

На правах рукописи



**Мальчуков Андрей Николаевич**

**АЛГОРИТМИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ  
СИСТЕМЫ ДЛЯ РАЗРАБОТКИ КОДЕКОВ  
ПОМЕХОУСТОЙЧИВЫХ КОДОВ**

Специальность 05.13.11 – «Математическое и программное обеспечение  
вычислительных машин, комплексов и компьютерных сетей»

**АВТОРЕФЕРАТ**  
**диссертации на соискание ученой степени**  
**кандидата технических наук**

Томск – 2008

Работа выполнена на кафедре Вычислительной техники  
ГОУ ВПО «Томский политехнический университет».

Научный руководитель:

доктор технических наук, профессор

**Н.Г. Марков**

Официальные оппоненты:

доктор физико-математических наук, профессор  
кандидат технических наук

**А.Ю. Матросова**  
**П.М. Острась**

Ведущая организация:

Институт вычислительной математики и математической геофизики  
СО РАН, г. Новосибирск.

Защита состоится 24 декабря 2008 г. в 14 ч. 30 мин., в ауд. 214 на заседании совета по защитах докторских и кандидатских диссертаций Д 212.269.06 при Томском политехническом университете по адресу: 634034, г. Томск, ул. Советская, 84, институт «Кибернетический центр» ТПУ.

С диссертацией можно ознакомиться в научно-технической библиотеке Томского политехнического университета по адресу: 634034, г. Томск, ул. Белинского, 55.

Автореферат разослан 21 ноября 2008 г.

Ученый секретарь совета  
по защитах докторских  
и кандидатских диссертаций,  
к.т.н., доцент



**М.А. Сонькин**

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность работы.** В настоящее время в связи с многократно возросшими объёмами передаваемой и сохраняемой информации ужесточились требования к её достоверности. Одним из самых перспективных методов решения проблемы обеспечения достоверности передачи и приёма информации является помехоустойчивое кодирование данных на основе корректирующих кодов. В последнее время коды, исправляющие ошибки, нашли применение во многих системах сбора, передачи и хранения данных. Наиболее известные из них – это сотовые системы связи, спутниковая спасательная система КОСПАС — САРСАТ, система глобальной морской связи Инмарсат, системы автоматизации технологических процессов, сети сбора информации с датчиков различного назначения и др. Во всех вышеприведённых примерах систем с помехозащищённой обработкой информации обычно используются наиболее простые и эффективные циклические корректирующие коды небольшой длины, которые, наряду с простотой кодирования и декодирования, отличаются достаточно большой корректирующей способностью.

Важно учесть, что в ряде случаев, в первую очередь, в системах сбора информации с датчиков возникает ситуация, когда пропускная способность каналов низка (например, передача данных из скважин по гидравлическому или электромагнитному каналам), что ведёт к требованию от помехоустойчивых кодов минимальной избыточности (длины кодового слова), достаточной для исправления заданного класса ошибок. В случае высокоскоростных шин передачи данных на печатных платах параллельным способом лишняя дорожка на дополнительный разряд избыточности привносит дополнительные затраты и наводки в схему, поэтому также необходимо использовать коды с минимальной избыточностью.

До сегодняшнего дня многие известные коды (Хэмминга, Рида-Маллера, Боуза-Чоудхури-Хоквенгема и т.п.) применяют не системно, нет методики выбора кодов, обеспечивающих требуемые корректирующие свойства при минимальной избыточности в заданном диапазоне информационных разрядов. Успешному применению помехоустойчивых кодов разработчиками систем сбора и передачи данных препятствует тот факт, что для непосредственного выбора помехоустойчивого кода под конкретные условия эксплуатации систем требуются глубокие знания теории помехоустойчивого кодирования.

Сегодня не известны автоматизированные системы, позволяющие разрабатывать кодеки помехоустойчивых кодов путём задания только длины информационного блока и корректирующей способности кода. Причём такие кодеки должны реализовываться на программируемых логических интегральных схемах (ПЛИС) или на микроконтроллерах. Известные системы по созданию кодеров помехоустойчивых кодов носят исследовательский характер и не имеют функционал, пригодный для практического применения. Для решения проблемы создания практически

значимых систем необходим комплексный подход, начиная от разработки структуры, алгоритмов функционирования систем и методов и алгоритмов поиска соответствующего помехоустойчивого кода, до создания программного обеспечения (ПО) таких автоматизированных систем для разработки кодеков помехоустойчивых кодов.

Таким образом, можно сделать вывод об актуальности задач разработки алгоритмического и программного обеспечения систем.

Исследования и разработки по теме диссертационной работы проводились в соответствии с утвержденным планом НИР Института «Кибернетический центр» ТПУ в 2005-2010 гг., а также поддержаны программой «У.М.Н.И.К.» Фонда содействия развитию малых форм предприятий в научно-технической сфере и грантом Томского политехнического университета для молодых учёных на проведение научных исследований.

**Цель работы и задачи исследования.** Целью диссертационного исследования является разработка алгоритмического и программного обеспечения системы для создания кодеков помехоустойчивых полиномиальных кодов, исправляющих независимые ошибки кратности от 1 до 4. Система должна позволять подготавливать данные для различных САПР при проектировании кодеков на ПЛИС или для программной реализации кодеков на микроконтроллерах в зависимости от задаваемых пользователем параметров (длина информационного блока и корректирующая способность кода).

Для достижения поставленной цели необходимо последовательное решение следующих задач.

1. Формирование требований к системе для разработки кодеков помехоустойчивых кодов и создание алгоритма её функционирования.
2. Разработка алгоритма поиска образующих полиномов и алгоритмов деления полиномов в арифметике по модулю два, а также исследование их эффективности. Решение данной задачи предполагает разработку ПО для проведения таких исследований.
3. Разработка алгоритма функционирования проектируемого кодера и создание на его основе шаблона для САПР при реализации кодеков на ПЛИС различных фирм-производителей.
4. Разработка ПО системы для создания кодеков помехоустойчивых полиномиальных кодов. Результатом решения этой задачи должны явиться структура ПО системы и программные модули системы, разработанные с использованием технологии объектно-ориентированного программирования.
5. Аprobация разработанного ПО системы на примере решения прикладных задач повышения достоверности передачи данных.

**Методы исследований.** В работе использованы методы теории множеств, теории вероятности, дискретной математики, теории информации и объектно-ориентированного программирования. Проверка эффективности разработанных алгоритмов и программных средств осуществлялась путём проведения компьютерных экспериментов с использованием специально

разработанного ПО и систем автоматизированного проектирования сторонних разработчиков.

