

# Оптимизация резервирования с разнесением чувствительных областей для сбоестойчивых систем на кристалле

П.О. Черняков<sup>1</sup>, Н.В. Желудков<sup>2</sup>, М.С. Ладнушкин<sup>3</sup>, А.А. Антонов<sup>4</sup>,  
В.Ю. Лазарев<sup>5</sup>

<sup>1</sup>ФГУ ФНЦ НИИСИ РАН, Москва, Россия, chernyakov@cs.niisi.ras.ru;

<sup>2</sup>ФГУ ФНЦ НИИСИ РАН, Москва, Россия, nvgel@cs.niisi.ras.ru;

<sup>3</sup>ФГУ ФНЦ НИИСИ РАН, Москва, Россия, maxsl@cs.niisi.ras.ru;

<sup>4</sup>ФГУ ФНЦ НИИСИ РАН, Москва, Россия, antonov@niisi.msk.ru;

<sup>5</sup>ФГУ ФНЦ НИИСИ РАН, Москва, Россия, lazarev@niisi.msk.ru

**Аннотация.** В данной работе описан модифицированный в сравнении с предыдущими разработками способ троирования с разнесением чувствительных областей цифровой синтезируемой логики, который позволяет добиться лучших показателей по занимаемой площади. Приведено сравнение характеристик троированных блоков, разработанных с использованием разных маршрутов. Результаты данной работы применимы в маршруте проектирования сбоестойчивых систем на кристалле и апробированы для тестового кристалла по технологии с проектными нормами 28 нм.

**Ключевые слова:** сбоестойчивость, тройное модульное резервирование, СнК, СБИС1.

## 1. Введение

Для современной электронной аппаратуры требуются всё более высокопроизводительные и функционально-сложные микроэлектронные изделия. Данная тенденция затрагивает и системы с повышенными требованиями к сбоестойчивости.

Одним из основных способов улучшения характеристик микросхем является переход на технологии с меньшими проектными нормами. Например, микроэлектроника для современных космических аппаратов разрабатывается уже по технологиям с проектными нормами 28 нм и меньше. При этом данная микроэлектроника должна быть стойкой к воздействию ионизирующего излучения, что означает стойкость к одиночным и дозовым эффектам.

Ситуация со стойкостью к накопленной дозе при уменьшении проектных норм зачастую не только не ухудшается, а даже улучшается, вследствие уменьшения объемов диэлектрика, в котором происходит накопление заряда.

Эффекты от одиночных событий можно условно разделить на отказы и сбои. К отказам в несиловой микроэлектронике относят в основном открытие паразитных тиристорных структур. К сбоям относят переключение ячеек памяти и триггеров в противоположное состояние и возникновение переходного процесса в комбинационных цепях.

С уменьшением проектных норм ситуация с

открытием паразитных тиристоров не ухудшается, так как это уменьшение зачастую сопровождается уменьшением напряжения питания.

А вот ситуация со сбоями значительно ухудшается с уменьшением проектных норм. Это связано с несколькими основными факторами:

- Уменьшение критического заряда, необходимого для переключения элемента в противоположное значение. Это ведет к тому, что частицы со всё меньшими линейными потерями энергии могут вызвать сбой. Также это означает, что частицы с одинаковыми линейными потерями энергии могут вызвать большее число сбоев.

- Увеличение плотности размещения, а как следствие накрытие большего числа элементов одной частицей, что в сочетании с уменьшением критического заряда ведет к росту вероятности множественных сбоев.

- Рост тактовой частоты, который ведет к росту вклада сбоев в комбинационной логике.

Также уменьшение проектных норм ведет к тому, что многие способы, ранее успешно применяющиеся для борьбы с одиночными эффектами, перестают работать. Например, применение ячеек DICE становится менее эффективным для норм ниже 100 нм. Применение некоторых разновидностей троирования (троирование только триггеров, троирование без разнесения) также становится неэффективным.

Эти факторы требуют уделять всё большее внимание защите от сбоев [1].

В составе систем на кристалле можно выделить готовые IP-блоки, например, блоки памяти, и синтезируемую логику. В данной работе основное внимание уделяется защите синтезируемой логики.

## 2. Применение троирования и предыдущие разработки

Одним из основных методов защиты от сбоев является модульное резервирование, то есть использование нескольких копий одного и того же блока. Наиболее распространено тройное модульное резервирование (ТМР) или троирование, то есть использование трех копий исходного блока. Хотя в некоторых случаях используются и резервирование с большей кратностью, а иногда только дублирование, когда достаточно определить лишь наличие расхождения.

