

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО  
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ  
РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
ИМЕНИ В.Ф. УТКИНА

**ИНФОРМАЦИОННЫЕ  
ТЕХНОЛОГИИ**

Межвузовский сборник научных трудов

Рязань 2022

УДК 004

Информационные технологии, межвузовский сборник научных трудов. – Рязань: ИП Коняхин А.В. (Book Jet), 2022. – 388 с.

**ISBN 978-5-907568-17-4**

Публикуются статьи о проблемах использования информационных технологий в науке, промышленности и образовании.

Сборник рассчитан на научных сотрудников, преподавателей, аспирантов и студентов высших учебных заведений.

Редакционная коллегия:

д-р техн. наук, проф. В.П. Корячко (ответственный редактор), д-р техн. наук, проф. В.А. Минаев (МУ МВД России им. В.Я. Кикотя), д-р техн. наук, проф. А.О. Фаддеев (Московский государственный технический университета имени Н.Э. Баумана), д-р техн. наук, проф. А.Д. Иванников (Институт проблем проектирования в микроэлектронике Российской академии наук), д-р техн. наук, проф. С.В. Скворцов (РГРТУ), д-р техн. наук, проф. Д.А. Перепелкин (РГРТУ), канд. техн. наук, доц. А.Н. Сапрыкин (ответственный секретарь).

Рецензенты:

д-р техн. наук, проф. А.И. Таганов (Рязанский государственный радиотехнический университет имени В.Ф. Уткина), д-р техн. наук, проф. А.П. Карпенко (Московский государственный технический университета имени Н.Э. Баумана).

**ISBN 978-5-907568-17-4**

©Рязанский государственный радиотехнический университет имени В.Ф. Уткина, 2022  
©ИП Коняхин А.В. (Book Jet), 2022

## СОДЕРЖАНИЕ

|   |    |
|---|----|
| <b>Агапов А.В.</b><br>Анализ посещаемости интернет-ресурсов .....   | 11 |
| <b>Аристархова М.С.</b><br>Исследование методов оптимизации и принятия решений .....  | 16 |
| <b>Артамонов А.Ю., Заикин С.Ю., Пасынков М.И., Ужегова М.Н.</b><br>Применение технологии машинного обучения для обнаружения волейбольного мяча на изображении ..... | 20 |
| <b>Башкин Ф.И.</b><br>Проектирование высокоплотных соединений .....   | 24 |
| <b>Беляев А.В., Дмитриева Т.А.</b><br>Исследование оптимизации программного кода и ее влияния на объем кода на примере решения задач на языке 1С .....              | 27 |
| <b>Бережнов А.Д.</b><br>Многокритериальные методы принятия решений .....  | 34 |
| <b>Боброва Д.В.</b><br>Смоли, применяемые в радиоэлектронной промышленности .....   | 37 |
| <b>Большаков А.С.</b><br>Архитектура и особенности разработки веб-API с помощью технологии ASP.NET .....  | 41 |
| <b>Большаков А.С.</b><br>Особенности и структура беспроводной передачи данных протокола ZIGBEE .....  | 44 |
| <b>Виклов А.В.</b><br>Использование В-деревьев в файловых системах .....  | 47 |
| <b>Власов П.А., Гагаринский В.Ю.</b><br>Повышение надежности передачи текстовых данных по каналу связи .....  | 51 |

|  |     |
|--|-----|
| <b><i>Вьюгина А.А., Баранова С.Н.</i></b><br>Исследование эффективности применения кодирования битовых плоскостей изображения при передаче по каналам связи..... | 55  |
| <b><i>Гагаринаская М.С., Васюкова Ю.А.</i></b><br>Увеличение качества передачи звука по каналу связи .....   | 61  |
| <b><i>Гальцова Д.В.</i></b><br>Алгоритмы, использующие деформацию трасс соединений.....  | 65  |
| <b><i>Гальцова Д.В.</i></b><br>Алгоритм трассировки печатных соединений на основе представления о каналах .....  | 69  |
| <b><i>Горин В.С.</i></b><br>Задача распределения печатных проводников в ограниченном пространстве.....   | 73  |
| <b><i>Гусева А.С.</i></b><br>Выбор конструктивных параметров активного элемента для аппарата лазерного облучения крови .....                                     | 79  |
| <b><i>Гуцин К.А.</i></b><br>Развитие современного мира на примере 3D моделирования .....   | 83  |
| <b><i>Даньков Д.С.</i></b><br>Обзор современных ПЛИС и САПР фирмы XILINX .....   | 87  |
| <b><i>Дубченко М.С.</i></b><br>Подсистема дистанционного банковского обслуживания «Интернет- Клиент».....  | 90  |
| <b><i>Дьяков М.С., Скворцов С.В.</i></b><br>Применение генетического алгоритма для решения транспортной задачи .....   | 96  |
| <b><i>Ежов Д.А.</i></b><br>Структура и принцип работы шасси многофункционального робота .....  | 101 |

|   |     |
|---|-----|
| <b>Ермилов В.В.</b><br>Использование языка Python как средства автоматизации отчетности в учебе .....                                     | 104 |
| <b>Ефимов Д.М., Пузиков А.Д., Игошина М.С., Сапрыкин А.Н.</b><br>Принцип проектирования soft-процессоров на основе современных ПЛИС ..... | 107 |
| <b>Захаров И.С.</b><br>Сравнительный анализ алгоритмов для решения задачи трассировки проводного монтажа .....                            | 115 |
| <b>Зинова В.О., Сергеев М.А.</b><br>Исследование и реализация алгоритма обратного размещения .  | 118 |
| <b>Иванчикова М.А., Шмаков А.Ю.</b><br>Механизм обнаружения и распознавания лиц на изображениях с применением нейронных сетей.....        | 123 |
| <b>Игошина М.С., Сапрыкин А.Н.</b><br>Особенность проектирования двухтактного Т-триггера в мультисреде Intel Quartus Prime.....           | 128 |
| <b>Калинкина А.А.</b><br>Структура и принцип работы «Умной розетки» .....   | 134 |
| <b>Калинкина А.А.</b><br>Обзор и исследование микроконтроллеров для системы управления «Умный дом».....                                   | 138 |
| <b>Калинкин А.А.</b><br>Сравнительный обзор САПР для моделирования Bluetooth-колонки.....   | 141 |
| <b>Калинкин А.А.</b><br>Методы устранения явления «земляная петля» при разработке Bluetooth-колонки .....                                 | 145 |
| <b>Кашцев А.Н.</b><br>Методы вычисления тригонометрических функций .....  | 149 |

|   |     |
|---|-----|
| <b>Капустина Е.В.</b><br>Методы контроля дефектов в электронных средствах .....   | 154 |
| <b>Киреев П.С., Скворцов С.В.</b><br>Архитектура археологической автоматизированной информаци-<br>онной системы.....                            | 158 |
| <b>Кожиков В.Ю.</b><br>Устройство горизонтирования .....  | 162 |
| <b>Корячко В.П., Епифанов А.С.</b><br>Принятие решений в интеллектуальных системах .....  | 165 |
| <b>Косорукова Е.В., Александрова П.О.</b><br>Разработка 3D-моделей средствами САПР.....   | 170 |
| <b>Кочepasов А.С.</b><br>Программные решения для учета рабочего времени сотрудников<br>вне помещений .....                                      | 174 |
| <b>Кошелев А.Д., Игошина М.С.</b><br>Особенности проектирования модулятора для усиления СВЧ-<br>сигнала радиолокационной бортовой системы ..... | 183 |
| <b>Кругман А.Г.</b><br>Разработка приложения для управления умным домом .....   | 187 |
| <b>Крыгина М.К.</b><br>Методы обработки сейсмоакустической информации морского<br>геофизического комплекса.....                                 | 190 |
| <b>Кулакова И.В.</b><br>Конструктивная блочная геометрия при построении примитивов<br>в программе FreeCAD .....                                 | 194 |
| <b>Кулева П.С.</b><br>Технология изготовления цветной газоразрядной индикаторной<br>панели.....   | 199 |
| <b>Медведева А.Ю.</b><br>Использование CRM в розничной аптечной сети .....  | 204 |

|   |     |
|---|-----|
| <b>Меньшина В.А.</b><br>Перспективы развития усилителей мощности СВЧ диапазона для радиосистем передачи информации .....  | 209 |
| <b>Михеев В.А.</b><br>Параметры, определяющие оптимальный микроклимат теплицы.....  | 212 |
| <b>Михеев В.А.</b><br>Структура и принцип работы устройства контроля микроклимата теплицы .....                           | 216 |
| <b>Муранов В.П.</b><br>Моделирование траектории частицы, пролетающей через поле квадрупольного масс-спектрометра.....     | 220 |
| <b>Нгуен Х.Х.</b><br>Анализ методов трассировки печатных плат.....  | 226 |
| <b>Нгуен Х.Х.</b><br>Реализация волнового алгоритма для определения кратчайшего пути на плоскости .....                   | 230 |
| <b>Немцев А.Е.</b><br>Силовые алгоритмы.....  | 234 |
| <b>Нижутин И.В.</b><br>Сравнительная характеристика популярных систем автоматизированного проектирования .....            | 238 |
| <b>Нижутин И.В.</b><br>Сравнительная характеристика систем автоматизированного проектирования электронных устройств ..... | 243 |
| <b>Новичков В.О.</b><br>Инфологическое проектирование реляционной базы данных....   | 249 |
| <b>Осин Д.Ю.</b><br>САМ программы в производственном процессе .....   | 255 |

|   |     |
|---|-----|
| <b>Облаков Н.Е.</b><br>Применение методов машинного обучения в оценке кредитоспособности заемщика.....  | 259 |
| <b>Перепелкин Д.А., Анисимов К.В.</b><br>Обзор программного обеспечения для моделирования систем нечеткого логического вывода .....                       | 265 |
| <b>Перепелкин Д.А., Потапкина Н.С.</b><br>Особенности проектирования устройств интернета вещей на основе технологии программно-конфигурируемых сетей..... | 275 |
| <b>Першин А.Н.</b><br>Системные вызовы в ОС Linux .....   | 281 |
| <b>Пещеров К.Ю.</b><br>Когенерационные солнечные электротепловые станции .....  | 285 |
| <b>Пырков А.В.</b><br>Расчет эффективной площади рассеивания .....  | 293 |
| <b>Радучич Д.В.</b><br>Выбор конструкции устройства приема, передачи и преобразования телеметрической информации.....                                     | 298 |
| <b>Салынов П.А., Скворцов С.В.</b><br>Структурно-семантическая кластеризация веб-страниц для извлечения данных из сети интернет.....                      | 302 |
| <b>Сапрыкина А.О., Игошина М.С.</b><br>Сравнительный анализ систем дистанционного обучения.....   | 306 |
| <b>Сапрыкин А.Н., Юрков Н.Ю.</b><br>Алгоритмизация в процессе разработки автоматизированной системы общени с пользователями.....                          | 315 |
| <b>Сафронова А.А.</b><br>Математическая обработка электрофизиологической информации на ЭВМ.....   | 319 |

|  |     |
|--|-----|
| <b>Селиверстов К.В.</b><br>Детектор границ градиентного типа с векторной маской .....  | 324 |
| <b>Савинкин С.С., Аникин А.А.</b><br>Основы BIM-проектирования .....   | 329 |
| <b>Синякова О.И.</b><br>Применение автоматической оптической инспекции ALD625 при контроле качества паяных соединений .....  | 336 |
| <b>Трубицын А.А., Кочергин Э.Г.</b><br>Применение системы компьютерной алгебры «MathCAD» для решения задачи Дирихле методом конечных элементов .....                         | 340 |
| <b>Турнаев С.С.</b><br>Искусственная нейронная сеть в машинном обучении .....  | 349 |
| <b>Украсова Д.А.</b><br>Аппроксимация сложных геометрических поверхностей развёртывающимися поверхностями .....  | 353 |
| <b>Фирсов Н.О.</b><br>Трассировка многослойных печатных плат алгоритмом Хейса .....  | 357 |
| <b>Фокин И.В.</b><br>Система штрихкодирования на производство .....  | 361 |
| <b>Фошина М.Р.</b><br>Использование среды MATLAB для обработки термограмм с целью выявления утечек тепловых сетей .....  | 364 |
| <b>Цедилина Т.А.</b><br>Конструктивное описание устройства, предназначенного для приема телеметрической информации, преобразования и выдачи её на средства регистрации ..... | 368 |
| <b>Чан В.А.</b><br>Алгоритм спектрального анализа сигналов лидарных отражений на фоне кусочно-стационарных помех .....   | 372 |

|   |     |
|---|-----|
| <b>Шаронов Д.В.</b><br>Обучение нейронной сети для генерации сайта .....  | 376 |
| <b>Швечкова О.Г., Артёмов Я.А.</b><br>Выбор программной реализации криптографического алгоритма<br>электронной цифровой подписи DSA.....              | 380 |
| <b>Яшкова В.В., Игошина М.С.</b><br>Особенности проектирования устройства сопряжения специали-<br>зированных цифровых вычислительных машин с ПК ..... | 384 |

УДК 004.77

**А.В. АГАПОВ**Рязанский государственный радиотехнический университет  
имени В. Ф. Уткина**АНАЛИЗ ПОСЕЩАЕМОСТИ ИНТЕРНЕТ-РЕСУРСОВ***В статье рассматриваются цели и способы анализа посещаемости Интернет-ресурсов, а также преимущества инструментов статистики.*

В настоящее время Интернет представляет собой различные ресурсы, содержащее в себе все, что может интересовать современного человека и активно используется для рекламы, продажи товаров и услуг.

Во время пандемии коронавируса COVID-19 и по сей день, стали крайне популярны сервисы, позволяющие возместить привычный образ жизни. К примеру службы доставки (Яндекс.Еда), интернет-магазины (OZON, Яндекс.Маркет), онлайн-кинотеатры (Кинопоиск), программы для организации видеоконференций (ZOOM) и тому подобные.

Начало появляться множество SEO-специалистов, работа которых заключается в привлечении посетителей на сайт путём оптимизации содержимого под запросы пользователей.

SEO продвижение (поисковая оптимизация сайта) представляет собой комплекс мероприятий, позволяющий повысить видимость и позиции сайта в поисковых системах для дальнейшей монетизации ресурса.

Одной из основных характеристик сайта является его посещаемость, т.е. количество человек, заходящих на него за определенный промежуток времени.

Уровень посещаемости сайта зависит от следующих факторов:

- наличия технических ошибок в коде сайта, оказывающих влияние на корректность работы и скорость загрузки;
- наличия полезного и уникального контента, привлекающего внимание целевой аудитории;
- размещения вечных обратных ссылок на авторитетных веб-площадках;
- использования контекстной рекламы, а также инструментов маркетинга;
- создания удобного пользовательского интерфейса, навигации и адаптивного дизайна сайта;

- предоставления комфортных условий для обратной связи с представителями целевой аудитории;
- применение современных механизмов и рекомендованных поисковиками инструментов SEO и др.

На каждый сайт можно установить специальный счетчик, который будет вести статистику посещений веб-сайта. Часто используются комплексные меры для частного продвижения сайтов.

Счётчик - это сервис, предназначенный измерения посещаемости сайтов, записывающий информацию о посетителях и их действиях. Он включает в себя код, размещающийся на страницах сайтов для сбора данных и движок, подсчитывающий полученную информацию и обрабатывающий её в виде статистического отчёта.

Счётчик является рейтингом, сортирующим сайты по посещаемости, а также используется в качестве каталога тематически структурированных рейтингов.

Очень часто счётчики также используются как аудит собственной статистики для рекламодателей.

Аудит сайта выступает как основная и незаменимая стадия SEO-оптимизации, включающая значительный перечень мероприятий, с использованием которых эксперты подвергают проверке технические и функциональные характеристики сайта.

Статистика Интернет-ресурса может потребоваться маркетологам при покупке рекламного места, т.к. важно до старта рекламной кампании проверить статистику посещаемости площадки, где планируется опубликовать рекламу. Это помогает избежать расходования бюджета на нецелевых посетителей.

Также это необходимо для анализа уровня конкурентоспособности. При наличии информации о посещаемости сайтов-конкурентов, можно проанализировать рынок, разработать план продвижения собственного сайта, получить новые источники увеличения трафика, а также успешно выделяться среди конкурентов.

Для проверки посещаемости существуют различные инструменты. Каждый из способов отличается своими особенностями, предоставляемой информацией.

Одним из таких инструментов является Яндекс.Метрика — аналитическая система от поисковой системы Яндекс. Сервис позволяет ознакомиться с пробной метрикой, чтобы изучить весь функционал. Аналитическая система показывает следующую информацию: количество посетителей за определенный период времени, источник трафика, типы устройств с которых выполнялись визиты, возраст, пол посети-

телей, показатель отказов, поисковые запросы и так далее. Виды сводки одной из тестовых метрик представлены на рисунках 1 и 2.

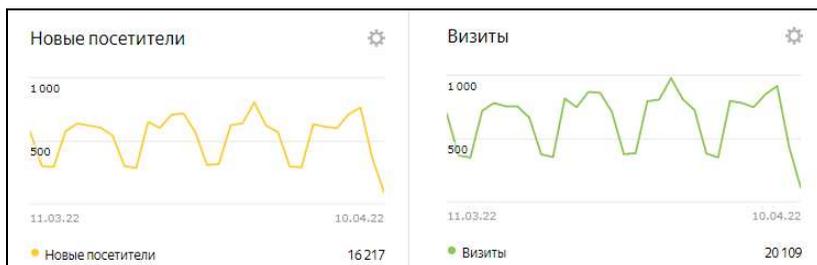


Рисунок 1 – Виджеты «Новые посетители» и «Визиты»

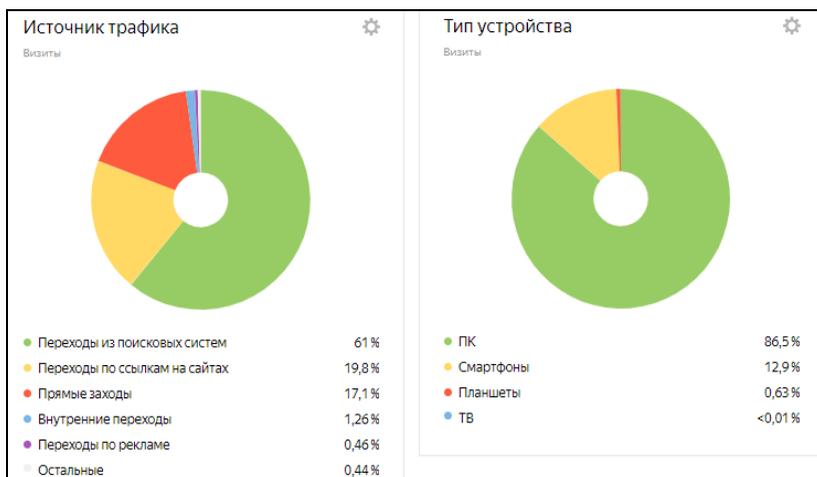


Рисунок 2 – Виджеты «Источник трафика» и «Тип устройства»

Главным преимуществом Яндекс.Метрики является удобный интерфейс и всплывающие подсказки для новичков, а также функция «Вебвизор», анализирующая поведение и действия посетителей на сайте.

Другой популярной системой аналитики выступает Google Analytics. Она составляет наглядное представление о посещаемости пользователей и вероятном доходе ресурса. В отличие от Яндекс.Метрики можно выделить следующие преимущества: объёмный набор фильтров и расширений для обнаружения необходимых сведений, возможность просмотреть подробную информацию о заказах и клиентах.

Значительным недостатком является отсутствие тестовых метрик для ознакомительной оценки функционала системной аналитики.

Различные онлайн-сервисы являются дополнительным источником информации о сайте. Их основным преимуществом является возможность получить конкретные цифры визитов на сайте конкурента без доступа к счетчикам, однако не обладают точными данными.

Одним из популярных сервисов является SimilarWeb. Он дает возможность посмотреть статистику в мобильной версии, а также узнать количество посещений, среднее время визита, показатель отказов, регион трафика, ключи. Присутствует возможность экспорта отчетности в Excel. К сожалению, большинство функций доступно лишь при платной подписке.

Аналогичным сервисом является SEMRush, включающий в себя русскоязычную версию и обладающий таким же функционалом, что и SimilarWeb.

Популярным для проверки статистики в России является бесплатный сервис 2IP.RU, но большинство инструментов при анализе опираются на информацию сторонних счетчиков (например, Яндекс.Метрика).

Если точная статистика по домену не нужна или нет возможности оформлять платный доступ к расширенному функционалу площадок, можно воспользоваться альтернативными методами оценки посещаемости.

Отличной платформой для проверки посещаемости ресурсов, на мой взгляд, является Alexa от компании Amazon. Она собирает рейтинг на основе статистических данных (соотношение уникальных пользователей, объем просмотра страниц), ключевые слова конкурентов и запросы.

Сбор информации производится благодаря плагину Toolbar, отображающему статистику ресурсов, количество обратных ссылок, определяющему позиции во встроенном рейтинге Alexa. Специальный код Certify, который можно добавить на проект, присваивает домену сертифицированный статус.

В качестве примера проведем конкурентный анализ сайта компьютерного супермаркета НИКС с помощью платформы Alexa. Результат представлен на рисунке 3.

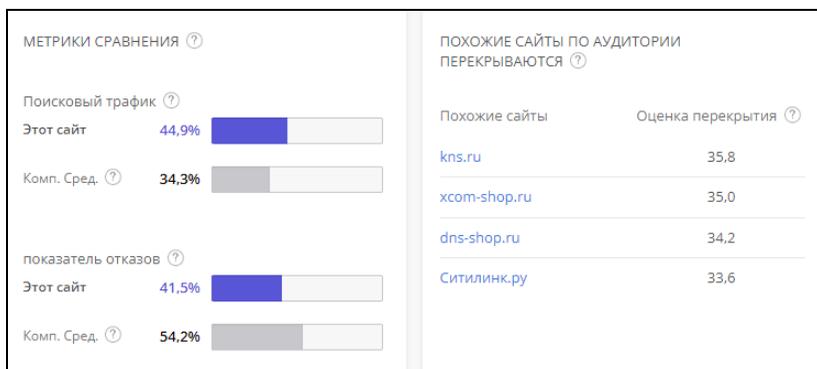


Рисунок 3 – Конкурентный анализ

Процент переходов на сайт из органического поиска равен 44,9%. Это очень хороший результат. Основным конкурентом является [kns.ru](http://kns.ru), т.к. у него самый высокий показатель перекрытия аудитории, который рассчитывается на основе анализа общих посетителей и/или ключевых слов поиска.

На основании вышесказанного, можно сделать вывод, что показатели посещаемости Интернет-ресурса, отображающие количество посетителей перешедших на сайт за определенный промежуток времени, позволяют оценить эффективность проекта и отдельных подразделов. Такие данные необходимо знать не только о собственном ресурсе, так и о ресурсах конкурентов.

Стоит отметить, что правильнее будет прибегать к комплексному подходу анализа посещаемости ресурса. Только таким образом получится извлечь более достоверную информацию о трафике и добиться положительных результатов.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Проверка статистики сайта [Электронный ресурс]. – Режим доступа: <https://amdg.ru/blog/proverka-poseshchaemosti-sayta/>.
2. Авианаш Кошик. Веб-аналитика. Анализ информации о посетителях веб-сайтов. М.: Вильямс, 2009. 464 с.
3. Яндекс.Метрика: инструкция по веб-аналитике – Бесплатное электронное издание. – Коллектив авторов под редакцией Екатерины Илюшкиной и Александра Косолюбова. – ООО «Ингейт Реклама», 2015. 49 с.

УДК 004.048

**М.С. АРИСТАРХОВА**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**ИССЛЕДОВАНИЕ МЕТОДОВ ОПТИМИЗАЦИИ И  
ПРИНЯТИЯ РЕШЕНИЙ**

*В статье рассматривается подход к принятию решений и оптимизации с помощью методов порогов несравнимости. Отмечаются положительные черты этого подхода, связанные с введением отношения несравнимости между альтернативами. Приведен пример использования данного метода на практике.*

Принятие эффективных решений – одна из наиболее актуальных проблем современности. Для оценки вариантов и выбора самого оптимального требуется обоснованный подход и специальные расчеты.

Методы порогов несравнимости ориентированы на решение задач с уже данными многокритериальными альтернативами, они не определяют количественно показатель свойства каждой из альтернатив, а только ставят условие преимущества одной кандидатуры над другой.

Группа методов порогов несравнимости характеризуется оригинальным подходом к сравнению альтернатив. Связь между любой парой альтернатив определяется последовательностью бинарных отношений.

Бинарные отношения между альтернативами строятся следующим образом: каждому из имеющихся критериев с числовой шкалой, ставится в соответствие некоторое целое число  $p$ , которое характеризует важность данного критерия.

Выдвигается гипотеза о превосходстве альтернативы  $a$  над альтернативой  $b$ . Множество  $I$ , состоящее из  $N$  критериев, разбивается на три подмножества:

- 1)  $I^+(a, b)$  – подмножество критериев, по которым  $a$  предпочтительнее  $b$ ;
- 2)  $I^-(a, b)$  – подмножество критериев, по которым  $a$  равноценно  $b$ ;
- 3)  $I^-(a, b)$  – подмножество критериев, по которым  $b$  предпочтительнее  $a$ ;

Далее формируется индекс согласия с нашим предположением о превосходстве альтернативы  $a$  над альтернативой  $b$ . Этот индекс определяется как:

$$C_{ab} = \frac{\sum_{i \in I^+, I^-} p_i}{\sum_{i=1}^n p_i}$$

Где  $\sum_{i \in I^+, I^-} p_i$  – отношение суммы весов критериев подмножеств  $I^+$  и  $I^-$ , а  $\sum_{i=1}^n p_i$  – общая сумма весов.

Индекс несогласия с предположением о превосходстве альтернативы  $a$  над альтернативой  $b$  определяется следующим образом: для критериев подмножества  $I^-(a, b)$  находятся разности оценок альтернатив  $b$  и  $a$ . Чтобы учесть возможную разницу длин критериев, разность оценок  $b$  и  $a$  относится к длине наибольшей шкалы:

$$D_{ab} = \max_{i \in I^-} \frac{I_b^i - I_a^i}{L_i}$$

$I_b^i - I_a^i$  – оценки альтернатив  $b$  и  $a$

$L_i$  – длина числовой шкалы по  $i$ -тому критерию

Свойства индекса согласия:

1)  $0 \leq C_{ab} \leq 1$

2)  $C_{ab} = 1$ , если подмножество  $I^\pm$  пусто;

3)  $C_{ab}$  сохраняет значение при замене одного критерия на несколько других критериев с тем же общим весом;

Свойства индекса несогласия:

1)  $0 \leq D_{ab} \leq 1$

2)  $D_{ab}$  сохраняет значение при введении более детальной шкалы по  $i$ -му критерию;

Основные этапы метода:

1) На основании заданных оценок двух альтернатив подсчитываются значения двух индексов: согласия и несогласия.

2) Задаются уровни согласия и несогласия, с которыми в дальнейшем сравниваются подсчитанные индексы для каждой пары альтернатив. Если индекс согласия выше заданного уровня, а индекс несогласия – ниже, то одна из альтернатив считается превосходить другую. В противном случае альтернативы являются несравнимыми.

3) Из множества альтернатив удаляются доминируемые. Оставшиеся образуют первую ячейку.

4) Вводятся более «малые» значения уровней согласия и несогласия (меньший по значению уровень согласия и больший уровень несогласия), при которых выделяются ячейки с меньшим количеством альтернатив.

5) В последнюю ячейку входят лучшие альтернативы.

Рассмотрим применение метода порога несравнимости на практике.

Постановка задачи:

Пусть некоторая фирма подбирает на работу программиста с целью создания сайтов. Для данной задачи наиболее приоритетными будут считаться кандидаты знающие языки программирования HTML/CSS, соответственно данный критерий будет иметь наибольшую значимость. На вакансию откликнулись 4 кандидата (P1, P2, P3, P4), которые прошли тестирование. Необходимо определить наилучшего кандидата.

Решение задачи:

Для решения этой задачи нужно сформулировать необходимые критерии и их важность по отношению друг к другу, значение которого в дальнейшем будем называть весом критерия. Предположим, что фирма сформулировала свой список критерий отбора кандидатов на должность программиста следующим образом: знание PHP/MySQL, HTML/CSS, C/C++, JavaScript. Пусть таблица оценок выглядит следующим образом:

Таблица 1 – Оценки кандидатов по критериям и их важность

| Критерии оценивания | Кандидаты на вакансию |    |    |    | Вес критерия |
|---------------------|-----------------------|----|----|----|--------------|
|                     | P1                    | P2 | P3 | P4 |              |
| PHP/MySQL           | 8                     | 10 | 7  | 6  | 9            |
| HTML/CSS            | 8                     | 9  | 7  | 10 | 10           |
| C/C++               | 3                     | 1  | 5  | 10 | 8            |
| JavaScript          | 10                    | 9  | 10 | 6  | 8            |

Для решения данной задачи нам необходимо воспользоваться методом порогов несравнимости:

1) Составить таблицу критериев и альтернатив, элементы которой являются оценками по критериям для альтернатив, а также определить вес каждого критерия;

2) На основе оценок построить матрицы согласия и несогласия;

3) Установить пороги, при которых у нас однозначно выстроится иерархия альтернатив;

Строим матрицу согласия, рассчитывая для каждой альтернативы значение согласия следующим образом:

$$C_{ab} = \frac{\sum_{i \in I^+, j = P_i} p_i}{\sum_{i=1}^n p_i}$$

Таблица 2 – Матрица согласия

|    | P1            | P2            | P3            | P4          |
|----|---------------|---------------|---------------|-------------|
| P1 |               | 13/35 = 0,371 | 26/35 = 0,743 | 18/35=0,514 |
| P2 | 19/35 = 0,543 |               | 19/35 = 0,543 | 19/35=0,543 |
| P3 | 15/35 = 0,429 | 15/35 = 0,429 |               | 17/35=0,486 |
| P4 | 20/35 = 0,571 | 20/35 = 0,571 | 20/35 = 0,571 |             |

Строим матрицу несогласия, рассчитывая для каждой альтернативы значение несогласия следующим образом:

$$D_{ab} = \max_{i \in I^a} \frac{I_b^i - I_a^i}{L_i}$$

Таблица 3 – Матрица несогласия

|    | P1            | P2            | P3            | P4            |
|----|---------------|---------------|---------------|---------------|
| P1 |               | (10-8)/10=0,2 | (5-3)/10=0,2  | (10-3)/10=0,7 |
| P2 | (3-1)/10=0,2  |               | (5-1)/10=0,4  | (10-1)/10=0,9 |
| P3 | (8-7)/10=0,1  | (10-7)/10=0,3 |               | (10-5)/10=0,5 |
| P4 | (10-6)/10=0,4 | (10-6)/10=0,4 | (10-6)/10=0,4 |               |

Определяем для нашей задачи пороги согласия и несогласия, в нашем случае будет  $C \geq 0,6$  и  $D \geq 0,6$ .

Строим матрицу превосходства по полученным данным. Для того, чтобы было значение «Истина» необходимо выполнить следующее условие  $C_i \geq C_{\text{заданное}}$  и  $D_i \leq D_{\text{заданное}}$ . В противном случае указываем значение «ложь».

Таблица 4 – Матрица превосходства

|    | P1   | P2   | P3     | P4   |
|----|------|------|--------|------|
| P1 |      | Ложь | Истина | Ложь |
| P2 | Ложь |      | Ложь   | Ложь |
| P3 | Ложь | Ложь |        | Ложь |
| P4 | Ложь | Ложь | Ложь   |      |

Из полученной матрицы превосходства можно сделать вывод, что наилучшим вариантом будет альтернатива P1.

Метод порогов несравнимости позволяет достаточно четко определить последовательность альтернатив, учитывая значимость каждого из критериев для итогового результата, причем гибкость в решении позволяет остановиться на определённом шаге и определить только ограниченное количество наиболее подходящих альтернатив, определить спорные альтернативы, что является важным при решении задачи выбора оптимального решения.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Лесин, В.В. Основы методов оптимизации / В.В. Лесин, Ю.П. Лисовец. – М.: Изд-во МАИ, 1995. – 344 с.
2. Шикин, Е. В. Исследование операций / Е. В. Шикин, Г. Е. Шикина. М.: ТК Велби, Изд-во Проспект, 2006. 280 с.
3. Ногин В. Д. Принятие решений в многокритериальной среде: количественный подход. / В.Д. Ногин – М.: ФИЗМАТЛИТ, 2005, 176 с.

УДК 004.032.26

**А.Ю. АРТАМОНОВ, С.Ю. ЗАЙКИН,  
М.И. ПАСЫНКОВ, М.Н. УЖЕГОВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПРИМЕНЕНИЕ ТЕХНОЛОГИИ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОБНАРУЖЕНИЯ ВОЛЕЙБОЛЬНОГО МЯЧА НА ИЗОБРАЖЕНИИ**

*Рассматривается процесс обучения и результаты применения модели нейронной сети для обнаружения волейбольного мяча на изображении.*

Проведение большинства спортивных мероприятий, в том числе волейбольных матчей, требует работы многочисленного судейского состава, так как счет и игровая статистика фактически ведутся в ручном режиме. Спорные вопросы (касание мяча или сетки, выход мяча за границы площадки и т.д.) на данный момент также решаются арбитрами. Использование видеоповторов позволяет повысить точность решений арбитра, однако занимает длительное время и не исключает ошибок.

В связи с этим предлагается разработать систему автоматического ведения статистики волейбольного матча с применением технологии глубокого обучения. Выбор метода решения обоснован скоростью работы нейронной сети, что имеет важное значение при работе в режиме реального времени.

В качестве первого этапа предлагается обучить модель нейронной сети для классификации изображений на два типа: изображения волейбольного мяча и прочие изображения. Модель реализуется на языке Python с применением библиотеки Keras.

Для обучения используется набор из 3000 изображений, треть из которых являются изображениями волейбольного мяча (рисунок 1).

Набор изображений делится на тренировочную и тестовую группы. Изображения предварительно приводятся к размеру 32×32 пикселя.



Рисунок 1 – Пример изображений из набора для обучения

Во избежание переобучения модели из-за малого числа исходных изображений применяется метод искусственного расширения подборки. Для этого модель дополняется слоями аугментации данных, трансформирующими исходные изображения [1]. В результате для каждого изображения из подборки создаются копии, полученные случайным отражением или поворотом оригинального изображения (рисунок 2).



Рисунок 2 – Пример результатов аугментации изображения

После подготовки изображений и создания модели нейронной сети проводится ее обучение. В качестве функции потерь выбирается логарифмическая функция, алгоритм оптимизации – Adam [2].

В течение первых пяти эпох обучения точность нейронной сети на тренировочной и тестовой подборках значительно разнятся, после чего точность на тестовой подборке постепенно увеличивается, достигая к восьмой эпохе практически 100% (рисунок 3). Обучение одной эпохи при указанных исходных данных занимает в среднем 30 секунд.

Значение функции потерь на тестовых данных увеличивается при выполнении четырех эпох обучения, после чего так же приближается к потерям на тренировочных данных, достигая нулевых значений к восьмой эпохе (рисунок 4).

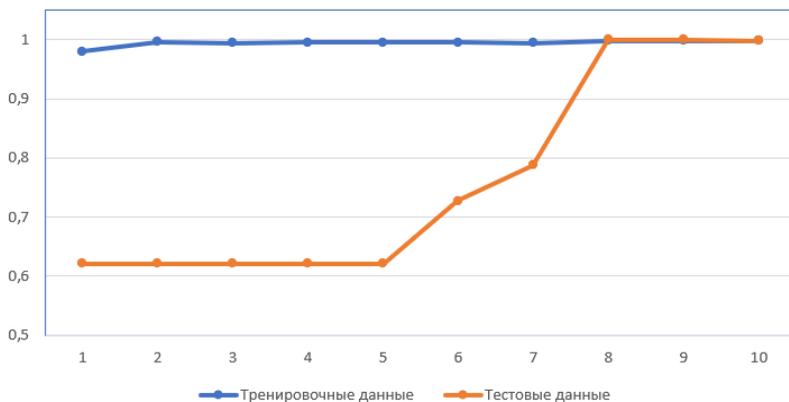


Рисунок 3 – График изменения значений точности по эпохам обучения



Рисунок 4 – График изменения значений функции потерь по эпохам обучения

Обученная модель может использоваться для классификации изображений (рисунок 5). Так, вероятность нахождения мяча на изображении слева оценивается моделью в 99.95%, на изображении справа – 0.46%.

Основным недостатком полученной модели является некорректное обнаружение мяча в случае, когда он занимает малую часть изображения. Разделение изображения на части размерами 32×32 пикселя и применение модели к каждой из этих частей занимает длительное время, что является неприемлемым, так как в дальнейшем модель предполагается использовать для работы с видеозаписью.

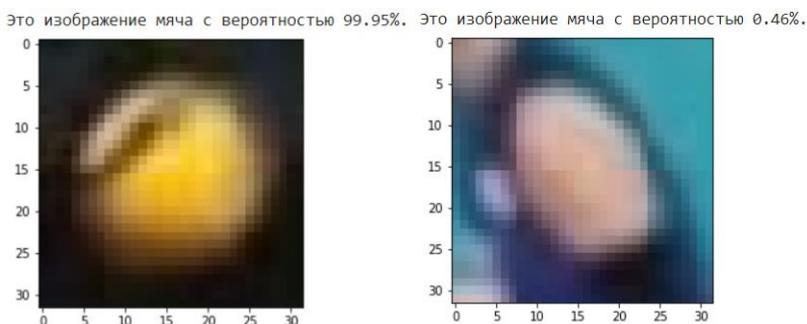


Рисунок 5 – Пример классификации изображений

На следующем этапе разработки предлагается организовать предварительную обработку изображения средствами библиотеки OpenCV. Это позволит уменьшить объем данных, обрабатываемых нейронной сетью, что приведет к ускорению ее работы.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Chollet, F. Deep Learning with Python. USA: Manning Publications. 2021. – 384 с.
2. Rosebrock, A. Deep Learning for Computer Vision with Python. PyImageSearch. 2017. – 332 с.
3. Николенко С., Кадури А., Архангельская Е. Глубокое обучение. Погружение в мир нейронных сетей. СПб: Питер. 2020. – 480 с.

УДК 621.3.049.75

**Ф.И. БАШКИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **ПРОЕКТИРОВАНИЕ ВЫСОКОПЛОТНЫХ СОЕДИНЕНИЙ**

*Рассматривается проблема проектирования высокоплотных соединений.*

Непрерывное улучшение микроэлектронной технологии, увеличение уровня внедрения микросхем, повышение функциональной наполненности электронного оборудования, рост скорости вычислительных процессов подразумевают непрерывное увеличение плотности распределения элементов межсоединений в печатных платах, изучения новейших технологий сборочно-монтажного изготовления, совершенствование критериев технологического обеспечения надёжности

Последующее возрастание плотности межсоединений возможно достичь в современных конструкциях печатных плат путём применения метода послойного наращивания с металлизированными отверстиями внутри слоёв печатной платы. Известно, что при заполнении глухих отверстий металлом, трассировочный потенциал печатных плат возрастает, как минимум, в 2.4 раза и упрощает процесс компьютерного топологического проектирования, причиной которого является произвольное размещение межслойных переходов по полю и объёму платы.

Анализ направлений увеличения плотности

Приоритетные пути наращивания плотности межсоединений в настоящее время:

- сокращение габаритов контактных площадок (КП)
- сокращение ширины проводников и зазоров;
- сокращение числа слоёв;
- применение многоуровневых соединений
- применение пространственных структур;
- применение интегрированных компонентов,

При имеющихся темпах развития микроэлектроники, когда размеры элементов межсоединений уже доведены до очень малых размеров (отверстий — до 0,05 мм, контактных площадок — до диаметра 0,2 мм).

Таблица 1 – Качественные оценки критериев направления модернизации производства и их числовые эквиваленты

| Сильно отрицательно | Средне отрицательно | Слабо отрицательно | Не влияет | Слабо положительно | Средне положительно | Сильно положительно |
|---------------------|---------------------|--------------------|-----------|--------------------|---------------------|---------------------|
| -3                  | -2                  | -1                 | 0         | 1                  | 2                   | 3                   |

Чтобы выпускать конкурентную продукцию, необходимо начинать модернизацию производства как можно раньше и проводить её поэтапно.

Исходя из выше сказанного, выделим условия, с помощью которых будем определять приоритетными пути совершенствования

- наращивание трассировочного пространства;
- сокращение длины соединений
- малая цена подготовки производства;
- малая цена производства;
- надежность изготавливаемой продукции.

Каждому критерию присвоим качественную оценку и её числовой эквивалент, необходимый для удобства подсчета. Используемые качественные оценки критериев представлены в табл. 1.

Итоговая оценка направлений представлена в табл. 2. Из таблицы видно, что увеличение плотности межсоединения невозможно без существенных затрат на уменьшение КП и сокращения ширины проводников и зазоров. Однако увеличение числа слоев, является простым и самым дорогим методом наращивания трассировочного пространства печатной платы, к тому же, данный метод понижает показатель надёжности. Сокращение размеров проводящих трасс не может быть бесконечным из-за возрастания их омического сопротивления в применяемой в печатной плате фольги. Самым действенным методом из трёх представленных является сокращение габаритов КП вокруг металлизированных отверстий.

Из выше указанных факторов следует, что для увеличения плотности межсоединений в гибких многослойных печатных платах (ГМПП) приоритетным методом является сокращение габаритов контактных площадок, но, для реализации увеличения трассировочного пространства при сокращении габаритов КП, требуется повысить точность совмещения элементов ГМПП на достаточном уровне.

Таблица 2 – Качественная оценка направлений увеличения плотности межсоединений

| Направление                              | Увеличение трассировочного пространства | Сокращение длины соединений | Затраты на подготовку производства | Затраты на изготовление | Надёжность | Итого |
|--|---|-----------------------------|------------------------------------|-------------------------|------------|-------|
| Сокращение габаритов-контактных площадок | 1                                       | 1                           | 2                                  | 3                       | 0          | 7     |
| Сокращение ширины проводников и зазоров  | 1                                       | 1                           | 2                                  | 2                       | -1         | 5     |
| Увеличение числа слоёв                   | 3                                       | -2                          | 2                                  | -2                      | -2         | -1    |
| Применение многоуровневых соединений     | 2                                       | 2                           | -1                                 | -1                      | 0          | 2     |
| Применение пространственных структур     | 2                                       | 1                           | -1                                 | -1                      | -2         | -1    |
| Применение второстепенных компонентов    | 3                                       | 3                           | -3                                 | -3                      | 2          | 2     |

Методов производства печатных плат достаточно много, но все они обладают минусами приведенными ниже:

- большая трудоемкость производства при малой надежности изделия для производства печатных плат с высоким классом точности;
- невозможно увеличение интеграции платы без уменьшения плотности межсоединений.

Главные методы уменьшения плотности межсоединений ГМПП:

- сокращение диаметра отверстий и контактных площадок;
- сокращение ширины проводящих трасс и зазора;
- увеличение числа слоёв платы;
- применение глухих и слепых отверстий.

Принимая во внимание то, что при действующих способах размещения микросхемы BGA термин шаг микросхемы допустимо называть условным. На практике печатные проводники допускается проводить по верхнему слою только с двух-трех внешних рядов выводов, в то

время как для оставшихся сигналов требуется создавать переходные отверстия рядом с контактными площадками, что не даёт в полной мере применять все достоинства компонентов в матричных типах корпусов.

Главной преградой на пути наращивания плотности межсоединений выступают контактные площадки большого диаметра, так как такие площадки сокращают трассировочное пространство между отверстиями. Габариты контактных площадок условно можно представить мишенью для сверлильного станка. Если погрешности пространственного совмещения элементов межсоединений высока, то больший размер контактных площадок просто необходим, чтобы обеспечить точное попадание в цель. Иначе говоря задача контактных площадок опоясывающих отверстия заключается в минимизации возможности смещения элементов межсоединений в слоях относительно друг друга, и, к всему этому, предотвратить создание отверстия за границей контактной площадки. Такие отклонения, обычно, вызываются размерной нестабильностью основного материала и смещением основания на некоторых этапах изготовления печатных плат.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. А. Медведев, В. Можаров Плотность межсоединений электронных компонентов // Печатный монтаж. 2011. № 3. С. 140–145.
2. Г.В. Мылов, И.В. Дрожжин Снижение плотности межсоединений многослойных печатных плат // Вестник РГРТУ. 2015. № 54. Часть 1.

УДК 004.9

**А.В. БЕЛЯЕВ, Т.А. ДМИТРИЕВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ИССЛЕДОВАНИЕ ОПТИМИЗАЦИИ ПРОГРАММНОГО КОДА И ЕЕ ВЛИЯНИЯ НА ОБЪЕМ КОДА НА ПРИМЕРЕ РЕШЕНИЯ ЗАДАЧ НА ЯЗЫКЕ 1С**

*Рассматриваются необходимость и практическая значимость оптимизации программного кода, а также ее влияние на объем кода. Исследование проведено на примере решения задач на языке 1С.*

Проблема оптимизации нередко является важнейшей частью при сдаче разработанного программного обеспечения заказчику и при ор-

ганизации его дальнейшей поддержки. Программное обеспечение может вызывать различные сбои, баги, эксплойты и другие ошибки, что приводит к различным проблемам в работе пользователя, и влечет за собой дополнительные сложности в работе самой программы. В данной статье будет рассмотрено возможное улучшение программного кода на примере решения задач на языке 1С.

В современном мире разработка программного обеспечения превратилась в одну из самых дорогостоящих индустрий, и любые ошибки и недочеты в процессе его создания могут привести к нежелательным результатам. Написание запутанного кода чревато проблематичным изменением и сопровождением готового продукта. Ошибки, не выявленные в ходе тестирования, приводят к снижению надежности и затягиванию сроков его внедрения. Поэтому актуальность разработки совершенного кода очень высока, так как она позволяет повысить его надежность. Очевидно, что программный код необходимо оптимизировать.

Оптимизировать программный код можно различными способами или методами, но нужно учитывать особенности каждой задачи [1, 2]. Под оптимизацией понимается улучшение чего-либо без явных изменений. С точки зрения программирования оптимизация включает в себя следующие основные моменты: изменения объема кода, так считается, что чем программа короче, тем она производительнее; производительность (сложность вычислений), то есть, в упрощенном понимании, это количество действий, которое потребуется совершить парсеру для выполнения инструкции; компиляция и декомпиляция или сериализация и десериализация – это скорость сборки продукта, запроса и так далее.

В данной статье остановимся на более подробном рассмотрении влияния оптимизации программного кода на объем кода. Проведем небольшое исследование на конкретном примере, который покажет, что не всегда программа тем производительней, чем меньше строк кода она содержит.

Для примера возьмем высокоуровневый язык 1С, типовую конфигурацию Бухгалтерия государственного учреждения, редакция 2.0 (2.0.82.16). На рисунках 1 и 2 приведены два примера функций, одна из которых оптимизирована, а другая нет.

```

Функция ДобавитьДокументPOST (Запрос)
// Веллев А.В. 02.12.21{
КодСостояния = 200;
СтрокаНоменклатураJSON = Запрос.ПолучитьТелоКакСтроку();
ЗаписьЖурналаРегистрации ("Получения данных Json", УровеньЖурналаРегистрации.Информация, "Выполнено",
СтрокаНоменклатураJSON, РежимТранзакцииЗаписиЖурналаРегистрации.Транзакционная);
ЧтениеJSON = Новый ЧтениеJSON();
ЧтениеJSON.УстановитьСтроку (СтрокаНоменклатураJSON);
// Веллев А.В. 02.12.21{
Данные = ПрочитатьJSON (ЧтениеJSON);
Если ЗначениеЗаполнено (Данные) Тогда
// Веллев А.В. 02.12.21{
//) Веллев А.В. 02.12.21
Если ТипИзч (Данные) = Тип ("Структура") Тогда
СоздатьДокументПлатежа (Данные, КодСостояния);
ИначеЕсли ТипИзч (Данные) = Тип ("Массив") Тогда
Для Каждого СтрокаДанных Из Данные Цикл
СоздатьДокументПлатежа (СтрокаДанных, КодСостояния);
КонецЦикла;
КонецЕсли;
ЗаписьЖурналаРегистрации ("Завершение работы с файлом json", УровеньЖурналаРегистрации.Информация, "Выполнено",
СтрокаНоменклатураJSON, РежимТранзакцииЗаписиЖурналаРегистрации.Транзакционная);
Иначе
ЗаписьЖурналаРегистрации ("Данные Json отсутствуют", УровеньЖурналаРегистрации.Информация, "Выполнено",
"Выполнено", РежимТранзакцииЗаписиЖурналаРегистрации.Транзакционная);
КонецЕсли;
//) Веллев А.В. 02.12.21
Если КодСостояния = 208 Тогда
Причина = "Already Reported";
Иначе
Причина = "Successfully";
КонецЕсли;
Ответ = Новый HTTPСервисОтвет (КодСостояния, Причина);
Возврат Ответ;
//) Веллев А.В. 02.12.21
КонецФункции

```

Рисунок 1 – Функция «ДобавитьДокументPost» (не оптимизирована)

На первый взгляд, эти две функции практически идентичны, они выполняют одно действие, но, вторая оптимизирована. При этом длина кода практически не уменьшилась. Связано это с тем, что во втором случае в некоторых местах не проводится запись в журнал сведений, так как вначале используется запись Json в этот журнал, и в дальнейшем ошибки можно идентифицировать по коду HTTP. Так же, добавляем секция «Попытка», что в данном случае необходимо.

```

функция ДобавитьДокументPOST(Запрос)
КодСостояния = 200;
СтрокаНоменклатураJSON = Запрос.ПолучитьТелоКакСтроку();
ЗаписьЖурналаРегистрации("Получения данных Json", УровеньЖурналаРегистрации.Информация, "Выполнено",
СтрокаНоменклатураJSON, РежимТранзакцииЗаписиЖурналаРегистрации.Транзакционная);
ЧтениеJSON = Новый ЧтениеJSON();
ЧтениеJSON.УстановитьСтроку(СтрокаНоменклатураJSON);
// Читаем данные json и преобразовываем их в структуру или массив
Данные = ПрочитатьJSON(ЧтениеJSON);
// Если данные присутствуют,
// Тогда проверяем, что это за тип данных
Если ЗначениеЗаполнено(Данные) Тогда
    // Если это структура, то проверим на item
    Если ТипЗнч(Данные) = Тип("Структура") Тогда
        Попытка
            Если ЗначениеЗаполнено(Данные.Item) Тогда
                Для Каждого СтрокаДанных Из Данные.Item Цикл ...
                    КонечЕсли;
                Исклечение;
                СоздатьДокумент(Данные, КодСостояния);
            КонечПопытки;
        ИначеЕсли ТипЗнч(Данные) = Тип("Массив") Тогда
            Для Каждого СтрокаДанных Из Данные Цикл
                СоздатьДокумент(СтрокаДанных, КодСостояния);
        КонечЦикла;
    КонечЕсли;
Иначе
    КодСостояния = 204;
КонечЕсли;

Если КодСостояния = 208 Тогда
    Причина = "Already Reported";
Иначе
    Причина = "Successfully";
КонечЕсли;
Ответ = Новый HTTPСервисОтвет(КодСостояния, Причина);

Возврат Ответ;
Конечфункции

```

Рисунок 2 – Функция «СоздатьДокументPost» (оптимизирована)

На рисунке 3 представлен вариант оптимизации процедуры «СоздатьДокумент». На 4 рисунке представлена процедура длиной более 1400 строк кода, на 3 рисунке 18 строчек кода, хотя они и выполняют одну роль, а именно, создают документ.

Связано это с тем, что создание документа было реализовано отдельно, а за все ошибки теперь отвечает переменная «Код состояния», которую так же легко идентифицировать по таблице HTTP. Например, если при выполнении запроса вернулся код 422, на рисунке 5, можно посмотреть, что он означает. Помимо этого можно реализовать возврат сообщения с дополнительной информацией, например, «422, Запрос был выполнен неудачно», также в качестве сообщения можно прикрепить описание ошибки: «422, ОписаниеОшибки()».

```

□ Процедура СоздатьДокумент (ДанныеJson, КодСостояния)
  //{
  // ЭтоКассовыйРасход = опеределяет, что будет создано. ЗКР или ПП.
  //}
  ЭтоКассовыйРасход = Ложь;
□ Если ДанныеJson.Свойство ("PayerAccount") Тогда
  // Условие, которое определяет размер счета
  // Если 11, т.е. казначейство, то это ЗКР, иначе ПП.
□ Если СтрДлина (Строка (ДанныеJson.PayerAccount)) = 11 Тогда
  ЭтоКассовыйРасход = Истина;
  КонецЕсли;
  Иначе
  КодСостояния = 422;
  Возврат;
  КонецЕсли;
  // Если существуют данные по json, то переходим к созданию
  // Иначе код 422, т.е. не можем отработать
□ Если ЗначениеЗаполнено (ДанныеJson) Тогда
  //Определяем, какой документ будем создавать
□ Если ЭтоКассовыйРасход = Ложь Тогда
  СоздатьДокументПлатежноеПоручение (ДанныеJson, КодСостояния);
  Иначе
  СоздатьДокументКассовыйРасход (ДанныеJson, КодСостояния);
  КонецЕсли;
  Иначе
  КодСостояния = 422;
  КонецЕсли;
КонецПроцедуры

```

Рисунок 3 – Процедура «СоздатьДокумент» (оптимизирована)

```

□ Процедура СоздатьДокументПлатежка (ДанныеJson, КодСостояния)
  ПолныйНайЗаполнения = "";
  НайденныйДокумент = "";
  СтатусДокумента = "";
  ЭтоКассовыйРасход = Ложь;
  Попытка
□ Если СтрДлина (Строка (ДанныеJson.PayerAccount)) = 11 Тогда □
  Исключение
  ЗаписьЖурналаРегистрации ("Ошибка при вычислении лицевого счета", УровеньЖурналаРегистрации.Ошибка, "Ошибка",
  "Во время вычисления лицевого счета была найдена ошибка:" + ОписаниеОшибки ());
  РежимТранзакцииЗаписиЖурналаРегистрации.Транзакционная);
  КонецПопытки;
□ Если ЗначениеЗаполнено (ДанныеJson) Тогда □
  КонецПроцедуры

```

Рисунок 4 – Процедура «СоздатьДокументПлатежка»  
(не оптимизирована)

- 422 Unprocessable Entity («необработываемый экземпляр»);
- 423 Locked («заблокировано»);
- 424 Failed Dependency («невыполненная зависимость»);
- 425 Too Early («слишком рано»);
- 426 Upgrade Required («необходимо обновление»);
- 428 Precondition Required («необходимо предусловие»)<sup>[11]</sup>;
- 429 Too Many Requests («слишком много запросов»)<sup>[11]</sup>;
- 431 Request Header Fields Too Large («поля заголовка запроса слишком большие»)<sup>[11]</sup>;

Рисунок 5 – Состояния HTTP ответов

Данный способ состоит в разделении программного кода, когда алгоритм работы кода не изменяется, но изменяется сам программный код. В рассматриваемом примере 1400 строк кода в одной процедуре преобразовано в 1600 строк кода в нескольких процедурах и функциях.

Рассмотрим более сложный пример для оптимизации программного кода, связанный с RLS (ограничением прав на уровне записи). На рисунках 6 и 7 представлены тексты запросов на языке 1С и на SQL. Запрос на рисунке 6 выглядит достаточно просто, но нужно разобраться, все ли в нем правильно.

```
#Если &ОграничениеДоступаНаУровнеЗаписейУниверсально #Тогда
#ДляРегистра("ИдентификаторыОбъектовМетаданных.РегистрНакопленияРасчеты", "Договор", "", "")
#Иначе
#ПоЗначениям( "РегистрНакопления.Расчеты","Чтение","",
"Компания","Договор.Подразделение.Компания",
"Кластеры","Договор.Подразделение.Кластер",
"Команды","Договор.Подразделение.Команда",
"Сквозды","Договор.Подразделение.Команда.Сквозд",
"Трайбы","Договор.Подразделение.Команда.Сквозд.Трайб", "", "", "", "", "", "", "", "", ""
#КонецЕсли
```

Рисунок 6 – Текст запроса в 1С

```
FROM dbo_AccumRg5191 T4
WHERE (EXISTS
(
SELECT 1
FROM
(
SELECT 1 AS SDBL_DUMMY
)
SDBL_DUAL
LEFT OUTER JOIN dbo_Reference59 T5
ON T4_Fld5192RRRef = T5_IDRRef
LEFT OUTER JOIN dbo_Reference64 T6
ON T5_Fld142RRRef = T6_IDRRef
LEFT OUTER JOIN dbo_Reference8944 T7
ON T6_Fld8957RRRef = T7_IDRRef
LEFT OUTER JOIN dbo_Reference8945 T8
ON T7_Fld8948RRRef = T8_IDRRef
WHERE EXISTS
(
SELECT 0x01 AS Q_001_F_000_
FROM dbo_Reference32 T9
INNER JOIN dbo_Reference23 T10
ON (T9_Fld634 = @P1)
AND EXISTS
(
SELECT 0x01 AS Q_002_F_000_
FROM dbo_InfoRg3177 T11
WHERE (T11_Fld3178_TYPE = 0x08
AND T11_Fld3178_RTRef = 0x00000020
AND T11_Fld3178_RRRef = T9_IDRRef)
AND (T11_Fld3179RRRef = T10_IDRRef)
)
AND T10_IDRRef IN
(
SELECT T12_Reference23_IDRRef AS Q_003_F_000RRRef
FROM dbo_Reference23_VT484 T12
INNER JOIN dbo_InfoRg3136 T13
ON (T13_Fld3138_TYPE = 0x08
AND T13_Fld3138_RTRef = 0x00000020
AND T13_Fld3138_RRRef = @P2)
AND (T13_Fld3137_TYPE = T12_Fld486_TYPE
AND T13_Fld3137_RTRef = T12_Fld486_RTRef
AND T13_Fld3137_RRRef = T12_Fld486_RRRef)
)
)
)
WHERE (
CASE
WHEN T6_Fld1195RRRef IS NULL THEN 0x01
```

Рисунок 7 – Текст запроса в SQL

Запрос на рисунке 7 выглядит гораздо сложнее. Самая большая проблема – разыменование полей и таблиц с RLS. В этом случае, зачастую, количество итераций запроса с ограничениями вытекает в огромный поток данных на уровне SQL, причем, для каждой строки (рисунок 8).

```
Row Count Spool (Lazy Spool)
Stores the data from the input into a temporary table in order to optimize rewinds.
1,0 rows per iteration, for a total of 54 772 240 rows over 54 772 240 iterations.
Node ID: 42
Physical Operation: Row Count Spool
Logical Operation: Lazy Spool
Estimated Execution Mode: Row
Estimated Rows: 54 772 240
Estimated I/O Cost: 0,0000000
Estimated CPU Cost: 5 482,7012240
Estimated Executions: 54 772 240,0
Estimated Operator Cost: 5 482,7000000 (69,6%)
Estimated Subtree Cost: 6 955,3200000
Estimated Row Size: 9 B
Estimated Data Size: 470 MB
```

Рисунок 8 – Количество итераций в SQL

Таким образом, проведенное исследование оптимизации программного кода при решении задач на языке 1С показало, что не всегда при оптимизации программного кода происходит уменьшение числа строк кода, т.е. нельзя оценивать оптимизирован программный код или нет только по данному параметру.

В результате, в данной статье была рассмотрена проблема оптимизации программного кода, доказана актуальность решения данной задачи. Также проведено исследование влияние оптимизации программного кода на такой параметр, как число строк кода на примере задач, решаемых на языке 1С. В дальнейшем планируется исследовать оптимизацию производительности, а также оптимизацию через компиляцию / декомпиляцию, сериализацию / десериализацию.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Типичные причины неоптимальной работы запросов и методы оптимизации [Электронный ресурс]. – Режим доступа: <https://its.1c.ru/db/metod8dev/content/5842/hdoc>. Дата обращения 10.04.2022 г.

2. Компиляция с оптимизацией [Электронный ресурс]. – Режим доступа: <https://www.ibm.com/docs/ru/aix/7.2?topic=techniques-compiling-optimization>. Дата обращения 12.04.2022 г.

УДК 004.048

**А.Д. БЕРЕЖНОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **МНОГОКРИТЕРИАЛЬНЫЕ МЕТОДЫ ПРИНЯТИЯ РЕШЕНИЙ**

*В данной статье будет рассматриваться подход к принятию решений и оптимизации с помощью методов компенсации. Полезность данного метода заключается в сравнении достоинств и недостатков альтернатив. Приведен пример употребления предоставленного метода на практике.*

Одной из характерных тенденций нынешнего периода научного образования является создание новых направлений, объединяющих различные области научных знаний. Среди таких направлений исследования входят в многокритериальные задачи принятия решений.

Вопрос о многокритериальной задаче сформулирован следующим образом: существует множество возможных альтернатив, каждая из которых оценивается по целому ряду критериев. Необходимо найти лучшую альтернативу. Таким образом, главная трудность заключается в неясности выбора наилучшего решения. Для его устранения используются две группы методов. В методах первой категории уменьшают количество критериев, по которым будут включать дополнительные предположения относительно процедуры сортировки критериев и сравнения альтернатив. В методах второй категории стремятся сократить количество альтернатив в начальном количестве и заранее исключить плохие альтернативы.

Стратегии методов компенсации заключаются в том, что аналитические консультанты пытаются сравнить оценки одних вариантов с оценками других с помощью одного из следующих "возможных" методов стратегий:

- определение полезности одной из альтернатив (т.е. определение предпочтений) и последующее сравнение (стратегия аддитивной модели);
- сравнение оценок полезности альтернатив по каждому критерию отдельно, затем сложение этих различий (модель аддитивных различий);
- сравнение плюсов и минусов одной и той же альтернативы в словесном анализе или компенсации по методу Б. Франклина.

Идея Б. Франклина (метод компенсации преимуществ и недостатков) основано на идее компромисса между несовместимыми оценками по двум или более критериям. Б. Франклин отметил, что при сравнении трудно одновременно учитывать все плюсы и минусы одной из альтернатив. Вот почему он занес плюсы и минусы альтернатив в два отдельных списка. И после тщательного анализа было определено, какой недостаток (или их совокупность) можно считать эквивалентным некоторым достоинствам (или их совокупности) этой альтернативы, после чего она была удалена из списка. Франклин назвал этот метод "нравственной алгеброй".

Основой систем компенсации является поиск удовлетворительных значений критериев. Эти процедуры также специализируются на систематическом поиске оптимального решения. Однако такой поиск выполняется другим способом: в свою очередь, определяется приемимое значение для каждого из критериев.

Переход от сравнения преимуществ по разным критериям к сравнению альтернатив может осуществляться несколькими способами. Среди них необходимо выделить построение кривых безразличия и сравнение различий в оценках альтернатив по критериям.

На рисунке 1 показана выборка кривых безразличия для двух критериев оценки альтернатив. Точки  $p_1$ ,  $p_2$ ,  $p_3$  называются начальными. Построение кривой безразличия  $k_i$  выполняется следующим образом:

1. Выбирается начальная точка  $p_1(x_1, y_1)$ , где  $x_1$ ,  $y_1$  - это значение критерия X, Y в этой точке;
2. Сложение  $d$  выбирается по критерию X (положительному или отрицательному) и определяется  $x_2 = x_1 + d$ ;
3. Значение  $y_2$  определяется критерием Y таким образом, чтобы точка  $p_1'(x_2, y_2)$  в полезности соответствовала точке  $p_1(x_1, y_1)$ ;
4. Кривая безразличия рисуется вдоль найденных точек.

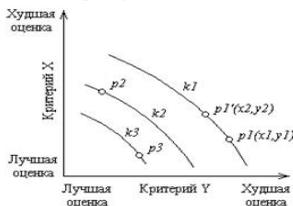


Рисунок 1 – Пример кривых безразличия

Когда поверхности безразличия строятся в пространстве с несколькими критериями, сравнить альтернативы чрезвычайно просто,

потому что поверхности (или начальные точки) могут быть упорядочены по полезности. Следовательно, точка  $p_2$  на рисунке 1 равна  $p_1$ .

Возникновение ряда похожих свойств бинарных отношений при использовании методов, сравнивающих альтернативы с различиями в оценках, требует построения изолированных методов итеративного типа, в которых информация, полученная во время назначений, не содержит элементов несовместимых друг с другом. По большей степени случаев, методы сравнения различимые в оценке, приводят к тому, что многокритериальные альтернативы приходится сравнивать напрямую.

Основной алгоритм решения задачи метода компенсации:

Заполняется матрица сравнения альтернатив. Находятся глобальные оценки. Метод компенсаций подразумевает попарное вычеркивание альтернатив на основе нахождения среди глобальных оценок максимального и минимального значения и вычеркивание их, тем самым компенсируя, и так до тех пор, пока не останется один вектор альтернативы.

Решение задачи:

Пусть нам нужно выбрать среди трех альтернатив  $A_1$ ,  $A_2$  и  $A_3$ . Введем вектор желаемых оценок по критериям  $K_1$ ,  $K_2$ ,  $K_3$ .

$$K = \{1, 4, 8\}$$

Введем вектора оценок альтернатив:

$$A_1 = \{1, 6, 7\}$$

$$A_2 = \{5, 3, 8\}$$

$$A_3 = \{4, 2, 2\}$$

Запишем полученные данные в таблицу сравнения альтернатив:

Таблица 1 – Таблица сравнения альтернатив

|          | $K_1$ | $K_2$ | $K_3$ | Глобальные критерии |
|----------|-------|-------|-------|---------------------|
| Критерии | 1     | 4     | 8     |                     |
| $A_1$    | 1     | 5     | 4     | $\mu_1$             |
| $A_2$    | 6     | 3     | 2     | $\mu_2$             |
| $A_3$    | 7     | 8     | 2     | $\mu_3$             |

Рассчитаем глобальные переменные  $\mu$  используя формулу:

$$\mu_i = u_{ij}v_1 + u_{ij}v_2 + u_{ij}v_3$$

Получим:

Таблица 2 – Таблица сравнения альтернатив

|          | $K_1$ | $K_2$ | $K_3$ | Глобальные критерии |
|----------|-------|-------|-------|---------------------|
| Критерии | 1     | 4     | 8     |                     |
| $A_1$    | 1     | 5     | 4     | <b>53</b>           |
| $A_2$    | 6     | 3     | 2     | <b>34</b>           |
| $A_3$    | 7     | 8     | 2     | <b>55</b>           |

Найдем максимальное и минимальное значение среди глобальных элементов и вычеркиваем вектор альтернатив этих элементов, получим, что оптимальным является вектор  $A_1$ , так как максимальное = 55, а минимальное = 34.

Таким образом, использование метода компенсации позволяет четко определить преимущества и недостатки каждой альтернативы и изучить наилучший вариант, исключив плюсы и минусы альтернатив, что является важнейшим параметром для решения задач принятия решений.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Многокритериальные методы принятия решений: проблемы и перспективы [Электронный ресурс]. – Режим доступа: <http://gaai.org/news/pii/ppt/Roizenzon.pdf>. Дата обращения 10.04.2022 г.
2. Захарова А.А. Лекция Методы оценки и сравнения многокритериальных альтернатив. 2016г.
3. Черноруцкий И.Г. Методы принятия решений: - учеб. Пособие / И.Г. Черноруцкий, Санкт-Петербург, 2012.

УДК 665.9

**Д.В. БОБРОВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### СМОЛЫ, ПРИМЕНЯЕМЫЕ В РАДИОЭЛЕКТРОННОЙ ПРОМЫШЛЕННОСТИ

*Данная статья посвящена обзору видам смол и их применению на производстве.*

Электроизоляционные смолы - это группа природных и синтетических материалов, которые способны создавать пленки из раствора или же расплава [1]. По своему химическому составу они могут иметь малую молекулярную массу или же обладать большой молекулярной массой. При обычной температуре их делят на твердые хрупкие вещества и жидкости с различной степенью вязкости. Чаще всего смолы невозможно растворить в воде, но при использовании растворителей это сделать удастся. Включают в себя добротные адгезионные и высочайшие диэлектрические качества (зависят от полярности).

Если смолы неполярные, то такие диэлектрические характеристики как  $\beta$ ,  $\rho$  и  $\epsilon$ , довольно высоки (например, полиэтилен). Механи-

ческие свойства формируются энергией межмолекулярного притяжения, где суммарный размер в значимой мере превосходит энергию валентных связей изнутри самой молекулы.

Синтетические смолы — это продукт, полученный в результате химического синтеза из высокомолекулярных соединений [2].

Электроизоляционные синтетические смолы в чистом виде — целлофан, полистирол, капрон и применяют для приготовления лаков и компаундов. И в совокупности с другими материалами — с растительными маслами, пластификаторами, отвердителями; наполнителями. Используя свойства полимеризации синтетических смол их используют в бытовой сфере, например, окна ПВХ и пластик. В большинстве случаев смолы являются токсичными или легковоспламеняющимися материалами, поэтому при работе с ними соблюдаются правила техники безопасности.

Рассмотрим некоторые виды смол:

**Природные (естественные) смолы** — продукты жизнедеятельности животных или же растительных организмов [3]. Примерами естественных смол в производстве являются канифоль, шеллак и копалы.

Канифолью называют хрупкую немного стеклообразную неправильной формы смолу, взятую из сока хвойных деревьев, предпочтительно из сосны. Но есть и смолы других хвойных пород: кедра, пихты и лиственницы. Пихтовый бальзам, различается высокой степенью прозрачности и нормированной характеристикой преломления, который используют в качестве клея для склеивания оптических линз. В её состав входят кислоты, а именно абиетиновая кислота, эфирное масло и другие вещества. Содержание кислот в канифоли составляет около 85–90%. Это термопластичный материал, размягчающийся при нагревании и растворим в спирте, бензине и минеральных маслах.

Шеллак собирают из гуммилака, по виду напоминающего смолу, которая образуется на ветвях тропических растений вследствие укуса пресмыкающегося. Чаще всего привозят его из Индии. Шеллак получают в облик чешуек — от светло-лимонного до темно-оранжевого цветового контраста в зависимости от высоты уровня очищения. Он состоит из алейритиновой ( $C_{16}H_{32}O_5$ ) и шеллоновой ( $C_{15}H_{20}O_5$ ) жирных кислот. Растворяется шеллак в спирте, аммиаке, в растворах едких щелочей, соды, буры. Шеллак неплохо сплавляется с канифолью, глифталями и др. Чаще всего его в виде растворов спирта и в форме порошка. В производстве лаков шеллак используется в ограниченных количествах и в виде порошка для изготовления некоторых сортов коллекционного миканита

Копалами называют смолы, окаменелости растительного происхождения, добываемые в основном в тропических странах. Копалы выглядят как твердые вещества в облике ломтей, цветового тона и отчётливости, характеризующихся высочайшей температурной средой плавления. Янтарь в распоряжении владеет высочайшей твердостью и температурой плавления если сравнить с другими ископаемыми смолами, он не может быть растворён в растворителе. Температура размягчения 175–200 °С, температурная обстановка плавления — выше 300 °С. Янтарь обладает значительными диэлектрическими свойствами, высоким противодействием изоляции, что обуславливает значительную стоимость диэлектрика для изготовления электроизмерительных приборных основ.

**Эпоксидными смолами** называются продукты конденсации многоатомных фенолов (дифенилолпропан) с соединениями, содержащими эпоксидную группу, например, эпихлоргидрин глицерина[4]. Реакция проходит в щелочной среде с раствором NaOH. Наибольшее применения для этого вида смолы получили дифенилолпропан и эпихлоргидрин

Как же проходит технологический процесс? Всего 4 стадии производства: конденсация, промывка, фильтрация и сушка.

В реактор с пароводяной рубашкой и мешалкой погружают эпихлоргидрин и ждут его нагревания до 40–50 °С. Далее постепенно вводят дифенилолпропан, где при этом работает мешалка. Потом ждут пока состав не будет однородной массы и тонкой струей из мерного стакана подливают раствор едкого натрия и проводят процесс конденсации при температуре 70 °С продолжением в два часа. После конденсации отключают подогрев аппарата и заливают воду, при этом, не прекращая перемешивание. Когда перемешивание заканчивают, то смоле дают отстояться. Далее происходит этап деления слоев (примерно в 50 °С). Настоявшийся водный слой (сверху) разделяют с оставшейся смолой и очищают при помощи теплой воды около трех раз при 40–50 °С. Все выше перечисленные действия не прекращают пока не будет полного снятия поваренной соли, возникшей при реакции. Контроль промывки осуществляется при помощи применения хлора и щелочи. Следующий этап-это сушка. При нагревании смолы до 40–50 °С, подключают к холодильнику и сушат пока не прекратиться конденсации воды в холодильнике. Этап продолжается до получения прозрачной пробы. Готовая смола сливается в алюминиевую тару. Конечный продукт может быть жидким, вязким и твердым.

Если брать за основу такие факторы как природа и соотношение компонентов, а также катализаторов, то **фенолоформальдегидные**

**смолы** делятся на два вида: термореактивные или резольные, термопластичные или новолачные смолы. В зависимости от температурных факторов, например, нагревании или долгосрочном хранении резольные переходят в неплавкое и нерастворимое состояние, а новолачные смолы термопластичны, сохраняют плавкость и растворимость при длительном хранении. Резольные смолы при переходе из плавкого растворимого состояния в неплавкое проходят три стадии:

Резолы — смолы первой стадии. При нагревании они начинают плавиться и растворяться в спирте, ацетоне и в водных растворах щелочей. Представляют собой смесь низкомолекулярных соединений продуктов реакции[5].

Резитолы — смолы второй стадии. Они возникают при нагревании или длительном хранении, представляют собой смесь смоляной смолы и высокомолекулярных неплавких и нерастворимых соединений. Они нерастворимы в спирте и ацетоне, но набухают в них и размягчаются при нагревании[5].

Резиты — смолы третьей стадии. Они содержат в основном трехмерные высокомолекулярные соединения. Они не плавятся и не размягчаются при нагревании, нерастворимы и не набухают ни в каких растворителях.[5].

Новолаки или новолачные смолы представляют собой термопластичные фенолальдегидные олигомерные поликонденсационные смолы. Новолаковые смолы представляют собой стекловидные вещества в виде частиц разной формы, окрашенные от светло-желтого до насыщенно-красного цвета. Они растворимы в спиртах, кетонах, сложных эфирах, фенолах и растворах щелочей. Обычно такие смолы разбухают под действием воды, но могут храниться длительное время при отсутствии доступа влаги. Новолаки более подвижны, чем термореактивные полимеры той же природы, линейные олигомеры. В их макромолекулах ядра бензола соединены между собой метиленовыми мостиками. Их главной особенностью по сравнению с резолами помимо их низкой молекулярной массы, является то, что новолаки почти не включают в состав метильные группы. Для того чтобы затвердеть новой смоле и переработать ее, необходимо добавить отвердитель в полимерную композицию, обычно гексаметиленetetрамин.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Смолы, используемые на производстве [Электронный ресурс]. – Режим доступа: [http://www.compitech.ru/html.cgi/arhiv/03\\_03/stat\\_166.htm](http://www.compitech.ru/html.cgi/arhiv/03_03/stat_166.htm).
2. Всё о синтетических смолах [Электронный ресурс]. – Режим

доступа: <https://building-ooo.ru/uncategorized/sinteticheskie-smoly-vidyprimeneniopredeleniefotoklassifikaciya/.html>.

3. Компоненты и технологии [Электронный ресурс]. – Режим доступа: <https://kit-e.ru/circuitbrd/prirodnye-smoly>.

4. Пакен А.М. Эпоксидные соединения и эпоксидные смолы - 5. Ленинград, Госхимиздат, 1962. – 964 с.

6. Применение фенолоформальдегидных смол [Электронный ресурс]. – Режим доступа: <https://wizard-aerosol.com/klej/kak-izgotovlyaetsya-i-gde-rime-pyaetsya-fenolformal-degidnaya-smola.html>.

УДК 004.415

**А.С. БОЛЬШАКОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **АРХИТЕКТУРА И ОСОБЕННОСТИ РАЗРАБОТКИ ВЕБ-API С ПОМОЩЬЮ ТЕХНОЛОГИИ ASP.NET**

*Рассматриваются особенности ASP.NET и анализ технологии с помощью которой можно реализовать веб приложение. Осуществляется применение паттерна MVC.*

ASP.NET Core – это кроссплатформенная высокопроизводительная платформа с открытым исходным кодом для создания современных облачных приложений, подключенных к Интернету. За счет разработанной архитектуры, имеются следующие достоинства: высокая степень защищенности, минимальное обслуживание и усовершенствованная эффективность.

Выделяют следующие особенности ASP.NET Core:

- Код ASP.NET компилируется, а не интерпретируется;
- Является объектно-ориентированной технологией;
- Встроенная поддержка создания и использования NuGet пакетов;
- Единый технологический стек для Web UI и Web APIs;
- Упрощенный и модульный конвейер HTTP запросов;
- Создание и запуск кроссплатформенных приложений ASP.NET Core для Windows, Mac и Linux;
- Поддержка множеств языков программирования;
- Среда основанная на облачной системе конфигурации;
- Открытый исходный код;

- Легкое выполнение развертывания и конфигурирования;
- Обслуживается средой CLR;
- Возможность одновременной работы с различными версиями ASP.NET Core;
- Возможность введения зависимостей;
- Расширяемость, основывающаяся на возможности добавления в проект дополнительных компонентов.

Приложения ASP.Net, разработанные с помощью паттерна MVC (Model-View-Controller), имеют соответствующий архитектурный шаблон: модель-представление-контроллер.

Модель описывает используемые в приложении данные, реализуют логику, которая связана непосредственно с данным. Часто объекты модели извлекают и сохраняют состояние модели в базе данных.

Представление отображает пользовательский интерфейс, также содержит логику, связанную с отображением данных. Обычно пользовательский интерфейс создается из данных модели.

Контроллеры – это компоненты, которые обрабатывают взаимодействие с пользователем, работают с моделью и выбирают представление для визуализации, отображающее пользовательский интерфейс. Пользователь использует представление и создает HTTP-запрос, который будет обрабатываться контроллером. Он содержит логику обработки запроса пользователя. Контроллер получает введенные данные и обрабатывает их. И в зависимости от результата, обрабатывает запрос и возвращает соответствующее представление в качестве ответа.



Рисунок 1 - Архитектура

Последовательность шагов взаимодействия пользователя с веб сайтом, разработанным с помощью технологии ASP.NET:

1. Пользователь осуществляет запрос;

2. Запрос доходит до веб сервера и перенаправляется на механизм маршрутизации;

3. На основе URL-адреса, механизм маршрутизации выбирает соответствующий контроллер;

4. Контроллер обращается к базе данных, используя построенную модель, чтобы получить соответствующие данные для введенного запроса;

5. Контроллер вызывает механизм просмотра и возвращается представление страниц;

6. Контроллер возвращает полученное представление;

7. Запрошенный ресурс отправляется обратно в браузер.

Проанализируем преимущества и недостатки технологии:

1. Единый набор технологий для шаблонов MVC и API;

2. Изменения структуры проекта;

3. Кроссплатформенность;

4. Встроенная поддержка внедрения зависимостей;

5. Сквозной проборс зависимостей и единый пакет SDK;

6. Открытый исходный код;

7. Использование всего одного языка для всего веб-приложения, используя его сильные стороны.

Недостатки:

1. Небольшое количество инструментов;

2. Поддержка и сопровождение устаревшего кода.

MVC и Web Api родственные технологии, MVC более универсальна и направлена на разработку широкого спектра приложений, Web Api предназначен для разработки веб-сервисов.

Веб-сервис – это набор компонентов способных принимать запросы от клиентов и предоставляющий для них набор методов, возвращающих данные, которые клиент хочет получить.

Визуализация данных у клиента приложения Web Api – это исключительно ответственность клиента. Api сервер считает, если клиент сделал запрос на данные, значит он знает, что с ними делать.

Клиент ориентированный на работу с API должен иметь собственные механизмы визуализации данных, сам реализовывать интерфейс для пользователя.

Веб-сервис может быть самостоятельным приложением или только частью.

Существуют 4 типа приложений в которых MVC и Web Api взаимодействуют:

- Одностраничные приложения;
- Приложения с общей моделью;

- Приложения с нативными клиентами;
- Сервисные веб-приложения (веб-службы).

Таким образом, сравнивая ASP.NET с другими широко используемыми средствами веб разработки, такими как PHP, Node.js или Ruby, можно выделить одно ключевое достоинство. Перечисленные платформы являются интерпретируемыми, в то время как C# - компилируемый язык. Это дает системе .NET серьезное преимущество в скорости работы. Все компоненты, не требуют интерпретатора, работают с фреймворком, который, в свою очередь, также скомпилирован и вызывает напрямую функции операционной системы, а большинство ошибок отлавливаются разработчиком в момент компиляции. Также наличие паттерна MVC позволяет разграничить написание кода на 3 основные части, что в свою очередь значительно добавляет удобство пользования.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Официальная документация ASP.NET [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/aspnet> (дата обращения 13.04.2022).
2. Официальная документация ASP.NET MVC Pattern [Электронный ресурс]. – Режим доступа: <https://dotnet.microsoft.com/en-us/apps/aspnet/mvc> (дата обращения 13.04.2022).

УДК 004.72

### **А.С. БОЛЬШАКОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ОСОБЕННОСТИ И СТРУКТУРА БЕСПРОВОДНОЙ ПЕРЕДАЧИ ДАННЫХ ПРОТОКОЛА ZIGBEE**

*Рассмотрены технические и практические особенности протокола беспроводной передачи данных ZigBee. Осуществлен обзор структуры сети.*

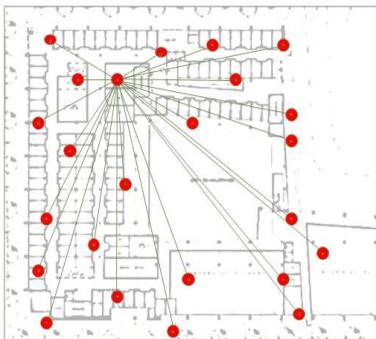
ZigBee это язык беспроводной связи один из самых надежных в мире стандартных протоколов. В системах промышленной телеметрии и управления поднимается вопрос передачи данных на небольшие расстояния. Беспроводные сети на базе стандарта IEEE 802.15.4 представляют собой альтернативу проводным соединениям в распределенных

системах мониторинга и управления и отличаются более гибкой архитектурой, требуют меньших затрат при их установке и эксплуатации.

В название не просто так присутствует слово Bee - пчела, Zig как производная зигзаг.

Технология имеет ячеистый тип соединения и зигзагообразную траекторию подачи сигнала, схожую с танцем пчелы при ее возвращении в улей, когда она сообщает собратьям в каком направлении она обнаружила нектар.

### Wi-Fi, Bluetooth



### ZigBee

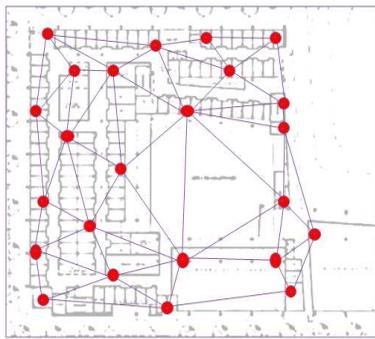


Рисунок 1 – Сравнение топологии “Звезда” и “Mesh”

Сигнал может передаваться как от устройства к контроллеру, так и от устройства к устройству, а затем к контроллеру что гарантирует безошибочную и быстрое срабатывание, это позволяет равномерно покрывать большие площади сохраняя надежное соединение с каждым из десятков устройств в отличие от уже известного протокола Wi-Fi, который строится по схеме звезда, где контроллер располагается в центре, а на лучах звезды находится устройство, они связываются с контроллером напрямую, при возникновении помех передача данных будет невозможно.

Спецификация IEEE 802.15.4 выдвигает требования к безопасности передачи данных – обязательное использование шифрования 128-битным ключом по стандарту AES на уровне “Звена данных”. Стандарт 802.15.4 предполагает наличие уникального 64-битного адреса на MAC уровне, а также наличия дополнительного 16-битного сетевого адреса (PAN-ID) для определения принадлежности данного устройства к той или иной WPAN (Wireless Personal Area Network).

На сегодняшний день абсолютное большинство устройств для умного дома выпускается на протоколе ZigBee и на это есть причины

низкое энергопотребление в отличие от протокола Wi-Fi. На одной 3V батарейке, датчик может проработать два года, все потому что, режим работы запускается только тогда, когда срабатывает заложенный устройством спящем сценарий. Например, датчик затопления находится в спящем режиме, ровно до того момента, пока не появится утечка воды, только после этого он проснется и отправит сигнал на контроллер.

Среди прочих достоинств стандарта следует отметить хорошую масштабируемость, возможность самостановления в случае сбоев и простоту настройки. При применении 64-битной адресации в единую сеть могут быть объединены свыше 60 тысяч ZigBee-устройств.

### Типовая структура сети ZigBee.

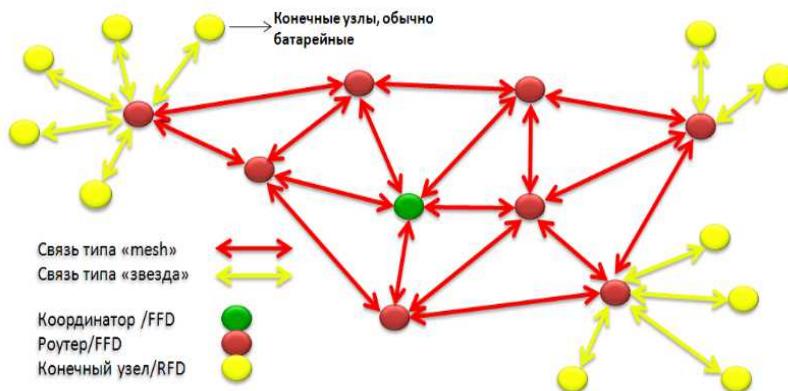


Рисунок 2 – Типовая структура сети ZigBee

**Координатор** — это узел, организовавший сеть. Именно он выбирает политику безопасности сети, разрешает или запрещает подключение к сети новых устройств, а также при наличии помех в радиоэфире инициирует процесс перевода всех устройств в сети на другой частотный канал.

**Роутер** — это узел, который имеет стационарное питание и, следовательно, может постоянно участвовать в работе сети. Координатор также является роутером. На узлах этого типа лежит ответственность по маршрутизации сетевого трафика. Роутеры постоянно поддерживают специальные таблицы маршрутизации, которые используются для прокладки оптимального маршрута и поиска нового, если вдруг какое-либо устройство вышло из строя.

**Конечное устройство** — это устройство, которое подключается к сети через родительский узел – роутер или координатор – и не участ-

вует в маршрутизации трафика. Все общение с сетью для них ограничивается передачей пакетов на «родительский» узел либо считыванием поступивших данных с него же. «Родителем» для таких устройств может быть любой роутер или координатор. Конечные устройства большую часть времени находятся в спящем режиме и отправляют управляющее или информационное сообщение обычно только по определенному событию (нажатие кнопки выключателя, открытие окна или двери). Это позволяет им долго сохранять энергию встроенного источника питания.

Теоретически можно говорить, что на частоте 2.4 гигагерца возможны помехи от других устройств, работающих в этом же диапазоне, например, Wi-Fi тоже работает в этом диапазоне, но практика показывает, что система четко и слаженно срабатывает без сбоев и задержки всего за 15 миллисекунд, специалисты не считают это недостатком.

Таким образом, использование протокола ZigBee позволит обеспечить необрывную структуру сети, снизить затраты на энергопотребление. Протокол обладает большим набором возможностей для организации локальной беспроводных сетей.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. ZigBee – протокол передачи данных [Электронный ресурс]. – Режим доступа: <https://xiaomi-smarthome.ru/zigbee/> (дата обращения: 14.04.22).
2. Скуснов Александр. Zigbee: обзор беспроводной технологии [Текст]. – Качество. Инновации. Образование. 2005. №47. С. 176-179.

УДК 004.04

### **А.В. ВИКЛОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ИСПОЛЬЗОВАНИЕ В-ДЕРЕВЬЕВ В ФАЙЛОВЫХ СИСТЕМАХ**

*В статье описывается устройство В-деревьев и их применение в работе файловых систем, его преимущества и недостатки.*

В-деревья являются сбалансированными деревьями поиска. Их можно назвать естественным обобщением бинарных деревьев, однако, в отличие от бинарных деревьев, В-деревья имеют не два дочерних узла, а гораздо больше. Таким образом, если каждый узел бинарного

дерева содержит всего два дочерних узла, то в В-дереве количество дочерних узлов может достигать до нескольких тысяч. Степень ветвления В-деревя обычно определяется исходя из практических задач, в которых оно используется. Такое дерево определяется следующими параметрами:

- Высота В-деревя может быть не более  $O(\log N)$ , где  $N$  – количество узлов.
- Каждый ключ дерева содержит не менее одного ключа.
- Количество дочерних узлов равно  $K + 1$ , где  $K$  – количество ключей в узле.

На рисунке 1 показано В-дерево со степенью ветвления, равной 3, состоящее из 20 узлов:

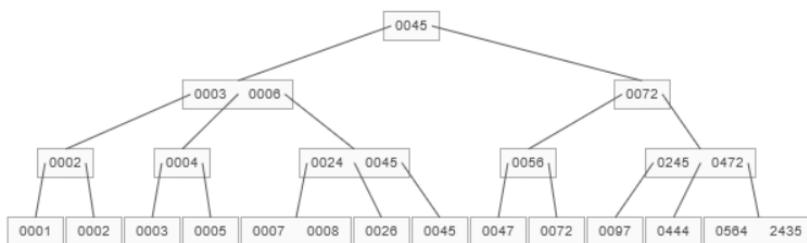


Рисунок 1 – Пример В-деревя

Благодаря такому подходу к хранению данных, мы получаем достаточно быстрый поиск требующейся информации, а также малое количество обращений к физическому носителю информации. Количество операций с носителем будет равно высоте дерева, т. е.  $O(\log_t N)$ , где  $t$  – минимальная степень дерева,  $N$  – количество узлов.

На практике часто встречается одна из разновидностей В-деревьев – В<sup>+</sup>-дерево. В нем информация хранится в листьях, а внутренние узлы содержат только ключи. Такой подход используется в различных файловых системах, таких как XFS, NTFS, BTRFS и APFS.

Такие файловые системы создаются из леса деревьев. Суперблок, имеющий фиксированное положение на диске является якорем. Он указывает на дерево, содержащее указатели корней других деревьев, которое индексирует В-деревья, составляющие файловую систему.

На примере файловой системы BTRFS можно увидеть, что деревья бывают следующих видов:

- **Вложенные тома:** хранят видимые пользователю файлы и каталоги. Каждый такой том реализован отдельным деревом. Вложенные тома могут быть выделены и скопированы, создавая

дополнительные В-деревья. Корни всех таких томов индексируются деревом корней дерева.

- **Дерево распределения экстентов (непрерывных областей диска):** отслеживает выделенные экстенты в элементах экстентов, и служит картой свободного пространства на диске. Все обратные ссылки на экстенст записывается в элементе экстенста. Это позволяет перемещать экстенст при необходимости или восстановить поврежденный блок диска. Взятые в целом, обратные ссылки умножают количество указателей на диск файловой системы на два. Для каждого прямого указателя существует ровно один обратный указатель.

- **Дерево контрольных сумм:** содержит элемент контрольной суммы для каждого выделенного экстенста. Элемент содержит список контрольных сумм на одну страницу экстенста.

- **Деревья фрагментов и устройств:** косвенный уровень для обработки физических носителей.

- **Дерево перемещений:** для специальных операций, связанных с перемещением экстентов, например дефрагментации данных.

На рисунке 2 показано общее представление структуры файловой системы.

Изменение файлов и каталогов, видимых пользователем, приводит к обновлению страницы и экстенста. Также происходят изменения для распределения экстентов, счетчиков ссылок и обратных указателей.

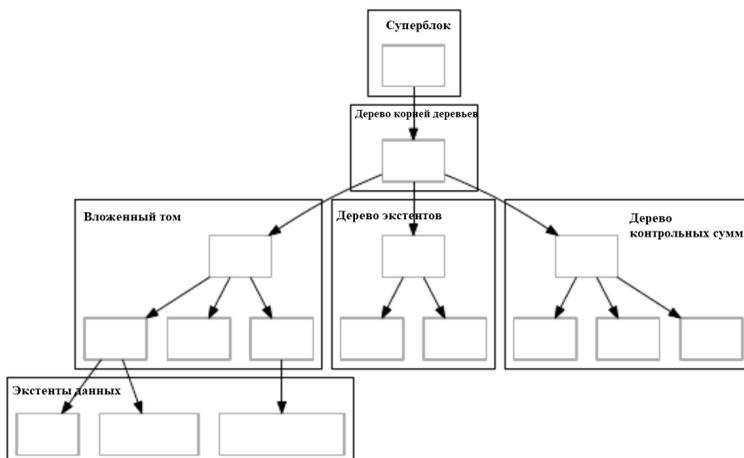


Рисунок 2 – Обобщенная схема В-деревьев, использующихся в файловой системе

Контрольные суммы данных и метаданных изменяются, эти обновления изменяют листья дерева контрольных сумм, вызывая волну изменений. Все эти модификации дерева фиксируются на самом верхнем уровне как новый корень в дереве корней дерева. Изменения накапливаются в памяти и после определенного промежутка времени или изменения достаточного количества страниц пакетно записываются в новые места на диске, образуя контрольную точку.

Такая система хранения информации позволяет осуществлять быстрый поиск нужных томов и секторов данных на носителях, а также с большой скоростью производить снимки системы и откат изменений в случае сбоев в работе компьютера, вызванных ошибками при перезаписи или некорректности различных данных, хранящихся на устройстве.

Однако при такой реализации файловой системы приходится решать проблемы с достаточно высокой сложностью операций записи, а также удаления файлов.

Другой недостаток заключается в поддержании непротиворечивости хранимой информации. В файловых системах непротиворечивость должна обеспечиваться путем транзакций – операций, имеющих такие свойства, как атомарность, согласованность, изоляция и долговечность. Запись в файловых системах, использующих В-деревья не атомарна, а значит целостность данных в них приходится обеспечивать различными системами восстановления от сбоев. Для этих целей при разработке файловых систем приходится применять дополнительные решения. Достаточно популярным решением является использование механизма копирования при записи (Copy on Write).

Суть данного механизма заключается в том, чтобы при изменении информации, записанной на некотором участке памяти, не перезаписывать данный участок, а лишь фиксировать изменения уже в новом месте на носителе информации. При таком подходе, в случае неудачного изменения данных, их можно легко восстановить, откатившись на более старую версию.

Система В-деревьев, хранящих ссылки на участки памяти, позволяет легко реализовать механизм копирования при записи в файловых системах, так как для этого достаточно создать новый указатель на изменения в файле и записать лишь новую его часть, а не перезаписывать файл целиком. Это также дает преимущества при использовании изменения одного участка памяти двумя или более различными процессами. Так как все изменения одного и того же

файла будут фиксироваться отдельно для каждого из таких процессов, а значит записываться в разные места на носителе, то между ними не может произойти конфликтов при обращении к данным.

Однако, у такого подхода есть свои недостатки. Один из них заключается в фрагментации данных. Поэтому в файловых системах, имеющих систему указателей, построенных на структурах В-деревьев и использующих копирование при записи, со временем может возникать потребность в процедуре дефрагментации.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Кормен, Т., Лейзерсон, Ч., Ривест, Р., Штайн, К. Алгоритмы: построение и анализ = Introduction to Algorithms / Под ред. И. В. Красикова. – 2-е изд. – М.: Вильямс, 2005. – 1296 с.

2. Ohad Rodeh, Josef Bacik, Chris Mason. BTRFS: The Linux B-tree Filesystem. Article in ACM Transactions on Storage, August 2013 [Электронный ресурс]. – Режим доступа: [https://www.researchgate.net/publication/262177144\\_BTRFS\\_The\\_linux\\_B-tree\\_filesystem](https://www.researchgate.net/publication/262177144_BTRFS_The_linux_B-tree_filesystem) (дата обращения 19.04.2022).

3. Официальный wiki BTRFS. Описание структур данных. [Электронный ресурс]. – Режим доступа: <https://btrfs.wiki.kernel.org/index.php/Trees>, (дата обращения 19.04.2022).

4. Sakis Kasampalis, Copy On Write Based File Systems Performance Analysis And Implementation, 2010. [Электронный ресурс]. – Режим доступа: <https://sakisk.me/files/copy-on-write-based-file-systems.pdf> (дата обращения 23.04.2022).

