

КУЧУКОВ ВИКТОР АНДРЕЕВИЧ

РАЗРАБОТКА МЕТОДОВ И ПРОГРАММНЫХ СРЕДСТВ ПОВЫШЕНИЯ  
ПРОИЗВОДИТЕЛЬНОСТИ ОТКАЗОУСТОЙЧИВЫХ ВЫЧИСЛИТЕЛЬНЫХ  
СИСТЕМ, РАБОТАЮЩИХ В МОДУЛЯРНОМ КОДЕ

Специальность: 2.3.5 – Математическое и программное обеспечение  
вычислительных систем, комплексов и компьютерных сетей (технические науки)

Автореферат  
диссертации на соискание ученой степени  
кандидата технических наук

Москва – 2024

Работа выполнена на кафедре вычислительной математики и кибернетики факультета математики и компьютерных наук имени профессора Н.И. Червякова федерального государственного автономного образовательного учреждения высшего образования «Северо-Кавказский федеральный университет» (ФГАОУ ВО СКФУ).

**Научный руководитель:**

**Бабенко Михаил Григорьевич**

доктор физико-математических наук, доцент,  
заведующий кафедрой вычислительной математики и  
кибернетики факультета математики и компьютерных  
наук имени профессора Н.И. Червякова  
ФГАОУ ВО СКФУ, г. Ставрополь

**Официальные оппоненты:**

**Соловьев Роман Александрович,**

доктор технических наук, член-корреспондент РАН,  
заместитель генерального директора по инновационной  
деятельности ООО «АльфаЧип»

**Феокистов Александр Геннадьевич,**

доктор технических наук, доцент, главный научный  
сотрудник лаборатории параллельных и распределенных  
вычислительных систем, Федеральное государственное  
бюджетное учреждение науки Институт динамики систем  
и теории управления имени В.М. Матросова Сибирского  
отделения Российской академии наук

**Ведущая организация:**

Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Южный федеральный университет»

Защита состоится «13» июня 2024 года в 15 ч. на заседании диссертационного совета 24.1.120.01 при Федеральном государственном бюджетном учреждении науки Институте системного программирования им. В.П. Иванникова Российской Академии Наук по адресу: 109004, г. Москва, ул. А. Солженицына, д. 25.

С диссертацией можно ознакомиться в библиотеке и на сайте Федерального государственного бюджетного учреждения науки «Институт системного программирования им. В.П. Иванникова Российской Академии Наук».

Автореферат разослан «\_\_» \_\_\_\_\_ 2024 г.

Ученый секретарь

диссертационного совета 24.1.120.01,  
кандидат физико-математических наук

Зеленов С.В.

## Общая характеристика работы

**Актуальность работы.** С момента появления первых вычислительных машин актуальной является проблема их надежной работы и гарантированной достоверности получаемых результатов. Усложнение вычислительных структур, условий их эксплуатации и решаемых задач требует разработки новых подходов к обеспечению достоверности вычислений и отказоустойчивости системы, под которой понимают возможность системы сохранять свою работоспособность после отказа одной или нескольких её составных частей. Таким образом, встает задача поиска путей оптимизации архитектуры вычислительных узлов распределенной среды по различным параметрам, таким как безопасность, скорость работы, отказоустойчивость и т.д. Для разработки отказоустойчивых вычислительных систем и оценки их эффективности необходимо специализированное программное обеспечение, позволяющее адаптировать разрабатываемую систему под конкретные потребности пользователя.

Для обеспечения отказоустойчивости с целью быстрого обнаружения сбоев применяют автоматический контроль ошибок, организация которого может быть решена путем введения избыточности. Все методы введения избыточности можно свести к двум обобщенным группам: структурное резервирование и информационное резервирование. Наиболее распространенными методами первой группы являются дублирование и мажоритарный принцип, а к основным методам второй группы относятся помехоустойчивые коды, арифметические коды, контроль в модулярном коде, который может быть представлен системой остаточных классов (СОК) и полиномиальной системой классов вычетов.

В качестве метода проектирования отказоустойчивых систем обработки информации, обладающих возможностью увеличить скорость вычисления за счет параллельного выполнения операций сложения и умножения, широко используемых в системах распределенной обработки данных, может быть выбрана система остаточных классов. СОК при вычислениях использует не все число, а информацию об его остатках на определенные модули, что позволяет проводить вычисления по независимым каналам с числами небольшой разрядности, за счет чего может быть достигнуто увеличение скорости вычислений. Введение избыточного основания позволяет обнаруживать ошибки вычислений. Введение ограничений на избыточное основание или введение дополнительных оснований позволяет не только обнаруживать, но и исправлять ошибки. Однако для широкого применения СОК необходима разработка методов создания программ и программных систем для реализации вычислительных узлов систем параллельной и распределенной обработки данных.

Разработка специализированных вычислительных систем, работающих в СОК, началась в 50-60-е годы в СССР, сейчас же система остаточных классов нашла применение в разработках Cisco Technology (патент US 7,027,598), Toshiba, Samsung Electronics (патент US 7,805,478) и Google Inc. (патент US 8,386,802).

Тем не менее, в СОК существует ряд операций, называемых немодульными, таких как определение знака числа, сравнение чисел и др., реализация которых является вычислительно сложной. Для выполнения данных операций требуется знание позиции числа на числовой прямой. Применение существующих методов выполнения немодульных операций в СОК обладает ограниченными функциональными возможностями и недостаточной скоростью обработки. Поэтому тщательного исследования требует вопрос оптимального выбора параметров немодульных операций и выработки общего метода, позволяющего увеличить скорость выполнения немодульных операций для задач обработки информации.

Значительный научный вклад в рассматриваемую область внесли отечественные и зарубежные исследователи: И.Я. Акушский, Д.И. Юдицкий, В.М. Амербаев, А.И. Галушкин, Н.И. Червяков, А.А. Коляда, А. Omondì и другие.

**Объектом** диссертационного исследования является теория обеспечения надежности данных.

**Предмет исследования** – методы и алгоритмы распределенной обработки данных с использованием модулярной арифметики.

**Целью** диссертационного исследования является повышение скорости работы отказоустойчивых вычислительных узлов обработки информации в распределенных средах за счет оптимизации вычислительно сложных процедур модулярного кода и разработки программных средств их проектирования.

**Научная задача** диссертационной работы состоит в исследовании и разработке математических методов и алгоритмов выполнения немодульных операций на основе различных форм позиционной характеристики числа, способных повысить скорость и отказоустойчивость обработки информации вычислительными узлами распределенной среды, работающими в модулярном коде.

Для решения поставленной общей научной задачи была произведена ее декомпозиция на ряд **частных задач**:

1. Разработка и модификация методов и алгоритмов выполнения вычислительно сложных операций в СОК, таких как перевод из позиционной системы счисления (ПСС) в СОК и из СОК в ПСС, определение знака числа и сравнение чисел для выполнения обработки данных.
2. Модификация методов для повышения отказоустойчивости обработки данных вычислительными узлами распределенной среды, работающими в модулярном коде.

3. Разработка программного комплекса выполнения немодульных операций в модульном коде для проектирования отказоустойчивых вычислительных узлов распределенной среды обработки данных.

**Методология и методы исследования.** Включают использование математического аппарата линейной алгебры, теории чисел, математического анализа, теории алгоритмов, теории надежности, численных методов, математического моделирования, теории вероятностей и математической статистики.

**Соответствие паспорту научной специальности.** Область исследования соответствует паспорту специальности 2.3.5 – Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей по следующим пунктам:

8. Модели и методы создания программ и программных систем для параллельной и распределенной обработки данных, языки и инструментальные средства параллельного программирования.

9. Модели, методы, алгоритмы, облачные технологии и программная инфраструктура организации глобально распределенной обработки данных.

