУКЦИЯ УСТАНОВКИ ДАТЧИКОВ НА ДРОН

1. Изготовление конструкции №1

Для полета важно было обеспечить стабильность и устойчивость при взлете и посадке. Первым этапом проектирования был чертеж модели, затем макет из картона, ввели корректировки в чертеж и отправились в HI-TECH цех для реализации проекта на лазерном гравере. Для скрепления частей между собой был использован метод шип-паз и специальный клей для столярных работ (Рис 3-5).

Рис. 3. Процесс скрепления частей

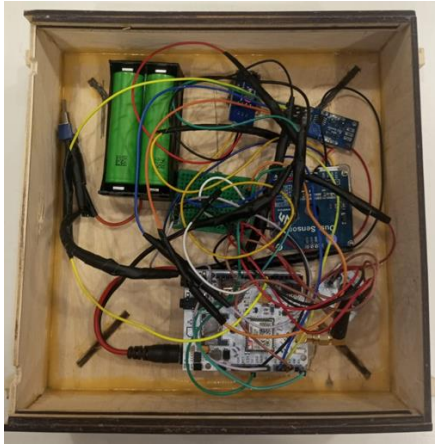


Рис. 4. Датчики внутри

Рис. 5. Конструкция №1

2. Тестовые взлеты с конструкцией №1

Тестовые взлеты состояли из удержания заданной позиции, движение по прямой, наклон, повороты, подъем/спуск. При взлете с закрепленными датчиками дрон колеблется из стороны в сторону. Конструкция имеет ряд существенных недостатков. Смещённый центр тяжести, и нехватка мощности двигателей вызывают вибрации. Габариты создают турбулентность, ухудшая управление. Датчики уязвимы к загрязнению и ударам.

Установленная коробка увеличивает тормозной путь, ограничивает обзор камеры и блокирует датчик позиционирования, что снижает стабильность на малой высоте. Исходя из результатов тестовых взлетов, было принято решение пересмотреть этапы изготовления и смоделировать другую конструкцию.

3. Этапы создания конструкции №2

Учитывая ошибки, допущенные при проектировании конструкции №1, был сделан эскиз, где датчики расположены сверху, в центре исходя из анализа недостатков предыдущей конструкции. Сначала был сделан макет из картона, были сделаны тестовые взлеты проверка стабильности взлетами с зависанием, поворотами, различными траекториями. С данной конструкцией дрон сохраняет стабильность. Изготовление корпуса: вырезали на лазерном станке ЧПУ акриловый пластик (ППМА) (Рис.6), а также распечатали на 3D принтере корпус для аккумулятора (Рис. 7). Рассмотрели способы закрепления грузов на дронах: встроенный отсек и защелка, стяжки и резинки, Velcro-лента. Была выбрана Velcro-лента, ввиду возможности моментальной установки и демонтажа батареи (Рис 8).

Рис. 6. Корпус из акрилового пластика

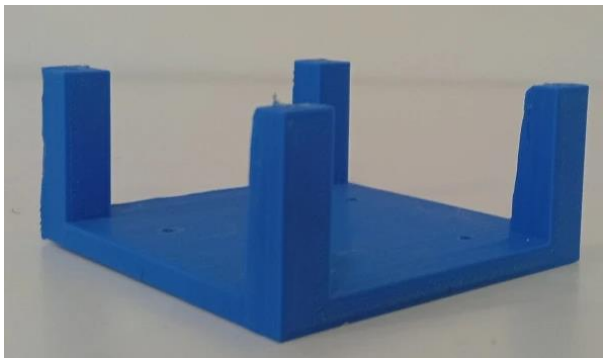


Рис. 7. Корпус для аккумулятора

Рис. 8. Закрепленная

Проведение измерений при опытах

а) Перед началом измерений датчики MQ-7 и MQ-135 были прогреты после включения в течение 2 минут в соответствии с инструкцией по эксплуатации, чтобы получать стабильные показания.

б) Измерения каждой величины проводились не менее 5 раз для последующего статистического анализа. Первичные данные вносились непосредственно в таблицы измерений.

в) Каждый результат измерения связан с погрешностью, которая выражается в тех же единицах, что и сама величина. Для нахождения погрешности использовались следующие формулы:

(1)$x_{\text{ср}} = \frac{1}{n}\sum_{i=1}^{n} x_i$ - среднее значение пяти измерений - принимается как показание качества воздуха в рассматриваемой точке

(2)$\Delta x_i = x_i - x_{\text{ср}}$ - отклонение каждого из пяти измерений

(3)$\Delta x_{\text{ср}} = \frac{1}{n}\sum_{i=1}^{n} \Delta x_{\text{ср}}$ - погрешность измерения

5. ИЗУЧЕНИЕ ДАННОГО СПОСОБА ИЗМЕРЕНИЯ

Воздушные потоки, создаваемые при полете дрона, могут изменить показание концентрации в воздухе вследствие двух процессов. Воздушный поток от дрона разбавляет пробы воздуха вблизи датчика, приводя к занижению показаний концентрации загрязняющих веществ. Было достигнуто максимальное расстояние между датчиком и дроном в ходе проектирования конструкции №2, при котором это влияние становится незначимым. Воздушные потоки перемешивают воздух. Ключевая проблема измерений качества воздуха с помощью дрона заключается в том, что естественное распределение загрязняющих веществ в атмосфере неоднородно по высоте (стратификация).

Стратификация обусловлена следующими факторами:

– Локализацией источников загрязнения: высота загрязнителя определяет, в каком слое атмосферы первоначально попадают загрязняющие вещества. Большинство

основных источников загрязнения находятся на уровне земли. Это приводит к максимальной концентрации загрязняющих веществ в нижнем слое [6, 7].

– Метеорологические параметры: температурная инверсия — это повышение температуры воздуха с высотой, обратное нормальному для тропосферы распределению. Приземные инверсии подавляют вертикальные движения 3оздуха, препятствуя рассеиванию водяного пара, пыли и загрязняющих веществ [8, 9]. Для высоких источников выброса (промышленные трубы) характерна парадоксальная зависимость: при высокой скорости ветра у земли наблюдается большая концентрация примеси. Горячая газовоздушная смесь из трубы получает дополнительную плавучесть, что создает поле вертикальных скоростей и эффективно уносит факел вверх. При слабом ветре подъем факела усиливается, и концентрация у земли снижается [10]. Атмосферное давление связано с количеством молекул воздуха в определённом объёме, что влияет на его плотность и вес [11].

– Физико-химические свойства веществ: плотность и молекулярная масса веществ влияют на их распределение по высоте. В более высоких слоях атмосферы концентрация более тяжёлых газов убывает быстрее по мере удаления от поверхности Земли [11]. Например, тяжёлые газы (SO_2, Cl_2) склонны скапливаться в нижних слоях, а лёгкие (CH_4, H_2) — рассеиваться вверх. Температура кипения и летучесть вещества тесно связаны и существенно влияют на скорость его рассеивания в окружающей среде. Эти параметры определяют, насколько быстро вещество переходит из жидкого или твёрдого состояния в газообразное, что, в свою очередь, влияет на его распространение в атмосфере, стойкость и потенциальное воздействие на окружающую среду [8].

Воздух с разных слоев поднимается наверх, перемешивается с тем, что был выше, при низкой высоте поднимаются частицы с земли. Таким образом, датчики берут показания из зоны интенсивного перемешивания, получается усредненная концентрация.

Опыт №1: «Изучение стратификации воздуха»

Цель: измерить изменение концентрации загрязняющих веществ при изменении высоты.

В каждой точке бралось 5 показаний и записывался результат вычислений по формулам 1,2,3 в таблицу №3 Далее вычислили разницу между показаниями для наглядности (Таблица 4).

Таблица 3. Результаты измерения качества воздуха на разных высотах

	Высота, м	CO, ppm	Твердые частицы, ug/m3	Влажность, %	Температура, C
L_1	0.53±	0.00±0.00	0,583±0.02	24,20±0.5	25,30±0.5

L_2	0.75±	0.00±0.00	0,58±0.02	23,75±0.5	25.05 ±0.5
L_3	1.0±	0.00±0.00	0,561±0.017	23,19±0.26	24,58±0.11
L_4	1.25	0.00±0.00	0,60±0.01	22.87±0.24	24,23±0.04
L_5	1.5	0.00±0.00	0,573±0.014	22,88±0.08	24,10±0.00
L_6	1.75	0.00±0.00	0,590±0.015	22,73±0.23	24,075±0.075
L_7	2.0	0.00±0.00	0,587±0.014	22.38±0.12	24,07±0.04
L_8	2.25	0.00±0.00	0,585 ±0.010	22,02±0.18	24,13±0.04
L_9	2.5	0.00±0.00	0,582±0.012	22,18±0.15	24,32±0.06
L_{10}	3,0	0.00±0.00	0,568±0.015	22.98±0.12	24,58±0.03
L_{11}	3,5	0.00±0.00	0,565±0.012	23,82 ±0.12	24,80±0.03

Таблица 4. Разница в показаниях, сделанных на разных высотах

Разница в показаниях	CO, ppm	Твердые частицы, $\frac{\mu g}{m^2}$	Влажность, %	Температура, °C
ΔL_1	0.00	-0.003	-0.45	-0.25
ΔL_2	0.00	-0.019	-0.56	-0.47
ΔL_3	0.00	0.039	-0.32	-0.35
ΔL_4	0.00	-0.027	0.01	-0.13
ΔL_5	0.00	0.017	-0.15	-0.025
ΔL_6	0.00	-0.003	-0.35	-0.005
ΔL_7	0.00	-0.002	-0.36	0.06
ΔL_8	0.00	-0.003	0.16	0.19
ΔL_9	0.00	-0.014	0.80	0.26
ΔL_{10}	0.00	-0.003	0.84	0.22

Вывод: наблюдается температурная стратификация с инверсионным слоем выше 2м. Из-за температурной инверсии подавляется вертикальное перемешивание. Слой теплого воздуха действует как крышка, не позволяя выбросам от земли рассеиваться вверх. При наличии источников загрязнения это привело бы к накоплению вредных веществ в приземном слое.

Опыт №2 «Определение изменения показаний концентрации загрязняющих веществ при циркуляции воздуха потоками, создаваемыми дроном»

Цель: оценить изменения концентрации загрязняющих веществ при измерениях качества воздух в статистическом положении и в динамическом положении, определить точку минимальной разницы.

Сначала мы измеряли качество воздуха на определенной высоте, затем в течение минуты запустили дрон и произвели измерение в состоянии полета. Показания отражены в таблице №5 и №6.

Таблица 5. Результаты измерений в неподвижном состоянии.

Статика					
Высота	CO, ppm	CO_2, ppm	Сырое значение CO2	Твердые частицы, $\frac{\mu g}{m^3}$	Температура, °C
0,5	0,00±0,00	400,00±0,00	232/1023±2	1.472±0,006	27.23±0,05

1,0	0,80±0,4	400,00±0,00	240/1023±7	1,622±0,018	24,17±0,4
1,5	0,20±0,3	400,00±0,00	231,0/1023±0,4	1,628±0,007	25,89±0,04
2,0	0,00±0,00	400,00±0,00	238/1023±1	1,428±0,006	25,1 ±0,3
2,5	1,00±0,00	400,00±0,00	262/1023±3	1,418 ±0.014	23,47±0,08
3,0	1,00±0,00	400,00±0,00	237/1023±1	1,65±0,01	27,9±0,2
3,5	1.00±0,00	400,00±0,00	235/1023±2	1,448±0,006	26,1±0,1

Таблица 6. Результаты измерений в подвижном состоянии.

Динамика					
Высота	CO, ppm	CO_2, ppm	Сырое значение CO2	Твердые частицы, $\frac{\mu g}{m^3}$	Температура, ℃
0,5	0,00±0,00	400,00±0,00	240/1023±10	1,41±0,01	26,1±0,8
1,0	0,2 ±0,16	400,00±0,00	241,6/1023±5,2	1,636±0,011	24,1±0,2
1,5	1,00±0,00	400,00±0,00	244,0/1023±5,2	1,606±0,017	24,13±0,13
2,0	0,00±0,00	400,00±0,00	241/1023±8	1,400±0,008	23,5±0,3
2,5	0,2±0,3	400,00±0,00	245,6/1023±1,3	1,44±0,01	24,8±0,3
3,0	1,00±0,00	400,00±0,00	245/1023±7	1,608 ±0,014	23,4±0,3
3,5	0,8±0,3	400,00±0,00	243/1023±4	1,47±0,04	24,2±0,4

Для определения высоты полета дрона, на которой влияние аэродинамических факторов минимально, найдено значение μ - величина, характеризующая относительное изменение показаний датчиков при измерениях в статике и динамике.

$$\mu = = \frac{|m_{\text{стат}} - m_{\text{дин}}|}{m_{\text{стат}}} \cdot 100\% ,$$

где $m_{\text{стат}}$ – показания в статике, $m_{\text{дин}}$ – показания в динамике (полете)

Были найдены значения μ на каждой высоте (Таблица 7).

Таблица 7. Относительное изменение показаний датчиков

	CO_2- углекислый газ			CO - угарный газ			Твердые частицы			Температура		
Высота	$m_{\text{стат}}$	Δm	μ, %	$m_{\text{стат}}$	Δm	μ, %	$m_{\text{стат}}$	Δm	μ, %	$m_{\text{стат}}$	Δm	μ, %

0,5	232	8	3,45	0,00	0	0	1.472	0,062	4,21	27.23	1,13	4,15
1,0	240	1,6	0,7	0,8	0,6	75	1,622	0,014	0,86	24,17	0,07	0,29
1,5	231	13	5,63	1	0,8	80	1,628	0,022	1,35	25,89	1,76	6,79
2,0	238	3	1,26	0,00	0	0	1,428	0,028	1,96	25,1	1,6	6,37
2,5	262	16,4	6,26	1,00	0,8	80	1,418	0,022	1,55	23,47	1,33	5,66
3,0	237	8	3,38	1,00	0	0	1,65	0,042	2,55	27,9	4,5	16,1
3,5	235	8	3,40	1,00	0,2	20	1,448	0,022	1,52	26,1	1,9	7,28

Для выбора оптимальной высоты применён комплексный подход, включающий следующие показатели:

1. Среднее значение - общий уровень влияния

2. Медиана - типичное влияние

3. Стандартное отклонение - однородность работы датчиков, разброс между датчиками

1. Среднее значение

Чтобы найти минимум общего уровня влияния на датчики, надо найти средние значения по следующей формуле:

$$\mu_{ср} = \frac{\sum \mu_i}{n},$$

где μ_i – значение каждого относительного изменения на высоте, n – количество значений.

Общее влияние минимально при высоте, равной 2 метра (Рис.9, таблица 8).

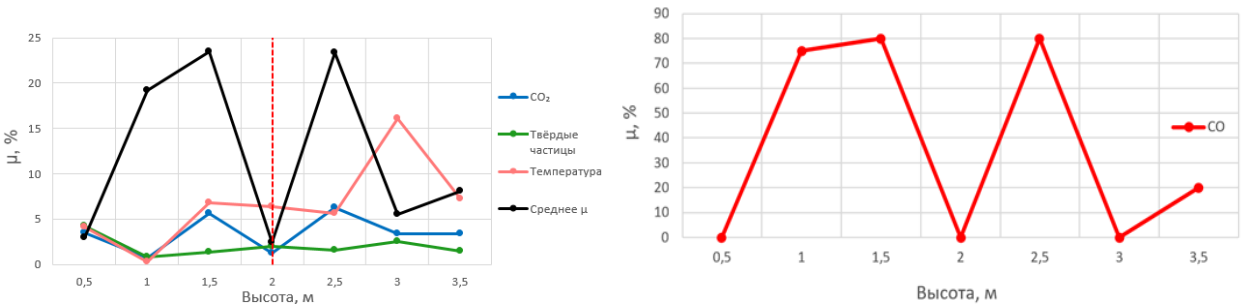


Рис. 9. График относительного изменения показаний датчиков по высоте

Таблица 8. Относительное изменение показаний датчиков по высоте

Высота	μ_{CO_2}, %	μ_{CO}, %	$\mu_{\text{тв.част.}}$, %	μ_t, %	$\mu_{\text{ср}}$, %
0,5	3,45	0	4,21	4,15	2,95
1,0	0,7	75	0,86	0,29	19,21
1,5	5,63	80	1,35	6,79	23,44
2,0	1,26	0	1,96	6,37	2,40
2,5	6,26	80	1,55	5,66	23,37
3,0	3,38	0	2,55	16,1	5,51
3,5	3,40	20	1,52	7,28	8,05

2. Медиана

Медиана рассчитывалась по высотам - для оценки типичного поведения датчиков на каждой высоте.

Были получены следующие результаты (Рис.10, таблица 9):

а) Высота 1,0 м имеет формально низкую медиану (0,78%), но относительное изменение показания CO высокое μ_{CO}=75%, высота 3,0 м имеет среднюю медиану (2,97%), но относительное изменение показания температуры высокое μ_t=16,1%)

б) Высота 2,0 м имеет низкую медиану (1,61%), значит типичный датчик работает отлично.

в) Высоты 1,5 м и 2,5 м имеют высокие медианы, не подходят для измерений.

г) Умеренные значения при высоте 0,5 м и 3,5м.

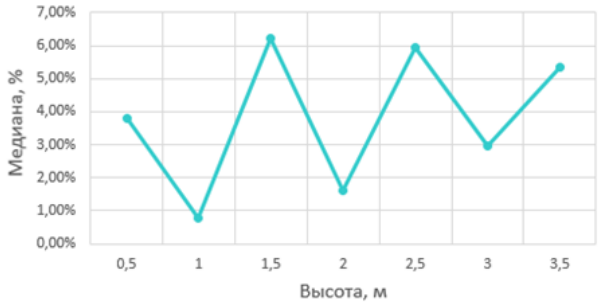


Рис. 10. Медианы значений μ

Таблица 9. Медианы значений μ при разных высотах

Высота, м	Данные μ (отсортировано)	Центральные значения	Медиана	Интерпретация
0,5	[0; 3,45; 4,15; 4,21]	3,45 и 4,15	3,80%	Умеренное влияние
1,0	[0,29; 0,7; 0,86; 75]	0,7 и 0,86	0,78%	Отличное (но μ_{CO}=75%)
1,5	[1,35; 5,63; 6,79; 80]	5,63 и 6,79	6,21%	Высокое влияние
2,0	[0; 1,26; 1,96; 6,37]	1,26 и 1,96	1,61%	Отличное
2,5	[1,55; 5,66; 6,26; 80]	5,66 и 6,26	5,96%	Высокое влияние
3,0	[0; 2,55; 3,38; 16,1]	2,55 и 3,38	2,97%	Хорошее (но μ_t=16,1%)
3,5	[1,52; 3,40; 7,28; 20]	3,40 и 7,28	5,34%	Среднее влияние

3. Стандартное отклонение

Стандартное отклонение показывает, насколько сильно данные разбросаны вокруг среднего, оценивает однородность системы - все ли датчики одинаково реагируют на дрон. Нам нужно, чтобы все датчики (CO_2, CO, твёрдые частицы, температура) показывали примерно одинаковые значения μ на одной высоте. Это значит, что стандартное отклонение (σ) должно быть минимальным.

Для каждой высоты стандартное отклонение (σ) рассчитано по формуле по четырем параметрам: CO_2, CO, твёрдые частицы и температура, относительно среднего значения $\mu_{ср}$ (Таблица 10)

$$\sigma = \sqrt{\frac{\sum(\mu_i - \mu_{cp})^2}{n}},$$

где μ_i - значения параметров, $\mu_{ср}$ - среднее значение, N - количество параметров.

Таблица 10. Стандартные отклонения при разных высотах

Высота	Стандартное отклонение (σ, %)	Оценка однородности
0,5	1,73	Небольшие различия между датчиками
1,0	32,21	Сильный разброс
1,5	38,11	Сильный разброс

2,0	2,84	Небольшие различия между датчиками
2,5	37,74	Сильный разброс
3,0	7,39	Заметный разброс в реакциях
3,5	7,45	Заметный разброс в реакциях

Оптимальные высоты для измерений: 0,5 м, 2,0 м (Рис.11). При измерениях на данных высотах различие между относительными изменениями показаний μ минимально.

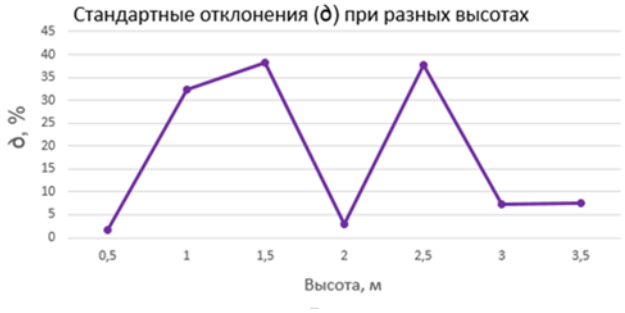


Рис. 11. Стандартные отклонения

Вывод: на основании комплексного анализа высота 2,0 м является оптимальной (Рис. 12), так как общее влияние минимально (среднее μ = 2,40%), медиана низкая (1,61%), что указывает на стабильность измерений, стандартное отклонение невелико (2,84%), что подтверждает согласованность работы датчиков. Высота 0,5 м также показывает хорошие результаты (низкое σ, умеренное влияние), однако на такой низкой высоте возможны дополнительные помехи от поверхности земли. Высоты 1,0 м, 1,5 м, 2,5 м и 3,0 м имеют существенные недостатки: высокий разброс данных, аномальные выбросы по отдельным датчикам, что делает их непригодными для точных измерений (Таблица 11).

6. ТЕСТИРОВАНИЕ УСТРЙОСТВА ПРИ НИЗКИХ ТЕМПЕРАТУРАХ

Так как при низких температурах батарея разряжается быстрее из-за того, что уменьшается скорость химических реакций, аккумулятор не может в прежнем режиме давать энергию, поэтому мы утеплили аккумулятор, который питает датчики, пенополиуретаном (Рис. 13).

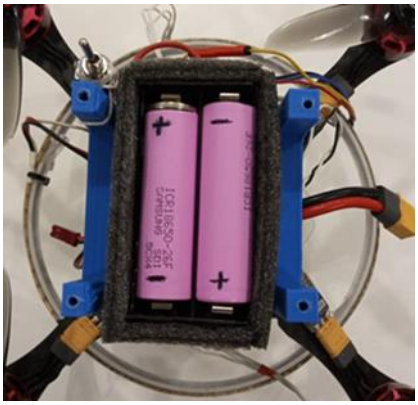


Рис. 13. Утепленный аккумулятор

Благодаря своей структуре, пенополиуретан обладает отличными теплоизоляционными свойствами, также он имеет низкую плотность, является лёгким материалом, что уменьшает нагрузку дрона. В первый раз датчики проработали 11 минут при температуре -30, но мы не проверили начальный заряд аккумулятора. Поняв свою ошибку, зарядили аккумулятор на полную мощность и повторили эксперимент при температуре -35°C (Рис.14).

Рис. 14. Процесс тестирования

Основная цель исследования заключалась в оценке стабильности работы и реакционной способности датчиков при низких температурах.

Результаты показали, что большинство датчиков демонстрировали удовлетворительную производительность при температурах до -35°C. Заряд батареи продержался 46 минут, этого достаточно для проведения измерений. Благодаря теплоизоляционному слою осуществилась длительная работа датчиков при низких температурах. В дальнейшем планируется проводить исследования для оптимизации характеристик существующих моделей или разработки новых, более устойчивых к температурным колебаниям.

7. ИНФОРМАЦИОННАЯ КАМПАНИЯ
Создание справочника и постера

В ходе работы мною был сделан печатный справочник-путеводитель и постер с целью распространения информации о загрязняющих воздух веществах и их вреде, о мерах безопасности при повышенных концентрациях в воздухе (Рис.15, 16).

Рис. 15. Справочник

Рис. 16. Постер «Чем мы дышим?»

Создание сайта

На данном сайте размещена информация о качестве воздуха в разных районах, о причинах загрязнения воздуха, загрязняющих веществах, рекомендации для здоровья, о систематизации экологического мониторинга качества воздуха с использованием дрона. Сайт имеет удобный механизм общения с системой, каждый гражданин может сообщить об экологической проблеме. Ссылка: https://kachestvvozdukha.tilda.ws/.

8. ИСПОЛЬЗОВАНИЕ РЕЗУЛЬТАТОВ

Области применения

Данный способ измерения качества воздуха может быть успешно интегрирован в различные сферы, в особенности в труднодоступные территории, ведь сбор данных в зонах с отсутствующей инфраструктурой минимизирует необходимость в сложных и опасных наземных экспедициях.

- Городская среда: обеспечивает оперативный анализ воздуха в разных районах для выявления очагов загрязнения и отслеживания распространения вредных веществ и установления их источников.
- Малонаселенные пункты республики Саха (Якутия): значительная экономия ресурсов за счет использования одного дрона для обслуживания группы соседних деревень. Мобильность позволяет отказаться от установки дорогостоящих станций в каждом малом пункте, обеспечивая при этом надежный мониторинг последствий лесных пожаров, переноса воздушных масс в малонаселенных пунктах, имеющих высокую социальную значимость для региона.
- Пожароопасные периоды: оценка задымлённости и состояния атмосферы для обоснования ограничительных мер и повышения оперативности решений при ЧС.
- Особо охраняемые природные территории: проведение экологического мониторинга в заповедниках и национальных парках без нарушения экосистем для оценки воздействия на уникальные природные объекты.

Таблица 11. Реализация разработки до действующего образца

№	Наименование этапа	Срок	Описание
1	Разработка защиты от метеорологических факторов	1,5 месяца	Улучшение конструкции с целью защиты от ветра, дождя.
2	Тестовые взлеты на улице	1 неделя	Проверка устойчивости к метрологическим факторам по ряду траекторий и маневров.
3	Создание электросхемы	1 месяц	Проектирование схемы, замена провода медными дорожками на текстолите. Припаивание компонентов к пластине, создается компактный и надежный модуль.
4	Сравнение с лабораторными газоанализаторами	2 недели	Сравнение показаний датчиков, установленных на дроне, с лабораторными

			приборами как завершающий этап перед сертификацией устройства
5	Документация для получения сертификата соответствия	2 месяца	получения сертификата соответствия: технический паспорт, техническое описание, руководство по эксплуатации, полученные результаты.

9. Сравнение с существующими реализованными аналогами

Таблица 12. Сравнение с аналогами

Фото	Название	Фирма	Плюсы	Минусы
	Дрон для мониторинга качества воздуха	PranaAir, Индия	Возможности пространственного картографирования с высоким разрешением доступ в отдаленные районы	Точность датчиков указана без численных значений. Интеграция с ГИС и детали калибровки неясны.
	Платформа DJI Matrice 300 RTK с газоанализатором Sniffer4D V2.0	BRLab Лаборатория дронов	Устройство имеет активный воздухозаборник. Газоанализатор легко снимается	Привязан к конкретной платформе (DJI Matrice). Стоимость высокая, данные по калибровке поверхностны.

10. АНАЛИЗ БИЗНЕС-ПРИВЛЕКАТЕЛЬНОСТИ РАЗРАБОТКИ

Потенциальные заказчики

- Государственные органы (Министерство экологии): для контроля соблюдения экологических норм, формирования отчетности о состоянии атмосферы и мониторинга зон ЧС.
- Промышленные предприятия (заводы, добывающие компании): для контроля выбросов в санитарно-защитных зонах, соблюдения квот и предотвращения штрафов.
- Муниципальные администрации: для оперативной проверки жалоб жителей на качество воздуха и защиты населения в периоды лесных пожаров.
- Природоохранные структуры (заповедники, лесничества): для дистанционного экологического мониторинга территорий без вмешательства в экосистему.

ЗАКЛЮЧЕНИЕ

1. Собран и запрограммирован прототип дрона-эколога, в который внедрены датчики для определения следующих параметров: угарный газ, твердые частицы, влажность, температура.
2. Исследование подтвердило, что выбор высоты измерений критически важен для минимизации помех, создаваемых дроном. Комплексный подход с использованием нескольких статистических метрик позволил объективно определить оптимальную высоту.
3. Благодаря теплоизоляционному слою осуществилась длительная работа датчиков при температуре -35ºС.
4. Разработан сайт, с помощью которого люди могут узнать информацию о качестве воздуха и позаботиться о своем здоровье, обратиться напрямую о возникшей проблеме.
5. В дальнейшем будет реализоваться работа с службами по охране экологии Якутска, создание эффективной системы практик предотвращения загрязнения воздуха.
6. В дальнейшем планируется проведение систематических исследований методической погрешности, обусловленной влиянием БПЛА на измеряемые параметры атмосферы. Это станет основой для возможной сертификации методики измерений с использованием БПЛА.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Как загрязнение воздуха разрушает наше здоровье : [Электронный ресурс] / Всемирная организация здравохранения: [сайт] — URL: https://www.who.int/ru/health-topics/air-pollution#tab=tab_1 (дата обращения: 28.11.2023). - Текст: электронный.

2. Мансуров, Р. Ш., Гурин, М. А., Рубель Е. В. Влияние концентрации углекислого газа на организм человека/ Р. Ш. Мансуров, М. А. Гурин, Е. В. Рубель// Universum: Технические науки: электрон. научн. журн.– 2017. – №8(41)) / [Электронный ресурс]. – Режим доступа: https://7universum.com/ru/tech/archive/item/5045

3. Воздействие загрязнённого воздуха на здоровье: [Электронный ресурс] /Фонд защиты окружающей среды: [сайт] — URL: https://www.edf.org/health/effects-of-air-pollution (дата обращения: 28.02.2023). - Текст: электронный.

4. Опаловский, А. А. Планета Земля глазами химика / А. А. Опаловский. — М.: Наука, 1990. — 224 с.

5. Канина, А.Р., Влияние загрязненного воздуха на здоровье человека / А. Р. Канина // Журнал

6. The Scientific Heritage – 2021. —№78— С. 15-16

7. Kayvon Sharghi Exploring Drone Aerodynamics with Computers / Kayvon Sharghi [Электронный ресурс] // Nasa : [сайт]. — URL: https://www.nasa.gov/aeronautics/exploring-drone-aerodynamics-with-computers/ (дата обращения: 10.11.2024).

8. Субботина, И. Е. , Ремезова, М.С., Буевич, А.Г., Сергеев, А.П., Баглаева, Е.М., Шичкин, А.В., Буторова, А.С., Сергеева, М.В. Вертикальное распределение характеристик запыленности приземного слоя атмосферного воздуха г. Екатеринбурга / И. Е. Субботина, М. С. Ремезова, А. Г. Буевич [и др.] // Траектория исследований – человек, природа, технологии. – 2022. – № 3(3). – С. 60-72 – УДК 502.057:504.3.054.

9. Рассеивание загрязняющих веществ в атмосфере [Электронный ресурс] // ГорЭко. – URL: https://goreco.ru/articles/rasseivanie-zagryaznyayushhih-veshhestv-v-atmosfere/ (дата обращения: 08.01.2026).

10. Хисматуллина А. Ф., Гасилов В. С., Чижова М. А., Хайруллина Л. И. Влияние метеорологических условий на распространение токсичных веществ в окружающем

пространстве [Электронный ресурс] // CyberLeninka. – URL: https://cyberleninka.ru/article/n/vliyanie-meteorologicheskih-usloviy-na-rasprostranenie-toksichnyh-veschestv-v-okruzhayuschem-prostranstve (дата обращения: 08.01.2026).

11. Вес воздуха: определение [Электронный ресурс] // TutorOnline. – URL: https://wika.tutoronline.ru/fizika/class/7/ves-vozduha (дата обращения: 08.01.2026).

12. Атмосфера Земли [Электронный ресурс] // РУВИКИ. – URL: http://ru.ruwiki.ru/wiki/Атмосфера_Земли (дата обращения: 08.01.2026).

ШИФРОВАНИЕ МЕТОДОМ ГАММИРОВАНИЯ

Жук Илья Владиславович

Звездный городок п, МБОУ СОШ имени В.М. Комарова с углубленным изучением английского языка, 11 класс

Секция: Информационная безопасность и цифровая криминалистика

Научный руководитель: Троицкий Игорь Иванович, к.т.н., доцент каф. ИУ8 МГТУ им. Н.Э. Баумана

АННОТАЦИЯ

В современном обществе информация — это один из самых ценных ресурсов, который требует защиты от несанкционированного проникновения лиц, не имеющих к ней доступа. Поэтому предметом исследования проекта определены следующие аспекты криптографии: шифрование методом гаммирования, относящееся к потоковому симметричному методу шифрования, а также генераторы случайных чисел.

Цель проекта: разработка и реализация универсальной модели гарантированно стойкого шифрования методом гаммирования на языке программирования Python для различных операционных систем (ОС), обеспечивающей защиту конфиденциальной текстовой информации при отправке сообщений по незащищённым каналам.

В ходе работы использовались следующие основные методы исследования и приёмы: для анализа теоретических основ шифрования методом гаммирования и разработки критериев криптостойкости использован анализ источниковой базы. Для оценки генераторов псевдослучайных чисел (ГПСЧ) на предмет соответствия криптостойкости - методы сравнительно-сопоставительного и табличного анализа. Для разработки программного обеспечения ПО - встроенные модули языка программирования Python.

На основе анализа основ шифрования методом гаммирования и функционирования генераторов случайных чисел (ГСЧ) разработана методика, на основании которой разработан и реализован прототип модели гарантированно стойкого шифрования. В ходе исследований был проведён анализ системных ГПСЧ различных ОС, разработана методика по передаче ключа для симметричного шифрования методом гаммирования, которая успешно реализована в модели гарантированно стойкого шифрования. По результатам исследований разработаны предложения по созданию универсального криптостойкого системного ГПСЧ для существующих и перспективных ОС.

ВВЕДЕНИЕ

В век, характеризованный возросшей ролью информационных взаимодействий инфраструктур и субъектов, осуществляющих сбор, формирование, распространение и использование информации, обеспечение безопасности её обмена как никогда важно и актуально [1].

Одним из основных методов для обеспечения безопасности информации является её шифрование. Актуальность шифрования информации определяется прежде всего тем, что в настоящее время информация как одна из самых основных составляющих безопасности общества, требует защиты от несанкционированного проникновения. В условиях растущих угроз кибербезопасности роль и значимость шифрования существенно возрастает.

Способов шифрования информации существует огромное множество. Одним из самых распространённых является шифрование методом гаммирования. Однако далеко не всегда при применении данного способа используются надёжные генераторы случайных чисел, что в свою очередь часто приводит к снижению стойкости шифрованной информации. Именно поэтому **целью проекта** является разработка и реализация универсальной модели гарантированно стойкого шифрования методом гаммирования на языке программирования Python для различных операционных систем, где наибольшее внимание уделяется разработке и применению криптостойких генераторов случайных чисел.

Для достижения указанной цели в работе поставлены следующие **задачи**:

1 Анализ шифрования методом гаммирования и функционирования генераторов случайных чисел.

2 Разработка методики шифрования/расшифрования методом гаммирования с использованием языка программирования Python.

3 Разработка и реализация прототипа модели гарантированно стойкого шифрования.

4 Исследование генераторов псевдослучайных чисел различных операционных систем.

5 Разработка методики по передаче ключа для симметричного шифрования методом гаммирования.

6 Разработка и реализация модели гарантированно стойкого шифрования методом гаммирования.

7 Разработка и реализация алгоритма повышения безопасности обмена шифрованными текстами.

8 Разработка предложений по созданию универсального криптостойкого системного ГПСЧ для существующих и перспективных ОС.

Разработка методики тестирования системных ГПСЧ на гарантированную

9 псевдослучайность в соответствии со стандартом NIST SP 800-22 Revision 1a.

10 Подтверждение результатов исследований по разработке и реализации универсальной модели гарантированно стойкого шифрования методом гаммирования на основе тестирования NIST.

Основная часть работы состоит из 6-и разделов. Дополнительные материалы (тексты разработанных методик, листинги программ) представлены в приложениях к работе.

В ходе работы над проектом были применены следующие основные методы исследований и приёмы решения задач:

1. Для исследования теоретических основ шифрования методом гаммирования и составления критериев криптостойкости проведён анализ источниковой базы, состоящей из 39 источников, включая 29 иностранных источников.

2. Для оценки ГПСЧ на предмет соответствия криптостойкости использовались методы сравнительно-сопоставительного и табличного анализа.

3. Для разработки ПО использовались встроенный модуль Python os ввиду возможности взаимодействия с системными ГПСЧ и библиотека cryptography для реализации соответствующих криптографических алгоритмов.

4. Для проведения тестирования ГПСЧ на гарантированную псевдослучайность в соответствии со стандартом NIST SP 800-22 Revision 1a используется командная строка Unix-подобных систем, её команды, сравнительно-сопоставительный и табличный анализ.

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ШИФРОВАНИЯ МЕТОДОМ ГАММИРОВАНИЯ И ФУНКЦИОНИРОВАНИЯ ГЕНЕРАТОРОВ СЛУЧАЙНЫХ ЧИСЕЛ, ПОСТАНОВКА ЗАДАЧИ

Шифрование методом гаммирования – это метод поточного симметричного шифрования (шифрование и расшифрование происходит с использованием одного ключа), подразумевающий последовательное сложение символов открытого текста с символами некоторой специальной последовательности, которая называется гаммой [2].

Основные преимущества шифрования информации методом гаммирования заключаются в следующем [2, 3, 4]: 1) гарантированная стойкость шифрованного текста; 2) большая скорость получения шифрованных текстов, так как операция сложения по модулю 2 очень быстро выполняется на вычислительных устройствах и поэтому наложение гаммы даже на очень большой текст выполняется практически мгновенно.

Однако основным недостатком шифрования методом гаммирования является: невозможность отправки одного шифрованного сообщения нескольким адресатам.

Необходимо отметить, что качество гаммы в свою очередь будет определять стойкость шифрования методом гаммирования [5, 6]. Поэтому для обеспечения гарантированной стойкости необходимо, чтобы последовательность символов в пределах периода гаммы обладала следующими свойствами [3]: была случайной (должна отсутствовать закономерность в появлении символов гаммы); распределение символов алфавита гаммы должно быть близко к равномерному (равновероятному) распределению; совпадала по размеру или была больше шифруемого открытого текста; применялась только один раз.

Несмотря на все достоинства шифрования методом гаммирования, одной из ключевых проблем его применения на практике является получение качественных гамм. Для процедур получения гамм можно использовать генераторы истинно случайных чисел (ГИСЧ) или криптостойкие генераторы псевдослучайных чисел (ГПСЧ).

ГПСЧ генерирует псевдослучайную последовательность чисел, которая была вычислена по определённой процедуре, но имеет все свойства случайной последовательности чисел в рамках решаемой задачи. Отличие истинно случайных последовательностей от псевдослучайных в этом случае заключается в невозможности предсказания (расчёта, определения) символов в ней [2].

ГИСЧ генерирует свои последовательности путём оцифровки физических или антропогенных процессов. Для большинства генераторов на базе физических и антропогенных процессов характерны определённые недостатки [3]: сложность решения вопроса передачи ключа по сравнению с ГПСЧ; медленная скорость генерации гамм; антропогенные процессы могут иметь скрытые зависимости – каждый пользователь обладает своим «подчерком» работы

с компьютером.

Таким образом, в работе принято, что для генерации гаммы используется ГПСЧ, генерирующий свои последовательности путём применения рекуррентных формул или полноценных алгоритмов. При этом криптостойкий ГПСЧ должен обладать следующими свойствами [6, 7]:

1) энтропийность ключа (инициализирующего значения для ГПСЧ) и его длина должны быть не менее 128 бит;
2) невозможностью определения следующих и предыдущих членов последовательности по известным **k** членам последовательности, которую сгенерировал ГПСЧ;
3) период, в течение которого генератор не повторяет свою последовательность, должен быть как можно больше, чтобы предотвратить возможность атак перебора (минимальный период, которым должен обладать ГПСЧ для криптографических приложений – 2^256);
4) распределение символов последовательности, которую генерирует ГПСЧ, должно быть близко равномерному (равновероятному).

Также одним из важнейших требованием криптостойкого ГПСЧ является эффективность алгоритма, а именно большая скорость его работы и малые затраты памяти.

При шифровании методом гаммировании ГПСЧ должны быть синхронными или самосинхронизирующимися [2]. В синхронном ГПСЧ гамма генерируется независимо от потока сообщения, в самосинхронизирующимся ГПСЧ гамма зависит от передаваемого вместе с сообщением заголовка длины **n** бит, который в свою очередь не является ключом для ГПСЧ. Для криптографических приложений инициализирующее значение, которое служит входными данными для ГПСЧ, должно быть защищённым. Поскольку ГПСЧ является детерминированным алгоритмом, то злоумышленник может вычислить исходное значение и тогда выходные данные также могут быть определены. Следовательно, исходное значение должно быть непредсказуемым. Фактически, само начальное значение должно быть случайным или псевдослучайным числом. Как правило, начальное значение может генерироваться с помощью системы, состоящей из ГСЧ и ГПСЧ, как это представлено на рисунке 1 [5]. При этом минимальным требованием по длине инициализирующего значения является 128 бит.

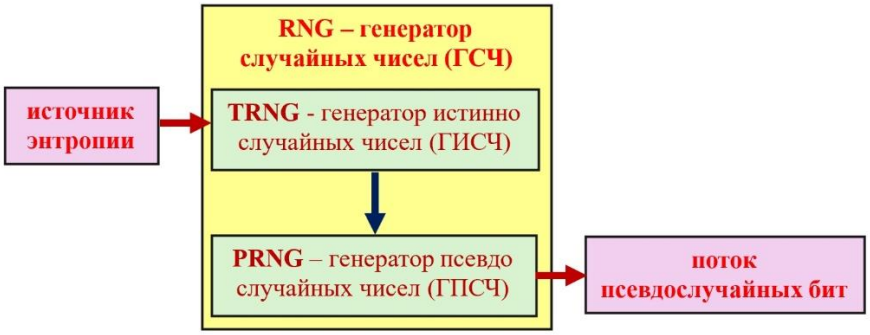


Рис. 1. Генерация начального значения для ГПСЧ

Необходимо отметить, что ГПСЧ в ГСЧ должен быть криптостойким, но помимо тех свойств, какими обладает ГПСЧ, генерирующий гамму, ГПСЧ в ГСЧ в соответствии со стандартом NIST SP 800-90A rev.1 должен обладать следующим свойством [10]: устойчивость к предсказаниям из будущего или прошлого, то есть, если состояние генератора в момент времени **t** стало известно атакующему, он не должен иметь возможность восстановить предыдущие выходные данные, сгенерированные до момента **t**, а также последующие выходные данные, сгенерированные после момента **t**.

Важной проблемой при шифровании методом гаммирования является вопрос передачи ключа (инициализирующего значения для ГПСЧ). Для этого должны использоваться ассиметричные системы шифрования (с открытыми ключами) или физические способы передачи ключа [3, 9].

Вывод: исходя из результатов анализа теоретических основ шифрования методом гаммирования были определены основные требования (осуществлена постановка задачи) к последовательности, которую генерируют ГИСЧ или ГПСЧ, который должен быть криптостойким и способен воспроизводить одинаковые последовательности у отправителя и получателя.

РАЗРАБОТКА И РЕАЛИЗАЦИЯ ПРОТОТИПА МОДЕЛИ ГАРАНТИРОВАННОГО СТОЙКОГО ШИФРОВАНИЯ МЕТОДОМ ГАММИРОВАНИЯ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON

Для создания прототипа модели гарантированно стойкого шифрования, а также генератора случайных чисел (ГСЧ), необходимого для работы программы, выполняющей функцию генерирования гаммы, определён модуль random языка программирования Python. Кроме того, для решения поставленной задачи использованы встроенные функции Python, такие как: ord() - для представления символов в виде целых чисел; bin() - для перевода целых чисел в двоичный вид; chr() - для представления целых чисел в виде символов; int() - для

перевода чисел из двоичной системы счисления в десятичную.

С учётом изложенного разработана обобщённая схема шифрования / расшифрования методом гаммирования, которая представлена на рисунке 2. В соответствии с представленной схемой разработана методика шифрования/расшифрования методом гаммирования с использованием языка программирования Python, которая представлена в приложении А.

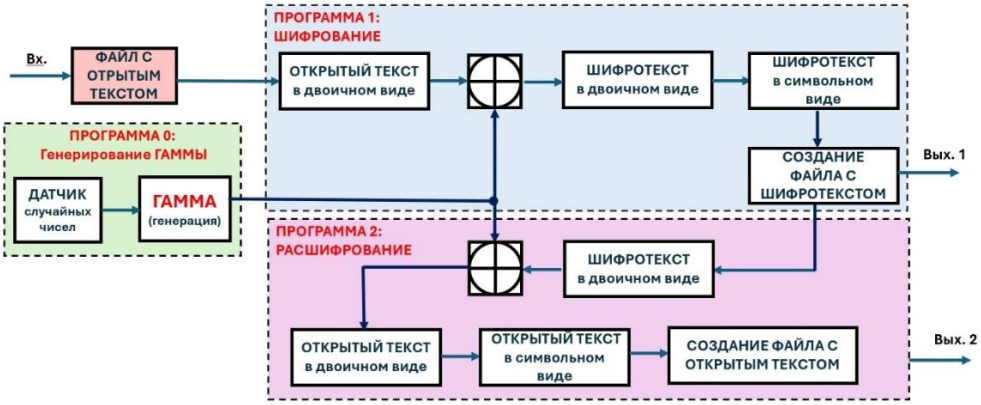


Рис. 2. Обобщённая схема шифрования/расшифрования методом гаммирования

Предлагаемая схема предполагает решение вопроса о передаче ключа при шифровании методом гаммирования. Поэтому для решения проблемы обмена ключами для ГПСЧ в современных условиях, когда вычислительные ресурсы компьютеров очень велики, целесообразно использовать ассиметричный метод шифрования RSA-4096 с обменом отпечатками ключей [11]. Указанный способ шифрования реализован в прототипе модели гарантированно стойкого шифрования методом гаммирования [12] с учётом разработанной методики по передаче ключа для симметричного шифрования методом гаммирования, представленной в приложении Б.

Несмотря на то, что в прототипе модели гарантированного стойкого шифрования была успешно реализована сущность шифрования методом гаммирования (успешно было произведено наложение случайной гаммы на открытый текст и его расшифровка), ГСЧ, основанный на базовых функциях модуля random языка программирования Python и который использовался при генерации гаммы, всё же являлся не криптостойким.

Объекты класса random.Random() (базовые функции модуля random) в качестве ГСЧ используют алгоритм Mersenne Twister (Вихрь Мерсенна), который генерирует числа с плавающей точкой 53-битной точности и имеет период 2^19937-1. Однако будучи полностью детерминированным, он подходит не для всех целей и совершенно не подходит для криптографических [13]. Соответственно, реализованный прототип модели гарантированно стойкого шифрования не обеспечивает необходимой конфиденциальности общения.

Применённый ассиметричный метод шифрования RSA-4096 уступает в

производительности другим методам аутентификации [11]. Поэтому для совершенствования предложенной модели целесообразно внедрить улучшенные контрмеры против инцидентов безопасности, связанных с перехватом информации и несанкционированным анализом трафика. Для улучшения безопасности разработанной модели в работе были использованы требования стандарта ГОСТ-Р-56205-2014 «Сети коммуникационные промышленные: Защищённость (кибербезопасность) сети и системы», в котором установлены концептуальные положения по обеспечению безопасности коммуникации [15].

Доработанная схема шифрования/расшифрования методом гаммирования с использованием соответствующих контрмер принята за основу разрабатываемой модели гарантированно стойкого шифрования методом гаммирования и представлена на рисунке 3.

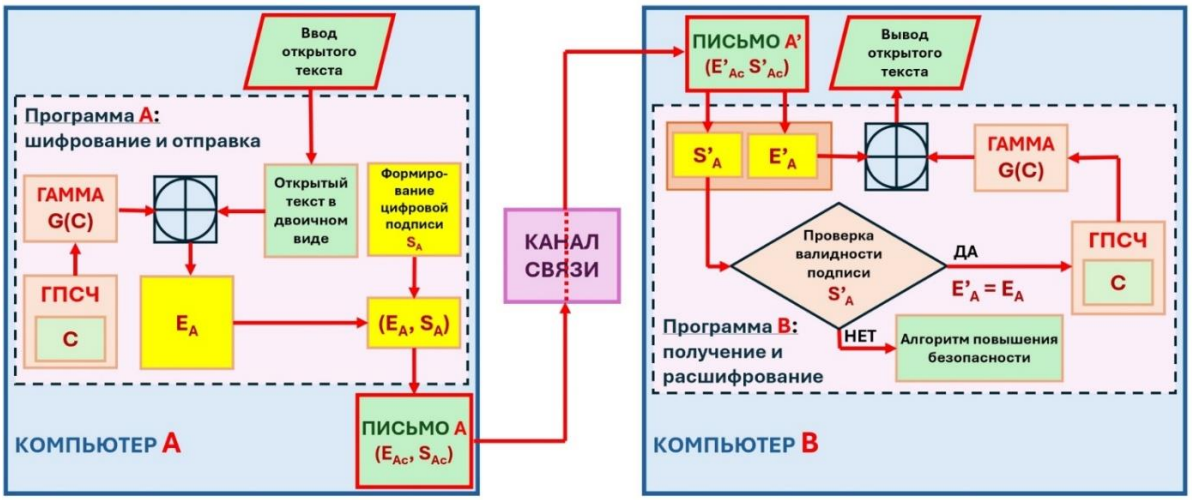


Рис. 3. Модель гарантированно стойкого шифрования методом гаммирования

На рисунке G – генерируемая криптостойкая последовательность псевдослучайных бит, гамма; C – общий секретный ключ; EA, SA – шифротекст и цифровая подпись в двоичном виде; E'A, S'A – полученные шифротекст и цифровая подпись в двоичном виде; EAc, SAc – шифротекс и цифровая подпись в символьном виде; E'Ac, S'Ac - полученные шифротекст и цифровая подпись в символьном виде.

Предлагаемые контрмеры включают в себя:

1. Цифровую подпись на базе эллиптической кривой – это результат криптографического преобразования данных, выполняемого с использованием операций над точками эллиптической кривой, который, при условии правильной реализации этого преобразования, предоставляет сервисы аутентификации источника, целостности данных и гарантию сохранения авторства подписавшегося.
2. Алгоритм повышения безопасности обмена шифрованными текстами (представлен в разделе 4), а также меру, направленную против несанкционированного анализа трафика.

ИССЛЕДОВАНИЕ КРИПТОСТОЙКИХ СИСТЕМНЫХ ГПСЧ В РАЗЛИЧНЫХ ОС

Для решения указанных выше проблем были проведены исследования других вариантов (проектных решений и способов) по реализации криптостойкого ГПСЧ с помощью возможностей языка программирования Python на различных ОС. При этом было принято решение: рассмотреть и исследовать возможность использования в модели гарантированно стойкого шифрования функции os.urandom() языка программирования Python, которая использует системные ГСЧ.

Анализ реализации функции os.urandom() на семействе Unix-подобных систем на базе ядра Linux

Основная особенность Linux заключается в том, что если доступен системный вызов getrandom(), то он блокируется до тех пор, пока не будет инициализирован системный пул энтропии (ядро собирает 256 бит энтропии) для повышения безопасности [5].

Getrandom() в режиме блокировки основан на работе специального символьного файла /dev/urandom с запретом генерировать псевдослучайные биты до тех пор, пока не наберётся достаточно начальной энтропии, и никогда не блокирует после этого момента [16]. Обобщённая схема Linux-ГСЧ (Linux-RNG) представлена на рисунке 4.

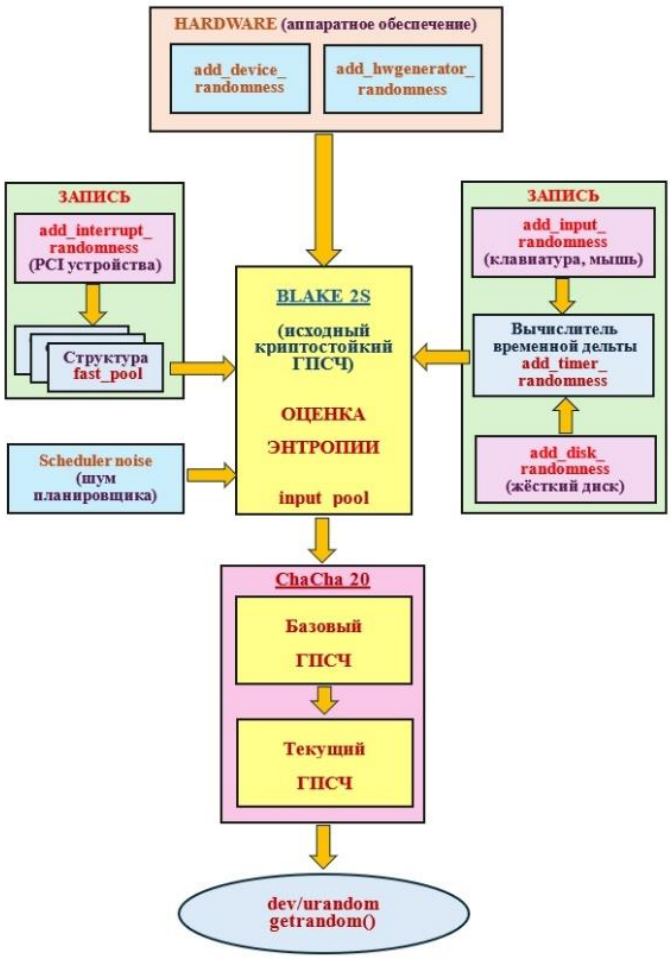


Рис. 4. Обобщённая схема Linux-ГСЧ

В соответствии с представленной схемой работу Linux-ГСЧ можно охарактеризовать следующим образом [16, 17, 18, 19]:

1. После возникновения аппаратного события, такого как прерывание, время события и его значение хэшируются в пул энтропии, который называется пулом входных данных для Blake2s.
2. Linux-ГСЧ (Linux-RNG) присваивает этому событию оценку энтропии.
3. При генерации данных из входного пула выполняется заключительная операция хэширования Blake2s.

Это означает, что все данные с момента инициализации пула математически объединяются и затем хэшируются. Полученные результирующие данные обрабатываются аналогично механизму извлечения и расширения с использованием Blake2s.

На первом этапе (этапе извлечения) новый ключ генерируется из объединённых выходных данных пула входных данных и выходных данных в 256 бит из дополнительного ГИСЧ на базе центрального процессора. Полученный дайджест сообщения (полученный результат хэширования с помощью Blake2s) используется для повторной инициализации хэш-контекста Blake2s с целью обработки последующих данных из источников шума.

Результат первого этапа используется для выполнения фазы расширения для генерации случайных битов размером 32 байта, которые используются для заполнения ключа базового ChaCha20 ГПСЧ.

4. ChaCha20 ГПСЧ работает путём многократного вызова операции блока текущего ChaCha20 ГПСЧ до тех пор, пока не будет сгенерировано запрошенное количество байт. Следовательно, функция вывода этого ГПСЧ основана на операции блока текущего ChaCha20.

Счётчик в структуре блоков ChaCha20 в Linux-ГПСЧ представляет из себя 64-битное значение, а каждый блок текущего ГПСЧ выводит 512 бит [17]. Применение операции блока ChaCha20 изменяет состояние текущего ГПСЧ, то есть значение счётчика состояния увеличивается на единицу после каждой операции блока ChaCha20 [17], тогда до повторения последовательности может быть сгенерировано 2^64*2^512=2^578 значений. Длина последовательности до повторения – 2^578, тогда в соответствии с определением периода [7] период текущего ГПСЧ ChaCha20=2^578.

Для проверки на соответствие Linux-ГСЧ требованиям криптостойкого ГСЧ в работе выполнен анализ на предмет криптостойкости выходной ГПСЧ, который генерирует сами случайные биты. Полученные результаты исследований представлены в таблице 2.

Таблица 2. Оценка криптостойкости алгоритма генерации случайных чисел Linux-ГПСЧ

ChaCha20 с использованием счётчика	Минимальные критерии криптостойкости ГПСЧ в системном ГСЧ	Соответствие
Вызывающая сторона не может вывести последующие случайные числа из полученного ею случайного числа. ChaCha20 устойчив к определению использованного ключа путём анализа уже сгенерированных случайных чисел [16]. А функция, которая обеспечивает секретность вправо, обладает свойством справа [7].	Устойчивость к предсказаниям из будущего или прошлого: если состояние генератора в момент времени **t** стало известно атакующему, он не должен иметь возможность восстановить предыдущие выходные данные, сгенерированные до момента **t**, а также последующие выходные данные, сгенерированные после момента **t**.	Да
1 ГПСЧ ChaCha20 соответствует требованиям DRG.3.3 [16]: «ГСЧ обеспечивает обратную секретность, даже если известно текущее внутреннее состояние». 2 ГПСЧ ChaCha20 соответствует требованиям DRG.3.2 [16]: «ГСЧ обеспечивает секретность вправо,	Невозможность определения следующих и предыдущих членов последовательности по известным **k**, сгенерированными ГПСЧ [7].	Да

даже если известно текущее внутреннее состояние».		
Длина ключа – 256 бит [16].	Минимальная длина ключа – 128 бит [5].	Да
Период – 2^578.	Минимальный период для ГПСЧ – 2^256 [6].	Да
Блоки ChaCha20 являются близким приближением к равномерному распределению, поддерживаемому лавинным эффектом [16].	Распределение символов последовательности для ГПСЧ, должно быть близко равномерному (равновероятному).	Да

Ключ для текущего ChaCha20 определяется младшими выходными битами базового ChaCha20. При этом ГПСЧ, который генерирует ключ для базового ChaCha20, должен быть также криптостойким соответственно. За генерацию ключа для базового ChaCha20 в Linux-ГПСЧ отвечает хэш-функция Blake2s. И хотя её нельзя назвать ГПСЧ, хэш-функция всё же имеет все свойства ГПСЧ. Роль хэш-функции огромна, так как она обеспечивает непредсказуемость поступивших данных в пул энтропии после их обработки [20, 21]. Результаты оценки криптостойкости алгоритма Blake2s при формировании ключа для ChaCha20 представлены в таблице 3.

Таблица 3. Оценка криптостойкости алгоритма Blake2s

Blake2s	Минимальные критерии криптостойкости ГПСЧ в системном ГСЧ	Соответ-ствие
Blake2s – криптостойкая хэш-функция [19].	Устойчивость к предсказаниям из будущего или прошлого: если состояние генератора в момент времени **t** стало известно атакующему, он не должен иметь возможность восстановить предыдущие выходные данные, сгенерированные до момента **t**, а также последующие выходные данные, сгенерированные после момента **t**.	Да
Безопасность Blake2 эквивалентна безопасности HMAC (при использовании секретного ключа), то есть хэш-функция проходит тест на следующий бит [19], функция обладает свойством справа [7].	Невозможность определения следующих и предыдущих членов последовательности по известным **k** членам последовательности, которую сгенерировал ГПСЧ [7].	Да
Длина ключа – 256 бит [16].	Минимальная длина ключа – 128 бит [13].	Да
Один блок Blake2s выводит 256-битный хэш, значит его период – 2^256.	Минимальный период для ГПСЧ – 2^256 [6].	Да

Хэш-функция Blake2 –современная хэш-функции с превосходными статистическими свойствами, что подразумевает равномерность распределения выходных данных [20].	Распределение символов последовательности для ГПСЧ, должно быть близко равномерному (равновероятному).	Да

Исходя из проведённого выше анализа на криптостойкость Linux-ГПСЧ можно сделать вывод, что реализация функции os.urandom() на семействе Unix-подобных систем на базе ядра Linux полностью подходит для модели гарантированно стойкого шифрования методом гаммирования.

Хэширование с помощью Blake2s с использованием ключа даёт баланс между криптостойкостью и эффективностью. Данное утверждение касается и ChaCha20 [20, 21].

АНАЛИЗ РЕАЛИЗАЦИИ ФУНКЦИИ OS.URANDOM() НА UNIX-ПОДОБНЫХ СИСТЕМАХ OpenBSD, MacOS и Free BSD

В OpenBSD чтение из устройства dev/urandom имеет примерно такую же схему, как и в Linux, но с небольшими отличиями: хэширование пула не происходит. Однако гораздо большее внимание уделяется сбору энтропии из аппаратных ГПСЧ, благодаря чему в пуле постоянно находится достаточно энтропии после его инициализации. Вместо ChaCha20 используется AES-256-CTR – ГПСЧ, утверждённый стандартом NIST SP 800-90A rev.1 [22, 23]. Таким образом чтение из dev/urandom при реализации функции os.urandom на OpenBSD подходит для реализации криптографически стойкого ГПСЧ.

Чтение случайных чисел из /urandom на MacOS зависит от устройства: если есть возможность использовать Secure Enclave, то случайные числа будут генерироваться с помощью шифрования методом AES-256 данных аппаратного ГСЧ, который использует физические источники шума. В противном случае используется криптостойкий ГПСЧ, детали работы которого засекречены [24]. Однако известно, что этот ГПСЧ соответствует стандартам NIST SP 800-90A rev.1 [23, 25]. Из чего можно сделать вывод, что чтение из dev/urandom при реализации функция os.urandom на MacOS подходит для реализации криптографически стойкого ГПСЧ.

В FreeBSD чтение из устройства dev/urandom использует алгоритм Fortuna, имеющий со схемой Linux следующие различия [26, 27]: 1) вместо одного входного пула используется 32 подпула для сбора энтропии, что защищает от атак, связанных с компрометацией одного источника; 2) преобразование данных в пуле происходит с помощью блочного AES шифрования; 3) генерация выходных данных происходит с помощью блочного шифрования AES со счетчиком, при этом ключ для шифрования обновляется каждые сгенерированный

Мбайт данных.

Можно сделать вывод, что реализация функции os.urandom() на других Unix-подобных системах полностью подходит для модели гарантированно стойкого шифрования методом гаммирования.

АНАЛИЗ РЕАЛИЗАЦИИ ФУНКЦИИ OS.URANDOM() НА ОС Windows

В настоящее время корпорация Microsoft, которая создала ОС Windows, не раскрывает подробностей алгоритма криптостойкого ГПСЧ (CSPRNG). Известно, что ГПСЧ Windows удовлетворяет NIST SP800-90 стандартам [28]. Исходя из чего можно сделать вывод (предположить), что реализация функции os.urandom на ОС Windows подходит для реализации криптографически стойкого ГПСЧ в модели гарантированно стойкого шифрования методом гаммирования.

Вывод:

1) генерация случайных чисел с помощью функции языка программирования Python os.urandom() для всех рассмотренных выше ОС является криптостойкой;
2) функция os.urandom() удовлетворяет критериям (условиям) для реализации ГСЧ в универсальной модели гарантированно стойкого шифрования на языке программирования Python для различных ОС.

РАЗРАБОТКА И РЕАЛИЗАЦИЯ МОДЕЛИ ГАРАНТИРОВАННО СТОЙКОГО ШИФРОВАНИЯ МЕТОДОМ ГАММИРОВАНИЯ

В процессе исследований, представленных выше, был затронут вопрос о передаче ключа при шифровании методом гаммирования. При этом отмечено, RSA-4096 обладает крайне низкой производительностью. Поэтому для решения проблемы получения общего ключа (инициализирующего значения для ГПСЧ) в модели гарантированно стойкого шифрования целесообразно использовать алгоритм X25519, обладающей крайне высокой производительностью благодаря использованию кривой Curve25519 [29] и одобренный стандартом RFC 7748: Elliptic Curves for Security [30].

Методика получения общего инициализирующего значения для ГПСЧ (ключа C) представлена в приложении B.

При разработке модели гарантированно стойкого шифрования методом гаммирования учтено, что несмотря на то, что функция os.urandom() использует системный ГСЧ, синхронизация системных ГПСЧ на различных устройствах невозможна ввиду постоянного обновления энтропии, уникальной для каждого устройства. Именно поэтому в качестве ГПСЧ, генерирующего гамму, используется реализованный с помощью Python алгоритм ChaCha20 со счётчиком.

В качестве внедряемой технической меры, защищающей целостность данных, а также обеспечивающей аутентификацию субъектов, целесообразно использовать безопасную и при этом самую эффективную цифровую подпись – цифровую подпись на базе эллиптической кривой Curve25519 в форме Эдвардса [31, 32, 33]. Полноценная реализация модели осуществлена с помощью следующих модулей:

1. Модуль 1: «Обмен публичными ключами для подписи», реализующий создание цифровой подписи. Модуль включает четыре программы: 1) абонент A подпись(генерация ключевой пары для себя и отправка публичного); 2) абонент B подпись(генерация ключевой пары для себя и отправка публичного); 3) абонент A(прием публичного ключа для подписи); 4) абонент B (приём публичного ключа для подписи).
2. Модуль 2: «Обмен сессионными ключами для подписи», реализующий методики получения общего ключа у абонентов. Модуль включает 4 программы: 1) optm абонент A генерация сессионной ключевой пары, отправка публичного сессионного ключа; 2) optm абонент B генерация сессионной ключевой пары, отправка публичного сессионного ключа; 3) optm абонент A (приём публичного ключа для сессии); 4) optm абонент B (приём публичного ключа для сессии).
3. Модуль 3: «Абоненты» - сессия (защищённый обмен данными между абонентами) с применением соответствующих контрмер, включающих применение цифровой подписи и повышение безопасности обмена шифрованными текстами. Модуль включает две программы: 1) абонент A ses и 2) абонент B ses.

Листинги ПО, включающие в себя программы всех модулей, представлены в приложении Г.

В ходе исследований разработаны и реализованы алгоритмы по улучшению безопасности модели гарантированно стойкого шифрования методом гаммирования, обобщённая схема которых представлена на рисунке 5.

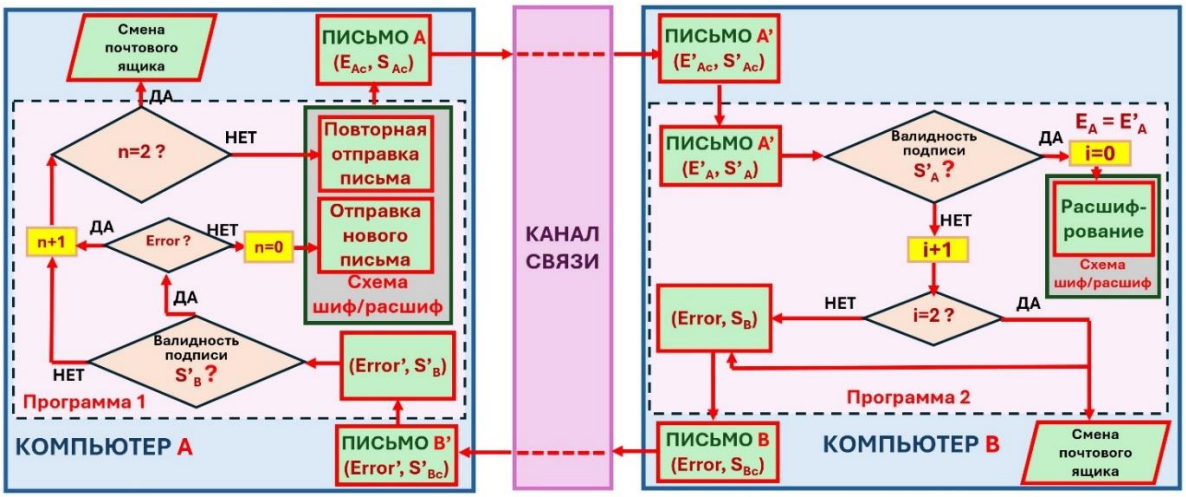


Рис. 5. Обобщённая схема алгоритмов повышения безопасности обмена шифрованными текстами

На рисунке n, i – счётчики, начальные значения которых равны 0; Error – сообщение об ошибке; SB – подпись от абонента B, получившего сообщение, не соответствующее подписи; S'Bc – полученная подпись от абонента B в символьном виде; S'B – полученная подпись от абонента B в двоичном виде.

В качестве контрмеры против пассивной атаки несанкционированного анализа трафика используются гаммы фиксированной длины для сообщений небольшой длины, чтобы злоумышленник не мог получить информацию об открытом тексте по длине шифротекста.

Взаимодействие между абонентами в случае, если в первой схеме шифрования/расшифрования методом гаммирования на рисунке 3 активируется ветвь «подпись не валидна», реализуется разработанным алгоритмом повышения безопасности обмена шифрованными текстами, описывающим внедряемую данным алгоритмом процедуру реагирования на инциденты безопасности, связанные с перехватом информации [20].

В соответствии с разработанным алгоритмом в случае полной компрометации среды для продолжения обмена текстовой информацией между абонентами происходит её смена, а в случае единичного перехвата информации и выявления этого – повторная отправка сообщения.

РАЗРАБОТКА ПРЕДЛОЖЕНИЙ ПО СОЗДАНИЮ УНИВЕРСАЛЬНОГО КРИПТОСТОЙКОГО СИСТЕМНОГО ГСЧ ДЛЯ СУЩЕСТВУЮЩИХ И ПЕРСПЕКТИВНЫХ ОС

В настоящее время ввиду накладываемых Западом санкций как никогда актуален вопрос совершенствования существующих и разработка перспективных системных ГСЧ.

С учётом проведённых исследований и полученных в работе научных результатов целесообразно за основу принять структуру Linux-ГСЧ (Linux-RNG). При этом необходимо

учитывать следующие характерные особенности современных Linux-ГСЧ.

1. Использование хэш-функций семейства BLAKE (Blake2, Blake2s и Blake 3) как самых быстрых и доступных в настоящее время криптостойких хэш-функций [20].
2. Использование ChaCha20 как основного алгоритма для генерации случайных чисел обеспечивает не только генерацию больших объёмов криптографических псевдослучайных данных, но и универсальность ГСЧ. Алгоритм ChaCha20 является независимым от специального оборудования и показывает значительные превосходства в скорости в сравнении с генерацией случайных бит с помощью AES шифрования на устройствах, где нет AES-оборудования, а также имеет сопоставимую скорость с генерацией случайных бит с помощью AES шифрования на тех устройствах, где есть AES-оборудование [34].

Исходя из основных требований к сбору и оценке энтропии для предлагаемого системного ГСЧ выделены следующие классы энтропии и входные пулы соответственно (за основы разделения взята структура Linux-ГСЧ) [16]:

1. Пул1 с энтропией на базе временных меток завершённых операций планировщика до первой инициализации базового ChaCha20.
2. Пул 2 с энтропией на базе данных драйверов устройств, включающий временные метки сбора информации. (Эти данные не будут инкрементировать счётчик энтропии, однако будут обеспечивать большее различие в генерации случайных последовательностей на разных устройствах, использующих одну и ту же ОС).
3. Пул 3 с энтропией на базе аппаратных ГПСЧ (при наличии таковых).
4. Пул 4 с энтропией на базе значений и временных меток нажатия клавиш, перемещения мыши и аналогичных событий.
5. Пул 5 с энтропией на базе временных меток произошедших событий на диске, типа событий.
6. Пул 6 с энтропией на базе временных меток аппаратных прерываний процессора 1.

N. Пул N с энтропией на базе временных меток аппаратных прерываний процессора N.

Для управления пулами предложено использовать счётчик энтропии (аналогично как в Linux-ГСЧ). Циклограмма функционирования предлагаемого универсального системного ГСЧ представлена в работе [12].

Вывод: Универсальные ГСЧ для существующих и перспективных ГПСЧ возможно разработать на базе комбинированных технологий Linux, MacOS, Windows, OpenBSD и FreeBSD. При этом целесообразно за основу принять технологии, реализующие ГСЧ на Linux системах.

ПОДТВЕРЖДЕНИЕ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЙ ПО РАЗРАБОТКЕ И РЕАЛИЗАЦИИ УНИВЕРСАЛЬНОЙ МОДЕЛИ ГАРАНТИРОВАННО СТОЙКОГО ШИФРОВАНИЯ МЕТОДОМ ГАММИРОВАНИЯ

Полученные выше результаты исследований позволили подтвердить криптостойкость используемых ГСЧ. Однако для практического доказательства данного вывода дополнительно было проведено тестирование NIST, которое является наиболее авторитетным и может быть принято за критерий достоверности полученных в работе результатов исследований [10, 35, 36].

Краткая характеристика тестирования в соответствии со стандартом NIST SP 800-22 Revision 1a [37]:

1. Тестирование включает в себя 15 независимых тестов первого уровня и 1 зависимый тест второго уровня, каждый из которых направлен на выявление определённых видов неслучайности: частотных аномалий, корреляций, периодичностей и других статистических отклонений. Тестирование проводится на большом объёме данных. Минимально рекомендуемая выборка – 100 последовательностей по 1000000 бит каждая.

   Результатом проведения теста на каждой последовательности является **p-value** - вероятность того, что эталонный истинно случайный генератор создаст последовательность, которая по проверяемому статистическому критерию будет выглядеть менее случайной, чем тестируемая последовательность. Последовательность проходит определённый тест, если p-value >= 0.01.
2. После проведения тестирования первого уровня создаётся набор P-valueT (набор значений теста второго уровня), каждое из которых определяется для набора p-value соответствующего статистического теста и показывает вероятность того, что идеально случайный генератор, на последовательностях которого был бы проведён этот же тест, дал бы набор p-значений, которые выглядят менее равномерно распределёнными, чем набор p-значений, который получился от последовательностей протестированного ГСЧ, а также набор proportion, каждое из которых показывает долю прошедших соответствующий тест последовательностей. В конечном итоге по оцениванию набора P-valueT в соответствии со стандартом и proportion в соответствии с инструкциями из файла finalAnalysisReport.txt определяется: обладают ли гарантированной псевдослучайностью последовательности, которые генерирует ГСЧ.

С учётом указанных выше требований разработана соответствующая «Методика тестирования ГСЧ в соответствии со стандартом NIST SP 800-22 Revision 1a»:

1. Создание минимально рекомендуемой выборки последовательностей бит ГСЧ в виде бинарного файла с использованием программы Python.
2. Созданный файл переносится на компьютер с Unix-подобной ОС для дальнейшего проведения тестирования последовательностей, так как тестирование должно проводиться в Unix-подобных операционных системах ввиду зависимости исходного кода от POSIX-специфичных системных вызовов.
3. Распаковка папки sts-2.1.2.zip разработчиков набора статистических тестов.
4. С используем утилиты make проводится чтение инструкции из файла Makefile, запускаются прописанные в нём команды в терминале для сборки итогового исполняемого файла из набора файлов с исходным кодом на языке С.
5. Запускается исполняемый файл с последующими инструкциями в терминале. На вход исполняемому файлу подаётся длина одной последовательности, сам бинарный файл с последовательностями, число последовательностей.
6. По завершению тестирования в папке experiments создаётся текстовый файл FinalAnalysis.txt с результатами первичного тестирования, который содержит инструкции по анализу набора долей последовательностей, прошедших определённый тест.
7. В соответствии с инструкциями из файла FinalAnalysis.txt и стандарта NIST SP-800 22 rev. 1 проводится 16-ый зависимый тест и анализ набора долей последовательностей, прошедших определённый тест, по результатам которых выдаётся заключение о гарантированной псевдослучайности ГСЧ (его генерируемых последовательностей).

В соответствии с разработанной выше методикой проведено тестирование системных ГСЧ и ГПСЧ на основе алгоритма ChaCha20. Результаты проведённого тестирования на примере ГСЧ ОС Windows и ГПСЧ на основе алгоритма ChaCha20 представлены в приложении Д.

ЗАКЛЮЧЕНИЕ ПО ТЕСТИРОВАНИЯ ГСЧ

1. В соответствии с проведённым 16-ым зависимым тестом (анализом набора P-valueT) и анализом данных из файла FynalAnalysis.txt следует, что все минимальные реальные значения P-valueT и долей больше или равны минимально допустимым значениям.

2. В соответствии со стандартом NIST SP 800-22 rev.1 и инструкциями из файла FinalAnalysis.txt можно сделать вывод, что последовательности системных ГСЧ, а также ГПСЧ, основанного на алгоритме ChaCha20 и реализующим функцию генерации гаммы, обладают гарантированной псевдослучайностью.

Вывод: Поставленные цели и задачи проекта выполнены в полном объёме.

ЗАКЛЮЧЕНИЕ

Актуальность гарантированно стойкого шифрования в современных условиях обусловлена тем, что информатизация и автоматизация всех сфер человеческой деятельности приводят к росту угроз несанкционированного доступа к информации. Поэтому необходимо постоянно поддерживать, обновлять и совершенствовать методы её защиты. Исходя из выше сказанного была поставлена следующая **цель**: разработка и реализация универсальной модели гарантированно стойкого шифрования методом гаммирования на языке программирования Python для различных ОС.

С учётом анализа основ шифрования методом гаммирования и функционирования генераторов случайных чисел были разработаны соответствующие методики шифрования/расшифрования методом гаммирования с использованием языка программирования Python, методика передачи ключа для симметричного шифрования методом гаммирования. На основе указанных методик разработан и реализован прототип модели гарантированно стойкого шифрования, программное обеспечение которого включает: 1) прототип ПО, способного генерировать криптостойкую гамму для шифрования текстовой информации; 2) прототип ПО, способного шифровать текст; 3) прототип ПО, способного расшифровывать текст; 4)прототип ПО, реализующего обмен ключами для ГПСЧ.

В ходе работы на основе анализа источниковой базы и метода сравнительно-сопоставительного анализа была исследована криптостойкость системных ГПСЧ различных операционных систем. Для подтверждения достоверности полученных теоретических результатов проведено тестирование NIST, которое подтвердило гарантированную псевдослучайность последовательностей используемых ГСЧ и ГПСЧ в модели. Разработаны и успешно реализованы в модели методика получения ключа для ГПСЧ, криптостойкий ГПСЧ, основанный на функциях языка программирования Python, алгоритмы по улучшению безопасности модели гарантированно стойкого шифрования, включающие в себя: цифровую подпись на базе эллиптической кривой, алгоритм повышения безопасности обмена шифрованными текстами, реализующий процедуру реагирования на инциденты безопасности, связанные с перехватом информации, контрмеру против несанкционированного анализа трафика. Модель успешно протестирована в условиях, когда абоненты используют разные устройства, что значительно расширяет её практическое применение по сравнению с прототипом. При этом выбор языка программирования Python позволит в дальнейшем модифицировать программную часть реализованной модели с минимальными трудозатратами.

Поставленная цель работы была достигнута: универсальная модель гарантированно стойкого шифрования методом гаммирования была успешно разработана и реализована

Указанная модель может быть реализована в различных системах для шифрования текстовой информации при отправке конфиденциальной информации по незащищённым каналам с устройств, использующие различные ОС. Полученные в работе результаты исследований предполагают проведение дальнейших исследований по развитию и совершенствованию универсальной модели гарантированно стойкого шифрования благодаря использованию различных языков программирования, расширению ее практической применимости; соответствующие предложения по созданию универсального криптостойкого системного ГПСЧ для существующих и перспективных ОС предполагают их дальнейшую реализацию и применение в различных системах.

Выражаю благодарность своему научному руководителю – Троицкому Игорю Ивановичу, доценту, кандидату технических наук, доценту кафедры ИУ-8 МГТУ им. Баумана.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Информатика. 11 класс / Урок 18. Информационное право и информационная безопасность // Информатика - Российская электронная школа – URL: https://resh.edu.ru/subject/lesson/6472/additional/166806/ (дата обращения 26.02.2026).
2. Мельников В.П., Клейменов С.А., Петраков А.М. Информационная безопасность и защита информации / Под ред. С.А.Клейменова. - М.: Издательский центр «Академия», 2012. – 336 с.
3. Анисимов В.А. Шифры гаммирования. Лекции // Дальневосточный государственный университет путей сообщения. - URL: https://sites.google.com/site/anisimovkhv/learning/kripto/lecture/tema6 (дата обращения 26.02.2026).
4. Ярочкин В.И. Информационная безопасность: Учебник для студентов вузов. - М.: Академический Проект: Трикста, 2004. – 544 с.
5. Stallings W. Cryptography and network security. – Pearson Education, publishing as Prentice Hall, 2016. – 754 p.
6. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. – М.: Диалектика, 2022. – 1040 с.
7. Слеповичев И.И. Генераторы псевдослучайных чисел. – Саратов: СГУ, 2017. – 118 с.
8. Зензин О.С., Иванов М.А. Стандарт криптографической защиты. - AES. Конечные поля / Под ред. М.А.Иванова – М.: КУДИЦ-ОБРАЗ, 2002. – 176 с.
9. Кучерик А.О. Защита информации. Курс лекций. – Владимир: Изд-во ВлГУ, 2017. – 104 с.
10. NIST SP 800-90C. Recommendation for Random Bit Generator (RBG) Constructions. – 146 p.
11. RSA-4096 vs ECC-256 / Popular Encryption Standards Comparison. – URL: https://mojoauth.com/compare-encryption-algorithms/rsa-4096-vs-ecc-256/#key-generation-and-management (обращения 26.02.2026).
12. Жук И.В. Шифрование методом гаммирования // XXVIII научно-образовательное соревнование молодых исследователей «Шаг в будущее». Сборник лучших работ конференции «Шаг в будущее». - М.: МГТУ им. Н.Э. Баумана, 2025. - С. 158-187.
13. Python: Generate pseudo-random numbers. - URL: https://docs.python.org/3/library/random.html (обращения 26.02.2026).
14. RFC 8017 - PKCS #1: RSA Cryptography Specifications Version 2.2 – November, 2016. - URL: https://datatracker.ietf.org/doc/html/rfc8017 (дата обращения 26.02.2026).
15. ГОСТ Р 56205-2014. Сети коммуникационные промышленные. Защищённость (кибербезопасность) сети и системы. Часть 1-1. Терминология, концептуальные положения и модели. (утв. и введён в действие Приказом Росстандарта от 10.11.2014 N 1493-ст). – 73 с.
16. Stephan M. Documentation and Analysis of the Linux Random Number Generator Version: 5.11 – Federal Office for Information Security, 2024. – 116 p.
17. Linux kernel source tree // Актуальный код 10. Linux-ГСЧ. - URL: https://web.git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/drivers/char/random.c?id=4df86c6ea5c37fe0452638f39a1e4b189da75c54 (дата обращения 26.02.2026).

18. Chapter 16. Security // Free BSD. - URL: https://docs.freebsd.org/en/books/handbook/security/index.html (дата обращения 26.02.2026).
19. RFC 7693 - The Blake2 Cryptographic Hash and Message Aunthentication Code (Mac). - November 2015. – 30 p. - URL: https://2rfc.net/7693 (дата обращения 26.02.2026).
20. Aumasson Jean-Philippe. Serious cryptography. A practical introduction to modern encryption / Foreword by Mathew D.Green. – San Francisco: No Strach Press, 2018. – 434 p.
21. Bernstein J. ChaCha, a variant of Salsa20 // Department of Mathematics, Statistics, and Computer Science (M/C 249). The University of Illinois at Chicago - Chicago, IL 60607–7045. – 6 p.
22. Random number generator. - URL: https://man.openbsd.org/arc4random.3 (дата обращения 09.03.2025).
23. NIST SP 800-90A Rev. 1. Recommendation for Random Number Generation Using Deterministic Random Bit Generators. – URL: https://csrc.nist.gov/pubs/sp/800/90/a/r1/final (дата обращения 09.03.2025).
24. Secure Enclave // Безопасность платформы Apple. - URL: https://support.apple.com/guide/security/secure-enclave-sec59b0b31ff/web (дата обращения 26.02.2026).
25. Barker Elaine, Kelsey John. Recommendation for Random Number Generation Using Deterministic Random Bit Generators / Nist special Publication 800-90A, Revision 1 // National Institute of Standards and Technology. – U.S. Department of Commerce, 2015. – 128 p.
26. Random: Avoid magic numbers // Free BSD. - URL: https://github.com/freebsd/freebsd-src/blob/main/sys/dev/random/fortuna.c (дата обращения 26.02.2026).
27. Актуальный код реализации алгоритма ChaCha20. - URL: https://web.git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/include/crypto/chacha.h?id=4df86c6ea5c37fe0452638f39a1e4b189da75c54 (дата обращения 26.02.2026).
28. BCryptGenRandom function (bcrypt.h). - URL: https://learn.microsoft.com/en-us/windows/win32/api/bcrypt/nf-bcrypt-bcryptgenrandom (дата обращения 26.02.2026).
29. Bernstein Daniel J. Curve25519: New Diffie-Hellman Speed Records // Public Key Cryptography - PKC 2006. – P. 207-228. – URL: https://link.springer.com/chapter/10.1007/11745853_14 (дата обращения 26.02.2026).
30. RFC 7748 : Elliptic Curves for Security – URL: https://www.rfc-editor.org/rfc/rfc7748# (дата обращения 26.02.2026).
31. NIST SP 800-56A Rev. 3 Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography. – URL: https://csrc.nist.gov/pubs/sp/800/56/a/r3/final (дата обращения 26.02.2026).
32. Наурас Х. Сабри. Цифровые подписи в ограниченных средах: Сравнительный анализ безопасности RSA, ECDSA и EdDSA. – URL: https://doi.org/10.24108/preprints-3113617 (дата обращения 26.02.2026).
33. Наурас Хуссейн Сабри. Методы снижения вычислительной сложности цифровой подписи для устройств с ограниченными ресурсами // Диссертация на соискание учёной степени кандидата технических наук. – Санкт-Петербург, 2025.

34. RFC 8439. – ChaCha20 and Poly1305 for IETF Protocols - URL: https://datatracker.ietf.org/doc/rfc8439/ (дата обращения 26.02.2026).
35. Barker Elaine, Chen Lily, Roginsky Allen, Vassilev Apostol, Davis Richard, Simon Scott. Recommendation for Pair-Wise Key Establishment Using Integer Factorization Cryptography / NIST Special Publication 800-56B Revision 2 // National Institute of Standards and Technology. – U.S. Department of Commerce, 2019. – 121 p.
36. Barker Elaine, Kelsey John. Digital Signature Standard (DSS) / Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899-8900 FIPS 186-5 // National Institute of Standards and Technology. – U.S. Department of Commerce; 2023. – 77 p.
37. NIST SP 800-22 Revision 1a. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. – 131 p.
38. Heninger Nadia, Durumeric Zakir, Wustrow Eric, Halderman J. Alex. Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices // 21-st USENIX Security Symposium. - URL: https://www.usenix.org/conference/usenixsecurity12/technical-sessions/presentation/heninger (дата обращения 26.02.2026).
39. BLAKE3. - URL: https://github.com/BLAKE3-team/BLAKE3#faq (дата обращения 26.02.2026).

ПРИЛОЖЕНИЕ А. МЕТОДИКА ШИФРОВАНИЯ/РАСШИФРОВЫВАНИЯМЕТОДОМ ГАММИРОВАНИЯ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА PYTHON

1. Представление открытого текста в виде файла в формате .txt.
2. Реализация функции «ГЕНЕРАЦИЯ ГАММЫ» (Работа программы 0: генерация гаммы).
   2.1. Вызывается функция «generating» в скрипте, подав ей на вход путь файла формата .txt.
   2.2. Получив на входе путь текстового файла, функция «generating» открывает файл в режиме чтения и считывает в одну строку весь текст, написанный в файле.
   2.3. С помощью встроенной функции «len» определяется длина открытого текста в файле, которая записывается как переменная K, после чего файл закрывается.
   2.4. Далее идёт сама генерация ГАММЫ, которая представляет из себя случайную последовательность единиц и нулей от генератора случайных чисел.

   Для создания датчика случайных чисел на языке программирования Python используется модуль random, который предоставляет функции для генерации случайных чисел, букв, случайного выбора элементов последовательности, и метод randint, который случайно выбирает целое число из диапазона [1; 2^24]. Для большой стойкости гаммы осуществляется выбор числа N тоже рандомно. Число N является случайным числом из диапазона [K + 5*K; 100*K]. Обращение к датчику случайных чисел происходит с помощью цикла for i in range(N). Каждый сгенерированное целое число переводится в двоичную систему счисления с помощью встроенной функции «bin» и добавляется в строку gamma.

   2.5 Далее список gamma записывается в программу 1 и программу 2 для дальнейшего сложения методом XOR с открытым и шифрованным текстом.
3. Реализация функции «ШИФРОВАНИЕ» (Работа программы 1: шифрование).
   3.1. Вызывается функция «coding» в скрипте, подав ей на вход путь файла формата .txt и гамму.
   3.2. Получив на вход путь текстового файла и гамму, функция «coding» открывает его в режиме чтения и считывает в одну строку весь текст, написанный в файле.
   3.3. С помощью цикла for каждый символ строки перебирается, после чего он переводится в целое число по кодировочной таблице UTF-8 с помощью встроенной функции «ord». Затем это целое число представляется в двоичном виде с помощью встроенной функции «bin».

3.4. Число, представленное в двоичном виде, добавляется в список lst. Таким образом в списке получим двоичную запись каждого символа текста.

3.5. Записанная в файл программы 1 в виде списка гамма, состоящая из блоков записей целых чисел в двоичном виде теперь, проверяется на то, чтобы каждый n-ный блок был равен длине двоичной записи n-ного символа открытого текста.

Для этого каждый **n**-ный блок гаммы до числа **K**, где **K** – число символов открытого текста, перебирается **циклом «for i in range(len(lst))»** и сравнивается с длиной двоичной записи **n**-ного символа открытого текста. В случае, если **n**-ный блок гаммы длиннее двоичной записи **n**-ного символа открытого текста, **n**-ный символ текста удлиняется с начала на **«0»*N**, где **N** - разница между длиной **n**-ного блока гаммы и **n**-ного символа открытого текста. В случае, если **n**-ный блок гаммы короче двоичной записи **n**-ного символа открытого текста, **n**-ный блок гаммы удлиняется с начала на **«0»*N**, где **N** - разница между длиной двоичной записи **n**-ного символа открытого текста и длины **n**-ного блока гаммы.

3.6. Для корректного сложения методом XOR список lst удлиняется на строку, состоящей из (начальная двоичная запись текста – длина гаммы) нулей, после чего эта строка разбивается на блоки, длина n-ого из которых будет равна длине (len(lst)+n)-ого блока гаммы.

3.7. Сложение методом XOR (сложение по модулю два) гаммы и открытого текста с помощью цикла «for i in range(len(lst))».

Каждый n-ый блок списка lst складывается с n-ым блоком гаммы методом XOR, после чего полученная двоичная запись n-ого символа шифрованного текста добавляется в список lst2, для его дальнейшего перевода в символьный вид.

3.8. Каждый элемент списка lst2, который является двоичной записью n-ного символа шифрованного текста, с помощью цикла «for i in range(len(lst2))» переводится в целое число с помощью встроенной функции «int». После чего данное целое число представляется в виде символа зашифрованного текста с помощью встроенной функции «chr» и добавляется в строку str_.

Вывод: Получившаяся строка str_ является шифрованного текста. Для записи в файл полученного шифрованного текста в программе 1 создается новый текстовый файл, в который записывается получившаяся строка str_.

4. Реализация функции «РАСШИФРОВАНИЕ» (Работа программы 2: расшифрование).

4.1. Для расшифрования файла в скрипте программы 2 необходимо вызвать функцию «decoding», подав ей на вход название файла в формате .txt с зашифрованным

текстом и гамму.

4.2. Получив на вход название файла и гамму, функция «decoding» открывает файл в режиме чтения и считывает весь шифрованный текст, написанный в файле, в одну строку.

4.3. С помощью цикла «for» каждый символ строки переводится в целое число по кодировочной таблице UTF-8 с помощью встроенной функции «ord», после чего это целое число представляется в двоичном виде с помощью встроенной функции «bin» и добавляется в список lst.

4.4. Записанная в файл программы 2 в виде списка гамма, состоящая из блоков записей целых чисел в двоичном виде, теперь проверяется на то, чтобы каждый n-ный блок был равен длине двоичной записи n-ного символа шифрованного текста.

Для этого каждый n-ный блок гаммы, где n<= длинны списка lst перебирается циклом «for i in range(len(lst))» и сравнивается с длиной двоичной записи n-ного символа шифрованного текста. В случае, если n-ный блок гаммы длиннее двоичной записи n-ного символа открытого текста, n-ный символ шифрованного текста удлиняется с начала на «0»*N, где N - разница между длиной двоичной записи n-ного блока гаммы текста и длины n-ного символа шифрованного текста. В случае, если n-ный блок гаммы короче двоичной записи n-ного символа шифрованного текста, n-ный блок гаммы удлиняется с начала на «0»*N, где N - разница между длиной двоичной записи n-ного символа открытого текста и длины n-ного блока гаммы.

4.5. Сложение методом XOR (сложение по модулю два) гаммы и шифрованного текста с помощью цикла «for i in range(len(lst))».

Каждый n-ый блок списка lst складывается с n-ым блоком гаммы методом XOR, после чего полученная двоичная запись n-ого символа расшифрованного текста добавляется в список lst2, для его дальнейшего перевода в символьный вид.

4.7. Каждый элемент списка lst2, который является двоичной записью n-ного символа расшифрованного текста, с помощью цикла «for i in range(len(lst2))» переводится в целое число с помощью функции «int». После чего это целое число представляется в виде символа по кодировочной таблице UTF-8 расшифрованного текста с помощью встроенной функции «chr» и добавляется в строку str2_.

Вывод: Получившаяся строка str2_ является расшифрованным текстом. Для записи полученного расшифрованного текста в файл программа 2 создаёт

конечный текстовый файл и записывает в него получившийся расшифрованный текст (строка str2_).

ПРИЛОЖЕНИЕ Б. МЕТОДИКА ПЕРЕДАЧИ КЛЮЧА ДЛЯ СИММЕТРИЧНОГО ШИФРОВАНИЯ МЕТОДОМ ГАММИРОВАНИЯ

Методика разработана в соответствии со стандартами NIST SP 800-56 B rev.2 и FIPS 186-5, где допустимая пара ключей RSA-4096 в своей базовой форме состоит из открытого ключа (n, e) и закрытого ключа (n, d) [35, 36].

1. Генерация пары приватных (n1, d1), (n2, d2) и публичных ключей (n1, e1), (n2, e2).
2. Вычисление хэша публичных пар с помощью одобренных хэш-функций (SHA-256, SHA-512 и др.).
3. Передача дайджеста хэш-функций по нескольким каналам.

   В случае, если получатели публичного ключа обнаруживают несовпадение хэшей, передача хэшей по каналам прерывается и производится повторно по новым каналам до тех пор, пока хэши во всех каналах не совпадут для публичного ключа (n1, e1) и публичного ключа (n2, e2) соответственно.
4. Получатель ключа для симметричного шифрования пересылает свой публичный ключ (n1, e1) отправителю, меняя каналы до тех пор, пока ключ не будет соответствовать хэшу отправителя ключа для симметричного шифрования.
5. Отправитель пересылает свой публичный ключ (n2, e2) получателю, меняя каналы до тех пор, пока ключ не будет соответствовать хэшу получателя ключа для симметричного шифрования.
6. Отправитель шифрует ключ для симметричного шифрования публичным ключом (n1, e1).
7. Отправитель подписывает своё сообщение с помощью своего приватного ключа (n2, d2).
8. Отправитель отправляет шифрованное сообщение.
9. Получатель проверяет подпись отправителя ключа для симметричного шифрования с помощью публичного ключа (n2, e2) и расшифровывает сообщение с помощью своего приватного ключа (n1, d1).

ПРИЛОЖЕНИЕ В. МЕТОДИКА ПОЛУЧЕНИЯ ОБЩЕГО ИНИЦИАЛИЗИРУЮЩЕГО ЗНАЧЕНИЯ ДЛЯ ГПСЧ (КЛЮЧА С)

Пусть абонент **A** – отправитель, абонент **B** – получатель. Уравнением кривой Curve25519 будет:

$$y^2 \equiv x^3 + 486662x^2 + x \pmod{2^{255} - 19} \quad (1)$$

1. Отправитель и получатель генерируют себе числа k_a и k_b длиною 256 бит соответственно с помощью криптографически стойкого ГСЧ. После генерации чисел они «маскируются» таким образом, чтобы каждый приватный ключ входил в диапазон [2^254, 2^255]. Маскированные числа и есть приватные ключи d_a и d_b для отправителя и получателя соответственно.

2.1. Отправитель формирует свой публичный ключ скалярно умножая d_a на точку G с помощью алгоритма «лестница Монтгомери», позволяющего находить X-координаты новых точек на эллиптической кривой при скалярном умножении не используя Y-координаты. G - точка на эллиптической кривой с координатой X, равной 9. Результат скалярного умножения с помощью «лестницы Монтгомери» и есть публичный ключ отправителя. Будем называть публичным ключом отправителя параметр

$$P_a=d_a\times G \quad (2)$$

2.2. Аналогичным образом получает свой публичный ключ получатель сообщения (в скалярном умножении вместо d_a используется d_b). Будем называть публичным ключом получателя параметр

$$P_b= d_b\times G \quad (3)$$

3. Происходит обмен публичными ключами P_a и P_b. В соответствии со стандартом NIST SP 800-56A rev.3 [30] необходимо провести проверку публичных ключей (P_b и P_a) на стороне отправителя и получателя. Они проверяют их с помощью проверки цифровых подписей на базе эллиптической кривой.

4.1. После обмена публичными ключами отправитель умножает скалярно d_a (приватный ключ отправителя) на публичный ключ P_b, используя алгоритм «лестница Монтгомери». Найденная X-координата точки в результате скалярного умножения - и есть секрет

$$C1'=d_a\times P_b = d_a\times(d_b\times G) \quad (4)$$

4.2. После обмена публичными ключами получатель умножает скалярно d_b (приватный ключ получателя) на публичный ключ P_a, используя алгоритм «лестница Монтгомери». Найденная X-координата точки в результате скалярного умножения - и есть секрет

$$C2'=d_b\times P_b= d_b\times(d_a\times G) \quad (5)$$

В соответствии с правилом ассоциативности для операций на эллиптических кривых отправитель и получатель получают общий секрет

$$C'=C1'=C2'=d_a\times(d_b\times G)=d_b\times(d_a\times G)=(d_a\times d_b)\times G \quad (6)$$

5. Для получения общего ключа C из C' требуется использование KDF (Key Derivation Function). Одной из одобренных KDF является HKDF-SHA256(salt, K_x, info, len_bytes). В качестве K_x необходимо использование C', len_bytes=32. Salt и info можно использовать пустыми, однако для повышения безопасности рекомендуется использовать контекстную информацию. В реализованной модели в качестве неё используется дата общения абонентов.

Использование HKDF-SHA256 в модели реализуется с помощью возможностей библиотеки cryptography: C = HKDF(algorithm=hashes.SHA256(), length=32, salt=None,info=datetime.now(zoneinfo.ZoneInfo("Asia/Kamchatka")).strftime("%Y-%m-%d-%H").encode(), backend=default_backend()).derive(C_prime).

Ключевая пара для цифровой подписи представляет из себя приватный ключ – маскированные с помощью функции cryptography Ed25519PrivateKey.from_private_bytes(my_seed) 32 байта криптостойкого ГСЧ, а также публичный ключ, полученный с помощью функции private_key.public_key() из приватного ключа. Подписание сообщений реализуется с использованием функций библиотеки cryptography следующим образом: signature_bytes = private_key.sign(message.encode('ascii'))

signature_b64 = base64.b64encode(signature_bytes).decode('ascii'), где message – отправляемое сообщение. Верификация подписи происходит также с использованием функций библиотеки cryptography следующим образом: private_key.verify(signature, message.encode()).

ПРИЛОЖЕНИЕ Г.РЕАЛИЗАЦИЯ ПРОГРАММНЫХ МОДУЛЕЙ МОДЕЛИ ГАРАНТИРОВАННО СТОЙКОГО ШИФРОВАНИЯ МЕТОДОМ ГАММИРОВАНИЯ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON

(листинги программ)

абонент А подпись(генерация ключевой пары для себя и отправк...

```
import smtplib
import time
import imaplib
import email
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives import serialization
import os

# Конфигурация email
sender_email = "test1communication@gmail.com"
sender_password = "gmqd hfva juhb aoyg"
receiver_email = "test2unluckcommunication@gmail.com"

def send_gmail(sender_email, sender_password, receiver_email, s
    try:
```

абонент А(прием публичного ключа для подписи).py - D:\Мои док...

```
import smtplib
import time
import imaplib
import email
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives import serialization
import os

# Конфигурация email
sender_email = "test1communication@gmail.com"
sender_password = "gmqd hfva juhb aoyg"
receiver_email = "test2unluckcommunication@gmail.com"
def save_public_key(public_key, filename):
    """
    Сохраняет публичный ключ в файл (аналогично приватному)
```

абонент В подпись(генерация ключевой пары для себя и отправка...

```
import smtplib
import time
import imaplib
import email
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives import serialization
import os

# Конфигурация email
sender_email = "test2unluckcommunication@gmail.com"
sender_password = "yexi cfhf cqbc qtrx"
receiver_email = "test1communication@gmail.com"

def send_gmail(sender_email, sender_password, receiver_email, s
    try:
```

абонент В(прием публичного ключа для подписи).py - D:\Мои док...

```
import smtplib
import time
import imaplib
import email
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives import serialization
import os

# Конфигурация email
sender_email = "test2unluckcommunication@gmail.com"
sender_password = "yexi cfhf cqbc qtrx"
receiver_email = "test1communication@gmail.com"
def save_public_key(public_key, filename):
    """
    Сохраняет публичный ключ в файл (аналогично приватному)
```

Рис. Г1. Модуль 1: «Обмен публичными ключами для подписи»

optm абонент А генерация сесионной ключевой пары, отправка п...

```
import smtplib
import time
import imaplib
import email
import base64
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives import serialization
from cryptography.hazmat.primitives.asymmetric.x25519 import X2
import os

# Конфигурация email
sender_email = "test1communication@gmail.com"
sender_password = "gmqd hfva juhb aoyg"
receiver_email = "test2unluckcommunication@gmail.com"
```

optm абонент А(прием публичного ключа для сессии).py - D:\Мои...

```
import imaplib
import email
import base64
import time
from cryptography.hazmat.primitives import serialization
from cryptography.hazmat.primitives.asymmetric.x25519 import X2

# Конфигурация email (абонент А)
sender_email = "test1communication@gmail.com"
sender_password = "gmqd hfva juhb aoyg"
receiver_email = "test2unluckcommunication@gmail.com"

# Загружаем публичный ключ подписи абонента В (Ed25519) для прс
with open("keyB_public.pem", "rb") as f:
    verifying_key = serialization.load_pem_public_key(f.read())

def save_x25519_key(pem_bytes: bytes):
```

optm абонент В генерация сесионной ключевой пары, отправка п...

```
import smtplib
import time
import imaplib
import email
import base64
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives import serialization
from cryptography.hazmat.primitives.asymmetric.x25519 import X2
import os

# Конфигурация email
sender_email = "test2unluckcommunication@gmail.com"
sender_password = "yexi cfhf cqbc qtrx"
receiver_email = "test1communication@gmail.com"
```

optm абонент В(прием публичного ключа для сессии).py - D:\Мои...

```
import imaplib
import email
import base64
import time
from cryptography.hazmat.primitives import serialization
from cryptography.hazmat.primitives.asymmetric.x25519 import X2

# Конфигурация email
sender_email = "test2unluckcommunication@gmail.com"
sender_password = "yexi cfhf cqbc qtrx"
receiver_email = "test1communication@gmail.com"

# Загружаем публичный ключ подписи (Ed25519) - ключ В
with open("keyA_public.pem", "rb") as f:
    verifying_key = serialization.load_pem_public_key(f.read())

def save_x25519_key(pem_bytes):
```

Рис. Г2. Модуль 2: «Обмен сессионными ключами для подписи»

абонент A ses.py - D:\Мои документы\Илюша\Подготовка в МГТУ...

