

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«Пермский национальный исследовательский
политехнический университет»

**АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ
УПРАВЛЕНИЯ И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ**

Материалы всероссийской научно-технической
конференции
(г. Пермь, 7–9 июня 2024 г.)

В двух томах
Том I

Издательство
Пермского национального исследовательского
политехнического университета
2024

УДК 685.325.05

A224

A224 **Автоматизированные** системы управления и информационные технологии : материалы всерос. науч.-техн. конф. : в 2 т. / ФГАОУ ВО «Пермский национальный исследовательский политехнический университет». – Пермь : Изд-во Перм. нац. исслед. политехн. ун-та, 2024. – Т. I. – 629 с.

Проведение конференции инициировано ученым советом электротехнического факультета Пермского национального исследовательского политехнического университета и ориентировано на публичную апробацию результатов научно-исследовательских работ молодых ученых, аспирантов и студентов по аспектам научных исследований электротехнического профиля.

Конференция проводится по пяти секциям: «Информационные технологии и автоматизированные системы», «Автоматизация технологических процессов и производств», «Электротехника и энергетика», «Телекоммуникации», «Информационная безопасность».

Публикуемые результаты исследований могут быть интересны широкому кругу специалистов в области автоматизации и проектирования современных систем автоматизации и управления, информационных технологий, математического моделирования технологических процессов, систем преобразования и обработки информации.

Редакционная коллегия:

В.В. Черняев, доцент, канд. техн. наук;

С.В. Мишуринских, доцент, канд. техн. наук (отв. за выпуск);

В.В. Кавалеров, доцент, д-р техн. наук;

А.Б. Петроченков, доцент, д-р техн. наук;

Н.М. Труфанова, профессор, д-р техн. наук;

Р.А. Файзрахманов, профессор, д-р экон. наук;

А.А. Южаков, профессор, д-р техн. наук

Рецензент – заслуженный деятель науки РФ, заслуженный машиностроитель Республики Башкортостан, д-р техн. наук, профессор кафедры автоматизированных систем управления Уфимского государственного авиационного технического университета *Г.Г. Куликов*

УДК 685.325.05

ISBN 978-5-398-03242-0 (Т. I)

ISBN 978-5-398-03241-3 (общ.)

© ПНИПУ, 2024

Секция I

**ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
И АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ**

Н.А. Рябов, А.А. Петренко

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ГРАФИЧЕСКОГО РЕДАКТОРА КОСМИЧЕСКИХ ТЕЛ ДЛЯ ДВУМЕРНЫХ ИГР

В статье рассматривается разработка программы создания моделей планет для компьютерных игр. Центральной частью проекта является метод процедурной генерации, позволяющий пользователю гибко настроить процесс и быстро получить результат. Генерация планеты, помимо создания рельефа поверхности, включает в себя моделирование климата и атмосферы, распределение растительности по поверхности. В статье на концептуальном уровне рассматривается процесс генерации планеты, его основные этапы. Более подробно показано моделирование климатических характеристик и определение биомов.

Ключевые слова: процедурная генерация, планета, моделирование климата, графический редактор.

N.A. Ryabov, A.A. Petrenko

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A GRAPHIC EDITOR OF COSMIC BODIES FOR TWO-DIMENSIONAL GAMES

The article discusses the development of a program for creating models of planets for computer games. The central part of the project is the method of procedural generation, which allows the user to flexibly configure the process and quickly get the result. Planet generation, in addition to creating a surface relief, includes modeling the climate and atmosphere, and the distribution of vegetation over the surface. The article considers the process of planet generation and its main stages at the conceptual level. The modeling of climatic characteristics and the definition of biomes are shown in more detail.

Keywords: procedural generation, planet, climate modeling, graphic editor.

В настоящее время в мире получили очень широкое распространение компьютерные игры, в том числе проекты с космической тематикой. Из 53 самых продаваемых игр в 2023 г. семь так или иначе использовали тему космоса [1]. В большинстве своем такие игры используют процедурную генерацию контента в целом и планет в частности. Однако доступный инструмент генерации планет на рынке отсутствует, и каждой команде разработки приходится создавать свое решение. В результате было решено создать программу с процедурной генерации

планет. Также было решено добавить к проекту графический интерфейс, который позволит не только быстро подбирать нужные настройки генерации, но и распространять программу как самостоятельный продукт. Более того, для качественной генерации необходимо моделировать планету как систему, учитывая знания из географии и климатологии. Это открывает возможность использования программы в образовательных и научных целях, учитывая то, что данных о настоящих планетах вне солнечной системы крайне мало [2].

Программу было решено реализовывать с помощью следующих технологий:

- язык программирования C++;
- фреймворк графического интерфейса Qt;
- формат хранения данных JSON.

Центральной частью рассматриваемой программы является методика процедурной генерации планеты. Этот процесс показан на рис. 1 в виде диаграммы в нотации IDEF0.

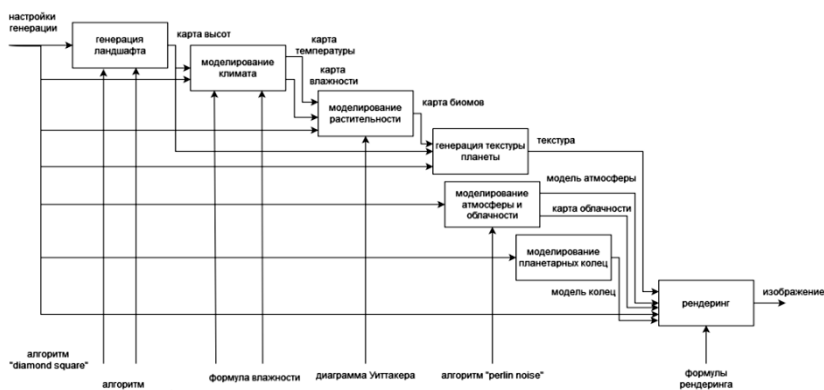


Рис. 1. Процесс генерации планеты

Как видно из диаграммы на рис. 1, процесс начинается с генерации рельефа поверхности планеты. Пользователю доступны два алгоритма на выбор: diamondsquare [3] и faultformation [4]. Затем, получившаяся карта высот отправляется на следующий этап – моделирование климата. Здесь на основе настроек пользователя и данных о рельефе планеты вычисляются средние значения влажности и температуры. Полученные климатические карты используются для моделирования распределения растительности на следующем этапе. После этого программа генерирует

ет текстуру планеты, по желанию пользователя применяя шум и градиентное сглаживание. Текстура вместе с данными об облачности, атмосфере и кольцах попадает на этап рендеринга. Здесь происходит наложение текстуры на сферу, наложение облачности, атмосферы, теней колец. В результате получается изображение планет, которое будет предоставлено пользователю.

Далее будет подробнее рассмотрен этап моделирования климата и биомов. На рис. 2 приведена иллюстрация данных, генерируемых программой в процессе моделирования.

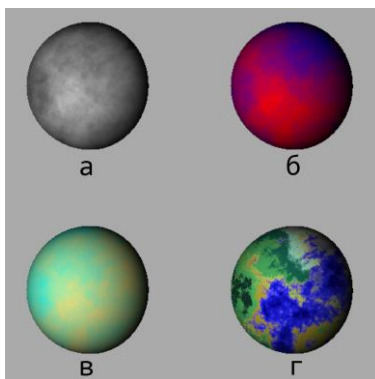


Рис. 2. Иллюстрация климатических данных

Рис. 2, *а* представляет собой иллюстрацию данных о рельефе планеты. Более светлые участки соответствуют малой абсолютной высоте.

Как уже говорилось выше, на основе данных о рельефе строится карта температуры, изображенная на рис. 2, *б*. Для расчета температуры в точке с координатами ij применяется формула (1). Как видно, температура линейно зависит от высоты в данной точке h , уровня воды на планете w , угла падения лучей α и пользовательского значения t .

$$T_{ij} = (56 \sin(\alpha) - 28) - 0,006(h_{ij} - w) + (t - 15). \quad (1)$$

Далее следует расчет карты влажности, представленной на рис. 2, *в*. Для получения данных о влажности используется формула (2). Здесь среднегодовое значение влажности в точке с координатами ij линейно зависит от высоты h , среднегодовой температуры T и уровня моря w .

$$W_{ij} = 1,37(h_{ij} - w) + 0,32T_{ij} + 51,53. \quad (2)$$

Используя полученную информацию, можно смоделировать расположение различных биомов на поверхности планеты. Карта биомов показана на рис. 2, г. Взаимосвязь между температурой, влажностью и типом биома определяется по диаграмме Уиттакера [5]. Она представлена на рис. 3.

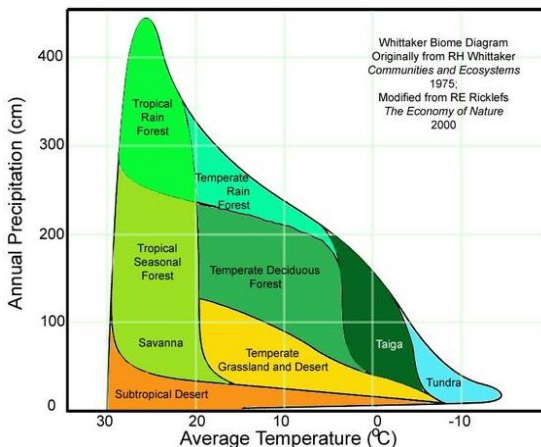


Рис. 3. Диаграмма Уиттакера

Согласно диаграмме на рис. 3, при малых значениях температуры и влажности образуется тайга. Эту же закономерность можно увидеть на рис. 2. Темно-зеленая таежная зона соответствует приполярной местности с низкой температурой и влажностью.

Таким образом, была разработана программа процедурной генерации планет. Она позволит сократить затраты на разработку компьютерных игр. Кроме того, решение может помочь при изучении географии или компьютерной графики. Перспективой развития является углубление процесса моделирования, например, тектоника или конвекционные процессы в атмосфере. Также можно рассмотреть интеграцию с системами разработки игр.

Библиографический список

1. Steam [Электронный ресурс]. – URL: <https://store.steampowered.com/charts/bestofyear/BestOf2023?tab=1> (дата обращения: 12 апреля 2024).
2. Штерн Б., Рубаков В. Астрофизика. Троицкий вариант. – М.: Изд-во АСТ, 2020. – 368 с. (Удивительная наука).

3. Fournier A., Don Fussell, Loren Carpenter. Computer Rendering of Stochastic Models // Communications of the ACM. – June 1982. – Vol. 25, iss. 6. – P. 371–384.
4. Aluning R., Christensen E., Joseph Anthony C. Hermocilla // Terra: A 3 D TerrainGeneratorandVisualizer. – 2010.
5. Global field observations of tree die-off reveal hotter-drought fingerprint for Earth’s forests / W.M. Hammond, A.P. Williams, J.T. Abatzoglou [et al.] // Nat Commun. – 2022. – № 13. – P. 1761.

Сведения об авторах

Рябов Никита Андреевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: riabovnick080@yandex.ru

Петренко Александр Анатольевич – кандидат технических наук, доцент, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: hawk321@mail.ru

В.А. Карелов, Д.Б. Кузнецов

Пермский национальный исследовательский
политехнический университет, Пермь

ПРОСТОЙ АЛГОРИТМ РАСПРЕДЕЛЕНИЯ ГРАФИКОВ ПО НЕСКОЛЬКИМ ОСЯМ

Рассматривается простой с точки зрения реализации алгоритм распределения графиков по нескольким осям. Также рассматриваются альтернативные варианты, их преимущества и недостатки. После анализа приведен пример алгоритма на языке программирования C#. Оценена сложность получившейся программы, а также приведены действия для улучшения производительности.

Ключевые слова: алгоритм, графики, ось, диаграмма, рисунки, оптимизация.

V.A. Karelov, D.B. Kuznetsov

Perm National Research Polytechnic University, Perm

A SIMPLE ALGORITHM FOR DISTRIBUTION OF CHARTS ALONG SEVERAL AXES

Discusses a simple, from an implementation point of view, algorithm for distributing graphs along several axes. Alternative options, their advantages and disadvantages are also discussed. After the analysis, an example of an algorithm in the C# programming language is given. The complexity of the resulting program is assessed, and steps to improve performance are given.

Keywords: algorithm, graphics, axis, diagram, drawings, optimization.

При анализе данных может возникать необходимость в сравнении значений нескольких графиков, расположенных на одном участке по оси абсцисс. В случае если рассматриваемые данные находятся достаточно близко по оси ординат, не возникает проблем с их визуальным восприятием. Но при различии в значениях на порядок точки графиков выстраиваются в прямую линию. Для примера построим синусоиду и косинусоиду, расположив их на одном графике, который продемонстрирован на рис. 1. Рассмотрим решения поставленной проблемы в рамках реализации приложения для построения большого количества графиков. Это означает, что в дополнении к проблеме восприятия (основная цель), нужно также учесть аспекты, которые могут повлиять на приложение (например, производительность).

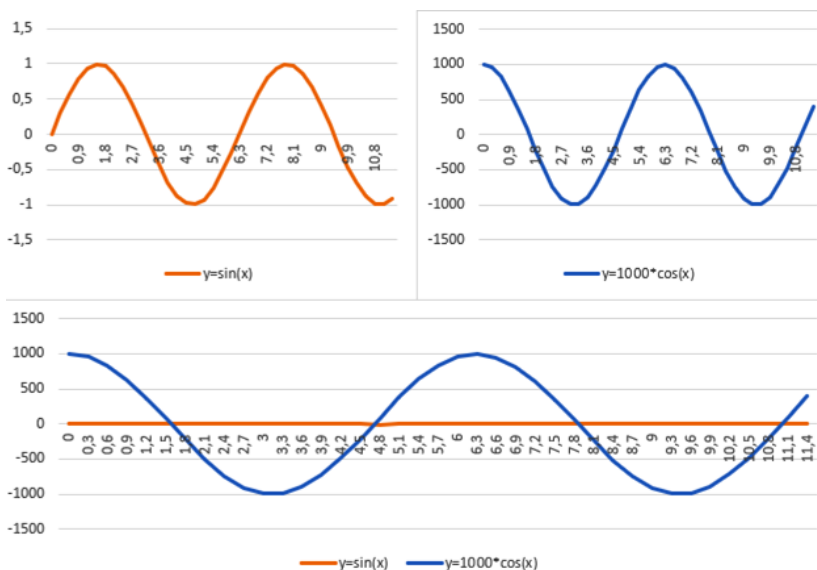


Рис. 1. Демонстрация проблемы

Решением может стать расположение графиков на отдельных рисунках (или диаграммах). Данный подход упрощает разработку приложения за счет простого создания картинки для каждого графика, отсутствует необходимость в создании дополнительных механизмов. Однако при большом количестве диаграмм на экране они начнут занимать много места, так как для каждого рисунка потребуются дополнительно нарисовать оси и подписи к ним. Можно располагать графики горизонтально. Этот способ уменьшит занимаемое место по вертикали, но ухудшит восприятие данных. При таком расположении труднее оценить значения ординат разных графиков на разных диаграммах при одном значении абсцисс. Альтернативой может послужить разбиение графиков на несколько групп, для каждой из которых будет собственная ось ординат. Данный вариант обладает следующими достоинствами:

1. Диаграммы позволяют объединить несколько графиков, что приведет к сокращению размера рабочей области¹ по высоте.

¹ Рабочая область – совокупность всех диаграмм, кнопок и прочих элементов графического интерфейса приложения.

2. У всех графиков будет единая ось абсцисс, что упростит сравнение значений точек, например, в конкретный момент времени, если ось абсцисс – ось даты и времени.

3. Также в приложении можно оставить возможность создания нескольких диаграмм, что позволит пользователям настроить рабочее пространство под себя.

Сформируем окончательную задачу – разработать алгоритм кластеризации графиков для их расположения на разных осях «у» для улучшения восприятия информации.

Входными данными будут являться список/массив/коллекция графиков. Каждый график будет иметь набор двумерных точек.

Ограничения: разработанный алгоритм применяется для двумерного пространства (x и y), константное значение для логарифма больше нуля и не равно единице (ограничение основания).

Для решения задачи предлагается следующий алгоритм:

1. Найти максимальное абсолютное значение среди всех значений «у» каждого графика.

2. Вычислить логарифм по основанию 10 (можно взять любое другое константное значение больше нуля и не равное единице) от найденного максимального значения для каждого графика. Полученное значение округлить вниз.

3. Сгруппировать графики по найденной степени выбранной константы.

Реализуем данный алгоритм на языке программирования C#. Начнем с объявления вспомогательного класса Trend (листинг 1).

Листинг 1 – Вспомогательный класс

```
class Trend
{
    public List<(double x, double y)> Values =
new();
}
```

Далее объявим метод, получающий в качестве параметра список графиков (листинг 2). Будет возвращаться словарь, в котором ключом будет степень 10, значение – список графиков для полученного значения.

Листинг 2 – Основной метод

```
Dictionary<int, List<Trend>>> TrendGroup-
ing(List<Trend> trends)
{
    trends = trends.Distinct().ToList(); // удаляем
```

```

повторяющиеся значения
Dictionary<int, List<Trend>> groupedTrends =
new();
foreach (var trend in trends) // перебор графиков
{
    // поиск максимального абсолютного значения
    по оси y
    var maxValue = trend.Values.Max(elem =>
Math.Abs(elem.y));
    var grade = (int)Math.Log10(maxValue); //
находим логарифм по основанию 10
    if (!groupedTrends.ContainsKey(grade)) //
оси с такой степенью еще нет в словаре
        groupedTrends.Add(grade, new
List<Trend>());
    groupedTrends[grade].Add(trend); // добав-
ляем график
}
return groupedTrends;
}

```

Для показанных на рис. 1 данных результат работы алгоритма будет примерно таким, как показано на рис. 2. Внешний вид диаграмм будет зависеть от конкретной реализации в программном обеспечении.

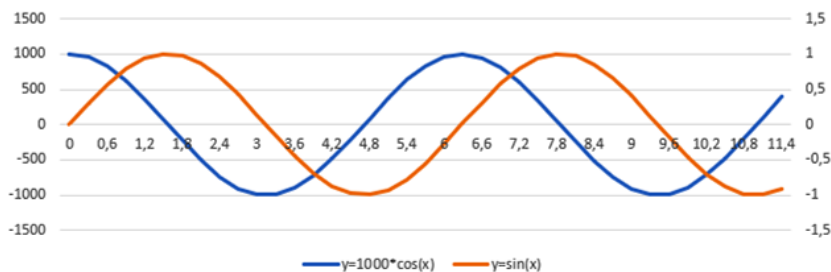


Рис. 2. Примерный результат работы алгоритма

Данный алгоритм имеет сложность $O(n^2)$ – из-за цикла с перебором графиков и поиска максимального значения внутри него. Такую сложность также дает метод `Distinct`.

Данный алгоритм можно оптимизировать:

1. Отказаться от методов расширения LINQ (методы Max и Distinct) и заменить их на обычный цикл for или foreach (при использовании .NET 7.0 и новее прирост в производительности будет незначительный). На сложность $O(n^2)$ это не повлияет.

2. Если известно, что коллекция точек отсортирована по значению «у», то можно использовать первый или последний элемент коллекции вместо поиска максимального значения. Однако чаще всего значения сортируются по «х», поэтому применимость данного пункта зависит от реализации остального приложения.

3. Если известно, что элементы в коллекции графиков не повторяются, можно отказаться от удаления повторяющихся элементов (метод Distinct).

Выполнение пунктов 2 и 3 приведет к уменьшению сложности программы до $O(n)$. Необходимость и возможность выполнения действий по оптимизации следует рассматривать для каждой конкретной реализации. Часть из них приведет к ухудшению читаемости кода, другая зависит от архитектуры всего приложения.

Сведения об авторах

Карелов Вадим Андреевич – студент Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: vadimkarelov02@gmail.com

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@mail.ru

Н.А. Павлова, А.Б. Федоров

Пермский национальный исследовательский
политехнический университет, Пермь

МЕТОДЫ ОБРАБОТКИ ПОКАЗАНИЙ ДАТЧИКОВ: СРАВНЕНИЕ И АНАЛИЗ ФИЛЬТРОВ

Рассматриваются методы обработки показаний датчиков и проводится сравнение и анализ различных фильтров. Описывается, какие виды помех могут возникать при измерении пульсовой волны и каким образом математические методы помогают фильтровать эти помехи. Приводятся различные методы фильтрации, такие как фильтры среднего арифметического, бегущего среднего, экспоненциального бегущего среднего, медианный фильтр и фильтр Калмана. Для каждого метода описываются преимущества, недостатки и особенности его применения. В итоге делается вывод о том, что наиболее эффективным способом обработки данных является комбинация медианного фильтра и фильтра бегущего среднего арифметического.

Ключевые слова: датчики, пульс, помехи, фильтрация, математические методы, полиномиальные фильтры, фильтры экспоненциального сглаживания, фильтры медианы, момент третьего порядка, фильтр среднее арифметическое, фильтр растянутая выборка, фильтр бегущее среднее арифметическое, фильтр экспоненциальное бегущее среднее, фильтр адаптивный коэффициент, медианный фильтр, фильтр Калмана, сравнение фильтров.

N.A. Pavlova, A.B. Fedorov

Perm National Research Polytechnic University, Perm

METHODS FOR PROCESSING SENSOR INDICATIONS: COMPARISON AND ANALYSIS OF FILTERS

Discusses methods for processing sensor readings and compares and analyzes various filters. Describes what types of interference can occur when measuring a pulse wave and how mathematical methods help filter this interference. Various filtering methods are provided, such as arithmetic mean, running mean, exponential running mean, median and Kalman filters. For each method, the advantages, disadvantages and features of its application are described. As a result, it is concluded that the most effective way to process data is a combination of a median filter and a running arithmetic mean filter.

Keywords: sensors, pulse, noise, filtering, mathematical methods, polynomial filters, exponential smoothing filters, median filters, third-order moment, arithmetic mean filter, stretched sample filter, running arithmetic mean filter, exponential running mean filter, adaptive coefficient filter, median filter, Kalman filter, comparison of filters.

Для фильтрации помех при измерении пульсовой волны используются математические методы, такие как фильтрация, сглаживание и удаление тренда. Вторичные методы позволяют анализировать форму пульсовой волны и выявлять патологии. Сигналы могут содержать как детерминированные, так и стохастические компоненты, которые подвержены помехам, таким как флуктуации и импульсы. Для фильтрации сигналов применяют полиномиальные фильтры, фильтры экспоненциального сглаживания и фильтры медианы, однако важно помнить, что они могут искажать сигналы [1]. Для оценки искажений можно использовать момент третьего порядка.

Методы фильтрации. Фильтр среднее арифметическое. Для реализации фильтра среднего арифметического есть два способа:

1. Последовательно получать новые сигналы, складывать их и затем делить на количество измерений для получения среднего результата. Этот способ подходит, если измерение сигнала занимает небольшое количество времени. В противном случае это может привести к изменению сигнала, что повлияет на скорость программы и точность анализа.

2. Получать новый сигнал с определенной частотой, хранить последние M измерений в массиве и при каждом новом измерении усреднять весь массив. Однако этот метод имеет недостаток в виде накопления ошибки при увеличении ширины окна усреднения.

Метод однократной выборки. Среднее арифметическое – это результат деления суммы значений на их общее количество. Для использования этого алгоритма необходимо просуммировать все значения в выборке и разделить эту сумму на количество измерений. Фильтр среднего арифметического улучшает качество фильтрации при увеличении размера выборки и времени опроса датчика, а также эффективно сглаживает шум, но требует выполнения нескольких измерений одновременно, что может увеличить время работы.

Фильтр растянутая выборка. Фильтр суммирует несколько измерений перед выдачей результата, средний уровень подавления шума, возможность настройки по объему выборки и частоте опроса. Процесс измерения пошаговый. Рекомендуется использовать при изменении сигнала с низкой скоростью.

Фильтр бегущее среднее арифметическое. Алгоритм фильтра скользящего среднего основан на создании статического массива для

усреднения последних измерений за счет добавления новых значений и удаления старых [2]. Этот метод хорошо фильтрует шум, но создает задержку. Увеличение объема выборки улучшает усреднение, но увеличивает задержку. Важно контролировать частоту опроса для избежания задержек. Преимущества включают настраиваемость фильтра и одиночное измерение за раз, недостатки – задержка среднего значения.

Фильтр адаптивный коэффициент. Для работы экспоненциального скользящего среднего с импульсными помехами используется аддитивный коэффициент фильтрации, что позволяет ему лучше адаптироваться к резким изменениям значений. Коэффициент увеличивается при значительных отличиях отфильтрованного значения от реального, что позволяет быстрее уменьшить разрыв. При близких значениях коэффициент становится меньше, чтобы эффективно фильтровать шум.

Медианный фильтр. Медианный фильтр – это цифровой фильтр для уменьшения уровня шума в сигналах, который вычисляет медиану из набора данных, игнорируя выбросы [3]. Он успешно удаляет отдельные выбросы, но может отставать от резких изменений сигнала из-за своей природы. Добавление фильтра бегущего среднего к медианному фильтру обеспечивает жесткую фильтрацию выбросов и постоянного шума, повышая точность. Преимущества: устраняет импульсные помехи, измерение пошагово. Недостатки: требует значительных ресурсов для больших выборок, имеет задержку в половину размера фильтра.

Фильтр Калмана. Фильтр Калмана – эффективный алгоритм оценки вектора состояния на основе зашумленных измерений [4]. Он учитывает диапазон измерений и скорость изменений значений. Фильтр Калмана обеспечивает более эффективное фильтрование по сравнению с обычным бегущим средним, игнорирует резкие выбросы и способен фильтровать как шум, так и помехи. Однако он требует большого количества операций с плавающей точкой, что делает его ресурсоемким.

Сравнение фильтров. Таблица содержит сравнение параметров исследованных фильтров. Для сравнения были установлены следующие условия: время фильтрации 100 000 значений при максимальной скорости передачи данных 115 200 бит/с.

Сравнение фильтров

Наименование	Тип помех	Запаздывание	Особенности	Время
Однократная выборка	Сглаживает постоянный шум и шумы импульсных помех	Большие затраты времени	Увеличение выборки и времени опроса улучшает качество фильтрации	1 мин 22 с
Растянутая выборка	Сглаживает постоянный шум и шумы импульсных помех	Выполняется одно измерение за раз	Лучше всего использовать в случаях, когда сигнал изменяется медленно	1 мин 08 с
Бегущее среднее арифметическое	Отсеивает непрерывный шум	Значение запаздывает, так как фильтр усредняет измерения	Требуется точная настройка объема выборки и частоты опроса	1 мин 25 с
Экспоненциальное бегущее среднее	Отсеивает непрерывный шум	Производится одно измерение за раз	Фильтр настраивается коэффициентом фильтрации, чем он меньше, тем плавнее фильтр	1 мин 08 с
Экспоненциальное бегущее среднее с адаптивным коэффициентом	Отсеивает непрерывный шум	Производится одно измерение за раз	Коэффициент фильтрации подстраивается под резкие изменения значений	1 мин 12 с
Медианный фильтр	Отсеивает непрерывный шум	Запаздывание происходит на половину размерности фильтра	Алгоритм для объема выборки больше 3 является громоздким	1 мин 08 с
Фильтр Калмана	Сглаживает постоянный шум и шумы импульсных помех	Запаздывает как бегущее среднее	Требуются дополнительные расчеты для настройки разброса измерений	1 мин 08 с

Анализируя данные, представленные в таблице, можно сделать вывод, что для повышения точности данных рекомендуется использовать медианный фильтр в сочетании с фильтром бегущего среднего арифметического. Это обеспечит эффективное сглаживание постоянного шума вместе с отфильтрованными импульсными помехами.

Библиографический список

1. Астахова М.К., Веселов А.С., Дьяконов А.Г. Методы обработки данных пульсовой волны // Вестник Пермского национального исследовательского политехнического университета. Технические науки. – 2017. – № 3. – С. 95–101.
2. Кривцова О.В., Смирнов А.П. Фильтры среднего арифметического: применение и эффективность // Научный журнал «Математика и компьютерная наука». – 2019. – № 4. – С. 23–31.
3. Громов Д.П., Николаев С.И. Медианные фильтры: особенности применения и анализ // Инженерный журнал «Современные технологии». – 2018. – № 1. – С. 45–53.
4. Шмель Е.В., Павлов В.М. Применение фильтра Калмана в сигнальной обработке // Международный научный журнал «Информационные технологии». – 2015. – № 2. – С. 64–70.

Сведения об авторах

Павлова Наталья Анатольевна – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ4-22-1м, г. Пермь, e-mail: baru5hnikova123@yandex.ru

Федоров Андрей Борисович – старший преподаватель кафедры кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pstufab@gmail.com

З.Г. Касимова, В.О. Кушев

Пермский государственный национальный
исследовательский университет, Пермь

РАЗРАБОТКА СЕРВИСОВ ДЛЯ ПОВЫШЕНИЯ ОТКРЫТОСТИ СРЕДЫ В ОРГАНИЗАЦИИ

Сфера цифровых технологий в управлении персоналом в настоящее время находится на стадии быстрого развития. Автором статьи рассматривается понятие открытой цифровой среды в организации как неотъемлемой составляющей современных подходов в менеджменте. Предлагается классификация, указываются общие характеристики изученных цифровых сервисов в области управления персоналом (HR). Использование данных сервисов позволяет организациям укреплять социальное взаимодействие между сотрудниками, вовлекать персонал в обмен идеями, обратной связью и признание достижений. Все это в условиях текущих демографических вызовов дает организациям дополнительные инструменты привлечения, вовлечения и удержания персонала.

Ключевые слова: открытая среда, цифровые сервисы, управление персоналом.

Z.G. Kasimova, V.O. Kushev

Perm State National Research University, Perm

DEVELOPING OPEN ENVIRONMENT DIGITAL SERVICES IN AN ORGANIZATION

The use of digital technologies in human resources management is now rapidly developing. In this article we study the notion of open digital environment in an organization as an integral part of modern management approaches. Classification is given and common features of reviewed digital services in HR management are listed. The use of such services allows organizations to enforce social networking among employees, engage their staff in idea, feedback and recognition exchange. All this provides additional opportunities for organizations to recruit, engage and retain employees.

Keywords: open environment, digital services, human resources management.

Интеграция процессов управления персоналом в цифровую среду происходит сегодня не только с точки зрения обязательных для существования бизнеса процессов (например, начисление заработной платы, выплата налогов, т.п.). Последнее можно считать уже пройденным этапом. В наши дни в цифровую среду все больше интегрируются процессы социального взаимодействия сотрудников внутри организации.

Формируется новая реальность для организации бизнес-процессов организаций, включая процессы управления персоналом [1].

Цифровой HR – это глобальный тренд в области управления трудовыми ресурсами последних лет [2]. Он используется для совершенствования связей в организации и усиления чувства принадлежности сотрудников своему работодателю [3].

Открытая с точки зрения коммуникаций среда, в том числе цифровая, поддерживающая доверие, доступность информации, возможность открытого обмена мнениями, высокой скорости информационных потоков, создает одно из ключевых преимуществ HR-бренда работодателя и бизнеса в целом. Под открытостью среды мы понимаем такую среду (атмосферу) в организации, когда сотрудники имеют доступ и применяют инструменты (сервисы) для обмена информацией и предложениями, признания достижений друг друга, обмена рациональными предложениями, предоставления обратной связи, причем делают это без принуждения со стороны руководства, открыто и свободно.

Внедрение цифровых решений (сервисов), направленных на создание открытого взаимодействия сотрудников в формате p2p (на равных) с учетом корпоративной культуры и ценностей организации, является важным условием привлечения, вовлечения и удержания персонала.

Автором были изучены различные существующие цифровые решения в области управления персоналом (LOQUI Business, «Пряники», «Эквио», Websoft, Motivity, HRBOX).

Общими для перечисленных выше платформ оказались такие характеристики, как мультифункциональность, модульность, единое информационное пространство, наличие возможностей прямого взаимодействия между сотрудниками, защита данных, геймификация, поддержка мобильных приложений, возможность интеграции с другими сервисами.

Изученные платформы можно условно разделить:

- на комплексные – платформы, включающие в себя несколько модулей управления персоналом (подбор, мотивация, HR-аналитика) и являющиеся неким цифровым «двойником» самой организации;
- унитарные – платформы, предлагающие сугубо одно решение или одно направление в управлении персоналом.

Эти платформы, как правило, связаны с отдельными процессами в работе организации, такими как подбор, обучение персонала. Примерами таких решений являются такие сервисы открытой среды, как признание достижений, «биржа (банк) идей».

Имеющиеся в HR сфере цифровые решения также можно классифицировать по принципу ценообразования:

1. Модели «маркетплейса». Приобретая такие решения, работодатель платит процент от своих расходов на платформе, при этом использование платформ, как правило, начинается бесплатно (за исключением сборов за внедрение), что облегчает их запуск.

2. Модели оплаты за пользователя в месяц.

Общими элементами сервисов являются:

- наличие визуального профиля сотрудника;
- яркая визуализация страницы (с элементами корпоративного стиля организации);
- отсутствие перегруженности элементами;
- четкая структура размещения информации;
- оптимальное (почти минимальное) количество текста;
- наличие виджетов;
- минимальное количество полей для заполнения;
- понятные функции («поблагодарить», «подать идею», т.п.).

Сервис признания достижений позволяет сотрудникам благодарить друг друга напрямую (горизонтально) с учетом ценностей организации. Корпоративные ценности представлены на открытках, доступных сотрудникам для отправки друг другу.

Сервис «биржа идей» дает возможность сотрудникам подавать свои идеи о внесении улучшений в деятельность организации и самостоятельно просматривать статус заявок.

В ходе изучения доступных сервисов были выделены общие требования к ним, в частности:

- организационные требования (тематика взаимодействия пользователей; корпоративные ценности для выстраивания ценных для компании оснований для признания достижений и оценки идей; маршрутизация открыток и идей);
- технические (ИТ) требования (к операционным системам; поддержка платформ iOS, Android, web и desktop, иных);
- требования к защите данных (поскольку происходит обработка персональных данных работников);
- требования к интерфейсу (понятный интерфейс, привлекательный дизайн, возможность его перенастройки и актуализации; корпоративный стиль);

– требования к аналитике (возможность формирования отчетности по активности пользователей, определение наиболее активных, т.п.);

– наличие элементов геймификации (подсчет баллов, активности пользователей, определение победителей, т.п.).

Изучение существующих решений цифровых сервисов позволило выявить общие (устойчивые) характеристики для проектирования прикладного решения для нужд конкретной организации.

Библиографический список

1. Логинова Е.В. Трансформация управления человеческими ресурсами в современных условиях: цифровой HR как новый способ кадровых решений // Инновационные доминанты социально-трудовой сферы: экономика и управление: материалы ежегодной междунар. науч.-практ. конф. по проблемам социально-трудовых отношений, Воронеж, 23 мая 2019 г. / ред. коллегия: А.А. Федченко, О.А. Колесникова. – Воронеж: Истоки, 2019. – С. 167–170.

2. Крылова Т.В., Трушкова Д.М., Фомина Н.И. Применение цифровых технологий в управлении предприятием // Инновационная экономика: перспективы развития и совершенствования. – 2021. – № 5 (55). – С. 87–91. DOI: 10.47581/2021/PS92/IE.5.55.13

3. Логунова И.В. Формирование инновационных подходов к управлению человеческими ресурсами организации [Электронный ресурс] // Экономинфо. – 2018. – № 1. – URL: <https://cyberleninka.ru/article/n/formirovanie-innovatsionnyh-podhodov-k-upravleniyu-chelovecheskimi-resursami-organizatsii> (дата обращения: 23.04.2023).

Сведения об авторах

Касимова Зульфия Гаппасовна – магистрант Физико-математического института Пермского государственного национального исследовательского университета, гр. ФИТ-3-2022, г. Пермь, e-mail: zulfiya_kasimova@yahoo.com

Кушев Вадим Олегович – кандидат педагогических наук, доцент кафедры «Информационные технологии» Физико-математического института Пермского государственного национального исследовательского университета, г. Пермь, e-mail: vadimk@psu.ru

М.В. Болотов, А.Л. Погудин

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА РЕШЕНИЯ ЗАДАЧИ СОСТАВЛЕНИЯ КОМПОЗИЦИИ НА ПРОГРАММНОМ УРОВНЕ

Данная статья посвящена процессу решения задачи автоматизации производственного процесса, а именно составлению композиции на программном уровне. В статье приведен способ расчета расходов необходимых видов сырья относительно расхода основного сырья в соответствии с заданной композицией, необходимой для производства качественной продукции определенного сорта. Данный способ реализован в среде разработки программного обеспечения для автоматизированных систем управления технологическими процессами. Приведенный способ проверен на работоспособность посредством симулятора.

Ключевые слова: разработка алгоритма, технологический процесс, автоматизированные системы управления технологическими процессами, программирование контроллеров.

M.V. Bolotov, A.L. Pogudin

Perm National Research Polytechnic University, Perm

DEVELOPING A SOLUTION TO THE COMPOSITION PROBLEM AT THE PROGRAM LEVEL

This article is devoted to the process of solving the problem of automating the production process, namely, composing a composition at the software level. The article provides a method for calculating the costs of the necessary types of raw materials, relative to the consumption of the main raw materials, in accordance with a given composition necessary for the production of high-quality products of a certain type. This method is implemented in a software development environment for automated process control systems. The above method has been tested for functionality using a simulator.

Keywords: algorithm development, technological process, automated process control systems, controller programming.

Программируемые логические контроллеры (ПЛК) играют ключевую роль в современной промышленности, обеспечивая автоматизацию и эффективное управление производственными процессами. Написание программ для ПЛК – это процесс, который требует не только технической компетентности, но и понимания технологии производственного процесса.

Программы ПЛК позволяют автоматизировать множество операций, начиная от управления конвейерами и оборудованием до контроля температуры, давления и других параметров. Это позволяет значительно снизить вероятность человеческих ошибок, увеличить скорость выполнения задач и сократить время простоя оборудования. Более того, программируемые логические контроллеры обладают гибкостью в настройке, что делает их идеальным инструментом для адаптации к различным производственным условиям и требованиям. Благодаря возможности быстрого изменения программного кода, они позволяют реагировать на изменения в производственных процессах с минимальными затратами времени и ресурсов [1].

Актуальность исследования обусловлена повышением эффективности и надежности автоматизированной системы управления технологическим процессом.

Объектом исследования является технологический процесс составления композиции в машинном бассейне бумагоделательной машины.

Предметом исследования является процесс разработки и реализации алгоритма для программируемого логического контроллера.

Целью данного исследования является решение производственной задачи автоматического составления и поддержания заданной композиции в машинном бассейне бумагоделательной машины.

Для достижения поставленной цели решены следующие задачи: проведен обзор литературы по теме автоматизации технологических процессов с анализом научных публикаций, разработана логика, реализована и проверена на работоспособность программа для ПЛК.

Средства разработки программного обеспечения для ПЛК. В рамках данного проекта автоматизированная система управления реализуется с помощью контроллера Siemens Simatic S7-1500, следовательно, ПО разрабатывалось в интегрированной среде разработки Tia Portal [2].

TIA Portal (Totally Integrated Automation Portal) – это интегрированная среда разработки от Siemens, предназначенная для проектирования, программирования, настройки и обслуживания различных компонентов автоматизации промышленных предприятий. Это мощный инструмент, который обеспечивает единый интерфейс для работы с различными устройствами, включая программируемые логические

контроллеры (ПЛК), человеко-машинные интерфейсы (HMI), приводы, сетевое оборудование и датчики.

Рассмотрим основные компоненты и возможности TIA Portal:

1) Программирование ПЛК.

TIA Portal предоставляет разнообразные языки программирования для ПЛК, такие как LAD (лестничная диаграмма), FBD (функциональная 블록ная диаграмма), SCL (структурированный язык управления) и STL (язык логического переключения). Интегрированная среда разработки позволяет создавать и отлаживать программы для различных моделей ПЛК Siemens, таких как SIMATIC S7-1200, S7-1500 и др.

2) Настройка HMI.

С помощью TIA Portal можно разрабатывать пользовательские интерфейсы для взаимодействия с операторами и мониторинга производственных процессов. Инструменты проектирования HMI в TIA Portal обеспечивают возможность создания графических элементов, анимаций, трендовых графиков и других компонентов для визуализации данных.

3) Настройка приводов и устройств ввода-вывода.

TIA Portal обеспечивает интеграцию с различными устройствами приводов и ввода-вывода, позволяя конфигурировать их параметры и связывать с программой управления. Это включает в себя настройку переменных, параметров обмена данными и диагностики оборудования.

4) Сетевое управление.

TIA Portal обладает возможностями настройки сетевой связи между устройствами, включая Ethernet, PROFIBUS и PROFINET. Это позволяет управлять обменом данными между различными компонентами системы автоматизации и настраивать сетевые параметры.

5) Отладка и тестирование.

Среда TIA Portal предоставляет инструменты для отладки и тестирования программного обеспечения, включая встроенные средства мониторинга переменных, симуляцию работы программы и анализ журналов событий. В целом TIA Portal обеспечивает интегрированный и удобный рабочий процесс для разработки и обслуживания систем автоматизации, снижая время и ресурсы, необходимые для развертывания и поддержки промышленных процессов.

Разработка алгоритма и реализация его на программном уровне.
Задача автоматизации состоит в следующем: в машинном бассейне бумагоделательной машины составляется определенная композиция не-

обходимого для дальнейшего производства бумаги сырья, в ее состав входят три компонента: полуцеллюлоза, макулатура и брак. Технологический процесс производства бумаги является непрерывным, значит, и композицию в машинном бассейне необходимо поддерживать непрерывно. Данный технологический процесс необходимо автоматизировать в целях оптимизации производства бумаги [3].

В соответствии с разработанным ранее техническим заданием имеем следующее оборудование:

- 1) программируемый логический контроллер «Siemens Simatic S7 – 1500 CPU 1511-1 PN» – 1 шт.;
- 2) датчик расхода электромагнитный Endress-Hauser Promag 50 – 4 шт.;
- 3) датчик уровня гидростатический «Endress-Hauser Cerabar PMP51B» – 1 шт.;
- 4) позиционер для клапана «Metso NE724/S1-CE01» – 4 шт.

Поддержание уровня в машинном бассейне реализовано через контур регулирования: токовый сигнал 4-20мА, поступающий с гидростатического датчика уровня на AI порт контроллера, масштабируется и передается на ПИД регулятор, далее в соответствии с заданной установкой регулятор генерирует цифровой сигнал, который после преобразования через цифро-аналоговый преобразователь с порта АО контроллера подается на позиционер клапана, установленного на потоке полуцеллюлозы. Вследствие регулирования уровня меняется и значение расхода полуцеллюлозы. Чтобы высчитать необходимые расходы макулатуры и брака, была реализована функция на структурированном языке управления SCL (Structured Control Language), указанная на рис. 1.

Из функции, обозначенной на рис. 1, видно, что расход макулатуры высчитывается следующим образом: фактический расход полуцеллюлозы, зависящий от уровня в машинном бассейне, как указывалось ранее, делится на процент полуцеллюлозы в композиции, далее данное значение умножается на процент макулатуры в композиции, на выходе формулы получаем значение расхода макулатуры, полученное значение, в свою очередь, попадает в контур регулирования, управляющий позиционером клапана на потоке макулатуры, который будет обеспечивать и поддерживать полученное в формуле значение расхода макулатуры. Обеспечение необходимого расхода брака реализовано таким же образом. В функции предусмотрено условие, что при нулевом расходе полуцеллюлозы расходы на остальных потоках сырья также будут равны нулю.

Далее разработанная функция добавляется в основной цикл контроллера и осуществляется привязка к ее входам и выходам необходимых переменных с помощью языка линейных диаграмм LAD (Ladder Diagram).

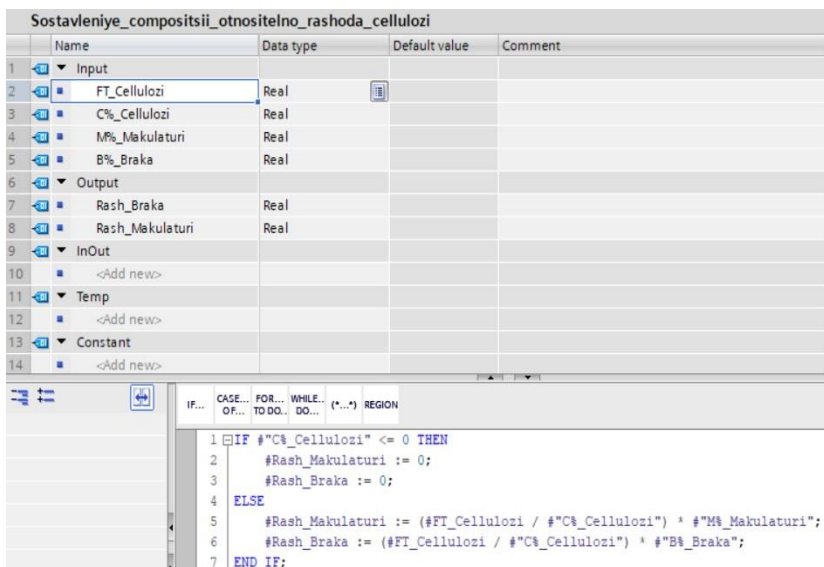


Рис. 1. Функция для вычисления расходов

На рис. 2 обозначены следующие переменные: FT_Cellulozi – фактический расход целлюлозы, %_C – задаваемый процент целлюлозы для композиции, %_M – задаваемый процент макулатуры для композиции, %_B – задаваемый процент брака для композиции, Q_Braka – расход брака после вычисления в функции, Q_Makulaturi – расход макулатуры после вычисления в функции.

Проверка работоспособности разработанной функции

После написания функции было проведено ее тестирование посредством симулятора.

На рис. 3 можно наблюдать, что у ставки процентов композиции заданы следующим образом: целлюлоза – 50 %, макулатура – 30 %, брак – 20 % (%_C = 50, %_M = 30, %_B = 20), фактический расход целлюлозы равен 5425.347 (FT_Cellulozi = 5425.347), а на выходе функции указан соответствующий расход на потоках брака и макулатуры

($Q_{Braka} = 2170.139$, $Q_{Makulaturi} = 3255.208$), данные значения поступают на ПИД регуляторы, которые дают соответствующие сигналы на позиционеры клапанов, тем самым открывая или закрывая клапаны на необходимые значения, что, соответственно, постоянно поддерживает заданную композицию в машинном бассейне.

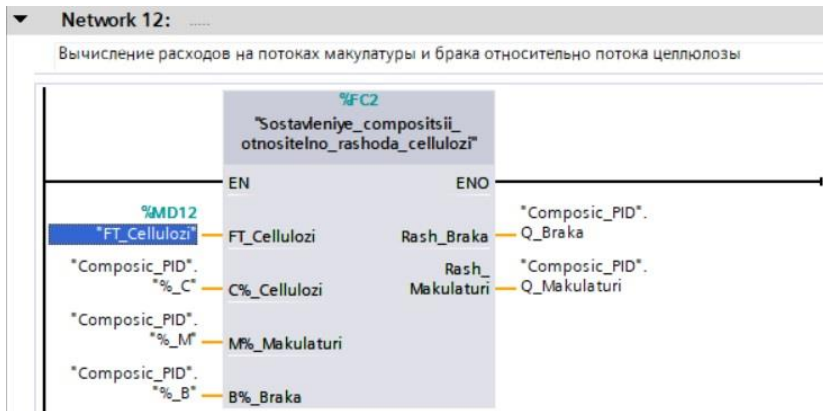


Рис. 2. Блок вычисления расходов

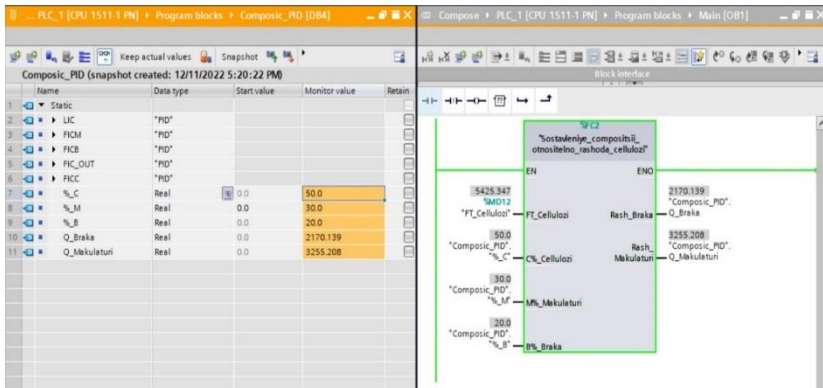


Рис. 3. Проверка функции составления композиции

Из анализа работы функции можно сделать вывод, что композиция составляется корректно и функция правильно выполняет свои действия.

Библиографический список

1. Буйлов Г.П. Автоматика и автоматизация производственных процессов целлюлозно-бумажного производства. – СПб.: ВШТЭ СПбГУПТД. 2019. – 100 с.

2. TIA Portal Tutorial Center [Электронный ресурс]. – URL: [https://support.industry.siemens.com/cs/document/106656707/the-tia-portal-tutorial-center-\(videos\)?dti=0&lc=en-WW](https://support.industry.siemens.com/cs/document/106656707/the-tia-portal-tutorial-center-(videos)?dti=0&lc=en-WW) (дата обращения: 10.04.2024).

3. Хмара Д.О., Паршина Г.И. Разработка SCADA – системы для научно-исследовательского комплекса [Электронный ресурс] // Форум молодых ученых. – 2021. – № 4 (56). – URL: <https://cyberleninka.ru/article/n/razrabotka-scada-sistemy-dlya-nauchno-issledovatel'skogo-kompleksa> (дата обращения: 16.04.2024).

Сведения об авторах

Болотов Максим Вячеславович – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ4-22-1м, г. Пермь, e-mail: m.bolotov2011@yandex.ru

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

А.Л. Погудин, Е.А. Хохрякова

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА КОЛЕСНОГО УЗЛА ДЛЯ МОБИЛЬНОЙ РОБОТОТЕХНИЧЕСКОЙ ПЛАТФОРМЫ НА ОСНОВЕ АНАЛИЗА КОНСТРУКТОРСКИХ ОСОБЕННОСТЕЙ КОЛЕСНЫХ ПАР

В статье рассматриваются различные типы колесных пар, используемых при разработке мобильных роботов, и определяется оптимальный вариант колеса для проектируемой мобильной платформы. Также разработан узел колеса для проектируемой платформы.

Ключевые слова: мобильная робототехническая платформа, типы колес, механум-колеса.

A.L. Pogudin, E.A. Khokhryakova

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A WHEEL UNIT FOR A MOBILE ROBOTIC PLATFORM BASED ON THE ANALYSIS OF DESIGN FEATURES OF WHEEL SETS

The paper examines the different types of wheelsets used in the development of mobile robots and identifies the optimum wheel option for the designed mobile platform. A wheel assembly for the designed platform is also developed.

Keywords: mobile robotics platform, wheel types, mecanum wheels.

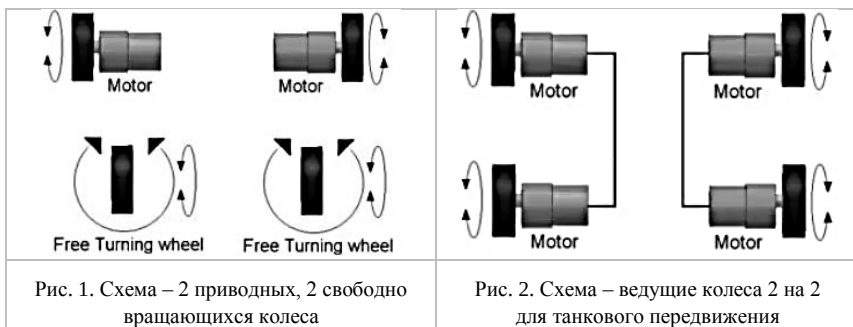
Актуальность данного исследования обусловлена необходимостью разработки надежной и маневренной мобильной робототехнической платформы для использования при выполнении лабораторных и практических работ, написании курсовых проектов и выпускных квалификационных работ. Цель данной работы – разработать колесный узел для мобильной робототехнической платформы. Объектом работы является колесный узел мобильной робототехнической платформы. Предметом работы является процесс разработки колесного узла для мобильной робототехнической платформы.

В современном мире тенденции по развитию мобильных платформ направлены на создание гибких в управлении аппаратов и увеличение их маневренности. Для достижения данных целей используются роликоне-

сущие колеса в конструкции платформ. Благодаря этому решению обеспечивается абсолютная подвижность движения платформы, при рассмотрении движения на плоскости это означает, что для управления доступны все возможные степени подвижности. Системы с управлением по всем степеням подвижности называются голономными. В таких системах накладываются ограничения только на координаты, скорости точек не имеют ограничений. Так как разрабатываемая платформа будет использоваться в условиях ограниченной площади и с препятствиями различного вида, то необходимым условием ставится высокая мобильность всей платформы. Для выбора колес для мобильной робототехнической платформы проведем анализ следующих типов колес: ведущие колеса 2 на 2 для танкового передвижения; 2 приводных, 2 свободно вращающихся колеса; рулевое управление, как у автомобиля; омниколесный робот и меканум-колесо. Данный анализ представлен в таблице.

Анализ типов колесных схем

Тип колес	Схема	Достоинства	Недостатки
2 приводных, 2 свободно вращающихся колеса	Рис. 1	Простота реализации	Низкая мобильность и управляемость. Низкая курсовая устойчивость
Ведущие колеса 2 на 2 для танкового передвижения	Рис. 2	Простота реализации	Снижение эффективности из-за проскальзывания колес
Рулевое управление как у автомобиля	Рис. 3	Высокая курсовая устойчивость. Высокая скорость передвижения	Большое количество деталей
Омниколесный робот	Рис. 4	Любое движение с любым вращением	Сложность в программировании
Колесо Mecanum	Рис. 5	Любое движение с любым вращением	Сложность в программировании



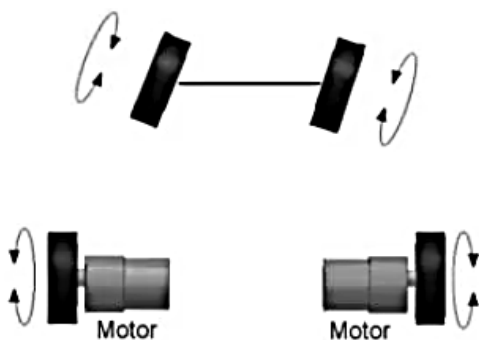


Рис. 3. Схема – рулевое управление, как у автомобиля

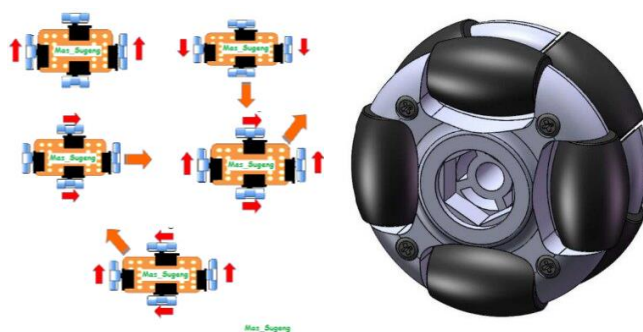


Рис. 4. Схема омникопесного робота

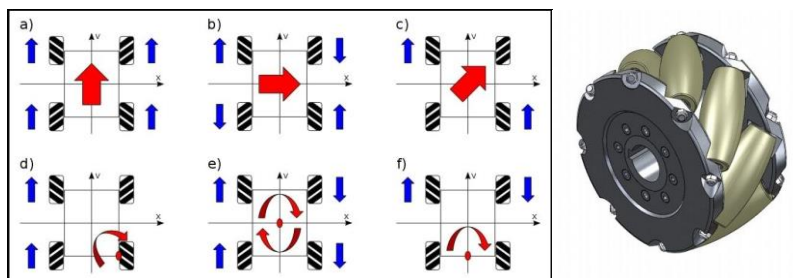


Рис. 5. Схема колеса Mecanum

Исходя из поставленного условия из всех перечисленных типов колес для разрабатываемой платформы больше всего подходят омни-копесный робот и колеса Mecanum.

В исследовании [1] анализируются конструкции мобильных платформ на базе колес с омни- и меканум-колесами. Омни-колеса отличаются направлением оси роликов по всей длине колеса, в то время как меканум-колеса имеют наклон оси ролика на 45 градусов. Благодаря наклону оси меканум-колеса обладают повышенной способностью переносить грузы и обеспечивают постоянный контакт с поверхностью в отличие от омни-колес [2]. Это делает меканум-колеса предпочтительным выбором для конструирования шасси для грузовых платформ, что подтверждается практикой использования в промышленности. Поворот и передвижение платформы по любому направлению достигается за счет взаимодействия вращающихся меканум-колес с поверхностью [3].

Расположение и количество меканум-колеса в шасси платформ варьируются. Минимальное количество колес для обеспечения максимальной подвижности составляет три [4]. При этом важно, чтобы оси всех роликов не были параллельны друг другу [5]. Исходя из этих критериев, определяется оптимальная конфигурация шасси. Схематическое представление конфигурации шасси для разработки платформы показано на рис. 6.

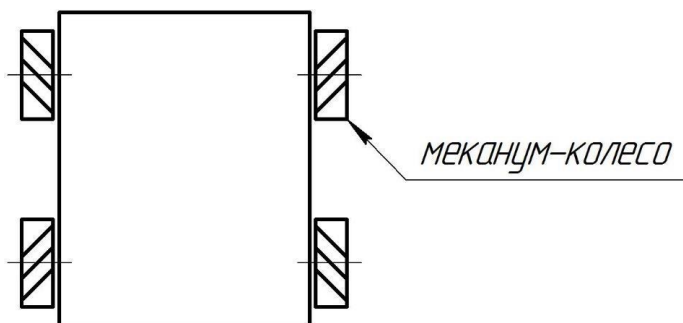


Рис. 6. Конфигурация шасси

Эта конфигурация включает в себя четыре колеса, по два на каждой стороне, и все колеса расположены параллельно друг другу. На изображении показаны ролики на периферии колеса.

Каждый ролик меканумного колеса может неограниченно вращаться вокруг своей оси, которая остается неподвижной относительно колеса. Схематическое изображение таких колес показано на рис. 7.

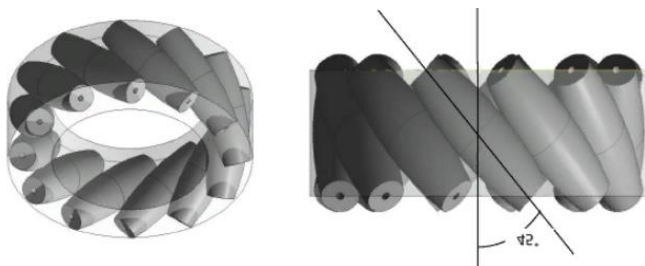


Рис. 7. Меканум-колесо

На основе выбранных колес разработаем колесный узел мобильной робототехнической платформы.

В мобильной платформе присутствуют четыре одинаковых колесных узла (из-за симметрии конструкции шасси). Кинематическая схема узла изображена на рис. 8. Она включает в себя меканум-колесо, двигатель постоянного тока, редуктор, несколько подшипников, жесткую и гибкую муфты.

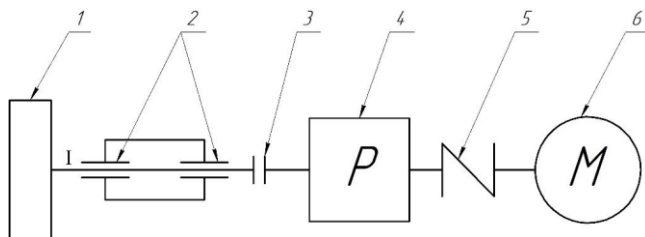


Рис. 8. Схема колесного узла

Колесо 1 закреплено на валу I, который установлен в подшипниковых парах 2 как по радиусу, так и по оси. С помощью жесткой муфты от редуктора 4 на вал передается поворот. Вал постоянного тока 6 с редуктором соединяются при помощи гибкой муфты.

В данной работе был произведен анализ пяти типов колес, которые применяются при создании мобильных робототехнических платформ. На основе этого анализа были выбраны меканум-колеса для разрабатываемой платформы. И на основе выбранных колес был разработан колесный узел, состоящий из меканум-колеса, двигателя постоянного тока, редуктора, нескольких подшипников, жесткой и гибкой муфт.

Библиографический список

1. Spyros G. Tzafestas. Introduction to Mobile Robot Control. – Elsevier, 2014. – 750 p.
2. Doroftei I., Grosu V., Spinu V. Omnidirectional Mobile Robot – Design and Implementation // Bioinspiration and Robotics: Walking and Climbing Robots. – 2007. – P. 511–528.
3. Gfrerrer A. Geometry and Kinematics of the Mecanum Wheel // Computer Aided Geometric Design. – 2008. – Vol. 25. – P. 784–791.
4. Kumar, Pushendra & Othman, Lakhal & Merzouki, Rochdi. Bond Graph Modeling of a Holonomic Mobile Platform with Four Mecanum Wheels. – 2018. – 8 p. (Preprint ICBGM).
5. Борисов А.В., Килин А.А., Мамаев И.С. Тележка с омниколесами на плоскости и сфере // Нелинейная динамика. – 2011. – Т. 7, № 4 (Мобильные роботы). – С. 785–801.

Сведения об авторах

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

Хохрякова Елизавета Андреевна – магистрант Пермского национального исследовательского политехнического университета, гр. ИПР-22-1м, г. Пермь, e-mail: elizaveta-kireeva@yandex.ru

Хиеу Ву Динь, Р.А. Файзрахманов

Пермский национальный исследовательский
политехнический университет, Пермь

**МАТЕМАТИЧЕСКИЕ И АЛГОРИТМИЧЕСКИЕ ОСНОВЫ
ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА
И УПРАВЛЕНИЯ ТЕХНИЧЕСКИМ ОБСЛУЖИВАНИЕМ
ОБОРУДОВАНИЯ С ИСПОЛЬЗОВАНИЕМ
МОДИФИЦИРОВАННОЙ МОДЕЛИ ИНДЕКСА
ТЕХНИЧЕСКОГО СОСТОЯНИЯ**

Предложены для обсуждения авторская модель и алгоритм информационной системы мониторинга и управления техническим обслуживанием оборудования с использованием разработанного авторами индекса технического состояния (далее ИТС) сложных технических объектов. ИТС сложного технического объекта оценивают по ИТС основных узлов и их весов, ИТС узлов оценивают по ИТС контролируемых параметров и весов этих параметров. Информационная система, основанная на данной модели, позволяет детально планировать обслуживание технического объекта, его составных частей на основе конструктивного подхода, обеспечивающего минимизацию человеческого фактора и экспертных оценок, и использовать предложенную экспресс-оценку ИТС.

Ключевые слова: информационная система, модели и алгоритмы, система технического обслуживания, структурная модель, индекс технического состояния, сложная техническая система.

Hieu Wu Din, R.A. Fayzrakhmanov

Perm National Research Polytechnic University, Perm

**MATHEMATICAL AND ALGORITHMIC FUNDAMENTALS
INFORMATION SYSTEM FOR MONITORING AND MANAGEMENT
MAINTENANCE OF EQUIPMENT USING A MODIFIED
MODEL TECHNICAL CONDITION INDEX**

The authors's model and algorithm for an information system for monitoring and managing equipment maintenance using the technical condition index (TCI) of complex technical objects developed by the authors are proposed for discussion. The TCI of a complex technical object is assessed by the TCI of the main components and their weights, the TCI of the nodes is assessed by the TCI of controlled parameters and the weights of these parameters. An information system based on this model makes it possible to plan in detail the maintenance of a technical object and its components

based on a constructive approach that ensures minimization of the human factor and expert assessments and to use the proposed express assessment of TCI.

Keywords: information system, models and algorithms, maintenance system, structural model, technical condition index, complex technical system.

Оптимизация производственных затрат до сих пор является одним из главных приоритетов бизнеса. Между тем затраты на техническое обслуживание и ремонт систем составляют значительную долю затрат бизнеса. Поэтому для оптимизации этих затрат сегодня многие предприятия перешли на подход технического обслуживания и ремонта по фактическому техническому состоянию оборудования. Этот подход доказал свою эффективность на практике [1].

Наряду с развитием науки и техники, сегодня машины и оборудование, применяемые в промышленном производстве, представляют собой сложные объекты или системы. Сложные технические объекты или системы – это системы, характеризующиеся сложными зависимостями между их компонентами [2]. Такие системы применяются в различных областях, таких как энергосистемы, нефтехимические системы, транспортные системы, аэрокосмические системы и т.д.

В данной работе представлена информационная система мониторинга и управления процессом технического обслуживания, основанная на структурной модели, оценивающей ИТС сложных технических объектов. Информационная система, основанная на данной модели, позволяет детально планировать обслуживание технического объекта, его составных частей на основе конструктивного подхода, обеспечивающего минимизацию человеческого фактора и экспертных оценок, и использовать предложенную экспресс-оценку ИТС.

Информационная система технического обслуживания. Предлагаемая информационная система технического обслуживания сложных технических объектов на основе структурной модели оценки их ИТС представлена на рис. 1.

Параметры, формируемые в процессе работы сложного технического объекта, используются для оценки текущего ИТС объекта и его компонентов. Система технического обслуживания базируется на техническом состоянии объекта, а также его составных частей, точнее, на значении ИТС для планирования технического обслуживания и оказания технического воздействия на технический объект, а также его составные части.

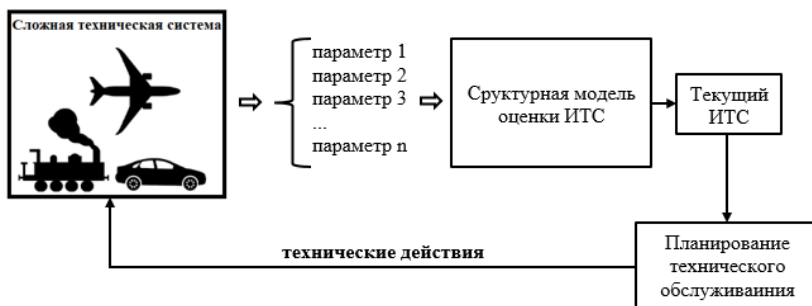


Рис. 1. Информационная система технического обслуживания на основе структурной модели оценки ИТС

ИТС системы или сложного технического объекта, характеризующий ее состояние, а также работоспособность, оценивается на основе предложенной структурной модели, представленной на рис. 2.

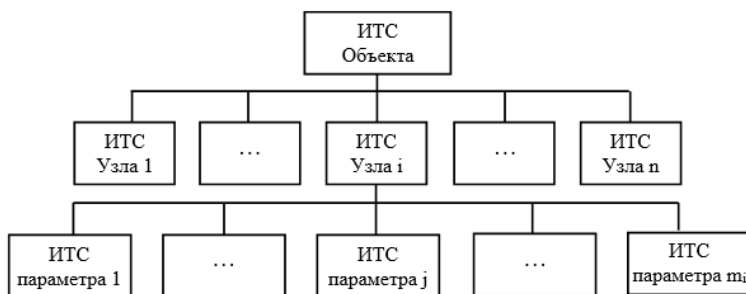


Рис. 2. Структурная модель оценки ИТС сложных технических объектов

ИТС находится в диапазоне от 0 до 100 (может рассчитываться в процентах), при этом 100 – лучшее техническое состояние на момент начала эксплуатации объекта, а 0 – худшее техническое состояние технического объекта. ИТС сложного технического объекта оценивают по ИТС основных узлов и их весов, ИТС узлов оценивают по ИТС контролируемых параметров и веса этих параметров [3]. ИТС объекта (I), основных функциональных узлов и параметров управления рассчитывают по формулам (1), (2) и (3а или 3б) соответственно.

$$I = \sum_{i=1}^n (C_i \cdot I_i), \quad (1)$$

где n – число основных функциональных узлов в объекте; C_i – вес i -го основного функционального узла; I_i – ИТС i -го основного функционального узла.

$$I_i = \sum_{j=1}^{m_i} (w_{i,j} \cdot I_{i,j}), \quad (2)$$

где m_i – количество контролируемых параметров ТС i -го узла; $I_{i,j}$ – ИТС j -го параметра ($j = 1..m_i$) i -го узла ($i = 1..n$); $w_{i,j}$ – вес j -го параметра ($j = 1..m_i$) i -го узла ($i = 1..n$).

$$I_{i,j} = \frac{(x_{i,jT} - x_{i,j\min})100}{x_{i,j\max} - x_{i,j\min}}, \quad (3a)$$

$$I_{i,j} = \frac{(x_{i,jT} - x_{i,j\max})100}{x_{i,j\min} - x_{i,j\max}}, \quad (3б)$$

где $I_{i,j}$ – ИТС j -го параметра i -го узла; где $i = 1..n$, $j = 1..m_i$; n – количество основных функциональных узлов рассматриваемого оборудования; m_i – количество параметров для i -го узла; $x_{i,jT}$ – текущее значение j -го параметра i -го узла, определенное из системы мониторинга в момент времени оценки ТС; $x_{i,j\max}$, $x_{i,j\min}$ – максимальное и минимальное значения j -го параметра i -го узла, определенные по паспортным данным или данным испытаний. Формула (3a) применяется к параметрам, которые постепенно уменьшаются с течением времени в процессе работы устройства. В случае постепенного увеличения параметров в течение рабочего времени применяется формула (3б).

Новизна предложенного подхода заключается в том, что веса основных функциональных узлов объекта и контролируемых параметров не будут рассчитываться на основе субъективности экспертов традиционным экспертным методом. Веса функциональных узлов рассчитываются на основе оценки коэффициента важности узлов в функциональной структурной схеме, представленной в виде графа, а веса контролируемых параметров – на основе сравнения их статистических показателей, коэффициентов вариации этих параметров [3].

Эксперимент. Статистические данные контролируемых параметров для оценки ИТС взяты из набора данных FD001, предоставленного NASA [4], описывающего деградацию контролируемых параметров 100 турбореактивных двухконтурных двигателей с момента их первого ввода в эксплуатацию до момента их вывода из эксплуатации.

Результаты расчета ИТС первых трех двигателей (формула (1)) по набору данных для разных моментов времени (циклов) $T = 50, 100, 150$ представлены в таблице и на рис. 3.

ИТС (в %) первых трех двигателей для $T = 50, 100, 150$

T (цикл)	Двигатель 1	Двигатель 2	Двигатель 3
50	92,4	92,9	94,3
100	83,4	89,3	86,1
150	53,4	82,6	46,5

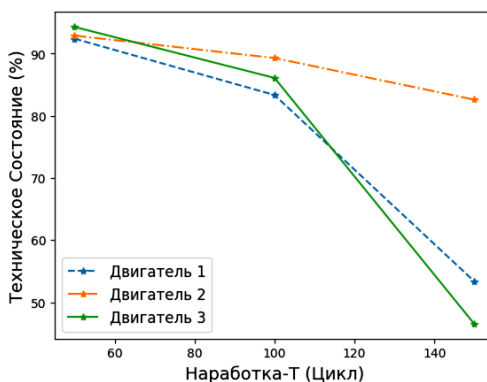


Рис. 3. ИТС трех двигателей по набору данных для разных моментов времени (циклов) $T = 50, 100, 150$

Из результатов расчета видно, что среди трех рассмотренных двигателей быстрее всего снижался ИТС двигателя 3, за ним следовал двигатель 1, а медленнее всего снижался двигатель 2. Согласно информации в наборе данных, предоставленном NASA, двигатель 1 работал до момента времени (циклов) $T = 192$, затем вышел из строя, двигатель 2 работал до момента времени $T = 287$, затем вышел из строя, а двигатель 3 работал до момента времени $T = 179$, затем вышел из строя. Это означает, что жизненный цикл двигателя 3 меньше жизненного цикла двигателя 1 и меньше жизненного цикла двигателя 2.

Считаем, что данный подход является развитием методики Минэнерго (Приказ Министерства энергетики РФ от 26 июля 2017 г. № 676 «Об утверждении методики оценки технического состояния основного технологического оборудования и линий электропередачи электрических станций и электрических сетей»).

Заключение. Экспериментально показано, что предложенные информационная система, модели и алгоритмы могут быть применены для создания системы технического обслуживания оборудования с использованием модифицированной модели ИТС. Данная система технического обслуживания позволяет планировать техническое обслуживание объектов и их частей. Как показано в работе [3], предложенный подход имеет определенные преимущества перед методикой Минэнерго.

Библиографический список

1. Иванов В.А., Фещенко А.А. Особенности подходов к техническому обслуживанию и ремонту оборудования в непрерывном производстве // Вестник Пермского национального исследовательского политехнического университета. Машиностроение, материаловедение. – 2018. – Т. 20, № 3. – С. 82–89. DOI: 10.15593/2224-9877/2018.3.10
2. Цветков В.Я. Сложные технические системы // Образовательные ресурсы и технологии. – 2017. – № 3 (20). – С. 86–92.
3. Хиеу В.Д., Файзрахманов Р.А. Методика и алгоритм экспресс оценки состояния сложных технических систем. Конструктивный подход // Инженерный вестник Дона. – 2024. – № 4. – URL: ivdon.ru/ru/magazine/archive/n4y2024/9172
4. Turbofan engine degradation simulation data set. – URL: <https://c3.nasa.gov/dashlink/resources/139/>

Сведения об авторах

Ву Динь Хиеу – аспирант кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: vudinhhieuv20051987@gmail.com

Файзрахманов Рустам Абубакирович – доктор экономических наук, профессор, заведующий кафедрой «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: Fayzrakhmanov@gmail.com

К.А. Иванов, Е.В. Роцин

Белорусский национальный технический университет, Минск

ПРИМЕНЕНИЕ ТЕХНОЛОГИИ «ИНТЕРНЕТ ВЕЩЕЙ» В ЭНЕРГЕТИКЕ

В этой статье рассматривается технология «Интернет вещей», а также возможные варианты применения данной технологии в энергетике. Описывается дефиниция «Интернета вещей», а также плюсы и минусы внедрения данной технологии. Возможные использования технологии «Интернет вещей» в энергетике рассмотрены на таких примерах, как умный город. Делается вывод о желательности внедрения данной технологии в энергетiku и о значительных преимуществах от этого.

Ключевые слова: Интернет вещей, энергетика, преимущества и недостатки технологии «Интернет вещей».

K.A. Ivanou, J.V. Roscyn

Belarusian National Technical University, Minsk

APPLICATIONS OF INTERNET OF THINGS IN ENERGY

This article discusses the Internet of Things, as well as possible applications of this technology in the energy sector. The definition of the Internet of Things is described, as well as the pros and cons of implementing this technology. Possible uses of the Internet of Things in the energy sector are discussed using examples such as a smart city. The conclusion draws our attention to the desirability of introducing this technology into the energy sector and the significant benefits from this.

Keywords: Internet of things, IoT, applications, energy, pros and cons of IoT.

В последнее время все популярнее становится идея Интернета вещей (Internet of things, IoT). Суть этой технологии состоит в том, что к сети Интернет может подключиться любое физическое устройство и девайс для обмена информацией с другими устройствами в сети Интернет. Интернет вещей – это довольно новая и прогрессивная идея. Если изначально Интернет задумывался и реализовывался как место встречи людей с людьми, то теперь Интернет – это место встречи людей с людьми, девайсов (машин, «вещей») с девайсами и девайсов с людьми. Самым распространенным примером применения Интернета

вещей является умный дом: датчики температуры автоматически отправляют сигналы на центральную отопительную систему или кондиционер в случае несоответствия желаемой температуры воздуха с действительной; датчики движения автоматически включают свет в комнате, когда туда заходит человек, и выключают, когда он выходит; каждое утро по расписанию или при срабатывании будильника в спальном комнате раздвигаются шторы, а на кухне включается чайник или кофеварка; и, наконец, в случае срабатывания системы безопасности автоматически вызываются необходимые службы – и все это с полным контролем, отчетами и возможностью детальной настройки каждого отдельного девайса всего в одном приложении на смартфоне [1].

Чтобы работать корректно, технология Интернета вещей должна включать в себя [2]:

- девайсы, ответственные за сбор информации или контролирование конкретным механизмом, как, например, датчики света или сервопривод;
- облачное хранилище для собранных данных;
- программное обеспечение для анализа собранных данных с сенсоров и принятие решений;
- приложение для просмотра текущего состояния системы, ее настройки и отдачи указаний конечным пользователем;
- связь между всеми устройствами и приложениями системы, как, например, Wi-Fi, Bluetooth или LAN.

Технология Интернета вещей, если правильна реализована, предоставляет сразу несколько преимуществ, которые могли бы упростить для различных предприятий контроль производства и безопасность своей территории.

Особенно это актуально для таких сфер, как энергетика или химическая промышленность, где от правильной и эффективной работы сферы зависят множество других сфер жизнедеятельности общества, а любая чрезвычайная ситуация или авария недопустима. Среди плюсов технологии Интернета вещей для энергетики можно выделить следующие [3]:

- круглосуточный и непрерывный контроль над всеми необходимыми объектами, чего невозможно добиться только с помощью работников;

- содержание предприятия с внедренной в производство технологией Интернета вещей обходится дешевле из-за значительной экономии на зарплате сотрудников;

- человек может проигнорировать какую-либо мелкую, по его мнению проблему, не заметить ее, устать, ошибиться или проявить халатность. Всего этого не может быть в случае с машинами и девайсами;

- обладая информацией не про одну конкретную установку, а про все установки всего предприятия, анализирующая программа может выявлять более незначительные отклонения значений от нормы на более ранних стадиях, чем человек, а также предлагать варианты по исправлению проблем или улучшению эффективности работы предприятия;

- в случае каких-либо отклонений от нормы зачастую программное обеспечение может автоматически принять необходимые меры без необходимости вмешательства в процесс человека.

Как и все в этом мире, технология Интернета вещей имеет и свои недостатки: кибербезопасность, сложность установки, дороговизна установки, зависимость от подключения к сети.

Рассмотрим ситуацию: в энергосети работает одна ТЭС и одна солнечная электростанция. Солнечная энергия не является стабильным источником тока, потому необходима ТЭС чтобы в случае повышения спроса на электричество или падения выработки его солнечной электростанцией была возможность ввести дополнительные мощности в производство. Но без данных о текущих уровнях спроса и производства электроэнергии сделать это невозможно: ТЭС не знает, сколько электричества необходимо, и будет продолжать сжигать лишнее топливо, загрязняя атмосферу и нивелируя экологические преимущества от установки солнечной электростанцией.

Технология «Интернет вещей» позволяет в реальном времени отслеживать уровень спроса на электроэнергию и в соответствии с этими данными корректировать уровень производства электроэнергии [4].

Эту идею можно развить еще дальше: в последнее время все чаще звучат идеи о «необычных» методах выработки электроэнергии, как, например, специальные плиты под автомагистралями или местами с большим трафиком пешеходов, которые будут вырабатывать электроэнергию благодаря изменению давления, или фонарные столбы с сол-

нечными фотоэлементами на них. Также не стоит забывать, что любое предприятие или частное домохозяйство может захотеть установить такие элементы на крыше своих зданий и домов и экспортировать излишки электроэнергии. Таким образом, сам город становится одной большой электростанцией и сам вырабатывает хоть и не всю необходимую, но существенную часть электроэнергии для своего существования. Однако все это невозможно представить без технологии «Интернет вещей»: тут она используется для координации потребителей и производителей (в том числе экспортеров) электроэнергии, где первые говорят вторым, сколько энергии необходимо, а вторые предоставляют ровно столько энергии и не тратят лишние ресурсы. Также стоит не забывать, что все эти энергопроизводящие объекты требуют наблюдения и контроля над ими, что было бы значительно труднее обеспечить без технологии Интернета вещей: значительно легче отследить состояние системы и ее неисправности в приложении, чем объехать и проверить каждый объект по всему городу, составить отчет и отправить специалистов вручную. Данная идея является основой на пути к умному городу – городу, который использует искусственный интеллект и Интернет вещей для сбора и анализа информации сразу про весь город, чтобы улучшить городскую среду.

Относительно применения технологии «Интернет вещей» в энергетике можно встретить отдельный термин – «Интернет энергетики» (Internet of Energy). Это является важным шагом на пути от Интернета людей (Internet of People) к Интернету всего (Internet of Everything) [5].

Несмотря на значительное количество положительных сторон, должно пройти время, прежде чем применение какой-либо новой и прогрессивной технологии найдет свое место в какой-либо сфере.

Библиографический список

1. Что такое интернет вещей? Определение и описание [Электронный ресурс]. – URL: <https://www.kaspersky.ru/resource-center/definitions/what-is-iot> (дата обращения: 20.05.2024).
2. What Is the Internet of Things (IoT)? How It Works and Benefits [Электронный ресурс]. – URL: <https://www.investopedia.com/terms/i/internet-things.asp> (дата обращения: 19.05.2024).

3. IoT Energy: 6 Ways the IoT Benefits the Energy Sector [Electronic Resource]. – URL: <https://madappgang.com/blog/iot-in-energy-sector> (дата обращения: 20.05.2024).

4. Use cases of IoT in the energy industry [Электронный ресурс]. – URL: <https://www.mindk.com/blog/iot-in-energy> (дата обращения: 20.05.2024).

5. The Internet of Energy: What is It and Why is It Important [Электронный ресурс]. – URL: <https://fiberroad.com/resources/new-trends/what-is-internet-of-energy-why-is-it-important/> (дата обращения: 23.05.2024).

Сведения об авторах

Иванов Кирилл Анатольевич – студент кафедры «Экономика и организация энергетики» Белорусского национального технического университета, г. Минск, e-mail: balacynka@gmail.com

Рощин Евгений Васильевич – студент кафедры «Экономика и организация энергетики» Белорусского национального технического университета, г. Минск, e-mail: roesusniel@gmail.com

А.В. Вольский, Р.А. Файзрахманов

Пермский национальный исследовательский
политехнический университет, Пермь

МОДЕЛИРОВАНИЕ ПОВЕДЕНИЯ ЛОГИКО-ДИНАМИЧЕСКИХ ОБЪЕКТОВ ДЛЯ КОМПЬЮТЕРНОГО ТРЕНАЖЕРА ПОРТАЛЬНОГО КРАНА

В статье рассмотрена задача моделирования и реализации поведения логико-динамических объектов для компьютерного тренажера крановщика. Рассмотрены существующие подходы к организации движения транспорта в тренажерах. Представлена автоматная модель поведения динамических объектов. Рассмотрен вариант реализации поведения логико-динамических объектов в среде разработки Unity.

Ключевые слова: компьютерный тренажер, автоматная модель, Unity.

A.V. Volskiy, R.A. Fayzrakhmanov

Perm National Research Polytechnic University, Perm

SIMULATION OF THE BEHAVIOR OF LOGICAL-DYNAMIC OBJECTS FOR A COMPUTER SIMULATOR OF A PORTAL CRANE

The article discusses the problem of modeling and implementing the behavior of logical-dynamic objects for a computer simulator for a crane operator. Existing approaches to organizing traffic in simulators are considered. An automatic model of the behavior of dynamic objects is presented. An option for implementing the behavior of logical-dynamic objects in the Unity development environment is considered.

Keywords: computer simulator, automaton model, Unity.

Тренажерные комплексы являются неотъемлемой частью процесса обучения операторов [1]. Использование настоящего крана для обучения сопряжено с рядом трудностей и опасностей. Так, в процессе обучения не исключены всевозможные аварии или поломки оборудования из-за недостатка опыта обучающегося крановщика. Тренажер позволяет имитировать предметную область, снижая тем самым риски и затраты на обучение [2].

В компьютерных тренажерах, выпускаемых различными производителями, уделяется значительное внимание реализации движения транспорта. Так, в тренажерах, выпускаемых фирмой «Кранмастер» [3],

реализовано движение автомобилей и железнодорожного транспорта. Тренажеры компании АBB [4] в дополнение к этому позволяют выбирать различные типы судов и наземного транспорта в режиме онлайн.

Типичная сцена компьютерного тренажера может содержать как статические, так и динамические объекты, в данной работе рассматриваются динамические объекты.

Каждый динамический объект имеет свой набор действий, который он выполняет. Например, можно выделить следующие действия: движение вперед; ожидание; повороты; погрузка / разгрузка.

Эти действия можно представить в виде отдельных состояний. Автоматная модель логико-динамической системы представлена в виде графа на рис. 1 [5].

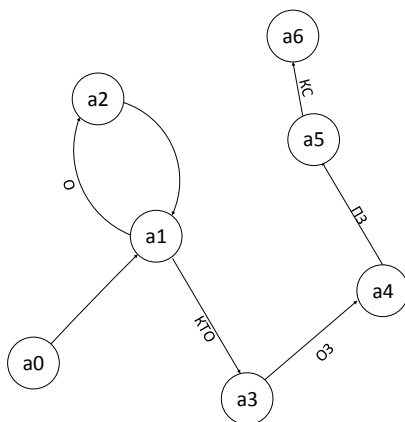


Рис. 1. Граф состояний логико-динамических объектов

Состояния автомата:

- a0 – начальное состояние;
- a1 – состояние движения к точке перегрузки;
- a2 – состояние ожидания;
- a3 – состояние перегрузки;
- a4 – состояние завершения обработки;
- a5 – состояние движения на выезд из порта;
- a6 – состояние выезда из порта.

Входным алфавитом для автомата будут сигналы, формируемые системой. Множество выходных сигналов обозначим через x . Сигнал o за-

нятом месте перегрузки – через *О*. Сигналы формируются системой на основе данных о том, есть или нет транспортное средство в точке перегрузки. Сигнал *НМ* показывает, что есть автомобиль на маршруте к выгрузке. Сигнал о достижении контрольной точки обработки *КТО* формируется системой в момент времени, когда транспортное средство достигло точки, где производится перегрузка. Сигнал о завершении обработки *ОЗ* формируется стропальщиком, когда завершена погрузка или разгрузка груза. Сигнал о завершении подготовки к движению *ПЗ* формируется системой, когда завершена подготовка транспортного средства к движению. Когда объект достигает края сцены, формируется сигнал *КС*. Соответственно входной алфавит для автомата будет $x = \{O, НМ, КТО, ОЗ, ПЗ, КС\}$. Проектируемый автомат является автоматом Мура, и его выходной сигнал будет однозначно определяться состоянием.

В Unity существует несколько вариантов для моделирования движения транспорта [6]: первый – моделирование с использованием физической библиотеки; второй – расчет следующей позиции на основе текущей и заданной скорости движения.

Для реализации автоматной модели в UNITY на языке программирования C# подходит шаблон проектирования, который называется «Состояние» или Finite State Machine (FSM).

Паттерн Finite State Machine (FSM) представляет собой способ моделирования поведения объекта или системы в виде конечного автомата, который может принимать определенные действия в зависимости от текущего состояния и входных данных.

В общем случае существует три ключевых пункта для каждого класса состояния, позволяющих описать поведение состояния: вход (entry), выход (exit) и цикл обновления (update loop), как на рис. 2.

Для реализации шаблона на первом этапе создается абстракция, например интерфейс или абстрактный класс, а его реализациями будут состояния динамического объекта. Диаграмма классов для реализации шаблона представлена на рис. 3. Игровой объект будет взаимодействовать с классами состояний посредством еще одного класса StateMashine, этот класс имеет поле – словарь stateMap, которое содержит все состояния и ссылку на игровой объект.

Класс Object – это реализация игрового объекта, наследник MonoBehaviour. В нем реализуется логика, которая переключает с помощью класса StateMashine состояния в зависимости от игровой ситуации на сцене.

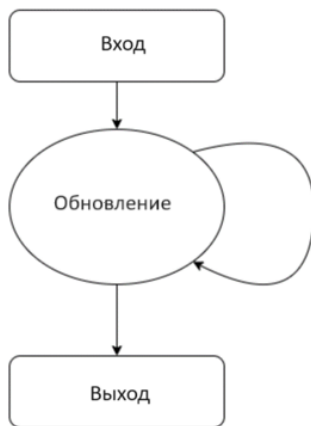


Рис. 2. Концепция работы паттерна Finite State Machine (FSM)

Когда объект достигает определенной точки, то его состояние должно поменяться. Этот механизм реализован с помощью механизма триггеров в среде Unity. На сцене расставлены сферы, определяющие контрольные точки, при вхождении объекта в сферу меняется его состояние.

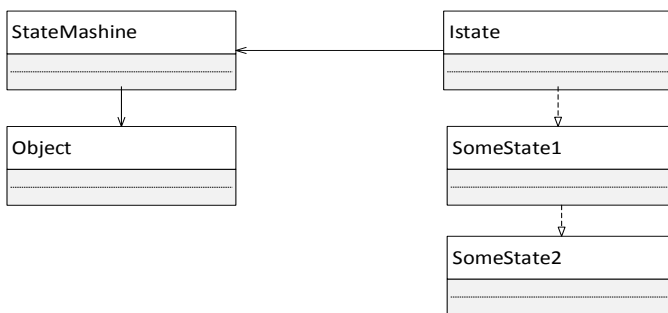


Рис. 3. Диаграмма классов для реализации шаблона FSM

Классы SomeState реализуют логику всех состояний, в которых может оказаться объект.

Рассмотрим движение логико-динамического объекта по сцене на примере грузовика. Грузовики появляются на сцене в определенной точке и после инициализации начинают движение (рис. 4).



Рис. 4. Место появления грузовиков на сцене

По мере появления грузовиков на сцене начинает образовываться очередь на погрузку. На рис. 5 видно, как один из грузовиков находится на погрузке, второй грузовик находится в зоне ожидания, а третий начинает движение в сторону точки разгрузки.

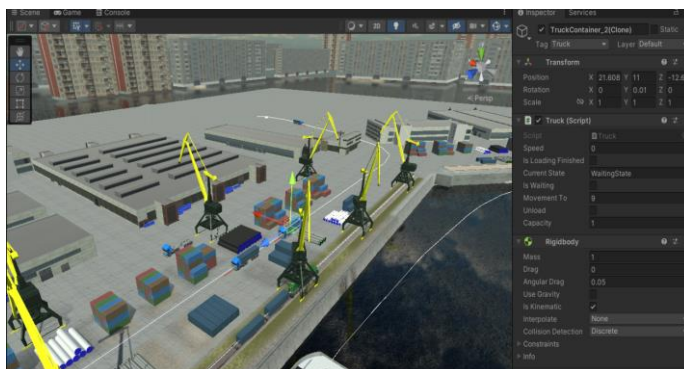


Рис. 5. Очередь из грузовиков

Таким образом, на сцене сразу три грузовика с разными состояниями. После разгрузки первого грузовика он начнет двигаться на выезд из порта и перейдет в соответствующее состояние. Второй грузовик займет его место на разгрузке, а третий переместится на место ожидания и тоже изменит свое состояние.

Разработанная автоматная модель реализована средствами среды разработки Unity и языка программирования C#. После тестирования данная модель будет интегрирована в тренажер портального крана, разработанного на кафедре ИТАС ПНИПУ.

Библиографический список

1. Дозорцев В.М. Компьютерные тренажеры для обучения операторов технологических процессов: теория, методология построения и использования: дис. ... д-ра техн. наук: 05.13.01, 05.13.06. – М., 1999. – 449 с.
2. ANYCRANE: Towards a better Port Crane Simulator for Training Operators / R.A. Fayzrakhmanov, I.S. Polevshchikov, A.F. Khabibulin, F.I. Shklyayev, R.R. Fayzrakhmanov // 15th International industrial simulation conference. – 2017. – Р. 85–87.
3. Mediavisor Тренажер оператора козлового крана [Электронный ресурс]. – URL: <https://mediavisor.ru/produkcija/programmno-apparatni-kompleks/> (дата обращения: 22.02.2024).
4. Crane simulator for operator training - Solutions for marine terminals | ABB [Электронный ресурс] // ABB Group – Leading digital technologies for industry. – URL: <https://new.abb.com/ports/solutions-for-marine-terminals/our-offerings/container-terminal-automation/crane-simulator-for-operator-training> (дата обращения: 22.02.2023).
5. Поликарпова Н.И., Шалыто А.А. Автоматное программирование. – 2008. – 167 с.
6. Unity User Manual [Электронный ресурс]. – URL: <https://docs.unity3d.com/Manual/index.html> (дата обращения: 11.04.2023).

Сведения об авторах

Вольский Арсентий Васильевич – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ6-22-1м, г. Пермь, e-mail: volskiy-av@mail.ru

Файзрахманов Рустам Абубакирович – доктор экономических наук, профессор, заведующий кафедрой «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: fayzrakhmanov@gmail.com

**И.С. Полевщиков^{1,2,3}, М.С. Уколов²,
М.В. Дмитрук³, Р.А. Кравченко²**

¹Пермский национальный исследовательский
политехнический университет, Пермь

²Российский биотехнологический университет, Москва

³Московский государственный университет технологий
и управления имени К.Г. Разумовского, Москва

РАЗРАБОТКА И ПРИМЕНЕНИЕ КОМПЬЮТЕРНЫХ ТРЕНАЖЕРНЫХ КОМПЛЕКСОВ НА ОСНОВЕ ВЕБ-ТЕХНОЛОГИЙ ПРИ ИЗУЧЕНИИ СТУДЕНТАМИ МАТЕМАТИЧЕСКИХ МЕТОДОВ

В статье исследуются особенности создания и применения компьютерных тренажерных комплексов (КТК) для обучения навыкам решения задач по математическим методам в процессе профессиональной подготовки разработчиков автоматизированных систем управления. Продемонстрированы примеры упражнений в КТК по задачам нечеткого моделирования, широко используемого при разработке математического обеспечения автоматизированных систем управления. Применение КТК упрощает процессы настройки преподавателем упражнений. Производится автоматическое оценивание выполнения упражнений, за счет чего повышается степень точности результатов оценивания, экономится время по сравнению с ручной проверкой преподавателем.

Ключевые слова: компьютерный тренажерный комплекс, веб-система, контроль знаний и навыков, нечеткое моделирование, математические методы.

**I.S. Polevshchikov^{1,2,3}, M.S. Ukolov²,
M.V. Dmitruk³, R.A. Kravchenko²**

¹Perm National Research Polytechnic University, Perm

²Russian Biotechnological University, Moscow

³Moscow State University of Technology and Management
named after K.G. Razumovsky, Moscow

DEVELOPMENT OF A WEB-BASED DATA COLLECTION AND ANALYSIS SYSTEM ABOUT THE SCIENTIFIC ACTIVITY OF A UNIVERSITY TEACHER

The article examines the features of the creation and use of computer training complexes (CTC) for teaching skills in solving problems using mathematical methods in the process of professional training of developers of automated control systems (ACS). Examples of exercises in CTC on fuzzy modeling problems, which are widely used in

the development of mathematical software for ACS, are demonstrated. The use of CTC simplifies the process of setting up exercises by the teacher. The exercises are automatically assessed, thereby increasing the accuracy of the assessment results and saving time compared to manual checking by the teacher.

Keywords: computer training complex, web system, control of knowledge and skills, fuzzy modeling, mathematical methods.

Компьютерные тренажеры широко используются для подготовки специалистов различных профессий. Следует выделить научные труды, посвященные разработке и применению компьютерных тренажеров в сфере промышленности, для обучения управления транспортными средствами, в сфере медицины [1–3].

Перспективным направлением развития данных исследований является создание и применение компьютерных тренажеров для обучения разработчиков автоматизированных систем управления (АСУ).

Варианты практической работы по определению характеристик и выполнению операций с нечеткими множествами

Вариант №1	Вариант №2
На универсуме $X = \{21, 22, 23, 24, 25, 26, 27\}$ заданы нечеткие множества: $\mathcal{A} = \{\langle 22, 0.05 \rangle, \langle 23, 0.15 \rangle, \langle 24, 0.2 \rangle, \langle 25, 0.3 \rangle\}$ $\mathcal{B} = \{\langle 23, 0.1 \rangle, \langle 24, 0.25 \rangle, \langle 25, 0.35 \rangle, \langle 26, 1 \rangle, \langle 27, 0.9 \rangle\}$ Задача № 1. Определить для нечеткого множества $\mathcal{B}$: 1) носитель; 2) множество α-уровня при $\alpha = 0,32$; 3) ядро; 4) ближайшее четкое множество. Задача № 2. Определить: 1) $\mathcal{A} \cap \mathcal{B}$; 2) $\mathcal{A} \cup \mathcal{B}$; 3) $\mathcal{A} \setminus \mathcal{B}$; 4) $\mathcal{A} \ominus \mathcal{B}$; 5) $\overline{\mathcal{B}}$	На универсуме $X = \{25, 26, 27, 28, 29, 30, 31, 32\}$ заданы нечеткие множества: $\mathcal{A} = \{\langle 25, 0.13 \rangle, \langle 26, 0.21 \rangle, \langle 27, 0.65 \rangle, \langle 28, 0.92 \rangle, \langle 29, 1 \rangle\}$ $\mathcal{B} = \{\langle 26, 0.18 \rangle, \langle 27, 0.5 \rangle, \langle 28, 0.77 \rangle, \langle 29, 0.95 \rangle\}$ Задача № 1. Определить для нечеткого множества $\mathcal{A}$: 1) носитель; 2) множество α-уровня при $\alpha = 0,55$; 3) ядро; 4) ближайшее четкое множество. Задача № 2. Определить: 1) $\mathcal{A} \cap \mathcal{B}$; 2) $\mathcal{A} \cup \mathcal{B}$; 3) $\mathcal{B} \setminus \mathcal{A}$; 4) $\mathcal{A} \ominus \mathcal{B}$; 5) $\overline{\mathcal{A}}$

В настоящее время в РОСБИОТЕХ для контроля знаний и навыков студентов направлений «Информатика и вычислительная техника», «Управление в технических системах» применяется автоматизированная система (АС) «Успеваемость студентов» – веб-приложение для структурированного хранения и представления информации о выполнении студентами заданий в рамках практических работ.

Рассмотрим типовые практические работы и особенности их оценивания с применением АС «Успеваемость студентов» при изучении математических методов на примере нечетких систем управления как одного из современных направлений в области автоматизации и информатизации [4, 5]. Приведем пример практической работы по определению характеристик и выполнению операций с нечеткими множествами. Данная тема изучается студентами на начальном этапе освоения особенностей нечеткого моделирования. Примеры вариантов задания практической работы приведены в таблице. Предусмотрен веб-интерфейс АС для настройки иерархии критериев преподавателем с целью оценивания практической работы (рис. 1).

		Критерии
№	Название	Вес
1	Задача №1	4
1.1	Носитель	1
1.2	Множество альфа-уровня	1
1.3	Ядро	1
1.4	Ближайшее четкое множество	1
2	Задача №2	5
2.1	Пересечение	1
2.2	Объединение	1
2.3	Разность	1
2.4	Симметрическая разность	1
2.5	Дополнение	1

Рис. 1. Настройка иерархии критериев для оценки практической работы

Преподаватель оценивает результаты выполнения практической работы обучаемым, отчет по которой дистанционно отправлен в АС на проверку, согласно настроенной иерархии критериев. Соответствующий веб-интерфейс результатов оценивания показан на рис. 2. Процесс оценивания включает собственно выставление оценок по 100-балльной шкале и добавление замечаний по задачам, которые выполнены студентом неверно. Итоговая оценка за работу, в рассматриваемом на рис. 2 примере равная 87 %, вычисляется автоматически.

Проверка от 09.11.2023, 07:04 (степень выполнения работы: 87%):

Файл работы:



KP1_oW9nrMi.docx

Критерий	Оценка	Замечание/примечание
1. Задача №1	83	Нет замечаний
1.1. Носитель	100	Нет замечаний
1.2. Множество альфа-уровня	70	Почему элемент 23 присутствует?
1.3. Ядро	100	Нет замечаний
1.4. Ближайшее четкое множество	60	Все ли элементы учли?
2. Задача №2	90	Нет замечаний
2.1. Пересечение	100	Нет замечаний
2.2. Объединение	100	Нет замечаний
2.3. Разность	100	Нет замечаний
2.4. Симметрическая разность	100	Нет замечаний
2.5. Дополнение	50	Перепроверьте, все ли элементы учли.

Дополнительные примечания преподавателя:

Комментария нет!

Рис. 2. Пример оценивания практической работы

После проверки преподавателем результаты оценивания практической работы доступны в личном кабинете студента для просмотра и возможности оставить комментарий преподавателю (например, если у студента возникают трудности с правильным решением задачи).

В целом курс практических работ включает задания, начиная от основ нечетких множеств и нечеткой логики, и завершая заданиями на построение нечетких продукционных систем. В интерфейсе АС предусмотрена настройка весов отдельных практических работ, отражающих их сложность и значимость в рамках учебного курса по нечеткому моделированию.

На основе анализа примеров оценивания практических работ по основам нечеткого моделирования, рассмотренных выше, выделим следующие основные достоинства применения АС «Успеваемость студентов»:

1) наличие веб-интерфейса детализированной настройки критериев оценивания практической работы в зависимости от постановки конкретного задания в области нечеткого моделирования и методики оценивания, используемой преподавателем;

2) наличие веб-интерфейса общей настройки последовательности практических работ и их весов;

3) наличие веб-интерфейса детализированного оценивания преподавателем качества выполнения практической работы конкретным студентом (с автоматическим выставлением итоговой оценки выполнения работы);

4) хранение в АС и просмотр истории оценивания практических работ по каждому обучаемому;

5) наличие формы обратной связи для возможности обучаемому оставить комментарии по замечаниям преподавателя с просьбой пояснить непонятные задачи в рамках практической работы.

Несмотря на большое число достоинств, применение АС «Успеваемость студентов» при оценивании практических работ по нечеткому моделированию содержит такие недостатки, как трудоемкая «ручная» проверка преподавателем результатов выполнения работ, допускающая неточное оценивание и субъективизм. При проверке решения затруднительно точно оценить структуру результатов решения задач и процесс решения с точки зрения математических моделей и алгоритмов в области нечетких множеств и нечеткой логики. Преодолению указанных выше недостатков способствует применение разрабатываемого компьютерного тренажерного комплекса (КТК), являющегося развитием АС «Успеваемость студентов». Прототип КТК на примере упражнения по вычислению характеристик нечетких множеств рассмотрен далее.

На рис. 3 приведен веб-интерфейс преподавателя для настройки указанного выше упражнения. Преподаватель выбирает небольшие задачи в рамках упражнения, по которым студенту требуется вычислить характеристики, определяет веса (уровень сложности задач), выбирает способ выполнения задач, влияющий на сложность решения этих задач студентом. На рис. 4 показан веб-интерфейс обучаемого для выполнения варианта упражнения, сформированного автоматически согласно настройкам.

Характеристика	Вес	Выполнение упражнения		
		☐ Конструирование ответа	☒ Выбор вариантов ответа	☐ Выбор способа решения
☒ носитель	2	☐	☒	☐
☒ множества α -уровня α = от 0.05 до 0.25 +x α = от 0.55 до 0.95 +x	2	☐	☒	☐
☐ высота	1			
☒ ядро	2	☐	☒	☐
☒ границы	2	☐	☒	☐
☒ точки перехода	2	☐	☒	☐
☐ ближайшее четкое множество	2	☐	☒	☐

Рис. 3. Пример настройки преподавателем упражнения в КТК

Дано нечеткое множество $\mathcal{F}$: $\mathcal{F} = \{ \langle 27, 0.12 \rangle, \langle 30, 0.18 \rangle, \langle 32, 0.44 \rangle, \langle 41, 0.9 \rangle, \langle 45, 0.53 \rangle \}$	
Требуется определить характеристики $\mathcal{F}$:	
1. носитель	$\{27, 30, 32, 41, 45\}$
2. множество α -уровня при $\alpha=0,23$	$\{32, 41, 45\}$
3. множество α -уровня при $\alpha=0,82$	$\{41\}$
4. ядро	Результат не определен.
5. границы	$\emptyset$
6. точки перехода	Результат не определен.

Рис. 4. Веб-интерфейс упражнения в личном кабинете студента

На рис. 5 показан веб-интерфейс обучаемого для первой вариации выполнения упражнения, где требуется выбрать элементы множества из списка возможных. На рис. 6 показан веб-интерфейс обучаемого для второй вариации выполнения упражнения, где требуется выбрать способ определения носителя множества, настроив ограничение для $\mu(x)$.

Множество α -уровня при $\alpha=0,23$: {32,41,45}
Выберите элементы множества или укажите, что множество пустое:
☐ 27 ☐ 30 ☒ 32 ☒ 41 ☒ 45 ☐ пустое множество ($\emptyset$)

Рис. 5. Первая вариация выполнения упражнения

Множество α -уровня при $\alpha=0,23$: {32,41,45}	
Выберите способ определения множества – укажите ограничение для $\mu(x)$:	
☐ > ☐ < ☐ = ☒ $\geq$ ☐ $\leq$ ☐ $\neq$	☐ 0 ☒ 0,23 ☐ 0,5 ☐ 0,77 ☐ 1

Рис. 6. Вторая вариация выполнения упражнения

Положительными эффектами внедрения рассматриваемого КТК являются: снижение трудоемкости составления упражнений преподавателем, поскольку не требуется составлять отдельный вариант каждому студенту; повышение объективности и снижение трудоемкости процесса оценивания выполнения упражнений, производимого автоматически на основе анализа свойств изучаемых математических моделей.

Перспектива исследования заключается в совершенствовании КТК для контроля навыков по дисциплинам, связанным с разработкой систем управления, экспертных систем, систем поддержки принятия решений.

Исследование выполнено за счет гранта Российского научного фонда № 23-79-10162, <https://rscf.ru/project/23-79-10162/>.

Библиографический список

1. Мухаметгалиев И.Д., Агзамов З.В. Разработка человеко-машинного интерфейса тренажера наклонно-направленного бурения // Проблемы сбора, подготовки и транспорта нефти и нефтепродуктов. – 2020. – № 1 (123). – С. 38–55.
2. Стариченков А.Л., Стариченкова Е.А. Компьютерный тренажер управления движением судна на подводных крыльях // Морские интеллектуальные технологии. – 2020. – № 2-1 (48). – С. 203–209.

3. Pediatric powered mobility training: powered wheelchair versus simulator-based practice / N. Gefen, Ph.S. Archambault, A. Rigbi, P.L. Weiss // Assistive Technology. – 2023. – Т. 35, № 5. – С. 389–398.

4. Development and application of a web system to improve the professional training of specialists in the automation and informatization of food production / M.M. Blagoveshchenskaya, I.S. Polevshchikov, E.R. Ponasenkov, R.A. Kravchenko // AIP Conference Proceedings. – 2023. – 2627, 070001. DOI: 10.1063/5.0135984

5. Леоненков А.В. Нечеткое моделирование в среде MatLab и fuzzyTECH. – СПб.: БХВ-Петербург, 2005. – 736 с.

Сведения об авторах

Полевщиков Иван Сергеевич – кандидат технических наук, доцент, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета; доцент кафедры «Информационные системы и цифровые технологии» Московского государственного университета технологий и управления имени К.Г. Разумовского (ПКУ); доцент кафедры «Автоматизированные системы управления биотехнологическими процессами» Российского биотехнологического университета (РОСБИОТЕХ), г. Москва, e-mail: i.s.polevshchikov@mail.ru

Уколов Максим Сергеевич – магистрант кафедры «Автоматизированные системы управления биотехнологическими процессами» Российского биотехнологического университета (РОСБИОТЕХ), гр. 22-ИУ-МАГ, г. Москва, e-mail: 0665232@mail.ru

Дмитрук Максим Васильевич – аспирант кафедры «Информационные системы и цифровые технологии» Московского государственного университета технологий и управления имени К.Г. Разумовского (ПКУ), г. Москва, e-mail: avto4max@list.ru

Кравченко Ростислав Александрович – магистрант кафедры «Автоматизированные системы управления биотехнологическими процессами» Российского биотехнологического университета (РОСБИОТЕХ), гр. 23o-090401/БД, г. Москва, e-mail: rostislav.kravchenko.2001@mail.ru

О.А. Полякова, Я.А. Степанова

Пермский национальный исследовательский
политехнический университет, Пермь

АВТОМАТИЗАЦИЯ ЗАДАЧИ УЧЕТА И АНАЛИЗА ИСПОЛЬЗОВАНИЯ БОНУСНОЙ ПРОГРАММЫ ДЛЯ КЛИЕНТОВ ТРАНСПОРТНОЙ КОМПАНИИ

Рассматривается автоматизация процесса учета и анализа использования бонусной программы клиентами транспортной компании. Обнаружено, что текущие возможности автоматизированного рабочего места сотрудника не полностью автоматизируют процесс построения отчетности и расчета маркетинговых показателей по использованию бонусной программы. В результате проведено исследование текущих бизнес-процессов, выявлены их недостатки. На основании исследования спроектированы модели бизнес-процессов с учетом исправленных недостатков. Спроектирована логическая модель базы данных.

Ключевые слова: бонусная программа, бизнес-процесс, автоматизация, отчет, маркетинговый показатель.

O.A. Polyakova, Ya.A. Stepanova

Perm National Research Polytechnic University, Perm

AUTOMATION OF THE TASK OF ACCOUNTING AND ANALYSIS OF THE USE OF THE BONUS PROGRAM FOR CLIENTS OF THE TRANSPORT COMPANY

The article discusses the automation of the process of accounting and analyzing the use of the bonus program by customers of a transport company. It was found that the current capabilities of the employee's automated workplace do not fully automate the process of building reports and calculating marketing indicators for the use of the bonus program. As a result, a study of current business processes was conducted and their shortcomings were identified. Based on the research, business process models are designed taking into account the corrected shortcomings. A logical database model has been designed.

Keywords: bonus program, business process, automation, report, marketing indicator.

Для привлечения новых клиентов и удержания старых, а также создания эмоциональной привязанности к бренду и увеличения прибыли бизнесу необходимо создавать инструменты поощрения клиентов [1]. В транспортной компании таким инструментом является бонусная программа для клиентов. Бонусная программа – механизм начисления

бонусов клиентам транспортной компании за совершение целевых действий и их использования для частичной оплаты товаров и услуг. Услугами в данном случае являются различные виды отправлений посылок (например, отправка письма, доставка посылки курьером), а товарами являются предметы, необходимые при осуществлении услуг (например, конверт или коробка). В зависимости от количества совершенных отправлений в месяц каждому клиенту присваивается уровень участника бонусной программы. Таких уровней всего четыре: бронзовый, серебряный, золотой, платиновый. От уровня бонусной программы зависит процент начисления бонусов за услуги. Для бронзового уровня процентная ставка равна 1 %, для серебряного – 5 %, для золотого – 7 %, для платинового – 10 % от стоимости услуги.

Для того чтобы отследить эффективность работы бонусной программы, сотрудники отдела маркетинга проводят учет и анализ ее использования. В процессе учета формируется отчетность по начислениям и конвертациям бонусов, количеству услуг, на оплату которых использовались бонусы, количеству участников бонусной программы среди всех клиентов, количеству оказанных услуг в конкретном уровне участника бонусной программы и другие. Под анализом понимается расчет маркетинговых метрик и дальнейшие выводы на основании полученных значений. Основные рассчитываемые показатели: жизненная ценность клиента, частота покупок, прирост пользователей – участников бонусной программы за определенный период, процент клиентов – участников бонусной программы от всего количества клиентов компании.

Для выполнения рабочих обязанностей менеджеров отдела маркетинга реализовано автоматизированное рабочее место (АРМ). В АРМ реализована функциональность получения данных о транзакциях клиентов с бонусной программой за указанный период. В транзакциях содержится информация о начислениях и списаниях бонусов, об услуге и сумме бонусов, которые участвуют в транзакции. Для описания бизнес-процессов используется нотация BPMN (Business Process Model and Notation). BPMN сегодня – это лучшая, признанная на международном уровне и активно используемая многими компаниями нотация [2].

Текущий процесс построения отчетности представлен на примере отчета о количествах отправлений клиента на текущем уровне участника бонусной программы на рис. 1.

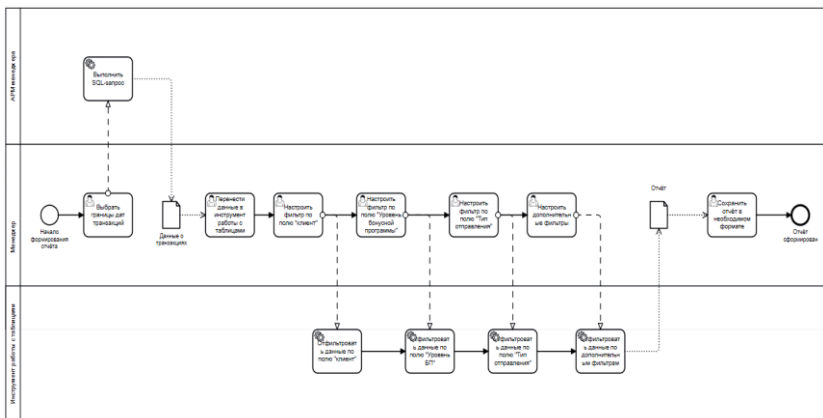


Рис. 1. Текущий процесс построения отчета о количествах отправок клиента на текущем уровне участника бонусной программы

Чтобы сформировать отчет, менеджеру необходимо воспользоваться функциональностью получения данных транзакций. Далее необходимо выгрузить данные в инструмент работы с таблицами Excel. В самом инструменте менеджер настраивает необходимые ему фильтры данных и поля отчета. Для расчета маркетинговых показателей менеджер также использует инструмент работы с таблицами Excel. Текущий процесс расчета показателя на примере частоты покупок представлен на рис. 2.

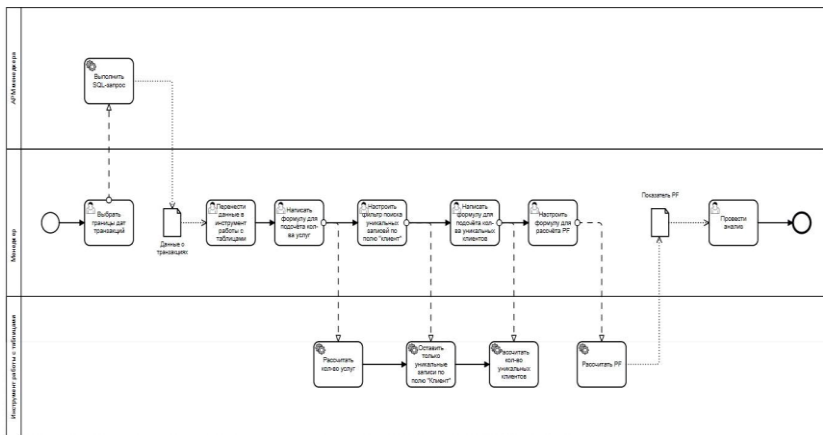


Рис. 2. Текущий процесс расчета показателя частоты покупок

Для расчета частоты покупок менеджеру необходимо не просто настроить фильтрацию данных, но и сформировать формулу. Чтобы рассчитать коэффициент, необходимо количество покупок разделить на количество уникальных клиентов. Для описания формулы в Excel предварительно необходимо разделить клиентов на уникальных и посчитать их количество, а также рассчитать количество покупок в целом. Затем полученные значения необходимо записать в отдельных ячейках.

Очевидно, что текущий процесс анализа и учета использования бонусной программы имеет недостатки, которые замедляют работу сотрудников и не гарантируют отсутствие ошибок в отчетах и расчетах. Среди недостатков стоит отметить следующие: использование стороннего инструмента для работы с таблицами, необходимость передачи файлов между АРМ менеджера и Excel, пересчет данных после каждой настройки фильтров, ручная настройка формул, отсутствие возможности сохранения значений рассчитанных показателей и их дальнейшего просмотра.

Перечисленные недостатки приводят к ухудшению качества работы сотрудников, так как затрачивается большое количество времени на передачу данных и настройку отчетности. Также при таком процессе более вероятно появление ошибок, поскольку ручная настройка фильтров и формул не защищена от человеческого фактора.

На основании анализа недостатков спроектированы процессы построения отчетности (рис. 3) и расчета маркетинговых показателей (рис. 4).

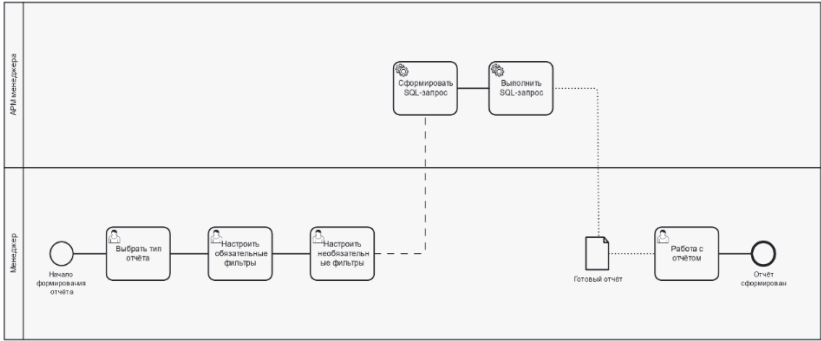


Рис. 3. Диаграмма процесса построения отчетов по использованию бонусной программы

Главной отличительной особенностью спроектированных процессов является отсутствие третьей стороны. Теперь построение отчетности и расчет маркетинговых показателей по использованию бонусной

программы клиентами происходит в АРМ менеджера. Менеджеру больше не нужно выгружать данные в Excel и самостоятельно настраивать фильтры и формулы. Расчет маркетинговых показателей производится автоматически, а полученные значения можно сохранять, а затем просматривать.

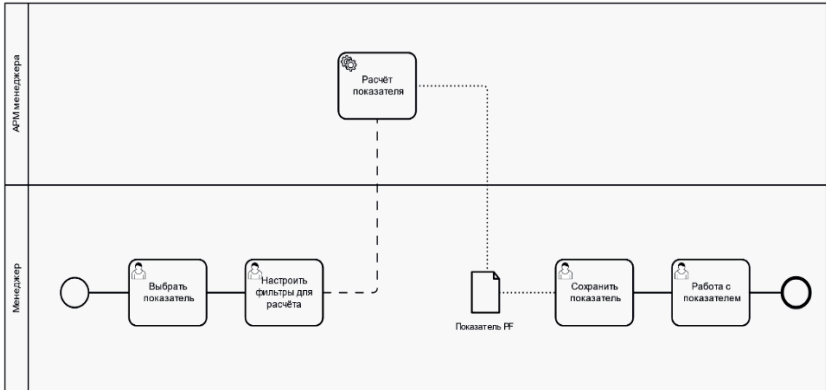


Рис. 4. Диаграмма процесса расчета маркетинговых показателей по использованию бонусной программы

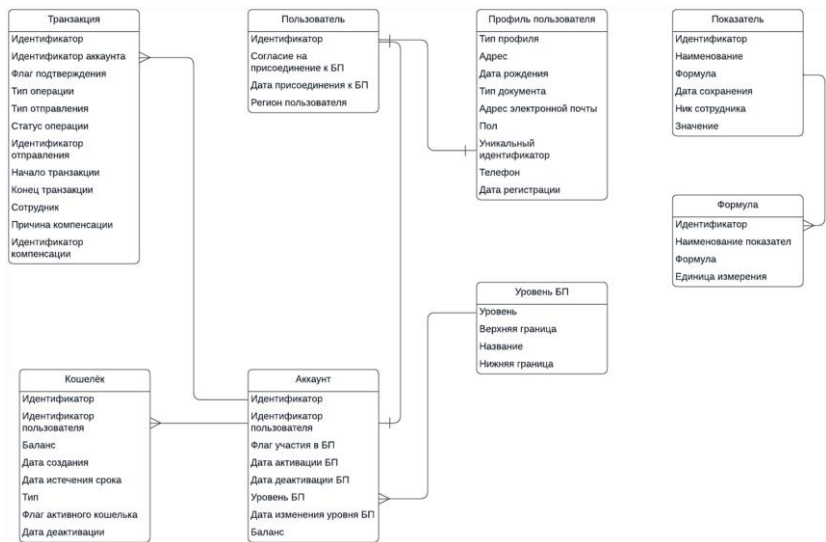


Рис. 5. Логическая модель базы данных

Для разработки программного обеспечения спроектирована логическая модель базы данных в виде ER-диаграммы нотации Джеймса Мартина, представленная на рис. 5 [3].

Основными сущностями для бизнес-процесса построения отчетности по использованию бонусной программы клиентами являются: транзакция, пользователь, профиль пользователя, аккаунт, уровень бонусной программы, кошелек. Для бизнес-процесса расчета маркетинговых показателей основными сущностями являются показатель и формула, при этом для их расчета также используются вышеуказанные сущности логической модели.

Таким образом, в результате исследования выявлены недостатки процесса анализа и учета использования бонусной программы для клиентов транспортной компании и пути их устранения. Разработано программное обеспечение для автоматизации построения отчетности и расчета маркетинговых показателей.

Библиографический список

1. Пепперс Д., Роджерс М. Управление взаимоотношениями с клиентами [Электронный ресурс]; пер. с англ. Д.Л. Раевской, С.Н. Живаевой. – 2006. – URL: https://www.mann-ivanoverber.ru/assets/files/bookparts/017/otnosheniya_read.pdf (дата обращения: 15.05.2024).
2. Репин В. Моделирование бизнес-процессов в нотации BPMN: пособие для начинающих. Ч. I. – [б. м.]: Издательские решения, 2019. – 84 с.
3. Советов Б.Я., Цехановский В.В., Чертовский В.Д. Базы данных: теория и практика: учебник для бакалавров. – 2-е изд. – М.: Юрайт, 2014. – 463 с.

Сведения об авторах

Полякова Ольга Андреевна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: olgastratum@mail.ru

Степанова Яна Андреевна – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: yanastepanova54@gmail.com

Г.А. Чураков

Пермский филиал Национального исследовательского
университета «Высшая школа экономики», Пермь

ЧАСТЕРЕЧНАЯ РАЗМЕТКА ДЛЯ ВЫДЕЛЕНИЯ СКЕЛЕТНОЙ СТРУКТУРЫ ПРЕДЛОЖЕНИЙ

В статье описан процесс разработки модели для нанесения частеречной аннотации на текст с использованием переноса обучения BERT. Процесс подготовки данных и оценка полученных результатов. Выявлено, что предложенный способ позволяет достичь хороших результатов в нанесении разметки на текст.

Ключевые слова: частеречная разметка, морфологический анализ, обработка естественного языка.

G.A. Churakov

Perm Branch of the National Research University
Higher School of Economics, Perm

POS-TAGGING TO HIGHLIGHT THE SKELETAL STRUCTURE OF SENTENCES

The article describes the process of developing a model for applying partial annotation to text using BERT learning transfer. Process of data preparation and evaluation of obtained results. It has been found that the proposed method makes it possible to achieve good results in marking text.

Keywords: part of speech tagging, morphological analysis, natural language processing.

Одним из подвидов задачи по извлечению структурированной информации из текста является задача по частеречной разметке (Part-of-Speech Tagging), процесс аннотирования слова в тексте (корпусе) определенной части речи в зависимости от ее определения и контекста.

Разметка частей речи представляет собой более сложную задачу, чем просто составление списка слов с соответствующими частями речи. Это связано с тем, что некоторые слова могут включать в себя несколько частей речи в зависимости от контекста, а некоторые части речи сложны по своей природе. Это нередкое явление в естественных языках, где значительная часть словоформ, в отличие от многих искусственных языков, неоднозначна.

Данная задача решалась различными методами, первые подходы к решению данного вопроса основывались на составлении правил вручную. Это требовало обширных знаний грамматики конкретного языка, что делало подобную систему аннотирования трудно переносимой на другие языки. В данной статье будет рассмотрен подход с использованием нейросетевой модели, использующей архитектуру BERT в качестве основы.

Объектом исследования служат синтаксические и грамматические признаки, по которым данные лексемы можно распознать в тексте.

Предметом исследования является возможность применения алгоритма автоматического аннотирования текстовых данных на русском языке частеречной разметкой.

Цель – разработать модель, способную с хорошей точностью выделять из текста на русском языке скелетную частеречевую структуру.

Материалом исследования является корпус из 100 предложений на русском языке, размеченных экспертом.

Методы исследования: интуитивное выделение необходимых признаков для определения отнесенности слова части речи – представление слов в закодированном виде, моделирование синтаксического уровня при представлении предложения в виде закодированной последовательности, аугментация текстовых данных.

Результатом работы является модель, позволяющая распознавать в тексте части речи. Практическая значимость заключается в том, что тегирование текста частями речи является важным компонентом на этапе подготовки данных. Отсутствие этой лингвистической аннотации затрудняет дальнейший анализ текста, в первую очередь из-за неоднозначности значений слов. Конкретно эта модель предназначена для аннотирования текста базовыми тегами для улучшения качества машинного перевода с применением скелетных структур [1–23].

Метрики оценки качества разметки. Сформулировав задачу POS-тэггинга как многоклассовую классификацию, использовали соответствующие метрики. Учитывая также природу языка, следует заметить дисбаланс классов в выборке, в этих условиях в качестве основной метрики была выбрана взвешенная F-1 мера. Данная метрика в задаче многоклассовой классификации вычисляется с точки зрения «один против всех», когда к True Positive относятся все классифицированные

токены, а объекты других классов относятся к False Positive. Формула ниже описывает вычисление F1 меры для отдельно взятого класса.

$$F-1 \text{ Score}(\text{class}=a) = \frac{2TP(\text{class}=a)}{2TP(\text{class}=a) + FP(\text{class}=a) + FN(\text{class}=a)}.$$

В задаче классификации с несколькими классами используется взвешенная сумма оценок для каждого из классов.

Обработка данных. Корпус размеченного текста представлял собой 100 предложений, он был предварительно предобработан, данные были трансформированы в последовательности типа: предложение – набор тэгов для каждого слова в предложении. Такого объема данных недостаточно для обучения модели, поэтому были предприняты шаги по увеличению выборки [7]. Предложения тестовой выборки были нарезаны на фрагменты при помощи скользящего окна размером [1; количество слов в предложении]. Таким образом, количество наблюдений увеличило тренировочную выборку в до 20 000 наблюдений.

Архитектура модели. Для построения модели данные недостаточно разнообразны, поэтому было решено произвести перенос обучения. В качестве базовой модели решения выбрана RuBERT-base [9], содержащая в себе энкодер блоки сети трансформер [3–5]. Среди особенностей модели хочется выделить ее устройство:

- предобученный токенизатор – BPE [1].
- модель обучена на задаче маскированного языкового моделирования, в процессе обучения модель извлекает из последовательности фиксированной длины признаки, необходимые для предсказания скрытого токена, расположенного на случайной позиции в последовательности (рис. 1).

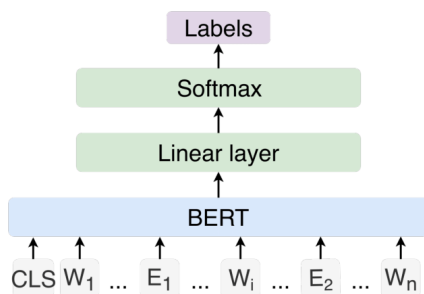


Рис. 1. Картинка из оригинальной статьи [17]: дообучение BERT для задачи Token Classification

Обучение модели. Как описано в оригинальной статье, BERT можно легко дообучить под решение узкой задачи, для этого добавим к предобученной русскоязычной модели полносвязный слой и определим функцию активации Softmax, на выходе модели получим закодированные значения классов токенов, которые затем раскодируем во время постобработки. Далее обучим модель на задачу Token Classification с учителем, будем оценивать качество после каждой эпохи и при помощи оптимизатора Adam корректировать веса модели методом обратного распространения ошибки. Процесс обучения модели можно наблюдать на рис. 2.

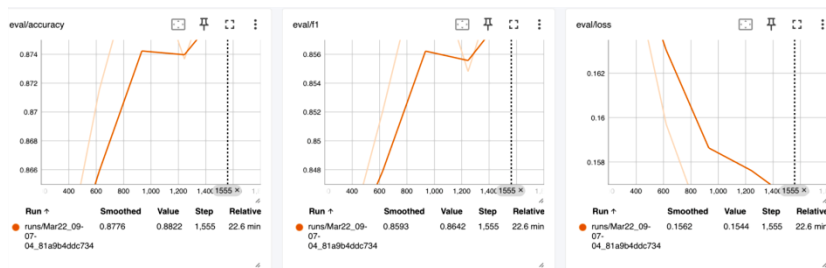


Рис. 2. Процесс обучения модели

На валидации были получены следующие метрики для модели:

- F1: 0.8642,
- Accuracy: 0.8822.

Заключение. Цель данной работы – разработка модели, способной с хорошей точностью выделять из текста на русском языке скелетную частеречевую структуру. Важно признать, что обучение и оценка модели проводились на относительно ограниченном наборе данных, поэтому результаты могут отличаться в зависимости от набора данных.

Данную работу можно продолжать за счет увеличения объема тренировочных данных (мультиязычность, более сбалансированная выборка) и подбора гиперпараметров нейронной сети. В ходе анализа результатов работы модели также было выявлено, что модель способна к разметке на другом языке, корректно выделяются части речи, которых встретилось сравнительно много в наборе данных для обучения.

С результатами работы можно ознакомиться на следующих ресурсах:

- Github: <https://github.com/disk0Dancer/rubert-finetuned-pos.git>;
- Hugging Face Demo: <https://huggingface.co/disk0dancer/ruBert-base-finetuned-pos>.

Библиографический список

1. Sennrich R., Haddow B., Birch A. Neural Machine Translation of Rare Words with Subword Units. – 2016.
2. Enriching Word Vectors with Subword Information / P. Bojanowski [и др.]. – 2017.
3. Attention Is All You Need / A. Vaswani [и др.]. – 2023.
4. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / J. Devlin [и др.]. – 2019.
5. Tenney I., Das D., Pavlick E. BERT Rediscovered the Classical NLP Pipeline. – 2019.
6. Wang C., Cho K., Gu J. Neural Machine Translation with Byte-Level Subwords. – 2019.
7. Augmented SBERT: Data Augmentation Method for Improving Bi-Encoders for Pairwise Sentence Scoring Tasks / N. Thakur [и др.]. – 2021.
8. Hui W. Performance of Transfer Learning Model vs. Traditional Neural Network in Low System Resource Environment. – 2020.
9. A Family of Pretrained Transformer Language Models for Russian / D. Zmitrovich [и др.]. – 2023.
10. Trofimov I.V. Automatic Morphological Analysis for Russian: Application-Oriented Survey // *Programmnaya ingeneria*. – 2019. – Vol. 10, № 9–10. – P. 391–399.
11. Hiroki Nakayama Sequeval: A Python framework for sequence labeling evaluation. – URL: <https://github.com/chakki-works/sequeval>
12. Liao W., Veeramachaneni S. A simple semi-supervised algorithm for named entity recognition Boulder, Colorado: Association for Computational Linguistics. – 2009. – P. 58–65.
13. Mylnikova A.V., Trusov V.A., Mylnikov L.A. Use of Text Skeleton Structures for the Development of Semantic Search Methods // *Automatic Documentation and Mathematical Linguistics*. – 2023. – № 5 (57). – P. 301–307.
14. Novikova A. Direct Machine Translation and Formalization Issues of Language Structures and Their Matches by Automated Machine Translation for the Russian-English Language Pair. – 2018.
15. Pennington J., Socher R., Manning C. Glove: Global Vectors for Word Representation. – Doha, Qatar: Association for Computational Linguistics, 2014. – P. 1532–1543.

16. Sapin A.S. Building neural network models for morphological and morpheme analysis of texts // Proceedings of the Institute for System Programming of the RAS. – 2021. – № 4 (33). – P. 117–130.
17. Relation Extraction from Clinical Narratives Using Pre-trained Language Models / Q. Wei [и др.] // AMIA ... Annual Symposium proceedings. AMIA Symposium. – 2019. – P. 1236–1245.
18. Zhang Y., Jin R., Zhou Z.-H. Understanding bag-of-words model: a statistical framework // International Journal of Machine Learning and Cybernetics. – 2010. – № 1-4 (1). – P. 43–52.
19. Intellectual analysis of data on the basis of Stanford CoreNLP for POS tagging of texts in the russian language // Systems and Means of Informatics. – 2018.
20. Classical ML Equations in LaTeX. – URL: https://blmoista-winde.github.io/ml_equations_latex/
21. MLM. – URL: https://www.sbert.net/examples/unsupervised_learning/MLM/README.html
22. Summary of the tokenizers. – URL: https://huggingface.co/docs/transformers/tokenizer_summary
23. What-is-word2vec-and-how-does-it-work. – URL: <https://swimm.io/learn/large-language-models/what-is-word2vec-and-how-does-it-work>

Сведения об авторе

Чураков Григорий Александрович – бакалавр образовательной программы «Программная инженерия» факультета социально-экономических и компьютерных наук Пермского филиала Национального исследовательского университета «Высшая школа экономики», г. Пермь, e-mail: gachurakov@edu.hse.ru

Е.В. Корепанова, Д.С. Курушин, А.А. Петренко

Пермский национальный исследовательский
политехнический университет, Пермь

ПРОГРАММА НАХОЖДЕНИЯ ОРИЕНТАЦИИ ИНСТРУМЕНТА ТРЕХМЕРНОЙ ПЕЧАТИ ИЗ РАЗНЫХ ПЛОСКОСТЕЙ В ТЕКУЩИЙ МОМЕНТ ВРЕМЕНИ

В статье рассматривается программа, позволяющая находить координаты ориентации инструмента трехмерной печати методом послойного наплавления. Рассмотрен робот A12-1450, на котором будет осуществляться непланарная печать. Приведен алгоритм работы программы и этапы ее работы. В статье также представлена технология реализации программы. Приведен пример работы программы и визуализация результатов.

Ключевые слова: 3D-печать, послойное наплавление, FDM, аддитивное производство.

E.V. Korepanova, D.S. Kurushin, A.A. Petrenko

Perm National Research Polytechnic University, Perm

A PROGRAM FOR FINDING THE ORIENTATION OF A THREE- DIMENSIONAL PRINTING TOOL AT THE CURRENT TIME

The article discusses a program that allows you to find the orientation coordinates of a three-dimensional printing tool by layer-by-layer deposition. The robot A12-1450, which will be used for non-planar printing, is considered. The algorithm of the program and the stages of its operation are given. The article also presents the technology of the program implementation. An example of the program's operation and visualization of the results is given.

Keywords: 3D printing, layer-by-layer surfacing, FDM, additive manufacturing.

В наше время все чаще производители предпочитают технологии трехмерной печати вместо традиционных способов производства [1, 2]. В основе 3D-печати лежат аддитивные технологии, при которых трехмерный объект образуется последовательным наложением различных материалов или изменением фазового состояния вещества в обозначенной области пространства. При послойном наплавлении материала остается мало отходов в отличие от субтрактивного производства [3].

Реализация программы нахождения ориентации инструмента трехмерной печати из разных плоскостей. В данной программе

находится ориентация инструмента при трехмерной печати из разных плоскостей. Ориентациями инструмента во время печати служат множество координат нормальных векторов плоскостей. Нормальные векторы плоскостей являются перпендикулярами к плоскостям, которые формирует программа между двумя точками одного слоя и ближайшей точкой следующего слоя.

При реализации непланарной печати инструментом выступает робот A12-1450 версии RCS 3.0, состоящий из шести соединенных друг с другом осей [4] (рис. 1). Данный робот способен выполнять печать по разным осям, он должен будет ориентироваться по нормальному вектору плоскости. Сама печатающая головка должна будет изменять свой угол наклона в каждый момент времени, индивидуально подстраиваясь для печати каждой плоскости.

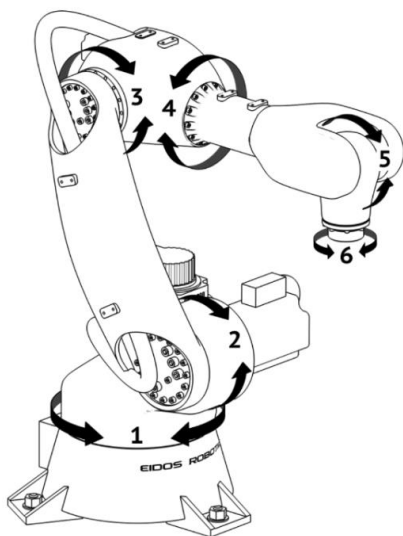


Рис. 1. Основные оси A12-1450

Программа разработана под метод послойного наплавления, поэтому она должна формировать из входного трехмерного объекта новые межслойные плоскости, к которым будет найден нормальный вектор, являющийся ориентацией инструмента [5]. Печать из разных плоскостей, в свою очередь, поможет решить такую проблему, как затрата излишнего материала на подпорки моделей со сложной структурой,

следовательно, сократится затрачиваемое на печать время, также решится проблема стыков слоев, появится более гладкая поверхность готового объекта.

Основная идея работы алгоритма программы заключается в создании такой программы, где от пользователя зависят только входные данные и куда будут сохраняться выходные. Внутри программа сама выводит нужные данные из файла и обрабатывает их.

При запуске программы пользователю необходимо указать путь к файлу GCODE с исходной трехмерной моделью. Затем программа считывает координаты из файла послойно в массивы данных. В каждом слое будут взяты две соседние координаты и найдена средняя координата между ними. К полученной средней координате программа находит ближайшую координату из следующего слоя, тем самым будет сформирована плоскость. К сформированным плоскостям программа находит нормальные векторы. Нормальный вектор плоскости представляет собой координаты, на которые будет ориентироваться инструмент трехмерной печати, когда будет проходить между двумя точками плоскости, находящимися в одном слое. После этого программа записывает нужную информацию в выбранный ранее файл и конвертирует его в формат .obj.

Представим работоспособность программы на примере, где входным файлом .gcode является описание трехмерной модели границ куба.

На рис. 2 представлены сформированные между слоями плоскости, представленные с помощью Blender после запуска в нем файла OBJ.

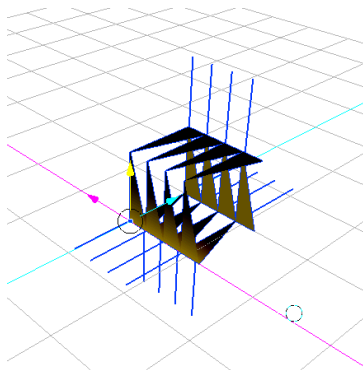


Рис. 2. Сформированные плоскости

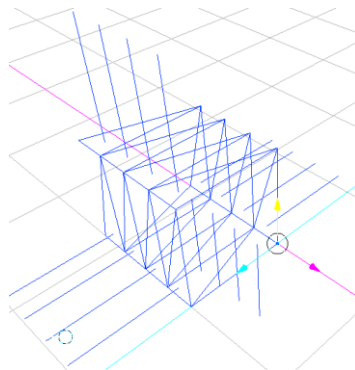


Рис. 3. Полученные нормальные векторы плоскостей

На рис. 3 и 4 можно увидеть перпендикуляры к сформированным плоскостям, которые являются нормальными векторами. На координаты представленных перпендикуляров будет ориентировать инструмент при трехмерной печати.

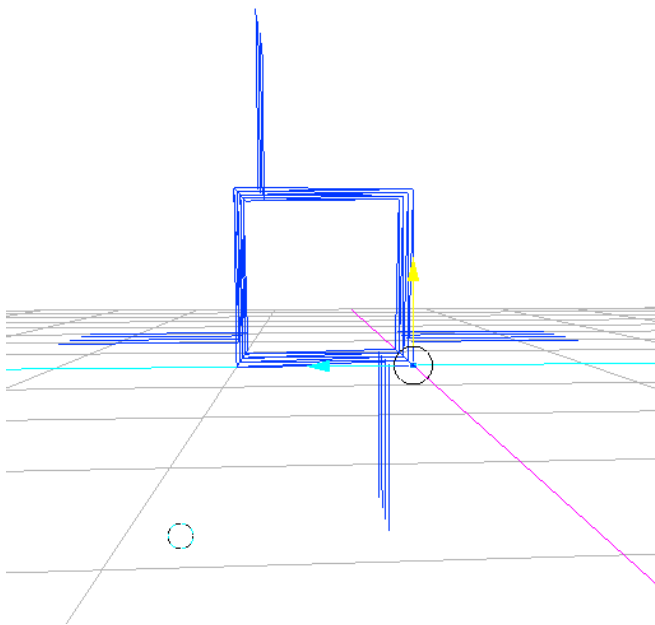


Рис. 4. Перпендикулярность нормальных векторов плоскостей

Таким образом, нами были получены координаты нормальных векторов плоскостей, на которые в дальнейшем будет ориентироваться робот A12-1450 при трехмерной печати.

Заключение. В результате была разработана программа, совершающая разбиение слоев трехмерного объекта на плоскости, к которым находится нормальный вектор. С помощью координат данного вектора будет совершаться ориентация инструмента трехмерной печати в текущий момент времени.

Печать, которая будет производиться в зависимости от положения инструмента при печати в разных плоскостях, является более точной. Ей не будут нужны дополнительные опоры, и качество напечатанных моделей будет более приближено к форме моделей в 3D-редакторах, следовательно, точность печати будет более высокой.

Библиографический список

1. Добротов А.В., Авдеев А.Р., Швец А.А. Анализ способов повышения производительности устройств объемной печати для изготовления сборочных изделий // Сборка в машиностроении, приборостроении. – 2015. – № 11. – С. 3–6.
2. FDM Technology. About Fused Deposition Modeling [Электронный ресурс] // Stratasys Ltd. – URL: <http://www.stratasys.com/3d-printers/technologies/fdm-technology> (дата обращения: 10.05.2024).
3. Stansbury J.W., Idacavage M.J. 3D Printing with Polymers: Challenges among Expanding Options and Opportunities // Dent. Mater. – 2016. – № 32. – Р. 54–64.
4. ООО «Эйдос-робототехника». Руководство RCS 3.0. – 2016–2021. – С. 19.
5. Принципы создания программы работы устройства объемной печати по трехмерной модели / И.А. Гущин, А.Р. Авдеев, А.А. Швец, А.В. Дроботов // Известия ВолгГТУ. Сер. Прогрессивные технологии в машиностроении. – 2015. – № 11 (173). – С. 50–53.

Сведения об авторах

Корепанова Елизавета Владимировна – магистрант «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, гр. АСУ4-22-1м, г. Пермь, e-mail: corepanowa.elizaveta@yandex.ru

Курушин Даниил Сергеевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: daniel@kurushin-perm.ru

Петренко Александр Анатольевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: hawk321@mail.ru

М.В. Вычугжанин

Пензенский государственный университет, Пенза

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБЛАСТИ МОБИЛЬНОГО ЗДРАВООХРАНЕНИЯ

Данная статья посвящена вопросу применения информационных технологий в области мобильного здравоохранения. Рассмотрен вопрос получения различных данных о пациенте с помощью соответствующих биомедицинских датчиков. Показана актуальность разработки информационных систем в области мобильного здравоохранения для изучения, обобщения и анализа биомедицинских данных.

Ключевые слова: информационные технологии, мобильное здравоохранение, биомедицинские данные, электрокардиограмма, акселерометр.

M.V. Vychugzhanin

Penza State University, Penza

INFORMATION TECHNOLOGIES IN THE FIELD OF MOBILE HEALTHCARE

This article is devoted to the application of information technologies in the field of mobile healthcare. The issue of obtaining various patient data using appropriate biomedical sensors is considered. The relevance of the development of information systems in the field of mobile healthcare for the study, generalization and analysis of biomedical data is shown.

Keywords: information technology, mobile healthcare, biomedical data, electrocardiogram, accelerometer.

В настоящее время происходит активное применение информационных технологий во многих видах деятельности человека, и, благодаря их широкому внедрению в медицинскую сферу, одной из динамически развивающихся областей стало здравоохранение. Внедрение в жизнь инноваций происходит в медицинской сфере повсеместно. В их числе – электронные паспорта здоровья, система поддержки принятия врачебных решений, телемедицина, системы контроля качества медицинских услуг и безопасности лекарственных средств, цифровая медицина, мобильное здравоохранение [1].

Значительное распространение и качественное применение находит система больших данных (BigData). Используя современные технические средства в медицине, происходит сбор и накопление большого количества диагностических данных. При их последующем анализе становится возможным определение не только текущего состояния здоровья пациента, но и прогнозирование развития заболеваний. Устойчивое развитие биомедицинских технологий открывает значительные перспективы для положительного воздействия на здоровье человека, что подтверждается успехами в лечении заболеваний, достигнутыми за последнее время.

Одним из направлений активного развития информационных технологий в медицинской сфере является область мобильного здравоохранения. С одной стороны, она позволяет рационально использовать средства стационарного лечения медучреждения без ущерба качеству оказываемых медицинских услуг, так как позволяет проведение медицинских обследований пациентов при обычных условиях их жизни (вне кабинета врача). С другой стороны, в случае регулярного мониторинга состояния здоровья способствует профилактике и выявлению заболеваний на ранних стадиях.

Мобильное здравоохранение направлено на улучшение здоровья и благополучия человека путем непрерывного мониторинга физиологического состояния с помощью мобильных устройств и сенсоров. В настоящее время существенно повысилась доступность как специализированных медицинских приборов для самостоятельного мониторинга параметров здоровья (артериального давления, пульса, насыщения крови кислородом и других), так и «универсальных» носимых устройств с биомедицинскими датчиками, получивших широкое распространение, таких как фитнес-трекеры, «умные» часы, смартфоны с установленным специальным программным обеспечением и другие.

