

**Министерство науки и высшего образования
Российской Федерации**

Государственный университет «Дубна»

Филиал «Протвино»

**МАТЕРИАЛЫ
1-ГО ЭТАПА 30-Й ЕЖЕГОДНОЙ
НАУЧНО-ПРАКТИЧЕСКОЙ КОНФЕРЕНЦИИ
СТУДЕНТОВ, АСПИРАНТОВ
И МОЛОДЫХ СПЕЦИАЛИСТОВ
ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА «ДУБНА»,
ФИЛИАЛ «ПРОТВИНО»
(г. Протвино, 8 – 10 апреля 2025 г.)**

**Протвино
2025**

УДК 62+3
ББК 94.3я431
М341

М 341 Материалы 1-го этапа 30-й ежегодной научно-практической конференции студентов, аспирантов и молодых специалистов государственного университета «Дубна», филиал «Протвино» (г. Протвино, 08-10 апреля 2025 г.): сб. материалов. – Протвино, 2025. – 86 [1] с. – Текст: электронный.

В сборнике представлены результаты научных исследований студентов филиала «Протвино» государственного университета «Дубна» в областях информационных технологий, машиностроения, гуманитарных и социальных наук.

Материалы публикуются в авторской редакции.

УДК 62+3
ББК 94.3я431

Оглавление

Ващенко В.А.

СТРУКТУРНО-СИНТАКСИЧЕСКИЙ И СЕМАНТИЧЕСКИЙ АНАЛИЗ АНГЛИЙСКИХ ПОСЛОВИЦ И ПОГОВОРОК С КОМПОНЕНТОМ “MAN” 5

Гречко О.Р.

АВТОМАТ КАЛАШНИКОВА: КАК ОРУЖИЕ СТАЛО ОДНИМ ИЗ САМЫХ УЗНАВАЕМЫХ РОССИЙСКИХ СИМВОЛОВ..... 8

Зайнутдинов С.И.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФРАКТАЛОВ В СРЕДЕ РАЗРАБОТКИ MICROSOFT VISUAL STUDIO 2022..... 14

Захаров Л.К.

ИССЛЕДОВАНИЕ СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ В РЕЖИМЕ ТОЧНОГО ПОЗИЦИОНИРОВАНИЯ 18

Королев С.В.

ЛАБОРАТОРНЫЙ ДВУХВАЛКОВЫЙ ПРОКАТНЫЙ СТАН..... 22

Краснова В.А.

ПОВЕДЕНИЕ МОЩНОСТИ СТАНДАРТНЫХ КРИТЕРИЕВ МНОЖЕСТВЕННЫХ СРАВНЕНИЙ НА НЕНОРМАЛЬНЫХ ВЫБОРКАХ 25

Липовцев И.А., Кириченко Ж.Д.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ОТЕЧЕСТВЕННЫХ И ЗАРУБЕЖНЫХ LLM-МОДЕЛЕЙ: ВОЗМОЖНОСТИ И ПРИМЕНЕНИЕ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ 32

Макаров А.О.

СРАВНЕНИЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ C++ И RUST С ТОЧКИ ЗРЕНИЯ РАЗРАБОТЧИКА 37

Минаев Н.С.

РЕАЛИЗАЦИЯ СИСТЕМЫ ОБНОВЛЕНИЯ ПРОШИВКИ НА ЯЗЫКЕ GO С ИСПОЛЬЗОВАНИЕМ ZIP-АРХИВОВ И ПРОВЕРКИ ХЭШЕЙ..... 43

Мусатов Р.В.

РЕАЛИЗАЦИЯ СИСТЕМЫ УПРАВЛЕНИЯ БИОТЕХНОЛОГИЧЕСКИМ АНАЛИЗАТОРОМ С ИСПОЛЬЗОВАНИЕМ SAN-ИНТЕРФЕЙСА НА ЯЗЫКЕ PYTHON.. 47

Никольский Ф.В., Осморский В.А.

ПРОЕКТИРОВАНИЕ ДИСПЕТЧЕРА СОБЫТИЙ НА ПЛАТФОРМЕ ПЛИС..... 50

Овсянников Е.С.

РАЗРАБОТКА ВЕБ-СЕРВИСА ДЛЯ АВТОМАТИЗИРОВАННОЙ ГЕНЕРАЦИИ И ПОДПИСАНИЯ ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСЬЮ PDF-ДОКУМЕНТОВ 56

Примакова Т.Г.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФРАКТАЛОВ В СРЕДЕ РАЗРАБОТКИ MICROSOFT VISUAL STUDIO 2022..... 60

Сизонов Т.Р.

ИСТОРИЯ КУРСКОЙ БИТВЫ..... 64

Телипов И.В.

3D-ПЕЧАТЬ: КРАТКИЙ ОБЗОР МЕТОДОВ ПЕЧАТИ ПЛАСТИКОМ И ОПЫТ ИСПОЛЬЗОВАНИЯ FDM-ТЕХНОЛОГИИ..... 69

Тихомиров М.А.

СЦИЕНТИСТСКАЯ МОДЕЛЬ ОБЩЕСТВА В ПРОИЗВЕДЕНИИ Ф. БЭКОНА «НОВАЯ АТЛАНТИДА» 73

Якимов Е.В.

РАСЧЁТ И АНАЛИЗ ПРОЧНОСТИ ИЗДЕЛИЙ МАШИНОСТРОЕНИЯ С ПОМОЩЬЮ CAE-СИСТЕМЫ 79

Яковлева С.И.

ИССЛЕДОВАНИЕ ВЛИЯНИЯ ТЕМПЕРАМЕНТА НА УСПЕШНОЕ ИЗУЧЕНИЕ ИНОСТРАННОГО ЯЗЫКА	84
--	-----------

СТРУКТУРНО-СИНТАКСИЧЕСКИЙ И СЕМАНТИЧЕСКИЙ АНАЛИЗ АНГЛИЙСКИХ ПОСЛОВИЦ И ПОГОВОРОВ С КОМПОНЕНТОМ “MAN”
STRUCTURAL-SYNTACTIC AND SEMANTIC ANALYSIS OF ENGLISH PROVERBS AND SAYINGS WITH THE COMPONENT “MAN”

*Филиал «Протвино» государственного университета «Дубна»
Секция «Социальные и гуманитарные науки»*

Автор: Ващенко Вера Александровна, студентка 1 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: к.филол.н. Леонова С.А., доцент кафедры общеобразовательных дисциплин филиала «Протвино» государственного университета «Дубна»

Author: Vashchenkova Vera Aleksandrovna, 1st year student of the Informatics and Computer Engineering program of the branch "Protvino" state University "Dubna".

Scientific adviser: Leonova Svetlana Anatolevna, Cand. Sci. (Philology), Assoc. Prof., the Department of General Subjects, of the branch "Protvino" state University "Dubna".

Аннотация

Статья посвящена анализу английских пословиц и поговорок с компонентом “man”. Отобран языковой материал, рассмотрены употребленные в нем формы и синтаксические функции компонента, а также структура выражений. Выделены и проиллюстрированы примерами тематические группы пословиц и поговорок. В статье использованы методы сравнительного анализа, наблюдения, количественного анализа и обобщения.

Abstract

The article is dedicated to the analysis of the English proverbs and sayings with the component “man”. The language material is selected, the forms used in it and the syntactic component functions as well as the expressions structures are considered. The proverbs and sayings thematic groups are defined and illustrated with examples. The methods of comparative analysis, observation, quantitative analysis and generalization are applied.

Ключевые слова: пословицы с компонентом “man”; синтаксис пословиц и поговорок; формы компонента “man”; синтаксическая функция компонента в пословице; тематические группы пословиц и поговорок.

Keywords: proverbs with the component “man”; proverbs and sayings syntax; “man” component forms; syntactic component function in the proverb; proverbs and sayings thematic groups.

Изучению фразеологии, в частности, пословиц, посвящено большое количество работ – например, повелительным структурам в их составе [6], семантике [5], а также фразеологическим единицам с конкретными компонентами [1].

Анализ отдельных, ранее не исследованных корпусов английских пословиц, представляет интерес для лингвистики, и является актуальным и оригинальным проектом.

На занятиях по иностранному языку в текстах мы встречали пословицы и поговорки, во многих из которых повторялось слово man. Мне стало интересно, есть ли какие-то общие черты у таких пословиц, как они строятся и какие темы освещают. Преподаватель поддержал мою идею и сказал, что нужно не просто поговорить о пословицах, а выработать собственные основания, по которым мы будем их исследовать.

Цель работы – изучить характеристики этих пословиц и дать им описание с точки зрения структуры, синтаксиса и семантики.

Для реализации цели были поставлены задачи:

- выделить группу пословиц и поговорок с компонентом man;
- рассмотреть историю слова man;

- провести анализ используемых форм компонента *man*, а также структуры пословиц и поговорок с ним;

- определить синтаксическую функцию слова *man* в пословицах;

- выделить тематические группы пословиц и поговорок.

Таким образом, объект исследования – английские пословицы и поговорки со словом *man*, а предмет – формы и синтаксис данного компонента, а также семантика единиц, содержащих его в своем составе.

Языковым материалом для исследования стали учебники по английскому языку для 1 и 2 курсов [9, 10], сборник коротких рассказов английских и американских авторов [7], а также сборники пословиц и поговорок на специализированных сайтах [2, 3]. Было выявлено всего 48 фразеологических единиц, из них 43 пословицы и 5 поговорок (89,6% и 10,4%).

Согласно «Словарю лингвистических терминов» О.С. Ахмановой, пословица – «образное законченное изречение, имеющее назидательный смысл» [4, с.341], а поговорка – иносказательное выражение аналогичного рода, «отличающееся от пословицы своей синтаксической незаконченностью» [4, с.328], т.е. пословица – это предложение, а поговорка – часть предложения, словосочетание.

По материалам этимологического словаря Dictionary.com – ведущего в мире цифрового словаря, который постоянно обновляется командой опытных лексикографов [8], первые упоминания слова *man* отмечены до 900 г.; произошло от протогерманского “*mann*” – «человек». В древнеанглийском имело значение «человек», «слуга», «вассал», в среднеанглийском – «человек», «человечество», «взрослый мужчина», «муж». Родственно словам:

Mann – из немецкого языка;

Mathr – из древнескандинавского;

Manna – из готского;

Mānu – из санскрита;

Maż – из польского;

Муж, мужик – из русского.

Слово *man* входит в так называемый список Сводеша (список из 100 базовых, важнейших слов каждого языка, по ним определяется родство – степень схожести наиболее устойчивого базового словаря разных языков).

В процессе анализа было выявлено использование трех форм данного слова:

1) Единственного числа – в 31 единице (64,6%):

A **man** of words and not of deeds is like a garden full of weeds. – Противник дел, любитель слов, подобен саду без плодов.

2) Множественного числа – в 10 пословицах и поговорках (20,8%):

Gluttony kills more **men** than the sword. – От чревоугодия гибнет больше людей, чем от меча.

3) Притяжательного падежа – в 7 единицах (14,6%):

Children are poor **men's** riches. – На что и клад, когда дети идут в лад.

Притяжательный падеж существительного во множественном числе использован только в 1 пословице (см. выше), в остальных 6 случаях (85,7%) употребляется форма единственного числа “*man's*”:

One **man's** meat is another man's poison. – Что для одного еда, то для другого яд. – Что полезно одному, то для другого вредно / Что русскому здорово, то немцу смерть.

Что касается структуры пословиц с компонентом “*man*”, представленных в количестве 43 предложений, 31 предложение (или 72,1%) является простым:

No **man** is wise at all times. – На всякого мудреца довольно простоты / Безумье и на мудрого бывает / И на большие умы живет промашка / Человека без ошибок не бывает.

Сложных предложений всего 12 (27,9%):

Men may meet but mountains never. – Гора с горой не сдвинется, а человек с человеком свидится.

Среди сложных предложений отмечено 4 сложносочиненных (33,3%):

Man proposes but God disposes. – Человек предполагает, а Бог располагает.

Сложноподчиненных предложений в 2 раза больше – 8 (или 66,7%):

Money often unmakes the **men** who make it. – Деньги часто губят тех, кто их наживает.

5 выделенных поговорок представляют из себя словосочетания, в которых компонент “man” является либо ядерным (главным, смысловнесущим) словом:

Every **man** to his taste. – На вкус и цвет товарищей нет / О вкусах не спорят.

Либо определением к ядерному слову:

As plain as the nose on a **man**'s face. – Ясно, как день.

Интересен анализ синтаксической функции в 43 отобранных нами пословицах (5 поговорок являются словосочетаниями, а не полноценными предложениями, в то время как синтаксическую роль слова можно определить только в предложении).

В процессе исследования выявлен интересный случай – в одной из пословиц компонент “man” встречается 2 раза в разных функциях – подлежащего и части сказуемого (см. примеры):

1) Чаще всего слово “man” в различных формах является подлежащим – в 26 пословицах (59,1%):

A **man** can die but once. – Двум смертям не бывать, а одной не миновать.

A hungry **man** is an angry **man**. – Если голоден – значит, зол.

2) Следующей по частотности функцией стало дополнение – в 10 пословицах (22,7%):
Early to bed and early to rise makes a **man** healthy, wealthy and wise. – Кто рано ложится и рано встает, здоровье, богатство и ум наживет / Кто рано встает – тому Бог подает.

3) В 6 случаях (13,6%) является определением (или частью определения):
The voice of one **man** is the voice of no one. – Один в поле не воин / Один гусь поле не вытопчет.

4) Реже всего слово “man” входит в состав сказуемого – 2 случая (4,6%):

A hungry **man** is an angry **man**. – Если голоден – значит, зол.

If my aunt had been a **man**, she would have been my uncle. – Кабы бабушка не бабушка, так была б она дедушкой / Если бы, да кабы во рту росли бобы, был бы не рот, а целый огород.

Выделение тематических групп пословиц и поговорок проводилось на основе определения общего посыла всего английского выражения и основывалось на дословном переводе и его внешних грамматических особенностях. Таким образом, было описано 6 групп:

1) Наиболее многочисленная тематика – это «Закономерности (правила) человеческой жизни» – 35 или 72,9%:

Time and tide wait for no **man**. – Время никого не ждет.

2) «Совет, рекомендация» – выражения, построенные преимущественно в повелительном наклонении (4 или 8,3%):

Give every **man** thy ear, but few thy voice. – Слушай больше, говори меньше.

3) «Сравнение» – 4 выражения (8,3%):

Bacchus has drowned more **men** than Neptune. – В стакане тонет больше людей, чем в море / Больше людей погибает в вине, нежели в воде / Кто вино любит, тот сам себя губит.

4) «Характеристика человека (описание)» – также немногочисленная группа (3 или 6,3%):

Like master, like **man**. – Каков хозяин, таков и слуга / Каков хозяин, таковы у него и собаки.

5) «Собственное видение» – 1 поговорка (2,1%):

To measure another **man**'s foot by one's own last. – Мерить на свой аршин / Мерять по себе.

6) «Ясно, понятно» – 1 поговорка (2,1%):

As plain as the nose on a **man**'s face. – Ясно, как день.

На основании проведенного анализа сделаны следующие выводы:

- группа пословиц и поговорок со словом man является старинной, назидательный смысл выражается преимущественно при помощи простых предложений;

- чаще всего (почти в 2/3 всех случаев) компонент “man” используется в форме единственного числа, в каждом 5-м выражении в форме множественного, и лишь изредка в притяжательном падеже;

- если пословица представляет из себя сложное предложение, то в 2/3 случаев оно сложноподчиненное;

- чаще всего компонент “man” в пословице является подлежащим (более чем в половине случаев), каждый 5-й – дополнением, реже – определением или его частью, и очень редко – частью сказуемого;

- особенностью и новизной данного исследования является выделение 6 тематических групп пословиц и поговорок. Самой частотной группой являются обычные полные предложения, выражающие общие закономерности и правила жизни; реже – советы и рекомендации в повелительном наклонении, предложения со сравнительной конструкцией и смыслом, а также описывающие человека, поговорки, выражающие собственное видение и очевидность чего-либо.

Детальное изучение и описание корпуса пословиц и поговорок с компонентом “man” может быть использовано при прохождении одноименной темы в рамках дисциплины «Иностранный язык», а также вносит свой вклад в изучение фразеологии и может стать основой для дальнейшего, более узкоспециализированного исследования в данной области.

Библиографический список

1. Алексеева Т.Е. Английские пословицы с компонентами-числительными: структурно-семантический анализ // Вестник Московского государственного областного университета. Серия: Лингвистика. 2022. № 4. С.6–17. DOI: 10.18384/2310-712X-2022-4-6-17.
2. Английские пословицы и поговорки. URL: <https://skyeng.ru/articles/anglijskie-poslovicy-i-pogovorki/> (дата обращения: 18.03.2025).
3. Английские пословицы и поговорки. URL: <https://english-proverbs.narod.ru/english-proverbs-j.html> (дата обращения: 20.03.2025).
4. Ахманова О.С. Словарь лингвистических терминов. 5-е изд. М.: Книжный дом «ЛИБРОКОМ», 2010. 576 с.
5. Гурин В.В., Батицкая В.В. Семантический анализ пословиц и поговорок с лексическим компонентом «heart» в английском языке // Litera. 2020. № 1. DOI: 10.25136/2409-8698.2020.1.29066. URL: <https://cyberleninka.ru/article/n/semanticheskij-analiz-poslovits-i-pogovorok-s-leksicheskim-komponentom-heart-v-angliyskom-yazyke/viewer> (дата обращения: 15.03.2025).
6. Косоножкина Л.В. Повелительные структуры английских пословиц // Филологические науки. Вопросы теории и практики. Тамбов: Грамота, 2017. №1 (67): в 2-х ч. Ч.2. С.133-136. URL: <https://www.gramota.net/article/phil20170138/fulltext> (дата обращения: 10.03.2025).
7. A book for reading and discussion / сост. Э.Л. Хавина. М. : Издательство «Менеджер», 2009. 208 с.
8. Dictionary.com – digital dictionary. URL: <http://www.dictionary.com> (дата обращения: 19.03.2025).
9. English Unlimited : B1 Pre-intermediate Coursbook with e-Portfolio (+ DVD-ROM) / A. Tibury. Cambridge : University Press, 2010. 160 p. : ill. ISBN 978-0-521-69777-4.
10. Streamline English. Destinations. An intensive English course for intermediates Students Edition / HartleyB., Viney P. Oxford University Press, 1997. 95 p.

УДК 623.44

Гречко О.Р.

АВТОМАТ КАЛАШНИКОВА: КАК ОРУЖИЕ СТАЛО ОДНИМ ИЗ САМЫХ УЗНАВАЕМЫХ РОССИЙСКИХ СИМВОЛОВ

**KALASHNIKOV ASSAULT RIFLE: HOW THE WEAPON BECAME ONE OF THE MOST
RECOGNISABLE RUSSIAN SYMBOLS**

*Филиал «Протвино» государственного университета «Дубна»
Секция «Социальные и гуманитарные науки»*

Автор: Гречко Олег Романович, студент 1 курса, направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна»

Научный руководитель: Борисова Татьяна Юрьевна, магистр психолого-педагогического образования, ассистент кафедры общеобразовательных дисциплин филиала «Протвино» Государственного Университета «Дубна».

Author: Grechko Oleg Romanovich, 1st year student of the Informatics and Computer Engineering program of the branch "Protvino" state University "Dubna".

Scientific adviser: Borisova Tatyana Yurievna, Master of Psychological and Pedagogical Education, Assistant of the Department of General Subjects, of the branch "Protvino" of the state university "Dubna".

Аннотация

Автомат Калашникова – один из самых популярных технических «брендов» России. Его популярность объясняется высокой эффективностью применения при простоте конструкции и надежности использования в различных условиях. Создатель автомата, Калашников М.Т., объяснял высокую надежность и долговечность своего детища фразой, ставшей крылатой: «солдат академий не кончает». Именно поэтому за очень короткий срок АК оказал существенное влияние на ход мировой истории: был принят на вооружение больше ста стран мира и стал настоящим символом борьбы за справедливость, и, в конечном счете, символом России.

Abstract

Kalashnikov assault rifle is one of the most popular technical "brands" in Russia. Its popularity is due to the high efficiency of application with simplicity of design and reliability of use in various conditions. The creator of the machine, Kalashnikov M.T., explained the high reliability and durability of his brainchild with a phrase that became winged: "the soldiers of the academies do not end." That is why, in a very short time, AK had a significant impact on the course of world history: more than a hundred countries of the world were adopted and became a real symbol of the struggle for justice, and, ultimately, a symbol of Russia.

Ключевые слова: автомат Калашникова - достижение русской оружейной мысли, история и конструкция, государственная символика и культурный символ.

Keywords: Kalashnikov assault rifle - achievement of Russian weapon thought, history and design, state symbols and cultural symbol.

«Я создавал оружие для защиты своей страны.

И это — основная цель, которой оно служит.

А за то, что АК попадает в горячие точки,
должны отвечать не оружейники, а политики,
которые не могут решать проблемы мирным путём.

Стреляет не оружие. Стреляет человек.»

Калашников М.Т.

Актуальность работы: в работе рассматривается история создания и распространения автомата Калашникова как одного из признанных всем миром технических брендов нашей страны, ставшем одним из важнейших элементов российского самосознания, поскольку всякое самосознание осуществляется через артефакты — инфраструктуру, индустрию, памятники культуры. В техногенной цивилизации таким значимым артефактом является собственность на технические бренды и автомат Калашникова наряду с космическим кораблем «Союз» стали важнейшими моментами самосознания (1) не только в нашей стране, но и во многих странах, борющихся за свою свободу и независимость. Поэтому и сегодня автомат Калашникова как символ непреклонной воли к борьбе за свободу и независимость становится частью общественного сознания развивающихся стран мира.

Проблема исследования: исследовать историю создания и распространения автомата Калашникова как культурного символа, отражающего борьбу за независимость, силу и выживание в сложных условиях.

Объект исследования: история создания и распространения автомата Калашникова.

Предмет исследования: особенности становления технического бренда автомата Калашникова в качестве части общественного сознания развивающихся стран мира.

Цель работы: исследование феномена автомата Калашникова как культурного символа развивающихся стран мира.

Задачи работы:

- изучить историю и конструкционные особенности создания автомата Калашникова;
- показать географию и причины распространения автомата Калашникова;
- выявить и доказать варианты использования технического бренда автомата Калашникова как культурного символа развивающихся стран мира.

Гипотеза: уникальность конструктивных особенностей технического бренда автомата Калашникова создают основания для надежного его использования борцами за независимость развивающихся стран мира и становления в качестве части общественного сознания и культурного символа.

Методы исследования: анализ и синтез исследуемого материала; эмпирические методы описания, сравнения и классификации; теоретические методы обобщения.

Когда мы говорим о российской цивилизации, мы интуитивно имеем ввиду в первую очередь культурный слой нашей жизни. Что такое русская культура? Это наша литература, наше искусство, наше православие. Наука – тоже часть русской культуры. Атомная энергетика, космические аппараты, полет Юрия Гагарина – это все достижения русской культуры и русской цивилизации. Разработку оружия мы также причисляем к наукоемким областям, и в этой области нам есть чем гордиться. Создание новых видов вооружения является в России непрерывным процессом, позволяющим обеспечивать нужды российской армии полностью своими силами. Весь этот потенциал позволяет российской цивилизации отстаивать свои геополитические интересы и продвигать идею многополярного мира. Среди наших «технических брендов» немало известных по всему миру. Но есть один бренд, который перешагнул черту массовости, став не просто оружием, а символом. Мы говорим об автомате Калашникова. Простое, надежное оружие, известное в каждом уголке земного шара.

Во время Второй Мировой войны страны массово переходили на оружие с промежуточным патроном. Это был патрон, по мощности больше пистолетного патрона, но меньше винтовочного. Т.е. подходящий для оружия, которое обеспечило бы скорострельность, дальность ведения огня и вдобавок обладало бы умеренной массой.

СССР также занимался разработкой оружия нового типа. В течение нескольких лет велся конкурс между различными группами оружейников. Но ни один из предложенных вариантов не соответствовал поставленным требованиям. Конкурс решили завершить ускоренными темпами и, выбрав три варианта из предложенных, провести испытания новых автоматов. В финал таким образом попали автоматы Булкина, Дементьева и Калашникова. После каждого проведенного тура конструктор получал возможность доработать автомат с учетом результатов прохождения испытаний. К каждому из автоматов было предъявлено большое количество претензий. И вот, перед третьим туром конкурса М.Т. Калашников решился на беспрецедентный шаг. Т.к. терять было уже нечего, оружейник решил внести кардинальные изменения в саму компоновку автомата, что по правилам конкурса было запрещено. Т.е. фактически Калашников предложил новый автомат! В третьем туре испытания автоматы замачивали в воде и замораживали, роняли на бетонный пол, погружали в песок. Т.е. моделировали самые разные жизненные ситуации. И автомат Калашникова продолжал стрелять несмотря ни на что. Вести стрельбу новый автомат Калашникова мог как одиночными выстрелами, так и очередью. Стрелять было удобно, к автомату легко и быстро приспосабливались разные люди. И при этом автомат Калашникова получился проще конкурентов в устройстве, а значит и дешевле в производстве.

Давайте соберем все характеристики вместе: удобство, надежность, простота, функциональность, низкая стоимость, неприхотливость в обслуживании. Все вместе создало

настолько сбалансированный эффект, что автомат всего за десять лет стал одним из самых востребованных в мире, а затем и самым узнаваемым. Триумфальное шествие гениального по своей простоте оружия продолжается и по сей день. До сих пор ни один автомат в мире не превзошел по этим характеристикам автомат Калашникова. Представьте себе: этот автомат способен выдержать самые суровые условия - от пыльных пустынь до заснеженных гор, и при этом всегда готов к бою. Его легко чистить, собрать можно за минуту, а в случае поломки - быстро отремонтировать. Неудивительно, что АК стал основным стрелковым оружием Советской армии, а затем и армий многих других стран по всему миру.

Конструкция автомата Калашникова поистине уникальна. Его основные компоненты - ствол, газовый поршень, возвратно-боевая пружина, затворная рама и ударно-спусковой механизм - работают как единый слаженный механизм. Благодаря этому АК-47 демонстрирует невероятную надежность и долговечность, выдерживая тысячи выстрелов без поломок.



Рис. 48. Основные части и механизмы автомата:

1 — ствол со ствольной коробкой, прицельным приспособлением и прикладом; 2 — крышка ствольной коробки; 3 — штык-нож; 4 — возвратный механизм; 5 — затворная рама с газовым поршнем; 6 — газовая трубка со ствольной накладкой; 7 — затвор; 8 — шомпол; 9 — цевье; 10 — магазин; 11 — пенал с принадлежностью

Рис. 1. Основные части и механизмы автомата

В самом верхнем положении (поставлен на предохранитель) переводчик огня препятствует движению назад затворной рамы, спусковой крючок и шептало одиночного огня заблокированы. В среднем положении (автоматический огонь) переводчик огня блокирует шептало одиночного огня. В самом нижнем положении (полуавтоматический огонь) спусковой крючок и шептало одиночного огня могут свободно двигаться. Как только спусковой крючок нажат, курок начинает вращаться, шептало одиночного огня блокирует курок после стрельбы. Спусковой крючок должен быть освобожден до того, как можно будет выстрелить еще. В полностью автоматическом режиме курок может двигаться пока спусковой крючок не отпустят. Курок фиксируется в нижнем положении шепталом автоспуска, чтобы не препятствовать работе затворного механизма.

Автомат Калашникова устроен по принципу поворотного затвора. То есть патрон запирается в стволе поворотом. Затвор поворачивается для запираения и отпираения с помощью двух боевых упоров. Боек может скользить назад и вперед внутри затвора. Шпилька предотвращает скольжение бойка слишком далеко назад. Вкладыш ствольной коробки соединяет ствол и ствольную коробку. Вкладыш ствольной коробки имеет выемки для боевых упоров затвора. Затвор поворачивается через паз в нижней части затворной рамы. Когда нажат спусковой крючок, курок отпускается, из-за этого ударник бьет по капсуле патрона, разбивая его и инициируя горение порохового состава в гильзе. Под действием пороховых газов

затворная рама движется назад, рычаг автоспуска выходит из-под выключателя, что обеспечивает поворот автоспуска. Действуя на курок, затворная рама поворачивает его и происходит взведение курка, который становится на шептало автоспуска. При повороте автоспуска, его рычаг встает на пути движения затворной рамы. Сцепление курка с передним шепталом не произойдет, так как спусковой крючок нажат. Не доходя до крайнего переднего положения, затворная рама выключателем действует на рычаг автоспуска и поворачивает его, при этом курок освобождается шепталом автоспуска и ударяет по ударнику и происходит выстрел. С помощью выбрасывателя, затвор хватает гильзу и выкидывает через отражательный выступ. Сжатая возвратная пружина выдвигает затворную раму вперед, это заставляет затвор толкать верхний патрон в патронник ствола. Стрельба прекращается при отпуске спускового крючка или отсутствии патронов. Цикл повторяется. Так и устроен автоматический режим.

Автомат Калашникова был создан в очень сложный момент истории. Один за другим разгорались прокси-конфликты холодной войны. И конечно, распространение АК напрямую связано с этими конфликтами. Влияние автомата Калашникова на ход мировой истории сложно переоценить. Этот надежный автомат был верным спутником революционеров, партизан и повстанцев, сыграв ключевую роль во многих конфликтах XX века. От Вьетнама до Афганистана, от Африки до Латинской Америки - везде, где шла борьба за независимость, можно было увидеть легендарный АК-47 в руках борцов за свободу.

В 1970-е появляется два новых термина - «калашниковизация» и «калаш». Термин «калашниковизация» означает быстрое оснащение автоматами стран Африки, Ближнего Востока, Центральной Азии. Ну а «калаш» - это сам АК или его копия, подделка. Войны за независимость потребовали большого количества «калашей». Именно поэтому фамилия русского конструктора стала одним из самых популярных русских слов в мире. Если посчитать все АК в мире, то эти автоматы составляют 15% от общего количества стрелкового оружия. Неудивительно, что АК попал в Книгу рекордов Гиннесса как самое распространенное оружие.

Приведем цитату из кинофильма «Оружейный барон». Герой Николаса Кейджа говорит: «Это самый популярный автомат в мире, его любят все вояки. Он изящно прост, собран из штампованной стали и фанеры, а весит всего 9 фунтов. Он не ломается, не заклинивает, не раскаляется. Он стреляет и в грязи, и в песке. Он так прост в применении, что из него может стрелять ребенок — и такое тоже случается. Советы выбили автомат на монетах, Мозамбик поместил его на свой флаг. После холодной войны «Калашников» стал для русских главной статьей экспорта, уже потом идут водка, икра и писатели-самоубийцы» [3].

Особенно хорошо, не понаслышке, знают автомат Калашникова в странах, столетиями находившихся в колониальном рабстве. Для этих стран АК стал не просто оружием, а символом борьбы и свободы. Революционеры Буркина-Фасо, партизаны Зимбабве, повстанцы Мозамбика — все они держали в руках автомат Калашникова во время своей борьбы за независимость. Подтверждением того, что автомат стал носителем мощных идей, мы видим на официальной символике этих государств. И это не шутка - гербы и флаги содержат изображение автомата Калашникова.

Коренные жители Зимбабве вели борьбу против колонистов с 1965 по 1979 год. Сейчас этот период называется «Освободительной войной народа Зимбабве». Автомат Калашникова был одним из главных орудий этого конфликта, как, впрочем, и многих других войн XX века. «Мотыга и автомат на гербе Зимбабве (в центре за щитом) символизируют борьбу за мир и демократию, а также гордость за высокое качество работы народ. Кроме того, они символизируют переход от войны к миру. Подпись гласит: «Единство, Свобода, Труд» (англ. Unity, Freedom, Work)»[2].

Мозамбик — бывшая португальская колония. Борьбу с колонистами вела организация повстанцев Мозамбика — ФРЕЛИМО. Интересно, что СССР поддерживал повстанцев в их борьбе за независимость. Бойцы ФРЕЛИМО даже проходили обучение в советских военных центрах, где, конечно же, учились стрелять из автомата Калашникова. «Описание герба Мозамбика закреплено 194 статьей Конституции, принятой в 1990 году. Согласно ей, автомат Калашникова нужен для защиты и бдительности. В свою очередь стебель кукурузы и сахарного тростника символизируют богатство, зубчатое колесо — рабочую силу и промышленность,

книга — образование, мотыга — крестьянство и производство, красная звезда — символ духа международной солидарности людей Мозамбика. Красное солнце символизирует создание новой жизни»[2]. Также Мозамбик – единственное на сегодняшний день государство, поместившее АК на свой флаг.

Восточный Тимор – государство, находившееся по индонезийской оккупацией. 25 лет борьбы, сотни тысяч убитых жителей и еще большее количество беженцев. Неудивительно, что жители Восточного Тимора символизировали АК, поместив его на свой герб. Автомат Калашникова на нём, лук и копье символизируют вековую борьбу за свободу, честь и достоинство, а также суверенитет Восточного Тимора.

Предводителя революции в Буркина-Фасо Томаса Санкары называли «Африканским Че Геварой». Санкара не просто хотел установить независимость, а провести социальную революцию. И эта борьба тоже не обошлась без автомата Калашникова. В 1984 года изображение автомата поместили на герб Буркина-Фасо. «Перекрещенные автомат и мотыга, так же, как и в Зимбабве, символизируют борьбу за мир и демократию. Раскрытая книга - огромное значение образования для будущего страны, а шестеренка - важность промышленного развития» [2]. Однако затем Санкару убили в ходе военного переворота, и в 1997 году герб был изменен, АК там больше не нашлось места.

Автомат Калашникова, без преувеличения, является одним из самых значительных изобретений в истории огнестрельного оружия. Его создание не только оказало огромное влияние на развитие стрелкового вооружения, но и изменило ход военной истории XX и XXI веков. Простота, надежность и эффективность АК сделали его предпочтительным выбором для армий и вооруженных формирований по всему миру, способствуя изменениям в тактике и стратегии ведения боевых действий.