**Научную новизну** полученных в работе результатов определяют.

1. Предложенный алгоритм функционирования автоматизированной системы для разработки кодеков помехоустойчивых полиномиальных кодов, позволяющий создавать проектные файлы кодеков, как для последующей программной реализации кодеков на микроконтроллерах, так и для САПР при проектировании кодеков на ПЛИС различных фирм-производителей.
2. Алгоритм поиска образующих полиномов, отличающийся от известных меньшим количеством итераций вычислений, необходимых для поиска образующего полинома. Используя его, получены образующие полиномы, на основе которых строятся эффективные помехоустойчивые полиномиальные коды.
3. Впервые предложенный и математически обоснованный матричный алгоритм деления полиномов в арифметике по модулю два.
4. Разработанный шаблон кодека помехоустойчивых полиномиальных кодов, реализованный на языке описания аппаратуры Verilog.

**Практическая ценность и реализация результатов работы.**

Практически значимыми являются созданные алгоритмы, ПО системы для разработки кодеков MagDiv и ПО для поиска образующих полиномов. Разработанное ПО системы MagDiv и ПО для поиска образующих полиномов имеют в своём составе кроссплатформенную библиотеку классов RCodeWords, также имеющую самостоятельную практическую ценность. ПО системы MagDiv функционирует на компьютерах под управлением ОС Windows 9x/NT/2000/XP/Vista. ПО для поиска образующих полиномов функционирует на суперкомпьютерном кластере «СКИФ-политех» под управлением ОС Linux или на компьютерах под управлением ОС Windows 9x/NT/2000/XP/Vista. Объём исходного кода разработанного ПО составляет более 2500 строк на языках C++ и Verilog.

Результаты работы внедрены в ООО «Технологическая Компания Шлюмберже» (в прошлом Томский филиал ООО «Сибирская Геофизическая Компания») и в ООО «ТомскНефтеГазИнжиниринг». Результаты внедрения подтверждены соответствующими актами.

**Апробация работы.** Основные результаты работы докладывались и обсуждались на следующих симпозиумах, конференциях и семинарах: региональная научно-практическая конференция «Молодежь и современные информационные технологии» (г. Томск, 2003 г.), Корейско-Русский Международный симпозиум науки и технологии “KORUS 2003” (Республика Корея, г. Ульсан, 2003 г.), Конференция-конкурс «Технологии Microsoft в информатике и программировании» (г. Новосибирск, 2004 г.), II, III, IV и VI Всероссийская научно-практическая конференция студентов, аспирантов и молодых учёных «Молодежь и современные информационные технологии» (г. Томск, 2004, 2005, 2006 и 2008 гг.), X, XIII и XIV Международная научно-практическая конференция студентов, аспирантов и молодых учёных

«Современные техника и технологии» (г. Томск, 2004, 2007 и 2008 гг.), Пятая юбилейная Всероссийская научно-практическая конференция «Средства и системы автоматизации» (г. Томск, 2005 г.).

Результаты диссертационной работы оценивались на конкурсах и получили следующие награды: медаль Министерства образования и науки РФ «За лучшую научную студенческую работу» в 2005 г. и медаль Российской академии наук для студентов высших учебных заведений в области информатики, вычислительной техники и автоматизации в 2006 г.

По результатам диссертационных исследований опубликовано 15 работ, в том числе 4 статьи в рецензируемых изданиях, рекомендованных ВАК РФ.

#### **Личный вклад.**

1. Постановка задач диссертационного исследования выполнена автором совместно с Н.Г. Марковым.
2. Разработка алгоритма функционирования системы для создания кодеков помехоустойчивых полиномиальных кодов выполнена автором совместно с А.Н. Осокиным.
3. Математическое обоснование матричного алгоритма деления полиномов по модулю два выполнено совместно с Ю.Б. Буркатовской.
4. Разработка остальных предложенных алгоритмов выполнена автором.
5. Разработка ПО системы MagDiv, ПО для поиска образующих полиномов и программных средств для проведения исследований эффективности предложенных алгоритмов проведена автором.
6. Шаблон проектируемого кодека помехоустойчивых полиномиальных кодов на языке Verilog разработан автором.

#### **Основные положения, выносимые на защиту.**

1. Созданное ПО системы MagDiv позволяет разрабатывать кодеки помехоустойчивых полиномиальных кодов, реализуемые как программно на микроконтроллерах, так и аппаратно на ПЛИС различных фирм-производителей.
2. Алгоритм поиска образующих полиномов позволяет получать полиномы, на основе которых строятся помехоустойчивые полиномиальные коды, имеющие меньшую избыточность в большинстве практически значимых случаях, чем у наиболее эффективных из известных полиномиальных циклических БЧХ-кодов.
3. Матричный алгоритм деления полиномов в арифметике по модулю два позволяет вычислять остаток от деления полиномов в 3-6 раз быстрее по сравнению с известным классическим алгоритмом деления полиномов.
4. Шаблон проектируемого кодека помехоустойчивых полиномиальных кодов, написанный на языке Verilog, позволяет создавать в САПР кодеки на ПЛИС различных фирм-производителей.

**Структура и объём диссертации.** Диссертация состоит из введения, четырех глав, заключения, списка литературы из 92 наименований и приложения. Основная часть диссертационного исследования изложена на 133 страницах машинописного текста. Работа иллюстрирована 59 рисунками и 10 таблицами.

## СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обосновывается актуальность диссертационной работы, формулируются цель и задачи исследования и приводится краткое содержание работы по главам.

**Первая глава** посвящена анализу существующих автоматизированных систем для разработки кодеков помехоустойчивых кодов, а также анализу современных методов и алгоритмов помехоустойчивого кодирования.

Проанализированы существующие автоматизированные системы для разработки кодеков помехоустойчивых кодов. Оказалось, что все они носят исследовательский характер и не доведены до уровня практического применения.

Анализ существующих систем для создания кодеков и имеющиеся тенденции проектирования кодеков с использованием современной элементной базы позволили сформулировать требования к практически значимым системам для разработки кодеков помехоустойчивых кодов.