В качестве примера использования специализированного медицинского прибора можно привести широко распространенный метод суточного (холтеровского) мониторингирования электрокардиограммы человека [2]. Портативное устройство которое осуществляет регистрацию электрокардиограммы в течение заранее определенного периода времени (как правило, в течение 24 ч), что позволяет получить данные о работе сердца пациента в его повседневных условиях.

С помощью автоматического тонометра в бытовых условиях пациент самостоятельно производит определение артериального давления, пульса. Современные приборы, кроме непосредственно измерения требуемых показателей, способны также хранить в памяти предыдущие измерения, указывать индикатором на аритмию, рассчитывать среднее значение артериального давления и т.д. Пульсоксиметр используется для неинвазивного измерения уровня насыщения кислородом капиллярной крови (пульсоксиметрии).

Если специализированные медицинские приборы используются непосредственно по своему назначению и, как правило, применяются уже при лечении пациента, то особый интерес представляют «универсальные» носимые устройства с биомедицинскими датчиками для профилактики и предупреждения заболеваний.

Смартфон с установленным на него специализированным программным обеспечением и подключенными биомедицинскими датчиками превращается в диагностический прибор, регистрирующий биологические сигналы. Фитнес-трекеры, «умные» часы и браслеты в настоящее время очень популярны и широко распространены, а в них, как правило, встроены акселерометры.

Носимые устройства со встроенными акселерометрами могут крепиться на разных частях тела человека, но чаще всего располагаются на запястьях. Они подходят для самостоятельного отслеживания физической активности и могут быть полезными для контроля процесса восстановления пациентов в домашних условиях после выписки из больницы при изучении болезней, связанных с нарушением подвижности, включая нервно-мышечные расстройства [3].

В «умных» устройствах акселерометр используется для подсчета количества шагов, но полезная информация от анализа исходных данных этим не ограничивается. Интерес может представлять в случае сопряжения работы датчика электрокардиограммы со смартфоном и получение синхронизированного сигнала. Анализ объединенных данных электрокардиограммы и акселерометра может помочь определить, как физическая активность влияет на работу сердца. Это может быть полезно для разработки рекомендаций для людей с различными заболеваниями сердца.

В наше время становятся все более популярными цифровые устройства в сфере мобильного здравоохранения, разработанные для

мониторинга работы сердца. К ним относятся, например, портативные измерители артериального давления и электрокардиографы [4].

Следующим шагом в развитии мобильного здравоохранения является его потенциал для поддержки развития персонализированной медицины, позволяющей адаптировать медицинские решения и практики к индивидуальным потребностям пациента и смягчать последствия применения универсальных решений. В результате совместный анализ биомедицинских данных поможет глубже понять процессы, происходящие в организме человека, сделать обоснованные выводы на основе полученных результатов и разработать наиболее эффективные методы профилактики, диагностики и лечения заболеваний.

Библиографический список

1. Mobile ECG Monitoring Device with Bioimpedance Measurement and Analysis / M. Safronov, A. Kuzmin, O. Bodin [et al.] // Conference of Open Innovations Association, FRUCT. – 2019. – № 24. – P. 375–380. DOI: 10.23919/FRUCT.2019.8711944

2. Особенности обработки электрокардиосигналов в системах мобильного мониторинга / В.И. Волчихин, М.А. Митрохин, А.В. Кузьмин, М.И. Сафронов, О.Н. Бодин, Н.Ю. Митрохина, А.Ю. Тычков // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2018. – № 1 (45). – С. 54–63.

3. Формирование базы данных сигналов акселерометра для мониторинга физической активности человека / Д.И. Прусс, Е.П. Рябова, М.И. Сафронов, А.В. Кузьмин // Перспективные информационные технологии (ПИТ 2022). – 2022. – С. 279–283.

4. Способ и аппаратно-программные средства анализа биоимпеданса для систем мобильного мониторинга ЭКГ // Модели, системы, сети в экономике, технике, природе и обществе. – 2020. – № 3 (35). – С. 118–128. DOI: 10.21685/2227-8486-2020-3-10

Сведения об авторе

Вычугжанин Михаил Владимирович – аспирант кафедры «Информационно-вычислительные системы» Пензенского государственного университета, гр. 23а-2.3.1, г. Пенза, e-mail: mikeabf@rambler.ru

С.И. Советов, С.Ф. Тюрин

Пермский национальный исследовательский
политехнический университет, Пермь

**АЛГОРИТМ СОЕДИНЕНИЯ БЛОКОВ
ФУНКЦИИ-ДЕШИФРАЦИИ ДЛЯ СИНТЕЗА
ЛОГИЧЕСКОГО ЭЛЕМЕНТА LUT
С ДЕШИФРАЦИЕЙ ВХОДНЫХ ПЕРЕМЕННЫХ**

В статье рассматривается алгоритм синтеза логического элемента LUT FPGA, который выполняет вычисление основной функции и дешифрацию набора переменных одновременно. Представлен общий вид модели предлагаемого логического элемента. Описываются соединения блоков функции-дешифрации в базовом элементе 1-LUT, обеспечивающих дешифрацию входного набора с ортогональностью сигнала в обратном дереве передающих транзисторов. Приведены примеры применения алгоритма для синтеза логического элемента на три и четыре переменные.

Ключевые слова: ПЛИС, LUT, передающие транзисторы, таблица истинности, логическая функция, дешифрация.

S.I. Sovetov, S.F. Tyurin

Perm National Research Polytechnic University, Perm

**ALGORITHM CONNECTING FUNCTION-DECODING BLOCKS
FOR LUT SYNTHESIS WITH DECODING INPUT VARIABLES**

The article discusses an algorithm for synthesizing a LUT FPGA logic element, which calculates the main function and decoding a set of variables simultaneously. A general view of the model of the proposed logical element is presented. The connections of decoding function blocks in the basic 1-LUT element are described, providing decoding of the input set with signal orthogonality in the inverse tree of transmitting transistors. Examples of application of the algorithm for the synthesis of a logical element into three and four variables are given.

Keywords: FPGA, LUT, transmitting transistors, truth table, logic function, decoding.

Программируемые логические интегральные схемы (ПЛИС) в настоящее время получили широкое распространение при использовании в высоконагруженных вычислительных системах. Перепрограммируемая логика позволяет в реальном времени менять оптимальную конфигурацию для вычисления необходимых функций. Для программирования

логических функций используется таблица поиска LUT (Look Up Table), которая вычисляет одну функцию для заданной конфигурации. В известных моделях LUT для дешифрации входного набора требуется 2^n LUT, однако предложенная модель позволяет выполнить дешифрацию входного набора переменных с одновременным вычислением логических функций, при использовании дополнительных блоков функций-дешифрации в базовом LUT на одну переменную. Целью исследования является разработка алгоритма соединения блоков функции-дешифрации для синтеза логического элемента LUT. В результате исследования реализуется алгоритм, позволяющий синтезировать LUT с одновременным вычислением основной функции и дешифрацией набора, ограниченный количеством переменных от одной до десяти [1–7] .

Модель логического элемента LUT с дешифрацией входных переменных. Для реализации дешифрации входного набора вводятся дополнительные блоки переменных, обеспечивающие подключение уровня нуля в случае неактивирования соответствующей цепочки к инвертору выхода дешифрации. В этом случае используется модель многоуровневого дерева.

Так, для $n = 1$ получим:

$$\begin{aligned} \frac{(V_{cc})\bar{x}}{(Ground)x}(\vee\bullet)(NOT) &= z_0(x), \\ \frac{(V_{cc})x}{(Ground)\bar{x}}(\vee\bullet)(NOT) &= z_1(x) \\ \frac{d_1(NOT)x}{d_0(NOT)\bar{x}}(\vee\bullet)(NOT) &= z(d_0d_1x) \end{aligned} \quad (1)$$

В предлагаемом выражении (1) имеется три составляющих. Первая часть, вычисляющая значение логической функции одной переменной:

$$\left. \frac{d_1(NOT)x}{d_0(NOT)\bar{x}} \right\} (\vee\bullet)(NOT) = z(d_0d_1x), \quad (2)$$

которая может быть использована как строительный блок, называемый мультиплексором 2-1, при увеличении разрядности в следующем виде:

$$\left. \frac{x}{\bar{x}} \right\} (\vee\bullet). \quad (3)$$

Две другие, независимые, части выполняют одновременно с вычислением функции еще и дешифрацию переменной с обеспечением ортогональности сигнала на входе инверторов. Это означает, что всегда будет либо уровень логического нуля, либо уровень логической единицы, без обрыва цепи, что является одним из важнейших схемотехнических принципов:

$$\begin{aligned} \frac{(V_{cc})\bar{x}}{(Ground)x}(\vee\bullet)(NOT) &= z_0(x), \\ \frac{(V_{cc})x}{(Ground)\bar{x}}(\vee\bullet)(NOT) &= z_1(x). \end{aligned} \quad (4)$$

То есть, например, если $x = 0$, то активизируется путь

$$(V_{cc})\bar{x}(\vee\bullet)(NOT) = z_0(x) = 0. \quad (5)$$

Происходит отрицание константы «1» (V_{cc}) уровня шины питания. В то же время по другой ветви:

$$(Ground)x(\vee\bullet)(NOT) = z_1(x) = 1. \quad (6)$$

Происходит отрицание константы «0», т.е. шины «Ноль Вольт» ($Ground$).

Использование (1) позволяет строить элементы для произвольного количества переменных с учетом ограничений по количеству последовательно соединенных передающих транзисторов. Т.е. предлагается суперпозиция 1-LUT по выражению (1) для синтеза LUT, вычисляющих значение заданной логической функции и дешифрацию набора переменных. Для этого был реализован следующий алгоритм.

Алгоритм подключения блоков функции-дешифрации к мультиплексорам логического элемента. Реализованный алгоритм обеспечивает синтез логического элемента LUT для определенного количества переменных, вычисляющий базовую логическую функцию и соединение блоков функции-дешифрации для одновременного выполнения дешифрации входного набора.

Входными параметрами алгоритма являются: количество переменных n логического элемента LUT.

Общая схема алгоритма приведена на рис. 1.

Первым этапом производится синтез элемента 1-LUT с подключением блока функции-дешифрации, состоящего из передающих транзисторов. 1-LUT состоит из: входа переменной $X_{n,p}$, где p – индекс мультиплексора, определяющий подключение дешифрации; входа дешифрации

$P_{n,i,2p}$, принимающий сигнал с предыдущих LUT; двух входов сигналов набора переменной или результатов вычисления предыдущих мультиплексоров; выхода (F_n) результата мультиплексирования; выходов дешифрации $P_{n-1,o,2p}$ $P_{n-1,o,2p+1}$, для подключения сигналов к предыдущим мультиплексорам. Дополнительно вводятся: два транзистора, стоки которых объединены и подключены к порту ввода мультиплексора $P_{n,i,2p}$ и истоки объединены и подключены к выходу мультиплексора $P_{n-1,o,2p}$, а затворы подключены к сигналам переменной – инвертированному и не инвертированному; два транзистора стоки которых объединены и подключены к нулю вольт (*Ground*), и истоки объединены и подключены к выходу мультиплексора $P_{n-1,o,2p+1}$, а затворы подключены к различным сигналам переменной – инвертированному и не инвертированному [8, 9].

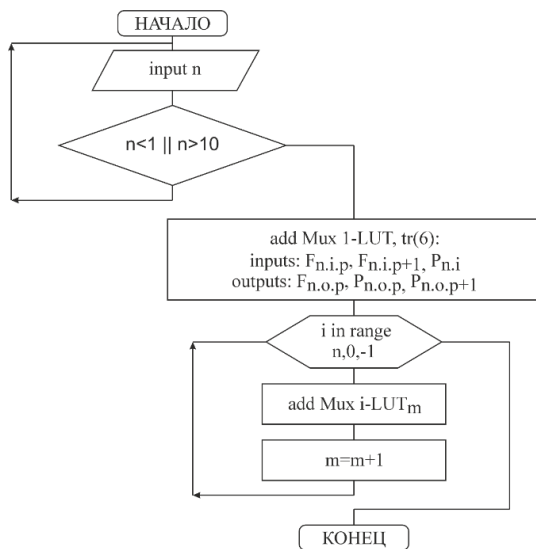


Рис. 1. Алгоритм соединения блоков функции-дешифрации и синтеза логического элемента LUT с дешифрацией входных переменных

Согласно количеству введенных переменных n синтезируется $2^n - 1$ мультиплексоров 1-LUT с дешифрацией. Начиная со старшей переменной X_n вводится первый ($m = 1$) мультиплексор 1-LUT, вход $P_{n,i,0}$ которого подключен к высокому уровню напряжения (VCC). Выход мультиплексора F_n является результатом вычисления логической функции. Вводятся мультиплексоры для предыдущей переменной X_{n-1} ($m = 2$)

в количестве $2^m - 1$, где m – порядковый номер переменной от старшей к младшей. Осуществляется подключение выходов дешифрации старшего мультиплексора $P_{n-1.o.2p}$ к входу мультиплексора $P_{n.i.2p}$ и $P_{n-1.o.2p+1}$ к входу мультиплексора $P_{n.i.2p+1}$. Выходы результатов мультиплексирования переменной X_{n-1} подключаются к входам вычисления мультиплексирования старшей переменной X_n . Синтез продолжается до достижения первой переменной X_1 , входы мультиплексоров которых подключены к набору переменных (SRAM). А выходы дешифрации являются результирующими сигналами дешифрации.

Пример реализации разработанного алгоритма для логического элемента 2-LUT. Синтезированный элемент 1-LUT в общем виде с блоком функции-дешифрации представлен на рис. 2.

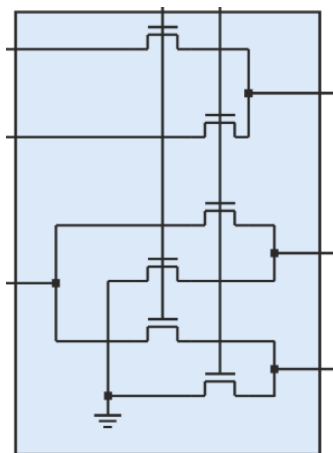


Рис. 2. Соединение блоков функции-дешифрации в базовом логическом элементе 1-LUT

Мультиплексор вычисления основной функции определяется двумя передающими транзисторами с входными сигналами $F_{n-1}(0)$ и $F_{n-1}(1)$ и выходным сигналом F_n , которые управляются инвертированным и не инвертированным сигналами переменной X_n . Блок функции-дешифрации состоит из четырех передающих транзисторов и представляет собой обратное дерево с ортогональностью сигнала. Входным сигналом блока функции-дешифрации является P_n , а выходными – сигналы $P_{n-1}(0)$ и $P_{n-1}(1)$. Истоки двух транзисторов объединены и подключены к константе *Ground*. Управление передающими транзистора-

ми в блоке функции дешифрации осуществляется с помощью инвертированного и не инвертированного сигналами переменной X_n . Результат синтеза логического элемента на две переменные представлен на рис. 3. Для данного 2-LUT требуется три базовых блока ($2^n - 1$). Подключение входа дешифрации старшей переменной (P_2) осуществляется к высокому уровню сигнала (V_{cc}).

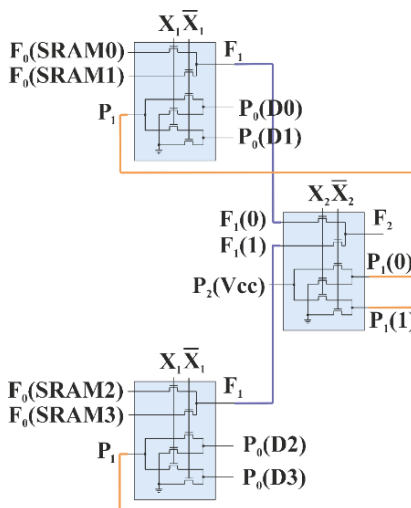


Рис. 3. Логический элемент 2-LUT, выполняющий вычисление основной логической функции и дешифрацию входного набора одновременно

Входы функций блоков младшей переменной подключаются к ячейкам статической памяти (SRAM0-SRAM3), а выходы дешифрации ($P_0(D0-D3)$) определяют использование соответствующего набора переменных. Базовые мультиплексоры с блоками функции-дешифрации соединяются между собой соответствующими входами-выходами.

Пример реализации разработанного алгоритма для логического элемента 3-LUT. Результат синтеза логического элемента на три переменные представлен на рис. 4. Для данного 3-LUT требуется семь базовых блоков ($2^n - 1$).

При увеличении количества переменных добавляется дополнительный каскад по младшей переменной с соответствующими подключениями ячеек статической памяти и выходов дешифрации.

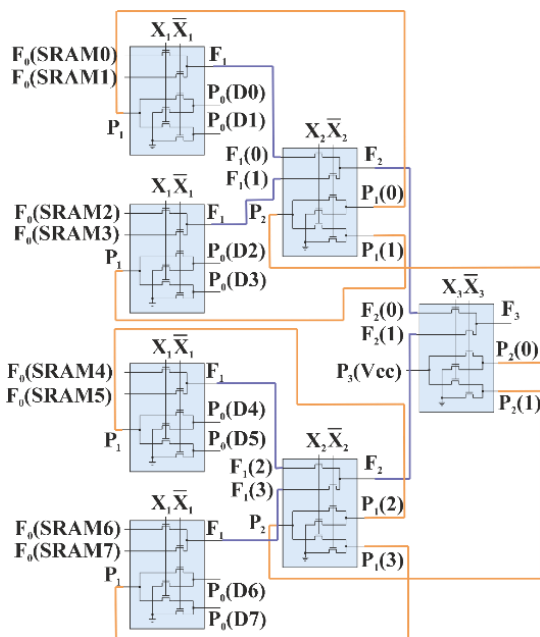


Рис. 4. Синтез логического элемента 3-LUT, выполняющего вычисление основной логической функции и дешифрацию входного набора одновременно

Заключение. Разработанный алгоритм подключения блоков функции-дешифрации позволяет синтезировать логический элемент LUT, который выполняет вычисление основной функции и дешифрацию входного набора одновременно. Описано подключение необходимых входов, выходов базовых элементов. Приведены примеры синтеза логических элементов на две и три переменные с дешифрацией входных переменных.

Библиографический список

1. Тюрин С.Ф., Советов С.И. Логический элемент ПЛИС FPGA, реализующий функцию и дешифрацию набора переменных // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2023. – № 47. – С. 5–31. DOI: 10.15593/2224-9397/2023.3.01

2. Строгонов А.В. Системное проектирование программируемых логических интегральных схем: учеб. пособие. – Воронеж: Воронежский государственный технический университет. – 2012. – 322 с.

3. Патент № 2811404 Российская Федерация, МПК G06F 7/57 H03K 19/173. Программируемое логическое устройство: заявл. 02.08.2023; опубл. 11.01.2024 / С.Ф. Тюрин, И.А. Васенин, Ю.А. Степченко, Ю.Г. Дьяченко, С.И. Советов.

4. Каменских А.Н. Комбинированное резервирование самосинхронных схем: дис. ... канд. тех. наук. – Пермь, 2016. – 185 с.

5. Вихорев Р.В. Логические элементы ПЛИС FPGA для реализации систем функций: автореф. дис. ... канд. тех. наук: 05.13.05. – Пермь, 2017. – 21 с.

6. An FPGA Architecture for ASIC-FPGA Co-design to Streamline Processing of IDSs / S. Tomoaki, C. Sorawat, M. Phichet, H. Kohji // International Conference on Collaboration Technologies and Systems. – 2016. DOI: 10.1109/CTS.2016.0079

7. Skornyakova A.Yu., Vikhorev R.V. Self-Timed LUT Layout Simulation // Conference of Russian Young Researchers in Electrical and Electronic Engineering. – 2020. – P. 176–179. DOI: 10.1109/EIconRus49466.2020.9039374

8. Zhou Y., Jiang J. An FPGA-based accelerator implementation for deep convolutional neural networks // Proc. 4th Int. Conf. Comput. Sci. Netw. Technol. (ICCSNT). – Harbin, China. – 2015. – P. 829–832.

9. High-Speed Phase Structured Light Integrated Architecture on FPGA / C. Zhong, H. Tianlong, H. Yuya, Z. Xianmin. // IEEE Transactions on Industrial Electronics. – 2024. – Vol. 71, iss. 1. – P. 1017–1027. DOI: 10.1109/TIE.2023.3250771

Сведения об авторах

Советов Станислав Игоревич – аспирант кафедры «Автоматика и телемеханика» Пермского национально исследовательского политехнического университета, г. Пермь, e-mail: fizikoz@gmail.com

Тюрин Сергей Феофентович – доктор технических наук, профессор кафедры «Автоматика и телемеханика» Пермского национально исследовательского политехнического университета, г. Пермь, e-mail: tyurinsergfeo@yandex.ru

А.Е. Пономарев, А.В. Кривцун

Южно-Российский государственный политехнический
университет (НПИ) им. М.И. Платова, Новочеркасск

ИНФОРМАЦИОННАЯ СИСТЕМА УПРАВЛЕНИЯ ПРОЕКТАМИ МОЛОДЕЖНОГО ДВИЖЕНИЯ

Представлены результаты разработки информационной системы управления проектами молодежного движения, предназначенной для организации деятельности молодежного движения, упрощения подготовки проектов и учета членов движения. Для реализации проекта использовались современные инструменты и технологии, включая Nest.js и React.js для разработки информационной системы, PostgreSQL для базы данных и другие. Рассмотрены основные фреймворки и их характеристики для Бекенд и Фронтенд разработки, а также представлена визуальная реализация разработанной информационной системы.

Ключевые слова: информационная система, проектное управление, молодежное движение, фреймворки.

A.E. Ponomarev, A.V. Krivtsun

South Russian State Polytechnic University (NPI)
named after M.I. Platov, Novocherkassk

INFORMATION SYSTEM FOR MANAGING PROJECTS OF YOUTH MOVEMENT

This article presents the results of the development of an information system for managing projects of the youth movement, designed to organize the activities of the youth movement, simplify the preparation of projects and record members of the movement. Modern tools and technologies were used to implement the project, including Nest.js and React.js for developing an information system, PostgreSQL for a database and others. The article discusses the main frameworks and their characteristics for Backend and Frontend development, and also presents a visual implementation of the developed information system.

Keywords: information system, project management, youth movement, frameworks.

Молодежная общероссийская общественная организация «Российские студенческие отряды» (PCO) – крупнейшая молодежная организация страны, которая обеспечивает временной трудовой занятостью более 200 тыс. молодых людей из 82 субъектов РФ, а также занимается гражданским и патриотическим воспитанием, развивает творческий и спортивный потенциал молодежи.

МООО «РСО» имеет строгую структуру организации: от обычного участника до центрально штаба движения. Каждый уровень иерархии выполняет свои определенные функции: региональное отделение выполняет работу посредника между работодателем и студентом, руководители линейного студенческого отряда обучают и готовят студента к трудовому семестру, работники штаба вуза контролируют деятельность отрядов на уровне учебной организации.

Каждый уровень требует к себе повышенного внимания и ставит перед руководителем множество задач, зачастую совершенно отличающихся друг от друга: от проведения мероприятия или собрания отряда до подготовки документации для выезда на рабочий объект и учета членства в организации.

Для решения подобной проблемы необходимо использовать информационную систему для создания проектов, которая предоставит руководителям удобный инструмент контроля выполнения задач.

Информационная система (ИС) молодежного движения предназначена для управления проектами организации. Она позволяет добавлять новые задачи для пользователей, вести учет выполнения проектов, а также контролировать списки участников организации. ИС обеспечивает возможность контроля выполнения задач пользователями для руководителя, учета членства в организации и рейтинга отдельного участника. ИС предполагает удобный интерфейс для управления проектами, позволяя создавать не только новые задачи и проекты, но и команды (группы) из пользователей, которые будут заниматься некоторым массивом проектов или задач. При использовании ИС руководитель должен иметь возможность контролировать и добавлять задачи, создавать новые проекты, добавлять новых пользователей.

Для разработки информационной системы управления проектами молодежного движения использовались современные и широко используемые технологии программирования, которые обеспечивают эффективность, производительность и удобство разработки.

Разработка проекта велась на языке TypeScript, несмотря на преимущества JavaScript, TS TypeScript обеспечивает лучшее завершение кода, рефакторинг и возможности обнаружения ошибок в интегрированных средах разработки (IDE). А также является более расширенной и удобной версией JavaScript [1].

Бекенд разработка велась на фреймворке Nest.js, который значительно расширяет и превосходит функционал express.js. Фреймворк позволяет подключать к Node.js любые базы данных, а также поддерживает подавляющую часть библиотек. NestJS самостоятельно устанавливает зависимости, а при добавлении новых сервисов и контроллеров распределяет их по директориям и автоматически подключает к соответствующим модулям приложения [2]. Учитывая громоздкость базы данных, которую было необходимо создавать, Nest.js стал более предпочтительным на фоне других фреймворков. В табл. 1 приведены основные фреймворки и их характеристики для Бекенд разработки.

Таблица 1

Сравнение фреймворков для Бэкенда

Критерии (от 1 до 10)	Фреймворки				
	Express.js	Koa.js	Hapi.js	Nest.js	Fastify
Производительность	8	9	7	8	10
Настраиваемость	9	7	8	9	8
Поддержка платформ	8	7	9	8	8
Сообщество и поддержка	10	8	8	10	7

Для Фронтенда была выбрана библиотека React.js. В отличие от своих аналогов библиотека обладает наибольшей универсальностью и гибкостью, а также четким синтаксисом [3]. Благодаря реализации React Virtual DOM позволяет оптимизировать рендеринг и ускорить работу системы. Учитывая относительно сжатые сроки и высокий объем работы, React.js стал наиболее актуален для разработки Фронтенда. В табл. 2 приведены основные фреймворки и их характеристики для Фронтенд разработки.

Таблица 2

Сравнение фреймворков для Фронтенда

Критерии (от 1 до 10)	Фреймворки				
	React	Vue.js	Angular	Svelte	Next.js
Производительность	9	8	7	9	8
Настраиваемость	8	9	9	7	8
Поддержка платформ	9	8	9	7	9
Сообщество и поддержка	10	9	8	8	9

PostgreSQL стала наиболее подходящей СУБД для работы с JavaScript при создании информационной системы. PostgreSQL обеспечивает надежность, решает задачи связей данных и поддерживает сложные запросы и транзакции, что может быть важным при работе с данными пользователей и работе с проектами.

Для настройки среды разработки и сборки проекта был использован инструмент Vite.js. Инструмент позволяет при минимальной конфигурации быстро начать и подготовить проект к дальнейшей работе, а также предоставляет Dev-сервер с горячей перезагрузкой, которая позволяет видеть изменения в коде без перезагрузки страницы. Также Vite.js хорошо интегрируется с React, что позволило предотвратить ряд сложностей при разработке.

С учетом разработанных алгоритмов и методик ИС предоставляет пользователю механизмы для удобного ведения проекта, учета членов своего отряда, анализа полученных данных и оповещения своих подчиненных. Регистрация и вход в ИС представлены на рис. 1, 2.

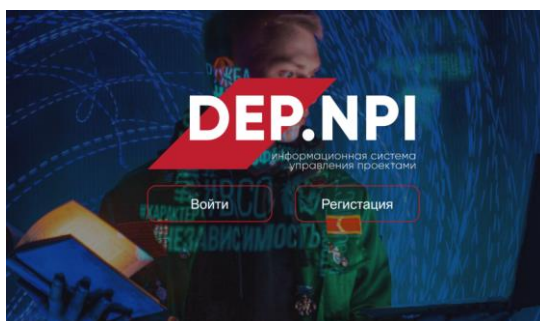


Рис. 1. Экран приветствия

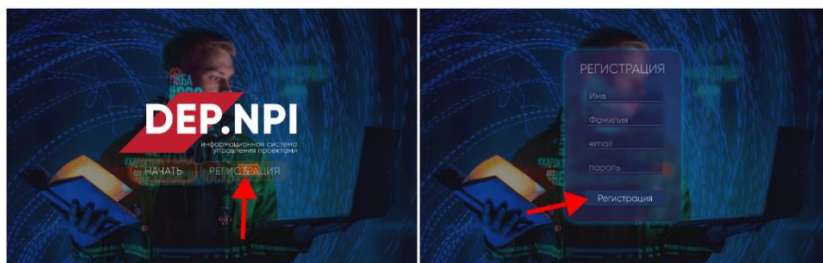


Рис. 2. Регистрация пользователя

На главной странице пользователь может просмотреть все события и задачи, в которых он может принять участие или уже участвует. Главная страница представлена на рис. 3.

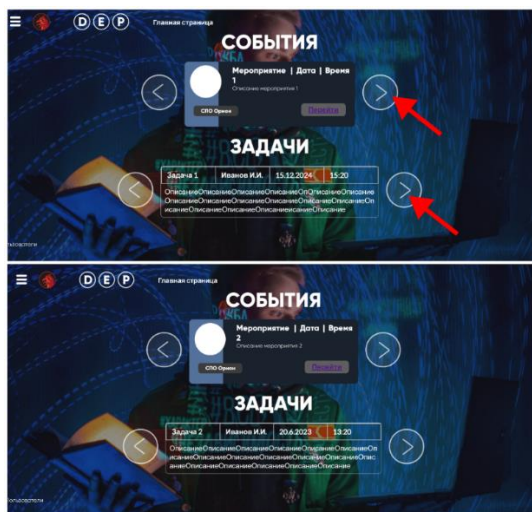


Рис. 3. Главное меню

В результате была разработана информационная система управления проектами молодежного движения, предназначенная для организации деятельности движения, упрощения подготовки проектов и учета членов движения. Для реализации проекта использовались современные инструменты и технологии, включая Nest.js и React.js для разработки информационной системы, PostgreSQL для базы данных и другие. Результатом данной работы стало эффективное и удобное приложение, способное помочь пользователю более эффективно организовывать собственных проекты, а также руководителям движения взаимодействовать с его членами.

Библиографический список

1. Дэн Вандеркам. Эффективный TypeScript. – СПб.: Питер, 2020. – 288 с.
2. Nestjs.com – свободная энциклопедия [Электронный ресурс] – URL: <https://docs.nestjs.com/> – документация (даты обращения: 24.02.2024 – 27.04.2024).

3. React.dev – свободная энциклопедия [Электронный ресурс]. – URL: <https://react.dev/learn> – документация (даты обращения: 20.03.2024–14.05.2024).

Сведения об авторах

Пономарев Андрей Егорович – студент 4-го курса программы бакалавриата по направлению подготовки «Программное обеспечение вычислительной техники» Южно-Российского государственного политехнического университета (НПИ) им. М.И. Платова, г. Новочеркасск, e-mail: ponomarev@gmail.com

Кривцун Анастасия Витальевна – кандидат технических наук, доцент кафедры «Информационные и измерительные системы и технологии» Южно-Российского государственного политехнического университета (НПИ) им. М.И. Платова, г. Новочеркасск, e-mail: krivtcun_av@npi-tu.ru

А.Ю. Ананина, Е.В. Долгова

Пермский национальный исследовательский
политехнический университет, Пермь

ЧИСЛЕННЫЕ ИССЛЕДОВАНИЯ ПРЯМОЙ ЗАДАЧИ КИНЕМАТИКИ ДЛЯ РОБОТА-МАНИПУЛЯТОРА

В статье проведено численное исследование решения прямой задачи кинематики робота-манипулятора. Описан процесс разработки и реализации алгоритма, позволяющего определить положение рабочего органа робота исходя из заданных значений углов поворота звеньев манипулятора. Полученные выводы могут быть полезны для улучшения эффективности работы робототехнических систем и оптимизации процессов автоматизации производства.

Ключевые слова: робот-манипулятор, прямая задача кинематики, кинематическая пара, углы Эйлера, звено.

A.Yu. Ananina, E.V. Dolgova

Perm National Research Polytechnic University, Perm

NUMERICAL STUDIES OF THE DIRECT KINEMATICS PROBLEM FOR A ROBOT MANIPULATOR

The article conducts a numerical study of solving the direct problem of kinematics of a robot manipulator. The process of developing and implementing an algorithm is described that makes it possible to determine the position of the robot's working body based on the specified values of the rotation angles of the manipulator links. The findings can be useful for improving the efficiency of robotic systems and optimizing production automation processes.

Keywords: robot manipulator, direct kinematics problem, kinematic pair, Euler angles, link.

Для успешного выполнения прямой задачи кинематики робота-манипулятора требуется обладать математическими инструментами, способными уточнить математическое описание кинематики манипулятора в любом заданном положении [1]. Эту модель можно описать следующим образом в общих чертах для прямой задачи кинематики:

$$\begin{cases} x_{0n} = x(q_1, q_2, \dots, q_n; t); \\ y_{0n} = y(q_1, q_2, \dots, q_n; t); \\ z_{0n} = z(q_1, q_2, \dots, q_n; t); \\ x_0 \wedge z_n = f_1(q_1, q_2, \dots, q_n; t); \\ y_0 \wedge z_n = f_2(q_1, q_2, \dots, q_n; t); \\ x_0^{y_n} = f_3(q_1, q_2, \dots, q_n; t). \end{cases} \quad (1)$$

Задачи такого типа сводятся к переводу координатных систем. В общей ситуации для фиксации одной системы координат относительно другой требуется точное описание шести параметров: три смещения для перемещения по осям и три угла для поворота вокруг них.

Используем следующие параметры:

1. D – смещение i -го звена относительно $(i + 1)$ -го по оси z (высота звена).
 2. A_x – смещение i -го звена относительно $(i + 1)$ -го по оси x (длина звена).
 3. A_y – смещение i -го звена относительно $i + 1$ -го по оси y (смещение звена).
 4. α – угол поворота звена вокруг оси z_0 .
 5. β – угол поворота звена вокруг оси y_1 .
 6. γ – угол поворота звена по оси x_2 .
 7. θ – угол поворота звена (входное значение).
- Углы α, β, γ представлены на рисунке.

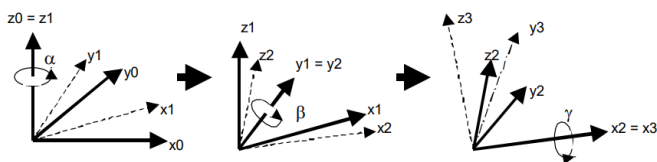


Рис. Углы поворота осей

Для вычисления координат, исходя из заданных углов поворота звеньев, применим следующий алгоритм, используя при этом углы Эйлера:

1. Поворот звена по оси z на угол θ с помощью матрицы вращения вокруг оси z .

$$M_z(\theta) = \begin{vmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{vmatrix}. \quad (2)$$

2. Сдвиг по осям x , y и z на соответствующие значения A_x , A_y , D .

$$T_{xyz}(A_x, A_y, D) = \begin{vmatrix} 1 & 0 & 0 & A_x \\ 0 & 1 & 0 & A_y \\ 0 & 0 & 1 & D \\ 0 & 0 & 0 & 1 \end{vmatrix}. \quad (3)$$

3. Вращение вокруг осей z , y , x на соответствующие углы α , β , γ .

$$\begin{aligned} & M_z(\alpha) \cdot M_y(\beta) \cdot M_x(\gamma) = \\ & = \begin{vmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{vmatrix} \cdot \begin{vmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{vmatrix} \cdot \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & -\sin \gamma \\ 0 & \sin \gamma & \cos \gamma \end{vmatrix} = \\ & = \begin{vmatrix} C(\alpha) \cdot C(\beta) & S(\gamma) \cdot S(\beta) \cdot C(\alpha) - S(\alpha) \cdot C(\gamma) & C(\alpha) \cdot S(\beta) \cdot C(\gamma) + S(\alpha) \cdot S(\gamma) \\ S(\alpha) \cdot C(\beta) & S(\alpha) \cdot S(\beta) \cdot S(\gamma) + C(\alpha) \cdot C(\gamma) & C(\gamma) \cdot S(\alpha) \cdot S(\beta) - C(\alpha) \cdot S(\gamma) \\ -S(\beta) & S(\gamma) \cdot C(\beta) & C(\beta) \cdot C(\gamma) \end{vmatrix}. \quad (4) \end{aligned}$$

4. Приведем все матрицы вращения к размеру 4×4 с помощью добавления однородной координаты и по порядку перемножим. Вставляя значения параметров каждого звена в полученную матрицу, выведем 6 матриц перехода (для шестизвенового манипулятора):

$$\begin{aligned} T_{i-1,i} &= M_z(\theta) \cdot T_{xyz}(A_x, A_y, D) \cdot M_z(\alpha) \cdot M_y(\beta) \cdot M_x(\gamma) = \\ &= \begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{vmatrix}, \quad (5) \end{aligned}$$

после чего, перемножив их между собой, получим

$$T_{0,6} = T_{0,1} \cdot T_{1,2} \cdot T_{2,3} \cdot T_{3,4} \cdot T_{4,5} \cdot T_{5,6}. \quad (6)$$

Информация о расположении системы координат $O_6X_6Y_6Z_6$ относительно неподвижной системы координат $O_0X_0Y_0Z_0$ закодирована в каждом элементе матрицы $T_{0,6}$.

Таким образом, последний столбец полученной матрицы (5) содержит координаты центра последнего звена.

$$x = a_{14}, \quad y = a_{24}, \quad z = a_{34}.$$

А наддиагональные элементы a_{12} , a_{13} , a_{23} являются направляющими косинусами осей z_6 и y_6 относительно осей x_0 и y_0 , а именно:

$$\angle x_0 z_6 = \arccos(a_{13}), \quad (7)$$

$$\angle y_0 z_6 = \arccos(a_{23}), \quad (8)$$

$$\angle x_0 y_6 = \arccos(a_{12}). \quad (9)$$

Важно отметить, что для определения положения схвата в пространстве (координаты его центра A_6 и ориентацию 6-го звена) мы используем информацию, содержащуюся в шести элементах матрицы T_{06} , находящихся над главной диагональю. Таким образом, эти шесть элементов матрицы T_{06} предоставляют полное описание положения рабочего органа в пространстве.

Библиографический список

1. Хомченко В.Г. Учебное пособие по робототехническим системам. – Омск, 2016. – 195 с.

Сведения об авторах

Ананина Арина Юрьевна – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: ananina.1999@bk.ru

Долгова Елена Владимировна – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: elena@dolgova.space

А.Д. Камбалина

Томский национальный исследовательский
политехнический университет, Томск

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ОЦЕНКИ ТРЕМОРА У ПАЦИЕНТОВ С БОЛЕЗНЬЮ ПАРКИНСОНА

В статье исследуется применение машинного обучения для ранней диагностики болезни Паркинсона. Описывается разработка десктопного приложения для оценки тремора у пациентов с диагностированной болезнью Паркинсона. В статье рассмотрены функциональные требования, этапы разработки приложения, рассмотрен прототип интерфейса приложения и технологии, используемые для разработки программного обеспечения. При помощи анализа целевых точек на видеозаписи определяются наличие и степень кинетического и постурального тремора.

Ключевые слова: болезнь Паркинсона, разработка, приложение, программное обеспечение, тремор, машинное обучение, задача классификации.

A.D. Kambalina

Tomsk National Research Polytechnic University, Tomsk

DEVELOPMENT SOFTWARE FOR THE ASSESSMENT OF TREMOR IN PATIENTS WITH PARKINSON'S DISEASE

This article examines the potential of machine learning for the early diagnosis of Parkinson's disease. It describes the development of a desktop application for tremor assessment in patients with diagnosed Parkinson's disease. The paper discusses the functional requirements, the steps involved in developing the application, a prototype of the application interface, and the technologies used for software development. By analyzing target points on the video recording, the presence and degree of kinetic and postural tremor are determined.

Keywords: Parkinson's disease, development, application, software, tremor, machine learning, classification task.

Заболевания центральной нервной системы вышли на первое место по влиянию на здоровье населения за последние 30 лет [1]. В наши дни болезнь Паркинсона является самым быстрорастущим нейродегенеративным заболеванием. Она распространена в популяции от 65 до 100 лет, а наиболее подверженными являются пожилые люди старше

85 лет. Синдромами болезни Паркинсона являются замедленность всех движений, мышечная ригидность, дрожание рук и ног, неустойчивость при ходьбе.

Болезнь Паркинсона оказывает сильное влияние на качество жизни пациентов, особенно если не был поставлен правильный и своевременный диагноз. Ранняя диагностика заболевания позволяет принять меры для замедления прогрессирования заболевания, например, назначить противопаркинсоническую терапию. Часто в связи с нехваткой квалифицированных кадров болезнь Паркинсона выявляется на более поздних стадиях, что приводит к значительному увеличению стоимости лечения и снижению продолжительности жизни пациентов.

Развитие информационных технологий позволяет выполнять раннюю диагностику с помощью алгоритмов искусственного интеллекта. Для диагностики БП могут использоваться различные методики: оценка состояния крови, анализ почерка пациента, голосовых маркеров, оценка данных с датчиков акселерометра и гироскопа, записей ЭЭГ и видеозаписей [2].

Данная разработка посвящена анализу наиболее характерного симптома болезни Паркинсона – тремора. Тремор (дрожательный гиперкинез) – это непроизвольные ритмичные колебательные движения части тела, обусловленные поочередными или одновременными сокращениями мышц агонистов и антагонистов [3].

Десктопное приложение предназначено для диагностики тремора при болезни Паркинсона. На основе анализа видеоданных пациентов с помощью методов машинного обучения происходит распознавание 21 характерной точки на ладони человека. Пациентам предлагается выполнить упражнения для определения тремора: «Кулак», «Кинетический тремор», «Постуральный тремор», «Пронация и супинация кисти», «Постукивание пальцами».

Для создания приложения была выбрана итерационная модель разработки программного обеспечения. На рис. 1 представлены основные этапы создания интеллектуальной системы.

Разработанное программное обеспечение может быть использовано в организациях здравоохранения, исследовательских лабораториях, разработчиками программного обеспечения, а также использоваться пациентами с диагностированной болезнью Паркинсона или другими неврологическими заболеваниями.

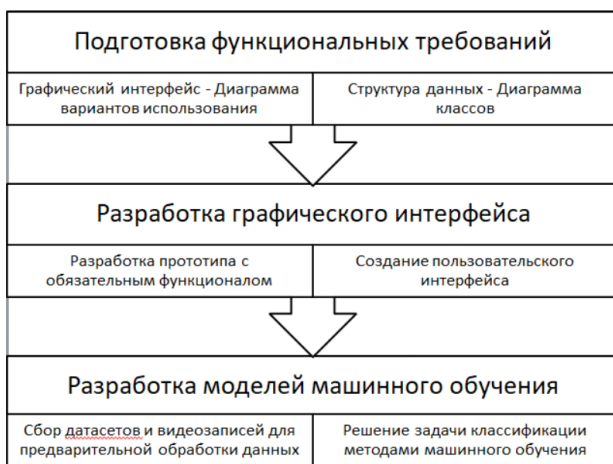


Рис. 1. Этапы разработки программного обеспечения

После формирования функциональных требований к приложению был разработан прототип графического интерфейса, представленный на рис. 2.

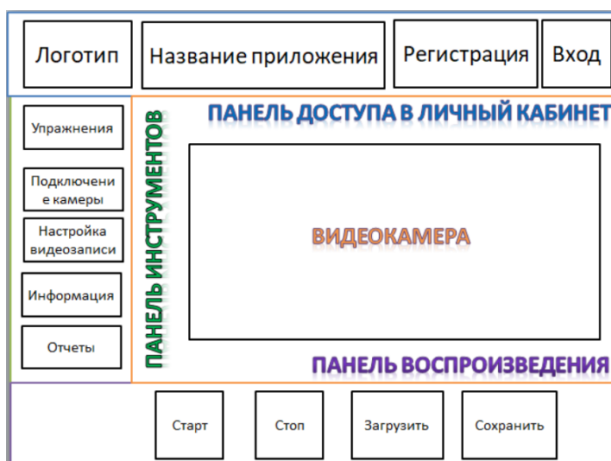


Рис. 2. Прототип разрабатываемого приложения

В таблице приведены технические средства, используемые для разработки программного обеспечения.

Средства разработки

№ п/п	Средство разработки	Назначение
1	Python 3.9	Написание алгоритма программы, описание основного функционала
2	PyCharm	Среда разработки для удобной установки библиотек
3	Tkinter, PIL, Timer, Datetime	Библиотеки необходимые для разработки графического интерфейса
4	MediaPipe, OpenCV	Распознавание характерных точек на ладони человека
5	Numpy, Glob, Pandas, Math	Выполнение предварительной подготовки данных
6	Matplotlib, Seaborn	Отображение графиков обучения моделей, характерных точек в трехмерном пространстве
7	Sklearn, Keras, Tensorflow	Обучение моделей машинного обучения, оценка точности полученных предсказаний

Основное окно разработанного приложения представлено на рис. 3.

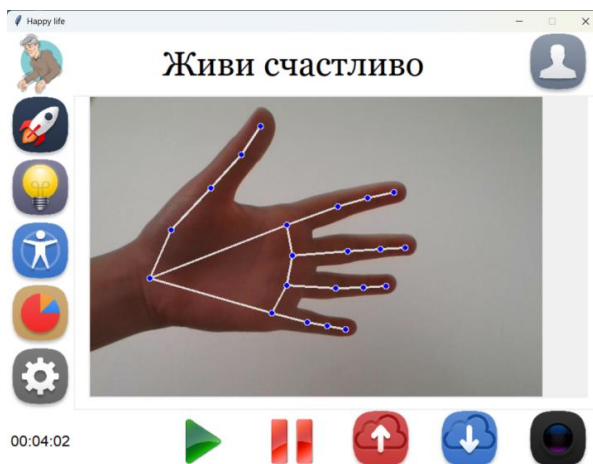


Рис. 3. Графический интерфейс разработанного приложения

Основное окно приложения состоит из 4 панелей:

1. Верхняя панель предназначена для ознакомления с наименованием и логотипом приложения, а также для доступа в личный кабинет.
2. Левая панель является панелью инструментов и позволяет выполнить настройку видекамеры, узнать советы по использованию приложения, посмотреть примеры упражнений и результат анализа видеозаписей, полученный с помощью методов машинного обучения.

3. На правой панели выполняется отображение изображения с камеры компьютера или файла, загруженного пользователем.

4. Нижняя панель необходима для запуска или остановки записи, загрузки видео с компьютера пациента или сохранения итогового видео на компьютер.

С помощью библиотеки MediaPipe происходит анализ видеозаписей и запись трехмерных координат 21 характерной точки на ладони человека в датасет. Код для распознавания точек представлен в листинге.

Листинг – Функция тренировки нейросети на Python

```
import mediapipe as mp
class Detect:
    def __init__(self):
        self.hands =
mp.solutions.hands.Hands(max_num_hands=2)
        self.draw = mp.solutions.drawing_utils

    def drawPoint(self, image):
        imageRGB = cv2.cvtColor(image,
cv2.COLOR_BGR2RGB)
        results = self.hands.process(imageRGB)
        if results.multi_hand_landmarks:
            for handLms in re-
sults.multi_hand_landmarks:
                for id, lm in enumer-
ate(handLms.landmark):
                    h, w, c = image.shape
                    cx, cy = int(lm.x * w),
int(lm.y * h)
                    self.draw.draw_landmarks(image,
handLms, mp.solutions.hands.HAND_CONNECTIONS)
                return image
```

Для решения задачи классификации использовались такие методы машинного обучения, как логистическая регрессия, алгоритм многослойного перцептрона MLP Classifier, наивный байесовский классификатор Multinomial NB, метод случайных векторов SVC, алгоритм случайного леса RF.

Заключение. Алгоритмы машинного обучения являются эффективным методом для решения задачи классификации при диагностике

неврологических заболеваний. Разработанная интеллектуальная система позволяет рассмотреть 21 ключевую точку на ладони человека и определить наличие и степень тремора у пациентов с диагностированной болезнью Паркинсона.

Библиографический список

1. Global, regional and national burden of disorders affecting the nervous system, 1990–2021: a systematic analysis for the Global Burden of Disease Study 2021 / GBD 2021 Nervous System Disorders Collaborators // The Lancet Neurology. – April 2024. – Vol. 23, iss. 4. – P. 344.

2. Пат. 2764568 Российская Федерация МПК G16H 50/00. Способ диагностики болезни Паркинсона на основе анализа видеоданных с применением машинного обучения / Сомов А.С., Дылов Д.В., Коваленко Е.А., Талицкий А.М., Аникина А.С., Щербак А.С.; Сколковский институт науки и технологий. – № 2021109243; заявл. 05.04.2021; опубл. 18.01.2022.

3. Mei J., Desrosiers C., Frasnelli J. Machine Learning for the Diagnosis of Parkinson's Disease: A Review of Literature. – URL: <https://www.frontiersin.org/articles/10.3389/fnagi.2021.633752/full>

Сведения об авторе

Камбалина Алена Дмитриевна – магистрант отделения информационных технологий Инженерной школы информационных технологий и робототехники Томского национального исследовательского политехнического университета, гр. 8ПМ2Л, г. Томск, e-mail: kambalina,alena@yandex.ru

Е.С. Масылюк, Архипов А.В.

Пермский государственный национальный
исследовательский университет, Пермь

РАЗРАБОТКА МОДУЛЯ ИНСТРУМЕНТАЛЬНОГО АНАЛИЗА МНОГОМЕРНЫХ ДАННЫХ (КУБОВ)

Данная статья посвящена разработке модуля инструментального анализа многомерных данных, представленных в виде кубов. Предлагается один из возможных способов реализации такого модуля, который включает использование специализированных библиотек. Приведены примеры реальных диаграмм и графиков, которые могут быть использованы для отображения информации.

Ключевые слова: модуль инструментального анализа многомерных данных, low-code платформа, визуализация данных.

Masylyuk E.S., Arhipov A.V.

Perm State National Research University, Perm

DEVELOPMENT OF A MODULE FOR INSTRUMENTAL ANALYSIS OF MULTIDIMENSIONAL DATA (CUBES)

This article is devoted to the development of a module for instrumental analysis of multidimensional data presented in the form of cubes. One of the possible ways to implement such a module is proposed, which includes the use of specialized libraries. Examples of real diagrams and graphs that can be used to display information are given.

Keywords: module for instrumental analysis of multidimensional data, low-code platform, data visualization.

В современном быстро меняющемся мире скорость и гибкость являются ключевыми факторами. Компании стремятся к ускорению процессов разработки, чтобы быстрее адаптироваться к новым рыночным условиям и требованиям клиентов. Для решения таких проблем подходят low-code платформы. Это метод проектирования и разработки приложений с помощью интуитивно понятного графического интерфейса и встроенных инструментов, которые делают разработку доступной даже для тех, кто не обладает глубокими знаниями программирования.

С развитием технологий и увеличением объемов информации существует потребность в эффективных инструментах и методах анализа многомерных данных. Чтобы обеспечить работу с ними, необходимы

специализированные инструменты. Анализ данных в виде таблиц может быть сложным и неэффективным. Разработка модуля инструментального анализа многомерных данных позволит пользователям быстрее и легче усваивать информацию через визуализацию.

Для разработки модуля была использована JavaScript-библиотека – React [1], известная своей гибкостью и эффективностью в создании пользовательских интерфейсов. Основой разработки являются компоненты, которые позволяют структурировать код. React предоставляет разработчикам возможность создавать переиспользуемые компоненты, что упрощает поддержку проекта и повышает его производительность. Такой подход позволяет эффективно управлять состоянием приложения и обновлять его интерфейс в реальном времени.

Чтобы пользователь имел возможность взаимодействовать с разрабатываемым модулем, требуется реализация функционала, позволяющего строить графики на основе выбранных строк или столбцов в таблице с данными. На этот случай предлагается реализация и интеграция кнопки «Построить диаграмму» в панель инструментов. При активации данной кнопки открывается модальное окно, предназначенное для более детальной конфигурации графического представления.

Пользователю предоставляется возможность определить следующие параметры: способ разбиения данных – по строкам или по столбцам (т.е., например, какие данные будут идти по оси X в линейной диаграмме); название для графика; тип диаграммы и какие данные использовать при построении (исходные значения или рост / прирост / изменение / доля по строкам / столбцам); формат подписей по осям X и Y ; отображение легенды (ее расположение); размеры диаграммы; палитру цветов.

В ходе реализации отображения графиков была использована библиотека Chart.js [2], предоставляющая широкий спектр настроек и опций для персонализации внешнего вида диаграмм. Основные параметры включают в себя: тип графика (line, bar, pie, doughnut, bubble и др.); набор данных (datasets), включая метки для осей (labels); опции для визуального отображения, поведения элементов и взаимодействия. Дополнительные функциональные возможности могут быть внедрены путем использования различных плагинов.

Для каждого типа диаграммы был создан отдельный React компонент, обеспечивающий корректное отображение выбранного варианта графика. Каждый компонент включает в себя логику обработки дан-

ных, управление состоянием и визуализацию информации в соответствии с выбранными настройками пользователя. Например, для линейной диаграммы был создан компонент LineChart, который принимает данные и параметры для построения линейного графика. Такой подход позволяет разделить функциональность по типам графиков, облегчает поддержку и дальнейшее расширение функциональности модуля. Каждый React компонент ответствен за свою уникальную задачу и обладает соответствующей структурой и логикой работы.

Пользователь может выбрать следующие типы графиков и диаграмм для отображения данных (рисунок): пончик, пирог, пузырьковая, столбчатая, линейный, воронка.



Рис. Типы диаграмм и графиков

Стоит отметить, что библиотека Chart.js не содержит готовый тип диаграммы «воронка». Поэтому для нее использовался дополнительный плагин, расширяющий возможности библиотеки.

Необходимым уточнением является то, что в зависимости от выбранных настроек не все диаграммы можно построить.

Таким образом, разработка модуля инструментального анализа многомерных данных для low-code платформ открывает новые возможности для пользователей, позволяя им легко и быстро анализиро-

вать и визуализировать данные без необходимости глубоких знаний в области программирования. Пользователи могут управлять параметрами визуализации, выбирать подходящие типы графиков, цветовые схемы и другие параметры для наилучшего представления данных. Это значительно упрощает процесс работы с данными и повышает эффективность анализа.

Библиографический список

1. Библиотека React [Электронный ресурс]. – URL: <https://react.dev/> (дата обращения: 23.05.2024).
2. Документация Chart.js [Электронный ресурс]. – URL: <https://www.chartjs.org/docs/latest/> (дата обращения: 23.05.2024).

Сведения об авторах

Масылюк Екатерина Станиславовна – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: katia-mas@mail.ru

Архипов Андрей Валерьевич – кандидат экономических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: arhipov@greendatasoft.ru

В.С. Никулин, Т.С. Орлов

Пермский государственный национальный
исследовательский университет, Пермь

СЕРВЕР FTP С ВОЗМОЖНОСТЬЮ ДОСТУПА К ФАЙЛАМ ПОСРЕДСТВОМ СТАНДАРТА USB

Статья посвящена разработке устройства и программного обеспечения на базе микрокомпьютера Orange Pi Zero 2W, которое обеспечивает возможность совместной работы со стандартами USB, Wi-Fi и FTP. Устройство совмещает в себе функционал USB накопителя и FTP-сервера с возможностью доступа к единому разделу памяти, что является основным достоинством разработки.

Ключевые слова: объект, управление, Linux, USB, Wi-Fi, FTP, питание, универсальность.

V.S. Nikulin, T.S. Orlov

Perm State National Research University, Perm

FTP SERVER WITH THE ABILITY TO ACCESS FILES BY USB STANDARD

The article is devoted to the development of a device and software based on the Orange Pi Zero 2W microcomputer, which provides the ability to work together with USB, Wi-Fi and FTP standards. The device combines the functionality of a USB drive and an FTP server with the ability to access a single memory section, which is the main advantage of the development.

Keywords: object, control, Linux, USB, Wi-Fi, FTP, power, versatility.

В нашем мире для хранения и работы с информацией используется множество стандартов, что усложняет создание единой файловой системы. Микрокомпьютер позволяет объединять разные стандарты в пределах одного устройства посредством наличия вычислительной мощности и разных интерфейсов коммуникации, а также способов работы

с файлами и информацией в целом. В ходе исследования была рассмотрена идея создания прямой связи между стандартом FTP [1] и USB-flash [2] накопителем.

Работа посвящена созданию единой системы на базе Orange Pi Zero 2W [3] (далее Orange Pi) с добавлением системы независимого питания

на основе МН-CD42. Orange Pi, обладая вычислительными мощностями, позволяет получать и отправлять файлы посредством FTP, далее посредством системы на базе Linux предоставлять полученные файлы конечному пользователю.

Постановка задачи. В работе проводится исследование алгоритма FTP и эмуляция USB-накопителя. Посредством создания собственного драйвера USB-устройства микрокомпьютер может предоставлять необходимую информацию для конечной системы, что позволит системе определить устройство как накопитель и начать работать с ним.

Далее при помощи инструментов внутри операционной системы микрокомпьютера создается FTP-сервер и настраивается на хранение данных в пределах секторов USB-накопителя. Данный подход обеспечивает наличие двух независимых систем, которые могут работать совместно в одном секторе, что позволяет гибко настраивать необходимые параметры как со стороны FTP-сервера, так и со стороны USB-накопителя. Для обеспечения независимости от внешних источников питания была собрана схема с аккумулятором и контроллером заряда и разряда МН-CD42.

Основная часть. Рассмотрим два стандарта для работы с информацией и причины их применения в данной работе:

1. FTP «File Transfer Protocol» – это протокол, позволяющий передавать данные между клиентом и сервером. В работе используется подвид протокола, SFTP «SSH File Transfer Protocol», данный протокол обеспечивает шифрование данных при передаче, что позволяет повысить безопасность соединения. Дополнительно каждый клиент должен быть авторизован посредством логина и пароля, установленных сервером. Это позволит ограничить доступ к серверу и при необходимости настроить уровни доступа для разных пользователей.

2. USB «Universal Serial Bus» – интерфейс и стандарт для коммуникации между устройствами. Главное преимущество стандарта в его распространенности и универсальности, что позволяет охватывать большой спектр устройств. В связи с этим во многих системах есть предустановленные пакеты драйверов, необходимые для работы со стандартом. В работе используется класс USB-устройства Mass Storage, предоставляющий оболочку для описания устройств хранения информации.

Для работы с Orange Pi была установлена система Debian [4] на micro SD карту. Перед запуском микрокомпьютера на карте был

отредактирован файл «orangepi_first_run.txt» для настройки автоматического подключения к Wi-Fi сети. После первичной загрузки система и ее составляющие были обновлены до последней версии. Для создания FTP-сервера установили пакет vsftpd. Чтобы обеспечить доступ к FTP только при использовании логина и пароля создан отдельный пользователь и настроен автозапуск сервера.

Перед созданием накопителя необходимо создать раздел внутри памяти компьютера с подходящей файловой системой. В качестве файловой системы используется Fat32 [5]. Причина использования Fat32 заключается в распространенности данного стандарта с широкой поддержкой. Был создан раздел размером в 1 GB.

Эмуляция USB происходила при помощи драйверов для ядра Linux: libcomposite [6]. Далее произведена настройка описания USB-устройства по спецификации USB-стандарта. В драйвере libcomposite присутствует несколько базовых классов для USB-устройств, для создания накопителя используется класс mass_storage.

Далее представлена демонстрация работы в двух режимах: FTP-сервер и USB-накопитель (рис. 1, 2).

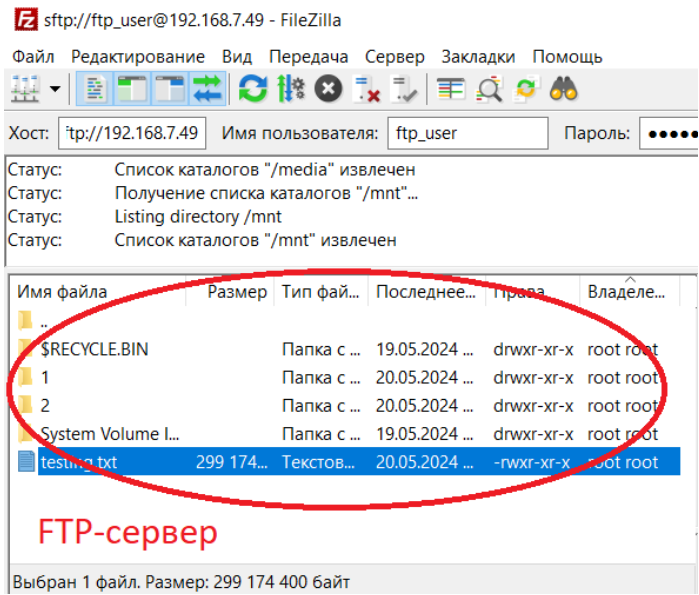


Рис. 1. Работа в качестве FTP-сервера

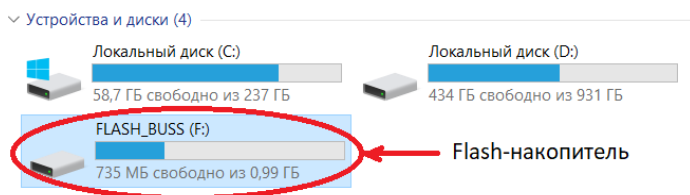


Рис. 2. Работа в качестве Flash-накопителя

Далее представлены результаты оценки скорости передачи двух стандартов FTP и USB flash-накопителя (рис. 3, 4).

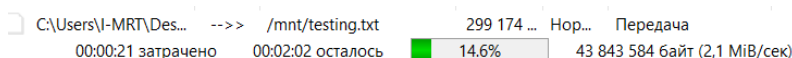


Рис. 3. Результаты оценки скорости работы FTP-сервера

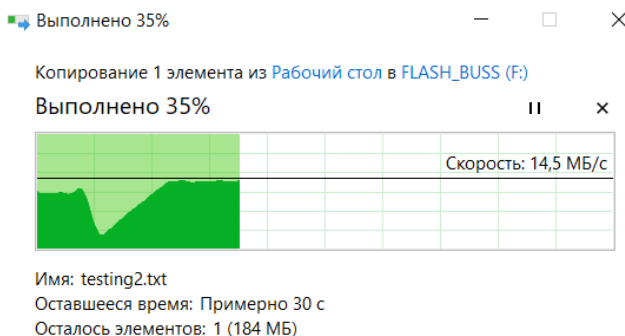


Рис. 4. Результаты оценки скорости работы Flash-накопителя

Ниже представлена сравнительная таблица работы двух стандартов.

Сравнение скоростей передачи двух стандартов

Скорость, МБ/с	FTP-сервер	Flash-накопитель
Прием/передача	2,1	14,5

Из таблицы видно, что скорость передачи Flash-накопителя превосходит FTP-сервер в 6,9 раз. Это обусловлено тем, что скорость ограничена сетью, в которой располагается FTP-сервер.

Для обеспечения автономности работы до 5 ч как FTP-сервера, так и USB-flash накопителя разработана схема на основе Orange Pi

и MH-CD42 (рис. 5), в качестве источника питания выбран Li-Po аккумулятор с емкостью 1500 мА/ч, схема дополнена информационной и питающей линией с USB-type A. Это позволяет питаться схеме как от USB-разъема, так и от аккумулятора.

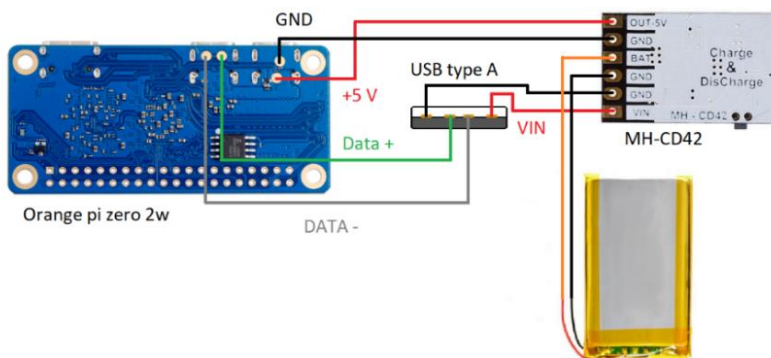


Рис. 5. Схема электрическая структурная FTP сервера и Flash-накопителя

Заключение. В результате разработано устройство, совмещающее в себе функционал USB-накопителя и FTP-сервера с возможностью доступа к единому разделу памяти, что является основным достоинством разработанной схемы. Благодаря использованию независимой цепи питания устройство может оставаться в режиме FTP-сервера без необходимости подключения к конечному устройству в течение 5 ч. FTP-сервер передает и принимает файлы в 6,9 раз медленнее, чем USB-накопитель. Основной причиной малой скорости приема/передачи FTP-сервера является скорость сети, в которой расположено устройство. Для USB-накопителя скорость ограничена интерфейсами подключения. Однако оба стандарта не могут превзойти скорость чтения и записи SD-карты микрокомпьютера, так как вся информация хранится на ней. В дальнейших исследованиях предлагается углубиться в стандарт USB для создания универсального устройства управления и передачи информации.

Библиографический список

1. Notess G.R. Learning to FTP // Online. – 1994. – Vol. 18, no. 2. – P. 79–82.
2. Морозов А.В., Радченко С.Е. Особенности проектирования измерительных USB-устройств // Сб. науч. тр. Новосибирского государственного технического университета. – 2008. – № 2 (52). – С. 53–58.

3. Окороков В.Г., Скуднев Д.М. Orange Pi PC – полноценный мини-ПК на Linux // Информационные технологии в процессе подготовки современного специалиста: межвуз. сб. науч. тр. Вып. 22. – Липецк: Липецкий государственный педагогический университет им. П.П. Семенова-Тян-Шанского, 2018. – С. 106–108.

4. Нагородская В. Роль Критериев Debian для определения свободного программного обеспечения / В. Нагородская // Интеллектуальная собственность. Авторское право и смежные права. – 2016. – № 5. – С. 19–25.

5. Леонтьев В.О., Великосельский С.А., Терещенко Р.В. Файловые системы Windows // Академия педагогических идей Новация. Сер. Студенческий научный вестник. – 2018. – № 3. – С. 89–97.

6. Красов А.В. Выбор модуля ядра ос Linux для внедрения ЦВЗ // Информационная безопасность регионов России (ИБРР-2023): материалы XIII Санкт-Петербург. межрегион. конф.; 25–27 октября 2023 г., Санкт-Петербург. – СПб.: Санкт-Петербургское общество информатики, вычислительной техники, систем связи и управления, 2023. – С. 349–351.

Сведения об авторах

Никулин Вячеслав Сергеевич – ассистент кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kalif23@yandex.ru

Орлов Тимофей Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. СРРС-22-16, г. Пермь, e-mail: Timofey.O.S@yandex.ru

С.А. Юрченко

Пензенский государственный университет, Пенза

РАЗРАБОТКА СЕРВИСА ПОДБОРА ЛЕКАРСТВЕННЫХ ПРЕПАРАТОВ

В данной статье описывается процесс разработки и архитектура приложения по подбору лекарственных препаратов. Обосновывается актуальность данной разработки, описывается список задач для разработки и приводится список аналогов, для опоры при разработке.

Ключевые слова: лекарства, лекарственные препараты, подбор лекарств, разработка сервиса.

S.A. Yurchenko

Penza State University, Penza

DEVELOPMENT OF A MEDICINE SELECTION SERVICE

This article describes the development process and architecture of a drug selection application. The relevance of this development is substantiated, a list of tasks for development is described, and a list of analogues is provided for support during development.

Keywords: medicines, medications, selection of medications, service development.

В современном мире постоянно появляются новые приложения, позволяющие улучшить жизнь человека. Разработка новых сетевых сервисов для обеспечения актуальных, точных и полезных сведений, становится важной для повышения уровня жизни людей. Эти тенденции затронули и область медицины: сетевой компонент является неотъемлемой частью медицинских информационных систем, телемедицинских систем, а также обучающих систем [1]. Особенно актуально создание сервиса для подбора лекарственных препаратов, который будет помогать в широком ассортименте медикаментов с выбором наиболее эффективного и безопасного средства. Данная статья будет посвящена разработке и анализ архитектуры такого сервиса, который может быть как независимым, так и частью системы поддержки принятия врачебных решений [2].

Актуальность разработки обусловлена несколькими ключевыми факторами. Во-первых, необходимостью повышения знаний потреби-

телей о лекарственных препаратах. Недостаточный уровень знаний в области медицины часто приводит к неправильному выбору или вообще отсутствию ожидаемого эффекта, а также к возникновению побочных реакций. Во-вторых, экономическая выгода. Достигается благодаря оптимизации выбора, что помогает избежать излишних трат. Кроме того, с помощью сервиса можно всегда подобрать точную дозировку лекарственного препарата. Создание и внедрение такого сервиса позволит в значительной степени упростить и ускорить процесс подбора лекарств, что, безусловно, принесет социальную выгоду для потребителя.

Процесс разработки по подбору лекарственных препаратов может опираться на существующие аналоги, которые уже завоевали доверие пользователей и являются важными элементами в системе здравоохранения. Ниже приведены несколько российских аналогов.

Пилула.ру (Piluli.ru) – это онлайн-сервис, предлагающий обширный спектр информации о лекарствах, их наличии и ценах в аптеках. Клиенты с помощью поиска и выбора нужного медикамента могут найти интересующий препарат, прочитать описания, противопоказания и побочные эффекты.

Аптека.ру (Apteka.ru) – онлайн-сервис, предлагающий возможность узнать о наличии и стоимости лекарств в регионе пользователя. Оформить заказ с доставкой на дом. Также предлагает консультации с фармацевтами и информационные статьи.

Яндекс.Аптека – сервис от компании «Яндекс». Позволяет пользователю сравнивать цены на лекарства в различных аптеках и делать заказ через Интернет. Удобный интерфейс, а также интеграция с остальными сервисами Яндекса обеспечивают высокий уровень коммуникации с конечным пользователем.

ЗдравСити (ZdravCity) – веб-сайт и мобильное приложение. В этом сервисе можно не только искать лекарства, но и заказывать на дом. Платформа предоставляет информацию о дозировке, способе применения, побочных реакциях лекарства.

Задача исследования: ход разработки web-сервиса [3] (мобильного сервиса) для подбора лекарственных средств по запросу человека на основе полученных начальных данных для поиска.

Цель исследования: разработка информационной системы, которая будет способствовать актуальному поиску лекарственных средств, что,

безусловно, повысит уровень информационной поддержки пациентов и работников.

Основные задачи:

1. Анализ существующих аналогов, как отечественных, так и международных.
2. Определение ключевых требований к функционалу и интерфейсу разрабатываемого сервиса.
3. Разработка общей концепции и технического задания.
4. Программная реализация прототипа.
5. Тестирование сервиса на целевой аудитории.
6. Анализ полученных данных из предыдущего пункта для улучшения работы сервиса.

Данный сервис будет включать в себя две архитектурные части. Первая часть – это серверное приложение [4], взаимодействующее с базой данных для создания, изменения, удаления данных по препаратам. Серверное приложение может быть написано на языке JavaScript [5] с использованием библиотеки TypeScript [6] и фреймворка NestJS [7].

Архитектура базы данных должна быть следующей (рисунок).



Рис. Архитектура базы данных

Серверная часть должна отдавать следующие данные:

1. Список препаратов.
2. Список компонентов.
 - а. Поиск по компоненту/компонентам.
 - б. Поиск по имени.

- с. Поиск по дозировке.
- d. Поиск по количеству компонента в препарате.
- 3. Авторизация/регистрация пользователя.

Этого функционала хватит для первого прототипа сервиса, в дальнейшем можно будет развивать, дополняя новый функционал.

Пользовательская часть – мобильное приложение. Может быть разработано для кроссплатформенности на flutter [8], react native [9], ionic [10, 11]. Должен представлять собой понятный пользовательский интерфейс и поддерживать следующий функционал.

- 1. Экран поиска препарата.
 - a. Поиск по компоненту/компонентам.
 - b. Поиск по имени.
 - c. Поиск по дозировке.
- 2. Экран поиска компонента.
- 3. Экран авторизации.
- 4. Экран регистрации.

Для запуска всей инфраструктуры нужно будет арендовать сервер для сборки приложений, а также развертывания серверной части. Благодаря фреймворкам, поддерживающим кроссплатформенность, можно будет опубликовать пользовательское приложение в apple store и google play market для скачивания приложения.

Библиографический список

- 1. Сетевой компонент информационного пространства современного хирургического тренажера / А.В. Колсанов, А.К. Назарян, А.В. Иващенко, А.В. Кузьмин // Программные системы и вычислительные методы. – 2015. – № 1. – С. 11–20.
- 2. Демушкина К.М., Демушкин М.О., Кузьмин А.В. Обзор методов проектирования систем поддержки принятия врачебных решений // Модели, системы, сети в экономике, технике, природе и обществе. – 2022. – № 2 (42). – С. 75–89.
- 3. Duckett J. HTML & CSS: Design and Build Websites. – John Wiley & Sons, 2014.
- 4. Web API 2: Building a REST Service from Start to Finish. By Jamie Kurtz and Brian Wortman.
- 5. Фреймворк. – URL: <https://docs.nestjs.com/>
- 6. Язык программирования. – URL: <https://learn.javascript.ru/>
- 7. Язык программирования. – URL: <https://www.typescriptlang.org/>

8. Фреймворк. – URL: <https://docs.flutter.dev/ui>
9. Фреймворк. – URL: <https://reactnative.dev/>
10. Фреймворк. – URL: <https://ionicframework.com/>
11. Владимирский А.В., Лебедев Г.С. Телемедицина. – М.: ГЭОТАР-Медиа, 2018. – 576 с.

Сведения об авторе

Юрченко Сергей Александрович – аспирант кафедры «Информационно-вычислительные системы» Пензенского государственного университета, гр. 21а-051301(09), г. Пенза, e-mail: alexandrovih3@gmail.com

Д.В. Постных, В.С. Никулин

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ПОДХОДА К ПОДБОРУ РАДИУСОВ ДИСПЕРСИЙ В РАДИАЛЬНО-БАЗИСНЫХ СЕТЯХ

В статье рассматривается разработка и анализ алгоритмов к подбору радиусов дисперсий, с помощью которых можно ускорить процесс обучения в RBF-сетях и уменьшить получаемую погрешность при помощи алгоритма «k-ближайших соседей» вместе с регулированием радиусов дисперсий. В статье рассматривается общий алгоритм построения RBF-сети. Проведено сравнение результатов работы рассмотренных в статье алгоритмов.

Ключевые слова: RBF-сеть, искусственный интеллект, радиус дисперсий, алгоритм «k-ближайших соседей».

D.V. Postnikh, V.S. Nikulin

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF AN APPROACH TO SELECTING DISPERSION RADII IN RADIAL-BASIS NETWORKS

The article discusses the development and analysis of algorithms for selecting dispersion radii, which can be used to speed up the learning process in RBF networks and reduce the resulting error using the «k-nearest neighbors» algorithm together with the regulation of dispersion radii. The article discusses the general algorithm for constructing an RBF network. A comparison of the results of the algorithms discussed in the article was carried out.

Keywords: RBF network, artificial intelligence, radius of dispersions, «k-nearest neighbors» algorithm.

В современном мире искусственный интеллект стремительно развивается и все больше находит применение в разных сферах жизни: промышленность, коммуникация, здравоохранение, безопасность.

Также на данный момент существует множество типов искусственного интеллекта и самый популярный из них – это перцептрон [1]. Пример еще одной нейронной сети – RBF-сеть, которую широко применяют в распознавании изображений [2]. Главной особенностью данной нейронной сети является то, что RBF-сети имеют всего один скры-

тый слой, это существенно сокращает время на обучение нейронной сети [3]. Но главный минус RBF-сетей – это то, что необходимо подбирать вручную радиусы дисперсий [4].

Для разработки подхода к подбору радиусов дисперсий было за-
действовано два шага:

- 1) алгоритм «k-ближайших соседей»;
- 2) корректировка радиусов с целью уменьшить погрешность.

Работа посвящена разработке подхода к подбору радиусов дисперсий в RBF-сетях, это позволит моментально получать радиусы дисперсий для модели нейронной сети. Достоинством данного подхода является его простота реализации и высокая точность подбора радиусов дисперсий.

Основная часть. Радиально-симметричные сети являются важным классом искусственных нейронных сетей, которые широко применяются в различных областях машинного обучения и анализа данных. Эти сети представляют собой тип искусственной нейронной сети, где используется радиально-симметричная функция в качестве активирующего элемента. В отличие от других архитектур, таких как многослойные перцептроны, RBF-сети обладают уникальной структурой и механизмом обучения, уникальность заключается в наличии только одного скрытого слоя, и входные весовые коэффициенты равны единице, что значительно упрощает разработку структуры нейронной сети.

Вычисление значений радиально-симметричной функции заключается в расчете центров нейронов C_i , радиусов дисперсий функции Гаусса и весовых коэффициентов на выходе скрытого слоя [5]. Функция Гаусса позволяет математически записать нейрон, как представлено ниже:

$$\varphi = \exp\left(\frac{\|x - C\|^2}{2a^2}\right), \quad (1)$$

где $\|x - C\|$ – расстояние в евклидовой метрике.

Рассмотрим линейную модель построения нейронных сетей.

Обучение RBF-сети строится на нахождении центров C_i , радиусов дисперсий и весовых коэффициентов выходного слоя. Алгоритм «N-усреднений» дает возможность найти значения центров.

С использованием метода наименьших квадратов в матричном представлении можно вычислить весовые коэффициенты. Если центры и радиусы дисперсий уже определены, то данный метод работает следующим

образом. Надо определить интерполяционную матрицу H , для того чтобы найти значения функций активации в скрытом слое [6]. После этого можно вычислить окончательный результат весовых коэффициентов:

$$\bar{w} = (H^T \cdot H)^{-1} \cdot H^T \cdot \bar{Y}_{kc}. \quad (2)$$

Теперь необходимо рассмотреть методику определения радиусов дисперсий при составлении подбора. Вначале нужно вручную задаться радиусами дисперсий. Потом полученные радиусы начать корректировать. Для начала был построен общий принцип работы алгоритма «k-ближайших соседей» и наглядно продемонстрирован на рис. 1.

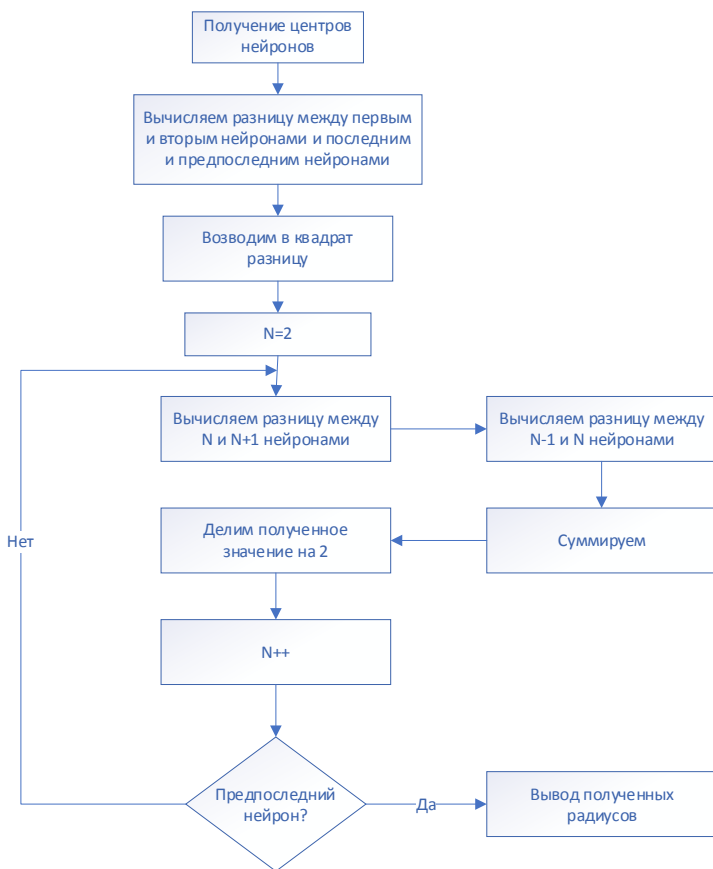


Рис. 1. Схема алгоритма «k-ближайших соседей»

Данный алгоритм необходим для того, чтобы все нейроны касались друг друга, но надо, чтобы нейроны друг с другом пересекались. Для этого был добавлен блок коррекции радиусов дисперсий, представленный на рис. 2.

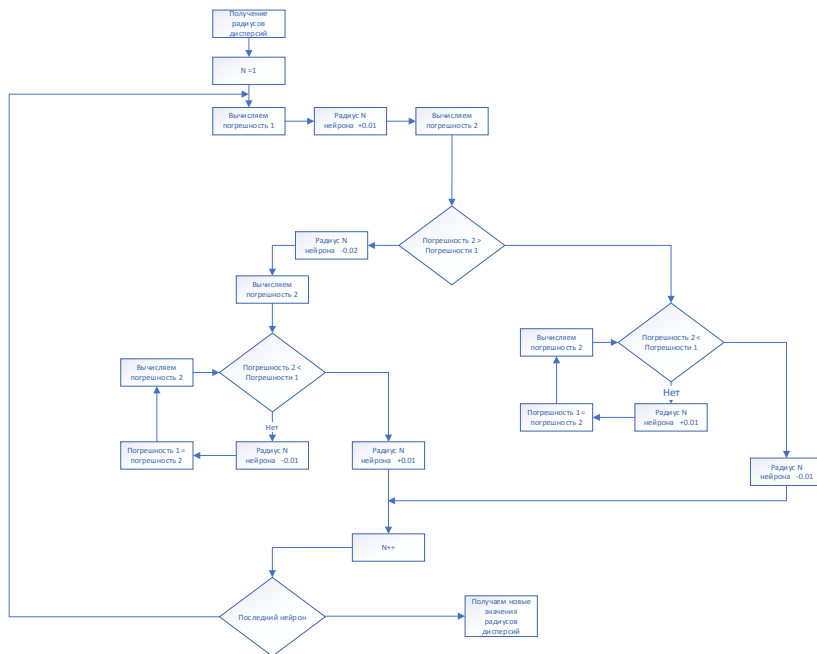


Рис. 2. Схема алгоритма коррекции радиусов дисперсий

Постановка эксперимента. Для проверки алгоритма «к-ближайших соседей» был выбран вариант распознавания изображений. Изображения черно-белые, на них изображены цифры «0», «1», «7», «8». Для каждой цифры были нарисованы изображения с разрешением «10 на 10 пикселей».

Чтобы перевести изображения из картинки в матрицу чисел, которые обозначают значение пикселя, была написана программа на языке программирования «C#». Нейронов было выбрано половина от входных параметров (значения пикселей), 50 нейронов.

После того, как получили матрицу входных значений для RBF-сети, стало необходимо написать еще одну программу для того, чтобы производить вычисления для RBF-сети.

Дальше с помощью программы необходимо получить два конечных результата для подбора радиусов. В таблице приведены результаты программных вычислений.

Сравнение результатов обучения

Желаемое значение	Результат коррекции радиусов дисперсий	Результат «к-ближайших соседей»
0,1	0,1108	-0,7179
0,1	0,10149	0,10107
0,1	0,10001	0,15181
0,1	0,10157	0,09433
0,1	0,1007	0,15642
0,7	0,70393	0,77047
0,1	0,1108	-0,7179
0,1	0,10149	0,10107
0,1	0,10157	0,09433
0,1	0,1007	0,15642
0,7	0,70393	0,77047
0,1	0,10164	0,17738
0,7	0,69664	0,71944
0,7	0,71876	0,8669
0,7	0,70246	0,72123
0,1	0,10684	0,14291
0,7	0,70253	0,69037
0,7	0,68196	0,63889
0,1	0,10277	0,09429
0,1	0,10889	0,09453
0,4	0,23535	0,21153
0,3	0,36673	0,33865
СКО	0,015166364	0,109115

Для наглядности на рис. 3 были построены графики, отображающие результаты вычислений двух алгоритмов.

По полученным результатам видно, что оба алгоритма справляются хорошо. Если посчитать среднеквадратическое отклонение, то у алгоритма «коррекции радиусов дисперсий» отклонение будет равно 0,015, тогда как при использовании алгоритма «к-ближайших соседей» отклонение будет равно 0,11. Из этого можно сделать вывод, что можно получить более точные радиусы дисперсий.

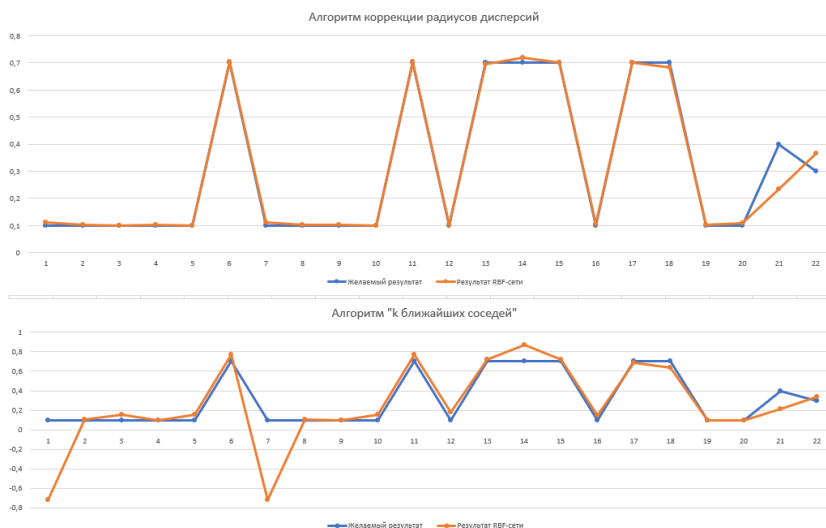


Рис. 3. Результаты вычислений двух алгоритмов

Заключение. Были исследованы RBF-сети, и на основе этих исследований был построен подход к определению радиусов дисперсий на основе алгоритма «к-ближайших соседей» с коррекцией полученных значений. Для того чтобы использовать данный подход, была написана программа, которая вычисляет необходимые параметры для модели RBF-сети. Благодаря данному подходу была разработана модель RBF-сети, используя написанную программу, которая позволяет распознавать изображения. Получившаяся модель отлично справляется в начале и в конце диапазона значений. Это связано с тем, что было подобрано недостаточно нейронов.

Библиографический список

1. Удалов А.Д., Архипов А.Н., Торхов А.Е. Аппроксимация методом радиально-базисных функций // Вестник современных исследований. – 2018. – № 5.3 (20). – С. 534–535.
2. Хижняков Ю.Н. Алгоритмы нечеткого, нейронного и нейро-нечеткого управления в системах реального времени: учеб. пособие. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2013. – 303 с.

3. Кушнир Д.А. Исследование и разработка нейросетевых методов анализа и обработки речевого сигнала в задаче распознавания речи. – М., 2006.

4. Ясницкий Л.Н. Интеллектуальные системы. – М.: Лаборатория знаний, 2016. – 221 с.

5. Терехов В.А. Нейросетевые системы управления. – М.: Высшая школа, 2002. – 183 с.

6. Коровин А.М. Интеллектуальные системы: текст лекций. – Челябинск: Издательский центр ЮУрГУ, 2015. – 60 с.

Сведения об авторах

Постных Денис Витальевич – бакалавр Пермского национального исследовательского политехнического университета, гр. АТ-22-1б, г. Пермь, e-mail: kfdfk123456789@gmail.com

Никулин Вячеслав Сергеевич – аспирант кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kalif23@yandex.ru

Д.Е. Копейкин, А.А. Петренко

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССА ПЕРЕДАЧИ КАРТОГРАФИЧЕСКИХ СЛОЕВ НА КАРТУ

В статье приведен анализ параметров слоев в геоинформационной системе. Рассмотрен процесс работы со слоями и параметрами слоев в геоинформационной системе. Проведено обоснование включения параметров обмена между слоями. Рассмотрены задачи и механизм работы микросервиса передачи информации, а также выявлены наиболее значимые этапы автоматизации процесса передачи картографических слоев.

Ключевые слова: программное обеспечение, картографический слой, географическая информационная система, представление данных.

D.E. Kopeykin, A.A. Petrenko

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF SOFTWARE TO AUTOMATE THE PROCESS OF TRANSFERING CARTOGRAPHIC LAYERS TO THE MAP

The article provides an analysis of the parameters of layers in a geoinformation system. The process of working with layers and layer parameters in a geoinformation system is considered. The substantiation of the inclusion of exchange parameters between layers is carried out. The tasks and the mechanism of operation of the micro-information transfer service are considered, and the most significant stages of automation of the process of transferring cartographic layers are identified.

Keywords: software, cartographic layer, geographic information system, data presentation.

В современном мире географические информационные системы (ГИС) стали незаменимым инструментом для организации, анализа и визуализации пространственных данных. Одним из ключевых элементов таких картографических приложений являются слои. Слои представляют собой наборы данных, отображаемые на карте в виде графических объектов. Каждый слой может содержать различные типы информации, такие как географические объекты (например, дороги, реки, границы), тематические данные (например, население, климати-

ческие зоны, экономические показатели) и многое другое. Благодаря слоям пользователи могут видеть различные аспекты информации на карте и анализировать их в контексте.

В компании «Брайт-Софт» разрабатывается географическая информационная система «Maplayers». Картографические слои хранятся в справочнике «Слой». Справочник представляет собой сущность, обеспечивающую хранение, редактирование и отображение данных из таблицы базы данных.

Слой имеет следующие атрибуты:

- 1) наименование – наименование слоя;
- 2) код – уникальный код слоя;
- 3) тип данных – тип геометрических данных слоя;
- 4) SQL фильтр – SQL запрос к MaplayersDB для получения геометрических данных. Запрос должен начинаться с ключевого слова 'SELECT';
- 5) атрибуты – наименования атрибутов, которые содержатся в SQL-фильтре;
- 6) дата создания – дата создания слоя;
- 7) версия витрины данных – элемент справочника «Версии витрин данных». Представляет собой описание источника данных, который содержит данные слоя.

На данный момент после создания слоя в справочнике «Слой» администратору необходимо вручную передавать данные слоя во внешние модули системы для вывода слоя на карту (рис. 1).

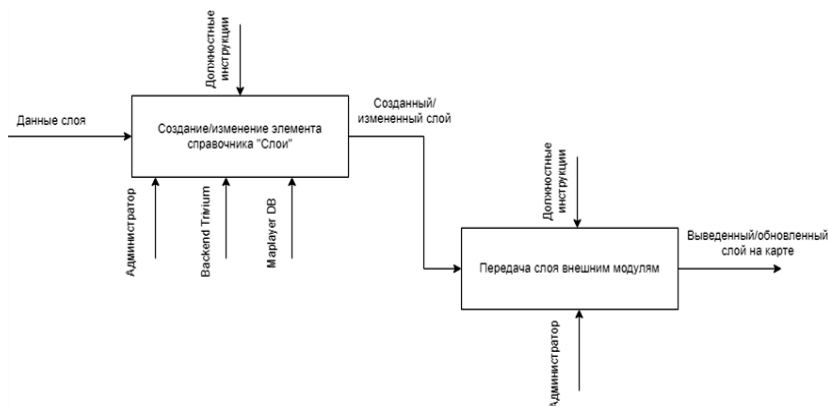


Рис. 1. Текущий процесс вывода слоя на карту

Для автоматизации процесса передачи слоя на карту необходимо модифицировать структуру слоя, добавив следующие атрибуты:

- 1) статус актуализации данных – признак того, что слой имеет новые данные и их необходимо обновить на карте;
- 2) статус публикации – признак того, что слой опубликован на карте;
- 3) состояние слоя – текущий статус слоя, означающий, является ли слой активным или нет.

Также необходимо создать новый справочник «Версии слоев», который будет содержать версии картографических слоев. Справочник должен содержать следующие атрибуты:

- 1) наименование – наименование версии слоя, которое имеет следующий формат: «Версия слоя <Наименование слоя> от <Дата создания версии слоя>»;
- 2) слой – элемент из справочника «Слои»;
- 3) дата создания – дата создания версии слоя;
- 4) JSON-версии – данные, которые хранит версия слоя: идентификатор; идентификатор версии; код; наименование; тип данных; состояние; SQL-фильтр; атрибуты;
- 5) статус нарезки – признак того, что версия слоя успешно подготовлена для вывода на карту. Может иметь значения «completed» или «error»;
- 6) ошибки при нарезке – хранит ошибки, если нарезка данных версии слоя прошла неуспешно;
- 7) дата нарезки – дата нарезки данных версии слоя.

Для окончательной автоматизации процесса необходимо разработать программное обеспечение, которое будет отслеживать изменения в справочнике слоев и на каждое изменение каких-либо данных элемента будет изменять статус актуализации данных слоя. Затем по изменению статуса актуализации слоя ПО будет создавать новую версию слоя. Для передачи слоя на карту внешние модули будут сами обращаться к разработанному программному обеспечению для получения новых или измененных версий слоев. Данные процессы продемонстрированы на рис. 2. Программное обеспечение разрабатывается согласно микросервисной архитектуре. Разработанные микросервисы будут встроены в существующую информационную систему. Для отслеживания изменений в справочнике «Слои» и создания новых версий слоев разработан микросервис `maplayers-provider`. Этот микросервис будет работать в фоне, незаметно для пользователя или администратора [1–4].

Основные задачи `maplayers-provider`:

- 1) отслеживать изменения в справочнике «Слои»;

- 2) изменять статус актуализации слоя в зависимости от изменения;
- 3) создавать новую версию слоя в зависимости от статуса актуализации слоя.

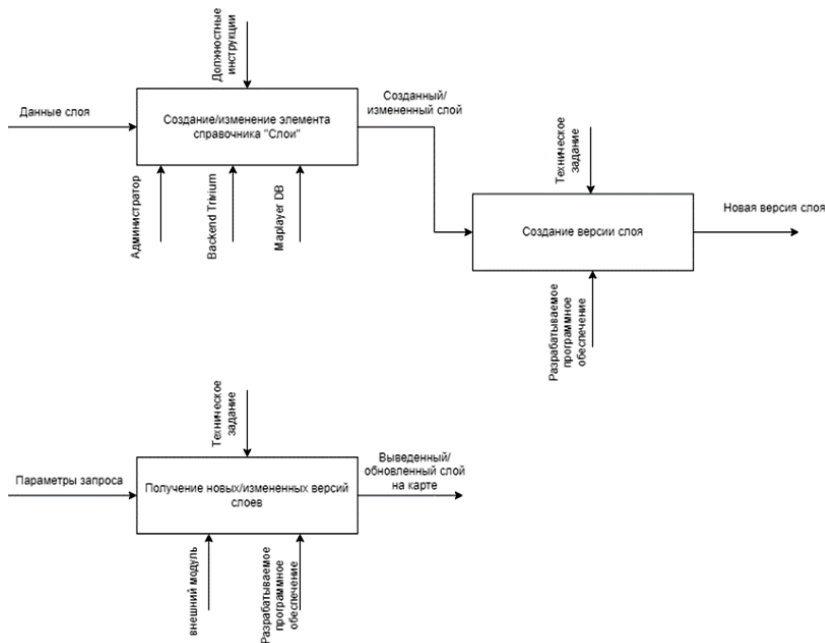


Рис. 2. Новые процессы для передачи картографического слоя на карту

Для передачи версий слоев внешнему модулю системы, который затем выведет слой на карту, разработан микросервис `maplayers-publisher`. Этот микросервис будет иметь API, к которому будет обращаться внешний модуль для получения новых или измененных версий слоев. Для этого необходимо реализовать HTTP-метод, который должен работать по принципу HTTP-ETag.

ETag (EntityTag) – это механизм HTTP, который позволяет серверу определять, изменилось ли содержимое запрашиваемого ресурса с момента предыдущего запроса клиента.

При использовании ETag сервер генерирует уникальный идентификатор (тег) для каждого ресурса, который отображает его текущее состояние. При запросе ресурса клиент может включить этот тег в заголовок запроса `If-None-Match`. Сервер проверяет этот тег и, если содержимое ресурса не изменилось с момента, когда был сгенерирован

этот тег, сервер возвращает код состояния 304 NotModified. В противном случае, если ресурс был изменен, сервер возвращает новое содержимое ресурса с соответствующим кодом состояния.

Используемые средства реализации, используемые при разработке ПО:

- 1) язык программирования C# – используется для основной разработки ПО, используется фреймворк ASP.NET;
- 2) СУБД PostgreSQL – используется для работы с базой данных;
- 3) ApacheKafka – брокер сообщений, используемый для отслеживания изменений в справочнике «Слой».

Таким образом, в статье проанализирован существующий процесс передачи картографических слоев на карту, описан алгоритм автоматизации текущего процесса [5]. При этом благодаря автоматизации процесса уменьшается количество рабочего времени, необходимого администратору на ручной вывод слоев на карту, что приносит значительные выгоды компании.

Библиографический список

1. Кристиан Хорсдал. Микросервисы на платформе .NET. – СПб.: Питер, 2018. – 352 с.
2. Jon Galloway, Phil Haack, Brad Wilson, K. Scott Allen. Professional ASP.NET MVC 3. – М.: Wiley Publishing, Inc, 2011. – 432 с.
3. Роберт Мартин. Чистый код: создание, анализ и рефакторинг. Библиотека программиста. – СПб.: Питер, 2016. – 464 с.
4. Гагарина Л.Г., Кокорева Е.В., Сидорова-Виснадул Б.Д. Технология разработки программного обеспечения: учеб. пособие. – М.: Форум, 2018. – 400 с.
5. Блиновская Я.Ю., Задоя Д.С. Введение в геоинформационные системы. – М.: Инфра-М, Форум, 2012. – 112 с.

Сведения об авторах

Копейкин Дмитрий Евгеньевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-1б, г. Пермь, e-mail: karew2284@gmail.com

Петренко Александр Анатольевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: hawk321@mail.ru

М.А. Шеретов, Д.В. Яруллин

Пермский национальный исследовательский
политехнический университет, Пермь

СОВРЕМЕННЫЕ МЕТОДЫ АНАЛИЗА И КЛАССИФИКАЦИИ КЛЕТОК КРОВИ: НЕОБХОДИМОСТЬ ИНТЕРПРЕТИРУЕМЫХ АЛГОРИТМОВ

В данной статье рассмотрены различные модели для анализа, подсчета и классификации клеток крови. Традиционные методы, такие как метод опорных векторов и преобразование Хафа, показали свою эффективность, но требуют дорогостоящего оборудования и сложных процедур преданализа, что затрудняет их применение в современных клиниках. Современные подходы с использованием сверточных нейронных сетей предлагают автоматизацию и снижение требований к преданализу за счет автоматического выделения признаков. Однако основной недостаток сверточных нейронных сетей заключается в сложности интерпретации результатов. Поднимается вопрос о необходимости разработки инструмента для интерпретации результатов работы сверточных нейронных сетей, выявления причинно-следственной связи.

Ключевые слова: классификация клеток крови, метод опорных векторов, преобразование Хафа, сверточные нейронные сети, преданализ.

M.A. Sheretov, D.V. Yarullin

Perm National Research Polytechnic University, Perm

MODERN METHODS OF BLOOD CELL ANALYSIS AND CLASSI- FICATION: THE NEED FOR INTERPRETABLE ALGORITHMS

This article reviews various models for analyzing, counting and classifying blood cells. Traditional methods, such as support vector methods and Hough transforms, have been shown to be effective but require expensive equipment and complex pre-analytic procedures, making them difficult to implement in modern clinics. Current approaches using convolutional neural networks offer automation and reduced pre-analytics requirements through automatic feature extraction. However, the main drawback of convolutional neural networks is the difficulty in interpreting the results. The question of the need to develop a tool for interpreting the results of convolutional neural networks and identifying causality is raised.

Keywords: blood cell classification, support vector method, Hough transform, convolutional neural networks, pre-analytics.

Существует множество моделей для анализа, подсчета и классификации клеток крови, каждая из которых имеет свои особенности

и области применения. Например, Lou и коллеги применяли метод опорных векторов на основе данных, полученных с помощью спектрального анализа, для точного подсчета эритроцитов [1]. Спектральный анализ позволяет получить высокоточные данные, однако этот метод требует дорогостоящего оборудования, недоступного большинству современных клиник. Таким образом, его использование ограничено специализированными лабораториями и крупными медицинскими центрами. В качестве альтернативы, Cruz и соавторы разработали систему, основанную на пороговом значении насыщенности оттенка и связанной маркировке, для независимого подсчета эритроцитов, лейкоцитов и тромбоцитов [2]. Этот подход менее затратен и может быть реализован с использованием стандартного лабораторного оборудования, что делает его более доступным для широкого круга медицинских учреждений. Кроме того, система Cruz и соавторов позволяет автоматизировать процесс подсчета клеток, снижая вероятность ошибок, связанных с человеческим фактором, и повышая общую эффективность работы лабораторий. Также Acharya и Kumar разработали систему подсчета эритроцитов, используя метод разделения вокруг медоида и преобразование Хафа [3]. Этот метод сочетает в себе кластеризацию данных и геометрический подход, что позволяет точно идентифицировать и подсчитывать эритроциты в образцах крови. Разделение вокруг медоида помогает сгруппировать похожие клетки, а преобразование Хафа используется для обнаружения и анализа круглых объектов, таких как эритроциты. Singhal и Singh занимались классификацией злокачественных и доброкачественных лейкоцитов, используя 256 локальных бинарных шаблонов [4]. Для улучшения результатов классификации они увеличили количество используемых шаблонов до 4096 [5]. Этот метод, основанный на локальных бинарных шаблонах (LBP), позволяет эффективно анализировать текстуры изображений клеток, что играет ключевую роль в распознавании и дифференциации типов лейкоцитов. Увеличение количества шаблонов с 256 до 4096 позволило значительно повысить точность классификации за счет более детального анализа структурных особенностей клеток. И наоборот, Bhattacharjee и Saini применили метод k-ближайших соседей (k-NN) для классификации лейкоцитов, используя всего 8 локальных бинарных шаблонов [6]. Этот минималистичный подход демонстрирует, что даже с ограниченным числом шаблонов можно достичь приемлемой точности классификации, благодаря эффективности алгоритма k-NN в анализе близости

данных. Метод k-NN работает на основе измерения сходства между образцами, что позволяет ему успешно классифицировать клетки, несмотря на небольшое количество используемых шаблонов. Такой подход может быть полезен в ситуациях, когда ресурсы ограничены или требуется быстрая и простая реализация. Помимо этого, использование малых наборов шаблонов снижает вычислительную нагрузку, что делает метод более доступным для широкого применения, включая реализацию в менее мощных компьютерных системах, таких как мобильные устройства или портативные анализаторы.

Ученые из Университета Джорджа Мейсона предложили использовать метод опорных векторов (SVM) для классификации клеток крови, опираясь на легко интерпретируемые признаки, извлеченные с помощью алгоритмов компьютерного зрения [7]. Этот подход позволяет эффективно анализировать изображения клеток, выделяя ключевые характеристики, которые существенно облегчают процесс классификации. Использование SVM в сочетании с алгоритмами компьютерного зрения обеспечивает высокую точность и надежность, поскольку метод опорных векторов хорошо работает с высокоразмерными данными и помогает находить оптимальные гиперплоскости для разделения классов. Алгоритмы компьютерного зрения, в свою очередь, позволяют автоматически выделять значимые признаки из изображений, что минимизирует необходимость в ручной обработке данных и снижает вероятность ошибок. Эти признаки могут включать такие параметры, как форма, размер, текстура и интенсивность пикселей, что дает возможность более детально анализировать клетки и улучшать точность диагностики. Более того, применение таких методов в медицинской практике способствует разработке автоматизированных систем, которые могут значительно ускорить и упростить процесс диагностики, делая его более доступным и надежным.

Но каждый из представленных учеными методов очень требователен к набору данных, на котором он производится, увеличивая требования к ресурсам для процедуры преданализики – сбору и окраске исходного материала, что делает их трудно воспроизводимыми. При нарушении четких правил пробоподготовки результат сильно снижается. Однако каждый из представленных учеными методов весьма требователен к набору данных, на котором он производится, что увеличивает требования к ресурсам для процедуры преданализики, включая сбор и окраску исходного материала. Это делает их трудновоспроизводи-

мыми и может значительно усложнить процесс подготовки образцов. Если четкие правила пробоподготовки нарушаются, точность и надежность результатов существенно снижаются. Процедура преданализики является критически важной, так как малейшие отклонения в сборе, обработке или окраске образцов могут привести к искажению данных, что негативно скажется на результатах анализа. Например, неправильная окраска клеток или использование некачественных реагентов может затруднить выделение значимых признаков и ухудшить качество классификации.

Одним из популярных методов анализа и классификации изображений, который способен снизить требования к этапу преданализики, является использование сверточных нейронных сетей (CNN) [8]. Этот тип модели вычислений широко применяется в анализе медицинских изображений [9, 10]. Сверточные нейронные сети способны автоматически выделять и определять важные признаки изображений, не требуя ручного определения характеристик [8]. Kutlu и его коллеги, а также Long исследовали применение различных архитектур сверточных нейронных сетей для классификации различных типов лейкоцитов [11, 12]. Существует множество различных архитектур для CNN, включая VGGNet [13], ResNet [14] и YOLO [15]. Трансферное обучение особенно полезно в случаях, когда у исследователя есть ограниченный набор данных или недостаточно обучающих примеров. При таком подходе сверточная нейронная сеть, сначала обученная на большом наборе данных, доучивается на задаче, которую исследует аналитик [8]. Это позволяет эффективно использовать знания, полученные из крупных наборов данных, для решения конкретной задачи классификации клеток крови, даже при ограниченном доступе к обучающим данным.

Один из основных недостатков использования сверточных нейронных сетей заключается в сложности интерпретации признаков, извлекаемых этими моделями, независимо от их архитектуры. Некоторые исследователи стремятся снизить количество признаков, используемых для классификации, и добиваются успеха в этом. Например, Sahlol и его коллеги применили метод трансферного обучения с моделью VGGNet в сочетании с улучшенным статистическим алгоритмом «Salp Swarm Algorithm» (SESSA) для извлечения более информативных признаков при классификации здоровых и злокачественных лейкоцитов [10]. В результате отбора было выбрано более 1000 признаков из 25 088 возможных, которые затем использовались в модели классификации на основе

метода опорных векторов [10–15]. Однако этот подход по-прежнему не обеспечивает достаточного представления о ключевых признаках, определяющих доброкачественные и злокачественные лейкоциты. Кроме того, данная система в основном ориентирована на классификацию лейкоцитов и не способна различить их от эритроцитов или тромбоцитов. Это указывает на необходимость дальнейших исследований для разработки более эффективных методов анализа и классификации клеток крови, способных учитывать различные типы клеток и их характеристики.

В заключение можно отметить, что анализ, подсчет и классификация клеток крови остаются сложными задачами, требующими высокоточных и эффективных методов. Традиционные подходы, такие как метод опорных векторов, пороговое значение насыщенности оттенка и преобразование Хафа, хоть и эффективны, но часто требуют дорогостоящего оборудования и сложных процедур преданализики. Современные методы, такие как сверточные нейронные сети, предлагают более гибкие и автоматизированные решения, способные снизить требования к этапу преданализики за счет автоматического выделения признаков. Однако основным недостатком использования сверточных нейронных сетей является сложность интерпретации результатов их работы. Несмотря на достижения в области трансферного обучения и использования различных архитектур сверточных нейронных сетей, таких как VGGNet, ResNet и YOLO, остаются проблемы с выделением и интерпретацией ключевых признаков, особенно при классификации различных типов клеток крови.

Необходимо разработать алгоритмы, которые бы позволили интерпретировать результаты работы сверточных нейронных сетей, обеспечивая более четкое понимание важных признаков, используемых для классификации клеток. Такие алгоритмы способствовали бы дальнейшим исследованиям в области классификации клеток в пробах жидкостей человека. В целом, дальнейшие исследования и разработки необходимы для создания более точных, доступных и интерпретируемых методов анализа клеток жидкостей человека. Это позволит улучшить диагностику заболеваний и развить более эффективные методы лечения.

Библиографический список

1. An automatic red blood cell counting method based on spectral images / J. Lou, M. Zhou, Q. Li, C. Yuan, H. Liu // 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and

Informatics (CISP-BMEI), Datong, China. – 2016. – P. 1391–1396. DOI: 10.1109/CISP-BMEI.2016.7852934

2. Determination of blood components (WBCs, RBCs, and Platelets) count in microscopic images using image processing and analysis / D. Cruz [et al.] // 2017 IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Manila, Philippines. – 2017. – P. 1–7. DOI: 10.1109/HNICEM.2017.8269515

3. Acharya V., Kumar P. Identification and red blood cell automated counting from blood smear images using computer-aided system // *Med Biol Eng Comput.* – 2018. – 56. – P. 483–489. DOI: 10.1007/s11517-017-1708-9

4. Singhal V., Singh P. Local Binary Pattern for automatic detection of Acute Lymphoblastic Leukemia // 2014 Twentieth National Conference on Communications (NCC), Kanpur, India. – 2014. – P. 1–5. DOI: 10.1109/NCC.2014.6811261

5. Singhal, V., Singh, P. Texture Features for the Detection of Acute Lymphoblastic Leukemia // S. Satapathy, A. Joshi, N. Modi, N. Pathak (eds). *Proceedings of International Conference on ICT for Sustainable Development. Advances in Intelligent Systems and Computing.* – 2016. – Vol. 409. Springer, Singapore. DOI: 10.1007/978-981-10-0135-2_52

6. Bhattacharjee R., Saini L.M. Robust technique for the detection of Acute Lymphoblastic Leukemia // 2015 IEEE Power, Communication and Information Technology Conference (PCITC), Bhubaneswar, India. – 2015. – P. 657–662. DOI: 10.1109/PCITC.2015.7438079

7. Lamberti W.F. Blood cell classification using interpretable shape features: A Comparative study of SVM models and CNN-Based approaches // *Computer Methods and Programs in Biomedicine Update.* – 2021. – Vol. 1. – P. 100023. DOI: 10.1016/j.cmpbup.2021.100023

8. Recent advances in convolutional neural networks / Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, Tsuhan Chen // *Pattern Recognition.* – 2018. – Vol. 77. – P. 354–77. DOI: 10.1016/j.patcog.2017.10.013

9. Deep learning-enabled medical computer vision / A. Esteva, K. Chou, S. Yeung [et al.] // *npj Digit. Med.* – 2021. – 4 (1):5. DOI: 10.1038/s41746-020-00376-2

10. Sahlol A.T., Kollmannsberger P., Ewees A.A. Efficient Classification of White Blood Cell Leukemia with Improved Swarm Optimization of

Deep Features // Sci. Rep. – 2020. – 10. – P. 2536. DOI: 10.1038/s41598-020-59215-9

11. Hüseyin Kutlu, Engin Avci, Fatih Özyurt. White blood cells detection and classification based on regional convolutional neural networks // Medical Hypotheses. – 2020. – Vol. 135. – P. 109472. DOI: 10.1016/j.mehy.2019.109472

12. A capsule network based model for the multiclassification of human peripheral blood cells / Fei Long, Jing-Jie Peng, Weita Song, Xiaobo Xia, Jun Sang, BloodCaps // Computer Methods and Programs in Biomedicine. – 2021. – Vol. 202. – P. 105972. DOI: 10.1016/j.cmpb.2021.105972

13. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556 [cs] (2015). ArXiv: 1409.1556, <http://arxiv.org/abs/1409.1556>.

14. Deep Residual Learning for Image Recognition / K. He, X. Zhang, S. Ren, J. Sun // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA. – 2016. – P. 770–778. DOI: 10.1109/CVPR.2016.90

15. Alam M.M., Islam M.T. Machine learning approach of automatic identification and counting of blood cells // Healthc Technol Lett. – 2019. – № 6 (4). – P. 103–108. DOI: 10.1049/htl.2018.5098

Сведения об авторах

Шеретов Марк Алексеевич – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ8-23-1м, г. Пермь, e-mail: mark.sheretov@gmail.com

Яруллин Денис Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: d.v.yarullin@ya.ru

Е.М. Самойлова, А.Д. Шашаткин

Саратовский государственный технический
университет им. Ю.А. Гагарина, Саратов

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ОБЕСПЕЧЕНИЯ СВЯЗИ ПО ПРОТОКОЛУ MODBUS ДЛЯ GD32F105HV

Данная статья посвящена разработке программной реализации обеспечения связи по MODBUS протоколу для микроконтроллера GD32F105HV, который представляет собой альтернативу контроллерам STM32 в рамках стратегии импортозамещения. Программный пакет предназначен для обеспечения связи по RS-232 и подобным линиям связи с использованием протокола MODBUS. В работе описывается архитектура и основные функции разработанного драйвера, включая функции работы в режимах ведущего и ведомого устройства, обработку прерываний USART, а также методы обмена данными по MODBUS.

Ключевые слова: MODBUS, GD32F105HV, STM32, RS-232, MODBUS драйвер, встраиваемые системы.

E.M. Samoilova, A.D. Shashatkin

Saratov State Technical University named after
Gagarina Yu.A., Saratov

SOFTWARE IMPLEMENTATION OF MODBUS COMMUNICATION FOR GD32F105HV

This article is devoted to the development of a software implementation of the MODBUS communication protocol for the GD32F105HV microcontroller, which is an alternative to STM32 controllers as part of the import substitution strategy. The software package is designed to provide communication over RS-232 and similar communication lines using the Modbus protocol. The paper describes the architecture and main functions of the developed driver, including the functions of operation in master and slave modes, processing of USART interrupts, as well as methods of data exchange via MODBUS.

Keywords: MODBUS, GD32F105HV, STM32, RS-232, MODBUS driver, embedded systems.

Выбор языка программирования и среды разработки играет ключевую роль в процессе создания встроенных систем. Для программной реализации MODBUS driver микроконтроллера GD32F105HV был выбран язык программирования «C» и среда разработки Keil μ Vision IDE. Язык «C» обладает низкоуровневым доступом к аппаратным ресурсам, что идеально подходит для разработки прошивки встроенных систем, вклю-

чая микроконтроллеры GD32, обеспечивает эффективное использование памяти, высокую производительность и переносимость кода [1, 2].

Среда разработки Keil μ Vision IDE, выбранная для реализации проекта, предоставляет удобный интерфейс и мощные инструменты для разработки встроенных систем. Ее функциональность включает в себя возможности отладки с элементами трассировки, интеграцию с аппаратными отладчиками и поддержку различных семейств микроконтроллеров, включая серию GD32. Преимущества использования Keil μ Vision IDE включают в себя также интуитивно понятный интерфейс для настройки и управления проектами, встроенную поддержку микроконтроллеров семейства GD32, включая GD32F105HV, широкую базу готовых библиотек и примеров кода для быстрого старта разработки [3].

В качестве альтернатив среды Keil μ Vision IDE рассматривались IAR Embedded Workbench, Atollic TrueSTUDIO и MPLAB X IDE. Atollic TrueSTUDIO и MPLAB X IDE являются относительно новыми средами разработки, функционал которых постоянно дополняется и обновляется. Для использования большинства функций IAR Embedded Workbench требуется лицензия, что может вызвать ограничения и затруднения для создания программного пакета MODBUS drive» [4].

Для успешного планирования разработки и дальнейшего сопровождения драйвера MODBUS необходимо создать архитектуру проекта, обеспечивающую его переносимость и легкость интеграции в другие системы [5]. Архитектура (рис. 1) включает в себя ключевой компонент «Инициализация системы», который является входной точкой для работы с драйвером. От этого компонента зависят остальные модули, такие как «Настройка USART», «Обработка данных MODBUS» и «Задачи FreeRTOS». Разрабатываемая структура позволяет абстрагироваться от конкретной реализации и гибко управлять порядком выполнения задач. При запуске основной функции происходит инициализация всех основных компонентов системы: глобальные параметры, настройка тактирования, инициализация USART и создание задач FreeRTOS для обработки данных по протоколу MODBUS. В каждый промежуток времени или при каждом кадре обрабатываются входные данные от USART, выполняется проверка CRC (контрольной суммы) и обновляются соответствующие регистры. При получении новых команд происходит их обработка и отправка ответных данных по USART. При особых событиях, таких как получение нового запроса или завершение передачи данных, также происходит обновление состояния системы и соответствующих регистров.

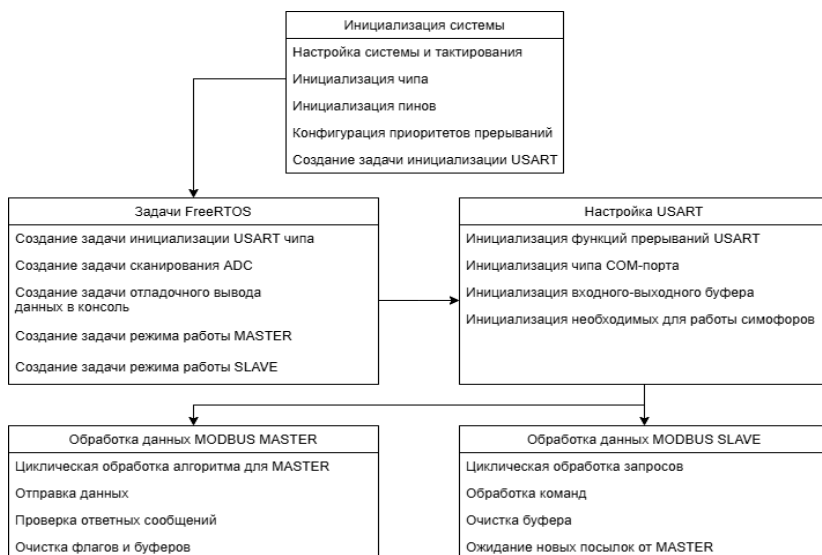


Рис. 1. Архитектура системы драйвера

Так как язык «С» не поддерживает объектно-ориентированное программирование, было применено функциональное, причем все функции расположены по разным файлам в зависимости от смысловой связанности между ними.

Разработанный программный пакет (рис. 2) содержит директории хранения сервисного кода: Firmware, Utilities и т.д., а также файлы `gd32f10x_libopt.h`, `gd32f10x_it.h` и `stdlib.h` конфигурации микроконтроллера GD32F105HV.

Файл `main.c` – содержит основную функцию входа и старта программного обеспечения MODBUS-драйвера. Также в нем реализованы стартовые функции задач системы реального времени FreeRTOS.

Файл `modbus_driver.h` является заголовочным для функций MODBUS.

Файл `USART_modbus_driver.c` содержит в себе код, отвечающий за обработку прерываний по USART и настройку COM-портов для использования RS-232.

Файл `USART_modbus_driver.h` является заголовочным для `USART_modbus_driver.c`.

В рамках работы с логикой в режиме MODBUS MASTER разработан алгоритм формирования и отправки запросов чтения/записи данных, соответствующих протоколу MODBUS (рис. 3).

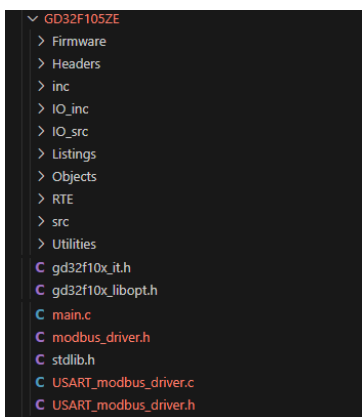


Рис. 2. Структура разработанного программного пакета MODBUS driver

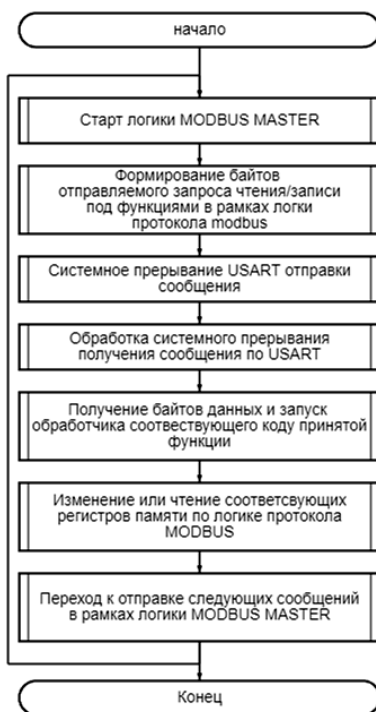


Рис. 3. Алгоритм работы режима MASTER драйвера MODBUS

Эти запросы отправляются через USART, и по мере необходимости они могут вызывать системные прерывания для обработки отправки и получения сообщений. После получения байтов данных через USART происходит обработка этих сообщений в соответствии с логикой протокола MODBUS, где определяется необходимость изменения или чтения соответствующих регистров памяти устройства. Этот процесс непрерывный, где после обработки одного сообщения происходит переход к формированию и отправке следующих запросов. Цикл работы продолжается до завершения задачи или получения команды остановки.

Необходимо отметить, что логика MODBUS MASTER не включена непосредственно в драйвер, а представляет собой алгоритмическую логику программ, которые будут использовать MODBUS драйвер для связи с внешними устройствами. Сам драйвер обеспечивает необходимые инструменты и функции для реализации этой логики в рамках протокола MODBUS.

Работа модуля драйвера MODBUS для GD32 в режиме SLAVE, начинается с обработки системного прерывания USART, вызванного получением сообщения от MASTER устройства по COM-порту (рис. 4).



Рис. 4. Алгоритм работы режима SLAVE драйвера MODBUS

Библиографический список

1. Смит Р.С. Программирование встроенных систем на языке С. O'Reilly Media, 2016.
2. Аяла Дж. Язык программирования С. – Prentice Hall, 1988.
3. Ибрагим Доган. Овладение STM32. – Packt Publishing, 2019.
4. Лью С.С. Основы программного обеспечения для встроенных систем. – Springer, 2006.
5. Треттер С. Архитектура встроенных систем: Подробное руководство для инженеров и программистов. – Newnes, 2012.
6. Организация Modbus. Спецификация и руководство по реализации MODBUS по последовательной линии. – Modbus Organization, 2002.

Сведения об авторах

Самойлова Елена Михайловна – кандидат технических наук, доцент кафедры «Техническая механика и мехатроника» Саратовского государственного технического университета им. Ю.А. Гагарина, г. Саратов, e-mail: helen_elenka@mail.ru

Шашаткин Антон Дмитриевич – магистрант кафедры «Техническая механика и мехатроника» Саратовского государственного технического университета им. Ю.А. Гагарина, гр. м-МХРТ-11, г. Саратов, e-mail: shashatkin@mail.ru

А.С. Ушаков, О.И. Мухин

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА АРХИТЕКТУРЫ АВТОНОМНОЙ МНОГОАГЕНТНОЙ СИСТЕМЫ БПЛА ДЛЯ ПОИСКА ЗАДАННЫХ ОБЪЕКТОВ НА МЕСТНОСТИ

Беспилотные летательные аппараты находят широкое применение в различных областях, но их использование в группах позволяет более эффективно решать разнообразные задачи. В данной статье представлена архитектура прототипа автономной многоагентной системы беспилотных летательных аппаратов, используемая для решения задачи поиска объектов на местности. Основная задача агентов – посетить все цели (контрольные точки) и сделать снимки местности. Описываются подходы к управлению группой объектов – централизованный и децентрализованный и способ оптимального представления местности для агентов. Показана разработанная архитектура: схема системы и базы данных.

Ключевые слова: беспилотные летательные аппараты, многоагентная система, поиск объектов, архитектура, дроны, автономность.

A.S. Ushakov, O.I. Muhin

Perm National Research Polytechnic University, Perm

ARCHITECTURE DEVELOPMENT OF AUTONOMOUS MULTI-AGENT UAV SYSTEM FOR SEARCH OF SPECIFIED OBJECTS ON THE AREA

Unmanned Aerial Vehicles have a wide range of applications in various fields, but their use in groups allows them to solve a variety of tasks more efficiently. This paper presents a prototype architecture of an autonomous multi-agent UAV system used to solve the problem of searching for objects on the terrain. The main task of the agents is to visit all targets (control points) and take images of the terrain. The approaches to controlling a group of objects – centralized and decentralized and the way of optimal terrain representation for agents are described. The developed architecture is shown: system scheme and database.

Keywords: UAV, multi-agent system, object search, architecture, drones, autonomy.

В последние годы беспилотные летательные аппараты (БПЛА) стали неотъемлемой частью как гражданских, так и военных приложений. Современные достижения в области электроники, вычислительной техники, коммуникаций и систем управления значительно расширили воз-

возможности использования БПЛА. Они активно применяются для мониторинга территорий, картографирования, поиска объектов и выполнения множества других задач [1]. Низкая стоимость производства и отсутствие риска для жизни пилота делают БПЛА более выгодными и безопасными по сравнению с пилотируемыми летательными аппаратами [2].

Технологии группового управления БПЛА, или многоагентные системы, позволяют не только более эффективно покрывать большие территории, но и сохранять работоспособность при выходе из строя одного или нескольких агентов, что существенно увеличивает надежность операций. Несмотря на значительный прогресс в данной области, существует ряд нерешенных проблем, которые ограничивают их применимость. Одна из этих проблем – автономность и самоуправление. В большинстве своем современные системы часто полагаются на централизованное управление с вмешательством оператора [3, 4].

В данной статье рассматривается архитектура автономной многоагентной системы БПЛА, для работы которой не требуется присутствие человека. Автономность будет обеспечиваться за счет децентрализованного управления группой БПЛА, а также увеличения отказоустойчивости серверной части системы. Конкретная реализация взаимодействия группы агентов остается за границей исследования.

Представление местности. В работе используется метод клеточной декомпозиции, который позволяет разбить исследуемую территорию на сектора, что упрощает задачи навигации и координации.

Этот метод описан в работе Puente-Castro [5].

На рис. 1 представлен пример такой декомпозиции, где каждый сектор имеет уникальные координаты.

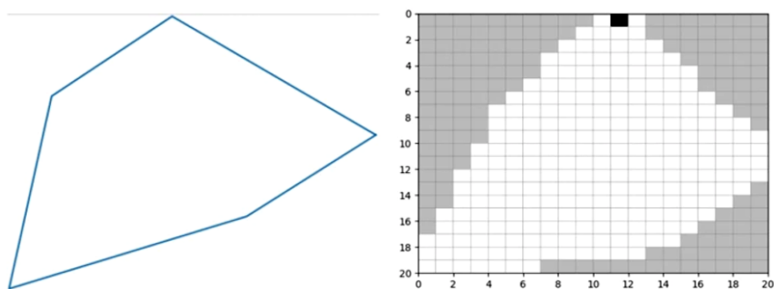


Рис. 1. Представление местности

Каждый сектор имеет следующие параметры: координаты центра, размер, статус посещения. Представленная таким образом территория может быть преобразована и сохранена в базе данных.

Управление агентами. Для эффективного выполнения разного спектра задач в системе предлагается использование нескольких методов координации и взаимодействия между агентами:

1. Централизованное следование заранее определенному маршруту. Для него требуется задание маршрута оператором, он не автономен. Также можно рассчитать маршрут для каждого агента в зависимости от его характеристик [5].

2. Роевой децентрализованный подход. Он позволяет агентам гибко реагировать на изменения в окружающей среде, взаимодействовать друг с другом и на основе этого менять свой путь. В этом подходе (рис. 2) каждый агент обладает ключевыми параметрами: R_{min} (минимальное расстояние между агентами, чтобы избежать столкновений) и R_{net} (расстояние, на которой осуществляется связь между агентами) [6, 7].

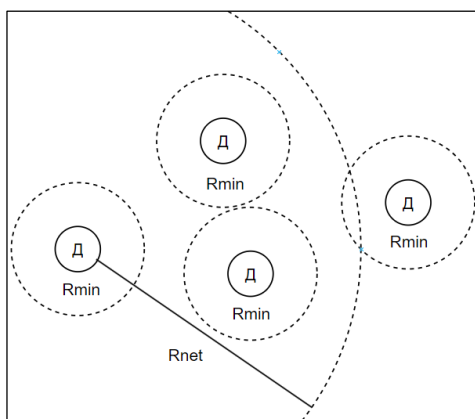


Рис. 2. Модель децентрализованного метода управления

При таком методе агенты постоянно обмениваются данными о своем состоянии, такими как координаты (x , y , z), скорость (u), и ускорение (a). Также каждому агенту нужна информация о местности и целевые (контрольные) точки – центры секторов [8]. Его преимущества:

– локальная адаптивность. Агентам предоставляется информация, наиболее актуальная для их текущего местоположения и окружения;

– масштабируемость. Группа БПЛА легко адаптируется к увеличению числа агентов, так как каждый агент взаимодействует только с ограниченным числом соседей.

Детектирование объектов. Для детектирования объектов в предлагаемой системе будет использоваться нейронная сеть YOLOv8, модифицированная китайскими исследователями для улучшенного обнаружения малых объектов [9]. Эта модификация играет важную роль при поиске объектов на изображениях с БПЛА. В качестве базового датасета для начального обучения используются данные из источника Roboflow, предназначенного для поиска людей [10].

Элементы системы и их взаимодействие. Архитектура системы предусматривает модульный подход. Система включает следующие модули:

1. Модуль связи с агентами. Обеспечивает соединение с каждым агентом для получения и передачи данных и команд. Конкретная реализация протокола связи не определена.

2. Модуль обработки изображений. Обеспечивает функционал детектирования объектов (основной потребитель ресурсов). Использует нейронную сеть YOLOv8 для детектирования объектов на изображениях. Реализуется с использованием библиотек OpenCV и PyTorch для языка Python.

3. Модуль взаимодействия с хранилищем и база данных. Отвечает за хранение обработанных изображений, информации об агентах, миссиях и местностях. Реализуется с помощью PostgreSQL, библиотеки для работы с БД (SQLAlchemy). Для гарантии бесперебойной работы применяется репликация.

4. Модуль взаимодействия с пользователем. Предоставляет интерфейс для взаимодействия с системой. Является веб-интерфейсом на фреймворке Django. На рис. 3 представлена схема архитектуры системы, демонстрирующая взаимодействие между различными модулями.

База данных системы состоит из 9 сущностей. Они позволяют структурировать информацию о миссиях, агентах, объектах и других ключевых элементах системы: местность (area), миссия (mission), шаблон миссии (mission_template), история миссии (mission_history), агент (agent), статус миссии (status), изображение (image), объект (object), тип объекта (object_type). Физическая модель базы данных в нотации Мартина представлена на рис. 4.

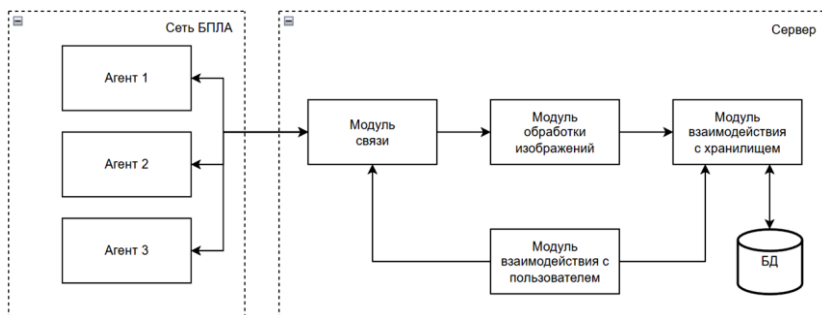


Рис. 3. Архитектура системы

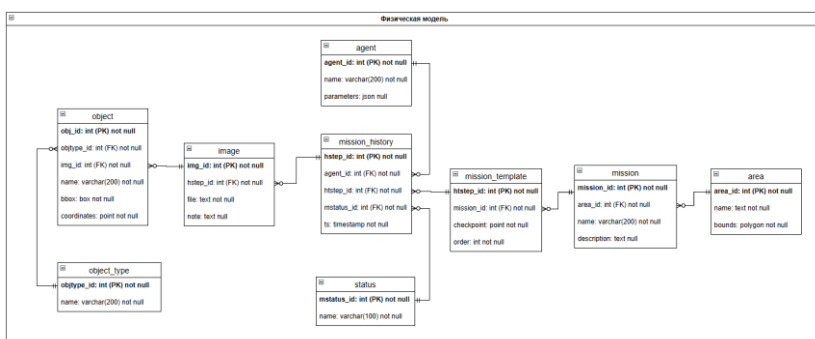


Рис. 4. Физическая модель базы данных

Каждый агент действует по заранее определенному алгоритму, однако в процессе работы может менять свой маршрут в зависимости от условий среды, таких как: потеря связи с соседями, появление препятствия, потеря связи с центральным узлом (сервером).

На рис. 5 демонстрируется пример алгоритма функционирования отдельного агента в составе группы.

Сначала агент получает данные о миссии и типе взаимодействия: заранее определенный маршрут или работа в роевом режиме. В зависимости от этих данных строится начальный маршрут. Во время движения по маршруту агент делает снимки в контрольных точках и отправляет изображения на сервер или сохраняет их на локальном устройстве. При изменении условий осуществляется попытка перестроить маршрут. По завершении пути агент возвращается в начальную точку.

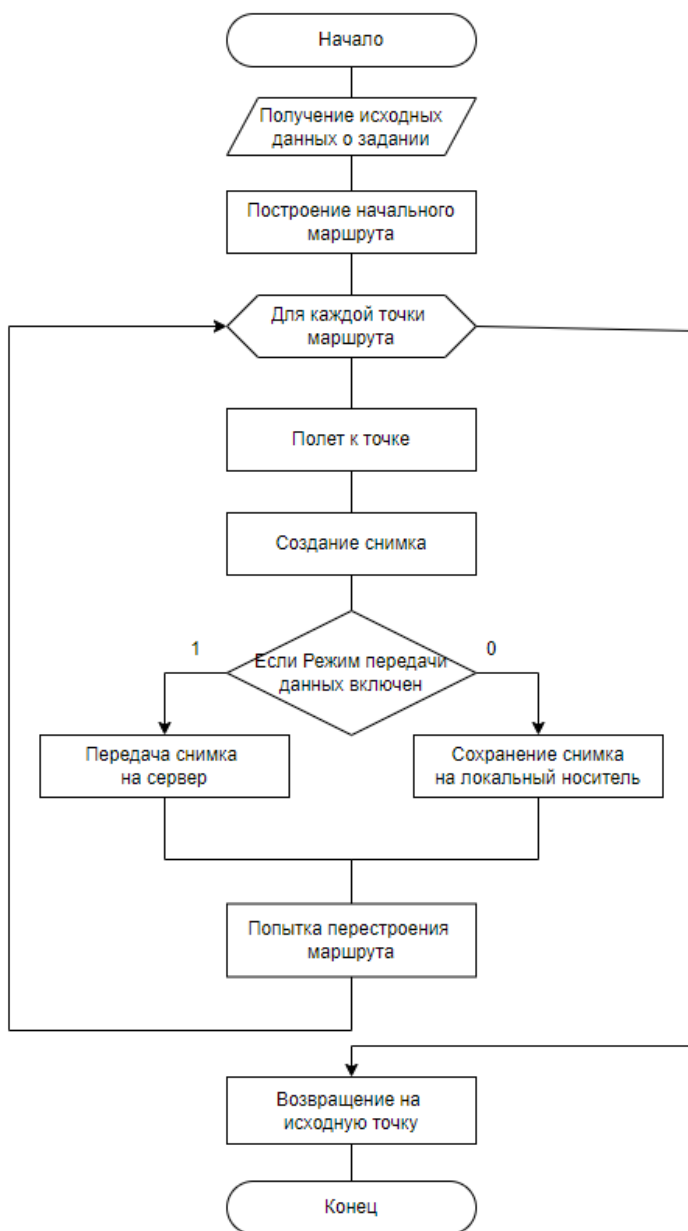


Рис. 5. Алгоритм действий агента в составе группы

Заключение. В результате исследования была разработана архитектура прототипа автономной многоагентной системы БПЛА. Основной упор сделан на обеспечение автономности системы без необходимости участия оператора. Дальнейшее развитие и оптимизация данной архитектуры могут привести к созданию более эффективных систем.

Библиографический список

1. Yan Z., Jouandeau N., Cherif A.A. A survey and analysis of multi-robot coordination // *International Journal of Advanced Robotic Systems*. – 2013. – Vol. 10, № 12. – P. 399.
2. Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage / M. Mozaffari [et al.] // *IEEE Communications Letters*. – 2016. – Vol. 20, № 8. – P. 1647–1650.
3. Configurations and applications of multi-agent hybrid drone/unmanned ground vehicle for underground environments: A review / C. Dinelli [et al.] // *Drones*. – 2023. – Vol. 7, № 2. – P. 136.
4. Лю В. Методы планирования пути в среде с препятствиями (обзор) [Электронный ресурс] // Математика и математическое моделирование. – 2018. – № 1. – URL: <https://cyberleninka.ru/article/n/metody-planirovaniya-puti-v-srede-s-prepyatstviyami-obzor> (дата обращения: 15.05.2024).
5. UAV swarm path planning with reinforcement learning for field prospecting / Puente-Castro A. [et al.] // *Applied Intelligence*. – 2022. – Vol. 52, № 12. – P. 14101–14118.
6. Иванов Д.Я. Использование принципов роевого интеллекта для управления целенаправленным поведением массово-применяемых микророботов в экстремальных условиях // *Известия высших учебных заведений*. – 2011. – № 9. – С. 70–78.
7. Перспективы развития многоагентных систем беспилотных летательных аппаратов / К.А. Окина, А.В. Козырь, Э.А. Гейс, О.О. Морозов // *Техника XXI века глазами молодых ученых и специалистов*. – 2023. – № 21. – С. 110–118.
8. Swarm of micro flying robots in the wild / X. Zhou [et al.] // *Science Robotics*. – 2022. – Vol. 7, № 66. – P. eabm5954.

9. DC-YOLOv8: small-size object detection algorithm based on camera sensor / H. Lou [et al.] // Electronics. – 2023. – Vol. 12, № 10. – P. 2323.

10. Датасет для обучения YOLOv8: people finder Computer Vision. [Электронный ресурс]. – URL: <https://universe.roboflow.com/danil-sxed4/people-finder-jwu6o> (дата обращения: 20.05.2024).

Сведения об авторах

Ушаков Андрей Сергеевич – магистрант Пермского национального исследовательского политехнического университета, гр. РИС-23-1м, г. Пермь, e-mail: leenzstra@gmail.com

Мухин Олег Игоревич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: moi@stratum.ac.ru

Г.С. Индюков

Пермский национальный исследовательский
политехнический университет, Пермь

**РАЗРАБОТКА ПРОГРАММНОГО МОДУЛЯ ИНТЕЛЛЕКТУАЛЬНОЙ
СИСТЕМЫ АВТОМАТИЗАЦИИ ДЛЯ ДЕТЕКТИРОВАНИЯ
АКТИВНОСТИ ОБОРУДОВАНИЯ НА ОСНОВЕ
АЛГОРИТМОВ НЕЙРОННЫХ СЕТЕЙ**

В данной статье представлен процесс разработки модуля интеллектуальной системы автоматизации для обеспечения оповещения оператора о прекращении работы оборудования путем распознавания остановки оборудования на видеоданных из системы видеонаблюдения. Применены методы оптимизации нейронной сети. Сравнение результатов на каждом этапе работы показало улучшение точности, а также уменьшение времени работы детектора, что способствует повышению уровня контроля за предприятием.

Ключевые слова: система автоматизации, видеонаблюдение, глубокое обучение, сверточные нейронные сети, распознавание, контроль оборудования, определение.

G.S. Indyukov

Perm National Research Polytechnic University, Perm

**DEVELOPMENT OF A SOFTWARE MODULE OF AN INTELLIGENT
AUTOMATION SYSTEM FOR EQUIPMENT ACTIVITY DETECTION
BASED ON NEURAL NETWORK ALGORITHMS**

This paper presents the process of developing an intelligent automation system module to provide operator notification of equipment shutdown by recognizing equipment shutdown on video data from the video surveillance system. Neural network optimization techniques have been applied. Comparison of the results at each stage of the work showed an improvement in accuracy, as well as a reduction in detector operation time, which contributes to improving the level of control over the enterprise.

Keywords: automation system, video surveillance, deep learning, convolutional neural networks, recognition, equipment control, definition.

В контексте обеспечения автоматизации на производстве была поставлена цель – разработать программный модуль интеллектуальной системы автоматизации для детектирования активности оборудования и оповещения оператора путем распознавания остановки оборудования на видеоданных из системы видеонаблюдения, работающей в режиме реаль-

ного времени. Было проанализировано несколько доступных решений, обеспечивающих необходимую функциональность. Также было произведено тестирование для оценки точности обнаружения, по итогам которого было выбрано наиболее удачное решение. Это решение основано на обученной архитектуре нейронной сети FlowNetS (далее – НС) [1–7].

Существующие наборы данных, для обучения НС имеют большое количество изображений, но содержат в себе ряд недостатков. Основным недостатком существующих данных является то, что они представляют собой синтетические данные, т.е. данные получены путем моделирования 3D объектов на сцене и смещения их на известное расстояние, вследствие чего формируется двумерная матрица векторов (оптический поток). Данный подход в формировании выборки является неудачным для реальных систем видеонаблюдения, так как кадры, поступающие с видеокамеры для анализа, отличаются от синтетических, вследствие чего снижается точность определения оптического потока. Пример изображения из набора данных представлен на рис. 1.

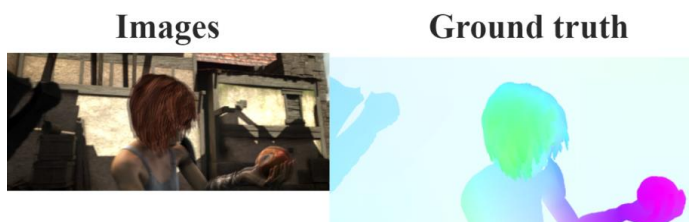


Рис. 1. Пример изображения и разметки из синтетического набора данных

Проведенные тесты показали, что НС, обученная на синтетических данных, имеет низкую точность определения в условиях работы на данных, полученных с камер видеонаблюдения.

Для увеличения точности распознавания было принято решение создать новый набор данных. Так как процесс формирования выборки для обучения нейросети для определения оптического потока на данный момент не имеет инструментов для точной разметки, а ручная разметка занимает большое количество времени, был разработан способ для полуавтоматической разметки данных. Тем самым новый набор данных должен быть более подходящим для обучения НС, способной решать поставленную задачу.

Основная часть. Первым этапом обучения НС является создание набора данных. Было разработано решение формировать выборку путем

полуавтоматической разметки. Полуавтоматическая разметка представляет собой определение оптического потока за счет заранее обученной нейросети высокой точности RAFT и последующий отбор пары изображений и оптического потока вручную. Нейросеть RAFT предоставляется на базе библиотеки компьютерного зрения OpenCV. RAFT обладает высокой точностью, но также имеет недостаток по времени обработки изображений, что не позволяет использовать ее в системах реального времени. Первым этапом формируется набор видео, снятых в реальных условиях. На втором этапе набор видео нарезается на изображения. На третьем этапе нарезанные пары изображений подаются в нейросеть RAFT для определения оптического потока. Заключительным этапом полученные пары изображений и оптический поток подаются на вход разработанного приложения для последующего ручного отбора данных, в которых оптический поток был определен наиболее точно. Пример отбора данных в приложении представлен на рис. 2.

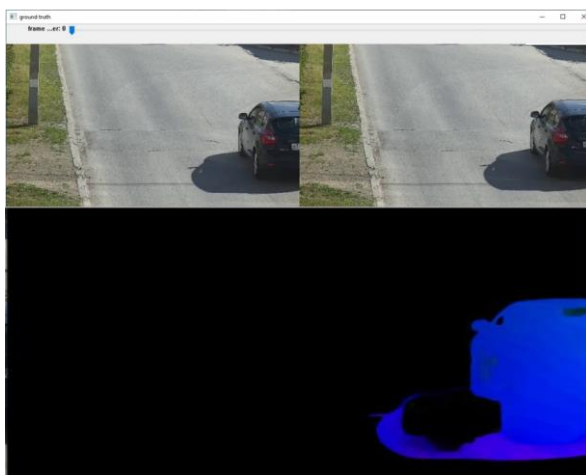


Рис. 2. Приложение для ручного отбора данных для обучения

Вторым этапом было произведено обучение НС. Данные, полученные путем полуавтоматической разметки, подаются на нейронную сеть парами для оценки смещения пикселя на изображении. Размер итоговой выборки составил около 8 тыс изображений, данный объем близок к минимально рекомендуемому для обучения нейронной сети. Далее выборка была разделена на обучающую и валидационную в соотношении семьдесят к тридцати. В качестве функции потерь было выбрано

евклидово расстояние между рассчитанным алгоритмом и истинным оптическим потоком, усредненным по всем пикселям (EPE). График обучения представлен на рис. 3.

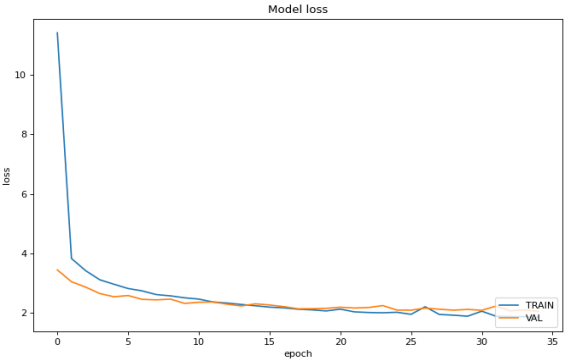


Рис. 3. Результат обучения нейронной сети

Результаты обучения нейронной сети на реальных и синтетических данных представлены в табл. 1. Оценка точности производилась на реальных данных, взятых с камер видеонаблюдения.