File Edit Format Run Options Window Help

```
import os
import smtplib
import imaplib
import base64
import time
import email
import os
import smtplib
import imaplib
import base64
import time
import email
import hashlib
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives.asymmetric.x25519 import X2
from cryptography.hazmat.primitives.serialization import load_p

# ---------- Конфигурация Gmail ----------
GMAIL_USER = "test1communication@gmail.com"
GMAIL_PASSWORD = "gmqd hfva juhb aoyg"
RECIPIENT = "test2unluckcommunication@gmail.com"
SENDER = GMAIL_USER

# ---------- Загрузка ключей Ed25519 (для подписи) ----------
with open("keyA_private.pem", "rb") as f:
    private_key_ed25519 = load_pem_private_key(f.read(), passwo

with open("keyB_public.pem", "rb") as f:
    public_key_b_ed25519 = load_pem_public_key(f.read())

# ---------- Загрузка ключей X25519 (для общего секрета ChaCha2
with open("skeyA_private.pem", "rb") as f:
    x25519_private_a = load_pem_private_key(f.read(), password=

with open("skeyB_public.pem", "rb") as f:
    x25519_public_b = load_pem_public_key(f.read())
```

Ln: 1 Col: 0

абонент B ses.py - D:\Мои документы\Илюша\Подготовка в МГТУ...

File Edit Format Run Options Window Help

```
import os
import smtplib
import imaplib
import base64
import time
import email
import os
import smtplib
import imaplib
import base64
import time
import email
import hashlib
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from cryptography.hazmat.primitives.asymmetric.ed25519 import E
from cryptography.hazmat.primitives.asymmetric.x25519 import X2
from cryptography.hazmat.primitives.serialization import load_p

# ---------- Конфигурация Gmail ----------
GMAIL_USER = "test2unluckcommunication@gmail.com"
GMAIL_PASSWORD = "yexi cfhf cqbc qtrx"
RECIPIENT = "test1communication@gmail.com"
SENDER = GMAIL_USER

# ---------- Загрузка ключей Ed25519 (для подписи) ----------
with open("keyB_private.pem", "rb") as f:
    private_key_ed25519 = load_pem_private_key(f.read(), passwo

with open("keyA_public.pem", "rb") as f:
    public_key_b_ed25519 = load_pem_public_key(f.read())

# ---------- Загрузка ключей X25519 (для общего секрета ChaCha2
with open("skeyB_private.pem", "rb") as f:
    x25519_private_a = load_pem_private_key(f.read(), password=

with open("skeyA_public.pem", "rb") as f:
    x25519_public_b = load_pem_public_key(f.read())
```

Ln: 1 Col: 0

Рис. Г3. Модуль 3: «Абоненты»

ПРИЛОЖЕНИЕ Д. РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ ГСЧ

Название теста	Минимальное реальное значение P-valueT	Минимальное допустимое значение P-valueT	Минимальная реальная доля прошедших тест последова-тельностей (Proportion)	Минимальная допустимая доля прошедших тест последова-тельностей	Мин. реальные значения P-valueT и долей больше или равны мин. допустимым значениям
Частотный (монобитовый) тест	0.071177	0.0001	100/100	96/100	да
Частотный блочный тест	0.699313	0.0001	100/100	96/100	да
Тест на последовательность одинаковых битов	0.867692	0.0001	99/100	96/100	да
Тест на самую длинную последовательность единиц в блоке	0.171867	0.0001	100/100	96/100	да
Тест рангов бинарных матриц	0.304126	0.0001	99/100	96/100	да
Спектральный тест	0.494392	0.0001	99/100	96/100	да
Тест на совпадение с неперекрывающимися шаблонами	0.008879	0.0001	96/100	96/100	да
Тест на совпадение с перекрывающимися шаблонами	0.978072	0.0001	99/100	96/100	да
Универсальный статистический тест Маурера	0.678686	0.0001	100/100	96/100	да
Тест на линейную сложность	0.867692	0.0001	99/100	96/100	да
Тест на периодичность	0.319084	0.0001	99/100	96/100	да
Тест приблизительной энтропии	0.455937	0.0001	99/100	96/100	
Тест кумулятивных сумм	0.224821	0.0001	100/100	96/100	да
Тест случайных блужданий	0.063482	0.0001	60/61	58/61	да
Тест вариантов случайных блужданий	0.051391	0.0001	59/61	58/61	да

Рис. Д1. Тестирование системного ГСЧ Windows

Название теста	Минимальное реальное значение P-valueT	Минимальное допустимое значение P-valueT	Минимальная реальная доля прошедших тест последова-тельностей (Proportion)	Минимальная допустимая доля прошедших тест последова-тельностей	Мин. реальные значения P-valueT и долей больше или равны мин. допустимым значениям
Частотный (монобитовый) тест	0.249284	0.0001	100/100	96/100	да
Частотный блочный тест	0.437274	0.0001	98/100	96/100	да
Тест на последовательность одинаковых битов	0.534146	0.0001	100/100	96/100	да
Тест на самую длинную последовательность единиц в блоке	0.739918	0.0001	99/100	96/100	да
Тест рангов бинарных матриц	0.867692	0.0001	99/100	96/100	да
Спектральный тест	0.401199	0.0001	100/100	96/100	да
Тест на совпадение с неперекрывающимися шаблонами	0.006196	0.0001	96/100	96/100	да
Тест на совпадение с перекрывающимися шаблонами	0.514124	0.0001	100/100	96/100	да
Универсальный статистический тест Маурера	0.383827	0.0001	98/100	96/100	да
Тест на линейную сложность	0.350485	0.0001	99/100	96/100	да
Тест на периодичность	0.236810	0.0001	99/100	96/100	да
Тест приблизительной энтропии	0.779188	0.0001	98/100	96/100	
Тест кумулятивных сумм	0.275709	0.0001	99/100	96/100	да
Тест случайных блужданий	0.082177	0.0001	59/60	57/60	да
Тест вариантов случайных блужданий	0.134686	0.0001	59/60	57/60	да

Рис. Д2. Тестирование ГПСЧ ChaCha-20

JARVIS 1 – ПЕРСОНАЛЬНЫЙ АССИСТЕНТ ОБЩЕГО НАЗНАЧЕНИЯ

Несмеянов Александр Витальевич
Москва г, ГБОУ Школа 1900, 11 класс

Секция: Информационные технологии

Научный руководитель: Смирнова Светлана Юрьевна, преподаватель ГБОУ Инженерная школа №1581

АННОТАЦИЯ

JARVIS 1 (Just A Rather Very Intelligent System) представляет собой исследовательскую и инженерную платформу персонального ассистента общего назначения. В отличие от классических голосовых помощников, ориентированных преимущественно на короткую команду и одноразовый ответ, система построена как многослойный контур: она принимает естественную речь, распознает русский и английский языки с поддержкой code-switching, маршрутизирует намерения, выполняет действия в операционной системе, извлекает знания из документов, сохраняет долговременный контекст и синтезирует голосовой ответ.

Центральная идея проекта заключается в разделении ассистента на специализированные подсистемы. Голосовой слой отвечает за взаимодействие в реальном времени, оркестратор - за классификацию намерений и экономию вычислений, Assistant Gateway - за безопасное выполнение действий, NRTDS RAG -за устойчивый поиск по документам, а когнитивная память - за жизненный цикл знаний от кратковременного эпизода до долговременной структуры. Такой подход делает проект не набором разрозненных моделей, а цельной AI-системой с управляемой архитектурой.

Ключевой результат: в рамках проекта сформирована модульная платформа из девяти основных компонентов и более чем двадцати интегрированных технологий. Система сочетает приватность локального выполнения, билингвальность RU/EN, гибридный RAG, долговременную память, voice orchestrator и gateway-действия в ОС.

ВВЕДЕНИЕ

Проект JARVIS 1 возник из практической проблемы: большинство доступных голосовых ассистентов либо завязаны на облачные сервисы, либо ограничены короткой сессией, либо не умеют естественно работать с несколькими языками внутри одной фразы. Для пользователя это означает зависимость от внешней инфраструктуры, передачу личных данных сторонним платформам и отсутствие полноценной долговременной памяти.

В исследовании была принята более широкая постановка. Ассистент должен не просто отвечать на реплики, а выступать как персональный вычислительный посредник между человеком и компьютером. Поэтому в проекте объединены обработка речи, синтез голоса, маршрутизация намерений, действия в ОС, документный поиск, память, безопасность и наблюдаемость.

Актуальность проекта определяется тремя направлениями:

1. Приватность. Пользовательские данные остаются в локальном контуре или контролируемой инфраструктуре.

2. Персонализация. Ассистент сохраняет историю, строит связи между темами и использует долговременный контекст. Третье

3. Универсальность. Один интерфейс объединяет диалог, документы, приложения и системные команды.

Таблица 1.

Проблема	Ограничение типичных решений	Подход JARVIS 1
Облачная зависимость	Данные уходят на серверы поставщика, требуется постоянный интернет	Локально ориентированная архитектура с контролируемыми внешними вызовами
Короткая память	Диалог ограничивается текущей сессией	Двухуровневая память Hippocampus -> Long-term
Монолингвальность	Нужно вручную выбирать язык или использовать отдельные режимы	Билингвальный ASR/TTS и code-switching RU/EN
Слабый поиск	Dense-only retrieval теряет ключевые слова и точные совпадения	Hybrid RAG: Dense + SPLADE + ColBERT + RRF

ЦЕЛИ ЗАДАЧИ И ИНЖЕНЕРНАЯ ГИПОТЕЗА

Основная цель работы - создание полнофункционального персонального ассистента, способного понимать естественную речь на русском и английском языках, выполнять системные действия, поддерживать долговременную память, извлекать знания из документов и работать в приватном локальном контуре.

Инженерная гипотеза проекта формулируется следующим образом: если разделить ассистента на специализированные слои, каждый из которых решает свой класс задач, то можно одновременно повысить качество ответа, снизить вычислительную нагрузку, обеспечить расширяемость и сохранить контролируемость всей системы.

- Научная задача ASR: проверить, может ли единый WhisperX/Faster-Whisper контур заменить каскад из language identification и отдельных моделей распознавания.
- Научная задача памяти: построить жизненный цикл знаний по модели Hipposampus > Long-term, где свежие эпизоды постепенно консолидируются в устойчивую долговременную структуру.
- Научная задача retrieval: сравнить dense-only подход с гибридным поиском, объединяющим семантический, sparse и token-level сигнал.
- Инженерная задача gateway: создать слой безопасных действий в ОС с валидацией путей, blacklist/whitelist политиками, fallback-режимом и диагностикой состояния.
- Инженерная задача производительности: проверить качество и применить INT8 квантизацию, потоковую обработку, очереди и маршрутизацию намерений для уменьшения задержки и нагрузки.

Методологический принцип: в проекте не выбиралась одна универсальная модель для всего. Напротив, система построена как ансамбль специализированных контуров: быстрый оркестратор классифицирует намерение, основная LLM отвечает за сложное рассуждение, RAG ищет документы, memory layer хранит персональный контекст, а gateway выполняет действия.

ОБЩАЯ АРХИТЕКТУРА JARVIS 1

Архитектура JARVIS 1 представляет собой модульную систему с девятью основными компонентами. Каждый компонент изолирует отдельный тип сложности: аудио, речь, намерения, инструменты, документы, память, безопасность, интерфейсы и мониторинг. Такая декомпозиция позволяет заменять модели и механизмы без полной перестройки проекта.

Системная карта end-to-end

Пользователь

-> микрофон/текстовый ввод

-> VAD + Wake Word + Audio Buffer

-> WhisperX Bilingual ASR

-> Recognition Result

-> Voice Orchestrator

-> простая команда: Assistant Gateway

-> действие в ОС

-> документный запрос: NRTDS RAG

-> найденный контекст

-> персональный контекст: Memory Read Pipeline

-> релевантные воспоминания

-> сложный диалог: Main LLM

-> Streaming ответ

-> Chatterbox TTS

-> Unified RVC

-> Post-FX

-> Динамик

Таблица 2.

Слой	Назначение	Ключевые технологии
Voice Input	Прием, фильтрация и сегментация аудио	SoundDevice, Silero VAD, openWakeWord
ASR	Распознавание RU/EN и смешанных фраз	WhisperX/Faster-Whisper Large V3, INT8
Routing	Классификация намерений до основной LLM	IBM Granite 4H-micro, intent policy

Gateway	Безопасное выполнение команд	Everything SDK, filesystem handler, app DB
RAG	Поиск по документам и знаниям	Jina-v3, SPLADE, ColBERT, RRF, ChromaDB
Memory	Долговременный персональный контекст	Hippocampus, MVNF, Hebbian links, decay
Voice Output	Синтез и обработка голоса	Chatterbox TTS, Unified RVC, EQ, Compressor, Reverb
Interfaces	Операционная поверхность	CLI, TUI, GUI, /app, /rag, API
Observability	Диагностика и устойчивость	Состояние, метрики, fallback, комплекс тестов

ПРИНЦИП ПОТОКОВОЙ ОБРАБОТКИ

Система спроектирована как набор независимых компонентов. SpeechListener принимает и подготавливает речь, LanguageModelProcessor обрабатывает распознанный текст, TextToSpeechSynthesizer создает аудиоответ, SpeechPlayer воспроизводит результат. Между ними используются очереди, что снижает риск блокировки всей системы одной долгой операцией.

Такой подход особенно важен для голосового ассистента: пользователь воспринимает задержку не как техническую метрику, а как качество диалога. Поэтому обработка делится на этапы, которые могут работать параллельно: пока LLM генерирует следующий фрагмент, TTS уже может синтезировать предыдущий, а player - воспроизводить готовое аудио.

REAL-TIME ГОЛОСОВОЙ КОНТУР

Голосовой контур решает задачу непрерывного преобразования физического аудиосигнала в структурированный запрос к ассистенту. Он должен отличать речь от шума, не реагировать на случайные звуки, быстро начинать запись после активации, корректно завершать фразу и передавать ASR именно тот фрагмент, который содержит смысловую команду.

Входной поток строится на сочетании Silero VAD и openWakeWord. VAD отвечает за определение голосовой активности, а wake word слой отделяет намеренное обращение к ассистенту от фоновой речи. После активации используется окно прослушивания и pre-roll

буфер, чтобы не потерять начало команды.

Таблица 3.

Этап	Функция	Инженерный смысл
Захват аудио	16 kHz mono PCM, короткие фреймы	Минимальная задержка и совместимость с речевыми моделями
Silero VAD	Определение голосовой активности	Снижение нагрузки на ASR и защита от шума
openWakeWord	Нейронная активационная фраза	Отделение команды от фоновой речи
Pre-roll buffer	Сохранение фрагмента до начала записи	Команда не обрезается на первом слове
Pause detector	Фиксация конца реплики	Передача в ASR законченного речевого сегмента
Echo guard	Защита от собственного TTS	Система не должна слушать сама себя

Результат: входной слой превращает непрерывный шумный аудиопоток в короткие смысловые сегменты, пригодные для распознавания. Это уменьшает задержку, экономит VRAM и снижает вероятность ложных команд.

WHISPER X БИЛИНГВАЛЬНЫЙ ASR И CODE-SWITCHING

Ключевым переходом в проекте стала замена раннего каскадного входного контура на единый WhisperX/Faster-Whisper ASR. Ранний подход с отдельным определением языка и разными ASR-ветками давал больше точек отказа: ошибка LID могла отправить фразу не в ту модель, а смешанная RU/EN речь требовала искусственного разбиения.

Целевой контур использует модель Large V3 в INT8-режиме. Это позволило сократить вычислительную стоимость без существенной потери качества. В презентационных результатах зафиксирована оптимизация ASR: INT8 квантизация дала ускорение порядка 3.2x, а качество распознавания было доведено до WER 3.2% для русского и 2.1% для английского на целевом наборе проверок.

Поддержка code-switching является одним из центральных научных результатов.

Пользователь может произнести фразу вида: “Запусти Chrome и найди document about machine learning”. В такой реплике русский задает действие, английский может задавать имя приложения или объект поиска, а ассистент должен сохранить смысл фразы как единого намерения.

Таблица 4.

Критерий	Каскадный подход	WhisperX/Faster-Whisper контур
Количество ASR/LID компонентов	Несколько моделей и ручная маршрутизация	Единая модельная точка
Code-switching	Сложная поддержка, риск ошибки LID	Естественная обработка смешанных фраз
Задержка	Выше из-за последовательных этапов	Ниже за счет единого прохода
Сопровождение	Сложнее обновлять и отлаживать	Проще менять параметры и модель
Диагностика	Ошибка может появиться в любой ветке	Ясная точка наблюдения ASR

Логика bilingual ASR

Аудиосегмент

-> нормализация сигнала

-> WhisperX/Faster-Whisper Large V3

-> сегменты с timestamps

-> word-level timestamps

-> языковая разметка слов и фразы

-> единый Aggregated Transcript

-> Recognition Result для LLM/Orchestrator

Научный смысл: в проекте билингвальность рассматривается не как наличие двух языков в настройках, а как способность одного голосового контура понимать естественную смешанную речь без ручного переключения режима.

ВЫХОДНОЙ ГОЛОСОВОЙ КОНТУР: CHATTERBOX TTS, RVC, POST-FX

Выходной контур отвечает за обратное преобразование: текстовый ответ ассистента должен стать узнаваемой, естественной и технически стабильной речью. В ранних вариантах билингвального TTS логичным решением были отдельные RU/EN ветки и последующее смешивание аудио. Но такой подход создавал слышимые стыки, усложнял синхронизацию и плохо масштабировался.

В целевой версии используется Chatterbox TTS как единая билингвальная модель. Она позволяет синтезировать русские, английские и смешанные ответы в одном голосовом тракте. Далее аудио проходит через Unified RVC, который выполняет voice conversion, и через post-FX pipeline: эквализацию, компрессию и реверберацию.

Таблица 5.

Компонент	Назначение	Результат
Chatterbox TTS	Билингвальный синтез RU/EN	Единый голосовой выход без ручного выбора языка
Voice cloning	Использование аудио-промпта голоса	Персонализация тембра
Unified RVC	Language-agnostic voice conversion	Одна модель трансформации для обоих языков
EQ	Тональная коррекция	Голос становится читаемее и ближе к целевому характеру
Compressor	Выравнивание динамики	Стабильная громкость без резких перепадов
Reverb	Пространственная окраска	Более завершенное и узнаваемое звучание

Важным инженерным решением является вынесение эффектов в отдельный post-processing слой. Это позволяет менять акустический характер ассистента независимо от TTS-модели и RVC. В результате голосовая система становится управляемой: качество, тембр, громкость и выразительность корректируются не хаотично, а через отдельные параметры

тракта.

MULTI-SPEAKER И РАЗДЕЛЕНИЕ ИСТОЧНИКОВ

Многоспикерный контур предназначен для сценариев, в которых рядом с ассистентом говорит не один человек или в аудио присутствуют перекрывающиеся речевые источники. Обычный ASR в таком случае получает смешанный сигнал и может объединить реплики, потерять часть слов или неверно определить намерение.

Для решения этой задачи в архитектуру введен слой ReSepFormer/SepFormer source separation. Он работает не как простая diarization-разметка, а как реальное разделение звуковых источников. После разделения каждый поток может быть обработан отдельно: очищен, нормализован и направлен в ASR.

Таблица 6.

Стадия	Описание	Зачем нужна
Source separation	Выделение до двух речевых источников	Снижение ошибок при перекрывающейся речи
Post-processing	Wiener filtering, spectral gating, band-pass, LUFS	Уменьшение bleed и артефактов
Per-speaker ASR	Распознавание каждого источника отдельно	Сохранение структуры диалога
Speaker formatting	Форматирование реплик по говорящим	LLM получает не кашу слов, а структурированный контекст
Confidence filtering	Отбрасывание слабых источников	Снижение ложных срабатываний

VOICE ORCHESTRATOR И ПОЛИТИКА МАРШРУТИЗАЦИИ

Voice Orchestrator является одним из главных отличий JARVIS 1 от типичных ассистентов. В классической схеме любой запрос отправляется в основную LLM. Это удобно на уровне прототипа, но неэффективно: запуск приложения, открытие папки или поиск файла не требуют тяжелого рассуждения большой модели.

В JARVIS 1 перед основной LLM размещен легкий классификатор намерений на базе IBM Granite 4H-micro. Его задача - быстро определить, относится ли запрос к диалогу, приложению, файлу, папке, документному поиску, системному действию или ошибке. Если

запрос является действием, он маршрутизируется в Assistant Gateway. Если требуется рассуждение, запрос передается в main LLM.

Таблица 7.

Параметр	Значение / результат
Модель оркестратора	IBM Granite 4H-micro
Размер	около 4B параметров
Квантование	4-bit профиль
Ориентировочная задержка intent classification	200-500 мс
Эффект на простых командах	до 3.5x ускорения относительно all-to-main-LLM baseline
Экономия вычислений	около 71% разгрузки GPU по времени на серии простых команд

Policy-routing

Recognition Result

-> Voice Orchestrator

-> LAUNCH_APPLICATION

-> Gateway Application Handler

-> OPEN_FOLDER/LIST_DIRECTORY

-> Gateway FileSystem Handler

-> SEARCH_FILES/FIND_DOCUMENT

-> Search or RAG Handler

-> GENERAL_CHAT

-> Main LLM

-> LOW_CONFIDENCE

-> fallback to Main LLM

-> NEED_CONFIRMATION

-> ask user

-> continue

Главная научно-инженерная ценность оркестратора заключается не только в

ускорении. Он меняет саму модель ассистента: система становится не просто генератором текста, а диспетчером действий. Команда пользователя рассматривается как намерение, для которого выбирается оптимальный исполнитель.

ASSISTANT GATEWAY И СЛОЙ ДЕЙСТВИЙ

Assistant Gateway отвечает за практическую сторону ассистента: запуск программ, поиск файлов, навигацию по папкам, работу с документами, URL и диагностические команды. Это слой, который превращает языковое намерение в безопасное действие в операционной системе.

Gateway построен вокруг нескольких обработчиков: Application Handler, File System Handler, Search Handler, Document RAG Handler и служебные компоненты health/status. Важная особенность - отделение классификации намерения от исполнения. Оркестратор решает, что хочет пользователь, а Gateway решает, как это выполнить корректно и безопасно.

Таблица 8.

Подсистема Gateway	Функции	Значение для ассистента
Application Handler	Fuzzy matching, Everything search, LLM-assisted .exe finder, launch stats	Запуск приложений по естественному названию
File System Handler	Навигация, открытие папок, listing, file info	Работа с локальной файловой структурой
Search Handler	Поиск файлов, recent files, URL opening	Быстрый доступ к объектам пользователя
Document RAG Handler	Индексация и поиск по документам	Ответы на вопросы по локальным материалам
Security Layer	Whitelist/blacklist, sandbox validation, path validation	Контролируемое выполнение действий
Health Layer	online/degraded/disabled статусы	Прозрачная диагностика и graceful degradation

Для запуска приложений используется многоступенчатая стратегия. Сначала выполняется поиск в базе известных приложений с fuzzy matching. Если найдено несколько

кандидатов, включается уточняющий выбор. Если приложение не известно базе, используется поиск через Everything и LLM-assisted предположение имен исполняемых файлов. После успешного запуска статистика обновляется, что повышает качество последующих решений.

Файловая подсистема работает не как произвольный доступ ко всему диску, а как контролируемый слой операций. Валидация путей, черные списки временных и системных директорий, ограничения на расширения и безопасная обработка ошибок защищают ассистента от опасных или некорректных действий.

NRTDS V2: ГИБРИДНЫЙ RAG С МНОЖЕСТВЕННЫМИ ИНДЕКСАМИ

Retrieval-augmented generation в JARVIS 1 развивается не как простой поиск по embedding, а как гибридная retrieval-система NRTDS v2. Причина в том, что один dense-поиск хорошо ловит смысловую близость, но хуже работает с точными терминами, именами файлов, номерами договоров, редкими техническими словами и token-level совпадениями.

Поэтому в проекте используется комбинация трех поисковых сигналов. Dense Search на Jina-embeddings-v3 отвечает за семантическое сходство. SPLADE добавляет sparse keyword-based сигнал и помогает находить точные слова. ColBERT выполняет late interaction на уровне токенов и повышает точность на сложных запросах. Reciprocal Rank Fusion объединяет результаты в единый рейтинг.

Таблица 9.

Компонент	Тип сигнала	Что решает
Jina-embeddings-v3	Dense semantic search, 1024-dimensional vectors, до 8192 tokens	Понимание смысла и переформулировок
SPLADE	Sparse vectors, term importance weighting	Точные слова, редкие термины, keyword-запросы
ColBERT	Token-level late interaction	Тонкое совпадение между запросом и фрагментами
RRF	Fusion по рангам	Устойчивое объединение результатов разных retrieval-методов
ChromaDB / Vector DB	Хранение dense-представлений	Быстрый семантический поиск
Document DB	Метаданные, статусы, версии	Управляемость индекса и диагностика

Экспериментальный результат: по данным исследовательской серии, full hybrid retrieval превосходит dense-only baseline на 37%: MRR@10 = 0.89 против 0.65. При этом задержка поиска остается в практическом диапазоне 800-1200 мс.

NRTDS v2 retrieval flow

Пользовательский запрос

-> нормализация и определение типа запроса

-> Dense retriever

-> SPLADE sparse retriever

-> ColBERT token-level retriever

-> Reciprocal Rank Fusion

-> фильтры, дедупликация, highlighting

-> Top-K контекст

-> LLM ответ с учетом найденных документов

Поддержка форматов PDF, DOCX, PPTX, XLSX, TXT и Markdown делает RAG-слой практическим инструментом для работы с реальными документами. Важным является не только сам поиск, но и операционная поверхность: команды диагностики, повторной индексации, сравнения версий, проверки ошибок и обслуживания индекса.

КОГНИТИВНАЯ СИСТЕМА ПАМЯЬТ HIPPOCAMPUS -> LONG-TERM

Система памяти JARVIS 1 построена не как простая история сообщений и не как единая векторная база. Ее идея вдохновлена биологической моделью: свежий опыт сначала попадает в кратковременный слой Hippocampus, затем через процесс консолидации переносится в долговременную память, где знания организуются по темам, подкастерам и связанным файлам.

Такой подход решает главную проблему персональной памяти: если сохранять все подряд, база быстро превращается в склад дубликатов. Если агрессивно сливать записи, теряются детали. Поэтому в проекте используется MVNF - Merge vs New File – механизм, который решает, нужно ли объединить новое воспоминание с существующим или создать отдельный файл памяти.

Таблица 10.

Слой памяти	Назначение	Механизмы
Hippocampus	Кратковременная память свежих эпизодов	Локальный RAG, Local MVNF, первичная классификация
Secretary	Нормализация и структурирование входа	Очистка, выделение фактов, определение темы
Comparator	Сравнение нового фрагмента с существующими	LLM-based semantic comparison
MVNF	Выбор merge или new file	Local и Global TopN режимы
Hebbian links	Ассоциативные связи между памятью	Усиление совместно активируемых элементов
Long-term Memory	Устойчивая база знаний	Темы подкластеры файлы, HDBSCAN
Weight & Decay	Забывание и усиление	Boost events, exponential decay, threshold cleanup

Консервативная политика MVNF оказалась ключевой для качества долговременной памяти. По результатам экспериментов подход prefer NEW при threshold 0.85 уменьшает потери информации на 82% по сравнению с агрессивными стратегиями слияния и обеспечивает стабильный рост базы без информационного коллапса.

6-ФАЗНАЯ КОНСОЛИДАЦИЯ

1. Подготовка: валидация файлов, очистка, получение embedding-представлений.
2. Переклассификация: уточнение темы с учетом контекста всего гиппокампа.
3. Глобальный RAG: поиск релевантных элементов во всей памяти.
4. Глобальный MVNF: консервативное решение merge vs new file.
5. Выбор политики: слияние с существующей памятью или создание нового узла.

6. Очистка: миграция связей, удаление временных файлов, обновление весов.

HEBBIAN АССОЦИАЦИИ И УПРАВЛЯЕМОЕ ЗАБЫВАНИЕ

Hebbian-связи реализуют принцип “элементы, которые активируются вместе, связываются”. Если несколько фрагментов памяти регулярно появляются в одном контексте, между ними усиливается ассоциативная связь. При чтении памяти это позволяет расширять поиск не только по embedding-сходству, но и по смысловым маршрутам, которые сформировались в истории пользователя.

Система весов и устаревания предотвращает переполнение памяти. Каждый узел получает динамический вес. При merge или полезном retrieval вес усиливается, например через boost event. Со временем применяется экспоненциальный decay: w(t) = w(0) x e^(-lambda t). Когда вес падает ниже порога, элемент может быть удален или исключен из активного контекста.

Экспериментальный результат: ассоциативные Hebbian-связи повысили качество recall: Recall@10 улучшился с 0.72 до 0.92. Это подтверждает, что память ассистента выигрывает от графовой структуры, а не только от векторной близости.

РАСШИРЕННАЯ МЕХАНИКА РАБОТЫ ПОДСИСТЕМ

В этом разделе система описывается не через программные модули, а через функциональные процессы. Такой взгляд важен для научного отчета: он показывает, как ассистент ведет себя в реальном сценарии, какие промежуточные решения принимает, и почему выбранная архитектура остается устойчивой при росте сложности.

ЖИЗНЕННЫЙ ЦИКЛ ГОЛОСОВОЙ КОМАНДЫ

Пользовательская команда начинается не в момент распознавания речи, а раньше - в момент появления акустического события. Микрофон непрерывно передает короткие аудиофреймы, но система не должна реагировать на каждый звук. Поэтому первый фильтр - VAD - решает, похож ли сигнал на человеческую речь. Это уменьшает число обращений к тяжелым моделям и защищает систему от бытового шума.

После появления речевой активности включается логика wake word. Она отделяет намеренное обращение от фонового разговора. Важно, что wake word не является заменой VAD: эти слои решают разные задачи. VAD отвечает на вопрос “есть ли речь”, а wake word - “обращаются ли к ассистенту”. Разделение этих задач делает контур устойчивее.

Когда wake word обнаружен, система переходит в окно активного прослушивания. В этот момент пользователь может произнести команду без повторного обращения. Pre-roll буфер

сохраняет часть аудио до начала явной записи, поэтому команда не теряет первые слоги. Это особенно важно для русского языка, где смысл может зависеть от приставки или первого слова.

Завершение записи определяется не фиксированной длительностью, а паузой. Такой подход позволяет одинаково обрабатывать короткие команды и длинные естественные формулировки. Например, “открой браузер” и “найди в документах презентацию про машинное обучение и открой папку с ней” имеют разную длину, но оба сценария завершаются по смысловой паузе.

После завершения записи аудио передается в bilingual ASR. На этом этапе система получает не только текст, но и временную структуру фразы. Word-level timestamps полезны для диагностики, последующего анализа качества и возможного расширения к мультимодальным сценариям, где речь нужно сопоставлять с действиями пользователя во времени.

Итогом voice input становится Recognition Result - нормализованное представление распознанной реплики. Оно содержит текст, язык или смешанный языковой профиль, длительность и служебные признаки. Дальше система уже работает не с аудио, а с намерением пользователя.

ПОЧЕМУ WHISPERX – КОНТУР ЛУЧШЕ КАСКАДА LID - ASR

Каскадный подход кажется естественным: сначала определить язык, затем выбрать русскую или английскую модель. Но при реальной речи этот подход ломается на смешанных фразах. Если пользователь говорит: “Запусти Chrome и найди document about neural networks”, то фраза не принадлежит одному языку. Попытка выбрать один язык приводит к потере части информации.

WhisperX/Faster-Whisper контур переносит задачу из режима “выбери язык перед распознаванием” в режим “распознай фразу целиком и восстанови языковую структуру внутри результата”. Это гораздо ближе к естественной билингвальной речи. Пользователь не обязан думать о настройке языка, потому что язык становится свойством фразы, а не ручным режимом системы.

INT8-квантизация в этом контуре играет двойную роль. С одной стороны, она ускоряет распознавание и снижает VRAM-потребление. С другой стороны, она делает большой ASR-пайплайн практичным для персонального ассистента, где ответ должен появляться быстро. Если модель точна, но задержка раздражает пользователя, она не подходит для голосового интерфейса.

С точки зрения исследования, результат важен не только как инженерная оптимизация. Он показывает, что качество голосового ассистента определяется не одной моделью, а согласованием модели, задержки, памяти GPU, обработки смешанного языка и поведения всей

pipeline-цепочки.

МЕХАНИКА VOIS ORCHESTRATOR

Оркестратор работает как предварительный когнитивный фильтр. Он не пытается заменить основную LLM, а решает более узкую задачу: классифицировать намерение и выбрать маршрут. В этом смысле он похож на диспетчера задач в операционной системе. Не всякая фраза требует глубокого рассуждения; часть фраз является командой к инструменту.

В обычной all-to-main-LLM архитектуре основная модель вынуждена обрабатывать и вопрос “объясни, как работает RAG”, и команду “открой Chrome”. Это приводит к лишним затратам. Оркестратор отделяет команды от рассуждений.

Если запрос относится к приложению, папке, файлу или документу, он направляется в Gateway. Если запрос является разговорным или неопределенным, он уходит в main LLM.

Наличие fallback-логики важно для качества. Оркестратор не обязан быть абсолютно уверенным во всех случаях. Если уверенность низкая, система не делает опасного действия, а передает запрос в диалоговый контур или уточняет намерение. Это делает архитектуру не только быстрой, но и безопасной.

Агрессивный и нормальный режимы маршрутизации решают компромисс между скоростью и осторожностью. В агрессивном режиме система быстрее выполняет очевидные действия. В нормальном режиме она чаще подтверждает действие или использует более консервативный маршрут. Такой параметр позволяет адаптировать ассистента к пользователю и контексту использования.

Экспериментальная выгода оркестратора выражается в снижении нагрузки на main LLM. По презентационным данным, intent-routing дает около 71% экономии вычислений по времени GPU на серии простых команд и ускоряет их обработку примерно в 3.5 раза. Это особенно важно для локальных систем, где вычислительные ресурсы ограничены.

МЕХАНИКА ASSISTANT GATEWAY ПРИ ЗАПУСКЕ ПРИЛОЖЕНИЙ

Запуск приложения по естественной речи сложнее, чем выполнение команды с точным путем к файлу. Пользователь говорит “запусти браузер”, “открой хром”, “запусти visual studio”, но в системе приложение может называться иначе, иметь несколько версий или быть перемещенным после обновления.

Gateway решает это через многоступенчатый поиск. Сначала используется база известных приложений и fuzzy matching. Если найден один очевидный кандидат, запуск выполняется сразу. Если найдено несколько кандидатов, включается уточнение. Если подходящего кандидата нет, система использует Everything-поиск и LLM-assisted .exe finder, который предлагает вероятные имена исполняемых файлов.

После успешного обнаружения приложения оно может быть добавлено в базу, а статистика запуска обновляется. Это превращает ассистента в обучающуюся систему: чем чаще пользователь запускает определенную программу, тем увереннее она выбирается в будущем.

Особенно важна обработка устаревших путей. В реальной ОС приложения обновляются, перемещаются и переустанавливаются. Поэтому Gateway не должен считать ошибку запуска окончательным провалом. Он может выполнить восстановление, переиндексацию, поиск нового пути и обновление записи в базе.

Такой подход делает запуск приложений похожим на устойчивую retrieval-задачу: система ищет не текстовый ответ, а исполняемый объект, оценивает кандидатов, учитывает историю и выполняет действие только после валидации.

МЕХАНИКА РАБОТЫ С ФАЙЛАМИ И ПАПКАМИ

Файловые действия требуют особенно аккуратной архитектуры. С одной стороны, пользователь ожидает естественных команд: “открой папку загрузки”, “покажи последние документы”, “найди файл с отчетом”. С другой стороны, ассистент не должен иметь бесконтрольную возможность выполнять произвольные операции с файловой системой.

Поэтому File System Handler работает как ограниченный слой навигации. Он проверяет путь, фильтрует системные и временные директории, применяет blacklist и whitelist, возвращает понятные ошибки и не делает опасных действий без необходимости. Это сохраняет удобство естественного языка, но удерживает систему в контролируемых границах.

Everything Search дополняет файловый слой быстрым индексным поиском. В обычном переборе файлов задержка могла бы быть неприемлемой, особенно на больших дисках. Everything дает быстрый доступ к кандидатам, а Gateway уже решает, какие из них безопасно показать, открыть или передать в RAG.

Таким образом, файловая подсистема не является простым wrapper над операционной системой. Это слой интерпретации пользовательского намерения, поиска, проверки безопасности и формирования результата в понятном для ассистента формате.

МЕХАНИКА ИНДЕКСИРОВАНИЯ ДОКУМЕНТОВ В NRTDS

Документный RAG начинается с индексирования. Для пользователя документ – это PDF, DOCX, PPTX или XLSX. Для retrieval-системы это набор фрагментов, метаданных, embedding-представлений, sparse-векторов и статусов индексации. Поэтому первым шагом является извлечение текста и нормализация структуры.

После извлечения текст делится на смысловые chunks с перекрытием. Перекрытие необходимо, потому что смысл часто находится на границе абзацев. Если резать документ

слишком жестко, важное утверждение может быть разделено между двумя фрагментами и стать невидимым для поиска.

Далее каждый chunk получает dense embedding. Этот слой отвечает за семантическую близость. Затем SPLADE строит sparse-представление, которое сохраняет силу точных терминов. Для кода и технических фрагментов применяется token-level подход ColBERT, позволяющий находить точные соответствия на более тонком уровне.

Индексация сопровождается метаданными: путь, формат, размер, статус, ошибки, время обработки, версия схемы. Это делает RAG эксплуатационным, а не лабораторным. Если документ не обработался, система может показать причину. Если изменилась схема, можно провести миграцию. Если документ устарел, его можно переиндексировать.

В результате NRTDS превращает локальную папку документов в поисковую базу знаний, которая поддерживает как смысловые вопросы, так и точные запросы по названиям, номерам, ключевым словам и техническим терминам.

МЕХАНИКА ПОИСКА В NRTDS

При поиске пользовательская формулировка сначала нормализуется. Система определяет, похож ли запрос на смысловой вопрос, поиск файла, точный термин, фрагмент кода или смешанный запрос. Это влияет на веса retrieval-источников.

Dense retrieval находит фрагменты, близкие по смыслу. Например, запрос “документ про долговременную память” может найти разделы, где прямо не написано слово “долговременная”, но описывается consolidation или long-term storage. Sparse retrieval, наоборот, силен в точных совпадениях: имена, аббревиатуры, формулы, номера, редкие термины.

ColBERT добавляет late interaction. Вместо того чтобы сводить весь fragment к одному vector, он сохраняет token-level взаимодействие между запросом и документом. Это особенно полезно для технического текста, где один токен может радикально изменить смысл.

RRF объединяет результаты без хрупкой нормализации разных шкал. Dense, SPLADE и ColBERT могут иметь разные score-диапазоны, но rank-based fusion устойчивее: важен порядок кандидатов внутри каждого источника. Если один документ стабильно появляется высоко в нескольких retrieval-списках, его итоговый рейтинг растет.

После fusion применяются фильтры, дедупликация и highlighting. В финальный контекст LLM попадают не все найденные документы, а компактный набор наиболее релевантных фрагментов. Это снижает контекстный шум и повышает качество ответа.

МЕХАНИКА ЗАПИСИ В ПАМЯТЬ

Запись в память начинается с события: новая реплика пользователя, важный факт, результат поиска, вывод из документа или заметка профиля. Система не должна сохранять все

как есть. Сначала вход нормализуется: удаляется шум, выделяются факты, определяется тип знания, тема и возможная связь с профилем пользователя.

Далее подключается Secretary - логический слой, который превращает сырой текст в структурированный snapshot. Он определяет, является ли запись фактом о пользователе, техническим знанием, эмоциональной заметкой, задачей, интересом или общим эпизодом.

После этого Comparator сравнивает новый snapshot с уже существующими элементами гиппокампа. Если найден близкий элемент, MVNF решает, нужно ли выполнить merge. Если сходство недостаточно надежно, создается новый файл памяти. Консервативность здесь принципиальна: потерянная деталь хуже, чем временно сохраненный отдельный фрагмент.

При записи обновляются веса и Hebbian-связи. Если новая информация связана с уже активными темами, эти связи усиливаются. Таким образом память не просто хранит факты, а постепенно строит карту отношений между ними.

Важный результат такого подхода - контроль роста. Память не становится хаотичной историей всех сообщений. Она проходит через структурирование, сравнение, консолидацию и забывание.

МЕХАНИКА ЧТЕНИЯ ПАМЯТИ

Чтение памяти начинается с запроса. Запрос кодируется в embedding-представление, после чего retriever находит кандидатов. Но в отличие от обычного RAG, memory retrieval учитывает не только similarity. Важны свежесть, вес узла, маршрут связи, частота использования и ассоциативные переходы.

Hebbian expansion расширяет начальный список кандидатов. Если найденный фрагмент памяти связан с другим важным узлом, система может добавить его в рассмотрение даже при более слабом прямом embedding-сходстве. Это имитирует человеческую ассоциацию: одна мысль вызывает другую не только по формальному сходству, но и по истории совместного использования.

Ranker объединяет признаки в итоговый score. Поэтому в контекст LLM попадает не самый похожий текст в вакууме, а наиболее полезная память с учетом семантики, важности, маршрута и времени. Такой подход снижает риск того, что старый, но случайно похожий фрагмент вытеснит действительно важный персональный факт.

В диалоговом сценарии память работает как фоновый контекст. Пользователь не обязан каждый раз повторять свои предпочтения, интересы или ранее обсуждавшиеся темы. Ассистент может восстановить релевантные сведения и использовать их в ответе.

КОНСОЛИДАЦИЯ КАК ЦЕНТРАЛЬНЫЙ НАУЧНЫЙ МЕХАНИЗМ ПАМЯТИ

Консолидация решает противоречие между скоростью записи и качеством долговременной структуры. Во время диалога система должна быстро сохранять новые эпизоды, поэтому они попадают в Hipposampus. Но долговременная память требует более дорогой обработки: переклассификации, поиска похожих элементов, сравнения и построения связей.

Шестифазный pipeline позволяет разделить эти режимы. Быстрая запись не блокирует диалог, а более глубокая обработка выполняется как отдельный цикл. Это напоминает биологическую идею: кратковременная память фиксирует свежий опыт, а затем во время консолидации переносит устойчивые знания в долгосрочное хранилище.

Фаза глобального RAG особенно важна: новая запись сравнивается не только с локальными соседями, но и с более широкой базой. Это снижает риск дублирования тем, которые уже давно существуют в long-term памяти.

Global MVNF принимает консервативное решение. Если уверенность в слиянии недостаточна, создается новый элемент. Такая стратегия лучше сохраняет детали и предотвращает информационный коллапс, когда разные факты насильно объединяются в один расплывчатый узел.

После консолидации связи мигрируют, веса обновляются, временные элементы очищаются. Память остается живой структурой, а не статическим архивом.

ПОЛЬЗОВАТЕЛЬСКИЙ ПРОФИЛЬ

Профиль пользователя является специализированной частью памяти. Он хранит не все реплики, а устойчивые сведения: факты, интересы, уровень знаний, эмоциональные заметки и предпочтения. Это позволяет ассистенту адаптировать стиль ответа, глубину объяснения и выбор инструментов.

Например, если пользователь часто работает с Python, нейросетями и документами, ассистент может ранжировать соответствующие воспоминания выше. Если пользователь предпочитает краткие ответы, система может снижать объем пояснений. Если обсуждалась подготовка к докладу, эта задача может оставаться активной в контексте.

Принципиально важно, что профиль не заменяет память. Профиль хранит устойчивые характеристики пользователя, а memory graph хранит эпизоды, документы, связи и знания. Вместе они формируют персонализацию без превращения всей истории в одну неструктурированную заметку.

СИНХРОНИЗАЦИЯ ГОЛОСА, LLM И TTS

Голосовой ассистент должен не только правильно ответить, но и сделать это с естественным темпом. Поэтому LLM processor работает в streaming-режиме: ответ может разбиваться на смысловые фрагменты, которые постепенно отправляются в TTS. Это уменьшает субъективное ожидание пользователя.

TTS, в свою очередь, получает не весь разговорный контекст, а текстовые фрагменты, готовые к озвучиванию. Если пользователь прерывает ассистента, SpeechPlayer и очереди должны корректно остановить воспроизведение и освободить тракт для новой команды. Такой interruptible behavior является важным условием естественного диалога.

Система также должна избегать самопрослушивания. Если ассистент произнес ответ, микрофон может уловить собственный голос. Echo guard и состояние currently speaking помогают не отправлять этот звук обратно в ASR как новую пользовательскую реплику.

БЕЗОПАСНОСТЬ, НАДЕЖНОСТЬ И НАБЛЮДАЕМОСТЬ

Так как JARVIS 1 не ограничивается генерацией текста и способен выполнять действия в ОС, безопасность является частью архитектуры, а не дополнительной функцией. Система должна проверять путь, тип операции, доступность компонента и потенциальный риск до фактического действия.

Таблица 11.

Механизм	Назначение	Практический эффект
Whitelist/blacklist путей	Запрет системных, временных и опасных директорий	Снижение риска некорректного доступа
Extension filtering	Контроль расширений и типов файлов	Предсказуемая работа с документами
Sandbox execution	Ограничение выполнения команд	Снижение риска ошибочного запуска
Component health	Статусы online/degraded/disabled	Понятная диагностика отказов
Fallback to LLM	Мягкое поведение при низкой уверенности	Система не ломает диалог при неопределенности
Slash-команды	Операционная поверхность без ASR	Ремонтопригодность и тестируемость

Наблюдаемость проявляется на нескольких уровнях: latency, real-time factor,

memory/VRAM peaks, error rate, состояние gateway-компонентов, статус индексации документов, журналы сессий и тестовые сценарии. Это переводит проект из стадии демонстрации в стадию управляемой инженерной системы.

ЭКСПЕРИМЕНТАЛЬНЫЕ РАЗВИЛКИ И РЕЗУЛЬТАТЫ

Развитие JARVIS 1 шло через серию сравнительных решений. Для каждого ключевого слоя проверялись альтернативные реализации: более простые baseline-подходы, частичные гибриды и целевые архитектуры. Такой путь важен, потому что итоговая система не выглядит случайной: каждый компонент появился как ответ на конкретное ограничение предыдущей версии.

Таблица 12.

Область	Испытанные варианты	Итоговый выбор	Причина
ASR	Каскад LID + отдельные ASR, WhisperX	WhisperX/Faster-Whisper Large V3 INT8	Меньше задержка, проще код, лучше code-switching
Квантование ASR	FP32, FP16, INT8, INT4	INT8	Лучший баланс скорости, VRAM и качества
RAG	Dense-only, Dense+BM25, Dense+SPLADE, Full Hybrid	Dense + SPLADE + ColBERT + RRF	MRR@10 0.89 и устойчивость к разным запросам
Memory storage	Flat storage, topic rules, cognitive memory	Hippocampus -> Long-term	Контролируемый рост и меньше дублей
MVNF policy	Threshold sweep 0.60..0.95	prefer NEW около 0.85	82% меньше потерь информации
Routing	All-to-main-LLM, orchestrator-first	Voice Orchestrator	71% экономии compute на простых командах
Wake word	Простой threshold, VAD+wake+window	Многошаговый gate	Меньше false positives при сохранении чувствительности
TTS	Отдельные RU/EN ветки, единый bilingual TTS	Chatterbox + RVC + Post-FX	Меньше стыков и лучше единый голосовой образ

СВОДНЫЕ МЕТРИКИ

Таблица 13.

Метрика	Достигнутое значение	Интерпретация
WER русский	3.2%	Высокая точность ASR на целевом RU-наборе
WER английский	2.1%	Стабильная работа EN-ветки
ASR ускорение	3.2x	Эффект INT8 квантизации
RAG MRR@10	0.89 vs 0.65 dense-only	Full hybrid лучше dense-only на 37%
RAG latency	800-1200 мс	Практический диапазон для интерактивного поиска
Orchestrator latency	200-500 мс	Быстрая классификация намерений
Compute saving	около 71%	Снижение нагрузки на main LLM/GPU
Hebbian recall	0.72 -> 0.92	Ассоциативные связи улучшают поиск памяти
MVNF information loss	на 82% меньше	Консервативная стратегия лучше агрессивного merge

МЕТОДИКА ЭКСПЕРИМЕНТАЛЬНОЙ ОЦЕНКИ

Экспериментальная оценка JARVIS 1 строилась не вокруг одной метрики, а вокруг набора практических показателей, соответствующих разным слоям ассистента. Для голосового контура важны WER, задержка, устойчивость к code-switching и субъективная естественность диалога. Для RAG важны MRR@10, recall, устойчивость к переформулировкам и задержка поиска. Для памяти важны потери информации, количество дублей, качество recall и стабильность роста базы.

Такой подход принципиален: универсальный ассистент нельзя корректно оценивать одной цифрой. Высокая точность ASR бесполезна, если система не умеет выполнить команду.

Быстрый запуск приложения не решает проблему, если ассистент забывает контекст пользователя. Сильная LLM не компенсирует плохой retrieval по документам. Поэтому оценка велась по подсистемам и по end-to-end сценариям.

В качестве baseline использовались более простые архитектуры: all-to-main-LLM для маршрутизации, dense-only retrieval для RAG, flat storage для памяти, каскад LID + ASR для билингвального входа, отдельные TTS-ветки для RU/EN синтеза. Затем эти baseline сравнивались с целевыми решениями, выбранными в JARVIS 1.

Результаты проверялись не только автоматическими метриками, но и ручным просмотром ошибок. Это особенно важно для voice assistant, потому что некоторые ошибки формально небольшие, но пользовательски критичные. Например, неверно распознанное имя приложения может полностью изменить действие, а незначительная ошибка пунктуации в обычном диалоге может быть незаметной.

Таблица 14.

Слой	Метрики	Зачем нужны
ASR	WER, latency, code-switching success rate	Проверка точности и пригодности для живого голосового ввода
TTS	MOS-оценка, отсутствие языковых швов, стабильность громкости	Оценка естественности и узнаваемости голоса
Orchestrator	Latency, confidence, action success, fallback rate	Проверка правильности маршрутизации
Gateway	Tool success rate, recovery rate, false action rate	Оценка надежности действий в ОС
RAG	MRR@10, Recall@K, latency, exact-match success	Проверка поиска по документам
Memory	Information loss, duplicate rate, Recall@10, graph expansion gain	Оценка долговременного контекста
End-to-end	User-perceived latency, completion success, interruption handling	Проверка ассистента как целой системы

СЦЕНАРИЙ 1: ЗАПУСК ПРИЛОЖЕНИЯ ПРИ ЕСТЕСТВЕННОЙ РЕЧИ

Пользователь произносит: “запусти visual studio”. Для обычной LLM это текстовая просьба, но для JARVIS 1 это действие. После ASR запрос попадает в оркестратор, который классифицирует его как запуск приложения. Далее Gateway ищет кандидатов в базе приложений. Если находится Visual Studio Code и Visual Studio 2022, система использует disambiguation-логику или учитывает историю запусков.

Если пользователь ранее чаще запускал Visual Studio Code, этот кандидат получает преимущество. Если уверенности недостаточно, ассистент уточняет выбор. После выбора Gateway валидирует путь, запускает приложение и обновляет статистику. В ответ пользователь слышит короткое подтверждение, а не длинное объяснение.

Научный смысл сценария в том, что естественный язык превращается в управляемую операционную процедуру. Здесь задействованы ASR, orchestrator, application database, fuzzy matching, filesystem validation, launch statistics и TTS. Успех сценария показывает, что ассистент способен действовать, а не только отвечать.

СЦЕНАРИЙ 2: ВОПРОС ПО ДОКУМЕНТАМ

Пользователь спрашивает: “найди, где в отчете описан hybrid RAG и объясни кратко”. В этом сценарии оркестратор определяет документный запрос и направляет его в RAG-контур. NRTDS выполняет несколько параллельных retrieval-режимов: dense-поиск находит смысловые фрагменты, SPLADE усиливает ключевые слова hybrid/RAG, ColBERT уточняет token-level совпадения.

После RRF fusion система получает ранжированный набор фрагментов. В main LLM передается не весь документ, а компактный контекст. Ответ становится обоснованным: ассистент может указать, что hybrid RAG объединяет Jina dense embeddings, SPLADE sparse retrieval, ColBERT late interaction и RRF.

Практическая ценность такого сценария особенно высока для учебных, юридических, медицинских и финансовых материалов. Пользователь может обращаться к локальной базе документов естественным языком, не зная точного имени файла, страницы или формулировки.

СЦЕНАРИЙ 3: ПАМЯТЬ В ДОЛГОМ ДИАЛОГЕ

В обычном ассистенте пользователь каждый раз заново объясняет контекст: над каким проектом он работает, какие темы ему интересны, какой стиль ответа предпочтителен. В JARVIS 1 эти сведения сохраняются в memory layer. Когда пользователь возвращается к проекту, ассистент может восстановить релевантные факты и не начинать с нуля.

Например, если ранее обсуждалась подготовка доклада по JARVIS 1, memory retrieval

может поднять сведения о презентации, RAG, системе памяти, code-switching и требованиях к отчету. Hebbian expansion добавит связанные темы, даже если пользователь прямо не назвал их в новом запросе.

Этот сценарий демонстрирует отличие memory layer от простой истории чата. История чата линейна и быстро переполняется. Когнитивная память структурирует знания, связывает их, усиливает полезные элементы и забывает неактуальные.

СЦЕНАРИЙ 4: СМЕШАННАЯ РУССКО-АНГЛИЙСКАЯ КОМАНДА

Пользователь говорит: “Открой папку projects и найди document about machine learning”. В такой фразе действие сформулировано на русском, объект файловой системы может быть английским, а смысловой запрос к документам также английский. Традиционный монолингвальный ассистент потребовал бы переключения языка или потерял бы часть фразы.

JARVIS 1 обрабатывает такую команду как единое намерение. ASR восстанавливает mixed-language текст, оркестратор определяет, что запрос содержит файловую и документную часть, Gateway ищет папку, а RAG при необходимости ищет документ. Ответ также может быть синтезирован в билингвальном режиме.

Этот сценарий важен потому, что реальная техническая речь часто смешанная. Названия библиотек, документов, приложений и терминов почти всегда англоязычные, даже когда пользователь говорит по-русски. Поддержка code-switching делает ассистента пригодным для реальной инженерной среды.

СЦЕНАРИЙ 5: МНОГОСПИКЕРНАЯ СРЕДА

В многоспикерном сценарии рядом с ассистентом могут говорить два человека. Если ASR получает смешанный сигнал напрямую, он может объединить реплики, потерять последовательность или неверно определить команду. ReSepFormer-сепарация позволяет выделить источники и обработать их отдельно.

После разделения каждый канал проходит постобработку: подавление bleed, spectral gating, band-pass filtering и loudness normalization. Далее ASR формирует реплики по говорящим, а LLM получает структурированный контекст. Например: первый пользователь задает вопрос, второй уточняет объект поиска.

Такой сценарий расширяет ассистента от индивидуального инструмента к групповой среде: учебная лаборатория, рабочее совещание, семейный ассистент или демонстрация проекта.

ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

Главный результат исследования состоит в том, что JARVIS 1 показывает жизнеспособность многослойного подхода. Ни одна подсистема сама по себе не делает ассистента универсальным. Универсальность возникает из согласования: ASR дает качественный текст, оркестратор выбирает маршрут, Gateway выполняет действия, RAG извлекает знания, память сохраняет контекст, TTS возвращает ответ в естественной форме.

Метрики подтверждают это на отдельных слоях. WhisperX/INT8 дает практичную скорость и точность. Hybrid RAG значительно превосходит dense-only. Оркестратор разгружает main LLM. Hebbian-связи улучшают recall памяти. MVNF снижает потери информации. Вместе эти результаты показывают не просто набор улучшений, а единую закономерность: специализация слоев повышает качество всей системы.

С инженерной точки зрения проект ценен тем, что учитывает эксплуатацию: health-статусы, fallback, slash-команды, GUI/TUI, диагностические команды RAG и app index, контроль безопасности. Это отличает исследовательский прототип от одноразовой демонстрации.

С научной точки зрения наиболее сильными являются четыре вклада: когнитивная память Hippocampus -> Long-term, гибридный retrieval Dense + SPLADE

+ ColBERT + RRF, intent-routing перед main LLM и real-time bilingual voice pipeline. Эти направления могут использоваться не только внутри JARVIS 1, но и как самостоятельные архитектурные идеи для других персональных AI-систем.

ИНТЕРФЕЙСЫ И ОПЕРАЦИОННАЯ ПОВЕРХНОСТЬ

JARVIS 1 имеет несколько способов взаимодействия. Голосовой интерфейс является основным пользовательским сценарием, но для эксплуатации, тестирования и обслуживания предусмотрены CLI, консольный gateway-чат, TUI/GUI и slash-команды. Это важно для зрелости проекта: сложную систему нельзя обслуживать только через голос.

Таблица 15.

Интерфейс	Назначение	Сценарий
Voice mode	Основной режим ассистента	Команды и диалог естественной речью
CLI	Запуск, загрузка моделей, прямой TTS	Разработка и эксплуатация

TUI	Терминальная визуализация и управление	Мониторинг без тяжелого GUI
PyQt GUI	Графическая панель состояния	Pipeline status, confidence, telemetry, spectrum
/app	Операции приложений	Индексация, поиск, запуск, aliases, stats
/rag	Операции RAG	Статус, reindex, search, errors, backup, migration
API	Интеграция с внешними клиентами	TTS endpoint и gateway routes

СРАВНЕНИЕ С АНАЛОГАМИ

JARVIS 1 не пытается победить коммерческие системы по каждому отдельному параметру. Google Assistant и Alexa сильнее по глобальной многоязычности и зрелости облачной инфраструктуры. Однако JARVIS 1 занимает другую нишу: приватная, расширяемая, локально ориентированная AI-платформа с долговременной памятью, gateway-действиями, гибридным RAG и настоящим RU/EN code-switching.

Таблица 16.

Критерий	Коммерческие ассистенты	Типичные open-source ассистенты	JARVIS 1
Приватность	Ограниченный контроль пользователя	Часто локально, но функционально беднее	Локально ориентированный контур
Долговременная память	Ограничена и непрозрачна	Обычно отсутствует	Hippocampus -> Long-term, MVNF, Hebbian
RAG по документам	Зависит от экосистемы	Чаще dense-only или отсутствует	Hybrid Dense + SPLADE + ColBERT + RRF
Действия в ОС	Ограничены платформой	Часто скриптовые	Assistant Gateway с безопасностью
Code-switching RU/EN	Есть частично, но не локально	Редко поддерживается на таком уровне	Целевой bilingual pipeline
Расширяемость	Закрытая платформа	Высокая, но мало интеграции	Модульная архитектура ассистента

Позиция проекта: главное преимущество JARVIS 1 не в одной отдельной метрике, а в балансе: приватность, расширяемость, память, действия, документы и билингвальность объединены в одной системе

ПРАКТИЧЕСКАЯ И КОММЕРЧЕСКАЯ ЗНАЧИМОСТЬ

Архитектура JARVIS 1 имеет прикладной потенциал в областях, где важны документы, приватность, память и последовательные действия. В юридической сфере такой ассистент может помогать искать положения договоров, поддерживать контекст дел и выполнять подготовительные операции с файлами. В медицине и административных организациях он может выступать как локальный помощник для работы с документами и справочной информацией. В финансовой сфере особенно важны быстрый поиск по договорам, номерам, суммам и связанным документам.

Модульность позволяет адаптировать систему к мощности заказчика. Можно отключать тяжелые компоненты, менять модели, использовать локальные или гибридные LLM, ограничивать инструменты, расширять gateway и подключать новые источники данных. Поэтому JARVIS 1 можно рассматривать не только как персональный проект, но и как прототип платформы для enterprise-ассистентов.

ПЕРСПЕКТИВЫ РАЗВИТИЯ

- Vision/VLM module: анализ экрана, OCR, помощь с отладкой, распознавание объектов.
- Система анимации и движения: интеграция с Arduino/Raspberry Pi, motor control, 3D-печатные элементы.
- Улучшенный профиль пользователя: временная динамика интересов, персональные рекомендации, адаптивный стиль общения.
- Расширение tool coverage: интернет-поиск, календарь, уведомления, электронная почта, умный дом.
- Мультипрофильность: поддержка нескольких пользователей, семейного режима и разделения персональных контекстов.

Перспективы проекта логично продолжают уже выбранную архитектуру. В систему не нужно добавлять новые возможности монолитно: каждый новый слой может подключаться через gateway, memory, RAG или orchestrator policy.

ЗАКЛЮЧЕНИЕ

JARVIS 1 является комплексным исследовательским и инженерным проектом, который объединяет технологии ASR, TTS, LLM, RAG, графовой памяти, безопасного выполнения действий и real-time аудиообработки. Важнейшее достижение проекта состоит в том, что ассистент построен как система, а не как одиночный чат-бот.

В ходе работы была сформирована устойчивая архитектура из взаимозаменяемых подсистем. WhisperX/Faster-Whisper контур обеспечил точное билингвальное распознавание и code-switching. Voice Orchestrator снизил нагрузку на основную LLM и превратил ассистента в диспетчер действий. Assistant Gateway дал безопасный доступ к приложениям, файлам и документам. NRTDS v2 показал преимущество гибридного retrieval. Когнитивная память обеспечила модель долговременного персонального контекста.

Научная ценность проекта заключается в сочетании нескольких направлений: когнитивная модель памяти, гибридный retrieval, маршрутизация намерений и real-time voice pipeline. Практическая ценность заключается в демонстрации того, что локальный ассистент может быть конкурентоспособным не за счет копирования коммерческих решений, а за счет иной архитектурной философии: приватность, модульность, управляемость и расширяемость.

Финальный вывод: JARVIS 1 доказывает возможность создания универсального персонального ассистента, который не ограничивается ответами на вопросы, а способен понимать речь, помнить, искать, действовать и развиваться как полноценная AI-платформа.

ПРИЛОЖЕНИ А. ТЕХНОЛОГИЧЕСКАЯ КАРТА

Таблица 17.

Группа	Технологии и методы
ASR	WhisperX/Faster-Whisper Large V3, INT8, word-level timestamps, language identification
Audio	SoundDevice, Silero VAD, openWakeWord, pre-roll, echo guard
TTS	Chatterbox bilingual TTS, voice cloning, Unified RVC, post-FX
LLM	Main LLM, streaming generation, conversation history, warmup
Routing	IBM Granite 4H-micro, intent classification, fallback policy
Gateway	Application Handler, FileSystem Handler, Search Handler, Security validation
RAG	Jina-embeddings-v3, SPLADE, ColBERT, RRF, ChromaDB
Memory	Hippocampus, Long-term graph, MVNF, Hebbian links, Weight/Decay
UI/Ops	CLI, TUI, PyQt GUI, slash commands, API

ПРИЛОЖЕНИЕ В. ФОРМУЛЫ И МЕТРИКИ

Таблица 18.

Обозначение	Формула / значение	Смысл
Exponential decay	w(t) = w(0) x e^(-lambda t)	Снижение веса устаревающих элементов памяти
RRF	RRF_score = sum(1 / (k + rank_i))	Слияние рангов нескольких retrieval-методов
Composite memory score	0.5*similarity + 0.2*route + 0.15*recency + 0.15*node_weight	Итоговое ранжирование воспоминаний
MRR@10	0.89 против 0.65	Преимущество гибридного RAG над dense-only
Recall@10	0.72 -> 0.92	Эффект Hebbian expansion

ПРИЛОЖЕНИЕ С. ПАСПОРТА ОСНОВНЫХ ПОДСИСТЕМ

Voice Input

Подсистема голосового ввода отвечает за превращение непрерывного аудиопотока в осмысленный речевой сегмент. Ее задача не сводится к записи микрофона: она должна отличить речь от шума, распознать обращение к ассистенту, сохранить начало команды и корректно определить конец фразы. Поэтому voice input использует многоступенчатую схему: audio capture, VAD, wake word, pre-roll buffer, pause detection и echo protection. Результатом работы является чистый аудиосегмент, который можно передать в ASR без лишней нагрузки и без риска, что ассистент начнет обрабатывать собственный голос.

Bilingual ASR

ASR-подсистема является первым интеллектуальным слоем. Она принимает аудиосегмент и превращает его в текст с временной и языковой структурой. Выбор WhisperX/Faster-Whisper Large V3 в INT8-профиле позволил объединить точность, скорость и поддержку RU/EN code-switching. Важнейший результат заключается в отказе от ручного переключения языков: пользователь может говорить естественно, смешивая русский язык с английскими терминами, названиями библиотек, приложений и документов.

LLM Processor

LLM Processor связывает распознанную реплику с диалоговой моделью, orchestrator route, slash-командами и TTS. Он поддерживает потоковую генерацию, историю разговора и передачу фрагментов ответа в синтез. Благодаря этому ассистент не ждет полного завершения длинного ответа, а может начать озвучивание по мере появления смысловых блоков. Этот слой отвечает за ощущение живого диалога и за согласование текстового reasoning с голосовым выводом.

Voice Orchestrator

Voice Orchestrator решает, какой маршрут нужен запросу. Если пользователь просит открыть приложение, искать документ или выполнить файловое действие, запрос не должен уходить в тяжелую main LLM как обычный разговор. Оркестратор классифицирует намерение, оценивает уверенность и выбирает маршрут: Gateway, RAG, main LLM, уточнение или fallback. Тем самым он экономит вычисления, снижает задержку и делает систему более похожей на настоящего агента.

Assistant Gateway

Gateway является исполнительным слоем. Он принимает структурированное намерение и выполняет действие: открывает приложение, ищет файл, показывает папку, запускает URL, обращается к RAG или возвращает диагностический статус. Gateway отделяет естественный язык от конкретных системных операций. Это важно для безопасности: ассистент

не выполняет произвольные команды напрямую, а проходит через handler-логику, валидацию и контролируемые политики.

NRTDS RAG

NRTDS v2 отвечает за поиск по документам и знаниям. Его особенность -гибридность. Dense embeddings дают семантическое сходство, SPLADE усиливает точные ключевые слова, ColBERT добавляет token-level совпадения, а RRF объединяет результаты. Такая система лучше подходит для реальных документов, где пользователь может спрашивать как смыслом, так и точными названиями, номерами, аббревиатурами или техническими терминами.

Memory System

Система памяти реализует долговременный персональный контекст. В отличие от истории чата, она имеет жизненный цикл: запись в Hippocampus, нормализация Secretary, сравнение Comparator, решение MVNF, Hebbian-связи, консолидация в Long-term и Weight/Decay обслуживание. Такая память не просто хранит прошлые фразы, а организует знания в структуру, где важные элементы усиливаются, связанные темы соединяются, а устаревшие сведения постепенно теряют вес.

Voice Output

Выходной голосовой контур превращает текстовый ответ в аудио. Chatterbox TTS обеспечивает билингвальный синтез, RVC формирует целевой голосовой характер, а post-FX pipeline делает речь акустически стабильной. Этот слой важен не только для красоты звучания, но и для целостности ассистента: если входная часть понимает code-switching, выходная часть также должна уверенно озвучивать смешанные RU/EN ответы.

Interfaces and Observability

Интерфейсы и наблюдаемость превращают проект в управляемую систему. CLI, TUI, GUI, slash-команды и API дают разные режимы доступа. Health-статусы, метрики, логи, очереди и диагностические команды позволяют видеть состояние компонентов. Это особенно важно для AI-системы, где ошибка может возникнуть в модели, индексе, аудио, устройстве вывода, gateway-handler или внешнем сервисе.

ПРИЛОЖЕНИЕ D. ИСПЫТАНИЕ РЕАЛИЗАЦИИ И ПРИЧИНЫ ВЫБОРА ТЕКУЩИХ РЕШЕНИЙ

История проекта важна для понимания финальной архитектуры. Итоговые решения не были выбраны заранее: они появились после сравнения более простых вариантов, выявления их ограничений и перехода к более устойчивым схемам.

ASR: отдельные языковые ветки

Первоначальная идея с отдельными моделями для разных языков выглядела логичной,

но плохо подходила для реального code-switching. Ошибка language identification в начале цепочки могла испортить весь результат. Единый WhisperX-контур оказался проще, быстрее и устойчивее.

TTS: параллельные RU/EN синтезаторы

Раздельный TTS давал контроль над языками, но создавал проблему стыков и усложнял сборку ответа. Chatterbox bilingual TTS позволил перейти к единому синтезу и сделать голосовой ответ более естественным.

RAG: dense-only

Dense-only хорошо работает на смысловых вопросах, но проигрывает на точных терминах, номерах, именах и технических фрагментах. Гибрид Dense + SPLADE + ColBERT + RRF стал целевым решением, потому что покрывает разные типы запросов.

Memory: flat storage

Плоское накопление воспоминаний быстро приводит к дублированию и снижению релевантности. Двухуровневая память с Hippocampus и Long-term решает эту проблему через консолидацию, MVNF и decay.

Routing: all-to-main-LLM

Отправка всех запросов в основную LLM проста, но неэффективна. Простые команды не требуют тяжелого reasoning. Voice Orchestrator снижает нагрузку и делает поведение ассистента более агентным.

Application search: только база приложений

Одна база быстро устаревает после обновлений и переносов программ. Интеграция fuzzy matching, Everything search, LLM-assisted exe finder и статистики запусков дала более устойчивый контур.

Wake word: простой порог

Один порог не выдерживает реальной акустической среды: либо слишком много ложных срабатываний, либо пропуски. Многоступенчатая схема VAD + wake word + listening window дает лучший баланс.

Memory merge: агрессивное слияние

Агрессивный merge уменьшает количество файлов, но теряет детали. Консервативный MVNF при threshold около 0.85 сохраняет больше информации и предотвращает смысловое смешивание разных фактов.

ПРИЛОЖЕНИЕ Е. ТЕРМИНОЛОГИЧЕСКИЙ СЛОВАРЬ

Таблица 19.

Термин	Пояснение
ASR	Automatic Speech Recognition, преобразование речи в текст
TTS	Text-to-Speech, синтез речи из текста
Code-switching	Смешение языков внутри одной фразы или диалога
VAD	Voice Activity Detection, определение голосовой активности

Wake word	Активационная фраза или нейронный детектор обращения к ассистенту
RAG	Retrieval-Augmented Generation, генерация ответа с использованием найденного контекста
Dense retrieval	Семантический поиск через embedding-векторы
Sparse retrieval	Поиск с учетом важности конкретных терминов и ключевых слов
ColBERT	Late interaction retrieval на уровне токенов
RRF	Reciprocal Rank Fusion, объединение списков результатов по рангам
MVNF	Merge vs New File, решение о слиянии или создании нового элемента памяти
Hebbian links	Ассоциативные связи между элементами памяти
Decay	Постепенное уменьшение веса устаревающих элементов
Gateway	Слой выполнения действий в ОС и инструментах
Orchestrator	Слой маршрутизации намерений перед основной LLM

РАЗРАБОТКА СИСТЕМЫ ПОСАДКИ КОСМИЧЕСКОГО АППАРАТА НА ПЛАНЕТУ МАРС

Семин Александр Витальевич

Москва г, ГБОУ Инженерная школа № 1581, 11 класс

Секция: Космические аппараты и ракеты-носители

Научный руководитель: Товарных Геннадий Николаевич, кандидат технических наук, доцент кафедры СМ-1 МГТУ им Н. Э. Баумана

АННОТАЦИЯ

Актуальность темы. Тема освоения Марса в настоящее время актуальна. Марс является идеальной планетой для колонизации, что может дать незаменимый опыт человечеству и создать огромный технологический рывок. Также освоение Марса способно решить будущую проблему нехватки ресурсов на Земле. Для науки освоение Марса станет платформой для новых открытий. Именно поэтому во многих странах разрабатываются проекты по освоению Марса. Одной из проблем освоения Марса является доставка грузов и людей на его поверхность.

Целью работы является разработка системы посадки космического аппарата для полетов на планету Марс. Аппарат предназначен для безопасной доставки экипажа и грузов с орбиты Марса на его поверхность.

Методы исследования: анализ литературы, исторический метод, математические методы: метод уравнений и неравенств, подготовка эскиза посадочного модуля.

Выводы и результаты исследований.

1. Рассмотрены существующие варианты космических аппаратов, которые спускались на поверхность Марса.
2. Проведены расчеты для трех вариантов конфигурации посадочного модуля.
3. Написаны программы на языке программирования Python, с помощью которых были получены теоретические данные.
4. Используя полученные данные, была выбрана наиболее выгодная конфигурация посадочного модуля.
5. Создан эскиз выгодного варианта конфигурации посадочного модуля.

ВВЕДЕНИЕ

Марс является наиболее перспективным объектом в солнечной системе. В отличие от Венеры, климат которой является слишком экстремальным для проведения исследовательских миссий, Марс является главным кандидатом для будущей колонизации. На это также указывают наличие атмосферы и водяного льда, которые могут помочь в колонизации планеты. Также необходимость полетов на Марс обуславливается проблемой истощения ресурсов Земли. Так, Марс может быть богат железом, алюминием, титаном, серой, хлором и многими другими полезными элементами. Также Марс может стать идеальной площадкой для новых научных баз. Именно поэтому во многих странах идет разработка проектов по освоению Марса.

Актуальность работы несомненна, так как от того, как совершит посадку полезный груз, зависит успех миссии по колонизации Марса.

Объектами моей исследовательской работы являются: формулы расчета массовых характеристик космического аппарата, коэффициентов, а также сил, действующих на космический аппарат; программы на языке программирования Python, написанные с помощью данных формул, позволяющие получить теоретические данные; эскиз посадочного модуля

Предмет работы – наиболее выгодная конфигурация системы посадки космического аппарата.

Проблема данной работы заключается в выборе наиболее выгодной конфигурации системы посадки космического аппарата, созданного на основе существующих космических технологий.

Основные цели моей работы заключаются в анализе различных конфигураций системы посадки космического аппарата и выборе наиболее выгодного варианта.

В ходе выполнения работы мной решаются следующие задачи:

1. Рассмотреть конфигурации посадочного модуля;
2. Проанализировать и выбрать самую выгодную конфигурацию.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Физические характеристики Марса

1. Масса и размер планеты

Марс - четвертая планета от Солнца, относящаяся к земной группе. По размерам он почти вдвое меньше Земли: его масса составляет около 10,7% земной, а диаметр - примерно 53%. Среди планет Солнечной системы он уступает по величине всем, кроме Меркурия. Из-за вращения планета немного сплюснута у полюсов: её средний экваториальный радиус составляет около 3396 км, а полярный - примерно 3375 км.

2. Атмосфера

Атмосфера Марса почти полностью состоит из углекислого газа и отличается очень низкой плотностью. Среднее атмосферное давление на поверхности планеты составляет примерно 6,1 мбар — это примерно в 160 раз меньше, чем на Земле. Из-за значительных перепадов высот на Марсе давление у поверхности может сильно различаться в разных районах. Высота однородной атмосферы составляет около 11,1 км, а общая масса всей атмосферной оболочки оценивается примерно в $2{,}5 \cdot 10^{16}$ кг.

Температура на поверхности Марса подвержена большим колебаниям: зимой на полюсах она может опускаться до −153 °C, а летом на экваторе подниматься до +20 °C. Наибольшая зарегистрированная температура воздуха, которую зафиксировал марсоход «Спирит», достигла +35 °C. В средних широтах типичные значения температуры меняются от −50 °C ночью зимой до 0 °C днём летом.

Эффективная температура планеты составляет 210 К (−63 °C), при этом средняя температура поверхности равна примерно 215 К (−58 °C). Это соответствует очень слабому парниковому эффекту атмосферы — всего около 5 °C.

4. Сила тяжести

Ускорение свободного падения на поверхности Марса примерно в 2,5 раза меньше, чем на Земле (примерно 39–40 % от земного значения). Средняя величина g на Марсе составляет около 3,721 м/с².

Как и на Земле, точное значение силы тяжести на Марсе не постоянно и может заметно различаться в разных регионах. Эти локальные вариации гравитационного поля связаны с неоднородностью плотности недр планеты. На них также влияют: различия в толщине коры, следы прошлой геологической активности, сезонные изменения массы полярных шапок, перераспределение атмосферной массы, изменения пористости и структуры поверхностных пород.

Изучение особенностей поверхностной гравитации Марса позволяет лучше понять внутреннее строение планеты и получить важные данные для планирования и безопасности будущих посадочных миссий.

5. Состав планеты

Химический состав поверхностного слоя марсианского грунта, полученный с помощью посадочных аппаратов, существенно различается в разных районах. Главным компонентом почвы является кремнезём, который содержит примеси гидратированных оксидов железа, придающие грунту характерный красноватый оттенок. Также в значительных количествах присутствуют соединения серы, кальция, алюминия, магния и натрия.

По данным зонда НАСА «Феникс», значения pH и ряд других характеристик марсианских почв близки к земным показателям, что теоретически позволяет рассматривать

возможность выращивания растений. Марсианский грунт содержит необходимые питательные элементы, способные поддерживать жизнь — как в прошлом, так и в настоящем или будущем. Кроме того, именно «Феникс» впервые подтвердил наличие водяного льда непосредственно в почве.

Позднее марсоход «Curiosity» в 2013 году (данные обнародованы в сентябре) показал, что содержание воды в подповерхностных слоях Марса заметно выше прежних оценок. В некоторых изученных породах доля воды может достигать до 2 % по массе.

6. Миссии к Марсу

История исследования Марса насчитывает уже более 60 лет и полна как триумфов, так и неудач. Вот основные вехи освоения Красной планеты и планы на будущее.

7. Первые попытки

В 1960 году СССР запускает станции серии 1М («Марс 1960А» и «Марс 1960Б»). Оба запуска, ставшие первыми попытками исследовать Марс, оказались неудачными из-за аварий ракет-носителей на ранних этапах, и аппараты не вышли на околоземную орбиту.

В 1962 году был запущен «Марс 1», который вышел на траекторию полёта к планете, но прервал связь за 106 млн км до цели. 15 июля 1965 года американский «Маринер-4» совершил первый успешный облёт Марса и передал 22 фотографии. Мир впервые увидел поверхность другой планеты - покрытую кратерами, холодную и мёртвую, что развеяло мифы о каналах и цивилизации.

2 декабря 1971 СССР первым сажает на Марс спускаемый аппарат «Марс-3». Он начал передачу сигнала, но прервался через 14,5 секунд - возможно, из-за чудовищной пылевой бури. Его изображение показано на рисунке А.1 Приложения А. В том же году американский «Маринер-9» стал первым искусственным спутником Марса и картографировал планету. Изображение «Маринер-9» и карта Марса, сделанная по его снимкам, показаны на рисунке А.2 Приложения А.

В 1975 году США отправляют две пары: орбитальный аппарат + посадочный модуль «Викинг». «Викинг-1» совершил первую в истории успешную и долгую посадку 20 июля. Позже, в 1976 году аппараты провели биологические эксперименты для поиска жизни (однозначных признаков не нашли), детально изучили почву и климат, передав тысячи снимков.

8. Первые марсоходы

Эра марсоходов началась с запуска американской миссии Mars Pathfinder в 1996 году. Миссия стала технологическим прорывом, впервые реализовав посадку с использованием подушек безопасности. Основной результат - успешная доставка на поверхность первого в истории автоматического самоходного аппарата - марсохода Sojourner. Его изображение показано на рисунке А.3 Приложения А. Это доказало концепцию мобильных исследований

Марса и позволило изучить химический состав пород в районе долины Арес-Валлис.

Следующими марсоходами, достигшими поверхности Марса, стали два идентичных марсохода второго поколения: Mars Exploration Rovers (NASA) - Spirit (запуск 10.06.2003) и Opportunity (запуск 07.07.2003). Их изображения показаны на рисунке А.4 Приложения А. Совершив посадку в январе 2004 года в разных полушариях Марса, они приступили к геологическим исследованиям. Проектная продолжительность их активной работы составляла 90 солов (марсианских суток) с планируемой проходимостью до 100 метров в день. Оба аппарата многократно превысили расчетный срок службы, проработав годы и собрав уникальные данные о прошлой водной активности на планете, что стало одним из самых впечатляющих достижений в истории роботизированных исследований.

9. Автоматизированные лаборатории

В 2005 году состоялся запуск Mars Reconnaissance Orbiter (MRO). Это многофункциональный орбитальный аппарат, оснащенный мощнейшими камерами и спектрометрами. MRO выполнил обширную программу по изучению климата, сезонных изменений атмосферы и детальной геологии. Ключевая задача заключалась в поиске следов гидротермальной активности и жидкой воды в недавнем прошлом. После завершения основной научной программы миссия была расширена: сегодня MRO служит критически важным ретранслятором данных для всех действующих марсоходов и посадочных модулей на поверхности, а также обеспечивает их высокоточную навигацию.

В 2007 году был запущен американский Phoenix - Стационарный посадочный модуль, предназначенный для изучения приполярных областей Марса. Основная научная программа была направлена на решение трех задач: оценка пригодности полярных грунтов для гипотетической жизни, изучение динамики таяния подповерхностного льда и анализ климатических изменений в районе посадки. Неофициальный девиз миссии — «За водой» — был полностью реализован: модуль впервые непосредственно подтвердил наличие водяного льда под тонким слоем грунта. Исследования Phoenix также внесли вклад в понимание радиационной и пылевой обстановки, необходимой для подготовки человеческой экспедиции.

Важным этапом в исследовании Марса стал запуск Mars Science Laboratory (MSL) / Curiosity в 2011 году. Его изображение показано на рисунке А.5 Приложения А. Данная миссия ознаменовала переход к качественно новому этапу изучения поверхности. Марсоход Curiosity представляет собой полноценную автономную химическую лабораторию тяжелого класса, значительно превосходящую своих предшественников (Spirit и Opportunity) по массе, размерам и сложности научного оборудования. Основная цель миссии — оценка потенциальной обитаемости Марса в древности, поиск органических соединений и изучение геологической истории кратера Гейла.

10. Современные миссии

В 2021 году к Марсу одновременно прибыли три миссии: Perseverance (NASA) - самый совершенный марсоход. Ищет следы древней жизни, собирает образцы грунта для будущей доставки на Землю, испытал вертолёт Ingenuity - первый летательный аппарат в атмосфере другой планеты. Тяньвэнь-1 (Китай) - первая полностью успешная миссия Китая. Орбитальный аппарат, посадочная платформа и марсоход Чжужун сделали Китай второй страной, успешно посадившей и управляющей марсоходом. Hope (ОАЭ) - орбитальный аппарат, первая межпланетная миссия арабского мира, изучает климат Марса.

11. Парашют для посадочного модуля

Многие из представленных аппаратов, которые достигли поверхности Марса, использовали систему парашютов для посадки на поверхность планеты. Использование парашюта обусловлено экономией топлива во время торможения в атмосфере планеты. Разработки систем парашютов проходят и в настоящее время.

Одной из таких разработок является парашютная система миссии ExoMars Европейского космического агентства (ESA). Это самая сложная из разработанных парашютных систем для Марса, успешно прошедшая испытания за Полярным кругом. Тест подтвердил её способность безопасно посадить марсоход Rosalind Franklin на поверхность планеты.

Испытания провели с помощью стратосферного гелиевого аэростата, поднявшего спускаемый модуль на высоту 29 км — почти втрое выше полёта пассажирских самолётов. На этой высоте удалось воссоздать условия марсианской атмосферы. Система последовательно раскрыла два крупных парашюта, эффективно замедлив капсулу.

Тестирование на Земле крайне сложно из-за разрежённой марсианской атмосферы (всего 1% от земной). Чтобы точно смоделировать сочетание низкой плотности и высокой скорости, капсулу сбросили с 29 км. В свободном падении она около 20 секунд разгонялась почти до звуковой скорости, после чего раскрылись парашюты.

Испытания прошли 7 июля 2025 года в центре Esrange (Кируна, Швеция) - месте с большим опытом стратосферных запусков с 1970-х годов. Изображение данного парашюта показано на рисунке А.6 Приложения А.

Посадка на Марс - один из самых рискованных этапов миссии. За шесть минут модуль должен сбросить скорость с 21 000 км/ч до мягкого касания, сохранив марсоход. Система включает тепловой щит, два основных парашюта (каждый с вытяжным) и тормозные двигатели.

Первый парашют диаметром 15 м схож с системами Viking (NASA, 1972), но улучшен по опыту Cassini-Huygens. Второй - 35 м - станет крупнейшим, когда-либо развёрнутым на Марсе или за пределами Земли. Изображение этого парашюта показано на рисунке А.6

Приложения А.

Система сертифицирована ещё в 2021 году, однако миссию приостановили. Новые тесты подтвердили сохранность парашютов после долгого хранения. Они сделаны из сверхлёгкого материала (40 г/м² - вдвое легче обычной бумаги). Парашюты изготовлены в Италии, вытяжные механизмы - в Нидерландах, контейнеры - в Чехии. Второй парашют состоит из 800+ м² ткани и более 4 км строп, его укладка занимает около трёх дней.

Компания ThalesAleniaSpace (Франция) курировала испытания как ответственная за сборку парашютной системы. Пока телеметрия анализируется, команда Vorticity изучает данные и высокоскоростную видеосъёмку, чтобы оценить профиль торможения и динамику раскрытия. Будущее миссии ExoMars пока остаётся под вопросом.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Практическая часть моей работы разделилась на несколько этапов:

1) Рассмотреть варианты конфигурации систем посадки космического аппарата;
2) Провести расчеты для этих конфигураций;
3) Проанализировать полученные данные и выбрать наиболее выгодную конфигурацию;
4) Создание эскиза системы посадки космического аппарата.

Варианты посадки космического аппарата на поверхность Марса

Разберем пять вариантов посадки космического аппарата (далее будет упоминаться как КА): только на ЖРД; только на парашюте; на тормозном щите и ЖРД; на тормозном щите, парашюте и ЖРД; на тормозном щите, парашюте и ЖРД; на тормозном щите, тормозном парашюте, основном парашюте и ЖРД.

1. Первый вариант: спуск с орбиты на ЖРД.

Выведем формулу для нахождения скорости, на которую нужно затормозить КА при спуске на поверхность Марса. На этапе спуска вектор скорости изменяет свое направление и не совпадает с направлением силы притяжения КА.

Скорость КА на орбите:

$$v_{\text{кр}} = \sqrt{\frac{\mu_M}{R_M + \text{Н}_{\text{ОРБ}}}} \tag{1}$$

$\text{Н}_{\text{ОРБ}}$ – высота орбиты

R_M – радиус Марса

Кинетическая энергия:

$$T = \frac{M \cdot v_{KP}^2}{2} = \frac{M \cdot \mu_M}{2 \cdot \left(R_{\text{М}} + H_{\text{ОРБ}}\right)} \tag{2}$$

М – масса КА

Изменение потенциальной энергии при посадке на Марс

$$\Pi = E_{\Pi 2} - E_{\Pi 1} = M \cdot \mu_Л \cdot \left[\left(-\frac{1}{R_{\text{М}} + H_{\text{ОРБ}}}\right) - \left(-\frac{1}{R_{\text{М}}}\right)\right] = M \cdot \mu_M \cdot \left(\frac{1}{R_{\text{М}}} - \frac{1}{R_{\text{М}} + H_{\text{ОРБ}}}\right) \tag{3}$$

Работа, которую надо совершить для спуска КА на поверхность Марса, равняется:

$$A = T + \Pi = M \cdot \mu_M \cdot \left(\frac{1}{R_{\text{М}}} - \frac{1}{2 \cdot (R_{\text{М}} + H_{\text{ОРБ}})}\right) \tag{4}$$

При спуске КА ему нужно реализовать кинетическую энергию

$$\frac{M \cdot v_1^{\,2}}{2} = M \cdot \mu_M \cdot \left(\frac{1}{R_{\text{М}}} - \frac{1}{2 \cdot (R_{\text{М}} + H_{\text{ОРБ}})}\right) \tag{5}$$

откуда находим скорость

$$V_1 = V_{кр} \cdot \sqrt{1 + \frac{2 \cdot H_{ОРБ}}{R_M}} = \sqrt{\frac{\mu_M}{R_M + \text{Н}_{\text{ОРБ}}}} \cdot \sqrt{1 + \frac{2 \cdot H_{ОРБ}}{R_M}} = \sqrt{\left(\frac{\mu_M}{R_M + \text{Н}_{\text{ОРБ}}}\right) \cdot \left(1 + \frac{2 \cdot H_{ОРБ}}{R_M}\right)} \tag{6}$$

Это идеальная (мгновенная) скорость, на которую необходимо затормозить КА при посадке. Она достигается в конце активного участка, который продолжается конечное время. На этом участке вектор скорости изменяет свое направление и не совпадает с направлением силы притяжения КА.

Для того, чтобы определить реальные затраты топлива на участке спуска, используется характеристическая скорость

$$\Delta V_X = V_1 + \Delta V_{грав} + \Delta V_{аэр} + \Delta V_{упр} \tag{7}$$

где v_1– мгновенная скорость КА, необходимая для спуска с заданной орбиты (определяется из закона сохранения энергии, без учёта дополнительных потерь);

$\Delta V_{аэр}$ - аэродинамические потери;

$\Delta V_{грав} \approx \int g_{\text{М}} \sin\theta(t) dt$ – дополнительные гравитационные потери на активном участке траектории спуска;

$\Delta v_{упр}$– дополнительные потери на управление КЛА,

$g_{\text{М}}$ - ускорение свободного падения на Марсе;

$\sin\theta(t)$ - изменение синуса угла тангажа на участке спуска.

Аэродинамические потери $\Delta V_{аэр}$ и потери на управление КА $\Delta v_{упр}$ составляют

небольшую величину и ими можно пренебречь.

Приближенно гравитационные потери скорости можно оценить следующим образом (6). Известно, что эти потери пропорциональны ускорению свободного падения и при старте с Земли на низкие орбиты равняются примерно $\Delta v_{грав.Зем} \approx 1200$ *м/с*.

Поэтому для спуска с Марса гравитационные потери скорости можно оценить как

$$\Delta v_{\text{грав}} = \frac{g_{\text{м}}}{g_{\text{зем}}} \Delta v_{грав.Зем} \tag{8}$$

В нашем случае получаем

$$\Delta v_{\text{грав}} = \frac{3{,}71}{9{,}81} \cdot 1200 = 454\ м/с$$

$$\Delta V_X = V_1 + \Delta V_{грав} = \sqrt{\left(\frac{\mu_M}{R_M + \text{Н}_{\text{ОРБ}}}\right) \cdot \left(1 + \frac{2 \cdot H_{ОРБ}}{R_M}\right)} + \Delta V_{грав}$$

Зададим основные исходные данные: $\mu_M = 4{,}283 \cdot 10^{13}$ м^3/с^2 , $R_M = 3{,}4 \cdot 10^6$ м, $H_{ОРБ} = 400$ км. Тогда

$$\Delta V_X = \sqrt{\left(\frac{4{,}283 \cdot 10^{13}}{3{,}4 \cdot 10^6 + 0{,}4 \cdot 10^6}\right) \cdot \left(1 + \frac{2 \cdot 0{,}4 \cdot 10^6}{3{,}4 \cdot 10^6}\right)} + 454 = 4185\ м/с$$

Зная характеристическую скорость Δv_X и массу КА, можно определить затраты топлива на спуск КА на поверхность Марса. Для этого используем формулу Циолковского К.Э.

$$v_X = -J \cdot \ln\left(\frac{M_0}{M_{\text{К}}}\right) = -J \cdot \ln\left(\frac{M_0}{M_0 - M_T}\right) \tag{9}$$

$$M_T = M_0 \cdot \left(1 - e^{\frac{-v_X}{J}}\right) \tag{10}$$

Пусть M_0 = 10000 кг, J = 3400 м/с. Тогда

$$M_T = 10000 \cdot \left(1 - e^{\frac{-4185}{3400}}\right) = 7080\ кг$$

В этом случае процент массы элементов системы посадки составит

$$\Delta\% = \frac{M_{СП}}{M_0} \cdot 100\% = \frac{M_T + M_{ТО} + M_{ДУ}}{M_0} \cdot 100\% = \frac{M_T + k_{ТО} \cdot M_T + k_{ДУ} \cdot F_P}{M_0} \cdot 100\% =$$

$$= \frac{M_T + k_{ТО} \cdot M_T + k_{ДУ} \cdot n_0 \cdot g_0 \cdot M_0}{M_0} \cdot 100\% = \frac{7080 + 0{,}05 \cdot 7080 + 0{,}0015 \cdot 2 \cdot 3{,}71 \cdot 10000}{10000} \cdot 100\% = 75{,}45\%$$

$k_{ТО}$ – коэффициент топлива

$k_{ДУ}$ – коэффициент двигательной установки

M_0 – изначальная масса

2. Второй вариант: спуск с орбиты на парашюте.

Выведем формулу для определения площади парашюта.

При установившемся движении можно положить

$F_T = F_{АЭР}$,

или

$$M \cdot g = c_X \cdot \frac{\rho \cdot v^2}{2} \cdot S_П \tag{11}$$

Здесь $S_П$ - площадь парашюта, c_X -коэффициент лобового аэродинамического сопротивления, ρ – плотность атмосферы, v – скорость.

Если для погашения скорости использовать парашютную систему, то можно использовать формулу (11), разрешив её относительно площади парашюта.

$$S_П = \frac{2 \cdot M \cdot g}{c_X \cdot \rho \cdot v^2} \tag{12}$$

Пусть M_0 = 10000 кг; $c_X = 1{,}6$; $\rho = 0{,}0131\, кг/м^3$; $g = 3{,}71\, м/с^2$.

Полагая скорость КА при расстоянии от поверхности Марса около одного метра равной v = 7,5 м/с (по аналогии со спуском пилотируемого КА «Союз»), получим

$$S_П = \frac{2 \cdot 10000 \cdot 3{,}71}{1{,}6 \cdot 0{,}0131 \cdot 7{,}5^2} = 62935\ м^2$$

В этом случае процент массы элементов системы посадки составит

$$\Delta\% = \frac{M_{СП}}{M_0} \cdot 100\% = \frac{M_П}{M_0} \cdot 100\% = \frac{k_П \cdot M_П}{M_0} \cdot 100\% = \frac{0{,}1 \cdot 62935}{10000} \cdot 100\% = 62{,}935\%$$

Принципы расчетов для остальных вариантов схожи, поэтому распишем универсальные формулы для этих трех случаев. Введем коэффициенты, с помощью которых будем убирать ненужные переменные умножением на «0»:

При установившемся движении (ускорение равно нулю) сила притяжения уравновешивается силой аэродинамического сопротивления атмосферы и можно положить

$$F_T = F_A \tag{13}$$

Или

$$M \cdot g = c_x \frac{\rho \cdot V_0^2}{2} S_{ТЩ}. \tag{14}$$

Здесь

$S_{ТЩ} = \pi \cdot r^2$ - площадь тормозного щита или парашюта. Выражая из (13) скорость падения, получим

$$V_0 = \sqrt{\frac{2M \cdot g}{c_x \cdot \rho \cdot S_{ТЩ}}}. \tag{15}$$

Если мы опускаемся с работающим ракетным двигателем, то в соответствии со вторым законом Ньютона можно записать

$$M \cdot \frac{dV}{dt} = M \cdot g - F_P - k_1 \cdot F_{A_1} - k_2 \cdot F_{A_2} \tag{16}$$

Здесь

$$F_P = n_0 \cdot M_0 \cdot g_0, \tag{17}$$

n_0 – коэффициент начальной тяговооруженности, $S_{П1}$ – площадь парашюта 1, $S_{П2}$ – площадь парашюта 2

Разделим уравнение (16) на переменную массу КА

$$M \cdot \frac{dV}{dt} = M \cdot g - F_P - k_1 \cdot c_x \frac{\rho \cdot V^2}{2} S_{П1} - k_2 \cdot c_x \frac{\rho \cdot V^2}{2} S_{П2} \tag{18}$$

Производную от скорости можно представить в разностном виде

$$\frac{dV}{dt} = \frac{V_{j+1} - V_j}{t_{j+1} - t_j} = \frac{V_{j+1} - V_j}{\Delta t}$$

Здесь (j+1) и (j) – последующий и предыдущий момент времени,

$\Delta t = t_{j+1} - t_j$ - промежуток времени.

Если промежуток времени стремится к нулю, то мы получаем производную.

Запишем (18) в удобном для проведения расчётов разностном виде

$$\frac{V_{j+1} - V_j}{\Delta t} = g_j - n_0 \cdot g_0 \cdot \frac{M_0}{M_j} - k_1 \cdot c_x \frac{\rho_j \cdot V_j^2}{2 \cdot M_j} S_{П1} - k_2 \cdot c_x \frac{\rho_j \cdot V_j^2}{2 \cdot M_j} S_{П2},$$

Или

$$V_{j+1} = V_j - \left[n_0 \cdot g_0 \cdot \frac{M_0}{M_j} + k_1 \cdot c_x \frac{\rho_j \cdot V_j^2}{2 \cdot M_j} S_{П1} + k_2 \cdot c_x \frac{\rho_j \cdot V_j^2}{2 \cdot M_j} S_{П2} - g_j \right] \cdot \Delta t \tag{19}$$

В виду малого изменения высоты полёта можно положить $g_j = g_0 = const$.

Высоту, которую пролетает КА определим из соотношения

$$V = -\frac{dh}{dt}, \quad или \quad V_{j+1} = -\frac{h_{j+1} - h_j}{\Delta t}$$

Откуда

$$h_{j+1} = h_j - V_{j+1} \cdot \Delta t \tag{20}$$

При работающем ракетном двигателе расходуется топливо, следовательно, масса КА уменьшается.

$$M_{j+1} = M_j - \Delta M_T \tag{21}$$

А масса израсходованного топлива увеличивается

$$M_{T,j+1} = M_{T,j} + \Delta M_T \quad (23)$$

Для определения ΔM_T используем известное соотношение для реактивной силы

$$F_P = \dot{m}_T \cdot w = \frac{dM_T}{dt} \cdot w \approx \frac{\Delta M_T}{\Delta t} \cdot w \tag{22}$$

Здесь w– удельный импульс двигателя (определяется выбранным топливом из справочников), $\dot{m}_T$ - секундный массовый расход топлива в двигателе.

Из (24) получаем

$$\Delta M_T = \frac{F_P \cdot \Delta t}{w} = \frac{n_0 \cdot M_0 \cdot g_0 \cdot \Delta t}{w} \tag{23}$$

Здесь M_0 – начальная масса КА, g_0–ускорение силы тяжести на поверхности планеты.

Текущее время определяем из соотношения

$$t_{j+1} = t_j + \Delta t \tag{24}$$

Плотность атмосферы определяем из соотношения

$$\rho_j = \rho_0 \cdot \left(1 - \frac{h_j}{46334}\right)^{3,3333} \tag{25}$$

Третий вариант: спуск с орбиты с использованием на первом этапе тормозного щита и на втором этапе ЖРД. Обозначим через H_1 высоту, на которой сбрасывается тормозной аэродинамический щит и включается ЖРД.

В данном случае аэродинамических сил парашютов нет, так что коэффициенты k_1 и k_2 равны 0.

Используя формулы (13-25), была написана программа на языке программирования

Python для вычисления нужных данных:

```
H3= 11000
DT=0.01
W1=3400
AM0=10000
GO=3.71
CX=0.89
PLO=0.0131
AKT=1.05
ADU=0.001
AKP=0.2
AKS=5.
CX0=0.89
print('      AM0            CX            W1')
print(AM0,CX,W1)
print('      AKT            ADU            AKP          AKS')
print(AKT,ADU,AKP,AKS)
print('**********************************************')
for i in range(1,3):
    H3=H3-1000
    ST0=0
    for j in range(1,4):
        ST0=ST0+25
        H1=H3
        AM1=AM0
        PL1=PLO*((1-H1/46334)**3.3333)
        V3=((2*AM1*GO)/(CX0*PL1*ST0))**0.5
        AM1=AM0-ST0*AKS
        FP=97600
        print('      PL1            V3')
        print(PL1,V3)
        for i in range(1,75000):
            FP=FP+1
            DMT=FP*DT/W1
            AMT1=0
```

```
T1=0
V1=V3
AM1=AM0-ST0*AKS
H1=H3
V2=V1-(FP/AM1-GO)*DT
while (V2-0.5) > 0:
    V2=V1-(FP/AM1-GO)*DT
    A2=(V1-V2)/DT
    H2=H1-V2*DT
    AM2=AM1-DMT
    AMT2=AMT1+DMT
    T2=T1+DT
    V1=V2
    H1=H2
    AMT1=AMT2
    AM1=AM2
    T1=T2
    PL1=PLO*((1.-H2/46334.)**3.333)
if (H2 - 0.5) <= 0 and (H2) >= 0:
    print('      H3            ST0            FP')
    print(H3,ST0,FP)
    print('      T2            V2            H2')
    print(T2,V2,H2)
    AMC=AKT*AMT2+ADU*FP+AKS*ST0
    DAMT2=AMC*100./AM0
    print('      AM1           AMT2          AMC         DAMT2')
    print(AM1,AMT2,AMC,DAMT2)
    print('#####################################################')
```

С помощью данной программы получены данные, необходимые для построения графиков.

