На правах рукописи

Ратников Максим Олегович

Применение конвейеризированных генераторов контрольных кодов для проведения анализа сбоеустойчивости систем на основе ПЛИС

2.3.2. – Вычислительные системы и их элементы

АВТОРЕФЕРАТ

диссертации на соискание ученой степени

кандидата технических наук

Москва – 2026

Работа выполнена в федеральном государственном бюджетном образовательном учреждении высшего образования «Московский Авиационный Институт (национальный исследовательский университет)».

Научный руководитель:	**Брехов Олег Михайлович** доктор технических наук, профессор, заслуженный деятель науки РФ
Официальные оппоненты:	**Романов Александр Юрьевич** доктор технических наук, доцент, профессор департамента компьютерной инженерии Московского института электроники и математики им. А.Н. Тихонова Национального исследовательского университета «Высшая школа экономики» **Заева Маргарита Анатольевна** кандидат технических наук, доцент, доцент кафедры №12 «Компьютерные системы и технологии» Федерального государственного автономного образовательного учреждения высшего образования Национальный исследовательский ядерный университет «МИФИ»
Ведущая организация:	Федеральное государственное унитарное предприятие «Научно-исследовательский институт «Квант»

Защита состоится «28» мая 2026 г. в 15.00 на заседании диссертационного совета 24.2.331.19 при МГТУ им. Н.Э. Баумана по адресу: 105005, Москва, ул. 2-я Бауманская, д. 5, стр. 1, зал Ученого совета ГУК МГТУ им. Н.Э. Баумана.

С диссертацией можно ознакомиться в научно-технической библиотеке и на сайте http://bmstu.ru.

Отзывы на автореферат в двух экземплярах, заверенные печатью учреждения, просьба направлять по адресу: 105005, Москва, ул. 2-я Бауманская, д. 5, стр. 1, Ученому секретарю диссертационного совета Д 24.2.331.19, Сакулину С.А, кафедра ИУ3.

Автореферат разослан «_____» _____ 2026 г.

Ученый секретарь
диссертационного совета 24.2.331.19
кандидат технических наук, доцент С.А. Сакулин

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность диссертационного исследования. Программируемые логические интегральные схемы (ПЛИС) являются экономически выгодной и гибкой платформой для малосерийной разработки электронных устройств. Однако, по сравнению со специализированными и полузаказными микросхемами, они обладают меньшей сбоеустойчивостью, что требует применения соответствующих мер, особенно при создании систем для ответственных применений.

Существующие методы обеспечения сбоеустойчивости делятся на: технологические, зависящие от производителя ПЛИС, и архитектурные, основанные на введении аппаратной или временной избыточности. Исследованию различных аспектов обеспечения сбоеустойчивости и тестирования ПЛИС, а также систем на их основе посвящены труды таких исследователей, как: О.М. Брехов, Е. Ю. Данилова, А. А. Огурцов, А.Ю. Романов, К. К. Смирнов, Д. В. Тельпухов, G. Canivet, M.G. Gericota, I. G. Harris, H. Jahanirad, N. Kumar, S. Mukherjee, M. Rozkovec, Y. Sato, T. Vanat и др.

Анализ литературы показал, что ключевой практической проблемой является необходимость выбора оптимального архитектурного метода на ранних этапах проектирования, до завершения разработки основного функционала. Существующие методики выбора таких методов требуют создания специализированных тестовых систем и сложного программного обеспечения, что увеличивает время и стоимость разработки. Кроме того, процедуры тестирования самой ПЛИС и её окружения часто неоптимальны, так как предполагают разработку множества тестовых конфигураций и обладают проблемами, связанными с масштабируемости и определением множественных сбоев.

В этой связи разработка универсальной тестовой системы для решения двух задач – сравнительного анализа методов сбоеустойчивости и тестирования ПЛИС – позволит ускорить создание целевого устройства и оптимизировать выбор его защитных механизмов. В качестве эффективной основы для такой системы предлагается использование конвейеризированных алгоритмов генерации контрольных кодов. Таким образом, тема диссертации «Применение конвейеризированных генераторов контрольных кодов для проведения анализа сбоеустойчивости систем на основе ПЛИС» является актуальной.

Объект исследования – программируемые логические интегральные схемы (ПЛИС).

Предмет исследования – методика тестирования и выбора архитектурных решений сбоеустойчивости ПЛИС, основанная на использовании конвейеризированных генераторов контрольных кодов.

Цель диссертационной работы состоит в создании и апробации единой методики тестирования и выбора архитектурных решений сбоеустойчивости ПЛИС, основанной на использовании конвейеризированных генераторов контрольных кодов.

Задачи диссертационной работы. Для достижения поставленной цели были

решены следующие основные задачи:

1. Проведение анализа существующих способов обеспечения сбоеустойчивости и определение требований к разрабатываемой методике.
2. Разработка методики выбора аппаратной платформы и архитектурных методов обеспечения сбоеустойчивости систем на базе ПЛИС.
3. Разработка методики тестирования ПЛИС и их окружения.
4. Разработка единой методики тестирования ПЛИС и выбора аппаратной платформы ПЛИС.
5. Апробация единой методики разработки ПЛИС.

Методы исследования. Для решения поставленных задач были использованы методы теории вероятностей, теории графов, аналитического исследования, имитационного моделирования, разработки функциональных моделей тестовых систем и программного обеспечения для обработки результатов моделирования.

Научные результаты и их новизна. В диссертационной работе получены и обоснованы следующие результаты:

1. На основе изучения современных тестовых систем ПЛИС и методов их построения предложен метод создания единой тестовой системы, основанной на генераторе контрольного кода. В отличие от ранее известных систем предложенный метод позволяет провести тестирование микросхемы, а также отработать и проверить решения, связанные с выбором архитектуры и аппаратной платформы создаваемой системы, на ранних этапах разработки до получения целевого функционального описания. Особенностью тестовых систем, основанных на генераторе контрольного кода, является простота разработки и возможность выбора контрольного кода для достижения характеристик, максимально близких к характеристикам целевой системы.
2. Разработана методика выбора аппаратной платформы и архитектурных методов обеспечения сбоеустойчивости систем на базе ПЛИС, использующая единую тестовую систему, основанную на генераторе контрольного кода. Данная методика позволяет выполнять оценку изменения основных характеристик системы, основанной на ПЛИС, в результате имплементации различных методов обеспечения надежности с учетом особенностей всех доступных аппаратных платформ и выбирать наиболее подходящие по характеристикам.
3. Разработана методика тестирования ПЛИС и их окружения, использующая единую тестовую систему, основанную на генераторе контрольного кода. Данная методика позволяет проверять работу ПЛИС, ее системного окружения (подсистема питания, каналы ввода-вывода данных, генераторы глобальных сигналов и т.д.) с использованием упомянутой ранее единой тестовой системы для выбора аппаратной платформы, а также инструментального программного обеспечения, применяемого в процессе создания конфигурационного файла ПЛИС.