УДК 004.65

### **П.А. ВЛАСОВ, В.Ю. ГАГАРИНСКИЙ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПОВЫШЕНИЕ НАДЕЖНОСТИ ПЕРЕДАЧИ ТЕКСТОВЫХ ДАННЫХ ПО КАНАЛУ СВЯЗИ**

*Рассматриваются теоретические аспекты и направления повышения надежности передачи текстовых данных по каналу связи.*

Надежность канала связи - это свойство обеспечения связи, при сохранении во времени значений определенных показателей качества в заданных условиях эксплуатации. Она является основным

фактором, отражающим влияние на работоспособность системы связи, главным образом внутреннего системного фактора. Внутренний системный фактор случайных отказов техники, вызывается в процессе устаревания аппаратуры, а также дефектами технологии производства этой аппаратуры. Ошибки персонала также учитываются.

В основном в комплекс свойств надежности канала входят следующие его свойства:

- безотказность – сохранение рабочего состояния канала в течение заданного времени;
- устойчивость – свойство сохранения рабочего состояния в условиях действия помех;
- корректируемость – свойство, которое заключается в предупреждении и последующем обнаружении причин возникновения повреждений, сбоев, отказов и поддержании состояния работоспособности путем доработки, улучшения;
- защищенность – невозможность реализации посторонних вмешательств.

Сетевую надежность реально обеспечить путем применения определенного оборудования. Например, вносится избыточность в сетевую структуру для повышения ее готовности. Таким образом формируется отказоустойчивость. Обеспечение аппаратной и структурной надежности составляют комплексное решение обеспечения надежности сети в целом. Аппаратная надежность ставит задачу повышения надежности элементов, входящих в сеть. Это могут быть каналы передачи данных, сетевое оборудование программное обеспечение. Во втором случае обеспечивается функционирование сети, которое связано с передачей данных. Для анализа этого направления используют характеристики связности графов заранее описанных моделей сети. Отказ аппаратуры в ее узлах или же отказ каналов передачи данных может нарушить связность сети.

Развитие сложных компьютерных сетей (КС) государственного и корпоративного уровня является одним из способов повышения надежности. При этом возможно расположение сегментов КС в различных субъектах страны на довольно значимом расстоянии друг от друга. Не видится возможным создание отдельной телекоммуникационной сети для каждой КС и по техническим, и по экономическим причинам. В конечном итоге необходимо интегрирование с сетями связи общего пользования (ССОП). Можно говорить о том, что подавляющее большинство современных КС интегрированы с сетью Интернет. Этот фактор приводит к повышению риска выхода элементов сети из строя в результате воздействия преднамеренных и непреднамеренных помех.

Из-за сбоя большого количества элементов сети вполне возможным видится ее разрушение сети и дальнейшая невозможность выполнять информационный обмен.

Приоритетными видами обеспечения надежности являются: увеличение дополнительных ресурсов в сети передаваемых данных, анализ оценки надежности технологических структур еще на этапе проектирования, вариативные способы маршрутизации. Увеличение дополнительных ресурсов для резервирования канала на случай повышения числа сообщений относится к достаточно дорогим проектным решениям. Маршрутизация распределяет свободный трафик по различным каналам и узлам, тем самым нивелируя его рост.

Современные методы и средства для оценки передачи информационных потоков между элементами интегрированных компьютерных сетей (ИКС) позволяют дать оценку надежности узлов и пропускной способности каналов и линий связи по отношению к заранее известным воздействиям при конфигурации типовых нагрузок (имеется ввиду наличие схемы сети, а также информации о характеристиках трафика и используемом аппаратном и программном обеспечении). Так как интегрированные компьютерные сети имеют свою специфику (наличие значительного количества элементов ССОП) такие данные очень быстро теряют свою актуальность. Это существенно снижает достоверность результатов. Помимо прочего рассматривается способ оценки качества используемых систем, а также предоставляемых услуг за выбранный промежуток времени. В этом случае требуется сбор данных за определенный промежуток времени, который, к слову, измеряется месяцами. Из-за регулярной волатильности как структуры ИКС, так и используемых в ее работе аппаратных и программных методик, эти данные не могут быть интегрированы с нужным уровнем достоверности. Вдобавок способ соответствия реальной структуры ИКС регулярной решетке на основе среднеарифметического числа узлов существенно ослабляет модель сети. Случайные графы достаточно наглядно описывают сетевые структуры больших размеров, которые включают в себя десятки тысяч узлов. Размерность ИКС обычно составляет несколько тысяч узлов. Это делает невозможным описание их структур при помощи случайных графов.

В том случае, если при кодировании информации используются оптимальные коды, то при появлении даже одной ошибки все сообщение или его значительная часть или составляющая может быть искажена. Рассмотрим пример. Кодирование элементарных сообщений a2a3a5a1a4 источника сообщений осуществляется с использованием кодовой таблицы.

Таблица 1 – Кодирование элементарных сообщений

| Сообщения | Кодовое слово |
|-----------|---------------|
| a1        | 00            |
| a2        | 01            |
| a3        | 10            |
| a4        | 110           |
| a5        | 111           |

После этого закодированное сообщение имеет вид 011011100110. В случае ошибки в первом знаке будет принято сообщение 111011100110, которое декодируется в слово a5a2a4a2a3. Искажение сообщения в результате одной ошибки случается ввиду того, что одно кодовое слово переходит в другое в результате замены одного или нескольких знаков.

На практике требуется компромисс между экономностью кода и защитой от ошибок.

Сначала убирается так называемая "бесполезная" избыточность, после этого вносится "полезная" избыточность. Она помогает обнаруживать ошибки и исправлять их.

Общеизвестными методами борьбы с помехами являются следующие:

- передача в контексте;
- передача с переспросом;
- дублирование сообщений.

Разберем подробнее каждый из этих способов.

*Передача в контексте.* Это хорошо известный способ, с которым скорее всего сталкивался каждый, кто, пытаясь передать по телефону со слабой слышимостью чье-либо имя, называл вместо букв, которые его составляют, например, первые буквы имени. Правильному воспроизведению искаженного сообщения помогает знание его смыслового содержания.

*Передача с переспросом.* Это просьба повторить переданное сообщение или его часть еще раз, в том случае, когда имеется контакт с источником сообщений.

*Дублирование сообщений.* В этом случае все также просто. Чтобы быть правильно понятым, нужное сообщение повторяют.

Введение избыточности - общая во всех перечисленных способах процедура увеличения надежности. По сути необходимо увеличить так или иначе объем передаваемого сообщения, чтобы была возможность его правильно расшифровать при наличии искажений.

Отмечаем, что избыточность замедляет скоростной поток передачи сообщений, так как только часть передаваемого сообщения несет реальный интерес для получателя этого сообщения, а избыток, вводимый для защиты от шумовых сигналов не несет в себе пользы.

Принято выбирать только те варианты избыточности, которые обеспечивают наилучшую помехоустойчивость при самом минимальном увеличении объема передаваемого сообщения.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Красильников В.А. Акустика. Исторический очерк // Вологодская областная универсальная научная библиотека. 2012.
2. Быховский М.А. История развития техники радиосвязи и вещания, теории связи. Часть 1. Теория связи. – М.: Техносфера, 2006. – 376 с.
3. Расчёт показателей надёжности радиоэлектронных средств : учеб.-метод. пособие / С. М. Боровиков, И. Н. Цырельчук, Ф. Д. Троян ; под ред. С. М. Боровикова. – Минск : БГУИР, 2010. – 68 с. : ил. ISBN 978-985-488-480-6.

УДК 004.932

**А.А. ВЬЮГИНА, С.Н. БАРАНОВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ КОДИРОВАНИЯ БИТОВЫХ ПЛОСКОСТЕЙ ИЗОБРАЖЕНИЯ ПРИ ПЕРЕДАЧЕ ПО КАНАЛАМ СВЯЗИ

*Рассматривается возможность улучшение помехоустойчивости передачи ландшафтных изображений по каналам связи, которое заключается в исследовании битовых плоскостей изображения с наложением на каждую плоскость кода Хемминга*

С каждым годом увеличивается объем информации, которая передается между различными устройствами по сети. Большую часть данных составляют разного рода изображения, несущие различное количество информации. В таком случае возникает необходимость в ускорении передачи данных. При этом возможны появления искажений сигнала при передаче по линиям связи. Перечисленные проблемы образуют задачу по исследованию битового трафика, которая сводится к анализу возможностей уменьшения количества передаваемых дан-

ных и обеспечения их надежности к шумовым помехам на линиях связи.

При передаче любой информации осуществляется ее кодирование для снижения возможных искажений. Шумовая составляющая непостоянна и оказывает влияние на передающееся сообщения случайным образом. Однако на стороне источника сообщения можно четко определить: какая часть передаваемых данных несет наибольшее количество информации. Для изображений важными составляющими являются старшие биты каждого пикселя, в то время как младшие разряды дают лишь некоторые отличия в оттенке серого, который добавляет сглаженности и не накладывает сильного изменения при восстановлении изображения на принимающей стороне. Чтобы иметь возможность отследить изменения и предотвратить искажение информации разрядов, следует разделить изображение на битовые плоскости с целью дальнейшей обработки отдельно взятой матрицы разрядов.

Под битовыми плоскостями понимается разбиение яркости пикселя на отдельные биты и создание отдельной матрицы для каждого разряда [1]. При выполнении данного преобразования исходная матрица яркостей, состоящая из значений, хранимых в виде байта, увеличивается в 8 раз при переходе на хранение в виде бит (1).

$$\mathbf{G}_{N \times N} = \mathbf{G}_{(N \cdot 8) \times N} \quad (1)$$

Затем для каждой плоскости выбираются соответствующие значения матрицы в зависимости от номера битовой плоскости  $k$ . Количество битовых плоскостей зависит от количества бит, необходимых для хранения одного пикселя. В данном случае под один пиксель был использован 1 байт, соответственно количество битовых плоскостей  $k$  может принимать значения от 1 до 8. Формирование матрицы битовой плоскости (2):

$$\mathbf{B}_{N \times N}^k = \begin{pmatrix} g_{k,1} & g_{k+8,1} & \dots & g_{k+8 \cdot N,1} \\ \vdots & & \ddots & \vdots \\ g_{k,N} & g_{k+8,N} & \dots & g_{k+8 \cdot N,N} \end{pmatrix} \quad (2)$$

В итоге были получены срезы изображения по каждому из восьми разрядов. Пример перехода к битовым плоскостям для изображения размером 256x256 представлен на рисунке 1.

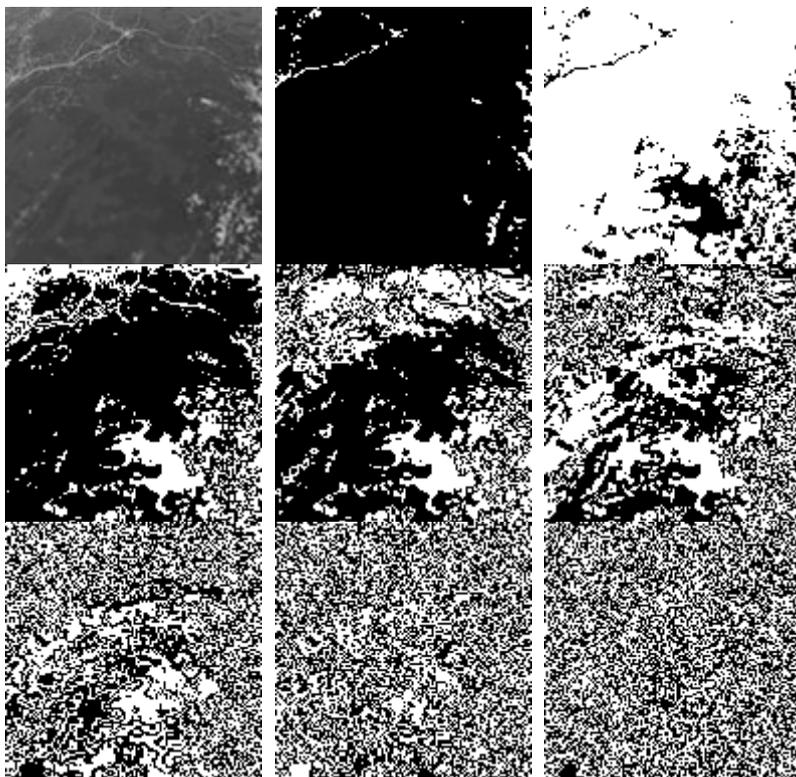


Рисунок 1 – Битовые плоскости изображения

Таким образом, можно предположить, что при передаче изображений по линиям связи основную опасность в искажении на принимаемой стороне представляет влияние шумов на старшие биты передаваемых данных. В таком случае для защиты от шумов при передаче информации следует использовать кодирование. Однако применение кода ко всей последовательности сильно утяжеляет цепочку данных и увеличивает время передачи.

Совокупность битовых плоскостей возможно собрать в сообщение в виде последовательности от матрицы со старшими разрядами в порядке убывания степени двойки.

При таком расположении бит по сравнению с исходным появляется большая вероятность попадания шума на значащий разряд. Однако сосредоточенность старших бит в начале последовательности позволяет наложить более сильную защиту только на часть всех данных

в сравнении с необходимостью защиты всей информации в исходной последовательности пикселей.

Наложение шумовой составляющей при рассмотрении битовых последовательностей можно представить как сумму двух матриц (3). Так как исходная последовательность представлена в битовом виде, то шум может быть представлен последовательностью 0 и 1, при этом сумма элементов двух матриц также будет оставаться в значениях 0 и 1.

$$\mathbf{A}_{M \times N} = \mathbf{B} + \mathbf{Z} = \begin{pmatrix} b_{1,1} + z_{1,1} & b_{2,1} + z_{2,1} & \dots & b_{M,1} + z_{M,1} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1,N} + z_{1,N} & b_{2,N} + z_{2,N} & \dots & b_{M,N} + z_{M,N} \end{pmatrix} \quad (3)$$

где  $\mathbf{B}$  – совокупная матрица битовых плоскостей, которые соединены от матрицы с большим разрядом к матрице с меньшим разрядом,  $\mathbf{Z}$  – матрица шумов.

Для кодирования тестовых изображений был выбран код Хэмминга. Его основная задача заключается в разбиении на классы кодируемой комбинации, каждая из которой представляет собой сумму по модулю 2 кода и образующих элементов данного класса [3]. Для исправления любой полученной ошибки необходимо определить класс смежности, каждому из которых соответствует некоторая контрольная последовательность символов. В двоичном коде при возникновении ошибки в определенном разряде опознавателя появляется 1.

В качестве эксперимента были взяты исходные сообщения в размере 8 и 16 бит и кодовые последовательности 4 и 5 бит соответственно. Единственным ограничением при использовании кода Хэмминга является то, что в сообщении будет обнаружено только одна ошибка [3]. В таком случае при увеличении шумовой составляющей существует вероятность искажения нескольких бит одного сообщения, а значит не срабатывание кода в соответствующей последовательности. При длине закодированного сообщения 12 бит выделяют 4 проверочных разряда: 1, 2, 4, 8. В таком случае группы проверочных разрядов формируются в соответствии с формулой 4.

$$\begin{aligned} a_1 &= a_2 \oplus a_5 \oplus a_7 \oplus a_9 \oplus a_{11} \\ a_2 &= a_3 \oplus a_6 \oplus a_7 \oplus a_{10} \oplus a_{11} \\ a_4 &= a_5 \oplus a_6 \oplus a_7 \oplus a_{12} \\ a_8 &= a_9 \oplus a_{10} \oplus a_{11} \oplus a_{12} \end{aligned} \quad (4)$$

Рисунок 2 демонстрирует различия между принимаемыми изображениями, которые прошли через модель линии передачи с наложением шумовой составляющей. Изображение 4.а показывает результат, который будет получен без использования любой кодовых комбинаций.

ций, в то время изображение 4.б и 4.в содержит код Хэмминга для 16 и 8 бит соответственно.

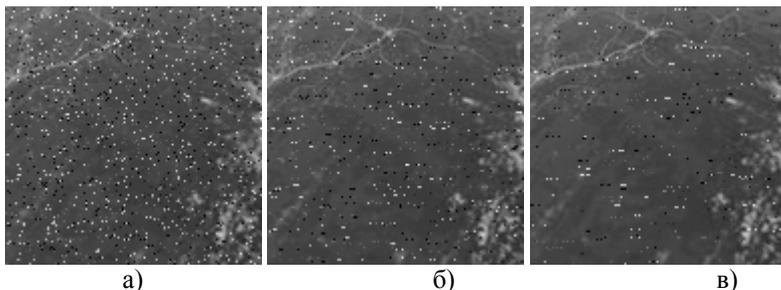


Рисунок 2 – Изображение с кодом Хэмминга: а) без кодирования, б) с кодом Хэмминга для 16 бит; в) с кодом Хэмминга для 8 бит

При таком подходе длина сообщения будет увеличена в 1,5 раза при использовании 4 бит кода на каждые 8 бит (для одной строки, содержащей 128 бит, последовательность будет 192 бита), при использовании 5 бит на каждые 16 бит увеличение будет меньше: на каждые 128 бит получится 168. Исходя из проведенного эксперимента и полученных результатов качества полученных изображений, можно сделать вывод, что кодирование 8 бит дает оптимальный результат по информативности полученного изображения.

Имея информацию о местонахождении значащих разрядов, можно направить на защиту такого фрагмента сообщения более сильный код. В рамках этой гипотезы было проведено моделирование передачи изображения с выбранным кодом Хэмминга для 8 бит, который применялся к 8, 6 и 4-м битовым плоскостям, остальные оставались в неизменном виде. Результаты представлены на рисунке 3.

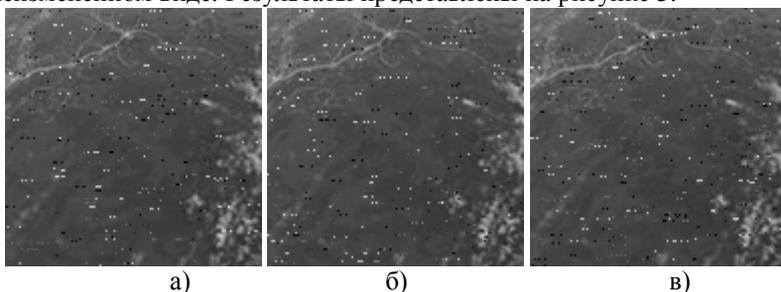


Рисунок 3 – Восстановленное изображение: а) с кодированием 8 плоскостей; б) с кодированием 6 плоскостей; в) с кодированием 4 плоскостей

Результаты проведенного эксперимента указаны в таблице 1. В итоге при уменьшении количества передаваемых бит ожидаемо возрастает вероятность восстановления изображения с ошибками. Однако информативность изображения от этого не сильно потеряна: протяженность дорог и другие элементы ландшафта не искажены.

Таблица 1 – Сравнительный анализ восстановленного изображений (размер 128x128)

| Количество кодируемых плоскостей | Процент ошибочных пикселей | Количество передаваемых бит |
|----------------------------------|----------------------------|-----------------------------|
| 8                                | 6.5 %                      | 196608 (128x1536)           |
| 6                                | 9.8 %                      | 180224 (128x1408)           |
| 4                                | 13.3 %                     | 163840 (128x1280)           |

Моделирование канала связи было произведено с помощью среды Matlab. Для этого были написаны функции, соответствующие следующим шагам:

1. Подготовка изображения к передаче, которая включает преобразование исходных данных в последовательность бит, пересортировка с учетом битовых плоскостей и кодирование.

2. Наложение шумовой составляющей с помощью готовой функции `imnoise()` с типом шума «соль и перец» [2].

3. Восстановление изображения, под которым подразумевается декодирование и переход к байтовому представлению пикселей.

Кодирование производилось на базе кода Хэмминга, для чего была разработаны 2 функции для сообщения длиной 8 и 16 бит. Алгоритм вычисления кода подробно описан в главе 2 [3].

Исследование проводилось на нескольких изображениях различного размера: от 32x32 до 256x256 пикселей. Шумовая составляющая, наложенная на исходную последовательность, варьировалась от 3 до 15%. Во всех случаях СКО не превышало 0,1 для кодированного изображения. Также стоит отметить, что при кодировании части битовых плоскостей, соответствующие значащим битам, результат не теряет информативности.

Проведенное исследование позволяет сделать вывод о возможности разбиения исходного сообщения на фрагменты, на которые можно наложить код различной мощности. Например, в случае с кодом Хэмминга выделять меньшую последовательность бит (в нашем эксперименте 8 бит), а на менее значащие разряды применять код для 16 бит или оставлять незакодированной. Кроме того, возможно применение каскадных кодов на часть всей последовательности, что позволит снизить влияние недостатка применяемого кода в единственном

виде (код Хэмминга позволяет выделить только одну ошибку в сообщении).

В результате проведенного исследования на основе кодирования битовой последовательности получены данные, позволяющие сделать вывод о возможности уменьшения объема сообщения при передаче по каналу связи. В таком случае дополнительная информация, под которой подразумевается кодовая последовательность, добавляется только для определенных фрагментов, несущих наиболее значимые данные. При этом максимально допустимая степень ошибки определяется конкретной задачей. Однако при этом не решается проблема избавления от артефактов при восстановлении сообщения. Применение каскадных кодов может позволить улучшить полученные результаты и является дальнейшим направлением для исследования.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений. Издание 4-е. Москва: Техносфера, 2018. – 1184 с.
2. Амос Гилат МАТЛАБ: Теория и практика. 5-е изд./ Пер. с англ. Смоленцев Н.К. – М.: ДМК Пресс, 2016. – 416с.
3. Костров Б.В. Технологии физического уровня передачи данных: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2017. – 224с.

УДК 004.45

### М.С. ГАГАРИНСКАЯ, Ю.А. ВАСЮКОВА

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### УВЕЛИЧЕНИЕ КАЧЕСТВА ПЕРЕДАЧИ ЗВУКА ПО КАНАЛУ СВЯЗИ

*Данная статья посвящена одному из методов повышения качества передачи звука по каналу связи, а также рассматриваются основные виды этих методов.*

В теории и практике технической надежности изделий термин наработка, как правило, широко используется для обозначения продолжительности работы изделия, выраженной в часах, циклах переключений или других единицах в зависимости от вида и функционального назначения изделия.

Под отказом понимается полная или частичная потеря работо-

способности изделия. Это становится возможным из-за отклонения одного или нескольких функциональных параметров от установленных норм, указанных в технической документации.

Под наработкой до отказа понимается весь период эксплуатации изделия с момента ввода в эксплуатацию (эксплуатации) до возникновения неисправностей. Надежность - комплексный признак, который в зависимости от назначения изделия и условий его использования, как правило, может использовать безаварийность, долговечность, сохраняемость и ремонтпригодность или, в некоторых случаях, определенные сочетания этих свойств.

Надежность канала связи – это свойство связи, сохраняющее во времени значения заданных показателей качества в заданных условиях эксплуатации. Именно надежность отражает влияние на работоспособность системы, прежде всего, внутреннего системного фактора - возможных отказов оборудования вследствие различных процессов старения оборудования, ошибок в технологии их производства или ошибок обслуживающего персонала соответственно.

Одним из методов повышения качества передачи звука по каналу связи является резервирование. Резервирование — универсальный принцип надежности, широко используемый в природе, технике и технологиях и впоследствии распространенный на другие стороны жизни человека.

Этот способ повышает характеристики надежности технических средств или может поддержать их на требуемом уровне за счет введения аппаратного резервирования посредством включения запасных (резервных) элементов и соединений. Как правило, дополнительно к минимуму, необходимому для выполнения заданных функций в данных условиях эксплуатации. Для достижения высокой надежности технических систем конструктивных, технологических и эксплуатационных мероприятий может быть недостаточно. В таких случаях применяется резервирование. Это особенно актуально для систем, у которых невозможно добиться требуемой надежности за счет повышения надежности их элементов.

К основным типам резервирования принято относить структурное, информационное и временное резервирование.

Структурное резервирование (в зарубежной литературе резервирование элементов) позволяет использовать резервные элементы транспортного средства. Суть этого вида резервирования заключается в том, что в минимально необходимый вариант системы вводятся другие элементы, узлы, устройства или элементы которых называются основными, или даже предусматривается использование нескольких идентичных систем вместо одной системы. При этом избыточные

компоненты, узлы, устройства и т.д. предназначены, в большинстве случаев, для выполнения эксплуатационных функций при выходе из строя соответствующих основных узлов, элементов и оборудования.

Информационное резервирование чаще всего предполагает использование избыточной информации. Простейшим примером реализации такого типа резервирования является многократная передача одного и того же сообщения по каналу связи. Можно привести другой пример – использование специальных кодов, возникающих в результате неисправностей и неполадок в работе оборудования. Здесь следует отметить, использование информационного резервирования предполагает также необходимость введения избыточных элементов.

Временное резервирование позволяет использовать избыточное время. В случае данного вида резервирования предполагается, что средство может продолжать работу после прерывания его деятельности из-за отказа путем его восстановления. При этом также предполагается, что на выполнение необходимых работ предоставляется время, чаще всего больше необходимого минимума.

Автономное резервирование является одним из общих вариантов. Оно заключается в использовании нескольких независимых объектов, выполняющих одну и ту же задачу. Каждое из этих устройств имеет собственный вход и выход и, в большинстве случаев, независимые источники питания. Примером объекта с автономным резервированием может быть набор устройств телемаркетинга. Они выполняют одну и ту же задачу, если каждое устройство имеет свои входные датчики, блоки записи (выхода) и источники энергии. Автономное резервирование обычно используется при проведении ответственных экспериментов в системах ответственного назначения.

Существует несколько методов резервирования промышленного Ethernet. *Метод агрегации линий связи* описан в стандарте IEEE 802.3. Этот метод использует два или более параллельных кабеля и порта для каждой линии связи, а объединение нескольких физических линий связи в один логический канал осуществляется с помощью протокола LinkAggregationControlProtocol (LACP). В этом случае группа (совокупность) линий связи и портов представлена одним логическим служебным интерфейсом с уникальным MAC-адресом. В протоколе LACP полные кадры Ethernet попеременно отправляются по параллельным линиям связи и объединяются в приемнике. Пропускная способность такого агрегированного канала прямо пропорциональна количеству физических линий. Если одна строка дает сбой, данные передаются поверх другой. Стандарт поддерживается многими производителями коммутаторов Ethernet.

*Метод резервирования.* Базовый Ethernet протокол STP (Spanning Tree Protocol), по-русски "протокол остового дерева" или "протокол связующего дерева", - протокол 2-го уровня модели OSI, который описан в стандарте IEEE 802.1D 1990 года. Первоначально протокол был использован для того, чтобы избежать петель в больших и сложных офисных сетях с мостами, которые могли иметь сложную запутанную топологию. С тех пор, как появился Ethernet этот протокол стал использоваться для резервирования сетей с коммутаторами. Цель STP протокола - конфигурация сети в виде дерева (т. е. без циклов) такая, что каждый узел сети (лист дерева) связан с корнем по пути с наименьшим временем доставки сообщений. Дерево сформировано путем отключения ветвей, которые могут образовывать физические (не логические) петли в сети. Именно так, при проектировании сети в нее могут быть добавлены избыточные ветви, цель которых - резервирование, которые будут логически отключены протоколом STP при формировании дерева сети. Недостатком STP протокола является часто недопустимо большое время перехода на резерв, а также невозможность резервирования связей между коммутатором и устройством-участником сети.

*Метод физического кольца.* Методы резервирования, основанные на расширенном протоколе RSTP, имеют слишком большое время переключения в режим ожидания (до 2 секунд). Но в современном мире многие приложения требуют сокращения этого времени до единиц миллисекунд (в робототехнике) или долей секунды (если речь идет о химических процессах). Поэтому некоторые компании разработали собственные нестандартные методы резервирования, которых в настоящее время насчитывается более 15. Эти методы основаны на использовании сети с круговой физической топологией. Одна из ветвей сети блокируется коммутатором. Из-за этого в нормальном режиме работы сеть приобретает топологию логической шины. После выхода из строя одной из ветвей мастер включает резервный порт. При этом подключается резервная ветка и переподключается сетевой граф. Работоспособность сети полностью восстанавливается. Одним из способов обнаружить сбой в сети является опрос, другим - отправка уведомления о сбое.

Наименьшее время переключения обеспечивает метод полного дублирования всей сети в целом. Второе его достоинство — сохранность при выходе из строя не только соединения между коммутаторами, но и самих коммутаторов, сетевых портов устройств и линий связи устройств с коммутатором. Недостаток - это высокая стоимость, поскольку метод предполагает использование всех сетевых устройств в

двойном количестве. Тип полного резервирования — текущее резервирование сети и конечных устройств. В этом случае получаются две полностью независимые системы автоматизации и резервируется не только сетевое оборудование, но и вся система в целом. К разновидности полного резервирования можно отнести одновременное резервирование сети и оконечных устройств. В таком случае получаются две полностью независимые системы автоматизации и резервированным получается вся система в целом. Для выбора одной из сетей и обнаружения отказа используются необходимые средства диагностики, которые, как правило, реализуются на основе стандарта IEEE 802.1p/Q.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Красильников В.А. Акустика. Исторический очерк [Электронный ресурс]. – Режим доступа: <https://www.booksite.ru/fulltext/1/001/008/008/737.htm> (дата обращения 11.03.2022).
2. Стретт Дж. В. Теория звука. М., 1955.
3. Дмитриев В.Л. Элементы линейной акустики: Учебное пособие. Уфа, 2008.
4. Быховский М.А. История электросвязи и радиотехники. Ч.1. Теория связи. – М.:Техносфера, 2006. – 376 с.
5. Быховский М.А. Развитие телекоммуникаций: на пути к информационному обществу, истории телеграфа, телефона и радио до начала XX века, 2012.
6. Расчёт показателей надёжности радиоэлектронных средств : учеб.-метод. пособие / С. М. Боровиков, И. Н. Цырельчук, Ф. Д. Троян ; под ред. С. М. Боровикова. – Минск : БГУИР, 2010. – 68 с.

УДК 044.021

**Д.В. ГАЛЬЦОВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **АЛГОРИТМЫ, ИСПОЛЬЗУЮЩИЕ ДЕФОРМАЦИЮ ТРАСС СОЕДИНЕНИЙ**

*В данной статье рассматриваются этапы создания алгоритма “гибкой” трассировки.*

В волновые, лучевые и в меньшей мере в канальные алгоритмы заложен последовательный принцип трассировки, что предполагает

оптимальное построение каждой отдельной трассы без прогноза на прокладку последующих соединений на монтажной плоскости. Одним из главных недостатков последовательных методов трассировки является жесткая фиксация на коммутационном поле ранее проложенных соединений, приводящая к ухудшению качества окончательного результата. В наибольшей степени этот недостаток проявляется при трассировке однослойных схем соединений.

В работах Г. Э. Широ были предложены способы повышения эффективности последовательных процедур трассировки. Один из них заключается в том, что ранее построенные конфигурации соединений могут деформироваться последующими конфигурациями. Это достигается за счет того, что на коммутационном поле (КП) выделяются области, в пределах которых проводники могут занимать любое место и сдвигаться последующими проводниками, т.е. трассы соединений жестко не фиксируются. Так, например, для печатной платы с фиксированной и регулярно расположенной сетью переходных отверстий трасса очередного соединения прокладывается по укрупненным ячейкам или зонам, расположенным между рядами переходов, как показано на рисунке 1.

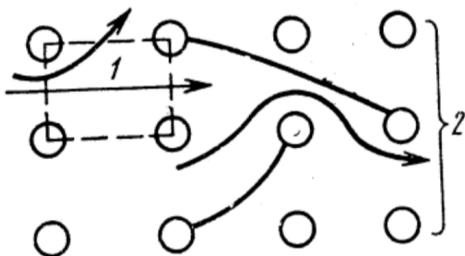


Рисунок 1 – Выделение областей для трассировки: 1 - укрупненная ячейка, 2 - зона

Другой способ состоит в стирании тех фрагментов ранее проложенных соединений, которые мешают трассировке очередного соединения. Для построения соединения применяется один из вариантов волнового алгоритма. Критерий оптимизации наряду с числом пересечений учитывает также длину прокладываемых соединений. Поиск и устранение мешающих соединений реализуются следующим образом:

1. при построении очередного соединения сначала допускается пересечение ранее проложенных соединений;
2. стираются те фрагменты проложенных соединений, которые с ним пересекаются;

3. указанные фрагменты включаются в список непроведенных соединений и подлежат последующей трассировке.

Процесс, таким образом, имеет итерационный характер. Возможные «зацикливания» устраняются соответствующим выбором критериев и степени их важности. Если же процесс не приводит к получению схемы без пересечений, то он существенно уменьшает число непроведенных соединений

Дальнейшее развитие идей «нежесткой» трассировки соединений выполнено Р.П. Базилевичем и др. В качестве модели КП используется система граней, через каждую из которых могут проходить несколько соединений. Форма граней может быть произвольной, как показано на рисунке 2. Вершинами граней являются выводы элементов, угловые точки запрещенных зон и краев поля и т.п. В отличие от дискретного рабочего поля (ДРП) полученная модель названа аналоговым рабочим полем (АРП). Такое представление оказывается удобным для реализации процесса трассировки в конструкциях нерегулярного типа с разногабаритными элементами. Как правило, это печатный платы аналоговых или цифровых устройств.

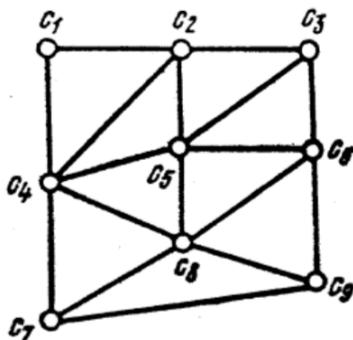


Рисунок 2 – Разбиение плоскости на треугольные грани

Для проведения соединений на АРП используется волновой алгоритм. Волна распространяется от некоторой вершины-источника через множество смежных граней, точнее через их ребра, до вершины-цели. При рассмотрении очередной грани учитываются текущие пропускные способности ее ребер и углов, т.е. максимальное число соединений, которые могут быть проведены через соответствующие ребра и углы.

Соседними к некоторому ребру  $a$ , вошедшему в очередной фронт волны, являются те ребра грани, которые достижимы из ребра  $a$  без пересечений (рисунок 3).

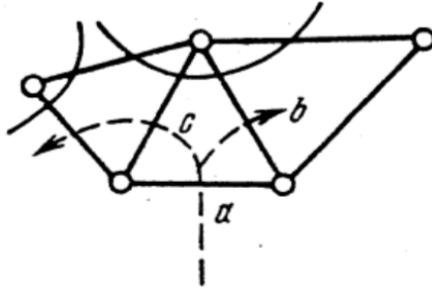


Рисунок 3 – Распространение волны

Если пропускная способность соседнего ребра исчерпана, оно считается запрещенным для распространения волны. После проведения пути осуществляется уменьшение пропускных способностей тех ребер, через которые прошла трасса пути.

Хотя описываемый метод и является последовательным, в процессе трассировки отсутствует жесткая фиксация отдельных проводников, благодаря чему создаются лучшие условия для проведения последующих соединений. Также представление монтажной плоскости в виде системы граней оказывается очень удобным для конструкций, использующих разногабаритные элементы. Однако главным недостатком «гибкой» трассировки является высокая вычислительная сложность алгоритма  $\approx O(n)^3 - O(n)^4$ , что приводит к большим затратам памяти ЭВМ и времени на получение решения.

В заключение можно сказать, что несмотря на то, что «гибкая» трассировка исключает возможность «жесткой» фиксации ранее проложенных трасс, создавая благоприятные условия для прокладки последующих соединений, широкого распространения она не получила. Гибкость модели говорит о том, что она меньше фиксирует метрические ограничения, следовательно, значительно сокращается пространство решений и поиск «хороших» решений становится легче, но труднее обеспечить соблюдение указанных ограничений.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. В. М. Куречик. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для вузов. – М.: Радио и связь. 1990. – 352 с.
2. В. А. Селютин. Машинное конструирование электронных устройств. – Москва: Сов. радио, 1977. – 383 с.

3. Широ Г. Э. Метод проектирования печатного монтажа, основанный на эвристических принципах. – В кн.: Методы разработки схем и конструкций цифровых систем. ЛДНТП, 1967.

4. Широ Г. Э., Лапидус Л. И. Метод трассировки печатных соединений. – В кн.: Применение вычислительных машин для проектирования цифровых устройств. Под ред. Н. Я. Матюхина. М., «Сов. радио», 1968, с. 199-215.

5. Базилевич Р. П., Парканий И. В., Стороженко Н. Ф., Ясеницкий О. З. Особенности использования аналогового рабочего поля для машинной трассировки межсоединений. – В кн.: Вычислительная техника. Т. VI/ политехн. Ин-т, 1974, с. 103-106.

УДК 044.021

**Д.В. ГАЛЬЦОВА**

Рязанский государственный радиотехнический университет  
имени Уткина В.Ф.

### **АЛГОРИТМ ТРАССИРОВКИ ПЕЧАТНЫХ СОЕДИНЕНИЙ НА ОСНОВЕ ПРЕДСТАВЛЕНИЯ О КАНАЛАХ**

*В данной статье рассматривается принцип канальной трассировки, а также алгоритм трассировки, предложенный Лазаревой Т.С.*

Алгоритмы трассировки, основанные на волновом принципе, входят в класс локально-оптимальных алгоритмов. В них трассировка осуществляется последовательно, т.е. сначала проводится трасса одного соединения или цепи, затем другого и т. д. При трассировке очередного соединения или цепи определяется трасса с минимальным значением функции цели, которой может быть либо длина, либо число используемых магистралей, либо некоторый комплексный показатель.

Основной недостаток локально-оптимальных алгоритмов заключается в том, что при трассировке очередного соединения нет никаких оснований для проведения неоптимальной (по выбранному критерию) трассы. Таким образом, при выборе критерия необходимо учитывать условия для проведения оставшихся соединений. Это и стремление уменьшить время вычислений привело к созданию алгоритмов, основанных на канальном представлении монтажного пространства. Кроме того, на некоторых этапах выполнения этих алгоритмов рассматривается взаимное расположение нескольких трасс.



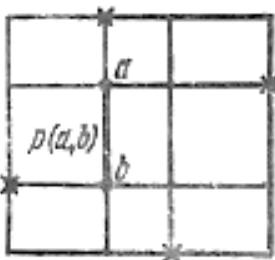


Рисунок 2 – Сеть каналов цепи

На поле двухслойной печатной платы каждому горизонтальному (вертикальному) ряду контактов модулей интегральных схем ставится в соответствие совокупность горизонтальных (вертикальных) магистралей, по каждой из которых возможно проведение печатного проводника к контактам данного ряда. Совокупность таких магистралей называется каналом, а их число – пропускной способностью канала.

Для всех цепей строятся ортогональные загрузки каналов. Затем для каждой цепи на ее ортогональной сети строится минимальное связывающее дерево методом волны. При построении дерева минимизируется длина его ветвей и обеспечивается равномерная загрузка каналов с учетом еще не трассированных цепей. Это достигается присвоением каждому ребру  $(a, b)$  исследуемой сети весового коэффициента:

$$p(a, b) = \frac{\max_{(a, b)} \beta_{is}}{\min_{(a, b)} \gamma_{is}} + kl(a, b),$$

где  $l(a, b)$  – длина соответствующего участка канала,  $k$  – коэффициент ( $0 \leq k \leq 1$ ).

Коэффициентом загрузки канала  $i$  на участке  $s$  назовем число проводников, использующих этот участок, и обозначим его  $\beta_{is}$ .

Текущей пропускной способностью  $\gamma_{is}$  канала  $i$  на участке  $s$  назовем число проводников, которые еще можно проложить на этом участке.

Текущей пропускной способностью  $\gamma_{is}$  канала  $i$  на участке  $s$  назовем число проводников, которые еще можно проложить на этом участке. На рисунке 3 показано распределение проводников в горизонтальном канале, а также изменение его загрузки и пропускной способности на отдельных участках. Предполагается, что начальная пропускная способность канала равна четырем.

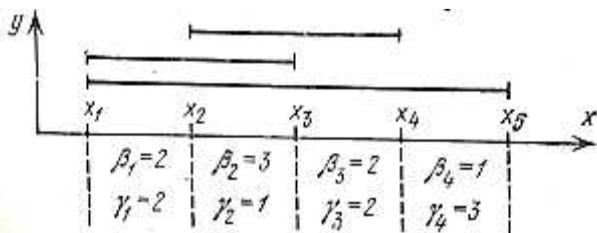


Рисунок 3 – изменение загрузки канала

После построения связывающих деревьев для всех цепей выполняется распределение проводников внутри каналов по магистралям, обеспечивающее минимальное число межслойных переходов.

Дальнейшее развитие алгоритма Лазаревой позволяет уменьшить число углов при построении дерева методом волны и улучшить процесс распределения проводников внутри каналов по магистралям.

В заключение можно сказать, что рассматриваемый алгоритм не гарантирует нахождение пути печатного соединения, если даже такой путь и существует, что резко повышает (до 30% и более) процент автоматически неразводимых соединений. Поэтому на заключительной стадии трассировки необходимо применять тот же волновой алгоритм, так как «ручная» доводка более 10% перемычек весьма трудоемка и в большинстве случаев неосуществима. Однако неволновые алгоритмы позволяют резко сократить затраты времени на проведение простейших по конфигурации соединений в условиях относительно свободного монтажного поля, что является их основным достоинством.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. В. М. Куречик. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для вузов. – М.: Радио и связь, 1990. – 352 с.
2. В. А. Селютин. Машинное конструирование электронных устройств. – Москва: Сов. радио, 1977. – 383 с.
3. Б. И. Бахтин. Автоматизация в проектировании и производстве печатных плат радиоэлектронной аппаратуры. –Л.: Энергия. Ленингр. отд-ние, 1979. –120 с.
4. Абрайтис Л. Б., Шейнаукас Р. И., Жилевичюс В. А. Автоматизация проектирования ЭВМ: Автоматизированное техническое проектирование конструктивных узлов цифровых устройств/ Под ред. Л. Б. Абрайтиса. – М.: Сов. Радио, 1978. – 272с.

5. Лазарева Т. С. Алгоритм трассировки печатных соединений на основе представления о каналах. – «Автоматика и вычислительная техника», Рига, 1969, №5.

УДК 681.3.06:51

**В.С. ГОРИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### ЗАДАЧА РАСПРЕДЕЛЕНИЯ ПЕЧАТНЫХ ПРОВОДНИКОВ В ОГРАНИЧЕННОМ ПРОСТРАНСТВЕ

*Рассматриваются формальная постановка и метод решения задачи определения расположения проводников заданной конфигурации в монтажном пространстве. Описывается способ решения задачи методом ветвей и границ.*

Предположим, что выводы двух рядом стоящих элементов **I** и **II** с односторонним расположением выводов необходимо соединить печатными проводниками, прокладываемыми слева (**A**) и справа (**B**) от элементов (рисунок 1).

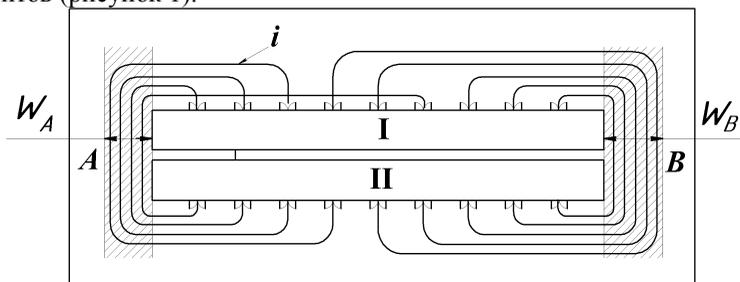


Рисунок 1 – Расположение проводников в монтажном пространстве

Для прокладки проводников выделены участки печатной платы шириной  $W_A$  и  $W_B$ . Длину  $i$ -го проводника, если его проложить слева от элементов, обозначим через  $l_i^A$ , а справа –  $l_i^B$  ( $i = 1, 2, \dots, n$ ). Известна также ширина отдельных проводников  $s_i$  ( $i = 1, 2, \dots, n$ ).

Требуется определить, с какой стороны (**A** или **B**) следует прокладывать конкретные проводники, чтобы суммарная длина их была минимальной и они умещались на выделенных участках.

Подобную задачу можно трактовать и по-другому, например, так.

Населенный пункт (город) разделен рекой (рисунок 2). Связь между противоположными частями города осуществляется двумя мостами, имеющими ограниченную пропускную способность  $W_A$  и  $W_B$  (например, по габаритам, грузоподъемности и т.п.). Склады продукции  $R_i$  и их потребители  $P_i$  ( $i = 1, 2, \dots, n$ ) находятся на противоположных берегах.

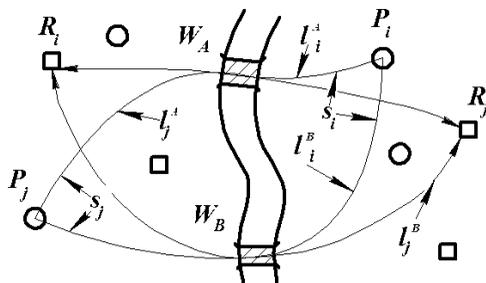


Рисунок 2 – Задача о двух мостах

Известно, что перевозка груза от  $i$ -го склада к потребителю равна  $l_i^A$  если дорога будет пролегать через мост А, и  $l_i^B$  если через мост В. Масса перевозимого груза из  $R_i$  в  $P_i$  составляет  $s_i$  единиц. Требуется так спланировать перевозки, чтобы суммарная длина перевозок была минимальна и при этом не нарушались ограничения, например, на суммарный объем перевозимого груза.

Сформулированная задача существенно отличается от так называемой «транспортной задачи» [1], так как, во-первых, перевозимый груз неоднороден, а, во-вторых, поставщики и потребители заранее известны.

#### *Дадим формальную постановку задачи.*

Обозначим через  $S = \|s_i\|$ ,  $i = 1, 2, \dots, n$ , матрицу-строку ширины проводников и введем булевы переменные  $y_i$  ( $i = 1, 2, \dots, n$ ), значение которых будет равно единице, если  $i$ -й проводник прокладывается через сторону А, и нулю, если через сторону В (рис. 1).

Задачу нахождения оптимального распределения проводников сформулируем так:

*Минимизировать:*

$$L = \sum_{i=1}^n [l_i^A \cdot y_i + l_i^B \cdot (1 - y_i)] \quad (1)$$

*при ограничениях:*

$$\sum_{i=1}^n [s_i \cdot y_i] \leq W_A \quad (2)$$

$$\sum_{i=1}^n [s_i \cdot (1 - y_i)] \leq W_B \quad (3)$$

$$s_i \geq 0; y_i = \{0, 1\}; i = 1, 2, \dots, n. \quad (4)$$

Здесь целевая функция (1) равна суммарной длине проводников, а ограничения (2) и (3) не позволяют превысить ширину участков платы **A** и **B**, выделенных для прокладки проводников, соответственно.

Следует отметить, что необходимым условием существования решения задачи будет выполнение неравенства:

$$W_A + W_B \geq \sum_{i=1}^n s_i \quad (5)$$

Сформулированная задача относится к задачам линейного целочисленного программирования. Для ее решения воспользуемся комбинаторным методом ветвей и границ [2].

Решение рассмотрим на примере.

Из графической модели задачи (рисунок 3) видно, что сумма длин любого двухконтактного проводника при распределении его в левый и правый участок одинакова:

$$l_i^A + l_i^B = \text{const}; i = 1, 2, \dots, n. \quad (6)$$

Однако, если  $j$ -й проводник имеет 3 и более контактов, то в качестве  $l_i^A$  и  $l_i^B$  необходимо брать отрезки провода, распределяемые в выделенные участки платы.

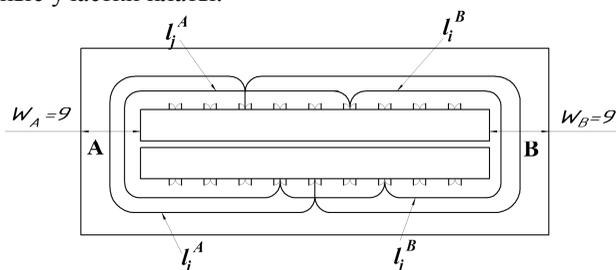


Рисунок 3 – Графическая модель задачи

Пусть  $W_A = W_B = 9$ .

Предположим, что ширина проводников известна, а длины их вычислены и приведены в таблице 1.

Таблица 1 – Исходное задание для решения задачи

|         |          |          |          |          |          |          |
|---------|----------|----------|----------|----------|----------|----------|
| $i$     | 1        | 2        | 3        | 4        | 5        | 6        |
| $l_i^A$ | <u>5</u> | <u>8</u> | 16       | <u>2</u> | 14       | 18       |
| $l_i^B$ | 15       | 12       | <u>4</u> | 18       | <u>6</u> | <u>2</u> |
| $s_i$   | 3        | 5        | 4        | 2        | 3        | 1        |

Проверим необходимое условие существования решения задачи

$$(5): \sum_{i=1}^n s_i = 3 + 5 + 4 + 2 + 3 + 1 = 18 = 9 \cdot 2 - \text{условие выполняется.}$$

За нижнюю границу целевой функции примем сумму минимальных элементов из второй и третьей строк таблицы без учета ограничений (2) и (3), т.е. будем считать, что проводники прокладываются по кратчайшим путям (см. табл. 1):

$$L_{min} = L_0 = 8 + 5 + 6 + 2 + 4 + 2 = 27.$$

Обозначим через  $G_0$  – множество всех решений задачи (в данном случае как допустимых, так и недопустимых) а через  $Q_0^A$  и  $Q_0^B$  – множество индексов проводников, прокладываемых по сторонам  $A$  и  $B$  соответственно. Тогда:  $Q_0^A = \{1, 2, 4\}$ ;  $Q_0^B = \{3, 5, 6\}$ . Заметим, что  $Q_0^A \cap Q_0^B = \emptyset$ . Будем считать,  $G_0 = Q_0^A \cup Q_0^B$ , а оценку  $\zeta(G_0) = L_{min} = 27$ .

Так как ограничение (2) не выполняется ( $\sum_{i=1}^n s_i \cdot y_i = 10 > W_A$ ), то будем разбивать множество  $G_0$  на два подмножества  $G_1^A$  и  $G_1^B$ , оставив в одном из них неизменные подмножества  $Q_{11}^A \equiv Q_0^A$  и  $Q_{11}^B \equiv Q_0^B$ , закрепив в них один из проводников, например,  $i$ , а в другое включим подмножества  $Q_{12}^A$  и  $Q_{12}^B$ , переставив в них выбранный проводник  $i$ , т.е.  $Q_{12}^A \equiv Q_0^A \setminus i^A$ , а  $Q_{12}^B \equiv Q_0^B \cup i^B$ .

Здесь через  $i^A$  и  $i^B$  обозначен индекс  $i$ -го провода, закрепленный за соответствующим подмножеством.

Правило выбора проводников для переноса в другое подмножество определим следующим образом: в первую очередь будем переносить из одного подмножества с недопустимым решением в другое проводник с минимальной разницей в длине при его перестановке. Это можно объяснить тем, что при такой перестановке нижняя граница целевой функции будет возрастать с меньшей скоростью. Можно также переносить такой проводник, который сразу обеспечит выполнение ограничений (2) и (3) задачи (если его удастся найти), но для этого придется отказаться от общей схемы решения и на каждом шаге реализовывать процедуру поиска такого проводника.

Итак, сначала попробуем перенести проводник с индексом 2. Получим:  $Q_{12}^A = Q_0^A \setminus 2^A$ ;  $Q_{12}^B = Q_0^B \cup 2^B$

$$G_1^A = Q_{12}^A \cup Q_{12}^B = \{1, 4\} \cup \{2, 3, 5, 6\}$$

$$G_1^B = Q_{11}^A \cup Q_{11}^B = \{1, 2, 4\} \cup \{3, 5, 6\}.$$

Жирным шрифтом здесь выделены индексы проводников, участвующих в ветвлении. В дальнейших перестановках рассматриваемой ветви они участвовать не будут.

Оценки полученных подмножеств составят:

$$\zeta(G_1^1) = 31; \zeta(G_1^2) = 27.$$

Проверяем ограничения для  $G_1^1: W_1^A = 5$ ; а  $W_1^B = 13$ ; для  $G_1^2: W_2^A = 10$ ; а  $W_2^B = 8$ . Ограничения не выполняются, следовательно, процесс решения необходимо продолжить.

Результат первого ветвления представлен на рисунке 4.

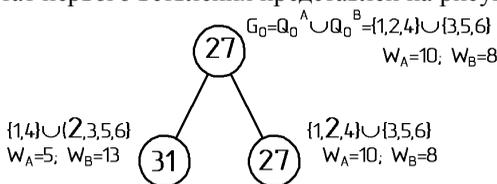


Рисунок 4 – Первое ветвление задачи

Из двух вершин с оценками  $\zeta(G_1^1) = 31$  и  $\zeta(G_1^2) = 27$  выбираем вершину с меньшей оценкой. Так как ограничения здесь не выполняются для стороны **A**, то следующим кандидатом на исключение из множества  $Q_{1,1}^A$  могут быть проводники с индексами **1** или **4**.

Согласно сформулированного выше правила разбиения возьмем 1-й проводник.

$$Q_{2,1}^A \equiv Q_{1,1}^A \text{ и } Q_{2,1}^B \equiv Q_{1,1}^B; Q_{2,2}^A = Q_{1,1}^A \setminus 1^A; Q_{2,2}^B = Q_{1,1}^B \cup 1^B.$$

Множество решений задачи:

$$G_2^1 = Q_{2,2}^A \cup Q_{2,2}^B = \{2, 4\} \cup \{1, 3, 5, 6\} \text{ и } G_2^2 = Q_{2,1}^A \cup Q_{2,1}^B = \{1, 2, 4\} \cup \{3, 5, 6\}.$$

Оценки полученных подмножеств составят:

$$\zeta(G_2^1) = 37; (G_2^2) = 27.$$

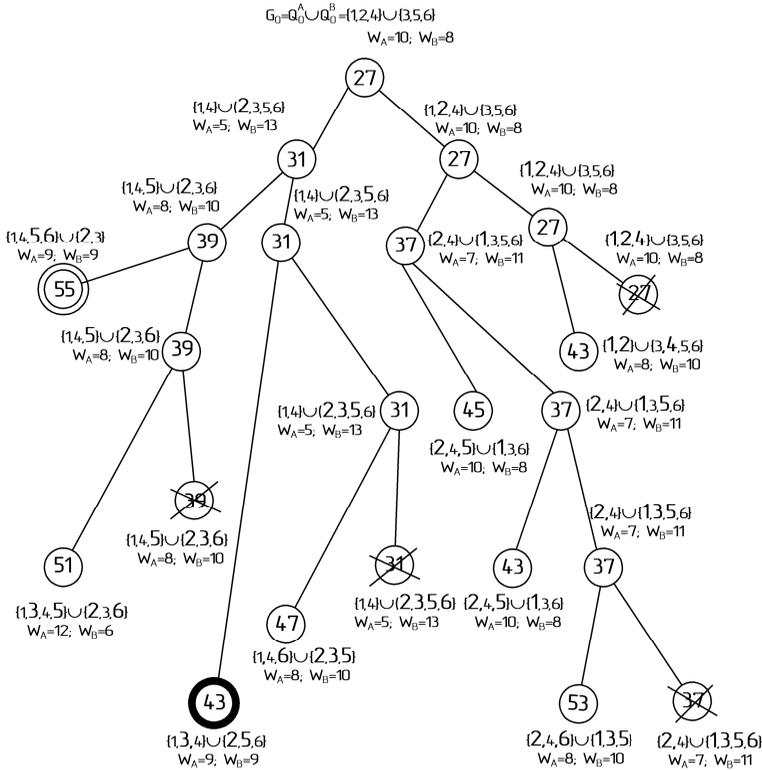
Проверяем ограничения для  $G_2^1: W_2^A = 7$ , а  $W_2^B = 11$ , для  $G_2^2: W_2^A = 10$ , а  $W_2^B = 8$ . Ограничения не выполняются, следовательно, процесс решения необходимо продолжить.

На следующем шаге получим  $G_3^1 = Q_{3,1}^A \cup Q_{3,1}^B = \{1, 2\} \cup \{3, 4, 5, 6\}$  и  $G_3^2 = Q_{3,2}^A \cup Q_{3,2}^B = \{1, 2, 4\} \cup \{3, 5, 6\}$  с оценками  $\zeta(G_3^1) = 46$ ;  $(G_3^2) = 27$ . Ограничения опять не выполняются, однако ветвить вершину  $G_3^2$  с оценкой 27 больше нельзя, так как переставлять проводники из множества  $Q_{3,2}^A$  в множество  $Q_{3,2}^B$  мы не можем. Эта вершина исключается из рассмотрения.

Следующей вершиной, подвергаемой разбиению, будет вершина  $G_1^1$  с наименьшей оценкой  $\zeta(G_1^1) = 31$ .

Дальнейшее решение показано на рисунке 5, на котором двойным кружком выделено допустимое решение (с оценкой  $\zeta(G_5^4) = 55$ ), а жирным кружком –  $\zeta(G_5^4)_{opt} = 43$  – оптимальное.

Оптимальной, по определению, является вершина (и соответствующее ей множество  $G_k^*$ ), которая соответствует допустимому решению и имеет оценку, не превышающую оценок других, еще не разветвленных вершин (т.е. которая быстрее всех «пришла к цели»).



Оценим трудоемкость решения. Очевидно, чтобы решить задачу методом полного перебора, необходимо рассмотреть  $6! = 720$  вариантов решения. В нашем случае применение метода ветвей и границ позволило уменьшить число рассматриваемых вариантов в  $720/22=32,7$  раз.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Пантелеев, А.В. Методы оптимизации в примерах и задачах задач: Учебное пособие / А.В. Пантелеев, Т.А. Легова. – СПб.: Лань, 2015. – 512 с.
2. Таха, Хемди А. Введение в исследование операций, 7-е издание.: Пер. с англ. – М.: Издательский дом "

УДК 621.373.826

**А.С. ГУСЕВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## ВЫБОР КОНСТРУКТИВНЫХ ПАРАМЕТРОВ АКТИВНОГО ЭЛЕМЕНТА ДЛЯ АППАРАТА ЛАЗЕРНОГО ОБЛУЧЕНИЯ КРОВИ

*Рассматриваются конструктивные параметры газового лазера и их влияние на выходные характеристики.*

В настоящее время лазеры повсеместно используются во многих отраслях человеческой деятельности. Лазеры стали стандартными инструментами в различных областях применения. Лазерные указки выделяют точки презентации в лекционных залах, а лазерные указатели направляют умные бомбы к их целям. Лазеры сваривают лезвия бритв, пишут узоры на объектах на производственных линиях, не прикасаясь к ним, удаляют нежелательные волосы и отбеливают татуировки. Лазерные дальномеры в космических зондах профилировали поверхности Марса и астероида Эрос беспрецедентной детализацией.

Лазер – это аббревиатура от «усиление света за счет стимулированного излучения». В физическом смысле слова лазер – это устройство, которое стимулирует атомы или молекулы излучать свет на определенных длинах волн и усиливает этот свет, обычно производя очень узкий пучок излучения. Для использования лазеров на практике требуется управлять различными его характеристиками, например, интенсивностью, мощностью или частотой лазерного излучения, положением луча во времени и тд. Управление лазерным излучением, в широком смысле, предполагает получение такого излучения, которое имеет перестраиваемые в каком-то диапазоне значений спектральные, энергетические, временные и пространственные характеристики.

Большую роль лазеры играют в медицине. Такая процедура как внутривенное лазерное облучение крови или ВЛОК заслуживает особого внимания. Внутривенное лазерное облучение крови – один из методов квантовой терапии крови, который широко используется отечественной и зарубежной практической медициной. В основе метода лежит метод облучения крови непосредственно в сосудистом русле через оптический волновод, вводимый обычно в любую легкодоступную вену. Источником излучения является лазерный аппарат, излучающий на длине волны 630 нм. К излучению данной длины чувствительны фоторецепторы, расположенные на поверхности клеток крови. Возбуждение и активация этих фоторецепторов приводит к запуску целого ряда важных биохимических реакций с развитием лечебного эффекта.

Для внутривенного лазерного облучения крови используется узкоспециализированная аппаратура, что обусловлено, необходимостью выполнять специальные санитарно-гигиенические требования, которые предъявляются к процедурным кабинетам. В качестве технического обеспечения ВЛОК можно применять гелий-неоновые лазеры (ЛГН). В таких установках конструкция активного элемента лазерного излучателя состоит из двух параллельно установленных зеркал. Для эффективного лечения онкологических заболеваний необходимо достигнуть оптимальной величины плотности потока лазерного излучения, а также рассчитать допуск, с которым будет производиться стабилизация лазерного излучения. Эти значения выходных параметров определяются номинальными значениями и допусками на конструктивные параметры элементов оптической системы, определяемые конструктором и технологией их изготовления, которые приведены в таблице 1.

В качестве выходной характеристики активного элемента лазерного аппарата определена плотность потока лазерного излучения, которая рассчитывается следующим образом:

$$E^p = 4.09^{N3} \times 0.99^{(d_1+d_2)} \times 0.99^{N4} \times 0.98^{N5} \times P_{\text{вх}} [R_1^2 (n_2 - 1)^2 / \pi D_{\text{вх}}^2 R_2^2 (n_1 - 1)^2]; \quad (1)$$

где,  $N3$  – число отражающих поверхностей;

$N4$  – число поверхностей, просветленных физическим методом гидролизаии;

$N5$  – число поверхностей, просветленных химическим методом;

$d_1, d_2$  – толщина по оси первого и второго зеркала соответственно;

$D_{\text{вх}}$  – диаметр лазерного пучка (плазмы) в оптической системе;

$R_1, R_2$  – радиусы кривизны первого и второго зеркала;  
 $n_1, n_2$  – показатели отражения/пропускания света соответствующими зеркалами;

$P_{\text{вх}}$  – мощность лазерный потока в оптической системе, определяемая высоковольтным источником питания.

Погрешности производства влияют на точность функционирования аппарата. Исследуем их влияние. Дифференцируя уравнение (1) как сложную функцию по переменным и заменяя дифференциалы функции малыми конечными приращениями, определим коэффициенты чувствительности и изменение допуска (погрешности) выходного параметра  $E$  в зависимости от допусков (погрешностей) на конструктивные параметры изготовленных комплектующих ( $R_1, R_2, n_1, n_2, d_1, d_2$ ) элементов. Подставив значения параметров из таблицы 1 в уравнения, получим:

$$\Delta E_{R_1} = 2 \times \delta R_1 / R_1 = 2 \times 0.99 / 33.65 = 0.0688;$$

$$\Delta E_{R_2} = -2 \times \delta R_2 / R_2 = -2 \times 1.6 / (-50.58) = 0.0692;$$

$$\Delta E_{n_1} = -2 \times \delta n_1 / (n_1 - 1) = -0.0154;$$

$$\Delta E_{n_2} = -2 \times \delta n_2 / (n_2 - 1) = 0.0079;$$

$$\Delta E_{d_1} = \ln 0.99 \times \delta d_1 = -0.002;$$

$$\Delta E_{d_2} = \ln 0.99 \times \delta d_2 = -0.004;$$

Таблица 1 – Значения конструктивных параметров активного элемента

| Обоз. | Наименование параметра                       | Раз-<br>мерность | Номи-<br>нальное<br>значение | Допуск                  |
|-------|--|------------------|------------------------------|-------------------------|
| $n_1$ | Показатель пропускания света первым зеркалом | -                | 1.5163                       | $\pm 20 \times 10^{-4}$ |
| $n_2$ | Показатель пропускания света вторым зеркалом | -                | 1.50628                      | $\pm 10 \times 10^{-4}$ |
| $R_1$ | Радиус кривизны поверхности первого зеркала  | мм               | 33.65                        | 0.99                    |
| $R_2$ | Радиус кривизны поверхности второго зеркала  | мм               | -50.58                       | 1.5                     |
| $d_1$ | Толщина первого зеркала по оси               | мм               | 2.0                          | $\pm 0.10$              |
| $d_2$ | Толщина второго зеркала по оси               | мм               | 4.0                          | $\pm 0.20$              |

|                 |  |     |     |   |
|-----------------|--|-----|-----|---|
| $P_{\text{вх}}$ | Средняя мощность лазерного излучения внутри активного элемента | мВт | 3.0 | - |
| $D_{\text{вх}}$ | Диаметр пучка лазерного излучения в оптической системе         | мм  | 3.0 | - |

Рассчитаем номинальное значение плотности лазерного излучения:

$$E = 4 \times 0.81 \times 0.9 \times 0.92 \times 3[(11.323 \times 0.256)/(0.09 \times 25.5 \times 0.26)];$$

$$E = 14 \text{ мВт/см}^2;$$

Запишем для  $E$  функцию чувствительности и подсчитаем относительное изменение допуска и относительное отклонение середины поля допуска плотности потока лазерного излучения  $E$ .

$$\delta E_{\text{отн}} = (N_{R_1}^2 K_{R_1}^2 (\delta R_1)^2 + N_{R_2}^2 K_{R_2}^2 (\delta R_2)^2 + N_{n_1}^2 K_{n_1}^2 (\delta n_1)^2 + N_{n_2}^2 K_{n_2}^2 (\delta n_2)^2 + N_{d_1}^2 K_{d_1}^2 (\delta d_1)^2 + N_{d_2}^2 K_{d_2}^2 (\delta d_2)^2)^{1/2}; \quad (2)$$

$$\delta E_{\text{отн}} = 0.0853;$$

$$\Delta E_{\text{отн}} = N_{R_1} (\Delta R_1 - 0.5\alpha_{R_1} \delta R_1) + N_{R_2} (\Delta R_2 - 0.5\alpha_{R_2} \delta R_2) + N_{n_1} (\Delta n_1 - 0.5\alpha_{n_1} \delta n_1) + N_{n_2} (\Delta n_2 - 0.5\alpha_{n_2} \delta n_2) + N_{d_1} (\Delta d_1 - 0.5\alpha_{d_1} \delta d_1) + N_{d_2} (\Delta d_2 - 0.5\alpha_{d_2} \delta d_2); \quad (3)$$

$$\Delta E_{\text{отн}} = 0.059;$$

Абсолютные значения допусков и отклонений координат середины поля допуска рассчитываются по формулам:

$$\delta E = E \times \delta E_{\text{отн}} = 14 \times 0.00853 = 1.19 \text{ мВт/см}^2;$$

$$\Delta E = E \times \Delta E_{\text{отн}} = 14 \times 0.0059 = 0.8 \text{ мВт/см}^2;$$

Значение координаты середины поля допуска плотности потока лазерного излучения с учетом рассчитанных значений отклонений координат середины поля допуска конструктивных параметров равно:

$$E = 14 + 0.08 = 14.8 \text{ мВт/см}^2;$$

Таким образом, при заданных конструктивных параметрах и допусках на данную оптическую систему, можно спрогнозировать расчетное значение плотности лазерного излучения, которое получилось равным  $14.8 \text{ мВт/см}^2$ , и допуск, который составляет  $1.19 \text{ мВт/см}^2$

Если полученные значения соответствуют требованиям ТЗ, то переходят к этапу производства, в противном случае – к корректировке режимов технологических операций с целью изменения производственных погрешностей или ужесточение допусков на конструктивные размеры элементов конструкции лазерной системы.

### **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. А.В. Гейниц, С.В. Москвин, А.А. Ачилов «Внутривенное лазерное облучение крови» - М.–Тверь: ООО «Издательство «Триада», 2012. – 336 с.
2. Е.А. Шахно, Физические основы применения лазеров в медицине – СПб: НИУ ИТМО, 2012, -129 с.
3. Электронные системы для прецизионного управления лазерным излучением: учеб. пособие. / В.А. Жмудь. – Новосибирск: Изд-во НГТУ, 2017. – 435 с.

УДК 004.925.8

### **К.А. ГУЩИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **РАЗВИТИЕ СОВРЕМЕННОГО МИРА НА ПРИМЕРЕ 3D МОДЕЛИРОВАНИЯ**

*Рассматривается актуальность 3D моделей и создание их с помощью современных программ.*

В современном мире все постоянно развивается: начиная от проектирования дизайна столовых приборов и заканчивая методами производства космического оборудования. Тоже самое касается и моделирования различных объектов – на смену 2D-моделирования приходят 3D-построения моделей в специальных программах.

Раньше представить дизайн здания мы могли лишь только по чертежу или рисунку, но благодаря появлению трехмерного моделирования стало доступным создание объемных изображений, которые помогают проще визуализировать строение.

3D-моделирование - это создание трехмерных изображений с помощью различных компьютерных программ. Благодаря этому стало возможным более точно визуализировать проектируемую модель: его объем, внешний вид и многое другое. Сейчас в компьютерной графике

можно воплотить очень реалистичные модели разной сложности, кроме того проектирование 3D-объектов позволяет представить модель со всех сторон и устранить недостатки до ее реализации. Таким объектом может быть автомобиль, холодильник или даже персонаж компьютерной игры.

Процесс создания 3D-модели осуществляется в несколько этапов:

- 1) Моделирование (создание математической базы);
- 2) Текстурирование (добавление текстуры объектам);
- 3) Симуляция динамики (используется для выявления недостатков при механических воздействиях);
- 4) Рендеринг (проекционное построение);
- 5) Компоновка (корректировка готовой модели).

Одни из самых популярных зарубежных программ: Autodesk 3ds Max, Autodesk Softimage, Blender и многие другие. Продуктами, которые находятся в свободном доступе для всех пользователей являются Blender, K-3D и SolidWorks. Более популярными российскими программными продуктами в этой области моделирования являются T-FLEX CAD 7.0 фирмы «Топ Системы» и «КОМПАС». Они имеют возможность оформления проектной и конструкторской документации.

Приведу пример своих работ в программах Blender и SolidWorks.

Blender – это многофункциональный инструмент для работы с трехмерной графикой, анимацией или даже создания компьютерных игр.

Процесс создания домика шамана (рисунок 1,2):

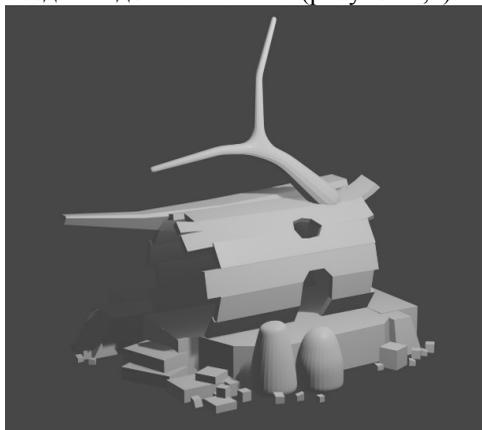


Рисунок 1 – Начальная стадия моделирования объектов



Рисунок 2 – Завершающий этап моделирования

С помощью работы в нескольких режимах программы (Object mode, Edit mode, Sculpt mode и др.) был создан данный элемент сцены.

SolidWorks – программное обеспечение САПР для автоматизации работ промышленного предприятия на этапах конструкторской и технологической подготовки производства. Эта программа позволяет создавать изделия любой сложности и назначения.

Примером работы в SolidWorks является собственная разработка кофемашины (рисунок 3, 4) для кафе.

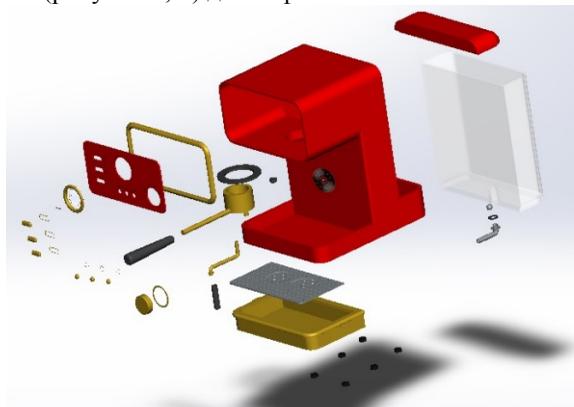


Рисунок 3 – Разбор кофемашины на детали



Рисунок 4 – Готовая модель кофемашины

На основе проделанной работы было выявлено преимущество 3D-моделирования:

1. Представление изделия в пространстве;
2. Нет необходимости использовать дополнительную физическую модель;
3. С помощью трёхмерных методов можно автоматически получить чертежи изделий;
4. Предварительная оценка недостатков перед реализацией;
5. Автоматизированный расчет свойств продукта, с помощью которых происходит ускорение процесса разработки;
6. Возможность быстрого корректирования модели;
7. Анализ моделей, для оценки функционирования проекта при заданных параметрах.

Исходя из вышесказанного можно сделать вывод, что 3D-моделирование хорошо развито в современном мире. Оно позволяет достаточно быстро и реалистично создавать проекты любой сложности с минимальными затратами.

#### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Алямовский, А.А. SolidWorks 2007/2008. Компьютерное моделирование в инженерной практике / А.А. Алямовский. – М.: СПб: БХВ-Петербург, 2008. – 192 с.

2. Большаков, В. 3D-моделирование в AutoCAD, КОМПАС-3D, SolidWorks, Inventor, T-Flex / В. Большаков, А. Бочков, А. Сергеев. – М.: Книга по Требованию, 2010. – 336 с.

3. Прахов, А. Blender. 3D-моделирование и анимация. Руководство для начинающих / А. Прахов. – М.: БХВ-Петербург, 2009. – 272 с.

УДК 004.031.6

**Д.С. ДАНЬКОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **ОБЗОР СОВРЕМЕННЫХ ПЛИС И САПР ФИРМЫ XILINX**

*Рассматривается краткое описание современного программного обеспечения, а также выпускаемой линейки ПЛИС фирмы XILINX.*

### **Программируемые устройства Xilinx.**

Впервые в 84 году 20 века Xilinx создала программируемые логические интегральные схемы (ПЛИС) с архитектурой FPGA (Field Programmable Gate Array), что позволяет оставаться фирме и по сей день главным глобальным производителем программируемых логических микросхем. Технологии Xilinx позволяют изготовителям электронной оснастки минимизировать опасности ввиду сокращения периода изготовления новых изделий и скорости их реализации в рынок. Производители проектируют и отлаживают свои специфичные изделия на базе ПЛИС Xilinx заметно быстрее, чем при применении общепринятых методов. Кроме того, ПЛИС Xilinx являются типовыми продуктами, реализованными для программирования, производителям не нужно ожидать создания прототипов или подвергать риску производство внушительными расходами на подготовку производства, как при использовании фиксированной логики или ASIC. В наши дни продукция Xilinx свободно применяется - от телекоммуникаций и бытовой электроники до машинного и промышленного оборудования.

### **FPGA Virtex FPGA, производимые компанией Xilinx.**

Линейка Virtex вывела термин "программируемая логика" на другой уровень, существенно увеличив классические возможности ПЛИС, реализующие потребности конструкторов высокопроизводительных систем. Самое актуальное поколение ПЛИС содержит семейство Virtex-4, особенность которого заключена в применении архи-

тектуры ASMBL (Advanced Silicon Modular Block), реализованной с помощью модульных логических блоков. В этом семействе содержатся три платформы - LX, FX и SX, которые различаются своими наборами функциональных параметров, наличием обширного диапазона интегрированных аппаратных IP-ядер, создающих полноценные заключения комплексного уровня. Базис IP-ядер содержит процессоры PowerPC (с интерфейсом APU), трехрежимный Ethernet MAC, последовательные приемопередатчики 622 Мбит/с - 10 Гбит/с, выделенные блоки DSP, высокочастотные блоки управления тактовой частотой и интерфейсные блоки, синхронизированные между собой. Базисные блоки Virtex-4 являются дальнейшим усовершенствованием аналогов прошлых семейств Virtex - Virtex, Virtex-E и т.д., реализуя восходящую совместимость с существующими разработками. Ключевыми достоинствами Virtex являются упрощение топологии печатной платы, ограничение ассортимента компонентов и сроков выхода продукта на рынок.

#### **ПЛИС Xilinx Spartan FPGA, производимые компанией Xilinx.**

Данные ПЛИС изготовлены для недорогого крупномасштабного производства. Реализуя набор логической емкости от 50 тысяч до 5 миллионов комплексных вентилей, современной платформой является Spartan-3, произведенная для создания дешевых решений, представляется самой недорогой среди всего разнообразия ПЛИС. На рис.1 изображен пример ПЛИС данной серии.



Рисунок 1 – ПЛИС Spartan-3 фирмы Xilinx

Вследствие ПЛИС Spartan-3 Xilinx попадает в такие области рынка глобального спроса, прежде являющимися для применения ASIC среднего уровня. Эффективная эксплуатация превосходств 90-нм технологии позволяет вести агрессивную целевую политику, предоставляя покупателям емкие и дешевые функциональные ПЛИС. Серия

Spartan-3 крайне распространена среди изготовителей электроники в разных областях рынка, например, бытовая электроника, цифровое видео и т.д. Последней произведенной серией является Spartan-3E, предоставляющая логическую емкость до 1,6 миллиона системных клапанов, до 376 программируемых выводов и пластичную архитектуру с наименьшими затратами на логический элемент в отрасли, а также линейку Spartan-3L, созданную для получения наименьшего значения энергопотребления. При использовании данных ПЛИС возможно достигнуть понижения статического энергопотребления на 98%, что позволяет использовать недорогие охлаждающие устройства, понизить габариты устройства и увеличить надежность системы в целом.

### **Продукты Xilinx, интегрированные в систему.**

Xilinx позиционируют программные средства, как ведущую составляющую программируемых логических решений. Со старта продаж Xilinx предоставили более 150 тысяч лицензий на систему ISE CAD. Xilinx предлагают создателям множество программных инструментов, размещенных в сети Интернет. Эти программные средства позволяют специалистам приобретать моментальный прямой доступ к технической помощи на веб-сайте Xilinx. Инструменты проектирования, представляемые Xilinx, являются самыми быстрыми в сфере и гарантируют уровень работоспособности значительно выше, чем у конкурирующих инструментов проектирования для FPGA.

### **Нововведения Xilinx в области систем проектирования.**

Программное обеспечение (ПО) Xilinx ISE является популярным средством, гарантирующим стабильную производительность изготавливаемых устройств с программируемой логикой, в то же время ISE представляется сравнительно дешевой продукцией. В последующей версии ПО ISE 8.1i Xilinx демонстрируют уникальную технологию ISE Fmax, предназначенную для повышения работоспособности производимых устройств, реализованных с помощью FPGA. Используя более продвинутые инструменты, например, ПО PlanAhead, разработчикам предоставляется возможность заметно сократить время создания проекта.

Разработчики, применяющие ПЛИС или ПЛИС CPLD, могут использовать современное оборудование, реализованное Xilinx. Для проектов, обладающие значительным логическим объемом и высокой работоспособностью, Xilinx предоставляет серию ПЛИС Virtex-4 — это форма аппаратной платы для реализации наиболее трудоемких проектов. Для крупномасштабных проектов изготавливаются ПЛИС серии Spartan-3 и CPLD, предназначенные для изготовления продукции, предназначенной для крупных партий при низких затратах. В то же

время Xilinx предоставляет производителю полный набор инструментов проектирования.

### **Конфигурация ISE.**

Полнофункциональной версией ПО ISE является ISE Foundation, содержащая в себе возможность функционирования со всеми ведущими сериями ПЛИС и ПЛИС CPLD от Xilinx. Является максимально полным инструментом производства для изделий, использующих программируемую логику, обеспечивающий необходимую работоспособность, взаимодействие с энергопотреблением, удешевлением продукта. ISE Foundation и ISE WebPACK, обладают возможности поддержания технологии Fmax, достигающей увеличения работоспособности ПЛИС Virtex-4 значительно выше, если проводить анализ с другими решениями компании.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Основные особенности семейства ПЛИС Xilinx Spartan [Электронный ресурс]. – Режим доступа: <https://www.macrogroup.ru/catalog/partgroup/52>.

2. ISE Design Suite [Электронный ресурс]. – Режим доступа: <https://www.xilinx.com/products/design-tools/ise-design-suite.html>.

УДК 004.65

**М.С. ДУБЧЕНКО**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПОДСИСТЕМА ДИСТАНЦИОННОГО БАНКОВСКОГО ОБСЛУЖИВАНИЯ «ИНТЕРНЕТ–КЛИЕНТ»**

*Рассматриваются основные понятия, а также преимущества и недостатки подсистемы «Интернет-Клиент».*

В настоящее время сама по себе система, которая обеспечивает обслуживание любых клиентов через Интернет, используя при этом только стандартный браузер и ключи защиты клиентов, являясь при этом мощным инструментом дистанционного банковского обслуживания (ДБО), отнюдь не является единственной востребованной и достаточной системой ДБО универсального банка. Необходимой данную систему делает одна из подсистем ДБО – «Интернет–Клиент».

#### **1. ОСНОВНЫЕ ПОНЯТИЯ**

**Дистанционное банковское обслуживание (ДБО)** – это общий термин, который предназначен для описания технологий предоставления банковских услуг на основании распоряжений, которые передаются клиентом удалённым образом (т.е. без ему не нужно посещать банк), чаще всего с использованием компьютерных и телефонных сетей.

**Подсистема «Интернет–Клиент»** – это основная подсистема, являющаяся одним из видов ДБО, которая предоставляет пользователю удалённый доступ к своим счетам, а также предоставляет возможность работы с депозитами и кредитными продуктами.

## **2. ВИДЫ ДБО**

Классификация технологий ДБО происходит по типам информационных подсистем (ИП) или, другими словами, программно–аппаратных средств, которые используются для осуществления различных банковских операций.

### **2.1 ПОДСИСТЕМА «КЛИЕНТ–БАНК»**

**Подсистема «Клиент–Банк»** – это один из видов подсистем ДБО, доступ к которому осуществляется при помощи персонального компьютера (ПК). Такие подсистемы позволяют отправлять платежи в любой банк, а также получать выписки по счетам (информацию о «движениях» средств на счёте) также из любого банка, обеспечивают ведение валютных счетов с удалённого рабочего места. При этом банки предоставляют клиенту: техническую и методическую поддержку при установке такой подсистемы, начальное обучение персонала клиента, обновление программного обеспечения (ПО) и сопровождение в процессе дальнейшей работы.

Для того чтобы обеспечить максимальную безопасность пользователей такой подсистемы, используются разнообразные системы криптографической защиты информации (СКЗИ), которые обеспечивают контроль целостности передаваемой в Банк информации и шифрование.

### **2.2 ПОДСИСТЕМА «ТЕЛЕФОН–БАНК»**

**Подсистема «Телефон–Банк»** – это один из видов подсистем ДБО, который обеспечивает передачу клиентской информации в банк в зависимости от различных способов её реализации:

- путём использования телефона и голосового меню (Interactive Voice Response – IVR);
- общение оператора телефонного обслуживания с клиентом (Call Center);
- средства передачи SMS-сообщений.

Такие подсистемы имеют ограниченный набор функций в сравнении с подсистемами «Клиент–Банк». Список функций следующий:

- информация об остатках на счетах клиентов;
- ввод заявок на предоставление факсимильной копии выписки по счёту;
- информация о суммах разнообразных поступлений в пользу клиента;
- ввод заявок на передачу факсимильной копии платёжного поручения;
- ввод заявки на исполнение подготовленного по шаблону поручения на перевод – средств;
- ввод заявок о проведении платежей, а также заказ наличности;
- оплата услуг отдельных компаний и организаций.

### 2.3 ПОДСИСТЕМА «ИНТЕРНЕТ–КЛИЕНТ»

Как было сказано ранее, **подсистема «Интернет–Клиент»** – это основная подсистема, являющаяся одним из видов ДБО, которая предоставляет пользователю удалённый доступ к своим счетам, а также предоставляет возможность работы с депозитами и кредитными продуктами. Такая подсистема реализуется в рамках ДБО BS-Client v.3. «Интернет–Клиент» справляется с задачами как информационного, так полноценного платёжно–расчётного обслуживания юридических и физических лиц, а также производит обмен сообщениями произвольного формата, осуществляет получение разнообразных выписок (в различных форматах и видах) и иной информации из любых банков.

Пользователь осуществляет вход при помощи браузера, а подсистема размещается на web–сервере банка. Все данные, которые предоставляет пользователь (это могут быть различные платёжные документы и выписки по счетам), будут доступны на web–сайте этого банка.

Для успешного внедрения подсистемы «Интернет–Клиент», она обладает двумя следующими особенностями:

- массовость внедрения, это выражается как в абсолютной простоте установки системы у клиента (максимум инсталляции должен выражаться в нажатии какого–нибудь setup.exe файла), так и в низкой стоимости самой системы и владения ею;
- абсолютная защищенность документооборота банка браузерным клиентом и юридическая значимость.

Любая подсистема «Интернет–Клиент», не обладающая этими двумя свойствами, обречена превратиться в подсистему «Клиент–Банк». Это сказывается на том, что приложение для работы с данной

подсистемой будет иметь огромное специальное ПО и ряд следующих недостатков:

- минимальный объём клиентской части – только система защиты и ключи в объёме одной дискеты;
- использование только стандартного НТТР-протокола;
- минимальный http-трафик "банк - клиента" в сеансе, возможность работы через проху-сервер;
- универсальность средства доступа к Сети - стандартный браузер (MS IE и др.);
- безусловная простота и массовость внедрения;
- привычный и удобный, но в то же время не громоздкий интерфейс.

### **2.3.1 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ ПОДСИСТЕМЫ «ИНТЕРНЕТ–КЛИЕНТ» ПРИ ПОМОЩИ СИСТЕМЫ BS–DEFENDER**

Безопасность в подсистеме «Интернет–Клиент» обеспечивает такая система, как BS–Defender. Она автоматически устанавливается при запуске setup.exe и осуществляет ряд следующих функций:

- проверка электронно–цифровой подписи (ЭЦП);
- дешифрация всех данных, которые отправляются в банк и подписей отдельных документов (ActiveX);
- подпись всех данных, которые отправляются в банк и подписей отдельных документов (ActiveX);
- архивирование, подпись, протоколирование и шифрование всего НТТР–трафика в банк и из банка для обеспечения значимого разбора конфликтных ситуаций.

Данная система обеспечения безопасности имеет следующий вид (рисунок 1) и основана на:

- собственной системе BS–Defender;
- аутентификации, авторизации и протоколировании;
- штатных средствах защиты операционной системы (ОС) и системы управления базой данных (СУБД);
- межсетевого экрана (firewall) и проху–сервера;
- произвольных внешних криптографических средств защиты (СКЗИ).

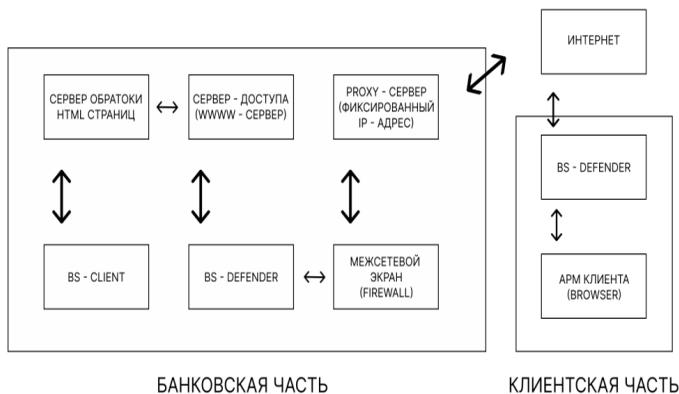


Рисунок 1 – Система обеспечения безопасности BS-Defender

Показанная на рисунке выше схема содержит такие элементы, как **proxy-server** (это промежуточный сервер или комплекс программ, который используется в компьютерных сетях и выполняет роль посредника между клиентом и целевым сервером, позволяющий клиенту как выполнять косвенные запросы к другим сетевым службам, так и получать ответы на эти запросы) и межсетевой экран – **firewall** (это программный или программно-аппаратный элемент компьютерной сети, который, осуществляет контроль и фильтрацию проходящего через него сетевого трафика в соответствии с заданными правилами). Этим она демонстрирует всю структуру безопасности в системе «Интернет–Клиент», которая включает в себя средства обеспечения безопасности информационных ресурсов банка. Схема работы межсетевого экрана представлена на следующей рисунке (рисунок 2):

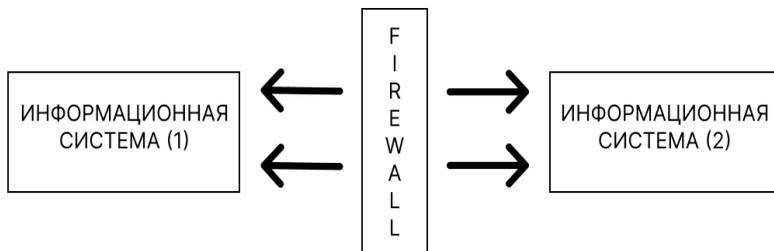


Рисунок 2 – Принцип работы межсетевого экрана

### 2.3.2 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ ПОДСИСТЕМЫ «ИНТЕРНЕТ–КЛИЕНТ» ПРИ ПОМОЩИ ТОКЕНОВ ГОСТ

Помимо обеспечения безопасности, которую предоставляет специальная система защиты BS–Defender, существует ещё второй способ предоставления безопасности для рассматриваемой подсистемы «Интернет–Клиент» - это токены ГОСТ.

Изначально **токены** – это такие устройства, которые обеспечивают дополнительный уровень защиты для подсистемы «Интернет–Клиент».

**Токен ГОСТ** – это некий USB–ключ, который предназначен для обеспечения надёжной защиты электронно–цифровой подписи (ЭЦП). Файл, имеющий ключ ЭЦП и записанный на токен ГОСТ, никогда не сможет покинуть этот токен. Также с токена невозможно скопировать любой вид информации (существует специальный сертификат ФСБ, который подтверждает надёжность данного устройства).

Параметры токен ГОСТ:

- устройство гарантирует безопасность подсистемы «Интернет–Клиент» – токен ГОСТ исключает возможность копирования ключей;
- мобильность использования – возможность работы в «недоверенной среде»;
- удобство и простота использования – простой процесс настройки токен ГОСТ;
- на любой токен ГОСТ можно записать до семи файлов различного назначения с различными ключами электронно–цифровой;
- надёжность устройства – время «жизни» не ограничено, практически не изнашивается/ломается.

#### ЗАКЛЮЧЕНИЕ

В настоящее время происходит очень стремительный рост дистанционного банковского обслуживания, благодаря чему все виды подсистем ДБО развиваются и улучшаются каждый день, появляются всё более новые и новые классификации, благодаря чему формируются и начинают существовать новые подсистемы. Из-за такого быстрого развития технологий в данной сфере, клиент всегда может выбрать более удобную для себя технологию предоставления банковских услуг.

#### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Дистанционное банковское обслуживание. – 2010. – 328 с.
2. Александр Турбанов, Александр Тютюнник. Банковское дело. Операции, технологии, управление. – М.: «Альпина Паблишер», 2010. – 688 с.

3. Центр исследований платёжных систем и расчётов. Мошенничество в платёжной сфере: Бизнес–энциклопедия. – М: Интеллектуальная литература, 2016. – 345 с.

УДК 004.023

**М.С. ДЬЯКОВ, С.В. СКВОРЦОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ РЕШЕНИЯ ТРАНСПОРТНОЙ ЗАДАЧИ**

*Представлена реализация генетического алгоритма для решения транспортной задачи по критерию стоимости. Определены особенности задачи, которые могут учитываться дополнительно по сравнению с использованием классических методов.*

**Введение.** Транспортная задача – это задача о нахождении такого плана перевозок, при котором их суммарная стоимость является минимальной. Если цены перевозок пропорциональны количеству перевозимого груза, то транспортная задача называется линейной, в противном случае – нелинейной.

Оптимальное решение линейной транспортной задачи по критерию стоимости можно получить методом потенциалов [1]. Однако применение более современных методов оптимизации предоставляет ряд новых возможностей. В частности, решение транспортной задачи с использованием генетического алгоритма (ГА) обеспечивает:

- получение результатов как для дискретных (целочисленных), так и произвольных значений управляемых переменных;
- использование нелинейной целевой функции;
- учет нескольких частных критериев оптимальности [2];
- получение целого списка «хороших» решений;
- поддержку параллельных вычислений [3] и др.

Общая структура простого ГА приведена на рисунке 1. Рассмотрим его реализацию применительно к решению транспортной задачи.

**Создание исходной популяции.** Главной проблемой при формировании начальной популяции является необходимость получения множества отличающихся начальных решений. Чтобы получить генетическое разнообразие особей начальной популяции, воспользуемся модифицированным методом северо-западного угла. Он отличается от классического варианта случайным выбором первой заполняемой

клетки таблицы перевозок. В эту клетку записывается минимально допустимое значение, а далее опорный план формируется классическим методом северо-западного угла.

В качестве примера на рисунке 2 приведены два опорных решения, где случайно выбранная клетка выделена серым цветом. В верхнем правом углу каждой клетки указана стоимость перевозки, сверху таблицы - запасы, справа – потребности.

**Кодирование хромосом.** Хромосомы будем представлять в виде вектора числовых данных (вещественных или целых), где каждый элемент (ген) соответствует количеству товара, доставляемого от конкретного поставщика конкретному потребителю. Иначе говоря, каждый ген определяет числовое значение некоторой клетки таблицы. Для решений, показанных на рисунке 2, хромосомы имеет следующий вид соответственно:

$$\bar{x}_1 = (0, 0, 0, 10, 0, 5, 5, 5, 15, 5, 0, 0);$$

$$\bar{x}_2 = (10, 0, 0, 0, 0, 0, 0, 0, 15, 5, 10, 5, 0).$$

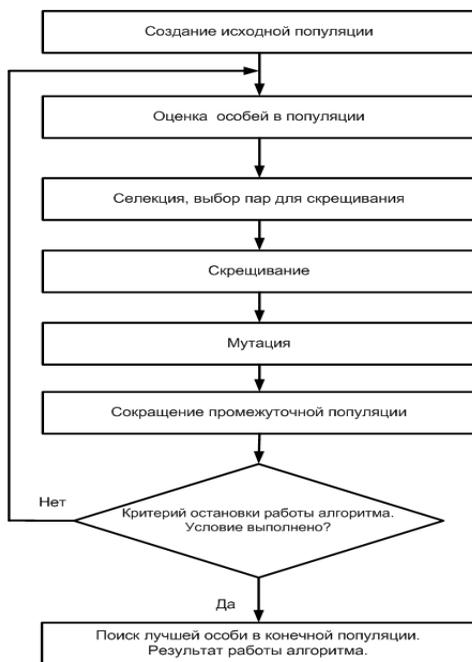


Рисунок 1 – Общая схема генетического алгоритма

|    |    |   |    |    |
|----|----|---|----|----|
| 15 | 10 | 5 | 15 |    |
| 5  | 5  | 5 | 4  |    |
|    |    |   | 10 | 10 |
| 5  | 6  | 8 | 4  |    |
|    | 5  | 5 | 5  | 15 |
| 2  | 3  | 5 | 4  |    |
| 15 | 5  |   |    | 20 |

|    |    |   |    |    |
|----|----|---|----|----|
| 15 | 10 | 5 | 15 |    |
| 5  | 5  | 5 | 4  |    |
| 10 |    |   |    | 10 |
| 5  | 6  | 8 | 4  |    |
|    |    |   | 15 | 15 |
| 2  | 3  | 5 | 4  |    |
| 5  | 10 | 5 |    | 20 |

Рисунок 2 – Два опорных решения

**Оценка особей.** Для вычисления значений функции приспособленности (фитнес-функции) требуется умножить значения каждого гена хромосомы на соответствующую стоимость перевозки и просуммировать полученные результаты. Введем в рассмотрение вектор стоимостей перевозок, который для представленной выше таблицы имеет следующий вид:

$$\bar{c} = (5, 5, 5, 4, 5, 6, 8, 4, 2, 3, 5, 4).$$

Тогда вычисление значений функции приспособленности можно реализовать, используя операцию скалярного умножения векторов

$$F(\bar{x}_1) = (\bar{c}, \bar{x}_1); \quad F(\bar{x}_2) = (\bar{c}, \bar{x}_2).$$

В итоге получим:

$$F(\bar{x}_1) = 5 \cdot 0 + 5 \cdot 0 + 5 \cdot 0 + 4 \cdot 10 + 5 \cdot 0 + 6 \cdot 5 + 8 \cdot 5 + 4 \cdot 5 + 2 \cdot 15 + 3 \cdot 5 + 5 \cdot 0 + 4 \cdot 0 = 175;$$

$$F(\bar{x}_2) =$$

$$5 \cdot 10 + 5 \cdot 0 + 5 \cdot 0 + 4 \cdot 0 + 5 \cdot 0 + 6 \cdot 0 + 8 \cdot 0 + 4 \cdot 15 + 2 \cdot 5 + 3 \cdot 10 + 5 \cdot 5 + 4 \cdot 0 = 115.$$

**Селекция.** Результаты вычисления функции приспособленности для всех особей текущей популяции позволяют выбрать наилучших для воспроизведения потомства. Для этого используется турнирный отбор, при реализации которого популяция разбивается на подгруппы из двух особей, после чего в каждой подгруппе выбирается лучшая путем сравнения значений функции приспособленности.