На рис. 1 показаны затраты топлива при разных значениях площади тормозного щита и высоты H_1.На рис. 2 показан процент массы элементов системы посадки.

$$\Delta\% = \frac{M_{СП}}{M_0}\cdot 100\% = \frac{M_{ТЩ}+M_T+M_{ТО}+M_{ДУ}}{M_0}\cdot 100\% = \frac{k_{ТЩ}\cdot S_{ТЩ}+M_T+k_{ТО}\cdot M_T+k_{ДУ}\cdot F_P}{M_0}\cdot 100\% =$$

$$= \frac{k_{ТЩ} \cdot S_{ТЩ} + M_T + k_{ТО} \cdot M_T + k_{ДУ} \cdot n_0 \cdot g_0 \cdot M_0}{M_0} \cdot 100\%$$

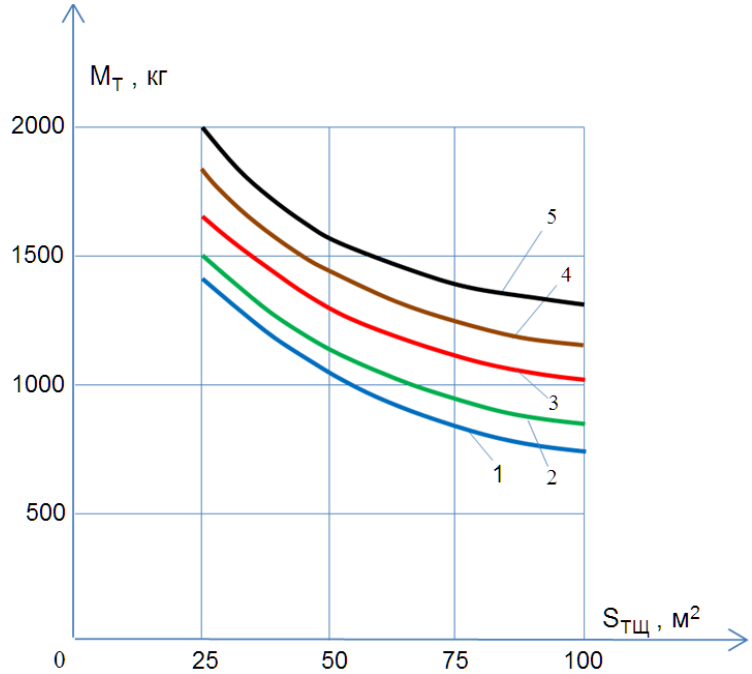


Рис. 1. Затраты топлива при спуске. 1 – H_1 = 1000 м; 2 – H_1 = 2000 м; 3 – H_1 = 4000 м;4 – H_1 = 6000 м; 5 – H_1 = 8000 м.

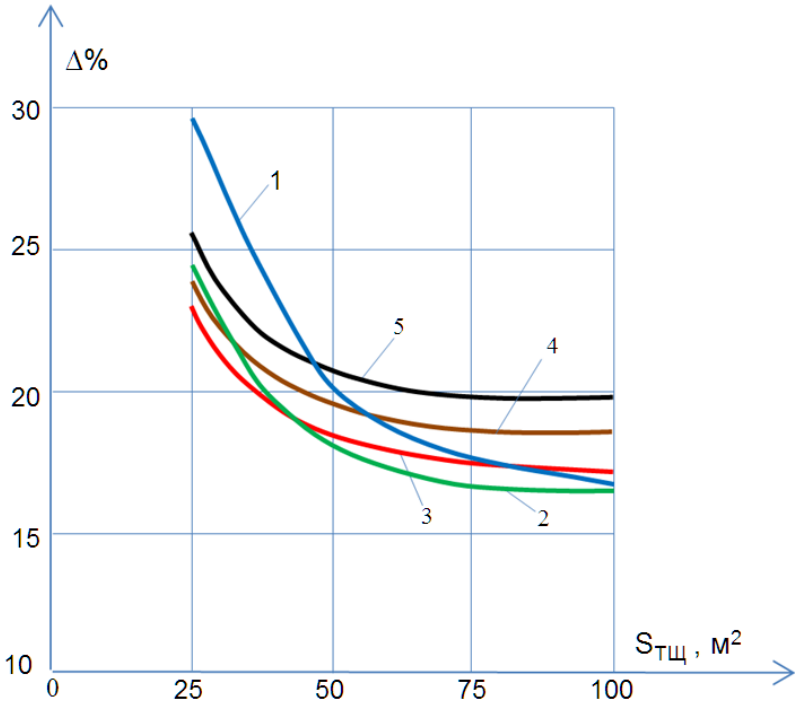


Рис. 2. Процент массы системы посадки при спуске. 1 – H_1 = 1000 м; 2 – H_1 = 2000 м; 3 – H_1 = 4000 м;4 – H_1 = 6000 м; 5 – H_1 = 8000 м.

Четвёртый вариант: спуск с орбиты с использованием на первом этапе тормозного щита, на втором этапе тормозного парашюта и на третьем этапе ЖРД. Обозначим через H_1

высоту, на которой сбрасывается тормозной аэродинамический щит и раскрывается тормозной парашют и через H_2 высоту сброса тормозного парашюта и включения ЖРД. В этом варианте для всех случаев высота H_1 = 9000 м и площадь тормозного аэродинамического щита равнялась $S_{ТЩ}$ = 50 м^2.

Аналогично третьему варианту, используем формулы (13-26), учитывая, что коэффициент k_1 будет равен 1 до того момента, когда высота достигнет отметки H_2, после чего парашют отстегнется и k_1 будет снова равен 0. Коэффициент k_2 будет равен 0 все время, т.к. парашют один.

Также составляем программу и получаем нужные данные:

```
H3=9000
DT=0.01
W1=3400.
AM0=10000
GO=3.71
PLO=0.0131
AKT=1.05
ADU=0.001
AKP=0.2
AKS=5
CX0=0.89
CX=1.6
print('      AM0          CX0          CX          W1')
print(AM0,CX0,CX,W1)
print('      AKT          ADU          AKP          AKS')
print(AKT,ADU,AKP,AKS)
H4=500
ST0=50
ST1=500
PL1=PLO*((1-H3/46334.)**3.3333)
V3=((2*AM0*GO)/(CX0*PL1*ST0))**0.5
AM1=AM0-ST0*AKS
print('<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<')
print('      AM1          H3          H4          ST0          PL1          V3 ')
print(AM1,H3,H4,ST0,PL1,V3)
T1=0.
```

```
V1=V3
H1=H3
while (H1-H4) > 0:
    V1=V1-(-GO+(CX*PL1*ST1*V1*V1)/(2.*AM1))*DT
    H1=H1-V1*DT
    T1=T1+DT
    PL1=PLO*(1.-H1/46334.)**3.333
print('*************************************************')
print('      T1          V1          H1          PL1          ST1')
print(T1,V1,H1,PL1,ST1)
AM1=AM1-AKP*ST1
FP=90000
for i in range(1,200000):
    FP=FP+1
    DMT=FP*DT/W1
    AM2=AM1
    V2=V1
    T2=0
    H2=H4
    AMT2=0
    fl = 0
    while (V2-0.5)>0 and fl == 0:
        fl = 0
        V2=V2-(FP/AM2-GO)*DT
        H2=H2-V2*DT
        AM2=AM2-DMT
        AMT2=AMT2+DMT
        T2=T2+DT
        PL1=PLO*((1-H2/46334)**3.333)
    if (H2-0.5) >= 0:
        print('*************************************************')
        print('      T2          V2          H2          FP')
        print(T2,V2,H2,FP)
        AMC=AKT*AMT2+ADU*FP+AKS*ST0+AKP*ST1
        DAM2=AMC*100/AM0
```

```
        print('      AM2          AMT2         AMC        DAM2')
        print(AM2,AMT2,AMC,DAM2)
        break
    else:
        if H2 >= 0:
            print('*********************************************')
            print('      T2           V2           H2           FP')
            print(T2,V2,H2,FP)
            AMC=AKT*AMT2+ADU*FP+AKS*ST0+AKP*ST1
            DAM2=AMC*100/AM0
            print('      AM2          AMT2         AMC        DAM2')
            print(AM2,AMT2,AMC,DAM2)
            break
```

На рис. 3 показаны затраты топлива при разных значениях площади парашюта и высоты H_2. На рис. 4 показан процент массы элементов системы посадки.

$$\Delta\% = \frac{M_{СП}}{M_0} \cdot 100\% = \frac{M_{ТЩ} + M_{P1} + M_T + M_{TO} + M_{ДУ}}{M_0} \cdot 100\% =$$

$$= \frac{k_{ТЩ} \cdot S_{ТЩ} + k_P \cdot S_{П1} + M_T + k_{TO} \cdot M_T + k_{ДУ} \cdot n_0 \cdot g_0 \cdot M_0}{M_0} \cdot 100\%$$

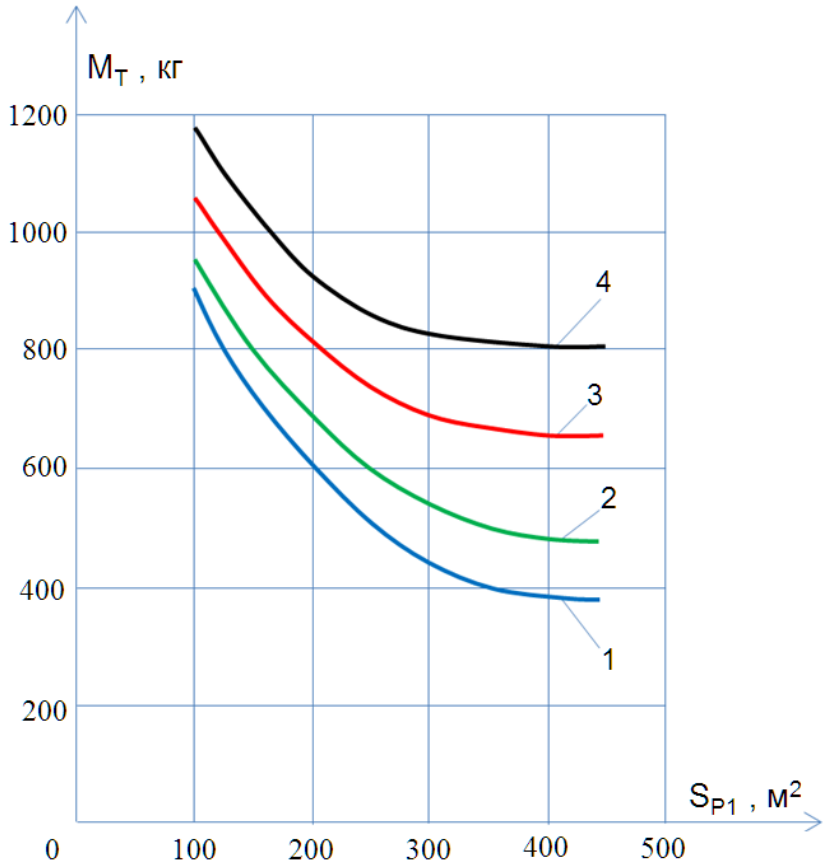


Рис. 3. Затраты топлива при спуске.

1 – H_2 = 500 м; 2 – H_2 = 1000 м; 3 – H_2 = 2000 м; 4 – H_2 = 3000 м.

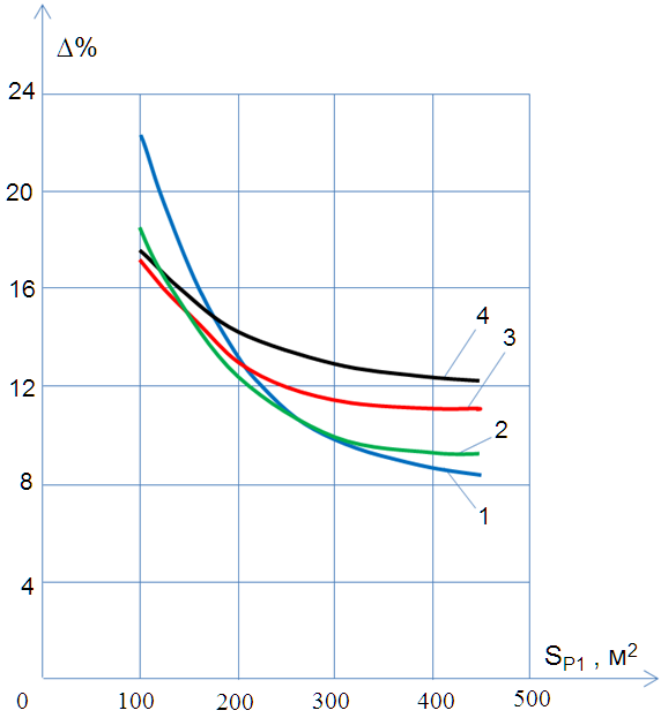


Рис. 4. Процент массы системы посадки при спуске.

1 – H_2 = 500 м; 2 – H_2 = 1000 м; 3 – H_2 = 2000 м; 4 – H_2 = 3000 м.

Пятый вариант: спуск с орбиты с использованием на первом этапе тормозного щита, на втором этапе тормозного парашюта, на третьем этапе основного парашюта и на четвёртом этапе ЖРД. Обозначим через H_1 высоту, на которой сбрасывается тормозной аэродинамический щит и раскрывается тормозной парашют и через H_2 высоту сброса тормозного парашюта и раскрытия основного парашюта и через H_3 высоту сброса основного парашюта и включения ЖРД. В этом варианте для всех случаев высота

H_1 = 9000 м, высота H_2 = 6000 м, площадь тормозного аэродинамического щита $S_{ТЩ}$ = 50 м^2, площадь тормозного парашюта S_{P1} = 200 м^2.

Также используем формулы (13-26), учитывая, что коэффициент k_1 будет равен 1 до того момента, когда высота достигнет отметки H_2, после чего тормозной парашют отстегнется и k_1 будет снова равен 0, а k_2 станет равен 1, так как раскроется основной парашют. Коэффициент k_2 снова станет равен 0 тогда, когда высота станет равна H_3.

Составляем программу на Python и находим данные:

```
H3=9000
H4=2000
H6=6500
DT=0.01
W1=3400
AM0=10000
GO=3.71
PLO=0.0131
AKT=1.05
ADU=0.001
AKP=0.2
AKS=5
ST0=100
ST1=500
ST2=1000
CX0=0.89
CX=1.6
H2 = 10000000
print('      AM0            W1            ST0')
print(AM0,W1,ST0)
print('      H3             H4            H6')
print(H3,H4,H6)
```

```
print('      PLO          CX0          CX          DT')
print(PLO,CX0,CX,DT)
H1=H3
T1=0
T3=0.5
H5=H3-H4
H7=H5-H6
AM1=AM0
AMT1=0.
PL1=PLO*(1.-H1/46334.)**3.333
V1=(2*AM1*GO/(CX0*PL1*ST0))**0.5

print('      V1             PL1')
print(V1,PL1)
AM1=AM0-ST0*AKS
B1=1
while (H2-H5)>0:
   V2=V1-(-GO+(CX*PL1*ST1*V1*V1)/(2.*AM1))*DT
   A2=(V1-V2)/DT
   ANX1=1.+A2/GO
   ANX2=(A2+GO)/9.81
   ANX3=(CX*PL1*ST1*V1*V1)/(2.*AM1*9.81)
   FA1=CX*PL1*ST1*V1*V1/2.
   FA2=CX*PL1*V1*V1/2.
   if B1>0:
      print('      ANX1       ANX2          ANX3          ST1          FA1          FA2')
      print(ANX1,ANX2,ANX3,ST1,FA1,FA2)
      print('      T2          V2          ANX1          ANX3          H2          PL1')
      B1=-1
   else:
      H2=H1-V2*DT
      T2=T1+DT
      V1=V2
      T1=T2
      PL1=PLO*(1-H2/46334)**3.333
```

```
        if (H2-H5) > 0:
            H1=H2
            if (T2-T3)>=0:
                print(T2,V2,ANX1,ANX3,H2,PL1)
                T3=T3+0.5
print( '*********************************************')
H1=H2
print('      T2          V2          ANX1          ANX3          H2          PL1')
print(T2,V2,ANX1,ANX3,H2,PL1)
print('*********************************************')
AM1=AM1-AKP*ST1
AM3=AM1
B1=1
while (H2-H7)>0:
    V2=V1-(-GO+(CX*PL1*ST2*V1*V1)/(2.*AM1))*DT
    A2=(V1-V2)/DT
    ANX1=1+A2/GO
    ANX2=(A2+GO)/9.81
    ANX3=(CX*PL1*ST2*V1*V1)/(2.*AM1*9.81)
    FA1=CX*PL1*ST2*V1*V1/2.
    FA2=CX*PL1*V1*V1/2.
    if B1>0:
        print('      ANX1        ANX2          ANX3          ST2          FA1          FA2')
        print(ANX1,ANX2,ANX3,ST2,FA1,FA2)
        print('      T2          V2          ANX1          ANX3          H2          PL1')
        B1=-1
    else:
        H2=H1-V2*DT
        T2=T1+DT
        V1=V2
        T1=T2
        PL1=PLO*(1.-H2/46334.)**3.333
        if (H2-H7) > 0:
            H1=H2
            if (T2-T3) >= 0:
```

```
            print(T2,V2,ANX1,ANX3,H2,PL1)
            T3=T3+4

print('*************************************************')
H1=H2
print('      T2          V2          ANX1          ANX3          H2          PL1')
print(T2,V2,ANX1,ANX3,H2,PL1)
print('*************************************************')
H11=H2
FP=36000.
V4=V2
for i in range(1,200000):
   FP=FP+50000
   DMT=FP*DT/W1
   AM1=AM3
   V1=V4
   H1=H11
   AMT1=0
   T1=0
   while (V2-0.5) > 0:
      V2=V1-(FP/AM1-GO)*DT
      A2=(V1-V2)/DT
      ANX1=1.+A2/GO
      ANX2=(A2+GO)/9.81
      H2=H1-V2*DT
      AM2=AM1-DMT
      AMT2=AMT1+DMT
      T2=T1+DT
      V1=V2
      H1=H2
      AMT1=AMT2
      AM1=AM2
      T1=T2
      PL1=PLO*((1.-H2/46334.)**3.333)
   if (H2-0.5)<0:
```

```
    if H2 >= 0:
        print('*********************************************')
        print('*********************************************')
        print('      H5          ST0          FP')
        print(H5,ST0,FP)
        print('      T2          V2          H2          ANX1          ANX2')
        print(T2,V2,H2,ANX1,ANX2)
        AMC=AKT*AMT2+ADU*FP+AKS*ST0+AKP*(ST1+ST2)
        DAMT2=AMC*100./AM0
        print('      AM1          AMT2          AMC          DAMT2')
        print(AM1,AMT2,AMC,DAMT2)
        break
    else:
        print('*********************************************')
        print('*********************************************')
        print('      H5          ST0          FP')
        print(H5,ST0,FP)
        print('      T2          V2          H2          ANX1          ANX2')
        print(T2,V2,H2,ANX1,ANX2)
        AMC=AKT*AMT2+ADU*FP+AKS*ST0+AKP*(ST1+ST2)
        DAMT2=AMC*100./AM0
        print('      AM1          AMT2          AMC          DAMT2')
        print(AM1,AMT2,AMC,DAMT2)
        break
```

На рис. 5 показаны затраты топлива при разных значениях площади основного парашюта и высоты H_3. На рис. 6 показан процент массы элементов системы посадки.

$$\Delta\% = \frac{M_{СП}}{M_0} \cdot 100\% = \frac{M_{ТЩ} + M_{П1} + M_{П2} + M_Т + M_{ТО} + M_{ДУ}}{M_0} \cdot 100\% =$$

$$= \frac{k_{ТЩ} \cdot S_{ТЩ} + k_П \cdot S_{П1} + k_П \cdot S_{П2} + M_Т + k_{ТО} \cdot M_Т + k_{ДУ} \cdot n_0 \cdot g_0 \cdot M_0}{M_0} \cdot 100\%$$

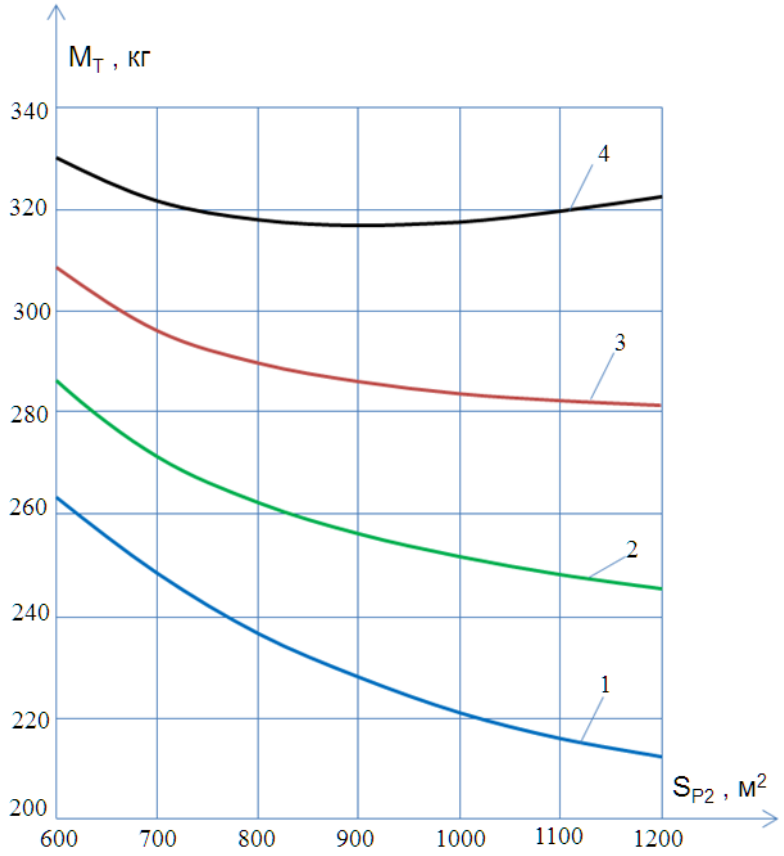


Рис. 5. Затраты топлива при спуске.

1 – $H_З$ = 200 м; 2 – $H_З$ = 300 м; 3 – $H_З$ = 400 м; 4 – $H_З$ = 500 м.

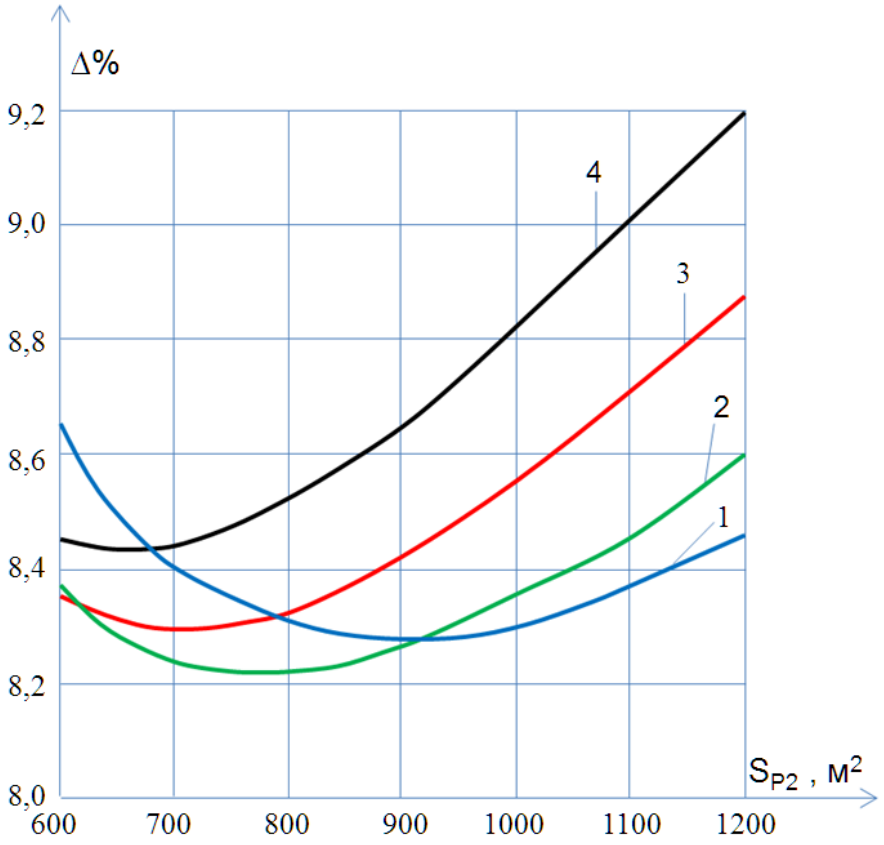


Рис. 6. Процент массы системы посадки при спуске.

1 – H_3 = 200 м; 2 – H_3 = 300 м; 3 – H_3 = 400 м; 4 – H_3 = 500 м.

На графиках видно, что этот вариант является наиболее выгодным, так как затраты топлива при спуске минимальные, как и процент массы системы. Следовательно, самой выгодной конфигурацией является вариант со спуском с орбиты с использованием на первом этапе тормозного щита, на втором этапе тормозного парашюта, на третьем этапе основного парашюта и на четвёртом этапе ЖРД.

Разработка эскиза

На рис.7 показан эскиз системы посадки КА.

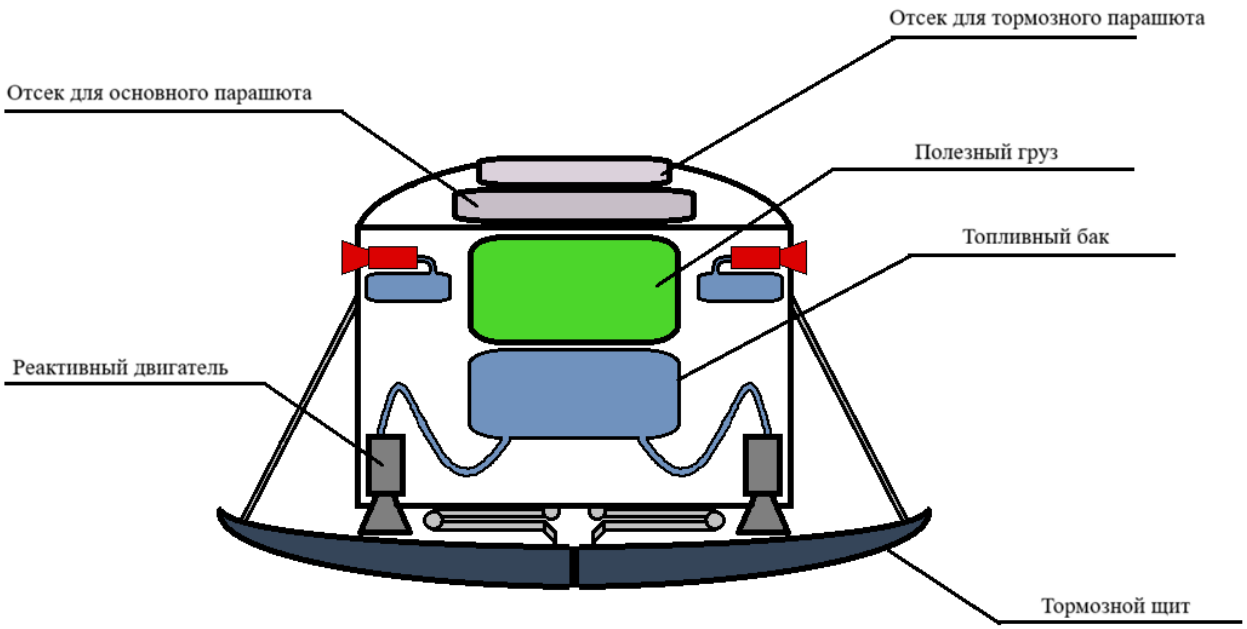


Рис.7. Эскиз системы посадки КА

ЗАКЛЮЧЕНИЕ

В ходе выполнения этой работы мне удалось достичь основных целей. Благодаря этой работе я смог многое узнать о космических миссиях, связанных с изучением Марса.

Применяя знания по физике и программированию, подкрепленные консультациями с моим научным руководителем, мне удалось определить наиболее выгодный вариант посадочного модуля, провести расчеты характеристик пяти вариантов комплектации посадочного модуля КА, который сможет в целости и безопасности доставить груз на поверхность Марса.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%80%D1%81
2. https://new-science.ru/exomars-uspeshnye-ispytaniya-samoj-slozhnoj-parashjutnoj-sistemy-dlya-marsa/
3. Пилотируемая экспедиция на Марс./ Под ред. А.С. Коротеева.- М.: Российская академия космонавтики им. К.Э. Циолковского, 2006, 320 с., илл.

ПРИЛОЖЕНИЕ А

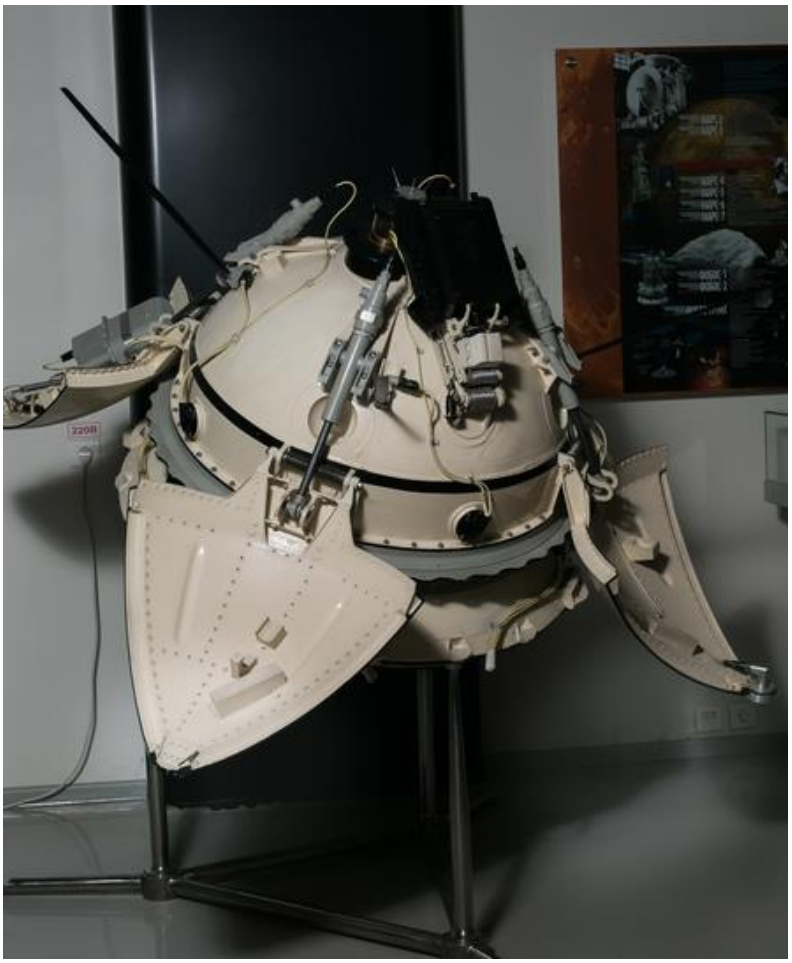


Рис. А.1. Космический аппарат Марс-3

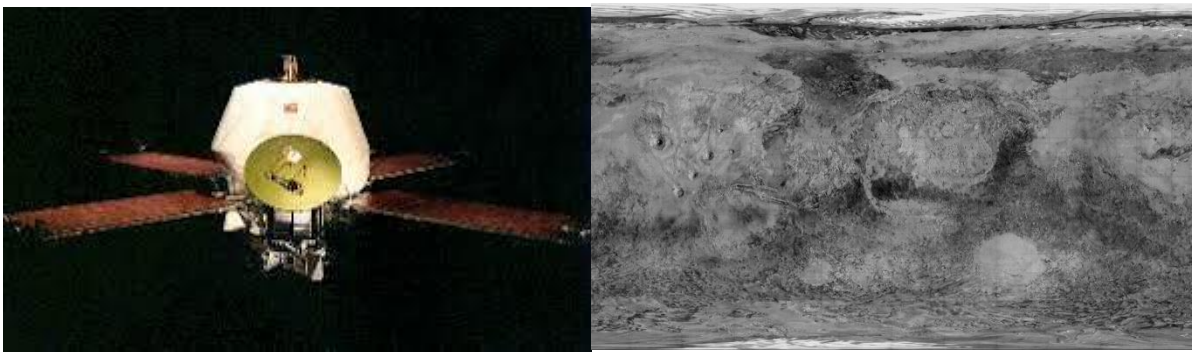


Рис. А.2. Маринер-9 и карта Марса, сделанная на основе его снимков

Рис. А.3. Марсоход Sojourner.

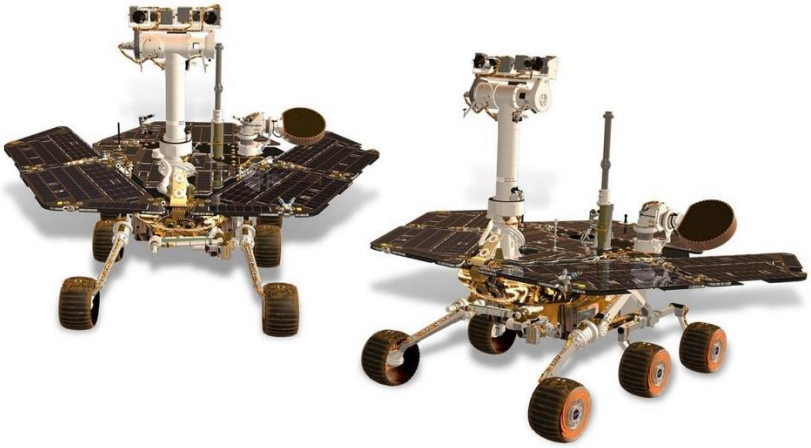


Рис. А.4. Марсоходы Spirit и Opportunity

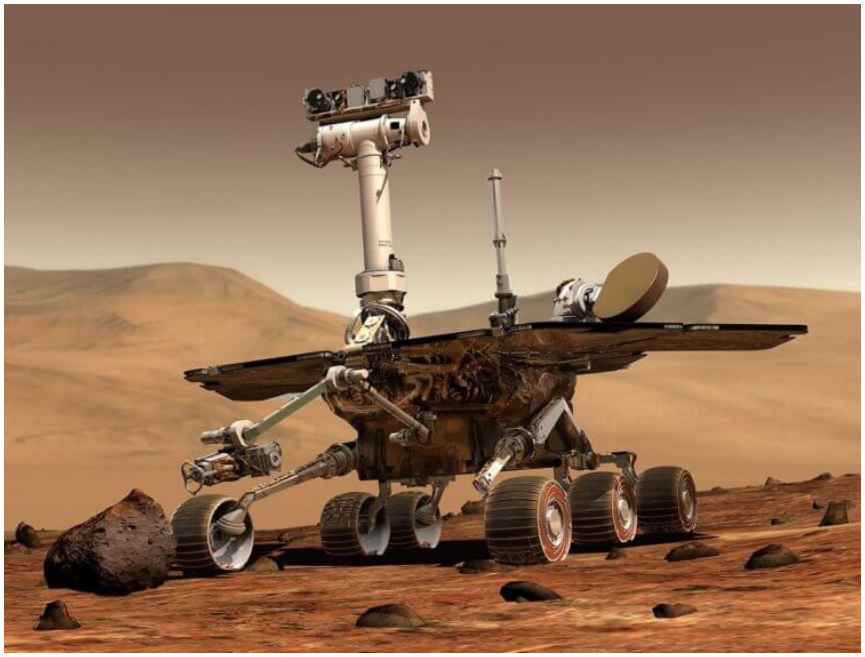


Рис. А.5. Марсоход Curiosity

Рис. А.6. Парашют миссии ExoMars

ПРОТОТИПИРОВАНИЕ ПРИБОРА ДЛЯ УДАЛЕННОГО УПРАВЛЕНИЯ ДАЧНЫМ УЧАСТКОМ С ПОМОЩЬЮ TELEGRAM-БОТА

Качка Мария Даниловна

Магадан г, МАОУ «Гимназия №30», 10 класс

Секция: Компьютерные системы

Научный руководитель: Минитаева Алина Мажитовна, к.т.н., доцент каф. ИУ6 МГТУ им. Н.Э. Баумана

АННОТАЦИЯ

В данной работе рассмотрено создание прототипа прибора для удаленного управления дачным участком с помощью Telegram-бота. Объектами автоматизации на дачном участке выступают дачный домик, теплица, парник, въездные ворота, уличное освещение.

Цель работы: сконструировать аппаратно-программное устройство на базе микроконтроллерной платы Ардуино, передающие значение с датчиков (температуры, влажности, освещения) на мобильный телефон, а также удаленно управлять освещением, обогревом, проветриванием в дачном домике, теплице, парнике, включать и выключать уличное освещение, открывать и закрывать ворота.

Прототип строится на базе микроконтроллера Ардуино, функционал прибора включает в себя получение данных с датчиков в реальном времени, и управление устройствами в доме и теплице, такими как нагреватели, освещение и проветривание, открытие и закрытие ворот, включение и выключение уличного освещения осуществляется удаленно с помощью разработанного Telegram-бота. Результатом работы стало созданный прототип устройства, демонстрирующий возможность удаленного контроля и управления объектами дачного участка.

ВВЕДЕНИЕ

По данным ВЦИОМ[1] дача или земельный участок в собственности или аренде есть у половины граждан (52%) России. Большинство дачников (66%) используют свои участки для выращивания сельхозпродуктов для семейного потребления.

Чаще всего на дачах сажают следующие овощи:

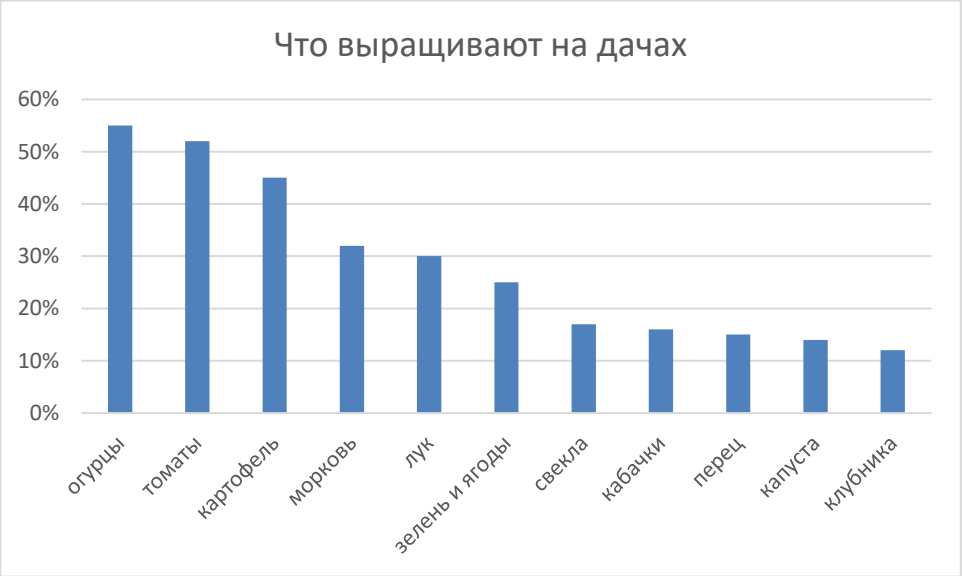


Рис. 1. Диаграмма

Для получения хорошего урожая требуется постоянный уход за растениями, а следовательно постоянное присутствие людей на даче. Но по данным ВЦИОМ постоянно на даче летом проживает – 28% дачников, несколько раз в неделю приезжают – 30% дачников, а несколько раз в месяц – 26%.

Большинство субъектов Российской Федерации (70-80%) относятся к зоне рискованного земледелия. Магаданская область также относится к зоне рискового земледелия. Тем не менее большое количество жителей г. Магадана покупают земельные участки и выращивают овощи и фрукты. Интерес жителей можно судить по большому количеству групп в WhatsApp и Telergram мессенджерах: «Фазенда», «Сад и огород», «Магаданская дача» и другие. Один из частых вопросов какой задают в группах, особенно осень и весной: «Какая температура воздуха на даче?». Нужно ли ехать укрывать грядки, или может нужно открыть парники или проветрить теплицы?

Следовательно, возникает вопрос как быстро узнать температуру в теплице и принять решение, нужно ли ехать спасать урожай или нет.

С другой стороны, в холодное время года, дачники также посещают свои дачные участки, и всегда хочется приехать на дачу, чтобы в домике было тепло, а на кухне был горячий чай.

---

https://wciom.ru/analytical-reviews/analiticheskii-obzor/sezon-u-dachi

Таким образом возникает вопрос каким образом удаленно контролировать объекты дачного участка, такие как теплица, парники, дом и другие объекты.

Основная цель работы: сконструировать аппаратно-программное устройство на базе микроконтроллерной платы Ардуино, передающие значение с датчиков (температуры, влажности, освещения) на мобильный телефон, а также удаленно управлять освещением, обогревом, в дачном домике, теплице, парнике, включать и выключать уличное освещение, открывать и закрывать ворота с минимальными денежными затратами.

Для достижения поставленной цели, нужно было решить ряд задач:

1. Познакомиться с существующими устройствами удаленного контроля дачного участка.
2. Спроектировать аппаратную часть устройства на базе микроконтроллера.
3. Написать программу для управления контроллером.
4. Написать Telegram-бот
5. Сконструировать модель устройства

АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Комплексного решения по автоматизации дачного участка, где автоматизируются все объекты дачи, в сети Интернет не были найдены. Но существует несколько компаний, которые предлагают решения по автоматизации дачного домика (система умный дом). В таблице 1 представлен описание базовой комплектации и стоимость.

Таблица 1. Сравнительная таблица систем «Умный дом»

Компания	Функционал	Стоимость
HouseClever[2]	Освещение + электрика "Умные выключатели" Управление теплыми полами Охранно-пожарная сигнализация Управление шторами Защита от протечек Сценарии и режимы работы Оповещение о событиях Управление с мобильного телефона или планшета	От 90.000 рублей
Дизайн студия Арт-Дом[3]	управление светом, шторами, нагрузками; возможность создания сценариев и работа по таймеру, датчикам	От 110.000 рублей
ELX-Device[4]	управление освещением; управление светодиодными лентами; управление розетками; управление кондиционером система защиты от протечки воды (ГВС и ХВС); управление шторами; датчики движения; управление теплым полом; управление радиаторам отопления; климат-контроль дополнительные группы освещения; бесплатное программное обеспечение Android & iOS мастер выключатель.	От 99.000 рублей

На рынке также существуют множество отдельных решений для дачных теплиц. Сравнительный анализ функционала и стоимости существующих аппаратных и программных решений «Умная теплица» (ОВЕН, SIMATIC S7, Priva Connext, АСУ МТ "Климат" от НПО «Автоматика») показал, что функционально системы практически идентичны. Они все

[2] https://umnyj-doms.ru/index/umnyj_dom_proizvoditel_kompanija/0-4

[3] https://art-dom.ru/ceny-na-oborudovanie-umnogo-doma-100-kvm

[4] https://um-d.ru/prices

отслеживают состояние датчиков влажности, освещенности, температуры, содержания углекислого газа, подают сигнал о выходе за пределы допустимых значений. Самое дешевое решение предлагает российская компания ОВЕН – от 300 тысяч рублей.

Для удаленного управления парниками существует несколько популярных моделей устройств для удаленного измерения температуры. Мы изучили характеристики приборов и составили таблицу 2.

Таблица 2. Сравнительная таблица устройств для удаленного контроля температур

	Alonio / Gsm термометр T2	Система GSM контроля температуры EctoControl "Температура"	GSM термодатчик ELEUS RC-312 для контроля температуры через сотовый телефон
Внешний вид			
Цена в рублях	8 437	16 450	10300
Описание	GSM термометр Alonio T2 контролирует два важнейших параметра: температуру и наличие электричества. Во всех критических ситуациях происходит отправка СМС на мобильный телефон владельца: при отключении электричества, при падении температуры ниже критического уровня, при восстановлении электропитания, при повышении температуры выше критического уровня. GSM термометр Alonio T2 будет полезен дачникам, владельцам домов и	Система GSM контроля температуры EctoControl "Температура" (Комплектация 2) с беспроводными датчиками. Основные функции: Контроль температуры воздуха (2 датчика) Контроль наличия сети 220В Оповещение о любых критических изменениях по SMS или звонком на мобильный телефон Возможность подключения дополнительных датчиков Отправка данных всех датчиков по запросу Встроенные реле (3шт) для управление	Модуль RC-312 позволяет дистанционно контролировать температуру объекта с использованием сети GSM. В модуль можно прописать до 4 температурных меток, при достижении которых модуль будет оповещать до 9 абонентов. Оповещение может осуществляться с помощью вызова или SMS. Также с сотового телефона можно дистанционно управлять подключенным к модулю электрическим и электронным оборудованием и программировать модуль SMS командами

	коттеджей, для постоянного контроля систем отопления в зимнее время.	электроприборами с помощью sms Поддерживает беспроводные датчики - встроенная радиосеть Работа с протоколом обмена RS-485 (Modbus)В комплект входит GSM блок EctoControl с функцией контроля сети 2 беспроводных датчика температуры Мощная выносная gsm-антенна Адаптер питания и аккумулятор (до 24 часов)	

Таким образом, чтобы автоматизировать дачный дом, теплицу и парник потребуются не менее 500.000 рублей.

ПОШАГОВАЯ РЕАЛИЗАЦИЯ ПРОЕКТА

ОБЩИЕ ТРЕБОВАНИЯ К АППАРАТНО-ПРОГРАММНОМУ УСТРЙОСТВУ ПРИБОРА

На основе ГОСТ 34.602-2020 «Техническое задание на создание автоматизированной системы» составим формализованное описание технического задания:

1. Общие сведения.
   1.1. Название продукта разработки (проектирования).
   «Прибор для для удаленного управления дачным участком с помощью Telegram-бота»
   1.2. Назначение продукта.
   - мониторинг микроклимата дачного участка;
   - удаленное управление освещением, проветриванием и обогрева теплицы;
   - удаленное управлением освещением, электрокамином, умной розеткой (чайником) дачного домика;
   - удаленное включение/выключение уличного освещения;
   - удаленное открытие закрытие ворот
2. Характеристика области применения продукта.
   2.1. Процессы и структуры, в которых предполагается использование продукта разработки.
   Владельцы дачных участков.

2.2. Характеристика персонала (количество, квалификация, степень готовности)

Пользователь:

- базовые навыки работы с компьютером
- умение пользоваться Telegram-бот.

3. Требования к продукту разработки.

3.1. Требования к продукту в целом.

Требуется спроектировать и разработать удаленную систему управления на основе микроконтроллера, включающую в себя:

- программное обеспечение для мониторинга температуры и освещенности;
- программное обеспечение для удаленного включения подсветки, обогрева и проветривания в теплице;
- программное обеспечение для удаленного включения обогрева и освещения дачного участка;
- программное обеспечение для управления въездными воротами;
- программное обеспечение для управления уличным освещением.

3.2. Аппаратные требования.

Серверная часть (минимальные требования):

- любой компьютер с USB-портом

Рабочая станция (минимальные требования):

- любой смартфон

3.3 Указание системного программного обеспечения (операционные системы, браузеры, программные платформы и т.п.).

Для севера:

- операционная систем – MS Wndows 7 или выше;
- интерпретатор языка программирования Python 3.

Для смартфона:

- мессенджер Telegram.

3.4 Указание программного обеспечения, используемого для реализации.

- операционная система MS Windows 10;
- Microsoft Office 2019
- язык программирования Python 3;
- Arduino IDE;
- Notepad++;
- Fritzing;

- Draw.io;
- OpenScad.

3.5. Форматы входных и выходных данных

Формат входных и выходных данных состоят из переменных строкового значения.

3.6. Источники данных их ввода в систему (программу), порядок вывода и хранения.

Источником данных являются датчики, подключенные к микроконтроллеру Ардуино, которые передаются на сервер, путем передачи через USB-шнур.

Вывод данных осуществляется путем выбора соответствующей кнопки в telegram-бот.

4. Требования к пользовательскому интерфейсу.

4.1. Общая характеристика пользовательского интерфейса

Пользовательский интерфейс управления имеет:

- меню;
- информация о текущей температуре, освещенности;
- информация о состоянии нагрузки (включен ли обогрев, свет, открыты/закрыты теплица, парник, ворота);
- элементы управления системами обогрева, освещения, проветривания.

5. Требования к документированию.

Не предусмотрено

АРХИТЕКТУРА ПРИБОРА

Общая архитектура устройства представлена на рисунке 2.

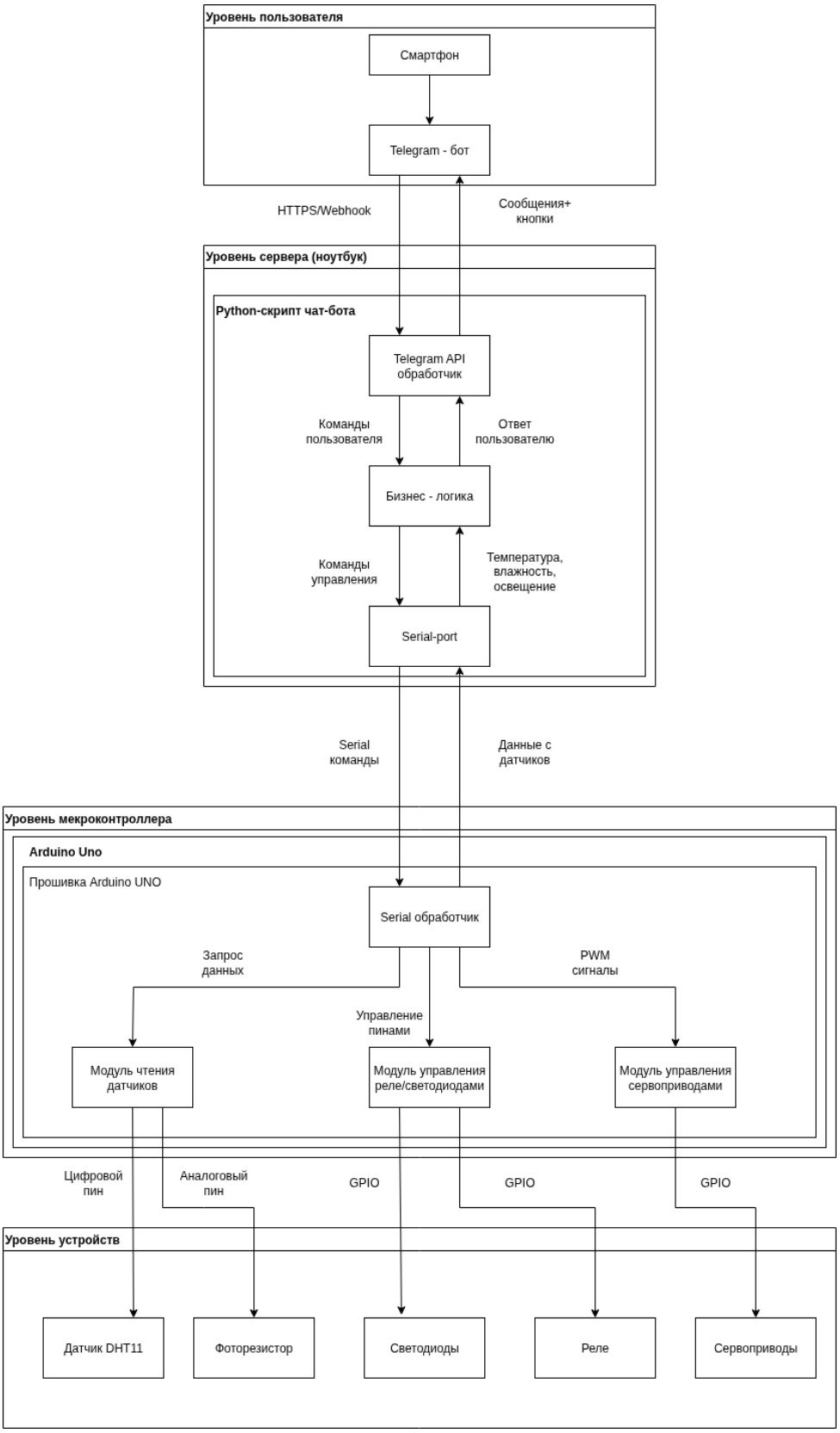


Рис. 2. Архитектура устройства

Принцип работы устройства заключается в следующем:

1. Датчики подключаются к микроконтроллеру Ардуино. Данные с датчиков Ардуино

передает в COM порт ноутбука.

2. На ноутбуке работает программа чат-бот, написанная на Python для Telegram.
3. Пользователь с помощью смартфона используя Telegram бот получает по запросу данные температуры, влажности и освещенности.
4. Если требуется включить удаленно систему обогрева, освещения или проветривания, пользователь отправляет команду в Telegram-бот.
6. Ноутбук обрабатывает команду пользователя, и отправляет команду микроконтроллеру на включение соответствующей системы.

СТРУКТУРНАЯ СХЕМА ПРИБОРА

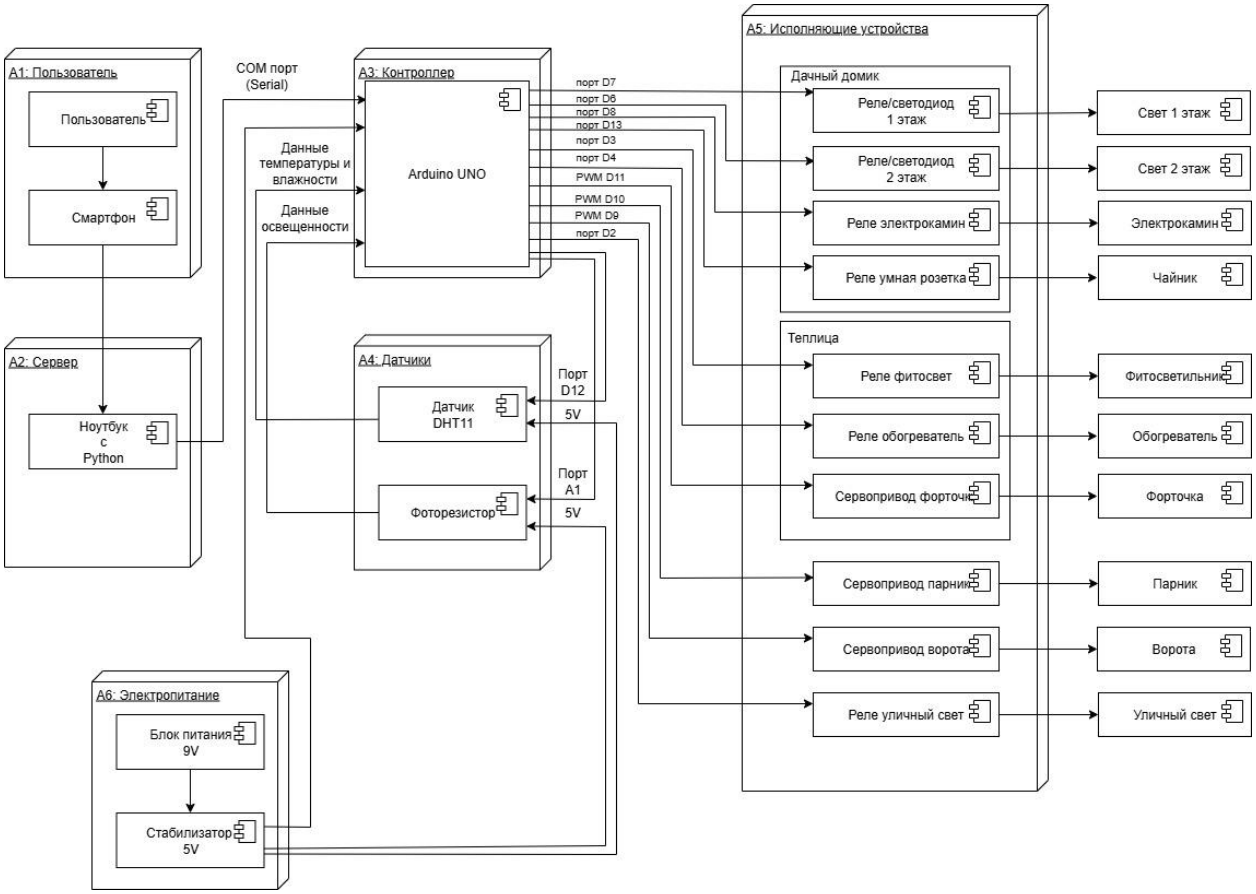


Рис. 3. Структурная схема устройства

В качестве основного компонента – выступает микроконтроллер Ардуино. Для получения значений температуры и влажности будем использовать цифровой датчик DHT11. Для определения освещенности в теплице будем использовать аналоговый датчик освещённости.

Для включения системы подогрева и подсветки растений будем использовать реле, подключённых к цифровым портам Ардуино. Для открытия форточек в теплице, парника и ворот можно использовать сервопривода.

В качестве системы освещения можно использовать, например светодиодные

светильники «Эра» (Приложение № 2).

Для обогрева теплицы – тепличный обогреватель Мегадор МТ150 (Приложение № 3).

В качестве электрокамина для дачного домика можно использовать - электрокамин Скандинавия Sale DK-WT с очагом Fobos (Приложение 4).

МОДЕЛЬ ПРИБОРА

Для проектирования нашего прибора построим модель. Общая схема прибора для модели представлена на рисунке 4. Принципиальная схема на рисунке 5.

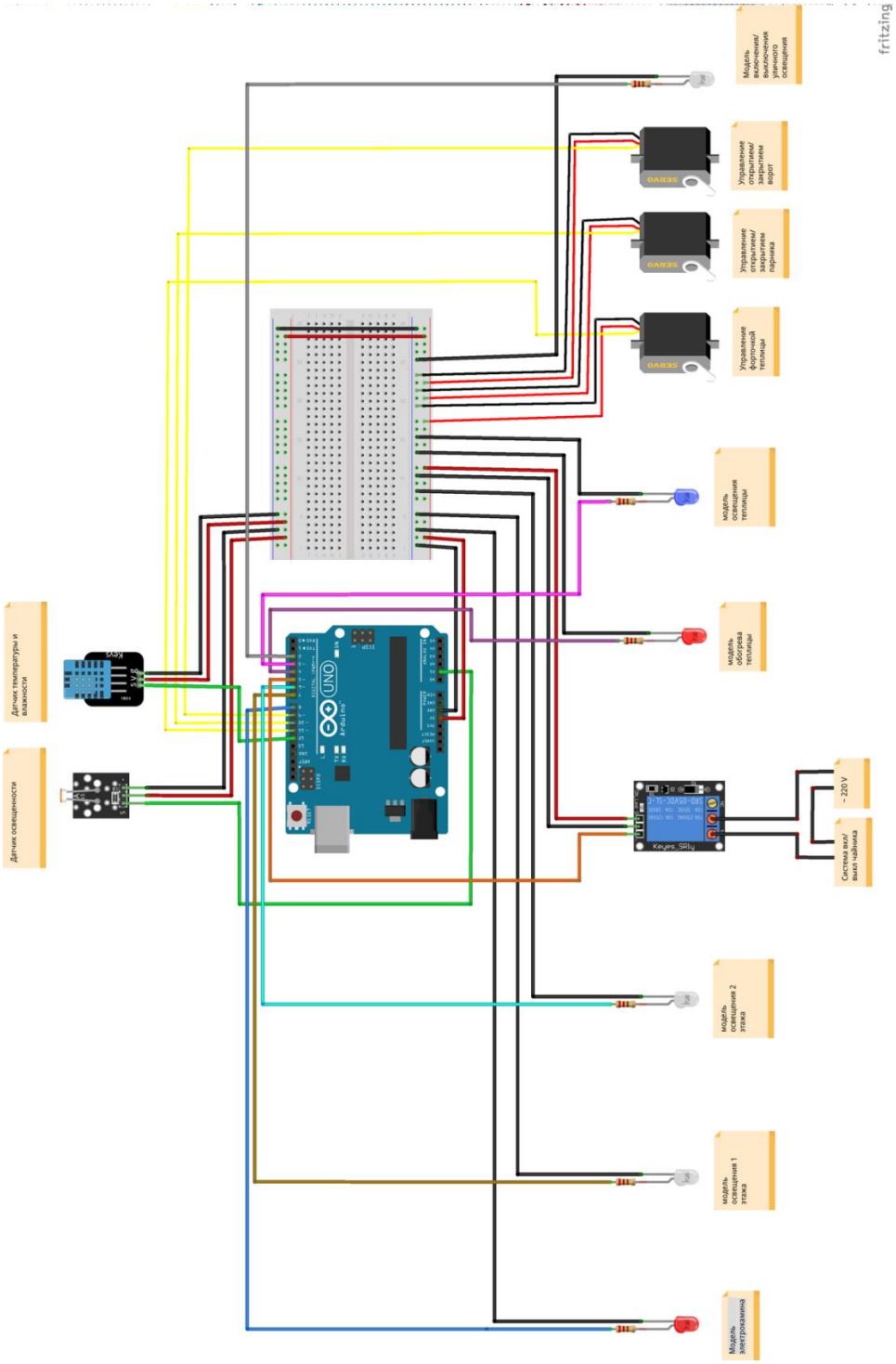


Рис. 4. Схема модели прибора

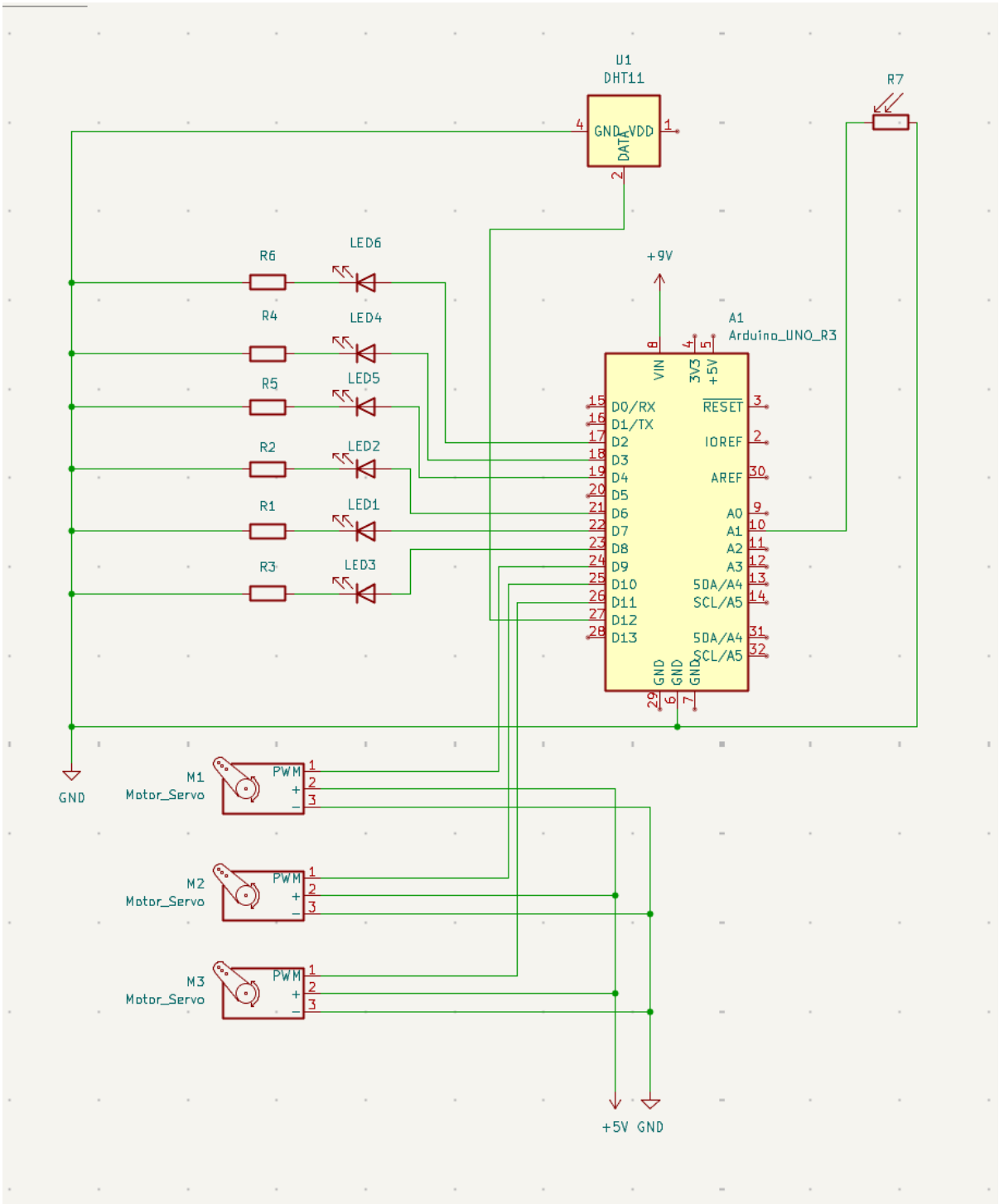


Рис. 5. Принципиальная схема модели прибора

Датчик температуры и влажности подключен к 12 цифровому порту Ардуино. В качестве датчика освещенности будем использовать фоторезистор, подключенный к аналоговому порту А1.

Красный светодиод, подключенный к 8 цифровому порту, будет имитировать электрокамин в дачном домике. Белые светодиоды -освещение 1 и 2 этажа домика, подключенных соответственно к 7 и 6 цифровым портам Ардуино. Управляющий вывод реле для умной розетки (в нее будет включен чайник) подключаем к 5 цифровому порту микроконтроллера.

Система освещения теплицы будет имитировать синий светодиод, подключенный к порту 3 на плате микроконтроллера. Красный светодиод будет моделировать включение системы обогрева теплицы подключенный к порту 4.

Сервоприводы подключаем к портам 11, 10 и 9 они будут имитировать открытие форточки в теплице, открытие парника и открытие (закрытие) ворот соответственно.

Листинг программы для микроконтроллера приведен в приложение 5.

СХЕМА ЧАТ-БОТА И ПРОГРАММА

Для удаленного получения данных с датчиков и управления системами освещения, проветривания и подогрева напишем чат-бот для мессенджера Telegram. Алгоритм работы чат-бота на рисунке 6. Схема чат-бота представлена на рисунке 7.

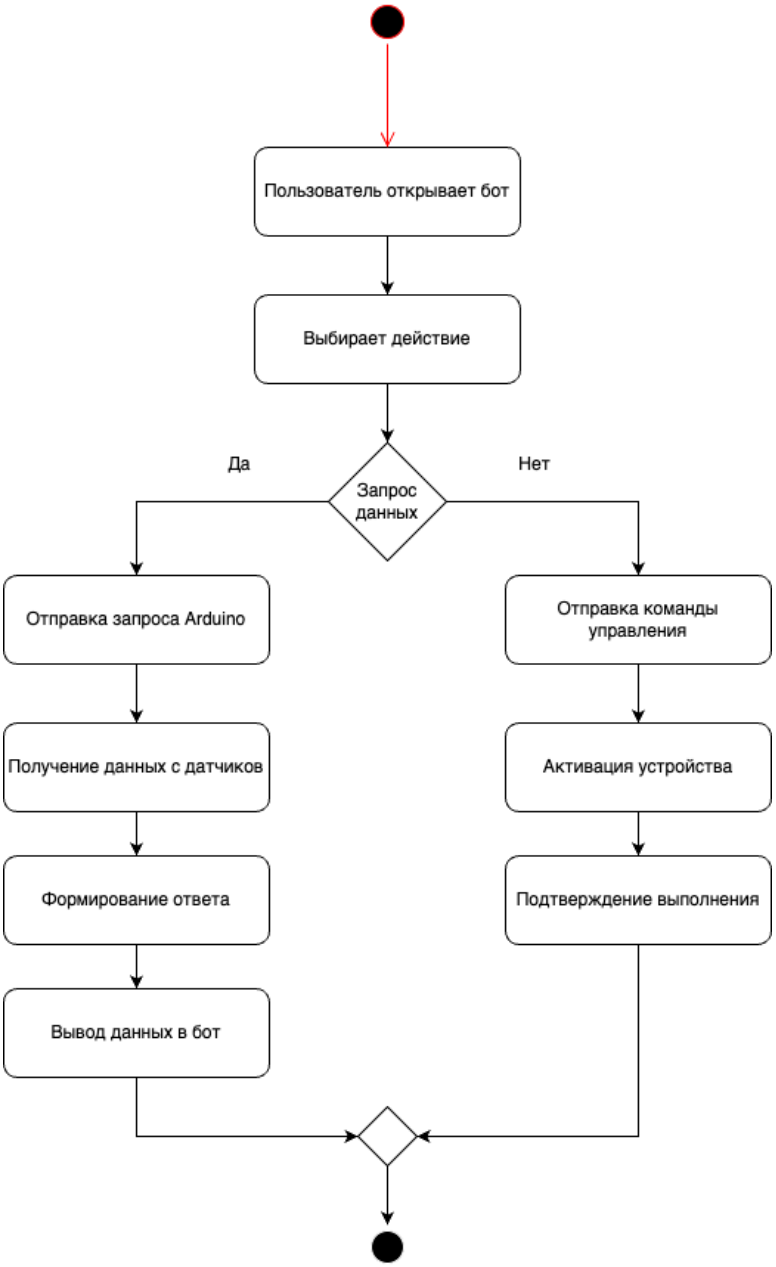


Рис. 6. Алгоритм работы чат бота

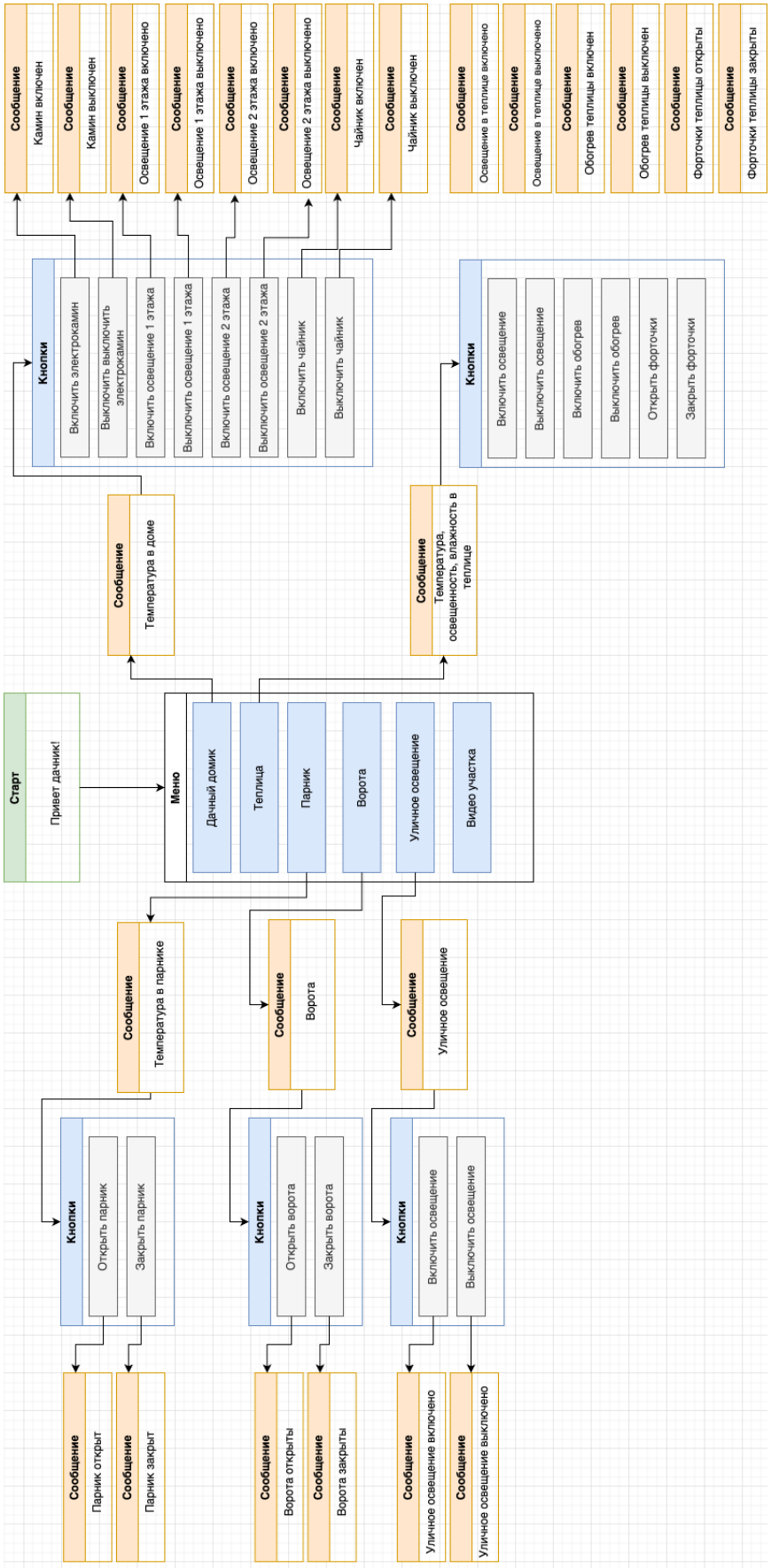


Рис. 7. Схема-чат бота

С помощью BotFatser зарегистрируем бот с именем GreenHouseMag_bot и получим токен.

Для написания чат бота будем использовать язык программирования Python.

Для работы с Ардуино портом нам потребуется библиотека serial, а для работы с telegram библиотека telebot.

```
import serial
import telebot
```

Создадим переменные:

```
#------------Сообщения для домика--------------#
r1='Камин включен'
r2='Камин выключен'
r3='Освещение 1 этажа включено'
r4='Освещение 1 этажа выключено'
r5='Освещение 2 этажа включено'
r6='Освещение 2 этажа выключено '
r7='Чайник включен'
r8='Чайник выключен '

#------------Сообщения для теплицы--------------#
g1='Освещение в теплице включено'
g2='Освещение в теплице выключено '
g3='Обогрев теплицы включен'
g4='Обогрев теплицы выключен'
g5='Форточки теплицы открыты'
g6='Форточки теплицы закрыты'

#------------Сообщения для парника--------------#
p1='Парник окрыт'
p2='Парник закрыт '

#------------Сообщения для ворот--------------#
v1='Ворота открыты'
v2='Ворота закрыты'

#------------Сообщения для ворот--------------#
u1='Уличное освещение включено'
u2='Уличное освещение выключено'

PORT='COM4'
BAUDRATE =9600
arduino=serial.Serial(PORT,BAUDRATE)
```

Для чтения данных с порта будем использовать строковую переменную data, куда будем записывать полученные значения температуры, влажности, освещенности:

```
# Получаем данные с порта
    data = arduino.readline().decode().rstrip()
    values = data.split(',')  # Разделяем данные предполагая, что значения разделены запятой
```

```
    # Сохранение значений в отдельные переменные
    value1 = values[0] # Значение температуры
    value2 = values[1] # Значение влажности
    value3 = values[2] # Значение освещенности
```

Создадим приветствие и кнопки главного меню (рисунок 6)

```
@bot.message_handler(commands=['start']) #стартовая команда
def start(message):
#Создаем кнопки меню
    markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
    btn1 = types.KeyboardButton('Дачный домик')
    btn2 = types.KeyboardButton('Теплица')
    btn3 = types.KeyboardButton('Парник')
    btn4 = types.KeyboardButton('Ворота')
    btn5 = types.KeyboardButton('Уличное освещение')
    btn6 = types.KeyboardButton('Видео участка')
    markup.add(btn1, btn2, btn3,btn4,btn5,btn6)
    bot.send_message(message.from_user.id, "Привет, дачник!", reply_markup=markup)
```

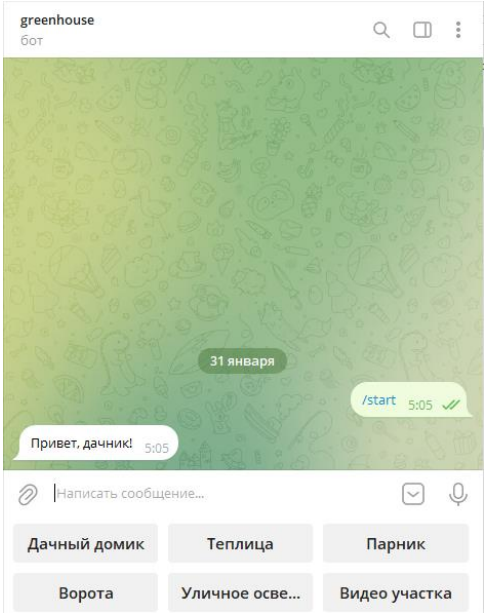


Рис. 8. Стартовое окно чат-бота

Напишем обработчик сообщений от пользователя:

```
 def get_text_messages(message):
#Получаем данные с порта
  data = arduino.readline().decode().rstrip()
  values = data.split(',')  # Разделяем данные предполагая, что значения разделены запятой
  # Сохранение значений в отдельные переменные
  value1 = values[0] # Значение температуры
  value2 = values[1] # Значение влажности
  value3 = values[2] # Значение освещенности
  if message.text == 'Дачный домик':
    markup = types.InlineKeyboardMarkup()
    button1 = types.InlineKeyboardButton("Включить электрокамин", callback_data='R1')
    button2 = types.InlineKeyboardButton("Выключить электрокамин",callback_data='R2')
    button3 = types.InlineKeyboardButton("Включить освещение 1 этаж",callback_data='R3')
    button4 = types.InlineKeyboardButton("Выключить освещение 1 этажа",callback_data='R4')
    button5 = types.InlineKeyboardButton("Включить освещение 2 этаж",callback_data='R5')
    button6 = types.InlineKeyboardButton("Выключить освещение 2 этажа",callback_data='R6')
    button7 = types.InlineKeyboardButton("Включить чайник",callback_data='R7')
    button8 = types.InlineKeyboardButton("Выключить чайник",callback_data='R8')
    markup.add(button1)
    markup.add(button2)
    markup.add(button3)
    markup.add(button4)
    markup.add(button5)
    markup.add(button6)
    markup.add(button7)
    markup.add(button8)

    bot.send_message(message.chat.id, f'Температура в доме:
{value1}'.format(message.from_user), reply_markup=markup)

  elif message.text == 'Теплица':
    markup = types.InlineKeyboardMarkup()
    button1 = types.InlineKeyboardButton("Включить освещение", callback_data='G1')
    button2 = types.InlineKeyboardButton("Выключить освещение",callback_data='G2')
```

```
        button3 = types.InlineKeyboardButton("Включить обогрев", callback_data='G3')
        button4 = types.InlineKeyboardButton("Выключить обогрев",callback_data='G4')
        button5 = types.InlineKeyboardButton("Открыть форточки", callback_data='G5')
        button6 = types.InlineKeyboardButton("Закрыть форточки",callback_data='G6')
        markup.add(button1)
        markup.add(button2)
        markup.add(button3)
        markup.add(button4)
        markup.add(button5)
        markup.add(button6)
        bot.send_message(message.chat.id, f'Освещенность в теплице:
{value3}'.format(message.from_user), reply_markup=markup)

    elif message.text == 'Парник':
        markup = types.InlineKeyboardMarkup()
        button1 = types.InlineKeyboardButton("Открыть парник", callback_data='P1')
        button2 = types.InlineKeyboardButton("Закрыть парник",callback_data='P2')
        markup.add(button1)
        markup.add(button2)

        bot.send_message(message.chat.id, f'Парник: {value3}'.format(message.from_user),
reply_markup=markup)

    elif message.text == 'Ворота':
        markup = types.InlineKeyboardMarkup()
        button1 = types.InlineKeyboardButton("Открыть ворота", callback_data='V1')
        button2 = types.InlineKeyboardButton("Закрыть ворота",callback_data='V2')
        markup.add(button1)
        markup.add(button2)

        bot.send_message(message.chat.id, f'Ворота: {value3}'.format(message.from_user),
reply_markup=markup)

    elif message.text == 'Уличное освещение':
        markup = types.InlineKeyboardMarkup()
```

```
    button1 = types.InlineKeyboardButton("Включить освещение", callback_data='U1')
    button2 = types.InlineKeyboardButton("Выключить освещение",callback_data='U2')
    markup.add(button1)
    markup.add(button2)

    bot.send_message(message.chat.id, f'Уличное освещение:
{value3}'.format(message.from_user), reply_markup=markup)
```

Обработчик команд пользователя для управление объектами дачного участка:

```
def answer(call):
#Отправляем данные Ардуино
#Домик
  if call.data == 'R1': #Включаем камин
    arduino.write(b'0')
    bot.send_message(call.from_user.id, r1)
  elif call.data=='R2': #Выключаем камин
    arduino.write(b'1')
    bot.send_message(call.from_user.id,r2)
  elif call.data=='R3': #включаем свет 1 этаж
    arduino.write(b'2')
    bot.send_message(call.from_user.id,r3)
  elif call.data=='R4': #выключаем свет 1 этаж
    arduino.write(b'3')
    bot.send_message(call.from_user.id,r4)
  elif call.data=='R5': #Включаем свет 2 этаж
    arduino.write(b'4')
    bot.send_message(call.from_user.id,r5)
  elif call.data=='R6': #Выключаем свет 2 этаж
    arduino.write(b'5')
    bot.send_message(call.from_user.id,r6)
  elif call.data=='R7': #Выключаем чайник
    arduino.write(b'6')
    bot.send_message(call.from_user.id,r7)
  elif call.data=='R8': #Выключаем чайник
    arduino.write(b'7')
```

```
        bot.send_message(call.from_user.id,r8)

#Теплица
    elif call.data=='G1': #Включаем освещение
        arduino.write(b'8')
        bot.send_message(call.from_user.id,g1)
    elif call.data=='G2': #Выключаем освещение
        arduino.write(b'9')
        bot.send_message(call.from_user.id,g2)
    elif call.data=='G3': #Включаем обогрев
        arduino.write(b'a')
        bot.send_message(call.from_user.id,g3)
    elif call.data=='G4': #Выключаем обогрев
        arduino.write(b'b')
        bot.send_message(call.from_user.id,g4)
    elif call.data=='G5': #Открываем форточку
        arduino.write(b'c')
        bot.send_message(call.from_user.id,g5)
    elif call.data=='G6': #Закрываем форточку
        arduino.write(b'd')
        bot.send_message(call.from_user.id,g6)
#Парник
    elif call.data=='P1': #Открываем парник
        arduino.write(b'e')
        bot.send_message(call.from_user.id,p1)
    elif call.data=='P2': #Закрываем парник
        arduino.write(b'f')
        bot.send_message(call.from_user.id,p2)
#Ворота
    elif call.data=='V1': #Открываем ворота
        arduino.write(b'r')
        bot.send_message(call.from_user.id, v1)
    elif call.data=='V2': #Закрываем ворота
        arduino.write(b't')
        bot.send_message(call.from_user.id,v2)
```

```
#Уличное освещение
    elif call.data=='U1': #Включаем
        arduino.write(b'x')
        bot.send_message(call.from_user.id, u1)
    elif call.data=='U2': #Выключаем
        arduino.write(b'z')
```

Полный листинг программы приведен в приложении 6.

ЭКОНОМИЧЕСКИЕ ЗАТРАТЫ НА РАЗРАБОТКУ

Комплектующие для прибора можно закупить на сайте dvrobot.ru. В таблице приведены цены на комплектующие на ноябрь 2024 года:

Таблица 3. Стоимость комплектующих

№ п/п	Наименование	Количество	Цена за единицу	Итого
1	Arduno Uno R3	1	980 р.	980 р.
2	Датчик температуры и влажности DHT11 - RB	1	124 р.	124 р.
3	Аналоговый датчик освещенности	1	247 р.	247 р.
4	Реле B1	2	198 р.	396 р.
5	Соединительные провода (комплект)	1	100 р.	100 р.
6	Макетная плата	1	163 р.	163 р.
7	Сервоприводы	3	255 р	765 р.
	ИТОГО			**2775 р.**

Таким образом для закупки комплектующих потребуется 2775 рублей.

Минимальная стоимость работы программиста по данным сайта frelans.ru составляет 1250 рублей/час. На разработку прибора было потрачено 12 часов.

Тогда общая стоимость разработки с учетом работы программиста составила 1250*12+2775 = 17775 рублей.

ОПИСАНИЕ РЕЗУЛЬТАТА ПРОЕКТА

В результате работы над проектом была создана модель прибора для удаленного получения значений температуры, влажности, освещенности и управления системами обогрева, подсветки и проветривания в теплице. (Рисунок 9).

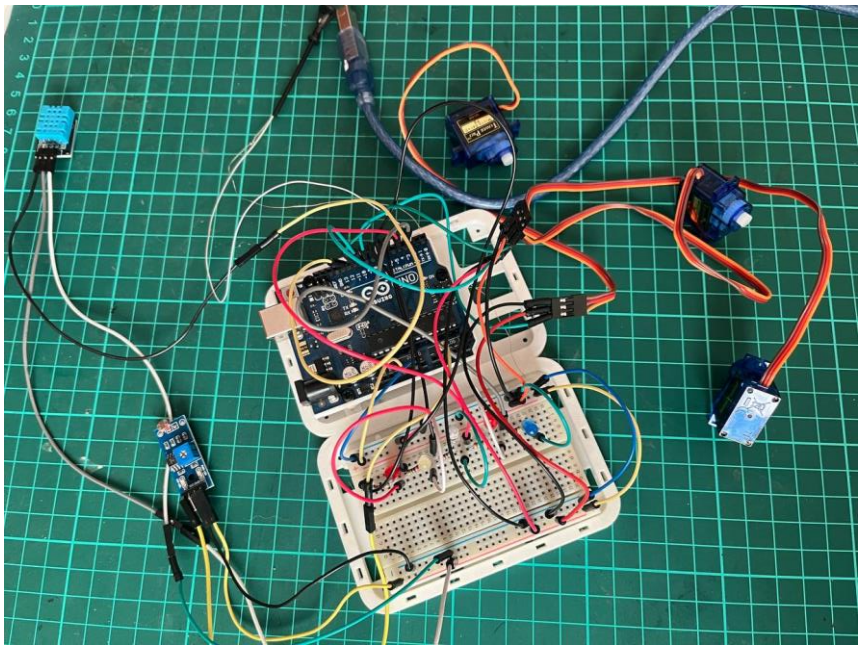


Рис. 9. Модель прибора на макетной плате

Для демонстрации модели, с помощью программы OpenScad были спроектированы объекты дачного участка: дачный домик, теплица, парник, забор и другие (Приложение № 7).

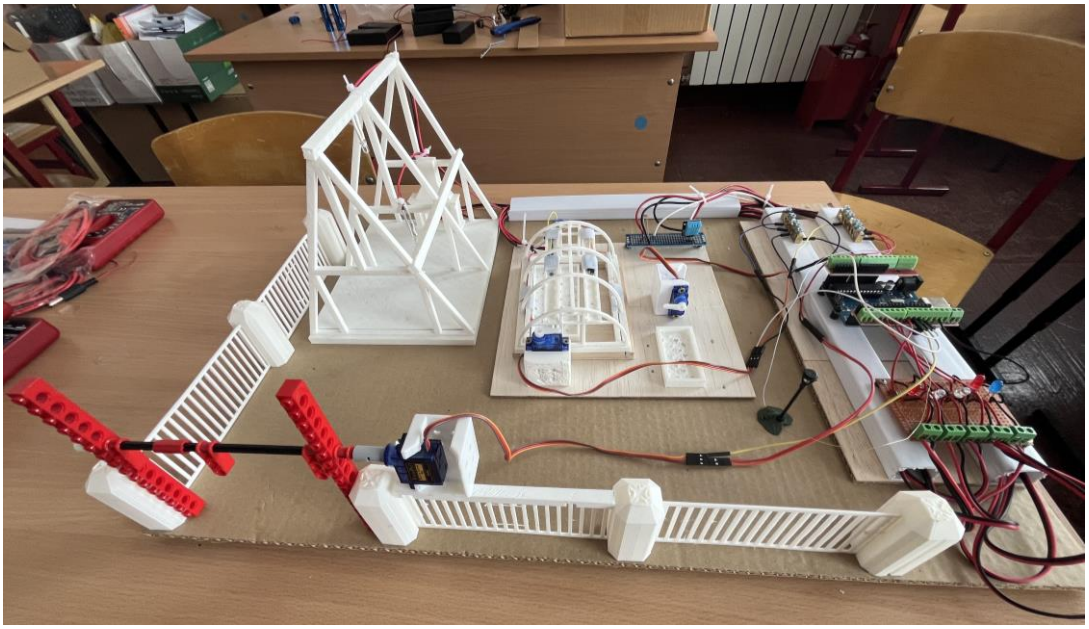


Рис. 10. Макет дачного участка

Для управления прибором был написан чат-бот (Рисунок 11, 12, 13, 14, 15)

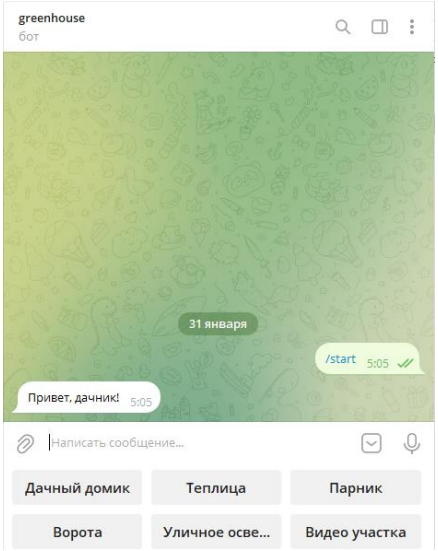


Рис. 11. Стартовый экран

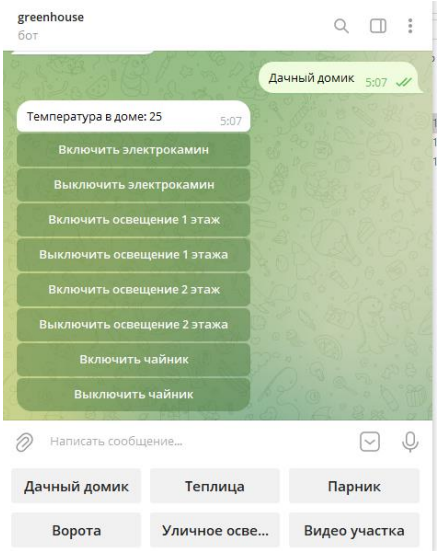


Рис. 12. Меню Дачный домик

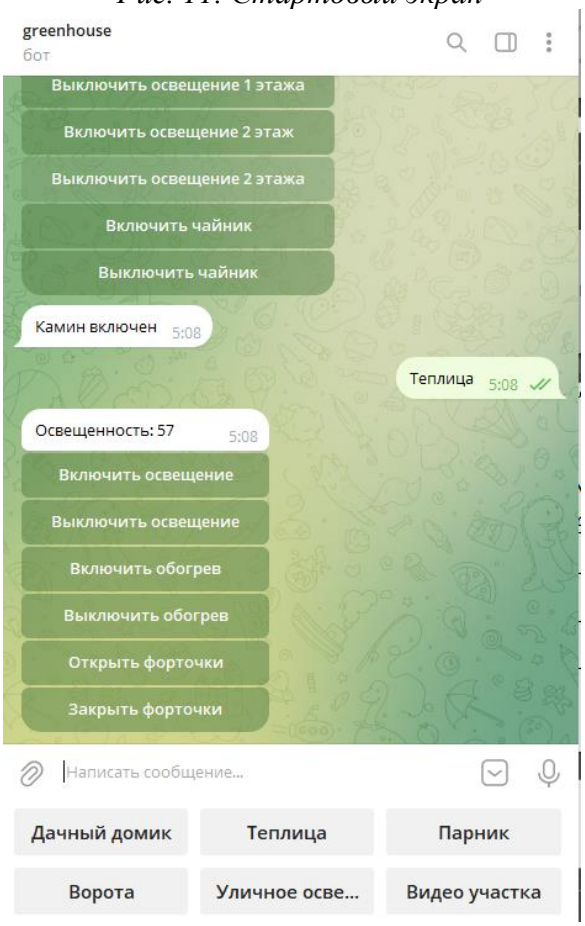


Рис. 13. Меню Теплица

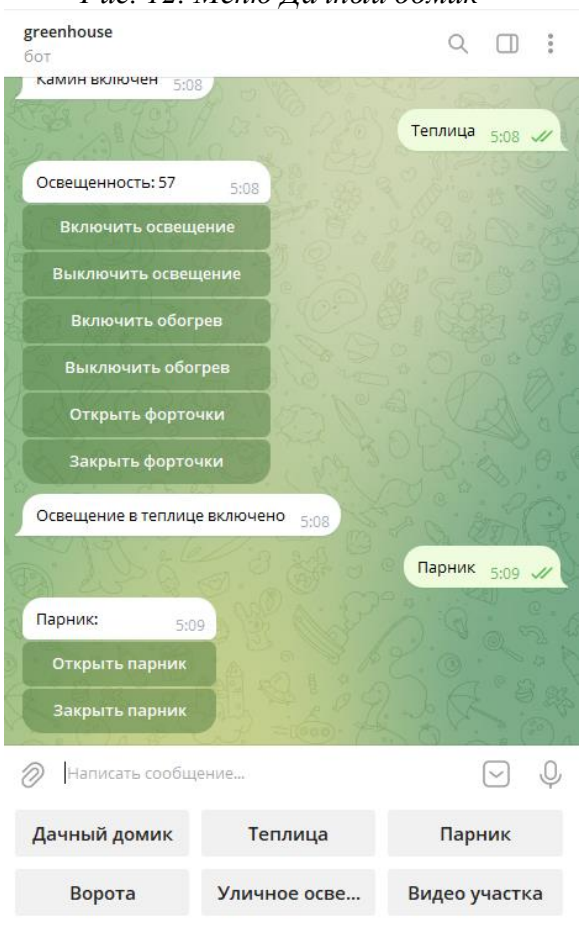


Рис. 14. Меню Парник

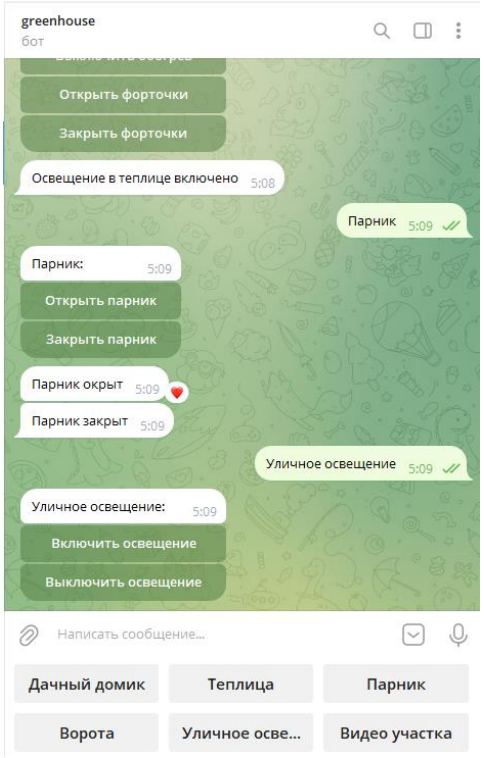


Рис. 15. Меню Уличное освещение

Демонстрация работы устройства и управление с помощью телеграм-бота представлено в видеоролик «Демонстрация прибора».

ЗАКЛЮЧЕНИЕ

В рамках данного проекта был разработан прототип прибора, который позволяет удаленно управлять и контролировать условия в теплице на дачном участке с помощью Telegram-бота. Прототип основан на платформе Arduino и включает в себя датчики для измерения температуры, влажности и освещения в теплице.

Проект был выполнен успешно, и его результатом стал работающая модель прототипа прибора. Прибор позволяет пользователям удаленно мониторить текущие значения параметров на дачном участке, такие как температура и влажность, а также управлять различными устройствами на дачном участке: светом, отоплением в дачном домике и теплице, открывать и закрывать парник, ворота, включать и выключать уличное освещение.

Основные функциональные возможности прототипа включают:

- Получение данных с датчиков температуры, влажности и освещения в реальном времени.
- Управление различными устройствами на дачном участке с помощью команд, отправляемых через Telegram-бота.
- Отображение текущих данных и состояния различных устройств в Telegram-чате.

Одной из главных преимуществ данного устройства является его удаленное управление через Telegram-бота. Это позволяет пользователям контролировать дачный участок даже находясь вдали него, что обеспечивает удобство и гибкость управления.

Однако, следует отметить, что данный прототип является всего лишь начальным шагом в разработке системы удаленного управления дачей. В дальнейшем проект будет доработан и расширен, например, добавлением дополнительных датчиков, функций автоматического регулирования параметров. Также предполагается использовать в качестве сервера и микроконтроллера - мини компьютер Raspberry Pi, а также будет разработана система безопасности дачного участка (противопожарная сигнализация, охранная сигнализация, система видеонаблюдения).

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Основная литература:

1. Бумелева К.Г. Бот в помощь. Книга-практикум по созданию чат-ботов и автоматизации бизнеса // https://www.litres.ru/book/kseniya-grigorevna-b/bot-v-pomosch-kniga-praktikum-po-sozdaniu-chat-botov-68765676/
2. Белов, А.В. Программирование микроконтроллеров для начинающих и не только / А.В. Белов. - СПб.: Наука и техника, 2016. - 352 с.
3. Соммер, У. Программирование микроконтроллерных плат Arduino/Freeduino / У. Соммер. - СПб.: BHV, 2016. - 256 с.
4. Arduino, датчики и сети для связи устройств: Пер. с англ. — 2-е изд. — СПб.: БХВ-Петербург, 2015. —544 с.: ил.
5. Виктор Петин – Электроника. Проекты с использованием контроллера Arduino. Санкт-Петербург «БХВ-Петербург» 2014
7. Python для детей. Самоучитель по программированию / Джейсон Бриггс ; пер. с англ. Станислава Ломакина ; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2017. — 320 с
8. Простой Python. Современный стиль программирования. — СПб.: Питер, 2016. — 480 с.: ил. — (Серия «Бестселлеры O'Reilly»)

Интернет-ресурсы:

1. GSM термометр ALONIO T2 // https://alonio.ru/catalog/alonio-t2/
2. Система GSM контроля температуры EctoControl "Температура"// https://ectostroy.ru/ready-made-solutions/temperature-1
3. GSM термодатчик ELEUS RC-312 для контроля температуры через сотовый телефон // https://agronoma.ru/vidyaevo/aksessuary-dlya-klimaticheskogo-oborudovaniya/gsm-termodatchik-eleus-rc-312-dlya-kontrolya-temperatury-cherez-sotovyy-telefon_754733
4. Ардуино UNO R3 // https://dvrobot.ru/catalog/mikrokontrollery/ arduino/_platy/14315/
5. Датчик температуры и влажности DHT11 – RB // https://dvrobot.ru/catalog/datchiki_i_moduli/temperatura_i_vlazhnost/_vlazhnost_temperatura/15350/
6. Аналоговый датчик освещенности // https://dvrobot.ru/catalog /datchiki_i_moduli/osveshchennost/15239/
7. Реле B1 // https://dvrobot.ru/catalog/ datchiki_i_moduli/rele/_elektromagnitnye/14811/
9. Фитосветильник светодиодный, 30 Вт, 530 мм, IP65, полный спектр, бело-красный, FITO-3x10W-LINE-Ra90, «ЭРА» // https://satom.ru/p/1332463231-fitosvetilnik-

svetodiodnyy-30-vt-530-mm-ip65-polnyy-spektr-belo-krasnyy-fito-3h10w-line-ra90-era/

10. Тепличный обогреватель МЕГАДОР МТ150 длина 150 см, мощность 400 Вт // https://irkshop.ru/dlya-doma-i-dachi/klimaticheskaya-tehnika/plintusnye-obogrevateli/teplichnyy-obogrevatel-megador-mt150-dlina-150-sm-moschnost-400-vt/

ПРИЛОЖЕНИЕ 1. СПИСОК КОМПЛЕКТУЮЩИХ ДЛЯ УСТРОЙСТВА[5]

Рис. 16. Ардуино UNO R3

Технические характеристики Arduino UNO R3:

1. Микроконтроллер: ATmega328p.
2. Рабочее напряжение: 5 В.
3. Входное напряжение (рекомендуемое): 7-12 В.
4. Входное напряжение (пределы): 6-20 В.
5. Цифровые порты ввода/вывода: 14 портов (из них 6 с ШИМ).
6. Аналоговые порты ввода: 6 портов.
7. Ток для портов: 40 мА.
8. Ток для 3.3В источника: 50 мА.
9. ППЗУ: 32 КВ (из них 2 Кб используются загрузчиком).
10. ОЗУ (SRAM): 2 Кб.
11. ПЗУ (EEPROM): 1 Кб.
12. Тактовая частота: 16 МГц.
13. Поддержка USB интерфейса и питания (без внешнего источника питания).
14. Поддержка ISP загрузки.
15. Поддержка последовательного (Serial UART TTL) соединение через контакты RX TX.
16. Поддержка контакта AREF.
17. Автоматический выбор источника питания (USB или внешний адаптер).
18. Защита USB-порта компьютера от перенапряжения и коротких замыканий (более 500мА – разрыв и дальнейшее восстановление при нормализации).

[5] https://dvrobot.ru/

Рис. 17. Датчик температуры и влажности DHT11

Датчик на основе сенсора DHT11, способен определять температуру и влажность окружающей среды. Он прост в использовании, имеет высокую точность и малое энергопотребление.