Соответствие диссертации паспорту научной специальности. Проблематика диссертации и полученные в ней результаты соответствуют паспорту специальности 2.3.2 – «Вычислительные системы и их элементы» по

пунктам 3 («Разработка научных подходов, методов, алгоритмов и программ, обеспечивающих надежность, сбое- и отказоустойчивость, контроль и диагностику функционирования вычислительных систем и их элементов»), 4 («Теоретический анализ и экспериментальное исследование функционирования вычислительных систем и их элементов в нормальных и экстремальных условиях с целью улучшения их технико-экономических и эксплуатационных характеристик») и 6 («Разработка научных подходов и методов, архитектурных и структурных решений, обеспечивающих эффективную техническую реализацию аппаратно-программных систем и комплексов за счет оптимизации применяемой электронной компонентной базы, элементов вычислительных систем и встраиваемого программного обеспечения».

Теоретическая и практическая значимость результатов работы. В диссертационной работе получены и обоснованы следующие результаты:

1. Разработана и прошла апробацию единая методика тестирования ПЛИС, которая позволяет оценивать характеристики сбоеустойчивых систем ПЛИС, увеличивать скорость и эффективность процессов верификации и тестирования до завершения разработки целевого функционального описания.
2. Разработаны и прошли апробацию тестовые системы на основе конвейеризированных генераторов CRC и MD4.
3. Полученные в диссертационной работе результаты были внедрены в производственную и научно-исследовательскую деятельность компаний АО «Крафтвэй», ООО «ПК «АКВАРИУС» и федерального государственного бюджетного образовательного учреждения высшего образования Московский Авиационный Институт (национальный исследовательский университет)».

Основные положения, выносимые на защиту:

1. Метод создания единой тестовой системы ПЛИС и выбора аппаратной платформы ПЛИС на основе конвейеризированных генераторов контрольных кодов.
2. Методика выбора аппаратной платформы и архитектурных методов обеспечения сбоеустойчивости систем на базе ПЛИС с использованием единой тестовой системы.
3. Методика тестирования ПЛИС и их окружения с использованием единой тестовой системы.
4. Единая тестовая система ПЛИС, созданная на основе применения конвейеризированных генераторов контрольных кодов.

Апробация работы. Результаты диссертационной работы докладывались и обсуждались на научно-технических конференциях:

1) «Научно-практическая конференция студентов и молодых ученых МАИ 2010», Москва, 2010 г.;

2) 18-я Международная конференция «Авиация и космонавтика», Москва, 2019 г.;

3) 19-я Международная конференция «Авиация и космонавтика», Москва, 2020 г.;

Публикации и личный вклад автора. Все теоретические и практические результаты получены автором лично. Функциональные описания, программы,

использованные для тестирования разработанных систем и для анализа результатов, разработаны автором лично. По материалам диссертационной работы опубликовано 5 научных статей общим объемом 5,3 п. л., в том числе 2 статьи в изданиях из списка, рекомендованного ВАК РФ, и 3 статьи в изданиях, индексируемых в наукометрической базе данных Scopus.

Достоверность полученных результатов. Достоверность подтверждена теоретическими выкладками и результатами практического применения.

Объем и структура работы. Диссертационная работа состоит из четырех глав, введения, заключения и приложения. Полный объем диссертации составляет 192 страницы, включая 34 рисунка и 31 таблицу; объем приложения – 30 страниц. Список литературы включает 157 наименований.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во **введении** показана актуальность ПЛИС как основы для разработки сбоеустойчивых вычислительных систем, обоснована актуальность темы тестирования таких систем и выбора наиболее эффективного метода обеспечения сбоеустойчивости. Далее сформулированы цель и задачи работы – исследование существующих методов обеспечения сбоеустойчивости и их выбора, разработка единой методики тестирования и выбора аппаратной платформы, а также метода обеспечения сбоеустойчивости; перечислены методы исследования, раскрыта научная новизна, теоретическая и практическая значимость работы, а также положения, выносимые на защиту.

В **первой главе** рассмотрены причины возникновения сбоев в микросхемах, работающих в условиях негативных внешних воздействий, общие принципы построения ПЛИС, особенности архитектуры ПЛИС в зависимости от производителя и приведены методы обеспечения сбоеустойчивости систем на базе ПЛИС. Далее определены критерии выбора наиболее эффективного метода обеспечения надежности и способы поиска такого метода. Также в первой главе проведен анализ основных особенностей ряда систем, которые подходят для тестирования микросхемы, ее окружения и выбора наиболее эффективного способа обеспечения сбоеустойчивости; сформулированы требования к тестовой системе (она должна: корректно проходить этапы синтеза, трассировки и генерации конфигурационного файла для целевой ПЛИС; обеспечивать максимальную чувствительность к сбоям и отказам, в том числе многократным; обеспечивать быстрое изменение степени логической загрузки ПЛИС без значительных исправлений исходного кода; обеспечивать соответствие между предполагаемым значением занятых логических ресурсов и значением, полученным с помощью оценки синтезатора и трассировщика; давать возможность вычислять значения, получаемые в процессе работы тестового устройства, заранее, и, соответственно, сверять их с данными, полученными в результате работы исследуемой ПЛИС). Кроме того, в главе проведен критический анализ существующих подходов к обеспечению сбоеустойчивости с целью выбора наиболее эффективного.