Модульное резервирование может осуществляться как на уровне макроблоков, например, СФ-блоков или целых микросхем, так и на уровне отдельных библиотечных элементов.

Первый вариант требует дополнительной и зачастую нетривиальной логики синхронизации блоков в случае расхождения результатов. Также при резервировании на уровне макроблоков актуальна проблема накопления сбоев. Всё это делает резервирование на уровне отдельных комбинационных и последовательностных элементов более предпочтительным.

По полноте резервирования ТМР на уровне отдельных ячеек можно разделить на [2]:

1. Локальное ТМР (ЛТМР), когда троируются только триггеры;
2. Распределенное ТМР (РТМР), когда троируются триггеры и комбинационные элементы;
3. Глобальное ТМР (ГТМР), когда троируются триггеры, комбинационные элементы и дерево синхросигнала.

Для борьбы со множественными сбоями элементы дополнительно могут разноситься таким образом, чтобы при попадании частицы сбивалось не более одной копии элемента из трех.

Существуют различные способы организации троирования и разнесения в рамках маршрута проектирования микросхемы.

Во ФГУ ФНЦ НИИСИ РАН применяется и активно развивается свой маршрут троирования. Различные его вариации для технологий с различными проектными нормами уже описываются в статьях [3-4].

В общем случае схему троирования в рамках маршрута можно описать следующим образом. Сначала проводится стандартный синтез и получается нетлист без троирования. Далее с помощью скрипта троирования каждый триггер заме-

няется на специальную ячейку. Вся комбинационная логика троируется. Для переходов от нетривиальных частей (например, памяти или падов) к троированным и наоборот добавляются специальные конструкции-обертки (wrappers).

В зависимости от требуемого уровня сбоестойчивости, проектных норм целевой технологии, имеющейся библиотеки элементов и некоторых других факторов специальная ячейка, на которую заменяется триггер, может отличаться.

Маршрут проиллюстрирован на рисунках 1-3.

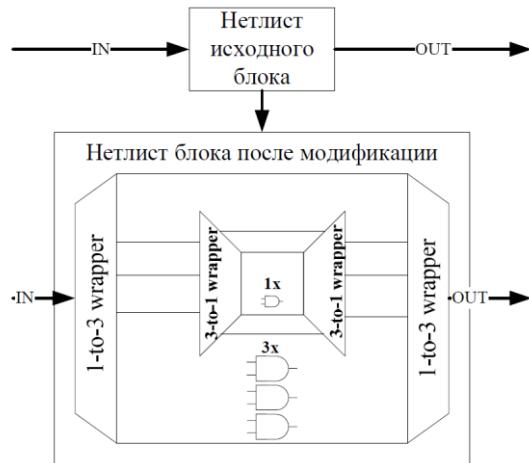


Рисунок 1. Применение троирования к исходному нетлистику [3]

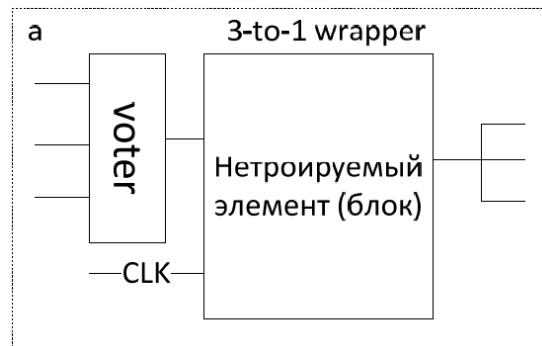


Рисунок 2. Интеграция нетривиального элемента [3]

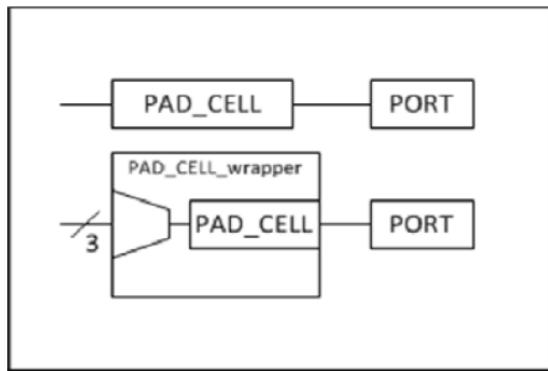


Рисунок 3. Сопряжение ячейки ввода/вывода с троиальным окружением [4]

### 3. Оптимизированный способ троирования с разнесением чувствительных областей

Для разработок по 65 нм использовалась ячейка, схематично представлена на рисунке 4. Аналогичная ячейка была подготовлена и для технологии 28 нм.