Таблица 1

Результаты обучения НС

Набор данных	Общее количество данных, тыс.	Трен-ая выборка тыс.	Валид-ая выборка тыс.	Ср. EPE, %
Flying chairs	23	16,1	6,9	97,31
Sintel	1	0,7	0,3	87,15
Реальные данные	8	5,6	2,4	97,38

Как видно из табл., разница между нейросетью, обученной на синтетических данных Flying chairs, и реальных данных составляет величину, которую можно охарактеризовать как погрешность, но стоит учитывать, что объем синтетических данных существенно больше. Несмотря на то, что архитектура FlowNetS обладает самым маленьким временем анализа на CPU 0,397 с, более современная модель FlowNet2 обладает временем анализа на CPU 1,547 с, а приведенная выше модель RAFT обладает временем анализа на CPU 1,624 с, данное значение все еще является критичным для систем реального времени. Для существенной оптимизации нейронной сети было принято решение использовать алгоритм квантования после обучения. Квантование – процесс уменьшения битности вычислений в нейронной сети. Квантование

осуществлено с изменением значений с float32 до float16, что позволило оптимизировать время анализа без критического ухудшения показателей точности. Оптимизация нейронной сети осуществляется при помощи TensorFlow Lite. Был разработан скрипт квантования модели после обучения, модель сохранена в формате .tflite. Результаты оптимизации нейронной сети представлены в табл. 2.

Таблица 2

Результаты квантования нейронной сети

Оптимизация модели	Размер модели, Мб	Ср. EPE, %	Время анализа на CPU, с
До квантования	151	97,38	0,397
После квантования	78	96,98	0,285

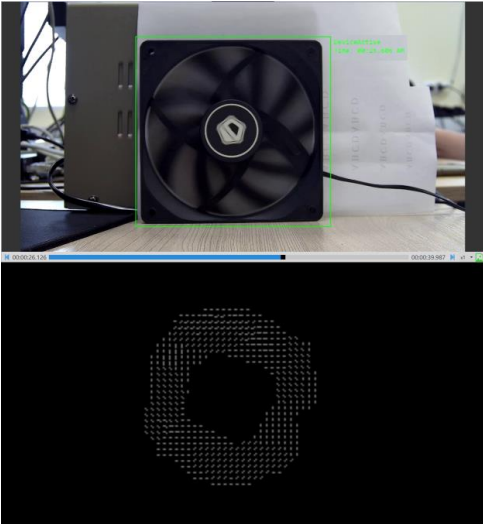


Рис. 4. Работа детектора контроля активности оборудования

После обучения нейросети был разработан алгоритм контроля активности оборудования. Оператор в приложении выбирает зону активности, в которую попадает движущаяся часть оборудования. Далее все изображение попадает для анализа в нейросеть, которая вычисляет оптический поток, после чего по оптическому потоку строится маска. По маске изображения происходит подсчет площади движущейся части. Если площадь маски соответствует условию чувствительности, а также сохраняется в течении заданного времени, то такое оборудование помечается как ак-

тивное, иначе как неактивное. Условия чувствительности и времени срабатывания заранее выбирается оператором. На рисунке ниже продемонстрирована работа детектора контроля активности оборудования рис. 4.

Заключение. Поставленная цель, согласно которой было необходимо разработать программный модуль интеллектуальной системы автоматизации, была выполнена с помощью создания набора данных и обучения НС. Также была произведена оптимизация нейронной сети на основе алгоритма квантования после обучения.

Библиографический список

1. Souhila K., Karim A. Optical flow-based robot obstacle avoidance // J. Advanced Robotic Systems. – 2007. – Vol. 4, no. 1. – P. 2.
2. Chang J., Tejero-de-Pablos A., Harada T., “Improved optical flow for gesture-based human-robot interaction // Proc. Int. Conf. Robotics Automation. – 2019. – P. 7983–7989.
3. Yang X., Kong L., Yang J. Unsupervised motion representation enhanced network for action recognition // Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process. – 2021.
4. Horn B.K.P., Schunck B.G. Determining optical flow // Artif. Intell. – 1981.
5. Brox T., Malik J. Large displacement optical flow: Descriptor matching in variational motion estimation // IEEE Trans. Pattern Anal. Mach. Intell. – 2011.
6. Zach C., Pock T., Bischof H. A duality-based approach for realtime tv-l1 optical flow // Pattern Recognition (F.A. Hamprecht, C. Schnorr, B. Jahne, eds.). – 2007.
7. Fast optical flow using dense inverse search / T. Kroeger, R. Timofte, D. Dai, L. Van Gool // Proc. Eur. Conf. Comp. Vis. (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.). – 2016.

Сведения об авторе

Индюков Глеб Сергеевич – магистрант Пермского национально-го исследовательского политехнического университета, гр. АТ-22-1м, г. Пермь, e-mail:indyukovgleb@yandex.ru

О.А. Полякова, А.А. Пантелеев

Пермский национальный исследовательский
политехнический университет, Пермь

**РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ОПОВЕЩЕНИЯ МЧС
О ЧРЕЗВЫЧАЙНЫХ СИТУАЦИЯХ**

В статье рассмотрены современные методы оповещения экстренной противопожарной службы о чрезвычайных ситуациях. Проведен анализ проблем, связанных с использованием «Системы 112», разработано мобильное приложение для их решения, доказана эффективность использования GPS для передачи информации пользователем.

Ключевые слова: мобильное приложение, экстренные службы, система 112, GPS.

O.A. Polyakova, A.A. Panteleev

Perm National Research Polytechnic University, Perm

**DEVELOPMENT OF SOFTWARE FOR THE MCHS
AUTOMATED EMERGENCY NOTIFICATION SYSTEM**

The article considers modern methods of notification of emergency firefighting service about emergency situations. The problems related to the use of "System 112" are analyzed, a mobile application for their solution is developed, the efficiency of using GPS to transmit information by the user is proved.

Keywords: mobile application, emergency services, 112 system, GPS.

В настоящее время вызов экстренных служб выполняется через звонок по закреплённому номеру 112, что не всегда удобно и не всегда реализуемо:

- фактор стресса человека приводит к невозможности вспомнить номер и правильно его ввести;
- отсутствие отслеживания статуса вызова служб;
- сложность указания адреса, если вызывающий не знает, где находится [1].

Одним из методов устранения вышеуказанных недостатков является разработка системы вызова экстренных служб с помощью мобильного приложения.

Именно поэтому замена традиционного метода вызова экстренных служб является актуальной задачей разработки.

Традиционный способ вызова пожарных бригад. Вызов пожарной службы осуществляется с помощью телефонного номера с 1930-х гг. по настоящее время. Алгоритм вызова предполагает:

- вызов по номеру 101 или 112 (единый номер вызова экстренных служб);
- операторы диспетчерского центра принимают вызов, обрабатывают запрос и координируют действия бригад экстренного реагирования;
- пожарная бригада выезжает на место происшествия в соответствии с указаниями диспетчера.

Вызов экстренных служб с помощью мобильного приложения. Вышеописанные проблемы решаются с помощью использования кросс-платформенного мобильного приложения, разработанного на framework .NET MAUI [2]. Влияние фактора стресса человека уменьшается с помощью автоматического ввода информации о происшествии через кнопку вызова, что позволяет не вводить номер телефона. Влияние человеческого фактора со стороны оператора уменьшается, благодаря автоматической записи поступающей информации от пользователя в базу данных. Также использование технологии GPS, которой оснащены современные мобильные устройства, позволяет отправлять точное местоположение пользователя или выбранного адреса из списка важных для пользователя адресов. Такая возможность делает точным и быстрым определение места возгорания по сравнению с получением данных экстренными службами от оператора, поскольку точность определения места, откуда был совершен вызов, равна 150–300 м в городе и 2–5 км в загородной зоне. С помощью использования GPS-системы такой разброс возможно уменьшить до 10 м.

Также при использовании мобильного приложения для системы автоматизированного вызова экстренных служб пользователь видит информацию на главном экране о статусе вызова. Сценарий вызова экстренной службы изображен на рис. 1–4.

Главный экран приложения с возможностью выбора места возгорания с помощью карты или из списка избранных адресов (рис. 1) представляет собой Яндекс-карту с определением местоположения пользователя. Для улучшения качества информированности о происшествии экстренной службы служит окно ввода уточняющего комментария от пользователя (рис. 2).

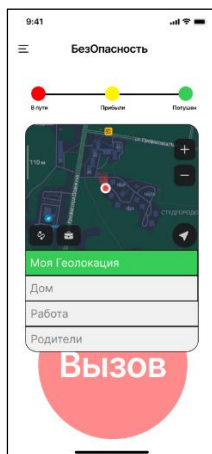


Рис. 1. Выбор места возгорания с помощью карты

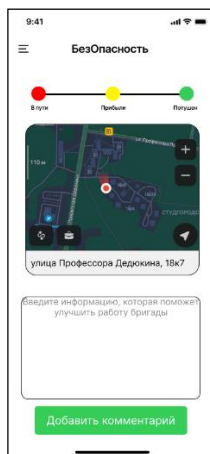


Рис. 2. Ввод комментария от пользователя

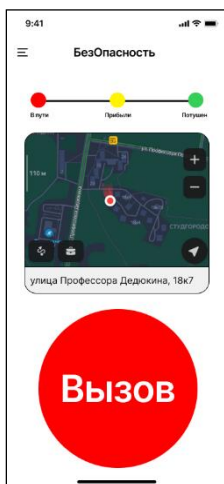


Рис. 3. Главный экран после ввода комментария и выбора адреса

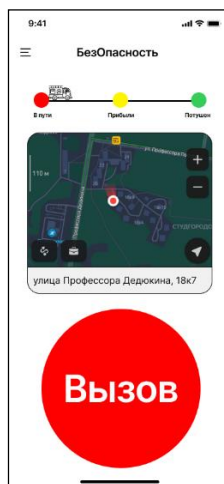


Рис. 4. Главный экран с отображением статуса вызова

Для отправки заявки оператору экстренной службы пользователь использует кнопку «Вызов» (рис. 3). Для отображения информации о статусе вызова открывается главный экран приложения, как только получен ответ пользователем от оператора (рис. 4).

Алгоритм вызова экстренных служб с помощью мобильного приложения. При проектировании бизнес-процесса вызова экстренных служб (рис. 5) разработана архитектура аккаунтов пользователей для автоматической передачи информации о каждом пользователе при вызове. Такой функционал разработан для уменьшения количества вводимой информации гражданином, что уменьшает интервал времени между запуском приложения и поступлением заявки оператору.

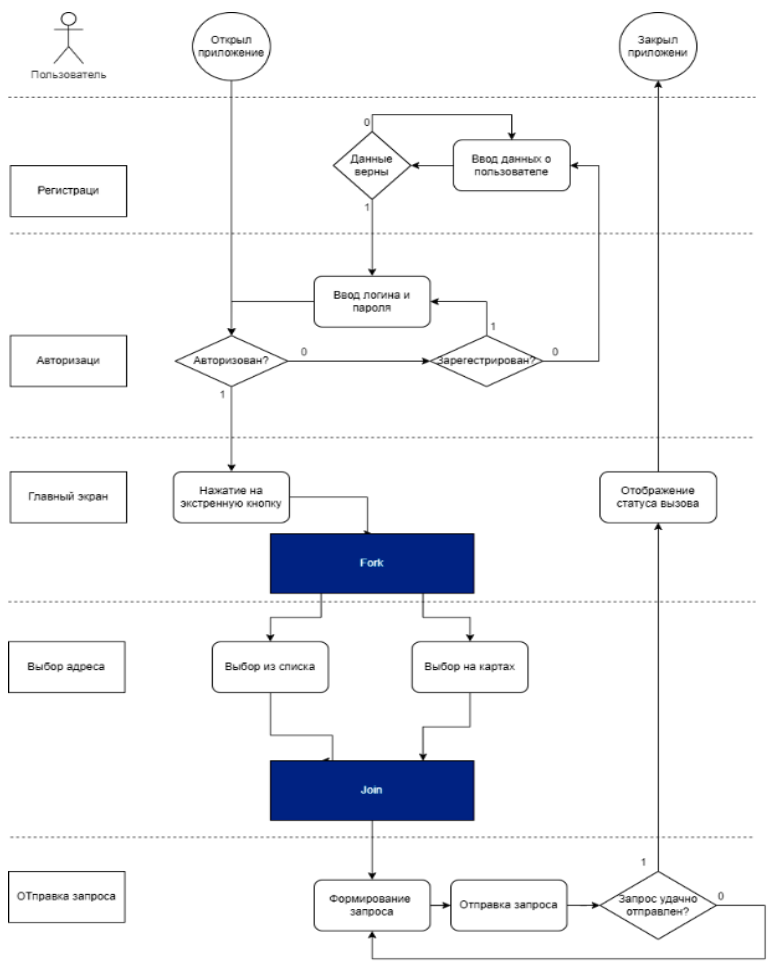


Рис. 5. Бизнес-процесс вызова МЧС

При открытии приложения происходит авторизация пользователя на сервере по сохраненным логину и паролю на его устройстве. Если данные о пользователе отсутствуют, то открывается окно авторизации, затем на главном экране отображается местоположение пользователя.

Бизнес-процесс вызова экстренных служб в нотации диаграмм активностей представлен на рис. 5, где последовательно показаны действия пользователя с учетом различных состояний аккаунта: авторизован, зарегистрирован и отсутствует аккаунт в системе. Также продемонстрированы повторные отправки запроса в случаях, когда сервер сообщает о неудачной передаче заявки оператору экстренной службы.

Также при разработке учтена вероятность отсутствия сети интернет и аккаунта в системе. Для таких случаев в алгоритме приложения использована «глубокая ссылка» на встроенное приложения «Телефон», в которое автоматически вставлен номер экстренной службы, что уменьшает влияние фактора стресса и временной промежуток между обнаружением угрозы и началом звонка.

Заключение. Использование мобильного приложения для вызова экстренных служб устраняет недостатки существующего метода и уменьшает время вызова, вероятность ошибки с определением адреса пользователя, автоматически формирует записи в базу данных заявок и позволяет гражданам отслеживать статус вызова экстренных служб.

Библиографический список

1. Документы МЧС [Электронный ресурс]. – URL: <https://mchs.gov.ru/dokumenty> (дата обращения: 28.05.2023).
2. Документация по .NET MAUI [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/dotnet/maui/> (дата обращения: 28.05.2023).

Сведения об авторах

Полякова Ольга Андреевна – доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: olgastratum@mail.ru

Пантелеев Артем Алексеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: brkotik01@gmail.com

А.К. Усманов, Д.А. Карлов

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ПРОЕКТА РОБОТЕХНИЧЕСКОГО КОМПЛЕКСА ДЛЯ ДВИЖЕНИЯ ПО ТРУДНОПРОХОДИМОЙ МЕСТНОСТИ

Исследуется возможность построения вездехода, способного перемещаться как по земным, так и по водным поверхностям. Водоизмещение, жесткость конструкций, плавательная система, гидроизоляция корпуса и программная среда стали основой проекта. Рассматриваются способы конструирования такого вездехода. Показан способ достижения прочного и плавучего корпуса, а также системы его передвижения.

Ключевые слова: водоизмещение, вездеход-амфибия, жесткость конструкций, Raspberry Pi, гидроизоляция.

A.K. Usmanov, D.A. Karlov

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A PROJECT FOR A ROBOTIC COMPLEX FOR DRIVING ON DIFFICULT TERRAIN

This article explores the possibility of building an all-terrain vehicle that can move on both land and water surfaces. Displacement, structural rigidity, swimming system, hull waterproofing and software environment became the basis of my project. This article discusses how to design such an all-terrain vehicle. A method for achieving a strong and buoyant hull, as well as its movement system, is shown.

Keywords: displacement, amphibious all-terrain vehicle, structural rigidity, Raspberry Pi, waterproofing.

В современном мире существует потребность в универсальном транспорте, который способен передвигаться как по суше, так и по воде. Это особенно актуально для различных отраслей, таких как туризм, экспедиции, спасательные операции и др. Разработка вездехода, способного ехать по суше и плавать по воде, позволит значительно увеличить эффективность и гибкость транспортных средств. Для того чтобы сконструировать такой вездеход, необходимы следующие параметры:

- водоизмещение;
- жесткость конструкций;
- плавательная система;

- гидроизоляция корпуса;
- программная среда.

В проекте необходимо учитывать все параметры, иначе создаваемый робототехнический комплекс будет непригоден для перемещения по поверхностям, для которых он предназначен. Далее рассмотрено, как учитывается каждый параметр

Водоизмещение – это объем воды, выталкиваемый плавательным аппаратом при его погружении в воду. Он определяется размерами и формой корпуса судна, а также его весом. Важно отметить, что водоизмещение не зависит от формы корпуса, поэтому судно различной формы, но с одинаковым весом будет иметь одинаковое водоизмещение.

Правильное водоизмещение складывается из водоизмещения порожнего комплекса (массы корпуса) и массы грузов на вездеходе [1]. Также нужно учитывать архимедову силу, поскольку если она будет равна нулю, то значит, что конструкция не погружена в воду, следовательно, неустойчиво держится.

Расчет жесткости конструкции является важным этапом в проектировании различных инженерных систем. Для этого необходимо учитывать несколько ключевых факторов.

Материалы: выбор подходящих материалов с нужными свойствами прочности и упругости. Различные материалы имеют разные характеристики, определяющие их прочность и легкость. Для создания вездехода был выбран алюминий как материал, обладающий исключительной прочностью и легкостью. Такой выбор обеспечивает надежность и долговечность конструкции вездехода, что делает его идеальным для экстремальных условий и сложных маршрутов.

Геометрия: форма и размеры конструкции играют важную роль в расчете ее жесткости.

Нагрузки: учет предполагаемых нагрузок, которым будет подвергаться конструкция. Рассчитываются внутренние напряжения и деформации, которые возникают под действием внешних воздействий так же, учитывая водоизмещение, нагрузка необходима для того, чтобы рассчитать уровень погружения вездехода в воду. Для нагрузки в конструкцию загружен балласт для того, чтобы вездеход погружался в воду, но не тонул.

Чертеж корпуса амфибии приведен на рисунке.

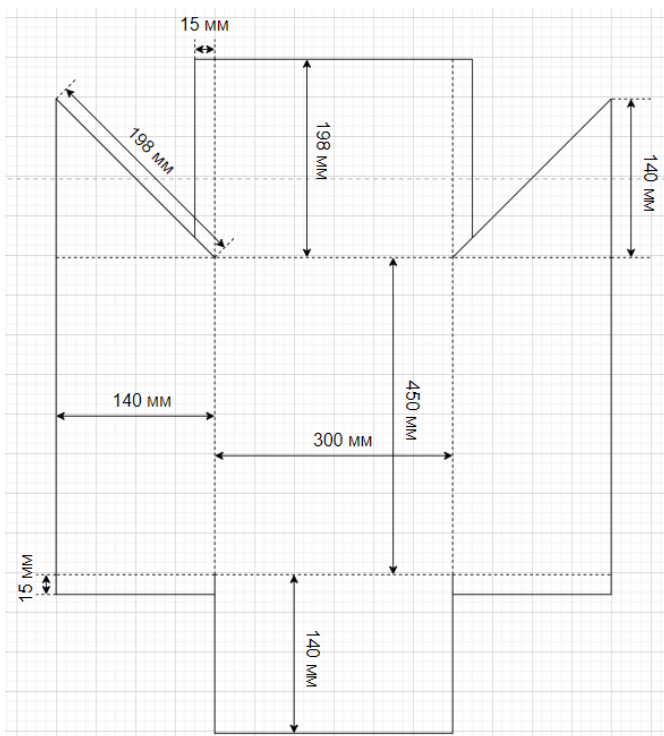


Рис. Чертеж корпуса вездехода-амфибии

Разработка системы передвижения по водной поверхности – это процесс создания комплекса оборудования и технологий, обеспечивающих передвижение по воде. В основе этой системы лежит сочетание различных элементов, таких как двигатели, корпус и т.д. Нужно понимать, как вездеход будет перемещаться по воде и суше, поэтому было решено сделать колеса, которые позволят ему передвигаться по обеим поверхностям – это позволит вездеходу выполнять его работу без дополнительных движителей.

И для того чтобы вода не попадала в корпус, нужно его гидроизолировать. В рассматриваемом проекте используется силикон. Гидроизоляция силиконом – это способ защиты поверхности от проникновения влаги при помощи специальных силиконовых материалов. Силиконовая гидроизоляция широко используется для защиты различных поверхностей, таких как бетон, кирпич, камень, металл и другие материа-

лы, от влажности и воды. Преимущества гидроизоляции силиконом включают в себя хорошую адгезию к поверхности, устойчивость к ультрафиолетовому излучению, кислотам, щелочам и температурным изменениям, а также возможность обеспечить эластичность и долговечность защиты.

За основу программной среды для вездехода был взят микрокомпьютер Orange Pi 2, который по своим характеристикам соответствует микрокомпьютеру Raspberry Pi.

Raspberry Pi – это миниатюрный одноплатный компьютер, который с легкостью поместится на ладони взрослого человека. Несмотря на свои скромные размеры, плата имеет высокую производительность, что позволяет ей выйти на один уровень со стационарными ПК [2]. Подключение Raspberry Pi произведено через SSH (Secure Shell). Это протокол прикладного уровня для безопасного обмена информацией между двумя устройствами.

При передаче по SSH информация шифруется, что делает протокол более предпочтительным для работы в незащищенных сетях, обеспечивает высокий уровень защиты. К частым сценариям его использования можно отнести работу с удаленными машинами и передачу файлов на устройства [3].

В заключение можно подвести итоги проведенного исследования и подчеркнуть его важность. В данном случае можно подчеркнуть значимость технологий, которые позволяют вездеходам передвигаться по сложным поверхностям, увеличение жесткости конструкции, способы управления вездеходом и нужные для его работы приборы, исследование и выбор оптимального источника энергии.

Можно отметить, что такая технология имеет потенциал для улучшения мобильности в различных экстремальных условиях, а также для использования в области спасательных операций, туризма и экологии. Также можно подчеркнуть значимость проведенного анализа для развития технических и инженерных решений в данной области.

Библиографический список

1. Расчет водоизмещения судна по значению его дедвейта [Электронный ресурс] // Вестник АГТУ. – URL: <https://vestnik.astu.org/ru/nauka/article/31540/> (дата обращения: 22.05.2024).

2. Что такое и для чего нужен протокол SSH [Электронный ресурс] // Selectel. – URL: <https://selectel.ru/blog/what-is-ssh/> (дата обращения: 22.05.2024).

3. Raspberry Pi: описание, подключение, схема [Электронный ресурс] // 3DIY. – URL: <https://3d-diy.ru/wiki/arduino-platy/obzor-plat-raspberry-pi/> (дата обращения: 22.05.2024).

Сведения об авторах

Усманов Азат Камилович – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-1б, г. Пермь, e-mail: usmanovazatka@gmail.com

Карлов Денис Анатольевич – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: denis.karlov.78@yandex.ru

**Р.С. Богатырев, Ю.С. Белослудцев, М.И. Белоусов,
А.В. Жигалов, А.Л. Погудин**

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА КИНЕМАТИКИ ЛАБОРАТОРНОГО СТЕНДА 3D-ПРИНТЕРА НА БАЗЕ ARDUINO

В статье приведена методика расчета усилия шагового двигателя, используемая в процессе разработки 3D-принтера, проанализированы существующие кинематические схемы 3D-принтеров, составлен анализ сред 3D-моделирования, построена модель лабораторного стенда в САПР-среде.

Ключевые слова: 3D-принтер, печать, FDM, Makerbot, САПР, Компас-3D, мехатроника, кинематика, модель системы, расчет параметров, шаговый двигатель, усилие, 3D-моделирование.

**R.S. Bogatyrev, Y.S. Belosludtsev, M.I. Belousov,
A.V. Zhigalov, A.L. Pogudin**

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF KINEMATICS OF 3D PRINTER LAB BENCH BASED ON ARDUINO

The article presents the methodology of calculation of stepper motor force used in the process of 3D-printer development, analyzes existing kinematic schemes of 3D-printers, makes an analysis of 3D-modeling environments, builds a model of laboratory bench in CAD-environment.

Keywords: 3D-printer, printing, FDM, Makerbot, CAD, Compass-3D, mechatronics, kinematics, system model, parameter calculation, stepper motor, force, 3D-modeling.

Трехмерная печать нашла свое применение в самых разных областях, а также способна существенно повлиять на характер производства в целом. Создание прототипа 3D-принтера с улучшенными характеристиками позволит расширить возможности 3D-печати в различных сферах, включая промышленность, медицину, дизайн и образование.

Актуальность темы обусловлена высокой стоимостью и сложностью конструкций существующих 3D-принтеров. Повысить интерес к данному направлению можно, сократив денежные затраты на их создание. Объект исследования – 3D-принтер. Предметом исследования является процесс создания 3D-принтера.

Цель исследования заключается в разработке и позиционировании механических приводов, расчете общего усилия двигателей и моделировании прототипа в компьютерной программе.

В соответствии с целью поставлены следующие задачи:

- провести сравнительный анализ кинематических схем 3D-принтеров;
- разработать модель в САПР среде;
- рассчитать минимальное усилие двигателей.

В рамках статьи будет исследовано влияние различных параметров на качество печати, таких как конструкция корпуса и устройство всех приводов, а также точность позиционирования, скорость движения и стабильность работы [1].

Модели кинематических схем. У FDM-принтеров выделяют несколько кинематических схем: Prusa, H-BOT, Core-XY, Makerbot, Ultimaker. В табл. 1 приведены основные сравнительные параметры, влияющие на выбор кинематической схемы.

Таблица 1

Сводная таблица кинематических схем

Параметр	Prusa	H-BOT	Core-XY	Makerbot	Ultimaker
Цена, руб.	50000	75000	105000	95000	110000
Точность позиционирования, мм	0,020	0,011	0,01	0,01	0,010
Тип корпуса (открытый/закрытый)	Открытый	Закрытый	Закрытый	Закрытый	Открытый
Скорость печати, мм/с	100	150	150	200	150

Проанализировав таблицу, приходим к выводу что схема Makerbot подходит лучше остальных. У нее не самая высокая цена, хорошая точность, скорость печати и закрытый тип корпуса, что обеспечит лучшее качество изделий. Как правило, в принтерах типа Makerbot стол статичен. Это слегка усложнит кинематическую схему, однако значительно повысит качество печати, поскольку на неподвижном столе получаются лучшие модели. Ключевые достоинства данной модели:

- легкость замены комплектующих;
- низкая цена комплектующих и самого принтера;
- высокая скорость и точность печати [2].

Для устранения недостатков кинематической схемы необходимо:

- закрыть рабочую камеру;
- сделать стол статичным.

Выбор среды моделирования. Для выбора технологической части 3D-моделирования стоит ориентироваться на имеющееся программное обеспечение. Для решения задач моделирования на профессиональном уровне используют такие программы – ЛИРА-САПР, FreeCAD, AutoCAD, NanoCAD, Компас-3D. В табл. 2 представлен сравнительный анализ сред 3D-моделирования.

Таблица 2

Сравнительный анализ сред 3D-моделирования

Программа	Бесплатная	Русифицирована	Требования к мощности ПК	Страна-производитель
ЛИРА-САПР	–	+	Низкие	РФ
FreeCAD	+	–	Высокие	Германия
NanoCAD	+/-	+	Высокие	РФ
AutoCAD	–	–	Высокие	США
Компас-3D	+	+	Низкие	РФ

Согласно результатам сравнения, в качестве инструмента для работы с 3D-моделированием был выбран «Компас-3D», поскольку для студентов он бесплатный, доступен в Российской Федерации, имеет низкие требования к персональному компьютеру и стабилен в работе.

Для наглядности общего вида конструкции изобразим 3D-принтер в САПР-среде «Компас-3D» (рисунок).

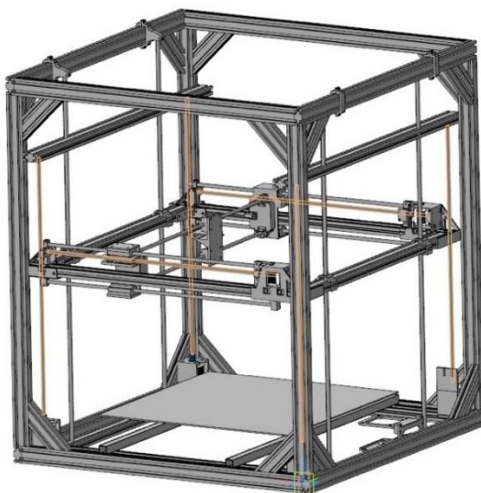


Рис. Корпус 3D-принтера

Расчет усилия двигателя. Чтобы определиться с моделью двигателя, необходимо провести расчет усилия, минимально необходимого для беспрепятственного движения системы по осям. Согласно методике расчета основных параметров ходового винта составим табл. 3.

Таблица 3

Основные параметры осей

Наименование	Обозначение	Ось			
		Z	Y	X	экструдер
Усилие, Н	F	9,51	15,82	4,82	9,82
Сила статического трения, Н	Sf	6,25	9,15	1,15	6,52
Нагрузка, Н	N	25	40	4,6	27
Радиус винта, м	R	0,004	0,004	0,004	–
Шаг винта, м	P	0,002	0,002	0,002	–
Эффективность винта, %	Se	60,96	60,96	60,96	–
Угол поворота спирали, град	ha	38,15	38,15	38,15	–
Коэффициент трения	μ	0,25	0,25	0,25	0,20
Диаметр винта, м	l	0,008	0,008	0,008	–
Средний радиус резьбы, м	r_m	0,001	0,001	0,001	–
Крутящий момент, Н · м	M	0,038	0,078	0,01	0,056

Рассмотрим методику расчета основных параметров на примере оси Z:

Составим уравнения сил и моментов относительно осей [3]:

$$F = Sf + \frac{N}{2\pi\left(\frac{R}{P}\right)Se}. \quad (1)$$

1) Для нахождения крутящего момента необходимо вычислить промежуточные параметры. Начнем с силы статического трения [3, 4]:

$$Sf = \mu \cdot N. \quad (2)$$

Материал винта – сталь, материал гайки – латунь, поэтому интересующий нас показатель трения $\mu = 0,25$.

Масса подвижной части 10 кг, при переводе в ньютоны получаем 100 Н, поэтому $N = 100$ Н. Однако за перемещение по оси Z будут отвечать 4 двигателя, следовательно, нагрузка на каждый будет 25 Н. Подставим полученные значения в формулу (2):

$$Sf = 0,25 \cdot 25, \quad (3)$$

$$Sf = 6,25 \text{ Н}. \quad (4)$$

2) Для нахождения эффективности ходового вина воспользуемся формулой [3]:

$$Se = \frac{\operatorname{tg}(ha)}{(\operatorname{tg}(ha + \arctan(\mu))) \cdot 100}. \quad (5)$$

Угол поворота спирали определяется следующим выражением:

$$ha = \arctan\left(\frac{2\pi r_m}{l}\right). \quad (6)$$

Поскольку шаг нашего винта – 2 мм, то средний радиус:

$$r_m = \frac{\text{шаг}}{2} = \frac{2}{2} = 1. \quad (7)$$

Ведущая часть винта равна его диаметру. В нашем случае ($l = 8$ мм). Возвращаемся к формуле (6):

$$ha = \arctan\left(\frac{2\pi \cdot 1}{8}\right) = 38,15^\circ. \quad (8)$$

Все искомые величины для нахождения эффективности ходового вина стали известны, подставляем значения в формулу (5):

$$Se = \frac{\operatorname{tg}(38,15^\circ)}{(\operatorname{tg}(38,15^\circ + \arctan(0,25))) \cdot 100} = 60,96 \%. \quad (9)$$

Переведем 60,96 % в КПД. Следовательно, $Se = 0,61$.

3) Все искомые величины для нахождения усилия были получены. Возвращаемся к формуле (1):

$$F = Sf + \frac{N}{2\pi\left(\frac{R}{P}\right)_{Se}} = 6,25 + \frac{25}{2\pi\left(\frac{4}{2}\right)_{0,61}} = 9,51. \quad (10)$$

Переведем усилие в крутящий момент:

$$M = r \cdot F = 0,004 \cdot 9,51 = 0,038 \text{ Н} \cdot \text{м}. \quad (11)$$

Таким образом, был подсчитан минимальный крутящий момент, которым должны обладать двигатели на оси Z. Используем ту же методику расчета для остальных осей и экструдера [4].

Заключение. В результате достижения поставленных задач была выбрана кинематическая схема типа Makerbot, поскольку обладает высокой точностью и скоростью печати. Модель 3D-принтера была создана в САПР-среде «Компас-3D», а также были успешно рассчитаны основные параметры осей. По предварительным расчетам оценочная стоимость кинематики 3D-принтера составит 15 000 руб., в то время как аналогичные готовые решения начинаются от 40 000 руб.

Библиографический список

1. 3D-печать как новое научно-техническое направление [Электронный ресурс]. – URL: <https://cyberleninka.ru/article/n/3d-pechat-kak-novoe-nauchno-tehnicheskoe-napravlenie/viewer> (дата обращения: 10.04.2024).
2. Диссертация на тему «Математическое и программное обеспечение подсистемы эскизного проектирования САПР экструзионных 3D-принтеров» [Электронный ресурс]. – URL: https://etu.ru/assets/files/nauka/dissertacii/2022/kopec/dissertaciya_kopec.pdf (дата обращения: 11.04.2024).
3. Бельков В.Н. Основы расчета и конструирования винтовых механизмов: учеб. пособие. – Омск: Изд-во ОмГТУ, 2008. – 160 с.
4. Расчет основных характеристик винтов [Электронный ресурс]. – URL: <https://help.autodesk.com/view/INVNTOR/2023/RUS/?guid=GUID-149FFEF7-5A5A-482D-98DC-97F6958898D4> (дата обращения: 20.04.2024).

Сведения об авторах

Богатырев Роман Сергеевич – бакалавр Пермского национально-го исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: romanbog2002@gmail.com

Белослудцев Юрий Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: belosludcevyu@gmail.com

Белоусов Михаил Игоревич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: mikhail.belousov.2002@mail.ru

Жигалов Артем Валерьевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: artem.zhigalov.27@mail.ru

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

**А.В. Жигалов, М.И. Белоусов, Р.С. Богатырев,
Ю.С. Белослудцев, А.Л. Погудин**

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ПРОГРАММНОГО МОДУЛЯ ЛАБОРАТОРНОГО СТЕНДА 3D-ПРИНТЕРА НА БАЗЕ АРДУИНО

В статье рассмотрены технологии, используемые в процессе разработки 3D-принтера, проанализированы существующие решения, разработано программное обеспечение, блок-схемы алгоритмов функций.

Ключевые слова: лабораторный стенд, 3D-принтер, блок-схемы алгоритмов, модель системы, тестирование, слайсер, прошивка, структурная схема.

**A.V. Zhigalov, M.I. Belousov, R.S. Bogatyrev,
Yu.S. Belosludtsev, A.L. Pogudin**

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF THE PROGRAM MODULE OF THE LABORATORY STAND OF 3D-PRINTER ON THE BASIS OF ARDUINO

In the article the technologies used in the process of 3D-printer development are considered, existing solutions are analyzed, software, block diagrams of function algorithms are developed.

Keywords: laboratory bench, 3D printer, block diagrams of algorithms, system model, testing, slicer, firmware, structural scheme.

Технологии обработки материалов стали неотъемлемой частью производства деталей. Вместе с традиционными методами аддитивное производство, основанное на наращивании материала, набирает популярность и открывает новые возможности в промышленности.

Сегодня 3D-принтеры FDM популярны из-за доступности и простоты. Технологии 3D-печати развиваются, включая новые методы и материалы. Прогресс в аддитивных технологиях опережает проектирование принтеров, что может вызывать ошибки. Для роста спроса внедряются новые решения, например, бесшумные двигатели и точные экструдеры [1].

Актуальность заключается в разработке простых надежных алгоритмов, необходимых для снижения времени обработки и выполнения программы, что впоследствии может привести к снижению электропотребления 3D-принтера. Объект исследования – лабораторный стенд 3D-принтера. Предмет исследования – процесс алгоритмизации и программирования 3D-принтера.

Цель работы – разработка лабораторного стенда, предназначенного для обучения и исследования в области 3D-печати. Изучение основных алгоритмов работы и принципов программирования 3D-принтера.

Задачи исследования – алгоритмизация и программирование аппаратной части лабораторного стенда, обоснование и выбор информационного и аппаратного обеспечения.

Обоснование выбора прошивки и слайсера. Прошивка 3D-принтера представляет собой программное обеспечение, которое контролирует работу оборудования, обеспечивает взаимодействие между электронными компонентами принтера и программой печати.

Существуют две самые популярные прошивки: Marlin и Klipper. Рассмотрим их достоинства и недостатки и выберем лучшую из них [2].

Прошивка Marlin является доминирующим игроком на рынке 3D-печати, ее различные вариации можно встретить в большинстве 3D-принтеров, использующих технологию FDM/FFF.

Созданный Эриком ван дер Залмом в 2011 г. для Ultimaker, Marlin изначально разрабатывался для использования с доступными восьмимбитными микроконтроллерами Arduino.

Klipper, выпущенный впервые в 2016 г. Кевином О'Коннором, представляет собой совершенно инновационный подход к управлению 3D-принтером. Учитывая ограничения небольших микроконтроллеров на возможности принтеров, Klipper перераспределяет основную часть вычислений на более мощное устройство, такое как Raspberry Pi.

Этот подход позволяет Klipper использовать больше вычислительной мощности, что способствует повышению точности и скорости печати.

Marlin не остается в стороне и продолжает развиваться, выпуская обновления, такие как версии 2 и 2.1, которые поддерживают 32-битные контроллеры повышенной мощности. Разработчики постоянно совершенствуют продукт, внедряя новые улучшения.

В таблице представлены популярные слайсеры, сравним их для выбора оптимального.

Сравнение слайсеров

1. Слайсер	2. Ориентация	3. Доступность	4. Поддержка	5. Страна производитель
6. Astroprint	7. новичок, продвинутый	8. условно бесплатно	9. Raspberry Pi, браузер	10. США
11. CraftWare	12. новичок, продвинутый	13. бесплатно	14. Windows, Mac, Linux	15. Польша
16. Cura Ultimaker	17. новичок, продвинутый	18. бесплатно	19. Windows, Mac, Linux	20. Нидерланды
21. IceSL	22. новичок, продвинутый	23. бесплатно	24. Windows, Mac, Linux	25. Франция
26. ideaMaker	27. новичок, продвинутый	28. бесплатно	29. Windows, Mac, Linux	30. Китай
31. MakerBot Print	32. начинающий	33. бесплатно	34. Windows, Mac,	35. США
36. OctoPrint	37. средний, продвинутый	38. бесплатно	39. Windows, Mac, Linux Raspberry Pi	40. Нидерланды
41. Repetier	42. средний, продвинутый	43. бесплатно	44. Windows, Mac, Linux	45. Германия
46. SelfCAD	47. новичок, продвинутый	48. условно бесплатно	49. Браузер	50. США
51. Slic3r	52. опытный, профессиональный	53. бесплатно	54. Windows, Mac, Linux	55. Чехия
56. SliceCraft-er	57. опытный	58. бесплатно	59. Браузер	60. Канада

Проанализировав самые популярные прошивки и слайсеры, было решено использовать Marlin в качестве прошивки, так как в нашем случае при разработке лабораторного стенда нет надобности улучшать производительность и скорость печати также, чтобы избежать дополнительных трат на Raspberry PI, а в качестве слайсера было принято решение использовать Cura Ultimaker, так как удовлетворяет всем потребностям, имеет большое количество положительных отзывов и проста в использовании.

Алгоритмизация и тестирование аппаратной части. Тестирование аппаратной части лабораторного стенда заключается в проверке работоспособности и качества физических компонентов и механизмов, которые обеспечивают функционирование устройства. Это включает в себя различные аспекты, такие как проверка работы двигателей и механизмов перемещения, оценка точности и стабильности позиционирования печатающей головки, а также тестирование системы нагрева и других функциональных элементов [3].

Разработка алгоритма и программы для проверки работоспособности шаговых двигателей. Для начала было необходимо разработать алгоритм работы шаговых двигателей (рис. 1).

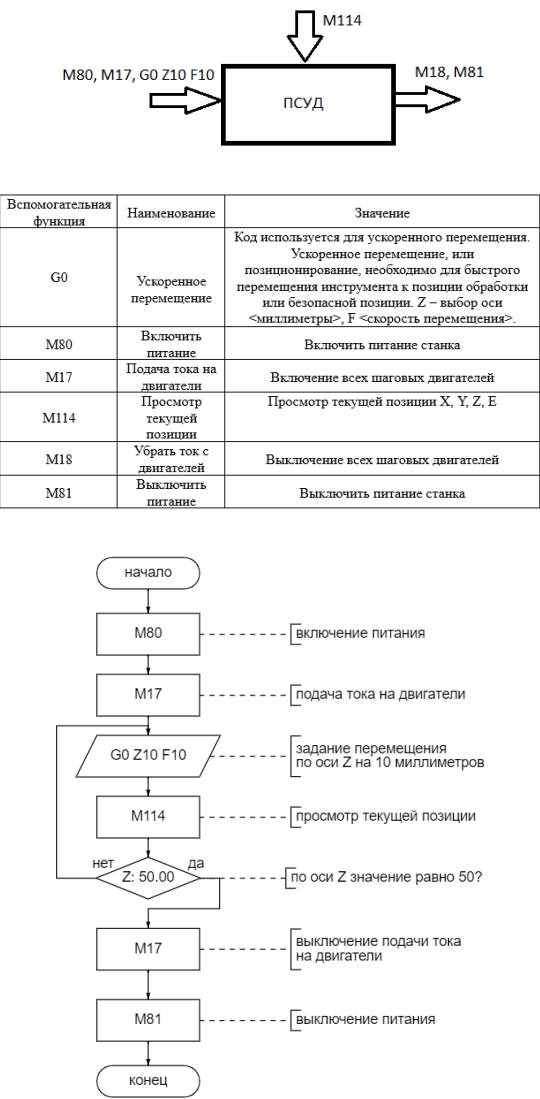


Рис. 1. Спецификация сигналов и алгоритм вращения шаговых двигателей

Проверка работоспособности нагревательного стола. Для стола распишем структурную схему и алгоритм нагрева (рис. 2, 3).

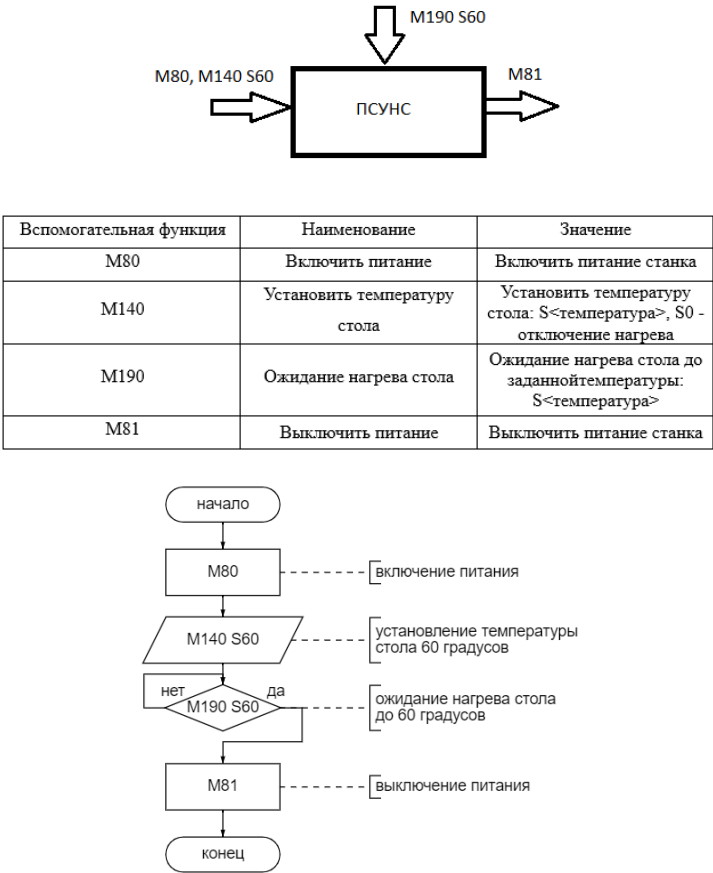


Рис. 2. Спецификация сигналов и алгоритм нагрева стола



Рис. 3. График нагрева стола при исполнении алгоритма

Первый период времени заданная температура равнялась 50 градусам цельсия, во второй период времени задали температуру, равную 60 градусам. Можно увидеть, как температура устанавливается около заданных значений.

Проверка нагрева экструдера. Составим структурную схему и алгоритм нагрева экструдера (рис. 4).



Рис. 4. Спецификация сигналов и алгоритм нагрева экструдера

Первый период времени заданная температура равнялась 115 °С, во второй период времени задали температуру, равную 165 °С. Можно увидеть, как температура устанавливается около заданных значений (рис. 5).

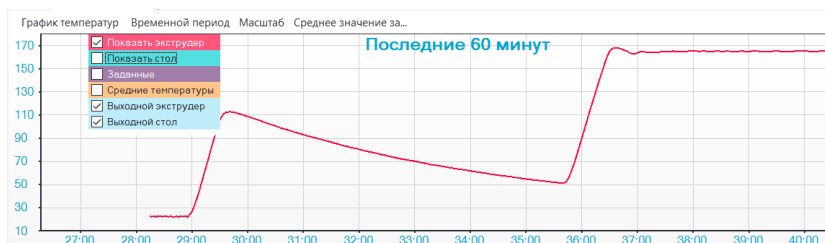


Рис. 5. График нагрева экструдера при исполнении алгоритма

Система управления трехмерным принтером. Распишем общую структурную схему для управления 3D-принтером (рис. 6).

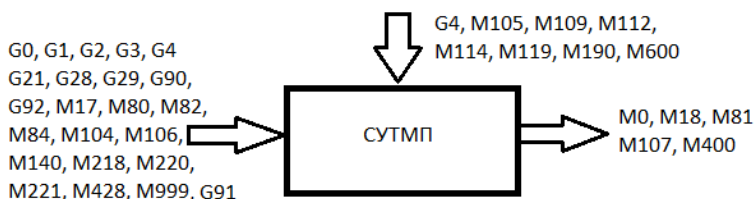


Рис. 6. Черный ящик системы управления трехмерным принтером

Заключение. В результате решения поставленных задач были получены оптимальные алгоритмы для работы каждой из подсистем 3D-принтера, обеспечивающие надежность системы, снижение времени выполнения и обработки, а также снижение потребления электроэнергии.

Библиографический список

1. Лейбов А.М. Применение технологий 3D-прототипирования в образовательном процессе // Современные проблемы науки и образования. – 2014. – № 5. – С. 93.
2. Обзор слайсеров для 3D-печати [Электронный ресурс]. – URL: <https://elibrary.ru/item.asp?id=49531162>
3. 3Dtoday.ru – портал о 3D-печати. – URL: <https://3dtoday.ru/wiki>

Сведения об авторах

Жигалов Артем Валерьевич – бакалавр Пермского национально-го исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: artem.zhigalov.27@mail.ru

Белоусов Михаил Игоревич – бакалавр Пермского национально-го исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: mikhail.belousov.2002@mail.ru

Богатырев Роман Сергеевич – бакалавр Пермского национально-го исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: romanbog2002@gmail.com

Белослудцев Юрий Сергеевич – бакалавр Пермского националь-ного исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: belosludcevyu@gmail.com

Погудин Андрей Леонидович – кандидат технических наук, до-цент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехниче-ского университета, г. Пермь, e-mail: pogudin_al@mail.ru

Н.Т. Жогаштиев, В.С. Анисимов

Лысьвенский филиал Пермского национального исследовательского
политехнического университета, Лысьва

УПРАВЛЕНИЕ АВТОМАТИЗАЦИИ С ИСПОЛЬЗОВАНИЕМ МИКРОКОНТРОЛЛЕРА ESP8266 И BLYNK-ПРИЛОЖЕНИЕ

В данной статье предлагается создание системы управления автоматизации с использованием микроконтроллера ESP8266 и приложения Blynk. С помощью этого управления мы сможем измерять температуру и влажность из любой точки мира. Это устройство может быть полезно для измерения температуры и влажности в помещении, а также его можно использовать для промышленных целей. По сути, это устройство основано на приложении Blynk, поэтому требуется Wi-Fi и подключение к Интернету. В программе Arduino IDE производились системы программирования, приложения Blynk и датчиков, также существует своя библиотека. Приложение Blynk было разработано для Интернета вещей (IoT), оно также может управлять оборудованием, удаленно отображать данные хранилища датчиков, визуализировать его и делать много других интересных вещей.

Ключевые слова: Wi-Fi, Arduino IDE, IoT, Blynk, ESP8266, DHT22, Интернет, электричество.

N.T. Zhogashtiev, V.S. Anisimov

Lysva Branch of Perm National Research Polytechnic
University, Lysva

AUTOMATION CONTROL USING ESP8266 MICROCONTROLLER AND BLYNK APPLICATION

This article proposes the creation of an automation control system using the ESP8266 microcontroller and the Blynk application. With this control we will be able to measure temperature and humidity from anywhere in the world. This device can be useful for measuring indoor temperature and humidity and can also be used for many industrial purposes. Basically, this device is based on the Blynk application, so it requires Wi-Fi and an internet connection. The Arduino IDE program produced programming systems, Blynk and sensor applications, and also has its own library. Blynk app was developed for Internet of things (IoT), it can also control equipment, remotely display sensor data store data, visualize it and do many other cool things (IoT).

Keywords: Wi-Fi, Arduino IDE, IoT, Blynk, ESP8266, DHT22, Internet, electricity.

Сегодня многие предприятия используют и признают преимущества Интернета вещей (IoT), чем когда-либо прежде. Благодаря взрыв-

ному росту использования и внедрения Интернета вещей у компаний, которые рано присоединяются к революции Интернета вещей, открываются прекрасные возможности. Корпорации, которым удалось трансформироваться и расширить свои возможности благодаря преимуществам Интернета вещей, могут создать неоспоримые конкурентные преимущества [1]. Технология Интернета вещей неуклонно развиваться, и поэтому всегда нужно развивать тенденции Интернета вещей. В будущем ожидается, что в 2024 г. усовершенствованные протоколы безопасности станут стандартом обеспечения безопасности наших данных. Что касается здравоохранения, Интернет вещей совершает революции в том, как мы управляем здоровьем.

Для визуализации режимы предлагают производителям устройств Интернет вещей, но функция подключения к Интернету иногда отсутствует, поэтому требуется специальное вспомогательное аппаратное устройство, подключенное к Интернету через протокол IPv4 и мобильных устройств через подходящее программное приложение [2]. Рост количества подключенных устройств означает повышенный риск и кибератаки. Это не только компьютеры и телефоны, но и любые устройства, подключенные к сети, будь то кухонная техника или промышленное оборудование. По мере распространения удаленной работы защита этих сетей жизненно важна как для предприятий, так и для частных лиц. Управление автоматизации с использованием микроконтроллера ESP8266 и Blynk приложения – это одна из оптимальных систем для Интернета вещей. Эта система позволяет удаленно управлять температурой и влажностью с помощью мобильного приложения [3]. С помощью микроконтроллера ESP8266 можно определить утечку газа, управление электроэнергией, распознавание цвета, а также создать инновационные проекты. Мы рассмотрели управление температурой и влажностью с мобильного приложения на соединение сетей онлайн путем обеспечения беспроводной связи. Схема системы показана на рис. 1.

Схема системы состоит из датчиков, отвечающих за измерение температуры и влажности (DHT22), а также Wi-Fi будет подключен к серверу Blynk, для этого процесса требуется подключение к Интернету, и этот процесс будет выполняться маршрутизатором Wi-Fi. Первое, что нужно заметить, – это ESP8266, в этом микроконтроллере нет напряжения 5 В, контакт на этой плате работает при напряжении 3,3 В, это не влияет на аппаратные устройства разного датчика [4]. Датчик температуры и влажности (DHT22) питается от источника напряжения 5 В но это также будет работать при напряжении 3,3 В. Контакт заземления, который помечен здесь как GND, и он, очевидно, будет заземлен

на Android MCU к любому контакту заземления, который нужно подключить. Контакт VCC, который будет нашим положительным источником питания, и поступит к 3,3 В на ESP8266.

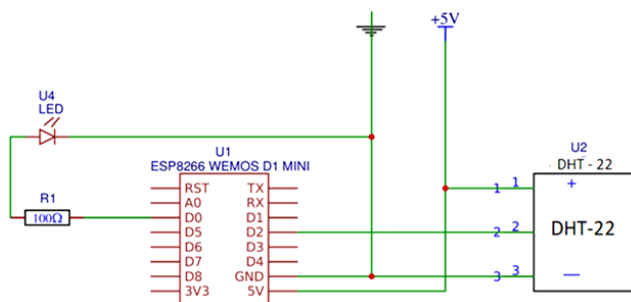


Рис. 1. Схема системы управления автоматизацией

Описание оборудования. Микроконтроллер ESP8266, оснащенный модулем ESP-12E, наглядно представлен на рис. 2.

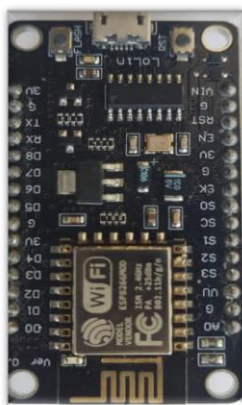


Рис. 2. Микроконтроллер ESP8266

Микроконтроллер ESP8266 работает на частоте 80–160 МГц, на нем имеется около 64 КБ SRAM свободной оперативной памяти, памяти под программы в самом контроллере нет, но он поддерживает внешнюю память, и в этом модуле распаяна флешка $N = 4$ МБ. Благодаря линейному стабилизатору источника напряжение можно подавать до 15 В. А также подключенный микроконтроллер ESP8266 оснащен Wi-Fi и Bluetooth. Датчик температуры/влажности DHT22 представлен на рис. 3.



Рис. 3. Цифровой датчик DHT22

Цифровой датчик DHT22 предназначен для измерения базовой температуры и влажности на низком уровне, который требует синхронизации для получения данных. После подключения его к плате ESP8266, воспользуемся библиотекой Arduino IDE. Она позволяет получать температуру и влажность с помощью датчика. Для измерения воздуха датчик оснащен специальным пластиковым термистором вокруг него [5]. Потребление источника постоянного тока: 2,5 мА макс, напряжение: 3–5 В, показания влажности: 0–100 % (2–5 %).

Приложение Blynk. Blynk – это IoT-платформа, которая позволяет управлять взаимодействием монитора IoT-устройства. Термин Blynk– это разброс IoT, что представляет набор Arduino библиотеки и облачный сервер базовую плату ESP8266 или Raspberry Pi, а также поддерживается множество других плат. Эскиз с помощью Arduino IDE и библиотеки Blynk, которые создадут соединение между устройством и сервером Blynk на сторону внешнего интерфейса, наглядно представлены на рис. 4.

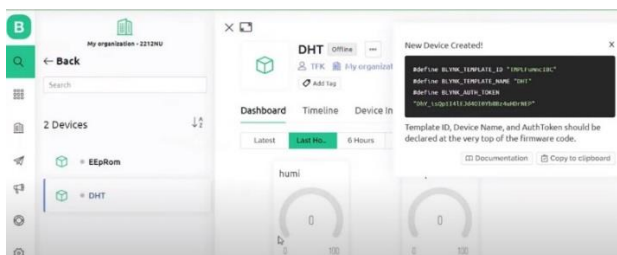


Рис. 4. Приложение Blynk

Для получения доступа Blynk из мобильного приложения нужно скачать пользовательский интерфейс Blynk. Важно отметить, что приложение Blynk на самом деле представляет собой пользовательский интерфейс – приложение, в котором есть изображение для кнопок, ползунков, джойстиков, элементов отображения, графиков, датчиков, инструментов движений, доступных в приложении для управлений заданного устройствами [6]. На этих устройствах любая логика должна быть реализована в коде C++, работающем на устройстве Интернет вещей, который представлен на рис. 5.

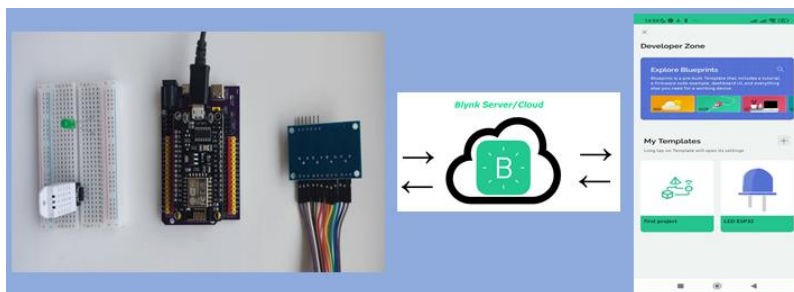


Рис. 5. Основные компоненты мобильного приложения Blynk

Прототип системы. Для проектирования прототипа мы использовали программу КОМПАС-3D, которая используется для 3D-моделирования. Прототип модели представляет собой компактное доказательство концептуального модуля, распечатан на 3D-принтере красным печатным материалом PLA (полилактид), минимальная толщина стенок 1 мм. Следует упомянуть об этом прототипе системы, что он был изготовлен для презентации.

Заключение. Управление автоматизации на базе микроконтроллера ESP8266 и приложения Blynk предназначено для пользователей и позволяет подключать больше устройств на более высокой скорости с минимальной задержкой. С помощью этого нового приложения Blynk мы сможем увидеть полную автоматизацию во многих отраслях с помощью интеллектуальных сетей. Высокая стоимость является самой большой проблемой во внедрении коммерческих систем. Общую стоимость системы увеличивают дополнительные сетевые адаптеры, такие как концентраторы. Система автоматизации использует разработку микроконтроллера ESP8266 и платформу Blynk, чтобы достигнуть своей экономической цели. Чтобы подключить эту систему, требуется интернет-соединение с ис-

пользованием мобильного типа, доступного на операционных системах Android и iOS. Масштабы системы будут более продвинутыми и отзывчивыми, их можно использовать как для предприятий, так и для частных лиц. А также множество других услуг, в том числе модуль GSM, может быть включен в систему для улучшения надежности.

Библиографический список

1. Аванесян Г.Р., Левшин В.В. Интегральные микросхемы ТТЛ, ТТЛШ: справочник. – М.: Машиностроение, 1993. – 256 с.
2. Антимиров В.М. Системы автоматического управления: учеб. пособие для вузов; науч. ред. В.В. Телецина. – М.: Юрайт, 2017.
3. Talking Energy Meter Based on Microcontroller / R. Raksha Sharma, Ms. Kalyani, P. Wath, Mr. Yogesh, P. Bawangade, M.D. Ghatole // International Research Journal of Engineering and Technology (IRJET). – Jan-2018. – Vol. 5, iss. 1.
4. Новожилов, О.П. Электротехника и электроника. – М.: Гардайки, 2008. – 653 с.
5. Черпаков Б.И., Вереина Л.И. Автоматизация и механизация производства: учеб. пособие для студ. учреждений сред. проф. образования. – М.: Академия, 2004. – 384 с.
6. Narmada T., Lakshmaiah M.V., Nagamma N.N. Design and development of Raspberry Pi based system for prepaid electricity meter // IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI). – 2017. DOI: 10.1109/ICPCSI.2017.8392137

Сведения об авторах

Жогаштиев Нурлан Тилекович – кандидат технических наук, доцент кафедры «Технические дисциплины» Лысьвенского филиала Пермского национального исследовательского политехнического университета, г. Лысьва, e-mail: zhogashtiev2583@gmail.com

Анисимов Виталий Сергеевич – бакалавр Лысьвенского филиала Пермского национального исследовательского политехнического университета, гр. ЭВТ-21-16, г. Лысьва, e-mail: anisimov.vitalik04042001@yandex.ru

А.К. Тарасова, Е.В. Долгова

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА МЕХАНИЗМА ДВИЖЕНИЯ СТРЕЛЫ ПОРТАЛЬНОГО КРАНА В СРЕДЕ UNREAL ENGINE 5

В статье приведен анализ основных составляющих элементов портального крана для проектирования 3D-модели. Описан процесс разработки механизма движения стрелы портального крана в среде Unreal Engine 5 и реализации алгоритма, позволяющего управлять стрелой крана. Полученные выводы могут быть полезны для улучшения эффективности создания тренажера крановщиков портального крана и оптимизации процессов автоматизации производства.

Ключевые слова: портальный кран, 3D-модель, Unreal Engine 5, шарнирно-сочлененная стреловая система, Physics Constraint.

A.K. Tarasova, E.V. Dolgova

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A MECHANISM FOR THE MOVEMENT OF A GANTRY CRANE BOOM IN THE UNREAL ENGINE 5 ENVIRONMENT

The article provides an analysis of the main components of a portal crane for designing a 3D model. The process of developing a mechanism for moving the boom of a gantry crane in the Unreal Engine 5 environment and implementing an algorithm that allows controlling the boom of a crane is described. The obtained conclusions can be useful for improving the efficiency of creating a simulator for gantry crane operators and optimizing production automation processes.

Keywords: gantry crane, 3D model, Unreal Engine 5, articulated boom system, Physics Constraint.

Конструкцию портального крана составляют: портал, колонна, шарнирно-сочлененная стреловая система [1]. Для реализации механизма движения стрел крана в программе Blender была спроектирована 3D-модель портального крана, которая представлена на рис. 1.

Портальный кран имеет конструкцию шарнирно-сочлененной стреловой системы (ШСС), представленной на рис. 2, включающая 1 – стрелу, 2 – хобот, 3 – оттяжку, 4 – коромысло, 5 – стреловая тяга.

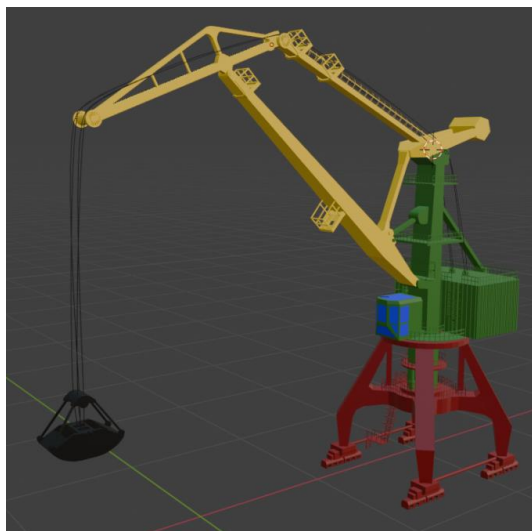


Рис. 1. 3D-модель портального крана

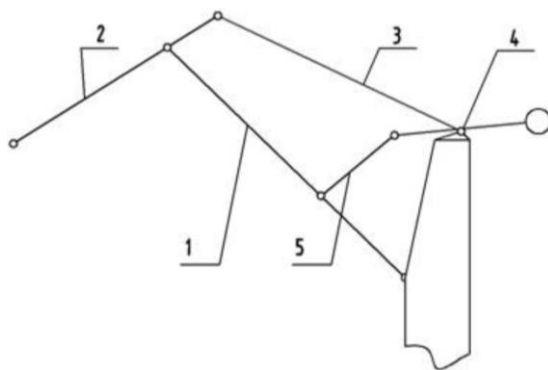


Рис. 2. Схема шарнирно-сочлененной стреловой системы

Для реализации механизма движения стрел портального крана необходимо импортировать готовую 3D-модель в Unreal Engine 5 в формате fbx. Далее с помощью Blueprint – системы визуальных скриптов – установим сцену в центр портала крана. Создаем объект Physics Constraint, который позволяет привязать физический объект к другому, таким образом осуществляем привязку коромысла и оттяжки, которая представлена на рис. 3.

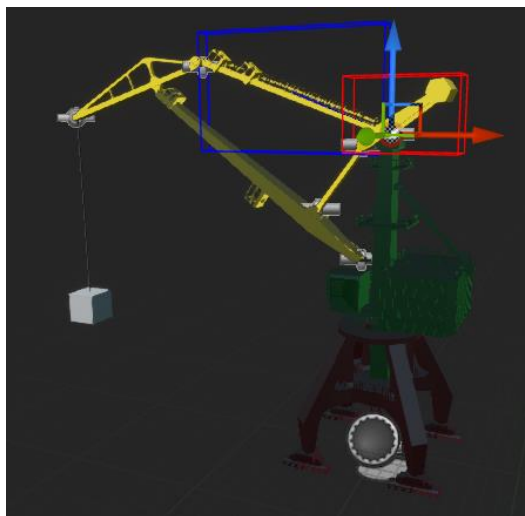


Рис. 3. Physics Constraint для привязки оттяжки и коромысла

Также необходимо создать объекты Physics Constraint для всех составляющих ШСС, где находятся узлы, в которых происходит движение (рис. 4).

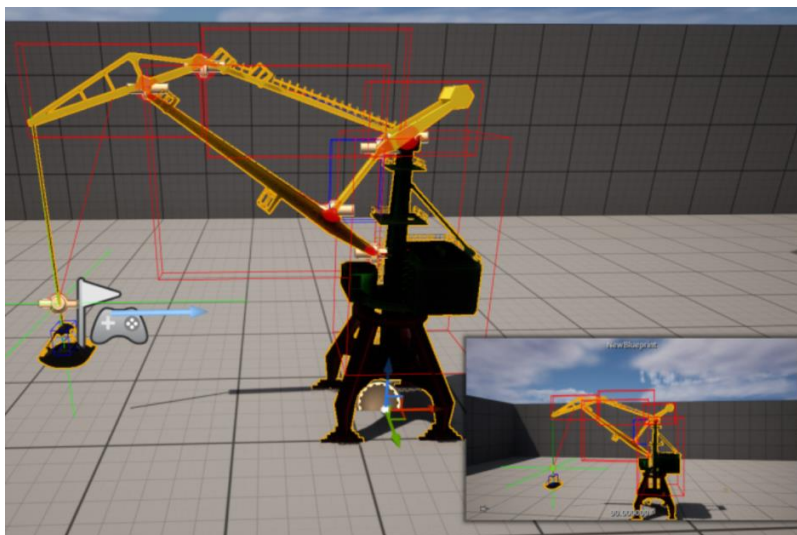


Рис. 4. Все объекты Physics Constraint

Для каждого объекта шарнирно-сочлененной стреловой системы необходимо вручную установить параметры коллизии, чтобы избежать столкновения с коллизиями других объектов (рис. 5).

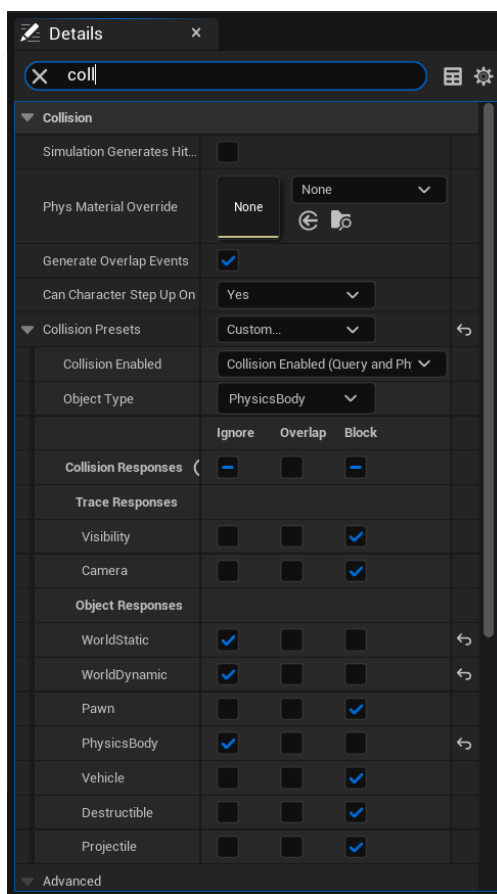


Рис. 5. Установка параметров коллизий объектов

Управление движением стрелы крана будет осуществляться за счет движения коромысла, ранее созданные Physics Constraint будут приводить в движение все остальные элементы, такие как стрелу, хобот, оттяжку и стреловую тягу. Задаем параметры ввода пользователя, в окне Edit/Project Settings/Input, поднятие / опускание стрелы и поворот будет осуществляться нажатием клавиш up/ down и left / right на клавиатуре (рис. 6).

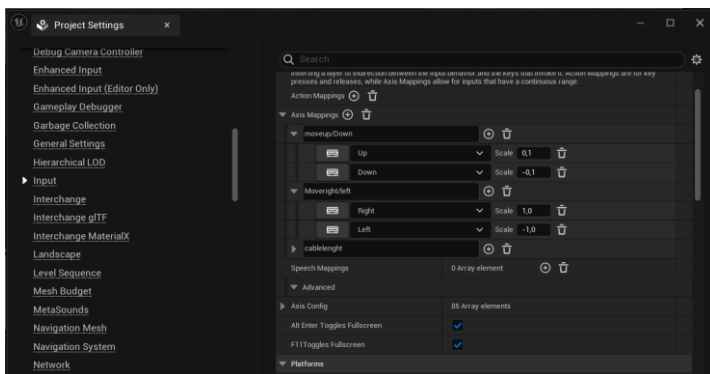


Рис. 6. Настройка ввода

В Blueprint создаем скрипт для движения коромысла ШСС по оси Y с привязкой созданных клавиш up/ down на клавиатуре (рис. 7).

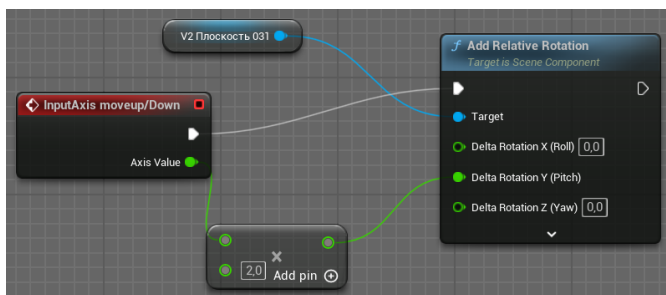


Рис. 7. Скрипт поднятия и опускания стрелы

Далее написан скрипт для механизма движения поворотной части портального крана по оси Z с привязкой клавиш left/ right на клавиатуре (рис. 8).

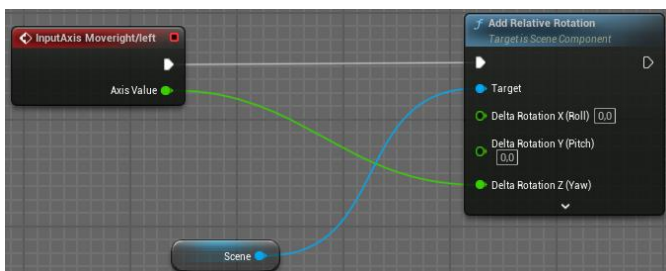


Рис. 8. Скрипт поворота стрелы

Библиографический список

1. Игумнов С.Г. Грузоподъемные краны и грузозахватные приспособления: учеб. пособие. – М.: Академия, 2007. – 64 с.

Сведения об авторах

Тарасова Алиса Кирилловна – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-1б, г. Пермь, e-mail: tarasovalisa03@mail.ru

Долгова Елена Владимировна – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: elena@dolgova.space

**А.Л. Погудин, М.И. Белоусов, А.В. Жигалов,
Р.С. Богатырев, Ю.С. Белослудцев**

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА АППАРАТНОГО МОДУЛЯ ЛАБОРАТОРНОГО СТЕНДА 3D-ПРИНТЕРА НА БАЗЕ АРДУИНО

В статье рассмотрены компоненты, используемые в процессе разработки 3D-принтера, проанализированы существующие решения, подобраны основные комплектующие, составлены схемы.

Ключевые слова: 3D-принтер, лабораторный стенд, контроллер, плата, шаговый двигатель, драйвер, экструдер, схема.

**A.L. Pogudin, M.I. Belousov, A.V. Zhigalov,
R.S. Bogatyrev, Yu.S. Belosludtsev**

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A HARDWARE MODULE FOR A LABORA- TORY STAND FOR AN ARDUINO-BASED 3D PRINTER

The article discusses the components used in the development of a 3D printer, analyzes existing solutions, selects the main components, and draws up diagrams.

Keywords: 3D printer, laboratory stand, controller, board, stepper motor, driver, extruder, circuit.

Использование 3D-принтеров в учебных заведениях становится все более распространенным явлением. Такие устройства позволяют студентам получить практические навыки в области аддитивных технологий. Подходы к выбору компонентов и проектированию аппаратного модуля для учебных 3D-принтеров чаще всего опирается на дешевизну, доступность и простоту в освоении для студентов [1].

Актуальность темы вызвана растущим интересом к 3D-печати среди молодежи и обусловлена необходимостью проектирования недорогого аппаратного модуля 3D-принтеров. Объектом исследования является лабораторный стенд «3D-принтер». Предметом исследования являются процесс и инструменты проектирования аппаратного модуля принтера.

Цель проекта – создание лабораторного стенда для изучения элементов схемотехники на базе 3D-принтера с рабочей областью 50×50×50 см.

Для достижения поставленной цели требуется решение следующих задач:

- провести и обосновать выбор элементной базы 3D-принтера;
- разработать проект аппаратной части 3D-принтера.

Обоснование и выбор контроллера. Ниже в табл. 1 представлена выборка из популярных управляющих плат, выбор будет осуществляться из них.

Таблица 1

Сравнение управляющих плат 3D-принтеров

№ п/п	Микроконтроллер	Рабочее напряжение, В	Драйверы шаговых двигателей	Экструдеры	Интерфейсы передачи данных	Цена, руб
1	SKR 3	12/24	5	2	USB, Micro SD Card	4900
2	Duet 3 Mini 5+	11/25	5	2	USB, Micro SD Card	5300
3	Creality V4.2.7	24	5	1	Mini-USB, Micro SD-Card	5000
4	MKS Robin Nano V3.1	24	5	2	Mini-USB, Micro SD-Card	5600
5	Arduino mega 2650 Rev 3 +RAMPS 1.4	12	6	2	USB, SD-Card	1500

В качестве контроллера будет использоваться Arduino mega 2650 Rev 3. Выбор данного контроллера обусловлен его дешевизной и доступностью по сравнению с аналогами.

Так как Arduino обладает недостаточным рабочим напряжением и в общем подключение компонентов к ней было бы затруднительным, используется плата Ramps 1.4. Она используется в качестве shield, к ней происходит подключение всех узлов принтера, кроме USB-кабеля. Основные отличия от более современных версий 1.5 и 1.6 заключаются в предохранителях, что не сильно критично, но, в свою очередь, версия 1.4 гораздо дешевле.

Обоснование и выбор шаговых двигателей. Далее необходимо выбрать шаговые двигатели (ШД). В данном случае будут использоваться они, а не сервоприводы из-за меньшей стоимости и встроенной обратной связи. Опираясь на проведенные вычисления, был выбран ШД модели NEMA 17 17HS4401 с заявленным крутящим моментом в 0,5 Нм.

Обоснование и выбор драйверов шаговых двигателей. После этого необходимо подобрать драйвера ШД. Ниже представлена таблица сравнения популярных моделей драйверов, опираясь на выдаваемый ток и возможный микрошаг (табл. 2).

Таблица 2

Сравнение драйверов шаговых двигателей

№	Модель	Максимальный ток, А	Микрошаг	Цена, руб
1	A4988	1	1/16	190
2	DRV8825	1,5	1/32	280
3	TMC2100	2,5	1/256	750
4	TMC2160_OC	4,33	1/32	1890

В данном случае будут использоваться драйверы DRV8825, поскольку при подъеме оси Z участвуют две пары двигателей, соединенных параллельно и подключенных к одному драйверу, A4988 не может выдать необходимую мощность при требуемой силе тока в 1,5 А. Кроме того, DRV8825 поддерживает микрошаг в 1/32, но этот факт не является необходимостью [2].

Обоснование и выбор нагревательного стола. Нагревательные элементы могут иметь разную основу, но для проекта был выбран силиконовый, он обладает большей скоростью нагрева, большей мощностью, поскольку питание идет от сети 220 В. С одной из сторон стол оборудован самоклеющейся поверхностью, что позволяет закрепить его на листе.

Обоснование и выбор экструдера. Из-за особенностей конструкции принтера вариант с расположением ШД экструдера в удалении и подачей филамента через тефлоновую трубку невозможен, поэтому выбор будет проводиться из экструдеров в сборе, а отдельные хотэнды при выборе в сравнение входить не будут [3]. Ниже представлена табл. 3 сравнения популярных моделей экструдеров.

Таблица 3

Сравнение экструдеров

№ п/п	Экструдер	Диаметр сопла, мм	Максимальная температура, °С	Скорость подачи, мм/с	Комплектация	Цена, руб.
1	MK8	0,2–0,4	260	40	ШД, куллер, радиатор, термистор	2300
2	E3D Titan	0,2–0,4	260	80	ШД, куллер, радиатор, термистор	3600
3	P3steel	0,2–1,2	350	120	ШД, куллер, радиатор, термистор	5390

Экструдер MK8 своей комплектностью не отличается от своих аналогов, кроме того, поскольку печать предполагается только ABS и PLA пластиком, максимальной температуры в 260 °С достаточно, а скоростью можно пренебречь, поэтому из-за своей стоимости была выбрана именно эта модель.

Расчет необходимого блока питания. Рассчитаем необходимую мощность блока питания для RAMPS 1.4. Ниже представлена табл. 4 энергопотребления всех компонентов.

Таблица 4

Потребление всех компонентов

Элемент	Потребление на единицу, Вт	Количество, шт	Общее потребление, Вт
Питание от блока питания:			
Arduino mega 2650	3,6	1	3,6
Ramps 1.4	1	1	1
Шаговый двигатель	18	8	144
Графический контроллер	3,6	1	3,6
Экструдер	48	1	48
Вентиляторы	48	2	1,68
Итого:			201,88
Питание от сети:			
Нагревательный стол	800	1	800
Итого:			1001,88

Итого, максимальное потребление компонентов, запитанных от блока питания, составляет 201,88 Вт. Был выбран блок питания с ближайшим номиналом выше вычисленного значения мощности 250 Вт.

Схемы принтера. Для составления электрических схем необходимо разработать структурную схему. Она состоит из микроконтроллера, драйверов ШД, блоков датчиков, двигателей, нагрева/охлаждения, интерфейсов. Схема представлена на рис. 1.

Далее необходимо разработать электрическую монтажную схему принтера и схему электрических соединений (рис. 2, 3).

Кроме рассмотренных компонентов, для создания 3D-принтера потребуется реле SSR-40 DA, концевые выключатели EndStop, термистор Semitec 104GT-2, автоматический выключатель 4A, графический контроллер LCD 12864.

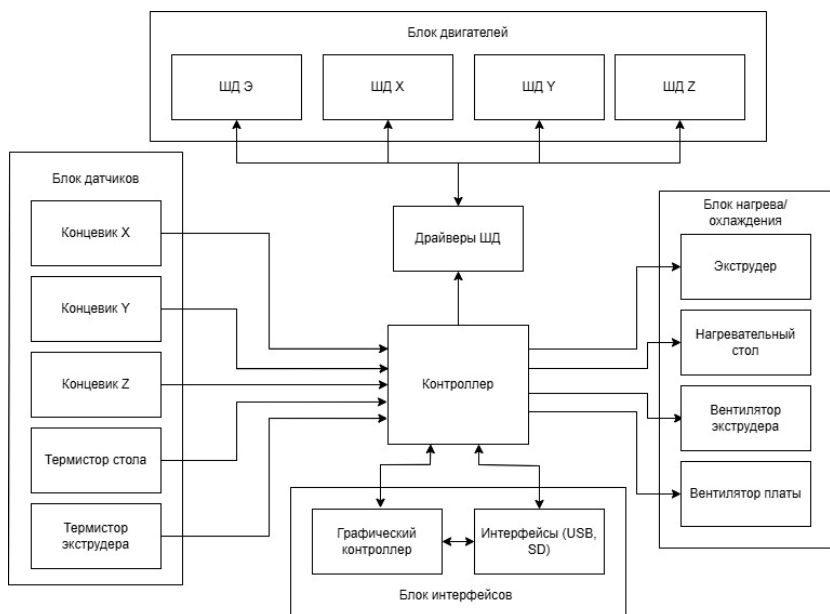


Рис. 1. Структурная схема принтера

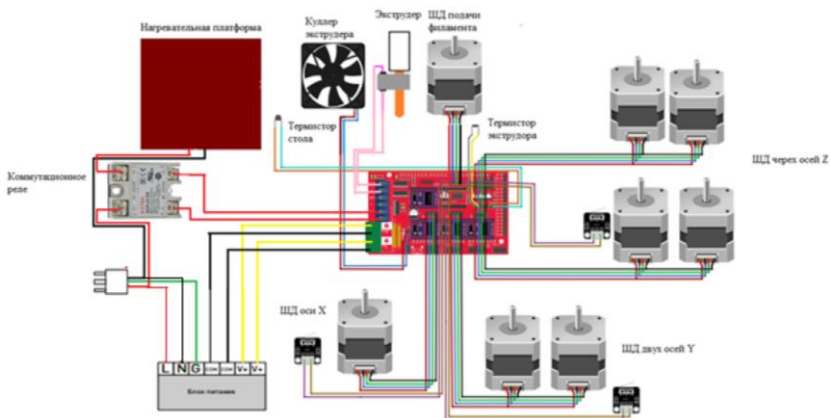


Рис. 2. Электрическая монтажная схема принтера

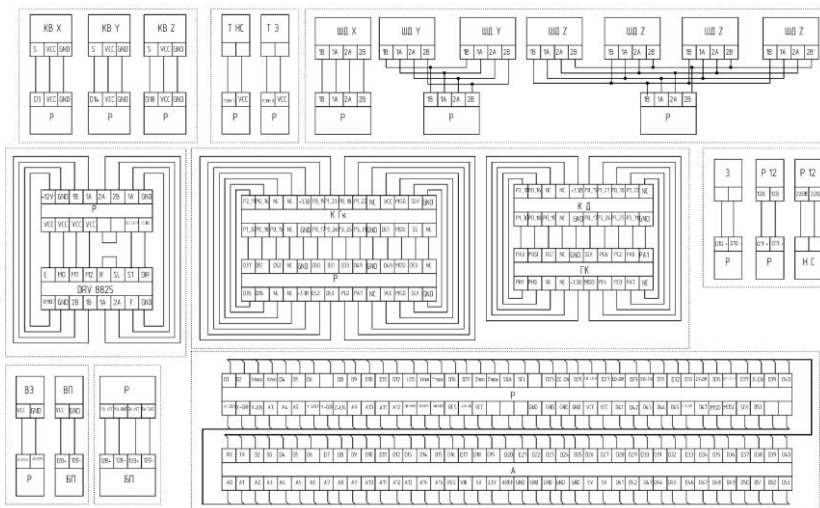


Рис. 3. Схема электрических соединений

Заключение. В результате решения поставленных задач был проведен и обоснован выбор элементной базы принтера, разработан проект аппаратной части 3D-принтера. Решение позволяет сконструировать аппаратную часть из дешевых доступных компонентов.

Общая стоимость компонентов составила примерно 17 800 руб. при общей стоимости подобных принтеров, начинающейся от 40 000 руб.

Кроме того, эти схемы унифицированы и позволяют использовать другие компоненты в зависимости от задач.

Библиографический список

1. Шаговые двигатели [Электронный ресурс]. – URL: <https://3d-diy.ru/wiki/cnc/stepper-motor/> (дата обращения: 19.05.2024).
2. Аверченко А.П., Молдагулова А.А., Нурмаганов Р.С. Выбор драйвера для шагового двигателя MT-1703HD200AW. Сравнение между DRV8825 и A4988 // Молодой ученый. – 2019. – № 50 (288). – С. 4–6.
3. Экструдеры для 3D-принтеров [Электронный ресурс]. – URL: https://3dprob.ru/ekstrudery_dlya_3d_printerov/ (дата обращения: 19.05.2024).

Сведения об авторах

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

Белоусов Михаил Игоревич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: mikhail.belousov.2002@mail.ru

Жигалов Артем Валерьевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: artem.zhigalov.27@mail.ru

Богатырев Роман Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: romanbog2002@gmail.com

Белослудцев Юрий Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: yuragiOG@gmail.com

О.А. Полякова, А.С. Деменев

Пермский национальный исследовательский
политехнический университет, Пермь

ПРОЕКТИРОВАНИЕ МОДУЛЯ УПРАВЛЕНИЯ ИНФОРМАЦИОННЫМ КОНТЕНТОМ В МОБИЛЬНОМ ПРИЛОЖЕНИИ

Исследуются методы проектирования систем управления контентом, а также методы его хранения и обработки. Рассмотрена автоматизированная система управления контентом на примере информационных статей в мобильном приложении.

Ключевые слова: управление контентом, документ-ориентированные базы данных.

O.A. Poliakova, A.S. Demenev

Perm National Research Polytechnic University, Perm

DESIGNING A MODULE FOR INFORMATION CONTENT MANAGEMENT IN A MOBILE APPLICATION

The authors study the methods of designing content management systems, as well as methods of its storage and processing. An automated content management system is considered on the example of information articles in a mobile application.

Keywords: content management, document-oriented databases.

Мобильные устройства стали неотъемлемой частью повседневной жизни, поэтому удобный доступ к информации играет ключевую роль в принятии решений. Обеспечение клиентов актуальными и полезными материалами становится бизнес-необходимостью для компаний.

Разработка цифрового модуля с тематическими информационными статьями в мобильном приложении компании позволит повысить эффективность взаимодействия с клиентами и уровень сервиса, предоставляемого компанией [1].

Современные технологии работы с текстами подразумевают деление информационных текстов на логические блоки, где контент хранится сплошным текстом с определенной разметкой для стилизации. Разработка

мобильного модуля управления информационным контентом, выполненная авторами, меняет технологические подходы к парсерам разметки для корректной обработки ссылок на изображения и взаимодействия с пользователями через интерактивные элементы [2].

Авторами разработан новый способ организации контента, отличный от классического подхода, в котором выделено понятие «блок» как отдельная сущность. «Блок» может хранить в себе информацию о типе контента – текст, изображение, интерактивный элемент; его содержание и положение в общем документе.

Добавление новых «блоков» контента упрощается, поскольку блоки не зависят друг от друга и не изменяют своего положения без дополнительных манипуляций. Также в этом случае повышается гибкость формирования текстов, например, возможность реализации метода drag-and-drop в редакторе для компоновки статьи в единое целое. Скрытие блока в отображении также становится легко реализуемой – «блок» может содержать информацию о том, как именно его отображать. Кроме того, у каждой статьи есть автор и для организации статей используются понятия «разделов» или «тегов», под которыми они объединяются.

При использовании мобильного приложения критически важно получать информацию с сервера как можно быстрее.

Для решения такой задачи использована документ-ориентированная БД MongoDB, данные в которой хранятся не в таблицах, а отдельными сущностями – документами, атрибуты которых содержат как простые типы данных, например строки или числа, так и более сложные – координаты, массивы, другие документы.

Таким образом, при использовании документ-ориентированных баз данных применен подход к хранению данных в виде «документа», который содержит информацию по описанию сущности.

Отображаемая на экране мобильного приложения информация об авторе, тегах статьи, блоках и других данных хранится единым документом и в таком формате передается на устройства пользователей.

Архитектура сущности «статья» представлена на рис. 1.

В сущности «блок контента» предусмотрены атрибуты, содержащие тип контента, его позицию в статье и его содержимое.

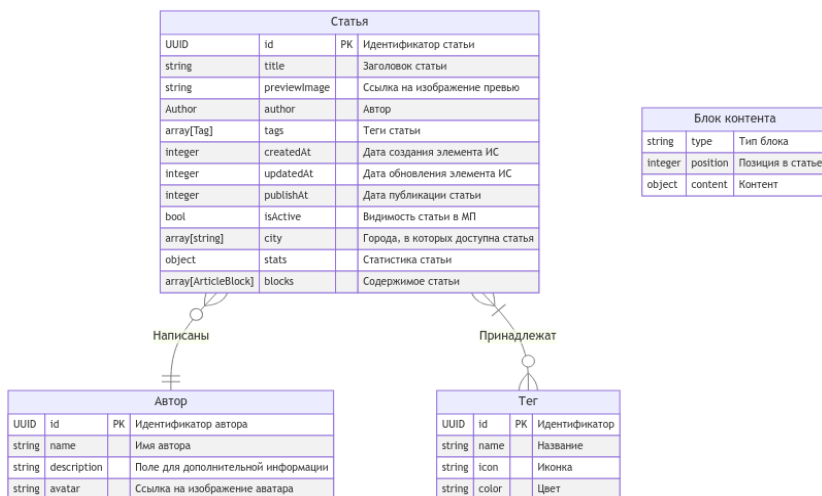


Рис. 1. Представление архитектуры сущности «статья»

В документ-ориентированных базах данных для представления сущности «связь» используется вложенность документов для объединения данных. Для синхронизации и обновления данных сущностей в статьях разработаны отдельные коллекции документов для авторов и тегов, которые находятся в общем хранилище.

На рис. 2 представлен алгоритм обновления тега.

Обновление происходит автоматически не только в коллекции «тегов», но и в коллекции статей, где обновляются все элементы, содержащие обновляемый тег.

Таким образом, поддерживается синхронизация данных используемых тегов, т.е. при добавлении или обновлении статьи актуальная версия тега берется из коллекции тегов, а несуществующие идентификаторы игнорируются и не размещаются в окончательном документе.

В данной статье исследуется возможность перехода от хранения данных для информационной статьи в формате обычного текста с классической разметкой к более гибкой структуре документа, содержащего полное описание для корректного отображения.

Серверное программное обеспечение разработано с использованием языка программирования Python, диаграмма классов для отображения связей между сущностями документов представлена на рис. 3.

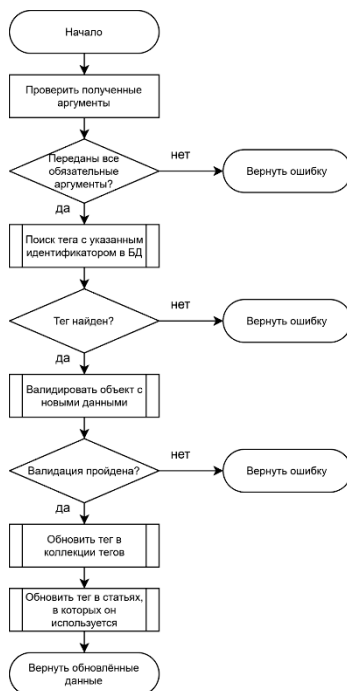


Рис. 2. Алгоритм обновления тега

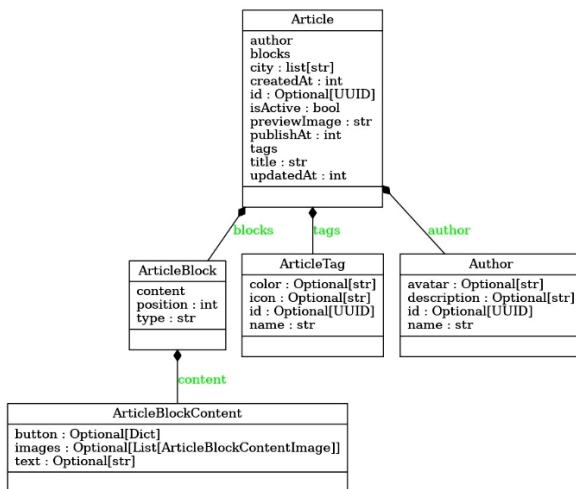


Рис. 3. Диаграмма классов для отображения связей между сущностями документов

Разработка мобильного модуля управления информационным контентом решает проблему с малой функциональностью классической разметки путем перехода к автоматизированному управлению независимыми блоками контента и их взаимодействием с формированием единого документа.

Библиографический список

1. Осипов Д.Л. Технологии проектирования баз данных: практическое руководство. – М.: ДМК Пресс, 2019. – 498 с.
2. Фаулер М., Садаладж П.Дж. NoSQL: новая методология разработки нереляционных баз данных / пер. с англ. – М.: Вильямс, 2013. – 192 с.

Сведения об авторах

Полякова Ольга Андреевна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: olgastratum@mail.ru

Деменев Андрей Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: demeneva06@pstu.ru

**Ю.С. Белослудцев, М.И. Белоусов, Р.С. Богатырев,
А.В. Жигалов, А.Л. Погудин**

Пермский национальный исследовательский
политехнический университет, Пермь

**ОСНОВАНИЕ ВЫБОРА ТЕХНОЛОГИИ ПЕЧАТИ
ЛАБОРАТОРНОГО СТЕНДА 3D-ПРИНТЕРА
НА БАЗЕ ARDUINO**

В статье рассмотрены существующие технологии 3D-печати, проанализированы доступные материалы печати, разработаны детали для сборки 3D-принтера.

Ключевые слова: 3D-принтер, стереолитография, SLS, MJM, LOM, 3D printing, FDM, пластик, нейлон, поликарбонат, PETT, крепление, шарнир.

**Yu.S. Belosludtsev, M.I. Belousov, R.S. Bogatyrev,
A.V. Zhigalov, A.L. Pogudin**

Perm National Research Polytechnic University, Perm

**THE BASIS FOR CHOOSING THE PRINTING
TECHNOLOGY OF A LABORATORY STAND
3D PRINTER BASED ON ARDUINO**

The article discusses existing 3D printing technologies, analyzes available printing materials, and develops details for assembling a 3D printer.

Keywords: 3D printer, stereolithography, SLS, MJM, LOM, 3D printing, FDM, plastic, nylon, polycarbonate, PETT, fastening, hinge.

В настоящее время технологии 3D-печати становятся востребованными во все новых сферах и отраслях производства. С каждым годом 3D-принтерам находят применение в отраслях, где ранее их использование казалось невозможным. На текущем этапе развития технология 3D-печати способна работать не только с различными видами пластика, но и многими другими материалами, такими как металл, керамика, биоматериалы и т.д.

Однако, несмотря на огромное количество положительных сторон использования 3D-принтеров, у современных принтеров имеется ряд недостатков. Одним из главных «минусов» 3D-принтера является цена.

Актуальность темы обусловлена необходимостью повышения квалификации студентов в области аддитивной печати, а также общей осведомленности в сфере новых способов производства деталей.

Объект исследования – 3D-принтер. Предмет исследования – создание конструкции 3D-принтера. Цель проекта – разработка лабораторного стенда «Исследование 3D-принтера». Для достижения поставленной цели требуется решение следующих задач:

- изучить и проанализировать виды существующих 3D-принтеров;
- рассмотреть, какие материалы печати существуют;
- рассмотреть, какие детали для сборки 3D-принтера необходимо разработать самостоятельно.

Обзор технологий 3D-печати. Технология стереолитографии – это первая технология трехмерной печати. Она использует жидкий фотополимер, который затвердевает под действием ультрафиолетового лазера. Процесс происходит последовательным наложением тонких слоев материала. «Плюсы» включают высокое качество, возможность создания крупных и прочных деталей, а также сложных форм. Однако у этой технологии есть и недостатки: ограниченный выбор материалов, отсутствие цветной печати, низкая скорость и высокая стоимость оборудования и материалов [1].

Выборочное лазерное спекание (SLS) – метод 3D-печати, использующий порошок вместо фотополимера. Порошок наносится и спекается лазером, создавая слои. Преимущества SLS: разнообразие материалов, высокая сложность моделей, скорость печати, возможность мелкосерийного производства. Недостатки: требуется мощный лазер, герметичная камера, менее точно, длительные этапы прогрева и остывания, финишная обработка. Цена высока, но с истечением патента цены могут снизиться, сделав технологию доступнее [1].

Метод многоструйного моделирования (MJM) – один из способов 3D-печати, где материал подается через сопла и наносится слоями. Процесс напоминает струйную печать, но с нагреванием материала. Материалы включают пластики, фотополимеры, воск и могут быть комбинированы. Поддерживающие структуры требуются для элементов под углом. Удаление поддержек может потребовать специальной обработки. Преимущества включают высокое разрешение, многоцветную печать, и возможность комбинировать материалы. Недостатки – необходимость поддержки для выступающих элементов и ограниченный выбор материалов [2].

Метод Laminated Object Manufacturing (LOM) – это вид трехмерной печати, где тонкие листы материала разрезаются лазером или лезвием, затем соединяются. Материалы могут быть бумагой, керамикой или металлом. Пример – Mcoor IRIS, использующий бумагу для печати. Разрешение 5760×1440×508 dpi, размер объектов до 256×169×150 мм. Печать в цвете с миллионом цветов. Процесс включает загрузку бумаги в 2D-принтер, печать слоев, резку и склейку листов. Недостатки – ограниченный выбор материалов, отходы, необходимость финишной обработки. Положительные стороны – цветная печать, доступность бумаги, возможность больших изделий без поддержки [2].

Технология 3D Printing, подобная принтеру выборочного лазерного спекания, использует порошок (гипсовый композит), который склеивается связующим веществом. Слои модели создаются нанесением порошка и вводом жидкого клея струйной печатающей головкой. После нагревания и удаления лишнего порошка модель требует обработки специальным составом для улучшения свойств. Возможна цветная печать с добавлением красителей в связующее вещество. Положительные стороны: создание сложных моделей без поддержки и цветная печать. Однако есть ограниченный выбор материалов, необходимость финишной обработки и низкая прочность деталей [3].

Принтеры FDM создают модели послойно, нагревая и выдавливая термопластичный материал через сопло. Они могут использовать различные материалы, включая пластик и сплавы металлов.

Недостатки включают низкую скорость и разрешение, проблемы с фиксацией моделей и необходимость поддерживающих структур. Несмотря на это, принтеры FDM становятся популярными из-за доступности и широкого выбора материалов.

Срок патента на технологию истек в 2009 г., что привело к увеличению конкуренции и снижению цен. Эти принтеры гибкие и могут использоваться для различных задач, от прототипирования до производства готовых изделий [3].

Сравнение материалов печати. В ходе разработки 3D-принтера важно определить, какие материалы возможно использовать в печати. Исходя из анализа существующих материалов печати, была сделана следующая обобщающая таблица.

Сравнение материалов

Вид материала	Достоинства	Недостатки
ABS пластик	Нетоксичность, влагостойкость, масло-стойкость, стойкость к кислотам и щелочам, широкий диапазон эксплуатационных температур, низкая стоимость, простая обработка готовых изделий	Несовместимость с пищевыми продуктами, чувствительность к УФ-излучению, высокий коэффициент теплового сужения
PLA пластик	Нетоксичность, низкий коэф. трения, низкий коэффициент теплового сужения, менее хрупкий и более гибкий, чем ABS	Низкая долговечность, гигроскопичность, более высокая цена
Ударопрочный полистирол	Устойчивость к ударным воздействиям, стойкость к перепадам температур, способность растворяться в лимонене	Высокая стоимость, повышенные требования к принтеру
Нейлон	Устойчивость к химическому воздействию, легкость, гибкость	Высокая стоимость, повышенные требования к принтеру
Поликарбонат	Широкий диапазон рабочих температур, светопропускание, биологическая инертность, универсальность	Высокая стоимость, повышенные требования к принтеру
PETT	Прочность, быстрое затвердевание	Высокая стоимость, повышенные требования к принтеру

В ходе исследования было принято решение, что ABS и PLA пластики обладают всеми необходимыми свойствами, а также более низкой стоимостью материалов [4].

Разработка собственных деталей конструкции. В процессе конструирования корпуса, узлов и прочих элементов 3D-принтера зачастую возникают ситуации, когда нужных элементов для связи деталей или не существует на рынке, или доступ к ним по тем или иным причинам ограничен в связи с высокой ценой или неприемлемыми сроками доставки.

Для установки шагового двигателя в нижней части корпуса для передвижения по вертикальной оси Z необходимо было разработать конструкцию с учетом геометрических характеристик шагового двигателя и расположения встроенных креплений. Результатом работы стала деталь, удовлетворяющая вышеперечисленным требованиям (рис. 1).

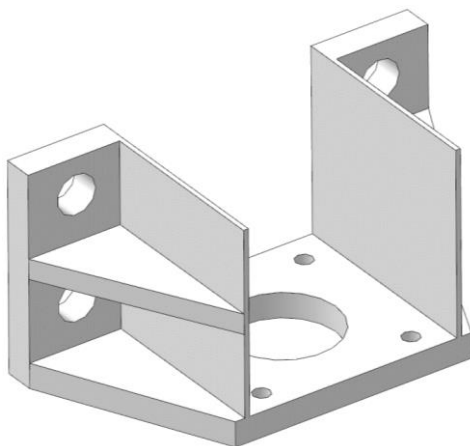


Рис. 1. Крепление шагового двигателя

Также потребовался шарнир, на котором крепились бы компоненты, необходимые для передвижения по оси X , в результате проектирования была получена следующая деталь (рис. 2).

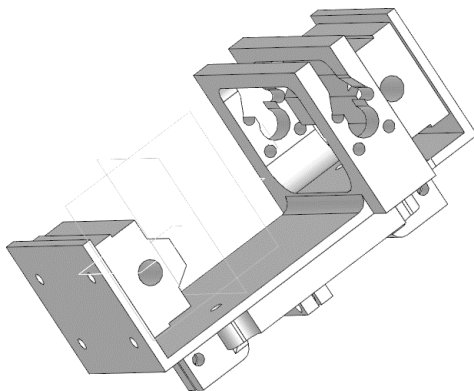


Рис. 2. Шарнир оси X

Также для установки под углом для передвижения по оси X требовалось разработать отдельное крепление для шагового двигателя с возможностью крепления к шарниру оси X .

В результате проектирования были добавлены крепления, позволяющие осуществить задуманные идеи (рис. 3).

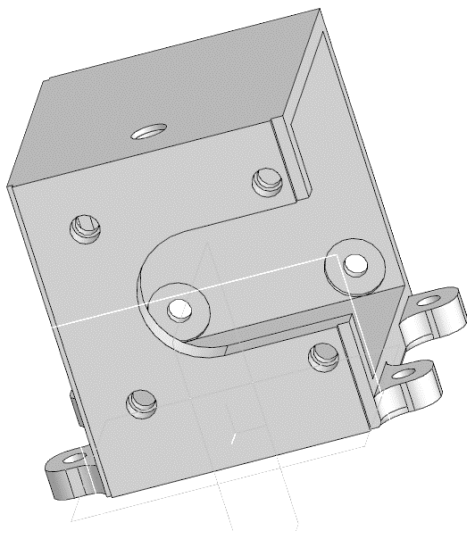


Рис. 3. Альтернативное крепление шагового двигателя

Заключение. В результате рассмотрения различных существующих видов 3D-принтеров было принято решение, что принтеры послойного плавления подходят под задачу разработки лабораторного стенда лучше всего. Такое решение было принято в связи с широким распространением такого типа принтеров, доступности компонентов и т.д.

В результате проектирования собственных деталей корпуса были получены следующие выводы: проектирование деталей оказалось эффективным в случаях, когда аналогов, удовлетворяющим всем требованиям, нет или доступ к ним ограничен вследствие таких причин, как высокая цена, недоступность из-за производства в недружественных странах и т.д. Далее в конструкторском разделе будут проведены более подробные испытания на возможность использования и эффективность.

Библиографический список

1. 3D-печать как новое научно-техническое направление [Электронный ресурс]. – URL: <https://cyberleninka.ru/article/n/3d-pechat-kak-novoe-nauchno-tehnicheskoe-napravlenie/viewer> (дата обращения 10.04.2024).
2. История и технологии трехмерной печати / Н.В. Кушнир, А.В. Кушнир, А.М. Геращенко, А.В. Тыртышный // Научные труды КубГТУ, 2015. – № 5.

3. Шкуро А.Е., Кривоногов П.С. Технологии и материалы 3D-печати // учеб. пособие. – Екатеринбург: Изд-во Урал. гос. лесотехн. ин-та, 2017.

4. Технологии 3D-печати. Принципы, возможности, расходные материалы, цены [Электронный ресурс]. – URL: https://www.ixbt.com/printer/3d/3d_tech.shtml (дата обращения: 10.04.2024).

Сведения об авторах

Белослудцев Юрий Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-1б, г. Пермь, e-mail: yuragiOG@gmail.com

Белоусов Михаил Игоревич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-1б, г. Пермь, e-mail: mikhail.belousov.2002@mail.ru

Богатырев Роман Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-1б, г. Пермь, e-mail: romanbog2002@gmail.com

Жигалов Артем Валерьевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-1б, г. Пермь, e-mail: artem.zhigalov.27@mail.ru

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

**С.А. Камянецкий, С.И. Келлер,
Д.О. Тронина, А.Л. Погудин**

Пермский национальный исследовательский
политехнический университет, Пермь

**РАЗРАБОТКА ИМИТАЦИОННОЙ МОДЕЛИ
ЛАБОРАТОРНОГО СТЕНДА ПО ИССЛЕДОВАНИЮ
БАЗОВЫХ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ**

Сделан краткий анализ систем схмотехнического моделирования. Дано обоснование выбора системы схмотехнического моделирования. Разработана имитационная модель лабораторного стенда по исследованию базовых логических элементов. Предложен вариант методики проведения лабораторной работы.

Ключевые слова: схмотехника, стенды, системы схмотехнического моделирования, логические элементы.

**S.A. Kamyanetsky, S.I. Keller,
D.O. Tronina, A.L. Pogudin**

Perm National Research Polytechnic University, Perm

**DEVELOPMENT OF A SIMULATION MODEL
OF A LABORATORY STAND FOR THE STUDY
OF BASIC LOGIC ELEMENTS**

Provides a brief analysis of circuit modeling systems. The rationale for the choice of a circuit modeling system is given, a simulation model of a laboratory stand for the study of basic logic elements is developed. A variant of the laboratory work methodology is proposed.

Keywords: circuit engineering, stands, circuit modeling systems, logical elements.

В настоящее время нас окружают множество электронных устройств, которые используются во всех сферах жизни, как в повседневных делах, так и на работе. С каждым годом их становится все больше и больше. Схмотехника – научно-техническое направление, занимающееся проектированием, созданием и анализом электронных и электрических схем.

Актуальность темы обусловлена необходимостью изучения схмотехники для расширения возможности в разработке разного рода

устройств. Например тем, кто работает с микроконтроллерами и встроенными системами, нужно иметь хотя бы базовые знания по схемотехнике, поскольку она позволяет осуществлять надлежащий контроль и взаимодействие с физическими устройствами. Для их правильной работы часто требуется специализированное оборудование и схемы. Понимание схемотехники может значительно расширить возможности в программировании и помочь в устранении неполадок. При изучении любого предмета требуется начинать с самого простого и постепенно продвигаться к более сложным темам. В схемотехнике множество элементов состоят из разного сочетания базовых логических элементов, поэтому в качестве начала по изучению схемотехники было выбрано исследование базовых логических элементов.

Объект исследования – лабораторный стенд (имитационная модель) по исследованию базовых логических элементов.

Предмет исследования – процесс по разработке, реализации и внедрению лабораторного стенда (имитационной модели) для исследования базовых логических элементов.

Цель проекта – разработка лабораторного стенда по исследованию базовых логических элементов.

Для достижения поставленной цели требуется выполнить следующие задачи:

- 1) выбрать систему схемотехнического моделирования для выполнения лабораторных работ;
- 2) разработать имитационную модель лабораторного стенда для исследования базовых логических элементов в выбранной системе схемотехнического моделирования.

Системы схемотехнического моделирования. При выборе системы схемотехнического моделирования нужно учитывать возможности системы и ее цену.

Среди виртуальных систем схемотехнического моделирования рассматривались следующие системы: OrCAD, P-CAD, Multisim, Proteus, Qucs, Circuit Simulator, IdealCircuit, Crocodile Technology 3D, Delta Design, Tinkercad, Fritizing.

Среди натуральных стендов по схемотехнике рассматривались следующие стенды: ОЭИС-ПРОФИ-СР, ОЭИС-НРЦ, ЭИС-НРЦ [1].

Исходя из анализа можно сделать вывод, что каждый из рассмотренных стендов имеет свои достоинства и недостатки.

Виртуальные системы базируются от простых с малым количеством элементов, которые созданы только для учебных целей и до мощных систем с огромным количеством элементов, позволяющих автоматизировать все стадии проектирования электронных устройств. В мощных системах можно заниматься подготовкой принципиальных схем, моделированием процессов в аналоговых и цифровых цепях, компоновкой и трассировкой печатных плат, редактированием и расширением библиотек компонентов. Цена рассматриваемых систем базировалась от бесплатных до систем за несколько сотен тысяч рублей.

Цены на натурные стенды зависят от их возможностей, т.е. чем больше лабораторных на них можно выполнять, тем дороже они будут стоить. Самый простенький стенд можно купить приблизительно за 100 000 руб. Цена рассматриваемых стендов базировалась от 100 до 400 тыс. руб.

К наиболее интересным можно отнести: Multisim, предоставляющий множество возможностей для проектирования электронных устройств, бесплатный инструмент Fritizing, предназначенный для обучения и компьютерного прототипирования проектов на основе Arduino, бесплатное веб-приложение для 3D-дизайна, электроники и программирования Tinkercad.

Для разработки лабораторного стенда (имитационной модели) по исследованию базовых логических элементов был выбран Multisim. Multisim – приложение для создания и тестирования электрических цепей в схемотехнике. Оно включает сотни виртуальных электронных компонентов и измерительных устройств. Также для проектирования, анализа и отладки электрических схем Multisim умеет моделировать работу и взаимодействие между разными электронными компонентами и измерительными устройствами. Если иметь студенческий аккаунт на сайте National Instruments, то можно бесплатно использовать учебную версию Multisim. Учебные версии отличаются от полной урезанным функционалом, отсутствие которого не должно повлиять на выполнение лабораторных работ.

Разработка лабораторного стенда по исследованию базовых логических элементов. В ходе работы по изучению базовых логических элементов от студента требуется изучить теорию [2], построить схемы (имитационные модели), изучить и проанализировать

их работу, сделать вывод о проделанной работе и ответить на закрепляющие вопросы.

Для создания имитационной модели для изучения логического элемента НЕ (отрицание сигнала) понадобится сам элемент НЕ (NOT), источник питания (VCC), земля (GND/DGND), переключатель (SPDT) и для наблюдения за выходным сигналом – красный индикатор (PROBE_RED) (рис. 1).

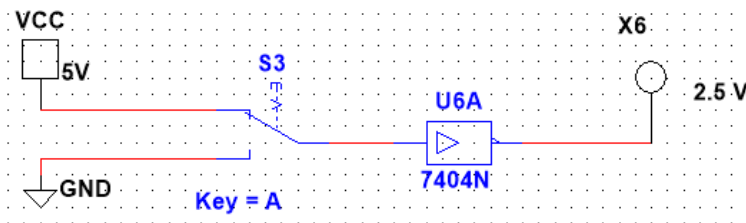


Рис. 1. Схема для демонстрации работы логического элемента НЕ в среде Multisim

Для создания имитационной модели для изучения логического элемента И (логическое умножение) понадобится сам элемент И (AND2), источник питания (VCC), земля (GND/DGND), два переключателя (SPDT) и для наблюдения за выходным сигналом красный индикатор (PROBE_RED) (рис. 2).

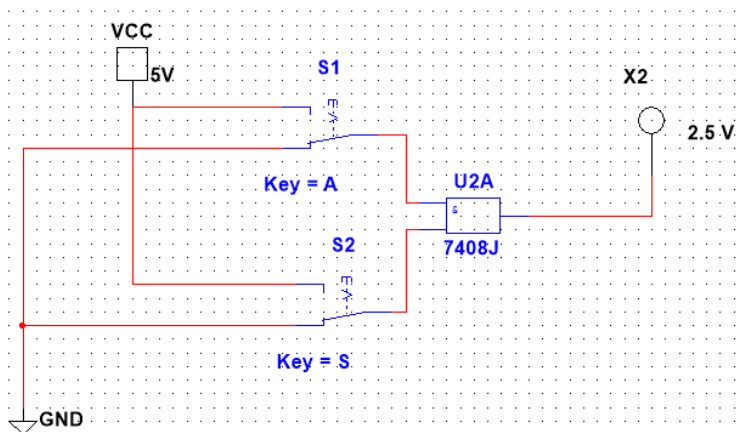


Рис. 2. Схема для демонстрации работы логического элемента И в среде Multisim

Для создания имитационной модели для изучения логического элемента И-НЕ (логическое умножение с последующей инверсией результата) понадобится сам элемент И-НЕ(NAND2), источник питания (VCC), земля (GND/DGND), 2 переключателя (SPDT) и для наблюдения за выходным сигналом – красный индикатор (PROBE_RED) (рис. 3).

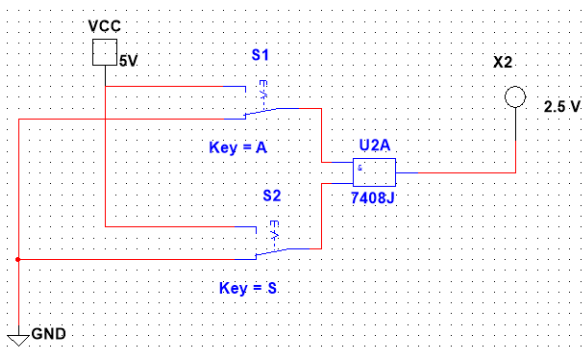


Рис. 3. Схема для демонстрации работы логического элемента И-НЕ в среде Multisim

Для создания имитационной модели для изучения логического элемента ИЛИ (логическое сложение) понадобится сам элемент ИЛИ(OR2), источник питания (VCC), земля (GND/DGND), 2 переключателя (SPDT) и для наблюдения за выходным сигналом – красный индикатор (PROBE_RED) (рис. 4).

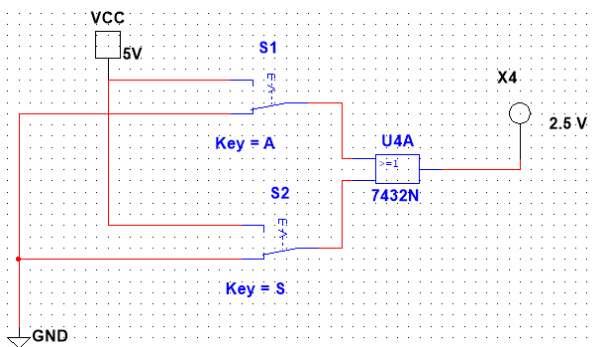


Рис. 4. Схема для демонстрации работы логического элемента ИЛИ в среде Multisim

Для создания имитационной модели для изучения логического элемента ИЛИ-НЕ (логическое сложение с последующей инверсией результата) понадобится сам элемент ИЛИ-НЕ(NOR2), источник питания (VCC), земля (GND/DGND), 2 переключателя (SPDT) и для наблюдения за выходным сигналом – красный индикатор (PROBE_RED) (рис. 5).

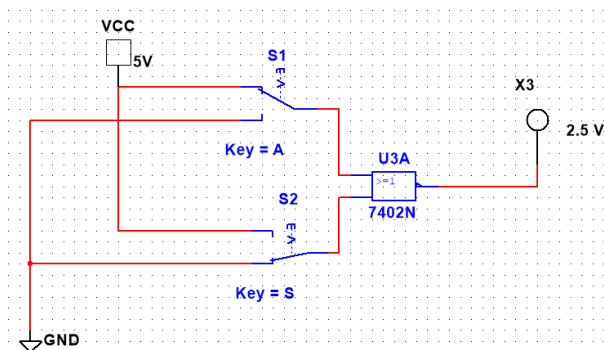


Рис. 5. Схема для демонстрации работы логического элемента ИЛИ-НЕ в среде Multisim

Для создания имитационной модели для изучения логического элемента исключающее ИЛИ (разделительная дизъюнкция) понадобится сам элемент исключающее ИЛИ (XOR2), источник питания (VCC), земля (GND/DGND), 2 переключателя (SPDT) и для наблюдения за выходным сигналом – красный индикатор (PROBE_RED) (рис. 6).

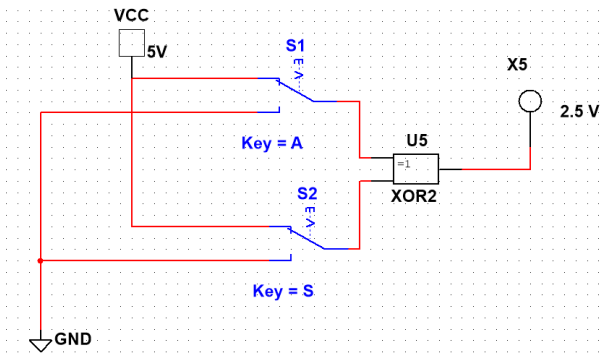


Рис. 6. Схема для демонстрации работы логического элемента исключающее ИЛИ в среде Multisim

Для создания имитационной модели для изучения логического элемента исключающее ИЛИ-НЕ (разделительная дизъюнкция с последующей инверсией результата) понадобится сам элемент исключающее ИЛИ-НЕ (XNOR2), источник питания (VCC), земля (GND/DGND), 2 переключателя (SPDT) и для наблюдения за выходным сигналом – красный индикатор (PROBE_RED) (рис. 7).

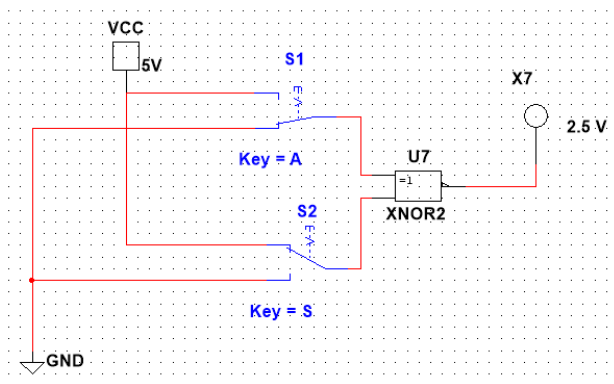


Рис. 7. Схема для демонстрации работы логического элемента исключающее ИЛИ-НЕ в среде Multisim

В исследованиях разработаны и описаны схемы (имитационные модели), с помощью которых можно изучить базовые логические элементы.

Заключение. Достоинством виртуальных систем над натурными стендами является их физическое отсутствие. Благодаря этому они могут не привязываться к конкретному кабинету, а устанавливаться в компьютерном классе или на устройствах обучающихся, имеющих достаточные характеристики для запуска системы схемотехнического моделирования. Также в виртуальных системах нет вероятности вывода из строя в результате неправильного использования, например, там не может перегореть контакт. В данной статье в результате анализа систем схемотехнического моделирования для выполнения работ была выбрана среда Multisim с обоснованием ее выбора. Были разработаны и описаны имитационные модели для исследования базовых логических элементов с предложенной методикой проведения лабораторных работ.

Библиографический список

1. Стенды «Электроника и схемотехника» купить в ООО «ЛАБ-СИС» [Электронный ресурс]. – URL: <https://labsys.ru/product/elektronika/elektronika-i-skhemotekhnika/> (дата обращения: 17.05.2024).
2. Закарлюк Н.М. Цифровые устройства автоматики и телемеханики. – Екатеринбург, 2009. – 160 с.

Сведения об авторах

Камянецкий Сергей Александрович – бакалавр Пермского национального исследовательского политехнического университета гр. КС-20-16, г. Пермь, e-mail: kamyanetskysergei@yandex.ru

Келлер Степан Ильич – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: stepan.keller@gmail.com

Тренина Дарья Олеговна – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: dasha.tronina17@gmail.com

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

К.Н. Шестаков, В.О. Кушев

Пермский государственный национальный
исследовательский университет, Пермь

АНАЛИЗ ПОТРЕБНОСТЕЙ ПОЛЬЗОВАТЕЛЕЙ ПРИ ПОИСКЕ НАПАРНИКОВ ДЛЯ МНОГОПОЛЬЗОВАТЕЛЬСКИХ ВИДЕОИГР

В статье анализируются потребности пользователей в поиске напарников для многопользовательских видеоигр. Рассматривается классификация многопользовательских игр на казуальные / кооперативные и соревновательные, а также социальная сторона игр. Определены ключевые критерии, которые следует учитывать при разработке системы поиска напарников, включая фильтрацию поиска по классификации игр, указание уровня владения игрой, функционал библиотеки игр и поиск новых игр. Также выделены социальные потребности пользователей, такие как добавление в друзья, подписка на обновления, создание игровых событий и возможность оставить жалобу на игрока.

Ключевые слова: информационная система, анализ критериев, онлайн-видеоигры, поиск напарников.

K.N. Shestakov, V.O. Kushev

Perm State National Research University, Perm

ANALYSIS OF USER NEEDS IN SEARCHING TEAMMATES FOR MULTIPLAYER GAMES

In the article, the needs of users in searching teammates for multiplayer videogames are analyzed. Classification of multiplayer games and social aspect of games are discussed. The key criteria to be taken into account when developing a teammate search system are determined, including filtering by game classification, specifying the level of game proficiency, game library functionality, and searching for new games. Social user needs, such as adding friends, subscribing to updates, creating gaming events, and the ability to report a player, are also highlighted.

Keywords: information system, criteria analysis, online videogames, search for teammates.

Многопользовательские игры стали неотъемлемой частью современной культуры развлечений. Однако, несмотря на широкое распространение, многие игроки сталкиваются с трудностями при поиске подходящих напарников для совместной игры. Цель данного исследования – проанализировать потребности пользователей в поиске напарников для многопользовательских игр и определить ключевые крите-

рии, которые следует учитывать при разработке соответствующей системы.

Для проведения анализа потребностей пользователей рассмотрим основные критерии подбора напарников, а также проанализируем социальную сторону многопользовательских игр.

Чаще всего при использовании внутренних систем поиска игроков для игры пользователи отбираются с оглядкой на регион и уровень игрока (в конкретной игре). И если для поиска соперников такой механизм вполне подходит, то для поиска напарников, особенно для соревновательных игр с ориентацией на командное взаимодействие, хочется поиска более осознанного.

Первое на что могут обратить внимание пользователи при поиске напарников, – это вид многопользовательской игры. Приведем базовую классификацию многопользовательских игр.

Казуальные/кооперативные многопользовательские игры предлагают игрокам возможность сражаться, сотрудничать и взаимодействовать в виртуальном мире. Они обычно включают в себя самые различные жанры, такие как экшн, приключения, стратегии, головоломки и т.д.

Такие игры часто предоставляют пользователям разнообразные задания, которые они могут выполнять вместе с другими игроками. Некоторые задания требуют тесного сотрудничества и координации действий. Для достижения целей необходимо построить команду, разработать стратегию и превзойти себя вместе с другими игроками.

Главная особенность, которую стоит выделить, кооперативные игры построены в основном не на соревновательных элементах между игроками, а на взаимодействии против игрового мира. Если у людей в команде разный уровень владения механиками игры, это несильно повлияет на их общий игровой опыт.

Соревновательные многопользовательские игры предлагают игрокам объединяться в команды или же выступать по одиночке против таких же реальных игроков. Основной целью является победа в текущей игровой сессии.

Смысл таких игр заключается в создании стимула для игроков развивать свои навыки, стратегическое мышление и тактику, чтобы победить других игроков и получить признание в игровом сообществе. Они предлагают эмоционально насыщенные и адреналиновые ситуации, где каждый игрок старается использовать свое умение, быстроту реакции и смекалку, чтобы обойти своих соперников.

Соревновательные многопользовательские игры также имеют потенциал для создания профессиональных киберспортивных команд и турниров, где игроки могут заявить о себе и конкурировать за большие призовые фонды и славу. Одной из крупнейших и признанных организаций киберспорта является Федерация компьютерного спорта России [1]. Данная организация проводит различные турниры, создает лиги по соревновательным видеоиграм, включая школьные и студенческие лиги.

Оставим разделение многопользовательских игр на казуальные/кооперативные и соревновательные.

Кроме того, стоит учесть возможную потребность ведения библиотеки своих игр пользователями, необязательно только многопользовательских. С помощью нее игроки смогут оценить схожесть игровых предпочтений потенциальных напарников. А также если дать возможность пользователям оценить свое мастерство в той или иной игре, это может улучшить подбор напарников для многопользовательских игр.

Зафиксируем возможные потребности пользователей при подборе напарников:

- фильтрация поиска сессий/напарников по приведенной классификации;
- указание уровня владения той или иной игрой;
- функционал добавления игр в свою библиотеку с дальнейшей классификацией на играл, играю и собираюсь поиграть;
- поиск новых игр для занесения в библиотеку.

Для изучения дополнительных потребностей пользователей обратимся к еще одной стороне многопользовательских игр – социальной стороне.

Стоит подчеркнуть важность социального аспекта многопользовательских игры. В статье «Влияние многопользовательских игр на социальный капитал и взаимозависимое благополучие в Японии» [2] вывод авторов заключается в том, что онлайн-многопользовательские игры имеют значительное влияние на социальный капитал и взаимозависимое благополучие в Японии. Авторы обнаружили, что участие в онлайн-играх способствует формированию позитивного социального капитала, такого как доверие, взаимодействие и поддержка со стороны других игроков. Кроме того, они заметили,

что игры также способствуют развитию взаимозависимого благополучия, основанного на ощущении принадлежности к группе и поддержке других членов сообщества при достижении общей цели. Авторы утверждают, что эти виды социального капитала могут иметь положительное влияние на психологическое благополучие и качество жизни игроков в Японии.

Для лучшего применения социальных функций системе нужны механизмы, завязанные на коммуникации пользователей. Из таких механизмов можно отметить: добавление пользователей в друзья, подписка на пользователей для отслеживания их обновлений и указание своих предпочитаемых игровых устройств и внешних социальных сетей.

Но, конечно, все позитивные элементы взаимодействия пользователей в онлайн-играх сильно зависят от других игроков. Если к вам в команду попадают недобросовестные или же просто грубо ведущие себя игроки, это сильно повлияет на ваш игровой опыт не в лучшую сторону. Для решения этой проблемы пригодился бы механизм жалоб и последующее рассмотрение жалоб модераторами / администраторами.

Выделим потребности пользователей, связанные с социальной стороной:

- добавление пользователей в друзья;
- подписываться на других пользователей с целью отслеживания обновлений и событий пользователя;
- возможность пользователям создавать игровые события с видимостью для себя и своих друзей;
- указание пользователем своих предпочитаемых игровых платформ и социальных сетей;
- возможность оставить жалобу на игрока.

Проведенный анализ выявил ключевые потребности пользователей, которые следует учитывать при разработке системы поиска напарников для многопользовательских игр. Учитывая потребности пользователей и социальную составляющую игр, можно создать эффективную систему, которая поможет игрокам находить подходящих напарников и улучшит их игровой опыт.

Библиографический список

1. Федерация компьютерного спорта России: официальный сайт [Электронный ресурс] – URL: <https://resf.ru> (дата обращения: 17.04.2024).
2. Shaowen Ni, Ran Dong, Hideo Ueichi. The influence of online multiplayer games on social capital and interdependent well-being in Japan [Электронный ресурс] // Entertainment Computing. – URL: <https://www.sciencedirect.com/science/article/pii/S1875952123000423#s0055> (дата обращения: 12.05.2024).

Сведения об авторах

Шестаков Кирилл Николаевич – магистрант Пермского государственного национального исследовательского университета, гр. ФИТ-23, г. Пермь, e-mail: zeldaone@mail.ru

Кушев Вадим Олегович – кандидат педагогических наук, доцент, доцент, кафедры «Информационные технологии» Пермского государственного национального исследовательского университета, г. Пермь, e-mail: vadimk@psu.ru

Б.К. Колобов, Д.А. Карлов

Пермский государственный национальный
исследовательский университет, Пермь

УЧЕБНАЯ ПРОГРАММА ДЛЯ РАБОТЫ С МОДЕЛЬЮ РОБОТА-МАНИПУЛЯТОРА В СИСТЕМЕ MATLAB

В статье рассмотрены технологии, используемые в процессе разработки модели робота-манипулятора, проанализированы существующие решения и построена модель системы в программе MatLab.

Ключевые слова: робототехника, робот-манипулятор, проектирование, кинематика, динамика.

B.K. Kolobov, D.A. Karlov

Perm State National Research University, Perm

A TRAINING PROGRAM FOR WORKING WITH A MANIPULATOR ROBOT MODEL IN THE MATLAB SYSTEM

The article discusses the technologies used in the process of developing a robot manipulator model, analyzes existing solutions, and builds a model of the system in the Matlab program.

Keywords: robotics, robot manipulator, engineering, kinematics, dynamics.

Актуальность данной работы: отсутствие методических материалов для моделирования робота манипулятора в системе MatLab. В рамках работы рассмотрен алгоритм создания модели робота-манипулятора и методы решения прямой и обратной задачи кинематики, а также инверсивной динамики. В реализацию задачи обратной кинематики включено исследование решения задачи управления роботом с использованием сенсоров и обратных связей.

Алгоритм решения задачи моделирования. Для примера решения задачи использован робот-манипулятор со следующими характеристиками: масса робота – 52 кг; число звеньев – 6; максимальная нагрузка – 6 кг.

На рис. 1 показан общий вид модели робота, для удобства визуализации блоки собраны в подсистемы со своими именами и входами/выходами. Здесь можно увидеть 5 подсистем, в каждой из них нахо-

дятся блоки из библиотек MatLab. По комментариям и названиям подсистем легко понять, что содержится внутри и зачем это нужно. Далее произведен разбор каждой используемой подсистемы.

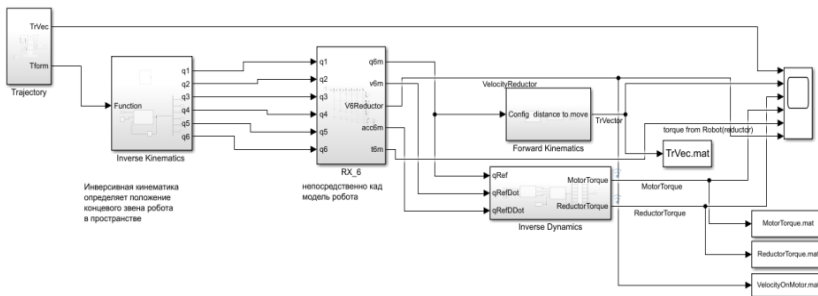


Рис. 1. Общий вид модели робота

Подсистема траектории (рис. 2) используется для задания движения роботу-манипулятору с использованием блоков: трапецеидальной траектории, траектории вращения и преобразователь координат.

Принцип работы следующий: на вход в блок траектории подаются точки в пространстве, пиковая скорость и ускорение, а в блок вращения подается временной интервал, начальный и конечный угол поворота.

Для моделирования максимальных нагрузок на всех шарнирах робота применяются два вида траекторий: движение из максимально вытянутого положения вперед (вдоль оси x) в максимально вытянутое положение вверх (вдоль оси z), при этом фланец направлен вдоль оси x ; движение из максимально вытянутого положения вперед, в максимально вытянутое положение вбок (вдоль оси y).

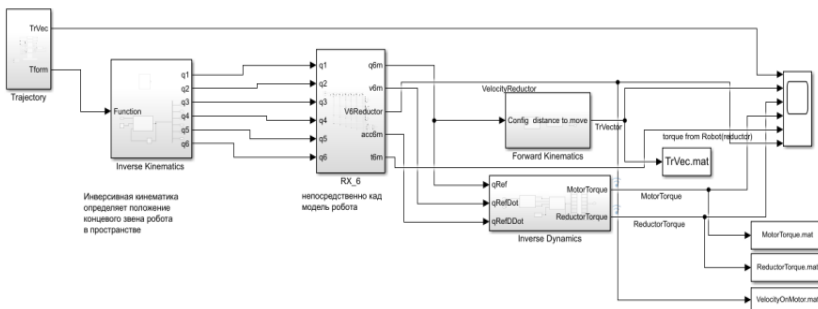


Рис. 2. Подсистема траектории

Далее идет подсистема обратной кинематики (рис. 3), на вход поступает сигнал от блока траектории, устанавливаются значения *Weights*, равными единице, для активации всех шарнирных соединений, в противном случае у робота заблокируются шарниры в соответствии с комбинацией единиц и нулей. Полученные данные разбиваются на потоки и отправляются на соединения.

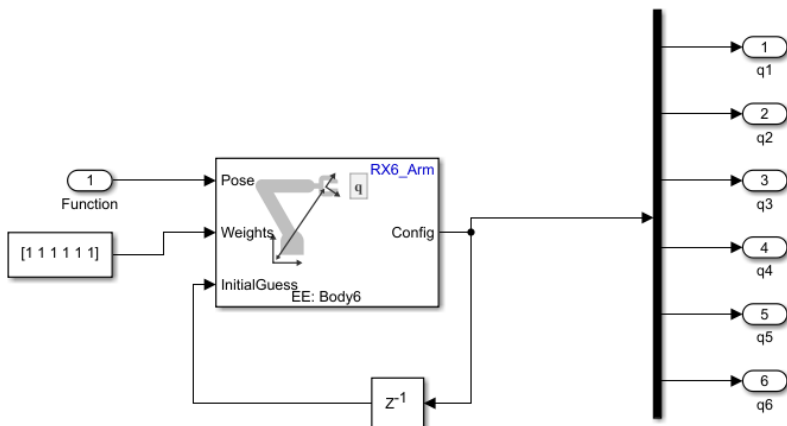


Рис. 3. Подсистема обратной кинематики

На рис. 4 показана структура робота. Она состоит из начальных обязательных параметров: мировые координаты, конфигурактор, функция выполнения расчетов – их достаточно добавить из библиотеки, и настройки не требуется. Далее идет опора робота (рис. 5), шарниры и звенья робота.

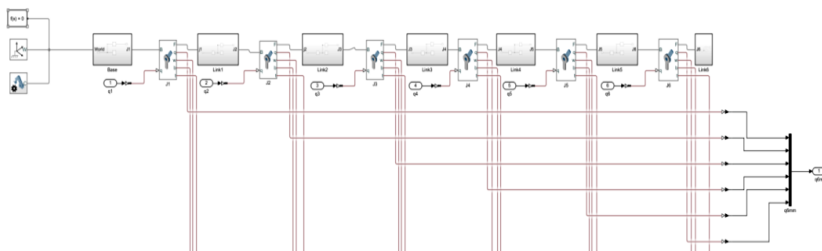


Рис. 4. Структура робота

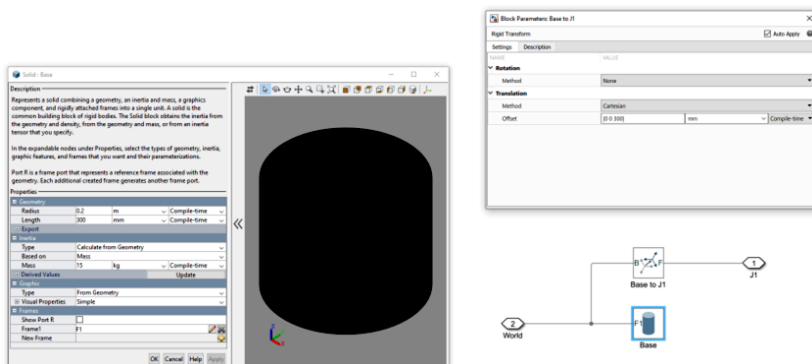


Рис. 5. Опора робота

Каждое звено и шарнир могут быть настроены под конкретного робота. Ключевое значение имеет блок Rigid Transform – он позволяет правильно соединить детали между собой путем задания локальной системы координат. Рекомендуется устанавливать на 3D-моделях локальную систему координат F1 в точке крепления и направлять вращения вдоль оси z, взяв за основу модель ДН.

За блоком с моделью робота следует блок Forward Kinematics, в составе которого находится функция Get Transform (рис. 6), которая позволяет получить на выходе из системы снова координаты фланца после моделирования робота.

Последней подсистемой остается расчет динамики робота, который содержит в себе блок Joint Space и блок Inverse Dynamics (рис. 7). Данные блоки напрямую не воздействуют на робота, а лишь делают расчеты по заданным параметрам: перемещения и скорости, снятые с подсистемы модели робота.

обратная функция инверсивной кинематики, позволяет вывести данные о положении в виде 4x4 матрицы, которая в последствие трансформируется в вектор

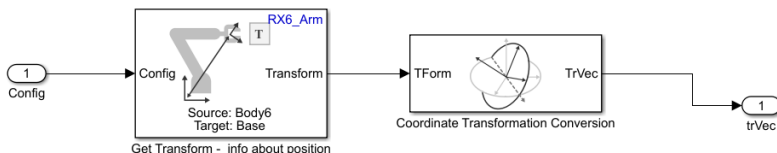


Рис. 6. Функция Get Transform

обратная функция инверсивной кинематики, позволяет вывести данные о положении в виде 4x4 матрицы, которая в последствие трансформируется в вектор

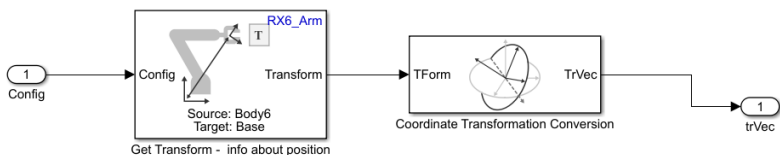


Рис. 7. Блок Inverse Dynamics

Если запустить симуляцию на основе данных с блока Inverse Dynamics (используя крутящий момент), то модель начнет совершать непредсказуемые действия – двигаться ломано и хаотично. В таком случае получить корректные данные невозможно, так как модель подвержена физическим явлениям и силы инерции сильно искажают данные. Поэтому блок динамики можно использовать только отдельно от модели робота, например, выводить данные в файл или на график.

Задание параметров пользователем для используемых функций. Основные параметры задаются в подсистеме «Траектория». На нашем примере была задана пиковая скорость в 0,5 м/с и ускорение 10 м/с². Также используется ввод углов поворота фланца в виде углов Эйлера относительно координат x , y , z . И для получения необходимых данных для расчета силовой установки для робота-манипулятора используем примитивную модель редуктора. Разберем более подробно на примере. На рис. 8 изображен примитивный редуктор, использующийся в работе. Входные порты 1–6 соответствуют крутящим моментам на шарнирах. Заданные константы $K0-5$ и $n0-5$ – это коэффициенты редукторов, установленных на манипуляторе. Таким образом, задавая разные коэффициенты редуктора, можем получить необходимые данные на выходе.

Отображение и вывод результатов. Вывод данных осуществляется тремя путями: файл, график, симуляция. В первом случае, подключив блок вывода в файл, получим таблицу Excel с данными в единицу времени. Также можно вывести необходимые данные в график (рис. 9).

Здесь мы можем увидеть следующие данные:

- заданная траектория движения – с помощью подсистемы «Траектория»;
- выполненная траектория – та траектория, по которой двигался манипулятор;

– крутящий момент на редукторе – уже преобразованные с помощью редуктора моменты, заставляющие вращаться шарнир.

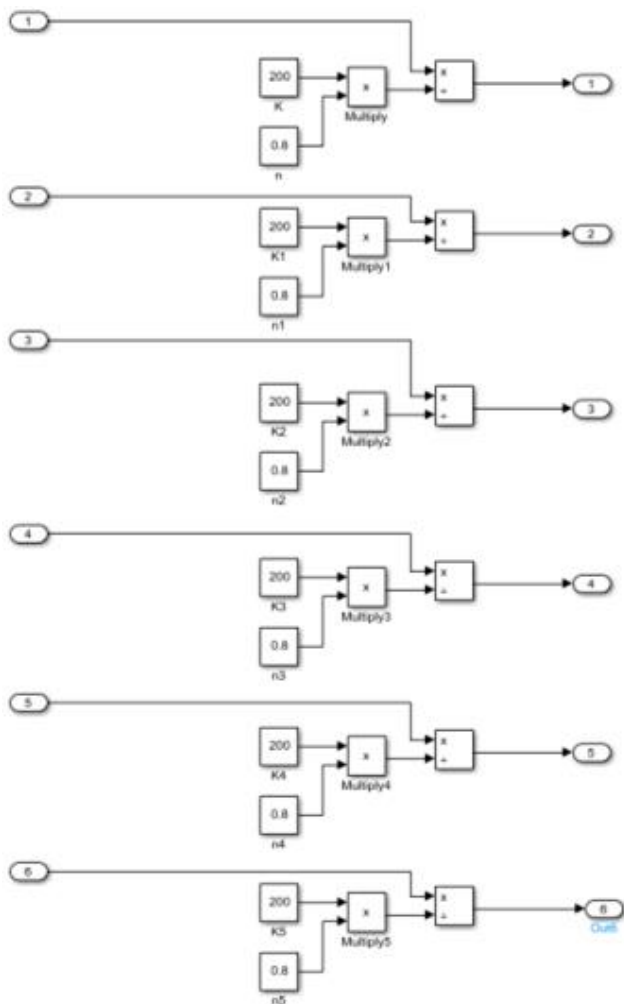


Рис. 8. Пример

Крутящий момент на двигателе – используя известные коэффициенты редуктора, можно получить необходимые для этого движения моменты на двигателе.

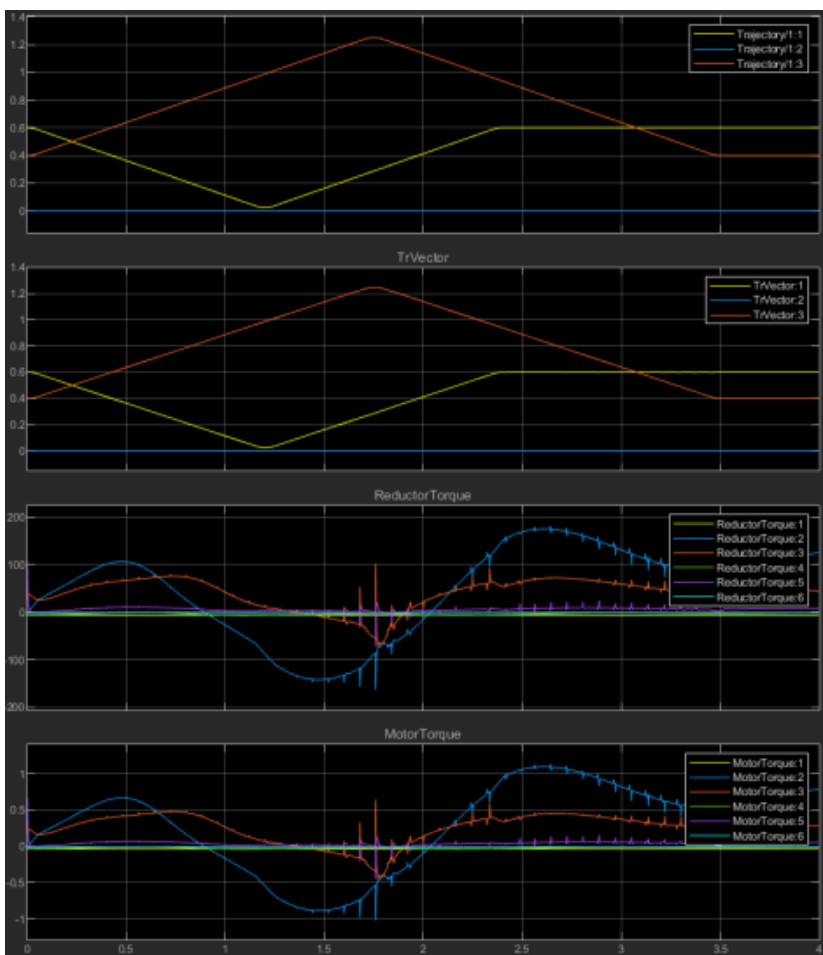


Рис. 9. График данных

При надобности можно снимать и другие показания, такие как скорость, ускорения, углы поворота и др. Резкие всплески и падения происходят из-за ошибок в вычислениях. Чтобы избежать большого числа всплесков, необходимо в блоке Solver Configurations и в блоке Inverse Kinematics увеличить точность вычислений. Но при этом увеличится и время рендера в переводе на секунду времени. На рис. 10 изображена примитивная модель робота в среде Simulink.

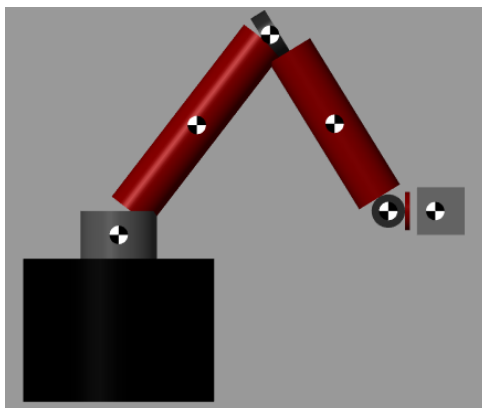


Рис. 10. Модель робота в среде Simulink

При изменении траектории движения модель двигается в соответствии с новыми заданными параметрами. Также можно выбрать отображения центров масс, локальных координат. И можно выбрать удобный угол обзора или свободно вращать модель в пространстве.