Технические характеристики:

1. Рабочее напряжение - 5 Вольт DC;
2. Диапазон измеряемых температур - 0-50°C с погрешностью ±2°C;
3. Диапазон измеряемой влажности - 20-90% RH с погрешностью ±5% RH;
4. Размер: 28 x 12 мм.

Рис. 18. Аналоговый датчик освещенности

Аналоговый датчик освещенности со встроенным фоторезистором, изменяющим свое сопротивление от яркости света, падающего на него. Таким образом, датчик может реагировать на изменение света, и давать сигнал, к примеру, на включение или выключение устройства:

фонаря, реле, насос и т.п.

Характеристики:

1. рабочее напряжение от 3 до 5 Вольт DC ;
2. выходной сигнал : аналоговый или цифровой;
3. наличие потенциометра настройки чувствительности;
4. размер: 37 х 15 мм

Рис. 19. Реле В1

Характеристики:

1. управляющее напряжение 5 Вольт DC;
2. нагрузка AC 250V/10A, DC 30V/10A;
3. быстродействие 10 мс;
4. размер: 45 х 25 мм;
5. вес: до 20 гр.

ПРИЛОЖЕНИЕ 2. ФИТОСВЕТИЛЬНИК СВЕТОДИОДНЫЙ

Фитосветильник светодиодный, 30 Вт, 530 мм, IP65, полный спектр, бело-красный, FITO-3x10W-LINE-Ra90, «ЭРА»[6]

Рис. 20. Лампа «ЭРА»

Лампа «ЭРА» полного спектра предназначена для выращивания зелени, рассады паслёновых, а также для досвечивания комнатных цветов. Прибор излучает комфортный для глаз человека свет, поэтому его можно использовать в жилых помещениях.

Применение лампы

- Досветка комнатных растений в сезоны нехватки солнечного света — осенью и зимой.
- Выращивание рассады в весенний период.
- Досветка цветущих растений для формирования завязей плодов.
- Освещение зелени для её быстрого роста, яркого цвета и насыщенного аромата.
- Досветка растений в теплицах.
- Выращивание зелени и плодовых растений в больших объёмах в рамках фермерского бизнеса.

Характеристики

- Горючесть пластика: не горючий.
- Диапазон рабочего напряжения в сети: 210–265 В.
- Индекс цветопередачи: 30 Ra.
- Количество светодиодов: 90 шт.
- Мощность заявленная (при 220 В): 30 Вт.

[6] https://www.eraworld.ru/catalog/2654645#detailInfoSectionContent

ПРИЛОЖЕНИЕ 3. ТЕПЛИЧНЫЙ ОБОГРЕВАТЕЛЬ

Тепличный обогреватель МЕГАДОР МТ150[7] длина 150 см, мощность 400 Вт

Рис. 21. МЕГАДОР МТ150

Описание МЕГАДОР МТ150

Тепличный обогреватель МЕГАДОР может быть использован как на небольших приусадебных участках, так и на крупных предприятиях. Владельцы круглогодичных промышленных теплиц могут значительно снизить затраты на электроэнергию, установив систему из нескольких обогревателей. А дачники могут защитить свой урожай от перепадов температур и заморозков. Обогреватель не имеет вилки и предназначен для подключения через терморегулятор, что позволяет легко поддерживать оптимальную температуру среды.

Основные характеристики:

Потребляемая мощность: 400 Вт

Размер: 1500x170x20 (длина x высота x ширина) мм

Степень защиты: Влагозащищенный IP66

Длина шнура: 3 м

Площадь отопления: для теплицы 6x2 метра требуется два обогревателя МТ150

[7] http://thermomir.ru/subitem/energosberegayuschiy-obogrevatel-megador-mt-150?ysclid=m6ikekley4824773642

ПРИЛОЖЕНИЕ 4. ЭЛЕКТРОКАМИН СКАНДИНАВИЯ SALE DK-WT С ОЧАГОМ FOBOS

Рис. 22. Электрокамин Скандинавия sale DK-WT с очагом Fobos

Электрокамин Скандинавия Sale DK-WT с очагом Fobos[8] изготовлен из МДФ и ламинированного ДСП в цвете чёрный с белым. Полностью разборный. Разработан в Скандинавском стиле – один из самых популярных современных стилей, распространенных по всему миру.

Портал укомплектован классическим электроочагом Fobos. В очаге визуальный эффект горящего огня создается с помощью проекционной LED-технологии. Присутствует звуковой эффект горения пламени и потрескивания дров. Очаг может работать как в режиме обогрева, так и в декоративном режиме без обогрева. В режиме обогрева в очаге предусмотрено 2 уровня регулировки обогрева, возможно переключение мощности обогрева 750-1500 Вт, поэтому в холодные дни данный электрокамин можно использовать как дополнительный источник тепла. Предусмотрена функция защиты от перегрева.

[8] https://liderkamin.ru/products/elektrokamin-skandinaviya-sale-bl-s-ochagom-fobos

ПРИЛОЖЕНИЕ 5. ЛИСТИНГ ПРОГРАММЫ ДЛЯ МИКРОКОНТРОЛЛЕРА АРДУИНО

```
#include <Servo.h>
#include <Arduino.h>
#include "DHT_Async.h"

#define DHT_SENSOR_TYPE DHT_TYPE_11

static const int DHT_SENSOR_PIN = 12;
DHT_Async dht_sensor(DHT_SENSOR_PIN, DHT_SENSOR_TYPE);
int val;
Servo servo1, servo2, servo3;

void setup() {
  pinMode(A1,INPUT);
  pinMode(8,OUTPUT); //камин
  pinMode(7,OUTPUT); //освещение 1 этажа
  pinMode(6,OUTPUT); //освещение 2  этажа
  pinMode(13,OUTPUT); //реле для умной розетки
  pinMode(3,OUTPUT); //освещение теплицы
  pinMode(4,OUTPUT); //обогрев теплицы
  pinMode(2,OUTPUT); //улмчное освнщение

  digitalWrite(5, LOW);
  analogWrite(A1,LOW);
  Serial.begin(9600);
  servo1.attach(11); // форточка теплицы
  servo2.attach(10); // парник
  servo3.attach(9); // парник
}


static bool measure_environment(float *temperature, float *humidity) {
  static unsigned long measurement_timestamp = millis();


  if (millis() - measurement_timestamp > 2000ul) {
    if (dht_sensor.measure(temperature, humidity)) {
      measurement_timestamp = millis();
      return (true);
    }
  }

  return (false);
}


void loop() {
  float temperature;
  float humidity;
  int light=analogRead(A1);

 if  (measure_environment(&temperature, &humidity)) {
```

```
    //Serial.print("T = ");
    //Serial.print(temperature, 1);
    //Serial.print(" deg. C, H = ");
    //Serial.print(humidity, 1);
    //Serial.print("%, ");
    //Serial.print("Light = ");
    //Serial.print(light*10);
    //Serial.println();

  }



  if (Serial.available()) {

   val = Serial.read(); // переменная val равна полученной команде

   if (val == '0') {digitalWrite(8, HIGH);} // включаем камин
   if (val == '1') {digitalWrite(8, LOW);} //  0 выключаем камин
   if (val == '2') {digitalWrite(7, HIGH);} // включаем освещение 1 этажа
   if (val == '3') {digitalWrite(7, LOW);} //  выключаем освещение 1 этажа
   if (val == '4') {digitalWrite(6, HIGH);} // включаем освещение 2 этажа
   if (val == '5') {digitalWrite(6, LOW);} //  выключаем освещение 2 этаа
   if (val == '7') {digitalWrite(13, HIGH);}     //включаем чайник
   if (val == '6') {digitalWrite(13, LOW);}     // выключаем чайник
   if (val == '8') {digitalWrite(3, HIGH);} // включаем освещение теплицы
   if (val == '9') {digitalWrite(3, LOW);} //  выключаем освещение теплицы
   if (val == 'a') {digitalWrite(4, HIGH);} // включаем обогрев теплицы
   if (val == 'b') {digitalWrite(4, LOW);} //  выключаем обогрев теплицы
   if (val == 'c') {servo1.write(0);}     //открываем форточку теплицы
   if (val == 'd') {servo1.write(180);}     // закрываем форточку теплицы
   if (val == 'e') {servo2.write(0);}     //открываем парник
   if (val == 'f') {servo2.write(180);}     // закрываем парник
   if (val == 'r') {servo3.write(0);}     //открываем ворота
   if (val == 't') {servo3.write(90);}     // закрываем ворота
   if (val == 'x') {digitalWrite(2, HIGH);} // при 1 включаем светодиод
   if (val == 'z') {digitalWrite(2, LOW);} // при 0 выключаем светодиод

  }
}
```

ПРИЛОЖЕНИЕ 6. ЛИСТИНГ ПРОГРАММЫ ЧАТ-БОТА

```
import serial
import telebot
from telebot import types
import random
from random import choice

bot = telebot.TeleBot('6607815331:AAHvY9UoBhmWmQ3z0uz8NtmyMosflFHWpFg')

#------------Сообщения для домика--------------#
r1='Камин включен'
r2='Камин выключен'
r3='Освещение 1 этажа включено'
r4='Освещение 1 этажа выключено'
r5='Освещение 2 этажа включено'
r6='Освещение 2 этажа выключено '
r7='Чайник включен'
r8='Чайник выключен '

#------------Сообщения для теплицы--------------#
g1='Освещение в теплице включено'
g2='Освещение в теплице выключено '
g3='Обогрев теплицы включен'
g4='Обогрев теплицы выключен'
g5='Форточки теплицы открыты'
g6='Форточки теплицы закрыты'

#------------Сообщения для парника--------------#
p1='Парник окрыт'
p2='Парник закрыт '

#------------Сообщения для ворот--------------#
v1='Ворота открыты'
v2='Ворота закрыты'

#------------Сообщения для ворот--------------#
u1='Уличное освещение включено'
u2='Уличное освещение выключено'

PORT='COM4'
BAUDRATE =9600
arduino=serial.Serial(PORT,BAUDRATE)


@bot.message_handler(commands=['start']) #стартовая команда
def start(message):
#Создаем кнопки меню
    markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
    btn1 = types.KeyboardButton('Дачный домик')
    btn2 = types.KeyboardButton('Теплица')
    btn3 = types.KeyboardButton('Парник')
    btn4 = types.KeyboardButton('Ворота')
    btn5 = types.KeyboardButton('Уличное освещение')
    btn6 = types.KeyboardButton('Видео участка')


    markup.add(btn1, btn2, btn3,btn4,btn5,btn6)
    bot.send_message(message.from_user.id, "Привет, дачник!", reply_markup=markup)
```

```
@bot.message_handler(content_types=['text'])

def get_text_messages(message):
 #Получаем данные с порта
   data = arduino.readline().decode().rstrip()
   values = data.split(',')  # Разделяем данные предполагая, что значения разделены запятой
   # Сохранение значений в отдельные переменные
   value1 = values[0] # Значение температуры
   value2 = values[1] # Значение влажности
   value3 = values[2] # Значение освещенности


   if message.text == 'Дачный домик':
      markup = types.InlineKeyboardMarkup()
      button1 = types.InlineKeyboardButton("Включить электрокамин", callback_data='R1')
      button2 = types.InlineKeyboardButton("Выключить электрокамин",callback_data='R2')
      button3 = types.InlineKeyboardButton("Включить освещение 1 этаж",callback_data='R3')
      button4 = types.InlineKeyboardButton("Выключить освещение 1 этажа",callback_data='R4')
      button5 = types.InlineKeyboardButton("Включить освещение 2 этаж",callback_data='R5')
      button6 = types.InlineKeyboardButton("Выключить освещение 2 этажа",callback_data='R6')
      button7 = types.InlineKeyboardButton("Включить чайник",callback_data='R7')
      button8 = types.InlineKeyboardButton("Выключить чайник",callback_data='R8')
      markup.add(button1)
      markup.add(button2)
      markup.add(button3)
      markup.add(button4)
      markup.add(button5)
      markup.add(button6)
      markup.add(button7)
      markup.add(button8)

      bot.send_message(message.chat.id, f'Температура на дачном участке:
{value1}'.format(message.from_user), reply_markup=markup)

   elif message.text == 'Теплица':
      markup = types.InlineKeyboardMarkup()
      button1 = types.InlineKeyboardButton("Включить освещение", callback_data='G1')
      button2 = types.InlineKeyboardButton("Выключить освещение",callback_data='G2')
      button3 = types.InlineKeyboardButton("Включить обогрев", callback_data='G3')
      button4 = types.InlineKeyboardButton("Выключить обогрев",callback_data='G4')
      button5 = types.InlineKeyboardButton("Открыть форточки", callback_data='G5')
      button6 = types.InlineKeyboardButton("Закрыть форточки",callback_data='G6')
      markup.add(button1)
      markup.add(button2)
      markup.add(button3)
      markup.add(button4)
      markup.add(button5)
      markup.add(button6)
      bot.send_message(message.chat.id, f'Освещенность на улице: {value3}'.format(message.from_user),
reply_markup=markup)

   elif message.text == 'Парник':
      markup = types.InlineKeyboardMarkup()
      button1 = types.InlineKeyboardButton("Открыть парник", callback_data='P1')
      button2 = types.InlineKeyboardButton("Закрыть парник",callback_data='P2')
      markup.add(button1)
      markup.add(button2)
```

```
        bot.send_message(message.chat.id, f'Освещеность на : {value3}'.format(message.from_user),
reply_markup=markup)

    elif message.text == 'Ворота':
        markup = types.InlineKeyboardMarkup()
        button1 = types.InlineKeyboardButton("Открыть ворота", callback_data='V1')
        button2 = types.InlineKeyboardButton("Закрыть ворота",callback_data='V2')
        markup.add(button1)
        markup.add(button2)

        bot.send_message(message.chat.id, f'Ворота '.format(message.from_user), reply_markup=markup)

    elif message.text == 'Уличное освещение':
        markup = types.InlineKeyboardMarkup()
        button1 = types.InlineKeyboardButton("Включить освещение", callback_data='U1')
        button2 = types.InlineKeyboardButton("Выключить освещение",callback_data='U2')
        markup.add(button1)
        markup.add(button2)

        bot.send_message(message.chat.id, f'Освещеность на улице: {value3}'.format(message.from_user),
reply_markup=markup)


@bot.callback_query_handler(func=lambda call: True)
def answer(call):
#Отправляем данные Ардуино
#Домик
    if call.data == 'R1': #Включаем камин
        arduino.write(b'0')
        bot.send_message(call.from_user.id, r1)
    elif call.data=='R2': #Выключаем камин
        arduino.write(b'1')
        bot.send_message(call.from_user.id,r2)
    elif call.data=='R3': #включаем свет 1 этаж
        arduino.write(b'2')
        bot.send_message(call.from_user.id,r3)
    elif call.data=='R4': #выключаем свет 1 этаж
        arduino.write(b'3')
        bot.send_message(call.from_user.id,r4)
    elif call.data=='R5': #Включаем свет 2 этаж
        arduino.write(b'4')
        bot.send_message(call.from_user.id,r5)
    elif call.data=='R6': #Выключаем свет 2 этаж
        arduino.write(b'5')
        bot.send_message(call.from_user.id,r6)
    elif call.data=='R7': #Выключаем чайник
        arduino.write(b'6')
        bot.send_message(call.from_user.id,r7)
    elif call.data=='R8': #Выключаем чайник
        arduino.write(b'7')
        bot.send_message(call.from_user.id,r8)

#Теплица
    elif call.data=='G1': #Включаем освещение
        arduino.write(b'8')
        bot.send_message(call.from_user.id,g1)
    elif call.data=='G2': #Выключаем освещение
```

```
        arduino.write(b'9')
        bot.send_message(call.from_user.id,g2)
    elif call.data=='G3': #Включаем обогрев
        arduino.write(b'a')
        bot.send_message(call.from_user.id,g3)
    elif call.data=='G4': #Выключаем обогрев
        arduino.write(b'b')
        bot.send_message(call.from_user.id,g4)
    elif call.data=='G5': #Открываем форточку
        arduino.write(b'c')
        bot.send_message(call.from_user.id,g5)
    elif call.data=='G6': #Закрываем форточку
        arduino.write(b'd')
        bot.send_message(call.from_user.id,g6)
#Парник
    elif call.data=='P1': #Открываем парник
        arduino.write(b'e')
        bot.send_message(call.from_user.id,p1)
    elif call.data=='P2': #Закрываем парник
        arduino.write(b'f')
        bot.send_message(call.from_user.id,p2)
#Ворота
    elif call.data=='V1': #Открываем ворота
        arduino.write(b'r')
        bot.send_message(call.from_user.id, v1)
    elif call.data=='V2': #Закрываем ворота
        arduino.write(b't')
        bot.send_message(call.from_user.id,v2)
#Уличное освещение
    elif call.data=='U1': #Включаем
        arduino.write(b'x')
        bot.send_message(call.from_user.id, u1)


    elif call.data=='U2': #Выключаем
        arduino.write(b'z')
        bot.send_message(call.from_user.id,u2)
bot.polling(none_stop=True, interval=0) #обязательная для работы бота часть
```

ПРИЛОЖЕНИЕ 7. ПРОЕКТИРОВАНИЕ ОБЪЕКТОВ ДАЧНОГО УЧАСТКА В ПРОГРАММЕ OpenScad

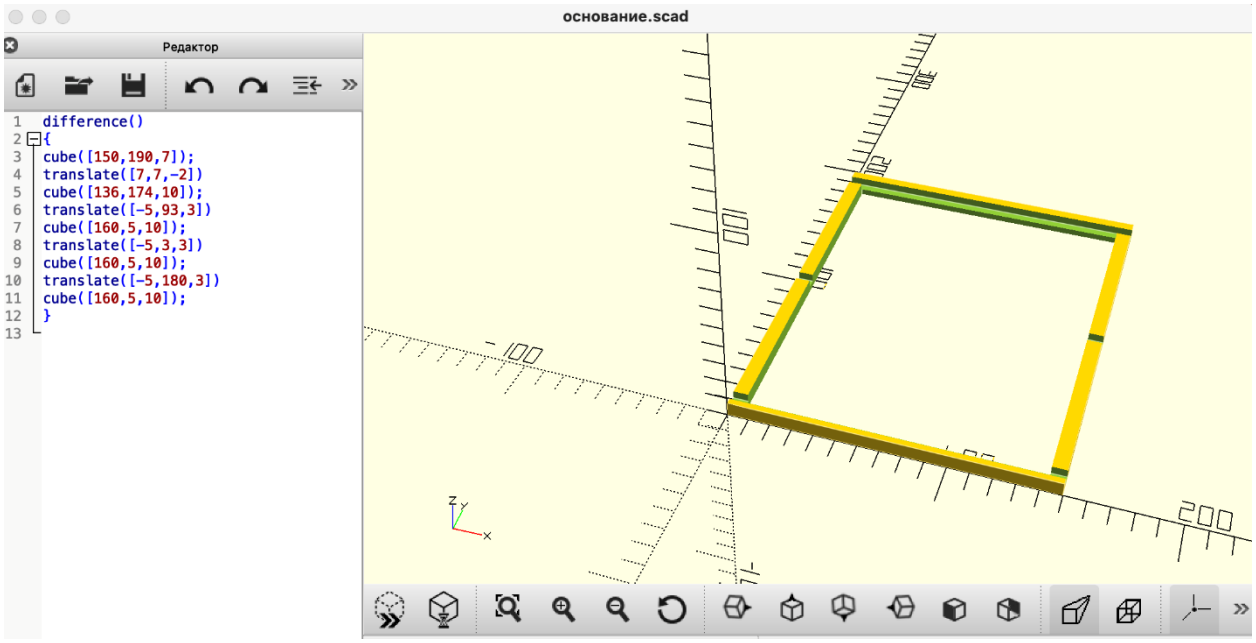


Рис. 23. Основание дачного домика

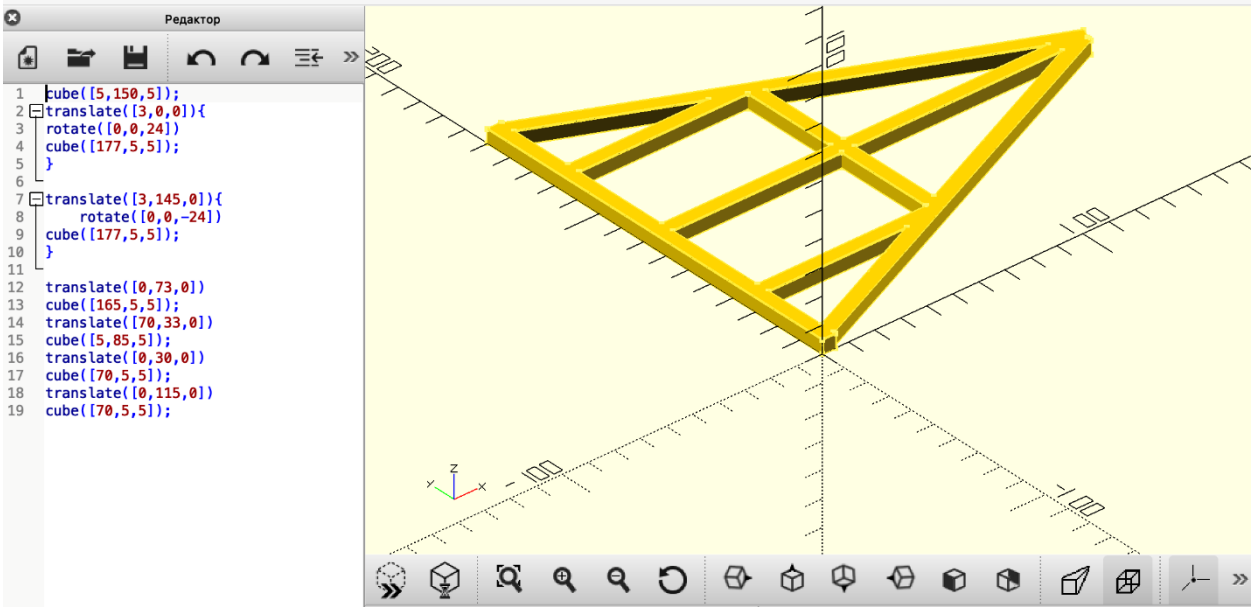


Рис. 24а. Стены домика

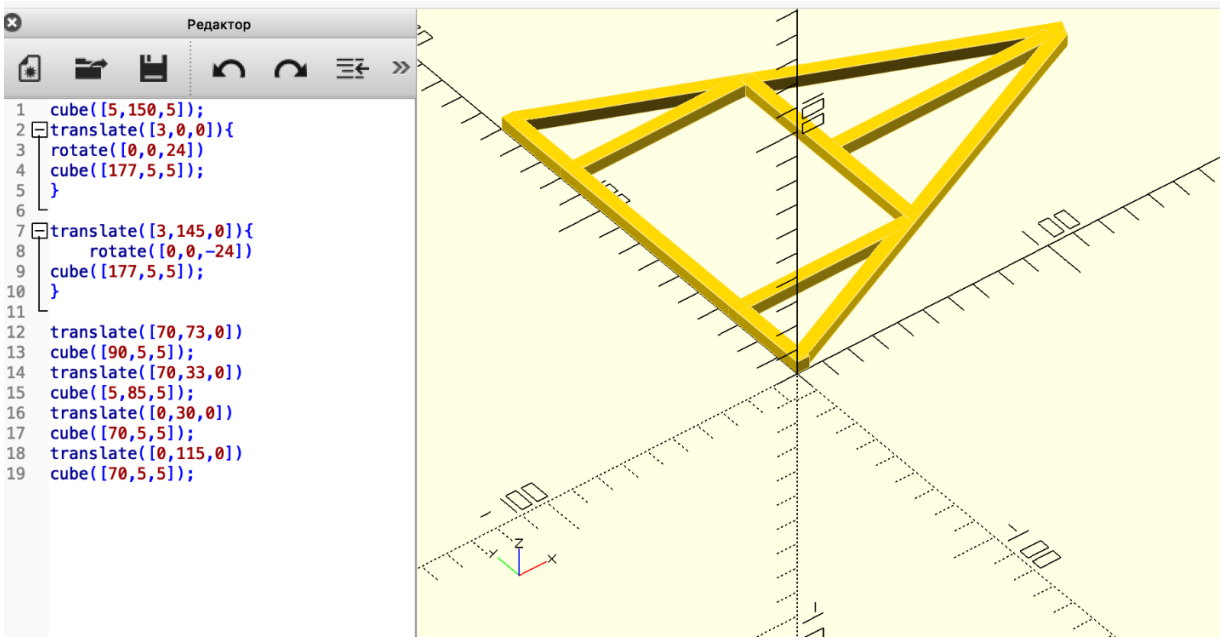


Рис. 24б. Стены домика

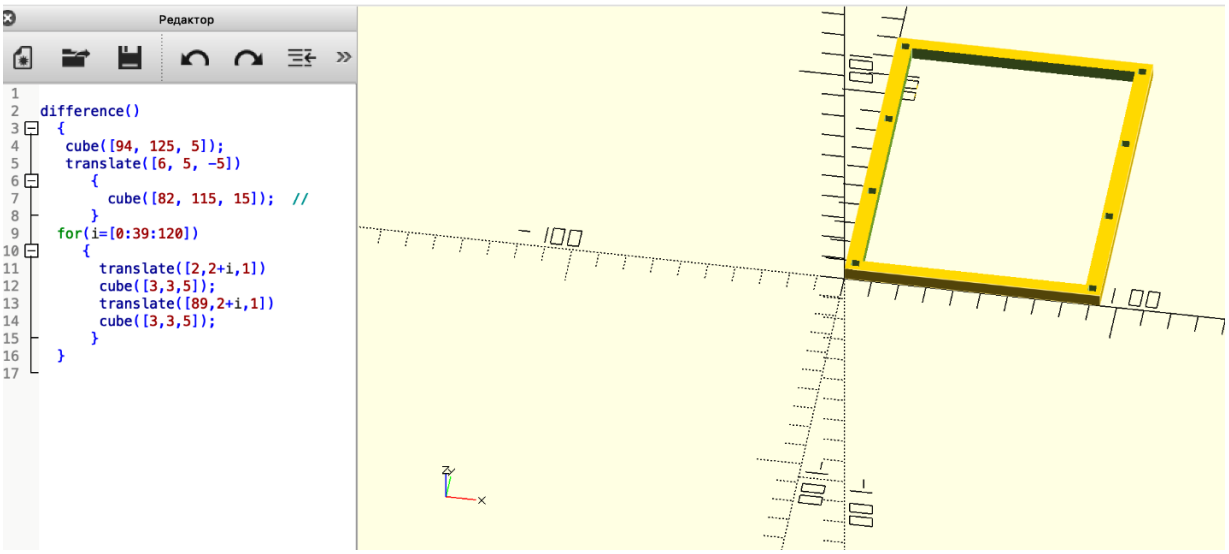


Рис. 25а. Теплица

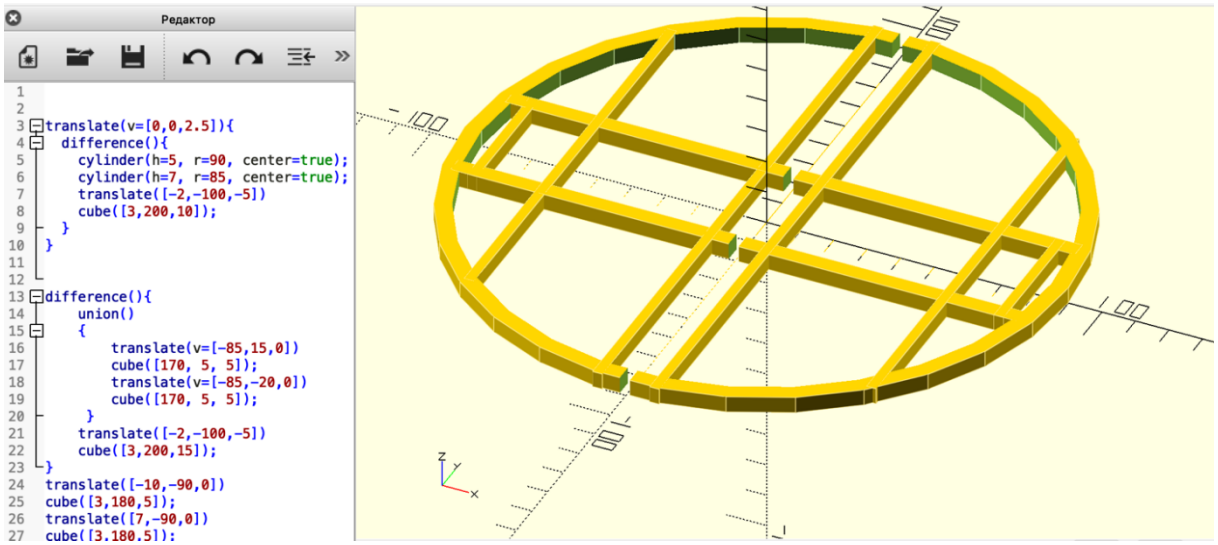


Рис. 25б. Теплица

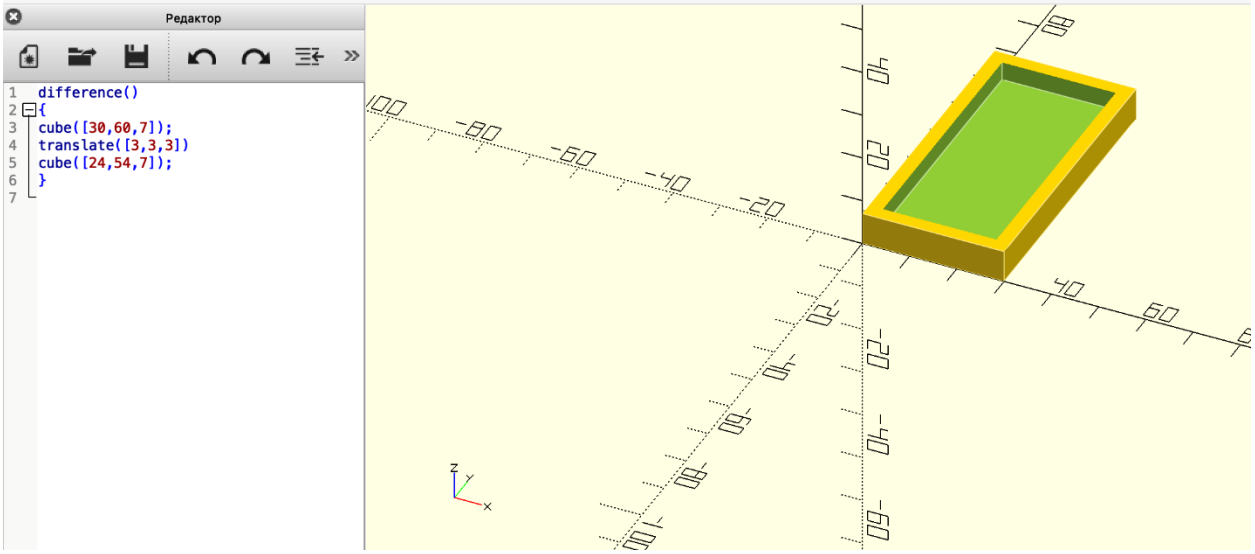


Рис. 26. Парник

ПЕРСПЕКТИВЫ ПРИМЕНЕНИЯ ГИБРИДНЫХ КОМПОЗИЦИОННЫХ МАТЕРИАЛОВ ДЛЯ СОЗДАНИЯ ИНОПЛАНЕТНОЙ ИНФРАСТРУКТУРЫ НА ПОВЕРХНОСТИ ЛУНЫ

Ганиева Дарина Рустемовна

Москва г, ГБОУ МО СП ФМЛ, 11 класс

Секция: Ракетно-космические технологии и средства выведения

Научный руководитель: Малышева Галина Владленовна, к.т.н., профессор МГТУ им. Н.Э. Баумана

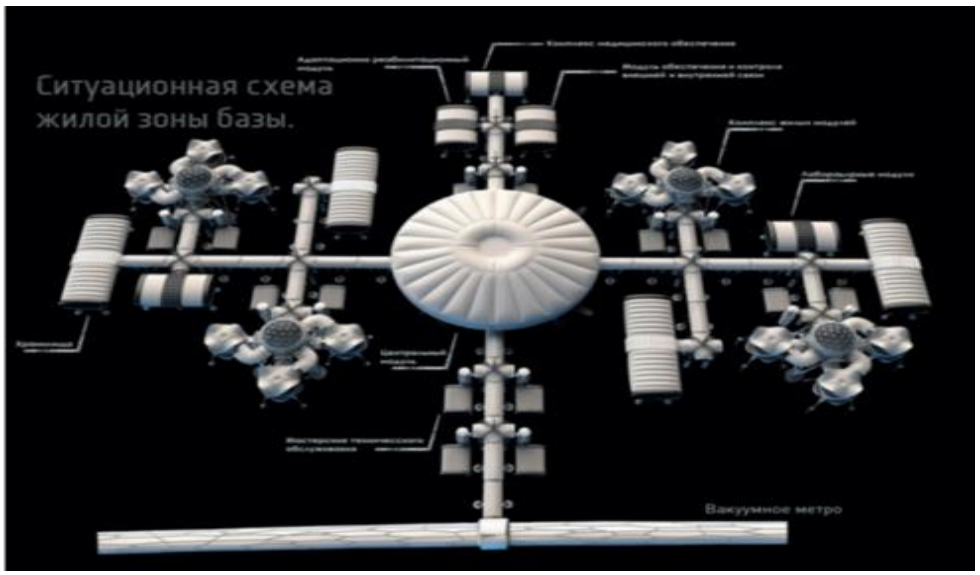
АННОТАЦИЯ

Для обеспечения долгосрочного жизнеспособного существования лунных поселений необходима безотходная модель, основанная на принципах замкнутого цикла. Ключевым решением является максимальное использование местных ресурсов (реголита), и обязательная глубокая переработка упаковки из первичных композиционных материалов. Путем создания на их основе гибридных вторичных композитов с реголитом можно производить элементы инфраструктуры, что снизит зависимость от земных поставок. Реализация этой стратегии требует тщательных материаловедческих исследований, в частности изучения термической стабильности композитов для определения безопасных режимов их переработки непосредственно на Луне.

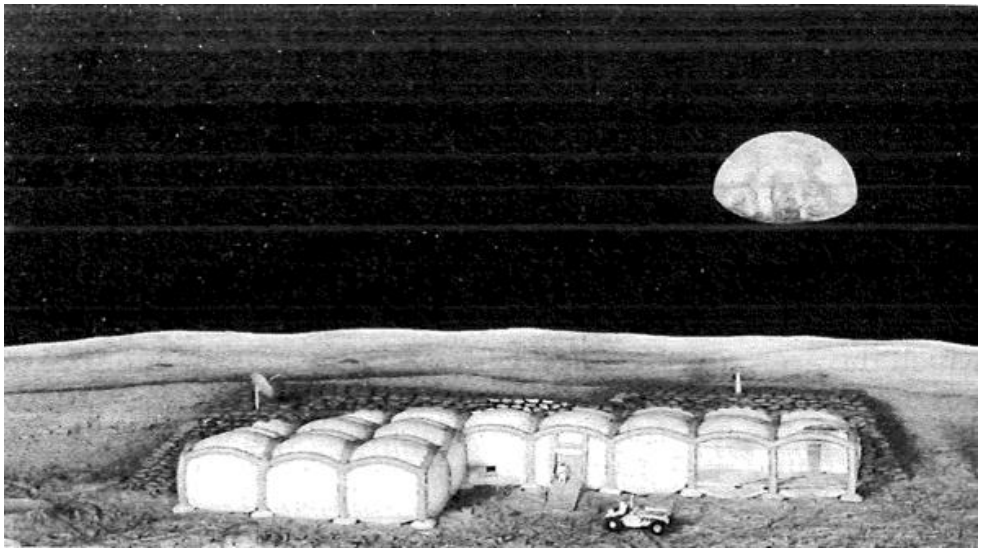
В работе рассмотрены перспективы использования полимерных матриц (полиэтилен низкой и высокой плотности, полипропилен, поливинилхлорид) в сочетании с мелкодисперсным аналогом лунного реголита. Экспериментальная часть включает термогравиметрический анализ полученных образцов в диапазоне температур от 20 до 1000 °C, соответствующих условиям лунного дня. Выполнена теоретическая оценка кинетических параметров термической деструкции. Установлено, что в диапазоне технологических температур формования (до 200 °C) не происходит деструкции полимерной матрицы, что подтверждает принципиальную возможность переработки вторичных композитов непосредственно на поверхности Луны с использованием естественного нагрева. Определен ряд термостойкости исследованных композитов, которые располагаются в следующем порядке: полипропилен, полиэтилен высокой плотности, полиэтилен низкой плотности, поливинилхлорид. Полученные результаты могут быть использованы при проектировании роботизированных комплексов для аддитивного производства строительных элементов лунной инфраструктуры.

ВВЕДЕНИЕ

Актуальность темы исследования: освоение Луны это уже не мечта, а реальная цель, над которой работают учёные и инженеры в России и в других странах. По планам российской космической программы после 2035 года на Луне должна появиться постоянная станция. Стратегически разумно рассматривать не временную базу, а растущее поселение, которое развивается по долгосрочному генеральному плану [1-5]. На рис. 1 представлены проекты лунных баз.



(а)



(б)

Рис. 1. Общий вид лунной базы: а) структурная схема; б) составляющие из надувных компонентов.

Такой подход требует не только технологической зрелости, но и безотходной модели, в которой любой ресурс должен быть возвращён в производственный цикл [6, 7]. Каждое оборудование, материалы и даже продукты питания для лунной базы будут доставлены в

защитной упаковке, которая превращается в обычный мусор. Всё это подчёркивает необходимость комплексного взгляда на лунное строительство и космоэкологию [8].

Стоимость доставки 1 кг груза на Луну составляет около 1 млн. долл. США. Поэтому будущее лунных поселений в принципе безотходности. Это значит, что нужно использовать всё что есть прямо на месте. Главный местный материал лунный грунт или реголит. Из него можно получать полезные вещества, делать дороги и строить стены для поселений, а также использовать в качестве наполнителя для композитов [7, 9]. Интересно, что идеи устойчивого развития по созданию экологичных материалов из возобновляемых или переработанных ресурсов становятся особенно важными в космосе. В условиях нехватки ресурсов и полной зависимости от собственной инфраструктуры экологически сбалансированный подход это не просто забота о природе, а необходимость для выживания.

Таким образом, формирование лунного поселения должно опираться не только на достижения инженерии и робототехники, но и на экологически сбалансированные, ресурсосберегающие принципы, которые сегодня активно развиваются в материаловедении, строительстве и энергетике [10, 11].

В качестве упаковочного материала используют первичные композиционные материалы на основе полиолефинов и поливинилхлорида, которые применяют для изоляции, изготовления поддонов, защитных контейнеров и других элементов, необходимых при доставке грузов на Луну. После использования такие изделия могут быть переработаны с добавлением мелкодисперсного лунного реголита, превращаясь во вторичный композиционный материал для внутренней отделки, перегородок, облицовки стен и полов. Это позволяет реализовать принцип безотходности, сократить зависимость от поставок с Земли.

Для реализации такого проекта необходимо изучить термические свойства композитов, поскольку их поведение при нагреве напрямую влияет на качество и надёжность готовых изделий. Чтобы исключить термическую деструкцию компонентов при переработке и формировании новых изделий, требуется провести термогравиметрический анализ материалов. Этот метод позволяет определить температурные интервалы, в которых материалы стабильны, и подобрать оптимальные режимы переработки без разрушения структуры композита.

Цель исследования заключается в обосновании возможности применения гибридных композитов на основе лунного реголита и полимеров (полиолефинов и поливинилхлорида) для создания элементов лунной инфраструктуры.

Объектами исследования являются гибридные композиты на основе полиолефинов и поливинилхлорида с мелкодисперсным природным наполнителем.

Для достижения цели определены задачи исследования, заключающиеся в изучении состояния вопроса производства гибридных композитных материалом с реголитом и в

исследовании кинетики термической деструкции композитов при тепловом воздействии:

- Анализ современных разработок гибридных композитов на основе термопластов и реголита.
- Прогнозирование теплофизических свойств гибридных композитов с реголитом.
- Исследование кинетики термической деструкции полученных композитов методом термогравиметрического анализа.
- Определение температуры начала термической деструкции как ключевого параметра верхней границы технологического окна формования изделий в режиме дневных температур на поверхности Луны.

СВОЙСТВА ГИБРИДНЫХ КОМПОЗИЦИОННЫХ МАТЕРИАЛОВ

ПОЛИМЕРНЫЕ МАТРИЦЫ

В рамках устойчивого развития перехода к экономике замкнутого цикла переработка термопластов становится приоритетным направлением и служит основой для получения новых композитных материалов с прогнозируемыми функциональными характеристиками. При этом ключевым аспектом является обоснованный выбор матричного полимера, который определяется его структурой и физическими свойствами. Доминирующую роль в производстве упаковки занимают такие полиолефины, как полиэтилен низкой плотности, полиэтилен высокой плотности и полипропилен, а также аморфный поливинилхлорид [12]:

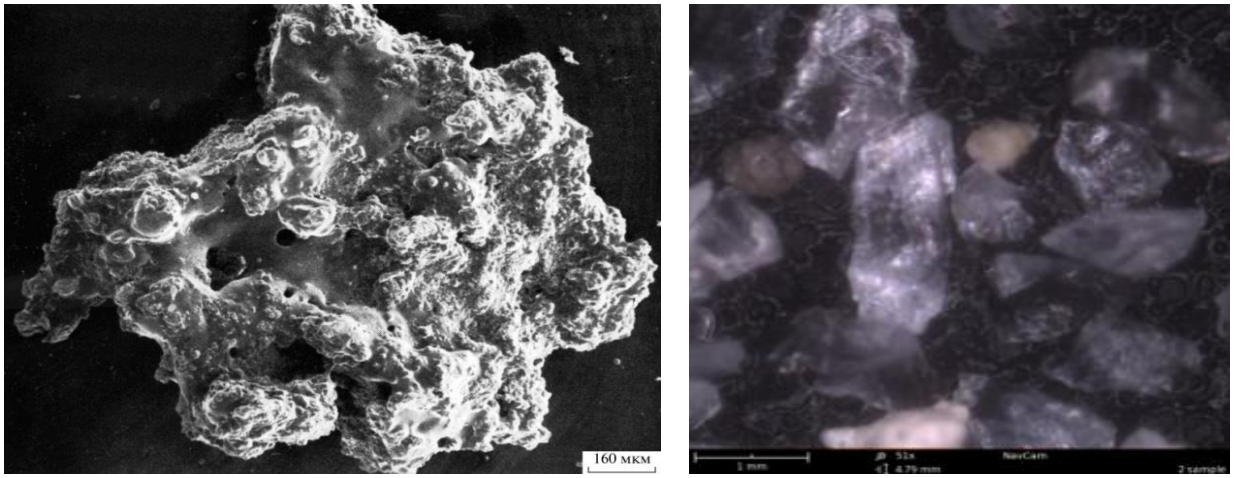
- полиэтилен низкой плотности выбран в качестве объекта исследования ввиду широкой доступности, обладает разветвленной структурой макромолекул, что способствует лучшему удержанию дисперсных наполнителей в объеме матрицы;
- полиэтилен высокой плотности характеризуется линейной структурой и высокой степенью кристалличности, что обеспечивает повышенную жесткость и прочность композитов на его основе, демонстрирует хорошую совместимость с гидрофобными наполнителями и стабильность свойств при многократной переработке, что подтверждается исследованиями композитов с зольными и минеральными добавками;
- полипропилен представляет интерес благодаря оптимальному сочетанию прочности, термостойкости (температура плавления 160–170 °C) и технологичности переработки;
- поливинилхлорид обеспечивает высокую адгезию к неорганическим наполнителям, включая мелкодисперсные оксиды металлов и базальтовые микроволокна.

Таким образом, выбранные термопласты могут быть использованы для разработки технологий рециклинга и создания новых композитов с мелкодисперсными наполнителями.

РЕГОЛИТ

Освоение Луны становится важной задачей для российской науки и всего мира. Страны заинтересованы в этом, потому что на Луне много редкоземельных металлов, которые могут быть использованы для развития технологий и промышленности.

Поверхность Луны покрывает рыхлый слой раздробленных горных пород, который называется реголитом. Он формировался на протяжении миллиардов лет под действием метеоритов и космического выветривания. Реголит состоит из мелкой пыли, обломков камней и стеклянных частиц, отличающихся острыми краями. Этот грунт представляет ценность как источник полезных ресурсов. На рис.2 представлен вид лунного реголита под микроскопом.



(а) *(б)*

Рис. 2. Реголит: а) частица лунного грунта; б) лабораторный аналог грунта.

Использование подлинного лунного реголита ограничено его высокой стоимостью и дефицитом при создании новых композитных материалов. В качестве аналога был подобран мелкодисперсный природный аналог. Обоснованием такого выбора послужила идентичность размеров его частиц частицам реголита. Таким образом, удалось создать доступную модель грунта для проведения предварительные испытания, не расходуя ценный инопланетный материал.

ПРОГНОЗИРОВАНИЕ ТЕПЛОФИЗИЧЕСКИХ СВОЙСТВ КОМПОЗИТОВ

Понимание теплофизических свойств композиционных материалов имеет ключевое значение для практического применения. Эти параметры описывают способность материала передавать тепло, выравнивать температуру и накапливать тепловую энергию. Для композитов, создаваемых на основе связующего и наполнителя, корректный прогноз этих характеристик необходим при эксплуатации в условиях перепадов температур.

При определении теоретического значения эффективной теплопроводности использовалась методика Г.Н. Дульнева и Ю.П. Заричняка, предназначенная для расчёта коэффициента теплопроводности структур с взаимопроникающими компонента, который

является надежным способом рассчитать свойства сложных материалов методом обобщенной проводимости [13]. Суть данного расчёта состоит в последовательном рассмотрении бинарных систем и анализе изменения коэффициентов теплопроводности при поэтапном введении каждого последующего компонента в исходный материал при необходимости. Метод анализирует типичный элемент композита: сколько в нем компонентов, какой они формы и как расположены друг относительно друга, что особенно важно для смесей с частицами лунного грунта, благодаря этому расчеты хорошо совпадают с реальными экспериментами.

Зависимость для определения обобщённой проводимости представлена в виде формулы 1. Для вычисления эффективного коэффициента теплопроводности использовалось специализированное программное обеспечение, позволяющее автоматизировать расчёты и повысить точность прогнозирования теплофизических характеристик гетерогенных систем.

$$\lambda_{эфф} = \lambda_1 \cdot \left[B_r^{\,2} + \frac{\lambda_2}{\lambda_1} \cdot \left(1 - B_r\right)^2 + 2 \cdot \frac{\lambda_2}{\lambda_1} \cdot \frac{B_r \cdot (1 - B_r)}{(\frac{\lambda_2}{\lambda_1} \cdot B_r + 1 - B_r)} \right] \quad (1)$$

$$B_r = 0,5 + A \cdot \cos(\frac{\varphi}{3}),$$

$$A = 1,\ \varphi = 2 \cdot \pi - \arccos(2 \cdot r_2 - 1) \text{ при } 0,5 \le r_2 \le 1,$$

$$A = -1,\ \varphi = 2 \cdot \pi - \arccos(1 - 2 \cdot r_2) \text{ при } 0 \le r_2 \le 0,5,$$

где λ_1, λ_2 -теплопроводности компонентов бинарной смеси;

r_2 -объемная концентрация добавляемого компонента;

B_r - относительный размер бруса ячейки при адиабатическом делении;

A-параметр, характеризующий влияние включений на теплопроводность;

φ - параметр, отражающий влияние структуры на теплопроводность.

В табл.1 указаны теплофизические свойства компонентов и предполагаемых на их основе композитов.

Таблица 1. Теплофизические характеристики реголита, термопластов и композитов на их основе.

Материал (50/50)	Плотность, ρ, кг/м³	Теплопроводность, Вт/(м·К)	Теплоемкость, c, Дж/(кг·К)
Реголит	1500	0,02 (0,002) *	800
ПЭНП	930	0,35	2200
ПЭВП	970	0,41	2300

ПП	910	0,24	2000
ПВХ	1450	0,19	1600
Композит1	1148	0,142 (0,107)	1664
Композит2	1178	0,159 (0,122)	1711
Композит3	1133	0,106 (0,084)	1547
Композит4	1475	0,072 (0,051)	1207

*-теплопроводность реголита на поверхности луны

По результатам расчёта теплофизических параметров установлено, что значения плотности и теплоёмкости композитов подчиняются правилу аддитивности. Несмотря на низкую теплопроводность реголита, его введение в полимерную матрицу приводит к умеренному снижению коэффициента теплопроводности композита относительно чистого полимера.

МЕТОДЫ ФОРМИРОВАНИЯ ИЗДЕЛИЙ В ДИАПАЗОНЕ ТЕМПЕРАТУР ЛУННОГО ДНЯ

Чтобы создать на Луне базы для жизни и хранения грузов разрабатываются методы формования изделий из лунного грунта с полимерными связующими. Методом плавления из реголита с помощью специализированной техники возможно изготовление строительных блоков лунного поселения. Рассматривается проект роботизированного комплекса, в состав которого входит 3Д-принтер. Для производства композитного материала на основе вторичной полимерной упаковки и лунного реголита целесообразно применение экструзионного шнекового гранулятора, который обеспечивает эффективное смешивание компонентов и подготовку сырья для аддитивного и формовочного производства изделий.

Источником получения полиолефинов и поливинилхлорида в условиях лунной базы является исключительно вторичное полимерное сырье, доставляемое с Земли в составе грузовых миссий в виде упаковочных материалов и тары, поскольку на Луне отсутствуют природные запасы данных полимеров. Что касается необходимого объема для целесообразности их вторичной переработки, то применение экструзионного шнекового гранулятора придает процессу накопительный характер до удобного момента для формования из них запланированных изделий, что позволяет осуществлять переработку поступающих отходов по мере их появления и хранить гранулят для единовременного запуска производства, делая технологию простой даже при постепенном поступлении сырья.

Исходное полимерное сырьё рекомендуется предварительно нагревать до 150°С в зависимости от типа термопласта, что соответствует естественному температурному режиму

лунного дня и позволяет использовать солнечную энергию для термической подготовки. Наполнитель не требует предварительного нагрева. Такой подход минимизирует энергопотребление и делает технологию рентабельной для внеземного производства. На рис.3 указан общий вид подобного оборудования.

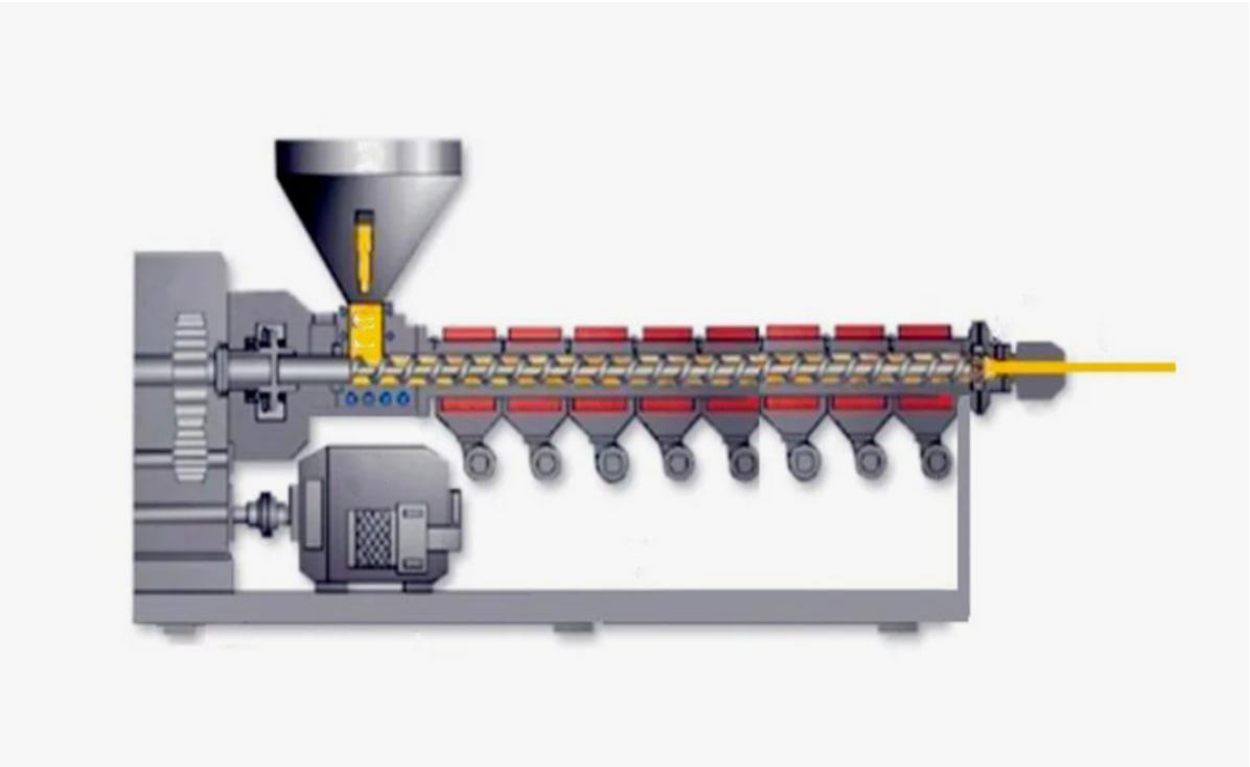


Рис. 3. Экструзионный шнековый гранулятор

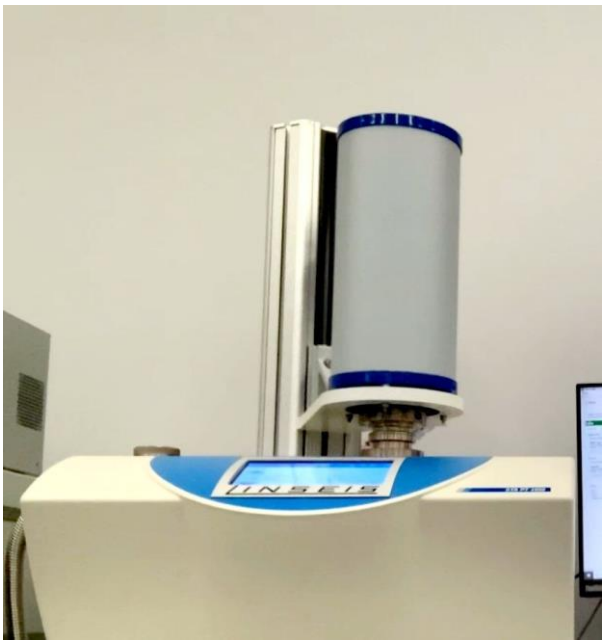
Накопленный таким образом композитное сырье на основе переработанных полиолефинов и поливинилхлорида в качестве связующего для гибридных композиционных материалов с лунным реголитом может быть использован для изготовления широкой номенклатуры элементов инфраструктуры лунного поселения. К числу таких изделий относятся трубопроводы систем жизнеобеспечения, водоотведения и вентиляции, кабельные каналы и электроизоляционные конструкции, напольные покрытия, стеновые панели и модульные перегородки, а также ёмкости для хранения воды, технических жидкостей и грузов.

В данных применениях полимерная матрица обеспечивает необходимую герметичность, эластичность и химическую стойкость, тогда как дисперсный реголитный наполнитель придаёт конструкционную прочность и износостойкость, что в совокупности создаёт технологически обоснованный и ресурсоэффективный материал для устойчивого функционирования лунной базы.

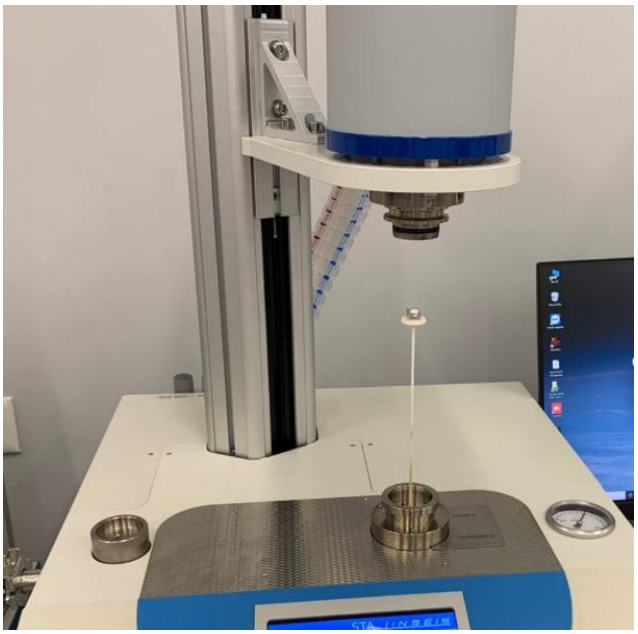
ЭКСПЕРИМЕНТАЛЬНАЯ ОЦЕНКА ТЕРМОСТОЙКОСТИ ГИБРИДНЫХ КОМПОЗИЦИОННЫХ МАТЕРИАЛОВ

МЕТОД ТЕРМОГРАВИМЕТРИЧЕСКОГО АНАЛИЗА

Для исследования использовался совмещенный термический термоанализатор STA PT1600 фирмы Linseis Messgeaete GmbH. Это универсальный исследовательский инструмент для измерения термофизических свойств: уменьшение или увеличение веса образца при контролируемом изменении температуры (ТГ); дифференциальный термический анализ (ДТА); дифференциальная сканирующая калориметрия (ДСК); сопряжение с системами АВГ (анализ выделяющихся газов) ИК-Фурье или масс-спектрометром. Прибор регистрирует несколько сигналов одновременно в ходе одного эксперимента, что позволяет специалисту сопоставлять данные об изменении массы, тепловых эффектах и газовыделении, что значительно упрощает понимание процессов и делает выводы более достоверными и повышает точность научного анализа. Общий вид оборудования показан на рис.4.



(а)

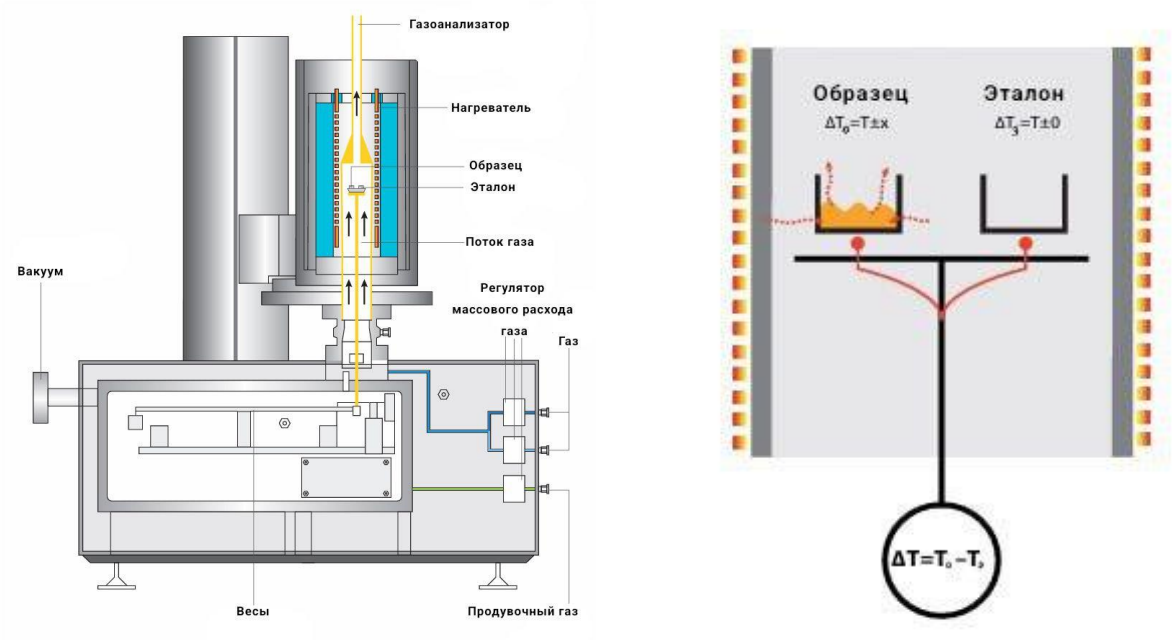


(б)

Рис. 4. Термоанализатор STA PT1600: а) общий внешний вид спереди; б) вид сверху, печная камера зафиксирована в верхнем положении.

Основа весов в анализаторе является рычаг. Высокочувствительные оптические преобразователи фиксируют перемещение рычага весов. Они наводят токи в магнитных катушках, которые компенсируют изменения веса, поддерживая рычаг весов в постоянном нулевом положении. Этот ток усиливается и воспроизводит ТГА-сигнал. Магнитные катушки используются также для электронного тарирования системы. Простая и надежная конструкция весов позволяет с высокой точностью определять изменения веса образцов в зависимости от температуры или времени в условиях контролируемого нагрева или охлаждения в

температурном диапазоне от -150 до 1750°С. Прибор обеспечивает точность измерения массы до 0,1 мкг и позволяет проводить исследования в различных газовых средах с контролируемым расходом газа. На рис.5 указано схематическое изображение оборудования.



(а) *(б)*

Рис. 5. Система термоанализа с анализатором газа: а) схема термоанализатора и газовых потоков; б) принцип измерения температурной разницы между образцом и эталоном.

На схеме рис.5(а) показано устройство термоанализатора и пути движения газовых потоков. Здесь можно проследить, как анализируемый газ поступает в камеру, взаимодействует с образцом, а затем направляется в анализатор газа для идентификации компонентов. Также обозначены основные узлы: высокоточные весы, печь, держатель образца и линии газовых потоков. На схеме видна система вакуумирования, обозначенная слева, что позволяет проводить исследования при пониженном давлении, а также регулятор массового расхода газа, обеспечивающий точный контроль газовой среды в процессе эксперимента. Система продувки, при которой продувочный газ подается снизу, защищает весовую систему от попадания продуктов разложения. Важной конструктивной особенностью является изоляция весов от высокотемпературной зоны. Весы расположены в нижней части прибора и термически защищены от печи, что обеспечивает стабильность измерений.

Схема рис.5(б) демонстрирует принцип измерения температуры тигля с образцом и эталоном. На схеме показано расположение термопар, одна из которых измеряет температуру исследуемого вещества, а другая контролирует инертный эталон. В процессе одновременного нагрева двух тиглей до температур, при которых происходит выделение или поглощение исследуемым образцом теплоты. Это регистрируется датчиком теплового потока, который измеряет разность тепловых потоков между тиглей. Зона образца, где происходят фазовые

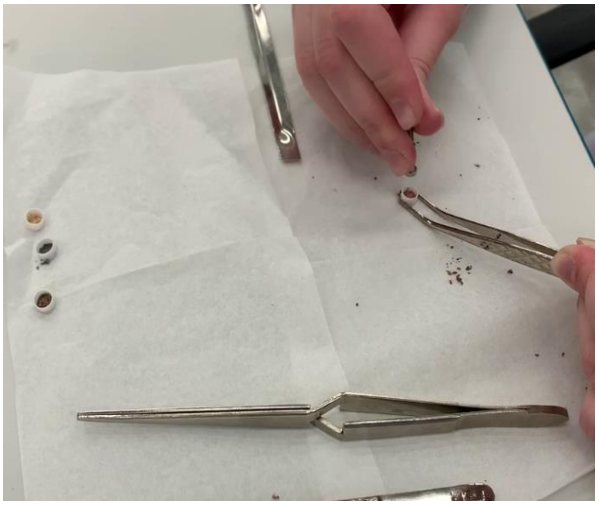
превращения и химические реакции, выделена оранжевым цветом.

Перед проведением анализа производится калибровка прибора. Используется набор калибровочных веществ (In, Sn, Ga, Pb, Zn, BaCO3, SrCO3, K2SO3), которые прописаны в файле DSCCAL.INI. Дополнительные вещества могут быть добавлены при необходимости и должны включать температурный диапазон, желаемый для измерений образца. Задание условий проведения опыта, управление прибором и запись получаемых данных на жесткий диск компьютера осуществляли при помощи программного обеспечения LINSEIS TAWIN. Для калибровки по массе применялись эталонные грузики с известной массой в диапазоне от 1 до 100 мг, что обеспечивало погрешность измерения массы не более ±0,1%.

МЕТОДИКА ОЦЕНКИ ТЕРМОСТОЙКОСТИ

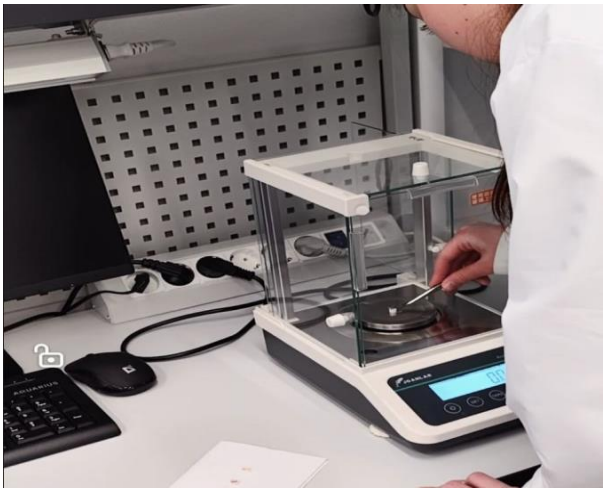
Термогравиметрические исследования проводились по ГОСТ Р 56721-2015 (ИСО 11358-1:2014). Перед исследованием образцы композитов подготовили в виде мелкой стружки массой около 15 мг и равномерно распределили по дну цилиндрического тигля из спечённого оксида алюминия. Толщина слоя навески составляла 10–15 % от общего объёма тигля. На рис.6 изображен процесс подготовления образцов для исследования.

(а)



(б)

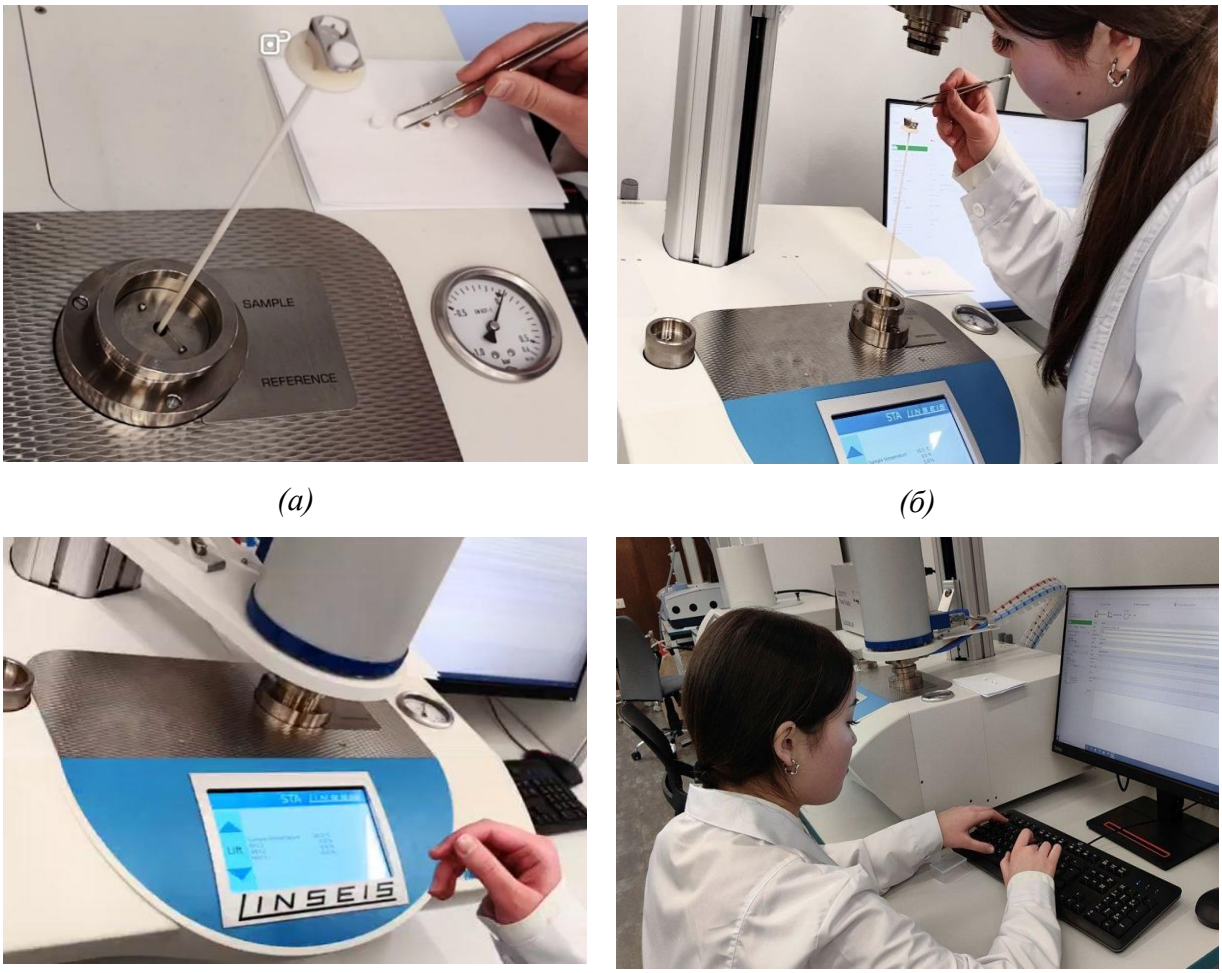
(в)



(г)

Рис. 6. Подготовка образцов к исследованию: а) стружка из композита; б) уплотнение образца в тигле; в) распределение стружки в тигле из оксида алюминия; г) замер на высокоточных весах.

Термический анализ проводили в диапазоне температур 20–1000 °С при постоянной скорости нагрева 10 °С/мин в защитной атмосфере инертного газа (аргон). На рис.7 изображен процесс запуска термогравиметрического анализа композита.



(а) *(б)*

(в) *(г)*

Рис. 7. Загрузка тиглей с образцами в термоанализатор: а) измерительная система; б) загрузка тигля в печь оборудования; в) запуск эксперимента после закрытия печной камеры; г) ввод данных в программное обеспечение.

Во время измерения на экране отображается график, где по оси X время или температура, по оси Y изменение массы. Начальный масштаб времени до 10 мин. При выходе данных за пределы шкалы масштаб автоматически корректируется, чтобы отобразить все значения. График строится с самого начала измерения и полностью сохраняется на экране. Под графиком показываются текущие числовые данные. Количество точек в массивах составляет несколько десятков тысяч. После завершения обработки выполняется визуальная проверка на отсутствие смещений. Затем обработанные массивы экспортируются в текстовые файлы. На рис.8.приведены графики изменения массы образцов.

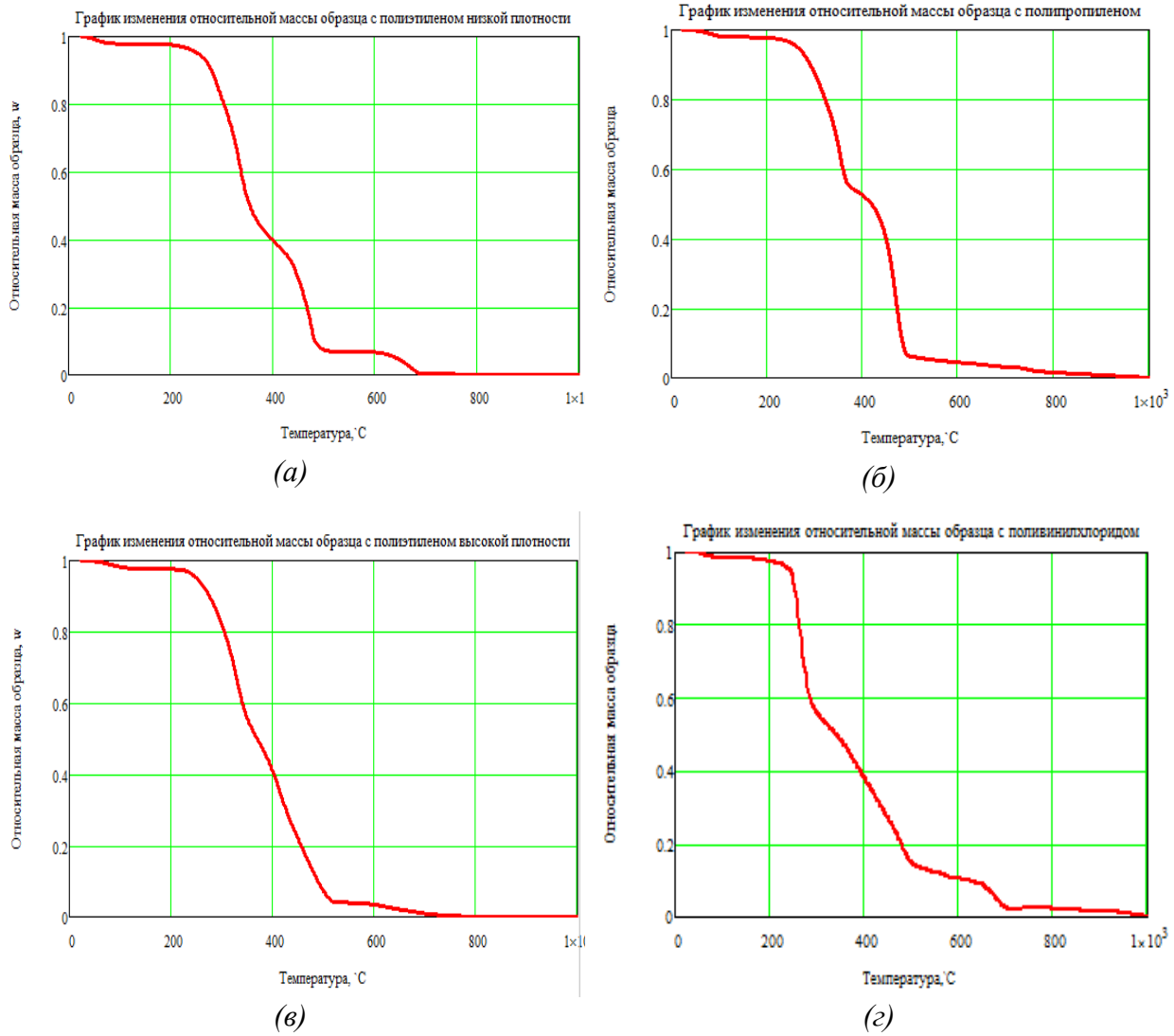


Рис. 8. График изменения относительной массы образцов от температуры композитов с: а) полиэтиленом низкой плотности; б) полиэтиленом высокой плотности; в) полипропиленом; г) поливинилхлоридом.

ТЕОРЕТИЧСКАЯ ОЦЕНКА СКОРОСТИ ХИМИЧЕКСИХ РЕАКЦИЙ

Теоретическая оценка скорости химических реакций применительно к термогравиметрическому анализ у композитов на основе термопластов позволяет перейти от простой регистрации потери массы к количественному описанию кинетики термической деструкции материала. Скорость реакции разложения зависит от температуры по уравнению Аррениуса, указанного в форм.2. По результатам ТГА могут быть рассчитаны: частотный фактор (A), энергия активации стадий (E), начальная относительная масса стадий (ω).

Для композитов на основе термопластов такая оценка позволяет прогнозировать их поведение в условиях высокотемпературной эксплуатации и даже идентифицировать отдельные стадии разложения многокомпонентной системы.

$$\frac{d\omega_j}{d\tau} = \sum_{j=1}^{m} \omega_j^{n_j} \cdot A_j \cdot \exp\left(-\frac{E_j}{R \cdot T}\right) \quad (2)$$

Где:

- j – номер стадии;
- m – номер и число стадии;
- T – температура, К;
- nj – порядок реакции j-й стадии;
- R – универсальная газовая постоянная, Дж/(моль*К);
- Ej–энергия активации j-й стадии, Дж/моль;
- Aj– частотный фактор j-й стадии, c^{-1};
- ω_j – безразмерная масса стадии.

Кинетические параметры E_j, A_j, ω_j описывают скорость химических реакции j-й стадии. Анализ кинетических параметров первых трех стадий термического разложения исследованных композитов выявил характерные особенности деструкции для каждого типа полимерной матрицы. Для всех изученных материалов наблюдается закономерное увеличение энергии активации и частотного фактора при переходе от начальных стадий к последующим, что свидетельствует о смене механизмов разложения по мере повышения температуры.

ИССЛЕДОВАНИЕ ВЛИЯНИЯ ТИПА ПОЛИМЕРА НА ТЕРМОСТОЙКОСТЬ

Первая стадия для всех композитов характеризуется минимальной потерей массы, что соответствует удалению летучих компонентов и низкомолекулярных примесей. Вторая стадия демонстрирует существенное возрастание как энергии активации, так и частотного фактора, сопровождаясь значительной потерей массы, что указывает на начало интенсивной деструкции полимерной матрицы. Третья стадия характеризуется дальнейшим ростом кинетических параметров, достигая максимальных значений энергии активации и частотного фактора, что отражает наиболее глубокие химические превращения в системе. При этом композиты на основе различных полимеров демонстрируют различную интенсивность массопотери на каждой стадии, что обусловлено спецификой химического строения полимерных цепей и их взаимодействием с наполнителем. Распределение относительной массы по стадиям существенно различается для разных типов композитов, что отражает различия в термической стабильности и механизмах термической деструкции полимерных матриц.

Анализ температурных характеристик показал, что композит на основе полипропилена демонстрирует наилучшую термическую стабильность с разложением основной массы и наиболее высокими температурами максимальной скорости разложения среди исследованных материалов.

Композиты на основе полиэтилена характеризуются близкими показателями термостойкости с температурами максимальной скорости разложения, причём полиэтилен высокого плотности проявляет несколько большую стабильность при высоких температурах.

Композит на основе поливинилхлорида обладает наименьшей термостойкостью, однако последующее образование углеродистых структур повышает устойчивость на третьей стадии.

На основании представленных данных термогравиметрического анализа и расчетных теплофизических характеристик установлено, что тип полимерной матрицы оказывает определяющее влияние на характер термической деструкции и эксплуатационную стойкость гибридных композитов. Для композитов на основе полиолефинов наблюдается схожий профиль термической стабильности, характеризующийся одностадийным механизмом разложения. Интенсивная потеря массы начинается с 250°C и высокие значения удельной теплоемкости и теплопроводности в составе композитов обеспечивают материалу способность аккумулировать тепловую энергию. В отличие от полиолефинов, композит на основе поливинилхлорида демонстрирует другой механизм термической деградации и обладает наименьшими значениями теплопроводности и теплоемкости, а также максимальной плотностью, приближающейся к плотности чистого реголита.

Таким образом, выбор полимерного связующего позволяет варьировать свойства материала под конкретные задачи лунной инфраструктуры: полиолефины целесообразно применять для элементов, требующих высокой удельной прочности и демпфирования тепловых ударов за счет высокой теплоемкости, тогда как поливинилхлорид предпочтителен для конструкций, где критичны высокие теплоизоляционные свойства, радиационная защита за счет высокой плотности и сохранение формы при длительном высокотемпературном воздействии благодаря коксованию. Введение лунного реголита в обоих случаях эффективно снижает горючесть полимеров и стабилизирует их теплофизические параметры.

ЗАКЛЮЧЕНИЕ

В ходе выполнения работы было проведено комплексное исследование возможности применения гибридных композиционных материалов на основе вторичных термопластов и мелкодисперсного наполнителя (аналога реголита) для создания элементов инфраструктуры на Луне. При анализе изменения кривой потери массы получено, что в диапазоне температур лунного дня (до 150–200°C) в исследованных композитах не происходит термическая деструкция компонентов, и формование изделий технологически возможно без разрушения полимерной матрицы. Это подтверждает принципиальную пригодность выбранных материалов для переработки непосредственно на месте с использованием естественного солнечного нагрева.

Методом термогравиметрического анализа установлены температурные интервалы стабильности для каждого типа композита. Полипропиленовые композиты продемонстрировали наилучшую термическую стабильность, что делает их предпочтительными для изготовления конструкционных элементов, подвергающихся нагреву. Композиты на основе поливинилхлорида, несмотря на более раннее начало деструкции, могут быть использованы для изделий, не требующих высокой термостойкости, либо в составе многослойных структур.

Полученные кинетические параметры и теплофизические параметры позволяют прогнозировать поведение материалов в реальных условиях эксплуатации и оптимизировать режимы переработки. Результаты работы обосновывают возможность реализации замкнутого цикла использования полимерных отходов на лунной базе: от защитной упаковки до строительных блоков, отделочных плит и элементов интерьера. Это существенно сокращает зависимость от дорогостоящих земных поставок и соответствует принципам космоэкологии и устойчивого развития.

В перспективе требуется проведение дополнительных исследований в условиях, максимально приближенных к лунным (вакуум, циклические перепады температур, радиация), а также разработка технологического регламента для роботизированных экструзионных комплексов. Предложенный подход может стать основой для создания безотходной инфраструктуры будущих лунных поселений, закладывая фундамент для долговременного присутствия человека на других небесных телах.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кричевский, С.В. Перспективы освоения космоса человеком. М.: Леланд, – 2021. – 320 с.
2. Круцков, В.В. Проблемы перехода от исследований Луны к её освоению / В.В. Круцков, А.В. Куликов // Космическая техника и технологии. – 2021. –№ 4(33). – С. 41-49.
3. Багров, А.В. Проблемы перехода от исследований Луны к её освоению / А.В. Багров, В.А. Леонов // ВКС. – 2020. – №3. – С. 22-33.
4. Багров А.В. Сценарий перманентного строительства лунного поселения / А. В. Багров, В. А. Леонов // Научное значение трудов К.Э. Циолковского: история и современность: Материалы 55-х Научных чтений памяти К.Э. Циолковского, Калуга, 15–17 сентября 2020 года. Том Часть 2. – Калуга: Индивидуальный предприниматель Стрельцов Илья Анатольевич, 2020. – С. 76-79.
5. Леонов В.А. Постоянная лунная станция как приоритет России в освоении ресурсов космоса // Воздушно-космическая сфера. 2021. №4. С. 56-76.
6. Емельянов, К.В. Порошкообразная высоконаполненная полимерная композиция на основе имитатора лунного грунта для изготовления строительного материала на Луне / К. В. Емельянов // Сборник трудов научно-исследовательских работ студентов МАИ: Материалы Международной молодёжной конференции, Москва, 13–16 апреля 2023 года. – Москва: Издательство "Перо", 2023. – С. 103-109.
7. Венгловская, Е.Н. Этапы строительства поселения на Луне / Е. Н. Венгловская // Научное значение трудов К.Э. Циолковского: история и современность: Материалы 55-х Научных чтений памяти К.Э. Циолковского, Калуга, 15–17 сентября 2020 года. Том Часть 2. – Калуга: Индивидуальный предприниматель Стрельцов Илья Анатольевич, 2020. – С. 89-91.
8. Семячков, А.И. Космоэкология / А.И. Семячков, К. Дребенштедт, В. Л. Берсенев. – Екатеринбург: Институт экономики УрО РАН, 2024. – 148 с.
9. Пхон, Т.Ч. Изучение структуры аналога лунного грунта и кинетики процессов растекания жидкостей по его поверхности / Т.Ч. Пхон, М.В. Худорожко, Г.В. Малышева // Дизайн. Материалы. Технология. – 2025. – № 2(78). – С. 102-108. – DOI 10.46418/1990-8997_2025_2(78)_102_108.
10. Энергообеспечение обитаемых лунных баз / В. Н. Акимов, Я. Б. Данилевич, А. Н. Коваленко [и др.] // Известия Российской академии наук. Энергетика. – 2009. – № 1. – С. 96-112.

11. Абдулин, В.В. 3D принтеры в строительстве, технология conturcrafting. Строительство на луне / В.В. Абдулин, К.С. Латухин, М.Ф. Амоян // Ресурсоэнергоэффективные технологии в строительном комплексе региона. – 2018. – № 9. – С. 464-467.
12. Ерохин, н. Критерии выбора матрицы при создании нанокомпозитов на основе термопластов / н. Ерохин, Л. В. Козырева // Вестник Федерального государственного образовательного учреждения высшего профессионального образования "Московский государственный агроинженерный университет имени В.П. Горячкина". – 2010. – № 2(41). – С. 97-101.
13. Дульнев Г.Н., Заричняк Ю.П. Теплопроводность смесей и композиционных материалов. Справочная книга. Л.: Энергия, 1974. 264 с.

ЭКОЛОГИЧЕСКИЕ ПОКАЗАТЕЛИ ДИЗЕЛЬНОГО ДВИГАТЕЛЯ, РАБОТАЮЩЕГО НА СМЕСЯХ НЕФТЯНОГО ДИЗЕЛЬНОГО ТОПЛИВА И ОЛИВКОВОГО МАСЛА

Егоров Александр Кириллович

Москва г, Школа «Покровский квартал», 10 класс

Секция: Энергетическое машиностроение

Научный руководитель: Марков Владимир Анатольевич, д.т.н., заведующий кафедрой Э2 МГТУ им. Н.Э. Баумана

АННОТАЦИЯ

В настоящее время наблюдается значительный интерес к альтернативным видам биотоплива, таким как различные органические масла (например, оливковое, подсолнечное). Особую актуальность это направление приобретает для утилизации отработанного фритюрного масла, ежегодный объем которого в России превышает 68 тысяч тонн**.** Его можно использовать в качестве добавки к дизельному топливу.

В работе проанализирован состав масла и его физические и химические изменения после использования во фритюре. Показано, что его свойства значительно меняются. Например, вязкость увеличивается в примерно в 2 раза, что необходимо учитывать при его добавке к дизельному топливу. Также проанализирована концентрация полициклических ароматических углеводородов (ПАУ) в отработанном фритюрном масле (показано, что концентрация ПАУ очень мала).

В рамках исследования проведены эксперименты по изучению экологических показателей дизельного двигателя, работающего на смеси дизельного топлива и оливкового масла. В результате установлено, что двигатель обладает лучшей топливной экономичностью, меньшей дымностью и меньшей концентрацией оксида углерода. Однако отмечено повышение концентрации монооксида углерода и оксидов азота.

ВВЕДЕНИЕ

На современном этапе развития двигателестроения проводятся многочисленные исследования возможностей использования в двигателях внутреннего сгорания различных альтернативных топлив [1-4]. Это объясняется не столько дефицитом традиционных нефтяных топлив, сколько поиском экологически чистых сырьевых ресурсов для производства альтернативных топлив. С этой точки зрения перспективно использование в ДВС топлив растительного происхождения (биотоплив) [5-10], что является актуальной задачей. Они относятся к малоуглеродным топливам, в молекулах которых содержится пониженное по сравнению с нефтяными топливами атомов углерода. Использование таких топлив в ДВС будет способствовать переходу к углеродно нейтральной энергетике. Кроме того, при выращивании сырья для производства биотоплив выделяется примерно такое же количество кислорода, которое потребляется при их сжигании в ДВС. В результате обеспечивает кругооборот углекислого газа CO_2 в атмосфере.

К перспективным биотопливам для дизельных двигателей относятся растительные масла и получаемые из них метиловые, этиловые и бутиловые эфиры [11-18]. При этом использование чистых растительных масел и их смесей с нефтяным дизельным топливом (ДТ) целесообразно в агропромышленных комплексах, где имеются избытки растительных масел и не требуются затраты на их транспортировку на перерабатывающие предприятия, а применение указанных эфиров растительных масел предпочтительнее в городских условиях [19]. Одним из положительных качеств чистых растительных масел является их практически полная биоразлагаемость в течение нескольких недель в почве и водоемах [20].

В 2025 году мировое производство всех растительных масел составило около 235 млн. тонн. При этом мировые объемы производства основных растительных масел примерно составили: пальмового масла – около 80 млн. тонн, соевого – примерно 70 млн. тонн, рапсового - около 35 млн. тонн, подсолнечного – примерно 22 млн. тонн [21].

Среди других видов растительных масел следует отметить оливковое масло (ОМ), получаемое из плодов оливы европейской (Olea europaea). Оно широко используемое в пищевой промышленности и системе общественного питания. Мировое производство ОМ в 2025 году достигло 3,5 млн. тонн [21]. При этом на долю стран, не входящих в ЕС, пришлось 40% мирового производства ОМ. Оливковое масло производят 58 стран на пяти континентах. Это масло традиционно производится в Испании, Италии, Греции. Среди новых участников рынка ОМ следует отметить Сальвадор, Эфиопию, Кувейт, Узбекистан, Азербайджан и Северную Македонию.

Следует также отметить, что применяются различные способы производства растительных масел, в частности ОМ [19]. Оливковое масло первого отжима получают из

оливок путём их механического холодного прессования. Обычно масло первого отжима может составлять до 80-85% от общего объёма производства. Из оливкового жмыха, оставшегося после холодного отжима масла, горячим прессованием извлекают дополнительное количество масла низкого качества («деревянное масло»). В оставшемся жмыхе еще содержится небольшое количество масла (5–10%), после производится дробление плодов с последующей экстракцией полученной массы химическими растворителями (чаще - нормальным гексаном). Экстракция позволяет довести долю извлекаемого из сырья ОМ до 99%. Однако полученное масло содержит остатки растворителя и примеси, поэтому его нельзя использовать в пищу. ОМ, полученное горячим прессованием и экстракцией, можно использовать в качестве моторного топлива.

Интерес к оливковому маслу как экологической добавки к нефтяному ДТ обусловлен еще и тем, что это масло широко используется как фритюрное масло. Наиболее распространенными фритюрными маслами являются пальмовое, соевое, подсолнечное и оливковое [22]. Указанные отработанные фритюрные масла могут быть использованы для производства экологически чистых моторных топлив (эфиров растительных масел) [23-25]. Эти масла являются значимым сырьевым ресурсом для производства моторных топлив. В частности, в Японии ежегодные отходы фритюрных растительных масел составляют от 400 до 600 тыс. т в год [26]. По данным на июль 2025 года, ежегодно в России собирают более 68 тыс. тонн отработанных фритюрных масел [27].

Исследованию свойств оливкового масла посвящены работы [28-31]. Проведены испытания двигателей, работающих на оливковом масле и его смесях с нефтяным ДТ, эфирах оливкового масла и смесевых топливах на их основе [32-39]. Вместе с тем, недостаточно изученными являются экологические характеристики дизельных двигателей, работающих на смесях нефтяного ДТ с оливковым маслом (ОМ), а также проблемы определения оптимального состава таких смесевых биотоплив.

Цель: исследовать работу дизельного двигателя, который использует в качестве топлива смеси ДТ с ОМ, а также определить оптимальный состав таких смесевых биотоплив.

Задачи:

- изучение химического состава оливкового масла (до и после фритюра) и дизельного топлива;
- проведения экспериментов;
- определение вредных веществ;
- определение интегрального критерия, характеризующего токсичность отработавших газов.

ФИЗИКО-ХИМИЧЕСКИЕ И МОТОРНЫЕ СВОЙСТВА РАСТИТЕЛЬНЫХ МАСЕЛ

Возможность использования растительных масел и их производных в качестве топлива для дизелей определяется их свойствами. Растительные масла состоят в основном (на 95-97%) из триацилглицеридов – органических соединений, сложных эфиров глицерина и различных жирных кислот, а также моно- и диацилглицеридов. Ацилглицериды, в свою очередь, содержат в своем составе молекулы различных жирных (карбоновых) кислот (в частности, стеариновой кислоты с формулой **$C_{17}H_{35}COOH$** или **CH_3-$(CH_2)_{16}$-COOH)**, связанных с молекулой глицерина $C_3H_5(OH)_3$ [19]. При этом жирнокислотный состав растительных масел включает как насыщенные кислоты (пальмитиновая, стеариновая, арахиновая, миристиновая кислоты), так и ненасыщенные жирные кислоты (линолевая, олеиновая, линоленовая кислоты).

В молекулах ацилглицеридов до трех молекул жирных кислот соединены с молекулой глицерина при помощи эфирных связей. Моноглицериды представляют собой моноэфиры глицерина и жирной кислоты. Они существуют в двух изомерных формах (рис. 1, *а*) [19]. Диглицериды – это диэфиры глицерина и жирных кислот. Их химические структуры представлены на рис. 1, *б*. Триглицериды являются триэфирами глицерина и жирных кислот, причем входящие в состав триглицеридов жирные кислоты могут быть одинаковыми и различными (рис. 1, *в*). На рис. 1, *г* показана структура триглицерида, образованного тремя наиболее характерными для растительных масел жирными кислотами – стеариновой, олеиновой и линолевой.

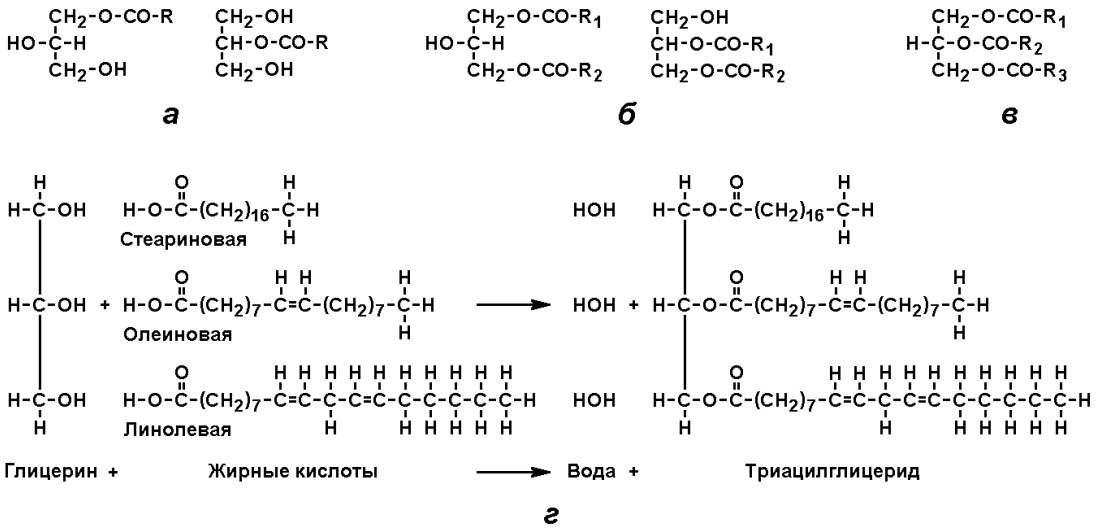


Рис. 1. Структурные формулы моноглицеридов (а), диглицеридов (б), триглицеридов (в) и схема образования триацилглицеридов, входящих в состав растительных масел (г). R, R_1, R_2, R_3 - насыщенные и ненасыщенные углеводородные радикалы

В молекулах жирных кислот углеводородный радикал *R* (типа $C_{11}H_{23}$, $C_{13}H_{27}$, $C_{15}H_{31}$ и т.д.) соединен с одной карбоксильной группой COOH. Обычно в состав растительных масел входят жирные кислоты с числом атомов углерода от 12 до 24, в небольшом количестве

присутствуют и жирные кислоты с нечетным числом атомов углерода. При этом в большинстве масел (соевом, рапсовом, подсолнечном и др.) преобладают ненасыщенные жирные кислоты. В рапсовом масле обычно наиболее значительным является содержание олеиновой, линолевой или эруковой кислот. В составе оливкового масла наиболее представительными являются пальмитиновая, олеиновая и линолевая жирные кислоты.

Жирнокислотный состав растительного зависит от сорта сельскохозяйственной структуры, места произрастания, способа извлечения масла и его очистки. При этом содержание в данном растительном масле различных жирных кислот может изменяться в достаточно широких пределах. Об этом свидетельствует приведенный в таблице 1 жирнокислотный состав наиболее распространенных растительных масел [19, 30, 40-42]. Жирнокислотный состав растительных масел определяется по методике ГОСТ 30418-96 «Масла растительные. Метод определения жирнокислотного состава» (см. также ГОСТ Р 51483-99). Состав и количество жирных кислот в ОМ во многом зависят от сорта оливок, способа производства, зрелости, климата и других факторов. По жирнокислотному составу оливковое масло представляет собой смесь триглицеридов жирных кислот с высоким содержанием эфиров олеиновой кислоты (55-83%, см. таблицу 1).

Таблица 1. Жирнокислотный состав растительных масел, выраженный в процентах от общего содержания жирных кислот

Жирные кислоты	Формула состава (условная формула состава)	Растительные масла				
		Оливковое	Пальмовое	Соевое	Рапсовое	Подсол-нечное
Додекановая (лауриновая)	$C_{12}H_{24}O_2$ (C 12:0)	0	0-0,5	0-0,1	0	0-0,1
Тетрадекановая (миристиновая)	$C_{14}H_{28}O_2$ (C 14:0)	0-1,5	0,5-2,0	0-0,2	0-0,2	0-0,2
Гексадекановая (пальмитиновая)	C 16:0	6,0-12,0	39,3-47,5	8,0-13,5	1,5-6,0	5,0-7,6
Гексадеценовая (пальмитолеиновая)	C 16:1	0,1-0,8	0-0,6	0-0,6	0-3,0	0-0,3
Гептадекановая (маргариновая)	C 17:0	0-0,1	0-0,2	0-0,1	0-0,1	0-0,2
Гептадеценовая (маргаринолеиновая)	C 17:1	0	0	0-0,1	0-0,1	0-0,1
Октадекановая (стеариновая)	C 18:0	0,5-5,0	3,5-6,0	2,0-5,4	0,5-3,1	2,7-6,5

Октадеценовая (олеиновая)	С 18:1	55,0-83,0	36,0-44,0	17,0-30,0	8,0-60,0	14,0-39,4
Октадекадиеновая (линолевая)	С 18:2	3,0-21,0	9,0-12,0	48,0-59,0	11,0-23,0	48,3-74,0
Октадекатриеновая (линоленовая)	С 18:3	0-1,9	0-0,5	4,5-11,0	5,0-13,0	0-0,3
Эйкозановая (арахиновая)	С 20:0	0,1-0,6	0-1,0	0,1-0,6	0-3,0	0,1-0,5
Эйкозеновая (гадолеиновая)	С 20:1	0,1-5,0	0-0,4	0-0,5	3,0-15,0	0-0,3
Эйкозадиеновая	С 20:2	0	0	0-0,1	0-1,0	0
Докозановая (бегеновая)	С 22:0	0,1-0,8	0-0,2	0-0,7	0-2,0	0,3-1,5
Докозеновая (эруковая)	С 22:1	0,2-0,7	0	0-0,3	2,0-60,0	0-0,3
Докозадиеновая	С 22:2	0	0	0	0-2,0	0-0,3
Тетракозановая (лигноцериновая)	С 24:0	0	0	0-0,5	0-2,0	0-0,5
Тетракозеновая (нервоновая)	С 24:1	0	0	0	0-3,0	0

Примечание: в условной формуле состава (например, С 12:0) первая цифра соответствует числу атомов углерода С в молекуле, вторая - числу двойных связей.

Поскольку в составе молекул ОМ и других растительных масел присутствуют атомы углерода С и водорода Н эти масла являются горючими веществами и могут применяться в качестве моторных топлив. Низкая испаряемость и высокая вязкость растительных масел исключают их использование в бензиновых двигателях. Но они могут успешно применяться в качестве топлива для дизельных двигателей. Этому способствуют сравнительно невысокая термическая стабильность растительных масел и приемлемая температура их самовоспламенения, равная 300-320 ºС и лишь немного превышающая температуру самовоспламенения дизельных топлив, составляющую 230-280 ºС. При этом цетановое число растительных масел обычно изменяется в пределах от 35 до 38 единиц (см. таблицу 2), что сопоставимо с цетановым числом дизельных топлив, равным 45-50 единиц [19, 43-47].

Отличительной особенностью растительных масел является наличие в их составе достаточно большого количества атомов кислорода (около 10-12 % по массе). Это приводит к

некоторому снижению теплотворной способности растительных масел. Так, низшая теплота сгорания ОМ составляет 37,3 мДж/кг против 42,5 мДж/кг у дизельных топлив. Но наличие в молекулах ОМ кислорода уменьшает максимальную температуру его сгорания в дизельных двигателях и значительно улучшает их экологические показатели.

При использовании в качестве добавки фритюрного масла необходимо также учитывать изменение его химического состава в результате использования и возможное наличие канцерогенных веществ.

Изменение химического состава фритюрного масла исследовалось в работе [48], где отмечалось, что при использовании растительных масел в качестве фритюрных происходит нагревание их до температур вплоть до 200 °С.

Таблица 2. Физико-химические свойства нефтяного ДТ и растительных масел

Физико-химические свойства	Нефтяное ДТ и растительные масла					
	ДТ	Оливковое	Пальмовое	Соевое	Рапсовое	Подсолнечное
Плотность при 20 °С, кг/м^3	830	914	918	924	916	923
Вязкость кинематическая, мм^2/с:						
- при 20 °С	3,8	71,2	-	72,4	75,0	65,2
- при 40 °С	2,5	37,0	-	32,0	36,0	30,7
- при 100 °С	1,0	8,4	8,6	7,7	8,1	7,9
Поверхностное натяжение при 20 °С, мН/м	27,1	33,0	33,1	33,3	33,2	33,3
Теплота сгорания низшая, мДж/кг	42,5	37,3	37,1	37,5	37,3	37,0
Цетановое число	45	37	38	37	36	35
Температура самовоспламенения, °С	250	318	315	318	318	320
Количество воздуха, необходимое для сгорания 1 кг вещества, кг/кг	14,3	12,4	12,6	12,5	12,4	12,4
Массовое содержание.						
- С	87,0	78,0	77,6	77,5	78,0	77,6
- Н	12,6	10,0	12,0	11,5	10,0	11,5
- О	0,4	12,0	11,4	11,0	12,0	10,9
Температура застывания, °С	-35	-12	+30	-12	-20	-18
Кислотное число, мг КОН/г	0,05	5,90	0,17	0,03	4,86	2,14
Коксуемость, % (масс.)	0,200	0,198	0,120	0,438	0,465	0,505

Примечание: представлены свойства ДТ малки Л по ГОСТ 305-2013.

В этом случае масло взаимодействует с атмосферным кислородом, что приводит к

окислению жиров с образованием альдегидов, кетонов, карбонильных кислот и т.д. Накапливающиеся продукты окисления склонны к реакциям полимеризации и поликонденсации, о чем свидетельствует увеличение вязкости масла.