**Скрещивание.** На очередной итерации ГА каждая особь, победившая в турнирном отборе, может подвергнуться скрещиванию с некоторой заданной вероятностью, причем второй родитель пары выбирается случайно из числа победивших в турнире.

При выполнении скрещивания основная проблема заключается в получении новых решений, удовлетворяющих постановке задачи. Иначе говоря, результаты скрещивания не всегда дают корректные результаты с точки зрения сохранения баланса между запасами и потребностями. Для реализации оператора скрещивания предлагается использовать следующий алгоритм, основанный на процедуре восстановления корректности решения.

1. Выполняется суммирование хромосом родителей  $\bar{x}_1$  и  $\bar{x}_2$  путем использования операции сложения векторов  $\bar{x}_3 = \bar{x}_1 + \bar{x}_2$ .

2. Проверяется корректность решения, соответствующего полученному вектору  $\bar{x}_3$ .

3. Если решение  $\bar{x}_3$  не удовлетворяет ограничениям задачи, то выполняется восстановление решения.

3.1. Для каждой клетки таблицы перевозок, полученное значение которой превышает допустимую величину запаса или потребности, присваивается нулевое значение.

3.2. По значениям ненулевых клетки изменяются соответствующие величины запасов и потребностей.

3.3. Выполняется распределение оставшихся запасов и потребностей методом северо-западного угла.

3.4. Если восстановленное решение не удовлетворяет ограничениям задачи, то оно уничтожается.

Например, для пары хромосом родителей  $\bar{x}_1$  и  $\bar{x}_2$ , которым соответствуют таблицы перевозок на рисунке 2, результат суммирования  $\bar{x}_3$  (шаг 1) показан на рисунке 3.

|    |   |    |   |   |   |    |    |
|----|---|----|---|---|---|----|----|
|    | 5 |    | 5 |   | 5 |    | 4  |
| 10 |   |    |   |   |   |    | 10 |
|    | 5 |    | 6 |   | 8 |    | 4  |
|    |   | 5  |   | 5 |   | 20 |    |
|    | 2 |    | 3 |   | 5 |    | 4  |
| 20 |   | 15 |   | 5 |   |    |    |

Рисунок 3 – Сумма хромосом родителей

Очевидно, что полученный план перевозок не удовлетворяет условиям задачи и должен быть восстановлен. Частичное решение, полученное как результат шагов 3.1 и 3.2, приведено на рисунке 4, где над таблицей и справа от нее указаны нераспределенные значения запасов и потребностей. Полностью восстановленное решение (после шага 3.3), удовлетворяющее ограничениям задачи, представлено на рисунке 5.

|    |   |   |   |    |   |
|----|---|---|---|----|---|
|    | 5 | 5 | 0 | 15 |   |
|    | 5 |   | 5 |    | 4 |
| 10 |   |   |   |    |   |
|    | 5 |   | 6 |    | 8 |
|    |   | 5 |   | 5  |   |
|    | 2 |   | 3 |    | 5 |
|    |   |   |   |    | 4 |
| 20 |   |   |   |    |   |

Рисунок 4 – Частичное решение

|    |    |    |   |    |    |
|----|----|----|---|----|----|
|    | 15 | 10 | 5 | 15 |    |
|    | 5  |    | 5 |    | 4  |
| 10 |    |    |   |    |    |
|    | 5  |    | 6 |    | 8  |
|    |    | 5  |   | 5  |    |
|    | 2  |    | 3 |    | 5  |
|    |    |    |   |    | 4  |
| 5  |    | 5  |   |    | 10 |

Рисунок 5 – Восстановленное решение

**Мутация.** После операции скрещивания промежуточная популяция особей включает потомков и их родителей. Каждая особь имеет шанс с некоторой вероятностью подвергнуться мутации. Оператор мутации реализован инверсией, где два случайно выбранных гена хромосомы меняются местами между собой. При этом возможно получение некорректного решения, к которому применяется рассмотренная процедура восстановления.

**Редукция.** На этом этапе ГА происходит сокращение промежуточной популяции до исходного размера. Родители, потомки и мутанты ранжируются по значению функции приспособленности и в следующее поколение включаются лучшие особи в количестве, равном начальному размеру популяции.

**Заключение.** Представленный ГА обеспечивает решение линейных транспортных задач, но легко может быть модифицирован для случая нелинейной целевой функции. При этом управляемые переменные могут принимать как целочисленные, так и вещественные значения. Исследования разработанного алгоритма показывают, что среднее по популяции значение целевой функции стабильно уменьшается при переходе к новому поколению, пока не будет получено наилучшее в данном эксперименте значение стоимости перевозки [4]. Следует также отметить, что размерность решаемой транспортной задачи мало влияет на время работы алгоритма, что позволяет решать задачи с большим числом поставщиков и потребителей.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Акулич И.Л. Математическое программирование в примерах и задачах. – М.: Лань, 2009. – 352 с.
2. Рутков–ская Д, Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. – М.: Горячая линия – телеком, 2006. – 452.
3. Лунин Д.В., Скворцов С.В. Разработка параллельного генетического алгоритма для решения задачи коммивояжера на платформе CUDA // Системы управления и информационные технологии. – 2015. – № 2 (60). – С. 50-55.
4. Дьяков М.С, Скворцов С.В. Решение транспортной задачи на основе генетического алгоритма // Новые информационные технологии в научных исследованиях (НИТ 2021): материалы XXVI Всероссийской научно-технической конференции; Рязань: ИП Коняхин А.В. (Book Jet), 2021. С. 166-167.

УДК 681.5

**Д.А. ЕЖОВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**СТРУКТУРА И ПРИНЦИП РАБОТЫ ШАССИ  
МНОГОФУНКЦИОНАЛЬНОГО РОБОТА**

Приводится описание структуры и принципа работы многофункционального робота.

Робот – это устройство, которое автоматически действует по заранее заложённой программе и получает информацию при помощи датчиков, робот самостоятельно выполняет производственные и иные операции.

При этом робот может, как получать команды от оператора так и действовать автономно. Сообщения об автоматических технических устройствах с более или менее сложной механикой, предназначенных не для выполнения конкретных работ, а только для демонстрации, занимают все большее место в средствах информации, большинство их называется роботами.

Роботы используются для решения различных задач в нашей жизни. Из-за разнообразия решаемых задач возникает необходимость конструирования многофункционального универсального робота, который можно подстроить под выполнение определённых функций, например: видеонаблюдение, поисковые работы, анализ температуры, влажности и других параметров в труднодоступных для человека местах. Данная проблема решается возможностью применения различных модулей, включающих в себя например видеокамеры, микрофоны, различные датчики и т.д.

Для размещения подобных модулей требуется универсальное шасси, главным критерием которого должна быть многофункциональность. Данное решение обеспечивает мобильность размещённых на нём устройств, позволяя выполнять сложные задачи в труднодоступных для человека местах.

**Устройство шасси многофункционального робота.**

Разрабатываемое шасси имеет следующие составные части:

1. Микроконтроллер Arduino Uno на основе логической микросхеме ATmega328. Данная микросхема является главным контроллером всей системы.

2. Драйвер двигателя L298N используемый для управления двигателями постоянного тока.

3. Зарядка для АКБ и плата балансировки, обеспечивающие электропитание системы.

4. Bluetooth приемник сигнала беспроводного джойстика (в данной модели используется джойстик от PS2 из за его дешевизны и доступности), а так же плата БМС для защиты питания данного приемника.

5. А так же корпус и колёсная/гусеченая база (в зависимости от условий эксплуатации).

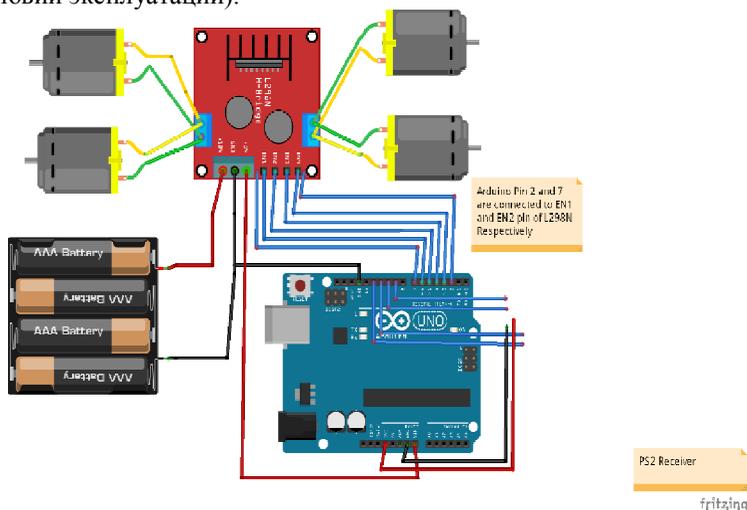


Рисунок 1 – Общая схема подключения

Шасси работает на основе заранее запрограммированного микроконтроллера Arduino Uno. Программирование микроконтроллера и выбранная элементная база позволяет выполнять следующие функции:

1. Хранение настроек в энергонезависимой памяти.  
2. Управление осуществляется с дистанционно (дальность до 20м).

3. Возможность дополнительной настройки управления модуля с джойстиком, путем программирования микроконтроллера Arduino Uno

#### **Совместимые с шасси модули:**

1. TTL модуль камеры OV670 300KP VGA Camera module – применяется для ведения видео наблюдения в труднодоступных для человека местах.

2. Микрофонный модуль на базе микросхемы MAX9814 – применяется для аудиозаписи в труднодоступных для человека местах.

3. Ультразвуковой датчик расстояния HC-sr04.
4. Серводвигатель mg946 – применяется для конструирования модулей с подвижными частями.
5. Модуль – манипулятор SG90 MG90S – используется для перемещения немассивных грузов.
6. Датчик температуры и влажности FLASH-I2C.
7. Датчик атмосферного давления BMP280.
8. Датчик влажности почвы.
9. Датчик качества воздуха CCS811 – позволяет определять значение Eсо2 и значение летучих органических веществ в воздухе (TVOC).
10. Датчик света FLASH-S22C.
11. Модуль обработки звука TREMA v2.0.

В данной статье приведены примеры наиболее популярных размещаемых модулей, поскольку в шасси построена на микроконтроллере Arduino UNO, имеется огромное количество модулей, заточенных под различные задачи.

#### **Требования к корпусу шасси:**

При проектировании корпуса нужно учитывать ряд требований к проектируемой системе:

1. Корпус должен обеспечивать влагозащиту, соответствующую дальнейшим условиям работы.
2. Корпус должен обеспечивать защиту от механических воздействий, соответствующую дальнейшим климатическим условиям работы системы.
3. Корпус должен обеспечивать возможность размещения на шасси необходимых модулей.

#### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Бейктал, Дж. Конструируем роботов на Arduino. Первые шаги / Дж. Бейктал. – М.: Лаборатория знаний, 2016. – 320 с.
2. Бербюк, В. Е. Динамика и оптимизация робототехнических систем / В.Е. Бербюк. – М.: Наукова думка, 2014. – 192 с.
3. Бройнль, Томас Встраиваемые робототехнические системы. Проектирование и применение мобильных роботов со встроенными системами управления / Томас Бройнль. – Москва: РГТУ, 2012. – 520 с.
4. Корсункий, В. А. Выбор критериев и классификация мобильных робототехнических систем на колесном и гусеничном ходу. Учебное пособие / В.А. Корсункий, К.Ю. Машков, В.Н. Наумов. – М.: МГТУ им. Н. Э. Баумана, 2014. – 862 с.

УДК 004.91

**В.В. ЕРМИЛОВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**ИСПОЛЬЗОВАНИЕ ЯЗЫКА PYTHON КАК СРЕДСТВА  
АВТОМАТИЗАЦИИ ОТЧЕТНОСТИ В УЧЕБЕ***Решение типовых учебных задач и генерация шаблонов отчетов с помощью скриптов на языке python.*

В учебной практике довольно часто процесс создания отчетов является однотипным и монотонным. Например, многие отчеты содержат большое количество скриншотов каждого этапа выполнения работы с подписями и пояснениями. Кроме этого многие этапы создания отчета всегда одинаковые. Из этого можно сделать вывод, что процесс можно автоматизировать.

В этой работе я предложу вариант автоматизации этого процесса с помощью языка программирования python и модулей python-docx для создания и редактирования документов типа doc и pdfplumber для извлечения данных из pdf документов.

Определим постановку задачи более четко. Пусть даны методические указания лабораторной работы в виде pdf файла. Для простоты будем считать, что один файл содержит методические указания только к одной лабораторной работе. Так же будем считать, что он всегда содержит следующую информацию: номер и название лабораторной работы, цель работы, контрольные вопросы. Требуется составить шаблон отчета в doc файле, содержащий титульный лист, за исключением информации о кафедре, студентах и проверяющем преподавателе.

Используемые модули размещены на PyPI, это значит их можно установить с помощью pip.

```
$ python3 -m pip install -U python-docx pdfplumber
```

Далее рассмотрим реализацию поставленной задачи. Сначала получим всю необходимую информацию из методических указаний. Для этого были использованы регулярные выражения. Код, извлекающий нужные данные:

```
import pdfplumber
import re
with pdfplumber.open(pdf_name) as pdf:
    pages = [page.extract_text() for page in pdf.pages]
```

```

text = ' '.join(pages)
lab =
re.search('Лабораторная\s*работа\s*№\s*d*\s*.*',
text)[0]
purpose =
re.search('Цель\s*работы: ([^\.] + \. [^\S\n]*\n)*',
text)[0]
quest = re.search('Контрольные
вопросы\s*(\d\.[^\.]?)*[\.\?][^\S\n]*\n)', text)[0]

```

Замечание: при обработки извлеченных данных следует учитывать добавление дополнительных пробельных символов (пробелы, табуляция, переходы на новую строку после каждой строки).

Далее сформируем шаблон отчета на основе полученных данных. Код создающий отчетный файл и вставляющий соответственные данные:

```

from docx import Document
from docx.shared import Pt
from docx.enum.text import WD_ALIGN_PARAGRAPH
from datetime import date
from docx.shared import Mm
doc = Document()
doc.add_paragraph(
'Mинистерство науки и высшего образования РФ\n'
'Федеральное государственное бюджетное
образовательное учреждение высшего образования\n'
'«Рязанский государственный радиотехнический
университет имени В.Ф. Уткина»\n\n\n'
'Кафедра \n\n\n\n'
+ lab +
'\n\n\n\n\n\n\n\n\n\n'
).alignment = WD_ALIGN_PARAGRAPH.CENTER
doc.add_paragraph(
'Выполнили:\n'
'Ст.гр.000 Бр0\n'
'Иванов И.И.\n'
'Сидоров В.В.\n'
'Проверил:\n\n\n\n\n\n'
).alignment = WD_ALIGN_PARAGRAPH.RIGHT
doc.add_paragraph('Рязань ' +
str(date.today().year) + ' г.').alignment =
WD_ALIGN_PARAGRAPH.CENTER
purpose = purpose.replace('\n', '')
purpose = purpose.replace(' ', ' ')
doc.add_paragraph(purpose).alignment =
WD_ALIGN_PARAGRAPH.JUSTIFY

```

```

    doc.paragraphs[3].paragraph_format.first_line_indent
t = Mm(15)
    doc.add_page_break()
    doc.add_paragraph('Вывод: ').alignment =
WD_ALIGN_PARAGRAPH.JUSTIFY
    doc.paragraphs[5].paragraph_format.first_line_indent
t = Mm(15)
    doc.add_paragraph(quest)
    doc.paragraphs[6].paragraph_format.first_line_indent
t = Mm(15)
    for p in doc.paragraphs:
        p.style.font.name = 'Times New Roman'
        p.style.font.size = Pt(14)
    doc.save('test.doc')

```

Важное замечание, данный способ применим только если методические указания в PDF файле представлены в виде текста. Иногда методичка представляет из себя скан бумажной версии. В этих случаях предложенный способ не применим.

Рассмотрим дополнительную задачу. Пусть в файл отчета добавляются все изображения из запускаемой директории с шаблоном подписей (Рисунок № - ). Будем считать, что изображения названы порядковым номером, а их расширение png, то есть: (№.png). Так будет понятно, в каком порядке их вставлять. Код реализующий эту задачу:

```

import glob
num = 1
context =
doc.paragraphs[4].insert_paragraph_before()
context.alignment = WD_ALIGN_PARAGRAPH.CENTER
for name in glob.glob('*.png'):
    context.add_run('\n').add_picture(name,
width=Mm(90))
    context.add_run('\n').add_text('Рисунок ' +
str(num) + ' - ')
    num = num + 1
doc.save('test.doc')

```

Замечания: предложенный способ имеет один недостаток. Он устанавливает одинаковый размер изображений. Пропорции будут соблюдены, но при этом очень маленькие изображения будут изображены очень большими, а очень большие изображения могут получиться очень маленькими и нечитабельными.

Итоговый скрипт представляет из себя весь указанный выше код. Он генерирует отчет, содержащий титульный лист, цель работы,

изображения с началом подписей, начало вывода, контрольные вопросы. Дальнейшие улучшения связаны с частными случаями. Например, можно извлекать нужные разделы методических указаний для нужных предметов (например раздел с теоретической частью) или вставлять результаты работы программы (в скрипте можно запустить написанную в ходе лабораторной работы программу и вставить результаты), или добавить раздел тестирования, в котором будут указаны результаты работы программы для различных тестовых данных.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Марк Лутц, Python. Карманный справочник, 5-е изд.: Пер. с англ. – Москва: ООО «И.Д. Вильямс», 2015. – 320 с.

УДК 004.3

**Д.М. ЕФИМОВ, А.Д. ПУЗИКОВ,  
М.С. ИГОШИНА, А.Н. САПРЫКИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПРИНЦИП ПРОЕКТИРОВАНИЯ SOFT-ПРОЦЕССОРОВ НА ОСНОВЕ СОВРЕМЕННЫХ ПЛИС**

*Рассматривается проектирование микропроцессорных ядер или soft – процессоров на базе программируемых логических интегральных схем. Приводятся понятия ПЛИС, системы на кристалле, программного процессора. Дается классификация микропроцессорных ядерных и сравнительный анализ их характеристик.*

Как правило, ПЛИС использовались, в качестве дополнительно инструмента к микропроцессору, который помогал ему эффективно реализовывать вспомогательные операций, а также обеспечивал определённый запас гибкости для построения цифровых интерфейсов.

В настоящее время ПЛИС, благодаря модификациям технических характеристик, позволяющих им увеличивать функциональность процессоров, превращаются в “систему на кристалле”.

Данная статья будет посвящена проектированию soft-процессоров, которые обеспечивают проект, созданный в базе ПЛИС, всеми элементами обычной микроконтроллерной системы, включая возможность программирования полученного устройства при помощи языков высокого уровня.

#### **Основные понятия**

Для начала рассмотрим, что такое ПЛИС, “Система на кристалле” (СнК) и soft–процессор.

**ПЛИС (программируемая логическая интегральная схема)** представляет собой интегральную микросхему, применяемую для создания конфигурируемых цифровых электронных схем. Ключевое отличие от обычных цифровых микросхем заключается в том, что логика работы ПЛИС не определяется при изготовлении, а задается посредством программирования (проектирования). В свою очередь, для программирования используются программатор и IDE (отладочная среда), которые позволяют задать желаемую структуру цифрового устройства в виде принципиальной электрической схемы или программы на высокоуровневых языках описания аппаратуры: Verilog, VHDL и др [4].

**Система на кристалле (СнК) или однокристалльная система** – это электронная схема, выполняющая функцию целого устройства и размещенная на одной интегральной схеме. В зависимости от назначения она может оперировать как цифровыми сигналами, так и аналоговыми, аналого-цифровыми, а также частотами радиодиапазона. Довольно часто применяются в портативных и встраиваемых системах.

**Soft–процессор или программный процессор** – это микропроцессорное ядро, получаемое в результате использования логического синтеза. Такие процессоры реализуются с помощью различных полупроводниковых устройств, которые содержат программируемую логику (ПЛИС), например:

- ASIC (специализированные заказные большие интегральные схемы (БИС)),
- FPGA (программируемая пользователем вентиляционная матрица),
- CPLD (сложные программируемые логические устройства).

Большинство подобных систем использует только один программный процессор, но максимальное число ядер ограничивается только ресурсами конкретной используемой ПЛИС [2].

Рассмотрим более детально схемы типа ASIC и FPGA. Для удобства результаты сравнительного анализа основных характеристик сведены в таблицу 1.

Таблица 1 – Сравнение ресурсов ASIC и FPGA

| Характеристика | ASIC                                  | FPGA                     |
|----------------|---------------------------------------|--------------------------|
| Регистры       | Отсутствуют ограничения по размещению | Размещены в ячейках FPGA |

|   |   |   |
|---|---|---|
| <b>Статическая память</b>               | Нет ограничений, используются компоненты, оптимизированные фабрикой   | Аналогична ASIC, количество блоков и разрядность заданы производителем  |
| <b>Логика</b>                           | Нет ограничений, возможна оптимизация путем выбора вентиля с большим быстродействием и энергопотреблением   | Сложность и быстродействие определяются архитектурой LUT. Мелкая гранулярность, требуется схемотехническая оптимизация с учетом характеристик LUT |
| <b>Умножители независимых операндов</b> | Сложность достижения высокого качества реализации, часто необходимо приобретение IP-ядра, оптимизированного под выбранный технологический процесс | Реализованы производителем FPGA и оптимизированы для выбранного производителем технологического процесса  |
| <b>Тактовые сети</b>                    | Требуется самостоятельная реализация тактовых сетей для обеспечения синхронного распространения тактового сигнала                                 | Реализованы производителем FPGA и оптимизированы для выбранного производителем технологического процесса  |

### Виды микропроцессорных ядер для ПЛИС

На данный момент ведущим производителем ядер для ПЛИС является компания Xilinx, а именно их ядра – MicroBlaze и PicoBlaze.

**MicroBlaze** – это soft-процессорное ядро, разработанное компанией Xilinx для использования в FPGA, которое реализуется с помощью стандартной логики и блоков памяти ПЛИС.

Архитектура MicroBlaze схожа с RISC-процессором с DLX-архитектурой. Характерная особенность ядра заключается в том, что оно может выполнять по одной инструкции за такт, сохраняя такую производительность практически в любых условиях [3].

Некоторые операционные системы с упрощенной защитой и виртуальной памятью, могут работать на MicroBlaze без блока управления памятью. С блоком управления памятью возможна работа операционных систем, которые требуют аппаратной поддержки страничной организации памяти и защиты.

**PicoBlaze** – это общее название серии свободно распространяемых soft-процессорных ядер, созданных фирмой Xilinx для своих

ПЛИС (FPGA и CPLD). В основе данной серии лежит архитектура 8-битного RISC-процессора; скорость работы на ПЛИС семейства Virtex 4 может достигать 100 MIPS. Для обращений к периферии процессор имеет 8-битную шину, позволяющую отдельно читать и записывать до 256 портов [1].

Особенности архитектуры обеспечивают наилучшие характеристики для ПЛИС семейства Spartan и Virtex. Кроме того, компанией Xilinx предусмотрено специальное программное обеспечение для трансляции исходного кода на языке Ассемблера и инструментальное программное обеспечение, созданное сторонними организациями (Mediatronix и др.).

Также существует PicoBlaze, представляющее собой альтернативное модифицированное ядро PicoBlaze, распространяемое под лицензией BSD.

Конкуренцию продуктам фирмы Xilinx составляет компания Altera с их ядром Nios II.

**Nios II** – это 32-разрядная микропроцессорная архитектура для soft-микропроцессоров, разработанная специально для ПЛИС фирмой Altera. Nios II является развитием архитектуры Nios и находит применение в различных встраиваемых приложениях, начиная от цифровой обработки сигналов до управляющих систем.

Nios II можно сравнить с MicroBlaze. В отличие от MicroBlaze, Nios II возможно лицензировать для реализации в СБИС (сверхбольшие интегральные схемы) при посредничестве Synopsys Designware.

Сравнительные характеристики микропроцессорных ядер представлены в таблице 2.

Таблица 2 – Сравнение характеристик микропроцессорных ядер.

| Характеристики      | Название  |                    |           |
|---------------------|-----------|--------------------|-----------|
|                     | Nios II   | MicroBlaze         | PicoBlaze |
| Производитель       | Altera    | Xilinx             | Xilinx    |
| Open Source         | Нет       | Нет                | Да        |
| Поддерживаемые шины | Avalon    | PLB, OPB, FSL, LMB | -         |
| Разрядность         | 32-битный | 32-битный          | 8-битный  |

### ПЛИС – аппаратная основа для создания soft-процессоров

Благодаря своим функциональным возможностям ПЛИС позволяют получить макет цифрового устройства в кратчайшие сроки с возможностью неограниченного внесения изменений без затрат материалов или комплектующих.

Проектирование процессора в базе ПЛИС схоже с разработкой программы. Программное и аппаратное обеспечение, необходимое для этого процесса включает в себя:

- 1) ПЛИС и макетная плата на базе ПЛИС;
- 2) САПР ПЛИС, которая поддерживает выбранную микросхему;
- 3) Программатор (jtag-кабели для прошивки и отладки ПЛИС);

Информация о доступности перечисленных устройств и программного обеспечения, относящиеся к двум ведущим компаниям-производителям ПЛИС сведены в таблицу.

Таблица 3 – Аппаратные и программные инструменты для реализации soft-процессора

| <b>Фирма – производитель</b>                                    | <b>Xilinx</b>           | <b>Altera</b>            |
|---|-------------------------|--------------------------|
| <b>Стартовый набор на базе семейства ПЛИС начального уровня</b> | Spartan – 3 Starter Kit | Cyclone – II Starter Kit |
| <b>САПР ПЛИС</b>  | ISE Webpack             | Quartus Web Edition      |
| <b>Программатор</b>   | Parallel Cable III      | ByteBlaster              |

### **Маршруты проектирования ПЛИС**

Ещё одним важным аспектом создания устройства в базе ПЛИС является маршрут проектирования интегральных микросхем. Рассмотрим некоторые наиболее популярные маршруты:

- Классический маршрут;
- Визуальное проектирование;
- Высокоуровневое проектирование.

**Классический маршрут проектирования.** При разработке системы на кристалле, содержащей процессорную часть, по данному маршруту конечным результатом будет являться вычислительная система, программирование для которой осуществляется стандартными программными средствами.

Данный тип маршрута проектирования отображен на рисунке 1.

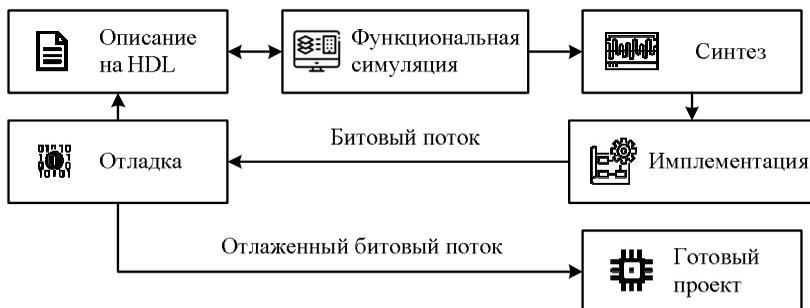


Рисунок 1 – Классический маршрут

Модификацией классического маршрута является проектирование с применением IP-ядер – основной маршрут в САПР Vivado.

IP-ядро можно сравнить с API в программировании. Это ранее разработанная и отлаженная часть проекта, выполняющая определенную функциональность. IP-ядра можно хранить отдельно в специальных репозиториях и включать в свои проекты по мере необходимости.

Расширенная версия классического маршрута представлена на рисунке 2.

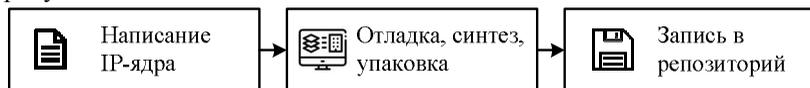


Рисунок 2 – Расширение классического маршрута

Конечный вид маршрута проектирования с использованием IP-ядер показан на рисунке 3.

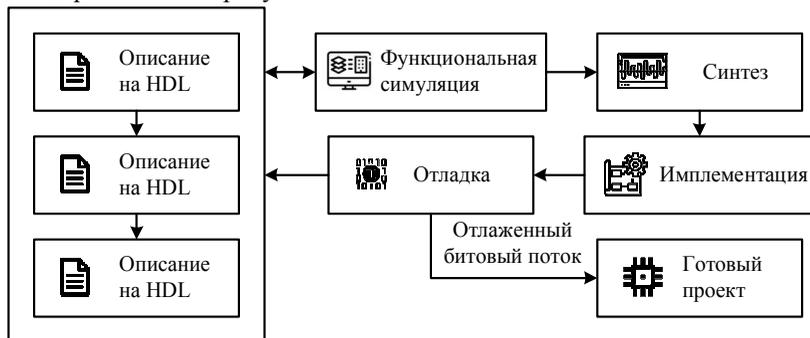


Рисунок 3 – Итоговый вид классического маршрута с применением IP-ядер

**Высокоуровневое проектирование.** В данном случае алгоритм, описываемый на языке C или C++, реализуется в программируемой

логике в виде IP-ядра, поле чего такое IP-ядро может быть включено в любой проект, а весь проект имплементирован по классическому маршруту.

Процесс высокоуровневого проектирования представлен на рисунке 4.

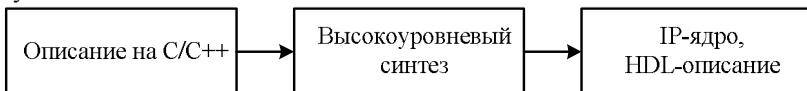


Рисунок 4 – Высокоуровневое проектирование

Одна из версий высокоуровневого маршрута проектирования применима для систем на кристалле.

В данном случае начальным этапом выступает написание программы на языке C или C++. Затем производится профилирование программного кода и определение функций, замедляющих работу системы. С целью повышения скорости работы этих функций они реализуются в программируемой логике и, при необходимости, подвергаются дальнейшей оптимизации.

Такой маршрут представлен на рисунке 5.



Рисунок 5 – Высокоуровневое проектирование для СнК

**Визуальное проектирование с использованием IP-репозитория.** Данный способ проектирования наиболее часто используется при моделировании систем на кристалле, но может применяться и для обычных ПЛИС [4]. Он позволяет создавать иерархические графические диаграммы и заполнять их из репозитория различных компонентами, кроме того предусмотрена возможность добавлять в диаграммы собственные модули на языках Verilog и VHDL.

Процесс визуального проектирования приведен на рисунке 6.

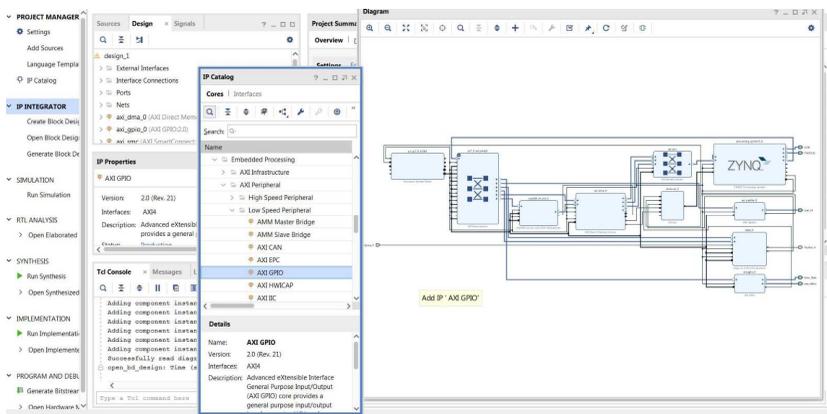


Рисунок 6 – Визуально проектирование

Поскольку проектирование процессорных устройств представляет собой огромную область, которую невозможно подробно описать в рамках одной статьи (или серии статей), данная статья позволяет провести анализ и сравнение характеристик проектирования soft-процессоров на базе различных программируемых логических интегральных схем, а также показывает различные виды и способы проектирования самих ПЛИС.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Булатов, В. Н. Микропроцессорная техника. Схемотехника и программирование: учебное пособие / В. Н. Булатов, О. В. Худорожков. – Оренбург: ОГУ, 2016. – 376 с.
2. Неелова, О. Л. Вычислительная и микропроцессорная техника: учебное пособие / О. Л. Неелова. – Санкт-Петербург: СПбГУТ им. М.А. Бонч-Бруевича, 2020. – 60 с.
3. Смирнов Ю.А. Основы микроэлектроники и микропроцессорной техники : учебное пособие / Ю. А. Смирнов, С. В. Соколов, Е. В. Титов. – 2-е изд., испр. – Санкт-Петербург: Лань, 2022. – 496 с.
4. Ушенина, И. В. Проектирование цифровых устройств на ПЛИС: учебное пособие / И. В. Ушенина. – Санкт-Петербург : Лань, 2022. – 408 с.

УДК 004.4'6

**И.С. ЗАХАРОВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ ДЛЯ РЕШЕНИЯ  
ЗАДАЧИ ТРАССИРОВКИ ПРОВОДНОГО МОНТАЖА**

*В данной статье проводится анализ некоторых алгоритмов трассировки проводного монтажа, описывается суть методов, проводится сравнение вычислительной сложности каждого из рассматриваемых алгоритмов.*

Трассировка представляет собой заключительный этап конструкторского проектирования ЭВА и заключается в поиске линий, соединяющих контакты элементов с одинаковым потенциалом, которые составляют проектируемое устройство. Это является довольно трудоемкой задачей, поскольку проектирование схем соединений связано с многообразием способов конструктивно - технологической реализации соединений, каждый из которых подразумевает под собой использование многообразия специфических критериев оптимизации и ограничений при алгоритмическом решении задачи трассировки. Если данную задачу рассматривать с математической точки зрения, то можно сказать, что она является одной из наиболее сложных задач выбора из множества оптимальных решений. [1]

Как правило, имеются два типа задач трассировки:

- трассировка проводного монтажа;
- трассировка печатного монтажа (соединений).

Далее будут рассмотрены алгоритмы для решения задачи трассировки проводного монтажа, который используют при объединении модулей, начиная со второго уровня иерархии, то есть между ячейками, в блоках, рамках. Самый распространённый метод оптимизации трасс проводных соединений основан на использовании алгоритмов построения минимальных связывающих деревьев. [2] В данной статье рассмотрим некоторые алгоритмы, которые опираются на указанный метод, а именно алгоритм Краскала, алгоритм Прима, алгоритм Дейкстры и алгоритм Борувка.

Алгоритм Краскала является «жадным» алгоритмом. «Жадные» алгоритмы берут решение из следующего шага, в котором выбирается наилучшее локальное решение из множества возможных, не принимая

во внимание глобальную структуру. Так как мы ищем дерево с минимальным весом, то алгоритм последовательно выбирает ребра с наименьшим весом, которые увеличат количество вершин дерева.

Алгоритм начинает работу с тривиального графа  $T = (V, \emptyset)$ . Упорядочим ребра в порядке неубывания их весов и будем добавлять ребра в  $T$  по порядку. Очередное ребро добавляется в  $T$  и убирается из списка, если это не приводит к образованию цикла. В противном случае оно просто удаляется из списка и рассматривается следующее ребро списка. Это повторяется до тех пор, пока число ребер в  $T$  не составит  $n - 1$ . Построенное дерево – КСД (кратчайшее связывающее дерево).

Трудоёмкость упорядочивания ребер –  $O(E \log E)$ . Очевидно, что при построении дерева в худшем случае будут рассмотрены все  $E$  ребер графа. Следовательно, этот алгоритм более подходит для построения КСД в графе с небольшим количеством ребер. Заметим, что  $|E| < |V|^2$ , поэтому  $\log|E| = O(\log V)$  и время работы алгоритма Краскала можно записать как  $O(E \log V)$

Так же, как и алгоритм Краскала алгоритм Прима является «жадным». Идея алгоритма принадлежит Приму, эффективную технику реализации предложил Дейкстра.

В алгоритме Прима используется такой же принцип связи ближайших вершин, что и в алгоритме Краскала. Но на каждом шаге в дерево включается ближайшая изолированная вершина. В основе данного метода лежат две теоремы:

*Теорема 1:* Каждая вершина кратчайшего связывающего дерева непосредственно связана хотя бы с одной ближайшей вершиной.

*Теорема 2:* Каждый измеряемый фрагмент поддеревя связан по крайней мере с одним из изолированных фрагментов кратчайшим ребром.

Исключение возникновения циклов обеспечивается посредством вычеркивания столбцов. При этом на каждом шаге добавляется одна компонента связности. [4]

Сложность алгоритма  $O(V^2)$ . Асимптотическое время работы алгоритма Прима можно улучшить за счет применения фибоначиевых пирамид. Фибоначиева пирамида представляет собой набор деревьев, упорядоченных в соответствии со свойством неубывающей пирамиды. Следовательно, для реализации очереди с приоритетами  $Q$  общее время работы алгоритма Прима улучшается до  $O(E + V \log V)$ . [3]

Голландский ученый Эдсгер Дейкстра предложил алгоритм, который способен находить все возможные кратчайшие пути из одной

начально заданной вершины графа до всех остальных. Данный алгоритм не лишен минусов, так как здесь существует невозможность обработки всех графов, имеющих ребра с отрицательным весом.

Если вершины хранятся в простом массиве и для поиска минимума используется алгоритм линейного поиска, временная сложность алгоритма Дейкстры составляет  $O(V * V + E) = O(V^2)$ . Если же очередь с приоритетами была реализована на основе фибоначчиевых куч, получается наилучшая оценка сложности  $O(V \log V + E)$ . [2]

Алгоритм Борувки – это алгоритм поиска минимального остовного дерева в взвешенном связанном неориентированном графе. На первом этапе каждая вершина принадлежит отдельному дереву.

Таким образом, на следующем шаге каждое дерево выбирает минимальное ребро, соединяющее текущее дерево с другим. Если минимальных ребер несколько, то выбирается ребро с наименьшим порядковым номером, это очень важно для правильной работы алгоритма. На следующем этапе все выбранные ребра добавляются в КСД, и алгоритм повторяется до тех пор, пока не останется одно дерево. Оставшееся дерево и будет искомым кратчайшим связывающим деревом.

Можно заметить, что на каждой итерации алгоритма Борувки число поддеревьев сокращается как минимум вдвое. Следовательно, алгоритм в худшем случае выполнит  $O(\log V)$  итераций. На каждой итерации мы в худшем случае просмотрим все ребра. Получается итоговая оценка алгоритма Борувки  $O(E \log V)$ . [2]

Таким образом, в данной статье были разобраны четыре алгоритма для решения задачи трассировки проводного монтажа, которые основываются на построении кратчайшего связывающего дерева. Алгоритм Краскала, как и алгоритм Борувки, имеют одинаковую вычислительную сложность. Вычислительную сложность алгоритмов Прима и Дейкстры можно повысить при помощи использования фибоначчиевых куч. Все вышесказанное сведено в таблицу 1.

Таблица 1 – Сравнения алгоритмов построения кратчайшего связывающего дерева

| Алгоритм                      | Вычислительная сложность |                                      |
|-------------------------------|--------------------------|--------------------------------------|
|                               | Матрица смежности        | Списки смежности + фибоначчиева куча |
| Краскала<br>(J.Kruskal, 1956) | $O(E \log V)$            |                                      |

|                               |                         |                   |
|-------------------------------|-------------------------|-------------------|
| Дейкстры<br>(E.Dijkstra,1959) | $O(V * V + E) = O(V^2)$ | $O(V \log V + E)$ |
| Прима<br>(R.Prim,1957)        | $O(V * V + E) = O(V^2)$ | $O(E + V \log V)$ |
| Борувка<br>(O.Bohuvka,1926)   | $O(E \log V)$           |                   |

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. В. М. Курейчик. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для вузов. – М.: Радио и связь. 1990. – 352с.

2. Алгоритма: построение и анализ, 2-е издание / Кормен, Томас Х., Лейзерсон, Чарльз И., Ривест, Роналд Л., Штайн, Клиффорд.; пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 1296 с.

3. Задачи маршрутизации: учеб. пособие / А. И. Ерзин, Ю. А. Кочетов; Новосиб. гос. ун-т. – Новосибирск: РИЦ НГУ, 2014. – 95 с.

4. Скиена С. Алгоритма. Руководство по разработке. – 2-е изд.: Пер. с англ. – Спб.: БХВ – Петербург. 2011. – 720с.

УДК 004.02

### В.О. ЗИНОВА, М.А. СЕРГЕЕВ

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### ИССЛЕДОВАНИЕ И РЕАЛИЗАЦИЯ АЛГОРИТМА ОБРАТНОГО РАЗМЕЩЕНИЯ

*Рассматривается одна из типовых конструктивных задач - размещение модулей в монтажном пространстве.*

Построение вычислительных модулей предполагает особую иерархию действий, а именно расслоение проектирование на конструктивные уровни. Расположение модулей в порядке возрастания: элемент, ячейка, панель, шкаф, ЭВС. Размещение – задача определения такого местоположения элементов в заданном монтажном пространстве, при котором наилучшим образом удовлетворяются некоторые требования [1].

В данной работе рассматривается метод обратного размещения, что является частью размещения модулей в монтажном пространстве. Он применяется в производстве печатных плат, разработке сетей, научной деятельности.

Классификация алгоритмов размещения приведена на рисунке 1.



Рисунок 1 – Классификация алгоритмов размещения

Пусть имеется регулярное монтажное пространство с уже фиксированными позициями:  $p = \{p_1, p_2, \dots, p_m\}$ , и имеется количество  $n < m$  элементов для размещения.

Считаем, что длина связей определяется расстоянием между геометрическими центрами соответствующих позиций, то есть:

$$d_{kq} = \sqrt{(x_k - x_q)^2 + (y_k - y_q)^2}. \quad (1)$$

Сформулируем математическую постановку задачи размещения:  $L(\Pi)$  – суммарная длина межэлементных соединений, соответствующая некоторому варианту размещения  $\Pi$ .

$ES$  – множество директивно размещенных элементов.  $S$  – множество индексов директивно размещенных элементов.

Рассмотрим некоторый произвольный заранее не размещенный элемент. Суммарная длина связей произвольного, заранее не размещенного элемента, со всеми директивными элементами.

$$l_{i\Pi(i)} = \sum_{S \in S} c_{iS} d_{\Pi(i)\Pi(S)}, \quad (2)$$

где  $\Pi(i)\Pi(S)$  – расстояние между  $i$  и  $S$ ;

$c_{iS}$  – элемент матрицы смежности ВНГ;

$d_{\Pi(i)\Pi(S)}$  – расстояние между  $\Pi(i)$  и  $\Pi(S)$ .

Тогда суммарная взвешенная длина межсоединений:

$$L_{\min}(\Pi) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n c_{ij} d_{\Pi(i)\Pi(j)} + \sum_{i=1}^n l_{i\Pi(i)}. \quad (3)$$

Таким образом, необходимо найти вариант размещения  $\Pi^* = \{\Pi^*(1), \Pi^*(2), \dots, \Pi^*(m)\}$ , при котором обеспечивается минимум целевой функции.

В методе обратного размещения для каждого из неразмещенных элементов рассчитывается некоторая оценка. Такая же оценка рассчитывается и для каждого посадочного места. Все элементы и посадочные места упорядочиваются, и осуществляется одновременное размещение всех элементов в позиции [1].

Даны:

Матрица смежности ВНГ –  $C = \|c_{ij}\|$  размерностью  $n \times n$  и матрица длины между позициями –  $D = \|d_{ij}\|$  размерностью  $m \times m$ .

Для каждого элемента  $e_i$  вычисляется суммарное число связей  $i$ -го элемента:

$$C_i = \sum_{\substack{e \in E \\ j \neq i}} c_{ij}. \quad (4)$$

Для каждого посадочного места высчитывается суммарная длина расстояний:

$$D_i = \sum_{\substack{p \in P \\ i \neq j}} d_{ji}, \quad (5)$$

где  $p$ - число посадочных мест.

В итоге все оценки  $C_i$  упорядочиваются по возрастанию, а  $D_i$  по убыванию.

Отличие метода обратного размещения от других алгоритмов размещения – его простота. Он может быть использован даже при ручной методике разработки схем для грубого начального варианта размещения элементов. [2].

Идея алгоритма проста. Осуществляют предварительную оценку каждого из размещаемых элементов и каждой свободной позиции. Далее производится упорядочивание элементов по возрастанию или убыванию введенных характеристик, и затем все элементы размещаются одновременно.

Временная сложность алгоритма равна  $O(a_n)$ .

В проекте использованы процедуры общего назначения:

`writeMatrix`, `getRandom`, `enterWithKeyboard`, `getFromFile`, `saveToFile`, `swap`, `order`, `multiplication`, `calculateLength`.

Алгоритм:

1 - ручной ввод, 2 - ввод из файла, 3 - случайные числа.

C - матрица смежности; D - матрица расстояний.

Если 1 – вводим матрицы C и D с клавиатуры. Размерности матриц должны совпадать. Если не совпадают то: `if(length(C) <> length(D)) then begin`. Если 2 - вводим название файла с матрицами C и D. Иначе- получаем случайное значение размерности матрицы. Вызываем функцию `getRandom`, передаем в нее матрицу C и размерность n для заполнения случайными числами. Вызываем функцию `getRandom`, передаем в нее матрицу D и размерность n для заполнения случайными числами. Устанавливаем списку  $C_i$  размерность n, устанавливаем списку  $D_i$  размерность n. Увеличиваем значение в списке  $C_i$  на значение из матрицы. сортируем  $C_i$  по возрастанию с помощью процедуры `order`. Сортируем  $D_i$  по убыванию. Устанавливаем размерность n для массива перестановок (номеров позиций). если ключ из списка  $D_i$  равен i, то формируем список `keys` в порядке перестановок.

Программа предназначена для автоматизации одновременного размещения всех элементов в позиции. Результаты работы данной программы, реализующей метод обратного размещения, могут быть использованы в качестве оптимального варианта трассировки проводных соединений печатной платы. Работа с программой доступна всем пользователям с установленными правами доступа.

Описание задачи. Требуется разместить элементы в определенные позиции платы, используя метод обратного размещения. Входными данными могут быть введенные вручную или загруженные из текстовых файлов матрицы. В результате анализа выходными данными будут получившаяся матрица с распределенными в свои позиции элементами, оценка длины для начальной и конечной матрицы.

Укрупненная схема алгоритма обратного размещения представлена на рисунке 2.

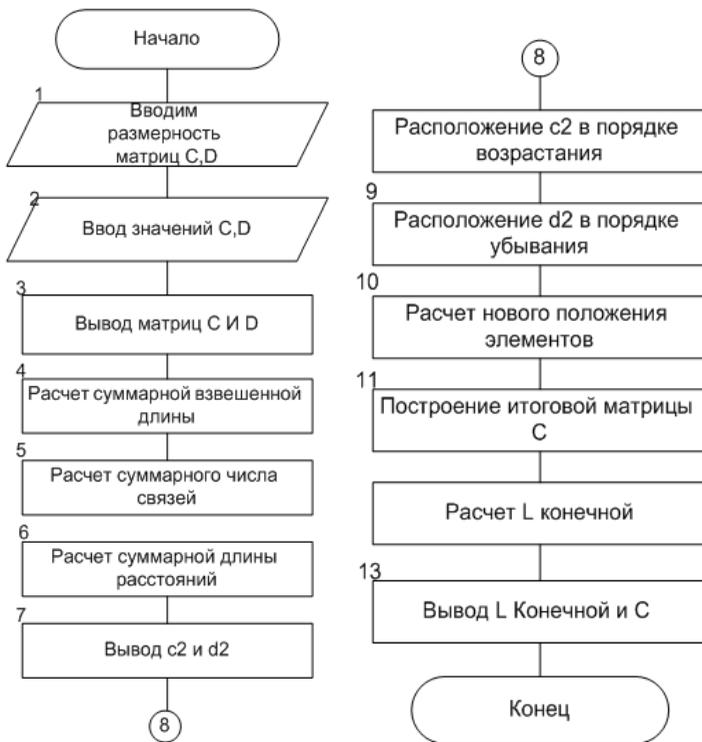


Рисунок 2 – Укрупненная схема алгоритма

Результат программы:

Требуется загрузить случайную матрицу. Ввод 3.

Матрица C:

```

0 6 2 4 3
6 0 4 4 3
2 4 0 3 4
4 4 3 0 5
3 3 4 5 0
  
```

Матрица D:

```

0 1 0 4 5
1 0 6 3 2
0 6 0 6 0
4 3 6 0 4
5 2 0 4 0
  
```

C: {(0,15),(1,17),(2,13),(3,16),(4,15)};

D: {(0,10),(1,12),(2,12),(3,17),(4,11)};

L начальная будет равна – 117.

C: {(2,13),(0,15),(4,15),(3,16),(1,17)};

D: {(3,17),(1,12),(2,12),(4,11),(0,10)};

L конечная будет равна – 110.

Итоговая матрица С:

|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 6 | 3 | 4 | 4 |
| 6 | 0 | 3 | 2 | 4 |
| 3 | 3 | 0 | 4 | 5 |
| 4 | 2 | 4 | 0 | 3 |
| 4 | 4 | 5 | 3 | 0 |

Продемонстрирован результат программы, реализующей алгоритм обратного размещения, при котором осуществляются предварительные оценки каждого из размещаемых элементов и каждой свободной позиции, после чего все элементы размещаются одновременно.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Дендобренко Б.Н., Малика А.С. Автоматизация конструирования РЭА. Учебник для вузов. - М.: Высш. школа, 1980. - 384с.
2. Зыков А.Г., Поляков В.И. Алгоритмы конструкторского проектирования ЭВМ. – СПб: Университет ИТМО, 2014.

УДК 004.048

**М.А. ИВАНЧИКОВА, А.Ю. ШМАКОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### МЕХАНИЗМ ОБНАРУЖЕНИЯ И РАСПОЗНАВАНИЯ ЛИЦ НА ИЗОБРАЖЕНИЯХ С ПРИМЕНЕНИЕМ НЕЙРОННЫХ СЕТЕЙ

*Рассматриваются технологии компьютерного зрения и принципы обучения нейронной сети, применяемые для обнаружения и распознавания лица человека на фотографии.*

В настоящее время обнаружение лиц и их распознавание все чаще используется в обычной жизни человека. Яркими примерами данных технологий могут служить такие повседневные вещи как, идентификация личности для разблокирования смартфона, фокусировка камеры телефона, когда необходим фокус на конкретном предмете или лице, подтверждение личности в системах контроля безопасности. Однако актуальной проблемой компьютерного зрения остается обнаружение объектов на изображении и видео в автоматическом режиме. Рассмотрим решение данной задачи, реализованное с помощью языка программирования Python, библиотеки компьютерного зрения OpenCV, и библиотеки Dlib,

которая содержит алгоритмы машинного обучения, а так же дополнительные инструменты для работы с изображением лиц.

Первым этапом технологии распознавания является обнаружение лица на фото. Обнаружение – это извлечение признаков из изображения для определения наличия на нем лица человека. Результатом обнаружения являются обозначенные границы найденного лица, формирующие область, из которой в дальнейшем будут извлекаться признаки черт лица.

После выполнения процедуры поиска границ на изображении выполняется эмбединг (embedding) – преобразование изображения в числовой вектор основных признаков лица.

$$F \left( \text{img} \right) = (0.201 \ 0.103 \ \dots \ 0.0021)$$

Рисунок 1 – Преобразование изображения в вектор признаков

Вторым этапом является распознавание лица в выделенной области. Для этого строят и обучают нейронную сеть. В качестве примера можно рассмотреть фотографию одного человека, но в различные промежутки времени. Будет наблюдаться естественное изменение черт лица от фотографии к фотографии. Вектора признаков фото будут отличаться, но совсем не значительно. На рисунке 2 показан пример векторов изображений до обучения.

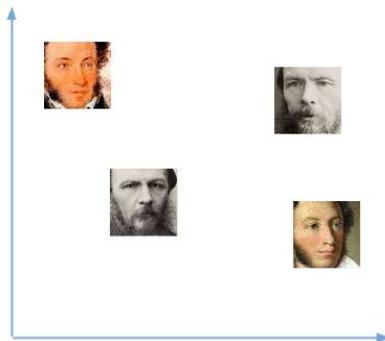


Рисунок 2 – Векторы изображений до обучения

В процессе обучения, нейронная сеть учится группировать вектора изображений, которые ближе друг к другу, а значит определяют лицо одного и того же человека. На рисунке 3 представлены расположения векторов изображения после обучения.

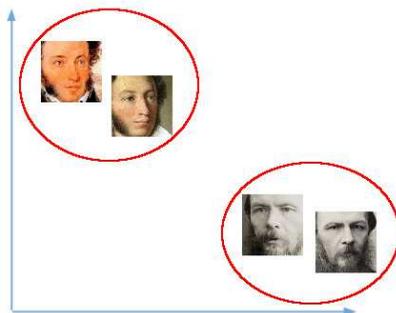


Рисунок 3 – Векторы изображений после обучения

Следующий этап – валидация. После обучения нейронной сети подготовлены данные о признаках лиц в отдельном файле, далее необходимо распознать контрольное изображение, которого нет в исходных данных для обучения. Для него вычисляется вектор признаков и сравнивается с уже имеющимися данными. Лицо будет считаться распознанным, если его вектор признаков совпал с имеющимся в исходном наборе или близок к нему. В противном случае изображение считается не распознанным. Графический пример этапа валидации показан на рисунке 4.

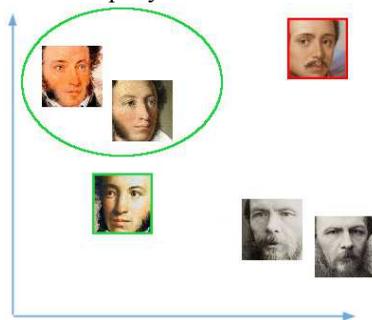


Рисунок 4 – Сравнение векторов признаков

При передаче фотографии А.С. Пушкина нейронная сеть смогла распознать фотографию, потому что для лица существовал набор числовых векторов и эмбединг новой фотографии был близок к имеющемуся набору, а для М.В. Ломоносова набора числовых векторов не было, поэтому сеть не смогла распознать лицо. Следует отметить, что для достижения хорошего результата точности при распознавании фотографий необходим достаточно большой набор входных изображений.

Рассмотрим программную реализацию алгоритма обнаружения лица на фотографии:

```
# Импорт необходимых библиотек
import cv2
import dlib
import os
import sys

# Путь к папке, в которую будут сохранены изображения лиц
face_out_dir = './face_out_dir'
# Путь к папке, в которой находится входной набор
фотографий
face_in_dir = './face_in_dir'
# Размерность выходных изображений лиц
size = 64

# Если не существует папка для выходных изображений лиц,
создаем ее
if not os.path.exists(face_out_dir):
    os.makedirs(face_out_dir)

# Детектор лиц
detector = dlib.get_frontal_face_detector()

index = 1
for (path, dirnames, filenames) in os.walk(face_in_dir):
    for filename in filenames:
        if filename.endswith('.jpg'):
            print('Обрабатываем изображение №%s' % index)
            img_path = path+'/'+filename
            # Чтение фото из файла
            img = cv2.imread(img_path)
            # Преобразовываем изображение в оттенок серого
            gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
            # Получим результат расположения лица на
            dets = detector(gray_img, 1)

            # i - индекс, d - значение
            # left - расстояние от левой стороны лица до левой границы
изображения;
            # right - расстояние от правой стороны лица до левой
границы изображения.
            # top- расстояние между верхней части лица и верхней
границей изображения;
            # bottom: расстояние от нижней части лица до верхней
границы изображения.
            for i, d in enumerate(dets):
                top = d.top() if d.top() > 0 else 0
                bottom = d.bottom() if d.bottom() > 0 else 0
                left = d.left() if d.left() > 0 else 0
                right = d.right() if d.right() > 0 else 0
```

```
face = img[top:bottom,left:right]
# Меняем размер выходного изображения лица
face = cv2.resize(face, (size,size))
cv2.imshow('image',face)
# Сохраняем изображение с найденным лицом
cv2.imwrite(face_out_dir+'/'+str(index)+'.jpg', face)
index += 1

key = cv2.waitKey(30) & 0xff
if key == 27:
    sys.exit(0)
```

В качестве примера возьмем 4 фотографии А.С. Пушкина. На рисунке 5 представлено содержимое папки face\_in\_dir.



Рисунок 5 – Исходные изображения

После запуска программы ожидаем завершения процедуры обнаружения лиц на фото. Результат работы программы – изображения с выделенной областью лица, которые помещаются в папку face\_out\_dir, их пример показан на рисунке 6.



Рисунок 6 – Результат выполнения программы

В ходе дальнейшего изучения технологии компьютерного зрения планируется реализация процедуры распознавания и исследование алгоритмов и методов машинного обучения, которые могут быстрее и более точно определять объекты на изображении.

*Работа выполнена при финансовой поддержке стипендии Президента РФ СП-2459.2021.5.*

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. С. Хайкин. Нейронные сети: полный курс. 2-е изд. М., "Вильямс", 2006. S.Haykin. Neural Networks and Learning Machines. 3rd Edition. Pearson, 2018.

2. А.А. Ежов, С.А. Шумский. Нейрокомпьютинг и его применения в экономике и бизнесе. М., МИФИ, 1998
3. В.В. Круглов, М.И. Дли, Р.Ю. Голунов. Нечёткая логика и искусственные нейронные сети. Физматлит, 2001.
4. С.С. Aggarwal. Neural Networks and Deep Learning. A Textbook. Springer International Publishing AG, 2018. DOI 10.1007/978-3-319-94463-0 ISBN 978-3-319-94462-3.

УДК 004.3

**М.С. ИГОШИНА, А.Н. САПРЫКИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ОСОБЕННОСТЬ ПРОЕКТИРОВАНИЯ ДВУХТАКТНОГО Т-ТРИГГЕРА В МУЛЬТИСРЕДЕ INTEL QUARTUS PRIME**

*В статье рассматривается способ реализации двухтактного Т-триггера в базе ПЛИС Cyclone IV E. Дается определение ПЛИС, кратко описываются преимущества выбранной интегральной схемы. Кроме того, в работе представлена VHDL-модель проектируемого устройства.*

#### **ПЛИС**

Программируемая логическая интегральная схема (ПЛИС, programmable logic device, PLD) — электронный компонент, используемый для создания цифровых интегральных схем. Функции ПЛИС, в отличие от обычных микросхем, задаются посредством программирования (проектирования), а не определяются при производстве. Для задания требуемой структуры цифрового устройства используются программатор и отладочная среда (IDE) [2]. В результате формируется принципиальная электрическая схема или программа на специальных языках описания аппаратуры: Verilog, AHDL, VHDL и др.

В настоящее время ПЛИС всё больше востребованы у современных разработчиков. Они могут выступать, как в качестве дополнительного инструмента моделирования, так и самодостаточной основы, позволяющей реализовывать сложные цифровые проекты. Отличительными чертами таких схем являются:

- Мощная логика;
- Внутрисистемное программирование;
- Широкий ценовой диапазон.

Благодаря вышеперечисленным характеристикам, а также разнообразному симбиозу своего технического и функционального испол-

нения ПЛИС занимают лидирующую строчку в списке гибких и универсальных платформ проектирования.

### ***Cyclone IV E***

ПЛИС линейки Cyclone IV, одной из самой популярной компании – производителя Altera, отличаются сравнительно низкой стоимостью, малой мощностью, а также наличием приемопередатчика [3]. Как правило, микросхемы используются для разработки высокопроизводительных систем, где определяющим критерием является высокая пропускная способность при низких затратах.

Ключевые характеристики Cyclone IV:

- От 6 до 150 тыс. логических элементов;
- До 6,3 Мб встроенной памяти;
- До 360 множителей  $18 \times 18$  для приложений с интенсивной обработкой DSP;
- Приложения для соединения протоколов общей мощностью менее 1,5 Вт.

Одна из версий данной ПЛИС – это схема Cyclone IV E, структура которой приведена на рисунке 1. Для проектирования двухтактного T-триггера, в качестве основы, выберем Cyclone EP4CE6E2208N.

ПЛИС состоит из логических элементов, основанных на четырех входных таблицах поиска (LUT), блоков памяти и множителей.

Каждый блок памяти обеспечивает 9 Кбит встроенной памяти SRAM, использующиеся для хранения данных, которые загружаются на плату при каждом включении устройства.

Встроенные блоки умножения могут реализовывать  $18 \times 18$  или два множителя  $9 \times 9$  в одном блоке. Предлагается полный набор IP-адресов DSP, конечная импульсная характеристика (FIR), быстрое преобразование Фурье (FFT) и функции генератора с числовым управлением (NCO).

Устройство ввода-вывода микросхем поддерживает программируемое удержание шины, подтягивающие резисторы, задержку, мощность привода и управление скоростью нарастания для оптимизации целостности сигнала, а также такие стандарты, как LVTTTL, LVCMOS, SSTL, HSTL, PCI и т.д.

Микросхемы включают до 30 глобальных тактовых сетей (GCLK) и до восьми PLL для обеспечения надежного управления тактовой частотой и синтеза.

Осуществляется поддержка интерфейсов SDR, DDR, DDR2 SDRAM и QDR II SRAM на верхней, нижней и правой сторонах устройства, кроме того они могут охватывать две или более сторон, чтобы обеспечить более гибкую конструкцию платы.

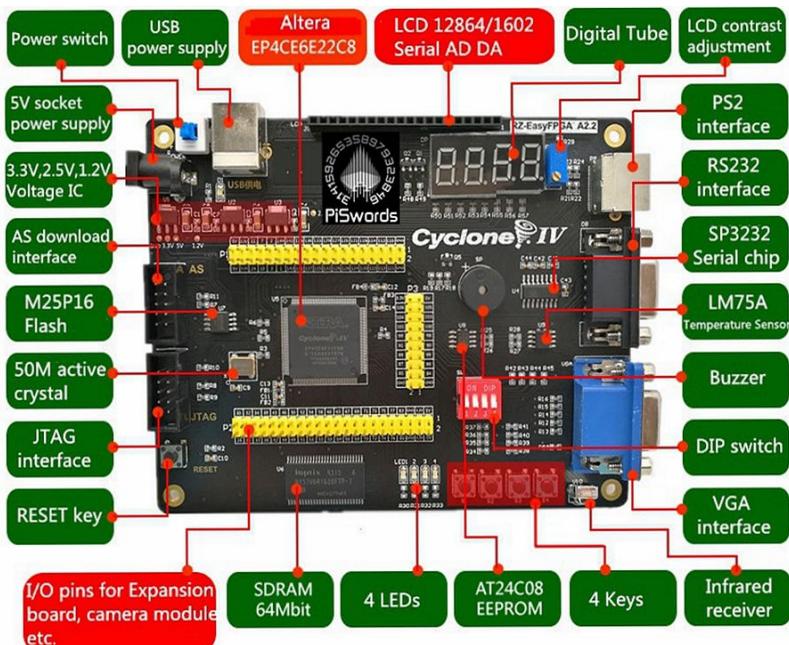


Рисунок 1 – Структура Cyclone IV E

### *Intel Quartus Prime*

Компания Altera предлагает собственное программное обеспечение для реализации сложных устройств.

Intel Quartus Prime – мультисреда, осуществляющая сквозное проектирования цифровых систем на кристаллах ПЛИС, а также продолжение модуля Quartus II, существовавшего до версии 15.0.

Платформа объединяет в себе моделирование, синтез, размещение элементов, трассировку соединений и верификацию, поддержку высокоуровневых языков описания аппаратуры – Verilog и VHDL.

Логическая схема, разработанная в Intel Quartus Prime, называется проектом. Проект может иметь иерархическую структуру, т.е. он состоит из множества файлов-модулей, где главный модуль (или объект верхнего уровня иерархии) содержит несколько дополнительных модулей.

Мультисреда позволяет сохранять в одном каталоге иерархическую структуру, а также всю техническую информацию, касающуюся выбранной микросхемы (например, назначение выводов, предыдущие варианты трассировки логической схемы, конфигурация проекта и т.д.). Такие файлы создаются в единой папке проекта, что обеспечива-

ет возможность легко переносить или копировать информацию в другие каталоги, компьютеры.

Процесс проектирования в системе Intel Quartus Prime включает в себя следующие операции:

*Ввод проекта (Design Entry)*. Схема проекта задается графическим способом или при помощи высокоуровневых языков описания аппаратуры (Verilog HDL, VHDL и др).

*Синтез (Synthesis)*. Проект синтезируется в схему, которая состоит из логических элементов (ЛЭ) и логических блоков (ЛБ) в микросхеме ПЛИС.

*Функциональное моделирование (Functional Simulation)*. Синтезируемая схема тестируется на предмет корректности функционирования во встроенном симуляторе, который моделирует зависимость состояния (выбранных разработчиком) сигналов схемы от времени. Это моделирование не учитывает временные задержки сигналов – логическими гонками между ЛЭ/ЛБ микросхемы ПЛИС.

*Трассировка (Fitting)*. Трассировщик выбирает оптимальное размещение и соединение ЛЭ/ЛБ, определенных в списке соединений (netlist) реальной микросхемы ПЛИС, а также «маршрут движения» в чипе для реализации необходимых связей между заданными ЛЭ/ЛБ.

*Временной анализ (Timing Analysis)*. Анализируются задержки распространения сигналов вдоль различных путей трассируемой схемы с целью обнаружения наличия или отсутствия логических гонок.

*Временное моделирование (Timing Simulation)*. Схема проходит тестирование для проверки функциональности и временных ограничений, учитываются реальные задержки выбранных сигналов.

*Программирование и конфигурация (Programming and Configuration)*. Разработанная схема размещается в ПЛИС при помощи программирования электронных связей между конфигурируемыми ЛЭ/ЛБ, что реализуется путем передачи конфигурационного файла с компьютера в микросхему ПЛИС.

### **Триггеры**

Триггер представляет собой логическое электронное устройство, способное оставаться длительное время в одном из двух устойчивых положений: «0» или «1». Составляет основу функциональных узлов последовательного типа. Основное назначение – хранение двоичной информации [1].

По схемотехническому исполнению и назначению выделяют следующую классификацию триггеров:

- По выполняемым функциям: RS, D, T, JK.

- По типу синхронизации: синхронные – информация со входа заносится принудительно под воздействием синхронизирующего импульса; асинхронные – информация со входа заносится без синхронизирующего импульса.

- По числу ступеней запоминания: одноктактные (одноступенчатые), двухтактные (двухступенчатые).

- По виду управления: со статическим управлением (потенциалом), с динамическим управлением.

*T-триггер* или триггер со счетным входом – логическое устройство с двумя устойчивыми состояниями и одним входом Т, изменяющее свое состояние на противоположное при поступлении на вход Т управляющего или счетного сигнала. Может быть построен на основе двухтактного RS-триггера. Структура двухтактного сканируемого Т-триггера с установкой представлена на рисунке 2, значения функционирования сведены в таблицу 1, таблица соответствия портов показана на рисунке 3.

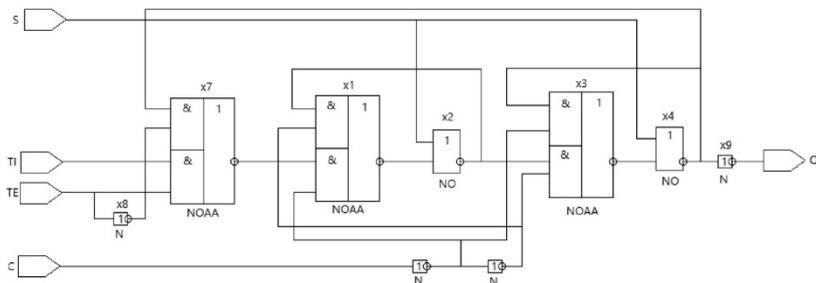


Рисунок 2 – Структура Т-триггера

Таблица 1 –Таблица истинности Т-триггера.

| S | Te | Ti | C    | Q           |
|---|----|----|------|-------------|
| 1 | -  | -  | [--] | 1           |
| 0 | 1  | -  | [01] | (Ti)        |
| 0 | 0  | -  | [01] | $\wedge(Q)$ |
| 0 | -  | -  | [1-] | N           |
| 0 | -  | -  | [00] | N           |

Обозначения в таблице функционирования D-триггера:

- Через «-» обозначено любое из (0,1) значение сигнала;
- В квадратных скобках показывается изменение сигнала (обычно синхросигнала), например, через [01] обозначается передний фронт сигнала (сигнал меняется из 0 в 1);

- Через [10] обозначается задний фронт сигнала (сигнал меняется из 1 в 0);
- Символ N обозначает неизменяемое (предыдущее) значение сигнала;
- Символ ^ является знаком инверсии (отрицания).

| tatl | From    | To      | Assignment Name | Value  | Enabled | Entity |
|------|---------|---------|-----------------|--------|---------|--------|
| 1    | ✓       | in CLK  | Location        | PIN_24 | Yes     |        |
| 2    | ✓       | out Q   | Location        | PIN_3  | Yes     |        |
| 3    | ✓       | in S    | Location        | PIN_88 | Yes     |        |
| 4    | ✓       | in Te   | Location        | PIN_90 | Yes     |        |
| 5    | ✓       | in Ti   | Location        | PIN_89 | Yes     |        |
| 6    | <<new>> | <<new>> | <<new>>         |        |         |        |

Рисунок 3 – Таблица соответствия портов Т-триггера

***VHDL-модель структурной схемы D-триггера:***Код файла **main.vhd**

library ieee;

use ieee.std\_logic\_1164.all;

entity main is

port(

CLK: in std\_logic; Q: out std\_logic

S, Te, Ti: in std\_logic );

end main;

architecture arch of main is

component trigger

port (CLK, S, Te, Ti: in std\_logic;

Q: out std\_logic);

end component;

begin

p:trigger port map(CLK, S, Te, Ti, Q);

end arch;

Код файла **trigger.vhd**

library ieee;

use ieee.std\_logic\_1164.all;

entity trigger is

port (CLK: in std\_logic;

S: in std\_logic; Q: out std\_logic;

Te: in std\_logic; Ti: in std\_logic);

end trigger;

architecture trigger\_arch of trigger is

```
signal value: std_logic:='0';
begin
process (CLK)
begin
  if rising_edge(CLK) then
    if (S='0' and Te = '1' and (Ti='1' or Ti='0')) then
value <= (not value);
    elsif (S='0' and Te = '0' and (Ti='1' or Ti='0'))
then value <= Ti;
    else value <= '0';
    end if;
Q <= value;
end if;
end process;
end trigger_arch;
```

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Музылева И. В. Триггеры и устройства на их основе: учебное пособие / И. В. Музылева, А. А. Муравьев. – Липецк: Липецкий ГТУ, 2020. – 66 с.
2. Наваби З. Проектирование встраиваемых систем на ПЛИС / З. Наваби; перевод с английского В. В. Соловьева. – Москва: ДМК Пресс, 2016. – 464 с.
3. Ушенина И. В. Проектирование цифровых устройств на ПЛИС: учебное пособие / И. В. Ушенина. – Санкт-Петербург: Лань, 2022. – 408 с.

УДК 004.384

**А.А. КАЛИНКИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### СТРУКТУРА И ПРИНЦИП РАБОТЫ «УМНОЙ РОЗЕТКИ»

*В данной статье проводится рассмотрение и анализ структуры и принципов работы «Умной розетки».*

В наше время прогресс не стоит на месте и все стремительнее новые технологии внедряются в нашу жизнь. Система управления «Умный дом» давно уже не новинка на современном рынке. Существует огромное количество различных способов автоматизировать

управление электрическими приборами будь это в квартире, либо же в загородном доме.

Одним из составляющих элементов любого «Умного дома», является «Умная розетка». Впервые эти уникальные устройства были спроектированы в научном институте Германии [1]. Сейчас же насчитывается огромное количество брендов, выпускающих «Умные розетки», отличающиеся по своему исполнению и функционалу.

Стоит начать с определения. Что же такое «Умные розетки» и как они работают? Умные розетки – это устройства, которые позволяют управлять различными бытовыми приборами на расстоянии [2]. К примеру, вы забыли выключить утюг дома, а вернуться уже нет возможности, в этом случае нажатием кнопки в приложении на своём смартфоне вы сможете отключить питание в розетке и тем самым спасти квартиру от пожара. Либо же вы можете вскипятить чайник, не вставая с дивана. Принцип работы такого устройства основан на замыкании-размыкании электрической цепи с помощью электромагнитного реле и контроллера, который собственно управляет этим реле, а также модуля, позволяющего управлять розеткой на расстоянии. Ниже на рисунке 1 приведена принципиальная схема «Умной розетки». Составляющие данного гаджета позволяют осуществлять включение и выключение бытовых приборов благодаря сетевым протоколам. Используя определенное программное обеспечение производится передача сигнала контроллеру, за счёт чего выполняются заданные команды.



Рисунок 1 – Принципиальная схема «Умной розетки»

Практически все «Умные розетки» подходят под разъемы современной бытовой техники и имеют ограничения по мощности 3 Квт. Такой мощности вполне достаточно даже для подключения сплит-системы, бойлеров и радиаторов.

Существуют различные варианты исполнения таких розеток. К самым популярным и более простым можно отнести изделия, которые имеют только один разъем, но также существуют более навороченные розетки, представляющие собой сетевой фильтр с тремя, четырьмя разъемами. Стоит отметить, что при отключении питания в доме «Умная розетка» не прекращает свою работу, так как внутри присутствует запасной источник питания [3]. Мощности такого питания будет вполне достаточно, чтобы осуществить оповещение владельца о падении напряжения.

«Умные розетки» бывают двух типов:

- Наружные
- Встроенные

Наружные – это розетки, которые не монтируются. Сзади корпуса такого изделия имеются два стержневых контакта, по аналогии с электрической вилкой, которая просто вставляется в штепсельные разъемы обычной встроенной розетки в доме. После этого производится первоначальная установка требуемых параметров работы, условия работы, а также временные параметры.

После выполнения настройки розетка полностью готова к использованию. Лицевая часть такой розетки имеет два штепсельных разъема, которые необходимы для подключения электрического прибора, над которым будет производиться контроль [3].

Встроенные розетки – это устройства, которые устанавливаются в готовые гнезда, как и обычные стационарные розетки. Монтаж таких устройств осуществляется в короб с выполнением кабельного подключения к системе внутреннего электроснабжения квартиры. Лицевая сторона, таких розеток представляет собой два вида штепсельных разъемов под разные вилки, а на внутренней стороне имеются зажимы для крепления «нуля», заземления и «фазы». Недостатком данного типа, является, то что так называемые «мозги» (внутренности) «умных» встраиваемых розеток, как правило, занимают больше места, чем обычные розетки. Это приводит к расширению места их установки.

### **Всевозможные функции «Умных розеток»**

Чем современнее и совершеннее устройство, тем больше функций оно в себя вмещает. Одной из самых популярных функций таких розеток, является дистанционное управление при помощи мобильных приложений. Также «Умные розетки» способны выполнять такие функции, как:

- Автоматическое выключение электрических приборов при падении напряжения, коротком замыкании, перегревании, а также при длительном вашем отсутствии;

- Включение и отключение электрических приборов по таймерам, в соответствии с заранее установленным временем работы;
- Контроль параметров таких, как температура, влажность, напряжение;

- Управление с помощью голоса;

- Экономия электроэнергии;

#### **Недостатки «Умных розеток»:**

- Невозможность использовать несколько устройств одновременно;
- Высокая стоимость в сравнении с обычными, стандартными розетками;
- Присутствует вероятность ложного срабатывания на включение/выключение устройств.
- Без подключения к сети Интернет невозможно управлять розеткой на расстоянии.

В заключении хотелось бы отметить, что «Умная розетка» это очень полезное, многофункциональное устройство в доме и его цена полностью оправдана функционалом. Это уникальное устройство способно включать, выключать, слушать, уведомлять, а также считать затраты и следить за состоянием электроприборов.

Разрабатывая свою «Умную розетку» изначально проанализировав большое количество уже существующих моделей, я пришла к выводу, что на данный момент розетки имеющие максимальное количество функций и возможностей по цене доступны далеко не всем. Это подвигло к тому, чтобы моя разработка была дешевле, чем уже присутствующие на рынке, а также имела функционал не хуже популярных моделей Xiaomi, Redmond и т.д. Добиться всего этого, я смогла, поставив перед собой предел по характеристикам, а также подобрав подходящую и доступную элементную базу и материалы, для изготовления корпуса.

#### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Дементьев А. Умный дом 21 века / ЛитагентРидеро, 2016.
2. Что такое умная розетка? [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/Умная\\_розетка](https://ru.wikipedia.org/wiki/Умная_розетка). (дата обращения: 21.03.2022).
3. Умные розетки: принцип работы, способы управления, виды, рейтинги. [Электронный ресурс] – Режим доступа: <https://future2day.ru/luchsie-umnye-rozetki> (дата обращения: 21.03.2022).

УДК 004.942

**А.А. КАЛИНКИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **ОБЗОР И ИССЛЕДОВАНИЕ МИКРОКОНТРОЛЛЕРОВ ДЛЯ СИСТЕМЫ УПРАВЛЕНИЯ «УМНЫЙ ДОМ»**

*В данной статье будут рассмотрены различные микроконтроллеры для системы управления «Умный дом».*

Микроконтроллер – это микросхема, по своей сути состоящая из центрального процессора и периферии, которая включает в себя таймеры, генераторы, преобразователи и другие подобные элементы. Очень часто микроконтроллер путают с микропроцессором, но у них есть значимые отличия. Микропроцессор выполняет только определенные логические и математические операции, а микроконтроллер имеет в своём составе ОЗУ, ПЗУ и микропроцессор.

В настоящее время существует огромное количество различных по своим свойствам и характеристикам микроконтроллеров. Они являются сердцем управления системой «Умный дом», превращая весь сложный механизм в уникальную сеть. Контроллер для умного дома — это блок управления, который, осуществляет контроль над устройствами, подключенными к нему. Он подаёт ответ на все сигналы, которые поступают с датчиков системы «умный дом», а также составляет отчёты о состоянии подключенных устройств. Создать микроконтроллер можно и своими руками, но, чтобы не усложнять себе жизнь проще купить его в магазине. На данный момент существует огромное количество контроллеров, различающихся по цене и возможностям. Для того, чтобы сделать правильный выбор необходимо оценить точность управления, размер, функционал и автоматизм микроконтроллера. В данной статье будут рассмотрены популярные на сегодняшний день контроллеры, такие как Arduino, Siemens. Овен.

### **I. Arduino**

Микроконтроллеры Arduino являются одними из самых популярных для систем управления «Умный дом», потому что их можно адаптировать под совершенно любой проект. Первое устройство этой марки, отдаленно напоминающее контроллер было создано в 2005 году итальянцем Массимо Банци [1]. Само название пошло в честь короля Ардуина и со временем развиваясь и совершенствуясь стало брендом аппаратно-программных средств для автоматизирования несложных

систем. В настоящее время микроконтроллеры Arduino представляют собой простое AVR устройство с уже готовой прошивкой. Для создания проекта на его основе пользователю будет необходимо только выбрать подходящие библиотеки из имеющихся, либо добавить новые. Сам микроконтроллер состоит из одной схемы, на которой присутствует несколько основных элементов. Стандартная комплектация платы Arduino состоит из чипа с постоянной памятью, набора транзисторов и резисторов, а также пинов для подключения внешних устройств [1]. Управление контроллером осуществляется с помощью специальных программ, которые устанавливаются на телефон, либо компьютер. Благодаря простому и доступному для новичков веб-интерфейсу, настройка устройства не составит большого труда.

Преимуществами Arduino является:

- Невысокая цена и возможность самостоятельной сборки;
- Большое количество дополнительного оснащения: датчики дыма, освещенности, Wi-Fi модули;

- Небольшие размеры;

- Доступность для непрофессионалов, простой процесс отладки;

Недостатками микроконтроллеров Arduino является: невысокая надежность; низкое качество разъемов.

## **II. Siemens**

Микроконтроллеры марки Siemens – это системы немецкого качества, которые применяются не только для автоматизации систем в быту, но и на производстве, в промышленности. Контроллеры этой компании выражены линейкой LOGO, которая участвует в создании Умного дома. Линейка микроконтроллеров LOGO от марки Siemens не сложная в освоении, в принципе, как и все продукты от этой компании [2]. Контроллеры данной линейки состоят из двух модулей. Один – LOGO Basic, включает в себя дисплей и клавиатуру для ввода программ автоматизации вручную. Другой – LOGO Pure разработанный для программирования «Умного дома» непосредственно с компьютера, через кабель интерфейса. Компания Siemens даёт возможность, также для самостоятельной разработки определённых режимов в работе, для чего существует специальная программа Soft Comfort. Память LOGO рассчитана на 200 программных операций. Всего существует 8 различных модификаций, которые между собой различаются напряжением питания, а также количеством каналов ввода/вывода. Кроме этого, серия LOGO включает в себя 9 видов модулей расширения для увеличения числа входов/выходов. Так же в данной серии присутствуют блоки питания для датчиков и модулей расширения. Используя LOGO, как центральный контроллер, с его помощью можно

создавать целые алгоритмы работы схем. Постоянные усовершенствования и новые модификация повышают эксплуатационные качества этого микроконтроллера.

Преимущества микроконтроллера LOGO от компании Siemens:

- Используются для широкого спектра задач автоматизации как жилых помещений, так и промышленности;
- Достаточное количество дополнительных оснащений, что позволяет адаптироваться к любой решаемой задаче;
- В качестве интерфейса, для обмена данными используется протокол Ethernet.
- Обеспечивает разработку программного комплекса для алгоритмов управления.

Недостатки микроконтроллера: высокая стоимость; поддерживаются только имеющиеся протоколы, что исключает подключение какого-либо нестандартного оборудования; большое количество ненужных данных, что значительно сказывается на надежности и снижает помехоустойчивость.

### **III. Овен**

Компания Овен была создана в 1991 году и не останавливается в развитии по сей день [3]. Контроллеры данной компании благодаря большому количеству различных протоколов и дополнительных модулей расширения универсальны во многих сферах применения, таких как водоснабжение, энергетика, автоматизация зданий и жизнеобеспечения, газораспределение, транспорт и др. В любой модификации микроконтроллеров Овен имеется 8 дискретных и аналоговых входов, что считается более чем достаточным для автоматизации вентиляционных установок. Средой программирования для всех микроконтроллеров данной компании является немецкая среда CODESYS 2.3, которая поддерживает поддерживает все 5 языков программирования стандарта МЭК 61131-3 (LD, FBD, IL, ST, SFC) [3]. Разрабатывая контроллеры, Овен отобрала самые популярные и востребованные функции аналогов мировых производителей, поэтому созданные компанией контроллеры ПЛК100/150 обладают всеми необходимыми функциями.

Преимущества контроллеров марки Овен:

- Выполнены в компактных корпусах.
- Имеют достаточное количество дискретных и аналоговых входов и выходов.
- Наличие встроенных часов, что позволяет создавать системы управления с учетом времени.
- Среда разработки для пользователей предоставляется совершенно бесплатно.

Недостатки контроллеров: относятся к средней ценовой категории; все модификации работают только от питания 220В; имеют только один модуль расширения.

Проанализировав все марки контроллеров, я пришла к выводу, что для разработки «Умной розетки», для системы управления «Умный дом» наиболее подходящим является семейство Arduino. С их помощью можно создать совершенно любую систему для «Умного дома». Большое количество дополнительного оснащения, позволит сделать «умную розетку» уникальной, с большим количеством функций и возможностей, а несложный процесс отладки системы позволит адаптировать розетку к любому мобильному приложению и даже позволит управлять ей с помощью чат-бота в Telegram. Контроллеры семейства Siemens подходят больше для промышленной автоматизации, а марки Овен хоть и имеют неограниченное количество возможностей, для использования непрофессионалами будут сложны в использовании.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Соммер, У. Программирование микроконтроллерных плат Arduino/Freeduino / У. Соммер. – СПб.: BHV, 2016. – 256 с.
2. Microcontroller Types and Application. [Электронный ресурс]. – Режим доступа: <http://www.electronicshub.org/microcontrollers>.
3. Петров, И. В. Программируемые контроллеры. Стандартные языки и приемы прикладного проектирования / И.В. Петров. – М.: Солон-Пресс, 2012. – 256 с.
4. Программируемые логические контроллеры: ОВЕН ПЛК. URL: [www.owen.ru](http://www.owen.ru)

УДК 004.9

**А.А. КАЛИНКИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **СРАВНИТЕЛЬНЫЙ ОБЗОР САПР ДЛЯ МОДЕЛИРОВАНИЯ BLUETOOTH-КОЛОНКИ**

*В данной статье будут рассмотрены различные системы автоматизированного проектирования для разработки 3D моделей и создания чертежей.*

История возникновения систем автоматизированного проектирования начинается еще в 60-х годах прошлого века. Первые попытки

разработки алгоритмов для создания таких САПР появились еще в начале 60-х годов, но возможности аппаратных средств на тот момент еще не позволяли получить 3D модель даже простейшего тела. [1]

Как известно, 3D тела получаются путём «выталкивания» двух-мерного эскиза. Таким образом, к концу 60-х годов впервые были разработаны программы, которые позволяли получить трехмерную модель на основе 2D эскиза.

К нашему времени, 3D моделирование и компьютерная графика значительно продвинулись вперед. Сейчас невозможно представить ни одной конструкторской разработки без 3D моделирования.

С появлением современных САПР для 3D моделирования значительно уменьшилось время разработки того или иного устройства. Создание 3D модели на ранних этапах разработки позволяет значительно уменьшить количество брака в дальнейшем производстве. [3]

Перед непосредственным сравнением систем автоматизированного проектирования, хотелось бы отметить, что в практически каждая САПР имеет параметрические возможности 2D и 3D моделирования, а именно – оформление чертежей и КД (конструкторской документации) в соответствии с ЕСКД.

Далее перейдем непосредственно к сравнению и анализу наиболее распространенных САПР в наше время. Сделаем выбор в пользу той САПР, которая обладает рядом преимуществ по сравнению с другими.

На мой взгляд, для разработки устройства для считывания файлов с внешнего носителя и воспроизведения на внешние динамики наиболее подходят следующие САПР: «SOLIDWORKS», «T-FLEX CAD», «КОМПАС-3D».

Для начала рассмотрим САПР «SOLIDWORKS»:

#### **I. «SolidWorks» – «Dassault Systemes»**

Данная система позволяет решать огромный комплекс задач, а именно:

- 1) 3D-проектирование деталей и сборок;
- 2) Создание КД в соответствии с ГОСТ;
- 3) Проведение инженерного анализа (динамика, частотный анализ, прочность, устойчивость и т.д.);
- 4) Проектирование оснастки;
- 5) Проведение анализа технологичности конструкции изделия;
- 6) Проведение анализа технологичности процессов изготовления;
- 7) Разработка программ для станков с ЧПУ. Интерфейс данной САПР достаточно удобен и приятен к вос-

приятно, на освоение программы не требуется много времени. В системе имеются все необходимые инструменты и функции для просмотра информации в области 3D модели, а именно:

- 1) Выбор видов (спереди, сверху, справа) и плоскостей;
- 2) Увеличение и уменьшение модели;
- 3) Создание разрезов и сечений.

Необходимо отметить тот факт, что САПР «SolidWorks» достаточно хорошо оптимизирована. Это связано с тем, что для повышения производительности, и уменьшения времени обработки графических процессов при работе с большими сборками, которые включают в себя тысячи деталей и “подборок”, создан специальный режим, который распределяет ресурсы ПК за счет отображения сокращенной информации о компонентах сборки.

Так же данная система может производить обмен данными с другими САПР за счет большого количества трансляторов различных форматов.

На мой взгляд, одним из недостатков данной САПР являются её системные требования. Дело в том, что далеко не на каждом простейшем ПК будет возможность запустить данную систему. Для установки и запуска с» необходимо достаточно “мощное” и современное аппаратное обеспечение.

Второй недостаток данной САПР – это высокая стоимость лицензии по сравнению с конкурентами.

Таким образом, делаем вывод, что «SolidWorks» возможна, но нежелательна к применению при разработке данного устройства.

## **II. «T-FLEX CAD» – «Топ Системы»**

Данная САПР отечественного производства имеет следующий функционал:

- 1) 3D проектирование деталей и сборок;
- 2) Создание КД в соответствии с ГОСТ;
- 3) Создание параметрического эскиза;
- 4) Параметрическое 2D и 3D проектирование на базе гибридной параметризации;
- 5) Прямой экспорт и импорт файлов в форматах других САПР;
- 6) Ассоциативная связь между 3D моделью и её чертежом;
- 7) Отдельный модуль для создания спецификаций;
- 8) Проведение инженерного анализа различных видов. [3]

В данной системе также имеются все необходимые инструменты и функции для просмотра информации в области 3D модели, а именно:

- 1) Выбор видов (спереди, сверху, справа) и плоскостей;

- 2) Увеличение и уменьшение модели;
- 3) Создание разрезов и сечений.

На мой взгляд, данная САД система является наиболее предпочтительной при разработке устройства для считывания файлов с внешнего носителя и воспроизведения на внешние динамики. Это связано с тем, что по сравнению с «SolidWorks» данная система имеет более приятный интерфейс и ряд преимуществ, которые отсутствуют у «SolidWorks», а именно: автоматическое формирование спецификации (при установке дополнительного модуля) в соответствии с ЕСКД, минимальные системные требования гораздо ниже, большая база 3D моделей стандартных элементов в соответствии с ГОСТ. Установка данной САПР более проста и доступнее по сравнению с конкурентами.

### III. «КОМПАС 3D» – «АСКОН»

Данная САПР по сравнению с предыдущими имеет меньший функционал, но базовый набор функций и команд для 2D и 3D проектирования, естественно, включены в данную систему, а именно:

- 1) 3D проектирование деталей и сборок;
- 2) Создание КД в соответствии с ГОСТ;
- 3) Выбор видов (спереди, сверху, справа) и плоскостей;
- 4) Увеличение и уменьшение модели;
- 5) Создание разрезов и сечений. [2]

Хотелось бы отметить, что в каждой САПР есть стандартная элементная база 3D моделей в соответствии с ГОСТ, а именно:

- 1) Крепежные элементы;
- 2) Соединители и разъемы;
- 3) Профили и швеллера;
- 4) Балки и трубы.

И т.д.

Хотелось бы отметить, что в «КОМПАС 3D» детализация 3D моделей стандартной элементной базы реализована хуже, чем в «T-FLEX» и «SolidWorks», на мой взгляд. Но в то же время, количество 3D моделей соединителей и разъемов в стандартной базе «КОМПАС» больше, чем у конкурентов.

Подводя итог всего вышесказанного, наиболее подходящей САПР для разработки устройства для считывания файлов с внешнего носителя и воспроизведения на внешние динамики является «T-FLEX CAD». Данная система, на мой взгляд, является наиболее сбалансированной и в ней включены все необходимые инструменты и функции для реализации поставленной задачи.

Таким образом, для написания ВКР наиболее подходящей является данная САПР система, именно в ней будет спроектирована 3D модель разрабатываемого устройства и оформлена КД.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. А.Г. Янишевская. История и развитие САПР. Омск: Изд-во ОмГТУ, 2008. – 48 с.
2. Основы проектирования в КОМПАС-3D v17. 2-е изд. / под ред. М. И. Азанова. – М.: ДМК Пресс, 2019. – 232 с.
3. Максимов С.В., Протасова С.В. T-Flex CAD. Начальный курс. Северодвинск 2011 г. – 215 с.

УДК 004.942

**А.А. КАЛИНКИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **МЕТОДЫ УСТРАНЕНИЯ ЯВЛЕНИЯ «ЗЕМЛЯНАЯ ПЕТЛЯ» ПРИ РАЗРАБОТКЕ BLUETOOTH-КОЛОНКИ**

*В данной статье будет рассмотрено явление «земляная петля» и способы борьбы с ним.*

Возникновение явления «земляная петля» достаточно часто встречается еще с 60х годов прошлого века. Например, в 60х – 70х годах при разработке промышленных усилителей так же часто появлялась «земляная петля». Эта проблема зачастую решалась при помощи заземления проводников к лепестку, луженному к корпусу. Данное решение проблемы - рабочее, но не совсем эффективное, на мой взгляд, так как большинство корпусов являлись стальными, а сталь – не самый лучший проводник среди металлов.

Чуть позже появились омедненные шины заземления, которые, на мой взгляд, более подходят для решения данной проблемы.

При разработке устройства для считывания файлов с внешнего носителя и воспроизведения файлов на внешние динамики, я естественно, задался вопросом о возникновении явления «земляная петля». Этот эффект сопровождается характерным писком и треском при включенном устройстве без проигрывания музыки или с проигрыванием, но в основном без. Хотелось бы отметить, что при использовании

различной элементной базы, писк и треск может значительно изменяться – увеличиваться или уменьшаться.

Также не могу не заметить, что наиболее современным и актуальным устройством для считывания файлов с внешнего носителя и воспроизведения файлов на внешние динамики является bluetooth-колонка. Далее речь пойдёт об устранении явления «земляная петля» в bluetooth-колонке.

Таким образом, передо мной появилась задача «обхода» данной проблемы. Для начала, давайте разберемся, что же такое «земляная петля» и как ее «обойти» при разработке устройства для считывания файлов с внешнего носителя и воспроизведения файлов на внешние динамики?

Если давать определение, то «земляная петля» - это явление, возникающее при соединении точек земли дважды и более раз. Говоря простыми словами, «земляная петля» - образование одного замкнутого витка или нескольких.

При разработке устройства для считывания файлов с внешнего носителя и воспроизведения файлов на внешние динамики, может возникнуть данная проблема, так как зачастую у bluetooth-модуля и усилителя питание одно. Именно из-за этого и может образовываться «земляная петля».

Способы устранения явления «земляная петля» бывают следующими:

1. С помощью фильтров, основанных на ферритовом кольце;
2. Подключение отдельного питания на bluetooth-модуль.
3. С помощью преобразователей напряжения.

Давайте рассмотрим каждый из способов по отдельности.

Первый способ:

Ферритовое кольцо – это пассивный электрический компонент, который изготавливается из феррита в форме кольца.[3] В свою очередь, феррит – это соединение на основе оксида железа и оксида других металлов.[1] Ферриты обладают свойствами ферромагнетиков и проводников.[1] Ферритовое кольцо предназначено для подавления ВЧ помех в электрических цепях.

Довольно часто используется данный способ для «обхода» «земляной петли». Но, на мой взгляд, имеет ряд недостатков:

1. Исключает наводки и помехи неполностью;
2. Происходит значительное ухудшение качества звука.

Таким образом, делаем вывод, что данный метод возможен к применению, но не желателен при разработке данного устройства, так как обладает рядом недостатков. При разработке устройства для счи-

тывания файлов с внешнего носителя и воспроизведения файлов на внешние динамики лучше воспользоваться одним из других возможных способов для устранения «земляной петли».

Второй способ:

Одним из самых простых методов «обхода» «земляной петли» - является подключение bluetooth-модуля к отдельному питанию. Таким образом, «земляная петля» разрывается и писк, и шум пропадают. Но данный способ имеет огромный ряд недостатков, а именно:

1. Возможность реализации данного метода только при том условии, что bluetooth-модуль и усилитель реализованы на разных платах, в противном случае – это невозможно;

2. Необходимость в дополнительном (причем более слабом) источнике питания. В следствии этого, возникает еще один ряд недостатков:

2.1. Нежелательное внедрение еще одного аккумулятора в конструкцию (в случае портативной акустики);

2.2. Значительное увеличение габаритов, в следствии добавления еще одного аккумулятора.

Таким образом, делаем вывод, что данный метод хоть и является достаточно простым в реализации, но подходит только для стационарных устройств, которые питаются от 220 В. При разработке моего устройства этот метод не эффективен и бесполезен.

Третий способ:

Данный метод основан на использовании преобразователя с изолированной «землей».

Что же такое преобразователь? Это электротехническое устройство, которое преобразует электрическую энергию с одними значениями параметров в электрическую энергию с другими значениями параметров.[2] Иными словами, преобразователи предназначены для гальванической развязки устройства по питанию.

Хотелось бы отметить, что, если питание устройства осуществляется от аккумулятора или гальванических элементов, то произвести преобразование напряжения до необходимого уровня возможно только с преобразователями типа DC-DC. Зачастую DC-DC преобразователи называются импульсными. Это связано с тем, что постоянное напряжение преобразуется в переменное (повышается или понижается) с частотой в десятки – сотни килогерц.

Один из возможных вариантов преобразователей, который подходит для устранения данного явления является преобразователь – «MORSUN В 1205S». Почему именно он? Данный преобразователь широко известен и часто используется при разработке различных уст-

ройств. Он отличается доступностью к покупке и небольшой стоимостью. Также данный преобразователь имеет небольшие габаритные размеры, поэтому не внесет значительных изменений в конструктив разрабатываемого устройства. «MORSUN B 1205S» имеет следующий ряд характеристик и преимуществ:

1. Продолжительная защита от коротких замыканий с автоматическим восстановлением от нагрузок;
2. Высокая эффективность преобразования энергии (КПД) до 80%;
3. Низкая проходная емкость до 20пФ;
4. Максимальный выходной ток 0.2А;
5. Шум на выходе 60мВ;
6. Вес – 1,3г;
7. Рабочий диапазон температуры от -40 до 105 °С.