Во **второй главе** предложена методика тестирования ПЛИС с помощью

тестовой системы, основанной на конвейеризированных функциях. Наиболее полно рассмотренным в главе 1 требованиям к тестовым системам соответствуют конвейерные системы, в которых каждая стадия состоит из набора логических элементов, реализующих выбранную синтезируемую функцию, и регистра для хранения вычисленного значения. Такие системы хорошо масштабируются путем увеличения количества стадий конвейера, позволяют не допускать усложнения логических функций, реализованных на каждой из ступеней. Также преимуществом конвейерных систем с повторяющимися функциями является их регулярная структура, что значительно упрощает работу по размещению логической нагрузки внутри ПЛИС.

В качестве логической нагрузки может быть выбран любой алгоритм, позволяющий выполнять побитную обработку входных данных, например, хэш-функции или алгоритмы генерации самокорректирующихся кодов. Значения обрабатываемых бит могут быть заданы: константами; значениями, установленными на входах ПЛИС (статическими или динамическими); результатом работы внутренних блоков ПЛИС (в том числе блоков самотестирования).

Конвейеризация алгоритма — это представление циклических алгоритмов в виде конвейерной структуры, в которой на каждой ступени реализуется очередная итерация (или часть итерации) цикла алгоритма генерации контрольного кода. Таким образом, ступень с номером i получает промежуточные данные и необходимые признаки от i–1 ступени, после чего в соответствии с выбранным алгоритмом выполняет вычисления на основе полученных данных и значения i-го бита входного потока. Вычисленные промежуточные значения и признаки передаются на i+1-ю стадию. Данные с последней стадии передаются на выходы тестовой системы. После окончания разгона такого конвейера на каждом такте работы тестовой системы на выход передается очередное значение контрольного кода.

Анализ возможности объединения в одной системе вышеперечисленных принципов (конвейеризация, использование контрольных кодов), а также требований к тестовой системе позволил предложить метод создания единой тестовой системы на основе использования конвейеризированного контрольного кода. Такая тестовая система принимает данные от входов ПЛИС и внутренних блоков и использует их в качестве исходных данных для вычисления контрольного кода. Генератор контрольного кода в такой системе конвейеризирован и после окончания разгона конвейера выдает очередное значение контрольного кода на каждом такте.

Все множество двоичных значений, поступающих на входы тестовой системы за все время тестирования, будем называть входным потоком. Тогда эталонный входной поток – множество двоичных значений, которые должны быть переданы на входы тестируемой системы (со входов ПЛИС или в результате корректной работы внутренних узлов тестируемой подсистемы). Фактический входной поток – множество двоичных значений, принятых устройством или полученных от внутренних узлов во время теста. Соответственно, тестовая система вычисляет контрольный код для фактического

входного потока. Сбои в работе устройства приводят к появлению отличий между эталонным и фактическим входными потоками и/или искажениям в контрольном коде. Сравнение полученных результатов выполняет внешний по отношению к тестовой системе блок – анализатор, результатом работы которого является признак наличия ошибки или синдром ошибки.

Общий вид такой тестовой системы приведен на Рис. 1 .

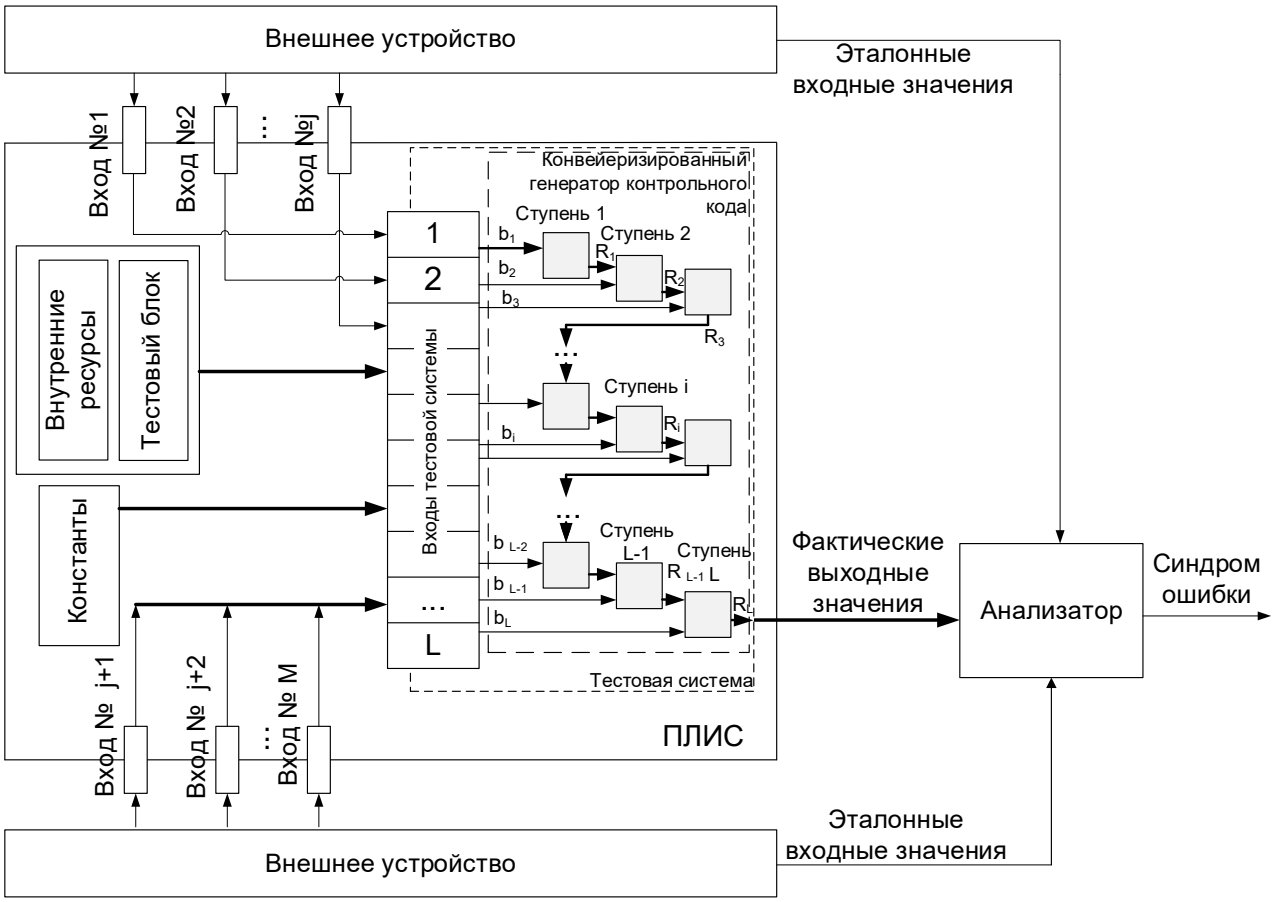

Рис. 1 Общий вид тестовой системы

Характерной особенностью конвейерных систем является обновление значений в запоминающих элементах на каждом такте работы устройства. Таким образом, при появлении сбоя в тестовой системе через несколько тактов изменится выходное значение, а после его исправления значение на выходах вернется к ожидаемому.

