На правах рукописи

Эннс Виктор Иванович

Методы и средства разработки специализированных гетерогенных конфигурируемых интегральных схем для вычислительной техники и систем управления

Специальность: 05.13.05 - элементы и устройства вычислительной техники и систем управления

Специальность: 05.13.12 - системы автоматизации проектирования

(технические науки)

Диссертация на соискание ученой степени доктора технических наук

Москва, 2022г.

Оглавление

Введени	1e			
Глава 1.	Анализ текущего состояния и направлений развития			
конфигу	урируемых специализированных интегральных схем			
1.1.	Обоснование роли гетерогенных конфигурируемых интегральных			
схем в р	ешении задачи комплектования ЭКБ специализированной			
аппарат	уры18			
1.2.	Текущее состояние и направления развития конфигурируемых			
интегра	льных схем			
1.2.1.	Обзор архитектурных решений и методов анализа структуры			
совреме	енных ПЛИС			
1.2.2.	Направления развития цифровых базовых матричных кристаллов 40			
1.2.3.	Существующие изделия, области применения и основные			
характе	ристики аналоговых блоков АЦ БМК и ПАИС 50			
1.3.	Выводы			
Глава 2.	Маршрут разработки, методы предварительного анализа и			
програм	имного прототипирования специализированных гетерогенных			
конфигу	урируемых интегральных схем. Оптимизация архитектуры ПЛИС 55			
2.1	Маршрут разработки специализированных гетерогенных			
конфигу	ируемых интегральных схем. Особенности выбора их			
	ональных олоков			
2.2 Критері	методы предварительного анализа конфигурируемых схем.			
конфигу	и и метрики оценки эффективности различных решении построения ирируемых схем			
23	Метод программного прототицирования конфигурируемых схем 72			
2.5	Новые эруитектурные решения. Аруитектура программируемой			
2.4 Повые архитектурные решения. Архитектура программируемой интеграцьной схемы типа «жук» 87				
2.5				
2.3.	93			
26	Buponu 94			
2.0. Глара 2				
гава 5. гетероге	математические модели для разработки конфигурируемых			
схем зан	сазчиков на их основе			
3.1	Терминология. Основные понятия и обозначения			
	• · · · · · · · · · · · · · · · · · · ·			

3.2 Теоретико-графовые модели и метрики. Иерархия проекта 10	0			
3.3 Теоретико-графовые модели и метрики. Технологическое				
отображение10	5			
3.4 Модели и метрики для решения задач оценки архитектуры базового				
кристалла на основе анализа схем заказчиков. Метрики для базового	~			
кристалла 11	0			
3.5 Модели и метрики для решения задач оценки архитектуры базового				
кристалла на основе анализа схем заказчиков. Метрики для схем заказчиков.	Д			
	т			
лля конфигурируемых гетерогенных схем в терминах обобщенной				
математической модели	4			
3.7 Выводы15	0			
Глава 4. Конструктивные и схемотехнические решени	R			
специализированных конфигурируемых интегральных схем повышенно	й			
надежности15	1			
4.1. Методы повышения отказоустойчивости и стойкости к воздействию				
неблагоприятных внешних воздействующих факторов конфигурируемых				
специализированных ИС на основе схем обнаружения неисправностей 15	1			
4.2. Методы повышения надежности конфигурационной и				
пользовательской памяти на основе помехоустойчивого кодирования с	7			
	/			
4.2.1. Актуальность применения корректирующих кодов для повышения 15	7			
122 Гранции сбластей неколирования а ниших блаковых колов на	'			
алгоритму Форни	9			
423 Апгоритм мягкого леколирования блоковых колов обеспечивающий	ź			
большую вероятность правильного декодирования, чем алгоритм Форни. 16	0			
4.2.4. Границы вероятности того, что сообщение не будет декодировано				
правильно	6			
4.2.5. Блоки памяти с самоконтролем17	1			
4.3. Схемотехнические методы повышения сбоеустойчивости элементов				
хранения конфигурируемых интегральных схем				
4.3.1. Варианты исполнения триггеров. Триггеры с элементами				
обнаружения сбоев	1			

4.3.2.	Результаты моделирования DICE-триггеров
4.3.3. в конфи	Результаты применения элементов памяти повышенной надежности гурируемых схемах
4.4.	Выводы
Глава 5. кристал	Особенности разработки перспективных базовых матричных лов и конфигурируемых аналого-цифровых интегральных схем 195
5.1. матричн	Методы построения интегральных схем цифровых базовых ных кристаллов
5.1.1. расшире	Разработка цифровых базовых матричных кристаллов с енными функциональными возможностями
5.1.2. матричн	Методы повышения экономической эффективности базовых ных кристаллов
5.2. конфигу	Методы разработки аналоговых и аналого-цифровых ирируемых интегральных схем
5.2.1. аналого компози	Принципы построения аналоговых ядер и на их основе аналоговых и -цифровых конфигурируемых интегральных схем. Декомпозиция и ция аналоговых СФ блоков
5.2.2. конденс источни	Упрощенные эквивалентные схемы резисторов на переключаемых аторах в z-области при включении их между виртуальной землей и ком сигнала
5.2.3. архитек аналого	Особенности схем заказчиков различных классов, влияющие на туру и характеристики базовых кристаллов конфигурируемых вых и аналого-цифровых интегральных схем
5.2.4. аналого	Разработка конфигурируемых специализированных аналоговых и -цифровых интегральных схем с оптимальной архитектурой для
заданно	го класса задач
5.3.	Выводы
Заключе	ение
Библиог	рафический список
Прилож	ение 1. Описание цифровых блоков БМК 283
Прилож предлож	ение 2. Характеристики типовых аналоговых схем с применением кенных моделей резисторов на переключаемых конденсаторах 312
Прилож аналого	ение 3. Электрические схемы аналоговых библиотечных элементов -цифрового БМК

Приложение 4. Аналого-цифровые базовые матричные кристаллы,	в которых
реализованы положения и результаты работы	
Приложение 5. Акты внедрения	

Список сокращений и толкований (глоссарий)

- АЦ БМК аналого-цифровой базовый матричный кристалл
- АЦП аналого-цифровой преобразователь
- БК базовый кристалл
- БМК базовый матричный кристалл
- БЯ базовая ячейка
- БЦЯ базовая цифровая ячейка
- ВВФ внешние воздействующие факторы
- ЗУ запоминающее устройство
- ЗЭ запоминающий элемент
- ИОН источник опорного напряжения
- ИС интегральная схема
- КМОП комплементарная логика на транзисторах металл-оксидполупроводник
- КНИ технология «кремний на изоляторе»
- ЛБ логический блок
- ЛЭ логический элемент
- НВВ неблагоприятное внешнее воздействие
- НИР научно-исследовательская работа
- ОЗУ оперативное запоминающее устройство
- ОУ операционный усилитель
- ОКР опытно-конструкторская работа
- ПА предварительный анализ
- ПАИС программируемая аналоговая интегральная схема
- ПАЦИС программируемая аналого-цифровая интегральная схема
- ПК переключаемый конденсатор
- ПЛИС программируемая логическая интегральная схема
- ПП программное прототипирование
- РПП регистр последовательного приближения
- РЭА радиоэлектронная аппаратура

- САПР система автоматизации проектных работ
- СБИС сверхбольшая интегральная схема
- СВЧ устройства устройства приема, обработки и передачи сверхвысокочастотного сигнала
- СнК система на кристалле
- СхЭ схема электрическая
- СФ блоки готовые сложно-функциональные блоки, выполняющие заданные функции, и используемые в качестве блоков микросхем
- ТЗЧ тяжелая заряженная частица
- ФАПЧ фазовая автоподстройка частоты
- ФШ фотошаблон
- ЦАП цифро-аналоговый преобразователь
- ЦОС цифровая обработка сигналов
- ЦПУ центральное процессорное устройство
- ШИМ широтно-импульсная модуляция
- ЭКБ электронная компонентная база
- ЭКБ ИП ЭКБ импортного производства
- ЭКБ ОП ЭКБ отечественного производства
- Базовый кристалл конфигурируемой интегральной схемы полупроводниковый кристалл или микросхема до этапа конфигурирования. Базовый кристалл содержит набор несоединенных функциональных элементов.

Гетерогенная конфигурируемая интегральная схема – ИС, основные функции которой выполняются конфигурируемыми и заказными СФ - блоками.

Конфигурируемая интегральная схема – ИС, функциональная схема которой определяется коммутационными цепями и элементами, состояния которых устанавливаются в соответствии со схемой заказчика на этапе конфигурирования. К конфигурируемым интегральным схемам относятся ПЛИС, ПАИС, ПАЦИС, БМК, АЦ БМК.

Специализированная интегральная схема – ИС, предназначенная для решения

определенного класса задач по функциональному назначению, по условиям эксплуатации, по конструктивному исполнению, или по иным критериям.

- Схема заказчика проект заказчика в виде электрической или структурной схемы заданного функционального назначения, предназначенный для реализации в заказной, полузаказной, или программируемой интегральной схеме.
- ASIC (application-specific integrated circuit) интегральная схема, спроектированная для решения конкретной задачи.

DSP (digital signal processor) – цифровой сигнальный процессор.

- DSP-блоки конфигурируемых микросхем СФ блоки аппаратной поддержки цифровой обработки сигналов в составе конфигурируемых интегральных схем на основе многоразрядных цифровых умножителей.
- LUT (look-up tables) комбинационная схема в составе логического элемента ПЛИС, логические состояния выходов которой определяются данными в ячейках памяти с адресом, заданным кодом входных сигналов.

Spice – компьютерная программа моделирования электронных схем

Введение

Диссертационная работа посвящена развитию методов и средств создания конфигурируемых интегральных схем, являющихся ключевыми элементами радиоэлектронных устройств. В работе исследуются методы разработки и применения микроэлектронных схем нового типа так называемых гетерогенных конфигурируемых интегральных схем, ориентированных на повышение эффективности за счет специализации и объединяющих в себе традиционно разделяемые стили проектирования: от полностью заказного, полузаказного до программируемых схем.

Решения, предложенные в работе, позволят добиться нового качества разработки и широкого применения конфигурируемых схем, улучшения их технических характеристик И снижения времени разработку на радиоэлектронной аппаратуры, приведёт повышению ЧТО К уровня безопасности государства за счет опережающего развития отечественной радиоэлектронной техники во многих отраслях промышленности.

Актуальность работы

Разработчикам отечественной аппаратуры в современных условиях необходима широкая номенклатура интегральных схем при небольших объемах их потребления. Задача создания полной номенклатуры электронной компонентной базы В PΦ трудновыполнима из-за ограниченности выделяемых на это средств. Решение проблемы заключается в широком применении программируемых и полузаказных интегральных схем, которые конфигурирования могут заменить большое за счет количество универсальных и заказных микросхем. На это указывают результаты экспертизы правильности применения импортной ЭКБ в проектах ГК «Роскосмос», которой систематизирована В рамках номенклатура потребляемых микросхем.

Конфигурирование интегральных схем предполагает наличие встроенных коммутационных ресурсов и блоков унифицированных

существенно логических аналоговых элементов, ЧТО снижает И эффективность использования площади кристалла. Улучшения эффективности можно достичь за счет гетерогенности, применяя в структуре конфигурируемых схем различные по функциональному назначению сложно-функциональные блоки (СФ блоки).

Дальнейшее повышение эффективности использования площади кристалла достигается интеграцией на одном кристалле не только заказных и программируемых блоков, но и полузаказных схем, то есть, размещением на кристалле программируемой гетерогенной схемы блоков базовых матричных кристаллов. Симбиоз схем, использующих различные стили проектирования, делает необходимым создание математического аппарата, моделей, алгоритмов и методов исследования и разработки таких схем, а также поиск новых технических решений.

Характеристики отечественных конфигурируемых ИХ CXEM, производительность функциональность И должны находиться на современном уровне, достигнутом крупными зарубежными компаниями. Такая задачи требует максимального учета специфики постановка отечественной аппаратуры, в которой будут применяться конфигурируемые схемы, что неизбежно приводит к их специализации. Актуальность задачи подтверждается тем, что аналогичные подходы, ориентированные на повышение эффективности за счет специализации и гетерогенности, исследуются в работах Агентства перспективных научно-исследовательских проектов в области обороны при Министерстве обороны США - DARPA (Defense Advanced Research Project Agency): DARPA CRAFT, DARPA SDH и других, используются в программах Европейского космического агентства.

Актуальным является поиск и совершенствование схемотехнических, архитектурных и конструктивных решений создания надежных и радиационно-стойких конфигурируемых схем специального назначения.

Объект исследования: конфигурируемые гетерогенные интегральные схемы, включая программируемые логические и аналого-цифровые

интегральные схемы (ПЛИС, ПАИС, ПАЦИС) и базовые матричные кристаллы (БМК, АЦ БМК), выполненные по проектным нормам отечественных микроэлектронных предприятий, как база для реализации интегральных схем конечного применения для систем целевого назначения.

Предмет исследования: структура, архитектура и схемотехника специализированных конфигурируемых гетерогенных интегральных схем, методы их разработки и настройки на функционирование в соответствии с задачами заказчика.

Цель диссертационной работы.

Целью работы является создание методов и средств разработки конфигурируемых интегральных схем нового типа гетерогенных ____ конфигурируемых интегральных схем, обладающих повышенной эффективностью использования площади кристалла И улучшенными характеристиками за счет специализации по областям применения и комбинации различных стилей проектирования, а именно: заказного, полузаказного - базовых матричных кристаллов и программируемых схем.

Задачи исследования.

В диссертационной работе исследуются методы разработки и применения гетерогенных конфигурируемых интегральных схем, а также используемые в методах математические модели и метрики.

Ввиду специфики отечественного рынка наибольший интерес представляют конфигурируемые микросхемы с повышенной надежностью функционирования в условиях неблагоприятного воздействия, методы разработки которых изучаются в работе.

В работе решаются следующие задачи.

 Исследование нового подхода к комплектованию специализированной аппаратуры на базе широкого применения гетерогенных конфигурируемых интегральных схем.

- Создание маршрута разработки гетерогенных конфигурируемых цифровых и аналого-цифровых интегральных схем, который позволяет конструировать базовые кристаллы интегральных схем с улучшенными техническими характеристиками для решения заданных целевых задач.
- Разработка методов предварительного анализа схем заказчиков с целью создания структуры и архитектуры гетерогенных конфигурируемых интегральных схем с повышенной эффективностью использования трассировочных и иных ресурсов для конкретных классов применений.
- 4) Разработка метода программного прототипирования, позволяющего оценить эффективность реализации схем заказчиков в базовом кристалле до фактического изготовления интегральной схемы за счет оперативной настройки САПР на соответствующие изменения в конструкции, схемотехнике и топологии базового кристалла.
- 5) Исследование и разработка новых обобщенных математических моделей гетерогенных конфигурируемых интегральных схем, позволяющих формализовать решение задач для различных стилей проектирования конфигурируемых интегральных схем.
- 6) Создание методов разработки аналоговых и аналого-цифровых гетерогенных конфигурируемых интегральных схем на основе формализации описания аналоговых блоков.
- Разработка новых схемотехнических решений, повышающих эффективность использования площади кристалла и функциональность базовых матричных кристаллов.
- Создание методов разработки конфигурируемых интегральных схем с повышенной живучестью, радиационной стойкостью и надежностью функционирования в условиях неблагоприятных внешних воздействий.
- 9) Организация и проведение работ по разработке и выпуску готовых изделий, спроектированных на основе предложенных методов.

Методы исследования.

В диссертационной работе для решения поставленных задач используются методы разработки и анализа интегральных схем, методы булевой алгебры, аппарат теории множеств, теория графов.

Научная новизна.

Научная новизна диссертационной работы заключается в теоретическом обобщении, исследовании и создании методов разработки гетерогенных конфигурируемых интегральных схем, ориентированных на целевого потребителя:

1) Предложен комплектования метод решения задачи специализированной аппаратуры электронной компонентной базой проблемно-ориентированными и универсальными микросхемами, которые разрабатываются на основе гетерогенных конфигурируемых интегральных схем В традиционного подхода заказного отличие OT стиля ИХ проектирования.

2) Предложена методика проектирования гетерогенных конфигурируемых интегральных схем, в которой в отличие от известных алгоритмов разработанные в работе методы предварительного анализа схем заказчиков и методы программного прототипирования базовых кристаллов с целью достижения высокой эффективности использования трассировочных и иных ресурсов объединены в единый маршрут, учитывающий различные стили проектирования на одном кристалле (заказной, полузаказной и программируемый).

 Разработана новая теоретико-множественная математическая модель проекта, формализующая различные стили проектирования гетерогенных конфигурируемых интегральных схем (заказной, полузаказной - базовых матричных кристаллов и программируемых схем) и подходы к решению разных задач разработки (от разработки гетерогенного базового кристалла с элементами программирования до создания конкретных зашивок

схем заказчиков). Известный в области разработки микросхем иерархический подход к описанию списка соединений обобщен, расширен и формализован с применением аппарата теории множеств, математической логики и теории графов. На базе разработанной модели осуществлена формализация соответствий между элементами базового кристалла конфигурируемой гетерогенной интегральной схемы и проектируемыми схемами заказчиков, функциональность которых требуется запрограммировать в базовом кристалле.

4) Предложен метод разработки аналоговых и аналого-цифровых гетерогенных конфигурируемых интегральных схем с использованием формализации описания аналоговых блоков. Известные подходы к анализу сложно-функциональных блоков и IP–ядер расширены введением понятий аналоговых ядер и их онтологий, лежащих в основе описаний на уровне макромоделей.

5) Разработана новая конструкция программируемого заказчиком логического элемента, построенного на ячейках базового матричного кристалла, которая в отличие от известного логического элемента типа LUT оптимизирована для выполнения востребованных логических функций.

6) Предложен новый построения метод специализированных конфигурируемых интегральных схем, повышенная живучесть, надежность и радиационная стойкость которых достигается использованием приемлемых и мягких решений, что в отличие от существующих решений позволяет обеспечить функционирование интегральных схем В условиях неблагоприятных внешних воздействий, значения характеристик которых превышают нормы, установленные для штатной работы.

На защиту выносятся следующие положения и научные результаты:

1) Метод решения задачи комплектования специализированной аппаратуры электронной компонентной базой на основе гетерогенных конфигурируемых интегральных схем, которые используются как для создания проблемно-ориентированных микросхем, так и для формирования

унифицированных рядов универсальных микросхем, что позволяет решить задачу создания минимально-необходимых комплектов ЭКБ для ключевых направлений специализированной аппаратуры не менее чем на 30%.

Новый маршрут разработки гетерогенных конфигурируемых 2) интегральных схем, использующих различные стили проектирования на одном кристалле, в рамках которого на основе предложенных методов предварительного анализа схем заказчиков И программного прототипирования базовых кристаллов с использованием стратегии последовательного приближения разрабатываются архитектура И электрическая схема базового кристалла конфигурируемой схемы. Маршрут позволяет провести оценку возможности размещения и трассировки проектов заказчиков и необходимое для этого количество коммутационных и иных ресурсов в базовом кристалле до фактического изготовления.

3) Новая теоретико-множественная математическая модель, формализующая различные стили проектирования гетерогенных конфигурируемых интегральных схем (заказной, полузаказной, программируемый), используемая для исследования и разработки базовых конфигурируемых кристаллов гетерогенных интегральных схем И программирования схем заказчиков на их основе.

4) Новые методы разработки аналоговых и аналого-цифровых гетерогенных конфигурируемых интегральных схем с использованием формализации описания аналоговых блоков на базе введенных в работе понятий аналоговых ядер и их онтологий, лежащих в основе макромоделей, что позволяет сократить время проектирования аналоговых конфигурируемых микросхем в несколько раз.

5) Новая конструкция программируемого заказчиком логического элемента, построенного на ячейках базового матричного кристалла, площадь топологии которого в два раза меньше известного логического элемента.

6) Новый метод повышения живучести, надежности и радиационной стойкости конфигурируемых интегральных схем на основе приемлемых и

мягких решений, позволяющий уменьшить вероятность их неработоспособности в условиях неблагоприятных внешних воздействий.

Реализация и внедрение результатов работы.

Разработанные в диссертационной работе модели, подходы и методы внедрены в АО «НИИМЭ», АО «НПО автоматики», АО «Микрон», АО «ДЦ «Союз», АО «НИИ «Субмикрон» и использованы при выполнении тринадцати НИОКР и одного аванпроекта по разработке и изготовлению программируемых и полузаказных интегральных схем: ОКР «Логика-2», НИР «Логика», ОКР «Логика-И5-РК», ОКР «Алмаз-1», ОКР «Алмаз-13», ОКР «Алмаз-14», ОКР «Преобразователь-3», ОКР «Алмаз-10», ОКР «Логика-И2», ОКР «Преобразователь-17», ОКР «Преобразователь-22», ОКР «Алмаз-14», ОКР «Алмаз-И3», ОКР «Преобразователь-17», ОКР «Преобразователь-22», ОКР «Алмаз-И1-РК», ОКР «Алмаз-И5», аванпроекте «Пемза-А1». Это подтверждено соответствующими документами о внедрении.

Достоверность полученных результатов.

Достоверность результатов диссертационной работы подтверждена многочисленными результатами экспериментальной проверки предложенных методов с использованием программного обеспечения и реализации в кремнии.

Практическая значимость работы заключается в разработке и внедрении оптимальных для своего класса задач микросхем ПЛИС и БМК по таким критериям, как технические характеристики, конструкторские решения, решаемые задачи, технико-экономические показатели. Результаты работы найдут применение при разработке базовых кристаллов конфигурируемых интегральных схем с улучшенными параметрами.

Апробация работы.

Основные результаты работы представлены и обсуждены на следующих конференциях и семинарах:

Всероссийская научно-техническая конференция «Проблемы разработки микро- и нано-электронных систем», МЭС-2018, 2021.

Международная научно-техническая конференция «Электронная компонентная база космических систем», 2012 – 2021 гг.

Российский форум «Микроэлектроника 2021», 2021 г.

Конференция «Актуальные вопросы поставок изделий электронной компонентной базы отечественного производства. Импортозамещение и обеспечение качества», 2018 г.

Научно-технический семинар-совещание «Применение отечественной микроэлектронной высокотехнологичной ЭКБ в РЭА специального назначения – проблемы импортозамещения и пути их решения», 2017 г.

Международный военно-технический форум «Армия-2018».

Научно-практический семинар «Проблемы создания специализированных радиационно-стойких СБИС на основе гетероструктур», 2013 – 2015 гг.

Международный промышленный форум «Радиоэлектроника. Приборостроение. Автоматизация», 2016 г.

Всероссийская научно-техническая конференция по радиационной стойкости электронных систем «Стойкость-2015».

Международная научно-техническая конференция «Проектирование систем на кристалле: тенденции развития и проблемы», 2010 г.

Международный семинар по теории информации «Сверточные коды», 1987 г.

Основное содержание работы.

<u>Во введении</u> обоснована актуальность решения проблем создания специализированных конфигурируемых микросхем. Определена цель диссертационной работы. Сформулирована постановка задачи исследования диссертации. Сформулированы основные положения, выносимые на защиту. Обосновано содержание глав диссертации.

<u>В первой главе</u> проведен анализ текущего состояния и направлений развития конфигурируемых интегральных схем.

Во второй главе приведен предложенный маршрут разработки специализированных гетерогенных конфигурируемых интегральных схем, предварительного описаны методы анализа И программного прототипирования, являющиеся ключевыми В маршруте разработки конфигурируемых схем. В главе рассмотрена предлагаемая архитектура программируемой интегральной схемы.

<u>В третьей главе</u> описаны полученные в ходе выполнения работы математические модели и метрики для разработки специализированных гетерогенных интегральных схем целевого назначения и программирования схем заказчиков на их основе.

<u>Четвертая глава</u> посвящена конструктивным и схемотехническим решениям, разработанным для применения в специализированных конфигурируемых интегральных схемах повышенной живучести и надежности.

<u>Пятая глава</u> посвящена особенностям разработки цифровых базовых матричных кристаллов и аналого-цифровых гетерогенных конфигурируемых интегральных схем.

Список публикаций по работе: список опубликованных по теме диссертации научных трудов содержит 64 работы, в том числе 17 работ без соавторов, 24 статьи в научных журналах, входящих в Перечень периодических изданий, рекомендованных ВАК РФ для публикаций основных результатов диссертаций на соискание степени доктора наук, 5 в изданиях, индексируемых в базе Scopus, одна монография.

Глава 1.Анализ текущего состояния и направлений развития конфигурируемых специализированных интегральных схем

1.1. Обоснование роли гетерогенных конфигурируемых интегральных схем в решении задачи комплектования ЭКБ специализированной аппаратуры

Разработчикам отечественной аппаратуры в современных условиях необходима широкая номенклатура интегральных схем при небольших объемах их потребления [1] - [2]. Задача создания полной номенклатуры PΦ базы электронной компонентной в трудновыполнима из-за ограниченности выделяемых на решение этой задачи средств. Выход из этой заключается в широком применении программируемых ситуации И полузаказных интегральных схем, которые за счет конфигурирования могут заменить большое количество универсальных и заказных микросхем [3]. На это указывают результаты экспертизы правильности применения импортной ЭКБ в проектах ГК «Роскосмос», в рамках которой систематизирована номенклатура потребляемых микросхем [4].

Исследование основных проектов ГК «Роскосмос» за 2016 – 2021 годы в рамках экспертизы правильности применения ЭКБ показало, что доля универсальных ИС, созданных конфигурированием базовых кристаллов может составить не менее 20 % (таблица 1.1). Другая часть - проблемноориентированные ИС реализуются, в основном, на базе конфигурируемых ПЛИС, ПАИС, БМК, АЦ БМК. Суммарная оценка возможного объема гетерогенных конфигурируемых схем в комплектовании специальной аппаратуры интегральными схемами общего назначения составляет не менее 30%.

Таблица 1.1 – Доля микросхем, которые могут быть реализованы на базе гетерогенных конфигурируемых микросхем, включая БМК, АЦ БМК, ПЛИС, ПАИС (экспертная оценка по результатам экспертизы правильности применения ЭКБ в проектах ГК Роскосмос)

Тип ЭКБ	Использование в	Возможна	Использование в
	аппаратуре (%)	реализация на	аппаратуре на основе
		конфигурируемых	конфигурируемых ИС
		ИС (%)	(%)
Память	15	-	-
Аналоговые	10	30	3
СВЧ	10	15	1,5
Питание	10	30	3
Процессоры	10	20	2
Цифровые	5	50	2,5
Контроллеры	15	50	7,5
ПЛИС	15	100	15
Заказные	5	50	2,5
Другие	5	-	-
В сумме	100	-	37

Конфигурирование интегральных схем предполагает наличие встроенных блоков унифицированных коммутационных ресурсов И существенно логических И аналоговых элементов, что снижает эффективность использования площади кристалла. Это приводит к тому, что количество используемых логических элементов в ПЛИС меньше в сотни раз, чем в БМК, и в тысячи раз, чем в заказных схемах. Улучшения эффективности можно достичь за счет гетерогенности, применяя в структуре конфигурируемых схем различные по функциональному назначению сложно-функциональные блоки (СФ блоки) и библиотечные элементы [5]. Архитектура, схемотехника и состав таких схем зависят от класса задач заказчиков, решения которых будет использована для данная конфигурируемая схема. Оптимизация архитектуры И схемотехники конфигурируемых схем, определение состава встроенных СФ блоков являются ключевыми задачами их разработки и зависят также от САПР

конфигурирования, которая является неотъемлемой частью комплекса разработки изделий заказчиков [6].

Хорошо известна мировая тенденция, направленная на массовое применение отлаженных СФ блоков в микросхемах. Использование таких блоков удешевляет и упрощает процесс разработки СБИС. Передовым достижением иностранной микроэлектроники в сфере ПЛИС являются гетерогенные ПЛИС, сочетающие в своем составе логические ресурсы традиционных ПЛИС с набором дополнительных СФ блоков, реализованных на кристалле в заказном виде. Такое решение является логичным воспроизведением широкого использования CΦ блоков. концепции характерной для современного маршрута разработки заказных микросхем. И подобное решение является эффективным, так как позволяет реализовать в программируемой логики наборы микросхемах стандартных блоков (высокоскоростные последовательные интерфейсы, контроллеры внешней динамической памяти и т.п.), характеристики которых не отличаются от их аналогов в составе заказных микросхем.

На рисунке 1.1 приведена программа Европейского Космического Агентства (ESA) по разработке микросхем радиационно-стойких ПЛИС для космического применения.

В работе предложено повышать эффективность использования площади кристалла за счет интеграции на одном кристалле не только заказных и программируемых блоков, но и полузаказных схем, то есть, размещением на кристалле программируемой гетерогенной схемы блоков базовых матричных кристаллов. Симбиоз схем, использующих различные стили проектирования, делает необходимым создание математического аппарата, моделей, алгоритмов и методов исследования и разработки таких схем, а также поиск новых технических решений.



Рисунок 1.1 – Программа по созданию стойких ПЛИС для космических проектов ESA (Европейского Космического Агентства)

Характеристики отечественных конфигурируемых схем, ИХ производительность И функциональность должны находиться на современном уровне, достигнутом крупными зарубежными компаниями. требует максимального Такая постановка задачи учета специфики отечественной аппаратуры, в которой будут применяться конфигурируемые схемы, что неизбежно приводит к их специализации.

Можно выделить несколько основных направлений специализации программируемых схем: по функциональному назначению, по условиям эксплуатации, по конструктивно-технологическому исполнению. B зависимости от функционального назначения различают интерфейсные микросхемы, микросхемы для решения вычислительных задач, задач управления сложными устройствами, задач с максимальным использованием Условия последовательностных схем, регистров, счетчиков, др. эксплуатации, которые определяются воздействующими внешними факторами (ВВФ), зависят от целевого назначения аппаратуры.

Ключевой организационной и научно-технической задачей в области ЭКБ в современных условиях в РФ является задача импортозамещения. Можно выделить несколько этапов импортозамещения, которые отличаются глубиной проработки технических и организационных вопросов (для

аппаратуры ракетно-космической техники представлены на рисунке 1.2). На первом этапе разработчики аппаратуры решают задачу замены микросхем импортного производства на отечественную ЭКБ без существенного архитектуры И электрических схем аппаратуры. Залача изменения экспертизы на этом этапе – аудит оправданности применения импортной ЭКБ. To определение возможности замещения есть импортных Уже комплектующих на отечественные. на ЭТОМ этапе экспертиза показывает, что, несмотря на имеющиеся проблемы, передовую аппаратуру изделий космического и специального назначения можно построить в основном на отечественной электронной компонентной базе.



Рисунок 1.2 - Этапы внедрения ЭКБ ОП в аппаратуру РКТ

На втором этапе предполагается разработка технических решений, позволяющих достигать передовых характеристик изделий на основе унификации аппаратуры. База унифицированных решений может быть сформирована на основе сбора данных в рамках опытно-конструкторских работ по созданию перспективной аппаратуры. Одновременно определяется комплект ЭКБ, минимально-необходимый для построения аппаратуры специального назначения. Использование унифицированных решений необходимо рассматривать, как обязательный элемент на этапе эскизного проектирования аппаратуры. Унификация должна охватывать не только конструкцию и схемотехнику изделий, но и ряды внешних воздействующих факторов, питающих напряжений и токов, интерфейсов, комплектов ИС, СФ блоков и библиотек, корпусов и модулей, САПР.

Второй этап – принципиальный, так как на нем процесс импортозамещения ЭКБ проходит «точку невозврата», после которой разработчикам аппаратуры становится удобнее и выгоднее применять отечественную ЭКБ.

Третий этап предполагает разработку архитектур и алгоритмов обработки сигналов и данных, настроенных на учет особенностей и ограничений отечественной ЭКБ, на способность обходить проблемы отсутствия отдельных типов отечественных микросхем. Архитектура должна оптимизироваться с целью максимального внедрения электронной компонентной базы отечественного производства. Заказные схемы на третьем этапе необходимо реализовывать на основе конфигурируемых гетерогенных интегральных схем.

В работе предложено решать задачу импортозамещения за счет «гибких» решений, то есть массового применения конфигурируемых полузаказных (цифровых и аналого-цифровых БМК) и программируемых (ПЛИС, ПАИС и ПАЦИС) микросхем, на которых возможна реализация как проблемно-ориентированных, так и универсальных интегральных схем [7] -[8].

Основным преимуществом отечественных разработчиков ЭКБ, которое позволяет в полной мере реализовать «гибкие» решения, является возможность совместной работы с разработчиками аппаратуры, обусловленная территориальной близостью и отсутствием языкового барьера.

Другим преимуществом является естественная унификация технологических рядов, типов микросхем, их технических и эксплуатационных характеристик, вытекающая из ограниченного количества российских микроэлектронных предприятий.

Стоимость и сроки разработки «зашивок» полузаказных и программируемых ИС на порядок меньше, чем стоимость и сроки разработки заказных проблемно-ориентированных и универсальных ИС. Особенно это важно при мелкосерийном выпуске и небольшом потреблении микросхем, характерном для аппаратуры специального назначения.

Задачу удовлетворения требований широкого круга разработчиков РЭА предлагается решать путем создания серии проблемно-ориентированных гетерогенных конфигурируемых схем, в составе которых, вместе с традиционными логическими блоками, размещаются встроенные СФ блоки, в том числе, блоки встроенных БМК, на которых можно реализовать различные интерфейсные и иные схемы в виде «зашивок» БМК. Это позволит без расширения номенклатуры, меняя «зашивки» встроенных БМК, выполнять требования многих разработчиков РЭА.

В работе впервые исследованы гетерогенные конфигурируемые интегральные схемы, в составе которых использованы все типы блоков: полузаказные, заказные и программируемые. В этом случае можно достичь оптимальных параметров по быстродействию, потребляемой мощности, Распределение функций между гибкими, площади кристалла. НО потребляющими и медленными программируемыми заказчиком блоками, быстрыми, малопотребляющими, но функционально детерминированными заказными блоками и компромиссными полузаказными блоками дает заказчикам возможность компоновки схем с наилучшими для целевых задач техническими характеристиками.

Традиционным подходом к решению задачи комплектования ЭКБ аппаратуры специального назначения является разработка специализированных и универсальных схем с применением заказного стиля

проектирования. Известно применение в РФ полузаказных ИС для генерации несложных универсальных логических схем, доля которых в аппаратуре незначительна. В данной работе на основе исследования одного из ключевых направлений российского рынка специализированной аппаратуры – рынка аппаратуры для изделий космического назначения, а также анализа технических возможностей современных гетерогенных конфигурируемых схем, сделан вывод, что с их помощью можно создать существенную часть электронной компонентной базы этой аппаратуры, в частности, можно спроектировать проблемно-ориентированные интегральные схемы и ряды минимально-необходимой номенклатуры универсальной ЭКБ.

Предложенный подход к решению проблемы импортозамещения проиллюстрирован на рисунке 1.3, на котором красным контуром обведены проблемно-ориентированные и универсальные интегральные схемы, спроектированные на базе конфигурируемых схем.



Рисунок 1.3 - Конфигурируемые интегральные схемы (обведены красным контуром) – основа комплектования специализированной аппаратуры

Состав и архитектуру конфигурируемой гетерогенной интегральной схемы, использующей все стили проектирования (заказной, программируемый и полузаказной), рассмотрим на примере комплектования аппаратуры, работающей в режиме реального времени.

Этот класс задач решает следующая аппаратура:

- 1. Электроприводы с электронным управлением, в том числе, преобразователи частоты и выпрямители;
- 2. Оптические датчики и изделия на их основе (пример сканирование профиля поверхностей на производстве);
- Устройства коммуникации для радиочастотного обмена информацией (пример – реализация алгоритмов РТК для определения координат комбайна с точностью до 10 см);
- Нейросетевые средства обработки информации (пример камнедетектор);
- Измерительные устройства с предварительной обработкой, в том числе, датчики давления и температуры (пример – датчики для нефтедобывающей промышленности);
- 6. Криптографические системы.

Отличительной особенностью класса задач реального времени является использование большого разнообразия специализированных алгоритмов и блоков, наряду с высокими требованиями к производительности некоторых требований к детерминированности наличием подсистем и времени операций. Наличие составе гетерогенной исполнения многих В конфигурируемой схемы встроенного БМК позволит реализовать на его основе такие высокопроизводительные блоки как ЦПУ уровня приложений или тензорный процессор, которые не входят в набор СФ блоков большинства современных микросхем ПЛИС, а их реализация на поле программируемой логики потребует значительного количества логических ресурсов микросхемы ПЛИС, давая при этом худшие характеристики, чем можно получить при реализации в полузаказном виде.

Исследуемые в работе проблемно-ориентированные гетерогенные конфигурируемые интегральные схемы позволят создать единую аппаратную платформу для разработки аппаратуры, работающей в масштабе реального времени. На рисунке 1.4 приведена структурная схема распределения основных функциональных блоков такой аппаратуры по составным элементам архитектуры гетерогенной конфигурируемо схемы.



Рисунок 1.4 – Элементы гетерогенной конфигурируемой схемы для применения в аппаратуре, работающей в масштабе реального времени

В приведенной на рисунке 1.4 структуре на «зашивках» встроенного БМК реализуются два больших класса задач аппаратуры, работающей в масштабе реального времени – выполнение графического интерфейса пользователя (зашивка 1) и нейросетевые алгоритмы (зашивка 2). Вычислительные средства, закладываемые в указанные зашивки, могут использоваться универсальным образом в аппаратуре, работающей в масштабе реального времени. К данным блокам предъявляются высокие требования по производительности, которые трудно удовлетворить при их реализации на логических элементах ПЛИС. Переменные части алгоритмов реального времени будут реализовываться программируемой частью микросхемы, т.е. на программируемых логических элементах.

1.2. Текущее состояние и направления развития конфигурируемых интегральных схем

1.2.1. Обзор архитектурных решений и методов анализа структуры современных ПЛИС

С момента появления первых ПЛИС оценка их коммутационных ресурсов стала представлять большой интерес для исследователей. Особенно актуальной данная проблема стала в связи с ростом логической емкости ПЛИС. На данный момент существует два основных подхода к оценке архитектуры ПЛИС. В основе первого подхода лежат экспериментальные методы, которые заключаются В прохождении полного маршрута проектирования на схемах из тестового набора [9] – [13]. Для данных методов характерна высокая точность оценки, сопровождаемая большими вычислительными и временными затратами. В последние годы развивается теоретический подход, позволяющий оценить структуру коммутационных ресурсов на ранних этапах, не прибегая к полному прохождению маршрута проектирования [14] – [21]. Применение его позволяет расширить спектр анализируемых архитектур и существенно сократить время разработки новых ПЛИС [22].

В данной работе представлен обзор и анализ современных методов и моделей оценки коммутационных ресурсов ПЛИС. Приводится сравнение методов и моделей, выполняется оценка их эффективности и возможности применения при создания отечественных ПЛИС.

Обзор коммутационных ресурсов ПЛИС. Ключевым элементом программируемых логических интегральных схем является логический блок (ЛБ), который состоит из N базовых логических элементов (БЛЭ), также называемых конфигурируемыми логическими блоками (КЛБ). Классический БЛЭ включает в себя триггер и логическую схему *LUT* (*look-up table*) с *K*-входами, которая может реализовать любую логическую функцию *K*-переменных (рисунок 1.5).



Рисунок 1.5 - Структура классического БЛЭ

На данный момент в зависимости от расположения логических блоков и структуры коммутационных ресурсов можно выделить два основных класса ПЛИС: островные и иерархические.

Структура островной (*island-style*) ПЛИС представлена на рисунке 1.6. В архитектуре конфигурируемые блоки равномерно распределены между узлами коммутации и линиями связи по всей площади кристалла, образуя матрицу. К островным ПЛИС можно отнести ПЛИС серии Xilinx 4000 [23], Altera Stratix II [24].



ЛБ - логический блок,К - блок коммутации,М - блок маршрутизации

Рисунок 1.6 - Структура островной ПЛИС

Важными элементами коммутационной структуры островной ПЛИС являются блоки коммутации (БК, connection box, CB) и блоки маршрутизации (БМ, switch box, SB). Блоки коммутации необходимы для подключения входов и выходов БЛЭ к горизонтальным и вертикальным трассировочным каналам шириной W и характеризуются коэффициентом разветвления F_c , а блоки маршрутизации предназначены для сегментации трассировочных каналов и характеризуются коэффициентом разветвления F_s [25] (рисунок 1.7).



Рисунок 1.7 - Блок маршрутизации $F_s = 6$ (а) и коммутации $F_c = 2(6)$ [14] В иерархических ПЛИС коммутационные ресурсы имеют многоуровневую структуру. Трассировка соединений между соседними БЛЭ не требует большого числа коммутаторов и не выходит за пределы нижнего уровня, в то время как для коммутации удалённых друг от друга на достаточно большое расстояние КЛБ необходим переход с нижних уровней коммутации на верхние и обратно. К иерархическим можно отнести ПЛИС Altera серии APEX [26] (рисунок 1.8), FLEX 10K [27]. Для связи БЛЭ в пределах группы используется локальная шина, а на верхнем уровне иерархии — непрерывные трассы *FastTrack*, связывающие отдаленные части схем.



Рисунок 1.8 - Структура иерархической ПЛИС Altera серии APEX [18]

В ПЛИС соединения для коммутации могут быть реализованы в двух исполнениях: *single-driver* и *multi-driver* (рисунок 1.9) [28]. В случае применения *multi-driver* исполнения используются двунаправленные соединения и ключи (буфера с третьим состоянием), в то время как в *single-driver* — однонаправленные. На данный момент наиболее распространенным типом являются *single-driver* соединения.



Рисунок 1.9 - Multi-driver (а) и single-driver (б) соединения в ПЛИС [28]

Модели и методы анализа коммутационных ресурсов ПЛИС. Разработка коммутационных методов анализа ресурсов началась с ПЛИС. Самыми появлением на первых ранними являются экспериментальные методы, суть которых состоит в прохождении полного маршрута проектирования на схемах из тестового набора [9] – [13] с анализом эффективности коммутационных последующим ресурсов разрабатываемой ПЛИС при помощи точных моделей площади, мощности и задержек [29], [30]. При использовании данных методов выбор оптимальной структуры коммутационных ресурсов зависит от разработчика, решения которого опираются на полученные результаты и опыт разработки.

На рисунке 1.10 представлен один из возможных маршрутов проектирования [10]. В качестве входных данных выступает описание архитектуры ПЛИС и набор тестовых схем (benchmark circuits). На первом этапе выполняется логический синтез (logic synthesis) и технологическое отображение (technology mapping) тестовых схем. На втором этапе выполняется кластеризация, размещение И трассировка полученных оптимизацией задержек. В качестве критериев выбора нетлистов с оптимальной архитектуры выступают площадь и бстродействие тестируемых схем. Аналогичный маршрут был предложен в работе [11]. В [12] fGREP, представлен алгоритм который позволяет вычислить

перегруженность блоков маршрутизации с учетом размещения схем в островных ПЛИС и оценить требуемую ширину канала *W*.



Рисунок 1.10 - Маршрут проектирования в САПР VPR [10]

Существуют экспериментальные методы, в которых не требуется полного прохождения маршрута проектирования. Например, в работе [13] был предложен модифицированный метод Монте-Карло для быстрой оценки коммутационных ресурсов ПЛИС. В данном методе в коммутационном графе ПЛИС случайным образом удаляется К% ребер, после чего для всех возможных наборов пар источник-приемник выполняется трассировка. Повторив операцию множество раз для различных вариантов модифицированного коммутационного графа, можно определить коммутационной ПЛИС. трассируемость анализируемой структуры Недостатком данного алгоритма является отсутствие многотерминальных цепей, которые сильно влияют на оценку архитектуры ПЛИС.

Ключевым преимуществом экспериментального подхода является его высокая точность. Тем не менее экспериментальный подход содержит ряд существенных недостатков, которые ограничивают его применение для анализа коммутационных ресурсов при разработке отечественных ПЛИС. Во-первых, прохождение полного маршрута занимает много времени, а для получения статистически достоверных результатов необходимо большое количество экспериментов на большом комплекте тестовых схем. С учетом того, что для современных ПЛИС, содержащих миллионы логических элементов, полный маршрут может занять несколько дней [31], анализ различных структур коммутационных ресурсов ПЛИС на наборах тестовых схем может занять недели и даже месяцы. Например, для анализа ПЛИС стандартной островной архитектуры на наборе из 20 тестовых схем потребуется не менее 130 дней [32]. Во-вторых, для прохождения полного маршрута требуется наличие полноценного САПР, оптимизированного с учетом каждой архитектуры [29], [33] – [35]. В-третьих, несмотря на точность экспериментальный подход не указывает на истинные причины, по которым одна структура эффективнее другой.

Проблемы экспериментального подхода, с которыми столкнулись разработчики ПЛИС, подтолкнули исследователей к разработке теоретических методов [14] – [21]. В отличие от экспериментальных методов теоретические дают возможность выполнить быстрый анализ и оценку коммутационных ресурсов ПЛИС и тем самым ускорить её разработку.

Работы по теоретическим методам [14], [15] базируются на стохастической модели Эль Гамаля [36] прогнозирования количества требуемых трассировочных ресурсов:

$$W = \lambda R / 2$$

где W — ширина канала, λ — среднее количество контактов ячеек, R — средняя длина цепей.

Браун и др. показали [14], что данная модель достаточно точно позволяет оценить ширину каналов в ПЛИС, и разработали стохастическую

модель, которая позволяет оценить вероятность успешной трассировки для островных ПЛИС с максимальной длиной соединений L = 1. Вероятность успешной трассировки в данной работе вычисляется по формуле:

Routability =
$$\frac{1}{C_T} \sum_{i=1}^{C_T} P(R_{Ci}),$$

где C_T — множество 2-х терминальных соединений, R_{Ci} — статистическое событие (успешная трассировка), $P(R_{Ci})$ — вероятность успешной трассировки C_i .

Работы Эль Гамаля и Брауна получили развитие в [16], где Дас и Вильтон представили новую аналитическую модель, которая позволяет оценить вероятность успешной трассировки в ПЛИС.

В работе [15] для оценки трассируемости схем в ПЛИС применяется модифицированная модель Эль Гамаля. Среднее количество контактов ячеек (λ') рассчитывается с учетом многотерминальных цепей, а для вычисления средней длины цепей применяется модель Файера [37]. В данной модели выделяется средняя длина внутренних (R_i) и внешних цепей (R_e) в области радиусом A, поэтому формула (1) преобразуется для вычисления требуемой ширины канала W_i и W_e :

$$W_i = \frac{\lambda' R_i}{2}, W_e = \frac{\lambda' R_e}{2}.$$

Зная значения данных величин, заданную ширину канала ПЛИС W и количество соединений W_d , связывающих напрямую выход элемента с соседними БЛЭ, авторы выводят условия, при которых схема трассируется в данной ПЛИС:

$$W_e > 0.5 + I_G$$
 или $W_i > I_G + I_D + 0.5$, схема нетрассируема,
 $W_e < I_G - 0.5$ и $W_i < I_G + I_D + 0.5$, схема трассируема.

Стохастические модели не получили большого развития из-за невозможности учета различных схем коммутаций и прямых соединений с различной длиной.
Другие методы оценки коммутационных ресурсов ПЛИС часто основаны на правиле — эмпирической зависимости Рента [38] между средними размерами схем и количеством их контактов:

$$T = kN^p$$

где T — среднее количество контактов схем, k — среднее количество контактов логических элементов в схемах (коэффициент Рента), N — количество логических элементов в схемах, p — экспонента Рента. Так, модель Файера опирается на правило Рента для оценки длины межсоединений.

Используя правило Рента, Дас и Вильтон в [17] представили аналитическую модель для оценки коммутационных ресурсов ПЛИС без опоры на экспериментальные данные. Входные данные модели разбиты на 3 группы:

- архитектурные параметры количество входов в БЛЭ К, количество БЛЭ в логическом блоке N, количество входов в логический блок I,
- схемные параметры параметр Рента схемы *p*, количество элементов в схеме после технологического отображения в библиотеку из 2-х входовых элементов *n*₂ и её глубину *d*₂,
- 3. среднее количество неиспользуемых входов БЛЭ ү.

На основе входных данных, модель позволяет вычислить количество БЛЭ в логическом блоке, необходимых для реализации схемы, среднее количество входов и выходов, используемых в КЛБ, и другие параметры, позволяющие оценить количество требуемых ресурсов ПЛИС.

Смит и др. в работах [18] – [19] представили решение проблемы определения оптимальных размеров транзисторов (*transistor sizing*) и параметров архитектуры ПЛИС (*N*, *K*, *I*, *F_c*) с помощью геометрического программирования.

Метод оценки трассируемости для широкого спектра архитектур без использования тестового набора схем *Wotan* на основе надежности сетей был

представлен в работе [20]. В основе данного метода лежат алгоритм на основе подсчета путей в коммутационном графе, методы из комбинаторики и теории вероятности, которые позволяют оценить вероятность перегруженности вершин и успешной трассировки источников сигналов до ближайших приёмников.

В качестве исходных данных *Wotan* использует коммутационный граф и распределение вероятности длин цепей P(l). Результатом является оценка трассируемости архитектуры ПЛИС, представленной в виде графа коммутаций. Маршрут *Wotan* состоит из 3 этапов:

1. вычисление перегруженности вершин коммутационного графа,

- 2. анализ трассируемости пар источник-приемник,
- 3. вычисление метрики трассируемости.

Ключевым преимуществом метода является данного его универсальность и независимость от структуры коммутации ПЛИС и необходимости отсутствие В прохождении полного маршрута проектирования. Но, как и в работе [14], *Wotan* при анализе трассируемости не учитывает многотерминальные цепи, что приводит к неточной оценке некоторых архитектур.

Несмотря на то, что теоретические модели позволяют обнаружить зависимости между параметрами архитектуры ПЛИС, они менее точны, чем экспериментальные, т.к. не учитывают все факторы, влияющие на точность модели. Поэтому помимо теоретического и экспериментального подхода получил развитие и смешанный подход к оценке коммутационных ресурсов ПЛИС [32]. Данный подход с одной стороны может быть точнее, чем теоретический, за счет построения модели на основе точных данных. С другой стороны, сильное влияние на качество моделей может оказывает набор тестовых схем и качество применяемого САПР.

Смешанный подход, примененный в работе [32], использует аналитическую модель анализа ПЛИС с островной архитектурой, которая позволяет оценить требуемую ширину канала *W*_{need}. Для получения

зависимостей использовался набор тестовых схем, которые были равномерно разделены на 2 группы:

- обучающую, по которой строится модель,
- контрольную, по которой оценивается точность модели.

Полученная зависимость требуемой ширины канала *W_{need}* имеет следующий вид:

$$W_{need} = W_{abs_min} + \frac{1}{\beta} \left(\frac{W_{abs_min}}{Fs} \right) \left(\frac{W_{abs_min}}{Fc_{in}} \right)^{\alpha in} \left(\frac{W_{abs_min}}{Fc_{out}} \right)^{\alpha out} + \frac{\lambda (L-1)}{4} \left(1 + \frac{1}{Fc_{in}^{\alpha in}} \right)^{\alpha out}$$

где W_{abs_min} — минимальная ширина канала, необходимая для успешной трассировки схемы заказчика в ПЛИС, вычисляется по модифицированной формуле (1), где R = 4.43, $\lambda = 0.88N + 3.2 = [I = 2N + 2] = 0.44I + 2.3$; второе слагаемое учитывает влияние коэффициента разветвления БК (F_{cin} , F_{cout} — коэффициенты для входов и выходов БЛЭ) и БМ, $\beta=3$, $\alpha_{in}=0.5$, $\alpha_{out}=0.25$ — параметры для подстройки модели; третье слагаемое учитывает влияние соединений L > 1. Если в ПЛИС контакты БЛЭ не эквивалентны (Eqv = 0), то средняя длина R в (1) заменяется на σR , $\sigma = 1.166$, а F_{cin} на F_{cin}/μ , где $\mu=0.33$.

В таблице 1.2 приведено сравнение существующих методов оценки коммутационных ресурсов ПЛИС. Как видно из таблицы большинство методов и моделей имеют похожий набор входных данных, используя который они дают оценку различным параметрам архитектуры ПЛИС.

Таблица 1.2 - Сравнение методов и моделей оценки коммутационных

ресурсов

	Входные данные	Оценка параметров ПЛИС
[21]	 1.Параметры логики ПЛИС (N, K, I), 2.Параметр Рента <i>р</i> схемы, 3.Количество 2-х входовых <i>LUT</i> ячеек в схеме. 	1.Оценка логических ресурсов ПЛИС (кол-во используемых входов <i>i</i> и выходов <i>o</i> , средний размер ЛБ <i>n_k</i> и др.)
[18]	1.Параметры логики ПЛИС, 2.Параметр Рента <i>р</i> схемы,	1.Оценка площади и задержек,2.Размеры транзисторов в

	3.Количество 2-х входовых <i>LUT</i> ячеек в схеме и её глубина.	коммутационных ресурсах		
[14]	 1.Параметры логики ПЛИС, 2.Параметры коммутационных ресурсов ПЛИС (<i>F</i>_{cin}, <i>F</i>_{cout}, <i>F</i>_s), 3.Средняя длина цепей <i>R</i>, 4.Ширина канала <i>W</i>. 	1.Оценка трассируемости схем в ПЛИС		
[15]	1.Параметры логики ПЛИС, 2.Средняя длина цепей <i>R</i> .			
[16]	 1.Параметры логики ПЛИС, 2.Параметры коммутационных ресурсов ПЛИС, 3.Кол-во цепей в схеме и их средняя/максимальная длина после размещения. 			
[20]	1.Коммутационный граф ПЛИС, 2.Распределение длин цепей <i>P</i> (<i>l</i>).			
[17]	 Параметры логики ПЛИС, Параметр Рента <i>р</i> схемы, Количество 2-х входовых <i>LUT</i> ячеек в схеме и её глубина. 	1.Оценка логических ресурсов ПЛИС		
[37]	1.Параметры коммутационных ресурсов ПЛИС ($F_{cin}, F_{cout}, F_s, L, Eqv$), 2.Средняя длина цепей R .	1.Требуемая ширина канала <i>W_{need}</i>		

Развитие направления программируемых интегральных схем привело к появлению большого числа методов и моделей для решения задачи анализа и оценки структуры трассировочных ресурсов. Существующие теоретические методы не обладают достаточной точностью и могут применяться лишь на ранних этапах разработки ПЛИС. Экспериментальные методы требуют для исполнения значительных вычислительных ресурсов и могут применяться лишь для финальной проверки архитектуры ПЛИС. Оптимальным подходом для анализа проектируемых конфигурируемых интегральных схем является разработка и развитие смешанных методов и алгоритмов оценки архитектуры схем, которые на основе правильной стратегии и специально созданного математического аппарата позволят получить лучшие характеристики и значительно сократить время разработки.

1.2.2. Направления развития цифровых базовых матричных кристаллов.

В работе [39] представлен БМК на 160 000 вентилей, построенный по 1 мкм топологическим нормам с применением трехслойной металлизации со встроенными схемами памяти. БМК выполнен по принципу «море вентилей» и содержит 1,3 млн. транзисторов. Как показано на рисунке 1.11, базовая ячейка содержит два типа элементов: тип А – имеет два р – МОП канальных и два п – МОП канальных транзистора, тип Б – имеет четыре п – МОП канальных транзистора с малыми размерами.



Рисунок 1.11 - Топология и эквивалентные схемы базовых ячеек типа A – (два р – канальных и два n – канальных МОП транзистора) и типа Б – (четыре n – канальных МОП транзистора с малыми размерами)

Большинство логических схем имеют элементы типа A, пространство между соседними элементами типа Б используется для трассировки межсоединений.

В работе [40] представлен БМК на КМОП вентилях, содержащий 4268 вентилей и 2304 - битную память, реализованные по 1,5 мкм топологическим нормам с двумя слоями металлизации. С транзисторами малых размеров среднее время задержки распространения сигнала составило 20 нс на вентиль. Представлена гибкая конфигурация встроенной памяти. Шаг первого слоя металла на основе Al составляет 6 мкм, второго – 8 мкм. Размер кристалла 9,68 мм на 9,69 мм.

На рисунке 1.12 представлены топологии базовых ячеек БМК. Базовая ячейка с двумя парами транзисторов имеет размер 27 мкм на 83 мкм.



Рисунок 1.12 – Топологии базовых ячеек БМК: топология базовой ячейки на 2-х транзисторах, топологии 2 – х и 4 – х входового вентиля NAND

Работа [41] представляет архитектуру БМК на основе «моря вентилей» со встроенной памятью. БМК рассчитан на работу при напряжении питания 3,0 – 3,6 В. Эффективная длина каналов МОП транзисторов составляет 0,65 мкм. Средняя задержка распространения сигнала на вентиль составляет 184 пс при двух стандартных нагрузках. Базовая ячейка включает два больших и два малых р- и п- МОП транзистора (рисунок 1.13).



Рисунок 1.13 - Базовая ячейка: а - топология базовой ячейки; б электрическая схема

В работе [42] рассмотрен БМК на основе КМОП транзисторов, организованных в матрицу по типу «море вентилей». КМОП микросхема изготавливалась по 0,8 мкм проектным нормам с первым уровнем металлической трассировки перпендикулярной к расположению блоков транзисторных структур.

На рисунках 1.14 и 1.15 изображены эскизы топологии областей БМК с обычным способом изоляции затворов.



Рисунок 1.14 – Эскиз топологии области БМК при использовании архитектуры RMC



Рисунок 1.15 – Эскиз топологии области БМК при использовании архитектуры СМС

Архитектура реализации макроячеек может быть колоннообразного типа (column macrocell - CMC), или архитектурой стандартного типа (row macrocel - RMC).

На рисунке 1.16 дана топология ячейки БМК при использовании СМС.





На рисунке 1.17 приведены примеры топологии макроячеек (3-х входового вентиля NAND), спроектированного по RMC и CMC.



Рисунок 1.17 - Топологии макроячеек: спроектированных по RMC и по CMC архитектурам

На рисунке 1.18 приведена конструкция БМК в целом.



Рисунок 1.18 - Конструкция БМК

В работе [43] приведена архитектура построения цифровых сигнальных процессоров на основе КМОП матрицы типа «море вентилей», созданной по 1-мкм технологии. На рисунке 1.19 приведены топологии трех типов ячеек на основе КМОП транзисторов: - тип А и тип В содержат две пары р - и n – канальных МОП транзисторов. Тип С содержит транзисторы без изоляции между соседними ячейками.



Рисунок 1.19 - Топологии трех типов ячеек на основе КМОП транзисторов

На рисунке 1.20 приведены топологии ячеек типа А и типа В.



Рисунок 1.20 - Топологии ячеек типа А и типа В

На рисунке 1.21 приведены две топологии макроячеек цифрового умножителя.



Рисунок 1.21 – Две топологии макроячеек умножителя без буферных инверторов

(a)

(Б)

Обычно БМК реализуются сериями, которые, имея одинаковую структуру матрицы, различаются логической емкостью, типом корпусов, количеством выводов. Так в работе [44] описана серия КМОП БМК с различным числом вентилей со встроенными СФ блоками, выполненная по 0,5 мкм проектным нормам с тремя слоями металлизации.

В работах [45] - [46] описаны БМК с различными особенностями архитектуры и схемотехники.

На рисунке 1.22 приведена топология базовой ячейки БМК и ее размер.



Рисунок 1.22 - Топология базовой ячейки БМК

На рисунке 1.23 приведена конструкция БМК с расположением основных блоков.



Рисунок 1.23 - Топология размещения основных блоков БМК

В работах приведены БМК с использованием различных питающих напряжений и схем сдвига уровней сигналов. На рисунках 1.24 и 1.25 приведены топологические решения для n-МОП и p-МОП транзисторов и электрическая схема, обеспечивающая сдвиг уровня сигналов.



Рисунок 1.24 - Топология n-МОП и p-МОП транзисторных структур



Рисунок 1.25 - Электрическая схема, обеспечивающая сдвиг уровней сигналов.

На рисунке 1.26 приведена топология типовой базовой ячейки БМК и ее размеры при 2-х мкм топологических нормах.



Рисунок 1.26 - Топология базовой ячейки и ее размеры при 2-х мкм топологических нормах

На рисунке 1.27 представлены примеры конфигураций расположения функциональных блоков.



Рисунок 1.27 - Примеры расположения функциональных блоков: а – четыре блока с одинаковыми размерами; б – четыре функциональных блока, реализованные на матрице логических элементов типа «моря вентилей»

Отечественными микроэлектронными компаниями освоены серии БМК информационной емкостью до 10 млн. базовых ячеек с рабочей частотой функционирования более 200 МГц. Основные параметры типовых представителей серий БМК приведены в таблице 1.3.