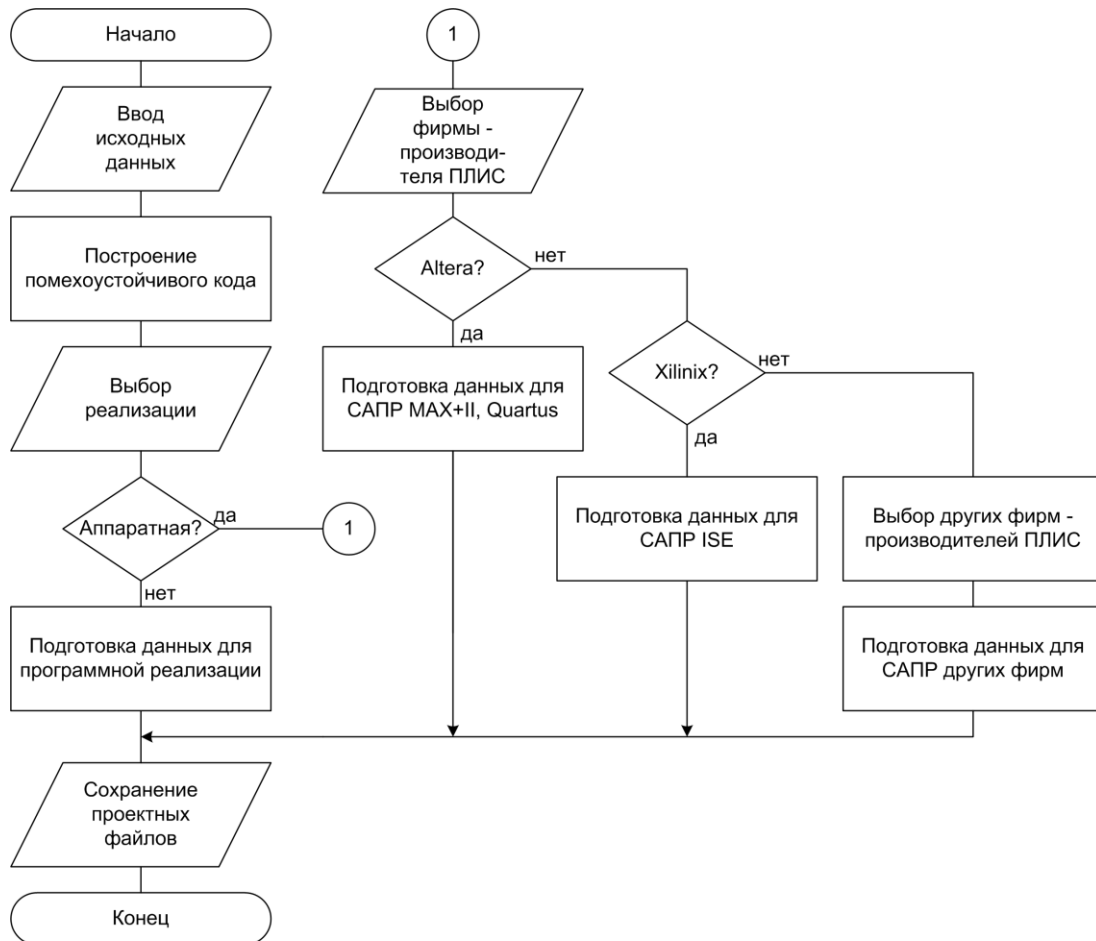
Значительной проблемой при создании таких систем является выбор помехоустойчивого кода. Для этого сформулированы требования к помехоустойчивому коду проектируемых кодеков и проведена классификация основных помехоустойчивых кодов. В результате анализа кодов показаны проблемы, связанные с невысокой эффективностью и корректирующей способностью большинства классов кодов, а также отмечены помехоустойчивые полиномиальные коды как наиболее перспективные. Выявлена необходимость решения актуальных задач разработки структуры и эффективного алгоритмического и программного обеспечения системы для создания кодеков помехоустойчивых полиномиальных кодов.

Сформулированы цель и задачи исследований.

Во **второй главе** рассматриваются алгоритмы, положенные в основу создаваемой системы, а также приводятся результаты их исследований.

На основе сформулированных в главе 1 требований к системе разработан алгоритм её функционирования (рис. 1). По исходным данным (длина информационного блока и количество исправляемых ошибок) система позволяет строить помехоустойчивый полиномиальный код. Затем, в случае выбора программной реализации кода, подготавливаются необходимые проектные данные для управляющей программы микроконтроллера. В случае выбора аппаратной реализации, создаются проектные файлы с учётом указанной фирмы-производителя ПЛИС и особенностей соответствующей САПР.

Предложен оригинальный алгоритм поиска образующих полиномов. В основу этого алгоритма положен способ уменьшения количества итераций вычислений, учитывающий зависимость корректирующей способности кода и веса образующего полинома.



**Рис. 1.** Схема алгоритма функционирования системы для разработки кодеков помехоустойчивых кодов

Оригинальный алгоритм поиска образующих полиномов описывается следующим образом.

**Начало.**

**Шаг 1.** Задаются разрядность информационного блока  $m$  и количество (кратность) независимых ошибок  $t$ .

**Шаг 2.** Вычисляется минимальная длина контрольного блока (старшая степень образующего полинома) по формуле для поиска «границы Хэмминга»:

$$k = \left\lceil \log_2 \left( \sum_{i=0}^t C_n^i \right) \right\rceil,$$

где  $k$  – старшая степень образующего полинома,  $n$  – разрядность кодового слова,  $\lceil a \rceil$  – большее целое от  $a$ ,  $C_n^i$  – число сочетаний  $i$  из  $n$ .

**Шаг 3.** Выбирается полином из множества, состоящего из полиномов со старшей степенью, равной  $k$ , и весом, равным минимальному расстоянию кода  $d_m$ . Минимальное расстояние кода, исправляющего  $t$  независимых ошибок, вычисляется по формуле:  $d_m = 2t + 1$ .

**Шаг 4.** Вычисляется минимальное кодовое расстояние кода (путём нахождения минимального веса среди всех кодовых слов, за исключением нулевого слова), строящегося на основе выбранного полинома.



**Шаг 5.** Если вычисленное кодовое расстояние удовлетворяет заданным требованиям, то переходим на шаг 7, иначе переходим на шаг 6.

**Шаг 6.** Если проверены все полиномы из множества (множество задано на шаге 3), то увеличиваем  $k$  на единицу. Переходим на шаг 3.

**Шаг 7.** Если все синдромы ошибок уникальны для построенного кода, то полином принимается в качестве образующего полинома и переходим на Конец, иначе переходим на шаг 6.

**Конец.**

Представлены результаты поиска образующих полиномов (табл.1), который проводился на суперкомпьютерном кластере «СКИФ-политех» с помощью специально разработанного ПО.