Однако влияние автомата Калашникова выходит далеко за рамки военного применения. Он стал культурным символом, отражающим борьбу, силу и выживание в сложных условиях. Изображение АК узнаваемо во всем мире и является частью поп-культуры 21 века. Силуэт автомата можно встретить в искусстве, музыке и даже на футболках модников по всему миру. Поистине, АК - это культовое оружие, ставшее неотъемлемой частью мировой культуры. Автомат появляется в кинематографе, АК можно встретить практически во всех наиболее популярных компьютерных играх: Counter-Strike, Call of Duty, Warface. В Египте, на берегу Синайского полуострова, установлен памятник Автомату Калашникова. В некоторых африканских странах АК появляется не только на государственных символах: новорожденным детям дают имя Калаш – в честь Автомата Калашникова. Это оружие стало частью общественного сознания, символизируя как непреклонную волю к борьбе за свободу и независимость, так и трагические последствия вооруженных конфликтов.

Библиографический список

1. Абсеметов М.О. К истории создания легендарного автомата Калашникова // Вестн. Том. гос. ун-та. 2015. №392. URL: <https://cyberleninka.ru/article/n/k-istorii-sozdaniya-legendarnogo-avtomata-kalashnikova> (дата обращения: 26.03.2025).
2. «Калаш» на флаге: 5 государств, на символике которых есть АК URL: <https://voicesevas.ru/news/21425-kalash-na-flage-5-gosudarstv-na-simvolike-kotoryh-est-ak-47.html> (дата обращения: 06.04.2025)
3. Козлов В. Интересные факты об автомате Калашникова, которые не соответствуют действительности. URL: <https://sci-hit.com/2016/12/interesnye-fakty-ob-avtomate-kalashnikova-kotorye-ne-sootvetstvuyut-deystvitelnosti.html> (дата обращения: 03.04.2025).
4. Оружейный барон (Lord of War, 2005), фильм (дата обращения 18.03.2025).
5. Пирогов К.С. Размышления о структуре национального самосознания// Вестник СПбГУ. Сер. 17. 2015. Выпуск 1.
6. Филатов А.С. Российский цивилизационный очаг оси мировой культуры // Культурологический журнал 2016/4(26).

**ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФРАКТАЛОВ В СРЕДЕ РАЗРАБОТКИ MICROSOFT
VISUAL STUDIO 2022**
SOFTWARE IMPLEMENTATION OF FRACTALS IN THE MICROSOFT VISUAL STUDIO 2022
DEVELOPMENT ENVIRONMENT

*Филиал «Протвино» государственного университета «Дубна»
Секция «Информационные технологии»*

Автор: Зайнутдинов Сергей Ильгамович, студент 3 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Губаева Милета Михайловна, старший преподаватель кафедры информационных технологий филиала «Протвино» государственного университета «Дубна».

Author: Zainutdinov Sergey Ilgamovich, 3rd year student of the Informatics and Computer Engineering program of the branch "Protvino" state University "Dubna".

Scientific adviser: Gubaeva Miletа Mihailovna, Senior Lecturer at the Department of Information Technology of the branch "Protvino" state University "Dubna".

Аннотация

В статье рассматриваются фракталы и их свойства. Реализация некоторых из них в среде разработки Microsoft Visual Studio и использование Microsoft Foundation Classes.

Abstract

The article discusses fractals and their properties. The implementation of some of them in the Microsoft Visual Studio development environment and the use of the Microsoft Foundation Classes.

Ключевые слова: фракталы, программирование, компьютерная графика, Microsoft Visual Studio, Microsoft Foundation Classes.

Keywords: fractals, programming, computer graphics, Microsoft Visual Studio, Microsoft Foundation Classes.

Цели работы:

1. Изучить некоторые фракталы, которые встречаются и имеют практическое применение в жизни;
2. Написать программу в среде разработки программного обеспечения Microsoft Visual Studio для построения фракталов – методы Мандельброта и метод Жюлиа.

Объект исследования – фракталы, а **предметом данного исследования** являются фракталы в компьютерной графике.

Фракталы – это особый класс геометрических объектов, обладающих свойствами самоподобия и сложной структурой. Их изучения стали возможным с развитием фрактальной геометрии, основанной на работах Бенуа Мандельброта в 1970-х годах. Фракталы нашли применение во многих сферах знания, включая математику, физику, биологию, информатику и искусство. Фракталы позволяют описывать и моделировать сложные системы, которые невозможно описать традиционными способами.

Ещё одной характерной чертой фракталов является создание схожих паттернов при любых масштабах, что делает их всесторонне пригодными для описания природных феноменов и форм.

Среди фрактальных объектов можно назвать облака, горные цепи, реки и даже различные биологические структуры.

Помимо этого, фрактальная геометрия применяется в обработке данных, компьютерной графике и теории хаоса, что свидетельствует о её междисциплинарном характере.

Фракталы принято делить на геометрические, алгебраические и стохастические.

- **Геометрические** — строятся на основе исходной фигуры, которая определённым образом делится и преобразуется на каждой итерации.
- **Алгебраические** — строятся на основе алгебраических формул.

- **Стохастические** — образуются в том случае, если в итерационной системе случайным образом изменяется один или несколько параметров.

Рассмотреть два самых известных фрактала: **множество Мандельброта** и **множество Жюлиа**.

- **Множество Мандельброта** — это классический фрактал, который исследуется путем итерации простой комплексной функции. Каждая точка в комплексной плоскости проверяется на принадлежность этому множеству.
- **Множество Жюлиа** — это семейство фракталов, получаемых также через итерации комплексной функции, но в отличие от множества Мандельброта, параметры функции фиксированы.

Компьютерная графика – раздел информатики, предметом которого является создание и обработка на компьютере графических изображений (рисунков, чертежей, фотографий и пр.).

Для написания приложения построения фракталов в среде разработки программного обеспечения Microsoft Visual Studio 2022 использовалась библиотека Microsoft Foundation Classes. Она позволяет облегчить разработку GUI-приложений для Microsoft Windows путём использования богатого набора библиотечных классов.

Далее представлена часть программного кода, реализующая построение множества Мандельброта (Рис. 1) и множества Жулиа (Рис. 2).

```
// множество Мандельброта
#define MaxIterationsIndex 255
void DrawStudyExample(HDC hdc)
{
    RECT rc = { 0,0,500,500 };
    if (hdc == NULL) return;
    Mandelbrot(hdc, 0, 0, rc.right, rc.bottom, -2.2, 1, -1.2, 1.2);
    ReleaseDC(NULL, hdc);
}

void Mandelbrot(HDC hdc, int xx, int yy, int cx, int cy, double minX, double maxX,
double minY, double maxY)
{
    double stepX, stepY;
    int i, j, iter;
    double x, y;
    stepX = (maxX - minX) / (double)cx;
    stepY = (maxY - minY) / (double)cy;
    y = minY;
    for (j = 0; j < cy; j++)
    {
        x = minX;
        for (i = 0; i < cx; i++)
        {
            iter = Iterations(x, y);
            SetPixel(hdc, xx + i, yy + j, IndexToColor(iter));
            x += stepX;
        }
        y += stepY;
    }
}

int Iterations(double x, double y)
{
    int i;
    double xx, yy, xk, yk;
    xx = x;
    yy = y;
    i = 0;
    while (xx*xx + yy*yy < 4)
    {
        xk = xx*xx - yy*yy;
        yk = 2*xx*yy;
        xx = xk;
        yy = yk;
        i++;
    }
    return i;
}
```

```

        while (xx*xx+yy*yy<=4.0)
        {
            xk = xx * xx - yy * yy + x;
            yk = 2.0 * xx * yy + y;
            xx = xk;
            yy = yk;
            i++;
            f (i >= MaxIterationsIndex)break;
        }
        return i;
    }
    COLORREF IndexToColor(int index)
    {
        return 10000 * (MaxIterationsIndex - index);
    }
    void CMY4aView::OnDraw(CDC* pDC)
    {
        CMY4aDoc* pDoc = GetDocument();
        ASSERT_VALID(pDoc);
        if (!pDoc) return;
        DrawStudyExample(pDC->m_hDC);
    }

    // множество Жюлиа
    #define MaxIterationsIndex 255
    void DrawStudyExample(HDC hdc)
    {
        RECT rc = { 0,0,500,500 };
        if (hdc == NULL) return;
        Julia(hdc, 0, 0, rc.right, rc.bottom, -1, 1, -1.2, 1.2);
        ReleaseDC(NULL, hdc);
    }
    void Julia(HDC hdc, int xx, int yy, int cx, int cy, double minX, double maxX,
double minY, double maxY)
    {
        double stepX, stepY;
        int i, j, iter;
        double x, y;
        stepX = (maxX - minX) / (double)cx;
        stepY = (maxY - minY) / (double)cy;
        y = minY;
        for (j = 0; j < cy; j++)
        {
            x = minX;
            for (i = 0; i < cx; i++)
            {
                iter = JuliaIterations(x, y);
                SetPixel(hdc, xx + i, yy + j, IndexToColor(iter));
                x += stepX;
            }
            y += stepY;
        }
    }
    int JuliaIterations(double x, double y)
    {
        int i;
        double xx, yy, xk, yk;
        xx = x;
        yy = y;
        i = 0;
        while (xx * xx + yy * yy <= 4.0)
        {
            xk = xx * xx - yy * yy + xJulia;

```

```

        yk = 2.0 * xx * yy + yJulia;
        xx = xk; yy = yk; i++;
        if (i >= MaxIterationsIndex) break;
    } return i;
}
COLORREF IndexToColor(int index)
{
    return index0.1(MaxIterationsIndex - index);
}
void CJULIAView::OnDraw(CDC* pDC)
{
    CJULIAVDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if (!pDoc) return;
    DrawStudyExample(pDC->m_hDC);
}

```

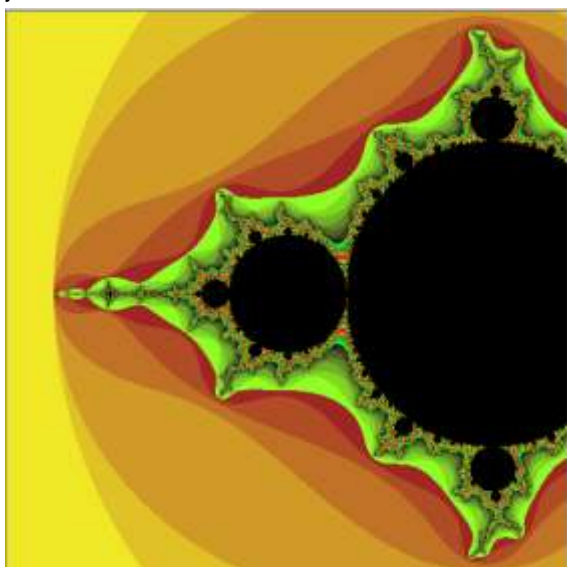


Рис. 1. Множество Мандельброта

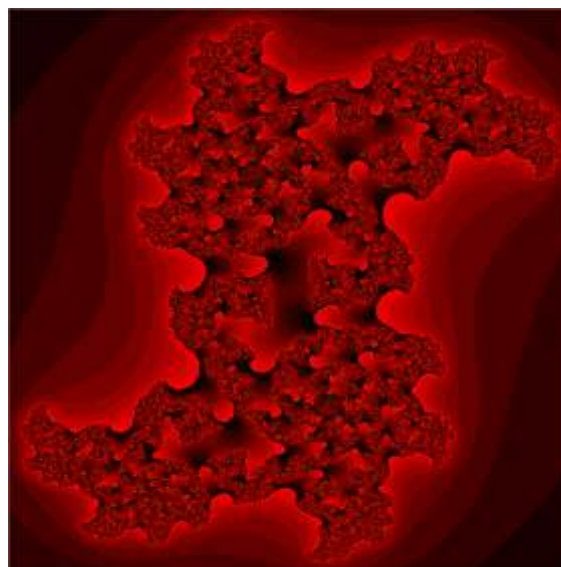


Рис. 2. Множество Жулиа

В данной работе были рассмотрены фракталы и разработаны программы для графического представления множеств Мандельброта и Жулии.

Для написания программ была использована среда разработки программного обеспечения Microsoft Visual Studio 2022 и библиотека Microsoft Foundation Classes.

На примере построения этих 2-х фракталов, зная формулы других фракталов, можно дополнять и развивать приложение.

Использование данного материала расширяет кругозор интересующихся математикой и усиливает познавательный интерес к ней, развивает пространственное представление и мышление.

Применение фракталов широко распространено в компьютерной графике при моделировании природных процессов, а также в математике и искусстве.

Библиографический список

1. Корнеев В. И. Программирование графики на C++. Теория и примеры : учеб. Пособие / В.И. Корнеев, Л.Г. Гагарина, М.В. Корнеева. - М.: ИД "Форум" : ИНФРА-М, 2019 - 517с.
2. Никулин Е. А. Компьютерная графика. Фракталы : учебное пособие для вузов / Е. А. Никулин. - 2-е изд., стер. - Санкт-Петербург : Лань, 2021. - 100 с. : ил.
3. Тараневич Ю. Ю. Математическое и компьютерное моделирование. Вводный курс: Учебное пособие. Изд. 3-е, испр. - М.: Едиториал УРСС, 2003 – 144 с.

ИССЛЕДОВАНИЕ СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ В РЕЖИМЕ ТОЧНОГО ПОЗИЦИОНИРОВАНИЯ

STUDY OF AUTOMATIC CONTROL SYSTEM IN PRECISE POSITIONING MODE

*Филиал «Протвино» государственного университета «Дубна»
Секция «Машиностроение»*

Автор: Захаров Леонид Константинович, студент 4-го курса направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Леонов Анатолий Петрович, к.т.н., доцент кафедры автоматизации технологических процессов и производств филиала «Протвино» государственного университета «Дубна».

Author: Zakharov Leonid Konstantinovich, 4th year student of the Automation of Technological Processes and Production program of the branch "Protvino" state University "Dubna".

Scientific adviser: Leonov Anatoly Petrovich, PhD, Associate Professor of the Department of Automation of Technological Processes and Production of the branch "Protvino" state University "Dubna".

Аннотация

Анализируется влияние упругих деформаций ходового винта на качество позиционирования объекта управления с точностью в единицы микрон. Рассматривается схема стенда для определения жесткости механических звеньев, приводятся экспериментальные характеристики. Рассматривается возможность учета упругих деформаций ходового винта при моделировании цифровых прецизионных систем управления.

Abstract

The influence of elastic deformations of the lead screw on the quality of positioning of the control object with an accuracy of a few microns is analyzed. The scheme of the stand for determining the rigidity of mechanical links is considered, experimental characteristics are given. The possibility of taking into account elastic deformations of the lead screw when modeling digital precision control systems is considered.

Ключевые слова:

цифровая прецизионная система управления, точное позиционирование, упругие деформации ходового винта, учет упругих деформаций ходового винта при моделировании.

Keywords:

digital precision control system, precise positioning, elastic deformations of the lead screw, taking into account the elastic deformations of the lead screw during modeling.

Проектирование современного высокопроизводительного оборудования, обеспечивающего высокую точность (как правило, это единицы микрон) позиционирования объекта управления (ОУ) в заданную координату является сложной инженерной задачей.

В прецизионных цифровых системах автоматического управления (ЦСАУ) при высокой точности (единицы микрон) позиционирования *актуальной* представляется *проблема* устранения колебаний объекта управления вокруг заданной координаты.

В подобных высокоточных системах поиск практически реализуемых алгоритмов управления осуществляется с помощью математического моделирования. Эффективность этого процесса в значительной мере определяется точностью приближения математической модели к реальной системе и теми допущениями, которые принимаются при построении такой модели.

Когда речь идет о точности перемещения ОУ в несколько микрон, среди прочего необходимо учитывать влияние упругих деформаций на точность позиционирования. Упругие деформации же обусловлены конечной жесткостью ходового винта на скручивание.

Объектом исследования в данной статье является ходовой винт, упругие деформации которого при точном позиционировании в заданную координату являются одной из причин возникающих колебаний.

Определение свойств ходового винта – предмет теоретических и экспериментальных исследований, выполненных в данной работе.

Цель исследования – выявить влияние упругих деформаций ходового винта на качество позиционирования ЦСАУ в заданную координату и необходимость их экспериментального определения для использования при построении адекватной математической модели ЦСАУ в режиме точного позиционирования.

Задачами исследования являются:

- математический анализ упругих деформаций ходового винта в режиме точного позиционирования ЦСАУ;
- разработка стенда для экспериментального определения свойств ходового винта в режиме микроперемещений;
- определение численных значений характеристик ходового винта в режиме точного позиционирования ЦСАУ.

В статье использованы материалы, полученные в результате исследований в НИЦ «Курчатовский институт» – ИФВЭ им. А.А. Логунова и в филиале «Протвино» государственного университета «Дубна».

Для начала проведем оценку устойчивости системы.

Основным демфирующим фактором в позиционной системе автоматического управления (САУ) является момент сопротивления M_c . Представим расчетную схему электромеханической части привода в виде классической двухмассовой модели (рис. 1) [2,21-24]. Обычно первую массу I образуют масса ротора двигателя и элементов между двигателем и упругим элементом, а вторую массу II – исполнительный орган и элементы между ним и упругим элементом [2,23]. Обе инерционные массы J_1 и J_2 связаны упругим элементом с коэффициентом жесткости c_B и моментом упругих сил $M_{упр}$, который характеризует сопротивление для первой массы и движущий момент для второй. В общем случае углы поворота (закручивания) вала двигателя $\varphi_{дв}$ и винта $\varphi_{в}$ не равны между собой и эта разница оказывает критичное для прецизионной САУ влияние. Движение двух массовой модели описывается системой из следующих уравнений [2,23]:

$$\frac{J_1 d_2 \varphi_{дв}}{dt^2} = M_{дв} - M_{упр} \quad (1)$$

$$\frac{J_2 d_2 \varphi_{дв}}{dt^2} = M_{упр} - M_{мс} \quad (2)$$

Произведя подстановку $M_{упр} = c_B(\varphi_{дв} - \varphi_{в})$ в выражения (1) и (2), получим уравнения движения (3) и (4) в операторной форме:

$$(T_1^2 p^2 + 1) \cdot \varphi_{дв} = \varphi_{в} + M_{дв}/c_B \quad (3)$$

$$(T_2^2 p^2 + 1) \cdot \varphi_{в} = \varphi_{дв} - M_c/c_B \quad (4)$$

где $(T_1 = \sqrt{J_1/c_B})$, $(T_2 = \sqrt{J_2/c_B})$.

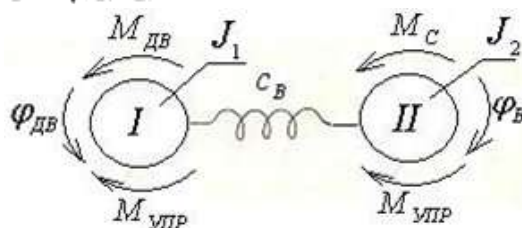


Рис. 1. Двухмассовая схема механической части позиционной системы управления

Наличие консервативных звеньев в уравнениях (3) и (4) негативно влияет на устойчивость системы управления. Необходимо, чтобы частоты, на которых будут появляться резонансные пики [3,217], не попадали в полосу пропускания проектируемой САУ. Данные резонансные частоты определяются из выражений [3,218] $\omega_{p1} = 1/T_1 = \sqrt{c_B/J_1}$ и $\omega_{p2} = 1/T_2 = \sqrt{c_B/J_2}$.

Расчет величин моментов инерции и коэффициента жесткости проводим по методике, изложенной в литературе [4,23]. Величину коэффициента жесткости ходового винта определяем по формуле:

$$c_B = \frac{G\pi d_0^4}{4l_B}$$

где $G[\text{Н/м}^2]$ – модуль упругости, $d_0[\text{м}]$ – диаметр винта, $l_B[\text{м}]$ – длина винта.

Выбирается наименьшая f_{pmin} из величин $f_{p1} = \omega_{p1}/2\pi$ и $f_{p2} = \omega_{p2}/2\pi$. Для обеспечения устойчивости работы САУ частота f_{pmin} должна в 2-3 раза превышать частоту среза f_{CP} [4,23]. Исходя из того, что в первом приближении для позиционных систем с механическими звеньями частота среза $f_{CP} = 50\text{Гц}$ является типовой [4,15], получаем, что для устойчивой работы должно выполняться условие:

$$f_{pmin} \geq 3f_{CP} \geq 150\text{ Гц} \quad (5)$$

Однако выполнение условия (5) ещё не гарантирует достижения требуемого качества управления позиционированием объекта. Необходимо также проанализировать влияние упругих деформаций на точность позиционирования при цифровом управлении.

При точном позиционировании в заданную координату за время управляющего импульса длительностью t_H необходимо обеспечить поворот вала двигателя на угол $\Delta\phi_{ТП}$, определяемый из выражения $\Delta\phi_{ТП} \geq \Delta\phi + \rho\Delta S$, где $\Delta\phi$ – угол закручивания вала; ΔS – линейное перемещение ОУ за время t_H ; ρ – коэффициент передачи шарико-винтовой пары.

Таким образом за время t_H должно произойти как закручивание вала, так и заданное линейное перемещение объекта на расстояние ΔS , равное единице отсчета цифрового датчика δ .

Для определения значений углов закручивания $\Delta\phi$ для данной конструкции механической части необходимо экспериментально определить его зависимость от момента двигателя M (то есть $\Delta\phi = f(M)$) и учесть его в процессе работы САУ.

На рис. 2 представлена функциональная схема стенда для определения жесткости механических звеньев системы. На валу двигателя крепится зеркало, предназначенное для отражения луча оптического коллиматора.

Методика определения экспериментальной зависимости $\Delta\phi = f(M)$ заключается в следующем: платформа фиксируется на упоре; к ходовому винту от электродвигателя ЭД прикладывается момент M ; под действием момента M вал двигателя вместе с винтом шарико-винтовой передачи (ШВП) [4,11] закручивается на определенный угол $\Delta\phi$, а вместе с ним перемещается и зеркало; в результате с помощью оптического коллиматора измеряется реальный угол $\Delta\phi$ закручивания ходового винта (который очевидно будет меньше угла поворота вала двигателя).



Рис. 2. Схема стенда для экспериментального определения жесткости механических звеньев

На рис. 3 приведен экспериментальный график зависимости $\Delta\phi = f(M)$. Его анализ показывает, что угол закручивания ходового винта $\Delta\phi \approx 33'$. При известных единице отсчета цифрового датчика $\delta = 5 \cdot 10^{-3}$ мм, длине перемещения $S = \delta = 5 \cdot 10^{-3}$ мм и коэффициенте передачи ШВП $\rho = 14 \cdot 10^{-3}$ мм/град величина полезного перемещения вала двигателя при точном позиционировании (отработке рассогласования величиной в единицу отсчета датчика) составит $\Delta\phi_{\text{дв}} = \delta \cdot 10^{-3} [\text{мм}] / \rho [\text{мм/град}] \approx 21.5'$. Общий угол поворота вала двигателя при точном позиционировании будет составлять:

$$\Delta\phi_{\text{ТП}} = \Delta\phi + \Delta\phi_{\text{дв}} \approx 55' \text{ и } \Delta\phi > \Delta\phi_{\text{дв}}.$$

Для обеспечения необходимой длины перемещения длительность управляющего импульса может оказаться избыточной в связи с тем, что привод уже начал движение, а вал, передающий момент ещё только в процессе скручивания. Датчик зафиксирует отсутствие перемещения, а система отправит управляющий импульс для увеличения момента, оценив происходящее, как недостаток момента. При этом момент двигателя достигнет такой величины, что объект управления «проскочит» заданную координату. Датчик зафиксирует несоответствие текущей координаты с заданной и подаст новый управляющий импульс, для перемещения в обратном направлении, и ошибка повторится. Таким образом в такой системе могут возникнуть колебания в окрестности заданной координаты позиционирования, а необходимая точность так и не будет достигнута.

Возможность предотвращения таких колебаний и возможность позиционирования в заданную координату с точностью $\pm\delta$ при используемом ходовом винте определяются при моделировании системы, когда при расчетах используются данные экспериментальной характеристики $\Delta\phi = f(M)$ (рис. 3).

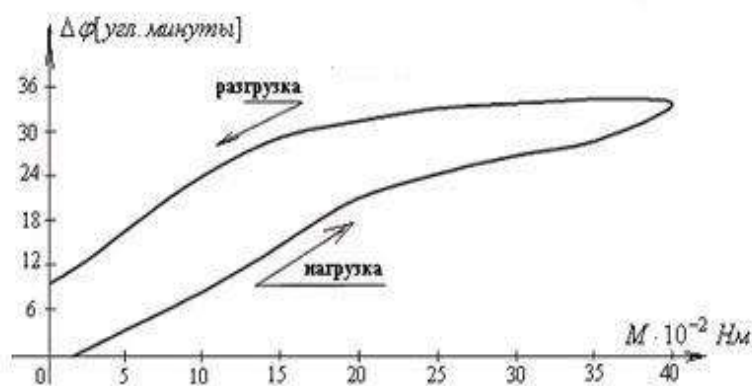


Рис. 3. Экспериментальные характеристики угла закручивания ходового вала

В ходе данной работы проанализирована актуальная проблема [4,31] учета упругих деформаций ходового винта на качество позиционирования в прецизионной цифровой системе управления перемещением объекта. Для анализа предложена двухмассовая модель механической части позиционной системы управления, определены требования к значению наименьшей из 2-х резонансных частот, обусловленных наличием консервативных звеньев. Исследован режим точного позиционирования, состоящий из процессов скручивания ходового винта и перемещения на расстояние, соответствующее заданной точности позиционирования. Для исключения колебаний объекта управления в заданной координате предлагается экспериментальное определение жесткости механических звеньев на специальном стенде и учет экспериментальной характеристики при моделировании системы.

Библиографический список

1. Бесекерский, В.А. Теория систем автоматического управления / В.А. Бесекерский, Е.П. Попов. – Изд. 4-е, перераб. и доп. – СПб., Профессия, 2007. – 217, 218 с.
2. Москаленко, В.В. Автоматизированный электропривод: Учебник для вузов / В.В. Москаленко. – М.: Энергоатомиздат, 1986. – 21-24 с.
3. Чернянский, П.М. Основы расчета динамических систем станков: Учебное пособие / П.М. Чернянский, В.В. Селезнева. – М.: Изд-во МВТУ им. Н.Э. Баумана, 1983. – 11, 15, 23, 31 с.
4. <https://root.cern/root/vmc/>

УДК 621.77

Королев С.В.

ЛАБОРАТОРНЫЙ ДВУХВАЛКОВЫЙ ПРОКАТНЫЙ СТАН LABORATORY TWO-ROLL ROLLING MILL

*Филиал «Протвино» государственного университета «Дубна»
Секция «Машиностроение»*

Автор: Королев Сергей Викторович, студент 3 курса, направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Сасов Анатолий Михайлович, к.т.н., доцент кафедры автоматизации технологических и производств филиала «Протвино» государственного университета «Дубна».

Author: Korolev Sergey Viktorovich, 3rd year student, of the Automation of Technological Processes and Production program of the branch "Protvino" state University "Dubna".

Scientific adviser: Sasov Anatoly Mikhailovich, PhD, Associate Professor of the Department of Automation of Technological Processes and Production of the branch "Protvino" state University "Dubna".

Аннотация

В работе приведена принципиальная схема двухвалкового прокатного стана, который предназначен для выполнения лабораторных работ в процессе учебы по специальности «Автоматизация технологических процессов и производств». Стан имеет небольшие габаритные размеры, низкую потребляемую мощность, безопасен в эксплуатации. Действующая модель стана используется в учебном процессе кафедры «Автоматизация технологических процессов и производств» Филиала «Протвино» государственного университета «Дубна».

Abstract

The paper presents a basic diagram of a two-roll rolling mill, which is intended for performing laboratory work during the study process in the specialty "Automation of technological processes and production". The mill has small dimensions, low power consumption, and is safe to operate. The

working model of the mill is used in the educational process of the Department of Automation of Technological Processes and Production of the Protvino Branch of the Dubna State University.

Ключевые слова: нажимной механизм, валки, гладкая бочка, ручьи, калибр, уравнивающее устройство, клеть, электропривод.

Key words: pressure mechanism, rollers, smooth barrel, grooves, caliber, balancing device, cage, electric drive.

Промышленный прокатный стан представляет собой комплекс технологического оборудования обеспечивающего обработку металлов давлением между вращающимися валками. Это высокопроизводительный, широко распространенный способ придания металлу требуемых форм и размеров. В современной индустрии до 90% выплавляемой стали и большую часть сплавов на основе цветных металлов подвергают прокатке.

Высокая эффективность этой технологии нашла соответствующее отражение в Федеративном государственном образовательном стандарте высшего образования РФ. Практически, во всех учебных дисциплинах, изучающих обработку металлов, включены разделы, посвященные изучению различных методов прокатки металлов. Так, в соответствии с планируемыми результатами изучения по дисциплине «Физические основы обработки металлов давлением», следует выполнение лабораторных работ. Наиболее эффективным методом закрепления компетенций студентов в этой области, являются практические занятия на работающем технологическом оборудовании. Таким образом, определилась актуальность настоящей работы.

Прокатный стан является одним из старейших инструментов по обработке металлов давлением. Простейшая конструкция стана состоит из двух стальных или чугунных цилиндрических валков. Обрабатываемый металл помещают между вращающимися валками. Под действием сил трения, обрабатываемая заготовка прокатывается между валками. В результате пластической деформации, металл изменяет геометрические размеры и форму. При этом происходит измельчение его микроструктуры, происходит наклеп. Металл упрочняется, становится тверже и более хрупким. Чтобы вернуть ему пластичность, необходимо нагревать до 1000 °C и более, в зависимости от его химического состава.

На прокатных станах производят как листовой металл, так и самого разнообразного профиля. Поэтому валки для получения листа имеют гладкую цилиндрическую поверхность, так называемую бочку. Для проката профиля, на валке делают кольцевые проточки - ручьи, геометрическая форма которых соответствует половинному сечению заданного профиля. Ручьи двух валков, соединившись, образуют калибр, который воспроизводит профиль изготавливаемой заготовки.

Типовые слитки металлов, проходящие первичную обработку на прокатных двухвалковых станах, имеют массу от единиц до нескольких десятков тонн. Для получения тонколистового проката используют многовалковые станы. Во всех случаях, прокатный стан это громоздкая машина, потребляющая много электроэнергии. К тому же, стальные заготовки должны иметь температуру более, чем тысяча градусов. Это создает дополнительные проблемы с безопасностью операторов находящихся в рабочей зоне. Поэтому промышленные прокатные станы не могут быть рекомендованы в качестве лабораторного оборудования.

Номенклатура серийно производимых лабораторных двухвалковых прокатных станов весьма ограничена. Так стан известной марки ДУО-360, имеет электрическую мощность главного привода 150 кВт. При этом занимаемая площадь 7000 × 4500 мм [2]. Менее габаритный стан ДУО-250, имеет все-таки достаточно большую мощность главного привода - 55 кВт и массу 8,5 тонн [3]. И наконец, стан ДУО-180, имея мощность главного привода 22кВт, обладает массой 1,8 тонн [4].

Перечисленное оборудование применяется в научно-исследовательских учреждениях, занимающихся исследованиями новых материалов и разработкой технологических процессов. В образовательных учреждениях, в процессе обучения студентов специальности «Металлургия», согласно образовательному стандарту, требуется применение полноценного технологического оборудования. Его эксплуатационные возможности должны обеспечивать реализацию реальных технологических процессов прокатки металлов, в режимах холодной и

горячей обработки металлов. В этих случаях, для этих целей, используют прокатный стан ДУО-130 [5]. Этот стан находит широкое применение в мелкосерийном производстве и при проведении научно-исследовательских работ.

Анализ технических характеристик вышерассмотренных двухвалковых прокатных станов приводит к следующему выводу. Для всех станов, серии ДУО, характерны крупные габариты, большая мощность и высокая производительность. В то время, как для учебных процессов требуется регистрация физических реакций и фазовых превращений, протекающих в металле при его обработке давлением. Поэтому прокатные станы серии ДУО, в качестве учебного оборудования применять нецелесообразно.

Конечной целью настоящей работы является проект учебного двухвалкового прокатного стана. Его технические возможности должны полностью соответствовать требованиям, предъявляемым к лабораторному оборудованию учебных заведений. Работа технологического оборудования в условиях учебного процесса кардинально отличается от заводских условий. Во-первых, учебный прокатный стан должен иметь небольшие габариты и массу. Для этих условий подходит вариант реверсивного двухвалкового стана с верхним холостым валком и нижним приводным валком. Во-вторых, обеспечивать безопасность от поражения электрическим током и механических подвижных деталей узлов привода валков. При этом прокатный стан должен быть прост в обслуживании.

Кроме этого, учитывая небольшие размеры стана, рабочую зону валков целесообразно разделить на две части. Одну часть отвести для лабораторных работ по теме листового проката, - гладкая часть, в виде бочки. Другую часть, с ручьями, для изучения физических процессов проката профиля. В электроприводе механизма нажимного вала следует использовать редуктор закрытого типа. Клеть монтировать в станинах закрытого типа [1]. Поступательное движение верхнего вала производить с помощью нажимного механизма и пружинного уравнивающего механизма.

Учитывая небольшую мощность электропривода ведущего вала, для проведения лабораторных работ с прокатным станом, следует использовать пластичные материалы. Для этих целей подходит так называемый «Умный пластилин» – хендгам, вещество на основе силикона. Его используют для лепки скульптур и художественных различных произведений. При этом он не маслянистый как традиционный пластилин, не прилипает к рукам и металлу. Выпускается под торговыми марками: Умный пластилин, Хэндгам, Фанпластик, Хэппигам.

Схема лабораторного двухвалкового прокатного стана приведена на рис. 1. В закрытом корпусе стана 1 установлены направляющие 2, в которых размещены следующие узлы и детали. В нижней части колонны, подшипники 3 ведущего вала 4, на нем установлен кулачек 5, датчика оборотов вала 7 и муфта шарнирная сдвоенная с промежуточной спаренной вилкой 6. Эта вилка соединена с выходным валом планетарного редуктора 8, интегрированного с червячным редуктором 9 с корпусом электромотора 10.