Для того, чтобы добиться устранения «земляной петли» при разработке устройства для считывания файлов с внешнего носителя и воспроизведения файлов на внешние динамики, необходимо к двум контактам преобразователя напряжения подключить bluetooth-модуль с усилителем. А один из оставшихся двух контактов преобразователя, допустим, подключить к кнопке вкл/выкл питания, а второй - к аккумулятору.

Таким образом, можно отметить, что явление «земляная петля» достаточно часто встречается при разработке устройств данного типа, но именно использование преобразователей напряжения при решении данной задачи, на мой взгляд, является самым оптимальным методом и эффективным. Причем, при разработке устройства для считывания файлов с внешнего носителя и воспроизведения файлов на внешние динамики необходимо использовать преобразователи именно DC-DC типа, в случае возникновения «земляной петли».

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Я. Смит, Х. Вейн. Ферриты. Архитектура, средства и методы. – М.: Издательство иностранной литературы, 1962 – 504 с.
2. Шустов М.А. Преобразователи напряжения. Книга 3. Альтекс-А, 2002 – 190 с.
3. Л.И. Рабкин, С.А. Соскин, Б.Ш. Эпштейн. Ферриты. Ленинград: Энергия, 1968 – 384 с.

УДК 004.021

**А.Н. КАЩЕЕВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**МЕТОДЫ ВЫЧИСЛЕНИЯ  
ТРИГОНОМЕТРИЧЕСКИХ ФУНКЦИЙ**

*В данной статье будут рассмотрены два метода вычисления основных тригонометрических функций: геометрический и аналитический.*

К прямым (основным) тригонометрическим функциям относят: функцию  $\sin(x)$  (синус)  $\cos(x)$  (косинус). Другие функции:  $\operatorname{tg}(x)$  (тангенс);  $\operatorname{ctg}(x)$  (котангенс);  $\operatorname{sec}(x)$  (секанс);  $\operatorname{cosec}(x)$  (косеканс) – являются производными от основных и определяются по следующим формулам:

$$\operatorname{tg}(x) = \frac{\sin(x)}{\cos(x)}; \quad \operatorname{ctg}(x) = \frac{\cos(x)}{\sin(x)}; \quad \operatorname{sec}(x) = \frac{1}{\cos(x)}; \quad \operatorname{cosec}(x) = \frac{1}{\sin(x)}.$$

Также у всех представленных функций есть обратные:  $\operatorname{arcsin}(x)$ ;  $\operatorname{arccos}(x)$ ;  $\operatorname{arctg}(x)$  и т.д. Однако в данной статье речь будет только об основных функциях.

Синусом угла  $\alpha$  (альфа) является отношение противолежащего катета « $a$ » к гипотенузе « $c$ » прямоугольного треугольника ABC (рисунок 1). Формула следующая:

$$\sin(\alpha) = \frac{a}{c}. \quad (1)$$

Косинусом угла  $\alpha$  является отношение прилежащего катета « $b$ » к гипотенузе « $c$ » прямоугольного треугольника ABC (рисунок 1). Формула следующая:

$$\cos(\alpha) = \frac{b}{c}. \quad (2)$$

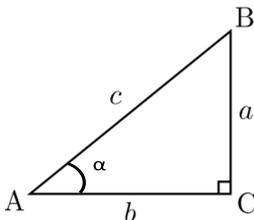


Рисунок 1 – Прямоугольный треугольник

Основная задумка этих функций заключается в сопоставление угла к отношению сторон прямоугольного треугольника.

Теперь возникает вопрос: как, собственно, вычислять эти функции при произвольном аргументе  $\alpha$ ? В настоящей статье рассмотрим два метода:

**Первый метод (геометрический)** дает приближенные значения из-за не идеальности схемы и субъективных восприятий наблюдателя. Для того чтобы понять принцип рассмотрим схему:

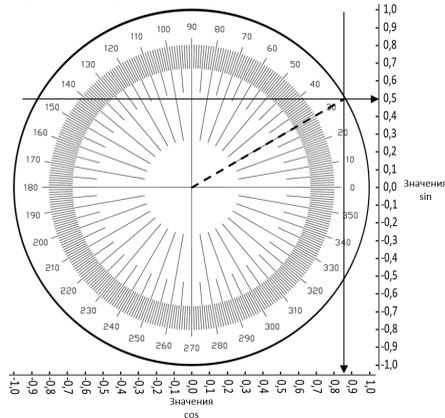


Рисунок 2 – Окружность для вычисления синуса и косинуса

На рисунке 2 приведена окружность и имеет 3 шкалы: в центре шкала градусов  $[0;360^\circ]$ ; в низу горизонтальная шкала значений косинуса  $[-1;1]$ ; вертикальная шкала значений синуса  $[-1;1]$ . Допустим, надо определить значения синуса и косинуса угла в 30 градусов. Для этого проведем прямую из центра окружности так, чтобы эта прямая проходила через риску в 30 градусов (на рис. 2 это показано штрих-линией). В результате мы получаем точку на окружности. Далее проецируем ее на вертикальную и горизонтальную шкалу. Места пересечения осей шкал и дадут значения синуса и косинуса заданного угла. На схеме:  $\sin(30^\circ)=0,50$ ;  $\cos(30^\circ)\approx 0,86$ .

Замети, что таким методом можно определять и обратные функции:  $\arcsin(x)$  и  $\arccos(x)$ .

**Второй метод (аналитический)** основан на разложении функции в ряд Тейлора, формула представлена ниже:

$$f(x) = \sum_{k=0}^N \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k + \Theta(x, x_0); \quad (3)$$

где  $f(x)$  – функция;  $\Theta(x, x_0)$  – остаточный член в форме Пиано;  $x$  – переменная;  $x_0$  – точка начала разложения (смещение). Если  $N \rightarrow \infty$  тогда  $\Theta(x, x_0) \rightarrow 0$ . Фактически начало разложения функции всегда происходит в

точке  $x=0$ , благодаря  $x_0$  можно «сместить» точку начала разложения в любую удобную.

Разложим в ряд Тейлора функцию синуса в точке  $x_0 = 0$ :

$$f(x) = \sin(x) = \sin(0) + \frac{\cos(0)}{1!}x^1 - \frac{\sin(0)}{2!}x^2 - \frac{\cos(0)}{3!}x^3 + \frac{\sin(0)}{4!}x^4 \dots \quad (4)$$

Запишем многочлен (4) в виде бесконечной суммы [1]:

$$\sin(x) = f(x, N) = \sum_{k=0}^{N \rightarrow \infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!} \quad (5)$$

Далее для формулы (5) найдем область сходимости по признаку Даламбера:

$$\lim_{k \rightarrow \infty} \left| \frac{f_{k+1}(x)}{f_k(x)} \right| = l(x) \quad (6)$$

$$l(x) \begin{cases} l(x) > 1 - \text{ряд\_расходится;} \\ l(x) < 1 - \text{ряд\_сходится;} \\ l(x) = 1 - \text{неопределенность.} \end{cases}$$

Для функции синуса:

$$\lim_{k \rightarrow \infty} \left| \frac{(-1)^{k+1} \frac{x^{2(k+1)+1}}{(2(k+1)+1)!}}{(-1)^k \frac{x^{2k+1}}{(2k+1)!}} \right| = \lim_{k \rightarrow \infty} \left| \frac{-x^2}{(2k+3)(2k+2)} \right| = x^2 \cdot 0 < 1.$$

Получается, формула (5) справедлива при любых  $x$ :  $(-\infty < x < +\infty)$ .

Аналогичным образом получим косинус:

$$\cos(x) = g(x, N) = \sum_{k=0}^{N \rightarrow \infty} (-1)^k \frac{x^{2k}}{(2k)!} \quad (7)$$

Область сходимости  $x$ :  $(-\infty < x < +\infty)$ .

В действительности для вычислений используют конечные суммы (5) и (7), однако здесь возникает следующая задача – какое следует взять целое положительное число  $N$ , чтобы обеспечить заданную точность?

Так как мы получили разложение в ряд Тейлора относительно  $x_0=0$  (см. рис. 3), то на границах интервала определения  $[-\pi < x < \pi]$  ошибка будет максимальна. Чтобы вычислить значения ошибок в точке  $\pi$  (или  $-\pi$ ) используем следующие формулы:

$$\varepsilon_{\sin} = |\sin(\pi) - f(\pi, N)| = |f(\pi, N)| \quad (8)$$

$$\varepsilon_{\cos} = |\cos(\pi) - g(\pi, N)| = |-1 - g(\pi, N)| = |1 + g(\pi, N)| \quad (9)$$

Благодаря этим формулам можно создать таблицу ошибок (см. таблица 1).

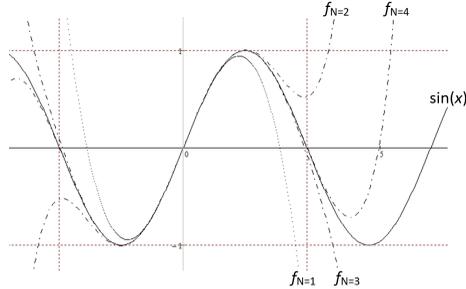


Рисунок 3 – Функция синуса и его аппроксимация полиномами

Таблица 1 – макс. ошибки

| sin |                            | cos |                            |
|-----|----------------------------|-----|----------------------------|
| $N$ | Макс. ошибка $\varepsilon$ | $N$ | Макс. ошибка $\varepsilon$ |
| 1   | 2,02612                    | 1   | 2,9348                     |
| 2   | 0,52404                    | 2   | 1,12391                    |
| 3   | 0,07522                    | 3   | 0,21135                    |
| 4   | $6,92527 \cdot 10^{-3}$    | 4   | 0,02398                    |
| 5   | $4,45160 \cdot 10^{-4}$    | 5   | $1,8291 \cdot 10^{-3}$     |
| 6   | $2,11426 \cdot 10^{-5}$    | 6   | $1,00470 \cdot 10^{-4}$    |
| 7   | $7,72786 \cdot 10^{-7}$    | 7   | $4,16781 \cdot 10^{-5}$    |
| 8   | $2,24195 \cdot 10^{-8}$    | 8   | $1,35260 \cdot 10^{-7}$    |
| 9   | $5,28918 \cdot 10^{-10}$   | 9   | $3,52908 \cdot 10^{-9}$    |
| 10  | $1,03481 \cdot 10^{-11}$   | 10  | $7,56507 \cdot 10^{-11}$   |
| 11  | $1,70395 \cdot 10^{-13}$   | 11  | $1,35647 \cdot 10^{-12}$   |
| 12  | $2,62671 \cdot 10^{-15}$   | 12  | $2,04281 \cdot 10^{-14}$   |

Благодаря этой таблице можно подобрать  $N$  для требуемой точности расчетов (для получения «надежного» результата рекомендуется брать с запасом  $N+1$ ).

Описанный выше аналитический метод хорошо реализуется на языках высокого уровня. Алгоритм вычисления функций синуса и косинуса на языке C++ представлена ниже:

```
#define _USE_MATH_DEFINES
#include <cmath>
#include <iostream>
#include <iomanip>
#include <math.h>
using namespace std;
const int N=9;

double bin(double number,int degree){ //функция возведения в целую степень
    double temp=1;
    for (int i=0; i<degree; i++) temp=temp*number;
    return temp;}
double factor (double x){ //функция факториала
    double temp=x;
    while (x>1){ temp=temp*(x-1); x--;}
}
```

```

    if (x==0) return 1; else return temp;}

double sinus (double x){ //функция синуса
    double temp=0;
    for(int i=0; i<=N; i++){temp=temp+(bin(-1,i)*bin(x,2*i+1))/factor(2*i+1);}
    return temp;}

double cosinus (double x){ //функция косинуса
    double temp=0;
    for(int i=0; i<=N; i++) {temp=temp+(bin(-1,i)*bin(x,2*i))/factor(2*i);}
    return temp;}

int main(){
    cout<<setprecision(20); //количество выводимых цифр
    setlocale(0, ""); //для вывода русских символов
    int k=1; //множитель знака числа
    bool flag=false;
    double x, gradus;
    cout <<"0 - rad;""<<endl<< //0 - для вычисления значений в радианах
        "1 - grad.""<<endl<<"flag = "; //1 - для вычисления значений в градусах
    cin>>flag;
    if (flag==0) cout <<"(rad) X= "; else cout <<"grad X= ";
    cin>>x;
    gradus=x; //хранит копию аргумента
    if (flag==1) x=(x*M_PI)/180; //перевод в радианы, если аргумент в градусах
    //приведение к интервалу -π<x<π
    while (x<-M_PI){ x=x+M_PI; k=k*(-1);}
    while (x>M_PI){ x=x-M_PI; k=k*(-1);}
    cout <<"Алгоритм:      sinus(" <<gradus<<" )=" <<k*sinus(x)<<endl;
    cout <<"Библиотечная функция: sin(" <<gradus<<" )=" <<sin(x)<<endl;
    cout <<"Алгоритм:      cosinus(" <<gradus<<" )=" <<k*cosinus(x)<<endl;
    cout <<"Библиотечная функция: cos(" <<gradus<<" )=" <<cos(x)<<endl;
}

```

Результат работы алгоритма приведен ниже:

```

0 - rad;
1 - grad.
flag = 1
grad X= 160
Алгоритм:      sinus(160) =0.34202014328090307727
Библиотечная функция: sin(160) =0.34202014332566887944
Алгоритм:      cosinus(160) =-0.93969262112205842463
Библиотечная функция: cos(160) =-0.93969262078590831688

```

Рисунок 4 – Результат выполнения программы

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Демидович Б.П., Марон И.А. Основы вычислительной математики. М.: Гос. изд-во физ.-мат. литературы, 1960. 659 с.

УДК 620.171

**Е.В. КАПУСТИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **МЕТОДЫ КОНТРОЛЯ ДЕФЕКТОВ В ЭЛЕКТРОННЫХ СРЕДСТВАХ**

*В статье приводится описание методов разрушающего и неразрушающего контроля изделий электроники.*

Контроль качества играет важную роль при обеспечении надежности. При выходе на рынок предприятиям необходим высокий уровень контроля и повторяемости производства, возможность гибкой перестройки производства с одного продукта на другой при максимально возможном уровне качества выпускаемой продукции.

Обеспечение качества и надежности является важной проблемой при сборке или монтаже электронных изделий. Поэтому большую роль играет оценка качества выпускаемой продукции, которая производится с помощью различных видов контроля.

Для качественного обнаружения неисправностей и повреждений используют различные методы разрушающего и неразрушающего контроля. Технология разрушающего контроля – это совокупность воздействий, приложенных к образцу для измерения и контроля показателей качества объекта, после которых нарушается пригодность изделия. В свою очередь неразрушающий контроль не нарушает целостности изделия.

Разрушающие методы применяют для контроля качества продукции путем выборочных испытаний конкретных образцов. По его результатам делают вывод о всей партии, к которой относились испытываемые изделия. Принципиальный недостаток методов разрушающего контроля заключается в его выборочности, то есть разрушаются одни изделия, а вводятся в эксплуатацию другие. Также для достоверности такого контроля необходимо сходство условий эксплуатации и испытаний. Наиболее эффективный метод разрушающего контроля выбирают исходя из поставленных задач и свойств исследуемых материалов

Наиболее часто в качестве механического разрушающего контроля используют тесты «на отрыв» и «на сдвиг». Их суть заключается в анализе значений критических нагрузок, после чего составляется

статистическая картина, на основе которой технолог корректирует режимы производства.

В настоящее время существуют множество методов неразрушающего контроля, позволяющих надежно определить опасные дефекты. С помощью неразрушающего контроля можно определить качество паяных соединений, выявить трещины и расслоения, а также определить параметры объекта контроля

Рассмотрим визуальный контроль качества пайки. Обычно он осуществляется с помощью бинокулярных микроскопов. Соединения, имеющую гладкую блестящую поверхность без следов перегрева и трещин обеспечивают качественные паяные соединения. Также с помощью визуального контроля можно оценить наличие трещин, спаев, избыток или недостаток припоя. Недостатками такого контроля считают: невозможность определения внутренней структуры объекта и скрытых дефектов, а также зависимость от субъективных обстоятельств.

Для обнаружения скрытых дефектов могут использоваться следующие методы: рентгенографический контроль и ультразвуковая микроскопия. Обычно их применяют на заключительных этапах производства, если разрушающий контроль экономически не оправдан.

Для выявления внутреннего строения образца используют рентгеноскопию. Рентгеновский контроль позволяет выявить трещины в паяных соединениях. Плотность среды зависит от количества поглощенных рентгеновских лучей. Проходящие через материал лучи ослабляются в различной степени позволяют определить наличие и расположение дефектов или неоднородностей. Основными преимуществами данного метода являются невысокая стоимость и простота.

Акустическая микроскопия является универсальным методом неразрушающего контроля, который используется для обнаружения и анализа скрытых дефектов. Этот метод позволяет получать изображения объектов, используя свойства звуковых волн. Суть заключается в том, что у любого материала есть собственное акустическое сопротивление – величина, показывающая сопротивление материала, при смещении частиц под воздействием акустических волн. Так, на границе между материалами ультразвуковой импульс разделяется на две составляющих, и чем больше разность сопротивлений на границе между материалами, тем контрастнее получается изображение. Таким образом, акустическая микроскопия позволяет исследовать срезы образца, не нарушая его целостности.

Ультразвуковая дефектоскопия выявляет дефекты объекта при помощи ультразвукового контроля. Принцип дефектоскопов основан

на эхо-импульсном методе: излучатель посылает в изделие короткие ультразвуковые импульсы и при встрече с преградой часть энергии возвращается. Отраженные колебания преобразуются приемником и поступают на экран дефектоскопа. Регистрируется интенсивность таких сигналов, тем самым можно определить размеры и расположение дефекта. Выделим положительные черты ультразвуковой дефектоскопии: данный метод позволяет проводить контроль с достаточно высокой скоростью и имеет при этом низкую стоимость.

Ультразвуковая инспекция основана на исследовании процесса распространения ультразвуковых колебаний в контролируемом объекте. Такой метод обладает разнообразием типов сканирования. Часть из них направлена на обнаружение мелких дефектов, что невозможно при других видах неразрушающего контроля, а другая часть позволяет получать достоверную информацию о любой точке или плоскости объекта, формировать трехмерные и послойные модели изделий, регистрируя при этом такие дефекты, как трещины, пустоты и отслоения в подложках и кристаллах.

Принцип действия ультразвукового микроскопа заключается в том, что механические свойства материалов определяются величиной акустического сопротивления. На границе между материалами ультразвуковой импульс разделяется на две составляющих: пропущенную и отраженную, и чем больше разница сопротивлений на границе раздела, тем амплитуда отклика больше и контраст изображения выше. Также, в большинстве случаев, показателем наличия пустот, расслоения и трещин будут являться отклики, обладающие отрицательной полярностью.

Фотоакустическая микроскопия применяется в электронике достаточно часто и является универсальным методом неразрушающего контроля. Данная технология использует термоупругий эффект: при освещении контролируемого объекта импульсным лазером генерируются акустические колебания. Энергия светового потока поглощается материалом, тем самым порождая световую волну, параметры которой зависят от характеристик материала. В результате чувствительный элемент, например, пьезоэлектрический детектор, регистрирует появившиеся термоупругие колебания. Данная технология позволяет решить множество задач контроля. Преимуществами фотоакустической микроскопии являются: однозначность получаемой информации; универсальность; возможность контроля на ранних этапах производства.

В акустической и сканирующей лазерной микроскопии первичную информацию получают в результате генерации, приема, а также взаимодействия чисто акустических колебаний, а в лазерной фотоаку-

стической микроскопии импульсное лазерное излучение взаимодействует с поверхностью твердого тела. Амплитуду генерируемых акустических колебаний можно определить, во-первых, по коэффициенту оптического поглощения лазерного излучения в каждой точке поверхности сканируемого объекта и, во-вторых, по мощному фоновому сигналу, возникающему по причине удаления различных загрязнений и окислов с поверхности объекта при первичном сканировании лазерным лучом. Однако на результирующей топограмме получаются изображения, расшифровать которые достаточно сложно.

Метод электронной микроскопии также широко применяется на предприятиях и является одним из наиболее эффективных методов контроля качества. В электронных микроскопах, в отличие от обычных, вместо световых лучей используется поток электронов вакуумной среде, а вместо оптических линз – электронное поле. Под воздействием электромагнитного поля электроны проникают на поверхность вытесняя фотоны, которые и формируют изображение на электронно-лучевой трубке. Полученное изображение можно рассмотреть на флюоресцирующем экране.

Для исследования методом электронной микроскопии к объектам предъявляются следующие требования: малая толщина и равномерность слоя. Основным преимуществом данного метода является высокая разрешающая способность.

Таким образом, технологии разрушающего и неразрушающего контроля позволяют повысить себестоимость и качество выпускаемой продукции. С развитием нанoeлектроники возрастает значимость неразрушающих методов контроля.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Бережной В. П., Дубицкий Л. Г. Выявление причин отказов РЭА. М.: Радио и связь. 1983.
2. Ланин В. Л., Достанко А. П., Телеш Е. В. Формирование токопроводящих контактных соединений в изделиях электроники. // Минск: Издат. Центр БГУ. 2007.
3. Гафт С., Матов Е. Выбор стратегии контроля в условиях перехода к бессвинцовым технологиям. // Технологии в электронной промышленности. 2005.
4. Васильев А. Оборудование для разрушающего и неразрушающего контроля в производстве микроэлектроники. // Поверхностный монтаж. 2007.
5. Шмаков М. Выбор системы рентгеновского контроля. Взгляд технолога. // Технологии в электронной промышленности. 2006.

УДК 004.42

**П.С. КИРЕЕВ, С.В. СКВОРЦОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **АРХИТЕКТУРА АРХЕОЛОГИЧЕСКОЙ АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

*Рассматриваются особенности архитектуры археологических автоматизированных информационных систем, а также эффективность их использования и рациональность разработки.*

В настоящее время активно набирает обороты процесс внедрения автоматизированных информационных систем в различные области науки, одной из которых является археология. В ходе своей работы археолог постоянно сталкивается с необходимостью обработки и анализа огромных массивов однотипных статистических данных. Внедрение археологических автоматизированных информационных систем (АИС) позволит значительно улучшить этот процесс.

В России в 1991 г. на базе СУБД Oracle была разработана комплексная автоматизированная музейная информационная система (КАМИС), внедренная на сегодняшний день в работу более 700 музеев страны для решения широкого круга задач хранения и обработки данных, в том числе и археологических [1]. Однако, в связи с нацеленностью на универсальность хранимых данных, архитектура КАМИС не может обеспечить необходимый уровень их атомарности для осуществления выборок по специфическим параметрам, относящимся к конкретным категориям археологических артефактов. В связи с этим возникает сложность структурирования данных для дальнейшего статистического анализа с целью интерпретации информации, полученной из вещественного исторического источника.

Таким образом, использование музейных АИС для хранения информации о вещественных источниках с целью её дальнейшей обработки методами математической статистики невозможно. Поэтому встает задача разработки отдельной информационной системы, либо модернизация уже существующей путём интеграции в неё отдельных модулей для работы с археологическими данными. Подобная археологическая АИС для сбора и анализа данных разрабатывается в настоящее время авторами.

Процесс проектирования любой информационной системы начинается с её архитектуры, которая, в свою очередь, состоит из двух

взаимосвязанных частей: базы данных (БД) и клиентского приложения.

Для разработки базы данных необходимо представить предметную область как систему дискретных параметров, состоящую из подсистем (таблиц) с набором свойств (атрибутов). В данном случае в качестве предметной области выступает вещественный источник.

Обобщённая модель описания вещественных источников для хранения информации о них в базе данных была разработана коллективом авторов во главе с Ю.Л. Щаповой [2]. Она складывается из двух компонентов: фиксации и описания. Фиксация состоит из наименования и рисунка. Описание включает внешние и внутренние признаки.

К внешним признакам относятся место находки, место хранения, сведения о публикации, данные о территориальной, хронологической и культурной принадлежности.

Внутренние признаки подразделяются на морфологию, материал, технологию изготовления и функции предмета. В морфологии выделяется набор конструктивных элементов и их форма, метрика, цвет, обработка поверхности и декор. Набор конструктивных элементов зависит от категории предмета.

Морфологические признаки, описывающие конструктивный элемент, могут подразделяются на топологические (внешние контуры и строение) и топографические (место расположения элемента и соединение элементов между собой). Для описания формы любого конструктивного элемента достаточно трёх основных признаков: форма, строение и вид. Среди топографических признаков необходимо учитывать место, на котором расположен элемент, плоскость, в которой соединены элементы, угол и линию соединения. Место расположения (локализация) имеет значение для дополнительных признаков, а не для основных. Метрика подразделяется на размеры предмета в целом, размеры отдельных конструктивных элементов и пропорции предмета. При описании цвета необходимо указывать цвет самого предмета, цвет всех его составных элементов. Декор подразделяется на знак и композицию знаков.

Описание материала раскладывается на материал самого предмета, материал поверхности покрытия, материал декора. Технология изготовления подразделяется на технологию получения материала, технологию изготовления предмета и инструментарий. Изготовление предмета в свою очередь подразделяется на изготовление основы, соединение конструктивных элементов, обработку поверхности, нанесение декора [3].

Описание большинства вышеназванных подсистем не зависит от категории вещественного источника. Единственная подсистема, зависящая от него – это набор и форма конструктивных элементов [2]. В качестве вещественного источника, данные о котором будут храниться в БД разрабатываемой АИС, были взяты русские средневековые керамические сосуды с клеймами на донцах. Для описания данной подкатегории необходимо учитывать такие специфические конструктивные элементы как тип глины, примесей, подсыпки и обжига, а также сохранность рисунка клейма.

В качестве архитектуры клиентской части информационной системы была выбрана MVC-модель (Model-View-Controller) или трёхуровневая архитектура – схема разделения системы на три отдельных уровня (модель, представление и контроллер), таким образом, что изменение каждого компонента может осуществляться независимо. Модель содержит данные и методы работы с ними, раскрывает функционал и инкапсулирует различные состояния приложения. Представление, или пользовательский интерфейс, визуализирует модель для пользователя, осуществляет отправку введённых им данных в контроллер, но при этом никак не производит их обработку. Контроллер реализует функционал приложения, то есть определяет, каким будет его ответ на взаимодействие пользователя с системой.

Разработка трёхуровневой архитектуры намного сложнее и требует заметно больше времени по сравнению с однозвенной архитектурой, имеющей аналогичный функционал. Однако преимущества, которые даёт использование MVC-модели для выбранной предметной области, переключают эти недостатки.

Археологическая АИС должна быть адаптирована под специфические условия работы археолога, когда доступ к сети Интернет в некоторых ситуациях может отсутствовать длительное время. Поэтому необходимо предусмотреть два режима работы системы: локальный – однопользовательский, и серверный – многопользовательский, с постоянным обменом данными с сервером через сеть [1]. Таким образом, слой модели будет взаимодействовать с двумя источниками данных.

Одним из хорошо стандартизированных и широко применяемых локальных источников данных является XML-файл. XML – технология, разработанная для управления структурированными данными и отображения этих данных в виде текстового файла [4]. Язык XML соответствует всем отраслевым стандартам и поддерживается многими базами данных и приложениями, что открывает возможность организации взаимодействия локального источника данных и базы данных АИС.

Для реализации серверной базы данных для АИС с MVC-архитектурой был выбран фреймворк среды разработки информационных систем Visual Studio – Entity Framework – технология объектно-реляционного сопоставления, которая позволяет разработчикам .NET работать с реляционными данными с помощью объектов, относящихся к домену [5]. Инструмент, входящий в состав Entity Framework, под названием EDM (Entity Designer Model) позволяет создавать базы данных на основе существующей концептуальной модели и наоборот. Кроме того, применение MVC-архитектуры совместно с Entity Framework обеспечивает возможность обмена данными между пользователем, клиентским приложением и двумя версиями БД: локальной и серверной, через единую модель данных.

Подводя итог, можно сделать вывод, что использование универсальных музейных АИС для хранения информации о вещественных источниках с целью её дальнейшего статистического анализа нецелесообразно. Поэтому для этой цели необходима разработка полноценных новых АИС, либо модулей к существующим музейным системам. Оптимальным выбором архитектуры для таких систем является трёхуровневая модель MVC, позволяющая организовать эффективное взаимодействие с пользователем и между собой локальной и серверной баз данных.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Киреев П.С., Скворцов С.В. Применение автоматизированных информационных систем в археологии // Новые информационные технологии в научных исследованиях (НИТ-2021): материалы XXVI Всероссийской научно-технической конференции. Рязань: ИП Коняхин А.В. (BookJet). 2021. С. 154 – 156.

2. Щапова Ю.Л., Лихтер Ю.А., Сарачев Т.Г., Столярова Е.К. Основные принципы изучения древних материалов и технологий // Вестник МГУ. Сер. 8. История. 2000. №4. С. 102 – 109.

3. Лихтер Ю.А. Вещественный источник и база данных – принципы взаимодействия. М.: ООО «ТМ Продакшн». 2015. 414 с.

4. Обзор XML в Excel [Электронный ресурс]. – Режим доступа: <https://support.microsoft.com/ru-ru/office/%D0%BE%D0%B1%D0%B7%D0%BE%D1%80-xml-%D0%B2-excel-f11faa7e-63ae-4166-b3ac-c9e9752a7d80>, свободный.

5. Entity Framework Tools в Visual Studio [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/ru-ru/visualstudio/data-tools/entity-data-model-tools-in-visual-studio?view=vs-2019>, свободный (03.11.2021).

УДК 002.1

**В.Ю. КОЖИКОВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**УСТРОЙСТВО ГОРИЗОНТИРОВАНИЯ***Рассматриваются вопросы разработки устройства горизонтирования.*

Устройство предназначено для работы в аппаратах по приспособлению опорных платформ к оборудованию с переносной гидравлической опорой. Данный инструмент направлен на увеличение скорости автоматического выравнивания опорной платформы в горизонтальной плоскости, автоматическое удержание ее в горизонтальном положении на заданной высоте, предотвращение потери устойчивости платформы в аварийных ситуациях.

Само по себе горизонтирование – это процесс совмещения отвесной линии и вертикали прибора, с помощью которого производится измерение.

По окончанию процесса создания модели устройства для выравнивания платформы разработчик должен получить соответственно следующий результат: платформа должна удерживать углы осей  $\alpha_x = 0$  и  $\alpha_y = 0$ ; не должно происходить отрыва опор от опорной поверхности; платформа должна автоматически и непрерывно поддерживать заданное оператором положение.

При процессе разработки устройства горизонтирования должно учитываться, что данное устройство будет эксплуатироваться в жестких и тяжелых условиях эксплуатации. Устройство и его блоки должны быть способны противостоять окружающей среде в течение всего заявленного срока службы. К устройству также предъявляются следующие технические требования: во-первых, это сохранение работоспособности в течение всего срока службы; во-вторых, модель должна иметь высокий уровень ремонтпригодности; в-третьих, устройство должно иметь высокий уровень технологичности.

Необходимое оборудование для разработки и сборки приспособления для горизонтирования: четыре единицы опорных стоек, система подачи жидкости к этим опорам, два датчика наклона, блок управления, а также блок остановки устройства в аварийной ситуации.

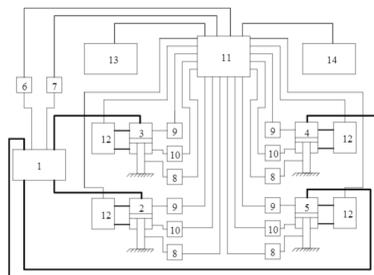


Рисунок 1 – Функциональная схема

По углам данного приспособления закреплены гидроцилиндры опор, а на самом основании имеются два регулятора давления: один в плоскости поршня, а второй в плоскости штока. На платформе датчики наклона находятся таким образом, чтобы измерительные оси датчиков были перпендикулярны диагональным вертикальным плоскостям платформы, но при этом необходимо выполнение двух условий: во-первых, измерительная ось датчика должна быть перпендикулярна диагонали платформы, во-вторых, измерительная ось датчика должна быть перпендикулярна к вертикали. Датчики наклона необходимы для измерения отклонения опорной платформы от горизонтали.

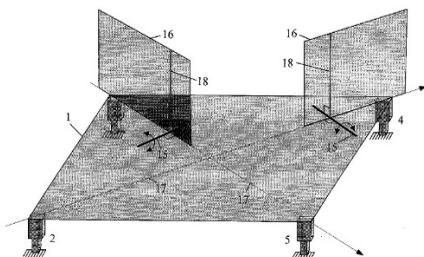


Рисунок 2 – Устройство выравнивания опорной платформы

Для измерения углов платформы применяются датчики, производимые в серийном производстве, что дает лёгкую доступность компонентов. Также используются серийно производимые гидроцилиндры опор. Чтобы предотвратить несчастные случаи при выходе из строя гидравлического привода, используются встроенные гидравлические замки.

Принцип работы устройства выглядит следующим образом: сначала аппаратура формирует аналоговый сигнал, который пропорционален углу наклона пола основания (платформы) машины относительно

но горизонта – это напряжение между цепями «ВЫХ 1» и «ВЫХ 2 ( $U_{оп}$ )»; затем формируются дискретные сигналы:

– «ВЫХОД 1» (цепи « $+L_{ВХ}$ », « $+L_{ВЫХ}$ ») – «сухой» контакт реле.

Контакт реле замыкается, когда происходит достижение угла наклона оси изделия порогового значения  $L > +L_{пор}$ ; размыкается – при  $L < (+L_{пор} - 0,5)$  градуса при положительных углах;

– «ВЫХОД 2» (цепи « $-L_{ВХ}$ », « $-L_{ВЫХ}$ ») – «сухой» контакт реле.

Контакт реле замыкается, когда происходит достижение угла наклона оси изделия порогового значения  $L > -L_{пор}$ , а размыкается при  $L < (-L_{пор} - 0,5)$  градуса при отрицательных углах. Пороговый угол  $L_{пор}$  регулируется в пределах от 1,0 до 5,0 градусов.

В зависимости от положения опор изменяются углы наклона платформы в горизонтальной плоскости.

В устройстве выравнивания также присутствует GPS- устройство. Принцип его работы следующий: если подать напряжение в одну из рабочих секций распределителя, то тогда поток перенаправляется в одну из рабочих плоскостей гидроцилиндра. Выдвижение или втягивание штока зависит от полярности сигнала, который был подан на секцию электрогидравлического распределителя. Встроенные гидравлические замки гидроцилиндров служат для предотвращения аварийной ситуации в случае утечек в гидравлических линиях. Предохранительный клапан используется для предотвращения аварийной ситуации, связанной с избыточным давлением в гидравлической системе в случае неисправностей в гидрораспределительном узле.

У данного устройства имеется полезная возможность одновременно выдвигать одну опору и при этом втягивать другую. Это достигается с помощью новых датчиков угла наклона, позволяющих независимо устранять угол наклона платформы в диагональных и вертикальных плоскостях.

Для управления устройством горизонтирования может использоваться стационарная панель оператора или пульт дистанционного управления. Пульт должен отвечать следующим характеристикам: климатическое исполнение В1, степень защиты IP65.

Достоинства устройства для выравнивания опорных платформ.

Во-первых, уменьшается время выравнивания платформы в горизонтальной плоскости, но при этом не ухудшается точность выравнивания.

Во-вторых, во время работы машины появляется возможность поддержания платформы в горизонтальном положении. Отсутствует изменение параметров угловой ориентации.

В-третьих, непрерывное автоматическое поддержание опорной платформы на заданной постоянной или переменной высоте. Предотвращение ситуации вытягивания стержней на максимальную длину, а также касания опорной поверхности колесами машины. За счет этого увеличивается запас управляемости по углам наклона платформы.

В-четвертых, предотвращение аварийных ситуаций, связанных с потерей устойчивости путем выдвижения опоры (опор) или аварийной остановки машины при сильном проседании грунта под какой-либо из опор во время эксплуатации.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Корятов М.С. «Автоматизация проектирования устройств управления положением платформы строительной машины». Омск: СибАДИ, 2011.

2. Система автоматизированного горизонтирования платформы радиолокационной станции [Электронный ресурс] URL: [https://rep.bntu.by/bitstream/handle/data/37091/Sistema\\_avtomatizirovannogo\\_gorizontirovaniya\\_platformy\\_radiolokacionnoj\\_stancii.pdf](https://rep.bntu.by/bitstream/handle/data/37091/Sistema_avtomatizirovannogo_gorizontirovaniya_platformy_radiolokacionnoj_stancii.pdf).

3. Святун Р.А. «Автоматическая система горизонтирования для размещения высокоточного оборудования». Донецк, 2014.

УДК 004.832.28

### В.П. КОРЯЧКО, А.С. ЕПИФАНОВ

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### ПРИНЯТИЕ РЕШЕНИЙ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ

*Рассматриваются подходы к принятию решений в интеллектуальных системах, затрагивающих интересы нескольких равноправных участников.*

При проектировании интеллектуальных систем, перед которыми потенциально может встать возможность какого-либо выбора, следует учитывать модели принятия решений, которыми будет руководствоваться разрабатываемая система. Правильное принятие решений способно увеличить полезность и эффективность системы.

Вопрос о правильности принятия решений не является новым. Он составная часть философского учения - утилитаризма, суть которого состоит в оценке результатов индивидуальных или коллективных действий (при этом средства их достижения не оцениваются) [1,2].

Зачастую процесс принятия решений у разрабатываемой интеллектуальной системы затрагивает интересы нескольких равноправных участников, поэтому следует рассматривать механизмы коллективного (кооперативного) принятия решений. В таком случае сама оценка будет строиться только на основе уровней полезности, которые достаются индивидуальным агентам (участникам).

Согласно теории утилитаризма, для каждого агента  $i$  определяется свой уровень полезности -  $u_i$ . Соответственно, для любого решения строится свой вектор, состоящий из индивидуальных полезностей агентов -  $(u_1, \dots, u_n)$ , где  $n$  - количество всех агентов. Вся необходимая информация заключена во множестве этих допустимых векторов полезностей.

По своей сути, выбор одного решающего вектора и является ключевой задачей правильного коллективного принятия решений, и как следствие правильного выбора для интеллектуальной системы. В зависимости от способа выбора различают два направления утилитаризма:

- Эгалитаризм - стремление уравнивать индивидуальные полезности;
- Классический утилитаризм - максимизация суммы индивидуальных полезностей.

Прежде чем мы перейдем к изучению этих подходов надо отметить два основных постулата коллективного принятия решений:

Первый из них, это принцип равенства. Равное распределение благ является наиболее простым и фундаментальным воплощением справедливости. В моделях благосостояния этот принцип означает уравнивание индивидуальных полезностей.

Второй принцип - это принцип единогласия. Суть его проста: если для всех агентов решение  $x$  лучше решения  $y$ , то решение  $y$  не должно быть принято. В модели благосостояния этот принцип называется также принципом оптимальности по Парето. И обычно требуют, чтобы выбранный вектор полезностей был оптимальным по Парето (или, другими словами, не доминировался по Парето никаким другим допустимым вектором полезностей). Оптимальное по Парето решение называется также эффективным решением.

Однако, зачастую, оба постулата могут быть несовместными и возникает широко известная дилемма равенство - эффективность.

**Эгалитаризм** является воплощением первого постулата коллективного принятия решений - принципа равенства, и подразумевает, что к равноправным агентам должно быть равное отношение.

Однако важно понимать главный недостаток эгалитарного подхода - существенное увеличение полезности всех агентов, кроме одного, не стоит даже небольшого уменьшения полезности этого единственного агента.

Рассмотрим принцип эгалитаризма на примере:

*Разрабатывается интеллектуальная система по оценке размещения совместного предприятия для пяти городов  $A, B, C, D$  и  $E$ , соединенных кольцевой дорогой (рисунок 1). Предприятие можно расположить в любом месте на кольцевой дороге. Требуется найти на этой дороге оптимальную точку размещения предприятия.*

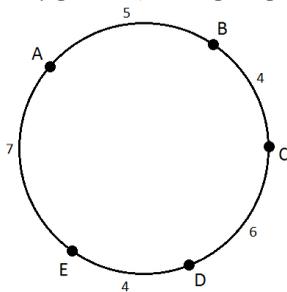


Рисунок 1 – Кольцевая дорога

Индивидуальная полезность агентов (в данном случае городов) будет измеряться расстоянием до предприятия со знаком минус, так как, чем ближе к предприятию расположен город, тем лучше.

Максиминная задача имеет два решения:

1. Точка  $x$  - на расстоянии 0,5 км от города  $C$  на дороге  $BC$ .
2. Точка  $y$  - на расстоянии 0,5 км от города  $C$  на дороге  $CD$ .

Имеем следующие векторы полезности для городов  $A, B, C, D$  и  $E$ :

1.  $u(x) = (-8.5, -3.5, -0.5, -6.5, -10.5)$
2.  $u(y) = (-9.5, -4.5, -0.5, -5.5, -9.5)$

Рассмотрим эти векторы из принципов эгалитаризма. Вектор  $u(x)$  содержит наименее оптимальное значение (-10.5 для города  $E$ ) по сравнению с вектором  $u(y)$ . Поэтому точка  $x$  нас не удовлетворяет, даже несмотря на то, что эта точка имеет более оптимальное решение для городов  $A$  и  $B$  по сравнению с точкой  $y$ .

**Классический утилитаризм** заключается в максимизации суммы индивидуальных полезностей агентов и, как следствие, максимизации общей полезности при заданных физических ограничениях. Можно сказать, что утилитаризм более правильный подход, т. к. он судит по заслугам агентов, а не по их потребностям, как в случае с эгалитаризмом.

Однако у утилитаризма есть один существенный минус. Согласно этому подходу индивидуальные полезности агентов рассматриваются только как способ производства общественного благосостояния и не более. А это означает, что утилитарная программа может не стесняясь пожертвовать интересами отдельного агента ради увеличения коллективной полезности.

Вернёмся к примеру про кольцевую дорогу (рисунок 1).

Совершенно очевидно, что для максимизации общей полезности совместное предприятие должно быть размещено в одном из городов  $A, B, C, D$  или  $E$ . Составим таблицу всех возможных значений, учитывая, что индивидуальная полезность агентов измеряется расстоянием до предприятия с отрицательным знаком.

Таблица 1 – Таблица векторов полезности при утилитарном подходе

| Полезность | $A$ | $B$ | $C$ | $D$ | $E$ |
|------------|-----|-----|-----|-----|-----|
| $u(A) =$   | 0   | -5  | -9  | -11 | -7  |
| $u(B) =$   | -5  | 0   | -4  | -10 | -12 |
| $u(C) =$   | -9  | -4  | 0   | -6  | -10 |
| $u(D) =$   | -11 | -10 | -6  | 0   | -4  |
| $u(E) =$   | -7  | -12 | -10 | -4  | 0   |

В точке  $C$  наблюдается наиболее оптимальное решение задачи. Сравним полученный вектор  $u(C)$  с вектором  $u(y)$ , полученным при эгалитарном подходе.

1.  $u(C) = (-9, -4, -0, -6, -10)$
2.  $u(y) = (-9.5, -4.5, -0.5, -5.5, -9.5)$

В случае эгалитарного подхода наиболее плохая оценка составляет "**-9.5**" для городов  $A$  и  $B$ . Утилитарный же подход предоставляет более выгодные значения для всех городов, кроме города  $E$ , что обуславливается существенным минусом утилитарной программы, о котором говорилось выше.

Поэтому если интеллектуальная система будет руководствоваться принципами эгалитаризма, то должна будет выбрать точку на расстоянии 0,5 км от города  $C$  на  $CD$  для размещения предприятия, а если принципами утилитаризма, то город  $C$  [3].

Рассмотрим ещё один пример:

*Группа разработчиков проектирует автопилот для управления машиной. Ключевым вопросом разработки становится решение машины в критических ситуациях. Разработчики рассматривают пример, когда ситуация на дороге вышла из под контроля и от действий автопилота могут пострадать две группы людей. Предполагается,*

*что одна из них меньше по численности, чем вторая. Какой стратегии надо придерживаться автопилоту, чтобы последствия принятого решения максимизировали коллективное благосостояние?*

Если разработчики будут придерживаться эгалитарного подхода, то группы пострадавших должны быть рассмотрены как равносильные элементы. В этом случае, обеим группам будут даваться равные шансы на выживание, не зависящие от численности людей, состоящих в них.

Если же подход разработчиков в данном вопросе будет соответствовать понятиям классического утилитаризма, то автопилот должен сохранить наибольшее возможное число жизней. В самом лучшем случае, ни одна из этих групп не должна пострадать, однако в случае рассмотренного примера, автопилот должен будет пренебречь интересами малочисленной группы в пользу многочисленной.

Как видно, коллективное благосостояние в обоих случаях максимизируется по-своему. Выбор конкретного варианта целиком и полностью ложится на плечи коллектива разработчиков, и именно они будут отвечать за последствия решений автопилота.

Подводя итог, отметим, что особый интерес для интеллектуальных систем заслуживают решения, затрагивающие интересы нескольких равноправных участников. В таком случае возникает несколько оптимальных векторов решения, выбор единственного из которых может определяться только подходом, заложенным в интеллектуальную систему.

Подход эгалитаризма даёт сильную внутреннюю устойчивость (за счёт уравнивания благ между агентами). И при этом полностью игнорирует внешнюю устойчивость - для уравнивания благ эгалитаризм готов уменьшить долю благ каждого практически до нуля.

Классический утилитаризм идет в обратном направлении: он максимизирует блага и этим гарантирует внешнюю устойчивость. Однако он полностью игнорирует внутреннюю устойчивость.

Эксперименты подтверждают разумность эгалитаризма там, где полезности выражают объективные потребности. Утилитаризм же резонно применять там, где заслуги агентов имеют существенный вес.

На данный момент именно утилитаризм является широко используемой моделью принятия решений в интеллектуальных системах, поскольку он может гарантированно обеспечить максимизацию общего благосостояния.

## **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Бережная, Е.В. Методы и модели принятия решений: Учебное пособие / Е.В. Бережная, В.И. Бережной. – М.: Инфра-М, 2016. – 384 с.

2. Семенов, С.С. Методы и модели принятия решений в задачах оценки качества и технического уровня сложных технических систем / С.С. Семенов, Е.М. Воронов, А.В. Полтавский, А.В. Крянев. – М.: Лепанд, 2019. – 516 с

3. Sanchez i Marre, Gibert, Evolution of Decision Support Systems, University of Catalunya, 2012

УДК 004.03

### **Е.В. КОСОРУКОВА, П.О. АЛЕКСАНОВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **РАЗРАБОТКА 3D-МОДЕЛЕЙ СРЕДСТВАМИ САПР**

*В статье рассматриваются основные категории САПР, их функции и задачи, которые они решают. Также описываются некоторые виды моделирования.*

Системы автоматизированного проектирования (САПР) в зависимости от своего функционала подразделяются на несколько категорий. Выделяют четыре категории САПР.

#### **Первая категория САПР.**

Данная категория является делением САПР на системы нижнего, среднего и верхнего уровней. Часто еще говорят легкие, средние и тяжелые САПР.

Системы нижнего уровня используются для выполнения практически всех работ с двумерными чертежами. Они имеют ограниченный набор функций для создание отдельных трехмерных моделей. С помощью этих систем выполняется почти 90% всех работ по проектированию. Область их работы – создание чертежей отдельных деталей и сборок. К этим системам относятся VersaCAD, AutoCAD. При этом здесь используется документооборот в основном на бумажных носителях.

Системы среднего уровня позволяют выполнять заметно больше функций, к которым относится не только создание 3D моделей, чертежей и сборочных единиц, но и проведение различных инженерных расчетов. Также в такие системы могут быть встроены модули для решения специализированных задач – автоматизации проектирования пресс-форм, электрических, гидравлических и прочих вспомогательных систем. В системы среднего уровня можно интегрировать модуль

подготовки управляющих программ для станков с ЧПУ. Их популярность очень высока. К ним можно отнести T-Flex, SolidWorks, Компас 3D. Пример моделирования в Компас 3D представлен на рисунке 2.

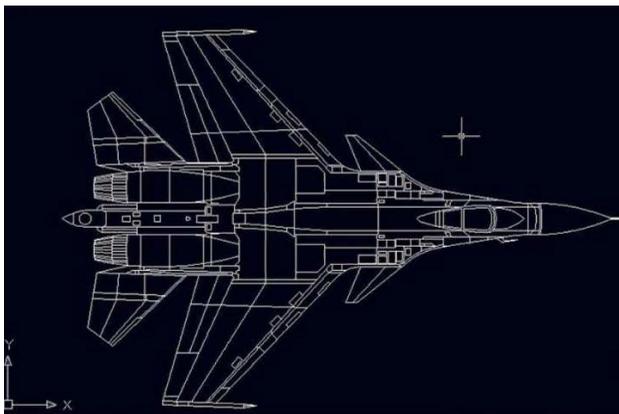


Рисунок 1 – Работа, созданная в Autocad

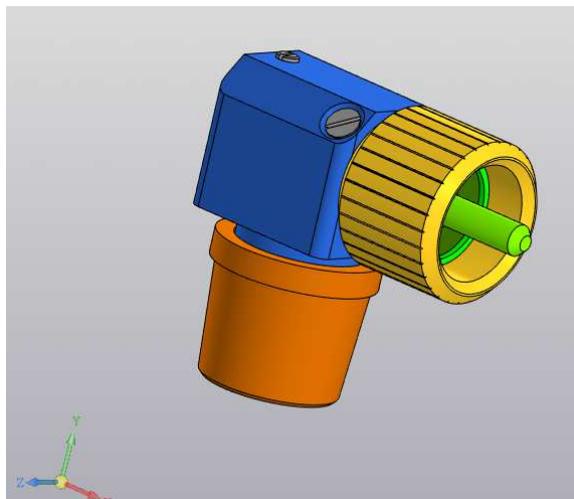


Рисунок 2 – «Разъем высокочастотный» в Компас 3D

Системы верхнего уровня обеспечивают создание проектно-технологической среды, в которой могут работать одновременно все участники проекта. Кроме того, создают условия для всего цикла создания изделия от концептуальной идеи до ее реализации. Возможно применение этих систем для решения наиболее трудоемких задач как:

1. Моделирование поведения сложных механических систем в реальном масштабе времени;
2. Проведение оптимизированных расчетов с визуализацией результатов;
3. Расчет температурных полей и теплообмена.

В состав систем верхнего уровня (Тяжелых САПР) входят:

- графические модули;
- модули для проведения расчетов;
- модули для проведения моделирования;
- постпроцессоры для станков с ЧПУ.

К данным системам можно отнести PTC Creo, Siemens NX.

Большинство пользователей отдают предпочтение САПР среднего и верхнего уровней. Здесь предполагается создание эскизов в строго определенных плоскостях, которые служат основой для построения трехмерных элементов. Непосредственно эскизы удобны в использовании, есть возможность различных взаимосвязей между элементами эскизов, определение размеров и ограничений.

#### **Вторая категория САПР.**

К данной категории относятся системы для промышленного дизайна, использующие в качестве способа работы поверхностное моделирование. Работать в этих системах можно в более свободном стиле без привязки к определенной координатной системе, без задания взаимосвязей и ограничений. К примеру, Rhinoceros 3D, SolidThinking.

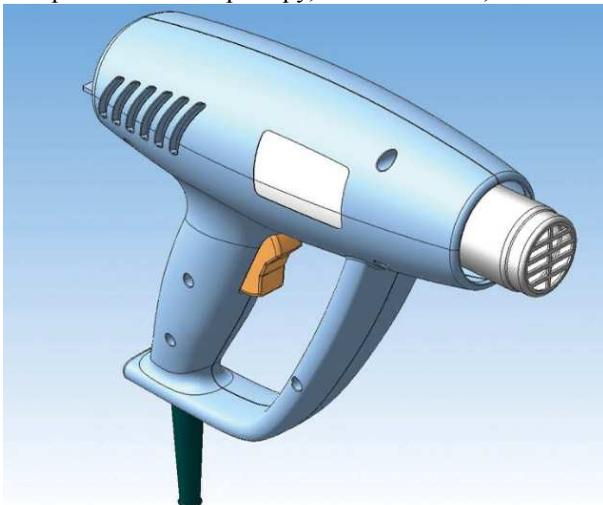


Рисунок 3 – Деталь, созданная поверхностным моделированием

### Третья категория САПР.

Это программы, предназначенные для решения узкоспециализированных задач. Здесь можно отметить такие системы как EXOCAD Dental CAD, PLANMECA PROMAX 3D – для стоматологов, FanReal, Cameo, PetternViewer – для швейной промышленности, Jewelry CAD Dream, Firestorm – для ювелиров.

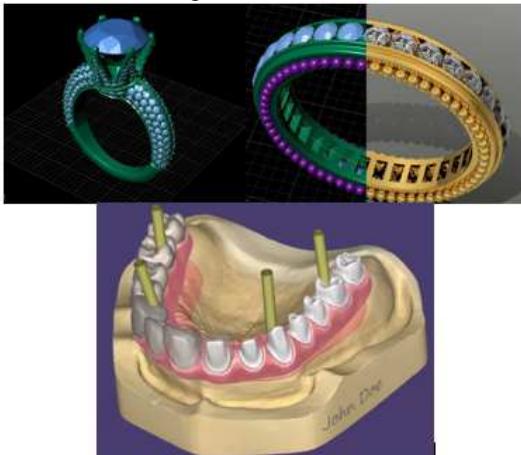


Рисунок 4 – Узкоспециализированное моделирование

### Четвертая категория САПР .

Это системы для художественного моделирования, создания анимации и визуализации.

Основная работа ведется с помощью полигонов и чем больше мы хотим детализировать наше изделие, тем больше полигонов используется. Пример полигонального моделирования представлен на рисунке 5.

Полигональное моделирование позволяет совершать манипуляции разного рода с сеткой 3D-объекта. Данный способ является основным видом моделирования, он дает возможность создавать объекты любой сложности с помощью соединения групп полигонов.

Выделяют 3 типа полигонального моделирования.

1. Низкополигональное (Low-Poly). Используется небольшое число полигонов, что экономит ресурсы.

2. Среднеполигональное (Mid-Poly). Направлено на необходимый результат при визуализации.

3. Высокополигональное (High-Poly). Используется большое число полигонов, что позволяет увеличить компьютерную производительность.

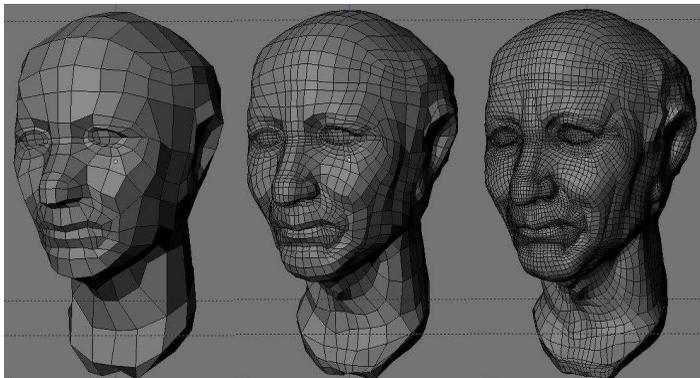


Рисунок 5 – Моделирование головы человека методом полигонов

Таким образом, можно вынести преимущества 3D-моделирования: наглядность; быстрота работы; нет необходимости в физической модели; функциональность; возможность расчета различных свойств изделия; гибкость изменения модели.

#### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. А.А. Черепашков. Компьютерная графика и геометрическое моделирование в машиностроении. СамГТУ, 2005.
2. Статья «3D моделирование в CAD» [Электронный ресурс]. – Режим доступа: <https://cadcamtutorials.ru/articles/cnc5>. (Дата обращения: 8.04.2022).

УДК 004.045

#### **А.С. КОЧЕПАСОВ**

МИРЭА Российский технологический университет

#### **ПРОГРАММНЫЕ РЕШЕНИЯ ДЛЯ УЧЁТА РАБОЧЕГО ВРЕМЕНИ СОТРУДНИКОВ ВНЕ ПОМЕЩЕНИЙ**

*Рассматриваются вопросы учёта рабочего времени сотрудников вне помещений и имеющиеся решения.*

В современном мире существует большое количество профессий, требующих от сотрудника нахождения вне привычных помещений – офисов, производственных цехов и собственных квартир. Самыми распространенными примерами таких профессий являются многие

обслуживающие должности в нефтегазовой промышленности и не только: линейные обходчики, инженеры по монтажу и обслуживанию оборудования, супервайзеры, путевые монтеры и многие другие. Обычно такие вакансии подразумевают вахтовый метод работы с ведением журналов, в которые заносится вся необходимая информация (о начале и окончании рабочего периода, о состоянии объектов, о выполненной работе), которая затем передается диспетчерам по завершении определенного промежутка времени.

Такой подход сложился традиционно и имеет несколько значительных недостатков:

- информация, которая может быть критически важной (например, сообщения о неисправностях), передается не сразу;
- невозможно проверить, насколько записи о выполненной работе в журнале соответствуют действительности;
- есть вероятность ошибки – сотрудник может внести некорректные данные или забыть сделать запись в журнале.

#### **Анализ информационных процессов в предметной области**

На предприятии учет рабочего времени и наработки осуществляется с помощью системы управления ресурсами (СУР). В профессиях, предусматривающих долгое нахождение вдали от технологических средств связи, информационные процессы происходят редко и чаще всего по некоторому расписанию. Обычно это процессы передачи информации, когда сотрудники передают данные, собранные во время рабочего процесса, в пункты управления (диспетчерам, начальству и т.д.). Примерами таких информационных процессов могут быть перечисленные ниже процессы:

1. Линейный обходчик при осмотре выделенного участка собирает показания манометров, а также осматривает общее состояние конструкций, находящихся на участке. Записанные показания и информация о состоянии конструкций после обхода и возвращения передается диспетчеру через радиостанцию или другие виды связи [1].

2. Монтер пути обязан сообщать непосредственному руководителю по факту выполнения ремонтных работ о состоянии выполненных работ, в частности, о неисправностях, которые было невозможно исправить на момент проведения ремонтных работ [2].

3. Лесничество предоставляют информацию о нарушениях и проблемах, найденных в подконтрольных им территориях, а также обеспечивают ведение документации по государственному учету лесов, государственному лесному кадастру и отраслевому учету состояния и использования земель лесного фонда [3].

Профессии, подразумевающие работу разъездного характера, существуют и в условиях постоянной доступности связи. Сюда относятся специалисты по различным ремонтным работам, техники по обслуживанию оборудования и систем, риэлторы.

### **Проблема учета рабочего времени сотрудников**

Согласно статье 91 Трудового Кодекса РФ, рабочее время - время, в течение которого работник в соответствии с правилами внутреннего трудового распорядка и условиями трудового договора должен исполнять трудовые обязанности, а также иные периоды времени, которые в соответствии с настоящим Кодексом, другими федеральными законами и иными нормативными правовыми актами Российской Федерации относятся к рабочему времени [4].

Работодатель обязан вести учет времени, фактически отработанного каждым работником [4].

Учет времени может помочь компании:

- выявлять неэффективно работающих сотрудников;
- поощрять и применять дисциплинарные взыскания в зависимости от количества наработанного времени и других факторов;
- понимать, какие бизнес-процессы занимают больше всего времени и как можно их оптимизировать.

Все эти улучшения в итоге приводят к увеличению выручки, что является первостепенной целью любого бизнеса. Кроме того, сотрудники будут иметь доказательства того, что они занимались полезной деятельностью в рабочее время, а руководителю – лучше понять рабочие процессы и своих подчинённых.

Особенно остро проблема учёта рабочего времени стоит для сотрудников с разъездным характером работы – не всегда понятно, что именно считать рабочим временем, а также непосредственное отслеживание становится труднее вследствие постоянного нахождения сотрудника вне расположения организации.

В письме Роструда от 12.12.2013 N 4209-ТЗ разъездной предлагается считать такую работу постоянного характера, при которой работник выполняет свои должностные обязанности вне расположения организации [5].

Несмотря на то, что для учёта рабочего времени сотрудников, работающих за компьютерами в офисе или удаленно, существует большое количество систем, решений, созданных непосредственно для учёта рабочего времени сотрудников вне помещений, на рынке на данный момент не так много.

### **Анализ характеристик имеющихся решений**

В данном разделе производится обзор программных продуктов и услуг, решающих задачи учёта рабочего времени сотрудников вне помещений.

### **Hubstaff Field**

Программный продукт Hubstaff Field является частью системы Hubstaff, предназначенной для учёта рабочего времени сотрудников, измерения эффективности работы, автоматизации выплат сотрудникам и увеличения доходов бизнеса путём определения наименее эффективных бизнес-процессов [6].

Данное решение позволяет:

- отслеживать местоположение сотрудников с помощью GPS: после установки приложения на смартфон сотрудника в центре управления можно просматривать маршруты передвижения и нынешнее местоположение;

- создавать зоны работы: на карте можно отметить зоны, в которых ожидается выполнение работ сотрудниками, и назначить их командам;

- учитывать время: при входе назначенных сотрудников в определённую зону начинается автоматический отсчёт времени, а при выходе приостанавливается или завершается совсем, если отмечено, что работы завершены;

- назначать задания: в центре управления есть возможность создавать и назначать задания для отдельных сотрудников или команд, указывая время начала работ, местоположения и другие детали;

- настраивать графики работ и отпусков: для каждого сотрудника можно создать график, в котором будут указаны рабочие дни, выходные и отпуска, также в график можно добавлять повторяющиеся задачи;

- настраивать бюджеты: для каждого задания или проекта можно создать бюджет и отслеживать, сколько средств было потрачено на различные расходы [6].

Программный продукт состоит из трёх приложений: мобильного приложения, сервера и центра управления. Мобильное приложение устанавливается всеми сотрудниками, которые выполняют работу вне помещений, а центр управления предназначен для руководящих должностей. Мобильное приложение собирает такие данные как время и геолокацию автоматически, а также позволяет вносить некоторые (например, отметки о выполнении работы или фотографии) вручную. Данные передаются на сервер, который находится в облаке или на устройствах организации, откуда доступ к ним можно получить через

центр управления, в котором на основе полученных данных руководитель принимает различные решения и настраивает рабочие процессы.

Определение местоположения обеспечивается с помощью технологии GPS, а передаются данные через сеть Интернет.

Тарифный план, включающий весь перечисленный функционал, стоит 10 USD за одного сотрудника в месяц, не менее двух сотрудников. Также доступна возможность составления собственного плана для крупных компаний с выбором необходимого функционала, расширенной технической поддержкой и увеличенным количеством запросов к API [6].

### **Мегафон Контроль кадров**

Данная услуга является решением для отслеживания перемещения разъездных сотрудников и контроля выполненных задач [7].

Функциональные возможности:

- контроль местонахождения сотрудников на карте в режиме онлайн по данным сотовой сети и спутниковым координатам;
- координация перемещений сотрудников между заказами, помощь в поиске объекта на карте;
- отправка SMS сотруднику или группе сотрудников через сайт услуги;
- контроль задач для сотрудников с фото- и геоподтверждением выполнения в личном кабинете и мобильном приложении;
- просмотр истории перемещений сотрудников на карте или в отчётах;
- контроль соблюдения рабочего графика, посещения заданных объектов и контроль времени [7].

Номера разъездных сотрудников регистрируются в личном кабинете услуги, который представляет собой веб-приложение. Кроме того, на смартфон сотрудника может быть установлено приложение, повышающее точность определения местоположения и позволяющее отправлять сообщения и фото руководителям. В личном кабинете есть возможность создавать зоны работ, присылать SMS-сообщения сотрудникам и проверять их местоположение.

### **Time Doctor**

Time Doctor – решение, используемое для учета рабочего времени сотрудников за компьютером. Несмотря на то, что большая часть функционала подразумевает работу сотрудника за персональным компьютером (рабочей станцией, ноутбуком), среди вариантов так же есть и мобильное приложение, предоставляющее следующие возможности:

- отслеживание таких действий пользователя, как включение и выключение устройства;

- совершение снимков экрана через случайные или регулярные интервалы;
- отслеживание используемых сотрудником приложений и посещаемых сайтов;
- напоминания сотрудникам о предстоящих задачах или слишком долгих перерывах;
- создание расписаний, назначение задач сотрудникам;
- создание отчетов о продуктивности сотрудников, выявление наиболее и наименее эффективных из них;
- автоматическая оплата труда в зависимости от отработанного времени [8].

Приложение не поддерживает отслеживание местоположения сотрудников и геозонирование, и поэтому может использоваться только как «записная книжка», в которую сотрудником вносятся данные о начале и завершении работ, что не исключает возможность попыток обмана, и полезно только в том случае, если выполнение работ ведётся на мобильном устройстве.

Все данные собираются и передаются в реальном времени по сети Интернет.

Решение имеет 14-дневный бесплатный период, а также три тарифных плана, отличающихся функционалом, количеством доступных для подключения отделов, длительностью хранения данных и доступностью технической поддержки. Стоят они соответственно 7 USD / 10 USD / 20 USD за одного сотрудника в месяц.

### **BioTime**

Основными направлениями деятельности решений компании Biolink Solutions являются системы массовой идентификации, защита информации, учет рабочего времени и контроль доступа с использованием биометрических технологий. Биометрическая система BioTime разработана для обеспечения учета рабочего времени и контроля доступа, и имеет версию в виде мобильного приложения [9].

Мобильное приложение позволяет контролировать удалённо работающих сотрудников или подтверждать присутствие на рабочих местах без установки терминалов контроля.

Для руководителя доступны следующие возможности:

- просмотр данных о выходе на работу, уходе и присутствии на рабочем месте по каждому сотруднику отдельно или с группировкой по должностям или отделам;
- получение push-уведомлений о присутствии на рабочем месте сотрудников;

– возможность настройки автоматических звонков или отправки SMS сотрудникам [10].

Используя приложение, сотрудник начинает учет рабочего времени, для чего определяется его местоположение, и подтверждает его при помощи отпечатка пальца или селфи. Данные затем передаются на сервер, где местоположение сотрудника сверяется с его местом работы, а отпечаток пальца или фото используются для подтверждения личности [10].

На момент написания статьи компания предлагает персонализированные решения для штатов сотрудников различных размеров с разным оборудованием, для получения предложения необходимо оставить заявку с указанием информации о компании.

В таблице 1 произведено сравнение рассмотренных решений по ключевым признакам.

Таблица 1 – Сравнительная таблица рассматриваемых решений

| Решение                 | Наличие функционала         |                |                       |                     |                       | Цена в месяц за сотрудника, RUB |
|-------------------------|-----------------------------|----------------|-----------------------|---------------------|-----------------------|---------------------------------|
|                         | Отслеживание местоположения | Геозонирование | Учет рабочего времени | Связь с сотрудником | Отслеживание действий |                                 |
| Hubstaff Field          | Да                          | Да             | Да                    | Нет                 | Нет                   | От 834                          |
| Мегафон Контроль кадров | Да                          | Да             | Нет                   | Да                  | Нет                   | От 120                          |
| Time Doctor             | Нет                         | Нет            | Да                    | Нет                 | Да                    | От 584                          |
| BioTime                 | Да                          | Да             | Нет                   | Да                  | Нет                   | Рассчитывается по заявке        |

Примечание. Цены в USD переведены в RUB по курсу Центробанка РФ на момент написания статьи.

Исходя из сравнительной таблицы, главными недостатками приведенных решений можно назвать:

– некоторые решения являются зарубежными, что может накладывать ограничения на их использование в государственных предприятиях;

– большинство решений не позволяет учитывать рабочее время сотрудников, пока те находятся вне помещений и/или заранее установленных геозон;

– рассмотренные решения имеют слабую интеграцию с популярными в России системами управления персоналом или же не имеют таковой вовсе.

Таким образом, прослеживается необходимость в разработке средства отслеживания и фиксации работы вне помещений.

Средства могут быть реализованы в виде устройства, выдаваемого сотруднику, которое выступает в роли терминала, отслеживающего местоположение с помощью геолокации по технологии GPS или используя сотовые вышки. При данной реализации подразумевается проектирование системы с клиент-серверной архитектурой (показанной на рисунке 1), где выдаваемое устройство является клиентом, собирающим данные и затем отправляющее их на сервер. Операторы, собирающие и анализирующие данные, затем могут получить доступ к ним через рабочие станции.

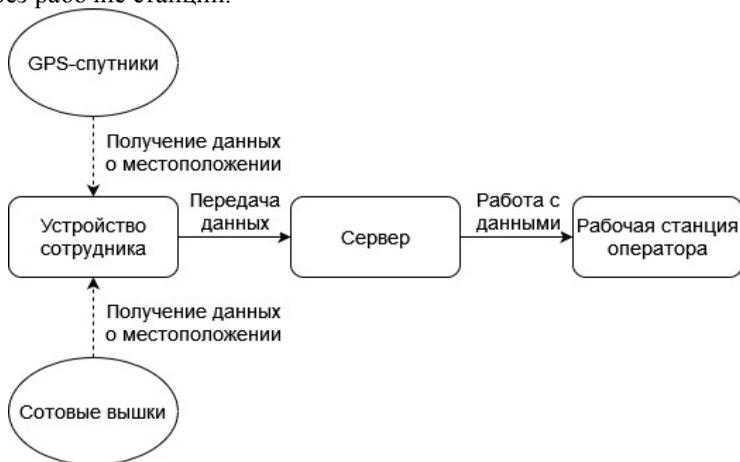


Рисунок 1 – Архитектура системы для учёта рабочего времени сотрудников вне помещений с использованием устройства у сотрудника

Другим вариантом такого средства может быть учёт рабочего времени сотрудника с помощью внешнего видеонаблюдения.

Видеонаблюдение возможно реализовать с помощью стационарных камер наблюдения при условии того, что работа выполняется на заранее известной и относительно небольшой территории, или же с помощью видеонаблюдения с беспилотных летательных аппаратов. Такой вариант подходит для обширных территорий, на которых нет возможности размещать стандартные камеры видеонаблюдения на регулярных интервалах.

В данном варианте также необходимо наличие фиксации появления сотрудника в кадре, что может выполняться как оператором, так и автоматизировано с помощью соответственного программного обеспечения.

В заключении можно отметить, что рассмотренные решения могут получить ограничения в использовании на российском рынке в текущих реалиях, и поэтому может понадобиться создание программных продуктов, которые смогут быть интегрированы в описанные архитектуры. Решения, спроектированные таким образом, позволят исправить проблему в нехватке средств учёта рабочего времени вне помещений, избегая описанных недостатков, чтобы помочь компаниям, имеющим большое количество сотрудников с работой разъездного характера, упростить учёт рабочего времени.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Должностная инструкция обходчика линейного 4-го разряда [Электронный ресурс]. – Режим доступа: <http://qualinfo.ru/official-instructions/obhodchik-linejnyj-4-go-razryada-2663/>.
2. Должностная инструкция монтера пути 3-го разряда [Электронный ресурс]. – Режим доступа: <https://instrukzii.ru/rabochie/zd/monter-puti-3-go-razryada.html>.
3. Тарифно-квалификационные характеристики должностей работников лесного хозяйства [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/9015836>.
4. Трудовой Кодекс Российской Федерации [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/901807664>.
5. Федеральная служба по труду и занятости: Письмо № 4209-ТЗ от 12.12.2013 [Электронный ресурс]. – Режим доступа: <https://www.klerk.ru/doc/484449/>.
6. Field Service Tracking Software | Hubstaff Field [Электронный ресурс]. – Режим доступа: <https://hubstaff.com/solutions/field-services>.
7. Контроль кадров [Электронный ресурс]. – Режим доступа: <https://kk.megafon.ru/landing>.

8. Simple and Accurate Time Tracking with Time Doctor | Time Doctor [Электронный ресурс]. – Режим доступа: <https://www.timedoctor.com/time-tracking-with-time-doctor.html>.

9. О компании Biolink Solutions [Электронный ресурс]. – Режим доступа: [https://www.biotime.ru/company\\_biotime/](https://www.biotime.ru/company_biotime/).

10. Мобильные решения – система Biotime [Электронный ресурс]. – Режим доступа: <https://www.biotime.ru/advantages/mobilnye-resheniya.php>.

УДК 621.396

**А.Д. КОШЕЛЕВ, М.С. ИГОШИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ МОДУЛЯТОРА ДЛЯ УСИЛЕНИЯ СВЧ-СИГНАЛА РАДИОЛОКАЦИОННОЙ БОРТОВОЙ СИСТЕМЫ**

*В статье рассматривается способ проектирования модулятора для усиления СВЧ-сигнала радиолокационной бортовой системы (БРЛС). Кратко описывается принцип работы устройства, его функциональные возможности и назначение. Разбираются ключевые сложности, возникающие в ходе проектирования устройства, а также приводятся преимущества предлагаемой конструкции.*

#### **Введение**

Одна из важнейших задач любого государства заключается в удовлетворение потребностей в создании вооружений и военной техники, обеспечивающей необходимый уровень обороноспособности страны. Основные приоритетные направления в этой области – разработка, производство и модернизация радиолокационных комплексов, средств вычислительной техники, радиолокационного оборудования, систем управления оружием, другой продукции военного, двойного и гражданского назначения.

По своему назначению, радиолокационная бортовая система (БРЛС) относится к навигационной пилотажной системе самолета и является информационной системой. С таким изделием самолет обладает тактическим преимуществом. Например, он может скрыться за складками рельефа местности и оказаться недоступным для визуального наблюдения со стороны противника. БРЛС помогает летчику ориентироваться в отсутствии визуальной видимости, повышает безопасность полета, предупреждая экипаж о наличии по курсу полета

опасных препятствий, в том числе проводов и опор линий электропередач (ЛЭП), в любое время суток и при любой погоде [3].

БРЛС состоит из нескольких съемных блоков, расположенных в носовой части самолета: передатчика, антенной системы, приемника, процессора обработки данных, программируемого процессора сигналов, пультов и органов управления и индикации [1].

Сегодня практически у всех бортовых РЛС антенная система представляет собой плоскую щелевую антенную решетку, антенну Кассегрена, пассивную или активную фазированную антенную решетку.

Выделяют несколько типов БРЛС: РЛС заднего или бокового обзора, поиска надводных целей, радиолокационные прицелы и т.д. [4].

В данной статье описывается конструкция модулятора, обеспечивающего формирование сигналов управления и напряжений питания комплексированного изделия СВЧ, необходимого для корректной работы БРЛС.

#### ***Модулятор для усиления СВЧ-сигнала БРЛС***

Конструктивно радиолокационная система состоит из нескольких съёмных блоков. Изделие, в состав которого входит модулятор усиления СВЧ-сигналов, выполняет следующие функции:

- Картографирование земной поверхности;
- Обнаружение подвижных и неподвижных целей с определением их координат;
- Обнаружение воздушных целей и степени их опасности;
- Обеспечение маловысотного полета с обнаружением опасных объектов, в том числе опор и проводов ЛЭП.

Для правильной работы блока и всей БРЛС модулятор должен удовлетворять следующим требованиям:

- Доступность и ремонтпригодность;
- Сохранение работоспособности в диапазоне температур окружающей среды от  $-50^{\circ}\text{C}$  до  $+80^{\circ}\text{C}$ , относительной влажности воздуха 95% при температуре  $+35^{\circ}\text{C}$ ;
- Наличие небольших габаритов и массы;
- Низкая себестоимость;
- Высокая экономическая эффективность.

Модулятор предназначен для генерирования мощных кратковременных импульсов электрических колебаний сверхвысокой частоты (СВЧ) со строго определенной периодичностью, задаваемой схемой синхронизации [2].

Разрабатываемое устройство выполнено в виде съемного узла, представляющего собой раму из полиамида, на которой установлены,

пять узлов, каждый из которых является печатной платой (ПП). Конструкция предусматривает электровентилятор, предназначенный для охлаждения модулятора, работает постоянно при наличии питающего напряжения 200В.

*Устройство запуска модулятора (УЗМ).* Его работа заключается в коммутации импульсов запуска передатчика (ИЗП) в зависимости от режима работы, создания необходимых задержек ИЗП, а также их оптической трансляции на формирователи.

*Формирователи,* в свою очередь, создают стабилизированное напряжения и напряжение накала управляющих электродов в различных режимах, а также усиливают приходящие с платы УЗМ импульсы ИЗП до требуемых.

*Стабилизатор* направлен на обеспечение напряжения смещения, необходимого для управляющих электродов.

*Трехфазный трансформатор,* подключенный первичными обмотками к сети 200В/400Гц, на вторичных обмотках формирует напряжения, необходимые для питания всех плат модулятора.

Конструкция устройства представлена на рисунке 1.

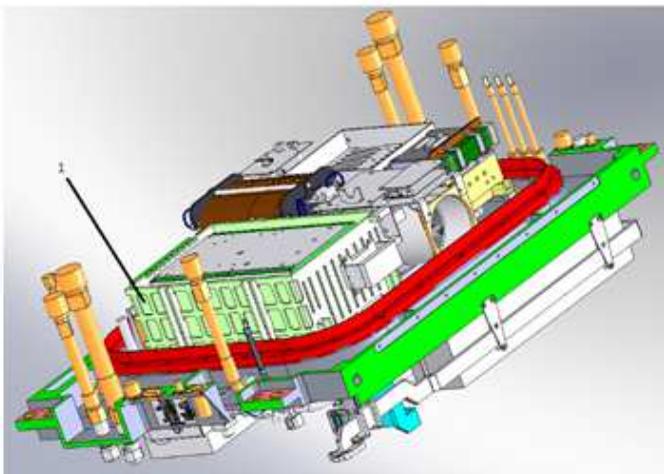


Рисунок 1 – Конструкция модулятора усиления СВЧ-сигналов БРЛС

Преимуществами подобной конструкции являются следующие особенности:

1. За счёт того, что модулятор выполняется на единой конструктивно-технологической и элементных базах, снижаются затраты на

материалы и детали, а соответственно и затраты на разработку самой конструкции.

2. Устройство наделено современными комплектующими, что позволяет повысить технические характеристики, уменьшить габариты и вес устройства, следовательно, снизить расходы на материалы.

3. За счёт всё тех же востребованных комплектующих модулятор является долговечным, что уменьшает долю амортизационных отчислений в расходах на его эксплуатацию, сокращает расходы на ремонт и убытки от простоев в ремонтах.

4. Применение стандартных и унифицированных элементов, использование высокопроизводительных методов при обработке и сборке деталей позволяют выпускать разрабатываемое устройство серийно.

5. Как уже было сказано ранее, конструкция каждого из пяти узлов представляет собой ПП, установленную в отдельный отсек корпуса модулятора. Благодаря подобной компоновке внутренних узлов достигается высокий уровень надёжности и виброустойчивости устройства.

#### **Заключение**

Рассматриваемая конструкция модулятора усиления СВЧ-сигналов БРЛС удовлетворяет требованиям, необходимым для корректной работы бортовой радиолокационной системы. Вышеперечисленные преимущества устройства являются доказательством целесообразности его разработки по экономическим, техническим и эксплуатационным показателям.

#### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Ахияров В.В., Нефедов С.И., Николаев А.И. Радиолокационные системы: учебное пособие /под редакцией А. И. Николаева. – 2-е изд. – Москва: МГТУ им. Баумана, 2018. – 349 с.

2. Бердышев В.П., Гарин Е.Н., Фомин А.Н. Радиолокационные системы: учебник. – Красноярск: СФУ, 2021. – 400 с.

3. Денисов В.П. Радиолокационные системы: учебно-методическое пособие /В. П. Денисов. – Москва: ТУСУР, 2012. – 21 с.

4. Масалов Е. В. Радиолокационные системы: учебное пособие / Е. В. Масалов. – Москва: ТУСУР, 2012. – 118 с.

УДК 004.514

**А.Г. КРУГМАН**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ УПРАВЛЕНИЯ  
УМНЫМ ДОМОМ**

*Рассматривается пример создания приложения для управления умным домом для устройств с операционной системой Android.*

Главная цель всех технических новинок – сделать жизнь человека максимально комфортной, особенно в рамках жилого пространства. С одной стороны, прогрессивные технологии помогают сделать дом теплым, светлым, энергоэффективным, а с другой, появляется столько подсистем, что управление всей этой инженерией становится непосильной задачей даже для продвинутого пользователя. «Умный дом» – это интеллектуальный комплекс, объединяющий все важные технические системы и предназначенный для того, чтобы управление ими стало простым и удобным. Обеспечивается данная простота за счёт удобного пользовательского интерфейса в приложении. Удобство гарантируется наличием всех необходимых настроек в одном месте прямо в руке у пользователя.

В текущей реализации в приложении будет возможность просматривать изображение с ip камеры, расположенной дома. А также возможность управлять питанием нескольких устройств.

Данное приложение будет состоять из разных модулей (рисунок 1). Модуль представляет из себя активность, взаимодействующую с пользователем через созданный интерфейс

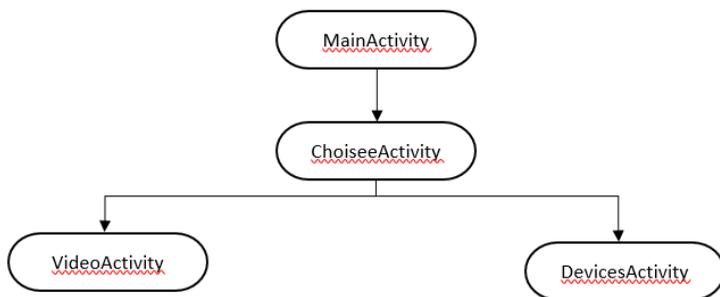


Рисунок 1 – Схема модулей приложения

При запуске разрабатываемого приложения загружается `MainActivity` и отображается окно с полями для ввода почты и пароля. Также присутствует возможность регистрации нового пользователя. Аутентификация реализована с помощью модуля `Firebase Authentication`. Интерфейс `MainActivity` представлен на рисунке 2,а.

После подключения `Firebase` к проекту и внесения необходимых зависимостей, были добавлены методы для регистрации и входа, а также объект класса `FirebaseAuth`, который был инициализирован в методе `onCreate()`.

Оповещение пользователя об успешности и провале действия осуществляется с помощью всплывающих сообщений.

Также к проекту подключена `Firestore Database`. В этой облачной базе данных хранится вся информация о пользователях.

После успешной авторизации, либо регистрации нового пользователя (и добавления соответствующего документа в БД) происходит смена `Activity`. Мы попадаем на `ChoiseActivity`. После перехода осуществляется чтение данных с БД с помощью переданного `UID` (индивидуальный для каждого пользователя. Присваивается пользователю при регистрации) с `MainActivity`. Интерфейс `ChoiseActivity` представлен на рисунке 2,б.

Видимость кнопок настраивается в зависимости от полученных данных. Если пользователь новый и у него ещё нет ссылки на камеру, то ему видны только поле для ввода и подтверждающая кнопка. Если у пользователя уже имеется ссылка на камеру, то ему доступны кнопки для перехода на `VideoActivity` и смены ссылки.

Оповещение (не)успешности получения данных осуществляется с помощью всплывающих сообщений. При нажатии на кнопку `Add link`, происходит вызов метода `updateStatusVideo(String link)`, который обновляет данные о ссылке в БД.

Если же ссылка уже имеется, то пользователь может нажать на кнопку `Video monitoring`. Выполнится переход на `VidoeActivity`. Описание активности будет далее. Также пользователю доступна кнопка `Change link`. При нажатии пользователю станут доступны поле для ввода новой ссылки и подтверждающая кнопка.

Также пользователю доступна кнопка `Devices`. При нажатии на эту кнопку будет выполнен переход на `DevicesActivity`.

На `VidoeActivity` пользователю доступна для просмотра камера, ссылка на которую была добавлена ранее. Интерфейс представлен на рисунке 2, в.

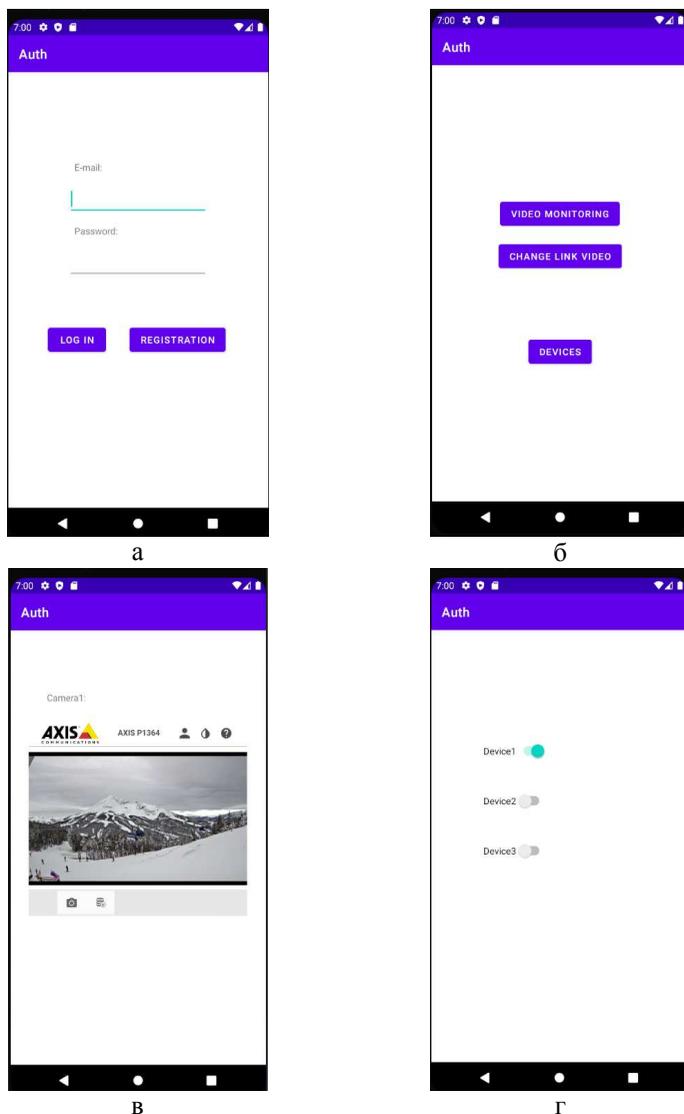


Рисунок 2 – Интерфейс модулей приложения

На DevicesActivity пользователю доступны переключатели для управления питанием приборов. После перехода на данную активность происходит чтение данных о состоянии приборов из БД и установление переключателей в соответствующие позиции. За это отвечает ме-

тод `setPositionSwitch(int number, Switch sw)`. Также при изменении положения какого-либо переключателя вызывается метод `updateStatusDevice(int number, boolean status)`. Его задача изменить значение в БД на значение соответствующее текущему положению переключателя. Интерфейс `DevicesActivity` представлен на рисунке 2,г.

В результате выполнения данной работы было написано приложение для управления умным домом для операционной системы Android. В приложении реализован просмотр изображения с ip камеры, а также управление питанием устройств. Чтобы создать функционал приложения необходимо было изучить документацию по Java и Firebase.

В будущем в приложение будут добавлены отображение прогноза погоды, датчики температуры, статистика энергопотребления.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Firebase Documentation. [Электронный ресурс]. – Режим доступа: <https://firebase.google.com/docs/auth> (дата обращения 01.04.2022).
2. Cloud Firestore. [Электронный ресурс]. – Режим доступа: <https://firebase.google.com/docs/firestore> (дата обращения 01.04.2022).

УДК 622.013

**М.К. КРЫГИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **МЕТОДЫ ОБРАБОТКИ СЕЙСМОАКУСТИЧЕСКОЙ ИНФОРМАЦИИ МОРСКОГО ГЕОФИЗИЧЕСКОГО КОМПЛЕКСА**

*В статье будут рассмотрены методы обработки сейсмоакустической информации и общие требования к созданию программного комплекса технических средств обработки информации морского геофизического комплекса на объекте эксплуатации.*

Сейчас большое распространение находит использование разнообразных сейсмоакустических методов исследований при разработке месторождений полезных ископаемых, а также для обнаружения угроз при добыче. Примеры угроз – выбросы газа, затопления, подводные оползни и разломы. Чтобы обнаружить различные геологические опасности во время разведывания ландшафта для крупных сооружений на акваториях, используются разные виды сейсмических исследований

(сейсмологические наблюдения, сейсмическое районирование, а также акустические исследования грунтовых образцов).

Одним из основных методов изучения строения поверхности земли, а также морского дна является сейсмоакустика. В общем случае задача сейсмоакустических исследований может быть определена как получение информации, при интерпретации которой выявляются геологические структуры, определяются их строение и основные характеристики.

В настоящее время популярна методика получения сейсмоакустической информации, в основе которой лежит использование мощных импульсов от источников взрывного типа, например, от пневматических излучателей (пневмопушек) или от электромагнитных излучателей (бумеров) и электроискровых разрядников (спаркеров).

Обработка материалов, полученных в ходе сейсмоакустических исследований, с целью получения документов для геологической интерпретации результатов и воздействия на весь процесс геолого-геофизических исследований, может быть выполнена только после накопления минимально необходимого для анализа объема сейсмоакустической информации. Решать эту задачу можно, как на борту судна, где производилась разведка сейсмоакустических данных, так и на предприятии. Ввод первичных данных в вычислительную систему (после их предварительного преобразования в устройстве сбора к виду, удобному для последующей обработки) осуществляется либо в режиме непосредственной передачи через соответствующее устройство связи, либо в режиме считывания с промежуточного физического носителя информации. Ввиду весьма больших объемов первичных данных, и удаленности баз, обработку первичных материалов исследований рекомендуется выполнять непосредственно на борту судна. Полученные данные подвергаются обработке с помощью вычислительной техники и специального программного обеспечения. При анализе продуктов обработки сейсмоакустической информации, исследователь должен определять форму и свойства слоев, на поверхности которых случилось преломление или отражение посылаемых упругих волн, а также глубину залегания пород. Считается, что при этом существенно сократится расход носителя при регистрации материалов исследований.

Сейсмоакустические данные представляют собой гармонический сигнал. Обработка включает в себя исследование спектральных характеристик информации. Одним из способов такой обработки является преобразование Фурье.

Ряд Фурье является математическим аппаратом, с помощью которого можно изучать периодические и непериодические функции,

раскладывая их на компоненты. Для разложения функции в ряд Фурье считают, что её можно представить, как сходящиеся тригонометрические ряды на конкретном интервале.

Удобно с помощью периодических функций представлять такие величины, как акустические и электромагнитные волны, токи и напряжения. В сейсмоакустике суть задачи преобразования Фурье – анализировать сигналы, раскладывая их на синусоиды (гармоники). Похожим образом функционируют органы слуха человека, раскладывая звук на отдельные колебания различных частот

Главная идея преобразования Фурье – каждую функцию возможно представить в виде бесконечной суммы синусоид. Характеристики каждой из этих синусоид – частота, начальная фаза и амплитуда.

Преобразования Фурье бывают различных видов и выбор зависит от вида представления сигнала. Рассмотрим два следующих:

- Дискретное преобразование Фурье (ДПФ);
- Быстрое преобразование Фурье (БПФ).

Алгоритмы, реализующие цифровую обработку сигналов, часто используют ДПФ. Прямое преобразование такого вида ставит в соответствие временной функции  $f(n)$ , которая определяется на основе  $N$ -точек измерений на исходном временном интервале, новую функцию  $F(k)$ , которая определяется на частотном интервале. Заметим, что функции на вышеупомянутых интервалах задаются различными способами. На первом – с использованием  $N$ -отсчетов, а на втором интервале – при помощи  $K$ -кратного спектра.

$$F(k) = \sum_{n=0}^{N-1} f(n) * e^{-j * \frac{2\pi}{N} * n * k}, k = 0, \dots, N - 1$$

$N$  - кратность частотного спектра и количественная характеристика сигнала за один период;

$k$  - индекс частоты.

Прямое дискретное преобразование можно сформулировать на основе вещественной и мнимой составляющих. Первая из них является массивом со значениями косинусоидальных составляющих, а мнимая – это массив, в котором хранятся синусоидальные составляющие.

$$\text{Re}(F(k)) = \sum_{n=0}^{N-1} f(n) * \cos \frac{2 * \pi * n * k}{N}, k = 0, \dots, N - 1$$

$$\text{Im}(F(k)) = - \sum_{n=0}^{N-1} f(n) * \sin \frac{2 * \pi * n * k}{N}, k = 0, \dots, N - 1$$

По формулам можно сделать вывод, что они позволяют разложить различные сигналы на гармоники (иначе говоря, синусоидальные

составляющие) с частотами, колебания которых варьируются от одного до  $N$  за период.

Чтобы вычислить один спектральный отсчет, понадобится  $N$  комплексных операций умножения и сложения. Следовательно, сложность ДПФ квадратична, и всего понадобится  $N^2$  вышеупомянутых операций.

БПФ – это математический алгоритм, при помощи которого есть возможность сократить количество выполняемых операций относительно рассмотренного ранее метода. В основе быстрого преобразования лежит разбиение заданного ряда отсчетов дискретного сигнала на некоторое количество промежуточных последовательностей. Отметим, что данный алгоритм высокоточен по отношению к стандартному ДПФ, так как при сокращении количества операций уменьшаются и общие ошибки округления.

Исходными данными для быстрого преобразования является  $N$ -отсчетный сигнал  $f(n)$ . Начальный сигнал разбивается на два равных массива: массив с четными отсчетами  $f_1(n) = f(2*n)$  и с нечетными отсчетами  $f_2(n) = f(2*n+1)$ . По итогу на частотном интервале функция  $F(k)$  будет идентифицироваться равенством:

$$F(k) = \sum_{n=0}^{N-1} f(n) * e^{-j*2\pi*n*k/N} = \sum_{n=0}^{\frac{N}{2}-1} f_1(n) * e^{-j*2\pi*n*k/N} + \sum_{n=0}^{\frac{N}{2}-1} f_2(n) * e^{-j*2\pi*(2n+1)*k/N}, k = 0, \dots, N-1$$

Следовательно, выводим выражения для нахождения  $F(k)$  на обоих частотных интервалах: первый интервал  $[0, \dots, N/2-1]$  и второй  $[N/2, \dots, N-1]$ :

$$F(k) = F_1(k) + e^{-j*2\pi*k/N}, k = 0, \dots, \frac{N}{2} - 1$$

$$F(k + \frac{N}{2}) = F_1(k) - e^{-j*2\pi*k/N}, k = 0, \dots, \frac{N}{2} - 1$$

Методика вычисления основана на разбиении входного сигнала  $f(n)$  на массив четных чисел  $f_1(n)$  и нечетных -  $f_2(n)$ . Потом для каждого массива вычисляется ДПФ, и образуются функции  $F_1(k)$  и  $F_2(k)$ . На последней итерации производятся операции сложения и вычитания с умножением одного из компонентов на экспоненциальный множитель

$$W^k = e^{-j*2\pi*k/N}$$

По итогу выполнения данных действий мы получим функцию  $F(k)$ .

При разбиении данного сигнала на два для получения функции  $F(k)$ , определенной на частотном интервале, понадобится примерно  $2 \cdot (N/2)^2 = N^2/2$  комплексных умножений, что в 2 раза меньше, по отношению к прямому методу вычисления.

Для решения задачи обработки сейсмоакустической информации считается более рациональным использовать быстрое преобразование Фурье, т. к. оно уменьшает количество операций и повышает точность.

В заключение, хотелось бы еще раз отметить, что при помощи преобразования Фурье можно обрабатывать сейсмоакустическую информацию (гармонический сигнал), накопленную в больших объемах.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Кузнецов О.Л., Чиркин И.А., Курьянов Ю.А, Рогоцкий Г.В., Дыбленко В.П. Сейсмоакустика пористых и трещиноватых геологических сред. Экспериментальные исследования. – М.: Государственный научный центр Российской Федерации – ВНИИгеосистем, 2004. – 320 с.
2. Калинин А.В., Калинин В.В., Пивоваров Б.Л. Сейсмоакустические исследования на акваториях. – Недра, Москва, 1983 г., 1983 стр., УДК: 850.334
3. Соллогуб В.Б., Чекунов А.В. Строение земной коры и верхней мантии по данным сейсмических исследований. - Издательство "Наукова Думка", Киев, 1977 г., 321 с.