**Научная новизна диссертационной работы:**

1. Модифицированы методы и алгоритмы перевода из позиционной системы счисления в СОК и из СОК в позиционную систему счисления, определения знака и сравнения чисел в СОК, отличающиеся от известных меньшей размерностью операндов и эффективной реализацией операций без необходимости нахождения остатка по большому модулю.
2. Модифицированы методы коррекции ошибок распределенной обработки и хранения информации в системе остаточных классов.
3. Разработан программный комплекс для выполнения немодульных операций вычислительными узлами распределенной среды, позволяющий повысить скорость и отказоустойчивость решения задач распределенной обработки данных.

**На защиту выносятся следующие положения:**

1. Метод перевода из СОК в позиционную систему счисления на основе модифицированной обобщенной позиционной системы счисления (ОПСС).
2. Приближенный метод определения знака и сравнения чисел в СОК на основе Китайской теоремы об остатках (КТО).
3. Алгоритм сравнения чисел и определения знака числа на основе функции ядра Акушского без критических ядер.
4. Метод обнаружения и локализации ошибок, основанный на использовании несбалансированной системы остаточных классов.

**Достоверность** полученных результатов обеспечивается строгостью проведения математических доказательств, при получении которых был использован научно-методический аппарат математического анализа, теории чисел и численных

методов, и подтверждается проведенным сравнительным анализом разработанных методов и алгоритмов с известными ранее с точки зрения скорости обработки данных.

**Практическая значимость** разработанных методов и программных средств заключается в возможности реализации на их основе вычислительных узлов распределенной среды, характеризующихся высокой скоростью и отказоустойчивостью обработки данных, достигаемых за счет модификации алгоритмов модулярной арифметики. Полученные результаты могут быть использованы в специализированных высокопроизводительных цифровых системах обработки информации, таких как распределенные и облачные вычислительные системы, системы автоматизированного контроля, функционирующие в непозиционной системе счисления. Разработанные в рамках диссертационного исследования методы обнаружения и локализации ошибок распределенной обработки и хранения данных и методы перевода из системы остаточных классов в позиционную систему счисления внедрены в организации ООО «Инфоком-С» в системе интеллектуального реагирования на инциденты и события «Darvis».

**Авторский вклад соискателя.** Все изложенные в диссертационной работе результаты получены при непосредственном участии автора. Из результатов работ, выполненных коллективно, в диссертацию включены только полученные непосредственно автором. В работах [1-3, 8, 26] автором рассмотрены проблемы вычисления позиционных характеристик числа и предложена модификация метода вычисления позиционной характеристики на основе обобщенной позиционной системы счисления. В работах [4, 5] автором предложено уточнение разрядности коэффициентов приближенного метода на основе КТО для сравнения чисел. В работах [7, 29] автором предложен алгоритм для построения монотонной функции ядра Акушского без критических ядер для сравнения чисел. В работах [6, 9, 10, 19, 25, 31] автором рассмотрены особенности распределенного хранения и обработки данных с возможностью обнаружения и коррекции ошибок, предложен приближенный метод коррекции одиночной ошибки в СОК с одним надежным контрольным модулем. В работах [25-31] приведен разработанный автором комплекс вычислительных узлов распределенной среды для выполнения модифицированных методов и алгоритмов вычисления немодульных операций в СОК, полученный с использованием программного комплекса для выполнения немодульных операций в СОК, на который получены свидетельства о государственной регистрации программ для ЭВМ.

**Внедрение.** Результаты диссертационной работы были использованы при выполнении проектов: гранта РФ № 19-71-10033 «Эффективная, безопасная и отказоустойчивая система распределенного хранения и обработки конфиденциальных данных с регулируемой избыточностью для проектирования мобильных облаков на маломощных вычислительных устройствах» (глава 2,

параграф 2.1), гранта РФФИ № 20-37-70023 «Разработка методов и алгоритмов быстродействующего, отказоустойчивого математического сопроцессора для проектирования вычислительных систем с повышенным уровнем безопасности и низким энергопотреблением» (глава 2, параграфы 2.3, 2.4), гранта Министерства науки и высшего образования Российской Федерации «Фундаментальные алгоритмы, технологии глубокого обучения и безопасности для облачного хранения и обработки данных» № 075-15-2021-1010 (глава 2, параграф 2.5), стипендии Президента РФ № СП-2236.2018.5 «Разработка мобильного устройства кодирования и передачи видеопотока с низким энергопотреблением в условиях ограничения вычислительной мощности», стипендии Президента РФ № СП-3186.2022.5 «Исследование и разработка методов повышения быстродействия и отказоустойчивости математического сопроцессора для защищенного хранения и обработки цифровой информации на основе модулярной арифметики» (глава 2, параграф 2.2). Работа выполнена в Северо-Кавказском центре математических исследований в рамках соглашения № 075-02-2024-1451 с Министерством науки и высшего образования Российской Федерации.

**Апробация работы.** Основные результаты работы докладывались на международных конференциях IEEE, таких как «2016 IEEE CONFERENCE ON QUALITY MANAGEMENT, TRANSPORT AND INFORMATION SECURITY, INFORMATION TECHNOLOGIES (IT&MQ&IS-2016)» (г. Нальчик, Россия), «2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus-2017)» (г. Санкт-Петербург, Россия), «2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus-2018)» (г. Санкт-Петербург, Россия), «2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW 2019)» (г. Рио-де-Жанейро, Бразилия), «ОТКРЫТАЯ КОНФЕРЕНЦИЯ ИСП РАН ИМ. В.П. ИВАННИКОВА (ISPRAS 2019)» (г. Москва, Россия), «2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus-2020)» (г. Санкт-Петербург, Россия), «Current Problems in Applied Mathematics and Computer Science (CPAMCS-2023)» (г. Ставрополь, Россия).

**Публикации по теме диссертации.** Основные результаты по теме диссертации изложены в 31 публикации, в том числе 8 в рецензируемых научных изданиях, рекомендованных ВАК при Минобрнауки России, 9 – в тезисах докладов, 16 публикаций в изданиях, входящих в международные базы цитирования Web of Science и Scopus, 7 патентах на изобретение. Получено 14 свидетельств о государственной регистрации программ для ЭВМ.

**Свидетельства о государственной регистрации программы для ЭВМ.** «Программа моделирования алгоритмов кодирования и декодирования данных для цифровой обработки сигналов на базе системы остаточных классов» № 2017660854

от 28.09.2017; «Модуль кодирования и декодирования данных в системе остаточных классов» № 2019610808 от 18.01.2019; «Распределенная система надежного хранения и обработки данных в мультиоблачной среде» № 2019611375 от 18.01.2019; «Программа избыточного кодирования и декодирования модулярного кода» № 2020610041 от 09.01.2020; «Программа подготовки файлов для распределенного хранения данных в облаках» № 2020618967 от 10.08.2020; «Программа восстановления полученных данных при распределенном хранении данных в облаках» № 2020619140 от 12.08.2020; «Система моделирования исправления ошибок в модулярном коде» № 2020660257 от 01.09.2020; «Модуль выбора оснований системы остаточных классов для оптимизации минимально избыточного кода» № 2020660531 от 04.09.2020; «Генератор Verilog-модулей перевода из позиционной системы счисления в систему остаточных классов» № 2022667102 от 14.09.2022; «Генератор Verilog-модулей умножителя с накоплением (MAC)» № 2022667010 от 13.09.2022; «Генератор Verilog-модулей перевода из системы остаточных классов в позиционную систему счисления» № 2022667011 от 13.09.2022; «Генератор Verilog-модулей восстановления числа в СОК с модулями специального вида» № 2023668328 от 25.08.2023; «Генератор Verilog-модулей исправления ошибки в избыточной системе остаточных классов» № 2023668861 от 05.09.2023; «Генератор Verilog-модулей исправления ошибки приближенным методом» № 2023669462 от 14.09.2023.

**Моделирование и вычислительный эксперимент** проведены на ASIC в среде RTL и физического синтеза Cadence Genus Synthesis Solution с использованием библиотеки osu018\_stdcells с использованием языков высокого уровня Java и Python для генерации модулей на языке описания аппаратуры Verilog и исследования свойств разработанных методов и алгоритмов.

**Структура диссертации.** Диссертация состоит из введения, трех глав, заключения и пяти приложений. Полный объем диссертации составляет 215 страниц, включая 30 рисунков и 33 таблицы. Список литературы содержит 145 наименований.