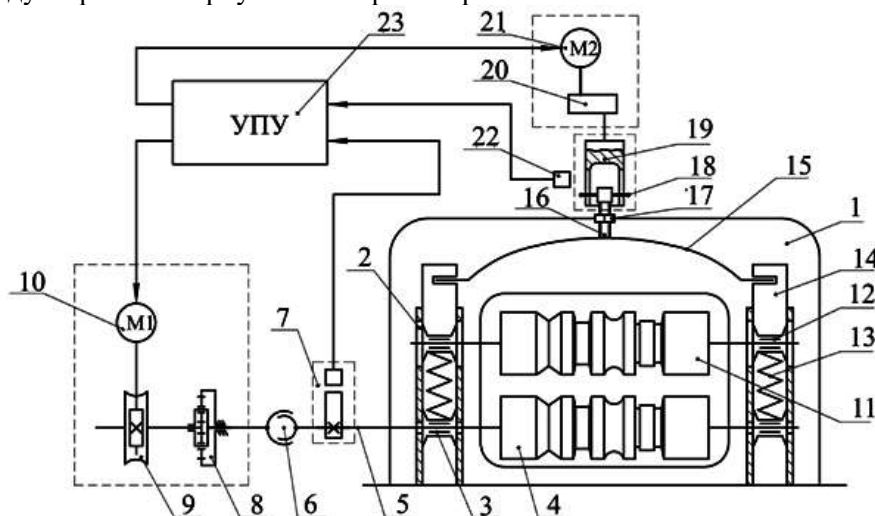


Рис.1. Схема лабораторного двухвалкового прокатного стана

Нажимной валок 11 установлен в подшипниках скольжения 12, которые опираются на пружины уравнивающего механизма 13. На подшипниках установлены опоры 14, соединяемые коромыслом 15. На коромысло опирается ходовой винт 16, составляющий с гайкой 17, кинематическую пару скольжения. Пальцы 18 головки ходового винта, входят в пазы вилки 19, образуя шпоночное соединение. Вилка установлена на валу редуктора 20 интегрированного с корпусом привода электродвигателя 21 нажимного механизма. При вращении по часовой стрелке, винт перемещает коромысло вниз, уменьшая расстояние между прокатными валками. При вращении винта в обратную сторону, расстояние между валками увеличивается. Информация о положении нажимного вала считывается с датчика положения 22.

В результате анализа технических характеристик, серийно выпускаемых двухвалковых прокатных станов, было сформулировано техническое задание на проектирование конструкции прокатного стана. На первом этапе создания стана, по разработанному комплекту эскизов, в Лаборатории технологий машиностроения Филиала «Протвино», изготовлены детали, необходимые для сборки лабораторного двухвалкового прокатного стана.

Стан изготовлен и используется, с ручным управлением, при изучении дисциплины «Физические основы обработки металлов давлением». Завершающим этапом создания лабораторного двухвалкового прокатного стана станет оснащение его устройством программного управления. В его функции должен входить набор команд, в совокупности выражающих основные физические закономерности обработки металлов на прокатных станах. Вспомогательные операции при выполнении лабораторных работ должны выполняться вручную.

Библиографический список

1. Колесников А.Г., Яковлев Р.А., Мальцев А.А. Расчет и конструирование прокатных станов. Конспект лекций. Часть 1. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2011. – 150 с.
2. Лабораторный стан ДУО-360. <https://ipdme.ru/stan-duo-360-laboratornyj> (дата обращения: 12.03.2025).
3. Лабораторный реверсивный стан ДУО-250. <https://ipdme.ru/stan-duo-250> (дата обращения: 12.03.2025).
4. Лабораторный реверсивный стан ДУО-180 <https://ipdme.ru/stan-duo-180-laboratornyj> (дата обращения: 12.03.2025).
5. Радионова Л. В., Фаизов С. Р., Лисовский Р. А. Лабораторный прокатный стан ДУО-130. Машиностроение: сетевой электронный научный журнал. 2017. Том 5, №4. С. 46-50. <https://indust-engineering.ru/issues/2017/2017-4.pdf> (дата обращения: 16.03.2025).

УДК 37.09

Краснова В.А.

ПОВЕДЕНИЕ МОЩНОСТИ СТАНДАРТНЫХ КРИТЕРИЕВ МНОЖЕСТВЕННЫХ СРАВНЕНИЙ НА НЕНОРМАЛЬНЫХ ВЫБОРКАХ POWER BEHAVIOR OF STANDARD CRITERIA MULTIPLE COMPARISONS IN NON- NORMAL SAMPLES

*Филиал «Протвино» государственного университета «Дубна»
Секция «Информационные технологии»*

Авторы: Краснова Вера Антоновна, студентка 3 курса направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Масликов Александр Альбертович, к.ф.-м.н., доцент кафедры общеобразовательных дисциплин филиала «Протвино» государственного университета «Дубна».

Author: Krasnova Vera Antonovna, 3rd year student of the program "Automation of technological processes and production" of the branch "Protvino" state University "Dubna".

Scientific adviser: Maslikov Alexander Albertovich, candidate of physical and mathematical sciences, associate professor of the department of general educational disciplines of the branch "Protvino" state University "Dubna".

Аннотация

Методом Монте-Карло исследуется влияние негауссовости выборок сравнительно небольшого объёма (10 элементов) на мощность критериев множественных сравнений (дисперсионный анализ с пост-тестом Тьюки). В качестве негауссовых распределений использовались распределение Вейбулла и гамма-распределение. Получены зависимости мощностей указанных критериев от величины эффекта, которые представлены в виде графиков. Сделаны выводы о применимости и свойствах критериев.

Abstract

The Monte Carlo method investigates the effect of non-Gaussian samples of a relatively small volume (10 elements) on the power of criteria for multiple comparisons (variance analysis with the Tukey post-test). The Weibull distribution and the gamma distribution were used as non-Gaussian distributions. The dependences of the capacities of these criteria on the magnitude of the effect are obtained, which are presented in the form of graphs. Conclusions are drawn about the applicability and properties of the criteria.

Ключевые слова: дисперсионный анализ, множественные сравнения, апостериорный критерий Тьюки, мощность.

Keywords: analysis of variance, multiple comparisons, post-hoc Tukey criterion, statistical power.

Данная работа является продолжением исследования влияния отклонений элементов выборок от нормального распределения на эффективность критериев множественных сравнений применительно к выборкам небольшого объёма (до 10 элементов) [1, 2].

При проверке статистических гипотез существует риск совершить ошибки двух видов - ошибки I-го рода, когда ошибочно отклоняется верная гипотеза и ошибки II-го рода, когда ошибочно принимается ложная гипотеза. Вероятность ошибки I-го рода называется уровнем значимости α , стандартом для него является значение 0,05. Вероятность ошибки II-го рода обозначают β . Ошибки I и II рода являются конкурирующими, уменьшение вероятности одной влечёт увеличение вероятности другой. другой полезной характеристикой является мощность критерия ($1 - \beta$). Чем больше мощность критерия, тем надёжнее он обнаруживает различия между выборками. Обычно считают приемлемой мощность порядка 0,8. Разумеется мощность зависит не только от объёма выборок (чем больше элементов в выборках, тем критерий оказывается мощнее), но и от величины эффекта, поэтому полезно иметь таблицы (или графики) для разных значений эффектов. Наиболее хорошо изучены различные зависимости мощностей критериев для гауссовых распределений, поскольку если нет надёжных оснований считать распределение отличным от нормального, то делается предположение о нормальности распределений в силу центральной предельной теоремы.

В классической формулировке для множественных сравнений используется Дисперсионный Анализ с последующим применением пост-теста сравнения. Традиционно считается наиболее мощным пост-тест Тьюки. Однако, аналитически доказана корректность этих процедур, только при условии нормальности распределений из которых производятся выборки (и гомогенности их дисперсий). Если не установлен факт нормальности, то в качестве альтернативы используют непараметрические ранговые критерии, которые обладают значительно меньшей мощностью. Поэтому представляется актуальным исследовать

правомерность использования параметрических критериев множественных сравнений для выборок из негауссовых распределений. Итак, универсальность непараметрических ранговых критериев вступает в противоречие с желаемой высокой мощностью характерной для параметрических критериев, которые, однако, традиционно применяются при условии нормальности распределений. Это индуцирует задачу исследовать вопрос – насколько оправдан отказ от использования традиционных параметрических критериев для негауссовых выборок.

Объект исследования, в нашем случае, это процедура множественных сравнений. При этом в качестве предмета исследования выступает вопрос зависимости мощностей параметрических критериев множественных сравнений в конфигурации дисперсионный анализ с пост-тестом Тьюки от ненормальности распределений, из которых берутся выборки.

Задача исследования провести компьютерный эксперимент-симуляцию путём извлечения и анализа методом Монте-Карло выборок из искажённых ненормальных распределений [3]. Использовались выборки сравнительно небольшого объёма (10 элементов), что характерно для некоторых дорогостоящих исследований, например, в фармакологии. Цель исследования проверить гипотезу о том, что использование традиционных параметрических критериев не приводит к катастрофическим последствиям. Это утверждение подкрепляется сравнением графиков, построенных на основе вычислений.

В качестве инструментария мы использовали программы, написанные в пакете Wolfram Mathematica (WM) (см. Приложение). Идея заключается в том, чтобы, следуя методу Монте-Карло, генерировать случайные выборки из различных распределений и применять к ним исследуемые тесты и таким образом осуществить компьютерный эксперимент. В WM есть возможность подключить генератор псевдослучайных чисел “Mersenne Twister”, который и был использован в работе. Вихрь Мерсенна – это генератор сдвигового регистра обобщенной обратной связи с огромным периодом ($2^{19937} - 1$), гарантирующим высочайшую степень случайности [4, 5]. Пакет WM позволяет формировать компактные программы для генерации большого числа случайных выборок из широкого набора библиотечных распределений.

Был выбран следующий формат компьютерного эксперимента. Каждый раз использовались четыре выборки из искажённых нормальных распределений, а именно распределение Вейбулла и гамма-распределение. Оба распределения ограничены с одной стороны, что представляется естественным для многих приложений (например, в контексте априорного отсутствия отрицательных значений для случайной величины). Используемое распределение Вейбулла имеет функцию плотности распределения: $f = \frac{2}{9} X \exp(-(X/3)^2)$,

гамма распределение имеет функцию распределения: $f = \frac{1}{16} X \exp(-X/4)$. Четвёрки распределений выбирались с равным, но переменным сдвигом относительно друг-друга, позволяя изменять наблюдаемый эффект. Величина сдвига измерялась в единицах среднеквадратического отклонения σ изучаемого распределения. Сдвиг изменялся от $0,0 \sigma$ до $0,7 \sigma$ с шагом $0,1 \sigma$. Каждое значение мощности ANOVA и post-hoc теста Тьюки [6] вычислялось по 10^5 сгенерированным четвёркам выборок. При этом погрешность метода Монте-Карло была пропорциональна $\sim 1/\sqrt{100000} = 0,00316$. Коэффициент пропорциональности оценивали «экспериментально», путём генерации серий по 1000 четвёрок. Далее по этой серии получалась оценка СКО для 1000, потом просто учитывалось, что у нас каждый раз происходило 10^5 генераций. В итоге оказывалось, что 95 %-ая погрешность для мощности в формате нашего исследования не хуже 0,004. Такая же процедура взаимного смещения была проведена со стандартным нормальным распределением, чтобы иметь возможность сравнить графики мощностей критериев, применённых к искажённым распределениям и к нормальному распределению. Число элементов в выборках было взято $=10$. Распределение Вейбулла и гамма имеют положительную асимметрию. Поэтому аналогично были исследованы случаи, когда 2 из 4-х распределения были инвертированы, т.е. имели отрицательную асимметрию.

Для наглядности ниже на графиках представлены примеры использованных распределений со сдвигом, из которых генерировались изучаемые выборки. На каждом из 3-х графиков (рис. 1 – 3) кривые распределений изображены со сдвигом $0,4 \sigma$.

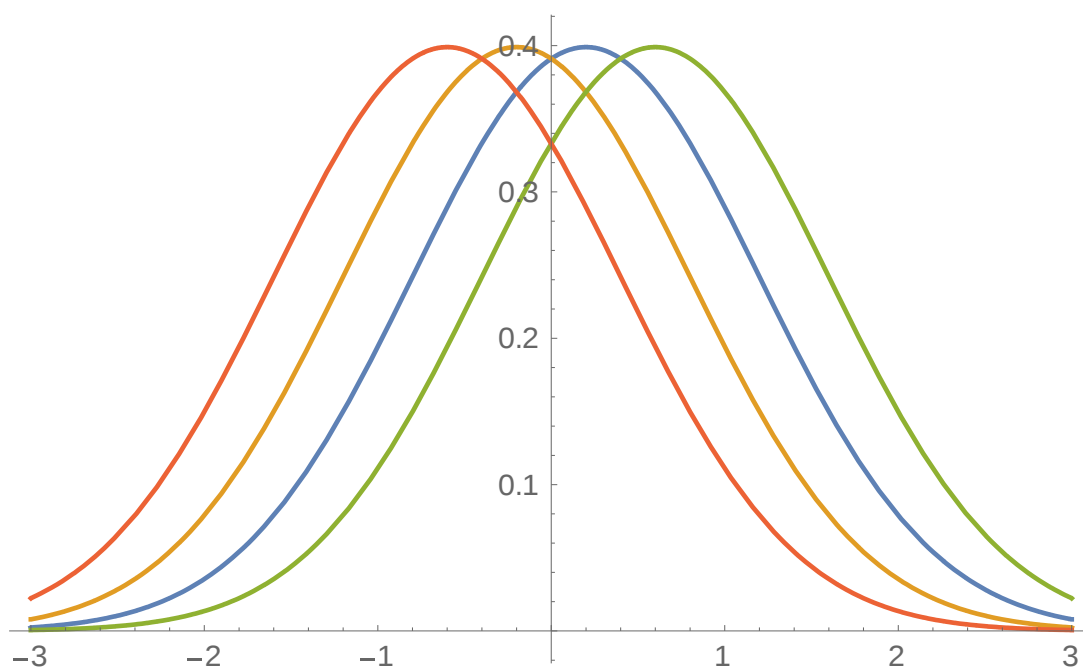


Рис. 1. Графики со сдвигом $0,4 \sigma$ для стандартного нормального распределения

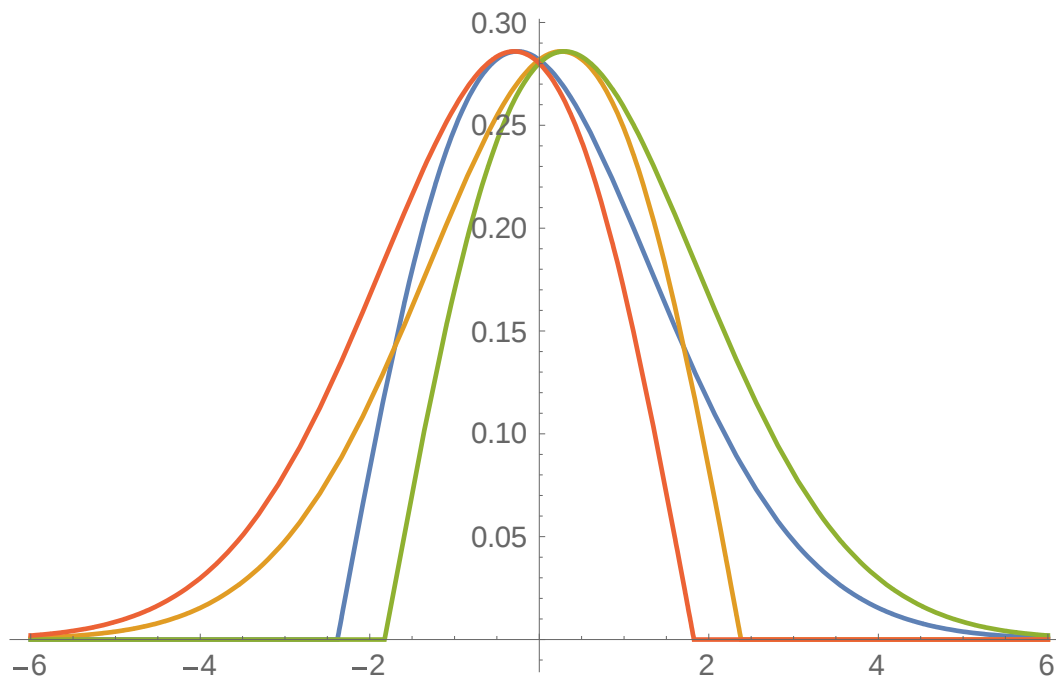


Рис. 2. Графики со сдвигом $0,4 \sigma$ для распределения Вейбулла с инверсией 2-х распределений

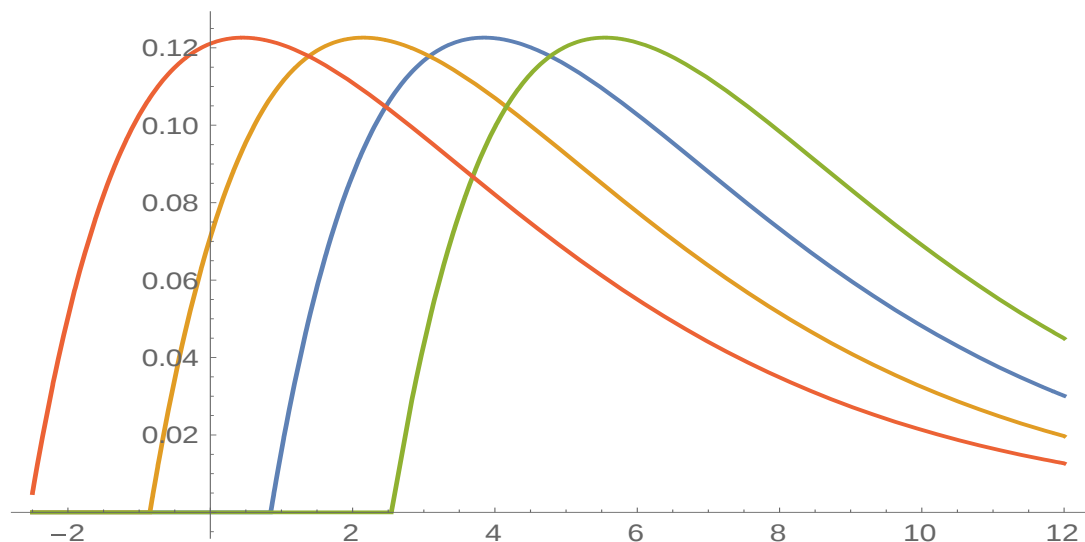


Рис. 3. Графики со сдвигом $0,4 \sigma$ для гамма-распределения

Для стандартного нормального распределения асимметрия и коэффициент островеершинности составляют соответственно $A = 0$, $E = 3$. Положительная-отрицательная асимметрия соответствует перекосу вправо-влево. Большой коэффициент островеершинности соответствует островеершинности распределения, а малый – обратной «деформации» распределения. Мы исследовали асимметричные распределения с повышенным показателем асимметрии и островеершинности. Для выбранных нами распределений Вейбулла и гамма (см.выше) эти коэффициенты составляют соответственно $A_1 = 0,63$; $E_1 = 3,25$; и $A_2 = 1,41$; $E_2 = 6$.

Значения мощности критерия Тьюки (Tukey HSD), полученные для наших множественных сравнений с изменяющимся сдвигом, нанесены на пять графиков (рис. 4) и соответствуют объёму выборок $=10$ элементов.

Таким образом по горизонтали отложен сдвиг крайних выборок от центральных в единицах СКО – σ , а по вертикали отложено значение мощности критерия Тьюки. Каждая точка есть результат применения теста Тьюки к 10^5 четвёрок сгенерированных выборок. При этом вычислялся результат дисперсионного анализа по тем же самым выборкам и сравнивался с результатом по Тьюки.

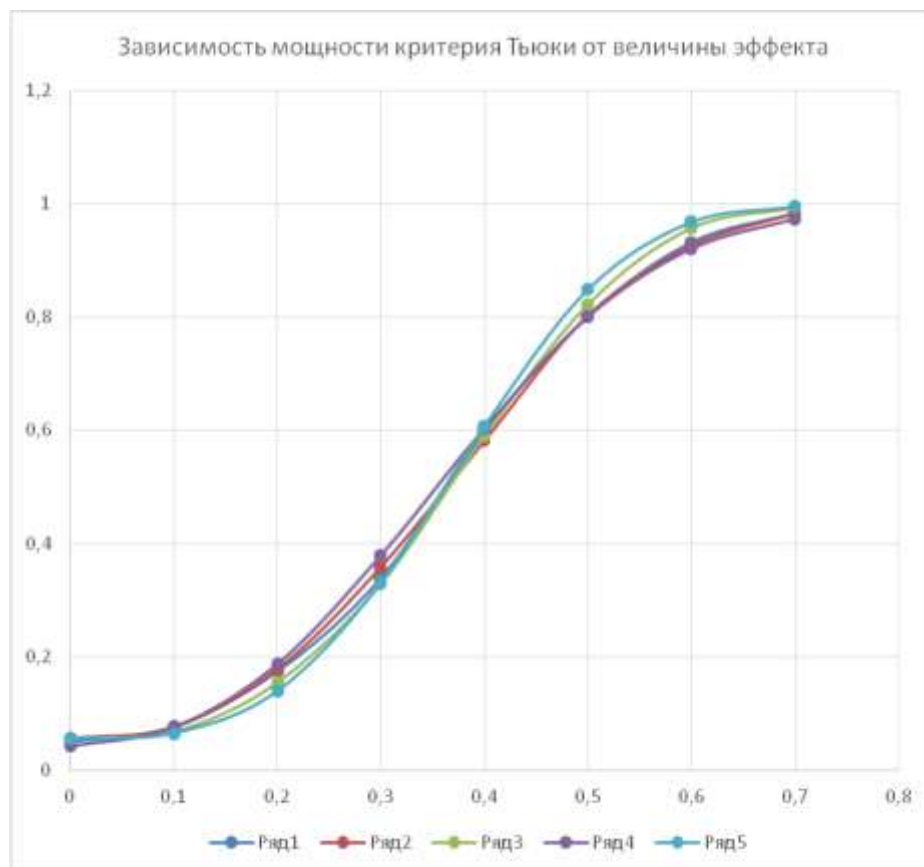


Рис. 4. Мощность критерия Тьюки (Tukey HSD (Honestly Significant Difference достоверные значимые различия)) для 10-ти элементных выборок

Соответствие цветов графиков распределения следующее: голубой – нормальное, оранжевый - Вейбулла, серый – Вейбулла с инверсией, жёлтый – гамма, синий – гамма с инверсией.

Таблица 1. Результаты вычислений мощностей критерия Тьюки

Эффект	Нормальное распределение	Распределение Вейбулла	Вейбулла с инверсией	Распределение Гамма	Гамма с инверсией
0	0,0504	0,0556	0,0558	0,0426	0,0564
0,1	0,0776	0,075	0,0672	0,0762	0,065
0,2	0,1748	0,1776	0,154	0,1876	0,1392
0,3	0,339	0,3578	0,3282	0,3794	0,3304
0,4	0,5972	0,581	0,5898	0,603	0,6074
0,5	0,8012	0,8022	0,822	0,7998	0,8488
0,6	0,9312	0,9256	0,9572	0,9198	0,9678
0,7	0,9824	0,983	0,993	0,9728	0,9952
	Ряд 1	Ряд 2	Ряд 3	Ряд 4	Ряд 5

По своей конструкции критерий Тьюки является пост-тестом для дисперсионного анализа, то есть применяется после проведения дисперсионного в случае, если последний обнаружил различия в каких-то выборках (каких именно он указать не в состоянии). Этот тест призван определить – какие именно выборки достоверно различаются и считается одним из наиболее мощных апостериорных тестов. Следуя описанной процедуре, мы всегда перед пост-тестом Тьюки выполняли дисперсионный анализ и оценивали его мощность в зависимости от эффекта. Результаты по мощности дисперсионного анализа представлены в Таблице 2.

Следует отметить, что распространённое мнение о том, что post-hoc тест Тьюки всегда имеет меньшую мощность, чем дисперсионный анализ (ДА) в данном случае не подтвердилось. Хотя обычно результат теста Тьюки согласуется с результатом ДА, оказалось, что критерий Тьюки иногда обнаруживал различия между выборками там, где ДА их не замечал.

Таблица 2. Результаты вычислений мощностей Дисперсионного анализа

Эффект	Нормальное распределение	Распределение Вейбулла	Вейбулла с инверсией	Распределение Гамма	Гамма с инверсией
0	0,0502	0,0558	0,0588	0,0448	0,0578
0,1	0,0816	0,0762	0,0666	0,078	0,0636
0,2	0,1784	0,1818	0,1582	0,1946	0,135
0,3	0,3536	0,367	0,3388	0,3884	0,3334
0,4	0,6052	0,597	0,6058	0,6222	0,6303
0,5	0,815	0,8164	0,8408	0,8146	0,867
0,6	0,9352	0,9328	0,9672	0,924	0,9768
0,7	0,9864	0,9854	0,9956	0,9766	0,9978

В целом следует отметить, что все графики мощности монотонно стремятся к единице с увеличением эффекта (сдвига) и с увеличением объёма выборок. При этом они воспроизводят форму кривой мощности для множественного сравнения выборок из нормальных распределений. Естественно, наблюдается отклонение от мощностей для нормальных распределений, в особенности для гамма-распределения с разносторонней асимметрией. Это связано с большим отклонением коэффициентов (моментов) от нормальных. Различие в графиках преимущественно наблюдается для распределений с инверсией, т.е. когда сравнивались выборки с разнонаправленной асимметрией. Причём при больших эффектах это отклонение направлено в сторону увеличения мощности, что говорит о том, что для выборок (распределений) с разнонаправленной асимметрией критерий Тьюки более эффективно (чем для нормальных выборок) обнаруживает большие различия.

В итоге мы склонны утверждать, что наша гипотеза для множественных сравнений вполне оправдана и использование классических параметрических критериев целесообразно даже при отсутствии нормальности распределений, что позволяет корректно производить множественные сравнения и делать адекватные выводы о наличии или отсутствии различий в центральных тенденциях выборок.

Библиографический список

1. Алдобаев В.Н., Артемьева А.Д., Масликов А.А. Исследование поведения классических критериев множественных сравнений, на ненормальных неоднородных распределениях,

- методом Монте-Карло // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. - 2021. - №3. - С. 72-80.
2. Алдобаев В.Н., Масликов А.А., Скворцова М.С. Исследование мощности критериев множественных сравнений, применённых к выборкам с негауссовым распределением элементов, Сборник «Государственный университет «Дубна». 30 лет в науке», 2024 г.
 3. Лемешко Б.Ю., Лемешко С.Б., Постовалов С.Н., Чимитова Е.В. Статистический анализ данных, моделирование и исследование вероятностных закономерностей. Компьютерный подход. - Изд-во НГТУ, 2011. - 888 с.
 4. Matsumoto, M. 623-Dimensionally Equidistributed Uniform Pseudorandom Number Generator. ACM Transactions on Modeling and Computer Simulation 8 / M. Matsumoto, T. Nishimura. – 1998. – No 1. – P. 3–30.
 5. Nishimura, T. Tables of 64-Bit Mersenne Twisters. ACM Transactions on Modeling and Computer Simulation 10 / T. Nishimura. – 2000. – No 4. – P. 348–357. DOI:10.1145/369534.369540.
 6. Тьюки, Джон (1949). "Сравнение индивидуальных средних в Дисперсионном анализе". Биометрия. 5 (2): 99–114.

УДК 004.8

Липовцев И.А., Кириченко Ж.Д.

**СРАВНИТЕЛЬНЫЙ АНАЛИЗ ОТЕЧЕСТВЕННЫХ И ЗАРУБЕЖНЫХ LLM-МОДЕЛЕЙ:
ВОЗМОЖНОСТИ И ПРИМЕНЕНИЕ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ**
COMPARATIVE ANALYSIS OF DOMESTIC AND FOREIGN LLMs: CAPABILITIES
AND APPLICATIONS IN AUTOMATION SYSTEMS

*Филиал «Протвино» государственного университета «Дубна»
Секция «Машиностроение»*

Авторы: Липовцев Илья Алексеевич, студент 4 курса направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна»; Кириченко Жанна Дмитриевна студент 4 курса направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Маков Пётр Владимирович, кандидат технических наук, доцент кафедры автоматизации технологических процессов и производств филиала «Протвино» государственного университета «Дубна».

Authors: Lipovtsev Ilya Alekseevich, 4th-year student of the program "Automation of Technological Processes and Production" of the branch "Protvino" state University "Dubna"; Kirichenko Zhanna Dmitrievna, 4th-year student of the program "Automation of Technological Processes and Production" of the branch "Protvino" state University "Dubna".

Scientific adviser: Makov Petr Vladimirovich, candidate of technical sciences, associate professor of the department of automation of technological processes and production of the branch "Protvino" state University "Dubna".

Аннотация

В данной работе проводится сравнительный анализ отечественных и зарубежных больших языковых моделей (LLM) с целью выявления их возможностей и перспектив применения в бизнесе и на производстве. Оценка моделей проводится на основе нескольких ключевых критериев, таких как скорость работы, точность генерации текста, решение логических задач и следование пользовательским инструкциям. В работе рассматриваются сильные и слабые стороны LLM.

Abstract

This paper presents a comparative analysis of domestic and international large language models (LLM) to evaluate their capabilities and potential for use in business and production. The models are

assessed based on several key criteria, including processing speed, text generation accuracy, logical problem-solving, and adherence to user instructions. The study explores the strengths and weaknesses of LLMs.

Ключевые слова: большие языковые модели, искусственный интеллект, нейросети, бизнес, производство, машинное обучение, обработка естественного языка.

Keywords: large language models, artificial intelligence, neural networks, business, production, machine learning, natural language processing.

Современные достижения в области искусственного интеллекта и машинного обучения приводят к бурному росту технологий, которые расширяют горизонты автоматизации, обработки информации и повышения эффективности процессов.

В последние годы значительное внимание научного сообщества сосредоточено на сравнительном анализе больших языковых моделей (LLM), разработанных как в России, так и за рубежом, с целью оценки их потенциала и выявления ограничений [1,3].

Актуальность: большие языковые модели (LLM) находят всё более широкое применение в таких сферах, как промышленность, наука и бизнес. При этом активное использование зарубежных LLM сопровождается рисками, связанными с безопасностью конфиденциальных данных. В связи с этим разработка и внедрение отечественных решений становится приоритетной задачей для укрепления цифрового суверенитета и обеспечения технологической самостоятельности.

Цель: Проведение сравнительного анализа отечественных и зарубежных LLM для выявления их сильных и слабых сторон, а также определения потенциала отечественных моделей в задачах автоматизации.

Задачи:

1. Изучить теоретическую базу и текущие тенденции развития LLM.
2. Оценить производительность отечественных и зарубежных моделей по ключевым параметрам: точность, скорость работы, универсальность.
3. Исследовать влияние промптов на результаты выполнения задач моделями.
4. Выявить области, в которых использование отечественных LLM наиболее эффективно.
5. Предложить рекомендации по применению отечественных LLM в условиях повышенных требований к информационной безопасности.

Объект исследования: большие языковые модели, применяемые для автоматизации обработки естественного языка.

Предмет исследования: эффективность применения отечественных (GigaChat, YandexGPT 3 Pro) и зарубежных (ChatGPT-4, Large-2) LLM в задачах автоматизации.

Методы исследования: Для достижения цели применяются методы аналогии и сравнения, а также эмпирическое тестирование. В рамках тестирования анализируется способность моделей решать задачи из различных областей (логика, математика, перевод) с и без использования промптов.

Проблема: Сравнительный анализ моделей позволит определить, насколько отечественные аналоги могут заменить зарубежные решения, а также выявить их преимущества и ограничения в контексте реальных задач. Использование зарубежных LLM может угрожать безопасности обрабатываемых данных, что особенно важно для критически значимых отраслей. При этом отечественные модели часто подвергаются критике за ограниченную точность и функциональность.

Гипотеза: Отечественные LLM, несмотря на текущее отставание, обладают потенциалом для конкурентоспособного использования в задачах автоматизации при условии их правильной настройки и применения в условиях, где критически важна информационная безопасность. GigaChat и YandexGPT 3 Pro ориентированы на русскоязычный рынок и учитывают национальные особенности языка, что делает их эффективными для русскоязычных текстов. ChatGPT-4 и Large-2 представляют международные решения, адаптированные под различные языки, что делает их универсальными для глобального применения.

Сначала оценивались возможности моделей в прикладных науках без промптов, чтобы определить их базовый уровень понимания и способности решать задачи. Использовалось по 20 задач из различных областей: физические – с easyfizika.ru, математические – с math-ege.sdamgia.ru, логические задачи – с chronolines.ru.

Промпт*(prompt) – это текстовая инструкция, направляющая ответ модели. Использование промптов позволяет пользователю точно формулировать задачу, повышая качество и релевантность ответов. В данном анализе промпты направили модели на решение конкретных вопросов, что позволило оценить их способность понимать контекст и адаптироваться к требованиям пользователя.

Каждая модель предназначена для выполнения определенного спектра задач. Основная цель – оценить их базовые когнитивные способности и универсальность без специальных указаний.

Анализ результатов без промптов.

1. Физика: Large-2 показал наилучшие результаты (75%), тогда как YandexGPT 3 Pro — лишь 25%. Gigachat-55%, а ChatGPT-4o mini-65%. Зарубежные модели, вероятно, обладают более обширными данными по физике.
2. Математика: ChatGPT-4 достиг 100% точности, GigaChat и Large-2 — 90%, YandexGPT 3 Pro — 50%. ChatGPT-4o mini имеет значительное преимущество в математических задачах.
3. Логика: Все модели показали сопоставимые результаты: ChatGPT-4o mini и Large-2 — 80%, GigaChat и YandexGPT 3 Pro — 70%. Зарубежные модели немного опережают отечественные.
4. Размер контекста: ChatGPT-4o mini превосходит остальные модели (128K токенов). Отечественные модели ограничены: YandexGPT 3 Pro — 8K токенов, GigaChat и Large-2 — 32K токенов.
5. Скорость ответа: Отечественные модели быстрее (9 секунд), ChatGPT-4o mini — до 15 секунд, Large-2 — 11 секунд. Быстрая скорость отечественных моделей делает их предпочтительными для оперативных приложений.