УДК 004.94

### И.В. КУЛАКОВА

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### КОНСТРУКТИВНАЯ БЛОЧНАЯ ГЕОМЕТРИЯ ПРИ ПОСТРОЕНИИ ПРИМИТИВОВ В ПРОГРАММЕ FREESAD

*Рассматривается использование конструктивной блочной геометрии во время построения примитивов в программе параметрического трёхмерного моделирования FreeCAD. Также рассматриваются базовые понятия и связь между данной геометрией и примитивами.*

Конструктивная блочная (твёрдотельная) геометрия (или же сокращенно КБГ) является технологией, которую применяют в моделировании твёрдых тел. Также она является парадигмой моделирования во многих традиционных системах автоматизированного

проектирования (САПР). Конструктивная блочная геометрия заключается в применении примитивных твёрдых объектов и выполнении с ними логических (булевых) операций для создания окончательной формы. К таким операциям относятся пересечение, вычитание (разность) и слияние (объединение). [1]

КБГ довольно часто называют способ моделирования в трехмерной графике и САПР, позволяющий делать сложную сцену/объект через битовые операции, чтобы комбинировать несколько иных объектов, из-за чего можно проще математически описывать сложные объекты, несмотря на то, что операции не всегда проходят с применением только простых тел. Поэтому посредством КБГ часто представляют модели/поверхности, которые визуально выглядят сложными, хотя на самом деле являются скомбинированными или декомбинированными простыми объектами. Иногда КБГ выполняется при помощи полигональных сеток, и может быть процедурной и/или параметрической. [2]

В КБГ используются простейшие тела, то есть тела, которые имеют простую форму и называются примитивами. К таким телам относятся цилиндр, куб, конус, сфера, пирамида и т.д. Иными словами, примитивами в конструктивной блочной геометрии являются простые геометрические фигуры, которые система может создавать, обрабатывать или хранить. Иногда примитивами называют подпрограммы, создающие подобные объекты. Наиболее простыми примитивами являются точка и отрезок прямой линии.

Набор доступных примитивов зависит от программного пакета. Поэтому одни программы позволяют создавать КБГ на базе объектов, а другие нет. [2]

При построении более сложных объектов применяют к их описаниям булевы (или же двоичные) операции на множествах, такие как объединение, разность и пересечение. С помощью таких операций примитивы могут быть скомпонованы в составные объекты, что представлено на рисунке 1.

Как видно на рисунке, объединением является слияние 2-х объектов (сферы и куба) в один, под разностью понимается вычитание одного объекта от другого, а пересечение выделяет общую часть обоих объектов. [2]

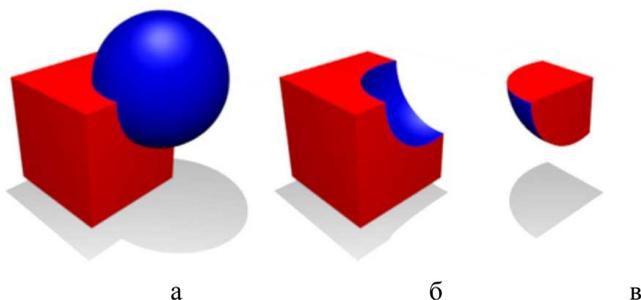


Рисунок 1 – Булевы (двоичные) операции на КБГ:  
а – булево объединение, б – булева разность,  
в – булево пересечение

Примитив может быть описан процедурой, которая принимает некоторые значения параметров. Например, чтобы создать сферу достаточно знать её радиус и положение центра.

КБГ широко распространена, имеет ряд практических применений и используется в случаях, когда нужна простота или математическая точность, например, для определения водонепроницаемости конструкции.

Конструктивная блочная геометрия применяется во многих программах. Например, на её применении основана технология Geo Mod, а также она используется в таких программах, как BRL-CAD, Feature Manipulation Engine, GtkRadiant, POV-Ray, sgCore, UnrealEd, 3D World Studio, Vectorworks, OpenSCAD и т.д. [2]

Также конструктивная блочная геометрия применяется и в наиболее широко используемых программах, например, SolidWorks и FreeCAD. Рассмотрим использование конструктивной блочной геометрии при построении примитивов в программе FreeCAD.

FreeCAD – это параметрическая система автоматизированного проектирования общего назначения с открытым исходным кодом, которая может использоваться для проектирования трёхмерных деталей.

В данной программе конструктивная блочная геометрия в большинстве случаев применяется с верстаком Part, позволяющим создавать примитивные объекты, а именно кубы, цилиндры и сферы, которые можно вместе соединять или же использовать для обрезки других объектов с помощью инструментов (например, при помощи команды «Обрезать»). [1]

Процесс работы конструктивной блочной геометрии представлен ниже на рисунке 2:

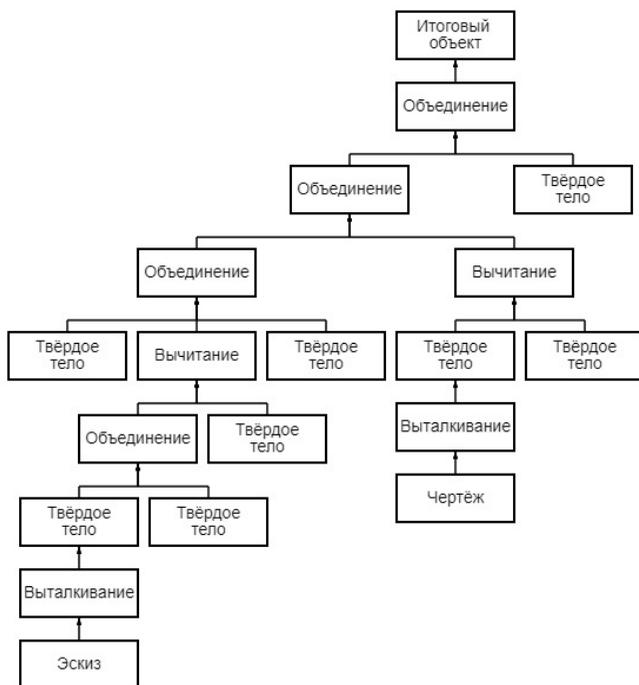


Рисунок 2 – Процесс работы конструктивной блочной геометрии

С этими примитивами можно выполнять любое число операций для создания других твёрдотельных объектов, после чего соединить/обрезать их, пока не будет получена итоговая форма. [1]

Верстак PartDesign в роли альтернативы применяет подход, называемый редактированием объектов с использованием функций редактирования, то есть создание базового твёрдого тела и добавление последовательных параметрических преобразований, чтобы получить конечное тело. Тела, созданные верстаком PartDesign, могут участвовать в логических операциях с другими объектами верстака Part. [1]

Пример того, как именно работает процесс конструктивной блочной геометрии, показан на рисунке 3:

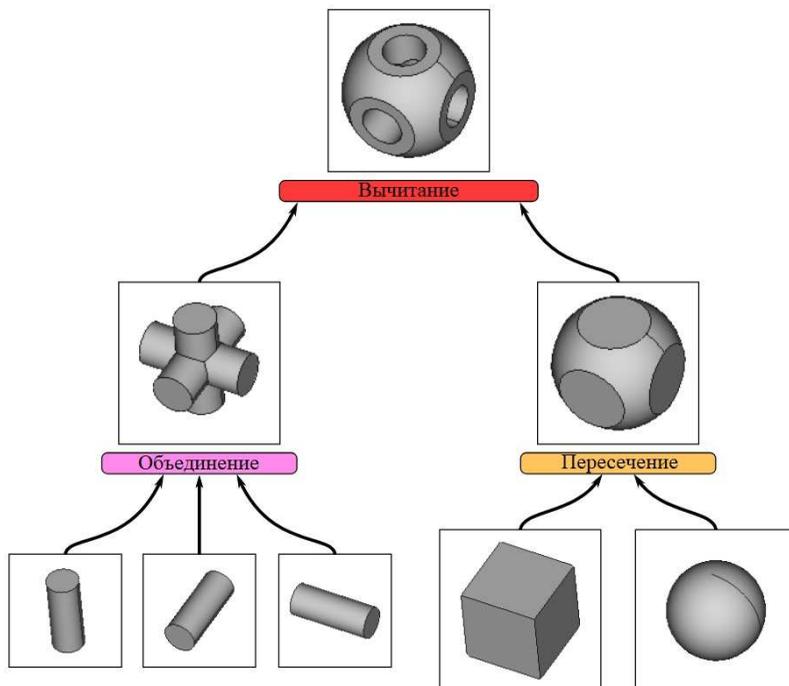


Рисунок 3 – Пример процесса работы КБГ

Как показано на этом рисунке, сложный объект может быть представлен в виде двоичного дерева, где в качестве «листьев» выступают объекты, а в качестве узлов – операции, а именно объединение, пересечение и вычитание (или же разность). [2]

Примитивные части соединяются с помощью объединения, вычисляется пересечение двух других примитивных частей и в итоге получается разность двух предыдущих фигур при помощи вычитания. [1]

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Конструктивная блочная (твёрдотельная) геометрия. [Электронный ресурс]. – Режим доступа: [https://wiki.freecadweb.org/Constructive\\_solid\\_geometry/ru](https://wiki.freecadweb.org/Constructive_solid_geometry/ru) (10.04.2022 г.)

2. Конструктивная сплошная геометрия. геометрия. [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Конструктивная\\_сплошная\\_геометрия](https://ru.wikipedia.org/wiki/Конструктивная_сплошная_геометрия) (10.04.2022 г.)

УДК 621.32

**П.С. КУЛЕВА**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**ТЕХНОЛОГИЯ ИЗГОТОВЛЕНИЯ ЦВЕТНОЙ ГАЗОРАЗРЯДНОЙ  
ИНДИКАТОРНОЙ ПАНЕЛИ**

*Рассматривается подробный технологический процесс создания газоразрядной индикаторной панели. Особое внимание уделяется освещению технологии изготовления трафаретных печатных форм.*

Газоразрядная индикаторная панель (далее ГИП) представляет собой ионный прибор для отображения информации, использующий тлеющий газовый разряд.

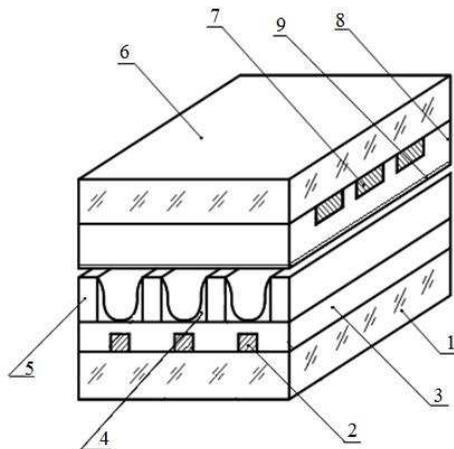


Рисунок 1 – Конструкция ячеек ГИП

Где 1 – нижняя пластина; 2 – электрод управления; 3 – диэлектрическое покрытие; 4 – люминофорное покрытие; 5 – диэлектрический барьер; 6 – верхняя пластина; 7 – электрод индикации; 8 – диэлектрическое покрытие; 9 – защитное покрытие.

Разработанная ГИП представляет собой две пластины: верхнюю и нижнюю, на которых сформированы конструктивные элементы. На верхней пластине располагаются электроды индикации, диэлектрическое покрытие и защитный слой, на нижней пластине - электроды

управления, диэлектрическое покрытие, диэлектрические барьеры и люминофорное покрытие.

Основные конструктивные элементы ГИП переменного тока изготавливаются методом трафаретной печати (кроме двух конструктивных элементов – эмиссионное покрытие наносится вакуумным напылением, шов герметизации - методом полива). Трафаретная печать является базовой технологией изготовления пластин электродных систем.

Суть метода трафаретной печати заключается в выдавливании перемещающимся по трафарету rakelом вязкого материала (пасты) через открытые участки трафарета для нанесения рисунка с заданной топологией на нижележащую подложку.

Рассмотрим более подробно технологию изготовления трафаретных печатных форм.

### ***Технология изготовления трафаретных печатных форм***

Первым в схеме изготовления ТПФ технологическим процессом, от выполнения которого зависят дальнейшие технические характеристики ТПФ, является процесс натяжения сетки на раму.

Натяжение сетки – электромеханическое. Так как в процессе натяжения и приклеивания сетки к раме не происходит автоматической корректировки заданных параметров, то в процессе выполнения указанной операции необходимо производить корректирующие процедуры с целью получения заданного натяжения, контролируемого с помощью тензометра.

На предварительно очищенную и обезжиренную ацетоном и чистящей пастой раму натягивается под углом  $22,5^\circ$  металлическая сетка и закрепляется с помощью двухкомпонентного клея.

После приклеивания сетки к раме и удаления излишков сетки, рама с сеткой хранится при температуре  $(21 \pm 2)^\circ\text{C}$  в течение 24 ч. За это время происходит уравнивание натяжения сетки по всему полю рамы после снятия подаваемой на неё нагрузки.

Следующей операцией является нанесение на сетку светочувствительной эмульсии (далее эмульсия). Эмульсию наносят вручную с помощью rakel-кюветы. С целью получения минимального разброса толщины эмульсии по всему полю (1,5-2,5 мкм), для каждого рисунка используют определённую конструкцию rakel-кюветы с целью исключения образования пузырей в объёме эмульсионного слоя, подбирают количество наносимых слоёв, количество интервалов и температуру сушки. При этом при каждом нанесении эмульсии осуществляется переворот рамы на  $180^\circ$  с целью достижения минимального разброса толщины слоя эмульсии.

Экспонирование эмульсии осуществляется на установке экспонирования, оснащённой металлогалогидной лампой и кварцевой линзой.

Далее технически сложным процессом является проявление эмульсии в связи с отсутствием его автоматизации. Эмульсия проявляется водой, нагретой до температуры 25-35°C с заданным давлением.

После проявления ТПФ подвергается сушке в установке сушки в течение 30-60 мин. для удаления остаточной влаги и эмульсии. Для обеспечения репродукционно-эксплуатационных свойств ТПФ подвергается дополнительному УФ облучению (экспонированию-задубливанию) в установке экспонирования.

После экспонирования ТПФ выдерживается при комнатной температуре в течение суток, с целью восстановления (релаксации) первоначальных свойств конструктивных материалов ТПФ, и затем передаётся на контроль качества. В результате изготовлены ТПФ, которые можно использовать для формирования конструктивных элементов пластин электродов индикации и управления опытных образцов ГИП с информационной емкостью 640x480 триад с шагом триад 0,3 мм.

### ***Технология формирования конструктивных элементов пластин электродов индикации***

#### *Предварительная термообработка пластин*

Для обеспечения совмещения конструктивных элементов в процессе изготовления блоков электродных систем стеклоподложки предварительно отжигаются. При этом происходит стабилизация свойств стекла – увеличивается плотность, а также снижаются напряжения в стекле.

#### *Нанесение биндера*

Для уменьшения растекания на основе порошка золота и обеспечения точностных характеристик при изготовлении электродов индикации на поверхность стеклянной пластины наносится впитывающее покрытие толщиной порядка 1 мкм на основе биндера (3% раствора этилцеллюлозы в терпинеоле) с последующей сушкой в печи инфракрасной сушки.

Кроме режима нанесения биндера, важную роль для получения качественного покрытия играют следующие факторы:

1. Условия вакуумной гигиены.
2. Состояние поверхности подложки.

### *Формирование электродов индикации*

Нанесение электродов на стеклянную подложку требует от используемого материала:

- хорошей адгезии к стеклу,
- пассивности по отношению к последующим этапам технологического процесса,
- высокой электропроводности,
- малых отклонений размеров,
- малого количества пор и других дефектов.

Печать электродной системы заданной геометрии согласно КД производится двумя пастами на основе порошка золота.

### *Формирование диэлектрического покрытия*

Одним из способов получения диэлектрических покрытий является нанесение пасты, которая далее высушивается и нагревается до получения однородной стекловидной диэлектрической плёнки. Температура затвердевания этого стекла должна быть ниже температуры размягчения подложки и проводящих элементов. Состав стекла должен исключать взаимодействие с проводником во время термообработки.

### *Технология формирования защитного эмиссионного покрытия*

Для формирования защитного эмиссионного покрытия используется метод напыления тонкой плёнки MgO – электронно-лучевое испарение (ЭЛИ), суть которого заключается в физическом распылении материала мишени MgO за счёт его бомбардировки ускоренным в электромагнитном поле и сфокусированным на поверхности мишени потоком электронов.

### ***Технология формирования конструктивных элементов пластин электродов управления.***

Предварительная термообработка пластин и нанесение подслоя под электроды производятся аналогично пластине индикации.

### *Формирование электродов управления*

Это самая важная операция для формирования всего печатного блока. В процессе работы были уточнены требования к ТПФ, оптимизированы режимы печати электродов, что обеспечило не только заданную ширину электродов и их суммарный шаг, но и позволило печатать все последующие конструктивные элементы (защитные покрытия, диэлектрические барьеры) с требуемым совмещением и качеством самого элемента. Особенно важно это для барьеров.

### *Формирование диэлектрических барьеров*

Перед формированием диэлектрических барьеров на электродах управления формируется диэлектрическое покрытие пастой на основе

порошков С82-3 и  $Al_2O_3$ , с последующей их сушкой.

Так как одна из функций барьеров – электрическая изоляция, в качестве их материала применяются диэлектрические пасты. Материал барьеров должен быть хорошо согласован с диэлектриком, закрывающим электроды управления. Поэтому в его состав вводят те же порошки стекол: С82-3 и алунд  $Al_2O_3$ . Для сохранения целостности диэлектрических барьеров они вжигаются одновременно с вжиганием диэлектрика.

Формирование диэлектрических барьеров это самая сложная и трудоёмкая операция при изготовлении блока управления. Режим печати барьеров должен обеспечить:

1 Совмещение между электродами и барьерами. Исходя из этого, режим печати барьеров выбирается близким к режиму печати электродов.

2 Формирование качественного барьера по всей площади пластин — обеспечение заданных в конструкторской документации ширины барьеров и их разбросов.

#### *Формирование люминофорных покрытий*

Нанесение люминофорного покрытия методом базовой технологии трафаретной печати осуществляется через один трафарет.

Шаг ячейки предъявляет очень жёсткие требования по распределению люминофора на стенках барьеров и дну ячейки – покрытие должно быть равномерным по толщине, без проколов и перемычек в виде люминофорных «мостиков» между барьерами.

Режим нанесения должен обеспечить равномерное заполнение ячеек по всей площади пластины. При этом не допускается затекания пасты в соседние люминофорные ячейки. Равномерность заполнения зависит в основном от угла наклона ракеля и качественного отрыва трафарета.

#### *Технология монтажно-сборочных операций*

На операцию сборки поступают две пластины: пластина электродов индикации и пластина электродов управления. При сборке обеспечивается взаимно-перпендикулярное расположение электродов индикации и электродов управления и совмещение их рабочих полей.

После штенгелевки ГИП производится герметизация с одновременным проведением откачки. Во время проведения откачки проводится очистка поверхности газовым разрядом, что позволяет обеспечить стабильность электрических параметров в процессе работы и снизить время тренировки после откачки. После тренировки проводится измерение электрических параметров. Затем к ГИП при-

паваются ленточные выводы. Место припайки защищается компаундом.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Левина Н.Н. Разработка базовой технологии создания особо плоских высокоинформативных плазменных панелей нового поколения для отображения тепловизионной информации // Научно-технический отчет. – 2013.

УДК 004.046

**А.Ю. МЕДВЕДЕВА**

МИРЭА – Российский технологический университет

### ИСПОЛЬЗОВАНИЕ CRM В РОЗНИЧНОЙ АПТЕЧНОЙ СЕТИ

*Рассматриваются особенности использования CRM (систем взаимодействия с клиентами) в аптечной сети при различном количестве обособленных подразделений и, соответственно, с различным денежным оборотом. Также особое внимание в данной статье уделено значимости CRM в фармацевтической среде.*

В общепринятом понимании CRM представляет собой некий инструмент сбора, хранения и анализа данных о разовых и постоянных клиентах, партнёрах, поставщиках и товарах, реализуемых той или иной компанией, то есть CRM является платформой управления взаимоотношениями с клиентами. Следует заметить, что существуют известные широкому кругу лиц стандартные версии CRM для розничного бизнеса (и в частности, для розничной аптечной сети), которые характеризуются следующими функциональными характеристиками и особенностями:

- типовой и достаточно простой интерфейс, который понятен каждому пользователю с минимальными техническими и специальными дополнительными навыками;
- возможность добавления новых вспомогательных полей;
- работа на основе браузера.

Использование типового и достаточно простого (минималистичного) интерфейса подразумевает, что на экране представлены по большей части только основные важные поля и функции, которые необходимы для работы продавца или фармацевта, если рассматривать аптечные сети или какие-либо фармацевтические компании. Графиче-

ский дизайн ограничивается элементарным цветовым разделением полей, изредка простыми контекстными картинками [1].

Особое внимание следует обратить на достаточно простую возможность создания дополнительных полей в CRM. Специальный конструктор позволяет генерировать новые таблицы баз данных, шаблоны интерфейсов и совершенствовать старые наработки. Такой конструктор очень удобен в работе и при этом не требует особых знаний и специальных навыков от специалистов IT-отдела, и самих пользователей. Стоит подчеркнуть, что это в свою очередь позволяет компаниям экономить на заработной плате IT-сотрудникам.

Стандартные версии CRM работают на основе браузера и, соответственно, не нуждаются в интеграции системы с внутренней компьютерной системой аптеки. Поэтому в большинстве случаев компаниям, использующим CRM для работы в данных системах, необходимы только компьютеры с выходом в глобальную сеть Интернет.

CRM для бизнеса стандартной конфигурации в фармацевтической отрасли способствует эффективно решать следующие задачи:

- наладить выгрузку интернет-заявок и документов в единую базу;
- пользоваться преимуществами электронных рассылок и смс-уведомлений;
- назначать конкретные задачи для каждого пользователя системы и персонала, внесенного в систему;
- организовывать различные скидки, акции, активно вести промо-деятельность;
- формировать информационное пространство для взаимодействия всех обособленных подразделений компании, складов и главного офиса;
- собирать информацию о клиентах, своевременно вносить изменения в данные сведения согласно запросу клиентов из-за обновления тех или иных сведений;
- контролировать актуальность информации и данных о всех клиентах, поставщиках, счетах, партнерах, заявках и т.д.;
- систематически обновлять список товаров, которые есть в наличии на складе на текущий момент;
- создавать отчеты по разным направлениям работы (продажи, маркетинг) и анализировать эффективность работы конкретного продавца [1].

CRM позволяет мгновенно обрабатывать информацию о клиентах, оперативно обновлять её, внося соответствующие актуальные корректировки, и исключает ошибки при выгрузке информации из

системы в том случае, если CRM спроектирована верно. На базе CRM достаточно нетрудно реализовать программу лояльности для постоянных клиентов и всех сотрудников аптечной сети.

Конечно, CRM со стандартным набором функций не всегда удовлетворяет все запросы от компаний, специализирующихся на розничной реализации аптечных медикаментов и медицинских изделий. Поэтому для аптечной сети в большинстве случаев следует подбирать продукт с персональными настройками. Это также относится и к другим организациям, которые реализуют ту или иную категорию товаров. Прежде всего это связано с тем, что CRM в первую очередь используется в крупных организациях для управления взаимоотношениями с клиентами.

CRM с персональной конфигурацией – это программа, настроенная под запросы конкретной розничной и/или оптовой аптечной сети. В данных системах автоматически учитываются вся специфика фармацевтической отрасли:

- проверка качества медицинских изделий и лекарственных препаратов по серийным номерам;
- отслеживание сроков годности медицинских изделий и медикаментов, отпускаемых физическим лицам без рецепта и по рецепту;
- специфический партийный учет товаров [1];
- особенности документооборота в аптечной сети;
- особенности осуществления ценообразования, установления системы скидок, баллов, бонусов, в зависимости от региона и конкретного обособленного подразделения, осуществляющего реализацию аптечных медикаментов, медицинских изделий и оборудования.

Персонализированная CRM для аптек имеет расширенный функционал и позволяет эффективно работать в таких направлениях фармацевтического бизнеса, как: генерирование коммерческих предложений, ведение учета расходов, работа с аптечным складом, использование единой информационной базы, работа с отчетностью, персонализация коммуникаций.

Существует множество различных компаний, которые работают в направлении разработки и доработки персональных конфигураций CRM для бизнеса под запросы конкретной аптечной сети или компании из иной сферы бизнеса. Главное – настроить интеграцию CRM со всеми каналами коммуникации, используемыми той или иной аптечной сетью, для достижения максимальной эффективности применения системы, так как это будет способствовать своевременному и оперативному контакту со всеми покупателями и поставщиками по интегрированным с CRM каналам коммуникаций.

Как известно, большинство CRM создаются для отдела продаж – менеджеров по продажам и маркетологов. Такие CRM содержат подробную базу клиентов со всеми сведениями: контактами, историей заказов. Покупателей можно сегментировать по категориям, рассылать им напоминания, анализировать их прохождение по воронке продаж, мониторить заключаемые сделки на всех этапах. Если при внедрении CRM-системы интегрировать её с внешними сервисами, функционал станет ещё шире: можно будет писать клиентам письма и звонить с помощью справочника контактов, находящегося в единой базе CRM.

На практике интегрированная система CRM обеспечивает координацию действий различных отделов, обеспечивая их общей платформой для взаимодействия с клиентами. С этой точки зрения назначение CRM – исправить ситуацию, когда отделы маркетинга, продаж и сервиса действуют независимо друг от друга, причем их видение заказчика часто не совпадает, а действия не согласованы.

С позиции управления бизнесом эффект от внедрения CRM проявляется в том, что процесс принятия решения за счет автоматизации переносится на более низкий уровень и унифицируется. Из-за этого повышается скорость реакции на запросы, растет скорость оборота средств и снижаются издержки. Наконец, CRM включает в себя идеологию и технологии создания истории взаимоотношений клиента и фирмы, что позволяет более четко планировать бизнес и повышать его устойчивость.

Компании, использующие CRM-системы, собирают все заявки с сайта в едином автоматизированном хранилище и не теряют их благодаря функции напоминаний. Там же содержится вся хронология покупок и обращений потребителей, и менеджеры могут делать повторные продажи. Следовательно, данным способом компания повышает общую прибыль с каждого клиента.

С точки зрения программной реализации CRM для розничных аптечных сетей достаточно удобно применять такую методологию разработки, как Agile Kanban. Это связано в первую очередь с тем, что для реализации какого-либо программного модуля CRM, автоматизирующего ключевые бизнес-процессы аптечной сети, необходимо структурировать последовательность шагов разработки. Методология Agile Kanban прекрасно подходит для разработки программных модулей CRM для аптечной сети и всей CRM в целом вне зависимости от количества обособленных подразделений аптеки и количества сотрудников, которые в дальнейшем будут работать в данной системе. Также данная методология подходит для разработки не только для аптечных

сетей, но и для других сфер бизнеса, таких как интернет-магазины и финансовые организации.

В целом Agile Kanban находит свое применение в современных развивающихся организациях, которые обладают достаточно большим штатом сотрудников. Следует заметить, что данная методология применима не только для IT отделов какой-либо организации, но и для отделов маркетинга и менеджмента. То есть Agile Kanban применяется в современных развивающихся отделах различных компаний, в которых постоянно что-нибудь нужно модернизировать и совершенствовать, внося какие-либо изменения и дополнения.

Важно отметить, что если на западном рынке количество CRM-систем измеряется сотнями, то в Российской Федерации в настоящее время представлен довольно узкий спектр решений, в основном крупных поставщиков, давно предлагающих свои ERP-системы. В то же время, количество отечественных разработчиков CRM в последние годы увеличивается почти что каждый месяц. Таким образом, перед российскими аптечными сетями уже стоит проблемы выбора из более чем двух десятков систем. При этом всегда стоит помнить, что всегда можно заказать индивидуальную CRM, которая будет создана с учетом всех особенностей той или иной компании. Конечно, данные решения между собой кардинально различаются, так как нацелены на разные сегменты рынка и имеют различные финансовые вложения для реализации, внедрения и эксплуатации.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. CRM for retail: выбираем CRM-систему для аптеки. Официальный сайт компании Manzana Group [Электронный ресурс]. – Режим доступа: <https://manzanagroup.ru/information/crm-for-retail-vybiraem-crm-sistemu-dlya-apteki/> (Дата обращения: 01.04.2022)

2. Оливер Гранд. Как увеличить и стабилизировать продажи за счет CRM? // Издательские решения. 2019 [Электронный ресурс]. – Режим доступа: <https://mybook.ru/author/oliver-grand/kak-uvlichit-istabilizirovat-prodazhi-zaschetcrm/reader/> (Дата обращения: 01.04.2022)

3. Лагунова А. Д. Информационные системы управления взаимоотношениями с клиентами: Практикум // МИРЭА – Российский технологический университет. 2020 [Электронный ресурс]. – Режим доступа: <https://e.lanbook.com/book/163876> (Дата обращения: 02.04.2022)

УДК 621.375.4

**В.А. МЕНЬШИНА**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**ПЕРСПЕКТИВЫ РАЗВИТИЯ УСИЛИТЕЛЕЙ МОЩНОСТИ  
СВЧ ДИАПАЗОНА ДЛЯ РАДИОСИСТЕМ  
ПЕРЕДАЧИ ИНФОРМАЦИИ**

*В статье приводится сравнение основных видов СВЧ-усилителей, используемых в современной электронике и радиосистемах, и их эффективность в решении поставленных задач.*

На сегодняшний день СВЧ-электроника является одной из основных ветвей развития, охватывающей всю индустрию электроники в целом. Возможности современных микроэлектронных и СВЧ-технологий дают толчок к развитию новых прикладных направлений производства, таких как индустрия телекоммуникаций, медицинские инженерные технологии, легкая и тяжелая промышленность, автомобилестроение, судостроение и космические технологии. Чтобы применение СВЧ-технологий оправдывалось с экономической точки зрения, наиболее приоритетным будет выбор СВЧ-приборов, которые имели бы такие показатели качества, как высокий КПД преобразования промышленной энергии в собственную СВЧ-энергию (в идеальных условиях 70-90%); высокая выходная мощность в непрерывном режиме (от 1 кВт); примитивные и доступные по цене источники питания (предпочтительнее всего напрямую подключать СВЧ-прибор к вторичной обмотке силового трансформатора промышленной электросети, не задействовав при этом выпрямители и фильтры); простота конструкции, высокий уровень надежности, длительный срок эксплуатации (не менее 2000-5000 часов); обеспечение продуктивной работы в условиях переменной нагрузки. Рассмотрим самые распространенные виды СВЧ-усилителей, используемые в современной электронике и отвечающие основным техническим требованиям.

**СВЧ-усилитель на твердотельной элементной базе.**

Наибольшее распространение в СВЧ-технике на данный момент получили твердотельные усилители на СВЧ транзисторах. Основными преимуществами таких усилителей являются их компактность и малый вес (что послужило развитию корпусных микросборок электронных устройств), возможность массового производства, ведущая к сниже-

нию удельной стоимости, большой показатель надёжности, высокий уровень прочности сборки, а также длительный срок службы — некоторые устройства могут находиться в использовании до 50 лет при должном соблюдении правил эксплуатации. Традиционным и наиболее приоритетным материалом для их изготовления является пришедший на смену кремнию арсенид галлия, благодаря низкому уровню вносимых шумов на СВЧ частотах. Применение твердотельных усилителей позволяет создавать усилительные элементы со сверхоктавной полосой и выходной мощностью более 100 Вт при КПД 40-60% в дециметровом диапазоне, и более 20-30 Вт при КПД 20-30% в сантиметровом. А к недостаткам транзисторных СВЧ-усилителей можно отнести ограничения по мощности, и по рабочим температурам, поскольку с увеличением температуры будет уменьшаться средняя мощность.

### **СВЧ-усилитель на базе лучевых приборов.**

Широко используемыми для усиления мощности СВЧ-приборами с лучевой структурой являются лампы бегущей волны, магнетроны и клистроны. Поскольку импульсные лампы бегущей волны могут работать на разных частотах и диапазонах, они обеспечивают необходимую мощность и стабильную работу прибора в широкой полосе частот и таким образом достигают высокого коэффициента усиления в пределах от 30 до 50 дБ. В основном они используются для усиления слабого сигнала, находящегося на уровне шумов самого усилителя. Принцип работы лампы бегущей волны заключается в том, что микроволновый сигнал, подаваемый на вход лампы, будет возбуждать в спирали прямую волну, а воздействие электронной волны начнет модулировать электронный поток. Одним из главных параметров для успешной работы лампы бегущей волны является его устойчивость. Но если такой усилитель станет неустойчивым, то, из-за наличия обратной связи, он начнет работать аналогично генератору и вместо усиления заданного сигнала начнет дублировать свой же сигнал со своей собственной частотой и посторонними параметрами. При этом усиление основного сигнала, поступающего на вход усилителя, прервется. Таким образом, чем выше будет коэффициент усиления усилителя, тем больше вреда ему нанесет обратная связь. В связи с этой неустойчивостью тяжело получить высокий коэффициент усиления, по причине чего усилитель на лампе бегущей волны уступает ряду других СВЧ-приборов. Поскольку для создания более мощного усилителя необходимо большое количество транзисторов, так как общая мощность складывается из мощностей каждого из них, это вынуждает применять дополнительные меры подавления отраженных волн. Например, включать в устройство специальные ферритовые вентили, уменьшаю-

щие усиление каскада, а вместе с этим дополнительные обратные связи, отрицательно влияющие на усиление, вследствие чего ухудшится выходная мощность и КПД. В связи с этим можно сделать вывод, что усилители на базе ЛБВ являются наименее перспективными среди рассматриваемых классов СВЧ-усилителей.

### **СВЧ-усилитель на базе приборов М-типа.**

При выборе СВЧ-прибора, как правило, опираются на условия, в которых он будет эксплуатироваться. Очень часто при выборе наиболее подходящего типа прибора за основное берется одно предъявляемое условиями работы усилителя требование, при этом все остальные требования отодвигаются в сторону. Усилитель М-типа характеризуется невысокими или умеренными значениями коэффициента усиления, высоким КПД, режимом отсечки в области насыщения, умеренными значениями полосы пропускания, компактными габаритами и небольшим весом. При этом попытка улучшить любую из перечисленных характеристик прибора повлечет за собой ущерб другим, что нарушает общее соответствие требованиям по всей совокупности параметров, предъявляемых устройству. Поэтому использование усилителей М-типа является наиболее результативным только тогда, когда их параметры в полном объеме отвечают назначению системы. Если рассматривать практическое применение, то усилитель М-типа чаще идет в дополнение к другим классам приборов, чем заменяет их. Обычно его применяют в оконченных каскадах когерентных усилительных цепочках, использующихся в когерентных системах связи оптических сигналов, где необходим высокий КПД и разнообразие рабочих режимов. А в предварительных усилителях мощности на входе оконечного каскада передатчика с более низким уровнем мощности, предпочтительнее будет использовать приборы с более высокими коэффициентами усиления пониженной мощностью, как например приборы с линейным пучком, используемые в фотоэлектронных приборах.

Твердотельные усилители мощности на базе СВЧ-транзисторов и приборы М-типа являются одними из самых распространенных классов электронных СВЧ-устройств, показав наибольшую эффективность в военной промышленности, используясь на Земле, в авиации, ракетостроении, и в космосе. Они имеют преимущества по сравнению с ЛБВ по надежности на большом сроке эксплуатации, стоимости, габаритам и массе, а также обеспечивают увеличение скорости передачи информации в несколько раз в зависимости от емкости и формата, а их использование совместно с цифровой обработкой сигналов предоставляет новые возможности для исследований в данной области. Таким образом, наиболее многообещающими направлениями в области теле-

коммуникаций и СПИ является создание твердотельных галлиевых усилителей на базе СВЧ-транзисторов и приборов М-типа, позволяющих в диапазоне микроволн сочетать коэффициент усиления 30-40 дБ, КПД 35-40%, и ширину обеспечивать рабочей полосы частот до 10-12%.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Перспективы развития СВЧ – усилителей [Электронный ресурс]. – Режим доступа: <https://cyberleninka.ru/article/n/perspektivy-gazvitiya-svch-usiliteley> (Дата обращения 29.03.2022).

2. Кишинский А.А. Сверхширокополосные твердотельные усилители мощности СВЧ диапазона: схемотехника, конструкции, технологии. Электроника и микроэлектроника СВЧ. Сборник статей VII Всероссийской конференции. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2018, С. 4-13.

3. Всероссийская научно-техническая конференция "Электроника и микроэлектроника СВЧ". Сборник докладов. Санкт-Петербург. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2021, 629 с.

УДК 628.8.02

**В.А. МИХЕЕВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПАРАМЕТРЫ, ОПРЕДЕЛЯЮЩИЕ ОПТИМАЛЬНЫЙ МИКРОКЛИМАТ ТЕПЛИЦЫ**

*Рассматриваются параметры, влияющие и определяющие микроклимат в теплице. Приводятся их оптимальные значения и способы их поддержания, а также описываются измерительные приборы для замеров значений*

Микроклимат – это комплекс метеорологических условий, которые необходимо обеспечить для комфортных условий проведения тех или иных работ в каком-либо помещении. В данном случае рассматривается помещение для выращивания сельскохозяйственных культур – теплица. В ней необходимо обеспечить особый микроклимат, который характеризуется следующими основными параметрами: температура, определенная газовая среда периодичность ее обновления, скорость движения воздуха, относительная влажностью и световой режим.

Для определения влияния внешних условий на микроклимат был сформирован ряд тезисов, которые раскрывают суть заявленной темы:

- какие факторы, относящиеся к климатическому комплексу, оказывают наибольшее влияние на рост той или иной культуры;
- каким образом можно создать и поддерживать требуемый микроклимат;
- с помощью каких электронных средств можно управлять микроклиматом в теплице, учитывая воздействие климатических факторов на работу этих устройств.

#### **Компоненты микроклимата теплицы**

Микроклимат зависит от целого ряда условий: от времени года, условия отопления и вентиляции. Все факторы, влияющие на микроклимат, разделяются на две группы:

1. Нерегулируемые, то есть зависят от условий климата в выбранной местности.
2. Регулируемые, которые зависят от площади и конструкции помещения, в нашем случае – теплицы, материалов, из которых изготовлена теплица [1].

Для растений, как для живых организмов, основными факторами микроклимата будут: температура воздуха, относительная влажность и скорость движения воздуха, а также интенсивность теплового и ультрафиолетового излучений. Необходимо рассмотреть конкретное влияние этих параметров на рост и развитие растений в теплице.

Температура влияет на растения, изменяет скорость роста и развития, поглощения, усвоения и передвижения воды и элементов минерального питания и синтеза органических соединений. Материал теплицы также оказывает влияние на температуру. Пленочные теплицы сильнее подвержены колебаниям температуры из-за слишком хорошего теплообмена, что нежелательно. Менее подвержены колебаниям температуры теплицы остекленные, так как стекло задерживает инфракрасное излучение внутри при надежной герметизации сооружения.

В организме растений вода является основной средой, в которой протекает обмен веществ. Для нормальной жизнедеятельности растения его клетки и ткани должны быть достаточно насыщены водой, особенно в тот период, когда в них активизируются жизненные процессы. Именно в это время содержание воды в клетках и тканях деятельных частей растений достигает 70—90 % [2].

Влажность воздуха поддерживается поливами растений, разбрызгиванием воды по конструкциям, испарением с поверхности грунта и листьев растений. Температура и влажность воздуха теплицы в

существенной степени поддерживаются режимами вентиляции (проектирования).

Свет — основной фактор формирования микроклимата в теплице. Количество и качество света изменяет необходимые параметры всех остальных факторов. Свет является источником энергии для растений. С его помощью растения образуют органическое вещество и регулируют все физиологические и биохимические процессы. Недостаток света снижает действие других факторов, и урожай снижается.

Количество и качество солнечной радиации оценивается по ее интенсивности, продолжительности, спектральному составу. Радиация бывает и рассеянной (отраженной).

Для растений наибольшее значение имеет физиологически активная радиация (ФАР) — коротковолновое излучение в диапазоне 380—710 нанометров (нм), которая является основной энергией для процесса фотосинтеза.

Интенсивность ФАР характеризуется количеством лучистой энергии на единицу площади в единицах времени и выражается в  $\text{кал}/\text{см}^2$ . ФАР в составе солнечной радиации составляет около 50%. Растения используют около 25% энергии инфракрасных лучей и более 80% видимой активной части солнечного спектра, а остальная отражается или пропускается поверхностью листьев [3].

#### **Средства автоматизации измерения параметров микроклимата теплицы**

Система управления микроклиматом состоит из: форточек, системы вентиляции, экранов, досветки, СИОД (системы испарительного охлаждения и доувлажнения), системы отопления, системы подачи  $\text{CO}_2$ . Экраны бывают двух видов: энергетические и затеняющие. Вентиляторы используют для создания равномерного температурного поля в теплице. СИОД способствует поддержанию оптимальной влажности растений и применяется при дефиците влажности в теплице. Также в теплице должны быть датчики контроля температуры, влажности, освещенности, рН — это 4 параметра, которые всегда должны контролироваться. Управляется система с помощью климатического компьютера, оснащенного различными датчиками (термометрами).

Датчики, которые могут использоваться в разрабатываемой системе контроля для измерения одного из двух важнейших показателей микроклимата - температуры: термометры сопротивления, термисторы и полупроводниковые датчики.

Термометр сопротивления — это миниатюрная катушка из медного или платинового провода, упакованная в защитный кожух. Принцип его действия основан на том, что при увеличении температуры сопро-

тивление металлического провода растет. Для металлов в большом интервале температур можно считать, что сопротивление линейно растет с ростом температуры[4]. Термопары – это генераторные датчики температуры, представляющие собой два проводника различных материалов, спаянные с одного конца друг с другом. Основным преимуществом термопар является широкий, зависящий от температуры плавления, диапазон измерения. Термопары изготавливаются из различных материалов и, следовательно, имеют различный диапазон рабочих температур. Принцип работы термопары – возникновение тока на спае разных металлов при их разной температуре. Измеряя силу тока в этой цепи можно рассчитать разность температур. Так же необходимо точно знать температуру холодного спае для расчета искомой температуры. Для этого холодную часть термопары помещают в термостат с постоянной известной температурой, что уменьшает погрешность, возникающую при измерении из-за подключения измерительной системы к разным материалам[5].

Так же существуют цифровые датчики температуры, имеющие в основе терморезистор. Измеренные значения кодируются с помощью АЦП в понятный для человека результат. Кодирование ускоряет процесс измерения и позволяет хранить результаты в устройстве.

Вторым важнейшим показателем микроклимата является влажность. Для ее измерения используются приборы, называемые гигрометрами. Чувствительный элемент гигрометра должен избирательно реагировать на изменение концентрации воды. Его реакцией может быть изменение внутренних свойств. Датчики для измерения влажности и температуры точки росы бывают емкостными, электропроводными, вибрационными и оптическими[4].

Оптические газовые датчики определяют точку росы, в то время как оптические гигрометры измеряют содержание воды в органических растворах по поглощению излучения ближнего ИК диапазона в интервале 1,9 ... 2,7 мк.

Емкостные датчики – это конденсаторы с воздушным зазором могут использоваться как датчики относительной влажности, поскольку от количества водяных паров в атмосфере зависит диэлектрическая проницаемость воздуха. Сложность заключается в измерении емкости, так как в отличие от сопротивления для точного замера емкости нужно дорогостоящее оборудование (RLC-метры). Более простой способ измерения – замер скорости заряда и разряда цепи.

Датчики влажности с выходом по напряжению – приборы, показывающие аналоговый сигнал – зависимость напряжения от влажно-

сти. Для точного измерения необходимо учитывать температуру окружающей среды, то есть калибровать датчик.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Понятия «производственный микроклимат» и его типы [Электронный ресурс]. – Режим доступа: <https://rosobrnauka.ru/nauki-oprigode/opredelenie-kharakteristika-faktori-i-znachenie-mikroklimata> (дата обращения 10.02.2022).
2. Роль фактора влажности в жизни растений [Электронный ресурс]. – Режим доступа: <https://www.activestudy.info/rol-faktoravlazhnosti-v-zhizni-rastenij> (дата 15.01.2022).
3. Микроклиматические факторы в теплицах и способы их регулирования [Электронный ресурс]. – Режим доступа: <https://www.activestudy.info/mikroklimaticheskie-factory-v-teplicax-i-sposoby-ix-regulirovaniya> (дата обращения 20.02.2022).
4. Дж. Фрайден Современные датчики. Справочник Москва: Техносфера, 2005. - 592 с.
5. Датчики и микроконтроллеры. Часть 2. Климат-контроль [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/259203/> (дата обращения 15.03.2022).

УДК 681.178.9

**В.А. МИХЕЕВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### СТРУКТУРА И ПРИНЦИП РАБОТЫ УСТРОЙСТВА КОНТРОЛЯ МИКРОКЛИМАТ ТЕПЛИЦЫ

*Приводится описание и структуры и принципа работы устройства контроля микроклимата теплицы.*

В данном случае, микроклимат – это комплекс метеорологических условий, которые необходимо обеспечить для комфортных условий проведения тех или иных работ в теплице. В ней необходимо обеспечить особый микроклимат, который характеризуется следующими основными параметрами: температура, определенная газовая среда периодичность ее обновления, скорость движения воздуха, относительная влажностью и световой режим.

Микроклимат зависит от целого ряда условий: от времени года, условия отопления и вентиляции. Все факторы, влияющие на микроклимат, разделяются на две группы:

1. Нерегулируемые, зависящие от условий климата в выбранной местности.
2. Регулируемые, зависящие от площади и конструкции теплицы, материалов, из которых она изготовлена[1].

Система управления микроклиматом может состоять из форточек, системы вентиляции, экранов, досветки, системы испарительного охлаждения и доувлажнения, системы отопления, системы подачи CO<sub>2</sub>. Экраны бывают двух видов: энергетические и затеняющие. Вентиляторы используют для создания равномерного температурного поля в теплице. Система испарительного охлаждения и доувлажнения способствует поддержанию оптимальной влажности растений и применяется при дефиците влажности в теплице. Также в теплице должны быть датчики контроля температуры, влажности, освещенности, pH – это 4 параметра, которые всегда должны контролироваться. Управляется система с помощью климатического компьютера, оснащенного различными датчиками (термометрами, измерителями влажности, датчиками углекислого газа и другими).

#### **Устройство контроля микроклимата.**

Разрабатываемая система – это универсальный контроллер-таймер для теплицы и других мест, где нужна автоматизация по таймеру или показателям микроклимата или другим датчикам.

В настоящее время ведение распространяется практика ведения собственного домашнего хозяйства. Это связано с рядом причин, например, пандемией, экономическим кризисом и другими. В таких условиях выгоднее и безопаснее иметь личное хозяйство[2]. Учитывая данный фактор – распространение дачных хозяйств - разрабатываемое устройство должно быть рассчитано на использование массовым потребителем. Это накладывает определенные условия и ограничения. Из вышесказанного следует ряд идей, которым разрабатываемое устройство должно удовлетворять:

1. Унифицирование комплектующих деталей и частей. Это позволит изготавливать конечные устройства и проводить ремонт в домашних условиях.
2. Удешевление производства и уменьшение конечной стоимости готового устройства, что увеличит круг потенциальных пользователей.

3. Уменьшение габаритных размеров устройства. Данное требование необходимо для возможности использования в теплицах различных размеров.

4. Простота и наглядность в управлении. Это необходимо для позитивного опыта использования.

В данной статье рассматривается устройство, которое обеспечивает контроль и поддержания микроклимата теплицы путем автоматического управления электрическими приводами, сервоприводами, системами освещения. Преимущество данного контроллера – низкая себестоимость, так как учитывается только стоимость радиоэлектронных компонентов и печатных плат, которые в данном случае имеют низкую стоимость, так как большинство из них являются достаточно простыми, но надежными радиоэлементами.

Разрабатываемая система выполняет контроль и поддержание температуры, влажности и светового режима в теплице в течение определенных временных отрезков. Контроллер имеет следующие составные части:

1. Микроконтроллер Arduino Nano на основе логической микросхеме ATmega328. Данная микросхема является главным контроллером всей системы.

2. 10 независимых каналов управления, к которым подключаются другие компоненты системы:

2.1. 7 каналов с логическим выходом, к которым подключаются реле, силовые ключи(транзисторы);

2.2. 2 канала для подключения сервоприводов;

2.3. 1 канал двустороннего управления линейным электроприводом с концевиками ограничения.

3. Бортовой датчик температуры и влажности воздуха.

4. 4 канала датчиков, к которым подключаются различные аналоговые датчики (температуры, влажности воздуха, почвы,), термисторы, цифровые датчики.

5. Модуль опорного времени с автономным питанием.

6. Большой LCD дисплей.

7. Орган управления – энкодер.

Система управления работает на основе заранее запрограммированного микроконтроллера Arduino Nano. Вся система размещается на печатной плате, что позволяет исключить проводной монтаж. Программирование микроконтроллера и выбранная элементная база позволяет выполнять следующие функции:

1. Хранение настроек в энергонезависимой памяти.

2. Аналоговые датчики не находятся под постоянным напряжением. Напряжение подается только в момент опроса, что увеличивает срок службы электрорадиоэлементов.

3. Несколько режимов работы: по таймеру, по условию измерения датчиков.

4. Подключение системы полива через сервоприводы.

5. Построение графиков измеряемых величин для контроля, статистики и улучшения климатических условий в теплице.

6. Сервисное меню, позволяющее осуществлять отдельный контроль каждого компонента системы.

7. Возможность отладки с помощью специального экрана.

Конструкция данного устройства представляет собой печатную плату, на выводах которой размещаются составные части контроллера. Размер самого габаритного элемента системы – экрана – определяет габаритные размеры платы. Оставшиеся комплектующие имеют небольшие размеры, что позволяет выполнить требование уменьшения габаритных размеров конечного устройства.

Корпус устройства выполняется с помощью 3D-печати. Это дает возможность сконфигурировать устройство как с точки зрения размещения в помещении и закреплении в нужном месте, так и с точки зрения состава устройства (выходы для различных датчиков, подключение дополнительных систем – все это требует выходов на корпусе, которые возможно добавить на этапе производства).

Радиоэлектронные компоненты, входящие в состав контроллера, нагреваются при работе. Часть системы, отвечающая за работу электропривода, греется сильнее всего, так как работает под большим напряжением, чем остальные компоненты. Для охлаждения в этой части корпуса предусмотрены дополнительные вентиляционные отверстия, что позволит устройству работать в нормальных условиях. Внешняя среда теплицы обычно характеризуется повышенной температурой воздуха. Но значения этой температуры входят в диапазон допустимых значений, при которых входящие электрорадиоизделия работают в нормальном режиме.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Понятия «производственный микроклимат» и его типы [Электронный ресурс]. – Режим доступа: <https://rosobrnauka.ru/nauki-oprigode/opredelenie-kharakteristika-faktori-i-znachenie-mikroklimate> (дата обращения 10.02.2022).

2. Земное притяжение: почему россияне возвращаются на дачи [Электронный ресурс]. – Режим доступа: <https://iz.ru/1304910/sergei->

gurianov/zemnoe-pritiazhenie-pochemu-rossiiane-vozvrashchaiutsia-nadachi (дата обращения 25.03.2022).

УДК 004.42

**В.П. МУРАНОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **МОДЕЛИРОВАНИЕ ТРАЕКТОРИИ ЧАСТИЦЫ, ПРОЛЕТАЮЩЕЙ ЧЕРЕЗ ПОЛЕ КВАДРУПОЛЬНОГО МАСС-СПЕКТРОМЕТРА**

*Рассматривается устройство квадрупольного масс-спектрографа и написание программы на языке C++ для определения траектории летящей частицы в двумерном пространстве в поле квадрупольного масс-спектрографа.*

Масс-спектрометрия является методом исследования и определения вещества, позволяющим определить концентрацию различных составляющих. Измерения в масс-спектрометрии основываются на ионизации компонентов. Это позволяет различать концентрацию определенных компонентов в веществе, на основании характеризующего их отношения массы к заряду. Если измерять интенсивность ионного потока, можно будет найти долю каждого компонента в веществе.

В масс-спектрометрии работают с ионизированным веществом (ионами молекул вещества). Методов ионизации, которые применяются в масс-спектрометрии, довольно много, среди них можно выделить:

- метод электронного удара
- метод бомбардировки быстрыми атомами
- метод поверхностной термоионизации
- метод ионизации в индукционно-связанной плазме

В масс-спектрометрии измеряемой величиной является отношение массы иона к его заряду. Непосредственно измерение этой величины производится в устройстве, называемом масс-анализатором. Результат измерений масс-спектрометра напрямую зависит от выбора типа масс-анализатора и ионного источника.

Масс-спектрометр - сложное устройство, представляющее собой структуру, представленную на блок-схеме (рисунок 1). Масс-спектрометр состоит из нескольких самостоятельных элементов, которые показаны на рисунке: источник ионов с системой хранения и подачи пробы, система транспортировки ионов из пробы к масс-

анализатору, масс-анализатор, детектор, система регистрации экспериментальных данных, система управления. Система из источника, системы транспортировки и масс-анализатора находится в вакууме и контролируется системой откачки и измерения вакуума.



Рисунок 1 – Блок-схема масс-спектрометра

Устройство квадрупольного масс-спектрометра представляет собой четыре симметрично размещенных электрода, отмеченных на рис.2 цифрой «3». Каждый электрод должен иметь совершенное сечение, такой электрод называется - монополюс. К монополюсам попарно подаются комбинации непрерывных и индукционных напряжений.

Под воздействием убыстряющего напряжения ионы вылетают из источника, обозначенного на рисунке 2 цифрой «1» и влетают в прибор параллельно осям электродов. Пройдя через диафрагму, отмеченную на рисунке 2 цифрой «2», ионы начинают колебаться вдоль осей под действием осциллирующего поля, создаваемого электродами. Амплитуда при этом возрастает, причем без изменения направления движения потока ионизированных частиц. Ионы с большим значением амплитуды, нейтрализуются в результате столкновений с электронами. Постоянную амплитуду обретают только ионы, у которых значения  $m/z$  (детектируемая масса) станут отвечать соответствию  $U/V$  (где,  $V$ - радиочастотная компонента,  $U$ -напряжение постоянного тока). Это дает им возможность беспрепятственно двигаться в квадруполе и в результате оказаться детектируемыми в приемнике, отмеченном на рисунке 2 цифрой «4».

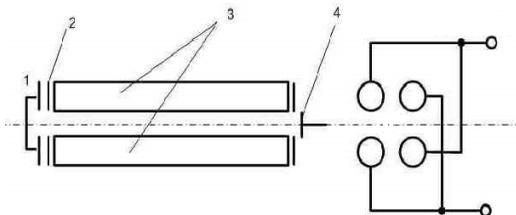


Рисунок 2 – Устройство квадрупольного масс-спектрометра

Разработанная программа призвана облегчить процесс нахождения траектории, и её визуализацию в двухмерном пространстве. Входными данными для нашей программы являются координаты по двум осям и начальная скорость. Математически движение частицы в поле масс-спектрографа описывается следующим уравнением:

$$\frac{d^2x}{dt^2} + \frac{2\sigma ey}{mr^2} (U - V \cos \Omega t) = 0 \quad (\text{для оси } y)$$

$$\frac{d^2x}{dt^2} + \frac{2\lambda ex}{mr^2} (U - V \cos \Omega t) = 0 \quad (\text{для оси } x).$$

Процесс выполнения программы основан на последовательном решении этого уравнения методом Рунге-Кутты 4 порядка с итерациями на каждом следующем моменте времени.

Одной из особенностей написания данной программы является задача переключения систем дифференциальных уравнений с заданным временным интервалом. Для реализации программы на этом этапе возьмем упрощенные уравнения движения частиц и сформируем из них две системы:

$$\begin{cases} \frac{d^2x}{dt^2} = a_1^2 x \\ \frac{d^2y}{dt^2} = -a_1^2 y \end{cases} \quad \begin{cases} \frac{d^2x}{dt^2} = -a_2^2 x \\ \frac{d^2y}{dt^2} = a_2^2 y \end{cases}$$

И теперь будем переключать их с определенным шагом  $t$ . В нашем случае этот шаг равен  $t=0.5$ . Таким образом, мы избегаем складывания погрешностей и образования «шумов». Реализация данного блока показана ниже.

```
using namespace std;
#define a1 3.6
#define a2 2.92
#define threshold 0.5
double fdX(double t, double U)
{
    return U;
```

```

}
double fdU(double t, double X)
{
int k = floor(t / threshold + 0.5);
if (k % 2 )
    return a1 * a1 * X;
    else
        return -a2 * a2 * X;
}
double fdY(double t, double V)
{
    return V;
}
double fdV(double t, double Y)
{
int k = floor(t / threshold + 0.5);
if (k % 2 == 0)
    return -a1 * a1 * Y;
    else
        return a2 * a2 * Y;
}

```

Также стоит упомянуть о реализации метода Рунге-Кутты с плавающим шагом в решении систем уравнений, описывающих движение частиц. В нашем случае мы с каждой итерацией решаем систему дифференциальных уравнений в каждой новой точке времени. Такую систему мы реализуем следующим образом:

```

k1X floor = h * fdX(t, U);
k1U = h * fdU(t, X);
k1Y = h * fdY(t, V);
k1V = h * fdV(t, Y);
k2X = h * fdX(t + h / 2, U + k1U / 2);
k2U = h * fdU(t + h / 2, X + k1X / 2);
k2Y = h * fdY(t + h / 2, V + k1Y / 2);
k2V = h * fdV(t + h / 2, Y + k1Y / 2);
k3X = h * fdX(t + h / 2, U + k2U / 2);
k3U = h * fdU(t + h / 2, X + k2X / 2);
k3Y = h * fdY(t + h / 2, V + k2Y / 2);
k3V = h * fdV(t + h / 2, Y + k2Y / 2);
k4X = h * fdX(t + h, U + k3U);
k4U = h * fdU(t + h, X + k3X);
k4Y = h * fdY(t + h, V + k3Y);
k4V = h * fdV(t + h, Y + k3Y);

```

$$X += (k1X + 2 * k2X + 2 * k3X + k4X) / 6;$$

$$U += (k1U + 2 * k2U + 2 * k3U + k4U) / 6;$$

$$Y += (k1Y + 2 * k2Y + 2 * k3Y + k4Y) / 6;$$

$$V += (k1V + 2 * k2V + 2 * k3V + k4V) / 6;$$

$$t += h;$$

, где h- шаг работы метода Рунге-Кутта, t- шаг времени.

Блок входных данных выглядит следующим образом:

```
double a = 0.05;
double b = 100;
double h = 0.05;
double t = 0;
double X = 1;
double U = 0;
double Y = 1;
double V = 0;
```

, где b и a задают отрезок времени, на котором программа работает (например, от 0 до 100 секунд), h- шаг работы программы, t- начальное время, X и Y- начальные координаты, U и V- начальная скорость по соответствующим координатам.

В программе реализован вывод данных в файл, с помощью процедуры myfile:

```
myfile << "Solution:" << endl;
cout << "t\tX\tY" << endl;
myfile << "t\tX\tY" << endl;
```

Пример выведенных данных:

| Solution: |        |        |
|-----------|--------|--------|
| t         | X      | Y      |
| 0         | 1      | 1      |
| 0,05      | 0,9894 | 0,9838 |
| 0,1       | 0,9577 | 0,9359 |
| 0,15      | 0,9056 | 0,8577 |
| 0,2       | 0,8343 | 0,7518 |
| 0,25      | 0,7452 | 0,6216 |
| 0,3       | 0,6402 | 0,4713 |
| 0,35      | 0,5217 | 0,3058 |
| 0,4       | 0,392  | 0,1304 |
| 0,45      | 0,254  | -0,049 |
| 0,5       | 0,1106 | -0,227 |

В файл записываются координаты и время, по ним в ПО «Microsoft Excel» строятся графики, следующего вида:

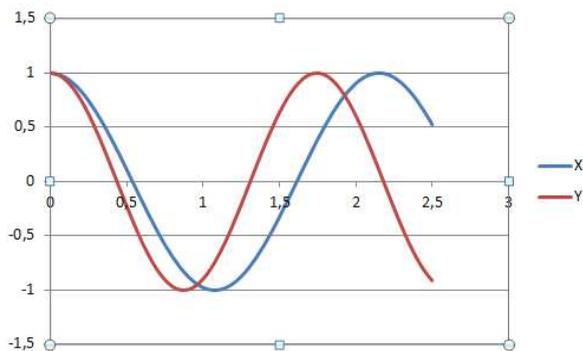


Рисунок 3 – Графики, полученные в результате работы программы

В результате работы мы получили рабочую программу, которая в комбинации с ПО «Microsoft Excel» позволяет определять и визуализировать траекторию частицы, пролетающей через поле квадрупольного масс-спектрометра в двумерном пространстве. Данное исследование призвано облегчить работу с масс-спектрометрами и автоматизировать расчет траектории. В перспективе существует возможность доработки программы для работы в трехмерном пространстве, что еще сильнее упростит задачу визуализации траектории частицы.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Дубков М.В., Иванов В.В. Система ввода и вывода ионов в гиперболических масс-спектрометрах на основе квадрупольных ячеек с продольным квадрупольным полем // Вестник РГРТУ. – 2012. – №3 (Выпуск 41). – С.104-107.
2. Гуров В.С., Дубков М.В., Буробин М.А., Иванов В.В., Николаев А.В., Харланов И.А. Квадрупольные масс-анализаторы на основе тонкостенных гиперболических электродных систем // Вестник РГРТУ. – 2017. – № 2 (Выпуск 60). – С. 141-147.
3. Роберт Лафоре. Объектно-ориентированное программирование в C++.
4. Газовый анализ с использованием квадрупольного масс-спектрометра: Лабораторная работа по курсу: Физические методы исследования. / Сост. Попов И.А., Кукаев Е.Н., Куксин А.Ю. – М.: МФТИ, 2012, - 36с.

УДК 004.021

**Х.Х. НГУЕН**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**АНАЛИЗ МЕТОДОВ ТРАССИРОВКИ ПЕЧАТНЫХ ПЛАТ**

*В статье рассматриваются алгоритмы трассировки на трудоемкость и точность результатов с целью определения наилучшего при решении конкретной задачи.*

**Введение.** В настоящее время следствием высоких темпов научно-технического прогресса является постоянно возрастающая необходимость улучшения электронно-вычислительной техники (ЭВТ) и радиоэлектронной аппаратуры (РЭА). Одной из многих трудоемких задач, которые возникают при автоматизации проектирования РЭА, является задача трассировки. Она заключается в построении соединений между выводами элементов, которые располагают в заданном монтажном коммутационном пространстве (МКП) в соответствии с принципиальной электрической схемой устройства с учетом конструктивных ограничений.

Когда дело доходит до печатных плат, прослеживается в основном связь с сетью из меди, проводки, изоляции, а также предохранителей, которые состоят из платы. В целом, существует большая вероятность того, что трассировка печатной платы будет несовместимой, если в ней есть какие-либо неполадки с любым данным электронным устройством. Более того, сопротивление трассировки является наиболее распространенным термином, используемым электронщиками в индустрии печатных плат. Под сопротивлением трассировки понимаются части печатной платы, которые противостоят току.

Трассировка печатной платы важна для каждой печатной платы, как ток, и она потратит больше времени и энергии на подгонку, если ошибочно обнаружить или рассчитать трассу. Что еще хуже, это может привести к ошибкам в системе для любого устройства.

Существует много алгоритмов трассировки: волновые, эвристические, каналные, гибкие и др. В данной статье мы рассмотрим волновой алгоритм (алгоритм Ли) и алгоритм Рабина, а также определим достоинства и недостатки указанных алгоритмов.

**Постановка задачи.** Задача трассировки соединений отдельных компонентов возникает на последних этапах проектирования электропоезда. Исходными данными для ее решения являются: принципиальная схема соединений, таблица размещения элементов схемы, конст-

руктивные и технологические ограничения. Необходимо реализовать соединения для того, чтобы соединения, которые принадлежат разным сетям, не имели пересечений в одном слое.

Задача трассировки печатного монтажа обычно заключается в себя четыре этапа:

1. Определение списка соединений, т.е. пар непосредственно соединенных между собой точек;
2. Распределение соединений по слоям;
3. Определение порядка разводки всех цепей и отдельных (двух-контактных) соединений;
4. Отслеживание отдельных связей.

**Волновой алгоритм (алгоритм Ли).** Большинство алгоритмов конфигурации проводов используют идею алгоритма волны лжи, который представляет собой процедуру поиска кратчайшего пути в графе.

В общем случае алгоритм Ли может заключаться в себя следующие этапы:

1. Монтажное поле разбивается на квадратные участки, размеры которых определяются допустимой шириной дорожек, зазорами между ними и размерами площадок.

2. Полевые площадки делятся на занятые и свободные. На занятых участках (ранее проложенных проводниках, установленных элементах и т.п.) прокладка трасс запрещена. Трассу можно проложить только через соседние участки, если они свободны.

3. Для ячейки, из которой собираемся построить маршрут, назначаются приоритетные направления. Для противоположной соты будут обратные приоритетные направления. Выбор направлений 1 и 2 производится из условия встречи волны с конечной точкой пути, а направления 3 и 4 используются для обхода препятствий.

4. Распространение волны заключается в присвоении участкам, соседним с ранее рассмотренными, определенного значения весовой функции. Вес ячеек  $k$ -го фронта функции равен:  $F_k = F_{k-1} + 1$  (1). Процедура распространения волны продолжается до тех пор, пока расширяющийся фронт волны не достигнет цели.

5. Прокладка маршрута состоит в переходе от достигнутой области к исходной по участкам, пройденным на этапе распространения волны, таким образом, чтобы значение функции  $F_k$  монотонно уменьшалось.

Пример работы алгоритма представлен на рисунке 1.

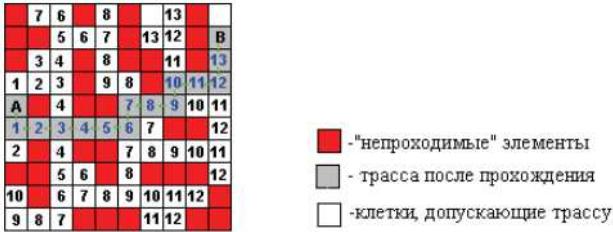


Рисунок 1 – Пример результата работы алгоритма Ли

Алгоритм Ли позволяет построить путь минимальной длины, но требует значительных затрат времени при работе на компьютере. При этом 90 % времени уходит на распространение волн, 10 % — на проводящие дорожки.

Основным недостатком алгоритма Ли является зависимость общей длины соединений от последовательности маршрута. От этого зависит и возможность реализации маршрутов.

**Алгоритм Рабина.** В алгоритме Рабина прирост производительности достигается за счет преимущественного распространения волны в направлении целевой ячейки. В алгоритме Ли, являющемся по сути реализацией метода динамического программирования, такой целенаправленности при поиске трассы нет. Алгоритм Рабина, реализующий другой комбинаторный метод — метод ветвей и границ, позволяет, как и алгоритм Ли, находить глобальный оптимум целевой функции [1].

Пусть требуется найти путь минимальной длины между ячейками А и В;  $C_i$  - некоторая ячейка ДРП, входящая в следующий фронт волны.

Значение весовой функции  $P_i$ , отнесенное к  $C_i$  – й ячейке фронта, определяется по формуле:

$$P_i = L(C_i, A) + \eta(C_i, B) \quad (2)$$

где  $L(C_i, A)$  – длина пути, пройденного волной от А до ячейки  $C_i$ .

$$L(C_i, A) = |x_{C_i} - x_A| + |y_{C_i} - y_A| \quad (3)$$

$\eta(C_i, B)$  – расстояние между ячейками  $C_i$  в В в ортогональной метрике.

$$\eta(C_i, B) = |x_{C_i} - x_B| + |y_{C_i} - y_B| \quad (4)$$

Видим, что  $\eta(C_i, B)$  - нижняя оценка длины пути из  $C_i$  в В, полученная в предположении, что на этом пути нет препятствий. Следовательно, выражение (2) в целом является нижней оценкой длины любого пути из А в В, проходящего через ячейку  $C_i$ .

По методу ветвей и границ оптимальное решение следует искать в подмножестве решений, имеющих наилучшую оценку. В соответствии с этим в формируемом очередном волновом фронте выбирается подмножество ячеек с минимальной оценкой (2), и распространение волны на следующем шаге осуществляется из ячеек этого подмножества.

Для уменьшения области поиска распространение волны осуществляется в первую очередь из ячейки с минимальной оценкой, которая получила последнюю оценку. Последовательность просмотра соседних ячеек такая же, как и в алгоритме Ли (верхняя, правая, нижняя, левая).

**Заключение.** В итоге, достоинства и недостатки каждого алгоритма представлены на следующей таблице:

Таблица 1 – Сравнимость характеристик алгоритмов

|                 |  |
|-----------------|--|
| Алгоритм Ли     | <p><b>Достоинства:</b> позволяет строить путь минимальной длины; легко распараллеливается при использовании нескольких начальных точек генерации волны; путь всегда будет найден, если она существует.</p> <p><b>Недостатки:</b> большой объем требуемой памяти ЭВМ и значительное время работы, связанное с просмотром большого количества ячеек ДРП при распространении волны.</p> |
| Алгоритм Рабина | <p><b>Достоинства:</b> сокращения зоны поиска распространение волны; увеличение быстродействия достигается за счёт преимущественного распространения волны.</p> <p><b>Недостатки:</b> построенный путь по данному алгоритму может быть не иметь минимальной длины.</p>   |

### СПИСОК ИСПОЛЪЗУЕМЫХ ИСТОЧНИКОВ

1. Муромцев Ю.Л., Орлова Л.П., Муромцев Д.Ю., Тютюнник В.М. Информационные технологии проектирования РЭС. Ч. 1: Основные понятия, архитектура, принципы: Учеб, пособие. Тамбов: Изд-во Тамб. гос. техн. ун-та, 2004.
2. Уваров А.С. Проектирование печатных плат. 8 лучших программ: М.: ДМК Пресс, 2009 С. 181, 195, 237.
3. Михайлов И. Е. Разработка модификации алгоритма волновой трассировки (алгоритма Ли) // Наука, техника и образование, 2016. №3 (21). С. 60-62.

УДК 004.421

**Х.Х. НГУЕН**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**РЕАЛИЗАЦИЯ ВОЛНОВОГО АЛГОРИТМА ДЛЯ  
ОПРЕДЕЛЕНИЯ КРАТЧАЙШЕГО ПУТИ НА ПЛОСКОСТИ**

*В статье рассматривается программная реализация волнового алгоритма для трассировки кратчайшего пути на плоскости и реализация алгоритма с помощью разработанного программного обеспечения.*

**Введение.** Волновой алгоритм (алгоритм Ли) является одним из наиболее уникальных алгоритмов трассировки. Он позволяет построить путь между двумя элементами в любом лабиринте.

**Постановка задачи и описание алгоритма.** При трассировке соединений с помощью волнового алгоритма вся плоскость трассировки разбивается на ячейки, которые образуют дискретное рабочее поле (ДРП). Ячейки, через которые можно проходить проводники, являются свободными. Некоторые клетки поля считаются занятыми, то есть к занятым следует отнести те ячейки, в которых должны быть размещены элементы и их контакты, подлежащие соединению. Кроме того, занятыми считаются те ячейки, через которые проходят ранее протянутые проводники, а также крайняя ячейка платы.

При использовании ДРП, включение элементарной ячейки в путь означает проведение печатного проводника, так как показано на рисунке 1.

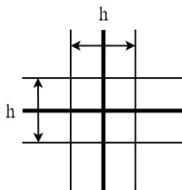


Рисунок 1 – Координатная сетка ячейки ДРП

Считаем, что основная координатная сетка сдвинута на  $h/2$  для того, чтобы пути идут от ячейки к ячейке, а не по координатным линиям ДРП.

На каждом шаге алгоритма некоторые ячейки ДРП заняты. Ячейки, попадающие в зону, запрещенную для трассировки: крайние

поля платы, зоны размещения элементов и их выводов, ранее проведенные проводники.

**Реализация алгоритма метода.** Рассмотрим работу алгоритма при поиске пути из ячейки А в ячейку В. Алгоритм состоит из двух частей. В первой части волна распространяется от источника к приемнику, то есть к начальной до конечной ячейки. Во втором выполняется обратный ход, в ходе которого построится путь из ячеек волны.

1. Распространение волны. Целью алгоритма является распространение цифровой волны от моделируемого на поле источника (ячейки А) до раскрытия фронта волны максимально широкой как ячейка В или до тех пор, пока фронт не сможет включать новые пустые ячейки. Все условия, которые должны быть соблюдены при трассировке связей, изложены в правилах, по которым движется фронт волны.

2. Построение трасса выполняется от конечного элемента к начальному. Если в результате его выполнения достигнута ячейка В, то выполняется вторая часть алгоритма — «проведение пути». В противном случае при заданных условиях путь из А в В не может быть проложен.

На этапе распространения волны очередное построение лицевой грани начинается с присвоения некоторых числовых значений (весов) ячейкам, примыкающим к ячейкам лицевой грани. Клетки с общим ребром обычно считаются соседними. Правило назначения весов (т. е. тип весовой функции) зависит от критериев оптимизации, используемых при трассировке. Итак, если длину соединения минимизировать, то весовое соотношение ячеек, которые присутствуют до  $k$ -й и  $(k-1)$ -й, будет следующим:

$$P_k = P_{k-1} + 1 \quad (1)$$

Если минимизируется количество пересечений трасс, то это соотношение может иметь вид:

$$P_k = \begin{cases} P_{k-1}, & \text{если } k - \text{ячейка свободна,} \\ P_{k-1} + 1, & \text{если } k - \text{ячейка является полузанятой.} \end{cases}$$

(полузанятыми называют ячейки, в которых допускается пересечение трасс).

Волна распространяется от ячеек очередного фронта с минимальным весом, до тех пор, пока последней ячейке не будет задано определенное значение весовой функции или она не сможет включать в себя ни одной пустой ячейки следующего фронта (во втором случае подключение от А к В невозможно). Достигнув ячейку В совершается «обратное» движение к ячейке А по ячейкам ДРП таким образом, чтобы значение весовой функции уменьшалось монотонно.

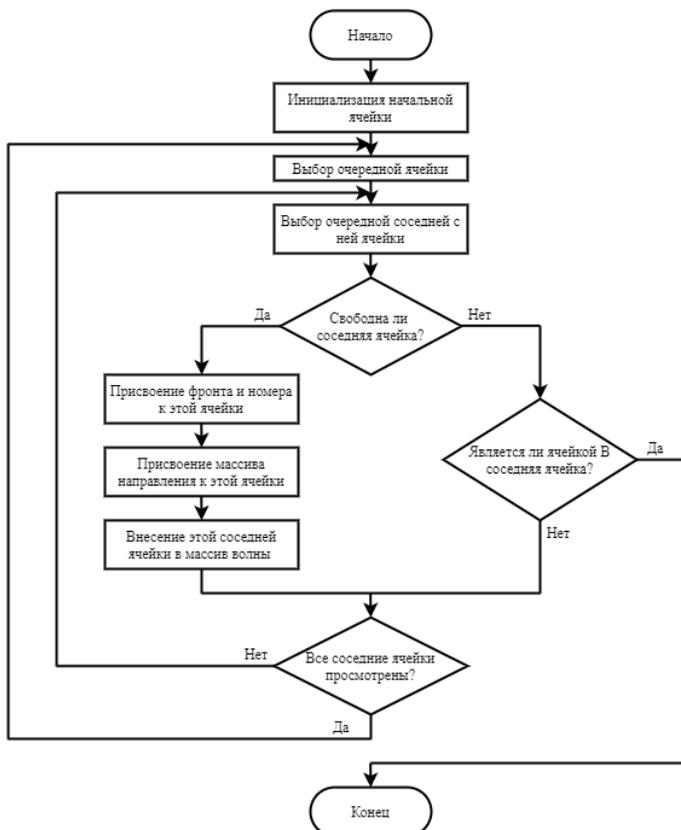


Рисунок 2 – Блок-схема алгоритма построения волны из А до В

### Программная реализация и демонстрация результатов вычислительных экспериментов.

В программе имеется возможность задать ширину и высоту поля. Пусть иметь дискретное рабочее поле с размером 12 на 8. Заполним такие размеры как начальные данные, появляется таблицу и в ней заметим место точек А, В и запрещенных ячеек, которые обозначаются буквой X, как показана на рисунке 3.

При распространении волны будем выбрать следующую последовательность просмотра соседних ячеек: верхняя, правая, нижняя, левая (↑, →, ↓, ←).

Сначала мы рассмотрим точку А с координатами (6, 2), которая имеет значение веса ячейки равно 0. При распространении волны от А, алгоритм последовательно строит  $\Phi_1(A), \Phi_2(A), \dots, \Phi_k(A)$  ее фронты.

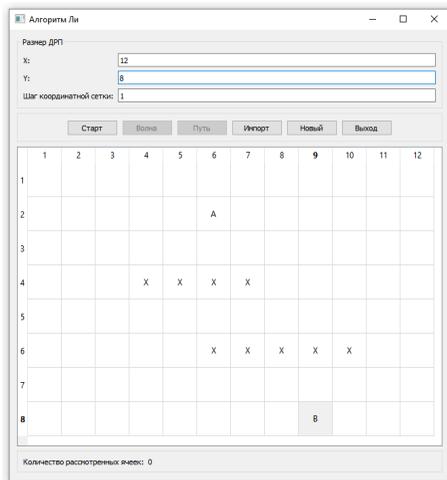


Рисунок 3 – Исходные данные для примера

Затем рассмотрим ячейку  $A_1$  с координатами (6, 1), у которой значение веса ячейки равно:  $P_{A_1} = |6 - 6| + |2 - 1| = 1$ . Ячейки, у которой значение веса такое же равно, имеет следующие координаты: (7, 2), (6, 3), (5, 2). Получаем,  $\Phi_1(A) = \{(6, 1), (7, 2), (6, 3), (5, 2)\}$ . Аналогично, строим  $\Phi_2(A), \dots, \Phi_k(A)$  фронты волны от A.

На рисунке 4 показывается результат распространения волн от точки A в точку B. Кратчайший путь обозначается звездочкой.

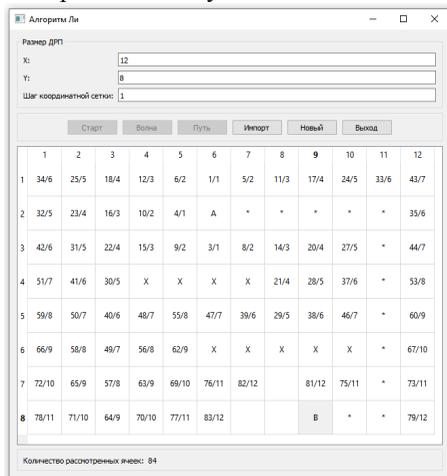


Рисунок 4 – Проведение кратчайшего пути между A и B

**Заключение.** По результатам тестирования следует, что данный алгоритм трассировки позволяет найти кратчайший путь между заданными точками, обходя препятствия и пропуская уже проложенные трассы. Однако при использовании этого метода с большим размером поля и количеством трасс, возможны значительные временные затраты на построение волнового фронта, поэтому метод нуждается в некоторой оптимизации. Исходя из особенностей метода, необходимо предельно ограничить направления поиска, исключить явно длинные пути и добавить критерии оптимизации, такие как количество поворотов и ширина трассы, следует учесть возможность пересечения трасс.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Широ Г.Э., Лapidус Л.И. «Метод трассировки печатных соединений // Применение вычислительных машин для проектирования цифровых устройств». – М.: Сов.радио, 1968. Уваров А.С. Проектирование печатных плат. 8 лучших программ: М.: ДМК Пресс, 2009 С. 181, 195, 237.
2. Михайлов И. Е. Разработка модификации алгоритма волновой трассировки (алгоритма Ли) // Наука, техника и образование, 2016. №3 (21). С. 60-62.
3. Гостин А.М., Сапрыкин А.Н. «Основы разработки объектно-ориентированного программного обеспечения».

УДК 004.65

### А.Е. НЕМЦЕВ

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### СИЛОВЫЕ АЛГОРИТМЫ

*Рассматриваются разновидности силовых алгоритмов, их преимущества и применимость.*

Семейство силовых алгоритмов размещения относят к алгоритмам визуализации графа, основанным на физических законах. Моделью, которая представляет из себя физическую систему, здесь выступает сам граф, в котором действуют определённые силы. В это системе могут использоваться различные типы сил вместе или же обособленно. Примеры сил используемых в силовых алгоритмах: сила притяжения, отталкивания, гравитации, магнитного поля и др.

### Основные понятия

Для начала рассмотрим в чём основная идея силовых алгоритмов, почему они популярны и с какими графами они работают.

**Силовой алгоритм** - это алгоритм, который рассматривает начальный граф как систему сил, чаще всего вершины представляются в виде физических тел, а рёбра в виде пружин. Благодаря такому представлению мы получаем возможность использовать простые физические формулы для нахождения размещения этого графа.

Алгоритмы основанные на физических моделях имеют преимущества:

- Они достаточно интуитивны, так как используют всем знакомые физические законы;
- Для графов размером около 150 вершин они выдают хороший результат;
- Результирующий граф выглядит намного проще и понятней;
- Их нетрудно запрограммировать, так как они не используют сложных вычислений.

Силовые алгоритмы работают со связными и неориентированными графами. Так как при их работе не учитывается направление рёбер.

Силовые алгоритмы относятся к непрерывно-дискретным методам размещения. Для данного класса алгоритмов размещение осуществляется на непрерывной плоскости и задание фиксированного набора позиций необязательно.

Они достаточно трудоёмки и требуют подбора многих коэффициентов опытным путем.

Также непрерывно-дискретные алгоритмы размещения из-за их трудоёмкости применяются лишь для задач небольшой размерности ( $n < 20$ ).

### Алгоритм Идеса

Одним из самых первых появившихся силовых алгоритмов является алгоритм Идеса. Его основная идея была в том, чтобы представить вершины как заряженные тела, а рёбра в виде пружин. Он предположил что если в такой физической системе вершинам дать возможность перемещаться под действием сил, то в какой-то момент граф будет размещен таким образом, что общая сила системы будет либо минимальна, либо равна нулю.

Особенностью данного алгоритма было то, что сила отталкивания действовала только между несмежными, а сила притяжения действовала только между смежными вершинами.

На первом этапе, высчитываются силы действующие на каждую из вершин, после чего силы складываются, образуя вектор, в направлении которого нужно переместить вершину.

$$f_{rep}(p_u, p_v) = \frac{c_{rep}}{|p_v - p_u|^2} - \text{формула силы отталкивания, где}$$

$c_{rep}$  является константой управляющей величиной силы.

$$f_{spring}(p_u, p_v) = c_{spring} \log \frac{|p_u - p_v|}{l} - \text{формула силы}$$

притяжения, где  $c_{spring}$  константа управляющая силой пружины, а  $l$  естественная длина пружины.

$|p_u - p_v|$  - это Евклидово расстояние между вершинами.

На втором этапе просматриваются все вершины и выбирается та на которую действует максимальная сила и перемещают её на длину вектора силы.

#### Алгоритм Фрюхтермана-Рейнгольда

Этот алгоритм оптимизирует алгоритм Идеса, так как исключает вычисление логарифмов, что существенно ускоряет работу алгоритма и уменьшает его сложность.

Помимо отсутствия нужды считать логарифмы, алгоритм Фрюхтермана-Рейнгольда отличается от Идеса тем что силы отталкивания действуют между всеми вершинами, а не только между несмежными, что немного увеличивает длину рёбер и субъективно делает результирующий граф более разборчивым.

Также в данном алгоритме вводится понятие “Идеальной длины”, которая зависит от площади пространства где будет расположен граф и количества вершин графа:

$$l = \sqrt{\frac{area}{|V|}}$$

Фрюхтерман и Рейнгольд предложили заменить константы, которые использовал Идес в своих вычислениях, на “идеальную длину”, что позволило изменить размер результирующего графа так, чтобы он занимал всё возможное пространство.

Также у этого алгоритма отсутствует потребность расчёта общей силы системы, для перемещения достаточно лишь общей суммы сил, действующих на одну конкретную вершину.

Силы в этом алгоритме рассчитываются следующим образом:

$$f_{spring}(p_u, p_v) = \frac{|p_u - p_v|^2}{l}$$

$$f_{rep}(p_u, p_v) = \frac{l^2}{|p_u - p_v|}$$

### Алгоритм Frick

На использовании одной пружинной модели не остановились, так как скоро были обнаружены некоторые проблемы при размещении несвязного или слабо связного графа, несвязные компоненты оказывались в разных сторонах пространства из-за отсутствия весомой силы притяжения. Это очень портило результирующий граф, так как основное преимущество силовых алгоритмов заключалось в эстетически приятном отображении графа.

Однако в работе (Frick et al 1995) было принято решение ввести силу гравитации, которая была бы прямо пропорциональна количеству инцидентных рёбер. Также из-за этого сильно увеличилась сходимость и скорость алгоритма.

Силы в этом алгоритме рассчитываются следующим образом:

$$f_{rep}(p_u, p_v) = \frac{l^2}{|p_u - p_v|^2}$$

$$f_{attr}(p_u, p_v) = \frac{|p_u - p_v|^2}{l \cdot \Phi(v)}$$

$\Phi(v)$  в формуле притяжения рассчитывается как:

$$\Phi(v) = 1 - \frac{\deg(v)}{2}$$

А сила гравитации:

$$F_{grav}(v) = \Phi(v) \cdot c_{grav}$$

Где  $c_{grav}$  является константой и задаётся изначально.

### Сравнение представленных алгоритмов

В таблице ниже представлено сравнение рассмотренных алгоритмов.

| Алгоритм                        | Реализация | Быстродействие | Ресурсозатратность | Сложность            |
|---------------------------------|------------|----------------|--------------------|----------------------|
| Алгоритм Идеса                  | Средняя    | Низкое         | Средняя            | $O(V)$<br>$O(U)$     |
| Алгоритм Фрюхтермана-Рейнгольда | Средняя    | Среднее        | Средняя            | $O(V^2)$ $O(U)$      |
| Алгоритм Frick                  | Сложная    | Высокое        | Высокое            | $O(V^2)$<br>$O(U^2)$ |

Исходя из этих данных можно сказать, что самым эффективным из силовых алгоритмов является алгоритм Frick, но из-за его сложности он не является самым популярным из рассматриваемого семейства алгоритмов.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Сапрыкин А.Н. Алгоритмические методы автоматизации конструирования электронных средств. Учебное пособие — Рязань: ИП Коныхин А.В. (Book Jet), 2021 – 116 с.
2. Kamada, T. An algorithm for drawing general undirected graphs / T. Kamada, S. Kawai // Information Processing Letters. – 1989. – Vol. 31. – P. 7–15.
3. Fruchterman T. M. J., Reingold E. M., Graph Drawing by ForceDirected Placement. Software -Practice and Experience, 1991, 21(11), 1129-1164.
4. Информационное общество: современное состояние и перспективы развития: сборник материалов X международного студенческого форума. – Краснодар: КубГАУ, 2018. – 358 с.
5. Перспективные информационные технологии (ПИТ 2018) [Электронный ресурс]: труды Международной научно-технической конференции / под ред. С.А. Прохорова. – Самара: Издательство Самарского научного центра РАН, 2018. – 1424 с.

УДК 658.5

### **И.В. НИЖУТИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА ПОПУЛЯРНЫХ СИСТЕМ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ**

*Приводится сравнение возможностей современных САД (САМ, САЕ) систем и их обзорные характеристики.*

Система автоматизированного проектирования — система, которая реализует технологию выполнения функций проектирования, представляющую собой организационно-техническую систему, предназначенную для автоматизации проектирования, состоящую из комплекса технических, программных и др. средств автоматизации ее

деятельности. Также для обозначения подобных систем часто используется аббревиатура САПР.

В современном мире целесообразно использовать системы автоматизированного проектирования по различным причинам. Во-первых, эти системы значительно сокращают сроки выполнения конструкторских работ, во-вторых, позволяют использовать чертежи проектных решений многократно, в отличие от чертежей, выполненных вручную, также САПР повышают точность выполнения чертежей. Во многих САПР есть системы инженерного анализа, которые позволяют проводить исследования спроектированных моделей. Эти САЕ системы сокращают время разработки изделия в целом, исключают ручные расчеты, а также сокращают затраты на проведение экспериментов над реальными изготовленными моделями.

Существует большое количество машиностроительных и др. САПР, и их выбор зависит от задачи, которую необходимо решить. САПР подразделяются на САД, САМ и САЕ-системы.

Под САД-системами (computer-aided design – компьютерная поддержка проектирования) понимают программное обеспечение, автоматизирующее труд инженера-конструктора и позволяющее решать задачи проектирования изделий и оформления технической документации при помощи компьютерных систем.

САМ-системы (computer-aided manufacturing – компьютерная поддержка изготовления) автоматизируют расчеты траекторий перемещения инструмента для обработки деталей на станках с ЧПУ и обеспечивают выдачу управляющих программ с помощью компьютерных систем.

САЕ-системы (computer-aided engineering – компьютерная поддержка инженерных расчетов) предназначены для решения инженерных задач, которые включают в себя анализ и расчет определенных параметров проектируемого изделия, например для расчетов конструктивной прочности, анализа тепловых процессов, расчетов гидравлических систем и механизмов и др.

Резюмируя вышеизложенное, использование в современном мире САПР является неотъемлемой частью в работе каждого инженера, и для упрощения решения любой инженерно-технической задачи можно подобрать наиболее подходящую САПР.

В данной статье к рассмотрению представлены САПР для черчения и твердотельного моделирования:

- КОМПАС 3D;
- PTC Creo Parametric.

**КОМПАС 3D**

«Компас» — семейство систем автоматизированного проектирования с возможностями оформления проектной и конструкторской документации согласно стандартам серии ЕСКД. Относится к САПР среднего уровня, разрабатывается российской компанией «Аскон». Название линейки не имеет общих связей с навигационным прибором, а является акронимом от фразы «комплекс автоматизированных систем». В торговых марках применяется написание заглавными буквами: «КОМПАС». Первый выпуск «Компаса» состоялся в 1989 году. Первая версия под Windows получила свет в 1997 году.

Система «Компас-3D» предназначена для создания ассоциативных трёхмерных моделей отдельных деталей (в том числе, деталей, формируемых из листового материала путём его гибки) и сборочных единиц, содержащих как оригинальные, так и стандартизованные конструктивные элементы. Быстро получать модели типовых изделий на основе проектированного ранее прототипа позволяет параметрическая технология. Огромное количество сервисных функций облегчают решение вспомогательных задач проектирования и обслуживания производства в целом.

«Компас-3D» включает следующие компоненты системы: система трёхмерного твердотельного моделирования, система автоматизированного проектирования «Компас-График» и модуль формирования спецификаций на сборочные единицы. Стоит также отметить, что «Компас-3D» - полностью импортнезависимая система. В основе «Компас-3D» лежит российское геометрическое ядро C3D (создано дочерней компанией АСКОН - C3D Labs) и собственные программные технологии.

Примеры работ, полученных с помощью программы КОМПАС 3D представлены на рисунках 1,2:

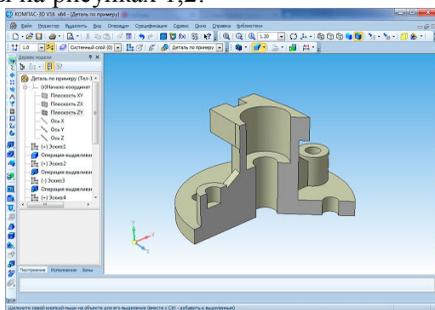


Рисунок 1 – 3D модель в «Компас 3D»

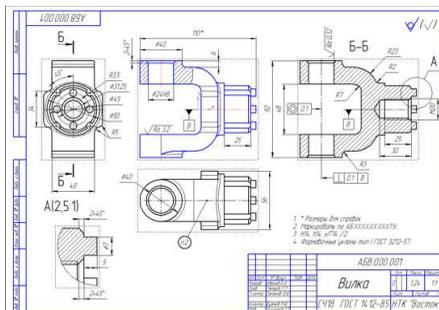


Рисунок 2 – Чертеж детали

Преимущества системы «Компас-3D»:

- Простой и понятный любому пользователю интерфейс;
- Использование трехмерного ядра собственной разработки (С3D);
- Полная поддержка ЕСКД и ГОСТ при проектировании и оформлении документации;
- Большой набор надстроек для проектирования отдельных разделов проекта;
- Гибкий подход к оснащению рабочих мест проектировщиков;
- Возможность интеграции с системой автоматизированного проектирования технологических процессов ВЕРТИКАЛЬ и другими системами.

### ***PTC Creo Parametric***

PTC Creo — это функциональный пакет программного обеспечения для конструирования изделий. Относится к «тяжелым» САПР. Он позволяет инженеру-конструктору или группам конструкторов создавать, анализировать, просматривать проекты изделий при дальнейшем конструировании, используя 2-х и 3-х-мерное моделирование CAD, прямое и параметрическое моделирование. Среда PTC Creo появилась на мировом рынке в июне 2011 года в качестве замены таким продуктам PTC, как Pro/Engineer, CoCreate и ProductView.

Разработчики создали приложение PTC Creo Parametric в качестве надежного базового программного обеспечения, которое позволяет пользователям расширять функциональность проектируемого изделия с помощью детальной проработки каждого отдельного компонента. По мере роста сложности изделий, система Creo предлагает расширенные возможности выполнения требований технического задания на этапе проектирования.

Важной особенностью PTC Creo является наличие средства структурного и теплового анализа конструкций деталей и сборочных единиц PTC Creo Simulate. PTC Creo Simulate включает в себя средства расчета конструкций, теплового анализа и расчета вибраций с помощью метода конечных элементов (МКЭ), что позволяет провести анализ поведения виртуальных 3D-прототипов еще до того как перейти к их физическому изготовлению.

Инженер с помощью интуитивного интерфейса программы прикладывает различные напряжения к цифровому прототипу и освобождает себя от рутинных расчетов. PTC Creo Simulate было разработано для того, чтобы любой инженер-конструктор смог выполнить простое имитационное моделирование.

В приложении PTC Creo Parametric, необходимо выбрать нужную вкладку в интерфейсе пользователя, а затем применить нагрузки, материалы и граничные условия к геометрии конструкции в PTC Creo Simulate. Программа сама полностью рассчитает и визуализирует результат. При необходимости можно вносить упрощения в модель (сохраняя ее адекватность) для создания расчетов и визуализаций в короткие сроки.

Возможности PTC Creo Simulate включают в себя:

- анализ конструкций,
- построение сетки,
- оптимизацию,
- линейный тепловой анализ установившегося режима,
- библиотеку материалов.

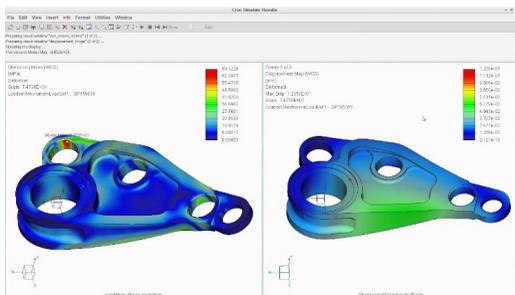


Рисунок 3 – Расчет статических нагрузок в Creo Simulate

В заключении можно подвести следующий итог: с помощью программы КОМПАС очень удобно осуществлять именно создание чертежей, поскольку эта САПР обладает большой библиотекой ГОСТ-

овских изделий и в целом очень хорошо адаптирована именно под российские стандарты.

Соответственно, 3D моделирование удобнее производить с помощью PTC Creo Parametric, поскольку эта САПР имеет более широкий функционал именно как САМ и САЕ система, который может понадобиться для автоматизации инженерных расчетов и анализа конструкций.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Системы автоматизированного проектирования [Электронный ресурс]. – Режим доступа: [https://www.tadviser.ru/index.php/Статья:САПР\\_Системы\\_автоматизированного\\_проектирования](https://www.tadviser.ru/index.php/Статья:САПР_Системы_автоматизированного_проектирования) (дата обращения 11.03.2022).

2. Компас 3D – Система трехмерного моделирования [Электронный ресурс]. – Режим доступа: <https://ascon.ru/products/7/review> (дата обращения 13.03.2022).

3. PTC Creo Simulate [Электронный ресурс]. – Режим доступа: <https://www.irisoft.ru/products/creo/ptc-creo-simulate> (дата обращения 14.03.2022).

УДК 658.5

### **И.В. НИЖУТИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА СИСТЕМ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ЭЛЕКТРОННЫХ УСТРОЙСТВ**

*Приводится оценка и сравнение возможностей современных E-CAD систем.*

Согласно ГОСТ 23501.0-79 САПР представляет собой организационно-техническую систему, состоящую из комплекса средств автоматизации проектирования, связанного с подразделениями организации, выполняющую автоматизированное проектирование.

В области классификации САПР используется ряд терминов, применяемых для классификации программных пакетов и средств автоматизации по целевому и отраслевому назначению.

По отраслевому назначению системы автоматизированного проектирования подразделяются на:

- MCAD - автоматизированное проектирование механических устройств, то есть машиностроительные САПР. Они в основном применяются в автомобилестроении, авиакосмической промышленности, судостроении, производстве товаров народного потребления. Также MCAD включают в себя разработку деталей и сборочных единиц (механизмов) с использованием параметрического проектирования, основанного на основе конструктивных элементов, технологий объемного и поверхностного моделирования.

- EDA или ECAD - САПР электронных устройств, печатных плат, интегральных схем, радиоэлектронных средств и др.