**Таблица 1.** Найденные образующие полиномы

| $\begin{matrix} t \\ m \end{matrix}$ | 1       | 2             | 3                   | 4                         |
|--------------------------------------|---------|---------------|---------------------|---------------------------|
| 1                                    | 111     | 11111         | 1111111             | 111111111                 |
| 2                                    | 1101    | 1101101       | 1110110101          | 1110101110101             |
| 3                                    | 1101    | 11001101      | 11101100101         | 111010011101001           |
| 4                                    | 1101    | 11100101      | 11101100101         | 1111000101100101          |
| 5                                    | 11001   | 110101001     | 11101100101         | 1111000100101101          |
| 6                                    | 11001   | 111010001     | 101101110001        | 10111000110110001         |
| 7                                    | 11001   | 111010001     | 110001110101        | 111000101001011001        |
| 8                                    | 11001   | 100111001     | 110001110101        | 110000111000110101        |
| 9                                    | 11001   | 100111001     | 110001110101        | 1110000110010010101       |
| 10                                   | 11001   | 1100001101    | 110001110101        | 1001100001010101101       |
| 11                                   | 11001   | 1101000011    | 110001110101        | 1000111110010100001       |
| 12                                   | 110001  | 1101000011    | 110001110101        | 11100001010110000011      |
| 13                                   | 110001  | 1101000011    | 110010101000101     | 10110001100101001001      |
| 14                                   | 110001  | 11000011001   | 1100001010010101    | 111001000000100110101     |
| 15                                   | 110001  | 11000011001   | 1100001010010101    | 111001110001000000101     |
| 16                                   | 110001  | 11000011001   | 1100001010010101    | 1111011000001000000101    |
| 17                                   | 101001  | 11000011001   | 11000011010000011   | 1111010000100011000001    |
| 18                                   | 101001  | 11000011001   | 11001000001100101   | 1110000100011100010001    |
| 19                                   | 101001  | 11000011001   | 110100001000010011  | 1011100100000001110001    |
| 20                                   | 101001  | 11000011001   | 100110010001100001  | 11110001000010100001001   |
| 21                                   | 101001  | 11000011001   | 100110010001100001  | 11001011000100000010101   |
| 22                                   | 101001  | 101100001001  | 100110010001100001  | 111100100000010001001001  |
| 23                                   | 101001  | 101100001001  | 110010100001100001  | 111100100000010001001001  |
| 24                                   | 101001  | 101100001001  | 110010100001100001  | 111010001000100001000011  |
| 25                                   | 101001  | 1100010010001 | 110010100001100001  | 111010001000100001000011  |
| 26                                   | 101001  | 1101010000001 | 110010100001100001  | 111010001000100001000011  |
| 27                                   | 1100001 | 1100010010001 | 1100100100010001001 | 1111001000010010000000011 |
| 28                                   | 1100001 | 1010110000001 | 1101000101000000011 | 1111001000010010000000011 |
| 29                                   | 1100001 | 1010110000001 | 1101000101000000011 | 1100110000000010001010011 |
| 30                                   | 1100001 | 1001110010101 | 1111001101000001111 | 1110111011100100110110111 |
| 31                                   | 1100001 | 1001110010101 | 1111001101000001111 | 1110111011100100110110111 |
| 32                                   | 1100001 | 1001110010101 | 1111001101000001111 | 1110111011100100110110111 |

В табл. 1 представлены образующие полиномы в виде двоичных чисел, позволяющие строить помехоустойчивые полиномиальные коды с разрядностью информационного блока  $m$ , равной от 1 до 32 бит, и исправляющие независимые ошибки кратности  $t$  от 1 до 4 соответственно.

Длины кодовых слов  $n_a$  полиномиальных кодов, получаемых с помощью образующих полиномов из табл. 1, приведены в столбце А табл. 2. Для сравнения полученных результатов в столбце Б приведены длины кодовых слов  $n_b$  БЧХ-кодов как наиболее эффективных из известных. Примеры результатов анализа параметров помехоустойчивых кодов, приведённых в табл. 2, представлены в виде гистограмм на рис. 2. БЧХ-коды считаются оптимальными для исправления двукратной ошибки. Однако, из рис. 2 видно, что полиномиальные коды в ряде случаев (например, для  $t=2$  и  $m=(1-4, 8-13, 22-24)$ ) обладают меньшей избыточностью, чем БЧХ-коды. В остальных случаях избыточность одинаковая. Это говорит о большей эффективности (меньшей избыточности) предлагаемых полиномиальных кодов.

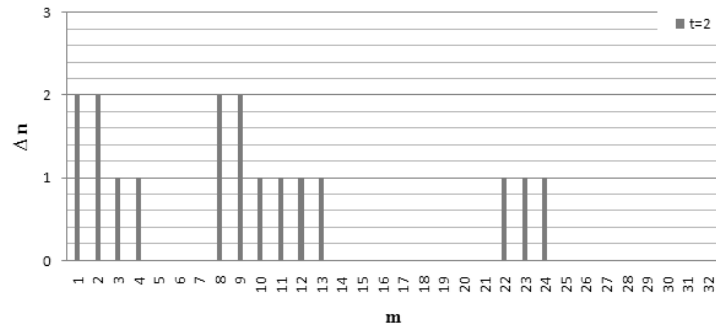
Предложена модификация алгоритма ускоренного деления полиномов Башина, которая получила название «двусторонний алгоритм деления полиномов», и разработан оригинальный алгоритм матричного деления полиномов в арифметике по модулю два.

Описание матричного алгоритма деления полиномов представлено ниже.