Таким образом, без промптов зарубежные модели показывают лучшие результаты в физике и математике, что делает их подходящими для систем автоматизации и вычислительных задач. Отечественные модели демонстрируют более высокую скорость ответа, что может быть преимуществом в задачах, требующих оперативности.

Для данных задач был составлен промпт: "Представь, что ты [специалист в этой области]. Тебе нужно решить задачу по [указываем раздел задачи]. Попробуй решить задачу 2-я разными способами, проанализируй результат и сравни подходы. После этого напиши ответ."

Анализ результатов с использованием промптов:

1. Физика: Все модели улучшили свои показатели. ChatGPT-4o mini достиг 95%, Large-2 — 80%, YandexGPT 3 Pro — 45%, Gigachat — 60%. Зарубежные модели лучше реагируют на направляющие инструкции.
2. Математика: ChatGPT-4o mini и Large-2 показали максимальные результаты (100%), GigaChat сохранил 90%, YandexGPT 3 Pro улучшился до 65%. Промпты важны для точности выполнения математических операций.
3. Логика: ChatGPT-4o mini показал лучший результат (85%), остальные модели улучшили свои результаты до 70–80%. Промпты способствуют лучшему выполнению логических задач.
4. Размер контекста и скорость ответа: Показатели остались неизменными. Модели с меньшим временем отклика (YandexGPT 3 Pro и GigaChat) сохраняют преимущество по скорости, что делает их полезными в задачах, где важна оперативность.

Выводы из анализа с использованием промптов:

- Все модели демонстрируют улучшение производительности при использовании специализированных инструкций.

- Использование промптов повышает точность выполнения задач и позволяет более полно раскрыть потенциал моделей для решения сложных и узкоспециализированных задач.
- Корректная и структурированная постановка запроса является ключевым фактором для достижения высоких показателей производительности.

Также важным аспектом применения языковых моделей в автоматизации является их способность к точному переводу технической документации. Перевод специализированных текстов требует знания терминологии и способности передавать смысл в строгом соответствии с исходными параметрами и инструкциями. Оценка перевода моделей позволяет выявить их применимость для задач, связанных с автоматизацией и обработкой технической информации, а также их ограничения в интерпретации специализированных понятий и параметров.

Для тестирования качества перевода был выбран отрывок с профессионального веб-ресурса ([Linguacontact](#)).

Оригинальный текст: Setting the pressure probe range. When using pressure probes with output signal in mA, the PbC parameter must be configured as Cur, while P04 and P20 (PbF, F04, and F20 in fans) will be set with their corresponding value. P04: Setting the output signal of the transducer corresponding to 4 mAP20: Setting the output signal of the transducer corresponding to 20 mA. The pressure value to configure must be relative pressure when the rLP parameter = rEL, and absolute pressure when the rLP parameter = AbS. Note: absolute pressure = relative pressure + 1 bar (14.5 PSI)

Профессиональный перевод: Установка диапазона датчиков давления. При использовании датчиков давления с токовым выходным сигналом в мА, для параметра PbC необходимо установить значение Cur, а для P04 и P20 (PbF, F04 и F20 для вентиляторов) должны быть установлены их соответствующие значения. P04: Установка выходного сигнала преобразователя соответствующим току 4 мА P20: Установка выходного сигнала преобразователя соответствующим току 20 мА. Значение давления при установке конфигурации должно быть относительным, когда параметр rLP = rEL, и абсолютным, когда параметр rLP = AbS. Примечание: абсолютное давление = относительному давлению + 1 бар (14,5 фунтов/дюйм²)

Для выполнения перевода с английского языка на русский, каждой из моделей был задан следующий запрос: *"Выполни перевод следующего технического фрагмента с английского языка, максимально точно передав его смысл и терминологические особенности. Удели внимание построению предложений и стилю, типичному для технической документации. При наличии узкоспециализированных терминов, не имеющих очевидного русского аналога, укажи оригинал либо предложи несколько вариантов перевода в скобках с пояснениями. Основная задача – точная передача содержания."*

На основании предоставленных переводов технического текста, вот текстовая версия ответов от каждой модели:

Ответ от ChatGPT-4o mini:

Установка диапазона датчика давления. При использовании датчиков давления с выходным сигналом в мА параметр PbC должен быть настроен как Cur, в то время как P04 и P20 (PbF, F04 и F20 для вентиляторов) будут установлены с соответствующими значениями. P04: Установка выходного сигнала преобразователя, соответствующего 4 мА. P20: Установка выходного сигнала преобразователя, соответствующего 20 мА. Давление, необходимое для настройки, должно быть относительным, когда параметр rLP = rEL, и абсолютным, когда параметр rLP = AbS. Примечание: абсолютное давление = относительное давление + 1 бар (14,5 PSI).

Ответ от Large-2:

Установка диапазона датчика давления. При использовании датчиков давления с выходным сигналом в мА параметр PbC должен быть настроен как Cur, а параметры P04 и P20 (PbF, F04 и F20 для вентиляторов) должны быть установлены на соответствующие значения. P04: Установка выходного сигнала преобразователя, соответствующего 4 мА. P20: Установка выходного сигнала преобразователя, соответствующего 20 мА. Значение давления, которое необходимо настроить, должно быть относительным давлением, когда параметр rLP = rEL, и

абсолютным давлением, когда параметр $rLP = AbS$. Примечание: абсолютное давление = относительное давление + 1 бар (14,5 PSI).

Ответ от GigaChat:

Настройка диапазона датчика давления. При использовании датчиков давления с выходным сигналом в мА параметр PbC необходимо настроить как «Cur», а параметры P04 и P20 (PbF, F04 и F20 у вентиляторов) будут настроены на соответствующие значения. P04: Настройка выходного сигнала преобразователя, соответствующего 4 мА. P20: Настройка выходного сигнала преобразователя, соответствующего 20 мА. Значение давления, которое нужно настроить, должно быть относительным давлением при значении параметра $rLP = rEL$ и абсолютным давлением при $rLP = AbS$. Примечание: абсолютное давление = относительное давление + 1 бар (14,5 psi).

Ответ от YandexGPT 3 Pro:

Настройка диапазона датчика давления При использовании датчиков давления с выходным сигналом в миллиамперах параметр PbC необходимо настроить как Cur, а P04 и P20 (в вентиляторах используются обозначения PbF, F04 и F20) нужно задать соответствующие значения. P04: установка выходного сигнала преобразователя, соответствующего 4 мА. P20: установка выходного сигнала преобразователя, соответствующего 20 мА. Необходимо настроить значение давления относительно параметра $rLP = rEL$ или абсолютного давления, если $rLP = AbS$. Примечание: абсолютное давление = относительное давление + 1 бар (14,5 фунта на квадратный дюйм).

Анализ результатов перевода

- ChatGPT-4o mini: Показал наилучший результат, обеспечив точный перевод терминов и параметров. Модель продемонстрировала высокую способность к интерпретации контекста и корректной передаче инструкций, что делает её подходящей для задач, где требуется высокая степень точности и точное соблюдение инструкций.
- YandexGPT 3 Pro: Выполнил перевод с базовым соответствием требованиям, но с некоторыми упущениями в передаче специфических терминов и технических инструкций. Модель может быть полезна для общего перевода, однако требует доработки для применения в высокоточных областях.
- GigaChat: Показал сопоставимые с YandexGPT 3 Pro результаты, справляясь с базовыми аспектами перевода. Однако качество передачи технической информации, особенно специфических терминов, уступает зарубежным аналогам, что может ограничивать его использование в критически важных приложениях.
- Large-2: Демонстрирует хорошее понимание текста, однако испытывает сложности с точной передачей инструкций и терминологии. Модель может быть полезна для менее специализированных текстов, но её эффективность снижается при работе с узкоспециализированной технической информацией.

Проведенный анализ показал, что ChatGPT-4o mini демонстрирует наилучшую точность перевода технической документации, что делает его эффективным инструментом для автоматизации задач, требующих высокой точности. YandexGPT 3 Pro и GigaChat обеспечивают приемлемое качество перевода для общего использования, но могут нуждаться в доработке при работе с узкоспециализированными текстами. Large-2 показал базовое понимание текста, но его применение целесообразно в задачах, не требующих детализированного соответствия техническим инструкциям.

Вывод: проведенное исследование показало, что отечественные большие языковые модели в среднем уступают зарубежным аналогам в точности выполнения задач, таких как решение логических, математических и прикладных задач. Однако, несмотря на это отставание, отечественные LLM остаются мощными инструментами, которые способны эффективно решать широкий круг задач в сфере автоматизации. Использование отечественных моделей, таких как GigaChat и YandexGPT 3 Pro, может стать надежной основой для автоматизации бизнес-процессов и производственных операций, особенно в тех условиях, где критически важна информационная безопасность. Эти модели обладают всеми ключевыми функциональными возможностями, характерными для современных LLM, и при правильной

интеграции способны повысить эффективность работы предприятий, снизить затраты на ручные операции и ускорить обработку данных.

В динамично развивающейся сфере LLM и генеративных моделей прорывы происходят каждые несколько месяцев, и хотя на текущий момент отечественные модели несколько отстают по ряду показателей, этот разрыв имеет тенденцию к сокращению. Интенсивное развитие технологий и улучшение алгоритмов могут в ближайшие годы свести разницу между отечественными и зарубежными моделями к минимуму.

Библиографический список

1. Жусип М. Н., Жаксыбаев Д. О. Сравнение чат-ботов с использованием трансформеров и нейросетей: исследование применения архитектур GPT и BERT // Вестник науки. 2024. №9(78). URL: <https://cyberleninka.ru/article/n/sravnenie-chat-botov-s-ispolzovaniem-transformerov-i-neyrosetey-issledovanie-primeneniya-arhitektur-gpt-i-bert> (дата обращения: 29.10.2024).
2. Соколова М. Е. ChatGPT и промпт-инжиниринг: о перспективах внедрения генеративных нейросетей в науку // Научно-технические исследования. 2024. №1. С. 94–106. URL: <https://cyberleninka.ru/article/n/chatgpt-i-prompt-inzhiniring-o-perspektivah-vnedreniya-generativnyh-neyrosetey-v-nauke> (дата обращения: 28.10.2024).
3. AI is here – and everywhere: 3 AI researchers look to the challenges ahead in 2024 // The Conversation. 2024. URL: <https://theconversation.com/ai-is-here-and-everywhere-3-ai-researchers-look-to-the-challenges-ahead-in-2024-218218> (дата обращения: 28.10.2024).

УДК 004.432

Макаров А.О.

СРАВНЕНИЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ C++ И RUST С ТОЧКИ ЗРЕНИЯ РАЗРАБОТЧИКА COMPARISON OF C++ AND RUST PROGRAMMING LANGUAGES FROM A DEVELOPER'S PERSPECTIVE

Филиал «Протвино» государственного университета «Дубна»
Секция «Информационные технологии»

Автор: Макаров Артем Олегович, инженер-программист, ООО «Систел».

Научный руководитель: Кульман Татьяна Николаевна, кандидат технических наук, доцент кафедры информационных технологий филиала «Протвино» государственного университета «Дубна».

Author: Makarov Artyom Olegovich, software engineer, LLC «Systel».

Scientific adviser: Kulman Tatiana Nikolaevna, Ph.D., associate professor of the department information technology of the branch "Protvino" state University "Dubna".

Аннотация

Проводится сравнение языков программирования C++ и Rust по нескольким характеристикам, важных с точки зрения разработчика.

Abstract

A comparison of C++ and Rust programming languages is conducted based upon several characteristics important to a developer.

Ключевые слова: безопасность, C++, Rust, система типов, эргономика, языки программирования.

Keywords: safety, C++, Rust, type system, ergonomics, programming languages.

В последнее время в сфере разработки программного обеспечения возрос интерес к языкам программирования с безопасным доступом к памяти. Так, например, Агентство

национальной безопасности (АНБ) США рекомендует отказаться от использования языков C и C++, поскольку они не являются безопасными с этой точки зрения [1]. В докладе АНБ утверждается, что 70% уязвимостей связаны с небезопасным доступом к памяти.

В качестве альтернативы C и C++ предлагаются такие языки, как Python, Java, C#, Go, Delphi/Object Pascal, Swift, Ruby, Rust и Ada. Большинство этих языков, однако, сильно ограничивают ручную работу с памятью, полагаясь на автоматическое управление её с помощью сборщика мусора (garbage collector). Это обстоятельство не позволяет использовать их для разработки некоторых типов приложений.

Среди предложенных альтернатив возможность ручного управления памятью предоставляет Rust, что ставит его в один ряд с другими системными языками программирования, включающие C и C++. Более строгая модель доступа к памяти в Rust'e гарантирует безопасную работу с ней, но дополнительно накладывает определённые ограничения на манипуляции с данными внутри программы.

Сообщество разработчиков C++ отреагировало на заявления АНБ, выступив с несколькими предложениями по внедрению в язык механизмов безопасного обращения с памятью. Среди них:

1. Библиотека и плагин для компилятора Clang [2];
2. Профили безопасности [3];
3. Документ Safe C++ [4];

Это говорит о том, что C++ остаётся актуальным и востребованным инструментом для разработки программного обеспечения.

Цель этой работы – выяснить, в каких случаях целесообразнее применять C++ или Rust в зависимости от требований к разрабатываемой программе.

Для достижения цели требуется выполнить следующие **задачи**:

1. Кратко рассмотреть историю развития языков C++ и Rust;
2. Сравнить эти языки по следующим характеристикам:
 - 2.1 Система типов;
 - 2.2 Обработка ошибок;
 - 2.3 Модель доступа к памяти;
 - 2.4 Удобство использования.
3. На основе сравнения сделать вывод о том, для разработки каких программ лучше подойдёт или Rust, или C++.

Краткая история

C++ появился в 1985, как *объектно-ориентированное* расширение к языку Си. Со временем в C++ была добавлена поддержка функциональной и обобщённой парадигм программирования. В 1998 язык был официально стандартизирован [5], включая стандартную библиотеку шаблонов, ныне являющаяся стандартной библиотекой языка. Начиная со стандарта C++11, вышедшего соответственно в 2011-м году, новые стандарты выходят каждые три года: последним опубликованным стандартом является C++23, активно ведутся работы над C++26.

Работа над языком Rust началась в 2006-м году в качестве личного проекта одного из разработчиков компании Mozilla – Хора Грэйдона. Версия 0.1 была опубликована в 2012-м году, что привлекло внимание других разработчиков, ставших вносить свой вклад в развитие языка. Многие из них имели опыт работы как с C++, так и с функциональными языками программирования, что в значительной мере повлияло на дизайн Rust'a.

В течение следующих трёх лет язык претерпел большие изменения. Наконец в 2015-м году была опубликована первая стабильная версия языка. Текущей его версией является 1.86.0. Начиная с 2018 года выпускаются так называемые редакции языка, имеющие несовместимые изменения по сравнению с предыдущей редакцией.

Как C++, так и Rust являются компилируемыми языками, имеют статическую систему типов и поддерживают основные парадигмы: функциональную, процедурную, обобщённую, структурную и объектно-ориентированную, хотя и с некоторыми оговорками в случае Rust'a [6].

Системы типов C++ и Rust

Оба языка имеют статическую систему типов и поддерживают вывод типов, иными словами компилятор способен самостоятельно вывести тип выражения. Это избавляет от необходимости указывать тип переменной при её объявлении и одновременной инициализации.

С точки зрения безопасности типов Rust является более строгим языком по сравнению с C++, поскольку запрещает любые *неявные* преобразования даже между фундаментальными типами. В C++ между базовыми типами разрешены неявные преобразования, которые достались языку в наследство от Си.

Rust позволяет проводить только *явные* преобразования с помощью: 1) оператора `as`; 2) типажей `Into<T>` или `From<T>`.

C++ предоставляет множество путей для неявного преобразования типов: 1) конструкторы классов, объявленных с одним аргументом; 2) оператор преобразования для пользовательского типа данных. В обоих случаях компилятор может прибегнуть к неявному преобразованию между двумя типами, если конструктор или оператор преобразования объявлены без ключевого слова `explicit` (от англ. – явный) [7]. Это может быть удобным в одних ситуациях и источником ошибок в других.

Таким образом, система типов в C++ является менее строгой по сравнению с таковой в Rust'е и, как следствие, более гибкой в обмен на более слабые гарантии в отношении типобезопасности.

Как уже было упомянуто выше, Rust и C++ поддерживают объектно-ориентированную парадигму. Оба языка позволяют инкапсулировать данные в пользовательском типе, ограничив к ним доступ и предоставив открытый интерфейс в виде функций. Перегрузка функций есть в C++, но отсутствует в Rust – полиморфизм, то есть возможность динамического изменения поведения программы, реализуется в нём с помощью типажей и алгебраических типов данных.

Типаж в Rust'е представляет собой набор функций, возможно, с реализацией по умолчанию, и схож с понятием интерфейса из объектно-ориентированных языков программирования. Реализация типажа типом означает реализацию всех функций, указанных в типаже.

На основе типажей в Rust'е осуществляется статический и динамический полиморфизм. В первом случае типаж ограничивает набор типов, принимаемых обобщённой функцией или пользовательским типом данных. Во втором – позволяет во время выполнения программы заместить один тип другим при условии, что оба они реализуют требуемый типаж. Типаж здесь играет роль базового типа, или класса, из объектно-ориентированных языков.

Типаж в Rust'е объединяют два разных подхода, применяемых в C++, – виртуальные функции в совокупности с их перегрузкой и наследованием и концепты. На механизме виртуальных функций построен динамический полиморфизм в этом языке. В отношении статического полиморфизма концепты выполняют ту же роль, что и типаж, однако явной реализации концептов в C++ не требуется – концепты лишь определяют требуемый набор функций и не более.

Стоит также отметить, что полиморфизма в обоих языках можно добиться с помощью алгебраических типов данных (АТД), известных как типы-суммы, или помеченные объединения (`tagged union`). Такие типы состояются из других типов. В один момент времени значением АТД может быть только один из типов, которые составляют этот АТД. В зависимости от того, какой тип в текущий момент содержится в АТД, программа может менять своё поведение, за счёт чего и достигается полиморфизм, хотя и следует учитывать, что добавить новых тип в уже существующий АТД без модификации его определения нельзя.

АТД определяется в Rust'е с помощью ключевого слова `enum`, в C++ – через использование `std::variant` из стандартной библиотеки языка. Преимуществом Rust'a в механизме АТД является встроенная в язык поддержка сопоставления с образцом (`pattern matching`), хотя это и не ограничивается исключительно алгебраическими типами данных. Конструкция `match` в Rust'е, которая и реализует этот механизм, является улучшенной версией оператора выбора `switch` из C++ и предоставляет такие возможности, как декомпозиция структур, сопоставление с вариантами АТД, объединение условий в одно и шаблоны поиска (`wildcards`). Схожий функционал был предложен для добавления в C++ в виде конструкции

inspect [8], однако на настоящий момент это предложение остаётся лишь идеей. Эмуляция сопоставления с образцом достижима в C++ лишь с использованием сторонних библиотек [9].

Подводя итог, можно сказать, что оба языка предоставляют схожие инструменты для работы с типами данных.

Обработка ошибок

Кратко коснёмся подходов к обработке ошибок, поскольку это является важной частью любой программы.

Для обработки ошибок в C++ обычно используется механизм исключений – при обнаружении ошибок функция возбуждает (throw) исключение, после чего программа разворачивает стек (stack unwinding) в поисках блока catch. При этом вызов этой функции должен находиться внутри блока try. Если блок catch так и не был найден, то по умолчанию программа аварийно завершается посредством вызова std::terminate.

Такой подход является простым и удобным, поскольку не требует явного обозначения, какие типы исключений функция может возбуждать. Также не требуется обработка исключений – вызывающий код может проигнорировать исключение, чтобы оно распространилось на уровень выше по стеку.

Однако эти два обстоятельства являются и недостатками этого подхода. Исключения можно проигнорировать слишком легко. При этом информация о том, какие именно исключения может сгенерировать та или иная функция, доступна лишь документации или в исходном коде функции, но не в её сигнатуре.

Подход к обработке ошибок в Rust'e является более явным. Функции могут или возбудить панику (panic), что приведёт к аварийному завершению программы, или вернуть специальный тип – Result<T, E>, где T – тип выражения, возвращаемого при безошибочном выполнении функции, а E – тип ошибки. Типы T и E могут быть абсолютно любыми. Тип Result<T, E> отмечен атрибутом must_use – компилятор выведет предупреждение, если значение этого типа не будет использовано.

Это ведёт к тому, что проигнорировать возникшую ошибку труднее, чем в C++, а информация о том, какого рода ошибку может вернуть функция, является более доступной для программиста, поскольку выражается в сигнатуре этой функции. С другой стороны, возбуждает ли функция панику или нет, можно узнать только в документации или её исходном коде.

Модель доступа к памяти

C++ в силу необходимости обратной совместимости с Си перенял от него и модель доступа к памяти, которую можно описать как свободную: программа имеет право обратиться по любому адресу в любой момент времени. Это может привести как к неопределённому поведению или критическому сбою по следующим причинам:

1. По запрошенному адресу находится невалидная структура данных;
2. Запрошенный адрес не принадлежит программе;
3. Несколько потоков выполнения пытаются одновременно модифицировать одну и ту же структуру данных в памяти.

Такая модель работы с памятью, хотя и является гибкой, давая программисту большую свободу в управлении ресурсами памяти, требует от него в то же время повышенного внимания к потоку данных в программе – времени жизни переменных, владении одних переменных другими, валидности указателей и ссылок и так далее.

Модель памяти в Rust основывается на нескольких правилах:

1. У переменной в один момент времени может быть только один владелец;
2. Ссылки, или заимствования, могут быть двух видов: *разделяемыми* (shared, только для чтения) или *исключительными* (exclusive, чтение и запись);
3. На одну и ту же переменную в один момент времени может быть или сколь угодно много разделяемых ссылок, или одна и только исключительная ссылка;
4. Заимствования *всегда* указывают на валидную память.

За соблюдением этих правил на *этапе компиляции* отвечает алгоритм проверки заимствований (borrow checker). Для контроля правила 3 применяется механизм под названием *tree borrows* [10]. Для соблюдения правила 4 компилятор отслеживает времена жизни переменных и их заимствований. В большинстве случаев он сам способен вывести время

жизни, исходя из контекста, однако, если заимствование является полем некоторой структуры, требуется явное указание времени жизни этого заимствования.

Механизм проверки заимствований снимает с разработчика обязанность следить за временем жизни ссылок, указателей и переменных, поскольку на этапе компиляции гарантирует безопасный доступ к памяти, но вместе с этим накладывает ограничения на манипуляции с ней, даже если разработчик имеет полное представление о потоке данных в программе.

Часть работ по проверке заимствований может быть перенесена на этап выполнения программы с помощью типа `RefCell`, который является контейнером для любого типа данных. Доступ к содержимому этого контейнера включает в себя проверку правила 3. Так как эта проверка происходит уже во время выполнения программы, это может повысить гибкость – при ошибке проверки заимствования программа может сообщить о ней и восстановить нормальный режим работы.

Тем не менее столь строгая модель доступа к памяти требует более тщательного проектирования архитектуры программы: чрезмерное использование ссылок может снизить открытость исходного кода программы к изменениям ввиду сложности отслеживания заимствований.

Удобство использования

Удобство использования является в большей степени субъективным аспектом, однако у обоих рассматриваемых нами языков есть несколько объективных как преимуществ, так и недостатков, указывающих на то, какой из них более удобный и эргономичный в разработке.

Начнём с синтаксиса. Rust разрабатывался с учётом недостатков синтаксиса C++ и поэтому превосходит его в этом плане за счёт следующих улучшений:

1. Возможность операторов, таких, как, `if`, `match`, `while`, `loop` и так далее, возвращать значение в качестве результата своего выполнения, например:

```
1. fn main() {  
2.     let condition = true;  
3.     let number = if condition { 5 } else { 6 };  
4.  
5.     println!("The value of number is: {number}");  
6. }
```

В результате выполнения оператора `if` в строке 3 переменной `number` будет присвоено значение 5, которое и будет выведено на экран. Это позволяет объявлять переменные неизменяемыми и при этом инициализировать их в этой же инструкции, если инициализация требует сложной логики. В C++ схожей функциональности можно достичь с помощью немедленно вызываемых лямбда-выражений (`immediately invoked lambda expressions`) или использования расширений компиляторов;

2. Упрощённый синтаксис объявления лямбда-выражений, или замыканий (`closures`). В C++ при объявлении замыканий обязательным требованием является наличие фигурных скобок, обрамляющих тело замыкания, тогда как в Rust их можно опустить, если тело замыкания содержит лишь один оператор. Это упрощает визуальное восприятие исходного кода программы;

3. Упрощённый синтаксис обобщённых функций и типов данных. Так, в C++ требуется использование ключевых слов `template` и `typename` (или `class`) при объявлении шаблонных функций и классов. В Rust достаточно перечислить имена шаблонных типов в угловых скобках.

Однако синтаксис Rust'a не лишён некоторых спорных деталей:

1. Более частое использование знаков пунктуации, например, для обозначения времён жизни (`'lifetime`), явного задания типа переменной (`var: type`), разрешения области видимости (оператор `::`) и обозначения типа возвращаемого функцией значения (`->`);
2. Сокращённые ключевые слова: `fn`, `mut`, `impl`, `str`, `mod`, `pub`, `dyn`;
3. Принудительные соглашения о наименовании сущностей в программе: к примеру, по умолчанию компилятор требует, чтобы имена переменных и функций писались с

использованием `snake_case`, имена типов – `CamelCase`, а константы – `SCREAMING_SNAKE_CASE` [11].

В совокупности эти обстоятельства, даже с учётом субъективного отношения к ним, могут оттолкнуть некоторых разработчиков несмотря на остальные достоинства языка.

Помимо более современного синтаксиса Rust предоставляет богатый набор инструментов для управления проектами и исходным кодом:

1. `rustup` – для установки, удаления и обновления других инструментов экосистемы языка;
2. `cargo` – система сборки и управления зависимостями проектов;
3. `rustfmt` – форматирование исходного кода;
4. `rustdoc` – для генерирования документации проекта на основе комментариев в исходном коде, написанных в специальном формате;
5. `rust-analyzer` – языковой сервер [12], и служит основой для реализаций интегрированных сред разработки;
6. `rustc` – компилятор Rust.

Наличие единой и развитой экосистемы вокруг языка в значительной мере упрощает разработку и развёртывание программ на разных платформах.

С этой точки зрения C++ проигрывает Rust’у, поскольку у первого существует множество систем сборки (`cmake`, `xmake`, `MSBuild`, `Makefiles`, `Meson` и т.д.), управления зависимостями (`conan`, `vcpkg`) и компиляторов (`clang`, `gcc`, `msvc` и др.), однако они не являются чем-то единым и стандартизированным, что усложняет их использование между собой, а также перенос программ с одной платформы на другую.

В дополнение ко всему выше сказанному Rust имеет встроенную поддержку юнит-тестирования и удобный механизм организации исходного кода с помощью модулей. В C++ до сих пор используется разделение исходных файлов на заголовочные и файлы реализации, а поддержка модулей не до конца реализована во всех основных компиляторах – `clang`, `gcc` и `msvc`.

Тем не менее за счёт своего возраста сообщество C++ успело создать большое число библиотек кода, покрывающих практически все возможные случаи, включая, но не ограничиваясь, графические интерфейсы, игровые движки, библиотеки для работы с сетью, синтаксические анализаторы, эмуляция языковых возможностей, реализации протоколов и так далее. Далеко не всегда на Rust может найтись библиотека, предоставляющая требуемый функционал. Это одна из причин высокой популярности C++ – на момент написания статьи язык занимает второе место в рейтинге TIOBE [13].

Заключение

На основе проведённого сравнения можно сделать следующие выводы:

1. За счёт своей гибкости и высокой свободы, даваемой разработчику, C++ подойдёт для программ, требования к которым часто меняются. Язык позволяет быстро вносить в исходный код изменения и при этом отложить исправление возможных ошибок до момента, когда требования к программе стабилизируются. Rust, с другой стороны, требует чаще проводить рефакторинг для соблюдения правил, накладываемых системой типов и алгоритмом проверки ссылок;
2. Rust же может зарекомендовать себя в проектах с повышенными требованиями к безопасности. Этот язык позволяет выразить сложные и заранее определённые правила с помощью своей системы типов. Также Rust подойдёт для разработки новых проектов с нуля, не привязанных к уже существующей кодовой базе, благодаря богатому набору утилит, входящих в его экосистему. Переписывать, однако, проект с C++ на Rust может оказаться нецелесообразным с точки зрения результата и потраченных на это ресурсов.

Библиографический список

1. Cybersecurity Information Sheet -- Software Memory Safety. — Текст : электронный // U.S. Department of Defense : [сайт]. — URL: https://media.defense.gov/2022/Nov/10/2003112742/-1/-1/0/CSI_SOFTWARE_MEMORY_SAFETY.PDF (дата обращения: 14.04.2025).

2. Aleksandr, Ryabikov C++ Memory safety (memsafe) single-header libraries and Clang compiler plugin for safe C++, which reduces errors for reference data types and safe memory management without breaking backward compatibility with old C++ code. / Ryabikov Aleksandr. — Текст : электронный // GitHub : [сайт]. — URL: <https://github.com/rsashka/memsafe> (дата обращения: 14.04.2025).
3. Bjarne, Stroustrup Safety Profiles: Type-and-resource Safe programming in ISO Standard C++ / Stroustrup Bjarne. — Текст : электронный // Open Standards : [сайт]. — URL: <https://www.open-std.org/jtc1/sc22/wg21/docs/papers/2023/p2816r0.pdf> (дата обращения: 14.04.2025).
4. Sean, Baxter Safe C++, P3390R0 / Baxter Sean. — Текст : электронный // Safe C++ : [сайт]. — URL: <https://safecpp.org/P3390R0.html> (дата обращения: 14.04.2025).
5. ISO/IEC 14882:1998 - Programming languages — C++. — Текст : электронный // ISO - International Organization for Standardization : [сайт]. — URL: <https://www.iso.org/standard/25845.html> (дата обращения: 14.04.2025).
6. Object-Oriented Programming Features of Rust. — Текст : электронный // The Rust Programming Language : [сайт]. — URL: <https://doc.rust-lang.org/book/ch18-00-oop.html> (дата обращения: 14.04.2025).
7. C++ reference. — Текст : электронный // User-defined conversion function : [сайт]. — URL: https://en.cppreference.com/w/cpp/language/cast_operator (дата обращения: 14.04.2025).
8. Pattern Matching, P1371R0 / Sergei Murzin, Michael Park, David Sankel, Dan Sarginson // Programming Language C++ Evolution, JTC1/SC22/WG21 - The C++ Standards Committee - ISOCPP : электронный журнал. — URL: <https://wg21.link/p1371r0>. — Дата публикации: 21.01.2019.
9. Bowen, Fu match(it): A lightweight single-header pattern-matching library for C++17 with macro-free APIs. / Fu Bowen. — Текст : электронный // GitHub : [сайт]. — URL: <https://github.com/BowenFu/matchit.cpp> (дата обращения: 14.04.2025).
10. Neven, Villani Tree Borrows / Villani Neven. — Текст : электронный // Neven Villani (Vanille) : [сайт]. — URL: <https://perso.crans.org/vanille/treebor/> (дата обращения: 14.04.2025).
11. Naming. — Текст : электронный // Rust API Guidelines : [сайт]. — URL: <https://rust-lang.github.io/api-guidelines/naming.html> (дата обращения: 14.04.2025).
12. Language Server Protocol. — Текст : электронный // Official page for Language Server Protocol : [сайт]. — URL: <https://microsoft.github.io/language-server-protocol/> (дата обращения: 14.04.2025).
13. Paul, Jansen TIOBE Index for April 2025 / Jansen Paul. — Текст : электронный // TIOBE : [сайт]. — URL: <https://www.tiobe.com/tiobe-index/> (дата обращения: 14.04.2025).

УДК 681.5

Минаев Н.С.

РЕАЛИЗАЦИЯ СИСТЕМЫ ОБНОВЛЕНИЯ ПРОШИВКИ НА ЯЗЫКЕ GO С ИСПОЛЬЗОВАНИЕМ ZIP-АРХИВОВ И ПРОВЕРКИ ХЭШЕЙ IMPLEMENTATION OF A FIRMWARE UPDATE SYSTEM IN GO USING ZIP ARCHIVES AND HASH CHECKING

*Филиал «Протвино» государственного университета «Дубна»
Секция «Информационные технологии»*

Автор: Минаев Никита Сергеевич, студент 4 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Питухин Петр Васильевич, кандидат физико-математических наук, доцент кафедры информационных технологий филиала «Протвино» государственного университета «Дубна».

Author: Minaev Nikita Sergeevich, 4th-year student of the Informatics and Computer Engineering program of the branch "Protvino" state University "Dubna".

Scientific supervisor: Pitukhin Petr Vasilievich, Candidate of Physical and Mathematical Sciences, Associate Professor of the Information Technology Department of the branch "Protvino" state University "Dubna".

Аннотация

В статье рассматривается разработка системы обновления прошивки для устройств, использующих модульный подход к организации программного обеспечения. Описана реализация системы на языке Go, которая эффективно обрабатывает ZIP-архивы с прошивками, осуществляет проверку их целостности с помощью хеширования.

Ключевые слова: обновление прошивки, язык GO, ZIP-архивы, хеширование, программирование, резервное копирование, командный интерфейс.

Abstract

The article examines the development of a firmware update system for devices utilizing a modular software architecture. It describes the implementation of a system in the Go programming language that efficiently processes firmware ZIP archives and verifies their integrity through hashing.

Keywords: firmware update, Go language, ZIP archives, hashing, programming, backup, command-line interface.

Цель работы

Целью данной работы является разработка системы обновления прошивки для устройства на языке Go, которая будет эффективно и надежно обрабатывать архивы прошивок в формате ZIP, обеспечивать их целостность с помощью хеширования, а также выполнять резервное копирование и прошивку отдельных модулей устройства. Особое внимание уделено оптимизации процесса обновления, обеспечению совместимости с различными версиями прошивки и созданию системы, способной к восстановлению после сбоя.

Объект и предмет исследования

Объект исследования - системы обновления прошивки для устройств, использующих модульный подход к организации программного обеспечения. Предмет исследования – методология и технологии реализации системы обновления прошивки с использованием языка Go, ZIP-архивов и хеширования.