В качестве примера приведена реакция (условная) взаимодействия триглицида олеиновой кислоты с кислородом:

$2C_{57}H_{104}O_6+131O_2 \rightarrow 2C_5H_{10}O+2C_4H_8O+ 2C_2H_4O_2+92CO_2+82H_2O$,

где – $C_{57}H_{104}O_6$ – формула триглицида олеиновой кислоты,

$C_5H_{10}O$ – формула пентаналя (альдегид),

C_4H_8O – формула бутанона (кетон),

$C_2H_4O_2$ – формула уксусной (этановой) кислоты.

В качестве примера приведена реакция полимеризации формальдегида (CH_2O):

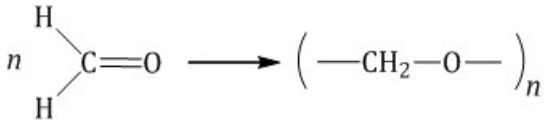


Рис 2. Реакция полимеризации формальдегида

Также из-за наличия воды в приготовляемых продуктах происходят реакции гидролиза – расщепление связей между глицерином и жирными кислотами и к формированию диглицеридов, моноглицеридов, свободных жирных кислот и глицерина.

В качестве примера приведена реакция (условная) взаимодействия триглицида олеиновой кислоты с водой:

$C_{57}H_{104}O_6+3H_2O \rightarrow C_3H_8O_3+3C_{17}H_{33}COOH$,

где – $C_3H_8O_3$– глицерин,

$C_{17}H_{33}COOH$ – олеиновая кислота.

Вопрос о увеличении вязкости фритюрного масла после использования является достаточно важным, так как это влияет на вязкость смеси дизельного топлива и фритюрного масла. В таблице 3 приведены данные по изменению вязкости фритюрного масла после использования из работы [49].

Таблица 3. Вязкость фритюрного масла до и после использования при 20 °С

Время использования, ч	Плотность, кг/м^3	Кинематическая вязкость, мм^2/с	Динамическая вязкость, Па ·с
0	925	28,0	0,0259
12	925	33,0	0,03048
36	925	55,3	0,0512

Из данных по вязкости (см. таблицу 3) следует, что вязкость масла увеличивается примерно в 2 раза после использования в течении 36 часов.

Другой отличительной особенностью растительных масел является его более тяжелый фракционный состав. Если нефтяные дизельные топлива обычно выкипают в температурном диапазоне от 180 до 340 ^{o}C, то для растительных масел это диапазон составляет от 250 до 360 ^{o}C. При этом растительные масла отличается от нефтяного ДТ заметно большей и вязкостью (см. таблицу 2). Еще большей вязкостью, как показывают данные (см. таблицу 3) обладает фритюрное масло после его использования. При температурах 20-40 ^{o}C кинематическая вязкость растительных масел на порядок превышает вязкость нефтяного ДТ. Плотность и поверхностное натяжение растительных масел также заметно выше, чем у нефтяного ДТ. Указанные отличия физических свойств растительных масел от соответствующих свойств нефтяного ДТ оказывают влияние на параметры процесса топливоподачи, и, следовательно, на показатели топливной экономичности и токсичности ОГ дизельного двигателя, работающего на растительных маслах. Кроме того, указанные различие свойств рассматриваемых топлив вызывают ряд проблем использования чистых растительных масел в качестве моторных топлив, среди которых отложение нагара и коксование деталей, образующих камеру сгорания (распылители форсунки, впускные клапаны, поршневые кольца), загрязнение смазочного масла, нестабильная работа в условиях низких температур и др. В связи с этим целесообразным представляется применение смесей нефтяного ДТ с небольшими добавками растительных масел (в частности, оливкового масла). В случае использования фритюрного масла из- за его большей вязкости количество добавляемого масла может быть еще меньше (см. таблицу 3). При этом достигается приближение свойств таких смесей к свойствам нефтяного ДТ (см. таблицу 4). При этом добавка ОМ к нефтяному ДТ может рассматриваться как экологическая добавка, улучшающая экологические показатели дизельных двигателей.

Отдельно рассмотрим вопрос наличия полиароматических (ПАУ) углеводородов в масле. Согласно ГОСТ 32511-2013 (EN 590:2009) на дизельное топливо [50] количество ПАУ в нем может достигать 8 %, однако, как показывают данные работы [51] их значение может быть значительно меньше, примерно 1 - 2%. Наличие ПАУ в отработанном фритюром масле исследовалось в работе [48]. В ней показано, что их количество в отработанном фритюрном масле незначительно (меньше 0,001 %).

Также потенциальным канцерогеном может быть акриламид (CH_2=CHC(O)NH_2 - амид акриловой кислоты), который может образовываться в продуктах, содержащих крахмал (картофель-фри, чипсы). Однако японские ученые не нашли его в отработанном фритюрном масле [52].

Таким образом, количество ПАУ в дизельном топливе будет уменьшаться пропорционально количеству добавленного масла, что улучшает экологичность двигателя.

Таблица 4. Физико-химические свойства нефтяного ДТ, ОМ и их смесей

Физико-химические свойства	Нефтяное ДТ и растительные масла				
	ДТ	ОМ	Смесь 90% ДТ и 10% ОМ	Смесь 80% ДТ и 20% ОМ	Смесь 70% ДТ и 30% ОМ
Плотность $\rho_т$ при 20 °С, кг/м^3	830	914	838	847	855
Вязкость $\nu_т$ кинематическая при 20 °С, мм^2/с	3,8	71,2	5,6	7,9	10,8
Поверхностное натяжение $\sigma_т$ при 20 °С, мН/м	27,1	33,0	27,7	28,3	28,9
Теплота сгорания низшая H_U, мДж/кг	42,5	37,3	42,0	41,5	41,0
Количество воздуха, необходимое для сгорания 1 кг вещества l_o, кг/кг	14,3	12,4	14,1	13,9	13,7
Массовое содержание. - С - Н - О	 87,0 12,6 0,4	 78,0 10,0 12,0	 86,1 12,3 1,6	 85,2 12,1 2,7	 84,3 11,8 3,9
Общее содержание серы, % (масс.)	0,200	0,002	0,180	0,160	0,141
Общее содержание ПАУ, % (масс.)	8	<0,001	7,1	6,3	5,4

Применение представленных в таблице 4 3-х разных смесей нефтяного ДТ и ОМ позволяет обеспечить требуемые показатели дизельного двигателя без изменения его настроек по максимальной подаче топлива и угла опережения впрыскивания топлива.

ОБЪЕКТ ИССЛЕДОВАНИЯ

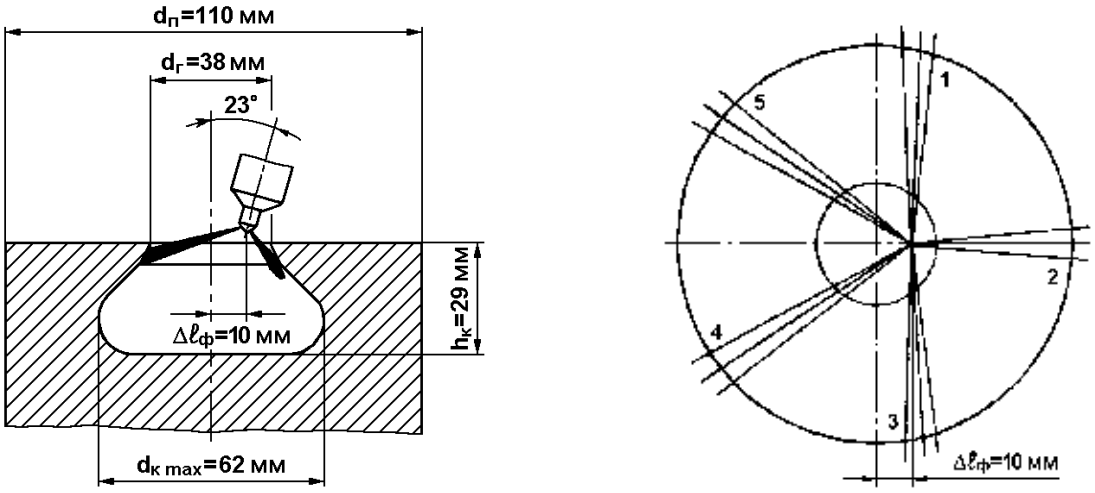
В качестве объекта исследования выбран дизельный двигатель без наддува типа Д-243 (4 Ч 11/12,5) производства ОАО «УКХ «ММЗ» (Беларусь), работающий в составе дизель-генераторной установки (ДГУ), вырабатывающей переменный электрический ток. В состав ДГУ входил также электрический генератор переменного тока типа *ECO-ECP* производства фирмы *RINA* (Италия). При испытаниях вырабатываемая электрогенератором электроэнергия потреблялась тремя тепловентиляторами, максимальная мощность каждого из которых составляет 12 кВт. Конструктивные и эксплуатационные параметры дизельного двигателя типа Д-243 приведены в таблице 5.

Таблица 5. Конструктивные и эксплуатационные параметры дизельного двигателя типа Д-243

Параметры	**Значение**
Тип двигателя	Четырехтактный, рядный, дизельный
Число цилиндров	4
Диаметр цилиндра на ход поршня	110/125
Степень сжатия ε	16,0
Общий рабочий объем iV_h, л	4,32
Тип камеры сгорания (КС), способ смесеобразования	КС типа ЦНИДИ (Центральный научно-исследовательский дизельный институт), объемно-пленочное смесеобразование
Номинальная мощность N_e, кВт	36
Номинальная частота вращения n, мин$^{-1}$	1500
Система питания	Разделенного типа
Топливный насос высокого давления (ТНВД)	Рядный типа *PP4M10U1f* фирмы *Motorpal* с центробежным регулятором
Диаметр плунжеров ТНВД $d_{пл}$, мм	10
Ход плунжеров ТНВД $h_{пл}$, мм	10
Длина нагнетательных топливопроводов $L_т$, мм	540
Форсунки	Типа ФДМ-22 производства НЗТА
Распылители форсунок	Распылителя типа 171.07.00 производства АЗПИ с пятью распыливающими отверстиями диаметром $d_р$=0,35 мм и проходным сечением $\mu_р f_р$=0,270 мм^2

Давление начала впрыскивания форсунок $p_{фо}$, МПа	21,5

Схема камеры сгорания двигателя с указанием расположения форсунки и схемой ориентации струй топлива в КС показана на рис. 2. Общий вид двигателя представлен на рис. 3, *а*, общий вид генератора типа *ECO-ECP* показан на рис. 3, *б*.



а) б)

Рис. 3. Камера сгорания дизельного двигателя типа Д-243 со указанием расположения форсунки (а) и схема ориентации струй топлива в КС (б): 1, 2, 3, 4, 5 – номера струй топлива

а) б)

Рис. 4. Общие виды дизеля типа Д-243 (а) и генератора типа ECO-ECP (б)

Исследуемый дизельный без наддува типа Д-243 имел номинальную мощность N_e=36 кВт при частоте вращения коленчатого вала n=1500 мин$^{-1}$ (см. табл. 4). Двигатель был оснащен форсунками типа ФДМ-22 производства НЗТА с распылителями типа 171.07.00 производства Алтайского завода прецизионных изделий. При испытаниях использовалось нефтяное ДТ марки «Л» по ГОСТ 305-2013 и нерафинированное оливковое масло Extra Virgine Olive Oil, производимое фирмой URZANTE (Испания), рисунок 4 [53]. Исследованы чистое дизельное топливо и его смеси с добавками 10, 20 и 30 % (по объему) оливкового масла. Некоторые

свойства исследуемых топлив приведены в таблице 4.

Рис. 5. Нерафинированное оливковое масло Extra Virgine Olive Oil, производимое фирмой URZANTESL (Испания)

Испытания двигателя проведены на режимах регуляторной характеристики при n=1500 мин$^{-1}$, формируемой регулятором, ДГУ, вырабатывающей переменный электрический ток. Этот ток удовлетворял требованиям, предъявляемым к току промышленных сетей. Исследованы режимы с нагрузками, примерно равными 0 (режим холостого хода), 33, 66 и 100% от полной (эффективная мощность двигателя N_e, соответственно, 0, 12, 24 и 36 кВт). При испытаниях статический УОВТ был равен θ = 13 градусов поворота коленчатого вала до верхней мертвой точки (ВМТ). В процессе испытаний определялись штатные параметры двигателя и показатели дымности и токсичности его отработавших газов (ОГ). Дымность ОГ измерялась с помощью дымомера «Инфракар Д1.01» ООО «Западприбор», г. Москва (рис. 5, *а*). Концентрации в ОГ токсичных компонентов (легких несгоревших углеводородов CH_x, монооксида углерода CO, оксидов азота NO_x) определялись газоанализатором «Инфракар 5М-3.01» ООО «Западприбор», г. Москва (рис. 5,*б*). Диапазоны измерений параметров двигателя и точность измерения каждого

параметра приведены в таблице 6.

а) б)

Рис. 6. Общие виды дымомера «Инфракар Д1.01» (а) и газового анализатора «Инфракар 5М-3.01» (б)

Таблица 6. Диапазоны измерений параметров дизельного двигателя Д-243 и точность их измерения

Измеряемый Параметр	**Диапазон измерения**	**Точность измерения**	**Относительная погрешность измерения**
Частота вращения	0-2500 мин$^{-1}$	±2,5 мин$^{-1}$	±0,1%
Мощность	0-40 кВт	±0,4 кВт	±1%
Расход топлива	0-12 кг/ч	±0,05 кг/ч	±0,4%
Расход воздуха	0-250 кг/ч	±2,5 кг/ч	±1%
Концентрация в ОГ:			
кислород O_2	0-21 % (об.)	±0,1% (об.) в диапазоне 0-3,3% (об.) ±3% (об.) в диапазоне 3,3-21% (об.)	±3% в диапазоне 0-3,3% (об.) ±14% в диапазоне 3,3-21% (об.)
диоксид углерода CO_2	0-16 % (об.)	±0,5 % (об.) в диапазоне 0-12,5% (об.)	±4% в диапазоне 0-12,5% (об.)
оксиды азота NO_x	0-4000 *ppm*	±100 *ppm* в диапазоне 0-1000 *ppm* ±10 *ppm*	±10% в диапазоне 0-1000 *ppm* ±0,25%

		в диапазоне 1000-4000 *ppm*	в диапазоне 1000-4000 *ppm*
монооксид углерода CO	0-5 % (об.)	±0,03 % (об.) в диапазоне 0-1% (об.)	±3% в диапазоне 0-1% (об.)
углеводороды CH_x	0-2000 *ppm*	±10 *ppm* в диапазоне 0-200 *ppm*	±5% в диапазоне 0-200 *ppm*
дымность ОГ K_X	0-100 % (Хартридж)	±1% (Хартридж) во всем диапазоне	±1% во всем диапазоне

С использованием полученных экспериментальных данных определены коэффициент избытка воздуха α, удельный эффективный расхода топлива g_e, условный (с учетом разной теплотворной способности исследуемых топлив) удельный эффективный расхода топлива g_e^*, и эффективный КПД η_e по формулам [19, 54]:

$$\alpha = \frac{G_{возд}}{l_0 G_{топл}},$$

$$g_e = \frac{G_{топл}}{N_e},$$

$$g_e^* = g_e \frac{H_{Ui}}{H_{eДД}},$$

$$\eta_e = \frac{3600}{H_U g_e},$$

где $G_{топл}$ и $G_{возд}$ – часовые расходы топлива и воздуха, кг/ч; l_o – стехиометрическое массовое отношение количества воздуха к количеству топлива (для нефтяного ДТ l_o=14,3, для смесей ДТ и ОМ – см. табл. 3); N_e – эффективная мощность двигателя, кВт; H_U – низшая теплота сгорания топлив (для нефтяного ДТ H_U=42,5, для смесей ДТ и ОМ – см. табл. 3); $H_{U\,i}$ – низшая теплота сгорания топлива *i*-того состава. Результаты испытаний и расчетов по приведенным формулам приведены в таблице 7.

Таблица 7. Параметры двигателя Д-243, работающего на различных нагрузочных режимах на различных видах топливах

Дизель Д-243, работающий на нефтяном ДТ													
№ режимов	n, мин$^{-1}$	N_e, кВт	$G_{топл}$, кг/ч	$G_{возд}$, кг/ч	α	g_e, г/(кВт·ч)	g_e^*, г/(кВт·ч)	η_e	K_X, % (Хартридж)	C_{NOx}, *ppm*	C_{CO}, %	C_{CHx}, *ppm*	C_{CO2}, %
1	1458	36,6	10,10	217	1,50	276,0	276,0	0,307	23,5	1564	0,025	12	8,97
2	1498	23,4	6,89	220	2,23	294,4	294,4	0,288	8,0	1116	0,007	11	5,94
3	1525	12,2	4,59	224	3,41	376,2	376,2	0,225	4,0	543	0,008	10	3,86
4	1572	0	2,47	231	6,54	-	-	-	3,0	160	0,009	10	2,04
Дизель Д-243, работающий на смеси 90% ДТ и 10% ОМ													
1	1452	36,6	10,15	218	1,52	277,3	274,0	0,309	18,0	1628	0,043	14	8,89
2	1496	23,2	6,90	221	2,27	297,4	293,8	0,288	5,0	1160	0,008	12	5,86
3	1521	12,1	4,60	225	3,47	380,2	375,6	0,225	2,5	557	0,010	12	3,82
4	1568	0	2,44	231	6,71	-	-	-	2,0	169	0,014	12	2,00
Дизель Д-243, работающий на смеси 80% ДТ и 20% ОМ													
1	1449	36,4	10,15	218	1,55	278,8	271,2	0,311	16,5	1677	0,054	15	8,86
2	1491	23,1	6,90	222	2,31	298,7	291,5	0,290	3,0	1175	0,015	14	5,82
3	1515	12,0	4,58	226	3,55	381,7	372,5	0,227	2,0	564	0,016	13	3,80
4	1566	0	2,44	232	6,84	-	-	-	1,5	176	0,019	13	1,98
Дизель Д-243, работающий на смеси 70% ДТ и 30% ОМ													
1	1449	36,3	10,15	220	1,58	279,6	269,8	0,314	12,5	1744	0,057	16	8,68
2	1493	23,0	6,88	223	2,37	299,1	288,6	0,293	2,0	1184	0,018	16	5,80
3	1520	12,0	4,60	227	3,60	383,3	369,9	0,229	1,5	580	0,017	15	3,78
4	1567	0	2,37	232	7,14	-	-	-	1,0	187	0,020	15	1,96

Примечание: указано объемное процентное содержание компонентов в ОГ; *ppm* – миллионные объемные доли C_{NOx} и C_{CHx}.

Полученные результаты испытаний двигателя Д-243 свидетельствуют о том, что его работа на смесях ДТ и ОМ характеризуется лучшей топливной экономичностью и пониженной дымностью ОГ (см. таблицу 7) во всем диапазоне исследованных нагрузочных режимов, что, в основном, связано с большими значениями коэффициента избытка воздуха α. Положительным фактором перевода двигателя на смеси ДТ и РМ является и меньшая концентрация в ОГ диоксида углерода (углекислого газа CO_2), что объясняется меньшим содержанием в молекулах этих топлив углерода С. Так, по данным работы [19] отношение Н/С (отношение количества атомов водорода Н и углерода С в молекулах топлив) для нефтяного ДТ составляет 1,76, а для

растительных масел - около 1,80. Но при работе на этих смесях отмечены повышенные концентрации в ОГ оксидов азота, монооксида углерода и углеводородов.

Следует отметить, что важнейшим режимом работы дизельных двигателей является режим максимальной мощности. Для определения оптимального состава смесей ДТ и ОМ использована методика, базирующаяся на расчете обобщенного критерия оптимальности в виде суммы частных критериев. При этом к наиболее значимым показателям дизельных двигателей относятся показатели топливной экономичности (в данной работе в качестве такого показателя выбран условный удельный эффективный расхода топлива g_e^*), дымность ОГ K_X, концентрации в ОГ оксидов азота C_{NOx} [19, 20, 54]. При проведении оптимизационных расчетов учитывались относительные (отнесенные к показателям двигателя, работающего на ДТ без добавления ОМ) значения указанных параметров, приведенные в таблице 8. Обобщенный критерий оптимальности $I_{обобщ}$ и обобщенный приведенный критерий $I_{обобщ\ прив}$ определялись в виде:

$$I_{обощ} = g_{e\ отн} + K_{X\ отн} + C_{NOx\ отн},$$

$$I_{обощприв} = \frac{I_{обобщ}}{3}.$$

Таблица 8. Значения параметров двигателя типа Д-243, работающего на различных видах топлива на режиме максимальной мощности

Вид топлива	Параметры дизельного двигателя на режиме максимальной мощности при n=2400 мин$^{-1}$							
	g_e^*, г/(кВт·ч)	$g_{e\ отн}^*$	K_X, %	$K_{X\ отн}$	C_{NOx}, *ppm*	$C_{NOx\ отн}$	$I_{обобщ}$	$I_{обобщ\ прив}$
100% ДТ	276,0	1,000	23,5	1,000	1564	1,000	3,000	1,000
90% ДТ и 10% ОМ	274,0	0,993	18,0	0,766	1628	1,041	2,800	0,933
80% ДТ и 20% ОМ	271,2	0,983	16,5	0,702	1677	1,072	2,757	0,919
70% ДТ и 30% ОМ	269,8	0,978	12,5	0,532	1744	1,115	2,625	0,875

При использовании описанной методики оптимальному составу смесевого топлива соответствуют минимальные значения критериев оптимальности $I_{обобщ}$ и $I_{обобщ\ прив}$. Данные таблицы 8 свидетельствуют о том, что для режима максимальной мощности оптимальным является топливо, содержащее 70% ДТ и 30% ОМ, при использовании которого значение обобщенного приведенного критерия оптимальности принимает минимальное значение $I_{обобщ\ прив}$=0,875. При таком составе топлива достигаются минимумы условного удельного

эффективного расхода топлива g_e^*=269,8 г/(кВт·ч) и дымности ОГ K_X=12,5 % по шкале Хартриджа, но отмечена максимальная концентрация в ОГ оксидов азота C_{NOx}=1744 *ppm*. Следует отметить, что при переводе двигателя с нефтяного ДТ на смесь 70% ДТ и 30% ОМ условный удельный эффективный расход топлива g_e^* снизился на 2,2%, дымность ОГ K_X уменьшилась почти вдвое, концентрация оксидов азота в ОГ возросла на 11, 5%.

В целом, проведенное исследование подтвердило возможность и эффективность использования смесей нефтяного ДТ и оливкового масла в качестве моторного топлива для дизельных двигателей. Применение этих смесей позволило улучшить показатели топливной экономичности и токсичность отработавших газов исследованного двигателя.

ЗАКЛЮЧЕНИЕ

На современном этап развития двигателестроения проводятся многочисленные исследования двигателей внутреннего сгорания, работающих на различных альтернативных топливах. К перспективным альтернативным топливам для дизельных двигателей относятся растительные масла и топлива, получаемые на их основе. Особый интерес вызывает возможность использования в качестве моторных топлив низкосортных и отработанных фритюрных растительных масел. К основным фритюрным маслам относится оливковое масло.

В работе проведены исследования дизельного двигателя типа Д-243, работающего в составе дизель-генераторной установки производства Минского моторного завода. Двигатель прошел испытания на режимах регуляторной характеристики, формируемой при частоте вращения n=1500 мин$^{-1}$ с нагрузками, равными 0 (режим холостого хода), 33, 66 и 100% от полной (0, 12, 24 и 36 кВт). Исследованы чистое дизельное топливо и его смеси с добавками 10, 20 и 30 % (по объему) оливкового масла.

Результаты испытаний двигателя Д-243 свидетельствуют о том, что его работа на смесях дизельного топлива и оливкового масла характеризуется лучшей топливной экономичностью, пониженной дымностью ОГ и меньшими концентрациями в них углекислого газа во всем диапазоне исследованных нагрузочных режимов. Но при работе на этих смесях отмечены повышенные концентрации в ОГ оксидов азота, монооксида углерода и углеводородов.

Разработана методика оптимизации исследованных смесевых топлив, базирующаяся на расчете обобщенного критерия оптимальности в виде суммы частных критериев. При этом в качестве наиболее значимых частных критериев выбраны условный удельный эффективный расхода топлива, учитывающий различную теплотворную способность исследованных смесей, дымность отработавших газов и концентрация в них оксидов азота. При этом учитывались относительные (отнесенные к показателям двигателя, работающего на нефтяном дизельном топливе) значения указанных параметров.

Оптимизационные расчеты с использованием разработанной методики, проведенные для режима максимальной мощности, показали, что оптимальным является топливо, содержащее 70% ДТ и 30% ОМ. При его использовании значение обобщенного приведенного критерия оптимальности принимает минимальное значение, равное 0,875. При таком составе топлива достигаются минимумы условного удельного эффективного расхода топлива - 269,8 г/(кВт·ч) и дымности ОГ - 12,5 % по шкале Хартриджа, но отмечена максимальная концентрация в ОГ оксидов азота - 1744 *ppm*. При переводе двигателя с нефтяного ДТ на смесь 70% ДТ и 30% ОМ условный удельный эффективный расход топлива снизился на 2,2%, дымность ОГ уменьшилась почти вдвое, концентрация оксидов азота в ОГ возросла на 11, 5%.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Егорычев В.С. Топлива химических ракетных двигателей: учебное пособие [Текст] / В. С. Егорычев — Самара: СГАУ, 2007 — 71 с.
2. Артемьев М., Ашуха. А Учебник по машинному обучению / Артемьев М., Ашуха. А [Электронный ресурс] // Яндекс.Образование : [сайт]. — URL: https://education.yandex.ru/handbook/ml (дата обращения: 14.12.2024).
3. Алемасов В.Е., Дрегалин А.Ф., Тишин А.П. Теория ракетных двигателей [Текст] / В.Е.Алемасов, А.Ф.Дрегалин, А.П.Тишин — 4-е изд. — Москва: Машиностроение, 1989 — 462 с.\
4. Добровольский М. Ф. Жидкостные ракетные двигатели : основы проектирования [Текст] / М. Ф. Добровольский — 4-е изд. — Москва: Машиностроение, 1989 — 462 с.
5. Штехер М. С. Топлива и рабочие тела ракетных двигателей [Текст] / М. С. Штехер — Москва: Машиностроение, 1976 — 302 с.
6. Паушкин Я. М. Ракетные топлива [Текст] / Паушкин Я. М. — Москва: Мир, 1975 — 188 с.
7. Мелькумов, Т. М. Ракетные двигатели [Текст] / Т. М. Мелькумов — Москва: Машиностроение, 1968 — 511 с.
8. Васильев, А. П. Основы теории и расчёта жидкостных ракетных двигателей [Текст] / А. П. Васильев — 3-е изд. — Москва: Высш. шк., 1983 — 703 с.
9. RDKit: Open-Source Cheminformatics Software / [Электронный ресурс] // https://rdkit.org/ : [сайт]. — URL: www.rdkit.org/docs/source/rdkit.Chem.rdMolDescriptors.html (дата обращения: 20.12.2024).
10. scikit-learn Machine Learning in Python / [Электронный ресурс] // scikit-learn : [сайт]. — URL: https://scikit-learn.org/stable/ (дата обращения: 20.12.2024).
11. Kegl B., Kelg M., Pehan S. Green Diesel Engines Biodiesel Usage in Diesel Engines. London: Springer-Verlag, 2013, 265 p.
12. Che Mat S., Idroas M.Y., Hamid M.F., Zainal Z.A. Performance and Emissions of Straight Vegetable Oils and its Blends as a Fuel in Diesel Engine: A Review // Renewable and Sustainable Energy Reviews, 2018, vol. 82, pp. 808–823. DOI: https://doi.org/10.1016/j.rser.2017.09.080.
13. Atmanli A., Ileri E., Yuksel B., Yilmaz N. Extensive Analyses of Diesel–Vegetable Oil-n-Butanol Ternary Blends in a Diesel Engine. Applied Energy, 2015, vol. 145, pp. 155–162. DOI: https://doi.org/10.1016/j.apenergy.2015.01.071.
14. Anantha Raman L., Deepanraj B., Rajakumar S., Sivasubramanian V. Experimental Investigation on Performance, Combustion and Emission Analysis of a Direct Injection Diesel Engine Fuelled with Rapeseed Oil Biodiesel. Fuel, 2019, vol. 246, pp. 69–74. DOI: https://doi.org/10.1016/j.fuel.2019.02.106.

15. Марков В.А., Девянин С.Н., Са Бовэнь, Нормуродов А.А. Исследование работы дизельного двигателя на смесевых и эмульгированных биотопливах с добавками рапсового масла // Двигателестроение. 2023. № 1. С. 70-90.
16. Девянин С.Н., Марков В.А., Неверов В.А., Болдырев А.В. Исследование процессов впрыскивания и распыливания топлива и параметров дизельного двигателя, работающего на биотопливах // Двигателестроение. 2025. № 3. С. 82-101.
17. Марков В.А., Девянин С.Н., Крылов В.И., Багров В.В. Перспективы использования биотоплива в дизельных двигателях // Транспорт на альтернативном топливе. 2012. № 6. С. 6-10.
18. Марков В.А., Багров В.В., Девянин С.Н. Необходимость и перспективы использования моторных топлив растительного происхождения // Грузовик. 2012. № 7. С. 38-47.
19. Марков В.А., Девянин С.Н., Семенов В.Г., Багров В.В., Зыков С.А. Моторные топлива, производимые из растительных масел. Рига, Lambert Academic Publishing, 2019, 420 с.
20. Марков В.А., Баширов Р.М., Габитов И.И. Токсичность отработавших газов дизелей. М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. 376 с.
21. Российский и мировой рынки масличных культур по состоянию 23.12.2025г. URL:https://www.kaicc.ru/sites/default/files/obzor_ maslichnye_rf_i_mir_23.12.2025.pdf.(дата обращения 14.01.2026).
22. Русагро: профессиональные фритюрные масла. URL: https://www.solpro.ru/wp- content/uploads/2024/07/booklet_solpro_vom_2024_web.pdf?ysclid=miblp9fmhy697294513 (дата обращения 14.01.2026).
23. Sanli H. An Experimental Investigation on the Usage of Waste Frying Oil-Diesel Fuel Blends with Low Viscosity in a Common Rail DI-Diesel Engine. Fuel, 2018, vol. 222, pp. 434–443. DOI: https://doi.org/10.1016/j.fuel.2018.02.194.
24. Morimune T., Yamaguchi H., Konishi K. Exhaust Emissions and Performance of Diesel Engine Operating on Waste Food-Oil. Transactions of the JSME, Ser. B, 2000, vol. 66, № 641, pp. 294–299. DOI: https://doi.org/10.1299/kikaib.66.294.
25. Dey P., Ray S. Optimization of Waste Vegetable Oil-Diesel Blends for Engine Performance: A Response Surface Approach. Arabian Journal for Science Engineering, 2020, vol. 45, pp. 7725–7739. DOI: https://doi.org/10.1007/s13369-020-04843-6.
26. Hamasaki K., Tajima H., Takasaki K. et al. Utilization of Waste Vegetable Oil Methyl Ester for Diesel Fuel // SAE Technical Paper Series. 2001. № 2001-01- 2021. P. 1-6.

27. России нужно свое биотопливо. Но почти все сырье для него уходит за рубеж. URL: https://mashnews.ru/rossii-nuzhno-svoe-biotoplivo.-no-syire-dlya-nego-uxodit-za-rubezh.html (Дата обращения 23.11.2025).
28. Blin J., Brunschwig C., Chapuis A., Changotade O., Sidibe S., Noumi E., Girard F. Characteristics of vegetable oils for use as fuel in stationary diesel engines - towards spesifications for a standard in west Africa // Renewable and Sustainable Energy Reviews. 2013. Vol. 22. P. 580-597. https://hal.archives-ouvertes.fr/hal-00814899.
29. Goettler H.J., Harwood R.F., Ziejewski M., Klosterman H.J. On the Thermal Decomposition and Residue Formation of Plant Oils // SAE Technical Paper Series. 1986. № 861582. P. 1-9. DOI: https://doi.org/10.4271/861582.
30. Кодекс Алиментариус (лат. «Продовольственный кодекс»). Жиры, масла и производные продукты: Пер. с англ. М.: Изд-во «Весь мир», 2007. 68 с.
31. Бурункова Ю.Э., Успенская М.В., Самуйлова Е.О. Растительные масла: свойства, технологии получения и хранения, окислительная стабильность. СПб: Университет ИТМО, 2020. 82 с.
32. Aksoy H.A., Kahraman I., Karaosmanoglu F., Civelekoglu H. Evaluation of Turkish Sulphur Olive Oil as an Alternative Diesel Fuel // Journalof the American Oil Chemists' Society (JAOCS). - 1988. - Vol. 65. - № 6. - P. 936-938. Doi:10.1007/BF02544514.
33. Che Mat S., Idroas M.Y. , Hamid M.F., Zainal Z.A. Performance and emissions of straight vegetable oils and its blends as a fuel in diesel engine: A review // Renewable and Sustainable Energy Reviews. 2018. Vol. 82. P. 808–823. Doi:10.1016/j.rser.2017.09.080.
34. Karonis D., Anastopoulos G., Lois E., Stournas S., Zannikos F.,Serdari A. Assessment of the Lubricity of Greek Road Diesel and the Effect of the Addition of Specific Types of Biodiesel // SAE Technical Paper Series. 1999. № 1999-01-1471. P. 1-6. Doi:10.4271/1999-01-1471.
35. Misra R.D., Murthy M.S. Straight vegetable oils usage in a compression ignition engine: A review // Renewable and Sustainable Energy Reviews. 2010, vol. 14, pp. 3005–3013. doi:10.1016/j.rser.2010.06.010.
36. Rakopoulos D.C, Rakopoulos C.D., Giakoumis E.G., Dimaratos A.M., Founti M.A. Comparative environmental behavior of bus engine operating on blends of diesel fuel with four straight vegetable oils of Greek origin: Sunflower, cottonseed, corn and olive // Fuel, 2011, vol. 90, issue 11, pp. 3439–3446. DOI: 10.1016/j.fuel.2011.06.009.
37. Sendari A., Fragioudakis K., Kalligeros S., Stournas S., Lois E. Impact of Using Biodiesels of Different Origin and Additives on the Performance of a Stationary Diesel Engine // Transactions of the ASME. Journal of Engineering for Gas Turbines and Power. 2000. Vol. 122. No 4. P. 624-631. Doi:10.1115/1.1289385.

38. Vrabie V., Scarpete D., Zbarcea O. Vegetable oils as alternative fuel for new generation of diesel engines. A review // Scientific proceedings XXIV international scientific-technical conference «trans & MOTAUTO ’16». 2016. Vol. 1. P. 105-109.
39. Пенкин А.Л., Смирнов В.С., Воробьев А.А., Орлов С.В., Опарина Е.В. Применение биодизельного топлива в условиях холодного климата // Грузовик. 2024. № 1. С. 25-29.
40. Ильина Г.Н., Ламоткин С.А., Колногоров К.П., Скаковский Е.Д. идентификация состава растительных масел хроматографическими и спектральными методами // Труды БГТУ. 2014. № 4. С. 207-210.
41. Кудряков Д.Н., Фролов Д.И. Сравнение жирно-кислотного состава различных видов пищевых масел // Инновационная техника и технология. 2023. Том 10. № 4. С. 11-16.
42. Паштецкая А.В., Бакова Н.Н., Палий А.Е., Карпова А.Н. Органолептические и физико-химические характеристики оливкового масла, полученного в условиях южного берега Крыма // Plant Biolody and Horticulture: Theory, innovation. 2020. №4 p. 83-93.
43. Кулиев Р.Ш., Ширинов Ф.Р., Кулиев Ф.А. Физико-химические свойства некоторых растительных масел // Химия и технология топлив и масел. 1999. № 4. С. 36-37.
44. Стрельцов В.В., Бугаев А.М. Перспективы использования в технике масел растительного происхождения // Вестник ФГОУ ВПО МГАУ. 2010. № 2. С. 47-49.
45. Марков В.А., Са Бовэнь, Неверов В.А., Девянин С.Н., Камалтдинов В.Г. Показатели токсичности отработавших газов дизельного двигателя, работающего на нефтяном дизельном топливе с добавками растительных масел // Двигателестроение. 2020. № 4. С. 18-24.
46. Марков В.А., Кулешов А.С., Неверов В.А., Са Бовэнь, Зенкин А.Н. Совершенствование процессов распыливания топлива и смесеобразования при работе дизеля на смесевых биотопливах // Двигателестроение. 2021. № 1. С. 3-12.
47. Марков В.А., Са Бовэнь, Неверов В.А., Камалтдинов В.Г. Исследование воспламеняемости растительных масел в камере сгорания дизельного двигателя // Двигателестроение. 2021. № 3. С. 16-21.
48. Моргунова, Е. М. Исследование образования потенциально опасных соединений при приготовлении продукции во фритюре / Е. М. Моргунова, А. В. Пчельникова, В. Н. Бабодей // Пища. Экология. Качество: труды XVII Международной научно-практической конференции, Новосибирск, 18–19 ноября 2020 года. – Екатеринбург: Уральский государственный экономический университет, 2020. – С. 418-422.

49. Максимец В.П., Федак Н.В., Кравченко Э.Ф., Баранов Д.В. Изменение масла при нагреве // Известия высших учебных заведений. Пищевая технология. 1993. № 1-2 (212-213). С. 52-54.
50. ГОСТ 32511-2013 (EN 590:2009). Топливо дизельное Евро. Технические условия. М: Стандартинформ, 2014. - 20 с.
51. Трофимова Т.П., Башкирова Т.В. Влияние технологических факторов на изменение жиров пищевых продуктов при кулинарной обработке / В сборнике: Товароведение, технология и экспертиза: инновационные решения и перспективы развития. Материалы национальной научно-практической конференции. Москва, 2020. С. 177-181.
52. Totani, Nagao & Yawata, Miho & Takada, Mitsunobu & Moriya, Masafumi. (2007). Acrylamide Content of Commercial Frying Oil // Journal of Oleo Science. 56. 103-106.
53. Масло оливковое URZANTE. URL: https://www.onlinetrade.ru/catalogue/maslo_rastitelnoe- c3573/urzante/maslo_olivkovoe_urzante_100_1_1_8423243016232- 3806518.html?utm_source=ydirect&utm_medium=cpc&YDTK&etext=&yclid=5625411605585985535&ybaip=1 (дата обращения 14.01.2026).
54. Машиностроение. Энциклопедия. Том IV. Двигатели внутреннего сгорания / Л.В. Грехов, Н.А. Иващенко, В.А. Марков и др. Под ред. А.А. Александрова, Н.А. Иващенко. М.: Машиностроение, 2013. 784 с.

СОЗДАНИЕ И ИССЛЕДОВАНИЕ БИОРАЗЛАГАЕМОГО ДРЕВЕСНО-КРАХМАЛЬНОГО КОМПОЗИТА ДЛЯ УПАКОВКИ

Филимонова Василиса Владимировна

Лобня г., МБОУ Лицей, 8 класс

Секция: Химические технологии

Работа выполнена без научного руководителя

АННОТАЦИЯ

Целью данной работы являлось создание и комплексное исследование биоразлагаемого древесно-крахмального композита (ДКК) для применения в качестве экологичной упаковки. В работе использовались методы испытания механических свойств и водопоглощения, а также методы оценки биоразлагаемости. В результате был разработан ряд рецептур на основе картофельного крахмала, древесной муки и глицерина, среди которых выделен оптимальный состав, демонстрирующий удовлетворительные прочностные характеристики и контролируемую скорость деградации. Установлено, что введение древесного наполнителя повышает жесткость композита, а тип и количество пластификатора напрямую влияют на его эластичность и устойчивость к влаге. Проведенные исследования подтвердили перспективность использования полученного материала для производства экологически безопасной одноразовой продукции, что позволяет сделать вывод о решении актуальной проблемы пластикового загрязнения.

ВВЕДЕНИЕ

Актуальность заключается в том, что пластиковое загрязнение— глобальная проблема современного мира. Биоразлагаемые материалы на основе возобновляемого сырья— это экологичная альтернатива для одноразовой упаковки, сельскохозяйственных изделий (горшки, этикетки).

Целью моего проекта является создание биоразлагаемого композита, с использованием крахмала, древесного наполнителя, пластификатора и различных добавок, изучение физических свойств композита

Для достижения цели необходимо выполнить задачи:

- Изучение проблематики проекта;
- Составление рецептур;
- Изучение и подготовка сырья;
- Изготовление базового композита;
- Изготовление различных версий композита;
- Комплексное изучение свойств всех композитов;
- Исследование биоразлагаемости;
- Изготовление прототипа;
- Результаты и выводы.

ИЗУЧЕНИЕ ПРОБЛЕМАТИКИ ПРОЕКТА

На данный момент пластиковые загрязнение мира составляет 12% всех твердых бытовых отходов (130 млн тонн ежегодно), при этом только 9% пластика перерабатывается. Исследование фиксирует рост производства пластмассы на 52% к 2040 году и подчёркивает необходимость глобальных согласованных мер. Все мы знаем, что привычный нам пластик производится из нефти, газа или угля. Но все это относится к невозобновляемым ресурсам. Первые полимерные биоразлагаемые материалы получилось создавать еще в начале прошлого века из-за нефтяного кризиса и экологических проблем. Первым биопластиком считается галалит (1897 г.), а научный прорыв совершил Морис Лемуан, выделив полугидроксибутират (ПГБ) в 1926 году. Современное усовершенствование биопластика обусловленно критической экологической ситуацией и запретами на пластик в ряде стран. Поэтому сейчас важно начинать использовать биоразлагаемые композиты наравне с пластиком из невозобновляемых ресурсов. Переход на биоразлагаемые материалы позволить значительно улучшать экологическую обстановку в мире.

СОСТАВЛЕНИЕ РЕЦЕПТУР

Основа моего биопластика – крахмал. Так как крахмал – природный полимер (полисахарид), при нагревании с водой (клейстеризации) он образует вязкий гель, который при высыхании или охлаждении затвердевает, склеивая составные компоненты.

Далее следует древесная мука. Чистый крахмал получается слишком липким, текучим и нестабильным при контакте с водой. Древесные частицы повышают жесткость, снижают усадку при сушке и удешевляет материал.

Пластификатор также является важной частью базовой рецептуры. Чистый крахмал хрупкий, чтобы композит стал гибким, как пластик, нужен пластификатор. В моей работе я буду использовать глицерин.

Добавки – желательная, но не обязательная часть композита. Добавки могут выполнять как эстетическую функцию (цвет), так и практическую (помогают устранить какие-либо несовершенства).

Эстетические природные добавки: свекла, куркума, шпинат, матча, бамбуковый уголь. Для фактурности можно добавить: песок, различные высушенные крупы (гречка, овес), чай, скорлупа, крупные опилки, специи.

Эстетические искусственные добавки: различные синтетические химические вещества для придания цвета (тартразин, кармуазин, индигокармин и т.д.), различные фактурные добавки (блестки, гранулы, слюда и т.д)

Практические добавки: воск (для гидрофобности), натуральные масла (создают водоотталкивающий барьер), лигнин (в качестве стабилизатора композита).

Для создания базовой рецептуры нам необходимо использовать:

- Крахмал в количестве 60-70%. Это основа. Меньше 50% – материал может стать более несвязным, больше 80% – материал будет слишком хрупким и дорогим.
- Пластификатор в количестве 20-30%. Менее 15% – материал будет хрупким, более 35% – материал станет липким.
- Наполнитель в количестве 10-20%. Меньше 10% – материал теряет жесткость, практически чистый крахмал, более 30% – дерево будет впитывать большое количество воды из-за чего придется использовать большее количество воды, что увеличивает срок сушки.

Из этого мы получаем базовую рецептуру:

Крахмал (60-70%), пластификатор (20-30%), наполнитель (10-20%)

ИЗУЧЕНИЕ И ПОДГОТОВКА СЫРЬЯ

В моем проекте я буду использовать кукурузный крахмал, так как он лучше в плане термоустойчивости, текстуры и густоты, в отличие от картофельного. Картофельный крахмал может сделать смесь более плотной и менее гибкой, теряет свои загущающие свойства при длительном кипении.

В качестве пластификатора я буду использовать глицерин, он является более доступным, нетоксичный, безопасен для окружающей среды и полностью разлагается.

Наполнитель – древесная мука крупной и мелкой фракции для создания и изучения различных вариаций композита. Древесина полностью разлагается в почве, является одним из самых экологичных видов наполнителей.

Добавки я буду использовать только практичные, а именно пчелиный воск для гидрофобности.

ИЗГОТОВЛЕНИЕ БАЗОВОГО КОМПОЗИТА

Для того чтобы крахмал показал свои связующие способности необходимо добавление воды. В среднем необходимо в 5 раз больше воды, чем крахмала. Так как какое-то количество воды впитывает древесина (в среднем от 100% до 300% и более от сухого вещества).

1. Первым шагом является – смешивание всех сухих ингредиентов, добавлением глицерина, а затем и воды.
2. Вторым шагом необходимо нагреть массу до загустения крахмала, а затем активно перемешиваем до однородной смеси.
3. Выкладываем массу в подходящую форму или же ждем остывания и вылепливаем желаемую форму.

Композит можно оставить на несколько дней в теплом помещении с проветриванием или уменьшить срок сушки с помощью дополнительных устройств (дома это могут быть духовка или аэрогриль). В среднем срок сушки около 72 часов.

ИЗГОТОВЛЕНИЕ РАЗЛИЧНЫХ ВИДОВ КОМПОЗИТА

Для получения идеальной рецептуры надо пройти через множество проб и ошибок. Можно проверить положительные действия практических добавок с помощью создания композита с ними. Метод изготовления композита остается такой же, но с различным соотношением ингредиентов.

1. Создание композита с использованием древесной муки крупной фракции. (Далее номер 1)
2. Создание композита с использованием пчелиного воска. (Далее номер 2)
3. Создание композита, где 50% – пластификатор (глицерин). (Далее номер 3)

4. Создание композите, где 50% – древесная мука мелкой фракции. (Далее номер 4)
5. Создание композита, где все ингредиенты относятся 1:1:1. (Далее номер 5)

КОМПЛЕКСНОЕ ИЗУЧЕНИЕ СВОЙСТВ ВСЕХ КОМПОЗИТОВ

Композит номер 1 получился самым жестким, имел самый короткий срок сушки, является хрупким, потому что представляет собой самый неоднородный материал, но из него легче всего собрать фигуру, имеет слабый запах дерева, крупные частицы дольше разлагаются, но они создают внутренние пустоты, это ускоряет проникновение воды и гифов грибов (в окружающей среде)

Композит номер 2 не размокает при длительном контакте с водой, но при этом срок разложения сильно увеличивается из-за того, что воск создает гидрофобный слой. Композит номер 2 первый начал уменьшаться в объеме, но сушился около 72 часов. Не такой хрупкий, как номер 1 из-за однородной консистенции (была использована мука мелкой фракции), присутствует слабый запах меда.

Композит номер 3 получился самым мягким, но из-за маленького количества крахмала получилось много трещин, имел долгий срок сушки ~85 часов, имеет запах глицерина, имеет самую быструю скорость разложения, так как высокая глицерина делает материал чрезвычайно гидрофильным, способствует быстрому вымыванию пластификатора и облегчает доступ микроорганизмов к крахмалу, что резко ускоряет биоразложение.

Композит номер 4 имеет самый долгий срок высыхания ~100 часов, так как на фазе смешивания ингредиентов потребовал ~60 грамм воды на 4 грамма крахмала, 8 грамм древесной муки мелкой фракции, 4 грамма глицерина. В разложении показал себя плохо из-за того, что воду впитывает равномерно, а не создает отдельные «каналы», которые положительно влияют на биоразложение, но при этом номер 4 долго сохраняет форму. После касания на руках остаются мелкие частицы древесной муки. Имеет отчетливый запах дерева.

Композит номер 5 больше всех уменьшился в размере, имел срок сушки ~60 часов, быстро разлагается, так как имеет сбалансированную систему (глицерин увеличивает скорость разложения, древесной муки не так много), своего рода «золотая середина» (нет сильных замедлителей-воска, нет экстремальных ускорителей-50% глицерина), присутствует очень слабый запах дерева, не является хрупким.

Таблица 1. Сравнение композитов

Номер	Разложение	Сушка	Хрупкость
1	2	1	4
2	5	3	3
3	1	4	1

4	4	5	5
5	3	2	2

В таблице композиты расставлены по местам (разложение, сушка – от самого быстрого, до самого медленного; в хрупкости от самого не хрупкого, до самого хрупкого)

ИЗГОТОВЛЕНИЕ ПРОТОТИПА

Для практической части моего проекта я создам упаковку-горшочек для рассады. Часто в магазинах рассаду продают в пластиковых горшочках, которые необходимо снять или срезать, но я хочу сделать горшочек, чтобы можно было сажать его с растением вместе. Для создания упаковки я выбрала композит номер 1, так как он имеет самую быструю скорость сушки, имеет быструю скорость разложения. Из минусов только хрупкость, но если же выберем самые не хрупкие, то там будет страдать либо скорость сушки, либо разложения. Метод приготовления был базовым, для создания горшочка было использовано:

1. Кукурузный крахмал—30 г
2. Глицерин—10 г
3. Древесная мука крупной фракции—10 г
4. Вода—150 г

Полная сушка заняла ~90 часов из-за толстых стенок.

ЗАКЛЮЧЕНИЕ

В ходе работы было выявлено несколько основных аспектов. Оптимальное количество компонентов это 60-70% крахмала, 20-30% глицерина, 10-20% древесной муки. Древесный наполнитель не только удешевляет материал, но и влияет на скорость биоразложения, водопоглощение. Введение воска отрицательно влияет на скорость биоразложения, но при этом является практической добавкой и если делать из композита с воском что-то другое, то качества воска могут хорошо повлиять (материал не размокнет).

В ходе работы были успешно созданы пять вариантов биоразлагаемого древесно-крахмального композита с различными наполнителями и соотношениями компонентов. Экспериментально подтверждено, что варьирование состава позволяет в широких пределах управлять ключевыми свойствами материала: от сверхбыстрого разложения (образец с 50% глицерина) до повышенной водостойкости (образец с пчелиным воском). Установлено, что оптимальным балансом между технологичностью, прочностью и прогнозируемой скоростью деградации обладает композит, изготовленный по базовой рецептуре. Полученные материалы могут быть рекомендованы для создания экологичной упаковки и агротехнических изделий с заданным сроком службы. В работе также был создан горшочек для рассады на продажу, что поможет уменьшить количество потребляемого пластика

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. https://nia.eco/2025/12/04/109940/
2. http://vernadsky.ru/de/proekty/zelvesna
3. http://biodat.ru/
4. https://www.ecosociety.ru/
5. http://ecoindustry.ru/
6. https://www.bio.tampomechanika.ru/stati/chto-takoe-biorazlagaemye-materialy/
7. https://leworld.org/tpost/3sl3rp2oc1-bioplastik-chto-eto-takoe-i-na-chto-obra
8. https://elementy.ru/nauchno-populyarnaya_biblioteka/431802/Plastiki_biologicheskogo_proiskhozhdeniya_Работа А. Лешина Пластики биологического происхождения, «Химия и жизнь» №9
9. https://clck.ru/3S9AoN
10. https://clck.ru/3S9BQB

АРХИБОТ MARKUS – АВТОНОМНЫЙ ГУСЕНЕЧНЫЙ РОБОТ ДЛЯ ИССЛЕДОВАНИЯ И 3D-КАРТОГРАФИИ ПЕЩЕР

Герасименко Георгий Андреевич

Омск г, БОУ «Лицей БИТ», 10 класс

Секция: Робототехнические системы

Научный руководитель: Сидоренко Елена Геннадьевна, учитель математики БОУ «Лицей БИТ»

АННОТАЦИЯ

Архибот MARKUS – это автономный гусеничный робот, предназначенный для исследования труднодоступных подземных пространств и создания точных цифровых двойников пещер. Робот может работать как автономно, так и под управлением оператора.

Мой робот оснащён лазерным лидаром 360° на гиростабилизированной платформе и 3-х осевым манипулятором. Манипулятор позволяет роботу извлекать образцы из пещер, а также самостоятельно преодолевать сложные участки. На захвате установлена FPV-камера, видеопоток с которой в режиме реального времени передаётся оператору для мониторинга и дистанционного управления. Этот видеопоток также записывается для обучения нейросетей, которые позже будут помогать роботу ориентироваться в похожих условиях. Автономная навигация и распознавания будут запускаться на микроПК Raspberry Pi, который будет установлен на роботе.

Для построения цифрового двойника используются лидар 360° и акселерометр. Автономный захват объектов обеспечивает лазерный дальномер, установленный на манипуляторе. Все датчики подключены к микроконтроллеру ESP32 и сейчас передают информацию на компьютер по беспроводному каналу Wi-Fi.

Примеры применения робота:

- Автономное исследование и 3D-картография пещер и других подземных пространств.
- Проведение археологических и геологических исследований.
- Использование в спасательных операциях для разведки завалов и опасных зон.
- Инспекция трубопроводов

Планируется внедрение автономной навигации, создание удобного ПО для работы с цифровыми двойниками, установка системы стереозрения и датчиков для анализа состояния среды в пещерах.

Архибот MARKUS — шаг вперёд в области автономной робототехники для изучения скрытых подземных миров.

Рис. 1. Архибот MARKUS

Ключевые слова: Картографическая робототехника, робот, 3Д моделирование, проектирование, FDM печать, ЧПУ фрезеровка, программирование на ESP32/Python, машинное зрение.

ВВЕДЕНИЕ

ОБОСНОВАНИЕ АКТУАЛЬНОСТИ

Исследование пещер и подземных пространств традиционно сопряжено с высокими рисками для человека: сложная навигация, опасная среда, ограниченная видимость. Современные технологии позволяют частично автоматизировать этот процесс с помощью роботизированных систем. Однако существующие решения либо слишком дороги, либо не обладают необходимой автономностью и адаптивностью для работы в условиях полного отсутствия освещения и сложного рельефа.

Целью данной работы является разработка и создание автономного гусеничного робота «Архибот MARKUS», способного выполнять 3D-сканирование, картографию и забор образцов в пещерах. Задачи проекта: проектирование и изготовление гусеничной платформы, разработка системы навигации на основе лидара и стереозрения, создание манипулятора с автономным захватом, интеграция систем управления и тестирование прототипа.

Актуальность проекта обусловлена растущим интересом к цифровизации природных объектов, необходимостью безопасного проведения археологических и геологических изысканий, а также потенциальным применением в спасательных операциях.

АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Проведённый анализ рынка и консультация со специалистом-спелеологом показали, что большинство роботов для подземных исследований — это либо дорогие промышленные решения (например, Boston Dynamics Spot с дополнительным оборудованием), либо малопроходимые колесные платформы. Специализированных, доступных по стоимости гусеничных роботов, объединяющих картографию, манипуляции и дистанционное присутствие для работы в пещерах, на рынке практически нет. См. Табл. 1. Это подтверждает новизну и практическую востребованность проекта «Архибот MARKUS»

Таблица 1. Аналоги

Параметр / Робот	«Архибот MARKUS»	Boston Dynamics Spot	Подводный робот UX-1Neo	CoSTAR / LUVMI-X (исслед. миссия)
Тип платформы	Гусеничная, наземная	Четвероногий (робопес)	Сферический, подводный	Наземный, колесный/гусеничный (в команде)
Основное назначение	Исследование и 3D-картография пещер, забор образцов.	Универсальные инспекционные и спасательные задачи.	Картография затопленных пещер и шахт.	Автономное исследование лавовых пещер для лунных/марсианских миссий.
Манипулятор	Есть. 3-осевой, для захвата образцов.	Есть (опция Spot Arm). 6 степеней свободы, грузоподъемность до 11 кг (подъем), до 25 кг (волочение).	Нет. Только камеры и сонар.	Вероятно, есть / специализированный. Включает сенсорный куб и системы спуска.
Система картографии	Есть. Лидар 360°, гиростабилизированная платформа.	Есть. Стереокамеры 360° для навигации.	Есть. Сонар, оптические и гиперспектральные камеры.	Есть. Основная задача — автономное построение 3D-карт.
Вес	8 кг.	32 кг.	~100 кг.	Данных нет (команда из нескольких роботов).
Время автономной работы	До 3 часов.	До 1.5 часов.	До 8 часов.	Данных нет (предположительно, несколько часов для миссии).
Примерная цена / доступность	Низкая. Себестоимость прототипа ~45 тыс. руб.	Очень высокая. Базовая версия от ~5.5 млн руб., коммерческая версия с манипулятором дороже.	Очень высокая. Специализированное научное оборудование.	Чрезвычайно высокая. Научно-космический прототип, не для коммерческой продажи.
Страна / Производитель	Россия. Персональный проект.	США. Boston Dynamics.	Международный консорциум (Португалия, Чехия, Венгрия и др.).	Европейский консорциум (DFKI, EKA и др.).

ДОРОЖНАЯ КАРТА

Я проанализировал существующие решения для исследования пещер и выработал следующую дорожную карту по разработке автономного гусеничного робота «Архибот MARKUS»:

Таблица 2. Дорожная карта

Этап	Краткое описание	Статус
1. Гусеничная платформа	Проектирование и изготовление шасси с подвеской, редуктором и обрезиненными гусеницами.	**Завершен**
2. Система картографии	Разработка и сборка гиростабилизированного лидара 360° и системы камер.	**Завершен**
3. Манипулятор	Проектирование, изготовление и доработка 3-осевого манипулятора с захватом и FPV-камерой.	**Завершен**
4. Электроника и ПО	Разработка схемы на ESP32, сборка прототипа и написание ПО.	**Завершен**
5. Лабораторные испытания	Тестирование всех систем прототипа в имитационной среде.	**Завершен**
6. Внедрение ИИ	Интеграция нейросетей для анализа видео и автономной навигации.	В планах
7. Полевые испытания	Оснащение дополнительными датчиками и тесты в реальной пещере.	В планах

ТЕХНИЧЕСКОЕ ОПИСАНИЕ МЕХАНИЗМОВ РАБОТЫ

ПРОЕКТИРОВАНИЕ И ИЗГОТОВЛЕНИЕ ГУСЕНИЧНОЙ ПЛАТФОРМЫ

Проектирование началось с создания полной 3D-модели в САПР КОМПАС-3D [2]. В процессе работы были освоены и другие инструменты, включая SolidWorks CAM для генерации управляющих программ (G-code) для трёхосевого ЧПУ-фрезера. Работа потребовала изучения режимов резания и свойств материалов. Ввиду недостаточного опыта неверный подбор режимов привёл к поломке нескольких фрез при обработке алюминиевого сплава. После фрезеровки детали были доработаны напильником, так как снять фаску с двух сторон на фрезерном станке с ЧПУ было очень сложно [1].

Для достижения расчётной скорости около 15 км/ч при использовании имевшихся в наличии мощных бесколлекторных двигателей (до 40 000 об/мин) потребовалось создание особого редуктора с общим передаточным отношением 1:60. Решение реализовано в виде компактной двухступенчатой системы. Первая ступень — это металлический планетарный редуктор от шуруповёрта с передаточным отношением 1:30, который я интегрировал непосредственно в ведущую шестерню для компактности всей конструкции. Вторую ступень я сделал на двух шестернях с передаточным отношением 1:2. Сделать эту ступень надёжной удалось не с первого раза, и её пришлось несколько раз модернизировать. Изначально использовавшиеся прямозубые шестерни, напечатанные из PETG, создавали чрезмерный шум и вибрации. Изучив литературу [5], я пришёл к выводу о необходимости перепроектирования шестерней на косозубые. С ними редуктор стал работать гораздо тише, однако от нагрева детали выходили из строя. В итоге я перешёл к композитному пластику, который работает при более высоких температурах, и распечатал шестерни из угленаполненного нейлона, см. рис. 1. Печать этим сложным композитным материалом потребовала самостоятельного изготовления термокамеры для своего 3D-принтера и подбора специального клея-карандаша для обеспечения адгезии первого слоя.

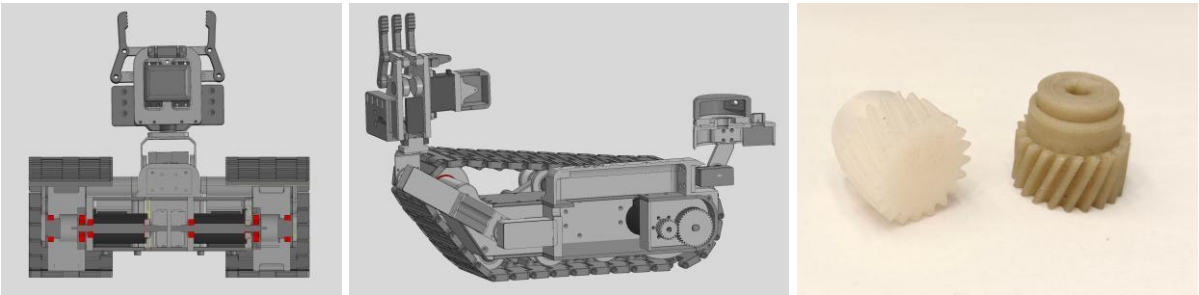


Рис.1. Шестерни и угленаполненного нейлона

В основе ходовой части я сконструировал независимую рычажную подвеску, оснащённую амортизирующими роликами. Эта конструкция обеспечивает автоматическое натяжение гусеничного полотна и кардинально повысила живучесть платформы, устранив характерную для предыдущих моделей проблему поломки роликов при наезде на препятствия за счёт эффективного гашения ударных нагрузок. Для повышения сцепления с поверхностью гусеничные траки я вручную обработал полиуретановым герметиком при помощи монтажного пистолета, см. рис.2. Этот метод позволил моему роботу уверенно передвигаться по скользким поверхностям (камень, лёд), а также снизил общий шум работы. Я пробовал печатать траки из гибких пластиков TPU, но они продемонстрировали недостаточное сцепление.

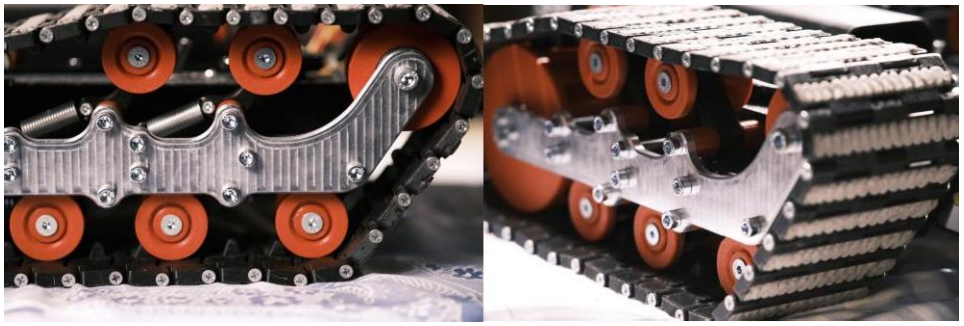


Рис.2. Гусеничные траки, обработанные полиуретановым герметиком

СИСТЕМА НАВИГАЦИИ И КАРТОГРАФИЯ

Для построения точного цифрового двойника пещеры был самостоятельно спроектирован и собран всенаправленный лазерный лидар 360°. В его основе я использовал лазерный дальномер VL53L0X. Данный датчик представляет собой поверхностно-излучающий лазер с вертикальным резонатором (VCSEL) с рабочей длиной волны 940 нм и фоточувствительную матрицу на основе однофотонных лавинных диодов (SPAD). Эта технология обеспечивает надежную работу в условиях повышенного уровня внешнего инфракрасного излучения и не требует дополнительной оптики, что упрощает конструкцию. Дальномер закреплён внутри вращающегося купола, см.рис.3. Питание и передача данных по протоколу I2C осуществляются через скользящие контакты и втулку. Вращение купола обеспечивает шаговый двигатель через ременную передачу, что позволяет сканировать пространство по полному кругу.

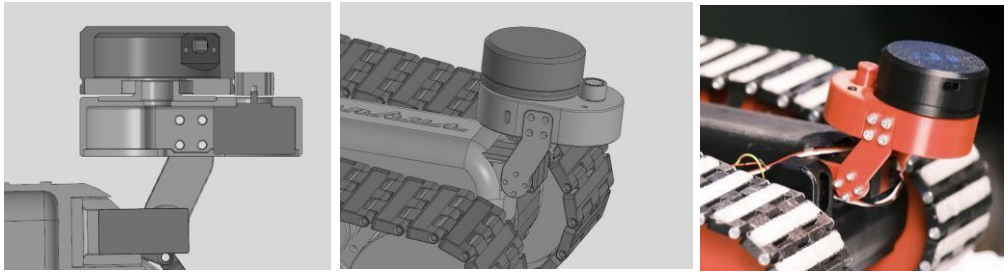


Рис.3. Закрепление дальнометра

Лидар установлен на гиростабилизированную платформу, что обеспечивает корректное сканирование и построение точного облака точек в условиях неровного рельефа. Акселерометр MPU-6050 в реальном времени измеряет углы наклона корпуса робота, позволяя системе программно компенсировать эти отклонения и постоянно удерживать ось сканирования в вертикальном положении.

Помимо лидара, для локальной навигации, избегания препятствий и мониторинга обстановки робот оснащён камерой спереди и дополнительной камерой сзади, см.рис.4. Видеопоток с этих камер в режиме реального времени передаётся на ПК оператора, а также записывается для последующего создания обучающей выборки. Эти данные будут использоваться для обучения нейросетевых моделей с целью детектирования объектов и обеспечения полностью автономного передвижения робота в будущем.

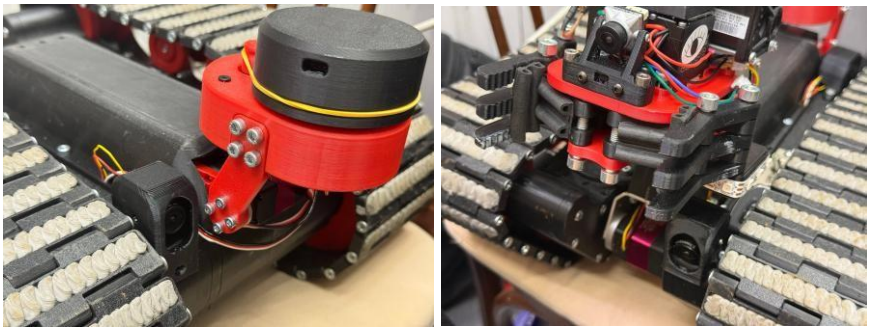


Рис.4. Камеры

ЭВОЛЮЦИЯ МАНИПУЛЯТОРА И СИСТЕМЫ ЗАХВАТА

Конструкция манипулятора прошла несколько итераций развития, я её переработал несколько раз, опираясь на изученный материал [3]. Первоначальный вариант представлял собой двухосевую руку с захватом на одном сервоприводе. Он обладал существенными недостатками: малой гибкостью, перегревом сервопривода и недостаточным усилием сжатия. Текущая версия представляет собой трёхосевой манипулятор с захватом на двух сервоприводах. Кинематическая схема включает две оси в суставах роборуки, обеспечивающих её позиционирование, и одну ось, вращающую клешню захвата, см.рис.5. Это решение позволило достичь нескольких важных улучшений. Во-первых, была достигнута существенная универсальность захвата, позволяющая надёжно удерживать предметы практически любой формы. Во-вторых, усилие сжатия увеличилось. Кроме того, конструкция стала более надёжной благодаря резервированию приводов: в случае отказа одного из них второй может прижать рабочую «лапку» к неисправной, сохранив возможность частичного захвата.

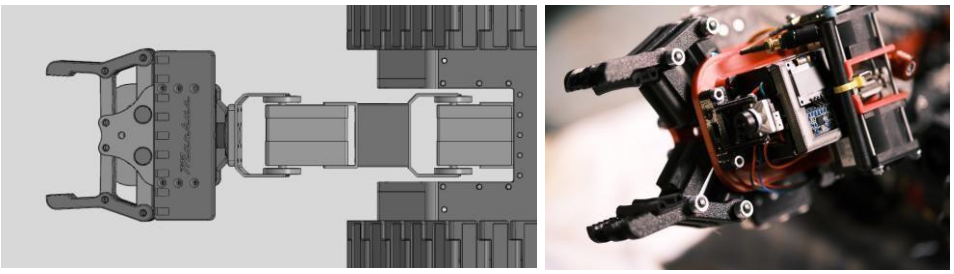


Рис.5. Кинематическая схема

Материал «лапок» также был усовершенствован. Изначально я печатал их из PETG, но в итоге были перепечатаны из ударопрочного и гибкого пластика TPU, см.рис.6, что позволило надёжно удерживать предметы сложной формы и повысило общую стойкость к механическим ударам.

Рис.6. «Лапки»

Для обеспечения точного дистанционного управления и мониторинга в труднодоступных зонах пещеры в захвате закреплена FPV-камера, предоставляющая оператору вид «от первого лица» в реальном времени. Эта система критически важна для визуального контроля процесса захвата образцов и исследования узких пространств.

ДИЗАЙНЕРСКОЕ РЕШЕНИЕ

В своей работе я всегда уделяю особое внимание промышленному дизайну, так как считаю, что красивый и продуманный внешний вид – это залог успеха проекта и важная часть инженерной культуры. Эскизы и концепции не рождаются сразу, порой приходится перерисовывать и корректировать идеи по несколько раз, чтобы конечное устройство было не только высокофункциональным и технологичным, но и эстетически цельным, удобным для восприятия и взаимодействия, см.рис.7. Сам процесс разработки для меня всегда начинается с классического инструмента – карандаша и листа бумаги, что позволяет свободно исследовать формы и композицию, прежде чем переходить к точному цифровому моделированию.

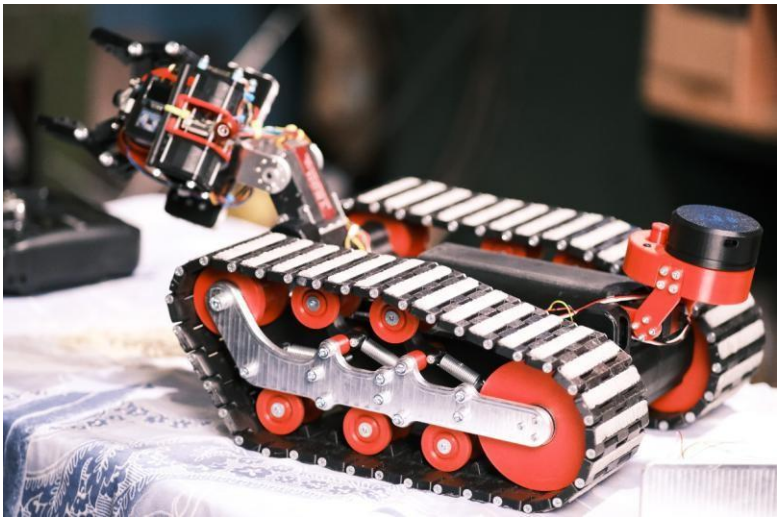


Рис. 7. Дизайн

Электроника, программирование и управление

Архитектура управления роботом построена по распределённому принципу для повышения надёжности и эффективности обработки данных. Структурную схему подключения электронных компонентов представлена в прил.2.

Основной модуль управления исполнительными механизмами и платформой построен на микроконтроллере ESP32, архитектура которого широко используется в DIY проектах [4]. Алгоритм работы программы представлен в прил.4. Этот центральный контроллер отвечает за управление сервоприводами манипулятора, взаимодействие с силовыми контроллерами ходовых двигателей, а также сбор и первичную обработку данных с лазерного лидара и акселерометра, см рис.8.

Для разгрузки основной шины данных и обеспечения работы интеллектуальных функций манипулятора была реализована вложенная система управления на микроконтроллере ESP32-S3. Этот дополнительный модуль расположен непосредственно в клешне захвата и выполняет несколько ключевых задач: управление лазерным дальномером для автономного определения расстояния до объекта, обработка данных с собственного гироскопа для стабилизации движения руки, контроль подсветки и вывод служебной информации на встроенный OLED-дисплей. Коммуникация с основной платой ESP32 осуществляется по шине I2C, что позволило радикально сократить количество проводов, тянущихся к подвижным элементам манипулятора, повысив надёжность и гибкость конструкции. Видеопоток с FPV-камеры, установленной на захвате, передаётся по отдельному радиоканалу на приёмник, что позволяет транслировать его как в FPV-очки для управления от первого лица, так и на ноутбук оператора. Для обеспечения гибкости видеомониторинга с передней и задней камер используется один видеопередатчик, подключённый к двум камерам через сплиттер с дистанционным переключением; переключение между камерами осуществляется тумблером на пульте оператора.

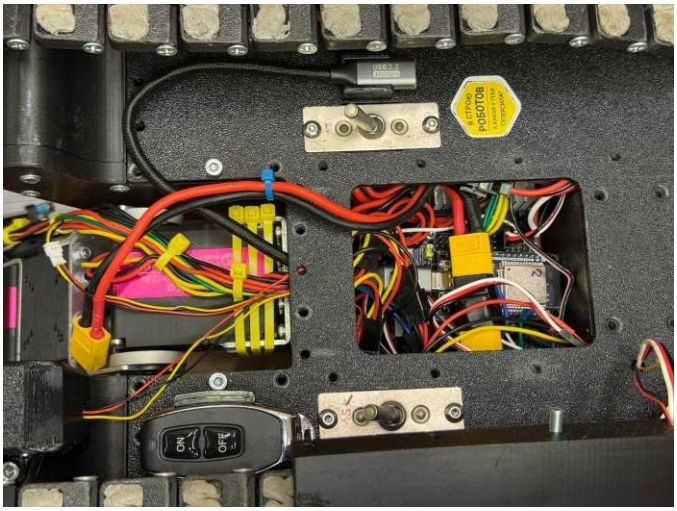


Рис. 8. Центральный контроллер

Высокоуровневые задачи обработки данных и построение цифрового двойника пещеры выполняются на ноутбуке оператора, см.рис.9. При нахождении робота в зоне приёма Wi-Fi сигнала оператор может в реальном времени выгружать облако точек отсканированной пещеры для построения и уточнения её 3D-модели.

Операторский интерфейс обеспечивает гибкое управление: оператор может использовать радиоуправляемый пульт для ручного режима (в том числе с FPV-очками) или управлять роботом с компьютера, получая полную телеметрию, 3D-карту в реальном времени и видеопотоки со всех камер, см.прил.3.

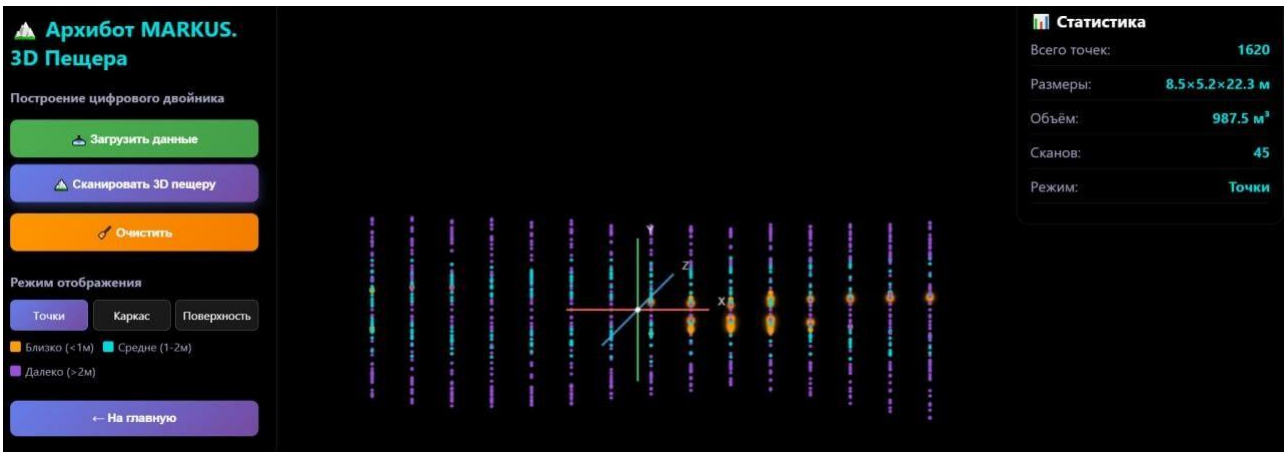


Рис. 9. Обработка данных

ИСПОЛЬЗОВАНИЕ РЕЗУЛЬТАТОВ И ИСПЫТАНИЯ

Функционирующий прототип робота Архибот MARKUS успешно прошёл комплексные испытания в условиях, имитирующих среду пещеры. Были проверены ключевые способности робота: преодоление снежных заносов и движение по неровному грунту подтвердили высокую проходимость и надёжность гусеничной платформы с независимой подвеской, см.рис.10.

В ходе тестов система лидара продемонстрировала способность автономно строить точное облако точек замкнутого пространства с радиусом действия до 5 метров, что является основой для построения цифрового двойника пещеры. Манипулятор с обновлённой конструкцией захвата надёжно захватывал и транспортировал объекты различной формы, подтвердив эффективность перехода на двойные сервоприводы и гибкие TPU-лапки.

Многоканальная система видеомониторинга (FPV-камера на захвате, а также передняя и задняя камеры) в сочетании с гибким переключением обеспечила оператору полный визуальный контроль над обстановкой как в режиме дистанционного управления по радио, так и при работе с компьютера, получающего телеметрию и предварительную 3D-карту. Испытания доказали, что робот готов к полевым испытаниям в реальных пещерах для решения задач картографии и сбора образцов.

Рис. 10. Испытания робота

ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ И КОММЕРЧЕСКИЕ ПЕРСПЕКТИВЫ

Разработанный робот «Архибот MARKUS» является практико-ориентированным решением для актуальных задач в различных сферах:

1. Научные исследования: Автономная картография пещер, сбор геологических и археологических образцов, мониторинг подземной среды без риска для человека.
2. Аварийно-спасательные работы (МЧС): Разведка завалов, поиск в опасных и недоступных зонах, оценка обстановки после техногенных катастроф или природных бедствий.

3. Образование и просвещение: Робот служит наглядной учебно-методической базой для демонстрации принципов современной робототехники, мехатроники, систем управления и цифрового моделирования.
4. Промышленный контроль и инспекция: Платформа обладает потенциалом для адаптации к задачам обследования труднодоступных инженерных сооружений, таких как коллекторы, тоннели и трубопроводы.

«Архибот MARKUS» занимает специализированную нишу на рынке робототехнических решений. Его целевыми заказчиками являются научно-исследовательские институты, инжиниринговые компании и подразделения МЧС — организации, готовые инвестировать в эффективное и безопасное решение для работы в сложных условиях. Ключевым конкурентным преимуществом проекта является его низкая себестоимость (около 45 тыс. рублей за прототип), которая на порядок ниже стоимости существующих промышленных аналогов (от 500 тыс. рублей).

Для оценки ёмкости рынка (Serviceable Obtainable Market, SOM) в России на горизонте 3-5 лет использован метод «снизу-вверх», см.Табл.3.

Таблица 3. Оценка объема рынка

Параметр анализа	Оценка для проекта «Архибот MARKUS»	Комментарий
Целевые сегменты (B2B / B2G)	Научные институты, инжиниринг, МЧС	Узкая, но специализированная ниша.
Ключевое ценовое преимущество	Себестоимость прототипа: ~45 тыс. руб.	Позволяет установить конкурентную цену в 5-10 раз ниже импортных аналогов.
Прогнозная цена единицы	900 тыс. руб.	Рассчитана как себестоимость + наценка на разработку и комплектацию.
Потенциальные клиенты (РФ), ед.	~100 организаций (вузы, компании, МЧС)	Оценка основана на количестве профильных организаций.
Реалистичный годовой объём продаж (SOM)	10-15 роботов	Консервативная оценка для первых лет коммерциализации.
Годовая ёмкость рынка (SOM)	13.5 млн руб.	Рассчитано: 15 ед. * 900 000 руб.
Совокупная ёмкость за 3 года	40.5 млн руб.	Консервативный прогноз при успешном выходе на рынок.

ЗАКЛЮЧЕНИЕ

В рамках проекта был спроектирован, изготовлен и испытан автономный гусеничный робот «Архибот MARKUS» для исследования пещер. Работа представляет собой законченный инженерный цикл: от идеи, рождённой в диалоге с учёным-спелеологом, через сложный процесс проектирования, многочисленные итерации и поиск технологических решений — к работающему прототипу, готовому к применению в реальных условиях. Записал видеовизитку с обзором и тестированием робота [6]. Подготовил 3D тур с виртуальным обзором Архибота MARKUS [7]. Технические характеристики робота представлены в прил.1.

Достигнутые результаты:

1. Разработана высокопроходимая гусеничная платформа с уникальной независимой подвеской и компактной двухступенчатой силовой передачей на основе планетарного редуктора и косозубых шестерён из угленаполненного нейлона. Обрезиненные полиуретановым герметиком гусеницы обеспечивают надёжное сцепление на скользких поверхностях.
2. Создана и интегрирована система 3D-картографии на базе самодельного лидара 360°, установленного на гиростабилизированной платформе с акселерометром. Это позволяет строить точные облака точек помещений радиусом до 5 метров.
3. Усовершенствована конструкция трёхосевого манипулятора: переход на два сервопривода в захвате и «лапки» из гибкого пластика TPE значительно увеличили усилие сжатия, надёжность и позволили захватывать предметы сложной формы.
4. Реализована распределённая система управления и многоканальное наблюдение: архитектура на микроконтроллерах ESP32/ESP32-S3, FPV-камера на захвате, а также передняя и задняя камеры с переключаемым видеопотоком обеспечивают оператору полный визуальный контроль и гибкое управление.

Перспективы развития проекта:

1. Внедрение ИИ: Интеграция нейросетей (распознавание объектов) и обеспечения полностью автономной навигации в сложных условиях.
2. Расширение сенсорного комплекса: Установка газоанализаторов, дозиметров, лидаров с большей дальностью для сбора комплексных данных о состоянии подземной среды.
3. Повышение автономности: Оптимизация энергопотребления, установка дополнительных аккумуляторов, разработка системы автоматической подзарядки для увеличения продолжительности миссий.
4. Коммерциализация: Создание коммерческой версии, патентование ключевых решений (подвеска, крепление лидара), адаптация платформы для задач

промышленной инспекции (трубопроводы, коллекторы) и поиск партнёров в сферах МЧС и геологоразведки.

Проект «Архибот MARKUS» стал значимым шагом в моей инженерной практике, позволив синтезировать знания в области механики, материаловедения, электроники и программирования для создания комплексной робототехнической системы, решающей социально-значимые научные и прикладные задачи.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Богданов В.К. Опиливание металла: методические указания к лабораторной работе/ В.К. Богданов. – Оренбург: – ИПК ГОУ ОГУ, 2007. – 30 с.
2. Никонов Вячеслав. КОМПАС-3D: создание моделей и 3D-печать./В.В.Никонов — СПб.: Питер, 2020. — 208 с.: ил. — (Серия «Учебное пособие»).
3. Робототехнические системы: Учебное пособие / В. Г. Хомченко — Омск: ОмГТУ – 195 стр., 2016.
4. Момот М. В. Мобильные роботы на базе ESP32 в среде Arduino IDE./ М. В. Момот— СПб.: БХВ-Петербург, 2020. — 272 с.
5. В. Е. Старжинский, Б. П. Тимофеев,Е. В. Шалобаев, А. Т. Кудинов. Пластмассовые зубчатые колеса в механизмах приборов. Расчет и конструирование. Справочное и научное издание/В. Е. Старжинский, Б. П. Тимофеев, Е. В. Шалобаев, А. Т. Кудинов. Под общ. ред. В. Е. Старжинского и Е. В. Шалобаева. Санкт-Петербург-Гомель: ИММС НАН Б, 1998. – 538 с.
6. Герасименко Георгий Андреевич. Обзор и создание Архиробота MARKUS / Г.А. Герасименко / [Электронный ресурс]. – Режим доступа: https://disk.yandex.ru/d/Lc0FgNV2IStmUA
7. Герасименко Георгий Андреевич. 3D тур с виртуальным обзором Архибота MARKUS на платформе sketchfab.com/Г.А. Герасименко / [Электронный ресурс]. – Режим доступа: https://sketchfab.com/3d-models/markus-8d22237c1b5a47d5af8c5166fee2738f

ПРИЛОЖЕНИЕ 1. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

Таблица 4. Технические характеристики

Наименование параметра	Значение
1. Общие параметры	
1.1. Эксплуатационная масса, кг	8
1.2. Габаритные размеры (Д×Ш×В), мм	500 × 400 × 300
1.3. Напряжение питания, В	16
1.4. Время автономной работы, ч	до 3
1.5. Основные материалы конструкции	Пластик PETG, нейлон (с углеродным наполнителем), TPU. Алюминиевый сплав
2. Ходовая часть и привод	
2.1. Тип движителя	Гусеничный
2.2. Тип подвески	Независимая рычажная
2.3. Количество ведущих двигателей, шт	2
2.4. Тип двигателей	Бесколлекторные, с интегрированным планетарным редуктором
2.5. Максимальная скорость передвижения, км/ч	~15
3. Система навигации и картографии	
3.1. Тип лидара	Всенаправленный (360°) на гиростабилизированной платформе
3.2. Чувствительный элемент лидара	Лазерный дальномер VL53L0X (VCSEL, 940 нм)
3.3. Максимальный радиус сканирования, м	5
3.4. Система ориентации в пространстве	Инерциальный измерительный модуль MPU-6050 (акселерометр, гироскоп)
3.5. Количество видеокамер, шт	3 (FPV на захвате, передняя обзорная, задняя обзорная)
4. Манипулятор	
4.1. Количество степеней подвижности	3
4.2. Конфигурация осей	2 оси в суставах, 1 ось вращения клешни
4.3. Количество приводов захвата, шт	2
4.4. Материал рабочих органов захвата	Термополиуретан TPU (марка A95)
4.5. Максимальное рабочее раскрытие захвата, мм	70
5. Система управления и связи	
5.1. Основной управляющий контроллер	Микроконтроллер ESP32
5.2. Локальный контроллер манипулятора	Микроконтроллер ESP32-S3
5.3. Интерфейс связи с датчиками	I^2C, ШИМ
5.4. Каналы связи с оператором	Wi-Fi (передача данных), аналоговый радиоканал 5.8 ГГц (FPV-видео)
5.5. Режимы управления	Ручное, автономное

ПРИЛОЖЕНИЕ 2. СТРУКТУРНАЯ СХЕМА ЭЛЕКТРОННЫХ КОМПОНЕНТОВ РОБОТА

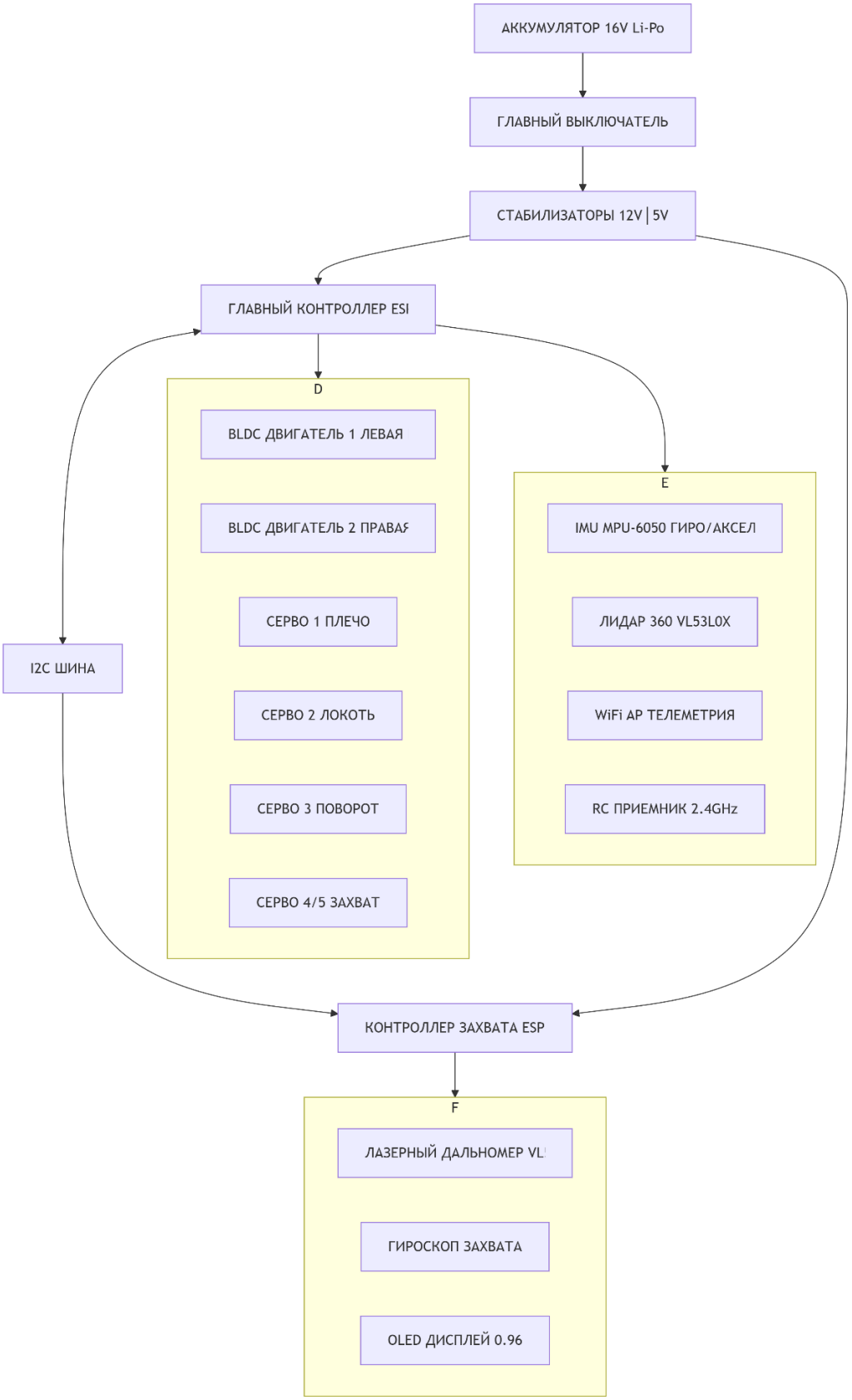


Рис. 11. Структурная схема электронных компонентов робота

ПРИЛОЖЕНИЕ 3. СТРУКТУРНАЯ СХЕМА ВИДЕОПОДСИСТЕМЫ И РАДИООБМЕНА РОБОТА

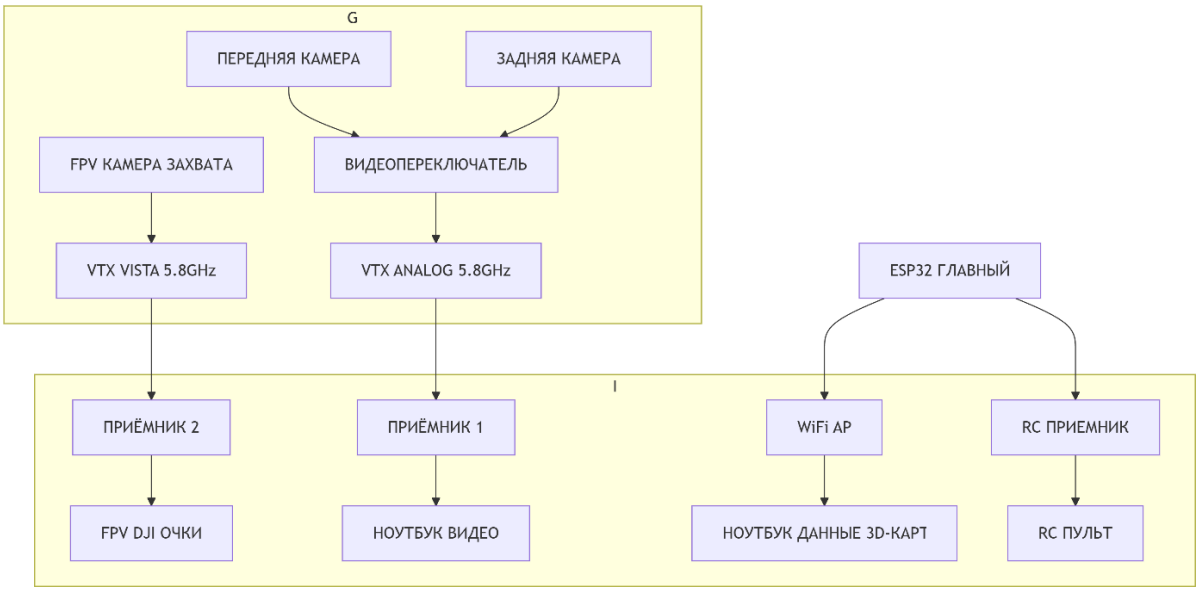


Рис. 12. Структурная схема видеоподсистемы и радиообмена робота

ПРИЛОЖЕНИЕ 4. АЛГОРИТМ РАБОТЫ ПРОГРАММЫ ДЛЯ ОСНОВНОЙ ESP32

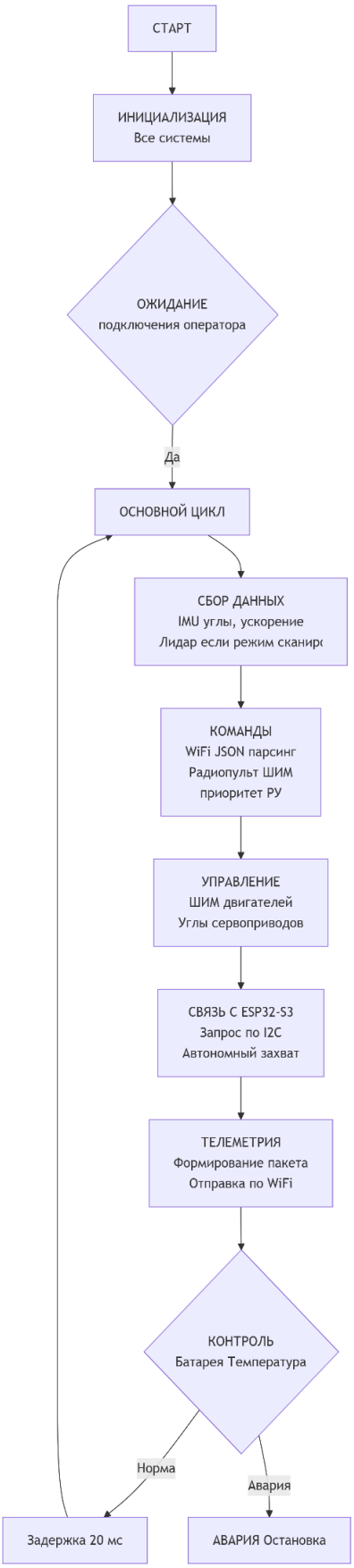


Рис. 13. Алгоритм работы программы для основной ESP32

РАЗРАБОТКА И СОЗДАНИЕ АВТОНОМНОГО МНОГОФУНКЦИОНАЛЬНОГО НАВИГАЦИОННОГО КОМПЛЕКСА «POLLUX NAVIGATOR»

Василенок Аксинья Васильевна

Москва г, ГБОУ Школа №1569 «Созвездие», 11 класс

Секция: Навигация и системы управления летательными аппаратами

Работа выполнена без научного руководителя

АННОТАЦИЯ

Мой проект «Pollux Navigator» представляет собой разработку автономного многофункционального навигационного комплекса, построенного на базе микроконтроллера Arduino Mega 2560. Устройство интегрирует данные с GPS-приемника, метеодатчика BME280 (температура, давление, влажность), с часов реального времени (RTC) и с акселерометра, предоставляя пользователю единый комплекс инструментов для навигации, мониторинга окружающей среды и обеспечения безопасности в условиях активного отдыха.

Думая над названием, я вдохновилась подвигами русских мореплавателей и полярных летчиков. Они использовали для навигации одну из ярчайших звезд в созвездии Близнецов под названием Поллукс. Так же, эта звезда является символом точности и надежности. Тем самым, я хотела отдать дань уважения легендарным русским путешественникам и советским летчикам - Никитину, Дежневу, Берингу, Чкалову, Леваневскому… Такое название, на мой взгляд, символизирует преемственность навигационных принципов - от астрономических ориентиров, использовавшихся мореплавателями и полярными летчиками прошлого века, к современным спутниковым технологиям, воплощенным в проекте.

Ключевые результаты разработки:

- Аппаратная платформа: Разработан и собран прототип проекта на основе Arduino Mega с интеграцией 8 периферийных модулей (GPS, BME280, OLED, SD-карта, RTC, зуммер, вибромотор, акселерометр). Выбор компонентов обусловлен критериями доступности, надежности и аппаратной совместимости.
- Программное обеспечение: Разработано комплексное ПО на C++ с использованием специализированных библиотек (TinyGPS++, Adafruit_SSD1306, RTClib и др.), реализующее многоуровневое меню, обработку прерываний от кнопок, сбор данных с датчиков и их логирование на SD-карту.

- Уникальный функционал: Реализована трехуровневая система SOS со световой, звуковой и тактильной (вибрационной) сигнализацией; адаптивные режимы работы для четырех типов местности с контекстными рекомендациями по выживанию; набор навигационных инструментов (3D-компас, трекер солнца, калькулятор площади).
- Тестирование: Проведены лабораторные и полевые испытания в условиях реального похода, подтвердившие точность измерений GPS (±2.5 м) и метеодатчика (±1°C для температуры, ±2 hPa для давления), а также время автономной работы до 14 часов в смешанном режиме.

Улучшения по сравнению с аналогами:

- Комплексность решения: В отличие от коммерческих аналогов (Garmin, Navitel, RGK), ориентированных в основном на навигацию, Pollux Navigator делает доступным как навигацию, так и метеомониторинг, имеет необходимые для походных условий утилиты и инструменты безопасности.
- Безопасность и обучение: Реализована уникальная система контекстных рекомендаций по выживанию, имеющая трехуровневый SOS-сигнал, которая отсутствует в большинстве коммерческих навигаторов.
- Экономическая эффективность: Себестоимость разработки значительно ниже коммерческих аналогов при сохранении широкой функциональности.
- Открытая платформа: В отличие от закрытых проприетарных систем, Pollux Navigator построен на открытой платформе (Arduino), что позволяет легко модифицировать и расширять его функционал под конкретные потребности пользователя.
- Адаптивность: Четыре предустановленных режима работы («Горы летом/Горы зимой», «Лес летом/Лес зимой») автоматически настраивают набор отображаемых данных и предоставляют контекстные рекомендации по выживанию и оценке рисков. Это особенно ценно для неподготовленных пользователей, так как устройство не просто показывает данные, а интерпретирует их к конкретной среде.

Разработанный мной «Pollux Navigator» наглядно демонстрирует, что создание эффективного и многофункционального навигационного комплекса, с упором на безопасность пользователя, возможно на основе доступных компонентов и открытых технологий. Это способствует созданию недорогих, но надежных решений для широкого круга любителей активного отдыха.

ВВЕДЕНИЕ

ОБОСНОВАНИЕ АКТУАЛЬНОСТИ ПРОЕКТА

Актуальность разработки обусловлена растущим интересом к активным видам туризма, который очень быстро развивается в России (именно наша страна является одной из самых популярных стран для туристических поездок, с 2021 года она входит в топ-20 самых посещаемых стран мира; туристов привлекает не только богатая история, многоликая культура, но и природное многообразие нашей страны) и одновременным отсутствием на рынке доступных комплексных решений, интегрирующих навигацию, мониторинг среды и функции безопасности в одном устройстве.

Анализ рынка выявил ряд проблем, присутствующих в существующих устройствах:

- Фрагментированность аппаратных решений: Пользователь вынужден применять несколько узкоспециализированных устройств (навигатор, метеостанцию, трекер), каждое из которых требует отдельного источника питания и отдельного обслуживания.
- Низкий уровень аналитической обработки данных: Большинство коммерческих устройств ограничиваются отображением сырых данных (raw data) без их комплексного анализа и применимости к конкретным условиям эксплуатации.
- Критическая зависимость от смартфонов: Использование мобильных приложений для навигации сопряжено с трудностями при отсутствии связи или интернета, а также, с высоким энергопотреблением и уязвимостью электроники смартфона к воздействию внешней среды (влага, температура, механические повреждения).
- Не адаптивность интерфейсов для кризисных ситуаций: Сенсорные интерфейсы и сложные меню коммерческих продуктов плохо приспособлены к использованию в состоянии стресса, когда требуется максимальная простота и скорость доступа к функциям безопасности.

Разрабатываемый комплекс «Pollux Navigator» предлагает подход к решению указанных проблем через:

- Аппаратную интеграцию ключевых функций в едином корпусе.
- Программную интерпретацию данных с датчиков для формирования прогнозов и предупреждений.
- Энергоэффективную архитектуру, обеспечивающую длительную автономную работу.

- Эргономичный интерфейс на основе физических кнопок и многоуровневого меню, оптимизированный для использования в условиях повышенной нагрузки.
- Практическая значимость проекта заключается в создании комплексного решения, повышающего безопасность и комфорт активных видов туризма, в том числе и за счет устранения ключевых недостатков существующих на рынке продуктов.

ЦЕЛИ И ЗАДАЧИ ПРОЕКТА

Целью проекта является разработка и создание функционального образца автономного многофункционального комплекса «Pollux Navigator», предназначенного для обеспечения навигации, мониторинга условий окружающей среды и повышения уровня безопасности для туристов и участников экспедиций.

Заявление об авторстве. Настоящая проектная работа выполнена мной, Василенок Аксинией Васильевной, самостоятельно. Все этапы - выбор концепции, подбор компонентов, схемотехника, изготовление и вытравливание платы, пайка, сборка прототипа, создание документации, инструкции, разработка и печать корпуса на 3D принтере, разработка программного обеспечения и проведение лабораторных и полевых испытаний - выполнены без посторонней помощи. В приложении приложены: фотопроцесса сборки, изготовления платы, скриншоты кусков кода с комментариями и логи полевых испытаний, документация к проекту.