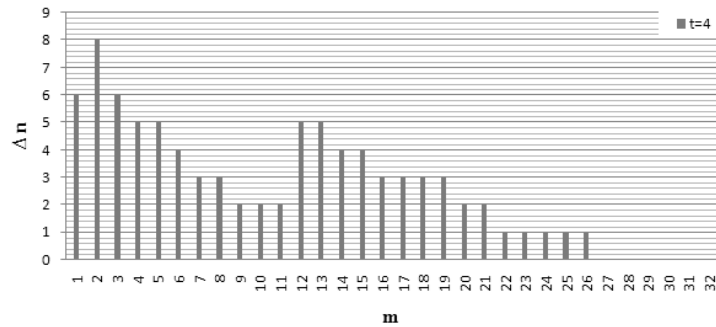
**Таблица 2.** Длины кодовых слов

| $\begin{matrix} t \\ m \end{matrix}$ | А  |    |    |    | Б  |    |    |    |
|--------------------------------------|----|----|----|----|----|----|----|----|
|                                      | 1  | 2  | 3  | 4  | 1  | 2  | 3  | 4  |
| 1                                    | 3  | 5  | 7  | 9  | 3  | 7  | 7  | 15 |
| 2                                    | 5  | 8  | 11 | 14 | 5  | 10 | 12 | 22 |
| 3                                    | 6  | 10 | 13 | 17 | 6  | 11 | 13 | 23 |
| 4                                    | 7  | 11 | 14 | 19 | 7  | 12 | 14 | 24 |
| 5                                    | 9  | 13 | 15 | 20 | 9  | 13 | 15 | 25 |
| 6                                    | 10 | 14 | 17 | 22 | 10 | 14 | 21 | 26 |
| 7                                    | 11 | 15 | 18 | 24 | 11 | 15 | 22 | 27 |
| 8                                    | 12 | 16 | 19 | 25 | 12 | 18 | 23 | 28 |
| 9                                    | 13 | 17 | 20 | 27 | 13 | 19 | 24 | 29 |
| 10                                   | 14 | 19 | 21 | 28 | 14 | 20 | 25 | 30 |
| 11                                   | 15 | 20 | 22 | 29 | 15 | 21 | 26 | 31 |
| 12                                   | 17 | 21 | 23 | 31 | 17 | 22 | 27 | 36 |
| 13                                   | 18 | 22 | 27 | 32 | 18 | 23 | 28 | 37 |
| 14                                   | 19 | 24 | 29 | 34 | 19 | 24 | 29 | 38 |
| 15                                   | 20 | 25 | 30 | 35 | 20 | 25 | 30 | 39 |
| 16                                   | 21 | 26 | 31 | 37 | 21 | 26 | 31 | 40 |
| 17                                   | 22 | 27 | 33 | 38 | 22 | 27 | 35 | 41 |
| 18                                   | 23 | 28 | 34 | 39 | 23 | 28 | 36 | 42 |
| 19                                   | 24 | 29 | 36 | 40 | 24 | 29 | 37 | 43 |
| 20                                   | 25 | 30 | 37 | 42 | 25 | 30 | 38 | 44 |
| 21                                   | 26 | 31 | 38 | 43 | 26 | 31 | 39 | 45 |
| 22                                   | 27 | 33 | 39 | 45 | 27 | 34 | 40 | 46 |
| 23                                   | 28 | 34 | 40 | 46 | 28 | 35 | 41 | 47 |
| 24                                   | 29 | 35 | 41 | 47 | 29 | 36 | 42 | 48 |
| 25                                   | 30 | 37 | 42 | 48 | 30 | 37 | 43 | 49 |
| 26                                   | 31 | 38 | 43 | 49 | 31 | 38 | 44 | 50 |
| 27                                   | 33 | 39 | 45 | 51 | 33 | 39 | 45 | 51 |
| 28                                   | 34 | 40 | 46 | 52 | 34 | 40 | 46 | 52 |
| 29                                   | 35 | 41 | 47 | 53 | 35 | 41 | 47 | 53 |
| 30                                   | 36 | 42 | 48 | 54 | 36 | 42 | 48 | 54 |
| 31                                   | 37 | 43 | 49 | 54 | 37 | 43 | 49 | 54 |
| 32                                   | 38 | 44 | 50 | 55 | 38 | 44 | 50 | 55 |

а



б



**Рис. 2.** Примеры зависимостей разницы  $\Delta n = n_b - n_a$  от  $m$  для корректирующей способности кода  $t=2$  (а) и  $t=4$  (б)

Матрица  $Z$  определяется следующим образом:

$$Z = \begin{bmatrix} z_{0,k-1} & \cdots & z_{0,0} \\ z_{1,k-1} & \cdots & z_{1,0} \\ \vdots & & \\ z_{m-1,k-1} & \cdots & z_{m-1,0} \end{bmatrix},$$

где  $k$  – разрядность контрольного блока,  $z_i(x) = R_{B(x)}(x^{n-1-i})$  (остаток от деления  $x^{n-1-i}$  на делитель  $B(x)$ ).

Матрица  $Y$  определяется следующим образом:

$$Y = \begin{bmatrix} y_{0,m-1} & \cdots & y_{0,0} \\ y_{1,m-1} & \cdots & y_{1,0} \\ \vdots & & \\ y_{m-1,m-1} & \cdots & y_{m-1,0} \end{bmatrix},$$

где  $y_i(x) = \frac{x^{n-1-i}}{B(x)}$  (частное от деления  $x^{n-1-i}$  на делитель  $B(x)$ ).

**Начало.**

**Шаг 1.** Даны полиномы:  $A(x)$  – делимое,  $B(x)$  – делитель.

**Шаг 2.** Вычислить матрицу  $Z$  для полинома  $B(x)$ .

**Шаг 3.** Вычислить матрицу  $Y$  для делимых разрядностью  $n$ .

**Шаг 4.** Вычислить остаток  $A' \cdot Z \oplus \tilde{A}$  (здесь и ниже  $\oplus$  – сложение по модулю два), где  $A'(x) = \sum_{i=k}^{n-1} a_i x^i$ ,  $\tilde{A}(x) = \sum_{i=0}^{k-1} a_i x^i$ .

**Шаг 5.** Вычислить частное  $A' \cdot Y$ .

**Конец.**

Для математического обоснования матричного алгоритма сформулирована и доказана теорема.

**Теорема.** Если  $Z$  – специальным образом сформированная матрица от  $B(x)$ , то остаток от деления  $A(x)$  на  $B(x)$  будет равен  $A' \cdot Z \oplus \tilde{A}$ .

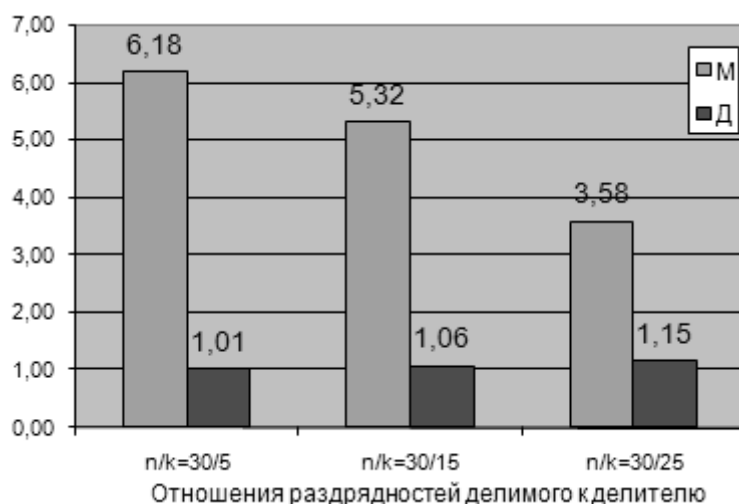
Для быстрого формирования матриц  $Z$  и  $Y$ , получены и доказаны методом математической индукции рекуррентные формулы:

$$z_{m-1}(x) = \sum_{i=0}^{k-1} b_i x^i; \quad y_{m-1} = 1;$$

$$z_i(x) = z_{i+1}(x) \cdot x \oplus z_{i+1,k-1} \cdot B(x); \quad y_i(x) = y_{i+1}(x) \cdot x \oplus z_{i+1,k-1},$$

где  $z_{i+1,k-1}$  – старший разряд  $z_{i+1}$ ;  $i = m-2, 0$ .