Введение

В современном мире устройства, использующие встроенные системы и микроконтроллеры, часто нуждаются в регулярных обновлениях прошивки для устранения ошибок, добавления новых функций и улучшения безопасности. Процесс обновления прошивки является важной частью жизненного цикла таких устройств, поскольку он позволяет пользователям получить актуальную версию программного обеспечения и гарантирует стабильную работу устройства.

Многие устройства используют ZIP-архивы для хранения обновлений, а также системы хеширования для проверки целостности данных. В данной работе рассматривается разработка системы обновления прошивки для устройств на языке Go, который является эффективным инструментом для создания приложений, работающих с архивами и обладающих высокой производительностью.

Архитектура системы. Диаграмма последовательности

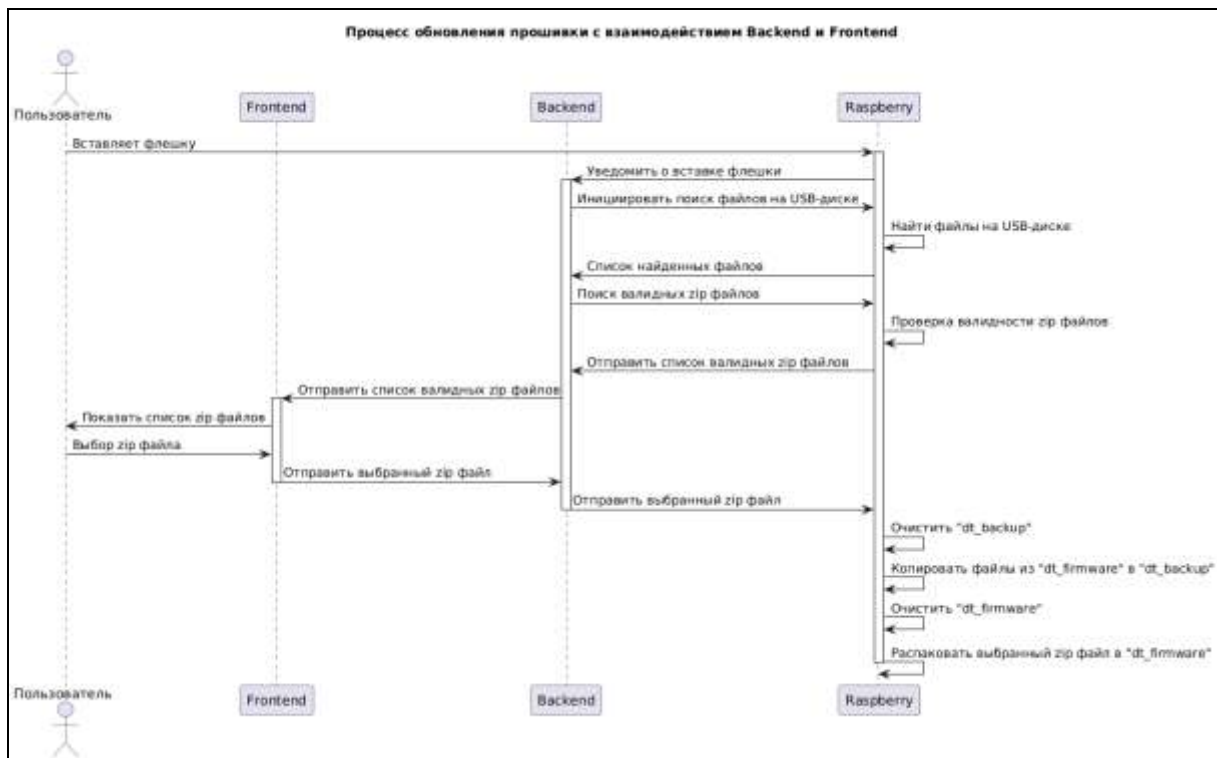


Рис.1. Процесс обновления прошивки

Методология

В ходе разработки системы обновления прошивки использовались следующие методики и технологии:

- Выбор языка программирования Go:** язык Go был выбран за его простоту в использовании, высокую производительность и отличную поддержку многозадачности. Это позволяет эффективно работать с архивами, выполнять операции параллельно и быстро обрабатывать большие объемы данных.
- Использование ZIP-архивов:** для хранения обновлений прошивки был выбран формат ZIP, поскольку он является стандартом де-факто для упаковки файлов и обеспечивает удобную работу с большими объемами данных.
- Проверка целостности с использованием хеширования:** для проверки целостности файлов прошивки используется хеширование. Каждый файл имеет заранее вычисленный хэш, который проверяется после извлечения из архива. Это позволяет гарантировать, что файл не был поврежден или изменен.
- Резервное копирование:** перед выполнением обновления система автоматически создает резервные копии всех файлов, чтобы в случае неудачи можно было вернуть устройство в рабочее состояние.
- Прошивка отдельных модулей:** система позволяет обновлять отдельные модули устройства, такие как фронтенд, бэкенд и прошивку для STM32, что дает гибкость при обновлении различных компонентов устройства.

Реализация системы

Система обновления прошивки состоит из нескольких ключевых компонентов:

- Сравнение версий прошивки:** перед выполнением обновления система сначала проверяет текущую версию прошивки и сравнивает ее с версией, представленной в архиве. Это помогает избежать ненужных обновлений и сэкономить ресурсы устройства.

2. **Резервное копирование файлов:** перед извлечением новых файлов из ZIP-архива создается резервная копия существующих данных. Это необходимо для восстановления системы в случае неудачного обновления.
3. **Извлечение файлов из ZIP-архива:** используемый пакет `archive/zip` в Go позволяет эффективно извлекать файлы и проверять их на целостность с помощью хеширования. Важно, чтобы архив содержал только необходимые файлы с их соответствующими метаданными.
4. **Проверка хэшей:** после извлечения файлов система проверяет хэши каждого из них. Хешируемые файлы сравниваются с предварительно сохраненными хэшами, и если хэш не совпадает, файл считается поврежденным.
5. **Процесс обновления:** обновление прошивки начинается с извлечения и проверки файлов, затем осуществляется их копирование в соответствующие директории устройства. В случае прошивки STM32, используется интерфейс командной строки для передачи обновлений.
6. **Логирование:** система ведет журнал всех операций, включая успешные и неудачные попытки обновления, что позволяет отследить процесс обновления и выявить проблемы в случае сбоя.

Пример кода

Пример кода для извлечения файлов из ZIP-архива и проверки хэшей:

```
package main
```

```
import (
    "archive/zip"
    "crypto/sha256"
    "fmt"
    "io"
    "log"
    "os"
    "path/filepath"
)
// Функция для проверки хэша файла
func checkFileHash(filePath, expectedHash string) bool {
    file, err := os.Open(filePath)
    if err != nil {
        log.Fatal(err)
    }
    defer file.Close()

    hash := sha256.New()
    if _, err := io.Copy(hash, file); err != nil {
        log.Fatal(err)
    }
    calculatedHash := fmt.Sprintf("%x", hash.Sum(nil))
    return calculatedHash == expectedHash
}
// Функция для извлечения файлов из ZIP-архива
func extractZIP(zipPath, destPath string) error {
    reader, err := zip.OpenReader(zipPath)
    if err != nil {
        return err
    }
    defer reader.Close()
    for _, file := range reader.File {
        filePath := filepath.Join(destPath, file.Name)
```

```

    if file.FileInfo().IsDir() {
        os.MkdirAll(filePath, file.Mode())
    } else {
        outFile, err := os.Create(filePath)
        if err != nil {
            return err
        }
        defer outFile.Close()

        inFile, err := file.Open()
        if err != nil {
            return err
        }
        defer inFile.Close()

        if _, err := io.Copy(outFile, inFile); err != nil {
            return err
        }
    }
}
return nil
}

```

Тестирование и результаты

После разработки системы было проведено тестирование с использованием различных ZIP-архивов, содержащих файлы прошивки для разных устройств. Все обновления прошли успешно, и резервные копии были восстановлены в случае ошибок.

В ходе тестов была проверена как скорость работы системы, так и надежность процесса обновления. В случае с ошибками в файлах (например, при повреждении ZIP-архива) система автоматически отклоняла обновление и восстанавливала систему из резервной копии.

Заключение

Разработанная система обновления прошивки на языке Go с использованием ZIP-архивов и хеширования представляет собой надежное решение для обеспечения стабильной работы устройств. Использование Go позволило создать систему с высокой производительностью и минимальными ресурсными затратами. В дальнейшем систему можно адаптировать под другие устройства и улучшать, добавляя новые возможности, такие как обновление по сети или добавление дополнительных проверок безопасности.

Библиографический список

1. Гибсон, Дж. "Современные системы обновлений прошивки". Изд-во "Техносфера", 2019.
2. Хенриксен, К. "Go: Программирование и разработка приложений". Изд-во "Бином", 2018.
3. Шмидт, М. "Программирование с использованием архивов в Go". Изд-во "Практика", 2020.

УДК 681.5

Мусатов Р.В.

**РЕАЛИЗАЦИЯ СИСТЕМЫ УПРАВЛЕНИЯ БИОТЕХНОЛОГИЧЕСКИМ
АНАЛИЗАТОРОМ С ИСПОЛЬЗОВАНИЕМ CAN-ИНТЕРФЕЙСА НА ЯЗЫКЕ PYTHON
IMPLEMENTATION OF A CONTROL SYSTEM FOR A BIOTECHNOLOGICAL ANALYZER
USING A CAN INTERFACE IN PYTHON**

Автор: Мусатов Руслан Вадимович, студент 4 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Питухин Пётр Васильевич, кандидат физико-математических наук, доцент кафедры информационных технологий филиала «Протвино» государственного университета «Дубна».

Author: Musatov Ruslan Vadimovich, 4th-year student of the Informatics and Computer Engineering program of the branch "Protvino" state University "Dubna".

Scientific supervisor: Pitukhin Petr Vasilievich, Candidate of Physical and Mathematical Sciences, Associate Professor of the Information Technology Department of the branch "Protvino" state University "Dubna".

Аннотация

В статье рассматривается разработка системы управления биотехнологическим анализатором с использованием CAN-интерфейса и языка Python. Описана реализация системы, обеспечивающей контроль температуры, измерение флуоресценции и автоматизацию механических операций.

Ключевые слова: биотехнологический анализатор, Python, CAN-интерфейс, температурный контроль, флуоресценция, автоматизация

Abstract

The article discusses the development of a control system for a biotechnology analyzer using a CAN interface and the Python programming language. It describes the implementation of a system that provides temperature control, fluorescence measurement, and automation of mechanical operations.

Keywords: biotechnology analyzer, Python, CAN interface, temperature control, fluorescence, automation.

Цель работы

Целью данной работы является разработка системы управления биотехнологическим анализатором на языке Python, обеспечивающей:

- Точный контроль температурных профилей.
- Измерение флуоресценции в реальном времени.
- Автоматизацию механических операций (открытие/закрытие крышки, измерение высоты пробирок).
- Надежную передачу данных через CAN-интерфейс.

Объект и предмет исследования

Объект исследования - системы управления лабораторным оборудованием для ПЦР-анализа. Предмет исследования – методология и технологии реализации системы управления анализатором с использованием Python и CAN-интерфейса.

Введение

Современные биотехнологические анализаторы требуют комплексного подхода к управлению, объединяющего контроль температурных режимов, оптические измерения и механические операции. CAN-интерфейс обеспечивает надежную коммуникацию между компонентами системы, а Python позволяет быстро разрабатывать гибкие решения.

Архитектура системы

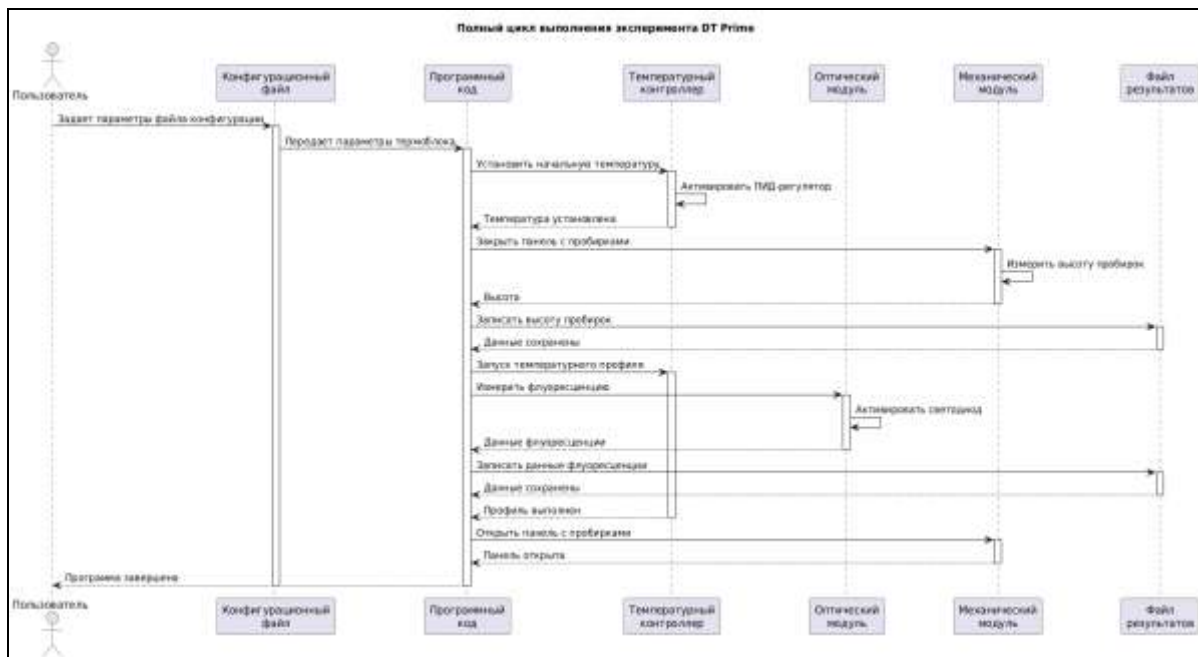


Рис. 1. Полный цикл выполнения эксперимента DT-Prime

Методология

В работе использованы следующие технологии:

1. Python - выбран за богатую экосистему библиотек и простоту разработки.
2. CAN-интерфейс - обеспечивает надежную связь с аппаратными модулями.
3. Многопоточность - для параллельного выполнения измерений и управления.
4. CSV-логирование - для сохранения экспериментальных данных.

Реализация системы

Основные компоненты:

1. Класс RealTime - ядро системы, обеспечивающее:
 - Обмен данными через CAN-интерфейс.
 - Управление аппаратными модулями.
 - Синхронизацию операций.

2. Температурный контроль:

```
def measure_temperature(self) -> list:
    response = self.send_cmd("XM", CanID.THERMO)
    if response.startswith("TP>"):
        return [float(temp) for temp in response[3:].strip().split()]
```

3. Измерение флуоресценции:

```
python
Copy
def measure_fluorescence(self, channel=3, led=211, exp=130):
    self.send_cmd(f"flt {int(exp/0.308)}", CanID.OPTIC)
    # ... измерение и обработка данных ...
    return fluorescence_values
```

4. Механические операции:

```
python
Copy
def close_lid(self):
    self.send_cmd("HCLOSE", CanID.MOTOR)
```

Пример выполнения программы:

```
device = RealTime() # Инициализация устройства
commands = parse_rta_file("protocol.rta") # Загрузка протокола
execute_commands(device, commands, "results.csv") # Запуск эксперимента
```

Тестирование и результаты

Система успешно протестирована в следующих режимах:

1. Длительные температурные профили (до 8 часов).
2. Параллельное измерение температуры и флуоресценции.
3. Автоматическое определение высоты пробирок.

Погрешности измерений:

- Температура: $\pm 0.1^{\circ}\text{C}$.
- Флуоресценция: $\pm 2\%$.
- Время отклика: < 100 мс.

Заключение

Разработанная система демонстрирует высокую надежность и точность управления биотехнологическим анализатором. Использование языка программирования Python позволило создать гибкое решение с возможностью дальнейшего расширения функциональности системы.

Библиографический список

1. CAN in Automation. "CAN Specification". - CiA, 2022
2. Tanenbaum A.S. "Structured Computer Organization". - Pearson, 2021
3. Python Software Foundation. "Python Documentation". - <https://docs.python.org>

УДК 681.51

Никольский Ф.В., Осморский В.А.

ПРОЕКТИРОВАНИЕ ДИСПЕТЧЕРА СОБЫТИЙ НА ПЛАТФОРМЕ ПЛИС FEATURES DESIGNING AN EVENT DISPATCHER ON AN FPGA PLATFORM

*Филиал «Протвино» государственного университета «Дубна»
Секция «Машиностроение»*

Авторы: Никольский Федор Владимирович и Осморский Владимир Александрович, студенты 4 курса направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Коковин Валерий Аркадьевич, к.т.н., доцент кафедры автоматизации технологических процессов и производств филиала «Протвино» государственного университета «Дубна».

Authors: Nikolsky Fedor Vladimirovich and Osmorsky Vladimir Alexandrovich, 4th year students of the Automation of Technological Processes and Production program of the branch "Protvino" state University "Dubna".

Scientific adviser: Kokovin Valery Arkadevich, candidate of technical sciences, associate professor of the department of automation of technological processes and production of the branch "Protvino" state University "Dubna".

Аннотация

В представленной работе приведены результаты разработки и исследования аппаратного цифрового устройства по управляемой обработке множества входных информационных сигналов (событий) на платформе программируемых логических интегральных схем (ПЛИС). Решается задача обработки, кодировки, запоминания и маршрутизации принятых событий с заданным приоритетом. Разработанное устройство позволяет выполнять масштабирование по числу обрабатываемых событий, обладает высоким быстродействием и легко интегрируется в

разрабатываемую или существующую автоматизированную систему управления любыми технологическими процессами. Для проверки соответствия разработанного устройства техническому заданию был разработан стенд в составе: оптические датчики, линейный модуль перемещения (ЛМП), сервопривод СПШ20, ПЛК 110-30.

Abstract

The presented work presents the results of development and research of a digital device for controlled processing of a set of input information signals (events) on the platform of programmable logic integrated circuits (FPGA). The problem of processing, encoding, storing and routing of received events with a given priority is solved. The developed device allows scaling by the number of processed events, has high speed and is easily integrated into a developed or existing automated control system for any technological processes. To check the compliance of the developed device with the technical specifications, a stand was developed consisting of: optical sensors, linear displacement module (LMP), servo SPSH20, PLC 110-30.

Ключевые слова: диспетчер событий, ПЛИС, автоматизированная система управления, Quartus Prime.

Keywords: event manager, FPGA, automated control system, Quartus Prime.

Структура любой автоматизированной системы управления (АСУ) содержит, как правило, стандартный набор функциональных подсистем: подсистема информационных датчиков, вычислитель, реализующий запрограммированный алгоритм задачи и исполнительные устройства. В централизованной АСУ информация от датчиков поступает прямо на подсистему ввода вычислительного устройства. Более сложно реализуется распределенная АСУ, когда алгоритм задачи распределяется между географически разнесенными подсистемами. В этом случае информацию от датчиков одной подсистемы необходимо передавать, с помощью коммуникационной сети, на другие подсистемы. Решение задачи приема и кодировки информации от датчиков выполняет устройство, названное диспетчером событий.

Актуальностью данной практической работы является построение диспетчера событий с минимальной задержкой по обработке событий на аппаратной платформе.

Основной **целью** является разработка и исследование компонентов обработки и распределения событий, обеспечивающих минимальное детерминированное время ее доставки потребителю. Для реализации поставленной цели необходимо решить следующие **задачи**: выполнить сравнительный анализ технических характеристик современных ПЛИС, оценить соотношение цены и быстродействия ПЛИС, разработать структурную схему стенда на основе промышленного оборудования для проверки работы диспетчера событий.

При выполнении практической работы были использованы следующие **методы**: разработка имитационной модели в пакете *AnyLogic*, выполнение временного моделирования в пакете *ModelSim Intel* для определения реального быстродействия диспетчера событий, проведение эксперимента на промышленном оборудовании для проверки работы диспетчера событий.

Объектом исследования является процесс обработки распределенных событий при управлении в реальном времени распределенными технологическими системами.

Предметом исследования является фактор времени, когда максимально быстрая обработка событий определяет достоверность информации. Таким образом, в работе решается проблема быстрой обработки событий и их кодировки. **Гипотезой** исследования является предположение о возможности реализовать проект диспетчера событий в современных ПЛИС с временем обработки одного события не хуже чем за 25 нс, при частоте сканирования 200 МГц.

В исследовательской работе решается **проблема**, связанная с проверкой возможности реализовать проект диспетчера событий с заданным быстродействием и проверкой его на промышленном оборудовании.

Анализ работ по выбранной тематике

Одной из целей исследования является решение задачи по анализу научных и практических работ по схожей тематике. Наглядным примером распределенной системы управления является АСУ ускорителя заряженных частиц [2,4]. Ускорительный комплекс

ИФВЭ состоит из трех ускорителей, каждый из которых распределен по большой территории комплекса. Для решения общей задачи, формирования пучка заряженных частиц заданной энергии, необходима синхронизация работы всех технологических подсистем комплекса. В этом случае информационные сигналы от датчиков распределенных подсистем необходимо специальным образом закодировать [2,5] и передать по сети MIL-STD-1553. Эта задача реализована аппаратно с использованием ПЛИС в системе таймирования. Разработанная общая таймерная система (ОТС) предназначена для генерации и распределения между контроллерами оборудования таймерной информации, необходимой для выполнения задач контроля и управления в рамках системы управления ускорительного комплекса ИФВЭ. Основными требованиями к ОТС являются открытость архитектуры и минимальное время доставки таймерной информации.

Похожую задачу решают авторы статьи [1,222], когда задачи по обработке информации распределяются между набором процессоров по заданному приоритету. Авторы отмечают потерю производительности, связанную со значительными временными затратами на выполнение процедур планирования и переключения контекстов задач при программной реализации диспетчеризации. Авторы статьи полагают, что одним из путей решения подобных проблем является аппаратная реализация функций диспетчеризации, которая в значительной степени снимает проблему временных потерь. Для этих целей использованы возможности аппаратной платформы ПЛИС.

Разработка проекта *Event_Manager* в пакете *Quartus Prime*

Анализ работ по рассматриваемой тематике показывает, что для достоверности событий, обрабатываемых диспетчером (для минимизации времени обработки диспетчером) следует при разработке диспетчера событий использовать аппаратный вариант, а именно ПЛИС. Главное достоинство ПЛИС – параллельная обработка больших массивов данных, что позволяет минимизировать время обработки.

Диспетчер событий реализован в виде проекта *Event_Manager* в пакете *Quartus Prime* фирмы *INTEL* на ПЛИС семейства *Cyclone 10 LP* [5]. Рассмотрим алгоритм работы проекта по функциональной схеме (рис. 1).

Ниже представлены обозначения сигналов, используемых в схеме (рис.1):

- *event*[3..0] - входные сигналы (события), поступающие на блок *LUT*;
- *clk* - глобальный тактовый сигнал от генератора 12 МГц, используемый для синхронизации работы всех устройств проекта;
- *clk1* - сигнал тактирования Схемы Двухфазной Синхронизации;
- *m*[3..0] - позиционный код, реализуемый Счетчиком и Дешифратором;
- *f1, f2* - сигналы тактирования блоков *RAM/FIFO* и *LUT*, разнесенные по времени на полпериода;
- *str* - сигнал начала работы блока Счетчик;
- *Adr*[3..0] - адреса ячеек памяти *RAM*, получаемые на выходе Сканера Входных Событий (*LUT*);
- *Data*[3..0] - закодированные данные об очередном событии, поступившем на вход Сканера Входных Событий (*LUT*).

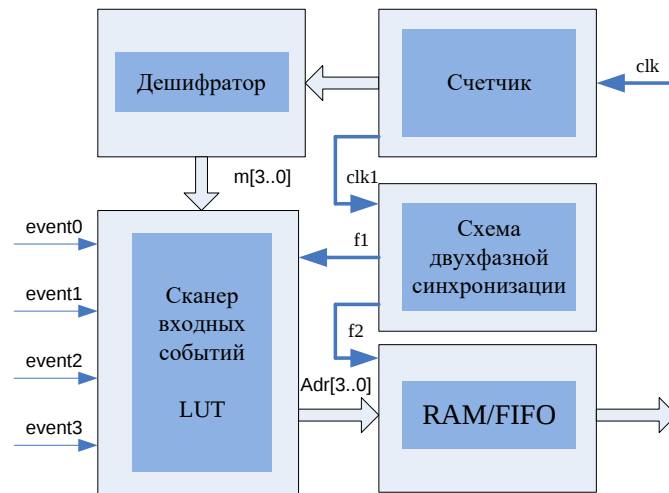


Рис. 1. Функциональная схема проекта *Event_Manager*

Работа диспетчера событий задается следующими требованиями:

- По сигналу “Сброс” все выходы входных триггеров устанавливаются в “0”, а блоки Счетчик и Дешифратор начинают формировать позиционный код $m[3..0]$, при этом логическая ‘1’ появляется в младшем разряде;
- Схема Двухфазной Синхронизации начинает формировать сигналы $f1$ и $f2$.
- Все входные события фиксируются на триггерах $T0..T3$ (в блоке *LUT*), которые сбрасываются в исходное состояние после обработки;
- Память *RAM* служит кодировщиком номера входного порта и добавлением служебной информации в передаваемое сообщение.
- Для согласования скорости обработки (сканирования) поступивших событий и скорости последовательной передачи данных используется эластичный буфер – *FIFO*.

Рассмотрим механизм сканирования входных сигналов $event[3..0]$ и превращения их в закодированные сообщения, подготовленные для передачи по коммуникационной сети. Для этого более детально представим блок Сканера Входных Событий (рис.2).

Алгоритм работы Сканера Входных Событий заключается в следующем:

1. После появления одного или нескольких событий устанавливаются в активное состояние соответствующие триггеры и на их выходах вырабатываются сигналы $p[3..0]$;
2. При совпадении активных сигналов p и m в каком-либо разряде, формируется один из сигналов $st[3..0]$, который вызывает останов счетчика и появление сигнала str ;
3. На адресных входах памяти *RAM*, с помощью сигнала $f1$, формируется адрес ячейки $Adr[3..0]$, в которой записаны адрес входного порта принятого события и служебная информация;
4. Сигнал str дает старт процессу передачи данных по последовательному каналу;
5. После обработки формируется сигнал $f2$, который сбрасывает триггер обработанного входа и запускает работу счетчика.

Алгоритм работы Сканера Входных Событий заключается в следующем:

- ✓ После появления одного или нескольких событий устанавливаются в активное состояние соответствующие триггеры и на их выходах вырабатываются сигналы $p[3..0]$.
- ✓ При совпадении активных сигналов p и m в каком-либо разряде, формируется один из сигналов $st[3..0]$, который формирует сигнала str и вызывает останов счетчика.
- ✓ На адресных входах памяти *RAM*, с помощью сигнала $f1$ и текущего значения сигналов $st[3..0]$, формируется адрес ячейки $Adr[3..0]$, в которой записаны номер входного порта принятого события и служебная информация.

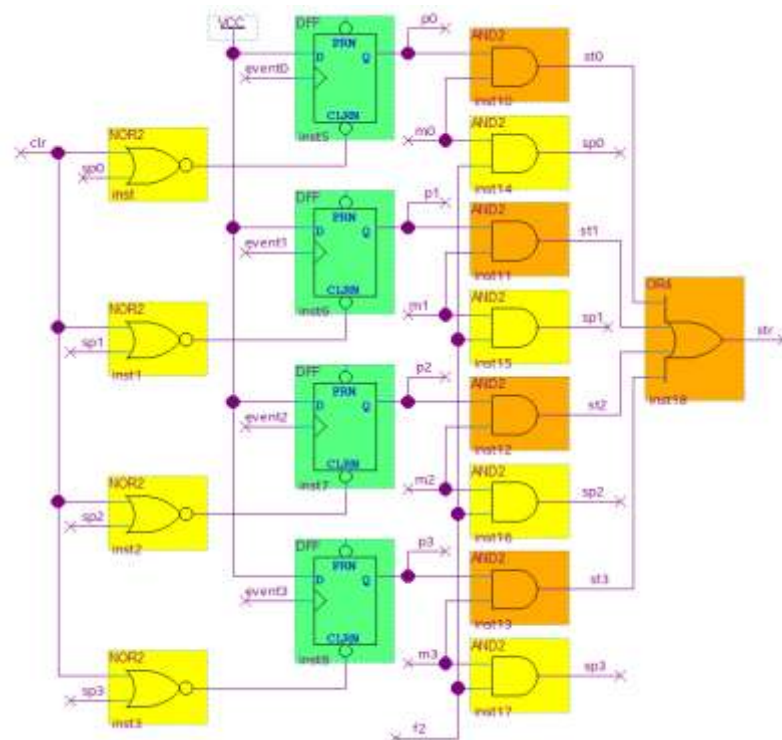


Рис. 2. Обработка параллельных потоков событий методом сканирования

- ✓ После обработки каждого активного триггера формируется сигнал $f2$, который сбрасывает триггер обработанного входа и запускает работу счетчика.
- ✓ Содержимое ячейки с текущим адресом $Adr[3..0]$ помещается в память типа *FIFO*.
- ✓ Далее запускается процесс передачи данных по последовательному каналу.

Разработка стенда для испытаний работы диспетчера событий

Для проверки работоспособности разработанного диспетчера событий с реальным оборудованием был собран стенд, состоящий из линейного модуля перемещения (ЛМП) с сервоприводом СПШ-20 [4], двух оптических датчиков, платы *CLK148* на ПЛИС с диспетчером событий и промышленного робота ЦПР-1П с управлением от ПЛК110. Для проверки работы была разработаны следующие программы: в пакете *MotoMaster* программа перемещения каретки ЛМП, в пакете *CoDeSys* программа управления роботом в ПЛК110 [3] и в ПЛИС проект *Event_Manager*. После запуска программы *MotoMaster* и загрузки всех программ начинается перемещение каретки ЛМП с пересечением оптической оси датчиков. При замыкании оптических датчиков сигнал приходит на ПЛИС, который после обработки диспетчером событий формирует сигнал *Pusk* на вход ПЛК. Этот сигнал запускает работу робота по заданному алгоритму. На рис.3 представлена диаграмма алгоритма работы оборудования испытательного стенда.

Перед началом работы испытательного стенда загружаются программы в ПЛК линейного модуля перемещения, загружается проект в ПЛИС с модулем диспетчера событий, а также загружается программ в ПЛК 110 фирмы Овен.

Далее для линейного модуля перемещения на компьютере запускается в Мотомастере разработанная программа и каретка модуля начинается перемещаться по направляющим.

При пересечении оптической оси датчиков происходит формирование события которое поступает на диспетчер событий реализованный на ПЛИС.

При заданной очередности срабатывания первого датчика и второго происходит формирование в диспетчере событий кода события, который поступает на ПЛК. После этого, согласно алгоритма программы ПЛК, происходит инициализация работы программы по управлению роботом.

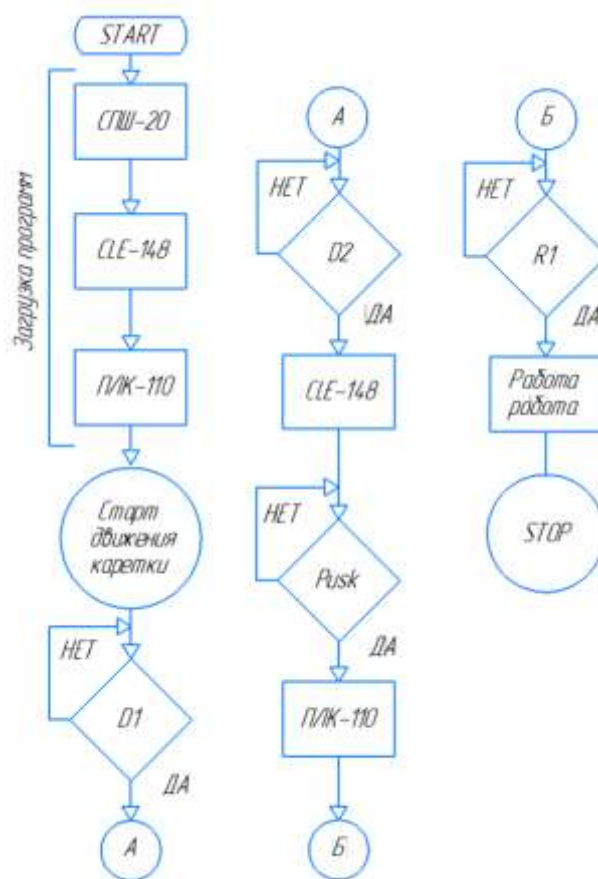


Рис. 3. Алгоритм работы испытательного стенда

Проверка работы диспетчера событий на стенде показала соответствие разработанного алгоритма и полученных результатов. Временное моделирование в пакете *ModelSim Intel* показало время обработки одного события не хуже чем за 25 нс, при частоте сканирования 200 МГц, что свидетельствует о подтверждении высказанной гипотезы. Уменьшение времени обработки возможно при использовании ПЛИС с большими скоростными характеристиками. На собранном стенде были проведены эксперименты с ЛМП, при которых рассматривались различные варианты пересечения оптических осей датчиков. В итоге была выбрана конфигурация, при которой сигналы на входы диспетчера событий передается при последовательном пересечении сначала оптической оси датчика D2, а затем датчика D1, и не передается при движении каретки ЛМП в обратном направлении. К достоинствам разработки можно отнести масштабируемость по числу событий и высокую скорость сканирования событий.

Библиографический список

1. Бикташев, Р. А. Реализация устройства аппаратной поддержки диспетчеризации задач для многопроцессорной системы на ПЛИС / Р. А. Бикташев, С. К. Шестаков // Оптико-электронные приборы и устройства в системах распознавания образов, обработки изображений и символьной информации. Распознавание -2013: Сборник статей XI международной научно-технической конференции. - Курск: ЮЗГТУ. - 2013. - С. 220 - 223.
2. Комаров, В. В. Базовые принципы построения общей таймерной системы ускорительного комплекса У-70 / В. В. Комаров. - Протвино: Гос. науч. центр Рос. Федерации Ин-т физики высоких энергий, 2004. - 7 с.: ил., табл.; 28 см. - (Государственный научный центр Российской Федерации Институт физики высоких энергий (ИФВЭ); 2004-46 ОУ У-70.
3. Программируемый логический контроллер ПЛК110, https://owen.ru/product/plk110_m02

4. Сервопривод СПШ20, <http://zaozmi.ru/catalog/spsh-tech.html>
5. TEI0003-03-QFCR4A Trenz Electronic FPGA module with Intel Cyclone 10 LP FPGA, <https://wiki.trenz-electronic.de/display/PD/TEI0003>

УДК 004.432

Овсянников Е.С.