- АЕС CAD - САПР в области строительства и архитектуры. Используются для проектирования промышленных объектов, зданий, мостов, дорог и т.п.

В данной статье рассматриваются только EDA или ECAD системы. EDA - комплекс программных средств, который создан для облегчения разработки электронных устройств. Этот комплекс позволяет создавать и модифицировать базу радиоэлектронных компонентов, создавать принципиальную электрическую схему проектируемого устройства и проверять целостность сигналов на ней. Современные программные пакеты могут выполнить автоматическую компоновку элементов, автоматически трассировать плату в соответствии с созданной ранее принципиальной схемой.

В данной статье к рассмотрению представлены следующие ECAD системы:

- P-CAD
- Altium Designer
- Mentor Graphics Expedition

#### 1. P-CAD

P-CAD — система автоматизированного проектирования электроники. Разработана компанией Altium, предназначена для проектирования печатных плат радиоэлектронных и вычислительных устройств. В 2002 году в России P-CAD являлась самой популярной EDA.

Система P-CAD включает в себя создание принципиальной электрической схемы; перенос схемы на печатную плату; получение для этой схемы чертежа печатной платы с размещенными компонентами; ручное размещение компонентов на печатной плате; ручную, интерактивную или автоматическую трассировку проводников; контроль ошибок в схеме и печатной плате; выпуск конструкторской и технологической документации.

Эта САПР состоит из четырех основных модулей (рисунок 1):  
 P-CAD Schematic – графический редактор схем (используется для создания принципиальной электрической схемы).  
 P-CAD PCB – графический редактор печатных плат (используется для размещения и редактирования компонентов на ПП).  
 P-CAD Autorouters – создан для автоматической трассировки проводников в соответствии с компоновкой, задаваемой пользователем в графическом редакторе P-CAD PCB.  
 P-CAD Library Manager – менеджер библиотек (осуществляет создание и ведение библиотек элементов, используемых при работе).

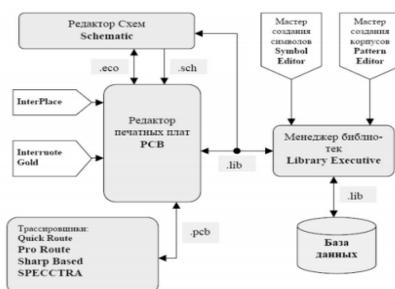


Рисунок 1 – Обобщенная структура системы P-CAD

Процесс проектирования печатной платы устройства в P-CAD выглядит следующим образом:

1. Сначала создается принципиальная электрическая схема (ЭЗ) в P-CAD Schematic. Условные обозначения элементов на схеме не формируют каждый раз заново, а берут из созданных заранее библиотек, которые включают в себя графические изображения символов и корпусов компонентов и другую необходимую текстовую информацию. Результатом работы на этом этапе является графический файл с расширением .sch;

2. Сама принципиальная электрическая схема не используется для трассировки печатной платы. Программный комплекс использует лишь информацию о соединениях внутри электрической принципиальной схемы и типах используемых компонентов. Поэтому вторым этапом создания ПП является создание файла электрических связей с описанием соединений элементов и их типов. По завершении этого этапа появляется отдельный файл списка соединений с расширением .net;

3. Далее в P-CAD PCB на основе созданного ранее файла списка соединений получают чертеж компоновки печатной платы с размещенными на ней компонентами. На чертеже должны быть указаны

линии электрических соединений между их выводами (чертежи корпусов компонентов автоматически загружаются из интегрированной библиотеки). Эта операция называется упаковкой схемы на плату, в результате образуется файл с расширением .pcb;

4. Следующий этап – создание контура печатной платы и размещение на ней компонентов. Эта операция производится пользователем вручную.

5. Автоматическая трассировка дорожек платы производится с использованием встроенного в систему автотрассировщика;

6. Программный комплекс P-CAD использует интегрированные библиотеки, содержащие текстовую и графическую информацию о компонентах. В графическом виде представлены корпуса и символы компонентов, а в текстовом – число секций в корпусе, имена и номера выводов.

## 2. ALTIUM DESIGNER

Altium Designer — САПР радиоэлектронных средств, разработанная компанией Altium. Ранее фирма разрабатывала и улучшала программный комплекс P-CAD, который получил большую популярность среди российских инженеров разработчиков электроники. В 2008 году фирма Altium заявила о прекращении поставки программных пакетов P-CAD, и предложила разработчикам использовать новую программную платформу Altium Designer, которая появилась в 2000 году и имела название Protel. В 2006 году был проведён ребрендинг программного комплекса, который получил текущее название. Последняя на сегодняшний день версия называется Altium Designer 20.

Сегодня Altium Designer — это программный комплекс, позволяющий реализовывать проекты электронных и радиоэлектронных средств на уровне программного кода или схемы с последующей передачей информации проектировщику печатной платы или ПЛИС.

Пакет Altium состоит из двух продуктов, базирующихся на единой платформе DXF. Возможность работы с тем или иным из них зависит от типа имеющейся лицензии:

Altium Designer Custom Board Front-End Design — Схемотехническое проектирование, проектирование ПЛИС и схемотехническое моделирование.

Altium Designer Custom Board Implementation — Проектирование печатных плат и ПЛИС.

В состав программного комплекса Altium Designer входит все необходимое для разработки, редактирования и отладки проектов на базе электрических схем и ПЛИС. Редактор печатных плат Altium De-

signer содержит в себе мощные средства интерактивного размещения компонентов и трассировки проводников, которые совместно с интуитивно-понятной системой установки правил проектирования максимально упрощают процесс разработки электрорадиоизделий. В состав программы входит автоматический трассировщик, в котором используются самые прогрессивные алгоритмы трассировки печатных проводников. Принципиальным отличием последней версии Altium Designer является поддержка компонентов в формате STEP для создания визуальной 3D модели, которые могут быть импортированы/экспортированы из машиностроительных САПР.

Работа над всеми частями проекта ведётся в единой оболочке Design Explorer, поэтому все изменения, внесённые на любом этапе разработки, автоматически передаются на все связанные с ними стадии проекта. Полностью поддерживаются все наработки в виде схем, плат и библиотек, разработанные в последних версиях P-CAD. Интерфейс системы также заметно улучшен в сравнении с P-CAD (рисунок 2).



Рисунок 2 – Интерфейс системы Altium Designer

### 3. MENTOR GRAPHICS XPEDITION

Mentor Graphics, Inc — корпорация, работающая в области автоматизации проектирования электроники (EDA). Основана в 1981 году в США. Компании Mentor Graphics Inc принадлежит лидерство по инвестициям в сфере опытно-конструкторских и научно-исследовательских работ в области проектирования электронных устройств.

Одним из самых востребованных продуктов Mentor Graphics на отечественном рынке является платформа Xpedition. Эта платформа для проектирования печатных плат используется при проектировании сложной электроники в различных модификациях. Основные элементы платформы включают в себя: проектирование в 3D, разработку многоплатных проектов, приёмы по ускорению создания проектов (режимы трассировок, иерархические группы), технологичность

производства и пригодность для испытаний, коллаборацию и сквозное проектирование, разработку аппаратных средств, функционал коллективной работы и управления проектным процессом.

После приобретения Altium компанией Siemens стали появляться релизы, формирующие объединенную цифровую среду для инженеров-конструкторов электроники, механиков, технологов и др. САПР Mentor Graphics ввиду своей высокой функциональности позволяет осуществлять сквозное проектирование печатных узлов.

Примеры интерфейса программы представлены на рисунке 3.

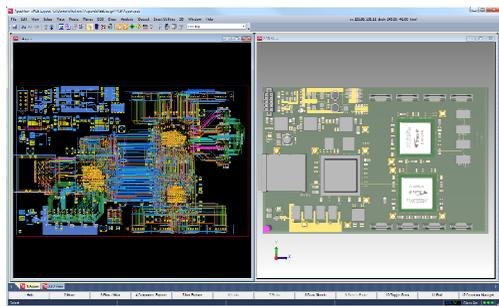


Рисунок 3 – Xpedition PCB Layout, проектирование сложной топологии и 3D визуализация

В заключении можно отметить, что каждая из рассмотренных платформ проектирования ПП имеет свои положительные и отрицательные стороны. Система P-CAD на сегодняшний день является устаревшей и ее применение в разработке ПП будет нецелесообразным, поскольку разработанные документы не смогут применяться для автоматизированного инженерного анализа и инженерных расчетов. Для этих целей более целесообразным будет использование таких систем, как Altium Designer или Mentor Graphics Xpedition.

Решения Mentor разработаны для обеспечения высокой производительности даже для самых сложных проектов и выходят за рамки проектирования, поскольку включают в себя функциональные возможности для исправления ошибок при проектировании, а также проверки на технологичность.

Altium очень прост в изучении, так как его структура не состоит из множества подпрограмм, как это выполнено у Mentor. Стоит так же отметить, что оба этих проекта являются САПРом для ПЛИС, что делает их еще более функциональными.

**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. P-CAD [Электронный ресурс]. – Режим доступа: <https://cxem.net/software/p-cad.php> (дата обращения 01.04.2022).
2. Altium Designer [Электронный ресурс]. – Режим доступа: <https://www.altium.com/ru/altium-designer> (дата обращения 03.04.2022)
3. Mentor Graphics [Электронный ресурс]. – Режим доступа: [https://www.orcada.ru/product/mentor-graphics/na-pechatnyh-platah/products\\_95.html](https://www.orcada.ru/product/mentor-graphics/na-pechatnyh-platah/products_95.html) (дата обращения 03.04.2022).

УДК 004.048

**В.О. НОВИЧКОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

**ИНФОЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ  
РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ**

*В статье рассматривается метод создание реляционной базы данных на основе ER-метода.*

**Инфологическое проектирование.**

Инфологическое проектирование основано на ER-методе.

Суть ER-метода: построение ER-диаграммы, отображающей в графической форме основные объекты предметной области, связи между ними и определение характеристик этих связей.

Основные понятия ER-метода:

- Сущность – это объект предметной области.
- Экземпляр сущности – конкретный представитель объекта.
- Связь – соединение между двумя и более сущностями.
- Экземпляр связи – конкретная связь между экземплярами объектов.
- Ключ сущности — множество атрибутов для однозначной идентификации сущности.

Рассмотрим проектирование базы данных с помощью ER-метода на примере базы данных по автосалону.

Выделение сущностей и связей в заданной предметной области: автомобиль, марка, модель, тип кузова, менеджер, клиент, договор, запчасть, продажа автомобилей.

В предметной области можно выделить следующие связи между сущностями:

- 1) Модель имеет марку
- 2) Автомобиль имеет Модель
- 3) Автомобиль имеет Тип кузова
- 4) К Продаже автомобиля относится Автомобиль
- 5) Для Продажи автомобиля заключается Договор
- 6) Запчасть имеет Модель
- 7) Запчасть имеет Тип кузова
- 8) Запчасть имеет Вид запчасти
- 9) К Продаже запчастей относится Запчасть
- 10) Для Продажи запчастей заключается Договор
- 11) Клиент заключает Договор
- 12) Менеджер заключает Договор

### Построение ER-диаграмм.

- 1) Связь «Модель имеет Марку»

Степень связи:

- Модель может иметь только одну Марку;
- Одну Марку могут иметь несколько Моделей.

Класс принадлежности:

- Модель обязательно имеет какую-то Марку;
- Существует Марка, которую не имеет ни одна Модель.



Рисунок 1 – ER-диаграмма Модель-Марка

- 2) Связь «Автомобиль имеет Модель»

Степень связи:

- Автомобиль может иметь только одну Модель;
- Одну Модель могут иметь несколько Автомобилей.

Класс принадлежности:

- Автомобиль обязательно имеет какую-то Модель;
- Существует Модель, которую не имеет ни один Автомобиль.



Рисунок 2 – ER-диаграмма Автомобиль-Модель

3) Связь «Автомобиль имеет Тип кузова»

Степень связи:

- Автомобиль может иметь только один Тип кузова;
- Один Тип кузова могут иметь несколько Автомобилей.

Класс принадлежности:

- Автомобиль обязательно имеет какой-то Тип кузова;
- Существует Тип кузова, который не имеет ни один Авто-

мобиль.



Рисунок 3 – ER-диаграмма Автомобиль-Тип кузова

4) Связь «К Продаже автомобиля относится Автомобиль»

Степень связи:

- К Продаже автомобиля может относиться только один Автомобиль;
- Один Автомобиль может относиться только к одной Продаже автомобиля.

Класс принадлежности:

- К Продаже автомобиля обязательно относится какой-то Автомобиль;
- Существует Автомобиль, который не относится ни к одной Продаже автомобиля.



Рисунок 4 – ER-диаграмма Продажа автомобиля-Автомобиль

5) Связь «Для Продажи автомобиля заключается Договор»

Степень связи:

- Для Продажи автомобиля заключается только один Договор;
- Один Договор заключается только для одной Продажи автомо-

биля.

Класс принадлежности:

- Для Продажи автомобиля обязательно заключается Договор;
- Существует Договор, который не заключается ни для одной

Продажи автомобиля.



Рисунок 5 – ER-диаграмма Продажа автомобиля-Договор

б) Связь «Запчасть имеет Модель»

Степень связи:

- Запчасть может иметь только одну Модель;
- Одну Модель могут иметь несколько Запчастей. Класс принадлежности:



Рисунок 6 – ER-диаграмма Запчасть-Модель

7) Связь «Запчасть имеет Тип кузова»

Степень связи:

- Запчасть может иметь только один Тип кузова;
- Один Тип кузова могут иметь несколько Запчастей. Класс принадлежности:

- Запчасть обязательно имеет какой-то Тип кузова;
- Существует Тип кузова, который не имеет ни одна Запчасть.



Рисунок 7 – ER-диаграмма Запчасть-Тип кузова

8) Связь «Запчасть имеет Вид запчасти»

Степень связи:

- Запчасть может иметь только один Вид запчасти;
- Один Вид запчасти могут иметь несколько Запчастей. Класс принадлежности:

- Запчасть обязательно имеет какой-то Вид запчасти;
- Существует Вид запчасти, который не имеет ни одна Запчасть.



Рисунок 8 – ER-диаграмма Запчасть-Вид запчасти

## 9) Связь «К Продаже запчастей относится Запчасть»

Степень связи:

- К Продаже запчастей может относиться несколько Запчастей;
- Одна Запчасть может относиться только к одной Продаже запчастей. Класс принадлежности:
- К Продаже запчастей обязательно относится какая-то Запчасть;
- Существует Запчасть, которая не относится ни к одной Продаже запчастей.



Рисунок 9 – ER-диаграмма Продажа запчастей-Запчасть

## 10) Связь «Для Продажи запчастей заключается Договор»

Степень связи:

- Для Продажи запчастей заключается только один Договор;
- Один Договор заключается только для одной Продажи запчастей. Класс принадлежности:
- Для Продажи запчастей обязательно заключается Договор;
- Существует Договор, который не заключается ни для одной Продажи запчастей.



Рисунок 10 – ER-диаграмма Продажа запчастей-Договор

## 11) Связь «Клиент заключает Договор»

Степень связи:

- Клиент может заключить несколько Договоров;
- Один Договор заключается только одним Клиентом. Класс принадлежности:
- Клиент обязательно заключает какой-то Договор;

- Договор обязательно заключается каким-то Клиентом.



Рисунок 11 – ER-диаграмма Клиент-Договор

## 12) Связь «Менеджер заключает Договор»

Степень связи:

- Менеджер может заключить несколько Договоров;
- Один Договор заключается только одним Менеджером.

Класс принадлежности:

- Существует Менеджер, не заключивший ни одного Договора;
- Договор обязательно заключается каким-то Менеджером.



Рисунок 12 – ER-диаграмма Менеджер-Договор

На основе сегментов ER-диаграммы построена общая ER-диаграмма (Рисунок 13).

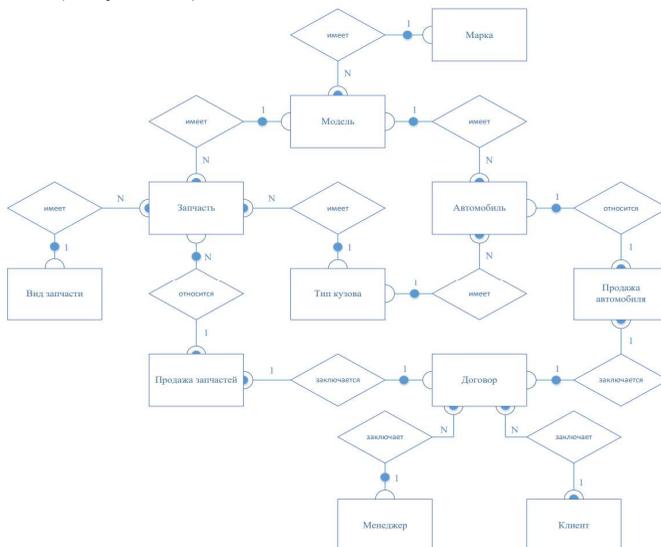


Рисунок 13 – Общая ER-диаграмма

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. М. Грабер. Введение в SQL. Изд. Лори, 2004. – 382с.
2. Гринченко Н. Н. Конспект лекций по дисциплине «БД и Клиент серверные приложения», 2021.
3. Ицик Бен-Ган Microsoft SQL Server 2008 Основы T-SQL:Пер. с англ. – СПб:БВХ-Петербург, 2009. – 432с.:ил.

УДК 004.4

**Д.Ю. ОСИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## САМ ПРОГРАММЫ В ПРОИЗВОДСТВЕННОМ ПРОЦЕССЕ

*Рассматривается процесс интеграции программ САМ в производственный процесс, способы их применения и их возможности.*

Computer-aided manufacturing (САМ-системы) – системы обработки и проектирования продукции на станках с числовым программным управлением (ЧПУ).

Сейчас САПР конструкторско-технологического проектирования гибких многослойных печатных плат (ГМП) основываются на модульном построении САПР, а так же на использовании общих баз данных на всех шагах производственного процесса. Такие программы имеют большие возможности моделирования технологических процессов и контроля производства.

Вопреки всем преимуществам, автономные САПР не могут решить все задачи, которые необходимо решить инженерам на этапе технологической подготовки производства гибких многослойных печатных плат (ГМП). Со временем, появилась необходимость использования сквозных интегрированных САПР, имеющие различные алгоритмы создания отдельных проектных процедур, а т.ж позволяющие обеспечить гибкую настройку по требованиям технологии производства ГМП. Сам по себе, термин «интегрированные САПР» подразумевает пакеты, позволяющие выполнять задачи САД/САМ/САЕ/РДМ систем автоматизированного проектирования (рисунок 1).

Из-за того, что требуется оперативно перестраиваться под потребности рынка необходимо уменьшать затраты времени на разработку и производство ГМП. В реализации этой задачи помогают САМ/САД/ЕДА -системы.

Существующие на данный момент системы автоматизированного проектирования и технологической подготовки производства ГМП являются сложными информационно-программными продуктами, требующими от исполнителя определенных навыков.



Рисунок 1 – САПР

Инженер во время этапа конструкторско-технологического проектирования должен иметь опыт работы во всех видах современных САПР (CAM/CAD/EDA), знать технические нормы и требования к оборудованию на производстве, иметь точные представления о сложности процесса технологической подготовки и производства ГМП.

Рассмотрим процесс (рисунок 2) совместной работы САМ/САD/ЕDА-систем в процессе конструкторско-технологического проектирования ГМП. По схеме видно, что первой идет разработка электрической схемы устройства, после чего начинается процесс проектирования печатного узла в ЕDА-системе. На третьем этапе проектирования ГМП выполняются все основные стадии технологического и конструкторского проектирования при помощи САD/САМ-систем. На выходе получаются готовые наборы конструкторской документации (КД) и комплекты управляющих файлов для станков с ЧПУ на производстве ГМП.

Особенно важным и исключительно ответственным этапом проектирования ПП становится технологическая подготовка уже разработанного конструкторского проекта ГМП к промышленному производству. Под этим подразумевается создание управляющих файлов для производства фотошаблонов, сверления отверстий на станках с ЧПУ, автоматизированной прозвонки плат на автоматах и установки компонентов на автоматических линиях. Как правило, все комплексы проектирования печатных плат имеют встроенные способы создания управляющих файлов, но, не смотря на это, существует ряд задач, которые

для технологии ГМП следует выполнять в специально предназначенных для этого программах.

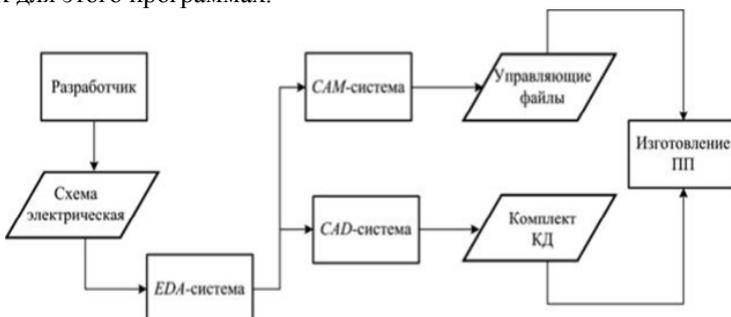


Рисунок 2 – Процесс работы САМ/САД/ЕДА-систем

Для выполнения этапа технологической подготовки производства ГМП необходимы средства для выполнения особенно качественной технологической подготовки производства на основе использования самых передовых методов и алгоритмов моделирования и оптимизации технологий. Поэтому, не смотря на ценовые характеристики предлагаемых к использованию САМ-систем, стоит при выборе технологической САПР остановиться на системе InCAM Frontline, обладающей лучшими техническими характеристиками в классе аналогичных систем.

В процессе конструкторско-технологическом проектировании ГМП одним из решающих считается требование сохранения целостности данных на протяжении всего пути проектирования (рисунок 3) - от выдачи исходного технического задания заказчиком до выпуска управляющих программ для оборудования с использованием технологии “ЧПУ” синтезированных интегрированной САПР ГМП.

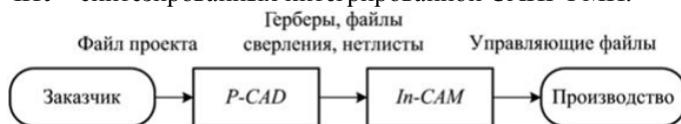


Рисунок 3 – Маршрут данных

Процесс конструкторско-технологического проектирования и производства ГМП на приборостроительном предприятии содержит следующие этапы:

- заключение договора с заказчиком;
- конструкторская проработка проекта ГМП с использованием P-CAD;
- технологическая подготовка проекта ГМП с использованием в технологическом цикле инструментальных средств САМ-станции InCAM;

-процесс автоматизированного производства ГМП.

Основываясь на проведенных исследованиях и анализе современной практики проектирования и производства ГМП предлагается общий алгоритм автоматизированной технологической подготовки производства ГМП с использованием системы InCAM.

П.1 Внесение исходных данных, которые были получены в системах проектирования печатных плат (OrCAD, Protel, P-CAD, PowerPCB и т.д). При этом необходимо данные из CAD программ конвертировать в форматы ODB++, HPGL, Gerber и другие.

П.2 Анализация и подготовка проекта с учетом критериев технолога.

П.3 Экспорт сформированного пакета управляющих файлов и технологической документации для оборудования с ЧПУ в цеха автоматизированного производства ГМП.

Данный обобщенный алгоритм технологической подготовки ГМП, который основывается на использовании современных САМ-систем, таких как InCAM Frontline, все же не дает возможности полной автоматизации всего процесса конструкторско-технологической подготовки производства ГМП. Это ограничение вызвано оригинальностью конструкции ГМП и сложностью технологии изготовления с использованием самого современного технологического оборудования. Развитие и улучшение приложений САПР InCAM в соответствии с новыми требованиями заключается в написании специальных программных модулей, которые будут оптимизировать файлы системы InCAM в форматы станков с ЧПУ, которые задействованы в технологическом процессе производства ГМП (рисунок 4).

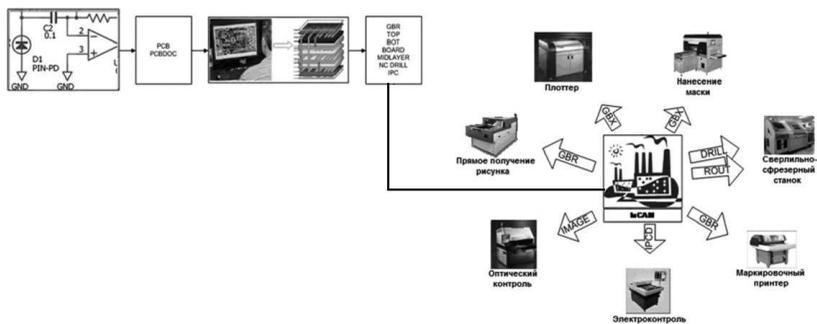


Рисунок 4 – Поток данных при технологической подготовке производства ГМП

Данная модернизация САПР InCAM позволяет во много раз снизить время выполнения типовых операций САПР и уменьшает возможность появления ошибок из-за человеческого фактора.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Мылов Г.В., Таганов А.И. Основы автоматизации конструкторско-технологического проектирования гибких многослойных печатных плат. // РГРТУ. – 2015.

2. Мылов Г.В., Таганов А.И. Методические основы автоматизации конструкторско-технологического проектирования гибких многослойных печатных плат. // РГРТУ. – 2014.

УДК 004.852

### **Н.Е. ОБЛАКОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ В ОЦЕНКЕ КРЕДИТОСПОСОБНОСТИ ЗАЕМЩИКА**

*Рассматривается задача оценки кредитоспособности заемщика путем построения модели на основе машинного обучения.*

С каждым годом растет популярность потребительских кредитов среди населения. Ранее оценкой кредитоспособности клиента должен был заниматься человек, но в последние годы благодаря развитию информационных технологий и методов машинного обучения для данной задачи достаточно разработать модель прогнозирования кредитоспособности, на основе исходных данных. Роль человека сводится к обработке исходных данных и загрузке их в нейронную сеть для дальнейшего анализа.

### **ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

#### **Постановка задачи**

Кредитоспособность заемщика – это возможность человека своевременно, с заданной регулярностью выплачивать взносы по текущему кредиту. Данная оценка является результатом анализа каждой заявки на выдачу кредита и именно на ее основе принимается решение о выдаче или отказе. Кредитоспособность анализируется на основе исходных данных о клиенте: кредитной истории, наличии стабильного заработка, материальном положении, наличии залогового имущества и

т.д.

В данной работе ставится задача разработки нейронной сети для автоматической оценки кредитоспособности клиента, после его обращения. Для реализации используем библиотеки python – Pandas и sklearn. В конечном счете необходимо подготовить исходные данные, создать и обучить нейронную сеть, оценить полученную модель, предсказать решения по кредиту.

## РЕАЛИЗАЦИЯ

### Подготовка данных

Данные для обучения нейронной сети содержатся в CSV-файле в виде таблицы на 1000 строк и содержит следующие столбцы:

DealId – номер сделки,

CreditSumm – сумма кредита,

Rating – рейтинг заемщика,

CreditHistory – кредитная история,

Verification – результат верификации,

VoteCI – вердикт кредитного инспектора,

VoteCM – вердикт кредитного менеджера,

PledgeSumm – сумма залога,

Decision – решение о выдаче кредита.

Сначала необходимо прочитать исходные данные с помощью библиотеки pandas удостовериться в их корректном отображении при помощи метода `.head()`, который позволяет вывести первые строки данных из импортированной таблицы.

|   | DealID | CreditSumm | Rating | CredHistory | Verifcation   | VoteCI | VoteCM | PledgeSumm | Decision |
|---|--------|------------|--------|-------------|---------------|--------|--------|------------|----------|
| 0 | 1      | 180000     | 7      | NaN         | отрицательный | NaN    | NaN    | 293000.0   | Отклонен |
| 1 | 2      | 290000     | 5      | зеленая     | положительный | NaN    | NaN    | NaN        | Одобен   |
| 2 | 3      | 420000     | 8      | NaN         | положительный | NaN    | NaN    | 507000.0   | Одобен   |
| 3 | 4      | 900000     | 9      | зеленая     | отрицательный | NaN    | NaN    | NaN        | Отклонен |
| 4 | 5      | 60000      | 5      | жёлтая      | отрицательный | NaN    | NaN    | NaN        | Отклонён |

Рисунок 1 – Первые строки тестовых данных

Данные, имеющие разное представление, но одинаковый смысл необходимо привести к общему виду. В нашем случае слова «Отклонён» и «Отклонен» имеют одинаковое значение, тогда как для программы это две совершенно разные строки. Меняем буквы и проверяем изменения в таблице.

```
#function replace Letters
def replace_letters(x):
    if isinstance(x, numbers.Number):
        return x
    return x.replace('ë', 'e')

raw['CredHistory']=raw['CredHistory'].apply(replace_letters)
raw['Decision']=raw['Decision'].apply(replace_letters)
raw.head()
```

|   | DealID | CreditSumm | Rating | CredHistory | Verification | VoteCI | VoteCM | PledgeSumm | Decision |
|---|--------|------------|--------|-------------|--------------|--------|--------|------------|----------|
| 0 | 1      | 180000     | 7      | NaN         | 0            | NaN    | NaN    | 293000.0   | Отклонен |
| 1 | 2      | 290000     | 5      | зеленая     | 1            | NaN    | NaN    | NaN        | Одобен   |
| 2 | 3      | 420000     | 8      | NaN         | 1            | NaN    | NaN    | 507000.0   | Одобен   |
| 3 | 4      | 900000     | 9      | зеленая     | 0            | NaN    | NaN    | NaN        | Отклонен |
| 4 | 5      | 60000      | 5      | желтая      | 0            | NaN    | NaN    | NaN        | Отклонен |

Рисунок 2 – Приведение данных к общему виду

Все данные, представленные в нецифровом формате необходимо перевести в цифру. Например, строки «Одобен» и «Отклонен» заменяются на «1» и «0» соответственно.

```
def decision_to_digit(x):
    if x=="Одобен":
        return 1
    return 0
```

Рисунок 3 – Переведение решения о выдаче в цифровой формат

Для быстрой замены пустых (NaN) значений во всей таблице можно использовать метод `.fillna(0)`. Стоит оговориться, что данный метод нужно использовать с осторожностью. В некоторых случаях пустота несет в себе дополнительную информацию и не может просто быть заменена на ноль. Например, в данном случае необходимо сохранить информацию об отсутствии залога, иначе модель будет работать некорректно. Для этого можно сформировать дополнительную синтетическую колонку `IsPledgeExist`, хранящую информацию о том, был ли предоставлен залог в принципе.

```
def is_pledge_exist(x):
    if math.isnan(x):
        return 0
    return 1
```

```
raw['IsPledgeExist']=raw['PledgeSumm'].apply(is_pledge_exist)
raw.head()
```

|   | DealID | CreditSumm | Rating | CreditHistory | Verfication   | VoteCl | VoteCM | PledgeSumm | Decision | IsPledgeExist |
|---|--------|------------|--------|---------------|---------------|--------|--------|------------|----------|---------------|
| 0 | 1      | 180000     | 7      | NaN           | отрицательный | NaN    | NaN    | 293000.0   | 0        | 1             |
| 1 | 2      | 290000     | 5      | зеленая       | положительный | NaN    | NaN    | NaN        | 1        | 0             |
| 2 | 3      | 420000     | 8      | NaN           | положительный | NaN    | NaN    | 507000.0   | 1        | 1             |
| 3 | 4      | 900000     | 9      | зеленая       | отрицательный | NaN    | NaN    | NaN        | 0        | 0             |
| 4 | 5      | 60000      | 5      | желтая        | отрицательный | NaN    | NaN    | NaN        | 0        | 0             |

Рисунок 4 – Создание дополнительной колонки IsPledgeExist

### Обучение модели

Чтобы иметь возможность оценить полученную в результате обучения модель, нужно разбить общий массив данных на тренировочные и тестовые. Для эффективного обучения и проверки стоит отводить под тренировочные – 75% данных, а под тестовые – 25%. В итоге получится четыре набора данных: тренировочный input ( $X_{train}$ ), тренировочный result ( $y_{train}$ ), тестовый input ( $X_{test}$ ) и тестовый result ( $y_{test}$ ).

```
#import data splitter
from sklearn.model_selection import train_test_split
```

```
#split data
X_train, X_test, y_train, y_test = train_test_split(X, y)
print("X_train length = "+str(len(X_train)))
print("y_train length = "+str(len(y_train)))
print("X_test length = "+str(len(X_test)))
print("y_test length = "+str(len(y_test)))
```

```
X_train length = 750
y_train length = 750
X_test length = 250
y_test length = 250
```

Рисунок 5 – Создание четырех наборов данных для обучения модели

Нормализация пространства признаков работает «из коробки» в пакете `sklearn`. Настраиваем `StandartScaler` на тренировочных данных с помощью метода `fit`. Нормализуем входные данные с помощью метода `transform`.

```
#import data scaler
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()

#fit only to the training data
scaler.fit(X_train)

StandardScaler(copy=True, with_mean=True, with_std=True)

# Now apply the transformations to the data:
X_train = scaler.transform(X_train)
X_test = scaler.transform(X_test)
```

Рисунок 6 – Нормализация входных данных

Класс MLPClassifier из пакета sklearn представляет собой классическую реализацию нейронной сети. Всё, что нужно сделать, это создать объект с необходимыми параметрами и обучить сеть с помощью метода fit, принимающего на вход inputs и results. Создадим трёхслойную сеть, по 13 нейронов в каждом слое. Ограничим максимальное число итераций обучения, чтобы не ждать слишком долго. Подадим на вход тренировочные данные.

```
#import NN (Multi-Layer Perceptron Classifier model)
from sklearn.neural_network import MLPClassifier

#create NN with 3 Layers, each contains 13 neurons; maximum iterations 500
mlp = MLPClassifier(hidden_layer_sizes=(13,13,13),max_iter=500)

#train our NN
mlp.fit(X_train,y_train)

MLPClassifier(activation='relu', alpha=0.0001, batch_size='auto', beta_1=0.9,
beta_2=0.999, early_stopping=False, epsilon=1e-08,
hidden_layer_sizes=(13, 13, 13), learning_rate='constant',
learning_rate_init=0.001, max_iter=500, momentum=0.9,
nesterovs_momentum=True, power_t=0.5, random_state=None,
shuffle=True, solver='adam', tol=0.0001, validation_fraction=0.1,
verbose=False, warm_start=False)
```

Рисунок 7 – Создание нейронной сети

Чтобы проверить и оценить качество полученной модели, используем для предсказания подготовленные ранее тестовые данные. Передадим методу predict тестовый input (X\_test). Импортируем инструменты для оценки. Выведем отчёт classification\_report по качеству модели, который в качестве аргументов принимает эталонные результаты (y\_test) и предсказанные только что значения. Точность предсказаний у созданной модели составляет 97%.

```
#Do predictions on test data
predictions = mlp.predict(X_test)
```

```
#Import evaluation tools
from sklearn.metrics import classification_report, confusion_matrix
```

```
#evaluate our predictions
print(classification_report(y_test, predictions))
```

|             | precision | recall | f1-score | support |
|-------------|-----------|--------|----------|---------|
| 0           | 1.00      | 0.95   | 0.97     | 135     |
| 1           | 0.94      | 1.00   | 0.97     | 115     |
| avg / total | 0.97      | 0.97   | 0.97     | 250     |

Рисунок 8 – Отчет по качеству модели

После того, как нейронная сеть создана и обучена, можно приступать к предсказаниям по данным без ответов. Для этого необходимо загрузить файл с обработанными данными и запустить обученную модель.

```
#Compile results
res = pd.DataFrame({'id': raw['DealID'], 'answer': predictions});
res.head()
```

|   | id   | answer |
|---|------|--------|
| 0 | 1001 | 0      |
| 1 | 1002 | 1      |
| 2 | 1003 | 0      |
| 3 | 1004 | 0      |
| 4 | 1005 | 0      |

Рисунок 9 – Результат работы нейросети

### Заключение

Для решения задачи оценки кредитоспособности заемщика была создана и обучена нейронная сеть и проведена оценка качества и работоспособности. Нейронная сеть реализована на языке Python. Для работы необходимо загрузить заранее подготовленные данные о заемщиках и запустить НС.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Дистанционное банковское обслуживание. – 2010. – 328 с.
2. Александр Турбанов, Александр Тютюнник. Банковское дело.

Операции, технологии, управление. – М.: «Альпина Паблишер», 2010. – 688 с.

3. Центр исследований платёжных систем и расчётов. Мошенничество в платёжной сфере: Бизнес–энциклопедия. – М: Интеллектуальная литература, 2016. – 345 с.

4. Гудфеллоу Я., Бенджио И., Курвилль А. Глубокое обучение / пер. с англ. А.А. Слинкина. 2-е изд. испр. М.: ДМК Пресс, 2018. 652 с.

5. Кузнецов В. П. Нейронные сети: практический курс: учеб. пособие / В. П. Кузнецов. Рязань, 2014. 72 с.

УДК 004.896

**Д.А. ПЕРЕПЕЛКИН, К.В. АНИСИМОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ОБЗОР ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ МОДЕЛИРОВАНИЯ СИСТЕМ НЕЧЕТКОГО ЛОГИЧЕСКОГО ВЫВОДА**

*Рассматриваются несколько прикладных программ, позволяющих синтезировать математические модели систем нечеткого логического вывода.*

Теория нечетких множеств, предложенная профессором Калифорнийского университета Лотфи Заде, стала основой для такого раздела математики как нечеткая логика, которая, являясь обобщением классической логики, нашла широкое применение во многих сферах: банковской (оценка риска выдачи кредита), производственной (автоматизация производства), информационной (в системах управления информационной безопасностью) и других.

Одним из основных понятий нечеткой логики является система нечеткого логического вывода. Она состоит из совокупности алгоритмов, которую, в общем случае, представляют в виде следующего набора: алгоритм фаззификации, алгоритм агрегирования, алгоритм нечеткого логического вывода, алгоритм дефаззификации.

Для создания систем нечеткого логического вывода используется специализированное программное обеспечение, позволяющее разработчику абстрагироваться от работы внутренних алгоритмов системы и сконцентрироваться на поставленной задаче.

**MatLab Fuzzy Logic Designer**

MatLab Fuzzy Logic Designer – это приложение, поставляемое вместе с дополнением Fuzzy Logic Toolbox для пакета прикладных программ MatLab. Для запуска Fuzzy Logic Designer в главном окне MatLab необходимо открыть вкладку «APPS», и в списке выбрать соответствующее приложение (группа CONTROL SYSTEM DESIGN AND ANALYSIS). Также запустить приложение можно написав в командной строке `fuzzyLogicDesigner`. Откроется окно, изображенное на рисунке 1.

Для создания новой системы нечеткого логического вывода требуется на верхней панели выбрать **File** → **New FIS...** В выпадающем списке будет предложено выбрать тип системы нечеткого вывода: **Mamdani** (Мамдани) или **Sugeno** (Сугено).

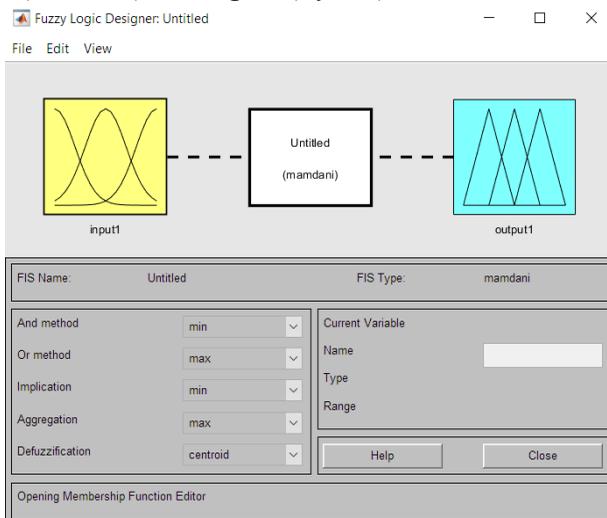


Рисунок 1 – Окно приложения Fuzzy Logic Designer

По умолчанию создается система типа Мамдани. После выбора типа системы снова откроется окно как на рисунке 1 с выбранным типом системы, который отображается в центральном прямоугольнике. Для системы можно выбрать методы, которые будут использоваться в качестве Т-норм (And method) и S-норм (Or method) а также методы импликации (Implication), агрегирования (Aggregation) и дефаззификации (Defuzzification). Доступных норм всего две: Заде и алгебраические. Среди доступных методов дефаззификации имеются усредняющие методы, такие как центра тяжести (centroid), центра площади (bisector), и методы, основанные на положении максимума. Это методы наименьшего максимума (som) среднего максимума (mom), наиболь-

шего максимума (lom). В литературе можно встретить и другие названия этих методов, такие как крайне левого, центрального и крайне правого максимума соответственно [1], [2].

Для добавления новой лингвистической переменной (ЛП) [3] необходимо на верхней панели выбрать **Edit** → **Add Variable...** и выбрать тип новой переменной: **Input** (входная) или **Output** (выходная).

Для редактирования ЛП на верхней панели надо выбрать **Edit** → **Membership Functions...** или два раза левой кнопкой мыши кликнуть по изображению той ЛП, которую требуется отредактировать. Откроется окно, изображенное на рисунке 2. Здесь можно выбрать тип функций принадлежности [4], [5] термов и изменить их параметры.

Для добавления новых термов необходимо на верхней панели нажать **Edit** → **Add MFs...**, выбрать тип функций принадлежности и количество новых термов.

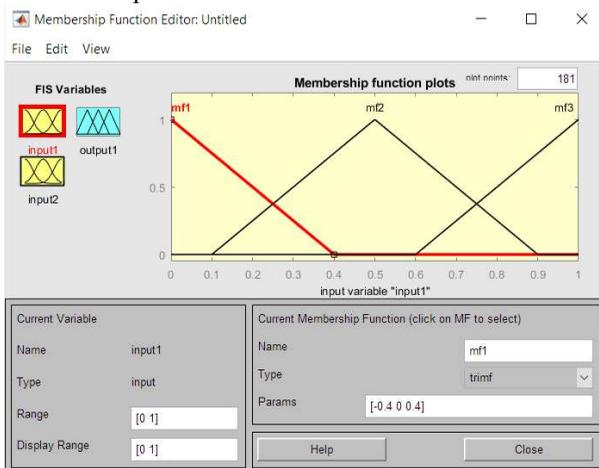


Рисунок 2 – Окно Membership Function Editor

После определения всех ЛП и их термов необходимо составить базу нечетких правил. Для этого на верхней панели надо выбрать **Edit** → **Rules...** или два раза кликнуть по центральному прямоугольнику с названием созданной системы нечеткого вывода. Откроется окно, в котором можно добавлять, изменять, удалять нечеткие правила. Для добавления нового правила достаточно выбрать по одному терму каждой ЛП и нажать **Add rule**. Агрегирование термов можно производить с помощью Т-нормы или S-нормы, к каждому выбранному терму можно применить операцию отрицания, установив флаг not.

Для просмотра поверхности отклика, синтезируемой созданной системой, на верхней панели надо нажать **View** → **Surface**. Откроется

просмотра поверхности (рисунок 4). В окне можно выбрать, носители каких ЛП будут располагаться по осям X, Y и Z.

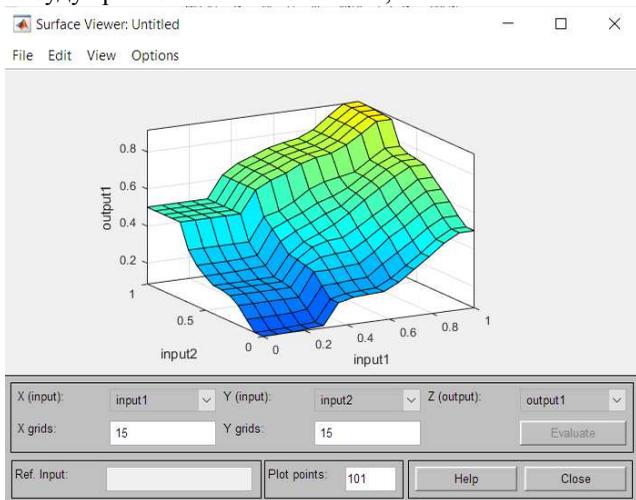


Рисунок 4 – Окно просмотра поверхности отклика

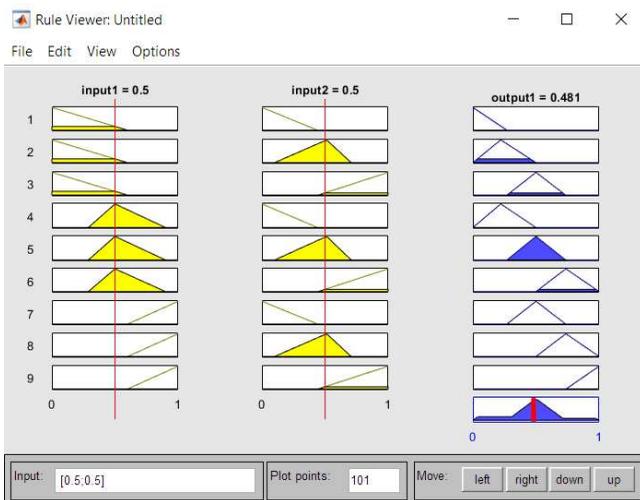


Рисунок 5 – Окно просмотра выполнения правил

Для того, чтобы посмотреть, как изменяется выходной сигнал в зависимости от входных необходимо на верхней панели нажать **View** → **Rules**. В открывшемся окне (рисунок 5) можно задавать значения

входных сигналов. Выходной сигнал будет изменяться соответствующим образом.

MatLab Fuzzy Logic Designer визуально привлекательная среда разработки систем нечеткого логического вывода с интерактивным и интуитивно понятным графическим интерфейсом. Встроенных инструментов вполне достаточно для построения большинства нечетких систем.

### Scilab fls Editor

Scilab – это свободно распространяемый пакет прикладных программ, аналог MatLab. В нем, тоже имеется специализированное дополнение для синтеза систем нечеткого логического вывода, которое называется аналогично дополнению в MatLab – Fuzzy Logic Loolbox и включает в себя приложение fls Editor. Для вызова приложения в командной строке Scilab необходимо набрать editfls. Откроется окно программы.

Для создания новой системы нечеткого логического вывода на верхней панели нужно нажать **File** → **New fls** и в раскрывшемся списке выбрать тип системы **Takagi – Sugeno** (Такаги – Сугено) или **Mamdani**. Справа появится новая система с названием NewName. Нажав на название системы, можно раскрыть список, состоящий из следующих наименований: Description, Inputs, Outputs, Rules. При нажатии на каждое наименование откроется соответствующий раздел.

В разделе Description задаются свойства системы: название (Name), тип T- и S-норм (T-Norm Class и S-Norm Class соответственно), тип отрицания (Complement), методы импликации (Implication Method), агрегирования (Aggregation Method), дефаззификации (Defuzzyfication Mathod). Стоит отметить, что здесь выбор предлагаемых методов шире, чем Fuzzy Logic Designer. Так в качестве T- и S- норм можно использовать нормы Заде, Эйнштейна, Ягера, Дюбуа, алгебраическая сумма (произведение).

Раздел Inputs предназначен для добавления, удаления и редактирования входных ЛП. Для добавления новой ЛП достаточно нажать на кнопку Add, которая располагается справа. Для ее редактирования можно нажать на кнопку Edit или выбрать соответствующую ЛП в списке слева. Откроется раздел, в котором можно производить манипуляции с термами редактируемой ЛП (рисунок 6).

Для добавления нового термина необходимо нажать на кнопку Add, находящуюся внизу, в появившейся строке ввести название термина (name), тип его функции принадлежности (type) и параметры (par).

Раздел Outputs аналогичен разделу Inputs, но в нем описываются выходные ЛП.

Раздел Rules предназначен для редактирования нечетких правил (рисунок 7). Его функционал аналогичен соответствующему окну в Fuzzy Logic Designer.

**sciFLT**

EDIT VARIABLE : \_\_\_\_\_

Information :

name : in1

range : 0 100

Nro. Member Function : 3

Member functions:

| name :                         | type : | par :            |
|--------------------------------|--------|------------------|
| <input type="checkbox"/> newmf | trimf  | 0.0 0.0 50.0     |
| <input type="checkbox"/> newmf | trimf  | 0.0 50.0 100.0   |
| <input type="checkbox"/> newmf | trimf  | 50.0 100.0 100.0 |

Рисунок 6 – Задание термов лингвистической переменной in1

RULE EDITION

```

R1: IF {in1 IS mf1} AND {in2 IS mf1} THEN {out1 IS f1} weighth=1
R2: IF {in1 IS mf1} AND {in2 IS mf2} THEN {out1 IS f2} weighth=1
R3: IF {in1 IS mf1} AND {in2 IS mf3} THEN {out1 IS f3} weighth=1
R4: IF {in1 IS mf2} AND {in2 IS mf1} THEN {out1 IS f2} weighth=1
R5: IF {in1 IS mf2} AND {in2 IS mf2} THEN {out1 IS f3} weighth=1
R6: IF {in1 IS mf2} AND {in2 IS mf3} THEN {out1 IS f4} weighth=1
R7: IF {in1 IS mf3} AND {in2 IS mf1} THEN {out1 IS f3} weighth=1
R8: IF {in1 IS mf3} AND {in2 IS mf2} THEN {out1 IS f4} weighth=1
R9: IF {in1 IS mf3} AND {in2 IS mf3} THEN {out1 IS f5} weighth=1

```

IF

in1 is: (0) ---- in2 is: (0) ---- out1 is: (0) ----

not  not  not

AND Weight: 1.0

OR

Delete rule Add rule Change rule

Рисунок 7 – Раздел редактирования нечетких правил

Для анализа работы созданной системы можно построить поверхность отклика или проверить работу правил, задавая конкретные значения входных сигналов. Для этого на верхнем меню надо нажать **View** и выбрать соответствующий инструмент **Surface** или **View Rules** (рисунок 8).

Программа Scilab fls Editor по своим функциональным возможностям практически аналогична Fuzzy Logic Designer, однако графический интерфейс менее привлекателен.

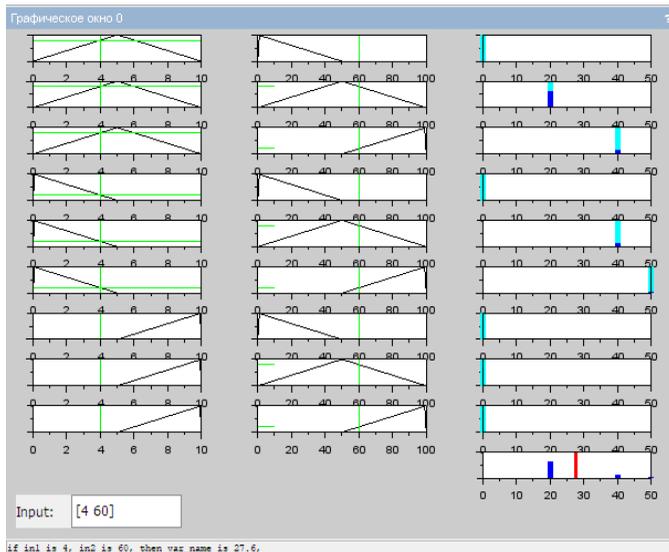


Рисунок 8 – Окно просмотра работы правил

### SimInTech

SimInTech – это среда, предназначенная для синтеза и анализа математических моделей различных технических систем. Так как нечеткая логика часто используется в автоматизированных системах управления, например [6], то разработчики позаботились о добавлении в программу инструментов нечеткого моделирования. Они располагаются во вкладке «Нечеткая логика» палитры компонентов. Библиотека компонентов нечеткой логики содержит блоки функций принадлежности, фаззификации, нечеткого вывода, T- и S-норм и нечеткого отрицания (рисунок 9).

В SimInTech нет отдельного приложения для создания системы нечеткого логического вывода, здесь она строится путем соединения соответствующих блоков между собой. Пример системы нечеткого логического вывода с двумя входными параметрами, созданной в SimInTech, представлен на рисунке 10.

Здесь значения входных сигналов задаются с помощью блоков «Константа». Сигналы подаются на два блока фаззификации, в которых определены термы ЛП. После фаззификации производится проце-

дура агрегирования с помощью блока «Минимум по входам». Далее сигналы, являющиеся степенями выполнения нечетких правил, передается в блок «Нечеткий вывод – Треугольная функция». Выходной сигнал системы нечеткого логического вывода подается на вход блока «Временной график», где может быть отображен.

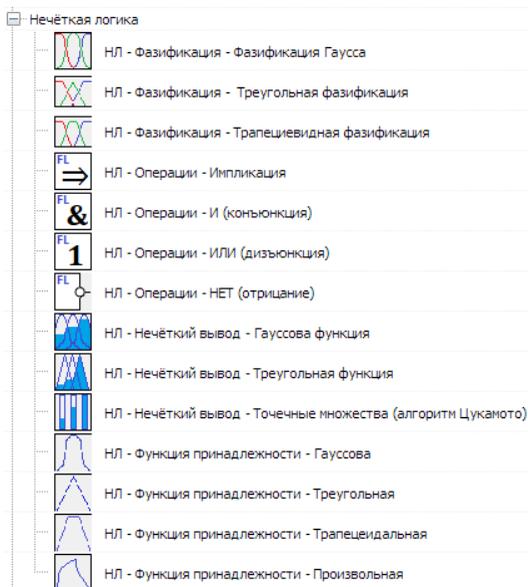


Рисунок 9 – Библиотека блоков для создания систем нечеткого логического вывода

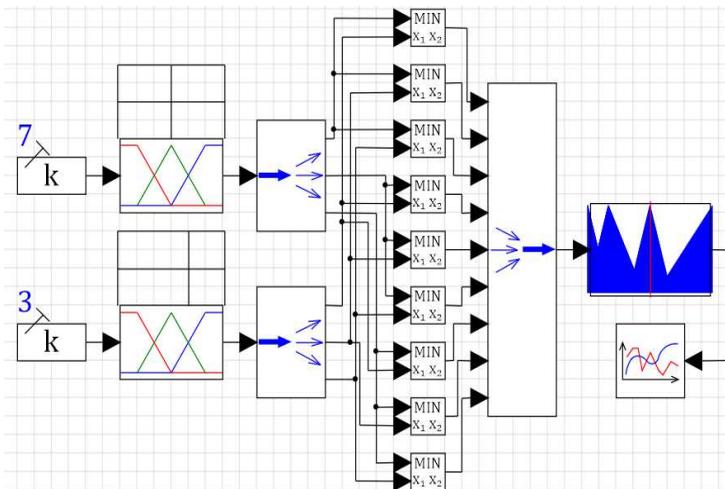


Рисунок 10 – Пример системы нечеткого логического вывода в среде SimInTech

Как видно из рисунка в данной среде проблематично создавать системы нечеткого логического вывода с количеством входных переменных более двух из-за необходимости процедуры агрегирования.

Сравним рассматриваемое программное обеспечение по количеству доступных алгоритмов для системы нечеткого логического вывода. Для этого составим таблицу 1.

Таблица 1 – Сравнение прикладных программ для синтеза систем нечеткого логического вывода

| Критерии сравнения                                | Название прикладной программы |           |           |
|---|-------------------------------|-----------|-----------|
|   | Fuzzy Logic Designer          | flsEditor | SimInTech |
| Количество доступных типов систем                 | 2                             | 2         | 2         |
| Количество доступных видов функций принадлежности | 11                            | 10        | 3         |
| Количество доступных норм                         | 2                             | 6         | 2         |
| Количество доступных типов импликации             | 2                             | 3         | 3         |
| Количество доступных методов дефаззификации       | 5                             | 5         | 1         |

Стоит отметить, что Fuzzy Logic Designer и SimInTech – это коммерческие продукты, а flsEditor – свободное программное обеспечение. Также ни в одной системе не учитывается математическая нечеткость функций принадлежности термов [7], что открывает пути к дальнейшему развитию программного обеспечения в этой области.

*Работа выполнена при финансовой поддержке гранта Президента РФ МД-3201.2022.1.6.*

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. ГОСТ Р МЭК 61131-7 – 2017. КОНТРОЛЛЕРЫ ПРОГРАММИРУЕМЫЕ. Часть 7. Программирование нечеткого управления - Москва: Изд-во ФГУП «СТАНДАРТИНФОРМ», 2017 – 53 с.
2. Анисимов К.В., Конюхов А.Н. Формирование навыков нечеткого моделирования в курсе «Основы теории нечетких множеств» // III Международный научно-технический форум СТНО-2020: сб. трудов / под ред. О.В. Миловзорова. Рязань: Рязан. гос. радиотехн. ун-т, 2020. Т. 10. С. 87-94.
3. Основы теории нечетких множеств. Часть 2: учеб. пособие / А.Н. Конюхов, А.Б. Дюбуа, А.С. Сафошкин; Рязан. гос. радиотехн. ун-т. – Рязань, 2018. – 108 с.
4. Основы теории нечетких множеств. Часть 1: учеб. пособие / А.Н. Конюхов, А.Б. Дюбуа, А.С. Сафошкин; Рязан. гос. радиотехн. ун-т. – Рязань, 2017. – 88 с.
5. Практикум по основам теории нечётких множеств и нечёткой логики. Часть 1. Нечёткие множества: учеб.-метод. пособие / К.В.Анисимов, А.Н.Конюхов; Рязан. гос. радиотехн. ун-т.– Рязань, 2022. – 68 с.
6. Анисимов К.В., Конюхов А.Н. Применение нечеткой логики в системе управления ориентацией малых спутников // IV Международный научно-технический форум СТНО-2021: сб. трудов / под ред. О.В. Миловзорова. Рязань: Рязан. гос. радиотехн. ун-т, 2021. Т. 3. С. 104-110.
7. Бухенский К.В., Конюхов А.Н., Дюбуа А.Б., Анисимов К.В. Промежуточные результаты исследований по проекту «трансформации нечеткости в системах нечеткого вывода» // IV Международный научно-технический форум СТНО-2021: сб. трудов / под ред. О.В. Миловзорова. Рязань: Рязан. гос. радиотехн. ун-т, 2021. Т. 5. С. 11-17.

УДК 004.772

**Д.А. ПЕРЕПЕЛКИН, Н.С. ПОТАПКИНА**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ УСТРОЙСТВ  
ИНТЕРНЕТА ВЕЩЕЙ НА ОСНОВЕ ТЕХНОЛОГИИ  
ПРОГРАММНО-КОНФИГУРИРУЕМЫХ СЕТЕЙ**

*Рассматриваются особенности проектирования устройства на основе платы Arduino Uno и модуля ESP8266, а также основные достоинства и области применения данного устройства. Также рассматриваются возможности применения технологии программно-конфигурируемых сетей для управления устройствами Интернета вещей.*

Программно-конфигурируемые сети (ПКС) – это архитектура, которая постоянно находится в развитии, чтобы подстраиваться под меняющиеся современные вычислительные системы и приложения [1].

Основное отличие ПКС от классических сетей состоит в том, что функции передачи данных и управления в сетевых устройствах отделены друг от друга. Данный вид сетей позволяет программировать функции управления сетью отдельно от структуры приложений. Управление организацией сетевых функций коммутаторов является главным отличием ПКС от классической сетевой архитектуры [2]. Этот вид управления не зависит от наличия конкретных сетевых соединений между сервером и устройствами (рисунок 1). Однако в ПКС отдельные устройства не принимают решения о передаче трафике [3].

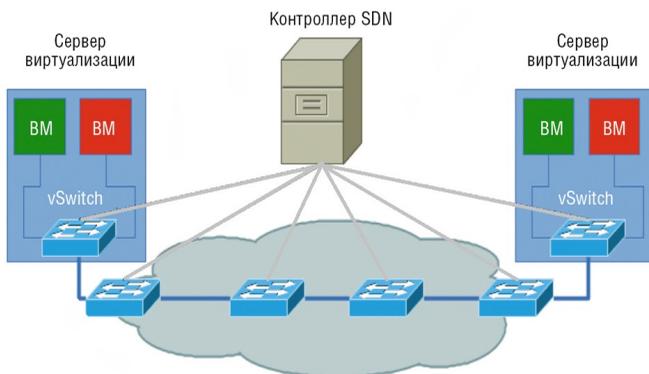


Рисунок 1 – Структура программно-конфигурируемой сети

Интернет вещей (Internet of Things, IoT) представляет собой совокупность физических или электронных устройств, подключенных к сети Интернет и имеющих разные размеры и возможности [4]. Устройства Интернета вещей подключаются к сети Интернет с использованием специальных сетевых протоколов, приложений, сетевых доменов, а также беспроводных технологий малого радиуса, сетей датчиков и т.д. (рисунок 2). На данном этапе развития IoT можно рассматривать как сеть сетей, в которой не только устройства, но и сферы применения устройств связаны между собой. Эти устройства поддерживаются в различных областях, таких как умные города, сфера здравоохранения, умные дома и т.д. [5].

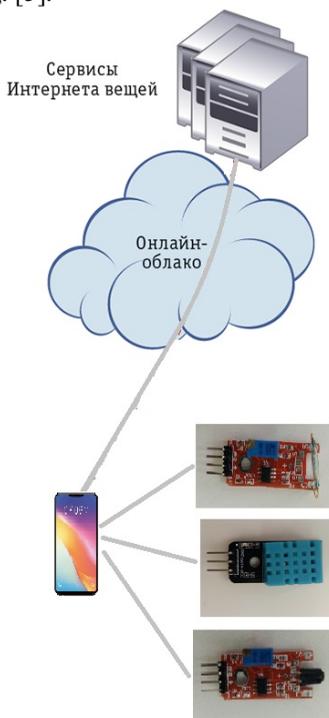


Рисунок 2 – Структура соединения устройств Интернета вещей

Использование ПК в сочетании с устройствами сети Интернета вещей позволяет разрабатывать системы контроля состояний конвейерных линий на промышленном производстве (контроль температуры, влажности, чистоты и т.д.). Все полученные данные можно собирать

на одном сервере, с помощью которого имеется возможность контролировать весь технологический процесс производства. Также ПКС применяются при разработках умных транспортных систем, умного дома, чтобы в любой момент из любой точки мира с доступом в сеть можно было не только выявлять наличие подключения сетевого устройства или датчика, но и управлять им [6].

В данной статье предложена модель на базе устройств Интернета вещей для наблюдения за влажностью и температурой в помещениях, контроля открытия/закрытия дверей и окон, наличия открытого огня и сигнализирования об этом. Сетевая инфраструктура между устройствами Интернета вещей построена на основе технологии ПКС. Информацию о состоянии датчиков и устройств предложено хранить на отдельном сервере с возможностью просмотра в режиме реального времени.

Для контроля открытия/закрытия дверей и окон в работе предлагается использовать герконовые датчики (герконы). Принцип работы датчиков основывается на изменении состояния двух разомкнутых или замкнутых контактов в колбе с инертным газом или с вакуумом под действием магнитного поля (магнита).

В зависимости от типа герконы могут быть замыкающими (под влиянием магнитного поля контакты в колбе замыкаются), размыкающими (под влиянием магнитного поля контакты в колбе размыкаются) и переключающими (по принципу работы ключа). Соответственно, устанавливая датчик и магнит на недвижимую и раскрывающуюся части дверей и окон, можно отслеживать наличие магнитного поля (в данном случае наличие/отсутствие магнитного поля будет сигнализировать об открытии/закрытии окон и дверей). Для приема сигнала с датчика применяется цифровой порт (рисунок 3).

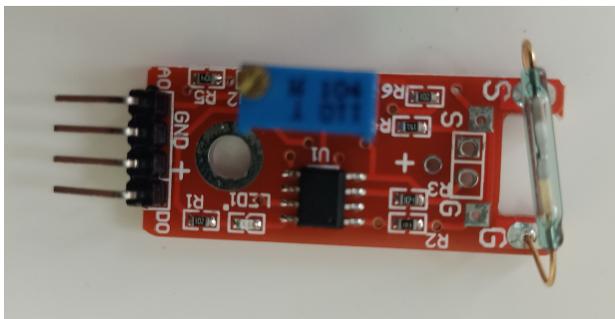


Рисунок 3 – Геркон

Кроме подстроечного резистора, используемого для настройки чувствительности, цифровой датчик геркона для Ардуино включает компаратор (микросхему) LM393. С помощью компаратора осуществляется генерация цифрового сигнала при срабатывании геркона. Соответственно, при наличии нуля, будет выдаваться сообщение «дверь/окно закрыто», при наличии единицы – открыто.

Датчик огня (flame sensor Arduino) позволяет установить наличие открытого пламени, используя инфракрасный приемник. На основе этого датчика создается пожарная сигнализация.

На датчике загорается индикатор, при этом при наличии нулевого сигнала выдается сообщение «пожар», если обнаружено пламя на расстоянии до 1 метра от сенсора, и «отсутствует открытый огонь» - при отсутствии пламени в пределах видимости инфракрасного приемника (рисунок 4).

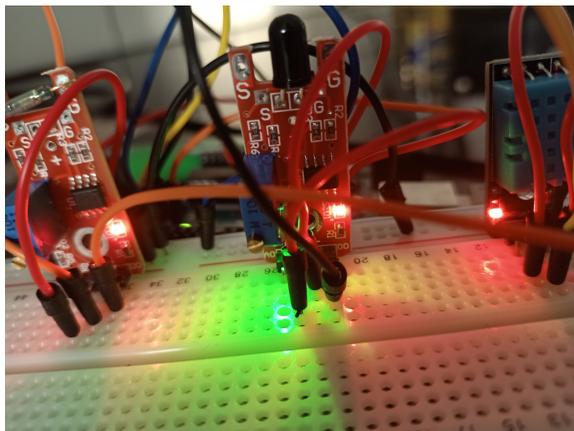


Рисунок 4 – Режим работы датчика огня

Аналоговый выход, находящийся на датчике, сообщает о наличии сигнала и его характеристике. Таким образом, благодаря сенсору можно определять как наличие открытого огня, так и то, настолько он сильный, а также площадь распространения пламени. Инфракрасный датчик реагирует на излучения в диапазоне 750 — 1100 нм [2].

Для отслеживания влажности и температуры в помещениях требуется соответствующий датчик, например, DHT22. Для измерения температуры на датчике находится термистор, для измерения влажности окружающего воздуха – емкостной датчик влажности. Для удобства отображения информации данный датчик может выпол-

нять аналого-цифровые преобразования и выдавать цифровой сигнал, который считывается с помощью микроконтроллера.

Термистор представляет собой специальный резистор, изготовленный на базе смешанных оксидов переходных металлов. Сопротивление термистора изменяется с изменением температуры, т.е. увеличение температуры окружающей среды приводит к падению сопротивления на датчике [3].

Емкостной датчик влажности является конденсатором с переменной емкостью. Конденсатор датчика находится в герметичном чехле с влагопоглощающим слоем, а также содержит токопроводящие обкладки из медной фольги на текстолите. При попадании частиц воды на этот слой, меняется его диэлектрическая проницаемость, и, соответственно, изменяется емкость конденсатора.

Датчик DHT22 (рисунок 5) имеет напряжение от 3 до 5 В, максимальный ток при преобразовании – 2,5 мА, интервал измерения влажности - 0% - 100%, интервал измерения температуры от -40 – до +125 градусов по Цельсию, период измерения - 2 секунды, частота – до 0,5 Гц.

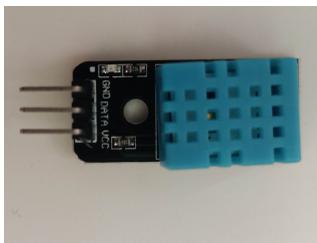


Рисунок 5 – Датчик DHT22

После загрузки соответствующей программы и подключения датчика, результат измерений можно просматривать в окне монитора порта. Там будут выводиться значения температуры и влажности, наличия открытого огня и открытия/закрытия дверей и окон.

Все полученные результаты с датчиков можно посмотреть также в окне монитора порта, сохранить и вывести на ЖК-экран, послать через bluetooth или по сети.

Для передачи собранных с датчиков данных через сеть Интернет предлагается использовать модуль ESP8266. Модуль позволяет легко подключить плату с датчиками к сети Интернет и предоставить возможность отправки получения данных по сети Интернет. После соединения контактов модуля ESP8266 с контактами платы Ардуино, остается только подключить модуль к сети с помощью Wi-Fi соединения. Выполнив физическое подключение, необходимо написать и загрузить

в плату соответствующий скетч, который превратит плату в сервер, отвечающий на запросы удаленных устройств или веб-клиента, собирающего данные с серверов или отправляющего на них эти данные (рисунок 6).

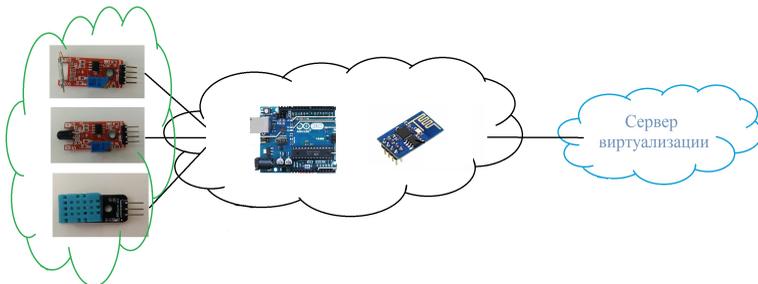


Рисунок 6 – Структура рассматриваемого устройства

*Работа выполнена при финансовой поддержке гранта Президента РФ МД-3201.2022.1.6.*

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Коннов А.Л., Анализ и проектирование программно-конфигурируемых сетей: учебное пособие / А.Л. Коннов – Оренбург: ОГУ, 2016. – 115 с.
2. Максимов Н.В., Компьютерные сети / Н.В. Максимов – Москва: Форум, 2013. – 464 с.
3. Кузьменко Н.Г., Компьютерные сети и сетевые технологии / Н.Г. Кузьменко – Санкт-Петербург: Наука и Техника, 2013. – 368 с.
4. Махмуд, О.А. Анализ структур сетей Интернета Вещей / О.А. Махмуд, О.А. Хуссейн, А.И. Парамонов // 73-я Всероссийская научно-техническая конференция, посвященная Дню радио – СПб.: СПбГЭУ «ЛЭТИ» им. В. И. Ульянова (Ленина), 2018. – С. 218–220.
5. Махмуд, О.А. Анализ развития приложений и изменения трафика Интернета Вещей / О.А. Махмуд, А.И. Парамонов // 74-я Всероссийская научнотехническая конференция, посвященная Дню радио – СПб.: СПбГЭУ «ЛЭТИ» им. В. И. Ульянова (Ленина). – 2019. – С. 158–159.
6. Боронин, П.Н. Интернет вещей как новая концепция развития сетей связи / П.Н. Боронин, А.Е. Кучерявый // Информационные технологии и телекоммуникации. – 2014. – № 3 (7). – С. 7–30.

УДК 004.451

**А.Н. ПЕРШИН**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**СИСТЕМНЫЕ ВЫЗОВЫ В ОС LINUX***Рассматриваются некоторые простейшие системные вызовы ОС Linux и их применение в программировании.*

В последнее время всё большую популярность приобретают устройства, которые работают под управлением ОС Linux, что приводит к растущему числу задач, которое должно на них выполняться, именно поэтому важно рассмотреть применение системных вызовов в программировании под ОС Linux. Так как это не обычные функции, то для передачи управления ядру необходима специальная подпрограмма, поэтому для упрощения их использования в GNU-библиотеке созданы специальные функции-оболочки. Совокупность системных вызовов позволяет прикладным программам обращаться к ядру системы.

В ОС Linux существует около 400 различных вызовов, данное число может варьироваться в зависимости от дистрибутива. Рассмотрим вызовы, которые часто используются при программировании, и начнём со связанных с чтением и записью.

Системный вызов `open()` используется для открытия файла и получения файлового дескриптора, который требуется для вызовов, которые используются в частности для операций чтения и записи. Есть два вида вызова:

```
int open (const char *name, int flag);  
int open (const char* name, int flag, mode_t mode);
```

Если функция смогла открыть файл, то возвращается целое положительное число, в противном случае возвращает `-1` и устанавливает `errno` в нужное значение ошибки. Аргумент `name` задаёт имя файла, `flag` представляет собой операцию поразрядного ИЛИ одного из режимов доступа (чтение, запись, чтение/запись) и модификатора доступа, `mode` задаёт режим доступа для созданного файла, если используется модификатор `O_CREATE` и файла с именем `name` не существует. Под именем файла подразумевается путь до файла и его название. Для комбинации `O_WRONLY | O_CREAT | O_TRUNC` в аргументе `flag`, сохранился специальный системный вызов `create()`, в настоящее время его можно легко реализовать в

пользовательском пространстве, благодаря наличию вызова `open` с тремя аргументами. Совместно с вызовом `open()` часто используются `read()` и `write()`.

Для описания системного вызова считывания `read()` следует привести его описание:

```
ssize_t read (int fd, void *buf, size_t len);
```

При каждом вызове `read()` считывает в `buf` не более `len` байт из файла с дескриптором `fd`. В случае успеха возвращает число байт, которое записалось в `buf`. Если возникает ошибка, то возвращается `-1` и в `errno` устанавливается соответствующее ей значение.

При работе с `read()` могут возникнуть ситуации, когда функция может вернуть целочисленное значение, которое меньше чем `len`, но больше нуля. Все считанные байты сохраняются в `buf`. Это чаще всего происходит при достижении конца файла или при прерывании процесса считывания сигналом. При следующем вызове, в котором соответствующим образом будут определены `buf` и `len`, данные запишутся в оставшуюся часть буфера `buf`.

Пример ниже показывает открытие файла `"/home/test.txt"`, а затем считывание `len` байт из него:

```
int fd;
fd = open ("/home/test.txt", O_RDONLY);
ssize_t ret;
while (len != 0 && (ret = read (fd, buf, len))!= 0)
{
    if (ret == -1)
        if (errno == EINTR)
            continue;
    perror ("read");
    break;}
len -= ret;
buf += ret;
}
```

Для записи информации используется системный вызов `write()`, его определение следующее:

```
ssize_t write(int fd,const void *buf,size_t count);
```

При вызове производится запись в файл, ассоциированный с дескриптором `fd`, некоторого количества байт начиная с `buf`. Максимальный размер записи определяет аргумент `count`. При успешной записи функция возвращает число байт, которые были записаны, противном случае возвращается `-1` и в `errno` записывается соответствующая ошибка.

Запись с помощью `write()` производится не сразу на диск, а в буфер ядра, это связано с разным быстродействием процессора и

жестких дисков, поэтому информация может просто не записаться на внешний носитель при аварийном отключении системы. Но благодаря такому механизму обеспечивается быстрое выполнение программы.

Простейшим примером применения вызова `write()` может быть запись в файл текстовой строки:

```
const char *buf = "Test syscall write";
ssize_t nr;
fd = open ("/home/test.txt", O_WRONLY);
nr = write (fd, buf, strlen (buf));
```

Если важно, чтобы данные точно попали на диск, то необходимо использовать системный вызов `fsync()`:

```
int fsync (int fd);
```

Вызов гарантирует что, все данные, которые связаны с файлом, будут записаны на диск. Принцип его работы в ожидании сообщения от жёсткого диска о том, что данные и метаданные были на него записаны. В случае успеха записи функция возвращает 0, а при ошибке возвращает -1.

В ОС Linux для чтения и записи также могут использоваться вызовы `pread()` и `pwrite()`:

```
ssize_t pread(int fd, void *buf, size_t count, off_t pos);
ssize_t pwrite(int fd, const void *buf, size_t count, off_t pos);
```

В отличие от стандартных вызовов `read()` и `write()` они получают файловую позицию в качестве аргумента и позволяют начинать чтение и запись с неё. Главное ограничение в том, что они применяются только для файловых дескрипторов, поддерживающих поиск, например обычных файлов. Возвращаемые значения и возможные ошибки, а также использование аналогичны функциям `read()` и `write()`, но в отличие от них после завершения работы не изменяют файловую позицию дескриптора.

При программировании в ОС Linux может возникнуть необходимость в уменьшении длины файла. Для этого используют системные вызовы `ftruncate()` и `truncate()`:

```
int ftruncate (int fd, off_t len);
int truncate (const char *path, off_t len);
```

Оба вызова выполняют одну задачу: усекают файл до длины `len`. Их различие в том, что первый работает с файловым дескриптором `fd`, а второй непосредственно с именем файла, записанным в `path`. Если размер файла был меньше `len`, то все данные, которые находились между концом файла и `len` удаляются, и становятся недоступными для считывания, а файл становится размером `len`. Если же размер файла

был меньше `len`, то происходит увеличение файла до длины `len`, а промежуток между концом файла и `len` заполняется нулями. Функции возвращают 0 в случае успешного выполнения и -1 при ошибке, записывая соответствующий код в `errno`. Рассмотрим пример усечения файла `test.txt` размером 19 байт, который содержит следующий текст:

```
Test syscall write.
```

Код ниже уменьшает размер файла до 5 байт:

```
int main()
{
    int fd, ret;
    fd = open ("/home/test.txt", O_WRONLY);
    ret = ftruncate (fd,5);
    if (ret == -1)
    {
        perror ("truncate");
        return -1;
    }
    return 0;
}
```

В результате работы программы, получается файл со следующим содержанием:

```
Test.
```

После завершения работы с файлом необходимо его закрыть. Для разрыва связи между дескриптором и файлом используется системный вызов `close()`:

```
int close(int fd);
```

Вызов разрывает связь между файлом и процессом, а также отменяет отображение дескриптора `fd`, который может снова использоваться ядром, в вызовах `open()` и `creat()`. Функция возвращает 0 при успешном завершении и -1 при неудаче.

Теперь рассмотрим системные вызовы для работы с процессами. В ОС Linux у каждого процесса есть свой уникальный `process id (pid)`, а также `ppid` у дочерних процессов. Для их получения используются системные вызовы `getpid()`, для получения `pid` процесса, и `getppid()` для получения `pid` родительского процесса.

Системный вызов `fork()`, используется для создания нового, идентичного текущему, процесса. Определение представлено ниже:

```
pid_t fork (void);
```

При успешном вызове создаётся новый процесс, текущий вызывающий становится родительским. Если `fork()` был вызван в родительском процессе, то функция возвращает `pid` дочернего, а если в дочернем, то 0. Простейший пример использования:

```
pid_t pid;
```

```
pid = fork ();
if (pid > 0)
    printf ("Родительский процесс cpid=%d\n", pid);
else if (!pid)
    printf ("Дочерний!\n");
else if (pid == -1)
    perror ("fork");
```

Чаще всего `fork()` используется для создания нового процесса с последующей загрузкой в него нового двоичного образа.

Описанные выше функции позволяют на начальном этапе разобраться в том, как системные вызовы используются в целом, и далее использовать полученную информацию для последующего изучения взаимодействия с ядром при прикладном программировании.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Лав Р. Linux. Системное программирование. 2-е изд. – СПб.: Питер, 2014. – 448 с.
2. Стивенс Р., Раго С. UNIX. Профессиональное программирование. 2-е изд. – СПб.: Символ-Плюс, 2007. – 1040 с.

УДК 536.248.2:532.529.5

### К.Ю. ПЕЩЕРОВ

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### КОГЕНЕРАЦИОННЫЕ СОЛНЕЧНЫЕ ЭЛЕКТРОТЕПЛОВЫЕ СТАНЦИИ

*Одной из главных задач при применении солнечных когенерационных станций является обеспечение потребностей электрической и тепловой энергией при использовании топливно-энергетических ресурсов в соответствии с требованиями по охране окружающей среды. Рассмотрены принципы построения когенерационных солнечных станций. Описана когенерационная солнечная электротепловая станция на базе отечественной солнечной энергетической установки КСЭУ2/5-30ТДК. Показано, что станции, построенные на основе когенерационных установок, по характеристикам электротехнических и электронных компонентов, входящих в их состав, имеют существенные отличительные особенности.*

### ВВЕДЕНИЕ

Солнечные электростанции (СЭС) уверенно завоёвывают «место под солнцем». Растёт их количество и разнообразие типов. В работе [1] приведено деление СЭС на 7 типов, в числе которых представлены

комбинированные СЭС, в которых согласно этой работе: «дополнительно устанавливаются теплообменные аппараты для получения горячей воды, которая используется как для технических нужд, так и для горячего водоснабжения и отопления». Получение горячей воды в таких СЭС, как правило, совмещено с процессом охлаждения солнечных элементов, что повышает в итоге их КПД.



Рисунок 1 – Фотография СФО



Рисунок 2 – Фотография СЭУ

Свыше 10 лет специалистами трёх предприятий (АО «НИИ вычислительных комплексов им. М.А. Карцева», г. Москва; ООО «СОЛЭКС-Р», г. Рязань; ООО «Технолидер», г. Рязань) разрабатываются и производятся установки на основе солнечных фотоэлектрических элементов (СФЭ). На рисунках 1, 2 представлены фотографии системы фотоэлектрического освещения (СФО) и солнечная энергетическая установка (СЭУ) с системой слежения в полярных координатах за солнцем, успешно прошедшие опытные испытания в НИИВК им. М.А. Карцева [2-4]. СФО эксплуатируются более 7 лет.

### Особенности построения СЭТС

За последние годы в мире значительно повысился интерес к когенерационным фотовольтаическим солнечным станциям, вырабатывающим одновременно электрическую и тепловую энергию. Этот интерес продиктован несколькими факторами.

1. В мире 95% коммерческих фотоэлектрических солнечных модулей (ФСМ) изготавливается на основе солнечных фотоэлектрических элементов (СФЭ) из кристаллического кремния. Эффективность СФЭ в таких ФСМ уже превышает 23%. Несмотря на то, что эта величина достаточно далека от оцениваемого сегодня теоретического предела КПД кремниевого СФЭ – 35%, дальнейший рост КПД коммерческих СФЭ сопряжен с большими финансовыми затратами, так как речь идет о создании принципиально новых промышленных технологий.

Так технология создания на поверхности кремния атомарного слоя оксинитрида гафния способна теоретически повысить КПД кремниевого СФЭ до 35% за счет передачи фотоном энергии не одно-

му, а двум электронам [8]. Однако сегодня эта технология находится на стадии теоретических исследований в мировых лабораториях. Перовскитные коммерческие СФЭ теоретически могут иметь КПД выше 25% при низкой себестоимости [9]. Но сегодня их долговечность низка, а пути ее повышения столь неопределенны, что говорить о скорой замене ими кристаллических СФЭ пока не приходится.

2. Современный кристаллический ФСМ из 100% падающей на него солнечной энергии только около 20% преобразует в электрическую энергию. Около 20% солнечной энергии от ФСМ отражается или проходит насквозь, а 60% превращается внутри ФСМ в тепло и теряется безвозвратно. Экономический выигрыш от эффективной утилизации тепловой энергии может оказаться гораздо выше, чем от повышения КПД СФЭ.

3. В подавляющем количестве домохозяйств, использующих для обеспечения своей жизнедеятельности электрическую энергию, в таких странах как Швеция, Норвегия, Дания, Республика Корея, Канада, США и в других странах, более 70% электроэнергии идёт на получение горячей воды. Применение для обеспечения таких объектов не просто солнечных установок, а СЭТС, вырабатывающих одновременно электрическую и тепловую энергию, существенно улучшает технико-экономические показатели использования солнечной энергии.

4. В последнее время в мире появились промышленные системы хранения тепла, в которых потери тепла, накопленного в летний период, и отдача его потребителю зимой не превышают 10%. Это обстоятельство значительно подняло спрос на солнечные системы в северных странах. Так в настоящее время в Финляндии объявлен конкурс на создание системы обеспечения теплом г. Хельсинки из альтернативных источников, и в первую очередь за счет солнечной энергии, с доведением объема выработки тепловой энергии к 2035 г до 1ТВт х час в год.

5. Несмотря на очевидные и существенные преимущества, СЭТС на рынке сегодня отсутствуют. Причина кроется в конструкции и технологии КСЭУ. С инженерной точки зрения оказалось технически сложно создать КСЭУ, стоимость производимой которой энергии была бы ниже стоимости энергии от обычной солнечной установки. Из наиболее продвинутых проектов в мире в области создания КСЭУ, можно выделить проекты компаний Cogenra Solar (входит в состав SunPower, США, 100% дочка Total в США по солнечной энергетике), компаний Airlight Energy, Швейцария и IBM, США. В статье [6] описаны некоторые из перечисленных проектов.

### Когенерационная солнечная станция на основе КСЭУ2/5-30ТДК

ООО «СОЛЭКС-Р», АО «НИИВК им. М.А. Карцева» и ООО «Технолидер» провели ряд исследований, в результате которых были определены структура и характеристики СЭТС. На рисунке 3 приведена структурная схема СЭТС, спроектированная для обслуживания котельной в пионерском лагере под Рязанью.

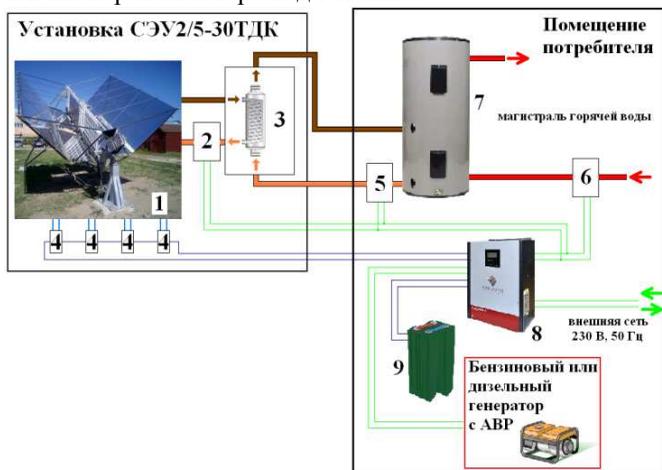


Рисунок 3 – Структурная схема когенерационной солнечной электротепловой станции на базе КСЭУ2/5-30ТДК

На рисунке 3 приняты обозначения:

- 1 – КСЭУ2/5-30ТДК
- 2 – насос КСЭУ2/5-30ТДК: UPML 25-105 130 PWN, GRUNDFOS
- 3 – теплообменник КСЭУ2/5-30ТДК: CB30-18H, Alfa Laval
- 4 – оптимизаторы мощности OM
- 5 – насос UPS25-40K 180, GRUNDFOS
- 6 – насосная группа потребителя 9 шт.
- 7 – бак аккумуляторный SOLAR SS 300
- 8 – инвертор Sunways Hybrid EVO 5048
- 9 – блок аккумуляторов из 32 шт. LT-LFP170B

СЭТС, построенные на основе когенерационных СЭУ, с точки зрения характеристик электротехнических и электронных компонентов, входящих в их состав, имеют ряд отличительных особенностей от имеющихся на рынке аналогов.

1. В КСЭУ2/5-30ТДК при концентрации солнечной энергии  $G=3$  и двусторонней чувствительностью СЭ каждый ФСМ при стандартных условиях имеет мощность около 600 Вт при токе 6 А. При малом ко-

личестве ФСМ в КСЭУ (4 шт.), а также при наличии зеркального концентратора, вызывающего в течение суток неравномерное освещение и локальные затенения поверхности ФСМ, обязательным условием эффективного функционирования СЭУ является наличие в каждом ФСМ оптимизаторов мощности (ОМ).

Одной из отличительных особенностей ФСМ является сильная зависимость напряжений и токов на их выходе от интенсивности солнечного излучения, температуры окружающей среды, режима работы нагрузки. Так как для электропитания большинства известных видов приемников электрической энергии необходимо стабильное напряжение, то преобразователи энергии должны обеспечивать требуемое качество, выполняя при этом функцию отбора максимальной мощности от солнечных модулей при разных сочетаниях внешних условий. Такой режим известен как поиск точки максимальной мощности или МРРТ (Maximum Power Point Tracking), который в настоящее время является основной функцией большинства устройств преобразования для фотовольтаических систем.

Устанавливаемые ОМ в ФСМ данной КСЭУ помимо режима МРРТ должны обеспечивать независимое суммирование вырабатываемой мощности каждым ФСМ. Такой режим работы повышает вырабатываемую электрическую энергию в течение светового дня более чем на 20%. Создание ОМ для КСЭУ с требуемыми параметрами по напряжению и токам, а также приемлемым КПД не менее 93% и низкой себестоимостью является одной из основных задач разработки. На рисунке 4 приведена структурная схема подключения ОМ в КСЭУ.

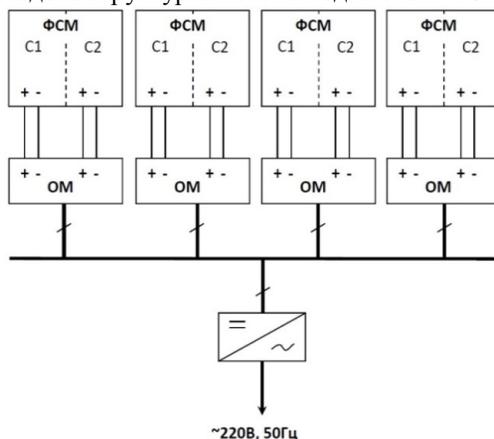


Рисунок 4 – Структурная схема подключения ОМ в КСЭУ

2. Инвертор при взаимодействии со всеми устройствами КСЭУ, в первую очередь с ОМ и контроллером, должен формировать напряжение переменного тока с приемлемым качеством электрической энергии.

3. Система контроля и управления зарядом/разрядом литиевых аккумуляторных батарей должна включать в себя устройства пассивных и активных балансиров.

4. В КСЭУ необходимо устройство управления (системный контроллер), решающее следующие задачи:

- управление производительностью циркуляционного насоса;
- формирование сигналов управления принудительным поворотом КСЭУ в крайнее положение при проведении регламентных работ, а также при нештатных ситуациях для исключения критического перегрева ФСМ;
- сбор, обработка и анализ состояния КСЭУ с выдачей контрольных сигналов.

### **Области применения СЭТС**

На сегодняшний день одной из главных задач при применении СЭТС стоит задача обеспечения потребностей в тепловой, а так же электрической энергии при использовании топливно-энергетических ресурсов. Многократно повторяемый тезис о том, что топливная энергетика приведёт уже через несколько десятилетий к катастрофическим изменениям жизни на нашей планете, должен был бы убедить человечество в необходимости повсеместного использования СЭТС.

Преимущества применения СЭТС многогранны. В плане экономичности есть возможность полного или частичного перехода предприятия или жилья с централизованного электро и теплоснабжения энергией на местное, со снижением затрат, в частности, на передачу электрической и тепловой энергии. Достаточно выгодным в плане энергосбережения является соединение СЭТС и абсорбционной охлаждающей установкой для производства электроэнергии, теплоты и холода (тригенерация). Появляется уникальная возможность продажи лишней вырабатываемой энергии при низкой себестоимости тепловой и электрической энергии, в сравнении с системами централизованного тепло и электроснабжения, - дополнительная экономическая целесообразность использования СЭТС. Затраты при применении СЭТС компенсируются за счет низкой себестоимости энергии в целом.

СЭТС, в значительной степени затребованы при создании систем бесперебойного электропитания (СБЭП), в частности, в СБЭП для гражданских и военных объектов, например, малых аэродромов и пограничных застав. В работе [10] рассматриваются СБЭП, в которых

источниками первичного электропитания применяются СЭТС и промышленная сеть электроснабжения. На рисунке 6 приведена структурная схема комплекса силовой электроники такой СБЭП, выполненной на основе модулей силового концентратора (МСК) [11], где  $U_{вх}$  - напряжение постоянного тока от СЭТС,  $U_{вх}$  - напряжение переменного тока от сети электроснабжения,  $U_{вых}$  - выходное напряжение переменного тока с МСК,  $U_{вых}$  - выходное напряжение постоянного тока с МСК. Наличие выходных напряжений с МСК переменного и постоянного тока различных значений обеспечивает требования потребителей электроэнергии объекта.

СЭТС на базе КСЭУ2/5-30ТДК обладает высоким экспортным потенциалом. По результатам маркетинговых исследований, выполненных зарубежными специалистами, объем рынка СЭТС оценивается более 20000 шт. со средним темпом наполнения 3000 шт. в год. Для продвижения СЭТС на международные рынки по КСЭУ2/5-30ТДК получены патенты: патент РФ в 2015 г., патент США в 2018 г., патент Республики Корея в 2019 г. Заканчивается патентование в Европейском патентном ведомстве.

В феврале 2020 года проект с использованием данной установки получил одобрение авторитетного швейцарского фонда Solar Impulse в качестве одного из 1000 лучших решений для защиты окружающей среды. По расчетам фонда одна установка снижает ежегодные выбросы углекислого газа в зависимости от региона размещения установки от 4 до 6 тонн за год.

### ЗАКЛЮЧЕНИЕ

1. Основная цель статьи – аргументация важности и необходимости повсеместного внедрения в народное хозяйство солнечной энергетики с вариантом перспективного направления в виде когенерационных солнечных электротепловых станций (СЭТС), Применение СЭТС по сравнению с обычными солнечными установками и солнечными коллекторами имеет существенную экономическую эффективность.

2. Анализ особенностей построения СЭТС позволил провести разработку установки КСЭУ2/5-30ТДК с высокими расчётными технико-экономическими показателями. Макетный образец КСЭУ2/5-30ТДК, в основном, подтвердил данные показатели.

3. Создание СЭТС на основе КСЭУ2/5-30ТДК должно привести к появлению на рынке солнечной энергетики нового класса эффективных солнечных установок, имеющих российский приоритет, подтверждённый рядом патентов [12-14].

4. Обзор вариантов применения СЭС даёт представление об их многообразии. Пример включения СЭС в состав системы бесперебойного электропитания (СБЭП) ответственного объекта является обоснованным, как с целью повышения надёжности функционирования электронного и электротехнического оборудования объекта, так и по возможности его обеспечением теплом.

5. Продолжительное сотрудничество предприятий АО «НИИ вычислительных комплексов им. М.А. Карцева», ООО «СОЛЭКС-Р» и ООО «Технолидер» в области разработок устройств и систем солнечной энергетики обеспечило достаточный задел для проведения опытно-конструкторской работы и последующей организации серийного производства СЭС и КСЭУ2/5-30ТДК.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Солнечная электростанция [Электронный ресурс]. – Режим доступа: [https://electric220.ru/news/solnechnaja\\_ehlektrostancija\\_schs/2019-04-07-1674](https://electric220.ru/news/solnechnaja_ehlektrostancija_schs/2019-04-07-1674).

2. Евтюхин А.С., Иванов А.В., Колосов В.А., Мухтарулин В.С., Парфёнов А.В., Худыш А.И. Солнечные электростанции // Электропитание. 2012. №1. С.11-19.

3. Иванов А.В., Колосов В.А., Мухтарулин В.С., Парфёнов А.В., Худыш А.И. Перспективы применения солнечных электростанций для бесперебойного электропитания информационных систем // Электропитание. 2012. №4. С.48-51.

4. Анисимов А.С., Жуковский С.И., Колосов В.А., Парфёнов А.В. Сравнительная характеристика солнечных энергетических установок // Электропитание. 2014. №1. С.45-46.

5. Иванов А.В., Худыш А.И., Парфёнов А.В. Когенерационные солнечные энергетические установки с повышенными удельными характеристиками // Электропитание. 2018. №1. С.52-59.

6. Колосов В.А., Парфёнов А.В., Саблин В.А., Худыш А.И. Когенерационные фотовольтаические солнечные системы // Электрическое питание. 2020. №1.

7. Иванов А.В., Ошкин Е.В., Худыш А.И. Комбинированная концентраторная фотоэлектрическая установка. Патент РФ № 2583317 от 29.01.2015.

8. M. Einzinger, T. Wu, J. F. Kompalla, H. L. Smith, C. F. Perkinson, L. Nienhaus, S. Wiefhold, D. N. Congreve, A. Kahn, M. G. Bawendi, and M. A. Baldo, “Sensitization of silicon by singlet exciton fission in tetra-cene”, Nature 571, pp. 90–94 (2019).

9. Tong, Z. Song, D. H. Kim, X. Chen, C. Chen, A. F. Palmstrom, P. F. Ndione, M. O. Reese, S. P. Dunfield, O. G. Reid, J. Liu, F. Zhang, S. P. Harvey, Z. Li, S. T. Christensen, G. Teeter, D. Zhao, M. M. Al-Jassim, Maikel F. A. M. van Hest, M. C. Beard, S. E. Shaheen, J. J. Berry, Y. Yan, K. Zhu, “Carrier lifetimes of  $>1 \mu\text{s}$  in Sn-Pb perovskites enable efficient all-perovskite tandem solar cells”, Science Vol. 364, Issue 6439, pp. 475-479 (2019).

10. Колосов В.А. Адаптивный комплекс силовой электроники системы бесперебойного электропитания для информационно-вычислительных объектов // Вопросы радиоэлектроники. Сер. ЭВТ. Вып. 4. 2014. С.107-115.

11. Колосов В.А. Модуль силового концентратора для системы бесперебойного электропитания РЭА. Патент РФ № 2498474 от 21.06.2013.

12. Патент США 10.148.224 от 4 декабря 2018. Combined concentrator photovoltaic installation.

13. Патент Республики Корея 10-2026003 от 20 августа 2019.