### **Основное содержание работы**

**Во введении** обоснована актуальность темы диссертации, сформулированы цель и задачи работы, выбраны объект и предмет исследования, показана научная новизна, практическая и теоретическая ценность полученных результатов, приведены основные положения, выносимые на защиту.

**В первой главе** рассмотрены методы повышения отказоустойчивости вычислительных систем, в частности введение избыточности. Выбор подходящего способа кодирования информации, обладающего необходимой корректирующей способностью, позволяет заметно снизить требования к отказоустойчивости оборудования. Однако проблема состоит не в том, чтобы просто повысить



отказоустойчивость за счет введения очень большой избыточности, а в том, как с помощью по возможности меньшей специальным образом вводимой избыточности достичь нужной степени надежности. Кроме того, для большинства кодов обнаружения и исправления ошибок характерно наличие двух групп цифр – информационной и контрольной. В информационную входят цифры, составляющие числовое значение закодированной величины, а в контрольную – цифры, дополнительно вводимые для целей обнаружения и коррекции возможных искажений, которые являются избыточными. Поскольку по контрольным частям компонентов арифметической операции нет возможности составить контрольную часть результата – исключается возможность контроля правильности выполнения арифметических операций. При этом к арифметическим кодам, у которых информационная и контрольная части кода равноценны, относится система остаточных классов. В системе остаточных классов числа представляются своими остатками от деления на выбранные основания, и все модульные операции могут выполняться параллельно над цифрами каждого разряда в отдельности. Добавление в систему остаточных классов одного избыточного модуля позволяет обнаруживать ошибки вычисления. Введение ограничений на избыточное основание или введение дополнительных оснований позволяет не только обнаруживать, но и исправлять ошибки. Однако в большинстве методы коррекции ошибок обладают высокой алгоритмической сложностью, снижение которой приводит к необходимости исследования и разработки математических методов и алгоритмов выполнения немодульных операций на основе различных форм позиционной характеристики числа, способных повысить скорость и отказоустойчивость обработки информации вычислительными узлами распределенной среды, работающими в модулярном коде.

**Во второй главе** приведены разработанные математические методы и алгоритмы выполнения немодульных операций в СОК. Система остаточных классов, как одна из непозиционных систем счисления, позволяет эффективно выполнять высокопроизводительные вычисления за счет отсутствия переносов между разрядами. Однако ряд операций в СОК, называемых немодульными, является ресурсоемким и требует тщательной оптимизации.

Рассмотрена проблема подбора модулей СОК для перевода чисел из позиционной системы счисления, рассмотрены алгоритмы нахождения остатка при делении на модули специального вида  $2^n$ ,  $2^n - 1$ ,  $2^n + 1$ : нейронная сеть конечного кольца, алгоритмы на основе периода и полупериода числа. Проблемой большинства этих алгоритмов является получение не искомого наименьшего неотрицательного вычета, а сравнимого с ним. Для решения этой проблемы предложен алгоритм, позволяющий повысить производительность перевода из ПСС в СОК за счет эффективного нахождения остатка от деления небольших чисел. Он основан на использовании периода и полупериода чисел:

$$|X|_{2^{n-1}} = \left| \sum_{j=0}^{r-1} 2^{j \cdot n} \cdot B_j \right|_{2^{n-1}} = \left| \sum_{j=0}^{r-1} B_j \right|_{2^{n-1}},$$

$$|X|_{2^{n+1}} = \left| \sum_{j=0}^{r-1} 2^{j \cdot n} \cdot B_j \right|_{2^{n+1}} = \left| \sum_{j=0}^{r-1} (-1)^j B_j \right|_{2^{n+1}},$$

а также алгоритма нахождения остатка числа  $X < 2p$  по модулю  $p$  по формуле:

$$x = \begin{cases} (X + c) \bmod 2^n, & \text{если } (X + c) \geq 2^n \\ X, & \text{если } (X + c) < 2^n \end{cases} \quad (1)$$

где  $c = 2^n - p$ ,  $n = \lceil \log_2 p \rceil$ .

Исследованы методы перевода из СОК в позиционную систему счисления на основе Китайской теоремы об остатках, обобщенной позиционной системе счисления, функции ядра и диагональной функции, для модулей специального вида. На основе ОПСС предложен модифицированный метод перевода из СОК в ПСС и расширения оснований, реализация которого в виде Алгоритма 1 обладает простотой и универсальностью, не зависит от вида модулей и сводится к умножению на треугольную матрицу и делению с остатком [1-3, 8, 26].

---

#### Алгоритм 1. Перевод числа из СОК в ПСС и расширение оснований.

**Вход:**  $X = (x_1, \dots, x_n)$ ,  $\{p_1, p_2, \dots, p_{n-1}, p_n\}$ ,  $p_{n+1}$ .

**Выход:**  $X$ ,  $|X|_{p_{n+1}}$ .

1.  $P_i = \frac{p}{p_i}$ ;  $|P_i^{-1}|_{p_i}$ ;  $B_i = P_i \cdot |P_i^{-1}|_{p_i}$
  2.  $w_1 = 1$ ;  $w_j = \prod_{i=1}^{j-1} p_i$ ,  $j = \overline{2, n}$
  3.  $B_i \xrightarrow{\text{ОПСС}} [b_{i,j}]$ ,  $i, j = \overline{1, n}$ ,  $B_i = \sum_{j=1}^n b_{i,j} \cdot w_j$
  4.  $U_i = \sum_{j=1}^i x_j \cdot b_{j,i}$ ,  $i = \overline{1, n}$
  5.  $\sigma_0 = 0$ ,  $\hat{x}_i = |\sigma_{i-1} + U_i|_{p_i}$ ,  $\sigma_i = \left\lfloor \frac{\sigma_{i-1} + U_i}{p_i} \right\rfloor$
  6.  $X = \sum_{i=1}^n \hat{x}_i \cdot w_i$ ,  $|X|_{p_{n+1}} = |\sum_{i=1}^n \hat{x}_i \cdot w_i|_{p_{n+1}}$
  7. **Возвратить**  $X$ ,  $|X|_{p_{n+1}}$
- 

Рассмотрена проблема сравнения чисел в СОК и определения знака числа. Реализация алгоритма сравнения чисел в СОК состоит из двух этапов. Первый этап – вычисление позиционной характеристики (ПХ) модулярных чисел  $X = (x_1, \dots, x_n)$  и  $Y = (y_1, \dots, y_n)$ . Второй этап – сравнение позиционных характеристик ПХ( $X$ ) и ПХ( $Y$ ) модулярных чисел в позиционной системе счисления [4, 5, 7, 28-30].

Произведена модификация приближенного метода на основе КТО для сравнения чисел в СОК, для чего в качестве позиционной характеристики рассмотрена следующая функция:

$$f(X) = \left| \sum_{i=1}^n \bar{k}_i x_i \right|_{2^N}, \quad (2)$$

где  $\bar{k}_i = \left\lfloor \frac{2^N |P^{-1}|_{p_i}}{p_i} \right\rfloor$ .

Для случая сравнение чисел в СОК с нечетным диапазоном справедливо утверждение, что если  $N = \lceil \log_2(-n + n \cdot P) \rceil$ , то функция  $f(X)$  – строго возрастающая [4].

Данное утверждение позволяет уточнить оценку необходимой точности вычисления коэффициентов приближенного метода на основе КТО для операций определения знака числа и сравнения чисел в СОК. Из утверждения следует, что введенная функция является строго монотонной, следовательно, ее можно использовать в качестве позиционной характеристики для сравнения чисел в СОК.