РАЗРАБОТКА ВЕБ-СЕРВИСА ДЛЯ АВТОМАТИЗИРОВАННОЙ ГЕНЕРАЦИИ И ПОДПИСАНИЯ ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСЬЮ PDF-ДОКУМЕНТОВ
DEVELOPMENT OF A WEB SERVICE FOR AUTOMATED GENERATION AND SIGNING OF PDF DOCUMENTS WITH DIGITAL SIGNATURE

*Филиал «Протвино» государственного университета «Дубна»
Секция «Информационные технологии»*

Автор: Овсянников Евгений Сергеевич, студент 4-го курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Кульман Татьяна Николаевна, к.т.н., доцент кафедры информационных технологий филиала «Протвино» государственного университета «Дубна».

Author: Ovsyannikov Evgeny Sergeevich, 4th year student of the Computer Science and Engineering program of the branch "Protvino" state University "Dubna".

Scientific adviser: Kulman Tatyana Nikolaevna, Ph.D., associate professor of the department information technology of the branch "Protvino" state University "Dubna".

Аннотация

Статья рассматривает процесс разработки веб-сервиса для автоматизированной генерации и подписания PDF-документов с помощью электронной цифровой подписи. Описаны ключевые аспекты реализации, включая выбор технологий, архитектурные решения и механизмы обеспечения безопасности.

Abstract

The article examines the development of a web service for automated generation and signing of PDF documents. It describes key implementation aspects, including technology selection, architectural solutions, and security mechanisms

Ключевые слова: веб-сервис, автоматизация, генерация документов, подписание PDF, электронные документы, безопасность.

Keywords: web service, automation, document generation, PDF signing, electronic documents, security.

Объект работы – процесс автоматизированной генерации и подписания PDF-документов.

Предмет работы – разработка веб-сервиса для автоматизированной генерации и подписания PDF-документов.

Цель данной работы – создание веб-сервиса [1], который автоматизирует процесс формирования и подписания PDF-документов, обеспечивая удобство, безопасность и сокращение времени обработки юридически значимых документов. Для достижения поставленной цели требуется решение следующих задач:

- разработать механизм автоматической генерации PDF-документов на основе HTML-шаблонов с динамическим заполнением данных;
- реализовать функционал обработки цифровых сертификатов для формирования электронной подписи;
- создать веб-интерфейс, позволяющий пользователям вводить данные и автоматически формировать документы;
- обеспечить интеграцию сервиса с другими системами для полного цикла обработки документов;

- протестировать разработанное решение для подтверждения его стабильности и соответствия требованиям.

Методы работы: анализ литературы и технологий, проектирование архитектуры, программная разработка, тестирование, документирование процесса разработки.

Гипотеза: автоматизация процесса генерации и подписания PDF-документов позволит значительно повысить скорость, удобство и безопасность работы с юридически значимыми документами.

Современные компании сталкиваются с необходимостью оптимизации внутренних процессов из-за роста объемов информации и требований к скорости обработки данных [2]. Одним из ключевых направлений такой оптимизации является автоматизация документооборота, включающая процессы создания, подписания, хранения и обмена документами. Автоматизация особенно актуальна для юридической и финансовой сфер, где точность и безопасность данных играют решающую роль.

Для лучшего понимания используемых в документе терминов и технологий, рассмотрим ключевые понятия:

1. PDF (Portable Document Format) - универсальный формат электронных документов, который сохраняет форматирование и структуру документа независимо от устройства или программного обеспечения.
2. HTML-шаблон - структура веб-страницы, используемая для динамического формирования документов с заполнением заданных полей.
3. Электронная подпись (ЭП) - цифровой эквивалент собственноручной подписи, обеспечивающий целостность и подлинность электронных документов.
4. КриптоПро - программное обеспечение для реализации криптографической защиты информации, включая создание и проверку электронной подписи.
5. Цифровой сертификат - электронный документ, подтверждающий подлинность ключа электронной подписи и связывающий его с владельцем.
6. REST API (Representational State Transfer Application Programming Interface) - стандартизированный подход к организации взаимодействия между программными компонентами через HTTP-протокол, обеспечивающий простоту интеграции и масштабируемость системы.
7. Микросервисная архитектура - подход к разработке программного обеспечения как набора небольших независимых сервисов, взаимодействующих через четко определенные API.

Обработка юридически значимых документов требует инструментов, обеспечивающих их корректное оформление, подписание и защиту от несанкционированных изменений [5].

Внедрение автоматизированных сервисов позволяет исключить ручное заполнение и подписание документов, сокращая время обработки, минимизировать ошибки за счёт стандартизации процессов, повысить уровень безопасности с применением криптографических технологий и обеспечить соответствие законодательным нормам.

Была разработана система, которая охватывает ключевые технологии, используемые для автоматизации документооборота: генерацию PDF-документов на основе HTML-шаблонов с использованием библиотеки Openhtmltopdf [4], электронную подпись документов с помощью криптографических инструментов, таких как КриптоПро [6], и интеграцию с государственными реестрами.

Система реализует электронную подпись документов с использованием современных криптографических алгоритмов асимметричного шифрования. Применение цифровых сертификатов и инфраструктуры открытых ключей (PKI) обеспечивает надежную аутентификацию и защиту документов от несанкционированных изменений [7]. Используемые криптографические методы гарантируют надежность и безопасность процессов обработки данных:

- целостность документа (проверку отсутствия изменений после подписания),
- неотрекаемость (невозможность отказа от авторства подписи),
- конфиденциальность данных.

Решение поддерживает работу с сертифицированными средствами электронной подписи, что придает документам юридическую силу, равнозначную собственноручно подписанным бумажным документам.

Это позволяет оперативно получать выписки из реестров саморегулируемых организаций (СРО), что является важной частью системы, так как выписки из этих реестров подтверждают статус и полномочия организаций и специалистов. В системе реализована работа с 7 реестрами СРО:

- СРО АУ – реестр саморегулируемых организаций арбитражных управляющих.
- СРО О – реестр саморегулируемых организаций оценщиков.
- СРО КИ – реестр саморегулируемых организаций кадастровых инженеров.
- СРО Медиаторов – реестр саморегулируемых организаций, занимающихся профессиональным урегулированием споров.
- СРО ОЭП – реестр саморегулируемых организаций операторов электронных площадок.
- СРО для специалистов в области банкротства.
- СРО для специалистов в области юридических услуг.

Теперь рассмотрим взаимодействие сервисов между собой.

Система построена на микросервисной архитектуре и включает следующие ключевые компоненты: сервис генерации PDF [4], сервис подписания PDF [6] и сервис веб-формы [3].

Все сервисы взаимодействуют через REST API. Взаимодействие сервисов представлено на рисунке 1.

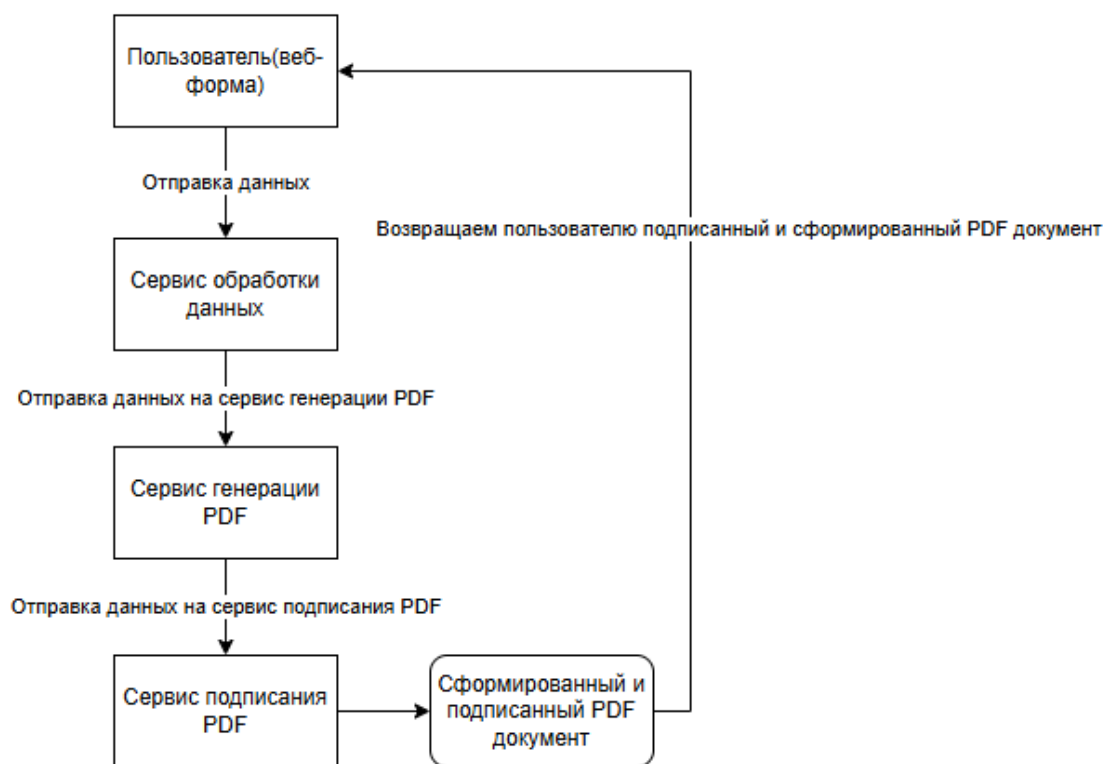


Рис. 1. Схема взаимодействия компонентов системы

Процесс взаимодействия сервисов можно описать следующим образом:

1. Веб-форма: Пользователь заполняет форму с данными о своей организации. Данные отправляются на сервер через AJAX-запросы, что позволяет избежать перезагрузки страницы.
2. Сервис генерации PDF: После того, как данные получены от веб-формы, сервис генерации PDF обрабатывает информацию, подставляя её в заранее подготовленные HTML-шаблоны. Он использует библиотеку Openhtmltopdf для создания PDF-

документа, сохраняя все стили и форматирование. Этот документ затем передается на сервис подписания.

3. Сервис подписания PDF: Сгенерированный PDF-документ передается на сервис подписания, который накладывает на него электронную подпись с использованием криптографических технологий, таких как КриптоПро. Подпись делает документ юридически значимым и защищенным от подделок.

4. Возвращение подписанного документа пользователю: Подписанный документ возвращается обратно через сервис веб-формы, и пользователь получает возможность скачать документ или получить его по электронной почте.

Таким образом, все компоненты системы: веб-форма, сервис генерации PDF и сервис подписания PDF – взаимодействуют между собой через REST API, что позволяет эффективно управлять процессом создания и подписания документов. Этот подход обеспечивает четкое разделение ответственности между сервисами, повышает масштабируемость системы и минимизирует риски, связанные с ручным трудом.

Разработанная система автоматизирует процессы документооборота, что позволяет значительно сократить время на обработку документов, повысить точность и прозрачность процессов, минимизировать влияние человеческого фактора и обеспечить удобный доступ к актуальной информации через интеграцию с государственными реестрами.

Система позволяет повысить эффективность работы организаций, снизить операционные затраты и обеспечить безопасность данных. Разработанная архитектура, основанная на современных технологиях, предоставляет гибкость в адаптации под различные потребности бизнеса, обеспечивая масштабируемость и соответствие нормативным требованиям.

Заключение

В ходе работы была реализована система для автоматизированной генерации и подписания PDF-документов, который решает ключевые задачи современного документооборота: повышение скорости обработки данных, минимизацию ошибок и обеспечение юридической значимости документов.

Основные результаты работы включают:

- Реализацию механизма генерации PDF на основе HTML-шаблонов с динамическим заполнением данных, что обеспечивает гибкость и удобство при создании документов.
- Интеграцию криптографических инструментов для формирования электронной подписи, гарантирующей целостность и аутентичность документов.
- Разработку интуитивного веб-интерфейса, упрощающего взаимодействие пользователей с системой.
- Обеспечение взаимодействия сервисов через REST API в рамках микросервисной архитектуры, что повышает масштабируемость и надежность системы.
- Поддержку работы с ключевыми реестрами саморегулируемых организаций (СРО), что расширяет функциональность сервиса для юридических и финансовых задач.

Таким образом, разработанный веб-сервис представляет собой эффективное решение для автоматизации документооборота, сочетающее удобство, безопасность и соответствие современным требованиям цифровизации бизнес-процессов.

Библиографический список

1. Бендик, М. А. Spring Boot. Создание современных веб-приложений / М. А. Бендик. – Санкт-Петербург: Питер, 2018. – 384 с.
2. Вершинин, А. И. Программирование на языке Java: учебник / А. И. Вершинин. – 2-е изд., перераб. и доп. – Москва: ДМК Пресс, 2020. – 472 с.
3. Исаев, А. А. Разработка веб-сервисов с использованием Java и Spring Framework: учебник / А. А. Исаев. – Москва: ДМК Пресс, 2019. – 432 с.
4. Никифоров, С. В. Основы работы с библиотеками для генерации PDF-документов в Java / С. В. Никифоров. – 3-е изд., перераб. и доп. – Екатеринбург: УрФУ, 2021. – 256 с.
5. Фролов, И. М. Основы криптографии и защиты информации: учебное пособие / И. М. Фролов. – Москва: Бином, 2019. – 324 с.

6. Ханов, В. Р. Основы цифровых подписей и криптографической защиты данных: монография / В. Р. Ханов, В. Д. Иванов. – Москва: КНОРУС, 2019. – 400 с.
7. Шевченко, В. В. Операции с PDF-документами в Java: руководство / В. В. Шевченко. – Москва: НТ Пресс, 2017. – 208 с.

УДК 004.457

Примакова Т.Г.

**ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФРАКТАЛОВ В СРЕДЕ РАЗРАБОТКИ MICROSOFT
VISUAL STUDIO 2022**
SOFTWARE IMPLEMENTATION OF FRACTALS IN THE MICROSOFT VISUAL STUDIO 2022
DEVELOPMENT ENVIRONMENT

*Филиал «Протвино» государственного университета «Дубна»
Секция «Информационные технологии»*

Автор: Примакова Татьяна Георгиевна, студентка 3 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна»

Научный руководитель: Губаева Милета Михайловна, старший преподаватель кафедры Информационных технологий филиала «Протвино» государственного университета «Дубна».

Author: Primakova Tatiana Georgievna, 3rd year student of the Computer Science and Engineering program of the branch "Protvino" state University "Dubna".

Scientific supervisor: Gubaeva Miletа Mihailovna, Senior Lecturer at the Department of Information Technology of the branch "Protvino" state University "Dubna".

Аннотация

В статье рассматриваются применение телеграм-бота для совершенствования физической формы и реализация такого бота на языке Python.

Abstract

The article discusses the use of a telegram bot to improve physical fitness and the implementation of such a bot in Python.

Ключевые слова: телеграм-бот, программирование, физическая подготовка, Python, библиотека Telebot.

Keywords: telegram bot, programming, physical training, Python, Telebot library.

Цели работы:

1. Изучить процесс создания телеграм-бота на языке программирования Python.
2. Написать собственного чат-бота с использованием библиотеки telebot.

Каждый человек, ставивший цель заниматься каким-либо спортом или физкультурой, сталкивался с трудностями при составлении тренировочной программы. Эти трудности не зависят от квалификации, подготовки или подкованности в вопросе.

Необходимость составлять план упражнений присутствует вне зависимости от окружающих факторов. Будь то занятия в группе или в одиночестве, на улице или в помещении. План нужен как профессиональным спортсменам, чтобы поддерживать в тонусе мышцы всего тела, а не только задействованные, так и новичкам, ещё не разбирающимся в массивах информации о тренировках.

Множество конференций и лекций проводится с целью обучить спортсменов, тренеров из разных областей, интересующихся физической подготовкой и педагогов составлению программ на короткий и долгий сроки.

Тем не менее, в наши дни есть более удовлетворяющие пути решения проблемы. Ведь мало составить одну программу, которая будет задействовать максимум групп мышц. Во-первых, её необходимо регулировать для каждого человека с учётом конкретных противопоказаний. Во-вторых, она должна выполняться в определённые сроки, чтобы, в

зависимости от цели, не занимать или заполнять всё время тренировки. И, что самое трудновыполнимое, она должна, по крайней мере, раз в некоторое время существенно изменяться, ведь любое действие при многократном повторении приедается. А мотивация при физических нагрузках – один из ключевых двигателей на пути к успеху.

Таким образом, становится очевидно, что на решение описанной выше проблемы отдельные специалисты тратят множество часов, оставаясь в постоянном поиске и совершенствовании программ, тогда как хорошо сделанное программное обеспечение может справляться с задачей более эффективно, тратить меньше ресурсов и иметь единую, но обновляющуюся базу, с которой работать будет гораздо удобнее, чем с множеством источников.

Чат-боты в мессенджерах – это тот инструмент, который в современном мире активно используется и имеет множество применений благодаря нескольким факторам:

1. Привычная среда. Любой бот, так или иначе, передаёт информацию. Принимать или отправлять её для всех удобнее в привычных обстоятельствах.

2. Готовые библиотеки для программирования. Они позволяют не погружаться в новую среду разработки, а программировать на привычном языке с теми инструментами, которые созданы специально для платформы или мессенджера.

С помощью ботов выгоднее оптимизировать бизнес-процессы, будь то работа поддержки, рассылка новостей, обновление курса валют или магазин. Крупные компании сокращают затраты на операторов благодаря ботам. Даже малому бизнесу выгоднее оплатить создание бота и содержание базового сервера, чем нанять работника.

Телеграм особенно способствовал распространению ботов. Кроме него нельзя не упомянуть множество компаний, имеющих собственных ботов или позволяющих создавать помощников на их платформе: Mail.ru Group, Meta*, OZON, Сбер и многие другие.

* Признана в РФ экстремистской организацией и запрещена.

«Технологии становятся практичнее и упрощаются – сейчас конструировать чат-ботов можно во многих инструментах без специальной подготовки. Они добавляются во множество процедур в организациях. Поэтому объединение сотрудников, RPA-роботов, чат-ботов в эффективные процессы – одно из ключевых направлений, которое обуславливает развитие интеграций чат-ботов в комплексные решения и платформы по роботизации», – отмечает Павел Борченко (генеральный директор компании Robin, разработчика первой российской RPA-платформы).

Это неочевидно среднестатистическому пользователю, но чат-боты действительно со временем могут стать частью повседневной рутины для всех.

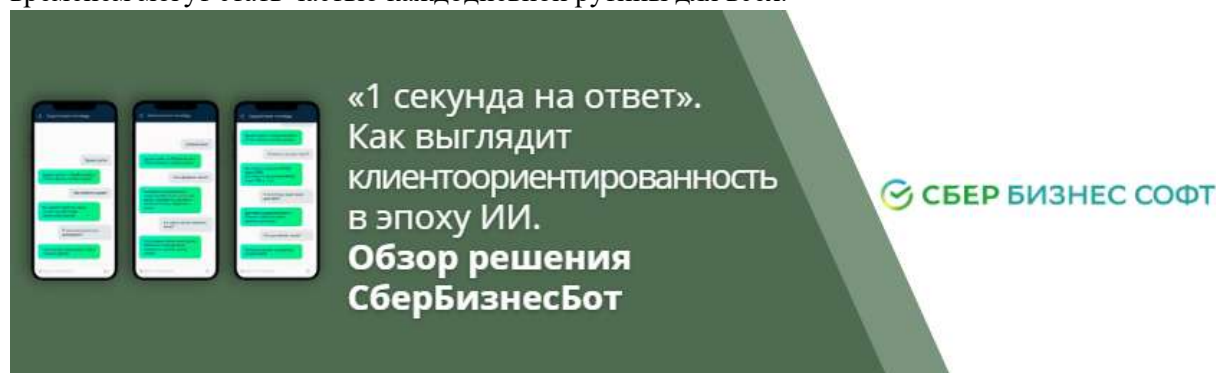


Рис. 1. Применение чат-ботов

Если рассматривать создание именно телеграмм-ботов, то для Python-разработчика есть несколько десятков разных библиотек, отвечающих данной цели. Они различаются популярностью, размером комьюнити и функциональностью. Самые популярные – это Aioogram (асинхронная), Python-telegram-bot (синхронная) и TeleBot (асинхронный и синхронный подходы на выбор). Асинхронное выполнение кода позволяет не останавливать работу бота в ожидании ответа пользователя.

При создании этого проекта использовалась библиотека Telebot. Именно по этой библиотеке нашлось достаточное количество качественного материала для обучения. В результате изучения библиотеки и функционирования телеграмм-ботов в целом стало ясно, что удобнее будет не вычленять из сообщений пользователя ключевую информацию и после чего переходить к выдаче упражнений, а создать структуру, переходы внутри которой будут осуществляться с помощью команд (изредка) и кнопок (в основном). Была изучена работа Inline, Reply и URL кнопок. В проекте используются все три вида кнопок для разных задач.

Пример использования Reply кнопки:

```
@bot.message_handler(commands=['start'])
def start(message):
    keyboard = types.ReplyKeyboardMarkup(row_width=5, resize_keyboard=True,
one_time_keyboard=True)
    button1 = types.KeyboardButton("☐ Поздороваться")
    keyboard.add(button1)
    bot.send_message(message.from_user.id, '☐ Добро пожаловать, ' + message.from_user.first_name
+ '.\n Я Ваш бот-помощник!', reply_markup=keyboard)

@bot.message_handler(content_types=['text'])
def call(message):
    if message.text == '☐ Поздороваться':
        menu(message)
```

В этом примере бот ожидает команды «/start» от пользователя, реагирует на неё приветственным сообщением и демонстрирует Reply кнопку "☐ Поздороваться". Если собеседник нажимает на неё, бот выводит командное меню.

```
def menu(message):
    bot.send_message(message.chat.id, '☐ /planner — ежедневник с добавлением задач на даты;' +
"\n☐ /workout_plan — составление программы тренировок.")
```

Основной функционал бота состоит в формировании соответствующей параметрам пользователя программе тренировок. Основные демонстрируемые разделы внутри программы – это:

1. 5 случайных упражнений на основные группы мышц;
2. Выбрать раздел и получить 5 упражнений на сегодня;
3. Упражнения для бегунов;
4. Разминка;
5. Отказ от ответственности.

С помощью такого распределения можно получить быстро и просто интересующую подборку для тренировки. Всего в боте более 180 упражнений, и, в зависимости от запроса, он выдаёт интересующую подборку, будь то упражнения для рук, живота, кардио или для бегунов – они нуждаются в особенной программе. Также бот напоминает пользователям о важности разминки для сохранения здоровья, из-за чего та вынесена в собственный раздел. Каждое упражнение бот сопровождает не только описанием верного выполнения, но и видео или gif материалом, чтобы изучение новых приёмов было нагляднее.

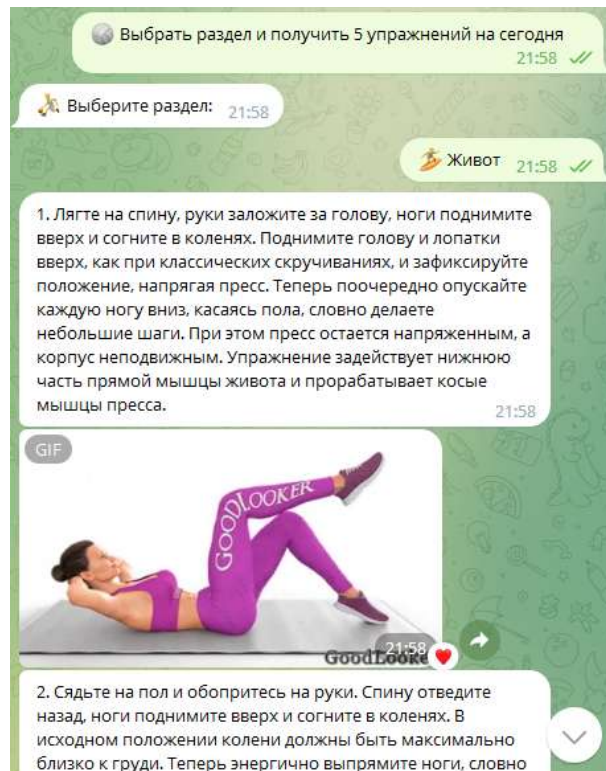


Рис. 2. Упражнения на мышцы живота с визуальным сопровождением

Важная особенность данного бота, почти не упомянутая ранее, заключается в том, что он способен не только составлять программы тренировок, но и вести ежедневник с задачами, заданными пользователем.



Рис. 3. Интерактивный ежедневник

На данный момент не подключена возможность запоминания ботом задач и выдачи их в конкретные временные промежутки, но эта функция в планах на реализацию в дальнейшем. С помощью базы данных в будущем этот бот сможет не только вносить задачи пользователя на конкретные даты и выдавать их, но и напоминать о них, а также об упражнениях. Более того, это откроет доступ к такой функции, как составление программы тренировки и распределение её на конкретный день с последующим напоминанием об этом. Таким образом, бот сможет выполнять функции не только умного ежедневника, но и персонального карманного настраиваемого тренера.

В работе были рассмотрены чат-боты и телеграм-боты в частности, а также разработан бот для составления тренировочных программ с минимальным количеством спортивного инвентаря. Для написания бота были использованы среда разработки программного обеспечения IDLE Shell 3.11.4 и библиотеки Telebot и Telebot_calendar, а также некоторые иные вспомогательные библиотеки для работы с датами и временем.

Использование данного бота расширяет кругозор интересующихся, усиливает познавательный интерес, развивает стремление к здоровому образу жизни, а также упрощает работу тех, кому необходимо на регулярной основе вести тренировки или уроки физкультуры.

Библиографический список

1. Стивен Ф. Лотт, Дастин Филлипс. Объектно-ориентированный Python - СПб.: Питер, 2024. - 706
2. John P. Mueller Beginning Programming with Python For Dummies. - Noida: Wiley India, 2014. - 408 с.

УДК 93/94

Сизонов Т.Р.

ИСТОРИЯ КУРСКОЙ БИТВЫ HISTORY OF THE BATTLE OF KURSK

*Филиал «Протвино» государственного университета «Дубна»
Секция «Социальные и гуманитарные науки»*

Автор: Сизонов Тихон Русланович, студент 1 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна»

Научный руководитель: Борисова Татьяна Юрьевна, магистр психолого-педагогического образования, ассистент кафедры общеобразовательных дисциплин филиала «Протвино» государственного Университета «Дубна».

Author: Sizonov Tikhon Ruslanovich, 1st year student of the direction "Computer Science and Engineering" of the branch "Protvino" state University "Dubna".

Scientific adviser: Borisova Tatyana Yurievna, Master of Psychological and Pedagogical Education, Assistant of the Department of General Subjects, of the branch "Protvino" state University "Dubna".

Аннотация

В статье рассматривается история подготовки и проведения Курской битвы как одного из важнейших сражений Великой Отечественной войны, предопределившей исход Второй мировой войны, излагается стратегия Курско-Белгородской наступательной операции. Анализируются соотношения количества противостоящих соединений с каждой из сторон, участвующих в величайшем танковом сражении Второй мировой войны, приводятся примеры героизма и мужества участников Курской битвы на Прохоровом поле. Раскрывается переломное влияние Курской битвы не только на ход Великой Отечественной войны, но и определяющее значение на итоги Второй мировой войны.

Abstract

The article examines the history of the preparation and conduct of the Battle of Kursk as one of the most important battles of the Great Patriotic War, which predetermined the outcome of World War II, sets out the strategy of the Kursk-Belgorod offensive operation. The ratios of the number of opposing formations on each side participating in the greatest tank battle of World War II are analyzed, examples of the heroism and courage of the participants in the Battle of Kursk on the Prokhorov Field are given. The turning point of the Battle of Kursk is revealed not only on the course of the Great Patriotic War, but also the decisive importance on the results of the Second World War.

Ключевые слова: Великая отечественная война, Курская битва, военная операция, история и стратегия, танковые соединения, итоги и значение.

Keywords: Great Patriotic War, Battle of Kursk, military operation, history and strategy, tank formations, results and significance.

«Если битва под Сталинградом предвещала закат немецко-фашистской армии, то битва под Курском поставила её перед катастрофой».

И.В. Сталин, 6 ноября 1943 г.

Обращение к истории Курской битвы, одного из поворотных сражений Великой Отечественной войны, вызывается особым величием этого события в контексте предстоящего празднования 80-летия Победы над фашизмом в 1945 году. Это величие связано прежде всего с концентрацией бронетанковых войск на относительно небольшой территории, с упорной и ожесточенной борьбой и стремлением любой ценой добиться победы над противником.

Мы помним, что коренным переломом в Великой Отечественной войне является Сталинградская битва, которая завершилась победой советской армии и дальнейшим наступлением. Последующие наступательные действия советской армии и ожесточенное ответное сопротивление немецких армий во время Харьковской оборонительной операции привели к созданию сложной ситуации в районе Курско-Белгородского плацдарма.

Образовавшаяся Курская дуга, как называли дугообразную линию фронта, могла дать преимущество как нашим, так и немецким войскам. По карте видно, что преимущество войск Красной Армии состояло в возможности зайти с флангов и разбить армии врага поодиночке, что привело бы к поражению немецких групп армии «Центр» и армии «Юг». Однако опасность для наших войск состояла в образовавшихся благоприятных для немецкой группировки условий по окружению советских войск в районе Курска и образованию Курского «котла».



Весна 1943 г. прошла под знаком споров о дальнейшем ходе войны. Советское командование понимало, что после поражения в Сталинградской битве Гитлеру просто необходимо использовать выгодное положение своих войск и нанести удар в районе Курского выступа. В то же время и Советский Союз мог использовать свои преимущества, тем более, что к этому времени мы превосходили Германию во всех сферах военной деятельности.

Так, например, блестяще работала наша разведка. О том, что немецкое командование готовит план генерального наступления в районе Курска, Генеральному штабу было известно уже в начале 1943 года из переданного в апреле 1943 года Сталину точного текста плана операции «Цитадель». Подписанная всеми службами вермахта директива плана операции ожидала лишь подписи Гитлера, который завизировал ее только 15 апреля, в то время как разведчик, добывший этот документ для советского командования, передал его в Москву уже 12 апреля. Об этом разведчике нам известно лишь его кодовое имя «Вертер» и больше ничего,

хотя некоторые исследователи и полагают, что таким советским агентом был личный фотограф Гитлера.

Итак, план «Цитадель» заключался в том, чтобы двумя ударами прорвать оборону советских фронтов, окружить советские войска, взяв их в котёл. Гитлер сосредоточил на курском направлении большое число дивизий, и главную роль среди них играли танковые дивизии. Не секрет, что Вторая Мировая война – это война танков, тяжелого оружия. Именно поэтому в фашистской Германии постоянно шла разработка новых видов вооружения, среди которых особо выделялись тяжелые мощные танки «Тигр», которому отводилась решающая роль в прорыве обороны советских войск, а также принятые немцами на вооружение в канун Курской битвы новый средний танк «Пантера» и истребитель танков САУ «Фердинанд». В дополнение к этому были существенно перевооружены немецкие военно-воздушные войска, которые пополнились истребителями «Фокке-Вульф» и штурмовиками «Хеншель 129», на применение которых немцы возлагали особые надежды.

Как мы видим, операция «Цитадель» готовилась со всей тщательностью. Германия нарастила темпы вооружения и перевооружения армии, активно комплектовала свои резервы и стягивала к Курской дуге значительные силы. Видимо из-за этого сроки начала и проведения операции переносились минимум четыре раза. Если по первоначальному замыслу «Цитадель» должна была начаться в мае, то началась она, в итоге, 5 июля 1943 года, что дало советским войскам возможность скрытно и хорошо подготовиться к отражению готовящегося удара.

Решающее значение оказало знание советским командованием планов противника, которое позволило для предупреждения удара сосредоточить на курском направлении значительные силы для защиты плацдарма по всей Курской дуге двумя фронтами. Со стороны Орла находился Центральный фронт под командованием Константина Константиновича Рокоссовского. Со стороны Белгорода концентрировался Воронежский фронт под командованием Николая Фёдоровича Ватутина. К началу Курской битвы общие соотношения сил были с незначительным перевесом в сторону немецких войск.

Общее соотношение сил и средств к началу июля 1943 года

	Немецкие войска	Советские войска
Численность войск	1 337 000	900 000
Численность орудий	19 300	10 000
Численность танков	3 300	2 700
Численность самолетов	2 650	2 000

Уже в тот момент, когда стало известно о планах противника, в Генштабе прошло совещание по вопросу о выборе характера предстоящих стратегических действий: оборонительных или наступательных. Единства по этому вопросу среди советских военачальников не было. Переход к обороне отстаивал генерал армии К.К. Рокоссовский. Н.Ф. Ватутин выступал за проведение наступательной операции. Рокоссовский обратил внимание на количество танков, которое было у немцев: больше, чем у советских войск. К тому же, было неизвестно, какое преимущество в бою будет у новых фашистских танков. Выбирая же оборонительную стратегию, можно было заранее создать многополосную оборонительную линию, максимально использовать преимущества ландшафта. Сталин поддержал позицию Рокоссовского. Совещание закончилось беспрецедентным решением: в районе Курской дуги советские войска переходят в оборону! Сторона, имевшая преимущество во времени, силах и средствах, только что переломившая ход войны, переходила к обороне. Это уникальный случай в истории!

Командующий Центральным фронтом не ошибся и проявил себя как блестящий стратег. По сути, Курская дуга была превращена в настоящую крепость. Подготовка глубокоэшелонированной обороны велась, прежде всего, как противотанковая. Но в результате получилась настолько мощная оборонительная система, что она сработала и как

противопехотная, и как противотанковая, и даже противовоздушная. Для этого при постройке всех препятствий и заграждений максимально использовались все особенности ландшафта, которые позволяли не только дополнительно защитить выстроенные фортификации, но и надежно замаскировать их в пересекающихся оврагах, небольших лесных массивах по склонам речек и воссозданных небольших деревьях.