Результаты исследования эффективности алгоритмов деления полиномов показали большее быстродействие оригинального матричного алгоритма деления полиномов по сравнению с остальными алгоритмами. На рис. 3 в виде гистограмм приведена в качестве примера часть результатов исследования эффективности алгоритмов деления полиномов для соотношений разрядностей делимого к делителю, равных 30/5, 30/15 и 30/25. Показано во сколько раз по быстродействию матричный и двусторонний алгоритмы деления полиномов превосходят классический алгоритм. Эти результаты получены с относительной ошибкой в пределах 3-4% при доверительной вероятности 95%.



**Рис. 3.** Сравнение быстродействия матричного и двустороннего алгоритмов деления полиномов относительно быстродействия классического алгоритма; М – случай матричного алгоритма, Д – случай двустороннего алгоритма

С целью выбора алгоритма функционирования кодека (в случае аппаратной реализации – структуры кодека) проведены исследования,

позволившие разработать алгоритмы функционирования кодера и декодера. Эти алгоритмы положены в основу шаблона проектируемого кодека, созданного на языке описания аппаратуры Verilog.

Предложены методы уменьшения требуемого объёма энергонезависимой памяти при аппаратной реализации кодеков помехоустойчивых полиномиальных кодов с синдромным декодированием. При этом предложены метод составления словаря и метод выборки исправляющей комбинации по принципу организации частично-ассоциативной кэш-памяти. Показано, что использование метода составления словаря более предпочтительно из-за его простоты реализации, в отличие от метода выборки исправляющей комбинации по принципу организации частично-ассоциативной кэш-памяти.

**Третья глава** посвящена описанию созданного ПО системы MagDiv для разработки кодеков помехоустойчивых полиномиальных кодов, кроссплатформенной библиотеки классов PCodeWords и ПО для поиска образующих полиномов.

Сформулирован набор требований к ПО системы MagDiv. С учётом этих требований разработана структура ПО этой системы (рис. 4). На рис. 4 жирными линиями обозначены потоки данных. Обращения к классам и их методам показаны нежирными линиями.

ПО системы MagDiv реализует приведённый на рис. 1 алгоритм и представляет из себя набор программных модулей, часть из которых в своей работе использует библиотеку классов PCodeWords. Другие модули системы используют эту библиотеку через методы класса DivLib. Входными данными для модулей системы является задаваемые пользователем параметры проектируемого кодека (длина информационного блока и корректирующая способность кода), файл данных образующих полиномов и файл данных шаблона проектируемого кодека. Результатом работы ПО являются файлы проекта кодека. В случае выбора пользователем программной реализации кодека на микроконтроллерах в проекте содержатся данные, необходимые для написания управляющей программы для микроконтроллера. В случае аппаратной реализации кодека на ПЛИС с использованием выбранной САПР проект кодека компилируется и затем конфигурируется конкретная ПЛИС. ПО системы MagDiv написано на языке C++ в среде разработки Borland C++ Builder 6.0. Выбор этой среды обусловлен развитыми средствами визуального программирования и большим разнообразием инструментов. На рис. 5 приведены примеры реализации пользовательского интерфейса системы.

Описывается кроссплатформенная библиотека классов PCodeWords, лежащая в основе разработанного ПО для поиска образующих полиномов и ПО системы MagDiv. Эта библиотека создана с учётом современных требований к разработке библиотек классов. Библиотека классов PCodeWords имеет самостоятельную практическую ценность, поскольку позволяет создавать на её основе исследовательское ПО в области полиномиальных вычислений и помехоустойчивого кодирования.