Рассмотрен случай сравнения чисел в СОК, когда один из модулей равен степени двойки. Так как модули СОК являются взаимно простыми числами, следовательно, четный модуль только один. Значит, без потери общности будем считать, что  $n$ -ый модуль имеет следующий вид  $p_n = 2^t$ . Так как  $p_n = 2^t$ , то используя свойство СОК, числа  $X, Y$  могут быть представлены в следующем виде:

$$X = A \cdot 2^t + x_n, Y = B \cdot 2^t + y_n.$$

Так как  $n$ -ый модуль СОК четный, следовательно, модули  $p_1, p_2, \dots, p_{n-1}$  являются нечетными числами, тогда  $P_n$  – нечетное число. Коэффициенты  $A$  и  $B$  удовлетворяют неравенствам:  $0 \leq A < P_n$  и  $0 \leq B < P_n$ . Вычислив значения  $A$  и  $B$  в СОК по модулям  $p_1, p_2, \dots, p_{n-1}$ , мы можем сравнить их.

Предложенный метод позволяет уменьшить вычислительную сложность алгоритма сравнения чисел в СОК. Эффективная реализация операции  $|X|_{2^N}$  позволяет уменьшить сложность программной реализации нахождения остатка от деления.

Также введена модификация функции ядра Акушского с заданными свойствами, а именно монотонностью, разработан алгоритм подбора параметров для функции ядра без критических ядер (Алгоритм 2), на основе которой построен алгоритм сравнения чисел и определения знака числа, который сводится к сравнению значений функции ядра и в случае их равенства к сравнению одного из остатков [7, 29].

---

**Алгоритм 2 Подбор параметров функции ядра специального вида для заданного набора модулей.**
**Вход:**  $\{p_1, p_2, \dots, p_n\}$ ,

**Выход:**  $w_1, w_2, \dots, w_n$ .

1.  $N = \lceil \log_2 P_n \rceil$
  2.  $w_i^* = \lfloor R(N) \cdot P_i^{-1} \rfloor_{p_i}$ , для  $i = 1, 2, \dots, n$ ,  $C_P^* = P \cdot \sum_{i=1}^n \frac{w_i^*}{p_i}$ , где  $R(N) = 2^N$ .
  3.  $Q = \frac{R(N) - C_P^*}{P}$ .
  4. **Если**  $Q < 0$ , **то**
    - 4.1.  $N = N + 1$  и перейти к строке 2
  5. **Иначе**
    - 5.1. перейти к строке 6
  6. Подобрать  $q_i$  так, чтобы  $Q = q_1 + q_2 + \dots + q_n$ .
  7.  $w_i = q_i \cdot p_i + w_i^*$  для  $i = 1, 2, \dots, n$
  8. **Если**  $C(p_k) = \sum_{i=1}^k w_i \lfloor \frac{p_k}{p_i} \rfloor \geq 0$  И  $\sum_{i=1}^n \left( \lfloor \frac{p_k}{p_i} \rfloor + 1 \right) w_i - w_k > 0$  для всех  $k = 1, 2, \dots, n$ , **то**
    - 8.1. **Возвратить**  $w_1, w_2, \dots, w_n$
  9. **Иначе**
    - 9.1.  $N = N + 1$  и перейти к строке 2
- 

При этом функция ядра находится по формуле

$$C(X) = \left| \sum_{i=1}^n C(B_i) \cdot x_i \right|_{C_P},$$

где  $C(B_i) = \frac{B_i C_P}{P} - \frac{w_i}{p_i}$  и  $B_i = P_i \cdot \lfloor P_i^{-1} \rfloor_P$ .

Рассмотрена реализация умножения с накоплением (МАС), которое является одной из основных операций цифровой обработки сигналов и нейронных сетей, в системе остаточных классов, в которой нахождение остатка от деления осуществляется с использованием периода и полупериода числа и формулы (1) [23].

Для коррекции одиночной ошибки рассмотрен метод проекций на основе Китайской теоремы об остатках с дробными значениями. Проекцией  $X_i$  числа  $X = (x_1, x_2, \dots, x_{n+1})$  по основанию  $p_i$  будет число, полученное вычеркиванием цифры  $x_i$  в представлении  $X$ . Если в упорядоченной системе остаточных классов проекция  $X_i$  числа  $X = (x_1, x_2, \dots, x_i, \dots, x_n, x_{n+1})$  по основанию  $p_i$  удовлетворяет условию  $X_i > \frac{\bar{P}}{p_{n+1}}$ , то цифра  $x_i$  правильная, если возможна лишь одиночная ошибка. Предложена схема обнаружения, локализации и исправления ошибок в СОК с двумя избыточными модулями на базе приближенного метода [25]. Для системы остаточных классов с

одним контрольным основанием  $p_1 < p_2 < p_3 < \dots < p_n < p_{n+1}$ , в которой контрольное основание надежное и не может содержать ошибки, введен метод обнаружения и локализации ошибок, основанный на использовании несбалансированной системы остаточных классов, в котором нет необходимости вычисления позиционной характеристики для каждой из проекций. Если  $p_{n+1} > p_n \cdot p_{n-1}$ , то введенный Алгоритм 3 для локализации и коррекции ошибки корректен [6, 9, 10].

---

### Алгоритм 3. Коррекция ошибки на основе метода проекций и КТО.

**Вход:**  $X' = (x'_1, x'_2, \dots, x'_n, x'_{n+1}), \{p_1, p_2, \dots, p_n, p_{n+1}\}, P = \prod_{i=1}^n p_i, \bar{P} = p_n \cdot P$

$\bar{P}_i = \bar{P}/p_i, i \in [1, n+1], w_i = \left| \bar{P}_i^{-1} \right|_{p_i}$ .

**Выход:**  $X$

1.  $S = \sum_{i=1}^{n+1} x'_i w_i \bar{P}_i$
  2.  $X = S \bmod \bar{P}$
  3. **Если**  $X < P$ , **то**
    - 3.1. **Возвратить**  $X$
  4. **Иначе**
    - 4.1.  $k = 1$
    - 4.2.  $X = S \bmod \bar{P}_1$
    - 4.3. **До тех пор, пока**  $X > P$  **и**  $k \leq n$  **выполнять**
      - 4.3.1.  $k = k + 1$
      - 4.3.2.  $X = S \bmod \bar{P}_k$
    - 4.4. **Возвратить**  $X$
- 

Таким образом, реализованы все операции, необходимые для построения высокопроизводительной и отказоустойчивой системы обработки данных, работающей в системе остаточных классов.

**В третьей главе** рассмотрена реализация разработанных во второй главе алгоритмов в виде архитектур вычислительных узлов распределенной среды для выполнения арифметических, в том числе немодульных, операций в СОК, предложен алгоритм проектирования отказоустойчивой вычислительной системы. Использование непозиционных систем счисления, таких как система остаточных классов, позволяет выполнять сложение и умножение чисел по параллельным вычислительным каналам без переноса разрядов между каналами, что позволяет повысить скорость выполнения арифметических операций. При этом размеры модулей СОК могут не совпадать с разрядностью современных ЭВМ и для достижения максимальной производительности необходима разработка

специализированных интегральных схем, таких как FPGA и ASIC, которые позволят производить вычисления над числами заданной разрядности.

Работа вычислительной системы, работающей в модулярном коде, в виде структурной схемы показана на рисунке 1.