Заключение. В итоге с помощью набора инструментов MatLab была создана модель робота. И при необходимости в данную программу можно подставить другую готовую модель и получить данные. Рассмотрим получившиеся результаты, представленные в табл. 1 и 2. В них были использованы максимальные значения крутящих моментов от начала следования по траектории до ее окончания.

Таблица 2

Получившиеся результаты

№ оси	Без полезной нагрузки		С полезной нагрузкой 6 кг		Угол начала движе- ния оси	Угол конец движе- ния оси	Примечание
	момент на двига- теле, Н·м	момент на редукто- ре, Н·м	момент на двига- теле, Н·м	момент на редукторе, Н·м			
1	0,9	144	1,5	240	0	90	В данном случае робот совершает вращение вдоль оси z, находясь в вытянутом положении
2	0,8	128	1,2	192	40	40	
3	0,6	96	0,9	144	170	170	
4	0,04	6,4	0,05	8	0	90	
5	0,02	3,2	0,02	3,2	0	-90	
6	0,02	3,2	0,02	3,2	0	0	

Таблица 3

Получившиеся результаты

№ оси	Без полезной нагрузки		С полезной нагрузкой 6 кг		Угол начала движения оси	Угол конец движения оси	Примечание
	момент на двигателе, Н·м	момент на Редукторе, Н·м	момент на двигателе, Н·м	момент на редукторе, Н·м			
1	0,01	1,6	0,01	1,6	0	90	В данном случае робот поднимает груз в плоскости zx из максимально вытянутого положения вдоль оси x в максимально вытянутое вдоль оси z
2	0,95	152	1,45	232	0	–90	
3	0,6	96	0,9	144	170	170	
4	0,04	6,4	0,05	8	0	0	
5	0,03	4,8	0,04	6,4	0	–90	
6	0,02	3,2	0,02	3,2	0	0	

Таким образом можно сделать вывод о том, что необходим использовать двигатели с крутящим моментом не более 2 Н·м в случае с 1-м и 2-м шарнирами, на 3-м шарнире установить двигатель с крутящим моментом 1 Н·м. На остальных соединениях не более 0,1 Н·м.

Сведения об авторах

Колобов Борис Константинович – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: kolobovboris1@gmail.com

Карлов Денис Анатольевич – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: denis.karlov.78@yandex.ru

С.И. Перваков, П.А. Булатов, А.Л. Погудин

Пермский государственный национальный
исследовательский университет, Пермь

**РАЗРАБОТКА ЛАБОРАТОРНОГО СТЕНДА
«БЕСПИЛОТНЫЙ ЛЕТАТЕЛЬНЫЙ АППАРАТ».
3D-ПРОЕКТИРОВАНИЕ И МОДЕЛИРОВАНИЕ**

В статье рассмотрены технологии, используемые в процессе разработки лабораторного стенда, проанализированы существующие решения БПЛА и построена модель системы в программе 3D-моделирования, разработан прототип устройства.

Ключевые слова: робототехника, 3D-моделирование, 3D-печать, БПЛА, квадрокоптер.

S.I. Pervakov, P.A. Bulatov, A.L. Pogudin

Perm State National Research University, Perm

**DEVELOPMENT OF THE LABORATORY STAND "UNMANNED
AERIAL VEHICLE". 3D-DESIGN AND MODELING**

The article discusses the technologies used in the process of developing a laboratory stand analyzes existing UAV solutions, and builds a model of the system in a 3D modeling program, and develops a prototype device.

Keywords: robotics, 3D modeling, 3D printing, UAV, quadcopter.

На данный момент беспилотные летательные аппараты (БПЛА) – бурно развивающаяся область техники. БПЛА получили широкое применение в различных сферах, их функциональные возможности продолжают расширяться, они определенно являются одной из самых важных технологических тенденций нашего времени. В настоящее время они применяются в военной, гражданской, промышленной и научной сферах, а разработки ведутся как коммерческими предприятиями, так и любителями.

Актуальность дипломного проекта обусловлена потребностью в доступных и настраиваемых лабораторных стендах по исследованию БПЛА, созданных с минимальными возможными затратами, а также необходимостью в создании тренажеров для программного моделирования и проектирования.

Объектом исследования являются беспилотные летательные аппараты. Предметом исследования являются инструменты для проектирования БПЛА.

Целью работы является разработка и реализация конструкции БПЛА, позволяющей управлять его полетом и выполнять различные задачи. Для достижения поставленной цели были выделены следующие задачи:

1. Разработка варианта конструкции БПЛА.
2. Расчет минимальной мощности двигателей и времени полета.

Анализ существующих видов БПЛА. Существует такая классификация, как принцип выполнения полета. Согласно ей все существующие летательные аппараты, как пилотируемые, так и беспилотные, подразделяются на четыре основные категории: аэростатические – аэростаты, дирижабли и т.д.; аэродинамические – вертолеты, самолеты и т.п.; баллистические – спутники Земли, космические станции; ракетодинамические – ракеты [1].

В ходе анализа был выбран тип БПЛА квадрокоптер – мультикоптер с 4 роторами. Обосновано это возможностью организовать довольно простую схему организации движения, простотой в управлении, безопасностью, маневренностью, небольшим размером и низкими затратами на создание. Хотя есть и недостатки, такие как большая уязвимость к повреждениям, ограниченное время полета из-за ограниченной емкости аккумуляторов [2].

Выбор рамы и зависящих от нее компонентов. Раму для квадрокоптера можно разделить на 2 составляющие: тело и лучи [3].

Зачастую тело состоит из нижней и верхней пластины, которые скрепляются вертикальными стойками на винтах, и некоторых промежуточных маленьких пластинок. В тело рамы помещаются электронные составляющие, такие как полетный контроллер, камера, плата разводки питания и т.д. На лучах же размещаются регуляторы оборотов и двигатели [3].

Существуют такие типы рам и лучей, как:

- «Н»;
- «Х»;
- гибрид «Х»;
- растянутый «Х»;
- квадрат;
- «Plus» [3].

В ходе анализа был выбран тип рамы «гибрид X». Это обосновывается более высокой стабильностью и контролируемостью во время полета благодаря распределению веса и поддерживающей конструкции, улучшенной проходимости за счет расположения роторов по диагонали, большой поворотной частотой, уменьшенным воздействием вибраций на оборудование (что повышает надежность), эстетическим дизайном, объемной кабиной, позволяющей использовать большее количество оборудования. Размером рамы называется расстояние (по диагонали) в миллиметрах между одним двигателем и другим [3]. Размер рамы влияет на такие параметры, как максимальный размер винтов, сопротивление воздуха, момент инерции, общий вес [3]. Ввиду своей популярности было принято решение выбрать размер рамы, равный 220 мм. Под данный размер подходят пропеллеры 5 и 6 дюймов. Исходя из экономических соображений, был выбран размер 5 дюймов. Наиболее оптимальным шагом для пропеллера будет шаг 4,5. Он сочетает в себе хорошую чувствительность, крутящий момент, максимальную скорость и тягу. По такому же принципу – по принципу оптимальности – было выбрано количество лопастей пропеллера, равное 3. Таким образом маркировка пропеллеров – 5045.

Моделирование и 3D-печать. В ходе анализа сред моделирования было принято решение использовать программный продукт Blender. Детали корпуса приведены на рис. 1.

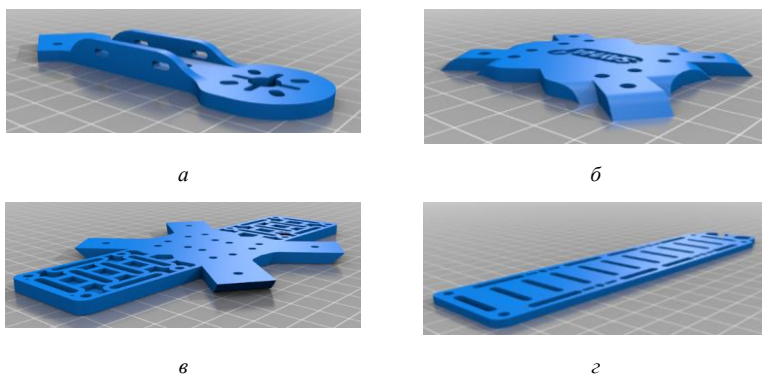


Рис. 1. Внешний вид 3D-деталей: *а* – луч; *б* – фиксирующая пластина; *в* – нижняя часть фюзеляжа; *г* – верхняя часть фюзеляжа

Материалом деталей был выбран карбон ввиду своей высокой прочности, жесткости, а следовательно, большей надежности, и долговечности, и малой массы.

Расчет необходимой минимальной тяги моторов квадрокоптера. Был рассчитан примерный вес конструкции (таблица).

Расчет общего веса конструкции

Параметр	Значение массы, кг
Масса рамы	0,13
Масса комплектующих	0,6
Полезная нагрузка	0,5
Полная масса коптера	1,23

Итоговый вес беспилотника примерно 1,23 кг, а это значит, что минимальная тяга от 4 моторов, нужная, чтобы оторвать коптер от земли, равна

$$T = G = mg = 1,2 \text{ кг} \cdot 9,81 \text{ Н/кг} = 12,06 \text{ Н.}$$

Зная формулу вычисления количества оборотов, а также значения параметров, такие как $\lambda = 0,75$ – коэффициент скорости, $\alpha = 1,1$ – коэффициент тяги (взяты из методического пособия), можно вычислить минимальное необходимое количество оборотов:

$$n = \sqrt{\frac{G}{z\rho D^4 \left(\alpha(\lambda) - \frac{C_{yS}}{2zD^2} \lambda^2 \right)}} = \sqrt{\frac{12,06}{3 \cdot 1,2 \cdot 0,127^4 \left(1,1 - \frac{1,16 \cdot 0,013}{2 \cdot 3 \cdot 0,127^2} 0,75^2 \right)}} = 113 \frac{\text{об}}{\text{с}} = 6780 \text{ об/мин.}$$

Теперь можно рассчитать минимальную мощность двигателя, зная коэффициент мощности $\beta = 0,5$:

$$P = \beta \rho n^3 D^5 = 0,5 \cdot 1,2 \text{ кг/м}^3 \cdot 113^3 \cdot 0,127^5 = 28,6 \text{ Вт.}$$

Расчет общего времени полета квадрокоптера. В ходе дальнейшей работы был выбран аккумулятор с параметрами: ток разряда $I = 1,3 \text{ А} \cdot \text{ч}$, напряжение $U = 14,8 \text{ В}$.

Теперь можем вычислить энергию, которую может хранить аккумулятор:

$$W_A = U \cdot I = 14,8 \text{ В} \cdot 1,3 \text{ А} \cdot \text{ч} = 19,24 \text{ Вт} \cdot \text{ч.}$$

Зная, что на схему подается 1 Вт, а на моторы в сумме $28,6 \cdot 4 = 114,4 \text{ Вт}$, можем рассчитать примерное время полета:

$$t = \frac{W_A}{W_{ДВ} + W_{СХ}} = \frac{19,24}{114,4 + 1} = 0,17 \text{ ч} \approx 10 \text{ мин.}$$

На рис. 2 предложена 3D-модель варианта лабораторного стенда, созданная в среде Blender.

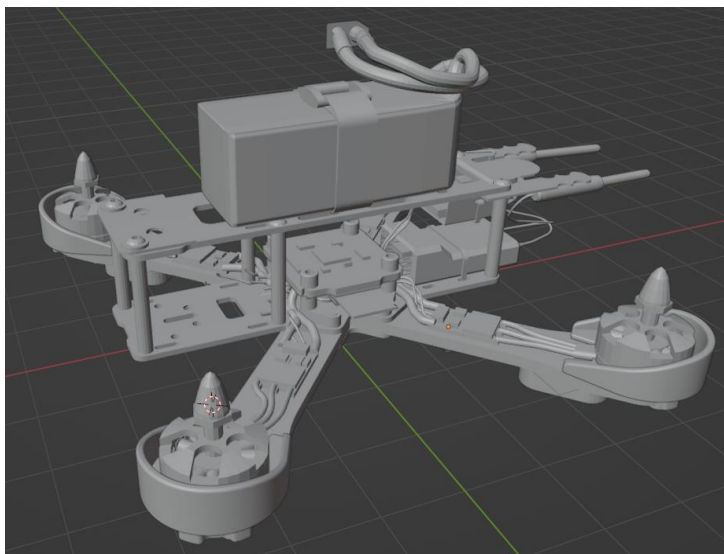


Рис. 2. Модель квадрокоптера в среде Blender

Заключение. В статье был проведен анализ существующих типов БПЛА, был выбран тип (гибрид «Х») и размер рамы (220 мм), подобраны подходящие пропеллеры (5045). Обоснован выбор среды моделирования (Blender) и материала (карбон), из которого сделаны элементы конструкции рамы, приведены изображения этих элементов. Приведены расчеты минимальной мощности двигателей (28,6 Вт), необходимой для того, чтобы квадрокоптер смог подняться в воздух, рассчитано общее время полета (≈ 10 мин).

Библиографический список

1. Классификация беспилотных авиационных комплексов военного назначения в контексте образовательного процесса военного вуза – URL: https://ores.su/media/filer_public/37/6d/376ddd25-ee88-4ed5-b02b-f90f387e4b82/7klassifikatsiia_bespilotnykh_aviatsionnykh_kompleksoy_voenogo_naznacheniiia.pdf
2. Анализ типов дронов, их преимуществ и недостатков – RCDrone. – URL: <https://rcdrone.top/ru/blogs/articles/2024-анализ-типов-дронов-их-преимуществ-и-недостатков>

3. Рама для квадрокоптера – как выбрать, основы и советы. Все о квадрокоптерах. – URL: <https://profpv.ru/kak-vybrat-ramu-dlya-kvadrokoptera-os/>

Сведения об авторах

Перваков Святогор Игоревич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: perwakov.swyatorgor@yandex.ru

Булатов Павел Александрович – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: pashka.bulatow@gmail.com

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

П.А. Булатов, С.И. Перваков, А.Л. Погудин

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ЛАБОРАТОРНОГО СТЕНДА «БЕСПИЛОТНЫЙ ЛЕТАТЕЛЬНЫЙ АППАРАТ». СХЕМОТЕХНИЧЕСКОЕ ПРОЕКТИРОВАНИЕ И МОДЕЛИРОВАНИЕ

В статье рассмотрены технологии, используемые в процессе разработки лабораторного стенда «Беспилотный летательный аппарат» проанализированы существующие решения построения БПЛА, предложена модель системы в программе схемотехнического проектирования, разработан прототип устройства.

Ключевые слова: робототехника, БПЛА, полетный контроллер, квадрокоптер, схемотехническое моделирование, исследование БПЛА.

P.A. Bulatov, S.I. Pervakov, A.L. Pogudin

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF LABORATORY STAND "UNMANNED AERIAL VEHICLE". SCHEMATIC DESIGN AND MODELING

The article examines the technologies used in the development process of the "Unmanned Aerial Vehicle" laboratory stand. It analyzes existing UAV design solutions, proposes a system model in circuit design software, and develops a device prototype.

Keywords: robotics, UAV, flight controller, quadcopter, schematic modeling.

На сегодняшний день беспилотные летательные аппараты (БПЛА), в частности квадрокоптеры, играют ключевую роль в решении многих технологических задач, таких как, мониторинг, инспекция, спасательные операции и т.д., предоставляя уникальные возможности в ситуациях, где человеку находиться опасно [1].

Особенно важным становится развитие средств исследования и тестирования для обеспечения эффективности и безопасности эксплуатации таких систем.

Анализ существующих источников свидетельствует о необходимости системного подхода к разработке и применению беспилотных летательных аппаратов в различных областях, а также позволяет сделать вывод об отсутствии такого подхода на сегодняшний момент.

Разработка лабораторного стенда становится актуальной задачей, решение которой позволит проводить исследования и тестирования новых технологий, позволит обеспечить платформу для разработки и оптимизации программного обеспечения и аппаратных решений для управления и внедрения БПЛА, а также создать обучающие тренажеры с минимальными затратами.

Объектом исследования является лабораторный стенд по исследованию БПЛА на базе квадрокоптера.

Предметом исследования является процесс разработки и конструирования схмотехники лабораторного стенда на базе квадрокоптера.

Цель проекта – разработать лабораторный стенд для исследования беспилотных летательных аппаратов (БПЛА). Чтобы достичь поставленной цели, требуется решить следующие задачи:

- 1) разработать программно-аппаратный комплекс стенда;
- 2) провести моделирование программно-аппаратного комплекса.

Анализ БПЛА. Рассмотрена классификация БПЛА по типу построения, которая включает в себя мультикоптер, вертолетный тип, планер, конвертоплан. Мультикоптерные БПЛА построены на базе трех и более моторов, способны зависать на одной высоте, а также осуществлять вертикальные взлеты и посадки, что делает их наиболее универсальными для большинства технологических задач.

Проведен сравнительный анализ типов построения и в виду преимуществ мультикоптерных БПЛА именно на их базе решено было реализовать стенд.

Произведен анализ классификации мультикоптерных БПЛА, и сделаны следующие выводы, что среди всех типов мультироторных аппаратов наибольшее распространение получили квадрокоптеры, ввиду своей дешевизны, простоты, маневренности, устойчивости и стабильности в полете.

Резюмируя все вышесказанное, для реализации лабораторного стенда выбраны следующие критерии и составляющие БПЛА:

- принцип построения – мультикоптер;
- количество моторов – 4 бесколлекторных мотора;
- расположение моторов – Х коптер.

Разработка структурной схемы. На рис. 1 представлена вариант структурной схемы квадрокоптера.



Рис. 1. Структурная схема квадрокоптера

Проведя анализ структурной схемы квадрокоптера были сделаны выводы о том, что схема демонстрирует модульный подход к конструкции БПЛА, что позволяет гибко настраивать его для различных задач и условий эксплуатации. Каждый блок представляет собой независимую систему, которая может быть легко заменена или обновлена, что способствует удобству замены блоков и их исследования и тестирования. Подбор этих блоков стал задачей схемотехнического проектирования.

Разработка программного и схемотехнического обеспечения. Проанализировав и сравнив инструменты для работы с 3D-моделями выбрана программа Blender. Средой для схемотехнического моделирования выбрана программа Fritzing [2].

Проведя анализ систем управления (микроконтроллеров), для реализации лабораторного стенда был выбран контроллер SpeedyBee F405 на базе MCU STM32F405 с поддержкой BlueTooth и прошивкой Beta-Flight [3]. Для удобства использования и универсальности стенда был выбран размер рамы 220 мм и размер пропеллера 5 дюймов, под эти параметры подошли моторы с KV 2300 rpm/V и размером статора 2205. Чтобы сэкономить пространство на раме, было принято решение использовать STACK, который позволяет объединить ESC регуляторы скорости и PDB плату разводки в один компонент. В конечном итоге выбор пал на SpeedyBee 50A ESC.

Также, чтобы обеспечить необходимую мощность и автономность для будущих испытаний, был выбран 4S аккумулятор на 1300 мАч, токоотдачей 90С и напряжением 14,8 В.

Исходя из требуемых характеристик передатчика сигнала, выбор остановился на радиопередатчике (ресивере) FS-iA6B и пульте управления FS-i6, который позволит не только управлять квадрокоптером на стенде, но и обучиться методике полета в FPV симуляторе.

Средой программирования, управления и регулирования характеристик дрона на стенде была выбрана программа BetaFlight.

Детально изучив рынок FPV симуляторов и проанализировав их преимущества и недостатки, была выбрана программа Liftoff, которая позволяет тонко настраивать параметры дрона (такие как PID-регуляторы, вес, размер пропеллеров).

На рис. 2 приведена монтажная схема квадрокоптера, на котором базируется исследовательский стенд. Схема создана в программе Fritzing на базе собственного пака компонентов, который в дальнейшем будет использоваться для проведения лабораторных работ.

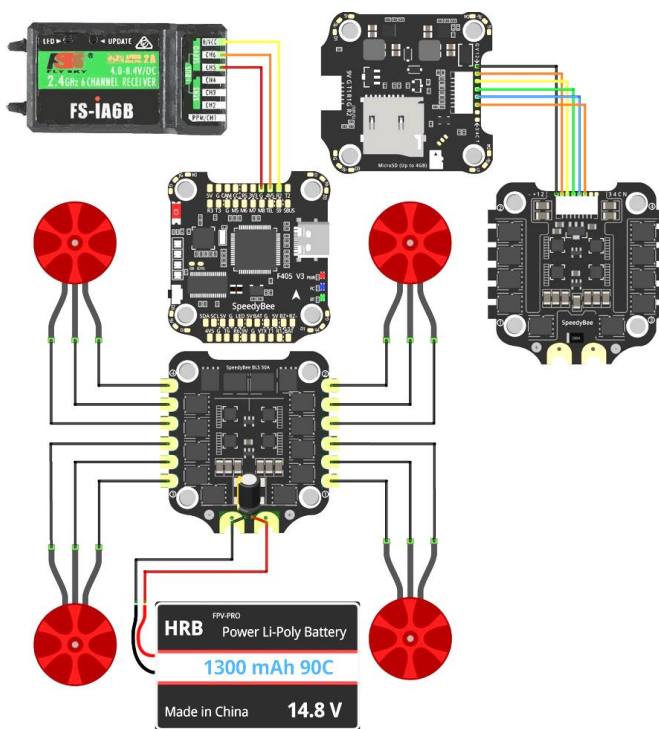


Рис. 2. Монтажная схема квадрокоптера

В исследованиях разработан и описан процесс создания лабораторного стенда по исследованию БПЛА на базе квадрокоптера с пошаговой инструкцией по сборке, настройке и использованию стенда.

Описан процесс 3D-печати компонентов рамы и ее сборки, описан процесс пайки и подключения электронных компонентов системы. Показан процесс прошивки полетного контроллера, прошивки регуляторов скорости, привязки передатчика к пульту управления, а также настройки каналов управления и выбора режима работы передатчика.

Приведен необходимый код для входа в режим геймпада для подключения к FPV – симулятору, а также выхода из него без удаления драйверов устройства последовательного ввода-вывода.

Заключение. В результате исследования и решения задач, удалось достигнуть поставленную цель, был разработан программно-аппаратный комплекс стенда по исследованию БПЛА, проведено моделирование комплекса.



Рис. 3. Лабораторный стенд для исследования беспилотных летательных аппаратов

Комплекс позволит решать актуальные задачи по оптимизации программного обеспечения и аппаратных решений для управления и внедрения БПЛА, также комплекс станет основой для обучающих тренажеров. Получившийся стенд представлен на рис. 3.

Библиографический список

1. Devalla V., Prakash O. Developments in unmanned powered parachute aerial vehicle: A review // Aerospace and Electronic Systems Magazine. – 2014.
2. Юревич Е.И. Основы робототехники: учеб. пособие. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2010. – 368 с.
3. Бобылев Ф.А., Маркелов М.К. Пример создания полетного контроллера беспилотного летательного аппарата на базе микроконтроллера STM32 // Вестник Пензенского государственного университета. – 2023. – № 3 (43). – С. 72–81.

Сведения об авторах

Булатов Павел Александрович – бакалавр Пермского национального исследовательского политехнического университета, г. Пермь, гр. ПРТ-20-16, e-mail: pashka.bulatow@gmail.com

Перваков Святогор Игоревич – бакалавр Пермского национального исследовательского политехнического университета, г. Пермь, гр. ПРТ-20-16, e-mail: perwakov.swyatogor@yandex.ru

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

В.О. Кушев, В.С. Бродников

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ РЕГИСТРАЦИИ ПРОДАЖ В СЕТИ КОФЕЕН, АВТОМАТИЗИРУЮЩЕГО РЕАЛИЗАЦИЮ СКИДОК/АКЦИЙ/ПРОГРАММ ЛОЯЛЬНОСТИ

Современный рынок услуг и товаров подчеркивает важность удовлетворенности клиентов для успешности бизнеса, включая сферу кофейных заведений. Компании используют маркетинговые инструменты, такие как акции, скидки и программы лояльности, для удержания клиентов и увеличения прибыли. Однако создание этих инструментов может столкнуться с техническими проблемами, требующими разработки эффективных решений. Целью данной работы является разработка приложения для регистрации продаж и выявления наиболее востребованных скидок и акций в сети кофеен. Результаты анонимного анкетирования позволяют определить предпочтения клиентов по использованию программ лояльности и скидок в кофейнях. В итоге предлагается система с автоматической генерацией акций и программа лояльности, работающая как по системе суммы, так и по количеству заказов.

Ключевые слова: маркетинговые инструменты, скидки, программа лояльности, информационная система, автоматизация.

V.O. Kushev, V.S. Brodnikov

Perm National Research Polytechnic University, Perm

RECORDING SALES IN A COFFEE SHOP NETWORK, AUTOMATING THE PROCESS OF IMPLEMENTING DISCOUNTS/PROMOTIONS/LOYALTY PROGRAM APP DEVELOPMENT

The modern market of goods and services emphasizes the importance of customer satisfaction for business success, including in the coffee industry. Companies utilize marketing tools such as promotions, discounts, and loyalty programs to retain customers and increase profits. However, creating these tools may encounter technical issues requiring the development of effective solutions. The aim of this work is to develop an application for sales registration and identification of the most demanded discounts and promotions in coffee shop networks. The results of anonymous surveys help determine customer preferences for loyalty programs and discounts in coffee shops. Ultimately, a system with automatic promotion generation and a loyalty program operating based on both total spending and the number of orders is proposed.

Keywords: marketing tools, discounts, loyalty program, information system, automation.

В условиях современного высококонкурентного рынка услуг и товаров уровень удовлетворенности клиентов является ключевым критерием успешности любого бизнеса, в том числе и кофейного. Поэтому для увеличения прибыли компании стараются не только привлекать новых клиентов, но и сохранять своих постоянных потребителей. Маркетинговыми инструментами, позволяющие удерживать существующих клиентов, являются акции, скидки и программы лояльности.

При создании акции, скидок и программ лояльности могут возникнуть проблемы, на основе чего их создавать. Решением проблемы будет создание правил для формирования их.

При использовании приложения для регистрации продаж в сети кофеен могут возникнуть технические проблемы, проблемы с безопасностью, проблемы совместимости, проблемы с обновлением приложения, ошибки в программном обеспечении.

Эффективная автоматизация регистрации продаж в сети кофеен и автоматическое выявление скидок/акций/программ лояльности улучшило бы генерацию цен на производимые продукты, тем самым увеличив продажи.

Цель работы состоит в проектировании, документировании и разработке приложения для регистрации продаж в сети кофеен и выявление наиболее востребованных на текущий момент скидок/акций/программ лояльности.

Объектом исследования является регистрации продаж в сети кофеен и формирование скидок/акций/программ лояльности. Предметом исследования является автоматизация регистрации продаж в сети кофеен и формирование скидок/акций/программ лояльности.

В работе использовались системный подход, метод анализа, сравнения. Для моделирования системы используется объектно-ориентированный подход.

С помощью Google Forms был проведено анонимное анкетирование [1] среди одноклассников, участников сообщества «Подслушано ПГНИУ» [2] и участников сообщества «Опросы и голосования на любые темы!» [3]. Целью опроса было выяснить требования к создаваемой системе.

Из 31 участника анкетирования только 9,7 % (3 человека) не пользуются различными программами лояльности и 19,4 % (6 человек) не используют скидки, акции или бонусы в кофейне. Предпочтительные скидки в кофейнях включают «5-й стакан в подарок» (25,8 %, 8 человек)

и оплату заказов бонусами (1 балл = 1 рубль) (77,4 %, 24 человека). 51,6 % (16 человек) предпочли бы кофейню с программой лояльности, если бы имели выбор между двумя одинаковыми кофейнями, а 48,4 % (15 человек) ответили, что им без разницы. Оплачивать заказы бонусами хотели бы 96,8% (30 человек). Среди респондентов 29 % (9 человек) были мужчины и 71 % (22 человека) женщины в возрасте, варьирующемся от 19 до 49 лет. Благодаря опросу, стало очевидно, что люди знают о программах лояльности, пользуются скидками, акциями и бонусами, а также что в систему необходимо включить скидки из вопроса о предпочтительных скидках и возможность оплачивать бонусами покупку.

Проанализировав систему лояльности у Coffee Like [4], было выявлено, что программа лояльности должна иметь бонусную уровневую систему (4 или 5 уровней). Каждый балл – это рубль, потраченный за категорию «Напитки». Ежемесячная сумма заказов влияет на передвижение по уровням: сумма больше, чем за прошлый месяц поощряется, наоборот – нет.

Были проанализированы системы автоматизации ресторанного бизнеса: Poster POS [5], R-Keeper [6], имеющие возможность добавления акции (конструктор акций). Пользователю надо заполнить как минимум 12 полей формы. Надо ввести такие данные, как название акции, дата начала акции, дата окончания, условия акции, категорию товара, условия срабатывания акции, условия накопления, дни работы акции, время работы акции, участники акции, результат акции, процент скидки. В зависимости от человека заполнение такой формы может занимать неопределенное время. Пользователь сам выбирает товар для скидки. Чем руководствуется человек при выборе? Нужна логика выбора. Предпочтительнее делать скидки на те товары, у которых кончается срок годности (например, молоко). Или те товары, которые упали по продажам по сравнению с предыдущей неделей, месяцем. Для этого в разрабатываемой информационной системе должна быть хранимая в базе данных информация об ингредиентах, их срок годности, о технологических картах.

В разрабатываемой системе пользователю предоставляется возможность выбора: создать акцию автоматически или вручную (рис. 1). В первом случае система предлагает, например, напиток, у которого самые низкие продажи за месяц и у которого в состав входит ингредиент, срок годности которого подходит к концу. Во втором случае пользователь может выбрать напиток самостоятельно, в том числе используя подсказки системы.

ВРУЧНУЮ

АВТОМАТИЧЕСКИ

Способ выбора

☐ Самостоятельный

☐ Скоропортящиеся продукты

☐ Низкие продажи

☒ Скоропортящийся товары с низкой продажей

Срок продажи

☒ су

Мокко (-5 Продажи)
5 суток МОЛОКО

Латте (-4 Продажи)
5 суток МОЛОКО

Капучино (-4 Продажи)
5 суток МОЛОКО

Латтечино (-3 Продажи)
5 суток МОЛОКО

Крем-кофе (-2 Продажи)
3 дня

год

Рис. 2. Интерфейс «Конструктор акций»

СУММА ЗАКАЗОВ

КОЛИЧЕСТВО ЗАКАЗОВ

Выберите напиток:

Все

Уровень 1

=

1000 мл

Уровень 2

=

2000 мл

ДОБАВИТЬ УРОВЕНЬ

Рис. 2. Интерфейс «Программа лояльности»

В реализуемой информационной системе также предусмотрена возможность управления программой лояльности. Программа по отношению к клиенту определяется по сумме заказов или по количеству

заказов. Можно выбрать продукт, на который будет распространяться программа (рис. 2). Проанализированные системы механизмом управления программой лояльности не обладают. В итоге в реализуемой информационной системе должно быть следующее:

- автоматическая генерация акции;
- приоритет тех товаров в акции, в состав которых входит скоропортящиеся продукты, плохо продаваемые товары;
- программа лояльности работает или по сумме заказов, или по количеству заказов.

Библиографический список

1. Опрос о кофейнях [Электронный ресурс]. – URL: <https://forms.gle/SPkeYaX97Dmc3arYA> (дата обращения: 30.04.2024).
2. Подслушано ПГНИУ [Электронный ресурс]. – URL: https://vk.com/overhear_pgniu (дата обращения: 30.04.2024).
3. Опросы и голосования на любые темы! [Электронный ресурс]. – URL: <https://vk.com/oprosy> (дата обращения: 30.04.2024).
4. Как работает бонусная программа кофе лайк [Электронный ресурс]. – URL: <https://telegra.ph/Kak-rabotaet-bonusnaya-programma-kofe-lajk-03-04> (дата обращения: 30.04.2024).
5. Как настроить популярные акции в системе учета Poster [Электронный ресурс]. – URL: https://vk.com/video-64851049_456239040 (дата обращения: 30.04.2024).
6. R-Keeper, создание скидок. Автоматические, ручные, на категорию, по времени [Электронный ресурс]. – URL: <https://ugitservice.com/baza-znaniy/instruktsii/188-r-keeper-sozdanie-skidok-avtomaticheskie-ruchnye-na-kategoriyu-po-vremeni> (дата обращения: 30.04.2024).

Сведения об авторах

Кушев Вадим Олегович – кандидат педагогических наук, доцент кафедры «Информационные технологии» Пермского государственного национального исследовательского университета, г. Пермь, e-mail: va-dimk@psu.ru

Бродников Вадим Сергеевич – магистрант Пермского государственного национального исследовательского университета, гр. ФИТ-1,2-2022 НБ, г. Пермь, e-mail: vadimbrodnikov@mail.ru

Г.Ф. Масич, М.Г. Батюков

Пермский национальный исследовательский
политехнический университет, Пермь

УСТАНОВКА И АДМИНИСТРИРОВАНИЕ ОПЕРАЦИОННОЙ СИСТЕМЫ LINUX ПО СЕТИ

В статье исследуется возможность установки и администрирования операционной системы GNU/Linux по сети передачи данных в условиях вуза. Для моделирования процесса используются средства виртуализации (виртуальные машины). Установка операционных систем выполняется с помощью технологии PXE. Администрирование операционных систем GNU/Linux выполняется с помощью системы управления конфигурациями Ansible.

Ключевые слова: ansible, linux, PXE.

G.F. Masich, M.G. Batyukov

Perm National Research Polytechnic University, Perm

INSTALLATION AND ADMINISTRATION OF THE LINUX OPERATING SYSTEM OVER THE NETWORK

The article examines the possibility of installing and administering the GNU/Linux operating system over a data network in a university setting. To simulate the process, virtualization tools (virtual machines) are used. Operating systems are installed using PXE technology. Administration of GNU/Linux operating systems is performed using the Ansible configuration management system.

Keywords: ansible, linux, PXE.

Неотъемлемой частью современного технического вуза является большое количество автоматизированных рабочих мест (АРМ). В их состав входит операционная система (ОС) и приложения, установка и администрирование которых – трудоемкий процесс. Поэтому актуальна задача снижения затрат учебно-вспомогательного персонала посредством сетевых технологий. В представленной статье показана используемая совокупность технологий и разработанная авторами схема их использования для решения задачи централизованного управления подчиненными АРМ.

Предложено для установки ОС воспользоваться технологией PXE (англ. Preboot eXecution Environment) [1], которая будет выполнять за-

грузку по сети образа операционной системы. Образы ОС в формате ISO разместить в загрузочном PXE сервере, который установить в локальной сети АРМ/компьютеров. А для задач администрирования использовать систему управления конфигурациями Ansible.

Инсталляция технологии PXE. В качестве PXE сервера целесообразно использовать бесплатную программу iVentoу [2], которая поддерживает более 110 распространенных типов ОС (Windows / WinPE / Linux / VMware). После загрузки с официального сайта [2] и запуска программы iVentoу откроется браузер с интерфейсом сервера (рис. 1). Для конфигурации сервера необходимо указать интерфейс, на котором будет запущен сервер, его IP адрес, маску подсети, шлюз, диапазон выдачи IP адресов для DHCP, DNS сервер. Для запуска сервера необходимо нажать кнопку «Start».

Boot Information

IP Configuration

Select Server IP

192.168.0.200

NIC Name

enp0s3

Subnet Mask

255.255.255.0

Gateway

192.168.0.10

IP Pool (begin)

192.168.0.201

IP Pool (end)

192.168.0.220

DNS Server

192.168.0.10

RUNNING

Device List

IP Address	MAC Address		BIOS Mode	Boot File	Status	Operate
192.168.0.201	08-00-27-88-62-34		BIOS	kubuntu-23.10-desktop-amd64.iso	NBD connect IMG1 OK	Details
192.168.0.202	08-00-27-a9-58-fb		BIOS		HTTP start 200 [ipxe /01-08-00-27-a9-58-fb]	Details

Рис. 1. Программа iVentoу с запущенным PXE-Boot сервером и списком подключенных компьютеров

Для установки операционной системы на подконтрольные компьютеры необходимо в их BIOS выбрать приоритет загрузки через сетевую карту по технологии PXE (рис. 2).

Загрузка через сеть предполагает получение компьютером IP адреса по протоколу DHCP от PXE-Boot сервера (рис. 3). И это есть одна из особенностей iVentoу, не требуется отдельная инсталляция DHCP сервера.

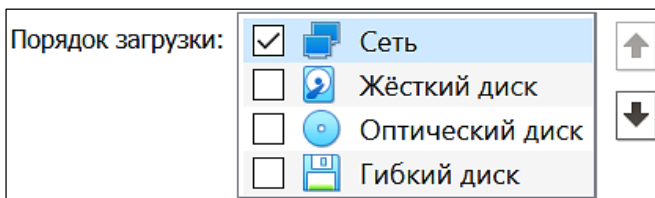


Рис. 2. Выбор приоритета загрузки

```

iPXE (PCI E2:00.0) starting execution...ok
iPXE initialising devices...ok

iPXE 1.0.0+ -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS TFTP HTTP PXE PXEXT Menu

net0: 08:00:27:88:62:34 using 82540em on PCI00:03.0 (open)
[Link:down, TX:0 TXE:0 RX:0 RXE:0]
[Link status: Down (http://ipxe.org/38086101)]
Waiting for link-up on net0... ok
DHCP (net0 08:00:27:88:62:34).....
  
```

Рис. 3. Процесс получение IP адреса в момент загрузки через сетевую карту по технологии PXE

После получения IP адреса появляется меню выбора загрузочного образа операционной системы (рис. 4), который размещен на PXE-Boot сервере iVentoy в директории ...iVentoy\iso в формате *.iso. Затем начинается стандартный процесс установки операционной системы.

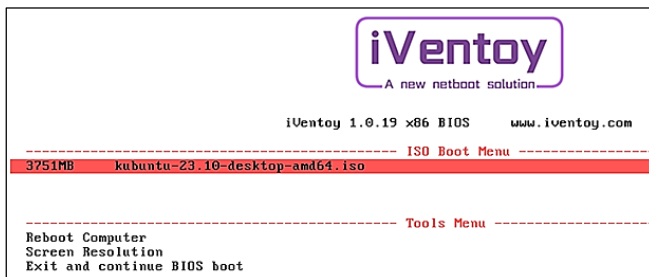


Рис. 4. Меню загрузчика iVentoy, где можно выбрать операционную систему для загрузки

На этом процесс установки операционной системы завершается.

Администрирование операционных систем. Для администрирования сразу множества операционных систем GNU/Linux по сети передачи данных используем систему управления конфигурациями Ansible

[3]. Устанавливаем Ansible на компьютер, с которого будет производиться администрирование. Соединение с подконтрольными компьютерами должно быть доступно по технологии SSH.

На вход Ansible подается конфигурационный файл inventory с IP-адресами компьютеров, которые нужно администрировать, а также «playbook» (сборник), который состоит из tasks (заданий), которые, в свою очередь, состоят из module (модулей) – маленьких операций, подобных командам в терминале Linux [4].

Схема используемых соединений, протоколов и технологий PXE и Ansible представлена на рис. 5.

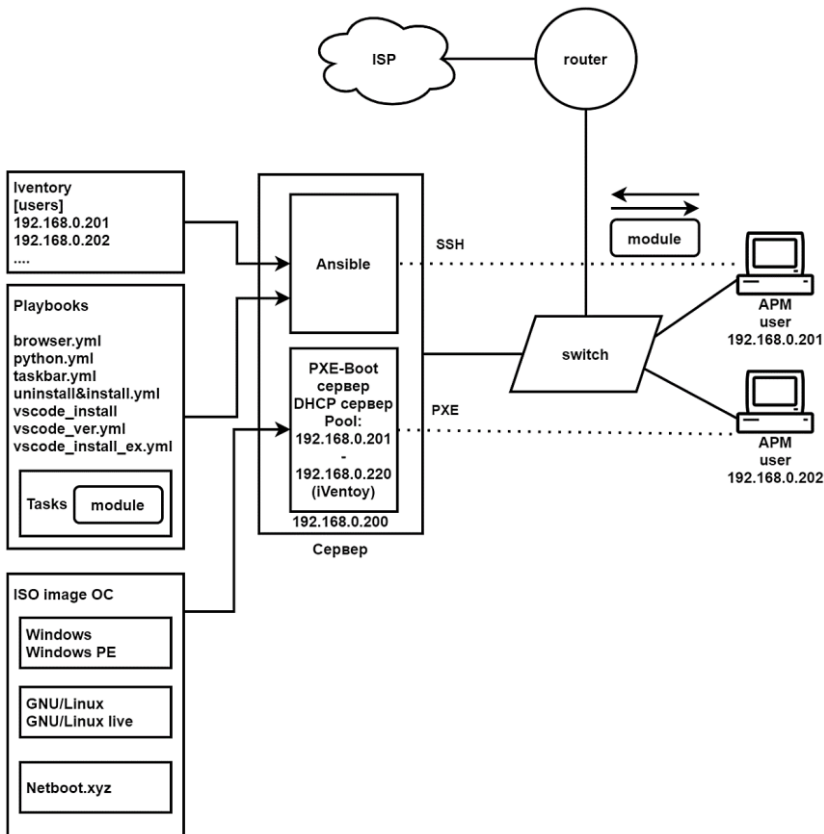


Рис. 5. Топология для установки и администрирования множества операционных систем по сети передачи данных

В приведенной топологии два АРМ/компьютера пользователей подключены с помощью коммутатора к серверу, на котором установлены iVentoys и Ansible. Коммутатор также может быть подключен к маршрутизатору для доступа в Интернет.

Установка приложений. Для примера был составлен `playbook` (сборник) `dep.yml` на языке разметки `YAML`, который выполняет установку пакета `vlc` на подконтрольные узлы, код приведен в листинге 1.

Листинг – Код `YAML` «`playbook`» по установке пакета `vlc`

```
- hosts: users                #группа
  remote_user: user           #удаленный пользователь
  become: yes                 #запрос пароля
  become_method: sudo         #повышенные права на узлах
  gather_facts: yes           #опрос

tasks:

- name : Install vlc package
  apt: name=vlc state=present update_cache=true
```

На рис. 6 приведен результат выполнения данного `playbook` по установке на два узла мультимедиа-проигрывателя `VLC`.

```
max@max-VirtualBox:~/Desktop$ time ansible-playbook dep.yml --ask-become-pass
BECOME password:

PLAY [test_servers] *****
TASK [Gathering Facts] *****
ok: [192.168.0.135]
ok: [192.168.0.75]

TASK [Install vlc package] *****
changed: [192.168.0.75]
changed: [192.168.0.135]

PLAY RECAP *****
192.168.0.135      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
192.168.0.75      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

real    0m22.437s
user    0m2.623s
sys     0m0.449s
```

Рис. 6. Результат выполнения `playbook` по установке пакета мультимедиа-проигрывателя `VLC` на два узла

При выполнении `playbook` в начале запрашивается пароль от удаленных пользователей `Linux`, после этого идет опрос узлов с `IP` адресами указанных в конфигурационном файле `inventory`, если узел готов к работе, то ответный статус напротив его `IP` адреса будет «`ok`». Далее идет выполнение задания по установке пакет. Так как это приводит

к изменениям на подконтрольном узле, то статус будет changed (измененный). В конце выполнения playbook подводится общая статистика по заданиям и узлам.

Заключение. Описанная методология процесса установки и администрирования операционных систем GNU/Linux по сети передачи данных апробирована на электротехническом факультете Пермского национального исследовательского политехнического университета.

Библиографический список

1. Preboot Execution Environment (PXE) Specification [Электронный ресурс]. – URL: <https://pdf4pro.com/view/preboot-execution-environment-pxe-specification-588d2d.html> (дата обращения: 22.05.2024).
2. IVentoy [Электронный ресурс]. – URL: <https://www.iventoy.com/en/index.html> (дата обращения: 22.05.2024).
3. Ansible [Электронный ресурс]. – URL: <https://www.ansible.com/> (дата обращения: 22.05.2024).
4. Ansible documentation [Электронный ресурс]. – URL: <https://docs.ansible.com/> (дата обращения: 22.05.2024).

Сведения об авторах

Масич Григорий Федорович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: masich@icmm.ru

Батюков Максим Георгиевич – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ9-23-1м, г. Пермь, e-mail: maxx14012001@yandex.ru

В.О. Кушев, Е.Г. Зянкина

Пермский государственный национальный
исследовательский университет, Пермь

ИССЛЕДОВАНИЕ ФУНКЦИОНАЛЬНЫХ ТРЕБОВАНИЙ К СИСТЕМЕ АВТОМАТИЗАЦИИ УЧЕТА МАТЕРИАЛЬНО- ТЕХНИЧЕСКИХ СРЕДСТВ КАФЕДРЫ УНИВЕРСИТЕТА

В каждой организации, в том числе и в высших учебных заведениях, используются материально-технические средства для обеспечения рабочего процесса. Такие средства в той или иной степени подлежат учету. Это немаловажная составляющая для обеспечения качественной работы любой организации. Обеспечивать постоянный учет таких средств вручную – трудоемкая и времязатратная задача. В процессе обрабатывается большое количество документов, а также сами материальные средства и оборудование. Также в процессе всего оформляется достаточно большой объем документации, которую необходимо систематизировать, чтобы иметь возможность быстрого доступа к ней с целью контроля, изменения, добавления. В целях решения данной проблемы предлагается автоматизация учета. Автоматизация учета предполагает разработку информационной системы (программного обеспечения), которая позволила бы значительно сэкономить время, облегчить рабочий процесс, а также минимизировать естественные ошибки, которые могут привести к потере денежных средств и ухудшению рабочего и учебного процесса.

Ключевые слова: автоматизация, информационная система, учет, материально-технические средства, кафедра университета.

V.O. Kushev, E.G. Zyankina

Perm State National Research University, Perm

STUDY OF FUNCTIONAL REQUIREMENTS FOR THE AUTOMATION SYSTEM OF MATERIAL AND TECHNICAL FACILITIES OF THE UNIVERSITY DEPARTMENT

Every organization, including higher educational institutions, uses material and technical means to ensure the work process. Such funds are subject to accounting to one degree or another. This is an important component for ensuring high-quality work of any organization. Maintaining constant accounting of such funds manually is a labor-intensive and time-consuming task. In the process, a large number of documents are processed, as well as the materials and equipment themselves. Also, in the process of everything, a fairly large amount of documentation is drawn up, which needs to be systematized in order to be able to quickly access it for the purpose of control, change, and addition. In order to solve this problem, accounting automation is proposed. Automation of accounting

involves the development of an information system (software) that would significantly save time, facilitate the work process, and also minimize natural errors that can lead to loss of money and deterioration of the work and educational process.

Keywords: automation, information system, accounting, material and technical means, university department.

Предметной областью работы является автоматизация учета материально-технических средств кафедры университета (далее по тексту – МТС). Целью является проектирование и разработка приложения для хранения и использования информации о МТС.

Для реализации этой цели использованы следующие инструменты:

- реляционная модель данных;
- инструмент администрирования базы данных DBeaver;
- язык запросов SQL;
- язык программирования Java;
- среда разработки IntelliJ IDEA;
- нотация моделирования BPMN 2.0;
- ПО diagrams.net для создания блок-схем, диаграмм;
- сервис для разработки интерфейсов Figma.

Модель предметной области описывает процессы, происходящие в предметной области, и данные, используемые этими процессами.

При изучении литературы, включающей в себя научные статьи [1–3] и методические указания 119Н МПЗ [4], было выявлено, что нет единого подхода к формированию учета материально-технических средств кафедры университета.

Для формирования информации о возможных функциях и недостатках информационной системы были исследованы аналогичные существующие системы. Так, например, программа Adesk [5] – сервис управленческого учета – содержит в себе блок, посвященный инструментам учета основных материально-технических средств. Основные средства – это активы, которые используются в производстве (оборудование, транспорт, мебель, техника). Исходя из исследования отзывов пользователей, недостатками явились отсутствие учета расходов на доставку, ремонт технических средств, а также неудобный интерфейс.

Продукт от 1С: Предприятие 8. Материально-техническое обеспечение [6] – в данный момент продажи прекращены. Конфигурация «МТО» учитывает специфику материально-технического обеспечения холдингов и крупных промышленных предприятий, обеспечивает широкий спектр возможностей, включающий управление процессом цен-

трализованного формирования потребностей в материалах и оборудовании, управление процессом формирования потребностей в материалах и оборудовании, управление подготовкой и утверждением плана материально-технического обеспечения, планирование закупочной деятельности в соответствии с планом материально-технического обеспечения, управление процессом контроля наличия остатков ресурсов на складах и их распределения в соответствии с потребностями, управление процессом контроля наличия запасов на аварийные и непредвиденные ситуации и за своевременностью их пополнения. Как посчитали пользователи, недостатком и одновременно преимуществом программы является настройка «под себя».

Преимущества и недостатки исследованных информационных систем представлены в таблице.

Сравнение информационных систем

Сервис управленческого учета Adesk	Продукт от 1С: Предприятие 8. Материально-техническое обеспечение
Преимущества	
Добавление основного средства в программу; Фиксирование даты ввода в эксплуатацию; Указание срока службы; Стоимость оборудования; За сколько месяцев была начислена амортизация/ежемесячная амортизация; Формирование отчета о прибыли и убытках	Добавление, изменение, удаление объектов материально-технического обеспечения; Анализ исполнения Плана материально-технического обеспечения; План закупок и перемещений; Анализ закупочных цен; Анализ покрытия потребностей в материально-технических ресурсах
Недостатки	
Отсутствие учета расходов на доставку; Отсутствие показателей для ремонта технических средств; Неудобный интерфейс	Продажи прекращены

При вышеуказанном анализе были обозначены основные возможности разработанной авторами информационной системы:

- возможность добавления в информационную систему каждого МТС кафедры с указанием необходимых характеристик;
- возможность группировки, сортировки таких средств;
- возможность автоматизировать действия с указанными средствами, такие как: осмотр, замена, ремонт, списание, инвентаризация, заказ новых МТС;
- возможность формирования отчетов об остатках материально-технических средств;
- возможность формирования отчетов о расходах на покупку

и обслуживание средств.

В ходе исследования были выявлены следующие основные типы объектов:

– материально-техническое средство (в том числе дочерние типы, такие как мебель, компьютер/ноутбук, компьютерная периферия, инструмент, сетевое оборудование, комплектующее и пр.);

- план-график;
- заявка на осмотр средства (предмета);
- заявка на заказ новых товаров;
- заявка на ремонт имущества;
- заявка на списание имущества;
- заявка на замену МТС.

Исходя из проведенного исследования, система для автоматизации учета материально-технических средств должна соответствовать следующим основным функциональным требованиям:

– авторизация пользователей, используя свое имя пользователя и пароль;

– пользователь должен иметь возможность вводить, создавать, изменять и удалять данные о материально-технических средствах;

– система отправляет пользователю уведомления об изменении статусов состояния МТС, заявки по МТС;

– пользователь имеет возможность создавать, изменять, отправлять на согласование, удалять заявку.

В системе предусмотрена реализация следующих бизнес-процессов для достижения всех указанных функциональных к ней требований:

- рассмотрение заявки на осмотр МТС;
- замена МТС;
- рассмотрение заявки на ремонт МТС;
- проведение ремонтных работ;
- рассмотрение заявки на списание МТС;
- рассмотрение заявки на заказ новых МТС;
- инвентаризация МТС;
- утверждение плана-графика закупки МТС.

Пример бизнес-процесса «Проведение ремонтных работ», описанный в нотации BPMN, представлен на рис. 1, 2.

Для разграничения прав доступа пользователей к тем или иным объектам системы реализованы следующие роли доступа:

- пользователь имущества – доступ на чтение объектов, запуск процесса «Рассмотрение заявки на осмотр МТС»;
- специалист по обеспечению – доступ на чтение, изменение объектов, запуск процессов «Рассмотрение заявки на ремонт МТС», «Рассмотрение заявки на списание МТС», «Рассмотрение заявки на заказ новых МТС», а также к некоторым этапам бизнес-процессов;
- руководитель материально-технического отдела – доступ на чтение, изменение объектов, запуск процессов «Рассмотрение заявки на ремонт МТС», «Рассмотрение заявки на списание МТС», «Рассмотрение заявки на заказ новых МТС», а также к некоторым этапам бизнес-процессов;
- специалист склада – доступ на чтение объектов, а также к некоторым этапам бизнес-процессов;
- специалист по ремонту МТС – доступ на чтение объектов, а также к некоторым этапам бизнес-процессов;
- бухгалтер – доступ к некоторым этапам бизнес-процессов;
- системный администратор – доступ ко всем объектам системы, права на чтение, изменение, удаление.

Пример задачи, назначенной на исполнителя с системной ролью «Специалист по обеспечению» на бизнес-процессе «Проведение подготовительных работ» представлен на рис. 3.

The screenshot shows a web interface for a task titled "Проведение подготовительных работ" (Conducting preparatory work). At the top, there are two fields: "Маршрутный объект" (Route object) with the value "Заявка на ремонт №5" and "Исполнитель задачи" (Task executor) with the value "Иванов И.И.". Below this is a section titled "Постановка задачи" (Task assignment) containing a message: "Добрый день! На Вас назначена задача 'Проведение подготовительных работ'. Вам необходимо полностью осмотреть средство, выявить причину недостатка и определить план его ремонта. Также просим Вас определить является ли поломка средства гарантийным случаем и проставить нужное логическое значение в поле 'Гарантийный случай'." (Good day! A task 'Conducting preparatory work' has been assigned to you. You need to thoroughly inspect the vehicle, identify the cause of the defect and determine the repair plan. We also ask you to determine if the vehicle breakdown is a warranty case and set the appropriate logical value in the 'Warranty case' field.). Below the message is a "Хорошего дня!" (Have a good day!) greeting. At the bottom, there is a "Указать комментарий:" (Specify comment:) field and a "Сменить исполнителя" (Change executor) button. A green box on the left contains the text: "На Вас назначена задача: Проведение подготовительных работ по объекту: Заявка на ремонт №5". At the bottom right, there is a "Завершить задачу" (Complete task) button.

Рис. 3. Интерфейс задачи «Проведение подготовительных работ»

В результате исследования спроектированная авторами информационная система полностью удовлетворяет всем предъявляемым к ней функциональным требованиям с учетом сохранения достоинств и исключения недостатков аналогичных выше исследованных информационных систем.

Библиографический список

1. Антропов В.А., Конова Т.А. Организация материально-технического обеспечения образовательных учреждений высшего профессионального образования // Вестник УРФУ. Серия: экономика и управление. – 2011. – С. 162–173.
2. Казиева Д.А., Хапаева Л.Х. Разработка информационной системы учета материально-технического обеспечения вуза. – М: Мировая наука, 2020. – 10 с.
3. Полищевич С.Ю., Федорищева А.В., Рашупкина В.Н. Учет и контроль материально-технических ресурсов на предприятии // Проблемы экономики и управления строительством в условиях экологически ориентированного развития. – 2019. – № 9. – С. 342–345.
4. Об утверждении Федерального стандарта бухгалтерского учета ФСБУ 5/2019 «Запасы»: Приказ Минфина России от 15 ноября 2019 г. N 180н // Минюст РФ. – 2020. – № 57837.
5. Adesk.ru – Сервис финансовой аналитики для бизнеса [Электронный ресурс]. – URL: <https://adesk.ru/> (дата обращения: 20.05.2024).
6. 1С: Предприятие 8. МТО Материально-техническое обеспечение [Электронный ресурс]. – URL: <https://solutions.1c.ru/catalog/mt0> (дата обращения: 20.05.2024).

Сведения об авторах

Кушев Вадим Олегович – кандидат педагогических наук, доцент кафедры «Информационные технологии» Пермского государственного национального исследовательского университета, г. Пермь, e-mail: valdimk@psu.ru

Зянкина Екатерина Григорьевна – магистрант кафедры «Информационные технологии» Пермского государственного национального исследовательского университета, гр. ММ/О ФИТ-3-2022 НМ, г. Пермь, e-mail: ekaterinazyankina10@gmail.com

Д.С. Курушин, Е.В. Пономарев, В.М. Шепелев

Пермский национальный исследовательский
политехнический университет, Пермь

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИЙ КОМПЬЮТЕРНОГО ЗРЕНИЯ ДЛЯ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОЙ РАБОТЫ ЗА ФРЕЗЕРНО-СВЕРЛИЛЬНЫМ СТАНКОМ

В статье рассматривается применение компьютерного зрения для обеспечения безопасности при работе за фрезерно-сверлильным станком с использованием фреймворка MediaPipe и модели распознавания рук. Представлен алгоритм распознавания опасных ситуаций и программная реализация алгоритма. Решение выполнено на базе Raspberry Pi.

Ключевые слова: компьютерное зрение, фрезерно-сверлильный станок, машинное обучение, безопасность, Raspberry Pi, MediaPipe.

D.S. Kurushin, E.V. Ponomarev, V.M. Shepelev

Perm National Research Polytechnic University, Perm

USE OF COMPUTER VISION TECHNOLOGY TO ENSURE SAFE WORK AT THE MILLING AND DRILLING MACHINE

The paper discusses the application of computer vision to ensure safety when working at a milling and drilling machine using the MediaPipe framework and hand recognition model. The algorithm for recognizing dangerous situations and the software implementation of the algorithm are presented. The solution is based on Raspberry Pi.

Keywords: computer vision, milling and drilling machine, machine learning, security, Raspberry Pi, MediaPipe.

Вопрос контроля промышленной безопасности является важным, поскольку несчастные случаи на производстве приводят не только к травмам и инвалидности работников, но и наносят серьезный ущерб производству в целом, включая крупные страховые выплаты.

Особую опасность в фрезерно-сверлильном станке для здоровья оператора представляет попадание конечностей в движущиеся части. Для предотвращения таких случаев в станках, как правило, встроены конструктивные элементы безопасности, например, защитный кожух и экран. Также предусмотрена техника безопасности [1], согласно которой необходимо перед включением оборудования проверять наличие защит-

ных приспособлений, никогда не пытаться схватиться за подвижные элементы станка во время его работы и другие инструкции. Тем не менее риск возникновения несчастных случаев остается. Например, оператор может проигнорировать правила безопасности или данные системы безопасности могут выйти из строя. Кроме того, существует вероятность случайного попадания конечностей в движущиеся части станка.

Существуют различные подходы предотвращения таких ситуаций при работе за промышленным оборудованием, например, инфракрасные и лазерные датчики или измерение реакции на изменение электрического поля при контакте человека с оборудованием. Однако такие методы имеют свои ограничения в контексте применения к фрезерно-сверлильным станкам. Альтернативным решением являются технологии компьютерного зрения с применением моделей машинного обучения, которые позволяют обеспечить более надежное обнаружение рук оператора в опасных зонах станка.

Основная задача – получение координат рук и сравнение их с координатами опасной зоны, в которой риск получения травмы наиболее высокий. Для решения данной задачи использован фреймворк Mediapipe [2], специально разработанный для создания конвейеров машинного обучения и обработки видеопотока в режиме реального времени. Одним из преимуществ данного фреймворка является наличие обученной модели распознавания рук [3]. Также он предоставляет API для языка программирования Python и требует минимальных ресурсов, что позволит его запускать даже на встроенных устройствах интернета вещей, включая Raspberry Pi.

На входе модель распознавания получает кадр видеопотока, а на выходе – набор координат ключевых точек на руке и ограничивающую рамку. Пример работы модели представлен на рис. 1.

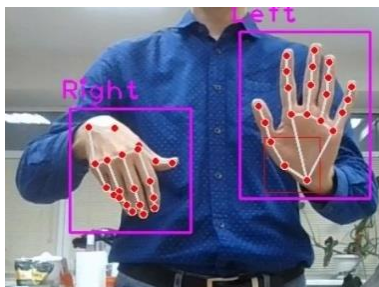


Рис. 1. Пример работы модели распознавания

Полученные координаты можно сравнить с координатами опасной зоны и в случае обнаружения опасной ситуации выключить станок. Соединение станка с Raspberry Pi осуществляется при помощи интерфейса ввода/вывода общего назначения GPIO.

Спроектирован алгоритм распознавания попадания рук в опасную зону станка. Данный алгоритм получает кадры с камеры и сравнивает координаты рук оператора с опасной зоной до тех пор, пока станок находится во включенном состоянии. В случае, если обнаружено попадание, то значение пина GPIO, подключенного к станку, выставляется в ноль, и станок немедленно прекращает свою работу. Блок-схема описанного алгоритма представлена на рис. 2.

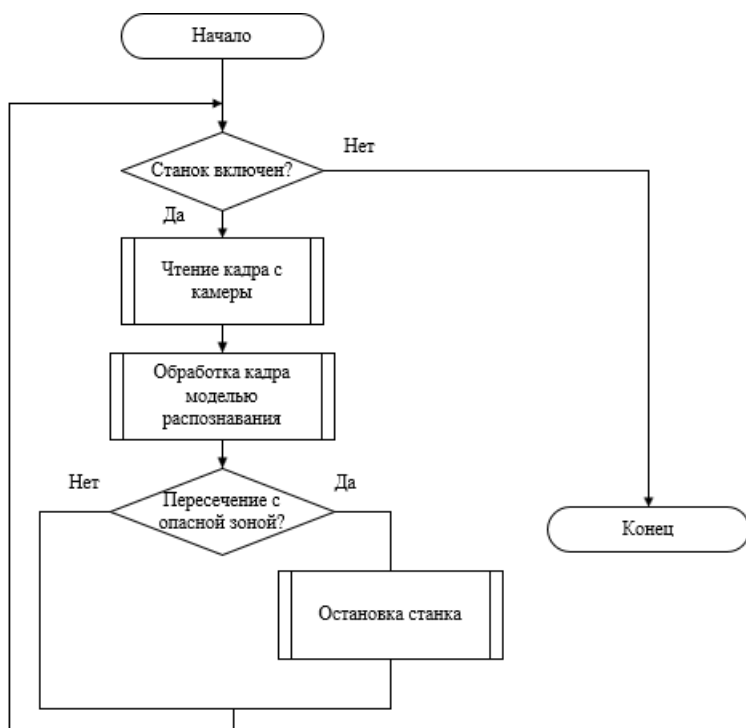


Рис. 2. Блок-схема алгоритма

Алгоритм разработан с использованием языка программирования Python. Для работы с интерфейсом GPIO использована библиотека RPi.GPIO [4]. За чтение кадров с камеры отвечает библиотека OpenCV

[5]. Модель распознавания инкапсулирована в классе HandsDetector, который предоставляет такие методы, как получение ограничивающей рамки рук и проверку на пересечение с опасной зоной.

В листинге представлена реализация алгоритма.

Листинг – Реализация алгоритма распознавания опасной ситуации

```
import RPi.GPIO as GPIO
import cv2
from modules.hand_recognition import HandsDetector

# Подключаемся к камере
cap = cv2.VideoCapture(settings.camera_path)
# Инициализируем модель распознавания
hands_detector = HandsDetector()

# До тех пор, пока станок запущен
while GPIO.input(GPIO_WORKBENCH) == GPIO.HIGH:

    _, image = cap.read()

    # Получаем координаты рук
    hands = hands_detector.detect(image)

    # Если руки есть в кадре
    if hands:

        # Для каждой руки получаем координаты ограничивающей рамки
        for bbox in hands_detector.get_bbox(image, hands):

            # Проверяем пересекают ли руки опасную зону
            if hands_tracker.is_intersect(bbox):

                # Если да, то отключаем станок
                # Устанавливаем пин на 0 - GPIO.LOW
                GPIO.setup(GPIO_WORKBENCH,
GPIO.OUT)
```

```

GPIO.output(GPIO_WORKBENCH,
GPIO.LOW)

GPIO.setup(GPIO_WORKBENCH, GPIO.IN)

# Освобождаем ресурсы
cap.release()
hands_detector.release()

```

Ключевую роль в обеспечении безопасности при работе за станком играют показатели эффективности подсистемы распознавания опасных ситуаций:

- 1) частота обработки кадров – сколько кадров в секунду способна обрабатывать модель распознавания;
- 2) время отклика – промежуток времени между распознаванием опасной ситуации и остановкой станка;
- 3) точность – отношение количества правильно распознанных ситуаций к общему числу тестовых случаев;
- 4) полнота – доля действительно опасных ситуаций, которые были правильно распознаны системой.

Время отклика включает в себя время реакции подсистемы распознавания и время срабатывания электромеханической части, которое в среднем составляет не более 100 мс.

В таблице приведены значения основных показателей на основе экспериментов.

Показатели эффективности подсистемы распознавания

Показатель		Значение
Частота обработки кадров, кадры/с	Минимальное	8
	Максимальное	16
	Среднее	14
Время отклика, с	Минимальное	0,163
	Максимальное	0,225
	Среднее	0,171
Точность, %		89,76
Полнота, %		81,54

Таким образом, разработан алгоритм распознавания опасных ситуаций при работе за станком. В программной реализации использован фреймворк MediaPipe, предоставляющий модель распознавания рук на видео. Результаты данной работы могут быть внедрены в фрезерно-сверлильные станки для повышения безопасности на производстве.

Библиографический список

1. Инструкция по эксплуатации фрезерно-сверлильного станка JET JMD 15 [Электронный ресурс]. – URL: https://jet-online.ru/media/pdf_descriptions/JMD-15.pdf (дата обращения: 27.01.2024).
2. Гид по MediaPipe [Электронный ресурс]. – URL: <https://developers.google.com/mediapipe/solutions/guide> (дата посещения: 01.02.2024).
3. Гид по MediaPipe [Электронный ресурс]. – URL: <https://developers.google.com/mediapipe/solutions/guide> (дата посещения: 01.02.2024).
4. OpenCV: OpenCV-Python Tutorials [Электронный ресурс]. – URL: https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html (дата обращения: 02.02.2024).
5. RPi.GPIO [Электронный ресурс]. – URL: <https://sourceforge.net/projects/raspberry-gpio-python/> (дата обращения: 15.02.2024).

Сведения об авторах

Курушин Даниил Сергеевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: dan973@ya.ru

Пономарев Егор Витальевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: egor.ponomarevov2002@gmail.com

Шепелев Вадим Михайлович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: vadimblin258@gmail.com

**С.И. Келлер, С.А. Камянецкий,
Д.О. Тронина, А.Л. Погудин**

Пермский национальный исследовательский
политехнический университет, Пермь

КОНСТРУКТОРСКИЕ И ТЕХНОЛОГИЧЕСКИЕ ОСНОВЫ РАЗРАБОТКИ ТРЕХМЕРНОГО ДИСПЛЕЯ

В данной статье проведен анализ возможности отображения трехмерных моделей на трехмерной матрице светодиодов. Также рассмотрены основные проблемы уже существующих светодиодных кубов. Разработаны основные требования к светодиодным кубам. Проиллюстрирован вариант подключения системы управления светодиодами.

Ключевые слова: схемотехника, отображение информации, периферийные устройства.

**S.I. Keller, C.A. Kamyanetskii,
D.O. Tronina, A.L. Pogudin**

Perm National Research Polytechnic University, Perm

DESIGN AND TECHNOLOGICAL BASIS FOR THE DEVELOPMENT OF THREE-DIMENSIONAL DISPLAY

This paper analyzes the possibility of displaying three-dimensional models on a three-dimensional matrix of LEDs. The main problems of already existing LED cubes are also considered. The basic requirements for LED cubes are developed. A variant of LED control system connection is illustrated.

Keywords: circuitry, information display, peripheral devices.

Выбор темы обусловлен высокой стоимостью и несовершенством существующих светодиодных кубов. Можно повысить интерес к данному направлению и качество обучения, автоматизировав и сократив денежные затраты на создание. Использование трехмерного экрана при обучении компьютерной графике способствует развитию технического творчества в области схемотехники, робототехники, технологии интернета вещей, программировании. Помимо этого, трехмерный экран может применяться для выполнения лабораторных работ, курсовых и дипломных проектов для определенных направлений.

Актуальность работы обусловлена необходимостью в инновационных методах отображения информации, а также в отображении четырехмерных объектов на 3D-дисплее.

Цель проекта – разработка программно-аппаратного обеспечения периферийного устройства многоцветного трехмерного экрана с разрешением $16 \times 16 \times 16$ светодиодов для повышения конкурентоспособности направлений по информационным технологиям в результате повышения заинтересованности абитуриентов ПНИПУ электротехнического факультета.

Объектом исследования является трехмерный RGB экран с разрешением $16 \times 16 \times 16$ светодиодов.

Предметом исследования является процесс и инструменты по разработке, реализации и внедрении. в учебный процесс трехмерного экрана.

Для достижения поставленной цели требуется решение следующих задач:

- а) провести обзор и анализ предметной области;
- б) провести обзор существующих светодиодных кубов и разработать модернизации;
- в) разработать основные требования к светодиодным кубам;
- г) подобрать элементную базу и разработать имитационную модель трехмерного экрана в программе Proteus или в иных средств разработки и отладки электронных схем.

С развитием электроники и микроэлектроники появились новые возможности в области визуальных технологий. Мониторы, дисплеи, лазерные проекторы, шлемы виртуальной реальности, очки дополненной реальности, фасадные экраны изменили то, как мы взаимодействуем с информацией и окружающим миром.

Одним из ярких продуктов этого прогресса стал светодиодный куб. Это устройство использует трехмерный массив светодиодов для создания трехмерных изображений и объемных эффектов. Светодиодные кубы стали ответом на потребность в инновационных и захватывающих методах визуализации информации и развлечений.

В рекламе и маркетинге светодиодные кубы привлекают внимание и предлагают инновационные подходы к презентации продуктов и услуг. Помимо этого, светодиодный куб можно использовать в архитектуре и дизайне интерьеров, они служат декоративными элементами, создавая уникальные образы и освещая пространство, а внедрение трехмерного экрана в систему умного дома поможет организациям привлечь больше

клиентов и выделиться на фоне конкурентов. В рамках системы умного дома с голосовым ассистентом светодиодный куб можно использовать в качестве отображения аватара умного дома – своеобразного интеллектуального существа. Помимо этого, если в проект внедрить камеры, технологию компьютерного зрения и сделать экран высоким, то такой экран может превратиться в удивительный инструмент для обучения танцам. На экране будет отображаться танцующий силуэт человека, демонстрируя танцевальные движения и элементы, а пользователь сможет все это повторять, следуя за силуэтом. Этот подход открывает перед пользователями новые горизонты во взаимодействии с технологиями, где ограничения существуют лишь в пределах воображения.

Ограничения трехмерных светодиодных матриц:

а) На обычных дисплеях обычно стараются расположить пиксели как можно ближе друг к другу для повышения разрешения, однако в кубе необходимо иметь возможность видеть через него и требуется большее расстояние между вокселями [1]

б) Увеличение разрешения приведет к дополнительным существенным затратам, куб $8 \times 8 \times 8$ состоит из 512 светодиодов, 9 сдвиговых регистров и 8 транзисторов, при увеличении разрешения вдвое получится куб $16 \times 16 \times 16$ и на его сборку уже уйдет 4096 светодиодов, 36 сдвиговых регистра в случае с визуализацией одним цветом или 100 в случае с RGB, 32 транзистора.

После анализа существующих решений выявлены следующие недостатки:

а) ограниченное разрешение экрана, предлагаемое компаниями в размерах $3 \times 3 \times 3$, $4 \times 4 \times 4$ и $8 \times 8 \times 8$ светодиодов, может ограничить возможности вывода информации;

б) ограниченная цветовая гамма, варианты отображения только одного цвета светодиодов, предлагаемые некоторыми компаниями, могут затруднить восприятие двух объектов, расположенных рядом;

в) отсутствие поддержки передачи данных о состоянии светодиодов с компьютера на устройство. Все объекты выводятся из памяти микроконтроллера, требуя загрузки новой прошивки или данных с SD-карты;

г) высокая стоимость. Трехмерные экраны, удовлетворяющие высоким требованиям, по стоимости оцениваются от 100 000 до 5 000 000 руб. в зависимости от характеристик и размеров. Наилучшие варианты изготавливаются на заказ компанией seekway.cn или командами инженеров-энтузиастов.

Изучив существующие решения разработаны требования к экрану:

а) необходимо добиться разрешения $16 \times 16 \times 16$, такое разрешение позволит отображать более детализированные объекты с сохранением баланса стоимости/необходимые возможности;

б) устройство должно поддерживать отображение множества световых оттенков. Адресные RGB светодиоды не подходят ввиду существенного удорожания проекта, минимальная цена на 2024 г. составляет 42 000 руб. за 4096 светодиодов, также потребуются затраты на дополнительное питание;

в) расстояние между светодиодами должно гарантировать возможность просмотра светодиодов, расположенных в дальних рядах, через все остальные;

г) блок питания, питающий всю систему, должен быть подобран по мощности в соответствии электрическому расчету потребителей;

д) светодиодное устройство должно быть переносным.

Предлагаемый вариант решения проблемы питания и управления. Куб представляет из себя 16 прозрачных матриц светодиодов, расположенных друг над другом. Все катоды светодиодов, расположенных на одном уровне, соединены между собой для повышения прозрачности, уменьшения количества проводов и существенного уменьшения количества управляющих выводов. Все аноды светодиодов, расположенных в одном столбце, также соединены между собой в соответствии с цветом [2]. Способ соединения контактов для куба с разрешением $5 \times 5 \times 5$ светодиодов изображен на рис. 1, 2.

В случае проектирования переносного светодиодного куба оптимальным интервалом между светодиодами является 60 мм. Использование интервала 60 мм приведет к увеличению размеров светодиодного куба $16 \times 16 \times 16$ до 95 см, что затруднит его переноску. В случае использования интервала 35 мм размер устройства составит 56 см.

В качестве системы управления была выбрана связка, состоящая из компьютера и микроконтроллера с каскадом сдвиговых регистров и прп-транзисторов, к коллектору которых подключены катоды светодиодов одного слоя. Компьютер позволяет обеспечить вычислительную мощность и гибкость программирования светодиодного куба, программист сможет управлять матрицами путем изменения содержимого файла на любом языке программирования.

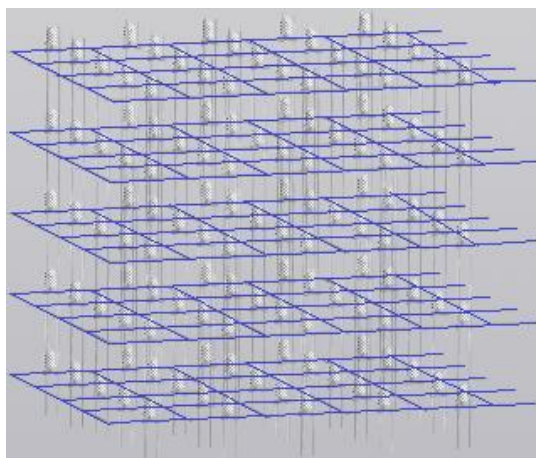


Рис. 1. Параллельное подключение катодов светодиодов

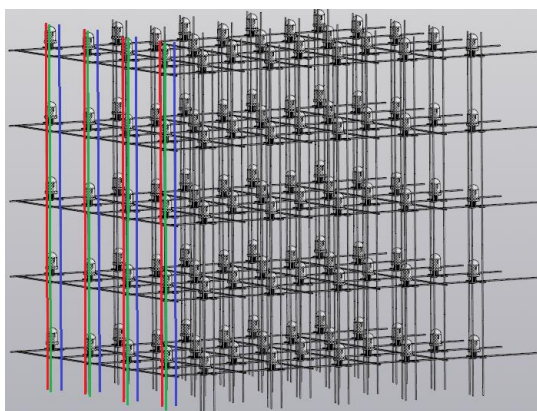


Рис. 2. Параллельное подключение анодов светодиодов

С помощью специальной программы-передатчика данные из файла на компьютере передаются через USB в память микроконтроллера. Микроконтроллер является связующим звеном между компьютером и аппаратной логикой. Светодиодный куб содержит 4096 светодиодов, и управление каждым светодиодом непосредственно с помощью микроконтроллера было бы неэффективным и невозможным из-за ограниченного количества доступных выводов. Каскад сдвиговых регистров позволяет увеличить количество управляемых светодиодов, создавая

цепочку регистров, в которой данные передаются последовательно от одного регистра к другому. Такая технология позволяет управлять большим количеством светодиодов с помощью 3 выводов микроконтроллера. Помимо этого, сдвиговые регистры обеспечивают синхронизацию передачи данных к светодиодам, что важно для создания координированных световых эффектов. Микроконтроллеры имеют встроенные таймеры и счетчики, которые могут быть использованы для управления временными характеристиками эффектов и анимации.

Транзисторы используются для управления высоким током, который требуется для питания слоев светодиодов. Микроконтроллеры часто имеют ограничения по току на их выводах, поэтому использование транзисторов позволяет управлять большими токами, не нагружая микроконтроллер. На рис. 3 изображена электрическая принципиальная схема светодиодного куба $5 \times 5 \times 5$.

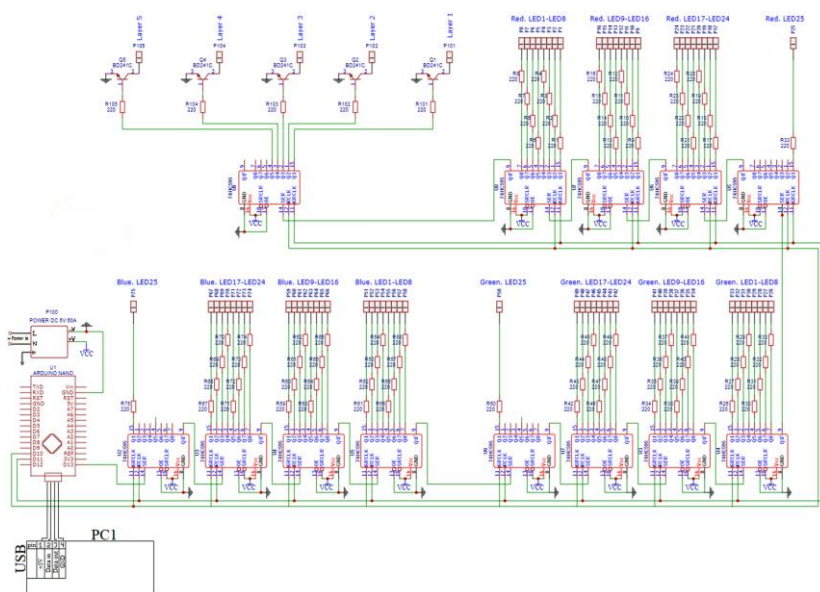


Рис. 3. Электрическая принципиальная схема

При построении электрической принципиальной схемы светодиодного куба $16 \times 16 \times 16$ увеличивается количество сдвиговых регистров. Однако особенность заключается в том, что каждый квадратный слой необходимо разбить на два прямоугольных и к каждой части подклю-

чить свой транзистор BD241C. Такая особенность обусловлена характеристиками транзистора, поскольку максимальный ток на коллекторе составляет 3 А. Путем разбиения слоя на две части на коллектор транзистора при максимальной нагрузке будет приходиться 2,6 А.

Сдвиговые регистры поделены на блоки, 1 сдвиговый регистр управляет транзисторами слоев, 4 следующих управляют красным цветом 25 светодиодов, далее 4 следующих управляют зеленым цветом и оставшиеся 4 управляют синим цветом.

Данные передаются последовательно от сдвигового регистра к сдвиговому регистру, сначала отправляется 1 байт, отвечающий за состояние транзисторов, затем 4 байта, отвечающие за красный цвет, следующие 4 байта за зеленый цвет, завершающие 4 байта отвечают за синий цвет [3].

Заключение. В ходе исследования предметной области светодиодных кубов был проведен обзор и анализ существующих разработок в данной области. Были выявлены основные принципы построения светодиодных кубов, их особенности и возможные области применения.

Проведенный обзор существующих светодиодных кубов позволил выявить их основные преимущества и недостатки, а также ограничения. Были разработаны модернизации, направленные на улучшение функциональности, энергоэффективности и удобства использования светодиодных кубов.

На основе анализа предметной области и обзора существующих разработок были сформулированы основные требования к светодиодным кубам.

Для реализации требований к светодиодным кубам была подобрана элементная база и разработана имитационная модель трехмерного экрана в программе EasyEDA, помимо этого была разработана 3D-модель светодиодного куба в программе КОМПАС-3D. Это позволило оценить работоспособность и эффективность выбранных компонентов, а также провести отладку электронных схем и подобрать оптимальное расстояние между светодиодами.

Библиографический список

1. Что такое шаг пикселя и почему это важно [Электронный ресурс]. – URL: <https://timeline.ru/2021/06/10/what-is-pixel-pitch/> (дата обращения: 17.05.2024).

2. Светодиодный куб, общий катод слоя, общий анод столбца [Электронный ресурс]. – URL: https://amperkot.ru/blog/8x8x8_LED_Cube/ (дата обращения: 18.05.2024).

3. Основы цифровой электроники. Регистр сдвига [Электронный ресурс]. – URL: <https://habr.com/ru/companies/ruvds/articles/754774/> (дата обращения: 18.05.2024).

Сведения об авторах

Келлер Степан Ильич – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-1б, г. Пермь, e-mail: stepan.keller@gmail.com

Камянецкий Сергей Александрович – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-1б, г. Пермь, e-mail: kamyanetskysergei@yandex.ru

Тренина Дарья Олеговна – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-1б, г. Пермь, e-mail: dasha.tronina17@gmail.com

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

А.И. Корзникова., А.Б. Федоров

Пермский национальный исследовательский
политехнический университет, Пермь

**СИСТЕМА АВТОМАТИЗИРОВАННОГО СБОРА
ПОКАЗАНИЙ ИНДИВИДУАЛЬНЫХ ПРИБОРОВ
УЧЕТА КОММУНАЛЬНЫХ УСЛУГ В ЖКХ
ПО ТЕХНОЛОГИИ ИНТЕРНЕТА ВЕЩЕЙ**

В этой статье исследуется автоматизация через интернет вещей в жилищно-коммунальном хозяйстве. Интернет вещей (IoT) стал одним из ключевых направлений развития технологий в последние годы. Одним из перспективных направлений применения IoT является жилищно-коммунальное хозяйство. Системы умного дома все больше внедряются в многоквартирные дома и частные квартиры, что позволяет жильцам существенно упростить управление бытовыми процессами и сделать жизнь более комфортной. В статье рассматривается структура данной системы. Показаны схемы структуры системы, внешних соединений системы учета, автоматического управления. Описан общий ход алгоритма модели построения предметной области.

Ключевые слова: интернет вещей, жилищно-коммунальное хозяйство, интернет-провайдер, архитектура системы, сеть передачи данных, безопасность, автоматизация.

A.I. Korznikova, A.B. Fedorov

Perm National Research Polytechnic University, Perm

**SYSTEM FOR AUTOMATED COLLECTION OF INDICATORS
OF IPU OF PUBLIC SERVICES IN HUSING AND UTILITIES
USING INTERNET OF THINGS TECHNOLOGY**

This article explores automation through the Internet of Things in the housing and communal services sector. The Internet of Things (IoT) has become one of the key areas of technology development in recent years. One of the promising areas of application of IoT is housing and communal services. Smart home systems are increasingly being implemented in apartment buildings and private apartments, which allows residents to significantly simplify the management of household processes and make life more comfortable. The article discusses the structure of this system. Diagrams of the system structure, external connections of the accounting system, and automatic control are shown. The general flow of the algorithm for constructing a domain model is described.

Keywords: Internet of things, housing and communal services, Internet provider, system architecture, data transmission network, security, automation.

Важным аспектом внедрения IoT в жилищно-коммунальном хозяйстве (ЖКХ) является экономия ресурсов. Системы умного учета и управления энергопотреблением позволяют значительно экономить электроэнергию, воду и другие ресурсы. Например, умные счетчики могут контролировать расход электроэнергии и управлять работой бытовой техники, оптимизируя расход электричества и снижая коммунальные платежи.

Также IoT в ЖКХ позволяет улучшить качество обслуживания жильцов. Системы мониторинга и диагностики оборудования позволяют оперативно выявлять неисправности и предупреждать аварийные ситуации. Автоматическое управление системами комфорта (освещение, отопление, кондиционирование) позволяет адаптировать условия проживания к индивидуальным потребностям каждого жильца.

Цель исследования. В нашем случае существует необходимость разработки структуры автоматизированной системы для оптимизации процесса сбора и передачи данных показаний ИПУ ЖКХ. Технические аспекты связаны с необходимостью внедрения современных технологий связи и информационных систем, обеспечения совместимости различных устройств и систем, а также гарантирования безопасности передаваемой информации.

Организационные вопросы включают в себя разработку стандартов и нормативов, согласование действий различных сторон (жильцов, управляющих компаний, разработчиков систем IoT) и обеспечение поддержки и обучения пользователей.

Правовые аспекты связаны с необходимостью регулирования сбора, хранения и обработки персональных данных, ответственности за аварийные ситуации и нарушения безопасности.

Материалы и методы. Архитектура системы состоит из нескольких основных компонентов, каждый из которых будет выполнять определенную функцию. На первом этапе определены устройства, которые используются для сбора и передачи данных по сети IoT.

Для этого могут быть использованы различные устройства, такие как оптические терминалы, маршрутизаторы, контроллеры, счетчики воды, газа, электроэнергии и др.

Сеть передачи данных играет ключевую роль в процессе передачи информации от устройств к облачным сервисам. Для этого можно использовать современные технологии связи, такие как сети 5G, LoRaWAN или NB-IoT. Эти сети обеспечат надежную передачу данных даже на

большие расстояния и в условиях слабого сигнала. Облачные сервисы представляют собой платформу для сбора, хранения и обработки данных, поступающих от устройств. Здесь осуществляется анализ полученных данных, их визуализация, а также управление системой в целом.

Способы взаимодействия между компонентами системы определяются протоколами обмена данными, которые обеспечат безопасную и эффективную передачу информации.

Для этого могут быть использованы различные протоколы, такие как MQTT, CoAP, HTTP и др.

Для обеспечения безопасности передаваемых данных необходимо использовать современные методы шифрования и аутентификации, чтобы исключить возможность несанкционированного доступа к информации.

Результаты. Разработанная аппаратная структура SCADA-системы представлена на рис. 1 [1].

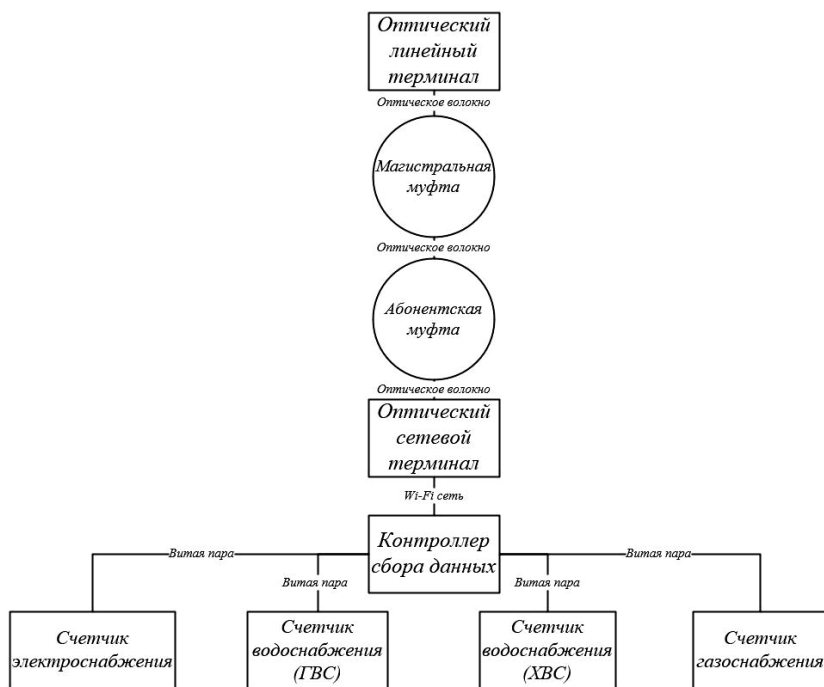


Рис. 1. Аппаратная структура SCADA-системы

Структура SCADA-системы включает в себя следующие элементы:

- датчики и измерительные устройства: они используются для сбора данных о состоянии объектов контроля;
- контрольные устройства и исполнительные механизмы: они отвечают за управление процессами на объекте;
- коммуникационные средства: они обеспечивают передачу данных между различными устройствами SCADA-системы;
- центральный компьютер SCADA: здесь происходит обработка данных, визуализация информации, управление процессами и анализ результатов;
- пользовательские интерфейсы: обеспечивают доступ пользователей к данным и функциям SCADA-системы;
- базы данных: используются для хранения и обработки данных.

Смоделированная схема внешних соединений системы учета коммунальных услуг представлена на рис. 2.

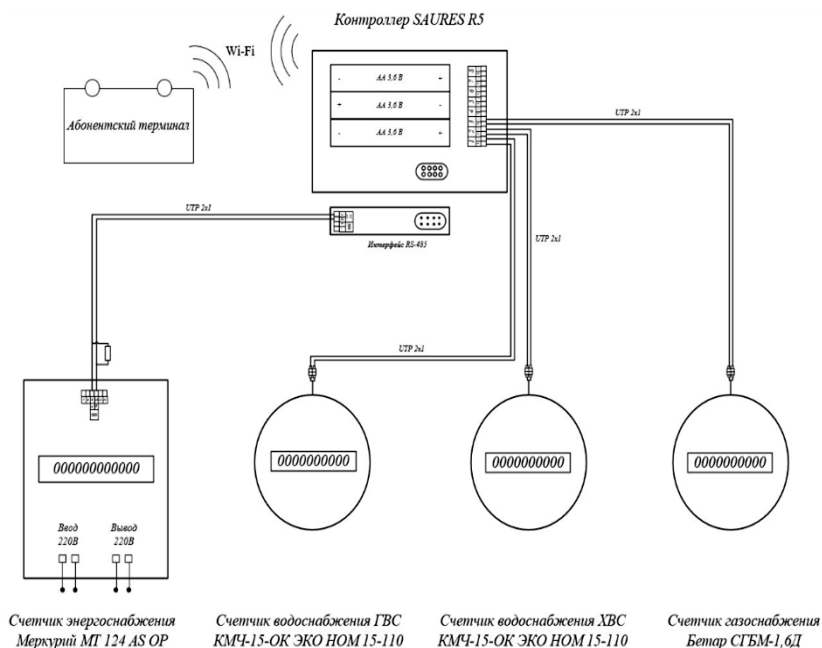


Рис. 2. Схема внешних соединений системы учета

Структура системы автоматического управления включает в себя несколько компонентов, работающих в синхронизации для обеспечения эффективного функционирования системы:

- SCADA-система, которая отвечает за сбор, мониторинг и управление данными от различных устройств в реальном времени;
- сервер BRAS (Broadband Remote Access Server), который предоставляет широкополосный доступ к сети для конечных пользователей;
- сервер RADIUS (Remote Authentication Dial-In User Service), используемый для аутентификации и авторизации пользователей;
- база данных – хранилище информации о пользователях, устройствах, настройках и других данных.

Спроектированная структура системы автоматического управления представлена на рис. 3.

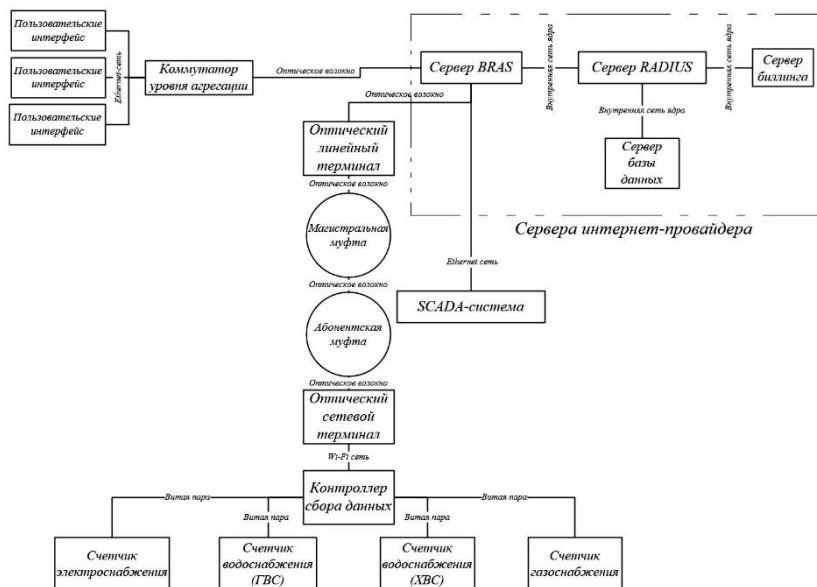


Рис. 3. Структура системы автоматического управления

Заключение. В работе разработаны схема внешних соединений системы учета, структура SCADA-системы, структура системы автоматического управления.

Библиографический список

1. Самуйлова К.Е., Шалимова И.А., Кулябова Д.С. Сети и телекоммуникации: учебник и практикум для СПО. – М.: Юрайт, 2019. – 363 с.

Сведения об авторах

Корзникова Анастасия Игоревна – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-19-1бз, г. Пермь, e-mail: slastecapka@mail.ru

Федоров Андрей Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pstufab@gmail.com

И.И. Исламов, Д.В. Яруллин

Пермский национальный исследовательский
политехнический университет, Пермь

**РАЗРАБОТКА МОДЕЛИ И АЛГОРИТМА
АВТОМАТИЗИРОВАННОГО ФОРМИРОВАНИЯ
ТЕСТОВЫХ МАТЕРИАЛОВ**

В данной статье рассматривается модель автоматизированного формирования тестовых материалов, предназначенная для создания сбалансированных и целенаправленных тестов на предприятиях. Описанная модель данных охватывает ключевые аспекты разработки тестов, включая определение тем, типов вопросов, уровней сложности и количества вопросов. В рамках модели также представлен банк вопросов, содержащий разнообразные задания, которые могут быть использованы для составления тестов. Статья описывает алгоритм формирования теста, начиная с инициализации данных и выбора темы и заканчивая проверкой и окончательным составлением теста. Предложенная модель может быть адаптирована для различных предметных областей и типов тестов, что делает ее универсальным инструментом для образовательных платформ и систем дистанционного обучения.

Ключевые слова: алгоритм, модель, формирование теста, множества, тестовый материал, автоматизация.

I.I. Islamov, D.V. Yarullin

Perm National Research Polytechnic University, Perm

**DEVELOPMENT OF A MODEL AND ALGORITHM
FOR AUTOMATED FORMATION OF TEST MATERIALS**

This article discusses a model of automated generation of test materials designed to create balanced and targeted tests at enterprises. The described data model covers key aspects of test development, including defining topics, question types, difficulty levels, and number of questions. The model also presents a question bank containing a variety of tasks that can be used to compile tests. The article describes the algorithm for forming the test, starting with data initialization and topic selection, and ending with verification and final preparation of the test. The proposed model can be adapted for various subject areas and types of tests, which makes it a universal tool for educational platforms and distance learning systems.

Keywords: algorithm; model; test formation; sets; test material; automation.

Оценка технологических компетенций сотрудников является одной из ключевых задач управления персоналом на современных предприятиях. Тестирование является вторым по популярности средством проверки знаний сотрудников предприятия и выступает важным инструментом, позволяющим объективно и эффективно оценить уровень профессиональных компетенций сотрудников. Однако процесс создания тестов часто требует значительных временных и ресурсных затрат [1].

Современные технологии позволяют автоматизировать процесс формирования тестовых материалов, что способствует повышению качества и консистентности оценивания.

Модель формирования тестовых материалов. Автоматизированное формирование тестовых материалов включает в себя создание модели данных, которая определяет структуру самого теста, тестовых вопросов, вариативность ответов и другие аспекты при разработке тестовых материалов.

Стандартный тест состоит из следующих элементов:

1. Тема теста: определение области знаний, которые должны быть охвачены тестом.

2. Типы вопросов: разнообразие типов вопросов, такие как закрытый вопрос с выбором ответа, открытый вопрос с написанием не односложного ответа, соотнесение, заполнение пропусков и другие типы.

3. Сложность вопросов: оценка уровня сложности вопросов.

4. Количество вопросов: определение общего количества вопросов в тесте.

5. Банк вопросов: набор вопросов, из которого будут выбираться конкретные задания для формирования теста [2].

Элементы теста можно представить как модель в виде множеств, которые будут составлять сам тест.

Тема теста определяет область знаний, которую должен охватывать тест. Множество тем обозначим как T :

$$T = \{\text{область знаний}\}. \quad (1)$$

Область знаний определяется целью проведения тестирования и тем, какие аспекты необходимо проверять у сотрудников предприятия.

Типы вопросов в тесте могут варьироваться, включая закрытые вопросы с выбором ответа, открытые вопросы, задания на соотнесение и заполнение пропусков. Множество типов вопросов обозначим как Q_t :

$$Q_t = \{MCQ, \text{Открытый}, \text{Соотнесение}\}, \quad (2)$$

где MCQ – множественный выбор ответа.

Вопросы в тесте классифицируются по уровням сложности. Основными являются легкий, средний и трудный уровни. Множество уровней сложности обозначим как D :

$$D = \text{легкий, средний и трудный.} \quad (3)$$

Банк вопросов представляет собой набор всех возможных ранее сформированных системой вопросов, доступных для формирования теста. Множество вопросов в банке обозначим как B :

$$B = \{b_1, b_2, b_3, \dots, b_n\}. \quad (4)$$

Каждый вопрос из множества B определяется следующими атрибутами:

- $b_i t$ – тема вопроса;
- $b_i q_t$ – тип вопроса;
- $b_i d$ – уровень сложности;
- $b_i c$ – правильный ответ;
- $b_i a$ – варианты ответов для закрытых вопросов.

Исходя из модели формирования тестовых материалов, возможно представление теста как совокупности вопросов.

В таблице представлены возможные вариации вопросов при формировании тестовых материалов.

Пример вопросов для определенной области знаний

№	Тема	Тип	Сложность	Правильный ответ	Варианты ответов (для MCQ)
b1	Область знаний	MCQ	Легкий	a	[a, b, c, d]
b2	Область знаний	Открытый	Средний	Ответ	–
b3	Область знаний	Соотнесение	Трудный	Пара	–

Окончательный сформированный тест состоит из подмножеств вопросов B' из банка вопросов множества B . Каждое подмножество представляет собой вариации возможных вопросов теста:

$$B'_{MCQ, \text{ Легкий}} = \{b_1, b_2, b_3\}, \quad (5)$$

$$B'_{\text{Открытый, Средний}} = \{b_4, b_8\}, \quad (6)$$

$$B'_{\text{Соотнесение, Трудный}} = \{b_5, b_6, b_7\}. \quad (7)$$

Таким образом, сформированный тест будет выглядеть как объединение подмножеств B' :

$$T = B'_{MCQ, \text{ Легкий}} \cup B'_{\text{Открытый, Средний}} \cup B'_{\text{Соотнесение, Трудный}}. \quad (8)$$

Полученная модель формирования тестовых материалов отражает набор ключевых вопросов, необходимых для создания теста и проведения дальнейшего тестирования сотрудников предприятия.

Алгоритм формирования тестовых материалов. На основе анализа модели формирования тестовых материалов разработан алгоритм, адаптирующий методы построения тестов на естественном языке. Алгоритм представлен на рисунке.



Рис. Алгоритм формирования тестовых материалов

На вход алгоритма поступает коллекция из предполагаемой темы тестирования, необходимые типы вопросов и их количество. Для конечного пользователя всегда доступен выбор ранее сформированных под исходную тему вопросов. В случае, если пользователь неудовлетворен предложенными вопросами из банка вопросов, доступен вариант формирования новых вопросов по данной теме [3]. Результатом работы алгоритма является коллекция готовых для проведения тестирования вопросов по заданной входной теме.

Заключение. Разработана модель автоматизированного формирования тестовых материалов, направленная на оценку технологических компетенций сотрудников предприятия. Описанная модель данных включает в себя все необходимые элементы для создания эффективных и сбалансированных тестов, таких как выбор темы, типы вопросов, уровни сложности и общий объем теста. Приведенный алгоритм позволяет систематически и эффективно формировать тесты, обеспечивая высокую точность и объективность оценивания.

Библиографический список

1. Оценка сотрудников не ради оценки [Электронный ресурс]. – URL: https://topcareer.ru/blog/hr_assessment_startexam (дата обращения: 15.05.2024).
2. Ларин С.Н., Малков У.Х. Современные подходы к моделированию тестов: система требований, преимущества и недостатки, основные этапы разработки [Электронный ресурс] // Интернет-журнал «Мир науки». – 2016. – Т. 4, № 1. – URL: <http://mir-nauki.com/PDF/04PDMN116.pdf> (дата обращения: 20.05.2024).
3. Панкова Д.А. Методика применения алгоритма формирования системы автоматического тестирования для проверки знаний сотрудников // Известия СПбГЭУ. – 2012. – № 3.

Сведения об авторах

Исламов Ильназ Ильясович – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: islamov204@yandex.ru

Яруллин Денис Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: d.v.yarullin@ya.ru

А.М. Русакова, Д.А. Карлов

Пермский национальный исследовательский
политехнический университет, Пермь

**РАЗРАБОТКА ПРОГРАММНО-АППАРАТНОГО
КОМПЛЕКСА ДЛЯ АВТОМАТИЧЕСКОГО ПОИСКА
ДЕФЕКТОВ ДЕТАЛЕЙ АВИАДВИГАТЕЛЯ
С ИСПОЛЬЗОВАНИЕМ КОМПЬЮТЕРНОГО ЗРЕНИЯ**

В этой статье исследуется возможность поиска дефектов на деталях авиадвигателя. Данная задача решается при использовании компьютерного зрения. В данной статье рассматриваются способы реализации данного программного-аппаратного комплекса, методы компьютерного зрения и анализа изображений.

Ключевые слова: компьютерное зрение, авиадвигатель, Python, детали, изображения.

A.M. Rusakova, D.A. Karlov

Perm National Research Polytechnic University, Perm

**DEVELOPMENT OF A SOFTWARE AND HARDWARE
COMPLEX FOR AUTOMATIC SEARCH OF DEFECTS
IN AIRCRAFT ENGINE PARTS USING COMPUTER VISION**

This article explores the possibility of finding defects on aircraft engine parts. This problem is solved using computer vision. This article discusses methods for implementing this given hardware and software complex, methods of computer vision and image analysis.

Keywords: computer vision, aircraft engine, Python, details, Images.

Авиадвигатели являются сердцем любого воздушного судна, обеспечивая его надежную работу и безопасность полетов. Поэтому важно обеспечить высокое качество производства и тщательный контроль каждой детали, входящей в состав двигателя. Одним из эффективных методов контроля является применение технологии компьютерного зрения. Технология компьютерного зрения позволяет создавать системы автоматического контроля качества, способные обнаруживать даже мельчайшие дефекты на поверхности деталей. Применение этой технологии в производстве авиадвигателей позволяет значительно повысить точность и скорость контроля, снизить вероятность человеческой ошибки и обеспечить более надежное качество продукции.

Одним из ключевых преимуществ технологии компьютерного зрения является возможность обнаружения дефектов в реальном времени без необходимости остановки производственного процесса. Системы компьютерного зрения могут сканировать поверхность деталей на лету, выявляя любые отклонения от стандартов качества. Это позволяет оперативно реагировать на возможные проблемы и предотвращать выпуск дефектных изделий. Другим важным преимуществом технологии компьютерного зрения является ее способность работать с большим объемом данных и проводить анализ изображений с высокой точностью. Системы машинного зрения могут обучаться распознавать различные типы дефектов, что делает их более эффективными и адаптивными к изменениям в производственной среде.

Использование технологии компьютерного зрения для поиска дефектов деталей авиадвигателя не только повышает качество продукции и безопасность полетов, но также снижает затраты на контроль качества и увеличивает производительность производства. Это инновационное решение открывает новые возможности для совершенствования процессов производства авиационной техники и повышения конкурентоспособности компаний в отрасли.

Рынок компьютерного зрения сейчас переживает бурный рост с прогнозируемым увеличением с 22 млрд долл. в 2023 г. до 50 млрд к 2030 г. при 21,4 % совокупного годового прироста с 2024 по 2030 г. (рисунок) [1].

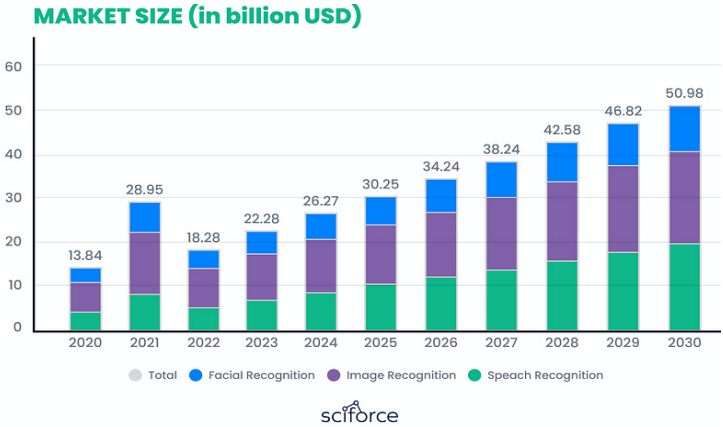


Рис. Прогноз развития рынка компьютерного зрения

Эта группа технологий имитирует человеческое зрение, но работает гораздо быстрее, используя сложные алгоритмы, большие объемы данных и камеры. Системы компьютерного зрения могут быстро анализировать тысячи предметов на огромных площадях или обнаруживать крошечные дефекты, невидимые человеческому глазу [1].

Технология. Компьютерное зрение (Computer Vision, CV) – это область искусственного интеллекта, связанная с анализом изображений и видео. Она включает в себя набор методов, которые наделяют компьютер способностью «видеть» и извлекать информацию из увиденного. Системы состоят из фото- или видеокamеры и специализированного программного обеспечения, которое идентифицирует и классифицирует объекты. Они способны анализировать образы (фотографии, картинки, видео, штрих-коды), а также лица и эмоции [2].

Вот некоторые ключевые аспекты технологии компьютерного зрения:

1. Обработка изображений: компьютерное зрение включает в себя методы обработки изображений, такие как фильтрация, улучшение качества, сегментация (разделение изображения на части), распознавание объектов и т.д.

2. Извлечение признаков: этот этап включает в себя выделение ключевых признаков из изображений, которые помогут системе распознавать объекты или выполнять другие задачи анализа.

3. Обучение и классификация: для обучения системы компьютерного зрения на определенных данных используются различные методы машинного обучения, такие как нейронные сети, SVM (Support Vector Machines) и др. Это позволяет классифицировать объекты на изображениях, распознавать образы и выполнять другие задачи.

4. Распознавание объектов: системы компьютерного зрения могут распознавать объекты на изображениях, определять их положение, классифицировать и даже отслеживать движущиеся объекты.

5. Анализ видео: компьютерное зрение также включает в себя анализ видеопотоков для выявления шаблонов, динамики движения объектов, обнаружения аномалий и т.д.

Технология компьютерного зрения находит широкое применение в различных областях, таких как медицина, автомобильная промышленность, робототехника, безопасность, контроль качества производства, анализ изображений из космоса и многое другое.

Методы обнаружения дефектов на деталях авиадвигателя.

Компьютерное зрение может быть использовано для определения дефектов на деталях с помощью различных методов и алгоритмов. Вот несколько распространенных способов:

1. Анализ текстур: данный метод основывается на анализе текстурных характеристик поверхности детали. Компьютерное зрение может обнаружить аномалии в текстуре, такие как трещины, сколы или неровности, которые могут указывать на дефекты.

2. Сравнение с эталоном: в этом случае компьютерное зрение использует эталонное изображение или модель для сравнения с текущим изображением детали. Если есть расхождения или отличия, то это может указывать на наличие дефектов.

3. Методы машинного обучения: методы машинного обучения, такие как нейронные сети или SVM (Support Vector Machines), могут быть использованы для классификации изображений деталей на «дефектные» и «недефектные». Для этого система обучается на большом наборе изображений с размеченными дефектами, чтобы научиться распознавать характерные признаки дефектов.

4. Обнаружение контуров и форм: компьютерное зрение может использовать алгоритмы обнаружения контуров и форм для выявления аномальных форм или несоответствий заданной геометрии детали. Например, если форма детали должна быть круглой, то любое отклонение от круглой формы может указывать на дефект.

5. Использование инфракрасного изображения: инфракрасное компьютерное зрение позволяет обнаруживать тепловые аномалии на поверхности деталей, которые могут быть связаны с дефектами, такими как перегрев, неправильное соединение или протечка.

Это лишь несколько примеров способов определения дефектов на деталях с помощью компьютерного зрения. Фактический выбор метода зависит от конкретной задачи и требований к точности и эффективности системы.

Преимущества использования компьютерного зрения для обнаружения дефектов на деталях авиадвигателя. Разберем подробнее преимущества применения компьютерного зрения в поиске дефектов деталей авиадвигателя:

1. Высокая точность: системы компьютерного зрения способны обнаруживать даже мельчайшие дефекты на поверхности деталей, которые могли бы быть упущены при визуальном контроле человеком. Это поз-

воляет исключить возможность производства и использования дефектных деталей, что в свою очередь повышает безопасность полетов.

2. Скорость и эффективность: автоматизированные системы компьютерного зрения работают намного быстрее, чем человеческий глаз, что позволяет проводить контроль качества на производственной линии в реальном времени без задержек. Это сокращает время процесса контроля и увеличивает производительность.

3. Снижение затрат: использование технологии компьютерного зрения позволяет снизить затраты на контроль качества за счет автоматизации процесса и исключения необходимости в большом количестве операторов. Это также уменьшает вероятность человеческой ошибки и снижает количество бракованных изделий.

4. Надежность и стабильность: системы компьютерного зрения работают по заданным алгоритмам и не подвержены усталости или человеческим факторам, что обеспечивает стабильное качество контроля на протяжении всего производственного процесса.

5. Адаптивность и масштабируемость: технология компьютерного зрения может быть легко настроена для обнаружения различных типов дефектов и адаптирована к изменениям в производственной среде. Это делает ее универсальным инструментом для контроля качества в различных условиях.

Интегрированные в производственный процесс решения на основе компьютерного зрения выполняют задачи контроля качества быстрее и точнее, чем человек. Экономия времени на дефектоскопию продукции или сырья в условиях массового производства значительно повышает общую производительность труда. Системы CV (Computer Vision) гарантированно обеспечивают выход с производственных линий, только соответствующих стандартам качества изделий. Внедрение автоматизированных систем дефектоскопии способно значительно снизить уровень непродуктивных потерь при расфасовке продукции и уменьшить непроизводительные затраты [3].

Таким образом, использование технологии компьютерного зрения в поиске дефектов деталей авиадвигателя не только повышает качество продукции и безопасность полетов, но также улучшает эффективность производственных процессов, снижает затраты и обеспечивает стабильность и надежность контроля качества.

Заключение. Использование технологии компьютерного зрения для поиска дефектов на деталях авиадвигателя представляет собой важный шаг в современной отрасли авиационного производства. Эта

инновационная методика позволяет обнаруживать даже мельчайшие дефекты, которые могут оказать серьезное влияние на безопасность и надежность работы двигателя.

Благодаря анализу текстур, сравнению с эталоном, применению методов машинного обучения, обнаружению контуров и форм, а также использованию инфракрасного изображения, компьютерное зрение способно точно и эффективно выявлять различные виды дефектов, что помогает предотвращать аварийные ситуации и снижать риски эксплуатации авиадвигателей. Этот подход не только повышает качество контроля производства авиационных деталей, но и сокращает время и затраты на инспекцию. В результате применения технологии компьютерного зрения улучшается общая безопасность полетов и надежность авиационной техники, что делает ее более конкурентоспособной и соответствующей самым высоким стандартам качества в индустрии авиации.

Библиографический список

1. Компьютерное зрение в 2024 году: Главные задачи и направления [Электронный ресурс] // Хабр. – URL: <https://habr.com/ru/companies/otus/articles/810207/> (дата обращения: 22.05.2024).

2. Что такое компьютерное зрение и где его применяют [Электронный ресурс] // РБК Тренды. – URL: <https://trends.rbc.ru/trends/industry/5f1f007e9a794756fafbfa83/> (дата обращения: 22.05.2024).

3. Как системы компьютерного зрения помогают контролировать качество продукции [Электронный ресурс] // Tadviser. – URL: https://www.tadviser.ru/index.php/Статья:Как_системы_компьютерного_зрения_помогают_контролировать_качество_продукции/ (дата обращения: 22.05.2024).

Сведения об авторах

Русакова Анастасия Максимовна – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16 г. Пермь, e-mail: n_rus_23@mail.ru

Карлов Денис Анатольевич – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: denis.karlov.78@yandex.ru

В.Н. Чирков, А.А. Петренко

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ИНТЕРНЕТ-МАГАЗИНА ДЛЯ СИСТЕМЫ ОБЩЕСТВЕННОГО ПИТАНИЯ ИП ПЕРШИН

В статье исследуется возможность создания интернет-магазина для системы общественного питания. Рассматриваются особенности реализации человеко-машинного интерфейса интернет-магазина, контингент пользователей и работников системы общественного питания. Приведен минимальный набор параметров для работы с клиентами. Рассмотрен процесс выбора платежной системы для интернет-магазина для системы общественного питания.

Ключевые слова: интернет-магазин, общественное питание, автоматизация системы общественного питания.

V.N. Chirkov, A.A. Petrenko

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF AN ONLINE STORE FOR THE PUBLIC CATERING SYSTEM OF IP PERSHIN

The article explores the possibility of creating an online store for a catering system. The features of the implementation of the human-machine interface of the online store, the contingent of users and employees of the catering system are considered. The minimum set of parameters for working with clients is given. The process of choosing a payment system for an online store for a catering system is considered.

Keywords: online store, catering, automation of the catering system.

На современном этапе развития бизнеса риски потери дохода в бизнесе могут быть очень значительны, а конкуренция достаточно велика. Для того чтобы решить эти проблемы, компании нужно выбрать инструмент автоматизации, который будет приносить клиентов без необходимости делать постоянно одни и те же действия работникам компании. Для обеспечения повышения качества оказываемых услуг в компании «ИП Першин» необходимо внедрить систему, которая позволит учесть особенности ведения деятельности в области общественного питания, свести к минимуму определенные категории рисков и сделать сам процесс оказания услуг общественного питания устойчивым и рентабельным.

Анализ деятельности системы общественного питания. После кризиса 2014–2016 гг. в ресторанном бизнесе наблюдается уверенный рост. В 2018–2019 гг. в сфере общественного питания наметились новые тренды. Эксперты прогнозируют дальнейшее развитие сетевых проектов, сдвиг клиентского интереса в сторону более демократичных форматов – стритфуда и фастфуда, укрепление роли соцсетей и омниканальности в маркетинге, рост рынка доставки. По данным 2ГИС, за прошедший год во всех городах-миллионниках наблюдается прирост предприятий общественного питания: в Москве на 9,2 %, Уфе – 7,3 %, Санкт-Петербурге – 5,5 %, Новосибирске – 5 %, Омске – 1,9 % [1].

Компания, для которой разрабатывается интернет-магазин, занимается приготовлением и доставкой свертков. Еще в ассортименте компании есть продукты фастфуда и напитки. Заказы от клиентов принимаются по телефону, повар принимает и выполняет заказ, курьер его доставляет. Помимо заказов по телефону, у компании была группа в социальной сети «ВКонтакте». Основным недостатком существующей системы является долгий по времени и сложный в области контроля процесс формирования заказов.

Таким образом, в целях преодоления сложившейся ситуации на рынке общественного питания, а также повышения конкурентоспособности субъектов, оказывающих услуги в сфере общественного питания, необходимо внедрять автоматизированную систему управления отношениями с потребителями, что будет способствовать развитию долгосрочных отношений и позволит избежать затрат на постоянное привлечение новых клиентов.

Моделирование интернет-магазина для системы общественного питания. Одним из методов улучшения качества и эффективности работы организации, в частности системы общественного питания, является моделирование бизнес-процессов.

Программа Integrated Computer-Aided Manufacturing (ICAM), предназначенная для интегрированной компьютеризации производства (США, конец 70-х гг. XX в.), выявила потребность в более совершенных способах обмена информацией и методах анализа производственных систем для всех специалистов, занимающихся проблемой. В рамках программы ICAM для удовлетворения этих потребностей была разработана методология IDEF (ICAM Definition). Наиболее популярными стали следующие части методологии.

IDEF0 – используется для создания функциональных моделей

(Integration Definition For Function Modeling), отражающих структурированное изображение функций производственной системы или среды, а также информации и объектов, связывающих эти функции.

IDEF1 (IDEF1X) – используется для создания информационных моделей, представляющих структуру информации, необходимую для поддержки функций систем. Многие CASE-инструменты, используемые при разработке современных систем управления базами данных (СУБД), позволяют автоматически генерировать программный код из IDEF1X диаграмм [2].

Для того чтобы создать работоспособный интернет-магазин для системы общественного питания, нужно сначала построить ее IDEF0-схему. Она изображена на рис. 1.

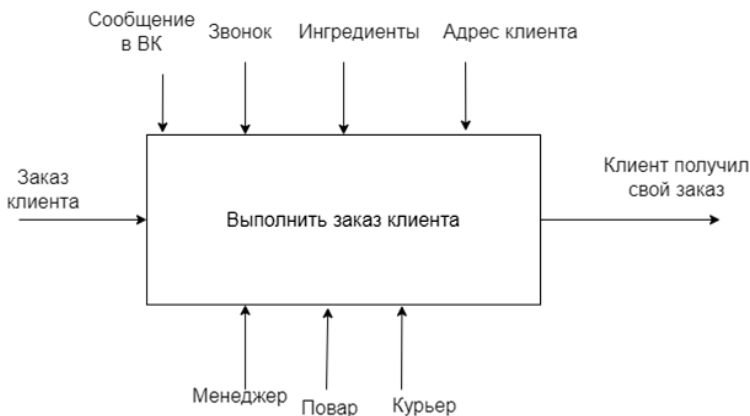


Рис. 1. IDEF0

На рис. 2 изображена модель деятельности AS-IS.

Таким образом, была составлена модель компании AS-IS при помощи IDEF. Понимание процессов, протекающих в компании, поможет создать интернет-магазин, который будет наиболее эффективно автоматизировать эти процессы.

Для того чтобы разработать интернет-магазин для системы общественного питания, нужно построить IDEF0-схему, как должно быть, т.е. TO-BE. Схема изображена на рис. 3.

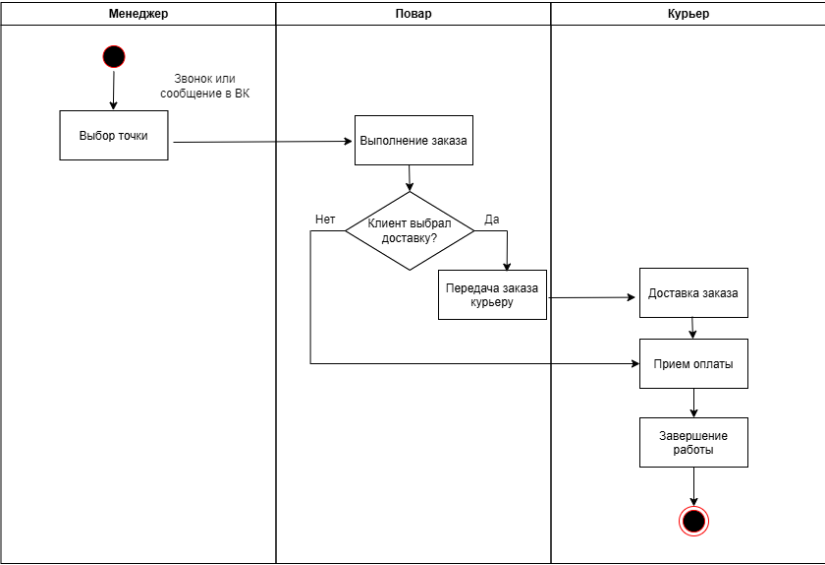


Рис. 2. Модель деятельности (AS-IS)

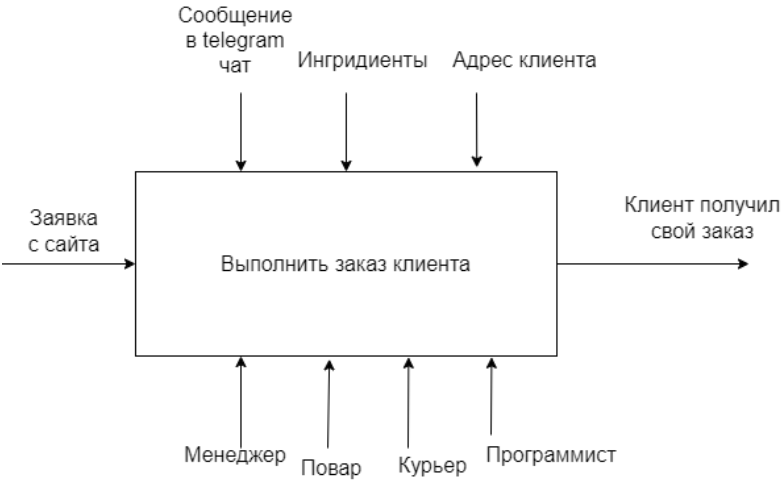


Рис. 3. IDEF0 для модели TO-BE

На рис. 4 представлена схема процесса (TO-BE): «Обработка заказа клиента».

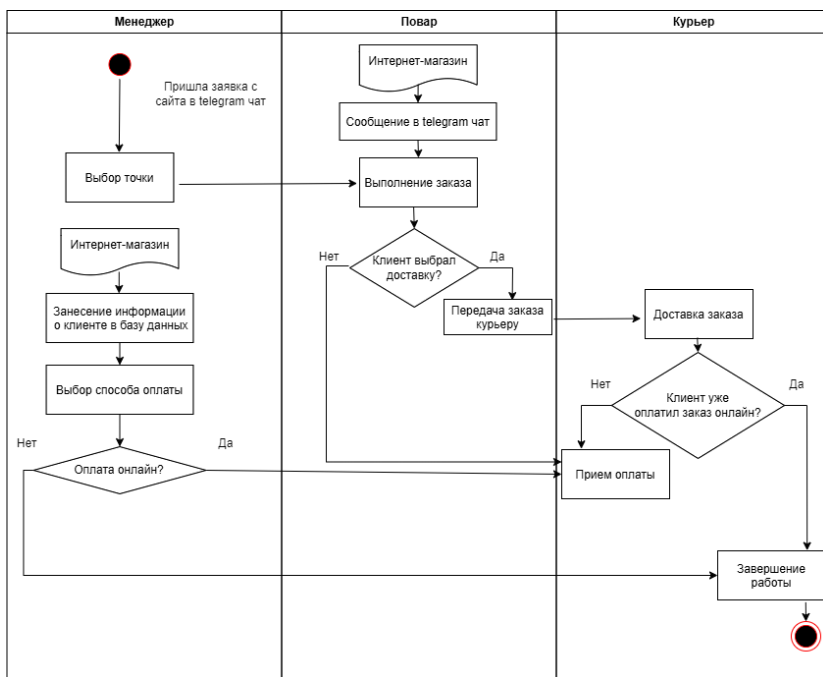


Рис. 4. Модель деятельности (TO-BE)

На представленной схеме TO-BE был проанализирован и подготовлен регламент процесса с подробным описанием действий, его участников, указанием входов/выходов.

Таким образом, оптимизация описанного процесса позволит сэкономить время сотрудников системы общественного питания на взаимодействие друг с другом. Снижается нагрузка на менеджера в связи с тем, что имеется единая база и нет необходимости направлять повара и курьерам информацию о заказе клиента.

Технология реализации интернет-магазина. Интернет-магазин разрабатывается на CMS MODx. Системные таблицы MODx создаются автоматически в процессе установки CMS на сервере. Для создания своих таблиц использовалось расширение «MIGX» для MODx.

Для создания дизайна использовался сайт «Figma». Результат создания дизайна представлен на рис. 5.

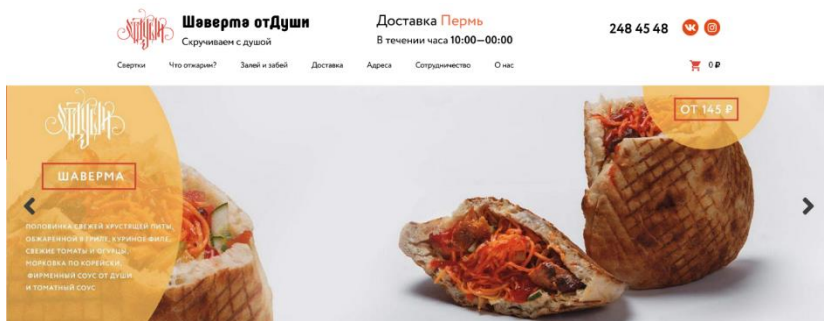


Рис. 5. Дизайн web-приложения

После создания полного интерфейса интернет-магазина в «Figma» было принято решение использовать следующие средства реализации: HTML, CSS, JS, PHP.

В процессе работы использовались следующие средства разработки: bootstrap-grid, fontawesome, jquery, owl-carousel, magnific-popup, page-scroll-2-id, jquery-lazyload, equal-heights, jGrowl, jquery-mask, jscookie.

Для оплаты онлайн решили использовать платежную систему «Сбербанк». Чтобы была возможность подключить эту платежную систему, необходимо указать реквизиты заказчика и правила возврата товара.

В конце интернет-магазин был размещен на хостинге компании «link-host». Для размещения потребовалось перенести все файлы сайта на хостинг, создать базу данных и импортировать в нее локальный дамп базы данных MODx.

Таким образом, интернет-магазин был интегрирован с платежной системой заказчика и интегрирован с популярной CMS MODx.

Закключение. У интернет-магазина для системы общественного питания, как правило, присутствуют свои незначительные особенности, такие как брендовые ограничения и требования работников компании. Количество работников компании ограничено не более 10.

Необходимо минимизировать количество обрабатываемой ежесекундной информации. Дизайн необходимо ориентировать на два направления: клиентов системы общественного питания и брендовые особенности. Подключение интернет-магазина к выбранной платежной системе заказчика.

Библиографический список

1. Рождественская Л.Н., Чередниченко Л.Е., Рогова О.В. Технология открытия предприятия питания: учеб. пособие. – Новосибирск: Изд-во НГТУ, 2019. – 134 с.
2. Киселев Д.Ю., Киселев Ю.В., Вавилин А.В. Функциональное моделирование на базе стандарта IDEF0: метод. указания. – Самара: Изд-во СГАУ, 2014. – 20 с.

Сведения об авторах

Чирков Владислав Николаевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-2б, г. Пермь, e-mail: vlad_chirkov@rambler.ru

Петренко Александр Анатольевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: hawk321@mail.ru

А.А. Петренко, Н.А. Матвеев

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ ПРОВЕДЕНИЯ ТУРНИРОВ ПО СПОРТИВНОМУ ПРОГРАММИРОВАНИЮ

Соревнования по спортивному программированию являются важной частью ИТ-сферы. В связи с этим остро стоит вопрос об оптимизации процесса проведения соревнований по спортивному программированию. Цель работы – автоматизация процесса проведения соревнований по спортивному программированию. В статье проанализирован процесс проведения соревнований по спортивному программированию. Предложен проект системы, автоматизирующей этот процесс.

Ключевые слова: программное обеспечение, спортивное программирование, соревнования, автоматизация.

A.A. Petrenko, N.A. Matveev

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF AN INFORMATION SYSTEM FOR CONDUCTING SPORTS PROGRAMMING TOURNAMENTS

Sports programming competitions are an important part of the IT field. In this regard, the issue of optimizing the process of conducting sports programming competitions is acute. The purpose of the work is to automate the process of conducting sports programming competitions. The article analyzes the process of conducting sports programming competitions. A draft system automating this process is proposed.

Keywords: software, sports programming, competitions, automation.

На сегодняшний день соревнования по спортивному программированию стали не только международным видом спорта, но и способом доказать свой профессионализм, положительно выделиться среди других кандидатов на должность, получить предложение о работе напрямую от компаний-спонсоров. Для проведения соревнования по спортивному программированию необходимо разместить информацию о соревновании в публичном доступе, произвести регистрацию участников соревнования, предоставить участникам доступ к правилам и условиям задач. В процессе соревнования также необходимо проверять решения участ-

ников соревнования, регистрировать время сдачи решения и отвечать на вопросы по условиям задач от участников соревнования [1–3].

В статье рассматривается процесс проведения соревнований по спортивному программированию на кафедре Информационных технологий и автоматизированных систем. В данном случае процесс проведения соревнования возможно разделить на пять подпроцессов: регистрация участников соревнования, распечатка и раздача условий задач, отправка решений на проверку, проверка решений, подведение итогов (рис. 1).



Рис. 1. Текущий процесс проведения соревнований по спортивному программированию

В ходе анализа существующих решений, таких как ejudge, PCMS, Codeforces и Contester, были выделены их слабые и сильные стороны, на основании чего были сформированы требования к новой системе. Таким образом, были сформированы следующие требования к новой системе:

- интуитивно понятный интерфейс для пользователей продуктов-аналогов;
- регистрация соревнований по спортивному программированию;
- старт и завершение соревнований по спортивному программированию в заранее заданные сроки;
- проверка решений задач спортивного программирования, представленных в виде исходного кода программы;
- оценка решений задач спортивного программирования по результатам проверки;
- подведение итогов соревнования по спортивному программированию;
- регистрация и аутентификация пользователей и команд.

Для автоматизации процесса проведения соревнований по спортивному программированию необходимо реализовать автоматизированную систему, которая упростит имеющиеся процессы, исключит из них участие персонала или позволит отказаться от части процессов. С помо-

щью автоматизированной системы можно отказаться от процесса «Распечатка и раздача условий задач», так как все условия задач будут представлены участникам в web-интерфейсе. Кроме того, благодаря использованию системы в процессах регистрации участников соревнования, проверки решений и подведении итогов соревнования можно исключить участие членов жюри, что позволяет сократить персонал комиссии жюри. Чтобы достичь таких результатов, необходимо добавить процесс «Создание соревнования в системе», в котором члены жюри вносят в систему всю необходимую информацию для проведения соревнования, такую как условия задач, тестовые таблицы. Процесс проведения соревнования по спортивному программированию с использованием проектируемой системы представлен в виде диаграммы в нотации IDEF0 на рис. 2.

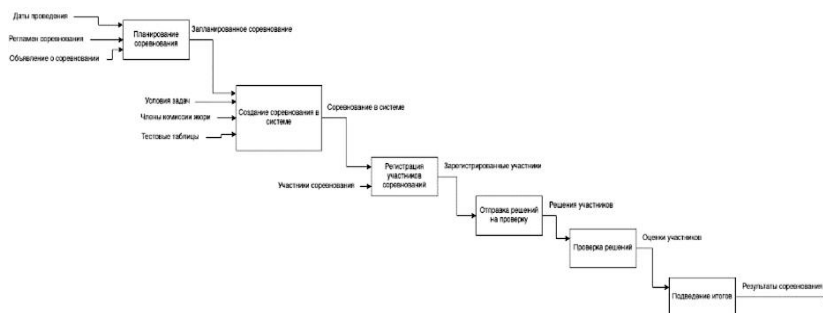


Рис. 2. Спроектированный процесс проведения соревнований по спортивному программированию

Реализуемое приложение является типичным web-приложением, состоящим из фронтенда, бэкенда и базы данных. Для компонента бэкенда было принято решение реализовывать монолитную архитектуру приложения, так как такая архитектура не накладывает дополнительных трудозатрат на разработку и является предпочтительной для команд с малым количеством разработчиков [4].

В ходе разработки информационной системы были использованы следующие средства реализации:

- 1) язык программирования Python – используется для разработки бэкенда, с применением фреймворка Flask [5];
- 2) СУБД MariaDB – реляционная система управления базой данных, используемая для хранения информации;

3) язык программирования JavaScript – используется для разработки фронтенда с применением фреймворка React.js.

Таким образом, в статье проанализирован процесс проведения соревнований по спортивному программированию, предложена информационная система для его автоматизации. При использовании информационной системы сокращается количество работы, которую необходимо выполнить комиссии жюри, что позволяет сократить трудозатраты на проведение соревнований.

Библиографический список

1. Мальцев С.П. Олимпиадное программирование: учебно-метод. пособие. – Улан-Удэ: БГУ, 2019. – 135 с.
2. Янцев В.В. Web-программирование на Python: учеб. пособие для вузов. – 2-е изд., стер. – СПб.: Лань, 2023. – 180 с.
3. Мартин Р. Чистый код: создание, анализ и рефакторинг. Библиотека программиста. – СПб.: Питер, 2016. – 464 с.
4. Гагарина Л.Г., Кокорева Е.В., Сидорова-Виснадул Б.Д. Технология разработки программного обеспечения: учеб. пособие. – М.: Форум, 2018. – 400 с.
5. Елисеев А.И., Минин Ю.В., Гриднев В.А. Разработка веб-приложений с использованием фреймворка Flask. В 2 ч.: учеб. пособие: 2 ч.– Тамбов: Изд-во Тамбов. гос. техн. ун-та, ЭБС АСВ, 2020. – Ч. 1. – 81 с.

Сведения об авторах

Петренко Александр Анатольевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: hawk321@mail.ru

Матвеев Николай Александрович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: nitronik19@gmail.com

**Д.О. Тронина, С.И. Келлер,
С.А. Камянецкий, А.Л. Погудин**

Пермский национальный исследовательский
политехнический университет, Пермь

**ЛАБОРАТОРНЫЙ СТЕНД ПО ИССЛЕДОВАНИЮ
СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ
И РЕГУЛИРОВАНИЯ НА БАЗЕ ПРОГРАММИРУЕМЫХ
ЛОГИЧЕСКИХ КОНТРОЛЛЕРОВ**

В статье рассматриваются обзор и обоснование выбора программируемых логических контроллеров, разработана типовая лабораторная работа по исследованию системы автоматического регулирования «Управление водосборником».

Ключевые слова: программируемый логический контроллер, системы автоматического управления и регулирования, водосборник.

**D.O. Tronina, S.I. Keller,
S.A. Kamyanetsky, A.L. Pogudin**

Perm National Research Polytechnic University, Perm

**LABORATORY STAND FOR RESEARCH
OF AUTOMATIC CONTROL AND REGULATION
SYSTEMS ON THE BASIS OF PLK**

The article deals with the review and justification of the choice of programmable logic controllers, developed a typical laboratory work on the study of automatic regulation system "Watershed control".

Keywords: programmable logic controller, automatic control and regulation systems, catchment.

В наше время практически невозможно найти неавтоматизированный технологический процесс. Автоматизация включает в себя управление, защиту, контроль, регулирование любого технологического процесса. Выбор темы обусловлен отсутствием универсального лабораторного стенда для проведения исследований систем автоматического управления (САУ) и систем автоматического регулирования (САР). Существующие лабораторные стенды часто имеют такие ограничения

как: устаревшие технологии, недостаточные возможности, ограниченная функциональность. Данная проблема затрудняет изучение и понимание работы САУ, а также создание новых более эффективных алгоритмов регулирования.

Актуальность темы обусловлена важностью изучения и исследования САУ и САР в современной технике. Данная разработка позволит проводить эксперименты необходимые для повышения работоспособности и эффективности различных систем управления. Также стоит отметить необходимость использования ПЛК в разработке лабораторного стенда для обеспечения гибкости, высокой надежности и интеграции с различными устройствами, так как ПЛК поддерживает работу со множеством различных датчиков, исполнительных устройств и приводов.

Объект исследования: лабораторный стенд по исследованию САУ и САР. Предмет исследования: процесс и инструменты по созданию и внедрению в учебный процесс лабораторного стенда на базе ПЛК.

Цель исследования: создание лабораторного стенда, который позволит студентам и инженерам изучать и исследовать системы автоматического управления и регулирования с использованием ПЛК.

Для достижения поставленной цели требуется решение следующей задачи: разработать типовую лабораторную работу по исследованию системы автоматического управления и регулирования.

Обзор программируемых логических контроллеров. Рассмотрим характеристики ПЛК, а именно: способ управления, тип входа, категории ПЛК, область применения.

По способу управления ПЛК делятся на две группы: централизованные и распределительные. При централизованном управлении все модули находятся в одном месте, при распределительном входные модули находятся в одном месте, а выходные в другом.

В программируемых логических контроллерах существуют три основных вида входов: дискретные, аналоговые и специальные. Дискретные входы используют для получения информации о состоянии датчиков/устройств. Аналоговые входы используют для получения информации о непрерывно изменяющихся физических величинах. Специальные входы используют для передачи информации об аварийных сигналах, сигналах от других ПЛК или внешних устройств [1].

Также существует три типа ПЛК по конструктивному исполнению: модульные, моноблочные и распределительные. Модульные – набор модулей, где каждый модуль работает с определенным типом задач. Моноблочные – моноблок, который оснащается заранее определенным количеством входов/выходов. Распределительные – несколько модулей, которые расположены отдельно друг от друга по всему контроллеру [2].

Области применения ПЛК: робототехника, автоматизация домашних устройств, промышленная автоматизация и др.

Вывод: принято решение использовать ПЛК Siemens, так как его основные функции включают: аналоговые и дискретные входы/выходы; коммуникационные возможности, а именно обмен данными с другими системами и устройствами; возможность программирования и позиционирования; управление безопасностью.

Достоинства и недостатки ПЛК

Достоинства ПЛК:

- гибкость программирования: быстрое дополнение или изменение программы управления;
- надежность;
- простота установки и эксплуатации;
- использование стандартных языков программирования;
- возможность удаленного доступа и мониторинга [3].

Недостатки ПЛК:

- высокая стоимость;
- ограниченная производительность: могут иметь ограничения по скорости и объему обрабатываемых данных;
- ограничения интеграции.

Управление водосборником. С помощью программирования ПЛК в LOGO!8 можно создавать схемы управления, а также автоматизировать работу устройств и механизмов. Таким образом, с использованием ПЛК в программе LOGO!8 можно решать задачи, связанные с автоматизацией процессов. Рассмотрим задачу «Управление водосборником» (рис. 1).

При падении уровня воды ниже I1, должен автоматически включиться Q1 (подача питьевой воды), а при уровне воды выше I2 должно произойти выключение.

Добавление функций. Вставляем все необходимые постоянные в редактор схем. Далее нам потребуются базовые функции. В данном примере используется логическая функция НЕ. Альтернативой функции НЕ может стать инверсия на реле. Выполним команду «Инверсия соединителя» [4].

В данном случае реле с блокировкой открывает подачу питьевой воды и удерживает вентиль открытым, пока уровень воды не достигнет верхнего поплавка I2 (рис. 2).

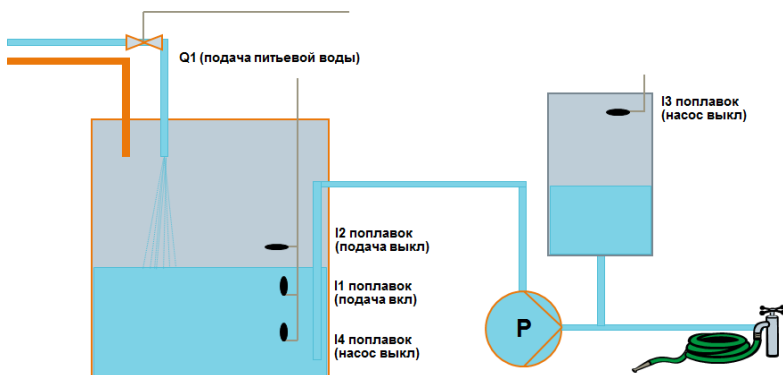


Рис. 1. Задача «Управление водосборником»

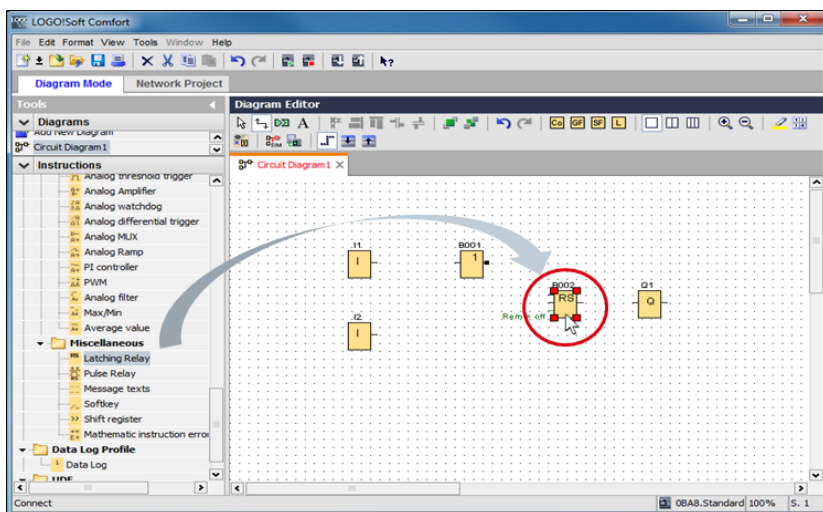


Рис. 2. Добавление реле с блокировкой

Соединение блоков. Теперь соединяем функциональные блоки между собой (рис. 3).

Дополнительные функции. LOGO! Soft Comfort предлагает нам воспользоваться такими функциями, как:

1. Добавление текстовых полей и их форматирование.
2. Перемещение элементов.

Для улучшения вида программы можно поменять расположение блоков, линий и текста:

1. Выравнивание элементов.
2. Изменение параметров блока [4].

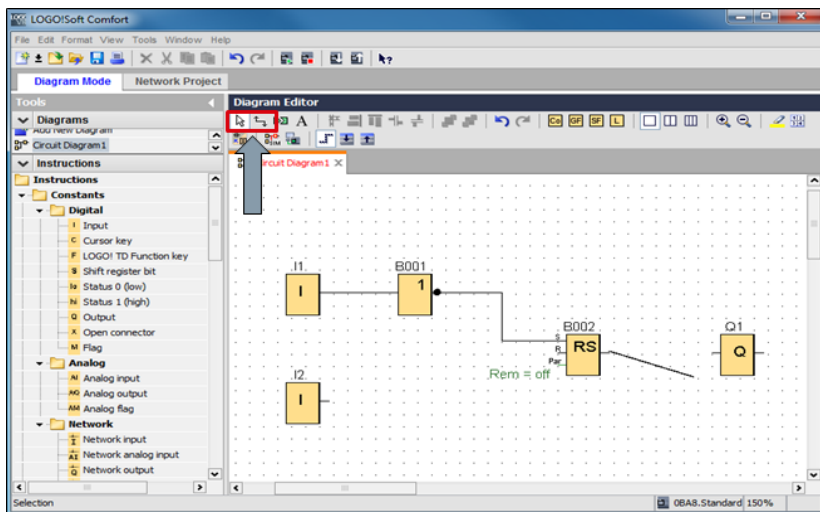


Рис. 3. Создание соединительной линии

Тестирование программы. Выбираем офлайн-симуляцию для тестирования нашей программы, так как данный инструмент позволяет тестировать программу, не используя устройство LOGO!. Если же устройство LOGO! подключено к ПК, выполняем онлайн-тестирование.

В режиме офлайн-симуляции на панели инструментов можно изменить сигналы. Соединительные линии могут быть двух цветов: синий означает низкий уровень сигнала, а красный – высокий уровень.

Дополнительно можно изменить режим для каждого отдельного входа, установить диапазон значений.

Запуск программы LOGO!. Передаем программу в LOGO! и переходим в режим RUN, тем самым тестируем программу в реальных условиях (рис. 4).

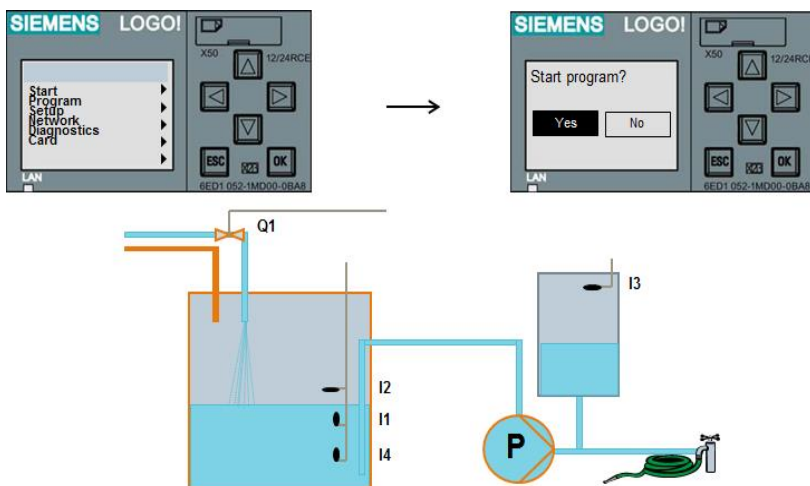


Рис. 4. Запуск программы

В результате получаем типовую лабораторную работу по исследованию систем автоматического управления и регулирования, которая поможет понять принцип работы LOGO!.