Наименование БМК	Производитель	Число ячеек, тыс.	Рабочая частота, МГц	Технология	Напряжение питания, В	Корпус
1592XM1T АЕЯР.431260.2 63 ТУ	АО «Ангстрем»	100	70	КМОП	5 ±10%	4236.208-2
5516БЦ1Т АЕЯР.431260.3 85 ТУ	АО «Ангстрем»	110	60	КМОП/КНС	5 ±10%	4229.132-3
5522БЦ4ВТ1 АЕЯР.431260.7 51 ТУ	АО «Ангстрем»	123	80	КМОП	5-3.3 ±10%	4236.208-2
5529TH034 AEHB.431260.0 13 TY	ФГУ НПК «Технологичес кий центр «МИЭТ»	800	100	КМОП КНИ	$3.3\pm10\%$	4239.68-1
5529TH114 AEHB.431260 111 TY	ФГУ НПК «Технологичес кий центр «МИЭТ»	2 300	100	КМОП КНИ	2,7-3.63	4245.256- 4; 4251.304-2
5510БЦЗТ АЕЯР.431260.9 01 ТУ	АО «НИИМЭ» АО «Микрон»»	1 000	100	КМОП КНИ, 250 нм	2,7-3.63	4245.240- 6.01
5540TH014 AEHB.431260.3 47 TY	АО «НИИМЭ» АО «Микрон»»	3 500	100	КМОП КНИ, 180 нм	1.8, 3.3 ±10%	4245.256- 4;
5540TH028 AEHB.431260.4 35TY	АО «НИИМЭ» АО «Микрон»»	10 000	200	КМОП КНИ, 90 нм	1.2, 3.3 ±10%	MK 8304.624- 1;

Таблица 1.3 – Параметры микросхем БМК отечественного производства

Другим подходом к разработке конфигурируемых интегральных схем является проектирование полузаказных схем типа eASIC компании Intel, в которых базовая ячейка аналогична логическому элементу ПЛИС, а конфигурирование (зашивка) производится одним слоем. eASIC расширяет функциональные возможности ПЛИС за счёт дополнительных решений, интегрированных в её структуру, и позволяет сгладить основные недостатки ПЛИС (высокую стоимость, высокое энергопотребление и др.). С этой целью исходная ПЛИС и её полузаказной eASIC аналог изготавливаются по одним проектным нормам, имеют идентичную структуру логических и трассировочных ресурсов и отличаются только способом конфигурирования.

Анализ направления развития цифровых БМК показывает, ЧТО логическая емкость БМК непрерывно растет, в БМК встраиваются заказные блоки, блоки Базовая ячейка БМК В первую очередь, памяти. совершенствуется для обеспечения возможности конфигурирования на ней простых аналоговых схем. В микросхемах БМК типа eASIC логические ячейки идентичны логическим элементам ПЛИС, а конфигурирование проводится одним слоем коммутации. Такой подход позволяет быстро и с гарантированной надежностью переносить проекты с ПЛИС на БМК, но сохраняет большинство недостатков ПЛИС.

Однако, существующие микросхемы БМК не обладают достаточной гибкостью применения, так как не имеют возможности оперативного программирования логических элементов заказчиком. В известной литературе не описаны способы построения программируемых заказчиком логических элементов, реализованных на базовых ячейках БМК, не исследованы также зависимости требуемых коммутационных ресурсов для реализации схем заказчиков различного объема на БМК, построенных по существующим и перспективным отечественным технологиям.

1.2.3. Существующие изделия, области применения и основные характеристики аналоговых блоков АЦ БМК и ПАИС

Известные отечественные базовые матричные кристаллы обладают ограниченными функциональными возможностями и не предназначены для реализации различных рядов универсальных аналоговых схем. Зарубежные конфигурируемые аналого-цифровые схемы строятся либо с использованием техники переключаемых конденсаторов, либо на основе ПЛИС со встроенными аналого-цифровыми преобразователями и решают небольшую часть задач аналоговой обработки сигналов (Таблица 1.4).

Параметр	AN221E04 (Anadigm, CIIIA)	A2F060 (Microsemi, CIIIA)
Встроенный АЦП	+	+
Разрядность АЦП, бит	8	10
Встроенный ЦАП	+	+
Разрядность ЦАП, бит	10	10
Количество ОУ, шт.	12	Х
Количество компараторов, шт.	12	Х
Встроенная энергонезависимая память	Х	Х
Встроенная программируемая цифровая часть (ПЛИС)	Х	+
Объем ПЛИС, вентилей	Х	100 000
Радиационная стойкость	Х	+
Дополнительные блоки: - Аналоговые фильтры - ФАПЧ - LDO	Х	Х

Таблица 1.4 – Характеристики зарубежных аналого-цифровых конфигурируемых схем

Анализ разработанных в рамках работы СФ блоков и возможности их реализации в конфигурируемых аналоговых интегральных схемах (АЦ БМК и ПАИС), выполненных по КМОП КНИ технологии уровня 180 нм, показывает возможность реализации универсальных микросхем с характеристиками, указанными в таблице 1.5.

Таблица 1.5 - Характеристики аналоговых и аналого-цифровых универсальных интегральных схем на основе «зашивок» АЦ БМК и ПАИС

Тип аналоговой схемы		Основные характеристики	
	Последовательного приближения	Количество разрядов: 8 16 Частота выборок: до 5 МВыб/с Количество каналов: до 16	
Аналого- цифровые преобразователи	Конвейерные	Количество разрядов: 10 16 Частота выборок: до 25 МГц Количество каналов: до 16	
	Дельта-сигма	Количество разрядов: до 24 Частота выборок: до 2 кГц Количество каналов: до 8	

	На резистивной	Количество разрядов: 10 16		
II	или емкостной	Время установления: до 20 нс		
цифро-	матрице	Количество каналов: до 8		
аналоговые		Количество разрядов: до 24		
преобразователи	Дельта-сигма	Время установления: до 100 мкс		
		Количество каналов: до 8		
		Коэффициент усиления: до 140 дБ		
	enguidaullu le	Частота единичного усиления: до 10 МГц		
малошумящие оп	срационные	Напряжение смещения: не более 50 мкВ		
усилители		Спектральная плотность шума: не более		
		10нВ/√Гц		
		Коэффициент усиления: до 100 дБ		
Быстродействуюш	цие операционные	Частота единичного усиления: до 100 МГц		
усилители		Напряжение смещения: не более 100 мкВ		
		Скорость нарастания: не менее 100 В/мкс		
Компараторы		Напряжение смещения: не более 100 мкВ		
		Время срабатывания: не более 5 нс		
		Выходное напряжение: 1,0 5,0В		
Источники опорно	ого напряжения	Температурный коэффициент: не более 50		
		ppm/ ⁰ C		
Супервизоры питания, схемы Reset		Контролируемое напряжение: 0,9 15В		
		Количество каналов: до 8		
ШИМ-контроллеры		Рабочая частота: 50 2000 кГц		
		Выходной ток: не менее 2А		
Линейные регуляторы напряжения		Падение напряжения: не более 0,2В		
		Входное напряжение: 4,5 40В		
типульсные DC-I	лепреобразователи	Выходной ток: не менее ЗА		
		Количество каналов: не менее 32		
мультиплексоры		Коммутируемое напряжение: до 60В		

Дальнейшим комбинация развитием ЭТОГО подхода является нескольких универсальных схем в одно «гибкое» решение на основе полузаказных И программируемых схем для конкретных областей применения (таблица 1.6). Особая роль отводится формированию зашивок АЦ БМК под определенные классы задач: акселерометры, схемы управления питанием, схемы обработки видеоизображения, усиления и оцифровки большого количества различных датчиков, схемы управления двигателями, многоканальные системы аналого-цифровой и цифро-аналоговой обработки и другие.

Таблица 1.6 - Примеры специализированных полузаказных схем на основе «зашивок» АЦБМК

Типы микросхем	Области применения
Микросхемы обработки датчиков физических величин (температуры, давления, ускорения, влажности и др.) Микросхемы обработки мостовых датчиков (магниторезистивных и др.) Микросхемы обработки шунтовых датчиков тока	Телеметрия, системы управления, измерительные приборы
Микросхемы обработки видеосигнала	Фотоприемные устройства
Преобразователи емкость-код, напряжение-частота и частота-напряжение	Акселерометры, измерительные приборы
Датчики вращения, микросхемы управления приводами и электродвигателями	Электродвигатели и приводы
ШИМ-регуляторы, драйверы силовых ключей, трансляторы уровней	Преобразователи напряжения, силовая электроника
Кварцевые генераторы, буферы тактовой частоты, преобразователи уровней, приемо-передатчики	Системы управления, измерительные приборы

Для разработки набора конфигурируемых аналоговых интегральных схем, на котором можно реализовать минимально-необходимые ряды универсальных аналоговых микросхем и проблемно-ориентированные микросхемы, необходимо развивать программное обеспечение, средства отладки и разработки таких схем. В результате предложенного в работе подхода помимо существенной экономии времени и средств на разработку будут получены новые свойства финальных изделий. Наглядным примером, представляющим преимущества описанного способа обработки аналоговых сигналов на АЦ БМК и ПАИС, является достижение минимальных массогабаритных характеристик. Практика показывает, что реализация типовых схем на АЦ БМК и ПАИС в аппаратуре позволяет более чем на порядок уменьшить размеры аналоговых модулей аппаратуры при улучшении их характеристик.

1.3. Выводы

Проведен анализ текущего состояния, направлений развития и роли гетерогенных конфигурируемых интегральных схем в комплектовании ЭКБ специализированной аппаратуры. Показано, что в основе решения задачи импортозамещения ЭКБ и ускоренного развития аппаратостроения в РФ может лежать широкое применение гетерогенных конфигурируемых интегральных схем.

Разработан метод комплектования специализированной аппаратуры проблемно-ориентированными и универсальными интегральными схемами, построенными на основе гетерогенных конфигурируемых схем.

Глава 2. Маршрут разработки, методы предварительного анализа и программного прототипирования специализированных гетерогенных конфигурируемых интегральных схем. Оптимизация архитектуры ПЛИС

2.1 Маршрут разработки специализированных гетерогенных конфигурируемых интегральных схем. Особенности выбора их функциональных блоков

Маршрут разработки конфигурируемых интегральных схем

Известные алгоритмы разработки ПЛИС состоят из нескольких основных этапов оптимизации программируемой части интегральной схемы: этапа выбора архитектуры ПЛИС, логического синтеза, кластеризации, размещения и трассировки схемы заказчика, временного анализа и оценки занимаемой площади.

В отличие от известных алгоритмов предлагаемый маршрут разработки базового кристалла гетерогенной конфигурируемой схемы объединяет на одном кристалле в единое целое разработку различных по стилю проектирования блоков (заказных, полузаказных и программируемых). Существующие теоретические методы анализа дополнены новыми методами и метриками, выделенными в отдельный этап маршрута – предварительный анализ, с целью повышения точности предварительной оценки требуемых трассировочных и иных ресурсов для реализации схем заказчиков заданных классов. Существующие экспериментальные алгоритмы, основанные на переборе оптимизируемых параметров, в данной работе развиты в метод программного прототипирования, использующий для достижения высокой эффективности трассировочных и иных ресурсов предложенную в работе стратегию последовательного приближения. Предлагаемый маршрут разработки позволяет провести оценку возможности размещения проектов заказчиков различных классов в базовых кристаллах конфигурируемых схем

и необходимые для этого параметры коммутационных и логических блоков до фактического их изготовления.

Базовая часть блок-схемы маршрута разработки специализированной гетерогенной конфигурируемой схемы для решения определенного класса задач приведена на рисунке 2.1. Алгоритм можно условно разбить на пять этапов. На первом этапе производится выбор и анализ типовых схем заказчиков заданного класса. На втором этапе проводится предварительный анализ (ПА) схем заказчиков с целью определения коммутационных и иных ресурсов базовых кристаллов конфигурируемых схем, необходимых для размещения и трассировки выбранных схем. В рамках предварительного анализа также производится оценка требуемого объема конфигурационной памяти, исполнения, состава и окружения СФ блоков. Предварительный значения анализ определяет граничные трассировочных ресурсов, оптимальный размер и состав логических и СФ – блоков. При проведении предварительного анализа используются разработанные в работе методики и метрики.

На третьем этапе на основе полученных ограничений определяется архитектура и разрабатывается эскизная схема базового кристалла конфигурируемой схемы. Архитектура может быть как уникальной, так и выбранной из множества существующих решений.

На четвертом этапе проводится программное прототипирование (ПП) – апробация базового кристалла конфигурируемой схемы до этапа физического изготовления на основе предложенной в работе стратегии последовательного приближения. На этапе программного прототипирования определяется достаточность функциональных, трассировочных и иных ресурсов ИС для размещения и трассировки конкретных схем заказчиков в базисе разрабатываемого базового кристалла. Программное прототипирование позволяет проверить возможность конфигурирования типовых для данного класса задач схем заказчиков в разрабатываемом базовом кристалле

конфигурируемой схемы с помощью САПР до этапа их физического изготовления.



Рисунок 2.1 – Базовая часть блок-схемы маршрута разработки специализированных гетерогенных конфигурируемых интегральных схем

На пятом этапе разрабатывается электрическая схема и топология конфигурируемой интегральной схемы. На рисунках 2.2 и 2.3 приведены отдельные части маршрута разработки.



Рисунок 2.2 – Определение основных параметров базового кристалла. Часть маршрута разработки



Рисунок 2.3 – Предварительный анализ. Часть маршрута разработки

Особенности выбора функциональных блоков специализированных конфигурируемых интегральных схем

Одной из важнейших частей маршрута разработки конфигурируемых выбора И интегральных схем является этап реализации сложнофункциональных блоков. Помимо собственно определения состава СФ блоков необходимо выбрать их исполнение (заказное, полузаказное, программируемое), сформировать окружение (схемное и структурное), задать интерфейсы связи и управляющие сигналы, а также провести надлежащее размещение на доступные позиции базового кристалла. Во многих случаях необходимо сформировать связки СФ блоков и шин коммутации. Например, блоки DSP требуют наличия в окружении блоков памяти определенной емкости, СФ блоки сериалайзеров-десериалайзеров

должны иметь возможность соединения с высокоскоростными шинами коммутации.

Специализация конфигурируемых интегральных схем определяется функциональным назначением, условиями эксплуатации, требованиями к конструкции, техническими характеристикам и иными параметрами. Состав и требования к функциональным блокам зависят, в первую очередь, от функционального назначения схемы. В настоящей работе рассмотрено несколько областей применения конфигурируемых схем и выбор необходимого для них состава встроенных блоков.

Характерными особенностями конфигурируемых микросхем, предназначенных для использования в современных системах связи типа «цифрового радио», являются:

- Широкое применение СФ блоков при реализации специализированных алгоритмов обработки данных в стандартных интерфейсах и протоколах системного уровня;
- Применение алгоритмов цифровой обработки сигналов (ЦОС);
- Применение ЦПУ уровня приложений или процессоров цифровой обработки сигналов (ПЦОС);
- Высокие требования к быстродействию и энергопотреблению;
- Использование высокоскоростных интерфейсов.

Приведенные особенности требуют от конфигурируемых микросхем:

- Высокую логическую емкость программируемой части;
- Наличие в составе аппаратных СФ блоков (DSP, блочного ОЗУ, высокоскоростных приемопередатчиков (SerDes);
- Наличие в составе аппаратного СФ блока ЦПУ уровня приложений;

К основным ресурсам конфигурируемых микросхем современных систем связи предъявляются следующие требования:

- Количество программируемых логических элементов: от 100 000 до 300 000;
- 2. Количество блоков DSP: от 400 до 1 500;
 - 60

- 3. Объем блочного ОЗУ: от 8Мбит до 27Мбит;
- 4. Количество блоков ФАПЧ: не менее 8.

Дополнительным условием является наличие набора программных СФ блоков, реализующих крупные интерфейсы и модули:

- ЦПУ уровня приложения (с рабочей частотой от 500МГц до 1ГГц);
- Процессора ЦОС для алгоритмов обработки данных;
- Стандартных цифровых интерфейсов: SPI, I²C, UART;
- Инфраструктуры шин ARM AMBA AXI/AXI-Stream;
- Приемников и передатчиков стандарта LVDS;
- Контроллера DMA;
- Средств отладки: ILA (Integrated Logic Analyzer) и т.п.;
- Контроллеров статической (SDRAM) или динамической памяти (DDR, DDR3);
- Контроллеров внешнего ПЗУ типа eMMC (NAND, SDHC (Secure Digital High Capacity));
- Контроллеров Ethernet MAC с интерфейсами RMII и SGMII для подключения к внешней физической части (Ethernet PHY);
- Интерфейса JESD204B.

Для реализации алгоритмов ЦОС также необходима реализация набора базовых алгоритмических и структурных модулей в виде программных СФ блоков:

- БПФ/ОБПФ (FFT/IFFT);
- Алгоритмов синтеза сигналов (Cordic, DDS (Direct Digital Synthesizer));
- Алгоритмов DUC (Digital Up Converter)/DDC (Digital Down Converter);
- Математических операций: умножения, суммирования, умножения с накоплением, деления, квадратного корня;
- Цифровых фильтров: КИХ (FIR), КИГ (CIC);

– Блоков памяти: FIFO, ПЗУ, ОЗУ (однопортовых, двухпортовых).

Для аппаратуры ЦАР/АФАР предъявляются аналогичные требования по логическим ресурсам ПЛИС, составу базовых алгоритмических и структурных СФ блоков, отличие состоит лишь в меньших требованиях к составу крупных СФ блоков, из которого исключаются:

- ЦПУ уровня приложений;
- Контроллеры внешней памяти (SDRAM, DDR, DDR3);
- Контроллеры внешнего ПЗУ типа еММС.

Исходя из потребностей целевых областей применения набор аппаратных СФ блоков гетерогенных конфигурируемых схем для систем связи и аппаратуры АФАР должен быть следующим:

- 1. ЦПУ уровня приложений RV64GC;
- 2. ЦПУ реального времени RV32IMAFC;
- 3. Блоки DSP;
- 4. Блоки ОЗУ;
- 5. Блоки высокоскоростных приемопередатчиков.
- 6. Блоки цифрового БМК, в которых реализуются в виде «зашивок» следующие схемы:
 - процессор RISC-V уровня приложений;
 - тензорный процессор;
 - блок статического ОЗУ (емкостью 1-1,5 Мбит);
 - последовательные интерфейсы Ethernet MAC и RapidIO;
 - интерфейс SpaceWire с аппаратной реализацией протокола высокого уровня (например, RMAP);
 - модули расширений для ЦПУ RISC-V реального времени (дополнительные расширения ISA RISC-V, специализированные вычислители, сопроцессоры и т.п.);
 - контроллер внешнего динамического ОЗУ (DDR);
 - набор криптографических алгоритмов (AES, 3DES, ГОСТ 28147 и т.п.);

– набор интерфейсов (CAN, UART, I²C, SPI, и т.п.).

Реализация СФ блоков осуществляется в заказном и полузаказном виде для достижения их лучших характеристик (в плане быстродействия и энергопотребления) при минимизации затрат площади кристалла. Характеристики основных функциональных блоков для технологии уровня 28 нм приведены ниже:

1. ЦПУ уровня приложений RV64GC

ЦПУ уровня приложений реализуется на базе открытой архитектуры ISA RISC-V, сопоставимой по производительности с лицензируемой архитектурой процессорных ядер компании ARM Holdings последних версий. Одним из важнейших компонентов аппаратуры систем связи 6-го поколения является ЦПУ уровня приложений типа Cortex-A5 и Cortex-A7. Ввиду сложности лицензирования процессорных ядер компании ARM Holdings и влияния на данный процесс санкционных ограничений открытая архитектура RISC-V выглядит идеальной заменой для аппаратного решения такого уровня. Для встроенных приложений решения на базе архитектуры RISC-V сопоставимы решениями ARM Holdings с компании ПО энергопотреблению быстродействию И BO всем диапазоне целевых применений от микроконтроллеров до ЦПУ уровня приложений среднего уровня, имея меньшую площадь кристалла.

Предварительные характеристики ЦПУ уровня приложений RV64GC следующие:

- Разрядность: 64-бит;
- Количество ядер: до 4;
- Набор инструкций: GC (IMAFDC);
- Рабочая частота: 800МГц;
- Кэш-память 1-го уровня: 32Кбайт для инструкций и данных;
- Кэш-память 2-го уровня: от 128 до 2048 Кбайт;
- Аппаратные MMU и TLB;
- Аппаратный FPU одинарной и двойной точности;

– Поддержка полноценных ОС типа Linux.

ЦПУ уровня приложений с такими характеристиками закрывает потребности в ЦПУ уровня Cortex-A5 и Cortex-A7 для аппаратуры систем связи нового поколения, он также подходит для использования в других областях применения в качестве основного процессора, работающего под управлением полноценной ОС.

2.ЦПУ реального времени RV32IMAFC

ЦПУ реального времени также реализуется на базе открытой архитектуры ISA RISC-V, сопоставимой производительности ПО С лицензируемой архитектурой процессорных ядер компании ARM Holdings последних версий. Данный ЦПУ соответствует по характеристикам ЦПУ Cortex-R5 компании ARM Holdings уровня И предназначен ДЛЯ вычислительных и управляющих задач в масштабе реального времени, работая под управлением ОС РВ (операционная система реального времени). На основе данного ЦПУ в гетерогенной конфигурируемой схеме может быть обработки эффективная процессорам цифровой организована замена сигналов (ЦОС), используемым в составе РЭА в виде дискретных компонентов (отдельных микросхем).

Предлагаемые для решения целевых задач характеристики ЦПУ реального времени RV32IMAFC следующие:

- Разрядность: 32-бит;
- Количество ядер: 1;
- Набор инструкций: IMAFC;
- Рабочая частота: 300-500МГц;
- Кэш-память 1-го уровня: 16Кбайт для инструкций и данных;
- Кэш-память 2-го уровня: отсутствует;
- Аппаратный TCM (Tightly Coupled Memory)
- Аппаратные MMU и TLB отсутствуют;
- Аппаратный FPU одинарной точности;
- Поддержка ОС РВ (RTEMS, FreeRTOS, Zephyr и т.п.).

ЦПУ реального времени с такими характеристиками закрывает потребности в процессорах ЦОС, используемых при реализации алгоритмов обработки данных радиотракта в аппаратуре систем связи, он также подходит для использования в других областях применения в качестве процессора ЦОС или управляющего ЦПУ, работающего под управлением ОС РВ.

3.Блоки DSP

Блоки DSP, реализующие операцию умножения с накоплением, являются аппаратным базисом для широкого спектра алгоритмов цифровой обработки сигналов. Набор стандартных и часто используемых алгоритмов ЦОС (например, БПФ/ОБПФ, КИХ-фильтры, КИГ-фильтры, Cordic и др.) входит в состав программных СФ блоков, поддерживаемых на уровне САПР конфигурируемой схемы.

В гетерогенной конфигурируемой схеме блок DSP реализуется в виде операции умножения 18-битного и 25-битного входных операндов, с последующей операцией суммирования с получением 48-битного результата. Такая реализация позволяет эффективно использовать алгоритмы ЦОС в формате с фиксированной точкой, нивелируя эффекты, связанные с ошибками округления данных, за счет использования одного из входных операндов большей разрядности (25-бит). Это также позволяет эффективно использовать данные блоки при реализации алгоритмов ЦОС в формате с плавающей точкой одинарной точности, для чего потребуется лишь 2 блока DSP48 формата 18х25, вместо 4-х блоков DSP48 формата 18x18. Дополнительно в состав блоков DSP включен предварительный сумматор, наличие которого позволяет сократить вдвое количество блоков DSP при реализации КИХ-фильтров с симметричными коэффициентами.

Применительно к аппаратуре систем связи важной характеристикой является количество блоков DSP и объем встроенного O3У, которые требуются для воплощения специализированных алгоритмов обработки данных радиотракта (алгоритмы корреляции, модуляторы, демодуляторы и

т.п.), поэтому количество блоков DSP гетерогенной схемы должен быть не менее 400 для данной области применения.

Для других областей применения наличие большого количества блоков DSP потребуется при активном использовании большинства алгоритмов ЦОС (в особенности, БПФ/ОБПФ с большим количеством отсчетов), цифрового диаграммообразования (ЦДО, широко используемого в аппаратуре ЦАР/АФАР), многоканальной обработки данных и т.п.

4.Блоки ОЗУ

В гетерогенной конфигурируемой схеме блок ОЗУ имеет емкость 36Кбит и обладает большими возможностями для его конфигурирования, в результате чего на его основе может быть реализован широкий набор разновидностей блоков памяти различных организаций и разрядностей (FIFO, ПЗУ, однопортовое ОЗУ, двухпортовое ОЗУ и т.п.).

Применительно к аппаратуре систем связи важной характеристикой является объем встроенного ОЗУ, который требуется для реализации специализированных алгоритмов обработки данных радиотракта (алгоритмы корреляции, модуляторы, демодуляторы и т.п.), поэтому объем блочного ОЗУ гетерогенной ИС должен быть не менее 8Мбит.

Для других областей применения наличие большого объема блочного ОЗУ конфигурируемой схемы потребуется при активном использовании некоторых алгоритмов ЦОС, для расширения внутренней памяти ЦПУ, для буферизации данных, передаваемых по высокоскоростным интерфейсам (например, Ethernet, PCI Express, JESD204B и т.п.).

5.Блоки высокоскоростных приемопередатчиков (SerDes)

Блоки высокоскоростных приемопередатчиков являются физической базой для реализации всех интерфейсов обмена данными на скоростях, близких к 1Гбит/с и выше. Данные блоки являются преимущественно аналоговыми, поэтому их реализация возможна только в виде полностью заказных СФ блоков. В составе гетерогенной ИС данные блоки реализуются в виде 8b/10b SerDes, используемых в реализации интерфейса физического

уровня (PHY) многих стандартных интерфейсов и протоколов (Ethernet 1000BASE-X, PCI Express, JESD204B и т.п.). Использование блоков SerDes разработчиками РЭА является затруднительным (ввиду сложности их функционирования и аналоговому поведению), поэтому на их основе могут быть реализованы стандартные интерфейсы и протоколы высокого уровня (Ethernet, JESD204B и др.) в виде программных СФ блоков.

В составе гетерогенной ПЛИС блоки SerDes обладают скоростью приема/передачи данных более 1,25Гбит/с. Уровни входных/выходных сигналов соответствуют уровням CML (Current Mode Logic).

Из анализа требований к аппаратуре систем связи 6-го поколения следует необходимость наличия в структуре гетерогенной конфигурируемой схемы СФ блока контроллера внешней динамической памяти (DDR или DDR3). Структурная схема гетерогенной конфигурируемой интегральной схемы современной системы связи 6-го поколения представлена на рисунке 2.4.



Рисунок 2.4 – Структурная схема гетерогенной конфигурируемой интегральной схемы современной системы связи 6-го поколения

2.2 Методы предварительного анализа конфигурируемых схем. Критерии и метрики оценки эффективности различных решений построения конфигурируемых схем

Для оценки требуемых параметров архитектуры и состава базового кристалла конфигурируемой схемы на основе анализа схем заказчиков

предлагается использовать пять методов и, соответственно, пять критериев и метрик предварительного анализа:

- метод оценки размера логического блока на основе правила Рента;
- определение оптимального размера логического блока с помощью итерационного алгоритма (определение количества входов/выходов и количества внутренних цепей логических блоков);
- метод оценки числа цепей, пересекающих вертикальные/горизонтальные разрезы (сечения) базового кристалла;
- оценка уровня перегруженности кристалла;
- оценка карты загруженности ключей коммутации в точках изгиба цепей.

Следует отметить, что начальными данными для предварительной оценки является структура логического элемента, общее количество логических элементов в базовом кристалле и форм-фактор базового кристалла.

Оценка трассировочных ресурсов на основе правила Рента.

Для грубой оценки требуемых трассировочных ресурсов базового кристалла будем использовать правило Рента, то есть эмпирическую зависимость вида $T = tB^p$, где $T = |P_r(s_{mu})|$ – количество сигнальных или трассировочных выводов схемы заказчика s_{mu} , $B = |E(s_{mu})|$ – количество элементов схемы заказчика s_{mu} , t – коэффициент Рента, p – параметр (экспонента) Рента.

Для определения параметра Рента некоторого проекта заказчика $\Pi_u = (S_u, L_u, s_{mu}, m_{mu})$ используется следующий алгоритм:

- на первом шаге формируется множество подсхем K={K_i}:
 ∀<u>i</u>∈ {0,1,...,|K|}: K_i⊆E(s_{mu}),
- на втором шаге строится график зависимости $T_i = |P_r(K_i)|$ от $|K_i|$ в двойном логарифмическом масштабе (рисунок 2.5),

и на финальном этапе вычисляется параметр Рента как угловой коэффициент полученной прямой:



 $p = \frac{\partial \ln T_i}{\partial \ln |K_i|}$

Рисунок 2.5 – График зависимости T_i от $|K_i|$ в двойном логарифмическом масштабе для схемы mega core (4086 логических элементов, 4123 цепей, *p*=0.6359)

Значения параметров Рента и вид кривой Рента зависят как от электрической схемы, так и от метода декомпозиции схемы заказчика на кластеры. Используя правило Рента и правило, описывающее убывание цепей коммутации конкретного блока в зависимости от расстояния до него, можно определить оптимальное разбиение составной части схемы заказчика на логические блоки.

Определение оптимального размера логического блока с помощью итерационного алгоритма. На этапе предварительного анализа базового кристалла размер и форма логического блока не заданы и по умолчанию после плоского размещения принимают некоторые значения. В ходе анализа определяется количество фактически задействованных входов/выходов ЛБ, количество внутренних цепей ЛБ (соединения, контакты которых находятся одного блока ЛБ). Итерационный алгоритм внутри предполагает использование комбинации алгоритмов кластеризации для формирования

логических блоков и размещения логических элементов внутри них с перебором различных вариантов их размеров и форм.

Оценка требуемой пропускной способности сечений базового кристалла. Требуемая пропускная способность разрезов (сечений) базового кристалла определяется путем подсчета числа проектных цепей, проходящих между рядами/столбцами логических блоков после процедуры их размещения. Каждая проектная цепь в разрезе учитывается не более одного раза. Таким образом, для заданной схемы заказчика s_{mu} пропускная способность вертикальных σ_X и горизонтальных σ_Y сечений вычисляется по формулам:

$$\sigma_X(s_{mu}, X_S^k) = |\{n: n \in N(s_{mu}) \& X_{min}(n) < X_S^k \& X_{max}(n) \ge X_S^k\}|$$

$$\sigma_Y(s_{mu}, Y_S^k) = |\{n: n \in N(s_{mu}) \& Y_{min}(n) < Y_S^k \& Y_{max}(n) \ge Y_S^k\}|,$$

где X_S^k, Y_S^k – координаты вертикальных/горизонтальных сечений,
 $X_{min}(n), Y_{min}(n), X_{max}(n), Y_{max}(n)$ – координаты цепей $n \in N(s_{mu}).$

Таблица 2.1 - Сводные данные нагруженности вертикальных и горизонтальных сечений

			_	
Схема	Среднее	Максимальный	Среднее	Максимальный
	значение	вертикальный	значение	горизонтальный
	вертикального	разрез	горизонтального	разрез
	разреза		разреза	
altor32_alu	77	122	117	146
module_i2c	100	136	124	177
ClaiRISC_core	90	137	146	186
microcode	116	187	150	233
pipeline_if	64	92	88	111
a23_decode	165	222	269	380
execute	541	901	538	875
a25_decode	179	234	222	272
pc_dma	75	100	129	173
sdrc_core	222	315	213	285
bch_dec_dcd_	107	102	172	247
univ_top	107	182	172	247
decode	103	163	133	180
a23_fetch	215	274	187	237
a25_fetch	250	341	260	362
cpu8080	252	327	240	300

pipeline_exe	125	235	208	308
altor32_exec	77	124	105	130
MC6809_cpu	81	117	153	211
FT816Float	116	153	119	161
altor32_regfile _sim	118	219	103	132
mega_core	117	165	155	212

В таблице 2.1 оценка производится на основе медианных и максимальных значений требуемой пропускной способности кристалла для набора типовых схем заказчиков.

Разрезы (сечения) базового кристалла проводятся в областях между рядами/столбцами ЛЭ, поскольку на начальной стадии разработки базовый кристалл не содержит информации о виде, размере и количестве логических блоков. Это необходимо для первоначальной мелкозернистой оценки требований к трассировочным ресурсам без учета их деления на верхний (межблочный) и нижний (внутриблочный) уровень.

Исходя из максимального значения расчетной пропускной способности сечения, можно вычислить максимальное требуемое количество соединений между соседними блоками логических элементов.

Оценка уровня перегруженности кристалла. Важной метрикой является уровень перегруженности областей размещения кристалла. Карта уровней строится следующим образом. Вся площадь кристалла делится на области таким образом, чтобы в каждую из них входил один логический блок. Тогда перегруженность области равна количеству цепей, в охватывающий прямоугольник которых попадает данная область (рисунок 2.6). Чем выше перегруженность области, тем больше предполагаемое количество цепей, проходящих через нее.


Рисунок 2.6 – Карта уровней перегруженности областей размещения базового кристалла для тестовой схемы

Оценка карты загруженности ключей коммутации в точках изгиба цепей. Карта загруженности ключей коммутации похожа на карту перегруженности областей, но рассчитывается уже для двухтерминальных цепей, на которые разбиваются реальные проектные цепи. Для каждой подобласти (пространства кристалла, находящегося между четырьмя блоками ЛБ) загруженность ключей равна сумме двухтерминальных Гобразных цепей, которые выполняют поворот в данной области кристалла (рисунок 2.7). Таким образом, значения карте соответствуют В максимальному количеству поворотов, которые должны быть доступны в переключательных блоках.



Рисунок 2.7 – Карта загруженности ключей для тестовой схемы

2.3 Метод программного прототипирования конфигурируемых схем

Значительно упростить и ускорить процесс разработки базового кристалла ПЛИС позволяют методы предварительного анализа и оценки архитектуры. Однако разработанные в работе методы предварительного анализа используют упрощенное описание базовых схем, что не позволяет точно оценить их архитектуру и особенности построения. Метод программного прототипирования, который лежит в основе нового этапа маршрута разработки конфигурируемых схем, предполагает точное описание и оперативное изменение архитектуры разрабатываемого базового кристалла, а также оценку эффективности таких изменений до фактического изготовления.

Формализация представления схем в САПР, поддержка языка CDL (Circuit Design Language) позволяют настроить программное обеспечение на соответствующие изменения в конструкции, схемотехнике и топологии разрабатываемой схемы и оценить возможность реализации различных схем заказчиков в базовом кристалле и оптимальность самой архитектуры базового кристалла.

Разработка архитектуры проблемно-ориентированных гетерогенных конфигурируемых интегральных схем представляет собой сложную задачу, которая решается в условиях жестких технологических ограничений и экономических требований [47] – [50]. Методам и моделям, используемым в процессе создания новых архитектур, посвящено немало научных исследований [51] - [54]. В подходе, используемом в данной работе, при архитектуры базовых кристаллов проблемно-ориентированных поиске конфигурируемых интегральных схем предлагается учитывать конкретные задачи и специфику целевого потребителя.

Метод программного прототипирования состоит из нескольких этапов.

На первом этапе происходит передача информации об эскизной схеме базового кристалла, разработанной на этапе предварительного анализа маршрута проектирования конфигурируемой микросхемы, в базу данных САПР, в которой схемотехническое описание базового кристалла в формате CDL и его топология в формате GDSII обрабатываются и анализируются. При обработке этих файлов структура программы автоматически подстраивается под архитектуру базового кристалла, формируя граф коммутаций, координаты логических блоков и карту памяти, на основании будет которой формироваться вектор прошивки. Возможность

автоматической подстройки под любую архитектуру позволяет разработчикам конфигурируемых схем заранее оценивать их трассируемость и находить слабые места архитектуры, а разработчикам САПР заранее отлаживать программное обеспечение под схемы заказчиков.

На следующем этапе проводится логический и топологический синтез и выполняется полный маршрут проектирования схемы заказчика [55]. Проводится графовая трансляция и технологическое отображение в базисе кристалла, а также декомпозиция списка соединений на отдельные группы или кластеры, размещение логических элементов на легальные позиции матрицы базового кристалла и трассировка соединений между ЛЭ с использованием коммутационных ресурсов, заложенных в архитектуре.

На заключительном этапе программного прототипирования анализируются полученные результаты, изменяется архитектура и модифицируется схемотехническое описание базового кристалла.

Программное прототипирование осуществляется в несколько итераций. При успешном выполнении полного маршрута проектирования схем заказчиков из набора схем определенного класса и соответствии результатов прототипирования всем заданным требованиям и ограничениям процесс программного прототипирования завершается. В рамках программного проготипирования определяются значения ключевых параметров архитектуры базового кристалла, необходимые для эффективной реализации заданного комплекта схем заказчиков.

В качестве стратегии при оптимизации параметров функциональных блоков базового кристалла в программном прототипировании используется приближения, метод последовательного который основан на последовательной оптимизации характеристик функциональных блоков по мере убывания их важности. В исходной архитектуре базового кристалла задаются начальные значения характеристик основных функциональных эффективности единиц. Определяются критерии оценки решений. Оптимизируемые функциональные единицы располагаются в порядке

убывания их важности. Назначаются допустимые значения снижения эффективности (уступки) для каждой функциональной единицы. Производится последовательная оптимизация характеристик по мере убывания важности функциональных единиц с учетом уступок.

Программное прототипирование позволяет использовать набор типичных проектных схем заказчиков в качестве основы для определения базовых параметров архитектуры разрабатываемых кристаллов. В работе представлен математический аппарат, основывающийся на теоретикомножественном подходе к обобщенному описанию проекта заказчика и самого базового кристалла. В рамках введенных формальных обозначений поставлена задача трассировки в составе топологического проектирования в базисе гетерогенных конфигурируемых интегральных схем. Математическая модель трассировочных ресурсов базового кристалла позволяет учесть широкий спектр способов программирования коммутационных элементов, а также разнообразие выполняемых ими логических функций.

В рамках программного прототипирования, наряду со стандартным контролем параметров схемы заказчика по отношению к параметрам обратная базового решается задача, кристалла, заключающаяся В значений ключевых параметров базового определении архитектуры кристалла, необходимых для эффективной реализации заданного набора схем заказчиков. Этап трассировки в этом случае предназначен для определения степени разводимости набора схем заказчиков в базисе прототипа разрабатываемого базового кристалла и формирования ограничений на его трассировочные ресурсы.

Основополагающей идеей программного прототипирования является анализ разводимости наборов схем заказчиков в базовом кристалле. Для решения задачи поле ПЛИС может быть представлено двумя способами: упрощенно в виде плоской матрицы логических элементов заранее определенного вида (например, 4-входовой LUT – элемент, реализующий таблицу истинности – с триггером); в виде двухуровневого блочного

представления. Как правило, потенциал двухуровневого представления наилучшим образом описывает островной тип архитектуры. В этом случае верхний уровень архитектуры кристалла может быть представлен в виде матрицы блоков логических элементов.

Среди характеристик схем заказчиков и базового кристалла необходимо выделить такие, которые существенным образом влияют на исход процедуры трассировки. Характеристики схем заказчиков, как правило, рассчитываются динамически на этапах размещения, глобальной трассировки (в случае островной архитектуры базового кристалла) или в процессе окончательной трассировки.

Основные характеристики схем заказчиков:

- требуемое количество элементарных связей между двумя логическими элементами или кластерами элементов (при превышении требуемого числа связей над фактически имеющимися в базовом кристалле в *N* раз, оценка трассируемости такой схемы будет различаться [56]), а также требуемая пропускная способность канала (или блока коммутации), требуемая связность блоков коммутации с каналами в маршруте для островной архитектуры;
- требуемая пропускная способность сечения по вертикали для заданного
 X, требуемая пропускная способность сечения по горизонтали для заданного Y (соответствующая характеристика базового кристалла фактическая пропускная способность вертикальных и горизонтальных сечений);
- ограничение на задержку критического пути (частоту работы устройства);
- степень ветвления цепей схемы заказчика (может быть использована для быстрой сортировки списка проектных соединений [57]);
- степень перегруженности отдельных трассировочных элементов, а также каналов для островной архитектуры, возникающая в процессе трассировки схемы заказчика на каждом из этапов трассировки [58],

[59], или как предварительная оценка результатов трассировки в целом на этапах глобальной трассировки [60] и размещения;

остаточный путь до приемника, рассчитываемый динамически в процессе трассировки для ускорения трассировки за счет направленности поиска пути [56].

Значительно упростить и ускорить процесс разработки базового кристалла позволяют предварительного методы анализа И оценки архитектуры. Существующие методы оценки, основанные на прохождении полного маршрута проектирования [10] – [13], используют упрощенное описание базовых схем, что не позволяет точно оценить их архитектуру, её тонкости и слабые места. Имеющиеся методы требуют описания схемы в специализированном формате, что ставит перед разработчиком архитектуры изучение дополнительную задачу – новых методов представления разработанного схемотехнического описания схемы.

Предложенный подход к оценке архитектуры базового кристалла использует формат схемотехнического описания схем – CDL (Circuit Design Language) [61], который широко распространён среди разработчиков интегральных схем (ИС) и используется в системах автоматического проектирования (САПР) компаний Cadence, Synopsys, Mentor Graphics. Описание схемы в данном формате автоматически генерируется из её графического представления в схемотехническом редакторе указанных компаний. Поддержка языка CDL, наряду с разработанной формализацией представления схем в САПР, позволяет гибко провести оперативную настройку программного обеспечения на соответствующие изменения в конструкции, схемотехнике и топологии разрабатываемой конфигурируемой схемы и оценить как эффективность реализации различных проектируемых схем заказчиков, так и эффективность самой архитектуры базового кристалла.

Разработанный метод, включающий в себя оперативную настройку на изменения архитектуры схемы и оценку ее эффективности, является новым

этапом маршрута проектирования конфигурируемой интегральной схемы. В отличие от классического прототипирования, подразумевающего тестирование системы на кристалле или ее отдельных СФ блоков в базисе готового кристалла ПЛИС [62] - [64], программное прототипирование включает в себя тестирование и оценку самого базового кристалла, содержащего элементы ПЛИС до его фактического изготовления.

Метод программного прототипирования, предложенный в данной работе, состоит из нескольких шагов.

- 1) Создается базовая архитектура, на основе которой будут разрабатываться прототипы и выполняться дальнейшие модификации. Архитектура может быть как уникальной, так и выбранной из множества существующих решений, отличающихся структурой трассировочных ресурсов [65] - [66] (островные и иерархические ПЛИС), структурой логического элемента (ЛЭ) и группы логических блоков (ГЛБ) (ЛЭ в ПЛИС фирмы Altera [67], конфигурационный логический блок в ПЛИС фирмы Xilinx [68], универсальные логические блоки (VersaTile) у фирмы Microsemi [69]). Выбрать архитектуру конечной схемы среди всего многообразия позволяют физические ограничения, заключающиеся в размере корпуса или в площади кристалла, необходимой для размещения конфигурационной памяти. Ограничения накладываются с учетом требований, заданных конкретными проектами заказчиков, которые выражаются в количестве программируемой логики, коммутационных возможностях схемы и определённом наборе СФ блоков.
- 2) На втором шаге происходит передача информации о разработанной САПР. схеме В базы данных Первоначально, с помощью специализированного программного обеспечения В САПР обрабатывается и анализируется схемотехническое описание схемы базового кристалла в формате CDL и ее топология в формате GDSII Структура программы [70]. автоматически подстраивается под

архитектуру ПЛИС, формируя граф коммутаций, координаты ЛБ и карту памяти, на основании которой будет формироваться вектор прошивки. Возможность автоматической подстройки под любую архитектуру позволяет разработчикам конфигурируемых схем заранее оценивать их трассируемость и находить слабые места архитектуры, а разработчикам САПР заранее отлаживать ПО под нужны заказчиков.

- 3) На следующем шаге выполняется полный маршрут разработки схемы заказчика, включающий в себя логический и топологический синтез [71]. Логический синтез состоит из графовой трансляции и технологического отображения в базисе целевого базового кристалла [72] – [73]. Топологический синтез, в свою очередь, состоит из декомпозиции списка соединений на отдельные группы или кластеры [74], размещения логических элементов на легальные позиции матрицы ПЛИС [75], трассировки соединений между ЛЭ с использованием коммутационных ресурсов, заложенных в архитектуре [76].
- 4) Завершающим шагом программного прототипирования является анализ полученных результатов, на основании которого подбираются новые параметры архитектуры и вносятся соответствующие модификации в схемотехническое описание базового кристалла. Программное прототипирование – процесс итерационный, поэтому этап внесения изменений в конструкцию схемы лишь завершает одну итерацию подбора подходящей архитектуры. Процесс прототипирования можно считать завершенным при соблюдении двух условий. Первым условием является соответствие результатов прототипирования всем заданным требованиям и ограничениям. Вторым условием является успешное выполнение полного маршрута разработки для ряда схем заказчиков. В случае несоблюдения этих условий в архитектуру базового кристалла вносятся соответствующие изменения и процесс повторяется до получения положительного результата.

Оперативную настройку конструкции и схемотехники базового кристалла на требования заказчиков, а также оперативную настройку САПР на соответствующие изменения в конструкции, схемотехнике и топологии базового кристалла обеспечивает формализация соответствий между элементами базового кристалла конфигурируемой гетерогенной интегральной схемы и проектируемыми схемами заказчиков.

На основе формализованного представления базового кристалла и проектируемой схемы были разработаны программные средства САПР, позволяющие реализовать предложенный метод. В качестве примера, эффективность показывающего разработанного метода, приведены результаты разработки базового кристалла с итерационным изменением его архитектуры. Ближайший аналог исходного базового кристалла – ПЛИС MAX II компании Altera, имеющая с ним схожую структуру ЛЭ, ГЛБ и трассировочных ресурсов. Целью изменений архитектуры является уменьшение объема требуемой конфигурационной памяти и наращивание логической емкости базовой схемы без потери достигнутого уровня трассируемости схем заказчиков на её основе.

В процессе прототипирования проводились изменения структуры ГЛБ и трассировочной архитектуры кристалла, в которую входят следующие типы межсоединений: локальная шина (Local), прямые связи (Direct Link – DL), шины R4C4 и R8C8 (R– row, C – column), длинные связи (LongBus), диагональные связи.

Также наряду с разрядностью перечисленных шин проводились изменения в структуре коммутаторов, аккумулирующих эти соединения. В данной архитектуре присутствуют три типа коммутаторов:

• switch_block (SB) – блок, объединяющий между собой шины R4C4/R8C8 и позволяющий соединить прямые связи с этими шинами;

• connection block (CB) – блок, собирающий шины R4C4/R8C8, прямые и длинные связи в локальную шину внутри ГЛБ;

• local connection block (LCB) – блок, распределяющий приходящие на локальную шины сигналы по всем необходимым ЛЭ в ГЛБ.

Локальная шина обеспечивает связь между ЛЭ внутри ГЛБ. Она подключена не только к каждому ЛЭ по отдельности, но и к глобальным межсоединениям строк и столбцов. Это обеспечивает прямую связь между ГЛБ и сводит использование глобальных шин к минимуму.

При этом ГЛБ могут быть соединены друг с другом в пределах одной строки двумя способами:

- прямая связь по локальной шине;
- шина R4, охватывающая четыре ГЛБ слева, четыре справа;
- шина R8, охватывающая восемь ГЛБ слева, восемь справа.

Прямая связь дает доступ к локальным шинам соседних ГЛБ, расположенным слева и справа, а также обеспечивает быструю передачу данных между ГЛБ и/или блоками ввода/вывода, не взаимодействуя с шинами R4 и R8. Каждая ГЛБ имеет соединения с шинами R4/R8 как в левую, так и в правую сторону.

Структура столбцовых межсоединений аналогична структуре строчных. Отличие лишь в том, что вместо шин R4/R8 для них используются шины C4/C8, которая позволяет соединить соседние ГЛБ в пределах одного столбца, охватывая четыре/восемь соседних ГЛБ вверх и столько же ГЛБ вниз.

Регулярная структура соединений в виде строк и столбцов фиксированной длины, позволяет предсказать точное время коротких задержек в распространении сигнала.

Имеющиеся в архитектуре длинные связи проходят через весь столбец или строку кристалла, позволяя соединить между собой дальние ЛЭ.

В качестве исходной схемы был взят прототип со следующими характеристиками трассировочной архитектуры: разрядность шин R4/C4 – 32 бита, разрядность шин R8/C8 – 64 бита, разрядность длинных связей – 10 бит, прямых связей – 10 бит, локальной шины– 22 бита. При этом пропускная

коммутационная способность блока локальных связей не равна 100%. Его структура разрежена, что снижает пропускную способность до 75%. Данная схема имеет 16 столбцов и 20 строк ГЛБ, при этом каждая ГЛБ состоит из 10 ЛЭ. Общая площадь схемы равна 3200 ЛЭ.

Прототипирование выполнялось с использованием тестовой проектной схемы s38417 из набора ISCAS'89 [77]. Результат её логического синтеза составляет 3184 ЛЭ и 3215 межсоединений.

Результаты прототипирования показаны в таблице 2.2. Она состоит из столбцов с именами текущего прототипа и прототипа, на основе которого он разработан, описания текущего прототипа и результатов анализа имплементации схемы на его основе. Результаты представлены в виде количества неразведённый цепей и объема конфигурационной памяти, приходящегося на одну ГЛБ и в среднем на один элемент этой группы.

Таблица 2.2 - Результаты программного прототипирования базового кристалла размером 16х20 ГЛБ

Имя прототипа	Имя пред. прототипа	Описание прототипа	Количество неразведенных цепей, шт.	Объем памяти на ЛЭ/ГЛБ, бит
1.0	-	Исходный базовый кристалл R4C4 = 32, R8C8 = 64, LongBus = 10, DL = 10, Local = 22	0	291,3 / 2913
1.1	1.0	Редукция коммутатора CB R4C4 = 32, R8C8 = 32 , LongBus = 10, DL = 10, Local = 22	114	278,7 / 2787
1.2	1.1	Редукция коммутатора CB R4C4 = 16, R8C8 = 32, LongBus = 10, DL = 10, Local = 22		252,5 / 2525
1.3	1.1	Редукция коммутатора CB R4C4 = 16, R6C6 = 24 , LongBus = 10, DL = 10, Local = 22		249,3 / 2493
1.4	1.3	Редукция коммутатора CB R3C3 = 24, R6C6 = 24, LongBus = 10, DL = 10, Local = 22	111	268,5 / 2685
1.5	1.4	Редукция коммутатора СВ R3C3 = 24, R6C6 = 24, LongBus = 10, DL = 10, Local = 22	113	249,3 / 2493

		R3C3 = 24, R6C6 = 24, LongBus = 10,			
16		DL = 5, Local = 22			
		добавлены связи DL ↗и ∠ (до SB) =			
		5			
	1.5		114	244,9 /	
	110	DL $(\land, \uparrow, \nearrow, \land \operatorname{do} SB, \nearrow \operatorname{do} SB)$		2449	
		доступны у 5 верхних ЛЭ из ГЛБ.			
		$\mathbf{D}\mathbf{I} (\boldsymbol{\zeta} \mid \boldsymbol{\lambda} \boldsymbol{\zeta} = \mathbf{S} \boldsymbol{R} \boldsymbol{\lambda} = \mathbf{S} \boldsymbol{R} $			
		DL $(2, 4, 3, 203B, 3003B)$			
		$R_{3}C_{3} = 24, R_{6}C_{6} = 24, LongBus = 10.$			
1.7	1.6	DL = 5, $Local = 22$	112	233,77	
		Удалены связи ∧ и ∠ до SB		2337	
		R3C3 = 24, R6C6 = 24, LongBus = 10,		246.1 /	
1.8	1.6	DL (\leftarrow , \rightarrow) = 10, DL= 5, DL \nearrow и \checkmark	113	240,17	
		(go SB) = 5, Local = 22,		2101	
1.0	1.0	R3C3 = 24, R6C6 = 24, LongBus = 10,	110	234,9 /	
1.9	1.8	$DL (\leftarrow, \rightarrow) = 10, DL = 5, Local = 22$	110	2349	
		удалены связи DL / $u \neq (д0 SD) - S$ R3C3 = 24 R6C6 = 24 LongBus = 10		2471/	
1.10	1.6	DL((5, 7, 4, 5)) = 5, DL=10, Local = 22	108	2471	
	1.10	$R_{3}C_{3} = 24, R_{6}C_{6} = 24, LongBus = 10,$			
1.11		DL((5, 2, 2, 5)) = 5, DL=10, Local = 22	114	235,9 /	
		удалены связи DL ↗и ✓ (до SB) = 5		2359	
	1.0	R4C4 = 32, R8C8 = 64, LongBus = 10,			
1.12		DL = 5, Local = 22	112	286,9 /	
		добавлены связи DL ∧и ∠ (до SB) =		2869	
		5 $P_{4}C_{4} = 22$ $P_{8}C_{8}^{0} = 64$ LongPug = 10			
1 1 3	1.12	R4C4 - 52, R6C6 - 64, LoligBus - 10, DL = 5 Local = 22	114	274,1 /	
1.15		улалены связи DL Ли ✓ (ло SR)	117	2741	
		Редукция коммутатора			
		connection_block			
		R3C3 = 24, R6C6 = 24, LongBus = 10,			
		$\mathbf{DL} = 4, \text{Local} = 22$			
		дооавлены связи DL / и \checkmark (до SB) =			
		1			
1.14		DL (лля направлений 5.1.7.5	115	235,3 /	
		до SB, 7 до SB) доступны у 4		2353	
		верхних ЛЭ из ГЛБ.			
		DL $(\checkmark, \downarrow, \lor, \checkmark, \checkmark \operatorname{do} SB, \lor \operatorname{do} SB)$			
		доступны у 4 нижних ЛЭ из ГЛБ			
		иентральным ЛЭ (3, 4, 5, 6 ЛЭ)			
		$R_{3}C_{3} = 24, R_{6}C_{6} = 24, LongBus = 10,$			
1.15	1.14	DL = 4, DL до SB = 8 , Local = 22	118	227,3 /	
		DL до SB внутри ГЛБ сокращена	110	2273	
		до 8			

1.16	1.15	Добавлен полный коммутатор на DL во все стороны P3C3 – 24 Р6C6 – 24 LongBus – 8	112	275,3 / 2753
		DL = 4, DL go $SB = 8$, $Local = 22$		2133
1.17	1 16	количество ЛЭ в ГЛБ увеличено до		
		16.	0	260 / 1311
	1.10	R4C4 = 32, R8C8 = 64, LongBus = 8,	0	209/4314
		DL = 4, DL до SB = 8, Local = 22		
1.18	1 17	R3C3 = 24, R6C6 = 48 , LongBus = 8,	0	275 / 4114
	1.1/	DL = 4, DL до $SB = 8$, $Local = 22$	0	2/5/4114

В представленной таблице жирным шрифтом выделены модификации текущего прототипа относительно предыдущего, указанного В соответствующем столбце. Из таблицы видно, что в прототипах 1.1-1.5 уменьшение памяти достигалось за счет редукции коммутатора connection block и уменьшения дальности и разрядности шин R/C. На прототипах 1.2-1.3 выявлено, что используемая в них длина и разрядность связей R/C недопустима, так как в этом случае трассируемость значительно падает.

В прототипах 1.6-1.15 предпринята попытка уменьшить объем занимаемой памяти, не потеряв трассируемость, за счет сокращения прямых связей. Во-первых, их разрядность уменьшается до 5 бит, а во-вторых, только одна половина ЛЭ имеет прямые связи вверх и вверх по диагоналям, а другая половина – только вниз и вниз по диагоналям. Подробнее доступные связи прототипа 1.6 показаны на рисунке 2.8.



Рисунок 2.8 - Схематическое изображение структуры прямых связей прототипа 1.6

Вследствие того, что при дальнейшей редукции связей DL, трассируемость будет только падать, в прототипе 1.16 снова добавлен полный коммутатор прямых связей, позволяющий каждому ЛЭ в ГЛБ соединиться с соседним. При этом, в целях экономии конфигурационной памяти, разрядность длинных связей сокращена до 8.

Уровень трассируемости, который в отличие от исходного кристалла, на прототипах 1.1-1.16 упал до ~113 не трассируемых цепей, повысило увеличение длин и разрядностей шин R/C, а также увеличение размера ГЛБ до 16 ЛЭ.

При достижении полной трассируемости, помимо сокращения объёма памяти, добавляется задача уменьшения максимальной и средней длины цепей. Продолжение процесса программного прототипирования с учетом новых поставленных задач показано в таблице 2.3. За исходный базовый кристалл взят прототип 1.16, с увеличением размера ГЛБ до 16 ЛЭ, количества ГЛБ и пропорциональным добавлением разрядностей

коммутационным шинам. Общая площадь такой схемы равна 4096 ЛЭ. В таблице отсутствует столбец с названием предыдущего прототипа, так как модификации вносятся последовательно в предыдущий прототип. Также вследствие того, что во всех прототипах полностью разведены все тестовые схемы, в таблице отсутствует столбец с количеством не трассируемых цепей.

Таблица 2.3 - Результаты программного прототипирования базового кристалла размером 16х16 ГЛБ

		F	FFFF			
		Объем	5 максимальных по длине			
Имя		памяти	цепей /			
прототи	Описание прототипа	на	средняя длина цепи			
па		ЛЭ/ГЛБ, бит	S38417	Ac97		
	В LAB 16 LE с геометрией		26/24/24/24/	77/77/77/77/		
	кристалла 16х16.	278 /	23	27		
2.1	R3C3 = 24, R6C6 = 48,	4454	23	27		
	LongBus = 8, $DL = 4$, Local = 34		8,56	9,97		
			27/27/27/23/	37/37/37/37/		
2.2	Добавлены прямые связи с	278 /	22	37		
2.2	LongBus на IO блоки	4454	/	/		
			8,52	10,25		
	Domining Kongerrano SD		19/19/19/19/	25/25/25/25/		
22		264 / 4230	19	25		
2.3	гедукция коммутатора ЗВ		/	/		
			7,54	8,42		
2.4	В коммутаторе SB	259 /	22/19/19/19/	28/28/27/27/		
	сокращены повороты шин	4150	19	27		
	R3C3 и R6C6 с 8 разрядов	7150	/	/		
	до 4		7,55	8,66		
	Сокращение локальных связей внутри ГЛБ с 10 до		22/19/19/19/	31/28/28/27/		
2.5		252 / 4042	19	27		
	8		/	/		
			8,31	9,60		
2.6	Релукция LCB < 75%		22/22/19/19/	28/28/28/28/		
		244 /	19	27		
		3914	/	/		
			8,31	9,68		

В качестве схемы заказчика для прототипирования была выбрана схема ac97 [78], результат логического синтеза которой составил 3732 ЛЭ и 3821 цепи.

В таблице 2.3 показано, что на этом этапе прототипирования уменьшение памяти достигается за счет редукции коммутаторов SB и LCB, а также изменением разрядности локальных связей и разрядности шин R3C3 и R6C6 на пересечении соединений строк и столбцов.

В результате программного прототипирования был выбран базовый кристалл, разработанный на основе прототипа 2.6. Средний объем памяти на один ЛЭ в этом прототипе меньше исходного на 47,3 бита. При этом не утрачен исходный уровень трассируемости межсоединений, а общий размер кристалла увеличен на 896 ЛЭ.

Таким образом, программное прототипирование позволило выбрать архитектуру базового кристалла, удовлетворяющую всем заданным требованиям, до разработки топологии и изготовления конфигурируемой схемы.

2.4 Новые архитектурные решения. Архитектура программируемой интегральной схемы типа «жук»

При разработке архитектуры конфигурируемой интегральной схемы и, в частности, организации цепей маршрутизации следует стремиться не только к уменьшению коммутационных ресурсов, но и к улучшению качества связей. Оптимизация шин коммутации и применение неполных коммутаторов с целью уменьшения площади базового кристалла может приводить к увеличению числа ключей и, соответственно, к увеличению длины связей. Это приводит не только к неэффективному использованию шин коммутации между логическими элементами, но и оказывает большое влияние на быстродействие, так как задержка на блоках коммутации вносит наибольший вклад в результирующую задержку распространения сигнала.

Диаграмма связанности конфигурируемой схемы, построенной по архитектуре островного типа с прямыми линиями коммутации (RC) между

блоками длиной 3,6,9 LAB, представлена на рисунке 2.9. Улучшение качества связей и уменьшение количества коммутационных ресурсов может быть достигнуто путем введения диагональных связей, подобных коротким связям (Direct Links), но с длинами 2 и 3 LAB. На рисунке 2.10 представлена диаграмма связанности модифицированной архитектура типа «Жук» с добавленными диагональными линиями коммутации (по 2 на каждое направление). В ней большее количество блоков LAB может быть подключено с использованием меньшего количества точек поворота и ключей, соответственно. Для подсоединения к диагональным линиям может быть использован уже существующий вывод коммутатора с выходов ЛЭ на RC линии, или создан выделенный. В первом случае увеличение площади ПЛИС будет меньше, но во втором гибкость маршрутизации будет выше, что дает дополнительное улучшение качества связей. Результаты исследования для первого случая приведены в таблице 2.4.



Рисунок 2.9 - Диаграмма связанности конфигурируемой схемы с архитектурой островного типа



Рисунок 2.10 - Диаграмма связанности конфигурируемой схемы с архитектурой типа «Жук»

	1	
		используемых ресурсов
	Проект с заполнением	Проект с заполнением
	92,2 % ЛЭ	60,2 % ЛЭ
Изменение площади схемы	-	+5,45%
Изменение количества RC3	+1,13%	+0,15%
Изменение количества RC6	-2,55%	-2,87%
Изменение количества RC9	-5,6%	-5,01%
Изменение количества		
коммутационных ресурсов	+0,98%	+0,39%

Таблица 2.4 – Сравнение показателей площади и количества

Предлагаемая архитектура "Жук" включает три вида связей,

расположенных в кластере логических элементов:

(RC+новые линии)

Направленные связи (Direct Link), которые обеспечивают передачу и выбранным прием данных между логическим элементом И расположенными в логическими элементами, непосредственной близости (справа, слева, сверху, снизу и по образованным данными направлениями диагоналям), как представлено на рисунке 2.11. Важно периферийных отметить, что направленные связи логических элементов не распространяются за пределы кластера;

- Двунаправленные локальные вертикальные и горизонтальные шины, которые проходят через весь кластер, как представлено на рисунке 2.12 (между пересекающимися вертикальными и горизонтальными связями стоят специальные коммутаторы, позволяющие указанным линиям связи обмениваться данными);
- Двунаправленные "Лапы жука", которые названы так из-за своего вида и изображены на рисунке 2.13.