Для достижения поставленной цели были поставлены следующие задачи:

1. Аппаратная реализация:
   - Разработана схема и собрана аппаратная платформа на базе микроконтроллера Arduino Mega 2560.
   - Обеспечена интеграция всех периферийных модулей: GPS-приемника, датчиков BME280, OLED-дисплея, модуля RTC, кардридера microSD, зуммера, вибромотора и кнопочного интерфейса.
   - Питание устройства осуществляется от литий-ионного (Li-Ion) аккумулятора формата 18650 с номинальным напряжением 3.7V и емкостью ~2200 mAh. Напряжение повышается до 5V с помощью импульсного повышающего преобразователя (DC-DC step-up) для стабильной работы всех компонентов системы. Для питания от внешнего источника (например, Power Bank) реализован разъем USB Type-C.
2. Программная реализация: Было разработано комплексное программное обеспечение на C++, включающее следующие ключевые компоненты:

3. Многозадачное ядро на основе конечного автомата (state machine). Обеспечивает стабильную работу всех датчиков и модулей, управление многоуровневым меню и обработку прерываний от кнопок с поддержкой коротких и долгих нажатий.
4. Система сбора и обработки данных: Реализованы алгоритмы чтения и фильтрации данных с GPS-приемника (NEO-7M), датчика BME280 (температура, давление, влажность) и часов реального времени (DS3231). На основе анализа барометрического тренда реализован прогноз погоды, а на основе комплекса данных - оценка рисков для четырёх сред.
5. Навигационные алгоритмы: Реализованы функции отображения текущих координат, скорости и курса; сохранения и загрузки маршрутов (до 50 точек) с SD-карты; вычисления расстояния по обратной формуле Винсенти (Vincenty's formulae) и азимута до цели по формуле гаверсинусов; функции «Возврат к точке».
6. Навигационные инструменты: Разработан набор инструментов, включающий 3D-компас с учётом магнитного склонения, калькулятор площади участка по точкам (метод Гаусса), трекер солнца с расчётом времени до заката и построитель профиля высот.
7. Аварийная подсистема: Реализована трехуровневая система SOS со световой (инверсия экрана), звуковой и тактильной (вибрация) сигнализацией, использующая кодирование Morse для идентификации.
8. Реализация системы безопасности:
   - Разработана и реализована многоуровневая система сигнализации SOS, которая активируется одной кнопкой.
   - Создана встроенная база знаний по выживанию и оказанию первой помощи, адаптированная под четыре режима.
   - Реализована функция автоматического логирования координат и метеоданных при активации режима SOS.
   - При активации третьего уровня (ALERT) каждые 3 минуты в файл SOS.LOG записываются текущие координаты.
9. Разработка пользовательского интерфейса:
   - Спроектирован и реализован интуитивный многоуровневый интерфейс на OLED-дисплее.

- Внедрена система индикации долгого нажатия кнопок (на дисплее появляется прогресс-бар, который постепенно заполняется, показывая пользователю оставшееся время удержания) для предотвращения случайных активаций.

10. Тестирование и отладка:

- Полевые испытания в лесных условиях.
- Для проверки работы устройства в условиях плотной кроны было проведено 3 полевых испытания в смешанном лесу, открытом поле в лесу, около реки.
- Перед каждым тестом навигатор полностью выключался, после чего выполнялся «холодный старт». В процессе фиксировались: количество принимаемых спутников, время до первого определения координат, стабильность положения и отклонение от контрольной точки (сверялось по онлайн картам на телефоне), продолжительность каждого испытания - не менее 30 минут.
- В ходе тестирования среднее число спутников составляло 6–8, точность - ±3–8 м (в зависимости от плотности листвы), время первого определения координат (при холодном старте) - 60–180 секунд. При ослаблении сигнала применялось сглаживание координат, что уменьшало резкие скачки положения.
- Результаты подтверждают работоспособность устройства в лесистой местности.
- Тестирование в плотной городской застройке.
- Для оценки работы в условиях отражённых сигналов (multipath) и ограничения видимости неба было проведено испытание в зоне высотной застройки (улица с высотными зданиями, середина дня) в городе Дубна. В условиях городского каньона среднее число спутников снижалось до 4–6.
- Проверена энергоэффективность и определено время автономной работы.
- Отработана надежность работы системы в условиях имитации экстремальных ситуаций.

Примечание о состоянии программного обеспечения:

Разработанное программное обеспечение представляет собой полнофункциональный прототип, успешно реализующий все заявленные в проекте функции: навигацию, мониторинг среды, систему безопасности и адаптивные режимы. Код написан на C++ с использованием стандартных для платформы Arduino библиотек и обладает модульной архитектурой.

Данная реализация является доказательством работоспособности концепции и предназначена для тестирования алгоритмов и пользовательского интерфейса. Его текущая

структура заложила основу для будущей разработки, так как легко поддается рефакторингу, оптимизации и расширению. Например, уже сейчас в коде зарезервированы интерфейсы и предусмотрена логика для добавления будущих режимов «Group» и «Pro», что демонстрирует его потенциал для развития. Код готов к использованию, но его архитектура открыта для дальнейшего совершенствования с точки зрения оптимизации потребления памяти, быстродействия и добавления нового функционала.

Примечание: поскольку проект является индивидуальным, распределение задач по исполнителям не требовалось. Все задачи были выполнены единственным участником.

АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

1. Профессиональные GPS-навигаторы (Garmin, Suunto)

   Что представляют собой: специализированные устройства для навигации, ориентирования и отслеживания маршрутов. Имеют защищённый корпус, длительное время автономной работы и заранее загруженные карты.

   Недостатки:

   - Высокая стоимость. Качественные модели сопоставимы по цене со стоимостью смартфонов.
   - Закрытость платформы. Пользователь не может кастомизировать функционал под свои специфические нужды (например, добавить расчёт риска схода лавины на основе локальных данных о температуре и давлении).
   - Ограниченный набор датчиков. Как правило, фокус именно на навигации, без глубокой интеграции датчиков (BME280) для комплексного анализа окружающей среды.
   - Функционал безопасности (SOS) часто является платной подпиской и требует подключения к спутниковой сети.

2. Спортивные smart-часы с функцией навигации

   Что представляют собой: умные часы с поддержкой GPS/ГЛОНАСС, функцией записи треков, пульсометром.

   Недостатки:

   - Маленький экран. Невозможно комфортно работать с картами и большими объёмами информации (например, читать руководство по выживанию).
   - Ограниченная автономность. При активном использовании GPS требуют ежедневной зарядки, что неприемлемо в многодневном походе.

- Хрупкость. Сенсорные экраны и общая конструкция не всегда адаптированы для экстремальных условий (удары, падения в воду, низкие температуры).
- Отсутствие специализированных режимов. Нет предустановленных сценариев для разных сред (лес зимой, горы летом и т.д.).

3. Мобильные приложения для outdoor-активностей (например, OsmAnd, Mapy.cz, AllTrails)

   Что представляют собой: приложения для смартфонов, использующие его GPS-модуль и возможности для навигации и записи треков.

   Недостатки:

- Полная зависимость от смартфона. Смартфон имеет крайне высокое энергопотребление, из-за чего, в походе, его батарея садится за несколько часов, но ее необходимо беречь для экстренных случаев.
- Уязвимость к погодным условиям. Смартфоны плохо переносят дождь, снег, отрицательные температуры и прямые солнечные лучи.
- Отсутствие связи. Функционал навигации и, что критично, SOS-сигнал недоступны в зонах без доступа сотовой сети.
- Отсутствие прямой интеграции с внешними датчиками для сбора метеоданных.

Недостатки, которые закрывает моя проектная работа "Pollux Navigator":

1. Автономность и энергоэффективность: Устройство построено на Arduino, что потребляет минимум энергии по сравнению со смартфоном, может работать от батарейки-аккумулятора долгое время.
2. Аппаратная независимость: Устройство не требует наличия сотовой сети или подключения к интернету. Все функции, включая полноценный SOS-сигнал с записью координат на SD-карту, работают оффлайн.
3. Комплексность решения: В отличие от узкоспециализированных аналогов, проект объединяет в одном устройстве:
   - Навигацию (GPS, компас, запись маршрутов, "трек назад").
   - Мониторинг среды (температура, давление, влажность, прогноз погоды на основе барометрического тренда).
   - Инструменты безопасности (многорежимная SOS-система со световой, звуковой и вибро-индикацией, руководство по выживанию и первой помощи, адаптированное под четыре типа местности).

- Специализированные инструменты (трекер солнца, калькулятор площади, построение профиля высот).

4. Адаптивность и кастомизация: Прошивка устройства позволяет легко добавлять новые режимы работы, датчики и функции, что невозможно в коммерческих продуктах с закрытым ПО.
5. Ценовая доступность: Стоимость компонентов проекта на порядок ниже цен профессионального навигатора с сопоставимым (но всё же ограниченным) функционалом.
6. Устойчивость к условиям: Использование монохромного OLED-дисплея и пассивных компонентов делает устройство более устойчивым к низким температурам и прямым солнечным лучам, по сравнению с дисплеем смартфона.

Для наглядности ключевые различия между разработанным устройством и коммерческими аналогами представлены в сводной таблице 1

Таблица 1. Ключевые различия между устройством и аналогом

Параметр	Pollux Navigator	Garmin GPSMAP 65s	Navitel G500	RGK NV-20
Цена	~ 4.000	~ 57.000	~ 8.000	~ 25.000
Навигационные системы	GPS (поддержка GLONASS запланирована в будущих версиях)	GPS, GLONASS, Galileo, QZSS, IRNSS (5 систем)	GPS, GLONASS	GPS, GLONASS, Beidou
Датчики	GPS, BME280 (температура, давление, влажность), RTC, акселерометр	Альтиметр, барометр, 3-осевой компас	Отсутствует	Отсутствует
Безопасность и SOS	Трехуровневая система SOS (свет, звук,	Отсутствует	Отсутствует	Отсутствует

	вибрация), руководство по выживанию и первой помощи			
Адаптивные режимы	4 режима (горы летом, горы зимой, лес летом, лес зимой) с контекстными и рекомендациями	Отсутствует	Отсутствует	Отсутствует
Интерфейс	Многоуровневое меню на OLED, 7 кнопок управления, индикация долгого нажатия	Сенсорное/кнопочное управление	Сенсорный экран	Джойстик и кнопки
Память и карты	MicroSD карта до 32 ГБ (хранение маршрутов, логов и точек интереса)	16 ГБ встроенной + MicroSD до 32 ГБ	Встроенная память	4 ГБ встроенной + MicroSD до 32 ГБ
Автономность	Около 14 часов в	16 часов (от 2х АА батареек)	Около 1 часа (от	15 часов (от 2х АА батареек)

	смешанном режиме (активный режим + режим энергосбережения)		аккумулятора 950 мАч)	
Защита от воздействий	Корпус обеспечивает базовую защиту (возможна доработка до IP67)	IPX7 (водонепроницаемость)	Стандартный корпус	IPX7 (водонепроницаемость)
Дополнительные функции	Метеомониторинг, компас, трекер солнца, калькулятор площади	Bluetooth, ANT+, подключение к смартфону	База радаров, обновляемая через ПК	Калькулятор, календарь, секундомер
Платформа	Открытая платформа (Arduino Mega 2560), возможность модификации	Закрытая проприетарная платформа	Закрытая проприетарная платформа	Закрытая проприетарная платформа
Обновления ПО	Возможность самостоятельного обновления и доработки	Обновления через фирменное ПО	Обновления через ПК	Обновления через ПК

Таким образом, мой проект «Pollux Navigator», не просто повторяет существующие на рынке решения, а предлагает уникальное, комплексное и доступное устройство, исправляющее ключевые недостатки аналогов и предназначенное для обеспечения безопасности и комфорта в условиях автономного пребывания в природной среде.

ОПИСАНИЕ РАЗРАБОТАННОГО РЕШЕНИЯ

СУТЬ РАЗРАБОТКИ

Разработанное устройство «Pollux Navigator» представляет собой портативный аппаратно-программный комплекс на базе микроконтроллера платы Arduino Mega 2560. Это многофункциональный навигатор и помощник для активного отдыха, предназначенный для повышения безопасности и эффективности передвижения в различных природных условиях

Ключевые функции устройства имеют:

1. Адаптивный интерфейс с четырьмя предустановленными режимами работы («Горы летом», «Горы зимой», «Лес летом», «Лес зимой»). В зависимости от выбранного режима, меняется набор отображаемой информации - справочных материалов (гиды по выживанию, номера спасательных служб, оценка рисков), обеспечивающих максимальное соответствие данных для текущей среды.

   Архитектура проекта является модульной и масштабируемой, что закладывает основу для будущего развития и добавления новых режимов работы, таких как «Group» для групповых походов и «Pro» для профессиональных экспедиций.

   Пользовательский интерфейс реализован по двухуровневой схеме. На верхнем уровне находятся четыре адаптивных режима работы («Горы летом», «Горы зимой», «Лес летом», «Лес зимой»), определяющие набор анализируемых рисков и содержание справочных материалов. На нижнем уровне главный экран устройства имеет четыре режима отображения (навигация, мониторинг среды, возврат к точке, сводная информация), которые оперативно переключаются соответствующими кнопками. Эта структура обеспечивает как адаптивность к внешним условиям, так и быстрый доступ к ключевым функциям

2. Навигация и ориентирование:

   - Спутниковые системы и работа приёмника. В качестве основного GNSS-модуля в прототипе используется модуль u-blox NEO-7M, обеспечивающий приём спутниковой системы GPS (L1 C/A). На текущем этапе проект ориентирован на GPS; поддержка дополнительных систем (GLONASS, Galileo, BeiDou) заявлена в дорожной карте, как приоритет для будущих версий.

Практические ограничения NEO-7M: приём отдельных систем увеличивает количество видимых спутников и улучшает устойчивость позиционирования в сложных условиях (лес, городские высокие застройки). Для обработки данных GPS применяется библиотека TinyGPS++. Чтобы уменьшить влияние случайных колебаний сигнала, используется сглаживание координат методом усреднения, что позволяет повысить стабильность отображаемого положения.

- Отображение текущих координат, скорости, количества спутников.
- Функция «Вернуться к точке»: при активации автоматически сохраняет текущие координаты и далее в реальном времени отображает азимут (направление) и расстояние до этой точки.
- Загрузка и отображение заранее подготовленных маршрутов с расчетом дистанции.

3. Набор навигационных инструментов: 3D-компас, построение профиля высот, трекер положения солнца, калькулятор площади участка по точкам.
4. Мониторинг окружающей среды: Отслеживание температуры, давления, влажности с помощью датчика BME280, построение истории температуры, простой прогноз погоды на основе изменения давления.
5. Комплексная система безопасности:
   - Тревожная кнопка SOS с двумя уровнями активации (мигающий экран и прерывистый звуковой/вибросигнал).
   - Автоматическое логирование координат при активации SOS.
   - Встроенные справочники по выживанию и оказанию первой помощи, адаптированные под каждый из 4-х режимов.
   - Экран с экстренными контактами.
6. Энергоэффективность: Контроль уровня заряда батареи, функция автоматического отключения дисплея при бездействии для экономии энергии.

ОБОСНОВАНИЕ ВЫБОРА ЭЛЕМЕНТНОЙ БАЗЫ

Выбор компонентов был обусловлен необходимостью обеспечения высокой надежности, функциональности и автономности устройства в полевых условиях.

1. Микроконтроллер Arduino Mega 2560: Выбран из-за его большого количества портов ввода/вывода (необходимых для подключения всех датчиков и модулей, подключения перспективных внешних датчиков в режиме «Pro» и «Group») и объема памяти, которого достаточно для работы сложного программного кода с большими массивами данных (маршруты, история измерений и т. д.).

2. Модуль GPS NEO-7M: обеспечивает точное определение местоположения, скорости и времени. Критически важен для всех навигационных функций.
3. Датчик BME280: комбинированный датчик (температура, давление, влажность). Выбран за высокую точность, низкое энергопотребление и наличие готовых библиотек для удобства разработки.
4. Модуль часов реального времени (RTC) на DS3231. Обеспечивает точное время, даже при отключении основного питания. Незаменим для ведения логов, прогноза погоды и астрономических расчетов (восход/закат).
5. Модуль карты памяти MicroSD. Используется для хранения маршрутов, логов координат и системных данных. Позволяет работать с большими объемами информации без использования ограниченных ресурсов встроенной памяти и памяти программного кода (EEPROM) Arduino Mega.
7. OLED-дисплей (0.96", 128x64, I2C): Выбран из-за высокой контрастности, низкого энергопотребления и компактных размеров. Интерфейс I2C экономит пины на плате.
8. Звуковой зуммер и вибромотор. Используются для тактильной и звуковой обратной связи, а также для подачи сигналов тревоги в режиме SOS.
9. Аккумулятор 9V 1200 mAh. Обеспечивает автономную работу устройства в течение многих часов.

СТРУКТУРНАЯ СХЕМА УСТРОЙСТВА

Устройство построено по модульному принципу. Центральным элементом является Arduino Mega 2560, к которому по шине I2C подключены датчик BME280, модуль RTC и OLED-дисплей. Модули GPS и MicroSD используют интерфейс SPI. Зуммер, вибромотор и кнопки управления подключены к цифровым и аналоговым выводам микроконтроллера. Питание всей системы осуществляется от литий-ионного аккумулятора.

АЛГОРИТМ РАБОТЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Программное обеспечение написано на C++ в среде Arduino IDE и построено на основе конечного автомата (state machine), что позволяет четко управлять многочисленными экранами и режимами работы.

Ключевые алгоритмические блоки:

1. Инициализация и главный цикл:
   - При включении происходит инициализация всех датчиков и модулей, загрузка пользовательских данных из EEPROM.
   - Основной цикл loop постоянно опрашивает состояние кнопок, обновляет данные с датчиков и GPS, обрабатывает команды и перерисовывает интерфейс.

2. Алгоритм обработки нажатий кнопок и навигации:
   - Реализована система «короткого» и «долгого» нажатия для доступа к разным функциям.
   - Для визуализации долгого нажатия используется индикатор прогресса (прогресс-бар).
   - Алгоритм обрабатывает продолжительность нажатия и меняет текущее состояние (экран) системы, что приводит к отрисовке соответствующего интерфейса.
3. Алгоритм работы системы SOS:
   - При активации, система, периодически (раз в 3 минуты), записывает текущие координаты на SD-карту.
   - В зависимости от выбранного уровня активируется мигание экрана (инверсия цветов), звуковой сигнал и вибрация по заданному паттерну (с помощью азбуки Морзе (…---…))
4. Алгоритмы навигации и расчетов:
   - Расчет расстояния и азимута до сохраненной точки реализован с использованием обратного алгоритма Винсенти (Vincenty's formulae), обеспечивающего высочайшую точность вычислений на поверхности эллипсоида WGS84. Для вспомогательных навигационных расчетов также используются функции библиотеки TinyGPSPlus.
   - Расчет площади участка по введенным точкам производится по методу Гаусса (формула площади многоугольника).
   - Прогноз погоды основан на анализе тенденции изменения атмосферного давления за последний час.
   - Расчет времени до заката производится по астрономическим формулам на основе координат и даты.

ВЫВОДЫ ПО ПРОДЕЛАННОЙ РАБОТЕ

В результате проведенной работы был успешно разработан и собран реализовать все запланированные функции, включая четыре адаптивных режима работы, систему навигации и мониторинга окружающей среды, а также комплексную систему безопасности с трехуровневым сигналом SOS.

Аппаратный комплекс демонстрирует стабильную работу: обеспечивается корректное чтение данных со всех датчиков, вывод информации на OLED-дисплей и запись данных на SD-карту. Разработанный пользовательский интерфейс с многоуровневым меню отличается

интуитивной понятностью и простотой использования. Устройство работает полностью автономно от аккумуляторной батареи.

Количественные результаты:

- Точность определения координат GPS: ±2.5 метра
- Точность измерения температуры: ±1°C
- Среднее количество одновременно принимаемых спутников: 8–10
- Время автономной работы: около 14 часов (при условии использования аккумулятора 1200 mAh и периодическом отключении дисплея).
- Вместимость маршрута: до 50 точек
- Время отклика интерфейса: менее 200 мс

Качественные характеристики:

- Универсальность применения в четырех различных сценариях
- Надежность работы при потере GPS-сигнала
- Практическая значимость для решения реальных задач туристов
- Инновационный подход к интеграции разносторонних функций

Таким образом, программная и аппаратная части устройства представляют собой законченный прототип. Важно отметить, что разработанный код является стабильной и функциональной основой, а не финальным продуктом. Его модульность и подробное документирование позволяют легко модифицировать систему, добавлять новые датчики и алгоритмы, что является главной целью на следующем этапе развития проекта - переходе от прототипа к серийному образцу. Основная новизна проекта заключается в реализации адаптивной логики работы, которая подстраивается под выбранную среду, а также в успешном объединении множества инструментов в единое компактное устройство.

НАПАРАВЛЕНИЕ ДЛЯ ДАЛЬНЕЙШЕГО РАЗВИТИЯ

Разработанное устройство обладает значительным потенциалом для масштабирования и адаптации под различные сценарии использования. Его модульная архитектура, как аппаратная, так и программная, позволяет реализовать следующие перспективные направления развития.

1. Pollux Navigator Group - Режим группового похода
   Данное направление превращает устройство из индивидуального гаджета в центральный элемент системы коллективной безопасности для туристических групп.

- Технологическая основа: для реализации потребуется интеграция недорогого радиомодуля, такого как HC-12, работающего на дальности до 1 км на открытой местности.
- Ключевые функции:
  - Групповой SOS и трекинг: при активации сигнала бедствия на одном устройстве все остальные участники группы немедленно получают тревожное оповещение с точными координатами пострадавшего.
  - Карта группы: на дисплее каждого участника в режиме реального времени отображаются условные значки других членов группы с указанием азимута и примерного расстояния до них, рассчитанного на основе разницы в координатах.
  - Виртуальный сбор группы: лидер группы может отправить сигнал «Собраться», после чего на всех устройствах будет подсвечено направление и расстояние до точки сбора (координаты лидера или заранее заданная точка).

2. Pollux Navigator Pro - Режим для профессиональных экспедиций
Это направление ориентировано на превращение устройства в профессиональный инструмент для сбора и анализа данных в полевых условиях научными экспедициями, геологами и экологами.
   - Технологическая основа: Акцент будет сделан на долговременную автономность и расширенный сбор данных через подключение внешних датчиков по шинам I2C, SPI или аналоговым входам.
   - Ключевые функции:
     - Расширенный метеомониторинг: Поддержка подключения внешних датчиков: анемометра (скорость ветра), флюгера (направление ветра), датчика УФ-индекса, pH-метра почвы и др. Все данные будут записываться на SD-карту с жесткой привязкой ко времени и координатам.
     - Геодезические функции: Введение улучшенных алгоритмов для точного вычисления площади сложных полигонов и построения детального профиля высот с расчетом общего перепада и крутизны склонов.
     - Режим полевого дневника: Возможность добавлять к точкам маршрута голосовые или текстовые метки (через внешнее подключение) для пометок, например: «Лагерь №1», «Образец породы», «Аномалия».

- Продвинутая система энергосбережения: Реализация режима «глубокого сна» для всех модулей с пробуждением по таймеру для фиксации координат, что позволит устройству работать в режиме трекера несколько дней от одного заряда.

3. Усовершенствование конструкции корпуса

Эволюция корпуса направлена на достижение максимальной надежности, эргономики и технологичности производства.

- Материал: для серийного производства планируется переход на ударопрочный ABS-пластик, обладающий высокой механической прочностью, стойкостью к истиранию и химикатам (кислотам, щелочам, солям), а также сохраняющий стабильность при температурах до +90°C.
- Конструкция:
  - Модульность: корпус останется разборным, с удобным доступом ко всем компонентам для ремонта и замены.
  - Герметичность: для защиты от влаги и пыли будет использована система каналов с резиновыми уплотнительными прокладками, что позволит достичь уровня защиты IP67.
  - Эргономика: форма будет доработана для еще более удобного хранения, носки на рюкзаке и использования в перчатках, с тактильно ощутимыми и крупными кнопками.
- Технология производства:
  - Прототипирование: на текущем этапе корпус печатается на 3D-принтере, что позволяет быстро и дешево тестировать новые версии, а в случае поломки - легко напечатать замену.
  - Серия: для финальной версии планируется переход на литье под давлением, что обеспечит высочайшую прочность, точность и герметичность, недостижимые при 3D-печати.

4. Оптимизация аппаратной платформы и пользовательского опыта

Эволюция аппаратной части является ключом к повышению доступности, надежности и удобства использования устройства.

Направления включают:

- Миниатюризация и модернизация вычислительного ядра: Переход с платформы Arduino Mega на более современный, компактный и энергоэффективный микроконтроллер (например, ESP32-S3 или STM32) позволит значительно уменьшить вес и габариты корпуса, увеличить

производительность и добавить новые интерфейсы связи (Wi-Fi/Bluetooth для режима «Group»).

- Совершенствование системы питания: Замена батареи 9В на литий-ионный аккумулятор с системой контроля заряда (BMS) повысит энергоемкость, безопасность и удобство зарядки через распространенные порты (USB-C).
- Апгрейд дисплея: Замена OLED-экрана на энергоэффективный монохромный LCD с подсветкой, аналогичный используемому в рациях, позволит значительно снизить энергопотребление, повысить читаемость при прямом солнечном свете и увеличить общую надежность устройства.
- Повышение защиты от воздействий: Реализация корпуса стандарта IP67 для защиты от пыли и влаги, а также использование более широкодиапазонных и ударостойких компонентов.

5. Разработка программного обеспечения и экосистемы

Создание вокруг устройства программной экосистемы значительно расширит его функционал и удобство.

- Мобильное приложение-компаньон: Разработка кроссплатформенного приложения для удобного планирования маршрутов, их загрузки на устройство, а также для углубленного анализа пройденных треков и собранных данных после похода.
- Веб-платформа: Создание онлайн-сервиса для обмена маршрутами, тематическими картами (например, картами природных рисков) и статистикой между пользователями сообщества.

6. Локализация и интернационализация

Для преодоления языкового барьера и расширения аудитории пользователей запланирована полная реализация многоязычного интерфейса.

- Двуязычная поддержка: Интерфейс будет переведен на русский и английский языки с возможностью переключения в настройках. Это решит две задачи:
  - Для российских пользователей: обеспечит комфорт и удобство использования, так как весь функционал будет доступен на родном языке.
  - Для международного продвижения: сделает устройство привлекательным для иностранных туристов, что поспособствует развитию туризма в России. Иностранные гости смогут уверенно пользоваться навигатором, что повысит их безопасность и удобство при путешествии по нашей стране.

- Расширение пула языков: В перспективе базу языков можно будет расширить за счет добавления других популярных языков (например, китайского, немецкого, испанского) для охвата более широкой международной аудитории.

7. Развитие программного обеспечения и алгоритмов
Развитие программной части направлено на превращение рабочего прототипа в оптимизированный, надежный и легко масштабируемый продукт. Ключевые задачи:

- Оптимизация использования ресурсов: Глубокая оптимизация потребления оперативной (RAM) и flash-памяти за счет агрессивного использования PROGMEM, замены строковых операций и оптимизации структур данных.
- Смена инструментария и рефакторинг: Переход с Arduino IDE на профессиональную платформу (PlatformIO) для улучшения управления зависимостями и процессом сборки. Рефакторинг кода в сторону объектно-ориентированного подхода с выделением классов (`NavigationManager`, `WeatherMonitor`, `SOSHandler`) повысит читаемость, упростит тестирование и дальнейшую разработку.
- Внедрение продвинутых алгоритмов: Реализация алгоритмов фильтрации (например, фильтр Калмана для данных GPS и акселерометра) для значительного повышения точности и сглаживания показаний.
- Улучшение отзывчивости интерфейса: Внедрение аппаратных прерываний для обработки кнопок вместо опроса (polling) сделает интерфейс мгновенно реагирующим на действия пользователя и более энергоэффективным.

Таким образом, проект «Pollux Navigator» является не просто законченным устройством, а гибкой платформой, открытой для модернизации и адаптации под самые разнообразные задачи, от обеспечения безопасности любителей туризма до помощи в работе профессионалам.

ДОРОЖНАЯ КАРТА

Этап 1: Проектирование и разработка архитектуры (Июнь, 1-2 недели)

- Содержание: была сформулирована основная идея устройства, проведен анализ существующих решений и составлено техническое задание. Определен полный перечень необходимых компонентов, спроектированы структурная и принципиальная электрические схемы.
- Результат: Готовый план реализации и закупка всех электронных комплектующих.

Этап 2: Сборка прототипа на макетной плате (Июнь, 3 неделя)

- Содержание: Выполнена пробная сборка устройства: каждый компонент был поочередно подключен и проверен на работоспособность. Все модули были соединены в единую систему на макетной плате.
- Результат: собранное аппаратное ядро устройства, готовое к программированию. (папка 1 «Сборка на макетной плате»)

Этап 3: Написание базового программного кода (Июнь, 4 неделя)

- содержание: Разработаны и отлажены низкоуровневые функции: инициализация датчиков (BME280, RTC, GPS), работа с дисплеем и считывание показаний с кнопок. Создан главный экран интерфейса и реализовано базовое переключение между экранами.
- Результат: Устройство начало корректно отображать данные со всех сенсоров и реагировать на нажатия кнопок. (папка 2 «начальный код»)

Этап 4: Реализация основных алгоритмов и функций (Июль, 1-2 недели)

- Содержание: Разработаны и интегрированы ключевые алгоритмы: расчет расстояния и азимута для функции возврата с использованием библиотеки TinyGPSPlus, работа с файловой системой на SD-карте (сохранение/загрузка маршрутов), реализация системы меню.
- Результат: Написан код для режимов SOS с звуковой и вибро-индикацией. Реализована основная функциональность: навигация, мониторинг, безопасность. (папка 2 «код включающий основные функции»)

Этап 5: Разработка дополнительных инструментов (Июль, 3 неделя)

- Содержание: Создан раздел навигационных инструментов: компас, трекер солнца, калькулятор площади. Разработаны алгоритмы для прогноза погоды на основе анализа изменения атмосферного давления. Интегрированы справочные материалы (гиды по выживанию, первая помощь, телефоны специальных служб).
- Результат: Полностью собранное программное обеспечение со всем запланированным функционалом.

Этап 6: Создание печатной платы и чертежей (Июль, 4 неделя - Август, 1 неделя)

- Содержание: Разработана схема и разводка печатной платы в приложении Sprint-Layout (папка 3 «Печатная плата»). Изготовлена печатная плата методом ЛУТ и сверление с помощью сверлильного станка, выполнена пайка всех компонентов (папка 3 «изготовление платы»). Созданы профессиональные чертежи платы и компонентов в приложении nanoCAD. (папка 5 «Чертежи платы и компонентов»)

- результат: Устройство собрано на профессиональной печатной плате (папка 4 «Собранная плата»).

Этап 7: Проектирование корпуса и полевые испытания (Август, 2-3 недели)

- Содержание: Разработана 3D-модель корпуса в приложении Moi3D на основе чертежей, напечатан пробный образец на 3D-принтере. Проведены полевые испытания в условиях реального похода: протестирован функционал устройства и эргономика корпуса. Испытания проводились во время похода, который проходил в Талдомском районе Московской области, вблизи СНТ «Вербилки». Были проведены испытания на пересеченной местности в лесу и в поле. (папка 7 «Тестирование в условиях реального похода»)
- результат: Выявлены недочеты в программном коде, недостатки корпуса, которые были исправлены в дальнейшем.

Этап 8: Финальная доработка проекта (Август, 4 неделя - Сентябрь, 1 неделя)

- Содержание: Внесены правки в программный код по итогам испытаний. Модифицирована 3D-модель корпуса, спроектирован и напечатан окончательный вариант. Выполнена сборка готового устройства. Написана и оформлена инструкция для пользователей. (папка 8 «Инструкция для пользователя Pollux Navigator»)
- результат: Получено готовое к использованию устройство. (папка 2 «Финальный код») (папка 6 «Финальный корпус)

Этап 9: Подготовка документации (Сентябрь, 2-3 недели)

- Содержание: Написана пояснительная записка, оформлен отчет по методическим рекомендациям. Создана презентация и видеодемонстрация работы устройства. (папка 7 «Видео работы»)
- Результат: Подготовлен полный пакет документов для подачи на конкурс.
- Анализ полученных результатов

В ходе тестирования рабочего прототипа устройства «Pollux Navigator» были получены следующие результаты.

1. Аппаратная часть и инициализация
    - Время полной инициализации всех модулей системы (OLED, BME280, RTC, SD-карта, GPS) от момента подачи питания до перехода в рабочий режим составило 8-15 секунд.
    - Статус инициализации компонентов при 20 запусках:
        - OLED-дисплей: 20/20 успешно
        - Датчик BME280: 20/20 успешно

- Модуль RTC: 20/20 успешно
- Модуль SD-карты: 18/20 успешно (в 2х случаях потребовалась повторная инициализация)
- GPS-модуль: 17/20 успешно (валидные данные получены в течение 30 секунд в 17случаях, в остальных случаях потребовалось от 30 до 60 секунд)

2. Работа GPS-модуля

- Средняя точность определения координат в условиях открытой местности: ±2.5 метра.
- Время до первого обновления координат после холодного старта: 25-45 секунд.
- Количество отслеживаемых спутников: 7-12 единиц.
- Время определения потери спутникового сигнала: 10 секунд.
- Скорость обновления координат в режиме непрерывного приема: 1 раз в секунду.

Таблица 2. Характеристики GPS-модуля в различных условиях

Условия тестирования	Точность	Количество спутников	TTFF
Открытая местность	±2.5 м	10-12	25 с
Лесной массив	±5.8 м	8-10	38 с

3. Энергопотребление и автономность

- Напряжение полностью заряженной аккумуляторной батареи: 8.4 В.
- Потребляемый ток в различных режимах:
  - Режим ожидания: 85 мА
  - Активный режим: 180-195 мА
  - Режим SOS: 220-235 мА
- Время автономной работы при отображении главного экрана: 14.1 ч.
- Напряжение, при котором система отображает 30% заряда: ~ 7.2 В.
- Полный разряд (0%) при напряжении ~ 6.4 В.
- Реализован режим энергосбережения, который активируется после 5 минут бездействия пользователя. В этом режиме отключается дисплей и периферийные модули, что значительно снижает потребление тока.

Таблица 3. Энергопотребление и автономность

Режим работы	Потребляемый ток (мА)	Время работы (часы)
Режим энергосбережения	85	14.1
Активный режим	190	6.3
Режим SOS	230	5.2
Смешанный режим (активный режим и режим энергосбережения)	108	14.1

4. Работа датчиков окружающей среды
   - Диапазон измерения температуры: -15°C до +45°C.
   - Разрешение измерения температуры: 0.1°C.
   - Диапазон измерения атмосферного давления: от 300 до 1100 гПа.
   - Разрешение измерения давления: 1 гПа.
   - Погрешность измерений температуры: ±0.5°C в диапазоне от -30°C до +40°C.
   - Интервал записи данных в историю температуры: 60 минут.

5 Функционал навигации
   - Точность определения расстояния до сохраненной точки: ±100 метров на дистанции 1 км.
   - Время расчета азимута: менее 500 мс.
   - Максимальное количество точек в маршруте: 50 точек.
   - Объем памяти, занимаемый одним маршрутом: 0.8-1.2 КБ.

6. Работа аварийного режима SOS
   - Время активации режима SOS: 1 с.
   - Интервал мигания дисплея: 500 мс.
   - Частота звукового сигнала: 400 Гц.
   - Паттерн звукового сигнала:
   - 200 мс включено / 200 мс выключено (в течение 2-секундного цикла)
   - Полный цикл повторяется каждые 2 секунды
   - Тактильная сигнализация: Вибромотор работает синхронно со звуковым сигналом
   - Потребляемый ток в режиме SOS: 220-235 мА
   - Дополнительные функции:
   - Автоматическое логирование координат в файл SOS.LOG каждые 3 минуты
   - Инверсия цветов дисплея для повышенной заметности

7. Работа пользовательского интерфейса
   - Время отклика на нажатие кнопок: менее 200 мс.
   - Время переключения между экранами: менее 300 мс.
   - Количество одновременно отображаемых строк текста: 4-5 строк.
   - Скорость прокрутки текста: 1 строка каждые 1.5 секунды.
8. Механические испытания корпуса
   - Высота падения без повреждений: 1.2 метра на бетонную поверхность.
   - Рабочая температура: -30°C до +50°C.
   - Водозащищенность: устойчивость к брызгам и дождю.

Таблица 4. Результаты механических испытаний

Тест	Условия	Результат
Падение	1.2 м на бетон	Без повреждений
Температура	-30°C до +50°C	Стабильная работа
Влажность	80% при +35°C	Корректная работа

9. Память и данные
   - Объем свободной оперативной памяти: Скетч использует 80418 байт (31%) памяти устройства после инициализации всех модулей.
   - Объем памяти EEPROM для хранения пользовательских данных: 1024 байта.
   - Объем памяти SD карты: от 8 до 32 ГБ (в зависимости от используемой пользователем SD карты)
     - На текущем этапе Pollux Navigator работает без встроенного картографического движка: навигация основана на координатах (отображение азимута и расстояния до точки) и загрузке заранее подготовленных маршрутов в формате GPX, хранение данных - на microSD. Это уменьшает требования к памяти и позволяет обеспечивать оффлайн-работу. Для будущих версий планируется внедрить поддержку офлайн-векторных тайлов (например, MBTiles) и растровых карт (PNG/XYZ), которые можно будет хранить на microSD и отображать в упрощённом виде на OLED/монохромном дисплее (потом - на энергоэффективном монохромном LCD).
   - Среднее время записи точки маршрута на SD-карту: 120 мс.

Все измерения проводились в условиях, приближенных к реальным условиям эксплуатации устройства.

ЗАКЛЮЧЕНИЕ

В результате проделанной работы был создан и протестирован рабочий прототип автономного навигационно-аварийного комплекса «Pollux Navigator».

1. Достижение поставленных целей

   Главная цель проекта - создание универсального устройства, объединяющего навигацию, мониторинг среды и функции безопасности - была успешно достигнута. Мне удалось реализовать все ключевые задачи:

   - Собрать аппаратную платформу на Arduino Mega 2560 с интеграцией семи периферийных модулей
   - Разработать комплексное ПО с реализацией многоуровневого меню, адаптивных режимов работы и всех заявленных алгоритмов
   - Внедрить трехуровневую систему безопасности с функцией оповещения и локальным хранением данных
   - Создать эргономичный интерфейс, ориентированный на использование в стрессовых ситуациях
   - Провести комплексные испытания в реальной среде, подтвердившие работоспособность и надежность устройства

2. Решение актуальных проблем

   Разработка решает проблемы современных outdoor-гаджетов:

   - Интеграция разрозненных функций в единое устройство
   - Преобразование сырых данных в практические рекомендации
   - Полная автономность работы без зависимости от смартфона
   - Интуитивный интерфейс, доступный в критических ситуациях

3. Конкурентные преимущества модели «Pollux Navigator»

   На мой взгляд, данная модель принципиально отличается от существующих аналогов, так как имеет:

- Расширенный функционал за счет адаптивной аналитики и обучающего модуля
- Открытую архитектуру, позволяющую модифицировать и адаптировать устройство под специфические задачи
- Значительное экономическое преимущество в стоимости при сопоставимой функциональности

4. Практические результаты

   Испытания подтвердили жизнеспособность концепции:

   - Точность GPS: ±2.5 м
   - Точность метеоданных: ±0.5°C

- Рабочий температурный диапазон: -15°C до +50°C
- Автономность: до 14 часов в смешанном режиме
- Механическая устойчивость: падение с 1.2 м

Разработанный мной комплекс представляет собой законченное решение, готовое к практическому применению. Проект демонстрирует возможность создания доступного многофункционального устройства для активного туризма с открытой архитектурой и адаптивными функциями. Полученные результаты подтверждают перспективность дальнейшего развития проекта в контексте импортозамещения и развития отечественных решений для outdoor-индустрии.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Аппаратная платформа и микроконтроллеры
2 Мартюшев, А.А. Программирование микроконтроллеров AVR: от Arduino к ассемблеру https://djvu.online/file/oQEDc0eZ0DvOF - Использована для глубокого понимания архитектуры ATmega2560 и оптимизации работы с периферией.
3. Evans, M. Arduino in Action / M. Evans, J. Noble, J. Hochenbaum. https://archive.org/details/arduinoinaction0000evan - Применена при разработке общей структуры проекта и управления пинами Arduino Mega 2560.
4. Документация на микроконтроллер ATmega2560 https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf - Использована для работы с регистрами ввода-вывода, таймерами и прерываниями.
5. Спутниковая навигация и GPS/ГЛОНАСС
6. Глобальная спутниковая система ГЛОНАСС. Интерфейсный контрольный документ (ИКД) https://docs.yandex.ru/docs/view?tm=1757350316&tld=ru&lang=ru&name=ИКД._Общ._опис._(ред.1%2C_20.09.13).pdf&text=Глобальная%20спутниковая%20система%20навигации%20ГЛОНАСС.%20Интерфейсный%20контрольный%20документ%20(ИКД)&url=https%3A%2F%2Fwiki.srns.ru%2Fimages%2Fb%2Fb4%2F%25D0%2598%25D0%259A%25D0%2594._%25D0%259E%25D0%25B1%25D1%2589._%25D0%25BE%25D0%25BF%25D0%25B8%25D1%2581._(%25D1%2580%25D0%25B5%25D0%25B4.1%2C_20.09.13).pdf&lr=213&mime=pdf&l10n=ru&sign=ae639e475abe8907640314190d85295d&keyno=0&nosw=1&serpParams=tm%3D1757350316%26tld%3Dru%26lang%3Dru%26name%3D%25D0%2598%25D0%259A%25D0%2594._%25D0%259E%25D0%25B1%25D1%2589._%25D0%25BE%25D0%25BF%25D0%25B8%25D1%2581._(%25D1%2580%25D0%25B5%25D0%25B4.1%2C_20.09.13).pdf%26text%3D%25D0%2593%25D0%25BB%25D0%25BE%25D0%25B1%25D0%25B0%25D0%25BB%25D1%258C%25D0%25BD%25D0%25B0%25D1%258F%2B%25D1%2581%25D0%25BF%25D1%2583%25D1%2582%25D0%25BD%25D0%25B8%25D0%25BA%25D0%25BE%25D0%25B2%25D0%25B0%25D1%258F%2B%25D1%2581%25D0%25B8%25D1%2581%25D1%2582%25D0%25B5%25D0%25BC%25D0%25B0%2B%25D0%25BD%25D0%25B0%25D0%25B2%25D0%25B8%25D0%25B3%25D0%25B0%25D1%2586%25D0%25B8%25D0%25B8%2B%25D0%2593%25D0%259B%25D0%259E%25D0%259D%25D0%2590%25D0%25A1%25D0%25A1.%2B%25D0%2598%25D0%25BD%25D1%2582%25D0%25B5%25D1%2580%25D1%2584%25D0%25

B5%25D0%25B9%25D1%2581%25D0%25BD%25D1%258B%25D0%25B9%2B%25D0%25BA%25D0%25BE%25D0%25BD%25D1%2582%25D1%2580%25D0%25BE%25D0%25BB%25D1%258C%25D0%25BD%25D1%258B%25D0%25B9%2B%25D0%25B4%25D0%25BE%25D0%25BA%25D1%2583%25D0%25BC%25D0%25B5%25D0%25BD%25D1%2582%2B%2528%25D0%2598%25D0%259A%25D0%2594%2529%26url%3Dhttps%253A%2F%2Fwiki.srns.ru%2Fimages%2Fb%2Fb4%2F%2525D0%252598%2525D0%25259A%2525D0%252594._%2525D0%25259E%2525D0%2525B1%2525D1%252589._%2525D0%2525BE%2525D0%2525BF%2525D0%2525B8%2525D1%252581._%2528%2525D1%252580%2525D0%2525B5%2525D0%2525B4.1%252C_20.09.13%2529.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3Dae639e475abe8907640314190d85295d%26keyno%3D0%26nosw%3D1 -Использована для понимания формата данных и планирования поддержки ГЛОНАСС в будущих версиях.

7. Hofmann-Wellenhof, B. Global Positioning System: Theory and Practice https://docs.yandex.ru/docs/view?tm=1757350246&tld=ru&lang=en&name=global-positioning-system-theory-and-practice-1992.pdf&text=Hofmann-Wellenhof%2C%20B.%20Global%20Positioning%20System%3A%20Theory%20and%20Practice%20%2F%20B.%20Hofmann-Wellenhof%2C%20H.%20Lichtenegger%2C%20J.%20Collins.&url=https%3A%2F%2Fnguyenduyliemgis.wordpress.com%2Fwp-content%2Fuploads%2F2014%2F09%2Fglobal-positioning-system-theory-and-practice-1992.pdf&lr=213&mime=pdf&l10n=ru&sign=d8d20f39e8343738932147a0367a8359&keyno=0&nosw=1&serpParams=tm%3D1757350246%26tld%3Dru%26lang%3Den%26name%3Dglobal-positioning-system-theory-and-practice-1992.pdf%26text%3DHofmann-Wellenhof%252C%2BB.%2BGlobal%2BPositioning%2BSystem%253A%2BTheory%2Band%2BPractice%2B%2F%2BB.%2BHofmann-Wellenhof%252C%2BH.%2BLichtenegger%252C%2BJ.%2BCollins.%26url%3Dhttps%253A%2F%2Fnguyenduyliemgis.wordpress.com%2Fwp-content%2Fuploads%2F2014%2F09%2Fglobal-positioning-system-theory-and-practice-1992.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3Dd8d20f39e8343738932147a0367a8359%26keyno%3D0%26nosw%3D1 . - Применена для понимания принципов работы GPS, вычисления координат и разбора NMEA-протокола.

8. Документация на модуль u-blox NEO-7M - Режим доступа: https://www.u-blox.com/en/product/neo-7-series - Использована для настройки модуля, чтения сырых данных и управления режимами энергосбережения.

9. Геодезические и навигационные алгоритмы
10. Vincenty, T. Direct and Inverse Solutions of Geodesics on the Ellipsoid with application of nested equations. T. Vincenty https://docs.yandex.ru/docs/view?tm=1757350123&tld=ru&lang=en&name=inverse.pdf&text=Vincenty%2C%20T.%20Direct%20and%20Inverse%20Solutions%20of%20Geodesics%20on%20the%20Ellipsoid%20with%20application%20of%20nested%20equations%20%2F%20T.%20Vincenty%20%2F%2F%20Survey%20Review.%20-%201975&url=https%3A%2F%2Fwww.cqsrg.org%2Ftools%2FGCDistance%2Finverse.pdf&lr=213&mime=pdf&l10n=ru&sign=68bd3e07fc4dd992db7655eb913e69e6&keyno=0&nosw=1&serpParams=tm%3D1757350123%26tld%3Dru%26lang%3Den%26name%3Dinverse.pdf%26text%3DVincenty%252C%2BT.%2BDirect%2Band%2BInverse%2BSolutions%2Bof%2BGeodesics%2Bon%2Bthe%2BEllipsoid%2Bwith%2Bapplication%2Bof%2Bnested%2Bequations%2B%2F%2BT.%2BVincenty%2B%2F%2F%2BSurvey%2BReview.%2B%25E2%2580%2594%2B1975%26url%3Dhttps%253A%2F%2Fwww.cqsrg.org%2Ftools%2FGCDistance%2Finverse.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3D68bd3e07fc4dd992db7655eb913e69e6%26keyno%3D0%26nosw%3D1 - Алгоритм Винсенти реализован в функции `vincentyDistance()` для точного расчета расстояния на поверхности эллипсоида WGS84.
11. Современная геодезия и навигация / под ред. В.Д. Большакова, Г.Н. Ганьшина. - М.: Недра, https://djvu.online/file/tP1FqTMuLB86C - Общая информация и Метод Гаусса для расчета площади (формула шнурования) реализован в функциях `calculateAreaGauss()` и `calculateAreaAccurate()`.
12. National Geospatial-Intelligence Agency (NGA). World Geodetic System 1984 (WGS 84). Its Definition and Relationships with Local Geodetic Systems https://ahrs.readthedocs.io/en/stable/wgs84.html - Использованы параметры эллипсоида WGS84 (константы `WGS84_A`, `WGS84_B`, `WGS84_F`) для всех геодезических расчетов.
13. Программирование, алгоритмы и структуры данных
14. Керниган, Б.У. Язык программирования С / Б.У. Керниган, Д.М. Ричи. - 2-е изд. https://docs.yandex.ru/docs/view?tm=1757349773&tld=ru&lang=ru&name=kernigan.pdf&text=Керниган%2С%20Б.У.%20Язык%20программирования%20С%20%2F%20Б.У.%20Керниган%2С%20Д.М.%20Ричи.%20-%202-е%20изд.&url=https%3A%2F%2Fcxo.lv%2Fimages%2Fstories%2FBooks%2Funsorted%2Fkernigan.pdf&lr=213&mime=pdf&l10n=ru&sign=2c4e9fe33c65591320a8336911a535e1&keyno=0&nosw=1&serpParams=tm%3D1757349773%26tld%3Dru%26lang%3Dr

u%26name%3Dkernigan.pdf%26text%3D%25D0%259A%25D0%25B5%25D1%2580%25D0%25BD%25D0%25B8%25D0%25B3%25D0%25B0%25D0%25BD%252C%2B%25D0%2591.%25D0%25A3.%2B%25D0%25AF%25D0%25B7%25D1%258B%25D0%25BA%2B%25D0%25BF%25D1%2580%25D0%25BE%25D0%25B3%25D1%2580%25D0%25B0%25D0%25BC%25D0%25BC%25D0%25B8%25D1%2580%25D0%25BE%25D0%25B2%25D0%25B0%25D0%25BD%25D0%25B8%25D1%258F%2BC%2B%2F%2B%25D0%2591.%25D0%25A3.%2B%25D0%259A%25D0%25B5%25D1%2580%25D0%25BD%25D0%25B8%25D0%25B3%25D0%25B0%25D0%25BD%252C%2B%25D0%2594.%25D0%259C.%2B%25D0%25A0%25D0%25B8%25D1%2587%25D0%25B8.%2B%25E2%2580%2594%2B2-%25D0%25B5%2B%25D0%25B8%25D0%25B7%25D0%25B4.%26url%3Dhttps%253A%2F%2Fcxo.lv%2Fimages%2Fstories%2FBooks%2Funsorted%2Fkernigan.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3D2c4e9fe33c65591320a8336911a535e1%26keyno%3D0%26nosw%3D1 - Является основой для написания всего программного кода на C++ в среде Arduino IDE.

15. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. https://docs.yandex.ru/docs/view?tm=1757349602&tld=ru&lang=ru&name=Priemy_obektno-orientirovannogo_proektirovania_Biblioteka_prog.pdf&text=Гамма%2C%20Э.%20Приемы%20объектно-ориентированного%20проектирования&url=https%3A%2F%2Favmim.com%2Fwp-content%2Fuploads%2F2018%2F09%2FPriemy_obektno-orientirovannogo_proektirovania_Biblioteka_prog.pdf&lr=213&mime=pdf&l10n=ru&sign=e209991da323069a3d136a575b05e648&keyno=0&nosw=1&serpParams=tm%3D1757349602%26tld%3Dru%26lang%3Dru%26name%3DPriemy_obektno-orientirovannogo_proektirovania_Biblioteka_prog.pdf%26text%3D%25D0%2593%25D0%25B0%25D0%25BC%25D0%25BC%25D0%25B0%252C%2B%25D0%25AD.%2B%25D0%259F%25D1%2580%25D0%25B8%25D0%25B5%25D0%25BC%25D1%258B%2B%25D0%25BE%25D0%25B1%25D1%258A%25D0%25B5%25D0%25BA%25D1%2582%25D0%25BD%25D0%25BE-%25D0%25BE%25D1%2580%25D0%25B8%25D0%25B5%25D0%25BD%25D1%2582%25D0%25B8%25D1%2580%25D0%25BE%25D0%25B2%25D0%25B0%25D0%25BD%25D0%25BD%25D0%25BE%25D0%25B3%25D0%25BE%2B%25D0%25BF%25D1%2580%25D0%25BE%25D0%25B5%25D0%25BA%25D1%2582%25D0%25B8%

25D1%2580%25D0%25BE%25D0%25B2%25D0%25B0%25D0%25BD%25D0%25B8%25D1%258F%26url%3Dhttps%253A%2F%2Favmim.com%2Fwp-content%2Fuploads%2F2018%2F09%2FPriemy_obektno-orientirovannogo_proektirovania_Biblioteka_prog.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3De209991da323069a3d136a575b05e648%26keyno%3D0%26nosw%3D1 - Архитектура программного обеспечения построена на основе паттерна «Конечный автомат» (State Machine), что отражено в перечислении `ScreenStates`.

16. Страуструп, Б. Язык программирования C++ https://docs.yandex.ru/docs/view?tm=1757349476&tld=ru&lang=ru&name=Straustrup-Yazyk_programmirovaniya_c.pdf&text=Страуструп%2C%20Б.%20Язык%20программирования%20C%2B%2B&url=http%3A%2F%2Fwww.8361.ru%2F6sem%2Fbooks%2FStraustrup-Yazyk_programmirovaniya_c.pdf&lr=213&mime=pdf&l10n=ru&sign=12477c6e0b5f9697f9423e4fde1634c0&keyno=0&nosw=1&serpParams=tm%3D1757349476%26tld%3Dru%26lang%3Dru%26name%3DStraustrup-Yazyk_programmirovaniya_c.pdf%26text%3D%25D0%25A1%25D1%2582%25D1%2580%25D0%25B0%25D1%2583%25D1%2581%25D1%2582%25D1%2580%25D1%2583%25D0%25BF%252C%2B%25D0%2591.%2B%25D0%25AF%25D0%25B7%25D1%258B%25D0%25BA%2B%25D0%25BF%25D1%2580%25D0%25BE%25D0%25B3%25D1%2580%25D0%25B0%25D0%25BC%25D0%25BC%25D0%25B8%25D1%2580%25D0%25BE%25D0%25B2%25D0%25B0%25D0%25BD%25D0%25B8%25D1%258F%2BC%252B%252B%26url%3Dhttp%253A%2F%2Fwww.8361.ru%2F6sem%2Fbooks%2FStraustrup-Yazyk_programmirovaniya_c.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3D12477c6e0b5f9697f9423e4fde1634c0%26keyno%3D0%26nosw%3D1.
- Использован для реализации сложных структур данных (например, `struct CalibrationData`) и объектно-ориентированного стиля.
17. Датчики и метрология
18. Bosch Sensortec. BME280 Digital Humidity, Pressure and Temperature Sensor https://docs.yandex.ru/docs/view?tm=1757349368&tld=ru&lang=en&name=3ba8b9a547c432e79276186dd829340c.pdf&text=%3A%20https%3A%2F%2Fwww.bosch-sensortec.com%2Fmedia%2Fboschsensortec%2Fdownloads%2Fdatasheets%2Fbst-bme280-ds002.pdf&url=https%3A%2F%2Froboparts.ru%2Fupload%2Fiblock%2F3ba%2F3ba8b

9a547c432e79276186dd829340c.pdf&lr=213&mime=pdf&l10n=ru&sign=11ba8120ce7832d2c2111c666d1391c1&keyno=0&nosw=1&serpParams=tm%3D1757349368%26tld%3Dru%26lang%3Den%26name%3D3ba8b9a547c432e79276186dd829340c.pdf%26text%3D%253A%2Bhttps%253A%2F%2Fwww.bosch-sensortec.com%2Fmedia%2Fboschsensortec%2Fdownloads%2Fdatasheets%2Fbst-bme280-ds002.pdf%26url%3Dhttps%253A%2F%2Froboparts.ru%2Fupload%2Fiblock%2F3ba%2F3ba8b9a547c432e79276186dd829340c.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3D11ba8120ce7832d2c2111c666d1391c1%26keyno%3D0%26nosw%3D1 - Использована для калибровки датчика, чтения данных и настройки режимов работы (функция `initBME280WithRetry()`).

19. Maxim Integrated. DS3231 Extremely Accurate I²C-Integrated RTC/TCXO/Crystal https://docs.yandex.ru/docs/view?tm=1757349310&tld=ru&lang=en&name=DS3231-DS3231S.pdf&text=Maxim%20Integrated.%20DS3231%20Extremely%20Accurate%20I²C-Integrated%20RTC%2FTCXO%2FCrystal&url=http%3A%2F%2Fwww.gaw.ru%2Fpdf%2FMaxim%2Frtc%2FDS3231-DS3231S.pdf&lr=213&mime=pdf&l10n=ru&sign=6d93f7c31f7ff91771499b7c810455a8&keyno=0&nosw=1&serpParams=tm%3D1757349310%26tld%3Dru%26lang%3Den%26name%3DDS3231-DS3231S.pdf%26text%3DMaxim%2BIntegrated.%2BDS3231%2BExtremely%2BAccurate%2BI%25C2%25B2C-Integrated%2BRTC%2FTCXO%2FCrystal%26url%3Dhttp%253A%2F%2Fwww.gaw.ru%2Fpdf%2FMaxim%2Frtc%2FDS3231-DS3231S.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3D6d93f7c31f7ff91771499b7c810455a8%26keyno%3D0%26nosw%3D1 . - Использована для работы с часами реального времени и обеспечения точного времени для логирования.
20. Безопасность, выживание и оказание первой помощи
21. ГОСТ Р 22.3.07-2014. Безопасность в чрезвычайных ситуациях. Оказание первой помощи пострадавшим в чрезвычайных ситуациях. Общие требования https://lic8-krasnoyarsk-r04.gosweb.gosuslugi.ru/netcat_files/32/50/GOST_R_22.3.07_2014_Bezopasnost_v_chrezvychaynyh_situatsiyah._Kul_tura_bezopasnosti_zhiznedeyatel_nosti.pdf - Тексты

руководств по первой помощи (массивы `firstAidGuides0[]` и т.д.) составлены в соответствии с данным стандартом.

22. ГОСТ Р 22.3.06-2020. Безопасность в чрезвычайных ситуациях. Средства обеспечения безопасности в горах. Общие технические требования. https://docs.yandex.ru/docs/view?tm=1757349091&tld=ru&lang=ru&name=4294847155.pdf&text=ГОСТ%20Р%2022.3.06&url=https%3A%2F%2Fohranatruda.ru%2Fupload%2Fiblock%2Fbfc%2F4294847155.pdf&lr=213&mime=pdf&l10n=ru&sign=44b36ccf55ce7713f8611b6be018b60d&keyno=0&serpParams=tm%3D1757349091%26tld%3Dru%26lang%3Dru%26name%3D4294847155.pdf%26text%3D%25D0%2593%25D0%259E%25D0%25A1%25D0%25A2%2B%25D0%25A0%2B22.3.06%26url%3Dhttps%253A%2F%2Fohranatruda.ru%2Fupload%2Fiblock%2Fbfc%2F4294847155.pdf%26lr%3D213%26mime%3Dpdf%26l10n%3Dru%26sign%3D44b36ccf55ce7713f8611b6be018b60d%26keyno%3D0 - Рекомендации по выживанию в горной местности (`survivalGuides0[]`, `survivalGuides1[]`) базируются на положениях данного стандарта.
23. International Commission for Alpine Rescue (ICAR). ICAR Guidelines https://www.alpine-rescue.org - Использованы международные рекомендации по оценке лавинной опасности и организации спасательных работ.
24. Пользовательский интерфейс и энергоэффективность
25. Adafruit Industries. SSD1306 OLED Driver: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-gfx-graphics-library.pdf
    - Библиотека `Adafruit_SSD1306.h` использована для управления дисплеем и отрисовки многоуровневого интерфейса.
26. Химмельблау.Д прикладное нелинейное программирование https://djvu.online/file/tTm5cnA49IVXa 1975. - Алгоритмы фильтрации данных GPS (экспоненциальное сглаживание в функции `cacheGPS()`) основаны на принципах, изложенных в данной литературе.
27. Документация платформы Arduino - Режим доступа: https://www.arduino.cc/reference/en/ - Использована для работы со стандартными библиотеками (`EEPROM.h`, `SD.h`, `SPI.h`, `Wire.h`) и базовыми функциями.

ПРИЛОЖЕНИЕ

Приложения расположены и удобно структурированы на Яндекс диске:

https://disk.yandex.ru/d/AbPEnFI76WMRTw

СБОРКА И ИСПЫТАНИЕ СОБСТВЕННОЙ МАГНИТНО-ИМПУЛЬСНОЙ УСТАНОВКИ ДЛЯ ЭЛЕКТРОМАГНИТНОЙ ФОРМОВКИ

Соловьев Артемий Андреевич

Москва г, ГБОУ УР «Лицей №41», 10 класс

Секция: Машиностроительные технологии

Научный руководитель: Бабичев Сергей Анатольевич, учитель физики ГБОУ УР «Лицей №41»

АННОТАЦИЯ

Цель — создать действующую установку для демонстрации преимуществ электромагнитной формовки и подтверждения её возможностей на практике.

Метод формовки — магнитно-импульсная формовка.

Экспериментально-аналитический метод работы над проектом состоит из следующих основных этапов:

Конструирование — разработка электрической схемы и компоновка узлов установки.

Сборка— монтаж высоковольтных цепей, намотка катушек, изготовление защитных кожухов.

Визуальный контроль — оценка целостности элементов установки и качества деформации заготовок после каждого разряда.

Проведение экспериментов.

Результаты: создана действующая магнитно-импульсная установка с энергией разряда до 10000 Дж (8 конденсаторов, 200 мкФ, 10 кВ). Разработана схема поджига игнитрона ИРТ-6 с зазором 0,4 мм. Экспериментально подтверждены четыре режима обработки: сжатие, расширение, штамповка и соединение разнородных металлов (алюминий + сталь).

Выводы: Созданная установка позволяет деформировать трубки разного диаметра и листовые заготовки без нагрева, демонстрируя преимущества перед традиционными методами. Экспериментально подтвержденные режимы обработки показывают перспективность метода для решения реальных технологических задач.

ВВЕДЕНИЕ

Машиностроение и обрабатывающая промышленность нуждаются в технологиях, позволяющих соединять и деформировать металлические заготовки с высокой точностью, без потери прочности и термического воздействия на материал. Традиционные методы, такие как сварка, пайка, механическая штамповка, обладают недостатками: при нагреве изменяется микроструктура металла, возникают внутренние напряжения и окалина, в результате чего требуется дополнительная механическая обработка, что снижает долговечность изделий.

Так же, сварка разнородных материалов, например алюминия и стали, затруднена и требует сложных технологических решений. Альтернативой является метод магнитно-импульсной формовки — холодной бесконтактной деформации металлов под действием импульсного электромагнитного поля. Предлагаемый мной метод лишен недостатков традиционных методов формовки. Ключевые преимущества, формирующие актуальность данной работы:

- Отсутствие нагрева: структура металла не меняется, не образуются хрупкие зоны;
- Высокая точность и скорость: процесс занимает доли секунды;
- Соединение разнородных материалов: например, можно прочно соединить алюминий и сталь, что крайне востребованно в промышленности, но практически невозможно сделать обычной сваркой без использования промежуточных слоев или специальных технологических процессов.

Цель работы — создать действующую установку для демонстрации преимуществ магнитно-импульсной формовки и подтверждения её возможностей на практике.

Задачи:

1. Изучить теорию и проанализировать промышленные аналоги;
2. Разработать электрическую схему установки, обеспечивающую безопасность и надежность;
3. Собрать установку, внедряя решения собственной разработки;
4. Провести серию экспериментов;
5. Проанализировать результаты и наметить пути развития.

ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ И АНАЛИЗ МЕТОДОВ МАГНИТНО-ИМПУЛЬСНОЙ ФОРМОВКИ

АНАЛИЗ ТРАДИЦИОННЫХ МЕТОДОВ СОЕДИНЕНИЯ И ФОРМООБРАЗОВАНИЯ МЕТАЛЛОВ

Рассмотрим существующие методы, которые в настоящее время применяются в промышленности.

Сварка (Рис.1) является одним из самых распространенных методов соединения металлов. При всех своих достоинствах она обладает недостатком — термическим воздействием на материал. В месте сварки металл расплавляется, это ведет к изменению его микроструктуры. Возникают внутренние напряжения, может образовываться окалина – окисел на поверхности раскаленного металла, меняются механические свойства. Особенно эта проблема остро стоит при сварке разнородных металлов. Например, при попытке сварить алюминий со сталью образуются хрупкие соединения, которые делают шов недолговечным.

Рис. 1. Электродуговая сварка черного металла

Пайка (Рис.2) хотя и осуществляется при более низких температурах, чем сварка, все равно требует нагрева. Это может приводить к отпуску или отжигу основного металла, снижению его прочности в зоне соединения. Также, прочность паяного шва, обычно ниже прочности соединяемых деталей.

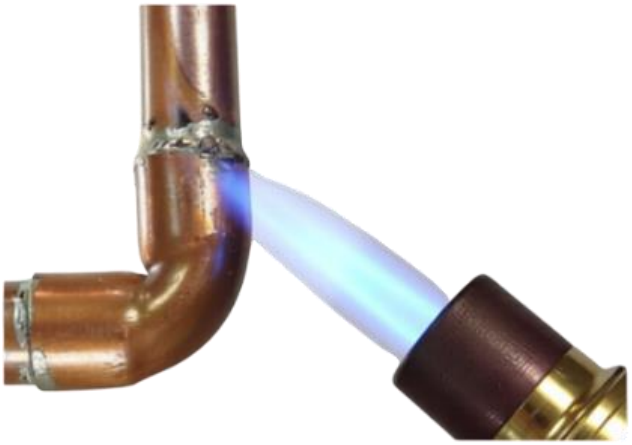


Рис. 2. Пайка меди

У механической и гидравлической штамповки также есть недостатки. Инструмент непосредственно взаимодействует с заготовкой, что приводит к его износу и требует периодической замены или восстановления оснастки. При штамповке сложных форм возникают проблемы с равномерностью деформации. Гидравлические прессы (Рис. 3) обладают

низкой энергоэффективностью при импульсных режимах работы, усилие передается через жидкую среду, что связано с потерями энергии.

Рис. 3. Гидравлический пресс

ФИЗИЧЕСКАЯ СУЩНОСТЬ МАГНИТНО-ИМПУЛЬСНОЙ ОБРАБОТКИ

В основе метода магнитно-импульсной обработки металлов лежит явление электромагнитной индукции и действие сил Ампера.

Принцип работы предлагаемого мною метода формовки: накопитель энергии (в моем случае — батарея конденсаторов) заряжается до высокого напряжения. При разряде на катушку-инструмент в цепи возникает импульсный ток огромной величины, сравнимый с током потребления небольшого города, который создает вокруг катушки переменное магнитное поле. Если в это поле поместить токопроводящую заготовку (алюминиевую трубку или лист), то в ней, согласно закону электромагнитной индукции, наводятся вихревые токи. Взаимодействие магнитного поля катушки с полем наведенных токов порождает силу Ампера, которая и совершает механическую работу по деформации металла.

ПРЕИМУЩЕСТВА МАГНИТНО-ИМПУЛЬСНОЙ ФОРМОВКИ ПЕРЕД АНАЛОГИЧНЫМИ МЕТОДАМ

Я выделил преимущества мной реализованной установки:

1. Без термического воздействия. Деформация происходит за счет электромагнитных сил, а не нагрева, поэтому структура металла остается неизменной. Не образуются хрупкие зоны, окалина, внутренние напряжения.
2. Бесконтактность. Отсутствие механического контакта между инструментом и заготовкой позволяет инструменту не изнашивается. Не требуется смазка, поверхность детали остается чистой.
3. Высокая скорость и точность. Процесс деформации происходит за микросекунды. За это время заготовка принимает форму, точно соответствующую геометрии катушки или матрицы.

Возможность соединения разнородных материалов. Метод позволяет надежно

4. соединять алюминий со сталью, что практически очень трудно сделать традиционной сваркой. В момент разряда происходит высокоскоростное соударение металлов без нагрева всего объема детали.
5. Энергоэффективность. Я выяснил, что установка накапливает энергию (до 10 кДж) относительно долго — несколько секунд, а отдает ее за микросекунды. Это позволяет получать большую импульсную мощность при небольшом потреблении энергии от сети.

СРАВНЕНИЕ С АНАЛОГИЧНЫМИ СОВРЕМЕННЫМИ УСТАНОВКАМИ МАГНИТНО-ИМПУЛЬСНОЙ ФОРМОВКИ

На данный момент, в России, такие установки производят в Самарском университете. Я изучил их разработку, сравнил с предлагаемой мной установкой.

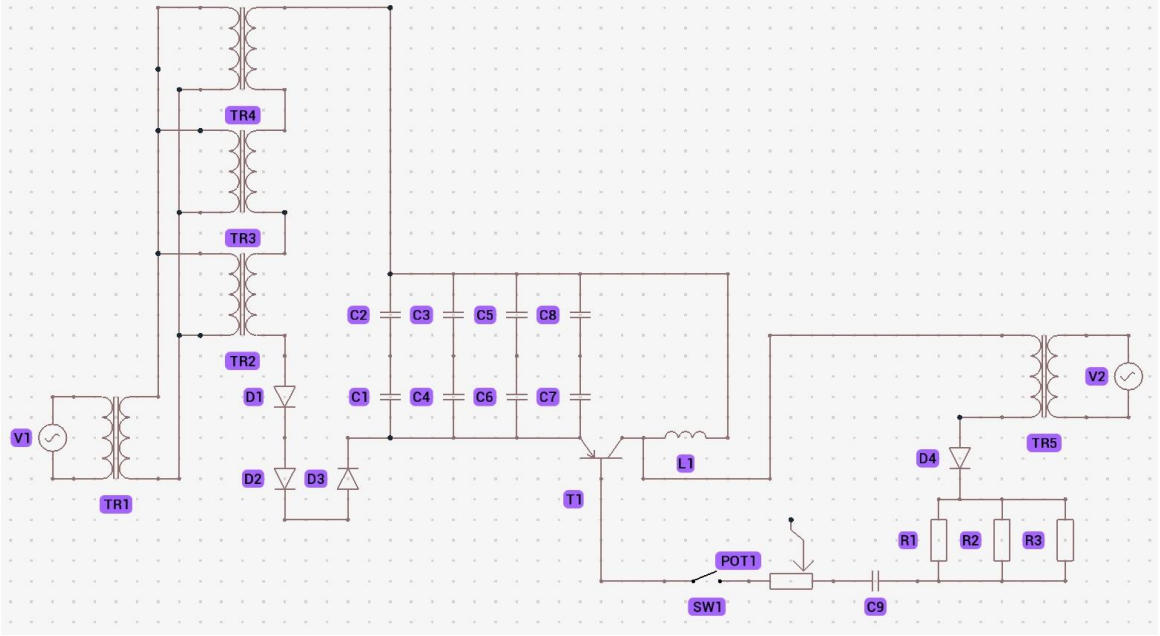
Принципиальное отличие заключается в схеме коммутации. В самарской установке используется воздушный искровой разрядник — устройство, в котором замыкание цепи происходит через пробой воздушного промежутка между электродами. У такого метода, на мой взгляд, есть несколько недостатков (нестабильность параметров, эрозия электродов, потери и помехи), от которых можно избавиться, используя иной тип коммутатора — игнитрон ИРТ-6. Это герметичный газоразрядный прибор, в котором коммутация происходит в защищенной среде, не зависящей от внешних условий, обладающий преимуществами:

- Стабильность параметров срабатывания при любых внешних условиях;
- Высокий ресурс (сотни тысяч циклов без деградации);
- Меньший уровень помех при работе;
- Компактность и отсутствие необходимости в обслуживании.

ГЛАВА 2. ПРАКТИЧСКАЯ РЕАЛИЗАЦИЯ МАГНИТНО-ИМПУЛЬСНОЙ УСТАНОВКИ

Для практической реализации выделяю следующие этапы:

1. Конструкторский этап. Проектирование электрической схемы. Разработка компоновки узлов, выбор элементной базы, проектирование системы управления и защиты.
2. Экспериментальный этап. Сборка установки, отладка режимов работы, проведение серии экспериментов.
3. Аналитический этап. Анализ полученных результатов, корректировка, внесение изменений в конструкцию.



Мной была спроектирована электрическая схема установки (Рис. 4).

Рис. 4. Схема магнитно-импульсной установки

Узлы установки:

ЛАТР TR1	0В – 250В
Повышающие трансформаторы для заряда TR2-4	0кВ – 10кВ
Диодные столбы D1, D2, D3	
Сборка конденсаторов C1-8	200 мкФ 10 кВ
Игнитрон ИРТ-6 T1	25 кВ 100 кА
Катушка-инструмент L1	
Искровой промежуток поджига SW1	0.4 мм ~1500
Ограничительный резистор POT1	10 Ом
Конденсатор C9	1 мкФ 2000 В
Ограничительные резисторы R1-3	2 МОм
Диодный столб D4	
Повышающий трансформатор TR5	220 В – 2000 В

Игнитрон, используемый в качестве коммутатора, рассчитан на напряжение 25 кВ и ток до 100 кА. Для его запуска требуется поджигающий импульс. Готовые промышленные блоки поджига либо дороги, либо сложны. Поэтому я разработал собственная схема поджига (Рис.5.) на основе искрового пробоя воздуха.

Суть решения: изготовлен искровой промежуток с регулируемым зазором. Экспериментальным путем мною установлен оптимальный зазор — 0,4 мм. Этот зазор обеспечивает стабильный пробой при напряжении около 2 кВ. При достижении этого

напряжения проскакивает искра, которая запускает игнитрон. Ток поджига ограничивается резистором, что защищает электроды от разрушения.

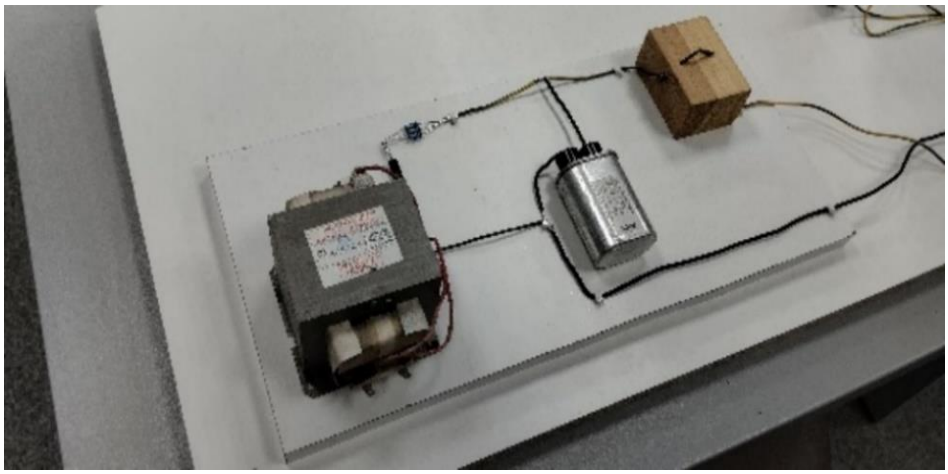


Рис. 5. Поджиг игнитрона экспериментальной магнитно-импульсной установки

Для точной дозировки энергии использую лабораторный автотрансформатор (ЛАТР) (Рис.6.). Он позволяет плавно регулировать напряжение заряда конденсаторов и подбирать оптимальный режим для разных заготовок.

Рис. 6. Лабораторный автотрансформатор

СБОРКА

На этапе сборки произвожу монтаж высоковольтных цепей, намотку катушек-инструментов и изготовление защитных кожухов.

При монтаже высоковольтных цепей все соединения должны выдерживать высокие импульсные токи и напряжения.

Для изготовления катушек-инструментов (Рис.7) используются различные материалы: медный обмоточный провод ПЭТВ-2 диаметром 1 мм и 2 мм, а также алюминиевый лакированный провод диаметром 0,8 мм. Количество витков подобрано под конкретную задачу, обычно от 7 до 10 витков.

Рис. 7. Катушка-инструмент

Изготовление защитных кожухов (Рис.8,9.). Все высоковольтные узлы закрываю защитными кожухами из оргстекла. Для их изготовления мною был собран небольшой станок.

Рис. 8. Кожухи из оргстекла

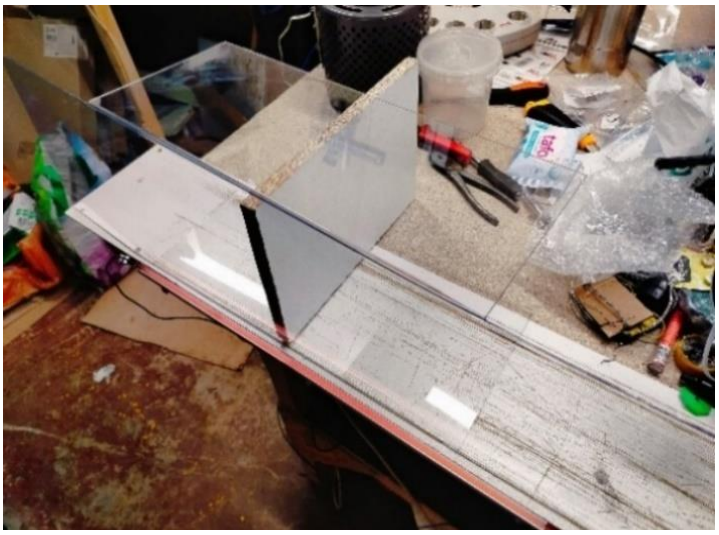


Рис. 9. Кожухи из оргстекла

Проведенные эксперименты подробно рассматриваются в главе 3.

ВИЗУАЛЬНЫЙ КОНТРОЛЬ И КОРРЕКТИРОВКА РЕЗУЛЬТАТОВ

На каждом этапе работы производились теоретические расчеты, которые затем проверялись экспериментально. Например, расчет энергоемкости батареи производился по формуле $E = (C \cdot U^2)/2$. Для батареи из восьми конденсаторов, собранных по схеме 4 параллельно и 2 последовательно, емкость составила 200 мкФ, максимальное напряжение — 10 кВ. Таким образом, максимальная запасаемая энергия составляет 10 кДж.

После каждого разряда проводился осмотр:

• состояния катушки (целостность изоляции, смещение витков);

• состояния заготовки (равномерность деформации, отсутствие трещин);

• состояния искровых промежутков;

• состояния изоляции.

Визуальный контроль позволил оперативно выявлять проблемы и корректировать параметры экспериментов.

ГЛАВА 3. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ И РЕЗУЛЬТАТЫ

ОПИСАНИЕ ЭКСПЕРИМЕНТАЛЬНОЙ МАГНИТНО-ИМПУЛЬСНОЙ УСТАНОВКИ

Части магнитно-импульсной установки:

- Накопитель энергии: батарея из 8 масляных конденсаторов (схема 4 параллельно, 2 последовательно)
- Емкость батареи: 200 мкФ
- Максимальное рабочее напряжение: до 10 кВ
- Максимальная запасаемая энергия: до 10 кДж
- Коммутатор: игнитрон (25 кВ, 100 кА)
- Система поджига: искровой промежуток с зазором 0,4 мм
- Регулировка напряжения: лабораторный автотрансформатор (ЛАТР)
- Катушки-инструменты: медный провод ПЭТВ-2 (1-2 мм) и алюминиевый провод (0,8 мм), 7-10 витков

Безопасность установки: высоковольтные узлы закрыты защитными кожухами из оргстекла, управление выведено на выносную панель с восьмиметровыми проводами, работа ведется в диэлектрических перчатках.

Порядок работы с магнитно-импульсной установкой

Этап 1. Подготовительный

Рис. 10. Защитные перчатки

- Надеть диэлектрические перчатки
- Убедиться в целостности защитных кожухов
- Проверить визуально состояние катушки и заготовки
- Разместить заготовку (трубку или лист с матрицей)

Этап 2. Подключение

Рис. 11. Подключение

- Подключить выбранную катушку-инструмент к выходным клеммам
- Закрыть защитный кожух рабочей зоны

Этап 3. Заряд

Рис. 12. Включение зарядного устройства

- Включить зарядное устройство
- Установить (плавно) ЛАТРом требуемое напряжение (от 2 до 8 кВ)

Этап 4. Разряд

Рис. 13. Подача разряда

- Отойти на безопасное расстояние с выносным пультом (8 м)
- Подать сигнал окружающим о готовности к разряду
- Нажать кнопку «Огонь» на выносном пульте
- Зафиксировать характерный хлопок при разряде

Этап 5. Контроль

- Подойти к установке
- Разрядить конденсаторы несколько раз
- Снять защитный кожух
- Извлечь заготовку
- Провести визуальный контроль деформации

Этап 6. Фиксация результатов

- Измерить полученные размеры (штангенциркулем)
- Записать параметры в журнал (U, E, кол-во витков, результат)
- При необходимости сфотографировать результат

Этап 7. Завершение

- При проведении серии экспериментов повторить цикл с другими параметрами
- По окончании работ убедиться, что установка обесточена и конденсаторы разряжены
- Привести рабочее место в порядок

ЭКСПЕРИМЕНТ 1: ВНЕШНЕЕ СЖАТИЕ

Цель: проверить способность установки равномерно обжимать трубчатые заготовки снаружи.

Методика: Алюминиевая трубка надевалась на оправку. Снаружи располагалась катушка-инструмент. Производился разряд.

Результаты: Катушка равномерно обжимает трубку по всей длине зоны воздействия. Деформация происходит точно в заданном месте без нагрева. Испытаны трубки диаметром от 5 до 20 мм. Во всех случаях получен положительный результат.

Вывод: Установка может работает в режиме внешнего сжатия.

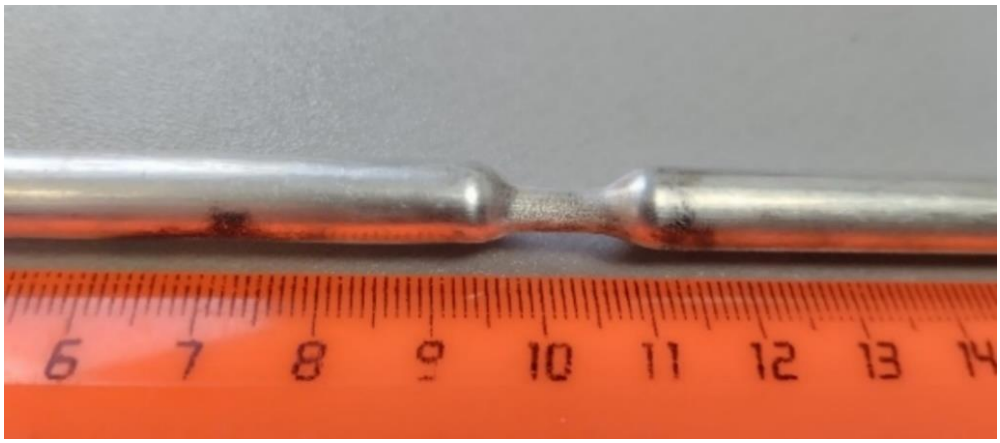


Рис. 14. Результат эксперимента: внешнее сжатие

ЭКСПЕРИМЕНТ 2: ВНУТРЕННЕЕ РАСШИРЕНИЕ

Цель: проверить способность установки расширять трубчатые заготовки изнутри.

Методика: Катушка помещалась внутрь алюминиевой трубки. Производился разряд.

Результаты: Катушка равномерно расширяет трубку. Диаметр увеличивался без образования трещин и складок.

Вывод: Режим внутреннего расширения может использоваться для посадки труб на наружные детали.

Рис. 15. Результат эксперимента: внутреннее расширение

ЭКСПЕРИМЕНТ 3: ШТАМПОВКА РЕЛЬЕФА

Цель: проверить возможность получения рельефных изображений на листовых заготовках.

Методика: между катушкой и матрицей с рисунком помещался лист алюминия. При разряде поле прижимало заготовку к матрице.

Результаты: на поверхности получилась точная зеркальная копия рельефа матрицы. При штамповке монет рисунок отпечатался четко.

Вывод: Установка может использоваться для рельефной штамповки.

Рис. 16. Результат эксперимента: штамповка

Рис. 17. Образовавшийся рельеф

ЭКСПЕРИМЕНТ 4: СОЕДИНЕНИЕ РАЗНОРОДНЫХ МАТЕРИАЛОВ

Цель: проверить возможность соединения алюминиевой трубки со стальным стержнем без нагрева.

Методика: Стальной стержень помещался внутрь алюминиевой трубки. Снаружи располагалась катушка. Производился разряд.

Результаты: Трубка обжала стержень. Соединение невозможно разорвать руками — разрушение происходило по алюминию, а не по границе соединения.

Вывод: Установка позволяет создавать неразъемные соединения алюминия со сталью без нагрева.

Рис. 18. Соединение меди и стали

Рис. 19. Соединение меди и стали

Рис. 20. Соединение алюминия и стали

Рис. 21. Соединение алюминия и стали

ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

Серия экспериментов подтвердила работоспособность установки и универсальность метода. Во всех случаях деформация происходила без нагрева. Диапазон заготовок (трубки 5-20 мм, лист) показывает гибкость установки. Эксперимент с соединением разнородных материалов имеет важное практическое значение.

Зафиксированы следующие результаты проведенных экспериментов:

Таблица 1. Внешнее сжатие

Материал трубки	U, В	E, Дж	N, витки	d нач., мм	d кон., мм
Алюминий	2000	400	9	8	7,8
Алюминий	4000	1600	6	8	7,5
Медь	6000	3600	14	8	6,0
Медь	8000	6400	7	8	4,5

Таблица 2. Внутреннее расширение

Материал трубки	U, В	E, Дж	N, витки	d нач., мм	d кон., мм
Алюминий	2000	400	8	20	20,5
Алюминий	4000	1600	6	20	20,9
Медь	6000	3600	12	18	20

Медь	8000	6400	5	18	23

Таблица 3. Штамповка рельефа

Материал заготовки	U, В	E, Дж	Толщина, мм	Результат
Алюминий	4000	1600	1,0	Нечеткий отпечаток
Алюминий	6000	3600	1,0	Среднее качество
Алюминий	6000	3600	1,0	Хороший отпечаток
Алюминий	8000	6400	1,0	Продавливание рисунка

Таблица 4. Соединение разнородных материалов

Материал трубки	Материал стержня	U, В	E, Дж	Результат
Алюминий	Сталь	4000	1600	Соединение держится
Алюминий	Сталь	6000	3600	Прочное соединение
Медь	Сталь	6000	3600	Соединение неразъемное
Алюминий	Сталь	8000	6400	Разрушается по металлу

ГЛАВА 4. НОВИЗНА И ПЕРСПЕКТИВЫ

НОВИЗНА ПРЕДЛАГАЕМЫХ РЕШЕНИЙ

Система поджига игнитрона на основе искрового пробоя с зазором 0,4 мм отличается простотой и надежностью.

Отработана технология изготовления катушек из различных материалов (медь ПЭТВ-2 1-2 мм, алюминий 0,8 мм) с различным числом витков.

Рис. 22. Провод эмалированный

Рис. 23. Выносная панель управления

Реализована система безопасности с защитными кожухами (Рис.8.) и выносным пультом (Рис.23).

ПРЕДЛОЖЕНИЯ ПО ПРАКТИЧЕСКОМУ ИСПОЛЬЗОВАНИЮ

Экспериментально-лабораторный практикум. Установка может использоваться для демонстрации явлений электродинамики.

Моделирование промышленных процессов. Установка позволяет исследовать влияние параметров на результат формовки.

Изготовление мелких партий деталей. Возможность быстрой переналадки делает установку пригодной для изготовления небольших партий.

ПЕРСПЕКТИВЫ РАЗВИТИЯ

1. Увеличение энергоемкости для работы с толстостенными заготовками.
2. Изготовление катушек сложной формы.
3. Исследование прочности соединений разнородных металлов.

ЗАКЛЮЧЕНИЕ

В процессе работы над проектом, мной были изучены принципы работы существующих промышленных аналогов, выявлены их недостатки, проведен сравнительный анализ с предлагаемой электромагнитной установкой. Разработана электрическая схема установки, которая обеспечивает накопление энергии и её импульсный разряд через катушку-инструмент. Были подобраны компоненты (конденсаторы, игнитрон, элементы управления. Также изготовлены сменные катушки-инструменты для различных режимов обработки. Внедрен новый тип коммутатора – игнитрон ИРТ-6, улучшающий характеристики аналогов. Собрана полностью работающая магнитно-импульсная установка с энергией до 10 кДж, что является практической значимостью работы. В качестве подтверждения работы установки, были проведены эксперименты по внешнему сжатию, внутреннему расширению, штамповке и соединению разнородных материалов. Намечены пути развития.

Этот проект стал для меня наглядным доказательством простой истины: за любым технологическим прорывом стоят базовые законы физики. Электродинамика, которую мы проходим на уроках, здесь работает в полную силу, превращая энергию электричества в реальное изменение формы металла. Мне удалось пройти весь путь самостоятельно — от идеи до работающего образца. Считаю, что задачи выполнены, цель достигнута.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Физические основы магнитно-импульсной обработки металлов. (б.д.). Получено из Скиф.Библиотека: https://de.donstu.ru/CDOCourses/structure/mash_stroit_tech/masp/316/book/6.%D0%AD%D0%9B%D0%95%D0%9A%D0%A2%D0%A0%D0%9E%D0%9C%D0%95%D0%A5%D0%90%D0%9D%D0%98%D0%A7%D0%95%D0%A1%D0%9A%D0%98%D0%95%20%D0%9F%D0%A0%D0%9E%D0%A6%D0%95%D0%A1%D0%A1%D0%AB/6.3.%20%D0%A4%

Д.Г. Черников, В. К. (3 04 2020 г.). ИССЛЕДОВАНИЕ ПРОЦЕССА МАГНИТНО-ИМПУЛЬСНОЙ ФОРМОВКИ С УЧЕТОМ ЭФФЕКТА ПРОСАЧИВАНИЯ МАГНИТНОГО ПОЛЯ СКВОЗЬ ЗАГОТОВКУ. Получено из https://cyberleninka.ru/article/n/issledovanie-protsessa-magnitno-impulsnoy-formovki-s-uchetom-effekta-prosachivaniya-magnitnogo-polya-skvoz-zagotovku/viewer

Магнитно-импульсная обработка металлов: от теории к практике. (20 05 2020 г.). Получено из https://m.vk.com/@vekomet-magnitno-impulsnaya-obrabotka-metallov-ot-teorii-k-praktike?ysclid=mm5mcvl6o8476604313

ПРИЛОЖЕНИЕ

Рис. 24. Общий вид установки

УСТРОЙСТВО КОНТРОЛЯ ИНФУЗИИ

Малахов Александр Дмитриевич

Москва г, ГБОУ Бауманская инженерная Школа №1580, 11 класс

Секция: Радиоэлектроника, лазерная и медицинская техника

Работа выполнена без научного руководителя

АННОТАЦИЯ

Анализатор низкотемпературных характеристик нефтепродуктов — прибор, измеряющий температуры помутнения, температуру потери текучести и температуру застывания нефтепродуктов. Это необходимо для оценки качества топлива. Такой прибор применим на НПЗ, перекачивающих станциях и АЗС. В проекте разработан его прототип, включающий холодильник на элементах Пельтье, насос постоянного давления и систему управления на базе Arduino с программной обработкой данных на Python. Все основные компоненты созданы и собраны, написана предварительная версия ПО.

ВВЕДЕНИЕ

Анализатор низкотемпературных характеристик нефтепродуктов – прибор, измеряющий температуры помутнения[1], текучести (начала кристаллизации)[2], застывания (потери текучести)[3] нефтепродуктов, в частности дизельных топлив и машинных масел.

Подобные устройства используются на перекачивающих станциях, нефтеперерабатывающих заводах.

Техническое задание: Прибор должен быть способен охлаждать анализируемый образец до низких температур (около -50 °C), соответствовать установленным стандартам лабораторных измерений для прохождения дальнейшей сертификации.

[1] Температура помутнения (cloud point): Температура топлива, при которой происходит образование кристаллической структуры парафина, вызывая помутнение при охлаждении в установленных условиях.
[2] Температура текучести (pour point): Минимальная температура нефтепродуктов, при которой наблюдается течение испытуемого образца при установленных условиях испытания
[3] Температура потери текучести (no-flow point): Это температура испытуемого образца нефтепродуктов, при которой образование кристаллической структуры парафина и/или увеличение вязкости затрудняют движение поверхностного слоя испытуемого образца при установленных условиях испытания.

ЦЕЛИ И ЗАДАЧИ

Цель работы – Создание прототипа прибора, его основных элементов, алгоритма работы анализатора. Прототип должен удовлетворять техническому заданию.

Задачи работы:

1) Разработка и испытания холодильника
2) Разработка и испытания насоса
3) Разработка алгоритма работы прибора

МЕТОДИКА РАЗРАБОТКИ

Прибор использует в своей работе методы, описанные в следующих документах:

1) *ASTM D6749-02 Standard Test Method for Pour Point of Petroleum Products (Automatic Air Pressure Method)*

Метод описывает способ фиксации факта застывания нефтепродукта с помощью подачи небольшого положительного давления к поверхности образца.

На рис.1 изображена схема, кратко описывающая работу прибора, приведенная в стандарте

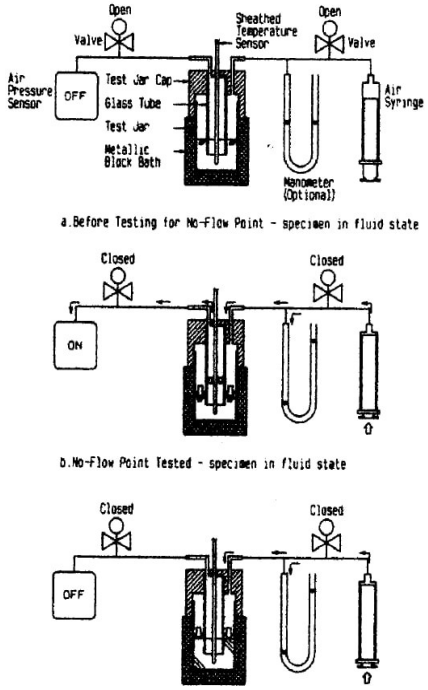


Рис. 1. Схема анализатора вязкости из документа

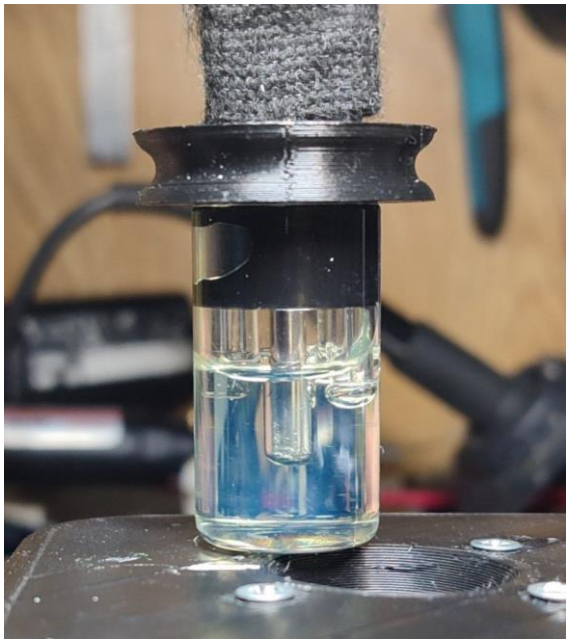


Рис. 2. Демонстрация прокачивания воздуха через образец

На рис.2 из трубки справа подаётся воздух к образцу. Термопара (PT100) в центре измеряет температуру, а стеклянная палочка слева, соединённая со светодиодом просвечивает образец.

2) *ASTM D97−7b Standard Test Method for Pour Point of Petroleum Products* .В методе описывается конструкция охлаждающей бани и принцип измерения температуры помутнения. На рис.3 изображен чертёж бани прибора, указанный в стандарте:

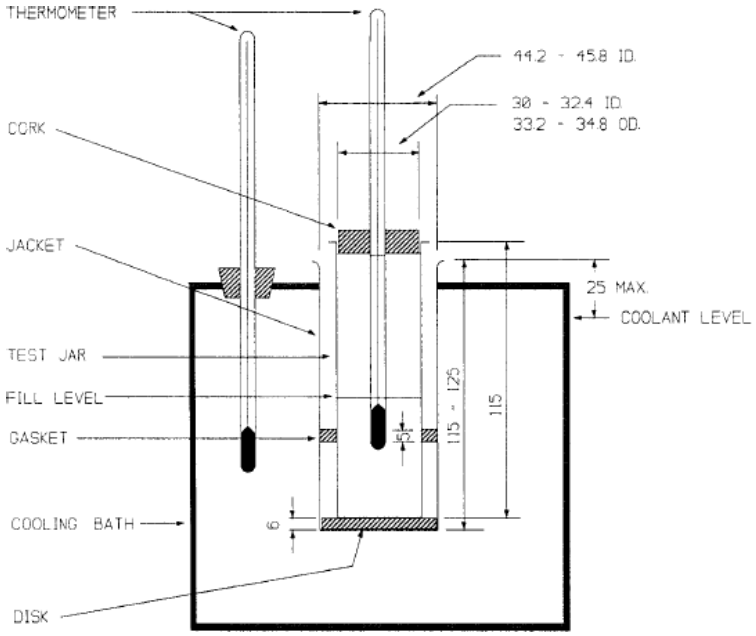


Рис.3. Чертёж бани из документа

На рис.4 приведён чертёж бани, разработанной на основе данных из *ASTM D97* в программе Компас 3D. Рис.5 — 3D модель бани прибора.

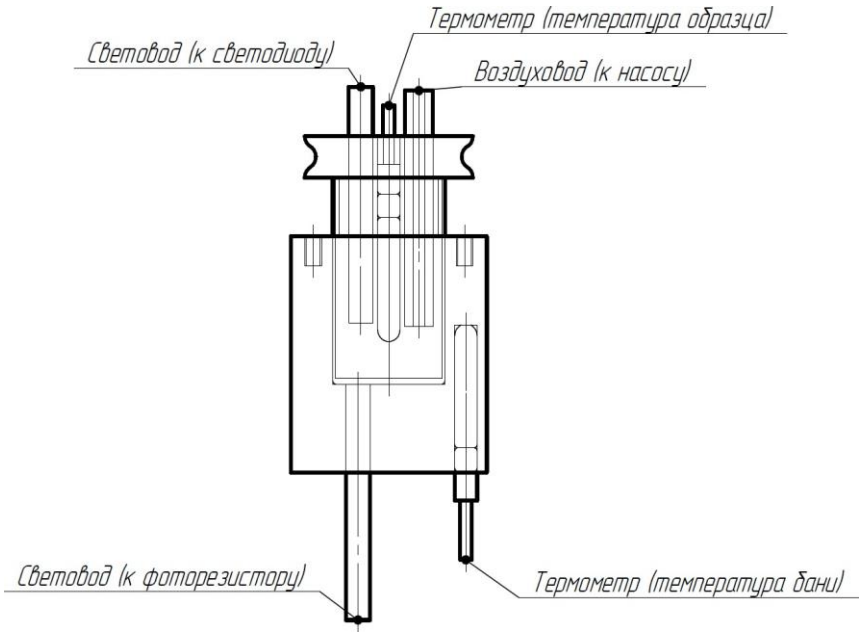


Рис. 4. Чертёж бани

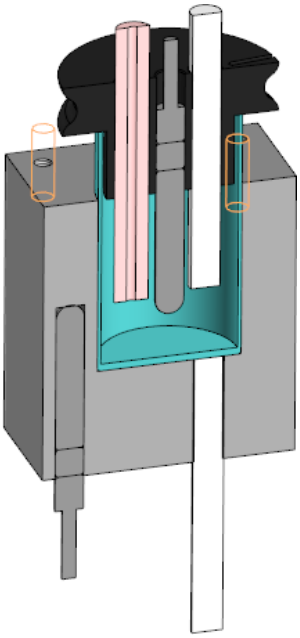


Рис. 5. 3D-модель

Фиксация температуры помутнения происходит с помощью измерения сопротивления фоторезистора (рис.6), находящегося под пробиркой с образцом

Рис. 6.1. Сопротивление фоторезистора с включённой лампой

Рис. 6.2. Сопротивление фоторезистора с выключенной лампой

Во время теста, анализатор просвечивает образец с помощью светодиода (рис.7)

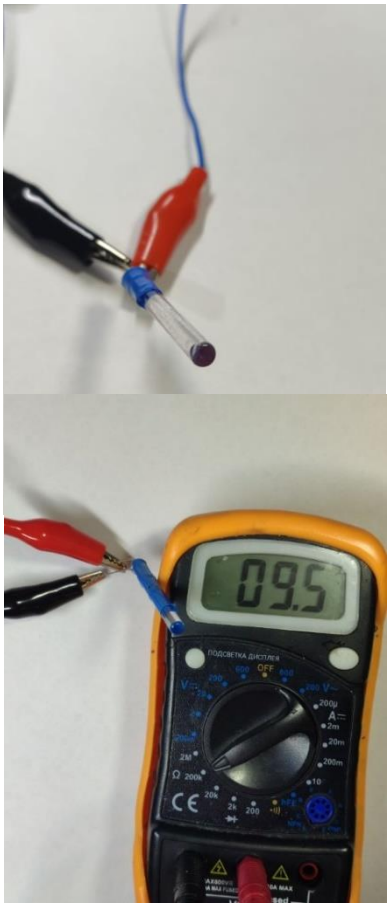


Рис. 7.1. Светодиод не горит

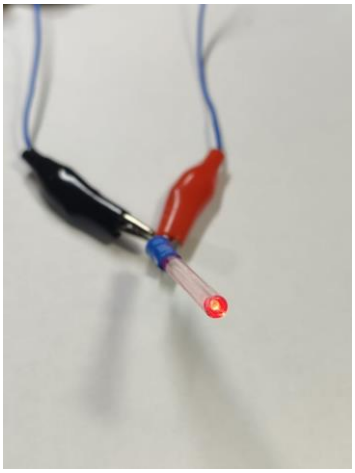


Рис. 7.2. Светодиод горит

ОСНОВНЫЕ ЭЛЕМЕНТЫ

1) Прибор охлаждает образец с помощью холодильника на элементах Пельтье[4].