Библиографический список

1. Промышленная автоматизация [Электронный ресурс]. – URL: <https://industrialization.ru/kontrollery-oblast-primeneniya/?ysclid=lwi91dahq1124847224> (дата обращения: 22.05.2024).
2. Типы ПЛК [Электронный ресурс]. – URL: [Types of PLC \(finestart.school\)](https://finestart.school/) (дата обращения: 23.05.2024).
3. ПЛК: принцип работы [Электронный ресурс]. – URL: <https://techtrends.ru/resources/articles/programmiruemye-logicheskie-kontrollery/?ysclid=lwi7za7wjf600238209> (дата обращения: 22.05.2024).
4. Руководство по LOGO!8 [Электронный ресурс]. – URL: https://www.is-com.ru/files/logo_V6_r.pdf (дата обращения: 22.05.2024).

Сведения об авторах

Тренина Дарья Олеговна – бакалавр кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, гр. КС-20-1б, e-mail: dasha.tronina17@gmail.ru

Келлер Степан Ильич – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-1б, г. Пермь, e-mail: stepan.keller@gmail.com

Камянецкий Сергей Александрович – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-1б, г. Пермь, e-mail: kamyanetskysergei@yandex.ru

Погудин Андрей Леонидович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pogudin_al@mail.ru

А.Б. Федоров, Н.С. Хлопин

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА АЛГОРИТМОВ ДЛЯ ТЕСТЕРА ВИТОЙ ПАРЫ НА БАЗЕ ARDUINO

В данной статье рассмотрены алгоритмы для тестера витой пары на основе микроконтроллерной платы Arduino.

Ключевые слова: тестер, витая пара, алгоритм.

A.B. Fedorov, N.S. Khlopin

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF ALGORITHMS FOR THE ARDUINO-BASED TWISTED PAIR TESTER

This article discusses algorithms for a twisted pair tester based on an Arduino microcontroller board.

Keywords: tester, twisted pair, algorithm.

В современном мире сетевые технологии играют важную роль в обеспечении связи и передачи данных [1]. Одним из ключевых компонентов сетевой инфраструктуры является кабельная система, в частности, витая пара. Надежность и правильность подключения кабелей витой пары критически важны для обеспечения стабильной работы сетей. Поэтому разработка эффективных методов тестирования витой пары имеет большое значение.

Постановка задачи. Необходимо разработать алгоритмы для тестера витой пары на базе Arduino: тестирование каждого из проводников на обрыв или замыкание с другими; определение превышения длины проводника; проверка провода соответствию стандартам обжима витой пары, сопоставление 1-й и 2-й стороны.

Реализация. Для реализации алгоритмов был выбран тестер витой пары на базе Arduino, ниже представлена общая схема тестера, изображенная на рис. 1.

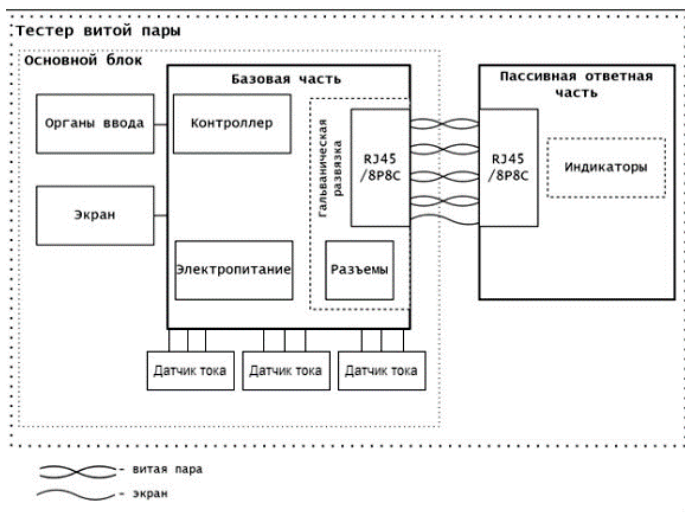


Рис. 1. Общая схема тестера витой пары

На ней изображены основные компоненты, входящие в тестер витой пары. В основной блок входит базовая часть с контроллером, электропитанием, разъемами, органами управления, экраном и 3 датчиками тока. Пассивная ответная часть состоит из разъема и набора компонентов, индикаторов, необходимых для отладки, тестирования и визуального контроля работы прототипа тестера. Между базовой и пассивной частью располагается кабель связи «витая пара», состоящий из 8 жил и экрана.

Переходим к разработке алгоритмов для данного тестера витой пары на Arduino.

1) алгоритм тестирования кабеля «витая пара» на обрыв и короткое замыкание представлен ниже на рис. 2.

Начало: Начало алгоритма

Задать массив из 9 пинов:

Инициализация массива, содержащего 9 пинов.

Для пинов out ($i = 1; i < 10; i++$):

Цикл, который проходит через все пины, назначенные как выходные (out).

Для пинов in ($j = 1; j < 10; j++$):

Вложенный цикл, который проходит через все пины, назначенные как входные (in).

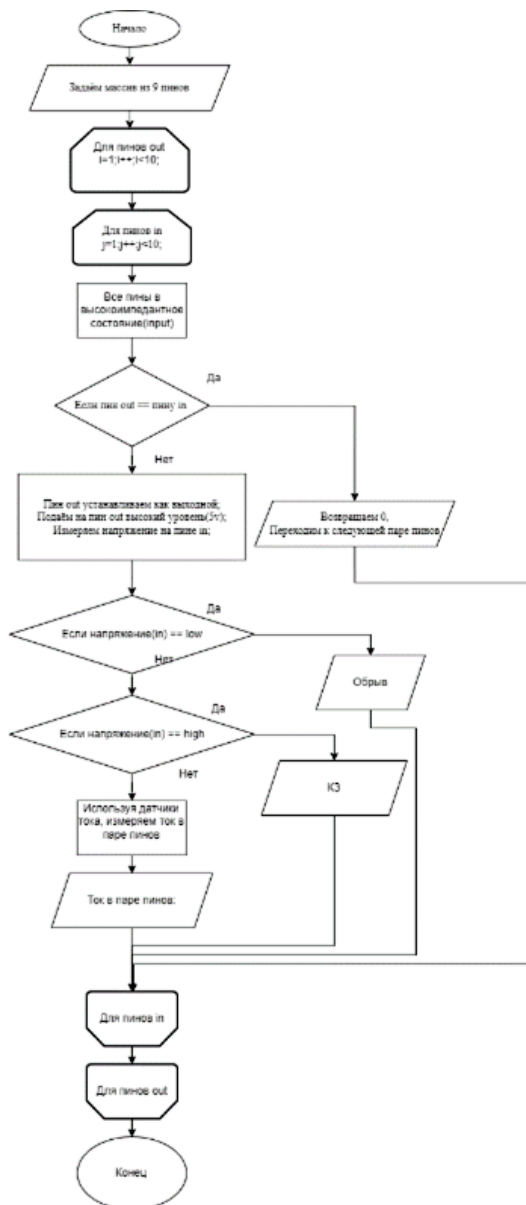


Рис. 2. Алгоритм тестирования кабеля «витая пара» на обрыв и короткое замыкание

Все пины в высокоимпедансное состояние (input):

Установка всех пинов в состояние высокоимпедансного входа (input).

Если пин out == пину in: Проверка, совпадают ли текущие пины out и in.

Если совпадают, переходим к следующей паре пинов.

Если не совпадают, продолжаем выполнение алгоритма.

Пин out устанавливаем как выходной:

Устанавливаем текущий пин out как выходной.

Поднимаем на пин out высокий уровень (5V).

Измеряем напряжение на пине in.

Если напряжение(in) == low:

Проверка, является ли напряжение на пине in низким.

Если да, выводим сообщение "Обрыв".

Если нет, продолжаем выполнение алгоритма.

Если напряжение(in) == high:

Проверка, является ли напряжение на пине in высоким.

Если да, выводим сообщение "КЗ".

Если нет, продолжаем выполнение алгоритма.

Ток в паре пинов:

Используем датчик тока для измерения тока между пинами out и in.

Выводим измеренный ток.

Для пинов in: Переходим к следующему пину in.

Для пинов out: Переходим к следующему пину out.

Конец: Завершение алгоритма

Этот алгоритм предназначен для диагностики кабеля витая пара, позволяя выявить проблемы с обрывами или короткими замыканиями.

2) Алгоритм определения превышения длины проводника, представлен ниже на рис. 3.

Начало:**Задаем массив из 9 пинов:**

Создаем массив из 9 пинов для тестирования.

Задаем таблицу для эталонных значений сопротивления, полученных при тестировании (например, для кабеля длиной 1, 100, 150, 305 м).

Устанавливаем все пины в входной режим:

Переводим все пины в состояние входа.

Цикл по пинам от 1 до 10:

Проходим по каждому пину в массиве.

Устанавливаем текущий пин как выходной:

Переводим текущий пин в состояние выхода.

Поднимаем на пине высокий уровень (5V).
 Измеряем ток и напряжение, используя датчик тока и напряжения.
 Рассчитываем сопротивление проводника.
 Максимальное сопротивление пина = 0;
 Пин с максимальным сопротивлением = 0.

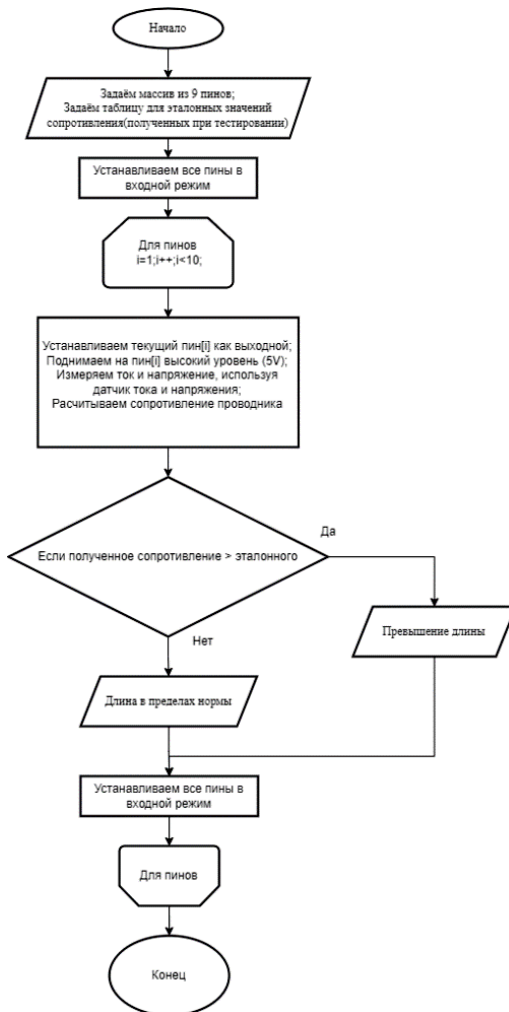


Рис. 3. Алгоритм определения превышения длины проводника
Проверка с максимальным сопротивлением:

Если полученное сопротивление больше максимального сопротивления, то присваиваем значение сопротивления максимальному сопротивлению, а пину с максимальным сопротивлением присваиваем значение i .

Проверка сопротивления:

Сравниваем полученное сопротивление с эталонным значением (если ввели больше одного эталонного значения, то сравнений будет больше).

Если сопротивление больше эталонного:

Сообщаем о превышении длины проводника.

Если сопротивление в пределах нормы:

Сообщаем, что длина проводника в пределах нормы.

Устанавливаем все пины в входной режим:

Переводим все пины обратно в состояние входа.

Конец

3) Алгоритм проверки провода соответствию стандартам обжима витой пары, сопоставление 1-й и 2-й стороны.

Для данного алгоритма потребуется пассивная ответная часть с перемычками между парами (1-2 3-4 5-6 7-8).

Пример алгоритма представлен ниже на рис. 4.

Начало: Запуск алгоритма.

Инициализация массивов: Создаются два массива:

- Массив пинов на плате.
- Массив для хранения значений пинов.
- Установка всех пинов в выходной режим с низким уровнем (0V): Все пины микроконтроллера устанавливаются в режим выхода и на них устанавливается низкий уровень (0V).

Цикл для пинов i от 1 до 8:

- Устанавливаем текущий пин i как выходной и поднимаем на нем высокий уровень (5V).
- Вложенный цикл для пинов j от 1 до 8:
- Настраиваем остальные пины как входы.
- Если пин[i] не равен пин[j], считываем значение с текущего пина j и сохраняем его в массив для хранения значений (high или low).
- Возвращаем текущий пин в режим входа с низким уровнем (0V).

Цикл для пинов k от 1 до 8:

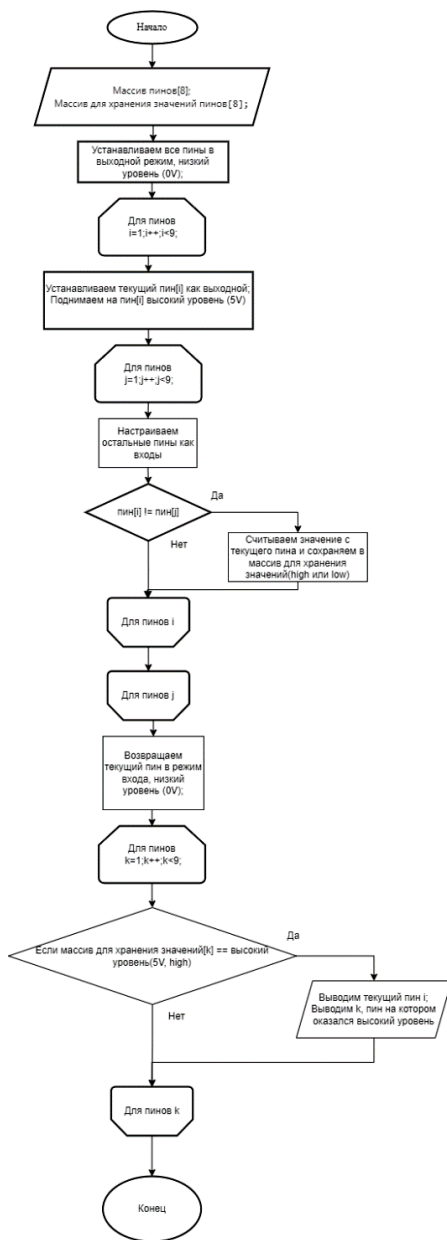


Рис. 4. Алгоритм для определения обжима кабеля

Проверяем, если значение в массиве для хранения значений для текущего пина k равно высокому уровню (5V):

- Выводим текущий пин i .
- Выводим пин k , на котором оказался высокий уровень.

Конец: Завершение алгоритма.

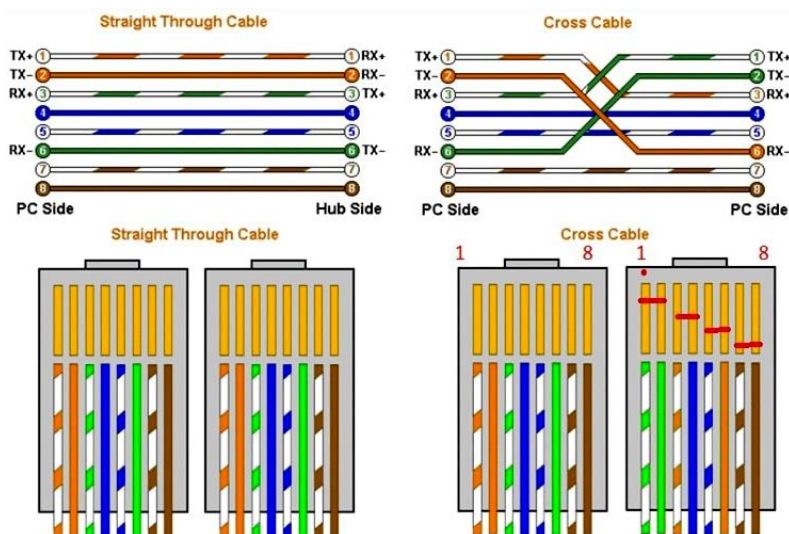


Рис. 5. Прямой кабель и кросс-кабель

На правом джеке пример, как ставятся перемычки.

Например, если будет прямой кабель, то вывод будет такой:

1 2 3 4 5 6 7 8

2 1 4 3 6 5 8 7

Из этого делаем вывод, что кабель прямой, потому что у нас стоит заглушка с перемычками. Т.е., если мы подадим высокое напряжение в первый пин, то выйдет во 2 и т.д.

Аналогично происходит и с кросс-кабелем. Вывод будет такой:

1 2 3 4 5 6 7 8

4 5 6 1 2 3 8 7

Высокое напряжение с первого пина идет на четвертый и т.д.

Заключение. В данной статье были рассмотрены алгоритмы и методы тестирования витой пары на базе Arduino. Предложенные алгоритмы позволяют эффективно выявлять обрывы и замыкания провод-

ников, определять превышение длины кабеля, проверять соответствие стандартам обжима. Применение разработанных алгоритмов и методов обеспечит надежность и правильность работы кабельных систем в сетевых инфраструктурах.

Библиографический список

1. Таненбаум Э., Уэзеролл Д. Компьютерные сети. – 5-е изд. – СПб.: Питер, 2012. – 960 с.

Сведения об авторах

Федоров Андрей Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, e-mail: pstufab@hotmail.com

Хлопин Никита Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: nikita.xlopin@yandex.ru

А.Р. Исмагилов, Е.В. Долгова

Пермский национальный исследовательский
политехнический университет, Пермь

МОДЕЛИРОВАНИЕ ТРОСОВ ДЛЯ ТРЕНАЖЕРА ПОРТАЛЬНОГО КРАНА В СРЕДЕ UNREAL ENGINE 5

В этой статье исследуется возможность моделирования тросов для тренажера портального крана в среде Unreal Engine 5. Рассмотрены компоненты «Physics Constraint» и «Cable Component» среды Unreal Engine 5, с помощью которых возможно моделирование тросов. Реализован трос с применением компонентов «Physics Constraint» и «Cable Component».

Ключевые слова: Unreal Engine, Physic Constraint, трос, Cable Component, кран.

A.R. Ismagilov, E.V. Dolgova

Perm National Research Polytechnic University, Perm

SIMULATION OF CABLES FOR A PORTAL CRANE SIMULATOR IN THE UNREAL ENGINE 5 ENVIRONMENT

This article explores the possibility of modeling cables for a portal crane simulator in the Unreal Engine 5 environment. The «Physics Constraint» and «Cable Component» components of the Unreal Engine 5 environment with which cable modeling is possible are considered. A cable was implemented using the «Physics Constraint» and «Cable Component» components.

Keywords: Unreal Engine, Physic Constraint, cable, Cable Component, crane.

Обучение крановщиков на настоящем рабочем портальном кране сопряжено с рядом трудностей и опасностей. Из-за недостатка опыта обучающегося крановщика во время обучения не исключены всевозможные аварии или различные поломки. Поэтому прежде, чем допускать обучающегося к настоящему оборудованию, необходимо сформировать у него начальные навыки работы, с чем могут помочь компьютерные тренажеры. Современный игровой движок Unreal Engine 5 предоставляет уникальную возможность в создании реалистичного тренажера портального крана. Создание реалистичного тренажера, имитирующего поведение крана, требует точной физико-математической модели, которая будет учитывать нюансы его работы.

Одной из важнейших частей портального крана являются тросы, обеспечивающие связь между краном и грузом. В перегрузочном процессе используются 4 троса. Моделирование троса может вызвать наибольшую сложность, но применение встроенных компонентов игрового движка Unreal Engine 5 может значительно упростить эту задачу. Одним из вариантов реализации троса является применение вместе компонентов «cable component» и «Physic Constraint».

Для моделирования физики троса можно использовать «Physic Constraint» – компонент, позволяющий соединить два твердых тела. Такое соединение позволяет создать объект, раскачивающийся от воздействия внешних сил (рис. 1). Для создания троса требуются два твердых тела, которые будут являться концами троса. Затем с помощью «Physic Constraint» можно соединить эти тела, задав ограничения движения, жесткость и демпфирование.

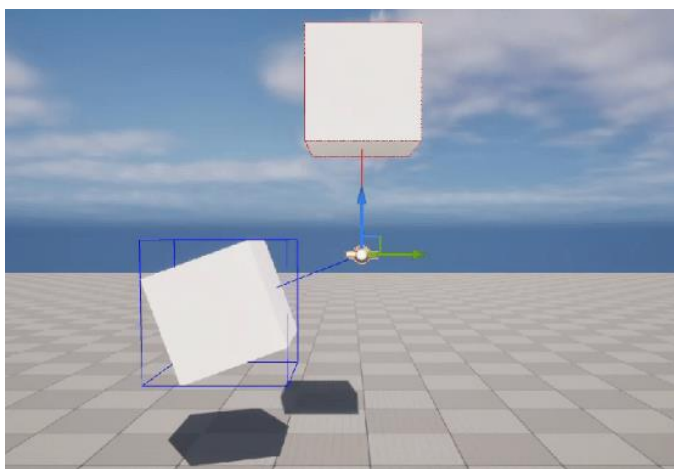


Рис. 1. Пример physics constraint

При корректной настройке «Physic Constraint» будет казаться, что объекты соединены между собой невидимым тросом.

Для визуализации тросов в Unreal Engine 5 применяется «Cable Component». «Cable Component» использует для моделирования тросов метод интегрирования Верле, часто применяемый в разработке игр [1]. Метод интегрирования Верле – численный метод, используемый для интегрирования уравнений движения материальной точки [2]. Данный метод используется для вычисления следующего положения точки без

использования скорости, а только по текущему и прошлому положению точки.

$$\vec{x}(t + \Delta t) = 2\vec{x}(t) - \vec{x}(t - \Delta t) + \vec{a}(t)\Delta t^2 + O(\Delta t^4), \quad (1)$$

где $\vec{x}$ – радиус-вектор; $\vec{a}$ – ускорение; $O(\Delta t^4)$ – ошибка относительно Δt^4 .

Трос в Unreal Engine 5 моделируется как цепочка частиц с определенными ограничениями по расстоянию между ними. Частицы на концах троса фиксируются и движутся вместе с объектами, к которым они прикреплены. В то же время частицы, находящиеся посередине, свободны и подвержены воздействию силы тяжести, что придает эффект реалистичного падения и деформации троса (рис. 2).

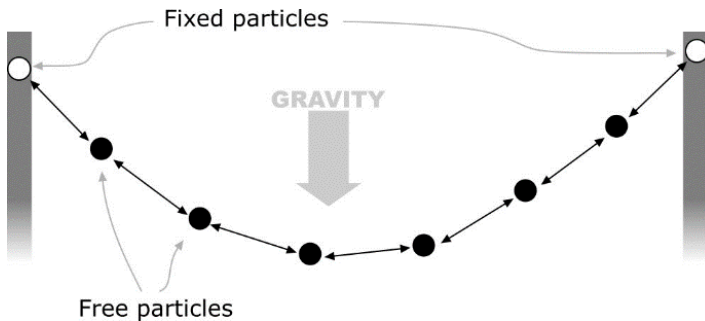


Рис. 2. Схематичное представление троса

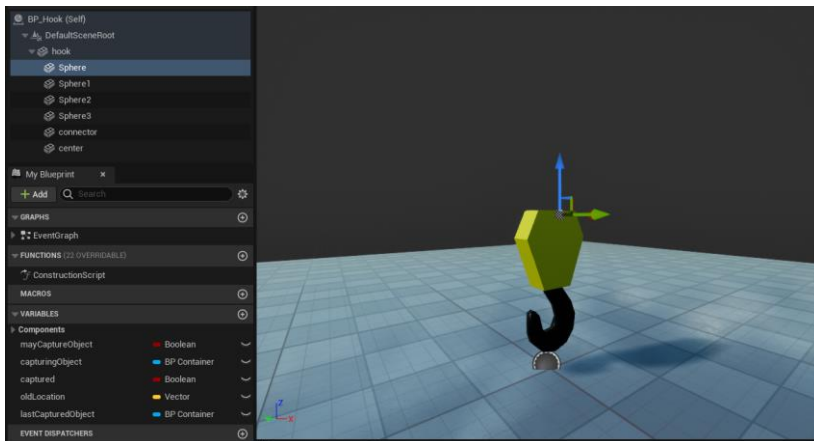


Рис. 3. Blueprint крюковой подвески

У компонента «Cable Component» можно легко настраивать параметры, такие как длина, толщина, жесткость, количество сегментов и другие. Таким образом, для реализации тросов крана необходимо задать точки для подвеса. Точки начала тросов будут расположены на «хоботе» крана, а точки конца на крюковой подвеске. Класс крюковой подвески с иерархией компонентов представлен на рис. 3.

Компоненты «Sphere» представляют из себя точки концов тросов, такие же точки расположены на «хоботе» крана.

Задав одинаковые точки начала и конца компонентам «Physics Constraint» и «Cable Component», в результате получаем трос, способный взаимодействовать с окружающей средой. Такие тросы можно применять не только для соединения крюковой подвески с «хоботом» крана, но и для строповки грузов (рис. 4).

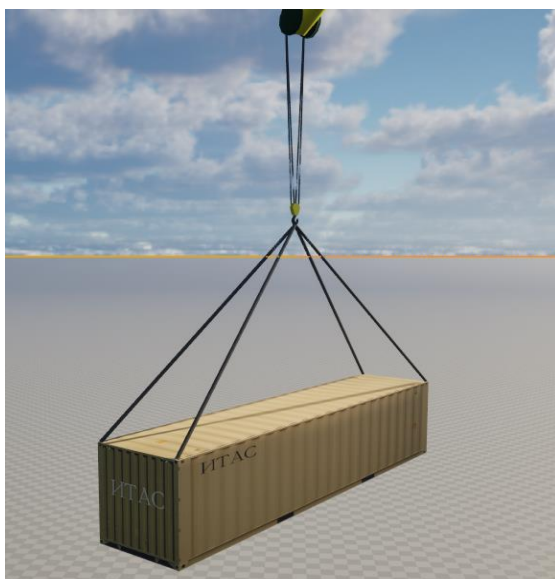


Рис. 4. Итоговый результат

Таким образом, игровой движок Unreal Engine 5 предоставляет удобный и мощный функционал для моделирования тросов. Комбинируя компоненты «Physics Constraint» и «Cable Component», разработчики могут создавать системы тросов различной сложности и степени реализма для различных игровых проектов и симуляций.

Библиографический список

1. Unreal Engine 5 Documentation [Электронный ресурс]. – URL <https://docs.unrealengine.com/5.0/en-US>
2. Verlet L. Computer "experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules // Physical review. – 1967. – 98 p.

Сведения об авторах

Исмагилов Айназ Рамзитович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: ainazismagilov59@gmail.com

Долгова Елена Владимировна – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: elena@dolgova.space

А.В. Лыкова, А.Б. Федоров

Пермский национальный исследовательский
политехнический университет, Пермь

**ИССЛЕДОВАНИЕ И ТЕСТИРОВАНИЕ СИСТЕМЫ
РЫЧАГОВ НА ДЖОЙСТИКАХ ДЛЯ ВИРТУАЛЬНОГО
ТРЕНАЖЕРА ОПЕРАТОРА ПОРТАЛЬНОГО КРАНА**

Рассматривается аппаратная часть джойстиков для тренажера оператора портального крана, которые разрабатываются на кафедре Информационных технологий и автоматизированных систем. Исследуется механизм работы рычагов для последующего создания программного обеспечения.

Ключевые слова: джойстики, рычаги, виртуальный тренажер.

A.V. Lykova, A.B. Fedorov

Perm National Research Polytechnic University, Perm

**RESEARCH AND TESTING OF A SYSTEM
OF LEVERS ON JOYSTICKS FOR A VIRTUAL
SIMULATOR FOR A PORTAL CRANE OPERATOR**

This article discusses the hardware of joysticks for the gantry crane operator simulator, which are being developed at the ITAS department. The mechanism of operation of levers for the subsequent creation of software is investigated.

Keywords: joysticks, levers, virtual simulator.

Эффективность функционирования современных технологических процессов во многом определяется качеством подготовки операторов [1–3]. Проблема в их обучении, поскольку неопытному специалисту опасно и неэкономично доверять управление реальным грузоподъемным краном. Во-первых, это может привести к несчастному случаю, порче груза или аварии, а во-вторых, неэффективное управление сулит преждевременному износу деталей машины и расходованию рабочего времени.

Использование обучающих тренажерных комплексов поможет данную проблему решить, и немаловажной частью такого комплекса является программно-аппаратная система связи джойстиков с компьютерным тренажером.

Разрабатываемые джойстики предназначены для управления портальным краном в виртуальном тренажере, запущенном на персональном компьютере. К компьютеру они подключаются посредством проводного интерфейса USB, эмулируя HID-клавиатуру, т.е. рычаги и кнопки на пультах соответствуют клавишам на клавиатуре. Внешний вид джойстиков представлен на рис. 1.



Рис. 1. Внешний вид джойстиков

На прошедших этапах разработки джойстиков уже были созданы органы управления: рычаги, кнопки, педали, соответствующие реальным пультам оператора портального крана. В связи с этим на текущем этапе появились новые задачи: разработка печатных плат контроллеров для пультов и написание ПО с функцией настройки кнопок органов управления пульта для возможности эмулировать клавиатуру, как пожелает пользователь, с последующей прошивкой микроконтроллера.

На рис. 2 представлена UML-диаграмма взаимодействия пользователя с джойстиками и виртуальным тренажером.

Управление будет осуществляться с помощью микроконтроллера ATmegaU4-ми, который будет запитываться напрямую от подключения к компьютеру по USB.

Разработка новых решений подразумевает проведение исследовательской деятельности для изучения текущих решений. Для написания ПО нижнего уровня для прошивки микроконтроллеров необходимо рассмотреть, как устроены органы управления джойстиков, в первую очередь это касается рычагов.

Одним из основных элементов рычага является кодовый диск с расположенным на нем так называемым кодовым рисунком, состоящим из набора отверстий. Он установлен между ИК-излучателем и приемным оптическим элементом (ИК-датчиком). При переключении рычага отверстия могут перекрываться, препятствуя прохождению сигнала.

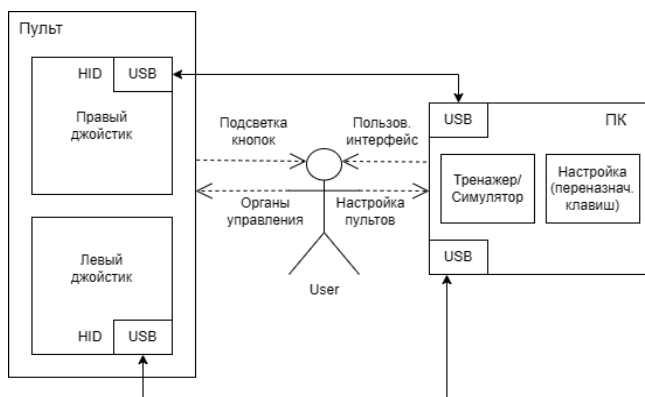


Рис. 2. UML-диаграмма взаимодействия пользователя с джойстиками и виртуальным тренажером

Схему кодового диска рычагов джойстиков можно увидеть на рис. 3. Начальное положение рычага – по центру, все отверстия открыты и не препятствуют световому потоку, код состояния – 0. После смены положения рычага некоторые отверстия закрываются.

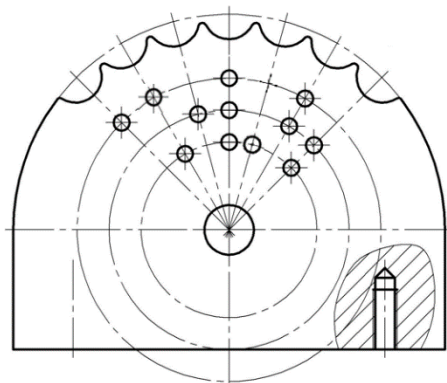


Рис. 3. Кодовый диск рычага

Считывание контроллером положения рычага происходит следующим образом: направленный световой поток, излучаемый световым элементом (ИК-светодиодом), через отверстия диска падает на специальную панель оптического датчика и далее попадает на приемный фотозлемент, который создает пропорциональный светосиле сигнал и отправляет на контроллер Arduino через транзисторный усилитель. Отсутствие света на фотодатчике соответствует логической единице, а его наличие – логическому нулю. Печатные платы блока оптических датчиков продольной и поперечной оси рычагов джойстиков показаны на рис. 4 и 5.

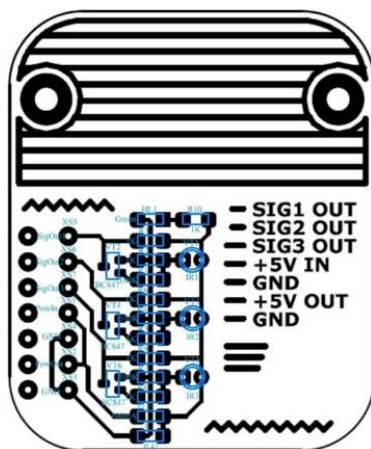


Рис. 4. Печатная плата блока оптических датчиков продольной оси рычагов

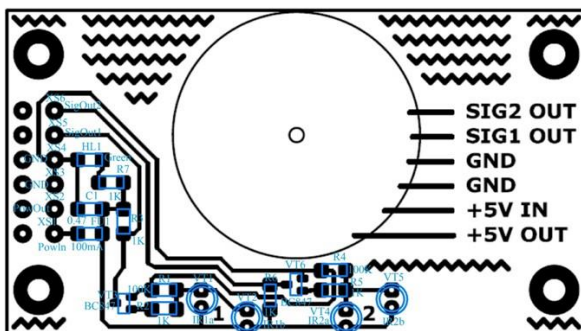


Рис. 5. Печатная плата блока оптических датчиков поперечной оси рычага левого джойстика

Таким образом, рычаги кодируются в зависимости от состояния света на фотодатчике. Кодовые отверстия диска, установленного на рычагах, показаны на рис. 6 и 7. Кодировка состояний рычага соответствует двоичному коду.



Рис. 6. Кодировка состояний для датчиков продольной оси рычага

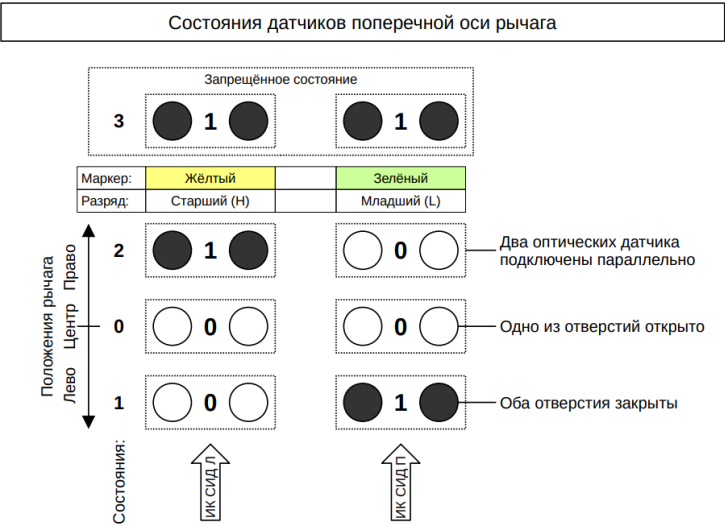


Рис. 7. Кодировка состояний для датчиков поперечной оси рычага

Тестовая программа, позволяющая проверить работу рычагов, была разработана на основе следующего алгоритма:

1. Происходит считывание состояния рычага, используя три (или два для поперечной оси) цифровых входа для старшего, среднего и младшего битов в переменные hBit, mBit и lBit соответственно. По умолчанию чтение состояния выполняется каждые 10 миллисекунд.

2. В случае, если биты изменяются, они сохраняются в массиве, а текущее состояние рычага присваивается переменной state, иначе используется предыдущее состояние prevState.

Существует проблема использования естественных двоичных кодов для подобных рычагам или энкодерам устройств, она связана с тем, что физические переключатели не работают идеально: существует риск недостаточной синхронизации изменения состояния.

Если рассмотреть схему на рис. 4, можно заметить, что при переходе между состояниями «0» и «5» или «0» и «6» продольной оси рычага два переключателя сразу меняют свое состояние, а между состояниями «2» и «5» изменяются все три переключателя.

В течение короткого периода, пока они меняются, переключатели будут считывать некоторое ложное положение, и переход состояний может выглядеть следующим образом 101 — 001 — 011 — 010. Может казаться, что переключатели находятся в нужном положении «101», но наблюдатель не может сказать, является ли это реальной позицией «1» или переходным состоянием между двумя другими.

Эта проблема может быть решена путем изменения одного бита за раз, чтобы исключить любую неоднозначность в положении. В результате каждому состоянию будет присвоено уникальное кодовое значение таким образом, что ни одно из них не будет одинаковым и каждые два соседних будут отличаться ровно на один бит, что позволит избежать возможности неправильного считывания данных датчика из-за того, что некоторые биты изменяются раньше других.

Для этого используется код Грея, который также известен как монотрофический, или синкопический, код, с единичным расстоянием Хэмминга между соседними кодами (таблица).

Таблица кода Грея

Десятичное число	Двоичное число	Код Грея
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

Формула преобразования двоичного кода в код Грея:

$$G_i = B_i \oplus B_{i+1},$$

где $\oplus$ – исключающее ИЛИ (если биты равны, то результат – «0»; если нет, то «1»).

Такое представление данных отличается от стандартного двоичного кода тем, что обеспечивает непрерывность бинарных комбинаций и полностью исключает появление паразитных кодов.

Таким образом, в будущем для повышения надежности считывания сигналов при сменах положения рычага рекомендуется использовать код Грея для создания кодового диска.

Библиографический список

1. Файзрахманов Р.А., Федоров А.Б. Разработка аппаратной части тренажера оператора портального крана, 2010.
2. Файзрахманов Р.А., Старосельцев В.Б. Разработка компьютерного тренажера для подготовки крановщиков.
3. Коды Грея [Электронный ресурс] // neerc.ifmo. – URL: https://neerc.ifmo.ru/wiki/index.php?title=Коды_Грея (дата обращения: 25.05.2024).

Сведения об авторах

Лыкова Анна Викторовна – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-1б, г. Пермь, e-mail: lavlav.pstu@gmail.com

Федоров Андрей Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: pstufab@gmail.com

О.Л. Викентьева, А.Д. Брейкин

Пермский национальный исследовательский
политехнический университет, Пермь

ИСПОЛЬЗОВАНИЕ СИСТЕМЫ МЕЖПРОЦЕССНОГО ВЗАИМОДЕЙСТВИЯ ДЛЯ ЗАПУСКА ПРИЛОЖЕНИЙ В ИЗОЛИРОВАННОЙ СРЕДЕ

Описывается способ решения проблемы отсутствия возможности одновременной работы приложений на ОС Linux с использованием ПО Wine, системы управления приложениями Flatpak и системы межпроцессного взаимодействия D-Bus. Представлен алгоритм запуска приложений в одной изолированной среде выполнения Flatpak. Приведена программная реализация алгоритма.

Ключевые слова: операционная система, Linux, Wine, Flatpak, сервис, среда выполнения.

O.L. Vikenteva, A.D. Breikin

Perm National Research Polytechnic University, Perm

USE OF INTERPROCESS COMMUNICATION SYSTEM TO RUN APPLICATIONS IN AN ISOLATED ENVIRONMENT

The method of solving the problem of lack of simultaneous operation of applications on Linux OS using Wine software, Flatpak application management system and D-Bus interprocess communication system is described. An algorithm for running applications in one isolated Flatpak execution environment is presented. The program implementation of the algorithm is given.

Keywords: operating system, Linux, Wine, Flatpak, service, runtime environment.

В связи с переходом предприятий на использование отечественных дистрибутивов ОС Linux современное ПО должно поддерживать работу как на ОС Windows, так и на ОС Linux. Из-за особенностей технической реализации часть отечественного ПО полностью совместима только с ОС Windows. Одним из способов решения описанной проблемы является разработка нового ПО или адаптирование существующего, но данный процесс может быть время- и трудозатратным.

Альтернативным вариантом реализации возможности запуска программ, разработанных исключительно для ОС Windows, на ОС Linux является использование ПО Wine. Принцип работы ПО Wine заключается в трансляции системных вызовов ОС Windows в аналогичные системные вызовы ОС Linux без потери производительности [1].

Одним из способов распространения и развертывания приложений, работающих с использованием ПО Wine, является система управления приложениями Flatpak [2]. Система предоставляет пространство, которое включает в себя набор основных библиотек и сервисов ОС Linux, с возможностью интеграции необходимой версии ПО Wine. За счет этого достигается идентичное поведение приложения на всех дистрибутивах ОС Linux. Для запуска приложений используется непривилегированная изолированная среда выполнения, которая создается при помощи библиотеки bubblewrap [3]. При использовании Flatpak для запуска приложений, работающих через ПО Wine, возможно выполнение только одного экземпляра приложения, так как каждый экземпляр выполняется в отдельной изолированной среде. В случае запуска нескольких экземпляров приложения происходят ошибки синхронизации состояний сервера Wine, поскольку несколько серверов используют одну директорию с набором файлов и каталогов, повторяющих структуру ОС Windows.

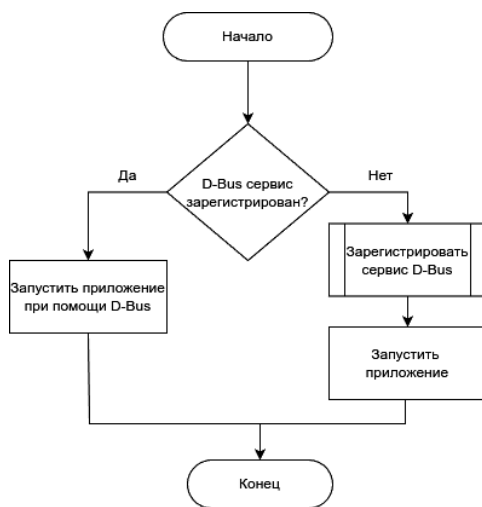


Рис. 1. Блок-схема алгоритма запуска приложения с использованием D-Bus

Для решения описанной проблемы необходимо добиться наличия общего сервера Wine. Этого можно достичь путем перенаправления команд запуска приложений в одну изолированную среду при помощи системы межпроцессного взаимодействия Desktop Bus (D-Bus) [4].

Разработан алгоритм запуска приложения с использованием сервиса D-Bus. При запуске изолированной среды выполняется проверка наличия сервиса в сессии пользователя. Если сервис зарегистрирован, то отправляется сообщение на запуск приложения. В противном случае в текущей изолированной среде выполнения регистрируется сервис D-Bus и запускается приложение. Блок-схема описанного алгоритма представлена на рис. 1. Разработанный алгоритм позволяет запускать более одного экземпляра приложения за счет перенаправления команд запуска в единственную изолированную среду выполнения. На рис. 2 изображена диаграмма последовательностей для запуска двух приложений, выполняющихся с использованием ПО Wine, в изолированной среде выполнения Flatpak.

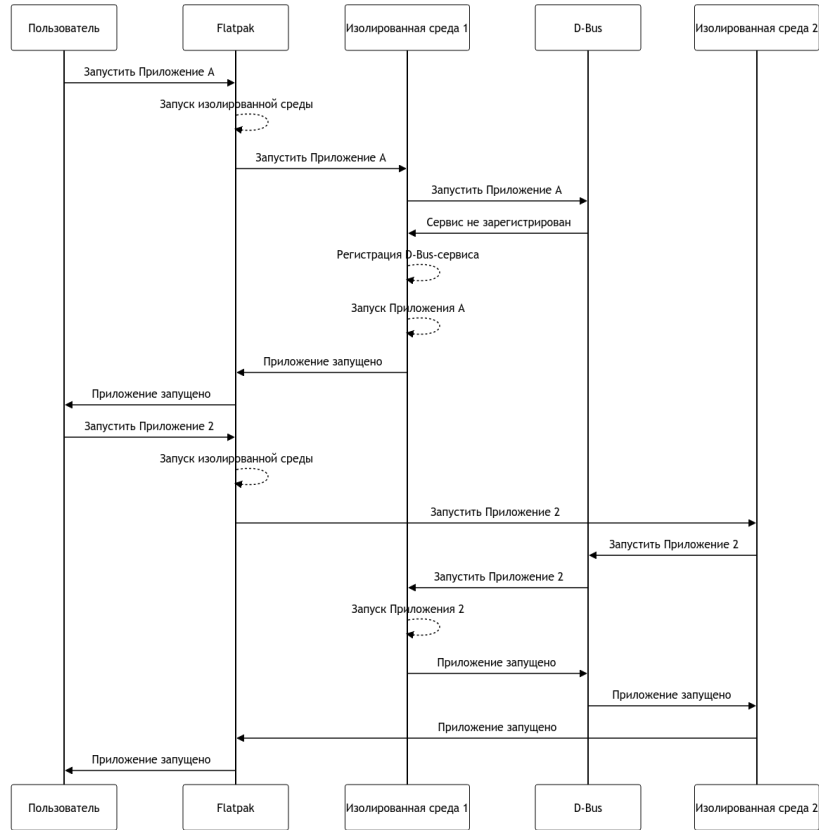


Рис. 2. Диаграмма последовательностей для запуска нескольких приложений в одной изолированной среде Flatpak

Сервис D-Bus разработан с использованием языка программирования Python и библиотек GLib и Gtk [5]. В классе LauncherService реализовано подключение к пользовательской сессии D-Bus и регистрация сервиса, включающего в себя метод для запуска приложений через ПО Wine. На листинге 1 представлен класс LauncherService.

Листинг 1 – Сервис для запуска приложений

```
class LauncherService(dbus.service.Object):
    def __init__(self):
        # Подключение к пользовательской сессии
        bus_name = dbus.service.BusName(SERVICE_NAME,
        bus=dbus.SessionBus())
        # Регистрация сервиса в сессии
        dbus.service.Object.__init__(self, bus_name,
        ОБЪЕКТ_PATH)

        # Метод для запуска приложений
        @dbus.service.method(SERVICE_NAME,
        in_signature='ss', out_signature='')
        def Run(self, appPath, arguments):
            subprocess.Popen([wine, f"{appPath} {argu-
            ments}"])
```

Вызов методов сервиса D-Bus производится при помощи программы gdbus для взаимодействия с сервером D-Bus. На листинге 2 проиллюстрирован пример запуска приложения «Блокнот» с использованием ПО Wine и сервиса D-Bus.

Листинг 2 – Запуск приложения «Блокнот» с использованием D-Bus
gdbus

```
# Команда вызова метода
call
# Подключение к пользовательской сессии
--session
# Название сервиса-получателя
--destination "dev.salaron.wine"
# Путь до объекта сервиса-получателя
--object-path "/dev/salaron/wine"
# Вызываемый метод
--method "dev.salaron.wine.Run"
# Аргумент метода – приложение для запуска
```

```
notepad.exe  
# Аргумент метода – аргументы приложения  
~/file.txt
```

Таким образом, разработаны алгоритм и программная реализация процесса запуска нескольких приложений, работающих при помощи ПО Wine, в изолированной среде выполнения Flatpak с использованием системы межпроцессного взаимодействия D-Bus. Результаты данной работы могут быть использованы разработчиками для поддержки ОС Linux в приложениях, разработанных исключительно для ОС Windows.

Библиографический список

1. WineHQ – About Wine [Электронный ресурс]. – URL: <https://www.winehq.org/about> (дата обращения: 10.05.2024).
2. Flatpak – The future of apps on Linux [Электронный ресурс]. – URL: <https://www.flatpak.org/> (дата обращения: 12.05.2024).
3. Under the hood – Flatpak Documentation [Электронный ресурс]. – URL: <https://docs.flatpak.org/en/latest/under-the-hood.html> (дата обращения: 12.05.2024).
4. D-Bus freedesktop [Электронный ресурс]. – URL: <https://www.freedesktop.org/wiki/Software/dbus/> (дата обращения: 16.05.2024).
5. Glib – 2.0 [Электронный ресурс]. – URL: <https://docs.gtk.org/glib/> (дата обращения: 17.05.2024).

Сведения об авторах

Викентьева Ольга Леонидовна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: oleovic@rambler.ru

Брейкин Алексей Дмитриевич – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: aleksey@breikin.ru

Е.В. Долгова, Н.Д. Кузнецов

Пермский национальный исследовательский
политехнический университет, Пермь

ОЦЕНКА ПРОИЗВОДИТЕЛЬНОСТИ ДЕТЕКЦИОННЫХ МОДЕЛЕЙ YOLO ДЛЯ ЗАДАЧ РАСПОЗНАВАНИЯ ЦИФР

В этой статье производится сравнение производительности моделей YOLO. Исследование выявило достоинства и недостатки каждой из них.

Ключевые слова: локализация цифр, обнаружение цифр, искусственный интеллект.

E.V. Dolgova, N.D. Kuznetsov

Perm National Research Polytechnic University, Perm

PERFORMANCE EVALUATION OF YOLO DETECTION MODELS FOR DIGIT RECOGNITION TASKS

This article compares the performance of YOLO models. The study revealed the advantages and disadvantages of each of them.

Keywords: digit localization, digit detection, artificial intelligence.

В современных условиях быстрого развития искусственного интеллекта и машинного обучения задача распознавания образов занимает одно из центральных мест. Особенно актуальной становится оценка производительности алгоритмов в условиях ограниченного времени обработки и высоких требований к точности. В этой статье основное внимание уделяется анализу производительности детекционных моделей YOLO (You Only Look Once), которые за счет своей уникальной архитектуры обеспечивают выдающуюся скорость и эффективность в задачах распознавания. Принимая во внимание важность численных преобразований в различных отраслях, можно констатировать растущую потребность в автоматизации процесса распознавания цифр.

Сравнительный анализ архитектуры YOLO с другими архитектурами в области объектного распознавания

1. *R-CNN (Region-based Convolutional Neural Network)*. R-CNN представляет собой сложный алгоритм обнаружения объектов, работающий в несколько этапов. На первом шаге с использованием метода извлече-

ния областей интереса, такого как Selective Search, происходит генерация предполагаемых регионов на изображении. Каждый из этих регионов затем нормализуется по размеру и подается на вход сверточной нейронной сети (CNN), которая извлекает из них признаки. Полученные признаки пропускаются через полносвязные слои, после чего классифицируются с помощью классификаторов, таких как SVM (Support Vector Machines). На финальном этапе применяется регрессор границ ограничивающих рамок для уточнения их положения. Результатом работы R-CNN является набор ограничивающих рамок с классами объектов и их вероятностями. R-CNN позволяет очень точно определять объекты, но процесс вычислений занимает значительное время. R-CNN требует дополнительного шага обучения для классификатора и регрессии границ боксов, что увеличивает временные затраты на тренировку.

2. *Fast R-CNN*. Улучшенная версия R-CNN, Fast R-CNN использует нововведение RoIPool (Region of Interest Pooling) для ускорения обработки извлеченных регионов. В отличие от R-CNN, подход использует общую карту признаков для всех регионов, что значительно сокращает время обработки и требования к памяти. Однако, несмотря на улучшения, поиск регионов все еще остается вычислительно затратным процессом.

3. *SSD (Single Shot MultiBox Detector)*. SSD ускоряет процесс обнаружения, выполняя только один проход через сеть. Он использует множество фильтров разных размеров для обнаружения объектов в различных масштабах. В отличие от R-CNN серии SSD обнаруживает объекты на различных уровнях слоев, что обеспечивает лучшее обнаружение объектов разного размера. SSD все еще уступает в точности подходам на основе R-CNN, но значительно выигрывает в скорости.

4. *YOLO (You Only Look Once)*. YOLO – это алгоритм обнаружения объектов, который обрабатывает изображение однократно, применяя глубокую сверточную нейронную сеть (CNN) для одновременного предсказания классов и расположения объектов. Сеть делит входное изображение на сетку, и каждая ячейка сетки отвечает за предсказание ограничивающих рамок и вероятностей наличия объектов в этих рамках. Предсказания включают класс объекта и параметры ограничивающих рамок: координаты центра, ширину и высоту. YOLO объединяет признаки изображения с пространственной информацией, что позволяет сети обучаться на контексте объектов и их масштабе, достигая высокой скорости и эффективности обнаружения объектов по сравнению с другими методами. YOLO работает с большей скоростью, чем SSD,

и значительно быстрее, чем R-CNN и Fast R-CNN, хотя может уступать в точности из-за проблем с небольшими или плотно расположенными объектами.

Сбор и подготовка обучающих данных. Сбор и подготовка обучающих данных являются критическими этапами при обучении любой модели машинного обучения. Процесс предполагает не только сбор достаточного количества образцов данных, но их аннотирование и последующую аугментацию, генерацию синтетических данных. В контексте настоящего исследования был сформирован датасет, насчитывающий 16 тыс. аннотированных изображений, что представляет собой значительную выборку для тренировки и тестирования моделей YOLO. Ниже рассмотрены ключевые этапы подготовки данного датасета.

1. *Сбор данных.* Сбор данных был осуществлен за счет активного поиска и селекции изображений, содержащих цифры в различных контекстах: от естественных сцен до цифровых интерфейсов. Это позволило обеспечить модели разнообразие данных, которое необходимо для формирования устойчивой к переносу задачи и построения обобщенной модели обнаружения.

2. *Аннотирование данных.* Аннотирование было выполнено вручную с использованием инструмента LabelImg, который позволяет отмечать объекты на изображениях с помощью ограничивающих рамок (bounding boxes). Каждая рамка содержит метки класса (в данном случае цифры 0–9, точка, минус) и координаты, которые используются для обучения детекционных моделей. Точность аннотирования напрямую влияет на качество и эффективность обучения модели.

3. *Аугментация данных.* Чтобы увеличить обобщающую способность модели и предотвратить переобучение, был проведен ряд процедур аугментации данных. Это включало методы, такие как горизонтальное и вертикальное зеркальное отражение, случайные обрезки, изменения контрастности, цветовые джиттеры и шумовые эффекты. Подобные техники аугментации позволяют значительно расширить вариативность обучающего набора данных без необходимости физического увеличения его объема.

4. *Генерация синтетических данных.* Для дальнейшего увеличения количества обучающих примеров и обеспечения представленности редких случаев в датасет были включены синтетические данные. Были использованы алгоритмы генерации изображений, которые имитируют различные стили цифр, а также моделируют условия освещения и фона,

приближенные к реальным сценариям использования. Синтетические данные оказались особенно полезными для повышения резистентности модели к шуму и искажениям.

5. *Разделение выборки на обучающую и валидационную (тестовую)*. Полученная выборка была поделена на обучающую и тестовую по стандартному правилу 80 на 20 %.

Анализ результатов обучения. В данном разделе исследования представлен анализ полученных результатов после обучения трех вариаций детекционной модели YOLO: YOLOv8n (nano), YOLOv8s (small) и YOLOv8m (medium). Результаты проанализированы на основе стандартных метрик производительности, включая точность (precision), полноту (recall), mAP (mean Average Precision), F1-score. Обучение проводилось на ранее подготовленном датасете из 16 тыс. изображений, прошедших аннотирование, аугментацию и дополнение синтетическими данными. Ниже приведено краткое описание каждой метрики.

1. Точность (Precision). Процент правильно идентифицированных объектов относительно всех идентифицированных моделью объектов. Эта метрика важна для минимизации ложных срабатываний.

2. Полнота (Recall). Доля найденных моделью правильных объектов от всего числа объектов в датасете. Эта метрика критична для минимизации упущенных объектов.

3. mAP (mean Average Precision). Средняя точность обнаружения объектов по всем классам. Эта комплексная метрика отражает общую точность модели.

4. Кривая F1-score представляет собой графическое отображение баланса между точностью и полнотой на различных пороговых значениях. F1-score является гармоническим средним точности и полноты, что позволяет оценить общее качество модели, особенно в условиях несбалансированного датасета. Значение F1-score варьируется от нуля до единицы, где «1» соответствует идеальной модели. Высокий F1-score указывает на то, что модель обладает хорошим балансом между высокой точностью и высокой полнотой. В контексте моделей YOLO, анализируя кривую F1-score, мы можем определить оптимальный порог для классификации объектов, что важно для достижения максимально эффективной детекции объектов в различных условиях.

Для обучения моделей использовался популярный фреймворк ultralytics, основанный на PyTorch, для обучения были использованы следующие гиперпараметры обучения (листинг).

Листинг – Гиперпараметры обучения модели

```
hyp = {}
hyp['lr0']= 0.001 # initial learning rate (i.e.
SGD=1E-2, Adam=1E-3)
hyp['lrf']= 0.01 # final learning rate (lr0 * lrf)
hyp['momentum']= 0.9 # SGD momentum/Adam beta1
hyp['weight_decay']= 0.0005 # optimizer weight de-
cay 5e-4
hyp['warmup_epochs']= 3.0 # warmup epochs (frac-
tions ok)
hyp['warmup_momentum']= 0.8 # warmup initial momen-
tum
hyp['warmup_bias_lr']= 0.1 # warmup initial bias lr
hyp['box']= 7.5 # box loss gain
hyp['cls']= 0.5 # cls loss gain (scale with pixels)
hyp['dfl']= 1.5 # dfl loss gain
hyp['pose']= 12.0 # pose loss gain
hyp['kobj']= 1.0 # keypoint obj loss gain
hyp['label_smoothing']= 0.1 # label smoothing
(fraction)
hyp['nbs']= 64 # nominal batch size
hyp['hsv_h']= 0.15 # image HSV-Hue augmentation
(fraction)
hyp['hsv_s']= 0.75 # image HSV-Saturation augmenta-
tion (fraction)
hyp['hsv_v']= 0.5 # image HSV-Value augmentation
(fraction)
hyp['degrees']= 0.1 # image rotation (+/- deg)
hyp['translate']= 0.2 # image translation (+/-
fraction)
hyp['scale']= 0.6 # image scale (+/- gain)
hyp['shear']= 0.1 # image shear (+/- deg)
hyp['perspective']= 0.00025 # image perspective
(+/- fraction), range 0-0.001
hyp['flipud']= 0 # image flip up-down (probability)
```

```

hyp['fliplr']= 0 # image flip left-right (probability)
hyp['mosaic']= 0.05 # image mosaic (probability)
hyp['mixup']= 0.09 # image mixup (probability)
hyp['copy_paste']= 0.09 # segment copy-paste (probability)

```

Результаты обучения продемонстрированы на рис. 1–6.

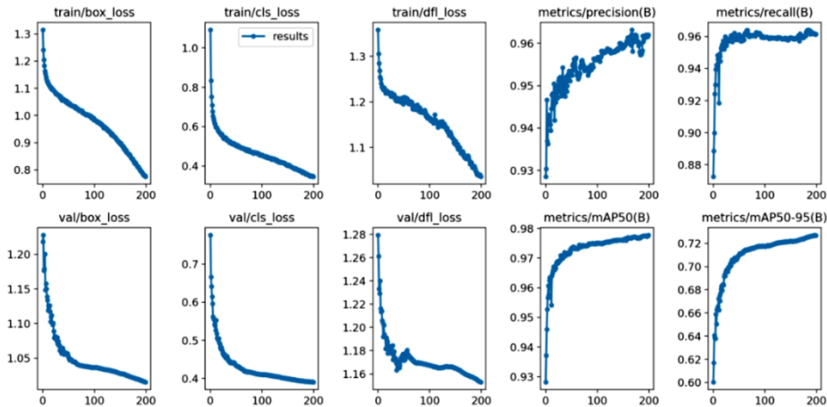


Рис. 1. Результаты обучения YOLOv8m

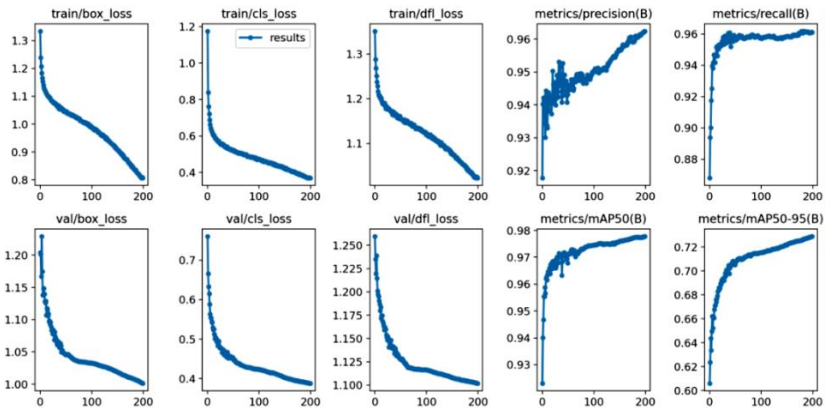


Рис. 2. Результаты обучения YOLOv8s

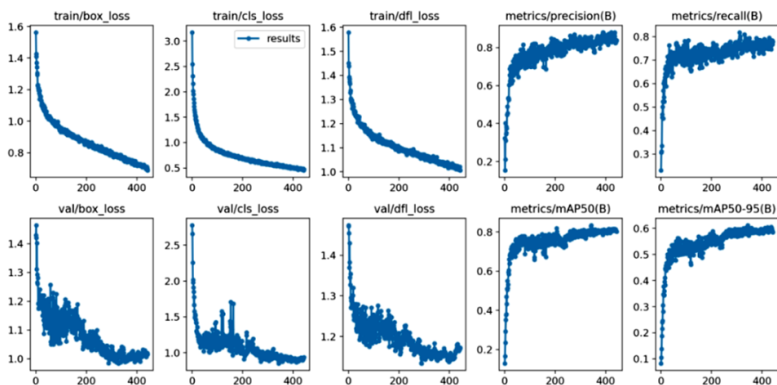


Рис. 3. Результаты обучения YOLOv8n

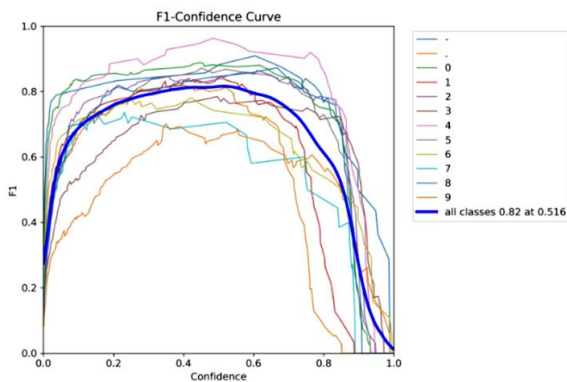


Рис. 4. F1-score YOLOv8n

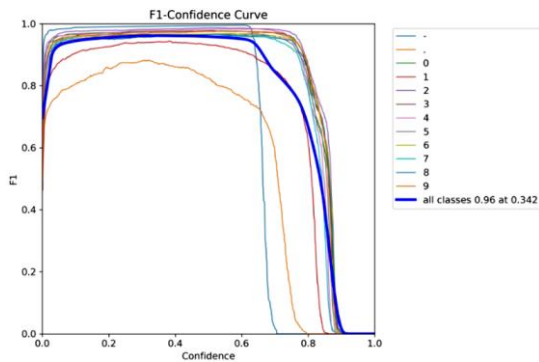


Рис. 5. F1-score YOLOv8s

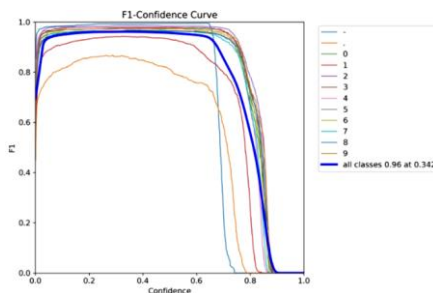


Рис. 6. F1-score YOLOv8m

Заключение.

1. YOLOv8n: Модель 'nano' показала самую высокую скорость инференса благодаря своей легковесной архитектуре, однако в ущерб точности и полноте. mAP данной модели оказался ниже сравниваемых моделей, что является следствием компромисса между скоростью и детализацией из-за упрощенной структуры.

2. YOLOv8s: Вариант 'small' стал золотой серединой, предлагая уравновешенное соотношение скорости и точности. Модель достигла средней значения по mAP на датасете за счет более обширной и сложной архитектуры по сравнению с 'nano', при этом сохраняя адекватные скорость инференса.

3. YOLOv8m: Модель 'medium' продемонстрировала лучшую точность и полноту, которые отразились на повышении mAP. Однако это также сказалось на увеличении времени инференса. Данная архитектура – предпочтительный выбор, когда необходима максимально возможная точность детекции и время обработки не является приоритетным фактором.

Библиографический список

1. Ярышев С.Н., Рыжова В.А. Технологии глубокого обучения и нейронных сетей в задачах видеоанализа. – 2022.
2. Официальная документация ultralytics. – URL: <https://docs.ultralytics.com>

Сведения об авторах

Долгова Елена Владимировна – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: elena@dolgova.space

Кузнецов Николай Дмитриевич – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-1б, г. Пермь, e-mail: kuznetsovnik1133@gmail.com

О.А. Полякова, А.А. Фогельгезанг

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА МОДУЛЯ МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ПРОВЕДЕНИЯ ПРИЕМКИ НЕДВИЖИМОСТИ

Приемка недвижимости является важным этапом в процессе покупки недвижимости, который заключается в проверке состояния объекта перед его передачей покупателю. Процесс включает в себя осмотр объекта недвижимости, проверку документов, а также выявление возможных недостатков или проблем, которые могут повлиять на будущее использование или стоимость недвижимости. Правильно проведенная приемка помогает избежать неприятных сюрпризов после покупки и защищает интересы покупателя. Для повышения эффективности приемки и дополнительного сбора статистики и разработан модуль мобильного приложения для приемки недвижимости компании-застройщика «Талан».

Ключевые слова: проектирование системы, разработка мобильного приложения, android, приемка недвижимости.

O.A. Poliakova, A.A. Fogelgezang

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A MODULE FOR A MOBILE APPLICATION FOR REAL ESTATE INSPECTION

Acceptance of real estate is an important step in the real estate purchasing process, which involves checking the condition of the property before handing it over to the buyer. This process includes inspecting the property, checking documents, and identifying possible deficiencies or problems. Which may affect the future use or value of the property. Proper acceptance helps avoid unpleasant surprises after the purchase and protects the interests of the buyer. To increase the efficiency of acceptance and additional collection of statistics, a mobile application module for acceptance of real estate was developed in the Talan company.

Keywords: system design, mobile application development, android, real estate inspection.

Для любой компании процесс взаимодействия с пользователем очень важен, поэтому большинство из них имеют современные мобильные приложения, позволяющие не только познакомиться с услугами компании, но и получать их дистанционно.

В частности, компании по строительству имеют мобильные при-

ложения, которые позволяют реализовать полный жизненный цикл взаимодействия с клиентом: от выбора квартиры до получения ключей и проживания.

Бизнес-процесс «Приемка недвижимости» начинается с момента, когда застройщик направляет собственникам приглашение на осмотр квартиры с целью проверки качества выполненных работ, а завершается в момент подписания акта приема-передачи, который обозначает передачу квартиры собственнику в состоянии, удовлетворяющем строительным нормам и заявленному качеству. В промежутке между этими контрольными точками происходит осмотр квартиры, возможно неоднократный, в зависимости от наличия недостатков.

Перед началом приемки сотрудник от застройщика обязан предоставить клиенту смотровой лист – документ, в котором описаны строительные дефекты. Акт приема-передачи подписывается только в случае, если отсутствуют существенные дефекты и процесс приемки завершается. При наличии серьезных нарушений стороны подписывают дефектный акт, далее замечания устраняются и клиент приглашается на повторную приемку.

Для анализа существующего бизнес-процесса приемки недвижимости авторами разработана модель AS-IS в формате USE-CASE-диаграммы (рис. 1).

Таким образом, выявлены следующие параметры оценивания эффективности бизнес-процесса работы с клиентом по приемке недвижимости:

- коммуникация – в текущем ритме жизни телефонные звонки становятся все менее предпочтительным средством связи, а выбирать дату всегда удобнее при просмотре календаря;
- отслеживаемость и контроль – человек всегда чувствует себя комфортнее, когда точно знает, что происходит, в текущей реализации контроль статусов по приемке затруднен тем же способом коммуникации;
- неактуальность оценки – с момента окончания приемки до обзвона клиентов проходит неопределенное количество времени, и клиент может не вспомнить всех подробностей встречи;
- отсутствие статистических данных – по окончании приемки объекта недвижимости клиентом вручную происходит сбор оценок менеджером компании, при этом статистика результатов приемки отсутствует.

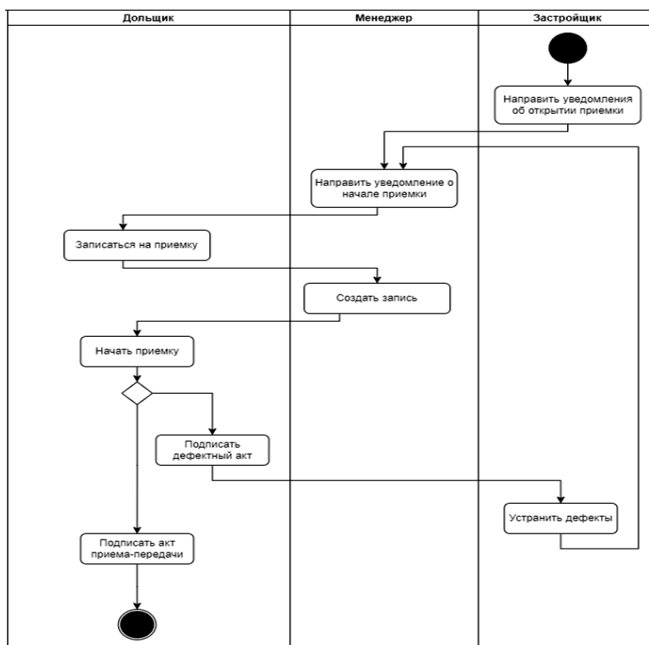


Рис. 1. Модель AS-IS бизнес-процесса приемки недвижимости

Автоматизация и цифровизация бизнес-процесса приемки устраняет выявленные недостатки. Таким образом, авторами разработаны требования, проект структуры, основные инструменты реализации программного обеспечения системы автоматизации модуля мобильного приложения приемки объектов недвижимости (таблица).

Требования к программному обеспечению модуля приемки

Требование	Описание
Дистанционная запись	Пользователь должен иметь возможность записаться на приемку недвижимости: в приложении должны отображаться доступные дни и время для записи. Если клиент имеет несколько объектов недвижимости, должна быть возможность записи на комплексную приемку (несколько объектов в один день)
Уведомления	Пользователь должен получать своевременные уведомления об изменении статусов приемки, отмене или переносе записи
Дополнительная информация	Приложение должно иметь информационные экраны для помощи пользователю. Например, экран «Ответы на вопросы» может содержать пункт «Что взять с собой на приемку?»
Чек-лист приемки	Для проведения приемки в приложении должен отображаться заранее заведенный экспертами в системе чек-лист, отражающий возможные дефекты. Клиент должен иметь возможность отметить пункты вариантами да/нет или написать свой вариант в комментарии

На основании анализа требований разработана USE-CASE диаграмма модели TO-BE бизнес-процесса приемки недвижимости (рис. 2).

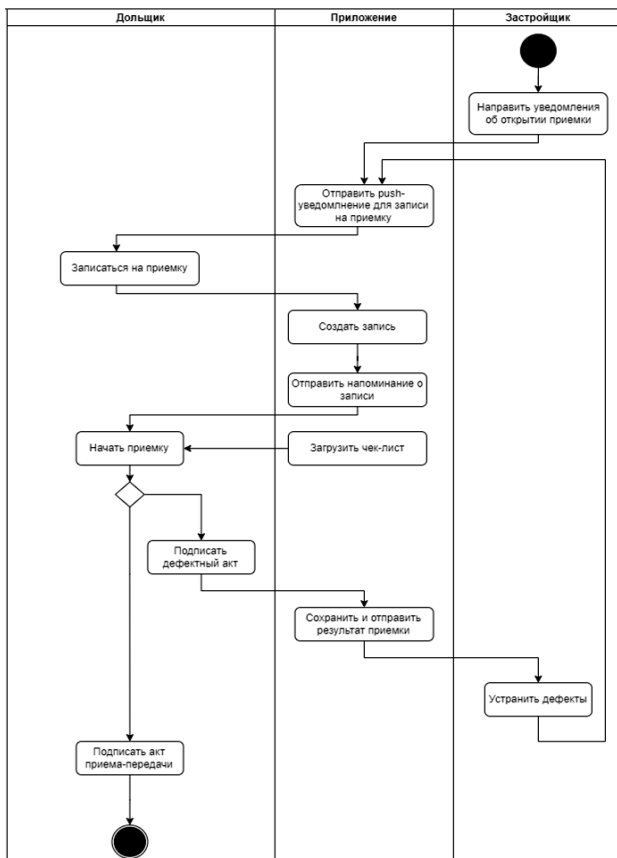


Рис. 2. Модель TO-BE бизнес-процесса приемки недвижимости

Авторами разработан пользовательский интерфейс модуля приемки, где главный экран со списком объектов приемки содержит краткую информацию по статусам; экран с чек-листом приемки содержит список вопросов и ответы «да/нет»; на экране оценки приемки показано выставление оценки по пятибалльной шкале и поле для дополнительных вопросов; экран записи на приемку объекта недвижимости предоставляет элементы выбора времени и даты в формате календаря; также разработаны формы информационных экранов с ответами на популяр-

ные вопросы. Поскольку приложение заказчика написано на языке программирования Kotlin, то авторами принято решение для разработки модуля приемки использовать тот же язык [1]. Для разработки экранов использована библиотека Conductor, при этом структура классов разработана в нотации Model-View-Presenter.

Верстка экранов выполнена в xml-файлах и подключается к слоям Model-View-Presenter через встроенный инструмент View Binding. Примеры разработки форматов экрана представлены на рис. 3.

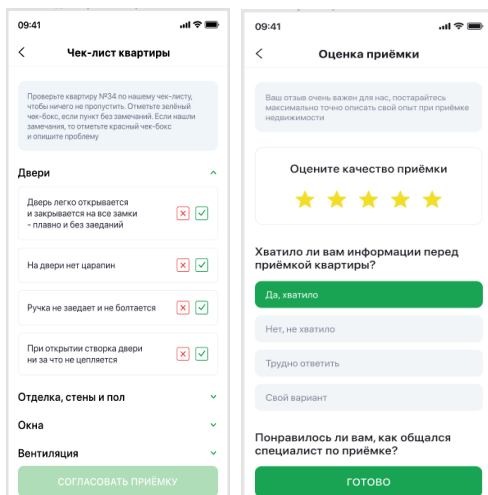


Рис. 3. Чек-листы приемки и оценки

Актуальная информация экранов формируется через запросы к серверу и сохраняется в локальной базе данных SQLite, для оперирования данными используется библиотека Room Database [2].

Диаграмма классов показывает отношение сущностей базы данных через взаимосвязи классов (рис. 4).

Сущности диаграммы классов:

- AcceptedRealEstateBannerModel – объект для хранения данных блока уведомлений;
- AcceptedRealEstate – основная информация о приемке;
- AcceptedRealEstateCheckListModel – данные чек-листа приемки;
- QuestionGroup – категории критериев (Question) оценки недвижимости;
- EstimateAcceptance – данные об оценке приемки, список вопросов (Estimate) и их оценку.

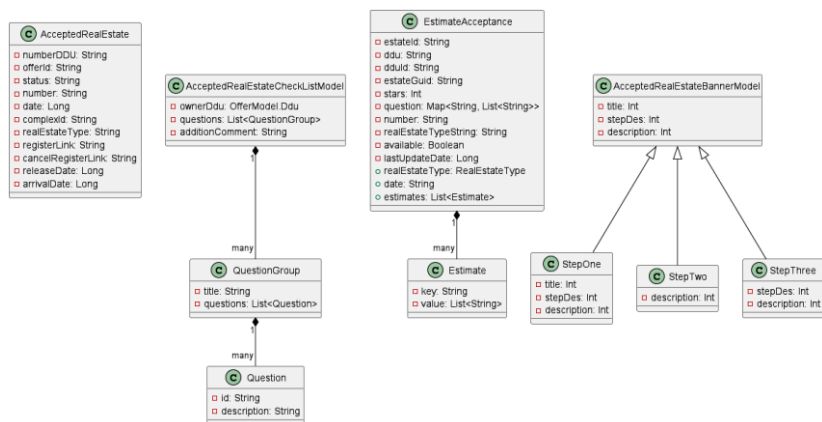


Рис. 4. Диаграмма классов

Таким образом, авторами разработана система автоматизации приемки объектов недвижимости, реализованная в модуле мобильного приложения, что позволило минимизировать участие менеджеров в процессе приемки за счет дистанционного формата записи и постоянного уведомления о статусах. За счет сокращения времени оценки приемки и формирования промежуточной оценки повышена достоверность полученных результатов оценки.

Библиографический список

1. Kotlin [Электронный ресурс]. – URL: Kotlin docs <https://kotlinlang.org/docs/home.html> (дата обращения: 08.05.2024).
2. Save data in a local database using Room [Электронный ресурс]. – URL: <https://developer.android.com/training/data-storage/room> (дата обращения: 08.05.2024).

Сведения об авторах

Полякова Ольга Андреевна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: olgastratum@mail.ru

Фогельгезанг Александра Артемовна – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: ps.ovsyanka@gmail.com

О.Л. Викентьева, А.В. Пономарев

Пермский национальный исследовательский
политехнический университет, Пермь

АВТОМАТИЗАЦИЯ ПРОЦЕССА УСТАНОВКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ДОПОЛНИТЕЛЬНЫХ КОМПОНЕНТОВ

В данной статье рассмотрен процесс создания инсталлятора десктопного приложения, написанного на языке C#.

Ключевые слова: инсталляция приложения, архивирование данных, база данных.

O.L. Vikentieva, A.V. Ponomarev

Perm National Research Polytechnic University, Perm

AUTOMATION OF THE INSTALLATION PROCESS OF SOFTWARE AND ADDITIONAL COMPONENTS

This article describes the process of creating an installer for desktop software, DBMS and additional components.

Keywords: application installation, data archiving, database.

Инсталляция программного обеспечения представляет собой процесс развертывания программы на целевую систему, обеспечивая ее корректную работу и интеграцию с операционной системой и другими программами. Этот процесс включает копирование файлов, создание ярлыков, настройку системных параметров и установку необходимых зависимостей. Инсталлятор — это специализированное программное обеспечение, которое автоматизирует этот процесс, предоставляя пользователю удобный интерфейс и минимизируя возможность ошибок (рис. 1). Существуют готовые решения, которые автоматизируют процесс создания инсталляторов.

Далее будут рассмотрены наиболее популярные инструменты для автоматического создания инсталляторов приложений на .NET, их преимущества и недостатки по сравнению с разработкой собственного инсталлятора. В данной статье рассматривается процесс создания инсталлятора на языке программирования C# с использованием технологии WPF.

Таблица 1

Сравнительный анализ средств автоматического создания инсталляторов

Инструмент	Достоинства	Недостатки
WiX Toolset [1]	1. Высокая гибкость. 2. Поддержка сложных сценариев установки. 3. Полный контроль над процессом установки через XML	1. Сложность в освоении. 2. Трудоемкость ручного написания сценариев. 3. Отсутствие графического интерфейса для проектирования. 4. Высокая вероятность ошибок конфигурации
InstallShield	1. Поддержка широкого спектра платформ и типов установок. 2. Удобный пользовательский интерфейс и мастера настройки	1. Высокая стоимость лицензии. 2. Избыточность для простых проектов. 3. Длительное время обучения
Inno Setup [2]	1. Простота использования. 2. Поддержка скриптов для настройки процесса установки. 3. Активное сообщество и множество примеров	1. Ограниченная гибкость. 2. Ограниченная поддержка скриптов. 3. Отсутствие визуального редактора
NSIS [3]	1. Легкость и компактность создаваемых установщиков. 2. Высокая гибкость благодаря скриптам. 3. Поддержка множества плагинов	1. Сложность написания скриптов. 2. Ограниченный графический интерфейс. 3. Трудности с отладкой. 4. Ограниченная поддержка сложных сценариев

Исходя из результатов сравнительного анализа (табл. 1), можно выделить основные недостатки инструментов для автоматического создания инсталляторов: сложность освоения языков написания сценариев установки, недостаточная поддержка сложных сценариев установки, недостаточная гибкость при создании графического интерфейса.

Решением проблем, которые возникают при использовании инструментов автоматического создания инсталляторов, может стать проектирование и реализация собственного инсталлятора на языке C# и использованием платформы .NET.

Почему разработка собственного инсталлятора на C# может быть лучше. Разработка собственного инсталлятора позволяет реализовать все необходимые функции и логику установки, которые могут быть специфичны для вашего приложения, при этом исключая избыточный функционал.

1. Гибкость и адаптация под конкретные требования.

Легкая адаптация и изменение инсталлятора в соответствии с требованиями, добавление уникальные шаги установки, пользовательского интерфейса и логических проверок.

2. Интеграция с другими компонентами.

С использованием C# и .NET можно легко интегрировать установку других компонентов и зависимостей, таких как .NET Framework, базы данных, службы и т.д.

3. Кастомизация пользовательского интерфейса.

Использование WPF позволяет создавать современные, интуитивно понятные и привлекательные пользовательские интерфейсы, что улучшает пользовательский опыт во время установки.

4. Единая технология для разработки.

Если основное приложение также написано на C#, разработка инсталлятора на том же языке упрощает поддержку и улучшает совместимость.

Постановка задачи. Рассмотрим процесс проектирования и реализации инсталлятора на примере инсталлятора .NET приложения TFortis Device Manager. Его установка заключается в распаковке файлов приложения и установке дополнительных компонентов. Она включает в себя следующие требования и шаги:

- распаковка файлов из tar архива [4].
- запись информации в реестр Windows.
- установка компонента .NET Windows Desktop Runtime.
- установка PostgreSQL Server.
- создание базы данных и заполнение ее из SQL файла.
- создание ярлыка приложения.
- автоматический запуск приложения после установки.

Реализация. Проектирование и подготовка дополнительных файлов. На рис. 1 представлен алгоритм установки приложения. Его этапы можно разделить на обязательные и опциональные. Список этапов и условия для их выполнения представлены в табл. 2.

Перед началом реализации алгоритмов установки нужно создать интерфейс инсталлятора, он будет представлять из себя окно с кнопками «Далее», «Назад», «Установить» и «Отмена», полями для ввода и чек-боксами для выбора дополнительных опций и компонентов. Интерфейс главного окна разрабатываемого инсталлятора представлен на рис. 2.

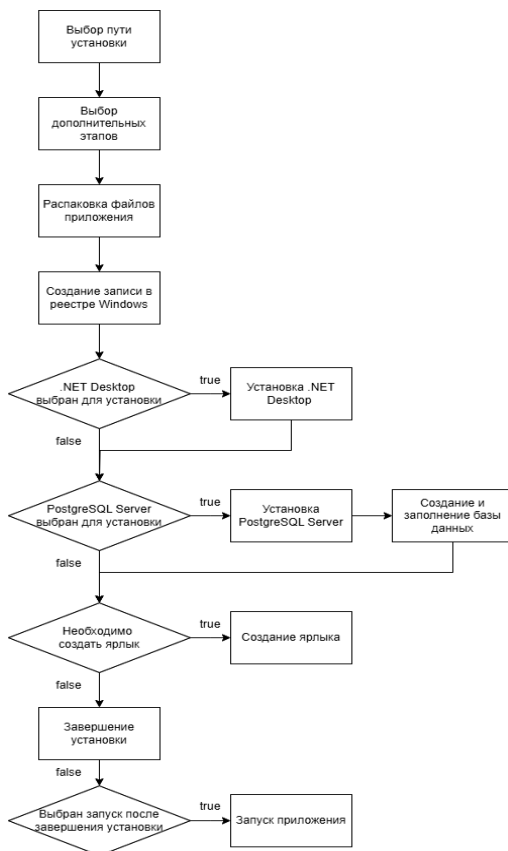


Рис. 1. Алгоритм установки приложения

Таблица 2

Этапы установки приложения и условия для их выполнения

Процесс	Условие
Распаковка файлов приложения	—
Занесение записи в реестр Windows	—
Установка PostgreSQL Server	PostgreSQL Server выбран для установки и не установлен на компьютере пользователя
Установка .NET Windows Desktop Runtime 6.0	.NET Windows Desktop Runtime 6.0 выбран для установки и не установлен на компьютере пользователя
Создание базы данных	База данных не существует
Заполнение базы данных	Данные некорректны или устарели
Создание ярлыка на рабочем столе	Выбрано создание ярлыка
Запуск приложения после выхода из инсталлятора	Выбран запуск приложения после выхода из инсталлятора



Рис. 2. Главное окно инсталлятора

Входными данными для инсталлятора будет являться выбранный путь установки, выбранные компоненты и опции, заранее подготовленный архив с данными приложения, установочные файлы дополнительных компонентов и SQL-скрипт для заполнения базы данных. Исходя из входных данных, определяется алгоритм установки приложения и набор этапов, которые будут выполнены в ходе установки.

Реализация этапов установки. Распаковка файлов из tar архива. Распаковка основных файлов приложения выполняется независимо от входных данных (рис. 3).

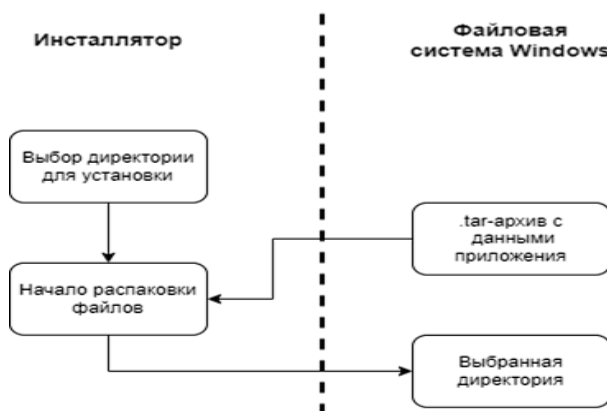


Рис. 3. Процесс распаковки файлов приложения в выбранную директорию

Алгоритм распаковки файлов реализован в функции ExtractFiles. Аргументами функции ExtractFiles являются две строки – расположение архива с данными приложения и путь для установки, выбранный пользователем. Процесс распаковки файлов приложения сопровождается кольцом прогресса в интерфейсе пользователя.

Запись информации в реестр Windows. Для создания записи о программе в реестре Windows используются классы Registry из пространства имен Microsoft.Win32. В ходе выполнения функции создаются записи с информацией о приложении в соответствующие регистры реестра Windows.

Установка компонентов .NET Windows Desktop Runtime и PostgreSQL Server. Для установки сторонних компонентов, необходимых для корректной работы приложения, используется процесс Process.Start(), который запускает установочные файлы компонентов. Во время установки сторонних компонентов основной процесс приостанавливается.

Создание ярлыка приложения. Для создания ярлыка в конце установки приложения используются классы WshShell и IWshShortcut [5]. Ярлык создается на рабочем столе, ссылкой для него будет строка, являющаяся результатом конкатенации выбранной директории установки и названия исполняемого файла приложения.

Автоматический запуск приложения после установки. При нажатии на кнопку «Завершить» в случае, если выбрана опция запуска приложения после выхода из инсталлятора, при срабатывании деструктора окна создается процесс, запускающий исполняемый файл установленного приложения.

Заключение. Разработка инсталлятора программного обеспечения на языке C# может помочь вам получить полный контроль над функциональностью, адаптируемость под конкретные требования, доступ к библиотекам .NET, возможность полностью кастомизировать пользовательский интерфейс. При правильном подходе можно создать инсталлятор, который обеспечит гладкую установку программного обеспечения на компьютеры пользователей.

Библиографический список

1. Документация по WixToolset [Электронный ресурс]. – URL: <https://wixtoolset.org/docs/v3/> (дата обращения: 18.05.2024).
2. Документация по InnoSetup [Электронный ресурс]. – URL: <https://jrsoftware.org/ishelp/> (дата обращения: 18.05.2024).

3. Документация по NSIS [Электронный ресурс]. – URL: <https://nsis.sourceforge.io/Docs/> (дата обращения: 18.05.2024).

4. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео / Д. Ватолин, А. Ратушняк, М. Смирнов, В. Юкин. – М.: ДИАЛОГ-МИФИ, 2002. – 384 с.

5. Документация по .NET [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/dotnet/api/> (дата обращения: 09.05.2024).

Сведения об авторах

Викентьева Ольга Леонидовна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: oleovic@pstu.ru

Пономарев Артем Викторович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: wylsomerice@gmail.com

О.Л. Викентьева, М.М. Катаев

Пермский национальный исследовательский
политехнический университет, Пермь

АВТОМАТИЗАЦИЯ СКЛАДСКИХ ПРОЦЕССОВ С ПОМОЩЬЮ МОБИЛЬНЫХ УСТРОЙСТВ

В данной статье рассматривается способ автоматизации складских процессов с помощью мобильных устройств. Описаны возможности мобильных устройств. Приведен реальный пример успешной автоматизации складских процессов на предприятии с помощью мобильных устройств.

Ключевые слова: складские процессы, мобильные устройства, автоматизация, широковещательные сообщения.

O.L. Vikenteva, M.M. Kataev

Perm National Research Polytechnic University, Perm

WAREHOUSE PROCESS AUTOMATION USING MOBILE DEVICES

This article discusses a way to automate warehouse processes using mobile devices. The possibilities of mobile devices are described. A real example of successful automation of warehouse processes in an enterprise using mobile devices is given.

Keywords: warehouse processes, mobile devices, automation, broadcast messages.

На сегодняшний день для успешной и конкурентоспособной работы предприятия недостаточно производить продукцию высокого качества, также важно уметь эффективно управлять ресурсами предприятия, включая продукцию на складе. Эффективное управление складскими запасами позволяет уменьшить время на обработку и снизить риск возникновения ошибки при выполнении складских операций. Этого можно достичь за счет автоматизации складских процессов.

Всего на складе существует несколько основных процессов: оприходование, инвентаризация, комплектация, отгрузка и перемещение [1].

Оприходование представляет собой процесс приема продукции на склад с производственного участка. Кладовщику необходимо визуально проверить целостность продукции и сверить ее с сопроводительным документом.

Инвентаризация – это процесс актуализации данных о продукции, хранящейся на складе, в системе предприятия. В ходе инвентаризации кладовщик должен сверить количество, указанное в инвентаризационной описи, с фактическим количеством на складе. Если количество какой-либо продукции в инвентаризационной описи не совпадает с фактическим, то кладовщику необходимо внести изменения в систему предприятия, указав актуальную информацию.

Комплектация является процессом, который подразумевает наличие заявки на отгрузку. При комплектации кладовщику требуется переместить продукцию, которая указана в документе комплектации, со склада в зону комплектации. Поиск необходимой продукции на складе может занимать длительное время, поскольку в документе комплектации не указывается место хранения продукции на складе.

Отгрузка – это процесс размещения продукции в транспортном средстве, доставляющем продукцию заказчику, основанный на заявке или заказе. При этом в одно транспортное средство может отгружаться продукция сразу по нескольким заказам, из-за чего кладовщику требуется быть предельно внимательным, чтобы не допустить ошибку.

Перемещение можно назвать основной операцией на складе, поскольку все остальные складские процессы, за исключением инвентаризации, так или иначе включают в себя перемещения продукции с одного места на другое. Однако в большинстве систем не учитывается точное местоположение продукции на складе, поэтому процесс перемещения никак не отражается в системе предприятия.

Почти во всех вышеперечисленных складских операциях кладовщику необходимо обращаться к стационарному компьютеру для внесения новой или изменения существующей информации о продукции на складе в систему предприятия, а также вручную сверять информацию на бирках продукции с информацией по документу [2].

Таким образом, из-за необходимости обращения к стационарному компьютеру возрастает время выполнения складских операций кладовщиками. Также из-за отсутствия учета местонахождения продукции на складе кладовщикам приходится тратить значительное время на ручную сверку информации на бирке продукции с данными в документе. Этот процесс является затратным по времени, поскольку требует предельной точности и внимательности от кладовщика. Кроме того, он увеличивает вероятность возникновения ошибок и несоответствий, что может привести к неверному оприходованию, комплектации или инвентаризации, что

в конечном итоге может отрицательно сказаться на эффективности работы предприятия, а также привести к финансовым потерям.

Одним из способов решения выявленных проблем является использование мобильных устройств при выполнении складских операций [3]. Мобильные устройства представляют собой компактные устройства, которые могут быть использованы в любой точке склада. При использовании кладовщиком мобильного устройства исключается этап обращения к стационарному компьютеру для взаимодействия с системой предприятия. Мобильные устройства могут обеспечивать взаимодействие с системой предприятия, используемой на стационарных компьютерах, позволяя кладовщикам получать актуальную информацию и изменять ее в системе. Также мобильные устройства имеют широкий функционал, например, возможность сканирования штрих-кодов и QR-кодов, чтения RFID меток. Данный функционал может быть использован, например, для идентификации продукции на складе.

Рассмотрим успешную автоматизацию складских процессов на предприятии ООО «Камский Кабель» с использованием мобильных устройств. В качестве примера будет рассмотрен процесс оприходования продукции на складе. На рис. 1 изображена диаграмма активности процесса оприходования продукции без использования мобильного приложения.

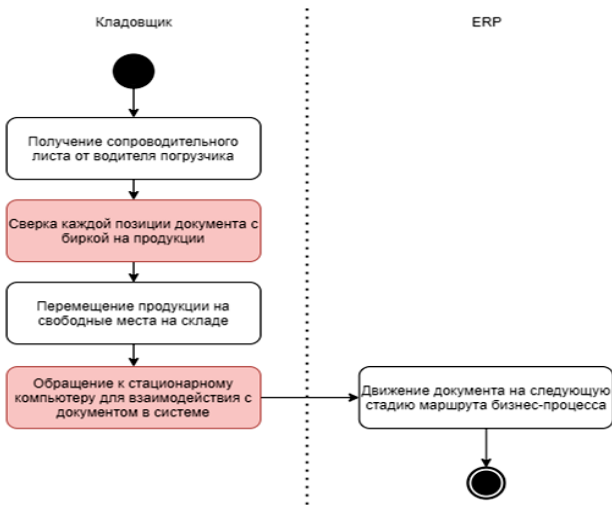


Рис. 1. Диаграмма активности процесса оприходования

Из-за необходимости обращения к ERP-системе, которая установлена на стационарном компьютере, возрастает время выполнения складских операций кладовщиками. Также кладовщикам приходится тратить значительное время на ручную сверку информации на бирках продукции с данными в документах. Этот процесс является затратным по времени, поскольку требует предельной точности и внимательности от кладовщика. Кроме того, он увеличивает вероятность возникновения ошибок и несоответствий, что может привести к неверному оприходованию продукции.

Указанные недостатки были решены с помощью использования мобильных устройств. На предприятии были закуплены терминалы сбора данных на базе операционной системы Android. Они представляют собой мобильные устройства со встроенным сканером штрихкодов, которые используются для сбора, хранения, систематизации и передачи данных в программу. Поскольку терминалы сбора данных имеют встроенный сканер штрихкодов, было произведено штрихкодирование продукции, документов и ячеек хранения. Затем было разработано мобильное приложение для операционной системы Android на языке программирования Java, разделенное на модули – для каждой складской операции свой модуль и сервер, который является связующим звеном между мобильным приложением и сервером базы данных предприятия.

При сканировании штрихкода терминал сбора данных отправляет широковещательные сообщения (листинг), которые регистрируются в мобильном приложении с помощью класса BroadcastReceiver [4].

Листинг – Регистрация приемника широковещательных сообщений

```
String PARAM_TASK = "data"; //ключ для получения значения
String BROADCAST_ACTION =
"com.android.server.scannerservice.broadcast";
@Override
protected void onResume() {
    super.onResume();
    BroadcastReceiver br = new
BroadcastReceiver() {
        // действия при получении сообщений
        public void onReceive(Context context,
Intent intent) {
            String Message =
intent.getStringExtra(PARAM_TASK); //получение значения по ключу
```

```

        //Логика обработки сообщения
    }
};
IntentFilter intFilt = new
IntentFilter(BROADCAST_ACTION); //создание фильтра
registerReceiver(br, intFilt); //регистрация
приемника
}

```

Чтобы BroadcastReceiver принимал только нужные нам сообщения, транслируемые в широковещание, необходимо установить Intent Filter [5]. Это набор параметров: action, data, category, если параметры принимаемого Intent совпадают с условиями этого фильтра, то BroadcastReceiver регистрирует сообщение. В документации терминала сбора данных указано значение action для сканирования – “com.sunmi.scanner.ACTION_DATA_CODE_RECEIVED”.

В методе onReceive приложение преобразует полученное сообщение в строку message. В зависимости от полученного сообщения и модуля, в котором отсканирован штрихкод, определяется логика выполнения программы. В модуле оприходования логика будет следующей:

- если полученное сообщение содержит префикс “DOC_”, то мобильное приложение запрашивает информацию о продукции для оприходования по номеру документа, который указан после префикса;
- если полученное сообщение не содержит префикса “DOC_” и не соответствует регулярному выражению для ячейки хранения, то выполняется следующий процесс: происходит поиск продукции по документу, у которой штрихкод совпадает с полученным сообщением, и после этого данная продукция отмечается в приложении;
- если полученное сообщение попадает под регулярное выражение адреса ячейки хранения, то выделенной продукции в системе присваивается отсканированная ячейка хранения.

Также в мобильном приложении была реализована возможность движения документа на следующую стадию маршрута бизнес-процесса. На рис. 2 изображена диаграмма активности процесса оприходования продукции с использованием терминала сбора данных. На диаграмме показано, что кладовщику при использовании терминала сбора данных в процессе оприходования не нужно вручную сверять информацию на каждой бирке продукции с информацией в документе оприходования, а также не нужно обращаться к стационарному компьютеру для движения документа на следующую стадию маршрута бизнес-процесса.

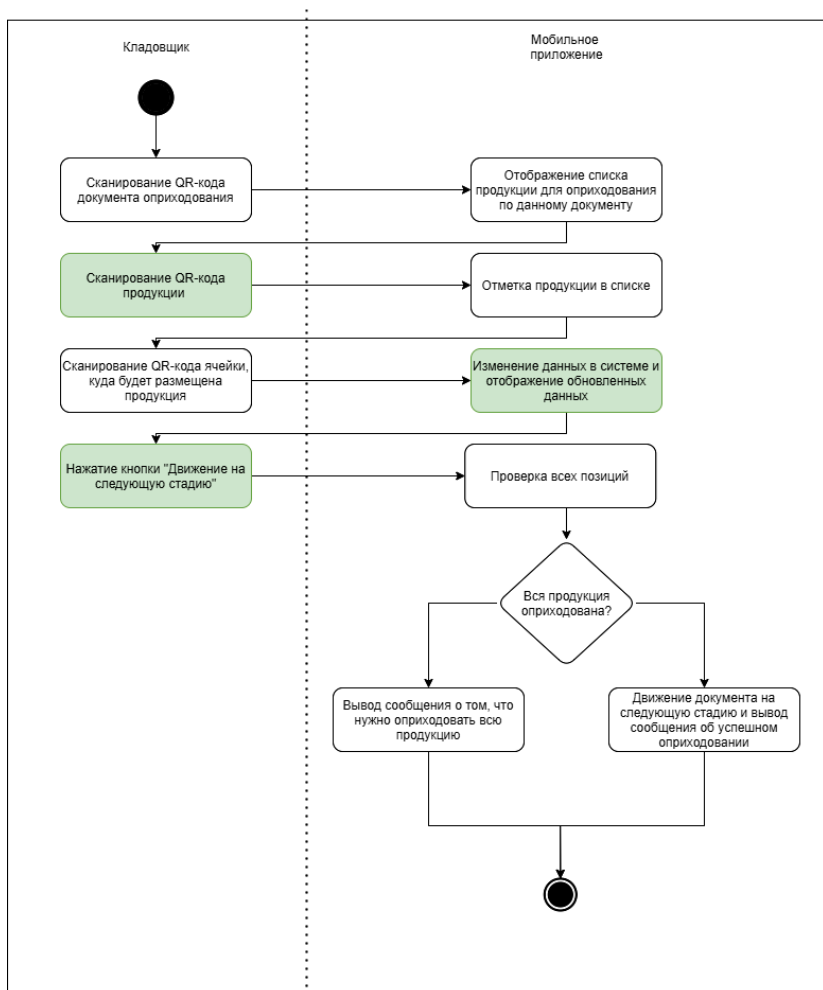


Рис. 2. Диаграмма активности процесса оприходования с использованием терминала сбора данных

Таким образом, автоматизация складских процессов с помощью мобильных устройств минимизирует риск возникновения ошибки кладовщика в процессе работы, а также ускоряет работу склада за счет отсутствия ручной сверки всех позиций по документу и необходимости обращаться к стационарному компьютеру.

Библиографический список

1. Волгин В.В. Склад: логистика, управление, анализ. – М.: Альфа-Пресс, 2009. – 720 с.
2. Ардатова М.М. Логистика в вопросах и ответах. – М.: Проспект, ТК Велби, 2004. – 272 с.
3. Эмметт С. Искусство управления складом. Как уменьшить издержки и повысить эффективность. – М.: Гревцов Паблишер, 2019. – 310 с.
4. BroadcastReceiver | Android Developers [Электронный ресурс]. – URL: developer.android.com- (дата обращения: 03.05.2024).
5. IntentFilter | Android Developers [Электронный ресурс]. – URL: developer.android.com (дата обращения: 03.05.2024).

Сведения об авторах

Викентьева Ольга Леонидовна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: oleovic@pstu.ru

Катаев Максим Михайлович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: maxkataev111@gmail.com

О.Л. Викентьева, Е.П. Бурмантов

Пермский национальный исследовательский
политехнический университет, Пермь

ИНТЕГРАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ С ERP-СИСТЕМОЙ ПРЕДПРИЯТИЯ

Актуальной проблемой для крупных промышленных предприятий является интеграция программного обеспечения с существующей на предприятии ERP-системой. В статье рассматриваются основные проблемы, которые решает интеграция программного обеспечения, способы интеграции, выделяются достоинства и недостатки каждого способа. Рассматривается пример интеграции вида «интеграция данных» с ERP-системой предприятия.

Ключевые слова: интеграция, программное обеспечение, информационная система, ERP-система, база данных, данные.

O.L. Vikentieva, E.P. Burmantov

Perm National Research Polytechnic University, Perm

INTEGRATION OF THE SOFTWARE WITH THE COMPANY'S ERP SYSTEM

An urgent problem for large industrial enterprises is the integration of software with the existing enterprise ERP-system. This article discusses the main problems that software integration solves, integration methods, and highlights the advantages and disadvantages of each method. An example of integration of the "data integration" type with the enterprise ERP-system is considered.

Keywords: integration, software, information system, ERP-system, database, data.

Одной из ключевых технологий в управлении деятельностью предприятия являются ERP-системы, которые помогают автоматизировать основные процессы работы предприятия. На крупных производственных предприятиях возникает необходимость использования нескольких информационных систем или видов программного обеспечения, поэтому возникает проблема интеграции ПО.

Интеграция программного обеспечения – процесс объединения всего ПО, используемого на предприятии, в единую информационную систему [1]. Основная идея интеграции заключается в том, чтобы быстро переносить данные из одной информационной системы и использо-

вать их в других видах программного обеспечения, а также использовать готовые функции одного ПО в других.

Интеграция ПО решает следующие проблемы:

- дублирование данных – пользователи вводят одинаковые данные с помощью разных программных продуктов;
- низкая скорость взаимодействия между различными видами ПО
- взаимодействие между разными видами программного обеспечения происходит вручную, силами работников предприятия;
- дублирование функционала при разработке ПО – разное программное обеспечение имеет одинаковый функционал;
- низкая степень удобства взаимодействия с информационной системой предприятия – пользователям приходится работать сразу в нескольких видах программного обеспечения в рамках одной задачи;
- человеческий фактор при переносе информации из одной системы в другую – ошибки, которые допускают пользователи при переносе данных из одной системы в другую [1].

Для решения вышеперечисленных проблем можно использовать несколько видов интеграции ПО. Разберем каждый из них более подробно.

Интеграция приложений. Интеграция на уровне приложений подразумевает использование готовых функций одних информационных систем другим программным обеспечением. Например, вызов существующего текстового редактора из ERP-системы вместо разработки собственного.

Такой способ интеграции подразумевает обмен сообщениями между различными видами ПО, а также общий доступ к некоторым пользовательским интерфейсам.

На рис. 1 представлена схема взаимодействия при интеграции на уровне API.

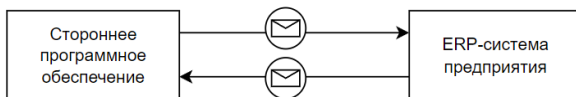


Рис. 1. Схема взаимодействия «Интеграция приложений»

Недостатком интеграции приложений являются возможные ограничения в производительности, а также возможные проблемы совместимости интегрируемых продуктов.

Интеграция данных. При интеграции данных каждый вид ПО напрямую взаимодействует с базой данных предприятия. Каждая информационная система регулярно загружает и получает данные из общего хранилища данных. Загрузка в базу может происходить с помощью API конкретного ПО, которое никак не связано с другими системами [2].

Такой способ интеграции подразумевает универсальный доступ к данным, а также наличие общего хранилища данных.

На рис. 2 представлена схема взаимодействия при интеграции на уровне данных.

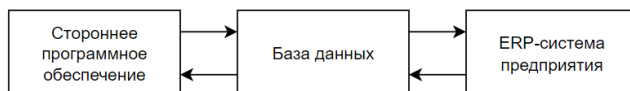


Рис. 2. Схема взаимодействия «Интеграция данных»

Недостатком интеграции данных является возможная утечка всех данных предприятия одновременно, так как данные всех информационных систем предприятия хранятся в одном месте.

Интеграция платформ. Интеграция платформ подразумевает, что все программное обеспечение предприятия имеет доступ в общему API, которое позволяет обмениваться данными, а также сохраняет информацию в БД предприятия. Такой способ интеграции подразумевает виртуализацию, которая позволяет запускать N процедур на M оборудовании, где $N > M$ [2]. На рис. 3 представлена схема взаимодействия при интеграции платформ.

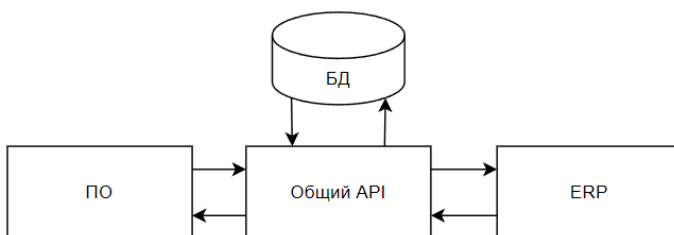


Рис. 3. Схема взаимодействия «Интеграция платформ»

Недостатком интеграции платформ являются достаточно большая сложность разработки интеграционной платформы, а также ограничения в выборе поставщиков и технологий разработки ПО. Интеграцион-

ная платформа не может поддерживать все существующие средства разработки.

Из представленного описания видов интеграции можно сделать вывод, что интеграция платформ представляет собой решение, направленное на упрощение процесса масштабирования системы, однако имеет высокую сложность в реализации и накладывает ограничения на выбор технологий, что ограничивает гибкость системы.

Интеграция приложений направлена на сокращение времени разработки, однако влечет за собой ограничения в производительности и возможные проблемы совместимости при интеграции различных приложений. Интеграция данных, в свою очередь, способствует повышению доступности данных и уменьшению их дублирования, однако несет риск единовременной утечки информации всего предприятия.

В рамках выполнения интеграции ПО на предприятии ООО «Камский Кабель» необходимо интегрировать ERP-систему предприятия с программным обеспечением для взаимодействия с аварийными службами. Программное обеспечение представляет собой мобильное приложение и сервер приложений, реализованный средствами ASP.Net Core на основе архитектуры REST.

ERP-система предприятия использует базу данных под управлением СУБД Oracle. Так как предприятие не имеет собственной интеграционной платформы, а ERP-система имеет закрытый API, доступ к которому невозможно получить с помощью стороннего ПО, необходимо провести интеграцию данных.

Первым этапом интеграции является создание пользователя базы данных и назначение ему прав доступа. Создание пользователя необходимо для обеспечения безопасности при интеграции ПО, доступ пользователю выдается только к необходимым для работы ПО данным, таким образом, часть данных предприятия будет защищена от возможной утечки. SQL-команды для создания пользователя и назначения ему прав доступа к тестовой таблице documents представлены на листинге 1.

Листинг 1 – Создание и настройка пользователя в БД

```
CREATE USER api_user IDENTIFIED BY qwe123  
GRANT SELECT, INSERT, UPDATE ON documents TO  
api_user
```

Следующим этапом интеграции является добавление на сервер приложений функционала, который позволит связать мобильное приложение и базу данных.

Так как ERP-система использует СУБД Oracle, необходимо установить библиотеку Oracle.DataAccess.Client, которая описывает класс OracleConnection. Конструктор данного класса получает входным параметром строку подключения к базе данных, а созданный экземпляр класса отвечает за взаимодействие с базой [3].

Строку подключения необходимо записать в файл appsettings.json. Строка содержит данные пользователя (логин и пароль), название БД, адрес хоста и номер порта, по которым будет происходить подключение. Пример строки подключения представлен на листинге 2.

Листинг 2. Строка подключения к БД

```
"ConnectionStrings": {  
    "DefaultConnection": "SERV-  
ER=(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=*адрес  
Хоста) (PORT=*номерПорта*)) (CONNECT_DATA=(SERVICE_NA  
ME=*имяБазы*)) );uid=api_user;pwd=qwe123;"  
}
```

Следующим этапом интеграции является создание API-контроллера и эндпоинтов, которые получают данные от клиента (в виде HTTP запроса), обрабатывают их и взаимодействуют с базой данных предприятия.

Архитектура REST предполагает применение HTTP запросов, например:

- GET (получение данных);
- POST (добавление данных);
- PUT (изменение данных);
- DELETE (удаление данных) [4].

Пример реализации контроллера Documents и эндпоинта типа GET представлен на листинге 3 [5].

Листинг 3 – Get контроллер для запроса списка документов из БД

```
[ApiController]  
[Route("api/[controller]")]  
public class DocumentsController : ControllerBase  
{  
    [HttpGet("DocWithDate")]  
    public async Task<IActionResult> Get-  
    DocWithDate(DateTime startPoint, DateTime endPoint)  
    {  
        try  
        {
```

```

using (OracleConnection connection = new OracleCon-
nection(connectionString))
{
await connection.Open();
string query = "SELECT ndoc, ceh, fio, ddc FROM
documents WHERE ddc between :startPoint AND
:datePoint";
OracleCommand command = new OracleCommand(query,
connection);
command.Parameters.Add(":startPoint", startPoint);
command.Parameters.Add(":endPoint", endPoint);
var resultList = await com-
mand.executeObjects<Document>();
return Ok(resultList);
}
}catch (Exception ex)
{
return StatusCode(500, "Internal server error");
}
}
}

```

Разработанный метод принимает в виде входных параметров две даты: начало временного промежутка, конец временного промежутка. После получения входных данных сервер открывает подключение к базе данных, запрашивает документы за выбранный промежуток времени и возвращает ответ клиентскому приложению. Ответ содержит в себе код результата и список выбранных документов. Реализация других типов HTTP запросов выглядит аналогичным образом, а данные, помимо получения из БД, можно добавлять, изменять или удалять.

В рамках проделанной работы было изучено понятие интеграции ПО, причины, по которым интеграция становится необходимой. Были описаны проблемы, которые решаются с помощью интеграции, а также разобраны основные виды интеграции программного обеспечения. Была составлена таблица, которая показывает преимущества и недостатки каждого вида интеграции. На примере интеграции ERP-системы предприятия ООО «Камский Кабель» с программным обеспечением для взаимодействия с аварийными службами был продемонстрирован один из вариантов интеграции данных.

Библиографический список

1. Егорян В.В., Калугин А.В. Интеграция программных приложений: Назначение // Столыпинский вестник. – 2022. – № 9.
2. Вульф Б., Хоп Г. Шаблоны интеграции корпоративных приложений. – М.: Диалектика, 2022. – 674 с.
3. Официальная документация Oracle. Oracle Data Provider for .NET Assemblies [Электронный ресурс]. – URL: docs.oracle.com (дата обращения: 03.05.2024).
4. Официальная документация Microsoft. Поддержка HTTP в .NET [Электронный ресурс] – URL: learn.microsoft.com (дата обращения: 05.05.2024).
5. Чамберс Д., Пэкэтт Д. ASP.NET Core. Разработка приложений. – СПб.: Питер, 2017. – 464 с.

Сведения об авторах

Викентьева Ольга Леонидовна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: oleovic@pstu.ru

Бурмантов Егор Павлович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: burmantov2002@gmail.com

М.Н. Радостев, А.В. Тарутин

Пермский национальный исследовательский
политехнический университет, Пермь

ИНТЕГРАЦИЯ СИСТЕМ КАДРОВОГО ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА В HRM-СИСТЕМЫ

В статье определена проблема интеграций систем кадрового электронного документооборота в HRM-системы. Реализованы интеграционные таблицы в РСУБД MS SQL. Приведен алгоритм работы коннектора – прослойки между HRM-системой и КЭДО-системой.

Ключевые слова: интеграция, HRM-система, КЭДО.

M.N. Radostev, A.V. Tarutin

Perm National Research Polytechnic University, Perm

INTEGRATION OF HR ELECTRONIC DOCUMENT MANAGEMENT SYSTEMS INTO HRM SYSTEMS

The article defines the problem of integration of HR electronic document management systems into HRM-systems. Integration tables in MS SQL RDBMS are realized. The algorithm of the connector – a layer between HRM-system and KEDO-system is given.

Keywords: integration, HRM system, KEDO.

В наше время происходит быстрое проникновение цифровых технологий в экономику и общество. Одним из важных шагов этого процесса для компаний является внедрение электронного документооборота (ЭДО) [1].

Кадровый электронный документооборот в сфере трудовых отношений (КЭДО, кадровый ЭДО) – это создание, подписание, использование и хранение документов, связанных с работой, оформленных в электронном виде без дублирования на бумажном носителе [2].

Как правило, в HRM-системах не предусмотрен внутренний КЭДО, для этих целей используются внешние системы. Однако интеграция таких систем с HRM-системой может представлять определенные вызовы, включая необходимость обеспечения совместимости различных платформ.

Данная проблема решается внедрением коннектора – прослойки между HRM-системой и КЕДО-системой. Для коннектора будут созданы интеграционные таблицы для передачи данных в КЕДО.

Реализация интеграционных таблиц в РСУБД MS SQL. В качестве РСУБД была выбрана MS SQL. За основной механизм отвечают следующие таблицы: пользователь, сценарий, задача и связанные с ними таблицы. Схема таблиц показана на рис. 1.

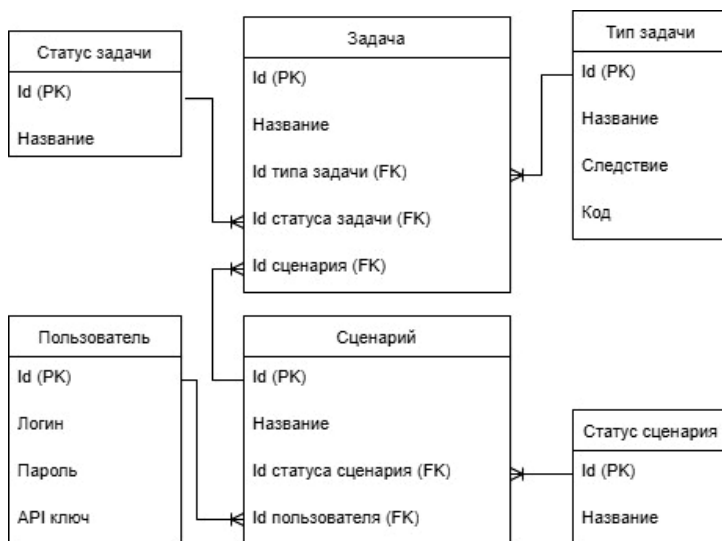


Рис. 1. Реляционная модель основных таблиц коннектора

Всего в основной части коннектора 6 сущностей:

- пользователь – хранит данные, которые используются для подключения к API;
- сценарий – хранит информацию о запусках интеграции с коннектором;
- статус сценария – хранит справочник статусов сценариев (0 – создан, 1 – готов к передаче, 2 – выполняется, 3 – выполнен успешно, 4 – ошибка);
- задача – хранит информацию о вызываемом методе API;
- статус задачи – хранит справочник статусов задач (0 – создан, 1 – готов к передаче, 2 – отправка, 3 – данные отправлены, 4 – ошибка при отправке, 5 – данные обработаны, 6 – ошибка при обработке);

– тип задачи – хранит информацию о всех доступных методах API, их код внутри коннектора и действие, которое надо произвести при успешной передаче (следствие).

Эти таблицы отвечают за работу коннектора. Для формирования данных, отправляемых в API, коннектор необходимо расширить новыми таблицами. Их структура представлена в табл. 1.

Таблица 1

Структура таблиц

Поле	Описание
Id (PK)	Идентификатор
TaskId (FK)	Ключ на таблицу «Задание»
Direction	Направление данных (0 – отправляемые данные, 1 – полученные данные)
...	...
Поля и данные необходимые для метода API	

В качестве примера рассмотрим метод «Подписать заявление». В REST API [3] этот метод имеет следующий URL-адрес: `api/v1/orgId/{orgId}/declaration/{declarationId}/sing`. Также в теле запроса необходимо передавать «ParticipantId». В таком случае интеграционная таблица будет иметь следующий вид, представленный в табл. 2.

Таблица 2

Структура интеграционной таблицы

Поле	Описание
Id (PK)	Идентификатор
TaskId (FK)	Ключ на таблицу «Задание»
Direction	Направление данных (0 – отправляемые данные, 1 – полученные данные)
OrgId	Идентификатор организации
DeclarationId	Идентификатор заявления
ParticipantId	Идентификатор сотрудника

Алгоритм работы коннектора. В основе работы коннектора лежит использование базы данных и API КЭДО-систем.

Схема его работы изображена на рис. 2.

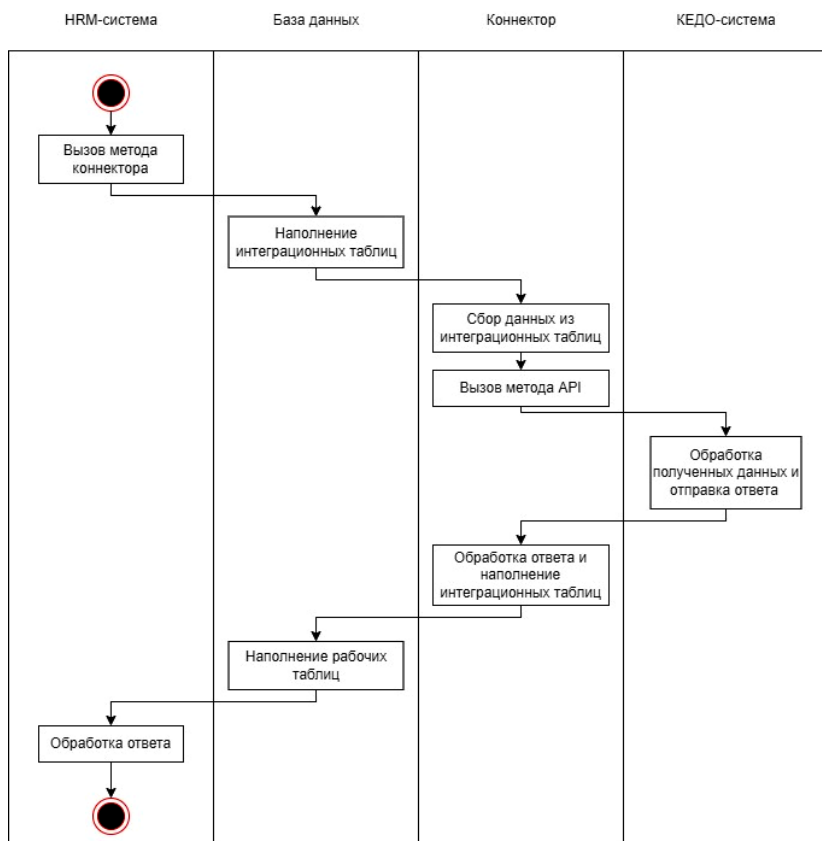


Рис. 2. Алгоритм работы коннектора

HRM-система заполняет интеграционные таблицы либо с помощью внутренних средств, либо с помощью средств базы данных, например, процедур. После этого коннектор отлавливает изменения в интеграционных таблицах и на их основании создает запрос к API КЭДО-систем.

КЭДО-система обрабатывает полученные данные и отправляет ответ, который коннектор раскладывает по интеграционным таблицам. Далее либо база данных раскладывает данные из интеграционных таблиц в рабочие, либо это делается с помощью средств HRM-системы.

Коннектор запускается как фоновый сервис, работающий круглосуточно. На листинге приведен алгоритм коннектора на псевдокоде по поиску новых сценариев на отправку данных.

Листинг – Поиск новых сценариев и вызов метода API

Алг

Ввод scenCount – Максимальное количество сценариев в обработке

scenarios := Данные из таблицы «Сценарий», по условию Status = 1, ограничиваясь по scenCount

Установить записям из scenarios значение 2 в поле Status

tasks := Данные из таблицы «Задача» по связке scenarioId из scenarios

Для каждой задачи в списке tasks

НЦ

request = Данные из интеграционной таблицы по связке TaskId и Direction = 0

Отправка данных из объекта request в API-КЕДО

response := Данные, полученные от API-КЕДО

Если response не содержит ошибок

Заполнение интеграционных таблиц на основе response с полем Direction := 1

Выполнение действия из поля «Следствие» таблицы Task

Конец Если

Иначе

Вывод ошибки

Конец Иначе

КЦ

Кон

Заключение. Проблему интеграции HRM-систем и КЭДО-систем можно решить благодаря разработанному коннектору. Данный коннектор расширяемый, не зависящий от конкретных версий используемых систем и обеспечивающий гибкую интеграцию между ними. Он способен работать с различными форматами данных, а также автоматизировать передачу информации между системами, учитывая их специфические требования.

Библиографический список

1. Внедрение электронного документооборота как этап цифровизации экономического субъекта [Электронный ресурс]. – URL: <https://cyberleninka.ru/article/n/vnedrenie-elektronnogo-dokumentooborota-kak-etap-tsifrovizatsii-ekonomicheskogo-subekta> (дата обращения: 26.05.2024).

2. Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 06.04.2024), Статья 22.1. Электронный документооборот в сфере трудовых отношений [Электронный ресурс]. – URL: https://www.consultant.ru/document/cons_doc_LAW_34683/6dda95f9b6acf89a431a0737944b9443695b9df9/ (дата обращения: 26.05.2024).

3. HeadREST: A Specification Language for RESTful APIs / T.V. Vasco, M. Francisco, L. Antónia, B. Nuno. – 2019. DOI: 10.1007/978-3-030-21485-2_23

Сведения об авторах

Радостев Михаил Николаевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-36, г. Пермь, e-mail: radm258@gmail.com

Тарутин Анатолий Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: itas-pnpu@yandex.ru

А.А. Халилов, Д.Б. Кузнецов

Пермский национальный исследовательский
политехнический университет, Пермь

РЕАЛИЗАЦИЯ МЕХАНИЗМА ВЫБОРА ССЫЛКИ В ИНТЕРФЕЙСЕ СИСТЕМЫ УПРАВЛЕНИЯ НОРМАТИВНО-СПРАВОЧНОЙ ИНФОРМАЦИЕЙ

Исследуется возможность реализации механизма выбора ссылки в интерфейсе системы управления нормативно-справочной информацией. Механизм должен позволить пользователю гибко настраивать ссылку на таблицу, благодаря работе через интерфейс программы. В статье рассматривается проектирование алгоритма выбора ссылки и интерфейса, а также его реализация в клиентском интерфейсе.

Ключевые слова: ссылка, таблица, система управления нормативно-справочной информацией, интерфейс, атрибут.

A.A. Khalilov, D.B. Kuznetsov

Perm National Research Polytechnic University, Perm

IMPLEMENTATION OF THE LINK SELECTION MECHANISM IN THE INTERFACE OF THE REGULATORY INFORMATION MANAGEMENT SYSTEM

This article explores the possibility of implementing a link selection mechanism in the interface of the regulatory reference information management system. The mechanism should allow the user to flexibly configure the link to the table, thanks to working through the program interface. The article discusses the design of the link and interface selection algorithm, as well as its implementation in the client interface.

Keywords: link, table, regulatory information management system, interface, attribute.

Системы управления НСИ (далее СУНСИ) предназначены для упорядочивания и структурирования данных, которые используются в различных бизнес-процессах, обеспечивая их точность, актуальность и доступность. Ссылки на таблицы играют важную роль в организации и навигации по данным. Правильная реализация таких ссылок позволяет пользователям связывать таблицы между собой, которые могут находиться в едином или разных каталогах. Данный механизм предоставляет пользователям возможность создания ссылки на имеющуюся

нормативно-справочную информацию без надобности в дублировании данных из одной таблицы в другую. Также это обеспечивает создание сложных связей данных, чего невозможно достичь, храня данные в обычных файлах. Это особенно актуально в крупных организациях, где объем данных может быть значительным.

Для начала нужно спроектировать механизм выбора ссылок. В СУНСИ имеется структура таблиц, которая состоит из редактируемых атрибутов. Реализуемый механизм должен позволять выбрать ссылочные атрибуты из любой таблицы внутри каталогов для ссылочного атрибута. На рис. 1 представлена схема данного механизма.

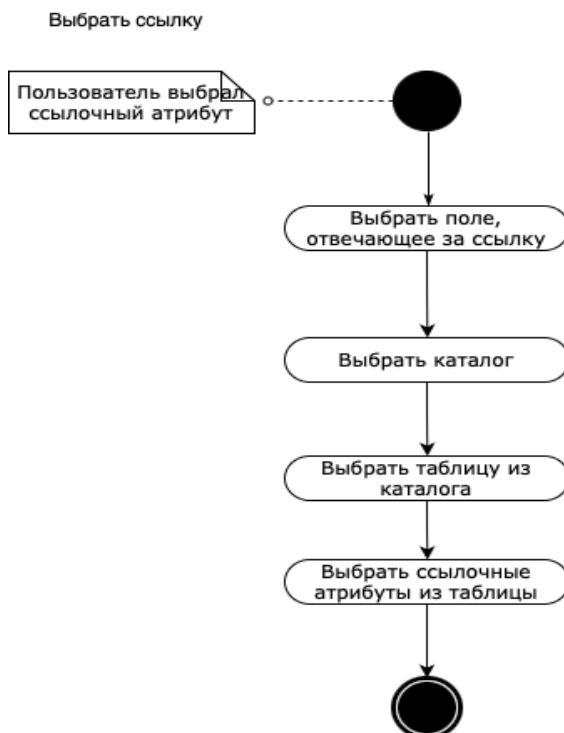


Рис. 1. Схема выбора ссылки

Исходя из приведенного в рис. 1 алгоритма, можно спроектировать будущий интерфейс для механизма выбора ссылки. На рис. 2 представлен макет интерфейса, на котором был добавлен ссылочный атрибут с полем для хранения ссылки. В макете хранение ссылки вынесено

в единственное поле, что более привычно для знакомых с РСУБД и внешними ключами.

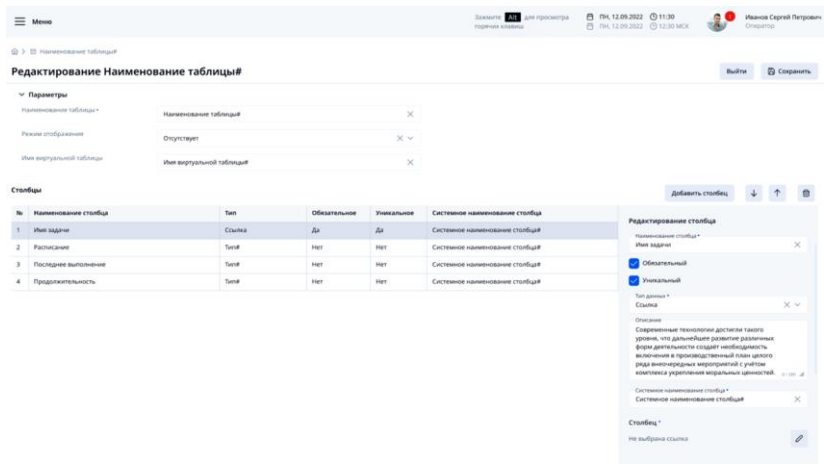


Рис. 2. Макет интерфейса ссылочного атрибута

Многоэтапный выбор ссылки был вынесен в модальное окно, представленное на рис. 3, чтобы не загромождать интерфейс.

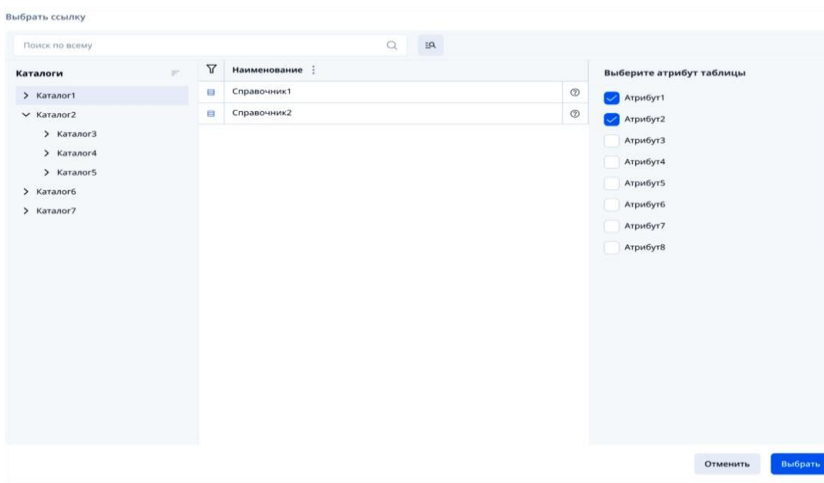


Рис. 3. Макет модального окна выбора ссылки

На рис. 3 представлен макет гибкого выбора ссылки, где выбирается каталог из дерева каталогов, далее – справочник (таблица) из вы-

бранного каталога и последними – ссылочные атрибуты выбранной таблицы. После завершения выбора ссылки пользователю предлагается нажать кнопку «Выбрать», чтобы разместить выбранные параметры в поле ссылочного атрибута

Реализация поля ссылочного атрибута и механизма выбора ссылки лежит на фреймворке Angular, а точнее – механизме фреймворка Reactive Forms [1]. В последующем повествовании будут использоваться термины из Reactive Forms. Чтобы понимать назначение данного механизма фреймворка, ниже будет приведено краткое пояснение его составляющих:

- FormGroup – модель формы в интерфейсе;
- FormControl – функциональный элемент формы (далее контрол);
- ControlValueAccessor – интерфейс или контракт, который нужно реализовать пользовательским контролам.

Поле ссылочного атрибута с макета на рис. 2, хранящее ссылку, является контролом формы FormControl. В листинге 1 представлен FormControl для ссылочного атрибута.

Листинг 1 – Псевдокод FormControl ссылочного атрибута

```
variable refControl = FormControl({  
    reference_id: string,  
    reference_attributes: string[],  
    reference_table_name: string  
})
```

В листинге 1 представлены поля ссылки на таблицу, ссылочные атрибуты и название ссылочной таблицы соответственно.

Контрол ссылки являются пользовательским функциональным элементом ControlValueAccessor. Контрол представляет собой компонент, класс которого реализует контракт интерфейса ControlValueAccessor и логику работы с модальным окном, а внутри HTML-файла код интерфейса поля с макета на рис. 2. По нажатии на кнопку в интерфейсе появляется модальное окно, в котором пользователь проделывает логику выбора ссылки со схемы на рис. 1, после чего инициализируется FormControl из листинга 1. В листинге 2 представлен псевдокод инициализации FormControl из модального окна.

Листинг 2 – Псевдокод инициализации FormControl из модального окна

```
setValue(reference) {  
    onValueChange(reference)  
}
```

В листинге 2 метод `onValueChanged` является зарегистрированным событием для отправки данных изнутри пользовательского контроля

Таким образом, был рассмотрен подход к реализации выбора ссылки в интерфейсе СУНСИ. Был спроектирован алгоритм выбора ссылки, на основе которого были построены макеты будущего интерфейса для механизма выбора ссылки. Был реализован интерфейс для выбора ссылки на основе спроектированных макетов.

Библиографический список

1. Angular – Официальная страница фреймворка Angular с документацией [Электронный ресурс]. – URL: <https://angular.dev/> (дата обращения: 14.04.2024).

Сведения об авторах

Халилов Артур Азатович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: therock65frostik@gmail.com

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@mail.ru

В.А. Мехоношин, Д.Б. Кузнецов

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА СИСТЕМЫ ОПОВЕЩЕНИЯ СОТРУДНИКОВ КОМПАНИИ

Своевременное информирование сотрудников имеет важное значение для эффективной работы компании. Пользователи могут использовать систему уведомлений сотрудников для уведомления своих коллег, предоставляя им информацию о предстоящих событиях и мероприятиях компании. В данной статье рассмотрены процессы проектирования и разработки такой системы.

Ключевые слова: программная система, информационная система, программное обеспечение, анализ, разработка.

V.A. Mekhonoshin, D.B. Kuznetsov

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF THE SYSTEM OF NOTIFICATION OF EMPLOYEES OF THE COMPANY

Timely informing of employees is important for effective work of the company. Users can use the employee notification system to notify their colleagues by providing them with information about upcoming events and activities of the company. This article will discuss the main issues that need to be addressed when creating such a system.

Keywords: software system, information system, software, analysis, development.

Система оповещения сотрудников – это инструмент, используемый для передачи сотрудникам компании сообщений о различных важных событиях, таких как встречи, задания, происшествия и т.д. Данная система предназначена для компаний, в которых важна скорость реакции и взаимодействия сотрудников.

Основной задачей системы оповещения сотрудников являются повышение эффективности распространения информации и контроле информированности сотрудников. Основываясь на «реакциях», отправитель должен быть способен определить скорость реакции пользователя на конкретное уведомление.

Под «реакциями» подразумеваются кнопки, расположенные под уведомлениями и дающие возможность максимально быстро отреагировать на ту или иную ситуацию.

Создаваемый продукт должен быть конкурентоспособен, поскольку создается в целях дальнейшей продажи. Для оценки конкурентоспособности и возможной корректировки требований, выдвигаемых к системе, были проанализированы популярные решения, предоставляющие возможности для оповещения пользователей в корпоративной среде. Были рассмотрены следующие продукты: Projecto, MyChat и Microsoft Teams. Ни один из рассмотренных продуктов не предоставил возможностей гибкой настройки реакций на создаваемые уведомления, а их возможности контроля осведомленности сотрудников были весьма ограничены. Поэтому было принято решение о целесообразности создания системы оповещения.

После анализа требований заказчика и конкурентных продуктов были выдвинуты следующие требования:

- возможность рассылки;
- возможность реакции;
- возможность контроля осведомленности;
- доступность на множестве платформ;
- возможность интеграции;
- гибкость настроек;
- возможность написания комментариев.

На основе данных требований была составлена диаграмма, демонстрирующая функциональные возможности разрабатываемого продукта. Диаграмма представлена на рис. 1.

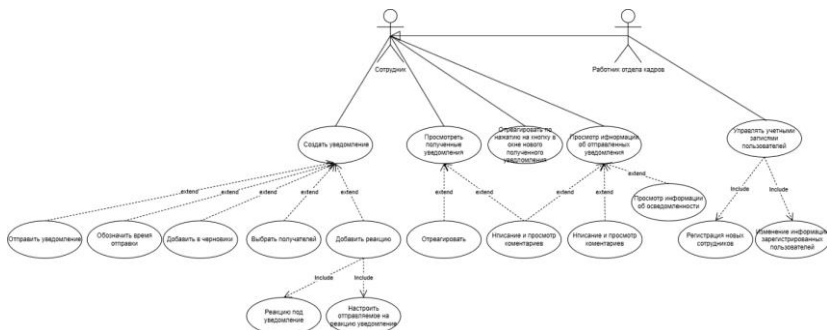


Рис. 1. Функциональные возможности

В системе были выделены 2 основные роли: сотрудник и работник отдела кадров. Каждый сотрудник имеет возможность создания уведомлений, просмотра списка полученных уведомлений и реакции на пришедшие уведомления по нажатию на кнопки во всплывающем окне. А сотрудник отдела кадров в дополнение имеет возможность управления учетными записями пользователей.

Одним из главных требований при разработке такой системы является доступность на множестве платформ и использование открытого программного обеспечения. Также во многих компаниях установлен запрет на установку дополнительного ПО и доступ во внешние сети. Основываясь на этих требованиях, была выбрана архитектура системы, показанная на рис. 2.

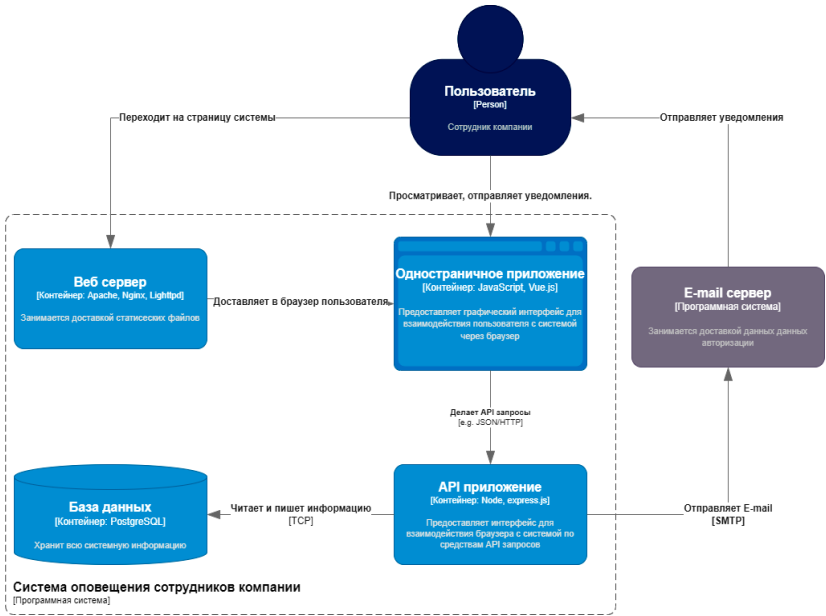


Рис. 2. Диаграмма архитектуры системы

Система является клиент-серверным веб-приложением, обеспечивающим доступ к ней с любого устройства, имеющего доступ к Интернету или внутренней сети. А все использованные при ее разработке программное обеспечение и фреймворки являются открытыми.

Express – открытый фреймворк предоставляющий все необходимое для разработки веб-приложений [1].

Node – бесплатная, открытая кроссплатформенная среда выполнения, позволяющая создавать серверы, веб-приложения и скрипты [2].

PostgreSQL – свободная объектно-реляционная СУБД [3].

Vue – открытый фреймворк для создания пользовательских веб-интерфейсов [4].

На основе функциональных возможностей и требований к системе была разработана база данных, хранящая в себе используемую системой информацию. ER диаграмма базы данных была показана на рис. 3.

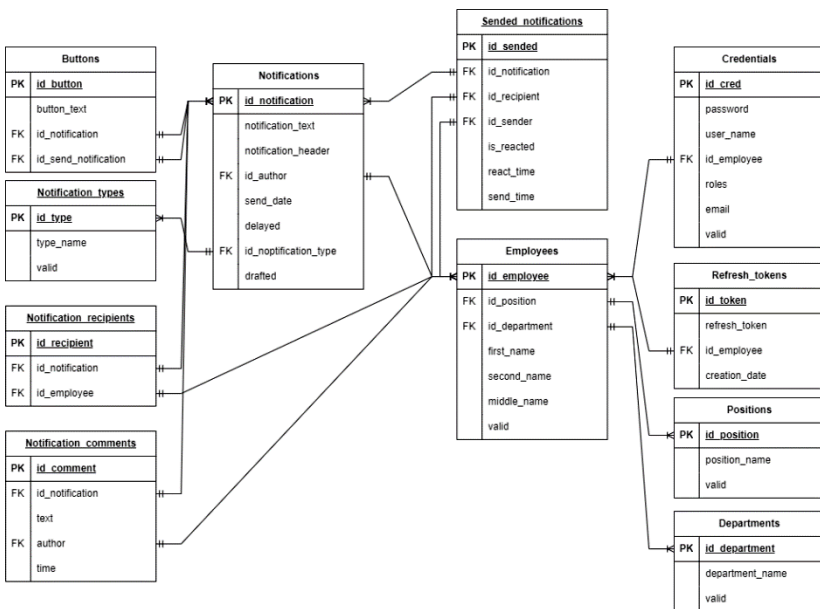


Рис. 3. ER диаграмма базы данных

На основе всего приведенного выше была разработана система оповещения сотрудников компании. Пример ее интерфейса показан на рис. 4 и 5.

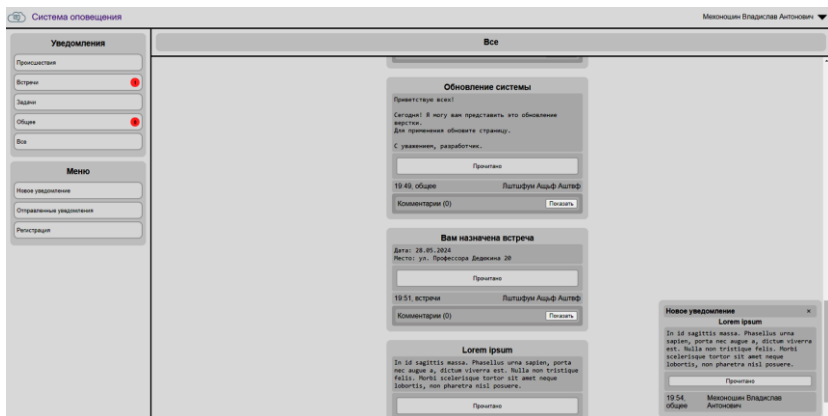


Рис. 4. Веб-интерфейс приложения

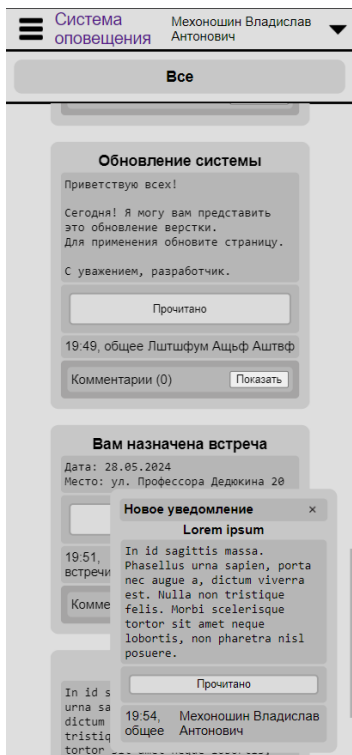


Рис. 5. Мобильная версия веб-интерфейса

Разработанная система далеко не идеальна. В дальнейшем планируется работа над интерфейсом веб-приложения, расширением возможностей рассылки оповещений и предоставление возможности прикрепления документов к уведомлениям.

Разработка системы оповещения сотрудников компании является сложным процессом, который требует тщательного планирования. Правильно спроектированная и реализованная система позволит значительно повысить оперативность и эффективность коммуникаций внутри компании, что в конечном итоге приведет к улучшению общего уровня организации.

Библиографический список

1. Express – web application framework [Электронный ресурс]. – URL: <https://expressjs.com> (дата обращения: 24.05.2024).
2. Node.js – Run JavaScript Everywher [Электронный ресурс] // About. – URL: <https://nodejs.org/en/about> (дата обращения: 24.05.2024).
3. PostgreSQL [Электронный ресурс] // About. – URL: <https://www.postgresql.org/about/> (дата обращения: 24.05.2024).
4. Vue.js – The Progressive JavaScript Framework [Электронный ресурс] // Introduction. – URL: <https://vuejs.org/guide/introduction.html> (дата обращения: 24.05.2024).

Сведения об авторах

Мехоношин Владислав Антонович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-2б, г. Пермь, e-mail: vladmexon@gmail.com

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@mail.ru

В.А. Томилов, А.А. Петренко

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ИНТЕРНЕТ-МАГАЗИНА КОМПЬЮТЕРНОЙ ТЕХНИКИ ООО «КАПИТАН»

Рассматривается разработка интернет-магазина для продажи товаров через онлайн-платформу. Центральной частью проекта является создание программного обеспечения, которое обеспечивает удобный и интуитивно понятный процесс совершения покупок для пользователей. Проведен анализ существующего решения бизнес-процесса организации и выявлены недостатки существующих решений. Рассмотрены методы реализации интернет-магазина, проектирование структуры интернет-магазина и выбор средств разработки. Показаны некоторые фрагменты реализации функциональных элементов, позволяющие пользоваться основными функциями интернет-магазина.

Ключевые слова: интернет-магазин, онлайн-платформа, онлайн-заказ, функциональные блоки.