Среди требований, предъявляемых к тестовой системе, может быть требование не только обнаруживать сбой, но и определять место его возникновения. Для решения этой задачи предлагается создание тестовой прошивки на основе конвейеризированного алгоритма вычисления самокорректирующихся кодов.

Предлагаемая методика тестирования состоит из следующих этапов.

1. Определение требуемых характеристик тестовой системы.
2. Выбор алгоритма вычисления контрольного кода.
3. Определение способа генерации и значения бит входного потока.
4. Разработка тестовой системы на основе выбранного алгоритма и имплементация ее в ПЛИС.

5. Вычисление (аналитически или с помощью системы моделирования) контрольных бит для эталонного входного потока.
6. Тестирование системы с помощью полученного ранее эталонного потока и получение с выходов тестируемой системы контрольных бит для фактического входного потока.
7. Сравнение контрольных бит, вычисленных для эталонного и фактического входных потоков. В случае расхождения:
7.1. Для генераторов контрольной суммы: фиксация факта сбоя.
7.2. Для генераторов самокорректирующихся кодов:

- подстановка контрольных бит, полученных от тестовой системы, на место контрольных бит в эталонном коде;
- вычисление синдрома ошибки для полученного кода;
- на основании вычисленного синдрома ошибки определение номера ступени конвейера, на которой произошел сбой.

8. Выводы о результатах тестирования.

Далее в работе приводятся примеры тестовых систем на основе конвейеризированных генераторов контрольных кодов.

Тестовая система на основе конвейеризированного генератора CRC. В тестовой системе на основе конвейеризированного генератора CRC каждая ступень конвейера представляется в виде регистра разрядностью 32 бита, логического элемента «сложение по модулю 2», мультиплексора, а порождающий полином задается константой. Входной поток представляется в роли входных данных для алгоритма.

Количество бит входного потока, поступающих в систему на каждом такте, равно количеству ступеней конвейера. Во время работы тестовой системы на очередном такте каждая ступень конвейера получает промежуточное значение контрольного кода, определяет и выполняет текущее действие над полученным промежуточным значением и соответствующим битом входного потока, после чего очередное промежуточное значение передается на следующую стадию. Так как представленная схема является альтернативный реализацией схемы генерации CRC, то она сохраняет все свойства исходной.

Тестовая система была реализована на SystemVerilog и синтезирована в базис ПЛИС от трех различных производителей. Проведенное тестирование (в том числе с внесением одиночных и множественных сбоев) и последующий синтез подтвердили выполнение требований, предъявленных к тестовой системе.

Тестовая система на основе конвейеризированного генератора MD4. По аналогии с тестовой системой на базе CRC была разработана тестовая система на основе хэш-кода MD4. Такая тестовая система обладает более близкими к реальным устройствам характеристиками, чем тестовая система на основе CRC, т.к. соотношение затрат на комбинационную логику и регистры у CRC примерно равно 1:1, а у MD4 (как и у реальных устройств) – 2:1. Тестовая система была реализована на SystemVerilog и синтезирована в базис ПЛИС от трех различных производителей. Проведенное тестирование (в том числе с внесением одиночных и множественных сбоев) и последующий синтез подтвердили выполнение требований, предъявленных к тестовой системе.

Тестовая система на основе конвейеризированного генератора кода Хэмминга. Использование CRC или MD4 в качестве основы тестовой системы обеспечивает выявление сбоев, но не позволяет определить место их возникновения. Для решения задачи определения количества и места возникновения сбоев предлагается использование конвейеризированных генераторов кодов, обнаруживающих и исправляющих ошибки.

Выбранный алгоритм генерации контрольных кодов преобразуется способом, аналогичным приведенному выше. Между ступенями конвейера передается промежуточное значение контрольного числа и служебная информация. Результатом работы этого конвейера является корректирующий код, состоящий из двух частей: информационной части (равна фактическому входному потоку) и контрольного кода. Для определения факта наличия сбоя и его места достаточно знать только значение контрольного кода и эталонный входной поток, поэтому каким-либо образом сохранять или передавать за пределы тестовой системы фактический входной поток нет необходимости.

В соответствии с алгоритмом, приведенным ранее, для выявления сбоя и его места сгенерируем код, состоящий из эталонного входного потока и полученной от устройства корректирующей части. Далее, в соответствии с выбранным алгоритмом, вычислим синдром ошибки. Полученный синдром ошибки указывает на место возникновения сбоя в тестируемой системе. Возможные сбои в тестовой системе могут возникнуть во входном потоке или в самом тестовом конвейере. Если синдром ошибки указывает на контрольные биты, то это говорит о наличии сбоя в самом тестовом конвейере. В противном случае имеется расхождение между эталонным и фактическим входными потоками.

В диссертации рассмотрена реализация конвейера с развернутым алгоритмом генерации кода Хэмминга, выявляющего две ошибки и исправляющего одну ошибку. Каждая из обрабатывающих стадий конвейера в такой тестовой системе обрабатывает один бит входного потока. Ступени конвейера с номерами $0,1,2,4,8\ldots2^n$ не выполняют никаких действий (необрабатывающие стадии) и представляют собой регистр. Для обеспечения выявления двойных сбоев вводится «бит двойного контроля» (ДК), который складывается по модулю 2 с очередным входным значением на каждой из обрабатывающих стадий.