Рисунок 2.11 – Направленные связи (узлами обозначены логические элементы, красными линиями обозначены существующие направленные связи)

		1.1				

Рисунок 2.12 – Локальные вертикальные и горизонтальные шины (узлами обозначены логические элементы, зелеными линиями обозначены существующие связи)



Рисунок 2.13 – Связи "Лапы жука" (узлами обозначены логические элементы, зелеными линиями обозначены существующие связи)

При рассмотрении одной "лапы жука", изображенной на рисунке 2.14, видно, что она обеспечивает связь между 24 логическими элементами. Данное значение выбрано в качестве первого приближения на основании разбиения кластера на области 4х4 логических элементов, что соответствует размеру LAB, используемого в существующих проектах.



Рисунок 2.14 – Структура "лапы жука" (узлами обозначены логические элементы, зеленой линей обозначена существующая связь, фиолетовыми линиями обозначены границы областей 4х4)

Таким образом, одна связь обеспечивает связь между тремя областями 4х4, связывая половину каждой области. Исходя из геометрии связи, наиболее подходящим размером кластера является массив 12х12 логических элементов.

Ширина каждой из связей для первой версии прототипа была определена количеством подключенных к ней логических элементов. В соответствии с этим, ширина горизонтальных и вертикальных связей равна 24 битам, ширина направленных связей равна 1 биту, ширина связей "лапы жука" равна 24 битам.

Помимо связей "лапы жука", передача данных между кластерами происходит посредством подключения соответствующих горизонтальных и вертикальных линий друг к другу через специальные блоки переключения (Рисунок 2.15).



Рисунок 2.15 – Соединение локальных связей 4-х кластеров между собой

Подключение кластеров к блокам ввода/вывода производится через горизонтальные и вертикальные линии, а так же через связи "лапы жука", которые служат исключительно для вывода информации с микросхемы.

2.5. Оценка эффективности базовых кристаллов конфигурируемых схем

Эффективность использования площади кристалла будем оценивать сравнением площадей, необходимых для размещения типовой схемы заказчика определенного класса в базисе конфигурируемой схемы и в базисе заказного кристалла, выполненного на основе стандартных ячеек. Проведем оценку эффективности базового кристалла ПЛИС. Для конфигурируемых схем других типов расчет проводится аналогично. Будем считать, что схема проекта заказчика полностью заполняет базовый кристалл ПЛИС.

Определим площади основных составных частей схемы.

 $S_{c\phi}$ – площадь встроенных СФ блоков ($S_{c\phi_i}$ – площадь одного блока – типа). Будем считать, что площади конкретных СФ –блоков в базисах ПЛИС и заказного кристалла совпадают. Блоки ввода-вывода также отнесем к СФ блокам. Отметим, что, как правило, общее количество СФ блоков в ПЛИС (для ПЛИС: $K_i^{c\phi_n}$ – количество СФ блоков одного типа и $N_{cn\phi}$ – количество их типов) будет значительно больше требуемого для реализации схемы заказчика количества СФ блоков, которое используется в заказной схеме (для заказной схемы: $K_i^{c\phi_3}$ – количество СФ блоков одного типа и $N_{c\phi_3}$ – количество их типов).

 $S_{\phi^{q}}$ – площадь функциональной части заказного кристалла схемы заказчика без СФ блоков, $S_{\phi^{q}i}$ – площадь ее составной части ($i = 1, \dots, N_{\phi^{q}}$), $K_{i}^{\phi^{q}}$ – количество составных частей –типа.

 S_{n_3} – площадь логических элементов ПЛИС, на которых может быть реализована выбранная схема заказчика, $S_{n_{3i}}$ – площадь составной части схемы заказчика. Определим, что, как и для заказного кристалла, $i = 1, \dots, N_{\phi_q}$, то есть разбиение на блоки в ПЛИС соответствует разбиению на блоки в заказной схеме.

 $S_{\rm KM}$ и соответственно $S_{{\rm KM}_i}$ для i – го блока – площадь коммутации и конфигурационной памяти ПЛИС, сопряженной с используемыми логическими элементами и составными частями, $i = 1, \dots, N_{\rm dy}$.

Будем считать, что в заказном кристалле нет выделенных коммутационных каналов, которые занимают отдельную площадь, а все линии связи располагаются над плотноупакованными активными слоями.

Тогда эффективность использования площади кристалла ПЛИС можно определить как отношение площадей заказной схемы к площади самой ПЛИС:

$$E_{1} = \frac{\sum_{i=1}^{N_{c\phi\pi}} K_{i}^{c\phi\pi} \cdot S_{c\phi_{i}} + \sum_{i=1}^{N_{\phi\pi}} K_{i}^{\phi\pi} \cdot S_{\phi\pi_{i}}}{\sum_{i=1}^{N_{c\phi\pi}} K_{i}^{c\phi\pi} \cdot S_{c\phi_{i}} + \sum_{i=1}^{N_{\phi\pi}} \alpha_{i} \cdot K_{i}^{\phi\pi} \cdot S_{\pi\pi_{i}}} = K_{1} \left[\frac{S_{c\phi}}{S_{\Sigma_{\pi}c\phi}} + K_{2} \cdot \frac{S_{\phi\pi}}{(S_{\pi\pi} + S_{KM})} \right], (1)$$

где S₃ – площадь заказного кристалла схемы заказчика, S_п – площадь ПЛИС, на которой располагается та же схема заказчика.

Из формулы (1) следует, что для повышения эффективности ПЛИС необходимо размещать в ПЛИС наиболее востребованные для заданного класса задач СФ блоки. То есть встроенные СФ блоки должны быть максимально задействованы в каждом случае применения ПЛИС, что подчеркивает важность специализации гетерогенных программируемых схем. В то же время для повышения эффективности необходимо минимизировать площади логических элементов, цепей коммутации и конфигурационной памяти при сохранении функциональных возможностей ПЛИС. Таким образом, эффективность использования площади кристалла ПЛИС зависит не только от площади логических блоков (ЛБ) и коммутаций, но и от количества и типов встроенных СФ блоков.

2.6. Выводы

Предложен маршрут проектирования конфигурируемых интегральных схем, в рамках которого на основе разработанных методов предварительного анализа схем заказчиков и программного прототипирования создается

архитектура и электрическая схема базового кристалла программируемой схемы.

Предложены методы предварительного анализа схем заказчиков с целью определения значений важнейших характеристик архитектуры разрабатываемых гетерогенных конфигурируемых схем с повышенной эффективностью использования трассировочных и иных ресурсов.

Предложен новый метод, используемый в маршруте разработки конфигурируемых гетерогенных интегральных схем - программное прототипирование с оценкой эффективности реализаций проектируемых схем заказчиков до фактического изготовления базового кристалла за счет оперативной настройки САПР на изменения в конструкции, схемотехнике и топологии базового кристалла в соответствии с предложенной в работе стратегией.

Предложены архитектурные решения конфигурируемых схем с улучшенной эффективностью трассировочных ресурсов.

Определены критерии эффективности использования площади базовых кристаллов конфигурируемых интегральных схем.

Глава 3. Математические модели для разработки

конфигурируемых гетерогенных интегральных схем целевого назначения и программирования схем заказчиков на их основе

В основе разрабатываемых математических (теоретикомножественных) моделей лежит формализация и обобщение понятия так называемого иерархического описания списка соединений (hierarchical netlist [79]).

Известный в области разработки микросхем иерархический подход к описанию списка соединений обобщен, расширен и формализован с применением аппарата теории множеств, математической логики и теории графов. Построенная математическая модель проекта объединяет как различные стили проектирования (полностью заказного, полузаказного – базовых матричных кристаллов, программируемых схем), так и подходы к решению различных задач разработки (от разработки базового гетерогенного кристалла с элементами программирования до создания конкретной зашивки схемы заказчика).

В отличие от существующих графовых моделей схем в работе учтены особенности различных исполнений частей гетерогенных конфигурируемых интегральных схем. Эти особенности обобщены в предложенной в работе иерархической модели, которая является общей как для различных стилей проектирования частей базового кристалла гетерогенной конфигурируемой схемы, так и для схемы заказчика, что позволяет формализовать задачу отображения схемы заказчика в базовом кристалле.

Одним из результатов построенной математической модели является формализация соответствий между элементами базового проекта конфигурируемой гетерогенной интегральной схемы (базового кристалла) и проектируемыми схемами от заказчика, функциональность которых требуется запрограммировать в кристалле от производителя. Это, в частности обеспечивает оперативную настройку структуры базового кристалла на

функциональные требования заказчиков. Кроме того, использование модели позволяет формализовать настройку САПР на соответствующие изменения в конструкции, схемотехнике и топологии базового кристалла.

Введен новый этап в маршруте проектирования конфигурируемых интегральных схем - так называемое "программное гетерогенных прототипирование" с оценкой эффективности реализаций различных проектируемых схем заказчиков до фактического изготовления самого базового кристалла за счет оперативной настройки САПР на соответствующие изменения в конструкции, схемотехнике и топологии базового кристалла. В отличии от классического прототипирования с применением готовых кристаллов ПЛИС, в данном случае речь идет о новом этапе разработки самого базового кристалла с элементами ПЛИС. Предложенная математическая модель подробно описывает трассировочные ресурсы базового кристалла, что позволяет учесть широкий спектр способов программирования коммутационных элементов, а также разнообразие выполняемых ими логических функций. Приведенное в работе формальное описание трассировочных ресурсов базового кристалла используется в рамках этапа программного прототипирования при анализе пространства архитектурных решений гетерогенных конфигурируемых интегральных схем.

3.1 Терминология. Основные понятия и обозначения

Иерархическое описание проекта можно определить, как упорядоченную четверку:

 $\Pi = (S, L, s_m, m_m)$ – иерархическое описание проекта;

 $S = \{s_i, i = 1, ..., |S|\}$ – множество схем в иерархическом описании проекта;

 $L \subset S$ – базис или подмножество базисных библиотечных подсхем для текущего уровня (или этапа) проектирования;

 $s_m \in S$, $s_m \notin L$ – главная схема или схема верхнего уровня.

m_m – модель базового кристалла верхнего уровня, в которой описаны особенности различных типов базовых кристаллов, раскрывающие различие их конструкций.

Для БМК $\Pi_b = (S, L, s_m, m_{mb}), S = L \cup s_m$.

Для ПЛИС $\Pi_{g} = (S, L, s_{m}, m_{mg}) - при наличии ограничений на верхнем$ $уровне, <math>m_{mg}$ может быть пустым;

Для заказной схемы $\Pi_a = (S, L, O, m_{ma}), S = L;$

Для схемы заказчика $\Pi_u = (S, L, s_m, \emptyset).$

Каждое из схемных описаний в иерархии проекта можно определить следующим образом:

 $\forall s \in S: s = (\mu(s), E(s), N(s), P(s), C(s)),$ где:

 $\mu(s)$ – уникальное имя схемы (строка символов);

 $E(s) = \{e_i, i = 1, ..., |E(s)|\}$ – множество элементов в схеме;

 $N(s) = \{n_i, i = 1, ..., |N(s)|\}$ – множество цепей (узлов) в схеме;

 $P(s) = \{p_i, i = 1, ..., |P(s)|\}$ – множество внешних выводов (контактов) схемы;

 $C(s) = \{c_i, i = 1, ..., |C(s)|\}$ – множество соединений (коммутаций) семы.

схемы.

Множество элементов характеризуется следующими данными:

 $\forall e \in E(s): e = (\mu(e), m(e), P(e)),$ где:

 $\mu(e)$ – уникальное имя элемента (строка символов);

 $m(e) \in S$ – модель элемента, представленная в иерархическом описании схемой следующего более низкого уровня иерархии;

 $P(e) = \{p_i, i = 1, ..., |P(e)|\}$ — множество выводов элемента, совпадающее по составу с множеством внешних выводов модели (связанных с ним взаимно-однозначным соответствием):

$$P(e) \leftrightarrow P(m(e)); |P(e)| = |P(m(e))|.$$

Множество внешних выводов схемы характеризуется следующими данными:

 $\forall p \in P(s): p = (\mu(p), \tau(p)),$ где:

 $\mu(p)$ – уникальное имя вывода;

 $\tau(p) \in \{\tau_{inp}, \tau_{out}, \tau_{bi}\}$ – тип вывода: вход, выход или двунаправленный.

Множество цепей (узлов) схемы характеризуется именем и соответствующим цепи набором соединений:

 $\forall n \in N(s): \mu(n) -$ имя цепи (узла), (строка символов);

C(s) – Множество соединений в схеме, определяемое как

подмножество пар:

$$C(s) = \left\{ (p,n): p \in \left(\bigcup_{i=1,\dots,|E(s)|} P(e_i) \cup P(s) \right), \quad n \in N(s) \right\},$$

таких, что, для любого контакта цепь единственная или не существует вовсе:

$$\forall p \in \left\{ \bigcup_{i=1,\dots,|E(s)|} P(e_i) \cup P(s) \right\}:$$
$$(\exists ! n \in N(s): (p,n) \in C(s)) \lor (\forall n \in N(s): (p,n) \notin C(s)) .$$

Другими словами, множество соединений в схеме определяется как однозначное отображение:

$$C^*(s) = \left\{ \left(\bigcup_{i=1,\dots,|E(s)|} P(e_i) \cup P(s) \right) \to (N(s) \cup \emptyset) \right\}.$$

При этом обратное отображение определяет собственно список соединений конкретной цепи и не может быть однозначным - количество соединений у каждой цепи должно быть не менее двух, в противном случае цепь будет считаться ошибочной или ложной:

$$C^{*-1}(s) = \left\{ N(s) \to \left(\bigcup_{i=1,\dots,|E(s)|} P(e_i) \cup P(s) \right) \right\},\$$
$$\forall n \in N(s) \colon |\{(p,n) \colon (p,n) \in C(s)\}| \ge 2.$$

Как правило, внешний вывод может быть у цепи только один:

$$\forall n \in N(s) : |\{(p,n): (p,n) \in C(s) \land p \in P(s)\}| \le 1.$$

В частности, для Spice – описаний [48] при этом имя внешнего контакта совпадает с именем цепи:

$$\forall n \in N(s), \forall p \in P(s): (p, n) \in C(s) \implies \mu(n) = \mu(p).$$

Для текущего этапа проектирования подсхемы базисного библиотечного уровня не содержат внутренних данных и представляют собой "черные ящики":

$$\forall s \in L: E(s) = \emptyset, N(s) = \emptyset, C(s) = \emptyset.$$

В этом случае моделирование подсхем нижнего уровня выполняется на основе встроенных моделей, и само описание "черных ящиков" может быть скрыто от разработчика. Например, на схемотехническом уровне проектирования к базисному библиотечному уровню относятся транзисторы, емкости, сопротивления, индуктивности и др.

Принципиальное отличие **гетерогенных** схем заключается в существенном расширения трактовки понятия "базисных подсхем". В случае **гетерогенных** схем в базис на определенных этапах проектирования наряду со стандартными элементами могут входит готовые сложно-функциональные блоки.

3.2 Теоретико-графовые модели и метрики. Иерархия проекта

Иерархическое описание проекта $\Pi = (S, L, s_m, m_m)$ определяет ациклический ориентированный граф $G(\Pi) = (V, A)$ вложенности схемных описаний по следующим правилам:

 $\forall s \in S: \exists ! v(s): v(s) \in V;$

$$\forall s \in S, \forall e \in E(s): (v(s), v(m(e))) \in A.$$

Таким образом, дуги в графе вложенности схемных описаний однозначно связаны с элементами подсхемы на определенном уровне иерархии:

$$\forall a \in A: \exists s \in S, \exists e \in E(s): a = (v(s), v(m(e))).$$

Рассматривая путь в графе вложенности схемных описаний от вершины $v(s_m)$ до заданной вершины $v(s) \in V$ можно идентифицировать уникальный экземпляр каждого вхождения подсхемы. Для идентификации конкретного экземпляра подсхемы могут использоваться следующие варианты:

V1: вектор или упорядоченное множество дуг в графе вложенности схемных описаний от корневой вершины до заданной $a = \{a_1, a_2, ..., a_n\};$

V2: вектор или упорядоченное множество соответствующих этим дугам элементов $e = \{e_1, e_2, ..., e_n\};$

V3: вектор или упорядоченное множество соответствующих этим элементам имен $\mu = \{\mu(e_1), \mu(e_2), ..., \mu(e_n)\};$

V4: уникальное иерархическое имя, полученное путем конкатенации из имен соответствующих элементов с уникальным разделителем.

Обозначим через $\psi(v(s))$ степень глубины иерархического описания части проекта с вершиной v(s), определяемую, как максимальную длину пути в графе вложенности $G(\Pi) = (V, A)$ от нижнего уровня до v(s):

$$\psi(v(s)) = \begin{cases} 0, \text{для } s \in L, \\ \max_{e \in E(s)} \psi(v(m(e))) + 1. \end{cases}$$

Значение $\psi(v(s_m))$ для главной схемы верхнего уровня характеризует степень вложенности иерархического описания проекта в целом. При $\psi(v(s_m)) = 1$ описание проекта носит название "плоского" списка соединений (flat netlist). В противоположность "плоскому" представлению иерархическим представлением обычно называют проект при $\psi(v(s_m)) > 1$. В нашем случае "плоское" представление – частный случай обобщенного иерархического.

Иерархическое описание может быть преобразовано в соответствующее ему "плоское" представление путем рекурсивной flatпроцедуры. Для заданного проекта $\Pi = (S, L, s_m, m_m)$ в "плоском" представлении целесообразно сохранить только те подсхемы, которые

фактически применялись в s_m . Обозначим через $\varphi(s, s_t)$ логическую функцию, определенную на Декартовом произведении $S \times S$, принимающую значение 1 тогда и только тогда, когда *s* фактически используется в s_t :

$$\varphi(s, s_t) = \left((s = s_t) \lor \left(\bigvee_{e \in E(s_t)} \varphi(s, m(e)) \right) \right).$$

Т.е., $\varphi(s, s_t) = 1$, если ($s = s_t$) или $\exists e \in E(s_t)$: $\varphi(s, m(e)) = 1$.

Пусть $\Pi = (S, L, s_m, m_m)$ - иерархическое описание проекта, $G(\Pi) = (V, A)$ - соответствующий ему ациклический ориентированный граф вложенности схемных описаний, $s_t \in S$ - одна из подсхем в проекте, и $v(s_t) \in V$ - соответствующая вершина в графе $G(\Pi) = (V, A)$.

Обозначим через $G^*(s_t) = (V^*(s_t), A^*(s_t))$ подграф графа G(P) = (V, A), индуцированный подмножеством вершин, соответствующих подмножеству подсхем, фактически используемых в s_t :

$$V^{*}(s_{t}) = \{v(s_{i}): \varphi(s_{i}, s_{t}) = 1\};$$

$$A^{*}(s_{t}) = \{(v(s_{i}), v(s_{j})): (v(s_{i}), v(s_{j})) \in A \& \varphi(s_{i}, s_{t}) \& \varphi(s_{j}, s_{t})\}$$

В терминах графа вложенности схемных описаний преобразование иерархического описания $G(\Pi) = (V, A)$ в "плоское" $G_f(\Pi) = (V_f, A_f)$ сводится к построению дерева глубины $\psi(v(s_m)) = 1$ с единственной не листовой вершиной (корнем дерева) $v(s_m)$ и множеством листовых вершин V_l , определяемых рекурсивно по следующим формулам:

$$\begin{split} V_f &= \{v(s_m) \cup V_l(s_m)\};\\ V_l(s_t) &= \begin{cases} v(s_t) \text{ при } s_t \in L, \psi\big(v(s_l)\big) = 0\\ \{\bigcup_i(V_l(s_i)): (v(s_t), v(s_i)) \in A\} \text{ при } s_t \notin L, \psi\big(v(s_l)\big) > 0;\\ A_f &= \{(v(s_m), v(s_i)): v(s_i) \in V_l(s_m)\}. \end{split}$$

Для заданного проекта $\Pi = (S, L, s_m, m_m)$ "плоское" представление $\Pi_f(\Pi) = (S_f, L_f, s_f)$ можно построить по следующим правилам:

 $L_f = \{s \colon (s \in L) \land \varphi(s, s_m)\};$

$$S_{f} = \{s_{f} \cup L_{f}\};$$

$$s_{f} = (\mu(s_{f}), E(s_{f}), N(s_{f}), P(s_{f}), C(s_{f})), \text{где:}$$

$$\mu(s_{f}) = \mu(s_{m});$$

$$P(s_{f}) \leftrightarrow P(s_{m});$$

Множество элементов в "плоском" представлении однозначно соответствует множеству листовых вершин в "плоском" варианте графа вложенности схемных описаний: $E(s_f) \Leftrightarrow V_l(s_m)$. Соответствующие листовым вершинам подсхемы и их выводы определяют формирование элементов "плоского" описания $e \in E(s_f), e = (\mu(e), m(e), P(e))$.

Имена элементов $\mu(e), e \in E(s_f)$ и имена цепей $\mu(n), n \in N(s_f)$ в "плоском" представлении должны быть уникальными и содержать информацию об именах элементов более высоких уровней иерархии, в состав которых входили подсхемы, содержащие рассматриваемый элемент, до раскрытия иерархии. Это достигается путем использования одного из векторных описаний, перечисленных выше (V1-V2-V3) или за счет конкатенации имен элементов с уникальным разделителем (V4).

Формирование списка элементов, списка цепей и коммутаций для конкретного экземпляра схемы в исходном иерархическом описании зависит от контекста, включающего две компоненты:

- идентификация экземпляра схемы в одном из вариантов, описанных выше (V1-V2-V3-V4);

- и назначение коммутаций внешних выводов схемы.

В общем случае в иерархическом описании проекта каждое определение схемы или подсхемы может использоваться много раз. В этом случае описание называют "свернутым" (folded). При этом граф вложенности схемных описаний $G(\Pi) = (V, A)$ представляет собой ациклический ориентированный граф, не обязательно дерево, и требует гораздо меньше ресурсов памяти по сравнению с "плоским" представлением.

В ситуации, когда каждое описание используется ровно один раз, граф вложенности схемных описаний является деревом с корневой вершиной $v(s_m)$ и листовыми вершинами $v(s), s \in L$. В этом случае описание называют "развернутым" (unfolded). В частности, необходимость в "развернутом" описании возникает на этапе топологического синтеза, когда каждый экземпляр подсхемы получает уникальные топологические координаты в глобальной системе координат. "Развернутое" представление по сложности сопоставимо с "плоским" представлением, однако может быть более экономным и эффективным по сравнению с "плоским" вариантом по организации хранения имен элементов и цепей.

Среди прочих иерархических описаний особо следует выделить двухуровневый вариант $\psi(v(s_m)) = 2$, наиболее точно отражающий архитектуру базовых кристаллов конфигурируемых гетерогенных схем "островного" типа [80]. При этом для проекта $\Pi = (S, L, s_m, m_m)$ наряду с обычным множеством элементов библиотечного уровня L выделяется промежуточный уровень "блоков", не входящих в $L: B = \{b_i\}, : B \cap L = \emptyset$. В этом случае процедура преобразования в "плоское" представление выполняется один раз на верхнем уровне, где в качестве библиотечного уровня из B преобразование в "плоское" представление с библиотечным уровнем L. Итоговое двухуровневое блочное представление проекта $\Pi_b(\Pi) = (S_b, L_b, s_b)$ содержит следующие компоненты:

$$L_{b} = \{s: (s \in L) \land \varphi(s, s_{m})\};$$

$$S_{b} = \{s_{b} \cup L_{b} \cup B\}, L_{b} \cap B = \emptyset;$$

$$s_{b} = (\mu(s_{b}), E(s_{b}), N(s_{b}), P(s_{b}), C(s_{b})), \text{где:}$$

$$\mu(s_{b}) = \mu(s_{m});$$

$$P(s_{b}) = P(s_{m}).$$

Двухуровневое блочное представление может быть как "свернутым", так и "развернутым". Вариант "свернутого" блочного представления в проектировании топологии предполагает одну и ту же топологическую реализацию для всех экземпляров блока.

3.3 Теоретико-графовые модели и метрики. Технологическое отображение

В данном разделе предлагается классификация компонент схемы для решения задачи настройки САПР на изменения в конструкции и схемотехнике базового кристалла в терминах технологического отображения схемы заказчика в базовом кристалле.

Традиционное понимание технологического отображения (technology mapping) подразумевает один из подэтапов логического синтеза с приведением схемы к базису библиотечных элементов с известными технологическими и топологическими реализациями.

В случае конфигурируемых гетерогенных интегральных схем технологическое отображение следует рассматривать как установление соответствия между множеством библиотечных элементов проекта заказчика с библиотечными элементами проекта самой конфигурируемой гетерогенной схемы.

Сменяемость технологических и конструкторских решений в связи с необходимостью подстройки конфигурируемых гетерогенных схем целевого назначения под конкретные приложения заказчика приводит к необходимости оперативной настройки САПР на соответствующие изменения в конструкции, схемотехнике и топологии. Для решения данной проблемы предлагается дальнейшее развитие обобщенной математической модели с классификацией различных вариантов элементов в базовом кристалле и в схеме заказчика.

Пусть $\Pi = (S, L, s_m, m_m)$ - *базовый проект* конфигурируемой или программируемой гетерогенной схемы (базовый кристалл), $\Pi_f(\Pi) = (S_f, L_f, s_f, m_f)$ - "плоский" вариант базового проекта, а $\Pi_u = (S_u, L_u, s_{mu}, m_{mu})$ - проектируемая схема от конечного заказчика,

функциональность которой требуется запрограммировать в кристалле от производителя. Как правило, список соединений схемы заказчика - это результат автоматического синтеза с RTL [79] описания в "плоском" варианте.

Для установления требуемого соответствия следует выделить разные типы и группы библиотечных элементов как в базовом проекте, так и в проектируемой схеме заказчика.

В библиотеке базового проекта следует выделить логические элементы L_{LE} , периферийные элементы ввода-вывода L_{IO} , сложно-функциональные макроблоки L_M , трассировочные элементы L_{RO} и иные вспомогательные элементы L_{BB} ("черные ящики"), не содержащие в своем составе ни один из перечисленных типов элементов L_{LE} , L_{IO} , L_M , L_{RO} , выполняющие вспомогательные функции, (например, для программирования памяти), не связанные с непосредственным отображением элементов проектируемой схемы заказчика:

$$L = L_{LE} \cup L_{IO} \cup L_M \cup L_{RO} \cup L_{BB}.$$

При раскрытии иерархии и преобразовании базового проекта в "плоское" или в двухуровневое блочное представление необходимо учитывать фактическое количество элементов каждого из перечисленных типов в каждой из подсхем на более высоких уровнях иерархии. Обозначим фактическое количество элементов перечисленных типов в некоторой заданной подсхеме *s* через $\sigma_{LE}(s)$, $\sigma_{IO}(s)$, $\sigma_M(s)$, $\sigma_{Ro}(s)$ соответственно. Тогда принадлежность подсхемы к множеству "черных ящиков" можно определить по факту отсутствия в ней элементов из перечисленных типов, независимо от того, имеет ли такая схема детализацию в терминах подсхем и элементов более низкого уровня:

 $s \in L_{BB} \equiv (\sigma_{LE}(s) + \sigma_{IO}(s) + \sigma_M(s) + \sigma_{RO}(s) = 0).$

При этом значение каждой из перечисленных функций подсчета элементов соответствующего типа $T \in \{LE, IO, M, Ro\}$ можно определить рекурсивно:

 $\sigma_T: S \to \mathcal{N}_0, \mathcal{N}_0 = \mathcal{N} \cup 0;$

$$\sigma_T(s) = \begin{cases} 1 \text{ при } s \in L_T, \\ 0 \text{ при } s \in L \setminus L_T, \\ \sum_{e \in E(s)} \sigma_T(m(e)) \text{ при } s \notin L. \end{cases}$$

По результатам таких подсчетов раскрытие иерархии при преобразовании базового проекта в "плоское" или в двухуровневое блочное представление можно ограничить (остановить) на уровне $L = L_{LE} \cup L_{IO} \cup L_M \cup L_{RO} \cup L_{BB}$.

Элементы $e: m(e) \in L_{LE} \cup L_{IO} \cup L_M$ используются для отображения библиотечных элементов проекта заказчика. Элементы $e: m(e) \in L_{RO}$ применяются для отображения цепей и коммутаций проекта заказчика, и на их основе в автоматическом режиме можно построить граф для решения задач трассировки.

Схемы библиотечного уровня $s \in L_{LE} \cup L_{IO} \cup L_M = L \setminus \{L_{BB} \cup L_{RO}\}$ могут быть запрограммированы под различные варианты функциональных решений с использованием программируемой памяти. Множество внешних выводов таких схем

 $P(s) = \{p_i, i = 1, \dots, |P(s)|\}, s \in L_{LE} \cup L_{IO} \cup L_M.$

можно разделить на 3 независимых подмножества по функциональному назначению:

$$P(s) = P_r(s) \cup P_m(s) \cup P_s(s)$$
, где

 $P_r(s)$ - подмножество сигнальных или трассировочных выводов для подключения внешних сигнальных цепей с применением трассировочных ресурсов из L_{Ro} ,

P_m(*s*) - подмножество программируемых выводов для управления различными вариантами функциональных решений;

 $P_s(s)$ - подмножество специальных выводов для подключения специальных сигналов (например, для выбора цепей питания, земли, синхронизации, сброса и др.), подключение которых требует специальной обработки, отличной от подключения обычных цепей или сигналов.
Мощность множества $|P_r(s)|$ определяет максимально допустимое количество выводов элемента в библиотеке L_u проекта $\Pi_u = (S_u, L_u, s_{mu})$. Некоторые из выводов $P_r(s)$ могут не использоваться в конкретном библиотечном элементе из L_u или быть замкнутыми на землю / питание.

Мощность множества $|P_m(s)|$ определяет длину вектора программирования для реализации конкретных функций и режимов работы библиотечных элементов. За счет разных вариантов программирования одному экземпляру $s \in L_{LE} \cup L_{IO} \cup L_M$ может соответствовать множество различных реализаций в библиотеке L_u с общим количеством равным $2^{|P_m(s)|}$. В частности, количество вариантов классического LUT (LookUp Table) [50] с *n* входами равно $2^{|P_m(s)|} = 2^{2^n}$.

Таким образом, формирование элементов $s_u \in L_u$, $s_u = (\mu(s_u), \emptyset, \emptyset, P(s_u), \emptyset)$ библиотеки L_u проекта $\Pi_u = (S_u, L_u, s_{mu})$ можно реализовать путем установки следующих соответствий для выводов библиотечных схем базового кристалла $P(s) = \{p_i, i = 1, ..., |P(s)|\}, s \in L_{LE} \cup L_{IO} \cup L_M$:

 $P_m(s) \to \mathcal{B}^{|P_m(s)|}, \mathcal{B} = \{0,1\};$ $P_r(s) \to P(s_u) \cup \{P_0, P_1, P_z\},$ где

*P*₀, *P*₁, *P*_z - условные обозначения вводов, предполагающие внешние соединения соответственно с узлом земли, питания или висячим узлом.

Без ограничения общности можно предполагать, что проектируемая схема от заказчика $\Pi_u = (S_u, L_u, s_{mu})$ задана в "плоском" варианте, получена как результат автоматического синтеза с RTL [79] описания или результат раскрытия иерархического описания в "плоское" представление, тогда $S_u = L_u \cup \{s_{mu}\}$.

Пусть $s_{mu} = (\mu(s_{mu}), E(s_{mu}), N(s_{mu}), P(s_{mu}), C(s_{mu})).$

Что касается внешних выводов схемы заказчика $p_u \in P(s_{mu})$, то для них возможны два режима обработки:

- Один вариант - назначение периферийных элементов из L_{IO} . При этом в зависимости от типа вывода $\tau(p_u) \in \{\tau_{inp}, \tau_{out}, \tau_{bi}\}$ программируются различные варианты периферийных элементов – входных, выходных или двунаправленных.

Второй вариант предполагает, что периферийные элементы уже назначены на этапе формирования схемы заказчика, и выводы схемы p_u ∈ P(s_{mu}) представляются собой внешние интерфейсы для моделирования.

Сам этап назначения периферийных элементов предполагает не только выбор конкретного типа периферийного элемента $s_{IO} \in L_{IO}$ с программированием:

$$P_m(s_{IO}) \to \mathcal{B}^{|P_m(s_{IO})|}, \mathcal{B} = \{0,1\},\$$

но и назначение конкретного периферийного элемента $e \in E(s_f)$, и следовательно, конкретного места размещения этого элемента в "плоском" представлении базового кристалла, т.е. установления соответствия (отображения):

$$P(s_{mu}) \to \{e : e \in E(s_f), m(e) = s_{IO}, s_{IO} \in L_{IO}\}\}$$

При этом сама процедура назначения конкретной периферийной площадки с размещением может выполнятся как в ручном или интерактивном режиме, так и в автоматическом режиме.

Аналогичная задача решается и для всех внутренних элементов схемы заказчика, как для простых логических элементов, так и для сложнофункциональных макроблоков:

$$E(s_{mu}) \rightarrow \{e \colon e \in E(s_f), m(e) \in \{L_{LE} \cup L_M \cup L_{IO}\}\}.$$

Установление отображения, в котором каждому элементу схемы заказчика устанавливается в соответствие элемент базового проекта, по сути своей есть размещение элементов схемы заказчика.

3.4 Модели и метрики для решения задач оценки архитектуры базового кристалла на основе анализа схем заказчиков. Метрики для базового кристалла

В данном разделе предлагается набор теоретико-графовых моделей и метрик для поиска оптимальной архитектуры ПЛИС. Основная идея заключается в том, что в процессе разработки архитектуры ПЛИС источником информации о том, какие архитектурные решения применять, должен быть анализ в первую очередь схем заказчиков от конкретного потребителя. На этапе предварительного анализа маршрута разработки базового кристалла на основе схем заказчиков определяются требования к коммутационным и иным ресурсам конфигурируемой схемы. На этапе программного прототипирования определяется достаточность этих ресурсов для конфигурирования схем заказчиков в базовом кристалле, разработанной архитектуры.

Для задачи анализа схем заказчиков поле ПЛИС может быть представлено в упрощенном виде: плоская (flat) матрица из логических элементов (LE) с заранее определенной конфигурацией (например, 4входовые LUT (Look Up Table или элемент, реализующий таблицу истинности) с триггером).

Другой вариант - двухуровневое представление для ПЛИС «островного типа». В этом случае верхний уровень ПЛИС, как правило, может быть представлен в виде матрицы блоков (CLB [81] в терминологии компании Xilinx или LAB [82] в терминологии компании Altera (ныне подразделения Intel)). При этом блок может быть представлен матрицей из логических элементов.

Ключевыми характеристиками конструкции базового кристалла, влияющими на полноту и эффективность реализации схем заказчиков в базовой матрице ПЛИС являются:

количество блоков (количество блочных элементов по вертикали и по горизонтали);

- размеры блока (размер блочной матрицы по вертикали и по горизонтали);

- количество внешних выводов блока; в том числе количество входящих трасс (входов блока), количество исходящих трасс (выходов блока), количество сквозных трасс (двунаправленных выходов блока, соединенных внутри блока);

- количество локальных трасс внутри блока;

- количество коммутаций логических элементов (LE); в том числе количество коммутаций для входов LE, количество коммутаций для выходов LE;

- количество связей между блоками и их структура (степень ветвления, пропускная способность вертикальных каналов между блоками, пропускная способность горизонтальных каналов между блоками; пропускная способность сечения по вертикали для заданного *X*, пропускная способность сечения по горизонтали для заданного *Y*; пропускная способность переключательной коробки (switch box) между каналами;

размеры межсоединений и их влияние на быстродействие схемы;

и другие.

Пусть задано представление базового кристалла $\Pi = (S, L, s_m, m_m)$ - в иерархическом виде; $\Pi_f(\Pi) = (S_f, L_f, s_f, m_f)$ - его "плоское" представление; $\Pi_b(\Pi) = (S_b, L_b, s_b, m_b)$ - двухуровневое блочное представление проекта.

Как уже было упомянуто ранее, в двухуровневом представлении проекта $\Pi_b(\Pi) = (S_b, L_b, s_b, m_b)$ выделяется, наряду с множеством элементов библиотечного уровня *L*, промежуточный уровень «блоков», не входящих в *L*: $B = \{b_i\}, : B \cap L = \emptyset$. Чаще всего логические элементы группируются в одинаковые блоки на схемотехническом уровне, тогда количество различных типов блоков $|B| = |\{b\}| = 1$.

Итоговое двухуровневое блочное представление проекта $\Pi_b(\Pi) = (S_b, L_b, s_b, m_b)$ содержит следующие компоненты:

$$L_b = \{s: (s \in L) \land \varphi(s, s_m)\};$$

$$S_b = \{s_b \cup L_b \cup B\}, L_b \cap B = \emptyset;$$

$$s_b = (\mu(s_b), E(s_b), N(s_b), P(s_b), C(s_b)), \text{где:}$$

$$\mu(s_b) = \mu(s_m); P(s_b) = P(s_m).$$

Выделяя для блочного представления, как и ранее в случае «плоского» варианта, различные типы подсхем библиотечного уровня, а именно логические элементы L_{LE} , периферийные элементы ввода-вывода L_{IO} , сложно-функциональные макроблоки L_M , трассировочные элементы L_{RO} и иные вспомогательные элементы L_{BB} («черные ящики»), не содержащие в своем составе ни один из перечисленных типов элементов L_{LE} , L_{IO} , L_M , L_{RO} , получим:

$$S_b = \{s_b \cup L_{LE} \cup L_{IO} \cup L_M \cup L_{RO} \cup L_{BB} \cup B\}.$$

Тогда подмножество блочных элементов есть $E_B(s_b) \subset E(s_b), E_B(s_b) =$ { $e: e \in E(s_b) \& m(e) \in B$ } и может быть представлено в виде матрицы блоков (LAB):

$$M_b = \{m_{ij} \colon m_{ij} \in E_B(s_b), \ m(m_{ij}) \in B, i = 1, \dots, I_b, j = 1, \dots, J_b \}.$$

Общее количество блоков в базовом кристалле определяется размером матрицы блоков:

$$|E_B(s_b)| = |M_b| = I_b \cdot J_b$$

Аналогично сам блок в двухуровневом представлении состоит из элементов более низкого уровня иерархии:

$$b = (\mu(b), E(b), N(b), P(b), C(b)),$$
 где:
 $E(b) = \{e: m(e), m(e) \in L_{LE} \cup L_{Ro} \cup L_{BB}\}.$

Тогда подмножество логических элементов блока есть $E_{LE}(b) \subset E(b)$, $E_{LE}(b) = \{e: e \in E(b) \& m(e) \in L_{LE}\}$ и может быть представлено в виде матрицы логических элементов в составе блока (LAB):

$$M_{LE} = \{m_{ij} \colon m_{ij} \in E_{LE}(b), m(m_{ij}) \in L_{LE}, i = 1, \dots, I_{LE}, j = 1, \dots, J_{LE} \}.$$

Общее количество логических элементов в блоке определяется размером матрицы *M_{LE}*:

$$|E_{LE}(b)| = |M_{LE}| = I_{LE} \cdot J_{LE}.$$

Очевидно, что множество логических элементов «плоского» представления $\Pi_f(\Pi) = (S_f, L_f, s_f)$ базового кристалла $E_{LE}(s_f) \subset E(s_f)$, $E_{LE}(s_f) = \{e : e \in E(s_f) \& m(e) \in L_{LE}\}$ можно в этом случае также представить в виде полной матрицы логических элементов в составе базового кристалла, и такая полная матрица может быть получена в результате суперпозиции матрицы блоков M_b и матрицы логических элементов в составе блока M_{LE} :

$$M_f = \{m_{ij} \colon m_{ij} \in E_{LE}(s_f), m(m_{ij}) \in L_{LE}, i = 1, \dots, I_f, j = 1, \dots, J_f \}.$$

Предполагается, что все логические элементы в двухуровневом блочном представлении локализованы в блоках:

$$\forall e \in E(s_b) \cup E(b): m(e) \in L_{LE} \to e \in E(b).$$

Другими словами, отсутствуют логические элементы на верхнем уровне иерархического двухуровневого блочного представления:

$$\nexists e: e \in E(s_b) \& m(e) \in L_{LE}.$$

Тогда общее количество логических элементов в составе базового кристалла определяется размерами матриц M_b , M_{LE} :

$$I_{f} = I_{b} \cdot I_{LE},$$

$$J_{f} = J_{b} \cdot J_{LE},$$

$$I_{E} = I_{b} \cdot J_{E},$$

$$I_{E} = I_{b} \cdot J_{$$

$$|E_{LE}(s_f)| = |M_f| = I_f \cdot J_f = |E_B(s_b)| \cdot |E_{LE}(b)| = I_b \cdot J_b \cdot I_{LE} \cdot J_{LE}.$$

Для анализа коммутационных или трассировочных возможностей блоков и логических элементов следует различать разные классы и типы их выводов. Ранее были определены классы выводов логических элементов. Такую же классификацию можно распространить на блоки в двухуровневом представлении и на трассировочные элементы:

$$P(s) = \{p_i, i = 1, ..., |P(s)|\}, s \in L_{LE} \cup L_{IO} \cup L_M \cup L_B \cup L_{RO}, P(s) = P_r(s) \cup P_m(s) \cup P_s(s),$$

где $P_r(s)$ - подмножество сигнальных или трассировочных выводов для подключения внешних сигнальных цепей с применением трассировочных ресурсов из L_{Ro} ,

P_m(*s*) - подмножество программируемых выводов для управления различными вариантами функциональных решений;

 $P_s(s)$ - подмножество специальных выводов для подключения специальных сигналов (например, для выбора цепей питания, земли, синхронизации, сброса и др.), подключение которых требует специальной обработки, отличной от подключения обычных цепей или сигналов.

Кроме этого, следует различать выводы по направлению распространения сигналов:

 $\forall p \in P(s): \tau(p) \in \{\tau_{inp}, \tau_{out}, \tau_{bi}\}$ – тип вывода: вход, выход или двунаправленный.

Количество сигнальных или трассировочных выводов $P_r(b)$ блока b характеризует коммутационные возможности блока на межблочном уровне:

$$d(b) = |P_r(b)|.$$

По аналогии с терминологией теории графов можно определить $d(b) = |P_r(b)|$ как локальную степень блока, Количество выходов блока – как исходящую локальную степень блока b:

$$d^{-}(b) = |P_{r}^{-}(b)|,$$
$$P_{r}^{-}(b) = \{p_{i}^{-} \in P_{r}(b) : \tau(p_{i}^{-}) = \tau_{out}\},$$

а количество входов – как входящую локальную степень блока b:

$$d^{+}(b) = |P_{r}^{+}(b)|,$$
$$P_{r}^{+}(b) = \{p_{i}^{+} \in P_{r}(b) : \tau(p_{i}^{+}) = \tau_{in}\}$$

Что касается двунаправленных выходов, они могут использоваться поразному, в том числе (но не только) для сквозных трасс. По аналогии с однонаправленными выводами их количество можно обозначить через $d^*(b)$:

$$d^{*}(b) = |P_{r}^{*}(b)|,$$
$$P_{r}^{*}(b) = \{p_{i}^{*} \in P_{r}(b) : \tau(p_{i}^{*}) = \tau_{bi}\}.$$

Адаптация терминологии степеней вершин из теории графов к выводам блоков и их дальнейшее использование для оценок трассировочных возможностей применительно к схемам заказчиков наиболее оправдано при корректном подключении всех выводов блоков, когда все двунаправленные выводы используются исключительно для сквозных трасс, а каждый вход и выход имеют коммутации на межблочном уровне с другими элементами из множества:

$$E_b(s_b) = \{e : e \in E(s_b) \& m(e) \in (B \cup L_{IO} \cup L_M \cup L_{RO})\}.$$

В общем случает требуется более детальный анализ графа коммутаций на межблочном уровне с индивидуальным определением степеней вершин в составе графа коммутаций для каждого блочного элемента. Для заданного двухуровневого блочного представления проекта $\Pi_b(\Pi) = (S_b, L_b, s_b)$, где $s_b = (\mu(s_b), E(s_b), N(s_b), P(s_b), C(s_b))$, граф коммутаций межблочного уровня можно построить, как смешанный граф или ориентированный граф (с преобразованием каждого из двунаправленных ребер в пару дуг противоположного направления) $G_B(\Pi_b) = (V_B, A_B)$ по следующим правилам:

Множество вершин V_B графа $G_B(\Pi_b)$ взаимно однозначно соответствует подмножеству элементов блочного уровня, за исключением "черных ящиков" (здесь также предполагается, что все логические элементы локализованы в блоках):

$$\{E_b(s_b) = \{e : e \in E(s_b) \& m(e) \in (B \cup L_{IO} \cup L_M \cup L_{RO})\}$$
$$\iff \{v(e) : v(e) \in V_B\}.$$

При этом элемент $e: e \in E(s_b)$, соответствующий вершине графа $v(e): v(e) \in V_B$, можно наименовать как элемент, ассоциированный с вершиной v(e).

Множество дуг A_B формируется из информации о соединениях цепей (узлов) на межблочном уровне - для каждой пары элементов $x \in E(s_b), y \in E(s_b)$, контакты которых соединены общей цепью и выполнены правила распространения сигнала в соответствии с типами портов:

$$\{\exists n \in N(s_b) : (P(x), n) \in C(s_b) \& (P(y), n) \in C(s_b) \& (\tau(P(x))) = \tau_{out} \lor \tau(P(x)) = \tau_{bi}) \& (\tau(P(y))) = \tau_{in} \lor \tau(P(y)) = \tau_{bi}) \}$$
$$\Leftrightarrow \{(v(x), v(y) \in A_B\}.$$

Для построенного коммутационного графа можно определить степени вершин и расстояния между вершинами в классическом понимании из теории графов [83]:

 $d^{-}(v) = |(v, u): (v, u) \in A_{B}|$ - локальная исходящая степень (полустепень исхода) вершины;

 $d^+(v) = |(u, v): (u, v) \in A_B|$ - локальная входящая степень (полустепень захода) вершины;

 $d(v) = d^{-}(v) + d^{+}(v)$ - полная локальная степень вершины.

При правильном построении должны выполняться следующие соотношения параметров графа коммутации с параметрами, определенными на основе анализа типов выводов блоков:

$$d^{-}(v(e)) \le d^{-}(m(e)) + d^{*}(m(e)),$$

$$d^{+}(v(e)) \le d^{+}(m(e)) + d^{*}(m(e)).$$

Количество коммутаций логического элемента на внутриблочном уровне (а также как в "плоском" представлении) может быть определено на основе информации о трассировочных выводах по аналогии с коммутациями блока следующим образом:

для подсхемы $s \in L_{LE}$ степень элемента s, т. е. количество цепей, подключенных к контактам s:

$$d(s) = |P_r(s)|.$$

Тогда количество коммутаций для выходов логического элемента – исходящая степень s:

$$d^{-}(s) = |P_{r}^{-}(s)|,$$
$$P_{r}^{-}(s) = \{p_{i}^{-} \in P_{r}(s) : \tau(p_{i}^{-}) = \tau_{out}\}$$

а количество коммутаций для входов логического элемента – входящая степень s:

$$d^+(s) = |P_r^+(s)|,$$

$$P_r^+(s) = \{ p_i^+ \in P_r(s) : \tau(p_i^+) = \tau_{inp} \}.$$

Двунаправленные выводы, как правило, не используются в логических элементах.

Коммутационный граф блочного уровня $G_B(\Pi_b) = (V_B, A_B)$ целесообразно использовать на предварительной стадии программного прототипирования до того, как спроектирована топология блоков базового кристалла. Если топологическая реализация блоков известна, и известны координаты выводов блока, то для более точного отображения схемы соединений вместо графа коммутаций может быть построен трассировочный граф, где множество вершин соответствует не элементам, а узлам схемы, и разные выводы элементов анализируются независимо друг от друга. В разделе 2.6 рассматривается полный трассировочный граф базового кристалла $G_{Ro}(\Pi) = (V_{Ro}, A_{Ro})$ применительно к плоскому представлению трассировочных ресурсов для реализации конкретного варианта схемы заказчика.

Для заданного двухуровневого блочного представления проекта $\Pi_b(\Pi) = (S_b, L_b, s_b, m_b)$, где $s_b = (\mu(s_b), E(s_b), N(s_b), P(s_b), C(s_b))$, трассировочный граф блочного уровня $G_B^*(\Pi_b) = (V_B^*, A_B^*)$ может быть построен аналогичным образом:

Множество вершин V_B^* графа $G_B^*(\Pi_b)$ взаимно однозначно соответствует подмножеству таких и только таких узлов блочного уровня $n \in N(s_b)$, для которых существует хотя бы одно соединение с элементом из подмножества трассировочных элементов или блоков:

$$\{n: n \in N(s_b) \& (\exists e \in E(s_b): m(e) \in L_{Ro}) \& (\exists p \in P(e): (p, n) \in C(s_b)) \Leftrightarrow \{v(n): v(n) \in V_B^*.$$

В качестве замечания необходимо отметить: предполагается, что в конструкции кристалла отсутствуют такие варианты, когда выводы блоков, макроблоков или периферийных элементов соединены друг с другом напрямую без трассировочных элементов.

Множество дуг A_B^* формируется из информации о соединениях цепей (узлов) на межблочном уровне - для каждой пары узлов $x \in N(s_b)$, $y \in N(s_b)$, контакты которых соединены трассировочным элементом и выполнены правила распространения сигнала в соответствии типами портов:

$$\{\exists e \in E(s_b) : m(e) \in L_{Ro} \& (P_1(e), x) \in C(s_b) \& (P_2(e), y) \in C(s_b) \& (\tau(P_2(e)) = \tau_{out} \lor \tau(P_2(e)) = \tau_{bi}) \& (\tau(P_1(e)) = \tau_{in} \lor \tau(P_1(e)) = \tau_{bi})\}$$
$$\Leftrightarrow \{(v(x), v(y) \in A_B^*\}.$$

Для построенного трассировочного графа блочного уровня также можно определить степени вершин и расстояния между вершинами в классическом понимании из теории графов [83].

Общее количество входов всех логических элементов в блоке в значительной степени определят необходимое количество локальных трасс $|N_{loc}(b)|$, где $N_{loc}(b) \subset N(b)$ - подмножество цепей (узлов) в блоке b, которые обеспечивают внутренние коммутации между входами и выходами логических элементов в блоке, а также могут быть использованы для последующих связей с внешними цепями за пределами блока. Поскольку общее количество элементов в блоке определяется размером матрицы M_{LE} : $|E_{LE}(b)| = |M_{LE}| = I_{LE} \cdot J_{LE}$, а количество коммутаций для входов логического элемента – *входящей степенью* $P_r^+(s), s \in L_{LE}$, то:

 $|N_{loc}(b)| \approx P_r^+(s) \cdot |E_{LE}(b)| = P_r^+(s) \cdot |M_{LE}| = P_r^+(s) \cdot I_{LE} \cdot J_{LE}.$

Реальное количество локальных трасс в блоке $|N_{loc}(b)|$ может быть меньше, чем указано в приближенном неравенстве за счет следующих возможностей:

- некоторые входы логических элементов могут использовать общие входные сигналы;

- некоторые входы логических элементов могут быть замкнуты на землю или питание;

- некоторые логические элементы могут не использоваться в конкретных блоках.

В связи с перечисленными факторами следует ориентироваться на понижающий коэффициент К_{loc} ∈ (0, 1]:

 $|N_{loc}(b)| = \mathcal{K}_{loc} \cdot P_r^+(s) \cdot |E_{LE}(b)| = \mathcal{K}_{loc} \cdot P_r^+(s) \cdot I_{LE} \cdot J_{LE}.$

Фактическое значение коэффициента К_{loc} может быть подобрано экспериментально на этапе программного прототипирования базового кристалла с анализом схем заказчиков от конкретного потребителя.

Также, как и в случае блоков, для более детального анализа можно построить граф коммутаций логических элементов на внутриблочном уровне (или в «плоском» представлении) с индивидуальным определением степеней вершин в составе графа коммутаций для каждого логического элемента.

Трассировочные возможности базового кристалла в значительной степени определяются количеством связей между блоками и их структурой, в том числе степенью ветвления цепей.

Для канальной архитектуры с четко выделенными областями каналов и переключательных коробок (switch box) ключевыми параметрами, определяющими трассировочные возможности базового кристалла являются:

- пропускная способность вертикальных каналов между блоками;

- пропускная способность горизонтальных каналов между блоками;

- пропускная способность переключательной коробки (switch box) между каналами.

Однако при разработке базового кристалла архитектура и топология могут отличаться от канальной. В этом случае следует ориентироваться на такие характеристики, как:

- пропускная способность сечения по вертикали для заданного Х;

- пропускная способность сечения по горизонтали для заданного Ү.

Под пропускной способностью сечений подразумевается максимально возможное количество трасс (цепей), пересекающих указанное сечение. Задача анализа пропускной способности сечений может быть сведена к известной проблеме поиска максимального потока, однако в силу вычислительной сложности последней требуется разработка эвристических оценок меньшей сложности.

Наиболее целесообразным представляется анализ пропускной способности сечений на межблочном уровне в глобальной системе координат базового кристалла. Такой вариант ближе других к анализу канальной архитектуры. Другие возможные варианты: сечений анализ на внутриблочном уровне в локальной системе координат блока и анализ полного кристалла в "плоском" представлении в глобальной системе координат.

При анализе трассировочных возможностей на межблочном уровне заданное двухуровневое представление проекта $\Pi_b(\Pi) = (S_b, L_b, s_b)$, где $s_b =$ $(\mu(s_b), E(s_b), N(s_b), P(s_b), C(s_b))$, подмножество блочных элементов $E_B(s_b) \subset E(s_b)$, $E_B(s_b) = \{e: e \in E(s_b) \& m(e) \in B\}$ целесообразно рассматривать в виде матрицы блоков:

 $M_b = \{m_{ij} : m_{ij} \in E_B(s_b), m(m_{ij}) \in B, i = 1, ..., I_b, j = 1, ..., J_b \}.$

Предполагается, что все блоки (LAB) базового кристалла имеют одинаковую схемотехническую и топологическую реализацию $m(m_{ij}) = b, B = \{b\}$, и для блока b известны границы в его локальной системе координат:

$$X_{min}(b), Y_{min}(b), X_{max}(b), Y_{max}(b).$$

Для каждого экземпляра блока m_{ij} известны координаты точки привязки в глобальной системе координат базового кристалла и ориентация:

$$X_0(m_{ij}), Y_0(m_{ij}), O_r(m_{ij}),$$

где $O_r(m_{ij}) \in \{O_0, O_R, O_{XY}, O_{XYR}, O_Y, O_{XR}, O_X, O_{XR}\},\$

при этом, индексация ориентаций указывает на отсутствие (0) или наличие поворотов (R - поворот против часовой стрелки на 90°) и зеркальных отображений (X, Y) относительно соответствующей оси.