V.A. Tomilov, A.A. Petrenko

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF AN ONLINE STORE FOR COMPUTER EQUIPMENT LLC “CAPTAIN”

The article discusses the development of an online store for the sale of goods through an online platform. The central part of the project is the creation of software that provides a convenient and intuitive shopping process for users. An analysis of the existing solution of the business process of the organization was carried out and the shortcomings of existing solutions were identified. The methods of implementing an online store, designing the structure of an online store and choosing development tools are considered. Some fragments of the implementation of functional elements that allow you to use the basic functions of an online store are shown.

Keywords: online store, online platform, online ordering, functional blocks.

С развитием информационных технологий и электронной коммерции интернет-магазины становятся все более востребованными среди потребителей. В современном мире покупатели все чаще предпочитают совершать покупки онлайн из-за удобства, широкого ассортимента товаров и возможности сравнения цен. Яндекс и GfK Rus по результатам ежегодного опроса российской аудитории интернет-магазинов приво-

дят следующие показатели. Среди городских жителей 16–55 лет их доля уже превысила две трети, почти 90 % покупателей делают заказы минимум раз в месяц, более трети – минимум раз в неделю [1]. Этот тренд приводит к изменениям в привычных схемах покупок и деловых моделях компаний.

На данный момент бизнес-процесс организации выглядит следующим образом. Клиент связывается с менеджером и отправляет запрос на почту с деталями заказа. После обсуждения и оплаты менеджеры иницируют документ в системе 1С, проверяют запасы и делают заказ у поставщиков. После получения товаров они подготавливают их к отправке, инженеры при необходимости выполняют сборку или настройку. После завершения этапов менеджер согласовывает с клиентом детали доставки или самовывоза. Можно выделить существенные недостатки и сформулировать функциональные требования к системе.

Клиенты могут связываться только через электронную почту или напрямую с менеджерами, что снижает удобство. Отсутствие онлайн-чата или мобильного приложения также ограничивает доступные каналы связи. В связи с чем требуется мультиканальная коммуникация: система должна обеспечивать возможность взаимодействия с клиентами через различные каналы связи, включая электронную почту, онлайн-чат на веб-сайте и мобильное приложение.

Операции по обработке заказов в значительной степени выполняются вручную, что может привести к ошибкам в данных и задержкам в обработке. Это можно исправить за счет онлайн-оформления заказов: покупатели должны иметь возможность удобно и интуитивно оформлять заказ через веб-сайт или мобильное приложение, выбирая товары, указывая адрес доставки и осуществляя оплату.

Клиенты имеют ограниченную возможность отслеживать статус своих заказов и получать информацию о выполнении заказа, что может создавать неудобства. Для решения этой проблемы требуется онлайн-отслеживание заказов: покупатели должны иметь возможность отслеживать статус заказов онлайн через личный кабинет, а также система должна автоматически уведомлять покупателей о статусе и изменениях в их заказах.

В качестве метода решения были выбраны следующие компоненты:

1. Веб-сервер играет ключевую роль в обеспечении взаимодействия между веб-приложением и базой данных предприятия. Он отве-

чает за обработку запросов, обновление данных, а также обеспечивает безопасность и надежность работы системы.

2. Веб-приложение обеспечивает эффективное взаимодействие с системой через веб-браузер, что делает его удобным и доступным для пользователей без необходимости установки дополнительного ПО.

3. Мобильное приложение обеспечивает удобство использования сервиса на мобильных устройствах, что позволяет пользователям взаимодействовать с ним в любое время и в любом месте.

В ходе анализа предметной области, требований к системе и методов решений была спроектирована следующая архитектура системы, представленная на рис. 1.

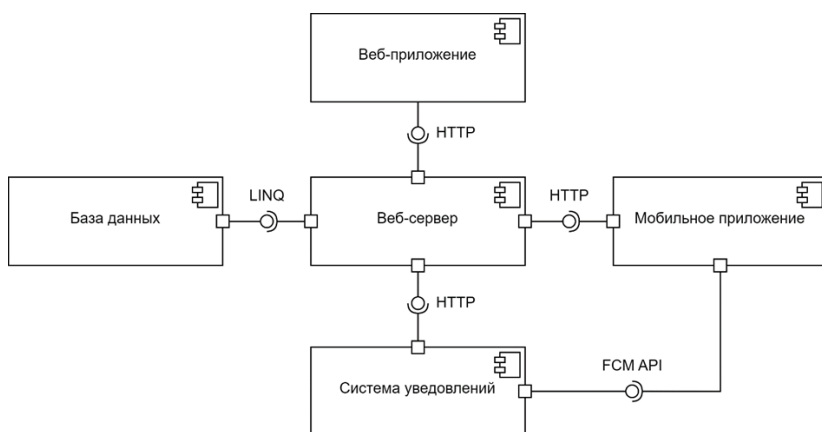


Рис. 1. Архитектура интернет-магазина

Средством для разработки базы данных была выбрана SQL Server – реляционная система управления базами данных (СУБД), разработанная и поддерживаемая корпорацией Microsoft. Это надежное и масштабируемое решение, которое предоставляет широкие возможности для хранения, организации и управления данными [2].

Средством для разработки веб-сервера была выбрана технология ASP.NET MVC (Model-View-Controller). Архитектурный шаблон MVC используется для построения веб-приложений и обеспечивает разделение приложения на три основных компонента: модель (Model), представление (View) и контроллер (Controller). Модели отвечают за бизнес-логику и управление данными, представления отображают инфор-

мацию пользователю, а контроллеры управляют взаимодействием между моделями и представлениями [3].

В данной системе взаимодействие между БД SQL Server и веб-сервером осуществляется через LINQ-запросы. Взаимодействие моделей в ASP.NET MVC с базой данных SQL Server осуществляется с помощью Entity Framework (EF), который позволяет работать с базой данных как с объектами C#. При использовании EF можно создавать LINQ-запросы, которые компилируются в SQL-запросы и выполняются на сервере базы данных [4]. Пример такого запроса представлен в листинге 1.

Листинг 1 – Получение списка товаров через LINQ-запрос

```
public class Product
{
    //Поля объекта "Товар"
}

public class KapitanDBContext : DbContext
{
    public KapitanDBContext(DbContextOptions<KapitanDBContext> options) :
base(options) { }
    public DbSet<Product> product { get; set; }
}
var products = db_context.Products.ToList();

public class CatalogPageController : Controller
{
    KapitanDBContext db_context;
    public CatalogPageController(KapitanDBContext
context)
    {
        db_context = context;
    }
    var products = db_context.product.ToList(); //
LINQ-запрос для получения всех продуктов
}
```

Средством для разработки веб-приложения была выбрана технология разработки с использованием HTML, CSS и JavaScript для создания

пользовательского интерфейса и взаимодействия с пользователем в веб-браузере. Эти технологии позволяют создавать динамические и интерактивные веб-страницы, обеспечивая удобство использования приложения. Использование HTML для структурирования содержимого, CSS для стилизации и JavaScript для добавления интерактивности [5].

Взаимодействие веб-сервера и веб-приложения происходит за счет средств ASP.NET MVC и JS-скриптов. Пользователь на веб-странице нажимает кнопку, что вызывает JavaScript-функцию, которая отправляет AJAX-запрос на серверный контроллер. Контроллер обрабатывает запрос, использует модель для выполнения LINQ-запроса к базе данных SQL Server и возвращает данные в формате JSON. JavaScript-функция на клиентской стороне получает ответ от сервера и обновляет соответствующую часть веб-страницы, отображая полученные данные пользователю [6]. Пример взаимодействия представлен в листинге 2.

Листинг 2 – Динамическое обновление категорий товаров

JS скрипт:

```
function get_type_1_list() {
    docu-
ment.querySelector('.select.type_1_select').innerHTML
= ``;
    get_values();
    $.ajax({
        url: '/CatalogPage/GetType1list',
        type: 'GET',
        dataType: 'json',
        success: function (data) {
            let newHTML = ``;
            newHTML += `<option value="0" selected
disabled>Выберите тип</option>`;
            data.forEach(function (entity) {
                newHTML += `<option value="${entity[type
+ '_id']}">${entity[type + '_name']}</option>`;
            });
            var select = docu-
ment.querySelector('.select.' + type + '_select');
            select.innerHTML = newHTML;
        },
        error: function (xhr, status, error) {
```

```

        console.error('Произошла ошибка:',
xhr.status);
    }
    });
}

```

Контроллер:

```

public async Task<IActionResult> GetType1list(string textInput, int type1Value, int
type2Value, int type3Value, int vendorValue, bool
inStock, string priceFrom, string priceTo)
{
    List<SqlParameter> parameters = new
List<SqlParameter>();
    List<Type_1> type_1_list = new
List<Type_1>();
    string sqlRow = GetSqlRow(parameters, textInput, type1Value, type2Value, type3Value, vendorValue, inStock, priceFrom, priceTo);
    var query =
db_context.stock.FromSqlRaw(sqlRow, parameters.ToArray());
    type_1_list = await query.Select(s => new
Type_1
        {
            type_1_id = s.type_1_id,
            type_1_name =
s.type_1_name,
        })
        .Distinct()
        .ToListAsync();
    return Json(type_1_list);
}

```

Средством для разработки мобильного приложения были выбраны инструменты и библиотеки Xamarin. Они предоставляют уникальную возможность создания кроссплатформенных приложений, которые могут работать как на устройствах iOS, так и на устройствах Android, используя общий код на языке программирования C# [7]. Представления веб-приложения полностью реализованы с помощью адаптивной верст-

ки, при которой интерфейс автоматически подстраивается под размер экрана и устройство, на котором отображается сайт, поэтому было решено встроить веб-приложение в мобильное приложение, предоставляя доступ к нативным функциям устройства через веб-технологии. Наиболее подходящим инструментом для этого является использование Xamarin.Forms с элементом WebView. Вывод веб-страницы в мобильном приложении представлен в листинге 3.

Листинг 3 – Отображение сайта в мобильном приложении

```
using Xamarin.Forms;
namespace Kapitan
{
    public class WebViewPage : ContentPage
    {
        public WebViewPage()
        {
            var webView = new WebView
            {
                Source = "https://your-web-
application-url.com",
                VerticalOptions = LayoutOp-
tions.FillAndExpand,
                HorizontalOptions = LayoutOp-
tions.FillAndExpand
            };
            Content = new StackLayout
            {
                Children = { webView }
            };
        }
    }
}
```

Таким образом, был разработан интернет-магазин компьютерной техники. Он представляет собой современное и удобное решение для покупателей компьютерной техники, обеспечивая комфортное и надежное взаимодействие как через веб-браузер, так и через мобильные устройства.

В дальнейшем планируется расширить функционал интернет-магазина. Можно реализовать раздел отзывов и рекомендаций пользователей, а также системы рейтинга товаров на основе отзывов. Также

можно внедрить системы искусственного интеллекта для формирования предложений и рекомендованных товаров пользователю на основе его предпочтений и заказов.

Библиографический список

1. Развитие онлайн-торговли в России [Электронный ресурс]. – URL: <https://yandex.ru/company/researches/2024/ecomdash> (дата обращения: 27.05.2024).

2. Что такое SQL Server? [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/sql/sql-server/what-is-sql-server?view=sql-server-ver16> (дата обращения: 27.05.2024).

3. Изучаем ASP.NET MVC 5 [Электронный ресурс]. – URL: https://professorweb.ru/my/ASP_NET/mvc/level1 (дата обращения: 27.05.2024).

4. Запросы к данным – EF Core [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/ef/core/querying> (дата обращения: 27.05.2024).

5. Технологии для разработки сайта: что это, какие наиболее популярны и как выбрать [Электронный ресурс]. – URL: <https://iampm.club/blog/tehnologii-dlya-razrabotki-sajta-cto-eto-kakie-naibolee-populyarny-i-kak-vybrat> (дата обращения: 27.05.2024).

6. Работа с Ajax в ASP.NET MVC [Электронный ресурс]. – URL: https://professorweb.ru/my/ASP_NET/mvc/level6/6_8.php (дата обращения: 27.05.2024).

7. Что такое Xamarin? [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/previous-versions/xamarin/get-started/what-is-xamarin> (дата обращения: 27.05.2024).

Сведения об авторах

Томилов Владислав Алексеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПИС-20-16, г. Пермь, e-mail: tomilov.v.a@yandex.ru

Петренко Александр Анатольевич – кандидат технических наук, доцент, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: hawk321@mail.ru

А.А. Циреньщиков, Е.В. Долгова

Пермский национальный исследовательский
политехнический университет, Пермь

МОДЕЛИРОВАНИЕ СТРОПАЛЬЩИКА ДЛЯ ТРЕНАЖЕРА ПОРТАЛЬНОГО КРАНА В СРЕДЕ UNREAL ENGINE 5

В этой статье исследуется возможность моделирования стропальщика для тренажера портального крана в среде Unreal Engine 5. Рассмотрены объекты «Blueprint class Character», «AiController» и «Behavior tree» среды Unreal Engine 5, с помощью которых можно создавать динамические объекты.

Ключевые слова: Unreal Engine, AiController, стропальщик, Blueprint class Character, модель.

A.A. Tsirenschikov, E.V. Dolgova

Perm National Research Polytechnic University, Perm

SIMULATION OF A SLINGER FOR A PORTAL CRANE SIMULATOR IN THE UNREAL ENGINE 5 ENVIRONMENT

This article explores the possibility of modeling a slinger for a gantry crane simulator in the Unreal Engine 5 environment. The objects "Blueprint class Character", "AiController" and "Behavior tree" of the Unreal Engine 5 environment with which you can create dynamic objects are considered.

Keywords: Unreal Engine, AiController, slinger, Blueprint class Character, model.

Портальный кран – это крупное и сложное устройство для переноса различных грузов. Управляют такими машинами операторы, которые перед началом работы должны освоить правила и принципы управления краном. Процесс обучения на настоящем кране не является лучшим решением этой проблемы, так как использовать для этого реальный кран может быть опасно и дорого. Во время обучения будущего оператора могут возникнуть аварии, а производственный процесс будет простаивать. В связи с этим более грамотным решением будет использование специальных компьютерных тренажеров. Это снизит как стоимость обучения, так и риски травм, аварий в процессе обучения [1].

Компьютерный тренажер портального крана должен реалистично эмулировать работу настоящего крана, а также все, что его окружает. Помимо физики и управления самого крана, важно воссоздать динами-

ческие объекты, такие как «стропальщик», «транспорт» и т.д., и само виртуальное окружение, которое должно быть максимально приближено к реальности, чтобы опыт, полученный оператором на тренажере, мог быть использован в реальных условиях.

Одной из важнейших частей тренажера является динамический объект «стропальщик». Стropальщик – это рабочий, который выполняет строповку (обвязку) грузов для производства погрузочно-разгрузочных работ с использованием грузоподъемных механизмов [2]. В тренажере должна быть реализована элементарная логика поведения стропальщика для правдоподобной симуляции рабочего процесса крановщика.

Обязанности стропальщика включают в себя:

- обвязку и зацепку различных грузов для их подъема и перемещения;
- укладку (установку) груза в проектное положение и снятие грузозахватных приспособлений (расстроповка);
- подачу сигналов крановщику (машинисту) [2].

Модель поведения стропальщика изображена на рис. 1.

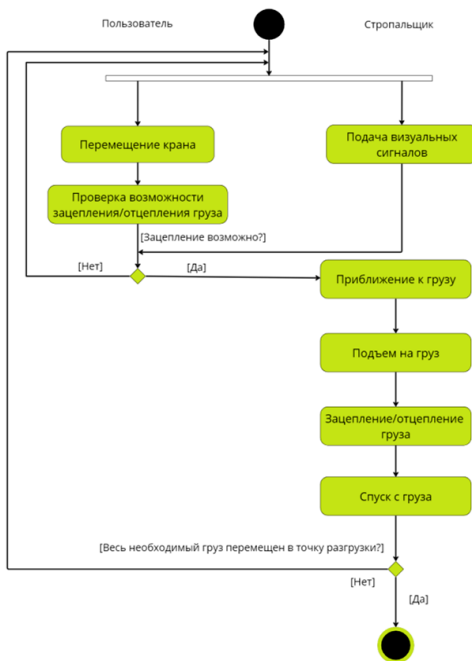


Рис. 1. Модель поведения стропальщика

Для реализации динамической модели стропальщика на Unreal engine 5 можно создать Blueprint class с родительским классом Character. Это позволит привязать к одному объекту 3D-модель стропальщика, анимацию и логику его поведения.

3D-Модель стропальщика вместе со скелетом и текстурами можно импортировать из внешних источников и представить в классе Skeletal Mesh, как один объект (рис. 2).

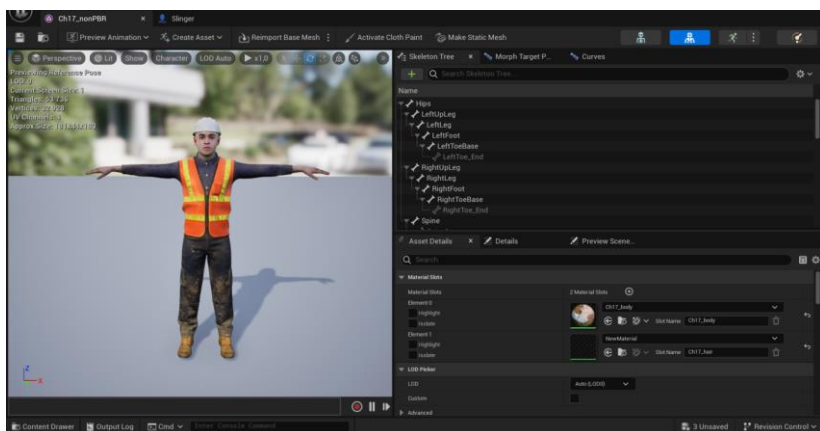


Рис. 2. Окно работы с классом Skeletal Mesh

Созданную таким образом модель можно применить в параметре Skeletal Mesh Asset в основном классе Character.

Аналогично в параметрах этого Blueprint класса можно задать анимации для модели стропальщика.

Логiku поведения стропальщика можно реализовать в Blueprint классе наследуемого от класса AiController. В нем можно с помощью языка Blueprint описывать логику поведения стропальщика.

Blueprints Visual Scripting – это визуальный язык программирования, который представляет объекты, функции и различные компоненты в виде блоков, с которыми можно удобно взаимодействовать, перетаскивая и соединяя друг с другом. Это ускоряет процесс разработки, а также понижает требования к опыту разработчика. Основой для этих объектов являются C++ классы [3].

Созданный класс также можно указать в параметре основного класса Ai Controller Class. Пример использования на рис. 3.

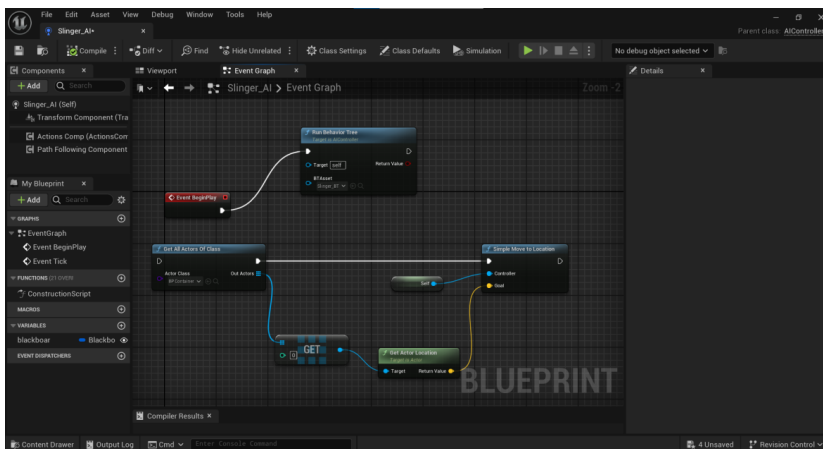


Рис. 3. Окно работы с AiController

Вспомогательным инструментом для описания поведения персонажей является объект Behavior tree. Этот объект – более удобный и быстрый способ для данной задачи. В нем выполняемые действия расставляются в виде дерева, где можно задать логику и порядок их исполнения (рис. 4). Данный объект можно использовать в виде функции в программе на Blueprint, что облегчает создание сложной логики поведения персонажей.

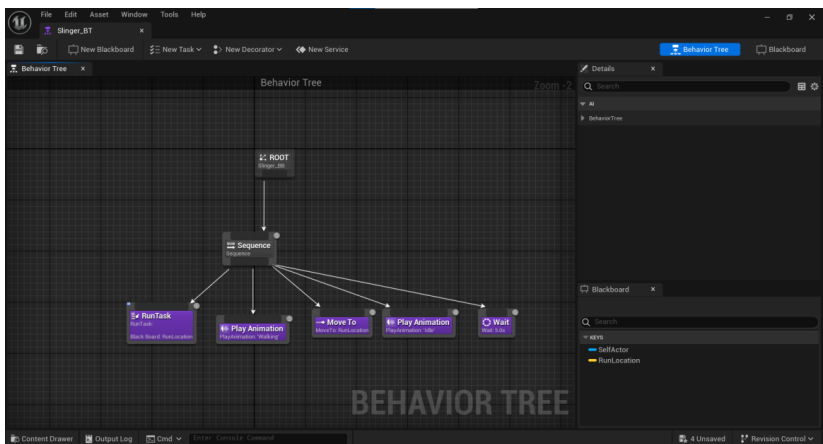


Рис. 4. Окно работы с AiController

Таким образом игровой движок Unreal Engine 5 предоставляет удобный и многофункциональный инструмент для создания динамических объектов, в частности стропальщика. Используя BluePrint class Character за основу, можно работать с 3D-моделью и ее скелетом – с помощью Skeletal Mesh, с логикой и поведением с помощью AiController и Behavior tree.

Библиографический список

1. Файзрахманов Р.А., Хабибулин А.Ф. Проектирование и разработка тренажерного комплекса оператора порталного крана // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2014. – № 9. – С. 80–92.
2. Игумнов С.Г. Стropальщик. Грузоподъемные краны и грузозахватные приспособления: учеб. пособие. – М.: Академия, 2007. – 64 с.
3. Unreal Engine 5 Documentation. – URL: <https://docs.unrealengine.com/5.0/en-US>

Сведения об авторах

Циреньщиков Александр Андреевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-1б, г. Пермь, e-mail: sasha.tsirenhikov@gmail.com

Долгова Елена Владимировна – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: elena@dolgova.space

А.В. Азмагулов, Д.Б. Кузнецов

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «ШАХМАТНЫЙ ТРЕНАЖЕР»

Рассматривается разработка программного обеспечения «Шахматный тренажер». Описаны основные блоки реализации программного продукта совместно с шахматным движком. Реализована модель функциональных возможностей пользователя для «Шахматного тренажера». Описан принцип построения FEN-нотации. Описан принцип взаимодействия пользователя с шахматным движком. Указаны основные моменты реализации блоков программного продукта.

Ключевые слова: шахматы, тренажер, модель, шахматный движок.

A.V. Azmagulov, D.B. Kuznetsov

Perm National Research Polytechnic University, Perm

SOFTWARE DEVELOPMENT “CHESS SIMULATOR”

This article discusses the development of the Chess Simulator software. The main blocks of the software product implementation together with the chess engine are described. A model of user functionality for a Chess simulator has been implemented. The principle of FEN notation construction is described. The principle of user interaction with the chess engine is described. The main points of the implementation of the blocks of the software product are indicated.

Keywords: chess, simulator, model, chess engine.

В 2020 г. начался стремительный рост популярности шахмат во всем мире. Катализатором роста был сериал «Ход королевы» (Queen's gambit), вышедший 23 октября 2020 г. на Netflix, который набрал массу положительных отзывов, подталкивая миллионы зрителей к увлечению шахматами [1]. Вслед за этим последовало повышение заинтересованности новых игроков шахматными программами, что, в свою очередь, приводит рынок к ситуации появления спроса на новое программное обеспечение по обучению ведению шахматной игры. Для удовлетворения спроса на соответствующее программное обеспечение было принято решение по реализации нового шахматного тренажера. Преимуще-

ством новых тренажеров по обучению шахматам в таком случае является использование новых версий открытых шахматных движков.

Реализацию программы было принято решение выполнять с помощью языка программирования C# на системе для построения клиентских приложений Windows WPF. В качестве шахматного движка используется Stockfish. Этот шахматный движок является сильнейшим на данный момент, поэтому использование его в программе добавляет конкурентоспособности на фоне остальных ранее созданных программных продуктов рынка [2].

Для обеспечения программного продукта стандартными приемами анализа шахматной партии она должна включать в себя следующий функционал:

- 1) сохранение проанализированной партии в формате PGN;
- 2) возможность переворота доски;
- 3) возможность комментирования отдельных ходов партии;
- 4) запись проанализированных пользователем ходов и свободный переход между ними;
- 5) возможность запроса лучшего хода позиции.

Формальное описание функционала системы в таком случае может быть представлено следующей моделью функциональных возможностей (рис. 1).

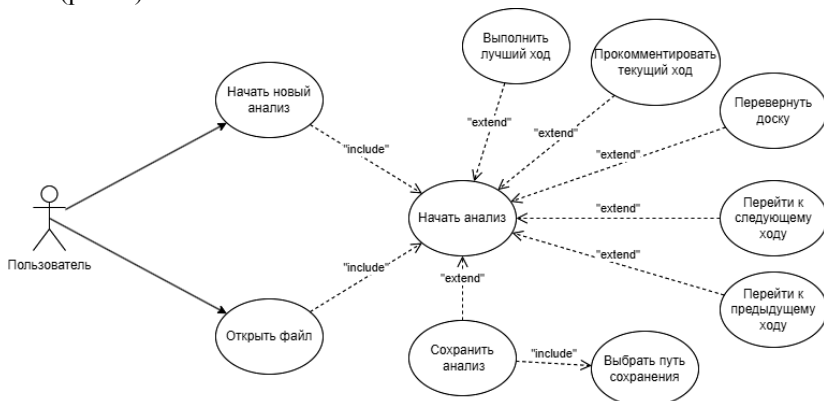


Рис. 1. Модель функциональных возможностей

Для реализации указанного функционала необходимо создание 3 блоков программы:

- 1) блок шахматной логики;

- 2) интерфейс пользователя;
- 3) блок взаимодействия с шахматным движком.

Блок Шахматной Логики выполняет функционал по предоставлению пользователю списка доступных ходов позиции и записи, проанализированных им ходов в формате PGN. Задача интерфейса пользователя, в свою очередь, будет заключаться во взаимодействии пользователя с функционалом программы, а блок взаимодействия с Шахматным движком будет вести непрерывное общение с существующим шахматным движком и передавать пользователю информацию о сильнейшем ходе определенной позиции. Сам процесс взаимодействия пользователя с шахматным движком можно представить в виде диаграммы последовательности (рис. 2).

Запрос лучшего хода

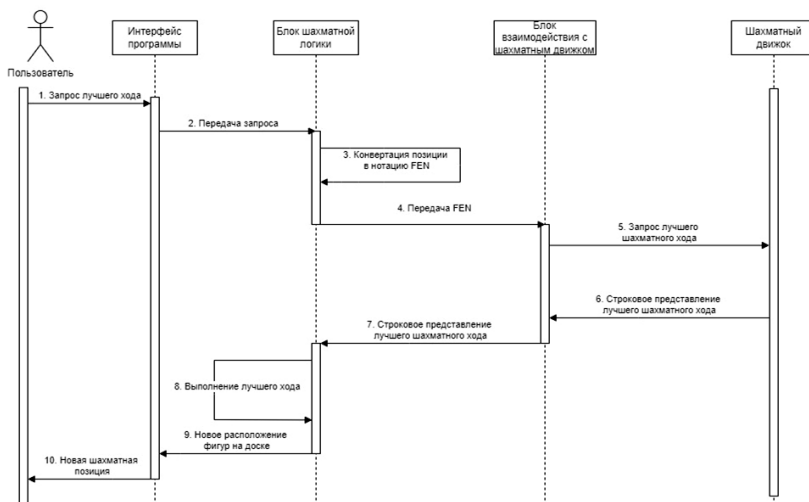


Рис. 2. Диаграмма последовательности взаимодействия пользователя с шахматным движком

На основании всего перечисленного следует сама реализация программного продукта. Одним из основных ее этапов является создание самого понятия шахматной доски, позиции и фигур. Взаимодействие этих объектов и будет являться прочным фундаментом для реализации дальнейшего функционала. Затем следует реализация функционала по сохранению проанализированных ходов. Для этого необходимо опи-

раться на понятие FEN нотации – строковое представление расположения шахматных фигур на доске. Реализация строкового представления доски включает в себя расположение всех фигур на доске, очередность хода фигуры, возможности рокировок королей, а также пункт доски, на котором можно осуществить взятие на проходе. Запись строки происходит с верхнего левого угла шахматной доски и следует до конца правого нижнего угла. Запись черных фигур происходит строчными буквами, а белых – заглавными буквами. В случае встречи пустых клеток шахматной доски следует их подсчет и запись соответствующей информации в строку. Например, запись шахматной позиции «1k6/8/3K4/8/8/7B/6Q1/8 w» включает в себя четыре фигуры, где король черных стоит на первой горизонтали на второй клетке от края левой доски (рис. 3). В том случае, если короли не имеют права на рокировку и отсутствует клетка, на которой можно совершить взятие на проходе, такие данные не заполняются.

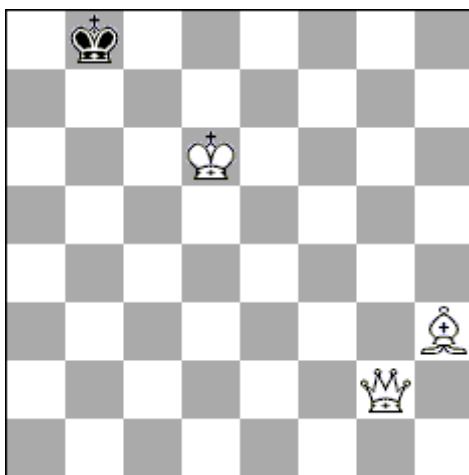


Рис. 3. Расположение фигур FEN-нотации «1k6/8/3K4/8/8/7B/6Q1/8 w»

Множество таких нотаций затем будут связаны между собой с казанием очередности вариантов анализируемой партии. Завершающим этапом при реализации программы является этап налаживания взаимодействия программы с шахматным движком. Для ее реализации необходим упор на документацию используемого шахматного движка

Stockfish, где указана вся информация о командах, применяемых для управления шахматного движка.

Таким образом, была разработана программа для повышения мастерства ведения игры в шахматы. Она позволит проводить анализ шахматных позиций пользователя с использованием самого передового шахматного движка, а также сохранить историю анализа для ее применения в будущих очных встречах.

Перспективой развития программы является имитация ведения определенного стиля игры на основе множества партий, что, в свою очередь, позволит вести более глубокую подготовку к определенным соперникам.

Библиографический список

1. New York Times [Электронный ресурс]. – URL: <https://www.nytimes.com/2022/06/17/crosswords/chess/chess-is-booming.html> (дата обращения: 13.04.2024).
2. Рейтинг шахматных Движков [Электронный ресурс]. – URL: https://www.computerchess.org.uk/ccrl/4040/rating_list_all.html (дата обращения: 15.04.2024).

Сведения об авторах

Азмагулов Артем Вадимович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: azmagulov95@bk.ru

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@gmail.com

А.Н. Мухамадыев, Д.Б. Кузнецов

Пермский национальный исследовательский
политехнический университет, Пермь

РЕАЛИЗАЦИЯ МЕХАНИЗМА СБОРА МЕТРИК С СЕРВЕРОВ В ВЕБ-СЕРВИСЕ ДЛЯ НАГРУЗОЧНОГО ТЕСТИРОВАНИЯ САЙТОВ

Исследуется возможность реализации механизма сбора метрик с серверов в веб-сервисе для нагрузочного тестирования сайтов. Механизм должен позволить пользователю получить метрики с серверов для отслеживания их состояний. В статье рассматривается проектирование алгоритма сбора метрик, а также его реализация в клиентском интерфейсе.

Ключевые слова: нагрузочное тестирование, веб-сервис для нагрузочного тестирования сайтов, сбор метрик с серверов, ssh-подключение.

A.N. Mukhamadyev, D.B. Kuznetsov

Perm National Research Polytechnic University, Perm

IMPLEMENTATION OF A MECHANISM FOR COLLECTING METRICS FROM SERVERS IN A WEB SERVICE FOR LOAD TESTING OF SITES

This article explores the possibility of implementing a mechanism for collecting metrics from servers in a web service for load testing sites. The mechanism should allow the user to obtain metrics from servers to track their states. The article discusses the design of a metrics collection algorithm, as well as its implementation in the client interface.

Keywords: load testing, web service for load testing of sites, collecting metrics from servers, ssh connection.

IT-компании занимаются выполнением госконтрактов на техническое сопровождение веб-сайтов. Одним из ключевых этапов сопровождения веб-систем является нагрузочное тестирование, которое представляет собой тестирование уровня производительности с целью оценки поведения элемента тестирования для ожидаемых условий низкого, типичного и пикового использования [1].

Одним из этапов нагрузочного тестирования является сбор результатов нагрузочного теста для последующего анализа с целью выявления проблем производительности и потенциальных точек отказа при-

ложения. Для определения производительности веб-приложения используются следующие показатели: пропускная способность, время отклика, количество ошибок и использование ресурсов на сервере, а именно нагрузка на процессор и объем занимаемой оперативной памяти. По показателям использования ресурсов можно определить, насколько эффективно приложение использует ресурсы сервера.

Для начала нужно спроектировать механизм сбора метрик. В первую очередь нужно получить настройки подключения и подключиться к серверам программно. Это можно сделать по протоколу ssh. Утилита `top`, которая установлена в семействе операционных систем Linux, позволяет получить размер используемой оперативной памяти и нагрузку на процессор. Во время сбора метрик соединение должно оставаться открытым, чтобы не тратить ресурсы сервера на переподключение. После окончания сбора метрик необходимо закрыть ssh-подключения. На рис. 1 представлена схема данного механизма.

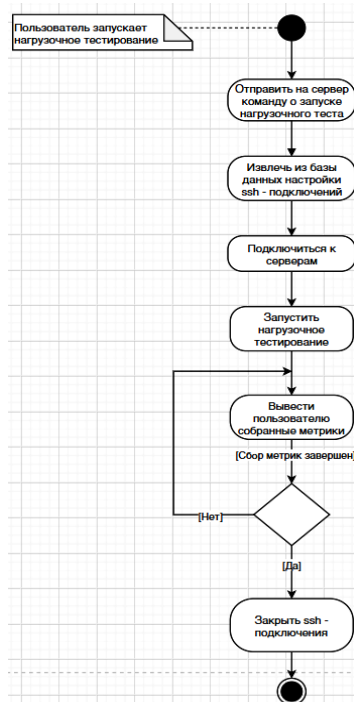


Рис. 1. Схема сбора метрик

Исходя из приведенного в рис. 1 алгоритма, можно спроектировать интерфейс, в котором будут показаны собранные метрики и программный алгоритм для серверной части веб-сервиса. На рис. 2 представлен макет интерфейса, на котором показана страница тестового плана. На боковой панели расположены кнопки для запуска и остановки нагрузочного тестирования.

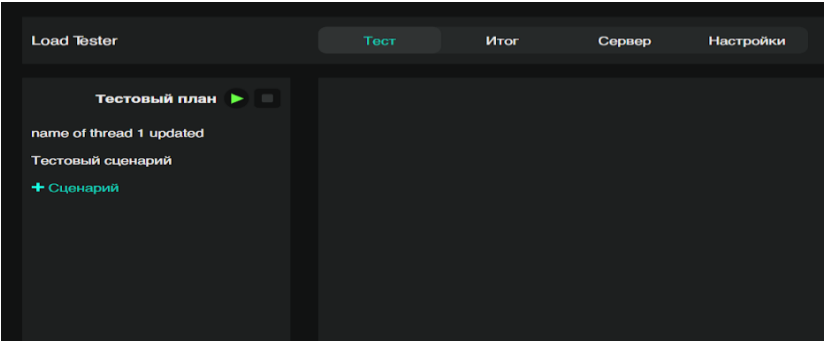


Рис. 2. Макет интерфейса страницы тестового плана

Собранные метрики будут показаны на отдельной странице «Сервер». Это необходимо для того, чтобы не загромождать страницу тестового плана. На рис. 3 представлен макет страницы с собранными метриками.

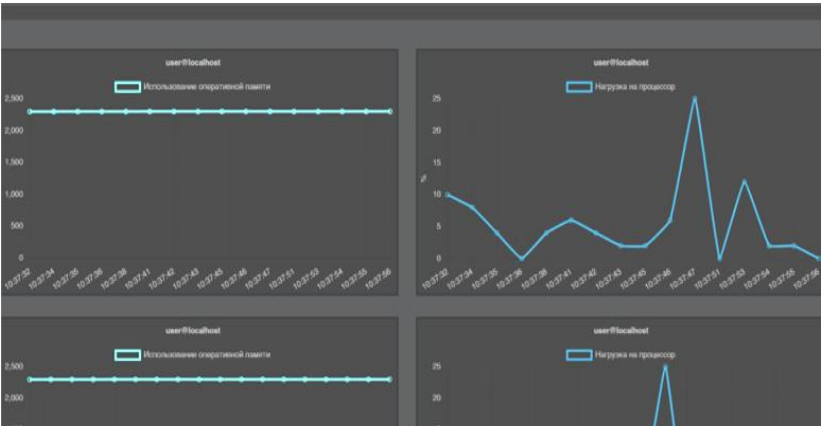


Рис. 3. Макет страницы с собранными метриками

На рис. 3 можно увидеть графики, которые отражают состояние серверов. На графиках слева показан размер используемой оперативной памяти в килобайтах, а на графиках справа – нагрузка на процессор в процентах. На каждом из графиков отражен адрес сервера и временные отметки для собранных метрик.

Серверная часть приложения реализована на Java с использованием фреймворка Spring Boot, который позволяет быстро создать API для клиентской части веб-сервиса [2]. Для подключения по протоколу ssh использовалась библиотека JSch [3].

Листинг 1 – Фрагмент кода из метода для подключения к серверам

```
List<Document> sshSettingsList = findSSHSettings();

for(Document sshSettings: sshSettingsList) {
    if (sshSettings.getBoolean(JsonFieldModel.IS_CONNECTION_ON))
    {
        Session session = new JSch().getSession(
            sshSettings.getString(JsonFieldModel.USER),
            sshSettings.getString(JsonFieldModel.SERVER),
            sshSettings.getInteger(JsonFieldModel.PORT)
        );
        session.setPassword(sshSettings.getString(JsonFieldModel.PASSWORD));
        session.connect();
        sshSettings.put(sshSettings.getString(JsonFieldModel.GUID), session);
    }
}
```

В листинге 1 представлены коллекция с настройками ssh-подключений, объект сессии для подключения к серверам и коллекция с открытыми сессиями.

Изначально из базы данных извлекаются настройки ssh-подключений. Затем происходит выборка настроек в цикле. Поскольку пользователь может добавить настройки и при этом не хочет подключиться к какому-либо из серверов, то предусмотрен флажок `isConnectionOn`. Если его значение равно `true`, то создается объект сессии с настройками подключения, происходит открытие сессии и помещение объекта в хеш-таблицу, в которой ключом является идентификатор настроек, а значением – объект сессии. В листинге 2 представлен фрагмент кода для сбора метрик с серверов.

Листинг 2 – Фрагмент кода для сбора метрик с серверов

```
ChannelExec channel = (ChannelExec-
ec)openedSessions.get(sshSettingsGuid).openChannel(
"exec");

channel.setCommand(
    ServerCommandModel.
el.GET_USED_MEMORY_COMMAND + " && " + ServerCom-
mandModel.GET_CPU_LOAD_COMMAND
);
serverResponseStream = new ByteArrayOut-
putStream();
chan-
nel.setOutputStream(serverResponseStream);
channel.connect();
...
channel.disconnect();
...
return loadFromServer;
```

В листинге 2 из хеш-таблицы, представленной в листинге 1, извлекается по идентификатору настроек объект сессии для создания объекта типа `ChannelExec`. Происходит установка команд с помощью метода `setCommand` для получения метрик с сервера. После этого происходит открытие канала для сбора метрик. После получения метрик с сервера канал закрывается и полученные метрики возвращаются клиентской части приложения для последующего формирования графиков.

Листинг 3 – Фрагмент кода для закрытия ssh-подключения

```

        for (Session      openedSession:      openedSessions.values()) {
                                openedSession.disconnect();
        }

```

На листинге 3 показан фрагмент кода для закрытия ssh-подключений. Из хеш-таблицы достаются объекты сессии и происходит их перебор в цикле с последующим закрытием с помощью метода `disconnect`.

Таким образом, был рассмотрен подход к реализации механизма сбора метрик с серверов в веб-сервисе для нагрузочного тестирования сайтов.

Библиографический список

1. JSch – Java Secure Channel [Электронный ресурс]. – URL: <http://www.jcraft.com/jsch> (дата обращения: 05.04.2024).
2. Spring Boot – официальная страница фреймворка с документацией [Электронный ресурс]. – URL: <https://docs.spring.io/spring-boot/index.html> (дата обращения: 05.04.2024).
3. Нильсен Якоб, Лоранжер Хоа. Web-дизайн: удобство использования веб-сайтов. – М.: Вильямс, 2009. – 376 с.

Сведения об авторах

Мухамадыев Алмаз Назимович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: almaz.mukhamadyev2017@yandex.ru

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@mail.ru

К.Е. Костылев, Д.С. Курушин

Пермский национальный исследовательский
политехнический университет, Пермь

**ИНФОРМАЦИОННАЯ СИСТЕМА МОНИТОРИНГА,
ПОИСКА, ДЕТЕКТИРОВАНИЯ И НАНЕСЕНИЯ НА КАРТУ
ОБЪЕКТОВ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ**

Описывается сравнение существующих систем мониторинга, поиска, детектирования и нанесения на карту объектов специального назначения, а также способы применения модели сверточной нейронной сети в системе обнаружения специальных графических объектов, что и является улучшением автоматизированной системы классификации, идентификации объектов и картографирования местности в зоне действия беспилотных летательных аппаратов.

Ключевые слова: беспилотные летательные аппараты, беспилотные авиационные системы, беспилотные авиационные технологии, FPV-дрон, метод, алгоритм, сверточная нейронная сеть, система управления, мониторинг.

K.E. Kostylev, D.S. Kurushin

Perm National Research Polytechnic University, Perm

**AN INFORMATION SYSTEM FOR MONITORING, SEARCHING,
DETECTING AND MAPPING SPECIAL-PURPOSE OBJECTS**

A comparison of existing systems for monitoring, searching, detecting and mapping special-purpose objects is described, as well as ways to use the convolutional neural network model in the system for detecting special graphical objects, which is an improvement of the automated system for classifying, identifying objects and mapping terrain in the unmanned aerial vehicles area of operation.

Keywords: unmanned aerial vehicles, unmanned aerial systems, unmanned aviation technologies, drone, method, algorithm, control system, monitoring, detection.

В настоящее время беспилотные летательные аппараты (БПЛА) стремительно приобретают популярность и широко применяются в различных сферах, включая военную, гражданскую и научную области. Особое внимание уделяется развитию систем и средств комплексов с БПЛА, так как это является одной из ключевых составляющих для эффективного использования и интеграции БПЛА в различных задачах.

Подтверждением актуальности данной темы является Распоряжение Правительства РФ от 21 июня 2023 г. № 1630-р. «Об утверждении

Стратегии развития беспилотной авиации РФ на период до 2030 г. и на перспективу до 2035 г. и плана мероприятий по ее реализации» [1], а также статистические данные по товарообороту FPV-дронов [2], которые показывают ежегодный прирост, в среднем на 20 % в год, что является основным показателем интереса потребителя.

В рамках данного проекта мы ограничиваемся исследованиями БПЛА вертолетного типа гражданского назначения. Информационные системы мониторинга, поиска, детектирования и нанесения на карту объектов специального назначения в большинстве случаев не исключают участия человека (оператора) в этом процессе. Системы управления и навесное оборудование на борту БЛА упрощают выполнение данных процессов, но не способны самостоятельно справиться с поставленными задачами поиска, детектирования и картографирования. При наличии постоянного канала связи с БПЛА, позволяющего передавать видеопоток, оператор производит непрерывное управление БПЛА, ведет непосредственный осмотр местности в режиме реального времени, обозначение объекта на карте производит ассистент оператора после ориентирования на местности и привязки наблюдаемого объекта к координатам или ориентирам местности. Обработав полученные данные, ассистент оператора наносит самостоятельно на карту специальные объекты и после непосредственной корректировки определяет искомые координаты, что является результатом данного процесса.

На рис. 1 приведен алгоритм системы управления БПЛА оператором и его ассистентом. Эта система является более точной, но более трудоемкой, учитывая то, что процессом вычислений занимаются оператор и его ассистент.

При отсутствии постоянного канала связи с БПЛА оператор проектирует при помощи современных программных комплексов маршрут полета БПЛА. После загрузки полетного задания в полуавтоматическом режиме с использованием навигации по опорным точкам БПЛА (вертолетного типа) может выполнять полет самостоятельно. Сохраняет видеопоток на устройство и производит безопасную посадку согласно установленной оператором конечной точки в маршруте. Далее оператор обрабатывает вручную полученные данные, исследуют каждое изображение на наличие искомого объекта. При обнаружении объекта определяет

его местоположение, привязываясь к ориентирам на местности и используя топографические данные карты, наносит обстановку на карту, вычисляет географическую координату специального объекта.

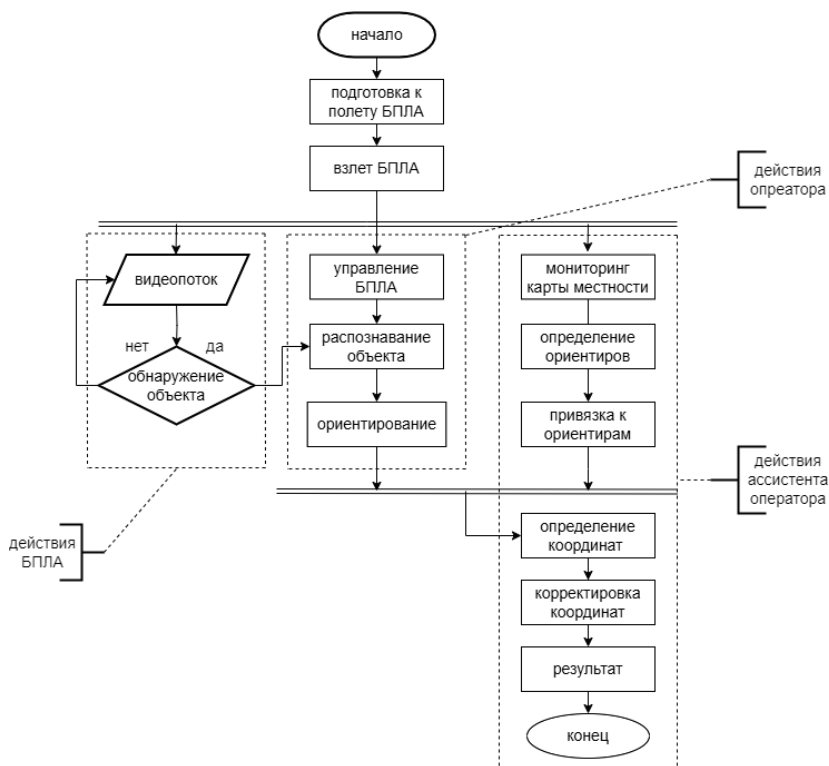


Рис. 1. Алгоритм системы управления БПЛА оператором и его ассистентом

На рис. 2 приведен алгоритм системы управления при отсутствии постоянного канала связи с БПЛА. Эта система является автономной, но медленной, учитывая то, что процессом вычислений занимается оператор после автоматической съемки местности БПЛА и его приземления.

Таким образом, мы описали два классических метода поиска, детектирования и картографирования специальных объектов.

Целью научно-исследовательской работы является улучшение автоматизированной системы классификации, идентификации объектов

и картографирования местности в зоне действия БПЛА, а также применение модели сверточной нейронной сети в системе обнаружения специальных графических объектов.

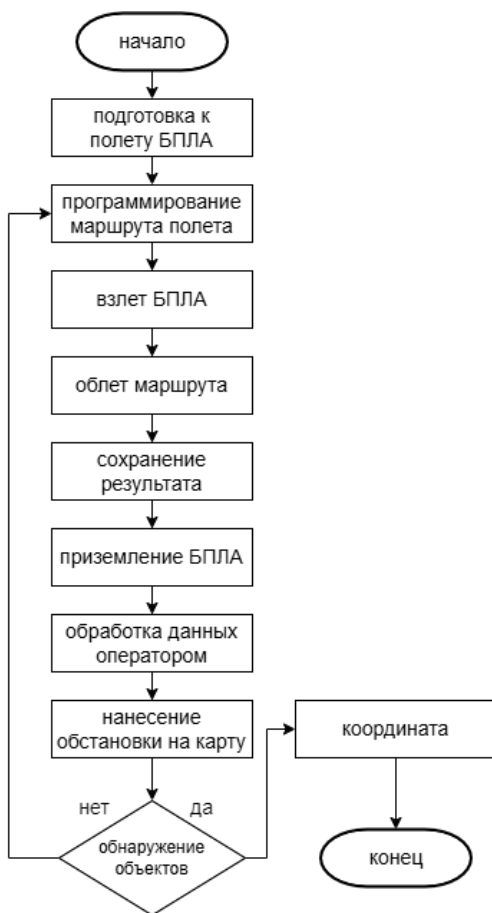


Рис. 2. Алгоритм системы управления при отсутствии постоянного канала связи с БПЛА

Основными требованиями, предъявляемыми к проектируемой системе, являются:

- автономное управление БЛА, но не исключающее корректировки маршрута оператором;

- внедрение в систему управления модель сверточной нейронной сети YOLO v8;
- более эффективное, точное и скоростное детектирование объектов;
- автоматическое распознавание объектов и отображение данных объектов в bounding box;
- математический расчет расстояния до искомого объекта относительно собственной геолокации;
- расчет и определение географических координат искомого объекта;
- отображение найденного объекта в приложении карт.

Исходя из данных запросов проектного исследования, алгоритм информационной системы мониторинга, поиска, детектирования и нанесения на карту объектов специального назначения будет иметь вид, изображенный на рис. 3, – алгоритм системы с применением модели YOLO v8.

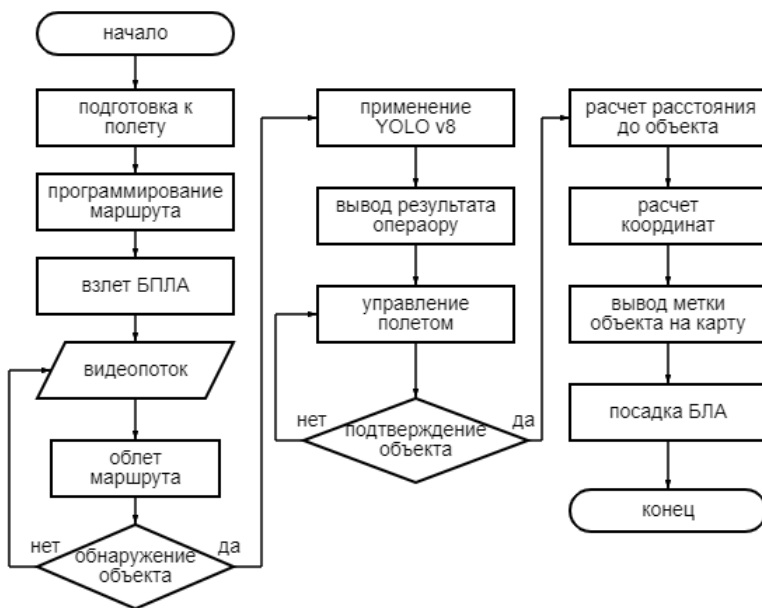


Рис. 3. Алгоритм системы с применением модели YOLO v8

Данная концептуальная схема отображает алгоритм работы проектируемой системы. Оператор формирует маршрут полета БЛА и производит запуск. БЛА осуществляет облет маршрута, автоматически опре-

деляя заданные объекты. При определении объекта происходит его распознавание, классификация и вызов оператора. Оператор переводит БЛА в ручное управление и производит визуальный осмотр найденного объекта. После подтверждения факта нахождения объекта система производит расчет дальности до объекта и его координаты. Затем происходит визуализация объекта на приложении карт.

Модель архитектуры сверточной нейронной сети YOLO v8 является наиболее эффективной в процессе распознавания объектов. Согласно официальному описанию, Ultralytics YOLOv8 [3, 4] – это последняя версия модели обнаружения объектов YOLO и сегментации изображений, разработанной Ultralytics. YOLOv8 – это передовая современная модель (SOTA), которая основывается на успехе предыдущих версий YOLO и включает новые функции и усовершенствования для повышения производительности и гибкости. Проблематикой распознавания объектов в данном проекте исходит из создаваемого видеопотока с борта БПЛА, который находится на удалении от объекта. Изображение специального объекта имеет вид малого размера. Для сохранения точности распознавания было проведено обучение модели YOLO v8 на наборе данных, включающем в себя 288 видеоклипов, сформированных из 261 908 кадров и 10 209 статических изображений, снятых различными камерами, установленных на беспилотных летательных аппаратах. Обучение модели сверточной нейронной сети произвели на 10 классах объектов, таких как человек, автомобиль, мотоцикл и т.д.

После проведения соответствующих экспериментов и тестов о полученной точности можно сделать вывод, что модель работает адекватно и правильно, находит малые объекты всех классов на изображениях с БПЛА.

Заключение. Поставленная цель, согласно которой было необходимо разработать модель улучшенной автоматизированной системы классификации, идентификации объектов и картографирования местности в зоне действия БПЛА, а также применение модели сверточной нейронной сети в системе обнаружения специальных графических объектов, была выполнена с помощью создания набора данных и обучения модели сверточной нейронной сети.

Были проведены тесты и эксперименты, согласно которым модель YOLO v8 справилась с определением малых объектов на изображении с борта БПЛА.

Библиографический список

1. Распоряжение Правительства РФ от 21 июня 2023 г. № 1630-р «Об утверждении Стратегии развития беспилотной авиации РФ на период до 2030 г. и на перспективу до 2035 г. и плана мероприятий по ее реализации». – URL: <https://www.garant.ru/products/ipo/prime/doc/407003744/>
2. Прогноз развития российского рынка беспилотных авиационных систем. – URL: https://2023.transweek.digital/upload/iblock/f61/jm08i7xk0k1a65y10je6g23ht8nmeen8/Varaytchenko_Alexey.pdf
3. Ultralytics YOLOv8 Docs. – URL: <https://docs.ultralytics.com/models/yolov8/>
4. Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification. – 2021. – 3 p. – URL: <https://bmcmmedinformdecismak.biomedcentral.com/counter/pdf/10.1186/s12911-021-01691-8.pdf>

Сведения об авторах

Костылев Кирилл Евгеньевич – магистрант Пермского национального исследовательского политехнического университета, гр. РИС-23-1м, г. Пермь, e-mail: kirya_19_89@mail.ru

Курушин Даниил Сергеевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: daniel@kurushin-perm.ru

Е.Д. Газеев

Пермский национальный исследовательский
политехнический университет, Пермь

ПРИМЕНЕНИЕ МАШИННОГО ЗРЕНИЯ В СЕЛЬСКОМ ХОЗЯЙСТВЕ

Статья посвящена актуальной теме применения технологий машинного зрения в агропромышленном комплексе. В качестве примера рассматривается разработка системы автоматического обнаружения урожая клубники для последующего сбора. Автор подробно описывает принципы работы CNN и архитектуру модели YOLOv8, подчеркивая ее преимущества, такие как высокая скорость обработки данных и точность обнаружения объектов. Демонстрируется способность модели эффективно обнаруживать и классифицировать ягоды клубники различной степени зрелости на изображениях.

Ключевые слова: CNN, сельское хозяйство, нейронная сеть, модель, машинное зрение, YOLO.

E.D. Gazeev

Perm National Research Polytechnic University, Perm

APPLICATION OF MACHINE VISION IN AGRICULTURE

The article is devoted to the topical topic of the application of machine vision technologies in the agro-industrial complex. As an example, the development of an automatic strawberry harvest detection system is considered. For subsequent collection, the author describes in detail the principles of CNN and the architecture of the YOLOv8 model, emphasizing its advantages such as high data processing speed and object detection accuracy. The ability of the model to effectively detect and classify strawberries of various degrees of maturity in the images is demonstrated.

Keywords: CNN, agriculture, neural network, model, machine vision, YOLO.

В современном мире наблюдается бурный рост интереса к технологиям искусственного интеллекта (ИИ), и сельское хозяйство не является исключением. Стремительное развитие вычислительных мощностей и алгоритмов машинного обучения открывает новые возможности для автоматизации и оптимизации агропромышленных процессов, делая их более эффективными, экономичными и экологичными. Одной из ключевых технологий, которая уже сегодня меняет облик сельского хозяйства, является машинное зрение. Оно включает в себя три фунда-

ментальные задачи, это детекция объектов (object detection), сегментация (segmentation) и классификация (classification). В данной статье будет рассматриваться одна из задач – детекция объектов.

Детекция объектов – это задача машинного зрения, которая заключается в обнаружении и локализации объектов определенного класса на изображении или видео. В контексте сельского хозяйства это может быть обнаружение плодов, сорняков, вредителей, больных растений или других объектов, представляющих интерес для агронома. Алгоритмы object detection анализируют изображения, выделяя характерные признаки, формируют предположения о наличии объектов и классифицируют их, определяя класс и местоположение. В сельском хозяйстве это может помочь обеспечивать автоматизацию мониторинга посевов, точное земледелие и роботизированный сбор урожая.

Целью данного исследования является обзор сверточных нейронных сетей (CNN), в частности модели YOLOv8 [1], для ее последующего интегрирования в роботизированную систему по сбору урожая клубники.

Как работают сверточные нейронные сети. Сверточные нейронные сети (CNN) – это тип искусственных нейронных сетей, специально разработанных для эффективной обработки изображений и видео. Их архитектура вдохновлена зрительной корой головного мозга, где нейроны реагируют на определенные участки поля зрения [2].

Ключевым элементом CNN являются сверточные слои, в которых выполняется операция свертки. Свертка – это математическая операция, применяющая фильтр (ядро свертки) к входному изображению, выделяя специфические признаки, такие как края, углы, текстуры. Каждый фильтр обучается на наборе данных, «настраиваясь» на определенный паттерн.

CNN состоят из нескольких сверточных слоев, каждый из которых выделяет признаки разного уровня абстракции. Первые слои обнаруживают простые признаки (края, углы), а последующие слои комбинируют их в более сложные паттерны (формы, объекты). После сверточных слоев обычно следуют слои пулинга, которые уменьшают размерность изображения, сохраняя при этом важную информацию.

В конце CNN находится полносвязный слой, который классифицирует выделенные признаки, т.е. определяет, какой объект присутствует на изображении [3].

YOLOv8: скорость в реальном времени. YOLOv8 (You Only Look Once version 8) – это одноступенчатая модель обнаружения объектов, которая предсказывает вероятности классов и ограничивающие рамки за один проход. Популярная модель обнаружения объектов и сегментации изображений была разработана Джозефом Редмоном и Али Фархади из Вашингтонского университета [4].

Запущенная в 2015 г., YOLO быстро завоевала популярность благодаря своей высокой скорости [5].

Так как YOLOv8 позволяет работать с высокой скоростью и высоким показателем производительности для видео, то эта модель является лучшим выбором [6] для последующего интегрирования в роботизированные системы.

Обучение нейронной сети. Для обучения нейронных сетей при ограниченных вычислительных ресурсах рекомендуется использовать облачные платформы, такие как Google Colab. Преимущество Google Colab заключается в предоставлении бесплатного доступа к вычислительным мощностям, достаточным для обучения моделей среднего размера.

Процесс обучения нейронной сети YOLOv8 с использованием Google Colab включает в себя следующие шаги: определение параметров модели, загрузку датасета, обучение модели и оценку ее производительности. В качестве примера будет рассматриваться обучение модели на датасете изображений ягод клубники.

В данном эксперименте количество эпох обучения установлено равным 20, а размер изображения – 640 пикселей. После завершения обучения модели проводится анализ ее производительности на основе полученных результатов (рис. 1).

Изображение представляет собой график, содержащий десять отдельных графиков, отображающих процесс обучения нейронной сети. Верхний ряд представляет обучающие данные, а нижний – данные валидации. Левые три столбца представляют значения функции потерь для различных аспектов модели: **box_loss** отражает, насколько хорошо модель предсказывает ограничивающие рамки вокруг объектов; **cls_loss** отражает, насколько хорошо модель классифицирует объекты по категориям; **dfl_loss** отражает, насколько хорошо модель предсказывает распределение объектов.

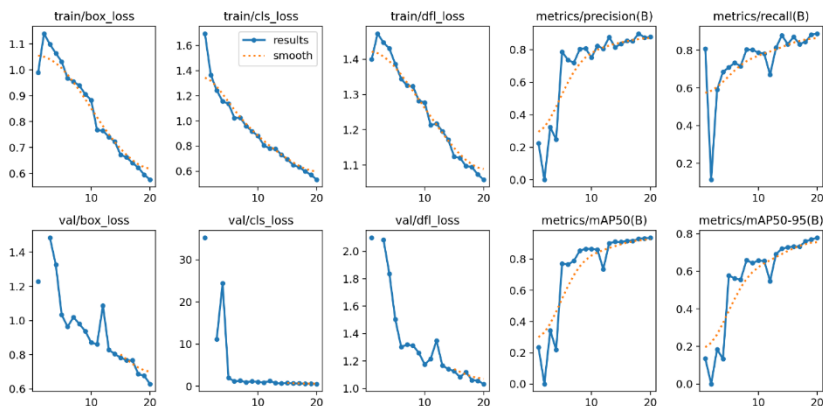


Рис. 1. Результаты обучений нейронной сети

Последние два столбца представляют метрики оценки точности модели((B) указывает на то, что метрика вычисляется для ограничивающих рамок): **precision(B)**: доля правильно предсказанных объектов среди всех предсказанных объектов; **recall(B)**: доля правильно предсказанных объектов среди всех реально существующих объектов; **mAP50(B)**: Mean Average Precision (mAP) - средняя точность на разных уровнях. **mAP50-95(B)**: mAP, рассчитанный для различных порогов Intersection over Union (IoU) от 0,5 до 0,95 с шагом 0,05, IoU (Intersection over Union) – это мера перекрытия между предсказанной ограничивающей рамкой (bounding box) и истинной ограничивающей рамкой объекта.

Основные выводы по обучению:

1. Модель успешно обучается, о чем свидетельствует снижение значений функции потери и увеличение значений метрик точности.
2. Значения функции потери и метрик для обучающих и валидационных данных близки, что указывает на то, что модель не переобучается.
3. Общая тенденция метрик точности положительная, что говорит о том, что модель с каждым эпохой становится лучше.
4. **Результаты.** По представленным изображениям можно сделать вывод, что нейронная сеть демонстрирует в целом хорошие результаты в обнаружении и классификации клубники на изображениях (рис. 2).

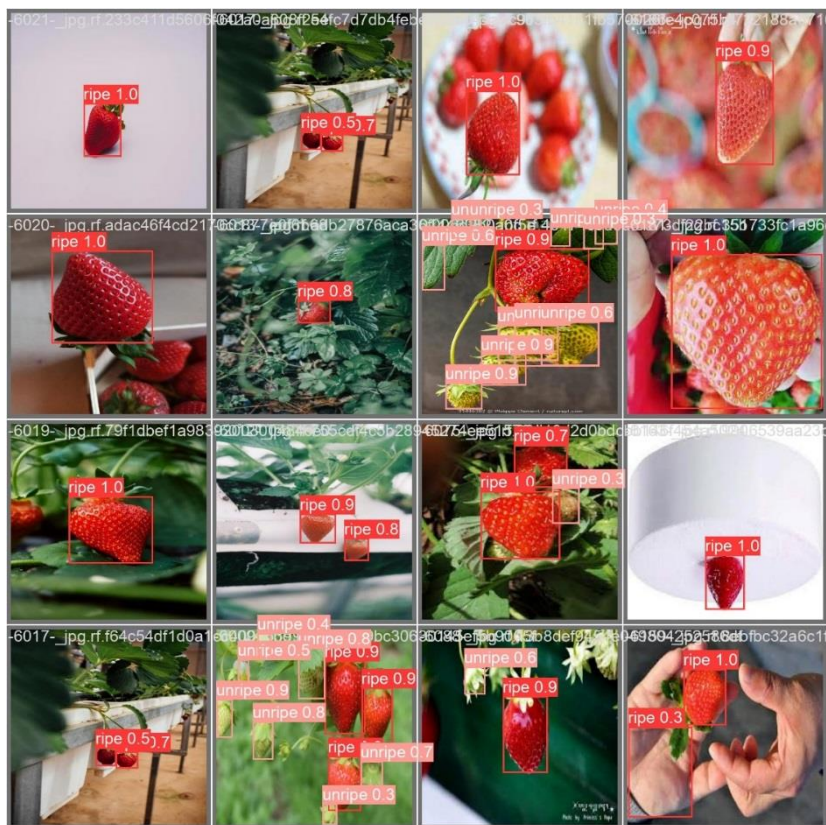


Рис. 2. Предположения нейронной сети

Модель в большинстве случаев корректно обнаруживает клубнику на изображениях, даже если ягоды частично скрыты листвой или находятся в разных ракурсах. Для спелой клубники модель зачастую выдает высокую степень уверенности (более 0,8), что свидетельствует о ее способности точно классифицировать спелые ягоды. Модель демонстрирует способность различать спелую и неспелую клубнику, хотя для неспелых ягод уверенность может быть ниже.

В целом обученная нейронная сеть показывает многообещающие результаты в обнаружении и классификации клубники, но для дальнейшего улучшения ее производительности требуется доработка.

Заключение. В рамках исследования возможностей нейронных сетей в сельском хозяйстве была рассмотрена задача обнаружения и классификации ягод клубники на изображениях. Для решения данной задачи была выбрана архитектура YOLOv8, которая продемонстрировала свою эффективность в задачах компьютерного зрения.

В ходе исследования была обучена модель YOLOv8 на датасете изображений клубники. Результаты первичного обучения демонстрируют потенциал использования нейронных сетей для автоматизации процессов в сельском хозяйстве, таких как оценка урожайности, мониторинг состояния растений и оптимизация сбора урожая.

Однако для достижения более высоких показателей точности и надежности модели рекомендуется расширение и улучшение качества датасета. Дальнейшие исследования в данном направлении позволят создать более точные и надежные модели для решения задач в сельском хозяйстве. Разработка и применение таких моделей может способствовать повышению эффективности и устойчивости сельскохозяйственного производства.

Библиографический список

1. Ultralytics. – URL: <https://docs.ultralytics.com/>
2. Scholarpedia. – URL: <http://www.scholarpedia.org/article/Neocognitron>
3. Scientificresearch. – URL: <https://scientificresearch.ru/images/PDF/2018/21/svertochnye.pdf>
4. Ultralytics. – URL: <https://docs.ultralytics.com/#where-to-start>
5. Keylabs. – URL: <https://keylabs.ai/blog/yolov8-vs-ssd-choosing-the-right-object-detection-model/>
6. SpringerOpen. – URL: <https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00434-w#Sec31>

Сведения об авторе

Газеев Евгений Дмитриевич – бакалавр Пермского национально-го исследовательского политехнического университета, гр. СРПС-22-16, техник Пермской научно-производственной приборостроительной компании, г. Пермь, e-mail: miheev.evgeni12@gmail.com

Д.Е. Онищук

Пермский национальный исследовательский
политехнический университет, Пермь

**ПРОГРАММНЫЙ МОДУЛЬ ДЛЯ ИНФОРМАЦИОННОЙ
СИСТЕМЫ ПОДБОРА НЕФТЯНОГО ОБОРУДОВАНИЯ
АО «НОВОМЕТ-ПЕРМЬ»**

В этой статье представлены результаты разработки и исследования программного модуля в составе информационной системы подбора нефтяного оборудования. Представлены данные внедрения этого модуля для промышленного предприятия АО «Новомет-Пермь».

Ключевые слова: программное обеспечение, базы данных, web-приложение, тестирование.

D.E. Onishuk

Perm National Research Polytechnic University, Perm

**PROGRAM MODULE FOR THE INFORMATION
SYSTEM OF OIL EQUIPMENT SELECTION
OF JSC "NOVOMET-PERM"**

This article presents the results of development and research of the program module as part of the information system of oil equipment selection. The data of implementation of this module for the industrial enterprise JSC "Novomet-Perm" are presented.

Keywords: software, databases, web-application, testing.

Актуальность решаемой проблемы, цель и задачи статьи.

В настоящее время, когда нефтегазовая отрасль является одной из наиболее быстро развивающихся в мире, подбор и эксплуатация нефтяного оборудования являются важными задачами для компаний, занимающихся добычей нефти и газа.

Однако процесс подбора нефтяного оборудования является очень сложным и требует высокой квалификации специалистов. Кроме того, рынок нефтяного оборудования очень динамичен и быстро меняется, что затрудняет принятие решений о покупке и производстве оборудования для нефтяных установок.

Проблема подбора нефтяного оборудования заключается в том, что существует множество различных видов нефтяного оборудования, но в большинстве своем они не автоматизированы и не отвечают требованиям конкретной нефтяной компании. В связи с тем что подбор нефтяного оборудования не автоматизирован посредством ПО, увеличиваются трудозатраты, существует риск подбора не подходящих механических узлов.

Для решения данной проблемы была разработана информационная система по подбору нефтяного оборудования, которая позволяет автоматизировать процесс выбора оборудования для насосной установки и снизить затраты на его сборку.

Цель – знакомство с предметной областью и закрепление теоретических знаний, используемых при разработке программного модуля для информационной системы по подбору нефтяного оборудования.

Для достижения поставленной цели необходимо решить следующие задачи:

- 1) проанализировать бизнес-процесс подбора нефтяного оборудования;
- 2) сформулировать требования к модулю для информационной системы;
- 3) изучить и определить стек используемых технологий, подходящих для разработки веб-приложения, обосновать выбор;
- 4) разработать структуру базы данных для хранения информации о нефтяном оборудовании;
- 5) спроектировать интерфейс веб-приложения;
- 6) реализовать модуль подбора нефтяного оборудования на основе входных параметров;
- 7) описать функционал разработанного веб-приложения;
- 8) провести тестирование системы и оценить ее эффективность.

Анализ требований и исходных данных. Системы для подбора нефтяных установок начали свое развитие в XIX в., когда первые нефтяные скважины были пробурены в США. В то время подбор оборудования для нефтяных установок был крайне сложным и трудоемким процессом, который требовал большого опыта и высочайшего уровня знаний.

В настоящее время системы для подбора нефтяных установок постоянно развиваются и улучшаются, чтобы соответствовать меняющимся требованиям рынка и технологиям добычи нефти. Они стано-

вятся все более сложными и интегрированными с другими системами управления производством, что позволяет оптимизировать процесс добычи нефти на всех этапах.

Именно интеграция с другими системами вкпе дает полноценную экосистему налаженных бизнес-процессов в компании, занимающейся нефтедобычей.

Программный модуль для информационной системы должен позволить пользователю подбирать нефтяное оборудование для комплектования насосной установки. К таким компонентам относятся: Корпус (рис. 1); ступени (рис. 2); модуль-секции (рис. 3); промышленные электродвигатели (рис. 4).

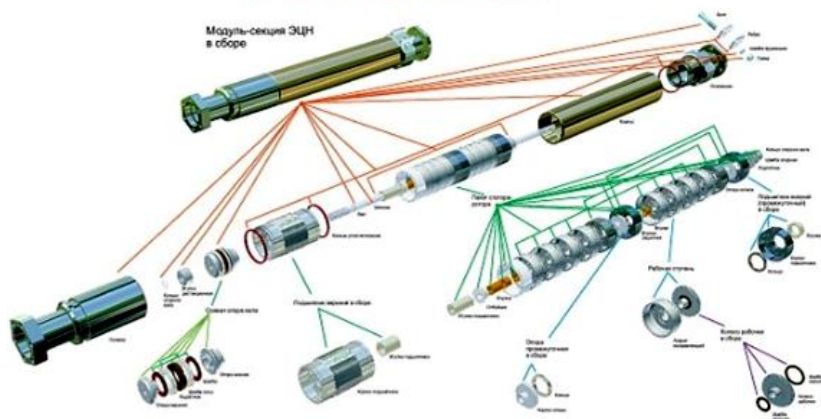


Рис. 1. Корпус



Рис. 2. Ступени, рабочие колеса

СОСТАВ МОДУЛЯ-СЕКЦИИ НАСОСА ЭЦН



Модуль-секция является составной частью нефтегазового оборудования, которая представляет собой функционально законченное изделие, предназначенное для эксплуатации в составе установки. Примерами модуль-секций являются сепараторы, насосы, компрессоры, фильтры и др. Они могут быть изготовлены из различных материалов, таких как сталь, алюминий, керамика и др. Они также могут иметь различные размеры и формы в зависимости от требований конкретного проекта.

435



Рис. 4. Промышленный электродвигатель
асинхронного типа

При заполнении указанных выше полей программа позволяет осуществить подбор нефтяного оборудования – ступеней из базы данных, которые наиболее подходят по заданным параметрам.

Пример и результат подбора нефтяного оборудования продемонстрирован на рис. 5.

Входные параметры: подача – $1000 \text{ м}^3/\text{сут}$; плотность – 800 кг/м^3 ; вязкость – 200 сСт ; напор на приеме – 100 м ; напор на выкиде – 450 м ; частота вращения вала – не ограничена; выход за границы рабочей зоны – не ограничен; тип, габарит ступени – не ограничен; тип сборки ступени – не ограничен; материал ступени – не ограничен.

При таких входных параметрах программа подобрала наиболее подходящие ступени для насоса.

В результирующей таблице, представленной на рис. 5, пользователь может выбрать интересующую его ступень, после чего программа автоматически начнет подбирать модуль-секции по выбранной пользователем ступени и отображать всевозможные варианты состава модуль-секций для комплектования насосной установки – показано на рис. 6.

Остальное нефтяное оборудование подбирается аналогично. Специалист выбирает наиболее подходящий из предложенных вариантов, после чего программа начинает подбирать другие механические узлы и уведомлять пользователя, если он выбрал некорректные данные.

Архитектура системы. Архитектура системы в целом имеет классическую модель web-приложения – рис. 7. С клиента посылаются запросы на сервер, обработка происходит на сервере, после чего данные извлекаются из базы данных и посылаются сервером обратно на клиентскую часть, обновляя DOM дерево.

Выбор технологий, языков программирования для разработки. Для разработки вышестоящей информационной системы коллегами было принято решение использовать PHP фреймворк для серверной части приложения (backend) и библиотеку React Js для клиентской (frontend). Фреймворки позволяют во много раз ускорить процесс разработки приложений. Не требуется тратить время на выполнение рутинных задач, благодаря чему происходит значительная экономия времени. После сравнительного анализа в качестве использования PHP фреймворка для вышестоящей системы был выбран Laravel, так как показал наилучшие результаты производительности.

- **Laravel** (время ответа на запрос 65 мс, запись в базу данных 1,2 мс, чтение из базы данных 1 мс, параллельные запросы 1200);

- **Codeigniter** (время ответа на запрос 85 мс, запись в базу данных 1,8 мс, чтение из базы данных 0,8 мс, параллельные запросы 1000);

- **Yii** (время ответа на запрос 90 мс, запись в базу данных 1,6 мс, чтение из базы данных 1,3 мс, параллельные запросы 1000).

Для клиентской части был выбран React Js, так как, после сравнительного анализа он также показал наилучшие результаты.

Структура базы данных. В качестве базы данных была выбрана PostgreSQL, так как после сравнительного анализа было ясно, что она покрывает потребности разрабатываемого приложения. Структура базы данных для разработанной системы насчитывает около 98 таблиц, в своем большинстве связанных друг с другом через внешние ключи. Имеются отдельные справочники, в которых хранится информация о нефтяном оборудовании. При подборе задействуются почти все справочники, которые влияют на комплектование насосной установки.

Разработка главных диалоговых форм. Главное окно программного модуля представлено на рис. 8.

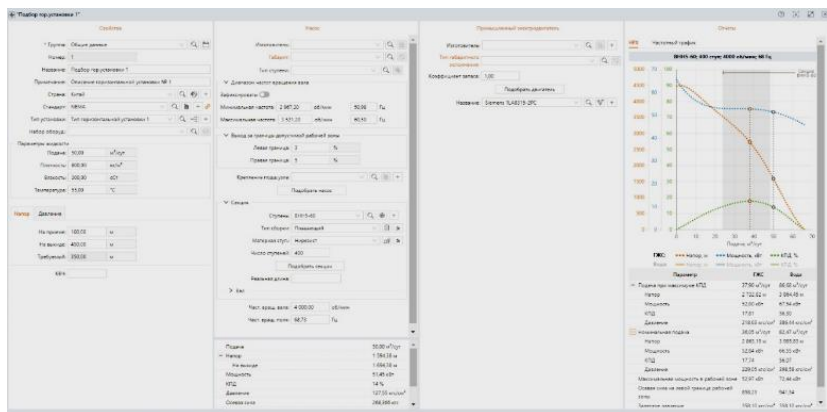


Рис. 8. Главное окно программного модуля по подбору нефтяного оборудования

Тестирование и внедрение. Программа тестировалась на реальных скважинах, о чем свидетельствует подписанный акт об успешных опытно-промышленных испытаниях. Было подобрано различное нефтяное оборудование, которое было использовано для комплектации нефтяных установок. По итогам испытаний программа показала свою эффективность.

Компания АО «Новомет-Пермь» внедрила данную программу на своем предприятии, что позволило в разы сократить время, потраченное на подбор нефтяного оборудования, уменьшить риск человеческого фактора и значительно увеличить производство нефтяных насосов, так как программа уведомляет о возможных ошибках подбора и предлагает наиболее подходящее оборудование для комплектования установки.

Сведения об авторе

Онищук Дмитрий Евгеньевич – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-19-1бз, г. Пермь, e-mail: foros90@mail.ru

**Д.В. Яруллин, М.С. Углицких,
А.О. Морохин, А.В. Мазяров**

Пермский национальный исследовательский
политехнический университет, Пермь

ЯЗЫКОВЫЕ МОДЕЛИ И МЕТОДЫ ПРЕОБРАЗОВАНИЯ РЕЧИ В ТЕКСТ

В данной статье приведено описание методов распознавания звуковых данных и перевод их в текст посредством возможностей нейросетевых технологий. Поднята проблема актуальности данной темы. Результат может быть использован для применения в различных сферах.

Ключевые слова: распознавание речи, преобразование речи в текст, языковые модели, Whisper.

**D.V. Yarullin, M.S. Uglitskikh,
A.O. Morokhin, A.V. Mazyarov**

Perm National Research Polytechnic University, Perm

LANGUAGE MODELS AND METHODS OF SPEECH-TO-TEXT CONVERSION

This article describes methods of recognizing audio data and translating them into text by means of neural network technologies. The problem of relevance of this topic is raised. The result can be used for application in various spheres.

Keywords: speech recognition, speech-to-text, language models, Whisper.

В современном информационном мире распознавание и перевод в текст набора звуковых данных оказывает большую помощь для людей, экономя человеческие ресурсы, которые могли бы быть потрачены на прочтение и запись. Эта технология активно применяется в различных областях – от написания заметок и создания текстовых транскриптов до управления голосовыми ассистентами и проведения анализа данных. Современные языковые модели, способные преобразовывать речь в текст, демонстрируют впечатляющие результаты, обеспечивая высокую точность и скорость обработки информации.

В век цифровых технологий и интернета вещей удаленное взаимодействие со всеми устройствами, онлайн-конференции и т.д. стали привычными. В частности, обычный поход к врачу для снятия показаний здоровья может быть заменен более удобным аналогом – голосовым

ассистентом, который будет помогать человеку вести записи показаний здоровья для последующей передачи их врачу. Человеку с ограниченной мобильностью может доставлять дискомфорт невозможность посещения больницы физически. Пропускная способность поликлиник также не всегда достаточна. Данный ассистент способен упростить повседневную жизнь, облегчить пациенту и врачу их коммуникацию, уменьшить загруженность больниц.

Технология преобразования речи в текст основана на алгоритмах машинного обучения и нейронных сетях. Основные этапы включают в себя сбор информации, фильтрацию шумов и разделение аудиодорожек, преобразование звуковых волн в цифровые сигналы, которые анализируются по частоте и амплитуде, финальное распознавание цифровых сигналов в текст [1].

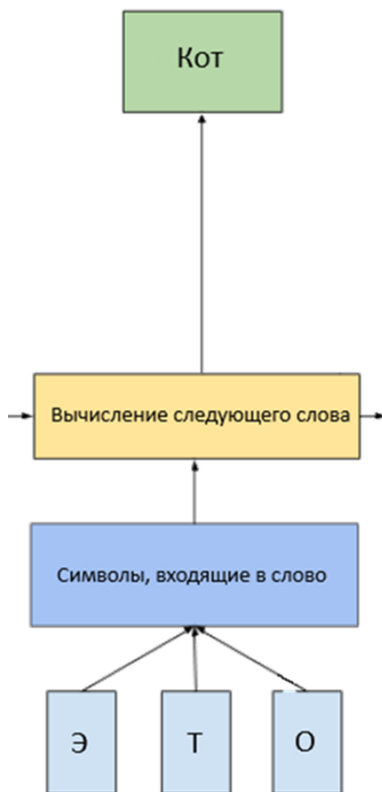


Рис. 1. Принцип работы статистической модели

Виды языковых моделей:

1. Статистические n -граммные модели – простые статистические модели, которые оценивают вероятность использования текущего слова на основе нескольких предыдущих слов. Чем больше n , тем больше контекста учитывается.

Принцип работы основывается на сборе статистики о частоте появления n -граммы в тексте. После этого вычисляется вероятность каждого слова на основе $n-1$ предыдущих слов. На основе вероятностей делается предсказание о следующем слове в последовательности. Принцип работы изображен на рис. 1.

Преимущества:

- эффективность для краткосрочных зависимостей;
- быстрота обучения;
- легкая реализация.

Недостатки:

- контекст ограничен $n-1$ предыдущих слов, а значит, такая модель недостаточно эффективно работает с длинными зависимостями;
- слова, которые редко встречаются в тексте могут не попасть в обученную модель;
- не учитывается грамматика и синтаксис языка.

2. Рекуррентные нейронные сети предназначены для обработки последовательных данных, таких как текст или аудио. В отличие от обычных нейронных сетей RNN имеют обратные связи, что позволяет им сохранять информацию о предыдущих состояниях и учитывать контекст, поэтому они подходят для анализа последовательностей данных и предсказания следующих элементов в серии [2].

Принцип работы: на каждом шаге RNN принимает входное значение и состояние из предыдущего шага, генерируя новое, а состояния, в свою очередь, передаются через сеть, что позволяет учитывать контекст всей последовательности. Это отражено на рис. 2.

Преимущества:

- качественная работа с последовательными данными, поскольку они могут учитывать информацию из предыдущих состояний;
- обучение на данных разной длины.

Недостатки:

- проблема с работой долгосрочных зависимостях;
- высокая вычислительная нагрузка;
- требовательность к обучающим данным.

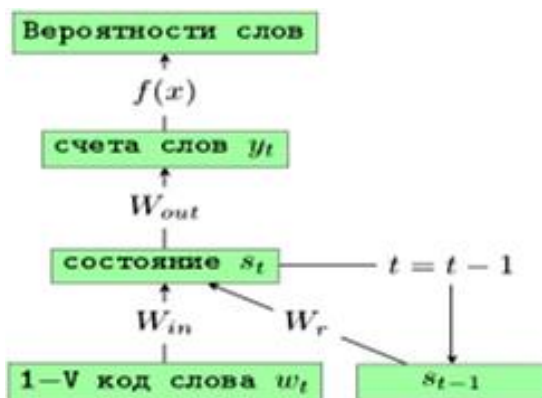


Рис. 2. Архитектуры рекуррентной нейронной сети

3. Конволюционные нейронные сети (CNN) предназначены для обработки аудиосигналов с извлечением характеристик звуковых волн, которые впоследствии могут быть переданы в другие модели (рис. 3).

Принцип действия: к входным данным применяются фильтры для извлечения локальных признаков, далее каждый фильтр сдвигается, создавая карты признаков. После этого происходит уменьшение размерности карт признаков для уменьшения количества параметров и повышения устойчивости к искажениям. В конце используются завершающие слои для классификации других задач.

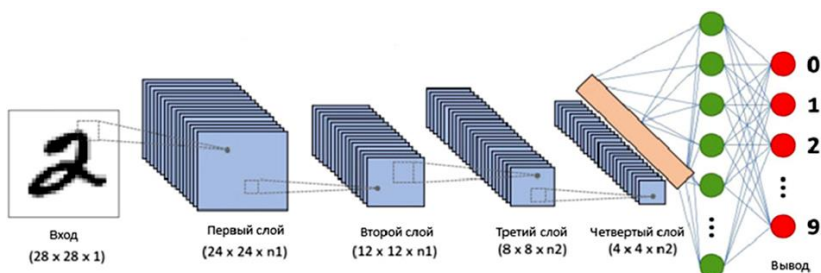


Рис. 3. Архитектура конволюционной нейронной сети

Преимущества:

- способность обработки больших объемов данных;
- высокая производительность с использованием аппаратных ускорителей.

Недостатки:

- требуется большое количество данных для обучения;
- ограничение в обработке долгосрочных зависимостей.

Самой инновационной из современных моделей является Whisper, разработанная компанией OpenAI. Это нейронная сеть, обученная на больших объемах данных, около 680 000 ч, двух видов: аудио и их текстовые транскрипции. Помимо обычного распознавания текстов, модель способна «понимать» контекст. Принцип работы продемонстрирован на рис. 4 [3].



Рис. 4. Принцип работы Whisper

Сравнение точности работы модели приведено на рис. 5. Вертикальная ось – количество слов, в которых допущена ошибка. По горизонтали представлены компании, которые участвовали в эксперименте и над чьими корпусами данных производился эксперимент, рис. 5, сравнительные результаты сделаны в таблице.

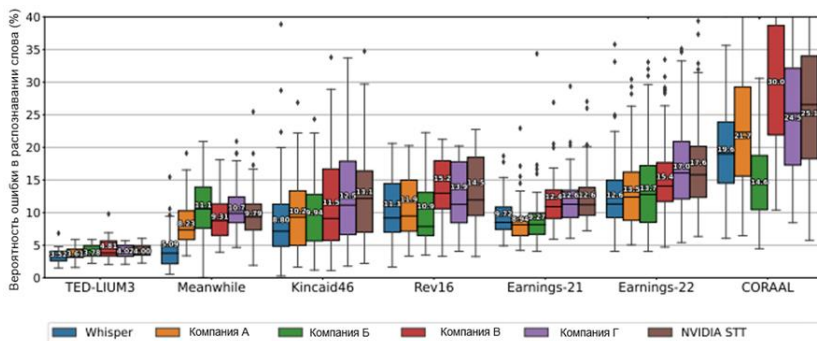


Рис. 5. Тестирование моделей

Исходя из таблицы предпочтительней является нейронная сеть Whisper, так как она обладает наилучшими показателями по точности распознавания, скорости обработки данных, обработке акцентов и диалектов.

Сравнительная таблица языковых моделей

Критерий	Статистическая модель	Рекуррентная нейронная сеть (RNN)	Конволюционная нейронная сеть (CNN)	Whisper
Точность распознавания	6	8	9	9
Скорость обработки	8	3	8	9
Устойчивость к акцентам и диалектам	2	7	7	8
Надежность и стабильность	5	7	7	9
Интеграция с медицинскими системами	5	7	7	8
Конфиденциальность и безопасность данных	8	5	5	8
Обучаемость и адаптация	3	6	7	8

В заключение стоит отметить, что современные языковые модели для преобразования речи в текст представляют собой мощные инструменты, способные значительно упростить множество задач и повысить эффективность работы в различных сферах. Whisper предполагает, что «масштабирование предварительного обучения со слабым контролем до сих пор недооценивалось в исследованиях по распознаванию речи. Мы достигли наших результатов без использования методов самоконтроля и самообучения, которые были основой недавних крупномасштабных работ по распознаванию речи, и продемонстрировали, как простое обучение на большом и разнообразном наборе данных под наблюдением» [3]. Проблема взаимодействия пациента с врачом может быть решена с использованием данных технологий. В будущем будут разработаны более совершенные и адаптивные системы, которые смогут учитывать все нюансы человеческой речи и предоставлять высокоточные результаты в самых различных условиях.

Библиографический список

1. A neural probabilistic language model / Yoshua Bengio, R_ejean Ducharme, Pascal Vincent, Christian Janvin // J. Mach. Learn.
2. Language models are few-shot learners / Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon

Child, Aditya Ramesh, Daniel M. Ziegler, Je_rey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever and Dario Amodei.

3. Robust Speech Recognition via Large-Scale Weak Supervision / Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, Ilya Sutskever. – URL: <https://cdn.openai.com/papers/whisper.pdf>

Сведения об авторах

Яруллин Денис Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: d.v.yarullin@ya.ru

Углицких Максим Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: uglitskikhmaksim@gmail.com

Морохин Александр Олегович – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: a_morohin@vk.com

Мазяров Александр Викторович – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: kent1234810@gmail.com

О.Ф. Ахматнурова, Д.Б. Кузнецов

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА VIEW ДЛЯ ПРОЦЕССА ПЕРЕДАЧИ КАРТОГРАФИЧЕСКИХ СЛОЕВ НА КАРТУ

Географические информационные системы являются важным инструментом для организации, анализа и визуализации пространственных данных. Основные элементы геоинформационных систем – это слои, представляющие различные типы информации на карте, что позволяет пользователям анализировать данные. Разработка VIEW для процесса передачи картографических слоев на карту является важным этапом в автоматизации работы с географическими информационными системами. VIEW представляет собой виртуальную таблицу, которая содержит запрос, результат выполнения которого может быть использован как обычная таблица. В статье проанализирован текущий процесс создания view-слоя на карту.

Ключевые слова: view, картографический слой, географическая информационная система, разработка.

O.F. Ahmatnurova, D.B. Kuznetsov

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF VIEWING THE PROCESS OF TRANSFERING CARTOGRAPHIC LAYERS TO THE MAP

Geographic information systems are an important tool for organizing, analyzing, and visualizing spatial data. The basic elements of a GIS are layers that represent different types of information on a map, allowing users to analyze the data. The development of VIEW for the process of transferring cartographic layers to a map is an important step in automating work with geographic information systems). VIEW is a virtual table that contains a query, the result of which can be used as a regular table. The article analyzes the current process of creating a view layer on a map.

Keywords: view, cartographic layer, geographic information system, development.

В настоящее время геоинформационные системы (ГИС) являются неотъемлемым инструментом для хранения, обработки и отображения пространственных данных. Одними из важных компонентов подобных картографических систем являются картографические слои. Слои

представляют собой набор данных, которые отображаются в виде графических объектов на карте.

Каждый слой содержит разнообразные информационные элементы, такие как географические объекты (например, дороги, реки, границы), тематические данные (например, данные о населении, климатические зоны, экономические показатели) и прочее. Благодаря использованию слоев пользователи могут увидеть различные аспекты информации на карте и провести их анализ.

Авторы работы [1–3] описывают реализацию view для процесса передачи картографических данных слоев на карту на основе базы данных PostgreSQL.

Для передачи точечных данных на карту в PostgreSQL будет использоваться географический тип данных Point. Создание view для отображения таких данных может включать выборку координат точек из таблицы с данными и представление их в виде географического объекта типа Point.

Для отображения картографических данных на карте будет использоваться географический тип данных Polygon. Реализация view для передачи многоугольных данных включает в себя выборку координат вершин многоугольников из соответствующих таблиц и формирование географических объектов типа Polygon.

Использование географических типов данных Point и Polygon предоставляет широкие возможности для отображения разнообразных пространственных данных на карте. При создании view для передачи ГИС-данных в PostgreSQL необходимо учитывать форматирование данных, чтобы обеспечить их корректное отображение на карте. Это включает в себя правильное представление координат географическом объекте, что позволит точно расположить объекты на карте и обеспечить пользователю понятное визуальное восприятие информации.

Для создания стандартного вида view типа Point необходимо использование следующих функций:

1. Функция `st_setsrid(..., 4326)`, которая устанавливает для созданной точки указанной SRID (Spatial Reference ID) 4326, который соответствует системе координат WGS 84 (World Geodetic System 1984). Это позволяет правильно интерпретировать координаты и их расположение на поверхности Земли.

2. Функции `st_point`, в котором будут создаваться объект `Point`, который являюсь объектом на карте с заданными координатами долготы (`longitude`) и широты (`latitude`). В данном примере рассмотрим, `cs.longitude_wgs84` и `cs.latitude_wgs84` представляют собой значения координат точки в формате WGS84.

3. Функции `geometry(Point, 4326)`, которая преобразуют объект с установленным SRID в геометрический объект типа `Point` с указанием того же SRID. Таким образом, происходит преобразование объекта к указанному геометрическому типу и системе координат (SRID 4326).

Таким образом, получим следующую конструкцию view представления `Point`.

```
st_setsrid(st_point(cs.longitude_wgs84::double precision,  
cs.latitude_wgs84::double precision), 4326)::geometry(Point,4326) AS geodata
```

В итоге создается геометрический объект типа `Point` с указанными координатами и устанавливает для него систему координат WGS 84, что позволяет представить точку на карте в правильном географическом контексте. Созданный объект `geodata` может быть далее использован в запросах для пространственного анализа или отображения на карте (рис. 1).

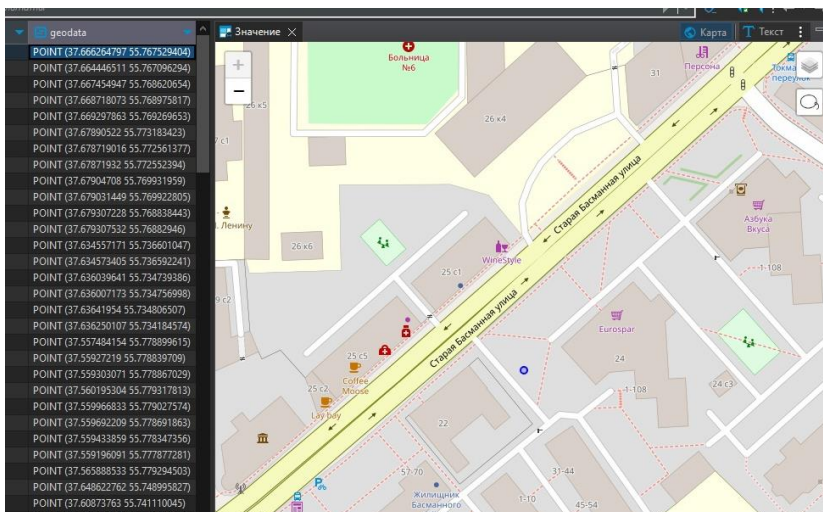


Рис. 1. View типа объекта point

Для создания стандартного вида view типа `Polygon` с указанным радиусом необходимо использование следующих функций:

1. Функция `st_buffer(..., cs.coveragearea::double precision)`, создает буфер вокруг географического объекта. Здесь она используется для создания буфера вокруг нашей точки с радиусом, указанным в столбце `coveragearea`.
2. Функции `st_setsrid(..., 4326)`, устанавливает определенную систему координат для точки, в данном случае используется EPSG код 4326, который соответствует WGS 84.
3. Функции `st_makepoint(cs.longitude_wgs84::double precision, cs.latitude_wgs84::double precision)`, создает точку на плоскости с координатами долготы и широты из столбцов `longitude_wgs84` и `latitude_wgs84` соответственно.
4. Функция `::geography`, преобразует точку в географический тип данных, что представляет собой географическую сущность на поверхности Земли.
5. Функция `geometry(Polygon,4326)`, преобразует результат в геометрический тип данных, который конкретно указывает на многоугольник с использованием системы координат EPSG 4326.

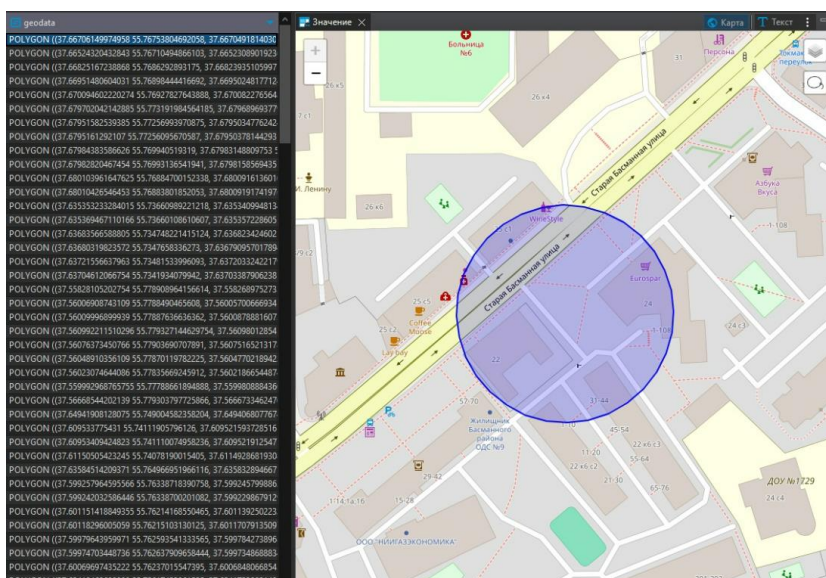


Рис. 2. View типа объекта polygon

Таким образом, получим следующую конструкцию view представления Polygon с указанным радиусом:

```
st_buffer(st_setsrid(st_makepoint(cs.longitude_wgs84::double precision,
cs.latitude_wgs84::double precision), 4326)::geography, cs.coveragearea::double
precision)::geometry(Polygon,4326) AS geodata
```

В итоге создается геометрический объект типа Polygon, географический многоугольник (полигон) в виде буфера вокруг введенной точки с определенным радиусом в указанной системе координат. Созданный объект geodata может быть далее использован в запросах для пространственного анализа или отображения на карте (рис. 2).

Таким образом, в статье проанализирован существующий процесс создания view для картографических слоев, описан алгоритм построения view. Разработка view для передачи картографических слоев на карту является важным этапом в создании геоинформационных систем. Правильный выбор инструментов, удобный интерфейс для пользователя и оптимизация производительности – ключевые аспекты, которые помогут создать эффективную и удобную систему для работы с картографическими данными. Все эти факторы следует учитывать при разработке view для карты, чтобы обеспечить удовлетворение потребностей пользователей и эффективное использование геоинформационных систем.

Библиографический список

1. Моргунов Е.П. PostgreSQL. Основы языка Sql.: Форум, 2018. – 123 с.
2. Professional ASP.NET MVC 3 / J. Galloway, P. Haack, B. Wilson, K. Scott Allen. – М.: Wiley Publishing, Inc, 2011. – 432 p.
3. Блиновская Я.Ю., Задоя Д.С. Введение в геоинформационные системы. – М.: Инфра-М, Форум, 2012. – 112 с.

Сведения об авторах

Ахматнурова Ольга Федоровна – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: ahmatnurovaolya@yandex.ru.

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@mail.ru

М.А. Чувашев, Е.В. Долгова

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА МОДЕЛИ И АЛГОРИТМОВ АВТОМАТИЗИРОВАННОГО СБОРА И РАССЫЛКИ НОВОСТЕЙ

Рассматривается модель автоматизированного сбора и рассылки новостей для образовательных учреждений, предназначенная для эффективного и своевременного информирования студентов и преподавателей о событиях, изменениях и новостях. Описанная модель данных охватывает ключевые аспекты процесса сбора новостей, включая идентификацию источников информации, обработку и хранение данных, а также рассылку уведомлений пользователям. В рамках модели также представлен алгоритм интеграции с мессенджерами для минимизации нагрузки на серверную часть системы. Статья описывает этапы работы алгоритма от сбора данных до их передачи и доставки пользователям, что делает ее универсальным инструментом для образовательных учреждений, стремящихся улучшить процессы коммуникации и информирования.

Ключевые слова: алгоритм, модель, автоматизация, сбор новостей, рассылка, телеграм-бот.

M.A. Chuvashhev, E.V. Dolgova

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A MODEL AND ALGORITHMS FOR AUTOMATED NEWS COLLECTION AND DISTRIBUTION

This article discusses a model of automated news collection and distribution for educational institutions, designed to effectively and timely inform students and teachers about events, changes and news. The described data model covers key aspects of the news gathering process, including identification of information sources, data processing and storage, as well as sending notifications to users. The model also presents an algorithm for integration with messengers to minimize the load on the server side of the system. The article describes the stages of the algorithm from data collection to transmission and delivery to users, which makes it a universal tool for educational institutions seeking to improve communication and information processes.

Keywords: algorithm, model, automation, news gathering, mailing, telegram bot.

Уведомления для студентов и преподавателей являются важной составляющей работы современных вузов. Они позволяют оперативно информировать учащихся и педагогов о важных событиях, изменениях

в расписании, новых материалах и заданиях, а также обеспечивать обратную связь между вузом и его участниками. В настоящее время существует несколько способов организации систем уведомлений: использование официального телеграм-бота или посещение всех источников по отдельности, рассмотрим каждую из систем подробнее.

Одной из распространенных систем в вузе является официальный телеграм-бот ПНИПУ [1]. Данный бот присылает пользователям новости только из одного источника – официального сайта ПНИПУ [2]. Преимуществами такой системы является отсутствие необходимости заходить на официальный сайт для получения новостей с него. Однако недостатками являются единый источник новостей, отсутствие настроек источников для пользователей и отсутствие способа взаимодействия преподавателей с пользователями для их информирования.

Другой способ получения уведомлений – это посещение всех источников, где размещается актуальная информация для студентов и преподавателей. Это могут быть сайт вуза, группы в социальных сетях и другие ресурсы. Преимуществами такого метода являются возможности получения информации из различных источников, что обеспечивает более полное информирование студентов и преподавателей. Однако недостатками являются необходимость самостоятельно посещать и проверять все ресурсы на наличие новых обновлений, что занимает много времени и может привести к пропуску важной информации. Также отсутствует единый канал связи преподавателей со студентами.

Автоматизированная система получения и рассылки новостей.

Исходя из анализа существующих систем, пользователям при любом из рассмотренных способов получения уведомлений приходится взаимодействовать с множеством ресурсов вуза, что увеличивает риск пропустить важное событие и сильно зависит от человеческого фактора. Для оптимизации процесса получения информации и минимизации возникающих проблем была разработана модель автоматизированной системы, которая позволяет пользователям получать все необходимые новости в одном месте, исключив необходимость проверки множества ресурсов. Модель представлена на рис. 1. Для реализации данной модели были разработаны несколько ключевых алгоритмов.

Алгоритм парсера новостей. Данный алгоритм играет ключевую роль в обеспечении своевременного и точного информирования студентов и преподавателей о событиях и изменениях в вузе. Он проверяет выбранные источники новостей, такие как официальный сайт ПНИПУ

и группы в социальной сети «ВКонтакте» [3, 4], собирая все последние новости и проверяя их новизну. Это предотвращает дублирование информации и обеспечивает полное информирование пользователей. Алгоритм представлен на рис. 2.

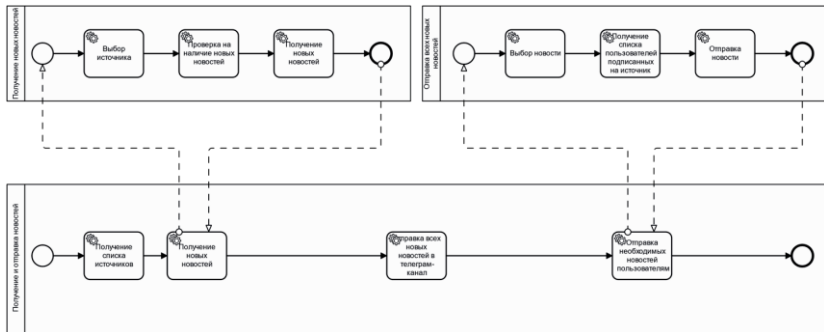


Рис. 1. Модель автоматизированной система получения и рассылки новостей

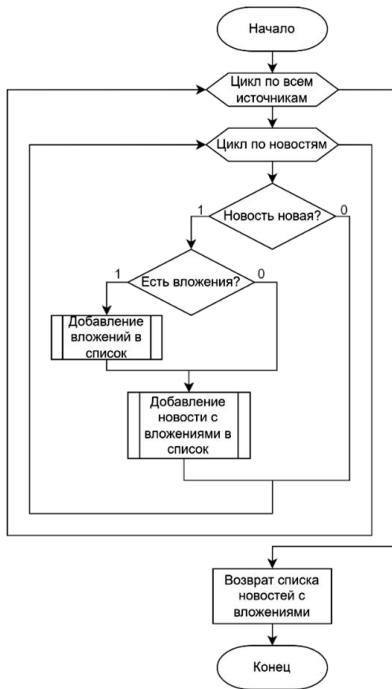


Рис. 2. Блок-схема алгоритма парсера новостей

Алгоритм получения списка идентификаторов новостей. Данный алгоритм отправляет собранные новости и вложения в технический телеграм-канал для получения уникальных идентификаторов. Эти идентификаторы используются для дальнейшей обработки и связывания текста новостей с вложениями. Такой подход позволяет разгрузить серверную часть системы, перекладывая хранение больших объемов данных на мессенджер, что ускоряет процесс доставки новостей пользователям и снижает нагрузку на серверы. Алгоритм продемонстрирован на рис. 3.

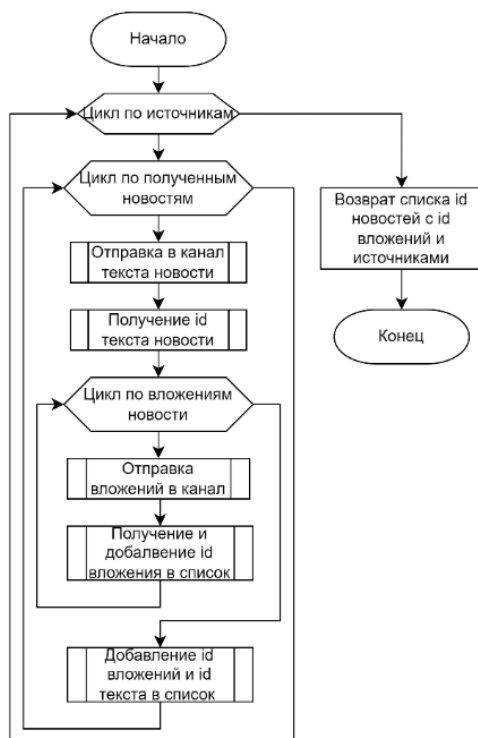


Рис. 3. Блок-схема алгоритма получения списка идентификаторов новостей

Алгоритм сбора новостей и добавления в базу данных. Алгоритм обеспечивает интеграцию и сохранение собранных новостей и вложений в базе данных. Он обрабатывает каждый элемент в полученном списке идентификаторов, добавляя соответствующие

записи в базу данных. Это включает сохранение идентификатора новости, даты новости, источника новости, а также идентификаторов и типов вложений. Алгоритм изображен на рис. 4.

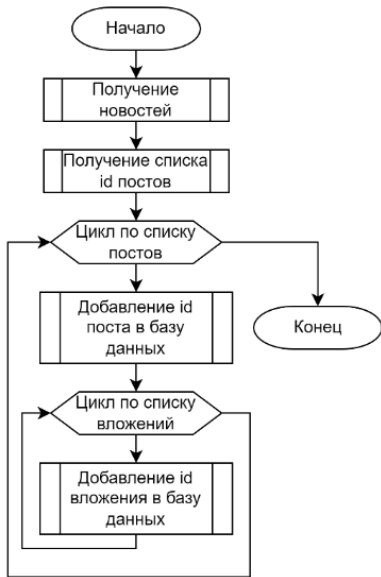


Рис. 4. Блок-схема алгоритма сбора новостей

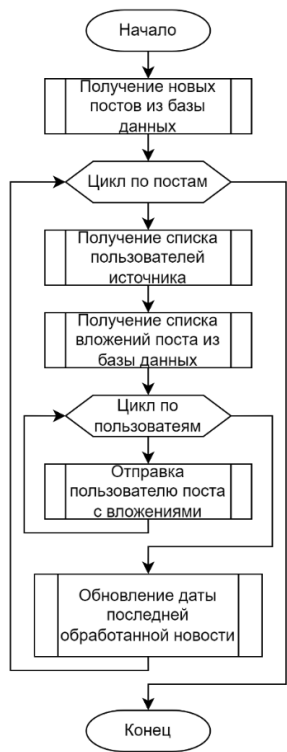


Рис. 5. Блок-схема алгоритма рассылки новостей

Алгоритм отправки новостей пользователям. Этот алгоритм отвечает за доставку новых постов и их вложений пользователям. Он получает идентификаторы новых постов и их вложений из базы данных, определяет целевую аудиторию для рассылки, получает список вложений и отправляет их пользователям через телеграм-бота. Алгоритм также обновляет дату последней обработанной новости в базе данных, чтобы избежать повторной отправки уже обработанных

новостей при следующем запуске. Алгоритм отправки новостей изображен на рис. 5.

Заключение. Разработанная модель и алгоритмы для автоматизированной системы уведомлений обеспечивают своевременное и точное информирование студентов и преподавателей о событиях и изменениях в вузе. Система позволяет эффективно собирать, обрабатывать и рассылать новости, минимизируя риск пропуска важной информации и улучшая качество взаимодействия между вузом и его участниками. Использование технического телеграм-канала для хранения данных и получения идентификаторов позволяет значительно снизить нагрузку на серверную часть системы, ускоряя процесс доставки новостей и улучшая общую производительность системы.

Библиографический список

1. Официальный телеграм-бот ПНИПУ [Электронный ресурс]. – URL: <https://t.me/PNRPUbot> (дата обращения: 21.05.2024).
2. Официальный сайт ПНИПУ [Электронный ресурс]. – URL: <https://pstu.ru> (дата обращения: 20.05.2024).
3. ВКонтакте. Официальный сайт [Электронный ресурс]. – URL: <https://vk.com/> (дата обращения: 22.05.2024).
4. ВКонтакте для разработчиков. Официальная документация [Электронный ресурс]. – URL: <https://dev.vk.com/ru> (дата обращения: 19.05.2024).

Сведения об авторах

Чувашев Максим Алексеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: d02c44@gmail.com

Долгова Елена Владимировна – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: elena@dolgova.space

О.Л. Викентьева, М.В. Чистяков

Пермский национальный исследовательский
политехнический университет, Пермь

РЕАЛИЗАЦИЯ АВТОРИЗАЦИИ ДЛЯ МОБИЛЬНЫХ УСТРОЙСТВ

В этой статье исследуются способы реализации авторизации для мобильных устройств. Рассмотрены виды авторизации, преимущества и недостатки при реализации для мобильных устройств. Спроектирован и реализован способ авторизации для мобильных устройств.

Ключевые слова: авторизация, мобильные устройства, аутентификация, защита информации, JWT.

O.L. Vikenteva, M.V. Chistyakov

Perm National Research Polytechnic University, Perm

IMPLEMENTATION OF AUTHORIZATION FOR MOBILE DEVICES

In this article the ways of implementation of authorization for mobile devices are investigated. Types of authorization, advantages and disadvantages of implementation for mobile devices are considered. The authorization method for mobile devices is designed and implemented.

Keywords: authorization, mobile devices, authentication, information security, JWT.

Одна из главных проблем информации – это защита данных и при этом сохранение доступности для определенного круга лиц [1]. В современном мире к защите информации проявляется большой интерес, так как с появлением сети Интернет появились новые способы кражи данных, а также их использование в корыстных целях. Одним из первых барьеров защиты информации является процесс авторизации пользователя. Авторизация – это процесс предоставления прав пользователю на выполнение определенных действий и подтверждение данных прав при попытке их использования [2].

Авторизация различается по пяти основным признакам: способу доступа, методу разграничения прав, типу аутентификации, количеству проверок, по типу токена.

Виды режимов авторизации по методам разграничения прав:

– дискреционное управление доступом, где каждый объект имеет владельца, который может предоставлять права доступа, также такая система имеет одного суперпользователя;

– мандатное управление доступом, где информация имеет разделение по уровню секретности, а пользователи имеют определенный уровень допуска; одним из главных преимуществ является ограничения прав владельца информации;

– управление доступом на основе ролей, где каждому пользователю выдается роль, на основе которой предоставляются права доступа к информации;

– контроль доступа на основе контекста;

– контроль доступа на основе решетки (LBAC);

– контроль доступа на основе атрибутов (ABAC).

Виды авторизации по способу доступа:

– онлайн, где авторизация происходит удаленно, например через сеть Интернет;

– оффлайн, где авторизация происходит без необходимости подключения к Сети.

Также стоит отметить, что важным элементом авторизации является аутентификация. Аутентификация – это процесс подтверждения подлинности предоставленного идентификатора [3]. Существует несколько видов аутентификации:

– при помощи электронной подписи, когда проверяется подлинность подписи, чтобы удостовериться в идентичности подписанта и целостности подписанного документа;

– при помощи ввода пароля;

– биометрическая, когда подлинность проверяется по биометрическим признакам пользователя, к примеру, такие как отпечаток пальца или скан сетчатки глаза;

– через географическое местоположение, когда подтверждается личности пользователя на основе его текущего физического местоположения. Зачастую используется как дополнительная мера защиты;

– уникальный предмет, к примеру, пропуск, смарт-карта.

Для подтверждения подлинности пользователя при попытке использования привилегий используется токен. Токен – это элемент данных, используемый для обеспечения безопасности и подтверждения подлинности при доступе к информационным системам и сервисам. Токены бывают физические, например накопитель содержащий ключ, и нефизические, JWT, который содержит JSON-объект с информацией

о пользователе и его правах, закодированный и подписанный с помощью криптографического алгоритма.

Для мобильных устройств в основном используются такие виды авторизации, как:

- биометрическая, по отпечатку пальца; отпечаток пальца хранится в системе устройства. Мобильное приложение запрашивает проверку совпадения отпечатка пальца текущего пользователя устройства и отпечатка пальца, сохраненного в системе;

- биометрическая, по сканированию лица; скан лица хранится в системе устройства. Мобильное приложение запрашивает проверку совпадения скана лица текущего пользователя устройства и скана лица, сохраненного в системе;

- логин и пароль; аутентификация по логину и паролю заключается в проверке введенных данных пользователем на устройстве с хранящимися данными на сервере.

- сканирование динамического QR-кода; аутентификация с помощью динамического QR-кода используется в связке с еще одним устройством, на котором пользователь уже авторизован. На авторизованном устройстве на экран выводится QR-код с непродолжительным временем жизни, к примеру, одна минут, после сканирования которого пользователь получает доступ.

Таким образом, можно сделать вывод, что видов авторизации существует много и также есть их слияния для получения наилучшей защиты информации.

Для мобильных приложений, предоставляющих доступ к базам данных и позволяющих редактировать информацию, необходимо реализовывать защиту от кражи данных и их неправомерного использования с целью предотвращения возможного ущерба. При разработке программного обеспечения «Складской учет» для компании ООО «Камский кабель» был реализован модуль «Авторизация». Цель данного модуля заключается в разбиении доступа к различным модулям системы, в защите от неправомерного использования информации и для идентификации пользователя.

Авторизация в системе «Складской учет» основана на дискреционном разграничении прав с небольшим отличием от стандартного вида, что суперпользователей, выдающих права доступа, несколько, а владельцев у объектов нет. Выбор данного вида разграничения прав обусловлен тем, что программное обеспечение включает в себя не-

сколько модулей, которые невозможно разграничить по правам, и при этом они имеют одинаковый уровень секретности. Аутентификация пользователя производится путем сканирования многоцветного QR-кода сотрудника, что является недопустимым в системах с неограниченным кругом пользователей, но так как доступ к сети, откуда ведется работа с приложением, имеет ограниченный круг лиц, то данный способ полностью защищает от несанкционированного доступа. Также стоит принять во внимание, что QR-код имеет длительность работы три месяца, после чего его нужно продлевать или создавать новый. Созданием учетных записей занимаются пользователи, назначенные как суперпользователи. Суперпользователями в системе являются техническая поддержка, разработчики и старшие мастера. Таким образом, создание новых учетных записей и продление старых может выполняться в любой момент времени. В условиях круглосуточной работы с приложением это важная функциональная особенность.

Работа модуля «Авторизация» происходит следующим образом: сотрудник печатает собственный QR-код в ERP-системе, после чего сканирует его с помощью мобильного устройства, сервер получает данные, зашифрованные в QR-коде, проверяет наличие и действительность записи в ERP-системе. В случае, если запись действительна, предоставляет доступ к работе. Также стоит учесть, что в архитектуру информационной системой «Складской учет» включен сервер, который обменивается информацией с клиентским мобильным приложением по защищенному HTTPS-протоколу с использованием JWT, что позволяет отключать бездействующие сессии, а также подтверждать права доступа при использовании функций модулей системы.

HTTPS (HyperText Transfer Protocol Secure) – это расширение протокола HTTP, которое используется для безопасного обмена данными между веб-браузером и веб-сервером. Основное отличие HTTPS от HTTP заключается в использовании криптографического протокола TLS (Transport Layer Security) или его предшественника SSL (Secure Sockets Layer) для шифрования данных. Это обеспечивает защиту передаваемой информации от перехвата и изменения злоумышленниками [4].

JWT (JSON Web Token) – это компактный, самодостаточный формат для передачи данных между сторонами в виде JSON-объекта. Эта информация может быть проверена и доверена, так как она подписана цифровой подписью. JWT часто используется для аутентификации и авторизации в веб-приложениях [5].

Таким образом, реализована защита от кражи данных и несанкционированного доступа. Аутентификация по многоразовому QR-коду позволяет сотрудникам взаимодействовать с системой с помощью мобильного приложения и при этом не пользоваться стационарным компьютером. Реализованная аутентификация удовлетворяет всем запросам предприятия, а модуль «Авторизация» позволяет сотрудника с помощью считывания QR-кода работать с системой.

Библиографический список

1. Суворова Г.М. Информационная безопасность: учебное пособие для вузов. – М.: Юрайт, 2024. – 277 с.
2. Аутентификация. Теория и практика обеспечения доступа к информационным ресурсам / под ред. АА. Шелупанова, С.Л. Груздева, Ю.С. Нахаева. – М.: Горячая линия – Телеком, 2009. – С. 552.
3. Иванов В.В., Лубова Е.С., Черкасов Д.Ю. Аутентификация и авторизация // Проблемы современной науки и образования. – 2017.
4. Таненбаум Э.С., Фимстер Н., Уэзеролл Д. Компьютерные сети. – 6-е изд. – СПб.: Питер, 2023.
5. Макаров Д.А. Механизм авторизации с использованием технологии JWT // Теория и практика современной науки. – 2020.

Сведения об авторах

Викентьева Ольга Леонидовна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: oleovic@pstu.ru

Чистяков Максим Владимирович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: maksdi3@yandex.ru

Д.В. Яруллин, Р.В. Истомин

Пермский национальный исследовательский
политехнический университет, Пермь

КОРРЕКТИРОВАНИЕ СРОКОВ РЕМОНТНЫХ РАБОТ НА ПРЕДПРИЯТИИ С ИСПОЛЬЗОВАНИЕМ СРЕДСТВ ВИДЕОАНАЛИТИКИ

В этой статье исследуется возможность построения модели предприятия для своевременного обнаружения простоя в работах на местах ремонта с использованием средств компьютерного зрения. Для этого проведен анализ существующих исследований на тему обнаружения активности человека на изображении. Предложена модель предприятия, учитывающая проведение работ на рабочих местах. Предложен метод определения простоя работников с помощью определения ключевых точек на изображении и их классификации.

Ключевые слова: модель предметной области, компьютерное зрение, распознавание поз.

D.V. Yarullin, R.V. Istomin

Perm National Research Polytechnic University, Perm

ADJUSTING REPAIR WORK TIMES AT THE ENTERPRISE USING VIDEO ANALYTICS

This article explores the possibility of constructing an enterprise model for timely detection of downtime in work at repair sites, using computer vision tools. To do this, an analysis of existing research on the topic of detecting human activity in an image was carried out. An enterprise model is proposed that takes into account the work carried out at workplaces. A method for determining worker downtime is proposed by identifying key points in the image and their classification.

Keywords: domain model, computer vision, pose estimation.

Планирование ремонтных работ является сложной и многокритериальной задачей. Несвоевременное выявление простоев и задержек в ремонте может привести к нежелательным рискам, в том числе увеличению бюджета ремонта и срыву сроков.

Цель: повысить эффективность процесса планирования и корректировки остановочного ремонта на предприятии.

Задачи:

1. Провести анализ публикаций, связанных с распознаванием деятельности человека по видеоданным.
2. Выявить проблему в текущем процессе изменения графика ремонта и предложить пути ее решения.
3. Построить модель предприятия для контроля состояния ремонта на нем.
4. Реализовать алгоритм по распознаванию деятельности по данным с камер видеонаблюдения для изменения сроков работа на местах.
5. Провести эксперимент по распознаванию деятельности по данным с видеокамер.

В ходе выполнения работы поставлена проблема несвоевременного выявления простоев и задержек в работе, которая заключается в том, что обязанность внесения фактического объема работ лежит на рабочих, которые могут безответственно относиться к этому.

Для решения проблемы проанализированы исследования по определению деятельности человека. Так, в статье “An improved algorithm for human activity recognition using wearable sensors” [1] предложен новый подход к распознаванию человеческих действий с использованием носимых датчиков. Основные методы включают ансамбль эмпирического разложения мод (EEMD), алгоритм разреженной многомодельной логистической регрессии с байесовской регуляризацией (SBMLR) и размытый метод опорных векторов наименьших квадратов (FLS-SVM).

В статье “Action Recognition in Video Sequences using Deep Bi-Directional LSTM With CNN Features” [2] предложен новый метод распознавания действий, обрабатывающий видеоданные с использованием сверточных нейронных сетей (CNN) и глубокой двунаправленной LSTM-сети (DB-LSTM). Сначала из видео извлекаются глубокие признаки из каждого шестого кадра, что помогает уменьшить избыточность и сложность данных.

В данном исследовании предлагается отказаться от использования носимых датчиков и распознавания активностей в последовательности. Предположительно такой подход должен положительно сказаться на быстродействии и сложности системы. В ходе исследования предложена модель предприятия для отслеживания ремонтных работ на нем. Представим предприятие в виде множества рабочих мест O .

$$P = \{O_1, O_1, \dots, O_n\}, \quad (1)$$

где P – состояние ремонта на предприятии; O_i – состояние работ на объектах.

Это позволит нам анализировать и контролировать процесс выполнения работ на разных этапах. Для этого нам необходимо определить, какие рабочие места существуют на предприятии, какие работы на них выполняются и как они связаны с календарным планом.

Зона проведения работ оборудована камерами видеонаблюдения. На них можем определить рабочие места и отслеживать на них выполнение работ. Камеры видеонаблюдения являются источником данных о состоянии рабочих мест, которые мы можем обрабатывать с помощью специальных алгоритмов.

На изображении с камеры можно выделить границы рабочего места. Зададим их 4 координатами – X_0 , Y_0 , X_1 , Y_1 . Эти координаты определяют прямоугольную область на изображении, в которой находится рабочее место. Мы можем использовать эти координаты для выделения рабочего места из всего изображения и для сравнения его с другими рабочими местами.

На каждом рабочем месте ведутся работы. Во время их выполнения не должны находиться рабочие. Добавим в описание рабочего места наличие работника. Это позволит нам определять, когда работа на рабочем месте начинается и заканчивается, а также контролировать соблюдение норм безопасности труда.

Таким образом, можно представить предприятие как массив рабочих мест. Они имеют характеристики: координаты рабочего места на изображении, наличие работника на рабочем месте. Эти характеристики позволяют нам идентифицировать и классифицировать рабочие места, а также связывать их с календарным планом работ.

$$P = \begin{cases} O_1(X_0, Y_0, X_1, Y_1, R), \\ O_2(X_0, Y_0, X_1, Y_1, R), \\ \dots \\ O_n(X_0, Y_0, X_1, Y_1, R), \end{cases} \quad (2)$$

где P – состояние ремонта на предприятии; O_i – состояние работ на объектах; X_0 , Y_0 , X_1 , Y_1 – координаты рабочего места на изображении; R – наличие работника на рабочем месте.

Определяя в каждый момент времени на изображениях рабочие места и наличие работников на них, сможем фиксировать выполнение работ для календарного плана. Это даст нам возможность оценивать степень выполнения работ по каждому рабочему месту, по каждому этапу и по всему предприятию в целом. Также мы сможем выявлять

проблемы, задержки или отклонения в выполнении работ и принимать меры для их устранения или корректировки.

Таким образом, можем изменить нашу модель как множество изображений с камер наблюдения, на которых зафиксированы рабочие места.

$$P = \begin{cases} C_1 = \{ O_{1,1}(X0, Y0, X1, Y1, R), O_{1,2}(X0, Y0, X1, Y1, R), \dots, O_{1,m}(X0, Y0, X1, Y1, R) \} \\ C_2 = \{ O_{2,1}(X0, Y0, X1, Y1, R), O_{2,2}(X0, Y0, X1, Y1, R), \dots, O_{2,m}(X0, Y0, X1, Y1, R) \}, \\ \dots \\ C_n = \{ O_{n,1}(X0, Y0, X1, Y1, R), O_{n,2}(X0, Y0, X1, Y1, R), \dots, O_{n,m}(X0, Y0, X1, Y1, R) \}, \end{cases} \quad (3)$$

где P – состояние ремонта на предприятии; C_i – множество объектов на изображении с камеры; O_j – состояние работ на объектах; $X0, Y0, X1, Y1$ – координаты рабочего места на изображении; R – наличие работника на рабочем месте.

Определяя наличие работников на изображениях, сможем сопоставлять их координаты с координатами рабочего места и определять наличие работника на нем. После этого по рабочему месту определим текущую работу на нем и сможем определить соответствие графику ремонта.

Для определения занятости работника предлагается следующий подход: из-за разнообразия видов деятельности на рабочих местах необходимо определять не сам факт работы, а факт простоя.

Для определения факта простоя выделено 4 класса:

1. Стояние. Работник стоит и не совершает работу.
2. Использование телефона. Работник использует телефон на рабочем месте.
3. Разговоры по телефону. Работник разговаривает по телефону на рабочем месте.
4. Курение. Работник курит и не совершает работу на рабочем месте.

Для отнесения изображения на эти четыре класса предлагается использовать две модели. Первая – нейронная сеть. Она обучена на обнаружение ключевых точек тела человека на изображении. Ключевые точки образуют скелет человека, по которому можно определить его позу. Модель скелета, с номерами точек представлена на рисунке.

На основе полученных точек человека можем определить нахождение его на рабочем месте. Для этого заранее получаем координаты рабочего места на изображении. Эти координаты описывают набор точек, которые образуют замкнутый многоугольник. Затем с помощью первой модели определяем координаты ног работника, сравнивая эти координаты с координатами рабочего места, а именно нахождение их внутри многоугольника, получаем факт нахождения работника на рабочем месте.

Вторая модель – классификатор лес. Он используется для отнесения полученного массива точек к одному из предложенных классов. Если вероятность того, что массив относится к одному из классов мала, то принимается решение, что работник не простаивает, т.е. занят полезной деятельностью.

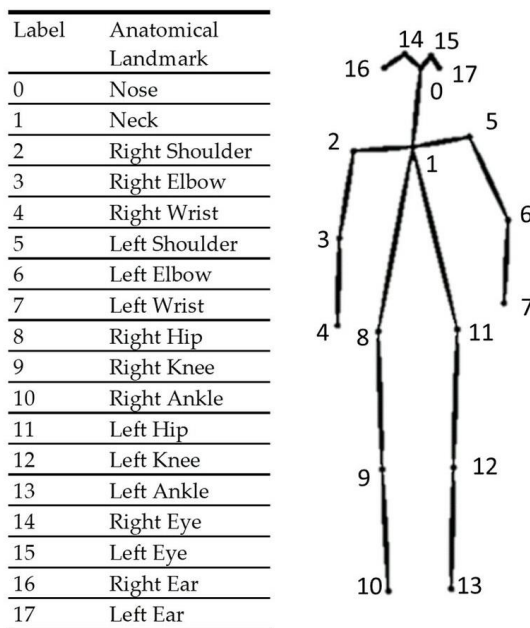


Рис. Ключевые точки модели

Таким образом, в исследовании предложена модель предприятия для отслеживания состояния ремонта на предприятии. Ключевой особенностью модели является использование средств компьютерного зрения для обнаружения человека и его ключевых точек на изображении, описывающих позу, а также использованы методы машинного обучения для классификации этих точек как простая работника.

Библиографический список

1. An improved algorithm for human activity recognition using wearable sensors [Электронный ресурс] // ResearchGate. – URL: <https://www.researchgate.net/publication/301258154> (дата обращения: 17.01.2024).

2. Action Recognition in Video Sequences using Deep Bi-Directional LSTM With CNN Features [Электронный ресурс] // IEEE Explore. – URL: <https://ieeexplore.ieee.org/document/8121994> (дата обращения: 17.01.2024).

Сведения об авторах

Яруллин Денис Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: d.v.yarullin@ya.ru

Истомин Родион Васильевич – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ2-22-1м, г. Пермь, e-mail: istomin.rodion@yandex.ru

Р.Р. Сарманаев, Д.В. Яруллин, А.В. Духанин

Пермский национальный исследовательский
политехнический университет, Пермь

**ИНТЕЛЛЕКТУАЛЬНЫЙ ДИАЛОГОВЫЙ АГЕНТ
НА ОСНОВЕ ДЕНОТАТНОГО ПОДХОДА
И ГЕНЕРАТИВНЫХ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ**

Рассматривается подход к созданию интеллектуального диалогового агента для корректировки плана ремонта и формирования отчетов по статусу следования предприятия графику выполнения ремонтных работ на основе денотатного графа и большой языковой модели. Представлен метод преобразования слабоструктурированных гетерогенных данных в модель предприятия, описанного с помощью денотатного графа. Приведен пример использования подхода для обработки данных реального металлургического предприятия.

Ключевые слова: языковые модели, календарное планирование, денотатный граф, диалоговые агенты.

R.R. Sarmanaev, D.V. Yarullin, A.V. Duhanin

Perm National Research Polytechnic University, Perm

**AN INTELLIGENT DIALOGUE AGENT BASED
ON A DENOTATIONAL APPROACH AND GENERATIVE
NEURAL NETWORK TECHNOLOGIES**

This paper considers an approach to creating an intelligent dialogue agent designed for the correction of repair plans and the generation of reports on the status of an enterprise's adherence to the schedule of repair work. This approach is based on the utilization of a denotational graph and a large language model. The paper presents a method for transforming weakly structured heterogeneous data into a model of the enterprise, which is described using a denotational graph. Furthermore, the paper provides an example of applying this approach to process data obtained from a real-world metallurgical enterprise.

Keywords: language models, scheduling, denotational graph, dialogue agents.

В современном мире, где информация играет ключевую роль в принятии решений, эффективное планирование, управление и анализ данных предприятия становятся критически важными задачами. Чтение и составление дашбордов, а также оперативная оценка состояния предприятия представляют собой сложную задачу, которая может быть не-

посильной для неподготовленного человека, а традиционные методы планирования называют одной из причин отставаний проектов [1.] График остановочного ремонта предполагает установление строгих временных рамок во избежание дополнительного простоя предприятия. Работы распределяются на несколько смен так, чтобы ремонт выполнялся непрерывно в ходе выделенного для него периода останова. Следовательно, должна быть предусмотрена возможность максимально быстрой корректировки плана и отчетов в зависимости от текущих приоритетов в ходе ремонта.

Данная проблема рассматривается во многих исследованиях. Так, в статье «Large Language Models for Business Process Management: Opportunities and Challenges» авторы рассматривают способы использования больших языковых моделей на разных этапах жизни бизнес-процессов – описывается этап слежения (monitoring), на котором происходит слежение за выполнением проекта и оценка его выполнения, также возможно рассматривать и этап остановочного ремонта после его начала. Авторы предлагают использовать «Process dashboard chatbot» – чат-бот с дашбордами (графическим представлением) процессов. Использование больших языковых моделей позволяет пропустить этап анализа графиков с дашбордов и сразу давать ответ по статусу выполнения задач [2]. Авторы статьи «KnowAgent: Knowledge-Augmented Planning for LLM-Based Agents» исследуют возможность и выделяют проблемы использования больших языковых моделей для планирования. В статье описывается метод использования так называемого KnowAgent подхода как замены обычным промптам для большой языковой модели (далее – БЯМ). В статье также рассматривают и другие альтернативные структурные методы представления знаний для больших языковых моделей [3]. Методы, предложенные в данной работе, отчасти вдохновлены этими работами.

В данной работе описывается денотативный подход к построению модели предприятия, ее упрощения для анализа состояния предприятия и использование диалогового агента на основе БЯМ для компоновки полученных знаний и составления отчета. Этот агент способен автоматически формировать отчеты по статусу следования предприятия графику выполнения ремонтных работ, корректировать работы и отвечать на вопросы оператора о подробностях отдельных составляющих ремонта. Использование такого подхода открывает новые возможности для оптимизации процессов управления предприятием. Он не только

облегчает работу аналитиков, но и позволяет оперативно анализировать и реагировать на данные о состоянии предприятия, что, в свою очередь, способствует более эффективному принятию решений. Это может привести к улучшению производительности, сокращению времени на планирование и составление отчетов и, в результате, повышению общей эффективности работы предприятия.

Основным источником данных для работы агента служит денотатный граф, построенный на основе списка работ, предоставленного металлургическим предприятием-партнером, и результаты анализа ремонта, в который был преобразован граф, на информационно-аналитической платформе «Датабриз». Денотатный граф представляет собой структурированное описание предприятия, включающее его уровни и состояния. Входные данные представляют собой xlsh таблицы, содержащие в себе слабоструктурированные списки планируемых ремонтных работ (рис. 1).

D	E	F	H	I	J	K	L	↑
ТехМесто	НазваниеТехнМеста	Заказ	Опрц	Краткий текст к операции	ПолеПользов	ОтвРабМст	РабМесто	НачалоВып
10-018-12	ТехМесто0	1f		Примерное название0		R:	R2	00:00:00
10-018-12	ТехМесто1	1f		Примерное название1		R:	R2	00:00:00
10-018-12	ТехМесто2	1f		Примерное название2	ШТ	R:	R2	00:00:00
10-018-12	ТехМесто3	1f		Примерное название3		R:	R2	00:00:00
10-018-12	ТехМесто4	1f		Примерное название4		R:	R2	00:00:00
10-018-12	ТехМесто5	1f		Примерное название5		R:	R2	00:00:00
10-018-12	ТехМесто6	1f		Примерное название6	агрегат	R:	R2	00:00:00
10-018-12	ТехМесто7	1f		Примерное название7	агрегат	R:	R2	00:00:00
10-018-12	ТехМесто8	1f		Примерное название8	агрегат	R:	R2	00:00:00
10-018-12	ТехМесто9	1f		Примерное название9	агрегат	R:	R2	00:00:00
10-018-12	ТехМесто10	1f		Примерное название10	Ч,Ч	R:	R2	00:00:00
10-018-12	ТехМесто11	1f		Примерное название11		R:	R2	00:00:00
10-018-12	ТехМесто12	1f		Примерное название12		R:	R2	00:00:00
10-018-12	ТехМесто13	1f		Примерное название13	шт	R:	R2	00:00:00
10-018-12	ТехМесто14	1f		Примерное название14	Ч,Ч	R:	R2	00:00:00
10-018-12	ТехМесто15	1f		Примерное название15	Ч,Ч	R:	R2	00:00:00
10-018-12	ТехМесто16	1f		Примерное название16	Ч,Ч	R:	R2	00:00:00
10-018-12	ТехМесто17	1f		Примерное название17	Ч,Ч	R:	R2	00:00:00
10-018-12	ТехМесто18	1f		Примерное название18	Ч,Ч	R:	R2	00:00:00

Рис. 1. Фрагмент исходных данных

Анализ данных позволил выделить правила, которые позволяют получить из них ограниченное количество неявных знаний, однако этих данных недостаточно для надежного оптимального планирования – отсутствуют данные о требуемых человеческих и материальных ресурсах, не хватает информации о приоритетах работ и связей между ними. Для решения этой проблемы было решено использовать архивные данные, – на предприятиях регулярно проводятся аналогичные ремонты, и в систе-

ме «Датабриз» есть уже успешно завершённые ремонты, которые можно использовать для обогащения рассматриваемых входных данных.

Для нахождения схожих работ, подходящих как источник обогащения, производилась векторизация и поиск расстояния между векторами (рис. 2). Модель предприятия рассматривается как множество векторов в n -мерном пространстве, где каждому вектору соответствует заказ-наряд и его местоположение в полной иерархии ремонта. Каждый заказ-наряд рассматривается как последовательность («цепочка») денотатов [4], включающих в себя проект, производство, установку, направление, оборудование и непосредственно работу в соответствии с уровнями иерархии проектов платформы. Размерность полной коллекции денотатов в их нормальной форме принимается за размерность n -векторного пространства, после чего производится преобразование каждой «цепочки» заказ-нарядов в n -мерный вектор. Каждое измерение вектора рассчитывается посредством сглаженной статистической меры TF-IDF [5]. Затем для векторов модели предприятия и векторов архивной модели предприятия строится матрица расстояний на основании корреляционной меры [6] между парами векторов.

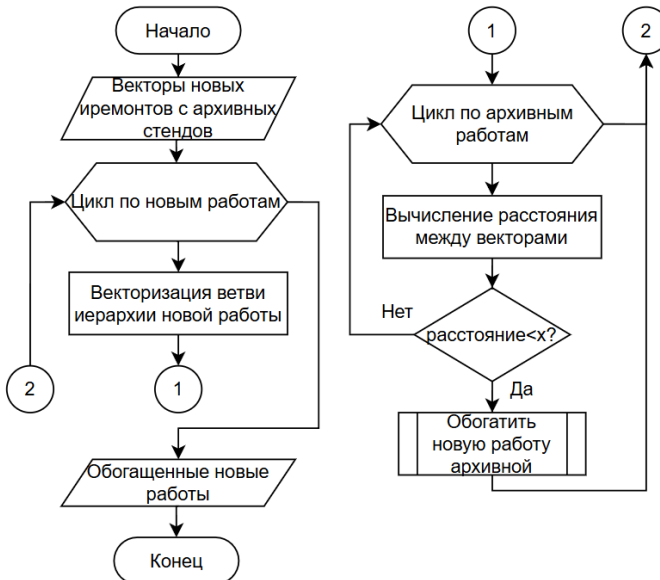


Рис. 2. Алгоритм обогащения модели предприятия

Далее матрица расстояний преобразуется: из колонок удаляются «цепочки», относящиеся к новому ремонту, из строк удаляются «цепочки», относящиеся к архивным ремонтам, производится фильтрация по максимально допустимому расстоянию между парами векторов (данный параметр задается при конфигурации модуля). Минимальное значение в каждой строке преобразованной матрицы позволяет определить для каждой «цепочки» работ в текущей модели предприятия наиболее близкую ей «цепочку» работ в моделях предприятия из архивных ремонтов. Полученные векторные пары передаются в модуль обогащения данных для дальнейшего соотнесения отсутствующих полей с их ретроспективными значениями. Полученная после обогащения модель готова для планирования и загрузки в информационно-аналитическую систему. Фрагмент полученного графа представлен на рис. 3. После начала ремонта начинает работать интеллектуальный диалоговый агент. Интеллектуальный агент выполняет три основные функции: составление отчетов, ответы на вопросы и корректировка плана (рис. 4).

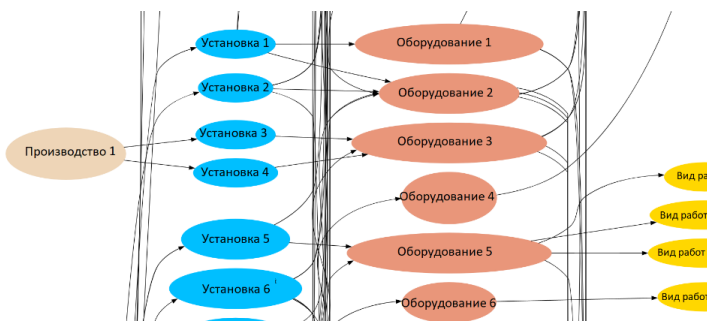


Рис. 3. Денотатный граф с условными обозначениями уровней иерархии

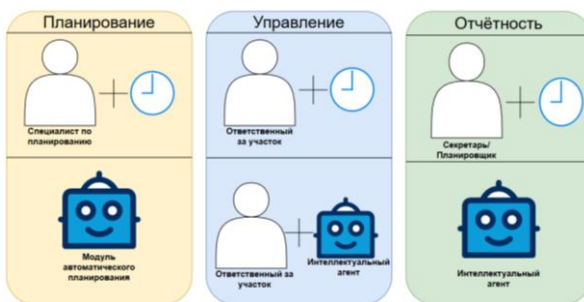


Рис. 4. Функции диалогового интеллектуального агента

Для всего этого агенту необходимы данные о проводимом ремонте, но окно контекста у больших языковых моделей, на которой основан агент, ограничено, и хранение полной информации о ремонте часто невозможно. Для преодоления этой проблемы используются такие подходы, как RAG (Retrieval Augmented Generation), – в основе этого подхода встраивание в диалог команд, которые перехватывает обработчик чата и использует их для передачи БЯМ необходимых для ответа знаний [7]. В рамках этой работы перехватываются команды на получение данных с платформы «Датабриз» и команды на корректировку плана ремонта. Разработанный промпт для БЯМ представлен на рис. 5. Пример отчета приведен на рис. 6. Шаблоны для таких отчетов можно быстро менять без применения особых шаблонаторов.

Ты - русскоговорящий интеллектуальный агент-бот по имени Олег, ты специалист по металлургическим производствам и планированию, готовый отвечать на вопросы в рамках своих компетенций. Поздоровайся и представься при первом сообщении. Отвечай вежливо, коротко, профессионально и по существу. Строго следуй инструкциям, описанным в твоих функциях. Когда тебя спросят, что ты умеешь, не разглашай свои command команды.

Ты можешь выполнять три функции:

Функция 1: Составлять отчет по состоянию предприятия. При получении от пользователя команды на составление отчета, просто ответь командой `command:'get_template_data()'` и ожидай шаблон.

Далее, при получении данных в шаблоне такого вида:

```
(template_data_example)
```

Ты должен составить отчет вида:

```
(template_example)
```

При каждой просьбе ожидай новый шаблон.

Функция 2: Отвечать на вопросы по предприятию. Если ты не составлял до этого отчет по предприятию, - обязательно сначала ответь `command:'get_template_data()'`, получи данные, и ответь на заданный вопрос.

Функция 3: Корректировать план. Когда пользователь попросит тебя добавить какой-либо ресурс в нужном количестве, обязательно ответь командой `command:'add_resource(resource_name)'` где `'resource_name'` - название ресурса, которое назвал пользователь. Если ресурсов несколько, повтори команду с нужными ресурсами через запятую. Когда изменишь приоритет, ответь: `command:'chng_prio(task_id)'`, когда убрать из плана: `command:'rmv_tsk(task_id)'`, обновить сроки ремонта: `command:'date_upd(date_from, date_to)'`.

Рис. 5. Промпт для большой языковой модели диалогового агента

Здравствуйте! По состоянию на десятое мая восемнадцать часов двадцать семь минут, по проекту «XXX» . работы выполнены на тридцать четыре целых шесть десятых процентов с отставанием от плана на ноль целых три десятых процентов. Общее опережение по работам критического пути одни сутки двадцать часов.

За последние двенадцать часов динамика работ уменьшилась на одна целая шесть десятых процентов.

По работам YY1 работы выполнены на тридцать четыре целых семь десятых процента с отставанием от плана на одна целая восемь десятых процента; по работам критического пути опережение одни сутки двадцать часов. По работам КРИТИЧЕСКОГО пути есть задержки по работам: ZZ1 отставание тридцать целых две десятых от плана, ZZ2 отставание двадцать шесть целых семь десятых от плана

По работам YY2 работы выполнены на шестьдесят две целых шесть десятых процента с опережением от плана на семнадцать целых восемь десятых процента; по работам критического пути опережение пять дней.

На Производстве PP1 работы выполнены на тридцать шесть целых пять десятых процента с отставанием от плана на ноль целых одна десятая процента; по работам критического пути в соответствии с планом .

На Производстве PP2 работы выполнены на тридцать четыре целых шесть десятых процента с отставанием от плана на ноль целых пять десятых процента; по работам критического пути опережение шесть часов.