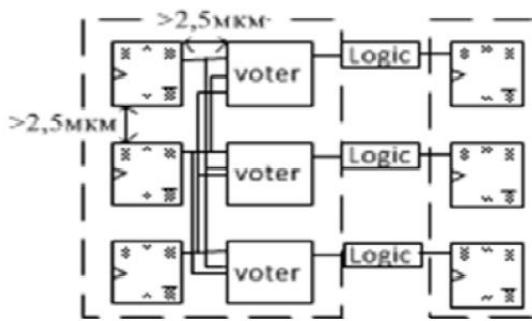


Рисунок 4. Схематичное изображение троированной ячейки для технологии 65 нм [4]

Она включает в себя три копии триггера и 3 копии воутеров. Все элементы разнесены между собой на расстояние не менее 2,5 мкм. Также есть как минимум половина расстояния разнесения между элементами и границами ячейки.

Один из недостатков данной ячейки в том, что пространство между разнесенными элементами в значительной степени пустует. И эта проблема становится более острой с уменьшением проектных норм, так как радиус разнесения должен сохраняться, а размеры элементов при этом уменьшаются. То есть большее пространство потенциально оказывается незадействованным.

Для разработок по 28 нм предложена модифицированная версия ячейки и скрипта троирования, которые позволяют значительно снизить потери площади.

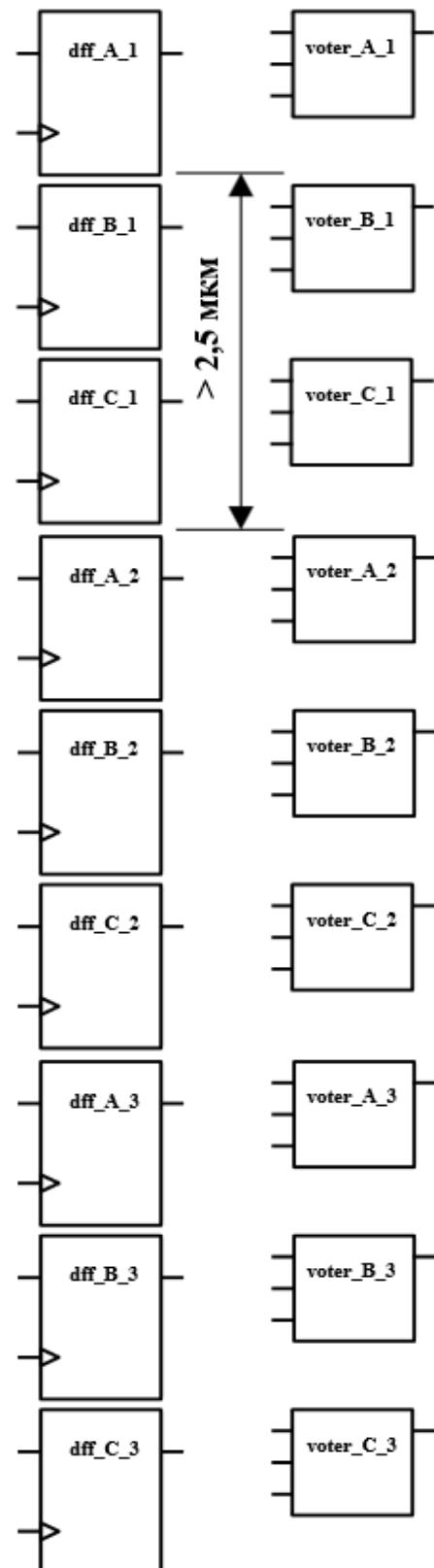


Рисунок 5. Схематичное изображение модифицированной ячейки

Новая ячейка состоит из 9 триггеров и 9 воутеров и позволяет сгруппировать в себе три троированных триггера с воутерами. Внутри ячейки триггеры и воутеры располагаются таким образом, чтобы три копии одного триггера располагались на расстоянии не менее 2,5 мкм. При этом триггеры, относящиеся к разным тройкам, могут находиться на меньшем расстоянии. Схематичное изображение новой ячейки приведено на рисунке 5.

Скрипт троирования был модифицирован, чтобы группировать триггеры по три и троировать их с использованием новой оптимизированной ячейки.

Такое расположение элементов позволяет повысить плотность размещения элементов и как следствие уменьшить площадь всей схемы. Что будет продемонстрировано далее на примере сравнения блоков, реализованных с использованием обычной и оптимизированной ячейки.