Пересчет из локальной системы координат блока (X₁, Y₁) в глобальную систему координат кристалла (X_g, Y_g) для ортогональной топологии осуществляется по формулам:

$$\begin{split} \mathbf{X}_{g} &= \Omega_{\mathbf{X}}(\mathbf{X}_{\mathbf{l}},\mathbf{Y}_{\mathbf{l}},\mathbf{X}_{0},\mathbf{Y}_{0},\mathbf{O}_{\mathbf{r}}), \ \mathbf{Y}_{g} = \Omega_{\mathbf{Y}}(\mathbf{X}_{\mathbf{l}},\mathbf{Y}_{\mathbf{l}},\mathbf{X}_{0},\mathbf{Y}_{0},\mathbf{O}_{\mathbf{r}}), \ \mathbf{\Gamma}\mathbf{d}\mathbf{e} \\ \\ \Omega_{\mathbf{X}}(\mathbf{X}_{\mathbf{l}},\mathbf{Y}_{\mathbf{l}},\mathbf{X}_{0},\mathbf{Y}_{0},\mathbf{O}_{\mathbf{r}}) &= \begin{cases} \mathbf{X}_{0} + \mathbf{X}_{\mathbf{l}} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{O}_{\mathbf{r}} \in \{\mathbf{0}_{0},\mathbf{0}_{\mathbf{X}}\}, \\ \mathbf{X}_{0} - \mathbf{X}_{\mathbf{l}} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{O}_{\mathbf{r}} \in \{\mathbf{0}_{\mathbf{Y}},\mathbf{0}_{\mathbf{X}\mathbf{Y}}\}, \\ \mathbf{X}_{0} - \mathbf{Y}_{\mathbf{l}} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{0}_{\mathbf{r}} \in \{\mathbf{0}_{\mathbf{R}},\mathbf{0}_{\mathbf{Y}\mathbf{R}}\}, \\ \mathbf{X}_{0} + \mathbf{Y}_{\mathbf{l}} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{0}_{\mathbf{r}} \in \{\mathbf{0}_{\mathbf{X}\mathbf{R}},\mathbf{0}_{\mathbf{X}\mathbf{Y}\mathbf{R}}\}; \\ \\ \boldsymbol{\Omega}_{Y}(\mathbf{X}_{l},\mathbf{Y}_{l},\mathbf{X}_{0},\mathbf{Y}_{0},\mathbf{0}_{\mathbf{r}}) &= \begin{cases} Y_{0} + Y_{l} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{0}_{\mathbf{r}} \in \{\mathbf{0}_{\mathbf{X}},\mathbf{0}_{\mathbf{X}\mathbf{Y}\mathbf{R}}\}, \\ Y_{0} - Y_{l} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{0}_{\mathbf{r}} \in \{\mathbf{0}_{\mathbf{X}},\mathbf{0}_{\mathbf{X}\mathbf{Y}}\}, \\ Y_{0} - X_{l} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{0}_{\mathbf{r}} \in \{\mathbf{0}_{\mathbf{X}},\mathbf{0}_{\mathbf{X}\mathbf{Y}\mathbf{R}}\}, \\ Y_{0} + X_{l} \ \mathbf{\Pi}\mathbf{p}\mathbf{\mu} \ \mathbf{0}_{\mathbf{r}} \in \{\mathbf{0}_{\mathbf{R}},\mathbf{0}_{\mathbf{X}\mathbf{R}}\}. \end{cases} \end{split}$$

В соответствии с формулами пересчета координаты центров блоков (X_c, Y_c) в глобальной система координат можно получить по формулам:

$$\begin{aligned} X_{c}(b) &= \frac{X_{min}(b) + X_{max}(b)}{2}, \\ Y_{c}(b) &= \frac{Y_{min}(b) + Y_{max}(b)}{2}, \\ X_{c}(m_{ij}) &= \Omega_{X}(X_{c}(b), Y_{c}(b), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij})), \\ Y_{c}(m_{ij}) &= \Omega_{Y}(X_{c}(b), Y_{c}(b), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij})). \end{aligned}$$

Соответственно, угловые точки $X_{min}(m_{ij})$, $Y_{min}(m_{ij})$, $X_{max}(m_{ij})$, $Y_{max}(m_{ij})$ находятся по формулам:

$$\begin{split} X_{1}(m_{ij}) &= \Omega_{X} \left(X_{min}(b), Y_{min}(b), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij}) \right), \\ Y_{1}(m_{ij}) &= \Omega_{Y} \left(X_{min}(b), Y_{min}(b), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij}) \right), \\ X_{2}(m_{ij}) &= \Omega_{X} \left(X_{max}(b), Y_{max}(b), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij}) \right), \\ Y_{2}(m_{ij}) &= \Omega_{Y} \left(X_{max}(b), Y_{max}(b), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij}) \right), \\ X_{min}(m_{ij}) &= min \left(X_{1}(m_{ij}), X_{2}(m_{ij}) \right), \\ Y_{min}(m_{ij}) &= max \left(X_{1}(m_{ij}), X_{2}(m_{ij}) \right), \end{split}$$

$$Y_{max}(m_{ij}) = max\left(Y_1(m_{ij}), Y_2(m_{ij})\right).$$

В приближенных расчетах, а также на этапе предварительного проектирования, когда окончательная топология блоков неизвестна, координаты выводов блока $X_P(p), Y_P(p), p \in P(m_{ij})$ можно связать с его центром:

$$X_P(p) \approx X_c(m_{ij}), Y_P(p) \approx Y_c(m_{ij}), p \in P(m_{ij}).$$

Более точные значения глобальных координат выводов блоков вычисляются на основе информации о локальных координатах выводов в системе координат блока $X_P^{loc}(p), Y_P^{loc}(p), p \in P(m(m_{ij}))$:

$$\begin{aligned} X_{P}(p) &= \Omega_{X}(X_{P}^{loc}(p), Y_{P}^{loc}(p), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij})), \\ Y_{P}(p) &= \Omega_{Y}(X_{P}^{loc}(p), Y_{P}^{loc}(p), X_{0}(m_{ij}), Y_{0}(m_{ij}), O_{r}(m_{ij})). \end{aligned}$$

Следует отметить, что приведенные формулы для размеров и координат справедливы не только для матрицы блоков логических элементов $M_b = \{m_{ij}: m_{ij} \in E_B(s_b), m(m_{ij}) \in B, i = 1, ..., I_b, j = 1, ..., J_b\}$, но также для периферийных элементов:

$$E_{IO}(s_b) \subset E(s_b), E_{IO}(s_b) = \{e : e \in E(s_b) \& m(e) \in L_{IO}\}$$

и для макроблоков:

$$E_M(s_b) \subset E(s_b), E_M(s_b) = \{e : e \in E(s_b) \& m(e) \in L_M\}.$$

Если в матрице блоков, как правило, применяется нормальная ориентация $O_r(m_{ij}) = O_0$, для макроблоков и периферийных элементов это не так, и пересчет с применением $\Omega_X(X_l, Y_l, X_0, Y_0, O_r)$, $\Omega_Y(X_l, Y_l, X_0, Y_0, O_r)$ становится более значимым, а именно, $\forall e \in E_{IO}(s_b) \cup E_M(s_b), m = m(e)$:

$$\begin{split} X_{c}(m) &= \frac{X_{min}(m) + X_{max}(m)}{2}, \\ Y_{c}(m) &= \frac{Y_{min}(m) + Y_{max}(m)}{2}, \\ X_{c}(e) &= \Omega_{X}(X_{c}(m(e)), Y_{c}(m(e)), X_{0}(e), Y_{0}(e), O_{r}(e)), \\ Y_{c}(e) &= \Omega_{Y}(X_{c}(m(e)), Y_{c}(m(e)), X_{0}(e), Y_{0}(e), O_{r}(e)). \\ X_{1}(e) &= \Omega_{X}(X_{min}(m(e)), Y_{min}(m(e)), X_{0}(e), Y_{0}(e), O_{r}(e)), \\ Y_{1}(e) &= \Omega_{Y}(X_{min}(m(e), Y_{min}(m(e), X_{0}(e), Y_{0}(e), O_{r}(e)), \end{split}$$

$$\begin{split} X_{2}(e) &= \Omega_{X} \Big(X_{max}(m(e), Y_{max}(m(em), X_{0}(e), Y_{0}(e), O_{r}(e)) \big), \\ Y_{2}(e) &= \Omega_{Y} \Big(X_{max}(m(e), Y_{max}(m(e), X_{0}(e), Y_{0}(e), O_{r}(e)) \big), \\ X_{min}(e) &= min \Big(X_{1}(e), X_{2}(e) \Big), \\ Y_{min}(e) &= min \Big(Y_{1}(e), Y_{2}(e) \Big), \\ X_{max}(e) &= max \Big(X_{1}(e), X_{2}(e) \Big), \\ Y_{max}(e) &= max \Big(Y_{1}(e), Y_{2}(e) \Big). \end{split}$$

В матрице блоков традиционно выравниваются координаты по горизонтали для элементов с равным первым индексом, и по вертикали для элементов с равным вторым индексом:

$$\begin{aligned} \forall i, j, k: X_c(m_{ij}) &= X_c(m_{kj}), \\ \forall i, j, k: Y_c(m_{ij}) &= Y_c(m_{ik}), \\ \forall i, j, k: X_{min}(m_{ij}) &= X_{min}(m_{kj}) \stackrel{\text{def}}{=} X_{min}^j(M_b), \\ \forall i, j, k: Y_{min}(m_{ij}) &= Y_{min}(m_{ik}) \stackrel{\text{def}}{=} Y_{min}^i(M_b), \\ \forall i, j, k: X_{max}(m_{ij}) &= X_{max}(m_{kj}) \stackrel{\text{def}}{=} X_{max}^j(M_b), \\ \forall i, j, k: Y_{max}(m_{ij}) &= Y_{max}(m_{ik}) \stackrel{\text{def}}{=} Y_{max}^i(M_b), \end{aligned}$$

при этом блоки в столбцах и строках упорядочены по соответствующим координатам:

$$\forall i, j, k: j < k \rightarrow X_c(m_{ij}) < X_c(m_{ik}).$$

$$\forall i, j, k: i < k \rightarrow Y_c(m_{ij}) < Y_c(m_{kj}),$$

При таких условиях выполняются следующие граничные равенства:

$$\begin{aligned} X_{c}^{min}(M_{b}) &= \min_{i,j} X_{c}(m_{ij}) = X_{c}(m_{11}), \\ Y_{c}^{min}(M_{b}) &= \min_{i,j} Y_{c}(m_{ij}) = Y_{c}(m_{11}), \\ X_{c}^{max}(M_{b}) &= \max_{i,j} X_{c}(m_{ij}) = X_{c}(m_{I_{b}J_{b}}), \\ Y_{c}^{max}(M_{b}) &= \max_{i,j} Y_{c}(m_{ij}) = Y_{c}(m_{I_{b}J_{b}}). \end{aligned}$$

В целях обобщения механизма упорядоченного индексирования на периферийные элементы и макроблоки целесообразно выделить следующие группы элементов по их расположению относительно матрицы блоков:

$$E_{top}(s_b) \subset E(s_b),$$

$$E_{top}(s_b) = \{e: e \in E(s_b) \& m(e) \in L_{IO} \cup L_M \& Y_c(e) > Y_c^{max}\};$$

$$E_{bottom}(s_b) \subset E(s_b),$$

$$E_{bottom}(s_b) = \{e: e \in E(s_b) \& m(e) \in L_{IO} \cup L_M \& Y_c(e) < Y_c^{min}\};$$

$$E_{left}(s_b) \subset E(s_b),$$

$$E_{left}(s_b) = \{e: e \in E(s_b) \& m(e) \in L_{IO} \cup L_M \& X_c(e) < X_c^{min}\};$$

$$E_{right}(s_b) \subset E(s_b),$$

$$E_{right}(s_b) = \{e: e \in E(s_b) \& m(e) \in L_{IO} \cup L_M \& X_c(e) > X_c^{max}\}.$$

образуют Возможные пересечения приведенных подмножеств множества элементов для углов. Расширяя механизм упорядоченного пределы матрицы индексирования за можно проиндексировать периферийные элементы и макроблоки следующим образом (один из индексов в ряде случаев остается неопределенным, поскольку размеры и порядок периферийных элементов и макроблоков не соответствуют размерам и порядку основной матрицы блоков):

$$\{m_{00}^{k}\} = E_{bottom}(s_{b}) \cap E_{left}(s_{b});$$

$$\{m_{0,J_{b}+1}^{k}\} = E_{bottom}(s_{b}) \cap E_{right}(s_{b});$$

$$\{m_{I_{b},+1,0}^{k}\} = E_{top}(s_{b}) \cap E_{left}(s_{b});$$

$$\{m_{I_{b}+1,J_{b}+1}^{k}\} = E_{top}(s_{b}) \cap E_{right}(s_{b});$$

$$\{m_{0,*}^{k}\} = E_{bottom}(s_{b}) \setminus E_{left}(s_{b}) \setminus E_{right}(s_{b});$$

$$\{m_{I_{b}+1,*}^{k}\} = E_{top}(s_{b}) \setminus E_{left}(s_{b}) \setminus E_{right}(s_{b});$$

$$\{m_{*,0}^{k}\} = E_{left}(s_{b}) \setminus E_{top}(s_{b}) \setminus E_{bottom}(s_{b});$$

$$\{m_{*,J_{b}+1}^{k}\} = E_{right}(s_{b}) \setminus E_{top}(s_{b}) \setminus E_{bottom}(s_{b}).$$

Обобщая спецификации граничных размеров столбцов и строк основной матрицы на периферийные элементы и макроблоки, можно определить следующие предельные значения координат окружения:

$$\begin{split} X^{0}_{max}(M_{b}) & \stackrel{\text{def}}{=} \max_{e \in E_{left}} X_{max}(e); \\ Y^{0}_{max}(M_{b}) & \stackrel{\text{def}}{=} \max_{e \in E_{bottom}} Y_{max}(e); \\ X^{\text{I}_{b}+1}_{min}(M_{b}) & \stackrel{\text{def}}{=} \min_{e \in E_{right}} X_{min}(e); \\ Y^{\text{J}_{b}+1}_{min}(M_{b}) & \stackrel{\text{def}}{=} \min_{e \in E_{top}} Y_{min}(e). \end{split}$$

С учетом индексации, расширенной на макроблоки и периферийные элементы за пределами основной матрицы, можно выделить следующие критические сечения для анализа пропускной трассировочной способности, как средние между соответствующими рядами или столбцами элементов и блоков:

$$X_{S}^{k} = \frac{X_{max}^{k}(M_{b}) + X_{min}^{k+1}(M_{b})}{2}, k = 0, \dots, I_{b},$$
$$Y_{S}^{k} = \frac{Y_{max}^{k}(M_{b}) + Y_{min}^{k+1}(M_{b})}{2}, k = 0, \dots, J_{b}.$$

На уровне блочного представления $\Pi_b(\Pi) = (S_b, L_b, s_b)$, где $s_b = (\mu(s_b), E(s_b), N(s_b), P(s_b), C(s_b))$ пропускная способность $\sigma(X_S^k), \sigma(Y_S^k)$ соответствующего сечения есть количество трасс (цепей), пересекающих сечение. Если топология и координаты цепей $X_{min}(n), Y_{min}(n), X_{max}(n), Y_{max}(n), n \in N(s_b)$ известны, то без учета цепей внутри блоков можно оценить пропускную способность по формулам:

$$\sigma_X(X_S^k) = |\{n: n \in N(s_b) \& X_{min}(n) < X_S^k \& X_{max}(n) \ge X_S^k\}|,\$$

$$\sigma_Y(Y_S^k) = |\{n: n \in N(s_b) \& Y_{min}(n) < Y_S^k \& Y_{max}(n) \ge Y_S^k\}|.$$

На определенных этапах проектирования возможны ситуации, когда топология или координаты цепей неизвестны. В этом случае требуется предварительная оценка координат узлов (цепей) и трассировочных элементов (ключей, управляемых буферов и др.).

Для предварительной оценки координат цепей и трассировочных элементов можно использовать пути (маршруты) в коммутационном графе $G_B(\Pi_b) = (V_B, A_B)$, вершины которого ассоциированы с элементами схемы, или в трассировочном графе $G_B^*(\Pi_b) = (V_B^*, A_B^*)$, вершины которого ассоциированы с узлами (цепями) схемы. Здесь и далее будет рассматриваться коммутационный граф $G_B(\Pi_b) = (V_B, A_B)$, при этом все, что связано с построением путей и оценкой расстояний, в равной степени справедливо и для трассировочного графа $G_B^*(\Pi_b) = (V_B^*, A_B^*)$. Выбор того или иного графа зависит от этапа проектирования, наличия или отсутствия информации о топологии блоков и координат выводов.

Примечание. Следует различать понятие цепи в схеме и цепи в графе. Цепь в схеме $s = (\mu(s), E(s), N(s), P(s), C(s))$ - это топологическая реализация узла $n \in N(s)$. Цепью в теории графов называют путь или маршрут без повтора ребер (дуг в случае ориентированного графа). Простой цепью в теории графов называют цепь без повтора вершин, т.е. путь или маршрут без повтора вершин и без повтора ребер (дуг) [83].

Среди путей (маршрутов) в ориентированном графе $G_B(\Pi_b) = (V_B, A_B)$ (или $G_B^*(\Pi_b) = (V_B^*, A_B^*)$) наибольший интерес представляют простые ориентированные цепи, т.е. маршруты без повторов вершин и без повторов дуг. Предполагается, что граф связный, т.е., существует путь из любой вершины в любую другую вершину. В этом случае для любой пары вершин $a \in V_B, b \in V_B$ существует простая цепь наименьшей длины, называемой расстоянием l(a, b). Возможны различные варианты определения длины цепи с точной оценкой расстояния между вершинами или оценкой задержки распространения сигнала. Простейший вариант оценки длины простой цепи количество дуг в цепи:

$$\forall a \in V_B : l(a, a) = 0;$$

$$\forall (a, b) \in A_B : a \neq b \rightarrow l(a, b) = 1;$$

$$\forall a \in V_B, \forall c \in V_B : (a, c) \notin A_B \rightarrow$$

$$l(a,c) = \underset{b:(a,b)\in A_B}{1} + \underset{b:(b,c)\in A_B}{\min} l(a,c) = \underset{b:(b,c)\in A_B}{1} + \underset{b:(b,c)\in A_B}{\min} l(a,b)$$

Известно, что в случае неориентированного графа расстояние l(a, b), определенное, как минимальная длина простой цепи, удовлетворяет всем аксиомам метрики в метрическом пространстве [83].

В случае ориентированного графа это не совсем так:

$$\begin{aligned} &\forall a \in V_B : l(a, a) = 0; \\ &\forall a \in V_B, \forall b \in V_B : l(a, b) \ge 0; \\ &\forall a \in V_B, \forall b \in V_B, \forall c \in V_B : l(a, b) + l(b, c) \ge l(a, c). \end{aligned}$$

Однако в ориентированном графе не соблюдается требование l(a, b) = l(b, a).

Тем не менее, для $l^*(a, b) = \min(l(a, b), l(b, a))$ фактически имеем расстояние в соотнесенном неориентированном графе с соблюдением всех аксиом метрики метрического пространства.

Как и в случае неориентированного графа, для каждой выделенной вершины $a \in V_B$, множество вершин V_B в связном графе разбивается на непересекающиеся подмножества вершин с равными расстояниями от вершин подмножества до выделенной вершины $a \in V_B$:

$$V_B = \bigcup_n V_n^+(a); V_n^+(a) = \{v: l(a, v) = n\};$$

 $V_B = \bigcup_n V_n^-(a); V_n^-(a) = \{v: l(v, a) = n\};$
 $V_B = \bigcup_n V_n^*(a); V_n^*(a) = \{v: l^*(a, v) = n\}.$
Очевидно, что $V_0^+(a) = V_0^-(a) = V_0^*(a) = \{a\}.$

Важно также отметить, что условие связности графа должно соблюдаться в силу необходимости реализации различных соединений схемы заказчика для всех вершин, ассоциированных с блоками логических элементов, периферийными элементами и макроблоками ($e \in E(s_b), m(e) \in$ $(B \cup L_{IO} \cup L_M)$). Что касается трассировочных элементов ($e \in E(s_b), m(e) \in$ L_{Ro}), то для них условие связности должно соблюдаться в варианте соотнесенного неориентированного графа, т.е. для $V_n^*(a)$, но не всегда соблюдается в ориентированном варианте для $V_n^+(a), V_n^-(a)$. Проверка условий связности, а также построение и анализ подмножеств $V_n^+(a), V_n^-(a), V_n^*(a)$ на начальном этапе проектирования могут существенно сократить время отладки проекта на этапе окончательной трассировки.

Анализ таких подмножеств позволяет выявить наиболее критические с точки зрения быстродействия схемы связи элементов базового кристалла и минимизировать количество критических связей на этапах размещения и трассировки при реализации проектов заказчиков. Обобщая схему разбиения на случай несвязного графа следует выделить подмножества вершин $V_{\infty}^+(a), V_{\infty}^-(a), V_{\infty}^*(a)$, для которых не существует пути соответствующего направления:

$$V_B \setminus V_{\infty}^{+}(a) = \bigcup_n V_n^{+}(a); V_n^{+}(a) = \{v: l(a, v) = n\};$$

$$V_B \setminus V_{\infty}^{-}(a) = \bigcup_n V_n^{-}(a); V_n^{-}(a) = \{v: l(v, a) = n\};$$

$$V_B \setminus V_{\infty}^{*}(a) = \bigcup_n V_n^{*}(a); V_n^{*}(a) = \{v: l^{*}(a, v) = n\}.$$

Одним из вариантов нахождения каждого их подмножеств $V_n^+(a), V_n^-(a), V_n^*(a)$ является алгоритм Дейкстры [84] с построением остовного дерева соответственно $T^+(a), T^-(a), T^*(a)$ с корнем в вершине, соответствующей (*a*), разделением вершин на слои $\{t_{i*}^+\}, \{t_{i*}^-\}, \{t_{i*}^*\}$ в соответствии с нумерацией и делением вершин исходного графа на слои $V_i^+(a), V_i^-(a), V_i^*(a)$, где ребра $\{a_{ij}^+\}, \{a_{ij}^-\}, \{a_{ij}^*\}$ обозначают связь каждой из последующих вершин с ее предшественником в кратчайшем пути от (*a*):

$$T^{+}(a) = (\{t_{ij}^{+}\}, \{a_{ij}^{+}\}); t_{00}^{+} \leftrightarrow a; \{t_{i*}^{+}\} \leftrightarrow V_{i}^{+}(a);$$

$$T^{-}(a) = (\{t_{ij}^{-}\}, \{a_{ij}^{-}\}); t_{00}^{-} \leftrightarrow a; \{t_{i*}^{-}\} \leftrightarrow V_{i}^{-}(a);$$

$$T^{*}(a) = (\{t_{ij}^{*}\}, \{a_{ij}^{*}\}); t_{00}^{*} \leftrightarrow a; \{t_{i*}^{*}\} \leftrightarrow V_{i}^{*}(a).$$

Построенные таким образом остовные деревья в значительной степени характеризуют структуру межсоединений и степень ветвления по отношению к выделенному исходному элементу, ассоциированному с (*a*).

Учитывая, как минимум, квадратичную сложность построения полного множества деревьев $T^+(v), T^-(v), T^*(v)$, целесообразно в практической

реализации ограничить глубину поиска некоторым предельным значением $n \leq n_{max}$. Само предельное значение n_{max} может быть разным для разных групп вершин, в зависимости от типа ассоциированных элементов схемы $m(e) \in (B \cup L_{IO} \cup L_M \cup L_{RO})$. Несмотря на вычислительную сложность, построение полного множества деревьев $T^+(v), T^-(v), T^*(v)$ оправдано, поскольку они могут быть использованы не только для анализа пропускной способности сечений, но и для решения целого ряда других задач, таких как, выявление критических с точки зрения быстродействия схемы путей в базовом кристалле, минимизации количества и длин критических связей на этапах размещения и трассировки при реализации проектов заказчиков.

Для коммутационного графа $G_B(\Pi_b) = (V_B, A_B)$, построенного по блочному представлению схемы $\Pi_b(\Pi) = (S_b, L_b, s_b)$, где $s_b = (\mu(s_b), E(s_b), N(s_b), P(s_b), C(s_b))$, каждая из вершин $v \in V_B$ соответствует некоторому элементу $e \in E(s_b), m(e) \in (B \cup L_{IO} \cup L_M \cup L_{RO})$. При этом, как правило, координаты блоков логических элементов, периферийных элементов и макроблоков ($e \in E(s_b), m(e) \in (B \cup L_{IO} \cup L_M)$) известны заранее, а координаты трассировочных элементов ($e \in E(s_b), m(e) \in L_{RO}$) на определенном этапе проектирования до разработки окончательной топологии базового кристалла неизвестны.

Пусть $\xi: V_B \to \{0,1\};$ $\xi(v) = \begin{cases} 1, если X_c(v), Y_c(v) \text{ известны или определены ранее} \\ 0, в противном случае. \end{cases}$

При $\xi(a) = 0$ можно оценить реальные координаты соответствующего элемента схемы, учитывая, что с ростом *n* должно увеличиваться и расстояние в метрике Манхэттена для центров соответствующих элементов:

$$\begin{split} L(v,a) &= L(a,v) = |X_c(v) - X_c(a)| + |Y_c(v) - Y_c(a)|, \\ \text{для } v \in V_n^*(a), \xi(v) = 1. \end{split}$$

При этом, заранее известны следующие координаты и расстояния:

$$L(v,u) = L(u,v) = |X_c(v) - X_c(u)| + |Y_c(v) - Y_c(u)|,$$
при $\xi(v) = \xi(u) = 1.$

Для каждой такой пары $(v, u), \xi(v) = \xi(u) = 1$ можно оценить приближенные значения координат $X_c(a), Y_c(a)$, исходя из ожидаемых соотношений пропорциональности расстояний:

$$\frac{\mathrm{L}(\mathrm{a},\mathrm{v})}{\mathrm{l}^*(\mathrm{a},\mathrm{v})} \approx \frac{\mathrm{L}(\mathrm{a},\mathrm{u})}{\mathrm{l}^*(\mathrm{a},\mathrm{u})} \approx \frac{\mathrm{L}(\mathrm{v},\mathrm{u})}{\mathrm{l}^*(\mathrm{v},\mathrm{u})}.$$

Наиболее точный результат следует ожидать для ситуации, когда $l^*(v, u) = l^*(v, a) + l^*(a, u)$. В этом случае логично искать значения координат $X_c(a), Y_c(a)$ путем линейной интерполяции пропорционально по отношению к $l^*(v, a), l^*(a, u)$ в интервалах:

$$X_c(a) \in [\min(X_c(v), X_c(u)), \max(X_c(v), X_c(u))],$$

$$Y_c(a) \in [\min(Y_c(v), Y_c(u)), \max(Y_c(v), Y_c(u))].$$

Очевидно, что $l^*(v, u) \le l^*(v, a) + l^*(a, u)$ в силу аксиомы треугольника для метрических пространств.

Аналогичный подход с интерполяцией возможен, но с меньшей точностью, при незначительных отклонениях от равенства:

$$l^*(v,u) + \varepsilon = l^*(v,a) + l^*(a,u), 0 \le \varepsilon \ll l^*(v,u).$$

Другие варианты с экстраполяцией за пределы известных интервалов представляются менее точными, но также возможны:

$$l^{*}(a,v) + \varepsilon = l^{*}(a,u) + l^{*}(u,v), 0 \le \varepsilon \ll l^{*}(a,v),$$

$$l^{*}(a,u) + \varepsilon = l^{*}(a,v) + l^{*}(v,u), 0 \le \varepsilon \ll l^{*}(a,u).$$

В первом случае логично искать значения координат $X_c(a)$, $Y_c(a)$ путем линейной экстраполяции со стороны (u), во втором случае - со стороны (v) пропорционально по отношению к $l^*(v, a)$, $l^*(a, u)$ за пределами интервалов:

$$X_c(a) \notin [\min(X_c(v), X_c(u), \max(X_c(v), X_c(u)],$$
$$Y_c(a) \notin [\min(Y_c(v), Y_c(u), \max(Y_c(v), Y_c(u)].$$

В связи с вышеизложенным предлагается следующий алгоритм эвристической оценки значения координат X_c(a), Y_c(a):

 Выбрать множество пар (v_i, u_i) с известными координатами, для которых ожидается наиболее точный результат (с минимальными значениями ε, предпочтительно в варианте с интерполяцией). 2. Вычислить значения Xⁱ_c(a), Yⁱ_c(a) с применением линейной интерполяции или экстраполяции.

3. Найти усредненные значения X_c(a), Y_c(a) с применением весовых коэффициентов k_i, подобранных в зависимости от ожидаемой точности каждого из вариантов:

$$X_{c}(a) = \frac{\sum_{i} k_{i} \cdot X_{c}^{i}(a)}{\sum_{i} k_{i}},$$
$$Y_{c}(a) = \frac{\sum_{i} k_{i} \cdot Y_{c}^{i}(a)}{\sum_{i} k_{i}}.$$

Для каждого из вариантов интерполяции или экстраполяции значения $k_i = k(\epsilon_i)$, где $k(\epsilon_i)$ — монотонно убывающая функция, можно подобрать, сравнивая результаты интерполяции или экстраполяции с фактическими значениями координат для вершин графа $v, \xi(v) = 1$.

Альтернативный метод для анализа пропускной трассировочной способности критических сечений может быть организован на основе методов поиска максимального потока [83], применительно к трассировочному графу блочного уровня $G_B^*(\Pi_b) = (V_B^*, A_B^*)$, вершины которого соответствуют узлам (цепям) схемы, и следовательно, портам выводов элементов.

Предполагается, что известна архитектура базового кристалла, известны координаты блоков логических элементов, периферийных элементов и макроблоков ($e \in E(s_b), m(e) \in (B \cup L_{10} \cup L_M)$). Координаты трассировочных элементов ($e \in E(s_b), m(e) \in L_{Ro}$) до разработки окончательной топологии базового кристалла могут быть неизвестны.

Если топология блоков логических элементов, периферийных элементов и макроблоков известны, то координаты соответствующих узлов схемы, ассоциированных с вершинами графа $G_B^*(\Pi_b) = (V_B^*, A_B^*)$, и следовательно, портами выводов элементов, могут быть вычислены на основе информации о локальных координатах выводов в системе координат подсхемы элемента (блока) $X_P^{loc}(p), Y_P^{loc}(p), p \in P(m(e))$:

$$\begin{split} X_P(p) &= \Omega_X(X_P^{loc}(p), Y_P^{loc}(p), X_0(e), Y_0(e), O_r(e)), \\ Y_P(p) &= \Omega_Y(X_P^{loc}(p), Y_P^{loc}(p), X_0(e), Y_0(e), O_r(e)). \end{split}$$

В противном случае, когда топология элементов неизвестна, координаты выводов блока $X_P(p), Y_P(p), p \in P(e)$ можно связать с его центром:

$$X_P(p) \approx X_c(e), Y_P(p) \approx Y_c(e), p \in P(e)).$$

Для заданных сечений X_{S}^{k}, Y_{S}^{k} пропускная способность может быть разной для разных направлений рспространения сигналов: $\sigma^{+}(X_{S}^{k}), \sigma^{+}(Y_{S}^{k})$ - в сторону увеличения соответствующей координаты, и : $\sigma^{-}(X_{S}^{k}), \sigma^{-}(Y_{S}^{k})$ - в сторону уменьшения соответствующей координаты.

Пусть $V_B^* = V_{\xi=1}^* \cup V_{\xi=0}^*$, где $V_{\xi=1}^* = \{ v : v \in V_B^* \& \xi(v) = 1 \}$ подмножество вершин графа $G_B^*(\Pi_b) = (V_B^*, A_B^*)$ с известными координатами X(v), полученными на основе вычисления связанных с ними портов элементов, и $V_{\xi=0}^* = \{ v : v \in V_B^* \& \xi(v) = 0 \}$ - подмножество вершин графа $G_B^*(\Pi_b) = (V_B^*, A_B^*)$ с неизвестными координатами.

Каждое из сечений разделяет множество вершин $V_{\xi=1}^*$ с известными координатами на две части:

$$V_{\xi=1}^{*} = V_{X-}^{*}(X_{S}^{k}) \cup V_{X+}^{*}(X_{S}^{k}),$$

где $V_{X-}^{*}(X_{S}^{k}) = \{v: X(v) \le X_{S}^{k}\}, V_{X+}^{*}(X_{S}^{k}) = \{v: X(v) \ge X_{S}^{k}\};$
 $V_{\xi=1}^{*} = V_{Y-}^{*}(Y_{S}^{k}) \cup V_{Y+}^{*}(Y_{S}^{k}),$
где $V_{Y-}^{*}(Y_{S}^{k}) = \{v: Y(v) \le Y_{S}^{k}\}, V_{Y+}^{*}(Y_{S}^{k}) = \{v: Y(v) \ge Y_{S}^{k}\};$

Для заданного графа $G_B^*(\Pi_b) = (V_B^*, A_B^*)$ транспортная сеть $N(X_S^k) = (V_N^*(X_S^k), A_N^*(X_S^k))$ для анализа максимального потока с истоком *s* и стоком *t* строится по следующим правилам:

$$\mathrm{V}_{\mathrm{N}}^{*}(X_{S}^{k}) = \mathrm{V}_{\mathrm{B}}^{*} \cup \{s, t\};$$

для анализа $\sigma^+(X_S^k)$ с истоком соединяются элементы множества $V_{X-}^*(X_S^k)$, со стоком - элементы множества $V_{X+}^*(X_S^k)$:

$$A_{N}^{*}(X_{S}^{k}) = A_{B}^{*} \cup \{(s, v), v \in V_{X-}^{*}(X_{S}^{k})\} \cup \{(v, t), v \in V_{X+}^{*}(X_{S}^{k})\};$$

для анализа $\sigma^{-}(X_{S}^{k})$ наоборот, со стоком соединяются элементы множества $V_{X-}^{*}(X_{S}^{k})$, с истоком - элементы множества $V_{X+}^{*}(X_{S}^{k})$:

$$A_{N}^{*}(X_{S}^{k}) = A_{B}^{*} \cup \{(s, v), v \in V_{X+}^{*}(X_{S}^{k})\} \cup \{(v, t), v \in V_{X-}^{*}(X_{S}^{k})\}.$$

Аналогично строится транспортная сеть $N(Y_S^k) = (V_N^*(Y_S^k), A_N^*(Y_S^k))$ для анализа максимального потока через горизонтальное сечение:

$$\mathrm{V}_{\mathrm{N}}^{*}(Y_{S}^{k}) = \mathrm{V}_{\mathrm{B}}^{*} \cup \{s, t\};$$

для анализа $\sigma^+(Y_S^k)$ с истоком соединяются элементы множества $V_{Y-}^*(Y_S^k)$, со стоком - элементы множества $V_{Y+}^*(Y_S^k)$:

$$A_{N}^{*}(Y_{S}^{k}) = A_{B}^{*} \cup \{(s, v), v \in V_{Y-}^{*}(Y_{S}^{k})\} \cup \{(v, t), v \in V_{Y+}^{*}(Y_{S}^{k})\};$$

для анализа $\sigma^{-}(Y_{S}^{k})$ наоборот, со стоком соединяются элементы множества $V_{Y-}^{*}(Y_{S}^{k})$, с истоком - элементы множества $V_{Y+}^{*}(Y_{S}^{k})$:

$$A_{N}^{*}(Y_{S}^{k}) = A_{B}^{*} \cup \{(s, v), v \in V_{Y+}^{*}(Y_{S}^{k})\} \cup \{(v, t), v \in V_{Y-}^{*}(Y_{S}^{k})\}.$$

Для решения задачи поиска максимального потока, и следовательно, для вычисления пропускной способности сечений, можно адаптировать известные алгоритмы, такие как, алгоритм Диница [85] с вычислительной сложностью $O(V\sqrt{E})$ для варианта с единичными потоками на дугах. Дальнейшее снижение вычислительной сложности можно обеспечить эвристическими подходами, используя вместо полного исходного трассировочного графа $G_B^*(\Pi_b) = (V_B^*, A_B^*)$ для конкретных координат сечения его подграф с выбором элементов, наиболее близких по координатам к заданному сечению. В частности, в матрице блоков достаточно рассмотреть соседние строки для горизонтальных сечений или столбцы для вертикальных сечений. При этом, отбор трассировочных элементов и узлов может быть ограничен предельным значением глубины поиска в исходном графе $G_{\rm B}^*(\Pi_{\rm h}) = (V_{\rm B}^*, A_{\rm B}^*)$.

Интерес также представляет независимый анализ отдельных частей сечений с ограниченным интервалом в направлении сечения: $[X_{min}(Y_S^k), X_{max}(Y_S^k)]$ для горизонтальных сечений или : $[Y_{min}(X_S^k), Y_{max}(X_S^k)]$

для вертикальных сечений. В частности, в матрице блоков можно независимо анализировать сечения, сопоставимые по размерам с размерами блоков.

3.5 Модели и метрики для решения задач оценки архитектуры базового кристалла на основе анализа схем заказчиков. Метрики для схем заказчиков

Схема заказчика после решения задач синтеза, декомпозиции (кластеризации) и размещения элементов может быть представлена в виде двух матриц: матрицы вентилей (аналогично матрице поля ПЛИС) и матрицы, описывающей связи между вентилями (Г-образные или прямые), например, по трем точкам.

При этом решение задачи декомпозиции определяет разбиение элементов схемы заказчика на кластеры, и каждому кластеру соответствует блок базовой ПЛИС.

Решение задачи размещения определяет отображение элементов схемы заказчика в полной плоской матрице ПЛИС или, в случае двухуровневого представления, в матрице блока для кластера.

Оценка качества результатов решения задач синтеза, декомпозиции (кластеризации) и размещения может быть построена по аналогичным критериям:

- размеры кластера (и сравнение с размером блока);

количество внешних выводов кластера; в том числе количество
 входящих трасс (входов кластера), количество исходящих трасс (выходов кластера);

- количество необходимых коммутаций библиотечных элементов; в том числе количество коммутаций для входов библиотечных элементов, количество коммутаций для выходов библиотечных элементов;

- количество связей между кластерами и их структура (степень ветвления, требуемая плотность вертикальных каналов, требуемая плотность горизонтальных каналов; реальное количество цепей, пересекающих сечение

(разрез) по вертикали для заданного X; реальное количество цепей, пересекающих сечение (разрез) по горизонтали для заданного Y; требуемая пропускная способность переключательной коробки (switch box) между каналами;

- оценка быстродействия схемы (длины цепей; расчет задержек на основе статического временного анализа);

и другие.

Важно отметить, что оценка реализуемости (и степень эффективности реализации) схем заказчиков в базовой матрице ПЛИС в значительной степени зависит от того, насколько эффективно и качественно решены задачи декомпозиции (кластеризации) и размещения, которые, как известно, являются NP-сложными и требуют значительных вычислительных ресурсов для получения вовсе не самых оптимальных результатов, а лишь эвристических оценок, близких к оптимальным. При этом следует также отметить, что классические подходы к решению задач декомпозиции (кластеризации) и размещения ориентированы на внешнее окружение (связь с периферийными элементами и макроблоками), а также на знание ключевых характеристик конструкции базового кристалла.

В нашем случае, наряду со стандартным маршрутом проектирования, речь идет и об обратной (о противоположной) задаче - об определении ключевых характеристик конструкции базового кристалла, необходимых для эффективной реализации некоторого набора характерных схем заказчиков. Отсюда возникает потребность в разработке иных алгоритмов, метрик и подходов, оценок.

Возможные подходы к решению такой проблемы состоят в следующем:

1. Решение задачи размещения некоторых типов или классов схем ускоренными эффективными методами безотносительно привязки К внешнему окружению (без привязки к периферийным элементам И макроблокам), без привязки к реальной архитектуре базового кристалла. В примера качестве такого типа И класса схем можно выделить

комбинационные схемы без обратных связей, для которых наиболее эффективным решением задачи размещения в одном из направлений представляется соблюдение топологического порядка (уровней, каскадов) от входов к выходам. При этом, все же решение задачи размещения в ортогональном направлении требует применения обычных подходов.

2. Разработка компиляторов параметризованных СФ блоков с известной архитектурой. В этом случае знание архитектуры позволяет решить задачу размещения алгоритмически для данного класса СФ блоков. В качестве примера такого типа СФ блоков можно привести блоки кодирования / декодирования на основе кодов Хэмминга с параметризацией по количеству разрядов, количеству обнаруживаемых / исправляемых ошибок и др.

3. Разработка новых метрик и алгоритмов оценки необходимых ресурсов на основе анализа степени связности вершин в графовых представлениях схем заказчиков из заданного набора.

Пусть задана проектируемая схема от заказчика, функциональность которой требуется запрограммировать в кристалле от производителя:

$$\Pi_u = (S_u, L_u, s_{mu}, m_{mu}),$$

где $s_{mu} = (\mu(s_{mu}), E(s_{mu}), N(s_{mu}), P(s_{mu}), C(s_{mu})).$

Как и ранее можно предполагать, что проектируемая схема заказчика задана в «плоском» варианте, получена как результат автоматического синтеза с RTL [79] описания или результат раскрытия иерархического описания в «плоское» представление, тогда $S_u = L_u \cup \{s_{mu}\}$.

Задача размещения элементов схемы заказчика по своей сути является задачей установления соответствия *Ю* между множеством элементов библиотечного уровня проекта заказчика и элементами библиотечного уровня проекта самой конфигурируемой или программируемой гетерогенной интегральной схемы:

$$\mathscr{D}: E(s_{mu}) \to \{e: e \in E(s_f), m(e) \in \{L_{LE} \cup L_M \cup L_{IO}\}\}.$$

Обозначим через $E_{LE}(s_{mu})$, $E_M(s_{mu})$, $E_{IO}(s_{mu})$ непересекающиеся подмножества множества элементов проекта заказчика соответствующего типа:

$$E(s_{mu}) = E_{LE}(s_{mu}) \cup E_M(s_{mu}) \cup E_{IO}(s_{mu});$$

 $m(\wp(e)) \in L_{LE}$ для $e \in E_{LE};$
 $m(\wp(e)) \in L_M$ для $e \in E_M;$
 $m(\wp(e)) \in L_{IO}$ для $e \in E_{IO}.$

Предполагается, для каждого элемента $e \in E(s_{mu})$ к моменту решения задачи размещения известен не только конкретный тип библиотечного элемента $T \in \{LE, IO, M\}$, но и конкретный экземпляр $e: e \in E(s_f), m(e) \in \{L_{LE} \cup L_M \cup L_{IO}\}$, а также конкретный экземпляр библиотеки проекта заказчика $s_u \in L_u$, $s_u = (\mu(s_u), \emptyset, \emptyset, P(s_u), \emptyset)$. Другими словами, установлены соответствия для выводов библиотечных схем базового кристалла $P(s) = \{p_i, i = 1, ..., |P(s)|\}, s \in L_{LE} \cup L_{IO} \cup L_M$:

 $P_m(s) o \mathcal{B}^{|P_m(s)|}, \mathcal{B} = \{0,1\} -$ для программируемых выводов и

 $P_r(s) \to P(s_u) \cup \{P_0, P_1, P_z\}$ - для сигнальных трассировочных выводов.

Наибольшую сложность представляет задача размещения логических элементов из множества $E_{LE}(s_{mu})$. В случае блочного варианта базового кристалла целесообразно разбиение задачи размещения на два этапа: декомпозиции (или кластеризации) и финального размещения в пределах блоков (кластеров). При решении задачи декомпозиции определяется разбиение элементов схемы заказчика на кластеры, и каждому кластеру ставится в соответствие блок логических элементов базовой ПЛИС.

При разбиении $E_{LE}(s_{mu})$ на кластеры множество кластеров $K = \{K_i\}$ представляет собой множество непересекающихся подмножеств элементов $e \in E_{LE}(s_{mu})$, образующее в совокупности полное множество $E_{LE}(s_{mu})$:

$$\forall i \in \{0, 1, \dots, |K|\}: K_i \subset E_{LE}(s_{mu}),$$

$$E_{LE}(s_{mu}) = K_1 \cup K_2 \cup K_3 \cup \dots \cup K_{|K|},$$

$$\forall i, j \in \{0, 1, \dots, |K|\}, i \neq j: K_i \cap K_j = \emptyset.$$

Корректная реализация кластерного разбиения элементов схемы заказчика в базовом кристалле возможна при соблюдении следующих условий:

1) Общее количество логических элементов схемы заказчика не должно превышать емкости базового кристалла:

$$|E_{LE}(s_{mu})| \leq |E_{LE}(s_f)|,$$

$$|E_{LE}(s_f)| = |M_f| = I_f \cdot J_f = |E_B(s_b)| \cdot |E_{LE}(b)| = I_b \cdot J_b \cdot I_{LE} \cdot J_{LE}.$$

2) Количество элементов в кластере не должно превышать размера блока:

$$|K_i| \le |E_{LE}(b)|,$$
$$|E_{LE}(b)| = |M_{LE}| = I_{LE} \cdot J_{LE}$$

3) Количество кластеров не должно превышать количества блоков $e \in E(s_b)$: $m(e) \in B$ в базовом проекте:

$$|K| \le |E_B(s_b)|,$$

$$E_B(s_b) = \{e \in E(s_b) \colon m(e) \in B\},$$

$$|E_B(s_b)| = |M_b| = I_b \cdot J_b.$$

Рассматривается схема заказчика $\Pi_u = (S_u, L_u, s_{mu})$, где $s_{mu} = (\mu(s_{mu}), E(s_{mu}), N(s_{mu}), P(s_{mu}), C(s_{mu})).$

Применительно к кластерному разбиению схемы заказчика ее соединения (цепи $N(s_{mu})$) можно разделить на 3 категории относительно рассматриваемого кластера K_i :

 наружные (внешние) – это цепи, которые никак не связаны с элементами в данном кластере:

$$\begin{split} N_{ext}(K_i) &\subset N(s_{mu}), \\ \forall n \in N_{ext}(K_i); \; \forall (P(e), n) \in C(s_{mu}): e \notin K_i, \end{split}$$

внутренние – это цепи, которые связаны только с элементами в данном кластере:

$$N_{int}(K_i) \subset N(s_{mu}),$$
$$\forall n \in N_{int}(K_i); \ \forall (p,n) \in C(s_{mu}): p \in P(e), e \in K_i,$$

 смежные – это цепи, которые связаны с элементами как внутри, так и вне кластера (возможно с внешними выводами схемы):

$$\begin{split} N_{adj}(K_i) &\subset N(s_{mu}), \\ \forall n \in N_{adj}(K_i) : \exists (p_1, n) \in C(s_{mu}), \exists (p_2, n) \in C(s_{mu}), \\ p_1 \in P(e_1) \land e_1 \in K_i \land (p_2 \in P(s_{mu}) \lor p_2 \in P(e_2) \land e_2 \notin K_i) \end{split}$$

Тогда количество внешних выводов кластера или его *степень* можно определить как количество смежных цепей (узлов):

$$d(K_i) = |N_{adj}(K_i)|.$$

Количество выходов или *исходящую степень* кластера – можно определить как количество смежных цепей (узлов), источником которых является элемент *внутри* кластера:

$$d^{-}(K_{i}) = |N^{-}(K_{i})|, N^{-}(K_{i}) \subset N_{adj}(K_{i}),$$

$$\forall n \in N^{-}(K_{i}) \exists (p, n) \in C(s_{mu}) : p \in P(e) \land e \in K_{i} \land \tau(p) = \tau_{out}$$

Аналогично количество входов или *входящую степень* кластера можно определить как количество смежных цепей (узлов), источником которых является элемент *вне* кластера или внешний вход:

$$\begin{aligned} d^{+}(K_{i}) &= |N^{+}(K_{i})|, \ N^{+}(K_{i}) \subset N_{adj}(K_{i}), \\ \forall n \in N^{+}(K_{i}) \ \exists (p,n) \in C(s_{mu}): \\ p \in P(s_{mu}) \ \land \ \tau(p) &= \tau_{inp} \lor p \in P(e) \ \land e \notin K_{i} \ \land \ \tau(p) = \tau_{out}. \end{aligned}$$

Количество входов / выходов кластера схемы заказчика должно соответствовать количеству входов / выходов блока базового кристалла:

$$d^{+}(K_{i}) \leq d^{+}(b) + d^{*}(b) = |P_{r}^{+}(b)| + |P_{r}^{*}(b)|,$$

$$d^{-}(K_{i}) \leq d^{-}(b) + d^{*}(b) = |P_{r}^{-}(b)| + |P_{r}^{*}(b)|.$$

В терминах степеней вершин коммутационного графа блочного уровня $G_B(\Pi_b) = (V_B, A_B)$ для вершин графа, соответствующих блокам логических элементов $v(e) \in V_B, m(e) \in B$, должны соблюдаться следующие ограничения:

$$d^+(K_i) \le d^+(v),$$

 $d^-(K_i) \le d^-(v).$

Количество внутренних цепей кластера ограничено количеством локальных трасс внутри блока. При этом локальные трассы внутри блока не только обеспечивают внутренние коммутации между входами и выходами логических элементов в блоке, но также могут быть использованы для последующей связи с внешними цепями за пределами блока. Поэтому их количество в значительной степени связано с количеством логических элементе:

$$|N_{loc}(b)| \approx P_{r}^{+}(s) \cdot |E_{LE}(b)| = P_{r}^{+}(s) \cdot |M_{LE}| = P_{r}^{+}(s) \cdot I_{LE} \cdot J_{LE},$$

 $|N_{loc}(b)| = \mathcal{K}_{loc} \cdot P_r^+(s) \cdot |E_{LE}(b)| = \mathcal{K}_{loc} \cdot P_r^+(s) \cdot I_{LE} \cdot J_{LE}, \ \mathcal{K}_{loc} \leq 1.$

где $K_{loc} \leq 1$ характеризует процент экономии локальных трасс за счет общих входных сигналов, подключенных к земле или питанию, незадействованных входов и самих логических элементов.

В некоторых архитектурах, наряду со входами, выходы логических элементов так же коммутируются с внешними сигналами не прямыми трассами, а через локальные трассы. В этом случае требуется более жесткий контроль на количество локальных трасс с учетом внешних соединений выходов:

$$\begin{split} |N_{loc}(b)| &\approx \left(P_r^+(s) + P_r^-(s) \right) \cdot |E_{LE}(b)| = \\ &= \left(P_r^+(s) + P_r^-(s) \right) \cdot |M_{LE}| = \left(P_r^+(s) + P_r^-(s) \right) \cdot I_{LE} \cdot J_{LE}. \\ |N_{loc}(b)| &= K_{loc} \cdot \left(P_r^+(s) + P_r^-(s) \right) \cdot |E_{LE}(b)| = \\ &= K_{loc} \cdot \left(P_r^+(s) + P_r^-(s) \right) \cdot I_{LE} \cdot J_{LE}, \ K_{loc} \le 1. \end{split}$$

С учетом специфики подключения входов и выходов логических элементов в блоке получаем следующее ограничение на количество внутренних цепей кластера *N_{int}*:

$$N_{int}(K_i) + d^+(K_i) \le |N_{loc}(b)|.$$

Или для архитектур, в которых выходы логических элементов так же коммутируются с внешними сигналами через локальные трассы:

$$N_{int}(K_i) + d^+(K_i) + d^-(K_i) \le |N_{loc}(b)|.$$

Задача размещения на блочном уровне для кластеров логических элементов сводится к выбору соответствующего элемента матрицы блоков для каждого из кластеров, т.е. к установлению отображения К \rightarrow M_b:

$$\mathcal{P}_{b}: K \to M_{b}; m_{ij} = \mathcal{P}_{b}(K_{k}).$$

Как в процессе итерационного решения задачи размещения на блочном уровне, так и по ее результатам, одним из критических факторов, обеспечивающих полноту последующей трассировки межсоединений, является соблюдение ограничений на фактическое количество цепей, пересекающих сечение (разрез) по вертикали для заданного X и фактическое количество цепей, пересекающих сечение (разрез) по горизонтали для заданного Y.

С учетом выбранного на очередном шаге итерационного размещения блоков $\mathcal{P}_b: K \to M_b; m_{ij} = \mathcal{P}_b(K_k)$ можно вычислить реальные координаты портов блоков (или центров блоков) и, следовательно, граничные координаты цепей схемы заказчика:

 $X_{min}(n), Y_{min}(n), X_{max}(n), Y_{max}(n), n \in N(s_{mu}).$

Следовательно, для заданной схемы заказчика *s_{mu}* требуемая пропускная способность сечений вычисляется по формулам:

$$\sigma_X(s_{mu}, X_S^k) = |\{n: n \in N(s_{mu}) \& X_{min}(n) < X_S^k \& X_{max}(n) \ge X_S^k\}|, \sigma_Y(s_{mu}, Y_S^k) = |\{n: n \in N(s_{mu}) \& Y_{min}(n) < Y_S^k \& Y_{max}(n) \ge Y_S^k\}|.$$

Для обеспечения 100% трассировки необходимо, чтобы требуемое для реализации схемы заказчика число цепей, пересекающих сечение, не превышало фактическую пропускную способность сечения в базовом кристалле:

$$\sigma_X(s_{mu}, X_S^k) \le \sigma_X(X_S^k),$$

$$\sigma_Y(s_{mu}, Y_S^k) \le \sigma_Y(Y_S^k),$$

В рамках прототипирования, наряду со стандартным контролем параметров схемы заказчика по отношению к параметрам базового

кристалла, решается обратная (противоположная) задача - задача определения ключевых характеристик конструкции базового кристалла, необходимых для эффективной реализации некоторого набора характерных схем заказчиков. Пусть задан некоторый набор схем заказчиков

$$S_{mu}^{1}, S_{mu}^{2}, \dots, S_{mu}^{N_{su}},$$

где $S_{mu}^{k} = \left(\mu(s_{mu}^{k}), E(s_{mu}^{k}), N(s_{mu}^{k}), P(s_{mu}^{k}), C(s_{mu}^{k})\right), k = 1, \dots, N_{su}.$

Тогда, первоначально должна быть решена задача кластеризации и размещения для каждой из схем заказчиков. Существенное отличие задач кластеризации и размещения в данной постановке заключается в том, что решение должно быть оптимальным с точки зрения реализации самой схемы, безотносительно параметров базового кристалла и без учета связей с периферийным окружением. Далее, аккумулируя ограничения для отдельно взятых схем заказчиков, получим следующие ограничения для параметров базового кристалла, соблюдение которых необходимо для реализации заданного набора схем заказчиков:

1) для общего количества логических элементов базового кристалла:

$$|E_{LE}(s_{f})| \ge \max_{k} |E_{LE}(s_{mu}^{k})|,$$
$$|E_{LE}(s_{f})| = |M_{f}| = I_{f} \cdot J_{f} = |E_{B}(s_{b})| \cdot |E_{LE}(b)| = I_{b} \cdot J_{b} \cdot I_{LE} \cdot J_{LE}.$$

2) для размеров блока базового кристалла:

$$|E_{LE}(b)| \ge \max_{k} \left(\max_{i} |K_{i}(s_{mu}^{k})| \right),$$
$$|E_{LE}(b)| = |M_{LE}| = I_{LE} \cdot J_{LE}.$$

3) для количества блоков базового кристалла:

$$|E_B(s_b)| \ge \max_k |K(s_{mu}^k)|,$$
$$|E_B(s_b)| = |M_b| = I_b \cdot J_b.$$

4) для количества трассировочных выходов блока базового кристалла:

$$d^{+}(b) + d^{*}(b) \ge \max_{k} \left(\max_{i} \left(d^{+} \left(K_{i}(s_{mu}^{k}) \right) \right) \right),$$
$$d^{+}(b) + d^{*}(b) = |P_{r}^{+}(b)| + |P_{r}^{*}(b)|.$$

5) для количества трассировочных входов блока базового кристалла:

$$d^{-}(b) + d^{*}(b) \ge \max_{k} \left(\max_{i} \left(d^{-} \left(K_{i}(s_{mu}^{k}) \right) \right) \right),$$
$$d^{-}(b) + d^{*}(b) = |P_{r}^{-}(b)| + |P_{r}^{*}(b)|.$$

6) для степеней вершин коммутационного графа блочного уровня $G_B(\Pi_b) = (V_B, A_B)$, соответствующих блокам логических элементов $v(e) \in V_B, m(e) \in B$, без учета результатов размещения кластеров по конкретным блокам базового кристалла:

$$\min_{v} d^{+}(v) \geq \max_{k} \left(\max_{i} \left(d^{+} \left(K_{i}(s_{mu}^{k}) \right) \right) \right),$$
$$\min_{v} d^{-}(v) \geq \max_{k} \left(\max_{i} \left(d^{-} \left(K_{i}(s_{mu}^{k}) \right) \right) \right).$$

7) для степеней вершин коммутационного графа блочного уровня $G_B(\Pi_b) = (V_B, A_B)$, соответствующих блокам логических элементов $v(e) \in V_B, m(e) \in B$, с учетом результатов размещения кластеров по конкретным блокам базового кристалла, для кластеров разных схем, размещенных в одну и туже позицию $e_i = \mathcal{P}_b(K_i)$:

$$d^{+}(v(e_{i})) \geq \max_{k} \left(d^{+} \left(K_{i}(s_{mu}^{k}) \right) \right),$$
$$d^{-}(v(e_{i})) \geq \max_{k} \left(d^{-} \left(K_{i}(s_{mu}^{k}) \right) \right).$$

8) для количества локальных трасс в блоке базового кристалла (LAB):

$$|N_{loc}(b)| \ge \max_{k} \left(\max_{i} \left(N_{int} \left(K_{i}(s_{mu}^{k}) \right) + d^{+} \left(K_{i}(s_{mu}^{k}) \right) \right) \right).$$

9) для пропускной способности сечений в базовом кристалле:

$$\sigma_X(X_S^k) \ge \max_k (\sigma_X(s_{mu}^k, X_S^k)),$$

$$\sigma_Y(Y_S^k) \ge \max_k (\sigma_Y(s_{mu}^k, Y_S^k)).$$
3.6 Теоретико-множественный подход к решению задачи трассировки для конфигурируемых гетерогенных схем в терминах обобщенной математической модели

Сменяемость технологических конструкторских решений, И необходимость подстройки конфигурируемых гетерогенных схем специального (целевого) назначения под конкретные приложения заказчика, САПР необходимость оперативной настройки на соответствующие изменения в конструкции, схемотехнике и топологии влекут за собой разнообразие форм и способов программирования трассировочных ресурсов для реализации цепей коммутаций проекта заказчика.

Стандартные методы разработки ПЛИС предполагают использования унифицированных трассировочных элементов, таких, как проходные ключи или ключи с буферами на выходе. Наличие в гетерогенных схемах специализированных макроблоков, сложно-функциональных блоков, блоков памяти, блоков логических элементов и др. приводит к необходимости применения различных вариантов трассировочных ресурсов, куда на ряду с проходными ключами и буферами могут входить более сложные элементы, реализующие логические функции инвертора, мультиплексора, буфера с третьим состоянием и др. Тем самым, стирается граница между этапами логического синтеза и этапом топологического синтеза в классическом понимании этих процессов. В данном случае сам этап топологического синтеза включает по необходимости решение задач логического ресинтеза на этапе трассировки или пост-трассировки.

Все это требует проработки новых математических моделей для решения задач трассировки с элементами логического ресинтеза в новом понимании.

В соответствии с ранее веденной терминологией для отображения цепей и коммутаций проекта заказчика применяются элементы подсхем базового проекта $e: m(e) \in L_{Ro}$. Также, как и схемы библиотечного уровня

 $s \in L_{LE} \cup L_{IO} \cup L_M = L \setminus \{L_{BB} \cup L_{RO}\}$ элементы трассировки могут программироваться. Принципиальное отличие состоит в том, что библиотечные элементы могут быть запрограммированы и характеризованы однократно для различных вариантов схем заказчиков. В тоже время трассировочные элементы программируются индивидуально под конкретный набор коммутаций проекта заказчика:

$$C^{*-1}(s_{mu}) = \left\{ N(s_{mu}) \to \left(\bigcup_{i=1,\dots,|E(s_{mu})|} P(e_i) \right) \right\},\$$

$$\forall n \in N(s_{mu}) \colon |\{(p,n) \colon (p,n) \in C(s_{mu})\}| \ge 2.$$

Предполагается, что к моменту старта процедуры трассировки решена задача размещения, т.е. установлено отображение, в котором каждому элементу схемы заказчика поставлен в соответствие элемент базового проекта:

$$E(s_{mu}) \rightarrow \{e : e \in E(s_f), m(e) \in \{L_{LE} \cup L_M \cup L_{IO}\}\}.$$

При этом элементы $s_u \in L_u$, $s_u = (\mu(s_u), \emptyset, \emptyset, P(s_u), \emptyset)$ библиотеки L_u проекта $\Pi_u = (S_u, L_u, s_{mu})$ реализованы путем установки следующих соответствий для выводов библиотечных схем базового кристалла $P(s) = \{p_i, i = 1, ..., |P(s)|\}, s \in L_{LE} \cup L_{IO} \cup L_M$:

$$P_r(s) \to P(s_u) \cup \{P_0, P_1, P_z\},\$$

где *P*₀, *P*₁, *P*_z - условные обозначения вводов, предполагающие внешние соединения соответственно с узлом земли, питания или висячим узлом.

При этом, обратное отображение $P(s_u) \rightarrow P_r(s)$ не обязательно является однозначным, и может отражать различные варианты внешних подключений.

Следовательно, не только каждому элементу схемы заказчика поставлен в соответствие элемент базового проекта, но и каждому контакту (выводу) библиотечного элемента проекта заказчика поставлены в соответствие контакты (выводы) $P(e_i)$ библиотечных элементов базового проекта $e_i \in E(s_f)$, и следовательно, узлы базового проекта:

$$C^{*-1}(s_{mu}) = \left\{ N(s_{mu}) \to \left(\bigcup_{i=1,\dots,|E(s_f)|} P(e_i) \right) \right\}.$$

Таким образом, задача трассировки для реализации проекта заказчика сводится к программированию проводящих путей, состоящих из трассировочных элементов, между выводами элементов и узлами базового проекта.

По аналогии со схемами библиотечного уровня множество внешних выводов трассировочных элементов

$$P(s) = \{p_i, i = 1, \dots, |P(s)|\}, s \in L_{Ro}.$$

можно разделить на 2 независимых подмножества по функциональному назначению:

$$P(s) = P_r(s) \cup P_m(s),$$

где $P_r(s)$ - подмножество сигнальных или трассировочных выводов для соединения трассировочных элементов между собой и с контактами (выводами) библиотечных элементов базового проекта;

P_m(*s*) - подмножество программируемых выводов для управления проводимостью трассировочных элементов путем назначения логических элементов программируемой памяти:

$$P_m(s) \rightarrow \mathcal{B}^{|P_m(s)|}, \mathcal{B} = \{0,1\}.$$

Среди сигнальных или трассировочных выводов $p \in P_r(s)$ следует различать выводы по направлению распространения сигнала: $\tau(p) \in \{\tau_{inp}, \tau_{out}, \tau_{bi}\}$ – вход, выход или двунаправленный.

Введем следующие обозначения:

пусть $\nu(p), p \in P_r(s) \cup P_m(s)$ - логическое значение на выводе p: $\nu: P(s) \to \mathcal{B}, \mathcal{B} = \{0,1\};$

 $c_i = v(p), p \in P_m(s)$ - логическое значение управляющего сигнала, $y_i = v(p), p \in P_r(s), \tau(p) \in \{\tau_{out}, \tau_{bi}\}$ - логическое значение выхода, $x_i = v(p), p \in P_r(s), \tau(p) \in \{\tau_{inp}, \tau_{bi}\}$ - логическое значение входа, $\phi(c_i)$ - логическая функция управления открыванием трассировочного элемента, принимающая значение 1 для открытого состояния, 0 - для закрытого,

 $\chi(x_i)$ - не константная логическая функция выхода в терминах входа для открытого состояния трассировочного элемента (в унарном варианте это инверсия или эквивалентность),

 $\rho_i(c_i, y_i, x_i): \mathcal{B} \times \mathcal{B} \times \mathcal{B} \to \mathcal{B}, \mathcal{B} = \{0, 1\}$ - характеристическая функция проводимости от x_i к y_i , принимающая значение 1 при корректных значениях c_i, y_i, x_i и 0 в противном случае.

Корректными трассировочными элементами будем считать те, для которых совокупность характеристических функций проводимости может быть определена в импликативной форме (в форме конъюнкции импликаций " ⇒ ") в следующем виде:

 $\rho(\boldsymbol{c},\boldsymbol{y},\boldsymbol{x}) \equiv \bigwedge_i \ \rho_i(c_i,y_i,x_i) \equiv \bigwedge_i \ (\phi_i(c_i) \Rightarrow (y_i = \chi_i(x_i))).$

Среди различных вариантов корректных трассировочных элементов можно выделить следующие:

- проходной ключ:

 $\rho(g, s, d) \equiv \rho(g, d, s) \equiv (g \Rightarrow s = d);$ - буфер:

 $\rho(\emptyset, y, x) \equiv (1 \Rightarrow y = x);$

- инвертор:

$$\rho(\emptyset, y, x) \equiv (1 \Rightarrow y = \sim x);$$

- мультиплексор без инверсии:

$$\rho(c, y, \mathbf{x}) \equiv (\sim c \Rightarrow y = x_1) \land (c \Rightarrow y = x_2), \mathbf{x} = (x_1, x_2)^T;$$

- мультиплексор с инверсией:

$$\rho(c, y, \mathbf{x}) \equiv (\sim c \Rightarrow y = \sim x_1) \land (c \Rightarrow y = \sim x_2), \mathbf{x} = (x_1, x_2)^T;$$

- буфер с третьим состоянием

 $\rho(c, y, x) \equiv (c \Rightarrow y = \sim x);$

- резистор:

$$\rho(\emptyset, m, p) \equiv \rho(\emptyset, p, m) \equiv (1 \Rightarrow p = m).$$

Совокупность корректных трассировочных элементов базового проекта $\Pi_f(\Pi) = (S_f, L_f, s_f)$, $s_f = (\mu(s_f), E(s_f), N(s_f), P(s_f), C(s_f))$ может быть преобразована в трассировочных смешанный граф или ориентированный граф (с преобразованием каждого из двунаправленных ребер в пару дуг противоположного направления) $G_{Ro}(\Pi) = (V_{Ro}, A_{Ro})$ по следующим правилам:

Множество вершин V_{Ro} графа $G_{Ro}(\Pi)$ взаимно однозначно соответствует подмножеству таких и только таких узлов $n \in N(s_f)$, для которых существует хотя бы одно соединение с элементом из подмножества трассировочных элементов:

$$\{n: n \in N(s_f) \& \left(\left(\exists e \in E(s_f): m(e) \in L_{Ro} \right) \& (\exists p \in P(e): (p, n) \in C(s_f) \right) \\ \Leftrightarrow \{v(n): v(n) \in V_{Ro} \}.$$

В качестве замечания необходимо отметить: предполагается, что в конструкции кристалла отсутствуют такие варианты, когда выводы логических элементов, макроблоков или периферийных элементов соединены друг с другом напрямую без трассировочных элементов.

Множество дуг графа $G_{Ro}(\Pi)$ формируется из импликативных форм характеристических функций проводимости трассировочных элементов $(\phi_i(c_i) \Rightarrow (y_i = \chi_i(x_i)))$. Для каждой пары $(y_i = \chi_i(x_i))$ создается новая дуга $a_{xy} = (v(x), v(y))$ между вершинами, соответствующими узлам $x \in N(s_f)$, $y \in N(s_f)$, имеющих соединения с соответствующими выводами трассировочных элементов. При этом дуги $a_{xy} = (v(x), v(y))$ наследуют соответствующие характеристики трассировочного элемента:

 $\phi(a_{xy})$ - логической функции управления открыванием трассировочного элемента, принимающей значение 1 для открытого состояния дуги $a_{xy} = (v(x), v(y)), 0$ - для закрытого,

 $\chi(a_{xy})$ - логической функции состояния вершины, для которой дуга a_{xy} трассировочного графа является входящей, в терминах состояния

вершин, для которых дуга a_{xy} графа является исходящей, при открытом состоянии трассировочного элемента.

Таким образом, задача трассировки для реализации проекта заказчика сводится к нахождению путей в трассировочном графе $G_{Ro}(\Pi) = (V_{Ro}, A_{Ro})$, соединяющих вершины трассировочного графа в соответствии с набором коммутаций узлов схемы заказчика $n \in N(s_{mu})$:

$$C^{*-1}(s_{mu}) = \left\{ N(s_{mu}) \to \left(\bigcup_{i=1,\dots,|E(s_f)|} P(e_i) \right) \right\}.$$

При этом направление искомых путей распространения сигнала определяется типом выводов коммутируемых элементов: от выходов $p_x P(e_i), \tau(p_x) \in \{\tau_{out}, \tau_{bi}\}$ к входам $p_y P(e_i), \tau(p_y) \in \{\tau_{inp}, \tau_{bi}\}$.