14. Патент №3252944 Европейского патентного ведомства от 11 ноября 2020. Combined concentrator photovoltaic installation.

УДК 623.4

### **А.В. ПЫРКОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **РАСЧЕТ ЭФФЕКТИВНОЙ ПЛОЩАДИ РАССЕЙВАНИЯ**

*Рассматривается понятие эффективной площади рассеивания, его история, вопрос расчета ЭПР для некоторых объектов, а так же общие принципы радиолокации и идеи «незаметности» летательных аппаратов.*

В процессе разработки современных летательных аппаратов перед конструкторами стоят несколько главных задач:

- совершенствование лётных характеристик
- снижение заметности на экранах радаров
- снижение заметности в инфракрасном диапазоне
- унификация производства

Способность к обнаружению радаром оценивается специальным параметром объекта, называемым эффективной площадью рассеивания или сокращенно ЭПР. Этот параметр представляет собой способ-

ность цели отражать энергию, которая падает на неё. Измеряется ЭПР в квадратных метрах ( $m^2$ ) и в физическом смысле определяется по формуле:

$$\sigma [db] = 10 \log \frac{\sigma}{\sigma_0}$$

Так же можно сказать, что ЭПР это абстрактная модель, которая представляет собой площадь изотропного рассеивателя показывающего в области нахождения предполагаемой цели такой же поток мощности на приемнике, что и у реальной воздушной цели. Отсюда можно сделать вывод, что если на приемное устройство не придет ответный радиосигнал, то летательный объект не будет виден на экране радара.

Первым об идеи создать «невидимый» для радаров самолет высказался советский ученый Уфимцев Петр Яковлевич. В процессе работы в «Центральном научно-исследовательском радиотехническом институте имени академика А. И. Берга», он написал книгу, которая называлась «Метод краевых волн в физической теории дифракции». В этой книге ученый изложил математическую модель, позволяющую создать летательный аппарат, который полностью отражал бы направленные на него радиосигналы. Но его идею не восприняли всерьез, так как для создания такого самолета необходимо было бы полностью забыть об аэродинамике и лётно-технических характеристиках. Так же в тот момент не существовало вычислительной машины необходимой мощности для расчета математической модели и системы управления такой сложной машиной.

Через какое-то время в США инициировали создание комитета «Хсом» по исследованию возможности создания самолета с использованием технологии «стелс». В процессе работы этого комитета было очень много попыток и ошибок, но решающей стала попытка Дэниса Оверхаузера создать программу расчета формы планера самолета на основе математической модели из книги Уфимцева «Метод краевых волн в физической теории дифракции». Этот шаг стал прорывным и через какое-то время в небо поднялся тактический малозаметный бомбардировщик F-117, в конечном итоге обязанный появлением советскому ученому.



Рисунок 1 – Тактический малозаметный бомбардировщик F-117

Непосредственно аналитические выражения для расчета ЭПР могут быть получены только для тел простой формы. Для тел со сложной геометрической формы в качестве средств измерения ЭПР применяются натурные испытания, а так же методы математического или физического моделирования.

Расчет ЭПР для простых форм производится по следующим формулам:

- $\sigma = \pi r^2$  – для шара, где  $r$ - радиус объекта.
- $\sigma = \frac{\pi d h^2}{\lambda}$  – для цилиндра, где  $d$ - диаметр основания,  $h$ - высота объекта,  $\lambda$ - длина волны облучающего сигнала.
- $\sigma = \frac{4\pi d s^2}{\lambda}$  - для металлической пластины, где  $s$ - площадь объекта.

Расчет ЭПР для уголковых отражателей (уголковый отражатель это три перпендикулярно расположенных поверхности, применяемые в качестве ложных целей или радио-контрастных ориентиров):

- $\sigma = \frac{4\pi}{3\lambda} a^4$  - для уголковых отражателей с треугольными гранями.
- $\sigma = \frac{4\pi}{3\lambda} 3a^4$  - для уголковых отражателей с четырехугольными гранями.

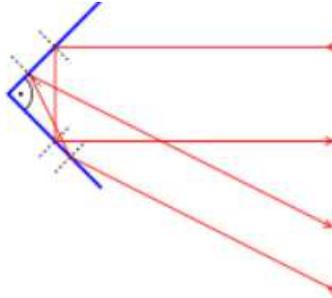


Рисунок 2 – Схема работы углового отражателя

Так как рассматриваемые объект довольно малы по отношению к длине волны облучающей РЛС, такие формулы дают незначительную ошибку. Чаще всего ЭПР таких тел измеряются не в  $\text{м}^2$ , а в  $\text{дБм}^2$  (это отношение децибелов от  $\text{м}^2$ ). Например, у металлической сферы, радиусом 5 мм ЭПР, определенная по первой формуле будет равна  $-41 \text{ дБм}^2$ .

Для точечных целей ЭПР рассчитывается по формуле:

$$\sigma = 4\pi r^2 \frac{E_2^2}{E_1^2}$$

$$E_2 = \frac{1}{\lambda} \int_s \frac{E_1}{r} \exp(-j * 2kr) \cos\theta dS$$

, где  $k$  – волновое число, а поле поверхности выглядит следующем образом:

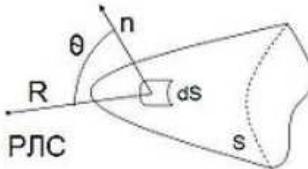


Рисунок 3 – Поле поверхности (s)

$$E_2 = \frac{E_1}{r\lambda} \exp(-j * 2kr_0) \int_s \frac{E_1}{r} \exp(-j * 2kr) \cos\theta dS$$

$$\frac{E_2}{E_1} = \frac{1}{r\lambda} e^{-2jk r_0} \int_s \frac{E_1}{r} e^{-2jk r} \cos\theta dS$$

Тогда формула расчета ЭПР для точечных целей будет выглядеть следующим образом:

$$\sigma = \frac{4\pi}{\lambda^2} \left| \int_s \frac{E_1}{r} e^{-j \frac{2\pi}{\lambda} r_0} \cos\theta dS \right|^2$$

Ниже в таблице приведены рассчитанные значения ЭПР точечных сложных целей:

Таблица 1.

| N  | Тип цели                                    | $\sigma_{ц} [M^2]$ |
|----|---|--------------------|
| 1  | Самолёт истребитель                         | 3-5                |
| 2  | Фронтовой бомбардировщик                    | 7-10               |
| 3  | Тяжёлый бомбардировщик                      | 13-20              |
| 4  | Бомбардировщик В-52                         | 40                 |
| 5  | Транспортный самолёт                        | 40-70              |
| 6  | "Стелс"                                     | 0,0001..0,01       |
| 7  | Бомбардировщик, В-2В                        | 0,75               |
| 8  | Многоцелевой истребитель F-22               | 0,1-0,08           |
| 9  | Подводная лодка в надводном положении       | 30-150             |
| 10 | Рубка подводной лодки в надводном положении | 1-2                |
| 11 | Малые суда (до 200 тонн)                    | 50-200             |
| 12 | Средние корабли (1000..10000 тонн)          | $(3-10)^2$         |
| 13 | Большие корабли (больше 10000 тонн)         | $> 10^2$           |
| 14 | Крейсер                                     | 12000-14000        |
| 15 | Автомобиль                                  | 1-3                |
| 16 | Крылатая ракета                             | 0,1-0,3            |
| 17 | Человек                                     | 0,8-1              |
| 18 | Птица в полёте                              | 0,795              |

В наше время в разработке находятся множество самолетов «невидимок» из них можно выделить ПАК ДА российского ОКБ Туполева и японский TD-X Shinshin. Некоторые самолеты, совмещающие приемлемые лётные характеристики и малозаметность уже эксплуатируются (российский СУ-57 и американский F-22 Raptor). «Стелс» технология используется не только в авиации, но и в кораблестроении. Например, в России эксплуатируются «Корветы проекта 20380».

В будущем, с развитием вычислительной техники станет возможной разработка программного обеспечения для расчета ЭПР более сложных форм планера. Это даст толчок к созданию нового поколения истребителей совмещающих «стелс» технологию и хорошие лётно-технические характеристики.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. «Метод краевых волн в физической теории дифракции», Уфимцев П. Я., изд. «Советское радио», 1962, тираж 6.5 т. экз.
2. Russian «Godfather of Stealth» at UCLA. – Engineer of California, 1992, v. 45, № 3, March, p. 10.
3. А. Н. Лагарьков, М. А. Погосян. Фундаментальные и прикладные проблемы стелс-технологий // Вестник РАН. – 2003. – Т. 73, № 9. – С. 848.
4. Heppenheimer, T. A. Stealth: First glimpses of the invisible aircraft now under construction. // Popular Science. – September 1986. – Vol. 229 – No. 3 – P. 74-79, 115-116 – ISSN 0161-7370.

УДК 681.3

**Д.В. РАДУЧИЧ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **ВЫБОР КОНСТРУКЦИИ УСТРОЙСТВА ПРИЕМА, ПЕРЕДАЧИ И ПРЕОБРАЗОВАНИЯ ТЕЛЕМЕТРИЧЕСКОЙ ИНФОРМАЦИИ**

*Рассматривается современный подход проектирования электронных средств, который показан на примере конкретного устройства*

В настоящее время в проектировании электронных средств (ЭС) применяется иерархический принцип. Согласно этому принципу, сложные устройства или системы состоят из более простых и типовых устройств. Начнем с того, что следует разделять ЭС по конструктивной и функциональной сложности. Эти две классификации разделяют две непосредственно связанные части ЭС как готового изделия – конструкцию и электрическую схему.

На рисунке 1 показаны функциональные уровни ЭС. Рассмотрим каждый уровень.

Радиоэлектронная (РЭ) система – это совокупность автономных радиоэлектронных комплексов и устройств. В случае изменения условий эксплуатации радиоэлектронные комплексы и устройства могут изменить свою структуру для решения поставленной задачи.

Радиоэлектронный комплекс (РЭК) – это радиоэлектронное средство в виде функционально законченной совокупности радиоэлектронных устройств. В случае изменения условий эксплуатации уст-

ройства могут изменить свою структуру для решения поставленной задачи.

Радиоэлектронное устройство (РЭУ) – это электронное средство, которое представляет собой функционально законченную сборочную единицу. Оно предусматривает функции приема, передачи и преобразования информации и выполнена на несущей конструкции.

Радиоэлектронный функциональный узел (РЭФУ) отличается от РЭУ тем, что не имеет самостоятельного применения.

Радиоэлементы (РЭ) и электротехнические элементы (ЭЭ) – элементы нулевого уровня функциональной иерархии. Входят в РЭФУ или напрямую в РЭУ.

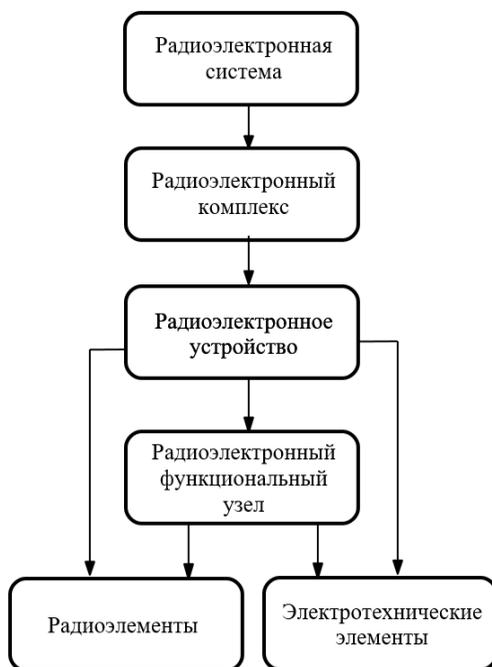


Рисунок 1 – Функциональные уровни ЭС

На рисунке 2 показаны конструктивные уровни ЭС. Рассмотрим каждый уровень.

Изделия электронной техники (ИЭТ) и электротехнические изделия (ЭТИ) образуют нулевой уровень иерархии. Данные изделия входят в радиоэлектронные ячейки или кассеты, основой которых является несущая конструкция первого уровня (НК1). Радиоэлектронные

блоки или рамы выполнены на основе несущей конструкции второго уровня (НК2). Они могут включать в себя как ЭС первого уровня иерархии, так и ЭС нулевого уровня иерархии.

В радиоэлектронные шкафы, пульта или стойки могут входить ЭС первого и второго уровня иерархии, они выполнены на основе несущей конструкции третьего уровня (НК3).

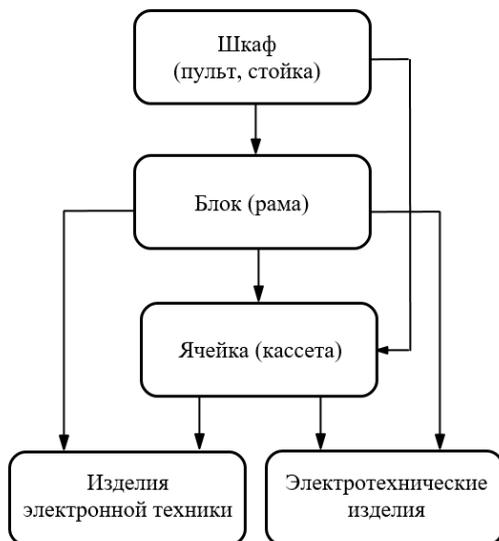


Рисунок 2 – Конструктивные уровни ЭС

Применим вышеупомянутый современный принцип проектирования ЭС к устройству приема, передачи и обработки телеметрической информации.

Разрабатываемое нами устройство будет состоять из несущего конструктива (каркаса), четырех ячеек и модуля питания. Конструктив устройства будет исполнен в виде блока 19” высотой 6U и должен соответствовать стандарту «Евромеханика».

«Евромеханика» - это модульный конструктив, служащий для размещения стандартных кассет, плат и конструкций, которые унифицированы по высоте, ширине и глубине.

Размер 19 дюймов (482,6 мм), базовый параметр в евроконструктивах, указывает на ширину панели блока. Единицы U указывают на высоту блока, в свою очередь U равна  $1\frac{3}{4}$  дюйма (44,45 мм).

У изделий, выполненных по стандарту «Евромеханика», есть ряд очевидных достоинств:

- благодаря вертикальному расположению ячеек наблюдается хорошая вентиляция и поддержание приемлемого теплового режима;
- ремонтпригодность и удобство обслуживания;
- универсальность;
- приемлемые цены и т.д.

На передней панели устройства будут находиться такие разъемы как:

- вилки в качестве выходов локальной вычислительной сети;
- технологические розетки;
- контрольные гнезда для контроля выходных напряжений 5 В;
- контрольные гнезда для контроля выходного напряжения 27 В.

В.

В задней панели устройства будут находиться следующие разъемы:

- клемма защитного заземления;
- розетка для обеспечения внешнего устройства напряжением электропитания 27 В;
- вилка для подключения к источнику переменного напряжения 220 В;
- вилка для приема сигналов единого времени;
- розетка для передачи сигналов единого времени;
- вилки для подключения к внешним источникам постоянного напряжения 27 В;
- розетки для приема ТМИ;
- вилки для имитационных выходов ТМИ.

Итак, в данной статье была затронута тема современного подхода проектирования электронных средств, а также на основании этого метода выбрана конструкция устройства.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Основы конструирования электронных средств: учеб. для студентов вузов / А.В.Зеленский, Г.Ф.Краснощекова. – Самара: Изд-во Самар. гос. аэрокосм. ун-та, 2014. – 228 с.
2. Радиоэлектронная аппаратура и основы её конструкторского проектирования / Н.И. Каленкович – Минск: БГИУР, 2008. – 200 с.

УДК 004.032.26

**П.А. САЛЫНОВ, С.В. СКВОРЦОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **СТРУКТУРНО-СЕМАНТИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ ВЕБ-СТРАНИЦ ДЛЯ ИЗВЛЕЧЕНИЯ ДАННЫХ ИЗ СЕТИ ИНТЕРНЕТ**

*В статье рассматривается проблема извлечения неструктурированных данных из источников в сети Интернет с использованием метода управляемой экстракции данных из веб-страниц. Предлагается методика построения шаблонов для выборки данных с использованием структурно-семантической кластеризации веб-страниц.*

В условиях активного развития современных интернет-технологий важность использования многих интернет-проектов зависит от их обеспечения необходимыми инструментами для сбора и обработки информации из веб-источников. Постоянно развивающиеся веб-технологии, а также растущее разнообразие форматов, типов и объёмов хранимой информации требуют совершенствования подходов решения вопроса извлечения необходимой информации из источников в сети Интернет.

На текущий момент времени для нахождения информации в сети Интернет активно применяется метод автоматического извлечения информации из источника с использованием шаблонов, построенных для выделения определённых параметров [1, 2].

В качестве основного способа построения большинства веб-сайтов используется подход, состоящий в использовании «повторяющихся структур данных». Повторяющиеся структуры, как правило, включают в себя: каскадные таблицы стилей CSS, стандартные форматы хранения различного контента (например, .FLV, .PNG, .PDF, .GIF), а также универсальные ссылки на ресурсы (URL-адреса).

Однако на данный момент современные методы автоматизированного извлечения данных на основе управляемой экстракции исследованы достаточно мало, что делает эту задачу достаточно актуальной. В данной задаче, существует несколько этапов, которые с учётом уровня развития современных IT-технологий, невозможно автоматизировать. К таким этапам относятся:

1) необходимость подбора списка основных страниц, используемых как основной пример для автоматического построения шаблонов извлечения данных;

2) необходимость выполнения разметки предварительно отобранных веб-страниц, используя для этого сторонние приложения.

Таким образом возникает необходимость совершенствования технологий извлечения данных, построенной на основе метода управляемой экстракции информации из веб-страниц, которая должна соответствовать следующим требованиям:

- 1) обеспечивать высокое качество шаблонов;
- 2) минимизировать участие человека в задаче настройки системы на требуемых сайтах.
- 3) обеспечивать возможность интеграции в существующие системы сбора структурируемой информации.

В данной работе предлагается методика автоматизированного построения шаблонов для выборки данных на основе структурно-семантической кластеризации интернет-страниц. Данная методика, включает в себя следующие этапы:

- 1) выбор администратором примеров веб-страниц сайта;
- 2) разметка администратором примеров веб-страниц;
- 3) кластеризация веб-страниц сайта;
- 4) выборка примеров веб-страниц из полученных кластеров;
- 5) построение шаблонов извлечения необходимой информации;
- 6) тестирование шаблонов.

Теоретически, при использовании данной методики, в первом пункте администратору, достаточно будет выбрать по одной веб-странице для каждого типа информации. Далее администратор вручную выполняет разметку полученных примеров веб-страниц сайта, с использованием специальной системы извлечения данных. При этом пользователю необходимо последовательно выделять все значимые, с точки зрения задачи, атрибуты и характеристики.

Процесс генерации шаблонов включает в себя два взаимосвязанные этапа:

- построение шаблонов атрибутов данных (формирование правил извлечения требуемой информации из веб-страниц);
- валидация шаблонов атрибутов данных (проверка качества шаблонов, полученных по итогам кластеризации элементов сайта, с использованием дополнительных примеров веб-страниц).

Такое разделение на 2 этапа вызвано тем, что данная методика построения шаблонов подразумевает использование только одного примера веб-страницы с требуемой информацией. Это позволяет уменьшить количество операций, выполняемых вручную.

В третьем пункте основным требованием к построению шаблонов атрибутов данных, остается соблюдение соответствия каждого атрибута отдельному шаблону.

Для автоматизированного построения шаблонов в системе извлечения данных используется «генератор шаблонов», включающий в себя следующие этапы:

1) представление структуры веб-страницы, выбранной пользователем в качестве примера, в виде элементов DOM-дерева;

2) поиск по структуре DOM-дерева вхождений требуемых атрибутов данных;

3) построение структурно-семантической модели шаблона для каждого атрибута данных и его значения. Такая модель представляет собой список, включающий следующие основные элементы:

– xpath-выражение, используемое для извлечения атрибута данных в структуре DOM-дерева;

– HTML-идентификатор или класс тега (`id="имя", class="имя"`), содержащего заданный администратором атрибут данных.

– содержимое тега, используемое непосредственно в процессе построения шаблонов атрибутов данных. Он задаётся в процессе разметки примера веб-страницы администратором системы.

Одним из важнейших факторов для успешной кластеризации веб-страниц является выбор метрик, используемых для определения степени сходства и различия полученных кластеров шаблонов между собой. Изменение метрики способно значительно повлиять на результаты кластеризации, из-за чего их зачастую определяют как базовый инструмент кластерного анализа.

В рамках предлагаемой методики построения шаблонов в качестве исходных данных для кластеризации предлагается рассматривать URL-адреса веб-страниц, основной характеристикой которых следует считать их структуру.

Учитывая, что URL-адреса представляют собой строки разной длины, в данном случае в качестве метрики целесообразно использовать расстояние Левенштейна, которое определяет количество перестановок символов, требуемых для превращения одной строки в другую.

Тестирование шаблонов для извлечения данных осуществляется непосредственно во время работы алгоритма кластеризации на основе отобранных веб-страниц. При этом основным фактором успешной кластеризации является получение такого списка шаблонов для каждого кластера, при котором извлечь информацию из веб-страницы выполняется в полном объеме. То есть для каждого определённого адми-

нистратором искомого атрибута, система должна вернуть корректное значение, даже в случае отсутствия самого атрибута.

По итогу процесса кластеризации мы получаем систему иерархически связанных кластеров, которая позволяет напрямую извлекать структурированные данные из отдельных подгрупп, не затрачивая время на поиск требуемой информации на каждой веб-странице сайта. Так же по завершению кластеризации формируется ограниченная группа дополнительных примеров веб-страниц. Это происходит по средством случайной выборки из наиболее подходящего класса.

Если же при использовании шаблона на странице не удаётся определить искомую информацию, то требуется уточнение параметров его структурно-семантической модели. При этом происходит переопределение xpath-выражения. В результате этой операции будет создан новый шаблон для извлечения данных, который также будет проходить процесс тестирования.

В статье рассмотрен подход, позволяющий выполнять управляемую экстракцию информации из веб-страниц в процессе извлечения структурированных данных из источников в сети Интернет. Предложена методика автоматического построения шаблонов для выборки данных с использованием структурно-семантической кластеризации, применение которой позволит избежать ручного описания этих шаблонов.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Чиркин Е. С. Некоторые проблемы автоматизированного извлечения данных из веб-страниц // Интернет и современное общество: сборник научных статей XVI Всерос. объединен. Санкт-Петербург, 2013. С. 291–294.

2. Вдовин И. В. Актуальные вопросы автоматического извлечения данных из веб-страниц // Перспективы развития информационных технологий: материалы XXIII Междунар. науч.-практич. конф. 30 января 2015 г. Новосибирск, 2015. С. 11–16.

3. Вдовин И. В. Применение технологии Web Mining к извлечению научных данных в сети Интернет // Информационные технологии в науке и производстве: материалы всерос. молодеж. науч.-техн. конф. Омск, 9–10 февраля 2015 г. Омск: Изд-во ОмГТУ, 2015. С. 10–14.

УДК 004.9

**А.О. САПРЫКИНА, М.С. ИГОШИНА**

Рязанский государственный университет имени С.А. Есенина,  
Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **СРАВНИТЕЛЬНЫЙ АНАЛИЗ СИСТЕМ ДИСТАНЦИОННОГО ОБУЧЕНИЯ**

*В статье приводится сравнительный анализ систем дистанционного обучения. Рассматриваются достоинства, недостатки российских и зарубежных СДО, кратко описываются возможности платформ.*

Системы дистанционного обучения (СДО) направлены на своевременное и легкодоступное тестирование (аттестацию) школьников, студентов, сотрудников из любой точки мира без каких-либо дополнительных затрат на передвижение.

СДО – это виртуальная обучающая платформа, наделенная инструментами, позволяющими создавать, управлять и контролировать образовательные курсы.

Выбор подобной системы зависит от вида поставленной задачи, которую СДО должна будет реализовать. Ключевыми критериями при определении необходимой среды могут стать:

- Стоимость и лицензионная политика;
- Доступность информации;
- Ведение статистики и отчетности;
- Простота и удобность интерфейса;
- Возможность коммуникации между педагогом и обучающимися;
- Возможность перенастройки системы, особенности редактирования и демонстрации учебных материалов.

На основе представленных выше факторов сравним некоторые наиболее востребованные российские и зарубежные платформы дистанционного обучения.

*iSpring Learn* является облачной платформой дистанционного обучения от компании *iSpring* – лидера на мировом рынке программ для создания электронных курсов.

СДО обладает простым интерфейсом, не требующим дополнительных временных затрат от администратора или пользователя на ознакомление.

Существует возможность редактировать дизайн личного аккаунта, например, изменение цветового оформления или логотипа, однако добавить новую функцию в систему невозможно.

Облачное хранилище поддерживает опцию загрузки неограниченного количества учебных видеороликов, аудиозаписей, презентаций, изображений и готовых курсов в форматах SCORM 1.2, SCORM 2004, AICC и Tin Can. Реализована функция создания интерактивных курсов, тестов и диалоговых тренажеров. Благодаря встроенному плееру возможно корректное отображение рабочих материалов на компьютерах и мобильных устройствах. Права доступа к обучающей информации легко перенастраиваются. Есть возможность организовать продажу курсов онлайн.

Администраторы системы могут выполнять следующие операции:

- Добавлять/удалять пользователей. Добавление происходит через email или с помощью импорта из CSV-файла.
- Редактировать информацию о пользователях, объединять их в группы для формирования групповых отчетов.

Кроме того, существует система рейтингов, очков, сертификатов, однако отсутствует календарь и расписание, позволяющие четко представлять планы обучения.

В пределах конкретного курса можно писать комментарии и вопросы, обмениваться файлами. Предусмотрена площадка для проведения вебинаров. Создание отдельных форумов, чатов, блогов и опросов в системе невозможно.

Данная СДО способна формировать 19 типов детальных отчетов по результатам дистанционного обучения, при этом нет возможности учитывать результаты очных занятий и конструктора для создания собственных форм отчетов.

Существует бесплатная пробная версия (14 дней). Для использования полного пакета возможностей среды необходимо приобрести подписку, стоимость которой составляет 56 000 рублей. Предусмотрена система скидок для учебных заведений (40%) и некоммерческих организаций (20%).

**Mirapolis LMS** является самодостаточным модулем, входящим в состав системы Mirapolis HCM, которая направлена на комплексную автоматизацию бизнес-процессов.

СДО разрабатывается индивидуально для каждого заказчика, на основе требуемого функционала и детально составленного технического задания. Полученную учебную платформу с уникальными возможностями можно изменять и масштабировать.

Среда способна загружать различные типы учебных данных, в том числе поддерживаются стандарты SCORM 1.2, SCORM 2004, AICC и Tin Can. Существует встроенный конструктор вопросов и тестов, которые возможно импортировать из XLS и QTI файлов. Для каждого учебного курса предусмотрено редактирование сценариев и условий его прохождения.

Так как СДО разрабатывается по индивидуальным требованиям заказчика, то система может быть оснащена различными инструментами коммутации пользователей, например, вебинары, чаты, форумы, блоги и пр.

Среда включает в себя более 120 шаблонов отчетов, а также конструктор, позволяющий извлекать из системы любые данные и создавать уникальные формы отчетов.

Стоимость платформы зависит от требуемых заказчиком функций, входящих в её состав. Минимальная стоимость проекта — 600 000 рублей в год.

**ShareKnowledge** – это система дистанционного обучения, в основе которой лежит площадка *Microsoft SharePoint*. Среда обеспечивает ведение учета, планирования смешанного обучения сотрудников, а также легко совместима с другими ИТ-системами компании.

Платформа оснащена простым и удобным интерфейсом. Администратор способен осуществлять такие функции, как настройка системы, управление дистанционным и очным обучением, пользователями и организациями.

В пользовательском каталоге СДО можно размещать документы, видео- и аудиоматериалы, а также электронные SCORM-курсы. Система позволяет создавать тесты и формировать составные курсы из учебных материалов, опубликованных в каталоге, очных тренингов и вебинаров. Отсутствует мобильная версия платформы.

Платформа оснащена функцией планирования обучения. Формировать расписание могут как руководители, назначая курсы обучающимся, так и сами сотрудники, подавая заявки на интересующие их курсы из каталога.

СДО поддерживает несколько базовых отчетов по сотрудникам, дистанционным курсам и очным тренингам. Также можно самостоятельно создавать и настраивать свои типы отчетов.

Данная среда является платной, её стоимость зависит от количества пользователей и дополнительных услуг по разработке. Минимальная стоимость сервиса — 24 000 рублей в год.

**Teachbase** – это сервис для организации дистанционного обучения, а также продажи онлайн-курсов. Отличительная черта системы

заключается в том, что она не поддерживает формат SCORM, однако работает с документами PowerPoint, Word, PDF, аудио- и видеофайлами, а также материалами видеохостинга YouTube. Оснащена площадкой для проведения и записи вебинаров.

Существуют минимальные опции для изменения среды, такие как добавление логотипа, выбор имени поддомена или возможность подключения своего домена.

Объем хранилища колеблется от 2 до 500 ГБ. Загружать в библиотеку можно учебные материалы в форматах ppt, pdf, video, MS Office. Есть встроенный редактор, позволяющий из готовых материалов формировать курсы и добавлять к ним тесты. Предусмотрена мобильная версия среды.

СДО обладает простым набором инструментов для управления пользователями, которых можно приглашать по email или импортировать из списка. Для формирования групп пользователей используются лейблы, которые работают по принципу тегов. Кроме того, поддерживается иерархия прав доступа: слушатель, специалист, менеджер.

Для обсуждения учебного материала в рамках курса можно обмениваться комментариями, также предусмотрена площадка для проведения вебинаров.

Платформа формирует 4 типа отчетов по критериям: слушатели, посещаемость, общая статистика и активные пользователи. Однако опция создавать свои макеты отчетов отсутствует.

Стоимость зависит от количества активных пользователей в системе, объема памяти в хранилище и периода подписки. Начальный тарифа на 30 пользователей и 2 ГБ памяти — 3200 рублей в месяц.

**WebTutor** — это платформа, состоящая из модулей, направленная на автоматизацию бизнес-процессов управления персоналом. Платформа способна создать корпоративную площадку, легко настраиваемую под индивидуальные требования заказчика. Благодаря этому на выходе получается система дистанционного обучения, учебно-информационный сервис или полноценный HR-портал.

Стандартный интерфейс СДО требует дополнительных временных затрат на изучение документации для корректной работы с ним.

Поддерживаются международные стандарты SCORM 2004, AICC и Tin Can, есть встроенный редактор тестов, а также собственный инструмент для разработки интерактивных электронных курсов — CourseLab, предусмотрена мобильная версия.

Модуль СДО «Персонал и организационная структура», позволяет добавлять, хранить, просматривать информацию о сотрудниках компании, их должностях и подразделениях.

С помощью модуля «Управление порталом» в систему можно добавить форумы, площадку для вебинаров, блоги, чаты, а также возможность проводить опросы.

Модуль «Учебный центр» обеспечивает предоставление аналитических отчетов о результатах обучения сотрудников и других данных системы, а также реализует функцию экспорта этих отчетов в формате Word, Excel, txt или HTML.

Стоимость системы зависит от количества используемых модулей и стоимости услуг по разработке и настройке системы под требования заказчика. Есть возможность аренды СДО. Стоимость минимальной версии СДО на 1000 пользователей, 1 ГБ памяти — 115 000 рублей в год.

**Docebo** является модульной средой для обучения сотрудников и продажи курсов. Основываясь на поставленных целях, пользователь может приобретать модули, расширяющие функционал системы. Отличительная особенность платформы — встроенный редактор учебного контента, позволяющий создавать простые программы обучения без использования дополнительных средств.

Платформа обладает простым интерфейсом, не требующим дополнительных IT-навыков. Поддерживается функция редактирования внешнего оформления, например, изменение цветовой схемы или логотипа. Администратор обладает правами масштабировать интерфейс для разных ролей, отбирать и размещать необходимые виджеты.

В СДО можно загрузить большое число учебных материалов в форматах PPT, PDF, docx, xls и video, электронные курсы в форматах SCORM, AICC и Tin Can. В системе не предусмотрен встроенный редактор дистанционных курсов, однако реализованы функции создания тестов и опросов. Просматривать учебные материалы можно офлайн в мобильном приложении на смартфонах и планшетах.

Администраторы способны осуществлять следующие операции:

- Добавлять пользователей через email, с помощью импорта из CSV- или XLS-файла или через SalesForce, если подключена интеграция с CRM-системой.
- Удалять пользователей, настраивать для них доступ, объединять в группы по различным критериям.
- Назначать курсы, как отдельным пользователям, так и целой группе учащихся.
- Добавлять в среду рейтинги, очки, сертификаты.

В платформе реализована функция, направленная на создание виртуальной социальной среды для общения сотрудников, назначения экспертов. Пользователи могут задавать вопросы экспертам, а также

предлагать контент, публиковать свои посты, оценивать и комментировать посты коллег.

СДО позволяет создавать 17 типов отчета: по пользователям, обучающим курсам, тестам, программам обучения и продажам. В том числе есть возможность учитывать результаты очных занятий.

В течение 14 дней можно бесплатно пользоваться демо-версией. Dosebo предлагает базовый тариф (от 300 пользователей с ограниченным функционалом) и расширенный тариф (от 500 пользователей с возможностью приобрести все модули). Каждый пользователь сверх выбранного тарифа оплачивается отдельно.

**Unicraft** — это облачная среда дистанционного обучения, поддерживающая встроенный редактор курсов.

Дизайн базовой версии системы менять нельзя, однако за дополнительную плату возможно расширение функционала по запросу пользователя.

СДО формирует уроки, состоящие из тестов, вопросов и экзаменов. Поддерживается выгрузка MP3, MP4 и PDF файлов, а также набор текста вручную, импорт видео с YouTube, Vimeo и файлов Google Docs.

Администратор способен назначать менеджеров для управления персоналом. Пользователей можно добавлять, удалять, редактировать о них информацию, записывать их на курсы и объединять в группы. Сотрудники добавляются вручную или путем импорта CSV-файла. Реализована система рейтингов и медалей. Пользователи получают уведомления о записи на курс только по email.

Платформа не поддерживает коммуникацию пользователей между собой.

Отчетность формируется путем экспорта CSV-файла, что также можно сделать с помощью меню групп.

СДО предлагает облачную подписку и коробочную версию. Коробочное решение выгоднее оформлять, если планируете закупать систему на долгий срок. Стоимость облачной версии при оплате за год колеблется в диапазоне 33 000 руб - 191 000 руб, в зависимости от количества пользователей.

**e.Queo** является средой, направленной на мобильное обучение, оптимальна для видеокурсов.

Интерфейс платформы интуитивно понятен: сотрудники без труда разберутся, как проходить курсы и тесты. Администратору не нужно читать документацию или иметь IT-навыки, чтобы настроить платформу. Отсутствует функция редактирования дизайна аккаунта самостоятельно.

В облачное хранилище можно загрузить неограниченное количество видеороликов, PDF файлов и курсов SCORM 1.2 и 2004. Во встроенном редакторе можно создавать видеокурсы и тесты. В тестах есть несколько видов вопросов: множественный выбор, выбор картинки. В видео можно добавлять интерактивные вставки (стоп-вопросы): выбор видео, множественный выбор, выбор картинки.

Администратор способен реализовывать следующие функции:

- Формирование группы пользователей, назначение руководителей для индивидуального управления. В системе 4 способа добавления пользователей: вручную, импорт файла .XLSX, генерация (автоматическое присвоение ID и пароля) и саморегистрация (пользователь сам задает логин и пароль).
- Просмотр статистики по курсам, тестам, опросам, разделам информации, достижениям и SCORM-курсам.
- Формирование рейтингов за счет накопительного свойства информации. Обнуление статистики, выгрузка в виде отчетов в формате Excel.

В системе предусмотрены быстрые переходы из приложения по ссылкам на другие платформы.

Платформа имеет облачную подписку и установку на сервер, стоимость которой колеблется от 1000 до 3000 рублей. Установка на сервер стоит от 3 000 000 рублей в зависимости от технических спецификаций заказчика.

***eTutorium LMS*** – одна из самых новых СДО на рынке. Платформа поддерживает весь минимально необходимый функционал для организации корпоративного онлайн-обучения: от создания различных учебных материалов до организации вебинаров и программ обучения.

Интерфейс системы простой, создание инфраструктуры курса занимает не более 15 минут, слушатели способны без специального инструктажа разобраться в личном кабинете и начать обучение.

Администратор системы может создавать персональные письма слушателям и преподавателям курса, генерировать сертификаты об успешном прохождении курса, используя свои макеты.

Реализована функция выгрузки в курс необходимого количества материалов любого формата, проработка кейса с помощью диалогового тренажера, использования видеофайлов, полученных на ресурсах Youtube или Vimeo. Система позволяет создавать тесты разной сложности и структуры, группировать их по тематике. Доступность разделов курса для слушателей и порядок их прохождения легко настраивается при планировании курса.

Данная СДО предусматривает иерархию ролей: администратор, модератор, преподаватель и студент. В зависимости от поставленных целей определенного курса администратор назначает права модератору. Преподаватель работает со студентами конкретного курса или группы, проверяет задания, отвечает на их вопросы. Обучающихся возможно объединять в группы и курсы. Добавлять пользователей в среду осуществляется через email или с помощью импорта из файла CSV.

Платформа способна в пределах отдельно взятого курса создавать форум/чат для коммутации обучающихся, где каждый студент обладает возможностью поддерживать обратную связь, задавать интересующие его вопросы преподавателю (модератору), за которым он закреплен.

Система реализует опцию формирования отчеты по различным элементам курса: отчеты о прохождении всего курса, модуля, теста (со статистикой по отдельным вопросам теста) или задания, а также аналитика по каждому студенту. Все формируемые отчеты можно скачать в формате pdf.

СДО предусматривает бесплатную версию, рассчитанную на 14 дней и 30 пользователей. По истечению указанного срока следует оформить подписку, минимальная стоимость которой начинается от 5700 руб./мес. за 50 пользователей.

Сравниваемые характеристики представленных выше систем дистанционного обучения для удобства сведены в таблицу 1.

Таблица 1 – Сравнительные характеристики СДО

| СДО                         | Сравниваемые характеристики                              |                                       |                                     |  |  |
|-----------------------------|--|---------------------------------------|-------------------------------------|--|--|
|                             | Цена   | Интерфейс, настройка                  | Отчеты                              | Коммутация   | Варианты использования   |
| <b><i>iSpring Learn</i></b> | Бесплатная пробная версия (14 дней), подписка 56000 руб. | Настройка дизайна, учебных материалов | 19 типов отчетов, нельзя созд. свои | Обмен файлами, площадка для вебинара, нельзя создавать форумы и чаты | Подходит компаниям для быстрого, самостоят. решения задач, связанных с обучением и аттестацией персонала |
| <b><i>Mirapolis LMS</i></b> | Мин. стоимость 600000 руб./год                           | Любые изменения                       | >120 шаблонов отчетов               | Вебинары, чаты, блоги, форумы и т.д.                                 | Крупные компании, автоматиз. бизнес-процессы   |

|                        |                                   |   |  |   |  |
|------------------------|-----------------------------------|---|--|---|--|
| <b>Share Knowledge</b> | 24 000 руб./год                   | Настройка системы, дистанц. и очного обучения       | Базовые отчеты, создание своих отчетов | Добавление форума, блога за отдельную плату                           | Компании, у которых есть портал на платформе SharePoint  |
| <b>Teachbase</b>       | 3200 руб./мес.                    | Добавление логотипа, подкл. домена                  | 4 типа отчетов, созд. свои нельзя      | Вопросы, коммент., площадка для вебинаров, форумы и блоги отсутствуют | Небольшие компании и частные лица                        |
| <b>WebTutor</b>        | 115000 руб./год.                  | Настройка внешнего вида, структуры                  | Отчеты в разных формах                 | Форумы, блоги, чаты, опросы   | Средние и крупные компании, реал. конкретные HR-процессы |
| <b>Docebo</b>          | Бесплатная демо-версия (14 дней)  | Изменение интерфейса и выбор виджетов за доп. плату | 17 типов отчетов                       | Вопросы, вебинары   | Крупные компании   |
| <b>Unicraft</b>        | От 33000 руб. до 191000 руб в год | Нельзя изменять дизайн                              | Отчеты разных типов                    | Отсут. средства общения   | Подходят для быстрого старта небольшим компаниям         |
| <b>e.Queo</b>          | 3000 руб. за 1 польз.ов.          | Настройка дизайна платформы                         | Отчеты по курсам, тестам, опросам      | Переходы из приложения по ссылкам                                     | Крупные компании с большим штатом сотрудников            |
| <b>eTutorium LMS</b>   | 5700 руб. за 50 польз. в месяц    | Индивид. письма для польз.ов., сертиф.              | Отчеты в разрезе курса                 | Форум   | Любой сегмент компаний, внедряющих электронное обучение  |

Все представленные системы дистанционного обучения обладают схожим набором инструментов и поддерживают базовые функции, необходимые для удобной и быстрой аттестации обучающихся. При выборе платформы следует ориентироваться на поставленные задачи, которые должна реализовывать СДО, а также экономические и функциональные требования.

**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Взрыв обучения: Девять правил эффективного виртуального класса / Мэттью Мердок, Трейон Мюллер; Пер. с англ. – М.: Альпина Паблишер, 2012 – 190 с.
2. Электронное обучение: инструменты и технологии / Уильям Хортон, Кэтрин Хортон ; [пер. с англ. Ю. В. Алабина]. – Москва : КуДиц-Образ, 2005. – 638 с.
3. Ценный кадр: Как построить эффективную систему обучения в компании / К. Мальцев – М.: «Альпина Диджитал», 2015, – 100 с.

УДК 004.021

**А.Н. САПРЫКИН, Н.Ю. ЮРКОВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**АЛГОРИТМИЗАЦИЯ В ПРОЦЕССЕ РАЗРАБОТКИ  
АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ОБЩЕНИЯ  
С ПОЛЬЗОВАТЕЛЯМИ**

*Рассмотрен процесс алгоритмизации при разработке программного продукта. Представлены основные алгоритмы работы автоматизированной системы общения с пользователями в системе дистанционного обучения.*

Одним из начальных этапов создания программного обеспечения является алгоритмизация. Под алгоритмизацией понимается процесс построения алгоритма решения задачи, результатом которого является выделение этапов процесса обработки данных, формальная запись содержания этих этапов и определение порядка их выполнения. Использование алгоритма позволяет получить из исходных данных желаемый результат. Составление алгоритма предшествует другим этапам разработки ПО, например, таким как написание кода программы, так как программный код опирается на составленный алгоритм.

Существует несколько способов описания алгоритмов. Одним из наиболее популярных способов является применение блок-схем. В блок-схемах используются геометрические фигуры, каждая из которых изображает какую-либо операцию или действие, а также этап процесса решения задачи. Каждая фигура называется блоком. Порядок выполнения этапов показывается стрелками, соединяющими блоки. Составление блок-схем регулируется международным стандартом ИСО 5807-85, в России иначе именующимся ГОСТ 19.701-90 [1].

Рассмотрим применение алгоритмизации при разработке автоматизированной системы общения с пользователями, используемой для взаимодействия пользователя с элементами системы дистанционного обучения через программу мгновенного обмена сообщениями Telegram.

Запуск автоматизированной системы общения с пользователями (или чат-бота) является первой стадией в работе программы. Блок-схема алгоритма изображена на рисунке 1.

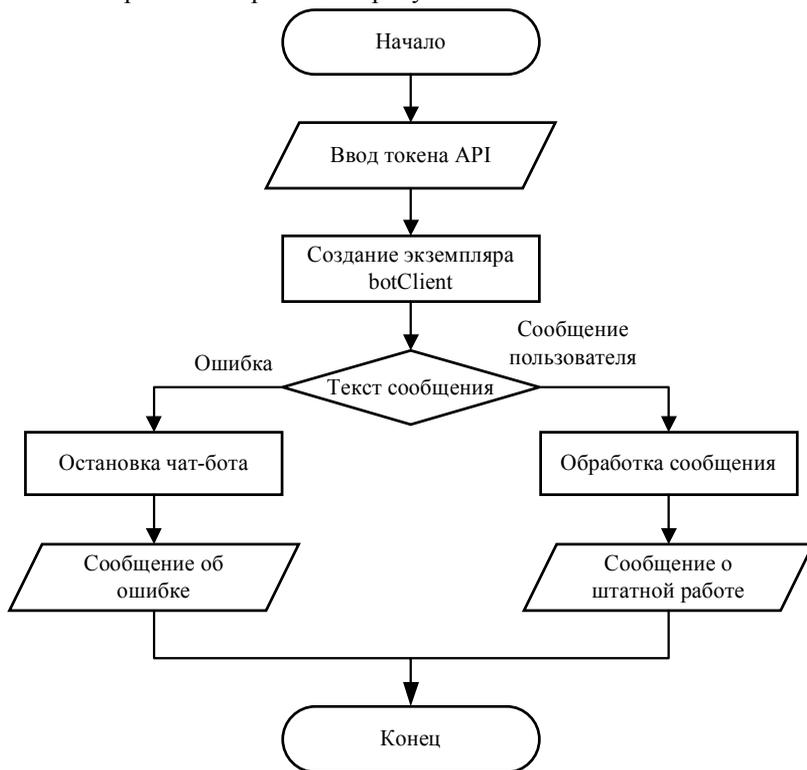


Рисунок 1 – Блок-схема процесса запуска автоматизированной системы общения с пользователями

Алгоритм начинается с ввода токена Telegram Bot API, используемого для идентификации бота в Telegram [2]. Следующим процессом является создание экземпляра класса TelegramBotClient, в котором в качестве параметра выступает токен чат-бота. На этой стадии бот запущен и готов принимать входящие сообщения. Для обработки полученных сообщений используется метод StartReceiving. В качестве

аргументов используется обработчик `DefaultUpdateHandler`, который активируется в случае получения удовлетворяющего требованиям бота пользовательского сообщения, и параметр завершения обработки `CancellationTokenSource`, активирующийся при возникновении программной ошибки. На схеме данное разветвление отображает символ «Решение» согласно ГОСТ 19.701-90. Если поступает сообщение об ошибке, бот будет остановлен с помощью метода `HandleErrorAsync`, отслеживающего возникающие исключения, а на экран будет выведено описание ошибки. Если поступает пользовательское сообщение, метод `HandleUpdateAsync` обрабатывает его и выводит на экран информацию об успешной обработке.

Следующей стадией работы чат-бота является авторизация пользователя в системе дистанционного обучения (СДО) через мессенджер. Авторизация в СДО позволяет пользователю запрашивать и получать данные по своим текущим учебным курсам, оценкам, доступным учебным материалам. Блок-схему алгоритма авторизации представлена на рисунке 2.

Изначально пользователь в мессенджере активирует команду авторизации и вводит логин и пароль для СДО. Запускается алгоритм авторизации. В ходе данного процесса создается экземпляр класса `HttpClient`, отвечающего за отправку HTTP-запросов и получения HTTP-ответов от ресурса, идентифицируемого по универсальному коду ресурса (URI). Затем формируется коллекция `Dictionary`, содержащая ключи и значения элементов `login` и `password`. Коллекция становится наполнением контейнера для кортежей "имя-значение" класса `FormUrlEncodedContent`. К экземпляру класса `HttpClient` добавляется ряд HTTP-заголовков. На основе заголовков и контейнера с логином и паролем формируется POST-запрос и отправляется на сервер СДО. Чат-бот получает HTTP-ответ с кодом состояния. Если код входит в категорию 4xx: `Client Error` (ошибка клиента), пользователь не сможет получить доступ к СДО. Если код входит в категорию 2xx: `Success` (успешно), то пользователь получает доступ и уведомление об успешной авторизации.

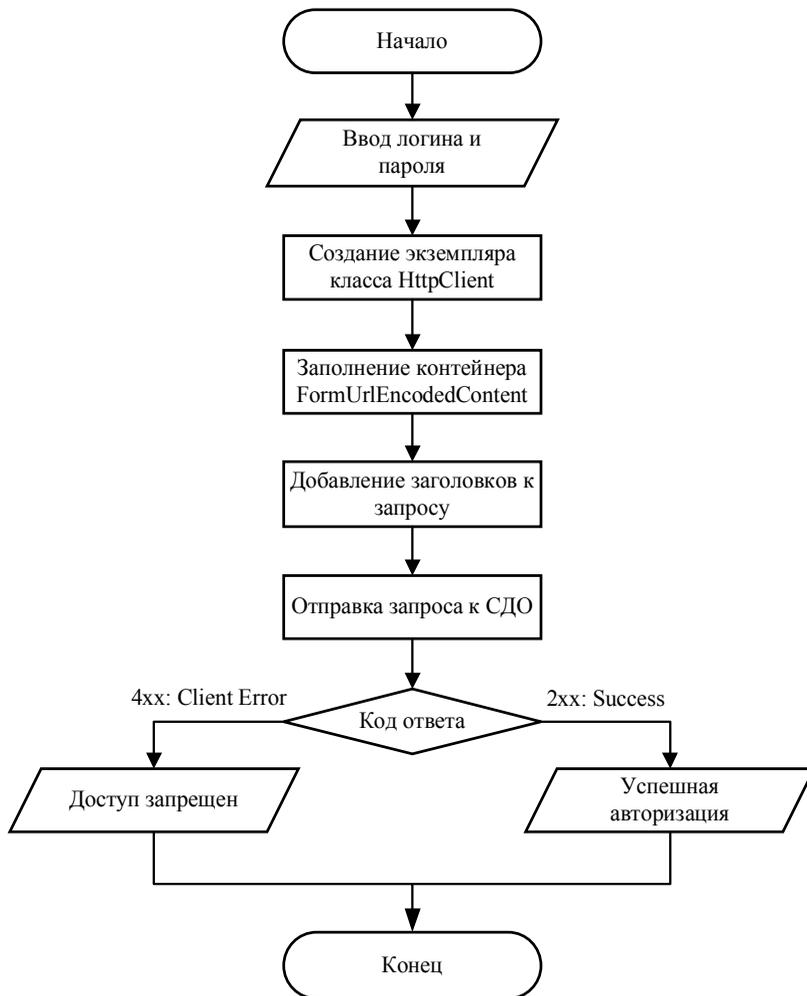


Рисунок 2 – Блок-схема процесса авторизации в СДО с помощью автоматизированной системы общения с пользователями

На основе приведенных выше алгоритмов разработаны соответствующие методы запуска автоматизированной системы общения с пользователями, обработки получаемых сообщений и авторизации в СДО через чат-бота. Разработанные методы используются для расширения дополнительного функционала программы, в который входят получение списка актуальных учебных курсов, оценок внутри кон-

кретного учебного курса, доступ к ответам на наиболее часто задаваемые вопросы и контактам. Алгоритмизация указанных процессов позволяет систематизировать разработку программы за счет наглядного представления реализуемых этапов.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. ГОСТ 19.701-90 (ИСО 5807-85) Схемы алгоритмов, программ, данных и систем [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/9041994> (дата обращения 15.04.2022);
2. A guide to Telegram.Bot library: – Режим доступа: <https://telegrambots.github.io/book/index.html> (дата обращения 15.04.2022);
3. Официальная документация по .Net – Режим доступа: <https://docs.microsoft.com/ru-ru/dotnet/api/system.net.http?view=net-6.0> (дата обращения 15.04.2022).

УДК 004.47

**А.А. САФРОНОВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### МАТЕМАТИЧЕСКАЯ ОБРАБОТКА ЭЛЕКТРОФИЗИОЛОГИЧЕСКОЙ ИНФОРМАЦИИ НА ЭВМ

*Рассматриваются основные виды математической обработки электрофизиологической информации, а так же их область применения.*

Одной из наиболее актуальных задач современной электроники и медицинской информатики является разработка электронных устройств, комплексов и систем мониторинга состояния человеческого организма для изучения параметров биомедицинских сигналов, моделирования алгоритмов, анализа и контроля основных параметров жизненно важных функций, а также предотвращения критических ситуаций.

Математическая обработка электрофизиологической информации на ЭВМ широко применяется при диагностике и лечении пациентов в условиях стационара. Уже много лет ведущие фирмы по производству электронной аппаратуры для электрофизиологических исследований («Nihon Kohder» и «San — Ei» — Япония, «Di—sa» Дания, «Nicolett» — США, «Bio — medi—ca» — Италия и др.) выпускают

компьютерные системы для получения вызванных потенциалов, спектрального анализа и т. д. Стандартизация методов обработки электрофизиологической информации и производство специализированного компьютерного оборудования сопровождается поиском новых методов, а также совершенствованием уже применяемых. Все чаще используется нейромониторинг; интраоперационный и послеоперационный мониторинг состояния пациентов с помощью ЭВМ.

На данном этапе развития использование компьютера востребовано в медицинских и физиологических исследованиях для обработки экспериментальных данных. Класс используемых алгоритмов очень широк — от стандартных статистических вычислений до сложных методов распознавания закономерностей и планирования экспериментов.

Механизм обработки записей электрофизиологических процессов носит многоступенчатый характер и представляет собой последовательное распределение характеристик (информационных признаков) записей, которые точно характеризуют многие состояния организма для диагностических ошибок, соответствующих состояниям, представляющим интерес. Следует отметить, что на каждом этапе обработки данных объем вывода на единицу времени ввода значительно уменьшается.

Наиболее часто при обработке электрофизиологических сигналов используются операции цифровой обработки сигналов (ЦОС): фильтрация, спектральный, корреляционный, статистический, регрессионный анализы, аппроксимация, полиномиальные преобразования.

**Фильтрация** применяется на этапе предварительной обработки и чаще всего имеет целью формирование рабочей полосы частот исходного сигнала и подавления помех (мешающих сигналов). Наиболее общее описание цифрового фильтра может быть представлено через дискретный аналог уравнения свертки

$$y_i = \sum_{k=0}^M h_k \cdot x_{i-k}, \quad (1)$$

где  $y_i$  —  $i$ -й отсчет исходного сигнала;  $h_k$  — импульсная переходная функция цифрового фильтра длиной  $M$  отсчетов.

Следует отметить, что в случаях анализа электрокардиограмм, электроэнцефалограмм как к сигналам нестационарным следует применять время — частотные фильтры. На практике исследователи часто используют для обработки перечисленных сигналов полиномиальные фильтры, рассматривая последние как стохастические.

Несомненно, что фильтрация присутствует практически во всех современных приборах и методах обработки и анализа сигналов. Она незаменима при первичной селекции данных и позволяет во многих случаях эффективно отсеивать грубые отклонения и промахи в измерениях. Та же фильтрация может помочь при устранении влияния стационарных помех в ограниченном диапазоне частот.

**Спектральный анализ** — область обработки и анализа сигналов, которой посвящено подавляющее большинство информации, связанной с обработкой электрофизиологических сигналов. Этот факт объясняется независимостью влияния на результаты исследования различного рода помех в математике алгоритма быстрого преобразования Фурье (БПФ) и преобразования Винограда. На основании этих алгоритмов построенные спецпроцессоры быстрой свертки и спектрального разложения в различных базисах (Фурье, Уолша, Адамара и их производных) построены все современные цифровые электрофизиологические приборы.

Спектральное преобразование сигнала используется для его частотного разложения в виде ряда периодических функций. Спектральный анализ предполагает представление выборки сигнала

$\{x_j\}$  объемом  $N = T/\Delta t$  по ортогональному базису  $F(f)$ , где  $T$  — длительность выборки,  $\Delta t$  — период квантования:

$$X(f, T) = \Delta t \cdot \sum_{n=0}^{N-1} x_n \cdot F(f), \quad n = \overline{0, N-1} \quad (2)$$

Обычно в качестве базиса выбирается базис Фурье для анализа гармонических (например, при энцефалографии) сигналов и базис Уолша для анализа бинарных сигналов (например, сигналы импульсной активности нейронов).

**Корреляционный анализ** часто используется в экспериментах для построения связи нескольких, но чаще двух сообщений:  $X$ ,  $Y$ . автокорреляционный анализ часто используется для изучения внутренних свойств и характеристик сигнала. При корреляционном анализе единственное значение коэффициента корреляции, как правило, неинформативно; самое интересное — это определение изменчивости отношений. При этом изучается динамика временного коэффициента взаимной корреляции.

Методы статистической обработки данных часто используются в биомедицинских исследованиях. Все эти методы основаны на расчете средней цены выбранного параметра.

Как известно, задачей любого исследования является выявление причинно-следственных связей. Зная точные причины событий, мы можем правильно интерпретировать наблюдаемые закономерности.

Корреляционный анализ не отражает причинно-следственный характер отношений. Этот метод оценивает как прочность соединения, так и его близость. Регрессионный анализ же используется для определения причинно-следственной связи между явлениями (переменными, признаками).

**Регрессионный анализ** имеет одну цель: решить задачу аппроксимации данных.

Преимуществом регрессионного анализа является возможность получения экспериментальной зависимости от влияния случайных факторов. На основе регрессионного анализа часто разрабатываются полиномиальные математические модели и рассчитываются их коэффициенты. Недостатком являются значительные вычислительные затраты при высокой степени многочлена. Регрессивные многочлены хороши для аппроксимации медленно меняющейся последовательности данных (в статистическом смысле). Они совершенно непригодны для тонкого анализа информации, передаваемой по сигналам, необходимым для выявления дифференциально-диагностических признаков заболевания.

В нелинейных системах незначительное изменение внешнего воздействия может привести к эффектам, не соответствующим первоначальному. Картина может представлять собой изменение функционального состояния (ФС) человека и животных.

С математической точки зрения ФС можно охарактеризовать как информационное пересечение определенных жизненных показателей в течение заданного цикла биологического процесса у человека или животных. Для решения задачи получения информации от электрофизиологических сигналов, которые могут быть использованы для получения устойчивых признаков ФС биологических объектов, были использованы некоторые специальные алгоритмы обработки. Среди известных алгоритмов наиболее близкими к математической обработке электрофизиологических сигналов являются алгоритмы по их математическому значению, наиболее подходящие — алгоритм восстановления динамики сложных систем по временной цепи данных и алгоритм отображения информации биологических сигналов по ее структуре. Эти методы анализа называются нелинейными методами. В отличие от линейных систем, это системы, структуры, свойства, состояния и процессы которых характеризуются нелинейными уравнениями.

Программное обеспечение комплекса для математической обработки электрофизиологической информации на ЭВС может включать в себя следующие программы: анализа биоэлектрической активности мозга, анализа вегетативных процессов (сердечного ритма и оценки сосудистого тонуса), обработки результатов спектрального, корреляционного и статистического анализов данных.

К первой группе программ относятся программы ввода биоэлектрической активности — спонтанной и вызванной на стимулы (коротко- и длиннолатентных) различной модальности (звуковой, зрительной, соматосенсорной). Последние можно регистрировать одновременно во многих отведениях (до 12). Это необходимо при топографическом анализе вызванных потенциалов для оценки функционального состояния различных отделов полушарий мозга.

Программы спектрального анализа с использованием быстрого преобразования Фурье и корреляционного анализа можно использовать для любых временных рядов.

Программы анализа вегетативных процессов — сердечного ритма, сосудистого тонуса — служат и для оценки функционального состояния мозга, так как эти процессы имеют центральную регуляцию. В программе анализа сердечного ритма помимо статистических характеристик (среднего значения, дисперсии, асимметрии, эксцесса, максимального и минимального значений) строится гистограмма зависимости длительности последующего сердечного цикла от предыдущего.

Математическая обработка электрофизиологической информации на ЭВМ, непрерывное расширение и создание программ и их функциональных характеристик, интеграция в единую систему гетерогенных биомедицинских компьютерных систем, обеспечивает полную совместимость в архитектуре программного обеспечения и широкие возможности для повторного использования кода разработанных элементов системы.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Водлазский Л. А. Основы техники клинической электрографии Москва: Медицина. —1966.
2. Шакин В. В. Вычислительная электрокардиография. — М.: Наука. —1981.
3. Дроздов Д. В. Персональный компьютер в качестве электрокардиографа – за и против. // М: Компьютерные технологии в медицине. —1996. —№1.

4. Селищев С.В. Автоматизированное проектирование биомедицинских электронных систем // Биомедицинские технологии и радиоэлектроника. – М.: №12, 2001.
5. Коэн Л. Время-частотные распределения//ТИИЭР. – Т. 77. – № 10. – 1989. – С. 86–96.
6. Трифанов В.Н. Методические основы синтеза динамических сетей: алгебраическое равновесие и статистика//Препринт 148., – Л.: ЛИИАН, 1985.

УДК 004.021

### К.В. СЕЛИВЕРСТОВ

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## ДЕТЕКТОР ГРАНИЦ ГРАДИЕНТНОГО ТИПА С ВЕКТОРНОЙ МАСКОЙ

*В статье приведена краткая классификация известных алгоритмов детектирования границ перепада яркостей на изображениях, более подробно обсуждаются градиентные методы выделения границ. Приведено описание, программная реализация и результаты обработки реальных изображений с помощью градиентного метода, основанного на оценивании частных производных в составе вектора градиента с помощью векторной маски.*

Переход от фоновых значений яркости на реальном изображении к большому (или меньшему) значениям в областях, представляющих интерес с позиций анализа изображения, происходит, как правило, не в виде скачка, а в виде постепенного изменения яркости. Такая особенность яркостей изображения создает неопределенность при выделении границ перепада яркостей. Известно большое число методов детектирования границ. В частности, это градиентные методы и методы сегментации изображения [1]. Практика применения методов детектирования подтвердила высокую эффективность градиентных методов. В настоящее время именно эти методы используются в реальных системах технического зрения.

Градиентные методы основаны на вычислении модулей и направлений градиентов скалярного поля яркостей изображения  $I(x_i, y_j) = I_{ij}$  с последующим пороговым сравнением модулей градиентов в направлении поля градиентов. Для этого в каждой точке (пикселе)  $(i, j)$  изображения вычисляются:

- оценки частных производных  $\frac{\partial I_{ij}}{\partial x}$  и  $\frac{\partial I_{ij}}{\partial y}$ ;

- модуль градиента  $|\nabla I_{ij}| = \sqrt{\left(\frac{\partial I_{ij}}{\partial x}\right)^2 + \left(\frac{\partial I_{ij}}{\partial y}\right)^2}$ ;

- направление вектора градиента – угол  $\alpha$ , образованный вектором градиента с положительным направлением оси  $Ox$ .

Исторически одними из первых градиентных методов детектирования границ были методы, основанные на применении масок Превитта

$$\begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} u \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

и Собела [1]. Маски Собела отличаются от масок Превитта усиление роли центральных пикселей при вычислении разностей и имеют следующий вид

$$\begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} u \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

Данные маски усиливают значения для линий в направлениях координатных осей.

Огромный шаг вперед в развитии градиентных методов выделения границ (линий) перепада яркости и формирования из этих линий контуров объектов на изображении был сделан Джоном Кенни [2]. Он сформулировал три критерия, определяющие хорошие детекторы границ:

- хорошее отношение сигнала к шуму;
- хорошая локализация;
- единственность отклика на границу.

С содержательной точки зрения первое требование означает, что детектор должен выделять истинные границы и не выделять ложные. Второй критерий предполагает, что выделенные точки границы должны располагаться максимально близко к истинному положению границы области.

Метод Кенни является многошаговым, что обусловлено множественностью ограничений, которым он должен удовлетворять. Программные реализации алгоритма Кенни в общем случае состоят из пяти шагов:

- размытие изображения для удаления шума;
- вычисление градиентов, их модулей и направлений градиентов (поля направлений);
- подавление немаксимумов (поиск локальных максимумов модуля градиента в направлении поля градиентов);
- пороговая фильтрация выделенных граничных точек по двум порогам (идентификация «сильных» и «слабых» линий);
- объединение границ в контуры (в компоненты связности).

В детекторе границ Кенни используется два порога: нижний и верхний. Пиксель, значение которого выше верхней границы, принимает максимальное значение, т. е. контур считается достоверным. Если значение пикселя не достигает нижнего порога — пиксель подавляется. Если его значение попадает в диапазон между порогами, то он принимает среднее значение, а решение о том, является ли он точкой границы, будет принято во время трассировки области неоднозначности. Задача трассировки сводится к распределению пикселей, получивших среднее значение. Если такой пиксель соприкасается с достоверным контуром, то его значение приравнивается к максимальному значению, и он становится частью границы, в противном случае он подавляется.

Несмотря на очевидные достоинства метода Кенни, он не свободен и от недостатков. Во-первых, как замечено в известной монографии Гонсалеса и Вудса [1], он стремится замыкать все контурные линии. Во-вторых, он имеет относительно большую вычислительную сложность, что создает проблемы при его применении в системах технического зрения реального времени. В связи с этим в работе [3] предложен новый метод детектирования границ.

В основу подхода к вычислению оценок частных производных в новом методе (детектор границ на основе векторной маски) положено достаточно реалистичное предположение, что срез изображения по строке или по столбцу в малой окрестности каждого пикселя адекватно аппроксимируется линейной функцией. Например, яркость пикселей в срезе по строке с номером  $i_0$  в окрестности точки с координатами  $(i_0, j_0)$  будет иметь вид:

$I_{i_0 j_0} = a + b \cdot x$ ,  $x \in \{j_0 - k, j_0 - k + 1, \dots, j_0 - 1, j_0, j_0 + 1, \dots, j_0 + k\}$ . Но тогда оценка частной производной по координате  $x$  будет иметь следующую

формулу 
$$\frac{\partial I_{i_0 j_0}}{\partial x} = \frac{\partial (a + bx)}{\partial x} = b$$
. Оценку коэффициента  $b$  в линей-

ной модели с конечной памятью  $(2k + 1)$  можно найти методом наименьших квадратов. Она будет иметь следующий вид [3]

$$\hat{b} = \frac{3}{(k(k+1)(2k+1))} (-k, -k+1, \dots, -1, 0, 1, \dots, k-1, k)$$

Коэффициентом  $\frac{3}{(k(k+1)(2k+1))}$  можно пренебречь, поскольку нас интересуют относительные значения частных производных. Тогда рабочая маска для оценивания частных производных (в каждом пикселе вдоль строки -  $\frac{\partial I_{ij}}{\partial x}$ , и вдоль столбца -  $\frac{\partial I_{ij}}{\partial y}$ ) примет следующий вид:

$$\hat{b} = (-k, -k+1, \dots, -1, 0, 1, \dots, k-1, k)$$

Построенная маска позволяет получать сглаженные оценки частных производных. Как следствие отпадает необходимость предварительного сглаживания изображения, без которого не может работать метод Кенни.

Длина  $2k+1$  окна скольжения влияет на точность оценки частной производной в условиях шума. Чем выше интенсивность случайной составляющей в составе изображения, тем большим должно быть значение параметра  $k$ . Однако большая длина маски может приводить к «задвоению» выделяемых линий.

Для того чтобы алгоритм детектирования обнаруживал истинные границы перепада яркости, отвечающие контурам объектов на подстилающей поверхности, и не обнаруживал ложные линии, по аналогии с методом Кенни вводятся два порога  $\Delta 1$  и  $\Delta 2$ . Пороги  $\Delta 1$  и  $\Delta 2$  должны удовлетворять условию  $|\nabla I_t|_{\min} < \Delta 1 < \Delta 2 < |\nabla I_t|_{\max}$ . Экстремумы в каждом пикселе ищутся по тому направлению, которое найдено для этого пикселя. Если в этом направлении выполняется неравенство  $|\nabla I_{t-1}| < |\nabla I_t| > |\nabla I_{t+1}|$  по соответствующему направлению, то в точке  $t$ , то есть в пикселе  $(i, j)$ , фиксируется локальный максимум модуля градиента.

Если при этом найденный локальный экстремум удовлетворяет неравенству  $|\nabla I_{ij}| > \Delta 2$ , то в ячейку  $(i, j)$  записывается 1, если же  $\Delta 1 < |\nabla I_{ij}| < \Delta 2$  то в ячейку  $(i, j)$  записывается 2. Линия из единиц,

называется «сильной» линией, а из «2» – «слабой» линией. Сильные линии являются основой контуров изображения. Пиксель слабой линии может присоединяться к сильному пикселю, если он связан с ним по схеме восьмисвязной области. В противном случае слабые пиксели не включаются в итоговое контурное изображение

На рисунке 1 приведен пример работы описанного алгоритма – исходное изображение (рисунок 1а) и отвечающее ему контурное изображение (рисунок 1б).



Рисунок 1а – Исходное изображение

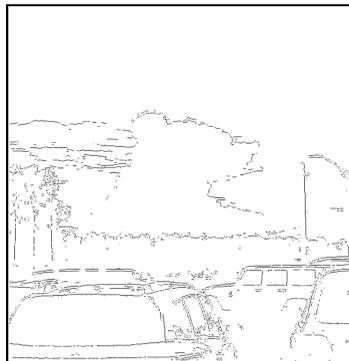


Рисунок 1б – Отвечающее исходному контурное изображение

### Заключение

Приведен краткий анализ методов детектирования границ перепада яркостей. Подробно описан метод детектирования градиентного типа, в котором оценки частных производных получаются с помощью векторной маски. Такая маска обеспечивает получение сглаженных оценок частных производных. Поскольку оценки частных производных получаются «сглаженными», то отпадает необходимость в предварительном сглаживании изображения. Соответственно сокращается вычислительная сложность алгоритма. Тогда как в методе Кенни сглаживание является обязательной процедурой даже для изображений с низким уровнем шума.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. - М.: Техносфера, 2005. – 1072 с.
2. Canny, John. A Computational Approach to Edge Detection / John Canny // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1986. – vol. PAMI-8 NO 6. – Pp. 679–698.

3. Новиков, А.И. Детектор границ градиентного типа для изображений подстилающей поверхности / А.И. Новиков, А.В. Пронькин // Вестник РГРТУ. – 2019. – № 68. – С. 68-76. DOI: 10.21667/1995-4565-2019-68-2-68-76.

УДК 004.65

**С.С. САВИНКИН, А.А. АНИКИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

## **ОСНОВЫ BIM-ПРОЕКТИРОВАНИЯ**

*Рассматриваются востребованное создание моделей на базе технологий BIM – проектирования, предпосылки к применению в различных инжиниринговых компаниях данной технологии; особенности и преимущества применения инновационных технологий при проектировании, управлении и эксплуатации объектами.*

На сегодняшний день осуществление контроля над жизненным циклом (ЖЦ) объекта, основанного на разнообразных технологиях BIM–проектирования, являются одними из самых важных условий эффективного проекта, задавая при этом вектор филиации идей компании. Едиными и самыми значимыми условиями к использованию в различных инжиниринговых компаниях данной технологии при проектировании служат следующие факторы:

- этапы становления компании;
- спрос клиентов;
- потребности высших органов проверки (фактор, который стал востребованным).

Также, нынче, многие строительные компании и организации, понимают, что им необходимо иметь высококачественные функции, которые взаимодействуют с большим количеством данных, поэтому они разрабатывают собственные проекты на базе систем автоматизированного проектирования (САПР), которые в дальнейшем способны справиться с такими задачами, как создание и управление информацией о здании или сооружении с чётко описанными бизнес-процессами на протяжении его полного жизненного цикла (ЖЦ). Данный метод полностью подходит под требования комплексного инжиниринга (включает полный комплекс услуг по обоснованию, разработке и реализации проекта, включая поставку объектов интеллектуальной собственности, оборудования и сдачу объекта под ключ) и обеспе-

чивает достаточно высокую конкуренцию другим компаниям, которые выполняют EPC/EPCM контракты (EPC/EPCM – Engineering, Procurement, Construction, Management – договорные обязательства, включающие весь спектр работ от концепции до введения объекта в эксплуатацию). Описание EPCM – контракта:

- Engineering (E) – это инжиниринг, проектирование, конструирование;
- Procurement (P) – это закупка, поставка;
- Construction (C) – это сооружение, строительство;
- Management (M) – управление, менеджмент.

#### **Основные понятия.**

BIM (Building Information Model/Modeling) – это информационная модель моделирования зданий. Она представляет из себя разнообразные предметы комплекса взаимосвязанных обслуживающих структур или объектов, например инженерные сети, дороги, мосты, порты, тоннели и т.д. То есть информационное моделирование здания (ИМЗ) – это информационный подход к контролю ЖЦ строительного объекта, который предполагает полный контроль всей информации о строительном объекте, в виде сбора и обработки этой же информации, учитывая все зависимости с другими конструкторскими объектами, которые имеют прямое или косвенное отношение к целевому объекту. Именно такой подход позволяет рассматривать несколько объектов в виде одного.

В основе BIM лежит ООП.

Объектно-ориентированное проектирование (ООП) – это методология построения моделей информационной системы (ИС), она основана на представлении программы в виде совокупности объектов, использующая механизм пересылки сообщений и классы, объединенные в иерархию наследования.

Технологии такого типа проектирования имеют следующие преимущества, которые делятся на две группы:

- краткосрочные (рисунок 1);
- долгосрочные(рисунок 2).



Рисунок 1 – Краткосрочные преимущества

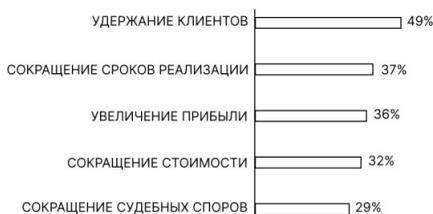


Рисунок 2 – Долгосрочные преимущества

### ***BIM и жизненный цикл проекта.***

Применение на практике разнообразных технологий данного моделирования для различных комплексов взаимосвязанных обслуживающих структур, которые составляют и обеспечивают основной функционал системы, берут в охват целиком весь ЖЦ любого проекта (рисунок 3).

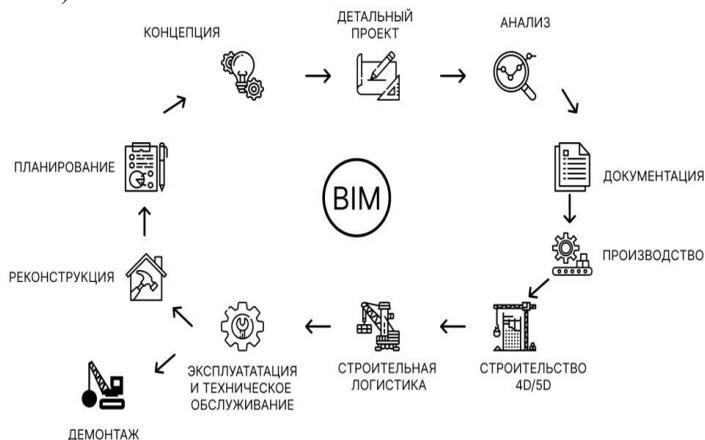


Рисунок 3 – Жизненный цикл проекта

### ***Процесс создания BIM-проекта.***

BIM – проектирование на предприятии должно быть организовано на базе единого платформенного информационно-технологического решения, способного распоряжаться, сохранять и управлять всей информацией в единой ИС, при этом объединяя все стадии создания сложного инженерного объекта в единую информационную BIM-модель. Это комплексное решение, позволяющее осуществлять управление данными всех частей проекта.

### ***Процесс создания BIM-модели.***

Данный процесс можно представить следующим алгоритмом:

– заказчик осуществляет запрос на создание;

– формирование технического задания, EIR (Employer Information Requirements – это информационные требования заказчика) (рисунок 4);

– формирование исполнителем ВЕР (BIM Execution Plan – это ответный документ проектировщика, расширяющий и дополняющий требования заказчика) (рисунок 4);

– предпроект, компоновка оборудования;  
 – разнообразные конструкторские работы;  
 – проектные работы;  
 – утверждение и согласование;  
 – создание различных видов документов, которые связаны с рабочим процессом;

– переходный процесс (т.е. происходит переход спроектированной модели в строительство);

– проверка актуальности модели в течение ЖЦ;

– осуществление процесса архивирования.

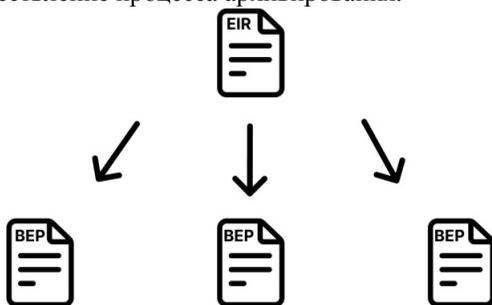


Рисунок 4 – Базовые документы, участвующие в процессе создания BIM-модели

Вся информация, хранящаяся, обрабатываемая и получаемая в BIM – модели представлена на следующем рисунке (рисунок 5):

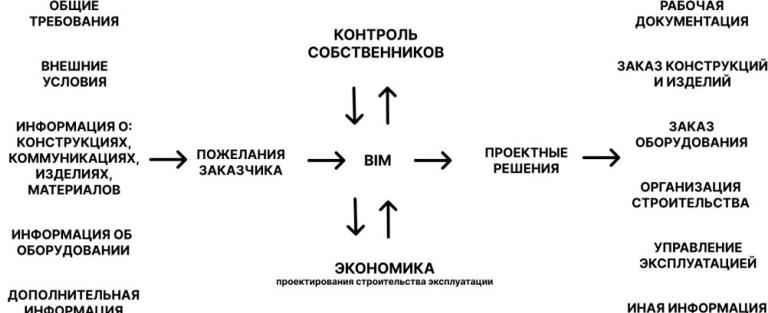


Рисунок 5 – Информация в BIM-модели

Развитие и разработка модели ведётся в среде общих данных (CDE) – это значит, что все заинтересованные лица имеют постоянный доступ к модели, которая наполнена полезной и актуальной информацией (рисунок 6):

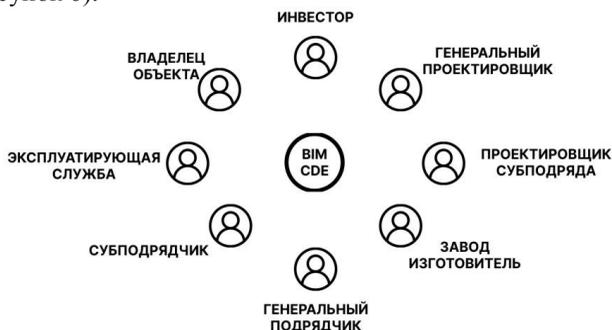


Рисунок 6 – Представление среды общих данных

Таким образом, BIM охватывает все стадии жизненного цикла здания или сооружения, непрерывно наполняя его ценной информацией.

#### ***Детализация проектных решений.***

На всех стадиях проекта должна быть реализована детализация в виде проработки ИМ до определённых уровней – LOD и LOI 100-500.

LOD (Level of Development) – это элемент (объект) BIM-модели, который включает в себя различные данные и знания об объекте с привязкой к определённому зданию или системе.

LOI (Level of Information) – это элемент (объект или носитель) информации в системе управления инженерными данными, которая отвечает за хранение и формирование информации по проекту.

Система уровней детализации или другими словами СУД включает в себя пять различных стадий.

Эти стадии являются характеристиками процесса разработки:

- LOD/LOI 100 – это стадия технических решений, которая предназначена для подготовки общей концепции объекта, а также на данной стадии используются элементы низкого уровня проработки;

- LOD/LOI 200 – это стадия проработки проектной документации;

- LOD/LOI 300 – это стадия проработки рабочей документации;

- LOD/LOI 400 – это стадия проработки рабочей документации;

- LOD/LOI 500 – это стадия проработки «As-build» («как построено»).

#### ***Единая информационная система bim-проекта.***

Единая Информационная Система BIM–проекта – это общее платформенное информационно–технологическое решение, в которую входят:

- ИС;
- взаимодействия ИС, которые обеспечивают качественное управление и сопровождение проекта на всех его стадиях ЖЦ, что в полной мере обеспечивает все необходимые требования комплексного инжиниринга.

***ВМ решения проблем.***

Проблемы, с которыми помогают справляться BIM – технологии представлены в следующей таблице (таблица 1):

Таблица 1 – Проблемы, решаемые при помощи BIM – технологий

| <b>Проблемы</b>                   | <b>Решения</b>  |
|-----------------------------------|---|
| Соответствия требованиям рынка    | Глубокий и всесторонний анализ в рамках ТЭО на основании вариантов BIM – модели LOD 200 стадии ПП; формирование материалов для маркетинга   |
| Оценки ресурсов и рисков          | Формирование первичных требований к оборудованию; оценка ТУ на подключение к коммуникациям; варианты оптимального расположения объекта на местности   |
| Согласованности                   | Все участники проекта работают в СОДи имеют оперативный доступ к информации с обратной связью; сроков – моделирование календарных и сетевых графиков производства работ на основе информационной модели |
| Сроков                            | Моделирование календарных и сетевых графиков производства работ на основе информационной модели   |
| Бюджетирования                    | Моделирование финансовых потоков на основе информационной модели 4D   |
| Сокращения исходного объема работ | Путём чётко сформулированного бюджета, сроков, соответствия состава работ проекту   |
| Отклонения от проекта             | Всесторонний анализ и утверждение технических решений при проектировании, проверка на коллизии  |

|                        |   |
|------------------------|---|
| Качества строительства | Применяемы материалы и оборудование заранее достоверноизвестны; объёмы работ точно определены; авторский контроль наоснове информационной модели  |
| Охраны труда           | Путём оптимального размещения и контроля элементов (ограждения, леса,пожарные гидранты и т.д.), обеспечивающих безопасность   |
| Эксплуатации           | Электронный паспорт объекта,включающий всю актуальную информацию об оборудовании, материалах, сроках службы, системахмониторинга и обслуживания   |
| Прозрачности           | Данные об объекте и ходе реализации доступны в облаке;роли и обязанности регламентируются требованиями EIR, ВЕР; все измененияоперативно вносятся в модели с возможностью обратной связи. |

### **Внедрение BIM-проектирования и его преимущества.**

Ныне существует множество различных способов использования технологий BIM – проектирования:

– 3D-визуализация:

При помощи данного способа появилась возможность создавать на экране трёхмерную подробную и правдоподобную модель здания, что очень сильно позволило облегчить многие задачи. С помощью данной функции можно преподнести проект, визуально его оценить, распечатать натуральный макет на 3D – принтере, который может воспроизводить трёхмерную печать, а также тестировать постройку и вносить различные коррективы.

– Объединение хранимых данных в группы:

Сведения, которые необходимы проектировщику, располагаются всегда в рамках одной программы, а именно в единой связке чертежей. Если, допустим, изменить один параметр, то автоматически изменения окажут воздействие на остальные элементы. Данный способ позволяет значительно снизить вероятность ошибки.

– Комплексное управление данными:

В обычный генеральный план входят:

– отдельные решения по проекту, которые всегда представлены в виде чертежей;

– документы, которые сопровождают чертежи.

В софте, который поддерживает BIM можно производить объе-

динение разнообразных сведений в рамках одной программы (т.е. в одном едином файле), что значительно позволяет облегчить работу.

К плюсам внедрения таких технологий можно отнести быстроту создания проекта, оптимизация работы с чертёжными и не только данными, экономию ресурсов разного вида, благодаря чему большую часть работы выполняет программа.

#### **Заключение.**

BIM – это ещё совсем новая технология, которая только недавно начала стремительно развиваться. Но при этом очень данная методология очень понравилась пользователям благодаря тому, что очень сильно упрощает процесс создания необходимой документации.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Талапов Владимир. Технология BIM. Суть и особенности внедрения информационного моделирования зданий. ДМК-Пресс, 2015г. 410с.
2. Бахарева О.В., Романова А.И., Талипова Л.Ф., Федорова С.Ф., Шиндина Т.А. О построении информационного моделирования развития рынка объектов капитального строительства // Журнал Интернет-банкинга и коммерции. – 2016. – Вып. 21. – № С3.
3. Бахарева О.В., Кордончик Д.М. Инвестиции в сохранение и развитие регионального культурного наследия. Библиотека BIM-элементов, представляющих национальные архитектурно-градостроительные памятники // Архитектура и инженерия. – 2019. – Вып. 4. – № 3. – С. 39–48.

УДК 62-932.4

#### **О.И. СИНЯКОВА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПРИМЕНЕНИЕ АВТОМАТИЧЕСКОЙ ОПТИЧЕСКОЙ ИНСПЕКЦИИ ALD625 ПРИ КОНТРОЛЕ КАЧЕСТВА ПАЯНЫХ СОЕДИНЕНИЙ**

*В статье рассматриваются особенности оптического контроля качества изделий на примере АОИ ALD625.*

Развитие технологии изготовления микросборок представляется одной из важнейших проблем современной электронной промышленности.

ности. В настоящее время наблюдается быстрый рост числа производителей элементной базы и оборудования в этой сфере деятельности. При этом одним из приоритетных направлений стало формирование технологий конструирования и сборки электронных узлов на плате с применением технологии поверхностного монтажа.

В производстве электронных средств, как и во всяком производстве, особое внимание уделяют контролю качества сборки. От качества сборки зависит надёжность изделия в целом.

Бесспорно, что паяные соединения в ходе создания электроаппаратуры должны проверяться при помощи неразрушающего контроля (НК). Этот вид контроля позволяет наблюдать, регистрировать и анализировать результаты взаимодействия с объектом контроля влияющих факторов (физических полей, т.е. излучений или веществ), при этом характер данного влияния напрямую зависит от химического состава, строения, состояния структуры контролируемого объекта.

С учетом типа физического взаимодействия с контролируемым объектом неразрушающий контроль разделяют на следующие базовые виды:

- оптический;
- электрический;
- тепловой;
- акустический;
- магнитный;
- вихретоковый;
- радиоволновой;
- радиационный;
- контроль проникающими веществами.

В серийном производстве РЭА обычно применяют два способа НК паяных соединений – оптический и электрический.

Наиболее известным способом является оптический контроль, осуществляемый специальным оборудованием, основанным на применении автоматической оптической инспекции (АОИ) с высокоточными камерами и специальным ПО для распознавания дефектов электронных компонентов. Однако острая нехватка оснащения для автоматического контроля электронных сборок в настоящее время приводит к сдерживанию развития электронной промышленности.

Рассматриваемый способ контроля качества позволяет выявить дефекты только в конце технологической линии, но далеко не всегда очевидно на каком именно шаге появились дефекты, чем они вызваны и какие действия нужно предпринять для того, чтобы не допустить их

появления в будущем. Существенным является тот факт, что чем позже найден дефект – тем дороже цена его ремонта

Системы АОИ используются для обнаружения дефектов, а также для сбора данных и статистического анализа производства. Автоматическая оптическая инспекция обладает значительными достоинствами перед визуальными методами контроля в быстроте, производительности и гибкости. Контроль качества монтажа с помощью систем АОИ можно использовать после установщика, но до печи, тогда с ее помощью можно контролировать:

- Отсутствие компонента;
- Смещение компонента;
- Неправильная полярность компонента;
- Неправильный номинал компонента (надпись на корпусе компонента).

В случае размещения после печи оплавления системы могут установить такие дефекты, как:

- Отсутствие компонента;
- Смещение компонента;
- Неправильная полярность компонента;
- Неправильный номинал компонента;
- Недостаток/избыток припоя;
- Неправильное создания галтели;
- Эффект «надгробного камня»;
- Приподнятый вывод микросхемы.

Нужно отметить, что исправление дефекта после печи оплавления более «дорогое» по сравнению с ремонтом сборки до оплавления припоя.

Последнее время основной акцент при разработке оборудования для АОИ делается на несложность управления, интуитивное меню, возможность быстрого создания инспектирующих программ и графический интерфейс.

Автоматическая оптическая инспекция ALD625 израильской компании Aleader – это универсальное средство для контроля плат на всех стадиях монтажа: после нанесения паяльной пасты, после установки компонентов и после пайки. Универсальность АОИ Aleader ALD625 заключается в её максимальных возможностях контроля дефектов. Система позволяет выявить такие дефекты нанесения паяльной пасты как нанесение с неправильным углом, отсутствие пасты, растекание пасты, недостаточное нанесение, неравномерное нанесение; дефекты установки компонентов такие как отсутствие компонента, установка с перекосом, эффект "надгробного камня", эффект "рек-

ламного щита", установка наоборот (относительно ключа), установка неверного типа компонента, целостности корпуса.



Рисунок 1 – АОИ ALD625

Отдельно стоит подметить способность системы АОИ Aleader ALD625 анализировать компоненты выводного монтажа по всем параметрам: планарность выводов, форма галтели, дефекты пайки, растекание флюса, перемычки. Так же значительным плюсом системы является возможность контроля маркировки изделия.

При всех вышеперечисленных возможностях, система АОИ Aleader ALD625 всячески упрощает работу оператору, так как для начала работы не потребуются значительных временных затрат на создание программы инспектирования.

Объединить предельное количество анализируемых параметров в одной системе позволила созданная компанией Aleader ALD625 разработка i3D. Данная технология основана на алгоритме сравнения изображений в цветовом коде, полученных на основе топографической модели формы компонента. При этом система не чувствительна к затенению компонентов, что снижает процент ошибочных срабатываний при контроле плат и повышает производительность.

В системах серии ALD применяется новый метод формирования изображения компонента при автоматической оптической инспекции. Способ "объёмной топографии" пришёл на смену классическим методам контроля плат, который основан на сравнении плоских изображений сегмента.

Новый, высокотехнологичный метод проверки, обладает следующими преимуществами:

1. Скорость и простота программирования аппаратуры (на создание программы тратится примерно 30 минут);

2. Отсутствие нужды принятия решения о дефекте оператором;
3. Более высокая эффективность проверки плат;
4. Более точное установление дефектов;
5. Возможность проведения инспекции выводных компонентов после объёмного монтажа;
6. Отсутствие "теневых" зон.

Метод топографии позволяет получить понятное программному обеспечению машины изображение для алгоритмического анализа, чего нет у классической технологии сравнения "плоских" изображений.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Современные системы АОИ. [Электронный ресурс]. – Режим доступа: <http://www.ostec-smt.ru/knowledgebase/catalogues-and-brochures/>.
2. Симонов А. А. Оптические методы контроля электронных сборок // Актуальные исследования. 2020. №10 (13). Ч.1. С. 59-64.
3. Урличич Ю.М., Данилин Н.С. Неразрушающий контроль паяных соединений в радиоэлектронной аппаратуре. [Электронный ресурс]. – Режим доступа: <https://ria-stk.ru/mi/adetail.php?ID=42208>.
4. Аверченков В.И., Самсоненко А.С. Автоматизация управления оптической инспекцией при контроле качества пайки печатных узлов // Вестник Брянского государственного технического университета. [Электронный ресурс]. – Режим доступа: <https://cyberleninka.ru/article/n/avtomatizatsiya-upravleniya-opticheskoy-inspektsiyei-pri-kontrole-kachestva-payki-pechatnyh-uzlov/viewer>.

УДК 004.021

### **А.А. ТРУБИЦЫН, Э.Г. КОЧЕРГИН**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ПРИМЕНЕНИЕ СИСТЕМЫ КОМПЬЮТЕРНОЙ АЛГЕБРЫ «МАТНСАД» ДЛЯ РЕШЕНИЯ ЗАДАЧИ ДИРИХЛЕ МЕТОДОМ КОНЕЧНЫХ ЭЛЕМЕНТОВ**

*Рассматривается решение электростатической задачи Дирихле методом конечных элементов (МКЭ) с помощью системы компьютерной алгебры.*

#### **1. Краткие теоретические сведения.**

Метод конечных элементов (МКЭ) обычно используют при решении задачи Дирихле в случае геометрически сложных областей [1].

Задача Дирихле заключается в следующем: необходимо найти распределение потенциала  $U$  в заданной области  $G$  при известном распределении потенциала  $\varphi$  на границе  $\Gamma$  области  $G$  [1].

### 1.1. Постановка задачи.

На первом этапе записывается уравнение Лапласа в двумерной области  $G$  с границей  $\Gamma$  и граничное условие (ГУ):

$$\begin{cases} \frac{\partial^2 \bar{U}}{\partial x^2} + \frac{\partial^2 \bar{U}}{\partial y^2} = 0, \\ \bar{U}(x, y)|_{\Gamma} = \varphi(x, y). \end{cases} \quad (1)$$

Далее из обеих частей уравнения Лапласа вычитается выражение:

$$\frac{\partial^2 \bar{U}}{\partial x^2} + \frac{\partial^2 \bar{U}}{\partial y^2},$$

и с учетом линейности операции дифференцирования записывается равенство.

$$\frac{\partial^2 (\bar{U} - \varphi)}{\partial x^2} + \frac{\partial^2 (\bar{U} - \varphi)}{\partial y^2} = - \left( \frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} \right) \quad (2)$$

На следующем этапе вводим функцию  $U = \bar{U} - \varphi$  и переформулируем задачу (1):

$$\begin{cases} \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = - \left( \frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} \right), \\ U(x, y)|_{\Gamma} = 0. \end{cases} \quad (3)$$

Представим решение в виде комбинации из некоторых  $N$  линейно независимых функций  $\omega_1(x, y), \omega_2(x, y), \dots, \omega_N(x, y)$ . Решение задачи (3) ищется в виде линейной комбинации на интервале  $[1, n]$ , а  $\varphi$  в виде линейной комбинации на интервале  $[n + 1, N]$ .

$$U(x, y) = \sum_{j=1}^n a_j \omega_j(x, y), \quad (x, y) \in G, \quad (4)$$

$$\varphi(x, y) = \sum_{j=n+1}^N a_j \omega_j(x, y), \quad (x, y) \in \Gamma, \quad (5)$$

Умножив уравнение Пуассона (3) на  $\omega_i(x, y)$  и проинтегрировав получаем:

$$\iint_G \left( \frac{\partial U}{\partial x} \frac{\partial \omega_i}{\partial x} + \frac{\partial U}{\partial y} \frac{\partial \omega_i}{\partial y} \right) dx dy = - \iint_G \left( \frac{\partial \varphi}{\partial x} \frac{\partial \omega_i}{\partial x} + \frac{\partial \varphi}{\partial y} \frac{\partial \omega_i}{\partial y} \right) dx dy. \quad (6)$$

Заменяем  $U$  и  $\varphi$  представлениями в виде линейных комбинаций (4), (5). Введем новое обозначение:

$$(\omega_i, \omega_j) = \frac{\partial \omega_i}{\partial x} \frac{\partial \omega_j}{\partial x} + \frac{\partial \omega_i}{\partial y} \frac{\partial \omega_j}{\partial y}, \quad (7)$$

$$\sum_{j=1}^n a_j \iint_G (\omega_i, \omega_j) dx dy = - \sum_{j=n+1}^N a_j \iint_G (\omega_i, \omega_j) dx dy. \quad (8)$$

### 1.2. Построение разностной схемы.

Помещаем в область  $G$  – некоторое количество  $n$  точек, а на границу  $\Gamma$  – множество  $(N - n)$  точек. Накладываем условия что

$$\omega_j(P_j) = 1, \quad \omega_j(P_i) = 0, \quad j \neq i.$$

Тогда:

$$U(P_j) = a_j, \quad j = 1, 2, \dots, n, \quad (9)$$

$$\varphi(P_j) = a_j, \quad j = n + 1, n + 2, \dots, N, \quad (10)$$

Получаем систему алгебраических уравнений относительно неизвестной  $U(P_j)$ .

$$b_i = \sum_{j=1}^n U(P_j) W_{ij}, \quad (11)$$

$$b_i = - \sum_{j=n+1}^N \varphi(P_j) W_{ij}, \quad (12)$$

$$W_{ij} = \iint_G (\omega_i, \omega_j) dx dy \quad (13)$$

## 2. Программа расчета распределения потенциала в сечении системы электродов методом конечных элементов в среде MathCAD

Сечение системы электродов, в которой необходимо произвести расчет, представлено на рисунке 1.

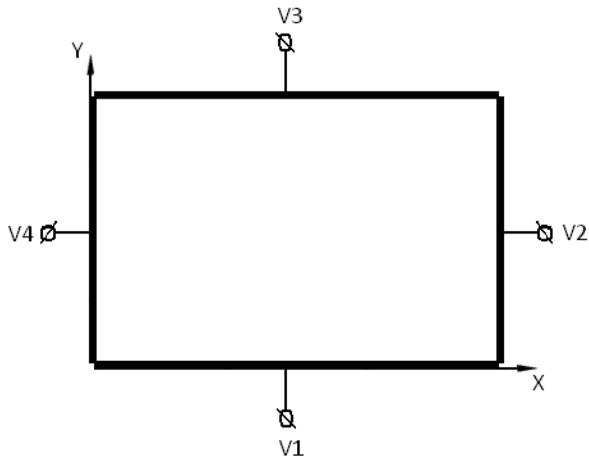
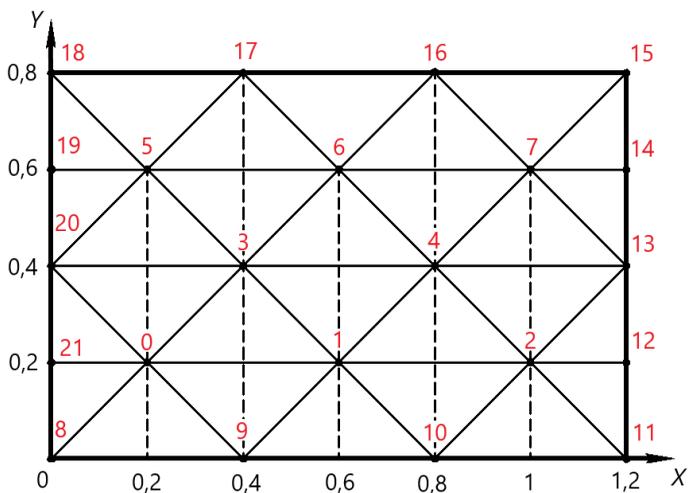


Рисунок 1 – Сечение системы электродов

Расположим в расчетной области  $G$ :  $n = 8$  – точек, а на её границе  $\Gamma$ :  $N - n = 14$ . Далее производим триангуляцию области  $G$ , т.е. разбиваем на конечные элементы (треугольники), вершинами которых являются точки  $N \in [0; 21]$ , рисунок 2.