Необычным и новаторским решением было проведение артиллерийской контрподготовки за час до начала наступления немецких войск, во время которой был нанесен мощный удар по местам сосредоточения живой силы и техники, а параллельно с этим наши воздушные армии Воронежского и Юго-Западного фронтов совершили налёт на полевые аэродромы врага. Точность и сила упреждающего удара была обеспечена действиями нашей разведки, обеспечившей получение фактической информации о действиях немецкой группировки «Юг» и возможность оперативной корректировки складывающейся на поле боя ситуации.

Проведенный контрудар привел к тому, что фашистские войска пошли в наступление с опозданием на час, поэтому крупнейшее танковое сражение Великой Отечественной войны – битва на Курской дуге, началось с поражения планов немецкого командования, а продолжилось полным их крахом в период с 5 июля по 23 августа 1943 года. За 50 дней на Курской дуге прошло множество боестолкновений. Некоторые из них были настолько ожесточенными, что сохранились в памяти как отдельные сражения и битвы. На Курской дуге ярко проявился героический характер советской армии, ярко проявившийся в примерах самоотверженности и стойкости духа советских летчиков и танкистов, ставших главными героями этой битвы.

Так, на сайте музея-диорамы в описании истории Курской битвы приводится несколько документальных свидетельств о примерах такого героизма в воздухе и на земле. Среди них особенно впечатляет «рассказ о Вальдемаре Сергеевиче Шаландине – гвардии лейтенанте, возглавлявшем танковую роту. Когда экипаж Шаландина стали обходить несколько вражеских танков, танк Вальдемара Шаландина вступил в бой с двумя «Тиграми» и одной «Пантерой». Гвардии лейтенант подпустил противника максимально близко и на дистанции прямого выстрела ударил вражеские танки по бортам. В этой танковой дуэли экипаж танка Шаландина уничтожил все три немецких танка и удержал занимаемый рубеж, несмотря на то, что его танк был подбит и загорелся. За этот бой гвардии лейтенант В.С. Шаландин посмертно был удостоен звания Героя Советского Союза» [1].

Второе свидетельство содержит рассказ о героях-летчиках – Иване Кожедубе, Александре Горовце и Алексее Маресьеве. «Младший лейтенант Иван Кожедуб одержал свои первые победы именно в Курской битве, а впоследствии стал самым результативным советским летчиком периода Великой Отечественной войны, за что был трижды удостоен звания Герой Советского Союза. ... Гвардии старший лейтенант Александр Горовец сбил в ходе битвы девять вражеских самолетов и был удостоен звания Героя Советского Союза посмертно. ... Летчик Алексей Маресьев, который к тому времени был известен тем, что после потери обеих ног научился летать на протезах, отлично проявил себя во время Курской битвы. Его долго не пускали в полёт, опасаясь, что он не справится с управлением в трудную минуту. Но в разгар боёв Маресьев совершал вылеты наравне со всеми. За время участия в Курской битве лётчик-истребитель Алексей Маресьев сбил 7 самолётов противника» [1].

В начале июля атаке подвергся крупный узел сопротивления советских войск — железнодорожная станция Прохоровка. Если Сталинградская битва – переломный момент всей войны, то танковое сражение под Прохоровкой - переломный момент Курской битвы. Это самое массовое танковое сражение, которое когда-либо знала история войны. 12 июля 1943 года под Прохоровкой сражались около 1300 танков с обеих сторон. Эта тяжелая, изнурительная битва закончилась неопределённо. Ни одна из сторон не сумела достичь намеченных целей. Если немецкие войска не смогли прорвать оборонительный рубеж, то и советским войскам не удалось перехватив инициативу, полностью разгромить вражеские войска.

Мы знаем несколько фильмов о сражении под Прохоровкой, но ни один из них не является документальным. Удивительный факт того, почему не сохранилось ни одного кадра Прохоровского танкового сражения, хотя на Курской дуге велась фото и кинохроника, можно попытаться объяснить не только несовершенством фото и киноаппаратуры тех лет, но и ужасом

той страшной картины Прохоровского сражения, что оставили его непосредственные участники. О том, насколько тяжелая, трагическая и героическая одновременно была обстановка на Прохоровском поле, оставили свои воспоминания непосредственные участники тех событий. Вот, например, танкист, старший техник-лейтенант Г.И. Пэнэжко писал: «... В памяти остались тяжёлые картины... Стоял такой грохот, что перепонки давило, кровь текла из ушей. Сплошной рев моторов, лязганье металла, грохот, взрывы снарядов, дикий скрежет разрываемого железа... От выстрелов в упор сворачивало башни, скручивало орудия, лопалась броня, взрывались танки. От выстрелов в бензобаки танки мгновенно вспыхивали. Открывались люки, и танковые экипажи пытались выбраться наружу. Я видел молодого лейтенанта, наполовину сгоревшего, повисшего на броне. Раненый, он не мог выбраться из люка. Так и погиб. Не было никого рядом, чтобы помочь ему. Мы потеряли ощущение времени, не чувствовали ни жажды, ни зноя, ни даже ударов в тесной кабине танка. Одна мысль, одно стремление — пока жив, бей врага» [2].

Выжившие участники этой страшной битвы в своих воспоминаниях рассказывали, как выбравшиеся из подбитых, горящих танков наши танкисты шли в рукопашную с немецкими танкистами, а если немецкий экипаж не выходил из танка, то наши солдаты в ярости стреляли в танки из пистолетов и били автоматами по люкам, чтобы «выкурить» оттуда врага. Если удавалось потушить подбитый танк, то советские солдаты снова забирались в него и шли в бой.

В последующие три дня немцы усилили натиск, ожесточенные бои велись чуть южнее Прохоровки. Советским войскам удалось не только выдержать этот натиск, но и откинуть немцев назад, в сторону Белгорода. С этого момента мы перешли к контрнаступательным операциям. Многие знают название плана Гитлера — «Цитадель». У наших контрнаступательных операций также были кодовые названия - «Кутузов» и «Полководец Румянцев».

Операция «Кутузов» была направлена на разгром группы армии «Центр» на орловском направлении. Развитие нашей операции хотя и не привело к полному разгрому группы армии «Центр», но нанесла ей крупное поражение и вынудила отступить за оборонительный рубеж «Хаген», восточнее Брянска. Завершением операции «Кутузов» произошло 5 августа 1943 года с вхождением советских войск в Орёл и освобождением города.

Операция «Полководец Румянцев» ставила своей задачей штурм Белгорода, который в это время удерживали фашистские войска. Немцы крепко засели в Белгороде, «превратив город в мощный оборонительный узел. Противотанковые рвы, огневые точки в каменных строениях, минные поля — таков был оборонительный обвод вокруг города. В самом Белгороде для ведения уличных боев были приспособлены целые жилые кварталы. Нашим войскам пришлось наступать по болотным низинам, в то время как противник владел высотами Меловых гор и заранее пристрелял возможные подходы к городу» [1].

Освобождение Белгорода началось с действий нашей авиации и артиллерии, которые своими ударами прокладывали дорогу советской пехоте. Ожесточенные бои шли буквально за каждый квартал, а иногда и за каждый дом, потому что фашисты предпринимали отчаянные попытки удержать свои позиции и оставить город в своих руках. Советские войска проявили необычайное мужество, настойчивость и упорство. О характере боев говорит количество погибших немецких солдат и офицеров. Более 3000 фашистов были убиты в Белгороде, который был освобожден к вечеру 5 августа.

Завершением наступательных операций «Кутузов» и «Полководец Румянцев» стало практически одновременное освобождение двух крупных города — Орла и Белгорода. Это было знаменательное событие, свидетельствовавшее о коренном переломе в ходе войны, в ознаменование которого в Москве 5 августа 1943 года был дан первый в истории Великой Отечественной войны артиллерийский салют! Страна салютовала не только доблести и храбрости советских воинов, но 12 залпами из 120 орудий было открыто заявлено о наступлении нового долгожданного этапа Великой Отечественной войны — освободительного, свидетельствовавшего, что война пошла в обратную сторону — на запад!

Подтверждением этого стало завершение Курской битвы, в ходе которой была освобождена в последующие дни территория Левобережной Украины, а 23 августа начался

успешный штурм города Харькова, который завершился освобождением Харькова. Так завершилась Курская битва, которая длилась 50 тяжелых дней и ночей в небе и на земле

Завершение Курской битвы ознаменовало собой окончательный перелом не только в Великой Отечественной войне, но и во Второй мировой войне. Более 30 немецких дивизий были разгромлены. Потери численного состава – полмиллиона человек. Наибольшие потери понесли танковые войска противника и авиация: было уничтожено 1600 танков и более 3500 самолетов. Курская битва привела к существенному изменению соотношения сил в войне и создала благоприятные условия для дальнейшего наступления Красной Армии.

«Главный инспектор танковых войск Вермахта генерал Х. Гудериан вынужден был признать: В результате провала наступления «Цитадель» мы потерпели решительное поражение. Бронетанковые войска, пополненные с таким большим трудом, из-за больших потерь в людях и технике на долгое время были выведены из строя... Инициатива окончательно перешла к русским» [1].

Библиографический список

1. Белгородский государственный историко-художественный музей-диорама «Курская битва. Белгородское направление», электронный ресурс// <https://31md.ru/istoriya-kurskoy-bitvy> (дата обращения 08.04.2025).
2. Государственный военно-исторический музей-заповедник «Прохоровское поле», электронный ресурс// <https://xn---ctbjbwiaaccdifcs7d.xn--p1ai/muzej-trete-ratnoe-pole-rossii.html> (дата обращения: 19.03.2025).
3. История курской битвы, электронный ресурс // <https://bitva.kursk.ru/history> (дата обращения: 09.04.2025).

УДК 67.02

Телипов И.В.

3D-ПЕЧАТЬ: КРАТКИЙ ОБЗОР МЕТОДОВ ПЕЧАТИ ПЛАСТИКОМ И ОПЫТ ИСПОЛЬЗОВАНИЯ FDM-ТЕХНОЛОГИИ 3D PRINTING: A BRIEF OVERVIEW OF PLASTIC PRINTING METHODS AND EXPERIENCE USING FDM TECHNOLOGY

Филиал «Протвино» государственного университета «Дубна»
Секция «Машиностроение»

Автор: Телипов Игорь Викторович, студент 3 курса направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Курзуков Геннадий Вячеславович, старший преподаватель кафедры «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Author: Telipov Igor Viktorovich, 3rd year student of the program "Automation of technological processes and productions" of the branch "Protvino" state University "Dubna".

Scientific supervisor: Kurzukov Gennady Vyacheslavovich, Senior Lecturer at the Department of Automation of Technological Processes and Productions of the branch "Protvino" state University "Dubna".

Аннотация

Рассматриваются методы 3D-печати для изготовления изделий. Приводится пример изготовления детали с использованием FDM-технологии.

Abstract

3D printing methods for manufacturing products are considered. An example of manufacturing a part using FDM technology is given.

Ключевые слова: 3D-печать, аддитивные технологии, 3D-принтер, прототипирование, 3D-модель, CAD-система, программа-слайсер.

Keywords: 3D printing, additive technologies, 3D printer, prototyping, 3D model, CAD system, slicer program.

3D-печать, или трёхмерная печать, представляет собой процесс изготовления деталей путем добавления материала, в отличие от механической обработки и традиционного формообразующего производства. Создание изделий таким способом относится к аддитивным технологиям (от английского *add* - добавлять).

С помощью 3D-печати автоматически производятся компоненты из различных материалов, таких как пластмассы, металлы и сплавы, биоматериалы, керамика. Методы 3D-печати используются в дизайне, строительстве и на транспорте, в производственной, аэрокосмической и медицинской отраслях [1].

3D-печать осуществляется с использованием 3D-принтеров и применяется, в частности, для прототипирования - изготовления объекта по чертежу или математической модели. В данном случае, аддитивное прототипирование - это процесс создания физического объекта при помощи 3D-принтеров методом послойного нанесения его структуры в соответствии с математической моделью [2].

Существует два основных вида 3D-печати пластиком.

FDM-печать (Fused Deposition Modeling) [3] - наиболее распространённая и доступная технология, суть которой заключается в послойном наплавлении пластиковой нити через нагретое сопло экструдера. Экструдер, закреплённый на трёхосевой системе позиционирования, перемещается по координатам X, Y и Z, нанося расплавленный материал в заданных программой местах. После нанесения каждого слоя материал быстро затвердевает, при этом для некоторых видов пластиков используется принудительное охлаждение с помощью вентиляторов.

FDM-системы предоставляют возможность гибкой настройки различных параметров печати, включая:

- температуру нагрева экструдера и рабочей платформы
- скорость перемещения печатающей головки
- толщину наносимого слоя
- интенсивность охлаждения

Эти параметры задаются непосредственно перед началом печатного процесса и во многом определяют качество конечного изделия.

При проектировании моделей для печати необходимо учитывать два критически важных параметра: габариты рабочей области принтера и толщину слоя. Для бытовых 3D-принтеров характерна область печати 200×200×200 мм, тогда как промышленные установки могут достигать 1000×1000×1000 мм. При необходимости печати крупногабаритных изделий на компактных устройствах модель можно разделить на составные части с последующей сборкой.

Толщина слоя в FDM-печати варьируется от 50 до 400 мкм и задаётся на этапе подготовки модели. Уменьшение толщины слоя повышает точность воспроизведения сложной геометрии и качество поверхности, тогда как увеличение сокращает время и стоимость печати. Оптимальным балансом между скоростью и качеством считается диапазон 150-200 мкм.

Фотополимерные технологии (SLA/DLP/LCD) основаны на принципе послойного отверждения жидкой смолы в прозрачной ванне. Различия между ними заключаются в источнике излучения: DLP использует проектор, LCD - жидкокристаллическую матрицу, а SLA - ультрафиолетовый лазер.

Каждая из технологий имеет свои **преимущества и ограничения**.

FDM-печать отличается простотой хранения и использования материалов, экономичностью производства, высокой скоростью изготовления, достаточной прочностью изделий.

Ограничения: сниженная точность мелких деталей, минимальный размер элементов около 0,1 мм.

Фотополимерная печать обеспечивает высокую детализацию (ограничена лишь разрешением проектора), идеальную гладкость поверхностей.

Недостатки: ограничения по габаритам изделий, более медленный процесс печати, необходимость постобработки изделий.

Выполненная проектная работа, заключается в изготовлении крыльчатки охлаждения поршневого компрессора (рис. 1), пришедшей в негодность в ходе эксплуатации. Таким образом, крыльчатка является **объектом** проекта.

Предмет проекта – различные характеристики крыльчатки: размеры, материал, механические свойства, стоимость изготовления.

Актуальность работы обусловлена отсутствием данной детали и ее аналогов в свободной продаже, что делает 3D-печать единственным доступным способом восстановления узла.

Технологическое обоснование. Выбор FDM-технологии обусловлен двумя ключевыми факторами:

- материал оригинальной детали – пластмасса;
- возможности FDM-печати по работе с полимерными материалами.

Основная рабочая гипотеза: FDM-технология пригодна для изготовления вышедшей из строя детали.

Цель проекта - создание работоспособной детали средствами 3D-печати.

Для достижения цели проекта необходимо решить **задачи**:

- выполнить точные замеры и создать цифровую 3D-модель в CAD-системе;
- конвертировать модель в формат, пригодный для печати;
- разработать управляющую программу с оптимальными параметрами печати;
- осуществить процесс послойного синтеза детали.

Контроль качества. После изготовления требуется: провести сравнительный анализ геометрии изделия; проверить функциональные характеристики; оценить соответствие техническим требованиям

Методологическая база. В работе применялся комплексный подход, включающий теоретический анализ, практические измерения, компьютерное моделирование, экспериментальную проверку.

Выполнение проекта

Разработка модели начинается с создания цифрового прототипа в CAD-среде с возможностью экспорта в STL-формат. Этот этап аналогичен традиционному 3D-моделированию, но требует дополнительных операций по подготовке к печати. Ключевые аспекты подготовки: определение геометрических параметров; оптимизация модели для FDM-технологии; учет технологических ограничений оборудования.



Рис. 1. Крыльчатка компрессора, требующая замены

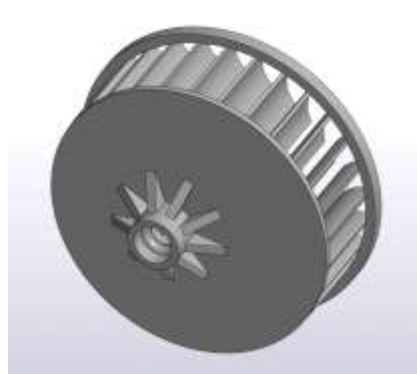


Рис. 2. Модель крыльчатки, созданная в КОМПАС-3D

В практике реверс-инжиниринга (восстановления чертежей и электронных моделей по готовой детали) существует несколько принципиально разных подходов. Наиболее доступным, хотя и требующим значительных временных затрат, остается ручной метод с использованием

стандартных измерительных инструментов - штангенциркуля, линейки и угломера. Этот способ не требует специального оборудования, но отличается высокой трудоемкостью.

Современным технологическим решением проблемы является 3D-сканирование. Однако применение этого метода связано со значительными финансовыми затратами как на приобретение аппаратуры, так и на проведение самих работ.

Компромиссным вариантом между стоимостью и автоматизацией процесса выступает фотограмметрия, которая является разновидностью 3D-сканирования. Этот способ особенно эффективен при работе со сложными формами, не требующими сверхвысокой точности.

В рамках данного проекта был выбран традиционный метод ручного реверс-инжиниринга.

Разработка цифровой модели осуществлялась в CAD-системе КОМПАС-3D, что позволило создать точную модель оригинальной крыльчатки. Полученная в результате модель представлена на рисунке 2.

После завершения проектирования трехмерная модель экспортируется в универсальный STL-формат, после чего загружается в специализированное программное обеспечение для генерации управляющего G-кода. Этот этап является критически важным, поскольку 3D-принтер, будучи, по своей сути, устройством с числовым программным управлением, требует точных команд для выполнения печати. Современные программы-слайсеры существенно упрощают процесс подготовки, автоматически преобразуя геометрическую модель в последовательность управляющих команд.

В процессе обработки модели программное обеспечение не только генерирует траектории движения печатающей головки, но и автоматически добавляет все необходимые технологические параметры: температурные режимы экструдера и рабочей платформы, процедуры калибровки и позиционирования, а также завершающие операции по охлаждению и отводу печатающего узла.

Для реализации данного проекта был выбран популярный слайсер Cura 3D от компании Ultimaker [4], который сочетает в себе широкий функционал с интуитивно понятным интерфейсом.

На рисунке 3 представлен вид модели в интерфейсе программы Cura 3D. Особенностью данного программного обеспечения является визуализация потенциально проблемных участков модели, где возможно провисание материала. Критические зоны выделяются цветовыми маркерами, что позволяет оператору заранее оценить необходимость использования поддерживающих структур или внесения изменений в параметры печати. Такой подход значительно повышает вероятность успешного завершения печатного процесса и получения качественного изделия.

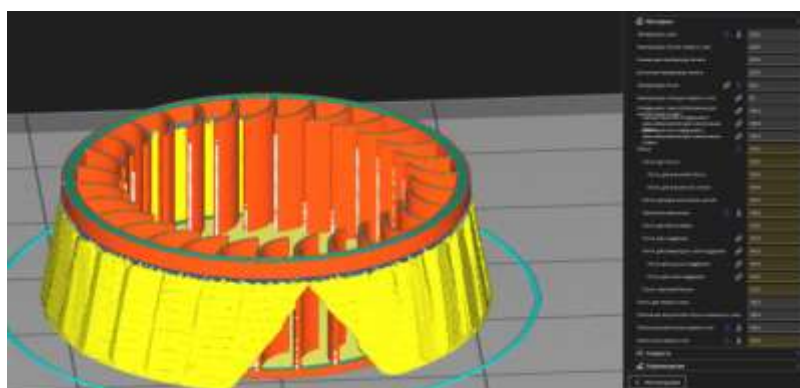


Рис. 3. Модель крыльчатки в программе-слайсере Cura 3D



Рис. 4. Крыльчатка, созданная методом 3D-печати

После задания значений температуры стола и экструдера, соответствующих физическим свойствам пластика, программа отправляется на 3D-принтер для печати детали. Созданная деталь показана на рисунке 4.

После полного охлаждения печатной платформы и рабочей поверхности можно приступать к извлечению готового изделия. На этом этапе требуется аккуратно удалить вспомогательные конструкции и выполнить финишную обработку, которая включает очистку детали от остатков поддерживающих структур и случайных пластиковых нитей, образовавшихся в процессе печати.

Для оценки качества готового изделия был проведен сравнительный анализ геометрических параметров. Он показал, что изготовленная деталь в полной мере соответствует оригиналу по форме и размерам. Установка новой крыльчатки в компрессорную систему подтвердила ее работоспособность - деталь успешно выдерживает эксплуатационные нагрузки и выполняет свои функции.

Выводы по результатам проекта

Факт создания полностью функциональной детали с требуемыми техническими характеристиками свидетельствует о достижении цели проекта. Все этапы работы, от проектирования до финальной проверки, были выполнены в полном объеме, что в конечном итоге привело к успешному результату.

Применение FDM-технологии, в данном случае, полностью оправдало себя, что подтверждает первоначальную гипотезу о возможности использования данного метода 3D-печати для решения поставленной производственной задачи.

Успешная реализация проекта подтвердила эффективность FDM-технологии для изготовления функциональных деталей, перспективность применения 3D-печати в ремонтных работах.

Библиографический список

1. Введение в аддитивные технологии. Т. 1. Обзор основных технологий 3D-печати : учебник / М.В. Канищев, Л.М. Ульев. – Москва : Издательский Дом НИТУ «МИСиС», 2023. – 352 с.
2. Основы компьютерной графики: 3D-моделирование и 3D-печать : учебное пособие / В. В. Лисяк ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2021. – 109 с.
3. Метод послойного наплавления материала (FDM). – URL: <https://iqb.ru/materials-and-technologies/technologies/fdm/> (дата обращения: 07.04.2025)
4. Учебник по Cura 3D — как пользоваться программой-слайсером Cura. – URL: <https://3dpt.ru/blogs/support/cura> (дата обращения: 07.04.2025)

УДК 321.01

Тихомиров М.А.

СЦИЕНТИСТСКАЯ МОДЕЛЬ ОБЩЕСТВА В ПРОИЗВЕДЕНИИ Ф. БЭКОНА «НОВАЯ АТЛАНТИДА»

THE SCIENTIST MODEL OF SOCIETY IN F. BACON'S WORK «NEW ATLANTIS»

*Филиал «Протвино» государственного университета «Дубна»
Секция «Социальные и гуманитарные науки»*

Автор: Тихомиров Михаил Андреевич, студент 3 курса, направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Борисова Татьяна Юрьевна, магистр психолого-педагогического образования, ассистент кафедры общеобразовательных дисциплин филиала «Протвино» государственного университета «Дубна».

Author: Tihomirov Mikhael Andreyevich, 3st year student of the direction "Automation of technological processes and production" of the branch "Protvino" state university "Dubna".

Scientific adviser: Borisova Tatyana Yurievna, Master of Psychological and Pedagogical Education, Assistant of the Department of General Subjects, of the branch "Protvino" state university "Dubna".

Аннотация

Неоконченное произведение Ф. Бэкона «Новая Атлантида» посвящено описанию идеального устройства общества, главенствующее положение в котором занимает наука и научно-технический прогресс. Будучи основоположником новой науки и философом экспериментального знания, Ф. Бэкон художественными средствами рисует картину счастливой жизни людей благодаря процессу постоянного изучения знаний всего мира в изолированном научном учреждении – Доме Соломона, что в конечном итоге приводит к созданию новой авторитарной системы, в которой развитие науки становится самоцелью. Тем самым Бэкон представляет своим читателям самим ответить на вопрос насколько такое возвеличивание науки способно регулировать общественную жизнь во всех её проявлениях.

Abstract

The unfinished work of F. Bacon "New Atlantis" is devoted to the description of the ideal structure of society, the dominant position in which is occupied by science and scientific and technological progress. Being the founder of the new science and the philosopher of experimental knowledge, F. Bacon paints a picture of the happy life of people with artistic means thanks to the process of constantly studying the knowledge of the whole world in an isolated scientific institution - the House of Solomon, which ultimately leads to the creation of a new authoritarian system in which the development of science becomes an end in itself. Thus, Bacon presents to his readers himself to answer the question of how much such an exaltation of science can regulate social life in all its manifestations.

Ключевые слова: утопия Фрэнсиса Бэкона «Новая Атлантида», наука и научно-технический прогресс, сциентистская модель общественного устройства.

Keywords: Francis Bacon's utopia "New Atlantis," science and scientific and technological progress, a scientist model of social structure.

Актуальность работы. Рассмотрение сочинения Ф. Бэкона «Новая Атлантида» о существовании некоего идеального государства, основанного на главенствующем положении науки и избранных ученых – общества мудрецов, в управлении которых сосредоточены и главные природные богатства страны, и важнейшие отрасли промышленности, позволяет раскрыть утопические основания такого существования государства, что является особенно значимым в условиях развития научно-технического могущества в XXI веке.

Проблема исследования. Выделить и показать утопичность, описанных Ф. Бэконом, оснований существования государства.

Объект исследования. История создания и содержание неоконченного произведения Ф. Бэкона «Новая Атлантида».

Предмет исследования. Наука и научно-технический прогресс в модели общественного и государственного устройства в «Новой Атлантиде».

Цель работы: Выделить признаки утопичности модели общественного и государственного устройства, описанного в «Новой Атлантиде».

Задачи работы:

- изучить историю создания и содержание неоконченного произведения Ф.Бэкона;
- выделить те положения Бэкона, в которых предсказывалась будущая структура организации научных институтов и организаций;
- выявить неправомерность утопической модели главенства научного знания в организации общественного устройства.

Гипотеза: несмотря на обоснованность предсказаний Бэкона об определяющей роли научного знания в развитии человеческого сообщества, созданная им модель общественного и государственного устройства обладает ярко выраженными признаками утопии.

Методы исследования: анализ и синтез исследуемого материала; эмпирические методы описания, сравнения и классификации; теоретические методы обобщения.

Атлантида Платона – мифическое государство, погибшее в природной катастрофе. Имя «Атлантида» стало синонимом таинственной, неизвестной, но могущественной земли. Название книги «Новая Атлантида» сразу отсылает читателя к утопии (греч. οὐ – не, нет и τόπος – место, т. е. место, которого нет), т.е. к описанию некоего идеального устройства общества. Но дело в том, что не вопросы социального устройства волновали Бэкона, а место и роль в обществе науки. Ее главенствующее положение - вот что утопично в произведении «Новая Атлантида». Один из основоположников новой науки и философ экспериментального знания Бэкон в этом сочинении пропагандирует научно-технический прогресс. Научные знания по Бэкону – это путеводный светоч, благодаря которому все люди могут жить счастливо.

«Новая Атлантида» начинается с морского путешествия. Случайным образом путешественники попадают в неисследованную часть Тихого океана. Шторм буквально приносит их к загадочному острову Бенсалем. В романе описывается вполне типичная для эпохи Великих географических открытий история. Управлять кораблем в ту эпоху таким образом, чтобы преодолевалась воздушная стихия было невозможно. Проще положиться на волю ветра и Бога, и, возможно, стать первооткрывателем новой земли. Поэтому ситуация с островом Бенсалем вполне заурядная. Но незауряден сам остров и живущие на нем люди. Во-первых, путешественники видят красивый город. Во-вторых, их встречают парламентарии в красивых одеждах. В-третьих, гостям предъявляют пергамент с текстом на нескольких языках: древнееврейском, древнегреческом, латинском и испанском. И, в-четвертых, этот текст запрещает путешественникам высадку на берег! Жители острова не реагируют даже на упоминание о привезенных товарах, им просто все равно! Парламентарии также отказываются принимать у путешественников вознаграждение (со словами «Как? Вторичная плата?»[1]).

Читатель может заметить, что рассказ о странном острове ведется от первого лица, но кто этот «я» - неизвестно. По тексту мы можем только предположить, что рассказчик обладает некоторой властью среди путешественников, т.к. ему подчиняется команда и его же приглашают в дальнейшем для беседы в самом главным лицом острова Бенсалем.

Развитие сюжета идет по всем законам классического утопического произведения. Путешественникам удастся высадиться на острове, они восхищаются увиденным, поражаются тому, как все устроено. Т.е. наконец Бэкон получает возможность изложить свои идеи относительно идеального общественного устройства. Автор уделяет мало внимания непосредственно острову Бенсалем. Мы не знаем даже насколько этот остров велик. Главным в описании Бэкона является возможность конструирования идеального общества, во главе которого стоит наука. Наука, или научное знание, воплощается на практике в организации, которое Бэкон назвал Домом Соломона. Целью Дома является «познание причин и скрытых сил всех вещей и расширение власти человека над природой, покуда все не станет для него возможным»[1].

В беседе с повествователем Отец Дома Соломона подробно описывает «сооружения и приборы», которыми располагают бенсалемцы для занятий наукой. Ф. Бэкон предвидел некоторые будущие открытия и изобретения, из-за чего его называют основателем жанра научной фантастики. В нижеприведенной таблице мы постарались систематизировать научные достижения Дома Соломона, а также показать, какие из этих достижений воплотились в жизнь в наше время.

Фантазия 1627 года	Реальность 2025 года
Водоемы для получения пресной воды из соленой и, наоборот, соленой из пресной.	Несколько способов опреснения воды (дистилляция, химическая очистка, ионный обмен, обратный осмос), минерализация воды.
Бурные потоки и водопады, служащие для получения многих видов движения.	ГЭС (гидроэлектростанции)
«Всякого рода двигатели для увеличения силы ветра, также обращаемой в различного рода движения» [1].	ВЭС (ветряные электростанции)
«Водоемы для получения настоев, насыщенных серой, железом, медью,	Химическая промышленность.

купоросом, где вода приобретает желаемые свойства быстрее, чем в природе» [1].	
«Водоем, называемый райским источником, способствующий сохранению здоровья и продлению жизни». «Комнаты здоровья с целебным воздухом». «Составы для укрепления мускулов, важнейших органов и самой жизненной субстанции» [1].	Комплекс мер для улучшения здоровья (ионизация воздуха, БАДы). Однако «волшебной таблетки» нет.
«Помещения, где искусственно вызывают и показывают различные явления природы: снег, дождь, искусственный дождь из различных твердых тел, гром, молнию» [1].	Компьютерное моделирование.
Зарождение из воздуха живых существ: лягушек, мух и некоторых других.	Отсутствует возможность зарождения жизни из воздуха.
Сады и огороды с почвами, благоприятными для разных растений. Получение новых видов растений, цветение и созревание плодов раньше срока.	Теплицы, удобрения, развитие сельского хозяйства.
Выращивание растений без семян, путем смешения почв, превращение одного растения в другое.	Отсутствует возможность вырастить растение без семян или «превратить» одно растение в другое.
Проведение опытов над животными: испытание ядов и хирургических средств. Сохранение жизнеспособности после удаления некоторых органов, оживление после смерти.	Опыты с участием животных проводятся. Но поддерживать жизнь после смерти невозможно.
Выведение новых пород животных (крупнее, мельче, с другим природным цветом, нравом и строением тела).	Селекция в животноводстве. Но практическая польза от этого другая, например, устойчивость к неблагоприятным условиям окружающей среды.
Выведение из гнили змей и рыб, их дальнейшее преобразование в зверей и птиц.	Отсутствует возможность выведения змей и рыб из гнили, «превращение» в другие виды невозможно.
Дома света, где производятся опыты со всякого рода светом (усиление света, передача на большие расстояния, окрашивание).	Изучение световых явлений и природы света.
Дома звука, где производятся опыты со звуком (новые виды гармоний, воспроизведение всех птиц и зверей, приборы для улучшения слуха, передача звуков по трубам на расстояния).	Акустика – изучение звука как механической волны. Слуховые аппараты.
Дома механики, где изготавливаются машины для всех видов движения (полет птиц, плавание под водой). Часовые механизмы и приборы, основанные на вечном движении.	Многочисленные средства передвижения, в том числе самолеты и подводные лодки. Некоторые часовые механизмы можно считать «вечными», но вечного двигателя не существует.

Мы видим, что многие научные фантазии Бэкона действительно воплотились в жизнь, кроме, разве что, тех проектов, которые были связаны с пониманием жизни и жизненных процессов. Сложно представить, насколько фантастическими идеями выглядели эти описания изобретений в 17-м веке. Верил ли в них читатель? Но что точно было принято с восторгом и верой – это устройство самого Дома Соломона. Бэкон описывает организацию этого учреждения в духе производственного предприятия: со специализацией ученых, с разделением труда на операции и функции. Именно эти мысли Бэкона спровоцировали популярность

произведения. Не фантазии об изобретениях, а фантазии об устройстве Дома Соломона стали предметом достаточно быстрого воплощения в реальность.

Итак, в Доме Соломона различные категории научных работников решают строго определенный круг задач. Часть из них имеет право покидать остров с целью получения научной информации из других стран. Другие работники обрабатывают эту информацию. Для самостоятельных научных изысканий имеется третья группа ученых. Четвертые занимаются изучением и обобщением уже имеющейся научной информации. Пятые разрабатывают методические пособия и т.д. Все это осуществится позже в первых Академиях наук, и в современной развитой структуре научно-исследовательской работы.

Необходимо подчеркнуть, что все описанные Бэконом исследования ведутся экспериментальным путем – ради изобретения чего то полезного. Но при этом деятельность Дома Соломона носит закрытый характер: «...на наших совещаниях мы решаем, какие из наших изобретений должны быть обнародованы, а какие нет. И все мы даем клятвенное обязательство хранить в тайне те, которые решено не обнародовать; хотя из этих последних мы некоторые сообщаем государству, а некоторые – нет»[1]. Можно сделать вывод, что Дом Соломона находится «над» властью: он не подчиняется никому. Также Дому Соломона принадлежит монополия на все открытия, а также монополия на международные отношения. Подобные монополии – это фактически признаки государственной власти. Иными словами, Дом Соломона и есть государственная власть на острове Бенсалема.