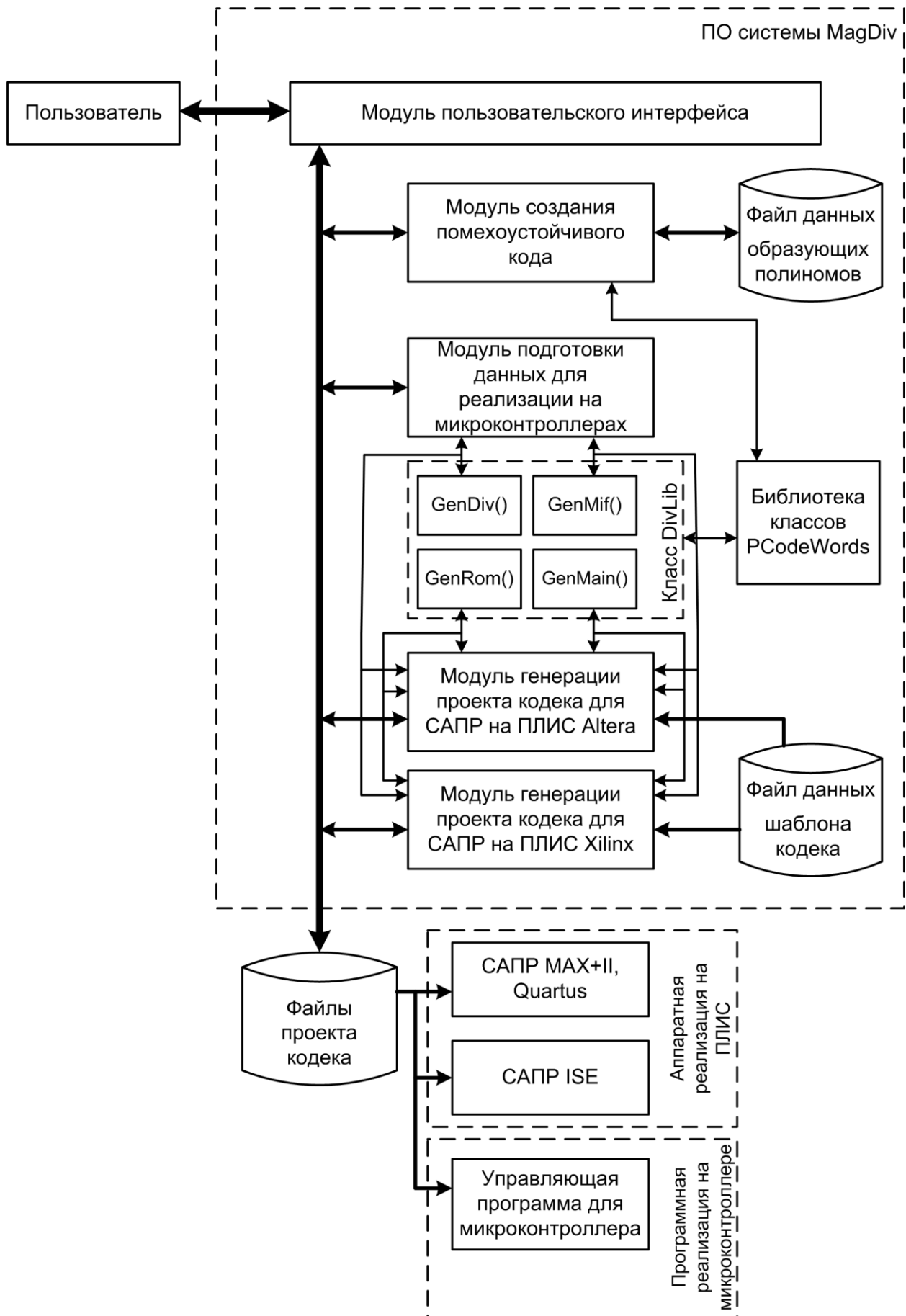
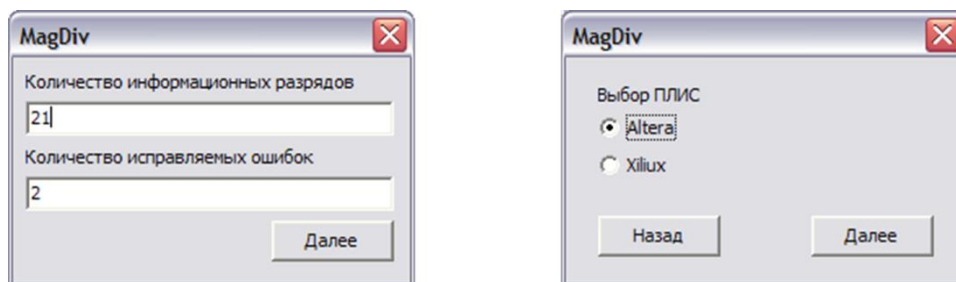


Рис. 4. Структура ПО системы MagDiv



**Рис. 5.** Примеры реализации пользовательского интерфейса ПО системы MagDiv

Приводится описание разработанного на языке C++ в среде программирования Microsoft Visual Studio C++ 6.0 ПО для поиска образующих полиномов. Указываются особенности программной реализации и описывается структура ПО. Дается описание разработанного интерфейса пользователя. Приводятся примеры результатов работы созданных программных средств для компьютера под управлением ОС Windows XP и для суперкомпьютерного кластера «СКИФ-политех» под управлением ОС Linux.

В **четвёртой главе** приводятся результаты апробации разработанного ПО системы MagDiv при решении прикладных задач повышения достоверности передачи данных на основе применения кодеков помехоустойчивых полиномиальных кодов.

Описываются результаты применения разработанного ПО системы MagDiv для решения задачи повышения достоверности передачи данных в системе управления процессом бурения и для решения задачи повышения достоверности передачи данных в инклинометрической системе для буровой установки. Разработаны соответствующие кодеки помехоустойчивых полиномиальных кодов. По результатам апробации сделан вывод о работоспособности и эффективности алгоритмического и программного обеспечения системы MagDiv. Указывается, что результаты диссертационного исследования внедрены в ООО «Технологическая Компания Шлюмберже» (в прошлом Томский филиал ООО «Сибирская Геофизическая Компания») и в ООО «ТомскНефтеГазИнжиниринг».

В **заключении** приведены основные выводы и результаты диссертационной работы.

В **приложение** вынесены акты о внедрении полученных результатов.

## ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ

Диссертационная работа посвящена созданию алгоритмического и программного обеспечения системы для разработки кодеков помехоустойчивых полиномиальных кодов.

В ходе выполнения диссертационной работы получены следующие основные научные и практические результаты и сделаны выводы.

1. Проведён анализ существующих автоматизированных систем для разработки кодеков помехоустойчивых кодов, а также анализ современных методов и алгоритмов помехоустойчивого кодирования. Сделан вывод о необходимости создания практически значимых систем для разработки таких кодеков, имеющих эффективное алгоритмическое и программное обеспечение.
2. Сформулированы требования к создаваемой автоматизированной системе, на основе которых разработан алгоритм её функционирования. Сформулированы требования к помехоустойчивому коду проектируемого кодека. Сделан вывод о необходимости разработки алгоритмов помехоустойчивого кодирования на основе помехоустойчивых полиномиальных кодов.
3. Разработан оригинальный алгоритм поиска образующих полиномов, отличающийся от известных меньшим количеством требуемых итераций вычислений. На основе предложенного алгоритма разработано ПО для поиска образующих полиномов. Проведены компьютерные эксперименты на суперкомпьютерном кластере «СКИФ-политех», в результате которых вычислены образующие полномы, позволяющие строить помехоустойчивые полиномиальные коды, имеющие в ряде практически значимых случаях меньшую избыточность, чем BCH-коды.
4. Разработан быстродействующий алгоритм матричного деления полиномов в арифметике по модулю два. Результаты компьютерного эксперимента подтвердили теоретические предположения о его эффективности по быстродействию. Матричный алгоритм позволяет создавать быстродействующие кодеки помехоустойчивых полиномиальных кодов.
5. Предложены методы уменьшения требуемого объёма энергонезависимой памяти при аппаратной реализации кодека с синдромным декодированием, которые позволяют создавать кодеки в условиях нехватки объёма такой памяти.
6. Разработаны структура ПО и программные модули системы MagDiv для создания кодеков помехоустойчивых полиномиальных кодов. В рамках этой системы разработан шаблон проектируемого кодека на языке описания аппаратуры Verilog. Шаблон позволяет ПО системы MagDiv создавать проектные файлы кодеков для САПР различных фирм-производителей ПЛИС.  
Разработано исследовательское ПО для поиска образующих полиномов, с помощью которого проведён компьютерный эксперимент и найдены образующие полиномы. Результаты применения этого ПО позволили исключить из ПО системы MagDiv ресурсоёмкую часть по подбору образующих полиномов.  
Объём исходного кода разработанного ПО составляет более 2500 строк на языках C++ и Verilog.
7. Разработана кроссплатформенная библиотека классов PCodeWords, реализующая предложенный матричный алгоритм деления полиномов в



арифметике по модулю два и входящая в состав ПО системы MagDiv и в состав ПО для поиска образующих полиномов. Эта библиотека имеет также самостоятельную практическую ценность для полиномиальных вычислений и помехоустойчивого кодирования.