Тестовая система была реализована на SystemVerilog и синтезирована в базис ПЛИС от трех различных производителей. В результате проведенной серии экспериментов подтверждено выполнение приведенных выше требований, предъявляемых к тестовой системе. Проведенное тестирование подтвердило выполнение требований детекции факта появления двойного сбоя и обнаружения места возникновения одиночного сбоя, предъявленные к тестовой прошивке.

В **третьей главе** диссертации предложена методика выбора аппаратной платформы и архитектурных методов обеспечения сбоеустойчивости систем на базе ПЛИС. Подробно рассмотрен первый (предварительный) этап и продемонстрировано его применение. Задачу выбора по заданным критериям способа обеспечения сбоеустойчивости можно разделить на два этапа:

предварительный, на котором отсекаются заведомо неподходящие способы обеспечения сбоеустойчивости; основной, на котором среди оставшихся способов обеспечения сбоеустойчивости выбирается наиболее эффективный по заданным критериям.

К критериям, по которым производится оценка каждого из анализируемых методов обеспечения сбоеустойчивости, относятся: количество исправляемых сбоев; количество обнаруживаемых сбоев; количество ресурсов, занятых вычислительной системой (площадь занятой части микросхемы); максимальная частота работы вычислительной системы и т. д. Для оценки производится вычисление вероятности неправильной работы тестовой системы в данном аппаратном базисе в заданных условиях. Цель предварительного этапа выбора наиболее эффективного способа обеспечения сбоеустойчивости – исключение из анализа заведомо неподходящих способов обеспечения сбоеустойчивости.

При выборе методики оценки необходимо учесть следующие факторы: 1) от выбора способа обеспечения сбоеустойчивости системы зависит множество решений по построению системы в целом, поэтому определить список возможных решений нужно как можно раньше; 2) решается задача предварительного анализа без реальной интеграции анализируемого метода обеспечения сбоеустойчивости в разрабатываемую систему; 3) требуется минимизировать затраты времени на предварительный этап выбора; 4) решается, в первую очередь задача качественной оценки применимости анализируемого способа обеспечения сбоеусточивости. Если оценка покажет, что имеется значительный недостаток ресурсов, то такой способ обеспечения сбоеустойчивости исключается из дальнейшего анализа. В противном случае, если значения на грани допустимых или лучше, то такой способ следует исследовать на основном этапе.

Предлагается следующая методика быстрого анализа изменения характеристик системы.

1. Выбор тестовой системы на основе конвейеризированного генератора контрольных кодов.

2. Имплементация в тестовую систему анализируемого метода обеспечения надежности.

3. Определение относительного изменения площади $k_{prot_{impl}} = \frac{S_{testsystemprot}}{S_{testsystembase}}$, где $S_{testsystemprot}$ – площадь защищенной тестовой системы; $S_{testsystembase}$ – площадь незащищенной тестовой системы.

4. Определение площади незащищенной целевой системы (S_{target_base}) (аналитически или экспериментально).

5. Вычисление предполагаемой площади защищенной целевой системы как:

$$S_{targetprot} = k_{prot_{impl}} \times S_{targetbase}$$

6. Оценка временных характеристик.

Основными преимуществами предлагаемой методики являются: быстрота и скорость оценки (так как конвейеризированные генераторы контрольных кодов имеют регулярную структуру, то имплементация анализируемой системы

обеспечения сбоеустойчивости менее трудоемка) и возможность проведения оценки без готовой целевой системы.

В случае проведения оценки без готовой целевой системы, её площадь определяется аналитически. Предлагается оценка на основе соотношения количества конфигурируемых блоков, используемых для реализации регистров и для реализации комбинационной логики между ними. В зависимости от системы это соотношение находится в диапазоне от 70:30 до 40:60. Тогда, зная количество регистров в системе, которое может быть оценено после определения алгоритма работы системы или разработки модели на языке высокого уровня, и соотношение площадей регистров и логических элементов, можно вычислить площадь всей системы. Поскольку при работе от одного синхросигнала максимально возможная частота работы системы определяется частотой работы самого медленного из фрагментов, задача анализа изменения временных характеристик сводится к поиску наименьшего значения из рабочих частот защищенной тестовой системы, незащищенной целевой системы, защищенных фрагментов, используемых в защищенной целевой системе. Оценка проводилась для трех ПЛИС: Microsemi (Actel) APA 1000 CQFP352, Altera (Intel) Stratix EP1S25F672C, Xilinx Virtex XQR4VSX55CF1140.

Применение методики продемонстрировано на примере имплементации системы обеспечения сбоеустойчивости, основанной на использовании кода Хэмминга, исправляющего одну ошибку и обнаруживающего две ошибки. Защищаемым элементом в такой системе являются регистры. В состав защищенного регистра входят: кодер, декодер, регистр для хранения защищенных данных, мультиплексор и цепь самовосстановления. При отсутствии данных для записи регистр на каждом такте перезаписывает хранящиеся данные (каждый раз осуществляется кодирование и декодирование слова данных), чем обеспечивается самовосстановление записанных данных.

В качестве тестового устройства был использован блок форматирования входных потоков данных. Он (в соответствии с заданным режимом) собирает данные с двух источников (данные от одного из них буферизируются) и передает их на выход. Модель устройства разработана на языке SystemVerilog; в его состав входят конечный автомат, арифметические и логические элементы, а также регистры разрядностью от 1 до 64 бит. Устройство относится к устройствам общего назначения, и для большинства таких систем наиболее вероятным соотношением между площадью логических элементов и регистров является 2:1. С учетом приведенных выше требований для дальнейшего анализа предлагается использовать конвейеризированный генератор хэш-кода MD4, который (по результатам проведенных экспериментов) имеет аналогичное соотношение затрат на реализацию комбинационных функций и регистров. Разрядности защищаемых частей регистров тестовой системы должны выбираться таким образом, чтобы примерно сохранять соотношение между разрядностями регистров в целевой системе. В каждой степени разработанного конвейера используются (согласно требованиям) следующие защищенные регистры: 32-х разрядный, 16-разрядный и два одноразрядных.