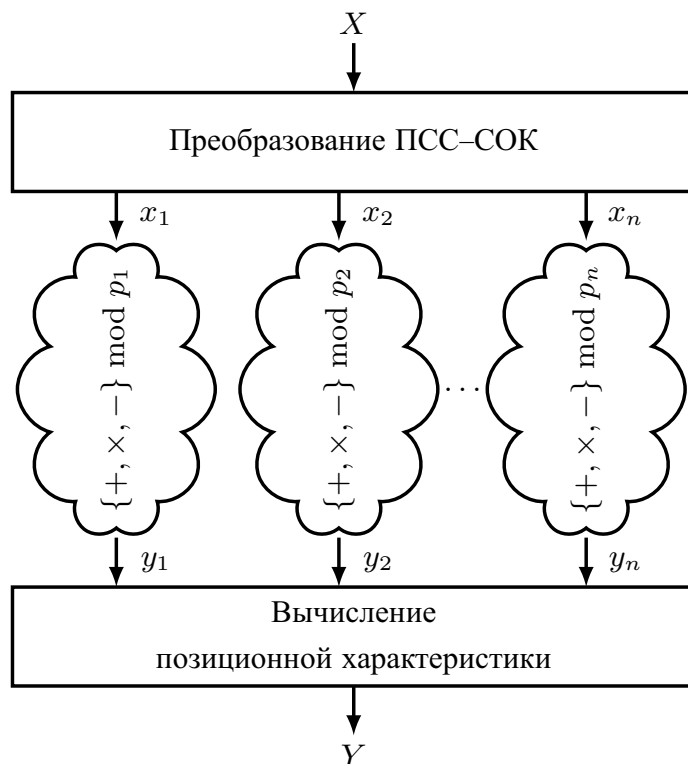


Рис 1. Структурная схема отказоустойчивой вычислительной системы, работающей в модулярном коде

Данная система содержит блоки перевода из позиционной системы счисления в систему остаточных классов, блоки выполнения модульных операций сложения, умножения и вычитания по модулям СОК, блок вычисления позиционной характеристики для выполнения сравнения чисел, определения знака, деления, а также для перевода из системы остаточных классов в позиционную систему счисления.

Первым шагом алгоритма проектирования отказоустойчивой вычислительной системы, работающей в модулярном коде, является выбор набора модулей СОК.

Вторым шагом алгоритма проектирования отказоустойчивой вычислительной системы является выбор метода перевода чисел из позиционной системы счисления в систему остаточных классов.

Третьим шагом алгоритма проектирования отказоустойчивой вычислительной системы, работающей в модулярном коде, является выполнение модульных операций, которые выполняются независимо по каждому из оснований, в частности с использованием распределенной среды и облачных вычислений.

Заключительным шагом алгоритма проектирования отказоустойчивой вычислительной системы является выбор метода вычисления позиционной характеристики числа для целей сравнения чисел, определения знака числа, перевода в позиционную систему счисления, коррекции ошибок вычислений.

Для реализации алгоритма проектирования отказоустойчивой вычислительной системы на основе рассмотренных во второй главе методов и алгоритмов разработан комплекс программ для работы с модулярным кодом, структура которого показана на рисунке 2. В открытом доступе есть библиотека [vscripts.ru](http://vscripts.ru), в которой представлены генераторы Verilog-модулей для реализации базовых операций в системе остаточных классов, однако блоков реализации сравнения чисел и обнаружения и коррекции ошибок там нет. Доступные коды для прямого и обратного преобразования были использованы в качестве аналогов для сравнения, отсутствующие методы были реализованы самостоятельно.

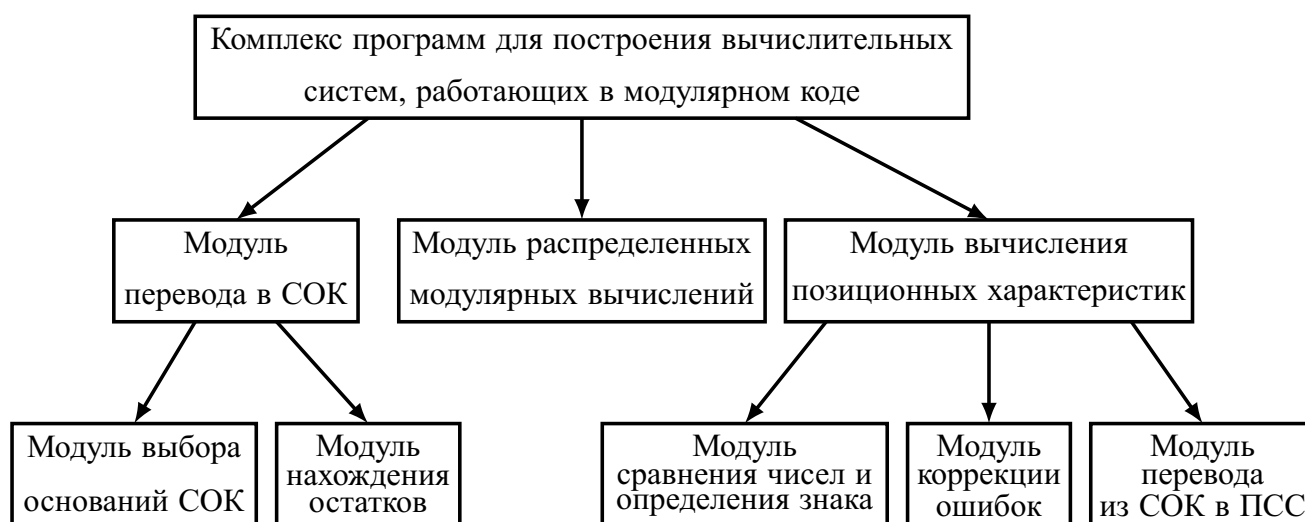


Рис 2. Структура программного комплекса для построения вычислительных систем, работающих в модулярном коде

В соответствии с рисунком 2, первым этапом построения отказоустойчивой вычислительной системы является работа модуля выбора оснований СОК, представленного программой №2020660531, которая позволяет проанализировать структуру динамического диапазона и оптимальность выбора оснований СОК с точки зрения избыточности данных и корректирующей способности кода.

После выбора оснований СОК используется модуль нахождения остатков, представленный комплексом программ №№ 2017660854, 2019610808, 2019611375, 2020610041, 2020618967, 2022667102, в котором реализованы методы периода и полупериода числа, нейронной сети конечного кольца. Программа №2022667102 позволяет генерировать Verilog-модули для моделирования вычислительных узлов на ASIC.

Модуль вычисления позиционных характеристик содержит модуль сравнения чисел и определения знака числа в СОК, модуль перевода из СОК в ПСС, выраженный в программах №№ 2017660854, 2019610808, 2019611375, 2020610041, 2020619140, 2022667011, 2023668328, модуль коррекции ошибок, реализованный комплексом программ №№ 2019611375, 2020610041, 2020619140, 2020660257, 2023668861, 2023669462.

Для оценки эффективности разработанных методов и алгоритмов проведено моделирование вычислительных узлов распределенной среды на ASIC в среде RTL и физического синтеза Cadence Genus Synthesis Solution с использованием библиотеки `osu018_stdcells`. В качестве критериев, позволяющих оценить эффективность разработанных методов и алгоритмов взяты время прохождения сигнала по интегральной схеме (пикосекунды, пс) и используемая площадь интегральной схемы (квадратные микрометры,  $\mu\text{м}^2$ ).

Предложенная модификация для перевода из ПСС в СОК на основе формулы (1) позволила получить наименьшие неотрицательные вычеты, при этом модифицированный метод на основе периода и полупериода числа позволили в среднем на 50% сократить необходимую площадь интегральной схемы, на 23% – время вычислений, по сравнению с нейронной сетью конечного кольца.

Проведено моделирование перевода из СОК в ПСС для метода на основе КТО, приближенного метода на основе КТО, ОПСС и предложенной архитектуры вычислительного узла распределенной среды на основе алгоритма 1, на которую получен патент № 2744815, модулей специального вида. В таблицах 1-2 представлены лучшие результаты моделирования времени и используемой площади для динамических диапазонов от 8 до 64 бит.

В среднем предложенный подход на основе модифицированной ОПСС на 23% быстрее и на 30% компактнее, чем метод КТО. При этом на диапазоне 32-48 бит предложенный подход быстрее на 46% и занимает на 50% меньшую площадь по сравнению с КТО. При сравнении с приближенным методом на основе КТО предложенный метод дает преимущество в среднем на 18% по времени и на 23% по площади только на диапазоне 32-48 бит. В среднем преимущество по сравнению с ОПСС по времени составило 15%, но метод на основе ОПСС занимает на 16% меньшую площадь. Таким образом, предложенный метод на основе модифицированной ОПСС позволяет повысить скорость вычислений и снизить площадь, являясь компромиссным решением между последовательной и компактной ОПСС, и параллельной, но громоздкой КТО.