В таблице 1 приведена площадь оригинальной (dffrs\_tmr) и оптимизированной (dffrs\_tmr\_3) ячеек по 28 нм.

Таблица 1. Площадь ячеек

Ячейка	Площадь, мкм <sup>2</sup>
dffrs_tmr	119
dffrs_tmr_3	163

Как видно из таблицы площадь оптимизированной ячейки менее чем в полтора раза больше, при том что она включает в себя втрое больше элементов, чем оригинальная.

#### 4. Сравнение реализаций с различным способом организации троированных ячеек

Вначале оптимизированный маршрут троирования был опробован на простом блоке, состоящем из трех регистров, а также кодера и декодера расширенного кода Хэмминга между ними.

Таблица 2. Характеристики реализаций блока с кодером и декодером Хэмминга

Название	Частота, МГц	Мощность, мВт	Площадь, мм <sup>2</sup>
ham_code_decode	870	9,2	0,026
ham_code_decode (оптим)	928	9,5	0,014

Блок был синтезирован с использованием оригинального и оптимизированного маршрута для

технологии с проектными нормами 28 нм. Для обоих вариантов была подготовлена топология. Характеристики получившихся реализаций приведены в таблице 2. В качестве частоты указана получившаяся предельная частота.

Из таблицы видно, что для данного блока использование оптимизированного маршрута позволило достичь большей частоты и почти вдвое меньшей площади. Рост мощности связан с ростом частоты.

В качестве более сложного блока был выбран контроллер внешней статической памяти. Данный блок не содержит в своем составе каких-либо готовых макроблоков. То есть он полностью синтезируется с использованием элементов из библиотеки стандартных ячеек.

Для данного контроллера проводился только синтез по оригинальному и оптимизированному маршруту. Топология не разрабатывалась. Оценочные характеристики реализаций по результатам синтеза приведены в таблице 3.

Таблица 3. Характеристики реализаций контроллера памяти по результатам синтеза

Название	Частота, МГц	Мощность, мВт	Площадь, мм <sup>2</sup>
MemCtrl_top	400	26,17	0,394
MemCtrl_top (оптим)	414	26,23	0,188

Из таблицы видно, что использование оптимизированного маршрута для данного блока позволило добиться вдвое меньшей занимаемой площади, а также небольшого прироста частоты. Незначительное увеличение мощности также, как и в случае прошлого блока, вероятнее всего связано с ростом частоты.

Важно отметить, что и блок контроллера внешней статической памяти, и блок с регистрами, кодером и декодером Хэмминга не содержат в своем составе готовых блоков, таких как массивы памяти, пады и т.д. Если бы они их содержали, то выигрыш по площади был бы меньше, так как такие блоки не троируются. Следующий блок, рассматриваемый в данной статье, именно такой.

Данный маршрут использовался при разработке тестового кристалла (ТК) по технологии 28 нм.

В данном случае тестовый кристалл – это микросхема, содержащая заказные блоки (памяти, пады и т.д.), которые нужно проверить перед изготовлением СнК. Помимо заказных блоков опытный образец содержит синтезируемую логику, в основном необходимую для проверки заказных блоков.

Был проведен синтез RTL-модели тестового

кристалла с использованием оригинального и оптимизированного маршрутов. Результаты синтеза приведены в таблице 4.

Таблица 4. Характеристики реализаций тестового кристалла, полученные по результатам синтеза

Название	Частота, МГц	Мощность, мВт	Площадь, мм <sup>2</sup>	Площадь заказных блоков(не троируется),мм <sup>2</sup>
TK	192	862,3	7,83	5,07
TK (оптим)	198	857,7	6,36	5,07

Первое на что стоит обратить внимание – большую часть площади микросхемы занимают макроботки. То есть нетроируемая часть микросхемы. При этом всё равно вполне заметен выигрыш по площади при применении оптимизированного маршрута.

Если рассматривать только синтезируемую часть, то выигрыши по площади будет около двух раз, как и в предыдущих рассматриваемых блоках, что отражено в таблице 5.

Таблица 5. Анализ площади тестового кристалла

Название	Суммарная площадь, мм <sup>2</sup>	Площадь заказных блоков (не троируется), мм <sup>2</sup>	Площадь синтезируемой логики
TK	7,83	5,07	2,76
TK (оптим)	6,36	5,07	1,29

Из таблицы видно, что использование оптимизированного маршрута, позволяет достичь меньшей площади и немного повысить максимальную частоту. Исходя из этого именно оптимизированный маршрут был выбран для дальнейшей разработки.