Холодильник представляет собой несколько элементов Пельтье, одной стороной приложенных к дюралюминиевому теплообменнику (рис 9), а другой стороной к водоблокам, через которые прогоняется проточная вода (рис 8).

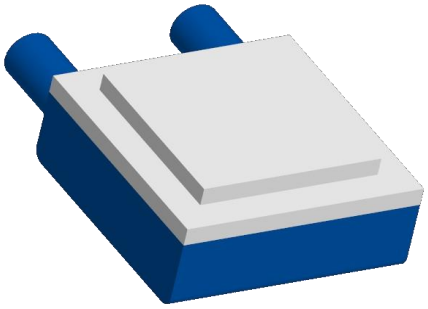


Рис. 8. Охлаждающий элемент

Дюралюминиевый теплообменник выполнен фрезеровкой (рис.9):

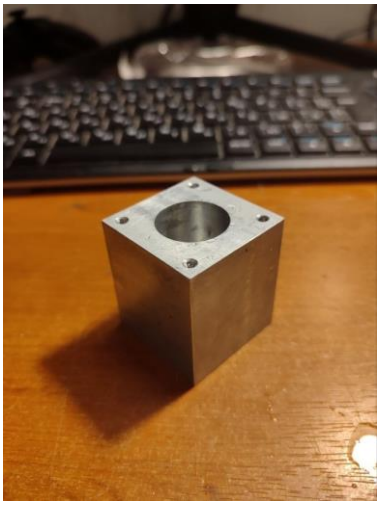


Рис. 9. Теплообменник

[4] Элемент Пельтье — термоэлектрический преобразователь, принцип действия которого основан на эффекте Пельтье — возникновении разности температур при протекании электрического тока

По чертежу, разработанному в Компас 3D (рис.10):

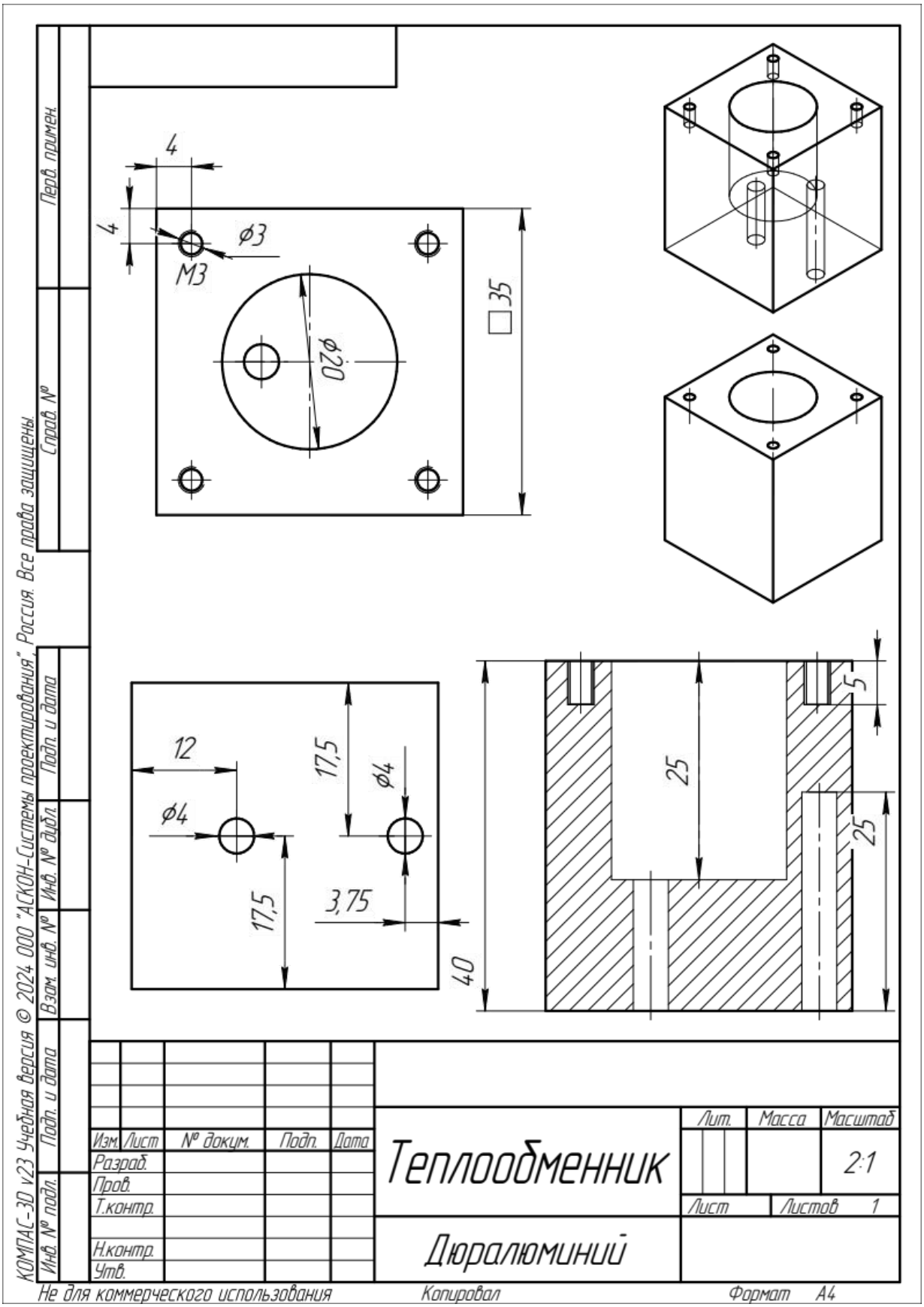


Рис. 10. Чертёж теплообменника

Собранный холодильник (рис.11,12):

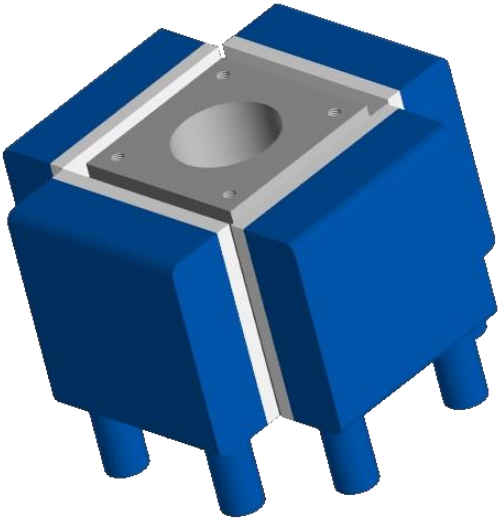


Рис. 11. 3D модель бани

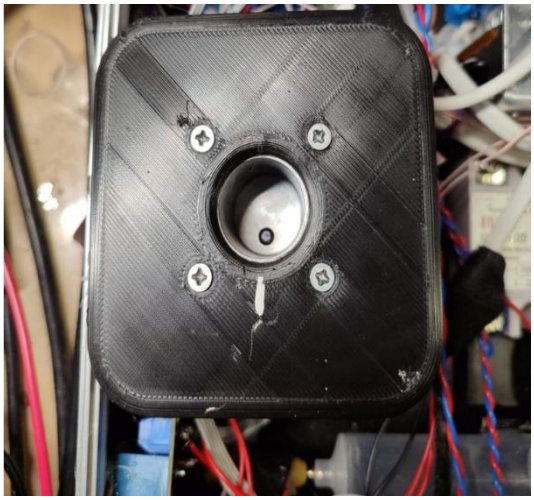


Рис. 12.1. Вид на холодильник сверху

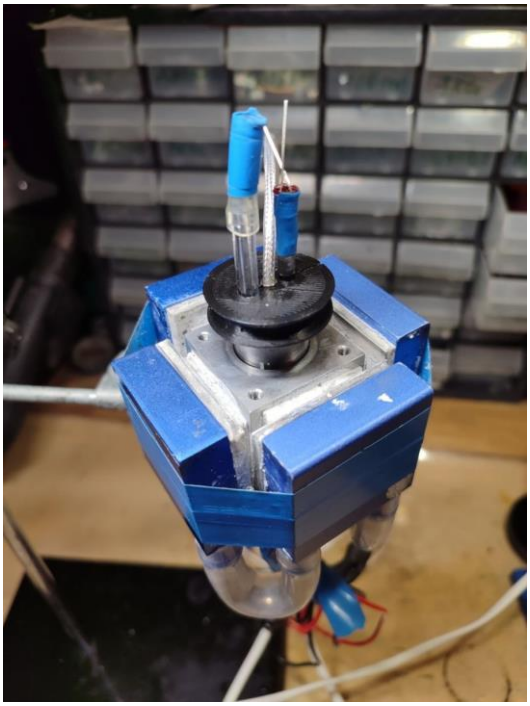


Рис. 12.2. Холодильник с пробкой

2) Другая немаловажная деталь прибора, это *насос*, способный выдавать постоянное давление (рис.14,15). Насос представляет собой шприц, поршень которого перемещается путём вращения ходового винта шаговым мотором :

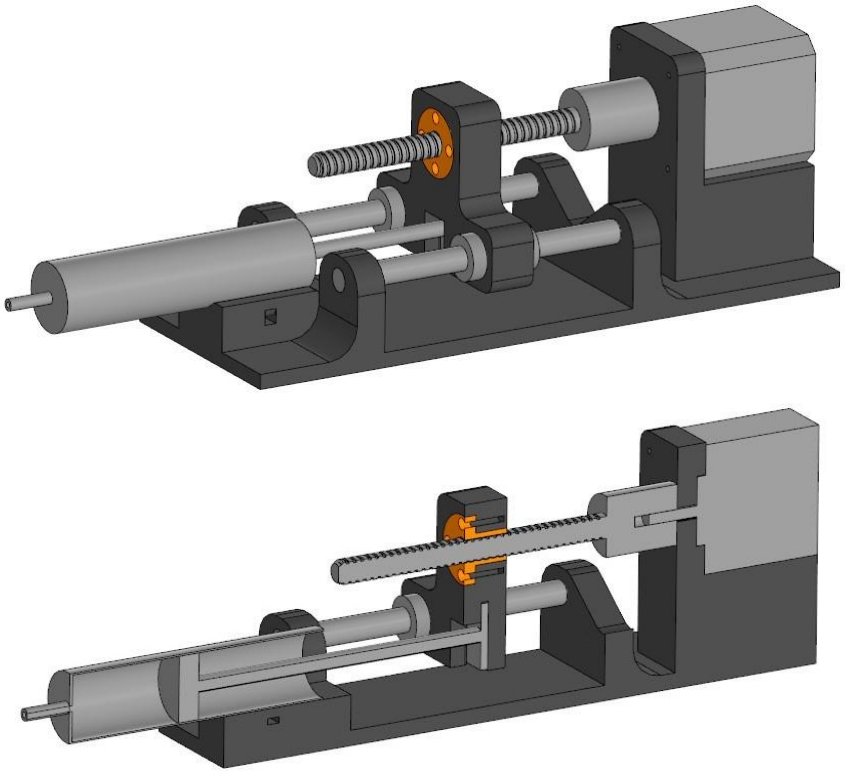


Рис. 14. 3D модель насоса и её сечение

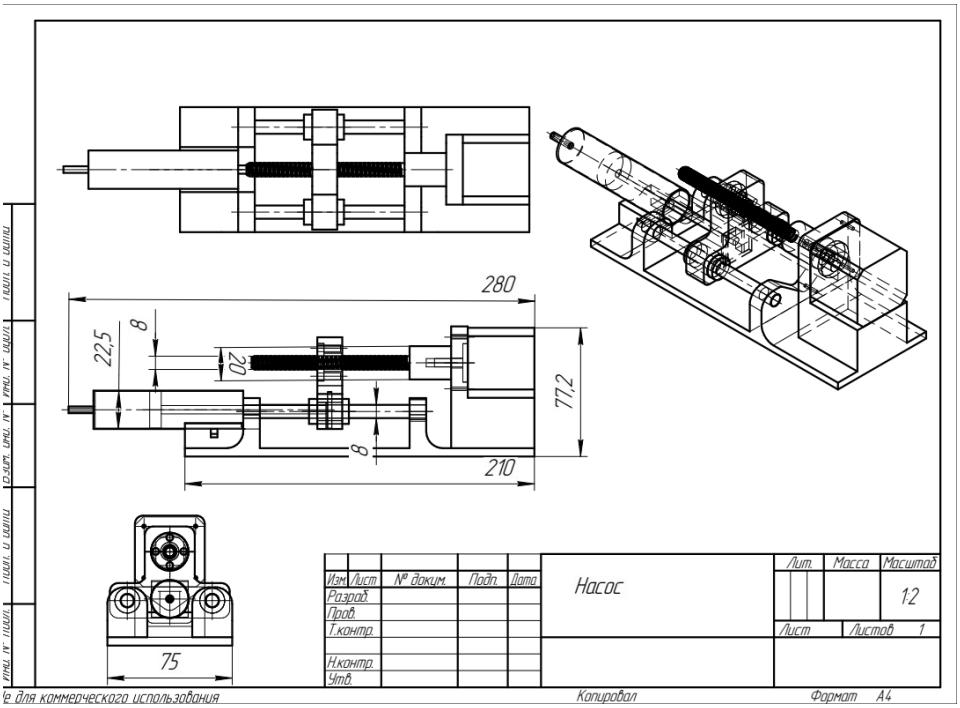


Рис. 15. Общий чертёж

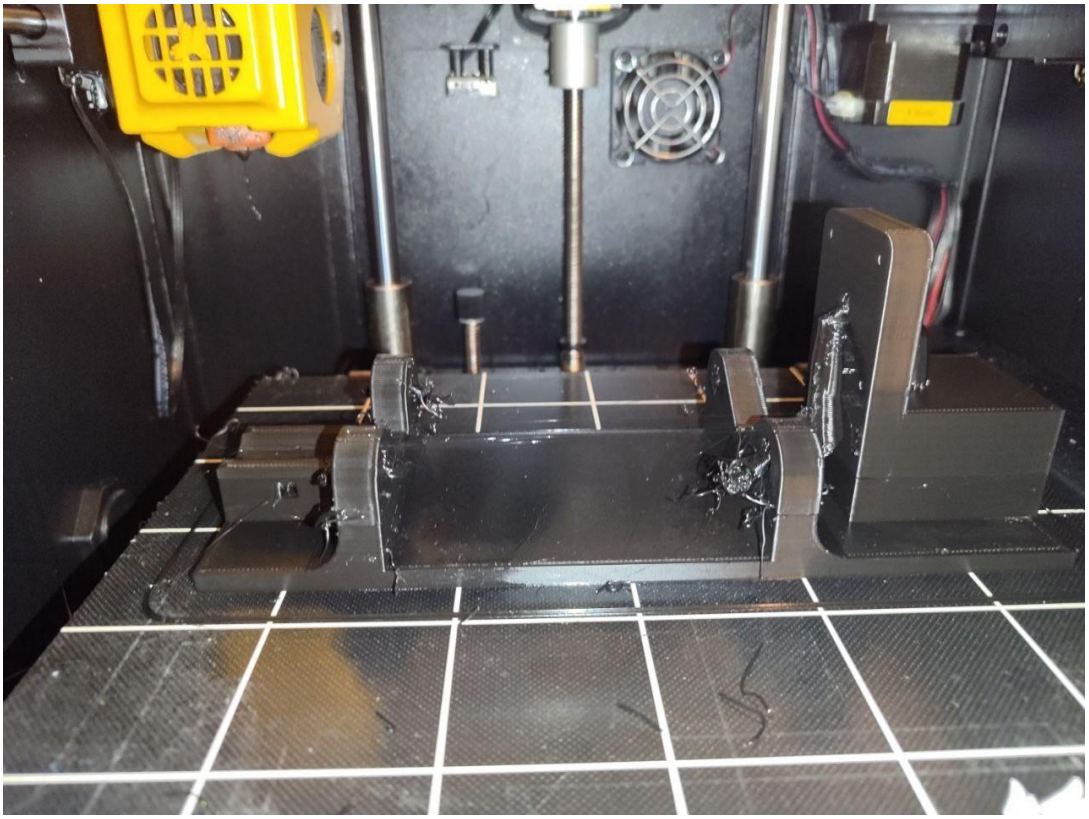


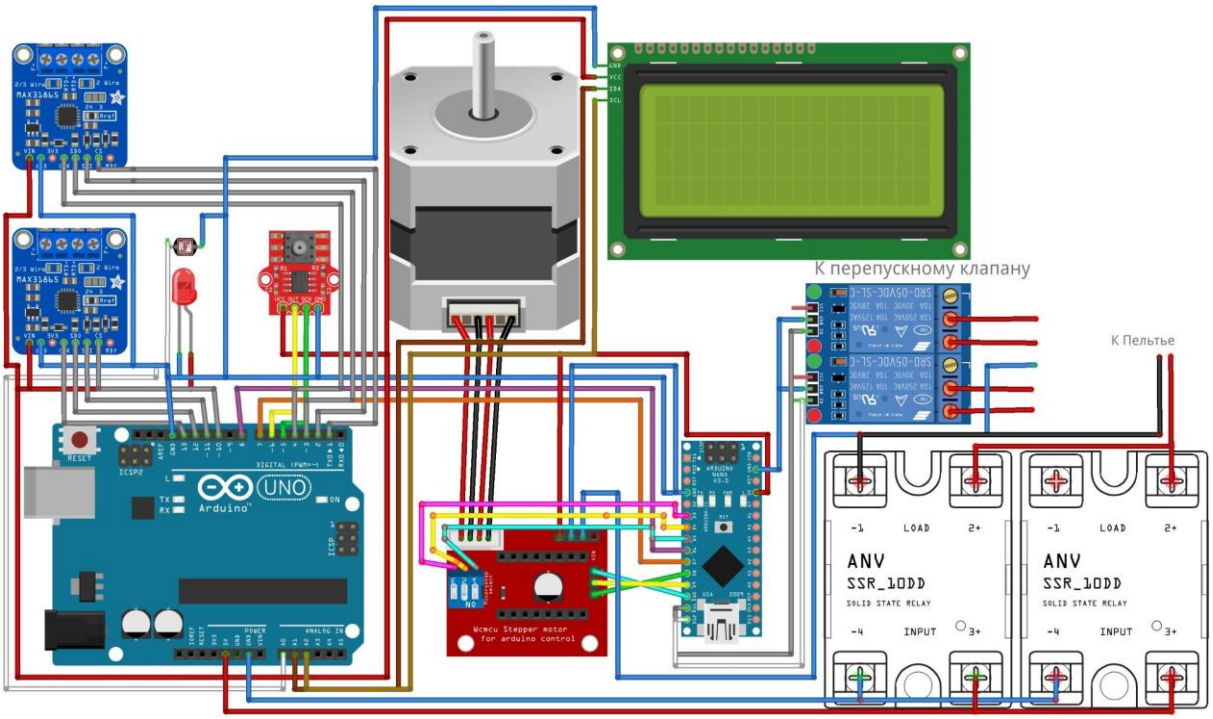
Рис. 16. Корпус насоса сразу после печати

Рис. 17. Собранный насос

Его корпус выполнен с помощью технологии FDM 3D печати (рис.16,17):

Прибор контролируют платы Arduino UNO и Arduino Nano. Они управляют холодильником и насосом, а также собирают данные с датчиков и передают их в COM-порт (рис 18)

Рис. 18. Схема подключения компонентов



Из порта данные принимает управляющая программа и оперирует ими согласно алгоритму.

Рис. 19. Сборка электроники «навесом»

Arduino Nano является контроллером мотора. Она общается с Uno через два цифровых пина и отвечает за управление драйвером мотора (насоса) A4988, а также перепускным клапаном.

Главная плата (Arduino UNO) получает данные с датчиков (температура, давление, сопротивление фоторезистора) и отправляет данные в COM-порт.

Компьютер получает данные из порта и обрабатывает их с помощью программы, написанной на языке Python. Затем он отправляет команды в порт, а arduino выполняет их. Также Arduino UNO включает и выключает холодильник, управляет мотором через контроллер (Arduino Nano)

Программа на phyton работает по алгоритму, представленном на рис.20:

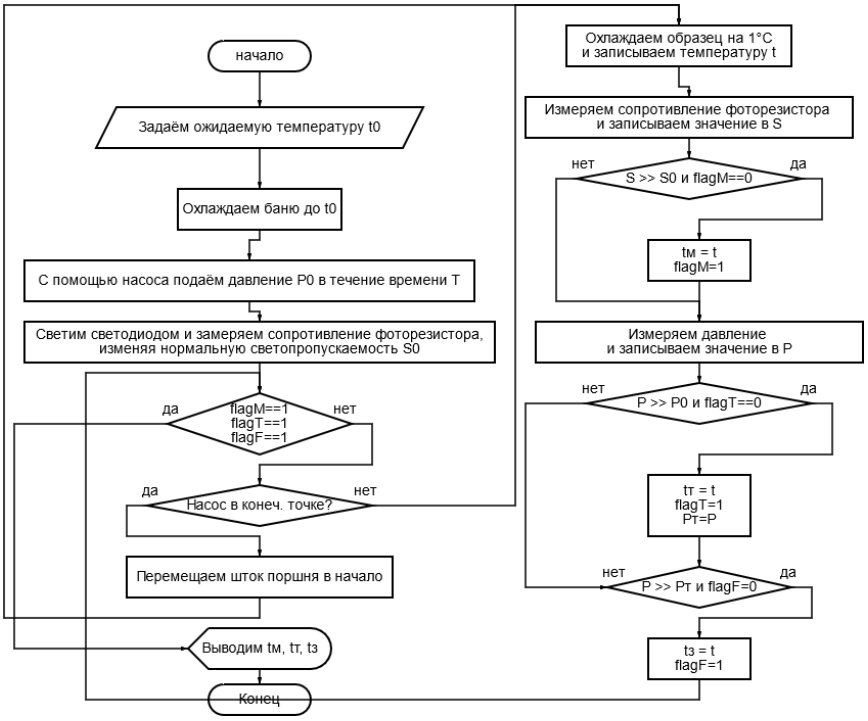


Рис. 20. Алгоритм работы прибора

Интерфейс Pre-Alpha версии программы (рис.21):

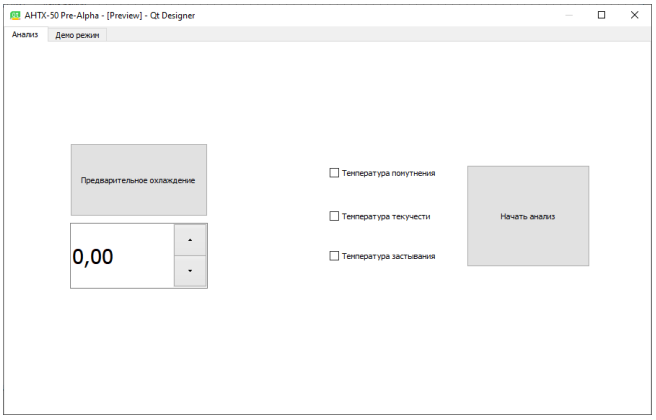


Рис. 21.1. Окно подготовки к анализу

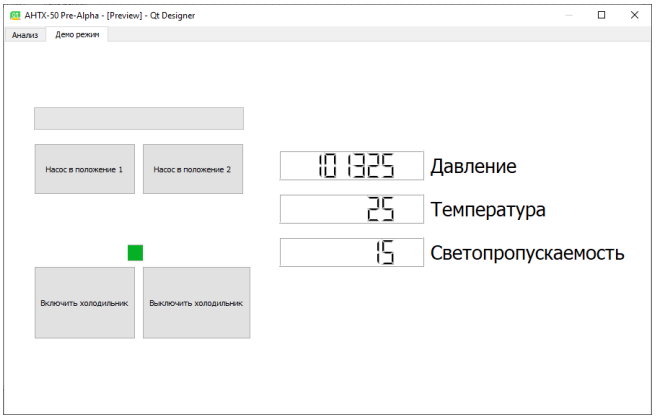


Рис. 21.2. Окно демо режима

РЕЗУЛЬТАТЫ

В процессе испытаний холодильника, при охлаждении с помощью чиллера (проточная вода температурой 5°C), была достигнута минимальная температура образца в -45,26°C (рис.22).

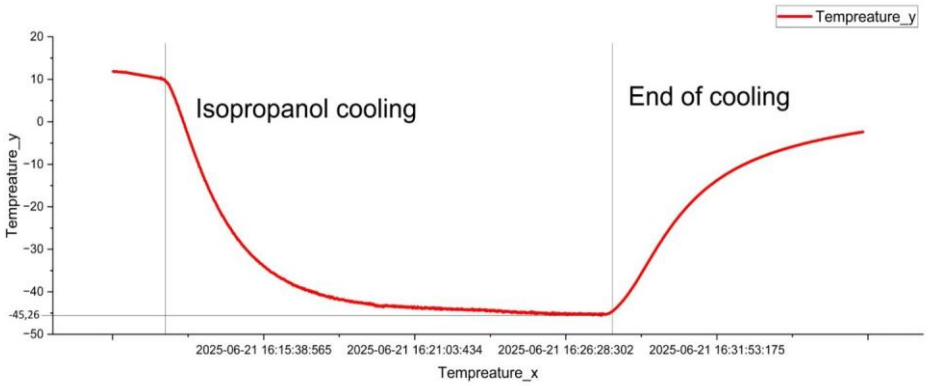


Рис. 22.1. График температуры изопропилового спирта при его охлаждении

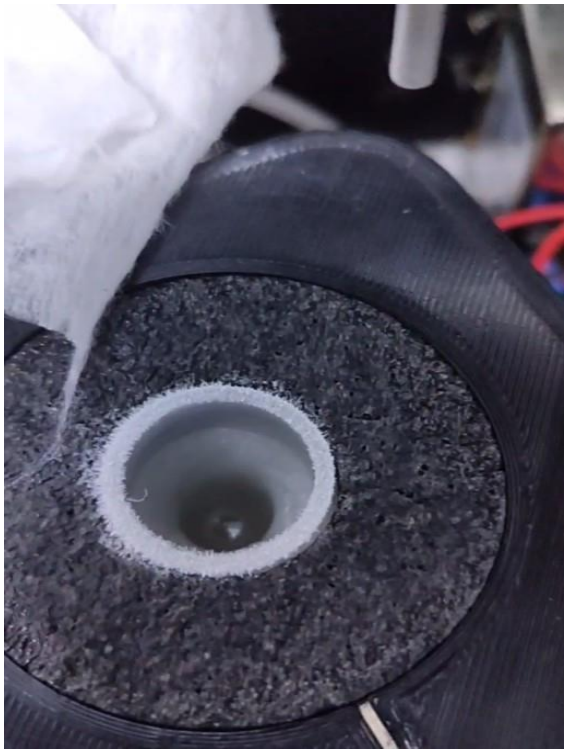


Рис.22.2: Замерзший образец-тетралин

Рис. 22.3. Пробирка с дизтопливом в бане

В результате испытаний компрессора получено, что он выдаёт постоянный поток воздуха, который приводит к поддержанию практически постоянного давления (рис.23), зависящему от скорости вращения вала.

Рис. 23. Показания манометра, подключенного к насосу

В качестве корпуса прибора был выбран алюминиевый кейс (рис.24):

Рис. 24.1. Прибор с закрытой крышкой

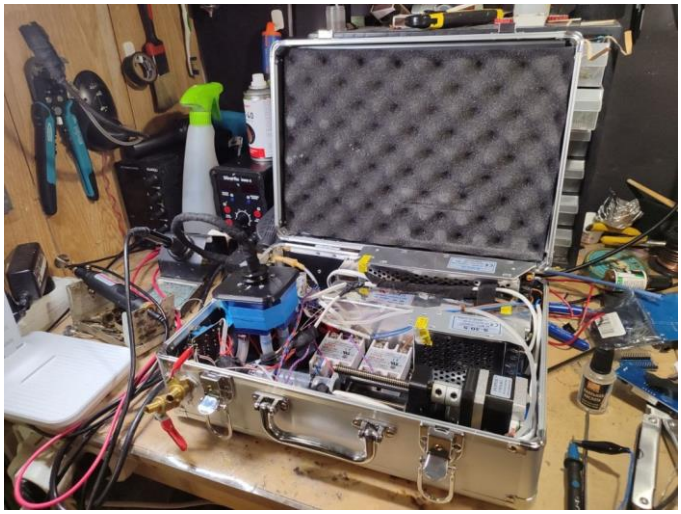


Рис. 24.2. Прибор с открытой крышкой

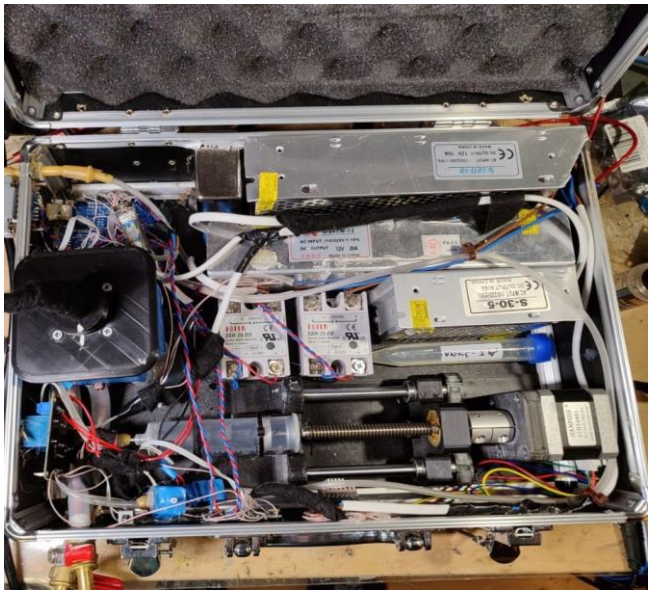


Рис. 24.3. Вид сверху

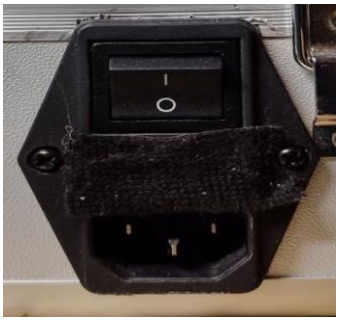


Рис. 24.4. Разъем питания

Рис. 24.4. Разъём для подключения пробки (сверху) и usb-b для подключения к компьютеру

ЗАКЛЮЧЕНИЕ

В результате работы были выполнены все поставленные задачи. Для окончательного выполнения цели необходимо прошить платы, написать программу на python по алгоритму, указанному на рис.19.

В процессе работы был разработан холодильник, способный охлаждаться до -45,26°C; насос, способный точно дозировать испускаемый объём газа (жидкости); алгоритм работы прибора и радиоэлектронная схема. Также прибор был полностью собран и создана Pre-Alpha версия управляющей программы.

В дальнейшем планируется:

- Написать полноценную программу с удобным интерфейсом.
- Разработать собственные платы с микроконтроллером, взамен Arduino.
- Переподключить электронику с помощью шлейфов.

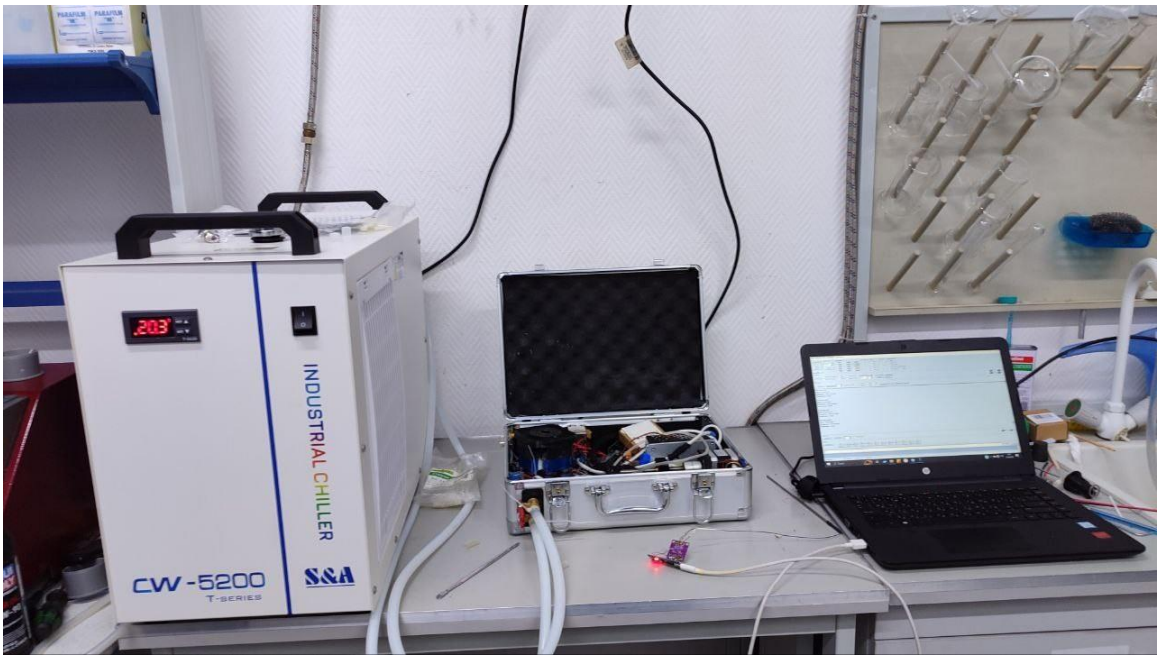


Рис. 25. Испытания

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ASTM D6749-02 (2018) Standard Test Method for Pour Point of Petroleum Products (Automatic Air Pressure Method)
2. ASTM D97-17b (2022) Standard Test Method for Pour Point of Petroleum Products
3. ASTM D2500-23 Standard Test Method for Cloud Point of Petroleum Products and Liquid Fuels
4. ГОСТ 25262-82 «Нефтепродукты жидкие. Метод определения температуры текучести»
5. ГОСТ 5066-2018 «Топлива моторные. Методы определения температур помутнения, начала кристаллизации и замерзания»
6. ГОСТ 32463-2013 «Нефтепродукты. Определение температуры потери текучести методом автоматического наклона
7. Автоматические приборы для анализа нефтепродуктов Tanaka Scientific Limited

ОБНАРУЖЕНИЕ ИЗМЕНЕНИЙ НА РАЗНОВРЕМЕННЫХ СНИМКАХ ГОРОДСКОЙ СРЕДЫ

Семенов Артём Александрович

Москва г, МБОУ СОШ № 2107, 11 класс

Секция: Математика и естественные науки

Научный руководитель: Семенов Александр Валерьевич, к.т.н., доцент кафедры «Технологии проектирования сложных технических систем» МФТИ

АННОТАЦИЯ

Актуальность. Задача обнаружения изменений на изображениях земной поверхности имеет важное значение для науки – она лежит в основе дистанционного зондирования, геоинформатики, экологического мониторинга, урбанистики и управления природными ресурсами. Точное выявление и интерпретация трансформаций ландшафта позволяют отслеживать динамику Земли как сложной системы и формировать научно обоснованные прогнозы её развития.

В контексте города эта задача приобретает особую практическую значимость. Раннее и надёжное выявление физических изменений в городской среде — таких как незаконная застройка, вырубка деревьев, появление свалок или последствия чрезвычайных ситуаций – является критически важным элементом систем управления умным городом. Однако её решение связано с рядом сложностей: изменения необходимо не просто зафиксировать, а интерпретировать – распознать, что именно изменилось, в каком контексте, с какой скоростью и к каким последствиям это может привести.

Современные подходы к обнаружению изменений, основанные на традиционных методах компьютерного зрения, сталкиваются с тремя ключевыми ограничениями:

1. Отсутствие семантической интерпретации: системы могут зафиксировать факт изменения – например, «появился объект» – но не способны распознать его природу («построен жилой дом», «установлен временный торговый павильон»), определить время, темп и контекст трансформации (например, связана ли застройка с инфраструктурным проектом или является самовольной).
2. Требовательность к разметке и дообучению: для распознавания каждого нового типа объектов или событий – будь то «незаконные свалки», «временные строительные ограждения» или «массовая вырубка деревьев» – требуется собирать размеченный

набор 120 данных и проводить повторное обучение модели. Это делает такие решения непригодными для оперативного мониторинга и адаптации к новым, непредвиденным сценариям.

3. Ограничение двумя снимками: большинство систем работают только с парой изображений («до» и «после»), что не позволяет выявлять нелинейные траектории развития – например, ускорение застройки до 2017 года, её замедление после 2018-го или циклические изменения в зелёных зонах. Такой подход игнорирует исторический контекст и не отражает реальную динамику городской среды.

Эти ограничения делают существующие инструменты неэффективными для задач устойчивого городского планирования, контроля за соблюдением градостроительных норм и оперативного реагирования на угрозы в условиях умного города. Особенно критичны для Москвы, где требуется оценка эффективности городских программ, контроль законности застройки и оперативное реагирование на ЧС.

Проблема: существует необходимость в современных методах и инструментах обнаружения изменений на снимках в разное время, но отсутствуют подходящие инструменты.

Целью работы является исследование возможностей открытых больших визуально языковых моделей (БВЯМ) обнаруживать и интерпретировать физические изменения в городской среде по нескольким разновременным снимкам, а также разработка на основе этих моделей инструмента для выполнения последующих исследований.

Методы: анализ литературы и существующих решений, эксперимент и сравнение БВЯМ применительно к задаче, систематизация полученных знаний и проектирование архитектуры веб-интерфейса.

Результаты работы:

1. Проведён анализ современных открытых БВЯМ:Qwen2.5-VL-3B, LLaVAOneVision-7B, InternVL3-8B и др. БВЯМ продемонстрировали хороший потенциал для применения к новым задачам обнаружения и интерпретации изменений в городской среде. Однако для этого требуется адаптация к геопространственному контексту (через промпты) и минимальное дообучение на коротких инструкциях.
2. Разработан инструмент Siuuu (Smart Interface for Urban Understanding and Updates) – веб-приложение на основе Streamlit, позволяющее использовать открытые БВЯМ для обнаружения и интерпретации изменений. Пользователь может задавать вопросы на естественном языке (например, «Как изменилась площадь парка за 2018–2022 гг.?») –такие запросы обрабатываются в режиме генерации текста и просить выделить изменения визуально (например, «Покажи все новые здания за этот период») – такие запросы обрабатываются в режиме сегментации. Модель

автоматически определяет тип запроса по формату и направляет его в соответствующий модуль, что позволяет избежать конфликта между задачами и исключает необходимость дообучения.

3. Прототип протестирован на представителе целевой аудитории – ООО ЦИТМ «Экспонента». Получено положительное экспертное заключение с рекомендациями по развитию продукта и возможной интеграции в системы мониторинга городской среды.

Перспективы развития данного направления связаны с дальнейшей интеграцией открытых БВЯМ в системы городского управления, в частности — в экосистему цифрового двойника «Мета-Москва». В ближайшей перспективе планируется адаптация Siuuu под запросы, валидация на отечественных снимках. Долгосрочная цель – создание легковесного, автономного и безопасного инструмента для оперативного семантического мониторинга городской среды, способного без дообучения интерпретировать динамику урбанизации, выявлять аномалии и генерировать аналитические отчёты, что соответствует стратегии ДИТ Москвы по развитию городского общего интеллекта.

ВВЕДЕНИЕ

Задача обнаружения изменений на разновременных снимках заключается в автоматическом сравнении нескольких изображений одной и той же территории, сделанных в разное время, для выявления различий в ландшафте, объектах или их состоянии. Эта задача является фундаментальной для дистанционного зондирования Земли. В контексте городской среды она приобретает особую практическую значимость. Выявление изменений позволяет контролировать такие процессы, как незаконная застройка, вырубка деревьев, появление свалок и трансформация инфраструктуры.

Многочисленные исследования учёных подтверждают практическую ценность анализа изменений на основе данных дистанционного зондирования (например, работы Барталёва С.А. [1], Лупян Е.А. [2], Станичной Р.Р. [3] и др., Станичный и Горячкин). В контексте градостроительства анализ исторических данных позволяет отслеживать тренды развития и предотвращать чрезмерную застройку. В экологическом мониторинге оценка изменений способствует управлению паводковыми рисками и снижению эффекта «городского теплового острова» [4].

Несмотря на прогресс в разработке моделей детекции изменений, существующие подходы обладают следующими ограничениями:

- ограничение, заключающееся в фокусе на генерации пиксельных карт изменений, приводит к последствиям на практике в виде отсутствия семантической интерпретации результатов;
- ограничение, связанное с рассмотрением только одного типа изменений, приводит к последствиям на практике как невозможность комплексного анализа;
- ограничение, выражающееся в отсутствии интеллектуального ответа на разнообразные запросы, приводит к последствиям на практике в виде низкой доступности для не-экспертов в ДЗЗ;
- ограничение, состоящее в зависимости от облачных API, приводит к последствиям на практике в виде рисков конфиденциальности данных.

С появлением больших визуально-языковых моделей (БВЯМ) возникла гипотеза о возможности их применения для решения поставленной задачи. Однако исследование показало, что использование БВЯМ без дополнительной архитектуры сопряжено с рисками. Модели склонны к «галлюцинациям»: они могут завышать площадь насаждений, ошибаться в интерпретации участков или выдумывать несуществующие объекты.

Цель исследования: разработка и тестирование агентной системы для обнаружения изменений на разновременных снимках городской среды на базе локальных больших

визуально-языковых моделей и специализированных инструментов компьютерного зрения.

Задачи:

1. Исследовать возможности современных открытых БВЯМ для работы со спутниковыми снимками.
2. Разработать способ решения задачи обнаружения изменений с применением агентного подхода и БВЯМ.
3. Разработать и реализовать алгоритм обнаружения изменений с применением агентного подхода и БВЯМ.

Новизна: предложен новый способ обнаружения изменений на двух разновременных снимках городской среды с применением локальной мультимодальной модели Qwen2VL. Решение основано на иерархической архитектуре с агентом, обеспечивающая снижение галлюцинаций локальной мультимодальной модели Qwen2VL за счёт интеграции инструментов объектно-ориентированного и пиксельно-ориентированного анализа изображений.

Практическая значимость: разработан инструмент Siuuu (Smart Interface for Urban Understanding and Updates) – веб-приложение на основе Gradio, позволяющее использовать открытые БВЯМ для обнаружения и интерпретации изменений. Пользователь может задавать вопросы на естественном языке (например, «Как изменилась площадь парка за 2018–2022 гг.?») –такие запросы обрабатываются в режиме генерации текста и просить выделить изменения визуально («Покажи все новые здания за этот период») – такие запросы обрабатываются в режиме сегментации. Модель автоматически определяет тип запроса по формату и направляет его в соответствующий модуль, что позволяет избежать конфликта между задачами и исключает необходимость дообучения.

Прототип протестирован на представителе целевой аудитории – ООО ЦИТМ «Экспонента». Получено положительное экспертное заключение с рекомендациями по развитию продукта и возможной интеграции в системы мониторинга городской среды.

Перспективы развития данного направления связаны с дальнейшей интеграцией открытых БВЯМ в системы городского управления, в частности – в экосистему цифрового двойника «Мета-Москва». В ближайшей перспективе планируется адаптация Siuuu под запросы, валидация на отечественных снимках. Долгосрочная цель – создание легковесного, автономного и безопасного инструмента для оперативного семантического мониторинга городской среды, способного без дообучения интерпретировать динамику урбанизации, выявлять аномалии и генерировать аналитические отчёты, что соответствует стратегии ДИТ Москвы по развитию городского общего интеллекта.

1. АНАЛИЗ СУЩЕСТВУЮЩИХ ТЕХНОЛОГИЙ ДЛЯ РЕШЕНИЯ ЗАДАЧЧИ ОБНАРУЖЕНИЯ ИЗМЕРЕНИЙ

1.1 ОБЗОР ТРАДИЦИОННЫХ ПОДХОДОВ

Современные методы обнаружения изменений на разновременных снимках дистанционного зондирования можно разделить на несколько основных категорий, каждая из которых имеет свои характерные особенности и ограничения. Как показано на рисунке 1, существующие подходы сталкиваются с фундаментальными проблемами, препятствующими их эффективному применению в реальных сценариях комплексного анализа городской среды.

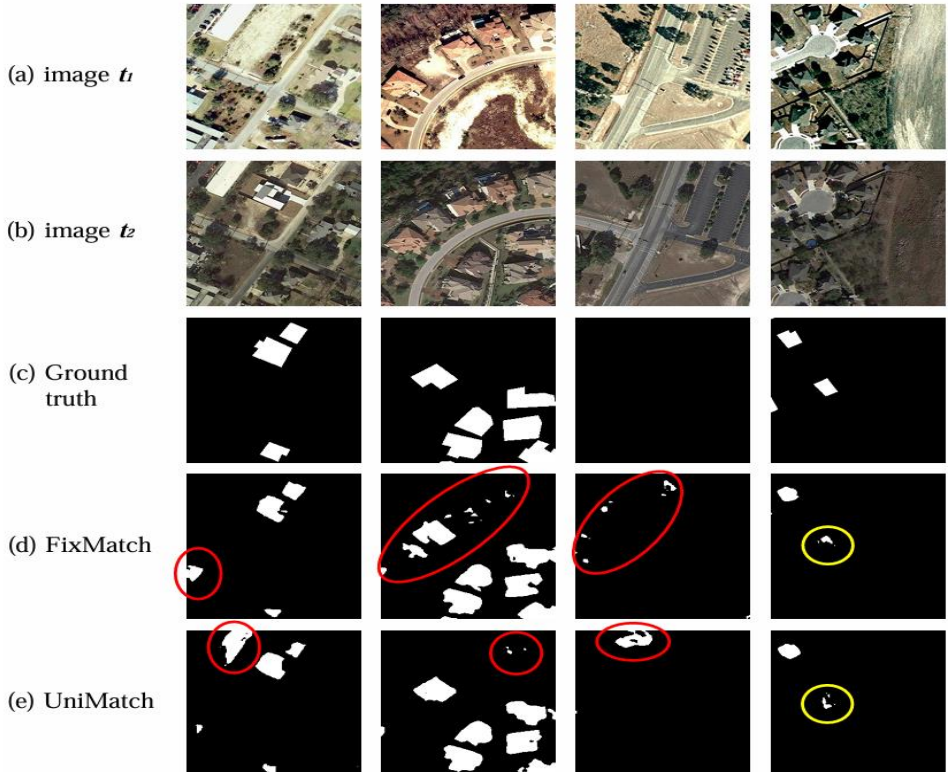


Рис. 1. Некоторые недостатки существующих решений

Алгебраические методы и метод главных компонент (PCA) представляют собой наиболее простой класс подходов к обнаружению изменений. Эти методы генерируют исключительно бинарные карты изменений, указывая лишь на сам факт наличия различий между снимками t_1 и t_2. Однако такая информация оказывается недостаточной для практического применения, поскольку не содержит никакой семантической интерпретации – невозможно понять, что именно изменилось: появились ли новые здания, изменилась ли дорожная сеть или произошла трансформация растительного покрова. Визуализация результатов (Рисунок 1, строки a-b) наглядно демонстрирует это ограничение: мы видим лишь области изменений без какой-либо информации об их природе [5].

Методы глубокого обучения, включая сверточные нейронные сети (CNN) и трансформер-архитектуры (BIT, SAM), представляют собой следующий уровень развития

технологий детекции изменений. Эти подходы демонстрируют высокую точность обнаружения изменений, что подтверждается сравнением с эталонной разметкой (рисунок 1, строка c - Ground truth). Однако, несмотря на превосходные количественные показатели, данные методы работают по принципу «черного ящика», предоставляя лишь пиксельные маски изменений без возможности семантической интерпретации. Они не способны отвечать на аналитические вопросы, требующие многоэтапного рассуждения, классификации типов изменений или адаптации к новым категориям объектов без полного переобучения модели.

Некоторые методы, например, FixMatch и UniMatch сталкиваются с рядом других проблем. Требуется обширная ручная разметка для каждого нового типа объектов, и, как показано на примерах (рисунок 1, строки d-e, где красные и желтые круги выделяют области ошибок), даже при относительно точной детекции границ изменений ни UniMatch, ни аналогичные методы не способны объяснить семантическую природу обнаруженных изменений [6].

В задаче обнаружения изменений семантическая сегментация критически важна для количественной оценки пространственных характеристик изменений – площади, формы, степени изменения объектов. HQ-SAM расширяет базовую модель SAM за счёт обучаемого токена высокого качества, интегрируемого в декодер масок и обеспечивающего точное восстановление детализированных границ. Слияние признаков раннего и финального уровней ViT-энкодера с токеном вывода – позволяет сегментировать сложные структуры даже при тонких различиях между разновременными снимками. Преимущества для задачи обнаружения изменений: гибкость промптинга (точечные, боксовые, текстовые запросы), возможность применения без дообучения. Ограничения: отсутствие встроенной поддержки анализа и описания изменений [7].

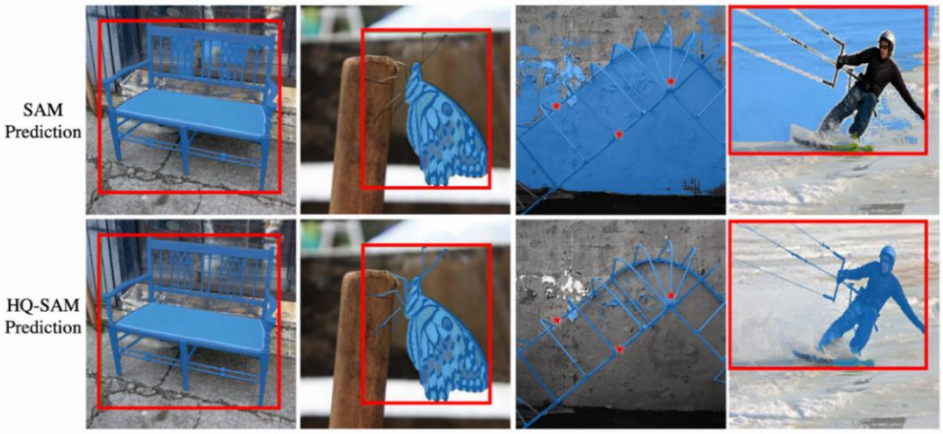


Рис. 2. Работа моделей HQ-SAM и SAM

Для задачи обнаружения изменений YOLO11 с поддержкой ориентированных ограничивающих рамок (OBB) реализует одноэтапную архитектуру детектирования, оптимизированную для аэрокосмических снимков с объектами под произвольными углами.

Модель обучена на наборе данных DOTA v1.0. Ключевое преимущество YOLO11 OBB – детектирование объектов с учётом ориентации, повышающее точность локализации зданий, транспорта и инфраструктуры по сравнению с традиционными. Ограничения для обнаружения изменений: представление объектов рамками не позволяет оценивать изменения площади или формы; отсутствие механизмов сравнения разновременных пар [8].

Рис. 3. Использование YOLO11 для обнаружения резервуаров

Традиционные подходы к детекции изменений на разновременных снимках дистанционного зондирования можно систематизировать в несколько основных категорий, каждая из которых обладает специфическими характеристиками, преимуществами и ограничениями. Понимание этих методов необходимо для обоснования перехода к более совершенным агентным архитектурам, способным преодолеть выявленные ограничения. Ниже представлена сравнительная характеристика основных подходов к детекции изменений.

Таблица 1. Сравнение традиционных методов обнаружения изменений

Категория методов	Конкретные методы	Преимущества	Ограничения
Алгебраические методы	Разность изображений, регрессия, отношение, анализ векторов изменений (CVA)	Простота реализации и вычислительная эффективность	Чувствительность к атмосферным условиям, сезонным изменениям, характеристикам сенсоров
Методы трансформации	PCA (метод главных компонент), MAD (метод многомерных изменений), ортогонализация Грама–Шмидта	Снижение корреляции между каналами, акцентирование вариаций, уменьшение размерности данных	Потеря интерпретируемости компонентов, сложность физического толкования результатов
Методы на основе классификации	Постклассификационное сравнение, прямая классификация (YOLO / YOLO11 OBB)	Семантическая интерпретация результатов, возможность анализа типов изменений	Накопление ошибок классификации, зависимость от точности предварительной

			классификации
Глубокое обучение	CNN, RNN, автоэнкодеры, GAN, трансформеры (BIT), SAM, CLIP	Высокая точность детекции, автоматическое извлечение признаков, устойчивость к шумам	Фокус на одном типе изменений, отсутствие интеллектуальной аналитики, необходимость больших обучающих выборок

Анализ представленных методов позволяет выделить три фундаментальные проблемы, препятствующие их эффективному применению в реальных сценариях мониторинга городской среды.

Первая проблема заключается в полном отсутствии семантической интерпретации результатов. Ни один из рассмотренных методов не может ответить на вопрос «что именно изменилось?». Даже при высокой точности детекции границ изменений (как показано на примере Ground truth в Рисунок 1, строка c), системы не способны классифицировать тип изменений – является ли обнаруженное изменение новым зданием, дорогой, вырубкой леса или сезонными изменениями растительности.

Вторая проблема связана с необходимостью дообучения модели для каждого нового класса объектов. Существующие подходы жестко привязаны к категориям объектов, на которых они были обучены. При появлении нового типа изменений или объектов требуется создание размеченного датасета и полное переобучение модели, что делает их экономически неэффективными для реальных приложений, где необходимо оперативно анализировать разнообразные изменения в динамически развивающейся городской среде.

Третья проблема обусловлена ограничением работы только с парой снимков (t_1 и t_2), что исключает возможность учета исторического контекста. Такой подход не позволяет отслеживать долгосрочные тренды изменений, отличать циклические (сезонные) изменения от устойчивых трендов и строить прогнозы дальнейшего развития ситуации.

Несмотря на выявленные ограничения, архитектуры Ultralytics YOLO11 и HQ-SAM обладают значительным потенциалом для преодоления указанных проблем в составе комплексных систем обнаружения изменений. Ultralytics YOLO11 с поддержкой ориентированных ограничивающих рамок, обеспечивая точное детектирование объектов под произвольными углами, может служить надёжным модулем первичной локализации изменений с высокой скоростью выполнения вывода, что критично для оперативного мониторинга. HQ-SAM, в свою очередь, благодаря механизму сегментации высокого качества в режиме нулевого выстрела и гибкости формирования запросов, способен обеспечивать детализированное выделение границ объектов без предварительного обучения, что снижает зависимость от

размеченных наборов данных. Их совместное применение в модульной архитектуре – где Ultralytics YOLO11 отвечает за быстрое обнаружение кандидатов на изменения, а SAM – за точную семантическую сегментацию и верификацию – открывает путь к созданию интерпретируемых, масштабируемых и адаптивных систем анализа изменений, способных работать с разновременными последовательностями и новыми классами объектов без полного переобучения.

1.2 ИССЛЕДОВАНИЕ СПОСОБНОСТИ БОЛЬШИХ ВИЗУАЛЬНЫХ ЯЗЫКОВ МОДЕЛЕЙ РЕШАТЬ ЗАДАЧИ АНАЛИЗА ИЗМЕНЕНИЙ

Проведён сравнительный эксперимент с открытыми большими языковыми моделями (БЯМ), развёрнутыми локально в квантованном формате. Экспериментальная оценка осуществлялась на наборе из 140 вопросов, охватывающих сценарии развития городской территории Москвы за период 2010–2025 годов, включая пары спутниковых снимков, сделанных для одной территории с разные интервалы времени. Снимки были подготовлены на городских районов Москвы с помощью онлайн-сервиса GoogleMaps в режиме спутниковой съемки (примеры приведены в Приложении 1).

Условия эксперимента:

- развёртывание: локальное, без использования облачных API (библиотека transformers);
- квантование: INT4 для оптимизации требований к памяти;
- оборудование: GPU с 8 ГБ видеопамяти (RTX 2070)
- входные данные: спутниковые снимки GoogleMaps (100 снимков);
- метрики оценки: точность количественных оценок, устойчивость к формулировкам промпта, поддержка нескольких снимков.

Для эксперимента были выбраны следующие модели.

Qwen2.5-VL-3B – архитектура Transformer с Qwen-VL визуальным энкодером. Модель поддерживает последовательный ввод нескольких снимков через промпт, обеспечивая высокую семантическую интерпретацию при корректной формулировке запроса. Компактный размер (3B) делает модель пригодной для локального развёртывания без необходимости дообучения [9].

LLaVA-OneVision-7B – архитектура LLaMA + CLIP-ViT. Модель демонстрирует ограниченную поддержку нескольких снимков через поочерёдную подачу изображений, показывает среднюю чувствительность к формулировкам промпта. Ключевая особенность – поддержка видеовхода, что открывает потенциал для анализа временных рядов. Стабильная работа обеспечивается при последовательной подаче изображений [10].

InternVL3-8B – архитектура Vision Transformer + InternLM. Модель демонстрирует

явную поддержку парных и множественных изображений, высокую интерпретацию изменений благодаря обучению на геопространственных и дистанционных данных. Признана одной из лучших среди открытых решений для задач дистанционного зондирования. Обеспечивает устойчивость к вариациям масштаба и разрешения снимков. Ограничения включают высокие требования к видеопамяти (8В параметров) и ограниченную документацию для кастомизации [11].

Модель Florence, разработанная Microsoft, представляет собой семейство больших визуально-языковых моделей, которые объединяют понимание изображений и текста в единой архитектуре на основе визуального энкодера DaViT и мультимодального трансформера «энкодер-декодер». Её преимущество заключается в унифицированном подходе к обработке промптов и обучении на наборе данных FLD-5B, что позволяет модели эффективно работать в режиме без обучения, выполняя различные задачи – от детекции объектов до сегментации – без необходимости предварительного дообучения, при этом компактность архитектуры (модели до 0,77 млрд параметров) сохраняет конкурентоспособность с более крупными аналогами. Однако главный недостаток Florence применительно к обнаружению изменений на спутниковых снимках заключается в её общей природе: будучи ориентированной на бытовые изображения, модель не учитывает специфику разновременных геопространственных данных, не поддерживает механизмы рассуждения о временных изменениях и, как следствие, требует либо дообучения под задачу анализа разновременных снимков, либо интеграции в более сложные агентные архитектуры [12].

Приведем результаты одного из примеров исследований. Применение Qwen2.5-VL для анализа трансформации городского района (2019–2025) продемонстрировало способность модели генерировать структурированные отчёты с количественными оценками:

Объект	Изобр. 1, 2019	Изобр 5, 2025
Жилые здания	8	16
Офисные здания	5	13
Здания под строительством	2	1
Строительные краны	3	2
Парковочные зоны	6	14
Автомобили	~1.2	~2.8
Зелёные насаждения	4	8
Расчищенные участки	12	1
Дорожная инфраструктура	2	10
Транспортные потоки	Умеренные	Очень плотные

Район претерпел полную трансформацию: от пустыря — к современному городскому кварталу.
Основные изменения:

1. Увеличение жилой и коммерческой застройки на +100% (по количеству зданий)
2. Развитие инфраструктуры — +400% по дорожным элементам
3. Улучшение экологического состояния — +100% по зелёным зонам
4. Рост транспортной активности — +133% по автомобилям

Основная проблема – галлюцинации!!!

Рис. 4. Пример исследования с применением Qwen2.5-VL, район метро Яхромская, Москва

Проведённое экспериментальное исследование выявило критическую проблему: галлюцинации при количественной оценке изменений. Модель склонна завышать или неточно интерпретировать количественные показатели без инструментальной проверки. Отсутствует семантическое понимание того, что составляет «изменение» в контексте дистанционного зондирования, и неспособность применять пространственную логику и пороговые значения.

Таблица 2. Оценка результатов работы модели Qwen2.5-VL

Показатель	**Оценка модели**	**Проверка (YOLO11 OBB + SAM)**	**Расхождение**
Жилая застройка	100%	45%	55%
Инфраструктура	400%	120%	280%
Зелёные насаждения	100%	60%	40%
Транспортная активность	133%	75%	58%

При количественной оценке результатов, помимо фундаментальной проблемы галлюцинаций, БВЯМ демонстрируют ряд ограничений, снижающих их эффективность при обнаружении изменений на снимках. Так, модели не умеют работать с локальными областями: они не различают полное изображение и его фрагмент, что мешает детальному анализу нужных участков. Кроме того, качество их ответов сильно зависит от формулировки запроса – даже небольшие изменения в промпте могут привести к разным результатам, что создает нестабильность. Также модели не поддерживают многошаговые вычисления, то есть не способны сохранять и использовать промежуточные данные, что необходимо для сложного анализа. Наконец, в них отсутствуют встроенные механизмы проверки достоверности ответов, что критично для задач, требующих высокой точности, например при мониторинге инфраструктуры или в геопространственной разведке.

Для преодоления выявленных ограничений были разработаны агентные подходы, интегрирующие БЯМ со специализированными инструментами компьютерного зрения.

Проанализировав существующие разработки, можно сделать следующий вывод: несмотря на наличие множества агентных систем, интегрирующих большие языковые модели с инструментами компьютерного зрения, ни одна из них не готова к эффективному решению задачи обнаружения изменений на спутниковых снимках. Это обуславливает необходимость создания принципиально новой архитектуры, и вот почему.

1. Существующие геопространственные модели решают смежные, но не целевые задачи. Ряд моделей разработан специально для работы со снимками, однако они не покрывают ключевую потребность – анализ изменений во времени.

SkyEyeGPT ориентирован на описание одиночных изображений и геопространственный анализ, но в них отсутствуют механизмы рассуждения о временных изменениях [13].

EarthGPT имеет ограниченную поддержку пар разновременных снимков (би-темпоральных данных) либо страдают низкой интерпретируемостью результатов, что критично для понимания причин изменений [14].

2. Существующие решения для Change Detection (обнаружения изменений) негибки. На сегодняшний день единственная система, нацеленная именно на анализ изменений – Change-Agent, – имеет фундаментальные архитектурные ограничения [15]. Она построена на обучении с учителем и привязана к конкретному набору данных (LEVIR-MCI). Это означает, что для обнаружения каждого нового типа объектов (например, не только строений, но и вырубок леса или подтоплений) модель приходится переобучать заново. В ней отсутствует модульность и возможность гибкой кастомизации под новые классы изменений без вмешательства в исходный код и повторного дорогостоящего обучения.

Сравнительный анализ существующих решений приведен в Приложении 2.

В результате проведённого анализа современного состояния в области больших визуально-языковых моделей (БВЯМ) и их применения для задач обнаружения изменений на спутниковых снимках, пришли к заключению, что ни одна из существующих систем не сочетает в себе три критически важных свойства: понимание специфики разновременных спутниковых данных, способность к логическим рассуждениям об изменениях и гибкую модульную архитектуру, позволяющую адаптировать модель под новые типы изменений без её полного переобучения. Именно это технологическое окно и призвана заполнить новая разработка.

1.3 ПОСТАНОВКА ЗАДАЧИ И ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА РАЗРАБОТКУ

На основании п. 1.1 и 1.2 формулируется научно-техническая задача: разработать архитектуру агентной системы для интеллектуального анализа изменений на разновременных спутниковых снимках, способную принимать запросы на естественном языке, автоматически декомпозировать сложные задачи на последовательность вычислительных шагов, привлекать специализированные инструменты компьютерного зрения и формировать интерпретируемые отчёты.

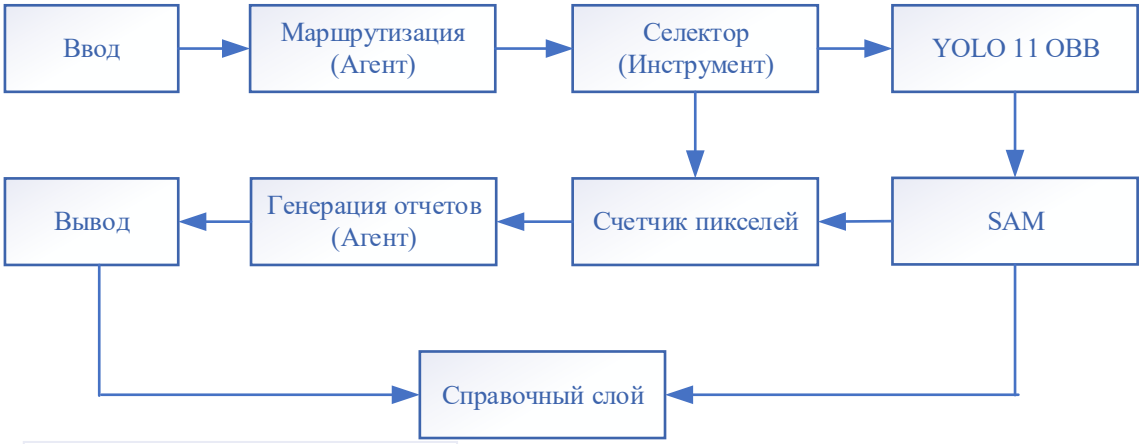
ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА РАЗРАБОТКУ СИСТЕМВ ОБНАРУЖЕНИЯ ИЗМЕНЕНИЙ

1. Цель: разработать систему анализа изменений на разновременных спутниковых снимках городской среды.
2. Основание для разработки: необходимость преодоления ограничений существующих систем в задачах геопространственного мониторинга.
3. Требования к функциональным характеристикам. Система должна обеспечивать решение следующих задач:
   - 3.1. Принимать на вход пару разновременных спутниковых снимков в формате RGB, а также текстовый запрос пользователя на русском языке.
   - 3.2. Автоматически декомпозировать сложный запрос на последовательность действий. Система должна определять, требуется ли для ответа подсчёт объектов, оценка площади, сравнение классов или другой вид анализа.
   - 3.3. Выполнять детекцию объектов с использованием модели YOLO11 OBB для подсчёта элементов (здания, транспорт, инфраструктура).
   - 3.4. Выполнять сегментацию с использованием модели Segment Anything для оценки площади (насаждения, застройка и др.).
   - 3.5. Использовать результаты работы инструментов (числа, проценты, площади) для ответа, исключая прямую генерацию числовых значений БВЯМ.
   - 3.6. Генерировать ответ, содержащий текстовое заключение и подтверждённые количественные данные, полученные в результате работы инструментов.
4. Требования к архитектуре и технологическому стеку
   - 4.1. Ядро системы (оркестратор): использование большой визуально-языковой модели Qwen2VL (в квантовании INT4) для понимания запросов и планирования. Инференс осуществляется локально через transformer.
   - 4.2. Управление памятью: применение LlamaIndex для векторного хранения истории диалога и извлечения релевантного контекста.
   - 4.3. Набор инструментов: YOLO11 OBB – для детекции и подсчёта объектов; Segment Anything (SAM) – для сегментации и расчёта площадей.
   - 4.4. Пользовательский интерфейс: реализация веб-интерфейса Gradio.
   - 4.5. Локализация: поддержка русского языка в промптах и интерфейсе.
5. Требования к техобеспечению – локальное развёртывание на ПК:
   - 5.1. GPU: 8 ГБ видеопамяти (NVIDIA RTX 2070 или аналог);
   - 5.2. Оперативная память: от 16 ГБ;
   - 5.3. Хранение: от 20 ГБ свободного дискового пространства.

2.СПОСОБ ОБНАРУЖЕНИЯ ИЗМЕНЕНИЙ НА ИЗОБРАЖЕНИЯХ С ПРИМЕНЕНИЕМ БВЯМ И АГЕНТОВ

2.1. ОБЩАЯ АРХИТЕКТУРА ПОДХОДА

Подход основан на агентной архитектуре, включающей большие языковые модели (LLM) со специализированными инструментами компьютерного зрения для анализа изменений на разновременных снимках дистанционного зондирования. Ключевым элементом подхода является модуль агент (планировщик), координирующий взаимодействие между языковой моделью и инструментами визуального анализа (рисунок 5).



ДАННЫЕ НА КАЖДОМ ЭТАПЕ:

1. Ввод: {Ipre, Icur, Q}
2. Маршрут: {тип_запроса, сложность, требуемые_инструменты}
3. Селектор: {последовательность: [YOLO, SAM, Счётчик]}
4. YOLO: {детекция_2009: 147, детекция_2020: 203}
5. SAM: {площадь_2009: 17.4%, площадь_2020: 25.9%}
6. Счётчик: {Δ : 5553, Δ: 48.6%}
7. Генератор отчетов: {финальный_ответ: " ОТЧЁТ ОБ ИЗМЕНЕНИЯХ..."}
8. Вывод: {текстовый_отчёт, карта_изменений}

Рис. 5. Структурная схема системы обнаружения изменений

Пользователь загружает разновременные изображения и формулирует запрос. Маршрутизатор классифицирует запрос и определяет тип задачи. Селектор инструментов формирует последовательность выполнения: [YOLO11 OBB, SAM, Подсчет пикселей]. Инструмент YOLO11 OBB выполняет детекцию объектов на обоих снимках, генерируя результаты (например, 147 зданий (2009) и 203 здания (2020)). Параллельно инструмент SAM выполняет семантическую сегментацию, определяя доли классов. Счётчик пикселей вычисляет абсолютное изменение площади. Агент генерации отчетов агрегирует все промежуточные результаты и синтезирует финальный ответ на естественном языке. Все данные сохраняются в справочном слое. Пользователь получает три формы представления результатов: текстовый отчёт, карту изменений.

2.2 ФОРМИРОВАНИЕ ЗАДАЧИ И ОПИСАНИЕ ПРИЗНАКОВОГО ПРОСТРАНСТВА

Для формализации задачи введем признаковое пространство $\boldsymbol{\chi}$, в котором работает агентная система. Входные данные представляются как кортеж:

$$x \in \boldsymbol{\chi} = \{\boldsymbol{I_{pre}}, \boldsymbol{I_{cur}}, \boldsymbol{Q}, \boldsymbol{H} \quad (1)$$

где:

$\boldsymbol{I_{pre}}, \boldsymbol{I_{cur}} \in \mathbb{R}^{\boldsymbol{H}\times\boldsymbol{W}\times 3}$ – разновременные RGB-изображения дистанционного зондирования (предыдущее и текущее);

$\boldsymbol{Q} \in \boldsymbol{V}$ – текстовый запрос пользователя на естественном языке (пространство вопросов V);

$H = \{(Q_1, A_1), \ldots, (Q_{k-1}, A_{k-1})\}$ – история диалога, обеспечивающая контекстную связность.

Целью агента является отображение $\boldsymbol{\Phi}$: $\boldsymbol{\chi} \rightarrow$**A**, где **A** – пространство аналитических ответов. Однако прямое отображение $\boldsymbol{\Phi}$ затруднено из-за модального разрыва между текстом и изображением. Поэтому вводится пространство инструментов **T**={T_1,...,T_n}}, где каждый инструмент T_i является функцией обработки изображений:

$$\boldsymbol{T_i} = \mathbb{R}^{\boldsymbol{H}\times\boldsymbol{W}\times 3} \rightarrow O_i$$

где O_i – пространство выходных данных инструмента (маски сегментации, bounding boxes, количественные метрики).

Введение пространства инструментов **T** позволяет декомпозировать сложную задачу анализа изменений на последовательность подзадач. Каждая подзадача решается специализированным инструментом, что обеспечивает точность вычислений и снижает вероятность галлюцинаций. Модуль планирования N координирует выбор и последовательность инструментов, обеспечивая семантическую согласованность между запросом пользователя и выполняемыми операциями.

Работа агента моделируется как итеративный процесс. В раунде h ответ A_i^h формируется на основе работы планировщика N и языковой модели L:

$$A_i^h = L(N(P), N(F), N(H_{<i}), N(Q_i), N(R_i^{<i}), N(F/A_i^{<i})) \quad (2)$$

где:

- L – языковая модель (бэкенд, Qwen2VL);
- N – модуль планировщик;
- P – системные промпты, определяющие роль и ограничения;
- F∈T – выбранный инструмент;
- $R_i^{<i}$ – результаты рассуждения и выполнения инструментов на предыдущих шагах.

Уравнение (2) формализует процесс принятия решений агентом как функцию от шести компонентов:

- *N(P)* – системные ограничения обеспечивают соблюдение роли агента и предотвращают галлюцинации;
- *N(F)* – выбор инструмента определяет вычислительные возможности на текущем шаге;
- $N(H_{<i})$ – история диалога обеспечивает контекстную связность многошаговых запросов;
- $N(Q_i)$ – текущий запрос определяет целевую задачу анализа;
- $N(R_i^{<i})$ – промежуточные результаты позволяют накапливать знания в ходе выполнения;
- $N(F/A_i^{<i}))$ – обратная связь от предыдущих действий обеспечивает адаптивность планирования.

Такая формализация позволяет агенту выполнять многошаговые рассуждения с верификацией каждого шага через инструменты, что критически важно для задач количественного анализа изменений.

Системные промпты P формализуются как вектор компонентов, обеспечивающих управление поведением модели:

$$P=(Pprefix, Pimage, Preference, Pformat, Psuffix) \quad (3)$$

где:

- *Pprefix* – определение роли и ограничений агента (например, «Вы – ассистент для анализа изменений на снимках ДЗЗ...»);
- *Pimage* – определение роли и ограничений агента (например, «Суффиксы `_pre`, `_cur`, `_crppre`»);
- *Preference* – уникальная идентификация изображений (например, «image/904796_be9519_landuse.png»);
- *Pformat* – формат вывода и протокол рассуждений (например, Мысль/Действие/Наблюдение/Ответ);
- *Psuffix* – дополнительные ограничения (например, «Никогда не выдумывайте несуществующие файлы»).

Разделение промптов на пять компонентов позволяет модульно управлять поведением агента. P_{prefix} и P_{suffix} обеспечивают соблюдение ограничений, P_{image} и $P_{reference}$ гарантируют точную привязку к визуальным данным, а P_{format} стандартизирует вывод для последующей обработки и управления вызовом инструментов. Каждый компонент обеспечивает управление

определённым аспектом поведения агента (роль, идентификация изображений, метаданные, формат вывода, ограничения).

3. РЕВЛИЗАЦИЯ АЛГОРИТМА ОБНАРУЖЕНИЯ ИЗМЕНЕНИЙ С ПРИМЕНЕНИЕМ АГЕНТНОГО ПОДХОДА И БВЯМ

3.1 ТЕХНОЛОГИЧЕСКИЙ СТЕК

PyCharm Professional/Community Edition выбрана в качестве основной интегрированной среды разработки (IDE) для реализации системы Siuuu. Структура проекта в PyCharm:

```
Siuuu /
├── .venv/                    # Виртуальное окружение Python
├── src/
│   ├── agent/
│   │   ├── controller.py         # AgentController (управление жизненным циклом)
│   │   ├── navigator.py          # PlanningNavigator (планировщик)
│   │   ├── router.py             # Агент-маршрутизатор
│   │   ├── selector.py           # Агент-селектор инструментов
│   │   └── reporter.py           # Агент-отчётчик
│   ├── toolkit/
│   │   ├── yolo11_obb.py         # Детекция объектов
│   │   ├── sam_segmentation.py   # Сегментация изображений
│   │   ├── pixel_counter.py      # Подсчёт пикселей
│   │   └── change_detector.py    # Бинарная детекция изменений
│   ├── memory/
│   │   └── llama_index_store.py  # Хранилище памяти на базе LlamaIndex
│   ├── interface/
│   │   └── gradio_ui.py          # Пользовательский интерфейс
│   ├── utils/
│   │   ├── image_registry.py     # Уникальная идентификация изображений
│   │   ├── prompt_templates.py   # Шаблоны промптов (P_prefix, P_image, etc.)
│   │   └── metrics.py            # Метрики оценки (Precision, Recall, Match)
│   └── main.py                   # Точка входа в приложение
├── data/
│   ├── images/                   # Разновременные снимки (_pre, _cur)
│   ├── masks/                    # Маски сегментации
│   └── reports/                  # Сгенерированные отчёты
```

```
├── tests/
│   ├── test_agent.py          # Тесты агента
│   ├── test_toolkit.py        # Тесты инструментов
│   └── test_interface.py      # Тесты интерфейса
├── requirements.txt           # Зависимости проекта
├── config.yaml                # Конфигурация системы
└── README.md                  # Документация проекта
```

Таблица 3. Сводная таблица библиотек и моделей

Компонент	Технология	Назначение
IDE	PyCharm	Среда разработки
Язык	Python	Основной язык программирования
Интерфейс	Gradio	Веб-интерфейс пользователя
LLM (локально)	Qwen2VL + Transformers	Локальная мультимодальная модель
Память	LlamaIndex	Векторное хранилище контекста
Векторная БД	ChromaDB	Хранение эмбеддингов
Детекция	YOLO11	Обнаружение объектов с OBB
Сегментация	SAM	Пиксельная сегментация
Обработка изображений	OpenCV, Pillow	Предобработка изображений
Вычисления	NumPy, PyTorch	Численные операции и инференс

В Приложении представлена UML-диаграмма классов.

Центральный класс AgentController (КонтроллерАгента) управляет жизненным циклом запроса и содержит методы: initialize() для инициализации системы, process_query() для обработки запроса и finalize() для завершения сессии.

Класс PlanningNavigator (НавигаторПланирования) реализует функционал модуля планирования и включает методы: decompose_task() для декомпозиции сложных запросов на подзадачи, select_tools() для выбора оптимальной последовательности инструментов и validate_tools() для валидации результатов.

Абстрактный класс ToolInterface (ИнтерфейсИнструмента) определяет общий интерфейс для всех инструментов и содержит методы: execute(image, params) для выполнения

инструмента, validate() для проверки входных данных и get_output() для получения результатов. От этого класса наследуются конкретные реализации: YOLO11OBB_Tool с методами detect_objects() и count_objects(), SAM_Tool с методами segment_area() и calculate_pixels(), а также класс PixelCounter (СчётчикПикселей) с методами count() и compare().

Класс MemoryStore (ХранилищеПамяти) представляет обёртку над векторной базой LlamaIndex и реализует методы: save_history() для сохранения истории диалога, retrieve_context() для извлечения контекста и clear_cache() для очистки кэша.

Класс GradioUI (ПользовательскийИнтерфейс) отвечает за взаимодействие с пользователем и содержит методы: render_input() для отображения элементов ввода, display_output() для визуализации результатов и download_results() для экспорта данных.

Связи между классами определены следующим образом: AgentController агрегирует PlanningNavigator, PlanningNavigator композирует ToolInterface, AgentController ассоциирован с MemoryStore, а GradioUI зависит от AgentController.

3.2 АЛГОРИТМ ОБНАРУЖЕНИЯ ИЗМЕНЕНИЙ

Процесс функционирования агентной системы Siuuu инициируется на прикладном уровне через пользовательский интерфейс Gradio, где осуществляется первичная регистрация входных данных. Как показано на рисунке 6, система принимает три основных компонента: разновременные изображения дистанционного зондирования (*Ipre*, *Icur*) размерности 512×512×3 и 512×512×3 (RGB-пространство), а также текстовый запрос QQ на естественном языке, сформулированный пользователем. Примером такого запроса может служить: «Какие изменения произошли в застройке?».

Критически важным аспектом на данном этапе является обеспечение уникальной идентификации изображений, что реализуется через модуль планирования. Каждому изображению присваивается уникальный идентификатор с временными суффиксами: '_pre' для предыдущего снимка и 'cur' для текущего. Например, файлы именуются как 'qianhai_2009_pre.png' и 'qianhai_2020_cur.png', где метаданные включают хэш-суммы файлов, временные метки и пути к данным. Эта система именования, описанная в формуле (2) через компонент P_{image}, обеспечивает семантическую согласованность на всех последующих этапах обработки и предотвращает неоднозначность ссылок в многошаговых рассуждениях.

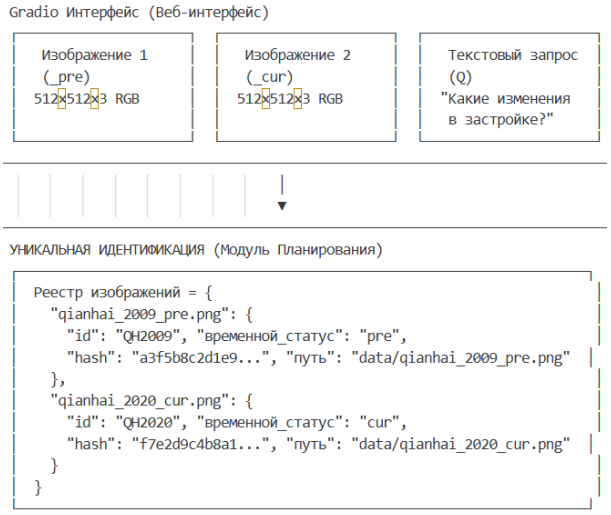


Рис. 6. Схема входных данных и потока информации

На рисунке 7 показана структурная схема агентов, состоящую из трех функциональных уровней, что соответствует математической модели, представленной в уравнении (1). На вершине иерархии расположен прикладной уровень (Gradio UI), обеспечивающий взаимодействие с пользователем и поддержку многошаговых диалогов. Центральным элементом архитектуры является навигатор планирования, который координирует работу трех специализированных агентов: агента-маршрутизатора, агента-селектора инструментов и агента-генератора отчетов.

Навигатор планирования интегрирует три функциональных слоя: уровень понимания задачи пользователя, справочный слой (хранилище памяти) и слой планирования. Уровень понимания отвечает за классификацию запроса и идентификацию изображений, используя компоненты промптов *Pprefix*, *Pformat* и *Psuffix* из уравнения (2). Справочный слой обеспечивает хранение истории диалога $H<i$, промежуточных результатов и цепочки рассуждений, что критически важно для объяснимости системы. Слой планирования реализует декомпозицию сложных задач на подзадачи и выбор оптимальной последовательности инструментов.

Исполняющий уровень содержит набор специализированных инструментов компьютерного зрения: YOLO11 OBB для детекции объектов, SegAnything (SAM) для пиксельной сегментации областей и PixelCounter для количественных расчётов. Результаты обработки передаются агенту-отчётчику, который синтезирует финальный ответ на естественном языке и формирует визуализацию изменений.

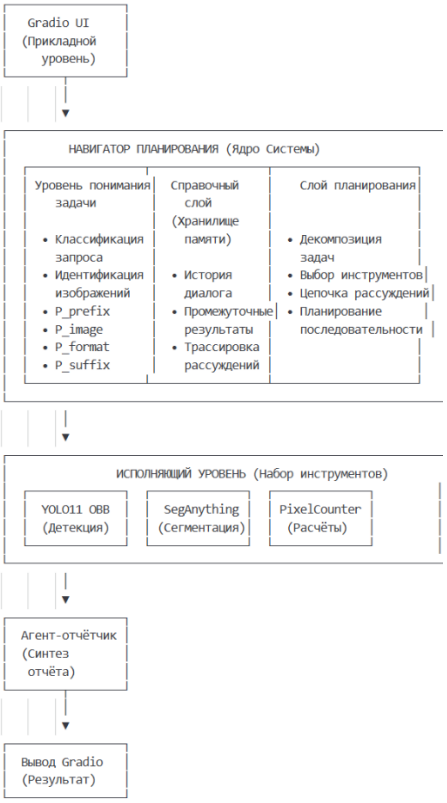


Рис. 7. Схема агентной архитектуры

На рисунке 8 показана схема работы первого агента системы – маршрутизатора, который выполняет первичную классификацию пользовательского запроса. На вход агент получает текстовый запрос, например: «Какие изменения произошли в застройке между этими снимками?». Используя промпт классификации на русском языке, агент анализирует семантику запроса и относит его к одному из четырёх типов, определённых в Таблице 3.

Таблица 3. Типы задач

Тип вопроса	**Инструменты**	**Выходные данные**	**Примеры вопросов**
Whether (есть ли изменения)	Binary Change Detection (SAM & CLIP)	Бинарная маска изменений	Есть ли заметные различия между изображениями, указывающие на изменения?
Size (площадь изменений)	Semantic Segmentation → Pixel Counting → Calculation	Процентное изменение площади	Оцените процент изменённой площади относительно общего размера изображения. Какая доля водных объектов увеличилась или уменьшилась в размере, выраженная в процентах?

			На локализованном участке, который я вырезал, какой процент площади претерпел изменения?
Number (количество объектов)	Object Detection (YOLO11 OBB) → Object Counting → Comparison	Дельта количества объектов	Между предыдущим и текущим изображениями произошло ли увеличение или уменьшение количества судов? Рассчитать изменение количества самолётов между двумя изображениями?
Class (класс изменений)	Scene Classification → Semantic Segmentation → Analysis	Отчёт о переходах классов	Какой класс покрывал всю область предыдущего изображения, и к какому классу теперь относится вся область?

В рассматриваемом примере запрос классифицируется как комбинированный тип «Number+Class» со средней сложностью (Medium), поскольку требует как подсчёта зданий, так и анализа класса землепользования. Выходные данные агента представляются в формате JSON и включают тип запроса, оценку сложности, перечень требуемых инструментов и обоснование выбора. Эта классификация определяет дальнейшую стратегию обработки и влияет на выбор инструментов из набора T, как указано в уравнении (1) через компонент F.

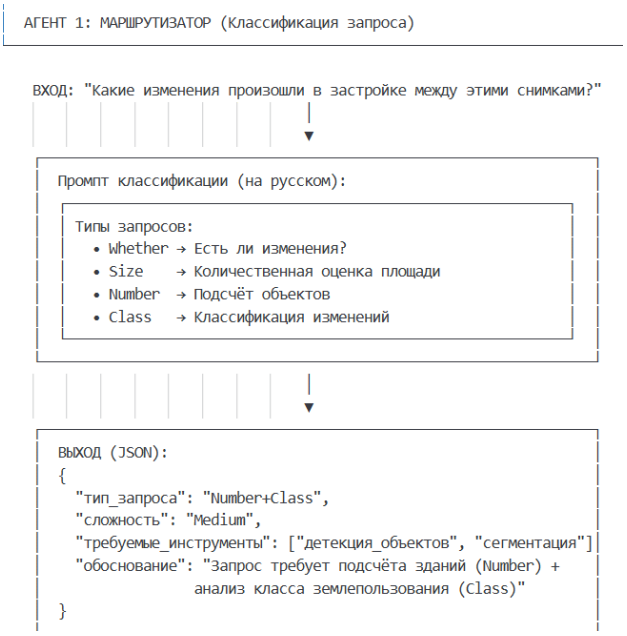


Рис. 8. Схема работы первого агента системы

Рисунок 9 описывает функционирование второго агента – селектора инструментов, который на основе результатов классификации формирует оптимальную последовательность выполнения операций. На вход агент получает структурированные данные: тип запроса «Number+Class» и оценку сложности «Medium». Используя информацию о доступных инструментах из набора (YOLO11 OBB, SAM, PixelCounter, ChangeDetector), агент планирует многошаговый конвейер обработки.

Для запроса о изменениях в застройке агент выбирает трёхступенчатую последовательность: на первом этапе применяется YOLO11 OBB для детекции и подсчёта зданий на обоих разновременных снимках; на втором этапе используется SAM для сегментации застроенных территорий и оценки площади изменений; на третьем этапе PixelCounter выполняет вычисление процентного изменения площади. Каждый инструмент в последовательности имеет чётко определённые входные параметры и ожидаемые выходные данные, что обеспечивает модульность и заменяемость компонентов без нарушения общей архитектуры.

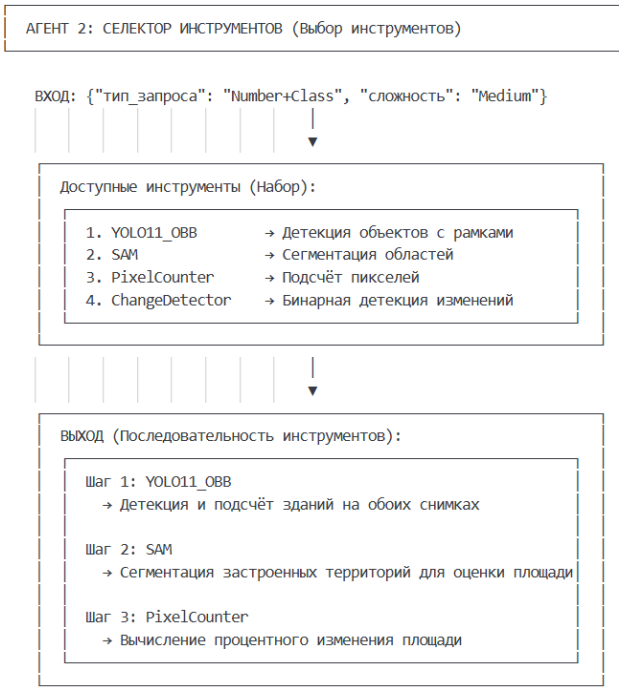


Рис. 9. Схема выбора инструментов (Агент 2: Селектор)

Первый инструмент – YOLO11 OBB (Oriented Bounding Boxes), предназначен для детекции объектов с ориентированными ограничивающими рамками. На вход инструмента подаётся изображение 'qianhai_2009_pre.png' размерности 512×512×3 в RGB-пространстве. Процесс обработки начинается с этапа предварительной обработки изображения: изменение размера с 512×512 до 1024×1024 пикселей, нормализация значений пикселей из диапазона [0-255] в [0.0-1.0] и сохранение цветового пространства RGB. Далее изображение поступает в архитектуру модели YOLO11 OBB.

Выходные данные инструмента представляются в формате JSON и включают массив детекций, где каждая детекция содержит: уникальный идентификатор, класс объекта (boat), оценку уверенности (например, 0.87), параметры ориентированной рамки (центр, ширина, высота, угол поворота) и координаты ограничивающей рамки в формате bbox_xyxy (рисунок 11).

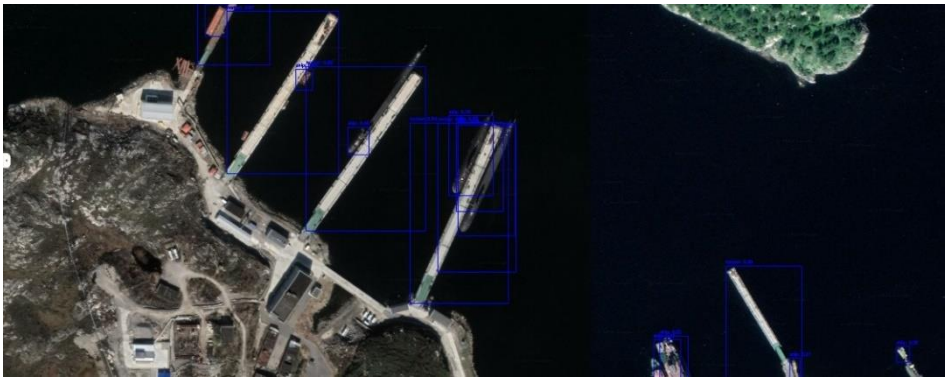


Рис. 10. Пример работы модели YOLO11 OBB (Детекция объектов)

Аналогичная обработка применяется к снимку 2020 года, где обнаружено 20 объектов типа корабль. На основе этих результатов вычисляется изменение:

$\Delta=33-20=13$ кораблей, что соответствует относительному приросту +65%.

На рисунке 11 представлена схема работы второго инструмента – SAM (Segment Anything Model), предназначенного для пиксельной сегментации областей. На вход модели подаётся то же изображение 'qianhai_2009_pre.png'.

```
ИНСТРУМЕНТ: SAM (Segment Anything Model)

ВХОД: qianhai_2009_pre.png (512×512×3 RGB)
                    |
                    ▼
КОДИРОВЩИК ИЗОБРАЖЕНИЯ (ViT-H/16):
  Встраивание патчей: 16×16
  Выход: 256-мерные векторы признаков
  Размер модели: 636 МБ
                    |
                    ▼
КОДИРОВЩИК ПОДСКАЗОК:
  Тип: автоматический (сетка точек)
  Плотность сетки: 32×32 точек
                    |
                    ▼
ДЕКОДИРОВЩИК МАСОК (на основе трансформера):
  Выход: бинарные маски для каждого класса
  Постобработка:
    • Морфологические операции (закрытие, открытие)
    • Удаление малых объектов (< 100 пикселей)
    • Назначение классов на основе контекста
                    |
                    ▼
ВЫХОД (Сегментация 2009 год):
  classes = {
    "building":   {"pixels": 45678, "percentage": 17.4%},
    "road":       {"pixels": 23456, "percentage": 8.9%},
    "vegetation": {"pixels": 89012, "percentage": 34.0%},
    "water":      {"pixels": 34567, "percentage": 13.2%},
    "barren":     {"pixels": 69287, "percentage": 26.5%}
  }
  processing_time: 1247 мс
                    |
                    ▼
ВЫЧИСЛЕНИЕ ИЗМЕНЕНИЯ ПЛОЩАДИ:
  2009: building = 17.4% (11419.5 м²)
  2020: building = 25.9% (16972.5 м²)
  Δ = 25.9% - 17.4% = +8.5 п.п.
  Δ_area = 5553 м² (+48.6%)
```

Рис. 11. Схема работы SAM (Segment Anything Model)

Выходные данные инструмента включают процентное распределение классов землепользования: здания (building) занимают 45,678 пикселей (17.4% от общей площади), дороги (road) – 23,456 пикселей (8.9%), растительность (vegetation) – 89,012 пикселей (34.0%).

Вывод: при обработке снимка 2020 года доля застроенных территорий увеличилась до 25.9% (67,890 пикселей), что соответствует абсолютному приросту площади Δarea=5553 м² или относительному изменению +48.6%.

Рис. 12. Пример сегментации изображений

На рисунке 13 приведена схема цепочки рассуждений (Справочный слой).

```
Шаг 1: Агент-маршрутизатор
  action: "classify_query"
  input: {"query": "Какие изменения в застройке?"}
  output: {"query_type": "Number+Class", "complexity": "Med"}
  timestamp: "14:30:22.145"
                    |
                    ▼
Шаг 2: Селектор инструментов
  action: "select_tools"
  input: {"query_type": "Number+Class"}
  output: {"tools": ["YOLO11_OBB", "SAM", "PixelCounter"]}
  timestamp: "14:30:22.289"
                    |
                    ▼
Шаг 3-6: Исполнитель инструментов (YOLO + SAM)
  action: "execute_yolo_obb" (2009) → 147 зданий
  action: "execute_yolo_obb" (2020) → 203 здания
  action: "execute_sam" (2009) → 17.4% building
  action: "execute_sam" (2020) → 25.9% building
  timestamp: "14:30:22.631" - "14:30:25.505"
                    |
                    ▼
Шаг 7: Счётчик пикселей
  action: "execute_pixel_counter"
  input: {"masks": ["..._seg_pre.png", "..._seg_cur.png"]}
  output: {"delta_absolute_m2": 5553.0, "delta_percent": 48.6}
  timestamp: "14:30:25.612"
                    |
                    ▼
Шаг 8: Агент-отчётчик
  action: "synthesize_report"
  input: {"tool_results": {...}}
  output: {"final_answer": "📊 ОТЧЁТ ОБ ИЗМЕНЕНИЯХ..."}
  timestamp: "14:30:25.834"
```

Рис. 13. Процесс цепочки рассуждений

Рисунок 13 демонстрирует функционирование справочного слоя (Reference Layer), который обеспечивает полную цепочку выполнения запроса и является ключевым компонентом для обеспечения объяснимости системы. Каждая сессия получает уникальный идентификатор

(например, 'sess_20250115_143022') и временную метку начала выполнения.

Процесс цепочки включает восемь последовательных шагов. На шаге 1 агент-маршрутизатор выполняет действие 'classify_query', получая на вход запрос пользователя и генерируя выходные данные с типом запроса и оценкой сложности. На шаге 2 селектор инструментов выполняет действие 'select_tools', формируя последовательность из трёх инструментов: YOLO11 OBB, SAM и PixelCounter. Шаги 3-6 фиксируют выполнение инструментов: детекция YOLO на снимке 2009 года, детекция YOLO на снимке 2020 года, сегментация SAM для 2009 года и для 2020 года. На шаге 7 счётчик пикселей вычисляет абсолютное и относительное изменение площади. На шаге 8 агент-отчётчик синтезирует финальный ответ (рисунок 14).

Рис. 14. Пример работы

Таким образом, разработанный алгоритм обнаружения изменений с применением агентного подхода и больших визуально-языковых моделей получил практическую реализацию в виде инструмента Siuuu (Smart Interface for Urban Understanding and Updates) – веб-приложения на основе Gradio, которое обеспечивает автоматическую классификацию запросов и их маршрутизацию в соответствующие модули обработки без необходимости дообучения моделей. Внедрение данной системы демонстрирует переход от теоретической архитектуры к готовому решению для мониторинга городской среды, позволяющему пользователям формулировать аналитические запросы на естественном языке и получать интерпретируемые результаты как в текстовом, так и в визуальном формате.

ЗАКЛЮЧЕНИЕ

В ходе работы достигнута поставленная цель – разработана и протестирована агентная система для обнаружения изменений на разновременных снимках городской среды на базе локальных БВЯМ и специализированных инструментов компьютерного зрения. Предложенный подход, сочетающий агентную архитектуру с модулем планирования и инструментами визуального анализа, позволяет преодолеть ограничения традиционных методов детекции изменений и минимизировать проблему галлюцинаций.

В рамках первой задачи проведено исследование возможностей современных открытых больших визуально-языковых моделей для работы со спутниковыми снимками. Сравнительный анализ пяти моделей выявил, что ни одна из рассмотренных моделей не готова к надёжной работе в изоляции из-за склонности к галлюцинациям при количественной оценке изменений. Установлено, что модели Qwen2.5-VL и InternVL3 демонстрируют наилучшие результаты в понимании запросов, однако требуют интеграции со специализированными инструментами компьютерного зрения для обеспечения точности вычислений.

В рамках второй задачи разработан способ решения задачи обнаружения изменений с применением агентного подхода. Предложена архитектура, состоящая из прикладного уровня, ядра системы (навигатор планирования) и исполняющего уровня. Ключевым элементом является модуль Planning Navigator, включающий три функциональных слоя: уровень понимания задачи пользователя, справочный слой для хранения истории и слой планирования для декомпозиции запросов. Разработаны запросы для каждого типа задач.

В рамках третьей задачи разработан и реализован алгоритм обнаружения изменений. Алгоритм включает шесть этапов: инициализацию данных, классификацию запроса, планирование последовательности инструментов, выполнение инструментов (YOLO11 OBB, SegAnything, PixelCounter), синтез ответа и визуализацию результатов. Практическая реализация осуществлена в виде веб-приложения Siuuu на базе Gradio. Кейс-стади на реальных сценариях мониторинга продемонстрировали практическую применимость системы.

Таким образом, решены все поставленные задачи. Предложенный подход позволяет перейти от фиксации изменений к интеллектуальному мониторингу динамики во времени с возможностью локального развёртывания. Перспективы развития связаны с интеграцией открытых больших визуально-языковых моделей в системы городского управления, в частности – в экосистему цифрового двойника «Мета-Москва». Планируется адаптация Siuuu под запросы, валидация на отечественных снимках. Долгосрочная цель – создание легковесного, автономного инструмента для оперативного семантического мониторинга городской среды.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Барталёв С.А., Егоров В.А., Лупян Е.А., Плотников Д.Е., Уваров И.А. Распознавание пахотных земель на основе многолетних спутниковых данных спектрорадиометра MODIS и локально-адаптивной классификации // Компьютерная оптика. 2011. Т. 35. № 1. С. 103–116. EDN: PJKQYB.
2. Лупян Е.А., Барталёв С.А., Ершов Д.В. Система оперативного мониторинга состояния земной поверхности на основе данных дистанционного зондирования // Современные проблемы дистанционного зондирования Земли из космоса. 2015.
3. Станичная Р.Р., Станичный С.В., Горячкин Ю.Н. Методика анализа изменчивости береговой линии по спутниковым данным // Экологическая безопасность прибрежной и шельфовой зон моря. 2016. № 1. С. 24–30.
4. Владимиров В.М. Дистанционное зондирование Земли : учебное пособие / В.М. Владимиров, Д.Д. Дмитриев, В.Н. Тяпкин, Ю.Л. Фатеев. - Красноярск : Сибирский федеральный университет, 2014. - 196 с. - ISBN 978-5-7638-3084-2.
5. Тёрк М., Дэвис Р. Компьютерное зрение. Передовые методы и глубокое обучение / М. Тёрк, Р. Дэвис ; пер. с англ. В. Яценков. – Москва : ДМК Пресс, 2022. – 690 с – ISBN 978-5-93700-148-1.
6. Shi W., Zhang M., Zhang R. Обнаружение изменений на основе искусственного интеллекта: современное состояние и вызовы // Remote Sensing. 2020. Vol. 12. № 10. 1688. https://doi.org/10.3390/rs12101688
7. Сделай SAM: Segment Anything Model в задачах компьютерного зрения (часть 1) // Хабр. – 2023. – 29 августа. – URL: https://habr.com/ru/companies/sberdevices/articles/757606.
8. Jocher G., Chaurasia A. YOLOv5 и YOLO11: модели детекции объектов. Zenodo, 2022. https://doi.org/10.5281/zenodo.7002879.
9. Qwen2.5-VL-3B. – URL: https://huggingface.co/Qwen
10. LLaVA-OneVision-7B. – URL: https://huggingface.co/lmms-lab
11. InternVL3-8B. – URL: https://huggingface.co/OpenGVLab/InternVL3-8B
12. Florence. – URL: https://huggingface.co/microsoft/Florence-2-large
13. Zhan Y., Xiong Z., Yuan Y. SkyEyeGPT: Unifying remote sensing vision-language tasks via instruction tuning with large language model // ISPRS Journal of Photogrammetry and Remote Sensing. – 2025. – Vol. 222. – P. 45-62.
14. Zhang W., Cai M., Ning Y., et al. EarthGPT-X: A spatial MLLM for multi-level multi-source remote sensing imagery understanding with visual prompting // IEEE Transactions on Geoscience and Remote Sensing. – 2025. – Vol. 63. – Article 4701016.

15. Liu C., Chen K., Zhang H., et al. Change-Agent: Towards interactive comprehensive remote sensing change interpretation and analysis // IEEE Transactions on Geoscience and Remote Sensing. – 2024. – Vol. 62. – Article 5530115.