Предположим, что $\tau = \{v_0, v_1, ..., v_n\}, v_i \in V_{Ro}$ - один из таких путей (маршрутов) с дугами $a_i = (v_i, v_{i+1}), i = 0, 1, ..., (n-1), a_i \in A_{Ro}$, тогда

 $\phi(\tau) = \bigwedge_{i=0}^{n-1} \phi_i(a_i)$ - логическая функция управления открыванием трассировочного пути, определяющая назначение соответствующих элементов программируемой памяти, а

 $\chi(\tau) = \chi_{n-1} \circ \chi_{n-2} \circ ... \circ \chi_0 = \chi_{n-1}(\chi_{n-2}(...(\chi_0(\nu_0)))))$ - логическая функция состояния концевой вершины пути - композиция (суперпозиция) соответствующих функций передачи сигнала для последовательности дуг.

В унарном варианте не константная логическая функция выхода в терминах входа - это отрицание (инверсия) или эквивалентность. В этом случае конечное значение $\chi(\tau)$ определяется подсчетом четности инверсных элементов в пути. При этом результирующая функция - отрицание при нечетном количестве отрицаний в пути и эквивалентность при четном.

Само по себе изменение логической функции пути с эквивалентности на любую другую требует переопределения (ресинтеза) логического элемента, соединенного с узлом схемы, соответствующим конечной вершине пути v_n по результатам трассировки.

3.7 Выводы

Разработана новая теоретико-множественная математическая модель проекта, объединяющая в себе различные стили проектирования (заказных, полузаказных - базовых матричных кристаллов и программируемых логических интегральных схем).

Предложена формализация соответствий между элементами базового конфигурируемой кристалла гетерогенной интегральной схемы И разрабатываемыми функциональность схемами заказчиков, которых требуется запрограммировать в базовом кристалле, что обеспечивает оперативную настройку конструкции и схемотехники базового кристалла на новые потребности от заказчиков и оперативную настройку САПР на соответствующие изменения в конструкции, схемотехнике и топологии базового кристалла.

Глава 4. Конструктивные и схемотехнические решения специализированных конфигурируемых интегральных схем повышенной надежности

4.1. Методы повышения отказоустойчивости и стойкости к воздействию неблагоприятных внешних воздействующих факторов конфигурируемых специализированных ИС на основе схем обнаружения неисправностей

Ввиду специфики отечественного рынка наибольший интерес в части специализации представляют микросхемы с повышенной живучестью, надежностью и радиационной стойкостью.

С уменьшением проектных норм становится возможным размещение сложных цифровых блоков и различных датчиков и детекторов на кристалле интегральной схемы, что позволяет улучшить ее характеристики. Так, например, в современных конвейерных АЦП применяются схемы цифровой калибровки, повышающие точность преобразования. Другим примером является применение цифровой компенсации в импульсных преобразователях напряжения, что позволяет оптимизировать динамические характеристики системы.

С помощью специальных схем и цифровой обработки можно решать и другие задачи помимо улучшения технических характеристик. В первую очередь это повышение живучести, надежности функционирования и радиационной стойкости.

Одним из основных параметров, определяющих возможность выполнения целевых функций радиоэлектронной системой и аппаратурой в условиях неблагоприятного внешнего воздействия (HBB), является живучесть. В известной литературе исследованы вопросы повышения живучести сложных технических систем в целом [86] – [89]. Однако

известные методы повышения живучести сложных технических систем не могут быть в полной мере применены к электронной компонентной базе.

В данной работе исследуются способы повышения живучести конфигурируемых интегральных схем. Введен новый термин, описывающий интегральной (допустимого) работу схемы режим приемлемого _ функционирования микросхемы, который определяется расширенными нормами параметров. Конфигурируемая интегральная схема рассматривается качестве конструктивной единицы системы, живучесть которой В необходимо обеспечить. В режиме приемлемого функционирования в неблагоприятного воздействия условиях параметры микросхемы отклоняются от характеристик в рабочем режиме на допустимую величину, что позволяет избежать катастрофического сценария работы аппаратуры и системы в целом.

С целью повышения живучести, надежности и радиационной стойкости предлагается в специализированную интегральную схему встраивать дополнительные узлы и СФ блоки детектирования и устранения отрицательных последствий неблагоприятных воздействий – неисправностей (отказов и сбоев) [90].

В режиме работы приемлемого функционирования при обнаружении неблагоприятного воздействия в микросхеме происходит активизация опорных элементов повышенной надёжности, запуск автокалибровки, включение схем резервирования, изменение в работе схем коррекции ошибок. При этом детектирование неисправности сопровождается сигнализацией обнаружения отказа или сбоя с определением вероятности ошибки (рисунок 4.1).

Требуемые параметры живучести, надежности и радиационной стойкости интегральных схем достигаются за счёт: совершенствования технологических процессов, применения традиционных конструктивных, топологических и схемотехнических методов (резервирования, помехоустойчивого кодирования и т.д.). Использование сложных алгоритмов

и схем самоконтроля, с помощью которых детектируются неисправности, в современных ИС уровня 180 нм и менее может дополнить традиционные приёмы.

Повышение стойкости к неблагоприятному внешнему воздействию конфигурируемых схем происходит за счет применения встроенных схем самоконтроля и алгоритмов, при реализации которых аппаратура и блоки интегральных схем работают в едином конструктивно-алгоритмическом пространстве.



Рисунок 4.1 - Блоки детектирования и устранения неисправностей

Для того, чтобы конфигурируемая интегральная схема могла работать в режиме приемлемого функционирования в условиях неблагоприятного воздействия, требуется:

- наличие документации, регламентирующей работу микросхемы в этом режиме (раздел в ТУ, КД, СИ и др.);
- наличие возможности конфигурирования микросхемы с настройкой на характеристики внешнего воздействия и изменения режимов работы;
- наличие интерфейса микросхема аппаратура для передачи информации о неблагоприятном воздействии и о переключении режимов работы;

- наличие необходимого количества датчиков, детектирующих неблагоприятное воздействие;
- наличие встроенных в интегральную схему средств обработки информации;
- наличие средств повышения живучести, надежности функционирования и сбоеустойчивости на уровне аппаратуры.

Определим, какое поведение конфигурируемой микросхемы в условиях НВВ можно считать приемлемым. Во-первых, не должно происходить катастрофического отказа микросхемы при отклонениях от нормы внешних сигналов и питающих напряжений. Во-вторых, если микросхема уже не в состоянии работать в штатном режиме, ее параметры не должны значительно To типовых значений. есть, важнейшим отклоняться OT является гарантированное функционирование микросхемы в рамках расширенных норм параметров. Например, если АЦП в условиях НВВ не в состоянии обеспечить 12 – разрядную точность, то, по крайней мере, не должна выходить за нормы 10 (или 8) – разрядной точности. Микросхема должна также выдавать сигналы об обнаружении НВВ, а также о возможных ошибках (рисунок 4.2).



Рисунок 4.2 - Приемлемое поведение микросхемы в условиях НВВ

Приемлемое функционирование предполагают: обнаружение HBB, сигнализацию обнаружения HBB, отсутствие генерации, работоспособность при предельных режимах внешних сигналов, включая питание и переход микросхем из рабочего режима в режим работы в условиях HBB.

В режиме работы в условиях НВВ происходит активизация опорных элементов повышенной надёжности, стабилизация питания, сигнализация ошибки (вероятности ошибки) в выходных сигналах, использование резервирования, использование корректирующих кодов, запуск режимов восстановления памяти и калибровки. При этом микросхема функционирует в рамках расширенных норм параметров.

Возможная структура микросхемы с приемлемым функционированием приведена на рисунке 4.3. Система с повышенной живучестью помимо применения описанных выше микросхем должна содержать датчики обнаружения HBB и схему управления, выполненную на интегральных схемах малой степени интеграции, которые обладают повышенной устойчивостью к HBB.



Рисунок 4.3 - Структура конфигурируемой микросхемы повышенной живучести

Предложенный подход хорошо согласуется с развитием архитектуры и алгоритмов работы современных конфигурируемых интегральных схем. Заказчик конфигурирует микросхему на выполнение основных целевых функций с учетом возможности возникновения неблагоприятного воздействия, при котором происходит переключение работы микросхемы в допустимый режим.

Наличие в микросхеме элементов обнаружения неисправностей (отказов и сбоев) позволяет для их устранения использовать предложенный в работе метод мягких решений. Термин «мягкие решения» в теории помехоустойчивого кодирования означает, что решение об ошибках принимается не на основе «жёстких» алгоритмов, а с использованием дополнительной информации о вероятности ошибки.

Известно, что для уменьшения вероятности ошибки в системе с помехоустойчивым кодированием детектор и декодер должны работать совместно, используя всю информацию с демодулятора сигнала [91]. В отличие от жесткого решения, при котором сигнал на выходе детектора является элементом того же поля, что и сигнал на входе модулятора, мягкое решение (термин введен в работе) основано на применении информации о вероятности ошибки. В работе рассмотрены алгоритмы мягкого решения и границы их применения в схеме, работающей в режиме приемлемого функционирования, предложены новые методы декодирования, определены границы вероятности того, что сообщение не будет декодировано правильно.

В работе предложен метод повышения надежности схем хранения информации с использованием мягких решений, в том числе, предложены новые алгоритмы исправления ошибок в блоках конфигурационной и пользовательской памяти, в которых детекторы неисправности, получают дополнительную информацию, обрабатывают ее и передает в декодер для улучшения его характеристик. Таким образом, обеспечивается эффективная совместная работа накопителя и периферийных схем блоков памяти с декодерами, исправляющими ошибки.

Области принятия правильного решения в режиме приемлемого функционирования в блоках коррекции ошибок приведены на рисунке 4.4 (на рисунке области приведены для кода с минимальным расстоянием Хэмминга равным пяти).



Рисунок 4.4 - Области приемлемого функционирования блока, использующего код с минимальным расстоянием Хэмминга равным пяти

4.2. Методы повышения надежности конфигурационной и пользовательской памяти на основе помехоустойчивого кодирования с применение мягких решений

4.2.1. Актуальность применения корректирующих кодов для повышения надежности функционирования блоков памяти

Развитие конфигурируемых интегральных схем в значительной мере определяется возможностью реализации конфигурационной и пользовательской памяти, требуемой емкости, и обеспечения необходимых эксплуатационных характеристик, к которым в первую очередь следует отнести характеристики надежности. Совершенствование памяти идет в направлении увеличения степени интеграции и достигается уменьшением размеров элементов и более плотной их компоновкой, что приводит к появлению большого количества случайных дефектов при производстве и относительно высокой интенсивности отказов и сбоев в процессе эксплуатации.

С блоков целью повышения надежности памяти применяют специальные схемотехнику, топологию и технологию [92]. Такие методы во многих случаях оказываются не эффективными, так как ограничены существующим уровнем научно-технического прогресса. Поэтому для решения указанной проблемы используются структурно-логические методы блоки повышения надежности, основанные на введении В памяти избыточных элементов, позволяющих нейтрализовать дефекты производства, отказы и сбои, появляющиеся в процессе эксплуатации.

Использование корректирующих кодов для увеличения надежности блоков памяти ограничено жесткими рамками, накладываемыми на временную и аппаратурную избыточность. С другой стороны, для повышения надежности памяти требуется увеличение корректирующей способности. Эти требования не всегда удается удачно совместить. Хорошие характеристики устройств хранения информации можно получить при эффективной совместной работе основных узлов памяти с декодерами, исправляющими ошибки, что достигается использованием оптимальных алгоритмов исправления ошибок и устройств, реализующих эти алгоритмы.

В работах ряда авторов создание надежных запоминающих устройств рассматривается как задача применения известных кодов и методов декодирования в стандартных микросхемах и блоках памяти [93] - [97].

В настоящем разделе описан метод декодирования блочных кодов, применение которого в конфигурационной и блочной памяти конфигурируемых интегральных схем позволяет повысить ее помехоустойчивость.

4.2.2. Границы областей декодирования q-ичных блоковых кодов по алгоритму Форни

Пусть элемент x_i входной последовательности X является q -ичным символом x_j , $1 \le j \le q$, причем q - степень простого числа, так что значения символов можно рассматривать как элементы поля GF(q) [98]. Элемент информации на выходе содержит две составляющие – решение \hat{x} , совпадающее с одним из q символов, и вес α , $\alpha \ge 0$, указывающий надежность этого решения. Принимаемое слово $\overline{\alpha}$ представляет собой последовательность из n решений \hat{x}_j и соответствующих им весов α_i .

Положим

$$S(\hat{x}, x) = \begin{cases} +1, x = \hat{x} \\ -1, x \neq \hat{x} \end{cases}$$

и определим, как и в [100], скалярное произведение

$$\overline{\alpha} \cdot \overline{\mathbf{X}} = \sum_{i=1}^{n} \alpha_i \, S(\hat{x}_i, x_i).$$

Выберем из скалярного произведения $\overline{\alpha} \cdot \overline{X}$ все возможные суммы, состоящие из d слагаемых $\alpha_i S(\hat{x}_i, x_i)$. Таких сумм C_n^d .

Пусть для некоторых $\overline{\alpha}$ и \overline{X} выполняются C_n^d условий:

$$\begin{cases} \sum_{i \in Q_{d,1}} \alpha_i S(\hat{x}_i, x_i) > 0; \\ \sum_{i \in Q_{d,C_n^d}} \alpha_i S(\hat{x}_i, x_i) > 0; \end{cases}$$
(3)

где $Q_{d,1} \div Q_{d,C_n^d}$ –все возможные наборы d позиций из n позиций.

<u>Теорема 1</u>. Код длиной n и с минимальным расстоянием d содержит не более одного кодового слова \overline{X} , удовлетворяющего условиям (3), при $\alpha_i > 0$.

<u>Доказательство</u>. Пусть соотношения (3) выполняются для некоторого кодового слова \overline{X} и пусть слова \overline{X}' - любое кодовое слово, находящееся от слова \overline{X} на расстоянии d или больше. Пусть наборы позиций T и U определяются соотношениями

$$x_i \neq x'_i, \hat{x}_i = x'_i, i \in T,$$

$$x_i \neq x'_i, \hat{x}_i \neq x'_i, i \in T,$$

$$|T| + |U| = d.$$

Тогда для некоторого ј

$$\sum_{i \in Q_j} \alpha_i S(\hat{x}_i, x_i) = \sum_{i \in T} \alpha_i - \sum_{i \in U} \alpha_i =$$
$$= A_T - A_U > 0;$$
$$\sum_{i \in Q_j} \alpha_i S(\hat{x}_i, x'_i) = \sum_{i \in T} \alpha_i S(\hat{x}_i, x'_i) +$$
$$+ \sum_{i \in U} \alpha_i S(\hat{x}_i, x'_i) = -A_T + B_U.$$

Здесь $B_U \leq A_U$

Таким образом, так как

$$A_T - A_U > 0,$$

$$-A_T + B_U < 0,$$

что и требовалось доказать.

<u>Следствие</u>. Если для некоторого кодового слова \overline{X} выполняются условия (3), то \overline{X} находится с помощью алгоритма декодирования Форни.

4.2.3. Алгоритм мягкого декодирования блоковых кодов, обеспечивающий большую вероятность правильного декодирования, чем алгоритм Форни

Описание алгоритма.

Метод заключается в многократном применении алгоритма Форни. Кратность таких попыток определяется числом K, $0 \le K \le n -d$. При K= 0 декодирование производится по алгоритму Форни. По мере увеличения K от 0 до n-d. вероятность правильного декодирования увеличивается (точнее не уменьшается) вплоть до значения, получаемого при декодировании по критерию максимума правдоподобия. Это утверждение справедливо, если в качестве детектора используется детектор правдоподобия. Например, для (3,2) - кода над GF(2) характеристики, получаемые при декодировании по критерию максимума правдоподобия, достигаются однократным применением алгоритма Форни.

Для реализации метода необходимо каждый шаг декодирования по алгоритму Форни провести q^{K} раз, меняя на первом шаге значения К символов, следующих за d-l стираемыми символами, на втором шаге значения К символов, следующих за d-3 стираемыми символами и т.д., с тем, чтобы на каждом шаге перебрать все возможные значения изменяемых символов. Из совокупности полученных кодовых слов выбирается слово X, имеющее максимальное скалярное произведение αX .

Области декодирования.

Введем следующие обозначения. Пусть решение \hat{X} отличается от кодового слова X в r позициях и от кодового слова X_j в m позициях, причем среди r и m позиций l - общие.

Тогда слово $\varphi(\alpha, X, X_j)$ представляет собой последовательность из *n* решений \hat{x}_i и *n* соответствующих им весов φ_i , которые в *l* позициях равны 0, а в остальных позициях совпадают с α_i . Обозначим через Φ_{ij} набор позиций, в которых кодовое слово X отличается от кодового слова X_{j} , находящегося от слова X на расстоянии *i*.

Предположим, что для некоторых α и X выполняются следующие условия:

$$\begin{cases} \sum_{i \in \Phi_{d+k,1}} \varphi_i S(\hat{x}_i, x_i) > 0, \\ \vdots \\ \sum_{i \in \Phi_{d+k, z_{d+k}}} \varphi_i S(\hat{x}_i, x_i) > 0, (4) \\ \sum_{i \in Q_{d+K, 1}} \alpha_i S(\hat{x}_i, x_i) > 0, \\ \vdots \\ \sum_{i \in Q_{d+K, 1}} \alpha_i S(\hat{x}_i, x_i) > 0, (5) \end{cases}$$

которые справедливы для всех $0 \le k \le K$, где $0 \le K \le n-d$, z_{d+k} - число кодовых слов, находящихся от слова X на расстоянии d+k.

<u>Теорема 2</u>. Код с длиной *n* и минимальным расстоянием *d* содержит не более одного кодового слова *X*, удовлетворяющего условиям (4, 5), при любом заданном $0 \le K \le n$ -*d*, и $\alpha_i \ge 0$.

<u>Доказательство</u>. Пусть соотношения (4, 5) выполняются для некоторого кодового слова X и пусть X_{d+k} - любое кодовое слово, находящееся от слова X на расстоянии d+k, $0 \le k \le n-d$. Пусть наборы позиций T_{d+k} и U_{d+k} определяется соотношениями

$$\begin{aligned} x_i &\neq x_{d+k,i}, \quad \hat{x}_i = x_i, \ i \in T_{d+k}; \\ x_i &\neq x_{d+k,i}, \quad \hat{x}_i = x_i, \ i \in U_{d+k}; \\ |T_{d+k}| &+ |U_{d+k}| = d + k, \ k < K; \\ |T_{d+k}| &+ |U_{d+k}| = d + K, \ k \ge K. \end{aligned}$$

Тогда для случая k<K

$$\sum_{i \in \Phi_{d+K,j}} \varphi_i(\alpha_i, x_i, x_{ji}) \cdot S(\hat{x}_i, x_i) =$$

$$= \sum_{i \in T_{d+k}} \varphi_i(\alpha_i, x_i, x_{ji}) - \sum_{i \in U_{d+k}} \varphi_i(\alpha_i, x_i, x_{ji}) = A_{T_{d+k}} - A_{U_{d+k}} > 0;$$
$$\sum_{i \in \Phi_{d+k,j}} \varphi_i(\alpha_i, x_{ji}, x_i) \cdot S(\hat{x}_i, x_{ji}) = A_{T_{d+k}} + B_{U_{d+k}}.$$

Здесь и $B_{U_{d+K}} = A_{U_{d+K}}$ и

$$\varphi_i(\alpha_i, x_i, x_{ji}) = \varphi_i(\alpha_i, x_{ji}, x_i).$$

Таким образом, так как

$$A_{T_{d+K}} - A_{U_{d+K}} > 0$$
, to $-A_{T_{d+K}} + B_{U_{d+K}} < 0$.

Для k≥К

$$\sum_{i \in Q_{d+K,j}} \alpha_i S(\hat{x}_i, x_i) + \sum_{i \in T_{d+K}} \alpha_i - \sum_{i \in U_{d+K}} \alpha_i = A_{T_{d+K}} - A_{U_{d+K}} > 0,$$
$$\sum_{i \in Q_{d+K,j}} \alpha_i S(\hat{x}_i, x_{ji}) = -A_{T_{d+K}} - B_{U_{d+K}} \le 0,$$

что и требовалось доказать.

<u>Следствие</u>. Если для некоторого кодового слова X и заданного $0 \le K \le$ *n-d* выполняются условия (5, 4), то X находится с помощью описанного выше алгоритма.

Варианты основного алгоритма декодирования.

Описанный алгоритм декодирования можно условно разбить на K+1 этапов. На первом этапе K = 0 и декодирование производится по алгоритму Форни. На втором этапе изменяются значения d - го, d-2 - го и т.д. символов и производится исправление ошибок к соответственно d-1, d-3 и т.д. стираний.

Области декодирования для каждого этапа описанного алгоритма не меньшие, чем области декодирования для предыдущих этапов. Если для α и X выполняются условия (5) при некоторых K=l, то эти условия выполняются также и при K = l+1. Это можно показать, например, следующим образом.

Пусть условия (5) выполняются при некотором К, но существует набор Q_{d+K+1, j} такой, что

$$\sum_{i \in Q_{d+K+1,j}} \alpha_i S(\hat{x}_i, x_i) \le 0.$$
(6)

Тогда для каждого слагаемого в левой части выражения (В) справедливо неравенство

$$\alpha_i S(\hat{x}_i, x_i) \leq -\sum_{i \in Q_{d+K}} \alpha_i S(\hat{x}_i, x_i) < 0.$$

Так как $\alpha_i \ge 0$, то это равносильно тому, что для всех $i \in Q_{K+1+d}$ $\hat{x}_i \ne x_i$. Но в этом случае

$$\sum_{i\in Q_{d+K}}\alpha_i S(\hat{x}_i, x_i) \le 0,$$

что противоречит первоначальному предположению.

Выбор искомого кодового слова из совокупности кодовых слов, полученных на различных этапах декодирования, производится по максимальному скалярному произведению. Обоснование этому дает теорема 3.

<u>Теорема 3.</u> Если для некоторого α и кодового слова X выполняются условия (4) и (5), то скалярное произведение $\alpha \cdot X$ больше скалярного произведения α с любым другим кодовым словом.

<u>Доказательство</u>. Пусть X₁ - любое кодовое слово, находящееся от слова X на расстоянии k. Пусть наборы позиций S, T, L и U определяются соотношениями

$$\begin{aligned} x_i &= x_{1i}, & i \in S, \\ x_i &\neq x_{1i}, & \hat{x}_i &= x_i, & i \in T, \\ x_i &\neq x_{1i}, & \hat{x}_i &= x_{1i}, & i \in L, \\ x_i &\neq x_{1i}, & \hat{x}_i &\neq x_i, & \hat{x}_i &\neq x_{1i}, & i \in U. \end{aligned}$$

Тогда

$$\alpha \cdot X = \sum_{i \in S} \alpha_i S(\hat{x}_i, x_i) + \sum_{i \in T} \alpha_i - \sum_{i \in L} \alpha_i - \sum_{i \in U} \alpha_i = A_S + A_T - A_L - A_U,$$

$$\alpha \cdot X_1 = \sum_{i \in S} \alpha_i S(\hat{x}_i, x_{1i}) - \sum_{i \in T} \alpha_i + \sum_{i \in L} \alpha_i - \sum_{i \in U} \alpha_i = A_S - A_T + A_L - A_U,$$

Если k < K, то A_T - $A_L > 0$ и поэтому

$$\alpha \cdot X > \alpha \cdot X_1,$$

если k = K, то $A_T - A_L - A_U > 0$, $-A_T + A_L < -A_U \le 0$ и поэтому $\alpha \cdot X > \alpha \cdot X_1$.

Если k > K, то $A_T - A_L - A_U > 0$.

Это несложно показать, используя тот факт, что все возможные суммы из K слагаемых среди k - положительны. Поэтому также $\alpha \cdot X > \alpha \cdot X_1$ что и требовалось доказать.

<u>Теорема 4</u>. Если т. α не образует одинаковое максимальное скалярное произведение ни с какими двумя кодовыми словами, то найдется кодовое слово X, для которого выполняются соотношения (4) и (5) при некотором $K \leq n - d$.

<u>Доказательство</u>. Выберем произвольно три кодовых слова X, X₁, X₂. Пусть наборы позиций B, P, R определяются соотношениями

$$\begin{aligned} x_i &\neq x_{1i}, \quad i \in B, \\ x_i &\neq x_{2i}, \quad i \in P, \\ x_{1i} &\neq x_{2i}, \quad i \in R. \end{aligned}$$

Пусть также для α и кодового слова Х

$$\begin{cases} \sum_{i \in B} \varphi_i S(\hat{x}_i, x_i) = A_1 > 0, \\ \sum_{i \in P} \varphi_i S(\hat{x}_i, x_i) = -A_2 < 0, \end{cases}$$
(7)

и для α и кодового слова X₂

$$\begin{cases} \sum_{i \in P} \varphi_i S(\hat{x}_i, x_{2i}) = A_2 > 0, \\ \sum_{i \in R} \varphi_i S(\hat{x}_i, x_{2i}) = A_3 < 0. \end{cases}$$
(8)

Покажем, что условия (7) и (8) не могут одновременно выполняться. Действительно,

$$A_3 = A_1 + A_2 > 0,$$

что противоречит условию (8).

Пусть в коде G - N кодовых слов $X_1, X_2, ..., X_N$. Предположим, что не существует кодового слова, для которого выполняются условия (4). Выберем из совокупности кодовых слов слово X_j , среди соотношений (4) у которого больше всего соотношений со знаком "больше". Пусть таких соотношений l. Заметим, что соотношения (4) не могут иметь одновременно знаки "больше" и "равно", не имея ни в одном соотношении знак "меньше", так как в этом случае несколько кодовых слов имеют с α одинаковое максимальное скалярное произведение, что противоречит условию теоремы.

Выберем из совокупности соотношений (4) для кодового слова Хј соотношение со знаком "меньше":

$$\sum_{i\in W}\varphi_i S(\hat{x}_i, x_{ji}) < 0.$$

Тогда для некоторого слова X_{m} , для которого $i \in W$, если $x_{ii} \neq x_{mi}$:

$$\sum_{i\in W}\varphi_i S(\hat{x}_i, x_{mi}) > 0.$$

Кроме того, если

$$\sum_{i\in B}\varphi_i S(\hat{x}_i, x_{ji}) > 0,$$

где $i \in B$, если $x_{ji} \neq x_{ri}$, то

$$\sum_{i\in B}\varphi_i S(\hat{x}_i, x_{mi}) > 0,$$

где і $\in R$ при $x_{mi} \neq x_{ri}$,

так как в противном случае одновременно выполняются условия (7), (8).

Таким образом для кодового слова X_m *l*+*l* соотношений имеют знак "больше", что противоречит первоначальное предположению.

Теоремы 3 и 4 показывают, что в общем случае при K=n-d и использовании в качестве детектора - детектора правдоподобия декодирование соответствует декодированию по критерию максимума правдоподобия. В детекторе правдоподобия информация не теряется. В двоичном случае на сигналы α не накладывается ограничений, поэтому детектор неисправности, выдающий сигналы α , в двоичном случае можно рассматривать как детектор правдоподобия.

Ясно, что декодирование по описанное алгоритму можно прекратить на определенном этапе, получая такие же результаты, как и при осуществлении всех К+1этапов декодирования. Для этого на каждом этапе декодирования необходимо проверять выполнение условий (4,5). Для уменьшения числа вычислений можно проверять выполнение только тех условий, в число слагаемых в левой части которых входит хотя бы одно слагаемое со знаком меньше или равно (в соответствующих позициях рассматриваемое кодовое слово отличается от принятого решения).

4.2.4. Границы вероятности того, что сообщение не будет декодировано правильно

По методу Чернова [100] – [101] можно получить границу вероятности того, что суммы случайных величин не превосходят заданных значений. Воспользуемся этим методом для нахождения границ вероятности того, что сообщение не будет декодировано правильно.

Предположим, что кодовые слова равновероятны, причем вероятность равенства символа x_i - одному из q значений для всех элементов равна 1 / q и не зависит от значений символа на остальных позициях.

Средняя вероятность P₁ того, что сообщение не будет декодировано правильно с помощью алгоритма Форни, определяется как средняя вероятность выполнения условий (3) [101].

Введем определение

$$f_1(X,Y) = \begin{cases} 0 \begin{cases} \sum_{i \in Q_{d,1}} \alpha_i S(\hat{x}_i, x_i) > 0, \\ \vdots \\ \sum_{i \in Q_{d,C_n^d}} \alpha_i S(\hat{x}_i, x_i) > 0, \\ 1, в \text{ остальных случаях.} \end{cases}$$

Тогда

$$f_{1}(X,Y) \leq \left[\exp\left(-S \cdot \sum_{i \in Q_{d,1}} \alpha_{i} S(\hat{x}_{i}, x_{i})\right) + \dots + \exp\left(-S \cdot \sum_{i \in Q_{d,C_{n}^{d}}} \alpha_{i} S(\hat{x}_{i}, x_{i})\right) \right],$$

$$P_{1} = \overline{f_{1}(X,Y)}$$

$$\leq \left[\exp\left(-S \cdot \sum_{i \in Q_{d,1}} \alpha_{i} S(\hat{x}_{i}, x_{i})\right) + \dots + \exp\left(-S \cdot \sum_{i \in Q_{d,C_{n}^{d}}} \alpha_{i} S(\hat{x}_{i}, x_{i})\right) \right],$$

$$= C_{n}^{d} \exp\left(-S \sum_{i=1}^{d} \alpha_{i} S(\hat{x}_{i}, x_{i})\right) = C_{n}^{d} [g(-S)]^{d},$$

где g(-S) - производящая функция моментов случайных величин $\alpha_i S(\hat{x}_i, x_i)$.

Средняя вероятность P₂ того, что сообщение не будет декодировано правильно с помощью предложенного алгоритма, определяется как средняя вероятность выполнения условий (4) и (5).

Введем определение

$$f_{2}(X,Y) = \begin{cases} \left\{ \begin{array}{l} \sum_{i \in \Phi_{d+k,1}} \varphi_{i}S(\hat{x}_{i},x_{i}) > 0, \\ \vdots \\ \sum_{i \in \Phi_{d+k,2}} \varphi_{i}S(\hat{x}_{i},x_{i}) > 0, \\ \sum_{i \in Q_{d+K,2}} \alpha_{i}S(\hat{x}_{i},x_{i}) > 0, \\ \sum_{i \in Q_{d+K,2}} \alpha_{i}S(\hat{x}_{i},x_{i}) > 0, \\ 1, \text{ в остальных случаях.} \end{cases} \end{cases} \end{cases}$$

Тогда

$$P_2 = f_2(X, Y) \le \sum_{j=0}^{K-1} z_{d+j} [g(-S)]^{d+j} + C_n^{d+K} [g(-S)]^{d+K}.$$

В работах [100], [102] определены границы возможного обмена между вероятностями стирания и ошибки при использовании наилучшего кода. Используя то же доказательство, можно определить границы возможного обмена между вероятностями стирания и ошибки при получении информации о стирании с детектора неисправности.

Если оказывается, что для всех $X \neq X_1$: $P(Y|X_1) > e^{\beta_0} P(Y|X)$, где β_0 любое неотрицательное число, то детектор выдает решение X_1 . Если указанному соотношению не удовлетворяет ни одно кодовое слово, то детектор выдает сигнал стирания. Критерий β определяется дополнительной информацией, получаемой с накопителя. Функция распределения этого критерия удовлетворяет условию

$$P(\beta \le x) \le K(x) \cdot Q,$$

а совместная вероятность события $\beta \ge x/$ и события, состоящего в выдаче декодером неправильного решения, удовлетворяет условию

$$P(\beta \ge x,$$
 неправильное решение) $\le \frac{1}{K(x)} \cdot Q.$

Здесь Q- непрерывная функция, зависящая от переходных вероятностей, а K(x)- монотонно и неограниченно возрастающая непрерывная функция, которая при $x = \beta_0$ принимает значение 1.

Рассмотрим случай, когда детектор выдает решение с одновременным указанием его веса α, а декодер кода РС осуществляет декодирование по алгоритму Форни с использованием условий (3).

Будем считать, что $\alpha = 0$ (стирание), если $\beta \leq \beta_0$; если значение β_1 таково, что $K(\beta_1) = 1/BQ$, то будем полагать $\alpha = +1$ при $\beta \geq \beta_1$. Введем определения

 $P_C(x)dx = P(x \le \beta \le x + dx, \text{правильное решение}),$ $P_e(x)dx = P(x \le \beta \le x + dx, \text{неправильное решение}).$

В наиболее благоприятном случае имеем

$$\begin{split} P(\beta \leq \beta_0) &= Q, \\ P_C(x)dx &= QK'(x)dx, \qquad \beta_0 \leq x \leq \beta_1, \\ P_e(x)dx &= QK^{-2}(x)K'(x)dx, \qquad \beta_0 \leq x \leq \beta_1, \\ P(\beta \geq \beta_1, C) &= P(\beta \geq \beta_1, \qquad \text{правильное решение}) = 1, \\ P(\beta \geq \beta_1, i) &= P(\beta \geq \beta_1, \qquad \text{неправильное решение}) = Q^2 \cdot B. \end{split}$$

Случайная величина α ', которая равна $\alpha(\beta)$ при правильном решении и $-\alpha(\beta)$ при неправильном решении, имеет производящую функцию моментов распределения

$$g(-S) = \overline{e^{-s\alpha'}} = P(\beta \le \beta_0) + e^{-S} \cdot P(\beta \ge \beta_1, C) + e^{S} \cdot P(\beta \ge \beta_1, i) + \int_{\beta_0}^{\beta_1} dx \left[P_C(x) e^{-S\alpha(x)} + P_e(x) e^{-S\alpha(x)} \right].$$

Найдем значение $\alpha(x)$, минимизирующее величину g(-S).

Для этого приравняем нулю частную производную

$$P(\beta \le x) \le K(x) \cdot Q,$$

$$\frac{\partial}{\partial \alpha(x)}g(-S) = -SP_C(x)e^{-S\alpha(x)} + SP_e(x)e^{-S\alpha(x)}.$$

и получим в результате

$$\alpha(x) = \frac{1}{2S} ln \frac{P_c(x)}{P_e(x)} = \frac{1}{S} ln K(x), \qquad \beta_0 \le x \le \beta_1.$$

Заметим, что $\alpha(x)$ может быть большей 1.

Имеем

$$g(-S) = Q + e^{-S} + e^{S}B \cdot Q^{2} + Q \int_{\beta_{0}}^{\beta_{1}} dx [2K^{-1}(x)K'(x)].$$

Вычисление интеграла дает

$$\int_{\beta_0}^{\beta_1} dx [2K^{-1}(x)K'(x)] = 2\ln\frac{K(\beta_1)}{K(\beta_0)} = 2\ln\left(\frac{1}{BQ}\right),$$

Для определения минимума P_1 - средней вероятности того, что сообщение не будет декодировано правильно по алгоритму Форни с использованием условий (3), найдем S, при котором достигается минимум g(-S):

 $g'(-S) = 0; S = lnQ + lnB^{1/2},$

Поэтому

$$g(-S) = (1 + 2ln(1/BQ) + 2\sqrt{B})Q,$$

И

$$P_1 \leq C_n^d [g(-S)]^d = C_n^d (1 + 2ln(1/BQ) + 2\sqrt{B})^d \cdot Q^d \approx \left[\frac{n}{d}\right]^d \cdot \frac{1}{\sqrt{2\pi d}} \cdot (1 + 2ln(1/BQ) + 2\sqrt{B})^d \cdot Q^d.$$

Здесь и далее величина *n* полагается произвольно большой, $n \gg d$.

При декодировании по критерию МОР будем считать, что

 $K(\beta_1) = Q^{-1}$.Тогда $P(\beta \ge \beta_1, i) = Q^2$. В этом случае всегда, как и требуется, $\alpha \le 1$.

Для определения минимума P_2 - средней вероятности того, что сообщение не будет декодировано правильно по критерию МОР, найдем S, при котором достигается минимум величины

$$S(1-\delta)+\mu(-S).$$

Приравнивая нулю частную производную, получим

Решая это уравнение относительно S, находим, что минимум имеет место при

$$S = lnQ^{-1} + ln\left(\frac{2}{1+2lnQ^{-1}} \cdot \frac{\delta}{2-\delta}\right).$$

Поэтому

$$g(-S) = Q(1 + 2lnQ^{-1}) + Q \cdot \left(\frac{2}{1 + 2lnQ^{-1}} \cdot \frac{\delta}{2 - \delta}\right)^{-1} + Q \cdot \left(\frac{2}{1 + 2lnQ^{-1}} \cdot \frac{\delta}{2 - \delta}\right) \approx Q \cdot \frac{1 + 2lnQ^{-1}}{\delta}.$$

И

$$P_{2} \leq e^{S(n-d)} [g(-S)]^{n} \approx e^{(n-d)(lnQ^{-1} + ln\frac{\delta}{1+2lnQ^{-1}})},$$
$$Q^{n} \cdot \left[\frac{1+2lnQ^{-1}}{\delta}\right]^{n} \approx Q^{d} \cdot \left[\frac{1+2lnQ^{-1}}{\delta}\right]^{d} = Q^{d} (1+2lnQ^{-1}) \left[\frac{n}{d}\right]^{d}.$$

Определим, в каком случае получаются лучшие результаты при использовании границ, определяемых условиями (3), чем при использовании границ, определяемых условием (1):

$$\begin{bmatrix} n\\ d \end{bmatrix}^{d} \cdot \frac{1}{\sqrt{2\pi d}} (1+2\ln\left(\frac{1}{BQ}\right) + 2\sqrt{B})^{d} \cdot Q^{d} < Q^{d} \cdot (1+2\ln Q^{-1})^{d} \cdot \left[\frac{n}{d}\right]^{d},$$
$$\frac{1}{\sqrt{2\pi d}} (1+2\ln\left(\frac{1}{BQ}\right) + 2\sqrt{B})^{d} < (1+2\ln Q^{-1})^{d},$$
$$2\sqrt{B} + 2\ln B^{-1} < \left[(2\pi d)^{\frac{1}{2d}} - 1 \right] \cdot (1+2\ln Q^{-1}).$$

P < P

4.2.5. Блоки памяти с самоконтролем

Для обеспечения необходимого уровня надежности блоков памяти конфигурируемых схем используются методы помехоустойчивого кодирования.

С целью повышения эффективности коррекции предлагается обнаруживать неисправности запоминающих элементов внутри блоков памяти, сигналы обнаружения отказов или сбоев выводить на специальные контрольные выводы и использовать при исправлении ошибок. Сигналы на контрольных выводах свидетельствуют о стирании выходных данных, либо задают значения коэффициентов недостоверности выходных данных и применяются в декодерах, исправляющих ошибки и стирания и декодерах мягкого решения [102] – [103].

Применение блоков памяти с самоконтролем позволяет при заданной корректирующей способности уменьшать избыточность, а при заданной избыточности увеличивать число исправляемых и обнаруживаемых ошибок.

В настоящем разделе исследуются структура и функциональные возможности блоков памяти с самоконтролем.

4.2.5.1. Требования, предъявляемые к блокам памяти с самоконтролем.

Кроме обычных требований, предъявляемых к блокам памяти по быстродействию, потребляемой мощности, надежности и т.д., запоминающие устройства (ЗУ) с самоконтролем должны удовлетворять и дополнительным требованиям.

1. Полнота самоконтроля.

В декодере исправляются ошибки стирания, либо используется мягкое решение. Количество необнаруженных схемами самоконтроля отказов и сбоев должно быть незначительным по сравнению с общим их количеством.

2. Наличие точных данных, характеризующих эффективность самоконтроля.

Определение характеристик запоминающих устройств с самоконтролем необходимо для расчета параметров надежности блоков памяти. Основной характеристикой является λ_H - интенсивность необнаруженных схемами самоконтроля отказов и сбоев запоминающих устройств.

Неисправности блоков памяти, которые обнаруживают схемы самоконтроля, не всегда приводят к ошибкам выходных данных, поэтому дополнительной характеристикой является вероятность ошибки выходного данного при наличии сигнала "неисправность".

3. Тестируемость схем самоконтроля.

Работоспособность схем самоконтроля можно проверить только при наличии в микросхеме схемы внесения ошибок.

4.2.5.2. Способы выполнения требований, предъявляемых к блокам памяти с самоконтролем.

В ЗУ обычно выделяют четыре функциональных блока: блок выборки слова (БВС), накопитель (Н), блок ввода-вывода данных (БВВ), блок управления (БУ). Функция блока управления связана с обеспечением нормальной работы блоков БВС, Н и БВВ, поэтому правильность работы трех последних блоков указывает и на нормальное функционирование БУ. Таким образом, для полноты контроля ЗУ достаточно проконтролировать правильность выполнения своих функций блоками БВС, Н, БВВ.

БВС состоит из адресных схем и дешифраторов и возбуждает разрядные и числовую шины в соответствии с адресными сигналами. Проконтролировать работу БВС можно с помощью записи в накопитель контрольного кода каждой шины. Контрольный код при обращении к ЗУ

считывается с возбужденных шин и сравнивается с контрольным кодом адреса.

Определение в ЗУ с самоконтролем соответствия выбранных шин четному или нечетному адресному слову позволяет проверять работу не только БВС микросхемы, но и адресных цепей. Для этого разряд проверки на четность адресного слова нужно формировать внешними схемами и передавать в блок памяти одновременно с адресными сигналами. Возможность контроля передачи адресных сигналов по внешним цепям существенное преимущество ЗУ с самоконтролем, так как обнаружение и исправление ошибок, вызванных сбоем или отказом адресных цепей, внешними средствами требует больших аппаратурных и временных затрат.

Контроль матрицы запоминающих элементов целесообразно проводить с помощью кодов. Наиболее простым является код с проверкой на четность. Одним из основных направлений развития устройств памяти большой емкости является использование помехоустойчивого кодирования для исправления ошибок [93] - [95]. Используемые коды позволяют не только исправлять одиночные, но и обнаруживать двойные ошибки, т.е. проводить контроль накопителя информации. Схемы контроля накопителя и схемы исправления ошибок в этом случае могут быть выполнены в виде одного общего блока.

4.2.5.3. Классификация отказов и сбоев элементов блоков памяти и способы их обнаружения.

Сбои и отказы различных элементов во время работы блоков памяти приводят к ошибкам в разрядных данных, считываемых из ЗУ. Далее рассмотрены неисправности различных схем, элементов и цепей ЗУ, последствия их проявления и способы обнаружения.

Вероятность возникновения одновременно двух и более ошибок в считываемом слове данных ЗУ пренебрежимо мала. Рассматриваемые способы позволяют обнаруживать одиночные отказы и сбои элементов микросхем ЗУ.

Запоминающие элементы. Одна из наиболее распространенных причин сбоев запоминающих элементов – воздействие тяжелых заряженных частиц (ТЗЧ). Эти частицы бомбардируют запоминающие элементы (ЗЭ), имея энергию, достаточную для изменения состояния последних.

Сбои в работе ЗЭ могут вызываться также помехами, причинами которых могут служить неправильные согласования электрических цепей кристалла или воздействия сигналов с соседних узлов и элементов.

Отказы запоминающих элементов возможны вследствие неисправности запоминающего элемента или контакта к ЗЭ и приводят к одиночным ошибкам внутреннего слова ЗУ.

Введение проверки внутреннего слова ЗУ на четность позволяет обнаруживать отказы и сбои ЗЭ. Для обнаружения сбоя двух соседних ЗЭ разряды, считанные с соседних ЗЭ числовой шины, следует отнести к различным внутренним словам, контролируемым на четность.

<u>Разрядные шины</u>. Попадание ТЗЧ в разрядную шину может вызвать изменение заряда на ней при считывании (особенно это характерно для диффузионных шин) и ошибку выходного данного. Такого вида сбой вызывает одиночную ошибку внутреннего слова ЗУ, считанного с разрядных.

Отказы разрядных шин возможны трех типов: обрывы, замыкания разрядных шин друг с другом и смыкания с другими слоями полупроводниковой структуры. Первый и третий типы отказов приводят к одиночным ошибкам, второй тип может приводить к ошибкам двух соседних разрядов слова, считанного с ЗЭ числовой шины.

Обнаружение всех типов ошибок разрядных шин следует проводить проверкой внутреннего слова ЗУ на четность. Для обнаружения двух соседних ошибок длину внутреннего слова нужно выбирать в два раза меньшей числа ЗЭ числовой шины.

<u>Усилители считыван</u>ия. Сбои усилителей считывания, вызванные ТЗЧ, или отказы, вызванные неисправностями элементов, приведут к одиночным ошибкам внутреннего слова ЗУ.

Обнаружение отказов и сбоев усилителей считывания неразличимо для схемы контроля от случаев "отказ и сбой ЗЭ" и "отказ и сбой разрядной шины".

<u>Числовые шины</u>. Отказы числовых шин возможны вследствие обрывов, замыканий шин друг с другом и с другими слоями полупроводниковой структуры, что приводит либо к отсутствию выборки числовой шины, либо к возбуждению на выходе больше одной шины.

Обнаружение таких отказов возможно простейшими схемами контроля, реагирующими на полное отсутствие или наличие двух и более выбранных шин.

<u>Адресные схемы</u>. Адресные схемы, как и рассматриваемые ниже дешифраторы, схемы ввода-вывода и схемы управления, являются периферийными схемами ЗУ. Сбои периферийных схем могут быть обусловлены ТЗЧ или различного рода помехами. Отказы периферийных схем вызываются нарушениями внутренних контактных узлов, нарушениями в проводящих слоях и другими причинами.

Отказы и сбои адресных схем могут привести либо к так называемой парафазной ошибке, то есть к наличию неисправности типа $A = \overline{A} = 0$ или $A = \overline{A} = 1$, где A и \overline{A} - парафазные выходы адресных схем, либо выдачу неверного кода адреса на дешифраторы.

Парафазные ошибки адресных схем приводят к отсутствию выбранной шины на выходе дешифратора или к возбуждению двух и более шин.

Такие ошибки можно обнаружить схемами, аналогичными схемам контроля числовых тин.

Неверный код адреса вызовет выбор дешифратором одной, но неверной шины. Для обнаружения такой ошибки необходимо образовать контрольный код адреса, например, сверткой по модулю 2, и сравнить этот код с контрольным кодом выбранной шины. Несовпадение кодов укажет на ошибку адресных схем.

<u>Дешифраторы</u>. Отказы и сбои в работе дешифраторов приводят либо к отсутствию на выходе выбранной шины, либо к возбуждению неверной шины, либо к возбуждению двух или большего числа шин. Первый и третий случаи неправильной работы дешифраторов обнаруживаются с помощью схем контроля, реагирующих на полное отсутствие или наличие двух и более возбужденных шин. Ошибка выбора дешифратором неверной шины аналогична соответствующей ошибке адресной схемы и обнаруживается аналогичными схемами контроля.

<u>Схемы управления</u>. Отказы и сбои схем управления могут приводить к неправильной работе отдельных схем и элементов ЗУ и обнаруживаются методами обнаружения ошибок соответствующих схем и элементов.

Некоторые неисправности схем управления могут приводить к неправильной работе всех или некоторых групп схем ЗУ. Такие неисправности обнаруживаются с помощью простых схем контроля основных напряжений и сигналов ЗУ.

4.2.5.4. Структура схем обнаружения отказов и сбоев узлов блоков памяти.

Обнаружение неисправностей всех элементов ЗУ можно осуществить с помощью четырех схем. Схема контроля внутреннего слова ЗУ обнаруживает отказы и сбои запоминающих элементов, разрядных шин, усилителей считывания и схем ввода-вывода. Две схемы определения соответствия кода адреса выбранной шине обнаруживают отказы и сбои адресных схем, дешифраторов и числовых шин. Схема формирования сигнала "неисправность ЗУ" обнаруживает отказы и сбои схем управления (см. табл. 4.1).

В настоящем разделе рассмотрена структура и функционирование перечисленных схем обнаружения неисправностей элементов ЗУ.

<u>Схема контроля внутреннего слова ЗУ</u>. Структурная схема контроля внутреннего слова ЗУ состоит из сумматоров по модулю 2, входы которых

подключаются через управляющие транзисторы к разрядным шинам, и схемы формирования проверочных разрядов.

Таблица 4.1 - Основные узлы и элементы блоков памяти и контролирующие их схемы обнаружения неисправностей

Основные схемы и элементы	Схемы обнаружения
ЗУ	неисправностей
Запоминающие элементы	Схемы контроля внутреннего
	слова ЗУ
Разрядные шины	- // -
Усилители считывания	- // -
Схемы ввода-вывода	- // -
Словарные шины	Схема определения
	соответствия кода адреса
	выбранной шине
Адресные схемы и	- // -
дешифраторы	
Схемы управления	Схема формирования сигнала
	"неисправность ЗУ"

<u>Схема определения соответствия кода адреса выбранной шине</u>. Отказы и сбои адресных схем, дешифраторов и числовых шин могут вызвать три типа неисправностей: возбуждение двух или более шин, отсутствие выбранных шин, возбуждение неверной шины.

Схема определения соответствия кода адреса выбранной шине содержит элемент "1 и только 1", с помощью которого обнаруживаются неисправности первого и второго типов. Определение неисправности третьего типа происходит в схеме с помощью сумматора по модулю 2, определяющего четность адресного кода, столбца постоянных запоминающих элементов, определяющего четность адреса возбужденной шины, и двухвходового элемента "исключающее ИЛИ", осуществляющего сравнение. Сигнал с выхода элемента "исключающее ИЛИ" укажет на несоответствие кода адреса выбранной шине,

Формирование сигналов неисправности производится как во время записи, так и во время считывания.

<u>Схема формирования сигнала "неисправность ЗУ"</u>. Схема предназначена для формирования контрольного сигнала в случае обнаружения отказа или сбоя элементов ЗУ, а также для обнаружения ошибок в работе схем управления. В состав схемы входят узлы контроля управляющих сигналов и схема формирования выходного сигнала.

При записи информации в ЗУ схема формирует контрольный сигнал в случаях обнаружения неисправности адресных схем, дешифраторов, числовых шин и схем управления. При считывании информации из ЗУ схема формирует контрольный сигнал в случаях обнаружения отказов и сбоев любых элементов ЗУ.

4.2.5.5. Особенности структуры и функционирования блоков памяти с самоконтролем.

<u>Структура ЗУ с самоконтролем</u>. Структурная схема ЗУ с самоконтролем содержит дополнительно к блокам ЗУ схему контроля внутреннего слова, две схемы определения соответствия кода адреса выбранной шине для контроля выборки разрядных и числовых шин и схему формирования контрольного сигнала.

<u>Функционирование ЗУ с самоконтролем</u>. При записи информации в ЗУ с самоконтролем одновременно с записью информационных разрядов происходит запись проверочных разрядов, сформированных в схеме контроля внутреннего слова. В случае обнаружения отказов и сбоев адресных схем, дешифраторов, числовых шин и схем управления на контрольном выводе микросхемы появляется сигнал "неисправность ЗУ". Для устранения ошибки записи, вызванного сбоем этих схем, следует произвести повторную запись данных в ЗУ. При считывании и регенерации данных в ЗУ с самоконтролем происходит обнаружение неисправности всех элементов и схем ЗУ. Появляющийся в случае обнаружения отказов и сбоев на контрольном выводе сигнал свидетельствует о стирании выходного данного.

Отказ или сбой ЗУ не обнаружится только в случае возникновения ошибок, к которым нечувствительны внутренние схемы контроля. Такие ошибки могут быть вызваны неисправностями одновременно двух и более соседних запоминающих элемента числовой шины.

Варианты ЗУ с самоконтролем. Применение кодов о минимальным расстоянием d > 2 внутри микросхемы расширяет возможности обнаружения неисправностей элементов ЗУ. В этом случае возможно обнаружение d - 1 ошибок внутреннего слова.

4.2.5.6. Блоки памяти с самоконтролем для декодера с мягким решением.

При декодировании блоковых кодов декодер может использовать информацию мягкого решения. Для большого класса декодеров, например при декодировании по критерию минимального обобщенного расстояния, такая информация поступает в виде веса α, определяющего надежность выдаваемого решения. В этом случае блок памяти с самоконтролем должен иметь несколько контрольных выводов, появление какой-либо комбинации сигналов на которых, указывает на определенное значение веса а. Определение комбинации сигналов, подаваемых на контрольные выводы, происходит внутри микросхемы и основано на использовании информации, поступающей с различных схем обнаружения неисправности. Так, например, в ЗУ с самоконтролем при поступлении сигнала неисправности со схемы контроля внутреннего слова ЗУ вероятность ошибки выходного данного меньше, чем при поступлении сигнала неисправности со схемы определения соответствия кода адреса выбранной шине. Поэтому, в этих случаях будут различные комбинации сигналов, установленные на контрольных выводах, и, соответственно, различные веса α. Значения весов α для каждой возможной комбинации контрольных сигналов устанавливает изготовитель микросхемы, указывая их в технических условиях или справочных данных, наряду с другими параметрами конфигурируемой интегральной схемы.
В выходном информационном слове с помощью встроенных схем самоконтроля становится известным местоположение искаженного символа.

Искаженный символ, местоположение которого известно, называется (стиранием). Кроме стертым символом стертых символов могут ошибки, местоположение существовать И которых неизвестно, или случайные ошибки. Если код имеет минимальное расстояние d и произошло е стираний и если

$$d = 2t + e + 1,$$

то эти стирания вместе с ошибками, число которых не превосходит t, могут быть исправлены. Таким образом, для исправления одного стирания требуется код с расстоянием не меньшим d=2. Такое расстояние обеспечивает код с проверкой на четность.

В ряде случаев необходима защита от двух ошибок в слове данных. Такие ошибки можно устранять методами исправления случайных ошибок. Для этого можно воспользоваться кодами БЧХ. Применение этих кодов требует большой избыточности и ухудшает быстродействие ЗУ. Однако, используя метод определения отказавших разрядных шин с помощью встроенных схем самоконтроля, эти две ошибки можно свести к одному стиранию и одной случайной ошибке, что дает существенный выигрыш в числе требуемых контрольных разрядов и скорости обработки информации.

В основе работы такого ЗУ лежит применение сведений о стертых символах, полученных методами, описанными в разделе. В устройстве используется расширенный код Хэмминга, обеспечивающий расстояние d=4. По синдрому кода Хэмминга определяется количество ошибок в слове. В случае одной ошибки исправление происходит стандартно. В случае двух ошибок в первую очередь происходит исправление стирания. После исправления стирания вычисляется синдром, по которому исправляется вторая ошибка. Более быстрым является коррекция синдрома после исправления стирания, то есть изменение тех его разрядов, при вычислении которых был использован неверный символ.

4.3. Схемотехнические методы повышения сбоеустойчивости элементов хранения конфигурируемых интегральных схем

4.3.1. Варианты исполнения триггеров. Триггеры с элементами обнаружения сбоев

Ячейки памяти и триггеры логических элементов, конфигурационной и пользовательской памяти специализированных программируемых интегральных схем космического назначения должны обеспечивать устойчивое функционирование при воздействии различных факторов космического пространства [104].

Одиночные сбои и отказы в интегральных схемах (ИС) в космическом пространстве могут быть вызваны воздействием тяжелых заряженных частиц (ТЗЧ). Воздействие ТЗЧ на ИС характеризуется величиной линейной передачи энергии (ЛПЭ) – потерей энергии частицы при прохождении единицы пути в материале мишени. Для бортовой аппаратуры космических аппаратов устанавливается уровень стойкости к воздействию ТЗЧ (пороговое значение ЛПЭ) по катастрофическим отказам и тиристорному эффекту, как правило, 60 МэВ×см²/мг [105] и более.

Элементы памяти, в том числе и триггеры, чувствительны к одиночным радиационным эффектам. Повысить сбое- и отказоустойчивость триггеров позволяет применение двойного и тройного модульного резервирования (ДМР и ТМР). Примеры схем ТМР триггеров приведены в [106], [107]. Они включают в себя три идентичных модуля и систему мажоритарного голосования. Снизить аппаратные затраты ТМР триггеров, сохранив параметры надежности схем, может позволить использование конструкции ячеек с самовосстановлением – в иностранной литературе они называются «dual interlocked storage cell» (DICE) [108] – и с выходной схемой фильтрации помех на основе мажоритарного голосования [109]. Далее в работе триггеры и защелки с такой конструкцией именуются как DICE триггеры и DICE защелки соответственно.

DICE триггеры могут быть исполнены различными способами в зависимости от организации входящих в них защелок: на мультиплексорах (обычно входит в состав стандартной библиотеки цифровых элементов), на элементах «Bus Holder», на комбинационной логике.

Триггеры на основе мультиплексоров

Мультиплексор, входящий в состав защелки, не имеющей входа асинхронной предустановки (сброса), изображен на рисунке 4.5а. Если выход Out соединить с In0, то мультиплексор превращается в защелку, где вход In1 играет роль входа защелкиваемых данных, а вход М – роль управляющего защелкиванием сигнала. При наличии «0» на входе М защелка находится в режиме хранения состояния, а при наличии «1» – в режиме записи. Для такой защелки режим хранения устанавливается, когда на входе М – «1», а режим записи – при наличии «0».

D триггер составляет пара защелок соединенных последовательно. Какой тип защелки находится ближе к входу, а какой – ближе к выходу, определяет способ синхронизации триггера. Если ближе к входу находится защелка с записью по «1», а на выходе с записью по «0», то триггер является тактируемым по спаду сигнала на входе С. Схема такого триггера изображена на рисунке 4.56. Если необходимо получить триггер, тактируемый по фронту (где данные с входа D передаются на выход Q только при переходе сигнала С из «0» в «1»), защелки следует переставить местами.



Рисунок 4.5 – Структурные схемы: а – защелка на мультиплексоре; б – D-триггер

Исполнение триггеров на элементах Bus Holder

Bus Holder – элемент памяти со слабой положительной обратной связью. Для записи информации используется шинный формирователь. На рисунке 4.6а изображена схема защелки, не имеющей входа асинхронной предустановки (сброса), запись в которую с входа D производится при наличии «1» на входе OE, а на рисунке 4.66 – при наличии «0». На рисунке 4.7 изображен функциональный аналог схемы, изображенной на рисунке 4.56.



Рисунок 4.6 – Защелки на Bus Holder: а – защелка на Bus Holder с сигналом разрешения высокого активного уровня; б – защелка на Bus Holder с сигналом разрешения низкого активного уровня



Рисунок 4.7 – Схема D-триггера на элементах Bus Holder и шинных формирователях

Исполнение триггеров на комбинационной логике

В триггер, называемый «ячейка памяти Канариса» [106] и являющийся функциональным аналогом триггеров, показанных на рисунке 4.56 и рисунке 4.7, входят две защелки, каждая из которых составлена из двух одинаковых комбинационных элементов. Данные для первой защелки должны поступать в прямом и инвертированном виде одновременно. Схема триггера приведена на рисунке 4.8.



Рисунок 4.8 – Схема соединения комбинационных элементов в D триггер

Преобразование защелки в DICE защелку

Преобразование пары триггеров в DICE триггер заключается в преобразовании входящих в них защелок в DICE защелки. Данное преобразование реализуется путем переключения цепей обратной связи пары защелок так, как это показано на рисунке 4.9. После такого преобразования пара независимых защелок превращаются в единую DICE защелку.



Рисунок 4.9 – Ячейки статического оперативного запоминающего устройства: а – схема двух параллельно работающих 6-ти транзисторных ячеек; б – схема 12-ти транзисторной ячейки DICE

На рисунке 4.96 переназначены, по отношению к схеме на рисунке 4.9a, 8 источников сигналов, идущих к затворам транзисторов, следующим образом:

- затвор транзистора nC соединен с истоком транзистора nB;

- затвор транзистора nA соединен с истоком транзистора nD;

- затвор транзистора pC соединен с истоком транзистора pB;
- затвор транзистора pA соединен с истоком транзистора pD.

Связи, оставшиеся без изменений:

- затвор транзистора nB соединен с истоком транзистора nA;

- затвор транзистора nD соединен с истоком транзистора nC;

- затвор транзистора pB соединен с истоком транзистора pA;

- затвор транзистора pD соединен с истоком транзистора pC.

Элементы самоконтроля в DICE триггерах

В работе предложено нарушения штатного функционирования триггеров, вызванные внешним воздействием, или иными помехами, определять с помощью дополнительных элементов в составе триггеров. Информация о потенциальном сбое полезна для организации работы устройства в целом, например, для принятия решения о перезаписи памяти, блоке информации В ИЛИ перезапуске счетчика. Знание местоположения возможной ошибки, называемой стиранием, позволяет эффективно корректировать ошибки, используя методы помехоустойчивого кодирования, или увеличивать эффективность резервирования.

Дополнительные элементы триггера, необходимые для самоконтроля, приводят к увеличению его площади.

Количество транзисторов в триггерах различных типов

В таблице 4.2 представлены данные по количеству транзисторов в триггерах различной организации для оценки аппаратных затрат. В качестве n-МОП транзисторов использовались транзисторы с кольцевым затвором, в качестве p-МОП – обычные прямоугольные транзисторы. Все триггеры имеют вход асинхронного сброса с низким активным уровнем.