8. Проведена апробация разработанного алгоритмического и программного обеспечения системы MagDiv при решении прикладных задач повышения достоверности передачи данных в системе управления процессом бурения и в инклинометрической системе для буровой установки. Разработаны соответствующие кодеки помехоустойчивых полиномиальных кодов. Результаты апробации подтвердили работоспособность и эффективность созданного алгоритмического и программного обеспечения системы MagDiv.

### ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

В рецензируемых изданиях, рекомендованных ВАК РФ

1. **Мальчуков, А.Н.** Быстродействующий кодек БЧХ-кодов на ПЛИС / Буркатовская Ю.Б., Мальчуков А.Н., Осокин А.Н. // Приборы и системы. Управление, контроль, диагностика. – 2006. – №3. – С. 21-23.
2. **Мальчуков, А.Н.** Быстродействующие алгоритмы деления полиномов в арифметике по модулю два / Буркатовская Ю.Б., Мальчуков А.Н., Осокин А.Н. // Известия Томского политехнического университета. – 2006. – Т. 309. – № 1. – С. 19-24.
3. **Мальчуков, А.Н.** Эффективность блоковых помехоустойчивых кодов / Мальчуков А.Н., Осокин А.Н. // Известия Томского политехнического университета. – 2006. – Т. 309. – № 7. – С. 102-105.
4. **Мальчуков, А.Н.** Система автоматизированного проектирования кодеров помехоустойчивых кодов короткой длины / Мальчуков А.Н., Осокин А.Н. // Известия Томского политехнического университета. – 2008. – Т. 312. – № 5. – С. 70-75.

Остальные публикации

5. **Мальчуков, А.Н.** Алгоритм ускоренного деления полиномов в арифметике по модулю два с использованием матричной арифметики / Буркатовская Ю.Б., Мальчуков А.Н., Осокин А.Н. // Труды X Юбилейной международной научно-практической конференции студентов, аспирантов и молодых учёных «Современные техника и технологии» – Томск: Изд-во Томского политехнического университета, 2004. – Т.2. – С.171-173.
6. **Мальчуков, А.Н.** Реализация кодера помехоустойчивого двоичного циклического кода в потоке параллельных данных на ПЛИС / Буркатовская Ю.Б., Мальчуков А.Н., Осокин А.Н. // Материалы V юбилейной Всероссийской научно-практической конференции «Средства и системы автоматизации» – Томск: Изд-во Томского государственного университета систем управления и радиоэлектроники, 2004. – С.102-105.

7. **Мальчуков, А.Н.** Структура быстродействующего кода помехоустойчивого двоичного циклического кода / Буркатовская Ю.Б., Мальчуков А.Н., Осокин А.Н. // Сб. трудов III Всероссийской научно-практической конференции «Молодежь и современные информационные технологии» – Томск: Изд-во Томского политехнического университета, 2005. – С. 272-273.
8. **Мальчуков, А.Н.** Алгоритм ускоренного деления в арифметике по модулю два / Мальчуков А.Н., Осокин А.Н. // Сб. трудов региональной научно-практической конференции «Молодежь и современные информационные технологии» – Томск: Изд-во Томского политехнического университета, 2003. – С. 12-13.
9. **Мальчуков, А.Н.** Матричный метод деления полиномов в арифметике по модулю два // Тез. докл. конференции-конкурса работ студентов, аспирантов и молодых учёных «Технологии Microsoft в информатике и программировании» – Новосибирск: Изд-во Новосибирского государственного университета, 2004. – С. 171-173.
10. **Мальчуков, А.Н.** Программная реализация алгоритмов деления полиномов в арифметике по модулю два // Сб. II Всероссийской научно-практической конференции «Молодежь и современные информационные технологии», – Томск: Изд-во Томского политехнического университета, 2004. – С.32-33.
11. **Мальчуков, А.Н.** Помехоустойчивое кодирование в кроссбар-нанокомпьютерах / Мальчуков А.Н., Осокин А.Н. // Труды XIII Международной научно-практической конференции студентов, аспирантов и молодых учёных «Современные техника и технологии» – Томск: Изд-во Томского политехнического университета, 2007. – С. 386-388.
12. **Мальчуков, А.Н.** Система проектирования кодеров помехоустойчивых кодов с развитыми функциями автоматизации / Мальчуков А.Н., Осокин А.Н. // Сб. трудов VI Всероссийской научно-практической конференции студентов, аспирантов и молодых учёных «Молодежь и современные информационные технологии» – Томск: Изд-во Томского политехнического университета, 2008. – С. 45-46.
13. **Мальчуков, А.Н.** Алгоритм поиска образующих полиномов для системы проектирования кодеров помехоустойчивых кодов // Труды XIV Международной научно-практической конференции студентов, аспирантов и молодых учёных «Современные техника и технологии» – Томск: Изд-во Томского политехнического университета, 2008. – Т. 2. – С. 340-341.
14. **Мальчуков, А.Н.** Методы минимизации требуемого объёма памяти при реализации табличного метода декодирования БЧХ-кода / Негрей Я.Е., Мальчуков А.Н., Осокин А.Н. // Сб. трудов IV Всероссийской научно-практической конференции студентов, аспирантов и молодых учёных «Молодежь и современные информационные технологии» – Томск: Изд-во Томского политехнического университета, 2006. – С. 389-391.

15. **Andrey N. Malchukov.** Algorithms of accelerated division on modulo 2 / Yulia B. Bourkatovskaya, Andrey N. Malchukov, Alexandr N. Osokin. // Proceedings of the 7th Korea-Russia International Symposium on Science and Technology “KORUS 2003” – 2003. – vol. 2. “Electrical Engineering and Inform” – pp. 189-193.
- Мальчуков, А.Н.** Алгоритм ускоренного деления по модулю два / Буркатовская Ю.Б., Мальчуков А.Н., Осокин А.Н. // Сб. трудов 7-го Корейско-Русского Международного симпозиума науки и технологии “КОРУС 2003”. – Республика Корея, Ульсан: Изд-во Университета Ульсан, 2003. – Т.2. – С. 189-193.