Для получения точного значения аппаратных затрат на реализацию целевой системы в соответствующем аппаратном базисе нужно синтезировать ее функциональное описание на языке описания аппаратного обеспечения. Для вычисления аппаратных затрат, необходимых для реализации защищенной системы и оценки частоты ее работы, были проведены эксперименты с синтезом защищенных и незащищенных регистров разной разрядности. В данном случае это регистры с разрядностями от 1 до 64.

Для оценки погрешности проведенного эксперимента анализируемая система обеспечения надежности была имплементирована в целевую систему и выполнен ее синтез. Далее были вычислены относительные погрешности для всех проведенных экспериментов.

Средняя относительная погрешность составила 9,9% (для оценки на основе аналитически определенного размера целевой системы) и 6,5% (для оценки на основе фактического размера целевой системы). Далее была проведена оценка погрешности определения частоты работы системы. Средняя относительная погрешность, полученная в результате эксперимента, составила 8,76%. По результатам предварительного этапа анализа был сделан вывод о том, что все анализируемые системы обеспечения сбоеустойчивости могут быть реализованы в любой из трех ПЛИС.

В **четвертой главе** диссертации представлен второй этап предлагаемой методики выбора аппаратной платформы и архитектурных методов обеспечения сбоеустойчивости систем на базе ПЛИС, основанной на применении конвейеризированных генераторов контрольных кодов. Исходными данными для такого анализа являются: функциональное описание тестовой системы на языке описания аппаратуры; программно-аппаратный комплекс, позволяющий выполнить синтез функционального описания в базисе выбранных микросхем; данные о сбоеустойчивости микросхемы.

Описание методики проведения анализа. Анализ архитектурных методов обеспечения сбоеустойчивости проводится для выявления лучшего (по заданным критериям) метода. В данной работе критерием эффективности была выбрана вероятность появления сбоев, не исправленных полученной системой. Системы делятся на два основных типа: незащищенные и защищенные. Незащищенными называем исходные системы до применения исследуемых методов обеспечения сбоеустойчивости. Соответственно, защищенные системы – это системы с применением анализируемых методов обеспечения сбоеустойчивости. Анализ архитектурных методов обеспечения сбоеустойчивости включает в себя такие этапы, как: 1) выбор (разработка) исходной системы; 2) имплементация выбранного метода (методов) обеспечения сбоеустойчивости в исходную систему и проверка полученной системы на соответствие требованиям; 3) оценка сбоеустойчивости защищенной системы для каждого из методов, участвующих в анализе; 4) анализ полученных данных и выбор наиболее эффективного метода.

Алгоритм сравнения (и в случае выбора аппаратной платформы, и в случае выбора метода обеспечения сбоеустойчивости) практически одинаков, поэтому далее рассматривался универсальный подход. Дальнейший анализ был проведен

с учетом ограничений, накладываемых ПЛИС: система должна занимать не большее количество ресурсов, чем доступно в данной ПЛИС, и работать на частоте, не меньшей, чем заданная.

Основываясь на данных о сбоеустойчивости, полученных в результате испытаний, было вычислено среднее количество сбоев в системе или ее фрагменте за единицу времени. Единица времени – один такт работы системы. Учитывая общее количество ресурсов (элементов) в рассматриваемом фрагменте, максимальное количество сбоев, которое может быть исправлено выбранным методом обеспечения сбоеустойчивости, и интенсивность сбоев, можно сделать вывод о вероятном количестве неисправленных ошибок в этом фрагменте за заданный промежуток времени.

В рассмотренном фрагменте тестовой системы, состоящем из защищаемого логического блока и дополнительных элементов, применяемых для реализации выбранного метода обеспечения сбоеустойчивости (такой фрагмент – защищенный, а защищаемый логический блок – базовый фрагмент), n – количество сбоев, исправляемых выбранной системой обеспечения сбоеустойчивости; пусть за время t в этом фрагменте произошло N_{frag} сбоев; тогда при $n<N_{frag,}$ система не сможет исправить сбои, и искаженное значение будет передано за пределы текущего фрагмента в последующие блоки; такой сбой может быть детектирован средствами самоконтроля системы или внешними устройствами.

Вероятность появления k сбоев подчиняется экспоненциальному закону распределения, т.е. $p_k(t) = \frac{(\lambda \times t)^k}{k!} e^{-\lambda * t}$, и имеет интенсивность сбоев $\lambda = q \times \sigma \times F$, где F [частиц/с×см^2] – поток частиц, т.е. количество частиц, прошедших через анализируемую область за единицу времени; σ [см^2/бит] – сечение сбоев, которое характеризует вероятность того, что частица, попавшая в устройство, изменит один бит; q – количество элементов в анализируемом фрагменте, подверженных сбоям. Зная эти параметры, можно определить вероятность появления сбоев в фрагменте микросхемы, который обеспечивает парирование на каждом такте не более n сбоев:

$$p\ \ (t) = 1 - \sum_{k=0}^{n-1} \frac{(\lambda \times t)^k}{k!} e^{-\lambda \times t}. \quad (1)$$

Аналогично можно определить вероятность

$$p_{syst} = 1 - \prod_{i=1}^{L} (1 - p_{err\ i}) = 1 - (1 - p_{err})^L \quad (2)$$

появления сбоев в системе из L таких фрагментов (одинаковых, и, соответственно, с одним и тем же $p_{err\ i}$). Таким образом, задача поиска наиболее эффективного метода обеспечения сбоеустойчивости сводится к поиску наименьшего значения p_{syst} из множества всех вычисленных в процессе анализа.

В общем случае: $p_{syst} = f(t, q, \sigma, F)$, где σ характеризует используемую аппаратную платформу, q – аппаратную платформу и используемый метод обеспечения сбоеустойчивости, F – воздействие внешних факторов, t определяется требованиями к системе, характеристиками системы и

характеристиками аппаратной платформы. В данной работе F и t – константы, заданные в требованиях к системе. Изменяемыми параметрами являются аппаратная платформа и используемый метод обеспечения сбоеустойчивости. Для каждой системы можно определить параметр $N_{cell\_clock} = t_{clock} \times \sigma \times F$ – среднее количество сбоев в ячейке за один такт.