	-		
Организация	Конструкция	Синхронизация	Количество
			транзисторов
	община <i>я</i>	по спаду	41
	ооычная	по фронту	
Па мультиплексорах (1)	DICE	по спаду	75
	DICE	по фронту	
На элементах Bus		по спаду	39
Holder (II)	ооычная	по фронту	

Таблица 4.2 – Аппаратные затраты на организацию триггеров

	DICE	по спаду	77
	DICE	по фронту	
На комбинационной логике (III)	обычная	по спаду	44
		по фронту	42
	DICE	по спаду	89
		по фронту	82

При составлении таблицы было учтено, что на выходе DICE триггеров дополнительная схема фильтрации помех, введена а схемы с мультиплексорной организацией дополнены элементами, предотвращающими сбой, описанный в [110]. В таблице также учтены и сдвоенные транзисторы. Приводимые данные являются иллюстрацией, пригодной для оценки дополнительных затрат при замене триггеров модернизируемого проекта на DICE триггеры.

Для примерной оценки аппаратных затрат на рисунке 4.10 представлена топология DICE триггера на базе Bus Holder.



Рисунок 4.10 - Топология DICE триггера на базе элементов Bus Holder, выполненная с использованием проектных норм 180 нм. Ориентировочная площадь триггера составляет 570 мкм²

4.3.2. Результаты моделирования DICE-триггеров

Результаты моделирования быстродействия триггеров

Ниже приводится сравнительный анализ характеристик, полученных моделированием работы триггеров различных типов. Моделирование проводилось для технологии «объемный кремний» с топологическими нормами 180 нм в диапазоне рабочих температур от -60 до +125 °C, и в диапазоне питающих напряжений 1.8 $B \pm 10\%$.

На базе триггеров был собран делитель частоты с коэффициентом деления 2. В таблице 4.3 приводится предельная частота переключения

делителя для худшего случая. Моделирование проводилось при нагрузке, равной восьми стандартным инверторам.

DICE триггеры на мультиплексорах является наиболее быстродействующими. Для худшего случая (в режиме: 125 С, 1.6 В) такие триггеры, как по спаду, так и по фронту, дают предельную частоту делителя 750 МГц.

В таблице 4.4 приведены данные по задержке переключения выходов триггера по отношению к фронту сигнала тактирования для худшего случая (в режиме: 125 С, 1.6 В).

Из таблицы 4.4 видно, что триггеры, построенные на базе комбинационной логики, уступают триггерам двух других типов по предельной тактовой частоте делителя частоты (см таблицу 4.3). В зависимости от типа синхронизации, время переключения выхода триггера в «0» может оказаться больше, чем время переключения в «1» в 1.5 раза и наоборот.

T			Режим моделирования				
I ИШ	Констр.	Синхр.	25 °C	125 °C	125 °C	-60 °C	-60 °C
триггера			1.8 B	1.6 B	2.0 B	1.6 B	2.0 B
	- 5	по спаду	1.90	1.00	1.40	1.50	2.15
т	ооычная	по фронту	1.80	0.95	1.30	1.50	2.10
1	DICE	по спаду	1.35	0.75	1.00	1.05	1.45
	DICE	по фронту	1.35	0.75	1.05	1.05	1.50
	обычная	по спаду	1.90	1.10	1.55	1.45	2.05
п		по фронту	2.00	1.10	1.50	1.55	2.20
11	DICE	по спаду	1.25	0.70	0.95	0.90	1.30
		по фронту	1.20	0.70	0.95	0.90	1.30
	o51.000	по спаду	1.40	0.75	1.00	1.05	1.50
TT	ооычная	по фронту	1.40	0.75	1.05	1.10	1.55
111	DICE	по спаду	1.05	0.55	0.75	0.95	1.10
	DICE	по фронту	1.20	0.65	0.85	0.95	1.35

Таблица 4.3 – Предельная частота делителя частоты на базе триггера в ГГц

Таблица 4.4 – Задержка переключения (пс) по отношению к фронту

			переключе	ение в «1»	переключение в «0»	
Орг-ция.	Констр.	Синхр.	Q	Q	Q	Q
I	ofrancia	по спаду	640	640	800	610
1	ооычная	по фронту	670	570	760	690

	DICE	по спаду	690	1110	860	920
DICE		по фронту	720	1030	830	990
II	ofining	по спаду	530	400	510	530
	ооычная	по фронту	480	530	670	510
	DICE	по спаду	710	910	660	936
	DICE	по фронту	650	1033	830	930
	обычная	по спаду	540	550	840	820
III		по фронту	730	730	490	460
	DICE	по спаду	900	1050	860	1160
	DICE	по фронту	1100	590	850	900

Моделирование влияния помех в DICE элементах памяти

Для моделирования помех были выбраны обычная защелка и DICE защелка мультиплексорной организации с высоким активным уровнем разрешения записи данных (т.е. запись данных в защелку производится при подаче «1» на вход «Enable»). Для имитации помех в обычной защелке было организовано замыкание стока и истока транзисторов в цепи положительной обратной связи. В DICE защелке аналогичные соединения были организованы в цепях одной из ветвей схемы.

На рисунке 4.11 представлены результаты моделирования для ситуаций, не приводящих к сбоям в DICE защелке. Помехи, вызываемые T3Ч, не приводят к утере информации, записанной и хранимой в DICE защелке. Для сравнения показано, что такие помехи приводят к искажениям данных в обычной защелке.

Был выбран интервал времени, равный 17 периодам подачи строба записи данных в защелки (сигнал Enable). В неактивной фазе сигнала Enable (потенциал «0») производилась смена состояния на входе Data.



Рисунок 4.11 – Результаты моделирования влияния помехи на сохранность информации в DICE защелке (для случаев, не приводящих к сбоям) и влияния помехи на сохранность информации в обычной защелке

Интервал времени разделен на 3 фазы:

1) На входе асинхронного сброса (сигнал Clear) поддерживался активный уровень («0»). На не инвертированном выходе защелки обычной конструкции и на не инвертированном выходе DICE защелки (сигналы q_SEE и q_DICE) сохранялось состояние «0».

2) На второй фазе уровень сигнала Clear – неактивен («1»). На не инвертированном выходе защелки обычного типа и на не инвертированном выходе DICE защелки (сигналы q_SEE и q_DICE, соответственно) сформировался меандр, фронт которого совпадает с началом появления «1» на входе Enable.

3) Уровень сигнала Clear также был неактивен («1»), на входы SEE_0 и SEE_1 подавались помеховые импульсы. При этом, импульсы имитации действия T3Ч приходили после переднего фронта (перехода из «0» в «1») импульса на входе Enable, и заканчивались до переднего фронта очередного импульса на входе Enable. Импульсы имитации действия T3Ч начинались до прихода заднего фронта (перехода из «1» в «0») импульса на входе Enable. На не инвертированном выходе защелки обычного типа (сигнал q_SEE) наблюдался сбой – принудительное установление состояния «0», когда появлялся «0» на входе SEE_0, и принудительное установление состояния «1», когда появлялся «1» на входе SEE_1. На не инвертированном выходе до SEE_1. На не инвертированном выходе ACC в SEE_2. Все так же наблюдался меандр, фронт которого совпадает с началом появления «1» на входе Enable – сбой для данной защелки не произошел.

На эпюрах видно, что помеха в цепях обратной связи DICE защелки все же возникала. В частности, в цепях с0 и а0 (выходы двух «плечей» DICE защелки, сигнал с которых передается на схему фильтрации помех) возникали помеховые импульсы. Эти импульсы совпадают по фронту с импульсами на входах SEE_0 и SEE_1. Однако дополнительная схема фильтрации помех на выходе «плечей» предотвращает передачу помех на выходные линии схемы.

На рисунке 4.12 представлены результаты моделирования для 4 ситуаций, приводящих к сбоям в DICE защелке. Эпюры показывают, что к сбоям приводит ситуация, когда помеха, вызванная ТЗЧ, возникает на положительном фронте сигнала Enable: либо записи информации в элемент памяти не происходит, либо запись в защелку происходит с задержкой, которая может оказаться недопустимой для быстродействующих схем.



Рисунок 4.12 – Результаты моделирования влияния помехи на сохранность информации в DICE защелке (в т.ч., для случаев, приводящих к сбоям)

При использовании помехоустойчивого кодирования исправляются ошибки в тех ситуациях, когда вследствие сбоя записи данных в тригтер не произошло. К подобным кодам относятся коды Хэмминга, применяемых также в устойчивых к воздействию ТЗЧ статических запоминающих устройствах [111]. Если необходимо полностью устранить сбой, то, в таком случае, допустим поиск других, более сложных технических решений (например, использование ТМР конструкции тригтера или конструкции с пространственным и временным параллелизмом; переход к топологии распределенных транзисторов, дающей, при попадании в них ТЗЧ, уменьшенные импульсы тока).

Для применения DICE элементов в узлах схем, в которых важно установление факта сбоя, можно построить детектор сбоев, показанный на рисунке 4.13. Результаты моделирования его работы приведены на рисунке 4.14.



Рисунок 4.13 – Структурная схема детектора сбоев в DICE защелке



Рисунок 4.14 – Результаты моделирования работы схемы обнаружения сбоев в DICE защелке

Здесь RS триггер переходит в исходное состояние по сигналу на входе R (после прихода активного уровня на линии Clear). Комбинационная логическая схема X формирует признак сбоя в зависимости от сигналов a0, c0 и Data. Пройдя через буферный элемент S, предназначенный для компенсации задержки в логической схеме X, импульс Enable своим положительным фронтом защелкивает выработанный признак в D триггере. RS триггер фиксирует наличие сбоя. Сигнал на выходе RS триггера можно использовать как команду для перезапуска всей системы.

D триггер и RS триггер могут формировать признак сбоя сразу для целой группы защелок одного типа. Тогда как логическая схема X в этом случае должна формировать общий признак сбоя для данной группы. Аппаратные затраты для построения каждой защелки ориентировочно увеличиваются на 20%.

4.3.3. Результаты применения элементов памяти повышенной надежности в конфигурируемых схемах

На рисунке 4.15 представлены топологии DICE триггера (рисунок 4.15а) и стандартного библиотечного триггера (рисунок 4.15б) базового матричного кристалла 5540TH014A, выполненного по технологии КМОП КНИ 180 нм. DICE триггер занимает 16 базовых ячеек БМК (одна ячейка – 2 пары комплементарных транзисторов), а библиотечный – 9 базовых ячеек.

Пороговое значение ЛПЭ, при котором не обнаруживается сбой в стандартном библиотечном триггере, составляет 17 МэВ×см²/мг, что было выявлено в ходе испытаний БМК 5540TH014A. Испытания также показали, что DICE триггеры обладают устойчивостью к сбоям, вызываемым частицами с ЛПЭ вплоть до 92 МэВ×см²/мг.





б)

Рисунок 4.15 – Топология ячеек БМК: а – DICE триггера; б – стандартного библиотечного триггера

4.4. Выводы

Предложены новые методы повышения живучести, надежности и радиационной стойкости конфигурируемых интегральных схем на основе приемлемых и мягких решений, которые используя дополнительную информацию о вероятных отказах и сбоях функционирования внутренних узлов, позволяют обеспечить корректную работу конфигурируемых схем в условиях неблагоприятных внешних воздействий.

Предложен алгоритм мягкого декодирования блоковых кодов, обеспечивающий большую вероятность правильного декодирования, чем алгоритм Форни.

Разработаны схемотехнические методы повышения сбоеустойчивости элементов хранения конфигурируемых интегральных схем.

Глава 5. Особенности разработки перспективных базовых матричных кристаллов и конфигурируемых аналого-цифровых интегральных схем

5.1. Методы построения интегральных схем цифровых базовых матричных кристаллов

Методы разработки и модели конфигурируемых на этапе производства полузаказных цифровых БМК со встроенными заказными СФ блоками описаны во второй и третьей главах данной работы. Методы повышения надежности и радиационной стойкости – в четвертой главе. В настоящем разделе исследуются подходы к разработке БМК с расширенными функциональными возможностями и улучшенными техникоэкономическими показателями.

5.1.1. Разработка цифровых базовых матричных кристаллов с расширенными функциональными возможностями

Типовая ячейка БМК

Области цифровых элементов в БМК выполнены на основе бесканального принципа построения матричных структур («море вентилей») и позволяют реализовывать как логические элементы, так и триггерные ячейки [112] – [114].

Элементарной составляющей области библиотечных элементов является 4-х транзисторная базовая цифровая ячейка (БЦЯ) [115]. Цифровые ячейки объединены в строки (ряды), ограниченные с двух сторон шинами земли и питания, группы строк собраны в виде матриц определенной размерности.

Стандартная базовая цифровая ячейка содержит 4 транзистора (2 р-МОП и 2 п-МОП) с объединенными затворами, представляющие собой вход

цифрового элемента. БЦЯ имеет несколько свободных контактов: вход (объединенные затворы), сток р-МОП транзистора, исток р-МОП транзистора, сток n-МОП транзистора, исток n-МОП транзистора. Данное количество контактов позволяет реализовать на основе БЦЯ библиотечные цифровые элементы комбинационной или последовательностной логики.

На рисунке 5.1 представлена топология стандартной базовой цифровой ячейки БМК.

Кроме ядра, типовой БМК в своем составе имеет периферийную часть, представляющую собой набор конфигурируемых ячеек типа «вход/выход» со встроенными элементами защиты от воздействия статического электричества.



Рисунок 5.1 - Топология типовой базовой цифровой ячейки БМК

Модернизированные БЦЯ, используемые для расширения функциональных возможностей БМК. Создание областей программируемых элементов в поле ядра цифрового БМК.

Объединение пар транзисторов по затвору обусловлено несколькими причинами:

- Использование комплементарной логики в библиотеке стандартных элементов БМК;
- Упрощение разводки внутри стандартных библиотечных элементов за счёт отсутствия лишних контактов к затворам отдельных транзисторов и металлических соединений, объединяющих их в комплементарные пары;
- Увеличение надёжности за счёт удвоенного количества контактов на затворах и их связи посредством одного технологического слоя (без переходов между слоями и межслойных контактных окон).

Недостатками такого варианта исполнения можно считать:

- 1. Неэффективную реализацию ключей и схем на их основе;
- Сложность получения симметричных задержек переключения элементов;
- Потенциальную сложность реализации некоторых многовходовых элементов.

Данные недостатки можно считать условными, т.к. библиотеки стандартных элементов для полузаказных схем в большинстве случаев не содержат проходные ключи, а содержат элементы КМОП – типа. Кроме того, количество транзисторов одного типа проводимости, включенных параллельно, для современных технологий ограничивается величиной не более 4-х и логические элементы с бо́льшим числом входов реализуются каскадированием элементов с меньшим числом входов. Симметричные задержки, как правило, реализуются только в элементах, предназначенных для построения тактовых деревьев, и их количество невелико относительно общего числа элементов библиотеки.

С целью реализации ряда аналоговых схем и ключей на базовых ячейках цифрового БМК ячейку необходимо модернизировать (рисунок 5.2). Суть модернизации заключается в разделении затворов и изменении соотношения размеров р - и п – канальных транзисторов.

Такое усовершенствование ячейки позволяет:

- Реализовывать стандартные проходные ключи;
- Создавать простые аналоговые схемы;
- Достигать симметричных задержек переключения;

Площадь модернизированной ячейки БМК превышает площадь стандартной ячейки не более чем на 20%.

6	⊠91 ⊠ on o	1 das 1		1.00		
973		(n			<u> </u>	
		CI N	44	+ • 0 0	⊢ in ⊠91	44
	×P4X			<u> 2</u>	X=X	
		<u></u>				

Рисунок 5.2 - Модернизированная ячейка БМК

В данной работе предложено использовать модернизированную ячейку БМК для создания программируемых элементов, то есть для создания областей ПЛИС в поле базовых ячеек БМК. Заказчик схемы, анализируя поставленную задачу, определяет, как ему распорядиться полем ячеек БМК: оставить их только для полузаказного проектирования, или выделить программируемую после изготовления область кристалла. Гибкость данного решения заключается в том, что на этапе разработки «зашивки» БМК заказчик задает размер и расположение областей и для полузаказного проектирования, И для логических схем, программируемых после изготовления. При этом базовый кристалл БМК не меняется от «зашивки» к «зашивке».

Для создания областей ПЛИС на основе модернизированных ячеек БМК должна быть предусмотрена возможность реализации на таких ячейках многовходовых комбинационных схем типа LUT, тригтеров, проходных ключей, запоминающих ячеек СОЗУ. Оценка требуемого количества базовых ячеек БМК для создания области ПЛИС показала, что для реализации типового блока ПЛИС, состоящего из 16-ти логических элементов с четырьмя входами каждый, необходимого количества ключей и хранящих элементов, а также реализации коммутационных блоков всех уровней, цепей синхронизации и других управляющих узлов, необходимо задействовать более 10 000 базовых ячеек БМК. Таким образом, в поле БМК, состоящем из 1 000 000 базовых ячеек, что составляет примерно 5 % от емкости

современного БМК, выполненного по технологии 90 нм, можно разместить 100 логических элементов ПЛИС, или около 3 000 системных вентилей.

В данной работе предложена конструкция логической части программируемого элемента, обладающая существенно меньшей площадью по сравнению со стандартной конструкцией. Двухвходовые логические элементы в стандартных ПЛИС реализуются на основе таблиц истинности 2-LUT. С помощью 2-LUT может быть реализован весь перечень двухвходовых логических функций, благодаря чему программируемые схемы обладают большой универсальностью. Для функционирования 2-LUT необходимо 4 ячейки конфигурационной памяти. Так как на практике при логическом синтезе проекта цифровой схемы нет необходимости в реализации всевозможных двухвходовых функций, а достаточно базиса из трех логических функций 2И-НЕ, 2ИЛИ-НЕ, НЕ, в работе была предложена схема комбинационной части логического элемента, которая требует всего 2 ячейки конфигурационной памяти (рисунок 5.3).



Рисунок 5.3 - Схема 2-LUT

Схема состоит из 10 МОП транзисторов, сформированных в пять КМОП пар. Входы 1,2 – входы двухвходового логического элемента; входы А, В – выходы 2 ячеек конфигурационной памяти. В отличие от стандартного 2-LUT гле определяется ячейки элемента. выход значением конфигурационной памяти, адрес которой соответствует комбинации предложенном входных сигналов, В элементе выход определяется результатом работы одной из четырех комбинационных схем, выбор которой осуществляется с помощью одно-транзисторных ключей и 2-х ячеек конфигурационной (B отдельных памяти. случаях, например при объединении входов, достаточно одной ячейки конфигурационной памяти).

В таблице 5.1 приведена таблица истинности такой схемы. Когда входы А,В находятся в состоянии логической «1», ключи переключают работу схемы в режим ИЛИ-НЕ, когда находятся в состоянии логического «0» - в режим И-НЕ. Остальные два режима предназначены для инверсии входных сигналов.

А	В	Lut_out
0	0	И-НЕ
0	1	НЕ для 2 входа
1	0	НЕ для 1 входа
1	1	ИЛИ-НЕ

Таблица 5.1 - Таблица истинности схемы 2-LUT

Преимуществом такой схемы является экономия площади при реализации комбинационной логики логических элементов ПЛИС на БМК (рисунок 5.4). Логическая часть схемы занимает 3 ячейки четырехтранзисторной ячейки БМК, в то время как стандартная реализация 2-LUT заняла бы 6 ячеек БМК. Помимо этого экономятся две ячейки памяти, которые занимают несколько ячеек БМК.



Рисунок 5.4 - Схема 2-LUT на ячейках БМК

Предложенная в данном разделе модернизированная ячейка подходит как для реализации цифровых схем, так и для реализации несложных аналоговых схем и программируемых элементов. Дальнейшее увеличение использования площади кристалла БМК эффективности достигается применением специализированных ячеек БМК, или выделением специальных областей программируемых логических схем, встроенных В поле стандартных ячеек БМК.

5.1.2. Методы повышения экономической эффективности базовых матричных кристаллов

При разработке кристаллов БМК типа «море вентилей» на современных технологических процессах (180 нм и меньше), вследствие возрастания числа технологических слоёв металлизации при уменьшении топологических норм, возникает вопрос о количестве слоёв, достаточных для трассировки проектов, т.к. использование всего набора металлических слоёв

для реализации многих проектов становится экономически нецелесообразным.

В отечественной технологии с топологическими нормами 180 нм присутствует 6 слоёв металлизации, при использовании которых в качестве переменных слоёв БМК требуется изготовить 11 фотошаблонов (ФШ), примерно 1/3 ОТ комплекта ΦШ составляющих полного данного технологического маршрута. Изготавливать такое количество ФШ для каждой зашивки БМК непродуктивно. В технологии уровня 28 нм используется 10 слоёв металлизации, поэтому изготавливать для каждой зашивки по 19 ФШ (около 40% от общего комплекта) с учетом стоимости изготовления ФШ нерентабельно.

Вопрос существенного снижения количества переменных слоёв БМК для современных технологических процессов становится одним из важнейших, так как от ответа на него зависит принципиальная применимость концепции БМК и её жизнеспособность.

Опыт реализации проектов на БМК, изготавливаемых по технологии с топологическими нормами 90 нм, показывает следующие эмпирические данные:

- Для проекта 1 с плотностью заполнения 73,4% потребовались все 6 трассировочных слоя (плюс 1 слой – металл 1 – используется для разводки внутри стандартных логических элементов);
- Для проекта 2 с плотностью заполнения 49,2% потребовалось 5 трассировочных слоёв (плюс 1 слой – металл 1 – используется для разводки внутри стандартных логических элементов).

Ниже приводится статистика по использованным трассировочным ресурсам проектов.

Использование трассировочных ресурсов проекта 1.

#		Routing #/	Avail	#Track	#Total	%Gcell
#	Layer	Direction	Track	Blocke	ed Gcell	Blocked
# #	M1	Н 5	7395	9773	16434916	77.70%

#	M2	V	51589	9217	16434916	18.21%
#	M3	Η	51589	9217	16434916	15.17%
#	M4	V	51574	9232	16434916	18.27%
#	M5	Η	59354	1452	16434916	4.41%
#	M6	V	15913	4355	16434916	16.49%
#	M7	Η	13749	6519	16434916	27.05%
#						
#	Total		301165	16.58%	115044412	25.33%
И	спользо	вание тр	ассирово	чных рес	урсов проек	ста 2.
#		Routing	#Avail	#Track	#Total ^o	%Gcell
		ncounne	// 1 1 V WII	// I luon	II I Otul	
#	Layer	Directi	on Track	Block	ed Gcell	Blocked
# #	Layer	Directi	on Track	Block	ed Gcell	Blocked
# # #	Layer M1	Directi H	on Track 51554	Block 9252	ed Gcell 16434916	Blocked 60.07%
# # # #	Layer M1 M2	Directi H V	on Track 51554 51584	9252 9222	ed Gcell 16434916 16434916	Blocked 60.07% 18.25%
# # # #	Layer M1 M2 M3	Directi H V H	on Track 51554 51584 51578	9252 9222 9228	ed Gcell 16434916 16434916 16434916	Blocked 60.07% 18.25% 15.19%
# # # # # # #	Layer M1 M2 M3 M4	Directi H V H V	on Track 51554 51584 51578 51564	Block 9252 9222 9228 9242	ed Gcell 16434916 16434916 16434916 16434916	Blocked 60.07% 18.25% 15.19% 18.31%
# # # # # #	Layer M1 M2 M3 M4 M5	Directi H V H V H	on Track 51554 51584 51578 51564 59419	9252 9222 9228 9242 1387	ed Gcell 16434916 16434916 16434916 16434916 16434916 16434916	Blocked 60.07% 18.25% 15.19% 18.31% 4.48%
# # # # # # #	Layer M1 M2 M3 M4 M5 M6	Directi H V H V H V	on Track 51554 51584 51578 51564 59419 15881	Block 9252 9222 9228 9242 1387 4387	ed Gcell 16434916 16434916 16434916 16434916 16434916 16434916 16434916	Blocked 60.07% 18.25% 15.19% 18.31% 4.48% 16.63%
# # # # # # #	Layer M1 M2 M3 M4 M5 M6 M7	Directi H V H V H V H	on Track 51554 51584 51578 51564 59419 15881 13747	Block 9252 9222 9228 9242 1387 4387 6521	ed Gcell 16434916 16434916 16434916 16434916 16434916 16434916 16434916 16434916	Blocked 60.07% 18.25% 15.19% 18.31% 4.48% 16.63% 27.06%
# # # # # # # #	Layer M1 M2 M3 M4 M5 M6 M7	Directi H V H V H V H	on Track 51554 51584 51578 51564 59419 15881 13747	Block 9252 9222 9228 9242 1387 4387 6521	ed Gcell 16434916 16434916 16434916 16434916 16434916 16434916 16434916	Blocked 60.07% 18.25% 15.19% 18.31% 4.48% 16.63% 27.06%

Если посмотреть на количество использованных трассировочных ресурсов кристалла, то заметна лишь существенная разница по 1-му слою металла, которая свидетельствует о большей плотности заполнения матрицы базовых цифровых ячеек (БЦЯ) БМК для проекта 1. Процент заполнения треков в остальных металлах примерно одинаков, однако эти данные ничего не говорят об их локальной плотности.

Ниже приведена статистика по количеству и распределению цепей проектов.

Статистика по цепям проекта 1.

Nets	69676478
metal layer M1	8659727
metal layer M2	32358148
metal layer M3	22911000
metal layer M4	5173834
metal layer M5	539575
metal layer M6	27751
metal layer M7	6443

Via Instances 27154384

Статистика по цепям проекта 2.

Nets	45574423
metal layer M1	6120117
metal layer M2	23433400
metal layer M3	13836431
metal layer M4	2153347
metal layer M5	27250
metal layer M6	2798
metal layer M7	1080

Via Instances 17809136

Данная статистика показывает, что в наиболее заполненных слоях (металлы 2 и 3) количество цепей в проекте 1 больше примерно на 10М (на 38% и 65% соответственно) в каждом, но более показательна статистика по металлам 4 и 5, в которых количество цепей не так велико, но в 2,4 и 19,8 раз больше, чем в проекте 2. Это говорит о том, что в проекте 1 есть области, перегруженные по разводке, для реализации цепей в которых трассировщику пришлось использовать 5-й, и даже 6-е и 7-е металлы (в данных проектах сетки земли и питания не являлись фиксированными, поэтому трассировщик не был ограничен в использовании самых верхних слоёв металлизации).

Наглядно данную статистику иллюстрируют рисунки 5.5, 5.4, 5.7.



Рисунок 5.5 – Топология металлов 3 (жёлто-зелёный) и 4 (коричневый) проекта 2



Рисунок 5.6 – Топология металлов 3 (жёлто-зелёный) и 4 (коричневый) проекта 1

На рисунке 5.5 видно, что в проекте 2 практически отсутствует высокая локальная плотность трассировки в 4-м слое металла. Тогда как на рисунке 5.6 отчётливо видны области с высокой плотностью трассировки в 4-м слое металла проекта 1, для одной из которых также наблюдается высокая локальная плотность трассировки в 5-м металле (рисунок 5.8).

Несмотря на то, что среднее количество задействованных в проектах трассировочных ресурсов примерно одинаково 25,33% и 22,85% и практически идентичен процент использования трассировочных ресурсов по

металлическим слоям со 2-го по 7-й, качественная картина трассировки в проектах сильно отличается. Таким образом плотность заполнения матрицы БЦЯ БМК является ключевым показателем, от которого зависит, сколько трассировочных слоёв необходимо на реализацию проекта.



Рисунок 5.7 – Топология слоёв металлов 3 (жёлто-зелёный), 4 (коричневый), 5 (розовый) проекта 1

Очевидно, что минимальное количество трассировочных слоёв для реализации проекта современных БМК равно 3: 1 слой – металл 1 – определяет соединения внутри стандартных логических элементов библиотеки БМК, 2 слоя – металлы 2 и 3 – являются вертикальными и

горизонтальными слоями трассировки внутрисхемных соединений. Также очевидным является то, что такого количества трассировочных слоёв будет достаточно только для проектов имеющих малую плотность заполнения на конкретной «зашивке» БМК и/или не требующих достижения предельно возможного быстродействия.

Логичным решением по уменьшению количества трассировочных слоёв является выделение нескольких верхних слоёв металлизации (для технологии 90 нм – 2-х слоёв – металл 6 и 7) исключительно под реализацию сетки земли и питания кристалла, которая делается фиксированной на весь спектр возможных зашивок. Также расположение конденсаторов должно быть унифицировано по всей области периферийных цифровых ячеек (ПЦЯ), чтобы обеспечить возможность назначения любой из пар выводов в качестве выводов аналоговой земли и питания, необходимых для таких элементов, как ФАПЧ (PLL) или приёмники/передатчики стандарта LVDS. Потенциальным недостатком такого решения будет невозможность трассировки в верхних слоях толстых металлов (т.е. металлов с самым низким удельным сопротивлением) глобальных и критических сигналов (тактовые сигналы, сигнал сброса и т.п.), но современные САПР умеют автоматически строить тактовые деревья и удовлетворять заданные для них требования при ограниченном количестве трассировочных слоёв.

В общем виде, для проектов с высокой плотностью заполнения матрицы БЦЯ БМК ограничителем по разводимости также может стать наличие в их составе специальных блоков, обладающих высокой степенью внутренней связанности. Например, в проекте 1 таким блоком стала кэшпамять 1-го уровня, собранная на триггерах библиотеки стандартных элементов БМК.

Таким образом, для технологии 90 нм с 7-ю слоями металлизации целевым диапазоном трассировочных слоёв можно считать диапазон от 3 до 5 металлов (включая 1-й металл, в котором реализуется разводка библиотечных элементов). Стоит отметить также, что для внутренней

трассировки некоторых сложных библиотечных элементов (например, триггеров типа dice) может использоваться не один, а два нижних слоя металлизации (1-й и 2-й металлы).

Далее рассмотрим возможность размещения и трассировки одного и того же проекта в БМК при разных плотностях заполнения матрицы БМК и ограничениях в количестве слоёв, доступных для трассировки.

В качестве примера проекта выбран аппаратный кодер JPEG, реализация которого в БМК занимает следующее количество ресурсов:

Туре	Instances	Area	Area %
sequential inverter buffer logic physical_c	37258 60255 5772 190529 3 ells 0	260806 60255.0 5772.00 98476.0 0.000	.000 36.0 00 8.3 0 0.8 000 54.9 0 0.0
total	293814 72	25309.0	00 100.0

Целевая рабочая частота определена в 100МГц; после логического синтеза временные ограничения были удовлетворены с запасом:

Clock Period _____ clk 10000.0 Cost Critical Violating Group Path Slack TNS Paths -----No paths 0.0 clk default No paths 0.0 I2O No paths 0.0 I2S 6997.9 0.0 0 S2O 9057.0 0.0 0 S2S 4078.0 0.0 0 _____ 0 Total 0.0

Для полной матрицы БМК (примерно 15М БЦЯ) данный проект достаточно мал и даёт около 4% плотности заполнения, поэтому под него пропорционально уменьшалось поле БЦЯ БМК, чтобы получить плотности заполнения примерно 20%, 40%, 57% и 65%. Физический синтез проводился для целевой тактовой частоты 10 МГц, чтобы исключить влияние временных ограничений на количество задействованных трассировочных слоёв.

Первой точкой выбран размер поля БЦЯ БМК, для которого данный проект даст примерно 20% плотности заполнения, и установлено ограничение на используемые слои трассировки – с 1-го по 3-й металл.

Для таких ограничений размещение и трассировка проекта прошли успешно, ниже приводится статистика по использованию трассировочных ресурсов и цепям проекта.

Использование трассировочных ресурсов проекта кодера JPEG:

#	Routing	#Avail	#Track	#Total	%Gcell		
# Layer	Directi	on Trac	k Block	ked Gee	ell Blocked		
# # M1	Н	23897	0	1429220	46.06%		
# M2	V	23916	0	1429220	0.00%		
# M3	Н	23897	0	1429220	0.00%		
" # Total Статисти	ка по цег	71710 1ям прое	0.00% кта кодер	4287660 ba JPEG:	15.35%		
Total M1 Wire Length = 1038759.01 Total M2 Wire Length = 12553246.31 Total M2 Wire Length = 11200200.41							
Total M4	Wire Len	gth =	0.00	L			
Total M5	Wire Len	igth =	0.00				
Total M6	Wire Len	igth =	0.00				
Total M7	Wire Len	ngth =	0.00				
Total Wire Length $= 24801395.73$							



Рисунок 5.8 – Трассировка проекта в 3-х металлах при начальной плотности заполнения 20% (металл 2 – красный, металл 3 – зелёный)

Стоит отметить, что итоговая плотность заполнения выросла с 20% до 38%, что связано с устранением нарушений setup/hold в проекте после построения тактового дерева, свидетельствуя о достаточной жёсткости заданных ограничений на использование 3-х слоёв металлизации для трассировки.

Второй точкой выбран размер поля БЦЯ БМК, для которого данный проект даст примерно 40% плотности заполнения, при тех же ограничениях на используемые слои трассировки – с 1-го по 3-й металл (рисунок 5.9).

Для таких ограничений размещение и трассировка проекта прошли успешно.



Рисунок 5.9 – Трассировка проекта в 3-х металлах при начальной плотности заполнения 40% (металл 2 – красный, металл 3 – зелёный)

Ниже приводится статистика по использованию трассировочных ресурсов и цепям проекта.

Использование трассировочных ресурсов проекта кодера JPEG:

#		Routing	#Avail	#Track	#Tota	l %Gcell		
#	Layer	Direction	on Track	Block	ked G	cell Blocked		
#								
#	M1	Н	12414	0	742716	77.57%		
#	M2	V	23916	0	742716	0.00%		
#	M3	Н	12414	0	742716	0.00%		
#								
#	Total	2	48744	0.00%	222814	8 25.86%		
Ст	Статистика по цепям проекта кодера JPEG:							
Total M1 Wire Length = 59409571								
Total M2 Wire Length = 1402738230								
Total M3 Wire Length = $12439229 57$								
Total M4 Wire Length = 0.00								
Total M5 Wire Length $= 0.00$								
Тс	otal M6	Wire Len	gth =	0.00				
Τc	otal M7	Wire Len	gth =	0.00				
Total Wire Length $= 27060707.59$								

Стоит отметить, что итоговая плотность заполнения выросла с 40% до 68%, что связано с устранением нарушений setup/hold в проекте после построения тактового дерева, свидетельствуя о достаточной жёсткости заданных ограничений на использование 3-х слоёв металлизации для трассировки.

Третьей точкой выбран размер поля БЦЯ БМК, для которого данный проект даст примерно 57% плотности заполнения, при тех же ограничениях на используемые слои трассировки – с 1-го по 3-й металл.

Для таких ограничений размещение и трассировка проекта не прошли успешно: плотность заполнения поля БМК выросла с 57% до 92% в результате чего появилось 352 неустранимых DRC нарушения (рисунок 5.10).

#Total number of DRC violations = 352 #Total number of net violated process antenna rule = 0 #Total number of violations on LAYER M1 = 27 #Total number of violations on LAYER M2 = 261 #Total number of violations on LAYER M3 = 64 #Total number of violations on LAYER M4 = 0 #Total number of violations on LAYER M5 = 0 #Total number of violations on LAYER M6 = 0 #Total number of violations on LAYER M7 = 0



Рисунок 5.10 – Трассировка проекта в 3-х металлах при начальной плотности заполнения 57% (металл 2 – красный, металл 3 – зелёный, белыми крестами отмечены DRC нарушения)

Поэтому для 57% плотности заполнения матрицы БМК ограничение на количество трассировочных слоёв было ослаблено на 1 слой – разрешено использовать металлические слои с 1-го по 4-й (рисунок 5.11).

Для таких ограничений размещение и трассировка проекта прошли успешно.



Рисунок 5.11 – Трассировка проекта в 4-х металлах при начальной плотности заполнения 57% (металл 2 – красный, металл 3 – зелёный, металл 4 – жёлтый)

Ниже приводится статистика по использованию трассировочных ресурсов и цепям проекта.

Использование трассировочных ресурсов проекта кодера JPEG:

#		Routing	, #Avail	#Track	k #Total	%Gcell
#	Layer	Direct	ion Track	Bloc	ked Gc	ell Blocked
#						
#	M1	Н	8746	0	929302	93.87%
#	M2	V	23916	0	929302	0.02%
#	M3	Н	8746	0	929302	0.00%
#	M4	V	23916	0	929302	0.00%
#						
#	Total		65324	0.00%	3717208	8 23.47%
C	гатисти	ка по це	пям проек	та коде	pa JPEG:	
		,	1		1	

Total M1 Wire Length = 342369.93Total M2 Wire Length = 11490437.06Total M3 Wire Length = 20153175.09Total M4 Wire Length = 10094248.64Total M5 Wire Length = 0.00Total M6 Wire Length = 0.00Total M7 Wire Length = 0.00

Total Wire Length = 42080230.73

Стоит отметить, что итоговая плотность заполнения выросла с 57% до 92%, что связано с устранением нарушений setup/hold в проекте после построения тактового дерева.

Четвёртой точкой выбран размер поля БЦЯ БМК, для которого данный проект даст примерно 65% плотности заполнения, и установлено ограничение на используемые слои трассировки – с 1-го по 5-й металл (рисунок 5.12).

Для таких ограничений размещение и трассировка проекта прошли успешно.



Рисунок 5.12 – Трассировка проекта в 5-х металлах при начальной плотности заполнения 65% (металл 2 – красный, металл 3 – зелёный, металл 4 – жёлтый, металл 5 – тёмно-красный)

Ниже приводится статистика по использованию трассировочных ресурсов и цепям проекта.

Использование трассировочных ресурсов проекта кодера JPEG:

#		Routing	#Avail	#Track	x #Total	l %Gcell	
#	Layer	Directi	ion Track	Bloc	ked Go	cell Blocke	ed
#							
#	M1	Н	7730	0	820910	95.63%	
#	M2	V	23916	0	820910	0.02%	
#	M3	Н	7730	0	820910	0.00%	
#	M4	V	23916	0	820910	0.00%	
#	M5	Н	7730	0	820910	0.00%	
#							
Total 71022 0.00% 4104550 19.13% Статистика по цепям проекта кодера JPEG:

Total M1 Wire Length = 361928.23Total M2 Wire Length = 10596280.87Total M3 Wire Length = 15464354.08Total M4 Wire Length = 8540908.88Total M5 Wire Length = 6717018.28Total M6 Wire Length = 0.00Total M7 Wire Length = 0.00Total M7 Wire Length = 0.00

Стоит отметить, что итоговая плотность заполнения выросла с 65% до 95%, что связано с устранением нарушений setup/hold в проекте после построения тактового дерева.

На рисунке 5.13 приведён график зависимости необходимого количества слоёв трассировки от начальной плотности заполнения матрицы БМК.



Рисунок 5.13 – График зависимости количества необходимых трассировочных слоёв от плотности заполнения матрицы БМК

Из графика видно, что при малом проценте начального заполнения обойтись укороченным набором 50%) возможно матрицы (меньше трассировочных слоёв, вплоть до минимального количества – 3. При начальном заполнении матрицы более 50% на трассировку проекта, скорее всего, придётся использовать либо все доступные трассировочные слои (для технологических процессов с нормами до 90 нм включительно), либо не менее 5 слоёв металлизации (для технологий с нормами 65 нм и менее). Также с увеличением плотности заполнения матрицы БМК САПР автоматического размещения И трассировки становится сложнее удовлетворять заданные временные ограничения, и если они установлены близкими к предельно-допустимым значениям, то на их реализацию потребуется больше доступных трассировочных слоёв.

5.2. Методы разработки аналоговых и аналого-цифровых конфигурируемых интегральных схем

5.2.1. Принципы построения аналоговых ядер и на их основе аналоговых и аналого-цифровых конфигурируемых интегральных схем. Декомпозиция и композиция аналоговых СФ блоков

Большинство аналоговых блоков построены на основе конкретных технических идей, реализованных В заданном элементном базисе Использование определенной технологии. этого факта позволяет систематизировать блоки с целью ИХ классификации, описания, характеризации и аттестации. Известные определения СФ блоков и IP – ядер, фактически, являются технико-организационными описаниями схем и не раскрывают их техническую суть. В работе предложен новый термин – аналоговое (цифровое) ядро, расширяющий понятие СФ блока, которое определяется технически реализованная система принципов как формирования и обработки аналогового сигнала.

Аналоговое ядро – система взаимосвязанных принципов формирования И обработки аналогового сигнала, реализующая функциональное предназначение и характеристики устройств данного типа. Т.е. это принципы, на основе которых построен блок с заданными характеристиками. иерархическая структура, состоящая Аналоговое ядро – ИЗ других аналоговых ядер или элементов. Любое аналоговое ядро можно разложить до отдельных определенная конфигурация элементов, но только ИХ обеспечивает уникальные свойства аналогового ядра.

Создание интегральной микросхемы обычно начинается с технического задания, которое включает в себя требуемые характеристики, режимы работы и т.д. Решением одной и той же задачи может служить набор аналоговых ядер с различными свойствами (рисунок 5.14).





В качестве примера рассмотрим источник опорного напряжения (ИОН) равный ширине запрещенной зоны. Этот тип ИОН использует принцип компенсации отрицательного температурного дрейфа опорного диода напряжением с положительным температурным дрейфом, которое вырабатывается специальным блоком – РТАТ генератором (рисунок 5.15).



Рисунок 5.15 - Принципиальная схема ИОН

Используя один и тот же принцип компенсации дрейфа, но различную архитектуру и элементный базис, можно получить ИОН на транзисторах в режиме слабой инверсии (рисунок 5.16а) и ИОН на операционном усилителе (рисунок 5.16б), которые будут различными аналоговыми ядрами.



Рисунок 5.16 - ИОН на транзисторах в режиме слабой инверсии (a) и ИОН на операционном усилителе (б)

Важным вопросом при разработке интегральных схем является их аттестация для различных технологий. Обычно для этого необходимо заново рассчитывать и тестировать изделие по всем технологическим нормам, чего можно избежать, используя понятия аналогового ядра, т.к. необходимо проверять правильную работу только тех принципов в составе ядра, которые зависят от технологии. Введенные понятия аналогового ядра и рассмотренные принципы систематизации позволяют:

- 1. Проводить классификацию и оценку существующих и новых конструктивных решений.
- 2. Выделять перспективные ядра, мало зависящие от изменения технологии.
- Описывать, в том числе на уровне макромоделей, аналоговые ядра, используя иерархию их построения. Использовать описания для моделирования схем.
- Автоматизировать пути построения блоков с заданными свойствами.
- 5. Автоматизировать разработку библиотек.
- 6. Проведение упрощенной аттестации спроектированного изделия.
- Систематизировать поток научно-технической информации, выделяя лишь информацию о новых аналоговых ядрах.

Для эффективного использования понятия аналогового ядра необходимо на основе общедоступных источников сформировать существующее на сегодняшний день множество принципов формирования и обработки аналогового сигнала и классифицировать его по степени Следующим шагом зависимости от технологии. является построение иерархического набора аналоговых ядер, что позволяет выбирать оптимальные решения и блоки для конфигурируемых интегральных схем специального назначения и проводить ускоренную их аттестацию.

Описание ядра

В качестве основного систематизирующего элемента в области разработки аналоговых схем предлагается использовать понятие аналогового ядра.

Аналоговое ядро – технически реализованная система принципов формирования и обработки аналогового сигнала [116] – [118].

Каждое ядро имеет свое назначение, которое является целью его образования. Используемая в ядре система принципов обработки сигнала реализует назначение ядра.

Ядро – это структура, содержащая другие ядра и приборы. Техническая реализация системы принципов в ядре заключается в подборе составляющих его ядер и приборов, и в организации взаимодействия между ними.

Приборы и ядра, входящие в состав данного ядра, образуют его базис. Приборы в аналоговых ядрах представляют собой абстрагированные модели интегральных элементов (транзисторов, резисторов, конденсаторов и т.д.), которые предоставляются выбранной технологией производства ИС. Одному и тому же интегральному элементу могут соответствовать разные приборы ядра, которые различаются по степени детализации модели и использованию ее свойств. Прибор является неделимой составляющей ядра. При этом, прибор, не имея внутренней структуры, описывается множеством атрибутов, являющимся подмножеством множества атрибутов ядра.

Ядро должно представлять реализуемые принципы в наиболее общем виде. Оно использует в своем составе ядра и приборы с наименьшей из возможных для реализации данных принципов и условий применения степенью детализации. Образование новых ядер, которые уточняют, детализируют существующие ядра, сопровождается изменением условий применения ядер, которые в данном случае становятся более узкими. При этом в рамках нового ядра должно быть обеспечено условие применимости каждого из ядер и приборов, входящих в его состав. Если условие применимости составляющих его ядер и приборов выходят за рамки образуемого ядра, то они транслируются в условия применимости образуемого ядра.

Таким образом, ядро может быть охарактеризовано:

- 1. Назначением (целью применения);
- 2. Принципом формирования свойств назначения (способом организации взаимодействия составляющих его частей (ядер и приборов));

- Составом (перечень ядер и приборов, используемых при построении ядра);
- 4. Условиями применимости (в том числе постусловиями).

Выбранная структура описания ядра позволяет получать на его основе фреймы и продукции, которые широко используются при построении баз знаний [119] – [123].

Ядрам присущи многочисленные способы построения иерархии, в которых реализуются различные механизмы исследования свойств наряду с образованием новых.

Ядро, как технически реализованная система, обладает некоторым набором входных и выходных переменных.

Классификация приборов

Приборы также, как и ядра можно классифицировать по назначению, принципу и условию применения. Классификация по составу невозможна изза отсутствия структуры у прибора. Под отсутствием внутренней структуры подразумевается, что прибор рассматривается как единое целое. Однако это не означает, что свойства прибора не могут быть представлены в виде эквивалентной схемы, содержащей множество элементов.

Один и тот же физический элемент может иметь представление в виде целого ряда приборов, которые могут существенно отличатся на разных уровнях абстракции в зависимости от назначения, используемого принципа и условий применения (рисунок 5.17).



Рисунок 5.17 - Фрагмент онтологии приборов

Так, в частности, МОП транзистор на одном из самых высоких уровней абстракции может быть представлен приборами в виде ключа или источника тока, управляемого напряжением в зависимости от того какие режимы работы транзистора предполагается использовать.

Одному и тому же прибору на высоко уровне абстракции могут соответствовать разные приборы на более низком уровне. Так, например, резистор может быть поликремневым, диффузионным, карманным или металлическим, а ключу могут соответствовать биполярные транзисторы, полевые транзисторы, электромагнитные реле и т.д.

Классификация ядер

Классифицировать ядра можно по каждой из характеристик, которая отличает одно ядро от другого (назначение, принцип, состав, условие). При этом оказывается удобным представление каждой составляющей описания ядра в виде онтологии.

Одной из основных задач классификации ядер является запись в компактной форме всех известных свойств конкретных ядер. Это становится

возможным благодаря широкому использованию механизма наследования свойств при построении иерархии. При этом все ядра, которые являются подклассами некоторого общего класса, обладают всеми свойствами общего класса (наследование свойств). Таким образом, нет необходимости перечислять все свойства ядра, достаточно точно указать положение ядра в классифицирующей иерархии.

Пример описания ядер компаратора напряжения и компаратора тока (в общем виде) с привлечением онтологий назначения и состава приведен на рисунке 5.18.

Аналоговые схемы осуществляют обработку сигналов. Это дает основание предложить построение онтологии назначений в терминах сигналов и операций над ними. В верхней части рисунка 5.18 представлен фрагмент возможной онтологии назначений. В нижней части фрагмент возможной онтологии по составу. Положение ядер компараторов напряжения и тока выделено утолщенной рамкой.



Рисунок 5.18 - Пример возможного описания ядер компаратора напряжения и компаратора тока (онтологии назначения и состава)

Используя принципы наследования в рамках онтологии и ядер в целом можно заключить, что ядро 1 (рисунок 5.18), являясь компаратором напряжения, осуществляет операцию сравнения напряжения. При этом имеет аналоговый вход и цифровой выход. В отличие от общего ядра компаратора напряжения, вход у ядра 1 – дифференциальный. При этом ядро 1 является общим для всех компараторов напряжения с дифференциальным входом.



Рисунок 5.19 - Пример возможного описания ядер компаратора напряжения и компаратора тока

В общем случае в описании ядер необходимо учесть онтологии принципов формирования и условий применения (например, рисунок 5.19). Тогда ядро 1 (рисунок 5.18) имея общее ядро «компаратор напряжения» (на рисунке 5.18 и рисунке 5.19), является низковольтным, с коммерческим температурным диапазоном, использует для преобразования разности усиление с положительной обратной сигналов связью, при ЭТОМ формирование разностного сигнала осуществляется управлении при дифференциальной парой. Аналогичные рассуждения можно провести и для ядра 2.

Аттестация блоков на основе аналоговых ядер

Аттестуемый аналоговый блок, используя определенные принципы обработки сигнала, обладая назначением, составом и условиями применения,

может быть описан в форме аналогового ядра. Анализ полученного ядра позволяет составить порядок аттестации блока.

Прежде всего необходимо проанализировать принципы, которые легли в основу построения связей между входящим в состав ядра приборами и ядрами.

Те свойства ядра, которые основаны на общих принципах, пошедших аттестацию в составе других ядер, не нуждаются в дополнительной проверке.

Далее необходимо проверить выполнение условий применения для каждого из ядер входящего аттестуемое ядро. Если условия применения ядер выполнены, а сами эти ядра аттестованы ранее, то свойства блока, связанные с наличием данных ядер, не нуждаются в дополнительной проверке.

Если использованные ядра имеют иерархическую структуру, то проверку условий применимости следует провести вплоть до ядер самого общего уровня.

В условиях применимости ядра одним из ключевых моментов является наличие требуемых приборов. При аттестации ядра в первую очередь необходимо обращать внимание на ядра, использующие приборы с наиболее детальным представлением, в которых используются специфические свойства интегральных элементов.

Выполнение условия применимости для данного ядра будет, в том числе, означать соответствие характеристик интегральных элементов требованиям к приборам ядра.

Таким образом, при аттестации аналоговых блоков на основе аналоговых ядер необходимо:

- 1. Поверить свойства блоков, связанные с наличием новых, не аттестованных ранее принципов обработки сигнала;
- 2. Проверить условия применимости ядра аттестуемого блока и ядер, входящих в его состав.

Таким образом, при аттестации блоков на основе аналоговых ядер, основной акцент делается на проверке новых идей и принципов, заложенных

в ядре блока. Верификация прочих функций блока будет обеспечена выполнением условий применимости использованных ядер и приборов.

Чем более общие принципы, ядра и приборы использованы при разработке блока, тем легче выполняются условия применимости и выше вероятность того, что данные принципы уже аттестованы.

Следует заметить, что аттестация блоков как схемотехнического и топологического решения, ни в какой мере не подменяет полного функционального контроля, необходимого при проверке конкретных кристаллов микросхем в ходе производства.

Декомпозиция и композиция аналоговых СФ – блоков.

В качествах примера применения понятий аналоговых ядер и их онтологий к сквозному иерархическому подходу была проведена разработка источника опорного напряжения с кусочной аппроксимацией на уровне макроблока и на уровне поведенческих блоков низкого и высокого уровней [124]. Блок высокого уровня применяется в основном на самом верхнем уровне иерархии системы на кристалле, включающей и аналоговые, и цифровые блоки, тогда как декомпозированный блок, состоящий из макромоделей более низкого уровня, может применяться для моделирования и исследования только аналоговой части системы. Данный подход позволяет формализовать и упростить процесс разработки схем смешанного сигнала, для этого на начальном этапе разработки системы архитектор кристалла формирует и разрабатывает общую структуру системы, состоящую из аналоговых и цифровых блоков высокого уровня абстракции. Далее, разработчики конкретных блоков сразу получают представление о требованиях к своим блокам и о требованиях к сигналам межсоединений. Например, разработчик цифровой части конфигурируемой схемы может использовать аналоговые макроблоки при разработке и моделировании логических блоков, при этом уменьшается вероятность применения им неверной полярности или уровней входных, выходных, тактирующих,

управляющих и прочих сигналов, которые используются и для аналоговой, и для цифровой частей.

Схема источника предназначена для формирования точного опорного напряжения с малой зависимостью от температуры, составляющей не более 10 ррт. Упрощенная электрическая схема источника опорного напряжения показана рисунке 5.20. Данную схему можно разделить на четыре основных функциональных блока: генератора тока РТАТ, генератор тока СТАТ, генератор кусочно-линейного тока, и выходной сумматор. В ходе разработки сложно функциональной микросхемы, в состав которого входит данный источник, для упрощения разработки и расчетов, источник напряжения возможно представить на различных уровнях абстракции, в зависимости от глубины иерархии схемы верхнего уровня, котором, на например, проводится моделирование.



Рисунок 5.20 - Электрическая схема источника опорного напряжения

Источник опорного напряжения можно представить в виде единого макроблока, в виде набора из нескольких макроблоков, полученных в результате декомпозиции схемы, либо на транзисторном уровне. Выходное напряжение в случае применения единого макроблока задается как кусочная функция следующего вида [125]: $V_{REF(1)} = I_{PTAT} K_{P1}(R_1 + R_2) + I_{CTAT} K_{C1}(R_1 + R_2 + R_3) \operatorname{при} K_T I_{CTAT} \ge I_{PTAT}$ $V_{REF(2)} = I_{PTAT} [K_{P1}(R_1 + R_2) + K_{P2}R_1]$

 $+ I_{CTAT}[K_{C1}(R_1 + R_2 + R_3) - K_{C2}R_1]$ при $K_T I_{CTAT} < I_{PTAT}$

При этом коэффициенты *K*_{P1}, *K*_{P2}, *K*_{C1}, *K*_{C2} можно найти при помощи соотношений:

$$\begin{split} K_{CTAT(1)} &= K_{C1}(R_1 + R_2 + R_3); K_{CTAT(2)} = K_{C1}(R_1 + R_2 + R_3) - K_{C2}R_1; \\ K_{PTAT(1)} &= \frac{K_{P1}(R_1 + R_2)(\mu + \xi) - \xi}{K_{CTAT(1)}} - \mu; \\ K_{PTAT(1)} &= \frac{[K_{P1}(R_1 + R_2) + K_{P2}R_1](\mu + \xi) - \xi}{K_{CTAT(2)}} - \mu. \end{split}$$

Где *К*_{*СТАТ*} и *К*_{*РТАТ*} обозначают коэффициенты усиления токов СТАТ- и РТАТ- генераторов.

Единый макроблок возможно путем декомпозиции разделить на несколько блоков более низкого уровня: РТАТ и СТАТ генераторы, генератор нелинейного тока и сумматор, как было показано выше. Каждый макроблок низкого уровня может формировать нелинейную зависимость напряжения от температуры, выходного тока или форма которой соответствует SPICE моделям. Применение макромоделей низкого уровня позволяет устранить необходимость применения в моделировании схем блоков. Моделирование смещения аналоговых схем, описанных макромоделями, проводится значительно быстрее, чем схем с SPICE моделями, в особенности, если в составе схем находятся операционные усилители. Далее приведен пример кода макромодели генератора кусочнолинейного тока. Из примера видно, что он значительно компактнее и в нем применяется меньше параметров, чем у схемы с SPICE моделями субмикронных КМОП транзисторов:

analog begin

iptat=(I(PTAT_IN)); ictat=(I(CTAT_IN)); diff=abs(ictat)-abs(iptat); end.

Для разработки макромоделей был проведен анализ каждого из составных блоков источника опорного напряжения для возможности макромоделей, создания адекватно представляющих эти блоки. Электрическая схема генератора тока РТАТ показана на рисунке 5.21. Схема представляет собой бета-умножитель, в котором транзистор TR_2 находится в подпороговом режиме за счет нахождения в его истоке резистора R, при этом на затвор этого транзистора поступает напряжение порога и превышение над порогом р-канального транзистора. При увеличении температуры пороги всех транзисторов уменьшаются, при этом уменьшение порога у TR_2 сильнее влияет на ток стока, чем уменьшение подвижности этого транзистора за счет нахождения его в подпороговом режиме. За счет этого увеличивается падение напряжение на резисторе R, что увеличивает токи в каждой из ветвей с увеличением температуры. Таким образом, выражение для выходного тока макромодели данного блока должны содержать формулы для падения напряжения на резисторе R, рассчитанное через зависящее от температуры превышение над порогом транзистора TR₂.



Рисунок 5.21 - Электрическая схема генератора РТАТ тока

Электрическая схема генератора тока РТАТ показана на рисунке 5.22. Операционный усилитель в составе схемы уравнивает потенциалы на обоих своих входах. Таким образом, на инвертирующем входе ОУ образуется напряжение, равное напряжению анод-катод диода D, имеющее СТАТ зависимость. Следовательно, падение напряжение на R₁ и ток через этот резистор также будут иметь СТАТ зависимость. Выражение для выходного тока макромодели должен содержать компактный блок операционного усилителя и выражение для напряжения анод-катод диода при заданном токе смещения, зависящее от температуры.



Рисунок 5.22 - Электрическая схема генератора СТАТ тока

Электрическая схема однокаскадного операционного усилителя (ОУ) показана на рисунке 5.23. Усилитель в данной схеме работает только на емкостную нагрузку, следовательно, добавлять второй каскад не следует.

Компактная модель может содержать только выражение для выходного напряжения, выраженное через разность входных сигналов и коэффициент усиления. Для оценки устойчивости петли обратно связи, образуемой данным ОУ и п-транзистором на его выходе, возможно добавление низкочастотного полюса в передаточную характеристику макромодели. На рисунках 5.24 и 5.25 показаны схемы кусочно-линейного тока и выходного сумматора. Соответствующий макромодели будут содержать только арифметические действия с входными токами и преобразование их в выходное напряжение.



Рисунок 5.23 - Электрическая схема однокаскадного операционного усилителя



Рисунок 5.24 - Электрическая схема генератора кусочно-линейного тока



Рисунок 5.25 - Электрическая схема выходного сумматора

Выходное напряжение получается путем суммирования всех представленных токов при помощи макромодели сумматора. Код макромодели сумматора:

```
analog begin
for (counter = 1; counter < 4; counter = counter + 1)
begin
case (counter)
1: r=r1;
2: r=r2;
3: r=r3;
Endcase
if (counter = 1)
begin
V(R1P, R1N) <+ I(R1P, R1N)*rtot;
end
else
begin
if (counter == 2)
begin
V(R2P, R2N) <+ I(R2P, R2N)*rtot;
end
else
begin
if (counter == 3)
begin
```

V(R3P, R3N) <+ I(R3P, R3N)*rtot; end end end end end

Зависимость выходного напряжения состоит из двух частей. В левой части это напряжение получается путем суммирования РТАТ и СТАТ токов, оба тока имеют нелинейную зависимость от температуры, включая зависимости второго и третьего порядка, следовательно, их сумма при равенстве коэффициентов первого порядка будет также нелинейной. Далее, РТАТ ток сильнее зависит от температуры в правой части графика, следовательно, для его компенсации добавляется дополнительный СТАТ ток, формируемый генератором кусочно-линейного тока.

Моделирование и внедрение разработанной модели источника в иерархию более высокого уровня возможно произвести как при помощи графических средств проектирования, например, Virtuoso Schematic (рисунок 5.26), так и при помощи создания общей текстовой модели, включающей макромодели более низкого уровня. Для моделирования макроблока необходимо Spectre создать описание моделирования, схемы для включающее либо блоки по отдельности, либо макромодель источника в целом.



Рисунок 5.26 - Графическая схема источника опорного блока с макромоделями

На рисунке 5.27 представлены зависимости выходных токов и напряжений разработанной модели источника опорного напряжения от температуры.



Рисунок 5.27 Зависимости выходных токов и напряжений макроблоков от температуры

5.2.2. Упрощенные эквивалентные схемы резисторов на переключаемых конденсаторах в z-области при включении их между виртуальной землей и источником сигнала

Одним из часто используемых узлов аналого-цифровых конфигурируемых интегральных схем являются элементы, обеспечивающие электрическое сопротивление за счет переключения конденсаторов (резисторы на ПК). Для анализа таких схем используется сложный

математический аппарат, учитывающий протекание заряда конденсаторов в цепях схемы на различных фазах переключения.

В работе предложено в качестве эквивалентной схемы резистора, выполненного на переключаемых конденсаторах, в *z*-области использовать модель протекания заряда между двумя узлами электрической схемы, находящимися под различными потенциалами.

Эквивалентная схема упрощается, если ПК резистор подключен к виртуальной земле входа ОУ или к источнику сигнала. Использование предложенных эквивалентных схем позволяет проводить расчет характеристик аналоговых схем, не применяя сложный временной анализ.

Эквивалентные резисторам схемы на ПК выполняются на основе ключей, конденсаторов И управляемых вложенными импульсами. Последовательность управляющих вложенных импульсов приведена на рисунке 5.28, схема формирования таких импульсов приведена на рисунке 5.29, основные эквивалентные схемы резисторов приведены в таблице 5.2. В приложении 2 приведены характеристики типовых аналоговых схем с моделей резисторов применением предложенных на переключаемых конденсаторах. В приложении используются следующие сокращения

$$H^{xy} \equiv H^{xy}(z)$$
, $xy \equiv 0e; 00; e0; ee$,

где xy— нечетный (0) или четный (e) полутакт выборки входного (x) и выходного (y) сигналов.



Рисунок 5.28 - Последовательность управляющих импульсов



Рисунок 5.29 - Схема формирования вложенных импульсов

Таблица 5.2 - ПК эквиваленты резисторов

Наименование	Схема электрическая	Эквивалентное сопротивление	Упрощенная эквивалентная схема в <i>z</i> области при включении ПК между виртуальной землей ОУ и источником напряжения
1	2	3	4
Параллельный ПК		$R = \frac{T}{C}$	
Последовательный ПК		$R = \frac{T}{C}$	U_{r}
Последовательно– параллельный ПК	$U_{1} \xrightarrow{\varphi_{1}} U_{2}$	$R = \frac{T}{2C}$	$ \begin{array}{c} $
Билинейный ПК		$R = \frac{T}{4C}$	$ \begin{array}{c} $
Инвертирующий ПК		$R = -\frac{T}{C}$	$U_{r} \qquad U_{z}$
Следящий ПК (последовательный)		$R = \frac{T}{C}$	U, U,

C(1-z¹) U, U.