Была разработана топология тестового кристалла, синтезированного с использованием оптимизированного маршрута. Характеристики получившейся топологической реализации приведены в таблице 6.

Таблица 6. Итоговые характеристики топологической реализации тестового кристалла

Название	Частота, МГц	Мощность, мВт	Площадь, мм <sup>2</sup>
TK	250	509,4	12,18

## 5. Анализ и результаты

Из характеристик получившихся реализаций видно, что оптимизированный маршрут троирования позволяет уменьшить итоговую площадь

микросхемы, а также немного повысить частоту.

При этом степень улучшения характеристик зависит от особенностей конкретного разрабатываемого блока, в первую очередь от наличия в нем памяти и других заказных блоков.

Изменение характеристик отражено в таблице 7. В ней для каждого блока приведены результаты оптимизированно маршрута, нормированные к результатам оригинального маршрута.

Таблица 7. Результаты оптимизированного маршрута, нормированные к результатам оригинального маршрута

Название	Частота, %	Мощность, %	Площадь, %
ham_code_decode (оптим)	106	103	53,85
MemCtrl_top (оптим)	103,50	100,22	47,72
TK (оптим), весь	103,13	99,47	81,22
TK (оптим), синтезируемая часть	-	-	46,74

Для синтезируемой логики выигрыш в площади от применения оптимизированного маршрута может составлять около 50%. Выигрыш в частоте не столь заметен и составляет единицы процентов.

Важно отметить, что способ группировки триггеров, использованный в данной работе, не обязательно является оптимальным. Этот аспект не изучался, потому что и предложенный способ позволил как уложиться в требования к опытному образцу, так и продемонстрировать лучшие характеристики в сравнении с оригинальным маршрутом.

Поиск и оценка других подходов к группировке триггеров является направлением дальнейшего исследования.

## 6. Заключение

В данной работе предложен оптимизированный с точки зрения организации троирования с разнесением маршрута разработки СБИС.

Оптимизация основана на использовании специальной ячейки, объединяющей три троированных триггера. Триггеры и воутеры в рамках этой ячейки расположены с чередованием таким образом, что копии, относящиеся к одному троированному триггеру, оказываются разнесены на расстояние не менее 2,5 мкм. При этом копии, относящиеся к разным доменам тро-

ирования могут находиться рядом. Такая организация ячейки позволяет значительно экономить место на кристалле. при разработке тестового кристалла по технологии с проектными нормами 28 нм.

Оптимизированный маршрут апробирован

## **Optimization of Redundancy with Separation of Sensitive Areas for Fault-tolerant System-on-chip**

**P. Chernyakov, N. Zheludkov, M. Ladnushkin, A. Antonov, V. Lazarev**

**Abstract.** This paper describes a modified compared to previous developments method of triple modular redundancy (TMR) with separation of sensitive areas of digital synthesized logic, which allows achieving better indicators in terms of occupied area. A comparison of the characteristics of blocks with TMR developed using different design flow. The results of this work are applicable in the design flow of fault-tolerant systems on a chip and were used in the development of a test chip using 28 nm technology node.

**Keywords:** fault tolerance, triple modular redundancy, SoC, VLSI

### **Литература**

1. P. E. Dodd, M. R. Shaneyfelt, J. R. Schwank, J. A. Felix. Current and future challenges in radiation effects on cmos electronics. “IEEE Transactions on Nuclear Science”, V. 57 (2010), No. 4, 1747–1763. ISSN:0018-9499. DOI: 10.1109/TNS.2010.2042613.
2. M. D. Berg, H. S. Kim, A. M. Phan, C. M. Seidleck, K. A. LaBel, J. A. Pellish, M. J. Campola. The effects of race conditions when implementing single-source redundant clock trees in triple modular redundant synchronous architectures. “Proceedings, 16th European Conference on Radiation and its Effects on Components and Systems (RADECS 2016)”, Germany, Bremen, 2016.
3. А.А. Антонов, А.О. Власов, Е.А. Гагарин. Применение троирования нетлиста в стандартном маршруте синтеза СБИС. «Труды НИИСИ», Т. 7 (2017), № 2, 120-124.
4. А.А. Антонов, А.О. Власов, Е.А. Гагарин, О.В. Мещерякова. Особенности разработки троированной СБИС по технологии 65нм. «Труды НИИСИ», Т. 8 (2018), № 3, 5-9.