Сравнение систем обеспечения сбоеустойчивости. В диссертации был проведен анализ четырех систем, основанных на конвейеризированном генераторе MD4: 1) без дополнительных средств обеспечения сбоеустойчивости (незащищенная система); 2) защищенная система с мажорированием (трехкратная избыточность и побитное голосование «2 из 3») на уровне отдельных ступеней конвейера; 3) защищенная система с использованием кода Хэмминга для защиты регистров в ступенях конвейера; 4) защищенная система с полным мажорированием конвейера (трехкратная избыточность и побитное голосование «2 из 3»). В рассматриваемых системах длительность цикла исправления сбоев равна длительности одного такта; соответственно, необходимо определить вероятность сбоя в исследуемых системах при $t = t_{clock}$.

Незащищенная система. В данной системе рассматриваемый фрагмент – одна ступень конвейера. В этой системе сбои не исправляются, т.е. неправильный результат появится уже после одного сбоя. Предполагая, что система имеет L ступеней, занимает $q_{systbase}$ аппаратных ресурсов, одна ступень занимает q_{stbase}, а $N_{cell\_clock} = t_{clock} \times \sigma \times F$ – среднее количество сбоев в ячейке за один такт, вероятность сбоя:

$$p_{base} = 1 - \left(1 - \left(1 - e^{-q_{stbase} \times N_{cell\_clock}}\right)\right)^{L}. \quad (3)$$

Поблочное мажорирование. Рассматриваемый фрагмент – одна ступень конвейера; $q_{syst_{MJR}}$ – полное количество ресурсов; система содержит L стадий, каждая из которых занимает $q_{st_{MJR}}$; на вычисление одного бита приходится $\frac{q_{stmjr}}{3 \times w}$ ресурсов, задействованных для реализации ступени, где w – количество разрядов регистра в незащищенной системе; $3 \times w$ – количество разрядов в регистре с трехкратной избыточностью, тогда $\lambda_{MJR} = \frac{1}{3 \times w} \times q_{stMJR} \times \sigma \times F$. Вероятность появления сбоя в одном мажорированном бите: $p_{mjrbit}(t) = 1 - e^{\left(-\lambda_{MJR_{step}} \times t\right)}$. В таком случае вероятность появления сбоя в конвейере длиной L:

$$p_{mjr} = 1 - \left(1 - \left(1 - \left(3 \times \left(1 - e^{\left(-\frac{1}{3*w} \times q_{stMJR} \times N_{cell\_clock}\right)}\right)^{2} \times \left(e^{\left(-\frac{1}{3*w} \times q_{stMJR} \times N_{cell\_clock}\right)}\right) + \left(e^{\left(-\frac{1}{3*w} \times q_{stMJR} \times N_{cell\_clock}\right)}\right)^{3}\right)\right)^{w}\right)^{L}. \quad (4)$$

Защита на основе кода Хэмминга. Цикл исправления сбоев равен одному такту; $q_{systHamm}$ – полное количество ресурсов, занятое системой; q_{stHamm} – количество ресурсов, занятое одной защищенной стадией; L – общее количество стадий. Искаженное значение на выходе фрагмента тестовой системы на основе кода Хэмминга появится при появлении двух и более ошибок в одном блоке.

Интенсивность сбоев: $\lambda_{Hamm} = q_{stHamm} \times \sigma \times F$. Тогда вероятность появления сбоя в конвейере длиной L:

$$p_{Hamm} = 1 - \left(e^{-\lambda_{Hamm} \quad t} + \lambda_{Hamm} \times e^{-\lambda_{Hammp} \times t}\right)^{L}. \quad (5)$$

$$p_{Hamm} = 1 - \left(e^{-q_{stHamm} \times N_{cell\_clock}} + q_{stHamm} \times N_{cell\_clock} \times e^{-q_{stHamm} \times N_{cell\_clock}}\right)^{L}. \quad (6)$$

Полностью мажорированная система. Защищаемый фрагмент – один канал, состоящий из L ступеней конвейера. Каждое слово данных проходит через все ступени и после выхода из конвейера мажорируется. При трехкратной избыточности всего в системе будет $3 \times L$ ступеней. Размер одной ступени в таком случае: $q_{stsystmjr} = \frac{q_{fullsystmjr}}{3 \times L}$; $q_{fullsystmjr}$ – количество ресурсов, занятое всей системой. В таком случае вероятность появления сбоя в одной ступени одного из мажорируемых конвейеров:

$$p_{stsystmjr} = 1 - e^{-\left(q_{stsystmjr} \times N_{cell_{clock}}\right)}. \quad (7)$$

В системе, состоящей из трех мажорируемых конвейеров:

$$p_{systmjr} = 1 - \left(1 - 3 \times \left(\left(1 - 1 - e^{-\left(q_{stsystmjr} \times N_{cell\_clock}\right)}\right)^{L}\right)^{2} \times \left(1 - \left(1 - 1 - e^{-\left(q_{stsystmjr} \times N_{cell\_clock}\right)}\right)^{L}\right) + \left(\left(1 - 1 - e^{-\left(q_{stsystmjr} \times N_{cell\_clock}\right)}\right)^{L}\right)^{3}\right) . \quad (8)$$

*Сравнение систем***.** По результатам анализа полученных выше функций для различного количества стадий конвейера при N_{cell_clock} равных 10^{-2}, 10^{-3}, 10^{-4} можно сделать следующие выводы:

- вне зависимости от N_{cell_clock} наибольшую сбоеустойчивость демонстрирует система, использующая поблочное мажорирование;

- эффективность применения защиты на основе мажорирования на уровне системы и защиты с использованием кода Хэмминга зависит от значения N_{cell_clock} и количества ступеней в конвейере. Чем меньше количество сбоев в ячейке происходит за один такт, тем больше ступеней в конвейере может быть задействовано до того, как применение защиты на основе полносистемного мажорирования перестанет быть эффективным.

Сравнение значений, полученных в результате проведенной оценки и в результате эксперимента, показало, что предложенная методика позволяет быстро провести качественную оценку выбранного способа обеспечения сбоеустойчивости и принять решение о его реализуемости и необходимости дальнейшего исследования с целью имплементации в целевую вычислительную систему.