Рисунок 2 – Триангуляция расчетной области  $G$

Пропишем начальные (входные) данные, которые потребуются для проведения расчетов.

1) Потенциалы на электродах:

Пусть на нижнюю грань прямоугольного сечения подается напряжение  $V_1 = 0.5 В$ , на правую грань:  $V_2 = 1 В$ , на верхнюю грань:  $V_3 = 0.5 В$  и на левую грань:  $V_4 = 0 В$ .

2) Граничные условия (ГУ):

Потенциалы  $\varphi_8, \varphi_9, \dots, \varphi_{21}$  в точках на границе (8, 9, ..., 21) приравняем соответствующим им потенциалам  $V_1, V_2, V_3$  и  $V_4$ .

3) Координаты внутренних и граничных точек:

Для определения координат точек создается две матрицы  $x$  и  $y$ . Координаты точек необходимы для определения площадей, длин оснований, высот и углов получившихся конечных элементов (треугольников).

Координаты всех точек вносятся строго по порядку их номеров.

Занесение входных данных показано на рисунке 3.

| Координаты внутренних и граничных точек: |  | 1. Дано.                         |                             |
|--|--|----------------------------------|-----------------------------|
|  |  | Заданные потенциалы на границах: |                             |
|  |  | $V_1 = 0.5$                      | $V_3 = 0.5$                 |
|  |  | $V_2 = 1$                        | $V_4 = 0$                   |
|  |  | Граничные условия:               |                             |
|  |  | $\varphi_8 = V_1$                | $\varphi_{12} = V_2$        |
|  |  | $\varphi_9 = V_1$                | $\varphi_{13} = V_2$        |
|  |  | $\varphi_{10} = V_1$             | $\varphi_{14} = V_2$        |
|  |  | $\varphi_{11} = V_1$             | $\varphi_{15} = V_3$        |
|  |  |                                  | $\varphi_{16} = V_3$        |
|  |  |                                  | $\varphi_{17} = V_3$        |
|  |  |                                  | $\varphi_{18} = V_3$        |
|  |  |                                  | $\varphi_{19} = V_4$        |
|  |  |                                  | $\varphi_{20} = V_4$        |
|  |  |                                  | $\varphi_{21} = V_4$        |
|  |  | Количество точек:                |                             |
|  |  | $n = 8$                          | Количество внутренних точек |
|  |  | $N = 22$                         | Количество всех точек       |

Рисунок 3 – Входные данные и граничные условия

Далее для определения матричных коэффициентов  $W_{ij}$  необходимо задать функции площади треугольника  $\Delta S_{ijk}$ , длины основания треугольника  $l_{jk}$ , высоты треугольника  $h_{ijk}$ , опущенной из точки  $i$ , и

косинуса угла между высотами  $\cos(h_{ijk}, h_{jki})$ , опущенными из точек  $i$  и  $j$ . Треугольник представлен на рисунке 4.

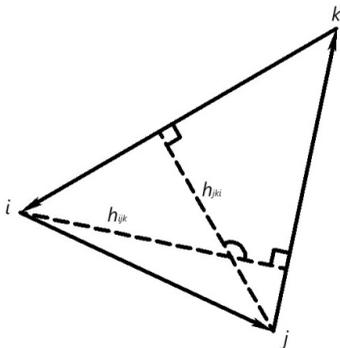


Рисунок 4 – Треугольник  $\Delta_{ijk}$

Все эти функции можно задать, зная координаты точек, и приняв стороны треугольников за вектора. Функции, заданные в среде MathCAD показаны на рисунке 5.

**2. Решение**

**2.1. Нахождение площади треугольников  $S(i,j,k)$**

$$S(i,j,k) := \frac{1}{2} \cdot |(x_j - x_i)(y_k - y_i) - (y_j - y_i)(x_k - x_i)|$$

**2.2. Нахождение длины основания треугольника  $l(j,k)$**

$$l(j,k) := \sqrt{(x_k - x_j)^2 + (y_k - y_j)^2}$$

**2.3. Нахождение высоты треугольника  $h(i,j,k)$  опущенной из точки  $i$  на основание  $l(j,k)$**

$$h(i,j,k) := \frac{2 \cdot S(i,j,k)}{l(j,k)}$$

**2.4 Нахождение косинуса угла  $cs(i,j,k)$  между векторами  $\vec{(k,j)}$  и  $\vec{(k,i)}$**

$$cs(i,j,k) := \frac{(x_k - x_j) \cdot (x_i - x_k) + (y_k - y_j) \cdot (y_i - y_k)}{\sqrt{[(x_k - x_j)^2 + (y_k - y_j)^2] \cdot [(x_i - x_k)^2 + (y_i - y_k)^2]}}$$

Рисунок 5 – Функции  $\Delta S_{ijk}, l_{jk}, h_{ijk}, \cos(h_{ijk}, h_{jki})$

Также перед нахождением матричных коэффициентов  $W_{ij}$  необходимо прописать три блока кода.

В первом блоке выписываются все точки, образующие конечные элементы (треугольники) с внутренними точками  $0, 1, \dots, (n-1)$  в мат-

рицу  $F$ . Количество столбцов такой матрицы будет равно количеству внутренних точек, а количество строк – максимальному количеству точек, образующих треугольники с какой либо из внутренних точек.

В нашем случае количество столбцов равно 8, а количество строк – 6. Номера точек в матрицу заносятся строго по порядку в направлении против часовой стрелки.

Матрица  $F$  показана на рисунке 6.

В первый столбец матрицы  $F$  вносятся все точки образующие треугольники с точкой 0, во второй столбец – все точки образующие треугольники с точкой 1, и т.д.

$$F := \begin{pmatrix} 3 & 4 & 13 & 6 & 7 & 17 & 16 & 15 \\ 20 & 3 & 4 & 5 & 6 & 18 & 17 & 16 \\ 21 & 0 & 1 & 20 & 3 & 19 & 5 & 6 \\ 8 & 9 & 10 & 0 & 1 & 20 & 3 & 4 \\ 9 & 10 & 11 & 1 & 2 & 3 & 4 & 13 \\ 1 & 2 & 12 & 4 & 13 & 6 & 7 & 14 \end{pmatrix} \quad M := \begin{pmatrix} 6 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6 \end{pmatrix} \quad p := M - 1 = \begin{pmatrix} 5 \\ 5 \\ 5 \\ 5 \\ 5 \\ 5 \\ 5 \\ 5 \end{pmatrix}$$

Рисунок 6 – Матрица  $F$

При нахождении  $W_{ij}$  появляется три случая:  $i = j, i \neq j$  и связи между точками  $i$  и  $j$  нет.

Второй блок необходим для нахождения  $W_{ij}$  при  $i \neq j$ . Этот блок позволяет найти индекс  $j$ -ой точки  $Q(i, j)_0$  и  $Q(i, j)_1$ , благодаря чему можно определить индексы точки  $k : Q_k(i, j)$  и точки  $m : Q_m(i, j)$ , которые составляют два треугольника с основанием  $ij$ , рисунок 7.

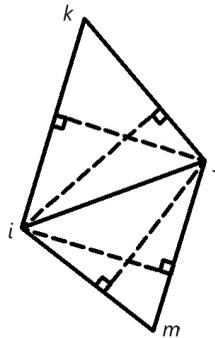


Рисунок 7 – Два треугольника с основанием  $ij$

Второй блок показан на рисунке 8.

$$\begin{aligned}
 Y(i) &:= \text{stack}(F^{(i)}, F^{(i)}) \\
 Q(i, j) &:= \text{match}(j, Y(i)) \\
 Q\_k(i, j) &:= Q(i, j)_0 + 1 \quad Q\_m(i, j) := Q(i, j)_1 - 1
 \end{aligned}$$

Рисунок 8 – Второй блок программы

Третий блок необходим для определения присутствия  $j$ -го элемента в столбце  $i$ .

Третий блок представлен на рисунке 9.

$$\begin{aligned}
 FF(i) &:= \left[ \overrightarrow{(F^{(i)} \geq 0) \cdot F^{(i)}} + (F^{(i)} < 1) \right] \\
 F\_0(i, j) &:= [(FF(i) = j) \cdot FF(i)] + j \cdot (FF(i) = j) \\
 Raz(i, j) &:= \begin{cases} \min(F^{(i)} - F\_0(i, j)) & \text{if } j = 0 \\ \min(FF(i) - F\_0(i, j)) & \text{otherwise} \end{cases}
 \end{aligned}$$

Рисунок 9 – Третий блок

После описания 1, 2 и 3 блоков, возможно быстро рассчитать все матричные коэффициенты  $W_{ij}$ , рисунок 10.

$$W(i, j) = \begin{cases} \frac{1}{h(i, F_{p, i}, F_{0, i})^2} S(i, F_{p, i}, F_{0, i}) + \sum_{d=1}^{p_i} \left( \frac{1}{h(i, F_{d-1, i}, F_{d, i})^2} S(i, F_{d-1, i}, F_{d, i}) \right) & \text{if } i=j \\ \frac{1}{h(j, Y(i)_{Q_k(i, j)})^2} \frac{1}{h(i, j, Y(i)_{Q_k(i, j)})} \cos(i, j, Y(i)_{Q_k(i, j)}) S(i, j, Y(i)_{Q_k(i, j)}) + \frac{1}{h(j, i, Y(i)_{Q_m(i, j)})^2} \frac{1}{h(i, j, Y(i)_{Q_m(i, j)})} \cos(i, j, Y(i)_{Q_m(i, j)}) S(i, j, Y(i)_{Q_m(i, j)}) & \text{if } i \neq j \wedge Raz(i, j) = 0 \\ 0 & \text{otherwise} \end{cases}$$

Рисунок 10 – Формула матричного коэффициента  $W_{ij}$

Далее находим значения параметров  $b_i$  по формуле (12) для точек  $0, 1, \dots, 7$ , рисунок 11.

$$\begin{aligned}
 i &:= 0, 1 \dots n - 1 \\
 &N-1 \\
 b_i &:= \sum_{j=n}^{N-1} (\varphi_j \cdot W(i, j))
 \end{aligned}$$

Рисунок 11 – Нахождение параметра  $b_i$  для точек  $0, 1, \dots, 7$

На финише записывается система уравнений, согласно выражению (11), и рассчитываются искомые значения потенциалов во внутренних точках  $0, 1, \dots, 7$ , рисунок 12.

|   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
|---|------------|------------|------------|------------|------------|------------|------------|-------|-----|-------|-------|-------|-------|-----|-------|
| $U_0 := 0$  | $U_1 := 0$ | $U_2 := 0$ | $U_3 := 0$ | $U_4 := 0$ | $U_5 := 0$ | $U_6 := 0$ | $U_7 := 0$ |       |     |       |       |       |       |     |       |
| <b>Given</b>  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(0,0) + U_1 \cdot W(0,1) + U_2 \cdot W(0,2) + U_3 \cdot W(0,3) + U_4 \cdot W(0,4) + U_5 \cdot W(0,5) + U_6 \cdot W(0,6) + U_7 \cdot W(0,7) = -b_0$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(1,0) + U_1 \cdot W(1,1) + U_2 \cdot W(1,2) + U_3 \cdot W(1,3) + U_4 \cdot W(1,4) + U_5 \cdot W(1,5) + U_6 \cdot W(1,6) + U_7 \cdot W(1,7) = -b_1$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(2,0) + U_1 \cdot W(2,1) + U_2 \cdot W(2,2) + U_3 \cdot W(2,3) + U_4 \cdot W(2,4) + U_5 \cdot W(2,5) + U_6 \cdot W(2,6) + U_7 \cdot W(2,7) = -b_2$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(3,0) + U_1 \cdot W(3,1) + U_2 \cdot W(3,2) + U_3 \cdot W(3,3) + U_4 \cdot W(3,4) + U_5 \cdot W(3,5) + U_6 \cdot W(3,6) + U_7 \cdot W(3,7) = -b_3$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(4,0) + U_1 \cdot W(4,1) + U_2 \cdot W(4,2) + U_3 \cdot W(4,3) + U_4 \cdot W(4,4) + U_5 \cdot W(4,5) + U_6 \cdot W(4,6) + U_7 \cdot W(4,7) = -b_4$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(5,0) + U_1 \cdot W(5,1) + U_2 \cdot W(5,2) + U_3 \cdot W(5,3) + U_4 \cdot W(5,4) + U_5 \cdot W(5,5) + U_6 \cdot W(5,6) + U_7 \cdot W(5,7) = -b_5$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(6,0) + U_1 \cdot W(6,1) + U_2 \cdot W(6,2) + U_3 \cdot W(6,3) + U_4 \cdot W(6,4) + U_5 \cdot W(6,5) + U_6 \cdot W(6,6) + U_7 \cdot W(6,7) = -b_6$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| $U_0 \cdot W(7,0) + U_1 \cdot W(7,1) + U_2 \cdot W(7,2) + U_3 \cdot W(7,3) + U_4 \cdot W(7,4) + U_5 \cdot W(7,5) + U_6 \cdot W(7,6) + U_7 \cdot W(7,7) = -b_7$  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| <b>Find</b> ( $U_0, U_1, U_2, U_3, U_4, U_5, U_6, U_7$ ) =  |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| <table border="1"> <tr><td>0.286</td></tr> <tr><td>0.5</td></tr> <tr><td>0.714</td></tr> <tr><td>0.393</td></tr> <tr><td>0.607</td></tr> <tr><td>0.286</td></tr> <tr><td>0.5</td></tr> <tr><td>0.714</td></tr> </table> |            |            |            |            |            |            |            | 0.286 | 0.5 | 0.714 | 0.393 | 0.607 | 0.286 | 0.5 | 0.714 |
| 0.286   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| 0.5   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| 0.714   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| 0.393   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| 0.607   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| 0.286   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| 0.5   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |
| 0.714   |            |            |            |            |            |            |            |       |     |       |       |       |       |     |       |

Рисунок 12 – Решение системы уравнений

Таким образом, произведен расчет распределения потенциала в плоскости симметрии системы электродов методом конечных элементов, найдены значения потенциалов во всех искомым точках.

Для расчетов достаточно внести значения координат внутренних и граничных точек в матрицы значений  $x$  и  $y$ , граничные условия и номера точек в матрицу  $F$ .

Представленный алгоритм расчета позволяет в реальном времени приближенно решить подобные типовые задачи с простой или сложной геометрией. Точность расчетов регулируется изменением количества внутренних и граничных точек.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Вычислительная техника в научных исследованиях. Численный эксперимент: Методические указания к практическим занятиям/Рязан. гос. радиотехн. акад.: Сост. А.А.Трубицын; Под ред. Э.П.Шеретова. Рязань, 1999. 46 с.

УДК 004.032.26

**С.С. ТУРНАЕВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**ИСКУССТВЕННАЯ НЕЙРОННАЯ СЕТЬ В  
МАШИННОМ ОБУЧЕНИИ**

*Рассматриваются основные определения, связанные с искусственной нейронной сетью, а также ее структура и основные отличия от обычной компьютерной программы, которые позволяют ей решать более сложные задачи.*

Нейронная сеть в машинном обучении, строится по принципу работы человеческого мозга, который состоит из миллионов нейронов. Мозг посылает и обрабатывает сигналы в виде электрических и химических сигналов. Передавать сигналы между собой нейронам позволяет специальная структурная связь, которая называется синапсом. Из большого такого количества нейронов формируются нейронные сети.

Искусственные нейронные сети представляют собой вычислительные алгоритмы, которые используются для имитации поведения биологических систем, состоящих из нейронов. Модель, состоящая из таких алгоритмов должна быть способна к машинному обучению, а также распознаванию образов, полученных на основе входных данных. Математическая модель представляет собой ориентированный граф, каждый узел которого имеет свой весовой коэффициент

Искусственная нейронная сеть обычно состоит из трех слоев, которые в свою очередь состоят из множества взаимосвязанных узлов.

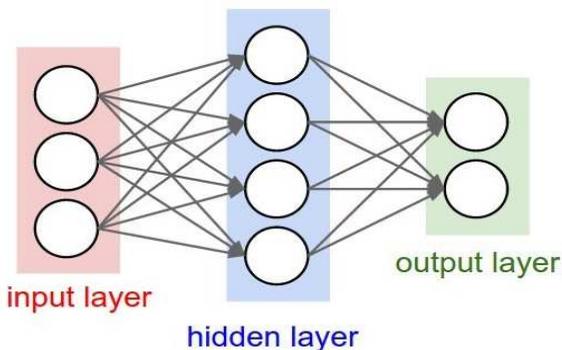


Рисунок 1 – Структура искусственной нейронной сети

Входной слой – это пассивный слой, он получает данные для каждого наблюдения и не обрабатывая их дублирует значения на множество своих выходов.

Скрытый слой – слой, который производит заданные преобразования входных переменных. Обработка выполняется через систему взвешенных соединений. Таких слоев может быть несколько.

Выходной слой – он возвращает выходное значение, соответствующее предсказанию переменной ответа. Правильность полученного результата во многом зависит от правильного выбора весов.

Нейронная сеть, рассмотренная выше называется многослойной, но существует и более простой вид сети, именуемой однослойной. Такая структура представляет собой сеть, в которой сигналы со входного слоя сразу попадают на выходной слой. Входной слой принимает и распределяет сигналы, а вычисления происходят в выходном слое. Такие структуры проще в реализации и могут применяться для простых вычислений.

Кроме количества слоев, нейронные сети можно классифицировать по распределению информации по синапсам:

1. Однонаправленные – сигналы перемещаются строго по направлению от входного сигнала к выходному. Движение сигнала в обратном направлении невозможно.

2. С обратными связями – сигналы двигаются и в прямом, и в обратном направлении. Результат с выхода, способен снова оказаться на входе.

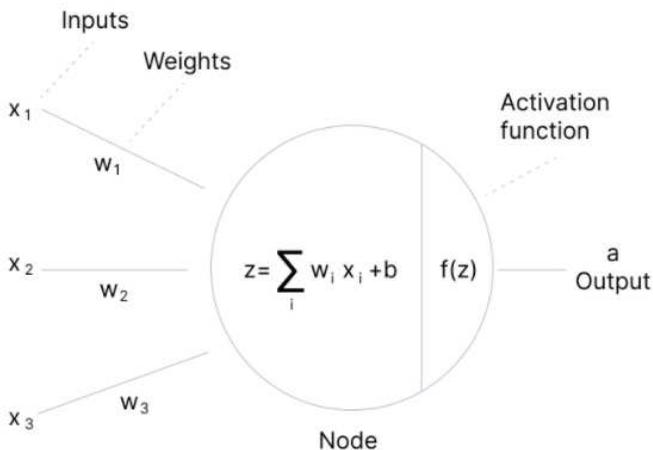


Рисунок 2 – Упрощенная схема обработки входных сигналов

Человеческий мозг анализируя входные обрабатывает и решает, должен ли нейрон быть активирован, в искусственной нейронной сети эту роль выполняет функция активации. Основная ее роль заключается в преобразовании суммированного взвешенного ввода от узла к выходу, либо следующему узлу. Различают три типа функций активации:

1. Ступенчатая функция активации – входное значение сравнивается с определенным пороговым значением, и принимается решение о том, будет нейрон активирован или нет.

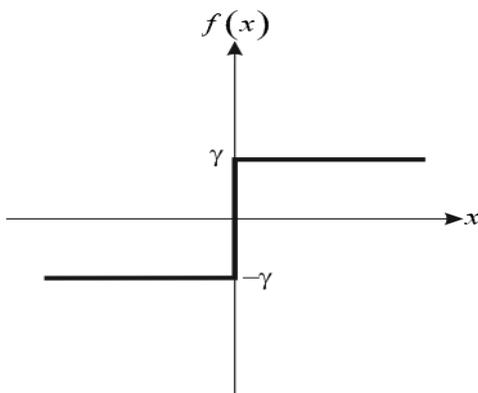


Рисунок 3 – Ступенчатая функция активации

2. Линейная функция активации – входное значение пропорционально выходному, однако у этой функции есть два больших недостатка: невозможность реализовать обратную связь, так как производная этой функции константа, а также независимо от количества скрытых слоев, выходной слой будет являться линейной функцией входного слоя.

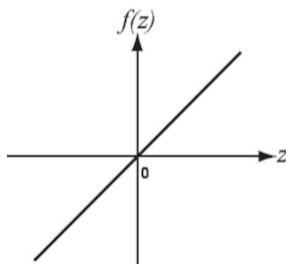


Рисунок 4 – Линейная функция активации

3. Нелинейные функции активации – эти функции позволяют реализовать обратную связь для того чтобы можно было корректировать весовые коэффициенты для достижения лучшего результата. Также, в отличие от линейной функции, появляется возможность использовать множество слоев, которые будут накладываться друг на друга. На рисунке ниже представлены нелинейные функции активации.

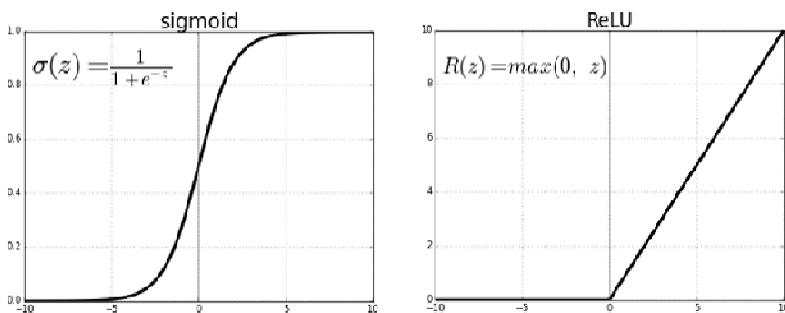


Рисунок 5 – Нелинейные функции активации (сигмоид и усеченная функция)

Искусственные нейронные сети применяются для решения сложных задач с которыми простые компьютерные программы справиться не могут. Такими задачами, могут быть распознавание образов, предсказание следующего шага (в основном используются на фондовом рынке), классификация входной информации по определенным параметрам – так работают кредитные роботы, которые позволяют быстро принять решение об одобрении или отказе кредита.

#### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Хайкин, Саймон. Нейронные сети: полный курс, 2-е издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2006.
2. Ф.М. Гафаров, А.Ф. Галимянов. Искусственные нейронные сети и приложения. – Казань, 2018.
3. Я. Гудфеллоу, И. Бенджио, А. Курвилль. Глубокое обучение, 2-е изд.: пер. с англ. А.А. Силкина.–М: ДМК Пресс, 2017.
4. А.Н.Васильев, Д.А.Тархов. Нейростеовое моделирование. Принципы. Алгоритмы. Приложения. СПб.: Изд-во Политехн. Ун-та, 2009.
5. Д.А.Тархов. Нейронные сети. Модели и алгоритмы. М., Радиотехника, 2005.

УДК 004.65

**Д.А. УКВАСОВА**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**АППРОКСИМАЦИЯ СЛОЖНЫХ ГЕОМЕТРИЧЕСКИХ  
ПОВЕРХНОСТЕЙ РАЗВЕРТЫВАЮЩИМИСЯ  
ПОВЕРХНОСТЯМИ**

*Рассматривается применение метода аппроксимации поверхностей сложной геометрической природы линейчатыми развертывающимися поверхностями. Также в данной статье будут показаны примеры поверхностного моделирования. Важность этого вопроса в настоящее время оправдана. В статье подчеркивается, что аппроксимация линейчатых поверхностей несет в себе важную функциональную задачу и имеет большое практическое значение при проектировании различных устройств.*

В мире абсолютно все объекты имеют определенные характеристики, такие как размер, цвет, поверхность, форма и множество других свойств, необходимых для идентификации их в пространстве. Логично, что одна из самых главных черт предмета - это поверхность. Ведь именно она дает более конкретное представление об изучаемом объекте. Благодаря информации о поверхности, ее строении, форме, неровностях, человек может представить данный объект в пространстве, не путая его с другими экземплярами. Но не каждую поверхность легко поддается описанию, именно поэтому и был разработан метод аппроксимации поверхности.

Аппроксимация (моделирование) поверхностей – научный метод, состоящий в замене одних объектов другими близкими к исходным, но более простыми. В настоящее время поверхностное моделирование используется в различных сферах деятельности, науке и промышленности. Поверхность задается различным и достаточно сложным образом. В результате становится возможно заменить сложную поверхность другими, более простыми поверхностями, свойства которых уже хорошо известны. Это упрощает изучение таких поверхностей. Например, можно преобразовать сложную поверхность кусочно-линейной поверхностью 2-го порядка, развертывающимися поверхностями или простейшими поверхностями вращения.

Более интересно приближение геометрических поверхностей развертывающимися поверхностями. Так как линейчатые поверхности наиболее хорошо изучены и обладают набором определенных свойств, которые широко используются на практике при конструировании. Способность развертываться на плоскость-одно из важных свойств, и

оно имеет большой практический смысл при моделировании различных компонентов и их деталей. Другое не менее важное свойство- постоянство касательной плоскости вдоль всей образующей. Оно значительно упрощает технологический процесс изготовления изделия с развертывающейся поверхностью. В таком случае поверхность детали обрабатывается вдоль всей образующей в прямолинейном направлении. В результате этого метод обработки такого изделия существенно упрощается. Применение такого вида моделирования во многих областях науки наиболее очевидно, и, следовательно, возникает особый интерес и потребность в теоретических исследованиях по данному вопросу. Данная статья призвана ознакомить с общим подходом решения вопроса линеаризации сложных поверхностей более простыми, развертывающимися. Также рассмотрим отдельные случаи. Известно, что положение плоскости в пространстве определяется 3-мя параметрами, два из которых зафиксируем. Третий, в свою очередь, будет меняться по некоторому закону. Предположим, этот третий параметр –это функция некоторой величины  $p: f(p)$ . Присваивая величине  $p$  значения от  $-\infty$  до  $+\infty$ , мы получим бесконечное множество значений третьего параметра, которое вместе с 2-мя первыми постоянными параметрами определяют бесконечное множество плоскостей- однопараметрическое семейство плоскостей. Поверхность, называемая огибающей семейства, есть ни что иное, как эта поверхность, множество касательных плоскостей которой совпадает с множеством плоскостей некоторого семейства. Известно, что линейчатую поверхность или совокупность прямых, зависящую от одного параметра можно описать перемещением прямой (образующей) по заданной линии (направляющей). Линейчатые поверхности могут быть двух типов: развёртывающимися и косыми. Развёртывающаяся поверхность- это такая поверхность, совмещенная с плоской поверхностью с помощью изгиба без образования складок и разрывов. При этом любая такая поверхность будет являться либо конусом, либо цилиндром, либо же поверхностью, состоящей из касательных к какой-либо кривой, находящейся в пространстве. Такая кривая- ребро возврата развёртывающейся поверхности (рисунок 1).

Плоскость  $P$ , которая пересекает ребро возврата  $L$ , образует в сечении с поверхностью некоторую кривую  $ABC$ , где  $B$ -точка возврата. В данном случае, ребро возврата будет являться особой линией развёртывающейся поверхности, где две её плоскости  $S1$  и  $S2$  соприкасаются друг с другом вдоль этой поверхности.

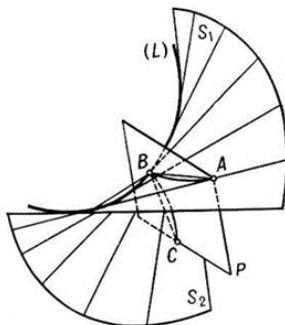


Рисунок 1 – Развёртывающаяся поверхность

Заметим, что развёртывающиеся поверхности характеризуются еще тем, что их касательная плоскость остается неизменной (постоянной) в разных точках одной и той же образующей. Это означает, что совокупность всех касательных плоскостей развёртывающейся линейчатой поверхности представляет собой однопараметрическое семейство. Иначе говоря, развёртывающаяся линейчатая поверхность – огибающая однопараметрического семейства плоскостей. Приведем следующие примеры огибающих. Определим некоторую плоскость  $\beta$ -мя параметрами: точкой  $A$ , которая взята на кривой линии  $L$ , касательной  $P$  к ней в данной точке, и точкой  $B$ , не лежащей на данной линии  $L$ . Затем будем перемещать точку  $A$  по заданной кривой. После перемещения, получим, однопараметрическое семейство плоскостей, огибающей которого будет некоторая коническая поверхность, имеющая вершину в точке  $S$ , и для которой направляющей будет  $L$ . В случае, если точку  $S$  удалить в бесконечность, то все плоскости такого семейства будут перпендикулярны некоторой плоскости  $\beta$ , не принадлежащей этому семейству. В ходе выполнения действий, огибающей нового семейства плоскостей станет цилиндрическая поверхность с образующими перпендикулярными плоскости  $\beta$ , и направляющей для которой будет кривая  $L$ . Следует отметить, что форма и взаимоположение исследуемой пары направляющих могут определить одну или несколько развёртывающихся поверхностей. В большинстве случаев, развёртывающаяся поверхность определяется 2-мя кривыми линиями, для которых каждая касательная плоскость к одной кривой линии касается другой кривой только в единственной точке. У косо́й линейчатой поверхности касательные плоскости различны во всех возможных точках одной и той же образующей. При условии перемещения точки касания вдоль образующей, данная касательная плоскость вращается вокруг этой образующей. Полный поворот касательной плоскости бу-

дет равен  $180^\circ$  (когда точка касания проходит всю образующую). На каждой образующей есть такая точка, которая делит ее на две части, и для каждой из этих двух частей полный поворот касательной равен  $90^\circ$ . Точка  $O$  будет являться центром образующей. (рисунок 2).

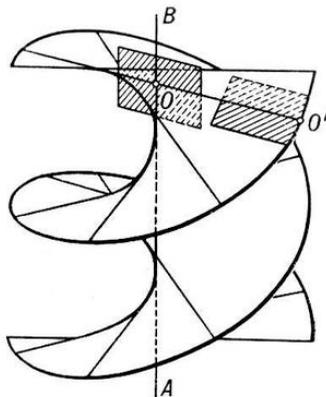


Рисунок 2 – Косая линейчатая поверхность

Отсюда следует, что тангенс угла между плоскостями, касающимися поверхности в центре  $O$ , или в некоторой точке  $O'$  этой образующей, пропорционален расстоянию  $OO'$ . Коэффициент пропорциональности будет являться параметром распределения линейчатой поверхности. Следовательно, абсолютная величина полной кривизны линейчатой поверхности убывает при удалении от центра образующей, и достигает наибольшего значения в центре данной образующей. Линией сжатия или стрикционной линией в данном случае будет являться геометрическое положение центров образующих. Например, у геликоида — линейчатой поверхности, которая описывается равномерным винтовым движением прямой вокруг некоей оси, пересекаемой движущейся прямой под прямым углом, такой линией сжатия является ось  $AB$  (рис. 2). Кроме того, если рассматривать такие линейчатые поверхности 2-го порядка, как гиперболический параболоид, однополостный гиперболоид, то видно, что они имеют две различные системы прямолинейных образующих. Только поверхность вращения 2-го порядка имеет две системы прямолинейных образующих. Как известно, параболоид можно охарактеризовать как незамкнутую нецентрально (т.е. не имеющая центра симметрии) поверхность 2-го порядка. В частном случае эллиптический параболоид можно получить с помощью движения параболы, ветви которой направлены вниз, по параболе, ветви которой направлены вверх.

Необходимо подчеркнуть, что у линейчатых поверхностей, изгибаемых друг на друга, в следствии процесса качения будет одна общая образующая. Именно на этом свойстве основано применение развертываемой поверхности в науке, производстве, архитектуре. В современном мире корпуса многих транспортных средств, конструкторских устройств и различных типов сооружений – это системы и комплексы сложно образованных поверхностей. Линейчатые поверхности исследуются и применяются во всех сферах человеческой деятельности: в технике, инженерии, при проектировании государственных архитектурных зданий, а также дорожных магистралей. Следовательно, изучение методов аппроксимации и их использование в современной жизни является актуальным вопросом, обусловленным востребованностью линейчатых развертываемых поверхностей, сочетающих в себе набор положительных качеств.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Постников М.М. Дифференциальная геометрия, – М., 1988. – 58с.
2. Позняк Э. Г., Шикин Е.В. Дифференциальная геометрия: Первое знакомство, –М., 2003. – 10 с.
3. А.П.Норден, Краткий курс дифференциальной геометрии, Москва, “Наука”, 1962. – 24 с.
4. Н.В. Грищенко, С.А. Семериков, А.А. Хараджян, Е.В. Чернов Сравнительный анализ методов аппроксимации, – Кривой Рог, 1998. – 4 с.

УДК 004.421

### Н.О. ФИРСОВ

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### ТРАССИРОВКА МНОГОСЛОЙНЫХ ПЕЧАТНЫХ ПЛАТ АЛГОРИТМОМ ХЕЙСА

*В статье рассматривается трассировка многослойных печатных плат с помощью алгоритма Хейса. Приведен пример использования данного алгоритма.*

Трассировка – важная конструкторская задача, которая заключается том, чтобы проложить соединение между контактами и избежать их пересечений.

При трассировке многослойных коммутационных схем часто не удается иметь в качестве рабочего поля одну плоскость. Благодаря чему рассмотрим алгоритм Хейса, который является обобщением алгоритма Ли и позволяет осуществлять построение соединений в пространстве.

Алгоритм Хейса один из немногих алгоритмов, который подходит для трассировки МПП. Он осуществляет поиск кратчайшего пути в многослойном ДРП между двумя заданными точками.

Для каждого слоя коммутации вводится дискретное рабочее поле ДРП $i$ , где  $i$  – номер слоя. Считается, что ячейки, принадлежащие разным слоям и лежащие друг на друге, могут быть соединены переходами во время трассировки.

Алгоритм Хейса включает в себя процедуру распространения числовой волны для каждого ДРП $i$  и процедуру проведения пути.

В процессе поиска пути запрещены пересечения уже созданных соединений. Следовательно, ячейки ДРП $i$  могут быть заняты и свободны. Занятой считается ячейка, в которой проходит маршрут определенного соединения, расположен переход, контактная площадка и т.д. Когда волна распространяется, свободным ячейкам ДРП $i$  ( $i = 1, 2, \dots, n$ ) присваивается индекс длины  $P_i$  и индекс количества переходов  $U_i$ .

При вычислении индекса длины  $P_i$  переходы учитываются при добавлении  $K$  единиц длины для каждого перехода. Индекс  $U_i$  используется в процессе проведения пути для минимизации переходов.

В процессе распространения волны для каждого из слоев  $i$  используются такие массивы как: ДРП $i$  – состояния ячеек  $i$ -го слоя,  $L_i$  – ячейки текущего волнового фронта в  $i$ -м слое,  $M_i$  – ячейка слоя  $i$ , смежная к ячейкам из  $L_i$ . При формировании очередного фронта  $i$ -го слоя, а также с ячейками из  $M_i$ , используются те свободные ячейки  $i$ -го слоя, в которых можно осуществить переход из других слоев и имеющие тот же индекс  $P$ .

В связи с тем, что алгоритм Хейса для многослойных схем требует приличного количества времени и памяти, рассмотрим его реализацию для двухслойных схем. Чтобы уменьшить размер массивов ДРП $1$  и ДРП $2$  будем использовать кодировку индексов  $P_1$  и  $P_2$  по модулю 3.

В самом начале алгоритма ячейка-источник  $A$  вводится в списки  $L$  и (или)  $L_2$ ; индекс длины  $P$ , а также индексы переходов  $U_1$  и  $U_2$  будут равны нулю. Списки  $M_1$  и  $M_2$  – пока что пустые. Цена одного перехода будет равна единице длины.

Очередной фронт формируется следующими операциями:

- 1) Увеличить индекс  $P$  на единицу;
- 2) Исследовать в ДРП<sub>1</sub> окрестность очередной ячейки списка  $L_1$ , выбрать свободные и немаркированные ячейки и записать их в список  $M_1$ ;
- 3) Отметить ячейки в ДРП<sub>1</sub> индексом  $P$  и индексом  $U$ , равным соответствующему индексу очередной ячейки из  $L_1$ ;
- 4) Повторить п.2 и п.3 для всех ячеек списка  $L_1$ ;
- 5) Очистить  $L_1$  и передать туда содержимое списка  $M_1$ , затем очистить список  $M_1$ ;
- 6) Добавить в список  $L_1$  те ячейки из списка  $L_2$ , в которых возможен переход из слоя 2 в слой 1 и которые не были отмечены в ДРП<sub>1</sub>, увеличив  $U_1$  на единицу;
- 7) Отметить эти ячейки в ДРП<sub>1</sub> индексом  $P$  и индексом  $U_1$ .
- 8) Выполнить п.2,...,п.7 для слоя 2, используя соответствующие списки.
- 9) Если  $B$  не найдена в списке  $L_1$  или  $L_2$  и оба списка  $L_1$  и  $L_2$  не пустые, повторить операции п.1 – п.8. Если списки  $L_1$  и  $L_2$  пустые, то искомым путь просто не существует. Если  $B$  обнаружена, то начать построение пути.

На рисунке 1 отображено распространение волны из ячейки  $A$ , отмеченной как источник на обоих слоях. Указаны значения индекса  $P$ . Ячейка  $B$  обнаруживается на 15-м шаге в слое 1.

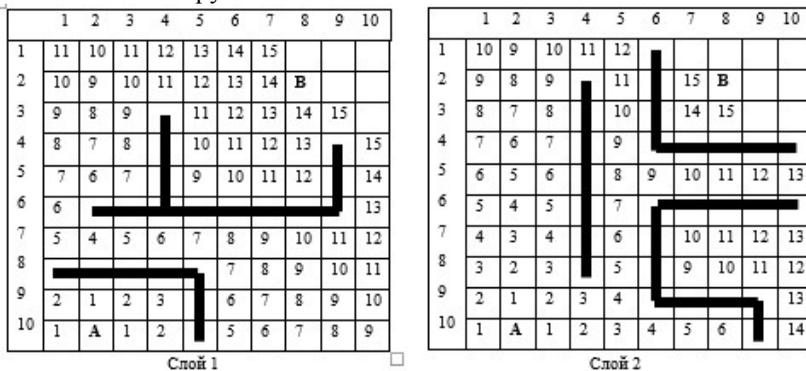


Рисунок 1 – Распространение волны

Чтобы построить путь нужно из ячейки  $B$ , идти в сторону уменьшения индекса  $P$ , прежде чем попасть в ячейку  $A$ . Если на каком-то шаге в окрестности ячейки есть две ячейки с одинаковым  $P$ , то выбирается ячейка, индекс  $U$  которой меньше. Если мы подходим к ячейке, вблизи которой нет ячейки с меньшим  $P$ , то нам нужно перейти на другой слой и продолжить путь.

На рисунке 2 показан путь для этого примера. Он имеет длину 15, а также переход в ячейке (5, 5).

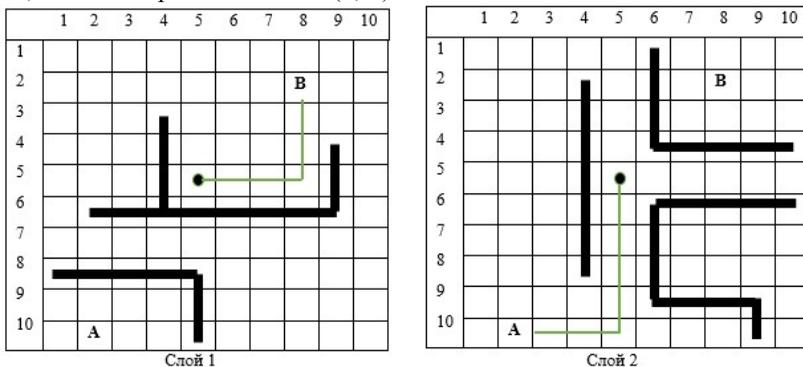


Рисунок 2 – Построенный путь

Следует отметить, что при использовании буферных списков для возможных переходов мы можем увеличить цену одного перехода до  $n$  единиц. В такой ситуации можно создать путь с меньшим количеством переходов, но большей длиной.

На рисунке 3 показана программная реализация алгоритма Хейса.

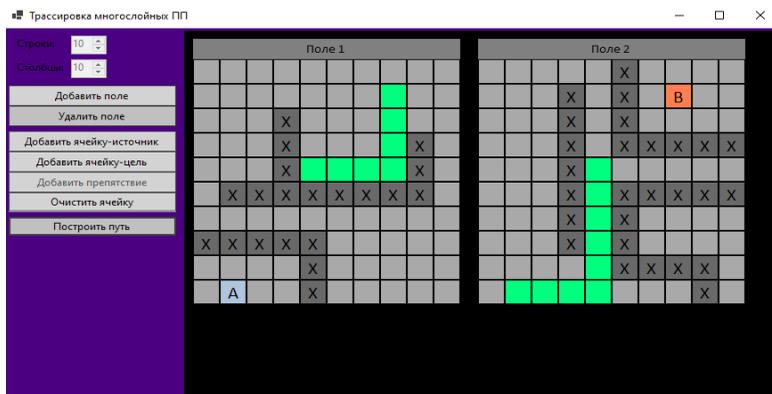


Рисунок 3 – Программная реализация алгоритма

В алгоритме Хейса волна распространяется независимо в каждом из  $n$  слоев схемы. Это приводит к значительным затратам времени и памяти при реализации алгоритма на ЭВМ. Кроме того, алгоритм Хейса всегда строит путь, если он существует, и этот путь всегда имеет минимальную длину.

Рассмотренный алгоритм является наиболее популярным при

трассировке многослойных печатных плат, но, несмотря на это, у него есть свои преимущества и недостатки, а также своя альтернатива – алгоритм Джейера. Этот алгоритм быстрее, чем алгоритм Хейса, благодаря одновременному исследованию всех  $n$  слоев схемы. Но, поскольку ячейка ДРП исключается из рассмотрения после исследования, алгоритм Джейера не гарантирует нахождения пути, если такой путь вообще существует.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Селютин В.А. Машинное конструирование электронных устройств / В.А. Селютин. – М.: Изд-во Советское радио, 1977. – 384 с.
2. Курейчик В.М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР / В.М. Курейчик. – М.: Изд-во Радио и связь, 1990. – 352 с.
3. Shopov G. Proteus PCB design example /G. Shopov 2018. – 389 с.

УДК 004.65

### И.В. ФОКИН

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### СИСТЕМА ШТРИХКОДИРОВАНИЯ НА ПРОИЗВОДСТВО

*Рассматривается метод кодирования информации о характеристиках продукции, позволяющий при помощи специального оборудования считать их. Технология штрихкодирования подразумевает уникальность каждого штрихкода, поэтому необходимо централизованное распределение кодов.*

Система автоматической идентификации продукции разработана для внедрения современных информационных технологий в компании и представляет собой программно-аппаратный комплекс, реализующий автоматическую идентификацию готовой продукции с использованием технологии штрихового кодирования.

Современные предприятия, особенно производственные, сталкиваются с большей нехваткой скорости выполнения основных операций. Сегодня такие задачи решаются с помощью систем управления процессами. Использование штрих-кодов чрезвычайно эффективно и экономично. Автоматизация здесь уже используется на этапе подготовки и ввода данных. С развитием применения ЭВМ не только для сложных научных и экономических расчетов, но и в повседневной ра-

боте компаний особое внимание уделяется все тому же «человеческому фактору» - ввод информации для дальнейшей механической обработки продолжается до сих пор человеком.

Технология штрих-кода означает уникальность каждого штрих-кода, поэтому коды должны распространяться централизованно. Существуют следующие типы:

- Код UCC/EAN-128.
- Чередование 2 на 5 (ITF) представляет собой полноразмерный штрих-код с высокой плотностью и переменной длиной. Требования этого кода определены в ГОСТ.
- Двумерные символы предназначены для кодирования больших объемов информации.

В двумерном коде можно хранить до 3 КБ данных.

Штриховое кодирование — это метод автоматизированного сбора данных, при котором источником данных является линейный код с чередующимися штрихами и пробелами разной ширины. При этом высота грифеля выбирается только для удобства чтения, которое осуществляется с помощью специальных оптических устройств — сканеров, часто называемых бар-сканерами.

Луч считывателя должен проходить через каждую полосу кода, чтобы она могла прочитать закодированную в ней информацию, поэтому ее высота должна быть достаточной.

Для облегчения конструкции за единицу толщины берут самый узкий штрих или пробел, называют модулем и используют в качестве основы. Другие штрихи или промежутки создают модули 2 и 3, т.е. толщина самого узкого штриха или промежутка равна 2 и 3. Поэтому все остальные поперечные линейные размеры штрихов и пробелов представляют собой сумму этих модулей. Каждая цифра кода EAN представляет собой комбинацию двух штрихов и двух пробелов. Ширина модуля является фиксированным значением для конкретного штрих-кода, хотя для одного и того же стандарта штрих-кода могут использоваться модули разных размеров, что может привести к получению разных изображений. Такой способ создания штрих-кодов значительно упрощает печать и последующее считывание. Модуль может варьироваться от 0,264 мм до 0,660 мм, особенно в символике уже упомянутого кода EAN-13.

Символ кодовой строки обычно состоит из четырех частей:

- сочетания «начальных» элементов, обозначающих начало слова и определяющих порядок чтения;
- набор информационных элементов, т.е. признаков, представляющих данные;

– сочетание элементов одного или нескольких контрольных символов, позволяющее осуществлять автоматическую проверку правильности считывания и расшифровку зашифрованных данных;

– сочетания «концевых» элементов, обозначающих конец слова.

Процесс нанесения штрих-кода может осуществляться различными способами, также можно оценить технологию, используемую для картирования заводских данных. Его можно распечатать отдельно на этикетках, а затем нанести. Третий вариант — вставить наклейки, которые уже есть в наличии на момент получения.

Использование систем штрихового кодирования на производстве и предприятии в целом необходимо интегрировать в процессы производства и хранения с целью снижения затрат на учет выпускаемой продукции и автоматизации производственных процессов (в том числе с применением станков с ЧПУ).

Системы штрих-кодов разрабатываются в зависимости от процессов производства и продаж с учетом особенностей каждой компании.

Каждое производственное предприятие обычно имеет несколько этапов (или процессов):

- 1) принять заказ;
- 2) производственное планирование;
- 3) выполнить заказ;
- 4) хранение;
- 5) отгрузка готовой продукции.

Все эти процессы должны осуществляться в единой информационной среде (системе учета компании), чтобы все участники процесса были своевременно проинформированы и не был исключен ручной повторный ввод данных.

Таким образом, эта система представляет собой полный цикл: получить заказ клиента, запустить его в производство, заполнить заказ производством, упаковать, отправить готовый заказ клиенту.

К каждому отдельному набору производственных функций и задач автоматизации подбираются устройства (сканеры штрих-кодов, терминалы сбора данных, принтеры штрих-кодов и т. д.), которые должны соответствовать требованиям и задачам.

Проектирование и установка интегрированной системы требует первоначальных проектных исследований, результатом которых является документирование проблемы, оборудования, программного обеспечения и специального программного обеспечения, необходимого для решения проблемы.

Преимущества автоматического распознавания открыты для всех уровней и отраслевых компаний. Они заключаются в расширении объемов продаж, повышении производительности, снижении затрат и повышении уровня услуг.

### **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Коздюк Н.В. Складской учет и аудит. М.: «Экспертное бюро», 2001.
2. Липкин Е. ИНДУСТРИЯ 4.0: Умные технологии – ключевой элемент в промышленной конкуренции М.: ООО «Остек-СМТ», 2017.
3. Красовский П.А., Ковалев А.И., Стрижов С.Г. Товар и его экспертиза. М., Центр экономики маркетинга, 240 с., 1999.

УДК 004.932

**М.Р. ФОШИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ИСПОЛЬЗОВАНИЕ СРЕДЫ МАТЛАВ ДЛЯ ОБРАБОТКИ ТЕРМОГРАММ С ЦЕЛЬЮ ВЫЯВЛЕНИЯ УТЕЧЕК ТЕПЛОВЫХ СЕТЕЙ**

*Разрабатывается программа для обработки тепловизионного изображения, полученного в форматах JPG, PNG, TIF, GIF для дальнейшего анализа теплопотерь.*

Тепловизионная диагностика проводится в целях обнаружения повреждений и дефектов исследуемых объектов. По результатам тепловизионной съемки должна быть составлена карта дефектов, выявленных при мониторинге тепловых сетей. При подборе правильного алгоритма обработки тепловизионного изображения увеличивается мобильность в устранении утечек тепловой сети. Поскольку нахождение места теплопотери требует длительное время, алгоритм обработки поможет незамедлительно принять решение об устранении данной утечки в конкретном месте теплосети.

При рассмотрении задачи цифровой обработки относительно тепловизионных изображений следует отметить, что у большинства тепловизоров предусмотрено программное обеспечение (ПО) для дальнейшего анализа результатов. Однако компьютерные программы в основном ориентированы на определённые модификации тепловизоров и формат bmt. Данную тему можно считать актуальной, поскольку

часто возникает необходимость обработки полученной термограммы в формате JPG [1]. Таким образом, целью работы является разработка программы, с помощью которой происходит определение самой горячей и холодной точек термограмм, полученных в формате JPG.

В качестве решения поставленной задачи обработки тепловизионного изображения целесообразно использование среды MATLAB, поскольку она является удобным инструментом для моделирования и исследования методов обработки изображений. При этом актуален программный пакет «Image Processing Toolbox», который включает в свой состав большое количество функций, реализующих распространенные методы обработки изображений. Он является приложением программной среды MATLAB [2].

Отметим, что с помощью функций программы MATLAB можно в реальном масштабе времени проводить мониторинг отопительной системы и эффективно анализировать полученные данные о температурном рельефе объектов, в том числе в динамике его изменения.

Результат выполнения программы обработки тепловизионного изображения представлен на нижеследующих рисунках 1...7.

Начальный вид окна просмотра представлен на рисунке 1.



Рисунок 1 – Окно после запуска программы

На панели находится 3 кнопки: «Загрузка изображения», «Температура в точке» и «Самая горячая точка». Нажатием первой кнопки производится выбор исследуемого изображения.

Вид окна после выбора изображения представлен на рисунке 2.

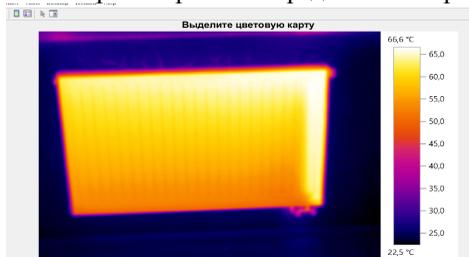


Рисунок 2 – Выделение шкалы температур

В поле появившегося окна выделяем температурную шкалу справа. Чем точнее будет выбрана данная область, тем будут точнее результаты дальнейшей обработки.

Затем, следует выбрать интересующую нас область для дальнейшего мониторинга температуры (см. рисунок 3).

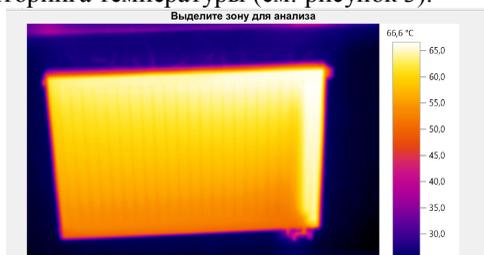


Рисунок 6 – Выбор области для анализа

Допустим, нас интересует температура нагрева труб в нижней правой части отопительного прибора.

В случае, если на тепловизионном изображении не указаны численные отметки температуры, диапазон её измерения вводится вручную. В данном случае, эти данные можно посмотреть на шкале справа (см. рисунок 4).



Рисунок 7 – Ввод температурного диапазона

Выбранная область изображения приведена на рисунке 5.

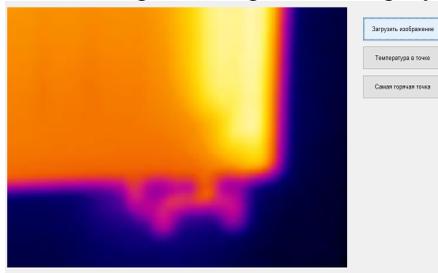


Рисунок 8 – Выбранная область термограммы

Найдем температуру в интересующей нас точке (см. рисунок 6).

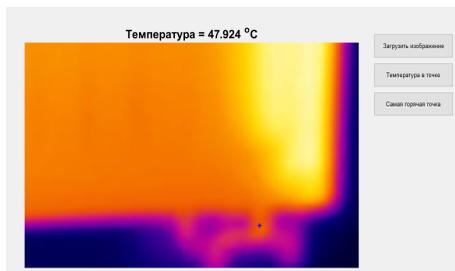


Рисунок 9 – Температура в выбранной точке

Также реализована возможность автоматически определить температуру самой горячей точки исследуемой области (см. рисунок 7).

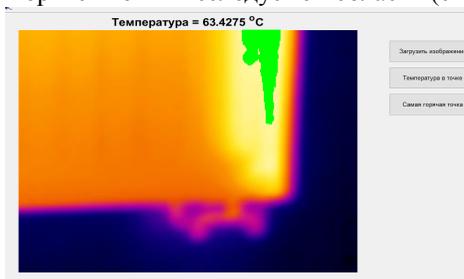


Рисунок 10 – Самая горячая точка исследуемой области

Разработанная компьютерная программа предназначена для проведения исследований теплооборудования, в данном случае — диагностики отопительных приборов.

Для проведения эксперимента использовался ноутбук «MagicBook Pro» фирмы «Honor» с 4-х ядерным процессором «AMD Ryzen 5 3500H», с тактовой частотой 2,10 ГГц и операционной системой «Windows 10 Home», 64 бит.

Таким образом, результатом работы является решение задачи обработки тепловизионных изображений, представленных не только в формате JPG, но и в других популярных форматах (PNG, TIF, GIF), что расширяет возможности обработки термограмм с целью выявления утечек тепловых сетей и для иных научно-практических задач. Созданные программные средства способны эффективно взаимодействовать с широким кругом тепловизионных приборов, сохраняющих термограммы в различных форматах. Это существенно упрощает формирование отчетов по тепловизионной диагностике различных объектов и анализ температурных карт отопительных приборов.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Андреев В.Г., Фошина М.Р. Цифровая обработка тепловизионных изображений в среде MATLAB: научная статья // Международный научно-исследовательский журнал. 2021. № 3 (105). ч. 1. С. 13-19.
2. MATLAB. The Language of Technical Computing. Using MATLAB. The Math Works, Inc. USA, 2000. 187 p.

УДК 004.9

**Т.А. ЦЕДИЛИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **КОНСТРУКТИВНОЕ ОПИСАНИЕ УСТРОЙСТВА, ПРЕДНАЗНАЧЕННОГО ДЛЯ ПРИЕМА ТЕЛЕМЕТРИЧЕСКОЙ ИНФОРМАЦИИ, ПРЕОБРАЗОВАНИЯ И ВЫДАЧИ ЕЁ НА СРЕДСТВА РЕГИСТРАЦИИ**

*В статье рассматривается создание устройства предназначенного для работы с телеметрией и его актуальность в настоящее время.*

Телеметрия – это область науки и техники, занимающаяся вопросами разработки и эксплуатации комплекса автоматизированных средств, обеспечивающих получение, преобразование, передачу по каналу связи, прием, обработки и регистрацию измерительной информации и информации о событиях с целью контроля на расстоянии состояния и функционирования технических и биологических систем различных объектов и изучения явлений природы.

Системы передачи телеметрической информации объединяют бортовую и наземную телеметрические системы. А также радиолинию космический аппарат – Земля. Бортовая телеметрическая система обеспечивает сбор информации от различных служебных и научных датчиков, преобразует выходные сигналы этих датчиков в цифровую форму, обеспечивает хранение информации в период между сеансами связи и формирование единого цифрового потока двоичных данных.

Создание малогабаритного устройства, предназначенного для приема телеметрической информации, преобразования и выдачи её на средства регистрации является актуальной научно-технической задачей. Цель изобретения - упрощение и повышение достоверности результатов. Достигается это путем замены процедуры измерения частоты в пределах каждого из радиоимпульсов методом периодомера на

процедуру обнаружения неперекрывающихся во времени радиоимпульсов с известным числом градаций частоты заполнения на фоне шумов и помех. Данное устройство является частью самостоятельного комплекса АРМ ТМИ.

Устройство предназначенное для приема телеметрической информации, преобразования и выдачи её на средства регистрации вместе с функциональным программным обеспечением исправно выполняет все поставленные функции:

1. Подготовку исходных данных для обработки и отображения результатов обработки телеметрической информации;
2. Прием и регистрацию полного потока телеметрической информации;
3. Обработку заданных параметров принимаемой телеметрической информации и отображение результатов на экране монитора в режиме реального масштаба времени;
4. Обработку и оценку принятой телеметрической информации и отображение результатов.
5. Архивирование и хранение зарегистрированных на АРМ ТМИ файлов полного потока телеметрической информации;
6. Выдачу результатов обработки телеметрической информации на русском и английском языках.

Разработка устройства ведется с учетом требований

1. Максимальная простота конструкции
2. Минимальная стоимость создания
3. Минимальная стоимость изготовления монтажных работ
4. Максимальное использование стандартных унифицированных и заимствованных сборных единиц и деталей

Готовое устройство является малогабаритным переносным, законченным блоком настольного размещения. Для удобства переноски предусмотрены боковые ручки и для устойчивости прорезиненные ножки.

Выбраны конструкционно-изоляционные материалы покрытия комплектующие изделия. В качестве материала корпуса была выбрана сталь 20, толщиной 1.6 мм оцинкованная и покрытая эмалью. Это обеспечит механическую прочность, требуемую надежность и безопасную работу в заданных условиях эксплуатации.

К подтверждению выше сказанного были проведены расчеты показательных коэффициентов надежности и вибропрочности и тепло-

го режима, эти показатели соответствуют норме и гарантируют исправность работы.

При разработке конструкторской документации ставится задача обеспечения допустимого теплового режима, т.е. для всех элементов должно выполняться условие границ работоспособности в рабочем диапазоне температур окружающей среды, в течение всего периода эксплуатации и при всех возможных режимах работы устройства. Расчет теплового режима показал, что использовать принудительное охлаждение на практике не требуется.

Оценка надежности проведена на этапе разработки рабочей конструкторской документации для получения расчетных значений показателей надежности изделия с целью проверки их соответствия заданным в техническом задании требованиям надежности. Оценка проведена на основе информации о составе, структуре, функциях и режимах работы изделий и сведений о надежности его составных частей.

Расчет на надежность проводится для элемента печатного монтажа, т.к. основной критический элемент устройства – это печатная плата или ячейки. А так же расчет этот показал, что у проектируемого устройства хороший запас наработки на отказ.

Современные устройства испытывают целый ряд механических воздействий, которые влияя на работу аппаратуры, снижают её надежность. Расчет на вибропрочность проводят с целью проверки способности изделий противостоять разрушающему действию вибрации и сохранять свои параметры после воздействия в пределах значений, указанных в техническом задании и программе испытаний на изделия.

Расчет на вибропрочность свидетельствует о том, что проектируемое устройство при вибрационных испытаниях будет работать исправно. Так же спроектирован технологический процесс сборки печатного узла и рассчитана технологичность конструкции, что характеризует готовность изделия производству.

Электропитание АРМ осуществляется от сети переменного электрического тока частотой  $(50 \pm 1)$  Гц, напряжением  $(220 \pm 22)$  В.

Особое внимание было уделено изучению прикладных программ используемых как при проектировании, так и создании рабочей конструкторской документации.

PTC Creo — набор функциональных приложений для автоматизированного 2D- и 3D-моделирования, схематического проектирования, визуализации и просмотра технических иллюстраций. С помощью

этого пакета были созданы 3х мерные модели всех деталей и собраны в общую сборку в масштабе 1:1.

Применение объемного моделирования позволило более реалистично представить устройство и улучшить его дизайн. При проектировании вероятность проявления ошибок на этапе производства опытного образца снижается, что ранее положительно сказалось на сроках и стоимости экспериментально-конструкторских работ.

Еще одним из главных достоинств 3д моделирования является простота создания чертежей. Это также исключает неточность и позволяет сэкономить на затраченном времени. При разработки конструкторской документации чертежи были переведены в систему проектирования КОМПАС 3D. По мнению многих конструкторов данная программа является лучшей программным продуктом для традиционно 2D проектирования и очень удобным средством выполнения проектов по стандартам РФ.

Внешний вид устройства представлен на рисунке 1.

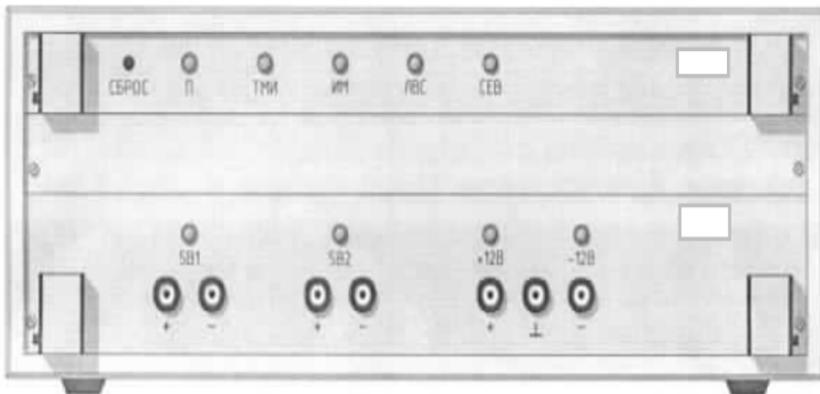


Рисунок 1 – Внешний вид устройства

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Некрасов М.В. система многопоточной обработки телеметрической информации космического аппарата в центрах управления полётами.
2. Статья «Профессиональная 3D-САПР для проектирования и конструирования изделий» [Электронный ресурс]. – Режим доступа: <https://www.autodesk.ru/products/inventor/overview> (дата обращения: 10.04.2022).

УДК 621.391.8

**В.А. ЧАН**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**АЛГОРИТМ СПЕКТРАЛЬНОГО АНАЛИЗА СИГНАЛОВ  
ЛИДАРНЫХ ОТРАЖЕНИЙ НА ФОНЕ  
КУСОЧНО-СТАЦИОНАРНЫХ ПОМЕХ**

*Рассматривается упрощённый адаптивный алгоритм спектральных оценок сигналов лидарных отражений на фоне аддитивных кусочно-стационарных помех, который даёт возможность уменьшить в 2,56...4,02 раза невязку между контрольным и модельным спектрами по сравнению с известными методами спектральных оценок с помощью авторегрессионного метода.*

В настоящее время, характеризующееся бурным развитием науки и техники, широко применяется технология обработки сигналов лазерных систем зондирования природных сред. Проблема уменьшения негативного влияния аддитивного кусочно-стационарного по мощности белого шума является актуальной задачей спектрального анализа во многих прикладных областях, в частности в системах обработки лидарных отражений от природных сред. Для решения обозначенной проблемы предлагается построить адаптивный алгоритм обработки световых отражений для повышения эффективности оценивания их спектра.

Результат моделирования отражений лидарного сигнала  $\mathbf{X}$  от наблюдаемого объекта может быть представлен в виде кусочно-стационарного случайного коррелированного процесса [1]:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}^{(1)} \\ \text{---} \\ \mathbf{X}^{(2)} \end{bmatrix}, \quad (1)$$

где  $\mathbf{X}^{(1)}$  и  $\mathbf{X}^{(2)}$  — два различных по своим статистическим свойствам фрагмента процесса  $\mathbf{X}$ . В дальнейшем предполагается, что в реализации  $\mathbf{X}^{(1)}$  воздействия шумов невелики, а последовательность  $\mathbf{X}^{(2)}$  наблюдений сильно зашумлена.

Матрицы автокорреляции  $\mathbf{R}_1$ ,  $\mathbf{R}_2$  для гауссовской огибающей унимодального энергетического спектра отражений лидарного сигнала и различных зашумлений (с относительными мощностями шума  $P_n^{(1)}$  и  $P_n^{(2)}$  соответственно) имеют вид:

$$\begin{cases} \mathbf{R}_1 = \tilde{\mathbf{R}}_c + P_n^{(1)}\mathbf{I}, \\ \mathbf{R}_2 = \tilde{\mathbf{R}}_c + P_n^{(2)}\mathbf{I}, \end{cases} \quad (2)$$

где  $\tilde{\mathbf{R}}_c = [\tilde{R}_{c,j,k}] = \exp[-(\pi\Delta FT|j-k|)^2/2,8]$  —  $(p+1)\times(p+1)$ -мерная корреляционная матрица с дисперсией, нормализованной к единице;  $j, k=1, 2, \dots, p$ ;  $\mathbf{I}$  —  $(p+1)\times(p+1)$ -мерная единичная матрица;  $\Delta F$  — ширина полосы энергетического спектра моды;  $T$  — период между дискретными наблюдениями (выборками);  $P_n^{(1)}$ ,  $P_n^{(2)}$  — относительная мощность некоррелированного шума изменений со временем обработки. Известно [2], что матрицы автокорреляции  $\mathbf{R}_1$ ,  $\mathbf{R}_2$  имеют эрмитову и теплицевую формы.

Отметим, что вектор  $\mathbf{a}$  авторегрессии AP-модели получается из решения системы линейных уравнений Юла – Уолкера, которые удобно выразить в следующем виде, эквивалентном известному виду [2]:

$$\mathbf{a} = -\mathbf{R}^{-1}\mathbf{i}, \quad (3)$$

где  $\mathbf{R}$  —  $(p+1)\times(p+1)$ -мерная автокорреляционная квадратная матрица,

$\mathbf{i}$  —  $(p+1)$ -мерный крайний левый вектор-столбец единичной матрицы.

Исходя из практических требований, в работе, чтобы получить более точное решение задачи спектрального оценивания предлагается модифицированная AP-модель, которая имеет вид:

$$\tilde{\mathbf{a}} = -[\gamma\mathbf{R}_1 + (1-\gamma)\mathbf{R}_2]^{-1}\mathbf{i}, \quad (4)$$

где  $\tilde{\mathbf{a}}$  — вектор авторегрессии предлагаемой модифицированной AP-модели;  $\gamma \in [0;1]$  — весовой коэффициент определяет доли  $\gamma$  и  $(1-\gamma)$  процессов с различными статистическими свойствами [3], т.е. в данном случае с разными шумовыми компонентами  $P_n^{(1)}$ ,  $P_n^{(2)}$  соответственно.

В практике предлагаемая AP-модель применима для обработки и оценки параметров отражения электромагнитных волн от протяженных объектов (гидрометеоров, дистанционное зондирование Земли), технической и медицинской диагностики.

Как показано ниже, результаты оценки спектральной плотности мощности (СПМ) с помощью авторегрессионных параметрических методов (с относительными мощностями шума  $P_n^{(1)}=10^{-8}$  и  $P_n^{(2)}=10^{-2}$ , порядок фильтра  $p=4$ , относительная ширина  $\Delta FT=0,1$  спектральной моды, весовой коэффициент  $\gamma=0,8$ ) введены на рисунке 1. Отметим, что подобные сигналы характерны для отражения электромагнитных

волн от гидрометеорологических образований в атмосфере [4], отражения света от неоднородных сред и т. д.

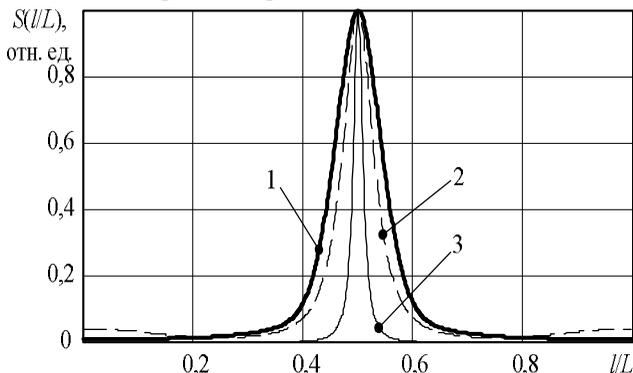


Рисунок 1 – Оценка спектральной плотности мощности

На рисунке 1 приведены условия проведения моделирования:  $S(l/L)$  — нормированная на свое максимальное значение СПМ в относительных единицах;  $l/L$  — относительная частота, число равно отношению номера  $l$  текущего спектрального отсчёта к общему числу  $L$  спектральных отсчётов ( $L=1000$ ); сплошная жирная линия 1 — энергетический контрольный спектр, полученный AP-моделей 20-го порядка по незашумлённому полезному коррелированному сигналу; пунктирная тонкая линия 2 — спектр, полученный предлагаемой моделью; сплошная тонкая линия 3 — на основе простой AR-модели.

Из анализа рисунка 1 можно видеть, что предлагаемый (пунктирная тонкая линия 2) подход дал результаты, более близкие к контрольному спектру (сплошная жирная линия 1). Качество оценивания спектральной плотности мощности у предлагаемой модели лучше, чем обычным параметрическим AP-методом.

Критерий нормированного квадрата длины  $E$  вектора  $\varepsilon$  невязки между векторами отсчетов СПМ, используемый для объективной описания эффективности предлагаемого подхода, имеет вид:

$$E = \varepsilon^T \varepsilon / L, \text{ где } \varepsilon = \mathbf{c} - \mathbf{s}, \quad (5)$$

где  $\mathbf{c}$  —  $L$ -мерный вектор СПМ контрольной модели, в качестве которой использован энергетический спектр, полученный AP-моделей большого ( $p=20$ ) порядка по незашумлённому полезному коррелированному сигналу;  $\mathbf{s}$  —  $L$ -мерный вектор СПМ, полученный соответствующими AP-методами;  $^T$  — знак транспонирования.

Проанализируем влияние порядка  $p$  на результаты сравнения адекватности моделирования при использовании различных подходов (см. таблицу 1).

Таблица 1 – Качество моделирования

| $p$ \ Модели | Обычная               | Предлагаемая          |
|--------------|-----------------------|-----------------------|
| 2            | 0,249                 | 0,078                 |
| 4            | 0,071                 | 0,018                 |
| 6            | 0,026                 | $7,461 \cdot 10^{-3}$ |
| 8            | 0,022                 | $5,502 \cdot 10^{-3}$ |
| 10           | $7,243 \cdot 10^{-3}$ | $2,828 \cdot 10^{-3}$ |

При формировании приведенной в таблице 1 зависимости величины  $E$  от порядка  $p$  моделей были приняты следующие параметры:  $L=1000$ ;  $\Delta FT = 0,1$ ;  $P_n^{(1)}=10^{-8}$  и  $P_n^{(2)}=10^{-2}$ . Проведенные исследования показывают (см. таблицу 1), что нормированные квадраты длин  $E$  векторов  $\epsilon$  невязки для оценки СПМ процесса предлагаемым подходом авторегрессии в широком диапазоне изменений порядка  $p$  имеют меньшие значения и следовательно, высокие адекватности по сравнению с известным методом. Так, например, при  $p=4$  выигрыши по критерию (5) у предлагаемого подхода к оценке спектра достигают 4,025 раз по сравнению с обычным АР-методом. При достижении той же эффективности, что и у обычной АР-модели, предлагаемый алгоритм спектрального оценивания даёт возможность снизить порядок  $p$  в 2,25 раза. Например, в случае  $E=0,018$  предлагаемая методика предполагает порядок  $p=4$  а для обеспечения той же адекватности  $E$  спектральных оценок с помощью обычной АР-модели необходим её порядок  $p=9$ .

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Колмогоров А.Н., Прохоров Ю.В., Ширяев А.Н. Вероятностно-статистические методы обнаружения спонтанно возникающих эффектов // Тр. МИАН. 1988. Т. 182. С. 4-23.
2. Марпл-мл. С.Л. Цифровой спектральный анализ и его приложения: пер. с англ. М.: Мир, 1990. 584 с.
3. Андреев В.Г., Чан Н.Л. Параметрический спектральный анализ унимодальных по спектру зашумленных сигналов // Вестник Рязанского государственного радиотехнического университета. № 3. Выпуск 57. Рязань: РГРТУ, 2016. С. 3-8.
4. Горелик А.Г., Мельничук Ю.В. О связи спектра флуктуации сигнала с движением рассеивателей в метеообъектах. ДАН, Т. 140, Вып. 3, 1961.

УДК 004.65

**Д. В. ШАРОНОВ**Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина**ОБУЧЕНИЕ НЕЙРОННОЙ СЕТИ ДЛЯ ГЕНЕРАЦИИ САЙТА**

*Рассматривается алгоритм обучения нейронной сети для генерации сайта по изображению его макета.*

Основной сложностью для автоматизации front-end разработки в данный момент является недостаток вычислительной мощности. Однако, алгоритмы глубокого обучения нейронных сетей могут позволить осуществить искусственную автоматизацию создания пользовательского интерфейса.

Процесс генерации HTML-разметки сайта можно представить в виде трёх шагов:

- 1) Передача нейронной сети изображения макета сайта;
- 2) Преобразование изображения в HTML-код;
- 3) Визуализированный вывод полученной разметки.

Для реализации данного алгоритма выбрана нейронная сеть долгой краткосрочной памяти (LTSP). В отличие от обычной рекуррентной сети, LTSP применима при большом объёме данных, т.к. не забывает о информации, полученной ранее.

Архитектура сети с долгой краткосрочной памятью представлена на рисунке 1.

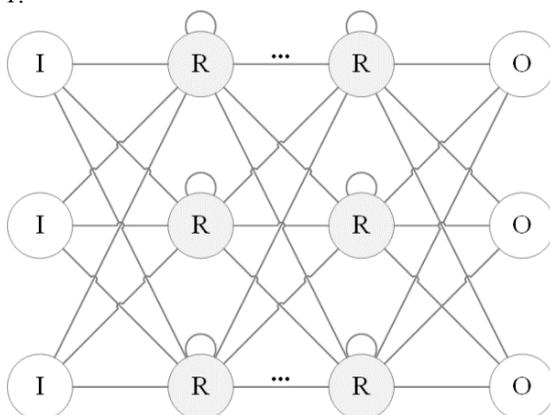


Рисунок 1 – Архитектура сети с долгой краткосрочной памятью

Здесь I – ячейка входного слоя, R – ячейка памяти, O – ячейка выходного слоя.

Для обучения нейронной сети, в неё передаются изображения макетов сайтов с соответствующими html-разметками.

Обучение заключается в последовательном предсказании HTML-тегов. Предсказывая очередной тег, нейронная сеть получает исходное изображение, а также все правильные предыдущие теги HTML-разметки. Алгоритм обучения представлен на рисунке 2.



Рисунок 2 – Алгоритм обучения нейронной сети

Пример обучающих данных представлен на рисунке 3.

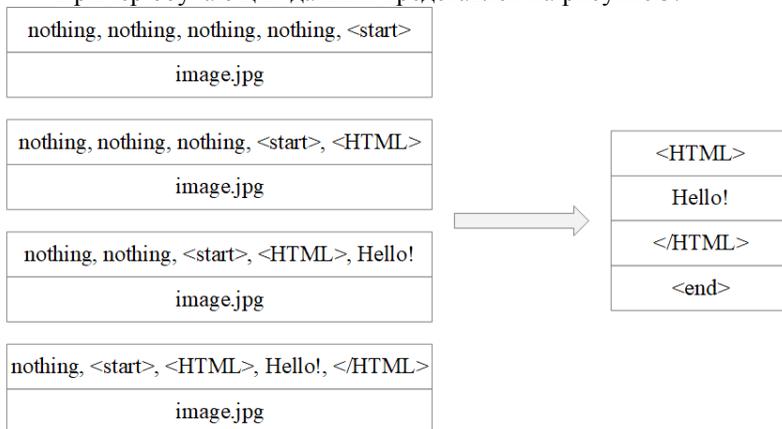


Рисунок 3 – Пример обучающих данных

Нейронная сеть преобразовывает изображения макета в списки значений пикселей от 0 до 255 в красном, зелёном и синем каналах.

Чтобы передать HTML-разметку нейронной сети, необходимо перевести её в понятный ей формат. Для этого можно использовать представление One-Hot Encoding, отображающее категориальные данные в виде двоичных векторов.

Пример представления HTML-тегов в One-Hot Encoding представлен на рисунке 4.

|         | <start> | <HTML> | HELLO! | </HTML> | <end> |
|---------|---------|--------|--------|---------|-------|
| словарь | 0       | 0      | 0      | 0       | 0     |
| <start> | 1       | 0      | 0      | 0       | 0     |
| <HTML>  | 0       | 1      | 0      | 0       | 0     |
| HELLO!  | 0       | 0      | 1      | 0       | 0     |
| </HTML> | 0       | 0      | 0      | 1       | 0     |
| <end>   | 0       | 0      | 0      | 0       | 1     |

Рисунок 4 – Пример представления HTML-тегов в One-Hot Encoding

На каждом шаге обучения нейронной сети передаются списки кодов, начиная с кода «начального» тега, а после добавляя каждый тег по одному. Выходные данные всегда состоят из одного тега.

Если длина переданного списка меньше максимальной, то он дополняется пустыми тегами (нулями).

Пример представления HTML-разметки в One-Hot Encoding представлен на рисунке 5.

|         | <start> | <HTML> | HELLO! | </HTML> | <end> |
|---------|---------|--------|--------|---------|-------|
| max     | 10000   | 01000  | 00100  | 00010   | 00001 |
| <start> | 00000   | 00000  | 00000  | 00000   | 10000 |
| <HTML>  | 00000   | 00000  | 00000  | 10000   | 01000 |
| HELLO!  | 00000   | 00000  | 10000  | 01000   | 00100 |
| </HTML> | 00000   | 10000  | 01000  | 00100   | 00010 |
| <end>   | 10000   | 01000  | 00100  | 00010   | 00001 |

Рисунок 5 – Пример представления HTML-разметки в One-Hot Encoding

Теги добавляются справа налево, что позволяет модели изучать их последовательность, вместо запоминания позиций тегов.

Пример обучающих данных в One-Hot Encoding представлен на рисунке 6.

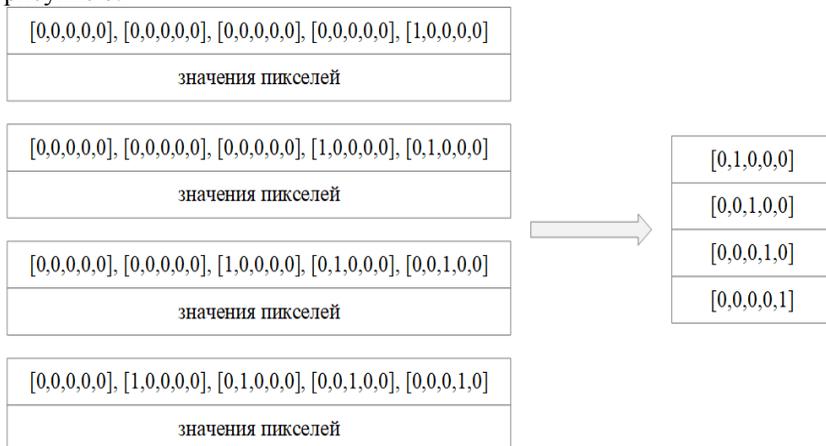


Рисунок 6 – Пример обучающих данных в One-Hot Encoding

При использовании обученной нейронной сети вместо передачи ей правильных тегов, передаётся HTML-разметка, сгенерированная ранее. Генерация начинается со «стартового» тега и заканчивается, когда нейронная сеть предскажет «конечный» тег или HTML-разметка достигнет максимальной длины.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Emil Wallner. Turning Design Mockups Into Code With Deep Learning [Электронный ресурс]. – Режим доступа: <https://blog.floydhub.com/turning-design-mockups-into-code-with-deep-learning/>.

2. Роман Котюбеев. Основные типы нейронных сетей. [Электронный ресурс]. – Режим доступа: <https://python-school.ru/blog/types-of-neural-nets/>.

УДК 004.056.5(075.8)

**О.Г. ШВЕЧКОВА, Я.А. АРТЁМОВ**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ВЫБОР ПРОГРАММНОЙ РЕАЛИЗАЦИИ КРИПТОГРАФИЧЕСКОГО АЛГОРИТМА ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСИ DSA**

*В статье рассматриваются различные способы программной реализации с учебными целями алгоритма электронной цифровой подписи DSA на основе критериев быстродействия, объема программного кода, объема памяти.*

Алгоритм DSA (Digital Signature Algorithm) широко известен в качестве алгоритма электронной цифровой подписи (ЭЦП) и до недавнего времени считался де-факто национальным алгоритмом шифрования многих зарубежных стран. DSA был представлен в августе 1991 г. и является запатентованным (автор патента — David W. Kravitz). Алгоритм DSA является частью алгоритма DSS (Digital Signature Standard — стандарт цифровой подписи), впервые опубликованного 15 декабря 1998). Последняя версия стандарта алгоритма DSA была выпущена в июле 2013 г. Относится к алгоритмам с открытым ключом, используется не для процедуры шифрования, а именно для создания ЭЦП. Математическая основа, определяющая трудность алгоритма, базируется на вычислительной сложности взятия логарифмов в конечных полях.

Общая схема работы алгоритма общеизвестна и укрупнённо может быть представлена из двух этапов: этапа формирования или постановки подписи и процесса проверки подлинности.

Сравнение реализаций алгоритма цифровой подписи DSA выполнялось на языках Java и Python как с использованием специализированных библиотек, так и без использования таковых.

В рамках проделанной работы были реализованы несколько вариантов алгоритма DSA на двух языках. Критериями сравнения были выбраны быстродействие (время исполнения программы), занимаемое место (объем памяти, используемый программой при работе) и затраты на реализацию (количество строк кода в программе).

Рассмотрим программу, реализующую алгоритм на языке Java без использования специализированных библиотек.

Программа реализует следующий алгоритм.

- Выбор криптографической хеш-функции  $H(x)$ .

- Выбор большого простого числа  $q$ , размерность которого  $N$  в битах совпадает с размерностью в битах значение хеш-функции  $H(x)$ .
- Выбор простого числа  $p$ , такого, что  $(p-1)$  делится на  $q$ . Битовая длина  $p$  обозначается  $L$  ( $2^{L-1} < p < 2^L$ ).
- Выбор числа  $g$  такого, что его мультипликативный порядок по модулю равен  $q$ . Для его вычисления можно воспользоваться формулой  $g = h^{(p-1)/q} \bmod p$ , где  $h$  — некоторое произвольное число,  $h \in (1; p-1)$  такое, что  $q \neq 1$ . В большинстве случаев значение  $h = 2$  удовлетворяет этому требованию.

При выполнении программы были получены результаты, представленные в таблице 1.

Таблица 1 – результаты выполнения программы без специализированных библиотек на языке Java

| Время выполнения, мс. | Объем памяти, байт | Объем кода, строк |
|-----------------------|--------------------|-------------------|
| 363                   | 5559               | 170               |

Для анализа и сравнения эффективности программных реализаций рассмотрена программа на языке Java, в которой использованы специализированные библиотеки для шифрования и криптографии (`java.security` и `javax.crypto`).

Программа реализует следующие этапы алгоритма.

- Выбор большого простого числа при помощи генератора случайных чисел.
- Создание публичного и приватного ключей.
- Зашифровывание сообщения.
- Верификация.
- Расшифровывание сообщения.

При выполнении программы были получены результаты, представленные в таблице 2.

Таблица 2 – результаты выполнения программы с использованием специализированных библиотек на языке Java

| Время выполнения, мс. | Объем памяти, байт | Объем кода, строк |
|-----------------------|--------------------|-------------------|
| 573                   | 17369              | 42                |

Рассмотрим программу, реализующую алгоритм цифровой подписи на языке Python без использования сторонних библиотек.

Программа реализует следующие этапы алгоритма.

- Выбор криптографической хеш-функции  $H(x)$ .

- Выбор большого простого числа  $q$ , размерность которого  $N$  в битах совпадает с размерностью в битах значение хеш-функции  $H(x)$ .
- Выбор простого числа  $p$ , такого, что  $(p-1)$  делится на  $q$ . Битовая длина  $p$  обозначается  $L$  ( $2^{L-1} < p < 2^L$ ).
- Выбор числа  $g$  такого, что его мультипликативный порядок по модулю равен  $q$ . Для его вычисления можно воспользоваться формулой  $g = h^{(p-1)/q} \bmod p$ , где  $h$  — некоторое произвольное число,  $h \in (1; p-1)$  такое, что  $q \neq 1$ . В большинстве случаев значение  $h = 2$  удовлетворяет этому требованию.

При выполнении программы были получены результаты, представленные в таблице 3.

Таблица 3 – результаты выполнения программы без специализированных библиотек на языке Python

| Время выполнения, мс. | Объем памяти, байт | Объем кода, строк |
|-----------------------|--------------------|-------------------|
| 169                   | 5545               | 140               |

В заключение рассмотрим программу, реализующую алгоритм цифровой подписи на языке Python с использованием специализированной библиотеки Crypto. При проектировании данного варианта программы следует учитывать, что указанная библиотека работает только с версией языка 3.9 и выше, хотя в версиях, предшествующих указанной, уже существует библиотека с тем же названием, но с гораздо меньшим функционалом, что не позволит реализовать алгоритм.

Программа выполняет следующие этапы алгоритма.

- Выбор большого простого числа при помощи генератора случайных чисел.
- Создание публичного и приватного ключей.
- Зашифровывание сообщения.
- Верификация.
- Расшифровывание сообщения.

При выполнении программы были получены результаты, представленные в таблице 4.

Таблица 4 – результаты выполнения программы с использованием специализированной библиотеки на языке Python

| Время выполнения, мс. | Объем памяти, байт | Объем кода, строк |
|-----------------------|--------------------|-------------------|
| 97                    | 95160              | 42                |

Обобщенный перечень результатов всех рассмотренных программных реализаций алгоритма ЭЦП DSA представлен в таблице 5.

Таблица 5 – обобщенный перечень результатов программных реализаций алгоритма ЭЦП DSA

| Способ реализации | Время выполнения, мс. | Объём памяти, байт | Объём кода, строк |
|-------------------|-----------------------|--------------------|-------------------|
| Java              | 363                   | 5559               | 170               |
| Java lib          | 573                   | 17369              | 42                |
| Python            | 169                   | 5545               | 140               |
| Python lib        | 97                    | 95160              | 42                |

Анализ полученных результатов показывает, что наибольшую скорость выполнения и наименьший объём строк кода имеет программа, реализованная на языке Python с использованием специализированной библиотеки. Заметным недостатком данной реализации служит тот факт, что программа является самой затратной по объёму памяти, превышая по этому показателю остальные более чем семикратно, поэтому абсолютно предпочтительной её назвать нельзя. Самое большое время выполнения имеет программа, реализованная на языке Java с использованием специализированных библиотек, она же занимает второе место по затратам памяти. Следовательно, эта реализация является наименее предпочтительной, несмотря на малый объём кода в программе. Сравнивая реализации на языках Java и Python без использования специализированных библиотек, можно прийти к выводу, что программа на языке Python более предпочтительно по критериям скорости работы и меньшему объёму кода в программе.

Как следует из результатов, представленных в таблицах, выбранным критериям эффективности в максимальной степени соответствуют две программные реализации на языке Python как наиболее предпочтительные по рассматриваемым характеристикам. Дальнейший выбор способа реализации остаётся за разработчиком, исходя из ресурса памяти и потребности в быстродействии.

Исходя из общих характеристик DSA освоение базовых приёмов и сравнительный анализ способов реализации данного алгоритма при изучении дисциплины «Защита информации» является эффективным способом приобретения практических навыков построения и использования алгоритмов электронной цифровой подписи.

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Васильков А.В., Васильков А.А., Васильков И.А. Информационные системы и их безопасность: учеб. Пособие. – М.: Форум, 2010. – 528 с.
2. Евдокимова Л.М., Корябкин В.В., Пылькин А.Н., Швечкова О.Г. Электронный документооборот и обеспечение безопасности стандартными средствами WINDOWS»: учеб. пособие. – М.: КУРС, 2017. – 296 с.
3. Швечкова О.Г., Марчев Д.В., Пылькин А.Н. Базовые криптографические алгоритмы защиты информации: учеб. пособие. – М.: КУРС, 2018. – 168 с.

УДК 004.716

**В.В. ЯШКОВА, М.С. ИГОШИНА**

Рязанский государственный радиотехнический университет  
имени В.Ф. Уткина

### **ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ УСТРОЙСТВА СОПРЯЖЕНИЯ СПЕЦИАЛИЗИРОВАННЫХ ЦИФРОВЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИН С ПК**

*В статье рассматривается способ реализации устройства сопряжения специализированных цифровых вычислительных машин с ПК. Дается определение устройства сопряжения, кратко описывается принцип его работы.*

#### ***Введение***

Расширение масштабов применения современных электронных приборов и систем, повышение требований к их производительности, рост степени интеграции элементной базы и связанные с этим особенности приводят к возрастанию потребности в качественных устройствах сопряжения.

В настоящее время персональный компьютер (ПК) используется не только для контроля состояния каких-либо внешних физических процессов, логической обработки полученной информации, а также в качестве основы для создания автоматизированных систем (АС). АС позволяют выполнять различные операции на высоком интеллектуальном уровне, проводить диагностику радиоэлектронной аппаратуры (РЭА). Для корректной реализации данных задач ПК, необходимо наличие внешнего электронного устройства для коммутации компьютера с объектом.

Так как в настоящее время существует огромное количество аппаратуры широкого спектра назначения, множество компаний, занимающихся её производством, а также разных стандартов, большей частью никак не совместимых между собой, актуальной проблемой в области систем является проблема совместимости различных видов устройств. Выход для этой ситуации – преобразователи интерфейсов.

В данной статье рассматривается устройство сопряжения специализированных цифровых вычислительных машин с ПК, позволяющее решать подобные вопросы коммутации.

### ***Специализированные вычислительные машины***

Специализированные вычислительные машины направлены на решение узких задач. Такие машины способны повысить эффективность средств вычислительной техники за счет того, что структурная и аппаратная интерпретация программ увеличивает точность и быстродействие устройств, упрощает математическое обеспечение, снижает аппаратные затраты.

Отличительной чертой специализированных компьютеров является постоянство структуры, определяемой классом решаемых задач, что позволяет существенно упростить коммутационные устройства.

Существует следующая классификация специализированных вычислительных машин:

*По способу представления информации* различают аналоговые, цифровые и гибридные машины.

*По назначению* можно выделить: управляющие машины, работающие в режиме реального времени, использующиеся для управления динамическими объектами (например, летательными аппаратами), а также моделирующие машины, применяющиеся для решения задач при помощи математических моделей реальных объектов.

### ***Устройства сопряжения***

В качестве устройства сопряжения обычно выступают адаптеры, переходники, выраженные интерфейсом USB, главная задача которых – обеспечение возможности передачи информации от внешних источников на ПК [1].

Особенность проектирования любого устройства сопряжения заключается в том, что оно подключается к уже готовой системе - персональному компьютеру. То есть при разработке внешнего устройства надо учитывать возможности ПК, его протоколы обмена, скоростные характеристики, особенности внешних интерфейсов компьютера.

Преобразователи интерфейса представляют собой эффективное средство расширения функциональности информационной системы, незаменимое в условиях отсутствия единого стандарта построения

вычислительных систем и систем передачи данных. Они применяются для обеспечения совместимости устройств с разными интерфейсами или изменения физического способа передачи информации.

Разные стандарты устройств предусматривают передачу данных по различным технологиям. Унифицировать протоколы и привести передаваемые данные к единому виду с помощью преобразователя интерфейсов невозможно.

Задача преобразователя – адаптировать вид данных, передаваемых между частями системы с различными протоколами для того, чтобы они были успешно приняты и расшифрованы элементом, использующим другой протокол.

Преобразование пакетов передаваемых данных происходит на программном уровне. Кроме изменения структуры передаваемых данных, программная составляющая отвечает за определение типов протоколов, используемых в системе, и выбор алгоритма для их согласования [2].

Преобразователи классифицируют по следующим параметрам:

- Стандарт – определяет тип устройств и протоколов, для которых возможно выполнение преобразований;
- Скорость передачи информации – характеризует максимальное количество информации, передаваемой в системе за единицу времени;
- Допустимое расстояние передачи – максимальная удаленность узлов системы, между которыми может быть выполнена передача данных без потерь их целостности;
- Тип передаваемого сигнала – определяет, какой параметр системы будет нести информационную нагрузку;
- Линия передачи – вид среды, через которую будут передаваться пакеты информации;
- Количество приемников и драйверов;
- Схема соединения составных частей системы.

Функциональная схема устройства сопряжения представлена на рисунке 1.

Устройство предназначено для соединения персонального компьютера по интерфейсу USB с вычислительными машинами, имеющими интерфейс CAN. Также на плате предусмотрена возможность подключения по интерфейсу JTAG.

В основе устройства сопряжения лежит 8-разрядный микроконтроллер, поддерживающий интерфейс CAN, но не имеющий интерфейс USB, поэтому на плате применяется микросхема преобразователя USB в параллельный интерфейс, подключенная к нему.

Контроллер является программируемым, т.е. он способен выполнять различные функции, в том числе управление, просмотр связи с компьютером, реализовывает большое количество задач, может работать без ПК, поскольку имеет память для нескольких операций.

В свою очередь, интерфейс CAN микроконтроллера проходит через трансивер, обеспечивающий корректные параметры сигналов CAN. Линии интерфейса JTAG проходят через аналогичный драйвер, позволяющий работать контроллеру с напряжением +3.3В и +5В.

К контроллеру подключена микросхема флэш-памяти, 3 кнопки и внешний дисплей. Эти компоненты предназначены для обеспечения возможности автономной работы адаптера, когда с ПК через USB берётся только питание. Например, может быть реализовано отображаемое на дисплее меню для выбора кнопками файла прошивки ПЛИС из флэш-памяти и его последующая загрузка на целевую машину без участия ПК.

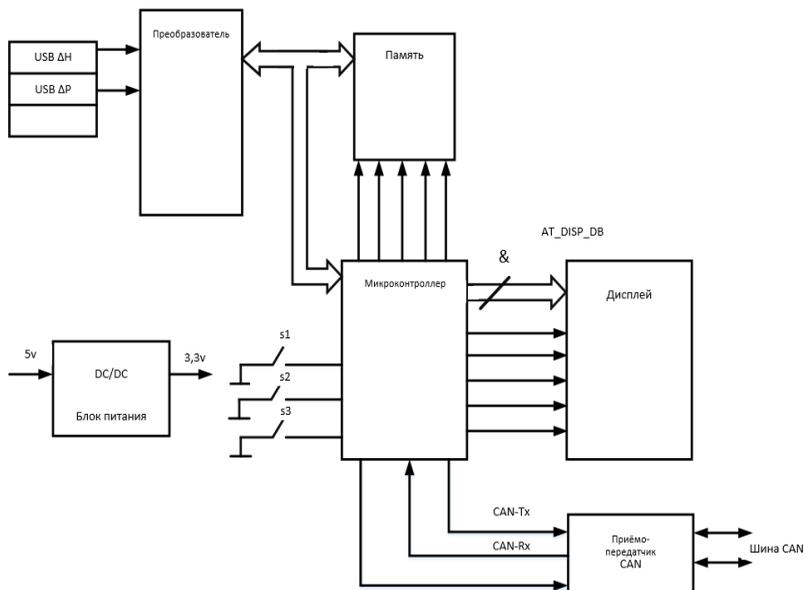


Рисунок 1 – Устройство сопряжения

### СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Ан П. Сопряжение ПК с внешними устройствами: учебное пособие / П. Ан. – Москва: ДМК Пресс, 2008. – 320 с.
2. Жмудь В. А. Устройства сопряжения с объектом: учебное пособие / В. А. Жмудь. – Новосибирск: НГТУ, 2019. – 172 с.

# Информационные технологии

Межвузовский сборник научных трудов

Компьютерная верстка: А.Н. Сапрыкин,  
П.О. Алексанова, Е.В. Косорукова

Подписано в печать 11.05.22. Формат бумаги 60x84 1/16.  
Бумага офсетная. Печать струйная. Усл. печ. л. 24,25.  
Тираж 100 экз. Заказ № 5191.

Издательство ИП Коняхин А.В. (Book Jet)

Отпечатано в типографии Book Jet  
390005, г. Рязань, ул. Пушкина, д. 18

Сайт: <http://bookjet.ru>  
Почта: [info@bookjet.ru](mailto:info@bookjet.ru)  
тел.: +7 (4912) 466-151

ISBN 978-5-907568-17-4