Каким образом организовано общественное устройство не совсем понятно, но некоторые выводы мы можем сделать. Во-первых, значимую роль в регуляции жизни общества играет религия. И не какая-то любая религия, а христианство. Божественное откровение снизошло на бенсалемян прямо с неба: однажды жители увидели большой столп света, а потом на его месте «остался лишь небольшой ковчежец, или ларец, кедрового дерева, нимало не подмоченный водою, хотя и плыл по ней»[1]. В ларце оказалась Библия и послание от апостола Варфоломея. Исходя из всего этого, можно сделать вывод, что народ Бенсалема, хоть и является технически развитым обществом, ни в коем случае не сомневается в существовании Бога. Учитывая опыты с жизнью, которые проводит Дом Соломона, и особенно «создание» жизни и разного рода «превращения», либо о подобных экспериментах основная масса жителей острова не знает, либо это объясняется более высокой степенью божественности народа Бенсалема, по сравнению с другими народами.

Во-вторых, на острове особо ценятся большие семьи. Установлен даже специальный уровень, к которому бенсалемянцу нужно стремиться: это 30 потомков в твоей семье. По достижении этого уровня отца семейства чествуют на «Празднике семьи». Данный «Праздник» - пример социального идеала для бенсалемян. «Праздновать его может всякий, кто породит не менее 30 детей и внуков» [1], расходы за праздник берет на себя государство, «Если кто-либо из семьи впал в нищету... изыскиваются способы помочь ему найти средства к существованию» [1] и т.д. Сам по себе «Праздник» представляет собой ритуал, с четко прописанным порядком действий и строгим распределением ролей для участников. С нашей точки зрения, это даже не праздник, а церемония, в ходе которой отца семейства прославляют всеми возможными способами, включая награду от государства (не медаль, но золотая виноградная ветвь). Мать семейства при этом может наблюдать за церемонией, но должна сидеть отдельно, за перегородкой (если она еще жива, по оговорке Бэкона).

В-третьих, можно сделать вывод, что на острове есть и беднота, и даже нищета (хотя путешественники этого не видят, или им этого не показывают). Есть слуги и чиновники, т.е. разные общественные классы и сословия. Это же подтверждает факт наличия в Бенсалеме денег и зарплат, а также разная одежда жителей острова, подчеркнуто богатая у высших слоев населения. При этом члены Дома Соломона – это элита, занимающая особое привилегированное положение в обществе.

В-четвертых, жители острова покорны и соблюдают «образцовый порядок». Все без сомнения принимают на веру не только божественные откровения, но и откровения Дома Соломона (не случайно руководители научной организации носят звание «Отцы Дома»). Это же происходит и с путешественниками. По прибытии на остров лицо, от имени которого ведется повествование, собирает всех своих путников и строго наказывает им «вести себя

смиренно и благочестиво, как и подобает христианам». Однако в дальнейшем надобность в подобных наставлениях отпадает. Более того, узнав, что им предлагают остаться на острове, чуть ли не вся команда готова остаться, за исключением руководителей экспедиции, с которыми и беседуют бенсалемицы. Обычно так и происходит: почти все путешественники, попадая на остров, остаются жить на нем, за все время только несколько человек вернулось в свои страны. Государство полностью обеспечивает оставшихся на острове до конца их дней.

Подобное описание общественного устройства порождает очень много вопросов. Складывается впечатление, что основная задача жителей острова – это обеспечение Бенсалема новыми жителями. Однако остров не бесконечен, зачем постоянно способствовать повышению демографии? Смиренность жителей и постоянное благочестивое поведение (и не только жителей, а также и путешественников) вызывают подозрение, особенно если учесть опыты, проводимые Домом Соломона с едой и питьем. Дом Соломона и жители острова – это пастухи и стадо, причем стадо покорное. Только 12 человек из Дома Соломона могут путешествовать в другие страны, жители же не покидают остров совсем, и, самое главное, нет попыток покинуть остров. Казалось бы, путешественники-ученые выполняют непростую работу по сбору данных в других странах, значит покидать остров должны рядовые сотрудники Дома. Однако Бэкон рассказывает о возвращении Отца Дома из дальних стран, т.е. путешествия – это привилегия элиты.

Сам Отец Дома возвращается торжественно, демонстрируя максимум величия и роскоши. Он въезжает в город на повозке, которая «сделана из кедрового дерева, украшенного позолотой и хрусталем; в передок ее были вделаны плиты из сапфиров в золотой оправе, а позади – такие же плиты из изумрудов перуанского цвета»[1]. Одежда Отца Дома поражает роскошью, чуть менее роскошно одеты и другие участники процессии. Когда европейцы приходят на прием к Отцу Дома, он встречает их «в богатом покое, убранном коврами и драпировками»[1]. Роскошь обстановки и общее почтение к Отцу Дома таковы, что в сознании европейцев он занимает позицию короля.

Таким образом, мы можем сказать, что власть на острове Бенсалем – это власть Дома Соломона. Но по задумке Бэкона должна быть власть знания! Удивительным образом, Бэкон сам себе начинает противоречить. В попытке возвысить науку, он возвышает отдельных представителей людей. Возможно, в этом и суть утопии. Бэкон считает, что суть знания в принесении пользы людям. Если нет пользы, то и наука не нужна. А что такое «польза»? Это то, что делает жизнь человека лучше. Мы видим, что в произведении акцент сделан на богатство и здоровье. Видимо, именно это Бэкон и считал первой задачей науки. Но что будет, если наука окажется сосредоточенной «не в тех» руках? Кому тогда станут доступны богатство и здоровье?

Бэкон считает, что в науке совпадают два человеческих стремления – к знанию и могуществу. Но могущество над природой и могущество над людьми – разные понятия. Если знание превращается в самоцель, то мы обязаны говорить о злоупотреблении знанием. Наука может подсказать, как сделать много золота и драгоценных камней (по Бэкону), но кто будет работать, обеспечивая огромное количество роскоши? А ведь Дом Соломона – самая богатая структура на острове. Какое количество научных открытий остается секретом Дома? А какое количество из этих «секретов» не оглашается, но фактически используется в управлении Бенсалемом? Каким образом происходит выбор преемников в Доме Соломона, при условии, что всеобщее образование Бэконом не предусмотрено?

Эти и многие другие вопросы показывают, что несмотря на верность предсказаний Бэкона о роли научного знания в развитии человеческого сообщества, созданная им модель государственного устройства – это утопия. «Законсервировавшаяся» в своем благополучии и развитии новая Атлантида, как и Платоновская Атлантида, обреченная на гибель в случае столкновения с реальным миром.

Библиографический список

1. Наука и власть в политической философии Ф. Бэкона: от реформаторства к социальной критике // В. В. Мархинин // Известия Саратовского университета. Новая серия. Серия: Философия. Психология. Педагогика. 2023. Т. 23, выпуск 2. С. 144–148.

2. Новая Атлантида // Ф. Бэкон // электронный ресурс <https://libking.ru/books/prose-/prose-classic/334096-frensis-bekon-novaya-atlantida.html> (дата обращения: 20.03.2025).

УДК 004.942

Якимов Е.В.

**РАСЧЁТ И АНАЛИЗ ПРОЧНОСТИ ИЗДЕЛИЙ МАШИНОСТРОЕНИЯ
С ПОМОЩЬЮ CAE-СИСТЕМЫ**
CALCULATION AND ANALYSIS OF THE STRENGTH OF MECHANICAL
ENGINEERING PRODUCTS USING THE CAE SYSTEM

*Филиал «Протвино» государственного университета «Дубна»
Секция «Машиностроение»*

Автор: Якимов Евгений Владимирович, студент 4 курса направления «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Курзуков Геннадий Вячеславович, старший преподаватель кафедры «Автоматизация технологических процессов и производств» филиала «Протвино» государственного университета «Дубна».

Author: Yakimov Evgeny Vladimirovich, 4th-year student of the direction "Automation of technological processes and production" of the branch "Protvino" state university "Dubna".

Scientific supervisor: Kurzukov Gennady Vyacheslavovich, senior lecturer of the department "Automation of technological processes and production" of the branch "Protvino" state university "Dubna".

Аннотация

Статья посвящена автоматизации расчётов прочности изделий машиностроения с использованием системы SolidWorks Simulation. Это позволяет повысить эффективность проектирования и анализа конструкций.

Abstract

The article is devoted to the automation of strength calculations of mechanical engineering products using the SolidWorks Simulation system. This makes it possible to increase the efficiency of design and analysis of structures.

Ключевые слова: CAD-системы, CAE-системы, расчёты на прочность, SolidWorks Simulation, вибрационные нагрузки, ударные воздействия, инженерный анализ.

Keywords: CAD-systems, CAE-systems, strength calculations, SolidWorks Simulation, vibration loads, shock effects, engineering analysis.

Прочность изделий машиностроения имеет важное значение с точки зрения надёжности и безопасности эксплуатации.

В настоящее время, в связи с широким применением систем автоматизированного проектирования (САПР) - CAD-систем (Computer-Aided Design) при проектировании и конструировании изделий, для расчётов конструкций изделий на прочность применяются САПР, относящиеся к классу CAE-систем (Computer-Aided Engineering) - программ и программных пакетов, предназначенных для решения различных инженерных задач.

Расчётная часть таких программ чаще всего основана на численных методах решения дифференциальных уравнений. Сейчас происходит, своего рода, возрождение распространения и использования численных методов компьютерного анализа в инженерной деятельности [1].

Актуальность темы заключается в том, что использование CAE-систем для расчётов конструкций на прочность на стадии проектирования позволяет сократить время на разработку новых изделий и повысить эффективность проектирования.

Проблема использования САПР для прочностных расчётов заключается в установлении достоверности результатов испытаний с помощью таких систем.

В данной статье приводятся результаты исследовательской работы, связанной с использованием CAE-системы SolidWorks Simulation для проверочного расчёта прочности уже готового изделия, заведомо прошедшего лабораторные испытания.

Объект исследования – каркас компрессорной установки электровоза с установленным на нем электродвигателем и компрессором.

Каркас представляет собой металлическую сварную конструкцию, силовые элементы которой изготовлены из швеллера 16У ГОСТ 8240-97. Каркас имеет четыре места крепления к раме тепловоза. Материал деталей каркаса Ст3сп ГОСТ 535-2005 (предел текучести 210 МПа). Масса каркаса – 150 кг.

На каркас установлен асинхронный двигатель НВА-55С массой 385 кг и компрессор ВУЗ,5/10-1450 массой 310 кг.

Предмет исследования – максимальные напряжения в критических точках каркаса, возникающие под действием синусоидальных вибраций или ударных нагрузок, и приводящие к появлению деформаций, которые могут привести к разрушению конструкции.

Цель исследования – получение результатов проверочного расчета прочности каркаса компрессорной установки при воздействии синусоидальной вибрации и одиночного удара в соответствии с группой механического исполнения М25 ГОСТ 30631-99 [2].

Задачи, которые было необходимо решить в ходе выполнения исследования:

- создать 3D-модель компрессорной установки;
- провести испытания каркаса с использованием CAE-системы, а именно:
 - определить резонансные частоты конструкции каркаса;
 - выявить влияние синусоидальных вибраций;
 - исследовать каркас на прочность при ударных нагрузках.

Методы, которые применялись при проведении исследования:

- изучение теоретических положений и нормативной документации;
- моделирование объекта исследования при помощи SolidWorks [3];
- компьютерные эксперименты в SolidWorks Simulation [4];
- сравнение полученных значений механических напряжений с максимально допустимыми значениями для данной конструкции;
- логический метод, позволяющий сделать вывод об определённой достоверности результатов проверочных расчётов с помощью SolidWorks Simulation для изделия, заведомо прошедшего лабораторные испытания, но при отсутствии результатов этих испытаний, а также позволяющий распространить вывод о допустимости применения этой САПР для предварительных расчётов изделий на прочность на стадии проектирования.

Гипотеза исследования: CAE-систему SolidWorks Simulation можно использовать для проверочных и предварительных инженерных расчётов на прочность конструкций машиностроения, с получением достаточно достоверных результатов.

Модель исследуемой установки

Модели каркаса и компрессорной установки, представлены на рисунках 1 и 2. Модели созданы с помощью CAD-системы SolidWorks.

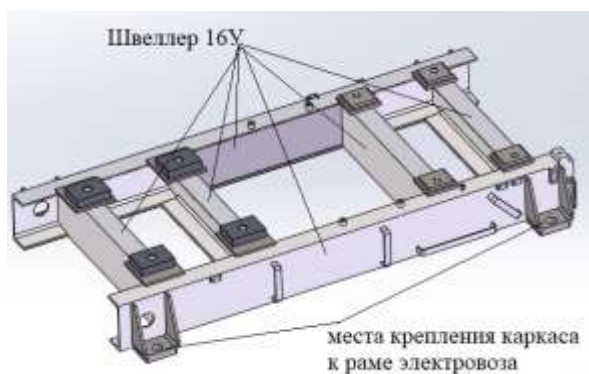


Рис. 1. Модель каркаса

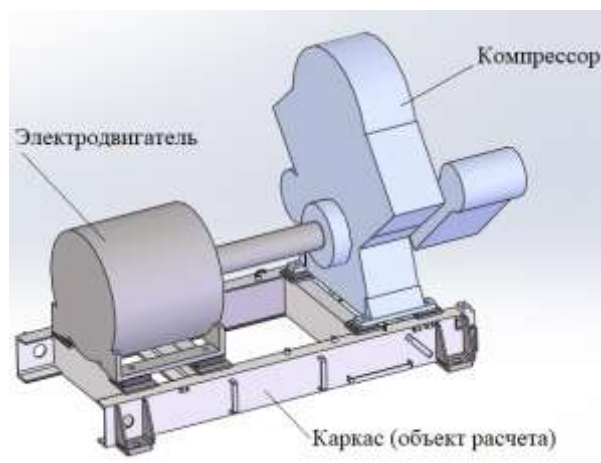


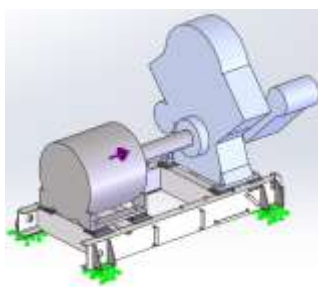
Рис. 2. Модель компрессорной установки

Исследование воздействия синусоидальной вибрации с помощью SolidWorks Simulation

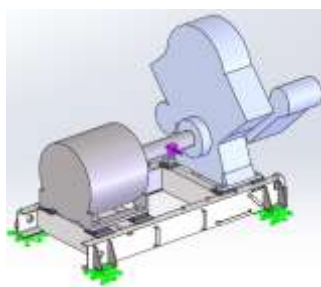
Определение резонансных частот конструкции каркаса

Воздействующий фактор – синусоидальная вибрация в трех направлениях (по отдельности) в диапазоне 0,5-100 Гц с ускорением 10 м/с^2 (1g).

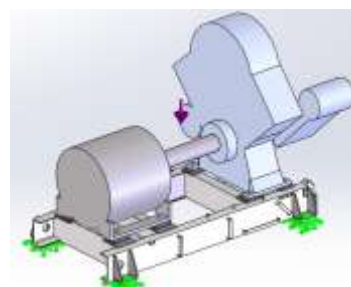
Расчетные схемы для определения резонансных частот при воздействии вибрации в продольном, поперечном и вертикальном направлении показаны на рисунке 3.



а) продольное направление



б) поперечное направление



в) вертикальное направление

Рис. 3. Расчетная схема расчета на воздействие синусоидальной вибрации.

Расчет проводился посредством исследования «Линейная динамика. Гармонические колебания». Параметры расчета и сетка конечных элементов показаны на рисунке 4.

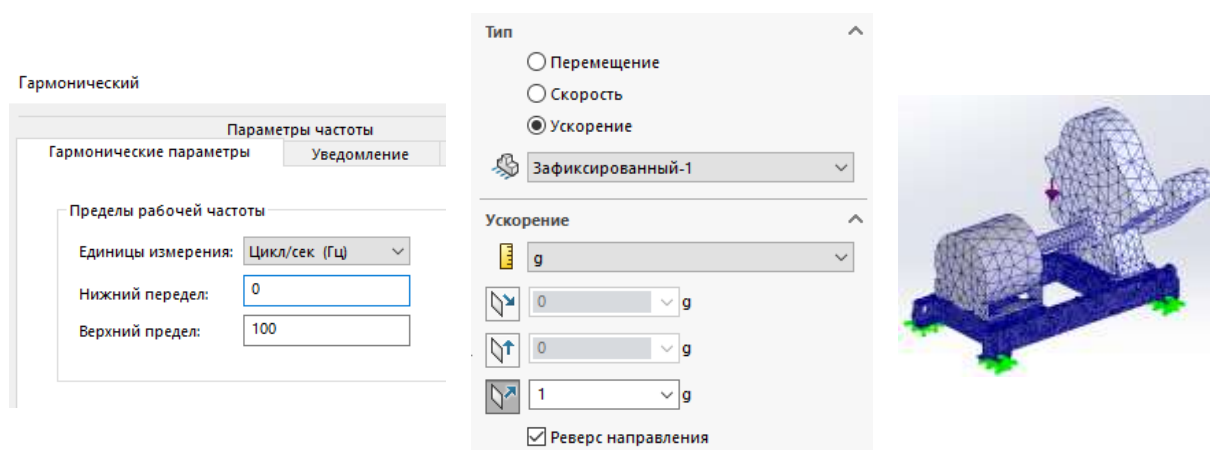
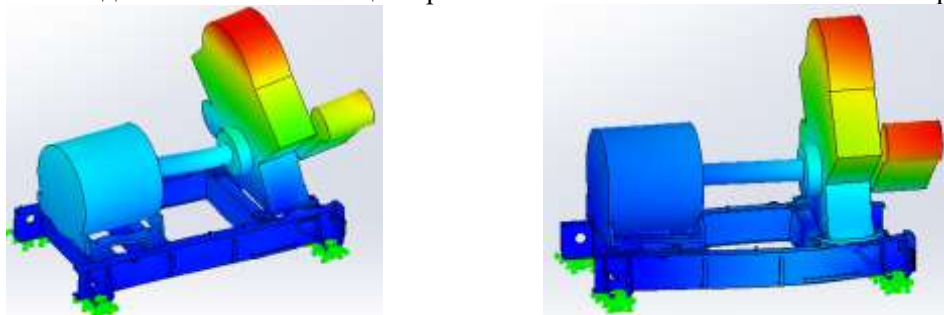


Рис. 4. Параметры расчета при воздействии синусоидальной вибрации, сетка конечных элементов.

В результате расчета определены резонансные частоты компрессорной установки в рассматриваемом диапазоне: 80 и 88 Гц. Формы колебаний показаны в масштабе на рисунке 5.



а) форма колебаний для частоты 80 Гц б) форма колебаний для частоты 88 Гц

Рис. 5. Формы колебаний компрессорной установки на резонансных частотах

Выявление влияния синусоидальных вибраций на конструкцию каркаса

Для каждой резонансной частоты определены максимальные напряжения, возникающие в конструкции каркаса, для каждого из направлений вибрации (рисунки 6-8). Максимальные напряжения указаны для элементов конструкции, за исключением зоны сварных швов, для упрощения расчета.

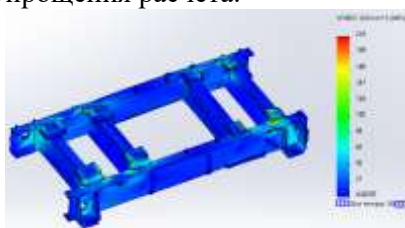


Рис. 6. Эпюры напряжений в продольном направлении вибрации (частота 80 Гц). Максимальные напряжения в конструкции не превышают 100 МПа.

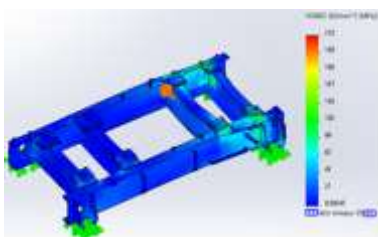


Рис. 7. Эпюры напряжений в поперечном направлении вибрации (частота 88 Гц). Максимальные напряжения в конструкции не превышают 120 МПа.

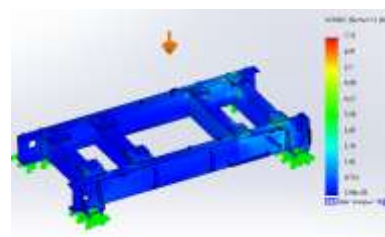


Рис. 8. Эпюры напряжений в вертикальном направлении вибрации (частота 80 Гц). Максимальные напряжения в конструкции не превышают 10 МПа.

Исследование каркаса на прочность при ударных нагрузках с помощью SolidWorks Simulation

Расчет проводился посредством исследования «Линейная динамика. Модальная временная история». Воздействующий фактор – одиночный удар в трех направлениях (по отдельности) длительностью 20 мс и ускорением 30 м/с² (3g).

Расчетные схемы для расчета прочности при воздействии одиночного удара в продольном, поперечном и вертикальном направлении аналогичны изображенным на рисунке 3.

Параметры расчета показаны на рисунке 9.

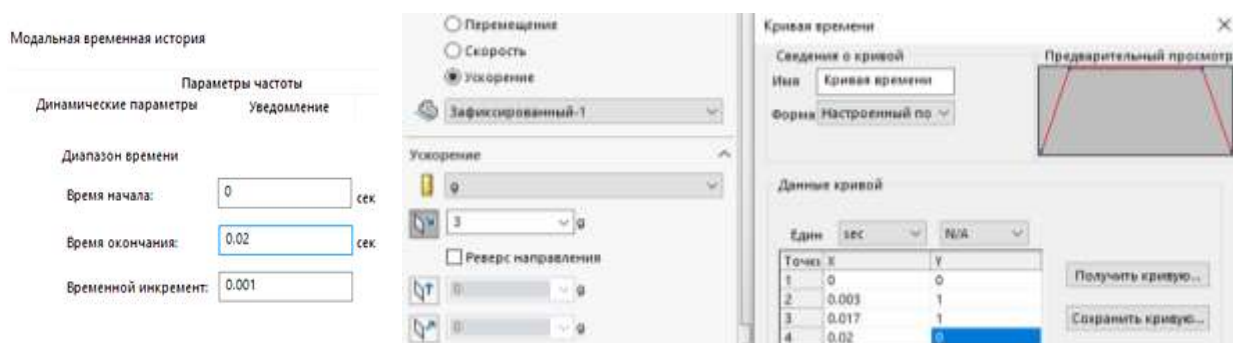


Рис. 9. Параметры расчета при воздействии одиночного удара

В результате расчета определены максимальные напряжения, возникающие в конструкции каркаса, для каждого из направлений одиночного удара. Максимальные напряжения указаны для элементов конструкции, за исключением зоны сварных швов, для упрощения расчета.

Результаты расчета показаны на рисунках 10, 11, 12.

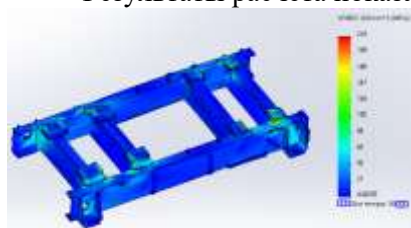


Рис. 10. Продольное направление удара.

Максимальные напряжения в конструкции не превышают 120 МПа.

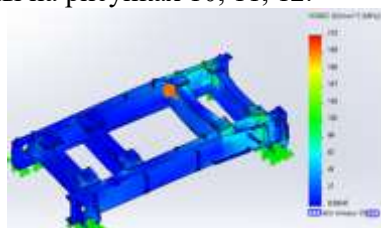


Рис. 11. Поперечное направление удара.

Максимальные напряжения в конструкции не превышают 140 МПа.

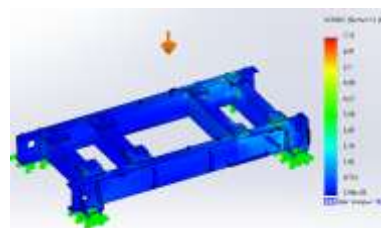


Рис. 12. Вертикальное направление удара.

Максимальные напряжения в конструкции не превышают 20 МПа.

Выводы

Проведённое исследование продемонстрировало эффективность применения CAE-системы SolidWorks Simulation для анализа прочности машиностроительной конструкций. Результаты расчётов подтвердили, что каркас компрессорной установки электровоза способен выдерживать заданные эксплуатационные нагрузки, включая вибрации и ударные воздействия.

Максимальные напряжения в конструкции не превысили 140 МПа, что значительно ниже предела текучести материала (210 МПа). Коэффициент запаса прочности (1,5) свидетельствует о достаточной надёжности каркаса при реальных условиях эксплуатации.

Полученные данные подтверждают гипотезу о возможности использования SolidWorks Simulation для предварительных и проверочных расчётов. Это позволяет сократить время и затраты на проектирование, минимизируя необходимость дорогостоящих физических испытаний. Таким образом, применение CAE-систем открывает новые перспективы для оптимизации инженерных разработок в машиностроении.

Библиографический список

1. SolidWorks Simulation. Инженерный анализ для профессионалов: задачи, методы, рекомендации / А. А. Алямовский. — 2-е изд., эл. — 1 файл pdf : 564 с. — Москва : ДМК Пресс, 2023.
2. МЕГАНОРМ Система нормативных документов: URL-<https://meganorm.ru/Data2/1/4294849/4294849944.htm> (дата обращения 02.04.2025)
3. SOLIDWORKS Web Help: URL -https://help.solidworks.com/2023/RUSSIAN/SolidWorks/sldworks/c_Frequency_Analysis.htm (дата обращения 02.04.2025)
4. SOLIDWORKS – официальный сайт: URL -<https://www.solidworks.com/product/structural-mechanics-engineer> (дата обращения 02.04.2025).

УДК 37.015.3

Яковлева С.И.

ИССЛЕДОВАНИЕ ВЛИЯНИЯ ТЕМПЕРАМЕНТА НА УСПЕШНОЕ ИЗУЧЕНИЕ ИНОСТРАННОГО ЯЗЫКА LEARN TEMPERAMENT TO FURTHER YOUR FOREIGN LANGUAGE LEARNING

*Филиал «Протвино» государственного университета «Дубна»
Секция «Социальные и гуманитарные науки»*

Авторы: Яковлева София Игоревна, студентка 4 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна».

Научный руководитель: Ерицын Сусанна Михайловна, старший преподаватель кафедры общеобразовательных дисциплин филиала «Протвино» государственного университета «Дубна».

Author: Yakovleva Sofia Igorevna, 4th year student of the Computer Science and Engineering program of the branch "Protvino" state University "Dubna".

Scientific adviser: Yeritsyan Susanna Mikhailovna senior lecturer of general educational subjects department of the branch "Protvino" state University "Dubna".

Аннотация

Статья рассматривает психологическую характеристику основных типов темперамента и специфику их влияния на успешность изучения иностранного языка.

Abstract

The article examines the psychological characteristics of the main types of temperament and the specifics of their influence on the success of learning a foreign language.

Ключевые слова: иностранный язык, темперамент, обучающиеся, индивидуально-психологические особенности.

Key words: foreign language, temperament, learners, individual psychological characteristics.

Объект работы - процесс обучения иностранному языку.

Предмет работы: темперамент как психологический аспект изучения иностранных языков.

Цель данной работы — доказать влияние темперамента на успешность изучения иностранного языка.

Для достижения поставленной цели требуется решение следующих задач:
изучить историю появления понятия о темпераменте для формулировки его современного определения.

Охарактеризовать четыре современных типа темперамента и их свойства.

Провести анкетирование среди студентов и проанализировать результаты анкетирования.

Разработать рекомендации учащимся с разными типами темперамента для успешного изучения иностранных языков.

Методы работы: анализ литературы по теме работы, описание, обобщение, теоретический анализ, анкетирование.

Гипотеза: предположим особенности каждого типа темперамента могут существенно влиять на процесс изучения иностранных языков.

Согласно данным Британского совета, английский язык занимает первое место по количеству носителей, на нём общаются более 1, 75 миллиарда человек, что составляет около 20% населения планеты; при этом лишь 375 миллионов человек владеют им как родным языком.

Сложности в освоении иностранных языков объясняются не только разницей в их структуре, но и психологическими факторами, связанными с личностью изучающего. Данная статья посвящена влиянию темперамента на успешность изучения языка.

Обратимся к истории. Первым понятие о темпераменте и его классификации в восточной философии сформулировал Гиппократ, который в IV веке до нашей эры начал исследовать причины поведения людей. Он пришёл к выводу, что причиной этих особенностей является наличие в организме человека определённых «жизненных соков», таких как желтая и черная желчь, лимфа и кровь. Соотношение указанных компонентов, как предполагал Гиппократ, формирует индивидуальные различия в темпераменте.

В начале XX века эту теорию развил И. П. Павлов, который показал зависимость между физиологическими основами темперамента и свойствами высшей нервной деятельности. Мнение Павлова о балансе процессов торможения и возбуждения подтверждает современное определение темперамента.

Темпера́мент (лат. *temperamentum* — «соразмерность», «надлежащее соотношение частей») — это совокупность устойчивых динамических особенностей психических процессов человека: темпа, ритма, интенсивности.

И.П. Павлов различил четыре типа нервной системы:

1. сильный, неуравновешенный (с преобладанием силы процесса возбуждения);
2. сильный, уравновешенный, подвижный;
3. сильный, уравновешенный, инертный;
4. слабый.

Эти общие типы нервной системы лежат в основе четырех современных типов темперамента.

Холерик — порывистый, отдаётся делу со всей страстью, но неуравновешенный, склонный к бурным эмоциональным всплескам и резким сменам настроения;

Сангвиник — энергичный, общительный, сравнительно легко переживающий неудачи и неприятности;

Флегматик — медлительный, спокойный, с устойчивыми стремлениями;

Меланхолик — эмоциональный, чувствительный, легкоранимый, подавленный;

Основными свойствами темперамента являются:

1. скорость возникновения психических процессов и их устойчивость (например, скорость восприятия, длительность сосредоточения внимания);
2. психический темп и ритм (например, продолжительность сосредоточения внимания, быстрота усваивания информации и скорость восприятия);
3. интенсивность психических процессов (например, сила эмоций, активность действий);
4. направленность психической деятельности на какие-либо объекты, независимо от их содержания (например, постоянное стремление человека к контактам с новыми людьми, к новым впечатлениям).

Анализ психологических характеристик различных типов темперамента и их свойств позволяют понять, как они влияют на освоение иностранных языков.

Изучая английский язык в университете в течение четырёх лет, я обратила внимание, что некоторым студентам легче даются устные задания, а другие, наоборот испытывая трудности с общением, легко справляются с письменными упражнениями.

Однкурсникам было предложено принять участие в анкетировании для определения своего темперамента на сайте <https://temperamenttest.org/ru-ru/>. Тест основан на классической методике

Айзенка, но с современными вопросами, которые моделируют стандартные ситуации поведения человека.

Всего было опрошено 32 студента. По результатам тестирования были сделаны выводы и разработана памятка для однокурсников «Ваш темперамент и его влияние на изучение иностранного языка».

Сангвиники (13 студентов), обладая общительностью и обаянием, легко осваивают язык, беспрепятственно вступая в общение. Диалоги, полилоги, инсценировки даются им без труда. Однако сангвиникам следует обратить внимание на недостаток концентрации и склонность к отвлечению, которые могут помешать им в освоении иностранного языка.

Холерики-экстраверты (8 студентов) также хорошо взаимодействуют с окружающими, что способствует их активной работе на уроках. Быстрота реакции и отличная память позволяют успешно изучать язык, хотя неуживчивость и переменчивое настроение могут негативно сказаться на процессе обучения.

Флегматики (7 студентов), наделенные спокойствием и терпеливостью, могут испытывать трудности в устном общении из-за интровертности и недостаточной вербальной памяти. Тем не менее, они успешно справляются с задачами, требующими усердия и терпения, особенно в письменных переводах.

Меланхолики (4 студента), склонные к излишней застенчивости и неуверенности, зачастую испытывают сложности в общении и преодолении языковых барьеров. Их прогресс в изучении языка может быть ограничен, хотя самостоятельная работа является их сильной стороной.

Как правило, холерики и сангвиники не проявляют заинтересованности в рутинной работе, требующей постоянного внимания, например, при написании сочинений. В то время как творческие задания, такие как проекты и презентации, могут значительно повысить их мотивацию. Меланхолики, напротив, успешно выполняют такие задания, в то время как флегматики могут делать это быстрее.

Таким образом, сравнивая результаты тестирования с исследованием, в котором было дано определение понятию «темперамент» в контексте истории его появления, охарактеризованы четыре современных типа темперамента и их свойства, можно сделать вывод, что особенности каждого типа темперамента могут существенно влиять на процесс изучения иностранных языков, что подтверждает выдвинутую гипотезу.

Результаты настоящего исследования имеют практическую значимость, поскольку вышеупомянутые приемы, задания, исследования помогут изучающим иностранный язык, а также преподавателям иностранного языка использовать сильные стороны студентов и компенсировать их недостатки для достижения успеха в изучении языка.

Библиографический список

1. Балан Д.В. «Связь между темпераментом и трудностями в изучении иностранного языка //Фестиваль педагогических идей «Открытый урок». 2021. URL: <http://festival.1september.ru> (дата обращения: 20.04.2025).
2. Методика обучения иностранному языку: учебник и практикум для вузов / О. И. Трубицина [и др.]; под редакцией О. И. Трубициной. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2024. — 457 с. — (Высшее образование). — ISBN 978-5-534-19070-0. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/555883> (дата обращения: 20.04.2025).
3. Хуторная, М. Л. Психология : учебное пособие : [16+] / М. Л. Хуторная ; науч. ред. Г. А. Быковская ; Воронежский государственный университет инженерных технологий. — Воронеж : Воронежский государственный университет инженерных технологий, 2020. — 57 с.:

Научное электронное издание

Материалы 1-го этапа 30-й ежегодной научно-практической конференции
студентов, аспирантов и молодых специалистов
государственного университета «Дубна», филиал «Протвино»
(г. Протвино, 8-10 апреля 2025 г.)

Изготовлено по адресу: 142281, Московская область, г. Протвино, ул. Северный проезд, д. 9