Таблица 1 – Моделирование времени перевода из СОК в ПСС, пс

Метод	Размерность, бит							
	8	16	24	32	40	48	56	64
КТО	2779	6232	13725	22117	28128	32890	44401	67670
Приближенный метод на основе КТО	4291	8450	9708	17009	18324	17690	19733	25703
ОПСС	4606	10214	14735	17537	20319	18521	27038	33588
Модифицированная ОПСС	3519	7178	11002	12788	16326	14672	28275	48984
Спецмодули $\{2^n - 1, 2^n, 2^n + 1\}$	2898	5546	7990	9383	12907	14162	16311	17514

Таблица 2 – Моделирование площади перевода из СОК в ПСС, мкм<sup>2</sup>

Метод	Размерность, бит							
	8	16	24	32	40	48	56	64
КТО	3075	16551	39430	71282	125940	128123	244818	321253
Приближенный метод на основе КТО	4364	15677	41187	60189	58853	79530	67753	118507
ОПСС	3697	13162	39128	47372	53511	44630	84413	100191
Модифицированная ОПСС	4001	16143	34591	51015	53584	43822	131380	145944
Спецмодули $\{2^n - 1, 2^n, 2^n + 1\}$	2290	4717	7186	8731	11097	13436	15118	17409

Проведено моделирование методов сравнения чисел и определения знака числа. В качестве аналогов для разработанных методов были взяты методы на основе КТО, приближенный метод на основе КТО, ОПСС. Предложен ряд архитектур вычислительных узлов распределенной среды для сравнения и определения знака, на которые получены патенты на изобретения (патенты РФ № 2747371 и 2751992, евразийский патент № 038389).

Архитектура вычислительного узла распределенной среды для сравнения чисел из евразийского патента № 038389 основана на Алгоритме 2 построения функции ядра Акушского с заданными свойствами, сравнение чисел в которой происходит за счет сравнения функция ядра чисел  $C(X)$  и  $C(Y)$ . Схема для сравнения чисел на основе модифицированной приближенного метода на основе КТО использует формулу (2). Результаты моделирования представлены на рисунке 3.

Архитектура вычислительного узла распределенной среды на основе функции ядра быстрее реализации на основе Китайской теоремы об остатках в среднем на 59%, на 15% быстрее приближенного метода на основе КТО, и на 22% быстрее ОПСС. При этом необходимая площадь интегральной схемы на 61% меньше, чем у КТО, на 16% – чем у приближенного метода на основе КТО, но в среднем на 80% больше, чем у ОПСС. Архитектура вычислительного узла, использующего уточненный приближенный метод на основе КТО, в среднем на 58% быстрее КТО, на 10% быстрее оценки приближенного метода на основе КТО, на 14% быстрее ОПСС. При этом уточненный приближенный метод на основе КТО требует в среднем на 59% меньшую

площадь по сравнению с КТО, на 9% меньше известного приближенного метода на основе КТО, но на 100% больше чем у ОПСС.

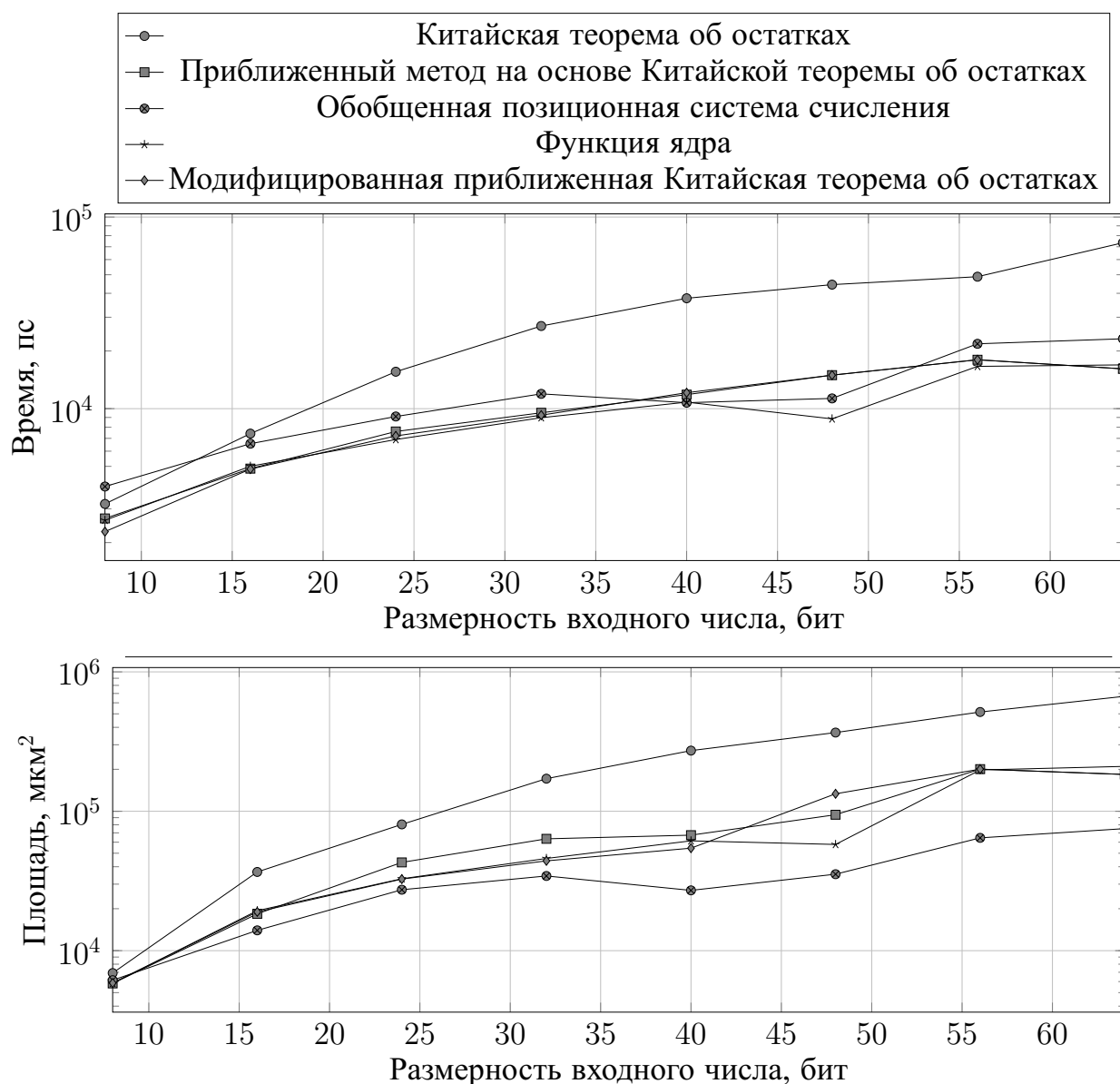


Рис 3. Результаты моделирования времени и площади для сравнения чисел

Приведено моделирование умножения с накоплением для двоичной системы счисления и системы остаточных классов. Реализация в СОК в среднем на 30% быстрее при приблизительно одинаковых занимаемых площадях.

Для обеспечения отказоустойчивости разработана архитектура вычислительного узла распределенной среды для обнаружения и коррекции ошибки модулярного кода (патент РФ №2653257). Особенностью данного подхода является использование памяти для восстановления чисел по каждой проекции.

Одним из возможных применений Алгоритма 3 коррекции ошибки по рабочему основанию с одним избыточным модулем может быть система распределенного хранения данных (патент РФ № 2780148), в которой выполняется подготовка исходных файлов для надежного распределенного хранения посредством перевода в

систему остаточных классов, удовлетворяющую свойству  $p_{n+1} > p_n \cdot p_{n-1}$ , и для восстановления полученных файлов, принятых из распределенной среды в случае ошибки или неполучения одной из частей файла по Алгоритму 3.

Предложенная система может быть реализована архитектурой, представленной на рисунке 4. Распределенное хранилище, показанное на рисунке 4, может быть представлено в виде группы облачных хранилищ, каждое из которых содержит один или несколько остатков. В случае отказа одного из распределенных хранилищ клиентская часть, выделенная пунктирной линией, позволяет исправить ошибку полученных данных.