Применение конвейеризированных генераторов контрольных кодов для проведения анализа методов обеспечения сбоеустойчивости. В качестве объектов исследования были выбраны следующие микросхемы: Microsemi (Actel) APA 1000, Altera (Intel) Stratix и Xilinx Virtex 4 RT. Минимальным логическим элементом для ПЛИС является базовая ячейка. Размеры, функциональность и название базовой ячейки зависят от производителя и серии микросхемы. В качестве элемента микросхемы, подверженного сбоям, в последующих вычислениях использовалась базовая ячейка ПЛИС. В работе

приведен пример оценки сбоеустойчивости анализируемых систем в потоке тяжелых заряженных частиц для тестовых систем на базе CRC и MD4.

Лучшей по критериям сбоеустойчивости и реализуемости в данных условиях можно назвать систему на основе ПЛИС Xilinx с реализованным поблочным мажорированием. Следующими по критериям сбоеустойчивости и реализуемости из рассмотренных в работе являются аналогичная система на базе ПЛИС Actel (Microsemi) и системы на основе кода Хэмминга. К недостаткам последних можно отнести значительно меньшую максимальную тактовую частоту систем, защищенных таким образом. Системы на основе мажорирования на уровне системы имеют меньшую сбоеустойчивость, чем системы на основе поблочного мажорирования или на основе кода Хэмминга, но при этом занимают меньше ресурсов, оказывают меньшее влияние на максимальную частоту работы системы и превосходят по критерию сбоеустойчивости незащищенные системы. Рассмотренные системы на базе ПЛИС Altera (Intel) обладают меньшей сбоеустойчивостью, что закономерно – рассматриваемая ПЛИС не позиционируется как сбоеустойчивая и/или рекомендуемая для использования в условиях негативных внешних воздействий. Результаты анализа согласуются с выводами, опубликованными ранее.

Также был проведен анализ эффективности предложенной методики оценки по критерию трудоемкости разработки целевого устройства. Использование предложенной методики позволило сократить на 48,3% трудоемкость этапа разработки защищенной целевой системы и на 7,9% – трудоемкость всей RTL разработки проекта при анализе по модели COCOMO II.

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

1. Проведен анализ существующих способов обеспечения сбоеустойчивости ПЛИС и определены требования к единой методике тестирования и выбора аппаратной платформы и архитектурных методов обеспечения сбоеустойчивости систем на базе ПЛИС.

2 Разработана методика использования конвейеризированных генераторов контрольных кодов для тестирования сбоеустойчивых систем ПЛИС и устройств на их основе.

3. Разработана методика использования конвейеризированных генераторов контрольных кодов для выбора аппаратной платформы и архитектурных методов обеспечения сбоеустойчивых систем ПЛИС и устройств на их основе, состоящая из предварительного и основного этапов. Особенностью методики является возможность использования ее на разных этапах разработки системы, включая этап тестирования окружения ПЛИС.

4. Экспериментально доказана эффективность разработанной методики для создания сбоеустойчивых систем ПЛИС и подтверждено уменьшение трудоемкости проекта при использовании предложенной методики в процессе разработки устройств на базе ПЛИС.

5. Разработаны специализированные тестовые системы на базе конвейеризированных генераторов кодов CRC и MD4 и описано их применение для разработки сбоеустойчивых систем и тестирования устройств на базе ПЛИС.

6. Проведено сравнение изменений характеристик защищенной системы в зависимости от используемой аппаратной платформы.

7. Проведено сравнение способов обеспечения сбоеустойчивости с использованием предложенной методики. Полученные результаты согласуются с выводами других исследований.

Внедрение результатов представляемого диссертационного исследования способствовало ускорению и упрощению процесса разработки устройств на базе ПЛИС за счет сокращения затрат на разработку отдельных тестовых систем и раннего выбора наиболее эффективного метода обеспечения сбоеустойчивости.

Основные публикации по теме работы:

1. Брехов О.М, Ратников М.О. Анализ архитектурных методов обеспечения сбоеустойчивости систем, основанных на ПЛИС, посредством применения конвейеризированных функций / О.М. Брехов, М.О. Ратников // Известия высших учебных заведений. Поволжский регион. Технические науки. 2015. № 2 (34). С. 56–68. (0,5 п. л. / 0,3 п. л). Личный вклад: разработка методики и тестовых систем, проведение сравнения и анализ полученных результатов.
2. Брехов О.М, Ратников М.О., Сравнительный анализ тестовых систем ПЛИС и их окружения / О.М. Брехов, М.О. Ратников // Труды МАИ. 2022. № 125. DOI: 10.34759/trd-2022-125-22. (1,5 п. л / 1,2 п. л.). Личный вклад: сбор материалов, классификация, анализ тестовых систем и оценка использованных подходов.
3. Brekhov O., Ratnikov M. Pipelined Error-detecting Codes in FPGA Testing / O. Brekhov, M. Ratnikov // Advances in Electrical and Computer Engineering. 2014, Vol. 14. № 2. P. 57–62. DOI:10.4316/AECE.2014.02010. (0,5 п.л. / 0,4 п.л.). Личный вклад: разработка и обоснование концепции тестовой системы на основе конвейеризированного генератора контрольных кодов, разработка тестовой системы и оценка ее применимости.
4. Brekhov O., Klimenko A., Kordover K., Ratnikov M. FPGA-Prototyping with Advanced Fault Injection Methodology for Tolerant Computing Systems Simulation / O. Brekhov, A. Klimenko, K. Kordover, M. Ratnikov // Distributed Computer and Communication Networks. DCCN 2015. Communications in Computer and Information Science. 2016. Vol. 601. P. 208–223. DOI:10.1007/978-3-319-30843-2_22. (1 п. л. / 0,2 п. л.). Личный вклад: обзор модели угроз, определение наиболее чувствительных к внешнему воздействию элементов системы и методов моделирования сбоев в таких элементах.
5. Brekhov O., Ratnikov M., Using pipelined control code generator as a reference system in FPGA’s selection as a hardware platform for fault-tolerance computing system / O. Brekhov, M. Ratnikov // Journal of Physics: Conference Series. 2021. Vol. 1925. DOI:10.1088/1742-6596/1925/1/012038. (0,25 п. л. / 0,2 п. л.). Личный вклад: разработка методики применения тестовой системы на основе конвейеризированного генератора контрольного кода, разработка исследуемых тестовых и целевой систем, проведение сравнения и анализ результатов.