5.2.3. Особенности схем заказчиков различных классов, влияющие на архитектуру и характеристики базовых кристаллов конфигурируемых аналоговых и аналого-цифровых интегральных схем

Аналоговые и аналого-цифровые схемы заказчиков, конфигурируемые на основе ряда АЦ БМК, можно разделить на несколько классов (таблица 5.3): схемы АЦП и ЦАП, схемы управления питанием (вкл. регуляторы и преобразователи напряжения), операционные И инструментальные усилители, компараторы, аналоговые ключи и мультиплексоры, схемы управления, ШИМ модуляторы, схемы телеметрии и обработки сигналов температуры, датчиков, датчики универсальные схемы, усилители, микроконтроллеры и др.

Для различных классов микросхем требуется наличие в составе АЦ БМК следующих аналоговых и аналого-цифровых СФ блоков: ИОН, АЦП, ЦАП, ΟУ, ключей (мультиплексоров), регуляторов напряжения, компараторов, усилителей, схем детектирования и сброса и др. Для ряда задач требуются также блоки цифровых интерфейсов (типа SPI), блоки однократно-программируемой памяти (типа ОТР), блоки генераторов (RC, ФАПЧ), датчики температуры, блоки цифрового кварцевых, блоки управления, включая несложные микропроцессоры и т.д.

Анализ большого количества схем заказчиков (более 100) показал, что ИОН используются не менее чем в 50% схем, ОУ – 45%, АЦП/ЦАП – 30%, ключи/мультиплексоры – 30%, регуляторы – 30%, блоки цифрового интерфейса типа SPI – 25%, генераторы – 20%, детекторы питания – 15%, блоки ОТР памяти – 15%. Для многих схем требуется наличие блоков несложной цифровой логики.

Помимо разнообразия аналоговых блоков для удовлетворения потребностей заказчиков необходимо предусмотреть в базовых кристаллах различные номиналы питающих напряжений. Наиболее востребованным является ряд напряжений питания аналоговых частей: 1.8 В, 3.3 В, 5.0В, ± 15В (30В). (Например, стандартным напряжением питания аппаратуры космического назначения является 27 В). Требуются также малогабаритные корпуса и схемы с микро-потреблением.

Класс схем	Схема	Особенности	Основные блоки
АЦП/ЦАП	Микросхема аналого- цифрового преобразователя с SPI интерфейсом.	Полузаказная микросхема представляет собой двух канальный АЦП последовательного приближения с масштабируемыми входными уровнями. Микросхема выполнена на основе аналого- цифрового базового матричного кристалла (АЦ БМК) 5400ТР045А. Микросхема предназначена для преобразование	 источник опорного напряжения отключаемый блок умножения частоты 12-ти разрядный аналого- цифровой преобразователь (АЦП) последовательного
		входного аналогового сигнала в 12-ти разрядный цифровой код и выдачу этого кода через последовательны интерфейс.	приолижения – 4) 12-ти разрядный регистр выходных данных 5) схема масштабирования входного сигнала
	Микросхема сигма-дельта АЦП электронного модуля измерения	Полузаказная микросхема на базе 5400ТР015 предназначена для применения в составе датчика влажности-температуры и представляет собой АЦП для применения в системах преобразования изменяемого значения емкости и сопротивления в цифровой код с последующей передачей сигналов на микропроцессор.	 АЦП ИОН ПДОУ (Полностью дифференциальный операционный усилитель) ЦАП
	СБИС обработки аналоговых сигналов на основе конфигурируемой аналого-цифровой матрицы	Микросхема на базе 5400TP015 осуществляет выделение огибающей входных амплитудно- модулированных сигналов с помощью устройств пикового детектирования-хранения (PDH), получения разностных сигналов со смежных каналов с последующей нормализацией для аналого-цифрового преобразования.	 блок цифрового контроллера ИОН ГТЧ (генератор тактовой частоты) Аналоговый мультиплексор- демультиплексор
	Микросхема 14- разрядного аналого- цифрового	Функциональное назначение микросхемы на базе 5400TP015: АЦП имеет расширенные возможности по приему	1) АЦП 2) ЦАП 3) Формирователь тактовых

преобразователя	входных тактовых сигналов. Возможна подача	сигналов
циклического типа	парафазных тактовых сигналов с размахом	4) Блок опорных напряжений
	цифровых КМОП уровней (0 – 3,3 В) и	5) Блок цифровой коррекции
	скважностью 2, LVDS уровней, однофазного	6) OY
	тактового сигнала с размахом в диапазоне от 0,3 В	,
	до 3,3 В со скважностью 2.	
	Возможно использование как встроенного, так и	
	внешнего источника опорного напряжения,	
	значение которого определяет максимальную	
	амплитуду входного сигнала.	
	Выходные данные представлены в	
	прямом/дополнительном коде, имеют цифровые	
	КМОП уровни 0 – 3,3 В. Данные записываются в	
	выходной регистр и могут быть считаны как по	
	фронту, так и по срезу сигнала OUTCLK.	
Микросхема цифро-	Полузаказная микросхема 5400ТР045А-002	1) источник опорного
аналогового	представляет собой двух канальный ЦАП на	напряжения
преобразователя с SPI	основе R-2R матрицы с последовательной или	2) 12-ти разрядный цифро-
интерфейсом.	параллельной загрузкой входных данных.	аналоговый преобразователь
	Микросхема выполнена на основе аналого-	(ЦАП)
	цифрового базового матричного кристалла	3) цифровой мультиплексор
	(АЦ БМК) 5400ТР045А.	4) блок SPI интерфейса
		5) цифровой блок управления
		6) выходной буфер
Микросхема 16-	Микросхема на базе 5400ТР045А представляет	1) ОУ
разрядного ЦАП	собой 16-разрядный цифро-аналоговый	2) Цифровой блок управления и
	преобразователь из бинарного кода в напряжение	последовательный интерфейс
	на основе архитектуры R-2R матрицы,	3) Цифровой блок калибровки
	интегрированный на одном кристалле вместе с	4) ОПП (однократно
	двумя высоковольтными операционными	программируемая память)
	усилителями, с управлением	5) ЦАП
	по последовательному интерфейсу.	
Микросхема	Микросхема на базе 5400ТР045А предназначена	1) Цифровой блок управления

низкопотребляющего	для передачи сигнала	2) ИОН
дельта-сигма цифро-	в форме тока по токовой петле с уровнем сигнала	3) ОУ
аналогового	от 4 мА до 20 мА,	4) ОПП (однократно
преобразователя для	которая поддерживает 4-х проводной	программируемая память)
передачи сигнала в форме	последовательный интерфейс для настройки	5) ЦАП
тока	и загрузки ЦАП	6) Блок SPI интерфейса
по токовой петле с		7) Внутренний частотный
уровнем сигнала от 4 мА		генератор
до 20 мА		
Микросхема 14-	Микросхема на базе 5400ТР065 представляет	1) ЦАП
разрядного ЦАП	собой 16-разрядный цифро-аналоговый	2) ОУ
	преобразователь из бинарного кода в напряжение	3) ОПП (однократно
	на основе архитектуры R-2R матрицы,	программируемая память)
	интегрированный на одном кристалле вместе с	4) цифровой блок управления и
	двумя высоковольтными операционными	коррекций
	усилителями, с управлением	
	по последовательному интерфейсу.	
Многоканальная	Полузаказная микросхема на базе	1) АЦП
микросхема обработки	5400ТР075 предназначена для независимого или	2) ЦАП
аналого-цифровых и	совместного использования многоканальных АЦП	3) ОУ
цифро-аналоговых	и ЦАП. Диапазон преобразуемых сигналов от	4) Линейные регуляторы
сигналов в диапазоне от	минус 10 В до 10 В	напряжения
минус 15 В до 15 В		5) Цифровой блок управления
		6) SPI интерфейс
Цифровой	Микросхема на базе К5400ТР045А предназначена	1) АЦП
преобразователь	для приёма, преобразования и выдачи сигналов	2) ИОН
сигналов	на телеметрическую станцию.	3) ОУ
	Микросхема выполняет следующие функции:	4) Цифровой блок управления
	-приём входного сигнала;	5) ЦАП
	-преобразование полученного сигнала;	
	-формирование полученного сигнала в	
	соответствии с заданным алгоритмом управления;	
	-выдача этого сигнала на телеметрическую	

		станцию.	
	Микросхема	Полузаказная микросхема на базе 5400ТР015	1) Сигма дельта АЦП;
	преобразователя частоты в	предназначена для преобразователя частоты в код	2) источник опорного
	код (преобразователи	с целью применения в составе приборов	напряжения (ИОН);
	емкость-код, напряжение-	автоматики.	3) блок накачки
	частота и частота-		4) встроенный датчик
	напряжение)		температуры;
			5) блок формирования опорных
			уровней
			6) блок формирования сброса
			при подаче питания;
			7) цифровой блок управления
			8) блок алгоритмической
			обработки;
			9) блок памяти;
			10) синхронный
			последовательный интерфейс
			(SPI).
Программиру	Программируемая	Микросхема 5400ТР045А-031 обеспечивает	1) ГТЧ (генератор тактовой
емые	аналоговая микросхема.	функции операционного усилителя, компаратора,	частоты)
универсальны		источника опорного напряжения, линейного	2) ИОН
е микросхемы		регулятора, электрометрического усилителя,	3) Ключи;
		неинвертирующего и инвертирующего усилителя	4) ОПП (однократно
		на операционном усилителе. При этом	программируемая память)
		обеспечивается возможность программирования	5) OY
		коэффициентов усиления, выходного напряжения	6) Линейный регулятор
		источника опорного напряжения и линейного	7) Компараторы
		регулятора, динамических характеристик блоков.	8) Мультиплексоры
Линейные	Линейный стабилизатор	Микросхема на базе 5400ТР045А представляет	1) ИОН
регуляторы	напряжения	собой стабилизатор напряжения, регулируемой	2) Блок настройки памяти
		положительной полярности, созданная для	3) Регулятор
		применения в устройствах, требующих	
		сверхнизкого падения напряжения на	

			регулирующем элементе, для токов до 300 мА.	
		Линейный регулятор	Полузаказные микросхемы 5400ТР085-003-1.2 (1.5,	1) Силовой интегральный ключ
		напряжения с	1.8, 2.5, 3.3, 3.5, 5.0, 9.0, 12.0, 15.0) представляют	2) Усилитель ошибки
		температурной защитой и	собой линейный регулятор напряжения	3) ИОН
		функцией ограничения	положительной полярности. Выходное напряжение	4)Схема защиты от низкого
		тока	настраивается на этапе производства и выбирается	напряжения питания
			из ряда – 1,2 В, 1,5 В, 1,8 В, 2,5 В, 3,3 В, 3,5 В, 5 В,	5) Схема температурной защиты
			9 B, 12 B, 15 B.	6) Схема ограничения тока
				7) Блок управления
Γ	Компараторы	Быстродействующий	Микросхема на базе 5400ТР045А представляет	1) Компаратор
		счетверенный компаратор	собой счетверенный быстродействующий	
			компаратор с rail-to-rail выходом. Каждый	
			компаратор имеет собственные выводы	
			положительного и отрицательного напряжения	
			питания.	
	ОУ	Радиационно-стойкий	Микросхема на базе 5400ТР045А содержит	1) ОУ
		счетверенный	четыре идентичных операционных усилителя	
		операционный усилитель	с программируемым током покоя.	
		общего применения с		
		программируемым током		
		покоя		
		Радиационно-стойкий	Микросхема на базе 5400ТР045А содержит	
		прецизионныи	прецизионныи операционныи усилитель. В режиме	2) Компаратор
		операционныи усилитель.	непрерывнои калибровки смещения нуля	3) ЧСОУ (чоппер
			операционныи усилитель обеспечивает низкии	стабилизированныи
			уровень шума 10 нВ/1 ц0,5, низкое смещение нуля	операционныи усилитель)
			менее 10 мкВ и температурный дрейф менее	
-	<i>V</i> люни		0,5 MKD/ C.	
	коммутатори	Аналоговый		2) Hudpopag Joruka
l	коммутаторы	мультиплексор 32-х	пизковольтным тридцати двухканальным	2) цифровая погика
	и мультиплексо	Капальный. Высокороля так и 32, у	аналоговым мультиплексором с дешифратором.	1) Клюни
l	nLI	изнальный зналоговий		2) Hudpopag Joruka
	րու	Kanandindin ananoi oddin	BBOOROBOIDTIIDIW JZ-A Ranajidiidiwi anajidi Obdiwi	ω_j $\mu_n \psi_{\rho} \partial \partial \alpha_j$ $\beta \partial \beta_n h h h h h h h h h h h h h h h h h h h$

	мультиплексор с функцией «холодный резерв».	мультиплексором с функцией «холодный резерв».	
	Восьмиканальный аналоговый коммутатор	Микросхема на базе 5400ТР054 является восьмиканальный аналоговый коммутатор с дешифратором для коммутации напряжения от минус 15 В до 15 В.	1) Ключи 2) Цифровая логика
	Восьмиканальный аналоговый коммутатор с дешифратором	Микросхема на базе 5400ТР054 является аналоговым коммутатором с дешифратором для коммутации напряжения от минус 15 В до 15 В.	1) Ключи 2) Дешифратор
	Микросхема многоканального цифрового коммутатора	Полузаказная микросхема на базе 5400ТР01Н4 предназначена для многоканальной предварительной обработки входных (коммутированных) сигналов с использованием компараторов напряжения и цифровой обработки.	 Аналоговые компараторы Схема для построения генератора на внешнем кварцевом резонаторе Делители напряжения Мультиплексоры Цифровой блок управления и интерфейсов
	Четырехканальный аналоговый ключ	Микросхем 5400ТР054-001 является четырехканальный аналоговый ключ со схемой управления для коммутации напряжений от минус 15 В до 15 В.	1) Ключи 2) Цифровая логика
	Высоковольтный четырехканальный аналоговый ключ с инверсной схемой управления	Микросхема 5400ТР054-002 представляет собой высоковольтный четырехканальный аналоговый ключ с инверсной схемой управления. ИС осуществляет коммутацию выводов в соответствии с управляющими сигналами на выводах С1, С2, С3, С4. Диапазон коммутируемого напряжения от VSSA + 1,5 B до VDDA – 1,5 B	1) Ключи 2) Цифровая логика
ШИМ- модуляторы, драйверы	Микросхема широтно- импульсного модулятора системы	Микросхема на базе 5400TP015 предназначена для замещения функционально аналогичного модуля на дискретных компонентах, содержит в своем	 Формирователь сигнала сброса Генератор тактовой частоты Счетчик тактовых

силовых	термостатирования	составе схемы, реализующие функции молуля.	ИМПУЛЬСОВ
ключей		встроенные компараторы цифро-аналоговые	4) Счетчик управляющих
транспяторы		преобразователи и управляющую цифровую	сигналов
ировней		преобразователи и управляющую цифровую	
уровней		Логику. Устройство прелизущено иля работы в системах	6) ПАП
		телеметринеских измерений ракетно косминеской	
		толиции	у) гегуляторы внутреннего
			питания
	M	Отечественные аналоги: отсутствуют.	1) &
	Микросхема	Полузаказная микросхема на базе 54001Р015	1) Формирователь сигнала сороса
	четырехканального	предназначена для замещения модуля на	2) Генератор тактовой частоты
	ШИМ-модулятора	дискретных компонентах и использования в	3) Счетчик тактовых
	системы	системах приборной автоматики в качестве ШИМ-	импульсов
	термостатирования	модулятора.	4) Счетчик управляющих
			сигналов
			5) АЦП
			6) ЦАП
			7) Регуляторы внутреннего
			питания
Цифровая	Микросхема транслятора	Полузаказная микросхема на базе 5400ТР125	1) Трансляторы уровней
логика	уровней	предназначена для преобразования входных	2) Цифровая логика
		логических уровней.	
	Микросхема буфера	Полузаказная микросхема на базе 5400ТР125	1) Буфер сигналов
	иифровых сигналов	прелназначена для увеличения нагрузочной	2) Цифровая логика
	· 11	способности вхолного нифрового сигнала.	
Микросхемы	Микросхема	Полузаказная микросхема на базе 5400ТР014	1) встроенный источник
обработки	многоканального опроса	предназначена для опроса резистивных или	опорного тока;
сигналов	датчиков температуры	диодных термодатчиков по четырёхпроводной	2) встроенный понижающий
латчиков		схеме (одновременно до 18 датчиков).	регулятор напряжения:
физических		Основная функция устройства – измерение	3)07
величин		напряжения на резистивных или лиолных латчиках	4) Мультиплексоры:
		тока с послелующим аналого-нифровым	5)ИОН
		преобразованием в 8-ми или 12-разрялный	6) Цифровой блок управления и
		пресоризованием в с ми или 12-разрядный	интерфейсов
		цифровой код.	интерфейсов.

	Микросхема	Микросхема на базе 5400ТР014 представляет	1) Аналоговый мультиплексор
	многоканального опроса	собой СБИС, предназначенную для опроса	напряжения
	датчиков температуры	резистивных и диодных термодатчиков по	2) Усилительный каскад
		четырёхпроводной схеме. Данная микросхема	3) Блок сброса
		предназначена для применения в составе	4) Цифровой блок управления
		различных систем для сбора информации о	5) АЦП
		температуре с целью терморегулирования и	6) ЦАП
		телеметрии.	7) SPI интерфейс
			8) ИОН
			9) Источник тока
			10) Формирователь опорных
			уровней
			11) RC-генератор
			12) Регуляторы напряжения
	Микросхема опроса	Микросхема 5400ТР015-043 построена на основе	1) OY
	шунтовых датчиков тока	комплекта из двух микросхем модуля с	2) ИОН
	(интерфейсная)	гальванической развязкой для опроса 8 шунтовых	3) Регуляторы напряжения
		датчиков. Микросхемы представляют собой две	4) Генератор частоты
		части одной системы:	5) Цифровой блок управления и
		- интерфейсная часть – микросхема 5400ТР015-043	интерфейсов.
		- измерительная часть – микросхема 5400ТР015-	6) Драйвер трансформатора
		044	7) Схема формирования сигнала
		Передача питания от интерфейсной микросхемы к	сброса
		измерительной, а также обмен информацией между	-
		микросхемами осуществляется через	
		трансформаторную развязку. Возможна	
		организация обмена без гальванической развязки.	
Микроконтро	Микроконтроллер	Микросхема 5400ТР105-001 представляет собой	1) ПЗУ
ллер	архитектуры 8051	микроконтроллер с архитектурой ядра 8051, со	2) Микроконтроллер
-	с встроенными блоками	встроенным управлением аналоговых блоков и с	3) Кварцевый генератор
	АЦП и ЦАП	увеличенным количеством универсальных	4) Блок ФАПЧ
		цифровых блоков ввода/вывода.	5) Мультиплексор

			6) ПЗУ настройки7) POR
			8) Супервизор питания
			9) Линейный регулятор
			10) O3Y
			11) RC-генератор
			12) ЦАП
			13) Аналоговый мультиплексор
			14) Термодатчик
			15) АЦП
			16) LDO 3.3 B
			17) ИОН
			18) Буфер
	Сбоеустойчивый	Микросхема на базе 5400ТР105	1) АЦП
	контроллер архитектуры	представляет собой малопотребляющий	2) ИОН
	8051 с напряжением	контроллер с архитектурой ядра 8051 для бортовых	3) МОУ
	питания до 15 В и	систем управления с диапазоном рабочих частот до	4) ОЗУ
	возможностью	8,0 МГц.	5) ПЗУ
	встроенного управления		6) ЦАП
	и конфигурирования		7) ФАПЧ
	аналого-цифровых блоков		8) Микроконтроллер
			9) Супервизор
			10) Мультиплексор
			11) Генератор частоты
			12) Линейные регуляторы
			13) Аналоговый Мультиплексор
Силовые	Драйвер шагового	Полузаказная микросхема на базе 5400ТР085	1) Силовой высоковольтный
схемы	двигателя.	представляет собой драйвер шагового двигателя.	ключ на основе Р-МОП
		Микросхема предназначена для управления	транзистора
		биполярным шаговым двигателем.	2) Силовой высоковольтный
			ключ на основе N-МОП
			транзистора;
			3) Блок управления силовыми

			ключами
			4) Цифровой блок управления
Датчик	Интегральный цифровой	Полузаказные микросхемы 5400ТР125-010-	 Регулятор напряжения
температуры	датчик температуры с	3.3 (5.0) предназначены	2) Компаратор
	однопроводным	для преобразования значения температуры в 16-ти	3) Термоэлемент
	интерфейсом.	разрядный цифровой код	4) OY
		и передачи его управляющему микроконтроллеру	5) Цифровой блок управления
		по однопроводному (1-wire) интерфейсу.	6) Блок памяти
		Напряжение питания: 3,3 В, 5,0 В. Настройка	
		напряжения питания производится на этапе	
		изготовления для каждой микросхемы	
		соответственно.	
Кварцевые	Микросхема	Полузаказная микросхема на базе 5400ТР015	1) Схема для построения
генераторы,	радиационно-стойкого	представляет собой СБИС, включающую набор	генератора на внешнем
буферы	прецизионного генератора	функциональных блоков, предназначенных для	кварцевом резонаторе
тактовой		построения термостабилизированного	2) Делитель частоты
частоты,		радиационно-стойкого прецизионного кварцевого	3) Стабилизатор
преобразоват		генератора опорной частоты.	4) Компараторы
ели уровней,		Микросхема может применяться в	5) ИОН
приемо-		радиоэлектронной аппаратуре специального	6) Цифровой блок управления
передатчики		назначения.	
Специализир	Микросхема обработки	Микросхема на базе 5400ТР015 предназначена для	1) Предварительный регулятор
ованные	входной информации и	обработки входных аналоговых сигналов от	напряжения
схемы	формирования	внешних датчиков и формирования	2) ЙОН
управления	исполнительных команд.	исполнительных команд.	3) Регулятор напряжения
		В состав микросхемы входят регулятор питания,	аналогового питания
		схема начальной установки (сброса) цифровой	4) Регулятор напряжения
		части, два генератора тактовых сигналов для	цифрового питания
		формирования тактовых импульсов разной частоты	5) Компараторы
			6) RC-генератор
			7) Формирователь сигнала сброса
			8) Цифровой блок управления
	Микросхема аналогового	Микросхема на базе 5400ТР014 предназначена для	1) ИУ (инструментальный

интерфейса	выполнения технологических операций на	усилитель)	
робототехнической	внешней поверхности перспективных модулей	2) Регуляторы напряжения	
транспортно-	пилотируемых комплексов и поддержки экипажа	3) АЦП	
манипуляционной	при внекорабельной деятельности.	4) OY	
системы". АЦ БМК		5) ИОН	
изготавливается по		6) Аналоговый коммутатор	
технологии кремний на		7) LVDS приемник и	
изоляторе с проектными		передатчик;	
нормами 0,24 мкм и		8) Цифровой блок управления	
номинальным			
напряжением питания –			
5 В. Микросхема			
выполнена в корпусе			
4238.108-3.			
БИС управления рулевы	м Микросхема на базе 5400ТР015 предназначена для	1) (АЦП);	
приводом	приёма, преобразования и выдачи командных	2) Цифровой фильтр (ЦФ);	
	сигналов на драйвер управления электроприводом.	3) Компаратор;	
		4) Цифровой блок управления	
		5) Мультиплексор (МП);	
		6) Счетчик (СЧ);	
		7) Декодер (ДК);	
		8) Регистр (РГ).	
Микросхема контроля	Полузаказная микросхема на базе 5400ТР015	1) SPI интерфейс	
памяти	предназначена для обеспечения взаимодействия с	2) Модуль управления и	
	микросхемой ПЗУ 1666РЕ014 АЕЯР.431220.981ТУ	обслуживания SPI интерфейса	
	посредством последовательного интерфейса.	3) Модуль стирания микросхемы	
		памяти	
		4) Внутренний коммутатор	
		5) Модуль формирования	
		диаграмм для ПЗУ	
«Микросхема		1) O3Y	
формирования импульс	ов нимерохома формиророния имими сор	2) Блоки ОПЗУ (однократно	
синхронизации и отсче	та	программируемое запоминающее	
	временных интервалов».	синхронизации и отсчета временных интервалов.	устройство)
-----------	--------------------------	--	------------------------------
			3) NOH
			4) Регулятор напряжения
			5) Кварцевый генератор
	Микросхема контроллера	Функциональное назначение микросхемы:	1) Регулятор напряжения;
	переключения каналов	Полузаказная микросхема на базе 5400ТР015	2) OY
		предназначена для контроля текущей ошибки	3) прецизионный операционный
		системы и выработки последовательности команд	усилитель
		переключения с основного на резервный канал	4) Компаратор
		усилителя при возникновении неисправности.	5) Входной цифровой фильтр
			управляющих команд;
Усилители	Микросхема	Микросхема на базе 5400ТР015	1) ОУ
	преобразователя	предназначена для использования в составе	2) Блок модулятора
	усилителя постоянного	нормирующего усилителя в системах сбора	3) Блок демодулятора
	тока (УПТ) с	данных.	4) Цифровой блок управления
	гальванической развязкой	Устройство позволяет преобразовывать входные	
		малые уровни токов и напряжений (относительно	
		высоких уровней) в выходной сигнал постоянного	
		напряжения величиной пригодной для дальнейшей	
		цифровой обработки.	
		Устройство предназначено для работы в	
		информационно-измерительных системах и	
		системах телеметрических измерений ракетно-	
		космической техники.	
		Отечественные аналоги: микросхема 1470ХП4Н2	

5.2.4. Разработка конфигурируемых специализированных аналоговых и аналого-цифровых интегральных схем с оптимальной архитектурой для заданного класса задач

На основе предварительного анализа схем заказчиков аналогоцифровых интегральных схем, приведенного в предыдущем разделе работы, была построена архитектура базового кристалла АЦ БМК, обладающая наилучшими техническими характеристиками для заданного класса задач. аналого-цифрового и цифро-аналогового преобразования, Для задач усилительных и универсальных схем, схем управления, схем телеметрии и обработки необходимо сигналов датчиков обеспечить хорошие динамические и статические параметры АЦП и ЦАП, хорошую точность ИОН и ОУ, малые сопротивления проходных ключей, большую емкость логических элементов. Наилучшие характеристики обеспечивает архитектура АЦ БМК, в которой точные аналоговые элементы и цифровые элементы разнесены по разным сторонам кристалла.

С учетом вышесказанного был разработан базовый кристалл микросхемы АЦ БМК универсального применения. Микросхема имеет блочную архитектуру [126] – [127].

Использованная блочная архитектура характеризуется модульным представлением отдельных функциональных узлов. Узлы содержат в своем составе матрицы ячеек (канальные или бесканальные). В таких кристаллах также могут быть выделены специализированные области – подматрицы для выполнения заранее определенных функций (логической обработки данных, памяти, аналого-цифрового преобразования и др.).

Структура разработанного изделия подразделяется на две функциональные области — цифровую и аналоговую.

Цифровая функциональная область БМК представляет собой матрицу, состоящую из набора 2-х транзисторных базовых цифровых ячеек (БЦЯ).

Базовые цифровые ячейки объединены в строки (ряды), ограниченные с двух сторон шинами земли и питания.

Аналоговая функциональная область представляет собой сложную конструкцию, включающую в себя сложно-функциональные блоки, эквивалентные двух-транзисторные аналоговые ячейки, наборы резисторов и конденсаторов.

В состав библиотечных блоков аналоговой области АЦ БМК входят: ОУ общего применения, прецизионные ОУ, полностью дифференциальные ОУ, компараторы общего назначения, компараторы стробируемые, цифроаналоговые преобразователи, аналого-цифровые преобразователи, источники опорного напряжения, блоки фазовой автоподстройки частоты, аналоговые мультиплексоры, генератор накачки отрицательного смещения на подложку.

В процессе производства изготавливается пластина, содержащая базовый кристалл, то есть нескоммутированную матричную структуру. На заключительном этапе производства осуществляется «персонализация» базового кристалла в требуемую матричную БИС. Пример слоев "персонализации" АЦ БМК представлен на рисунке 5.30.



Рисунок 5.30 – Слои для коммутации СБИС на основе АЦ БМК

При реализации аналого-цифровых устройств на основе разработанных микросхем АЦ БМК после этапа схемотехнического проектирования

электрической схемы осуществляется топологическая разработка устройства. Электрические соединения выполняются с помощью металлических шин.

Области цифровых элементов в БМК выполнены на основе бесканального принципа построения матричных структур («море вентилей») и позволяют реализовывать требуемое количество двухвходовых логических вентилей и триггерных ячеек.

Элементарной составляющей в области цифровых элементов является 2-х транзисторная базовая цифровая ячейка (БЦЯ). Цифровые ячейки объединены в строки (ряды), ограниченные с двух сторон шинами земли и питания. Группы строк собраны в виде матриц определенной размерности.

Каждая цифровая ячейка содержит 2 транзистора (1 р-МОП и 1 п-МОП) Каждая БЦЯ имеет 6 свободных контактов: затворы р-МОП и п-МОП транзисторов, сток р-МОП транзистора, исток р-МОП транзистора, сток п-МОП транзистора, исток п-МОП транзистора. Данное количество контактов позволяет реализовать на основе набора БЦЯ библиотечные цифровые элементы комбинационной и последовательностной логик. На рисунке 5.31 представлено структура базовой цифровой ячейки.



Х - узлы контактирования транзисторов

Рисунок 5.31 – Структурная схема базовой цифровой ячейки АЦ БМК

Аналоговая область БМК содержит в своем составе группы (ряды) ячеек транзисторов n-МОП и p-МОП типа (комплементарных пар).

Так же в состав АЦ БМК включены группы резисторов и конденсаторов.

Аналоговые ячейки (комплементарные транзисторы) объединены в группы, представляющие собой ряды транзисторов с одинаковыми геометрическими размерами (ячейки одного варианта), что позволяет обеспечить необходимое согласование при использовании структур с общим центром. Транзисторы распределены по группам симметрии в зависимости от своих геометрических размеров. Группы чередуются друг с другом и с наборами резисторов, поэтому возможно разрабатывать мультиплицируемые аналоговые блоки. Отдельные группы транзисторов можно использовать для построения согласованных аналоговых элементов (дифференциальных усилителей, токовых зеркал, сумматоров и др.). Аналогично расположены группы резисторов. Чередование этих групп позволяет создавать сложные аналоговые узлы.

АЦ БМК в своем составе имеет периферийную часть, представляющую собой набор контактных площадок типа вход/выход со встроенными элементами защиты от воздействия статического электричества.

Структурная схема на уровне модулей и элементов представлена на рисунке 5.32. Состав кристалла – в таблице 5.4.

Условное обозначение блока	Назначение		
КР	Реконфигурируемая контактная площадка		
mobile_digital_ 18V	Блок цифровых ячеек 1,8 В		
mobile_digital_ 5V	Блок цифровых ячеек 5 В		
4090bit MEM	Блок однократно программируемой памяти		
A5V	Блок резисторов и транзисторов 5 В		
A18V	Блок резисторов и транзисторов 1,8 В		
R	Блок резисторов номиналом 25 кОм		

Таблица 5.4 – Состав кристалла АЦ БМК

На рисунке 5.32 представлена общая структура БМК, состоящая из ядра микросхемы и периферийной части.



Рисунок 5.32 – Электрическая структурная схема базового матричного кристалла.

При построении конкретного библиотечного элемента и реализации схемы устройства в целом может быть задействовано произвольное число компонентов как отдельного модуля, так и компоненты из нескольких модулей одновременно. Электрические схемы аналоговых библиотечных элементов АЦ БМК приведены в Приложении 2.

Общий вид топологии кристалла приведен на рисунке 5.33.

l≽.					200 200 200 200 200 200 200 200 200 200
	H	H	+	#	

Рисунок 5.33 – Общий вид топологии АЦ БМК

Базовый кристалл АЦ БМК разработан для КМОП КНИ технологии. В таблице 5.5 представлен набор СФ блоков, одновременно реализуемых в базовом кристалле.

Функциональные блоки	Кол-во
Операционный усилитель общего применения (ОУ1), шт.	40
Прецизионный операционный усилитель (ОУ2), шт.	12
Быстродействующий операционный усилитель (ОУЗ), шт.	12
Компаратор общего применения (К1), шт.	30

Таблица 5.5 – Состав СФ блоков, реализуемых на АЦ БМК

Быстродействующий компаратор (К2), шт.	8
Аналого-цифровой преобразователь (АЦП), шт.	6
Цифро-аналоговый преобразователь (ЦАП), шт.	6
Блок памяти ОПЗУ, шт.	1
Мультиплексор, шт.	2
Источник опорного напряжения (ИОН), шт.	3
Встроенный линейный регулятор, шт.	6
Операционный усилитель общего применения с калибровкой смещения нуля (ОУ4), шт.	10
Прецизионный операционный усилитель с автоматической подстройкой смещения нуля, (ОУ5), шт.	6
Быстродействующий ОУ с автоматической подстройкой смещения нуля (ОУ6), шт.	6
Компаратор общего применения с настройкой смещения нуля (КЗ), шт.	10
Быстродействующий компаратор с автоматической калибровкой смещения нуля (К4), шт.	4
Прецизионный компаратор (К5), шт.	12
Драйвер внешнего силового ключа, шт.	2
Генератор тактовых сигналов, шт.	1

Ключевой особенностью топологии кристаллов микросхем АЦ БМК является существенное сокращение числа слоев «персонализации» при разработке матричных СБИС. Реализованная топологическая конструкция АЦ БМК позволяет при разработке произвольной «зашивки» ИС использовать 12 реконфигурируемых слоев (6 слоев металла, 5 слоев контактов и слой верхней обкладки конденсатора). Данное техническое решение существенно улучшает технико-экономические показатели АЦ БМК.

На рисунках 5.34 – 5.35 представлены топологические конструкции некоторых блоков.



Рисунок 5.34 – Топология полностью дифференциального ОУ



Рисунок 5.35 – Топология ИОН

Все библиотечные элементы реализованы на согласованной матричной структуре, состоящей из сгруппированных в аналоговые макроячейки активных и пассивных компонентов (КМОП транзисторов, резисторов, конденсаторов). На таких макроячейках однотипных элементов возможно проектировать основные аналоговые функциональные узлы большинства типов аналоговых и аналого-цифровых СБИС с сохранением высокой степени согласования и удовлетворения требований по точностным

характеристикам. В качестве примера на рисунках 5.36 – 5.37 представлены фрагменты топологии описанных матричных структур.



Рисунок 5.36 – Фрагмент топологии матричной структуры, содержащей согласованные транзисторы и резисторы для построения аналоговых блоков



Рисунок 5.37 – Фрагмент топологии матричной структуры, содержащей согласованные диоды

Разработанные микросхемы аналого-цифровых БМК обладают следующими особенностями:

- Широкий температурный диапазон: от минус 60 до плюс 125°С.
- Диапазон номинального питания от 3 до 5,25 В.
- Стойкость к воздействию специальных факторов.
- Возможность использования библиотечных элементов для реализации цифро-аналогового и аналого-цифрового преобразования.
- Возможность реализации широкого спектра ОУ и компараторов.
- Возможность реализации алгоритмов цифровой обработки сигналов.
- Возможность построения многоканальных систем обработки аналоговых сигналов.
- Возможность расширения диапазонов напряжения питания и входных/выходных сигналов.
- Возможность реализации схем управления питанием.
- Возможность изменения всех слоев персонализации, при незначительном увеличении стоимости.
- Конструктивные особенности:
- Большое число согласованных транзисторов и резисторов.
- Возможность произвольного расположения конденсаторов.
- Большой объем цифровых вентилей.
- Возможность трассировки 6-тью уровнями металлизации.
- Возможность реализации достаточно сложных цифровых блоков.
- Архитектурно-схемотехнические особенности:
- Наличие модуля высокосогласованных резисторов для построения АЦП и ЦАП на их основе.
- Наличие высокосогласованных наборов транзисторов n- и p- типа для реализации ОУ и компараторов с низким смещением нуля.
- Наличие модуля цифровых элементов для реализации систем автокалибровки смещения нуля.

- Наличие диодных матриц для реализации источников опорного напряжения
- Наличие однократно программируемого блока памяти.
- Специальная расширенная библиотека радиационно-стойких аналоговых и цифровых элементов.
- Возможность параметризации при разработке блоков ОУ по току потребления и скорости нарастания, и компараторов по задержке переключения и току потребления.

5.3. Выводы

Разработаны методы, позволяющие повысить функциональность и эффективность использования площади цифровых базовых матричных кристаллов.

Разработан метод проектирования аналоговых блоков гетерогенных конфигурируемых интегральных схем, в основе которого лежит использование предложенных в работе понятий аналоговых ядер и их онтологий.

Предложена конструкция конфигурируемых специализированных аналоговых и аналого-цифровых интегральных схем с оптимальной архитектурой для заданного класса задач.

Заключение

В диссертационной работе предложены теоретические обобщения и методы решения научных проблем разработки гетерогенных конфигурируемых интегральных схем, имеющих важное хозяйственное значение.

Основные результаты работы состоят в следующем:

1) Разработан метод комплектования специализированной аппаратуры проблемно-ориентированными и универсальными интегральными схемами, построенными на основе гетерогенных конфигурируемых схем.

2) Определены критерии эффективности использования площади базовых кристаллов конфигурируемых интегральных схем.

3) Разработана новая теоретико-множественная математическая модель проекта, объединяющая в себе различные стили проектирования (заказных, полузаказных - базовых матричных кристаллов и программируемых логических интегральных схем).

4) Предложена формализация соответствий между элементами базового кристалла гетерогенной конфигурируемой интегральной схемы и разрабатываемыми схемами заказчиков, функциональность которых требуется запрограммировать в базовом кристалле, что обеспечивает оперативную настройку конструкции и схемотехники базового кристалла на новые потребности от заказчиков и оперативную настройку САПР на соответствующие изменения в конструкции, схемотехнике и топологии базового кристалла.

5) Предложены методы предварительного анализа схем заказчиков с целью определения значений важнейших характеристик архитектуры разрабатываемых гетерогенных конфигурируемых схем с повышенной эффективностью использования трассировочных и иных ресурсов.

6) Предложен маршрут проектирования конфигурируемых интегральных схем, в рамках которого на основе разработанных методов предварительного анализа схем заказчиков и программного прототипирования создается архитектура и электрическая схема базового кристалла программируемой схемы.

7) Предложен новый метод, используемый в маршруте разработки гетерогенных конфигурируемых интегральных схем - программное прототипирование с оценкой эффективности реализаций проектируемых схем заказчиков до фактического изготовления базового кристалла за счет оперативной настройки САПР на изменения в конструкции, схемотехнике и топологии базового кристалла в соответствии с предложенной в работе стратегией.

8) Разработаны новые схемотехнические решения, позволяющие повысить функциональность и эффективность использования площади базовых матричных кристаллов.

9) Предложены новые методы повышения живучести, надежности и радиационной стойкости конфигурируемых интегральных схем на основе приемлемых и мягких решений, которые используя дополнительную информацию о вероятных отказах и сбоях функционирования внутренних узлов, позволяют обеспечить корректную работу конфигурируемых схем в условиях неблагоприятных внешних воздействий.

10) Разработан новый метод проектирования аналоговых блоков гетерогенных конфигурируемых интегральных схем, в основе которого лежит использование предложенных в работе понятий аналоговых ядер и их онтологий.

11)Предложенные методы, алгоритмы и решения внедрены в АО «НИИМЭ», АО «НПО Автоматики», АО «Микрон», АО «ДЦ «Союз», АО «НИИ «Субмикрон» при разработке большого числа конфигурируемых интегральных схем: ПЛИС, ПАЦИС, ПАИС, БМК, АЦ БМК.

Библиографический список

- [1] Красников Г.Я., Панасенко П.В., Волосов В.А., Щербаков Н.А. Тенденции развития технологии сложнофункциональной гетерогенной ЭКБ // Международный форум "Микроэлектроника-2018", 4-я Международная научная конференция "Электронная компонентная база и микро электронные модули". Сборник тезисов. 2018. С. 341-344.
- [2] Эннс В.И. Гибкие решения проблем импортозамещения ЭКБ для специальной техники. Электроника. Наука, технология, бизнес. №3 (174), 2018 г., с. 1 – 4
- [3] Эннс В. И. СнК, БМК или ПЛИС: выбор варианта исполнения цифровой интегральной схемы // Компоненты и технологии. 2018. № 4 (201). С. 100-102.
- [4] Красников Г.Я., Эннс В.И. Экспертиза применения электронной компонентной базы при создании аппаратуры космического И назначения специального важный _ шаг В решении задачи импортозамещения // 3: Электронная техника. Серия Микроэлектроника, 2021. Вып. 1 (181). С. 12-17.
- [5] Красников Г. Я. и др. Разработка и изготовление на отечественном предприятии по технологии с минимальными топологическими нормами не более 0, 18 мкм библиотеки аналоговых IP блоков для использования в составе сверхбольших интегральных схем "система на кристалле". – Минпромторг РФ, 2017. – №. 13411.1400099. 11.056.
- [6] Чочаев Р.Ж., Железников Д.А., Иванова Г.А., Гаврилов С.В., Эннс В.И. Модели и методы анализа структуры коммутационных ресурсов ПЛИС // Известия вузов. Электроника. 2020. Т. 25 №5. С. 410-422. DOI: 10.24151/1561-5405-2020-25-5-410-422.
- [7] Алексеев В.В., Телец В.А., Эннс В.И., Эннс В.В. Импортозамещение
 ЭКБ: базовые матричные кристаллы // ЭЛЕКТРОНИКА: НТБ. 2016.
 № 2.

- [8] Кобзев Ю., Эннс В.И., Эннс В.В., Иванов Д., Корепанов И. Состав и области применения программируемых схем серии 5400 // Компоненты и технологии, №12, 2017 г., с.12 – 14..
- [9] Li X., Yang H., Zhong H. Use of VPR in Design of FPGA Architecture // 2006 8th International Conference on Solid-State and Integrated Circuit Technology Proceedings. Shanghai, China: IEEE, 2006. P. 1880–1882.
- [10] Luu J. et al. VPR 5.0: FPGA CAD and Architecture Exploration Tools with Single-Driver Routing, Heterogeneity and Process Scaling. Monterey, California, USA: ACM, 2008. P. 133–142.
- [11] Parvez H. et al. A new coarse-grained FPGA architecture exploration environment // 2008 International Conference on Field-Programmable Technology. Taipei, Taiwan: IEEE, 2008. P. 285–288.
- [12] Kannan P., Balachandran S., Bhatia D. On metrics for comparing routability estimation methods for FPGAs // Proceedings 2002 Design Automation Conference (IEEE Cat. No.02CH37324). New Orleans, LA, USA: IEEE, 2002. P. 70–75.
- [13] Gao Hai-xia et al. A novel Monte-Carlo method for FPGA architecture research // Proceedings. 7th International Conference on Solid-State and Integrated Circuits Technology, 2004. Beijing, China: IEEE, 2004. Vol.3. P. 1944–1947.
- [14] Brown S.D., Rose J., Vranesic Z.G. A stochastic model to predict the routability of field-programmable gate arrays: 12 // IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 1993. Vol. 12, № 12. P. 1827–1838.
- [15] Chan P.K., Schlag M.D.F., Zien J.Y. On routability prediction for fieldprogrammable gate arrays // Proceedings of the 30th international on Design automation conference - DAC '93. Dallas, Texas, United States: ACM Press, 1993. P. 326–330.
- [16] Das J., Wilton S.J.E. Towards development of an analytical model relating FPGA architecture parameters to routability // ACM Trans. Reconfigurable Technol. Syst. 2013. Vol. 6, № 2. P. 1–24.

- [17] Das J. et al. An Analytical Model Relating FPGA Architecture to Logic Density and Depth // IEEE Trans. VLSI Syst. 2011. Vol. 19, № 12. P. 2229– 2242.
- [18] Smith A.M. et al. Concurrently optimizing FPGA architecture parameters and transistor sizing: Implications for FPGA design // 2009 International Conference on Field-Programmable Technology. Sydney, Australia: IEEE, 2009. P. 54–61.
- [19] Smith A.M., Constantinides G.A., Cheung P.Y.K. FPGA Architecture Optimization Using Geometric Programming // IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 2010. Vol. 29, № 8. P. 1163–1176..
- [20] Petelin O., Betz V. Wotan: Evaluating FPGA Architecture Routability without Benchmarks // ACM Trans. Reconfigurable Technol. Syst. 2018. Vol. 11, № 2. P. 1–23.
- [21] Lam A. et al. An analytical model describing the relationships between logic architecture and FPGA density // 2008 International Conference on Field Programmable Logic and Applications. Heidelberg, Germany: IEEE, 2008.
 P. 221–226.
- [22] Gavrilov S.V., Zheleznikov D.A., Zapletina M.A., Chochaev R.Z., Enns V.I. Layout Synthesis Design Flow for Special-Purpose Reconfigurable Systemson-a-Chip // Russian Microelectronics. 2019 48(3), P. 176-186. DOI:10.1134/s1063739719030053
- [23] XC4000E and XC4000X Series Field Programmable Gate Arrays / Xilinx.
 URL: https://www.xilinx.com/support/documentation/data_sheets/4000.pdf
 (дата обращения: 04.06.2020)
- [24] Stratix II Device Handbook, Volume 1. / Intel. URL: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literatur e/hb/stx2/stratix2_handbook.pdf (дата обращения: 10.06.2020)
- [25] Rose J., Brown S. Flexibility of interconnection structures for field-programmable gate arrays // IEEE J. Solid-State Circuits. 1991. Vol. 26, № 3. P. 277–282.

- [26] APEX 20K Programmable Logic Device Family Data Sheet / Intel. URL: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literatur e/ds/archives/apex.pdf (дата обращения: 26.05.2020)
- [27] FLEX 10K Embedded Programmable Logic Family Data Sheet / Intel. URL: https://edg.uchicago.edu/~bogdan/prefred/doc/parts/10k.pdf (дата обращения: 10.06.2020)
- [28] Lemieux G. et al. Directional and single-driver wires in FPGA interconnect // Proceedings. 2004 IEEE International Conference on Field- Programmable Technology (IEEE Cat. No.04EX921). Brisbane, NSW, Australia: IEEE, 2004. P. 41–48.
- [29] Frolova P.I., Chochaev R.Z., Ivanova G.A., Gavrilov S.V. Delay Matrix Based Timing-driven Placement for Reconfigurable Systems-On-Chip // 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), St. Petersburg and Moscow, Russia, 2020, P. 1799-1803.
- [30] Стемпковский А.Л., Гаврилов С.В., Каграманян Э.Р. Методы логиковременного анализа заказных блоков СБИС // Известия ВУЗов. Электроника. – 2008. - № 5. – С. 41- 50. (Stempkovskij A.L., Gavrilov S.V., Kagramanjan E.R. Metody logiko-vremennogo analiza zakaznyh blokov SBIS // Izvestija VUZov. Elektronika. – 2008. - № 5. – S. 41- 50.)
- [31] Murray K.E. et al. Titan: Enabling large and complex benchmarks in academic CAD // 2013 23rd International Conference on Field programmable Logic and Applications. Porto, Portugal: IEEE, 2013. P. 1–8.
- [32] Fang W.M., Rose J. Modeling routing demand for early-stage FPGA architecture development // Proceedings of the 16th international ACM/SIGDA symposium on Field programmable gate arrays - FPGA '08. Monterey, California, USA: ACM Press, 2008. P. 139.
- [33] Gavrilov S.V., Zheleznikov D.A., Khvatov V.M., Chochaev R. Z. Clustering optimization based on simulated annealing algorithm for Reconfigurable

Systems-on-Chip // 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering. 2018. P. 1492-1495.

- [34] Gavrilov S.V., Zheleznikov D.A., Chochaev R.Z. Simulated Annealing Based Placement Optimization for Reconfigurable Systems-on-Chip // 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Moscow, 2019. P. 1597-1600.
- [35] Vasilyev N., Ryzhova D., Tiunov I. Resynthesis for FPGA during technology mapping stage // 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Moscow, 2019. P. 1644-1647.
- [36] Gamal A.E. Two-dimensional stochastic model for interconnections in master slice integrated circuits // IEEE Trans. Circuits Syst. 1981. Vol. 28, № 2. P. 127–138.
- [37] Feuer. Connectivity of Random Logic: 1 // IEEE Trans. Comput. 1982. Vol.
 C-31, № 1. P. 29–33.
- [38] Landman B.S., Russo R.L. On a Pin Versus Block Relationship For Partitions of Logic Graphs: 12 // IEEE Trans. Comput. 1971. Vol. C–20, № 12. P. 1469–1479.
- [39] Suehiro Y., Miura D., Naioh M., Tsutsumi S., Shirato T. A 120K Gate usable CMOS sea of gates packing 1.3M transistors // IEEE 1988 Custom Integrated circuits conference. - P. 20.5.1 – 20.5.4.
- [40] Sano T., Matsukuma S., Hashimoto K., Ohuchi Y., Kudo O., Yamamoto H. A 20 ns CMOS Functional Gate Array with a Configurable Memory // IEEE International Solid –State Circuits Conferens.-1983. – P. 146 – 148.
- [41] Ko U., Schenck S., Svejda F., La S. 0,65 mkm, low voltage low-power optimized CMOS gate array // Texas Instruments Incorporated, 8390 LBJ Freeway.- M/S 3671.- Dallas, Texas 75243.- Phone: (214) 997-2968, Fax: (214) 997-2090.

- [42] Okabe M., Okuno Y., Arakawa et al. A 400K transistor CMOS sea-ofgates array with continuous track allocation // IEEE Journal of Solid –State Circuits. -October 1989. No 5- Vol. 24.- P. 1280 - 1286.
- [43] Green J- M., Klar H. A CMOS gate arrays architecture for digital signal processing applications // IEEE Journal of Solid – State Circuits. -March 1996. - No. 3.-Vol. 31.- P. 410 – 418.
- [44] Yung Y.S., Robles R. A complete 0.5 μ high performance array family // IEEE.- 1995.- P. 53 – 56. (Internet)
- [45] Van Noije W. A. M., Declerk G. J. Advanced CMOS Gate Array Architecture Combining "Gate Isolation" and Programmable Routing Channels // IEEE Journal of solid – state circuits.- April 1985.- No.2.-Vol. SC-20.-P. 469 – 480.
- [46] Sakashita K., Ueda M., Arakava T., Asai S., Fujimura T., Ohkura I. A 10K-Gate CMOS Gate Array Based on a Gate Isolation Structure // IEEE Transactions on Electron Devices.- February 1985.- No2.- Vol. ED 32.- P. 493 442.
- [47] Красников Г.Я., Панасенко П.В., Волосов В.А., Щербаков Н.А. Тенденции развития технологии сложнофункциональной гетерогенной ЭКБ // Международный форум «Микроэлектроника-2018», 4-я Международная научная конференция «Электронная компонентная база и микроэлектронные модули». Сборник тезисов. 2018. С. 341-344.
- [48] Красников Г.Я. Возможности микроэлектронных технологий с топологическими размерами менее 5 нм // Наноиндустрия. 2020. Т. 13. № S5-1 (102). С. 13-19.
- [49] Красников Г. Я. и др. Разработка и изготовление на отечественном предприятии по технологии с минимальными топологическими нормами не более 0, 18 мкм библиотеки аналоговых IP блоков для использования в составе сверхбольших интегральных схем «система на кристалле». – Минпромторг РФ, 2017. – №. 13411.1400099. 11.056.

- [50] Эннс В. И. СнК, БМК или ПЛИС: выбор варианта исполнения цифровой интегральной схемы // Компоненты и технологии. 2018. № 4 (201). С. 100-102.
- [51] Hammerquist M., Lysecky R. Design space exploration for application specific FPGAs in system-on-a-chip designs // 2008 IEEE International SOC Conference, Newport Beach, CA, USA, 2008. – PP. 279-282.
- [52] Parvez H., Marrakchi Z., Kilic A., Mehrez H. Application-Specific FPGA using heterogeneous logic blocks // ACM Trans. Reconfigurable Technol. Syst., 2011. – Vol. 4, No.3, Article 24. – 14 p.
- [53] Luu J, et. al. VTR 7.0: Next Generation Architecture and CAD System for FPGAs // ACM Trans. Reconfigurable Technol. Syst., 2014. – Vol. 7, No. 2, Article 6. – 30 pages.
- [54] Nasartschuk K., Herpers R., Kent K.B. Visualization support for FPGA architecture exploration // 2012 23rd IEEE International Symposium on Rapid System Prototyping (RSP), Tampere, Finland, 2012. – PP. 128-134.
- [55] Гаврилов С.В., Железников Д.А., Заплетина М.А., Хватов В.М., Чочаев Р.Ж., Эннс В.И. Маршрут топологического синтеза для реконфигурируемых систем на кристалле специального назначения / С. В. Гаврилов, Д. А. Железников, М. А. Заплетина [и др.] // Микроэлектроника. 2019. Т. 48. № 3. С. 211-223.
- [56] Kannan P., Balachandran S., Bhatia D. On metrics for comparing routability estimation methods for FPGAs // Proceedings 2002 Design Automation Conference (IEEE Cat. No.02CH37324), New Orleans, LA, USA, 2002. – PP. 70-75.
- [57] Железников Д. А., Заплетина М. А., Хватов В. М. Решение задачи трассировки межсоединений для реконфигурируемых систем на кристалле с различными типами коммутационных элементов // Электронная техника. Серия 3: Микроэлектроника, 2018. – №. 4. – С. 31-36.

- [58] McMurchie L., Ebeling C. PathFinder: A Negotiation-Based Performance-Driven Router for FPGAs // Third International ACM Symposium on Field-Programmable Gate Arrays, Napa Valley, CA, USA, 1995. – PP. 111-117.
- [59] Заплетина М.А., Железников Д.А., Гаврилов С.В. Иерархический подход к трассировке реконфигурируемой системы на кристалле островного типа // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС), 2020. – №3. – С. 16-21.
- [60] Жуков Д.В., Железников Д.А., Заплетина М.А. Применение SATподхода к трассировке блоков коммутации для реконфигурируемых систем на кристалле // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС), 2020. – № 1. – С. 26-32.
- [61] Doman D. Engineering the CMOS Library: Enhancing Digital Design Kits for Competitive Silicon // John Wiley & Sons Ltd., 2012, 256 p.
- [62] Amos D., Lesea A., Richter R. FPGA-based Prototyping Methodology Manual: Best Practices in Design-for-prototyping // Synopsys Press, 2011., 470 p
- [63] Ohba N., Takano K. An SoC design methodology using FPGAs and embedded microprocessors // In Proceedings of the 41st annual Design Automation Conference (DAC '04). Association for Computing Machinery, New York, NY, USA, pp. 747–752.
- [64] Чочаев Р.Ж., Железников Д.А., Иванова Г.А., Гаврилов С.В., Эннс В.И. Модели и методы анализа структуры коммутационных ресурсов ПЛИС // Известия высших учебных заведений. Электроника. 2020. Т. 25. № 5. С. 410-422.
- [65] Li X., Yang H., Zhong H. Use of VPR in Design of FPGA Architecture // 2006 8th International Conference on Solid-State and Integrated Circuit Technology Proceedings. Shanghai, China: IEEE, 2006. P. 1880–1882.
- [66] S. Gandhare and B. Karthikeyan, Survey on FPGA Architecture and Recent Applications // 2019 International Conference on Vision Towards Emerging

Trends in Communication and Networking (ViTECoN), Vellore, India, 2019, pp. 1-4.

- [67] MAX II Device Handbook. // Altera Corp. [Электронный ресурс]. Системные требования: AdobeAcrobat Reader. Режим доступа: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literatur e/hb/max2/max2 mii5v1.pdf (дата обращения: 01.04.2021)
- [68] UltraScale Architecture Configurable Logic Block User Guide // Xilinx, Inc.
 [Электронный ресурс]. Системные требования: Adobe Acrobat Reader.
 Режим доступа: https://www.xilinx.com/support/documentation/user_guides/ug574ultrascale-clb.pdf (дата обращения: 01.04.2021)
- [69] ProASIC3 nano FPGA Fabric User's Guide // Microsemi Corp.
 [Электронный ресурс]. Системные требования: Adobe Acrobat Reader.
 Режим доступа: http://www.ibselectronics.com/ibsstore/datasheet/Microsemi/PA3_nano_UG
 .pdf. (дата обращения: 01.04.2021)
- [70] GDSII™ Stream Format Manual, Release 6.0 // Calma Company.
 [Электронный ресурс]. Системные требования: Adobe Acrobat Reader.
 Режим доступа: http://bitsavers.informatik.uni-stuttgart.de/pdf/calma/GDS_II_Stream_Format_Manual_6.0_Feb87.pdf.
 (дата обращения: 01.04.2021)
- [71] Гаврилов С.В., Железников Д.А., Заплетина М.А., Хватов В.М., Чочаев Р.Ж., Эннс В.И. Маршрут топологического синтеза для реконфигурируемых систем на кристалле специального назначения // Микроэлектроника. 2019. Т. 48. № 3. С. 211-223.
- [72] Васильев Н.О., Тиунов И.В., Рыжова Д.И. Метод логического ресинтеза схем в маршруте проектирования на ПЛИС // МЭС 2020 Проблемы разработки перспективных микро- и наноэлектронных систем. 2020 (МЭС-2020). Выпуск 4. С. 39-44.

- [73] Иванова Г. А., Рыжова Д. И., Гаврилов С. В., Васильев Н.О., Стемпковский А.Л. Методы и алгоритмы для логико-топологического проектирования микроэлектронных схем на вентильном и межвентильном уровне для перспективных технологий с вертикальным затвором транзистора // Микроэлектроника. 2019. Т. 48. № 3. С. 201-210
- [74] Гаврилов С.В., Железников Д.А., Чочаев Р.Ж., Хватов В.М. Алгоритм декомпозиции на основе метода имитации отжига для реконфигурируемых систем на кристалле // Проблемы разработки перспективных микро- и наноэлектронных систем. 2018. Выпуск І. С. 199-204.
- [75] Гаврилов С. В., Железников Д. А., Чочаев Р. Ж., Эннс В. И. Адаптация метода моделирования отжига для размещения элементов в базисе реконфигурируемых систем на кристалле // Электронная техника. Серия 3. Микроэлектроника. 2018. Вып. 4 (172). С. 55-61.
- [76] Заплетина М.А., Железников Д.А., Гаврилов С.В. Иерархический подход к трассировке реконфигурируемой системы на кристалле островного типа // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. № 3. С. 16-21.
- [77] Brglez F., Bryan D., Kozminski K. Combinational profiles of sequential benchmark circuits // IEEE International Symposium on Circuits and Systems, vol.3., Portland, OR, USA, 1989, pp. 1929-1934.
- [78] Usselmann R. AC 97 Controller IP Core // Rudolf Usselmann.[Электронный ресурс].Режим доступа:https://opencores.org/projects/ac97 (дата обращения: 03.04.2021).
- [79] Эннс В.И., Гаврилов С.В., Хватов В.М., Курбатов В.Г. Проектирование ПЛИС и реконфигурируемых СнК с использованием методов программного анализа и прототипирования // Микроэлектроника. 2021. Т. 50 №6, с. 467-480.

- [80] Чочаев Р.Ж., Железников Д.А., Иванова Г.А., Гаврилов С.В., Эннс В.И.
 Модели и методы анализа структуры коммутационных ресурсов ПЛИС
 // Известия вузов. Электроника. 2020. Т. 25 №5, с. 410-422.
- [81] UltraScale Architecture Configurable Logic Block User Guide // Xilinx, 2017.URL:https://www.xilinx.com/support/documentation/user_guides/ug57 4-ultrascale-clb.pdf
- [82] Intel Stratix 10 Logic Array Blocks and Adaptive Logic Modules UserGuide //Intel,2020.URL:https://www.intel.com/content/dam/www/programmable/u s/en/pdfs/literature/hb/stratix-10/ug-s10-lab.pdf
- [83] Харари Ф. Теория графов / Пер. с англ. Изд. 5, доп. // М.: Ленанд, 2018.
 304 с.
- [84] Dijkstra E. W. A note on two problems in connexion with graphs // Numerische Mathematik, 1959. Vol.1, Is.1. PP. 269–271.
- [85] Dinitz Y. Dinitz' Algorithm: The Original Version and Even's Version // Theoretical Computer Science: Essays in Memory of Shimon Even / Oded Goldreich, Arnold L. Rosenberg, and Alan L. Selman. Springer, 2006. P. 218–240.
- [86] Гузаиров М.Б., Гвоздев В.Е., Давлиева А.С., Тесленко В.В. Оценка живучести аппаратно-программных комплексов по среднему значению показателя целевой эффективности.// Информационные и математические технологии в науке и управлении. 2018г. №1 (9), с. 106-112.
- [87] Дедков В.К., Водолазский В.И., Мухин А.В., Фесечко А.И. Живучесть и безопасность сложных технических систем. // Вопросы теории безопасности и устойчивости систем. 2002. №4, с. 63-69.
- [88] Додонов А.Г., Ландэ Д.В. Живучесть информационных систем. К.: Наук. Думка. 2011. 256с.
- [89] Махутов Н.А., Резников Д.О., Петров В.П., Куксова В.И. оценка живучести сложных технических систем // Проблемы безопасности и чрезвычайных ситуаций. 2009. №3. С. 47-66.