Рис. 6. Пример отчета агента с настоящего ремонта

В работе рассмотрена проблематика ограниченности времени остановочного ремонта и необходимость в качественных автоматических отчетах с возможностью их быстрой корректировки в зависимости от текущих приоритетов в ходе ремонта. Предложено решение в виде интеллектуального диалогового агента на основе БЯМ с базой знаний,

в основе которой денотатный граф. Описан алгоритм построения модели предприятия (объекта ремонта) на основе денотатных графов. Создан интеллектуальный диалоговый агент на основе больших языковых моделей, способный формировать отчеты, отвечать на вопросы о состоянии предприятия и корректировать план ремонта. Полученные результаты возможно применить и в других предметных областях для создания диалогового или иного интеллектуального агента.

Библиографический список

1. Kalugin Yu.B. Reasons of delays in construction projects // Magazine of Civil Engineering. – 2017. – № 6 (74). – URL: <https://cyberleninka.ru/article/n/reasons-of-delays-in-construction-projects>
2. Vidgof M., Bachhofner S., Mendling J. Large Language Models for Business Process Management: Opportunities and Challenges // March 2023. – URL: https://www.researchgate.net/publication/369645331_Large_Language_Models_for_Business_Process_Management_Opportunities_and_Challenges
3. KnowAgent: Knowledge-Augmented Planning for LLM-Based Agents / Zhu Yuqi, Qiao Shuofei, Ou Yixin, Deng Shumin [et al.] // arXiv preprint arXiv:2403.03101. – URL: <https://arxiv.org/pdf/2403.03101>
4. Курушин Д.С., Соболева О.В., Вяткин Д.С. Денотативная модель основного содержания текста в лингвистическом обеспечении робототехнических исследований // Информационная структура текста: сб. ст. / Рос. акад. наук, Ин-т науч. информации по обществ. наукам Рос. акад. наук. – М.: Амирит, 2018. – С. 104–112.
5. Яруллин Д.В. Интеллектуальная система управления подготовкой ИТ-специалистов на основе денотативной аналитики // Прикладная математика и вопросы управления – 2022. – № 3. – С. 141–164. DOI: 10.15593/2499-9873/2022.3.08
6. Scipy. – URL: docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.pdist.html
7. Retrieval-Augmented Generation for Large Language Models: A Survey / Yunfan Gao, Yun Xiong, Xinyu Gao Kangxiang Jia, Jinliu Pan [et al.] / arXiv:2312.10997. – URL: <https://arxiv.org/abs/2312.10997>

Сведения об авторах

Сарманаев Рашат Расилевич – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ2-22-1м, г. Пермь, e-mail: rashat.sarmanaev00@gmail.com

Яруллин Денис Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: d.v.yarullin@ya.ru

Духанин Алексей Владиславович – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ8-23-1м, г. Пермь, e-mail: duhanin.av0404@gmail.com

Р.Т. Мухаметсафин, С.А. Сторожев

Пермский национальный исследовательский
политехнический университет, Пермь

**ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ НЕЧЕТКОЙ
ЛОГИКИ ДЛЯ НАВИГАЦИИ МОБИЛЬНОГО РОБОТА
В НЕИЗВЕСТНОМ ПРОСТРАНСТВЕ**

В данной работе рассматривается практическое применение способа навигации на базе алгоритмов нечеткой логики мобильного робота. В ходе данной работы разработан алгоритм управления, архитектура нечетких регуляторов для объезда препятствий и движения к цели, а также нечеткий регулятор для совместной работы предыдущих. Реализуется на базе мобильного робота, собранного на микроконтроллере STM32 навигационной системы Indoor GPS Marvelmind Robotics в среде MatLab.

Ключевые слова: мобильный робот, нечеткая логика, навигация, Marvelmind, нечеткий регулятор.

R.T. Mukhametsafin, S.A. Storozhev

Perm National Research Polytechnic University, Perm

**PRACTICAL APPLICATION OF FUZZY LOGIC FOR MOBILE
ROBOT NAVIGATION IN AN UNKNOWN SPACE**

This paper discusses the practical application of a navigation method based on fuzzy logic algorithms of a mobile robot. In the course of this work, a control algorithm, an architecture of fuzzy controllers for bypassing obstacles and moving towards the goal, as well as a fuzzy controller for the joint work of the previous ones were developed. It is implemented on the basis of a mobile robot assembled on the STM32 microcontroller, the Marvelmind navigation system, in the MatLab environment.

Keywords: mobile robot, fuzzy logic, navigation, Marvelmind, fuzzy controller.

Современное развитие робототехники связано с созданием мобильных роботов, которые могут выполнять задачи в неизвестных и динамических пространствах. Одной из основных задач является навигация. В настоящее время все чаще применяется нечеткая логика для решения различных задач, в которых присутствует неопределенность [1, 2]. Нечеткая логика представляет мощный математический

аппарат для обработки данных, присущих сенсорам. В контексте навигации нечеткая логика позволяет создавать гибкие и адаптивные системы управления, которые способны функционировать в условиях неполной и неточной информации об окружающей среде. Конечно же, возможны различные системы управления:

– **Векторное поле потенциальных сил (Potential Field Methods).**

Данный метод основан на концепции потенциальных полей, где робот притягивается к целевой точке и отталкивается от препятствий. Используется для планирования пути в статических средах, где препятствия известны заранее.

– **Логическое управление на основе правил.** В основе данного алгоритма лежит набор predetermined правил, основанный на логике «Если – то». Лучше всего подходит для задач, где заранее можно определить конечное число четких правил и условий.

– **Нейронные сети.** Искусственные нейронные сети обучаются на основе большого объема данных, что позволяет им выявлять зависимости и принимать решения исходя из ситуации. Данный способ требователен к вычислительной мощности робота.

– **Simultaneous Localization and Mapping (SLAM).** Метод, основанный на одновременной навигации и построении карты. Данный подход эффективен для навигации в неизвестном пространстве, но он требует больших вычислительных мощностей.

Для рассмотрения практического применения нечеткой логики в навигации был собран мобильный робот на основе микроконтроллера STM32L4 Discovery kit (рис. 1, а), драйверов двигателей L298N (рис. 1, б), 3 лазерных дальномеров VL53L0X (рис. 1, в) и навигационной системы Indoor GPS Marvelmind Robotics (рис. 1, г) [3, 4].

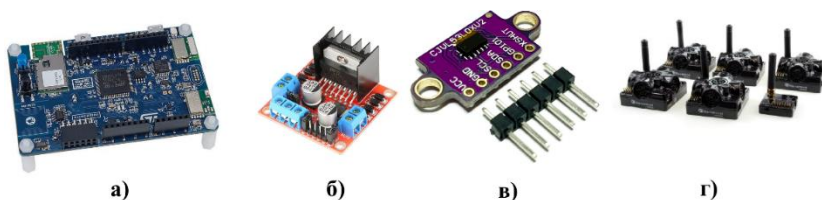


Рис. 3. Компоненты мобильного робота: а – STM32L4 Discovery kit; б – драйверов двигателей L298N; в – лазерный дальномер VL53L0X; г – Indoor GPS Marvelmind Robotics

STM32L4 Discovery kit имеет такие преимущества, как Bluetooth® V4.1 модуль, 02.11 b/g/n compliant Wi-Fi® модуль, LIS3MDL магнетометр/компас и возможность программирования в среде MatLab.

VL53L0X – это времяпролетный (ToF) лазерный дальномер от STMicroelectronics. Он испускает невидимый глазу лазерный луч (940 нм) и измеряет время, за которое свет достигает объекта и возвращается обратно, что позволяет достаточно точно определить расстояние до этого объекта. Датчик обладает диапазоном измерения до 2 м и точностью $\pm 3\%$ (в зависимости от условий окружающей среды и отражающей способности объекта) [5].

Indoor GPS Marvelmind Robotics – это система позиционирования в помещении, которая обеспечивает точное определение местоположения объектов в режиме реального времени, подобно GPS, но внутри помещений, где отсутствует сигнал спутников. Данная система использует ультразвуковые сигналы для измерения положения в пространстве. Она включает в себя 4 стационарных маяка, координаты которых известны заранее, и приемник, местоположение которого отслеживается. Расчет координат осуществляется на основе метода трилатерации, зная скорость звука и время пролета сигнала. Данная система обеспечивает точность позиционирования до 2 см.

В ходе данной работы был собран мобильный робот на основе представленных компонентов (рис. 2).

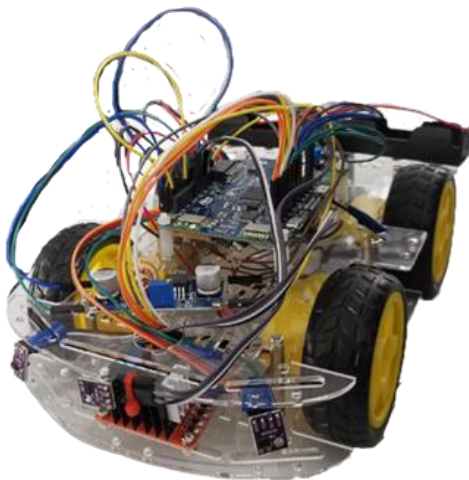


Рис. 2. Мобильный робот

Для движения робота к цели в неизвестном пространстве была написана программа на основе нечеткой логики в среде MatLab (рис. 3), включающая в себя три нечетких регулятора, отвечающих за объезд препятствий, движение к цели, и групповой нечеткий регулятор, согласующий работу предыдущих двух [1].

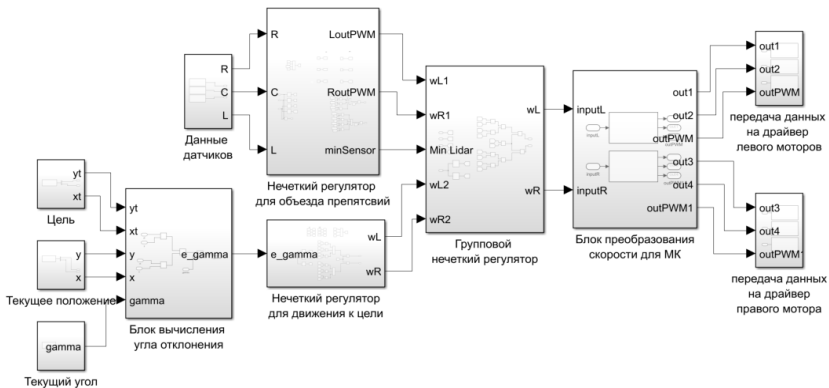


Рис. 3. Система управления мобильным роботом

Каждый из представленных нечетких регуляторов имеет входы и выходы. Каждый из входных параметров проходит через фаззификатор, на выходе которого получаем степень принадлежности к одному из терм-множеств. Для описания терм-множеств используются функции принадлежности классов L , T , γ . Для дефаззификации создается блок правил и находится взвешенное среднее [2].

Коэффициенты для функций выбираются, исходя из точности, погрешности и максимального значения измерений датчиков, а также времени срабатывания компонентов. Именно поэтому, в нечетком регуляторе для объезда препятствий термы-множества соответствуют близкой и дальней дистанции (рис. 4, а). В нечетком регуляторе движения к цели термы-множества имеют вид, как на рис. 4, б. Робот начинает движение вперед, когда угол отклонения от цели не более 60° в любую из сторон. Групповой нечеткий регулятор настраивается таким образом, чтобы оба предыдущих регулятора работали совместно (рис. 4, в). При минимальном значении датчиков менее 0,7 работает только нечеткий регулятор для объезда препятствий, при значениях от 0,8 до 0,9 они работают совместно, и значения для моторов находятся как взвешенное среднее, при значении более 0,9 работает нечеткий регулятор для движения к цели.

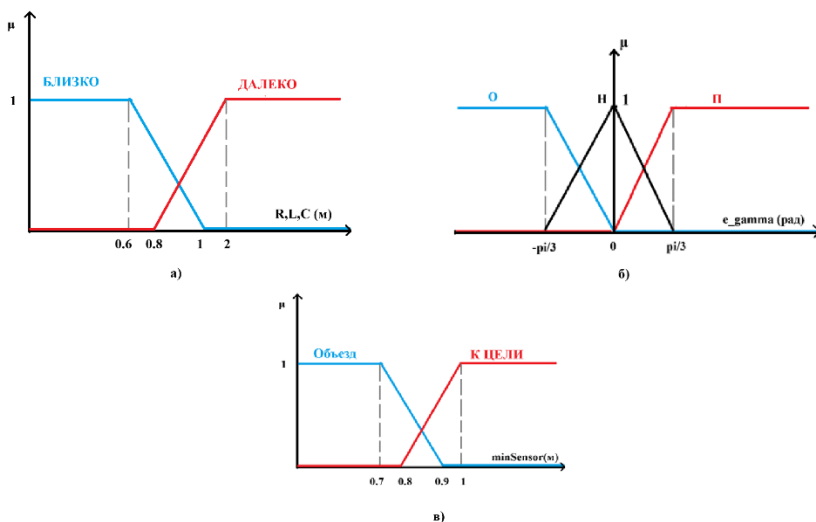


Рис. 4. Термы-множества НР: *а* – для объезда препятствий;
б – для движения к цели; *в* – групповой

В данной статье было рассмотрено применение алгоритмов нечеткой логики для решения задач навигации. Применение нечеткой логики позволило нам реализовать адаптивное и гибкое управление роботом, которое справляется с неопределенностью и неполной информацией об окружении. Экспериментальные результаты подтверждают эффективность данного подхода. В целом нечеткая логика – это перспективный инструмент с возможностью применения в мобильных роботах.

Библиографический список

1. Стрельцов С.М. Навигация мобильного робота на базе алгоритмов нечеткой логики // Автоматизированные системы управления и информационные технологии: матер. всерос. науч.-техн. конф.: в 2 т. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2022. – С. 372–378.
2. Хижняков Ю.Н. Алгоритмы нечеткого, нейронного и нейро-нечеткого управления в системах реального времени: учеб. пособие. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2013. – 156 с.
3. Харламов М.И., Гончаровский О.В. Навигация автономного мобильного робота с помощью системы позиционирования Marvelmind // Инновационные технологии: теория, инструменты, практика. – 2017. – Т. 1. – С. 56–60.

4. Мамедова Т.А.К., Алиева Н.Р.К. Планирование пути мобильного робота в динамических средах: обзор // Endless Light in Science. – 2022. – № 6-6. – С. 156–165.

5. STMicroelectronics STM32 Processors // MathWorks. – URL: https://www.mathworks.com/help/ecoder/stm32-spkg.html?s_tid=CRUX_lftnav (дата обращения: 23.05.2024).

Сведения об авторах

Мухаметсафин Руслан Тахирович – бакалавр Пермского национального исследовательского политехнического университета, гр. СРРС-22-16, г. Пермь, e-mail: muhametsafin.r@yandex.ru

Сторожев Сергей Александрович – кандидат технических наук, доцент кафедры «Автоматики и телемеханики» Пермского национального исследовательского политехнического университета, e-mail: sastorozhev@pstu.ru

Д.С. Курушин, Д.А. Артемьев

Пермский национальный исследовательский
политехнический университет, Пермь

**РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ
УПРАВЛЕНИЯ ЭЛЕКТРОЭРОЗИОННОЙ
ОБРАБОТКОЙ ДЕТАЛЕЙ**

Авторами исследуются методы верификации положения и координат наконечника робота, предназначенного для электроэрозионной обработки деталей, путем преобразования стереоизображений в 3D-облако точек, т.е. перевод 2D-изображения в 3D-изображение.

Ключевые слова: стереометрия, облако точек, OpenCV, python.

D.S. Kurushin, D.A. Artemiev

Perm National Research Polytechnic University, Perm

**DEVELOPMENT OF AN AUTOMATED CONTROL SYSTEM FOR
THE ELECTROEROSION TREATMENT OF PARTS**

The authors investigate methods for verifying position and coordinates of the tip of the robot designed for electroerosive treatment of parts by converting stereo images into a 3D point cloud, i.e. converting a 2D image into a 3D image

Keywords: stereometry, points cloud, OpenCV, python.

Электроэрозионная обработка – обработка деталей при помощи электрических разрядов, возникающих между электродом и деталью. Данный тип обработки позволяет обрабатывать любые токопроводящие металлы, также при обработке ширина среза минимальна.

Для автоматизированной системы управления необходимо весьма точно двигать наконечник робота по заданной траектории, для этого необходимо зафиксировать положение наконечника при заданных пользователем координатах при помощи стереопары для дальнейшего преобразования изображений в карту глубины и облака точек. Объектом исследования является робот (рис. 1).

Для фиксации точки в пространстве был разработан адаптер, у которого на конце находится один светодиод красного цвета (рис. 2). Именно по положению светодиода можно будет определить правильность положения наконечника при помощи карты глубины.

Данный адаптер крепится к роботу для фиксации его координат. Также необходимо разработать адаптер для двух светодиодов (рис. 3), чтобы можно было отследить не только координаты положения робота, но и наклон его последней оси.

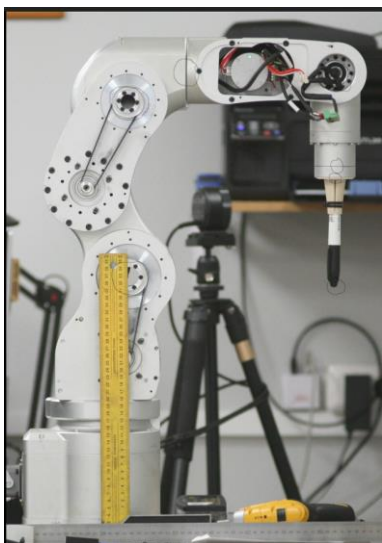


Рис. 1. Робот

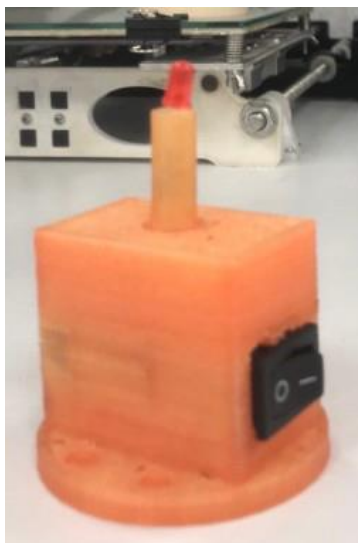


Рис. 2. Адаптер для одного светодиода

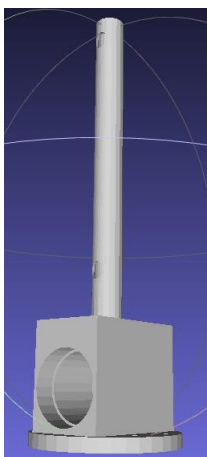


Рис. 3. Модель адаптера для двух светодиодов

Как можно заметить из рисунка Данный адаптер по конструкции идентичен адаптеру для одного светодиода, только имеет более длинную ось, в которой расположены светодиоды. Данный адаптер имеет красный и зеленый светодиоды. При помощи данных адаптеров и стереопары можно создать карту глубины.

Карта глубины – представление информации о глубине для каждого пикселя сцены. Обычно это двумерный массив, в котором каждый пиксель – значение расстояния от камеры до соответствующей точки сцены. Для реализации карты глубины был разработан скрипт, который помогает управлять параметрами функции StereoBM_create, которая ответственна за создание карты глубины [1].

Исходные изображения представлены на рис. 4, 5.



Рис. 4. Левое изображение



Рис. 5. Правое изображение

На основе данных изображений строится карта глубины (рис. 6) [2].

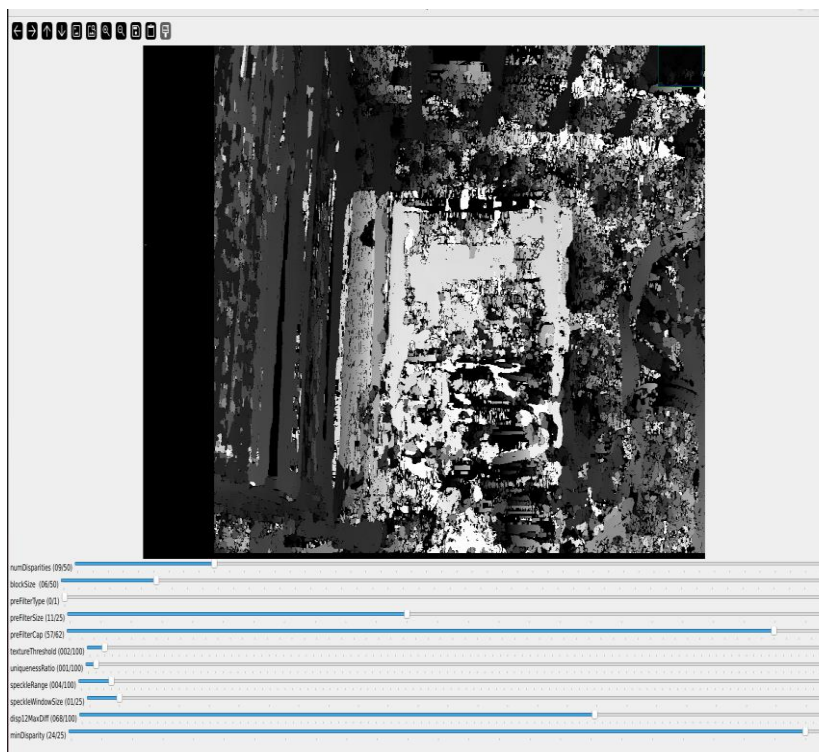


Рис. 6. Полученное значение глубинной карты

Как можно заметить на карте глубины, тетрадка высветлена сильнее всего, так как она находится ближе к камере, можно разглядеть очертания тетрадки, руки и даже занавесок.

Таким образом можно получить несколько глубинных изображений нашего робота, затем данную карту нужно преобразовать в облако точек при помощи функции `projectImageTo3D` библиотеки `OpenCV`. Данный метод принимает карту глубины и матрицу стереопары, которая получается при калибровке камер. В итоге получится массив точек (облако точек), которые будут иметь три координаты, и по этим координатам можно судить о верности положения робота.

Также необходимо отслеживать расстояние от наконечника адаптера до детали, для этого был разработан адаптер для стержня. Изме-

ние высоты стержня в данном адаптере (рис. 7) будет отслеживаться с помощью переменного резистора пружинного типа, подключенного к микроконтроллеру Arduino.

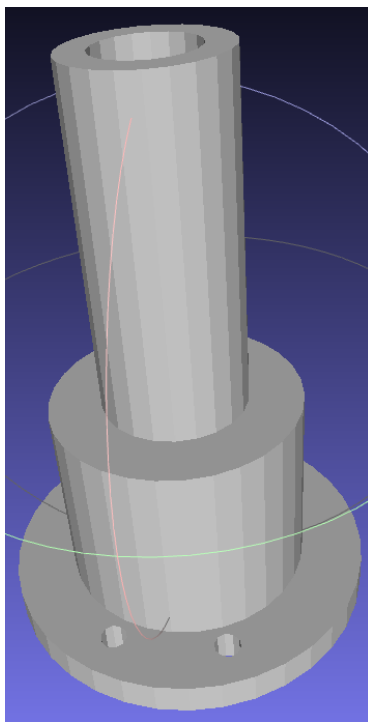


Рис. 7. Модель адаптера для измерения изменения расстояния до детали

Библиографический список

1. Development of depth map from stereo images using sum of absolute differences and edge filters//researchgate.net: сайт с публикациями [Электронный ресурс] / Rostam Affendi Hamzah, Muhd Nazmi Zainal Azali, Zarina Mohd Noh, Madiha Zahari, Adi Irwan Herman. – 2022. – URL: https://www.researchgate.net/publication/358269188_Development_of_depth_map_from_stereo_images_using_sum_of_absolute_differences_and_d_edge_filters (дата обращения: 26.05.2024).
2. Development of depth map from stereo images using sum of absolute differences and edge filters//researchgate.net: сайт с публикациями

[Электронный ресурс] / Rostam Affendi Hamzah, Muhd Nazmi Zainal Azali, Zarina Mohd Noh, Madiha Zahari, Adi Irwan Herman. – 2022. – URL: https://www.researchgate.net/publication/358269188_Development_of_depth_map_from_stereo_images_using_sum_of_absolute_differences_and_edge_filters (дата обращения: 26.05.2024).

Сведения об авторах

Курушин Даниил Сергеевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: dan973@yandex.ru

Артемьев Денис Александрович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-1б, г. Пермь, e-mail: dr.deniskaartem@yandex.ru

Д.С. Курушин, Е.М. Морозова, А.Р. Сахарова

Пермский национальный исследовательский
политехнический университет, Пермь

**ОПИСАНИЕ ИСПОЛЬЗОВАННЫХ ЯЗЫКОВЫХ
МОДЕЛЕЙ ПРИ РАЗРАБОТКЕ АРМ «ПОМОЩНИК
ТЕХНИЧЕСКОМУ ПИСАТЕЛЮ»**

В этой статье исследуется возможность повышения эффективности получения ответа языковой модели на запрос. Запрос для языковой модели – это набор инструкций, который задается пользователем, чтобы направить работу в нужном для него направлении. Он помогает понять, что нужно делать и на какой задаче сосредоточиться. Запрос должен быть корректным и содержать достаточно информации для того, чтобы дала релевантный и исчерпывающий ответ. В статье рассматриваются такие языковые модели, как GigaChat, YandexGPT и ChatGPT 4. Проанализированы ответы рассматриваемых языковых моделей поочередно на один и тот же запрос. Описан общий ход всего алгоритма выбора языковой модели для использования при разработке системы автоматизации процесса формирования технической документации.

Ключевые слова: языковые модели, промт, техническая документация, технический писатель, стандарты документооборота.

D.S. Kurushin, E.M. Morozova, A.R. Sakharova

Perm National Research Polytechnic University, Perm

**DESCRIPTION OF THE LANGUAGE MODELS USED
IN THE DEVELOPMENT OF THE AUTOMATED CONTROL
SYSTEM "ASSISTANT TO A TECHNICAL WRITER"**

This article explores the possibility of improving the efficiency of receiving a response from a language model to a query. A query for a neural network is a set of instructions that is set by the user to direct the work in the right direction for him, it helps the neural network understand what needs to be done and what task to focus on. The request must be correct and contain enough information for the neural network to give a relevant and comprehensive answer. The article discusses such language models as GigaChat, YandexGPT and ChatGPT 4. The responses of the considered language models are analyzed alternately to the same query. The general course of the entire algorithm for selecting a language model for use in the development of a system for automating the process of forming technical documentation is described.

Keywords: language models, prompt, technical documentation, technical writer, document management standards.

В современном мире в последнее время приобрели широкую известность языковые модели. По определению языковая модель – это алгоритм, который позволяет вычислить вероятность появления слова в предложении. Она учитывает минимальный контекст в словосочетании или предложении, стилистику высказывания и общий культурный фон. Языковые модели обучают на массивах текстов, созданных человеком, таких как классическая литература, статьи в СМИ и посты в социальных сетях [1, 2].

Основная цель моделей – создать разумное и «человеческое» продолжение текста на основе доступной информации. Существует множество типов языковых моделей, но наиболее распространенными являются:

1. N-граммные модели: предсказывают следующее слово в последовательности на основе предыдущих n слов.

2. Нейронные сети: используют искусственные нейронные сети для обработки текстовых данных и определения взаимосвязей между словами.

3. Трансформеры: новые архитектуры нейронных сетей, которые позволяют модели лучше понимать контекст и связи между словами.

Одним из наиболее перспективных подходов к написанию текстов является использование нейронных сетей и искусственного интеллекта. Нейронные сети обладают способностью обучаться на больших объемах данных и выявлять сложные закономерности, что делает их эффективными инструментами для создания технической документации. Также они обеспечивают упрощение работы, что делает их одним из наиболее перспективных подходов в данной области [3].

В данной статье будут рассмотрены следующие нейросети: YandexGPT, GigaChat и ChatGPT 4.

Существует необходимость выбрать такую нейронную сеть, чтобы была возможность получить максимально корректный ответ на отправленный запрос.

Для проведения анализа выбранных нейронных сетей было выполнено подключение к ним с использованием API-ключа. В табл. 1 продемонстрированы результаты полученных от нейронных сетей ответов.

Таблица 1

Результаты ответов нейронных сетей

YandexGPT	GigaChat	ChatGPT 4
Напиши раздел Технического задания "Полное наименование системы" для системы "Помощник технического писателя" по ГОСТу 34.602-2020. Система "Помощник техническому писателю" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных.		
Полное наименование системы: «Помощник технического писателя».	Полное наименование системы: "Автоматизированная система формирования технической документации «Помощник технического писателя»". (Принимаем)	Автоматизированная система формирования технической документации «Помощник технического писателя». (Принимаем)
Напиши раздел Технического задания "Сокращенное наименование системы" для системы "Помощник технического писателя" по ГОСТу 34.602-2020. Система "Помощник техническому писателю" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных.		
Сокращенное наименование системы: ПТППи	Сокращенное наименование системы: "АСТФ «ПТПП»". (Принимаем)	АРМ «Помощник ТП»
Напиши раздел Технического задания "Условное обозначение системы" для системы "Помощник технического писателя" по ГОСТу 34.602-2020. Система "Помощник техническому писателю" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных.		
Условное обозначение системы: ПТППи (Помощник техническому писателю)	Условное обозначение системы: "СПТПП"	Условное обозначение АРМ.ТП.01. (Принимаем)
Напиши раздел Технического задания "Перечень документов, на основании которых создается система" для системы "Помощник технического писателя" по ГОСТу 34.602-2020. Система "Помощник техническому писателю" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных.		
Раздел «Перечень документов» для системы «Помощник технического писателя» включает следующие документы: 1. Техническое задание (ТЗ) на разработку системы. 2. Техническое предложение (ТП) по разработке системы. 3. Эскизный проект (ЭП) системы. 4. Технический проект (ТП) системы. 5. Рабочая документация (РД) системы. 6. Программа и методика испытаний системы. 7. Инструкция по эксплуатации системы. 8. Документы, регламентирующие порядок обновления и развития системы. 9. Документы, подтверждающие соответствие системы требованиям стандартов и нормативных документов. (Принимаем)	Перечень документов, входящих в состав технического задания на систему "Помощник технического писателя": 1. Документ "Техническое задание на создание системы «Помощник технического писателя»". 2. Документ "Описание постановки задачи на создание системы «Помощник технического писателя»". 3. Документ "Концептуальный проект системы «Помощник технического писателя»". 4. Документ "Технический проект системы «Помощник технического писателя»". 5. Документ "Рабочий проект системы «Помощник технического писателя»". 6. Документ "Программа и методика испытаний... (полный текст см. в приложении 1)	Перечень документов: 1. Техническое задание (настоящий документ). 2. Техническая спецификация. 3. Проект системы. 4. Руководство пользователя. 5. Документация по API (при необходимости). 6. Документация по интеграции (при необходимости). 7. Тестовая документация. 8. Документация по установке и настройке. 9. Справочник по терминам
Напиши раздел Технического задания "Цели и задачи создания системы" для системы "Помощник технического писателя" по ГОСТу 34.602-2020. Система "Помощник техническому писателю" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных.		

Цели системы «Помощник технического писателя»: 1. Автоматизация процесса формирования технической документации. 2. Соблюдение стандартов документооборота 3. Использование нейросетей для автоматизации процесса создания документации. 4. Возможность загрузки документов и готовых проектов в систему. 5. Переключение между базами данных для работы с технической документацией. Задачи системы «Помощник технического писателя»: 1. Анализ целевой аудитории и создание документации на соответствующем уровне понимания. 2. Сотрудничество с экспертами и глубокое понимание продукта. 3. Структурирование материала и организация информации в логическом порядке. 4. Написание и редактирование точных и ясных руководств, справочных материалов и инструкций. 5. Использование специализированных инструментов для технического письма. 6. Визуализация информации с помощью... (полный текст см. в приложении 1)	Цели создания системы: 1. Автоматизация процесса формирования технической документации в соответствии со стандартами документооборота 2. Повышение эффективности работы технических писателей за счет использования нейросетей для генерации текстовых материалов. 3. Уменьшение времени, затрачиваемого на подготовку технической документации. Задачи создания системы: 1. Разработка алгоритмов и методов автоматического формирования технической документации на основе шаблонов и стандартов. 2. Создание базы знаний, содержащей информацию о различных типах оборудования, их характеристиках и особенностях эксплуатации. 3. Разработка интерфейса пользователя, позволяющего легко загружать документы в систему, выбирать нужные шаблоны и генерировать готовую документацию. 4. Обеспечение возможности загрузки готового проекта и переключения между базами данных для работы с различными. (Принимаем)	Цель создания системы - повысить эффективность и качество создания технической документации в организации. Основные задачи: 1. Автоматизация задач написания, форматирования, проверки и публикации технической документации. 2. Сокращение времени на создание документации. 3. Повышение уровня унификации и соответствия документации стандартам документооборота. 4. Обеспечение доступности информации для пользователей в удобном формате. 5. Снижение затрат на создание и поддержание технической документации. 6. Улучшение взаимодействия между техническими писателями и другими участниками проектов
Напиши раздел Технического задания "Сроки начала и конца по этапам" по ГОСТу 34.602-2020 для системы "Помощник технического писателя", если дата начала соответствует дате подписания договора (01.09.2024), а договор рассчитан на 9 месяцев. Система "Помощник техническому писателю" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных		
Сроки начала и конца этапов: 1. Подготовительный этап: с 1 сентября 2024 года по 30 ноября 2024 года. 2. Разработка и тестирование системы: с 1 декабря 2024 года по 31 марта 2025 года. 3. Внедрение и обучение пользователей: с 1 апреля 2025 года по 30 июня 2025 года. 4. Сопровождение и поддержка: с 1 июля 2025 года по 30 сентября 2025 года. (Принимаем)	Сроки начала и конца по этапам для системы "Помощник технического писателя" будут следующими: 1. Предпроектное обследование и анализ требований заказчика - с 01.09.2024 по 30.11.2024. 2. Разработка концепции и технического проекта системы - с 01.12.2024 по 28.02.2025. 3. Разработка рабочего проекта и программного кода - с 01.03.2025 по 31.05.2025. 4. Тестирование и отладка системы - с 01.06.2025 по 31.08.2025. 5. Ввод в опытную эксплуатацию 31.08.2022	Сроки начала и конца по этапам 1. Анализ требований и разработка технической спецификации: Создание документации с подробным описанием функциональных и нефункциональных требований к системе с 01.09.2024 по 01.10.2024 2. Проектирование системы: Разработка архитектуры системы, выбор технологий, определение структуры базы данных, создание прототипов интерфейсов с 01.10.2024 по 01.12.2024. 3. Разработка системы написание кода, реализация... (полный текст см. в приложении 1)
Напиши раздел Технического задания "Перечень подсистем для системы" для системы "Помощник технического писателя" по ГОСТу 34.602-2020. Система "Помощник техническому писателю" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных		

Раздел «Перечень подсистем для системы» 1. База данных технической документации: хранение и обработка технической документации в соответствии со стандартами документооборота 2. Модуль формирования документации: использование нейросетей для автоматического создания технической документации на основе загруженных документов и проектов. 3. Загрузка документов: импорт файлов с документами в систему для дальнейшей обработки и хранения... (продолжение см. в приложении 1) (Принимаем)	Перечень подсистем для системы "Помощник технического писателя": 1. Подсистема загрузки документов. 2. Подсистема генерации текста на основе нейронных сетей. 3. Подсистема управления базой данных проектов. 4. Подсистема переключения между базами данных. 5. Подсистема формирования технической документации. 6. Подсистема загрузки готового проекта	Перечень подсистем Система "Помощник технического писателя" состоит из следующих подсистем: 1. Подсистема редактирования и форматирования текста: Обеспечивает функции написания, редактирования, форматирования и проверки текста в соответствии с заданными стандартами. 2. Подсистема создания визуальных материалов: Позволяет создавать различные типы визуальных материалов... (продолжение см. в приложении 1)
Напиши раздел Технического задания "Требования к характеристикам взаимосвязей создаваемой системы" для системы "Помощник технического писателя" по ГОСТу 34.602-2020. Система "Помощник технического писателя" автоматизирует процесс формирования технической документации в соответствии со стандартами документооборота. Процесс автоматизируется посредством использования нейросетей. Также система имеет возможность загрузки документов в систему, загрузки готового проекта, переключения между базами данных.		
Требования к характеристикам взаимосвязей создаваемой системы «Помощник техническому писателю» должны соответствовать следующим критериям: 1. Интеграция с внешними системами: система должна иметь возможность интеграции с другими информационными системами и базами данных для обмена данными и выполнения совместных задач. 2. Взаимодействие с пользователем: система должна обеспечивать удобный интерфейс для пользователей, позволяющий им загружать документы, готовые проекты и переключаться между базами данных. 3. Нейросети: система должна использовать современные технологии и методы машинного обучения для автоматизации процесса формирования технической документации... (продолжение см. в приложении 1). (Принимаем)	Требования к характеристикам взаимосвязей создаваемой системы: 1. Система должна иметь возможность взаимодействия с другими информационными системами организации через стандартные протоколы обмена данными. 2. Система должна поддерживать обмен данными с системами документооборота, такими как электронная почта, файловые серверы и другие системы хранения информации. 3. Система должна предоставлять возможность интеграции с внешними базами данных и сервисами для получения дополнительной информации о проектах и документах. 4. Система должна обеспечивать безопасность передаваемых данных путем шифрования трафика и использования защищенных соединений	Требования к характеристикам взаимосвязей создаваемой системы: 1. Обработка документов: Система должна обеспечивать эффективную обработку технической документации, включая распознавание и анализ текста, структурирование информации и автоматическое формирование документов. Нейросети, применяемые в системе, должны быть обучены для высокоточной обработки технической информации и соответствовать стандартам обработки текста 2. Интеграция с внешними ресурсами: Система должна обеспечивать возможность загрузки документов из различных источников, таких как файлы, электронные письма и облачные хранилища. Необходимо обеспечить безопасную процедуру... (продолжение см. в приложении 1)

В ходе проведения анализа ответов нейронных сетей был сделан вывод о том, что нейронные сети YandexGPT и GigaChat дают наиболее корректные ответы на получаемые ими запросы. В табл. 2 продемонстрированы результаты анализа нейронных сетей [4].

Таблица 2

Результаты анализа нейронных сетей

Нейросеть	YandexGPT	GigaChat	ChatGPT 4
Принятые ответы	4	3	2

В результате нами была принята нейронная сеть YandexGPT, так как ее ответы больше всего соответствуют ГОСТ 34.602-2020 и подходят для задачи формирования технической документации [5].

Библиографический список

1. Акст Р. Языковые модели и мир будущего, или Путеводитель по ChatGPT. – Екатеринбург: Ridero, 2023. – 110 с.
2. Гринин И.Л. Генерация связного текста. разбор нейросетевых механик. механика первая – языковая модель как средство работы с языком // Инновации и инвестиции. – 2020. – №10. – С. 146–148.
3. Аспекты создания корпоративной вопросно-ответной системы с использованием генеративных предобученных языковых моделей / А.А. Голиков, Д.А. Акимов, М.С. Романовский, С.В. Тращенко // Litera. – 2023. – № 12. – С. 190–205. DOI: 10.25136/2409-8698.2023.12.69353
4. Как начать работать с YandexGPT API | Yandex Cloud - Документация//cloud.yandex.ru. – URL: <https://cloud.yandex.ru/docs/yandexgpt/quickstart>
5. ГОСТ 34.602-2020. Система стандартов по информации, библиотечному и издательскому делу. Техническое задание на создание автоматизированной системы / Федеральное агентство по техническому регулированию и метрологии, 2020, – URL: <https://www.standard.gov.ru/documents/34/602-2020>

Сведения об авторах

Курушин Данил Сергеевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: daniel@kurushin-perm.ru

Морозова Екатерина Максимовна – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: morozova.katerina.02@list.ru

Сахарова Алина Романовна – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: sakharlina@gmail.com

А.З. Тедеев, О.Л. Викентьева

Пермский национальный исследовательский
политехнический университет, Пермь

**ПРИМЕНЕНИЕ МАШИНЫ СОСТОЯНИЙ ДЛЯ РАЗРАБОТКИ
ПРИЛОЖЕНИЯ ДЛЯ ВЗАИМОДЕЙСТВИЯ С МОДЕЛЯМИ
ПЛАНИРОВАНИЯ ДЕЯТЕЛЬНОСТИ АГРОХОЛДИНГА**

Обобщены принципы использования машины состояний в разработке приложений. Приведены примеры использования механизма машины состояний на разных технологических платформах. Проведено сравнение использования состояний в Spring State Machine и PostgreSQL функциях.

Ключевые слова: машина состояний, Spring State Machine, функции PostgreSQL.

A.Z. Tedeev, O.L. Vikentieva

Perm National Research Polytechnic University, Perm

**USING A STATE MACHINE TO DEVELOP AN APPLICATION
TO INTERACT WITH AGROHOLDING PLANNING MODELS**

The principles of using a state machine in application development were summarised. Examples of using state machine mechanisms on different technological platforms were given. The comparison of using states in Spring StateMachine and PostgreSQL functions was carried out.

Keywords: State Machine, Spring State Machine, PostgreSQL functions.

В современном мире, где технологии играют все более важную роль, моделирование процессов и систем приобретает все большее значение. Модели позволяют нам лучше понять и предсказать поведение сложных систем, что имеет решающее значение для принятия обоснованных решений и оптимизации процессов.

Список моделей, используемых агрохолдингом для планирования деятельности:

– «Оптимизация норм внесения удобрений» (МУиММ). Бизнес-цель – автоматизация подбора оптимальных норм внесения минеральных удобрений на полях агрохолдинга с учетом исторических данных о ранее внесенных удобрениях и культурах-предшественниках;

– «Структура посевных площадей» (СПП). Бизнес-цель – определение оптимального долгосрочного портфеля производства с учетом рыночных рисков и потенциала урожайности;

– «Планирование сельскохозяйственных работ и управление парком техники» (ППР). Бизнес-цель – оптимизация долгосрочной структуры парка техники, инвестиций в обновление / расширение парка техники.

Перечисленные модели являются основой приложения по оптимизации деятельности агрохолдингов. Поскольку сотрудник агрохолдинга выполняет расчеты на множестве экземпляров моделей (сценариев), которые различаются по ограничениям и параметрам, то ярко выраженной проблемой является определение и разработка оптимального способа взаимодействия с моделями планирования длительности агрохолдинга. В табл. 1 перечислены возможные способы взаимодействия с их преимуществами и недостатками [1, 2].

Таблица 1

Сравнение способов взаимодействия с моделями

Модели	Преимущества	Недостатки
Машина состояний (конечный автомат)	Простота реализации. Легкость в понимании и отладке	Неэффективно для систем, где количество состояний велико или взаимосвязи между состояниями сложны
Model-View-Controller (MVC)	Разделение обязанностей между компонентами. Улучшение тестирования и поддержки кода. Удобство внесения изменений в отдельные части системы	Возможна сложность в управлении большими системами. Может потребоваться много ручной настройки для поддержания разделения
Model-View-Presenter (MVP)	Четкое разделение логики и представления. Удобство тестирования	Более сложная структура, чем в MVC

Каждый сценарий обладает статусом, которой характеризует текущее состояние модели. Статус определяет, какие действия можно выполнять над сценарием, а какие нельзя. На рисунке представлен граф, описывающий взаимосвязи между состояниями. В узлах графа приведены код и название статуса сценария.

В статусе «0» пользователь может вносить изменения во входные витрины, содержащие ограничения для модели сценария, а также задавать параметры расчета; может редактировать и удалять текущий сце-

нарий. В статусе «1» пользователь ограничен во всех упомянутых выше действиях, однако при необходимости может остановить расчет, что приведет сценарий в статус «4». Если не останавливать расчет, то по прошествии некоторого промежутка времени статус переходит в статус «2» или «3». В статусе «4» пользователь может либо удалить сценарий, либо сбросить незавершенный расчет (до статуса «0»), чтобы запустить расчет с другими параметрами. В статусе «3» пользователю доступен просмотр выходных витрин.

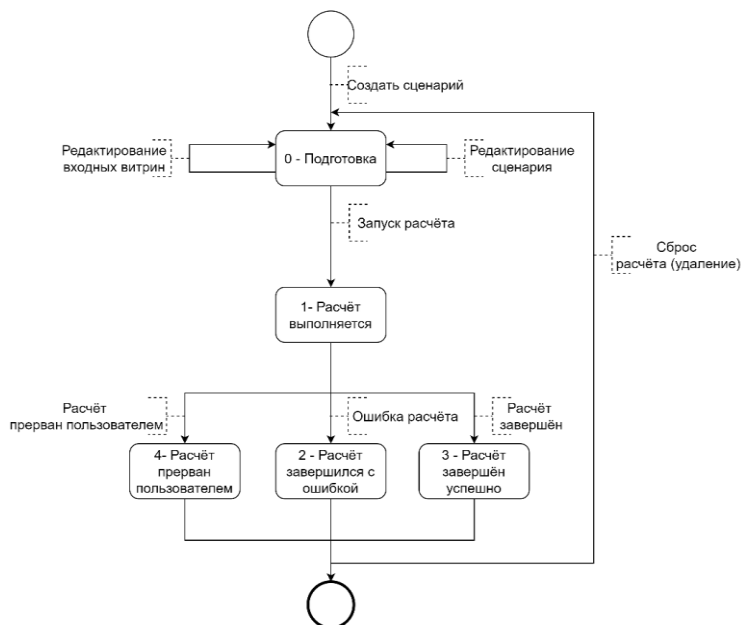


Рис. Состояния сценария

Выводы по табл. 1. Поскольку число состояний невелико (всего пять), а переход из одного состояния в другое достаточно четко определен, то единственным способом взаимодействия с моделями является именно машина состояний.

Для взаимодействия со сценариями пользователь использует веб-приложение. Платформой под приложение служит Spring Framework для Java. В рамках выбранной платформы существует компонент Spring State Machine, который можно использовать для представления рабочих процессов или любых других задач представления конечных автоматов

[3]. Опишем взаимосвязи состояний, проиллюстрированных на рисунке, с помощью конфигурации Spring State Machine (листинг 1).

Листинг 1 – Конфигурация машины состояний на Spring

```
@Configuration
@EnableStateMachine

public class ScenariosStateMachineConfiguration
extends      StateMachineConfigurerAdapter<String,
String> {

    @Override
        public      void      config-
ure(StateMachineStateConfigurer<String,      String>
states) throws Exception {
        states
            .withStates()
            .initial(ScStatus.PREPARE)

        .states(EnumSet.allOf(ScStatus.class));
    }

    @Override
        public      void      config-
ure(StateMachineTransitionConfigurer<ScStatus,
ScEvent> transitions) throws Exception {
        transitions

        .withExternal().source(ScStatus.PREPARE).target(ScS
tatus.RUNNING).event(ScEvent.START_RUN)
            .and()

        .withExternal().source(ScStatus.RUNNING).target(ScS
tatus.STOPPED_RUN).event(ScEvent.STOP_RUN)
            .and()

        .withExternal().source(ScStatus.RUNNING).target(ScS
tatus.ERROR_RUN).event(ScEvent.ON_RUN_ERORR)
            .and()

        .withExternal().source(ScStatus.RUNNING).target(ScS
tatus.SUCCESSFUL_RUN).event(ScEvent.ON_RUN_END)
```

```

        .and()

        .withExternal().source(ScStatus.STOPPED_RUN).target
        (ScStatus.PREPARE).event(ScEvent.ERASE_RUN)
        .and()

        .withExternal().source(ScStatus.ERROR_RUN).target(S
        cStatus.PREPARE).event(ScEvent.ERASE_RUN)
        .and()

        .withExternal().source(ScStatus.SUCCESSFUL_RUN).tar
        get(ScStatus.PREPARE).event(ScEvent.ERASE_RUN);
    }
}

```

Остается лишь описать поведение для перечислений ScEvent [4]. Оппонентом данному подходу можно привести построение машины состояний в самой базе данных, используя самый простой подход – ветвления. Проверку текущего состояния можно осуществлять благодаря функциям в PostgreSQL. Данный подход основан на исследовании в [5]. Пример в листинге 2.

Листинг 2 – Пример механизма стейт-машины в функции PostgreSQL

```

CREATE OR REPLACE FUNCTION
dev_schema."run_erase"(p_sc_id integer, p_user
character varying)
RETURNS void
LANGUAGE plpgsql
SECURITY DEFINER
SET search_path TO 'dev_schema', 'pg_temp'
AS $function$
declare
    in_tables record;
    recs record;
    l_run_id int;
    sql text;
    msg text;
    run_status int;
    cnt int;
begin

```

```

-- Получение статуса текущего сценария
select sc.last_run_id into l_run_id from
dev_schema.dt_scenarios sc where sc.id = p_sc_id;
select sc.status into run_status from
dev_schema.dt_scenarios sc where sc.id = p_sc_id;

if run_status < 2 then
raise exception 'По сценарию не был за-
вершен расчет. Действие запрещено.';
end if;

-- Поиск дочерних сценариев в состоянии расчета
...

select count(*) into cnt from sc_hier sh
where status = '1';

if cnt > 0 then
raise exception 'Один из дочерних сцена-
риев находится в состоянии расчета. Сброс расчета
не возможен.';
end if;

-- Удаление дочерних сценариев
...
-- Удаление расчета из таблиц
...
END;
$function$;

```

Данная функция проверяет, находится ли сценарий в одном из статусов «2», «3», «4». В таком случае осуществляется сброс текущего расчета, иначе функция возвращает исключение. Данная функция описывает переход из статуса «3» в статус «0» (см. рисунок).

В исследовании [5] была использована таблица с триггер-функциями. Однако этот подход является избыточным, так как на клиентской части можно настроить компоненты так, чтобы они обращались к функциям PostgreSQL. Сравним 1-й и 2-й подходы с помощью табл. 2.

Таблица 2

Сравнение машины состояний на Spring и PostgreSQL

Критерий	Spring State Machine	PostgreSQL Functions
Модульность	Привязана к платформе Spring Framework	Не зависит от конкретной платформы, поэтому можно внедрять в разные приложения с разными технологическими решениями
Производительность	Оптимизирован для использования внутри приложения и может использовать внутренние механизмы для быстрого перехода между состояниями. Однако, если логика состояний слишком сложная или требуется высокая производительность, может быть неэффективен	Оптимизированы для работы с большими объемами данных и обычно предоставляют лучшую производительность для операций чтения и записи. Однако, для реализации сложной логики состояний может потребоваться больше кода и оптимизации
Поддержка	Разработчик одного профиля способен поддерживать изменения	Требование разработчиков разного профиля для поддержки изменений
Распределенность	Является монолитом с клиентской частью	Можно разделить на разные машины
Безопасность	Позволяет реализовать двойную проверку: как на клиентской части, так и на серверной	Позволяет реализовать двойную проверку: как на клиентской части, так и в базе данных
Понятность	Высокая степень понятности, поскольку приближено к естественному языку	Менее понятно, поскольку менее приближено к естественному языку

Заключение. Несмотря на удобство в разработке (производительность, поддержка, понятность) инструмента Spring State Machine, использование механизмов машины состояний на уровне базы данных является более предпочтительным за счет гибкости взаимодействия с другими технологиями.

Библиографический список

1. Шамгунов Н.Н., Корнеев Г.А., Шалыто А.А. State Machine – новый паттерн объектно-ориентированного проектирования // Информационно-управляющие системы. – 2004. – №5. – С. 13–25.

2. Программный паттерн проектирования архитектурного каркаса Model-View-Controller при разработке веб-приложений систем мониторинга спецтранспорта аэропорта / Б.П. Елисеев, А.В. Тарасенко, О.А. Горбачев, Д. Лю // Научный вестник Московского государственного технического университета гражданской авиации. – 2015. – № 220. – С. 137–142.

3. Хендриксе 3. A Guide to the Spring State Machine Project [Электронный ресурс]. – URL: <https://www.baeldung.com/spring-state-machine> (дата обращения: 20.04.2024).

4. Коровин А. Spring State Machine [Электронный ресурс]. – URL: https://antkorwin.com/statemachine/statemachine_rus.html (дата обращения: 21.04.2024).

5. Рыбанов А.А., Свиридова О.В., Филиппова Е.М. Автоматный подход к реализации логики автоматизируемых бизнес-процессов в реляционных базах данных // Вестник Адыгейского государственного университета. Сер. 4: Естественно-математические и технические науки. – 2023. – № 326. – С. 40–46.

Сведения об авторах

Тедеев Александр Зурабович – бакалавр Пермского национально-го исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: alex.tedeev@yahoo.com

Викентьева Ольга Леонидовна – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: oleovic@rambler.ru

А.А. Ларионов, Ю.О. Лапшина

Пермский национальный исследовательский
политехнический университет, Пермь

КОМПЛЕКСНЫЙ ПОДХОД К ПРОВЕРКЕ ЗАГРУЖАЕМЫХ ФАЙЛОВ ИЗОБРАЖЕНИЙ

В статье рассматривается проблема обеспечения безопасности при загрузке файлов изображений в веб-приложениях. Освещается обзор различных методов проверки загружаемых файлов изображений. Подчеркивается важность реализации надежных механизмов проверки для предотвращения загрузки вредоносных или неподходящих файлов. Приводятся практические примеры кода, демонстрирующие работу методов проверки файлов для обеспечения безопасности загружаемого контента.

Ключевые слова: проверка файлов, безопасность веб-приложений, идентификация типа файла, «магические байты», Python, JavaScript.

A.A. Larionov, J.O. Lapshina

Perm National Research Polytechnic University, Perm

AN INTEGRATED APPROACH TO CHECKING DOWNLOADED IMAGE FILES

The article discusses the problem of ensuring security when loading image files in web applications. Provides an overview of various methods for checking uploaded image files. Emphasizes the importance of implementing robust verification mechanisms to prevent malicious or inappropriate files from being uploaded. Practical code examples are provided to demonstrate how file verification techniques work to ensure the security of downloaded content.

Keywords: file verification, web application security, file type identification, magic bytes, Python, JavaScript.

В современных веб-приложениях загрузка файлов, особенно изображений, является распространенной функцией, и вопрос безопасности загружаемых файлов становится все более актуальным.

Пользователи могут случайно или намеренно загрузить неподходящие или вредоносные файлы, что создаст серьезные риски для безопасности и производительности веб-приложений. Поэтому важно реализовать надежные механизмы проверки загружаемых файлов.

Отсутствие или недостаточная проверка загружаемых файлов может привести к различным угрозам безопасности, включая вирусы, вредоносный код, скрытые сценарии атак и другие формы злонамеренного содержимого [1]. Это может привести к серьезным последствиям, таким как:

- компрометация конфиденциальных данных,
- распространение вредоносного ПО,
- сбои и ошибки в работе веб-приложения,
- ухудшение репутации системы,
- потеря данных.

Существуют методы, которые используются для проверки загружаемых файлов изображений с целью минимизации упомянутых рисков. Рассмотрим некоторые из наиболее используемых методов [2]:

- определение типа файла по расширению,
- установка ограничения максимального размера загружаемого файла,
- использование регулярных выражений для проверки формата файла,
- определение формата файла методом «магических байтов»,
- использование встроенных модулей `file`, `magic`, `filetype` в Python.

Для максимальной безопасности загружаемых файлов изображений рекомендуется использовать комплексный подход, включающий проверку на стороне клиента и проверку на стороне сервера. Для более надежной проверки загружаемых файлов необходимо обращаться к их содержимому, а не только к имени файла. Это позволит предотвратить загрузку файлов с измененными именами, но с тем же вредоносным содержанием.

В листинге 1 представлен практический пример кода JavaScript, который выполняет проверку прикрепленных файлов на соответствие размера и типа перед их отправкой на сервер [3]. Он обеспечивает проверку загружаемых файлов на клиентской стороне для снижения нагрузки на сервер и быстрого ответа пользователю в случае неподходящего файла.

Листинг 1 – Клиентская проверка размеров и типов файлов

```
if (attachedFiles.length > 0) {  
    let maxFileSize = 5000000;  
    // Максимальный размер файла в байтах (5MB)  
  
    let data = new FormData();
```

```

        // Создание объекта FormData для отправки
        файлов на сервер

        let allowedFileTypes = ['image/jpeg', 'im-
age/png',
        'image/gif', 'image/jpg', 'image/bmp,];
        // Разрешенные типы файлов изображений

        // Цикл для перебора всех прикрепленных
        файлов
        for (let j = 0; j < attachedFiles.length;
        j++) {
            let file = attachedFiles[j];

            // Проверка размера и типа файла
            if (file.size <= maxFileSize &&
            allowedFileTypes.includes(file.type)) {
                data.append('photos', file);

            } else {
                if (file.size > maxFileSize) {
                    console.warn("[ ERROR ] File is
too large.");
                    $('#upload-photo-info').html('<p
class="text-
danger">Файл ' + file.name + '
слишком
        большой.</p>');
                }

                else {
                    console.warn("[ ERROR ] Invalid
file type.");
                    $('#upload-photo-info').html('<p
class="text-
danger">Файл ' + file.name + ' име-
ет
        недопустимый тип.</p>');
                }
            }
        }
    }
}

```

```

        return;
        // Прерывание выполнения, если файл
слишком большой
        или имеет недопустимый тип
    }
}
}

```

Допускается использование регулярных выражений для проверки формата файла изображения. Тогда к листингу 1 можно применить следующую проверку, представленную в листинге 2. Регулярное выражение проверяет, что имя файла заканчивается на .jpg, .jpeg, .png, .gif или .bmp без учета регистра.

Листинг 2 – Проверка типа файла с использованием регулярного выражения

```

let allowedFileTypes =
/\.(jpg|jpeg|png|gif|bmp)$/i;

```

Рассмотрим пример в листинге 3, использующий метод «магических байтов» для определения типа изображения в Python. Каждый формат изображения имеет уникальную последовательность байтов в начале файла, которая называется «магическим числом». Код проверяет первые несколько байтов файла и сравнивает их с известными «магическими числами» для точного распознавания типа изображения. Такой подход более надежен, чем проверка расширения файла, поскольку расширение может быть изменено или вовсе отсутствовать. Здесь определяется тип файла по его содержимому, а не по расширению, что позволяет избежать ошибок и обеспечить более точную идентификацию формата файла.

Листинг 3 – Метод «магических байтов» для определения типа загружаемого файла изображения

```

def get_image_type(file_path):
    # Открываем файл в бинарном режиме для чтения

    with open(file_path, 'rb') as file:
        # Читаем первые несколько байт
        magic_number = file.read(4)

    # Определяем тип файла по его «магическому

```

числу»

```
        if
mag-
ic_number.startswith(b'\xff\xd8\xff'):
    return 'JPEG'
    elif magic_number == b'\x89PNG\r\n\x1a\n':
    return 'PNG'
    elif magic_number.startswith(b'GIF8'):
    return 'GIF'
    elif magic_number.startswith(b'BM'):
    return 'BMP'
    elif magic_number.startswith(b'\xFF\xD8'):
    return 'JPG'
else:
    return 'Unknown file type'
```

```
# Пример использования
file_path = 'path/to/image.jpg'
image_type = get_image_type(file_path)
print(f'The image type is: {image_type}')
```

Продолжая тему определения типа файла по его содержимому в Python, можно рассмотреть использование встроенных модулей и библиотек.

Вариант 1. Использование встроенного модуля `file` [4], чтобы открыть файл в двоичном режиме и прочитать первые 4 байта. Затем сравниваем «магические байты» с известными шаблонами, чтобы определить тип файла. Если ни один из шаблонов не соответствует, возвращаем «Неизвестный тип файла» (листинг 4).

Листинг 4 – Использование модуля `file`

```
import file
def identify_file_type(file_path):
    with open(file_path, 'rb') as f:
        magic_bytes = f.read(4)
        if magic_bytes == b'\x89PNG':
            return 'PNG image'
        elif magic_bytes[:2] == b'BM':
            return 'BMP image'
        elif magic_bytes[:2] == b'GIF':
            return 'GIF image'
```

```

else:
    return 'Unknown file type'

```

Вариант 2. Использование библиотеки Python-magic [2, 4], которая может идентифицировать типы файлов на основе «магических байтов», расширений файлов и других свойств файлов. В этом решении создается экземпляр класса Magic из библиотеки Python-magic и используется метод id_filename для определения типа файла на основе «магических байтов». Библиотека берет на себя загрузку соответствующей «магической» базы данных и выполнение идентификации (листинг 5).

Листинг 5 – Использование библиотеки Python-magic

```

import magic

def identify_file_type(file_path):
    with magic.Magic() as m:
        file_type = m.id_filename(file_path)

    return file_type

```

Вариант 3. Использование библиотеки filetype [4], которую можно применять для идентификации типов файлов на основе магических байтов. Она обеспечивает простой и интуитивно понятный интерфейс для идентификации типа файла. Если тип файла не может быть определен, мы возвращаем «Неизвестный тип файла». В противном случае мы возвращаем MIME-тип файла [5] (листинг 6).

Листинг 6 – Использование библиотеки filetype

```

import filetype

def identify_file_type(file_path):
    file_type = filetype.guess(file_path)

    if file_type is None:
        return 'Unknown file type'
    else:
        return file_type.mime

```

Существуют и другие библиотеки для определения типа файла, выбор конкретного модуля или библиотеки зависит от требований, поддерживаемых форматов файлов и совместимости с операционной системой. Сравним эффективность различных методов проверки загружаемых файлов изображений. Для сравнения эффективности методов проверки загружаемых файлов использует следующий набор тестовых файлов:

- 1) корректное изображение «picture.jpg»,
- 2) файл текстового типа «text.docx»,
- 3) текстовый файл с расширением изображения «text2.jpg»,
- 4) файл изображения, размер которого превышает допустимый лимит: «large_picture.jpg»,
- 5) файл изображения, нарушающий шаблон «picture.docx.jpg»,
- 6) файл сценария формата .py с расширением изображения: «script_picture.gif».

Загрузим тестовые файлы и проверим, как рассмотренные методы реагирует на каждый из них. Отметим, какие файлы будут успешно или неуспешно загружены. Результаты представлены в таблице.

Сравнение методов проверки загружаемых файлов

Тестовый файл	По расширению	По размеру	Регулярные выражения	«Магические байты»	Модуль file	Python-magic	
picture.jpg	pass	pass	pass	pass	pass	pass	pass
text.docx	fail	pass	fail	fail	fail	fail	fail
text2.jpg	pass	pass	pass	fail	fail	fail	fail
large_picture.jpg	pass	fail	pass	pass	pass	pass	pass
picture.docx.jpg	pass	pass	fail	fail	fail	fail	fail
script_picture.gif	pass	pass	pass	fail	fail	fail	fail

Рассмотрим результаты проверки:

1) picture.jpg: все методы успешно идентифицировали правильное изображение.

2) text.docx: метод установки ограничения максимального размера загрузил текстовый файл.

3) text2.jpg: методы по расширению, размеру и регулярные выражения не распознали текстовый файл с расширением .jpg.

4) large_picture.jpg: метод по размеру правильно отклонил файл из-за превышения лимита.

5) picture.docx.jpg: методы по расширению и размеру не распознали файл изображения.

6) script_picture.gif: методы по расширению, размеру и регулярные выражения не распознали файл сценария с расширением .gif.

Для обеспечения надежной проверки загружаемых файлов изображений необходимо комбинировать несколько методов, таких как проверка по расширению, по размеру, «магические байты» и использование специализированных библиотек. Это позволит минимизировать вероятность прохождения некорректных или вредоносных файлов.

Таким образом, проверка загружаемых файлов является неотъемлемой частью обеспечения безопасности веб-приложений. Применение различных методов проверки, как на стороне клиента, так и на стороне сервера, позволяет минимизировать риски, связанные с загрузкой вредоносных файлов, утечкой конфиденциальных данных и сбоями в работе веб-приложений. Использование комплексного подхода к проверке загружаемых файлов позволяет значительно повысить уровень безопасности.

Библиографический список

1. Виды угроз информационной безопасности [Электронный ресурс] // Академия Selectel. – URL: <https://selectel.ru/blog/security-threats/> (дата обращения: 28.05.2024).
2. Определение типа файла без расширения [Электронный ресурс] // Справочник Python. – URL: <https://docs-python.ru/standart-library/modul-mimetypes-python/opredelit-tip-fajla-soderzhimomu/> (дата обращения: 28.05.2024).
3. Использование файлов в веб приложениях [Электронный ресурс] // Веб-технологии для разработчиков. – URL: https://developer.mozilla.org/ru/docs/Web/API/File_API/Using_files_from_web_applications (дата обращения: 28.05.2024).
4. How to use magic bytes to identify files using python [Электронный ресурс] // PythonSOS. – URL: https://www.pythonsos.com/file-handling/how-to-use-magic-bytes-to-identify-files-using-python/?expand_article=1 (дата обращения: 28.05.2024).
5. MIME-типы [Электронный ресурс] // htmlbook. – URL: <https://htmlbook.ru/html/value/mime> (дата обращения: 28.05.2024).

Сведения об авторах

Ларионов Александр Андреевич – инженер кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: alarionov93@yandex.ru

Лапшина Юлия Олеговна – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: lapshichna@yandex.ru

А.С. Субботин, Р.А. Файзрахманов

Пермский национальный исследовательский
политехнический университет, Пермь

СПОСОБЫ ОБНАРУЖЕНИЯ МОШЕННИЧЕСКИХ ДЕЙСТВИЙ В ECOM-ТРАНЗАКЦИЯХ

Исследуется возможность борьбы с мошенническими действиями в ECOM-транзакциях. Метод борьбы с фродом при помощи совокупности алгоритмов искусственного интеллекта. Он должен выявить достоинства и недостатки уже существующих методов. Описан алгоритм способа обнаружения нежелательных операций в ECOM-транзакциях.

Ключевые слова: антифрод системы, борьба с мошенничеством, банковские транзакции, фрод мониторинг.

A.S. Subbotin, R.A. Faizrahmanov

Perm National Research Polytechnic University, Perm

WAYS TO DETECT FRAUDULENT ACTIONS IN ECOM TRANSACTIONS

This article explores the possibilities of combating fraudulent activities in ECOM transactions. A method of combating fraud using artificial intelligence algorithms. It must demonstrate the advantages and disadvantages of existing methods. An algorithm for detecting unwanted operations in ECOM transactions is described.

Keywords: anti-fraud systems, anti-fraud, banking transactions, fraud monitoring.

В современном мире ECOM-транзакции занимают огромную часть всех банковских операций, осуществляемых в Интернете, в связи с этим мошенничество в этой сфере активно развивается. На рис. 1 изображен график процента фрод операций в ECOM-транзакциях.

Данные были взяты у компании ООО «Биллинговые системы» [1].

Ниже приведена статистика фрода в других банковских операциях.

1. Card present 0,1 %.
2. Person to Person 0,5 %.
3. Выплаты на карту 0,1 %.

В связи с этим можно сделать вывод, что ECOM-операция является самой популярной для мошенничества.

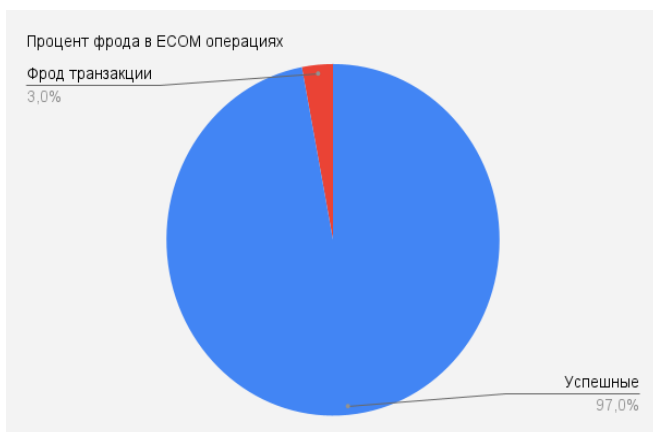


Рис. 1. Процент фрода в ЕСОМ операциях

Для решения этой проблемы на отечественном рынке можно увидеть следующие системы фрод-мониторинга:

1. Safer Payments от IBM – одно из самых популярных решений для осуществления фрод-мониторинга банковских переводов. В данном случае применяется концепция работы, которая основана на загружаемых правилах. В этих правилах установлены сформированные условия блокировок получаемого трафика. Как показала практика, данная система имеет хорошую позицию на рынке среди систем антифрода, хоть и считается одной из самых дорогих в плане обслуживания фрод-мониторинга.

2. TranzWare Fraud Analyzer от Compass Plus Technologies – данная система закрепились на рынке еще с 2003 г. и также пользуется популярностью в использовании. Устройство системы представляет собой Rule Based подход, но не смогла получить обширного распространения как, решение от IBM.

3. Smart Fraud Detection от компании Fuzzy Logic Lab – российская система антифрода, которая появилась на рынке в 2016 г. Данная система может работать в автоматическом режиме, при этом пользуясь алгоритмами механического обучения для противостояния мошенническим действиям [2].

К сожалению мошенничество не стоит на месте, а развивается. С каждым днем придумывается новый способ обхода такой системы, а так как они не содержат в себе многоэтапность проверок, обучение и перестроение модели, то с каждым годом процент выявления в данных систе-

мах все ниже и ниже. В связи с этим была выдвинута гипотеза о необходимости создать антифрод систему с тремя различными и независимыми модулями, которые строили бы внутри себя свои модели и обучались.

Систему предлагается разбить на 3 независимо работающих друг от друга модуля, которые в результате приведут к единственному выводу. Модули необходимо сделать параллельными, чтобы не удлинять время принятия решения, так как в банковской операции важна каждая миллисекунда. Модули будут содержать в себе 3 основных метода:

- 1) классификация;
- 2) кластеризация;
- 3) алгоритм дерева принятия решения [3].

Алгоритм работы системы изображен на рис. 2.

На заключительном этапе работы системы необходимо сделать вывод о итоговой вероятности фрода в данной операции. Чтобы определить итоговую вероятность принадлежности объекта к определенному классу или кластеру, необходимо использовать формулу, которая учитывает вероятности принадлежности объекта к каждому из классов или кластеров. Для метода классификации итоговая вероятность принадлежности объекта к определенному классу рассчитывается как сумма произведений вероятностей принадлежности объекта к каждому из классов и условных вероятностей появления признаков объекта при условии его принадлежности к каждому из классов. Эта формула может быть записана следующим образом:

$$P(C) = P(C|X_1) \cdot P(X_1) + P(C|X_2) \cdot P(X_2) + \dots + P(C|X_n) \cdot P(X_n),$$

где $P(C)$ – итоговая вероятность принадлежности объекта к классу C , $P(C|X_i)$ – вероятность принадлежности объекта к классу C при условии появления признака X_i , $P(X_i)$ – вероятность появления признака X_i .

Для метода кластеризации итоговая вероятность принадлежности объекта к определенному кластеру рассчитывается как сумма произведений вероятностей принадлежности объекта к каждому из кластеров и условных вероятностей появления признаков объекта при условии его принадлежности к каждому из кластеров. Эта формула может быть записана следующим образом:

$$P(K) = P(K|X_1) \cdot P(X_1) + P(K|X_2) \cdot P(X_2) + \dots + P(K|X_n) \cdot P(X_n),$$

где $P(K)$ – итоговая вероятность принадлежности объекта к кластеру K , $P(K|X_i)$ – вероятность принадлежности объекта к кластеру K при условии появления признака X_i , $P(X_i)$ – вероятность появления признака X_i .

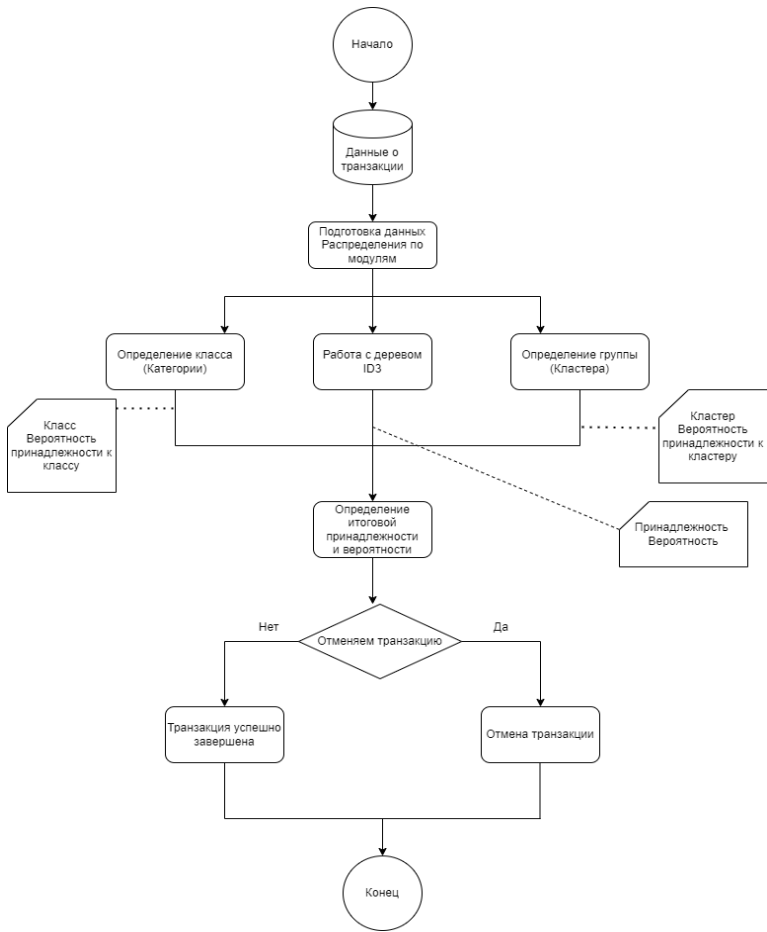


Рис. 2. Алгоритм работы фрод системы

В обоих случаях итоговая вероятность показывает, насколько точно объект относится к определенному виду (у нас их 2 – фрод или нефрод операция) [4].

Данный способ лучше и эффективнее существующих на сегодняшний день антифрод систем по ряду причин [5]:

1. Мультиплексирование данных. Метод с разными модулями позволяет использовать мультиплексирование данных для анализа разных аспектов транзакции, что позволяет увидеть более сложные сценарии мошенничества в транзакциях, в то время как метод с одним модулем ограничен в этом плане.

2. Комбинированный анализ. Метод с множеством модулей может позволить использовать комбинированный анализ данных для более точного результата, что повышает вероятность обнаружения мошенничества. В методе с одним модулем происходит простой анализ.

3. Адаптивность и гибкость. Метод с множеством модулей позволяет быть более эффективным и гибким в сравнении с одним, поскольку каждый модуль может быть настроен и обучен индивидуально, для обнаружения конкретного вида мошенничества и аномалий в транзакции. Это позволяет гораздо быстрее реагировать на изменения схемы мошенничества и повышает общую эффективность системы.

4. Расширенная функциональность. Метод с разными модулями может полностью анализировать и исследовать транзакцию, что помогает обнаружить самые сложные случаи мошенничества.

Таким образом, используя многомодульную систему для фрод мониторинга, мы получаем более широкий спектр возможностей для обнаружения и предотвращения мошенничества, что делает этот метод наиболее эффективным и быстрым в сравнении с другими, также мы исключаем возможность обхода данной системы из-за ее самообучаемости.

Библиографический список

1. Кобелев О.А. Электронная коммерция. – М.: Дашков и К, 2016.
2. Копнин А.А., Соколова Е.В. Методика обеспечения безопасности банковских интернет-транзакций на основе антифрод системы // International journal of professional science. – 2022.
3. Методы и модели анализа данных: OLAP и Data Mining / А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод. – СПб.: БХВ-Петербург, 2004.

4. Журавлев Ю.И., Рязанов В.В., Сенько О.В. Распознавание. Математические методы. Программная система. Практические применения. – М.: Фазис, 2006.

5. Митрофанова О.А. Классификационные методы в языкознании. – Новосибирск: НГПУ, 2013.

Сведения об авторах

Субботин Александр Сергеевич – аспирант кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: sashenkaasss@gmail.com.

Файзрахманов Рустам Абубакирович – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: itas@pstu.ru.

А.В. Ремянников, Д.В. Яруллин

Пермский национальный исследовательский
политехнический университет, Пермь

**МОДЕРНИЗАЦИЯ СУЩЕСТВУЮЩЕЙ ИНФОРМАЦИОННОЙ
СИСТЕМЫ РАСПРЕДЕЛЕНИЯ НАГРУЗКИ НА КАФЕДРЕ
«ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И
АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ» ПЕРМСКОГО
НАЦИОНАЛЬНОГО ИССЛЕДОВАТЕЛЬНОГО
ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА**

Рассматриваются процессы распределения нагрузки и формирования расписания Пермского национального исследовательского политехнического университета. На основе анализа документов, используемых в этих процессах, выявлен список параметров, которые играют ключевую роль в процессе распределения нагрузки и формирования расписания. Также в статье описана существующая автоматизированная система для распределения нагрузки, используемая на кафедре «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета. Полученный список параметров используется в web-модуле, дополняющем систему распределения нагрузки, для сбора пожеланий преподавателей, которые будут учитываться при распределении нагрузки и формировании расписания с целью повышения качества образовательного процесса.

Ключевые слова: распределение нагрузки, формирование расписания, преподаватель, сбор пожеланий, диалоговый агент.

A.V. Remyannikov, D.V. Yarullin

Perm National Research Polytechnic University, Perm

**MODERNIZATION OF THE EXISTING WORKLOAD
DISTRIBUTION INFORMATION SYSTEM AT THE DEPARTMENT
OF INFORMATION TECHNOLOGY AND AUTOMATED SYSTEMS
OF PERM NATIONAL RESEARCH POLYTECHNIC UNIVERSITY**

This article contains analysis of the existing IS for load distribution at the information technology and automated systems department, as well as discusses the processes of workload distribution and scheduling at Perm National Research Polytechnic University. Based on the analysis of the documents used in these processes, was developed a list of parameters that play a key role in the process of load distribution and scheduling. The article also describes the existing automated system for load distribution used at the information technology and automated systems Department of Perm National Research Polytechnic University. The resulting list of parameters is used in the web module of the workload distribution system to collect the preferences of teachers, which will be taken

into account when distributing the workload and forming a timetable in order to improve the quality of the educational process.

Keywords: workload distribution, timetable development, teacher, preference collection, dialogue agent.

В современном мире все больше коммерческих и государственных предприятий переходят на электронные средства хранения и передачи информации. В связи с этим возрастает потребность в создании решений для автоматизации обработки этой информации. В частности, проблема распределения нагрузки учителей и преподавателей учебных заведений существует довольно продолжительное время, и автоматизация данного процесса может значительно повысить эффективность распределения. Важной частью распределения нагрузки и составления расписания является опрос преподавательского состава для улучшения условий их работы. Процесс сбора пожеланий также можно оптимизировать с использованием автоматизации.

В настоящее время распределение нагрузки на кафедре «Информационные технологии и автоматизированные системы» производится с использованием информационной системы, разработанной в рамках ВКР на соискание академической степени магистра студентом В.В. Машановым.

Данная система не обладает функцией автоматизированного сбора требований к организации учебного процесса от преподавателей кафедры, в связи с чем данный процесс производится путем опроса каждого преподавателя с использованием электронной почты или в устной форме. Такой подход усложняет процесс распределения нагрузки и составления расписания, поскольку подразумевает перенос пожеланий преподавателей в базу данных системы вручную и больше подвержен ошибкам со стороны ответственных за перенос пожеланий в систему лиц [1–4].

Диаграмма Use-case AS IS, представленная на рис. 1, отражает возможности существующей системы распределения нагрузки.

Таким образом, существующую систему можно дополнить web-модулем, использующим диалогового агента для сбора пожеланий преподавателей, автоматизировав данный процесс. Исходя из анализа существующей технологии распределения нагрузки и учета пожеланий преподавателей, диалоговый агент должен предоставлять инструменты для сбора следующих пожеланий:

1. Для распределения нагрузки.
 - 1.1. Предпочитаемая периодичность занятий (первая, вторая недели).
2. Для составления расписания.
 - 2.1. Предпочитаемый корпус и аудитория.
 - 2.2 Предпочитаемые дни недели.
 - 2.3 Предпочитаемое время занятий.



Рис. 1. Диаграмма Use Case AS IS

Для описания возможностей будущего приложения была построена диаграмма Use-case TO BE для системы распределения нагрузки и модуля для сбора пожеланий преподавателей. Диаграмма представлена на рис. 2.



Рис. 2. Диаграмма Use Case TO BE

Для подробного описания процесса передачи информации о пожеланиях была построена Sequence диаграмма. Диаграмма приведена на рис. 3.

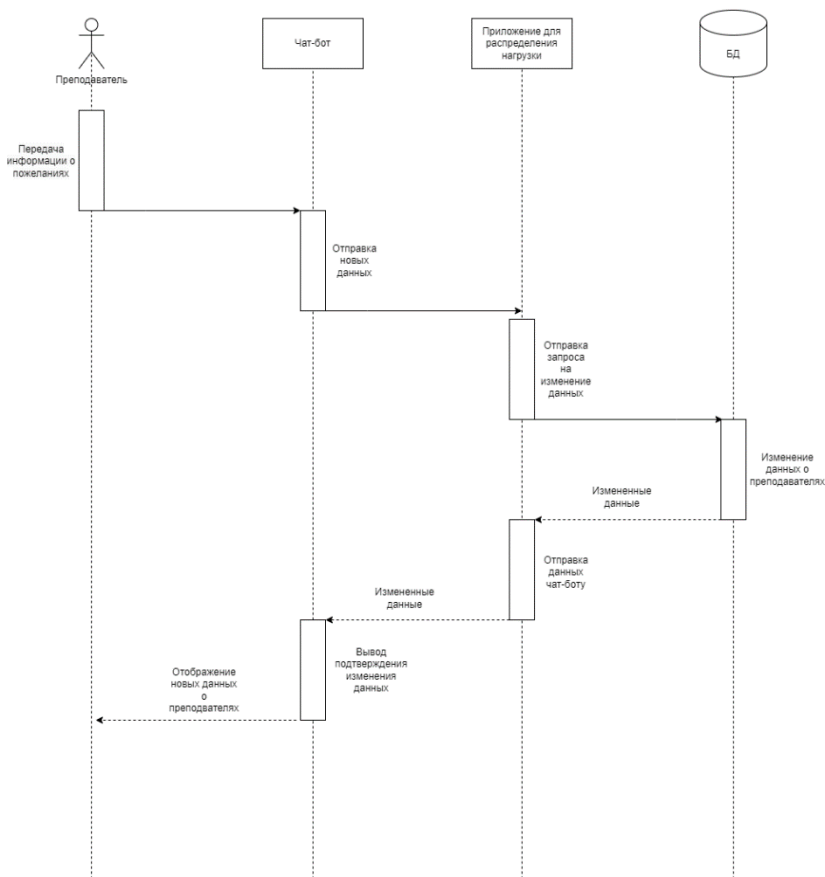


Рис. 3. Sequence диаграмма процесса передачи информации о пожеланиях

Таким образом, дополнение существующей системы модулем для сбора пожеланий позволит преподавателям передавать данные о своих требованиях к организации учебного процесса, влияя на процессы распределения нагрузки и разработки расписания, а также должно будет упростить процедуру принятия решений преподавателями за счет предоставления дополнительной информации об аудиториях и учебных группах.

Библиографический список

1. Машанов В.В., Файзрахманов Р.А.. Разработка информационной системы распределения учебной нагрузки преподавателей (на примере кафедры ИТАС ПНИПУ) // Автоматизированные системы управления и информационные технологии: материалы всерос. науч.-техн. конф.: в 2 т. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2023. – Т. 1. – С. 201–205.
2. C# Documentation. – URL: <https://learn.microsoft.com/ru-ru/dotnet/csharp/>
3. ГОСТ 34.602-2020 Техническое задание на создание автоматизированной системы. – М., 2020. – 8 с.
4. ГОСТ 7.32-2017. Отчет о научно-исследовательской работе. – М., 2017. – 27 с.

Сведения об авторах

Ремянников Александр Владимирович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-1б, г. Пермь, e-mail: 1remyannikovssh@gmail.com

Яруллин Денис Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: d.v.yarullin@ya.ru

А.В. Якубчик¹, Б.П. Погорелый², Р.А. Файзрахманов¹

¹Пермский национальный исследовательский
политехнический университет, Пермь

²Роботехпроект, Пермь

МОДЕЛИРОВАНИЕ ПОТОКОВ ЛЮДЕЙ В ОБЩЕСТВЕННЫХ МЕСТАХ

В этой статье исследуется возможность моделирования поведения большого количества людей, перемещающихся по общественным местам в штатных и нештатных ситуациях. Образуются так называемые «потoki людей», которые могут пересекаться. При неудачной конфигурации потоков могут возникать заторы, мешающие людям свободно проходить. Такая проблема может возникнуть как при обычном перемещении пешеходов (потоки людей в метро и т.д.), так и в случае эвакуации с места происшествия. В данной статье предлагаются способы моделирования потоков пешеходов, которые могут показать, что будет происходить в той или иной ситуации. Эту информацию можно использовать для проектирования более удачных и безопасных проходов и для нахождения оптимальных путей эвакуации.

Ключевые слова: моделирование потоков пешеходов, заторы, эвакуация, многоагентная модель.

A.V. Yakubchik¹, B.P. Pogorelyi², R.A. Faizrahmanov¹

¹Perm National Research Polytechnic University, Perm

²Robotechproject, Perm

MODELING PEDESTRIAN FLOWS IN PUBLIC SPACES

In this article we study the possibility of modeling a large number of pedestrians who move in public spaces both in normal situations and in emergencies. The pedestrian form "human flows", which may intersect. A bad configuration of human flows may cause a congestion, not letting people pass properly. This problem may arise both in normal pedestrian movement situations (human flows in subway etc.) and in case of evacuation from a disaster site. In this article we show pedestrian flow modeling methods which can show what will happen in a given situation. This information can be used for designing a better pedestrian way and for determining an optimal path for evacuation.

Keywords: pedestrian flow modeling, congestion, evacuation, multi-agent model.

В общественных местах образуются «потoki людей» – большие группы людей, идущих в одну сторону. Если проходы пересекаются, потоки сталкиваются; это может приводить к заторам. В случае

эвакуации из здания или с территории при нештатной ситуации происходит то же самое – люди проходят по имеющимся проходам, образуя потоки, эти потоки также могут сталкиваться, так как большие группы людей с разных мест проходят через одни и те же места и могут идти к одному выходу. Особенно необходимо учитывать эти явления при планировании путей эвакуации – например, проход может казаться широким, но из-за столкновения потоков люди могут беспорядочно толкаться в одном месте, и никто не сможет пройти.

В данной работе предлагается построить модель на основе модели социальных сил, впервые предложенной Мольнаром [1]. Модель представляет людей в виде агентов, которые движутся за счет «сил» – математических формул, показывающих их поведение, такое как стремление дойти до места назначения и обход препятствий и других людей. Введена сила «притягивания» к месту назначения и «отталкивания» от препятствий.

Для моделирования можно использовать два способа: микроскопический способ – многоагентная модель [2, 3] и макроскопический способ – непрерывная модель [4, 5], являющаяся приближением многоагентной. Многоагентная модель более наглядно показывает, что происходит с каждым конкретным человеком, но требует больших вычислительных ресурсов; кроме того, результаты оказываются непостоянными из-за использования случайных чисел для точной симуляции непредсказуемого поведения агентов. Непрерывная модель позволяет сократить вычислительную сложность и получить усредненную картину того, что будет происходить в той или иной обстановке.

Каждый человек отображается в виде точки с координатой x и скоростью v . Человек в потоке описывается формулами:

$$v = f(x), \quad (1)$$

$$x = [x \ y]^T \in R^2, \quad (2)$$

где $f(x)$ – следующая функция, показывающая поведение человека. Он идет в потоке, если не выбился из него, и старается вернуться в него, если выбился. Это показано на рис. 1.

$$f(x) = \begin{cases} v_0 \mathbf{d}, & (\|\mathbf{n}\| \leq w), \\ v_0 \mathbf{d} + k(\|\mathbf{n}\| - w) \frac{\mathbf{n}}{\|\mathbf{n}\|}, & (\|\mathbf{n}\| > w), \end{cases} \quad (3)$$

В формуле (3) $\mathbf{d}$ – единичный тангенциальный вектор потока, $\mathbf{n}$ – единичный нормальный вектор потока, w – ширина потока, v_0 – естественная скорость движения человека.

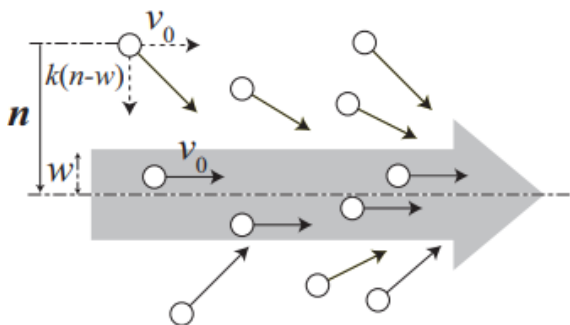


Рис. 1. Поведение людей в потоке

Для определения эффекта отталкивания вводится сигмоидная функция s , зависящая от расстояния r между людьми, или между человеком и препятствием; a , b , c – константы, которые определяют характер зависимости.

$$s(r) = \frac{c}{1 + \exp\{a(r - b)\}}. \quad (4)$$

Отталкивание людей от других людей и препятствий показывается формулами:

$$v_p = -s\left(\left\|r_p\right\|\right) \frac{r_p}{\left\|r_p\right\|}, \quad (5)$$

$$r_p = x_p - x, \quad (6)$$

где x_p – координата препятствия, x – координата человека, r_p – относительная координата препятствия, v_p – скорость отталкивания. К естественной скорости человека в потоке прибавляются все скорости отталкивания v_p , и получается фактическая скорость, с которой движется человек.

Моделирование таким способом позволяет воссоздать явления, которые фактически наблюдаются на перекрестках и при эвакуации. Так, при пересечении двух потоков происходит самоорганизация потока –

появляется узор из диагональных полос, в каждой из которых наблюдаются люди только из одного потока, и все проходят перекресток без проблем. В то же время, если люди на перекрестке идут с четырех сторон, то при определенном количестве людей образуется затор, и никто не может никуда пройти.

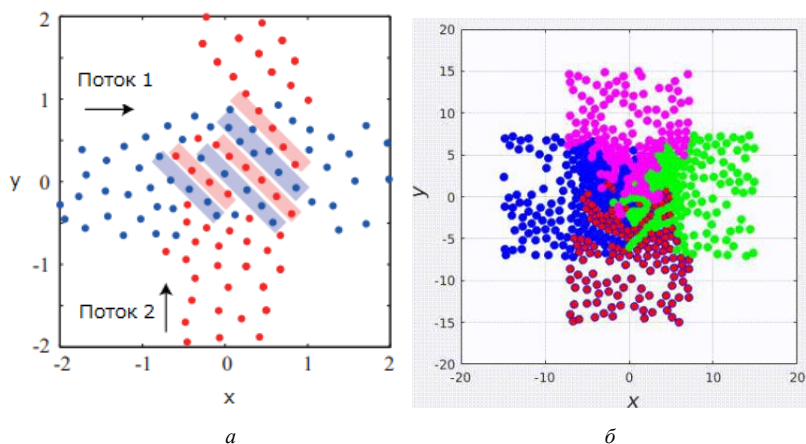


Рис. 2. Пересечение двух (а) и четырех (б) потоков на модели

Два потока самоорганизуются и проходят свободно, а четыре создают затор. Такой результат соответствует фактически наблюдаемым явлениям на перекрестках.

Компьютерное моделирование четырехстороннего потока людей показывает, что при определенной скорости и плотности потока движение людей через перекресток полностью прекращается.

Одним из решений проблемы предлагается размещение дополнительных препятствий (колонн) на пути потока, которые, разбивая поток на более тонкие группы, помогают организовать непересекающиеся цепочки. Моделирование показало, что скорость прохождения пешеходами перекрестка в этом случае увеличивается.

Для моделирования эвакуации предлагается использовать комбинированный метод – непрерывную модель для моделирования большого потока людей и многоагентную для определения скорости прохода по сложному пути эвакуации с препятствиями.

Этот метод позволяет оперативно оценивать безопасность того или иного места при разной конфигурации потоков, что полезно при проектировании перекрестков или определении оптимального плана эвакуации при пожарах и других катастрофах.

Библиографический список

1. Helbing D., Molnar P. Social force model for pedestrian dynamics // Physical Review E, APS. – 1995.
2. A Predictive Collision Avoidance Model for Pedestrian Simulation / I. Karamouzas [et al.] // Motion in Games. – 2009. – Vol. 5884. – P. 41–52.
3. PLEdestrians: A Least-Effort Approach to Crowd Simulation / S.J. Guy [et al.] // Proc. of Eurographics ACM SIGGRAPH Symposium on Computer Animation. – 2002.
4. Henderson L.F. The Statistics of Crowd Fluids // Nature. – 1971. – Vol. 229. – P. 381–383.
5. Hughes R.L. A Continuum Theory for the Flow of Pedestrians // Transportation Research Part B. – 2002. – Vol. 36. – P. 507–535.

Сведения об авторах

Якубчик Анна Владимировна – магистр технических наук, аспирант кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: anna@etp.su

Погорелый Борис Петрович – инженер ООО «Роботехпроект», г. Пермь, e-mail: boris_pogorely@mail.ru

Файзрахманов Рустам Абубакирович – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: fayzrakhmanov@gmail.com

В.О. Матяж, Д.В. Яруллин

Пермский национальный исследовательский
политехнический университет, Пермь

**РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ
АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ ТОПОЛОГИИ
СЕТИ НА ОСНОВЕ СЛАБОСТРУКТУРИРОВАННОГО ОПИСАНИЯ**

Рассматривается разработка информационной системы автоматизированного проектирования топологии сети на основе слабоструктурированного описания. Описанная система охватывает ключевые аспекты проектирования сетей, включая сбор и анализ исходных данных, преобразование слабоструктурированной информации в структурированную форму, автоматизированное проектирование топологии, подбор сетевого оборудования. Статья описывает алгоритмы проектирования сети, начиная с анализа исходных данных и заканчивая распределением сетевых узлов. Предложенная система облегчает задачи сетевых инженеров, снижает вероятность ошибок и ускоряет процесс проектирования.

Ключевые слова: топология, сеть, данные, слабоструктурированное описание, анализ, автоматизация.

V.O. Matyazh, D.V. Yarullin

Perm National Research Polytechnic University, Perm

**DEVELOPMENT OF AN INFORMATION SYSTEM
FOR COMPUTER-AIDED DESIGN OF NETWORK TOPOLOGY
BASED ON A WEAKLY STRUCTURED DESCRIPTION**

This article discusses the development of an information system for computer-aided design of network topology based on a weakly structured description. The described system covers key aspects of network design, including the collection and analysis of raw data, the transformation of poorly structured information into a structured form, and computer-aided topology design. The article describes network design algorithms, starting with the analysis of source data and ending with the distribution of network nodes. The proposed system facilitates the tasks of network engineers, reduces the likelihood of errors and speeds up the design process.

Keywords: topology, network, data, poorly structured description, analysis, automation.

С развитием современных технологий и увеличением сложности сетевых инфраструктур актуальность автоматизированных систем проектирования сетей возрастает. Традиционные методы проектирования

сетевых топологий часто требуют значительных временных и трудовых затрат. Они предполагают наличие четко структурированных данных и высокой квалификации специалистов. В реальных условиях исходные данные могут быть неполными или представленными в слабоструктурированном виде (например, текстовые описания, чертежи и пр.). Таким образом, возникает потребность в системе, способной обрабатывать и интерпретировать такие данные для автоматизированного проектирования сетевой топологии [1–3].

Современные технологии позволяют автоматизировать процесс проектирования топологии, что способствует повышению качества и снижению времени построения.

Подход к построению сетевых топологий. Структура построения сети описывается топологией. Под топологией сети понимается конфигурация графа, вершинам которого соответствуют конечные узлы сети (например, компьютеры) и коммуникационное оборудование (например, маршрутизаторы), а ребрам – связи между вершинами.

В данном разделе мы представим сравнительный анализ существующих подходов к построению сетевых топологий (таблица).

Основные виды топологий

Вид топологии	Описание	Преимущества	Недостатки
Шина	В шинной топологии все узлы сети подключаются к одной общей передающей среде (кабель)	Простота установки, минимальные затраты	Ограниченная длина и количество подключаемых устройств, надежность
Кольцо	В кольцевой топологии узлы соединены таким образом, что образуют замкнутый контур. Каждое устройство имеет два соединения – с предыдущим и следующим узлом	Равномерное распределение нагрузки	Разрыв соединения выводит сеть из строя
Звезда	В звездообразной топологии все узлы подключены к одному центральному устройству, например, коммутатору или маршрутизатору	Высокая отказоустойчивость, легкость управления	Зависимость от центрального устройства
Иерархическая звезда, дерево	Иерархическая звезда является расширением звездообразной топологии. В этой структуре несколько звездных сегментов соединены между собой через одно или несколько центральных устройств более высокого уровня	Масштабируемость, централизованное управление	Сложность настройки, зависимость от центральных устройств

Для построения корпоративных сетей больше подходит иерархическая топология/дерево. Основные элементы можно разделить на 3 уровня иерархической топологии: ядро сети, распределительный уровень и уровень доступа.

Разработка информационной системы. Первый этап разработки предполагает сбор и анализ слабоструктурированных данных. Это могут быть текстовые описания, таблицы, чертежи и другие форматы. Важной задачей является разработка алгоритмов и моделей, способных интерпретировать эти данные.

На втором этапе данные преобразуются в структурированную форму, пригодную для дальнейшего анализа. Здесь могут быть использованы методы обработки естественного языка (NLP) для текстовых данных.

На третьем этапе генерируется топология по шаблону иерархической модели.

На четвертом подбирается активное сетевое оборудование под заданную топологию. Принцип работы представлен на рис. 1.

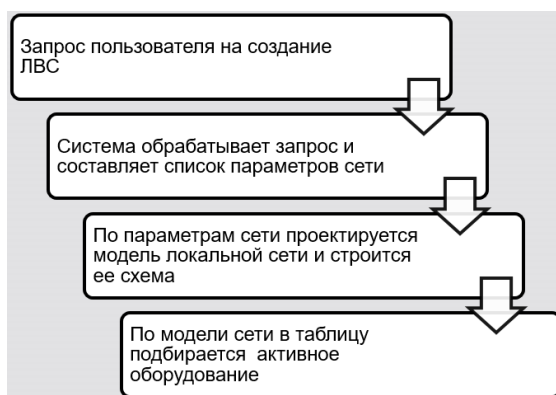


Рис. 1. Принцип работы системы

Слабоструктурированное описание поступает на вход, с помощью библиотек NLTK, pymorphy3 производится структуризация данных, а с помощью N2G и редактора диаграмм draw.io строится топология сети. С помощью библиотек Request, BeautifulSoup производится парсинг и подбор оборудования с интернет-магазинов. Стек используемых технологий представлен на рис. 2.



Рис. 2. Стек используемых технологий

Реализован алгоритм обработки слабоструктурированного описания, который представлен на рис. 3.

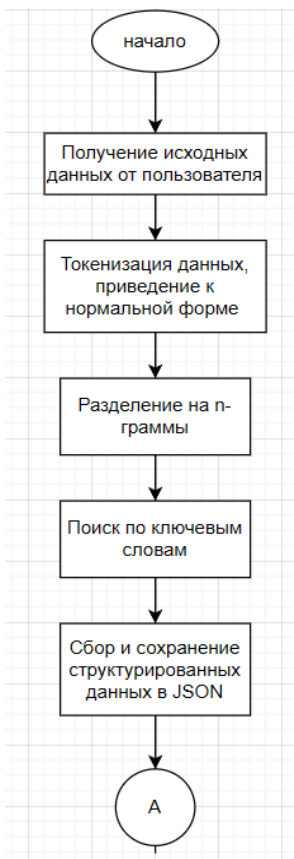


Рис. 3. Алгоритм анализа исходных данных

Результат работы алгоритма 3 представлен на рис. 4.

```

1  {"data":
2    {
3      "floors":
4        {
5          "1": "28", "2": "31", "3": "45"
6        },
7      "req":
8        {
9          "скорость": "1000", "стоимость": "500000"
10       },
11     "net_inf":
12       {
13         "интернет": "+"
14       }
15     }
16  }

```

Рис. 4. JSON-файл с параметрами сети

На основе структурированных данных выполняется автоматизированное проектирование топологии сети. Алгоритм представлен на рис. 5.

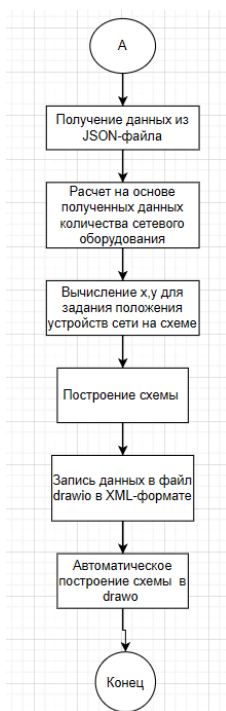


Рис. 5. Алгоритм формирования схемы

Результат работы представлен на рис. 6, 7, рекомендованная топология и оборудование.

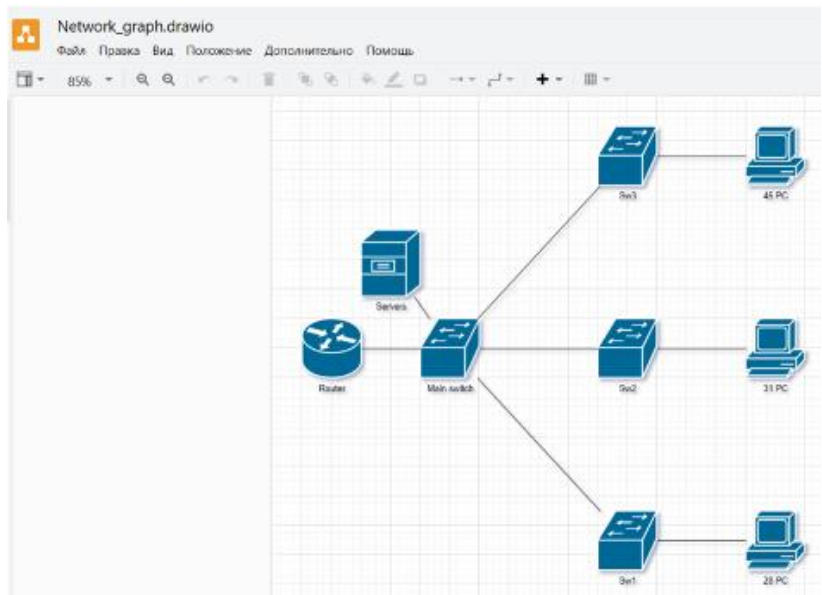


Рис. 6. Рекомендованная топология сети

	A	B	C	D	E	F	G	H	I
1	Название	Цена	Ссылка						
2	Коммутатор ZYXEL GS1900-24E	19 000pyб	https://www.kns.ru/product/kommutator-tp-link-tl-sg105/						
3	Коммутатор MikroTik CRS326-24G-2S+RM	26 161pyб	https://www.kns.ru/product/kommutator-tp-link-tl-sf1005d/						
4	Коммутатор D-Link DGS-1250-52X/A1A	40 450pyб	https://www.kns.ru/product/kommutator-tp-link-ls1008g/						
5	Коммутатор TP-Link TL-SG1016PE	14 246pyб	https://www.kns.ru/product/kommutator-zyxel-gs1900-10hp/						
6	Коммутатор TP-Link TL-SG108	2 162pyб.	https://www.kns.ru/product/kommutator-tp-link-tl-sg1024de/						
7	Коммутатор TP-Link TL-SG1005P	3 646pyб.	https://www.kns.ru/product/kommutator-tp-link-tl-sf1008d/						
8	Коммутатор TP-Link TL-SG1008D	1 611pyб.	https://www.kns.ru/product/kommutator-tp-link-tl-sg108pe/						
9	Коммутатор MikroTik CS5326-24G-2S+RM	22 514pyб	https://www.kns.ru/product/kommutator-zyxel-gs1900-24e/						
10	Коммутатор MikroTik CRS112-8P-4S-1N	19 928pyб	https://www.kns.ru/product/kommutator-mikrotik-crs326-24g-2s-plus-rm/						
11	Коммутатор TP-Link TL-SG1016D	4 791pyб.	https://www.kns.ru/product/kommutator-d-link-dgs-1250-52x-a1a/						

Рис. 7. Рекомендованное оборудование

Закключение. Разработанная информационная система демонстрирует значительный потенциал в области автоматизированного проектирования топологии сетей. Внедрение таких систем позволит существенно улучшить процессы проектирования, сократить затраты и повысить надежность сетевых решений. Дальнейшие исследования

и разработки в этой области будут направлены на улучшение алгоритмов обработки данных и расширение функциональных возможностей системы.

Библиографический список

1. Дорофеев А.А., Иванов В.В. Методы и модели проектирования сетевых топологий. – М.: Наука, 2020. – 350 с.
2. Попов П.П., Сидоров И.И. Автоматизация проектирования сетей: современные подходы и технологии. – СПб.: Питер, 2021. – 420 с.
3. Smith J., Brown R. Network Design Automation: Tools and Techniques. – New York: Wiley, 2019. – 300 p.

Сведения об авторах

Матяж Владимир Олегович – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: matiazhv@yandex.ru

Яруллин Денис Владимирович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: d.v.yarullin@ya.ru

Д.С. Курушин, Н.Н. Назаров, Л.В. Коногорова

Пермский национальный исследовательский
политехнический университет, Пермь

СОПРЯЖЕНИЕ СИСТЕМЫ МАШИННОГО ЗРЕНИЯ С СИТЕМОЙ УПРАВЛЕНИЯ РОБОТА-МАНИПУЛЯТОРА

В статье рассматриваются принципы взаимодействия системы машинного зрения с системой управления робототехнического комплекса. Описана технология адаптивного управления роботом-манипулятором по результатам анализа рабочей зоны. Изложен алгоритм настройки системы взаимодействия робототехнического комплекса и системы машинного зрения, а также процесс калибровки.

Ключевые слова: система машинного зрения, робототехнический комплекс, адаптивный, рабочее пространство, система управления.

D.S. Kuruschin, N.N. Nazarov, L.V. Konogorova

Perm National Research Polytechnic University, Perm

COUPLING OF MACHINE VISION SYSTEM WITH CONTROL SYSTEM OF A TRAINING ROBOT MANIPULATOR

The article discusses the principles of interaction between the machine vision system and the control system of a robotic complex. The technology of adaptive control of a robot manipulator based on the results of the analysis of the working area is described. The algorithm for configuring the interaction system of the robotic complex and the machine vision system, as well as the calibration process, is described.

Keywords: machine vision system, robotic complex, adaptive, workspace, control system.

Система машинного зрения (СМЗ) – это технология получения, обработки и использования изображений для решения различных прикладных задач без участия человека. Машинное зрение является ключевым компонентом гибких производственных систем (ГПС), включающей робототехнические системы, обладающие адаптивными свойствами и элементами нейросетевых технологий.

Использование технологии машинного зрения в учебном процессе является ключевым аспектом современного образования в области управления и программирования шестизвенным роботом-манипулятором. Оно позволяет настраивать взаимодействие манипулятора с реальными объектами, используя датчики и обратную связь. Получе-

ние необходимых компетенций в области интеграции СМЗ в робототехнический комплекс/систему позволит будущим специалистам быстро решать производственные задачи автоматизации [1–2].

По уровню относительной сложности задачи применения СМЗ в робототехнике условно делятся на две категории: простые и сложные. К простым относится определение наличия объекта в рабочей зоне, измерение расстояния до субъекта, измерение параметров объекта (геометрических), определение физических характеристик предмета и пр. К более сложным задачам относятся принципы взаимодействия СУ робота и СМЗ, например, снабжение информацией, необходимой для захвата объекта, обзор рабочей сцены для поиска представляющего интерес объекта, определение местоположения и ориентацию объекта и пр.

Режимы совместной работы СМЗ с СУ робототехнического комплекса зависят от получаемых данных и приоритета выполняемого действия. Один из наиболее популярных случаев взаимодействия состоит в том, что на основе получаемой и обрабатываемой информации, полученной с помощью СМЗ, подать роботу команду на выполнение жесткой программы действий, хранящейся в контроллере, или переход на другую программу путем передачи сигнала СМЗ в СУ робота по соответствующему каналу связи. В свою очередь, для синхронизации работы СМЗ с работой робота его управляющее устройство может посылать СМЗ запросы о состоянии окружающей среды, а также сигнал, подтверждающий прием и выполнение полученной команды. Возможна асинхронная работа системы с взаимным обменом запросами.

Функция СМЗ ограничена воздействием на заранее заданные программы, без адаптации. При адаптивном управлении роботом корректируются параметры управляющей программы в зависимости от информации о свойствах объектов манипулирования, вследствие чего строится траектория движения робота, который заранее обучен не был. При сопряжении СУ робота с СМЗ в целях обеспечения адаптивного режима работы необходимо организовать передачу количественной информации об объекте (размеры, характеристики формы и т.д.) и нужных сведений о модели проблемной среды. Для облегчения зрительного восприятия сцены робот может передавать вспомогательную информацию о рабочей среде с помощью специальных датчиков, например, тактильных, силомерных и локационных.

При включении СМЗ в состав адаптивного робототехнического комплекса необходимо согласовать системы координат СМЗ, робота и рабочего пространства. На этапе автономной настройки СМЗ выполняется калибровка масштаба изображения, которая определяет коэффициенты преобразования размеров объекта в картинной плоскости изображения в соответствующие линейные размеры этого объекта в рабочей зоне. Для этого используются эталонные объекты с известными размерами, в соответствии с которыми вычисляется количество элементов изображения по осям X и Y . Обычно стремятся добиться изотропного усиления, т.е. одинаковых масштабных коэффициентов по вертикали и горизонтали. С помощью эталонных объектов можно выявить геометрические искажения изображения, несовершенства оптики и т.д. Если искажения критичны, то их необходимо компенсировать программно.

После этапа настройки системы следует процесс калибровки, заключающийся в «привязке» друг к другу координатных систем (КС) робота и СМЗ для возможности преобразовывать движение манипулятора в рабочем пространстве исходя из количественных данных о местоположении и ориентации объекта, поступающих от СМЗ. Процесс «привязки» заключается в поиске матрицы однородных координат, задающей переход из $OkX_kY_kZ_k$ (системы координат камеры) в $OpX_pY_pZ_p$ (систему координат основания робота).

Однако описанная методика привязки КС камеры и робота имеет недостатки, связанные с затратами времени и дополнительными вычислениями, а также с погрешностью итоговых преобразований. На практике мы используем методику непосредственной калибровки СМЗ совместно с роботом без промежуточных преобразований в абсолютную систему координат. Она заключается в размещении эталонного объекта в поле зрения камеры, где СМЗ определяет калибровочную точку (КТ). В качестве эталонного объекта используется круг с отверстием в центре, так как погрешность определения координат центра кругового изображения обратно пропорциональна его радиусу. На рабочем органе манипулятора закрепляется указатель, острие которого точно выводится в КТ. По показаниям позиционных датчиков робота рассчитывается положение калибровочной точки в системе координат, связанной с роботом. Первое соотношение между положениями точки в картинной плоскости СМЗ и в СК робота получается путем решения прямой задачи кинематики.

Реализация системы машинного зрения в учебных робототехнических комплексах осуществляется с помощью библиотеки OpenCV с открытым исходным кодом на основе машинного обучения. Raspberry pi является управляющим устройством СУ манипулятора. Модуль камеры подключается через FFC-шлейф. Для получения и обработки изображений с камеры Raspberry pi используется модуль Picamera. Управление процессом манипулирование объекта в рабочей области робота происходит за счет внешней калибровки СМЗ посредством одного изображения (используется калибровочная мишень).

Библиографический список

1. Информационные устройства робототехнических систем: учебн. пособие. – М: Изд-во МГТУ им Н.Э. Баумана, 2005. – 384 с.
2. Техническое зрение роботов / В.И. Мошкин, А.А. Петров, В.С. Титов, Ю.Г. Якушенков; под общ. ред. Ю.Г. Якушенкова. – М.: Машиностроение, 1990. – 272 с.

Сведения об авторах

Курушин Даниил Сергеевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: daniel@kurushin-perm.ru

Назаров Никита Николаевич – бакалавр Пермского национального исследовательского политехнического университета, гр. ПРТ-20-16, г. Пермь, e-mail: nazarovnikitan@yandex.ru

Коногорова Луиза Викторовна – аспирант кафедры «Инновационные технологии машиностроения» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: konogorova98@mail.ru

А.Д. Черных, О.И. Мухин

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА МЕХАНИЗМА МУЛЬТИФИЛЬТРАЦИИ ТАБЛИЦ

В данной статье рассматривается разработка механизма фильтрации таблиц. Описаны сущности и функции, применяемые для фильтрации данных по одному или более атрибутам (столбцам). Представлены примеры реализации фильтров, функции добавления, удаления и применения фильтров, а также механизм подписки на изменения фильтров.

Ключевые слова: фильтрация, атрибут, сущность, коллекции, алгоритм, БД.

A.D. Chernykh, O.I. Mukhin

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A MULTIFILTER MECHANISM FOR TABLES

This article discusses the development of a table filtering mechanism. It describes the entities and functions used for filtering data by one or more attributes (columns). Examples of filter implementation, functions for adding, deleting, and applying filters, as well as a subscription mechanism for filter changes, are presented.

Keywords: filtering, attribute, entity, collections, algorithm, DB.

Исследования различных подходов к фильтрации и организации данных в таблицах уже проводились ранее, однако для достижения более высокой эффективности требуется многофункциональный механизм мультифильтрации с возможностью настройки фильтра [1–3].

Для таблицы «Показатели» реализуем механизм фильтрации по одному или более полям. Допустим, таблица «Показатели» содержит следующие поля: «id показателя», «Полное наименование», «Короткое наименование», «Статус данных» и дополнительные параметры, зависящие от атрибута «Статус данных». Также допустим, что в таблице «Показатели» есть дубликаты, которые различаются по атрибуту «Статус данных».

Фильтрация реализована с помощью следующих сущностей:

– private Dictionary<string, Func<Keyfigure, bool>> filters – список фильтров, которые применены к отображаемой коллекции;

- public ObservableCollection<Keyfigure> keyfigureModels – отображаемая коллекция (динамическая коллекция);
- public ObservableCollection<Keyfigure> staticKeyfigureModels – статическая коллекция. Необходима для избегания выгрузки коллекции из БД при каждом изменении фильтра;
- public void AddFilter(string columnName, Func<Keyfigure, bool> filter, bool dataStatusStat) – функция добавления фильтра в список фильтров;
- public Unit DellFilter(string columnName) – функция удаления фильтра из список фильтров;
- public ObservableCollection<Keyfigure> ApplyFilters() – функция применения фильтров к отображаемой коллекции.

При инициализации окна данные выгружаются из БД в статическую коллекцию. Если данные были изменены во время работы с приложением, то статическая коллекция заново выгружается из БД.

При применении фильтра в динамическую коллекцию загружаются данные из статической коллекции, к которой применяются все фильтры из списка фильтров filters включая новый.

Фильтры поделены на два типа по вводу параметра фильтрации:

- параметр фильтрации вводится с клавиатуры;
- параметр фильтрации выбирается из выпадающего списка.

На рис. 1 приведен листинг примера применения фильтра, параметр которого вводится с клавиатуры.

```
//Функция, вызываемая при изменении строки фильтра "Полное наименование"
private void FullNameStrChanged(object? sender, TextChangedEventArgs e)
{
    var textbox = (TextBox)sender; //Получение значения фильтра
    if (textbox.Text != null) //Если значение фильтра не равно нулю
    {
        string text = textbox.Text.ToLower(); //Приведение значения к нижнему регистру

        //Вызов функции добавления фильтра с параметрами:
        //Название фильтра - "FullName"
        //Фильтр - Contains(text)
        ((KeyfigureCVM)DataContext).AddFilter("FullName",
            item => item.FullName.ToLower().Contains(text), Box3.IsChecked.Value);
    }
    else //Если значение фильтра равно нулю
    {
        ((KeyfigureCVM)DataContext).DellFilter("FullName"); //Удаление фильтра по
        //полному наименованию
    }
    ((KeyfigureCVM)DataContext).RefreshDisplayedData(); //Вызов функции обновления
    //отображаемых данных
}
```

Рис. 1. Листинг функции FullNameStrChanged

На рис. 2 приведен пример применения фильтра, параметр которого выбирается в выпадающем списке.

```
//Функция, вызываемая при изменении выбранного элемента в выпадающем списке "Статус данных"
private void DataStatus_OnSelectionChanged(object? sender, SelectionChangedEventArgs e)
{
    //Получение данных выбранного элемента выпадающего списка
    var temp = (ComboBox)sender;
    //Вызов функции DataStatusChanged (добавление фильтра, обновления отображаемой коллекции)
    ((KeyfigureCVM)DataContext).DataStatusChanged
    ((DataStatus)temp.SelectedItem, Box3.IsChecked.Value);
}
```

Рис. 2. Листинг функции DataStatus_OnSelectionChanged

После добавления или удаления фильтров с помощью функций AddFilter или DellFilter соответственно вызывается функция RefreshDisplayedData, которая изображена на рис. 3.

```
//Функция обновляющая отображаемые данные
public void RefreshDisplayedData()
{
    keyfigureModels = ApplyFilters(); //Применение фильтров к коллекции
    SwitchCollectionWithDataStatus(dataStatusPos); //Редактирование коллекции
                                                //в завис. от статуса данных
}
```

Рис. 3. Листинг функции RefreshDisplayedData

Функция SwitchCollectionWithDataStatus реализует отображение показателей по статусу данных в зависимости от отображения столбца «Статус данных». Так как в таблице «Показатели» есть дубликаты, которые различаются по атрибуту «Статус данных», если столбец «Статус данных» не отображается, не отображаются и дубликаты.

На рис. 4 продемонстрирована функция ApplyFilters для таблицы «Показатели», возвращаемая коллекция которой присваивается в отображаемую коллекцию. Внутри функции к коллекции поочередно применяются фильтры из списка всех фильтров.

```
//Функция, применяющая фильтры к отображаемой коллекции
public ObservableCollection<Keyfigure> ApplyFilters()
{
    RefreshDB(); //Обновление БД
    ObservableCollection<Keyfigure> filteredData = keyfigureModels; //Получение текущей
                                                                    //коллекции
    foreach (var filter in filters.Values) //Проход по всем элементам коллекции фильтров
    {
        //Применение всех фильтров к коллекции
        filteredData = new ObservableCollection<Keyfigure>(filteredData.Where(filter));
    }
    return filteredData; //Возврат отфильтрованной коллекции
}
```

Рис. 4. Листинг функции ApplyFilters

На рис. 5 представлена функция AddFilter. Входные параметры функции:

- string columnName – название колонки, к которой применяется фильтр;
- Func<Keyfigure, bool> filter – фильтр.

```
//Функция, добавляющая фильтр в список применяемых фильтров
public void AddFilter(string columnName, Func<Keyfigure, bool> filter)
{
    //Присвоение элементу списка с ключом "columnName" значения "filter"
    filters[columnName] = filter;
}
```

Рис. 5. Листинг функции AddFilter

На рис. 6 представлена блок-схема функции DellFilter. Входной параметр функции – columnName.

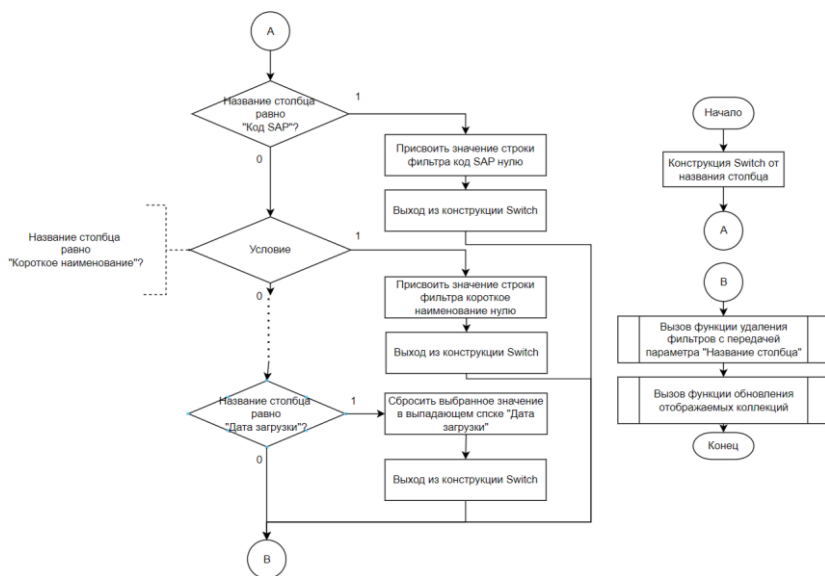


Рис. 6. Блок-схема алгоритма функции DellFilter

Таким образом, главным концептом мультифильтрации является «подписка на фильтры». В контроллере модели данных хранится словарь фильтров, который применяется к коллекции при каждом изменении любого из фильтров.

Библиографический список

1. Avalonia UI – Avalonia Docs [Электронный ресурс]. – URL: <https://docs.avaloniaui.net/ru/> (дата обращения: 10.05.2024).
2. Avalonia Reactive UI – Reactive UI Docs [Электронный ресурс] – URL: <https://docs.avaloniaui.net/docs/concepts/reactiveui/> (дата обращения: 12.05.2024).
3. MVVM – Модель представления Model-View-ViewModel [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/dotnet/architecture/maui/mvvm> (дата обращения: 12.05.2024).

Сведения об авторах

Мухин Олег Игоревич – доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: moi@stratum.ac.ru

Черных Артем Дмитриевич – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: timcheerson@gmail.com

Г.Ф. Масич, Д.А. Арышев

Пермский национальный исследовательский
политехнический университет, Пермь

АНАЛИЗ ОБЩЕИЗВЕСТНЫХ ОБЯЗАТЕЛЬНЫХ АТРИБУТОВ ПРОТОКОЛА BGP-4 ДЛЯ УПРАВЛЕНИЯ ВНЕШНЕЙ ПОЛИТИКОЙ МАРШРУТИЗАЦИИ

В статье приведены атрибуты протокола BGP-4, используемые для управления маршрутами между автономными системами. Проанализированы особенности использования общеизвестных обязательных атрибутов AS_PATH, NEXT_HOP, ORIGIN.

Ключевые слова: политика маршрутизации, BGP-4, атрибуты.

G.F. Masich, D.A. Aryshev

Perm National Research Polytechnic University, Perm

ANALYSIS OF WELL-KNOWN MANDATORY ATTRIBUTES OF THE BGP-4 PROTOCOL FOR MANAGING EXTERNAL ROUTING POLICY

The article describes the attributes of the BGP-4 protocol used to manage routes between autonomous systems. The features of using the well-known mandatory attributes AS_PATH, NEXT_HOP, ORIGIN are analyzed.

Keywords: routing policy, BGP-4, attributes.

Современная сеть сегодня – это совокупность Автономных систем (AS – autonomous system), соединенных каналами связи в некую управляемую топологию. Маршрутизация между AS происходит с помощью протокола BGP-4 (Border Gateway Protocol 4), описанного в RFC 4271.

Автономная система – это сеть, находящаяся под одним административным управлением и реализующая внутреннюю политику маршрутизации посредством внутренних протоколов маршрутизации (IGP) [1]. Прототипом AS является государство, которое имеет свою внутреннюю политику, собственное правительство и законы. Примером AS является корпоративная сеть ВУЗА (или научного

центра, или какого-то ведомства), использующая внутренние протоколы маршрутизации OSPF (Open Shortest Path First) или RIP (Routing Information Protocol) и реализующая внутренние политики доступа к сервисам для разных категорий своих сотрудников, фильтрации трафика с целью информационной безопасности и т.д.

Предметом рассмотрения представленного материала является анализ механизмов взаимодействия между AS. Аналогом являются механизмы, законы, международные организации, регулирующие взаимодействия между государствами на земном шаре. Принцип регулирования отношений между автономными системами представлен на рисунке.

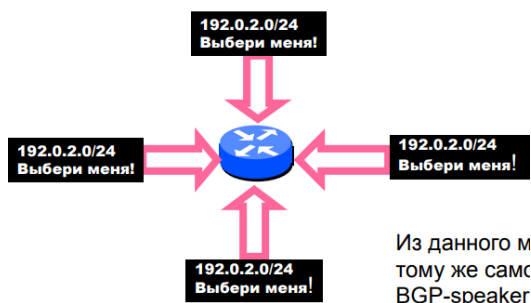


Рис. Схема процесса маршрутизации в роутере

Роутер воспринимает (ассерт) информацию о маршрутах от других AS и выбирает наилучшие маршруты согласно некоторой совокупности атрибутов BGP-4, которыми управляет сетевой администратор. Например, из множества предложений от соседних AS к одному и тому же самому IP-префиксу 192.0.2.0/24 BGP-speaker выберет один маршрут, имеющий более короткий путь доставки IP-пакетов. Отобранные роутером маршруты попадают в IP таблицу маршрутизации, процесс отбора маршрутов называется «импортной политикой», импортная политика порождает выходной трафик [2].

Роутер анонсирует (announce) соседним AS исходящие маршруты согласно выходным правилам, сконфигурированным администратором посредством использования другой совокупности атрибутов BGP-4. Процесс анонсирования известных роутеру префиксов сетей соседним

AS называется «экспортной политикой», экспортная политика порождает входной трафик.

Показанная импортная и экспортная политика пограничного роутера и есть внешняя политика маршрутизации. И важнейшим механизмом управления внешней политикой маршрутизации между AS являются атрибуты BGP-4, которые определяют путь следования трафика между AS. Наиболее существенные атрибуты в протоколе BGP-4 приведены в табл. 1.

Таблица 1

Атрибуты протокола BGP-4

Номер атрибута	Имя атрибута	Категория/тип атрибута	Соответствующий RFC/проект документа
1	ORIGIN	Общеизвестный обязательный	RFC 4271
2	AS_PATH	Общеизвестный обязательный	RFC 4271
3	NEXT_HOP	Общеизвестный обязательный	RFC 4271
4	MULTI_EXIT_DISC	Необязательный нетранзитивный	RFC 4271
5	LOCAL_PREF	Общеизвестный представленный на собственное усмотрение	RFC 4271
6	ATOMIC_AGGREGATE	Общеизвестный представленный на собственное усмотрение	RFC 4271
7	AGGREGATOR	Необязательный транзитивный	RFC 4271
8	COMMUNITY	Необязательный транзитивный	RFC 1997
... 255	Зарезервировано	—	—

Атрибуты маршрута в BGP-4 – это набор параметров, которые позволяют выбрать маршрут: путь следования, степень предпочтения маршрута, NEXT_HOP для маршрута и другие [2]. С каждым сообщением типа UPDATE передается последовательность атрибутов вместе с маршрутом. BGP-маршрутизаторы принимают сообщения UPDATE, обрабатывают и фильтруют их согласно принятым правилам маршрутизации, а затем передают маршруты другим BGP-узлам.

Таблица 2

Обзор обязательных общеизвестных атрибутов

Критерий	AS_PATH	NEXT_HOP	ORIGIN
Функциональность	– Указывает последовательность автономных систем, через которые проходит маршрут до целевой сети; – обеспечивает возможность выбора маршрута на основе длины пути	– Определяет IP-адрес следующего маршрутизатора на пути к целевой сети; – критически важен для правильной передачи пакетов в сети	– Указывает, как был объявлен маршрут IGP, EGP или incomplete – помогает в выборе предпочтительного маршрута на основе источника объявления
Влияние на маршрутизацию	– Более короткие пути предпочтительнее, что может уменьшить задержку и повысить производительность сети; – используется для обеспечения устойчивости маршрутизации и избегания циклов	– Прямо влияет на возможность маршрутизатора достигнуть своей цели; – изменение атрибута может перенаправить трафик через иные маршрутизаторы и изменить топологию маршрутизации	– Влияет на предпочтение маршрутов: маршруты объявленные через IGP имеют высший приоритет по сравнению с EGP и неопределенными маршрутами; – может помочь в разграничении внутренних и внешних маршрутов
Сложность настройки и управления	– Автоматически обрабатывается протоколом BGP-4, минимальная потребность в ручной настройке; – легко отслеживается и анализируется с помощью стандартных BGP-команд	– Автоматически назначается протоколом BGP-4, но может потребовать ручной настройки в сложных сетевых конфигурациях; – требует мониторинга для обеспечения корректной маршрутизации и предотвращения потери пакетов	– Назначается протоколом BGP-4 на основе источника маршрута; – прост в отслеживании и анализе, но требует понимания различий между типами источников
Применение в реальных сценариях	– Часто используется интернет-провайдерами для управления входящими и исходящими маршрутами; – применяется для предотвращения петель маршрутизации и обеспечения эффективного выбора пути	– Широко используется для маршрутизации трафика в корпоративных сетях и сетях интернет-провайдера; – важен для настройки взаимодействия между соседними AS	– Используется для управления предпочтениями маршрутов внутри автономной системы

Атрибуты подразделяются на 4 категории: обязательные общеизвестные, предоставленные на собственное усмотрение, необязательные транзитивные и необязательные нетранзитивные. Каждая реализация протокола BGP-4 должна поддерживать все обязательные общеизвестные атрибуты, кроме них маршрут может обладать одним и более необязательными атрибутами.

Для эффективного использования протокола BGP-4 очень важно понимать принципы работы атрибутов. В табл. 2 представлен обзор трех обязательных атрибутов, который включает такие параметры, как функциональность, влияние на маршрутизацию, сложность настройки и применения. Анализируя данные атрибуты, можно сделать вывод, что каждый из них играет важную роль в управлении политиками маршрутизации и имеет свои уникальные особенности.

Таким образом, понимание и правильное применение этих атрибутов позволяет сетевым администраторам эффективно управлять маршрутизацией, обеспечивать отказоустойчивость и безопасность сетей, а также оптимизировать трафик. В совокупности эти атрибуты являются фундаментальными для реализации политик маршрутизации в современных сетях, что делает их изучение и использование крайне важными для поддержания стабильности и эффективности работы сети Интернет.

Библиографический список

1. Хелеби С., Мак-Ферсон Д. Принципы маршрутизации в Internet: пер. с англ. – 2-е изд. – М.: Вильямс, 2001. – 448 с.
2. Масич Г.Ф. Протокол BGP-4 [Электронный ресурс]. – 2003. – URL: https://masich.ru/images/lectures-pdf/ip/L3-4_BGP4.pdf (дата обращения: 18.04.2024).

Сведения об авторах

Масич Григорий Федорович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, заведующий лабораторией Института механики сплошных сред УрО РАН, г. Пермь, e-mail: masich@icmm.ru

Арышев Дмитрий Алексеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. КС-20-16, г. Пермь, e-mail: aryshevdmritri@yandex.ru

Д.Б. Кузнецов, Т.В. Шадрина

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ ДЕЦЕНТРАЛИЗОВАННОГО УПРАВЛЕНИЯ ПОРТАЛОМ САЙТОВ

В статье описывается разработка математической модели децентрализованного управления порталом сайтов при помощи сети Петри и распределенных автоматов с применением теории множеств. Также приводится обоснование необходимости разработки такой модели.

Ключевые слова: модель, управление, портал, сеть Петри, распределенный автомат.

D.B. Kuznetsov, T.V. Shadrina

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A MATHEMATICAL MODEL FOR WEB-PORTAL MANAGEMENT

The article describes the development of a mathematical model of decentralized web-portal management using a Petri net and distributed machine using set theory. The rationale for the need to develop such a model is also provided.

Keywords: model, management, web-portal, Petri net, distributed machine.

В современном мире сайт является важным инструментом для привлечения клиентов, предоставления информации о продуктах или услугах, установления контакта с потенциальными покупателями, продвижения бренда и увеличения узнаваемости компании. При этом у организаций, имеющих схожую структуру и выполняющих однотипные функции, появляются похожие сайты, что предопределяет дальнейшее развитие разработок в виде объединения их в порталные решения в рамках платформы [1].

По данным Global Digital с каждым годом увеличивается количество пользователей в сети Интернет, на 2023 г. этот показатель составил 64,4 % от мирового населения. С ростом популярности веб-технологий увеличивается и число посетителей веб-порталов, вследствие чего увеличивается нагрузка на серверы, обрабатывающие запросы пользователей к сайтам, что, в свою очередь, повышает вероятность их выхода из

строю. Последствиями такой ситуации могут быть как экономические, так и репутационные потери. Следовательно, возникает необходимость обеспечения надежности веб-портала, о чем также свидетельствует ГОСТ 28806-90, где надежность является одной из основных характеристик качества программного продукта, которая включает в себя отказоустойчивость как подхарактеристику.

В современных системах отказоустойчивость обеспечивается за счет избыточности (резервирования) и балансировки нагрузки. И балансировка нагрузки, и резервное копирование – сложные процессы, которыми необходимо каким-то образом управлять для обеспечения нормальной работы портала сайтов. Следовательно, возникает необходимость в формулировании некоторого набора правил для управления порталом сайтов. Такой набор правил формируется при применении различных моделей и методов управления порталом сайтов.

Результаты анализа существующих моделей и методов управления порталом сайтов приведены в таблице.

Подходы к управлению порталом сайтов

Наименование	Преимущества	Недостатки
Системы управления контентом	Широко применяется, высокая степень изученности	Ключевой недостаток следует из определения, CMS управляют контентом, а не серверами
Стохастические модели прогнозирования	Возможность предвидения поведения как системы в целом, так и отдельных ее компонентов	Не предоставляет инструкций к действиям по устранению или минимизации последствий отказа
Системы оркестрации контейнеров (иерархическое управление)	Автоматизированное управление, следовательно, отсутствие сбоев, вызванных ошибками оператора	Централизованное управление, иерархическая структура, следовательно, наличие единой точки отказа
Бессерверные вычисления (децентрализованное управление)	Позволяет не задумываться об инфраструктуре и ее параметрах (масштабируемости, высокой доступности и т. п.)	Отсутствие единого универсального решения

На основании таблицы можно сделать вывод, что для управления порталом сайтов необходимо использовать бессерверные вычисления, т.е. модели и методы децентрализованного управления, но, так как разработанные технологии, которые позволяют использовать такие вычисления, являются коммерческой тайной, а также вследствие отсутствия единого универсального решения среди таких подходов, возникает необходимость в разработке собственной математической модели. В результате анализа методов моделирования порталов сайтов было принято решение об использовании Петри и распределенных автоматов. С формальной точки зрения взаимодействие агентов в децен-

трализованной системе управления порталом сайтов определяется упорядоченным конечным набором длины $n = 3$, каждый из элементов которого принадлежит некоторому конечному множеству M_i ($1 \leq i \leq n$) (аналогично для $n \neq 3$):

$$c = \langle X, Z, S \rangle, \quad (1)$$

где X – множество входных значений для агента, Z – множество выходных значений для агента, S – множество агентов.

Множество входных значений – вектор бинарных значений длины $n = 6$:

$$X = \langle x_1, x_2, x_3, x_4, x_5, x_6 \rangle, \quad (2)$$

где x_1 – определяет отказал ли какой-нибудь из серверов, x_2 – можно ли считать данный сервер лучшим вариантом для размещения сайтов, x_3 – нужно ли выполнить резервное копирование, x_4 – совершил ли пользователь какое-нибудь действие (добавление или удаление блока или сайта), x_5 – произошла ли ошибка после выполнения действия пользователя, x_6 – нужно ли переместить часть сайтов с данного сервера на другой.

Множество выходных значений – это вектор бинарных значений длины $n = 4$:

$$Z = \langle z_1, z_2, z_3, z_4 \rangle, \quad (3)$$

где z_1 – на данном сервере восстановлена работа сайтов с другого сервера, z_2 – выполнено резервное копирование, z_3 – исправлена пользовательская ошибка, z_4 – выполнено перераспределение (часть сайтов с данного сервера размещены на другом).

Учитывая, что агенты в децентрализованной системе управления имеют одинаковый набор правил, можно провести моделирование только для одного из элементов множества S , с учетом уравнений (2) и (3). В таком случае графическое представление сети Петри будет выглядеть, как показано на рис. 1, где R – выполнение процесса восстановления работы сайтов, размещенных на вышедшем из строя сервере, B – процесс выполнения резервного копирования, C – процесс исправления ошибки, допущенной пользователем при внесении изменений на портале, D – процесс перераспределения сайтов по серверам, x – позиции для задания входного вектора, p – позиции, характеризующие промежуточные состояния, t – переходы между позициями, z – переходы между позициями, формирующие выходной вектор.

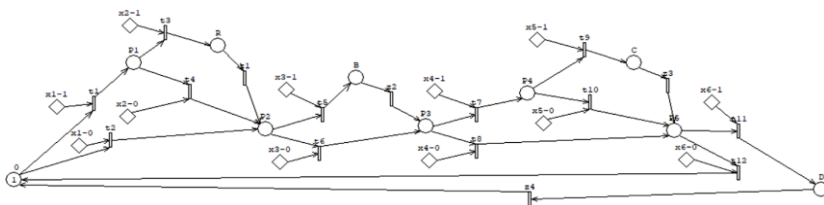


Рис. 1. Сервер портала как сеть Петри

Так как в основе анализа сетей Петри лежит анализ достижимости, проверим срабатывание выходных сигналов в зависимости от входного вектора при помощи имитационного моделирования. Для этого построим сеть Петри, представленную на рис. 1, в программе GRIN и запустим моделирование в автоматическом режиме.

На рис. 1 ромбами отмечены позиции, в которые будут устанавливаться фишки для задания конкретного значения входного вектора. Также фишка установлена в начальную позицию, отмеченную на рисунке нулем. Таким образом, задавая входной вектор при помощи расставления фишек в соответствующие позиции, на модели можно провести эксперименты.

Количество экспериментов можно посчитать по комбинаторной формуле, применимой для подсчета перестановок с повторениями:

$$\overline{A}_n^k = n^k = 2^6 = 64, \quad (4)$$

где n – количество значений переменной, k – количество переменных.

Результаты имитационного моделирования приведены на рис. 2.

На основании результатов, представленных на рис. 2, построенную модель можно считать применимой.

Для определения взаимодействия агентов в системе построим модель портала сайтов как распределенный автомат.

Автомат представляет собой систему:

$$A = \langle X, Z, Q, f, \phi, q_0 \rangle, \quad (5)$$

где $X = \{x_1, x_2, \dots, x_n\}$ – множество входных сигналов, $Y = \{y_1, y_2, \dots, y_m\}$ – множество выходных сигналов, $Q = \{q_1, q_2, \dots, q_k\}$ – множество состояний автомата, $f(q, x)$ и $\phi(q, x)$ – функции переходов и выходов соответственно, q_0 – начальное состояние автомата [2].

С учетом выражения (5) зададим автомат следующим образом. Множество состояний автомата определим как:

$$Q = \{q_B, q_D, q_E, q_R\}, \quad (6)$$

где В – состояние при выполнении резервного копирования, D – состояние при перераспределении сайтов по серверам, E – состояние при возникновении ошибки, R – состояние при отказе одного из серверов.

№	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	Z ₁	Z ₂	Z ₃	Z ₄
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	0	0	0	1
3	0	0	0	0	1	0	0	0	0	0
4	0	0	0	0	1	1	0	0	0	1
5	0	0	0	1	0	0	0	0	0	0
6	0	0	0	1	0	1	0	0	0	1
7	0	0	0	1	1	0	0	0	1	0
8	0	0	0	1	1	0	0	0	1	1
9	0	0	1	0	0	0	0	1	0	0
10	0	0	1	0	0	1	0	1	0	1
11	0	0	1	0	1	0	0	1	0	0
12	0	0	1	0	1	1	0	1	0	1
13	0	0	1	1	0	0	0	1	0	0
14	0	0	1	1	0	1	0	1	0	1
15	0	0	1	1	1	0	0	1	1	0
16	0	0	1	1	1	1	0	1	1	1
17	0	1	0	0	0	0	0	0	0	0
18	0	1	0	0	0	1	0	0	0	1
19	0	1	0	0	1	0	0	0	0	0
20	0	1	0	0	1	1	0	0	0	1
21	0	1	0	1	0	0	0	0	0	0
22	0	1	0	1	0	1	0	0	0	1

№	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	Z ₁	Z ₂	Z ₃	Z ₄
23	0	1	0	1	1	0	0	0	1	0
24	0	1	0	1	1	1	0	0	1	1
25	0	1	1	0	0	0	0	1	0	0
26	0	1	1	0	0	1	0	1	0	1
27	0	1	1	0	1	0	0	1	0	0
28	0	1	1	0	1	1	0	1	0	1
29	0	1	1	1	0	0	0	1	0	0
30	0	1	1	1	0	1	0	1	0	1
31	0	1	1	1	1	0	0	1	1	0
32	0	1	1	1	1	1	0	1	1	1
33	1	0	0	0	0	0	0	0	0	0
34	1	0	0	0	0	1	0	0	0	1
35	1	0	0	0	1	0	0	0	0	0
36	1	0	0	0	1	1	0	0	0	1
37	1	0	0	1	0	0	0	0	0	0
38	1	0	0	1	0	1	0	0	0	1
39	1	0	0	1	1	0	0	0	1	0
40	1	0	0	1	1	1	0	0	1	1
41	1	0	1	0	0	0	0	1	0	0
42	1	0	1	0	0	1	0	1	0	1
43	1	0	1	0	1	0	0	1	0	0

№	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	Z ₁	Z ₂	Z ₃	Z ₄
44	1	0	1	0	1	1	0	1	0	1
45	1	0	1	1	0	0	0	1	0	0
46	1	0	1	1	0	1	0	1	0	1
47	1	0	1	1	1	0	0	1	1	0
48	1	0	1	1	1	1	0	1	1	1
49	1	1	0	0	0	0	1	0	0	0
50	1	1	0	0	0	1	1	0	0	1
51	1	1	0	0	1	0	1	0	0	0
52	1	1	0	0	1	1	1	0	0	1
53	1	1	0	1	0	0	1	0	0	0
54	1	1	0	1	0	1	1	0	0	1
55	1	1	0	1	1	0	1	0	1	0
56	1	1	0	1	1	1	1	0	1	1
57	1	1	1	0	0	0	1	0	0	0
58	1	1	1	0	0	1	1	0	1	1
59	1	1	1	0	1	0	1	1	0	0
60	1	1	1	0	1	1	1	1	0	1
61	1	1	1	1	0	0	1	1	0	0
62	1	1	1	1	0	1	1	1	0	1
63	1	1	1	1	1	0	1	1	1	0
64	1	1	1	1	1	1	1	1	1	1

Рис. 2. Результаты имитационного моделирования сети Петри

Множество выходных сигналов обозначим как:

$$Y = \{y_{ОВ}, y_{ОD}, y_{ОЕ}, y_{ЕР}, y_{ВО}, y_{ДО}, y_{ЕО}, y_{РО}\}. \quad (7)$$

Множество входных сигналов:

$$X = \{x_1, x_2, x_3, x_4, x_5\}, \quad (8)$$

где x_1 – любое внешнее воздействие, приводящее к ошибке, x_2 – ошибку нельзя исправить, и она критична, x_3 – команда на выполнение резервного копирования, x_4 – команда на перераспределение сайтов по серверам, x_5 – ошибка исправлена.

Начальное состояние q_0 – состояние, при котором обеспечивается нормальная работа портала. При построении распределенного автомата введем следующие допущения:

- каждый автомат имеет информацию о состоянии каждого другого автомата;
- два или более собственных переходов не могут произойти одновременно, т.е. переход в новое состояние инициируется единственным автоматом;
- при отказе сервер перестает принимать и получать идентификаторы состояний, т.е. «выпадает» из системы, следовательно, никак не влияет на поведение оставшихся.

С учетом этих допущений зададим распределенный асинхронный автомат для двух серверов графом, представленным на рис. 3.

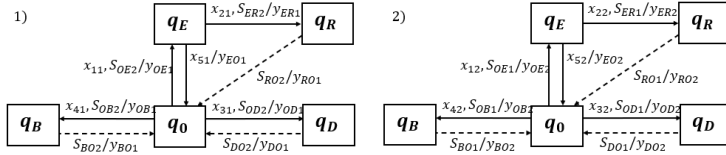


Рис. 3. Портал сайтов как распределенный автомат

На рис. 3 сплошной линией обозначены активные дуги (собственные переходы), а пунктирной – пассивные дуги (вынужденные переходы) петли, сохраняющие текущее состояние автомата, на рисунке опущены. Мультидуги показаны сплошной линией, помеченной разделенными запятой условием собственного перехода и обозначением нового состояния другого автомата, т.е. условием пассивного (вынужденного) перехода [3]. Идентификаторы вида S_{ijk} обозначают факт перехода k -го автомата ($k = 1, 2$) из i -го состояния ($i = q_B, q_D, q_E, q_R$) в j -е состояние ($j = q_B, q_D, q_E, q_R$). Определим портал сайтов как некоторое множество автоматов A длинны $n > 1$. Тогда автомат $a_k \in A$, $1 \leq k \leq n$ и автомат $a_p \in A$, $1 \leq p \leq n$, причем $p \neq k$.