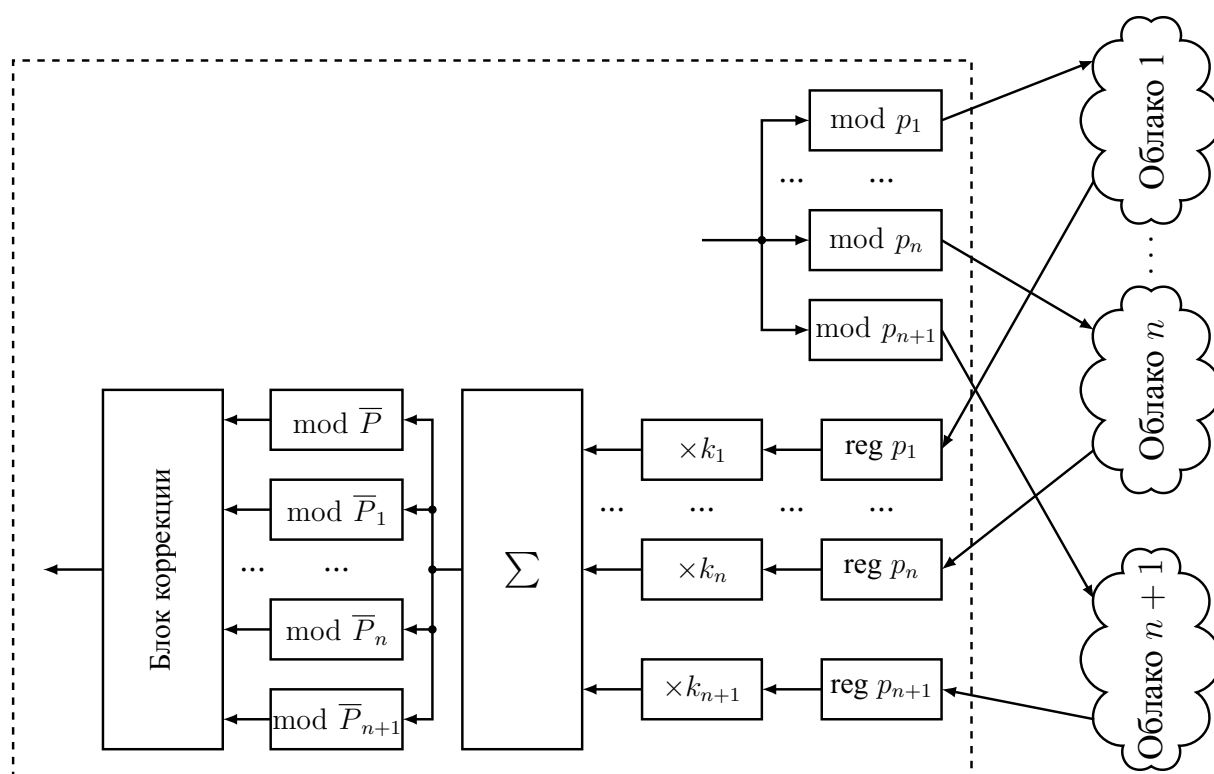


Рис 4. Архитектура системы распределенного хранения данных

Рассмотрено моделирование Алгоритма 3 коррекции одиночной ошибки с одним избыточным модулем. Результаты моделирования представлены в таблицах 3-4 для времени и площади соответственно, где алгоритм с двумя избыточными основаниями обозначен как I, Алгоритм 3 с одним избыточным основанием обозначен как II, приближенный метод на основе КТО с двумя избыточными основаниями, описанный в патенте РФ № 2653257 обозначен как III, приближенный метод на основе КТО с одним избыточным основанием [10] обозначен IV.

Метод на основе Китайской теоремы об остатках в среднем имеет на 52,15% меньшее время вычисления по сравнению с методом с одним избыточным основанием, но имеет на 140% большую площадь. Адаптация приближенного метода с одним избыточным основанием имеет в среднем на 0,43% большее время

вычислений по сравнению с приближенным методом на основе КТО с двумя избыточными основаниями, но на 16,96% меньшую площадь.

Таблица 3 – Моделирование времени коррекции одиночной ошибки, пс

Кол-во раб. модулей	Алгоритм	Покрываемый рабочий диапазон			
		8	16	24	32
4	I	15447	22338	33628	43106
	II	25097	39429	84673	119623
	III	11369	15876	21509	31825
	IV	11476	15707	24436	29211
5	I	13701	18392	29237	38483
	II	22903	36698	76512	111360
	III	12424	15646	23134	27172
	IV	12374	16354	22028	27055
6	I	16409	20503	26721	34146
	II	29077	41193	66569	66711
	III	15867	17340	22252	27402
	IV	15542	18717	21620	26845

Таблица 4 – Моделирование площади коррекции одиночной ошибки, мкм<sup>2</sup>

Кол-во раб. модулей	Алгоритм	Покрываемый рабочий диапазон			
		8	16	24	32
4	I	156978	320430	672019	1093897
	II	60759	132332	319916	538922
	III	186419	393166	816822	1470043
	IV	148011	314708	684626	1206436
5	I	164846	306875	646014	1091506
	II	62136	131767	287343	493900
	III	258575	422888	877680	1448269
	IV	200183	347296	715083	1329741
6	I	280610	356641	657436	1080177
	II	94223	155115	282315	399128
	III	444720	566191	916218	1393492
	IV	343388	473257	783562	1281555

### Основные результаты и выводы по работе

Проведенное исследование показывает, что для уменьшения вычислительной сложности алгоритмов обработки информации можно использовать систему остаточных классов, которая позволяет выполнять параллельно алгоритмы вычисления арифметических операций сложения и умножения чисел, но при этом возникает задача уменьшения вычислительной сложности выполнения следующих операций: перевода из ПСС в СОК, перевода из СОК в ПСС, определения знака числа

и сравнения чисел, обнаружения и исправления ошибок и др. Полученные в диссертационном исследовании результаты позволяют сделать следующие выводы:

1. Предложенный метод для перевода из СОК в ПСС на основе модифицированной обобщенной позиционной системы счисления за счет сокращения количества операций позволяет сократить время вычислений по сравнению с обобщенной позиционной системой счисления и за счет уменьшения размеров операндов сократить используемую площадь по сравнению с приближенным методом на основе Китайской теоремой об остатках.
2. Модифицированный приближенный метод определения знака и сравнения чисел в СОК на основе КТО позволяет повысить скорость вычислений по сравнению с приближенным методом на основе Китайской теоремы об остатках за счет уточнения необходимой точности вычислений коэффициентов приближенного метода и замены операции нахождения остатка от деления взятием младших бит числа.
3. Алгоритм сравнения чисел и определения знака числа на основе функции ядра Акушского без критических ядер позволяет за счет модуля специального вида повысить эффективность и точность целочисленных вычислений.
4. Использование надежного специального избыточного модуля в избыточной СОК с одним контрольным модулем позволяет не только обнаружить ошибку по рабочему модулю, но и локализовать её, при этом снижая требуемую площадь при сохранении скорости вычислений.
5. Комплекс программ выполнения немодульных операций в модулярном коде для проектирования отказоустойчивых вычислительных узлов распределенной среды обработки данных позволяет повысить скорость и отказоустойчивость вычислений.

### **Публикации по теме диссертации**

#### **Статьи автора в журналах, рекомендованных ВАК РФ, Scopus, Web of Science**

1. Кучуков, В.А. Реализация фильтра повышения резкости в системе остаточных классов на FPGA / В.А. Кучуков, Н.И. Червяков // Инфокоммуникационные технологии. – 2015. – Т. 13, №.4. – С. 361-365.
2. Разработка нового нейросетевого метода вычисления модульного умножения в системе остаточных классов / Н.И. Червяков, М.Г. Бабенко, А.Н. Черных [и др.] // Нейрокомпьютеры: разработка, применение. – 2016. – №. 10. – С. 41-48.
3. Исследование эффективных методов перевода чисел из системы остаточных классов в позиционную систему счисления на FPGA / Н.И. Червяков, В.А. Кучуков, Н.Н. Кучеров, Н.Н. Кучукова // Современная наука и инновации. – 2017. – №. 3. – С. 46-52.