- [90] Эннс В.И. Цифровые решения современных отказоустойчивых аналоговых интегральных схем. Сборник трудов XIII научно-практического семинара «Проблемы создания специализированных радиационно-стойких СБИС на основе гетероструктур», г. Нижний Новгород, НИИИС, 2013 г., с. 5 7.
- [91] Форни Д. Каскадные коды. М., «Мир», 1970, 207 с.
- [92] Красников Г.Я., Зайцев Н.А., Красников А.Г. Современное состояние разработок в области энергонезависимой памяти // Нано- и микросистемная техника. 2015, № 4 (177), с. 60-64.
- [93] Самофалов К.Г., Корнейчук В.И., Городний А.В., Структурнологические методы повышения надежности запоминающих устройств. М., «Машиностроение», 1976.
- [94] Эннс В.И. Избыточные интегральные схемы постоянной памяти. -Электронная техника, 1984, сер. 10, вып. 3, с. 30-33.
- [95] Фастов С.А., Эннс В.И. Методы построения быстродействующих ЭСЛ ОЗУ со встроенными схемами исправления ошибок. - Вопросы кибернетики/АН СССР. Научный совет по комплексной проблеме «Кибернетика». - М., 1988, с. 113-129.
- [96] Osman F.I. Error-Correcting Technique for Random-Access Memories. IEEE Journal of Solid-State Circuits, vol. Sk-17, №5, Octjber 1982, pp.877-881.
- [97] Chen C.L. Hsiao M.Y. Error- Correcting Cjdes for Semiconductor MemoryApplications: A State-of-the-Art Review. IBM Journal of Research and Development, 1984, March, vol. 28, №2, pp. 124-134.
- [98] Питерсон У., Уэлдон Э. Коды, исправляющие ошибки. М., «Мир», 1976.
- [100] Эннс В.И. Новые границы областей декодирования некоторых методов исправления ошибок с использованием мягкого решения. III международный семинар по теории информации «Сверточные коды;

связь с многими пользователями». Тезисы докладов. Сочи, 1987, с. 172-174.

- [101] Думер И.И. Выбор метода каскадного декодирования в зависимости от уровня шума в канале. УШ Всесоюзная конференция по теории кодирования и передачи информации. Часть II, Куйбышев, 1981, с. 61-65.
- [102] Эннс В.И. Интегральные микросхемы запоминающих устройств с самоконтролем для оперативной и внешней памяти ЭВМ. Вопросы кибернетики/АН СССР. Научный совет по комплексной проблеме «Кибернетика». М., 1984, с. 93-111.
- [103] Эннс В.И. БИС ЗУ с самоконтролем. Вопросы кибернетики/АН СССР. Научный совет по комплексной проблеме «Кибернетика». М., 1985, с. 174-182.
- [104] Богатырев Ю.В., Ластовский С.Б., Огородников Д.А., Чеховский В.А., Темирбулатов М.С., Эннс В.И., Шмигельский В.Н. Воздействие гаммаизлучения на параметры счетного триггера. Сб. науч. тр. VII Междунар. науч. конф., Минск, 2016 г., изд. центр БГУ, с. 150–153.
- [105] Чумаков, А. И. и др. Радиационная стойкость изделий ЭКБ: Научное издание // М.: НИЯУ МИФИ. 2015.
- [106] Телец, В. и др. ПЛИС для космических применений. Архитектурные и схемотехнические особенности // Электроника: наука, технология, бизнес. – 2005. – №. 6. с. 44-49.
- [107] Petrović, V., Schoof, G., Stamenković, Z. Fault-tolerant TMR and DMR circuits with latchup protection switches // Microelectronics Reliability. – 2014. – T. 54. – No. 8., p.1613-1626.
- [108] Тюрин, С. Ф. Радиационно-устойчивая ячейка SRAM //Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2014. – №. 12. – С.14-30.

- [109] Maru, A. et al. DICE-based flip-flop with SET pulse discriminator on a 90 nm bulk CMOS process // IEEE Transactions on Nuclear Science. – 2010. – T. 57. – №. 6. – P.3602-3608.
- [110] Темирбулатов, М. С., Эннс, В. И., Бобровский, Д. В. Методы повышения стойкости к сбоям счетного триггера // Электронная техника. Серия 3: Микроэлектроника. – 2016. – №. 2. – С.77-81.
- [111] Красников, Г. Я. и др. Исследование сбоеустойчивости СОЗУ с функцией исправления одиночных сбоев при воздействии ТЗЧ // Электронная техника. Серия 3: Микроэлектроника. – 2018. – № 1(169). – С.68-76.
- [112] Нидеккер Л.Г., Эннс В.И. Маршрут разработки цифровой СБИС на основе БМК // научное сообщество студентов XXI столетия. Технические науки: Электрон сб. ст. по матер. XXVIII студ. Междунар. научн.-практ. конф. – Новосибирск : Изд.: «СибАК», 2015. Н. 1 (27).
- [113] Нидеккер Л.Г., Шмигельский В.Н., Эннс В.И. Проектирование цифровых схем на основе БМК с использованием стандартных элементов. Электронная техника. Серия 3 «Микроэлектроника». Выпуск 1 (161).Москва, 2016 г., с. 23 – 25.
- [114] Green J- M., Klar H. A CMOS gate arrays architecture for digital signal processing applications // IEEE Journal of Solid – State Circuits. -March 1996. - No. 3.-Vol. 31.- P. 410 – 418.
- [115] Денисов А.Н., Фомин Ю.П., Коняхин В.В., Федоров Р.А., под редакцией Саурова А.Н. Библиотека функциональных ячеек для проектирования полузаказных микросхем серий 5503 и 5507. М., «АйБиПринт», 2012 г., 300 с.
- [116] Эннс В.В., Эннс В.И., Кобзев Ю.М. «Аналоговые ядра как основа аналоговых интегральных схем», Материалы Международной научнотехнической конференции «Фундаментальные проблемы радиоэлектронного приборостроения», 24-28 октября 2006г., М., Часть 3, с. 193-195.

- [117] Эннс В.В., Эннс В.И. Принципы построения низковольтовых аналоговых ядер// INTERMATIC — 2007, Материалы V Международной научно-технической конференции. 23-27 октября 2007 г. Часть 1. - Москва: МИРЭА. - С.243-245.
- [118] Кобзев Ю.М., Эннс В.И. Аналоговые ядра основа как библиотек высокоуровневого описания блоков. аналоговых Международная научно-техническая конференция с элементами научной школы для молодежи «Проектирование систем на кристалле: развития и проблемы». Тезисы тенденции докладов. Москва, Зеленоград, 2010, с. 16.
- [119] Стюарт Рассел, Питер Норвиг. Искусственный интеллект. Современный подход. Второе издание. Издательский дом «Вильямс», Москва, Санкт-Петербург, Киев 2006г., 1408 с.
- [120] Казённов Г.Г. Основы проектирования интегральных схем и систем. БИНОМ, Лаборатория знаний, 2005г., 295 с.
- [121] Джон ван Гиг. Прикладная общая теория систем. М., Мир, 1981, 336 с.
- [122] Michael K. Smith, Chris Welty, Deborah L. McGuinness, "OWL Web Ontology Language Guide", W3C Recommendation, 10February 2004.
- [123] Franz Baader, Deborah L. McGuinness, Daniele Nardi, Peter F. Patel-Schneider. "The Description Logic Yandbook: Theory, implementation, and applications", Cambridge University Press, 2003, 574 pages.
- [124] Szajda K.S., Sodini C.G., Bowman H.F. A low noise, high resolution silicon temperature sensor // IEEE Journal of Solid-State Circuits. – September, 1996. – Vol. 31, p. 1308-1313.
- [125] Эннс А.В., Эннс В.И. Реализация метода кусочной аппроксимации источника опорного напряжения. Международная научно-техническая конференция c элементами научной школы для молодежи «Проектирование кристалле: тенденции систем на развития И проблемы». Тезисы докладов. Москва, Зеленоград, 2010, с. 68.

- [126] Алексеев В.В., Телец В.А., Эннс В.И., Эннс В.В. Импортозамещение ЭКБ: базовые матричные кристаллы // ЭЛЕКТРОНИКА: НТБ. 2016. №2.
- [127] Кобзев Ю., Эннс В.И., Эннс В.В., Иванов Д., Корепанов И. Состав и области применения программируемых схем серии 5400 // Компоненты и технологии, №12, 2017 г., с.12 – 14.

Приложения

Приложение 1. Описание цифровых блоков БМК

Схемы библиотечных элементов ячеек ввода/вывода

На основе ячеек ввода/вывода БМК возможны конфигурации различных схем входных/выходных буферов цифровых и аналоговых сигналов с реализацией схем защиты от воздействия электростатического разряда. На рисунках ниже представлены примеры реализуемых элементов.

На рисунке П.1 представлен библиотечный элемента входного буфера цифрового сигнала с размахом входного и выходного напряжений до +3,3В. На рисунке П.2 представлен библиотечный элемента для подключения положительного или отрицательного питания с элементом электростатической защиты типа «земля-питание» и рабочим напряжением до +3,3В.



Рисунок П.1 – Структурная схема входного буфера цифрового сигнала с размахом входного и выходного напряжений до +3,3В



Рисунок П.2 - Схема электрическая элемента электростатической защиты типа «земля-питание»

СФ блоки ЗУ

Для БМК были разработанны IP ядра реконфигурируемого ЗУ с организацией 256 х 9, 8 разрядов информационных и один контроль на четность.

Каждый блок памяти может быть использован как виде двухпортовой памяти (один порт для чтения, другой для записи) так и в качестве FIFO. Всего в устройстве имеется 24 кБайт блоков такой памяти. Блоки конструктивно расположены по углам кристалла по 24 блока на каждом углу. Всего 96 блоков.

Каждый блок памяти может быть сконфигурирован в качестве FIFO или ОЗУ блока с независимым выбором синхронного или асинхронного режима чтения и записи. Кроме того, имеются программируемые флаги, которые, например, позволяют установить проверку или генерацию четности. В таблицах П.1 и П.2 описаны интерфейсные сигналы блоков ОЗУ и FIFO. Один блок памяти предназначен для работы на частоте вплоть до 150МГц (максимальная скорость при нормальных условиях). Блоки памяти можно группировать, увеличивая разрядность и/или адресность памяти, для создания желаемой организации памяти (смотри рисунок П.2). Таким

образом, можно обеспечить ширину в 9 (один блок), 18 и 36 бит, а так же глубину в 256, 512, 768 и 1024 слова.

На рисунке П.4 показан пример оптимального использования памяти. Для генерации трех групп блоков с различными разрядностью и адресацией на рисунке П.5 показан пример, как можно параллельно соединить ОЗУ блоки для создания дополнительных портов чтения.

Сигнал ЗУ	Биты	Вход/выход	Описание
WCLKS	1	Вход	Клок записи используется для синхронизации во время
			записи
RCLKS	1	Вход	Клок чтения используется для синхронизации во время
			чтения
RADDR<0:7>	8	Вход	Адрес чтения
RBLKB	1	Вход	Выбор блока для чтения (активный 0)
RDB	1	Вход	Импульс чтения (активный 0)
WADDR<0:7>	8	Вход	Адрес записи
WBLKB	1	Вход	Выбор блока для записи (активный 0)
DI<0:8>	9	Вход	Биты входных данных <0:8>, <8> может использоваться в
			качестве входного бита четности
WRB	1	Вход	Импульс записи (активный 0)
DO<0:8>	9	Выход	Биты выходных данных <0:8>, <8> может использоваться в
			качестве выходного бита четности
RPE	1	Выход	Ошибка четности при чтении (активная 1)
WPE	1	Выход	Ошибка четности при записи (активная 1)
PARODD	1	Вход	Выбирает генерацию/проверку четности. 1 – нечетная, 0 –
			четная.

Таблица П.1 - Интерфейсные сигналы блока памяти ОЗУ

Таблица П.2 - Интерфейсные сигналы FIFO блока

FIFO сигнал	Биты	Вход/Выход	Описание
WCLKS	1	Вход	Клок записи используется для синхронизации во время
			записи
RCLKS	1	Вход	Клок чтения используется для синхронизации во время
			чтения
LEVEL<0:7>	8	Вход	Устанавливает уровень для переключения флагов
			заполненности
RBLKB	1	Вход	Выбор блока для чтения (активный 0)
RDB	1	Вход	Импульс чтения (активный 0)
RESET	1	Вход	Сбрасывает FIFO указатели (активный 0)
WBLKB	1	Вход	Выбор блока для записи (активный 0)
DI<0:8>	9	Вход	Биты входных данных <0:8>, <8> может
			использоваться в качестве входного бита четности
WRB	1	Вход	Импульс записи (активный 0)
FULL, EMPTY	2	Выход	FIFO флаги. FULL предотвращает запись и EMPTY
			предотвращает чтение
EQTH,	2	Выход	EQTH находится в 1, когда в FIFO хранится количество
GEQTH			слов, указанное в сигнале LEVEL. GEQTH находится в
			1, когда в FIFO хранится количество слов равное или
			больше LEVEL.
DO<0:8>	9	Выход	Биты выходных данных <0:8>, <8> может
			использоваться в качестве выходного бита четности
RPE	1	Выход	Ошибка четности при чтении (активная 1)
WPE	1	Выход	Ошибка четности при записи (активная 1)
LGDEP<0:2>	3	Вход	Конфигурирует глубину FIFO по формуле 2 ^{LGDEP+1}
PARODD	1	Вход	Выбирает генерацию/проверку четности. 1 – нечетная,
			0-четная.







Рисунок П.5 - Параллельное соединение блоков памяти

Ниже приведены результаты моделирования (рисунки П.6 и П.7) разработанного IP ядра реконфигурируемого ЗУ с организацией 256х9, 8 разрядов информационных и один контроль на четность.
Режим FIFO. Синхронная запись и чтение. Генерация контрольного бита четности (контрольный бит – нечетный).



Рисунок П.6 - Схема моделирования ЗУ



Рисунок П.7 - Результаты моделирования

Синхронное чтение. Временная диаграмма приведена на рисунке П.8,

ее параметры – в таблице П.3.



Рисунок П.8 - Временная диаграмма синхронного чтения

Символ	Описание	Макс	Мин	Тип.	Едини- цы	Приме- чание
CCYC	Время цикла	8.8		7.5	нс	
СМН	Время нахождения синхросигнала в 1	4.4		3.75	нс	
CML	Время нахождения синхросигнала в 0	4.4		3.75	нс	
ECBA	Сигнал EMPTY после фронта RCLKS	7.5		2.9	нс	
FCBA	Среза сигнала FULL после фронта RCLKS	5		1.93	нс	*
OCA	Появление данных DO после фронта RCLKS	8.7		3	нс	
RDCS	Установка RDB перед фронтом RCLKS		1	1	нс	
RDCH	Удержание RDB после фронта RCLKS		0.1	0.5	нс	
RPCA	Появление сигнала RPE после фронтаRCLKS	10.1		3.9	нс	
HCBA	Появление EQTH или GEQTH	8		2.75	нс	
Idyn	Ток потребления	7.08	1.7	1.98	мА	

Таблица П.3 - Параметры временных диаграмм синхронного ч	тения
--	-------

Примечание: * К срезу RCLKS сигналы FULL и EMPTY не привязаны

Синхронная запись. Результаты моделирования приведены на рисунке П.9, параметры временных диаграмм – на рисунке П.10 и в таблице П.4.



Рисунок П.9 - Результаты моделирования синхронной записи



Рисунок П.10 - Параметры временных диаграмм синхронной записи

Таблица П.4 - Параметры временных диаграмм синхронной записи

Символ	Описание	Макс	Мин	Тип.	Единицы
CCYC	Время цикла	8,2		7,5	нс
СМН	Время нахождения синхросигнала в 1	4,1		3,75	нс
CML	Время нахождения синхросигнала в 0	4,1		3,75	нс
DCS	Установка данных перед фронтом WCLKS		0	1	нс
DCH	Удержание данных после фронта WCLKS	1,8		0,7	нс
FCBA	Появление сигнала FULL после переднего фронта WCLKS	7,6		2,8	нс
ECBA	Срез сигнала EMPTY после переднего фронта WCLKS	7		2,7	нс
HCBA	Появление EQTH или GEQTH после переднего фронта WCLKS	7,9		2,5	нс
WPCA	Появление WPE после фронта WCLKS	4,6		2	нс
WPCH	Старый сигнал WPE после фр, WCLKS	4		1,5	нс
WRCS, WBCS	Установка WRB&WBLKB перед фронтом WCLKS		1	1	нс
WRCH	Удержание WRB&WBLKB после фронта WCLKS		0,1	0,5	нс
Idyn	Ток потребления	7,34	1,87	1,36	мА

Режим FIFO. Асинхронная запись и чтение. Режим сброса.

Генерация контрольного бита четности (контрольный бит – четный).

Схема моделирования - на рисунке П.11, результаты моделирования – на рисунке П.12, временная диаграмма – на рисунке П.13, ее параметры в таблице П.5.



Рисунок П.11 – Схема моделирования



Рисунок П.12 - Результаты моделирования



Рисунок П.13 - Временная диаграмма асинхронного чтения

Символ	Описание	Макс	Мин	Тип.	Единицы	Примечания
ERDA	Появление EMPTY после среза RB	7.6		2.9	нс	Не фронт RB
FRDA	Появление среза FULL после среза RB	4.6		1.7	нс	Не фронт RB
ORDA	Появление DO после среза RB	8.3		2.8	нс	
ORDH	Старые данные DO после среза RB	5.2		2	нс	
RDCYC	Время цикла	8.5		7.5	нс	
RDH	Время нахождения WB в "1"	3		3.75	нс	
RDL	Время нахождения WB в "0"	5.5		3.75	нс	
RPRDA	Появление нового RPE после среза RB	10.5		4.1	нс	
RPRDH	Старый RPE после среза RB	8		1.5	нс	
THRDA	Появление EQTH или GETH после среза RB	8		2.4	нс	
Idyn	Ток	6.6	2	2.91	мА	

Таблица П.5 - Параметры временных диаграмм асинхронного чтения

Асинхронная запись. Результаты моделирования приведены на рисунке П.14, временная диаграмма – на рисунке П.15, ее параметры в таблице П.6.



Рисунок П.14 - Результаты моделирования асинхронной записи



Рисунок П.15 - Временные диаграммы асинхронного записи

Символ	Описание	Макс	Мин	Тип.	Единицы	Примечания
DWRH	Удержание DI после фронта WB	1.3		1,5	нс	PARGEN не активен
DWRH	Удержание DI после фронта WB		0,1		нс	PARGEN активен
DWRS	Установка DI перед фронтом WB		0,4	0,5	нс	PARGEN не активен
DWRS	Установка DI перед фронтом WB		2,5	3,1	нс	PARGEN активен
EWRA	Появление среза ЕМРТУ после среза WB	6,7		2,6	нс	Не фронта WB
FWRA	Появление сигнала FULL после среза WB	7,2		2,7	нс	Не фронта WB
THWRA	Появление EQTH или GETH после среза WB	5,7		2,2	нс	Не фронта WB
WPDA	Появление нового WPE после DI	2,7		1,7	нс	
WRCYC	Время цикла	8		7,5	нс	
WRH	Время нахождения WB в "1"	4		3,75	нс	
WRL	Время нахождения WB в "0"	4		3,75	нс	
Idyn	Ток	6,6	0,05	1,5	мА	

Таблица П.6 - Параметры временных диаграмм асинхронной записи

Режим SO3У. Синхронная запись и чтение. Проверка контрольного бита четности (контрольный бит – четный). Схема моделирования приведена на рисунке П.16, результаты моделирования – на рисунке П.17, временная диаграмма – на рисунке П.18, ее параметры в таблице П.7.



Рисунок П.16 – Схема моделирования



Рисунок П.17 - Результаты моделирования



Рисунок П.18 - Временная диаграмма синхронного чтения

Символ	Описание	Макс	Мин.	Тип.	Единицы	Примечания
CCYC	Время цикла	8.5		7.5	нс	
СМН	Время нахождения синхросигнала в 1	4.25		3.75	нс	
CML	Время нахождения синхросигнала в 0	4.25		3.75	нс	
RACS	Установка адреса перед фронтом RCLKS		0	1	нс	
RACH	Удержание адреса после фронта RCLKS	1.3		0.5	нс	
RDCS	Установка RDB перед фронтом RCLKS		1	1	нс	
RDCH	Удержание RDB после фронта RCLKS		0.1	0.5	нс	
OCA	Появление новых данных DO на выходе после фронта RCLKS	8.1		3	нс	
RPCA	Появление сигнала ошибки четности чтения после фронта RCLKS	11.4		3.9	нс	
Idyn	Ток	7.34	0.261	2.32	мА	

Таблица П.7 - Параметры временных диаграмм синхронного чтения.



Синхронная запись. Результаты моделирования приведены на рисунке П.19, временная диаграмма – на рисунке П.20, ее параметры в таблице П.8.

Рисунок П.19 - Результаты моделирования



Рисунок П.20 - Временная диаграмма

Символ	Описание	Макс	Мин.	Тип.	Единицы	Примечания
CCYC	Время цикла	7.5		7.5	нс	
СМН	Время нахождения синхросигнала в 1		3.75	3.75	нс	
CML	Время нахождения синхросигнала в 0		3.75	3.75	нс	
DCS	Установка данных перед фронтом WCLKS		1	1	нс	
DCH	Удержание данных после фронта WCLKS	1.8		0.7	нс	
WDCS	Установка адреса перед фронтом WCLKS		0	1	нс	
WACH	Удержание адреса после фронта WCLKS	1.3		0.5	нс	
WRCS, WBCS	Установка WRB&WBLKB перед фронтом WCLKS		1	1	нс	
WRCH, WBCH	Удержание WRB&WBLKB после фронта WCLKS	0.5		0.5	нс	
WPCA	Появление сигнала ошибки четности записи после фронта WCLKS	5.6		2.1	нс	
Idyn	Ток	7.63	0.302	1.7	мА	

Таблица П.8 - Параметры временных диаграмм

Реализация блоков ПЗУ для БМК на библиотеке стандартных ячеек

Для реализации ПЗУ рекомендуется использовать средства Verilog языка - *case* конструкцию.

В связи с тем, что при реализации ПЗУ имеется большое число фиксированных состояний (таблица «1» и «0») использование стандартных программных средств проектирования позволяет минимизировать физический объем проектируемого ПЗУ. Реализуемое таким образом ПЗУ, на базе ячеек ядра получается наиболее компактным и быстродействующим. Все современные средства синтеза пректа имеют хорошие способности обрабатывать и понимать такие конструкции именно как ПЗУ блоки.

Пример проектирования блока ПЗУ с произвольным содержимым ячеек для блока имеющего 1024 8 битных слов приведён ниже.

Verilog код блока:

module rom1024x8 (clk, addr, en, dout);

inputclk;

input [9:0] addr; *input* en; *output* [7:0] dout;

reg [9:0] radr; reg [7:0] rom;

always @(posedgeclk)

if (en) radr<= addr;

always @(radr)

case (radr)

10'h000: rom = 8'h06;10'h001: rom = 8'h00;10'h002: rom = 8'h2F; 10'h003: rom = 8'h03; 10'h004: rom = 8'h01;10'h005: rom = 8'h15;10'h006: rom = 8'h00; 10'h007: rom = 8'h14;10'h008: rom = 8'h12;10'h009: rom = 8'h14; 10'h00A: rom = 8'h11; 10'h00B: rom = 8'h01;10'h00C: rom = 8'h0F;10'h18D: rom = 8'h21; 10'h18E: rom = 8'h03:10'h18F: rom = 8'h17;10'h190: rom = 8'h8F;

10'h191: rom = 8'h0B;

.....

(далее расписывается вся таблица кодов, заносимая в ПЗУ. Убрано для уменьшения объёма текста)

10'h192: rom = 8'h02; 10'h193: rom = 8'h03; 10'h194: rom = 8'h17; 10'h195: rom = 8'h94; 10'h196: rom = 8'h0B; 10'h197: rom = 8'h02; default: rom = 8'h00; endcase assigndout = rom; endmodule

Простота реализации и компактность записи в проекте показывает эффективность данного подхода к проектированию ПЗУ.

Размер в логических ячейках получившейся схемы:

Тип яче	ейки Кол-во	
sequential inverter logic	10 17 1066	тригтера инверторы комбинаторика
 total	1094	Итого 1094 ячейки

В качестве примера синтезировано таким способом простейшее ПЗУ размерами 4адреса на 4разряда.

На рисунке П.21 представлена схема реализации описанного выше ПЗУ.



Рисунок П.21 - Схема реализации описанного выше ПЗУ

Умножитель частоты, блок PLL

Структурная схема разработанного блока PLL приведена на рисунке П.22.



Рисунок П.22 - Структурная схема

В своем составе блок PLL содержит следующие блоки:

- 1. Генератор, управляемый напряжением,
- 2. Делитель частоты,
- 3. Блок сравнения фаз,
- 4. Интегратор,
- 5. Блок блокировки выходной частоты.

Конденсаторы либо высокочастотные керамические, либо сдвоенные. В случае сдвоенных конденсаторов, один из них обязательно должен быть высокочастотный керамический емкостью не менее 10 нФ. Шунтирующие конденсаторы должны располагаться на плате в непосредственной близости к соответствующим выводам микросхемы.

Функциональное описание и режимы работы блока PLL.

Разработанная схема позволяет умножать опорную входную частоту на 8 коэффициентов. Все коэффициенты представлены в таблице П.9.

			-		
№ п/п	Коз	оффици	ент	Умножение	
	Sel1	Sel2	Sel3		
1	0	0	0	32	

Таблица П.9 - Коэффициенты умножения

2	0	0	1	2
3	0	1	0	128
4	0	1	1	8
5	1	0	0	64
6	1	0	1	4
7	1	1	0	256
8	1	1	1	16



Рисунок П.23 - Электрическая схема PLL

Результаты моделирования блока PLL.

На рисунке П.25 представлены графики максимальной выходной частоты блока PLL. Моделирование происходило при 3 различных температурах: -60 °C, +27 °C, +120 °C, и при 3 напряжениях питания: +1,6 B, +1,8 B, +2 B.

Для проверки максимальной частоты использовалась схема включения, приведенная на рисунке П.24. Особенность данной схемы включения заключается в том, что на цепь обратной связи (вывод rc) подается максимально возможное напряжение (vdd). В этом случае ГУН будет выдавать максимальную частоту сразу.



Рисунок П.24 - Схема включения для проверки максимальной частоты



Рисунок П.25 - Максимальная частота при VDD=1,8 В Т=27 °C

Результаты моделирования максимальной частоты блока PLL представлены в таблице П.10.

Таблица П.10 Результаты моделирования максимальной частоты -60 27 120

T, °C	-60			-60 27			120		
VDD, B	1,6	1,8	2	1,6	1,8	2	1,6	1,8	2
F, МГц	197,4	245,2	289,6	172,1	210,7	245,1	153,6	183,9	211,3

Как видно по таблице П.10 минимальная частота составляет 153,6 МГц при температуре 120 °С и напряжению питания 1,6 В. Дополнительно также следует учесть, что данная частота является максимально возможной (напряжение в цепи обратной связи равной напряжению питания). Рекомендуется отступить от данной границы минимум 20 % (то есть выбирать режим работы при котором напряжение RC составляет примерно 1,45 В рисунок П.25). Следовательно, как видно на рисунке П.25 максимальная выходная частота блока PLL составляет 145 МГц. Что касается минимальной частоты, то рекомендуется использовать режим, при котором напряжение на RC составляет не менее 0,8 В. Результаты моделирования приведены в таблице П.11.

Таблица П.11 - Результаты моделирования минимальной частоты

T, °C	-60			, °C -60 27				120	
VDD, B	1,6	1,8	2	1,6	1,8	2	1,6	1,8	2
F, МГц	7,77	6,77	5,74	18,3	16	13,8	27,6	24,4	24,4

Разработка блоков приемника и передатчика LVDS.

Структурная схема разработанного блока LVDS приемника и передатчика приведена на рисунке П.26.



Рисунок П.26 - Структурная схема LVDS

Таблица П.12 - Таблица выводов блоков LVDS приемника и передатчика

Наименование	Назначение вывода				
вывода					
gnd	Вывод отрицательного питания или общий				
vdd	Вывод положительного питания				
In	Вход LVDS передатчика (лог. Уровни 0 и vdd)				
Out_n	«Отрицательный» выход LVDS передатчика				
Out_p	«Положительный» выход LVDS передатчика				
In_p	«Положительный» вход LVDS приемника				
In_n	«Отрицательный» вход LVDS приемника				
Out	Выход LVDS приемника				

LVDS использует дифференциальный сигнал с небольшим размахом. На рисунке П.27 показана типовая упрощенная схема линии связи стандарта LVDS. Выходной каскад передатчика содержит источник тока (типовое значение 3,5 мА), а на входе приемника, имеющего большое входное сопротивление, стоит согласующий резистор 100 Ом, на котором при протекании тока возникает напряжение размахом 350 мВ. При переключении состояния передатчика направление тока изменяется, что позволяет приемнику четко определять логические «нуль» и «единицу».



Рисунок П.27 - Типовая упрощенная схема линии связи стандарта LVDS

Конденсаторы либо высокочастотные керамические, либо сдвоенные. В случае сдвоенных конденсаторов, один из них обязательно должен быть высокочастотный керамический емкостью не менее 10 нФ. Шунтирующие конденсаторы должны располагаться на плате в непосредственной близости к соответствующим выводам микросхемы

Результаты моделирования блока LVDS.

Схема моделирования представлена на рисунок П.28



Рисунок П.28 - Схема моделирования

На рисунке П.29 представлен график работы при входной частоте 1 МГц.



Рисунок П.29 - График работы при входной частоте 1 МГц Т=27 °С, VDD=1,8 В

Результаты моделирования LVDS представлены в таблицах П.13 и П.14.

			2		1				
T, ⁰C	-60			27			120		
VDD, B	1,6	1,8	2	1,6	1,8	2	1,6	1,8	2
Udiff,мВ	272,7	373	473,4	262,6	350,1	434,3	245	318,9	389,9
Usin, B	1,027	1,161	1,293	1,027	1,151	1,274	1,022	1,14	1,262

Таблица П.13 Результаты моделирования LVDS на частоте 250 МГц

Таблица П.14 Результаты моделирования LVDS на частоте 1 МГц

			•		-				
T, °C		-60			27			120	
VDD, B	1,6	1,8	2	1,6	1,8	2	1,6	1,8	2
Udiff,мВ	262,8	369,7	476	261,6	349,4	438,3	245,6	320,1	396

Использование ЈТАС интерфейса в БМК

На рисунке П.30 показана типовая схема данного интерфейса (из стандарта: IEEE Std 1149.1-2001)



Рисунок П.30 - Типовая схема ЈТАС интерфейса БМК

Обычно данный порт используется для тестирования подключения микросхем на плате. Каждая БИС («зашивка» БМК) требует генерации е уникального узла JTAG т.к. каждая БИС имеет специфические входные и выходные контакты, и их расположение. В связи с этим наиболее целесообразно для каждой БИС формировать свои уникальные цепи узла

JTAG. Далее рассмотрены способы генерации узла JTAG при синтезе схемы проекта в САПР Cadence.

Вставка JTAG требует наличия в библиотеке ряда ячеек: В библиотеке ядра: latch (триггер-защелка по уровню). В библиотеке ячеек ввода-вывода: для порта TDO – ячейка с третьим состоянием. Входные порты для TDI, TMS, TRST с подтяжкой к уровню логической единицы. Все элементы – ячейки, блоки (сдвиговый регистр, мультиплексоры регистры) присутствуют в тестовой микросхеме 5540TH014AДля вставки TAP контроллера и Boundary Scan ячеек в маршруте используется программа Genus (от Cadence Design Systems)

Ниже приведен стандартный маршрут вставки генерации тестов для контроля узла JNFG. Далее приведены команды для программы Genus:

Определение внешних портов для JTAG:

define_dft tap_port TDI -type tdi -hookup_pin TDI_IO/Y
\${DESIGN_NAME}

define_dft tap_port TCK -type tck -hookup_pin TCK_IO/Y
\${DESIGN NAME}

define_dft tap_port TMS -type tms -hookup_pin TMS_IO/Y
\${DESIGN NAME}

define_dft tap_port TRST -type trst -hookup_pin TRST_IO/Y
\${DESIGN_NAME}

define_dft tap_port TDO -type tdo -hookup_pin TDO_IO/A
\${DESIGN_NAME}

Определение JTAG инструкций (из JTAG спецификации) define_dft jtag_instruction_register -name JTAG_IR -length 3; define_dft jtag_instruction -name IDCODE -opcode 101 -register DEVICE_ID \

-length 32 -capture 11110000000000000000001101001111;

define_dft jtag_instruction -name SCANMODE -opcode 100 -register BYPASS;

> define_dft jtag_instruction -name EXTEST -opcode 010; define_dft jtag_instruction -name BYPASS -opcode 111; define_dft jtag_instruction -name PRELOAD -opcode 001; define_dft jtag_instruction -name SAMPLE -opcode 001;

Вставка JTAG в нетлист проекта *insert_dft boundary_scan –design \$DESIGN_NAME* Генерация тестовых векторов для JTAG интерфейса *write_et_bsv -directory out/jtag/22july_et_atpg_jtagscan_\${DESIGN_NAME} -library* "~/libs/Lib GAL v4 30.v" \$DESIGN NAME

Приложение 2. Характеристики типовых аналоговых схем с применением предложенных моделей



резисторов на переключаемых конденсаторах















Приложение 3. Электрические схемы аналоговых

библиотечных элементов аналого-цифрового БМК

Наименование элемента: <u>Операционный усилитель общего применения</u> (OY mun 1)

Операционный усилитель общего применения реализован в виде дифференциального операционного усилителя с rail-to-rail входом и каскадной нагрузкой. Выбранная конструкция ОУ обеспечивает коэффициент усиления более 70 дБ, запас по фазе более 60° и напряжение смещение менее 10 мВ, что полностью удовлетворяет ТЗ. Символ и электрическая схема операционного усилителя представлена на рисунке П.31, выводы ОУ – в таблице 5.15.





Рисунок П.31 – Символ и электрическая схема ОУ общего применения

Вывод	Назначение	Описание		
VDD	Вывод	Вывод напряжения питания		
IB	Вывод	Вывод установки тока		
		смещения		
VSS	Общий вывод	Общий вывод		
INP	Вход	Неинвертирующий вход		
INM	Вход	Инвертирующий вход		
OUT	Выход	Выход ОУ		

Таблица П.15 – Выводы ОУ общего применения

В рамках использования операционного усилителя общего применения при построении электрических схем устройств на основе аналого-цифрового БМК возможна параметризация (подстройка) блока ОУ: путем подключения/отключения конденсаторов в цепях частотной коррекции для регулировки частоты единичного усиления и требуемого запаса по фазе, а также путем подстройки номинала токозадающего резистора в схеме формирования токов смещения для регулировки скорости нарастания выходного сигнала операционного усилителя.

Наименование элемента: <u>Прецизионный операционный усилитель (OV</u> <u>mun 2)</u>

Прецизионный операционный усилитель реализован в виде дифференциального двухкаскадного операционного усилителя с входной составной парой на р-МОП транзисторах на токовых зеркалах с активным регулированием и пониженным напряжением согласования. Выбранная конструкция ОУ обеспечивает запас по фазе больше 60°, коэффициента усиления более 90 дБ и напряжение смещения менее 3 мВ.

При использовании прецизионного операционного усилителя для построения электрических схем устройств на основе аналого-цифрового БМК возможна параметризация (подстройка) блока ОУ: путем подключения/отключения конденсаторов в цепях частотной коррекции для регулировки частоты единичного усиления и требуемого запаса по фазе, а также путем подстройки номинала токозадающего резистора в схеме

321

формирования токов смещения для регулировки скорости нарастания выходного сигнала операционного усилителя.

На рисунке П.32 представлены символ и электрическая схема прецизионного операционного усилителя.





Рисунок П.32 – Символ и электрическая схема прецизионного ОУ

Наименование элемента: Б<u>ыстродействующий полностью</u> <u>дифференциальный операционный усилитель (OV mun 3)</u>

Операционный усилитель реализован в виде полностью дифференциального телескопического операционного усилителя с инвертирующими и неинвертирующими входами и выходами. Выбранная конструкция ОУ обеспечивает напряжение смещения менее 16 мВ и позволяет получить скорость нарастания более 120 В/мкс.

При использовании полностью дифференциального операционного усилителя для построении электрических схем устройств на основе аналого-

цифрового БМК возможна параметризация (подстройка) блока ОУ: путем подключения/отключения конденсаторов в цепях частотной коррекции для регулировки частоты единичного усиления и требуемого запаса по фазе, а также путем подстройки номинала токозадающего резистора в схеме формирования токов смещения для регулировки скорости нарастания выходного сигнала операционного усилителя. На рисунке П.33 представлены символ и электрическая схема прецизионного операционного усилителя.





Рисунок П.33 – Символ и электрическая схема быстродействующего полностью дифференциального ОУ

Наименование элемента: <u>Компаратор общего применения (Комп.</u> <u>mun 1)</u>

Компаратор общего применения реализован в виде блока сравнения аналоговых сигналов с одним (инверсным) выходом. Компаратор по входу rail-to-rail с каскадной нагрузкой. Выбранная конструкция позволяет захватить весь входной диапазон и обеспечить задержку переключения менее 0,8 мкс. Электрическая схема компаратора представлена на рисунке П.34.


Рисунок П.34 – Электрическая схема компаратора общего применения

При использовании схемы компаратора общего применения для построения электрических схем устройств на основе аналого-цифрового БМК возможна параметризация (подстройка) блока компаратора: путем параллельного подключения/отключения транзисторов в выходном буфере для регулировки драйверной способности компаратора, а также путем подстройки номинала токозадающего резистора в схеме формирования токов смещения для регулировки быстродействия компаратора.



Рисунок П.35 – Разрешающая способность К1

Наименование элемента: <u>Быстродействующий компаратор (Комп.</u> <u>mun 2)</u> Компаратор реализован в виде пяти каскадного блока

быстродействующего компаратора для сравнения аналоговых сигналов

(рисунок П.36). Данная конструкция обеспечивает задержку переключения компаратора менее 0,4 мкс, что полностью удовлетворяет ТЗ (рисунок 5.78).



Рисунок П.36 – Электрическая схема стробируемого компаратора

При использовании схемы быстродействующего компаратора для построения электрических схем устройств на основе аналого-цифрового БМК возможна параметризация (подстройка) блока компаратора: путем подстройки номинала токозадающего резистора в схеме формирования токов смещения для регулировки быстродействия компаратора. Питания аналогового ядра компаратора и цифрового выхода разделены.



Рисунок П.37 – Задержка переключения быстродействующего компаратора

Наименование элемента: <u>Компаратор общего применения с</u> <u>подстройкой смещения нуля (Комп. тип 3)</u>

Компаратор общего применения реализован в виде блока сравнения аналоговых сигналов с одним (инверсным) выходом. Компаратор по входу rail-to-rail с каскодной нагрузкой. Выбранная конструкция позволяет захватить весь входной диапазон и обеспечить задержку переключения менее 0,8 мкс. Электрическая схема компаратора представлена на рисунке П.38. В компаратор общего применения встроена функция калибровки смещения нуля, что позволяет получить напряжение смещения менее 3 мВ, требуемое по ТЗ (рисунок П.39).



Рисунок П.38 – Электрическая схема компаратора общего применения с калибровкой смещения нуля

Главная особенность этого блока, то что верхняя дифференциальная пара п-МОП и нижняя пара н-МОП транзисторов калибруется отдельно. П-МОП калибруется ближе к напряжению «земли», н-МОП в середине диапазона питания. С обеими дифференциальными парами соединены дополнительный блоки подстройки.



Рисунок П.39 – Общая схема компаратора с блоком калибровки

Калибровка происходит следующем образом (рисунки П.40, 5.41):

Вначале калибруется смещение p-дифференциальной пары, затем калибруется смещение n-дифференциальной пары. В процессе к двум плечам

дифференциальной пары добавляются «довески» из таких же транзисторов, компенсируя смещение в ту или другую сторону.



Рисунок П.40 – Общая диаграмма калибровки



Рисунок П.41 – Тактовая диаграмма процесса калибровки

Наименование элемента: <u>ОУ общего применения с подстройкой</u> <u>смещения нуля (OV mun 4)</u>

Операционный усилитель общего применения реализован в виде дифференциального операционного усилителя с rail-to-rail входом и токовыми зеркалами с каскадной нагрузкой. Достигнуть требуемое по T3 значение смещение нуля позволяет функция калибровки. Электрическая схема ОУ представлена на рисунке П.42.



Рисунок П.42 – Электрическая схема ОУ общего применения с калибровкой смещения нуля

Главная особенность этого блока, то что верхняя дифференциальная пара п-МОП и нижняя пара н-МОП транзисторов калибруется отдельно. П-МОП калибруется ближе к напряжению «земли», н-МОП в середине диапазона питания. С обеими дифференциальными парами соединены дополнительные блоки подстройки (рисунки П.43, П.44).



Рисунок П.43 – Блок подстройки дифференциальной пары



Рисунок П.44 – Общая схема ОУ общего применения с блоком калибровки смещения нуля

До калибровки смещение 7 мВ/ После калибровки 80 мкВ.

Наименование элемента: <u>Цифро-аналоговый преобразователь (ЦАП)</u>

Цифро-аналоговый преобразователь, возможный к реализации и использованию в аналого-цифровом БМК, представляет из себя ЦАП на основе резистивных R-2R матриц. Типовая структура ЦАП данного типа состоит из набора резисторов и набора ключей и взята из предыдущих итерация ЦАП на БМК.

Электрическая схема ЦАП представлена на рисунке П.45.



Рисунок П.45 – Электрическая схема цифро-аналогового преобразователя

ЦАП состоит из согласованной резистивной матрицы, набора ключей, цифровой схемы на базе стандартных ячеек, реализующей дешифратор для осуществления сегментирования 5 старших значащих разрядов, тактируемого входного регистра для записи текущего состояния управляющих сигналов на аналоговые ключи (запись входного цифрового кода) и блока коррекции тока опорного напряжения.

На рисунке П.46 представлена схема блока коррекции тока опорного напряжения.

Блок коррекции позволяет выровнять ток в момент переключения ЦАП на протяжении всей шкалы, улучшая его характеристики.



Рисунок П.46 – Схема блока коррекции тока

На кристалле в непосредственной близости с описанными элементами электрической схемы ЦАП расположен аналоговый модуль и набор резисторов, которые могут быть использованы для буферизации выходного напряжения ЦАП или его масштабирования при необходимости.

Наименование элемента: АЦП последовательного приближения

Структура АЦП взята из предыдущих итераций АЦП на БМК. Электрическая схема АЦП последовательно приближения представлена на рисунке П.49.



Рисунок П.47 – Электрическая схема АЦП последовательного приближения

Запуск процедуры преобразования осуществляется с помощью подачи сигнала ADC_START в виде положительного импульса (логической единицы) продолжительностью не менее 1 периода тактовой частоты сигнала CLK. В течении времени, равного длительности импульса ADC_START, происходит выборка входного сигнала встроенным в АЦП УВХ. По срезу сигнала ADC_START начинается фактическая процедура преобразования, включающая в себя задержку на установление УВХ, и последовательное формирование выходного кода.

Время преобразования не превышает 14 периодов сигнала ADC CLK ADC START. По завершению после сигнала среза процедуры преобразования на выводе ADC READY формируется положительный (логическая единица). Одновременно фронтом сигнал с сигнала ADC_READY на выводах ADC_Q12 ... ADC_Q1 формируется выходной цифровой код, эквивалентный аналоговому напряжению в УВХ.

Наименование элемента: Линейный регулятор напряжения

В основе конструкции линейного регулятора лежит стековая структура, обеспечивающая входное напряжение более 10 В (рисунок П.48). Входная стабилизирующая дифференциальная пара сделана на р транзисторах, из-за опорного уровня 1 В. Линейный регулятор предназначен для стабилизации выходного напряжения.

Вход 10 В, выход 3.3 В.



Рисунок П.48 - Электрическая схема линейного регулятора

Наименование элемента: Аналоговый мультиплексор

Аналоговый мультиплексор предназначен для коммутации аналоговых сигналов.

Электрическая схема блока многоканального аналогового мультиплексора представлена на рисунке П.49.



Рисунок П.49 – Электрическая схема аналогового мультиплексора Многоканальный аналоговый мультиплексор состоит из 8 аналоговых ключей.

Электрическая схема аналогового ключа представлена на рисунке П.50.



Рисунок П.50 - Электрическая схема аналогового ключа

Наименование элемента: <u>Коммутационный тестовый блок цифровой</u> <u>части</u>

Коммутационный тестовый блок состоит из серии последовательно соединённых ячеек и представлен на рисунке П.51.



Рисунок П.51 – Коммутационный тестовый блок

Приложение 4. Аналого-цифровые базовые матричные кристаллы, в которых реализованы положения и

Наименование / назначение / область применения	Особенности
Аналого-цифровые базовые матричные	1. Возможность применения двух типов корпусов: 108-ми выводного и
кристаллы 5400ТР014 и	48-ми выводного.
5400ТР015 являются основой для	2. Наличие встроенных «жестких» СФ блоков: ИОН, АЦП (12 и 14 бит),
проектирования и изготовления полузаказных	ЦАП, различные типы ОУ и компараторов.
аналоговых и аналого-цифровых СБИС средней	3. Наличие матриц согласованных аналоговых компонентов для
сложности.	построения произвольных схем.
Предназначены для применения в устройствах	4. Емкость цифровой части ~100 тыс. вентилей позволяет реализовывать
автоматики и аналоговой/аналого-цифровой обработки	цифровые схемы средней сложности.
(разрядность 10-16 бит, частота дискретизации до 5	5. Номинальное напряжение питания – 3,3В
МВыб/c).	
Аналого-цифровые базовые матричные	1. Возможность применения компактных безвыводных корпусов двух
кристаллы 5400ТР045 и 5400ТР045А являются основой	типов: 48-ми выводного и 28-ми выводного.
для проектирования и изготовления полузаказных	2. Архитектура аналоговой части в виде идентичных матриц
микросхем в компактных безвыводных корпусах	согласованных аналоговых компонентов для построения
и предназначены для замены отдельных дискретных	«хорошую» мультиплицируемость библиотечных элементов.
компонентов импортного производства (ОУ,	3. Наличие библиотеки аналоговых СФ блоков
компараторы, АЦП, ЦАП), а также для построения	4. Емкость цифровой части ~30 тыс. вентилей (1.8В) и ~50 тыс. вентилей
аналого-цифровых схем малой и средней сложности.	(5В) позволяет реализовывать цифровые схемы малой и средней
Могут применяться в устройствах автоматики и	сложности.
аналоговой/аналого-цифровой обработки (разрядность	5. Наличие конфигурационного ПЗУ (370 бит) для реализации
10-14 бит, частота дискретизации до 1 МВыб/с).	внутренних программируемых настроек олоков при реализации схем на БМК (в т.н. упрошает процесс настройки кристациов на пластине)
	6. Номинальное напряжение питания – 5В
Аналого-цифровые базовые матричные	1. Возможность применения компактных корпусов трех типов: 48-ми
кристаллы 5400ТР054, 5400ТР055 и 5400ТР055А	выводного и 28-ми выводного и 16 выводного.
являются основой для проектирования и изготовления	2. Архитектура аналоговой части в виде идентичных матриц

результаты работы

полузаказных микросхем в компактных корпусах, имеющих в своем составе высоковольтные компоненты (ключи, мультиплексоры, ОУ). Предназначена для замены отдельных дискретных компонентов импортного производства, а также для построения схем малой и средней сложности с высоковольтными опциями.	 3. 4. 5. 6. 7. 	согласованных аналоговых компонентов на основе 5В-ых транзисторов для построения произвольных низковольтных аналоговых схем без «жестких» СФ блоков обеспечивает «хорошую» мультиплицируемость библиотечных элементов. Аналоговые транзисторные матрицы позволяют реализовывать и цифровые схемы средней сложности. Эквивалентная емкость цифровой части ~100 тыс. вентилей (5В). Наличие в составе БМК транзисторов с рабочим напряжением не менее 30В для построения отдельных высокольтных блоков. Наличие библиотеки аналоговых СФ блоков: низковольтных и высоковольтных. Наличие конфигурационного ПЗУ (не менее 32 бит) для реализации внутренних программируемых настроек блоков при реализации схем на БМК (в т.ч. упрощает процесс настройки кристаллов на пластине) Номинальное напряжение питания – 5В.
Аналого-цифровые базовые матричные кристаллы 5400TP064 и 5400TP065 являются основой для проектирования полузаказных схем и содержат в своем составе ОУ различных типов, компараторы, встроенные АЦП и ЦАП, блок однократно программируемой памяти, драйверы силовых ключей. Предназначены для разработки аналого- цифровых проектов средней и высокой сложности (многоканальные системы, АЦ-преобразование, цифровая обработка и т.д.)	1. 2. 3. 4. 5. 6. 7.	Возможность применения корпусов двух типов: 108-ми выводного и 48-ми выводного. Архитектура аналоговой части в виде идентичных матриц согласованных аналоговых компонентов для построения произвольных схем без «жестких» СФ блоков обеспечивает «хорошую» мультиплицируемость библиотечных элементов. Наличие библиотеки аналоговых СФ блоков Емкость цифровой части ~300 тыс. вентилей (1.8В) и ~200 тыс. вентилей (5В) позволяет реализовывать цифровые части средней и повышенной сложности. Наличие ПЗУ емкостью 1 кбайт Наличие конфигурационного ПЗУ (460 бит) для реализации внутренних программируемых настроек блоков при реализации схем на БМК (в т.ч. упрощает процесс настройки кристаллов на пластине) Номинальное напряжение питания – 5В
Аналого-цифровые базовые матричные	1.	Возможность применения корпусов двух типов: 108-ми выводного и 48-ми выволного.
для проектирования и изготовления полузаказных	2.	Архитектура аналоговой части в виде идентичных матриц

микросхем, имеющих в своем составе высоковольтные компоненты (ключи, мультиплексоры, ОУ). Предназначены для разработки аналого- цифровых проектов средней и высокой сложности (многоканальные системы, АЦ-преобразование,	3.	согласованных аналоговых компонентов на основе 5В-ых транзисторов для построения произвольных низковольтных аналоговых схем без «жестких» СФ блоков обеспечивает «хорошую» мультиплицируемость библиотечных элементов. Наличие в составе БМК транзисторов с рабочим напряжением не менее 30В для построения отдельных высокольтных блоков.
цифровая обработка и т.д.) с высоковольтными опциями.	 4. 5. 6. 7. 	Наличие библиотеки аналоговых СФ блоков: низковольтных и высоковольтных. Емкость цифровой части ~200 тыс. вентилей (1.8В) для реализации цифровых частей средней сложности. Наличие конфигурационного ПЗУ (не менее 110 бит) для реализации внутренних программируемых настроек блоков при реализации схем на БМК (в т.ч. упрощает процесс настройки кристаллов на пластине) Номинальное напряжение питания – 5В.
Аналого-цифровые базовые матричные кристаллы 5400TP085 являются основой для проектирования полузаказных схем и предназначены для реализации высоковольтных драйверов силовых ключей, линейных и импульсных регуляторов, схем управления двигателями и	1. 2.	48-выводной корпус. Архитектура аналоговой части в виде идентичных матриц согласованных аналоговых компонентов на основе 5В-ых для построения произвольных низковольтных и высоковольтных аналоговых схем для построения произвольных схем позволяет реализовывать основные аналоговые устройства – АЦП, ЦАП, ИОН, ОУ и т.д.
аналогичных. АЦ БМК предназначен для систем, функционирующих с напряжениями до 30 В и токами нагрузки до 2,0 А.	 3. 4. 5. 6. 7 	Аналоговые транзисторные матрицы позволяют реализовывать и цифровые схемы средней сложности. Эквивалентная емкость цифровой части ~100 тыс. вентилей (5В). Наличие специализированных матриц высоковольтных транзисторов, на основе которых возможна реализация силовых ключей с напряжениями до 30 В и токами нагрузки до 2,0 А. Наличие библиотеки аналоговых СФ блоков: низковольтных и высоковольтных. Наличие конфигурационного ПЗУ для реализации внутренних программируемых настроек блоков при реализации схем на БМК (в т.ч. упрощает процесс настройки кристаллов на пластине) Номинали ное напряжение питания.

Аналого-цифровой базовый матричный кристалл	1.	48-выводной корпус.
5400ТР105 является основой для	2.	Преобладание в архитектуре БМК цифровой части и элементов
проектирования полузаказных схем и предназначен для		памяти по отношению к аналоговым компонентам.
выполнения цифровой обработки сигналов и реализации	3.	Емкость цифровой части ~300 тыс. вентилей (1.8В) и ~300 тыс.
сложных нифровых вычислительных устройств с		вентилей (5В) для реализации цифровых частей средней и
	4	повышенной сложности.
	4.	Наличие 113У емкостью 4 коаит.
микроконтроллерного ядра, олоков ОЗУ и 113У.	Э. С	Наличие отладочного ОЗУ емкостью 4 коаит.
	0. 7	Наличие ОЗУ емкостью 250 оаит.
Отличительной особенностью микросхем	1.	Наличие конфигурационного 1139 (430 ойт) для реализации
является низкий ток потребления элементной базы и		внутренних программируемых настроек олоков при реализации схем
наличие необходимого числа аналоговых блоков для	8	на вили (в 1.4. упрощает процесс настроики кристаллов на пластине).
реализации аналого-цифрового и цифро-аналогового	0.	
интерфейса.		
1 1		ироизвольных слем позволяет реализовывать основные аналоговые устройства – АШП ПАП ИОН ОУ и т п
	9	Наличие библиотеки микропотребляющих аналоговых СФ блоков.
	10	. Номинальное напряжение питания – 5В.
Аналого-цифровые базовые матричные	1.	Возможность применения корпусов двух типов: компактного 6-ти
кристаллы 5400ТР125 являются основой лля		выводного и 28-ми выводного (используемых выводов -14).
проектирования полузаказных схем и предназначены	2.	Емкость БМК ~15 тыс. универсальных 4-х транзисторных вентилей
		(5В) для реализации как цифровых частей малой сложности, так и
для реализации компактных стоиких вспомогательных		аналоговых блоков.
микросхем: супервизоры питания, линеиные регуляторы	3.	Наличие конфигурационного ПЗУ (320 бит) для реализации
напряжения, трансляторы цифровых уровнеи, источники		внутренних программируемых настроек блоков при реализации схем
опорного напряжения, цифровые буферы и прочее.		на БМК (в т.ч. упрощает процесс настройки кристаллов на пластине).
	4.	В составе БМК присутствуют резисторы, диоды и конденсаторы, что
		вместе с транзисторами позволяет реализовывать основные
		аналоговые устройства – ИОН, ОУ, а также АЦП или ЦАП.
	5.	Наличие библиотеки специализированных СФ блоков.
	6.	Номинальное напряжение питания – 5В.

Приложение 5. Акты внедрения



Россия, 124460, Москва, Зеленоград, улица Академика Валиева, дом 6, стр. 1 тел.:+7 495 229 7000 факс +7 495 229 7773, e-mail: niime@niime.ru, web: www.niime.ru

УТВЕРЖДАЮ Генеральный директор АО «НИИМЭ», академик РАН Г.Я. Красников «25» 02 2022г.

Акт внедрения

результатов диссертационной работы В.И. Эннса «Методы и средства разработки специализированных гетерогенных конфигурируемых интегральных схем для вычислительной техники и систем управления» на соискание ученой степени доктора технических наук

Предложенные в работе Эннса Виктора Ивановича технические решения, модели и методы проектирования были применены при выполнении ряда НИОКР. В рамках ОКР «Логика-2», «Алмаз-14», «Логика-И2» были анализа программного методы предварительного И использованы прототипирования, которые позволили улучшить эффективность трассировочных ресурсов и уменьшить площадь кристаллов программируемых интегральных схем не менее чем на 20% при сохранении ключевых параметров. В рамках ОКР «Алмаз-1», «Алмаз-13», «Логика-И5-РК» были разработаны базовые матричные кристаллы, «зашивки» которых выполняются с высокой обладают дополнительными эффективностью И экономической функциональными возможностями, разработанными в рамках диссертационной Использование формализованных описаний аналоговых блоков, работы. приведенных в диссертации, в ОКР «Преобразователь-3» и «Алмаз-10» позволило ускорить проектирование «зашивок» в несколько раз.

Зам. генерального директора,

главный конструктор

А.С. Кравцов

УТВЕРЖДАЮ Генеральный конструктор АО, «НПО Автоматики» ALC: С.Ф. Дерюгин 02 2022 r. IN DEB 5.55

АКТ ВНЕДРЕНИЯ

результатов диссертационной работы Эннса Виктора Ивановича на соискание ученой степени доктора технических наук

«Методы и средства разработки специализированных гетерогенных конфигурируемых интегральных схем для вычислительной техники и систем управления»

Акт внедрения свидетельствует о том, что на предприятии Акционерное общество «НПО Автоматики» были использованы результаты диссертации Эннса Виктора Ивановича. В предложенные в работе методы решения задачи комплектования частности, специализированной аппаратуры электронной компонентной базой использованы при разработке «зашивок» и внедрению в аппаратуру микросхем базовых матричных кристаллов позволили ускорить создание аппаратуры методы Предложенные 55010БЦЗТ. ШЮГИ.468332.093 на базе конфигурируемых микросхем, улучшить ряд TTX и повысить надежность ее функционирования.

Заместитель генерального конструктора

Н.В. Дудин



АО «ДИЗАЙН ЦЕНТР «СОЮЗ» ИНН/КПП: 7735143270/773501001 ОГРН: 1157746403033 124482, г. Москва, г. Зеленоград, корпус 100 Телефон/Факс: 8 (499) 995-25-18 E-mail: mail@dcsoyuz.ru Web: https://dcsoyuz.ru УТВЕРЖДАЮ Заместитель генерального директора по науке АО «Дизайн Центр «СОЮЗ»

why operful 2022 r.

Кобзев Ю.М.

АКТ ВНЕДРЕНИЯ

результатов диссертационной работы Эннса Виктора Ивановича на соискание ученой степени доктора технических наук.

Тема диссертации: «Методы и средства разработки специализированных гетерогенных конфигурируемых интегральных схем для вычислительной техники и систем управления»

Настоящий акт составлен в том, что на предприятии Акционерное общество «ДИЗАЙН ЦЕНТР «СОЮЗ» в 2017 – 2022 гг. были внедрены следующие результаты диссертационной работы В.И. Эннса:

- методы предварительного анализа и программного прототипирования;
- маршрут разработки гетерогенных конфигурируемых интегральных схем;

 методы разработки аналоговых и аналого-цифровых конфигурируемых интегральных схем с использованием моделей и формализованных описаний аналоговых блоков.

Указанные результаты были использованы при разработке и освоении серийного производства аналого-цифровых конфигурируемых интегральных схем серий 5400TP014, 5400TP015, 5400TP035, 5400TP045, 5400TP045A, 5400TP064, 5400TP065, а также для создания проблемно-ориентированных и универсальных микросхем на их основе.

Применение результатов работы В.И. Эннса позволило уменьшить площадь базовых кристаллов конфигурируемых интегральных схем не менее чем на 20% при сохранении их функциональных возможностей и сократить сроки проектирования не менее чем в два раза.

Начальник отдела топологического проектирования

Кирдянов А.А.



Акционерное общество «Научно-исследовательский институт «Субмикрон» (АО «НИИ «Субмикрон»)

124498, г. Москва, г. Зеленоград, Георгиевский проспект, д. 5, стр. 2, этаж 4, помещение I, комната 50 тел. (499) 731-89-31;факс (499) 731-27-53;e-mail: <u>submicron@se.zgrad.ru</u>

ОКПО 07520671; ИНН 7735574607; КПП 773501001; ОГРН 5107746066192; ОКТМО 45927000

УТВЕРЖДАЮ

Первый заместитель Генерального аиректора – Главный Конструктор О «НИИ «Субмикрон» В.Ю. Гришин MM 7all 2022 r.

АКТ ВНЕДРЕНИЯ

результатов диссертационной работы Эннса Виктора Ивановича на соискание ученой степени доктора технических наук

«Методы и средства разработки специализированных гетерогенных конфигурируемых интегральных схем для вычислительной техники и систем управления»

Настоящим актом подтверждаем, что на предприятии АО «НИИ «Субмикрон» были использованы результаты диссертационной работы Эннса Виктора Ивановича. Предложенные в работе методы проектирования базовых матричных кристаллов и построения специализированной аппаратуры на их основе использованы при разработке «зашивок» и внедрении в аппаратуру микросхем базовых матричных кристаллов 5540TH014 для изделий СВУ и СВУ-М. Разработанные в диссертационной работе методы позволили ускорить создание аппаратуры, уменьшить ее массогабаритные параметры, улучшить характеристики, включая быстродействие и мощность потребления, и повысить надежность функционирования.

Заместитель Главного конструктор



АО «Микрон» УТВЕРЖДАЮ ОЗ СТВО ТО Микрон" Тенеральный директор

mikron

АО «Микрон»

24.02

Г.Ш. Хасьянова

2022г.

Акционерное общество «Микрон» ИНН: 7735007358, ОГРН: 1027700073466 ул.Академика Валиева, д.6, стр. 1, Зеленоград, Москва, 124460 тел.: 8 (800) 200-7129 факс: +7 (495) 229-7702 www.mikron.ru E-mail: mikron@mikron.ru

Акт внедрения

результатов диссертационной работы В.И. Эннса «Методы и средства разработки специализированных гетерогенных конфигурируемых интегральных схем для вычислительной техники и систем управления» на соискание ученой степени доктора технических наук

Настоящий акт свидетельствует 0 TOM, что результаты диссертационной работы Эннса Виктора Ивановича внедрены на предприятии АО «Микрон» и использованы в проектно-конструкторской документации микросхем серий 5510БЦЗТ, 5510ХСЗ, 5540TH028. 5540ТН014, 5400БК2У. Предложенные в диссертационной работе методы позволили улучшить технико-экономические показатели изделий и укомплектовать электронной компонентной базой важнейшую аппаратуру специального назначения.

Заместитель генерального директора-О. Ранчин