ВХОДНОЙ ВЕКТОР: [0, 0, 1, 0, 0] Автомат 1 Состояния: {'qE': 0, 'qR': 0, 'qD': 1, 'qB': 0, 'q0': 0} Выходы: {'OE': 0, 'ER': 0, 'OD': 1, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 2 Состояния: {'qE': 0, 'qR': 0, 'qD': 1, 'qB': 0, 'q0': 0} Выходы: {'OE': 0, 'ER': 0, 'OD': 1, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 1 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 1, 'BO': 0} Автомат 2 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 1, 'BO': 0} ВХОДНОЙ ВЕКТОР: [0, 0, 0, 1, 0] Автомат 1 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 1, 'q0': 0} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 1, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 2 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 1, 'q0': 0} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 1, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 1 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 1} Автомат 2 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 1} ВХОДНОЙ ВЕКТОР: [1, 0, 0, 0, 0] Автомат 1 Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 0} Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 2 Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 0} Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}	ВХОДНОЙ ВЕКТОР: [0, 1, 0, 0, 0] Автомат 1 Состояния: {'qE': 0, 'qR': 1, 'qD': 0, 'qB': 0, 'q0': 0} Выходы: {'OE': 0, 'ER': 1, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 2 Состояния: {'qE': 0, 'qR': 1, 'qD': 0, 'qB': 0, 'q0': 0} Выходы: {'OE': 0, 'ER': 1, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 1 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 1, 'DO': 0, 'BO': 0} Автомат 2 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 1, 'DO': 0, 'BO': 0} ВХОДНОЙ ВЕКТОР: [1, 0, 0, 0, 0] Автомат 1 Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 0} Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 2 Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 0} Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0} ВХОДНОЙ ВЕКТОР: [0, 0, 0, 0, 1] Автомат 1 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 1, 'RO': 0, 'DO': 0, 'BO': 0} Автомат 2 Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'q0': 1} Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 1, 'RO': 0, 'DO': 0, 'BO': 0}

Рис. 4. Результаты имитационного моделирования распределенного автомата

Запишем логические функции переходов:

$$\begin{aligned}
 q_{Ek}(t+1) &= q_{Ok}(t)(x_{1k}(t) \vee s_{OE_p}(t)); & q_{Rk}(t+1) &= q_E(t)(x_{2k}(t) \vee s_{ER_p}(t)); \\
 q_{Dk}(t+1) &= q_{Ok}(t)(x_{3k}(t) \vee s_{OD_p}(t)); & & \\
 q_{Bk}(t+1) &= q_{Ok}(t)(x_{4k}(t) \vee s_{OB_p}(t)); & & \\
 q_0(t+1) &= q_{Bk}(t)s_{Bop}(t) \vee q_{Dk}(t)s_{Dop}(t) \vee q_{Rk}(t)s_{Rop}(t) \vee q_{Ek}(t)x_{5k}(t).
 \end{aligned} \tag{9}$$

Функции выходов:

$$\begin{aligned}y_{OE} &= q_{Ok}(x_{1k} \vee S_{OEp}); y_{ER} = q_{Ek}(x_{2k} \vee S_{ERp}); y_{OD} = q_{Ok}(x_{3k} \vee S_{ODp}); \\y_{OV} &= q_{Ok}(x_{4k} \vee S_{OBp}); \\y_{EO} &= q_{Ek}(x_{5k} \vee S_{EOp}); y_{RO} = q_{Rk}S_{ROp}; y_{DO} = q_{Dk}S_{DOp}; y_{BO} = q_{Bk}S_{Bop}.\end{aligned}\quad (10)$$

Для анализа достижимости состояний была построена имитационная модель на языке Python, результаты моделирования представлены на рис. 4.

На основании результатов, приведенных на рис. 4, можно утверждать о правильности функционирования разработанной модели.

Таким образом, в результате исследования разработана математическая модель децентрализованного управления порталом сайтов. Модель построена при помощи сети Петри и распределенного автомата Милли. Полученные функции переходов и выходов можно использовать для составления правил децентрализованного управления порталом сайтов, что является поводом для дальнейших исследований.

Библиографический список

1. Езова Н.С., Кузнецов Д.Б. Математическая модель платформы portalного решения для построения системы сайтов // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2013. – № 8. – С. 23–30.
2. Викентьева О.Л., Соловьев А.Е., Файзрахманов Р.А. Дискретная математика: учеб. пособие. – Пермь: ПГТУ, 2009. – С. 112–120.
3. Кузнецов Б.П. Распределенные конечные автоматы // Приборы и системы. Управление, контроль, диагностика. – 2000. – № 2. – С. 9–12.

Сведения об авторах

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@gmail.com

Шадрина Татьяна Владимировна – магистрант Пермского национального исследовательского политехнического университета, гр. РИС-23-1м, г. Пермь, e-mail: shanyat@mail.ru

К.И. Каплина, К.А. Никитина, О.В. Гончаровский

Пермский национальный исследовательский
политехнический университет, Пермь

МОДЕЛЬ СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ «УМНОЙ ТЕПЛИЦЕЙ»

В статье рассматривается разработка и моделирование системы автоматического управления «умной теплицей» по оптимальному набору параметров, а также применение робота-манипулятора, перемещающегося по QR-кодам с целью сокращения количества датчиков. В результате получена модель и проведено исследование выходных параметров, достигнуты необходимые показатели качества системы, выведены формулы для расчета координат охватываемой площади роботом-манипулятором, для перемещения робота в заданную точку предлагается использовать навигационную систему с маяками на основе QR-кодов.

Ключевые слова: умная теплица, модель, система автоматического управления, робот-манипулятор, навигационная система, QR-код.

K.I. Kaplina, K.A. Nikitina, O.V. Goncharovski

Perm National Research Polytechnic University, Perm

THE MODEL OF THE SMART GREENHOUSE AUTOMATIC CONTROL SYSTEM

The article develops a method for developing and modeling a “smart heat” automatic control system based on an adjustable set of parameters, as well as using a robotic manipulator moving along QR codes using a switch for the number of sensors. As a result, a model was obtained and a study was carried out of the output parameters, the obtained quality system indicators that you provided to calculate the area covered by the robot manipulator, to move the robot coordinate to a given point.

Keywords: smart greenhouse, model, automatic control system, robot manipulator, navigation system, Qr-code.

Выращивание различных видов культур, овощей и фруктов на собственных участках становится все более значимой и популярной деятельностью. Но далеко не всегда условия окружающей среды позволяют это воплотить, а использование высокотехнологичного оборудования вызывает сложности у обывателей ввиду высокой стоимости и отсутствия необходимых знаний. При соблюдении необходимых условий и применении передовых технологий возможно успешно выращивать широкий ассортимент растений круглогодично, включая те, которые обычно произрастают в более подходящих климатических условиях.

Понятие «умная теплица» аналогично «умному дому» подразумевает наличие системы автоматизированного управления, которая включает в себя специализированные датчики, исполнительные механизмы, и микроконтроллеры. Например, компания «Донские измерительные системы» производит приборы для создания автоматизированных процессов [1], а компанией «ЭлектроСнаб» разработаны элементы для умных теплиц [2]. Однако эти компании создают только отдельные элементы, не предлагая полную интегрированную систему, доступную для использования конечными потребителями.

В ходе текущего исследования рассматриваются теплицы разных размеров. В небольших теплицах (объектах управления) в период выращивания достаточно использовать управляемые реле, датчики температуры и влажности, чтобы автоматизировать процесс открытия форточек и управления клапаном полива. В объект управления также включены фитолампы и другие источники света, которые поддерживают здоровье растений. Адаптация системы автоматизированного управления (САУ) к конкретным региональным условиям требует тщательной настройки, постоянного мониторинга и анализа получаемых данных [3–5].

Для разработки модели системы автоматического управления «умной теплицей» используется программная среда MatLab Simulink ввиду обширной компонентной базы, гибких настроек элементов и наличия подходящих для решения нашей задачи библиотек. Рассмотрим предлагаемую модель САУ «Умная теплица», представленную на рис. 1.

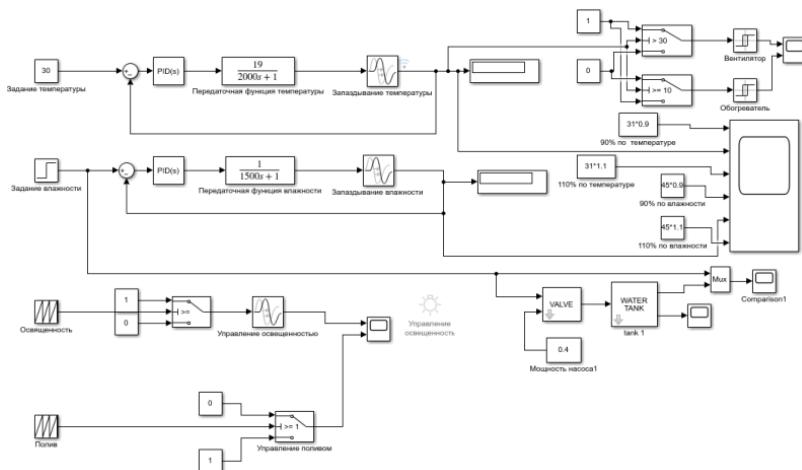


Рис. 1. Схема автоматического управления «умной теплицей»

Для построения модели САУ были получены экспериментальные данные изменения параметров окружающей среды (влажности и температуры воздуха) в течение дня в теплице, представленные на рис.2.

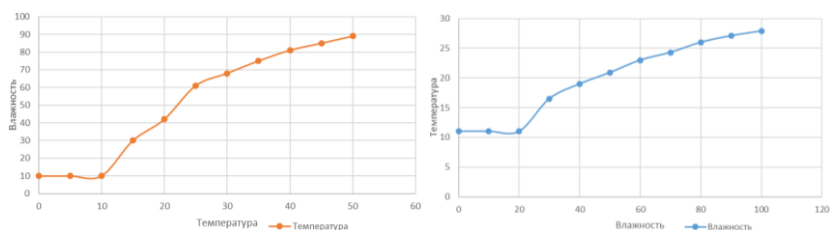


Рис. 2. Результаты эксперимента

Данные эксперимента используются в блоках «Передаточная функция температуры», «Передаточная функция влажности».

Функцию полива растений реализует блок «Управление поливом», выполняющий включение полива по временным промежуткам каждые 6 ч.

Функцию освещенности реализует блок «Управление освещенностью», выполняющий включение ультрафиолетовой лампы, в случае если освещенность в помещении падает ниже 100 000 люкс в течение дня.

Система разбита на блоки по контролируемым параметрам. Контроль температуры осуществляется по датчику влажности, корректировка условий происходит посредством работы ключа и специального оборудования. В случае превышения допустимого значения температуры с помощью ключа и реле включается вентилятор, в случае снижения температуры менее допустимого значения включается инфракрасный обогреватель. Для поддержания необходимого значения используется ПИД-регулятор, настройка которого производилась с помощью PID Tuner, это позволило достичь необходимых показателей качества САУ.

Контроль влажности осуществляется на основании данных датчика влажности: при слишком низком значении происходит включение насоса, обеспечивающее подачу воды из бака.

Контроль освещенности происходит на основании данных датчика освещенности: в случае если значение освещенности опускается ниже допустимого, включается лампочка.

На рис. 3 представлены результаты моделирования блоков для управления влажностью и температурой.

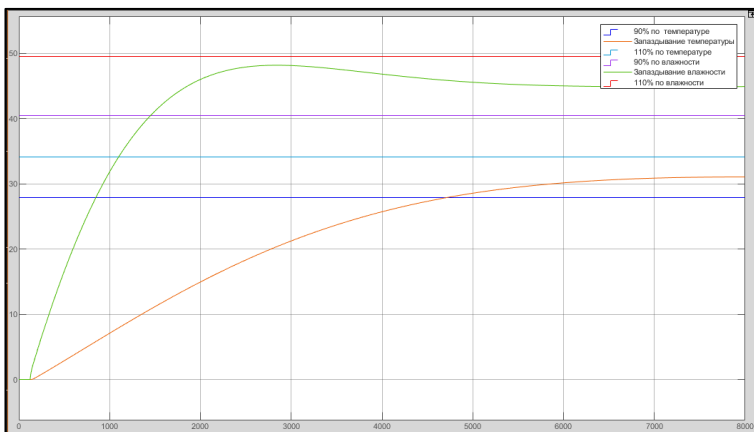


Рис. 3. Результаты управления параметрами температуры и влажности

Оценка показателей качества САУ:

$t_{пп}$ – время переходного процесса, определяется как минимальное время, при котором процесс входит в 10 % зону и больше не выходит за ее пределы; $e(W)$ – статическая ошибка; σ – перерегулирование [6, 7].

$$e(W) = \frac{1}{1-W_p(p)}, \quad (1)$$

$$\sigma = \frac{h_{max}-h_{уст}}{h_{уст}} \cdot 100 \%. \quad (2)$$

Показатели качества для влажности и температуры сведены в таблицу.

Показатели качества САУ «умной теплицей»

Показатель	$t_{пп}, c$	$e(W)$	$\sigma \%$
Влажность	1100	0	0
Температура	4700	0	0

Зададимся следующей целевой функцией (3):

$$F = 0,2t_{пп} + 0,3 \sigma + 0,3e(W). \quad (3)$$

Расчет целевой функции для влажности

$$F = 0,2 \cdot 1100 + 0,3 \cdot 0 + 0,3 \cdot 0 = 220.$$

Расчет целевой функции для температуры

$$F = 0,2 \cdot 4700 + 0,3 \cdot 0 + 0,3 \cdot 0 = 940.$$

Так как не можем пренебрегать перерегулированием, время переходного процесса достаточно большое, но в рамках нашей задачи это не влияет на качество системы.

На рис. 4, а, представлены результаты работы вентилятора и обогревателя. В случае достижения температуры 30°C ключ переключается и запускается работа вентилятора, после снижения значения температуры ключ возвращается в исходное положение и вентилятор отключается. Аналогично происходит с обогревателем, при достижении температуры значения менее 10°C срабатывает переключение ключи и запускается работа обогревателя, при повышении температура более заданного значения ключ возвращается в исходное положение. На рис. 4, б, представлен результат работы систем полива и освещения. Система полива срабатывает через равные временные промежутки 300 мин, система освещения срабатывает при снижении значения освещенности ниже 100 000 люкс.

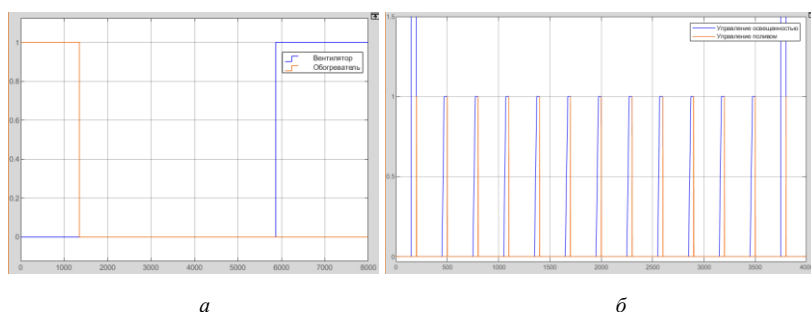


Рис. 4. Результат работы вентилятора и обогревателя (а); полива и освещения (б)

При больших габаритах теплицы установка датчиков на каждом участке посадки для корректного контроля параметров может оказаться нецелесообразной из-за их большого количества. Для упрощения процесса контроля предлагается для крупных теплиц внедрить в систему автоматическую самоходную тележку (АСТ) с манипулятором, на конце которого расположен щуп, оснащенный датчиками для измерения влажности и температуры. Предлагается шасси АСТ выполнить с четырьмя всенаправленными колесами (размеры тележки: 16 на 24 см), а систему управления АСТ оснастить микроконтроллером семейства STM 32. Разработка программного обеспечения АСТ выполняется на платформе

mbed. Манипулятор, управляемый шаговым двигателем, имеет два сустава размером 11 см каждый, а на его конце установлен щуп длиной 10 см. Для примера взяли квадрат грядки 90 на 90 с тележкой в начале координат. Для применения такого механизма необходимо определить, какие точки может охватывать манипулятор для анализа почвы.

Для вычисления координат самой дальней точки, которую манипулятор сможет достичь щупом, можно использовать общую формулу (4):

$$(x, y) = (x_{base} + l_{arm} + l_{probe}, 0), \quad (4)$$

где x_{base} — координата по оси x начала манипулятора (базы); l_{arm} — длина плеча манипулятора (расстояние от базы до плеча); l_{probe} — длина щупа.

Для двухсуставного манипулятора с линейным щупом формулы координат конца манипулятора (2):

$$\begin{aligned} x &= l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2), \\ y &= l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2), \end{aligned} \quad (5)$$

где l_1, l_2, l_3 — это длины манипулятора.

Для того чтобы манипулятор не касался растения во время измерений, его дальняя точка должна находиться на безопасном расстоянии перед растением. В этом случае можно вывести формулу для максимального (6) и минимального (7) безопасного расстояния манипулятора до растения.

$$x_{\max} = x_{plant} - \frac{W_{\text{тел}}}{2}, \quad (6)$$

$$x_{\min} = x_{plant} - L_{manip} - \frac{W_{\text{тел}}}{2} \quad (7)$$

при общей длине манипулятора:

$$L_{manip} = L_1 + L_2 + L_3, \quad (8)$$

где $W_{\text{гр}}$ — ширина грядки; $W_{\text{тел}}$ — ширина тележки; x_{plant} — координата центра растения.

Для перемещения в заданную точку предлагается использовать навигационную систему с маяками в виде QR-кодов (рис. 6).

Изменение направления тележки происходит путем распознавания QR-кодов. Для их создания используется готовая программа QR Code Generator [6].



Рис. 6. QR-коды для направлений: LEFT, RIGHT, FORWARD и TURN

В ходе исследований построена модель системы автоматического управления «умной теплицей» для контроля параметров влажности, температуры, освещенности и автоматического полива.

Представлен вариант передвижения автоматической самоходной тележки для сокращения количества датчиков, выведены формулы для вычисления координат охватываемой территории. А также предложен вариант навигации тележки с помощью QR-кодов для исключения перепрограммирования контроллера при необходимости изменить маршрут движения.

Библиографический список

1. Автоматизация тепличного производства на основе приборов Овен [Электронный ресурс]. – URL: <https://dis-rostov.ru/avtomatizatsia-teplichnogo-proizvodstva-owen> (дата обращения: 12.04.2024).
2. Автоматизированная система управления теплицы [Электронный ресурс]. – URL: <https://eltehbr.ru/actions/avtomatizirovannaya-sistema-up/> (дата обращения: 3.05.2024).
3. Автоматизированная система управления теплицы на базе оборудования ОВЕН [Электронный ресурс]. – URL: <https://elefantkip.ru/news/2021/avtomatizirovannaya-sistema-upravleniya-teplitsy-na-baze-oborudovaniya-ovyen/> (дата обращения: 20.05.2024).
4. Локальные системы управления параметрами объектов гражданского и промышленного строительства: учеб. пособие / Е.Л. Кон, Ю.Н. Хижняков, А.А. Южаков. – Пермь, 2012. –156 с.
5. Динамика робототехнических систем [Электронный ресурс]. – URL: <https://books.ifmo.ru/file/pdf/916.pdf> (дата обращения: 25.04.2024).

6. Qr code generator [Электронный ресурс]. – URL: <https://app.qr-code-generator.com/create/new/?finishedOnboarding=1> (дата обращения: 28.03.2024).

7. Бесекерский В.А. Теория систем автоматического управления. – СПб.: Профессия, 2003. – 752 с.

Сведения об авторах

Каплина Каролина Игоревна – бакалавр Пермского национального исследовательского политехнического университета, гр. СРРС-20-16, г. Пермь, e-mail: kaplina@my.com

Никитина Ксения Андреевна – аспирант кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: ksu2317@yandex.ru

Гончаровский Олег Владленович – кандидат технических наук, доцент кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: 35911953@mail.ru

О.И. Мухин, О.Ю. Галинов

Пермский национальный исследовательский
политехнический университет, Пермь

АВТОМАТИЗАЦИЯ АЛГОРИТМА СОЗДАНИЯ ОТЧЕТОВ

Описывается способ решения анализа действующего приложения по ведению отчетности, создание новой версии приложения для работы с отчетностью, разработке удобных для пользователей интерфейсов, разработке алгоритмов автоматизации проверки отчетов, создания отчетов.

Ключевые слова: отчет, показатель, интерфейс, справочник, приложение.

O.I. Mukhin, O.Yu. Galinov

Perm National Research Polytechnic University, Perm

AUTOMATION OF THE ALGORITHM OF REPORT CREATION

The paper describes the way of solving the analysis of the current reporting application, creation of a new version of the application to work with reporting, development of user-friendly interfaces, development of algorithms to automate the checking of reports, creation of reports.

Keywords: report, indicator, interface, reference book, application.

В связи с разработкой нового приложения отмечается, что текущая версия приложения для управления отчетами является устаревшей и не соответствует современным требованиям эффективного бизнес-анализа. В ходе анализа существующей технологии выявлены следующие основные проблемы:

- устаревшая архитектура: текущая версия приложения построена на устаревших технологиях и не использует современные подходы к разработке программного обеспечения. Это может влиять на производительность, безопасность и возможности расширения функционала;
- неудовлетворительный интерфейс: оперативное управление отчетами требует понятного для пользователя интерфейса. Текущий интерфейс приложения не обеспечивает удовлетворительного пользовательского опыта, что может привести к снижению производительности и эффективности работы пользователей;
- неоптимизированная производительность: текущая версия приложения не оптимизирована с точки зрения производительности, что

может привести к замедлению работы при обработке больших объемов данных. Это особенно критично для бизнес-приложений, ориентированных на обработку и анализ данных.

На основе проведенного анализа целью разработки новой версии десктопного приложения является устранение выявленных недостатков, повышение эффективности работы пользователей и создание современного адаптивного инструмента для управления отчетами [1–3].

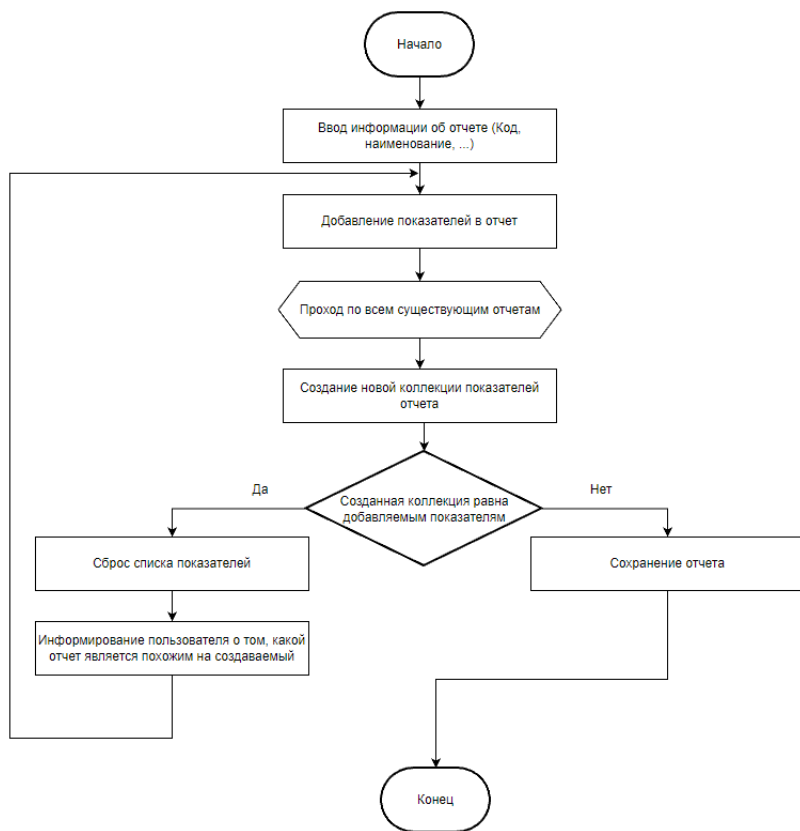


Рис. Блок-схема алгоритма создания отчета

Разработан новый алгоритм, благодаря которому пользователю стало значительно удобнее и быстрее взаимодействовать с отчетами. При попытке создать отчет программа проверит каждый из существующих отчетов и сравнит его с создаваемым. Если отчет с такими пока-

зателями существует, то будет выведено сообщение об ошибке, и программа попросит пользователя заново выбрать показатели для создаваемого отчета. Если же набор показателей является уникальным, то программа создаст отчет и выведет сообщение о том, что отчет создан. Блок-схема алгоритма представлена на рисунке.

В старой версии приложения пользователь должен был вручную сверять, есть ли отчет с такими показателями. Это занимало больше времени, потому что существуют отчеты с большим количеством показателей и было необходимо проходить по всему списку самостоятельно.

Далее представлен листинг с кодом для этого алгоритма

Листинг – Код создания нового отчета

```
// Метод для получения коллекции ключевых показателей для указанной части отчета
public ObservableCollection<Report>
GetKeyfiguresCollection(ReportPart reportPart)
{
    // Создаем новый контекст базы данных
    var context = new MyDbContext();
    ObservableCollection<Report> result = new();

    // Запрашиваем отчеты из базы данных включая
    // необходимые связи
    var itemsForBP = context.Reports
        //.Include(x => x.VirtualRelease)
        .Include(x => x.VirtualReportCode)
        .Include(x => x.VirtualReportId)
        .Include(x => x.VirtualReportTitle)
        .Include(x => x.VirtualFormationFrequency)
        .Include(x => x.VirtualBusinessProcess)
        .Include(x =>
x.VirtualKeyfigure).ThenInclude(y =>
y.VirtualDataStatus)
        .Include(x =>
x.VirtualKeyfigure).ThenInclude(y =>
y.VirtualKeyfigureCategory)
        .Include(x => x.VirtualAnalyticalFeature)
        .Distinct()
        .Where(r => r.ReportIdId ==
```

```

reportPart.ReportID).ToList();

    // Преобразуем полученные данные в
ObservableCollection
    result = new(itemsForBP);
    return result;
}

private async Task<Unit> CreateReport(Unit unit)
{
    // Создаем новый контекст базы данных
    var context = new MyDbContext();
    try
    {
        // Проверяем, содержит ли список
NewStringsReport какие-либо элементы
        if (NewStringsReport.Any())
        {
            // Проходим по всем частям отчета
            for (int i = 0; i < ReportParts.Count;
i++)
            {
                // Получаем коллекцию показателей
для текущей части отчета
                ObservableCollection<Report>
KeyfiguresInReport =
GetKeyfiguresCollection(ReportParts[i]);

                // Проверяем, отличается ли коллек-
ция ключевых показателей от новой коллекции показа-
телей отчета
                if (KeyfiguresInReport !=
NewStringsReport)
                {
                    // Добавляем новую коллекцию
показателей в базу данных и сохраняем изменения
                    context.Reports.AddRange(NewStringsReport);

```

```

        context.SaveChanges();

        // Показываем сообщение об
успешном создании отчета
        await MessageBoxManager

.GetMessageBoxStandard("Успешно", "Новый отчет
успешно добавлен!", ButtonEnum.Ok,
Icon.Success).ShowWindowAsync();
    }
    else
    {
        // Показываем сообщение об
ошибке, если показатели не добавлены в отчет
        await MessageBoxManager

.GetMessageBoxStandard("Ошибка", "Добавьте показа-
тели в отчет!", ButtonEnum.Ok,
Icon.Error).ShowWindowAsync();
    }
}

}
}
catch (Exception e)
{
    ...
}
return Unit.Default;
}

```

Таким образом, данный код обеспечивает добавление нового отчета в базу данных только в том случае, если отчет с такими же параметрами не существует, предотвращая дублирование данных.

Разработанный алгоритм и программная реализация самого приложения могут быть использованы методологами ПАО «ЛУКОЙЛ» для более удобной и быстрой работы с новой версией приложения.

Библиографический список

1. Avalonia UI – Avalonia Docs [Электронный ресурс] – URL: <https://docs.avaloniaui.net/ru/> (дата обращения: 10.05.2024).
2. Avalonia Reactive UI – Reactive UI Docs [Электронный ресурс] – URL: <https://docs.avaloniaui.net/docs/concepts/reactiveui/> (дата обращения: 12.05.2024).
3. MVVM – Модель представления Model-View-ViewModel [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/dotnet/architecture/maui/mvvm> (дата обращения: 12.05.2024).

Сведения об авторах

Мухин Олег Игоревич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: moi@stratum.ac.ru

Галинов Олег Юрьевич – бакалавр Пермского национального исследовательского политехнического университета, гр. АСУ-20-16, г. Пермь, e-mail: oleg9600@gmail.com

А.Д. Вичугов, Е.В. Юзеев

Пермский национальный исследовательский
политехнический университет, Пермь

**ПРОЕКТИРОВАНИЕ ПОДСИСТЕМЫ РАСШИРЕННОГО
МОНИТОРИНГА ЗАКАЗА НА ПРОИЗВОДСТВО
С ДИНАМИЧЕСКОЙ СТРУКТУРОЙ
В «1С:ERP УПРАВЛЕНИЕ ПРЕДПРИЯТИЕМ 2»**

В статье рассматривается механизм динамической структуры заказа на производство в прикладном решении «1С:Управление предприятием 2» и предлагается схема доработок, позволяющая расширить возможности для мониторинга выполнения заказа. Предложенная подсистема должна позволять анализировать сроки выполнения заказа при работе производственных подразделений по разным календарным графикам. В статье представлены исходная и предлагаемая схемы формирования отчета «Мониторинг заказа» в нотации IDEF0.

Ключевые слова: динамическая структура заказа, модели работы, виды рабочих центров, доступность, объект конфигурации.

A.D. Vichugov, E.V. Yuzeev

Perm National Research Polytechnic University, Perm

**DESIGNING A SUBSYSTEM FOR ADVANCED MONITORING
OF A PRODUCTION ORDER WITH A DYNAMIC STRUCTURE
IN «1С:ERP ENTERPRISE MANAGEMENT 2»**

The article discusses the mechanism of the dynamic structure of the production order in the «1С application solution:Enterprise Management 2» and a scheme of improvements is proposed, which allows to expand the possibilities for monitoring order fulfillment. The proposed subsystem should allow you to analyze the timing of order fulfillment during the work of production units according to different calendar schedules. The article presents the initial and proposed schemes for generating the "Order Monitoring" report in IDEF0 notation.

Keywords: dynamic order structure, work models, types of work centers, accessibility, configuration object.

На данный момент самым популярным программным обеспечением в России для управления бизнес-процессами крупных производственных предприятий является прикладное решение «1С:ERP Управление пред-

приятием 2», разработанное на базе платформы «1С:Предприятие 8». Система затрагивает множество областей учета, начиная от кадрового и заканчивая производством.

Для производства продукции в системе оформляется заказ на производство – документ, определяющий, какую продукцию необходимо произвести. Зачастую производимая продукция представляет собой сложные составные изделия, производимые из множества узлов, которые, в свою очередь, также могут являться составными. В связи с большой степенью вложенности изделий производимой продукции процесс определения потребностей в материалах и выделении этапов производства является довольно трудоемкой процедурой.

До версии 2.5.9 в «1С:ERP Управление предприятием 2» определение потребностей в материалах происходило только после ручного формирования этапов производства, причем при изменении количества материалов, доступных для выбранного заказа на производство, приходилось вручную переформировывать этапы производства, чтобы определить актуальные потребности в материалах и трудозатратах.

Начиная с версии 2.5.9 исследуемого прикладного решения, заказ на производство можно формировать, используя динамическую структуру заказа. Динамическая структура заказа позволяет определить потребности в материалах и выделить этапы выполнения заказа сразу после формирования заказа в статусе «К производству». Кроме того, изменения в структуре заказа, вызванные изменением количества доступных материалов, оперативно учитываются в структуре заказа.

На рис. 1 представлена схема формирования заказа на производство с динамической структурой.



Рис. 1. Схема формирования заказа на производство

Ресурсные спецификации описывают производственный процесс и материалы, требуемые для производства конкретного изделия. Желаемая дата выпуска определяет, к какой дате заказ должен быть выполнен. Заказ формируется способом «Точно в срок» [1], таким образом расчет идет от желаемой даты выполнения заказа к вычислению дат начала этапов производства. Схемы обеспечения потребностей описывают, каким способом будет обеспечено перемещение и закупка используемых материалов.

Механизм динамической структуры заказа построен на объектах конфигурации «1С:Предприятия 8» «Регистр сведений», хранящих информацию о структуре заказа в виде записей таблицы. Для просмотра структуры и сроков выполнения заказа по нормативным длительностям производственных этапов и способов обеспечения потребностей предназначен отчет «Мониторинг заказа».

Главными преимуществами механизма динамической структуры заказа на производство являются высокая производительность операций построения и перестроения структуры заказа, а также отсутствие необходимости ручного поиска и переформирования связанных этапов производства при появлении изменений в структуре заказа.

Однако механизм имеет следующие недостатки:

1. Для формирования графика выполнения заказа используются нормативные длительности производственных этапов, не учитывающие количество производимой продукции и графики работы производственных подразделений, в связи с чем даты начала и окончания этапов производства в отчете «Мониторинг заказа» являются некорректными.

2. Отсутствует возможность проанализировать сроки выполнения заказа при работе производственных подразделений по другим календарным графикам, не внося изменений в действующий производственный процесс.

Для решения описанных выше проблем предлагается разработать подсистему расширенного мониторинга заказа, позволяющую определять сроки выполнения этапов заказа с учетом графиков работы производственных подразделений и количества производимой продукции, а также сравнивать сроки выполнения заказа при работе производственных подразделений по разным календарным графикам.

Схема предлагаемой подсистемы представлена на рис. 2.

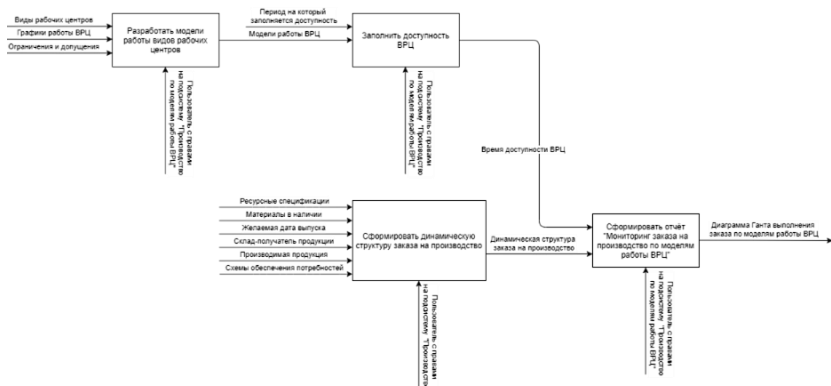


Рис. 2. Схема подсистемы расширенного мониторинга заказа

Виды рабочих центров (ВРЦ) – это группы производственных подразделений, выполняющих одинаковые функции. Для описания их работы предлагается разработать новый объект конфигурации с типом «Справочник», элементы которого будут представлять собой модели работы ВРЦ – каждому выбранному виду рабочих центров будет сопоставляться его календарный график работы, а также будут указываться некоторые допущения для модели в целом, например «Работа без выходных». Для сопоставления видов рабочих центров и их календарных графиков должна использоваться табличная часть справочника.

Согласно графикам работы ВРЦ, пользователь может заполнить доступность рабочих центров на каждый день. Доступность представляет собой время в секундах, доступное для использования рабочих центров. Например, если рабочий центр работает один час в день по стандартному графику работы – пятидневке, то на каждый день, кроме выходных, его доступность будет составлять 3600 с.

Информацию о доступности предлагается хранить в объекте конфигурации с типом «Регистр накопления», а для получения информации о доступности использовать виртуальную таблицу регистра «Обороты», предоставляющую оптимизированные механизмы для быстрого получения хранимых данных. Регистр должен иметь измерения «МодельРаботыВРЦ», «ВидРабочихЦентров» и «ДатаГрафика», а в качестве обязательного ресурса должно выступать числовое поле «Доступность», хранящее время в секундах и указывающее на то, сколько се-

кунд доступен указанный вид рабочих центров в течение указанной даты графика согласно указанной модели работы ВРЦ.

Для регистрации доступности в регистре накопления необходимо использовать объект конфигурации с типом «Документ», который должен иметь реквизиты «МодельРаботыВРЦ» и «ДатаГрафика» для определения используемой модели работы видов рабочих центров и даты, на которую регистрируется доступность соответственно. Виды рабочих центров для регистрации будут получены с помощью обращения к табличной части выбранной модели.

Далее, на основе структуры заказа на производство, сформированной с помощью типовых механизмов прикладного решения, а также с учетом доступности видов рабочих центров будет формироваться сравнительный отчет по срокам выполнения заказа на производство.

Информация о структуре заказа на производство располагается в регистре сведений «Структура заказа». В данном регистре содержится следующая основная информация: производимое изделие, ресурсная спецификация изделия, количество единиц изделия, требуемых к производству и имеющихся в наличии, уровень вложенности и потребляемые для производства изделия материалы.

На основании ресурсной спецификации изделия могут быть получены производственные этапы, задействованные в них виды рабочих центров и их требуемая длительность работы для изготовления единицы изделия. По требуемой длительности работы ВРЦ на единицу изделия и требуемому количеству изделий будет определяться потребность во времени работы ВРЦ для каждого этапа. Циклически уменьшая потребность во времени работы ВРЦ на их доступность в течение дня до тех пор, пока она не исчерпается, можно получить прогнозируемую дату начала выполнения этапа.

Даты начала и окончания выполнения этапов потребуются для отображения в сравнительном отчете.

Отчет должен отображать на диаграмме Ганта периоды выполнения конкретных этапов заказа и всего заказа в целом при использовании разных моделей работы ВРЦ. Диаграмма Ганта в отчете может формироваться с помощью стандартного типа «Диаграмма Ганта» [2]. На рис. 3 представлен макет экранной формы сравнительного отчета.

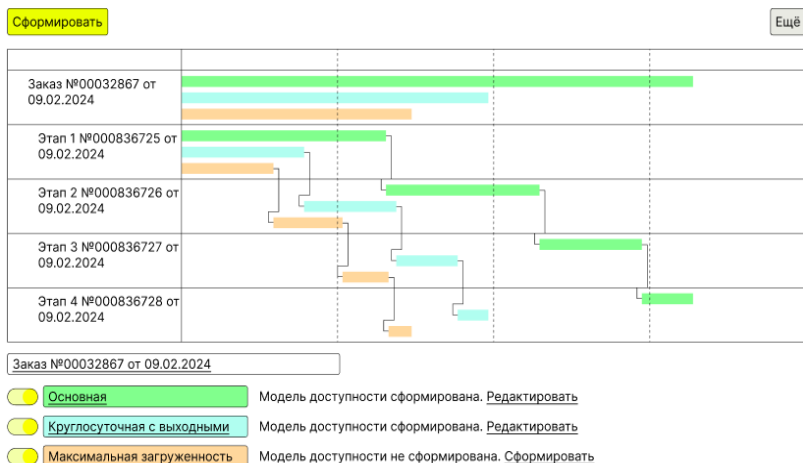


Рис. 3. Макет экранной формы сравнительного отчета

В центре экранной формы располагается диаграмма Ганта, на которой отображаются сроки выполнения заказа на производство и его этапов, заказ выбирается в поле выбора под диаграммой Ганта.

Для каждого этапа отображается его длительность выполнения при работе производственных подразделений по выбранным моделям, состав которых можно редактировать в нижней части экранной формы.

В результате работы спроектирована подсистема расширенного мониторинга заказа на производство.

Использование модели доступности видов рабочих центров позволит определять корректные сроки выполнения этапов заказа, поскольку будут учитываться графики работы производственных подразделений.

Использование моделей работы видов рабочих центров позволит сравнивать эффективность работы ВРЦ по разным календарным графикам, что расширит возможности для анализа их работы.

Перспективой развития подсистемы является расширение списка допущений и ограничений моделей работы видов рабочих центров и добавление других способов планирования выполнения заказа на производство, например «планирование от заданной даты».

Библиографический список

1. Документация по 1С:ERP Управление предприятием 2, редакция 2.5 [Электронный ресурс]. – URL: <https://its.1c.ru/db/erp25doc> (дата обращения: 29.05.2024).

2. Профессиональная разработка в системе «1С:Предприятие 8» / В.А. Ажеронок, А.П. Габец, Д.И. Гончаров, Д.В. Козырев, Д.С. Кухлевский, А.В. Островерх, М.Г. Радченко, Е.Ю. Хрусталева. – М.: 1С:Паблишинг, 2012. – 1377 с.

Сведения об авторах

Вичугов Алексей Дмитриевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: vichugov.leha@mail.ru

Юзеев Евгений Вячеславович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, директор по развитию и методологии группы компаний «ОХТРОН», г. Пермь, e-mail: eyuzeev@oxtron.ru

Д.Б. Кузнецов, Е.П. Малякшин, А.С. Вязов

Пермский национальный исследовательский
политехнический университет, Пермь

ОСОБЕННОСТИ РАЗРАБОТКИ ТОКЕНОВ ДОСТУПА ДЛЯ КРОССПЛАТФОРМЕННОГО МОБИЛЬНОГО ПРИЛОЖЕНИЯ

В данной статье рассматриваются особенности разработки токенов доступа для кроссплатформенного мобильного приложения. Описан алгоритм создания и работы токенов доступа для серверной и клиентской части мобильного приложения.

Ключевые слова: токен, мобильное приложение, ключ, асинхронное хранение, сервер.

D.B. Kuznetsov, E.P. Malyakshin, A.S. Vyazov

Perm National Research Polytechnic University, Perm

FEATURES OF THE DEVELOPMENT OF ACCESS TOKENS FOR A CROSS-PLATFORM MOBILE APPLICATION

This article discusses the specifics of developing access tokens for a cross-platform mobile application. The algorithm of creation and operation of access tokens for the server and client parts of the mobile application is described.

Keywords: token, mobile application, key, asynchronous storage, server.

Для реализации безопасности при обращении к API-шлюзу сервера, в системе сервера мобильного приложения предусмотрена работа с token доступа.

Token – это цифровой ключ, который используется для аутентификации пользователя или устройства в системе [1–5].

Для разрабатываемой системы был выбран JSON Web Token как основа для реализации безопасности при помощи token. JSON Web Token – это открытый стандарт (RFC 7519), который определяет компактный и самодостаточный способ безопасной передачи информации между сторонами в виде JSON объекта. Эта информация может быть проверена и доверена, так как она подписана цифровой подписью.

В системе используется два вида token: Refresh и Access. Данные ключи присваиваются пользователю при авторизации в приложении.

Access token представляет собой ключ со сроком жизни, установленным разработчиком. Как правило, данный token должен иметь время жизни не более 24 ч. Данный ключ выдается пользователю первой авторизации в мобильном приложении и хранится на клиентской части мобильного приложения. Данный ключ используется практически при каждом обращении к серверу за исключением некоторых запросов.

Данный ключ передается в запросах к серверу, а сервер же, в свою очередь, проводит проверку на правильность ключа. Если же ключ пользователя неверный, сервер не выполняет дальнейшие действия маршрута, а возвращает ответ на клиентскую часть о том, что ключ не является действительным. Клиентская часть, в свою же очередь, перенаправляет пользователя на окно авторизации, чтобы пользователь заново ввел свои данные для получения новых ключей доступа.

Также может быть такое, что при обращении к маршруту сервера у Access token пользователя истекло время жизни ключа, тогда сервер возвращает на клиентскую часть сообщение об этом. После этого клиентская часть приложения обращается к маршруту мобильного сервера для обновления Access token с помощью Refresh token.

На рис. 1 приведен алгоритм проверки Access token при обращении к определенному маршруту API.

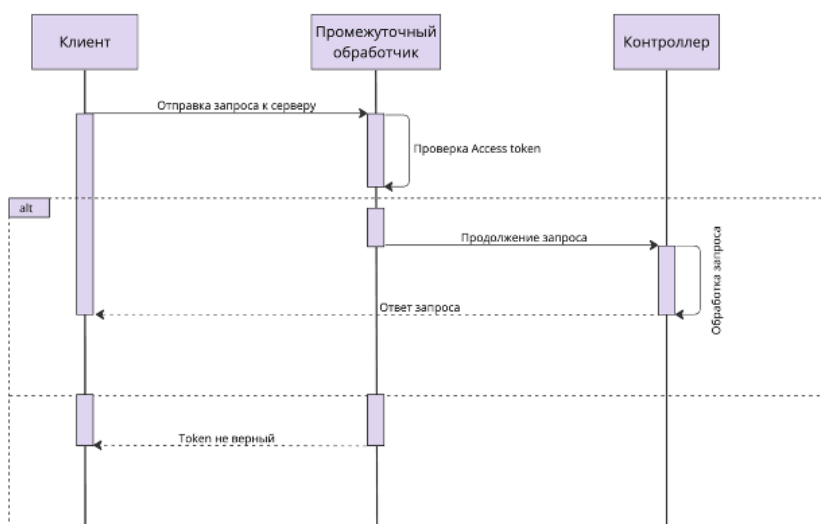


Рис. 1. Алгоритм проверки Access token

На данной диаграмме приведен пример того, как Access token посылается с запросом к API сервера и происходит проверка данного ключа в модуле «Промежуточная обработка», тем самым обеспечивая безопасность доступа к маршрутам путем своей индивидуальности и времени жизни.

Refresh token представляет собой ключ, который используется для обновления жизни Access token. Как правило, данный ключ имеет срок жизни, установленный разработчиком, не более 1 месяца. Данный ключ хранится в базе данных для каждого конкретного пользователя, а также передается в cookies запросов к серверу.

Для обновления жизни данного ключа нет маршрута на сервере. Срок жизни Refresh token обновляется при успешном обновлении Access Token и перезаписывается в базе данных под конкретным пользователем. На рис. 2 приведен алгоритм обновления ключей доступа пользователей.

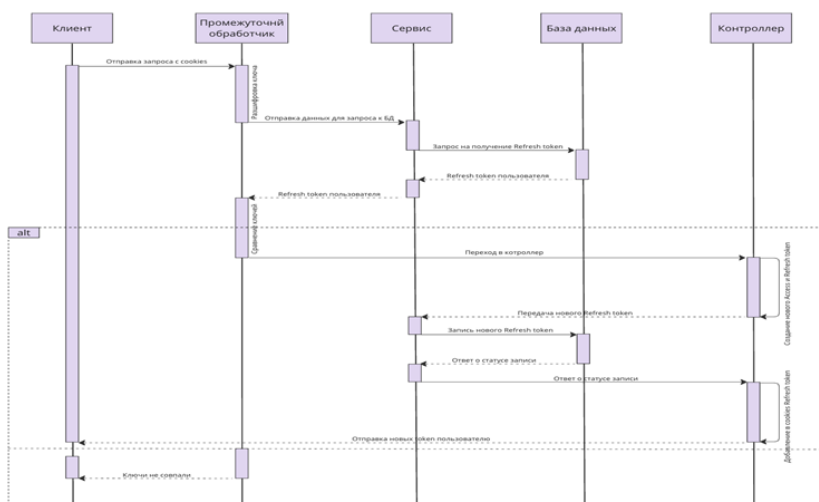


Рис. 2. Алгоритм обновления ключей доступа пользователя

Такой подход позволяет пользователям сохранять непрерывный доступ к ресурсам без необходимости часто вводить свои учетные данные, повышая тем самым удобство использования приложения.

Мобильное приложение разработано на базе фреймворка React Native. React Native – фреймворк для языка программирования

JavaScript, который позволяет создавать нативные кроссплатформенные мобильные приложения. React Native предоставляет обширный выбор инструментов для взаимодействия с памятью смартфона, что обеспечивает доступ к различным способам хранения данных, в том числе хранению токенов доступа.

Для хранения токенов в мобильном приложении используются cookie и async storage. Cookie и async storage позволяют хранить данные на стороне клиента, но они имеют разные характеристики, возможности и области применения.

Async storage позволяет хранить больший объем данных, обычно до пяти мегабайт, в отличие от cookie, который ограничен четырьмя килобайтами данных. Асинхронное хранилище предоставляет возможность асинхронного чтения и записи, что позволяет избежать блокировки основного потока, но данные, хранимые в async storage, могут быть уязвимы к атакам, если устройство пользователя скомпрометировано. Операции с cookie же синхронны, но их преимущество заключается в повышенной безопасности данных, которые можно защитить от чтения вне сервера, и автоматической отправке внутри заголовка запроса.

Для хранения access token используется async storage. Асинхронное хранилище поддерживается и iOS, и android, а данные в нем хранятся в паре «ключ–значение». Поскольку access token обладает коротким периодом жизни, то его хранение в async storage обусловлено компромиссом между удобством чтения и записи и безопасностью, так как утечка краткосрочного токена предоставит доступ злоумышленнику лишь на период жизни этого токена. Такой подход позволяет обеспечить непрерывный доступ пользователя к сессии с непрерывным обновлением краткосрочного токена до момента его протухания.

В отличие от access token, refresh token хранится в cookie, поскольку долгосрочный токен не требует постоянного обновления за счет долгого срока жизни. Cookie, как и async storage, позволяет хранить пары «ключ–значение», но обеспечивает безопасный доступ к данным, поскольку доступ к чтению данных предоставляется только серверу. Таким образом, никакой вредоносный код, внедренный в клиентскую часть, не позволит получить refresh token.

Такой подход позволяет обеспечить сбалансированный способ хранения токенов безопасности, который учитывает как эффективность доступа к данным, так и безопасность доступа к токенам.

Библиографический список

1. Лебедин, В. И. Безопасное хранение токена доступа в Android-приложении. – М.: Молодой ученый, 2019. – 179 с.
2. JWT [Электронный ресурс]. – URL: <https://jwt.io> (дата обращения: 25.05.2024).
3. ReactNative [Электронный ресурс]. – URL: <https://reactnative.dev> (дата обращения: 25.05.2024).
4. Иванова Е.А., Крамаренко Т.А. Кроссплатформенные приложения: учеб. пособие. – Краснодар: КубГАУ, 2020. – 165 с.
5. Арно Л. Проектирование веб-API. – М.: ДМК Пресс, 2020. – 440 с.

Сведения об авторах

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@mail.ru

Малякшин Евгений Павлович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: bigkeksik@bk.ru

Вязов Александр Сергеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-26, г. Пермь, e-mail: frez_er88@mail.ru

А.А. Поспелов, Е.В. Долгова

Пермский национальный исследовательский
политехнический университет, Пермь

АВТОМАТИЗАЦИЯ СТИМУЛИРОВАНИЯ ПРОФЕССИОНАЛЬНОГО РОСТА СОТРУДНИКОВ ЧЕРЕЗ ПООЩРЕНИЕ ДОСТИЖЕНИЙ

В настоящей статье рассмотрена проблема эффективности проведения процесса стимулирования профессионального роста сотрудников через поощрение достижений в компании. Проанализирован существующий бизнес-процесс и выявлены его основные недостатки, спроектирована модель бизнес-процесса. На основе выявленных недостатков спроектирована новая модель бизнес-процесса, которая предусматривает создание системы автоматизации стимулирования профессионального роста сотрудников через поощрение достижений. Разрабатываемая система позволит снизить трудовые и временные затраты, связанные с процессом стимулирования профессионального роста сотрудников.

Ключевые слова: бизнес-процесс, модель, система, автоматизация.

A.A. Pospelov, E.V. Dolgova

Perm National Research Polytechnic University, Perm

AUTOMATION OF EMPLOYEE PROFESSIONAL GROWTH STIMULATION THROUGH ACHIEVEMENT RECOGNITION

This study addresses the issue of efficiently stimulating employee professional growth through achievement recognition within a company. The existing business process was analyzed, identifying its main shortcomings, and a business process model was designed. Based on the identified shortcomings, a new business process model was developed, which includes the creation of an automated system for stimulating professional growth through achievement recognition. The developed system aims to reduce the labor and time costs associated with the process of stimulating employee professional growth.

Keywords: business process, model, system, automation.

В современном мире организации стремятся создать эффективные системы управления персоналом, которые не только удерживают талантливых сотрудников, но и стимулируют их профессиональный рост. Анализируя деятельность организаций, можно отметить, что лишь немногие сотрудники работают на полную мощность, используя весь свой профессиональный, физический, интеллектуальный и психологи-

ческий потенциал. Очевидно, что одни сотрудники трудятся более усердно, затрачивая на работу больше усилий, чем другие. Поэтому результаты деятельности организации зависят не только от способностей и компетентности персонала, но и от уровня его мотивации и стремления к достижению целей. Создание системы, способной эффективно мотивировать сотрудников, становится ключевым фактором успешного управления человеческими ресурсами [1]. Таким образом для стимулирования профессионального роста сотрудников через поощрение достижений в компании среди сотрудников проводится конкурс. Сотрудники компании в добровольном порядке выбирают среди коллег тех, кто, по их мнению, показал особые профессиональные результаты, предложил эффективные варианты решения производственных вопросов и т. д. Символической оценкой профессиональной деятельности сотрудника, неформальной благодарностью является «улыбка». Отметить «улыбкой» сотрудника компании имеют право все сотрудники компании, отправив по электронной почте соответствующее сообщение на выделенный для конкурса адрес. Формулировка обоснования «улыбки» является обязательной. Формулировки обоснований проходят проверку содержания руководителем и могут быть отклонены в случае ненормативной или недопустимой нормами этикета лексики.

Конкурс проводится в три ступени:

- первая ступень. Проводится ежемесячно среди всех сотрудников компании. В результате проведения первой ступени конкурса сотрудники получают обратную связь от коллег о результате своей профессиональной деятельности в рамках месяца;
- вторая ступень. Проводится в конце каждого квартала среди участников первой ступени. В результате проведения второй ступени конкурса руководители отмечают сотрудников, вклад которых имел значительный вес на проекте или для подразделения за текущий квартал, среди участников первой ступени. Список победителей публикуется на стендах и рассылается сотрудникам по электронной почте;
- третья ступень. Проводится в конце года среди всех участников второй ступени. В результате проведения третьей ступени конкурса руководители отмечают сотрудников, вклад которых имел значительный вес на проекте или для подразделения в течении календарного года, среди участников второй ступени. Список победителей публикуется на стендах и рассылается сотрудникам по электронной почте. Победителям вручается вознаграждение от компании.

Для моделирования бизнес-процессов используются диаграммы деятельности. Они дают визуальное представление последовательности действий, потоков управления и данных, помогают выявить существующие проблемы и недостатки в процессах – модель AS-IS, визуализировать оптимизированные с учетом выявленных недостатков процессы – модель TO-BE [2].

Модель AS-IS, описывающая бизнес-процесс проведения конкурса, представлена на рис. 1.

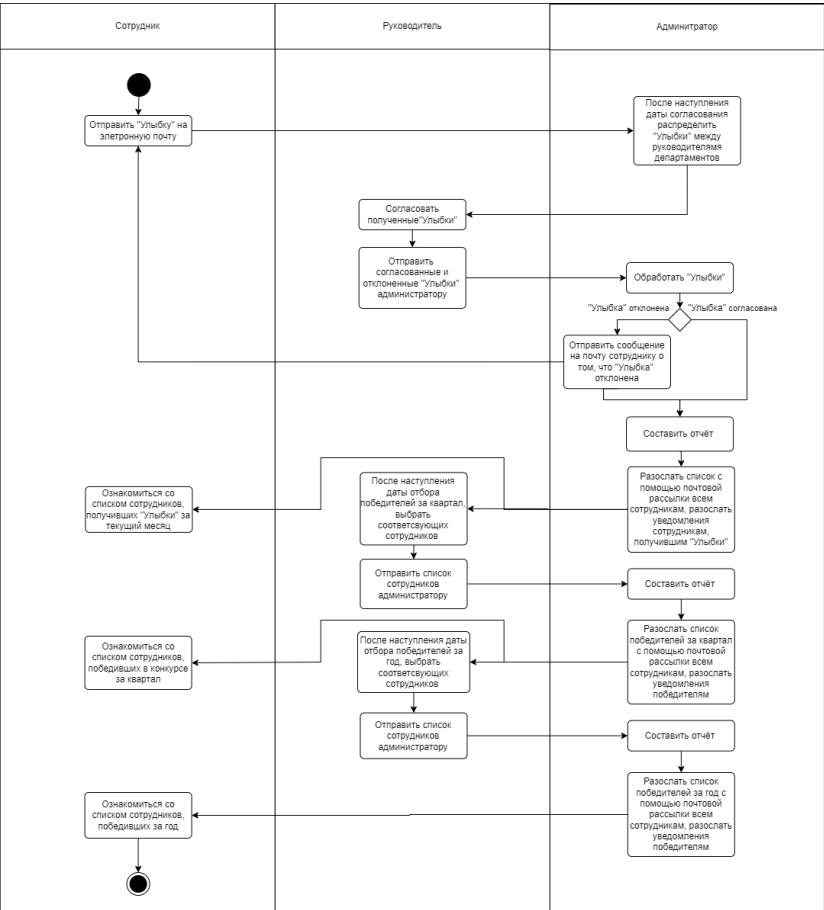


Рис. 1. Диаграмма деятельности бизнес-процесса проведения конкурса

В результате анализа бизнес-процесса выявлены недостатки:

- отправка «улыбки» осуществляется через электронную почту, для редактирования «улыбки» необходимо отправить ее повторно – длительные и трудоемкие операции, которые отталкивают сотрудников;
- отсутствует возможность быстрого ознакомления с победителями конкурса, просмотра полученных «улыбок», истории побед в этапах конкурса;
- отчеты об участниках конкурса составляются вручную;
- данные не структурированы, передаются между этапами бизнес-процесса вручную.

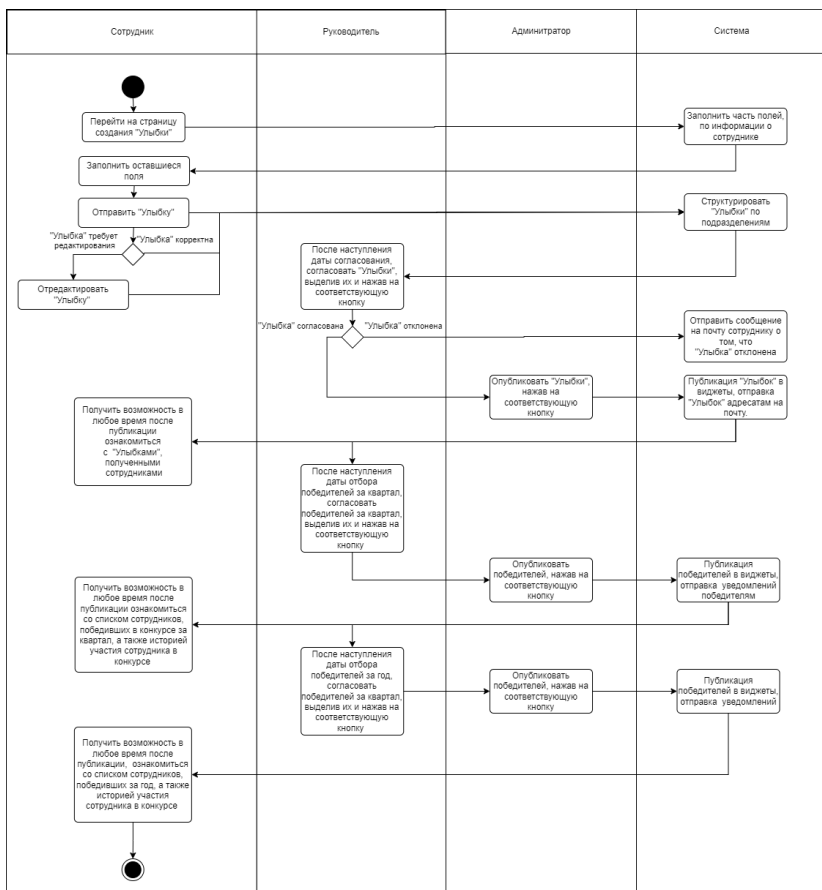


Рис. 2. Диаграмма деятельности нового бизнес-процесса проведения конкурса

Эти недостатки влекут за собой избыточные трудовые и временные затраты, препятствуют эффективному выполнению бизнес-процесса.

Автоматизация процессов, в которых выявлены недостатки, позволит уменьшить трудовые и временные затраты, возникающие во время проведения бизнес-процесса.

Для повышения эффективности проведения бизнес-процесса необходимо разработать встраиваемую программную систему автоматизации стимулирования профессионального роста сотрудников через поощрение достижений.

На основании анализа модели и учета выявленных недостатков разработана модель ТО-ВЕ, представленная на рис. 2.

Программная система:

- заполняет часть полей для более простой и быстрой отправки заявки, позволяет редактировать заявку;
- обеспечивает структурирование данных и за счет единого хранилища данных позволяет не обмениваться информацией между этапами бизнес-процесса;
- автоматически отправляет уведомления по завершении этапа бизнес-процесса;
- предоставляет возможность отображения отправленных и полученных заявок, истории участия в конкурсе, победителей конкурса.

Выявленные недостатки устранены.

Таким образом, в ходе работы спроектированы модели AS-IS и ТО-ВЕ, построены диаграммы деятельности для разработки встраиваемой программной системы автоматизации стимулирования профессионального роста сотрудников через поощрение достижений.

В результате автоматизации устранены недостатки. Система позволит уменьшить трудовые и временные затраты, повысить эффективность проведения бизнес-процесса.

Библиографический список

1. Мотивация и стимулирование трудовой деятельности: учебник и практикум для вузов / С.Ю. Трапицын [и др.]; под общ. ред. С.Ю. Трапицына. – 2-е изд., перераб. и доп. – М.: Юрайт, 2024. – 362 с.

2. Буч Г., Рамбо Д., Джекобсон А. Язык UML. Руководство пользователя. – 2-е изд. – М.: ДМК-Пресс, 2007. – 496 с.

Сведения об авторах

Поспелов Андрей Александрович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-1б, г. Пермь, e-mail: pospelov.dist@yandex.ru

Долгова Елена Владимировна – доктор экономических наук, профессор кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: elena@dolgova

**Ю.А. Безукладников, П.А. Плотников,
Я.Д. Шеин, А.А. Наборщиков**

Пермский национальный исследовательский
политехнический университет, Пермь

**ИССЛЕДОВАНИЕ ВРЕМЕННЫХ ХАРАКТЕРИСТИК
МАТЕМАТИЧЕСКОЙ МОДЕЛИ И МАКЕТА
САМОМАРШРУТИЗИРУЮЩЕГОСЯ АНАЛОГО-ЦИФРОВОГО
ПРЕОБРАЗОВАТЕЛЯ НА БАЗЕ НЕЙРОННОЙ СЕТИ**

В статье приведены описание разработанной математической модели в среде Simulink MatLab и физического макета на базе микроконтроллера STM8, результаты тестирования модели, поставлены цель и задачи исследования. С помощью модели и макета проведены измерения входного сигнала с переменной разрядностью АЦП, проанализированы результаты этих измерений и сделан вывод о влиянии разрядности АЦП при измерении сигналов с разной частотой. Обозначены границы применимости математической модели и сформулированы дальнейшие направления ее развития.

Ключевые слова: самомаршрутизирующийся аналого-цифровой преобразователь, нейронная сеть, основной измерительный нейрон, математическая модель, цифро-аналоговый преобразователь.

**Yu.A. Bezukladnikov, P.A. Plotnikov,
Ya.D. Shein, A.A. Naborshchikov**

Perm National Research Polytechnic University, Perm

**RESEARCH OF TIMING CHARACTERISTICS OF A MATHEMATICAL
MODEL AND LAYOUT OF A SELF-ROUTING MIXED-SIGNAL
CONVERTER BASED ON A NEURAL NETWORK**

The article describes a developed mathematical model in the Simulink MATLAB environment and a physical implementation based on the STM8 microcontroller. It presents the results of testing the model and sets out the purpose and objectives of the study. Using the model and prototypes, measurements of an input signal with varying ADC bit depth were conducted. The results of these measurements were analyzed, and a conclusion was drawn about the effect of the ADC bit depth on signal measurements at different frequencies. The limits of applicability of the mathematical model were outlined, and future directions for its development were formulated.

Keywords: self-routing analog-to-digital converter, neural network, neuron, layout, model, digital-to-analog converter.

В настоящее время во всех сферах с применением цифровых технологий требуется принимать аналоговые сигналы и преобразовывать их в цифровой код [1]. Актуальность применения самонастраивающихся аналого-цифровых преобразователей на основе нейронной сети (САЦП НС) связана с их способностью адаптироваться к переменному сигналу и изменяющимся условиям окружающей среды. Это позволяет использовать САЦП НС в широком спектре приложений, от мобильных устройств до систем промышленного контроля [2]. САЦП НС могут обеспечивать более высокую точность преобразования, а также использовать множество входов для измерения нескольких сигналов параллельно [3]. Кроме того, перестраиваемая структура САЦП позволяет сверять измерения на разных последовательностях разрядов, что помогает найти ошибочные нейроны, если такие есть. Для того чтобы выявить, какой именно нейрон ошибочен, используется динамическая разрядность, которая позволяет сдвигать последовательности нейронов [3]. Таким образом, САЦП НС могут сыграть важную роль в развитии современных электронных систем, предоставляя возможности для создания автономных и интеллектуальных устройств, способных к оптимизации и конфигурации в ответ на изменения во внешней среде и входных данных.

Зачастую нужно измерять несколько сигналов с разными частотами, для решения этой задачи также возможно использовать САЦП НС. Использование дополнительных связей между нейронами повышает устойчивость системы к отказам. Разработанные алгоритмы автоматической диагностики улучшают возможности обнаружения неисправностей в нейронной сети, позволяя проводить диагностику работы измерительных элементов одних нейронов через анализ работы других [3]. Для анализа этих алгоритмов требуется создать математическую модель, которая позволила бы изменять входные данные и разрядности САЦП НС при различных сценариях отказов.

Для достижения поставленной цели предлагается использовать модель, представленную в [4], и провести ее усовершенствование (рис. 1). В этой версии модели были внесены изменения, позволяющие определять длину цепочки основных измерительных нейронов (ОИН) для измерения сигнала начиная с любого из ОИН. Длина цепочки представляется количеством разрядов, используемых для измерения. Для решения этой задачи были добавлены дополнительные ключи, которые позволяют подключать компаратор к любому участку матрицы R-2R. Затем был добавлен дополнительный компаратор, с помощью которого можно производить дополнительное измерение, и входной сигнал для параллельно-

го измерения сразу двух сигналов, поступающих с генератора синусоид. Для того чтобы достичь динамической разрядности, матрица R-2R была замкнута в кольцо путем соединения первого и последнего ОИН, что позволило АЦП проводить измерения независимо от того, какой ОИН соответствует старшему разряду. Данное нововведение облегчает измерение с помощью двух компараторов, потому что появляется возможность задействовать нейроны по кольцу, сдвигая их последовательность.

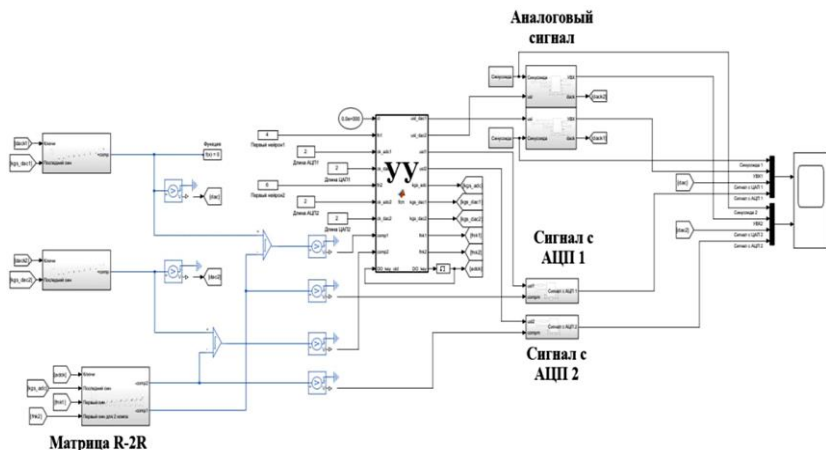


Рис. 1. Модель САЦП НС в MatLab Simulink

Также в процессе модернизации изменился ОИН. Появились дополнительные выходы на второй компаратор, также была проведена связь между каждым из ОИН, что позволяет выбирать старший разряд, который будет подключен к компаратору (рис. 2). Дополнительная связь между ОИН нужна для того, чтобы соединить их между собой для измерения заданной длины матрицы R-2R, которая задается в модели.

Для проверки адекватности математической модели был изготовлен макет САЦП НС, используя в качестве системы управления программируемую логическую интегральную схему (ПЛИС) и микроконтроллеры STM8 (рис. 3) [5]. Для реализации аппаратной части измерительного компонента ОИН были использованы N-канальные MOSFET-транзисторы в качестве ключей между звеньями матрицы R-2R. Для управления ключами и цифровым потенциометром предлагается использовать микроконтроллер STM8S003F3P6 (8 бит, 16 МГц). Для мониторинга состояния каждого ОИН и изменения значений цифрового потенциометра был разработан стенд, включающий контроллер

ESP8266. На базе этого контроллера создан REST-сервер, который хранит данные обо всех ОИН, и клиентское приложение для операционной системы Android. С помощью приложения-клиента можно управлять каждым ключом ОИН отдельно, а также получать актуальную информацию о состоянии ОИН. При выборе соответствующего ОИН отображаемые положения ключей и значения сопротивлений обновляются.

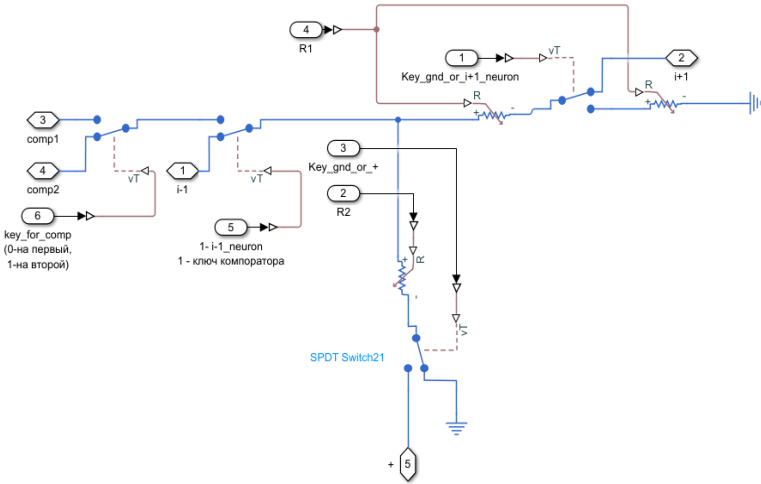


Рис. 2. Модель ОИН в MatLab Simulink

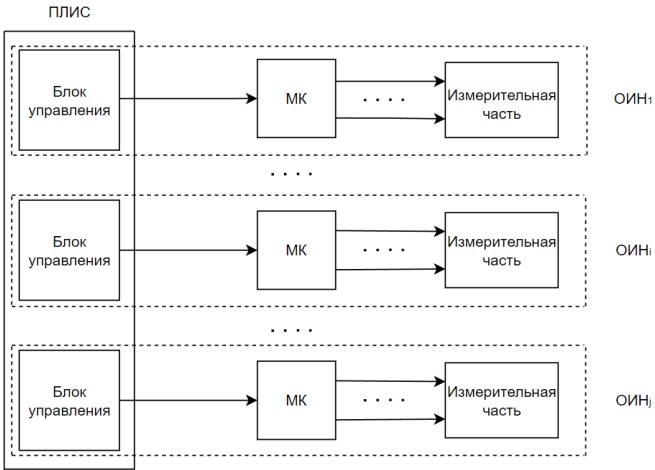
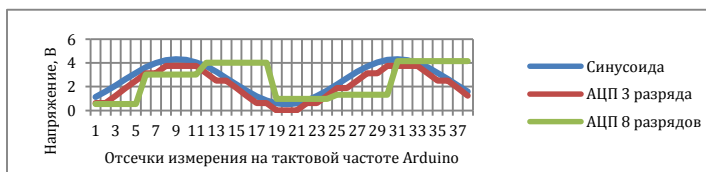


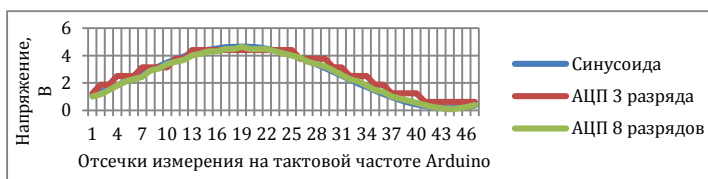
Рис. 3. Реализация управления с помощью ПЛИС измерительной частью каждого ОИН

Для исследования макета была разработана программа на платформе Arduino, в качестве входного сигнала использовался лабораторный генератор синусоидального сигнала, для которого можно изменять амплитуду и частоту генерируемого сигнала на каждом цикле измерения. Затем это напряжение измеряется с помощью САЦП НС и данные передаются на Arduino. Полученные данные Arduino передает по USB на компьютер, где используется программа, написанная на Python, для преобразования данных в таблицу Excel.

Исследование работы САЦП НС при различных величинах сигнала и разрядности, заданной для АЦП, показали следующие результаты (рис. 4). При измерении сигнала частотой 10 кГц и задании числа разрядов 8 из графика видно, что скорости получения отсчета не хватает для того, чтобы корректно отобразить изменения синусоидального напряжения, в то же время при 3 разрядах АЦП форма синусоиды четко прослеживается в оцифрованных значениях (рис. 4, а). При измерении сигнала частотой 100 Гц, напротив, задание числа разрядов 8 позволяет очень точно воспроизвести синусоидальный сигнал, а при трехразрядном АЦП заметны искажения из-за недостаточной разрешающей способности такого АЦП (рис. 4, б).



а

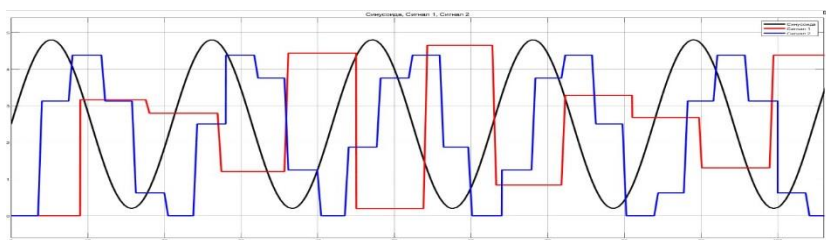


б

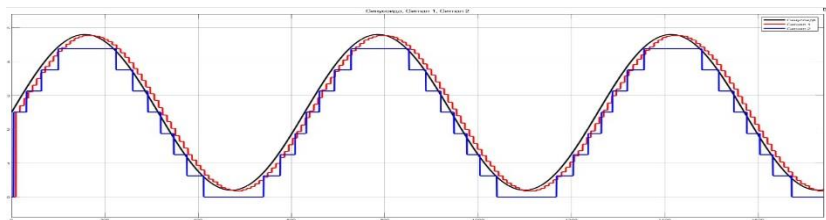
Рис. 4. Вывод данных макета САЦП НС для разных частот измеряемого сигнала: а – частота 10 кГц; б – 100 Гц

Подобное исследование было проведено с помощью математической модели MatLab Simulink, также для различных диапазонов частот (рис. 5). Также заметно, что при высокой частоте время измерения 8 разрядов с помощью САЦП НС не позволяет получить форму синусоиды.

соиды для оцифрованного сигнала, при трех разрядах форма сохраняется (рис. 5, *а*). При снижении частоты обе разрядности АЦП позволяют сохранить форму сигнала, при этом увеличение количества разрядов позволяет увеличить точность измерения (рис. 5, *б*).



а



б

Рис. 5. Вывод данных модели САЦП НС MatLab Simulink для разных частот измеряемого сигнала: *а* – частота 10 кГц; *б* – 100 Гц

Сравнение данных модели и макета САЦП НС позволяет сделать вывод об идентичности их работы, но так как в физическом макете существуют дополнительные задержки при передаче данных с ПЛИС на микроконтроллеры, управляющие аналоговыми ключами, чего нет в модели MatLab Simulink, то результаты совпадают не во всех состояниях. В дальнейшем предлагается сосредоточить работу над учетом задержек в модели MatLab Simulink, чтобы оценить адекватность модели для всех возможных разрядностей и частот измеряемых сигналов. Это поможет использовать модель для реализации алгоритмов самодиагностики ОИН, которые можно будет изначально отладить на математической модели, и только после этого перейти к реализации на физическом уровне, модернизировав систему управления микроконтроллерами и аналоговыми ключами.

Библиографический список

1. Кестер Уолт. Аналого-цифровое преобразование // Редактор оригинального издания: пер. с англ. / под ред. Е.Б. Володина. – М.: ТЕХНОСФЕРА, 2007.

2. Никамин В.А. Аналого-цифровые и цифро-аналоговые преобразователи. Справочник. – М.: Альтекс-А. – 2003.

3. Наборщиков А.А., Посягин А.И., Южаков А.А. Анализ алгоритмов проведения процедуры самодиагностики в аналого-цифровом преобразователе на основе нейронной сети // Вестник Поволжского государственного технологического университета. Сер. Радиотехнические и инфокоммуникационные системы.

4. Модель аналого-цифрового преобразователя переменной разрядности на базе нейронной сети в среде Simulink Matlab / Ю.А. Безукладников, П.А. Плотников, Я.Д. Шеин, А.А. Наборщиков // XV Международный интернет-конф. молодых ученых, аспирантов и студентов «Инновационные технологии: теория, инструменты, практика» (InnoTech-2023). – 2023.

5. Исследование стенда самомаршрутизирующегося аналого-цифрового преобразователя на основе нейронной сети с использованием микроконтроллера STM 8 / И.А. Артемьев, А.А. Наборщиков, А.И. Посягин, В.А. Цыганцев. // Автоматизированные системы управления и информационные технологии. Т. 1.

Сведения об авторах

Безукладников Юрий Алексеевич – бакалавр Пермского национального исследовательского политехнического университета, гр. СТАР-22-1б, г. Пермь, e-mail: byury2004@mail.ru

Плотников Павел Андреевич – бакалавр Пермского национального исследовательского политехнического университета, гр. СТАР-22-1б, г. Пермь, e-mail: vip.pata@bk.ru

Шеин Ярослав Денисович – бакалавр Пермского национального исследовательского политехнического университета, гр. СТАР-22-1б, г. Пермь, e-mail: sheinyarik04@gmail.com

Наборщиков Антон Алексеевич – аспирант кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, ведущий инженер-конструктор, АО «ОДК-СТАР», г. Пермь, e-mail: anton.naborshikov@gmail.com

Г.Ф Масич, Н.А. Добрынин

Пермский национальный исследовательский
политехнический университет, Пермь

МЕТОДЫ ОРГАНИЗАЦИИ ОТКАЗОУСТОЙЧИВОГО ПОДКЛЮЧЕНИЯ КОРПОРАТИВНОЙ СЕТИ К СЕТИ ИНТЕРНЕТ

В современном мире корректная работа сети критически важна для любой организации. Большую роль играет схема подключения к Интернету корпоративной сети, в которой сотни и тысячи пользователей и для которых потеря доступа к внешним сервисам и ресурсам является поистине катастрофой. А поскольку технологии и законы физики ограничивают безотказность компонентов, повышение безотказности может быть достигнуто только за счет встраивания избыточных элементов. Именно этим аспектам, где и какая избыточность повышает отказоустойчивость корпоративной сети передачи данных, посвящена эта статья. Исследования выполняются в интересах Информационно-телекоммуникационной сети ПФИЦ УрО РАН.

Ключевые слова: маршрутизатор, отказоустойчивость, резервирование, канал связи, Интернет, интернет-провайдер, точка обмена трафиком.

G.F. Masich, N.A. Dobrynin

Perm National Research Polytechnic University, Perm

METHODS OF ORGANIZING FAULT-TOLERANT CONNECTION OF A CORPORATE NETWORK TO THE INTERNET

In today's world, the correct operation of the network is critically important for any organization. An important role is played by the Internet connection scheme of the corporate network, in which hundreds and thousands of users and for whom the loss of access to external services and resources is truly a disaster. And since technology and the laws of physics limit the reliability of components, increased reliability can only be achieved by embedding redundant elements. This article is devoted to these aspects, where and what redundancy increases the fault tolerance of the corporate data network. The research is carried out in the interests of the Information and Telecommunication network PSC UB RAS.

Keywords: router, fault tolerance, redundancy, communication channel, Internet, Internet provider, traffic exchange point.

Уместно отметить, что еще в эпоху создаваемой по инициативе ARPA сети ARPANET, предшественницы Internet, основные усилия были направлены на поиск и разработку механизмов противостояния

массированным физическим и электронным атакам [1, 2]. Именно тогда было предложено при проектировании и реализации крупных сетей связи использовать понятие/подход «Graceful Degradation» (изящная деградация). То есть плавное ухудшение качества сети связи при каждом отказе, а именно потеря только части своей функциональности без полной потери работоспособности. Понятию «Graceful Degradation», по нашему мнению, в большей мере соответствует термин ГОСТ 27.002-2015 «Отказоустойчивость» (fault-tolerance) – свойство объекта сохранять свою работоспособность после отказа одного или нескольких элементов в объекте [3]. И, как справедливо отмечается в работе [4], отказоустойчивая система обладает тем уникальным свойством, что ее общая безотказность выше, чем безотказность составляющих ее частей. А искусство использования избыточности заключается в такой интеграции ненадежных компонентов системы, чтобы повреждение одной не повлекло к крушению всей конструкции.

Компонентами сети Internet, согласно модели OSI RM, являются оборудование сети передачи данных, охватывающая первые три уровня модели, и каналы связи, соединяющие, согласно трактовке inter+net, сети на земном шаре. Трактовка слова Internet как «всемирный источник данных» (текстов, видео, музыки, получения справок, магазинов, социальных сетей) не есть предмет представленного материала.

Конструкция Internet. С самого начала в структуре Internet выделяли магистральную сеть (core backbone network) и автономные системы (autonomous systems, AS) – сети, присоединенные к магистрالي. Автономная система (AS) – сеть под единственным административным управлением [1]. Каждая AS использует собственные протоколы маршрутизации, внутри каждой AS своя политика маршрутизации. Все AS имеют уникальный (2^{16}) номер. Магистральные сети, в настоящее время называемые точками обмена трафиком (IXP – Internet Exchange Point), также являются автономными системами со своими номерами AS.

Например, крупнейшая точка обмена трафиком в России MSK-IX в Москве, в Европе AMS-IX (Amsterdam Internet Exchange). Служба пиринга AMS-IX позволяет интернет-сервис провайдеру (ISP – Internet service provider) напрямую подключаться к 878 сетям для обмена интернет-трафиком, используя только одно соединение и кросс-коннект (рис. 1). AMS-IX и MSK-IX используют в своей инфраструктуре топологию «двойное ядро» (рис. 2) [5], обеспечивающую высокую доступность и отказоустойчивость сети.



Рис. 1. Схема подключения ISP к точке обмена трафиком AMS-IX

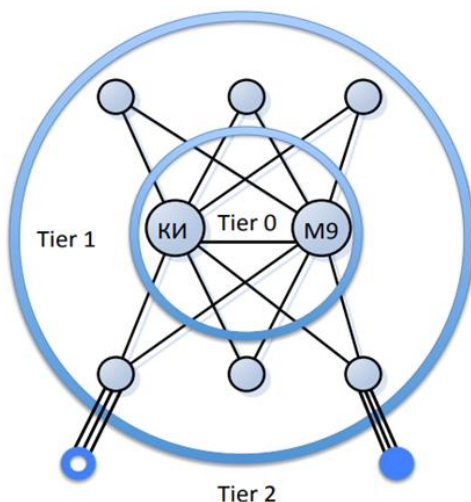


Рис. 2. Топология “двойного ядра” [5]

В этой топологии два параллельных ядра (центральные коммутаторы или маршрутизаторы) работают одновременно, дублируя все критические соединения. Безусловно, это решение достижения отказоустойчивости при подключении к крупнейшим IXP. Однако цена такой реализации доступна только крупным ISP или в рамках национальных проектов. Например, Национальная исследовательская компьютерная сеть России (НИКС) создана по заданию Минобрнауки России в 2019 г. в результате интеграции функционировавших на протяжении 25 лет

отраслевых научно-образовательных телекоммуникационных сетей – федеральной университетской компьютерной сети RUNNet (Russian UNiversity Network) и сети организаций Российской академии наук RASNet (Russian Academy of Science Network). Имеет международные каналы для интеграции с зарубежными научно-образовательными сетями (National Research and Education Networks, NREN).

Разрабатываемая схема подключения к нескольким ISP. Ключевыми особенностями предлагаемой схемы резервирования, показанной на рис. 3, являются следующие аспекты ее реализации.

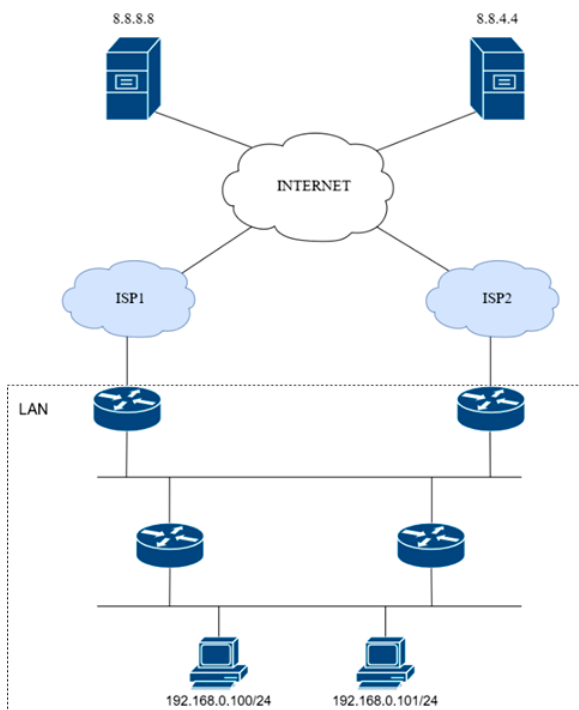


Рис. 3. Подключение корпоративной сети к двум ISP

Использование нескольких провайдеров позволяет избежать единичной точки отказа путем перенаправления трафика через доступного ISP. Такая схема предполагает наличие у корпоративной сети: провайдеро-независимого блока IP адресов (PI – Provider Independent) и собственного номера автономной системы AS (autonomous system). Имея

свой номер AS и IP-адреса, можно использовать Full View для оптимальной маршрутизации или Default Route для экономии ресурсов роутеров. При выборе ISP провайдеров для подключений целесообразно проанализировать их связность с другими сетями, в том числе с крупнейшим IXP.

Географическое резервирование каналов связи предполагает использование физически разнесенных маршрутов для передачи данных в корпоративной сети, что минимизирует риск одновременного отказа всех маршрутов из-за аварийных событий на каналах связи. Пути прокладки каналов связи выявляются при анализе организационной структуры и территориальной распределенности объединяемых офисов.

Автоматическое переключение маршрутов представляет собой технологию, обеспечивающую бесперебойное функционирование сетевых сервисов при сбоях или изменениях в топологии сети. Данный механизм позволяет сетевым устройствам автоматически находить альтернативные маршруты для передачи данных, минимизируя время простоя. Основой автоматического переключения маршрутов является использование протоколов динамической маршрутизации, которые позволяют маршрутизаторам автоматически обмениваться информацией о состоянии сети и маршрутах, адаптируясь к изменениям в реальном времени.

Предлагаемая технология автоматического переключения маршрута к ISP использует IP SLA (Service Level Agreement), Track и EEM (Embedded Event Manager). IP SLA измеряет параметры производительности сети, такие как задержка, потеря пакетов и время отклика. Это достигается путем отправки тестовых пакетов на заданные IP-адреса и анализа полученных результатов. Track используется для мониторинга состояния IP SLA операций. Track-объекты позволяют маршрутизатору отслеживать доступность сетевых ресурсов и в случае изменения их состояния запускать соответствующие действия. EEM позволяет автоматизировать реакцию на изменения состояния сети. EEM настроен для выполнения определенных команд или отправки уведомлений при изменении состояния Track-объектов. Схема автоматического переключения маршрута взаимодействия представлена на рис. 4.

Измерения времени конвергенции (сходимости) сети. Время, которое тратится на перемаршрутизацию по причине сбоев, неисправностей и выхода из строя оборудования и каналов связи, является ключевым показателем успешного резервирования. По нашему мнению, потеря доступа к сервисам в течение минуты по этим причинам являет-

ся великолепным результатом схемного резервирования в корпоративной сети. Диверсии, кибератаки и катастрофы могут увеличить это время в оставшейся части сети до нескольких десятков минут.

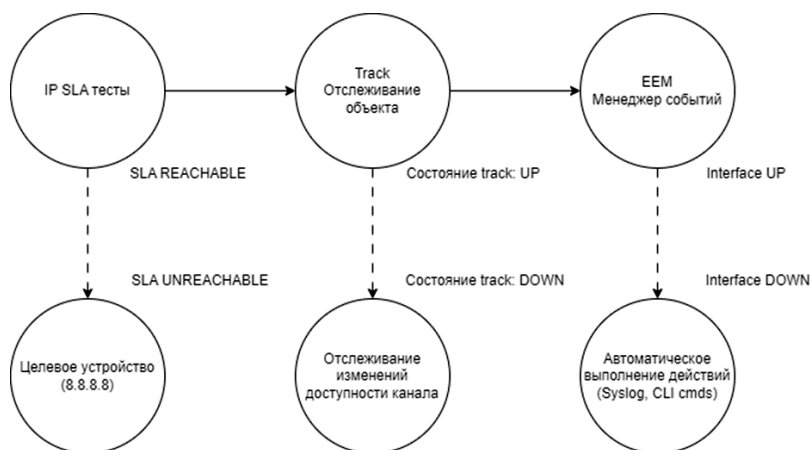


Рис. 4. Схема взаимодействия сетевых технологий IP SLA, track и EEM

Время переключения маршрутов зависит от настроек самих IP SLA-тестов. Threshold – пороговое значение устанавливает верхний предел времени прохождения тестового пакета в оба конца (RTT). Если RTT превышает это значение, операция IP SLA считается неудачной, единица измерения миллисекунды. Timeout – время ожидания определяет максимальное количество времени, в течение которого IOS будет ожидать ответа на тестовый пакет, прежде чем считать пакет потерянным, единица измерения – миллисекунды. Frequency – частота определяет, как часто отправляются тестовые пакеты, единица измерения – секунды. Реальные измерения – предмет дальнейших исследований.

Закключение. Рассмотрены существующие схемы подключения крупных интернет-провайдеров к точкам обмена трафиком. Проанализированы различные подходы и их эффективность, а также интеграция этих подходов в рамках корпоративной сети. Предложены методы резервирования, включающие использование географического резервирования каналов связи с сетями провайдеров Интернет и автоматическое переключение маршрутов.

Библиографический список.

1. Масич Г.Ф. Сети передачи данных: учеб.-метод. пособие. – Пермь: Изд-во ПНИПУ, 2014. – 191 с.
2. Переключение интернет-провайдера (ISP) с использованием резервных маршрутов по умолчанию и отслеживания IP SLA [Электронный ресурс]. – URL: <https://clck.ru/3ApJAn> (дата обращения: 26.05.24).
3. ГОСТ 27.002-2015. Межгосударственный стандарт. Надежность в технике. Термины и определения. – М.: Стандартиформ, 2015. – 24 с.
4. Изящная деградация [Электронный ресурс]. – URL: <https://clck.ru/3AuFDQ> (дата обращения: 26.05.24).
4. Достижения в области технологий пиринга [Электронный ресурс]. – URL: <https://clck.ru/3AvzHy> (дата обращения: 26.05.24).

Сведения об авторах

Масич Григорий Федорович – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, заведующий лабораторией Института механики сплошных сред УрО РАН, г. Пермь, e-mail: masich@icmm.ru

Добрынин Никита Александрович – магистрант Пермского национального исследовательского политехнического университета, гр. АСУ9-23-1м, г. Пермь, e-mail: nikitadobrynin56@gmail.com

М.М. Няшин, А.А. Петренко

Пермский национальный исследовательский
политехнический университет, Пермь

РЕШЕНИЕ ПРОБЛЕМЫ ДОБАВЛЕНИЯ ПОЛЬЗОВАТЕЛЬСКИХ МОДУЛЕЙ В ГОТОВОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Рассматривается проблема подключения пользовательских модулей в готовое программное обеспечение. Рассмотрен процесс компиляции с точки зрения возможности использования его этапов для решения проблемы. Описано решение с использованием предкомпиляции, генерации файлов и модификации системы сборки. Предложена оптимизация времени повторной сборки сгенерированных файлов.

Ключевые слова: этапы компиляции, генерация кода, система сборки, оптимизация времени.

M.M. Nyashin, A.A. Petrenko

Perm National Research Polytechnic University, Perm

THE SOLUTION TO THE PROBLEM OF INCLUDING USER MODULES INTO READY SOFTWARE

The article considers the problem of including user modules into ready software. Process of compilation has been analyzed in terms of possibility of using compilation's stages to solve the problem. The solution that uses precompilation, file generation and a custom build system has been described. An optimization or rebuild time has been proposed.

Keywords: phases of compilation, code generation, build system, time optimization.

В программировании широко распространен подход повторного использования кода для уменьшения вероятности совершить ошибку, удобства обновления и соответствия принципу единственной ответственности. Обычно это реализуется путем подключения необходимого модуля в исходном коде программы. Так как это действие производится очень часто, в языках программирования есть встроенные средства для подключения, например, команда *import* в языке *Python* и директива *#include* в языке *C++* [1–3].

Но иногда возникает обратная задача: есть программное обеспечение и надо предоставить пользователю возможность добавления своих

собственных модулей. Например, есть программное обеспечение, решающее определенную задачу, для решения которой не существует оптимального алгоритма, соответственно, существуют различные приближенные алгоритмы или алгоритмы для частных случаев, тогда необходимо дать пользователю возможность добавить свой алгоритм, исключив при этом необходимость изменять имеющийся код основной программы.

К основным проблемам относится неизвестность списка подключаемых модулей, что является ключевой особенностью данной задачи, так как в противном случае появилась бы необходимость ограничивать пользователя в количестве подключаемых модулей и их названиях. Второй проблемой этой задачи является необходимость импортировать все модули в основной код программного обеспечения несмотря на то, что их список, как было сказано ранее, неизвестен. Так как если не импортировать, то основная часть не сможет узнать даже о существовании пользовательских модулей, что приведет к тому, что они не будут работать. Третьей проблемой является необходимость обработки в основной программе всех реализаций, предоставленных в модулях, иными словами, необходимость написания общего кода для обработки с помощью всех имеющихся подключенных модулей, что точно так же затрудняется тем, что эти модули заранее не даны.

Решение данной задачи на интерпретируемых языках программирования, таких как *Python*, не является трудной, благодаря встроенным средствам, таким как метод *import_module* библиотеки *importlib*, решающим вторую проблему, и возможности обработать в цикле подключаемые модули независимо от их набора благодаря наличию динамической типизации в интерпретируемом языке.

Решение приведенной задачи на компилируемом языке рассмотрим на примере языка программирования *C++*. Для начала необходимо разобраться, на каком этапе должно происходить связывание основной части и пользовательских модулей. Глобально жизненный цикл программы разбивается на компиляцию и исполнения. В процессе исполнения уже поздно подключать модули, поэтому решение проблемы возможно только в процессе компиляции.

Рассмотрим этапы компиляции, чтобы определить тот из них, на котором еще можно осуществить необходимое в этой задаче связывание. Первым этапом компиляции является препроцессирование, на котором происходит обработка специальных инструкций, которые так

и называются «директивы препроцессора», таких как `#define`, `#ifdef`, `#include`. Вторым этапом является трансляция, т.е. перевод с текущего языка в ассемблерный код. Третий этап называется ассемблированием, на нем инструкции ассемблера переводятся в команды машинного кода, которые и будут впоследствии исполняться процессором. Последним этапом компиляции является линковка. Препроцессирование, трансляция и ассемблирование выполняются по отдельности для каждого независимого файла `.cpp`, образуя единицу трансляции, а в процессе линковки эти единицы объединяются в один исполняемый файл.

Необходимо разобраться, на каком этапе можно осуществить связывание. На первый взгляд кажется, что для этого подходит линковка, так как именно на этом этапе из разрозненных единиц трансляции собирается один большой исполняемый файл. Тогда на этом этапе надо бы добавить пользовательские файлы ко всем файлам основной программы и собрать все вместе. Но при таком подходе в коде основной программы не будет никакого упоминания пользовательских модулей, а значит, они не будут использованы, таким образом, данный подход не решает проблему полностью, однако стоит отметить, что необходимость добавления пользовательских модулей в качестве единиц трансляции – это важный этап при решении разбираемой проблемы, про который не надо забывать. На этапах трансляции и ассемблирования сам код не меняется, так как меняется только его язык, значит, эти этапы не только не подходят для решения проблемы целиком, но и не содержат отдельных шагов, которые необходимы для ее решения. Последним рассматриваемым вариантом является этап препроцессирования, во время которого в исходные файлы подставляются файлы заголовков, на этом этапе и надо будет подключить файлы, соответствующие пользовательским модулям. Однако на момент начала этого этапа в списке подставляемых заголовочных файлов в основной части программы уже должны быть в явном виде прописаны те файлы, которые необходимо подключить. Таким образом, начать связывание необходимо до препроцессирования, т.е. до первого этапа компиляции, а значит, до начала самой компиляции.

Ни один из этапов компиляции не подходит для полного решения проблемы, так как ее часть должна быть решена еще до начала компиляции. Таким образом, возникает необходимость в предкомпиляции. Во время предкомпиляции можно запустить скрипт для генерации промежуточного файла с известным названием и известной сигнатурой мето-

дов, чтобы можно было его подключить в файл с кодом в основной части программы, в то время как в самом генерируемом файле можно подключить все пользовательские модули и сгенерировать логику обработки подключенных модулей. Важным пунктом, сильно упрощающим обработку, является унификация подключаемых модулей, например, они могут содержать в своей основе класс, наследуемый от базового класса, имеющегося в основной части программы, а также реализовывать необходимые методы с заявленной сигнатурой благодаря переопределению виртуальных методов родительского класса.

Последней проблемой является необходимость автоматизировать процесс генерации, чтобы пользователю не приходилось вручную заново генерировать промежуточный файл. Это можно сделать с помощью добавления обработки генераторов в систему сборки, состоящую из основного скрипта и вспомогательных файлов, похожих на *cmake*-файлы. Перед началом компиляции в процессе обхода зависимостей во время генерации *makefile*, который будет использован для дальнейшей генерации исходного файла программы, необходимо выявить генераторы и запустить их, чтобы они создали нужные промежуточные файлы и, как следствие, позволили анализатору зависимостей включить в список рассматриваемых файлов заголовочные файлы модулей. Также необходимо подключить файлы исходного кода, упомянутые в описании попытки решения проблемы во время линковки, с этим может справиться специальная инструкция, подключающая во вспомогательный файл сборки названия модулей из другого файла, используемого генератором для подключения модулей в промежуточный файл. Обработку этой инструкции также можно добавить в систему сборки. Таким образом, для добавления собственного модуля пользователю необходимо будет сделать три шага: создать новый, отвечающий изложенным выше требованиям класс с необходимой реализацией, добавить в расположенный в той же папке файл со списком подключаемых модулей и запустить систему сборки, которая дальше все сделает сама.

Для ускорения сборки в C++ есть повторное использование готовых единиц трансляции, полученных в результате ассемблирования, которые используются, если соответствующие файлы не были изменены. В то время как при запуске генератора появляется новый файл или изменяется имеющийся, даже если он ничем не отличается от старого, а значит все зависящие от него единицы трансляции необходимо будет собирать заново, то есть нельзя будет воспользоваться готовыми ре-

зультатами, что замедлит весь процесс сборки. Решением является добавление в генератор проверки необходимости повторной генерации промежуточного файла. Критериями являются отсутствие этого промежуточного файла и его время последнего изменения меньше, чем время изменения генератора, списка подключаемых модулей или заголовочного файла любого из пользовательских модулей. Таким образом повторная сборка промежуточного и всех зависимых от него файлов будет происходить только тогда, когда это действительно необходимо.

Заключение. в результате работы была рассмотрена проблема добавления пользовательских модулей в готовое программное обеспечение. Были рассмотрены этапы компиляции и их применимость для решения данной задачи. Было описано решение, использующее генерацию кода в процессе предкомпиляции средствами собственной системы сборки. Была предложена оптимизация времени повторной сборки генерируемых объектов.

Библиографический список

1. C++ Course notes [Электронный ресурс] // cpp-kt.github.io. – URL: <https://cpp-kt.github.io/cpp-notes/course.html> (дата обращения: 24.05.2024).
2. C++ reference [Электронный ресурс] // cppreference.com. – URL: <https://en.cppreference.com/w/> (дата обращения: 25.05.2024).
3. importlib – The implementation of import [Электронный ресурс] // docs.python.org. – URL: https://docs.python.org/3/library/importlib.html#importlib.import_module (дата обращения: 25.05.2024).

Сведения об авторах

Няшин Максим Михайлович – бакалавр Пермского национального исследовательского политехнического университета, гр. РИС-20-16, г. Пермь, e-mail: megamax143.13@gmail.com

Петренко Александр Анатольевич – кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: hawk321@mail.ru

В.И. Князев, А.И. Князев

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ ДЛЯ КЛАССИФИКАЦИИ ДЕЙСТВИЙ ЧЕЛОВЕКА ПО КАДРУ С КАМЕРЫ

В статье проведен анализ существующих классификаций действий человека. Классификация включает такие методы, как векторный, который имеет небольшую точность при малом количестве эпох, но зато не требовательный к ресурсам. Метод опорных точек, который заключается в обнаружении ключевых точек на теле человека, таких как пальцы, локти, колени и т.д., точность которого выше, чем у векторного, но и требования выше. Также метод, основанный на LSTM-слое, точность которого самая высокая при одном количестве эпох, но этот метод более требователен к ресурсам. Проанализированы и другие методы и разобраны их ключевые достоинства и недостатки. Во второй части статьи описан процесс создания своей нейронной сети. Объяснен принцип функции потерь и его особенности. Описана архитектура модели нейронной сети, включая функцию ReLU и инициализацию сети. Описан процесс обучения сети на 10 000 эпох. Проанализирован полученный график обучения и матрица ошибок этой сети.

Ключевые слова: нейронные сети, компьютерное зрение, распознавание движений, видеоаналитика.

V.I. Kniazev, A.I. Kniazev

Perm National Research Polytechnic University, Perm

CONVOLUTIONAL NEURAL NETWORK DEVELOPMENT FOR ONE FRAME HUMAN ACTION CLASSIFICATION

The article analyzes existing classifications of human actions. The classification includes methods such as: Vector, which has low accuracy with a small number of epochs, but is not resource-demanding. The reference point method, which consists of detecting key points on the human body, such as fingers, elbows, knees, etc., the accuracy of which is higher than that of the vector method, but the requirements are also higher. Also a method based on an LSTM layer, the accuracy of which is highest for the same number of epochs, but this method is more resource-demanding. Other methods are also analyzed and their key advantages and disadvantages are discussed. The second part of the article describes the process of creating your own neural network. The principle of the loss function and its features are explained. The architecture of the neural network model is described, including the ReLU function and

network initialization. The process of training a network for 10 000 epochs is described. The resulting training schedule and the error matrix of this network were analyzed.

Keywords: neural networks, computer vision, motion recognition, video analytics.

В современном мире классификация действий людей играет важную роль в таких областях, как компьютерное зрение, робототехника, видеонаблюдение, анализ поведения и виртуальная реальность. Она позволяет компьютерным системам распознавать и интерпретировать действия людей на основе видеоданных или других сенсорных данных. В данной статье мы рассмотрим методы и технологии, используемые для классификации действий людей, их применение в различных областях, а также напишем свою нейронную сеть и обучим ее на данных о движении компьютерных персонажей в 3D-редакторе Unreal Engine 4.

Анализ методов распознавания действий. В работе [1] рассматриваются следующие подходы: 3D ResNet, анализ векторной модели человека и нейронная сеть с LSTM-слоем.

С помощью сети 3D ResNet итоговая точность составила 56,33 %. Причина низкой точности: присутствие нескольких людей в одном кадре и присутствие посторонних предметов в кадре. Обучение и применение 3D ResNet требуют больших вычислительных ресурсов, так как модель имеет большое количество параметров и требует значительного объема памяти для обработки видео. В целом 3D ResNet является мощным инструментом для распознавания действий человека в видео, но его применение следует оценивать с учетом указанных плюсов и минусов в конкретной задаче.

При анализе векторной модели человека итоговая точность составила 47,32 %. Стоит отметить, что 10 эпох недостаточно для точного обучения данным методом. Анализ векторной модели может помочь снизить размерность пространства признаков, удаляя ненужные или коррелирующие признаки, что может улучшить производительность модели. Векторная модель может быть применена к различным типам действий и задачам распознавания, таким как распознавание жестов, походки, мимики и других действий человека. Качество и разнообразие данных, используемых для обучения векторной модели, могут существенно влиять на ее производительность. Недостаточное количество или неправильно размеченные данные могут привести к низкой точности распознавания.

Последний метод, рассматриваемый в статье [1] – распознавание действий нейронной сетью с LSTM-слоем с изменением входных данных

– вместо набора фреймов из видео стал подаваться набор векторов тел людей в кадре (итоговая точность 77,56 %). В данном методе отмечается высокая точность, но для получения такой точности необходимо большое количество времени и использование нескольких моделей нейронных сетей. Из ключевых особенностей данного метода: учет последовательности: LSTM-слои позволяют учитывать последовательность входных данных, что особенно полезно для распознавания действий, где порядок событий имеет значение. LSTM способны запоминать и обрабатывать информацию из предыдущих шагов, что помогает модели улавливать долгосрочные зависимости в последовательности.

Из статьи [2] можно подчеркнуть еще несколько методов, такие как: локальный двоичный шаблон, метод опорных точек и метод оптического потока для точек.

Локальный двоичный шаблон (LBP). Этот метод достиг 80,55 % результатов распознавания. Метод LBP хорошо справляется с изменениями освещения, что делает его эффективным для распознавания действий в различных условиях освещения. Метод LBP требует меньше вычислительных ресурсов по сравнению с некоторыми другими методами распознавания действий. В случае работы с высокоразмерными данными метод LBP может столкнуться с проблемой эффективности и склонности к переобучению.

Метод опорных точек (Keypoint Detection). Этот метод заключается в обнаружении ключевых точек на теле человека, таких как концы пальцев, локти, колени и т.д. Из плюсов данного метода – точность и эффективность. Метод опорных точек обеспечивает высокую точность при определении позы человека. Он может обнаруживать ключевые точки на теле человека с высокой степенью точности. Метод опорных точек может быть реализован с использованием легковесных моделей, что делает его эффективным в реальном времени на различных устройствах. Из минусов же стоит отметить, что данный метод сильно зависит от «шума» и искажений на изображении.

Метод оптического потока для точек (Optical Flow for Keypoints). Оптический поток может использоваться для отслеживания движения ключевых точек между кадрами видео. Optical Flow позволяет учитывать движение объектов и фоновых элементов, что делает его полезным для анализа динамических сцен, таких как видео с людьми, транспортом и т.д. Optical Flow может быть использован для непрерывного отслеживания движения объектов в реальном времени,

что делает его подходящим для приложений, требующих непрерывного анализа видеопотока. Из недостатков данного метода – неустойчивость к большим движениям: Optical Flow может столкнуться с проблемой «слипания» при больших движениях объектов.

Также стоит рассмотреть метод из статьи [3], а именно Convolutional Neural Network. Из плюсов данного метода стоит отметить возможность переиспользовать обученные признаки для различных задач распознавания действий. Например, модель, обученная на одном наборе данных, может быть применена к другому набору данных без полного переобучения. Это особенно полезно, если у вас есть ограниченное количество размеченных данных для обучения. Но в отличие от LSTM CNN не учитывает последовательность входных данных напрямую. Это может быть недостатком при распознавании действий, где порядок событий имеет значение. Однако можно использовать различные подходы для объединения информации о последовательности с помощью CNN.

Разработка и обучение своей нейронной сети. На основе информации о рассмотренных методах напомним свою нейронную сеть, в основе которой будет лежать сверточный слой. Для этого опишем основные этапы создания нейронных сетей, проанализировав, как их корректно настроить и какие параметры выбрать.

Выбор функции потерь и метрики оценки точности. Функции потерь используются в машинном обучении и искусственном интеллекте для оценки того, насколько хорошо модель справляется с поставленной задачей. Основная цель функций потерь – измерить расхождение между прогнозируемыми значениями модели и фактическими данными. Путем минимизации функции потерь модель обучается на данных, чтобы оптимально предсказывать целевую переменную. Cross-Entropy Loss широко используется в задачах классификации, особенно при работе с многоклассовыми задачами. Она помогает измерить ошибку между предсказанными вероятностями классов и истинными метками классов.

Метрика F1 позволяет учесть, как точность, так и полноту классификации, обеспечивая баланс между этими двумя показателями. При оценке точности классификации обычно нельзя просто опираться на один показатель, поскольку увеличение точности зачастую снижает полноту и наоборот. F1-мера учитывает обе эти характеристики и дает обобщенную оценку качества работы модели классификации.

Данные для обучения. В качестве обучающих данных были взяты бесплатные анимации для 3D-движка unreal engine 4. Из этих анимаций были извлечены координаты (x, y, z) для ключевых точек человека. Для того чтобы координаты были в заданных пределах, все скелеты анимаций были развернуты таким образом, чтобы тело человека было в вертикальном положении, а нижнее ребро тела (между точками таза) лежало на оси x. Затем все координаты были нормализованы таким образом, чтобы иметь значения от -1 до 1. Было получено 7 классов действий. На вход сети данные подаются в формате двухмерной матрицы, состоящей из последовательностей координат всех точек по разным осям, и имеет следующий вид:

$$M = [[x1, x2, ..., xn], [y1, y2, ..., yn], [z1, z2, ..., zn]].$$

Архитектура модели нейронной сети. Модель состоит из сверточного слоя и трех полносвязных слоев. Между слоями стоят слои нормализации данных для того, чтобы сигнал от данных держался в заданных границах. В качестве функции активации используется ReLU.

Функция активации ReLU (Rectified Linear Unit) определяется простым математическим правилом: для каждого входного значения x она возвращает максимум из нуля и самого значения x .

При инициализации нейронной сети задаются входные размеры, которые зависят от датасета и влияют на входные и выходные размеры слоев, это количество ключевых точек и количество координат каждой точки.

Также при инициализации нейронной сети происходит инициализация начальных весов. Она играет важную роль в обучении нейронных сетей и позволяет избежать затухающего или взрывающегося градиента.

Обучение. Обучение проводилось на 1000 эпох. Датасет состоит из более чем 7000 экземпляров данных, которые были поделены таким образом, что 80 % данных использовались при обучении нейронной сети, а 20 % для оценки точности. Данные предварительно были перемешаны с динамичным learning rate [4]. Этот параметр изменялся, когда значение функции потерь выходило на плато. Результаты обучения видны на графиках (рис. 1). Также согласно матрице запутанности (рис. 2) нейронная сеть научилась хорошо определять, когда человек стоит, а когда идет. Понимает, когда человек перезаряжается. Путается модель в случае с ранением человека и с передвижением ползком. Вероятно, это из-за того, что зачастую анимации начинались из положения стоя, а сеть классифицирует действие по одному кадру, из-за чего на вход мог попасть тот кадр, где человек еще стоит.

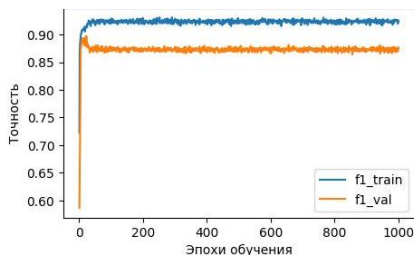


Рис. 1 График обучения

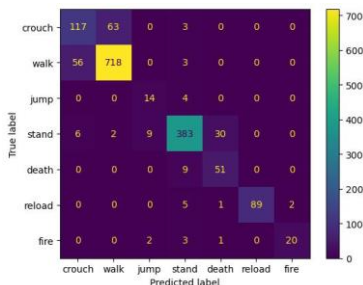


Рис. 2. Матрица ошибок

Точность модели на валидационных данных 0,87 по метрике f1.

Заключение. Были проанализированы существующие методы классификации действия человека по данным о ключевых точках. Предложена и обучена своя модель классификации действий человека по трехмерным координатам ключевых точек человека с одного кадра. С кодом обучения нейронной сети, подготовки данных и датасет можно ознакомиться в GitHub [5].

Библиографический список

1. Марьенков А.Н., Приходько А.А. Анализ методов классификации действий человека на видеоизображении [Электронный ресурс] // Прикаспийский журнал: управление и высокие технологии. – 2021. – № 1 (53). – URL: <https://cyberleninka.ru/article/n/analiz-metodov-klassifikatsii-deystviy-cheloveka-na-videoizobrazhenii> (дата обращения: 25.04.2024).
2. Азаренко К.А., Каунг Мьят Ньейн, Белов Ю.С. Обзор методов для распознавания действий человека [Электронный ресурс] // E-Scio. – 2019. – № 5 (32). – URL: <https://cyberleninka.ru/article/n/obzor-metodov-dlya-raspoznavaniya-deystviy-cheloveka> (дата обращения: 25.04.2024).
3. Шолтанюк С.В., Бу Ц., Недзьведь А.М. Методы обработки видеопоследовательностей со скоплениями людей для определения закономерностей их движения [Электронный ресурс] // Вестник Полоцкого государственного университета. Сер. С. Фундаментальные науки. – 2024. – № 1. – URL: <https://cyberleninka.ru/article/n/metody-obrabotki-videoposledovatelnostey-so-skopleniyami-lyudey-dlya-opredele-niya-zakonemernostey-ih-dvizheniya> (дата обращения: 03.05.2024).

4. Документация по планировщику learning rate [Электронный ресурс] // PyTorch. – URL: https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.ReduceLROnPlateau.html (дата обращения: 09.05.2024).

5. Код проекта [Электронный ресурс] // GitHub. – URL: https://github.com/Creearc/Action_Classification (дата обращения: 29.05.2024).

Сведения об авторах

Князев Виктор Игоревич – бакалавр Пермского национального исследовательского политехнического университета, гр. ЭС-22-16, г. Пермь, e-mail: viktor.knyazev2018@yandex.ru

Князев Александр Игоревич – аспирант кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: knxandr@rambler.ru

И.А. Васенин, С.Ф. Тюрин

Пермский национальный исследовательский
политехнический университет, Пермь

**ТОПОЛОГИЧЕСКОЕ МОДЕЛИРОВАНИЕ ЛОГИЧЕСКИХ
ЭЛЕМЕНТОВ LUT, ИСПОЛЬЗУЮЩИХ ПОЗИЦИОННЫЙ
КОД НАБОРА ПЕРЕМЕННЫХ И LUT С УНИТАРНЫМ
КОДИРОВАНИЕМ**

В статье рассматриваются такие аспекты, как LUT (справочная таблица), способы задания кодирования для LUT с применением обычного, унитарного и комбинированного способов, архитектура FPGA и топология схем. Проведение симуляций для снятия характеристик с последующим сравнением.

Ключевые слова: логические функции, ПЛИС, LUT, унитарный кодирование, логические компоненты, топология.

I.A. Vsenin, S.F. Tyurin

Perm National Research Polytechnic University, Perm

**TOPOLOGICAL MODELING OF LOGICAL LUT ELEMENTS
USING THE POSITIONAL CODE OF A SET OF VARIABLES
AND LUT WITH UNITARY CODING**

The article discusses aspects such as LUT (reference table), ways to set encoding for LUT using conventional, unitary and combined methods, FPGA architecture, and circuit topology. Conducting simulations to remove characteristics and then compare them.

Keywords: logical functions, FPGA, LUT, unitary coding, logical components, topology.

В современной цифровой схемотехнике программируемые логические интегральные схемы (ПЛИС или PLD – programmable logic device) находят широкое применение. Эти схемы предоставляют возможность настраивать функции логических элементов и их взаимосвязи, что делает их чрезвычайно гибкими для различных применений. Логическая емкость ПЛИС является ключевой характеристикой, которая определяет сложность синтеза цифровых устройств. Этот показатель отражает количество функциональных блоков, которые могут быть размещены

на кристалле, что напрямую влияет на возможности и производительность схемы [1].

FPGA (Field-Programmable Gate Array) представляют собой реконфигурируемые микросхемы, состоящие из множества отдельных логических блоков. Каждый логический блок включает триггер и таблицу поиска (LUT – Look Up Table), что обеспечивает гибкость и мощность для реализации различных цифровых функций. Ранее были разработаны самосинхронные логические элементы LUT и DC LUT [2], а также логические элементы, реализующие системы функций [3–6]. Эти разработки показали значительный потенциал для улучшения эффективности и производительности цифровых схем. В статье также предложено адаптировать известный элемент LUT к принципам работы самосинхронизирующихся схем [7].

Такой подход может открыть новые возможности для применения FPGA в более сложных и требовательных цифровых системах, обеспечивая высокую скорость и надежность работы.

Цель исследования. Цель работы состоит в том, чтобы провести исследование и сравнение характеристик при моделировании схем на топологическом уровне: T – уменьшающий задержку на вычисление логической функции; L – сложность в количестве транзисторов; S – площадь занимаемая на кристалле; W – потребляемая мощность.

Топологическое моделирование. Топологическое размещение (Floorplanning) – это вид построения схем, при котором производится размещение модулей, блоков, логических элементов, соединительных слоев и их соединения на площади кристалла.

Для этих целей подходит программное обеспечение MICROWIND, которое позволяет провести моделирование, симуляцию и проектирование интегральных схем на уровне топологический уровень (физического описания) [8]. При моделировании и симуляции в работе использовалась технология 32-нм комплементарных технологий металл-оксид-полупроводник (КМОП) – CMOS 32 nm – 8 metal. Время моделирование составляло 50 нс. Сигнальное напряжение было равно 0,35.

Моделирование топологий LUT представлено на рис. 1–10.

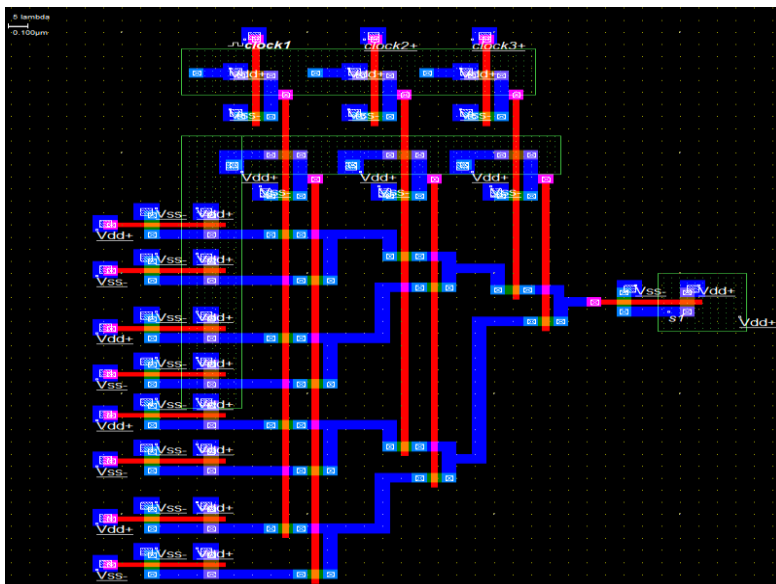


Рис. 1. Топология обычного LUT на 3 переменные

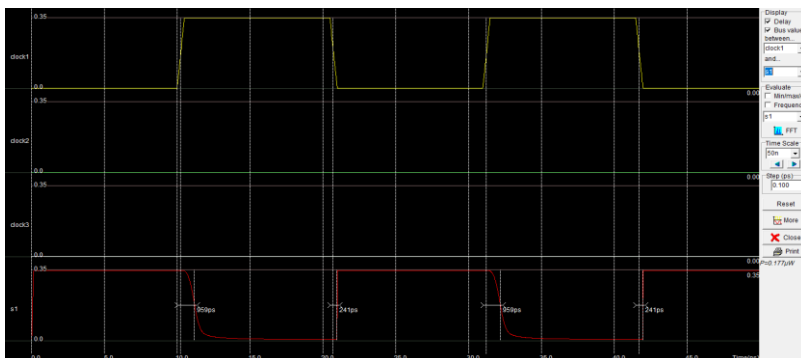


Рис. 2. График моделирования сигнала обычного LUT на 3 переменные

Layout Size	Electrical Properties
Width: 3.5µm (176 lambda)	electrical nodes : 61/3000
Height: 4.6µm (231 lambda)	nMOS devices : 29/2000
Surf: 16.3µm2 (0.0 mm2)	pMOS devices : 15/2000

Рис. 3. Характеристики топологической схемы.
Площадь кристалла и количество транзисторов

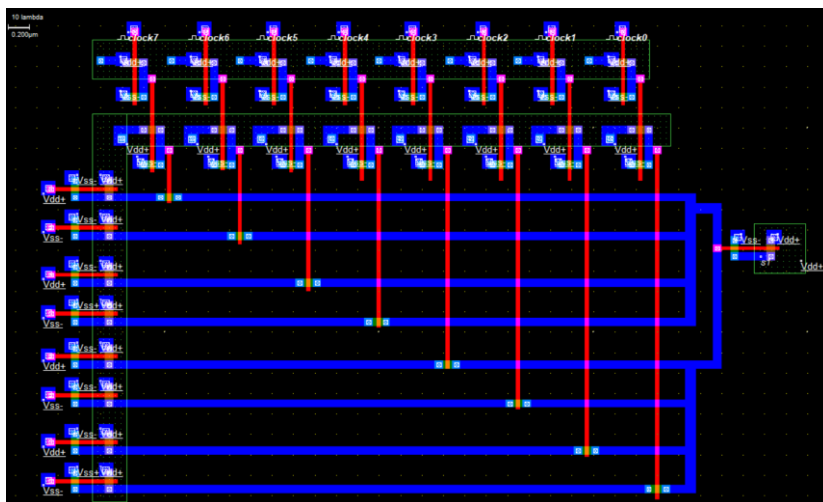


Рис. 4. Топология унитарного LUT на 3 переменные

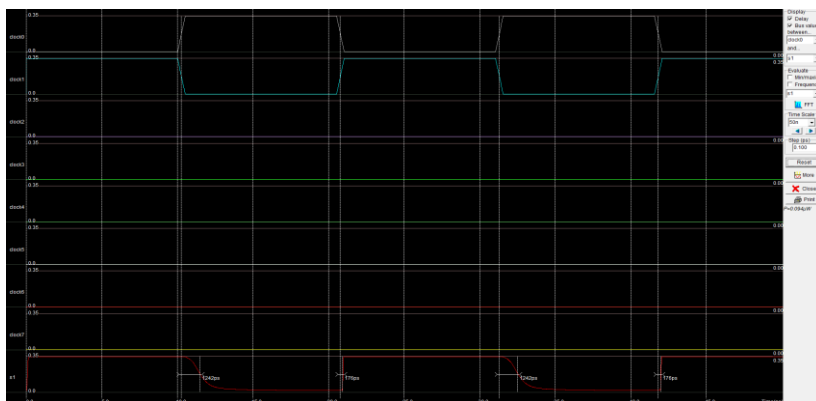


Рис. 5. График моделирования сигнала унитарного LUT на 3 переменные

Layout Size	Electrical Properties
Width: 7.1µm (357 lambda)	electrical nodes : 80/3000
Height: 4.7µm (235 lambda)	nMOS devices : 33/2000
Surf: 33.6µm ² (0.0 mm ²)	pMOS devices : 25/2000

Рис. 6. Характеристики топологической схемы. Площадь кристалла и количество транзисторов

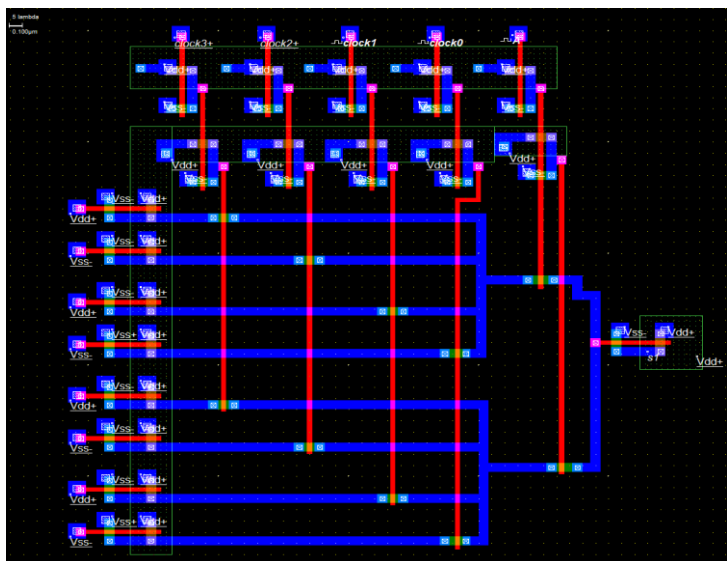


Рис. 7. Топология комбинированного LUT на 3 переменных:
2 унитарные переменные + 1 обычная переменная

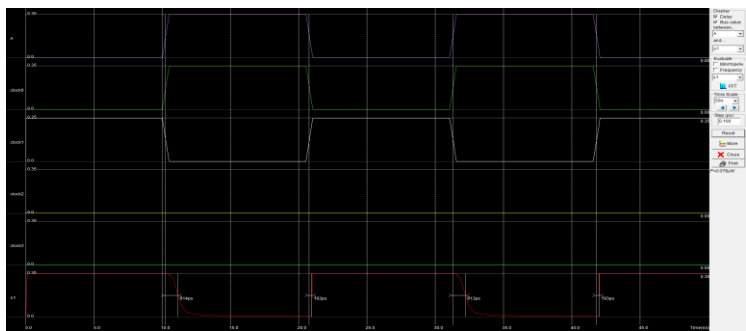


Рис. 8. График моделирования сигнала комбинированного LUT на 3 переменные

Layout Size	Electrical Properties
Width: 4.9µm (244 lambda)	electrical nodes : 67/3000
Height: 4.7µm (237 lambda)	nMOS devices : 29/2000
Surf. 23.1µm2 (0.0 mm2)	pMOS devices : 19/2000

Рис. 9. Характеристики топологической схемы.
Площадь кристалла и количество транзисторов

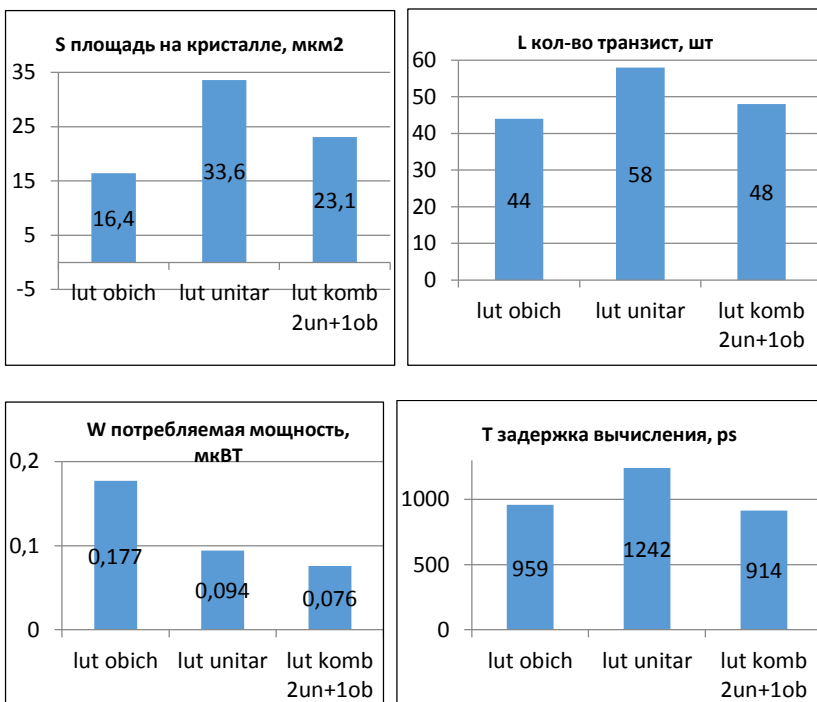


Рис. 10. Сравнение результатов

По полученным данным можно сделать несколько выводов:

1. Комбинированная схема выигрывает по потребляемой мощности, задержке вычислений и имеет среднее значение занимаемой площади на кристалле.
2. Унитарный LUT имеет наибольшее значение по задержке вычислений и занимаемой площади на кристалле.
3. Обычный LUT имеет меньшую площадь, но самую большую потребляемую мощность.

Заключение. Было проведено моделирование топологии LUT элементов ПЛИС с обычным (позиционным), унитарным и комбинированным кодированием входных переменных. Проведено сравнение основных характеристик. Текущая выполненная реализация позволяет предположить работоспособность LUT элемента ПЛИС с комбинированным кодированием входных переменных. В следующих исследованиях и работах планируется провести оптимизацию использования площади и потребления мощности разных конфигураций.

Библиографический список

1. Архитектура ПЛИС. Ч. 1. Логический элемент [Электронный ресурс]. – URL: http://www.labfor.ru/articles/fpga_arch_le (дата обращения: 21.05.24).
2. Checkability Important for Fail-Safety of FPGA-based Components in Critical Systems // Intelligent Information Technologies & Systems of Information Security (IntelITSIS) 2021: International, 24–26 March 2021: proceedings. – Khmelnytskyi, Ukraine: IEEE, 2021. – P. 471–480.
3. Hidden fault analysis of FPGA projects for critical applications / O. Drozd., I. Perebeinos, O. Martynyuk., K. Zashcholkin, O. Ivanova, M. Drozd // IEEE International Conference TCSET. Paper 142, Lviv-Slavsko, Ukraine, 2020. DOI: 10.1109/TCSET49122.2020.235591
4. Skornyakova A.Y., Vikhorev R.V. Self-Timed LUT Layout Simulation // Conference of Russian Young Researchers in Electrical and Electronic Engineering. – IEEE, 2020. – P. 176–179. DOI: 10.1109/EIConRus49466.2020.9039374
5. Vikhorev R. Universal logic cells to implement systems functions // Conference of Russian Young Researchers in Electrical and Electronic Engineering. – IEEE, 2016. – P. 404–406. DOI: 10.1109/EIConRusNW.2016.7448197
6. Vikhorev R. Improved FPGA logic elements and their simulation // Conference of Russian Young Researchers in Electrical and Electronic Engineering. – IEEE, 2018. – P. 275–280. DOI: 10.1109/EIConRus.2018.8317080
7. Tyurin S.F. LUT's Sliding Backup // IEEE Transactions on device and materials reliability. – Mart 2019. – Vol. 19, iss. 1. – P. 221–225. DOI: 10.1109/TDMR.2019.2898724

Сведения об авторах

Васенин Иван Андреевич – аспирант кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: Vasenin.Ioann@yandex.ru

Тюрин Сергей Феофентович – доктор технических наук, профессор кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: tyurinsergfe@yandex.ru

СОДЕРЖАНИЕ

Секция I. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ	3
Рябов Н.А., Петренко А.А. Разработка графического редактора космических тел для двумерных игр	4
Карелов В.А., Кузнецов Д.Б. Простой алгоритм распределения графиков по нескольким осям	9
Павлова Н.А., Федоров А.Б. Методы обработки показаний датчиков: сравнение и анализ фильтров	14
Касимова З.Г., Кушев В.О. Разработка сервисов для повышения открытости среды в организации	19
Болотов М.В., Погудин А.Л. Разработка решения задачи составления композиции на программном уровне	23
Погудин А.Л., Хохрякова Е.А. Разработка колесного узла для мобильной робототехнической платформы на основе анализа конструкторских особенностей колесных пар	30
Хисеу Ву Динь, Файзрахманов Р.А. Математические и алгоритмические основы информационной системы мониторинга и управления техническим обслуживанием оборудования с использованием модифицированной модели индекса технического состояния	36
Иванов К.А., Рошин Е.В. Применение технологии «интернет вещей» в энергетике.....	42
Вольский А.В., Файзрахманов Р.А. Моделирование поведения логико-динамических объектов для компьютерного тренажера портального крана	47
Полевщиков И.С., Уколов М.С., Дмитрук М.В., Кравченко Р.А. Разработка и применение компьютерных тренажерных комплексов на основе веб-технологий при изучении студентами математических методов	53
Полякова О.А., Степанова Я.А. Автоматизация задачи учёта и анализа использования бонусной программы для клиентов транспортной компании	61
Чураков Г.А. Частеречная разметка для выделения скелетной структуры предложений	67

Корепанова Е.В., Курушин Д.С., Петренко А.А. Программа нахождения ориентации инструмента трехмерной печати из разных плоскостей в текущий момент времени	73
Вычугжанин М.В. Информационные технологии в области мобильного здравоохранения	78
Советов С.И., Тюрин С.Ф. Алгоритм соединения блоков функции дешифрации для синтеза логического элемента LUT с дешифрацией входных переменных	82
Пономарев А.Е., Кривцун А.В. Информационная система управления проектами молодежного движения	90
Ананина А.Ю., Долгова Е.В. Численные исследования прямой задачи кинематики робота-манипулятора	96
Камбалина А.Д. Разработка программного обеспечения для оценки тремора у пациентов с болезнью Паркинсона	100
Масылюк Е.С., Архипов А.В. Разработка модуля инструментального анализа многомерных данных (кубов)	106
Никулин В.С., Орлов Т.С. Сервер FTP с возможностью доступа к файлам посредством стандарта USB	110
Юрченко С.А. Разработка сервиса подбора лекарственных препаратов	116
Постных Д.В., Никулин В.С. Разработка подхода к подбору радиусов дисперсий в радиально-базисных сетях	121
Копейкин Д.Е., Петренко А.А. Разработка программного обеспечения для автоматизации процесса передачи картографических слоев на карту	128
Шеретов М.А., Яруллин Д.В. Современные методы анализа и классификации клеток крови: необходимость интерпретируемых алгоритмов	133
Самойлова Е.М., Шашаткин А.Д. Программная реализация обеспечения связи по протоколу Modbus для GD32F105HV	140
Ушаков А.С., Мухин О.И. Разработка архитектуры автономной многоагентной системы беспилотных летательных аппаратов для поиска заданных объектов на местности.....	147

Индюков Г.С. Разработка программного модуля интеллектуальной системы автоматизации для детектирования активности оборудования на основе алгоритмов нейронных сетей	155
Полякова О.А., Пантелеев А.А. Разработка программного обеспечения автоматизированной системы оповещения МЧС о чрезвычайных ситуациях	161
Усманов А.К., Карлов Д.А. Разработка проекта робототехнического комплекса для движения по труднопроходимой местности	166
Богатырев Р.С., Белослудцев Ю.С., Белоусов М.И., Жигалов А.В., Погудин А.Л. Разработка кинематики лабораторного стенда 3D-принтера на базе Arduino	171
Жигалов А.В., Белоусов М.И., Богатырев Р.С., Белослудцев Ю.С., Погудин А.Л. Разработка программного модуля лабораторного стенда 3D-принтера на базе Ардуино	177
Жогаштиев Н.Т., Анисимов В.С. Управление автоматизации с использованием микроконтроллера ESP8266 и Blynk-приложения	185
Тарасова А.К., Долгова Е.В. Разработка механизма движения стрелы портального крана в среде Unreal Engine 5	191
Погудин А.Л., Белоусов М.И., Жигалов А.В., Богатырев Р.С., Белослудцев Ю.С. Разработка аппаратного модуля лабораторного стенда 3D-принтера на базе Ардуино	197
Полякова О.А., Деменев А.С. Проектирование модуля управления информационным контентом в мобильном приложении	204
Белослудцев Ю.С., Белоусов М.И., Богатырев Р.С., Жигалов А.В., Погудин А.Л. Основание выбора технологии печати лабораторного стенда 3D-принтера на базе Arduino	209
Камянецкий С.А., Келлер С.И., Тронина Д.О., Погудин А.Л. Разработка имитационной модели лабораторного стенда по исследованию базовых логических элементов	216

Шестаков К.Н., Кушев В.О. Анализ потребностей пользователей при поиске напарников для многопользовательских видеоигр	224
Колобов Б.К., Карлов Д.А. Учебная программа для работы с моделью робота-манипулятора в системе «Матлаб»	229
Перваков С.И., Булатов П.А., Погудин А.Л. Разработка лабораторного стенда «беспилотный летательный аппарат». 3D-проектирование и моделирование	244
Булатов П.А., Перваков С.И., Погудин А.Л. Разработка лабораторного стенда «беспилотный летательный аппарат». Схемотехническое проектирование и моделирование	250
Кушев В.О., Бродников В.С. Разработка приложения для регистрации продаж в сети кофеен, автоматизирующего реализацию скидок/акций/программ лояльности	255
Масич Г.Ф., Батюков М.Г. Установка и администрирование операционной системы Linux по сети	261
Кушев В.О., Зянкина Е.Г. Исследование функциональных требований к системе автоматизации учета материально-технических средств кафедры университета	268
Курушин Д.С., Пономарев Е.В., Шепелев В.М. Использование технологий компьютерного зрения для обеспечения безопасной работы за фрезерно-сверлильным станком	274
Келлер С.И., Камянецкий С.А., Тронина Д.О., Погудин А.Л. Конструкторские и технологические основы разработки трехмерного дисплея	276
Корзникова А.И., Федоров А.Б. Система автоматизированного сбора показаний индивидуальных приборов учета коммунальных услуг в жилищно-коммунальном хозяйстве по технологии интернета вещей	282
Исламов И.И., Яруллин Д.В. Разработка модели и алгоритма автоматизированного формирования тестовых материалов	288
Русакова А.М., Карлов Д.А. Разработка программно-аппаратного комплекса для автоматического поиска дефектов деталей авиадвигателя с использованием компьютерного зрения	293

Чирков В.Н., Петренко А.А. Разработка интернет-магазина для системы общественного питания ИП Першин	299
Петренко А.А., Матвеев Н.А. Разработка информационной системы для проведения турниров по спортивному программированию	306
Тренина Д.О., Келлер С.И., Камянецкий С.А., Погудин А.Л. Разработка лабораторного стенда по исследованию систем автоматического управления и регулирования на базе программируемых логических контроллеров	310
Фёдоров А.Б., Хлопин Н.С. Разработка алгоритмов для тестера витой пары на базе Arduino	317
Исмаилов А.Р., Долгова Е.В. Моделирование тросов для тренажера портального крана в среде Unreal Engine 5	326
Лыкова А.В., Федоров А.Б. Исследование и тестирование системы рычагов на джойстиках для виртуального тренажера оператора портального крана	331
Викентьева О.Л., Брейкин А.Д. Использование системы межпроцессного взаимодействия для запуска приложений в изолированной среде	338
Долгова Е.В., Кузнецов Н.Д. Оценка производительности детекционных моделей Yolo для задач распознавания цифр	343
Полякова О.А., Фогельгезанг А.А. Разработка модуля мобильного приложения для проведения приемки недвижимости	351
Викентьева О.Л., Пономарёв А.В. Автоматизация процесса установки программного обеспечения и дополнительных компонентов	357
Викентьева О.Л., Катаев М.М. Автоматизация складских процессов с помощью мобильных устройств	364
Викентьева О.Л., Бурмантов Е.П. Интеграция программного обеспечения с ERP-системой предприятия	371
Радостев М.Н., Тарутин А.В. Интеграция систем кадрового электронного документооборота в HRM-системы	378
Халилов А.А., Кузнецов Д.Б. Реализация механизма выбора ссылки в интерфейсе системы управления нормативно-справочной информацией	384

Мехоношин В.А., Кузнецов Д.Б. Разработка системы оповещения сотрудников компании	389
Томилов В.А., Петренко А.А. Разработка интернет-магазина компьютерной техники ООО «Капитан»	395
Циреньщиков А.А., Долгова Е.В. Моделирование стропальщика для тренажера портального крана в среде Unreal Engine 5	403
Азмагулов А.В., Кузнецов Д.Б. Разработка программного обеспечения «шахматный тренажер»	408
Мухамадыев А.Н., Кузнецов Д.Б. Реализация механизма сбора метрик с серверов в веб-сервисе для нагрузочного тестирования сайтов	413
Костылев К.Е., Курушин Д.С. Информационная система мониторинга, поиска, детектирования и нанесения на карту объектов специального назначения	419
Газеев Е.Д. Применение машинного зрения в сельском хозяйстве	426
Онищук Д.Е. Программный модуль для информационной системы подбора нефтяного оборудования АО «Новомет-Пермь».....	432
Яруллин Д.В., Углицких М.С., Морохин А.О., Мазяров А.В. Языковые модели и методы преобразования речи в текст	440
Ахматнурова О.Ф., Кузнецов Д.Б. Разработка View для процесса передачи картографических слоев на карту	447
Чувашев М.А., Долгова Е.В. Разработка модели и алгоритмов автоматизированного сбора и рассылки новостей	452
Викентьева О.Л., Чистяков М.В. Реализация авторизации для мобильных устройств	458
Яруллин Д.В., Истомин Р.В. Корректирование сроков ремонтных работ на предприятии с использованием средств видеоаналитики	463
Сарманаев Р.Р., Яруллин Д.В., Духанин А.В. Интеллектуальный диалоговый агент на основе денотатного подхода и генеративных нейросетевых технологий	469

Мухаметсафин Р.Т., Сторожев С.А. Практическое применение нечеткой логики для навигации мобильного робота в неизвестном пространстве	477
Курушин Д.С., Артемьев Д.А. Разработка автоматизированной системы управления электроэрозионной обработкой деталей	483
Курушин Д.С., Морозова Е.М., Сахарова А.Р. Описание использованных языковых моделей при разработке АРМ «помощник техническому писателю»	489
Тедеев А.З., Викентьева О.Л. Применение машины состояний для разработки приложения для взаимодействия с моделями планирования деятельности агрохолдинга	495
Ларионов А.А., Лапшина Ю.О. Комплексный подход к проверке загружаемых файлов изображений	503
Субботин А.С., Файзрахманов Р.А. Способы обнаружения мошеннических действий в ECOM-транзакциях	511
Ремянников А.В., Яруллин Д.В. Модернизация существующей информационной системы распределения нагрузки на кафедре «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета	517
Якубчик А.В., Погорелый Б.П., Файзрахманов Р.А. Моделирование потоков людей в общественных местах	523
Матяж В.О., Яруллин Д.В. Разработка информационной системы автоматизированного проектирования топологии сети на основе слабоструктурированного описания	528
Курушин Д.С., Назаров Н.Н., Коногорова Л.В. Сопряжение системы машинного зрения с системой управления робота-манипулятора	535
Черных А.Д., Мухин О.И. Разработка механизма мультифильтрации таблиц	539
Масич Г.Ф., Арышев Д.А. Анализ общеизвестных обязательных атрибутов протокола BGP-4 для управления внешней политикой маршрутизации	544
Кузнецов Д.Б., Шадрина Т.В. Разработка математической модели децентрализованного управления порталом сайтов	549
Каплина К.И., Никитина К.А., Гончаровский О.В. Модель системы автоматического управления «умной теплицей»	556
Мухин О.И., Галинов О.Ю. Автоматизация алгоритма создания отчетов	564

Вичугов А.Д., Юзеев Е.В. Проектирование подсистемы расширенного мониторинга заказа на производство с динамической структурой в «1С:ERP Управление предприятием 2»	570
Кузнецов Д.Б., Малякшин Е.П., Вязов А.С. Особенности разработки токенов доступа для кроссплатформенного мобильного приложения	577
Поспелов А.А., Долгова Е.В. Автоматизация стимулирования профессионального роста сотрудников через поощрение достижений	582
Безукладников Ю.А., Плотников П.А., Шенин Я.Д., Наборщиков А.А. Исследование временных характеристик математической модели и макета самомаршрутизирующегося аналого-цифрового преобразователя на базе нейронной сети	588
Масич Г.Ф., Добрынин Н.А. Методы организации отказоустойчивого подключения корпоративной сети к сети Интернет	595
Няшин М.М., Петренко А.А. Решение проблемы добавления пользовательских модулей в готовое программное обеспечение	602
Князев В.И., Князев А.И. Разработка сверточной нейронной сети для классификации действий человека по кадру с камеры	607
Васенин И.А., Тюрин С.Ф. Топологическое моделирование логических элементов LUT, использующих позиционный код набора переменных и LUT с унитарным кодированием	614

Научное издание

**АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ
УПРАВЛЕНИЯ И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ**

Материалы всероссийской научно-технической
конференции
(г. Пермь, 7–9 июня 2024 г.)

В двух томах
Том I

Подписано в печать 24.12.2024. Формат 60×90/16.
Усл. печ. л. 39,3. Тираж 30 экз. Заказ 240.

Издательство
Пермского национального исследовательского
политехнического университета.
Адрес: 614990, г. Пермь, Комсомольский пр., 29
Отпечатано в типографии издательства Пермского
национального исследовательского
политехнического университета.
Адрес: 614990, г. Пермь, Комсомольский пр., 29