4. Эффективное сравнение чисел в системе остаточных классов на основе позиционной характеристики / М.Г. Бабенко, А.Н. Черных, Н.И. Червяков [и др.] // Труды Института системного программирования РАН. – 2019. – Т. 31, № 2. – с.187-202.
5. Positional Characteristics for Efficient Number Comparison over the Homomorphic Encryption / M. Babenko, A. Tchernykh, N. Chervyakov [et al.] // Programming and Computer Software. – 2019. – Т. 45, №. 8. – С. 532-543.
6. Модификация алгоритма обнаружения и локализации ошибки в системе остаточных классов / А. Гладков, В. Кучуков, М. Бабенко [и др.] // Труды Института системного программирования РАН. – 2022. – Т. 34. – №. 3. – С. 75-88.
7. Babenko, M. Improved modular division implementation with the Akushsky core function / M. Babenko, A. Tchernykh, V. Kuchukov // Computation. – 2022. – Т. 10. – № 1. – С. 9
8. Performance Analysis of Hardware Implementations of Reverse Conversion from the Residue Number System / V. Kuchukov, D. Telpukhov, M. Babenko [et al.] // Applied Sciences. – 2022. – Т. 12. – №. 23. – С. 12355.
9. Modified Error Detection and Localization in the Residue Number System / A. Gladkov, V. Kuchukov, M. Babenko [et al.] // Programming and Computer Software. – 2022. – Т. 48. – №. 8. – pp. 598-605.
10. Kuchukov, V. Study of a Redundant Residue Number System for Single Error Correction / V. Kuchukov, M. Babenko, S. Al-Galda // Advances in Systems Science and Applications. – 2023. – Т. 23. – №. 04. – С. 31-39.

#### **Другие публикации автора по теме диссертации**

11. Fast modular multiplication execution in residue number system / N.I. Chervyakov, M.G. Babenko, V.A. Kuchukov [et al.] // 2016 IEEE Conference on Quality Management, Transport and Information Security, Information Technologies (IT&MQ&IS). – IEEE, 2016. – С. 30-32.
12. Новая схема хранения информации в облачной среде на основе системы остаточных классов и схем разделения секрета / Н.И. Червяков, М.Г. Бабенко, Н.Н. Кучеров [и др.] // Современная наука и инновации. – 2017. – № 4 (20). – С. 21-25.
13. Chervyakov, N.I. Research of effective methods of conversion from positional notation to RNS on FPGA / N.I. Chervyakov, M.G. Babenko, V.A. Kuchukov // 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus). – IEEE, 2017. – С. 277-281.
14. Анализ методов обнаружения движения в цифровых системах видеонаблюдения / В.А. Кучуков, М.Г. Бабенко, Е.А. Кучукова, Н.Г. Гудиева // Современная наука и инновации. – 2018. – №3. – с. 8-14.

15. A high-speed residue-to-binary converter based on approximate Chinese Remainder Theorem / N.N. Kucherov, V.A. Kuchukov, N.N. Kuchukova, A.E. Shangina // 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus). – IEEE, 2018. – С. 325-328.
16. AC-RRNS: Anti-collusion secured data sharing scheme for cloud storage / A. Tchernykh, M. Babenko, N. Chervyakov [et al.] // International Journal of Approximate Reasoning. – 2018. – Т. 102. – С. 60-73.
17. Data Reliability and Redundancy Optimization of a Secure Multi-Cloud Storage Under Uncertainty of Errors and Falsifications / A. Tchernykh, M. Babenko, V. Kuchukov [et al.] // 2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). – IEEE, 2019. – С. 565-572.
18. Weighted two-levels secret sharing scheme for multi-clouds data storage with increased reliability / V. Miranda-Lopez, A. Tchernykh, M. Babenko [et al.] // 2019 International Conference on High Performance Computing and Simulation (HPCS). – IEEE, 2019. – С. 915-922.
19. Kuchukov, V. The application of modular arithmetic for matrix calculations/ V. Kuchukov, M. Babenko // 2019 Ivannikov Ispras Open Conference (ISPRAS). – IEEE, 2019. – С. 49-54.
20. Kuchukov, V. Cloud-fog-edge Computing Model for Video Surveillance Based on Modular Arithmetic / V. Kuchukov, A. Nazarov, I. Vashchenko // 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus). – IEEE, 2020. – С. 374-376.
21. Increasing reliability and fault tolerance of a secure distributed cloud storage / N.N. Kucherov, M.G. Babenko, A. Tchernykh [et al.] // ICCS-DE. – 2020. – С. 166-180.
22. Efficient implementation of error correction codes in modular code / N.N. Kucherov, V.A. Kuchukov, E. Golimblevskaia [et al.] // ICCS-DE. – 2021. – pp. 107-118.
23. Кучуков, В.А. Применение системы остаточных классов для повышения эффективности операции умножения с накоплением / В.А. Кучуков, Н.Н. Кучеров // Вестник современных цифровых технологий. 2022. № 12. С. 38-45.
24. An Efficient Method for Comparing Numbers and Determining the Sign of a Number in RNS for Even Ranges / A. Tchernykh, M. Babenko, E. Shiriaev [et al.] // Computation. – 2022. – Т. 10. – №. 2. – С. 17.

#### **Патенты на изобретения**

25. Пат. 2653257 Российская Федерация, Устройство обнаружения и коррекции ошибки модулярного кода / Червяков Н.И., Кучуков В.А., Бабенко М.Г., Кучукова Н.Н.; заявитель и патентообладатель ФГАОУ ВО "Северо-Кавказский федеральный университет". – № 2017126350/08; заявл. 21.07.2017; опубл. 07.05.2018, Бюл. № 13.

26. Пат. 2744815 Российская Федерация, Устройство для перевода чисел из системы остаточных классов и расширения оснований / Бабенко М.Г., Кучуков В.А., Черных А.Н., Кучеров Н.Н.; заявитель и патентообладатель ФГАОУ ВО "Северо-Кавказский федеральный университет". – № 2020120649; заявл. 22.06.2020; опубл. 16.03.2021, Бюл. № 8
27. Пат. 2747371 Российская Федерация, Устройство определения знака числа, представленного в системе остаточных классов / Бабенко М.Г., Кучуков В.А.; заявитель и патентообладатель ФГАОУ ВО "Северо-Кавказский федеральный университет". – № 2020134778; заявл. 22.10.2020; опубл. 04.05.2021, Бюл. № 13
28. Пат. 2751992 Российская Федерация, Устройство сравнения чисел, представленных в системе остаточных классов / Бабенко М.Г., Кучуков В.А.; заявитель и патентообладатель ФГАОУ ВО "Северо-Кавказский федеральный университет". – № 2020134772; заявл. 22.10.2020; опубл. 21.07.2021
29. Евразийский патент на изобретение №038389, Устройство сравнения и определения знака чисел, представленных в системе остаточных классов / Дерябин М.А., Бабенко М.Г., Кучуков В.А., Назаров А.С., Кучеров Н.Н.; заявитель и патентообладатель ФГАОУ ВО "Северо-Кавказский федеральный университет". – № 202090736; заявл. 14.04.2020; опубл. 20.08.2021
30. Пат. 2767450 Российская Федерация, Способ определения знака числа в системе остаточных классов / Бабенко М.Г., Кучуков В.А., Черных А.Н., Кучеров Н.Н.; заявитель и патентообладатель ФГАОУ ВО "Северо-Кавказский федеральный университет". – № 2021108953; заявл. 01.04.2021; опубл. 17.03.2022 Бюл. № 8
31. Пат. 2780148 Российская Федерация, Система распределенного хранения данных / Бабенко М.Г., Кучуков В.А., Кучеров Н.Н., Гладков А.В.; заявитель и патентообладатель ФГАОУ ВО "Северо-Кавказский федеральный университет". – № 2021138986; заявл. 27.12.2021; опубл. 19.09.2022 Бюл. № 26