ПРИЛОЖЕНИЕ 1. ПРИМЕРЫ СНИМКОВ

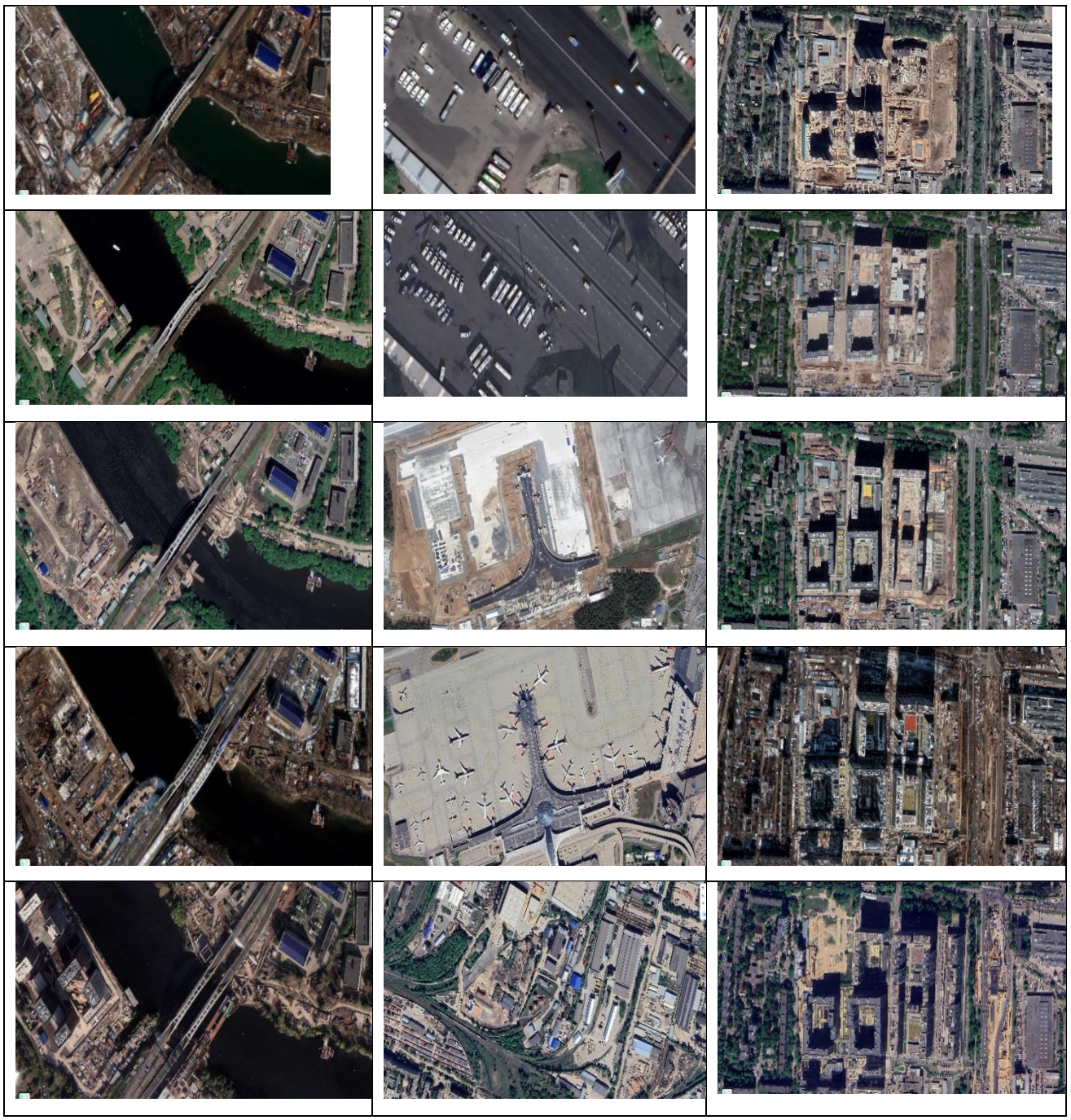


Рис. 15. Примеры снимков

ПРИЛОЖЕНИЕ 2. СРАВНИТЕЛЬНЫЙ АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Таблица 4. Сравнительный анализ существующих решений

Модель / Система	Архитектура	Поддержка нескольких снимков	Семантическая интерпретация изменений	Поддержка локального анализа	Требуется дообучение	Примечания / Ключевые ограничения
Qwen2.5-VL-3B	Transformer + Qwen-VL энкодер	**Высокая** (последовательный ввод через промпт)	Высокая при корректном промпте, но есть количественные галлюцинации	**Да** (INT4, 8GB GPU)	Нет	Компактность, пригоден для локального развертывания, но критическая проблема с точностью количественных оценок.
LLaVA-OneVision-7B	LLaMA + CLIP-ViT	**Ограниченная** (поочередная подача)	Средняя, чувствительна к формулировке промпта	**Да** (INT4, 8GB GPU)	Нет	Потенциал для временных рядов через поддержку видео, но не оптимизирован для пар снимков.
InternVL 3-8B	Vision Transformer + InternLM	**Явная** (парные и множественные)	**Высокая** (обучение на геоданных)	**Да** (INT4, 8GB GPU)	Нет	Лучшая интерпретация среди открытых решений, но высокие требования к памяти и сложность кастомизации.
Florence / Florence-2	DaViT энкодер + мультимодальный трансформер	Формально поддерживает, но специфика не учтена	**Низкая** (ориентация на бытовые снимки)	**Да** (до 0.77B параметров)	**Требуется**	Универсальность и компактность, но не учитывает темпоральную логику и специфику мультивременных геоданных.

ControlLLM / Visual ChatGPT	LLM + общие CV-инструменты	Зависит от инструментов	Низкая (не учитывают специфику ДЗЗ)	**Ограниченная** (требуют облачный API)	Нет	Универсальные агенты, низкая производительность на данных ДЗЗ.
RSGPT / SkyEyeGPT	Специализированные геомодели	Не поддерживают (одиночные снимки)	Средняя (только для одного снимка)	Зависит от реализации	Да (под задачу)	Для описания статики, но отсутствуют механизмы рассуждения о временных изменениях.
EarthGPT / GeoChat	Мультимодальные модели ДЗЗ	**Ограниченная**	Средняя (низкая интерпретируемость)	Зависит от реализации	Да (под задачу)	Частичная поддержка пар снимков, но страдают от низкой интерпретируемости результатов.
Change-Agent	Агент для Change Detection	**Поддерживает**	Высокая	Да	**Обязательно** (Supervised)	Единственная система, нацеленная на изменения, но привязана к LEVIR-MCI. **Негибкая**: требует переобучения для новых классов.

ПРИЛОЖЕНИЕ 3. UML-ДИАГРАММА КЛАССОВ

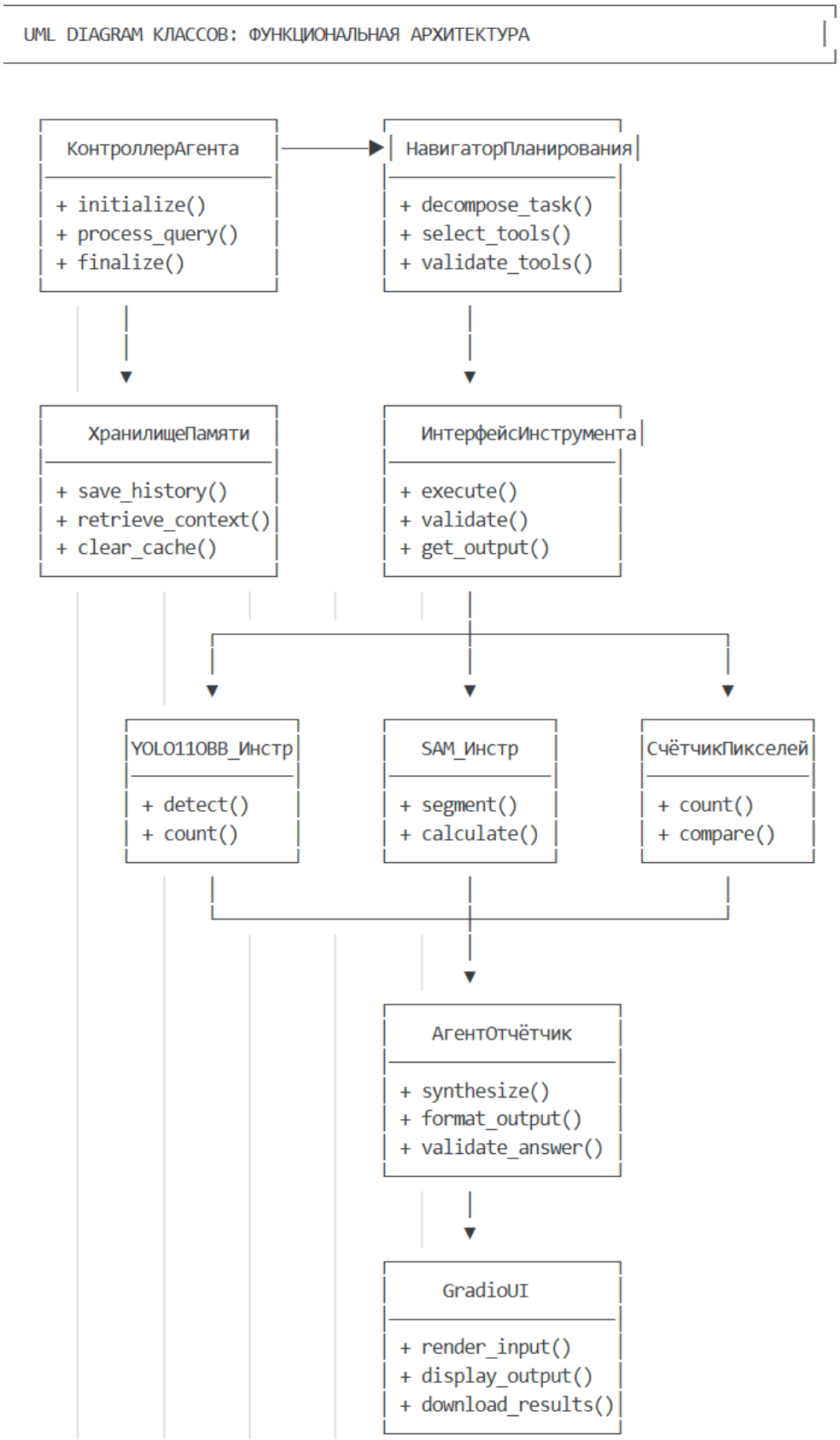


Рис. 16. UML-диаграмма классов

ПРИЛОЖЕНИЕ 4. ИНТЕРФЕЙС SIUUU

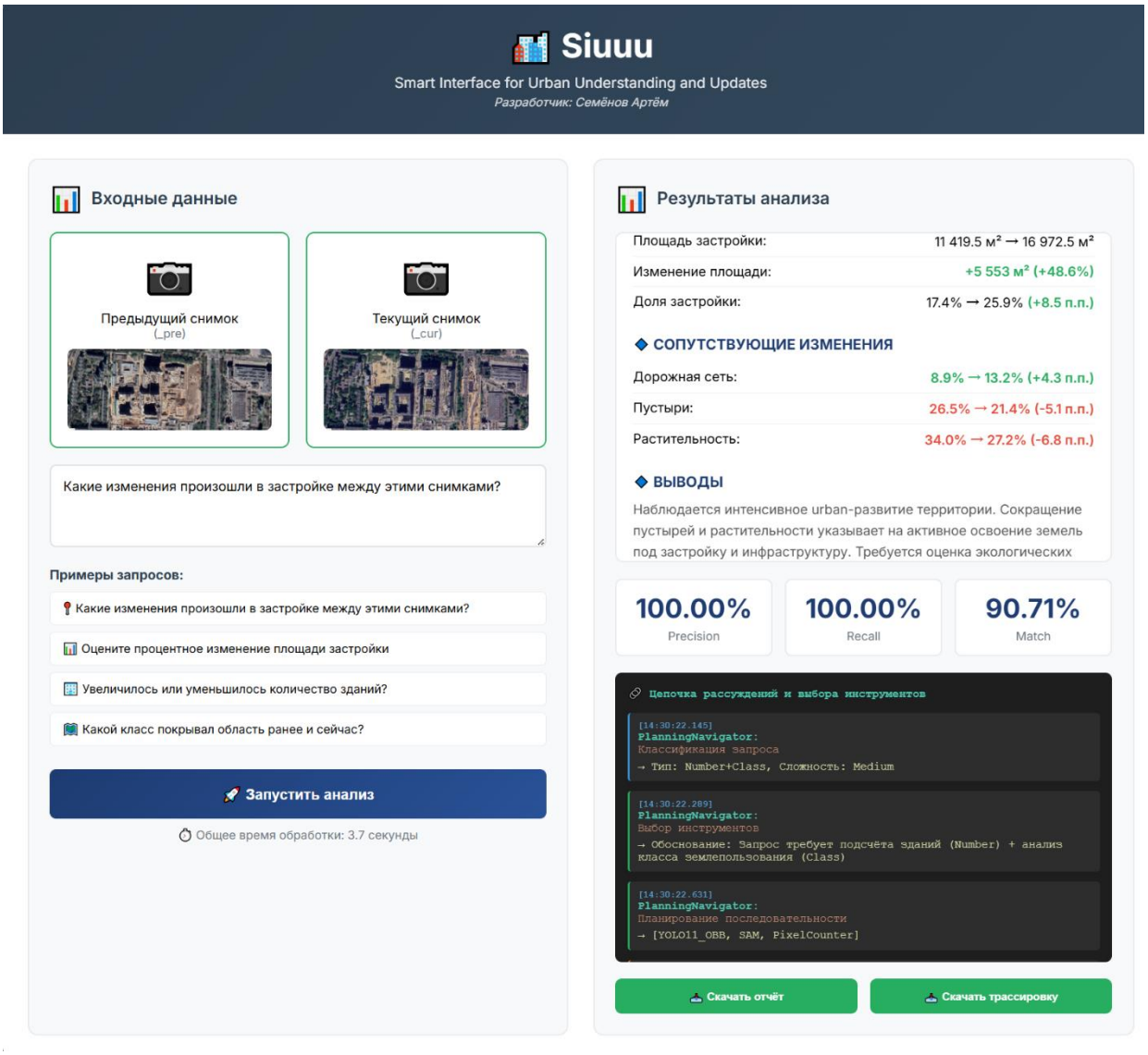


Рис. 17. Интерфейс Siuuu

СОЗДАНИЕ И ИССЛЕДОВАНИЕ ХРОМАТОГРАФИЧЕСКИХ СВОЙСТВ АФФИННОГО СОРБЕНТА НА ОСНОВЕ АПИКАЛЬНОГО ДОМЕНА БЕЛКА ШАПЕРОНА GROEL

Сырнева Дарья Тимофеевна

Москва г, ГБОУ Инженерная школа № 1581, 11 класс

Секция: Биотехнология

Научный руководитель: Шульга Николай Андреевич, преподаватель МГМУ имени И.М. Сеченова

АННОТАЦИЯ

В последние годы интерес к аффинным сорбентам на основе биологических лигандов существенно возрос благодаря их селективности и биосовместимости. Использование целого белка GroEL в качестве лиганда для хроматографических носителей ранее продемонстрировало потенциал в очистке сложных смесей, однако крупные размеры и сложная доменная организация GroEL ограничивают эффективность таких систем. В данной работе представлен подход к созданию аффинного сорбента с использованием рекомбинантного апикального домена GroEL (Apical Domain, AD), иммобилизованного на твердофазной матрице. Результаты работы расширяют понимание структурнофункциональных взаимосвязей белка GroEL и открывают перспективы для разработки сорбентов, адаптированных к задачам биотехнологии.

ВВЕДЕНИЕ

Шапероны представляют собой ключевые элементы клеточной системы контроля качества белков, обеспечивающие правильный фолдинг белков, предотвращение агрегации и участвующие в рефолдинге денатурированных белков. Среди молекулярных шаперонов особое место занимает белок GroEL из Escherichia coli. [1, 2]

Обоснование выбора темы и актуальность: в последние годы интерес к аффинным сорбентам на основе биологических лигандов существенно возрос благодаря их селективности и биосовместимости. В связи с этим актуальной задачей становится разработка сорбентов на основе изолированных функциональных доменов шаперона, способных сохранять ключевые взаимодействия с субстратами при упрощении конструкции.

Гипотеза: апикальный домен GroEL (AD) обладает селективностью к денатурированным/неправильно свернутым белкам благодаря сохранению гидрофобных участков, ответственных за связывание. Это отличает его от коммерческих сорбентов, которые часто нацелены на специфические домены и участки белковой молекулы.

В данной работе представлен подход к созданию аффинного сорбента с использованием рекомбинантного апикального домена GroEL (Apical Domain, AD), иммобилизованного на твердофазной матрице. Результаты работы расширяют понимание структурно-функциональных взаимосвязей белка GroEL и открывают перспективы для разработки сорбентов, адаптированных к задачам биотехнологии.

Цель: разработка сорбентов на основе изолированных функциональных доменов шаперона, способных сохранять аффинность к белкам неправильной конформации.

Задачами, которые были поставлены в ходе исследования включали:

1. Разработать метод иммобилизации апикального домена GroEL (AD) на хроматографической матрице с сохранением функциональной активности.
2. Охарактеризовать физико-химические свойства сорбента (емкость, стабильность, селективность) в модельных системах с использованием нативных и денатурированных белков.
3. Изучить профили связывания белков в клеточных лизатах до и после теплового стресса методом SDS-PAGE.
4. Оценить применимость сорбента для диагностики белкового стресса в биотехнологических культурах на примере реальных образцов.
5. Оптимизировать условия хроматографии для повышения эффективности очистки и детекции белков.

ЛИТЕРАТУРНЫЙ ОБЗОР

1. Апикальный домен шаперона GroEL

Белок GroEL - ключевой представитель шаперонинов у Escherichia coli. Он играет центральную роль в поддержании протеостаза, обеспечивая фолдинг широкого спектра полипептидов.

Фолдинг (от англ. folding) — это процесс, при котором полипептидная цепь сворачивается в свою природную трёхмерную структуру, определяющую её функциональные свойства. Этот процесс включает в себя образование специфических внутримолекулярных связей, таких как водородные, ионные и гидрофобные взаимодействия, которые обеспечивают стабилизацию определённой конформации белка. Правильный фолдинг необходим для того, чтобы белок мог выполнять свои биологические функции, так как его активность зависит от правильности пространственной структуры. [4]

Олигомерная структура белка GroEL, состоящего из двух семичленных колец, формирует полость, где происходит АТФ-зависимый фолдинг белка. GroEL также обеспечивает фолдинг не только вновь синтезируемых белков, но и тех, которые подверглись денатурации вследствие изменения температуры, pH или других стрессовых факторов.

Элементом, который определяет специфичность GroEL, является апикальный домен (AD), расположенный в дистальной части каждой субъединицы. AD шаперонина GroEL играет ключевую роль в связывании субстратов благодаря своим гидрофобным участкам, которые взаимодействуют с обнаженными областями денатурированных белков. Исследования показывают, что именно апикальный домен является важным для специфичности связывания белков и их последующего фолдинга. [3]

Таким образом, GroEL является белковым компонентом, отвечающим за поддержание белкового гомеостаза в клетке, и его уникальная способность к корректному сворачиванию белков делает его ценным инструментом не только для клеточной биологии, но и для разработки методов очистки белков, таких как аффинная хроматография.

2. Аффинная хроматография

Аффинная хроматография — это один из самых эффективных методов очистки белков, который обладает высоким разрешением и способностью очищать компоненты в тысячи раз. Этот метод основан на взаимодействии молекулы-мишени и специфического лиганда, который связан с матрицей, в то время как мишень

находится в подвижной фазе как часть сложной смеси. После этапа улавливания обычно следует промывка и элюирование, в результате чего получается высокоочищенный белок.

Биологическое взаимодействие между лигандом и молекулой-мишенью может быть результатом образования водородных связей, электростатического, Ван-дер-Ваальсового или гидрофобного взаимодействия. [7]

Аффинная матрица, с которой связан лиганд, должна быть макропористой, обладать высокой химической и физической стойкостью. Это касается как материала носителя, так и присоединенного лиганда, которые не должны реагировать на растворители, используемые в процессе, а также не должны разрушаться или повреждаться ферментами и микроорганизмами, которые могут присутствовать в образце. Размер пор должен быть хотя бы в пять раз больше среднего размера биомолекулы, чтобы обеспечить легкий доступ к иммобилизованным лигандам. Поскольку чаще всего подвижная фаза представляет собой водный раствор, носитель должен быть гидрофильным.

Выбор лиганда зависит от понимания характера взаимодействий между лигандом и молекулой-мишенью. Лиганд должен специфически связываться с молекулой-мишенью и быть стабильным в различных условиях связывания и элюирования.

Характеристика иммобилизованного лиганда является важным фактором, определяющим полезность аффинной смолы. Это также включает способность воспроизводимого изготовления смолы с аналогичными эксплуатационными характеристиками. Поэтому важно определить оптимальные условия реакции, включая количество лиганда, необходимое для синтеза, чтобы обеспечить стабильность производительности.

Присоединение лигандов к обычным поверхностям может варьироваться в зависимости от целей. Для белков разумной отправной точкой является концентрация лиганда 2,5–10 мг/мл. В зависимости от химического состава активации и реакционной способности лиганда существуют протоколы для широкого диапазона pH и буферных условий. [5, 6]

3. Методы создания аффинных хроматографических сорбентов

Одними из наиболее распространённых аффинных сорбентов являются макропористые агарозные смолы. Во многих исследованиях были предложены метод модификации агарозных смол для создания аффинных сорбентов. [5, 6]

Классическим методом создания аффинных сорбентов является активация агарозы бромцианом. Агароза - полисахарид с гидроксильными группами на своей

поверхности, подвергается химической модификации с целью создания реакционноспособных участков, которые могут быть использованы для связывания лиганда. Процесс активации заключается в обработке агарозы бромцианом, в результате чего на её поверхности образуются цианогруппы. Эти группы могут вступать в реакции с функциональными группами лигандов, таких как аминогруппы, что делает агарозу пригодной для дальнейшего связывания с лигандами. [8, 9]

На этапе активации бромциан (BrCN) замещает водород на гидроксильных группах, образуя цианогруппы, которые обладают высокой реакционной способностью. Эти цианогруппы затем могут сшиваться с аминогруппами или другими функциональными группами, присутствующими на молекуле лиганда. Этот процесс создаёт стабильные ковалентные связи между лигандом и носителем, что позволяет использовать агарозный сорбент для специфической очистки молекул-мишеней. Реакция происходит при нейтральных или слабощелочных pH, что обеспечивает оптимальные условия для формирования стабильных и прочных связей между лигандом и агарозой.

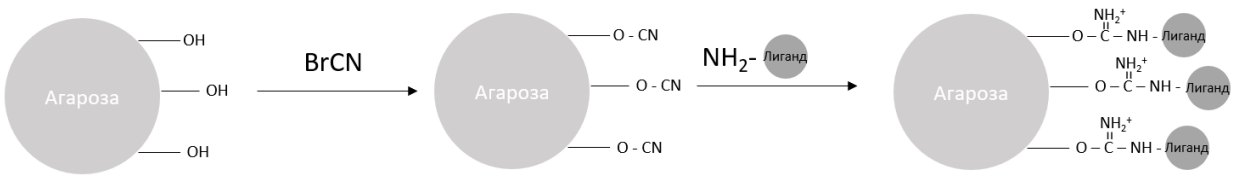


Рис. 1. Принципиальная схема активации материала агарозы и ковалентное связывание лиганда

Бромциан (BrCN) является высоко токсичным веществом, поэтому связи с опасностями, связанными с использованием бромциана, на рынке появились коммерчески доступные агарозные смолы с предварительной активацией, такие как Sepharose®. Эти смолы уже активированы с помощью различных реагентов, что исключает необходимость работы с токсичными веществами и позволяет ускорить подготовку к хроматографическим процессам.

МАТЕРИАЛЫ И МЕТОДЫ

1. Индукция теплового стресса в культуре клеток E. coli

   LB (Luria-Bertani) для бактериальных культур. Использовали LB среду следующего состава: 10 г/л триптона, 5 г/л экстракта дрожжей, 5 г/л NaCl. pH = 7,5. После добавления необходимого объема деионизированной воды и полного растворения все среды были автоклавированы.

   Свежую колонию штамма E.coli инокулировали в 10 мл LB среды. И термостатировали при 37 C° измеряя оптическую плотность раствора в начале и каждый час после, пока OD 600 (оптическая плотность раствора, измеренная на длине волны 600 нм. Этот параметр используется в микробиологии для оценки плотности клеток в растворе или их концентрации в жидкой культуре) не достигнет 0.8.

   Затем культуральную жидкость разделяли на 2 равные части. Одну часть подвергали тепловому стрессу в течении 30 минут при 42 C°. Другую часть оставляли при 37 C°. После инкубации клетки собирали центрифугированием при 10,000 g 10 минут. Промывали клеточный осадок в холодном фосфатном буфере pH 7.4 (137 мМ NaCl; 2,7 мМ 10 мМ Na_2HPO_4; 1,8 мМ NaH_2PO_4). Лизировали клетки с помощью буфера лизиса pH 7.5 (1% (v/v) Triton X-100; 50 мМ Tris-HCl; 150 мМ NaCl). Добавляли протеазный ингибитор (1 мМ PMSF - фенилметилсульфонил фторид) и хранили на льду до хроматографического разделения не более 1 часа.

2. Создание аффинного хроматографического сорбента на основе AD GroEL.

   Раствор для связывания лиганда готовили следующим образом: 8.4 г натрия гидрокарбоната и 29.2 г натрия хлорида поместили в мерную колбу вместимостью 1000 мл, растворили в дистиллированной воде и довели объем раствора до метки тем же растворителем. С помощью pH-метра проверили pH раствора и, при необходимости, корректировали его до значения 8.3, добавляя 1 М раствор гидроксида натрия (NaOH) для повышения pH или 1 М раствор соляной кислоты (HCl) для его понижения.

   Раствор для блокировки активированных групп готовили следующим образом: растворяли 1 г этаноламина в 100 мл дистиллированной воды, чтобы получить 1 М раствор, и корректировали pH до 8.0 с помощью 1 М HCl.

   Раствор для промывки 1 (Трис - гидрохлоридный буферный раствор pH 8,0) готовили следующим образом: 1.21 г трис(гидроксиметил)аминометана (Tris) и 2.92 г натрия хлорида растворили в 100 мл дистиллированной воды. Доводят pH до 8,0 потенциометрически с помощью 1 М раствора хлористоводородной кислоты.

Раствор для промывки 2 (ацетатный буферный раствор pH 4,0) готовили следующим образом: 0.58 г уксусной кислоты, 2.92 г хлорида натрия и 0.82 г ацетата натрия растворили в 100 мл дистиллированной воды. С помощью pH-метра проверили pH раствора и, при необходимости, корректировали его до 4.0, добавляя 1 М раствор уксусной кислоты для понижения pH.

Сшивка лиганда и хроматографического сорбента проходит в несколько стадий.

1) Связывание лиганда. 10 мл раствора для связывания лиганда, содержащего 50 мг апикального домена GroEL смешивали в конической колбе с 5 г предварительно активированной смолы CNBr-activated Sepharose™ 4B (Cytiva, США). Оставляли при температуре 4 С° на 12 часов при постоянном перемешивании на орбитальном шейкере.
2) Промывка. Полученный раствор фильтровали с использованием вакуумной фильтровальной установки на стеклянном фильтре и промывали 10 мл раствора для связывания лиганда 5 раз, не допуская высыхания смолы.
3) Блокирование активных групп. Помещали сорбент в 50 мл раствора для блокировки активированных групп и оставляли на 2 часа на орбитальном шейкере.
4) Промывка. Полученный раствор подвергали промывке на стеклянном фильтре с использованием 20 мл раствора для промывки 1 и 20 мл раствора для промывки 2, повторяя цикл промывки трижды.

3. Подготовка хроматографической системы

В аффинной хроматографии белков используются следующие типы подвижных фаз - буфер для нанесения и буфер для элюирования. Буфер для нанесения представляет благоприятные условия для связывания белков с лигандом на поверхности сорбента и, как правило, служит растворителем образца и хроматографического сорбента. Буфер для элюирования служит для снижения степени ассоциации белков с лигандом и снятия их с хроматографического сорбента. Элюирование может быть специфическим и неспецифическим, в первом случае вносят конкурентную молекулу, которая обладает большим сродством к лиганду, тем самым вытесняя целевой белок из комплекса с лигандом, во втором случае используют изменение физико-химических свойств раствора, таких как значение pH и ионная сила. Достаточно часто применяют метод градиентной элюции, при котором состав подвижной фазы меняется с течением времени и происходит постепенная смена буфера нанесения на буфер элюции. Такой подход позволяет добиться лучшего разделения сложных белковых смесей.

В нашем случае подбор специфического сорбента для анализа сложной смеси затруднен, так как лиганд взаимодействует с множеством белков неправильной третичной структуры, поэтому был выбран метод неспецифической элюции градиентом ионной силы или метод элюции градиентом соли. При повышении концентрации соли в подвижной фазе повышается ионная сила раствора и ослабляются силы взаимодействия белков с лигандом хроматографической матрицы.

Буфер для нанесения: помещали 0.6 г трис(гидроксиметил)аминометана и 8.77 г натрия хлорида в мерную колбу емкостью 1 литр, и доводили объем раствора водой до метки. При необходимости корректировали pH до значения 8.0 с помощью 1 М раствора HCl.

Буфер для элюирования: 0.6 г трис(гидроксиметил)аминометана и 58.5 г натрия хлорида в мерную колбу емкостью 1 литр, и доводили объем раствора водой до метки. При необходимости корректировали pH до значения 8.0 с помощью 1 М раствора HCl.

5,0 г хроматографического сорбента, приготовленного к описано выше, помещали в коническую колбу и добавлял 1,25 мл буфера для нанесения. Взвесь помещали в набивную хроматографическую колонку HR 16/10 (Cytiva, США), размещали горизонтально и приливали буфер для нанесения по стенкам колонки, не допуская поднятия зерен сорбента. Закрывали хроматографическую колонку и подключали к хроматографу BioRad NGC Scout 100 (BioRad, США). Промывали сорбент 50 мл буфера для нанесения.

Условия хроматографирования

1) Скорость потока: 2 мл/мин
2) Длина волны детекции: 280 нм
3) Нанесение образца: 5 мл
4) Промывка: 10 мин
5) Градиентная элюция: повышение концентрации NaCl от 0.15 М до 1.0 М в течение 30 минут.
6) Сбор фракции белка: фракции собирали в пробирки типа Эппендорф вместимостью 2 мл.
7) Полученные образцы лиофилизировали и хранили при - 20 С° до использования.

4. Электрофорез в полиакриламидном геле

Электрофорез — это процесс перемещения заряженных частиц в растворе под

действием электрического поля. Для разделения белков в денатурирующих условиях использовался полиакриламидный гель (ПААГ) с натрия додецилсульфатом (SDS). В процессе денатурации белки разворачиваются до первичной структуры, и гидрофобные участки становятся экспонированными. SDS, благодаря своему гидрофобному углеродному радикалу, взаимодействует с этими участками, стабилизируя развернутую форму белка. Разделение белков в геле происходит под воздействием электрического поля пропорционально размеру белка.

Для проведения электрофореза необходимы следующие растворы:

30% раствор акриламида: для приготовления этого раствора необходимо растворить 29 г акриламида и 1 г метилен-бис-акриламида в 70 мл воды.

Разделяющий гель. Состав раствора для разделяющего геля: 2,42 мл 12,5%-ного раствора акриламида, 1,5 мл 375 мМ Tris-HCl с pH 8.8 и 45 мкл 0,1%-ного раствора SDS.

Концентрирующий гель. Состав раствора для концентрирующего геля: 0,85 мл 5%-ного акриламида, 0,13 мл 130 мМ Tris-HCl с pH 6.8, 10 мкл 0,1%-ного раствора SDS.

Буфер для электрофореза (pH 8.3): для приготовления буфера нужно растворить 15,1 г Tris-base (25 мМ), 94 г глицина (250 мМ) и добавить 50 мл 0,1%-ного раствора SDS.

Буфер для образцов (pH 8.0): состав буфера для образцов: 0,02 М Tris-HCl, 0,0021 М ЭДТА, 2% SDS, 10% 2-меркаптоэтанола, 20% глицерина, 0,002% бромфенолового синего.

Раствор для фиксации белков в ПААГ: для фиксации белков в ПААГ смешивали 125 мл этанола, 50 мл уксусной кислоты и 325 мл воды на общий объём 500 мл.

Раствор для окрашивания белков: для окрашивания белков в ПААГ смешивали 125 мл этанола, 50 мл уксусной кислоты, 325 мл воды и 1,25 г Кумасси R-250 на общий объём 500 мл.

Раствор для отмывки: для отмывки после окрашивания используется раствор, состоящий из 50 мл уксусной кислоты, 50 мл этанола и 400 мл воды на объём 500 мл.

Приготовление гелей для электрофореза. Гели для электрофореза изготавливаются путем полимеризации акриламида между стеклами специальной установки. Гель состоит из двух слоев – разделяющего, с высокой плотностью полимеров акриламида и необходимый для разделения белков, и концентрирующего, с более низкой плотностью и необходимый для концентрирования белков на границе двух слоев. Процесс полимеризации инициируется добавлением к растворам для приготовления гелей персульфата аммония (PSA) и тетраметилэтилендиамина

(TEMED). Сначала между стеклами заливают раствор для приготовления разделяющего геля, а затем сверху наслаивают раствор для приготовления концентрирующего геля. Лунки для образцов формируют специальным инструментом – «гребенкой», которую помещают в слой концентрирующего геля. Стекла с гелем хранят при температуре 4 С° до использования.

После заливки геля готовили образцы. Для этого размораживали образцы, добавляли к ним 100 мкл буфера для образцов, затем нагревали при +95°С в течение 5 минут.

В камере для электрофореза фиксировали стекла с гелем и камеру заполняли буфером для электрофореза и наносили образцы в лунку в количестве 20 мкл. Также в первую лунку каждого геля наносили маркер молекулярного веса Pierce™ Unstained Protein MW Marker. Электрофорез проводили под напряжением 100 В до момента пока краситель не опустится к нижнему фронту геля. После этого гель вынимали и документировали с помощью сканера.

РЕЗУЛЬТАТЫ

В результате проведенного исследования были получены хроматограммы элюирования лизата клеточной культуры E.coli в денатурирующих и не денатурирующих условиях, собраны фракции градиентного элюирования с аффинного хроматографического сорбента на основе апикального домена шаперона GroEL. Проведено электрофоретическое разделение фракций элюирования для определения их белкового профиля.

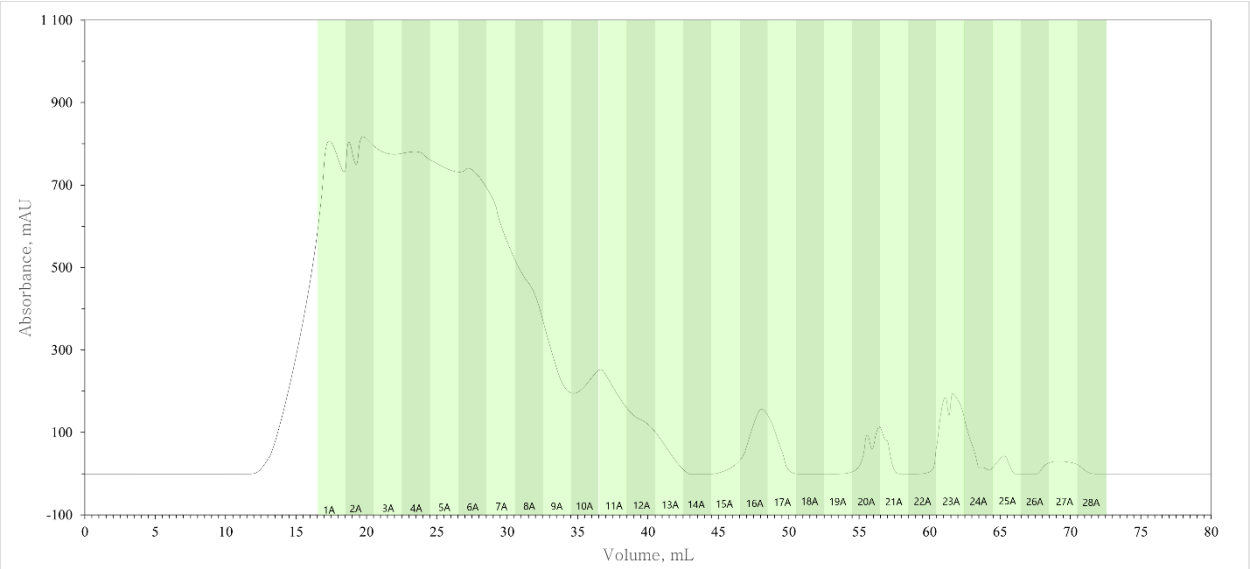


Рис. 2. Хроматограмма испытуемого образца лизата E.coli в не денатурирующих условиях. Цифрами внутри зеленых окон сбора фракций указан их номер для последующего анализа.

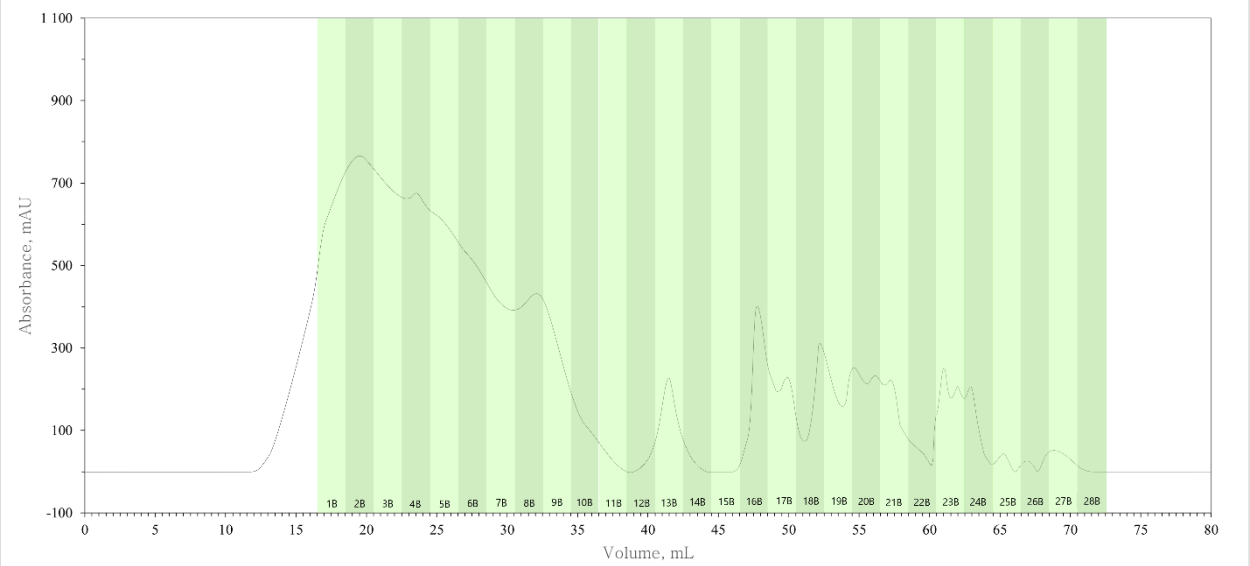


Рис. 3. Хроматограмма испытуемого образца лизата E.coli в денатурирующих условиях. Цифрами внутри зеленых окон сбора фракций указан их номер для последующего анализа.

Типичным представлением хроматограмм в аффинной хроматографии является график зависимости оптической плтности элюата от объема подвижной фазы или колоночных объемов.

Хроматографические профили лизатов E. coli, полученные на разработанном сорбенте отличаются по количеству и составу пиков. Анализ показал увеличение количества белков обладающих аффинностью к сорбенту в культуре клеток, подвергнутых тепловому стрессу, что подтверждает способность сорбента к диагностике клеточного стресса. Электрофоретический анализ показал отличия в белковом составе элюируемых фракций. Электрофореграммы представленны на рисунке 4.

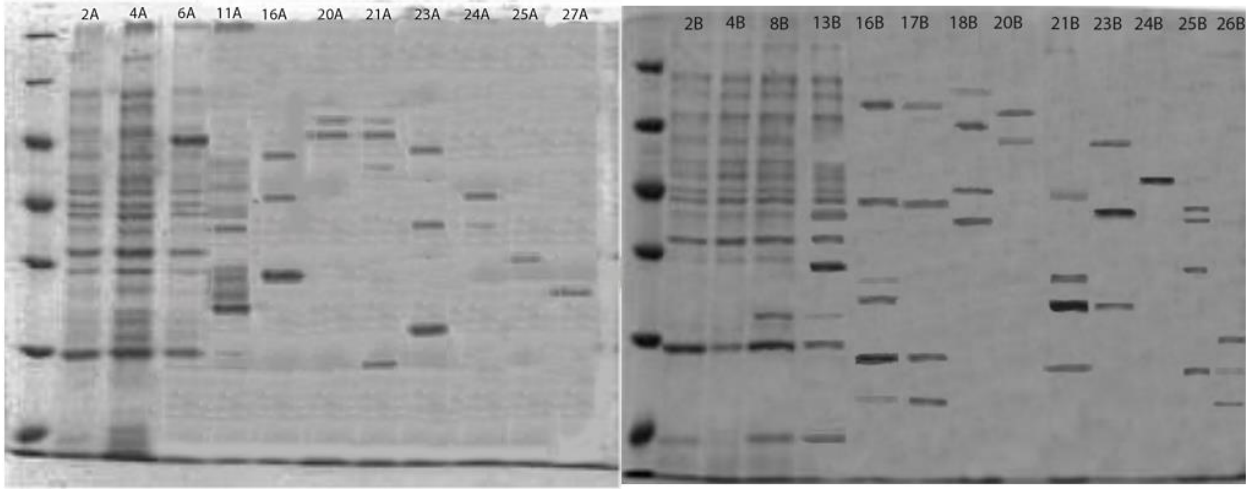


Рис. 4. Электрофореграммы фракций лизата E.coli. Справа – фракции лизата контрольной группы, слева – фракции лизата культуры клеток подвергнутых тепловому стрессу. Цифрами над дорожками геля указаны номера анализируемых фракций.

ЗАКЛЮЧЕНИЕ

1. В результате полученных данных можно сделать вывод о стабильности и эффективности полученного аффинного сорбента
2. Аффинный хроматографический сорбент на основе апикального домена шаперона GroEL
   2.1. Показал аффинность к некоторым белкам нативного лизата E.coli, а также к денатурированным белкам в результате воздействия теплового стресса.
   2.2. Показал способность к надежному разделению сложной белковой смеси.
   2.3. Может быть применен в качестве инструмента для диагностики клеточного стресса в биотехнологических процессах
3. Полученные электрофореграммы демонстрируют способность сорбента к разделению денатурированных или частично денатурированных белков.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Saibil H. R. Chaperone machines for protein folding, unfolding and disaggregation // Nature Reviews Molecular Cell Biology. 2013.
2. Fenton W. A., et al. Structure and function of the GroEL chaperonin // Protein Science. 1994.
3. Xu Z., et al. The crystal structure of the asymmetric GroEL–GroES–(ADP)7 chaperonin complex // Nature. 1997.
4. Clare D. K., et al. ATP-triggered conformational changes delineate substrate-binding and -folding mechanics of the GroEL chaperonin // Cell. 2012.
5. David S. Hage. High-performance affinity chromatography: a powerful tool for studying serum protein binding // Journal of Chromatography B. 2002. T. 768, № 1. C. 3-30
6. Coelho, L. C. B. Protein Purification by Affinity Chromatography / L. C. B. Coelho. – In: Protein Purification. – January 2012. – Federal University of Pernambuco.
7. Ибрагимов А.Н., Бикмуллин А.Г., Сатаева Д.А., Лопухов Л.В., Зайнуллин Л.И., Алимова Ф.К. Хроматографические методы очистки белков. Казань, 2013.
8. Wilchek, M., & Miron, T. A simplified method for cyanogen bromide activation of agarose for affinity chromatography // Analytical Biochemistry. 1982 126(1), 223-229.
9. March, S. C., Parikh, I., & Cuatrecasas, P. A simplified method for cyanogen bromide activation of agarose for affinity chromatography. // Analytical Biochemistry. 1974. 60(1), 149-152.

РАЗРАБОТКА МОБИЛЬНОГО УСТРОЙСТВА ДИСТАНЦИОННОЙ ТАКТИЛЬНОЙ ИНДИКАЦИИ ПРОСТРАНСТВА ДЛЯ НЕЗРЯЧИХ (УДТИП)

Чеглаков Владимир Павлович

Москва г, ГБОУ Школа № 444, 11 класс

Секция: Проекты для лиц с ограниченными возможностями здоровья

Научный руководитель: Граськина Елена Евгеньевна, учитель информатики ГБОУ «Московская международная школа»

АННОТАЦИЯ

В современном обществе обеспечение равных возможностей и повышение качества жизни для людей с ограниченными возможностями здоровья является важнейшей социальной задачей, отвечающей принципам гуманизма и инклюзивного развития. Особую значимость приобретает проблема мобильности лиц с глубокими нарушениями зрения (инвалидов по зрению).

Актуальность и проблема

Существующие средства ориентации для незрячих (тактильная трость, собаки-поводыри) имеют ограниченную дальность обнаружения препятствий, а современные электронные устройства зачастую сложны в интерпретации сигналов.

Цель работы состоит в создании доступного мобильного ассистента-помощника, обеспечивающего интуитивное восприятие пространства за счет дистанционного тактильного контакта.

Принцип работы и техническое решение

Разработанное Устройство Дистанционной Тактильной Индикации Пространства для незрячих (УДТИП) основано на комбинации двух ключевых технологий: многозонного тонального (дальномерного) сканирования и многоканального виброотклика. Это позволяет транслировать информацию о расстоянии до объектов не в виде абстрактных звуков и цифр, а в виде тактильного образа, где удаленность обьекта кодируется частотой вибрации конкретного вибромотора.

Новизна работы

На уровне любительской разработки экспериментально обоснован тезис о том, что синергия многозонного сканирования и многоканальной вибрации создает эффект «дистанционной кожи». Обратная связь становится интуитивно понятной, так как имитирует

естественное физическое взаимодействие: чем ближе препятствие, тем сильнее (или выше частота) «касание» соответствующей зоны тела.

Экспериментальная база и результаты

Апробация прототипа УДТИП – мобильного ассистента-помощника для людей с потерей зрения проведена на базе центра «Московское долголетие» с участием людей с нарушением зрения. Обратная связь от респондентов подтвердила высокую адаптацию к прибору и низкий порог вхождения благодаря интуитивной тактильной индикации.

Перспективы развития

Полученные в ходе экспериментов с УДТИП данные позволяют сделать вывод, что предложенная архитектура устройства является основой для дальнейших направлений развития, таких как миниатюризация (при переходе на более компактную элементную базу): повышение точности классификации за счет внедрения дополнительных алгоритмов распознавания типов препятствий; Улучшение эргономики (адаптация формы и расположение элементов под анатомические особенности пользователя УДТИП.

Заключение

Разработанный мобильный ассистент-помощник доказывает эффективность применения комбинированного тактильно-дальномерного подхода для пространственной ориентации незрячих, предлагая простое, недорогое и функциональное решение для повышения качества жизни людей с инвалидностью по зрению.

ВВЕДЕНИЕ

АКТУАЛЬНОСТЬ ТЕМЫ УЧЕБНО-ИССЛЕДОВАТЕЛЬСКОГО ПРОЕКТА

В современном обществе обеспечение равных возможностей и повышение качества жизни для людей с ограниченными возможностями здоровья является важнейшей социальной задачей, отвечающей принципам гуманизма и инклюзивного развития. Особую значимость приобретает проблема мобильности лиц с глубокими нарушениями зрения (инвалидов по зрению). Самостоятельное и безопасное передвижение в пространстве – ключевой фактор их социальной интеграции, независимости и доступа к образованию, труду, культурной жизни [1].

Несмотря на существование традиционных средств ориентации, таких как белая трость, их функциональные возможности существенно ограничены. Белая трость позволяет выявлять препятствия лишь при непосредственном физическом контакте, не предоставляя заблаговременной информации о препятствиях на расстоянии, особенно расположенных выше пояса или ниже уровня земли (ступени вниз, люки, бордюры), а также о динамических объектах. Существующие электронные аналоги и специализированные устройства (умные трости, сенсорные браслеты) зачастую обладают рядом существенных недостатков, таких как высокая стоимость, ограниченная дальность и точность сканирования, громоздкость, недостаточное время автономной работы, а главное – не интуитивная или информативно бедная обратная связь (чаще всего звуковая, которая может мешать слуховой ориентации, или простая ненаправленная вибрация).

Таким образом, наблюдается острая потребность в создании доступного, надежного и эргономичного технического решения, способного предоставлять незрячему пользователю детальную и своевременную информацию о структуре окружающего пространства в режиме реального времени. Наиболее перспективным представляется подход, основанный на бесконтактном сканировании пространства и преобразовании полученных данных в интуитивно понятную тактильную обратную связь, локализованную в рукоятке устройства. Разработка такой системы непосредственно отвечает на запросы целевой группы, способствуя повышению их безопасности, уверенности и самостоятельности при передвижении, что определяет высокую социальную и практическую значимость учебно-исследовательского проекта (см. рис 1, 2, 3).

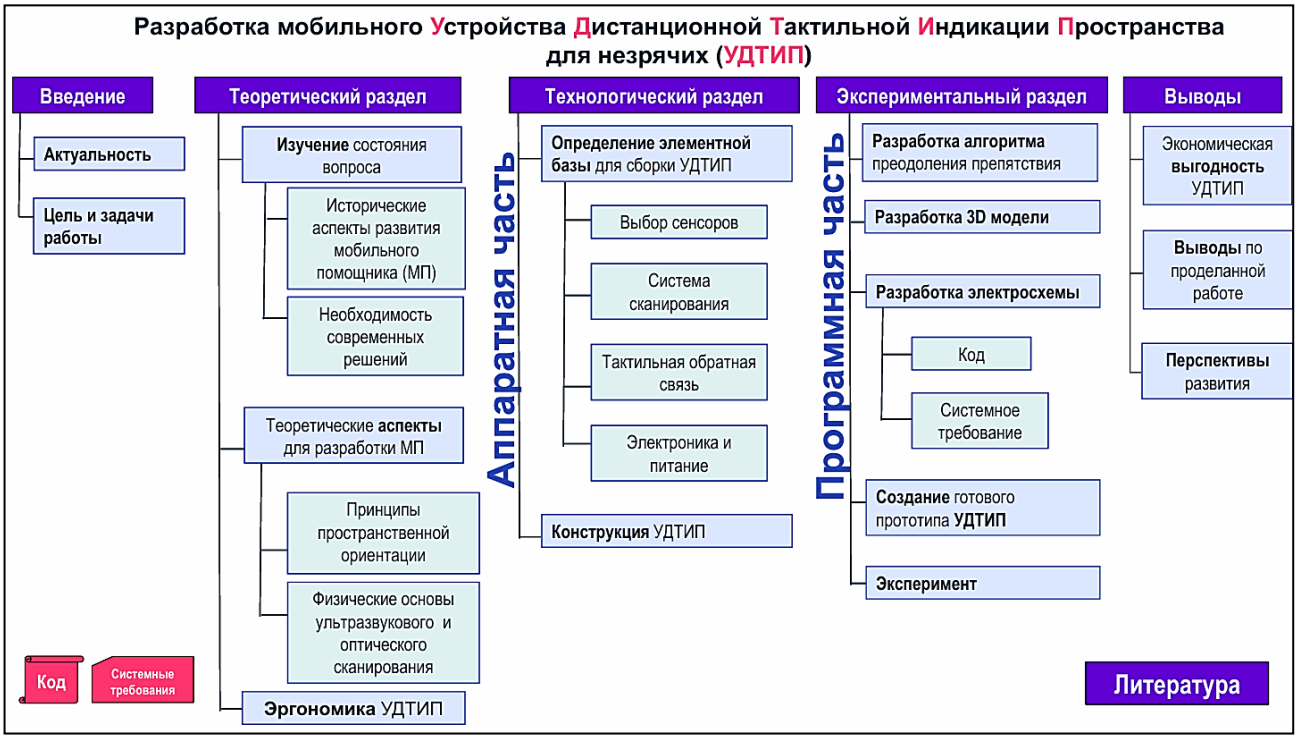


Рис. 1. Каркас работы.

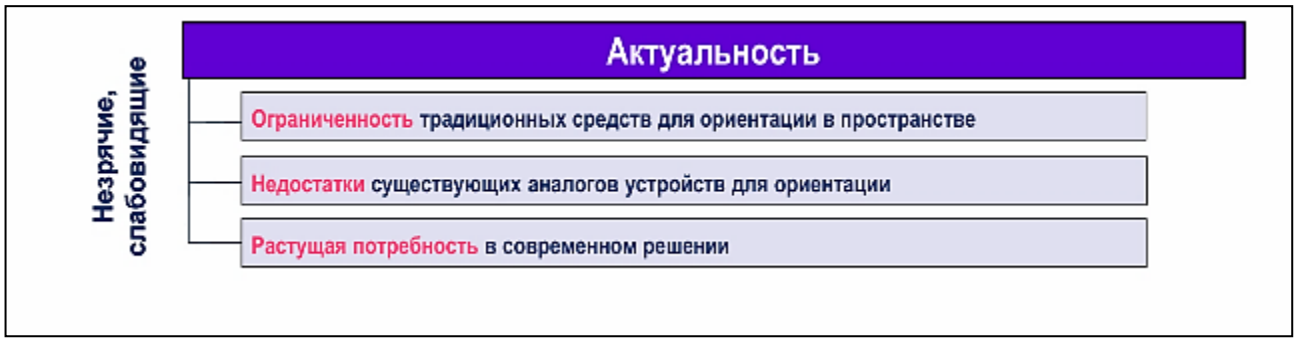


Рис. 2. Актуальность идеи мобильного помощника-ассистента

ЦЕЛЬ РАБОТЫ

Основной целью данного учебно-исследовательского проекта является разработка, изготовление и экспериментальная апробация мобильного ассистента-помощника для незрячих, (людей с особенностями по зрению) – функционального прототипа Устройства Дистанционной Тактильной Индикации Пространства (УДТИП).

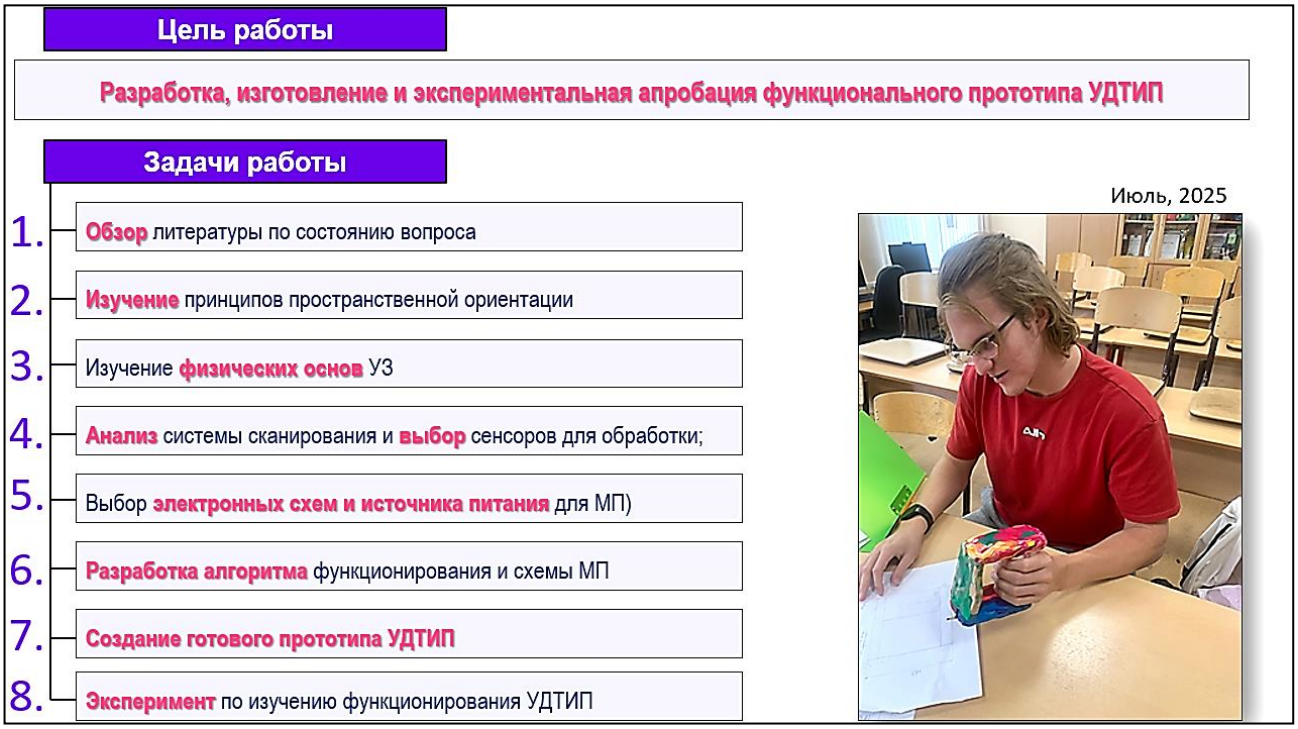


Рис. 3. Цель работы

ЗАДАЧИ РАБОТЫ

Для достижения поставленной цели в рамках проекта необходимо решить следующие ключевые задачи (см. рис. 4):

1. Обзор литературы по состоянию вопроса
2. Изучение принципов пространственной ориентации
3. Изучение физических основ УЗ
4. Анализ системы сканирования и выбор сенсоров для обработки
5. Выбор электронных схем и источника питания для МП[9]
6. Разработка алгоритма функционирования и схемы МП
7. Создание готового прототипа УДТИП
8. Эксперимент по изучению функционирования УДТИП

[9] МП-Мобильный Помощник

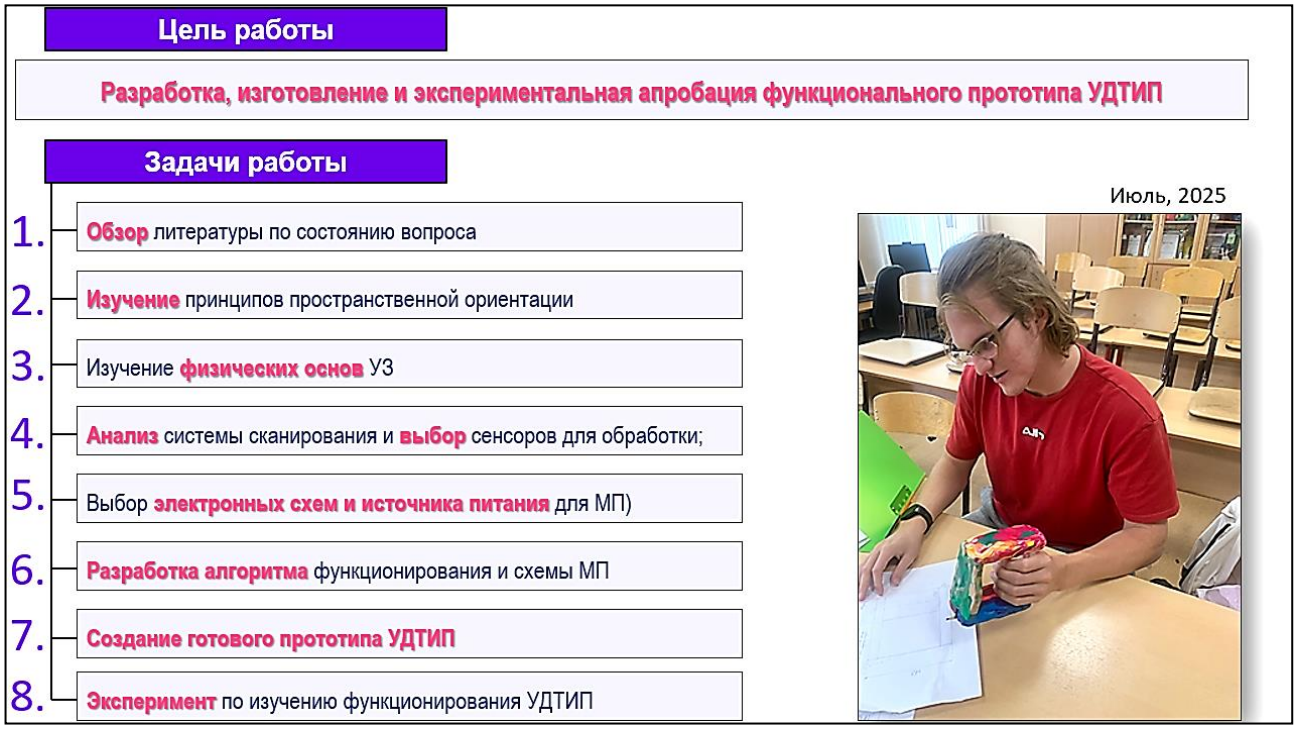


Рис. 4. Задачи работы

ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ И АНАЛИЗ

1.1 ИЗУЧЕНИЕ СОСТОЯНИЯ ВОПРОСА

1.1.1 Исторические аспекты развития мобильных помощи

Исторические аспекты развития мобильных ассистентов-помощников, в частности, показывают, что средства ориентации претерпели длительную эволюцию. Начиная от простейших посохов, описанных уже в древних источниках, человечество пришло к стандартной белой трости, первое широкомасштабное использование которой датируется XX веком, когда активно действовали такие общества, как Королевский национальный институт слепых в Великобритании и аналогичные по всему миру [1]. Стандартом уже является белая трость, которая стала не только инструментом, но и международным символом слепоты. Тем не менее, принципиальное ограничение – необходимость фактического контакта с препятствием — побудила человечество к поиску технологических решений. Уже во второй половине XX века появлялись первые электронные устройства, использующие ультразвук или инфракрасное излучение для дистанционного распознавания препятствий, обычно с звуковой индикации.

1.1.2 Необходимость современных мобильных ассистентов- помощников

Численность людей с глубокими нарушениями зрения неуклонно растет. Этому способствуют старение населения (катаракта, макулодистрофия), рост диабета (диабетическая ретинопатия), а также травмы зрения вследствие катастроф и военных конфликтов, включая СВО. Для этой расширяющейся группы самостоятельное передвижение остается сложной задачей. Традиционная белая трость, выявляя препятствия только при контакте, не дает

заблаговременного предупреждения о разновысотных (низкие ветви, ступени вниз) или динамических препятствиях. Существующие электронные аналоги часто страдают от высокой стоимости, громоздкости и, что критично, использования звуковой обратной связи, маскирующей важные для ориентации окружающие звуки. Сочетание роста потребности, ограниченности традиционных средств и недостатков электронных решений создает острую потребность в доступном, компактном устройстве с тактильной обратной связью, не мешающей слуху.

1.2 ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ ДЛЯ РАЗРАБОТКИ МОБИЛЬНЫХ ПОМОЩНИКОВ

1.2.1 Принципы пространственной ориентации

Принципы пространственной ориентации у незрячих людей значительно отличаются от таковых у зрячих и основаны на компенсаторном использовании сохранившихся анализаторов. Важнейшую роль в этом процессе играют (рис. 5)

1. Слух: этот принцип позволяет определять направление и расстояние до источников звука, а также использовать эхолокацию, которая включает оценку расстояния до препятствий по отраженному звуку, например, с помощью щелчков трости или цоканья языка.
2. Осязание (тактильная и кинестетическая чувствительность): прямой контакт с объектами предоставляет информацию о форме, текстуре, размере и расстоянии.
3. Обоняние и терморецепция: эти чувства могут предоставить дополнительную контекстуальную информацию [1].

Эффективное устройство для дистанционного тактильного интерфейса (УДТИП) должно дополнять, а не заменять или мешать этим естественным механизмам ориентации. Тактильная обратная связь выбрана как наиболее подходящий канал, так как она активно использует высокочувствительный тактильный анализатор пользователя, не конкурируя со слухом, который является критически важным для пространственного восприятия и безопасности на улице.

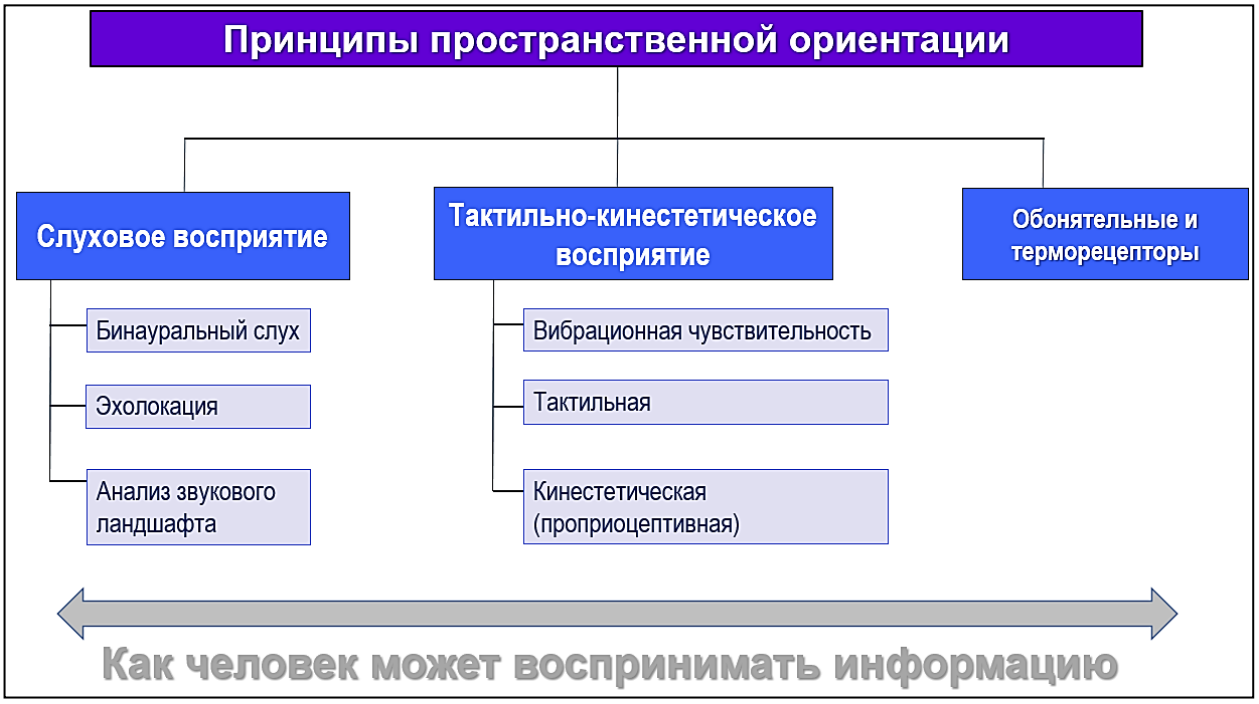


Рис. 5. Принципы пространственной ориентации

1.2.2 Физические основы УЗ/оптического сканирования

Существует два типа сканирования для дистанционного обнаружения препятствий: ультразвуковое (УЗ) и оптическое (лидар). Оба метода основаны на измерении времени, необходимого для прохождения сигнала до объекта и обратно, но они значительно различаются по физическим принципам и практическому применению. Ультразвуковое сканирование использует звуковые волны высокой частоты (40-50 кГц). Когда волна сталкивается с препятствием, она отражается обратно к датчику. Ключевые преимущества ультразвуковой технологии (рис. 6):

1. Возможность обнаружения прозрачных объектов (таких как стеклянные двери, витрины и лужи), которые могут быть невидимы для оптических систем.
2. Высокая устойчивость к атмосферным помехам: туман, дождь и снегопад не оказывают влияния на эффективность работы.
3. Низкое энергопотребление (менее 5 мВт на датчик), что обеспечивает до 8 часов автономной работы.
4. Стоимость системы в 10 раз ниже, чем у оптических аналогов [1-5].

Оптическое сканирование (лидар) применяет лазерные импульсы в инфракрасном или видимом спектре. Основное преимущество этой технологии заключается в ее очень узкой диаграмме направленности (менее 0,5°), что позволяет достигать миллиметровой точности. Скорость света (3·10^8 м/с) остается постоянной и не зависит от внешних условий, что исключает температурные погрешности. Тем не менее, у этой технологии есть серьезные недостатки для навигации незрячих (см. рис. 6):

1. Полная неспособность обнаруживать прозрачные поверхности.
2. Деградация сигнала в условиях дождя, снега и тумана из-за рассеяния света.
3. Слепящий эффект при прямом солнечном свете.
4. Энергопотребление свыше 200 мВт и стоимость от $150 за датчик.

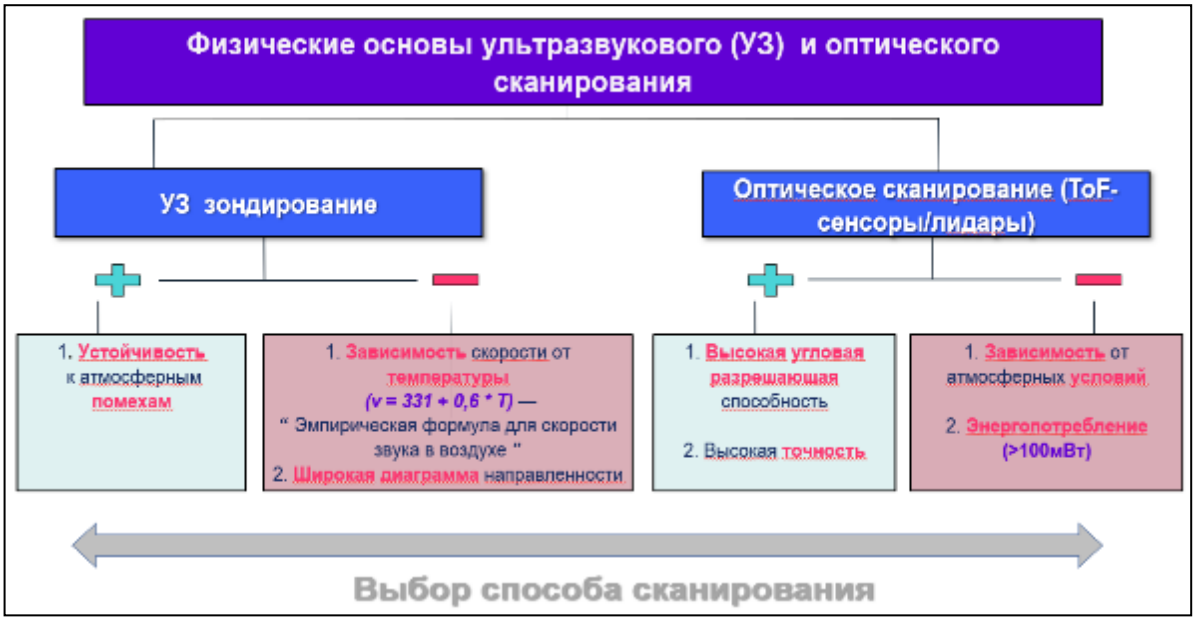


Рис. 6. Физические основы ультразвукового (УЗ) и оптического сканирования

1.3 ЭРГОНОМИКА РАЗРАБАТЫВАЕМОГО МОБИЛЬНОГО АССИСТЕНТА-ПОМОЩНИКА

Современные средства ориентации для незрячих, несмотря на многолетние усилия по их развитию, сталкиваются с серьезными ограничениями. Белая трость, созданная Джеймсом Биггсом в 1921 году, по-прежнему остается символом независимости. Её простота и надежность не вызывают сомнений: легкий алюминиевый стержень, отражающий свет для лучшей видимости, позволяет обнаруживать бордюры и ступени через тактильный контакт. Однако трость оказывается бесполезной при столкновении с препятствиями выше пояса, такими как низкие ветви деревьев, дорожные знаки и строительные конструкции. Динамические объекты, например, велосипедисты или детские коляски, также остаются незамеченными до момента столкновения.

Тактильная плитка, разработанная Сэйити Мияке в 1965 году, произвела революцию в инфраструктуре: рельефные полосы на земле помогают ориентироваться, а конусы предупреждают о переходах. Однако её статичность стала слабым местом — за пределами вокзалов и аэропортов плитка практически не встречается. В парках, на грунтовых тропах или в исторических районах она оказывается бесполезной, оставляя пользователей без необходимой поддержки.

Электронные аналоги пытались решить эти проблемы. Sunu Band (2015 года) в виде браслета использует ультразвук для сканирования окружающего пространства, но его звуковые сигналы могут стать помехой: они заглушают звуки приближающегося транспорта, шаги

прохожих и даже шелест листьев — все то, что незрячие используют для ориентации. WeWALK (2019 года) интегрировал датчики в трость и добавил навигацию через смартфон. Однако за функциональность приходится расплачиваться весом в 850 г — после 20 минут использования рука начинает неметь от усталости, а цена в $1500 делает устройство недоступным для многих [5, 6].

Выходя из минусов и плюсов других решений,

УДТИП должен обладать следующими характеристиками (см. рис. 7):

1. Легкость и эргономичность: вес не превышает 420 г, анатомичная рукоятка для удобства использования.
2. 3D-сканирование угроз: способность комплексно обнаруживать ямы, ступени, препятствия на уровне головы и динамические объекты.
3. Бесшумный тактильный интерфейс
4. Энергоэффективность
5. Социальная и экономическая доступность

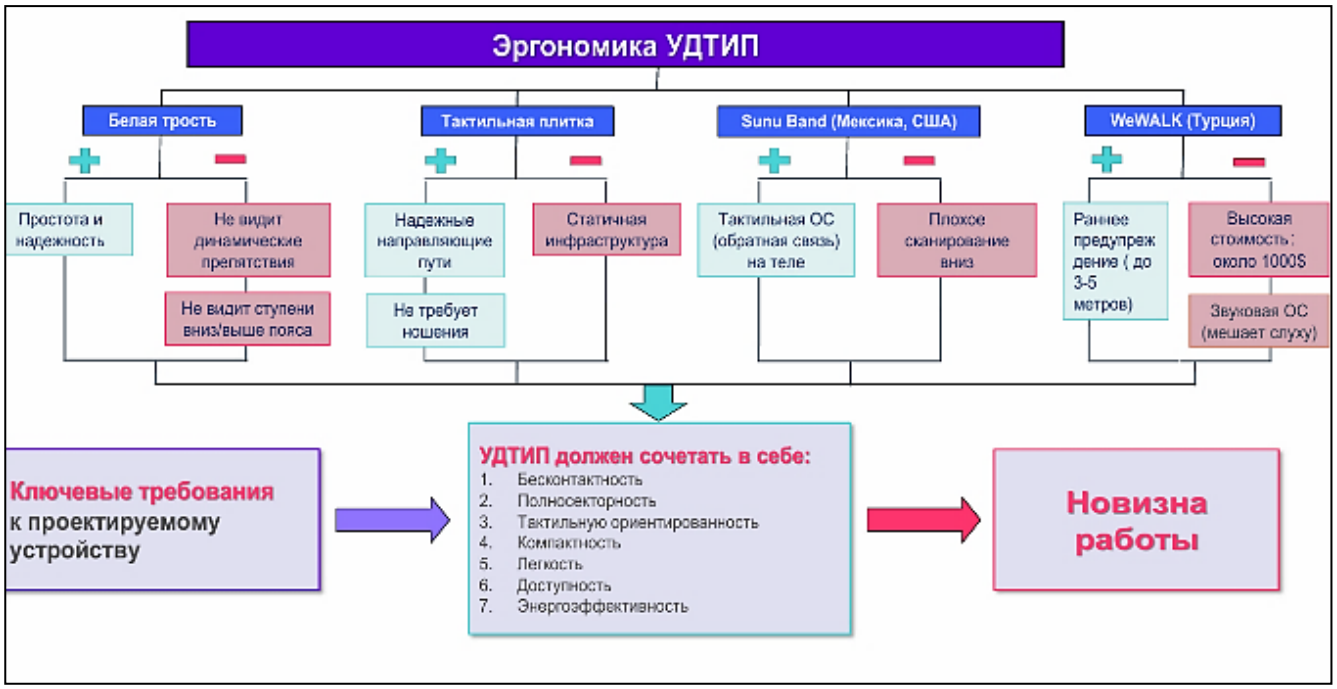


Рис. 7. Эргономика УДТИП

ГЛАВА 2. ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ

2.1 СИСТЕМА СКАНИРОВАНИЯ

Для того чтобы устройство не просто сообщало о наличии препятствия, а давало пользователю понятную информацию о том, где именно находится объект, необходимо получать расстояния сразу в нескольких направлениях. Если измерять расстояние только в одной точке, устройство может лишь “вибрировать”, но не объясняет, препятствие слева или справа, сверху

или снизу. Поэтому в проекте было принято решение использовать пространственное разбиение: по горизонтали — на левый, центральный и правый сектор, а по высоте — на верхний, центральный и нижний уровень.

В результате формируется виртуальная карта из 9 зон (3×3), где каждая зона соответствует определенному участку пространства перед пользователем. Такое разбиение является компромиссом между информативностью и скоростью работы: 9 зон достаточно, чтобы различать препятствия по направлению и высоте, и при этом цикл измерений можно выполнять быстро, сохраняя комфортную задержку обновления тактильной “картинки”.

Сенсорный модуль состоит из трёх ToF-датчиков расстояния, установленных по вертикали под углами +16° (верх), 0° (центр), −16° (низ). Верхний датчик отвечает за препятствия на уровне головы и плеч (например, навесы), центральный — за препятствия на уровне корпуса (стены, двери, мебель), нижний — за препятствия у земли (бордюры, ступени). Для повышения информативности без увеличения числа датчиков применяется программная технология ROI (Region of Interest, область интереса): каждый датчик последовательно измеряет расстояние в трёх горизонтальных секторах (лево/центр/право). Таким образом, за один цикл получаются 9 значений, и на их основе строится виртуальная матрица сканирования 3×3, которая затем напрямую отображается на матрицу вибромоторов на рукоятке.

2.2 ВЫБОР СЕНСОРОВ

Ключевым инженерным решением, определяющим успех проекта, является выбор технологии, способной обеспечить устойчивое измерение расстояния и возможность формирования нескольких направлений для построения матрицы 3×3. На раннем этапе рассматривались два основных подхода: ультразвуковые датчики расстояния и оптические ToF-сенсоры [2]. Для выбора подхода было составлено сравнительное сопоставление параметров, важных именно для УДТИП (см.таб. 1)

Таблица 1. Сравнение ультразвукового и оптического ToF-подхода

Характеристика	Ультразвуковой датчик (например, DFRobot SEN0591, US-016)	Оптический ToF-сенсор (например,VL53L1X)
Принцип работы	Измерение времени прохождения звука	Измерение времени прохождения инфракрасного света
Разрешение	Одна зона, одна точка данных	Одна зона, одна точка данных
Угол обзора	Узкий луч, до 15°, или широкий "конус" 30-60°	27° по диагонали
Дальность	До 3 м	До 4 м

Ключевое ограничение	Низкое угловое разрешение, невозможность создания карты глубины	Требует более сложной обработки данных

Проанализировав таблицу № 1, можно сделать вывод, что для прототипа УДТИП оптимальным является оптический ToF-подход, так как он позволяет реализовать секторизацию поля зрения и получить данные для построения матрицы 3×3. Поэтому в проекте выбран сенсор VL53L1X в количестве трёх штук, установленных вертикально, и применена программная технология ROI для формирования трёх горизонтальных секторов на каждом датчике. Это решение обеспечивает достаточно информативную “карту” из 9 зон.

2.3 ТАКТИЛЬНАЯ ОБРАТНАЯ СВЯЗЬ

Для преобразования сложной пространственной информации в интуитивно понятные тактильные ощущения будет использоваться массив актуаторов, размещенный в рукоятке устройства. Чтобы выбрать наиболее эффективные моторы, я сравнил два основных типа:

Таблица 2. Сравнение моторов

Характеристика	ERM-моторы (Eccentric Rotating Mass)	LRA-актуаторы (Linear Resonant Actuator)
Принцип работы	Вращение несбалансированного груза	Колебания магнитной массы на пружине
Время отклика	Длительное время старта/остановки (20-50 мс)	Быстрое время старта/остановки (около 10 мс)
Энергоэффективность	Менее эффективны, потребляют больше энергии	Более энергоэффективны, особенно на резонансной частоте
Сложность управления	Простое управление постоянным током	Требуется переменный ток на резонансной частоте
Диапазон эффектов	Ограниченная вибрация	Широкий спектр четких и разнообразных эффектов

Проанализировав таблицу № 2, я пришел к выводу, что идеальным выбором для моего проекта являются LRA-актуаторы. Их быстрое время отклика и способность создавать короткие и четкие тактильные сигналы позволяют передавать временные сигналы, что, согласно исследованиям, имеет решающее значение для человеческого восприятия пространственной информации [2, 3]. Это позволит создать информативный тактильный «язык» для пользователя.

2.4 ЭЛЕКТРОНИКА И ПИТАНИЕ

Выбор электроники и подсистемы питания критически важен для обеспечения производительности и автономности устройства. Для обработки данных с массива сенсоров и

управления тактильной обратной связью в реальном времени требуется мощный 32-битный микроконтроллер. Простые 8-битные платы, такие как Arduino Uno, часто не подходят по скорости и объему памяти: прошивка с библиотеками VL53L1X, обменом по I2C и дополнительной логикой нередко не помещается в доступную Flash-память и работает медленнее, чем требуется [4]. Поэтому сравним два подходящих семейства MCU: Teensy 4.1 и ESP32-S3:

Таблица 3. Сравнение микроконтроллеров

Характеристика	Teensy 4.1	ESP32-S3
Ядро	Cortex-M7 (600 МГц)	Tensilica Xtensa LX7 (до 240 МГц)
Память	1 МБ ОЗУ 8 МБ Flash	512 КБ + 16 КБ RTC ОЗУ 8 МБ Flash
Производительность	ARM Cortex-M7	Xtensa LX7
Энергопотребление	Высокое (100+ мА на 600 МГц)	Ультранизкое (нА в режиме Standby)

Проанализировав таблицу № 3 можно сделать вывод, что оба семейства, обладая высокой производительностью и достаточным объемом памяти для обработки данных, идеально подходят для нашего проекта.

Для обеспечения длительной автономной работы в УДТИП будут использоваться литий-ионные или литий-полимерные аккумуляторы. Их ключевым преимуществом является высокая удельная плотность энергии (до 330 Вт·ч/кг) и компактность. Чтобы обеспечить приемлемое время работы, будут рассмотрены батареи с емкостью не менее 1500–2000 мА·ч.

Таблица 4. Сравнение емкостей аккумуляторов

Тип аккумулятора	**Емкость**	**Ориентировочная розничная цена**
Литий-ионные или литий-полимерные аккумуляторы	1500 мА·ч	~$9.99
	1800 мА·ч	~$12.99
	3000 мА·ч	~$15.00–$25.00

Проанализировав таблицу № 1.4.2, литий-ионные или литий-полимерные аккумулятор с емкостью 3000 мА·ч стал идеальным выбором для данного проекта.

2.5 КОНСТИТУЦИЯ

Прототип УДТИП выполнен в удобном и продуманном корпусе: в передней части размещается сенсорный модуль, а в рукоятке — тактильный модуль и питание. Такая схема

нужна, чтобы датчики “смотрели” вперёд, а вибрация передавалась прямо в ладонь.

Компоновка основных компонентов показана на рис.8.1. Аккумулятор 18650 расположен вдоль корпуса ближе к рукоятке для баланса по массе. Платы электроники установлены на монтажных площадках (условно верхняя и нижняя): это упрощает сборку и обслуживание. Сенсоры размещены в передней зоне вертикальным рядом (верх/центр/низ), что соответствует трем уровням сканирования.

На рисунке 2 показан тот же прототип под углом: хорошо видно размещение матрицы вибромоторов 3×3 на внутренней стенке рукоятки. Такое расположение выбрано для максимальной проводимости тактильного сигнала в зоне хвата. В конструкции важно разнести силовую проводку моторов и сигнальные линии I2C (SDA/SCL), сохраняя общий провод GND единым, чтобы снизить помехи и избежать сбоев датчиков.

Также, изображены (см. рис. 10) все этапы от натурной модели до создания готового прототипа мобильного помощника УДТИП.

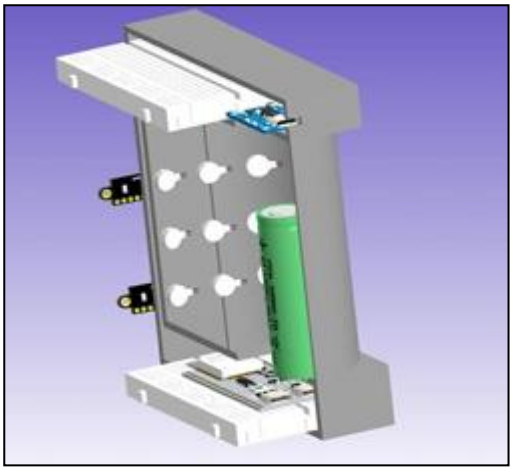


Рис. 8. 3Д модель, вид 1

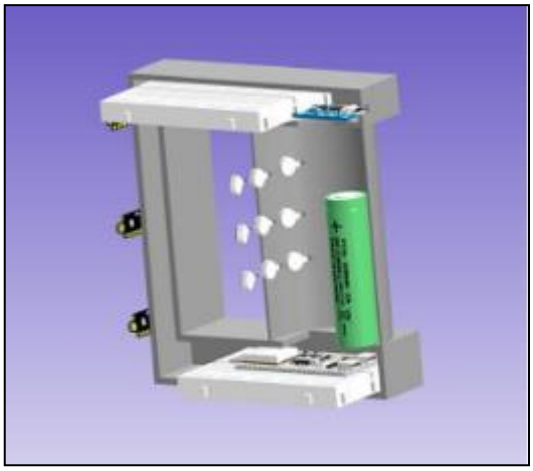


Рис. 9. 3Д модель, вид 2

Рис. 10. Конструкция УДТИП

ГЛАВА 3. РЕЗУЛЬТАТЫ И ВЫВОДЫ

3.1 ОБЛАСТЬ ПРИМЕНЕНИЯ

УДТИП предназначен для повышения безопасности и самостоятельности людей с глубокими нарушениями зрения за счет дистанционного обнаружения препятствий и передачи информации в тактильной форме. Устройство может использоваться как средство технической реабилитации, дополняющее традиционные навыки ориентирования и белую трость: оно не заменяет трость, а расширяет «дальность предупреждения», позволяя заранее оценивать обстановку перед собой. Это особенно важно для препятствий, которые трость выявляет поздно или не выявляет вовсе (навесы/выступы на уровне головы и груди, препятствия на уровне корпуса, отдельные перепады высоты).

Целевые группы применения включают людей с нарушениями зрения, возникшими вследствие заболеваний (возрастные изменения, диабетические осложнения), травм, аварий и последствий военных конфликтов, включая СВО. Для части людей, получивших травмы зрения, устройство может быть полезно на этапе адаптации и восстановления навыков самостоятельного передвижения (реабилитационные программы, занятия по ориентированию и мобильности).

3.2 ПЕРСПЕКТИВЫ РАЗВИТИЯ

Перспективы развития проекта разделяются на аппаратные, программные и конструктивные улучшения (см. рис. 11). Аппаратная часть: переход от макетной сборки к печатной плате (повышение надежности, уменьшение массы и количества проводов), добавление фильтрации питания и развязки силовой части вибромоторов от логики (уменьшение помех, повышение стабильности I2C), унификация разъёмов и креплений датчиков для повторяемости сборки.

Сенсорная часть: уточнение углов установки датчиков и параметров ROI, подбор режимов дальномера под реальную среду (компромисс дальность/скорость/шум измерений). При необходимости повышения детализации возможно расширение «матрицы пространства» (например, больше секторов по горизонтали) или переход на многозонные сенсоры, однако это должно сопровождаться проверкой того, что пользователю по-прежнему легко интерпретировать тактильный «рисунок».

Программная часть: улучшение классификации препятствий (устойчивость решений во времени, снижение ложных срабатываний), введение пользовательских профилей чувствительности (интенсивность, дальность реакции), расширение набора тактильных шаблонов и обучение пользователя «словарю сигналов». Отдельно перспективно подключение инерциального датчика (IMU) для компенсации наклона руки и повышения корректности распознавания препятствий в нижней зоне и перепадов высоты.

Рис. 11. Перспективы развития

3.3 РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

В ходе работы реализован функциональный прототип, работающий по принципу «сенсорный ввод → обработка → тактильный вывод». Сенсорный модуль построен на трёх ToF-датчиках, размещенных вертикально под углами +16° (верх), 0° (центр), −16° (низ). Для повышения информативности применена программная область интереса (ROI): каждый датчик последовательно измеряет расстояние в трёх горизонтальных секторах (лево/центр/право). Таким образом формируется виртуальная матрица сканирования 3×3 (9 значений за цикл), которая далее отображается на матрицу 3×3 вибромоторов в рукоятке.

Обработка данных выполняется микроконтроллером по шине I2C. Для подключения нескольких одинаковых датчиков на одну шину используется последовательная инициализация и адресация (через XSHUT), что устраняет конфликт стандартных адресов. Тактильный вывод реализован как независимое управление девятью каналами: базовый режим — градиентная вибрация «ближе — сильнее», дополнительно могут применяться предупреждающие шаблоны для выделения опасных ситуаций.

Дополнительно, прототип мобильного ассистента-помощника был представлен на собрании Совета ветеранов, где проведены демонстрационные испытания с участниками старшего возраста. Испытания показали, что 3 из 4-х участников с закрытыми глазами смогли

преодолеть маршрут с препятствиями «проем», «стена спереди», «навес», ориентируясь по тактильной обратной связи устройства. Полученный результат является первичной апробацией (малое число участников и упрощённые условия), но подтверждает работоспособность выбранной концепции (см. рис. 12, 13).

Рис. 12. Экспериментальная апробация УДТИП в Совете ветеранов

Рис. 13. Экспериментальная апробация УДТИП в Совете ветеранов

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Всемирная организация здравоохранения. Слепота и нарушения зрения (Blindness and vision impairment) [Электронный ресурс]. — 2023. — URL: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment (дата обращения: 02.07.2025).
2. STMicroelectronics. VL53L1X. Датчик измерения расстояния по времени пролёта (Datasheet) [Электронный ресурс]. — 2024. — URL: https://www.st.com/resource/en/datasheet/vl53l1x.pdf (дата обращения: 02.07.2025).
3. Pololu. Библиотека Arduino для VL53L1X (репозиторий) [Электронный ресурс]. — URL: https://github.com/pololu/vl53l1x-arduino (дата обращения: 04.07.2025).
4. Espressif Systems. ESP32-S3 Series. Описание микросхемы (Datasheet) [Электронный ресурс]. — URL: https://www.espressif.com/sites/default/files/documentation/esp32-s3_datasheet_en.pdf (дата обращения: 05.07.2025).
5. Orientatech. Sunu Band: описание устройства [Электронный ресурс]. — URL: https://www.orientatech.es/en/sunu-band/ (дата обращения: 05.07.2025).
6. WeWALK. Руководство пользователя (User Guide) [Электронный ресурс]. — URL: https://wewalk.io/en/guide/ (дата обращения: 05.07.2025).
7. Прата С. Язык программирования C++ : лекции и упражнения : рассматривается новый стандарт C++11 : перевод с английского / С. Прата. — 6-е изд. — Москва ; Санкт-Петербург : Диалектика, 2018. — 1244 с. — ISBN 978-5-907114-00-5.

ПРИЛОЖЕНИЕ 1. ЛИСТИНГ ПРОГРАММЫ

```
1  #include <Wire.h>
2  #include <VL53L1X.h>
3
4  const int PIN_SCL = 9;
5  const int PIN_SDA = 10;
6
7  const int XSHUT_TOP    = 5;
8  const int XSHUT_MIDDLE = 6;
9  const int XSHUT_BOTTOM = 7;
10
11 const uint8_t ADDR_TOP    = 0x30;
12 const uint8_t ADDR_MIDDLE = 0x31;
13 const uint8_t ADDR_BOTTOM = 0x32;
14
15 const int motorPins[9] = {12,13,14, 15,16,17, 18,19,20};
16 const int PWM_FREQ = 25000;
17 const int PWM_RES  = 8;
18 const int motorCh[9] = {0,1,2, 3,4,5, 6,7,8};
19
20 const int MAX_MM = 2000;
21 const int MIN_MM = 150;
22
23 const uint8_t ROI_W_LR = 5;
24 const uint8_t ROI_W_C  = 6;
25 const uint8_t ROI_H    = 16;
26
27 const uint8_t ROI_CENTER_L = 151;
28 const uint8_t ROI_CENTER_C = 199;
29 const uint8_t ROI_CENTER_R = 239;
30
31 VL53L1X sTop, sMid, sBot;
33 static inline int clampi(int x, int lo, int hi){
34   if (x < lo) return lo;
35   if (x > hi) return hi;
36   return x;
37 }
38
39 int distToPWM(int mm){
40   if (mm <= 0) mm = MAX_MM;
41   mm = clampi(mm, MIN_MM, MAX_MM);
42   long pwm = (long)(MAX_MM - mm) * 255L / (long)(MAX_MM - MIN_MM);
43   return clampi((int)pwm, 0, 255);
44 }
45
46 uint16_t readROI(VL53L1X &s, uint8_t w, uint8_t h, uint8_t center){
47   s.setROISize(w, h);
48   s.setROICenter(center);
49   uint16_t d = s.readSingle();
50   if (s.timeoutOccurred()) return 0;
51   return d;
52 }
53
54 bool initOne(VL53L1X &s, int xshutPin, uint8_t newAddr){
55   digitalWrite(xshutPin, HIGH);
56   delay(10);
57   if (!s.init()) return false;
58   s.setAddress(newAddr);
59   s.setDistanceMode(VL53L1X::Medium);
60   s.setMeasurementTimingBudget(33000);
61   s.setTimeout(100);
62   return true;
63 }
```

```
65  void pwmSetup(){
66    for (int i = 0; i < 9; i++){
67      ledcSetup(motorCh[i], PWM_FREQ, PWM_RES);
68      ledcAttachPin(motorPins[i], motorCh[i]);
69      ledcWrite(motorCh[i], 0);
70    }
71  }
72
73  void setup(){
74    pwmSetup();
75
76    Wire.begin(PIN_SDA, PIN_SCL);
77    Wire.setClock(400000);
78
79    pinMode(XSHUT_TOP, OUTPUT);
80    pinMode(XSHUT_MIDDLE, OUTPUT);
81    pinMode(XSHUT_BOTTOM, OUTPUT);
82
83    digitalWrite(XSHUT_TOP, LOW);
84    digitalWrite(XSHUT_MIDDLE, LOW);
85    digitalWrite(XSHUT_BOTTOM, LOW);
86    delay(20);
87
88    initOne(sTop, XSHUT_TOP, ADDR_TOP);
89    initOne(sMid, XSHUT_MIDDLE, ADDR_MIDDLE);
90    initOne(sBot, XSHUT_BOTTOM, ADDR_BOTTOM);
91  }
92
93  void loop(){
94    uint16_t d[3][3];
95
96    d[0][0] = readROI(sTop, ROI_W_LR, ROI_H, ROI_CENTER_L);
97    d[0][1] = readROI(sTop, ROI_W_C,  ROI_H, ROI_CENTER_C);
98    d[0][2] = readROI(sTop, ROI_W_LR, ROI_H, ROI_CENTER_R);
99
100   d[1][0] = readROI(sMid, ROI_W_LR, ROI_H, ROI_CENTER_L);
101   d[1][1] = readROI(sMid, ROI_W_C,  ROI_H, ROI_CENTER_C);
102   d[1][2] = readROI(sMid, ROI_W_LR, ROI_H, ROI_CENTER_R);
103
104   d[2][0] = readROI(sBot, ROI_W_LR, ROI_H, ROI_CENTER_L);
105   d[2][1] = readROI(sBot, ROI_W_C,  ROI_H, ROI_CENTER_C);
106   d[2][2] = readROI(sBot, ROI_W_LR, ROI_H, ROI_CENTER_R);
107
108   for (int r = 0; r < 3; r++){
109     for (int c = 0; c < 3; c++){
110       int idx = r*3 + c;
111       ledcWrite(motorCh[idx], distToPWM((int)d[r][c]));
112     }
113   }
114
115   delay(20);
116 }
```

ПРИЛОЖЕНИЕ 2. ОТЗЫВ НА ЭКСПЕРИМЕНТ В СОВЕТЕ ВЕТЕРАНОВ

Московская городская общественная организация

пенсионеров, ветеранов войны, труда, Вооружённых Сил

и правоохранительных органов.

Совет Ветеранов Дмитровского района

Северного административного округа г. Москвы

127644, г. Москва, тел./факс 8-495-483-07-71

Карельский бульвар, дом 5

«___» _____ 2026г.

ОТЗЫВ

5 декабря в Совете ветеранов Дмитровского района города Москвы состоялась презентация разработки прототипа носимого ассистивного устройства для навигации людей с глубокими нарушениями зрения (УДТИП). Автором данного изобретения является учащийся 11 класса предпрофессионального проекта «Инженерный класс в Московской школе» ГБОУ г. Москвы «Школа № 444» Чеглаков Владимир.

На встрече автор разработки представил принцип работы системы: считывание расстояния до препятствий лазерными датчиками и преобразование данных в тактильные сигналы на рукоятке устройства.

Демонстрация показала практическую ценность решения и его ориентированность на повышение безопасности и самостоятельности пользователя в городской среде.

Отмечаем актуальность проекта для социальной поддержки людей с ограниченными возможностями по зрению и потенциал дальнейшего развития устройства. Мы высоко оценили инженерный подход автора, грамотную постановку задачи и уверенное владение технической частью.

Проект вызвал живой интерес у аудитории и получил немалую заинтересованность и положительный отклик. Рекомендуем в дальнейшем продолжить работу над усовершенствованием прототипа, уделив внимание удобству использования, надежности и испытаниям в реальных условиях.

Отдельная благодарность за поддержание интереса у подрастающего поколения к инженерному делу, созданию устройств для лиц старшего поколения с ограниченными возможностями, научному руководителю – Заслуженному учителю г. Москвы, преподавателю ГБОУ «Московская международная школа» Е.В. Граськиной.

Председатель Совета ветеранов **Н.Б.Жиронкин**

ПРИЛОЖЕНИЕ 3. ОТЗЫВ

Отзыв
на учебную

проектную работу
ученика 11-го класса ГБОУ Школа №444
Чеглакова Владимира Павловича

Тема работы: «Разработка мобильного помощника для незрячих. Устройство Дистанционной Тактильной Индикации Пространства (УДТИП)»

В современном обществе обеспечение равных возможностей и повышение качества жизни для людей с ограниченными возможностями здоровья является важнейшей социальной задачей, отвечающей принципам гуманизма и инклюзивного развития. Особую значимость приобретает проблема мобильности лиц с глубокими нарушениями зрения (инвалидов по зрению). Самостоятельное и безопасное передвижение в пространстве – ключевой фактор их социальной интеграции, независимости и доступа к образованию, труду, культурной жизни.

Основной целью данного учебно-исследовательского проекта является разработка, изготовление и экспериментальная апробация Мобильного помощника - функционального прототипа Устройства Дистанционной Тактильной Индикации Пространства (УДТИП) для лиц с нарушениями зрения.

Выполненная работа производит весьма сильное впечатление. Для ученика 11-го класса это исключительно зрелый, самостоятельный и очень жизненный, практически ценный проект. Видно, что автор не просто выполнил теоретическое задание, а решил реальную проблему, с которой сам столкнулся в жизни. Это очень высокий уровень проектной деятельности школьника.

Сильные стороны работы.

1. Практико-ориентированный подход «изнутри». Работа основана на личных наблюдениях и понимании конкретной жизненной ситуации. Автор не выдумал проблему, а увидел её своими глазами: как непросто жить людям «с особенностями по зрению» в цифровую эпоху. Это придаёт проекту уникальную достоверность и ценность, так как это есть конкретная инклю
2. зивная, практически значимая разработка.

2. Профессиональный уровень анализа. Для школьника проведен структурированный анализ. Автор не просто перечислил системы, но и аргументированно объяснил, почему они не подходят для жизни в современных условиях людей с особенностями по зрению. Это демонстрирует серьезное метапредметное и инженерное мышление.

3. От идеи - к реальному устройству. Самая интересная и практико-ориентированная часть - собранный своими руками работающий прототип «Мобильного помощника для слепых» - это Устройство Дистанционной Тактильной Индикации Пространства (УДТИП) . При выполнении учебной проектной работы в 11 классе автор прошёл полный цикл: от выявления проблемы и схемы на бумаге до монтажа и тестирования всех элементов устройства. Навыки работы с инструментом, понимание электрических цепей и умение довести замысел до материального результата вызывают большое уважение.

4. Ясность, логика и доступность. Работа написана и оформлена четко, логично и понятно. Все этапы — актуальность, анализ, разработка, сборка, выводы — информационно хорошо структурированы. Видно, что автор полностью владеет материалом и может ясно обьяснить значимость своей идеи любому

Рекомендации и перспективы. Разработанную и реализованную опытную модель устройства - Мобильный помощник для слабовидящих (УДТИП) нужно развивать и оптимизировать, например, сделать более эргономичный корпус на 3D-принтере, продумать оптимизацию по затратам на изготовление.

Работа ярко показывает склонность и талант автора к инженерно-техническому творчеству, рационализаторству и заботе о людях «с особенностями».

Заключение.

Это не просто «хорошая школьная проектная работа». Это пример проекта, который решает реальную задачу помощи слабовидящим и слепым. Автор проявил нестандартное мышление, исследовательскую жилку, инженерную смекалку и получил результат, который поможет инвалидам по зрению качественно жить в наше сложное время.

Работа заслуживает высокой оценки и является образцом для работ школьников по данному инклюзивному направлению.

Рецензент:

Ведущий научный сотрудник
Отделения биоэнергетики НИИ
Физ -хим биологии им. А.Н. Белозерского
МГУ им. М. В. Ломоносова

Доктор биологических наук

_____Вайс В.Б.

«07» 02 2026 г.

МГТУ

ИМ. Н.Э. БАУМАНА

1830